

การชื้อน้ำประปาโดยใช้บัตรแม่เหล็ก

Purchase Water by Magnetic Card



โดย

นาย สุภชัย กาญจนเลขา 42015197

นาย อภิสิตท์ สีบวงส์ 42015204

นาย เอกรินทร์ เกษสุริยงค์ 42015207

อาจารย์ที่ปรึกษา

อาจารย์ อธิธิฐมิ บุญพิคำ

ปริญญาโทสำหรับปริญญาตรีวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมึ...  
เลขทะเบียน... 46250  
วัน, เดือน, ปี... 2 1 ส.ค. 2546

b.....  
i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

611289673

ปริญญาโท ปีการศึกษา 2544


ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การชื้อน้ำประปาโดยใช้บัตรแม่เหล็ก

( Purchase Water by Magnetic Card )

- ผู้จัดทำ
1. นาย สุภชัย กาญจนเลขา
  2. นาย อภิสิทธิ์ สืบวงศ์
  3. นาย เอกรินทร์ เกษสุริยงค์



*(Signature)*  
.....  
อาจารย์ที่ปรึกษา  
( *(Signature)* )


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การซื้อน้ำประปาโดยใช้บัตรแม่เหล็ก

(Purchase Water by Magnetic Card)

1. นาย สุภชัย กาญจนเลขา รหัส 42015197
2. นาย อภิสิตธิ์ สีบวงศ์ รหัส 42015204
3. นาย เอกรินทร์ เกษสุริยงค์ รหัส 42015207

โครงการนี้ได้รับการตรวจสอบแล้ว พร้อมทั้งจะทำการสอบได้

  
.....  
(อัครกัญญา กัญญา)  
.....  
อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

โครงการนี้สำเร็จลุล่วงไปด้วยดีเนื่องจากบุคคลหลายท่านโดยมี อ. อธิภูมิ บุญพิคำ อาจารย์ที่ปรึกษา ผู้ช่วยให้คำแนะนำแก้ไข ขอบคุณพี่ชาย พี่ ชัยณรงค์ คงสุนทร ที่ให้เครื่องมืออุปกรณ์สถานที่ทำงาน ขอบคุณอาจารย์ประจำภาควิชาอิเล็กทรอนิกส์ทุกท่านที่ให้คำปรึกษา และแนะนำ และขอบคุณเพื่อนๆทุกคนที่ให้คำแนะนำ เอกสารข้อมูล เป็นอย่างดี กระผมขอแสดงความขอบคุณทุกท่านมา ณ ที่นี้



นาย	ศุภชัย	กาญจนเลขา
นาย	อภิสิทธิ์	สีบวงส์
นาย	เอกรินทร์	เกษสุริยงค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การจำหน่ายน้ำประปาโดยใช้บัตรแม่เหล็ก

### Purchase Water By Magnetic Card

นาย ศุภชัย กาญจนเลขา

นาย อภิลิทธิ์ สีบวงศ์

นาย เอกรินทร์ เกษสุริยงค์

บทคัดย่อ

โครงการนี้เสนอเครื่องจำหน่ายน้ำประปาโดยใช้บัตรแม่เหล็ก บัตรนี้จะเก็บข้อมูลเกี่ยวกับรหัสมูลค่าของผู้ใช้ โดยใช้บัตรนี้ซื้อน้ำซึ่งสามารถระบุเป็นจำนวนลิตร จำนวนเงิน หรือ กดแบบ Manual (กดเปิด กดปิดได้)

นอกจากนี้ยัง Update ข้อมูลจากบัตร โดยสามารถเขียนข้อมูลจากบัตร โดยป้อนข้อมูลผ่านทาง keyboard ของชุดควบคุมนำเอาข้อมูลที่ป้อน ไปเก็บในรีจิสเตอร์ของเครื่อง

การทำงานของระบบประปານี้เครื่องอ่านบัตรแม่เหล็ก เมื่อข้อมูลของบัตรแม่เหล็กตรงกับข้อมูลที่เก็บไว้ในหน่วยความจำของเครื่องเครื่องจะทำงาน โดยการส่งสัญญาณไปเปิด ปิดน้ำโดยวาล์วโซลินอยด์ ปริมาณน้ำที่ไหลวัดจากมิเตอร์น้ำ อ่านค่าจากเซนเซอร์เพื่อมาควบคุมอีกครั้งหนึ่ง เมื่อน้ำที่จะจำหน่ายหมดจะแจ้งให้ทราบและคิดเงินตามจำนวนจริงที่จ่ายน้ำ และทุกครั้งที่จ่ายน้ำเสร็จจะแจ้งมูลค่าที่เหลือของบัตร สถานะการทำงานต่างๆผ่านทางจอแสดงผล LCD โดยขั้นตอนการทำงานต่างๆผ่านการประมวลผลจาก Microcontroller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## PURCHASE WATER BY MAGNETIC CARD

Mr. Supachai Kanjanalakha  
Mr. Apisit Suebwong  
Mr. Aekkarin Katsuriyong  
Mr. Ittibhoom Boonpikum ( Advisor )

### ABSTRACT

This project presents a construction of purchasing public water using a magnetic card. This card is used to purchase water in amount of liters , and credit top either with manual control (on/- off)

In addition , updating the information on the magnetic card can be done by writing new information of the card through a keyboard of a control unit which will store information in a register. After the magnetic card has been read and verified the information that matched the information of the system. The system is activated and send control signal to the control unit on/off solinoid valve. The amount of water is a monitored by a water meter that would also send signal to micro to update the credit left ,and amount of water that displayed on LCD.

## สารบัญ

เรื่อง		หน้า
บทที่ 1	บทนำ	1
	1.1 วัตถุประสงค์ของโครงการ	1
	1.2 ความเป็นมาของโครงการ	1
	1.3 ขอบเขตและขั้นตอนวิธีการดำเนินงาน	1
บทที่ 2	ทฤษฎีและหลักการ	3
	2.1 คุณลักษณะของวงจรแม่เหล็ก	4
	2.2 การบันทึกข้อมูลลงบนบัตรแม่เหล็ก	8
	2.3 รูปแบบการบันทึกข้อมูล	8
	2.4 การอ่านข้อมูลจากบัตรแม่เหล็ก	9
	2.5 คุณสมบัติของบัตรแม่เหล็ก ATM	10
	2.5.1 ตำแหน่งแทร็คที่สองของบัตรแม่เหล็ก	11
	2.5.2 ชุดรหัสข้อมูลในแทร็คที่สอง	11
	2.6 การบันทึกข้อมูลบนบัตรแม่เหล็ก	13
	2.7 เครื่องอ่านบัตรแถบแม่เหล็ก ของ Panasonic รุ่น ZU-M2121S451	15
	2.8 ทฤษฎีของไมโครคอนโทรลเลอร์ MCS – 51	20
	2.9 ทฤษฎีเกี่ยวกับ 8255	24
	2.10 ทฤษฎีเบื้องต้นเกี่ยวกับ MT8870	33
	2.11 ข้อมูลเกี่ยวกับ ไอซีเบอร์ ISD 2590	39
บทที่ 3	การออกแบบโครงการและการสร้าง	42
	3.1 คำนำ	42
	3.2 Hard Ware	42
	3.2.1 การออกแบบภาค Control	43
	3.2.2 การออกแบบ Key Board	46
	3.2.3 การต่อใช้งานเครื่องอ่านบัตรแถบแม่เหล็ก	47
	3.2.4 การต่อใช้งาน LCD	48
	3.2.5 การออกแบบภาคตรวจจับสัญญาณ Ringing และ ถอดรหัสความถี่ DTMF และระบบตอบกลับ	49
	3.2.6 การออกแบบหน่วยตรวจสอบการจ่ายน้ำ	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
3.3 SOFT WARE	54
<b>บทที่ 4</b> ผลการทดลอง	<b>65</b>
4.1 คำนำ	65
4.2 ภาค Control	65
4.3 ภาค INPUT	66
4.3.1 KEY BOARD	66
4.3.2 วงจรตรวจจับสัญญาณ RINGING	66
4.3.3 วงจรถอดรหัสสัญญาณความถี่ DTMF	66
4.4 ภาค Output	66
4.4.1 LCD Display (16 ตัวอักษร 2 บรรทัด )	66
4.4.2 ส่วนวงจรขับ โซลีนอยด์วาล์ว	67
4.4.3 วงจรสร้างเสียงตอบกลับทางโทรศัพท์	67
<b>บทที่ 5</b> สรุปและวิจารณ์ผลการทดลอง	<b>68</b>
<b>ภาคผนวก</b>	
ก. แสดงรูปเครื่องขายน้ําประปาโดยใช้บัตรแม่เหล็ก	
ข. แสดงวงจรและลายทองแดงของโครงการ	
ค. แสดงโปรแกรมควบคุมการทำงานของเครื่อง	
<b>เอกสารอ้างอิง</b>	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

	หน้า	
รูปที่ 2.1	วงจรของหัวบันทึก	5
รูปที่ 2.2	B-H Curve	5
รูปที่ 2.3	การเปลี่ยนแปลงเนื่องจากขนาดของ H	6
รูปที่ 2.4	ฮิสเทอรีซิส ลูปของเหล็ก	7
รูปที่ 2.5	การบันทึกข้อมูล	9
รูปที่ 2.6	การอ่านข้อมูล	10
รูปที่ 2.7	แสดงตำแหน่งแตร็คที่สองของแม่เหล็ก ATM ( ISO7811/5 )	11
รูปที่ 2.8	แสดงหัวบันทึกและแถบแม่เหล็กในขบวนการบันทึกข้อมูลบนแผ่นแม่เหล็ก	13
รูปที่ 2.9	แสดงสนามแม่เหล็กรอบแกนของหัวบันทึก	14
รูปที่ 2.10	แสดงกระแสพัลส์ รูปแบบของแตร็คที่ถูกบันทึก ฟลักซ์แม่เหล็กที่พื้นผิวแตร็คข้อมูลแลลักษณะแรงดัน ไฟฟ้าของการบันทึก	15
รูปที่ 2.11	ภาพแสดง ความสัมพันธ์ของสัญญาณ Data,Clock และ Present ของเครื่องอ่านแถบแม่เหล็ก	17
รูปที่ 2.12	แสดงโครงสร้างหลักของ MCS-51	21
รูปที่ 2.13	แสดงตำแหน่งขา MCS – 51	22
รูปที่ 2.14	วงจรภายในของ 8255	24
รูปที่ 2.15	แสดงตำแหน่งขาของ 8255	25
รูปที่ 2.16	แสดงรหัสการควบคุมพอร์ทของ8255	26
รูปที่ 2.17	แสดงลักษณะที่ set ในโหมด 1 เป็น Input	28
รูปที่ 2.18	แสดงลักษณะที่ set ในโหมด 1 เป็น Output	29
รูปที่ 2.19	แสดงลักษณะการทำ Hand Shake ในโหมด 1	30
รูปที่ 2.20	สัญญาณส่วน INPUTของ 8255 โหมด 1	30
รูปที่ 2.21	สัญญาณ OUTPUT ของ 8255 โหมด 1	31
รูปที่ 2.22	แสดงลักษณะการทำงานโหมด 2	32
รูปที่ 2.23	แสดงสัญญาณการทำ Hand Shake Mode 2	33
รูปที่ 2.24	แสดงโครงสร้างภายในของ MT8870	34
รูปที่ 2.25	แสดงความถี่ที่ได้จากภาคกรองความถี่	34
รูปที่ 2.26	แสดงวงจรตรวจสอบสัญญาณอย่างง่าย	36
รูปที่ 2.27	แสดงการต่อวงจรภาคอินพุท	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูปที่ 2.28 แสดงรายละเอียดขาของ MT8870	38
รูปที่ 2.29 แสดงวงจรใช้งานเบื้องต้นของ MT8870	38
รูปที่ 2.30 แสดงตำแหน่งขาของ ไอซีบันทึกเสียงเบอร์ ISD2590	39
รูปที่ 2.31 แสดง Block Diagram ในแต่ละภาคภายในตัวไอซีเบอร์ ISD 2590	40
รูปที่ 3.1 บล็อกไดอะแกรมแสดงการทำงานของโครงการ	43
รูปที่ 3.2 แสดงวงจรภาคส่วนควบคุม	44
รูปที่ 3.3 แสดงลักษณะการต่อ Keyboard ไปใช้งาน	46
รูปที่ 3.4 แสดงการต่อเครื่องอ่านบัตรแม่เหล็กไปใช้งาน	47
รูปที่ 3.5 แสดงการต่อ LCD ไปใช้งาน	48
รูปที่ 3.6 วงจรตรวจจับสัญญาณกริ่งเรียก	49
รูปที่ 3.7 วงจรถอดรหัสสัญญาณความถี่ DTMF	50
รูปที่ 3.8 วงจรสร้างเสียงตอบกลับโดยใช้ไอซีเบอร์ ISD2590	51
รูปที่ 3.9 แสดงวงจรการควบคุมวาล์วปิด/เปิดน้ำ	52
รูปที่ 3.10 แสดงลักษณะของ โซลินอยด์วาล์ว	52
รูปที่ 3.11 วงจรเซ็นเซอร์ปริมาณน้ำ	53
รูปที่ 3.12 Flow Chart แสดงการทำงานของระบบทั้งหมด	55
รูปที่ 3.13 Flow Chart แสดงการ SET ระบบเครื่อง	56
รูปที่ 3.14 Flow Chart แสดงการเลือกฟังก์ชันของการชื้อน้ำ	57
รูปที่ 3.15 Flow Chart แสดงการทำงานของโปรแกรมเซทระบบทางโทรศัพท์	58
รูปที่ 3.16 Flow Chart แสดงการทำงานของโปรแกรมการชื้อน้ำแบบระบุจำนวนเงิน	62
รูปที่ 3.17 Flow Chart แสดงการทำงานของโปรแกรมการชื้อน้ำแบบระบุจำนวนลิตร	63
รูปที่ 3.18 Flow Chart แสดงการทำงานของโปรแกรมการชื้อน้ำแบบ MANUAL	64

## สารบัญตาราง

		หน้า
ตารางที่ 2.1	ชนิดและการใช้งานบัตรแม่เหล็ก	3
ตารางที่ 2.2	ตัวอย่างการใช้งานบัตรแม่เหล็ก	4
ตารางที่ 2.3	แสดงรหัสข้อมูลตัวเลขสำหรับเทร็คที่สอง	12
ตารางที่ 2.4	แสดงข้อมูลที่ถูกบันทึกในบัตรแม่เหล็ก	15
ตารางที่ 2.5	แสดงข้อมูลของรหัส BCD สำหรับเครื่องอ่านแถบแม่เหล็ก	19
ตารางที่ 2.6	แสดงรหัสควบคุมกับการทำงานของพอร์ท	28
ตารางที่ 2.7	แสดงค่าที่ถอดรหัสดูได้จากความถี่ต่าง ๆ	35



## บทที่ 1

### บทนำ

#### 1.1 วัตถุประสงค์ของโครงการ

- 1.1.1 เพื่อนำเครื่องอ่านแถบบัตรแม่เหล็กไปประยุกต์ใช้งาน
- 1.1.2 ได้รับความรู้เกี่ยวกับข้อมูลที่บันทึกในบัตรแม่เหล็ก
- 1.1.3 เพื่อศึกษาเกี่ยวกับการออกแบบระบบที่ควบคุมด้วยไมโครคอนโทรลเลอร์

#### 1.2 ความเป็นมาของโครงการ

ในปัจจุบันเทคโนโลยีได้มีการพัฒนาไปอย่างมากและต่อเนื่อง การดำเนินชีวิตในปัจจุบันได้มีการนำเอาสิ่งอำนวยความสะดวกต่าง ๆ มาใช้เพื่อตอบสนองความต้องการของมนุษย์ ซึ่งหนึ่งในจำนวนนั้นมีการนำเอาบัตรแม่เหล็กมาใช้รวมอยู่ด้วย อาทิ เช่น บัตร ATM , บัตรเครดิตต่างๆ , บัตรเช็คเวลาในการทำงานของคนงานตามสถานประกอบการ และบัตรอื่นๆอีกมากมาย

โครงการนี้จึงเห็นความสำคัญของการศึกษากระบวนการทำงานของบัตรแม่เหล็ก เพื่อนำมาเป็นความรู้และพัฒนาระบบที่มีการนำบัตรแม่เหล็กมาใช้งาน โดยโครงการนี้จะสร้างวงจรที่นำเอาข้อมูลในบัตรแม่เหล็กที่อ่านได้จากเครื่องอ่านบัตรแม่เหล็กไปประยุกต์ใช้งานในการติดต่อหรือทำงานร่วมกับอุปกรณ์อื่นๆโดยจะใช้ไมโครคอนโทรลเลอร์เป็นตัวประมวลผลข้อมูล

#### 1.3 ขอบเขตและขั้นตอนวิธีการดำเนินงาน

ขั้นตอนการดำเนินงาน คือ

1. ศึกษาค้นคว้าข้อมูลที่เกี่ยวข้องกับโครงการ
  - การใช้งาน CPU MCS-51
  - การใช้งาน ไอซีขยาย พอร์ต 8255
  - การใช้งาน RAM
  - มาตรฐานของบัตรแม่เหล็กแบบต่างๆ
  - ศึกษาค้นคว้าข้อมูลในการนำเครื่องอ่านไปประยุกต์ใช้งาน

2. ทำการออกแบบวงจรต่างๆ และเขียนโปรแกรมควบคุม

- ออกแบบซอฟต์แวร์ในส่วน แสดงผล LCD , ส่วนควบคุม RELAY , การ INTERFACE กับ 8255 และ RAM
- ออกแบบ ฮาร์ดแวร์และซอฟต์แวร์ ในการประยุกต์ใช้บัตรแม่เหล็ก
- ออกแบบวงจรการสั่งงานผ่านระบบ โทรศัพท์
- ทดลองวงจรในส่วนต่างๆ



## บทที่ 2

### ทฤษฎีและหลักการ

บัตรแม่เหล็กที่ใช้กันอยู่ในปัจจุบัน มีด้วยกันมากมายหลายชนิด ดังแสดงไว้ในรูปที่ 2.1 ในยุคแรกๆ บัตรแม่เหล็กจะมีเนื้อวัสดุเป็นพลาสติกพีวีซีชนิดแข็งและแบบที่มีเนื้อวัสดุเป็นกระดาษ แต่ในปัจจุบันได้พัฒนาเนื้อวัสดุของบัตรแม่เหล็กมาเป็น Polyethylene Therphthalate หรือที่เรียกย่อๆ ว่า PET

ตารางที่ 2.1 ชนิดและการใช้งานบัตรแม่เหล็ก

ชนิด	เนื้อวัสดุ	ความหนาของบัตร	การใช้งาน
บัตรพลาสติก (Plastic Card)	-PVC ชนิดแข็ง -PVCA ชนิดแข็ง	0.76 มม.	-บัตร ATM -บัตรเครดิต -บัตรประจำตัว(ID)
PET Card	-Polyethylene Telephthalate(PET)	0.2 –0.4 มม.	-บัตรโทรศัพท์ -บัตรซื้อปั้งต่างๆ (Prepaid Card)
บัตรกระดาษ (Composite Paper Card)	-กระดาษอย่างดี -กระดาษเคลือบ พลาสติก -กระดาษบันทึกข้อมูล	0.2 –0.76 มม.	-ตั๋วรถไฟ -ตั๋วทางด่วน -ตั๋วเครื่องบิน -บัตรโรงแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตารางที่ 2.2 ตัวอย่างการใช้งานบัตรแม่เหล็ก

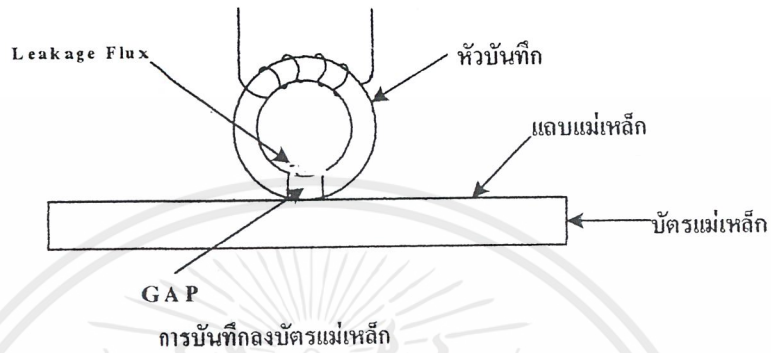
การใช้งาน	ชนิดของบัตร
การเงิน การธนาคาร	- บัตร ATM - บัตรเครดิต - บัตรเติมน้ำมัน - บัตรหลักทรัพย์ - หลักซื้อปิ้ง
การคมนาคม	- ตั๋วรถไฟแม่เหล็ก - ตั๋วผ่านทางด่วน - ตั๋วเครื่องบิน - ตั๋วจอดรถ
การสื่อสาร	- บัตรโทรศัพท์
การดำเนินงาน (OA,FA)	- บัตรประจำตัวพนักงาน - บัตรประจำตัวนักเรียน - บัตรเครื่องถ่ายเอกสาร - บัตรตรวจโรค - บัตรโรงแรม
การรักษาความปลอดภัย	- บัตรปิด และเปิดตู้กด

### 2.1 คุณลักษณะของวงจรมแม่เหล็ก

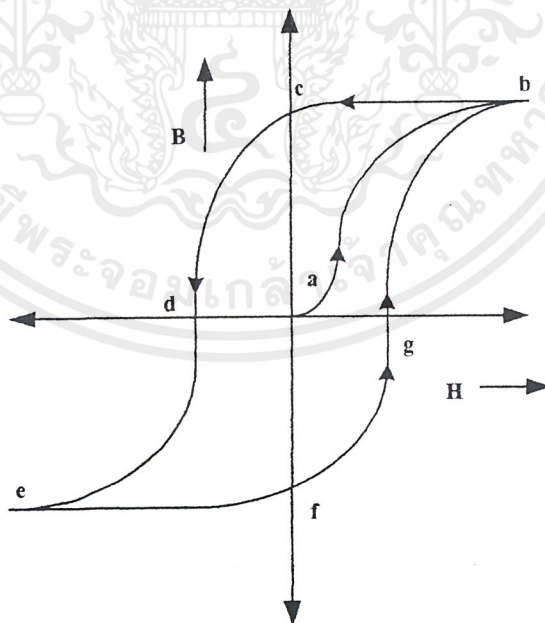
คุณลักษณะของวงจรมแม่เหล็กสามารถแสดงได้ด้วยกราฟดังรูปที่ 2.2 โดยเป็นการพล็อตระหว่าง "ฟลักซ์แม่เหล็ก" (Flux Density) B และ "สนามแม่เหล็ก" (Magnetizing Force) H ทั้งนี้ความสัมพันธ์ของ B-H เป็นความสัมพันธ์ระหว่างแรงดันกับกระแสฟลักซ์แม่เหล็ก หมายถึงจำนวนเส้นแรงแม่เหล็กต่อตารางนิ้วที่ผ่านพื้นที่หน้าตัดของแกน ส่วน Magnetizing Force นั้นเท่ากับจำนวนแอมแปร์-รอบ (ampere turn) คูณกับความยาว แกนในการทำงาน หากสนามแม่เหล็ก (H) มีค่าเพิ่มขึ้นทีละน้อยจากศูนย์ ฟลักซ์แม่เหล็ก B ก็จะค่อย ๆ เพิ่มขึ้นเป็นสัดส่วนด้วย (0-a-b) จนกระทั่งถึงจุดอิ่มตัวที่ b จากนั้นถ้าแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คร่อมขดลวดตกลงเป็นศูนย์ฟลักซ์แม่เหล็ก B จะไม่กลับไปที่ศูนย์ แต่มันจะลดลงตามเส้น โค้งจาก b-c และถึงจุด c เมื่อ H กลายเป็นศูนย์ โดยมีแมกนีจูดหรือขนาด 0-c



รูปที่ 2.1 วงจรของหัวบันทึก



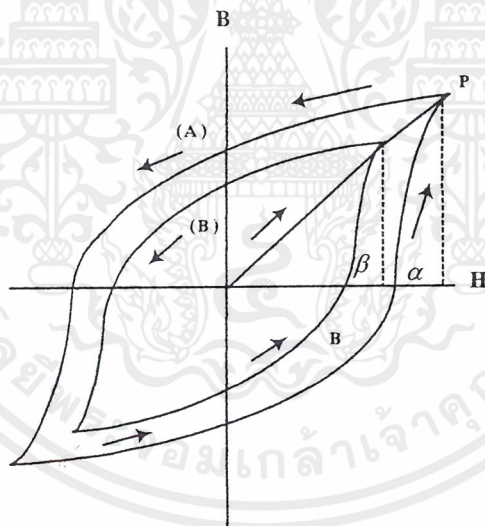
รูปที่ 2.2 B-H Curve

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าหากลองป้อนกระแสหรือ  $H$  ในทิศทางตรงกันข้ามกับขดลวด ค่า  $B$  จะเป็นไปตามโค้งจาก  $c-f-e$  ที่จุด  $e$  เป็นจุดที่มีการอ้อมตัวตรงข้ามกันของวัสดุแม่เหล็กจะเห็นได้ว่าหากเราป้อน แรงดัน ไฟฟ้า กระแสสลับเข้าที่ขดลวดตามรูปที่ 2.1 แล้วค่า  $B$  จะเดินทางเกิดเป็นวงรอบ หรือลูป (loop) ขึ้นคือ จาก  $0-a-b-c-d-e-f-g-b$  เราเรียกลูปดังกล่าวว่า "ฮิสเตอร์รีซิส"

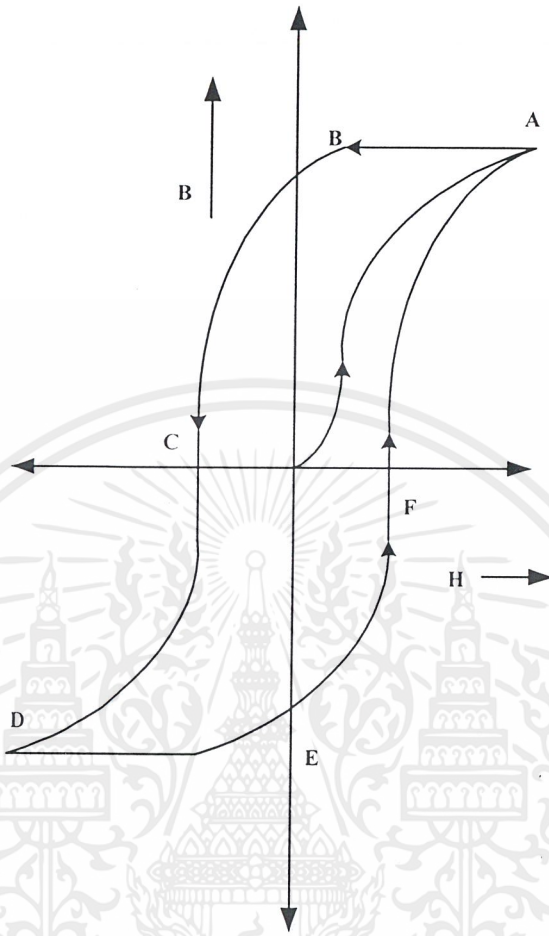
- ระยะห่างระหว่างจุดตัดแกน ( $0$ ) ถึง  $b$  เรียกว่า "Magnetization Curve ของวัสดุ"
- Closed loop เรียกว่า "Hystereses Loop"
- ค่าของ Flux density ( $B$ ) จาก  $0$  ถึง  $c$  เรียกว่า "retentivity ของ magnetic substance"
- ค่าของ magnetizing curve ( $H$ ) จาก  $0$  ถึง  $d$  เรียกว่า "coerecive force"

จากรูปที่ 2.2 นั้นหากปริมาณหรือขนาดของ  $H$  (ซึ่งป้อนเข้าไปครั้งแรก) หยดที่จุดที่มีค่าต่ำ ๆ จุดหนึ่งแล้วกลับ จะมีผลทำให้เกิดเป็นลูปเล็ก (เล็กกว่าครั้งแรก) ขึ้นดังแสดงดังรูปที่ 2.3



รูปที่ 2.3 การเปลี่ยนแปลงเนื่องจากขนาดของ  $H$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 ฮิสเทอริซิส ลูปของเหล็ก

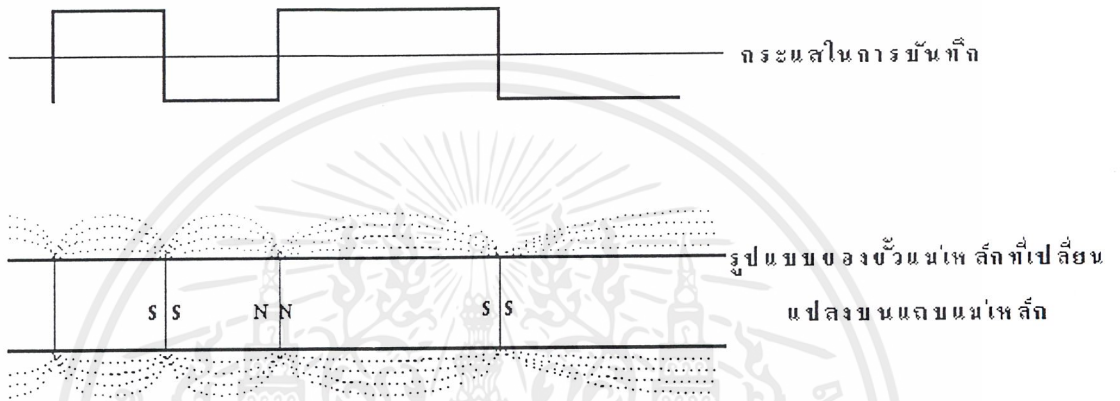
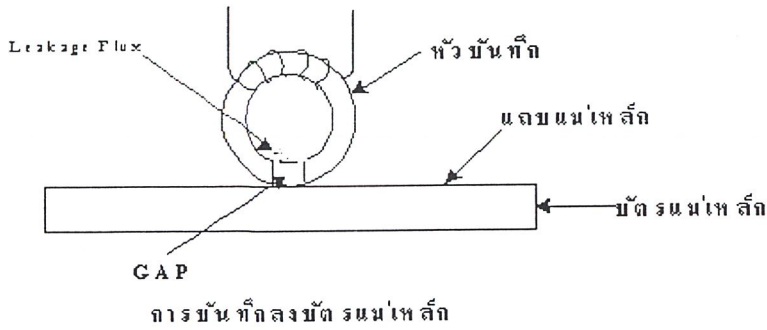
ในกรณีนี้หากป้อนแรงแม่เหล็กเข้าที่ตำแหน่งเริ่มแล้วแรงแม่เหล็ก (A) จะยังคงอยู่แม้ เอาแรงแม่เหล็กออกแล้วก็ตาม และหากป้อนสนามแม่เหล็กเข้าไป (B) ก็จะคงอยู่ด้วยเช่นกัน อันนี้จึงเป็นการอธิบายถึงหลักการจำ (memory) ของการบันทึกข้อมูลด้วย hysteresis phenomenon "B-H characteristic" อีกนัยหนึ่งก็คือ ฟังก์ชันความจำ (memory function) ของการบันทึกนั้นก็คือเปลี่ยนแปลงรูปคลื่นของข้อมูลที่บันทึกให้อยู่ในรูป ของฟลักซ์แม่เหล็ก ในกรณีนี้จะกล่าวในเรื่องเครื่องบันทึกบัตรแม่เหล็กจากที่กล่าวมาข้างต้นเป็นทฤษฎีพื้นฐานซึ่งเป็นหลักการของเครื่องบันทึกเทปต่างๆ ไป ซึ่งนำมาเป็นแนวทางสำหรับเครื่องอ่านและบันทึกแถบแม่เหล็กเท่านั้น โดยลำดับต่อไปจะกล่าวถึงการบันทึกข้อมูลลงบนบัตรแม่เหล็ก

## 2.2 การบันทึกข้อมูลลงบนบัตรแม่เหล็ก

การบันทึกข้อมูลลงบนบัตรแม่เหล็กจะใช้วิธีการบันทึกแบบดิจิทัล ในลักษณะเช่นเดียวกับที่ใช้ในแผ่นฟลอปปีดิสก์ หรือ เทปแม่เหล็กสำหรับเครื่องคอมพิวเตอร์ทั่ว ๆ ไป การบันทึกข้อมูลลงบนแถบแม่เหล็กนั้นจะต้องป้อนกระแสฟลัดซึ่งมีทั้งด้านบวกและด้านลบ พร้อมทั้งมีขนาดเพียงพอเข้าที่ขดลวดของหัวบันทึกซึ่งกดอยู่บนแถบแม่เหล็กที่เคลื่อนด้วยความเร็วคงที่ ดังในรูปที่ 2.5 (a) แถบแม่เหล็กจะเกิดเป็นแม่เหล็กถาวรขนาดเล็กเรียงตัวกันตามหัวบวกหรือลบของฟลัดและความกว้างของฟลัดสัญญาณที่บันทึกเนื่องจากกระแสฟลัดที่ใช้ในการบันทึกมีขนาดเพียงพอที่หัวบันทึก จะทำให้แถบแม่เหล็กมีสนามแม่เหล็กอ้อมตัวได้ดังนั้นเมื่อทำการบันทึกมีขนาดเพียงพอที่หัวบันทึก จะทำให้แถบแม่เหล็กอ้อมตัวได้ดังนั้นเมื่อนำการบันทึกข้อมูล ตัวข้อมูลที่เคยมีอยู่จะถูกเขียนทับและหายไป เหลือเพียง ข้อมูลใหม่เท่านั้น

## 2.3 รูปแบบการบันทึกข้อมูล

รูปแบบการบันทึกข้อมูลลงบนบัตรแม่เหล็กส่วนใหญ่ จะใช้รูปแบบ F2F หมายถึง Two Frequency Coherent Encoding และ FM หมายถึง Frequency modulation การบันทึกข้อมูลในลักษณะเช่นนี้จะบันทึกข้อมูล (Data) และ Clock เข้าไว้ในแทรคเดียวกัน นอกเหนือจากนี้ยังมีการบันทึกที่ข้อมูลและ Clock แยกคนละแทรค เช่นแบบ NRZI(Non Zero Inverted Recording) ซึ่งมีความจุในการบันทึกค่า

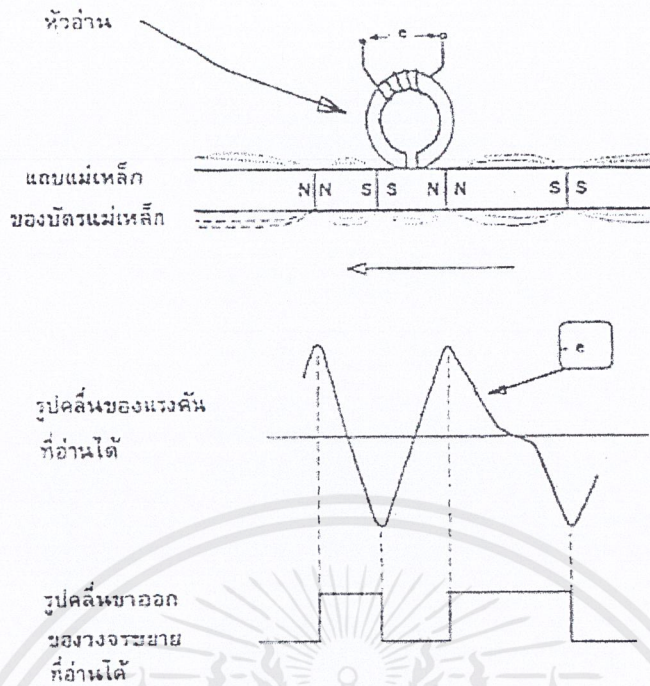


รูปที่ 2.5 การขั้วที่ข้อมูล

#### 2.4 การอ่านข้อมูลจากขั้วแม่เหล็ก

การอ่านข้อมูลจากขั้วแม่เหล็ก ทำได้โดยให้หัวอ่านสัมผัสกับแถบแม่เหล็กซึ่งเคลื่อนที่ด้วยความเร็วคงที่ในรูปที่ 2.6 ฟลักซ์ที่เกิดจากแม่เหล็กถาวรขนาดเล็กบนแถบแม่เหล็กจะผ่านจากเก็บของหัวอ่านไปยังแกน (Core) การเปลี่ยนแปลงของ Flux ตามข้อมูลนั้น

จุดสูงสุด (Peak) ของแรงดันที่อ่านได้นั้นจะตรงกับจุดที่สนามแม่เหล็กบนแถบแม่เหล็กกลับทิศทางพอดี ดังนั้นถ้าขยายแรงดันนี้ขึ้นและตรวจหาจุดสูงสุด (Peak) ด้วยวิธีดิฟเฟอเรนเชียลแล้วเปลี่ยนเป็นสัญญาณพัลส์ ก็จะได้ข้อมูลที่บันทึกอยู่ในขั้วแม่เหล็ก (รูปที่ 2.5)



รูปที่ 2.6 การอ่านข้อมูล

ในระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็กที่ได้ออกแบบไว้ ได้นำบัตรแม่เหล็กมาใช้งาน เป็นบัตรผ่านสำหรับการเข้าออกระบบ โดยใช้ข้อมูลบนบัตรแม่เหล็กเป็นข้อมูลประจำตัวของผู้ถือบัตร สำหรับการพัฒนาระบบในขั้นตอนนี้ ได้นำบัตร ATM ดังกล่าวมีรูปแบบของข้อมูลที่บันทึกอยู่บนแถบแม่เหล็กและคุณลักษณะต่างๆ ของบัตรแม่เหล็ก เป็นไปตามมาตรฐานสากล INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO) โดยรายละเอียดของบัตรแม่เหล็ก ATM จะนำเสนอในหัวข้อ 2.5 ส่วนรูปแบบการบันทึกข้อมูลลงบัตรแม่เหล็กนั้น ได้นำเสนอรูปแบบการอ่านหลายรูปแบบซึ่งจะกล่าวได้ถึงในรายละเอียดต่อไปนี้

## 2.5 คุณสมบัติของบัตรแม่เหล็ก ATM

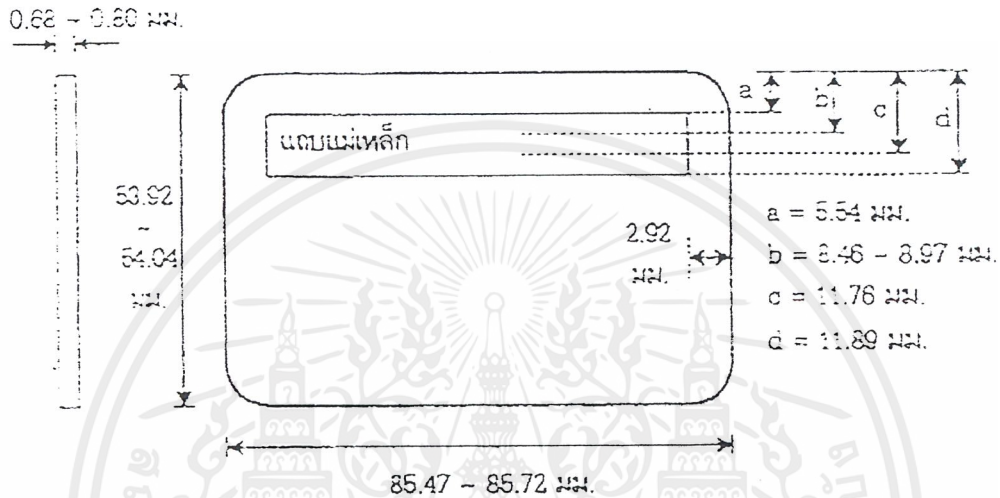
แถบแม่เหล็กของบัตรแม่เหล็ก ATM หรือบัตรเครดิตของธนาคารพาณิชย์ทั่วโลก จะมีแทร็คบันทึกข้อมูลจำนวน 3 แทร็ค โดยที่แทร็คที่สองของแถบแม่เหล็กจะบันทึกตัวเลขที่อ้างอิงกับหมายเลขบัญชีของผู้ถือบัตร เพื่อใช้เป็นข้อมูลคีย์สำหรับการคำนวณรหัส PIN ( Personal Identification Number ) ของบัตรแม่เหล็กใบนั้นๆ โดยข้อมูลในแทร็คที่สองนี้ จะเป็นตัวเลขชุดเดียวกับชุดตัวเลขที่ปั๊มอยู่บนบัตรแม่เหล็ก ATM หรือบนบัตรเครดิต เช่น บัตร ATM ของธนาคารกรุงเทพ จำกัด บัตรเครดิตของ AMEX, VISA และ MASTER เป็นต้น จากข้อมูลดังกล่าวจึงได้นำข้อมูลที่บันทึกอยู่ในแทร็คที่สองของแถบแม่เหล็กมาใช้งานเป็นข้อมูลประจำตัวของผู้ถือบัตร และขอเสนอรายละเอียดที่เกี่ยวข้องกับข้อมูลในแทร็คที่สองเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แถบแม่เหล็กมาใช้งานเป็นข้อมูลประจำตัวของผู้ถือบัตร และขอเสนอรายละเอียดที่เกี่ยวข้องกับข้อมูลในแทร็คที่สองเท่านั้น

### 2.5.1 ตำแหน่งแทร็คที่สองของบัตรแม่เหล็ก

ตำแหน่งแทร็คที่สองของบัตรแม่เหล็ก ATM เป็นไปตามมาตรฐาน ISO 7811/5 ดังแสดงในรูปที่ 2.7



รูปที่ 2.7 แสดงตำแหน่งแทร็คที่สองของแม่เหล็ก ATM (ISO7811/5)

จากรูปที่ 2.7 บริเวณแทร็คที่สองจะอยู่ระหว่าง

เส้นขนาน (b) ค่า 8.97 มิลลิเมตร (0.353 นิ้ว) (ค่าสูงสุด)

หรือ ค่า 8.46 มิลลิเมตร (0.333 นิ้ว) (ค่าต่ำสุด) กับ

เส้นขนาน (c) ค่า 11.76 มิลลิเมตร (0.463 นิ้ว)

### 2.5.2 ชุดรหัสข้อมูลในแทร็คที่สอง

ข้อมูลที่บันทึกในแทร็คที่สองของบัตรแม่เหล็กเป็นตัวเลขอย่างเดียว โดยที่ตัวเลข 1 ตัวจะประกอบด้วยบิตข้อมูลแบบ BCD 4 บิต และบิตพาริตี ISO ได้ระบุจำนวนข้อมูลสูงสุดที่สามารถบันทึกในแทร็คที่สองไว้ไม่เกิน 40 ตัว (รวมสัญลักษณ์เริ่มต้นและสิ้นสุด) ส่วนชุดรหัสข้อมูลตัวเลขแต่ละตัวสำหรับแทร็คที่สอง แสดงไว้ในตารางที่ 2.3

ตารางที่ 2.3 แสดงรหัสข้อมูลตัวเลขสำหรับแตร็คที่สอง

P	b4	b3	B2	B11	รหัส
1	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	2
1	0	0	1	1	3
0	0	1	0	0	4
1	0	1	0	1	5
1	0	1	1	0	6
0	0	1	1	1	7
0	1	0	0	0	8
1	1	0	0	1	9
1	1	0	1	0	A
0	1	0	1	1	B1
1	1	1	0	0	A
0	1	1	0	10	B2
0	1	1	1	1	A
1	1	1	1	0	B3

จากตารางที่ 2.3

A เป็นตำแหน่งของสัญลักษณ์ที่ใช้เฉพาะในระบบควบคุมทาวฮาร์ดแวร์

B เป็นสัญลักษณ์ของการเริ่มต้นของข้อมูล ( start sentinel )

C เป็นสัญลักษณ์ตัวแยกข้อมูล ( separator )

D เป็นสัญลักษณ์ของการสิ้นสุดของข้อมูล ( stop sentinel )

2.5.3 รูปแบบของข้อมูลที่บันทึกในแตร็คที่สองลงบนบัตรแม่เหล็ก ATM

ข้อมูลที่บันทึกในแตร็คที่สองลงบนบัตรแม่เหล็ก ATM ของธนาคารพาณิชย์ทั่วไป

SYN	B1	ข้อมูล	B2	ข้อมูล	B2	ข้อมูล	B3	LRC	SYN
-----	----	--------	----	--------	----	--------	----	-----	-----

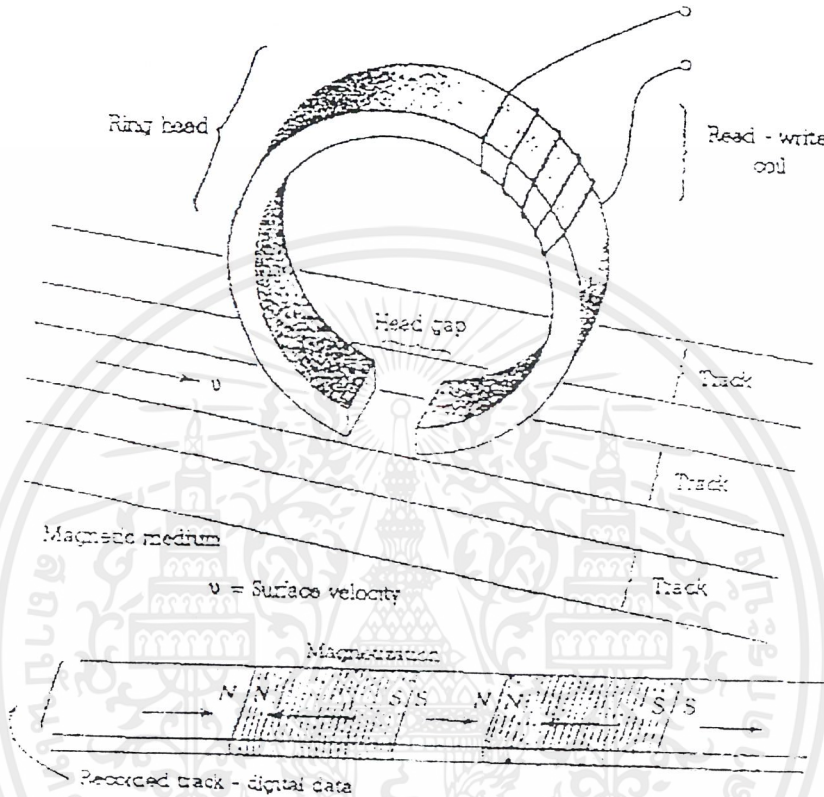
SYN : Synchronization character

LRC : Longitudinal redundancy check

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6 การบันทึกข้อมูลบนบัตรแม่เหล็ก

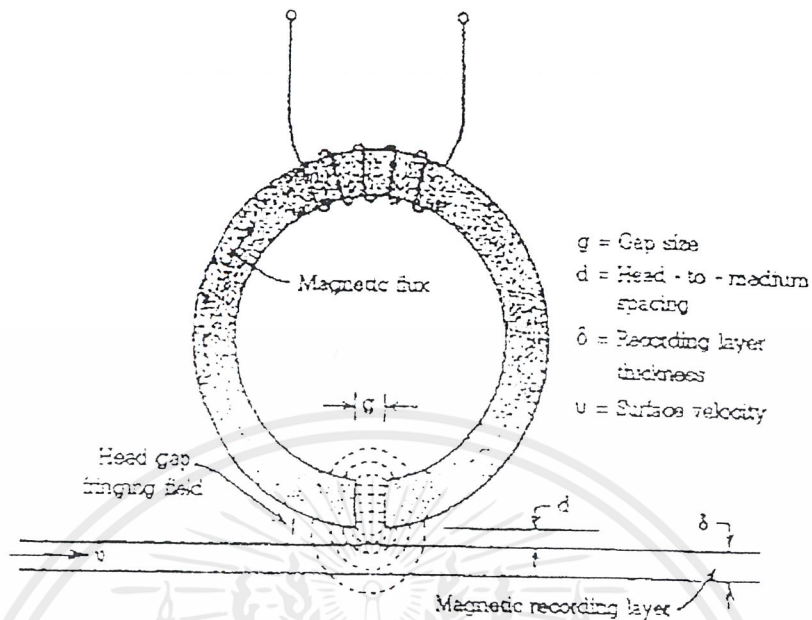
การบันทึกข้อมูลบนแถบแม่เหล็ก ประกอบด้วย แถบแม่เหล็กที่ใช้เป็นตัวกลางและหัวแม่เหล็กที่ใช้เป็นหัวบันทึก ซึ่งเคลื่อนที่ด้วยความเร็วคงที่ ดังแสดงในรูปที่ 2.8



รูปที่ 2.8 แสดงหัวบันทึกและแถบแม่เหล็กในขบวนการบันทึกข้อมูลบนแผ่นแม่เหล็ก

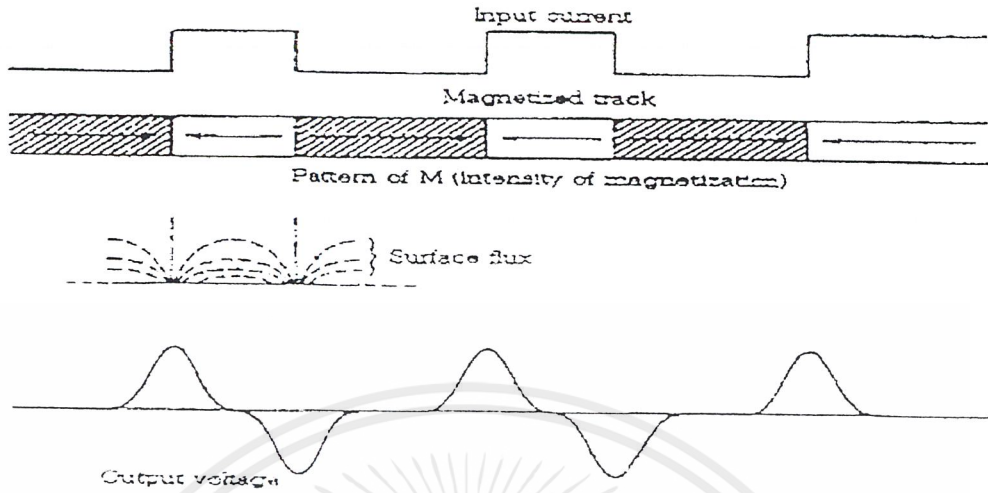
จากรูปที่ 2.8 การบันทึกข้อมูลลงบนแผ่นแม่เหล็ก จะขึ้นอยู่กับคุณสมบัติของสารแม่เหล็กในแถบแม่เหล็กและหัวแม่เหล็กที่จะนำมาใช้เป็นบันทึก โดยที่แกนของหัวบันทึกโดยทั่วไป จะเป็นแกนที่มีคุณสมบัติทางแม่เหล็กอย่างอ่อนๆ พันด้วยขดลวด และที่แกนมีช่องว่างอยู่ (gap) ส่วนบริเวณของสารแม่เหล็กจะถูกบันทึกที่ข้อมูลเรียกว่าแทร็ค (track) โดยแต่ละแทร็คจะเรียงขนานกันบนแถบแม่เหล็ก สัญญาณเอาต์พุตจากขดลวดที่พันรอบแกนจะเป็นสัดส่วนกับอัตราการเปลี่ยนแปลงฟลักซ์ของหัวบันทึก และความกว้างของแทร็ค การบันทึกข้อมูลลงบนบัตรแม่เหล็กจะใช้วิธีการป้อนกระแสพัลส์ทั้งด้านบวกและด้านลบ พร้อมทั้งมีขนาดเพียงพอเข้าที่ขดลวดของหัวบันทึกที่วางอยู่ใกล้กับแถบแม่เหล็ก เมื่อป้อนกระแสพัลส์จะก่อให้เกิดสนามแม่เหล็กรอบบริเวณแก๊ปของแกนหัวบันทึก ซึ่งสนามแม่เหล็กนี้ จะใช้ในการบันทึกข้อมูล แล้วแสดงในรูปที่ 2.9 ส่วนรูปแบบของกระแสพัลส์ แรงดันไฟฟ้า และหัวแม่เหล็กในแถบแม่เหล็ก เมื่อมีการบันทึกแล้วแสดงในรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 แสดงสนามแม่เหล็กรอบแก็ปของหัวบันทึก

จากรูปที่ 2.9 ค่าความกว้างแก็ปต้องมีค่าน้อยกว่าความกว้างของแตรีกข้อมูลเสมอ โดย ISO ได้ระบุความกว้างแก็ปของหัวบันทึกไว้ประมาณ 0.00625 มิลลิเมตร (0.00025 นิ้ว) หรือน้อยกว่า และค่าแก็ปของหัวอ่านมีค่าประมาณ 0.025 มิลลิเมตร (0.001 นิ้ว) หรือน้อยกว่า สำหรับค่าความหนาของเนื้อแถบแม่เหล็ก ( $\delta$ ) ISO ระบุไว้มีค่าไม่เกิน 0.0388 มิลลิเมตร



รูปที่ 2.10 แสดงกระแสพัลส์ รูปแบบของแทร็คที่ถูกบันทึก ฟลักซ์แม่เหล็กที่พื้นผิวแทร็คข้อมูลและลักษณะแรงดันไฟฟ้าของการบันทึก

2.7 เครื่องอ่านบัตรแถบแม่เหล็ก (Magnetic Card) ของ Panasonic รุ่น ZU-M2121S451

Magnetic Card หรือ บัตรแถบแม่เหล็ก ที่เรารู้จักกันดีในรูปแบบของบัตรเงินสด บัตร ATM บัตรเครดิต บัตร VISA และบัตรอื่นๆอีกมากมาย ฯลฯ เป็นการ์ดแถบแม่เหล็ก ซึ่งบนตัวบัตรเองจะบันทึกข้อมูลและรายละเอียดต่างๆของบัตร ไว้ในรูปของเส้นแรงแม่เหล็กภายในส่วนที่เป็นแถบแม่เหล็กในการ์ด โดยที่แถบแม่เหล็กที่อยู่บนการ์ด ซึ่งเรียกว่า "Track" นั้น ตามปกติแล้วจะถูกแบ่งออกเป็น 3 ส่วนด้วยกัน ซึ่งในแต่ละส่วนจะใช้เก็บข้อมูลซึ่งมีความหนาแน่นและลักษณะของข้อมูลที่มีความแตกต่างกันดังนี้

ตารางที่ 2.4 แสดงข้อมูลที่ถูกบันทึกในบัตรแม่เหล็ก

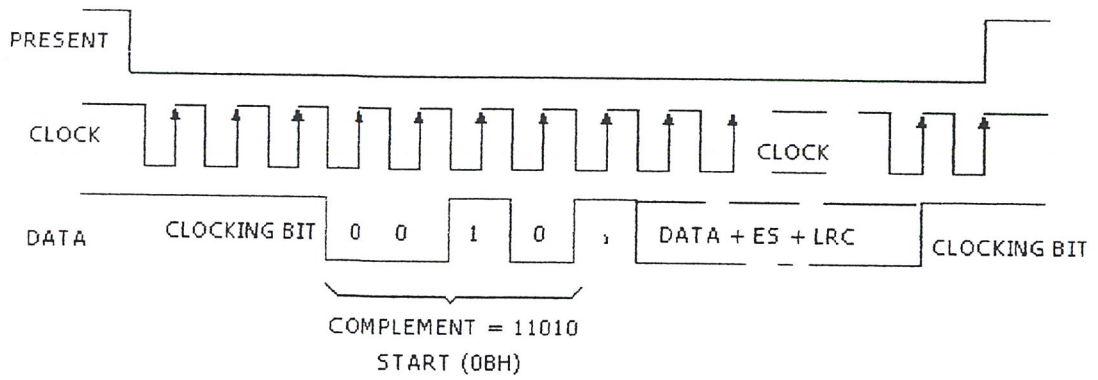
แถบแม่เหล็กที่	ความหนาแน่นในการบันทึก	การเข้ารหัสข้อมูล	จำนวนตัวอักษรสูงสุด	ลักษณะของข้อมูลที่เก็บในบัตร
Track 1	210 BPI	ALPHA	79	ชื่อเจ้าของบัตรและหมายเลขบัตร
Track 2	75 BPI	BCD	40	หมายเลขบัตรและวันหมดอายุ
Track 3	210 BPI	BCD	107	หมายเลขบัตรและรหัสพิเศษ

\*\*\*หมายเหตุ\*\*\*ความหนาแน่นของการบันทึกข้อมูลมีหน่วยเป็น BPI (Byte Per Inch)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้ในการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเครื่องอ่านแถบแม่เหล็ก ของ Panasonic รุ่น ZU-M2121S451 นี้จะอ่านได้เฉพาะข้อมูล ที่บันทึกด้วยความละเอียด 75 BPI คือ เฉพาะในส่วนขอแถบแม่เหล็กแถบที่ 2 (Track 2) เท่านั้น โดยมีสัญญาณที่ใช้เชื่อมต่อเพื่ออ่านข้อมูลจากเครื่องอ่านแถบแม่เหล็กทั้งหมด 5 สัญญาณ โดยเป็นสัญญาณเอาต์พุตส่งออกมาจากเครื่องอ่านแถบแม่เหล็กทั้งหมด และมีขาไฟเลี้ยงวงจรที่ต้องต่อให้เครื่องอ่านอีก ดังนี้

1. ไฟเลี้ยงวงจร (+VCC) ซึ่งต้องมีค่าเป็น +5VDC ซึ่งต่อจากภายนอกให้เครื่องอ่านแถบแม่เหล็ก
2. สัญญาณข้อมูล (Data) เป็นสัญญาณข้อมูลเอาต์พุตส่งออกมาจากเครื่องอ่านแถบแม่เหล็ก โดยการอ่านสัญญาณข้อมูลนี้ต้องอ่าน แบบ อนุกรม ทีละบิต โดยต้องพิจารณาให้สัมพันธ์สอดคล้องกับสัญญาณนาฬิกา(Clock) ของบิตด้วย ซึ่งสัญญาณข้อมูลที่อ่านได้นี้จะกลับสภาวะกับสัญญาณจริง คือมีสภาวะเป็นตรงข้ามกันอยู่ ดังนั้น เมื่ออ่านสัญญาณจากเครื่องอ่านแถบแม่เหล็กได้แล้วจะต้องกลับสภาวะของสัญญาณ (Complement) เสียก่อนจึงจะได้ข้อมูลที่ถูกต้อง
3. สัญญาณนาฬิกา (Clock) เป็นสัญญาณ Clock เอาต์พุตส่งออกมาจากเครื่องอ่านแถบแม่เหล็ก ใช้เป็นสัญญาณอ้างอิงในการอ่านสัญญาณข้อมูลจากบิต โดยการอ่านสัญญาณข้อมูลแต่ละบิตนั้น ต้องอ่านในขณะที่สัญญาณนาฬิกาเป็นขอบขาขึ้น (Rising Edge) เสมอ ซึ่งสัญญาณข้อมูลที่อ่านได้นั้นจะเริ่มต้นจากบิตที่มีนัยสำคัญต่ำสุด(LSB) ก่อนเป็นอันดับแรก
4. สัญญาณ Present เป็นสัญญาณสถานะเอาต์พุตส่งออกมาจากเครื่องอ่านแถบแม่เหล็ก ทำงานที่ลอจิกศูนย์เมื่อสัญญาณนี้เกิด ("0") จะบ่งบอกให้ทราบว่าเครื่องอ่านแถบแม่เหล็กเริ่มต้นส่งข้อมูลออกมา หรือมีการนำบัตรแม่เหล็กไปรูดผ่านหัวอ่านสัญญาณของเครื่อง เมื่อสัญญาณนี้หมดลง (กลับเป็นลอจิกหนึ่ง) อีกครั้งหนึ่งแสดงว่าข้อมูลจากการอ่านในครั้งนั้นถูกส่งออกไปหมดสิ้นเรียบร้อยแล้ว ซึ่งเราอาจใช้ประโยชน์จากสัญญาณนี้ ส่งเป็นสัญญาณ Interrupt ให้ CPU เพื่อทำการอ่านข้อมูล



รูปที่ 2.11 ภาพแสดง ความสัมพันธ์ของสัญญาณ Data, Clock และ Present ของเครื่องอ่านแถบแม่เหล็ก

5. สัญญาณ Card In Detect เป็นสัญญาณอุปกรณ์ตรวจจับ ปกติมีค่าเป็น "1" แต่เมื่อมีการสอดใส่บัตรเข้าไปยังช่องรับบัตร สัญญาณนี้จะมีค่าเป็น "0" ค้างไว้ตลอด และจะกลับมีค่าเป็น "1" อีกครั้งก็ต่อเมื่อดึงบัตรออกจากช่องรับบัตรเรียบร้อยแล้ว

6. สัญญาณ End Stop เป็นสัญญาณจากอุปกรณ์ตรวจจับที่อยู่ภายในช่องรับบัตรด้านในสุด โดยปกติมีค่าเป็น "1" แต่เมื่อมีการเสียบใส่บัตรเข้าไปยังช่องรับบัตรจนสุด ก็จะทำให้สัญญาณนี้มีค่าเป็น "0" ค้างไว้ตลอด และจะกลับมีค่าเป็น "1" อีกครั้งก็ต่อเมื่อดึงบัตรกลับออกมา

7. Ground เป็นจุดอ้างอิงของสัญญาณต่างๆ ระหว่างเครื่องอ่านแถบแม่เหล็กกับบอร์ดไมโครฯ การอ่านข้อมูลจากบัตรแถบแม่เหล็ก

การใช้เครื่องอ่านแถบแม่เหล็กเพื่อให้อ่านข้อมูลจากบัตรแถบแม่เหล็กนั้น ต้องสอดใส่บัตรเข้าไปในช่องรับบัตรจากแถบด้านหน้า ตามทิศทางและตำแหน่งที่กำหนดไว้เท่านั้น โดยให้สังเกตตำแหน่งของหัวอ่านกับส่วนที่เป็นแถบแม่เหล็กของบัตรคืออยู่ในแนวเดียวกัน สำหรับการอ่านข้อมูลจากบัตรนั้นขอแนะนำให้อ่านในขณะที่สอดบัตรเข้าไปในช่องรับบัตรเท่านั้น เพราะถ้าไปอ่านในขณะที่ดึงบัตรกลับออกมาจะทำให้ได้ข้อมูลที่ถอยหลังกลับ ซึ่งจะทำได้ยากแก่การถอดรหัสที่ได้จากการอ่าน และเนื่องจากเครื่องอ่านแถบแม่เหล็กสามารถอ่านข้อมูลจากบัตรแถบแม่เหล็กได้เฉพาะข้อมูลของ Track2 เท่านั้น ดังนั้นจึงขอกว่าถึงเฉพาะวิธีการอ่านข้อมูลจาก Track2 เพียงอย่างเดียว โดย format ของข้อมูลในแถบแม่เหล็กในส่วนที่ 2 (Track2) ซึ่งใช้บันทึกสัญญาณด้วยความหนาแน่นของสนามแม่เหล็ก 75 BPI ซึ่งจะบรรจุจำนวนรหัสของข้อมูลต่างๆ ใน Track นี้ ได้สูงสุดไม่เกิน 40 ตัวอักษร โดยนับรวมรหัสควบคุมและรหัสตรวจสอบด้วยซึ่งมีรายละเอียดดังนี้คือ

SS	PAN	FS	Addition Data	ES	LRC
----	-----	----	---------------	----	-----

SS = รหัสของข้อมูลแสดงการเริ่มต้นของข้อมูลในบิต (Start Sentinel) มีรหัสข้อมูลเป็น 0BH (;)

PAN = ข้อมูลแสดงหมายเลขของบิต (มีจำนวนข้อมูลในส่วนนี้สูงสุดไม่เกิน 19 หลัก)

FS = รหัสข้อมูลแสดงการแบ่งแยกข้อมูล (Field Separator) มีรหัสข้อมูลเป็น 0DH(=)

Additional Data = เป็นข้อมูลเพิ่มเติมอื่นๆของบิต เช่น เดือน/ปี ที่ออกบิตและหมดอายุ

ES = รหัสข้อมูลแสดงการสิ้นสุดของข้อมูลในบิต (End Sentinel) มีรหัสข้อมูลเป็น 0FH (?)

LRC = ข้อมูลตรวจสอบความผิดพลาด (Longitudinal Redundancy Check)

0000000000	SS	Data , Data , Data , ...,Data	ES	LRC	0000000000
------------	----	-------------------------------	----	-----	------------

จากภาพซึ่งใช้แสดง ลักษณะการจัดเรียงของข้อมูลที่เก็บไว้ในแถบแม่เหล็กในส่วนของ Track2 ซึ่งจัดเรียงลำดับความสำคัญจากซ้ายไปขวาแบบอนุกรมทีละบิต โดยจะสังเกตเห็นว่ามีข้อมูลซึ่งเป็น โลกิก ศูนย์ หรือที่เรียกว่า "Clocking Bit" นำหน้าและปิดท้ายข้อมูลจริงอยู่ ซึ่งลักษณะการจัดเก็บข้อมูลของ Track2 จะเก็บข้อมูลด้วยการเข้ารหัสแบบ "Modul5" ซึ่งในแต่ละชุดข้อมูลจะประกอบด้วยข้อมูล 5 บิต โดย 4 บิตแรก (D0..D3) เป็นรหัสข้อมูลแบบ BCD ส่วนบิตที่ 5 (D4) เป็นพาริตีบิตแบบคี่ (ODD) เพื่อใช้ตรวจสอบความถูกต้องของข้อมูลในแต่ละไบต์ (5-Bit) ที่อ่านได้ ซึ่งหากค่าของบิตพาริตีผิดพลาด แสดงว่าการอ่านข้อมูลนั้นล้มเหลว โดยเริ่มทำการอ่านข้อมูลเมื่อสัญญาณ Present เริ่มเป็นศูนย์ก่อน และทำการอ่านข้อมูลในทุกๆรอบขาขึ้นของสัญญาณนาฬิกาเสมอ ซึ่งข้อมูลในส่วนก่อนเริ่มต้นและหลังจากสิ้นสุดของการอ่านนี้จะมีค่าเป็นศูนย์ ( สัญญาณ Data = Logic "1" ) นำหน้าและปิดท้ายข้อมูลจริงอยู่ ซึ่งเรียกว่า "Clock Bit" ซึ่งข้อมูลในส่วนนี้เราไม่ต้องสนใจ แต่ให้ตรวจสอบและรองจนกว่าจะเริ่ม เป็นข้อมูล Start บิต (0BH หรือ 11010 ในที่นี้เรียงลำดับความสำคัญจากซ้ายไปขวา ซึ่งบิตเริ่มต้น ของรหัส Start หรือ 0BH ต้องเริ่มต้นด้วย 1 เป็นบิตแรกเสมอ) ดังนั้นในการอ่านเราต้อง รองจนกว่าจะพบสัญญาณข้อมูลมีค่าเป็น "1" (สัญญาณ Data = Logic "0" เพราะกลับสภาวะกันอยู่) จึงเริ่มเก็บ ข้อมูลชุดละ 5 บิต ไปเรื่อยๆจนถึงรหัส จบ(0FB) ซึ่งเมื่อพบรหัสจบแล้วจะต้องมีข้อมูลตามมาอีก 1 ไบต์ ซึ่งเป็นข้อมูลสำหรับตรวจสอบความผิดพลาดของข้อมูลการอ่านทั้งหมดใน Track2 เรียกว่า "LRC" ซึ่งค่าของ "LRC" สามารถหาได้จากการนำเอาข้อมูลในแต่ละ ไบต์ (ไม่คิดพาริตีบิต) ตั้งแต่เริ่มต้น จนถึงสุดมาทำการ XOR กับศูนย์ แล้วนำผลลัพธ์ที่ได้ไป XOR กับไบต์ถัดไปเรื่อยๆตามลำดับ ซึ่งหากผลลัพธ์สุดท้ายที่ได้ไม่เท่ากับ ค่าของ "LRC" ที่อ่านมาได้แสดงว่าการอ่านข้อมูลทั้งหมดล้มเหลว

สำหรับข้อมูลรหัส BCD ที่ใช้สำหรับอ่านแถบแม่เหล็กนี้จะเป็นข้อมูลชุดละ 5 บิต โดยเป็นข้อมูลจริง 4 บิต และเป็นรหัสตรวจสอบพาริตีอีก 1 บิต ซึ่งมีจำนวนทั้งหมด 16 อักขระ ดังนี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.5 แสดงข้อมูลของรหัส BCD สำหรับเครื่องอ่านแถบแม่เหล็ก

Parity	D3	D2	D1	D0	Character	Function
1	0	0	0	0	0 (0H)	Data
0	0	0	0	1	1 (1H)	Data
0	0	0	1	0	2 (2H)	Data
1	0	0	1	1	3 (3H)	Data
0	0	1	0	0	4 (4H)	Data
1	0	1	0	1	5 (5H)	Data
1	0	1	1	0	6 (6H)	Data
0	0	1	1	1	7 (7H)	Data
0	1	0	0	0	8 (8H)	Data
1	1	0	0	1	9 (9H)	Data
1	1	0	1	0	: (AH)	Control
0	1	0	1	1	; (BH)	Start Sentinel
1	1	1	0	0	< (CH)	Control
0	1	1	0	1	= (DH)	Field Separator
0	1	1	1	0	> (EH)	Control
1	1	1	1	1	? (FH)	End Sentinel

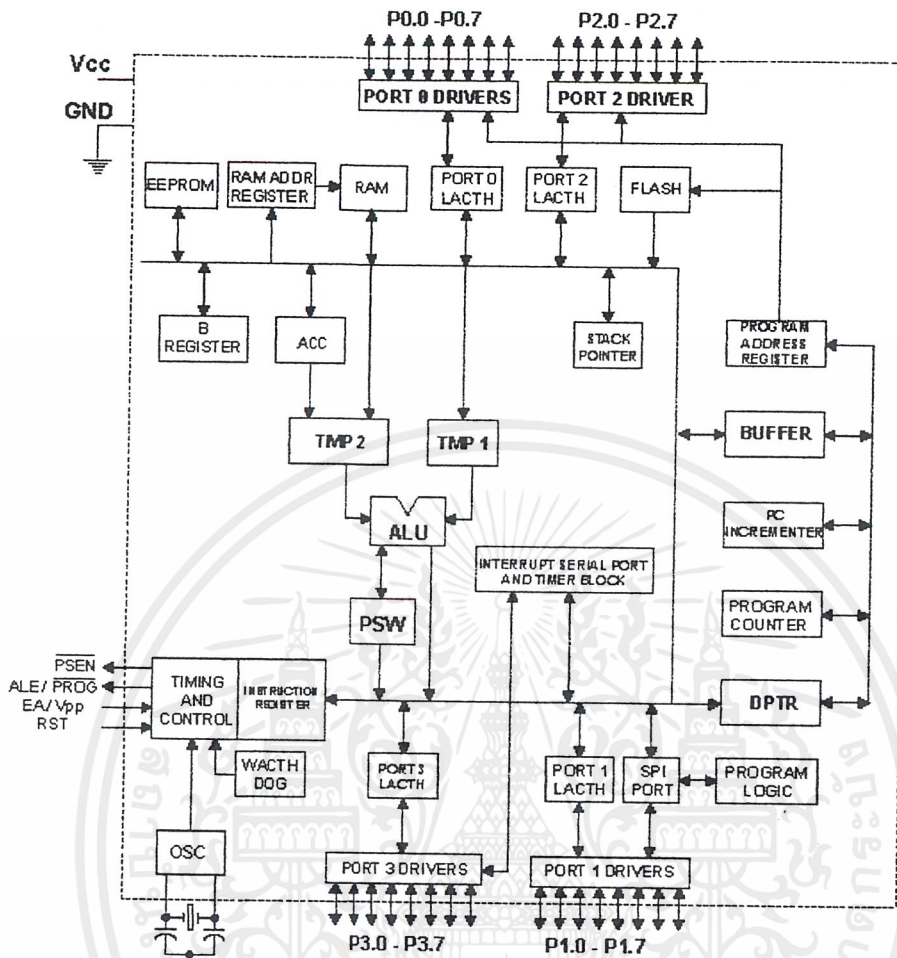
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.8 ทฤษฎีของไมโครคอนโทรลเลอร์ MCS – 51

ไมโครคอนโทรลเลอร์ MCS – 51 ประกอบด้วย ไมโครคอนโทรลเลอร์หลายรุ่นซึ่งมีสถาปัตยกรรมที่เหมือนกัน เพียงแต่มีขนาดหรือจำนวนหน่วยการทำงานภายในที่แตกต่างกัน อาทิเช่น ใช้เทคโนโลยีในการผลิตที่แตกต่างกัน เป็นต้น

ไมโครคอนโทรลเลอร์เบอร์ที่นับว่าเป็นเบอร์พื้นฐานสำหรับตระกูล MCS – 51 นี้ ได้แก่ เบอร์ 8051 8031 โดยเบอร์ 8051 จัดเป็นสมาชิกตัวแรกของตระกูล มีคุณลักษณะพื้นฐานดังต่อไปนี้

- หน่วยประมวลผลกลางขนาด 8 บิต
- ความสามารถในการอ้างตำแหน่งของหน่วยความจำโปรแกรม 64 กิโลไบต์
- ความสามารถในการอ้างตำแหน่งของหน่วยความจำข้อมูล 64 กิโลไบต์
- หน่วยความจำแบบ ROM ภายในจำนวน 128 ไบต์
- พอร์ตอินพุต / เอาต์พุตแบบขนาน จำนวน 32 เส้น
- วงจรนับ / จับเวลาขนาด 16 บิต
- วงจรสื่อสารแบบอนุกรมแบบฟูลดูเพล็กซ์
- วงจรการควบคุมการอินเตอร์รัปต์ จากแหล่งสัญญาณ 6 ประเภท
- วงจรออสซิลเลเตอร์ภายใน



รูปที่ 2.12 แสดงโครงสร้างหลักของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(T2) P1.0	1	40	VCC
(T2 EX) P1.1	2	39	P0.0 (AD0)
P1.2	3	38	P0.1 (AD1)
P1.3	4	37	P0.2 (AD2)
(SS) P1.4	5	36	P0.3 (AD3)
(MOSI) P1.5	6	35	P0.4 (AD4)
(MISO) P1.6	7	34	P0.5 (AD5)
(SCK) P1.7	8	33	P0.6 (AD6)
RST	9	32	P0.7 (AD7)
(RXD) P3.0	10	31	EA/VPP
(TXD) P3.1	11	30	ALE/PROG
(INT0) P3.2	12	29	PSEN
(INT1) P3.3	13	28	P2.7 (A15)
(T0) P3.4	14	27	P2.6 (A14)
(T1) P3.5	15	26	P2.5 (A13)
(WR) P3.6	16	25	P2.4 (A12)
(RD) P3.7	17	24	P2.3 (A11)
XTAL2	18	23	P2.2 (A10)
XTAL1	19	22	P2.1 (A9)
GND	20	21	P2.0 (A8)

รูปที่ 2.13 แสดงตำแหน่งขา MCS – 51

การทำงานแต่ละขาของไมโครคอนโทรลเลอร์ตระกูล MCS – 51 มีดังนี้

**ขาพอร์ท 0 (P0.0 – P0.7)** ใช้เป็นขาสำหรับพอร์ท 0 ขนาด 8 บิต พอร์ทนี้ สามารถใช้งานเป็นอินพุต – เอาท์พุต พอร์ททั่วไปได้ โดยหากใช้งานเป็นอินพุตพอร์ท ต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ทนี้ (มีสถานะ High impedance) นอกจากใช้งานเป็นอินพุต – เอาท์พุตแล้วพอร์ทเบอร์ 0 นี้ยังใช้ในการติดต่อกับหน่วยความจำสำหรับเก็บ โปรแกรมและข้อมูลภายนอกชิปอีกด้วย โดยส่งค่าแอดเดรสไบท์ต่ำ (A0 – A7) และมัลติเพลกซ์กับการส่งข้อมูล (D0 – D7) จากหน่วยความจำภายนอก ในระหว่างการเขียนหรือการอ่านข้อมูล

**ขาพอร์ท 1 (P1.0 – P1.7)** ใช้เป็นขาสำหรับพอร์ท 1 ขนาด 8 บิต สามารถใช้งานเป็นอินพุต – เอาท์พุต พอร์ททั่วไปได้ หากต้องการใช้งานเป็นอินพุต พอร์ทต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ท

**ขาพอร์ท 2 (P2.0 – P2.7)** ใช้เป็นขาสำหรับพอร์ท 2 ขนาด 8 บิต สามารถใช้งานเป็นอินพุต – เอาท์พุต พอร์ททั่วไปได้ โดยหากใช้งานเป็นอินพุต พอร์ทต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ท

ขาพอร์ท 3 (P3.0–P3.1) ใช้เป็นขาสำหรับพอร์ท 3 สามารถใช้งานเป็นอินพุทพอร์ททั่วไปได้ หากต้องการใช้งานเป็นอินพุทพอร์ทต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ทนี้ นอกจากนี้ยังสามารถใช้งานในหน้าที่พิเศษต่าง ๆ อีกหลายอย่างดังนี้

ขา Vcc สำหรับต่อแหล่งจ่ายแรงดันกระแสตรงขนาด 5 โวลท์

ขา Vss สำหรับต่อกราวด์

ขา XTAL1 ต่อคริสตัลจากภายนอก เป็นสัญญาณ อินพุทให้วงจรออสซิลเลเตอร์

ขา XTAL2 ต่อคริสตัลจากภายนอก เป็นสัญญาณ เอาต์พุท ออกจากวงจรออสซิลเลเตอร์

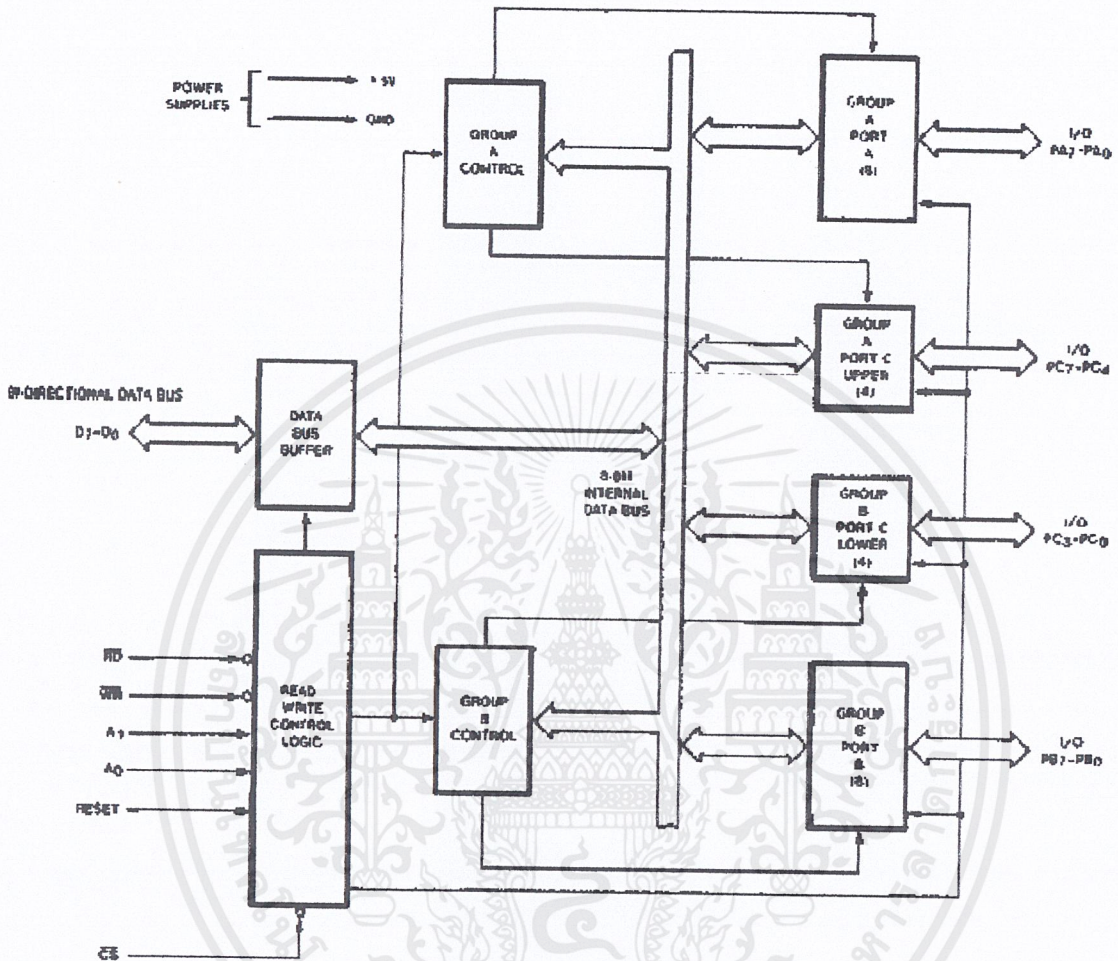
ขา EA / Vpp ใช้เลือกให้ MCS-51 ทำงานจากโปรแกรมภายในหรือภายนอกชิป โดยหากขา นี้เป็น 0 หมายถึงให้ใช้โปรแกรมภายนอก และหากขานี้เป็น 1 หมายถึงบังคับให้ MCS-51 ใช้โปรแกรมภายในชิป

ขา RST ใช้รีเซ็ตวงจรเพื่อเริ่มต้นการทำงานใหม่ เมื่อต้องการรีเซ็ตชิป MCS-51 ต้องป้อน 1 อย่างน้อย 2 คาบเวลา (Machine cycle) ของ CPU ระหว่างที่แหล่งกำเนิดความถี่ (Oscillator) ยังทำงานอยู่

ขา ALE / PROG เป็นขาสำหรับใช้ส่งสัญญาณออกไปภายนอก เพื่อควบคุมการ Latch ค่า แอดเดรสไบต์ต่ำจากพอร์ท 0 ในระหว่างการติดต่อหน่วยความจำสำหรับเก็บโปรแกรมหรือข้อมูลภายนอก นอกจากนี้ ขา ALE ยังใช้สำหรับควบคุมการเขียน โปรแกรมลงใน EPROM อีกด้วย

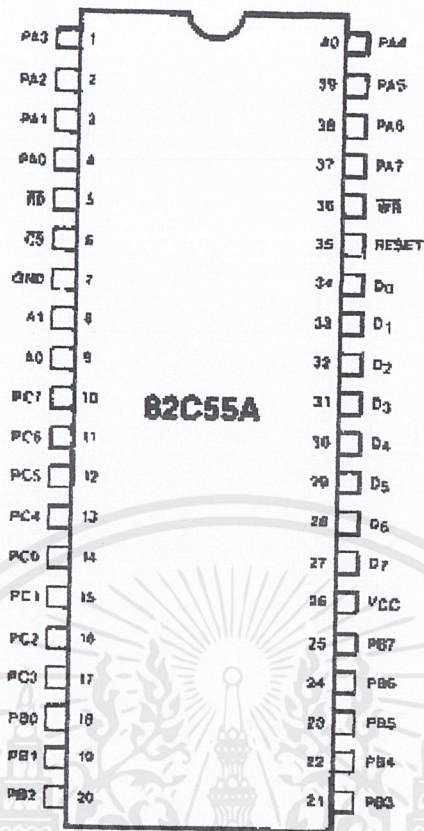
ขา PSEN ใช้ส่งสัญญาณ สไตรบเพื่ออ่านคำสั่งจากโปรแกรมที่เก็บไว้ในหน่วยความจำภายนอกชิป และเมื่อใช้โปรแกรมจากหน่วยความจำโปรแกรมภายในชิปจะไม่มีสัญญาณออกมาจาก ขา PSEN

## 2.9 ทฤษฎีเกี่ยวกับ 8255



รูปที่ 2.14 วงจรภายในของ 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 แสดงตำแหน่งขาของ 8255

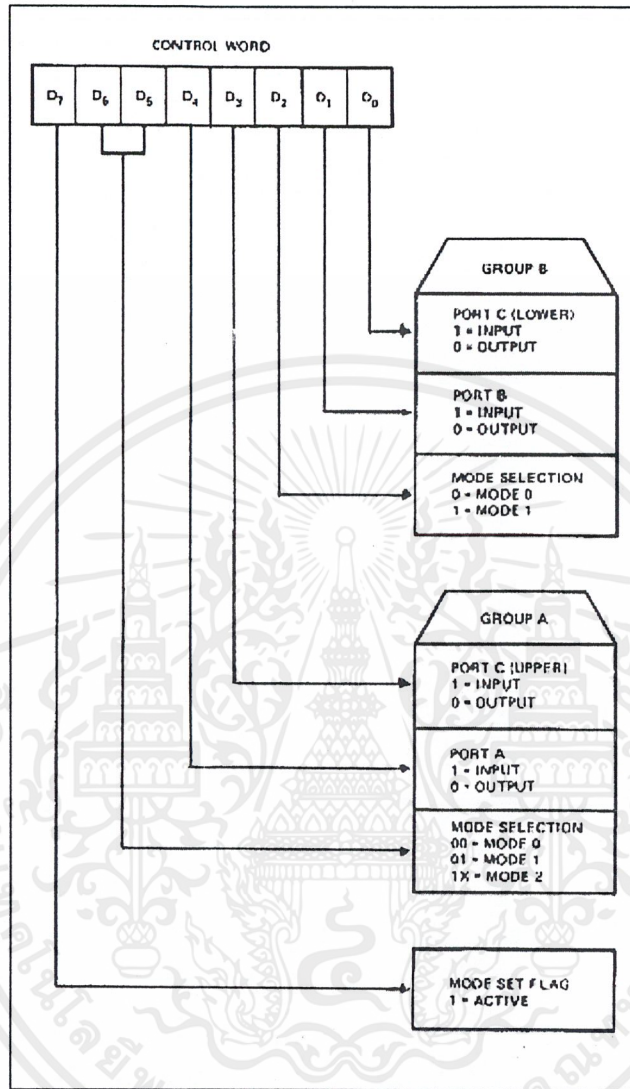
เนื่องจาก 8255 สามารถโปรแกรมตั้งงานให้เป็น Input หรือ Output ได้ตามต้องการจึงได้มีการนำมาใช้งานกันอย่างแพร่หลาย ถ้าดูจากรูปจะเห็นว่า มีขา PA, PB, PC อย่างละ 8 Bit และนี่คือขาใช้งานซึ่งเราเรียกว่า Port A, B, C ดังนั้นเมื่อเราใช้ Port สุดท้ายจะเป็น Port ที่สำคัญมากในการใช้งาน เพราะ 8255 จะใช้งานได้ก็ต่อเมื่อเขียนคำสั่งเข้าไปที่ Port นี้ก่อนเป็นอันดับแรก เรียกว่า Port Control ซึ่งเมื่อมีการเปลี่ยนแปลง Address A<sub>0</sub> และ A<sub>1</sub> จะหมายถึงการติดต่อ Port ดังนี้

A <sub>1</sub>	A <sub>0</sub>	Port ที่ถูกติดต่อ	แทน Binary A <sub>0</sub> และ A <sub>1</sub>
0	0	Port A	0
0	1	Port B	1
1	0	Port C	2
1	1	Port Control	3

Cs หมายถึง เมื่อมีสัญญาณที่ขานี้ Chip 8255 ตัวนั้นสามารถทำงานได้ และ A<sub>0</sub> A<sub>1</sub> เป็นตัวเลือก Port ภายในเนื่องจาก 8255 ในระบบอาจมีหลายตัว ซึ่งในส่วนนี้ถูกแยกด้วยส่วน Decoder อยู่แล้ว ดังนั้นถ้าต้องการติดต่อกับ Port ของ 8255 ตัวใดตัวหนึ่งจะต้องนำตำแหน่งที่ Decoder มา OR เข้ากับ Port ภายในของ 8255 ที่ต้องการนั่นก็คือ A<sub>0</sub> และ A<sub>1</sub>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รหัส Control ที่จะใช้เขียนเข้าไปใน Port Control แบ่งได้ตาม Bit ดังนี้



รูปที่ 2.16 แสดงรหัสการควบคุมพอร์ทของ 8255

#### ความหมายแต่ละ Bit

D7 แสดงถึงรหัสควบคุมให้เริ่มทำงาน ( 1 = ทำงาน ) คือ จะมีผลทำให้ 8255 รับรู้ในสิ่งต่อไป ใน Bit ต่างๆ ที่จะกำหนดให้เพราะฉะนั้นเวลาจะสั่งงานหรือหน้าที่ให้กับ 8255 Bit นี้ จะเป็น 1 เสมอ

D6 และ D5 เป็นการเลือกโหมดในการทำงานของ Port A ซึ่งมี 3 โหมดใน 8255

D4 กำหนดให้ Port A เป็น Input หรือ Output โดย

0 = Output Port

1 = Input Port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D3 กำหนดให้ Port C บนเป็น Input หรือ Out put โดย

0 = Output Port

1 = Input Port

D2 เป็นการเลือกโหมด ให้กับ Port B

0 = Output

1 = Input

D0 กำหนด Port C ล่าง (PCO-PC3)เป็น Input หรือ Output โดย

0 = Output

1 = Input

**MODE** มีด้วยกัน 3 โหมดดังนี้

Mode 0 เป็นโหมด Input/Output แบบพื้นฐาน ที่ใช้กันทั่วไปโดยโหมด นี้ส่งเป็น Input หรือ Out Put ได้ทั้ง 3 พอร์ต ( A,B,C )

Mode 1 เป็น โหมด Input/Output ที่มีการตรวจสอบสัญญาณซึ่งกันและกัน (Hand Shaking)ระหว่าง 8255 กับอุปกรณ์ภายนอกโดยใน โหมดนี้จะใช้ส่งงานเป็น Input หรือ Output ได้ที่ Port A และB เท่านั้นส่วน Port C จะใช้เป็นส่วนสำหรับตรวจสอบสัญญาณความพร้อมซึ่งกันและกัน

Mode 2 เป็น โหมด Bidirectional คือ Port จะเป็น ได้ทั้ง Input และ Output ภายใน Port เดียวกัน และในโหมดนี้ยังมีการตรวจสอบความพร้อมโดยใช้ Port C ในการตรวจสอบที่เหลือคือ Port B ที่ผู้จะใช้จะส่งให้อยู่ในโหมด 0 หรือ 1 ได้อย่างอิสระ

#### รายละเอียด MODE 0

ทุกครั้งที่มีการจ่ายไฟให้กับระบบหรือมีการ Reset 8255 ก่อนที่จะเริ่มใช้งาน 8255 ต้อง ส่งข้อมูลไปที่port Control 8255 นี้เสียก่อนเพียงครั้งแรกจากนั้นก็สามารุใช้คำสั่งในการ In หรือ Out ของ 8255 ตามที่ต้องการ

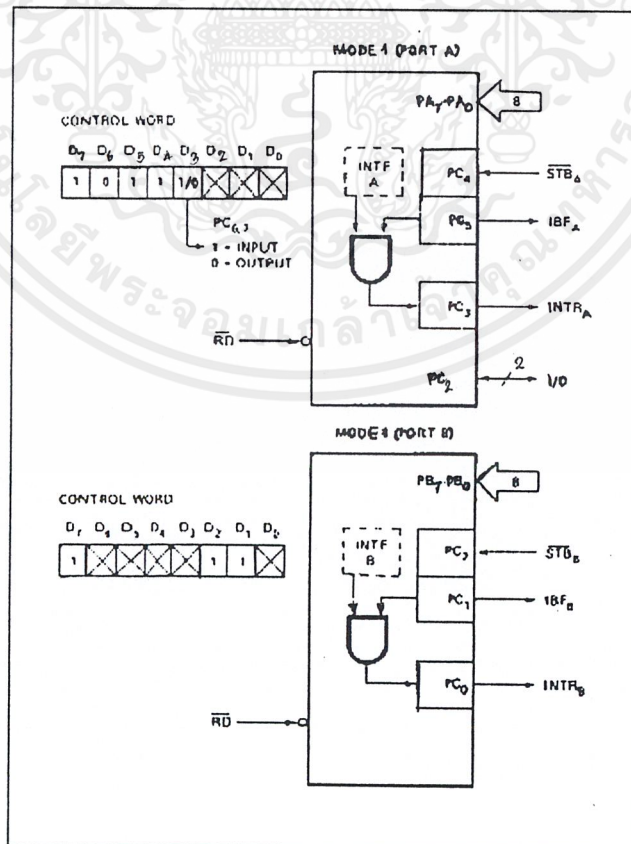
ตารางที่ 2.6 แสดงรหัสควบคุมกับการทำงานของพอร์ต

MODE 0 Port Definition

A		B		GROUP A		#	GROUP B	
D <sub>4</sub>	D <sub>3</sub>	D <sub>1</sub>	D <sub>0</sub>	PORT A	PORT C (UPPER)		PORT B	PORT C (LOWER)
0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUTPUT
0	0	0	1	OUTPUT	OUTPUT	1	OUTPUT	INPUT
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	OUTPUT	INPUT	4	OUTPUT	OUTPUT
0	1	0	1	OUTPUT	INPUT	5	OUTPUT	INPUT
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUTPUT
0	1	1	1	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUTPUT
1	0	0	1	INPUT	OUTPUT	9	OUTPUT	INPUT
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	INPUT	OUTPUT	11	INPUT	INPUT
1	1	0	0	INPUT	INPUT	12	OUTPUT	OUTPUT
1	1	0	1	INPUT	INPUT	13	OUTPUT	INPUT
1	1	1	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	INPUT	INPUT	15	INPUT	INPUT

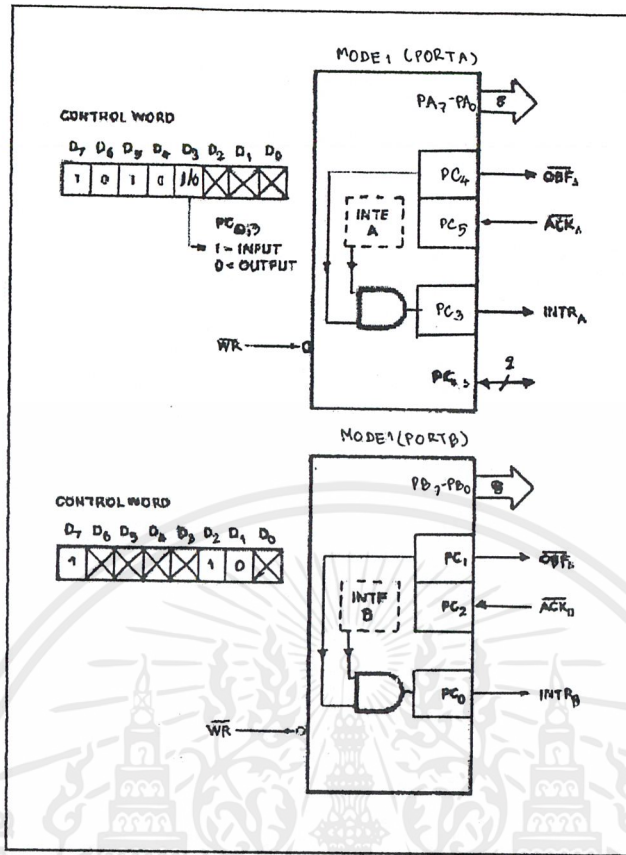
รายละเอียด MODE 1

รูปลักษณะที่ Set ในโหมด 1



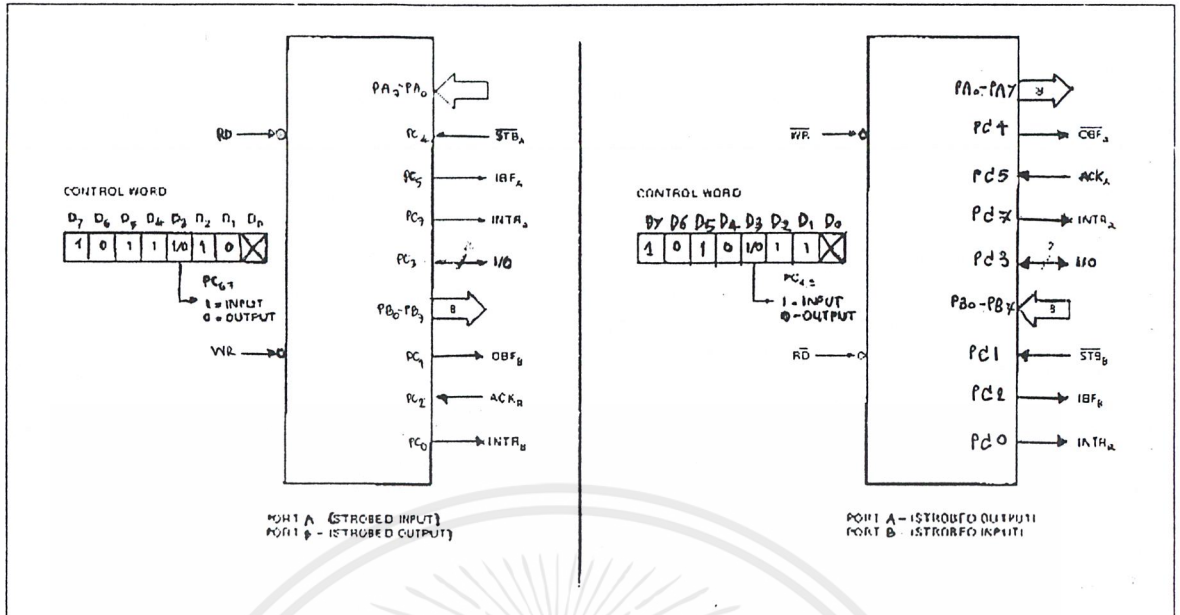
รูปที่ 2.17 แสดงลักษณะที่ set ในโหมด 1 เป็น Input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 แสดงลักษณะที่ set ในโหมด 1 เป็น Output

แนวความคิดของการใช้ Input- Output แบบ Hand Shake ก็เพื่อให้มีการซิงโครไนซ์ระหว่างอุปกรณ์ภายนอกที่ทำงานได้ช้า เช่นการส่งข้อมูลของ Computer ออกไปยังเครื่อง Printer เนื่องจาก Printer เป็น เมคคานิกส์ ย่อมทำงานช้ากว่า Computer ดังนั้นถ้า Computer ส่งข้อมูลไปอย่างเดียวโดยไม่มีการตรวจสอบความพร้อมของ Printer แล้วก็จะมึข้อมูลบางส่วนที่ Printer ไม่สามารถพิมพ์ได้(ข้อมูลหาย) เนื่องจากขณะที่ Printer กำลังพิมพ์ตัวอักษรตัวแรกอยู่ CPU ที่อยู่ในเครื่อง Printer ก็ไม่สามารถมาอ่าน Input จากที่ Computer ส่งมาได้ต้องรอจนกว่าจะพิมพ์ตัวอักษรนั้นเสร็จจึงจะมารับข้อมูลใหม่ได้แสดงรูปที่ 2.18 ใช้ 8255 เป็น Input และ Output แบบ Hand Shake



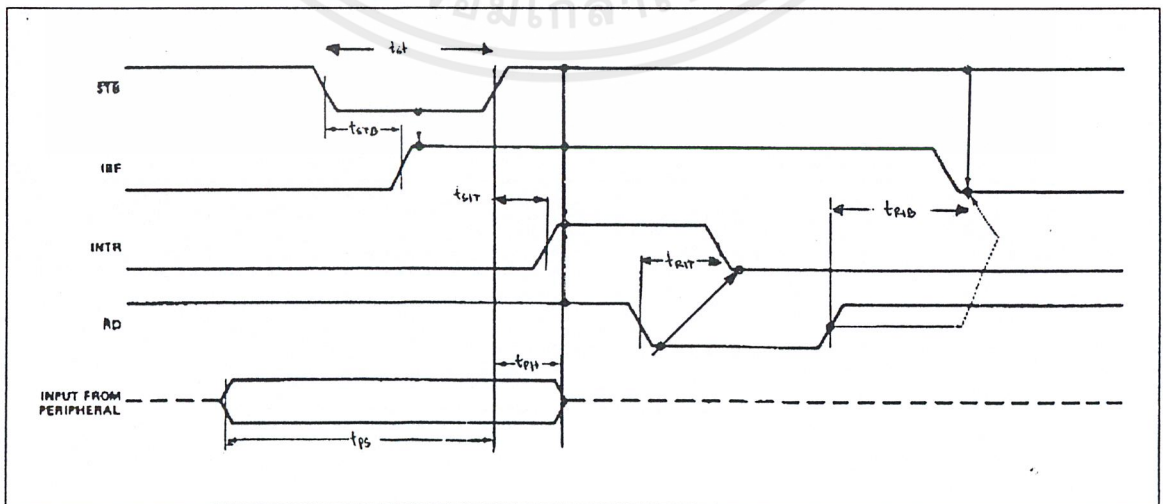
รูปที่ 2.19 แสดงลักษณะการทำ Hand Shake ในโหมด 1

ส่วน INPUT

STB (Strobe input) เป็นขา Input ทำงานที่ Logic 0 โดยเมื่อนานี้เป็น 0 ข้อมูลที่อยู่บนสายข้อมูล ที่เชื่อมต่อขาของ Port (A หรือ B) จะถูกอ่านเข้ามายัง Input Buffer ภายในตัว 8255

IBF (Input Buffer Full) เป็นขา Output ทำงานที่ Logic 1 ซึ่งจะตอบสนองต่อสัญญาณ Stb คือเมื่อ Stb เป็น 0 ซึ่งบอกว่ามีข้อมูลอยู่ใน Input Latch Buffer เรียบร้อยแล้วเวลานี้ก็จะ Active เป็น 1 และจะ Reset กลับเป็น 0 อีกครั้ง ต่อเมื่อมีการอ่าน Data จาก Port 8255

INTR (Interrupt Request) เป็นขา Output ทำงานที่ Logic 1 ซึ่งใช้สำหรับ Interrupt CPU โดยจะมีการทำงานก็ต่อเมื่อมีการ Set Flop Bit Inte ให้เป็น 1 และสัญญาณ Stb จาก 0 เปลี่ยนมาเป็น 1 จะทำให้ขา INTR ACTIVE เป็น 1 และเวลานี้จะถูก Reset กลับมาเป็น 0 ก็ต่อเมื่อมีการอ่านข้อมูลจากตัว 8255



รูปที่ 2.20 สัญญาณส่วน INPUT ของ 8255 โหมด 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

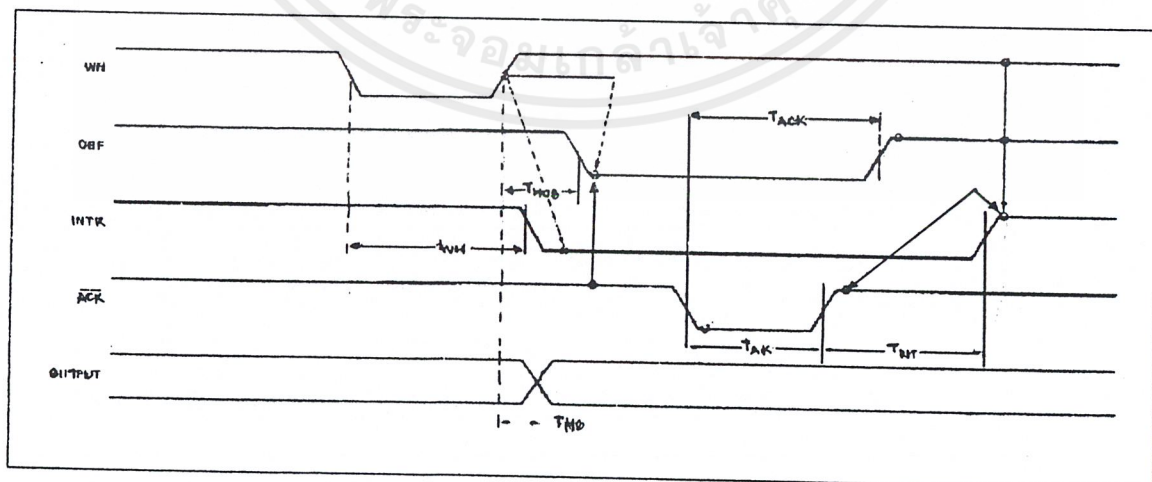
อธิบายได้ว่าเมื่ออุปกรณ์ภายนอกต้องการส่งข้อมูลมาให้ที่ 8255 อุปกรณ์ภายนอกจะทำการอ่าน IBF จาก 8255 ว่าตอนนี้ Inputlatch Buffer ของ 8255 นั้นว่างอยู่หรือไม่ โดยถ้า IBF = 0 หมายถึง Input Latch ของ 8255 นั้นว่างอยู่ อุปกรณ์ภายนอกที่ส่ง Data ออกไปยัง Line จากนั้นก็ส่ง Stb = 0 ไปบอก 8255 ว่าตอนนี้มี Data อยู่บน Line แล้วนะ 8255 ก็จะอ่าน Data นี้เข้าไปเก็บไว้ใน Inputlatch Buffer เมื่อเรียบร้อยแล้ว 8255 ก็จะส่ง IBF = 1 ออกไป เพื่อบอกอุปกรณ์ภายนอกว่าตอนนี้มี Data แล้ว ไม่ต้องส่งข้อมูลมาอีกนะ เมื่ออุปกรณ์ภายนอกอ่าน IBF = 1 แล้วก็ Reset Stb = 1 และถ้ามีการ Set ให้ 8255 สามารถส่งสัญญาณ Interrupt ไปบอก CPU ได้ ในขณะที่ Stb ถูก Reset จะทำให้ขา Intr ของ 8255 จะ Active เป็น 1 อยู่ และสัญญาณ Intr กับ IBF จะถูก Reset กลับเป็น 0 เพื่อบอกอุปกรณ์ภายนอกว่า ตอนนี้ Input latch Buffer ภายในว่างแล้วนะ ส่งข้อมูล Byte ถัดไปมาได้ ก็โดยที่ CPU ส่งสัญญาณมาอ่าน Data ภายในตัว 8255 นั้นและ Intr ก็จะ Reset กลับมาเป็น 0 ในขณะที่อ่าน Data จาก 8255 นั้น

### ส่วน OUTPUT

OBF (Output Buffer full) เป็นขา Output Active ที่ Logic 0 โดยขานี้จะ Active โดยอุปกรณ์ภายนอกนั้นต้องส่ง ACK = 1 มาให้ เพื่อบอกว่าฉันพร้อมที่จะอ่าน Data แล้วให้ CPU ส่ง Data มาได้ดังนี้ เมื่อ CPU เขียนข้อมูลเข้ามาที่ 8255 ตอนนี้ก็จะทำให้ OBF = 0 และ OBF จะ Reset กลับเป็น 1 ก็ต่อเมื่อ ACK ที่อ่านมาได้เท่ากับ 0

ACK (Acknowledgeinput) เป็นขา Input โดย Active ที่ Logic 0 ซึ่งถ้าขานี้เป็น 0 จะหมายถึง Output Buffer ภายใน 8255 ไม่ว่างห้าม CPU เขียน Data เข้ามาที่ 8255

INTR (Interrupt Request) เป็นขา Output เพื่อไป Interrupt Cpu โดย Active ที่ Logic 1 ซึ่งการจะ Active เกิดจาก OBF = 1 และ INTE = 1 และจะถูก Reset เป็น 0 เมื่อมีการเขียน Data เข้าไปที่ 8255



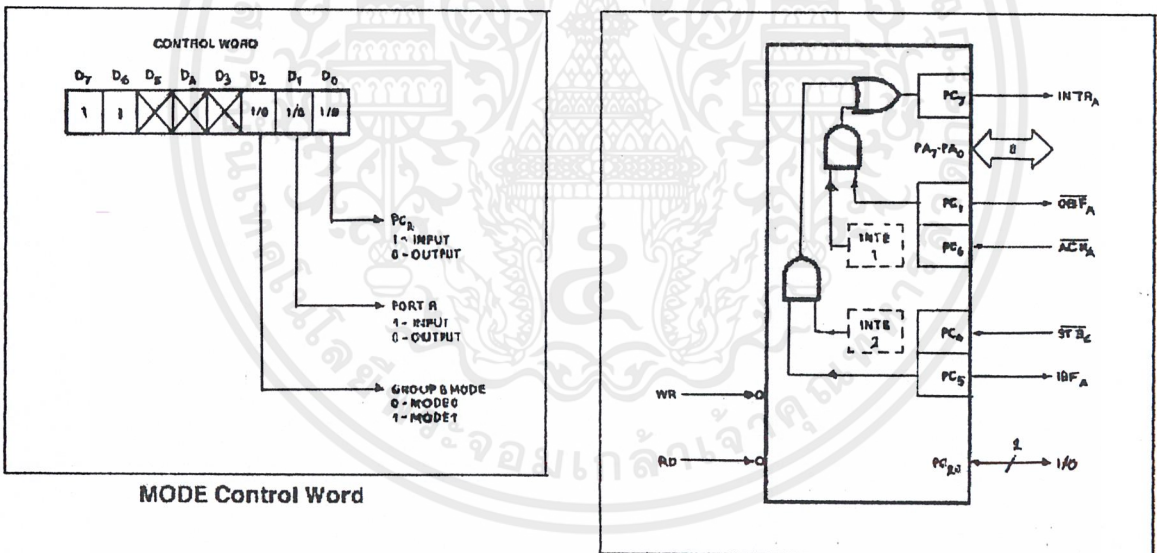
รูปที่ 2.21 สัญญาณ OUTPUT ของ 8255 โหมด 1

อธิบายได้ดังนี้

ก่อนที่ CPU จะทำการเขียนข้อมูลเข้าไปที่ 8255 ควรจะตรวจสอบดูก่อนว่าอุปกรณ์ภายนอกมาอ่าน Data ไปเรียบร้อยแล้วหรือยัง โดยดูจากสัญญาณ ACK ว่าเท่ากับ 1 หรือไม่ ถ้าเป็น 1 ก็ให้ CPU เขียนข้อมูลเข้ามาที่ 8255 ได้ เมื่อมีการเขียนข้อมูลเข้ามาแล้ว OBF จะ Active Low จนกระทั่ง ACK Active เป็น 0 ถึงจะทำให้ OBF Reset กลับมาเป็น 1 และถ้ามีการ Set ให้มีการ Interrupt พอ ACK กลับมาเป็น 1 แล้วก็จะทำให้เกิด Interrupt โดย Inter Active เป็น 1

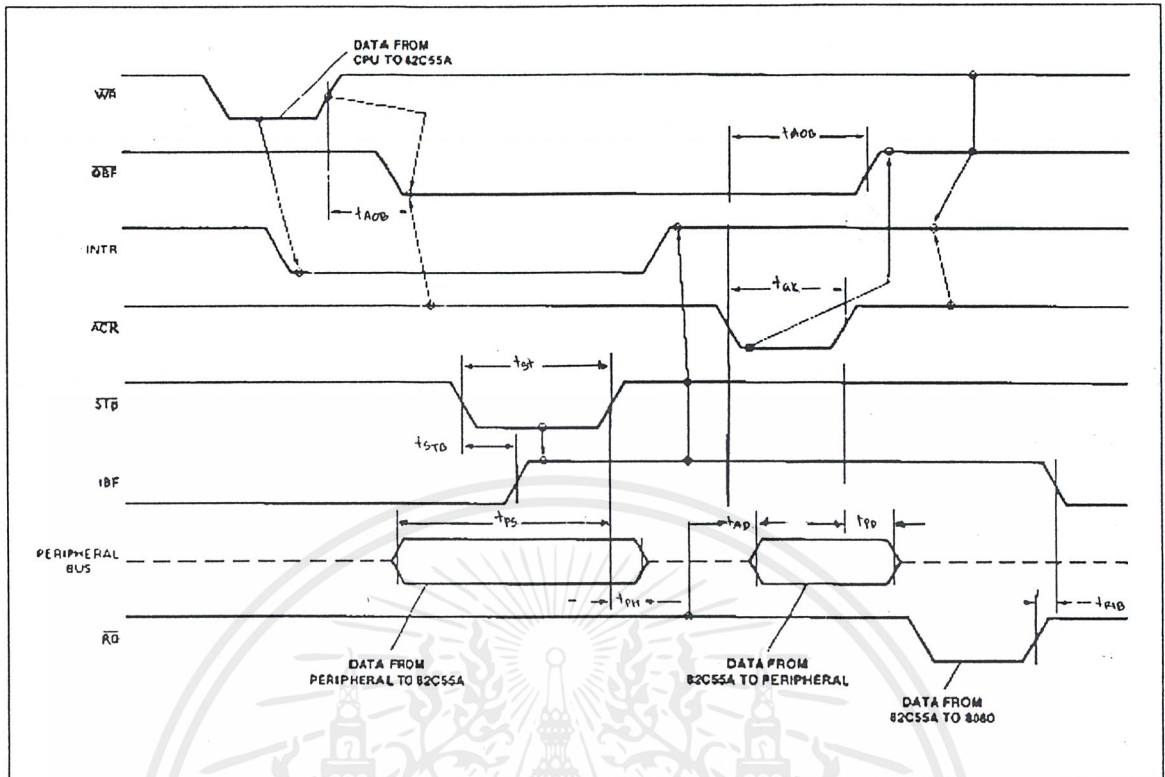
**รายละเอียด MODE 2**

ในโหมดนี้ก็คือ การรวมกันของโหมด 1 ในส่วน Input และ Output มาไว้ในส่วนเดียวกัน โดยใช้ Port A เป็นตัวเชื่อมต่อ นั่นก็คือ Port B จะเป็น Port แบบสองทิศทาง นี่ยังเป็นระบบที่มีการ Hand Shake ดังเช่น โหมด 1 โดยใช้ Port C เป็นตัวตรวจสอบ โดยสัญญาณต่างๆ นั่นก็คือ สัญญาณในส่วน Input และ Output ของโหมด 1 มารวมกันนั่นเอง โดยแต่ละ Bit ของ Port C เป็นดังนี้



**รูปที่ 2.22 แสดงลักษณะการทำงานโหมด 2**

จากรูปจะเห็นว่า Port C จะมี Bit เหลือที่ให้อีก 3 ขา คือ PC0 – PC2 และ Port B เหลือทั้ง Port โดย Port B ยังกำหนดให้เป็น โหมด 0 หรือ โหมด 1 ได้อีก ในกรณี Port B Set เป็นโหมด 0 PC0-PC2 สามารถกำหนดให้เป็น Input หรือ Output ได้แต่ถ้าเป็นโหมด 1 PC0-PC2 จะใช้ในการ Hand Shake โดยสามารถสรุปได้ดังรูปที่ 2.22



รูปที่ 2.23 แสดงสัญญาณการทำ Hand Shake Mode 2

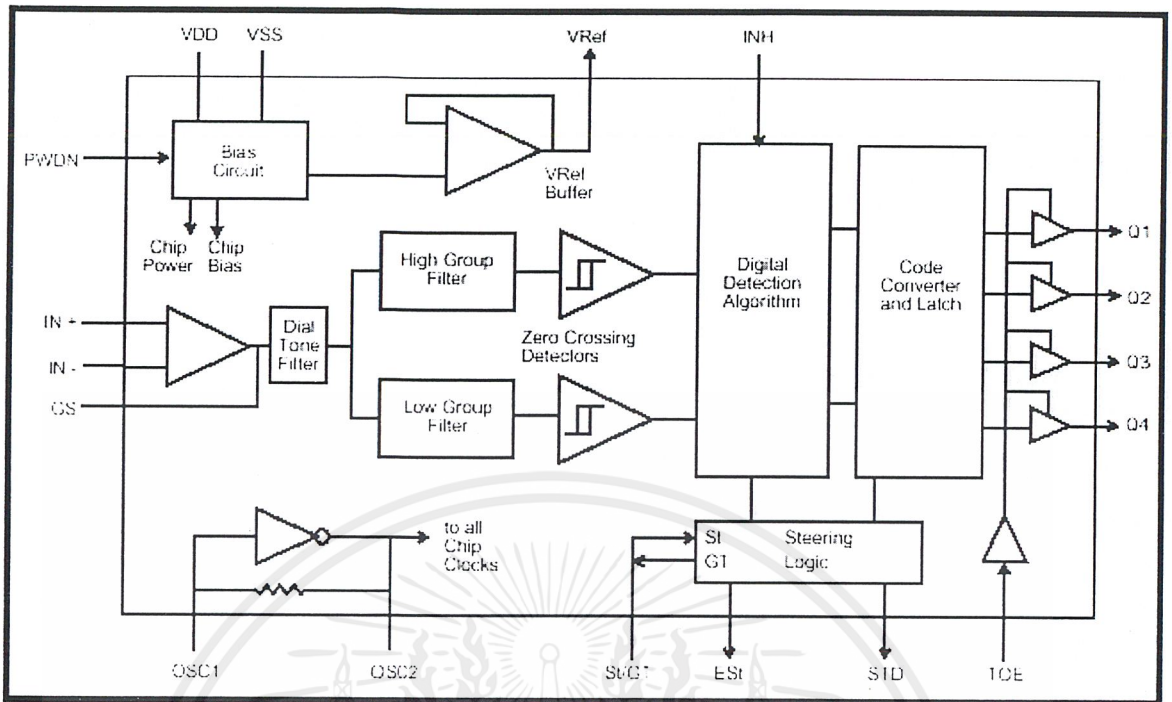
ประโยชน์ของโหมด 2 ในกรณีที่เรากำลังทำงานที่ต้องใช้ความเร็วอย่างมากโดย CPU เพียง 1 ตัวไม่สามารถที่ควบคุมได้ยกตัวอย่างเช่น ให้มีการรับข้อมูลทาง Serial Port จากอุปกรณ์ภายนอกเข้ามาแล้วทำการประมวลผลเพื่อกำหนดค่า จากนั้นนำค่านี้ออกไปแสดงผลที่แผงไฟโฆษณาแล้วให้มีการนำค่าที่แผงไฟนั้นไปพิมพ์ที่เครื่องพิมพ์ ซึ่งที่แผงไฟต้องมีการแสดงผลตลอดเวลาห้ามหยุด ดังนั้น CPU ที่ 1 ก็ทำการรับข้อมูลทาง Serial Port และ Scan แผงไฟส่วน CPU ที่ 2 ใช้ในการ Print ข้อมูลที่ Printer โดยการใช้โหมด 2 ในการเชื่อมต่อ CPU ที่ 1 และ CPU ที่ 2 โดย CPU ที่ 1 เพียงแต่ส่งค่าต่างๆ ออกไปยัง CPU ที่ 2 เพื่อให้ CPU ที่ 2 ทำงาน Print ไปก็จะทำให้ CPU ที่ 1 ไม่เสียเวลาในการพิมพ์

## 2.10 ทฤษฎีเบื้องต้นเกี่ยวกับ MT8870

### 2.10.1 โครงสร้างของ MT8870

โครงสร้างภายในของ MT8870 ประกอบไปด้วยวงจรรองความถี่และวงจรถอดรหัสฟังก์ชันทางดิจิทัลเป็น ไอซีที่สร้างโดยใช้เทคโนโลยี ISO-CMOS ในส่วนของวงจรรองความถี่ใช้เทคนิคของสวิทช์คาปาซิเตอร์ฟิลเตอร์ สำหรับรองความถี่สูงและต่ำ ส่วนวงจรถอดรหัส ใช้เทคนิคการนับทางดิจิทัล เพื่อตรวจจับและถอดรหัสทั้ง 16 ความถี่เป็นเลขฐานสองขนาด 4 บิต และเช็คช่วงเวลาที่ยืนยันเข้ามา ส่วนภาคอินพุท (Input) เป็นออปแอมป์ ซึ่งสามารถปรับอัตราขยาย (Gain) ได้โดยต่ออุปกรณ์ภายนอกเอาท์พุท (Output) เป็นวงจรแลตช์ (Latch) 3 สถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

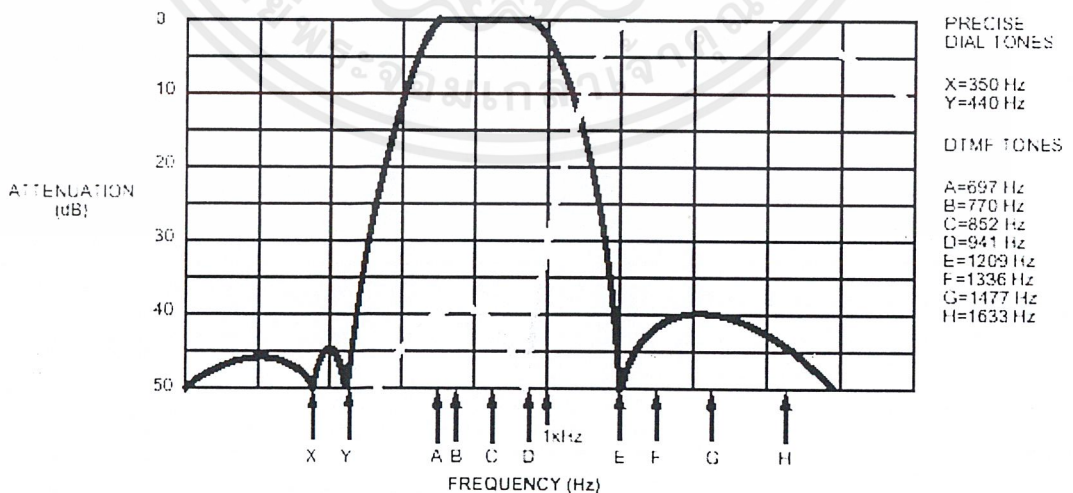


รูปที่ 2.24 แสดงโครงสร้างภายในของ MT8870

2.10.2 ฟังก์ชันการทำงานภายในของ MT887

ภายใน MT8870 ประกอบด้วยส่วนสำคัญ 5 ส่วนคือ

1) ภาคกรองความถี่ (Filter Section) ในส่วนนี้จะทำการแยกสัญญาณ DTMF ที่เข้ามาเป็น 2 กลุ่มความถี่ คือ ช่วงความถี่สูงและความถี่ต่ำ โดยใช้วงจร กรองความถี่อันดับ 6 ชนิดสวิตซ์ คาปาซิเตอร์ ซึ่งความถี่ที่แยกได้มี 2 ช่วงความถี่สูงและช่วงความถี่ต่ำ



รูปที่ 2.25 แสดงความถี่ที่ได้จากภาคกรองความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

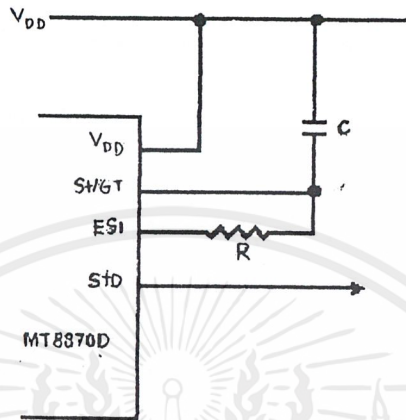
2) ภาคถอดรหัส (Decoder Section) ความถี่ DTMF ที่ถูกรองเรียบร็อยแล้วจะผ่านเข้าวงจรถอดรหัสความถี่ออกเป็นตัวเลข โดยเทคนิคการนับแบบ ดิจิตอลและ มีการตรวจสอบความถี่ที่เข้ามาเป็นความถี่มาตรฐาน DTMF หรือไม่ เพื่อป้องกันความถี่อื่นเข้ามาผสม เมื่อตรวจสอบว่าความถี่นั้นถูกต้องสัญญาณที่ขา Est (Early Steering) ก็จะแอ็คทีฟ สำหรับค่าที่ถอดรหัสได้จากความถี่ต่าง ๆ นั้นแสดงดังตารางที่ 2.7

ตารางที่ 2.7 แสดงค่าที่ถอดรหัสได้จากความถี่ต่าง ๆ

F <sub>LOW</sub>	F <sub>HIGHT</sub>	NO	TOE	Q4	Q3	Q2	Q1
697	1209	1	H	0	0	0	1
697	1336	2	H	0	0	1	0
697	1477	3	H	0	0	1	1
770	1209	4	H	0	1	0	0
770	1336	5	H	0	1	0	1
770	1477	6	H	0	1	1	0
852	1209	7	H	0	1	1	1
852	1336	8	H	1	0	0	0
852	1447	9	H	1	0	0	1
941	1336	0	H	1	0	1	0
941	1209	*	H	1	0	1	1
941	1477	#	H	1	1	0	0

3) ภาคตรวจสอบสัญญาณ (Steering Circuit) ก่อนที่จะมีการถอดรหัสความถี่ออกไปที่เอาต์พุท จะมีการตรวจสอบช่วงความถี่ที่เข้ามาว่ามีระยะเวลาตามกำหนดหรือไม่ โดยสังเกตจากระยะเวลาการกดปุ่มโทรศัพท์ซึ่งต้องกดปุ่มให้มีความถี่ออกมาเป็นช่วงเวลาพอสมควร มิฉะนั้นวงจรส่วนนี้จะไม่รับ โดยถือว่าสัญญาณนั้นไม่ถูกต้อง ส่วนช่วงเวลายาวเท่าใด สามารถตั้งได้โดยใช้ RC ต่อภายนอกสัญญาณที่ขา Est จะเป็น "High" นานใกล้เคียงกับระยะเวลาที่มีความถี่ DTMF เข้ามาจากรูปที่ 2.23 เมื่อขา Est เป็น "High" ที่ให้ Vc สูงขึ้นตัวเก็บประจุ C จะคายประจุทำให้แรงดัน Vc สูงขึ้นจนถึงค่า เทนเซอร์โวลต์วงจรถอดรหัสจึงจะถอดรหัสออกเป็นตัวเลขขนาด 4 บิต สำหรับคำว่า การ์ดไทม์ (Guard Time) นั้นหมายถึง ช่วงคาบเวลาของความถี่ที่เข้ามาซึ่งจะต้องนานเท่ากับ หรือมากกว่าช่วงเวลา ที่เรที่ตั้งไว้จึงจะได้รับการยอมรับ ว่าสัญญาณความถี่นั้นถูกต้อง หรือพูดได้ว่าเวลาที่เรที่ตั้งไว้โดย RC ก็คือการ์ดไทม์นั่นเอง เมื่อสัญญาณความถี่เข้ามานาน หรือเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

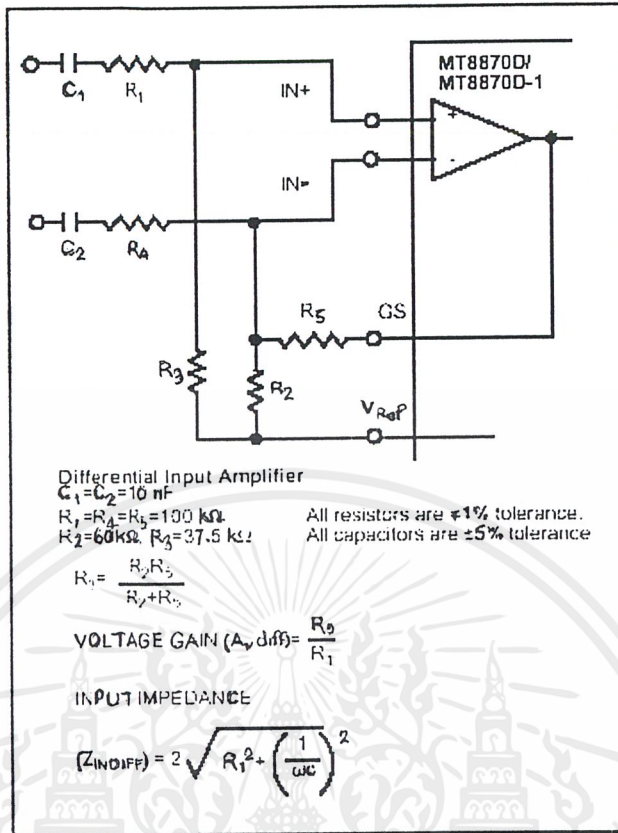
หรือพูดได้ว่าเวลาที่เราคั้งไว้โดย RC ก็คือคาร์คไทม์นั่นเอง เมื่อสัญญาณความถี่เข้ามานาน หรือมากกว่าเวลาที่คั้งไว้จึงจะสามารถแปลงเป็นตัวเลขได้ ถ้าสัญญาณความถี่เข้ามาสั้นกว่าก็จะไม่มีการถอดรหัสเป็นตัวเลขออกไป



รูปที่ 2.26 แสดงวงจรตรวจสอบสัญญาณอย่างง่าย

#### 4) ภาคขยายสัญญาณต่าง ๆ (Differential Input)

วงจรส่วนอินพุท ของ MT8870 เป็นภาคขยาย ออปแอมป์ ที่สามารถปรับอัตราขยายโดยต่อวงจรภายนอกเพิ่มเข้าไปดังรูปที่ 2.24 แสดงการต่อวงจรภายนอกเข้ากับอินพุท ซึ่งสามารถคำนวณอัตราขยาย ความแตกต่างของ อินพุทและอิมพีแดนซ์ได้



รูปที่ 2.27 แสดงการต่อวงจรภาคอินพุต

5) ภาคกำเนิดความถี่ (Oscillator Section)

ในภาคนี้ภายใน ไอซี จะมีวงจรเวลาอยู่ภายใน เพียงแต่ต่อแร่คริสตอล ขนาด 3.58 MHz ก็ สามารถใช้งานได้ทันที

คุณสมบัติของ MT8870

- ก) เป็นตัวรับและถอดรหัสความถี่ DTMF
- ข) กินไฟน้อยใช้ไฟเลี้ยงระดับเดียวกับ TTL
- ค) สามารถตั้งอัตราขยายภายในตัวไอซีได้
- ง) สามารถปรับการ์ดใหม่ได้
- จ) เป็นไอซีคุณภาพสูง

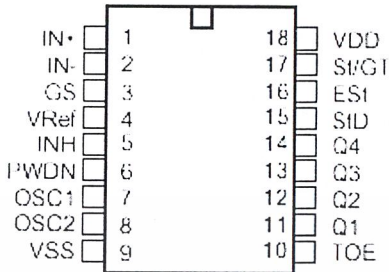
2.10.3 การนำ MT8870 ไปใช้งาน

- 1) นำไปใช้งานด้านรีโมทคอนโทรล
- 2) เครื่องป้องกันโทรศัพท์ทางไกล
- 3) ใช้งานเกี่ยวกับเครดิตการ์ด
- 4) ใช้งานร่วมกับคอมพิวเตอร์

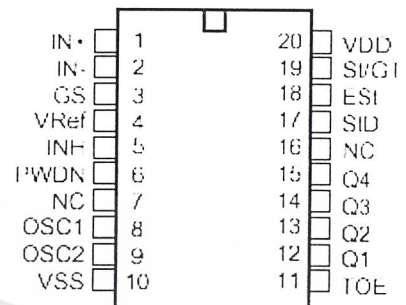
5) ใช้งานในชุมสายขนาดเล็กหรือ PABX

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับโรงเรียนในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 6) ใช้งานด้านโทรศัพท์ทั่วไป
- 7) เครื่องกันขโมย
- 8) การควบคุมอุปกรณ์ทางโทรศัพท์

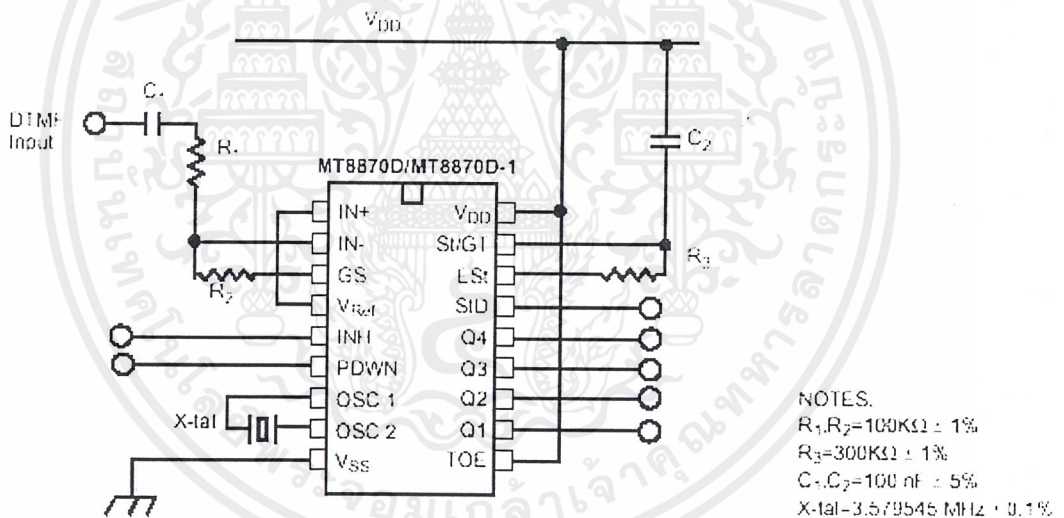


18 PIN PLASTIC DIP/SOIC



20 PIN SSOP

รูปที่ 2.28 แสดงรายละเอียดขาของ MT8870



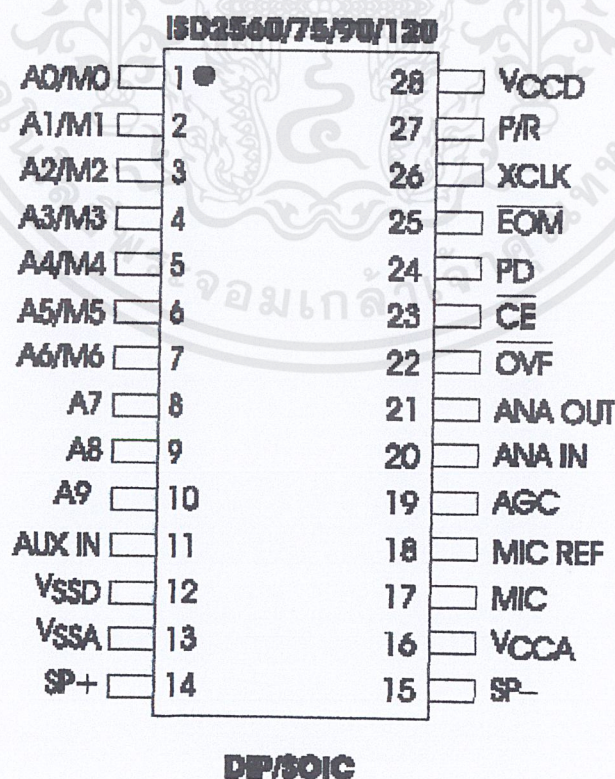
รูปที่ 2.29 แสดงวงจรใช้งานเบื้องต้นของ MT8870

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.11 ข้อมูลเกี่ยวกับ ไอซีเบอร์ ISD 2590

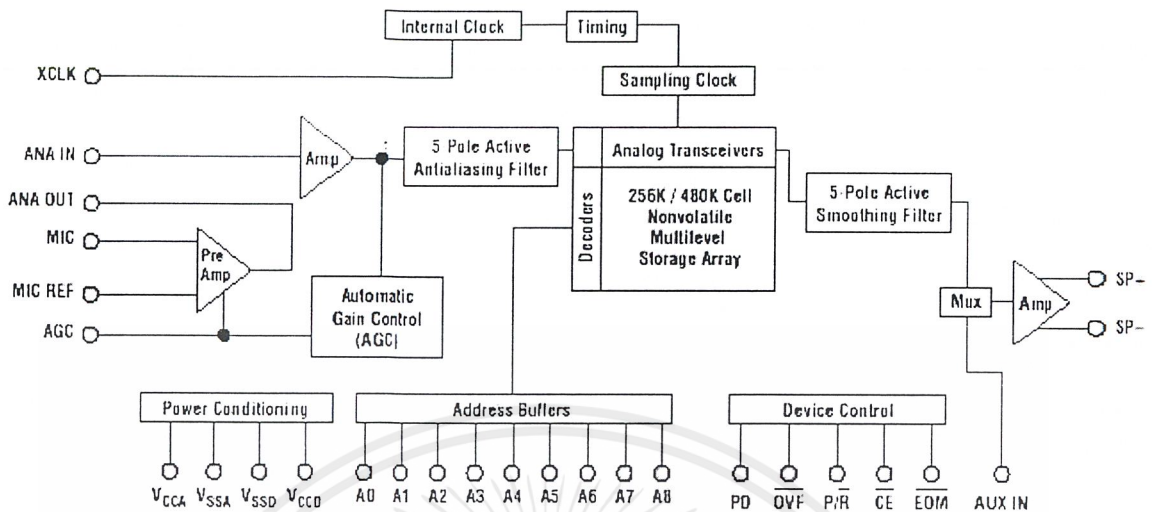
ไอซีเบอร์ ISD 2590 เป็นไอซีที่สามารถบันทึกและสามารถเล่นกลับได้ในตัวเองโดยการต่อวงจรใช้งานไม่ยุ่งยาก อาศัยอุปกรณ์ภายนอกไม่กี่ตัว ไม่ต้องต่อกับภาคขยายก็ใช้งานได้เลย ซึ่งสามารถสรุปคุณสมบัติได้ดังนี้

- สามารถบันทึกและเล่นกลับได้อย่างง่ายดาย
- มีประสิทธิภาพในการบันทึกและเล่นกลับที่ให้เสียงเหมือนต้นกำเนิดเสียง
- ควบคุมการบันทึกและเล่นกลับได้โดยสวิตช์ หรือควบคุมด้วยไมโครคอนโทรลเลอร์
- ระยะเวลาในการบันทึกและเล่นกลับตั้งแต่ 45,60,75,90 วินาที ตามแต่เบอร์
- ต่อ cascade ได้โดยตรงเพื่อเพิ่มระยะเวลาให้ยาวมากขึ้นได้
- มีระบบ stand by ปิดการทำงานอัตโนมัติเมื่อไม่มีการบันทึกหรือเล่นกลับนานเกินไป
- ข้อมูลที่ทำการบันทึกไว้จะไม่สูญหาย เมื่อไม่จ่ายไฟให้กับไอซี
- เก็บข้อมูลได้นาน 100 ปี
- สามารถตำแหน่ง address ที่จะทำการบันทึกและเล่นกลับได้
- สามารถกำหนดโหมดการทำงานได้



รูปที่ 2.30 แสดงตำแหน่งขาของไอซีบันทึกเสียงเบอร์ ISD2590

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.31 แสดง Block Diagram ในแต่ละภาคภายในตัวไอซีเบอร์ ISD 2590

การบันทึกเสียงจะมีวิธีการบันทึกโดยอาศัยเทคโนโลยี DAST (Direct Analog Strobe Technology) ซึ่งจะทำให้ได้เสียงเหมือนกับต้นกำเนิดเสียงมาก และไอซีเบอร์ ISD 2590 สามารถบันทึกเสียงได้นาน 90 วินาที ใช้ Sampling Rate 4000 Hz

การรับอินพุตสามารถทำได้ 2 วิธี คือ

- ทางขา MIC เป็นการรับเสียงมาจากไมโครโฟน ดังนั้นจะต้องเชื่อมต่อ ANA OUT กับ ANA IN ด้วยเพื่อให้อินพุตเข้าไปเก็บใน Storage

- ทางขา ANA IN เป็นการรับสัญญาณ Analog มาเก็บใน Storage โดยตรง

สำหรับทางด้านเอาต์พุต มี MUX สำหรับเพื่อทำการเลือกที่จะขยายสัญญาณที่บันทึกเก็บไว้ หรือขยายสัญญาณจากภายนอกที่ขา AUX IN

การใช้งานของขาต่างๆ มีดังนี้

ขา 1-10 ชื่อ Address/Mode Input ( $A_0$ - $A_9$  /  $M_0$ - $M_6$ ) ใช้เป็น Address เริ่มต้นการบันทึก และเล่นกลับ

ขา 11 ชื่อ Auxilliary Input (AUX IN) ใช้รับอินพุตจากภายนอก

ขา 12, 13 ชื่อ Ground Input ( $V_{SSA}$ ,  $V_{SSD}$ ) ต่อกับ Ground

ขา 14, 15 ชื่อ Speaker Output เป็นขาเอาต์พุตต่อกับลำโพง

ขา 16, 28 ชื่อ Voltage Input ( $V_{CCA}$ ,  $V_{CCD}$ ) ขารับแรงดันไฟเลี้ยง

ขา 17 ชื่อ Microphone Input (MIC) รับสัญญาณอินพุต

ขา 18 ชื่อ Microphone Reference (MIC REF) ต่อกับนี้กับ Ground Analog

ขา 19 ชื่อ Automatic Gain Control Input เป็นอินพุตเพื่อควบคุมปรับอัตราการขยายของปริแอมป์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา 20 ชื่อ Analog Input (ANA IN) รับสัญญาณที่ผ่านปริแอมป์

ขา 21 ชื่อ Analog Output (ANA OUT) เป็นขาเอาต์พุตของวงจรปริแอมป์

ขา 22 ชื่อ Overflow Output ( $\overline{OVF}$ ) แสดงการสิ้นสุดการเล่นกลับหรือหน่วยความจำภายในตัวไอซีได้ถูกอ่านหมดแล้ว

ขา 23 ชื่อ Chip Enable Input ( $\overline{CE}$ ) ต้องได้รับ Pluse "0" เพื่อให้เกิดการเปลี่ยนแปลงระหว่างการเล่นกลับและการบันทึก

ขา 24 ชื่อ Power Down (PD) รักษาระดับการสิ้นเปลืองพลังงาน

ขา 25 ชื่อ End Of Message/Run Output (EOM) เป็นส่วนของอุปกรณ์ non volatic

ขา 26 ชื่อ External Clock Input (XCLK) ใช้กำหนดความถี่สัญญาณนาฬิกาในการสุ่มสัญญาณ

ขา 27 ชื่อ Playback/Record( $P/\overline{R}$ ) ใช้ในการกำหนดเล่นหรือบันทึก



## บทที่ 3

### การออกแบบโครงงานและการสร้าง

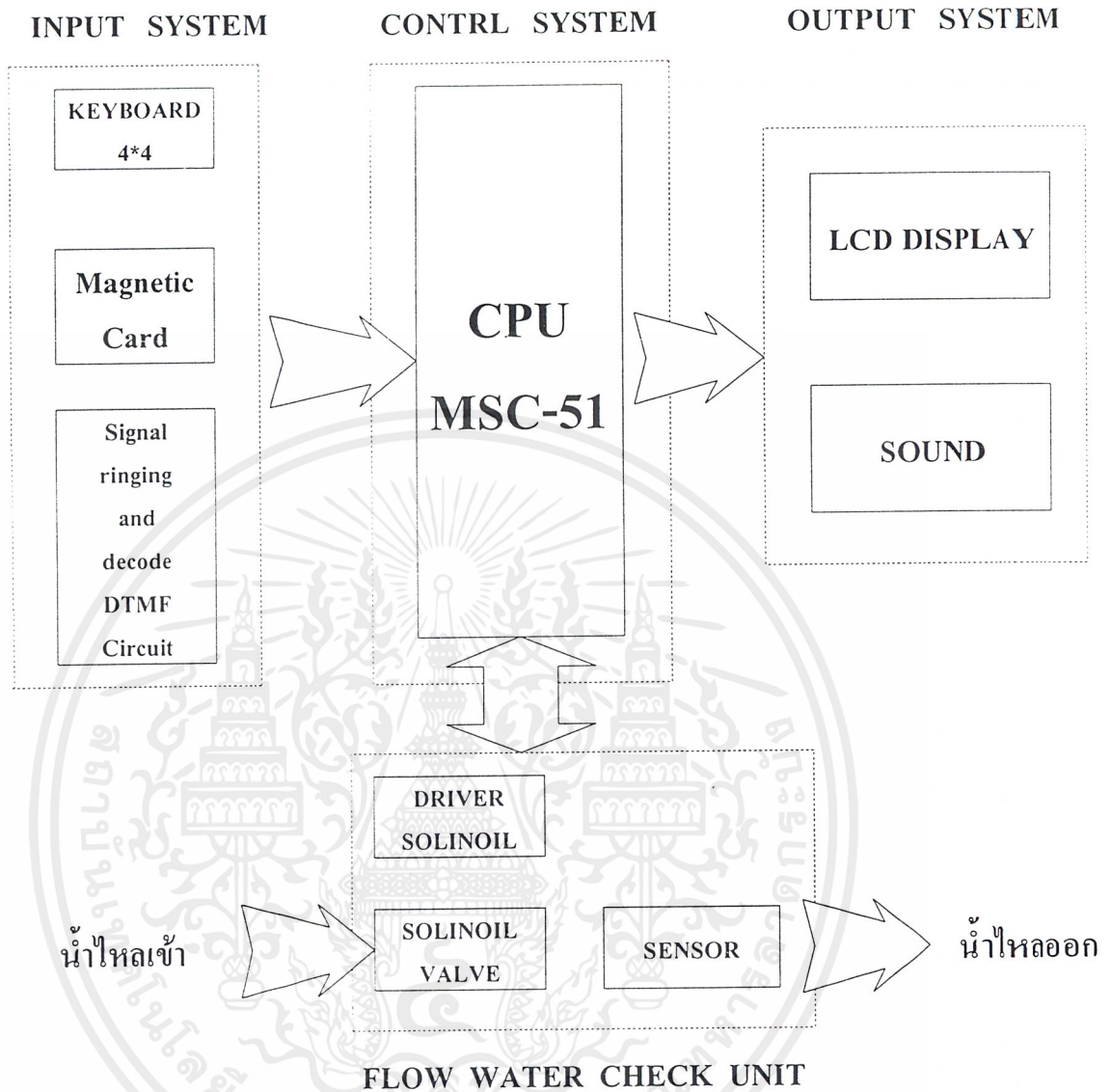
#### 3.1 คำนำ

ในการออกแบบโครงงานนี้จะมีส่วนประกอบหลักอยู่ 2 อย่าง คือ Hardware และ Soft ware ในการออกแบบ เราจะสร้าง Hardware ส่วนประกอบของวงจรต่างๆขึ้นมาก่อนจากนั้น จึงทำการเขียนโปรแกรมควบคุมให้ส่วนประกอบต่างๆ ของ Hardware ทำงานตามจุดประสงค์ที่เราต้องการ

#### 3.2 Hard Ware

ในการออกแบบ Hard Ware โครงงานนี้ จะมีส่วนประกอบหลักๆ คือ

- ภาคประมวลผล (MCS-51)
- Key Board แบบ 4x4
- เครื่องอ่านบัตรแม่เหล็ก ( ของ Panasonic รุ่น ZU-M2121S451 )
- LCD Display
- ภาคควบคุมการปิด/เปิดวาล์วน้ำ
- ภาคตรวจจับและถอดรหัสความถี่ DTMF
- ภาคเสียงตอบกลับ



รูปที่ 3.1 บล็อกไดอะแกรมแสดงการทำงานของโครงการ

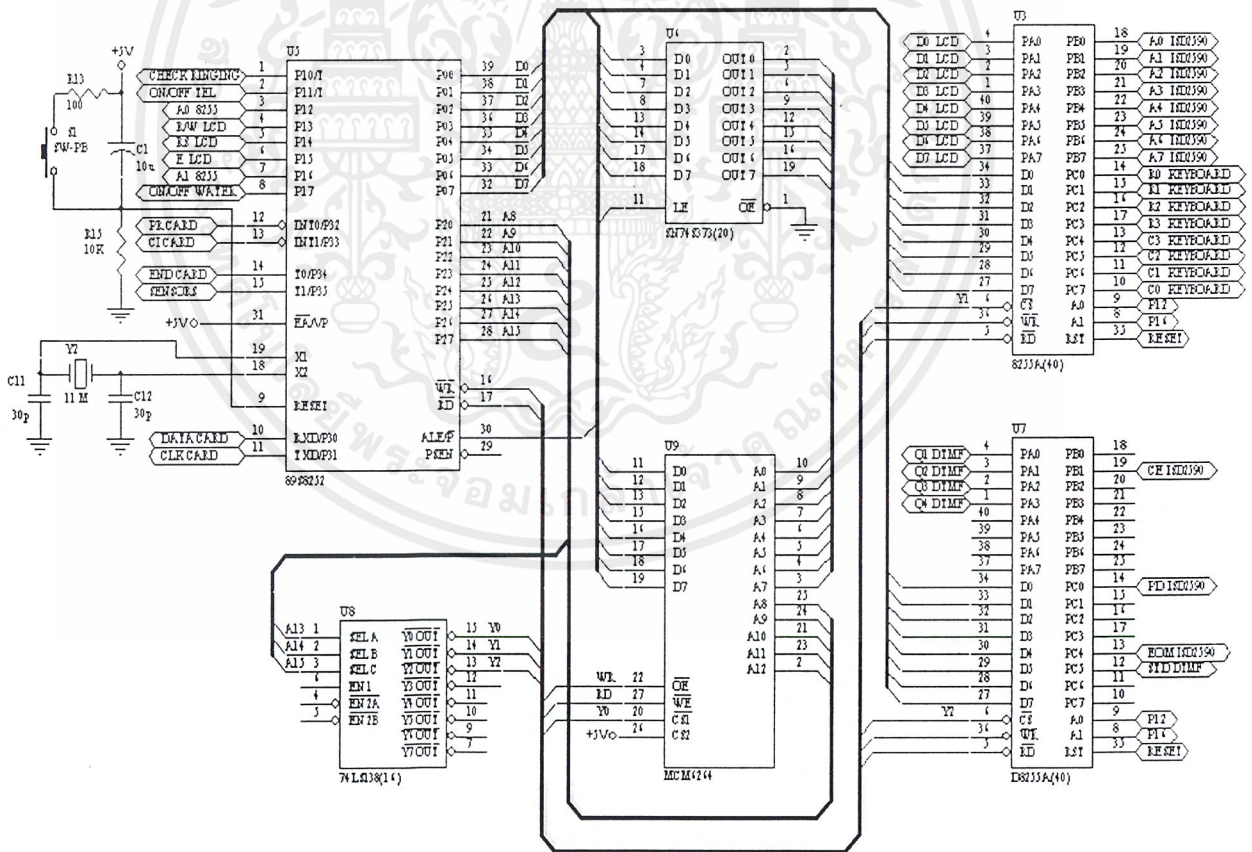
### 3.2.1 การออกแบบภาค Control

เนื่องจาก CPU เบอร์ 89S8252 ซึ่งเป็นตระกูล MCS - 51 มี Flash Memory ในตัว ซึ่งสามารถใช้พื้นที่นี้ ในการเขียนโปรแกรมได้ 8 KB และมีพอร์ตในตัว 4 พอร์ต คือ พอร์ต 0-พอร์ต 3 ตัวมันสามารถที่จะเขียนหรือลบโปรแกรมผ่าน RS-232 ได้ทำให้สะดวก และง่ายในการออกแบบ เมื่อใช้ Flash Memory ในตัวมันเอง เป็นตัวเก็บโปรแกรม เราจึงนำพอร์ต ทั้ง 4 ไปควบคุมการทำงานต่างๆ ของอุปกรณ์แต่เนื่องจากได้มีการออกแบบให้มีการใช้แรม ภายนอกด้วย จึงทำให้เหลือพอร์ตต่อใช้งานเพียง 2 พอร์ต คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต 1 และ พอร์ต 3 ซึ่งไม่เพียงพอดต่อการนำไปต่อเป็น I/O ของระบบทำให้ต้องขยายพอร์ตโดยใช้ 8255 2 ตัว

A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>	A <sub>7</sub>	A <sub>6</sub>	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	External RAM
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0000H - 1FFFH
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	8255 ตัวที่ 1(U3)
0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	2000 - 2003H
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8255 ตัวที่ 2(U7)
0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	4000 - 4003H



รูปที่ 3.2 แสดงวงจรภาคส่วนควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### การต่อ Port ใช้งานมีดังนี้

- Port 0.0 - Port 0.7 เป็น Input Data/Output Data และ Address (A0-A7) ในการติดต่อกับหน่วยความจำภายนอก (RAM)

- Port 1.0 ใช้เช็คสัญญาณ Ringing
- Port 1.1 เป็น Output Data ใช้ในการยกหูและวางหูระบบโทรศัพท์
- Port 1.2 เป็น Output Data กำหนด Address A0 ของ 8255
- Port 1.3 - Port 1.5 เป็น Output Data ควบคุมการใช้ LCD
- Port 1.6 เป็น Output Data กำหนด Address A1 ของ 8255
- Port 1.7 เป็น Output Drive ให้กับทรานซิสเตอร์ On-Off เพื่อให้ Relay ตัดต่อไฟให้วาล์ว

### น้ำทำการเปิด-ปิด

- Port 2.0-2.7 เป็น Address (A8-A10) ติดต่อกับหน่วยความจำภายนอก
- Port 2.5-2.7 เป็น Output ใช้ Decode กำหนดตำแหน่งการทำงานของ External RAM และ 8255
- Port 3.0-3.4 เป็น I/P ใช้งานกับเครื่องอ่าน การ์ด
- Port 3.5 เป็น I/O Sensor ตรวจสอบการไหลของน้ำ

### การใช้งาน 8255 (U3)

PA0-PA7 เป็น Output Data ออก LCD

PC0-PC3 เป็น Output สแกน Key

PC4-PC7 เป็น Input รับการกด Key

### การใช้ SN 74S373(U6)

- ใช้เป็นตัวแลตช์ Address ไบท์ค่า (A0-A7) เมื่อ CPU ต้องการติดต่อกับหน่วยความจำภายนอก โดยจะต่อสัญญาณแลตช์จากขา ALE ของ CPU เข้ากับขา LE ของตัวแลตช์

### การใช้ 74LS138

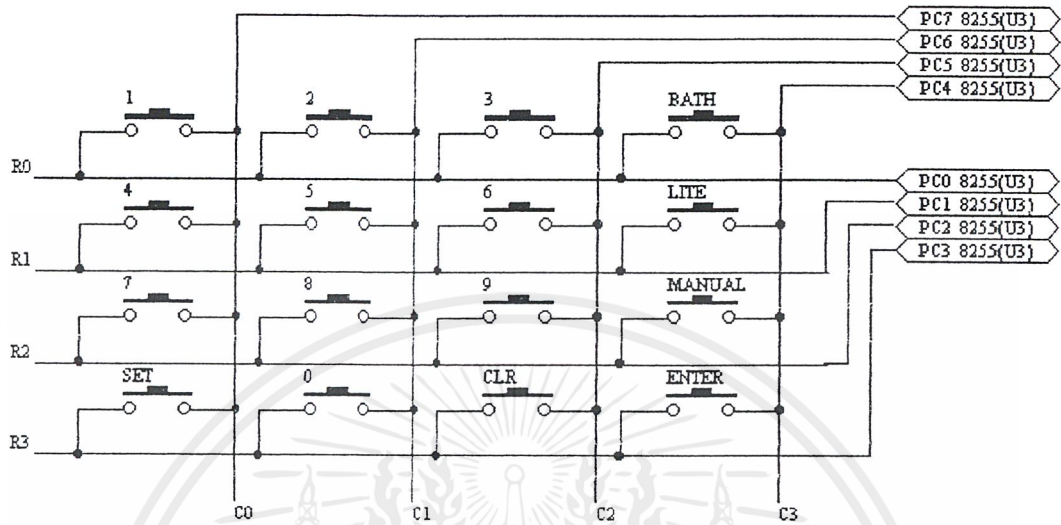
- ใช้เป็นตัว Decode ตำแหน่งการใช้งานของอุปกรณ์ภายนอก

### การใช้ MCM 6264 (8KB)

- ใช้เป็น RAM ภายนอกเก็บข้อมูลชั่วคราวเพื่อใช้ในการ Compare รหัส Card ของผู้ใช้งานกับข้อมูลที่มีอยู่ในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.2 การออกแบบ Key Board



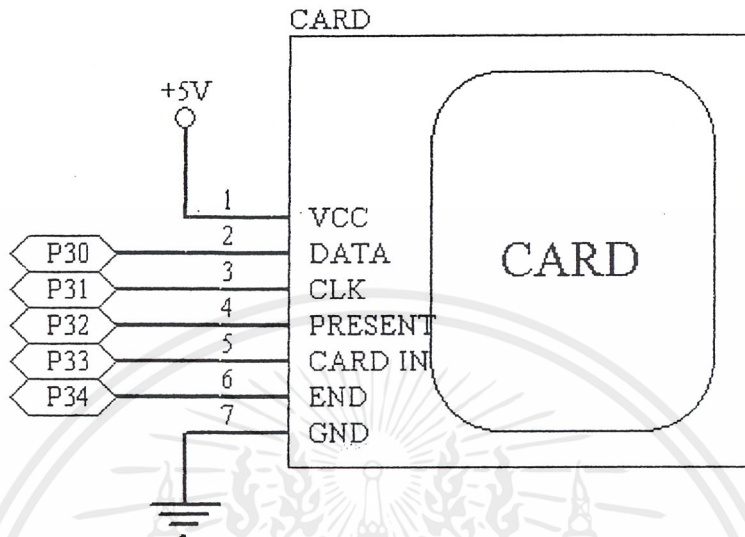
รูปที่ 3.3 แสดงลักษณะการต่อ Keyboard ไปใช้งาน

#### การใช้งาน KEYBOARD

- ใช้ PC0 ของ 8255 (U3) ทำการสแกนคีย์ทาง R0
- ใช้ PC0 ของ 8255 (U3) ทำการสแกนคีย์ทาง R0
- ใช้ PC0 ของ 8255 (U3) ทำการสแกนคีย์ทาง R0
- ใช้ PC0 ของ 8255 (U3) ทำการสแกนคีย์ทาง R0
- ใช้ PC0 ของ 8255 (U3) ทำการสแกนคีย์ทาง C0
- ใช้ PC0 ของ 8255 (U3) ทำการสแกนคีย์ทาง C0
- ใช้ PC0 ของ 8255 (U3) ทำการสแกนคีย์ทาง C0
- ใช้ PC0 ของ 8255 (U3) ทำการสแกนคีย์ทาง C0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.3 การต่อใช้งานเครื่องอ่านบัตรแม่เหล็ก



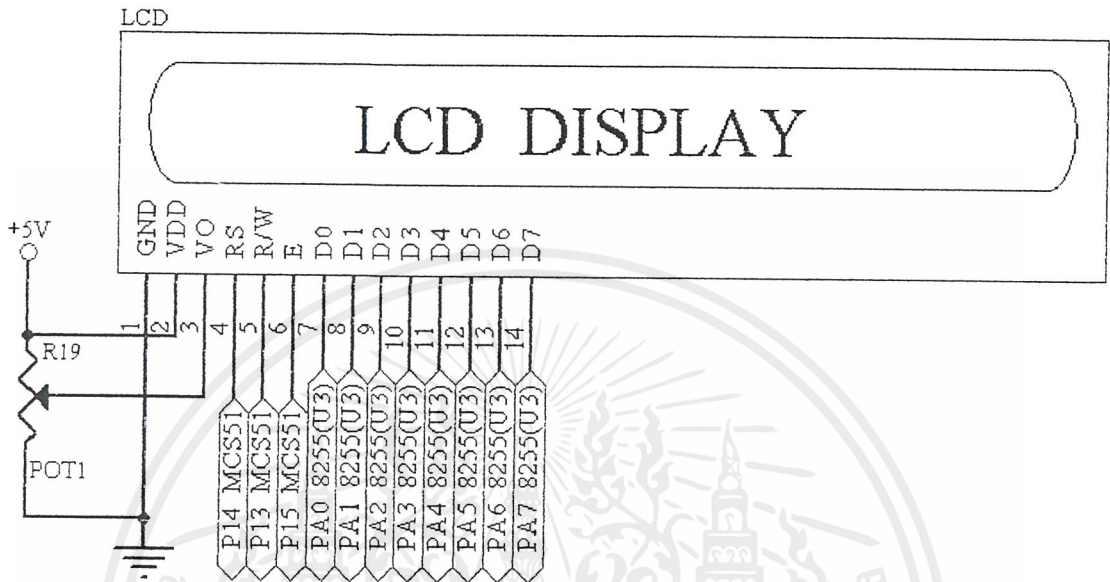
รูปที่ 3.4 แสดงการต่อเครื่องอ่านบัตรแม่เหล็กไปใช้งาน

#### การต่อใช้งานมีดังนี้

- ใช้ Port 3.0 รับข้อมูลที่ส่งออกมาจากการอ่านบัตร
- ใช้ Port 3.1 รับ Clock จากเครื่องอ่านบัตรเพื่อใช้เป็นสัญญาณอ้างอิงในการอ่านข้อมูลจากบัตร
- ใช้ Port 3.2 รับสัญญาณจากเครื่องอ่านบัตรแม่เหล็กบอกให้ CPU รู้เพื่อทำการอ่านข้อมูลปกติสัญญาณจะเป็น 1 เมื่อมีการส่งข้อมูล สัญญาณนี้จะเป็น 0
- ใช้ Port 3.3 รับสัญญาณจากอุปกรณ์ตรวจจับของเครื่องอ่านบัตรปกติสัญญาณมีค่าเป็น 1 แต่เมื่อมีการเสียบบัตรสัญญาณนี้จะมีค่าเป็น 0
- ใช้ Port 3.4 รับสัญญาณจากอุปกรณ์ตรวจจับที่อยู่ด้านในสุดของเครื่องอ่านบัตร บอกให้ CPU รู้ว่ามีการเสียบบัตรเข้าไปสุดแล้ว

### 3.2.4 การต่อใช้งาน LCD

จะใช้ LCD ขนาด 16 ตัวอักษร 2 บรรทัดแบบ Dot Maxtix 5×7



รูปที่ 3.5 แสดงการต่อ LCD ไปใช้งาน

#### การต่อใช้งาน LCD

- ขา VDD ต่อไฟ +5 V
- ขา VO ต่อไฟ +5 V ผ่าน VR เพื่อปรับความสว่างของจอ
- ใช้ Port 1.3 ควบคุมการอ่านหรือเขียน LCD
- ใช้ Port 1.4 ควบคุม LCD บอกให้ LCD รู้ว่าข้อมูลที่ส่งมาให้เป็นข้อมูลคำสั่งหรือ Data ที่ใช้

ในการแสดงผล

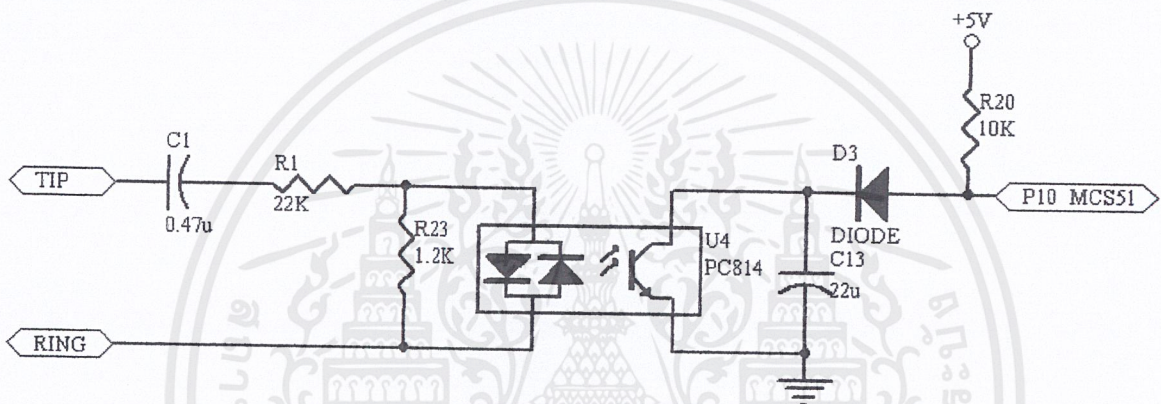
- ใช้ Port 1.5 ควบคุมให้ LCD ทำงาน
- ขา D0-D7 รับข้อมูลที่เป็นข้อมูลคำสั่ง หรือ ข้อมูลที่จะให้แสดงผล จาก Data bus โดยผ่าน

Port A ของ 8255 (U3)

3.2.5 การออกแบบภาคตรวจจับสัญญาณ Ringing และถอดรหัสความถี่ DTMF และระบบตอบกลับ ส่วนนี้จะเป็นส่วนที่เป็นระบบ เกี่ยวข้องกับสัญญาณ โทรศัพท์ และอุปกรณ์ที่ใช้ในการควบคุมต่างๆ ซึ่งจะประกอบด้วยส่วนหลักๆ 3 ส่วนคือ

### 1. วงจรตรวจจับสัญญาณกริ่งเรียก

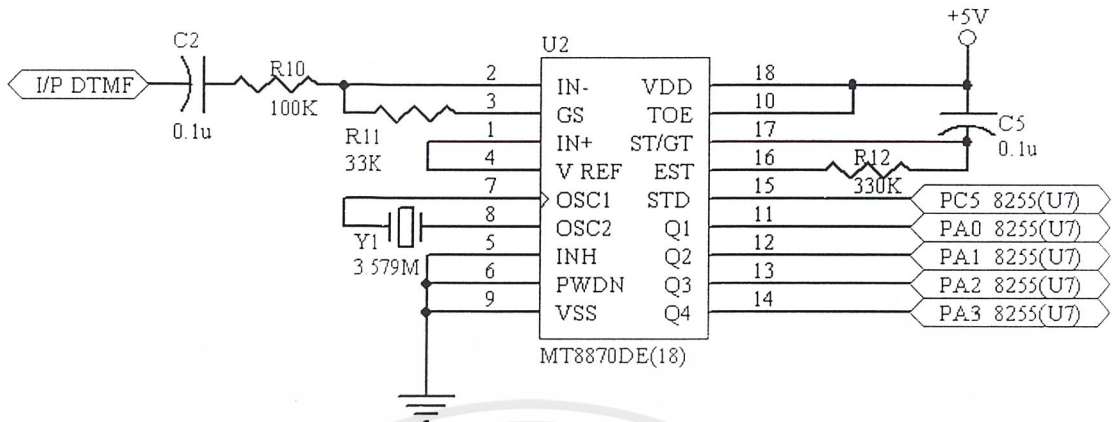
วงจรตรวจจับสัญญาณกริ่งเรียกนี้มีอุปกรณ์หลักที่ใช้ในการตรวจสอบได้แก่ Opto isolator เบอร์ PC 814 ซึ่งภายในเป็น Transistor แบบ NPN มีตัวเก็บประจุป้องกันไฟฟ้ากระแสตรงไม่ให้เข้ามารบกวน เพราะจะทำให้ตรวจจับสัญญาณได้ยากและมีตัวต้านทานต่ออยู่ทางด้านอินพุตเพื่อกำหนดแรงดันที่เหมาะสมในการป้อนให้กับ LED ภายในตัว Opto isolator



รูปที่ 3.6 วงจรตรวจจับสัญญาณกริ่งเรียก

### 2. วงจรถอดรหัสสัญญาณความถี่ DTMF

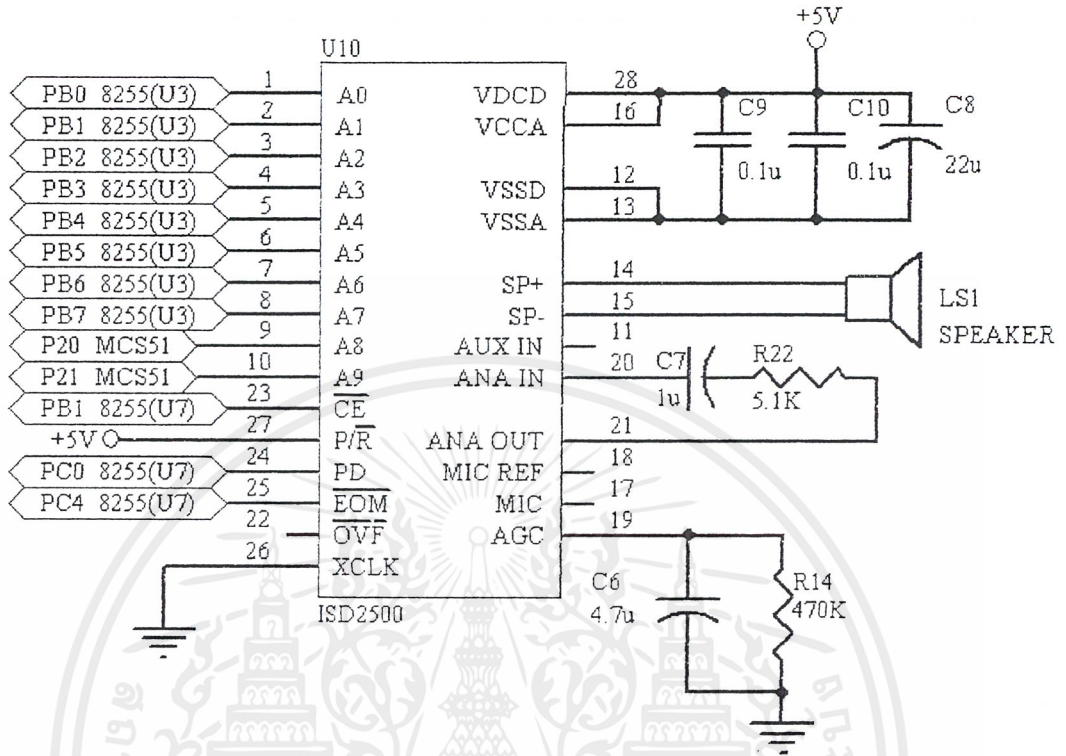
วงจรถอดรหัสสัญญาณความถี่ DTMF จะมีหน้าที่ตรวจสอบความถี่ที่ได้จากการกดคีย์หมายเลขจากโทรศัพท์ของผู้เรียกเข้า อุปกรณ์ที่สำคัญได้แก่ ไอซีเบอร์ MT8870 ซึ่งจะเป็นตัวถอดรหัสความถี่ DTMF ของโทรศัพท์ โดยรหัสความถี่ที่ได้ออกมาจะเป็นเลขไบนารี แล้วนำไปป้อนให้กับระบบควบคุมทำการตรวจสอบ และหน่วยประมวลผลเพื่อที่จะสั่งงานให้ระบบอื่นทำงานตามการกดคีย์ การออกแบบวงจรถอดรหัสความถี่ DTMF สามารถทำได้โดยอาศัยวงจรพื้นฐานของ MT8870



รูปที่ 3.7 วงจรถอดรหัสสัญญาณความถี่ DTMF

### 3. วงจรสร้างเสียงตอบกลับทางโทรศัพท์

วงจรสร้างเสียงตอบกลับ เป็นวงจรที่ใช้ทำหน้าที่บอกขั้นตอนการทำงานผ่านระบบโทรศัพท์ อุปกรณ์หลักจะใช้ไอซีบันทึกเสียงเบอร์ ISD2590 จะใช้การต่อวงจรดังรูปที่ 3.8 ซึ่งจะเป็นวงจรพื้นฐานที่ใช้กับไอซีตัวนี้ โดยจะทำการบันทึกเสียงไว้ที่ Address ต่างๆ ของไอซีก่อน จากนั้นเวลาใช้งานจะทำการเลือกข้อความที่บันทึกไว้ Address นั้นๆ มาทำการเล่นกลับแล้วส่งสัญญาณผ่านออกไปสู่คู่สายโทรศัพท์ บอกให้ทราบขั้นตอนต่างๆ ที่เครื่องต้องการรับคำสั่ง



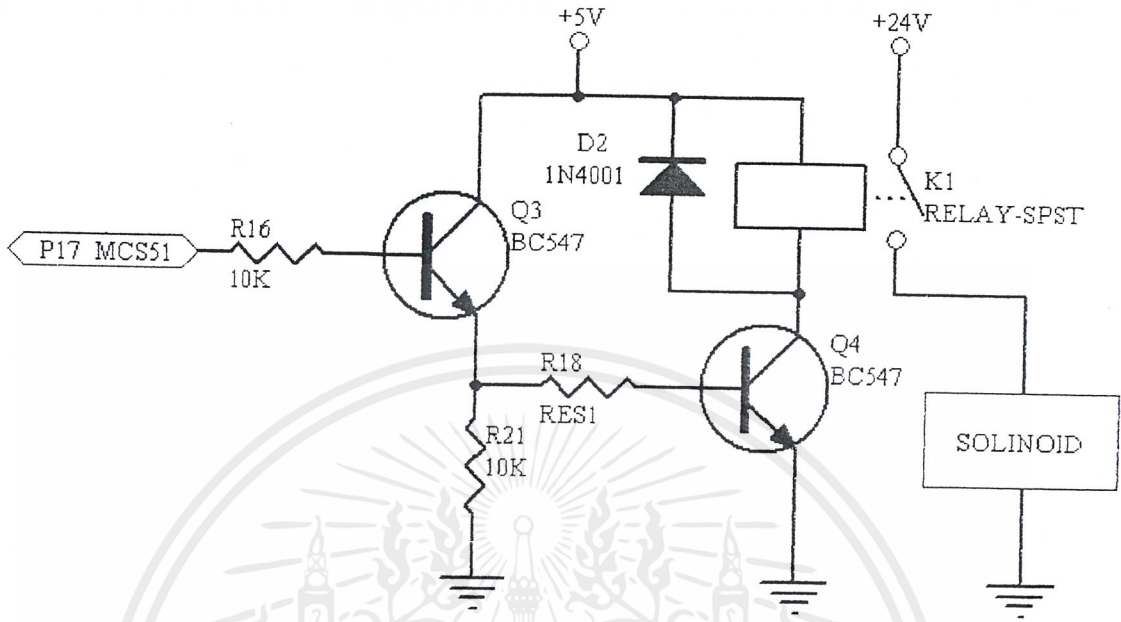
รูปที่ 3.8 วงจรสร้างเสียงตอบกลับโดยใช้ไอซีเบอร์ ISD2590

3.2.6 การออกแบบหน่วยตรวจสอบการจ่ายน้ำ

หน้าที่หลักของส่วนนี้คือจะคอยปิด/เปิดวาล์วให้น้ำไหลหรือไม่ไหล และยังคงเช็คปริมาณการไหลของน้ำ แล้วจะส่งสัญญาณให้ระบบควบคุมทราบ ระบบควบคุมจะทำการประมวลผลและสั่งงานให้ระบบต่างๆทำงานขึ้นตอนต่อไป หน่วยตรวจสอบการจ่ายน้ำรวมการทำงานของ 3 ส่วนดังนี้

1. ชุดวงจรขับโซลินอยด์วาล์ว

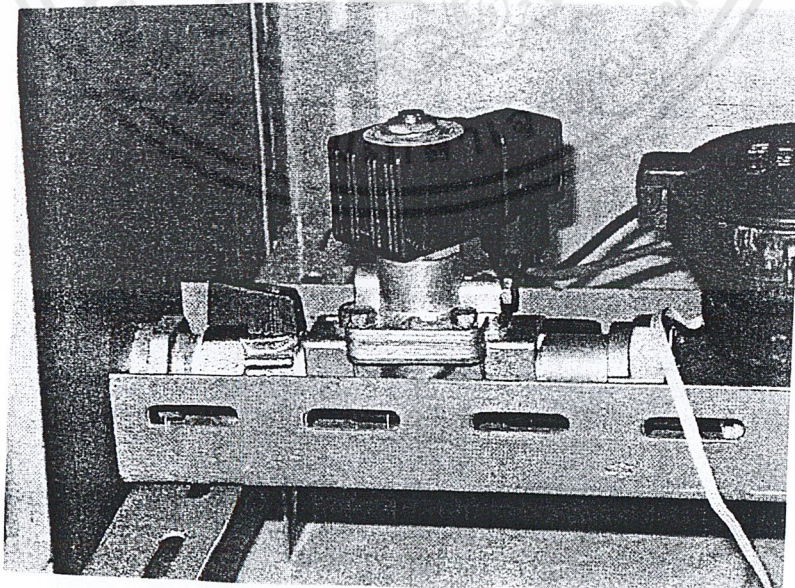
วงจรขับโซลินอยด์วาล์วนี้ซึ่งจะเป็นวงขับรีเลย์ โดยไมโครคอนโทรลเลอร์จะสั่งงานสภาวะลอจิก High ส่งผ่านไปให้ทรานซิสเตอร์นำกระแสเพื่อทำให้รีเลย์ทำงาน หน้าสัมผัสทำการต่อไฟฟ้าเป็นผลทำให้โซลินอยด์วาล์วทำงาน



รูปที่ 3.9 แสดงวงจรการควบคุมวาล์วปิด/เปิดน้ำ

## 2. โซลินอยด์วาล์ว

โซลินอยด์วาล์วจะเป็นตัวเปิดและปิดการไหลของน้ำ โดยการทำงานจะอาศัยแรงดันขนาด 24 โวลต์ ในการเปิด/ปิดวาล์ว

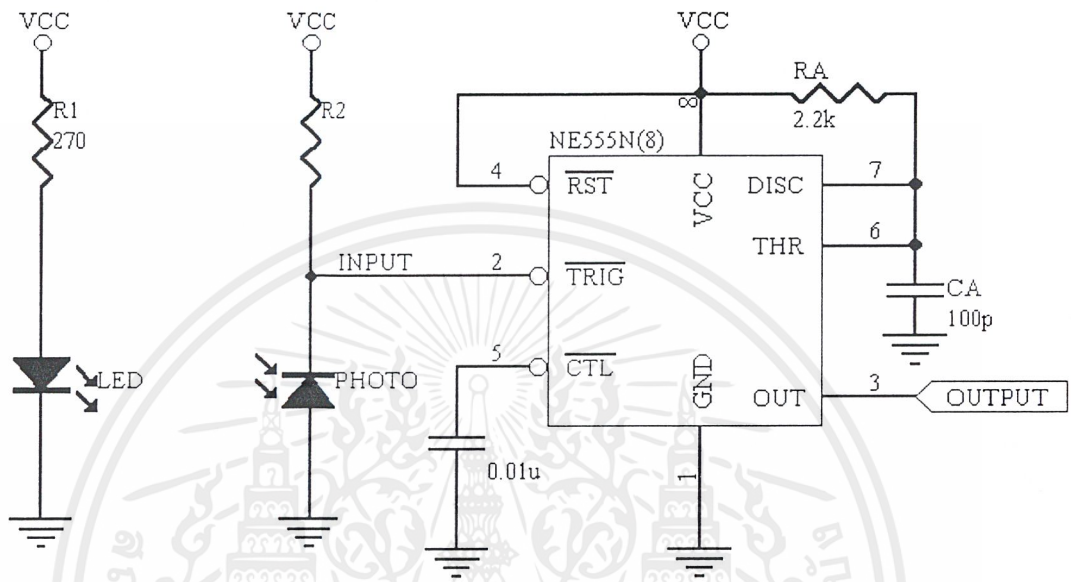


รูปที่ 3.10 แสดงลักษณะของโซลินอยด์วาล์ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. ชุดเซ็นเซอร์

ชุดเซ็นเซอร์นี้จะนำเอาไมโครคอนโทรลเลอร์มาทำการตัดแปลง โดยติดอุปกรณ์เซ็นเซอร์เข้าไป



รูปที่ 3.11 วงจรเซ็นเซอร์ปริมาณน้ำ

หลักการทำงานคือเมื่อน้ำไหลก็จะทำให้แกนซึ่งด้านล่างติดอยู่กับใบพัด ด้านบนติดกับแผ่น สำหรับสะท้อนแสง เมื่อน้ำไหลแผ่นสะท้อนแสงก็จะเกิดการหมุนแล้วสะท้อนแสงไปยัง Photo diode ทำให้ Photo diode นำกระแสเกิดเป็น Input Pulse Trig เข้าไปสู่วงจร Monostable เพื่อทำการสร้าง Pulse ที่มีระดับ Logic high และ Low ซึ่งจะส่งไปประมวลผลโดยไมโครคอนโทรลเลอร์

### 3.3 SOFT WARE

Soft Ware เป็นโปรแกรมควบคุมการทำงานของระบบในส่วนของเทอมนี้ โปรแกรมหลักๆมีดังนี้

#### 3.3.1 Program Set Password

ใช้ในการตั้งรหัสผ่าน 4 หลักให้กับเครื่อง เพื่อใช้สำหรับการเติมจำนวนเงินให้กับบัตรแต่ละใบ

#### 3.3.2 Program Key In

ใช้รับข้อมูลจากแป้น Key Board เข้ามาให้ระบบทำงานตาม Function Key นั้นๆ

#### 3.3.3 Program Check Card Insert & Read Data

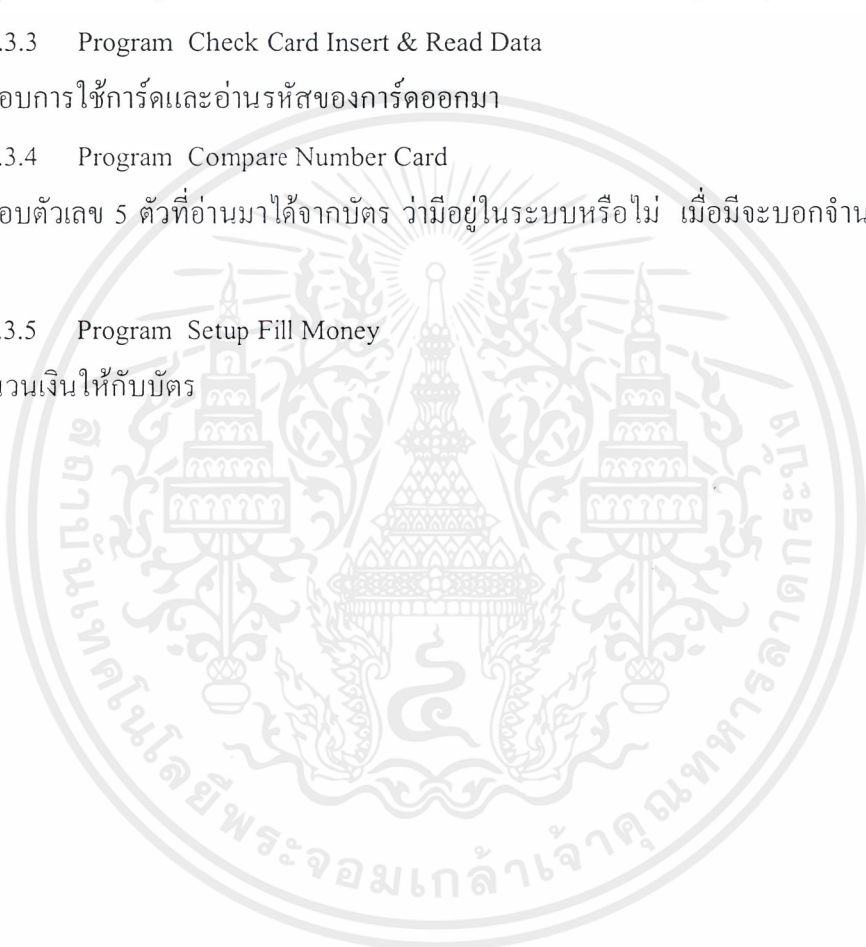
ใช้ตรวจสอบการใช้การ์ดและอ่านรหัสของการ์ดออกมา

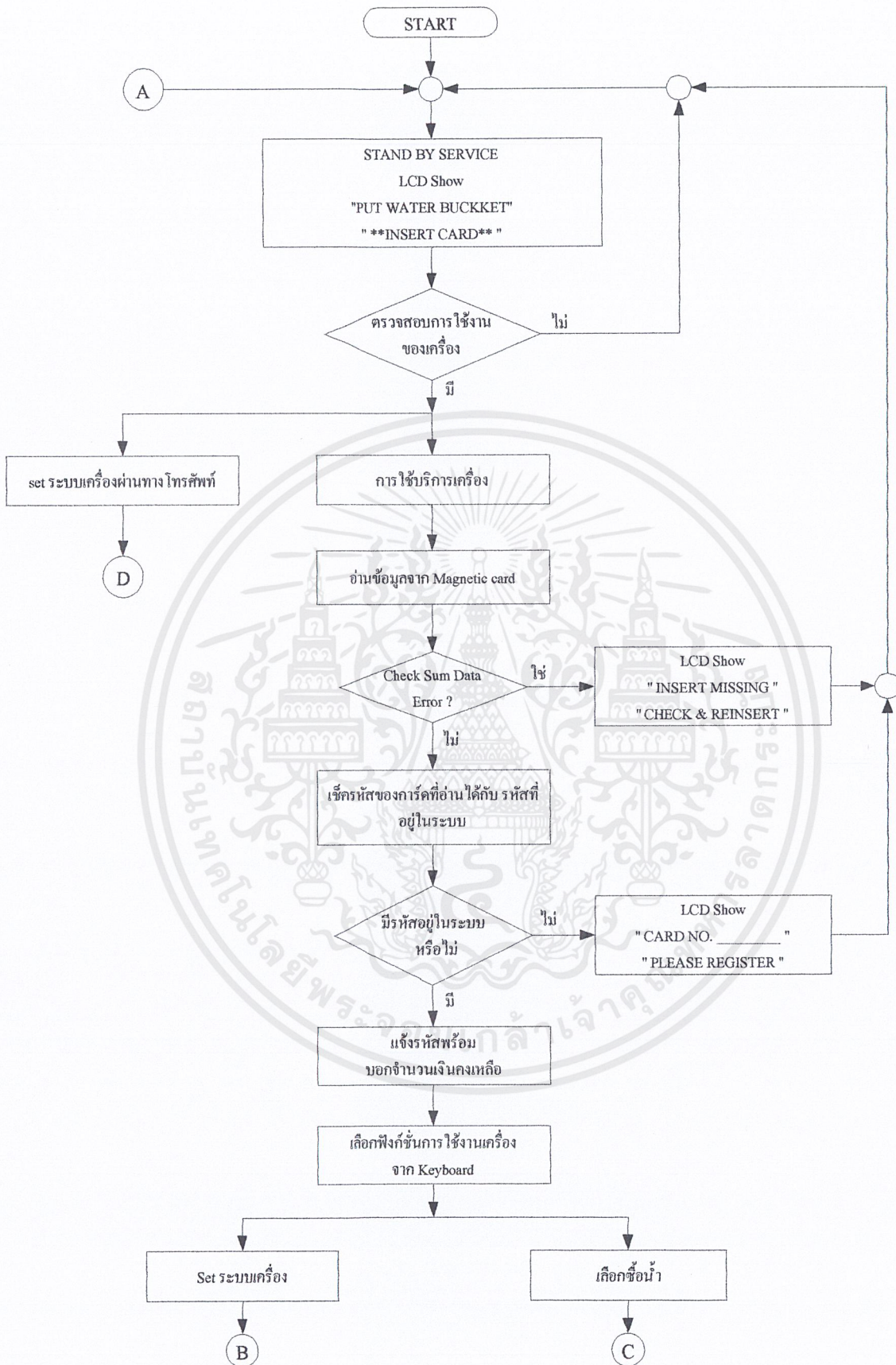
#### 3.3.4 Program Compare Number Card

ใช้ตรวจสอบตัวเลข 5 ตัวที่อ่านมาได้จากบัตร ว่ามีอยู่ในระบบหรือไม่ เมื่อมีจะบอกจำนวนเงิน ของบัตรนั้นๆ

#### 3.3.5 Program Setup Fill Money

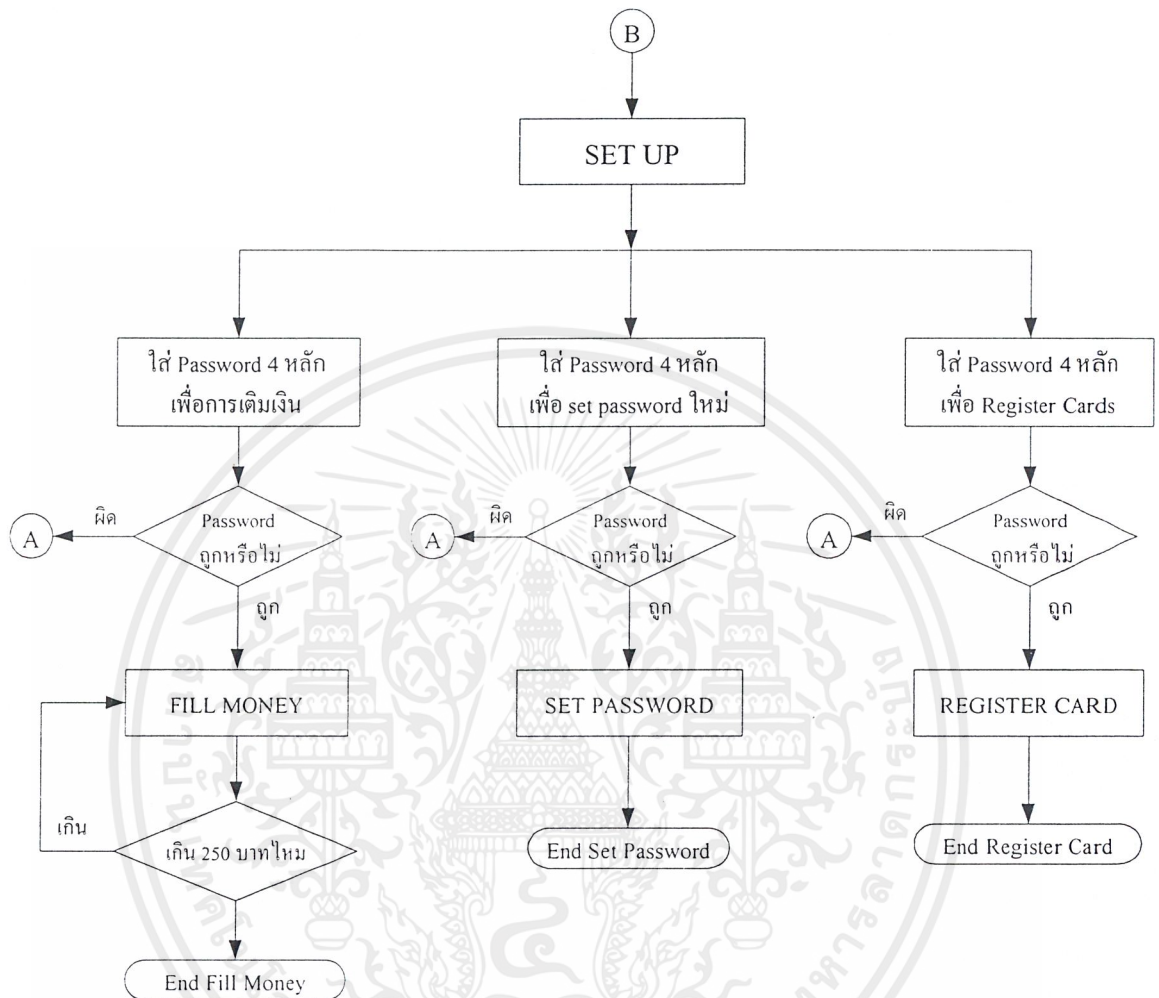
ใช้เติมจำนวนเงินให้กับบัตร





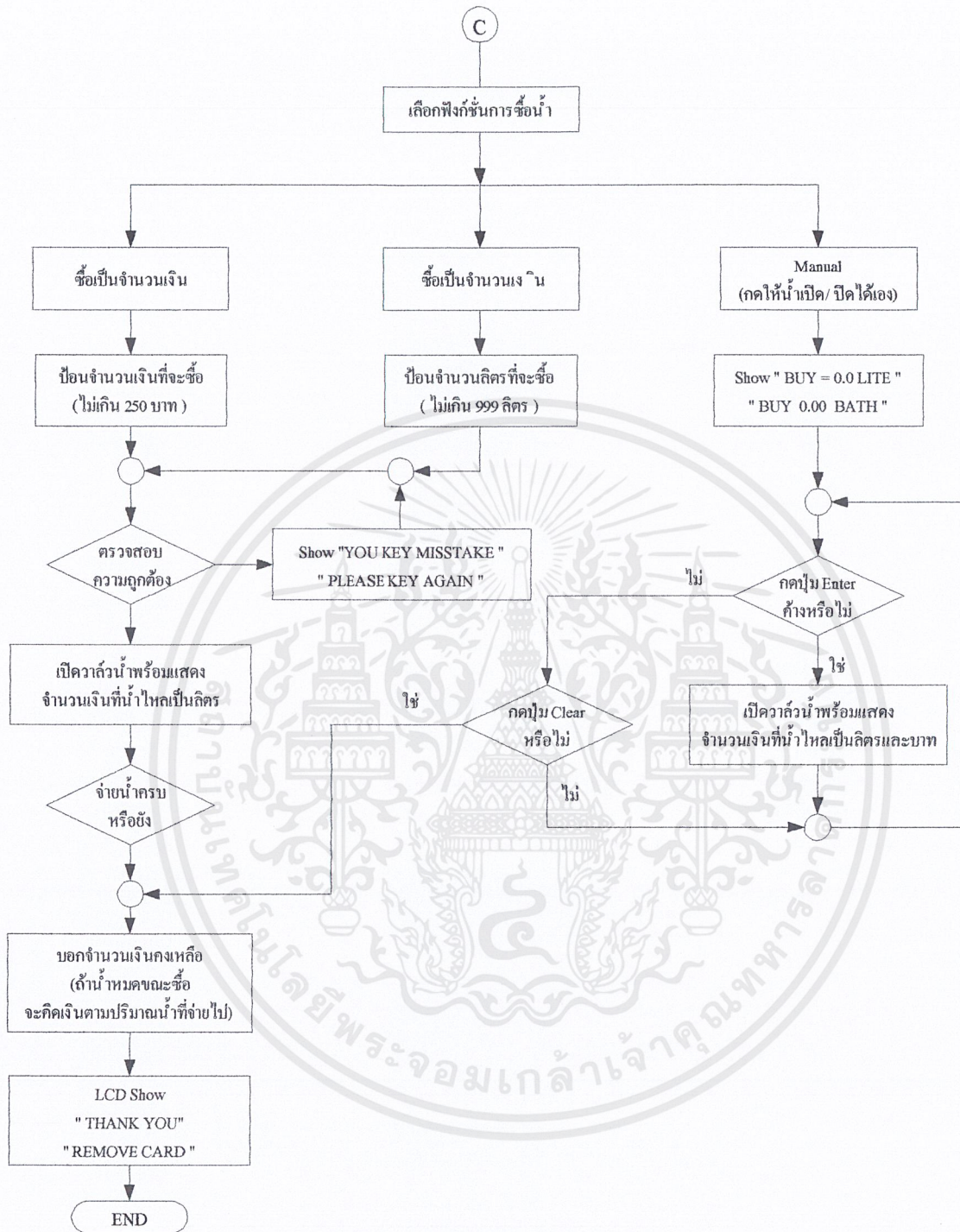
รูปที่ 3.12 Flow Chart แสดงการทำงานของระบบทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



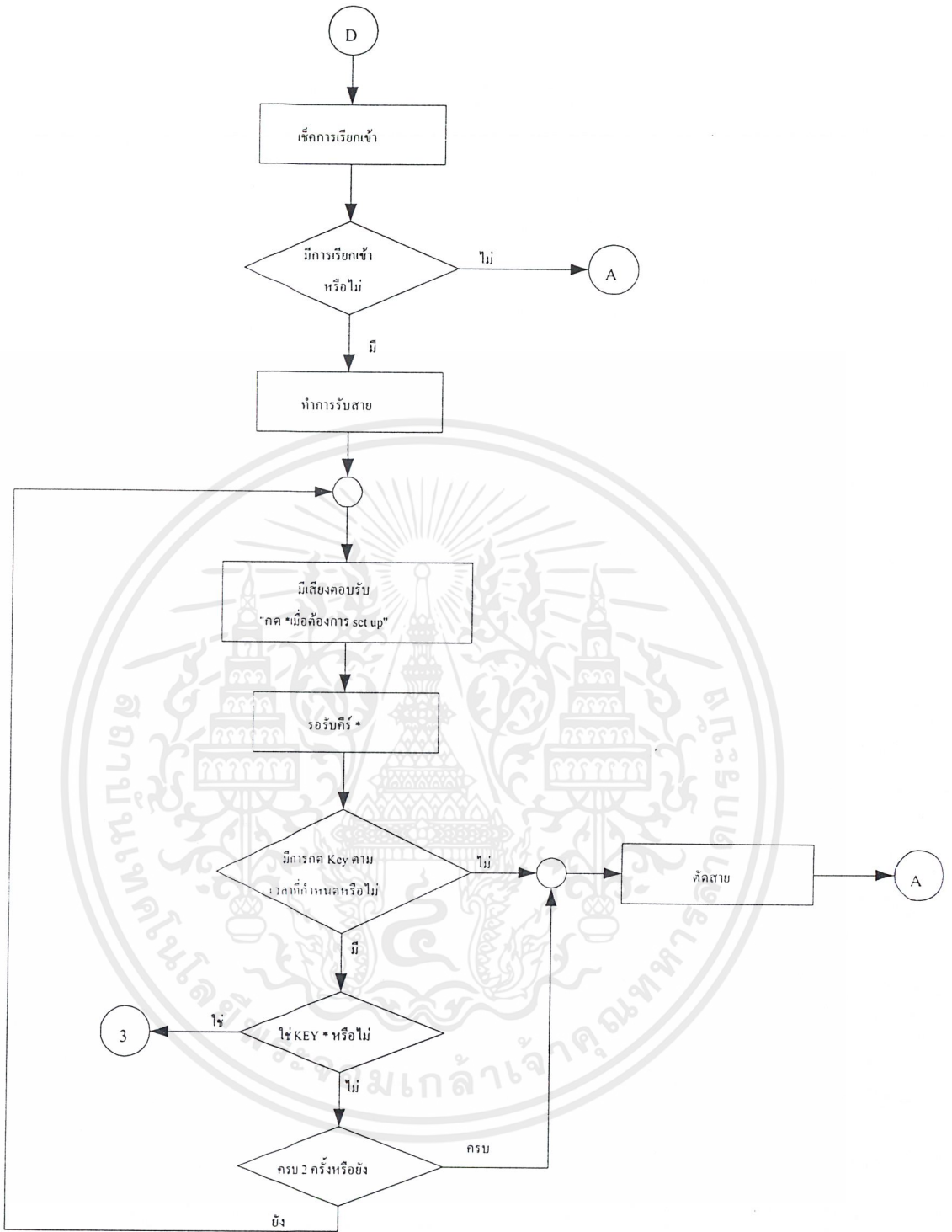
รูปที่ 3.13 Flow Chart แสดงการ SET ระบบเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



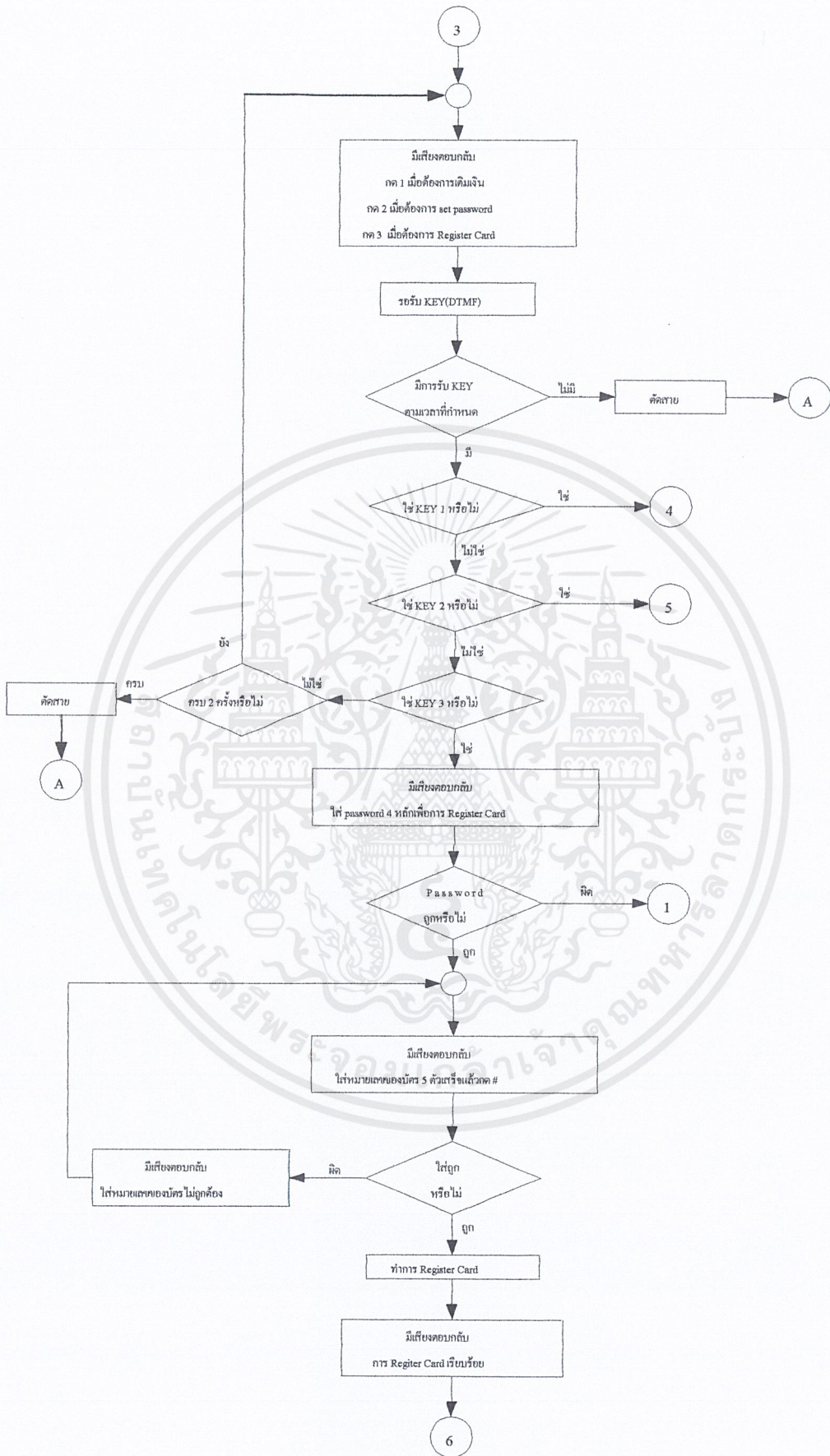
รูปที่ 3.14 Flow Chart แสดงการเลือกฟังก์ชันของการซื้อน้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



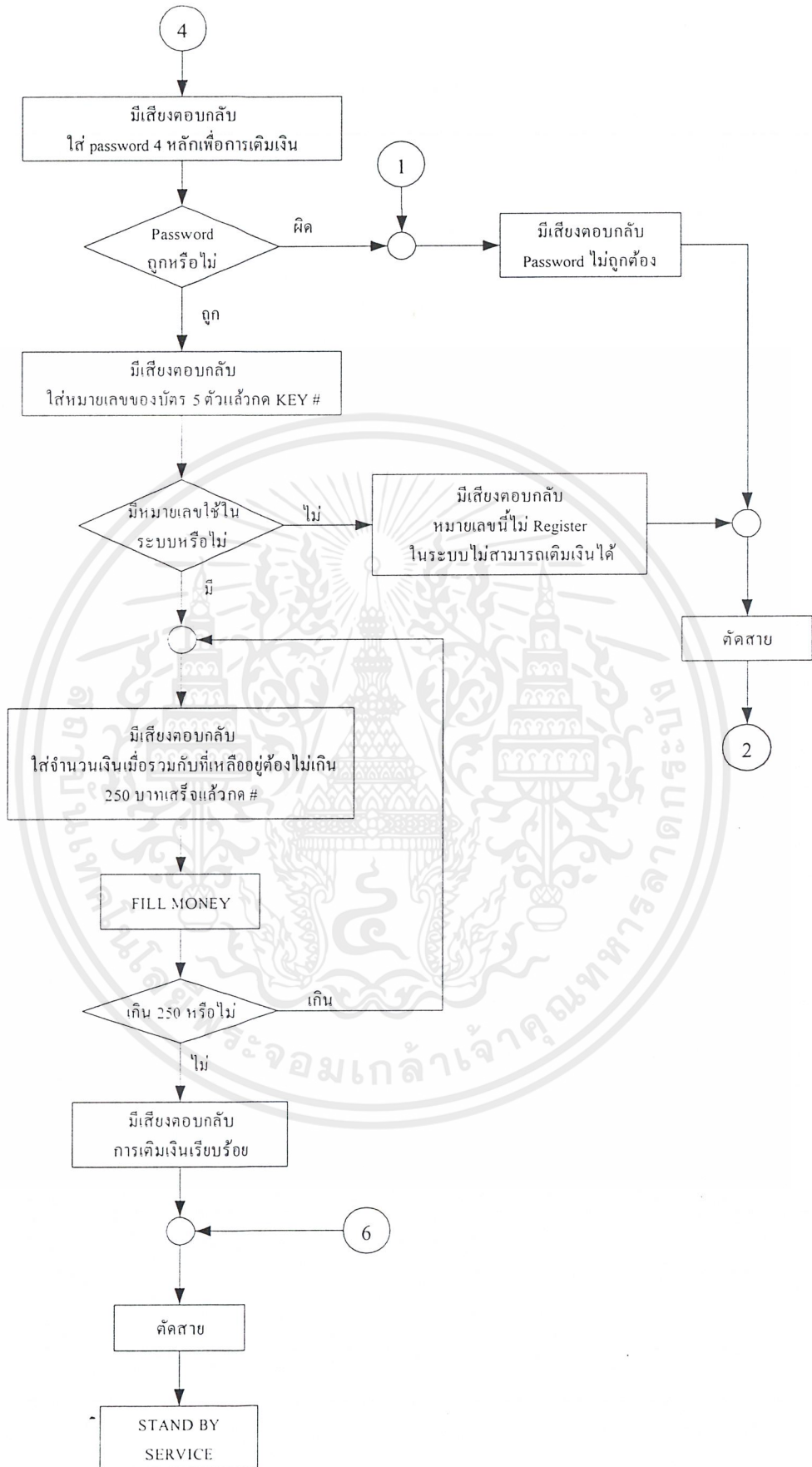
รูปที่ 3.15 Flow Chart แสดงการทำงานของโปรแกรม SET ระบบเครื่องทางโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



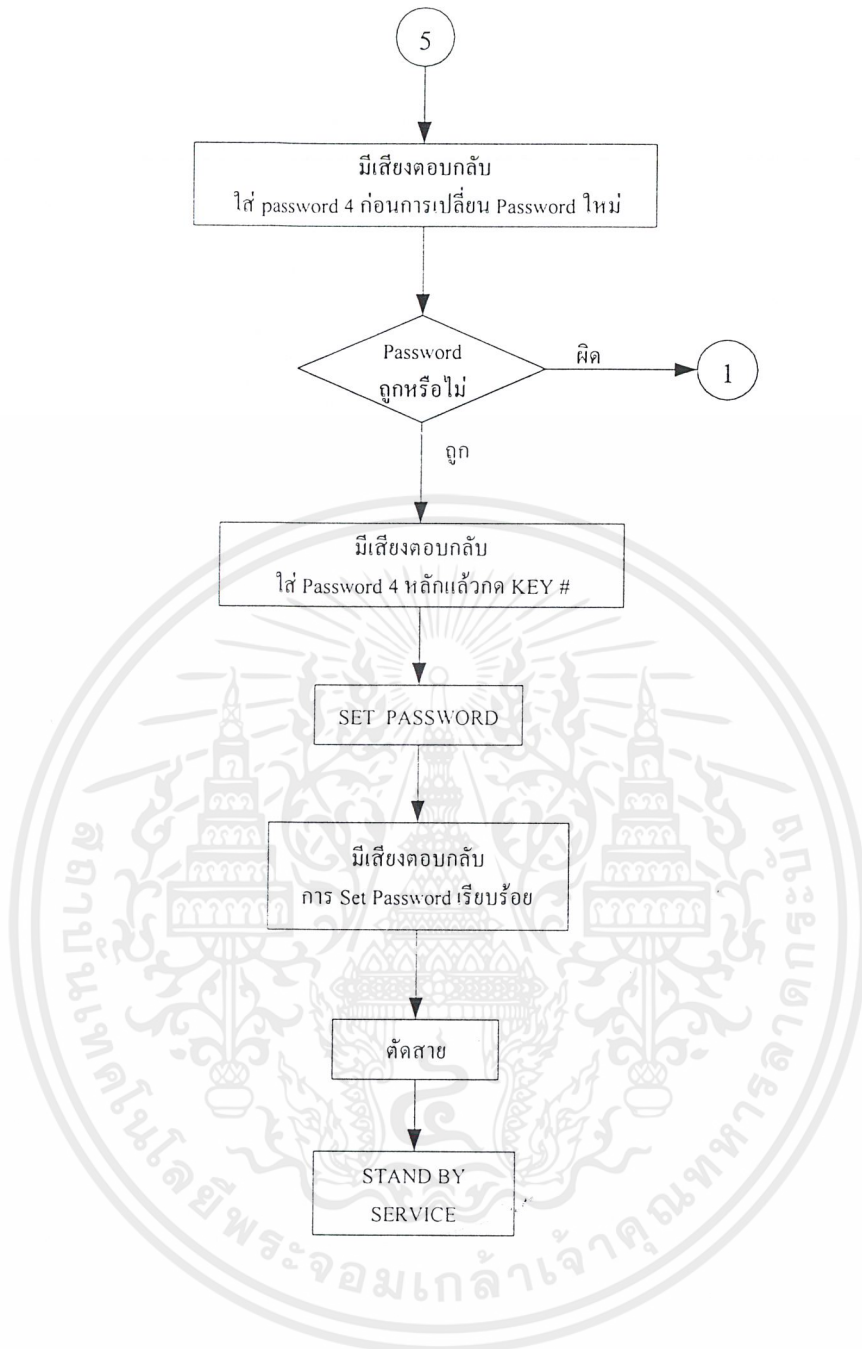
รูปที่ 3.15 (ต่อ) Flow Chart แสดงการทำงานของโปรแกรม SET ระบบเครื่องทางโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



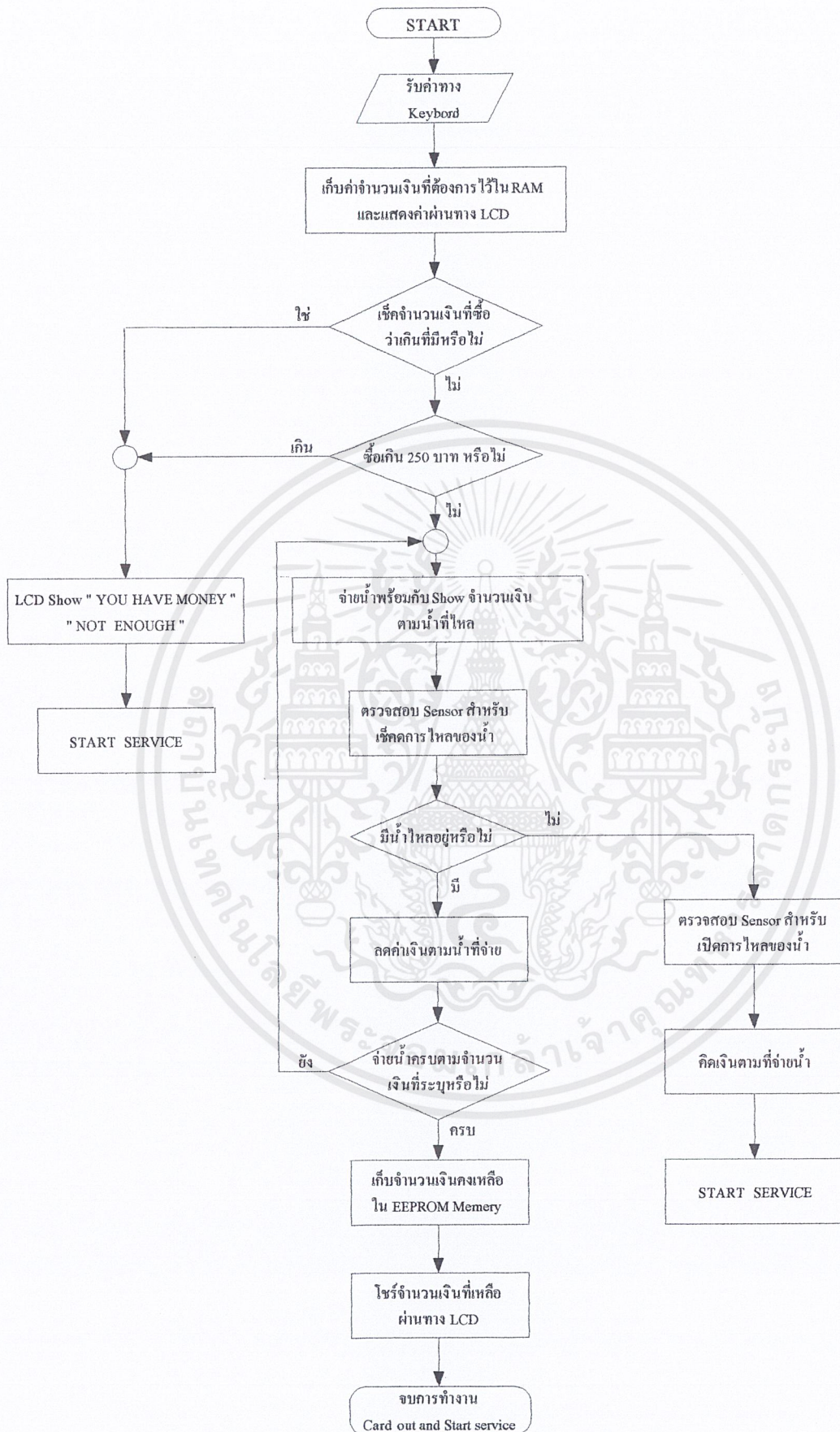
รูปที่ 3.15 (ต่อ) Flow Chart แสดงการทำงานของโปรแกรม SET ระบบเครื่องทางโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



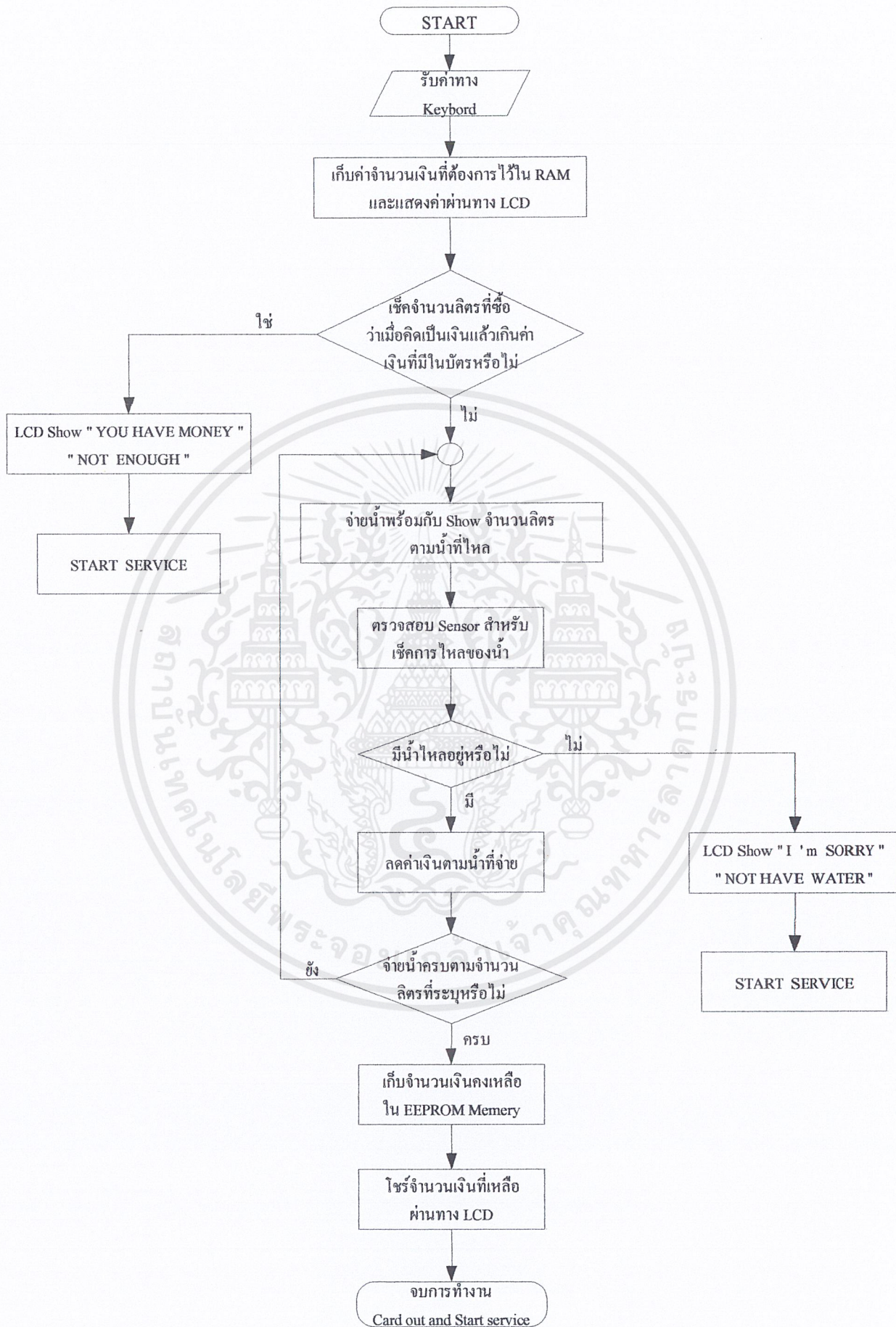
รูปที่ 3.15 (ต่อ) Flow Chart แสดงการทำงานของโปรแกรม SET ระบบเครื่องทางโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

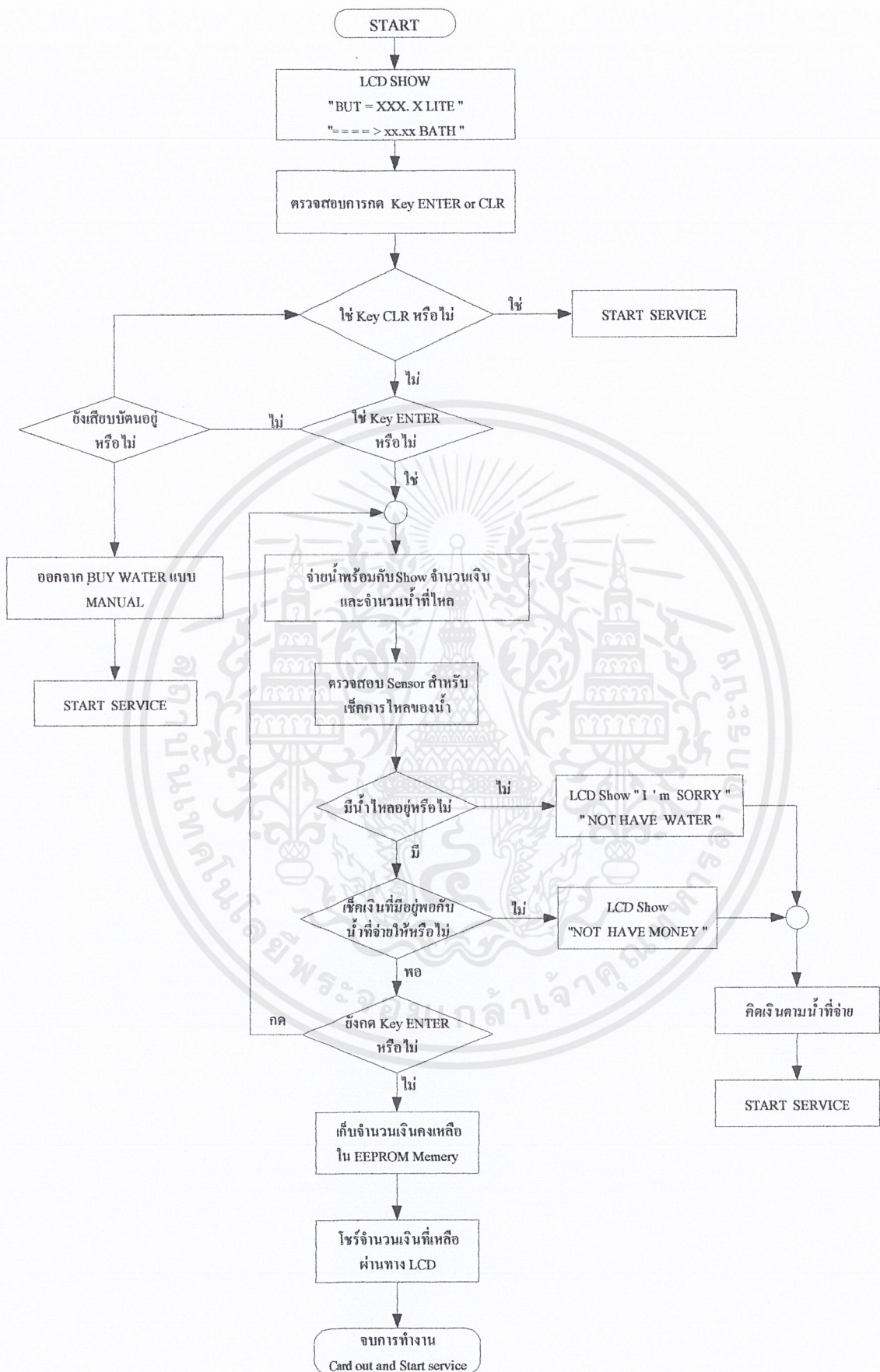


รูปที่ 3.16 Flow Chart แสดงการทำงานของโปรแกรมการชื้อน้ำแบบระบุจำนวนเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
**รูปที่ 3.17 Flow Chart แสดงการทำงานของโปรแกรมการชื้อน้ำแบบระบุจํนวนลิตร**  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 Flow Chart แสดงการทำงานของโปรแกรมการซื้อน้ำแบบ Manual

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการทดลอง

#### 4.1 คำนำ

ในบทนี้จะกล่าวถึงผลการทดลองของการทำงานแต่ละส่วน เพื่อเป็นแนวทางการปรับปรุงแก้ไข  
โครงการ ซึ่งพอสรุปได้ดังนี้

#### 4.2 ภาค Control

ภาค Control จะประกอบด้วยส่วนประมวลผลกลางที่ใช้ AT8998252 ต่อร่วมกับอุปกรณ์ Interface  
External Ram (8 KB), Decoder (74LS138), Latch (74S373) และ Input/Output Port (8255)

ส่วนประมวลผลกลาง เป็นหัวใจสำคัญของระบบ โดยจะอ่านโปรแกรมคำสั่งมาตีความหมายและ  
ปฏิบัติตามโปรแกรมคำสั่งนั้น

ดังนั้นเราสามารถจะทดสอบการทำงานของระบบภาคควบคุมนี้ได้โดยการเขียนโปรแกรมทดสอบ  
อย่างง่ายได้ดังนี้

```

ORG 0000H
MOV A,#0FFH
MOV DPTR,#1000H ;ให้ DPTR ชี้ตำแหน่ง Address Ram ภายนอก
MOVX @DPTR,A
CLR A
MOV DPTR,#1000H ;ให้ DPTR ชี้ตำแหน่ง Address Control Word ของ 8255
MOV A,#88H ;กำหนดให้ Port PB เป็น Output
MOVX @DPTR,A
MOV DPTR,#1000H ;ให้ DPTR ชี้ตำแหน่ง Address Ram ภายนอก
MOVX A,@DPTR
MOV DPTR,#2001H ;ให้ DPTR ชี้ Address Port B ของ 8255
MOVX @DPTR,A
END

```

ในการใช้ไมโครคอนโทรลเลอร์ตรวจเช็คขา PB0 - PB7 ทุกขาต้องมีโวลต์ตกคร่อมอยู่ในภาวะ Logic High ทุก  
ขาแสดงว่า ภาค Control ทำงานได้ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 ภาค INPUT

#### 4.3.1 KEY BOARD

สำหรับ KEY BOARD นี้ การตรวจเช็คทำได้ง่าย คือ ใช้มัลติมิเตอร์ ย่านโอห์ม วัดตำแหน่งของ สวิตช์ที่ Key แต่ละค่า ผลที่ออกมาเมื่อเวลาเรากด Key คือค่าความต้านทานที่ได้จะเกือบเป็น ศูนย์โอห์ม แสดงว่า KEY BOARD สามารถทำงานได้จริง

#### 4.3.2 เครื่องอ่านบัตรแม่เหล็ก

สำหรับการทดสอบทำได้โดยต่อแหล่งจ่ายไฟ +5 V ให้กับเครื่องอ่านบัตรแม่เหล็กใช้มัลติมิเตอร์ ย่านโวลท์วัดที่ตำแหน่ง Card In จะได้ระดับ Logic เป็น High และถ้าเราเสียบบัตรเข้าไปในเครื่องจะต้อง ได้แรงดันระดับ Logic เป็น Low และถ้าวัดที่ขา Present จะได้แรงดันเป็นระดับ Logic Low ใช้ scope วัด สัญญาณ Clock จะต้องได้สัญญาณ Clock ออกมาที่ขา Clock ของเครื่องอ่านบัตรด้วย เมื่อผลเป็นอย่างนี้ แล้วแสดงว่าเครื่องอ่านบัตรแม่เหล็กทำงานได้

#### 4.3.3 วงจรตรวจจับสัญญาณ RINGING

สำหรับการทำงานของวงจร ทำได้โดยต่อคู่สายโทรศัพท์เข้าทาง Input ของวงจร แล้วใช้มัลติ มิเตอร์วัดแรงดันที่เอาต์พุทของวงจร โดยสังเกตว่ามีการเปลี่ยนระดับแรงดันขณะที่มีสัญญาณ Ringing เข้า มา และระดับแรงดันต้องเปลี่ยนอยู่ระดับลอจิก Low และ High ด้วยจึงจะทำให้ไมโครคอนโทรลเลอร์ ประมวลผลได้ ถ้าเป็นเช่นนี้ก็แสดงว่าวงจรที่ออกแบบมานั้นสามารถใช้งานได้

#### 4.3.4 วงจรถอดรหัสสัญญาณความถี่ DTMF

การทดลองการทำงานของวงจร ทำได้โดยต่อคู่สายโทรศัพท์เข้ากับวงจรแมตซ์อิมพีแดนซ์ของคู่ สายโทรศัพท์ จากนั้นก็เอาเอาต์พุทของวงจรแมตซ์อิมพีแดนซ์ต่อเข้ากับอินพุทของวงจรถอดรหัส ใช้ LED ต่อไว้ที่ Q1 – Q4 เพื่อแสดงผล และใช้มัลติมิเตอร์วัดแรงดันที่ขา STD จากนั้นทำการกดคีย์ โทรศัพท์ สังเกตที่ LED จะมีรหัสไบนารีออกมาและจะค้างอยู่ตลอดไปจนกว่าจะมีการกดคีย์ใหม่ รหัสไบนารีของคีย์ตัวใหม่จะออกมาแทน ข้อสังเกตอีกข้อหนึ่งที่ต้องสังเกตพร้อมกับเอาต์พุท Q1 – Q4 คือที่ขา STD ในช่วงปกติจะอยู่ระดับ High แต่เมื่อมีการส่งรหัสไบนารีของคีย์ใหม่มาทางที่ เอาต์พุท ระดับแรงดัน ที่ขา STD จะทำการเปลี่ยนระดับแรงดันมาเป็นลอจิก Low อยู่ชั่วขณะ จากนั้นจะเปลี่ยนเป็นลอจิก High เช่นเดิม ถ้าผลการทดลองที่ได้เป็นดังนี้ แสดงว่าวงจรที่ทำการออกแบบสามารถถอดรหัสความถี่ DTMF ได้ และสามารถใช้งานได้แน่นอน

#### 4.3.5 วงจรเซ็นเซอร์วัดปริมาณการไหลของน้ำ

จากการทดลองวงจรเซ็นเซอร์สามารถตรวจสอบการหมุนของแผ่นสะท้อนแสงอันเนื่องมาจากการไหลของน้ำได้- ซึ่งสามารถสร้าง พัลส์ ที่มีระดับ Logic high และ Low ที่เหมาะสมสำหรับนำไป ประมวลผลยังไมโครคอนโทรลเลอร์ต่อไป

#### 4.4 ภาค Output

##### 4.4.1 LCD Display (16 ตัวอักษร 2 บรรทัด )

สำหรับการทดสอบ LCD สามารถทำได้โดยเขียน โปรแกรมทดสอบ แล้วดูผลที่จ่อว่าเป็นไปตามที่เราต้องการหรือไม่

##### 4.4.2 ส่วนวงจรขับ โซลีนอยด์วาล์ว

สำหรับการทดสอบวงจรสามารถทำได้โดยเราป้อน Voltage ระดับลอจิก High ที่ขา R16 ซึ่งเป็นการไปอัสให้แก่วงจร จะทำให้วงจรทำงาน ผลที่ได้จะต้องทำให้ รีเลย์ต่อหน้าสัมผัส ส่งแรงดัน ไปขับโซลีนอยด์วาล์วต่อไป

##### 4.4.3 วงจรสร้างเสียงตอบกลับทางโทรศัพท์

การทดลองทำโดยต่อวงจรตามวงจรพื้นฐาน แล้วทำการบันทึกเสียงในแต่ละข้อความใน Address ที่ต่างๆ กัน จากนั้นจะมาทำการเลือกที่ Address แล้วทำการเล่นกลับฟังข้อความว่าถูกต้องหรือไม่ ถ้าถูกต้องแสดงว่าวงจรสร้างเสียงตอบกลับทางโทรศัพท์สามารถใช้งานได้



## บทที่ 5

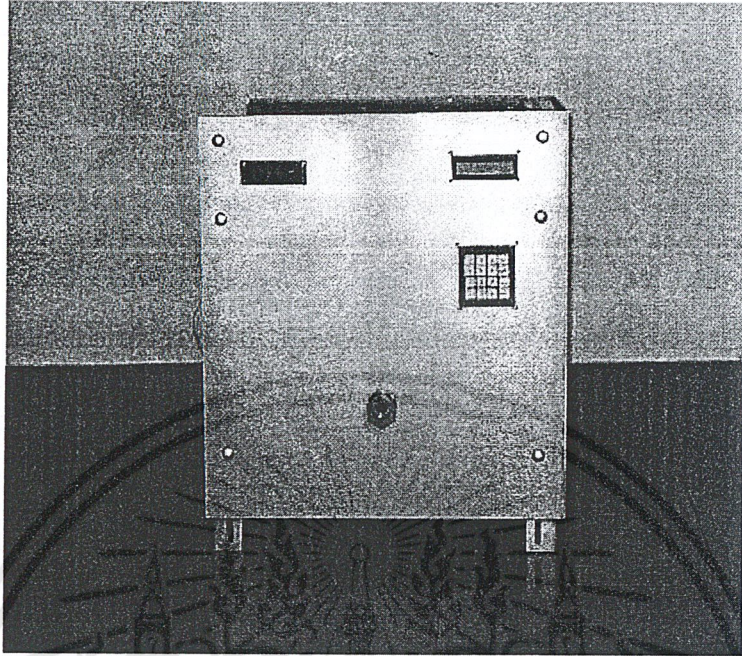
## สรุปและวิจารณ์ผลการทดลอง

การทำงานในภาคควบคุมสามารถประมวลผลและติดต่อกับส่วน Input และ Out put ของระบบได้เป็นอย่างดี ส่วนภาคตอบรับโทรศัพท์ ที่พบปัญหาคือ ถ้าความแรงของสัญญาณ DTMF น้อยจะทำให้การตรวจสอบสัญญาณ DTMF ไม่สามารถกระทำได้ ในส่วนเซ็นเซอร์วัดปริมาณการไหลของน้ำปัญหาที่พบ คือ ถ้าหากแรงดันน้ำที่ไหลเข้าเครื่องมีแรงดันไม่มากพอจะทำให้การตรวจสอบปริมาณน้ำของวงจรเซ็นเซอร์เกิดการผิดพลาด และในทางกลับกัน หากน้ำมีแรงดันที่มากพออีกทั้งยังมีระดับแรงดันที่คงที่จะทำให้การตรวจสอบปริมาณการไหลของน้ำเป็นไปได้ด้วยดี

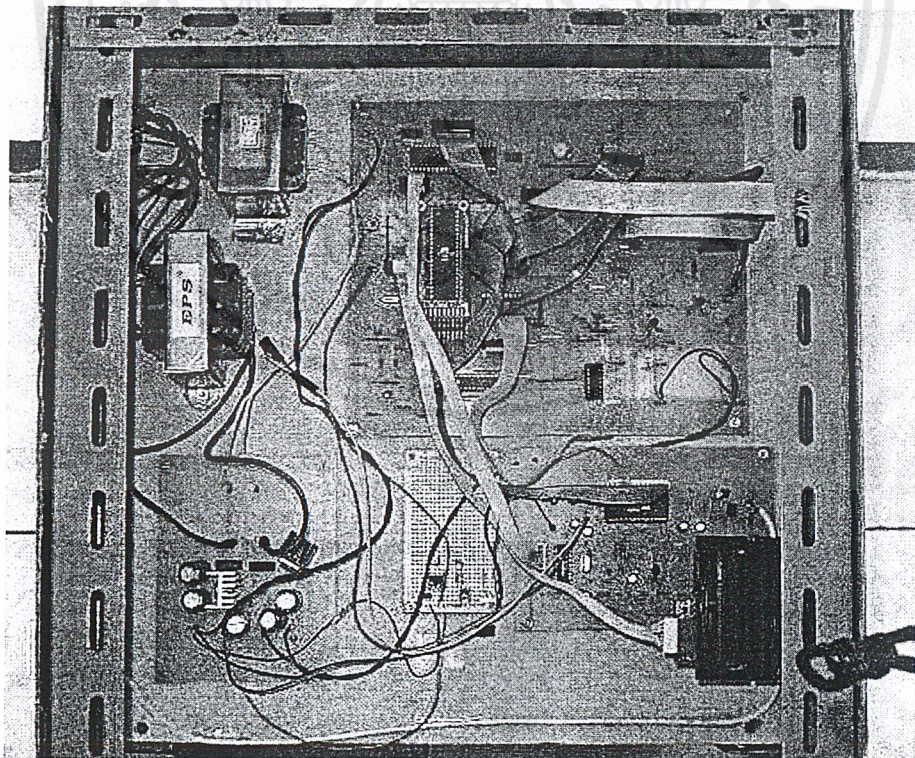




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

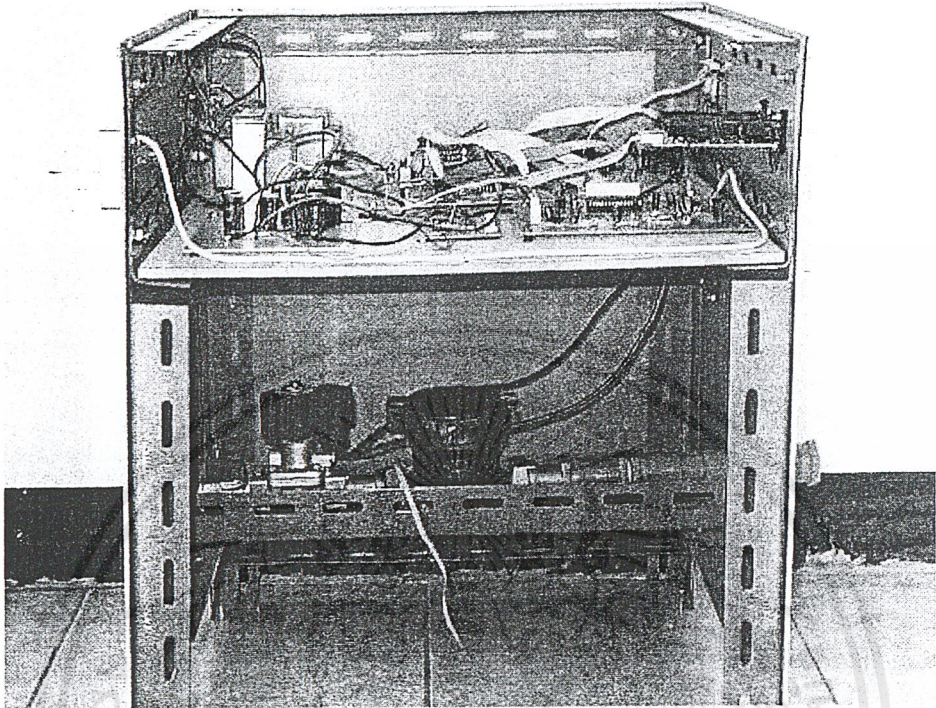


รูปแสดงด้านหน้าของเครื่องขายน้



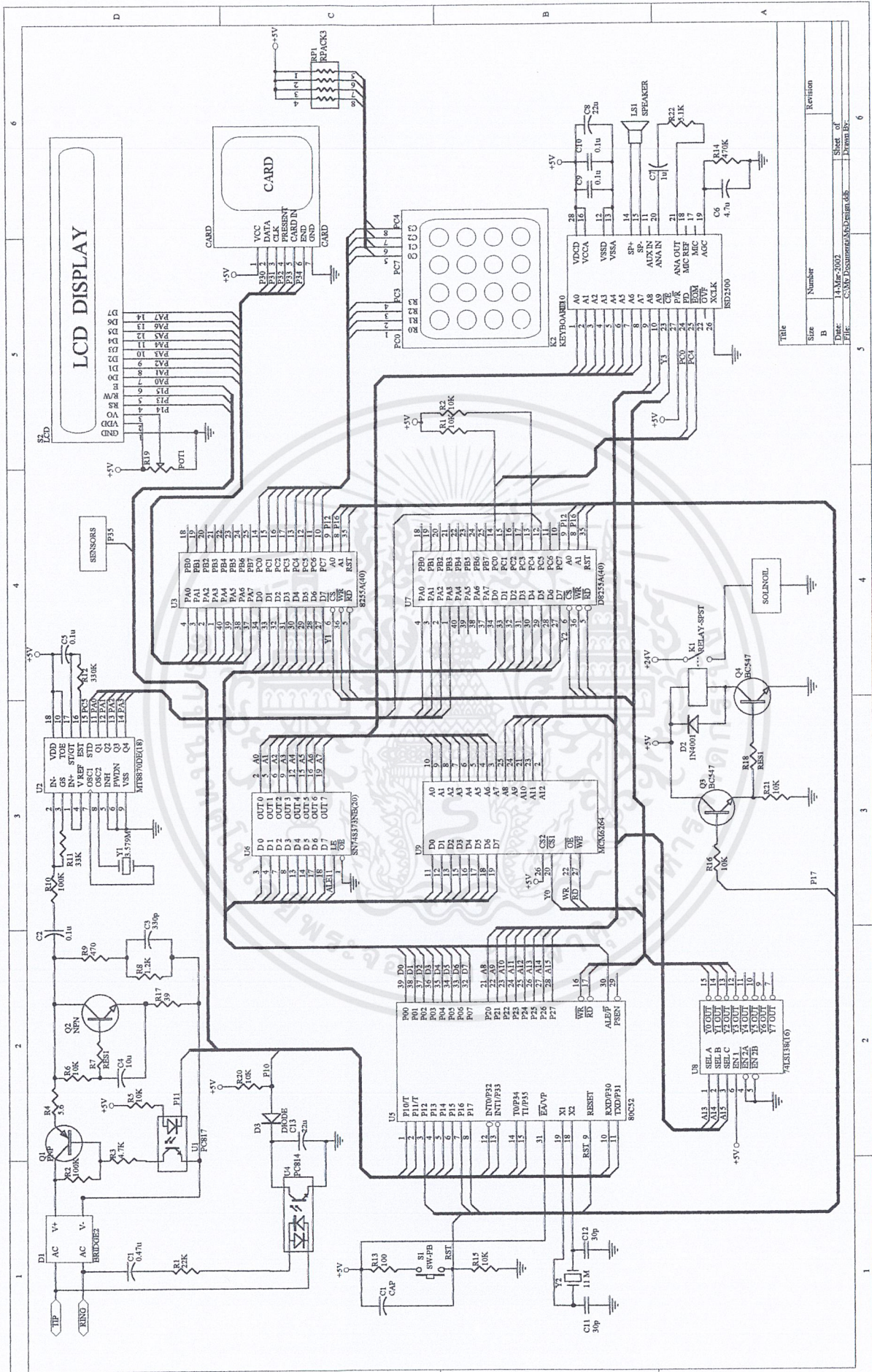
รูปแสดงวงจรภายในเครื่องขายน้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดงด้านข้างของเครื่องขายน้

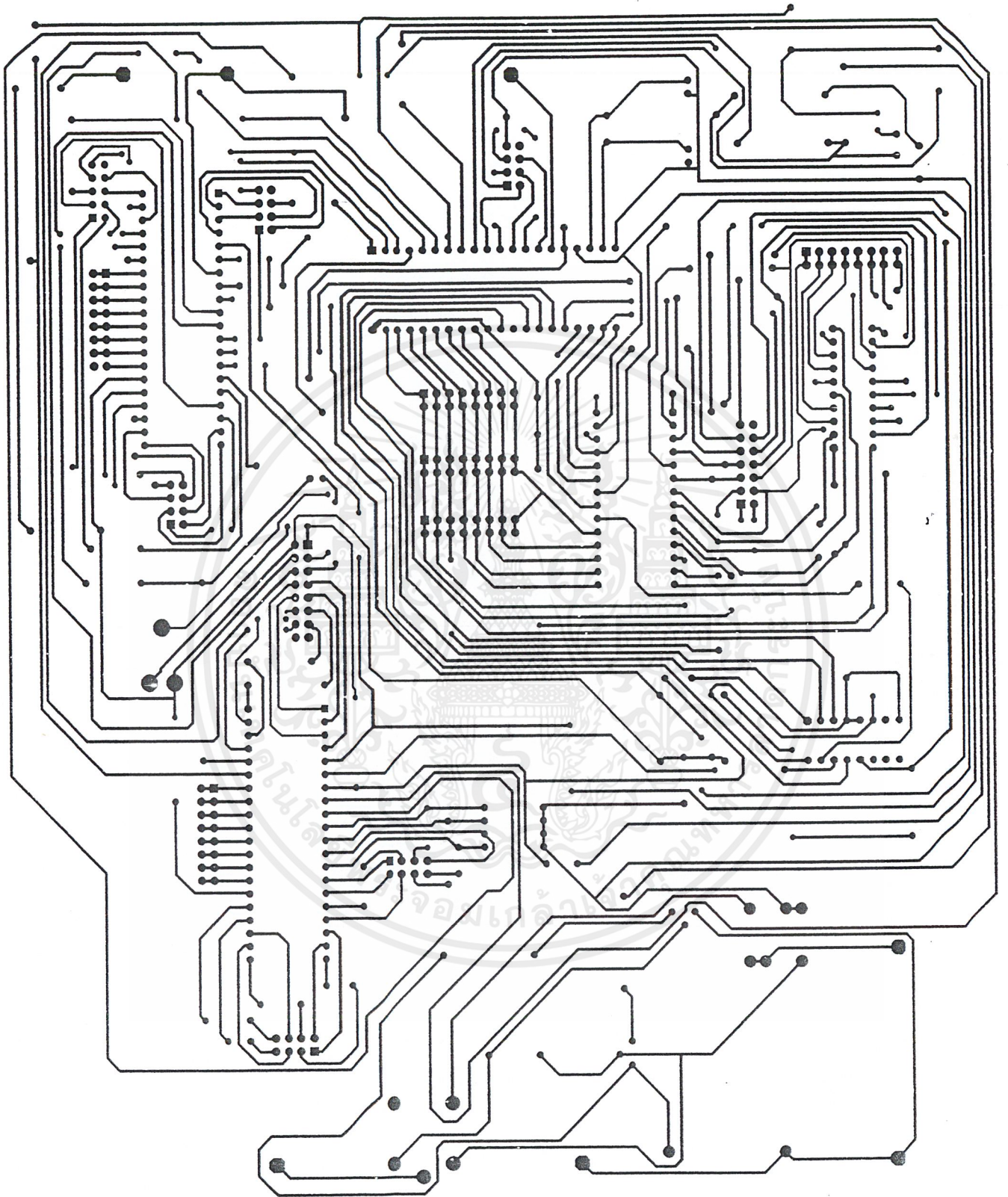
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Size	Number	Revision
B		
Date:	14-Mar-2002	Sheet of
File:	C:\306\Documents\306Design\68	Param.Bst

**แสดงวงจรโดยรวมของโครงการ**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบุคคลิกใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



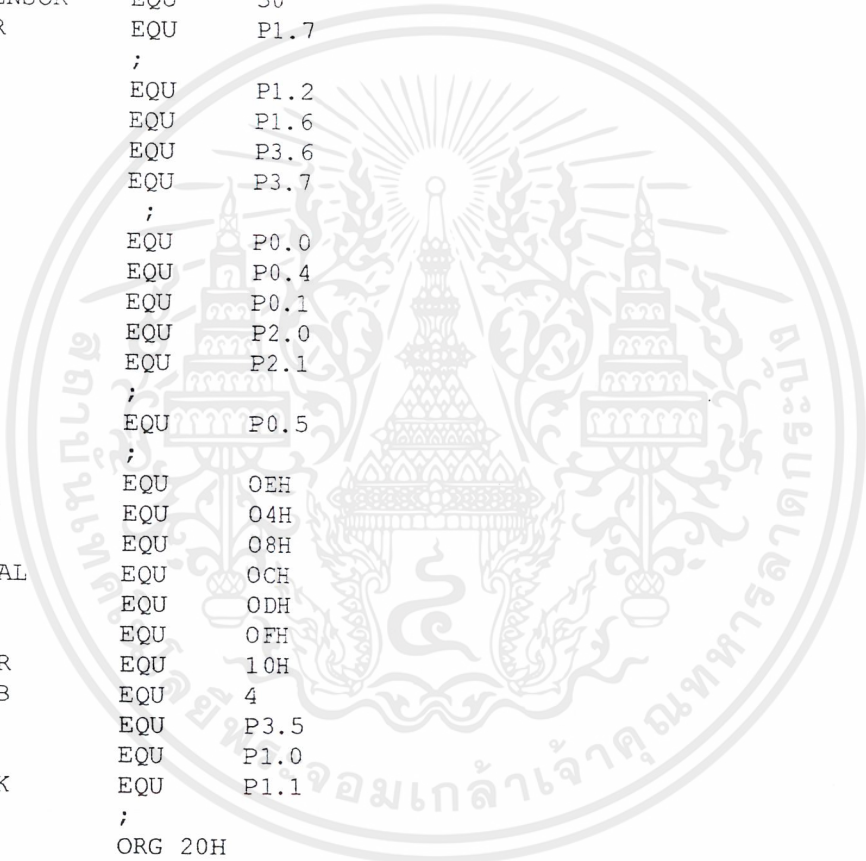
รูปแสดงลายทองแดงของโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1:          ; #####
2:          ; PROGRAM PURCHASE WATER BY MAGNETIC CARD
3:          ; #####
4:          ;
5:  EE_PROM      EQU      96H
6:  EE_PASSWORD  EQU      0001H
7:  EE_MONEY     EQU      0011
8:  EE_CODECARD  EQU      02E1H
9:  EE_TOTALCARD EQU      0006H
10: RS_LCD      EQU      P1.4
11: RW_LCD      EQU      P1.3
12: E_LCD       EQU      P1.5
13: DATA_CARD  EQU      P3.0
14: PRESENT     EQU      P3.2
15: CLK_CARD    EQU      P3.1
16: CARD_IN     EQU      P3.3
17: END_STOP    EQU      P3.4
18: ROUND_SENSOR EQU      30
19: ON_WATER    EQU      P1.7
20:          ;
21: A0_8255     EQU      P1.2
22: A1_8255     EQU      P1.6
23: WR_SIG      EQU      P3.6
24: RD_SIG      EQU      P3.7
25:          ;
26: ISD_PD      EQU      P0.0
27: ISD_EOM     EQU      P0.4
28: ISD_CE      EQU      P0.1
29: A8_ISD      EQU      P2.0
30: A9_ISD      EQU      P2.1
31:          ;
32: MT_STD      EQU      P0.5
33:          ;
34: KEY_ZERO    EQU      0EH
35: KEY_BATH    EQU      04H
36: KEY_LITE    EQU      08H
37: KEY_MANUAL  EQU      0CH
38: KEY_SET     EQU      0DH
39: KEY_CLR     EQU      0FH
40: KEY_ENTER   EQU      10H
41: TOTAL_SUB   EQU      4
42: SENSOR      EQU      P3.5
43: CHK_RING    EQU      P1.0
44: LIFT_HOOK   EQU      P1.1
45:          ;
46:          ORG 20H
47: CONTROL_BIT: DS 1
48: CHECKSUM:   DS 1
49: KEEP_DATA:  DS 25
50: KEEP_KEY:   DS 1
51: DATA_DISP: DS 16
52: KEEP_MONEY: DS 5
53: POSITION_SUB: DS 1
54: RAM_BATH:   DS 4
55: RAM_LITE:   DS 4
56: KEEP_BUY:   DS 1
57: KEEP_HAVE:  DS 1
58: KEEP_FILL:  DS 3
59: KEEP_PWD:   DS 4
60: RAM_PWD:    DS 4
61: KEEP_LITE:  DS 4
62: RAM_SPARE:  DS 5
63: KEEP_CODECARD: DS 3
64: RAM_NUMCARD: DS 1

```



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

65:  KEEP_DTMF:      DS 5
66:  RAM_CODECARD:   DS 3
67:                ;
68:  START           EQU    CONTROL_BIT.0
69:  STOP            EQU    CONTROL_BIT.1
70:  ERROR           EQU    CONTROL_BIT.2
71:                ;
72:                ;
73:                ;
74:                ORG 0000H
75:  DE_POW:        MOV R7,#2
76:  LOOP3:         MOV R0,#0FFH
77:  LOOP2:         MOV R1,#0FFH
78:  LOOP1:         NOP
79:                DJNZ R1,LOOP1
80:                DJNZ R0,LOOP2
81:                DJNZ R7,LOOP3
82:                ;
83:  ;===== INITIAL EQUIPMENT INTERFACT =====
84:                MOV PSW,#00H
85:                CLR ON_WATER
86:                LCALL INIT_8255
87:                LCALL INIT_LCD
88:                LCALL INIT_ISD
89:                MOV DPTR,#EE_PASSWORD
90:                LCALL READ_EE
91:                CJNE A,#0FFH,START_1
92:                MOV R0,#KEEP_PWD
93:                LCALL SET_PASSWORD
94:                ;
95:  TR_PWD1:       MOV R0,#KEEP_PWD
96:                MOV R5,#4
97:                MOV DPTR,#EE_PASSWORD
98:  TR_PWD2:       MOV A,@R0
99:                LCALL WRITE_EE
100:               INC R0
101:               INC DPTR
102:               DJNZ R5,TR_PWD2
103:               SJMP START_3
104:               ;
105:  START_1:       MOV R5,#4
106:               MOV R0,#KEEP_PWD
107:               MOV DPTR,#EE_PASSWORD
108:  START_2:       CLR C
109:               CLR A
110:               LCALL READ_EE
111:               MOV @R0,A
112:               INC R0
113:               INC DPTR
114:               DJNZ R5,START_2
115:  START_3:       CLR A
116:               ;
117:  BEGIN:         CLR A
118:               MOV DPTR,#POWER1
119:               LCALL SHOW_ROW
120:               MOV A,#001H
121:               LCALL WRLINE
122:               MOV BPTR,#POWER2
123:               LCALL SHOW_ROW
124:               MOV A,#002H
125:               LCALL WRLINE
126:               MOV R5,#02H
127:               LCALL DELAY_SHOW
128:

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ทำงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

129: CALL_LOOP: MOV R0,#00H
130:          SETB C
131:          SETB LIFT_HOOK
132:          MOV C,CHK_RING
133:          JC CHK_CARDIN
134: IN1:      MOV R2,#0FFH
135:          DJNZ R2,$
136:          MOV C,CHK_RING
137:          JC IN2
138:          SJMP IN1
139:          ;
140: IN2:      INC R0
141:          MOV A,R0
142:          CJNE A,#03H,IN3
143:          SJMP LINK
144:          ;
145: IN3:      LCALL DET_RING
146:          SJMP IN1
147: LINK:     CLR LIFT_HOOK
148:          LCALL TEL_SERVICE
149:          ;
150: CHK_CARDIN:SETB CARD_IN
151:          JB CARD_IN,CALL_LOOP
152:          MOV CONTROL_BIT,#00H
153:          SETB DATA_CARD
154:          SETB PRESENT
155:          SETB CLK_CARD
156:          LCALL GET_DATA
157:          ;
158: SH_DATA:  MOV A,KEEP_DATA+1
159:          CJNE A,#0FH,START_FUNC
160:          LCALL DATA_ERROR
161:          LJMP WAIT_OUT
162: START_FUNC:MOV DPTR,#MYNUM_1
163:          LCALL SHOW_ROW
164:          MOV A,#01H
165:          LCALL WRLINE
166:          MOV DPTR,#MYNUM_2
167:          LCALL SHOW_ROW
168:          MOV A,#02H
169:          LCALL WRLINE
170:          CLR ON_WATER
171:          MOV A,#10
172:          LCALL GOTO
173:          MOV R0,#KEEP_DATA+1
174: DSP_NUM1: MOV A,@R0
175:          CJNE A,#0FH,DSP_NUM2
176:          MOV A,#'.'
177:          LCALL WRBYTE
178:          SJMP COMPARE
179:          ;
180: DSP_NUM2: ADD A,#30H
181:          LCALL WRBYTE
182:          INC R0
183:          AJMP DSP_NUM1
184:          ;
185:          ; ;===== COMPARE CARD 5 DIGIT =====
186: COMPARE:  MOV R4,#00H
187:          MOV R2,#TOTAL_SUB
188:          MOV A,KEEP_DATA+1
189:          SWAP A
190:          MOV R0,#KEEP_DATA+2
191:          ADD A,@R0
192:          MOV R7,A

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

193:      MOV A,KEEP_DATA+3
194:      SWAP A
195:      MOV R0,#KEEP_DATA+4
196:      ADD A,@R0
197:      MOV R6,A
198:      MOV A,KEEP_DATA+5
199:      MOV R5,A
200:      MOV DPTR,#SUB1
201:  AGAIN1:  LCALL CHK_NUMCARD
202:      CJNE R3,#11H,HAVE_NUM
203:      INC R4
204:      DJNZ R2,AGAIN1
205:      ;
206:      LCALL CHK_NUMCARD1
207:      CJNE R3,#11H,HAVE_COE
208:      ;
209:  NO_REGI:  MOV DPTR,#REGIS_2
210:      LCALL SHOW_ROW
211:      MOV A,#02H
212:      LCALL WRLINE
213:      LJMPL WAIT_OUT
214:      ;
215:  HAVE_NUM:  INC R4
216:  HAVE_COE:  MOV POSITION_SUB,R4
217:      MOV DPTR,#EE_MONEY-5
218:  HAVE_NUM1:  INC DPTR
219:      INC DPTR
220:      INC DPTR
221:      INC DPTR
222:      INC DPTR
223:  HAVE_NUM2:  DJNZ R4,HAVE_NUM1
224:
225:
226:  ; ;===== SHOW MONEY TO LCD DISPLAY =====
227:  DISP_MONEY:  MOV A,#45H
228:      LCALL GOTO
229:      MOV R5,#03H
230:      MOV R0,#KEEP_MONEY
231:  DIS_MONEY4:  LCALL READ_EE
232:      MOV @R0,A
233:      CJNE A,#0FFH,DIS_MONEY2
234:      MOV A,#' '
235:      SJMP DIS_MONEY3
236:  DIS_MONEY2:  ADD A,#30H
237:  DIS_MONEY3:  LCALL WRBYTE
238:      INC R0
239:      INC DPTR
240:      DJNZ R5,DIS_MONEY4
241:      MOV A,#49H
242:      LCALL GOTO
243:      MOV R5,#02H
244:  DIS_STANG1:  LCALL READ_EE
245:      MOV @R0,A
246:      CJNE A,#0FFH,DIS_STANG2
247:      MOV A,#'0'
248:      SJMP DIS_STANG3
249:  DIS_STANG2:  ADD A,#30H
250:  DIS_STANG3:  LCALL WRBYTE
251:      INC R0
252:      INC DPTR
253:      DJNZ R5,DIS_STANG1
254:      MOV A,#40H+16
255:      LCALL GOTO
256:      MOV A,KEEP_MONEY

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

257:          ANL A, #0FFH
258:          ANL A, KEEP_MONEY+1
259:          ANL A, KEEP_MONEY+2
260:          CJNE A, #0FFH, SELECT_FUN
261:
262:          MOV DPTR, #NO_MONEY2
263:          LCALL SHOW_ROW
264:          MOV A, #02H
265:          LCALL WRLINE
266:
267: SELECT_FUN: LCALL KEY_IN
268:          CJNE A, #KEY_SET, FUNC_S
269:          LJMP SETUP
270: FUNC_S:    CJNE A, #KEY_CLR, FUNC1
271:          LJMP REMOVE
272: FUNC1:    CJNE A, #KEY_BATH, FUNC2
273:          SJMP BUY_BATH
274: FUNC2:    CJNE A, #KEY_LITE, FUNC3
275:          LJMP BUY_LITE
276: FUNC3:    CJNE A, #KEY_MANUAL, FUNC4
277:          LJMP BUY_MANUAL
278: FUNC4:    SJMP SELECT_FUN
279:          ;
280: REMOVE:   MOV DPTR, #REMOV_1
281:          LCALL SHOW_ROW
282:          MOV A, #01H
283:          LCALL WRLINE
284:          MOV DPTR, #REMOV_2
285:          LCALL SHOW_ROW
286:          MOV A, #02H
287:          LCALL WRLINE
288:          LJMP WAIT_OUT
289:          ;
290:          ;===== PROGRAM BUY WATER WITH BATH =====
291: BUY_BATH:  MOV DPTR, #BATH_1
292:          LCALL SHOW_ROW
293:          MOV A, #01H
294:          LCALL WRLINE
295: START_BATH: MOV DPTR, #BATH_2
296:          LCALL SHOW_ROW
297:          MOV A, #02H
298:          LCALL WRLINE
299:          ;
300:          MOV RAM_BATH, #0FFH
301:          MOV RAM_BATH+1, #0FFH
302:          MOV RAM_BATH+2, #0FFH
303:          MOV A, #46H
304:          LCALL GOTO
305:          MOV R5, #00H
306:          MOV R0, #RAM_BATH
307: AGAIN_BATH: LCALL KEY_IN
308:          CJNE A, #KEY_CLR, BA_1
309:          CJNE R5, #00H, AGAIN_B2
310:          LJMP START_FUNC
311: AGAIN_B2:  SJMP START_BATH
312: BA_1:     CJNE A, #KEY_BATH, BA_2
313:          SJMP AGAIN_BATH
314: BA_2:     CJNE A, #KEY_LITE, BA_3
315:          SJMP AGAIN_BATH
316: BA_3:     CJNE A, #KEY_MANUAL, BA_4
317:          SJMP AGAIN_BATH
318: BA_4:     CJNE A, #KEY_SET, BA_5
319:          LJMP SETUP
320: BA_5:     CJNE A, #KEY_ENTER, BA_6

```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

321:          CJNE R5,#00H,EXECUTE_B1
322:          SJMP AGAIN_BATH
323: EXECUTE_B1: CJNE R5,#01H,EXECUTE_B2
324:          MOV RAM_BATH+2,RAM_BATH
325:          MOV RAM_BATH+1,#0FFH
326:          MOV RAM_BATH,#0FFH
327: EXECUTE_B2: CJNE R5,#02H,EXECUTE_B3
328:          MOV RAM_BATH+2,RAM_BATH+1
329:          MOV RAM_BATH+1,RAM_BATH
330:          MOV RAM_BATH,#0FFH
331: EXECUTE_B3: LJMP COMP_BATH
332: BA_6:      CJNE A,#KEY_ZERO,BA_7
333:          CJNE R5,#00H,AGAIN_B3
334:          SJMP START_BATH
335: AGAIN_B3:  INC R5
336:          MOV A,#00H
337:          SJMP SHOW_BATH
338: BA_7:      INC R5
339:          LCALL VERT_KEY
340: SHOW_BATH: MOV @R0,A
341:          ADD A,#30H
342:          LCALL WRBYTE
343:          INC R0
344:          CJNE R5,#03H,AGAIN_BATH
345:          MOV A,#50H
346:          LCALL GOTO
347: EXECUTE_B5: LCALL KEY_IN
348:          CJNE A,#KEY_ENTER,EXECUTE_B4
349:          SJMP COMP_BATH
350: EXECUTE_B4: CJNE A,#KEY_CLR,EXECUTE_B5
351:          LJMP START_BATH
352:          ;
353: COMP_BATH: MOV R0,#RAM_BATH+2
354:          MOV R1,#EE_MONEY+2
355:          LCALL CHK_ENOUGH
356:          CJNE A,#11H,MISS_BATH
357:          MOV R0,#RAM_BATH
358:          LCALL CHK_250
359:          CJNE A,#22H,TRUE_BATH
360: MISS_BATH: LCALL NOT_MONEY
361:          LJMP START_FUNC
362: TRUE_BATH: MOV RAM_BATH+3,#00H
363:          MOV RAM_BATH+4,#00H
364:          MOV A,KEEP_MONEY
365:          CJNE A,#0FFH,ASSU_1
366:          MOV KEEP_MONEY,#0
367: ASSU_1:    MOV A,KEEP_MONEY+1
368:          CJNE A,#0FFH,ASSU_2
369:          MOV KEEP_MONEY+1,#0
370: ASSU_2:    MOV A,KEEP_MONEY+2
371:          CJNE A,#0FFH,ASSU_3
372:          MOV KEEP_MONEY+2,#0
373: ASSU_3:    MOV A,KEEP_MONEY+3
374:          CJNE A,#0FFH,ASSU_4
375:          MOV KEEP_MONEY+3,#0
376: ASSU_4:    MOV A,KEEP_MONEY+4
377:          CJNE A,#0FFH,DISP_BATH
378:          MOV KEEP_MONEY+4,#0
379: DISP_BATH: MOV RAM_SPARE,#0FFH
380:          MOV RAM_SPARE+1,#0FFH
381:          MOV RAM_SPARE+2,#0
382:          MOV RAM_SPARE+3,#0
383:          MOV RAM_SPARE+4,#0
384: UPDATE_B2: MOV DPTR,#MANUAL_B1

```

```

385:          LCALL SHOW_ROW
386:          MOV A,#01H
387:          LCALL WRLINE
388:  DISP_B_L:  MOV A,#05H
389:          LCALL GOTO
390:          MOV R5,#6
391:          MOV R0,#RAM_SPARE
392:  DISP_B_3:  CJNE R5,#3,DISP_B_4
393:          MOV A,'#.'
394:          LCALL WRBYTE
395:          DEC R5
396:  DISP_B_4:  CJNE @R0,#0FFH,DISP_B_1
397:          MOV A,'# '
398:          SJMP DISP_B_2
399:  DISP_B_1:  MOV A,@R0
400:          ADD A,#30H
401:  DISP_B_2:  LCALL WRBYTE
402:          INC R0
403:          DJNZ R5,DISP_B_3
404:          ;
405:          ;
406:          SETB ON_WATER
407:          LCALL CHK_SENSOR
408:          CJNE A,#22H,ADD2_BATH
409:          LJMPP DISP_SORRY
410:          ;
411:  ADD2_BATH: MOV A, RAM_SPARE+4
412:          CJNE A,#09H,ADD2_BATH4
413:          MOV RAM_SPARE+4,#0
414:          MOV A, RAM_SPARE+3
415:          CJNE A,#09H,ADD2_BATH3
416:          MOV RAM_SPARE+3,#0
417:          MOV A, RAM_SPARE+2
418:          CJNE A,#09H,ADD2_BATH2
419:          MOV RAM_SPARE+2,#0
420:          MOV A, RAM_SPARE+1
421:          CJNE A,#09H,ADD2_BATH1
422:          MOV RAM_SPARE+1,#0
423:          MOV A, RAM_SPARE
424:          SJMP ADD2_BATH0
425:          ;
426:  ADD2_BATH4: INC RAM_SPARE+4
427:          SJMP SATTLE_B1
428:  ADD2_BATH3: INC RAM_SPARE+3
429:          SJMP SATTLE_B1
430:  ADD2_BATH2: INC RAM_SPARE+2
431:          SJMP SATTLE_B1
432:  ADD2_BATH1: CJNE A,#0FFH,ADD2_LI1_1
433:          MOV RAM_SPARE+1,#0
434:  ADD2_LI1_1: INC RAM_SPARE+1
435:          SJMP SATTLE_B1
436:  ADD2_BATH0: CJNE A,#0FFH,ADD2_LI0_1
437:          MOV RAM_SPARE,#0
438:  ADD2_LI0_1: INC RAM_SPARE
439:          MOV A, RAM_SPARE
440:          CJNE A,#09H,SATTLE_B1
441:          MOV RAM_SPARE+1,#09
442:          MOV RAM_SPARE+2,#09
443:          MOV RAM_SPARE+3,#09
444:  SATTLE_B1: MOV A, KEEP_MONEY+4
445:          CJNE A,#0,SATTLE_B2
446:          SJMP SATTLE_B7
447:  SATTLE_B2: DEC KEEP_MONEY+4
448:  SATTLE_B6: SJMP CONTI_BATH

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ใช้สำหรับการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

449: ;
450: SATTLE_B7: MOV KEEP_MONEY+4,#09H
451:           MOV A,KEEP_MONEY+3
452:           CJNE A,#0,SATTLE_B3
453:           SJMP SATTLE_B8
454: SATTLE_B3: DEC KEEP_MONEY+3
455:           SJMP CONTI_BATH
456: SATTLE_B8: MOV KEEP_MONEY+3,#09H
457:           MOV A,KEEP_MONEY+2
458:           CJNE A,#0,SATTLE_B4
459:           SJMP SATTLE_B9
460: SATTLE_B4: DEC KEEP_MONEY+2
461:           SJMP CONTI_BATH
462: SATTLE_B9: MOV KEEP_MONEY+2,#09H
463:           MOV A,KEEP_MONEY+1
464:           CJNE A,#0,SATTLE_B5
465:           SJMP SATTLE_B10
466: SATTLE_B5: DEC KEEP_MONEY+1
467:           SJMP CONTI_BATH
468: SATTLE_B10:MOV KEEP_MONEY+1,#09H
469:           DEC KEEP_MONEY
470: CONTI_BATH:MOV R0,#RAM_BATH
471:           MOV R1,#RAM_SPARE
472:           MOV R5,#05H
473: CON_B3:   MOV A,@R1
474:           XRL A,@R0
475:           JNZ CON_B4
476:           INC R0
477:           INC R1
478:           DJNZ R5,CON_B3
479:           LJMP QUIT_M
480: CON_B4:   LJMP UPDATE_B2
481:           ;
482: ;===== PROGRAM BUY WATER WITH LITE =====
483: ;
484: BUY_LITE: MOV DPTR,#BATH_1
485:           LCALL SHOW_ROW
486:           MOV A,#01H
487:           LCALL WRLINE
488: START_LITE:MOV DPTR,#LITE_2
489:           LCALL SHOW_ROW
490:           MOV A,#02H
491:           LCALL WRLINE
492:           MOV RAM_LITE,#0FFH
493:           MOV RAM_LITE+1,#0FFH
494:           MOV RAM_LITE+2,#0FFH
495:           MOV A,#46H
496:           LCALL GOTO
497:           MOV R5,#00H
498:           MOV R0,#RAM_LITE
499: AGAIN_LITE:LCALL KEY_IN
500:           CJNE A,#KEY_CLR,LI_1
501:           CJNE R5,#00H,AGAIN_L2
502:           LJMP START_FUNC
503: AGAIN_L2: SJMP START_LITE
504: LI_1:     CJNE A,#KEY_BATH,LI_2
505:           SJMP AGAIN_LITE
506: LI_2:     CJNE A,#KEY_LITE,LI_3
507:           SJMP AGAIN_LITE
508: LI_3:     CJNE A,#KEY_MANUAL,LI_4
509:           SJMP AGAIN_LITE
510: LI_4:     CJNE A,#KEY_SET,LI_5
511:           LJMP SETUP
512: LI_5:     CJNE A,#KEY_ENTER,LI_6

```

ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

513:          CJNE R5,#00H,EXCUTE_L1
514:          SJMP AGAIN_LITE
515:  EXCUTE_L1: CJNE R5,#01H,EXCUTE_L2
516:          MOV  RAM_LITE+2,RAM_LITE
517:          MOV  RAM_LITE+1,#0FFH
518:          MOV  RAM_LITE,#0FFH
519:  EXCUTE_L2: CJNE R5,#02H,EXCUTE_L3
520:          MOV  RAM_LITE+2,RAM_LITE+1
521:          MOV  RAM_LITE+1,RAM_LITE
522:          MOV  RAM_LITE,#0FFH
523:  EXCUTE_L3: LJMP COMP_LITE
524:  LI_6:     CJNE A,#KEY_ZERO,LI_7
525:          CJNE R5,#00H,AGAIN_L3
526:          SJMP START_LITE
527:  AGAIN_L3: INC R5
528:          MOV  A,#00H
529:          SJMP SHOW_LITE
530:  LI_7:     INC R5
531:          LCALL VERT_KEY
532:  SHOW_LITE: MOV @R0,A
533:          ADD  A,#30H
534:          LCALL WRBYTE
535:          INC  R0
536:          CJNE R5,#03H,AGAIN_LITE
537:          MOV  A,#50H
538:          LCALL GOTO
539:  EXCUTE_L5: LCALL KEY_IN
540:          CJNE A,#KEY_ENTER,EXCUTE_L4
541:          SJMP COMP_LITE
542:  EXCUTE_L4: CJNE A,#KEY_CLR,EXCUTE_L5
543:          LJMP START_LITE
544:          ;
545:  COMP_LITE: MOV  A,KEEP_MONEY
546:          CJNE A,#0FFH,TRUE_LITE
547:          MOV  RAM_BATH,#0FFH
548:          MOV  RAM_BATH+1,RAM_LITE
549:          MOV  RAM_BATH+2,RAM_LITE+1
550:          MOV  R0,#RAM_BATH
551:          LCALL MERGE_HEX
552:          MOV  R2,A
553:          MOV  R0,#KEEP_MONEY
554:          LCALL MERGE_HEX
555:          CLR  C
556:          SUBB A,R2
557:          JC  MISS_LITE
558:          JZ  COMP_LITE2
559:          SJMP TRUE_LITE
560:  COMP_LITE2: MOV  A,KEEP_MONEY+3
561:          CJNE A,#0FFH,COMP_LITE3
562:          MOV  A,#0
563:  COMP_LITE3: MOV  R0,#RAM_LITE+2
564:          CLR  C
565:          SUBB A,@R0
566:          JC  MISS_LITE
567:          SJMP TRUE_LITE
568:  MISS_LITE: LJMP MISS_BATH
569:  TRUE_LITE: MOV  RAM_LITE+3,#00H
570:          MOV -A,KEEP_MONEY
571:          CJNE A,#0FFH,ASSUM_1
572:          MOV  KEEP_MONEY,#0
573:  ASSUM_1:  MOV  A,KEEP_MONEY+1
574:          CJNE A,#0FFH,ASSUM_2
575:          MOV  KEEP_MONEY+1,#0
576:  ASSUM_2:  MOV  A,KEEP_MONEY+2

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

577:          CJNE A,#0FFH,ASSUM_3
578:          MOV KEEP_MONEY+2,#0
579: ASSUM_3:  MOV A,KEEP_MONEY+3
580:          CJNE A,#0FFH,ASSUM_4
581:          MOV KEEP_MONEY+3,#0
582: ASSUM_4:  MOV A,KEEP_MONEY+4
583:          CJNE A,#0FFH,DISP_LITE
584:          MOV KEEP_MONEY+4,#0
585: DISP_LITE: MOV KEEP_LITE,#0FFH
586:          MOV KEEP_LITE+1,#0FFH
587:          MOV KEEP_LITE+2,#0
588:          MOV KEEP_LITE+3,#0
589: UPDATE_L2: MOV DPTR,#MANUAL_L1
590:          LCALL SHOW_ROW
591:          MOV A,#01H
592:          LCALL WRLINE
593: DISP_L_L:  MOV A,#6
594:          LCALL GOTO
595:          MOV R5,#5
596:          MOV R0,#KEEP_LITE
597: DISP_L_3:  CJNE R5,#2,DISP_L_4
598:          MOV A,#'.'
599:          LCALL WRBYTE
600:          DEC R5
601: DISP_L_4:  CJNE @R0,#0FFH,DISP_L_1
602:          MOV A,#' '
603:          SJMP DISP_L_2
604: DISP_L_1:  MOV A,@R0
605:          ADD A,#30H
606: DISP_L_2:  LCALL WRBYTE
607:          INC R0
608:          DJNZ R5,DISP_L_3
609: ;
610:          SETB ON_WATER
611:          LCALL CHK_SENSOR
612:          CJNE A,#22H,ADD1_LITE
613:          LJMP DISP_SORRY
614: ;
615: ADD1_LITE: MOV A,KEEP_LITE+3
616:          CJNE A,#09H,ADD1_LITE3
617:          MOV KEEP_LITE+3,#0
618:          MOV A,KEEP_LITE+2
619:          CJNE A,#09H,ADD1_LITE2
620:          MOV KEEP_LITE+2,#0
621:          MOV A,KEEP_LITE+1
622:          CJNE A,#09H,ADD1_LITE1
623:          MOV KEEP_LITE+1,#0
624:          MOV A,KEEP_LITE
625:          SJMP ADD1_LITE0
626: ;
627: ADD1_LITE3: INC KEEP_LITE+3
628:          SJMP SATTLE_1
629: ADD1_LITE2: INC KEEP_LITE+2
630:          SJMP SATTLE_1
631: ADD1_LITE1: CJNE A,#0FFH,ADD1_LI1_1
632:          MOV KEEP_LITE+1,#0
633: ADD1_LI1_1: INC KEEP_LITE+1
634:          SJMP SATTLE_1
635: ADD1_LITE0: CJNE A,#0FFH,ADD1_LI0_1
636:          MOV KEEP_LITE,#0
637: ADD1_LI0_1: INC KEEP_LITE
638:          MOV A,KEEP_LITE
639:          CJNE A,#09H,SATTLE_1
640:          MOV KEEP_LITE+1,#09

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

641:          MOV KEEP_LITE+2,#09
642:          MOV KEEP_LITE+3,#09
643:  SATTLE_1: MOV A,KEEP_MONEY+4
644:          CJNE A,#0,SATTLE_2
645:          SJMP SATTLE_7
646:  SATTLE_2: DEC KEEP_MONEY+4
647:  SATTLE_6: SJMP CONTI_LITE
648:          ;
649:  SATTLE_7: MOV KEEP_MONEY+4,#09H
650:          MOV A,KEEP_MONEY+3
651:          CJNE A,#0,SATTLE_3
652:          SJMP SATTLE_8
653:  SATTLE_3: DEC KEEP_MONEY+3
654:          SJMP CONTI_LITE
655:  SATTLE_8: MOV KEEP_MONEY+3,#09H
656:          MOV A,KEEP_MONEY+2
657:          CJNE A,#0,SATTLE_4
658:          SJMP SATTLE_9
659:  SATTLE_4: DEC KEEP_MONEY+2
660:          SJMP CONTI_LITE
661:  SATTLE_9: MOV KEEP_MONEY+2,#09H
662:          MOV A,KEEP_MONEY+1
663:          CJNE A,#0,SATTLE_5
664:          SJMP SATTLE_10
665:  SATTLE_5: DEC KEEP_MONEY+1
666:          SJMP CONTI_LITE
667:  SATTLE_10:MOV KEEP_MONEY+1,#09H
668:          DEC KEEP_MONEY
669:  CONTI_LITE:MOV R0,#RAM_LITE
670:          MOV R1,#KEEP_LITE
671:          MOV R5,#04H
672:  CON_L3:   MOV A,@R1
673:          XRL A,@R0
674:          JNZ CON_L2
675:          INC R0
676:          INC R1
677:          DJNZ R5,CON_L3
678:          LJMP QUIT_M
679:  CON_L2:   LJMP UPDATE_L2
680:          ;
681:          ;===== PROGRAM BUY WATER WITH MANUAL =====
682:          ;
683:  BUY_MANUAL:CLR ON_WATER
684:          MOV A,KEEP_MONEY
685:          CJNE A,#0FFH,ASSUME1
686:          MOV KEEP_MONEY,#0
687:  ASSUME1:  MOV A,KEEP_MONEY+1
688:          CJNE A,#0FFH,ASSUME2
689:          MOV KEEP_MONEY+1,#0
690:  ASSUME2:  MOV A,KEEP_MONEY+2
691:          CJNE A,#0FFH,ASSUME3
692:          MOV KEEP_MONEY+2,#0
693:  ASSUME3:  MOV A,KEEP_MONEY+3
694:          CJNE A,#0FFH,ASSUME4
695:          MOV KEEP_MONEY+3,#0
696:  ASSUME4:  MOV A,KEEP_MONEY+4
697:          CJNE A,#0FFH,CHK_HAV_M2
698:          MOV -KEEP_MONEY+4,#0
699:          ;
700:  CHK_HAV_M2:LCALL HAVE_MONEY
701:          CJNE A,#22H,CLR_RAM
702:          LJMP MISS_BATH
703:          ;
704:  CLR_RAM:  MOV RAM_LITE,#0FFH

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

705:      MOV RAM_LITE+1,#0FFH
706:      MOV RAM_LITE+2,#0
707:      MOV RAM_LITE+3,#0
708:      MOV RAM_BATH,#0FFH
709:      MOV RAM_BATH+1,#0
710:      MOV RAM_BATH+2,#0
711:      MOV RAM_BATH+3,#0
712:
713:      ;===== SHOW TOTAL BATH & LITE FOR BUY MANUAL =====
714:      ;
715:      SH_STATUS: MOV DPTR,#MANUAL_L1
716:              LCALL SHOW_ROW
717:              MOV A,#01H
718:              LCALL WRLINE
719:              MOV DPTR,#MANUAL_B2
720:              LCALL SHOW_ROW
721:              MOV A,#02H
722:              LCALL WRLINE
723:      ;
724:      STATUS_L:  MOV A,#6
725:              LCALL GOTO
726:              MOV R5,#5
727:              MOV R0,#RAM_LITE
728:      STATUS_3:  CJNE R5,#2,STATUS_4
729:              MOV A,#'.'
730:              LCALL WRBYTE
731:              DEC R5
732:      STATUS_4:  CJNE @R0,#0FFH,STATUS_1
733:              MOV A,#' '
734:              SJMP STATUS_2
735:      STATUS_1:  MOV A,@R0
736:              ADD A,#30H
737:      STATUS_2:  LCALL WRBYTE
738:              INC R0
739:              DJNZ R5,STATUS_3
740:      ;
741:      STATUS_B:  MOV A,#46H
742:              LCALL GOTO
743:              MOV R5,#5
744:              MOV R0,#RAM_BATH
745:      STATUS_7:  CJNE R5,#3,STATUS_8
746:              MOV A,#'.'
747:              LCALL WRBYTE
748:              DEC R5
749:      STATUS_8:  CJNE @R0,#0FFH,STATUS_5
750:              MOV A,#' '
751:              SJMP STATUS_6
752:      STATUS_5:  MOV A,@R0
753:              ADD A,#30H
754:      STATUS_6:  LCALL WRBYTE
755:              INC R0
756:              DJNZ R5,STATUS_7
757:              MOV A,#50H
758:              LCALL GOTO
759:      ;
760:      CH_STOP:   CLR ON_WATER
761:              MOV P0,#0FFH
762:              LCALL WR_PCLOW
763:              CLR P0.3
764:              LCALL WR_PCLOW
765:              LCALL DELAY
766:              LCALL RD_PCHIG
767:              JB P0.5,CH_CARDOUT
768:              LCALL DELAY

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสำนักงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

769:          JNB P0.5,$
770:          LCALL DELAY
771:  QUIT_SER: LJMP QUIT_M
772:          ;
773:  CH_CARDOUT:SETB CARD_IN
774:          NOP
775:          NOP
776:          JB CARD_IN,QUIT_SER
777:          ;
778:  CH_ENTER_M:MOV P0,#0FFH
779:          LCALL WR_PCLOW
780:          CLR P0.3
781:          LCALL WR_PCLOW
782:          LCALL DELAY
783:          LCALL RD_PCHIG
784:          JB P0.4,CH_STOP
785:          LCALL DELAY
786:          JB P0.4,$
787:  ON_VALVE_M:SETB ON_WATER
788:          LCALL CHK_SENSOR
789:          CJNE A,#22H,ADD_LITE
790:  DISP_SORRY:CLR ON_WATER
791:          MOV DPTR,#NO_WATER1
792:          LCALL SHOW_ROW
793:          MOV A,#01H
794:          LCALL WRLINE
795:          MOV DPTR,#NO_WATER2
796:          LCALL SHOW_ROW
797:          MOV A,#02H
798:          LCALL WRLINE
799:          MOV R5,#25H
800:          LCALL DELAY_SHOW
801:          LJMP QUIT_M
802:  ADD_LITE:  MOV A,RAM_LITE+3
803:          CJNE A,#09H,ADD_LITE3
804:          MOV RAM_LITE+3,#0
805:          MOV A,RAM_LITE+2
806:          CJNE A,#09H,ADD_LITE2
807:          MOV RAM_LITE+2,#0
808:          MOV A,RAM_LITE+1
809:          CJNE A,#09H,ADD_LITE1
810:          MOV RAM_LITE+1,#0
811:          MOV A,RAM_LITE
812:          SJMP ADD_LITE0
813:          ;
814:  ADD_LITE3: INC RAM_LITE+3
815:          SJMP ADD_BATH_M
816:  ADD_LITE2: INC RAM_LITE+2
817:          SJMP ADD_BATH_M
818:  ADD_LITE1: CJNE A,#0FFH,ADD_LI1_1
819:          MOV RAM_LITE+1,#0
820:  ADD_LI1_1: INC RAM_LITE+1
821:          SJMP ADD_BATH_M
822:  ADD_LITE0: CJNE A,#0FFH,ADD_LI0_1
823:          MOV RAM_LITE,#0
824:  ADD_LI0_1: INC RAM_LITE
825:          MOV A,RAM_LITE
826:          CJNE A,#09H,ADD_BATH_M
827:          MOV RAM_LITE+1,#09
828:          MOV RAM_LITE+2,#09
829:          MOV RAM_LITE+3,#09
830:  ADD_BATH_M:MOV A,RAM_BATH+3
831:  อักษรนี้เป็นเอกสารของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
832:          MOV RAM_BATH+3,#0

```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

833:      MOV A, RAM_BATH+2
834:      CJNE A, #09H, ADD_BATH2
835:      MOV RAM_BATH+2, #0
836:      MOV A, RAM_BATH+1
837:      CJNE A, #09H, ADD_BATH1
838:      MOV RAM_BATH+1, #0
839:      MOV A, RAM_BATH
840:      SJMP ADD_BATH0
841:      ;
842: ADD_BATH3: INC RAM_BATH+3
843:      SJMP CUT_MONEY1
844: ADD_BATH2: INC RAM_BATH+2
845:      SJMP CUT_MONEY1
846: ADD_BATH1: INC RAM_BATH+1
847:      SJMP CUT_MONEY1
848: ADD_BATH0: CJNE A, #0FFH, ADD_BAO_1
849:      MOV RAM_BATH, #0
850: ADD_BAO_1: INC RAM_BATH
851:      MOV A, RAM_BATH
852:      CJNE A, #09H, CUT_MONEY1
853:      MOV RAM_BATH+1, #09
854:      MOV RAM_BATH+2, #09
855:      MOV RAM_BATH+3, #09
856:      MOV R5, #22H
857:      ;
858: CUT_MONEY1: MOV A, KEEP_MONEY+4
859:      CJNE A, #0, CUT_MONEY2
860:      SJMP CUT_MONEY7
861: CUT_MONEY2: DEC KEEP_MONEY+4
862:      MOV A, KEEP_MONEY+4
863:      CJNE A, #0, CUT_MONEY6
864:      LCALL HAVE_MONEY
865:      CJNE A, #22H, CUT_MONEY6
866:      LCALL NOT_MONEY
867:      LJMP QUIT_M
868: CUT_MONEY6: LJMP CONTI_WATER
869:      ;
870: CUT_MONEY7: MOV KEEP_MONEY+4, #09H
871:      MOV A, KEEP_MONEY+3
872:      CJNE A, #0, CUT_MONEY3
873:      SJMP CUT_MONEY8
874: CUT_MONEY3: DEC KEEP_MONEY+3
875:      SJMP CONTI_WATER
876: CUT_MONEY8: MOV KEEP_MONEY+3, #09H
877:      MOV A, KEEP_MONEY+2
878:      CJNE A, #0, CUT_MONEY4
879:      SJMP CUT_MONEY9
880: CUT_MONEY4: DEC KEEP_MONEY+2
881:      SJMP CONTI_WATER
882: CUT_MONEY9: MOV KEEP_MONEY+2, #09H
883:      MOV A, KEEP_MONEY+1
884:      CJNE A, #0, CUT_MONEY5
885:      SJMP CUT_MONEY10
886: CUT_MONEY5: DEC KEEP_MONEY+1
887:      SJMP CONTI_WATER
888: CUT_MONEY10: MOV KEEP_MONEY+1, #09H
889:      DEC KEEP_MONEY
890: CONTI_WATER: LJMP SH_STATUS
891: QUIT_M:   CLR ON_WATER
892:      LCALL SH_OFF_WA
893:      MOV A, KEEP_MONEY
894:      CJNE A, #0, END_MANUAL
895:      MOV KEEP_MONEY, #0FFH
896:      MOV A, KEEP_MONEY+1

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

897:          CJNE A,#0,END_MANUAL
898:          MOV KEEP_MONEY+1,#0FFH
899: END_MANUAL:LCALL UPDATE_EE
900:          LCALL REMAIN_MONEY
901:          LJMP REMOVE
902:          ;
903:          ;
904:          ;*****
905:          ;          SUBROUTINE
906:          ;*****
907: ;***** CONTROL WORD 8255 *****
908: INIT_8255: CLR A
909: NUM1_8255: MOV P0,#10001000B
910:          SETB A0_8255
911:          SETB A1_8255
912:          CLR WR_SIG
913:          SETB RD_SIG
914:          MOV P2,#03FH
915:          LCALL DE_8255
916:          SETB WR_SIG
917:          MOV P2,#0FFH
918:          LCALL DE_8255
919:          ;
920: NUM2_8255: MOV P0,#10011000B
921:          SETB A0_8255
922:          SETB A1_8255
923:          CLR WR_SIG
924:          SETB RD_SIG
925:          CLR P2.7
926:          SETB P2.6
927:          CLR P2.5
928:          LCALL DE_8255
929:          SETB WR_SIG
930:          MOV P2,#0FFH
931:          LCALL DE_8255
932:          RET
933:          ;
934: DE_8255:  MOV R7,#0FFH
935: DEY_1:    DJNZ R7,$
936:          RET
937:          ;
938: WR_PORTA: CLR A0_8255
939:          CLR A1_8255
940:          CLR WR_SIG
941:          SETB RD_SIG
942:          MOV P2,#03FH
943:          LCALL DE_8255
944:          SETB WR_SIG
945:          LCALL DE_8255
946:          RET
947:          ;
948: WR_PORTB: SETB A0_8255
949:          CLR A1_8255
950:          CLR WR_SIG
951:          SETB RD_SIG
952:          CLR P2.7
953:          CLR P2.6
954:          SETB P2.5
955:          LCALL DE_8255
956:          SETB WR_SIG
957:          SETB P2.7
958:          SETB P2.6
959:          SETB P2.5
960:          RET

```

เอกสารนี้เป็นเอกสารที่ให้บริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

961:      ;
962: WR_PCLOW: CLR A0_8255
963:      SETB A1_8255
964:      CLR WR_SIG
965:      SETB RD_SIG
966:
967:      MOV P2,#03FH
968:      LCALL DE_8255
969:      SETB WR_SIG
970:      LCALL DE_8255
971:      RET
972:      ;
973: RD_PCHIG: CLR A0_8255
974:      SETB A1_8255
975:      CLR RD_SIG
976:      SETB WR_SIG
977:      CLR P2.7
978:      CLR P2.6
979:      SETB P2.5
980:      LCALL DE_8255
981:      RET
982:      ;
983: RD_PORTA2: CLR A0_8255
984:      CLR A1_8255
985:      CLR RD_SIG
986:      SETB WR_SIG
987:      CLR P2.7
988:      SETB P2.6
989:      CLR P2.5
990:      LCALL DE_8255
991:      RET
992:      ;
993: RD_PCHIG2: CLR A0_8255
994:      SETB A1_8255
995:      CLR RD_SIG
996:      SETB WR_SIG
997:      CLR P2.7
998:      SETB P2.6
999:      CLR P2.5
1000:     LCALL DE_8255
1001:     RET
1002:     ;
1003: WR_PORTB2: SETB A0_8255
1004:     CLR A1_8255
1005:     CLR WR_SIG
1006:     SETB RD_SIG
1007:     CLR P2.7
1008:     SETB P2.6
1009:     CLR P2.5
1010:     LCALL DE_8255
1011:     SETB WR_SIG
1012:     SETB P2.7
1013:     SETB P2.6
1014:     SETB P2.5
1015:     RET
1016:     ;
1017: WR_PCLOW2: CLR A0_8255
1018:     SETB A1_8255
1019:     CLR WR_SIG
1020:     SETB RD_SIG
1021:     CLR P2.7
1022:     SETB P2.6
1023:     CLR P2.5
1024:     LCALL DE_8255

```

1023: สารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า  
 1024: ไม่วากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1025:      SETB WR_SIG
1026:      SETB P2.7
1027:      SETB P2.6
1028:      SETB P2.5
1029:      RET
1030:      ;
1031: ;***** INITIAL LCD *****
1032: INIT_LCD: CLR RS_LCD
1033:          CLR RW_LCD
1034:          MOV P0,#038H
1035:          LCALL WR_PORTA
1036:          LCALL EPLUSE
1037:          MOV R7,#0FFH
1038:          DJNZ R7,$
1039:          MOV P0,#00FH
1040:          LCALL WR_PORTA
1041:          LCALL EPLUSE
1042:          MOV R7,#0FFH
1043:          DJNZ R7,$
1044:          MOV P0,#06H
1045:          LCALL WR_PORTA
1046:          LCALL EPLUSE
1047:          MOV R7,#0FFH
1048:          DJNZ R7,$
1049:          MOV P0,#01H
1050:          LCALL WR_PORTA
1051:          LCALL EPLUSE
1052:          MOV R7,#0FFH
1053:          DJNZ R7,$
1054:          RET
1055:      ;
1056: INIT_ISD: SETB ISD_CE
1057:          LCALL WR_PORTB2
1058:          SETB ISD_PD
1059:          LCALL WR_PCLOW2
1060:          CLR A8_ISD
1061:          CLR A9_ISD
1062:          RET
1063:      ;
1064: ;;***** READ DATA IN EEPROM AT89S8255 *****
1065: READ_EE: MOV EE_PROM,#00001000B
1066:          MOVX A,@DPTR
1067:          MOV EE_PROM,#00000000B
1068:          RET
1069:      ;
1070: ;;***** WRITE DATA TO EEPROM IN AT89S8255 *****
1071: WRITE_EE: MOV EE_PROM,#00011000B
1072:          MOVX @DPTR,A
1073:          LCALL DTSED
1074:          MOV EE_PROM,#00000000B
1075:          RET
1076:      ;
1077: DTSED:  MOV R2,#6
1078: DTSED1: MOV R3,#0
1079:          DJNZ R3,$
1080:          DJNZ R2,DTSED1
1081:          RET
1082:      ;
1083: ;***** SET PASSWORD 4 DIGIT *****
1084: ;** OUTPUT=@R0 **
1085: SET_PASSWORD: MOV DPTR,#PASSWORD_1
1086:          LCALL SHOW_ROW
1087:          MOV A,#1 ;เป็นการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
1088:          LCALL WRLINE

```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1089:      MOV DPTR,#PASSWORD_2
1090:      LCALL SHOW_ROW
1091:      MOV A,#2
1092:      LCALL WRLINE
1093:      MOV R5,#20H
1094:      LCALL DELAY_SHOW
1095: START_PWD:  MOV DPTR,#PASSWORD_3
1096:      LCALL SHOW_ROW
1097:      MOV A,#1
1098:      LCALL WRLINE
1099: START_PWD1: MOV DPTR,#PASSWORD_4
1100:      LCALL SHOW_ROW
1101:      MOV A,#2
1102:      LCALL WRLINE
1103:      MOV A,#4BH
1104:      LCALL GOTO
1105:      MOV R5,#00H
1106: START_PWD2: LCALL KEY_IN
1107:      ;
1108:      CJNE A,#KEY_BATH,PWD_1
1109:      SJMP START_PWD2
1110: PWD_1:  CJNE A,#KEY_LITE,PWD_2
1111:      SJMP START_PWD2
1112: PWD_2:  CJNE A,#KEY_MANUAL,PWD_3
1113:      SJMP START_PWD2
1114: PWD_3:  CJNE A,#KEY_SET,PWD_4
1115:      SJMP START_PWD2
1116: PWD_4:  CJNE A,#KEY_CLR,PWD_5
1117:      CJNE R5,#0,CLR_PWD1
1118:      SJMP START_PWD2
1119: CLR_PWD1: DEC R0
1120:      DJNZ R5,CLR_PWD1
1121:      SJMP START_PWD1
1122: PWD_5:  CJNE A,#KEY_ENTER,PWD_6
1123:      SJMP START_PWD2
1124: PWD_6:  CJNE A,#KEY_ZERO,PWD_7
1125:      MOV A,#00H
1126:      SJMP PWD_8
1127: PWD_7:  LCALL VERT_KEY
1128: PWD_8:  MOV @R0,A
1129:      ADD A,#30H
1130:      LCALL WRBYTE
1131:      INC R0
1132:      INC R5
1133:      CJNE R5,#4,START_PWD2
1134: END_PWD1: MOV A,#50H
1135:      LCALL GOTO
1136:      LCALL KEY_IN
1137:      CJNE A,#KEY_CLR,END_PWD2
1138:      DEC R0
1139:      DEC R0
1140:      DEC R0
1141:      DEC R0
1142:      SJMP START_PWD1
1143: END_PWD2: CJNE A,#KEY_ENTER,END_PWD1
1144:      RET
1145:      ;
1146: ;***** READ DATA FOR MAGNETIC CARD *****
1147: GET_DATA: LCALL CLR_BUFF
1148:      MOV CHECKSUM,#00H
1149:      MOV R1,#KEEP_DATA-1
1150:      CLR START
1151: ;การนี้เป็นเอกสารที่สแกนไว้เพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
1152:      CLR ERROR

```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1153:      NOP
1154:      NOP
1155:      JNB CARD_IN, WAIT_CARD1
1156:      LJMP CALL_LOOP
1157: WAIT_CARD1: MOV R4, #02H
1158: WAIT_CARD2: MOV R7, #03H
1159:      MOV R6, #0FFH
1160:      MOV R5, #0FFH
1161: WAIT_STR:  SETB PRESENT
1162:      NOP
1163:      DJNZ R5, WAIT_STR2
1164:      SJMP WAIT_STR3
1165: WAIT_STR2: JB PRESENT, WAIT_STR
1166:      SJMP GET_DATA0
1167: WAIT_STR3: MOV R5, #0FFH
1168:      DJNZ R6, WAIT_STR
1169:      MOV R5, #0FFH
1170:      MOV R6, #0FFH
1171:      DJNZ R7, WAIT_STR
1172:      DJNZ R4, WAIT_CARD2
1173:      LCALL MISS_INS
1174: WAIT_OUT: SETB CARD_IN
1175:      CLR ON_WATER
1176:      NOP
1177:      NOP
1178:      JNB CARD_IN, WAIT_OUT
1179:      LJMP BEGIN
1180:      ;
1181: GET_DATA0: LCALL GET_CLK
1182:      MOV C, DATA_CARD
1183:      CPL C
1184:      JB DATA_CARD, GET_DATA0
1185:      SETB START
1186:      MOV R3, #25
1187: GET_DATA1: MOV R2, #05H
1188:      CLR A
1189:      INC R1
1190: GET_DATA2: JB START, GET_DATA3
1191:      LCALL GET_CLK
1192:      MOV C, DATA_CARD
1193:      CPL C
1194: GET_DATA3: RRC A
1195:      CLR START
1196:      DJNZ R2, GET_DATA2
1197:      RR A
1198:      RR A
1199:      RR A
1200: GET_DATA4: ANL A, #0FH
1201:      JB STOP, GET_DATA5
1202:      PUSH ACC
1203:      XRL A, CHECKSUM
1204:      MOV CHECKSUM, A
1205:      POP ACC
1206: GET_DATA5: MOV @R1, A
1207:      JB STOP, GET_DATA7
1208:      CJNE A, #0FH, GET_DATA6
1209:      SETB STOP
1210:      SJMP GET_DATA1
1211: GET_DATA6: DJNZ R3, GET_DATA1
1212:      INC R1
1213:      MOV @R1, #0FH
1214:      SETB STOP
1215:      SJMP GET_DATA1
1216: GET_DATA7: CJNE A, CHECKSUM, CHKSUM_ERR

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ควรแก้ไข ทงสิน อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1217: CLR ERROR
1218: DATA_TRUE: RET
1219: ;
1220: CHKSUM_ERR: SETB ERROR
1221: LCALL DATA_ERROR
1222: LJMP WAIT_OUT
1223: ;
1224: ;***** TELEPHONE SERVICE FOR REGISTER_CARD,FILL_MONEY,SET_PASSWORD *****
1225: DET_RING: MOV R1,#015H
1226: DET1: MOV R2,#0FFH
1227: DET2: MOV R3,#0FFH
1228: DET3: MOV C,CHK_RING
1229: JNC HAV_RING
1230: DJNZ R3,DET3
1231: DJNZ R2,DET2
1232: DJNZ R1,DET1
1233: LJMP CHK_CARDIN
1234: HAV_RING: RET
1235: ;
1236: TEL_SERVICE: MOV DPTR,#STANBY1
1237: LCALL SHOW_ROW
1238: MOV A,#01H
1239: LCALL WRLINE
1240: MOV DPTR,#STANBY2
1241: LCALL SHOW_ROW
1242: MOV A,#02H
1243: LCALL WRLINE
1244: ;
1245: ;
1246: COU_VOI1: MOV R2,#02H
1247: VOICE1_GRI: MOV DPTR,#VOICE_1
1248: CLR A8_ISD
1249: CLR A9_ISD
1250: LCALL ISD_2500
1251: LCALL DTMF_1DIG
1252: MOV A,KEEP_DTMF
1253: ;
1254: CJNE A,#0BH,NO_DTMF
1255: SJMP COU_VOI2
1256: NO_DTMF: DJNZ R2,VOICE1_GRI
1257: SETB LIFT_HOOK
1258: LJMP BEGIN
1259: ;
1260: COU_VOI2: MOV R2,#02H
1261: VOICE2_GRI: MOV DPTR,#VOICE_2
1262: CLR A8_ISD
1263: CLR A9_ISD
1264: LCALL ISD_2500
1265: LCALL DTMF_1DIG
1266: ;
1267: MOV A,KEEP_DTMF
1268: DTMF_3: CJNE A,#03H,DTMF_2
1269: LJMP VOICE11_GRI
1270: ;
1271: DTMF_2: CJNE A,#02H,DTMF_1
1272: LJMP VOICE8_GRI
1273: ;
1274: DTMF_1: CJNE A,#01H,NOT_DTMF
1275: SJMP COU_VOI3
1276: NOT_DTMF: DJNZ R2,VOICE2_GRI
1277: SETB LIFT_HOOK
1278: LJMP BEGIN
1279: ;
1280: ; ===== TEL. FOR MONEY =====

```

สารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1281: COU_VOI3:      MOV R2,#00H
1282: VOICE3_GRI:     MOV DPTR,#VOICE_3
1283:                 CLR A8_ISD
1284:                 CLR A9_ISD
1285:                 LCALL ISD_2500
1286:                 LCALL DTMF_4DIG
1287:                 MOV R0,#KEEP_DTMF
1288:                 LCALL VERT_DTMF
1289:                 MOV R0,#KEEP_DTMF+1
1290:                 LCALL VERT_DTMF
1291:                 MOV R0,#KEEP_DTMF+2
1292:                 LCALL VERT_DTMF
1293:                 MOV R0,#KEEP_DTMF+3
1294:                 LCALL VERT_DTMF
1295:                 LCALL COMP_DTMF
1296:                 CJNE R3,#22H,PASS_DTMF
1297:                 MOV DPTR,#VOICE_15
1298:                 SETB A8_ISD
1299:                 CLR A9_ISD
1300:                 LCALL ISD_2500
1301:                 SETB LIFT_HOOK
1302:                 LJMP BEGIN
1303:                 ;
1304: PASS_DTMF:      MOV DPTR,#VOICE_4
1305:                 CLR A8_ISD
1306:                 CLR A9_ISD
1307:                 LCALL ISD_2500
1308:                 LCALL DTMF_5DIG
1309:                 ;
1310:                 MOV R2,KEEP_DTMF
1311:                 CJNE R2,#0AH,DTM_1
1312:                 MOV KEEP_DTMF,#00H
1313:                 ;
1314: DTM_1:          MOV R2,KEEP_DTMF+1
1315:                 CJNE R2,#0AH,DTM_2
1316:                 MOV KEEP_DTMF+1,#00H
1317:                 ;
1318: DTM_2:          MOV R2,KEEP_DTMF+2
1319:                 CJNE R2,#0AH,DTM_3
1320:                 MOV KEEP_DTMF+2,#00H
1321:                 ;
1322: DTM_3:          MOV R2,KEEP_DTMF+3
1323:                 CJNE R2,#0AH,DTM_4
1324:                 MOV KEEP_DTMF+3,#00H
1325:                 ;
1326: DTM_4:          MOV R2,KEEP_DTMF+4
1327:                 CJNE R2,#0AH,DTM_5
1328:                 MOV KEEP_DTMF+4,#00H
1329:                 ;
1330: DTM_5:          MOV KEEP_DATA+1,KEEP_DTMF
1331:                 MOV KEEP_DATA+2,KEEP_DTMF+1
1332:                 MOV KEEP_DATA+3,KEEP_DTMF+2
1333:                 MOV KEEP_DATA+4,KEEP_DTMF+3
1334:                 MOV KEEP_DATA+5,KEEP_DTMF+4
1335:                 ;
1336:                 ;
1337: COMPARET:      MOV R4,#00H
1338:                 MOV R2,#TOTAL_SUB
1339:                 ;
1340:                 MOV A,KEEP_DATA+1
1341:                 SWAP A
1342:                 MOV R0,#KEEP_DATA+2
1343:                 ADDI A,@R0
1344:                 MOV R7,A

```

นี่เป็นเอกสารที่สงวนลิขสิทธิ์ของสำนักงานส่งเสริมการค้าในต่างประเทศ ณ นครเชียงใหม่ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ณาการณใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1345:      ;
1346:      MOV A,KEEP_DATA+3
1347:      SWAP A
1348:      MOV R0,#KEEP_DATA+4
1349:      ADD A,@R0
1350:      MOV R6,A
1351:      ;
1352:      MOV A,KEEP_DATA+5
1353:      MOV R5,A
1354:      ;
1355:
1356:      MOV DPTR,#SUB1
1357: AGAIN_T1:  LCALL CHK_NUMCARD
1358:      CJNE R3,#11H,HAVE_NUMT
1359:      INC R4
1360:      DJNZ R2,AGAIN_T1
1361:      ;
1362:      LCALL CHK_NUMCARD1
1363:      CJNE R3,#11H,HAVE_NUMT3
1364:      ;
1365: DIS_CONNECT2: MOV DPTR,#VOICE_7
1366:      CLR A8_ISD
1367:      CLR A9_ISD
1368:      LCALL ISD_2500
1369:      SETB LIFT_HOOK
1370:      LJMP BEGIN
1371:      ;
1372: HAVE_NUMT:  INC R4
1373: HAVE_NUMT3: MOV POSITION_SUB,R4
1374:      MOV DPTR,#EE_MONEY-5
1375: HAVE_NUMT1: INC DPTR
1376:      INC DPTR
1377:      INC DPTR
1378:      INC DPTR
1379:      INC DPTR
1380: HAVE_NUMT2: DJNZ R4,HAVE_NUMT1
1381:      ;
1382:      MOV R5,#05H
1383: RD_MONEYT:  MOV R0,#KEEP_MONEY
1384: RD_MONEYT1: LCALL READ_EE
1385:      MOV @R0,A
1386:      INC R0
1387:      INC DPTR
1388:      DJNZ R5,RD_MONEYT1
1389:      MOV A,KEEP_MONEY
1390:      ANL A,#0FFH
1391:      ANL A,KEEP_MONEY+1
1392:      ANL A,KEEP_MONEY+2
1393:      CJNE A,#0FFH,VOICES5_GRI
1394:      MOV DPTR,#VOICE_16
1395:      LCALL ISD_2500
1396:      ;
1397: VOICES5_GRI: MOV DPTR,#VOICE_5
1398:      CLR A8_ISD
1399:      CLR A9_ISD
1400:      LCALL ISD_2500
1401:      MOV KEEP_FILL,#0FFH
1402:      MOV KEEP_FILL+1,#0FFH
1403:      MOV KEEP_FILL+2,#0FFH
1404: AGAIN_DTMF0: MOV R3,#00H
1405:      MOV R1,#KEEP_FILL
1406:      ;
1407: AGAIN_DTMF: LCALL DTMF_1DIG
1408: FILL_TEL1:  CJNE A,#0BH,FILL_TEL2

```

```

1409:      MOV DPTR,#VOICE_17
1410:      SETB A8_ISD
1411:      CLR A9_ISD
1412:      LCALL ISD_2500
1413:      SJMP AGAIN_DTMF0
1414:      ;
1415: FILL_TEL2: CJNE A,#0CH,FILL_TEL3
1416:      CJNE R3,#00H,EXCUTE_T1
1417:      SETB LIFT_HOOK
1418:      LJMP BEGIN
1419:      ;
1420: EXCUTE_T1: CJNE R3,#01H,EXCUTE_T2
1421:      MOV KEEP_FILL+2,KEEP_FILL
1422:      MOV KEEP_FILL+1,#0FFH
1423:      MOV KEEP_FILL,#0FFH
1424:      ;
1425: EXCUTE_T2: CJNE R3,#02H,EXCUTE_T3
1426:      MOV KEEP_FILL+2,KEEP_FILL+1
1427:      MOV KEEP_FILL+1,KEEP_FILL
1428:      MOV KEEP_FILL,#0FFH
1429:      ;
1430: EXCUTE_T3: LJMP SUM_FILLT
1431:      ;
1432: FILL_TEL3: CJNE A,#0AH,FILL_TEL4
1433:      CJNE R3,#00H,AGAIN_DTMF1
1434:      MOV DPTR,#VOICE_18
1435:      SETB A8_ISD
1436:      CLR A9_ISD
1437:      LCALL ISD_2500
1438:      SJMP AGAIN_DTMF0
1439:      ;
1440: AGAIN_DTMF1: INC R3
1441:      MOV A,#00H
1442:      SJMP FILL_TEL6
1443:      ;
1444: FILL_TEL4: INC R3
1445:      LCALL VERT_FILL
1446: FILL_TEL6: MOV @R1,A
1447:      INC R1
1448:      CJNE R3,#03H,AGAIN_DTMF
1449:      LCALL DTMF_1DIG
1450:      CJNE A,#0CH,FILL_TEL5
1451:      SJMP SUM_FILLT
1452:      ;
1453: FILL_TEL5: MOV DPTR,#VOICE_19
1454:      SETB A8_ISD
1455:      CLR A9_ISD
1456:      LCALL ISD_2500
1457:      LCALL DTMF_1DIG
1458:      CJNE A,#0CH,FILL_TEL5
1459:      ;
1460: SUM_FILLT: MOV R0,#KEEP_FILL+2
1461:      MOV R1,#KEEP_MONEY+2
1462:      MOV R5,KEEP_MONEY
1463:      MOV R6,KEEP_MONEY+1
1464:      MOV R7,KEEP_MONEY+2
1465:      LCALL ADD_BATH
1466:      CJNE A,#11H,ERR_FILLT
1467:      MOV R0,#KEEP_MONEY
1468:      LCALL CHK_250
1469:      CJNE A,#22H,TRU_FILLT
1470:      MOV KEEP_MONEY,R5
1471:      MOV KEEP_MONEY+1,R6
1472:      MOV KEEP_MONEY+2,R7

```

นี่เป็นเอกสารที่สงวนลิขสิทธิ์โดยมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1473:          LJMP ERR_FILLT
1474:          ;
1475: TRU_FILLT:  MOV R4, POSITION_SUB
1476:          MOV DPTR, #EE_MONEY-5
1477: TRUE_FT1:   INC DPTR
1478:          INC DPTR
1479:          INC DPTR
1480:          INC DPTR
1481:          INC DPTR
1482:          DJNZ R4, TRUE_FT1
1483:          ;
1484:          MOV R5, #03H
1485:          MOV R0, #KEEP_MONEY
1486: TRUE_FT2:   MOV A, @R0
1487:          LCALL WRITE_EE
1488:          INC DPTR
1489:          INC R0
1490:          DJNZ R5, TRUE_FT2
1491:          ;
1492: FILL_TELOK: MOV DPTR, #VOICE_6
1493:          CLR A8_ISD
1494:          CLR A9_ISD
1495:          LCALL ISD_2500
1496:          SETB LIFT_HOOK
1497:          LJMP BEGIN
1498:          ;
1499: ERR_FILLT:  LJMP VOICES5_GRI
1500:          ;
1501: ;===== TEL. FOR EDIT PASSWORD =====
1502: VOICES8_GRI: MOV DPTR, #VOICE_8
1503:          CLR A8_ISD
1504:          CLR A9_ISD
1505:          LCALL ISD_2500
1506:          LCALL DTMF_4DIG
1507:          MOV R0, #KEEP_DTMF
1508:          LCALL VERT_DTMF
1509:          MOV R0, #KEEP_DTMF+1
1510:          LCALL VERT_DTMF
1511:          MOV R0, #KEEP_DTMF+2
1512:          LCALL VERT_DTMF
1513:          MOV R0, #KEEP_DTMF+3
1514:          LCALL VERT_DTMF
1515:          LCALL COMP_DTMF
1516:          CJNE R3, #11H, OFF_HOOK1
1517:          MOV DPTR, #VOICE_9
1518:          CLR A8_ISD
1519:          CLR A9_ISD
1520:          LCALL ISD_2500
1521:          ;
1522: AGAIN_EDIT1: MOV R3, #00H
1523:          MOV R1, #KEEP_PWD
1524: AGAIN_EDIT:  LCALL DTMF_1DIG
1525: EDIT_TEL1:  CJNE A, #0BH, EDIT_TEL2
1526:          MOV DPTR, #VOICE_17
1527:          SETB A8_ISD
1528:          CLR A9_ISD
1529:          LCALL ISD_2500
1530:          SJMP AGAIN_EDIT1
1531:          ;
1532: EDIT_TEL2:  CJNE A, #0CH, EDIT_TEL3
1533:          MOV DPTR, #VOICE_17
1534:          SETB A8_ISD
1535:          SETB A9_ISD
1536:          LCALL ISD_2500

```

สารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1537:          SJMP AGAIN_EDIT1
1538:          ;
1539: EDIT_TEL3:  MOV R0, #KEEP_DTMF
1540:          LCALL VERT_DTMF
1541:          MOV @R1, KEEP_DTMF
1542:          INC R1
1543:          INC R3
1544:          CJNE R3, #04H, AGAIN_EDIT
1545:          ;
1546: TEL_EDITOK: MOV R1, #KEEP_PWD
1547:          MOV R5, #04H
1548:          MOV DPTR, #EE_PASSWORD
1549: EDIT_WRIEE: MOV A, @R1
1550:          LCALL WRITE_EE
1551:          INC R1
1552:          INC DPTR
1553:          DJNZ R5, EDIT_WRIEE
1554:          MOV DPTR, #VOICE_10
1555:          CLR A8_ISD
1556:          CLR A9_ISD
1557:          LCALL ISD_2500
1558:          SETB LIFT_HOOK
1559:          LJMP BEGIN
1560:          ;
1561: OFF_HOOK1: MOV DPTR, #VOICE_15
1562:          SETB A8_ISD
1563:          CLR A9_ISD
1564:          LCALL ISD_2500
1565:          SETB LIFT_HOOK
1566:          LJMP BEGIN
1567:          ;
1568: ;===== TEL. FOR REGISTER CARD =====
1569: VOICE11_GRI: MOV DPTR, #VOICE_11
1570:          CLR A8_ISD
1571:          CLR A9_ISD
1572:          LCALL ISD_2500
1573:          LCALL DTMF_4DIG
1574:          MOV R0, #KEEP_DTMF
1575:          LCALL VERT_DTMF
1576:          MOV R0, #KEEP_DTMF+1
1577:          LCALL VERT_DTMF
1578:          MOV R0, #KEEP_DTMF+2
1579:          LCALL VERT_DTMF
1580:          MOV R0, #KEEP_DTMF+3
1581:          LCALL VERT_DTMF
1582:          LCALL COMP_DTMF
1583:          ;
1584:          CJNE R3, #11H, OFF_HOOK1
1585:          MOV DPTR, #VOICE_12
1586:          SETB A8_ISD
1587:          CLR A9_ISD
1588:          LCALL ISD_2500
1589:          ;
1590: RECI_DTMF1: MOV R2, #00H
1591:          MOV R1, #RAM_SPARE
1592:          ;
1593: RECI_DTMF:  LCALL DTMF_1DIG
1594:          CJNE A, #0BH, NEX_DTMF
1595:          MOV DPTR, #VOICE_17
1596:          SETB A8_ISD
1597:          CLR A9_ISD
1598:          LCALL ISD_2500
1599:          SJMP RECI_DTMF1
1600:          ;

```

สารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่วารณใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1601: NEX_DTMF:      CJNE A, #0CH, NEX_DTMF2
1602:                MOV DPTR, #VOICE_17
1603:                SETB A8_ISD
1604:                CLR A9_ISD
1605:                LCALL ISD_2500
1606:                SJMP RECI_DTMF1
1607:                ;
1608: NEX_DTMF2:      CJNE A, #0AH, STRO_CODE
1609:                CJNE R2, #05H, STRO_CODE1
1610:                SJMP NEX_DTMF1
1611:                ;
1612: STRO_CODE1:     MOV A, #00H
1613: STRO_CODE:      MOV @R1, A
1614:                INC R1
1615:                INC R2
1616:                CJNE R2, #05H, RECI_DTMF
1617:                ;
1618: NEX_DTMF1:      MOV DPTR, #VOICE_19
1619:                SETB A8_ISD
1620:                CLR A9_ISD
1621:                LCALL ISD_2500
1622:                LCALL DTMF_1DIG
1623:                CJNE A, #0CH, NEX_DTMF1
1624:                ;
1625:                CLR A
1626:                MOV A, RAM_SPARE
1627:                SWAP A
1628:                ADD A, RAM_SPARE+1
1629:                MOV KEEP_CODECARD, A
1630:                ;
1631:                CLR A
1632:                MOV A, RAM_SPARE+2
1633:                SWAP A
1634:                ADD A, RAM_SPARE+3
1635:                MOV KEEP_CODECARD+1, A
1636:                ;
1637:                CLR A
1638:                MOV A, RAM_SPARE+4
1639:                MOV KEEP_CODECARD+2, A
1640:                LJMP CAN_REGISTER
1641:                ;
1642: VOICE14_GRI:    MOV DPTR, #VOICE_14
1643:                SETB A8_ISD
1644:                CLR A9_ISD
1645:                LCALL ISD_2500
1646:                SETB LIFT_HOOK
1647:                LJMP BEGIN
1648:                ;
1649: VOICE20_GRI:    MOV DPTR, #VOICE_20
1650:                SETB A8_ISD
1651:                CLR A9_ISD
1652:                LCALL ISD_2500
1653:                SETB LIFT_HOOK
1654:                LJMP BEGIN
1655:                ;
1656: OFF_HOOK:      MOV DPTR, #VOICE_15
1657:                SETB A8_ISD
1658:                CLR A9_ISD
1659:                LCALL ISD_2500
1660:                SETB LIFT_HOOK
1661:                LJMP BEGIN
1662:                ;
1663: กอ; **** กอ; CHECK WATER FLOW **** ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
1664: CHK_SENSOR:    PUSH 04

```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1665:          MOV R4,#10
1666: CHK_SEN1:  MOV R7,#15H
1667: CHK_SEN3:  MOV R6,#0FFH
1668: CHK_SEN4:  MOV R5,#0FFH
1669: CHK_SEN5:  JB SENSOR,CHK_SEN2
1670:          DJNZ R5,CHK_SEN5
1671:          DJNZ R6,CHK_SEN4
1672:          DJNZ R7,CHK_SEN3
1673:          SJMP NO_WATER
1674: CHK_SEN2:  LCALL DELAY
1675:          JB SENSOR,$
1676:          DJNZ R4,CHK_SEN1
1677:          POP 04
1678:          MOV A,#11H
1679:          RET
1680: NO_WATER:  MOV A,#22H
1681:          POP 04
1682:          RET
1683:          ;
1684: ;*****SUB CHECK HAVE MONEY *****
1685: ;
1686: HAVE_MONEY: CLR A
1687:          MOV A,KEEP_MONEY
1688:          ORL A,KEEP_MONEY+1
1689:          ORL A,KEEP_MONEY+2
1690:          ORL A,KEEP_MONEY+3
1691:          ORL A,KEEP_MONEY+4
1692:          CJNE A,#00H,NOT_HAV_M
1693:          MOV A,#22H
1694:          RET
1695: NOT_HAV_M: MOV A,#11H
1696:          RET
1697: ;
1698: ;***** SUB SHOW "YOU HAVE MONEY NOT ENOUGHT *****
1699: ;
1700: NOT_MONEY: MOV DPTR,#MISS_B1
1701:          LCALL SHOW_ROW
1702:          MOV A,#01H
1703:          LCALL WRLINE
1704:          MOV DPTR,#MISS_B2
1705:          LCALL SHOW_ROW
1706:          MOV A,#02H
1707:          LCALL WRLINE
1708:          MOV R5,#20H
1709:          LCALL DELAY_SHOW
1710:          RET
1711:          ;
1712: ;***** SUB UPDATE MONEY TO EE_PROM *****
1713: ;
1714: UPDATE_EE: MOV R4,POSITION_SUB
1715:          MOV DPTR,#EE_MONEY-5
1716: TRUE_B1:   INC DPTR
1717:          INC DPTR
1718:          INC DPTR
1719:          INC DPTR
1720:          INC DPTR
1721:          DJNZ R4,TRUE_B1
1722:          MOV R5,#05H
1723:          MOV R0,#KEEP_MONEY
1724: TRUE_B2:   MOV A,@R0
1725:          LCALL WRITE_EE
1726:          INC DPTR
1727:          INC R0
1728:          DJNZ R5,TRUE_B2

```

สารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1729:          RET
1730:          ;
1731:          ;
1732: ;;;;% % CHECK NUMBER CARD % % % % % % % %
1733: CHK_NUMCARD: CLR C
1734:          CLR A
1735:          MOVC A,@A+DPTR
1736:          INC DPTR
1737:          SUBB A,R7
1738:          JZ N_SUB1
1739:          INC DPTR
1740:          INC DPTR
1741:          SJMP NOT_EQUAL
1742: N_SUB1:   CLR C
1743:          CLR A
1744:          MOVC A,@A+DPTR
1745:          INC DPTR
1746:          SUBB A,R6
1747:          JZ N_SUB2
1748:          INC DPTR
1749:          SJMP NOT_EQUAL
1750: N_SUB2:   CLR C
1751:          CLR A
1752:          MOVC A,@A+DPTR
1753:          INC DPTR
1754:          SUBB A,R5
1755:          JZ EQUAL
1756:          ;
1757: NOT_EQUAL: MOV R3,#11H
1758:          RET
1759:          ;
1760: EQUAL:    MOV R3,#22H
1761:          RET
1762:          ;
1763:          ;
1764: ;===== read code card from EE_CODECARD to RAM_NUMCARD =====
1765:          ;
1766: CHK_NUMCARD1: MOV KEEP_CODECARD,R7
1767:          MOV KEEP_CODECARD+1,R6
1768:          MOV KEEP_CODECARD+2,R5
1769: READ_CODE:  MOV DPTR,#EE_TOTALCARD
1770:          LCALL READ_EE
1771:          CJNE A,#0FFH,TR_NUMCARD2
1772:          MOV R3,#11H
1773:          RET
1774: TR_NUMCARD2: CJNE A,#0,TR_NUMCARD1
1775:          MOV R3,#11H
1776:          RET
1777:          ;
1778: TR_NUMCARD1: MOV RAM_NUMCARD,A
1779:          MOV R6,A
1780:          MOV R7,#00
1781:          ;
1782: STA_NUM1:  MOV DPTR,#EE_CODECARD
1783: STA_NUM2:  MOV R0,#RAM_CODECARD
1784:          MOV R5,#03H
1785: STA_NUM3:  CLR C
1786:          CLR A
1787:          LCALL READ_EE
1788:          MOV @R0,A
1789:          INC R0
1790:          INC DPTR
1791:          DJNZ R5,STA_NUM3
1792:          ;

```

ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1793:      MOV A, RAM_CODECARD
1794:      SUBB A, KEEP_CODECARD
1795:      JNZ N_CODE1
1796:      MOV A, RAM_CODECARD+1
1797:      SUBB A, KEEP_CODECARD+1
1798:      JNZ N_CODE1
1799:      MOV A, RAM_CODECARD+2
1800:      SUBB A, KEEP_CODECARD+2
1801:      JNZ N_CODE1
1802:      ;
1803:      INC R7
1804:      MOV A, R7
1805:      ADD A, #TOTAL_SUB
1806:      MOV R4, A
1807:      MOV R3, #22H
1808:      RET
1809:      ;
1810: N_CODE1:  INC R7
1811:      DJNZ R6, STA_NUM2
1812:      MOV R3, #11H
1813:      RET
1814:      ;
1815: ; ##### SET UP PRICE, LITE, FILL MONEY, #####
1816: SETUP:  MOV DPTR, #SET_1
1817:      LCALL SHOW_ROW
1818:      MOV A, #01H
1819:      LCALL WRLINE
1820:      MOV DPTR, #SET_2
1821:      LCALL SHOW_ROW
1822:      MOV A, #02H
1823:      LCALL WRLINE
1824:      MOV R5, #20H
1825:      LCALL DELAY_SHOW
1826:      MOV DPTR, #SET_3
1827:      LCALL SHOW_ROW
1828:      MOV A, #01H
1829:      LCALL WRLINE
1830:      MOV DPTR, #SET_4
1831:      LCALL SHOW_ROW
1832:      MOV A, #02H
1833:      LCALL WRLINE
1834: START_SET:  LCALL KEY_IN
1835:      CJNE A, #01H, NEXT_SET1
1836:      LJMP FILL_MONEY
1837: NEXT_SET1:  CJNE A, #02H, NEXT_SET2
1838:      SJMP EDIT_PWD
1839: NEXT_SET2:  CJNE A, #03, NEXT_SET3
1840:      LJMP REGISTER_CARD
1841: NEXT_SET3:  CJNE A, #KEY_CLR, START_SET
1842:      LJMP START_FUNC
1843:      ;
1844: EDIT_PWD:  MOV DPTR, #OLD_PWD1
1845:      LCALL SHOW_ROW
1846:      MOV A, #01H
1847:      LCALL WRLINE
1848:      MOV R0, #RAM_PWD
1849:      LCALL START_PWD1
1850:      MOV A, RAM_PWD
1851:      XRL A, KEEP_PWD
1852:      JNZ ERROR_PWD
1853:      MOV A, RAM_PWD+1
1854:      XRL A, KEEP_PWD+1
1855:      JNZ ERROR_PWD
1856:      MOV A, RAM_PWD+2

```

ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1921:      MOV A,#02H
1922:      LCALL WRLINE
1923:      MOV R5,#20H
1924:      LCALL DELAY_SHOW
1925:      LJMP START_FUNC
1926:      ;
1927: CAN_FILL:  MOV DPTR,#FILL_M1
1928:      LCALL SHOW_ROW
1929:      MOV A,#01H
1930:      LCALL WRLINE
1931: START_FILL: MOV DPTR,#FILL_M2
1932:      LCALL SHOW_ROW
1933:      MOV A,#02H
1934:      LCALL WRLINE
1935:      MOV KEEP_FILL,#0FFH
1936:      MOV KEEP_FILL+1,#0FFH
1937:      MOV KEEP_FILL+2,#0FFH
1938:      MOV A,#48H
1939:      LCALL GOTO
1940:      MOV R5,#00H
1941:      MOV R0,#KEEP_FILL
1942: AGAIN_FILL: LCALL KEY_IN
1943:      CJNE A,#KEY_CLR,FILL1
1944:      CJNE R5,#00H,AGAIN_F2
1945:      LJMP START_FUNC
1946: AGAIN_F2:  SJMP START_FILL
1947: FILL1:    CJNE A,#KEY_BATH,FILL2
1948:      SJMP AGAIN_FILL
1949: FILL2:    CJNE A,#KEY_LITE,FILL3
1950:      SJMP AGAIN_FILL
1951: FILL3:    CJNE A,#KEY_MANUAL,FILL4
1952:      SJMP AGAIN_FILL
1953: FILL4:    CJNE A,#KEY_SET,FILL5
1954:      LJMP SETUP
1955: FILL5:    CJNE A,#KEY_ENTER,FILL6
1956:      CJNE R5,#00H,EXCUTE_F1
1957:      SJMP AGAIN_FILL
1958: EXCUTE_F1: CJNE R5,#01H,EXCUTE_F2
1959:      MOV KEEP_FILL+2,KEEP_FILL
1960:      MOV KEEP_FILL+1,#0FFH
1961:      MOV KEEP_FILL,#0FFH
1962: EXCUTE_F2: CJNE R5,#02H,EXCUTE_F3
1963:      MOV KEEP_FILL+2,KEEP_FILL+1
1964:      MOV KEEP_FILL+1,KEEP_FILL
1965:      MOV KEEP_FILL,#0FFH
1966: EXCUTE_F3: SJMP SUM_FILL
1967: FILL6:    CJNE A,#KEY_ZERO,FILL7
1968:      CJNE R5,#00H,AGAIN_F3
1969:      SJMP START_FILL
1970: AGAIN_F3:  INC R5
1971:      MOV A,#00H
1972:      SJMP SHOW_FILL
1973: FILL7:    INC R5
1974:      LCALL VERT_KEY
1975: SHOW_FILL: MOV @R0,A
1976:      INC R0
1977:      ADD A,#30H
1978:      LCALL WRBYTE
1979:      CJNE R5,#03H,AGAIN_FILL
1980:      MOV A,#50H
1981:      LCALL GOTO
1982: EXCUTE_F5: LCALL KEY_IN
1983:      CJNE A,#KEY_ENTER,EXCUTE_F4
1984:      SJMP SUM_FILL

```

ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1985: EXECUTE_F4:    CJNE A,#KEY_CLR,EXECUTE_F5
1986:                LJMPP START_FILL
1987:                ;
1988: SUM_FILL:      MOV R0,#KEEP_FILL+2
1989:                MOV R1,#KEEP_MONEY+2
1990:                MOV R5,KEEP_MONEY
1991:                MOV R6,KEEP_MONEY+1
1992:                MOV R7,KEEP_MONEY+2
1993:                LCALL ADD_BATH
1994:                CJNE A,#11H,ERR_FILL
1995:                MOV R0,#KEEP_MONEY
1996:                LCALL CHK_250
1997:                CJNE A,#22H,TRU_FILL
1998:                MOV KEEP_MONEY,R5
1999:                MOV KEEP_MONEY+1,R6
2000:                MOV KEEP_MONEY+2,R7
2001:                ;
2002: ERR_FILL:      MOV DPTR,#ERR_FILL1
2003:                LCALL SHOW_ROW
2004:                MOV A,#01H
2005:                LCALL WRLINE
2006:                MOV DPTR,#ERR_FILL2
2007:                LCALL SHOW_ROW
2008:                MOV A,#02H
2009:                LCALL WRLINE
2010:                MOV R5,#20H
2011:                LCALL DELAY_SHOW
2012:                LJMPP FILL_MONEY
2013: TRU_FILL:     MOV R4,POSITION_SUB
2014:                MOV DPTR,#EE_MONEY-5
2015: TRUE_F1:      INC DPTR
2016:                INC DPTR
2017:                INC DPTR
2018:                INC DPTR
2019:                INC DPTR
2020:                DJNZ R4,TRUE_F1
2021:                MOV R5,#03H
2022:                MOV R0,#KEEP_MONEY
2023: TRUE_F2:      MOV A,@R0
2024:                LCALL WRITE_EE
2025:                INC DPTR
2026:                INC R0
2027:                DJNZ R5,TRUE_F2
2028: SEW_SER:      LJMPP START_FUNC
2029:                ;
2030:                ;
2031: REGISTER_CARD:MOV DPTR,#FILL_PWD
2032:                LCALL SHOW_ROW
2033:                MOV A,#01H
2034:                LCALL WRLINE
2035:                MOV R0,#RAM_PWD
2036:                LCALL START_PWD1
2037:                MOV A,RAM_PWD
2038:                XRL A,KEEP_PWD
2039:                JNZ ERROR_PWD2
2040:                MOV A,RAM_PWD+1
2041:                XRL A,KEEP_PWD+1
2042:                JNZ ERROR_PWD2
2043:                MOV A,RAM_PWD+2
2044:                XRL A,KEEP_PWD+2
2045:                JNZ ERROR_PWD2
2046:                MOV A,RAM_PWD+3
2047:                XRL A,KEEP_PWD+3
2048:                JNZ ERROR_PWD2

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2049:          JB LIFT_HOOK,KEY_REGIS
2050:          LJMP CAN_REGISTER
2051:          ;
2052: ERROR_PWD2:  MOV DPTR,#ERR_PWD1
2053:          LCALL SHOW_ROW
2054:          MOV A,#01H
2055:          LCALL WRLINE
2056:          MOV DPTR,#ERR_PWD2
2057:          LCALL SHOW_ROW
2058:          MOV A,#02H
2059:          LCALL WRLINE
2060:          MOV R5,#20H
2061:          LCALL DELAY_SHOW
2062:          LJMP START_FUNC
2063:          ;
2064: KEY_REGIS:  MOV R0,#KEEP_DTMF
2065: STAR_CODE:  MOV DPTR,#REGISTER1
2066:          LCALL SHOW_ROW
2067:          MOV A,#01H
2068:          LCALL WRLINE
2069: STAR_CODE1: MOV DPTR,#MYNUM_1
2070:          LCALL SHOW_ROW
2071:          MOV A,#02H
2072:          LCALL WRLINE
2073:          MOV A,#4BH
2074:          LCALL GOTO
2075:          MOV R5,#00H
2076:          LCALL KEEP_REGIS
2077:          MOV A,KEEP_DTMF
2078:          SWAP A
2079:          ADD A,KEEP_DTMF+1
2080:          MOV KEEP_CODECARD,A
2081:          MOV A,KEEP_DTMF+2
2082:          SWAP A
2083:          ADD A,KEEP_DTMF+3
2084:          MOV KEEP_CODECARD+1,A
2085:          MOV KEEP_CODECARD+2,KEEP_DTMF+4
2086:          ;
2087: ;          ===== SUB REGISTE CARD =====
2088: ;          input= @R0= KEEP_CODECARD
2089: CAN_REGISTER: MOV DPTR,#EE_TOTALCARD
2090:          LCALL READ_EE
2091:          ;
2092:          CJNE A,#0,REG_2
2093:          MOV R4,#1
2094:          MOV RAM_NUMCARD,#1
2095:          SJMP ST_REGIS1
2096:          ;
2097: REG_2:      CJNE A,#0FFH,REG_1
2098:          MOV R4,#1
2099:          MOV RAM_NUMCARD,#1
2100:          SJMP ST_REGIS1
2101:          ;
2102: REG_1:      CJNE A,#100,ST_REGIS
2103:          LJMP NO_REGIS
2104:          ;
2105: ST_REGIS:   MOV R4,A
2106:          MOV RAM_NUMCARD,A
2107:          ;
2108: ST_REGIS1:  MOV DPTR,#EE_CODECARD-3
2109: ST_REGIS2:  INC DPTR
2110:          INC DPTR
2111:          INC DPTR
2112:          DJNZ R4,ST_REGIS2

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2113:          ;
2114:          MOV R5,#03H
2115:          MOV R0,#KEEP_CODECARD
2116: ST_REGIS3:  MOV A,@R0
2117:          LCALL WRITE_EE
2118:          INC DPTR
2119:          INC R0
2120:          DJNZ R5,ST_REGIS3
2121:          ;
2122:          INC RAM_NUMCARD
2123:          MOV DPTR,#EE_TOTALCARD
2124:          MOV A, RAM_NUMCARD
2125:          LCALL WRITE_EE
2126:          JB LIFT_HOOK,COMPLETE_RE
2127:          LJMP VOICE14_GRI
2128:          ;
2129: COMPLETE_RE: MOV DPTR,#COMP_RE1
2130:          LCALL SHOW_ROW
2131:          MOV A,#01H
2132:          LCALL WRLINE
2133:          MOV DPTR,#COMP_RE2
2134:          LCALL SHOW_ROW
2135:          MOV A,#02H
2136:          LCALL WRLINE
2137:          MOV R5,#0FH
2138:          LCALL DELAY_SHOW
2139: GO_NEWSER:  LJMP START_FUNC
2140:          ;
2141: NO_REGIS:   JB LIFT_HOOK,NO_REGIS1
2142:          LJMP VOICE20_GRI
2143:          ;
2144: NO_REGIS1:  MOV DPTR,#NOT_REG1
2145:          LCALL SHOW_ROW
2146:          MOV A,#01H
2147:          LCALL WRLINE
2148:          MOV DPTR,#NOT_REG2
2149:          LCALL SHOW_ROW
2150:          MOV A,#02H
2151:          LCALL WRLINE
2152:          MOV R5,#20H
2153:          LCALL DELAY_SHOW
2154:          LJMP START_FUNC
2155:          ;
2156: ;          %%%%%%%%% SUB CHECK BUY OVER OR NOT ? %%%%%%%%%
2157: ;%%%%%%%% KEEP_MONEY= you have, RAM_BATH= you buy %%%%%%%%%
2158: ;%%%%%%%% B= you have, R4= you buy, R2= sesasult %%%%%%%%%
2159: CHK_ENOUGH: MOV R0,#KEEP_MONEY
2160:          LCALL MERGE_HEX
2161:          MOV B,A
2162:          MOV R0,#RAM_BATH
2163:          LCALL MERGE_HEX
2164:          MOV R4,A
2165:          MOV A,B
2166:          SUBB A,R4
2167:          JC NOT_ENOUGH
2168:          MOV R2,A
2169:          MOV A,#11H
2170:          RET
2171: NOT_ENOUGH: MOV A,#22H
2172:          RET
2173:          ;
2174: ; %%%%%%%%% SUB MERGE 3 DIGIT TO 1 BYTE %%%%%%%%%
2175: ; %%%%%%%%% R0=input, A=output %%%%%%%%%
2176: MERGE_HEX:  MOV A,@R0

```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2177:          MOV R4,#00H
2178:          CJNE A,#0FFH,NEXT_M100
2179:          CLR A
2180:          SJMP END_M100
2181: NEXT_M100:  CJNE A,#00H,M100_1
2182:          SJMP END_M100
2183: M100_1:    MOV R5,A
2184:          CLR A
2185: MERGE_100:  ADD A,#100
2186:          DJNZ R5,MERGE_100
2187: END_M100:   ADD A,R4
2188:          MOV R4,A
2189:          INC R0
2190:          ;
2191:          MOV A,@R0
2192:          CJNE A,#0FFH,NEXT_M10
2193:          SJMP END_M10
2194: NEXT_M10:   CJNE A,#00H,M10_1
2195:          SJMP END_M10
2196: M10_1:     MOV R5,A
2197:          CLR A
2198: MERGE_10:  ADD A,#10
2199:          DJNZ R5,MERGE_10
2200: END_M10:   ADD A,R4
2201:          MOV R4,A
2202:          INC R0
2203:          ;
2204:          MOV A,@R0
2205:          CJNE A,#0FFH,END_M1
2206:          CLR A
2207: END_M1:    ADD A,R4
2208:          RET
2209:          ;
2210: ;%%%%%%%% ENABLE PLUSE LCD %%%%%%%%%
2211: EPLUSE:   PUSH 07
2212:          SETB E_LCD
2213:          MOV R7,#0FFH
2214:          DJNZ R7,$
2215:          CLR E_LCD
2216:          POP 07
2217:          RET
2218:          ;
2219: ;%%%%%%%% SUB USE KEEP ASCII CODE ON RAM (16DIGIT) %%%%%%%%%
2220: ;FOR SHOW ON LCD 1 LINE
2221: ;R5=COUNTER
2222: ;R0=POSITION
2223: ;DPTR=ASCII CODE DATA FOR MOVE ON INTERNAL RAM(SHOW ON LCD)
2224: SHOW_ROW: PUSH 00
2225:          MOV R5,#16
2226:          MOV R0,#DATA_DISP
2227: SHOW1:    MOV A,#00
2228:          MOVC A,@A+DPTR
2229:          MOV @R0,A
2230:          INC R0
2231:          INC DPTR
2232:          DJNZ R5,SHOW1
2233:          POP 00
2234:          RET
2235:          ;
2236: ;%%%%%%%% WRITE ON LCD FOR DISPLAY %%%%%%%%%
2237: WRLINE:   CJNE A,#001H,WRL2
2238:          MOV A,#000H
2239:          LCALL GOTO
2240:          SJMP WRLM

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2241: WRL2:      MOV A,#040H
2242:             LCALL GOTO
2243: WRLM:      MOV R5,#16
2244:             MOV R1,#DATA_DISP
2245: WRL:       MOV A,@R1
2246:             LCALL WRBYTE
2247:             INC R1
2248:             DJNZ R5,WRL
2249:             RET
2250:            ;
2251: ;%%%%%%%% CONTROL WORD DEFIND POINT START ON LCD %%%%%%%%%
2252: GOTO:      SETB ACC.7
2253:             MOV P0,A
2254:             CLR RS_LCD
2255:             CLR RW_LCD
2256:             CLR E_LCD
2257:             LCALL WR_PORTA
2258:             LCALL EPLUSE
2259:             RET
2260:            ;
2261: ;%%%%%%%% USE SHOW EACH DIGIT ON LCD %%%%%%%%%
2262: WRBYTE:    SETB RS_LCD
2263:             MOV P0,A
2264:             LCALL WR_PORTA
2265:             LCALL EPLUSE
2266:             RET
2267:            ;
2268: ;%%%%%%%% DELAY SHOW FOR LCD %%%%%%%%%
2269: DELAY_SHOW:MOV R6,#0FFH
2270: DE2:      MOV R7,#0FFH
2271:             DJNZ R7,$
2272:             DJNZ R6,DE2
2273:             DJNZ R5,DELAY_SHOW
2274:             RET
2275:            ;
2276: ;%%%%%%%% CLEAR BUFFER OF DATA %%%%%%%%%
2277: CLR_BUFF:  PUSH B
2278:             MOV B,#25
2279:             MOV R0,#KEEP_DATA
2280: CLR_BUF1:  CLR A
2281:             MOV @R0,A
2282:             INC R0
2283:             DJNZ B,CLR_BUF1
2284:             POP B
2285:             RET
2286:            ;
2287: ;%%%%%%%% GET MAGNETIC CLOCK %%%%%%%%%
2288: GET_CLK:   JB CLK_CARD,$
2289:             NOP
2290:             NOP
2291:             JNB CLK_CARD,$
2292:             NOP
2293:             NOP
2294:             RET
2295:            ;
2296: ;%%%%%%%% PARITY ERROR & READ DATA ERROR %%%%%%%%%
2297: DATA_ERROR:MOV DPTR,#REIN_1
2298:             LCALL SHOW_ROW
2299:             MOV A,#1
2300:             LCALL WRLINE
2301:             MOV DPTR,#REIN_2
2302:             LCALL SHOW_ROW
2303:             MOV A,#2
2304:             LCALL WRLINE

```

เอกสารนี้เป็นเอกสารต้นฉบับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2305:          RET
2306:          ;
2307: ;%%%%%%%% MISSING INSERT CARD!!!! %%%%%%%%%
2308: MISS_INS:  MOV DPTR,#MISS1_IN
2309:          LCALL SHOW_ROW
2310:          MOV A,#01H
2311:          LCALL WRLINE
2312:          MOV DPTR,#MISS2_IN
2313:          LCALL SHOW_ROW
2314:          MOV A,#02H
2315:          LCALL WRLINE
2316:          RET
2317:          ;
2318: ;%%%%%%%% RECEIVER KEYBOARD %%%%%%%%%
2319: ;OUT=A 01(1),02(2),03(3),04(BATH),05(4),06(5)
2320: ;07(6),08(LITE),09(7),0A(8),0B(9),0C(MANUAL)
2321: ;0D(SET),0E(0),0F(CLR),10(ENTER) TOTAL 16 KEY
2322: KEY_IN:    MOV P0,#0FFH
2323:          LCALL WR_PCLOW
2324: ROW1:     MOV P0,#1111110B
2325:          LCALL WR_PCLOW
2326:          LCALL SCAN_KEY
2327:          MOV A,#00H
2328:          ADD A,R3
2329:          CJNE A,#00H,HAVE_KEY
2330:          MOV P0,#0FFH
2331:          LCALL WR_PCLOW
2332:          ;
2333: ROW2:     MOV P0,#11111101B
2334:          LCALL WR_PCLOW
2335:          LCALL SCAN_KEY
2336:          MOV A,#04H
2337:          ADD A,R3
2338:          CJNE A,#04H,HAVE_KEY
2339:          MOV P0,#0FFH
2340:          LCALL WR_PCLOW
2341:          ;
2342: ROW3:     MOV P0,#11111011B
2343:          LCALL WR_PCLOW
2344:          LCALL SCAN_KEY
2345:          MOV A,#08H
2346:          ADD A,R3
2347:          CJNE A,#08H,HAVE_KEY
2348:          MOV P0,#0FFH
2349:          LCALL WR_PCLOW
2350:          ;
2351: ROW4:     MOV P0,#11110111B
2352:          LCALL WR_PCLOW
2353:          LCALL SCAN_KEY
2354:          MOV A,#0CH
2355:          ADD A,R3
2356:          CJNE A,#0CH,HAVE_KEY
2357:          MOV P0,#0FFH
2358:          LCALL WR_PCLOW
2359:          ;
2360:          SJMP ROW1
2361: HAVE_KEY: RET
2362: SCAN_KEY: MOV R3,#00H
2363:          LCALL RD_PCHIG
2364:          JB P0.7,COLUMN2
2365:          LCALL DELAY
2366:          MOV R3,#01H
2367:          SJMP KEY_OUT1
2368: COLUMN2:  JB P0.6,COLUMN3

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของกรมส่งเสริมการค้าระหว่างประเทศ กระทรวงพาณิชย์ ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2369:          LCALL DELAY
2370:          MOV R3,#02H
2371:          SJMP KEY_OUT2
2372: COLUMN3:  JB P0.5,COLUMN4
2373:          LCALL DELAY
2374:          MOV R3,#03H
2375:          SJMP KEY_OUT3
2376: COLUMN4:  JB P0.4,NO_ROW
2377:          LCALL DELAY
2378:          MOV R3,#04H
2379:          SJMP KEY_OUT4
2380: NO_ROW:   SETB RD_SIG
2381:          MOV P2,#0FFH
2382:          LCALL DELAY
2383:          RET
2384:          ;
2385: KEY_OUT1: JNB P0.7,$
2386:          LCALL DELAY
2387:          SETB RD_SIG
2388:          MOV P2,#0FFH
2389:          RET
2390: KEY_OUT2: JNB P0.6,$
2391:          LCALL DELAY
2392:          SETB RD_SIG
2393:          MOV P2,#0FFH
2394:          RET
2395: KEY_OUT3: JNB P0.5,$
2396:          LCALL DELAY
2397:          SETB RD_SIG
2398:          MOV P2,#0FFH
2399:          RET
2400: KEY_OUT4: JNB P0.4,$
2401:          LCALL DELAY
2402:          SETB RD_SIG
2403:          MOV P2,#0FFH
2404:          RET
2405:          ;
2406: ;;;;;;;;; DELAY CPU EXECUTE ;;;;;;;;;
2407: DELAY:    MOV R7,#0FFH
2408:          DJNZ R7,$
2409:          RET
2410:          ;
2411: ;;;;;;;;; CONVERT KEYCODE TO KEY NUMBER ;;;;;;;;;
2412: VERT_KEY: PUSH 02
2413:          PUSH 05
2414:          MOV R2,A
2415:          MOV R5,#00
2416:          MOV DPTR,#VERT
2417: VERT_N1:  CLR A
2418:          CLR C
2419:          MOVC A,@A+DPTR
2420:          INC R5
2421:          INC DPTR
2422:          SUBB A,R2
2423:          JNZ VERT_N1
2424:          MOV A,R5
2425:          POP 05
2426:          POP 02
2427:          RET
2428:          ;
2429: VERT_FILL: PUSH 02
2430:          PUSH 05
2431:          MOV R2,A
2432:          MOV R5,#00

```

ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2433:          MOV DPTR,#VERT1
2434: VERT_FIL1: CLR A
2435:          CLR C
2436:          MOVC A,@A+DPTR
2437:          INC R5
2438:          INC DPTR
2439:          SUBB A,R2
2440:          JNZ VERT_FIL1
2441:          MOV A,R5
2442:          POP 05
2443:          POP 02
2444:          RET
2445:          ;
2446: VERT_DTMF: PUSH 02
2447:          PUSH 05
2448:          MOV A,@R0
2449:          MOV R2,A
2450:          MOV R5,#00
2451:          MOV DPTR,#VERT_DT1
2452: VERT_D1:  CLR A
2453:          CLR C
2454:          MOVC A,@A+DPTR
2455:          INC R5
2456:          INC DPTR
2457:          SUBB A,R2
2458:          JNZ CHK_VERT
2459:          CJNE R5,#10,SEEN
2460:          MOV A,#00
2461:          MOV @R0,A
2462:          POP 05
2463:          POP 02
2464:          RET
2465:          ;
2466: SEEN:    MOV A,R5
2467:          MOV @R0,A
2468:          POP 05
2469:          POP 02
2470:          RET
2471:          ;
2472: CHK_VERT: CJNE R5,#10,VERT_D1
2473:          MOV A,#0FFH
2474:          MOV @R0,A
2475:          POP 05
2476:          POP 02
2477:          RET
2478:          ;
2479:          ;
2480: ;%%%%% ADD BATH 3 BYTE NOT OVER 999 BATH %%%%%
2481:          ;@R0=i/p, @R1=i/p and o/p(KEEP_MONEY)
2482: ADD_BATH: PUSH 05H
2483:          MOV R5,#03H
2484: START_ADD: MOV A,@R0
2485:          CJNE @R0,#0FFH,BATH1
2486:          CJNE @R1,#0FFH,BATH2
2487:          MOV @R1,#0FFH
2488: BATH2:   SJMP BATH3
2489: BATH1:   CJNE @R1,#0FFH,BATH4
2490:          MOV @R1,A
2491: BATH3:   DEC R0
2492: BATH6:   DEC R1
2493:          DJNZ R5,START_ADD
2494:          MOV A,#11H
2495:          POP 05H
2496:          RET

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2497:          ;
2498: BATH4:    ADD A,@R1
2499:          MOV @R1,A
2500:          SUBB A,#0AH
2501:          JC  BATH3
2502:          CJNE R5,#01H,BATH7
2503:          MOV A,#22H
2504:          POP 05H
2505:          RET
2506:          ;
2507: BATH7:    MOV @R1,A
2508:          DEC R0
2509:          CJNE @R0,#0FFH,BATH5
2510:          MOV @R0,#01H
2511:          SJMP BATH6
2512: BATH5:    MOV A,@R0
2513:          ADD A,#01H
2514:          MOV @R0,A
2515:          SJMP BATH6
2516:          ;
2517: ;%%%%%%%% CHECK NOT OVER 250 BATH %%%%%%%%%
2518:          ;@R0=i/p
2519: CHK_250:  MOV A,@R0
2520:          CJNE A,#0FFH,CHK_FILL2
2521:          SJMP FILL_OK
2522: CHK_FILL2: SUBB A,#1
2523:          JZ  CHK_DI2
2524:          JC  FILL_OK
2525: FILL_MISS: MOV A,#22H
2526:          RET
2527:          ;
2528: FILL_OK:  MOV A,#11H
2529:          RET
2530:          ;
2531: CHK_DI2:  INC R0
2532:          MOV A,@R0
2533:          SUBB A,#5
2534:          JZ  CHK_DI3
2535:          JC  FILL_OK
2536:          SJMP FILL_MISS
2537: CHK_DI3:  INC R0
2538:          MOV A,@R0
2539:          CJNE A,#00H,FILL_MISS
2540:          SJMP FILL_OK
2541:          ;
2542: ;%%%%%%%%%% SUB SHOW OFF WATER %%%%%%%%%%%
2543: SH_OFF_WA: MOV DPTR,#HAVE_M1
2544:          LCALL SHOW_ROW
2545:          MOV A,#01H
2546:          LCALL WRLINE
2547:          MOV DPTR,#STAR_2
2548:          LCALL SHOW_ROW
2549:          MOV A,#02H
2550:          LCALL WRLINE
2551:          MOV R5,#10H
2552:          LCALL DELAY_SHOW
2553:          RET
2554:          ;
2555: ;%%%%%%%%%% SHOW YOU REMAIN MONEY UPDATE %%%%%%%%%%%
2556: REMAIN_MONEY: MOV DPTR,#REMAIN_1
2557:          LCALL SHOW_ROW
2558:          MOV A,#001H
2559:          LCALL WRLINE
2560:          MOV DPTR,#REMAIN_2

```

เอกสารนี้เป็นเอกสารที่... ใช้ในการทำงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2561:          LCALL SHOW_ROW
2562:          MOV A,#02H
2563:          LCALL WRLINE
2564:          MOV A,#45H
2565:          LCALL GOTO
2566:          MOV R5,#03H
2567:          MOV R0,#KEEP_MONEY
2568: REM_MONEY4: MOV A,@R0
2569:          CJNE A,#0FFH,REM_MONEY2
2570:          MOV A,#' '
2571:          SJMP REM_MONEY3
2572: REM_MONEY2: ADD A,#30H
2573: REM_MONEY3: LCALL WRBYTE
2574:          INC R0
2575:          DJNZ R5,REM_MONEY4
2576: ;
2577:          MOV A,#49H
2578:          LCALL GOTO
2579:          MOV R5,#02H
2580: REM_STANG1: MOV A,@R0
2581:          CJNE A,#0FFH,REM_STANG2
2582:          MOV A,#'0'
2583:          SJMP REM_STANG3
2584: REM_STANG2: ADD A,#30H
2585: REM_STANG3: LCALL WRBYTE
2586:          INC R0
2587:          DJNZ R5,REM_STANG1
2588:          MOV A,#40H+16
2589:          LCALL GOTO
2590:          MOV R5,#25H
2591:          LCALL DELAY_SHOW
2592:          RET
2593: ;
2594: ;          %%% READ ISD_2500 %%%
2595: ;
2596: ISD_2500:  SETB ISD_CE
2597:          LCALL WR_PORTB2
2598:          SETB ISD_PD
2599:          LCALL WR_PCLOW2
2600:          SETB P2.7
2601:          SETB P2.6
2602:          SETB P2.5
2603: ;
2604: PLAY_BACK: CLR C
2605:          CLR A
2606:          MOVC A,@A+DPTR
2607:          CLR ISD_PD
2608:          LCALL WR_PCLOW2
2609:          MOV P0,A
2610:          LCALL WR_PORTB
2611: ;
2612: PLUSE_ISD: SETB ISD_EOM
2613:          LCALL RD_PCHIG2
2614:          JNB ISD_EOM,$
2615:          SETB P2.7
2616:          SETB P2.6
2617:          SETB P2.5
2618: ;
2619:          CLR ISD_CE
2620:          LCALL WR_PORTB2
2621:          LCALL DELAY_ISD
2622:          SETB ISD_CE
2623:          LCALL WR_PORTB2
2624:

```

ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2625: WAIT_PLAY:   SETB ISD_EOM
2626:                LCALL RD_PCHIG2
2627:                JB ISD_EOM,$
2628:                SETB P2.7
2629:                SETB P2.6
2630:                SETB P2.5
2631:                ;
2632:                SETB ISD_PD
2633:                LCALL WR_PCLW2
2634:                RET
2635:                ;
2636: DELAY_ISD:     PUSH 05
2637:                PUSH 06
2638:                MOV R7,#04H
2639: DE_ISD2:      MOV R6,#0FFH
2640: DE_ISD1:      MOV R5,#0FFH
2641:                DJNZ R5,$
2642:                DJNZ R6,DE_ISD1
2643:                DJNZ R7,DE_ISD2
2644:                POP 06
2645:                POP 05
2646:                RET
2647:                ;
2648: ;             %%% KEEP KEY DTMF %%%
2649:                ;
2650: DTMF_5DIG:    MOV R4,#05H
2651:                SJMP STA_DTMF1
2652: DTMF_4DIG:    MOV R4,#04H
2653:                SJMP STA_DTMF1
2654: DTMF_3DIG:    MOV R4,#03H
2655:                SJMP STA_DTMF1
2656: DTMF_1DIG:    MOV R4,#01H
2657:                ;
2658: STA_DTMF1:    PUSH 00
2659:                PUSH 03
2660: STA_DTMF:     MOV R0,#KEEP_DTMF
2661:                ;
2662: NEW_DTMF:     LCALL DELAY_DTMF
2663:                LCALL STROE_DTMF
2664:                MOV @R0,A
2665:                INC R0
2666:                LCALL DE_PRESS
2667:                DJNZ R4,NEW_DTMF
2668:                POP 03
2669:                POP 00
2670:                RET
2671:                ;
2672: DELAY_DTMF:   MOV R3,#40H
2673: WAIT_DTMF2:   MOV R6,#0FFH
2674: WAIT_DTMF1:   MOV R5,#0FFH
2675: WAIT_DTMF:    CLR C
2676:                SETB MT_STD
2677:                LCALL RD_PCHIG2
2678:                MOV C,MT_STD
2679:                SETB RD_SIG
2680:                SETB P2.7
2681:                SETB P2.6
2682:                SETB P2.5
2683:                JNC WAIT_DTMF3
2684:                RET
2685:                ;
2686: WAIT_DTMF3:   DJNZ R5,WAIT_DTMF
2687:                DJNZ R6,WAIT_DTMF1
2688:                DJNZ R3,WAIT_DTMF2

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2689:          SETB LIFT_HOOK
2690:          LJMP BEGIN
2691:          ;
2692: STROE_DTMF: MOV P0,#0FH
2693:          LCALL RD_PORTA2
2694:          MOV A,P0
2695:          ANL A,#0FH
2696:          SETB RD_SIG
2697:          SETB P2.7
2698:          SETB P2.6
2699:          SETB P2.5
2700:          RET
2701:          ;
2702: DE_PRESS:  SETB MT_STD
2703:          LCALL RD_PCHIG2
2704:          MOV C,MT_STD
2705:          JC DE_PRESS
2706:          RET
2707:          ;
2708: ;%%%%%%%% COMPARE KEYING PASSWORD FROM DTMF %%%%%%%%%
2709:          ;
2710: COMP_DTMF: MOV A,KEEP_PWD
2711:          XRL A,KEEP_DTMF
2712:          JNZ ERR_DTMF
2713:          MOV A,KEEP_PWD+1
2714:          XRL A,KEEP_DTMF+1
2715:          JNZ ERR_DTMF
2716:          MOV A,KEEP_PWD+2
2717:          XRL A,KEEP_DTMF+2
2718:          JNZ ERR_DTMF
2719:          MOV A,KEEP_PWD+3
2720:          XRL A,KEEP_DTMF+3
2721:          MOV R3,#11H
2722:          RET
2723:          ;
2724: ERR_DTMF:  MOV R3,#22H
2725:          RET
2726: ;%%%%%%%% keep code card for register %%%%%%%%%
2727: ;input= KEEP_DTMF ,output keep to KEEP_CODECARD
2728: KEEP_REGIS: LCALL KEY_IN
2729:          CJNE A,#KEY_BATH,CODE_1
2730:          SJMP KEEP_REGIS
2731: CODE_1:    CJNE A,#KEY_LITE,CODE_2
2732:          SJMP KEEP_REGIS
2733: CODE_2:    CJNE A,#KEY_MANUAL,CODE_3
2734:          SJMP KEEP_REGIS
2735: CODE_3:    CJNE A,#KEY_SET,CODE_4
2736:          SJMP KEEP_REGIS
2737: CODE_4:    CJNE A,#KEY_CLR,CODE_5
2738:          CJNE R5,#00H,CLR_CODE1
2739:          SJMP KEEP_REGIS
2740: CLR_CODE1: DEC R0
2741:          DJNZ R5,CLR_CODE1
2742:          LJMP STAR_CODE1
2743: CODE_5:    CJNE A,#KEY_ENTER,CODE_6
2744:          SJMP KEEP_REGIS
2745: CODE_6:    CJNE A,#KEY_ZERO,CODE_7
2746:          MOV A,#00H
2747:          SJMP CODE_8
2748: CODE_7:    LCALL VERT_KEY
2749: CODE_8:    MOV @R0,A
2750:          ADD A,#30H
2751:          LCALL WRBYTE
2752:          INC R0

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับใช้ในการเรียนการสอนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2753:          INC R5
2754:          CJNE R5,#05H,KEEP_REGIS
2755: END_CODE1: MOV A,#50H
2756:          LCALL GOTO
2757:          LCALL KEY_IN
2758:          CJNE A,#KEY_CLR,END_CODE2
2759:          DEC R0
2760:          DEC R0
2761:          DEC R0
2762:          DEC R0
2763:          DEC R0
2764:          LJMP STAR_CODE1
2765: END_CODE2: CJNE A,#KEY_ENTER,END_CODE1
2766:          RET
2767:          ;
2768:          ;
2769:          PASSWORD_1: DB "***SET PASSWORD**"
2770:          PASSWORD_2: DB " FOR FILL MONEY "
2771:          PASSWORD_3: DB " DEFIND 4 DIGIT "
2772:          PASSWORD_4: DB " PASSWORD=      ."
2773:          VERT:      DB 01H,02H,03H,05H,06H,07H,09H,0AH,0BH
2774:          VERT1:     DB 01H,02H,03H,04H,05H,06H,07H,08H,09H
2775:          VERT_DT1:  DB 01H,02H,03H,04H,05H,06H,07H,08H,09H,0AH
2776:          POWER1:   DB "PUT WATER BUCKET"
2777:          POWER2:   DB " INSERT CARD.** "
2778:          REIN_1:   DB " PLEASE REINSERT"
2779:          REIN_2:   DB "  CARD AGAIN  "
2780:          MISS1_IN: DB "INSERT MISSING!!"
2781:          MISS2_IN: DB "CHECK & REINSERT"
2782:          MYNUM_1:  DB " CARD No=>  "
2783:          MYNUM_2:  DB "HAVE=      BATH"
2784:          REGISTER1:DB " REGISTER CARD. "
2785:          REGIS_2:  DB "PLEASE REGISTER."
2786:          NO_MONEY2:DB " HAVE = 0 BATH "
2787:          REMOV_1:  DB " THANK YOU  "
2788:          REMOV_2:  DB " (REMOVE CARD) "
2789:          SET_1:    DB " 1= FILL MONEY. "
2790:          SET_2:    DB " 2=SET PASSWORD "
2791:          SET_3:    DB " 3=REGISTER CARD"
2792:          SET_4:    DB " ***** "
2793:          OLD_PWD1: DB " PLEASE KEY OLD "
2794:          ERR_PWD1: DB "PASSWORD MISSING"
2795:          ERR_PWD2: DB "!!!! SORRY !!!!!"
2796:          FILL_PWD: DB " PLEASE KEYIN "
2797:          FILL_M1:  DB "YOU WANT TO FILL"
2798:          FILL_M2:  DB "MONEY=      BATH"
2799:          ERR_FILL1:DB "YOU FILL MISTAKE"
2800:          ERR_FILL2:DB "NOT OVER 250BATH"
2801:          BATH_1:  DB "YOU WANT TO BUY "
2802:          BATH_2:  DB "=====>      BATH."
2803:          MISS_B1: DB " YOU HAVE MONEY "
2804:          MISS_B2: DB " NOT ENOUGH !!!"
2805:          MANUAL_B1:DB "BUY=      BATH"
2806:          MANUAL_B2:DB "====>      BATH"
2807:          MANUAL_L1:DB "BUY =      LITE"
2808:          NO_WATER1:DB " ***I'M SORY*** "
2809:          NO_WATER2:DB " NOT HAVE WATER "
2810:          LITE_2:   DB "=====>      LITE."
2811:          LITE_1:   DB " NEW SETTING.  "
2812:          HAVE_M1:  DB "***OFF WATER****"
2813:          WATER_1:  DB " PAYING WATER.  "
2814:          STAR_2:   DB "*****"
2815:          REMAIN_1: DB " YOU ARE REMAIN "
2816:          REMAIN_2: DB " ==>      BATH"

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับใช้งานเพื่อการศึกษานานาชาติเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2817:      COMPLETE: DB " COMPLETE..... "
2818:      INCOMPLETE:DB "INCOMPLETE....."
2819:      CHARACTER:DB "123456789"
2820:      NOT_REG1: DB "NO REGISTER CARD"
2821:      NOT_REG2: DB " OVER EE_PROM  "
2822:      COMP_RE1: DB " REGISTER_CARD  "
2823:      COMP_RE2: DB "      OK.      "
2824:      STANBY1:  DB " PLEASE WAIT   "
2825:      STANBY2:  DB "SYSTEM'S SETTING"
2826:      ;
2827:      ;
2828:      ;
2829:      SUB1: DB 52H,08H,00H
2830:      SUB2: DB 82H,42H,02H
2831:      SUB3: DB 00H,00H,00H
2832:      SUB4: DB 82H,42H,00H
2833:      ;
2834:      VOICE_1:DB 00H
2835:      VOICE_2:DB 12H
2836:      VOICE_3:DB 43H
2837:      VOICE_4:DB 58H
2838:      VOICE_5:DB 67H
2839:      VOICE_6:DB 96H
2840:      VOICE_7:DB 0A0H
2841:      VOICE_8:DB 0BFH
2842:      VOICE_9: DB 0D8H
2843:      VOICE_10: DB 0E8H
2844:      VOICE_11: DB 0F8H
2845:      VOICE_12: DB 14H
2846:      VOICE_13: DB 28H
2847:      VOICE_14: DB 37H
2848:      VOICE_15: DB 47H
2849:      VOICE_16: DB 50H
2850:      VOICE_17: DB 60H
2851:      VOICE_18: DB 7CH
2852:      VOICE_19: DB 98H
2853:      VOICE_20: DB 0A8H
2854:      ;
2855: BEAK:  END

```

## เอกสารอ้างอิง

1. ก้องเกียรติ ณ สีมา, “ทฤษฎีและการใช้งานไมโครเมอร์ ไอซี 555”, อิเลคทรอนิกส์เวิร์ลด์, พ.ศ. 2528, หน้า 19–21
2. สุนทร วิทสุรพจน์, “การใช้งานไมโครคอนโทรลเลอร์ ตระกูล 8051”, พ.ศ. 2527



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้