

ระบบการซื้อสินค้าที่ตู้ขายของผ่านอาร์เอฟไอดี

PURCHASING ITEMS FROM VENDING MACHINE VIA RFID



T103993

โดย

นาย ชีระศักดิ์ แซ่โจ้ว
นาย พรรษวุฒ อรุณกมล
นางสาว วันวิสาข์ จินดาแก้ว

เลขหมู่.....
เลขทะเบียน...103993
วัน,เดือน,ปี... 28 ต.ค. 2552



ปฏิญานินพจน์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

ภาควิชา
วิศวกรรมโทรคมนาคม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสาร
ผู้ตรวจ

ระบบการซื้อสินค้าที่ตู้ขายของผ่านอาร์เอฟไอดี

PURCHASING ITEMS FROM VENDING MACHINE VIA RFID

โดย

นาย ชีระศักดิ์ แซ่โจ้ว รหัส 48010405

นาย พรระฆูฒ อรุณกมล รหัส 48010585

นางสาว วันวิสาข์ จินดาแก้ว รหัส 48010811

อาจารย์ที่ปรึกษา

ผศ.ดร.พิเชฐ

ม่วงนวล

รศ.ดร.ไกรสิน

สงัดวัฒนา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2551

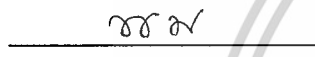
ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง ระบบการซื้อสินค้าที่ตู้ขายของผ่านอาร์เอฟไอดี

PURCHASING ITEMS FROM VENDING MACHINE VIA RFID

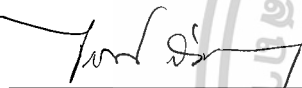
ผู้จัดทำ

- | | | |
|--------------------|-----------|----------|
| 1. นาย ชีระศักดิ์ | แซ่ใจ้ว | 48010405 |
| 2. นาย พรธรรม | อรุณกมล | 48010585 |
| 3. นางสาววันวิสาข์ | จินดาแก้ว | 48010811 |



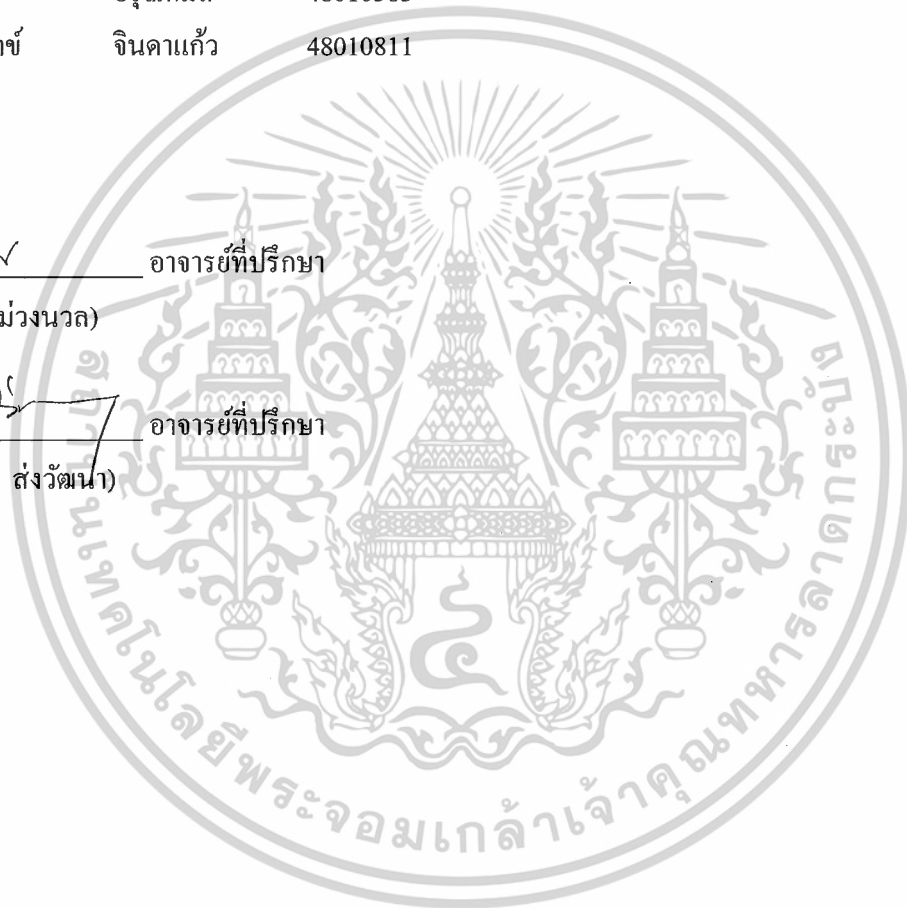
(ผศ.ดร.พิเชฐ ม่วงนวล)

อาจารย์ที่ปรึกษา



(รศ.ดร.ไกรสิน ส่องวัฒนา)

อาจารย์ที่ปรึกษา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบการซื้อสินค้าที่ตู้ขายของผ่านอาร์เอฟไอดี

PURCHASING ITEMS FROM VENDING MACHINE VIA RFID

โดย นาย ธีระศักดิ์ แซ่โจ้ว รหัส 48010405

นาย พรรษวุฒ อรุณกมล รหัส 48010585

นางสาว วันวิสาข์ จินดาแก้ว รหัส 48010811

อาจารย์ที่ปรึกษา

ผศ.ดร.พิเชฐ ม่วงนวล

รศ.ดร.ไกรสิน ส่งวัฒนา

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอการซื้อสินค้าจากตู้ขายของอัตโนมัติ โดยนำเอาเทคโนโลยีอาร์เอฟไอดีมาประยุกต์ใช้งาน ผู้ใช้บริการสามารถซื้อสินค้าโดยใช้อาร์เอฟไอดีแท็ก หรือ แทชซิม ซึ่งภายในจะเก็บจำนวนเงินไว้ เมื่อผู้ใช้บริการซื้อสินค้าเสร็จแล้ว ระบบก็จะทำการหักเงินและแสดงยอดเงินคงเหลือที่อยู่ในแท็ก หรือ แทชซิม ผ่านทางจอแอลซีดีให้ผู้บริการทราบ อีกทั้งยังมีระบบการตรวจสอบจำนวนสินค้าภายในตู้ เมื่อสินค้าหมดจะมีการส่งข้อความแจ้งเตือนผ่านระบบข้อความสั้นไปยังผู้ประกอบการ

ABSTRACT

This thesis proposes the purchasing items from vending machine by using RFID technology. The customer can buy the goods by using RFID tag or Touch Sim. When the customer finish a transaction, the system will deduct the money in RFID tag or Touch Sim and show the remaining balance on LCD to inform the customer. The system is also able to check stock in the machine. If the goods are sold out, the system will send an alert SMS to the entrepreneur.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
สารบัญ	III
สารบัญรูปภาพ	VI
สารบัญตาราง	IX

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ	1
1.5 ส่วนที่ บริษัท ทรู คอร์ปอเรชั่น ให้มา	2

บทที่ 2 ทฤษฎีและหลักการ

2.1 กล่าวนำ	3
2.2 ประวัติความเป็นมาของ RFID	3
2.2.1 ความหมายของ RFID	3
2.2.2 ส่วนประกอบของระบบ RFID	4
2.2.3 ประเภทของ RFID Tag	5
2.2.4 เครื่องอ่าน (Reader)	6
2.2.5 คุณสมบัติของระบบ RFID	7
2.2.6 ย่านความถี่สำหรับ RFID	8
2.3 ไมโครคอนโทรลเลอร์พีไอซี (PIC)	9
2.4 ไมโครคอนโทรลเลอร์เอวีอาร์ (AVR)	11
2.4.1 คุณสมบัติของไมโครคอนโทรลเลอร์ AVR	11
2.5 โมดูล (Module) ที่เกี่ยวข้องในการทำโครงการ	12
2.5.1 โมดูลแอลซีดี (Module LCD)	12
2.5.2 ส่วนประกอบหลักของโมดูล LCD	12
2.6 พอร์ตอนุกรม RS-232	13

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนสำหรับโรงเรียนนี้และห้ามเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ทางการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2	ระดับสัญญาณของ RS232	15
2.6.3	อัตราการส่งข้อมูล (Baud rate)	15
2.6.4	รูปแบบการสื่อสารแบบอนุกรม	15
2.7	ระบบ GSM	16
2.7.1	โครงสร้างของระบบเครือข่าย GSM	17
2.7.2	AT COMMAND	18
2.7.2.1	หลักการรับส่ง SMS ของโทรศัพท์มือถือ	18
2.7.2.2	กลุ่มคำสั่งในการส่ง SMS	19
2.8	IC 555	20
2.8.1	วงจรโมโนสเตเบิล (Mono-stable Multivibrator)	22
2.8.2	วงจรอะสเตเบิล (A-stable Multivibrator)	23
2.9	Stepping motor	23
2.9.1	การควบคุมการทำงานของ Stepping motor แบบ 4 เฟส	24
บทที่ 3 การออกแบบและการสร้าง		
3.1	การออกแบบระบบการซื้อสินค้า	27
3.2	ส่วนประกอบของเครื่องอ่าน / เขียนอาร์เอฟไอดี	28
3.2.1	คุณลักษณะหลักของเครื่องอ่าน/เขียนอาร์เอฟไอดี	28
3.3	ส่วนประกอบของอาร์เอฟไอดีแท็ก	30
3.3.1	คุณลักษณะหลักของอาร์เอฟไอดีแท็ก	30
3.3.2	ส่วนประกอบของบัตร Mifare S50	30
3.4	ส่วนประกอบของโมดูลแอลซีดี	31
3.5	ส่วนประกอบของคีย์แพด	32
3.6	ส่วนประกอบของสเต็ปมอเตอร์	33
3.6.1	การขับสเต็ปมอเตอร์แบบฟูลสเต็ป 2 เฟส	33
3.7	ภาพรวมของวงจรที่ใช้ในระบบการซื้อสินค้า	35
3.8	การออกแบบระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด	36
3.9	ส่วนประกอบของวงจรตรวจจับวัตถุ	36
3.9.1	หลักการสร้างสัญญาณความถี่ 38 กิโลเฮิร์ต	37
3.9.2	TSOP 4838	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.10 ส่วนประกอบของโมดูลจีเอสเอ็ม (GSM Module)	38
3.10.1 โครงสร้างของบอร์ด ET-GSM SIM300CZ V1	38
3.10.2 การเชื่อมต่อ โมดูลจีเอสเอ็มกับโปรแกรมไฮเปอร์เทอร์มินอล	40
3.10.3 ชุดคำสั่งเอทีคอมมานด์ที่ใช้ในการส่งข้อความสั้น	40
3.11 ภาพรวมของวงจรที่ใช้ในระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด	41
3.12 การออกแบบโปรแกรมเติมเงิน	42
3.12.1 ส่วนประกอบของเครื่องอ่าน/เขียนอาร์เอฟไอดี	43
3.13 โพล์ซาร์ตของระบบการซื้อสินค้าที่ตู้ขายของผ่านอาร์เอฟไอดี	44
3.13.1 โพล์ซาร์ตในส่วนของระบบการซื้อสินค้า	44
3.13.2 โพล์ซาร์ตในส่วนของระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด	45
บทที่ 4 ผลการทดลอง	
4.1 ผลการทดลองจากการใช้งานระบบการซื้อสินค้าผ่านอาร์เอฟไอดี	46
4.2 การทดลองใช้งาน โปรแกรมเติมเงิน	51
4.3 การทดลองของระบบแจ้งเตือนสินค้าหมดผ่านทางข้อความสั้น (SMS: Short Message Service)	55
บทที่ 5 สรุป	
5.1 สรุปผลการทดลอง	59
5.2 ปัญหาที่พบ	59
กิตติกรรมประกาศ	
เอกสารอ้างอิง	
ภาคผนวก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 ส่วนประกอบหลักของ RFID	3
รูปที่ 2.2 การใช้ Tag กับ เครื่องอ่าน Tag	4
รูปที่ 2.3 RFID Tag แบบ Passive	5
รูปที่ 2.4 RFID Tag แบบ Active	5
รูปที่ 2.5 โครงสร้างภายในของ Reader	6
รูปที่ 2.6 RFID Reader	7
รูปที่ 2.7 ย่านความถี่ ISM Band ที่นำระบบอาร์เอฟไอดีในปัจจุบัน	9
รูปที่ 2.8 โครงสร้างลักษณะของไมโครคอนโทรลเลอร์ PIC เบอร์ 18F4550	10
รูปที่ 2.9 โครงสร้างลักษณะของไมโครคอนโทรลเลอร์ AVR	11
รูปที่ 2.10 หน้าจอแสดงผล LCD ขนาด 16x2	12
รูปที่ 2.11 ลักษณะภายนอกและรายละเอียดขาของ RS-232	13
รูปที่ 2.12 การเชื่อมต่ออุปกรณ์ภายนอกผ่าน DB 9 แบบ 3 เส้น และการเชื่อมต่อแบบ Null Modem ตามลำดับ	15
รูปที่ 2.13 ระดับสัญญาณของ RS232C และระดับสัญญาณของ TTL	15
รูปที่ 2.14 การสื่อสารแบบซิงโครนัส (Synchronous)	16
รูปที่ 2.15 การสื่อสารแบบอะซิงโครนัส (Asynchronous)	16
รูปที่ 2.16 โครงสร้างภายในไอซี 555	21
รูปที่ 2.17 วงจร Mono-stable Multivibrator	22
รูปที่ 2.18 วงจร A-stable Multivibrator	23
รูปที่ 2.19 โครงสร้างของ Stepping motor	24
รูปที่ 3.1 ส่วนประกอบหลักของฮาร์ดแวร์ในส่วนของระบบการซื้อสินค้า	27
รูปที่ 3.2 เครื่องอ่าน/เขียนอาร์เอฟไอดี รุ่น SL015M	29
รูปที่ 3.3 การเชื่อมต่อเครื่องอ่าน/เขียนอาร์เอฟไอดี รุ่น SL015M กับไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) เบอร์ ATmega64	29
รูปที่ 3.4 บัตร Mifare S50	30
รูปที่ 3.5 ส่วนประกอบของบัตร Mifare S50	31
รูปที่ 3.6 การเชื่อมต่อ โมดูลแอลซีดีกับพอร์ตเอชของไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) เบอร์ ATmega64	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.7 คีย์แปดขนาด 4x4	32
รูปที่ 3.8 การเชื่อมต่อคีย์แปดกับพอร์ตบีของไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) เบอร์ ATmega64	32
รูปที่ 3.9 ETT สเต็ปมอเตอร์	33
รูปที่ 3.10 การเชื่อมต่อสเต็ปมอเตอร์กับพอร์ตดีและเอฟของไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) เบอร์ ATmega64	34
รูปที่ 3.11 ภาพรวมของวงจรที่ใช้ในส่วนของการระบบการซื้อสินค้า	35
รูปที่ 3.12 ส่วนประกอบหลักของฮาร์ดแวร์ในส่วนของการระบบการแจ้งเตือนเมื่อซื้อสินค้าภายในตู้ขายของหมด	36
รูปที่ 3.13 วงจรกำเนิดสัญญาณความถี่ 38 กิโลเฮิร์ต	36
รูปที่ 3.14 TSOP 4838 โมดูลรับสัญญาณอินฟราเรด 38 กิโลเฮิร์ต	37
รูปที่ 3.15 โมดูล ET-GSM SIM300CZ V1	38
รูปที่ 3.16 ส่วนการเชื่อมต่อโมดูลจีเอสเอ็มเข้ากับคอมพิวเตอร์	40
รูปที่ 3.17 ขั้นตอนการส่ง SMS ด้วยชุดคำสั่ง AT Command ผ่านทางโปรแกรม Hyper Terminal	40
รูปที่ 3.18 ภาพรวมของวงจรที่ใช้ในระบบการแจ้งเตือนเมื่อซื้อสินค้าภายในตู้ขายของหมด	41
รูปที่ 3.19 ส่วนของโปรแกรมเติมเงินที่ได้ออกแบบ	42
รูปที่ 3.20 วิธีการเติมเงินใส่บัตร (อาร์เอฟไอดีแท็ก)	42
รูปที่ 3.21 ภาพเมื่อมีการเติมเงินใส่ในบัตร (อาร์เอฟไอดีแท็ก) แล้ว	43
รูปที่ 3.22 เครื่องอ่าน/เขียนอาร์เอฟไอดี รุ่น RFID-M1	44
รูปที่ 3.23 โพล์วาร์ดส่วนของการระบบการซื้อสินค้า	44
รูปที่ 3.24 โพล์วาร์ดส่วนของการระบบการแจ้งเตือนเมื่อซื้อสินค้าภายในตู้ขายของหมด	45
รูปที่ 4.1 หน้าจอ LCD เมื่อยังไม่มีการใช้งาน	46
รูปที่ 4.2 เมื่อผู้ใช้บริการเลือกสินค้าชนิดที่ 14	47
รูปที่ 4.3 เมื่อผู้ใช้บริการเลือกสินค้าชนิดที่ 2	47
รูปที่ 4.4 รูปเมื่อผู้ใช้บริการนำ Tag มาแตะที่เครื่องอ่านเพื่อทำการหักเงินและจ่ายสินค้า	48
รูปที่ 4.5 สัญญาณที่ขา Tx ของ RFID module	48
รูปที่ 4.6 สัญญาณที่ขา Rx ของ RFID Module	49
รูปที่ 4.7 สัญญาณที่ขา 16 และ 15 ตามลำดับของ ULN 2003 แบบตามเข็มนาฬิกา	49
รูปที่ 4.8 สัญญาณที่ขา 14 และ 13 ตามลำดับของ ULN 2003 แบบตามเข็มนาฬิกา	50
รูปที่ 4.9 สัญญาณที่ขา 16 และ 15 ตามลำดับของ ULN 2003 แบบทวนเข็มนาฬิกา	50
รูปที่ 4.10 สัญญาณที่ขา 14 และ 13 ตามลำดับของ ULN 2003 แบบทวนเข็มนาฬิกา	51
รูปที่ 4.11 ขั้นตอนการเติมเงินเริ่มต้นให้ Touch Sim	51
รูปที่ 4.12 ผลที่ได้จากการเติมเงินเริ่มต้น	52
รูปที่ 4.13 ขั้นตอนการเติมเงินเพิ่มใส่ Touch Sim	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.14 ผลลัพธ์จากการเติมเงินเพิ่มใส่ Touch Sim	53
รูปที่ 4.15 ผลจากการอ่านค่า Touch Sim	53
รูปที่ 4.16 การแก้ไขข้อมูลใน Touch Sim กรณีที่เติมเงินเกินไป 15 บาท	54
รูปที่ 4.17 ผลลัพธ์ที่ได้จากการแก้ไขข้อมูล	54
รูปที่ 4.18 สัญญาณความถี่ที่ 38 kHz จาก IC555	55
รูปที่ 4.19 สัญญาณการเก็บประจุและคายประจุของคาร์ปาซิเตอร์	56
รูปที่ 4.20 สัญญาณเป็น 1 เมื่อยังมีสินค้าคงเหลือภายในตู้	56
รูปที่ 4.21 สัญญาณเป็น 0 เมื่อสินค้าภายในตู้หมด	57
รูปที่ 4.22 ข้อความที่ได้รับ	57
รูปที่ 4.23 ตู้ขายของอัตโนมัติ (Vending Machine)	58
รูปที่ 4.24 ตู้ขายของอัตโนมัติเมื่อมองจากด้านบน	58



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 คุณสมบัติไมโครคอนโทรลเลอร์ PIC 18F4550	9
ตารางที่ 2.2 รายละเอียดต่างๆของโมดูล LCD	12
ตารางที่ 2.3 รายละเอียดขาของ RS-232	14
ตารางที่ 2.4 หน้าที่ของแต่ละขาของไอซีเบอร์ 555	21
ตารางที่ 2.5 ตารางการควบคุมแบบ Full step 1 เฟส	25
ตารางที่ 2.6 ตารางการควบคุมแบบ Full step 2 เฟส	25
ตารางที่ 2.7 การควบคุมแบบ Half wave	26
ตารางที่ 3.1 ข้อมูลขาโมดูลอาร์เอฟไอดี	28
ตารางที่ 3.2 การทำงานของขดลวดแบบฟลูตตีป 2 เฟส	33
ตารางที่ 3.3 ข้อมูลของเครื่องอ่าน/เขียน RFID - M1	43



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

เนื่องจากปัจจุบันมีเครื่องขายของอัตโนมัติเพิ่มมากขึ้น แต่เครื่องขายของส่วนใหญ่ยังเป็นแบบหยอดเหรียญ ทำให้บางครั้งผู้ใช้บริการเกิดความไม่สะดวก เนื่องจากมีเหรียญไม่เพียงพอ นอกจากนี้เครื่องขายของอัตโนมัติที่มีอยู่ในปัจจุบันยังมีข้อเสียอยู่ตรงที่ เมื่อสินค้าในเครื่องขายของหมด ทางผู้ประกอบการจะไม่สามารถทราบได้ในทันที ทำให้เกิดผลกระทบต่อการค้าเงินธุรกิจเป็นอย่างมาก

ด้วยเหตุข้างต้นทำให้ผู้จัดทำมีความสนใจในการจัดทำโครงการที่เกี่ยวกับการทำระบบการซื้อสินค้าที่เครื่องขายของอัตโนมัติโดยใช้เทคโนโลยีอาร์เอฟไอดี (RFID: Radio Frequency Identification) และ การทำระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด โดยเครื่องขายของอัตโนมัติจะทำการติดเครื่องอ่าน/เขียนอาร์เอฟไอดี (RFID Read/Write Module) แทนเครื่องหยอดเหรียญ ผู้ใช้บริการสามารถที่จะซื้อสินค้าโดยการใช้อาร์เอฟไอดีแท็ก (RFID Tag) หรือ ทัชซิม (Touch Sim) ซึ่งภายในจะเก็บจำนวนเงินของผู้ใช้บริการไว้ และในส่วนของระบบการแจ้งเตือนสินค้า จะทำการติดเซนเซอร์ไว้เพื่อทำการตรวจสอบสินค้า เมื่อสินค้าหมด ระบบจะสั่งให้โมดูลจีเอสเอ็ม (GSM Module) ทำการส่งข้อมูลผ่านระบบข้อความสั้น (SMS: Short Message Service) เพื่อแจ้งให้ผู้ประกอบการได้ทราบได้ทันทีว่าตำแหน่งของเครื่องขายของอยู่ที่ไหน และสินค้าประเภทใดหมด

1.2 วัตถุประสงค์

- 1.2.1 เพื่อศึกษาการทำงานของเครื่องอ่าน/เขียนอาร์เอฟไอดี
- 1.2.1 เพื่อศึกษาการส่งข้อมูลผ่านระบบข้อความสั้น โดยใช้โมดูลจีเอสเอ็ม
- 1.2.3 เพื่อทำการศึกษาการใช้งานของไมโครคอนโทรลเลอร์พีไอซี (PIC) และเอวีอาร์ (AVR)
- 1.2.4 เพื่อศึกษาการเขียนโปรแกรม Visual C# 2008

1.3 ขอบเขตของโครงการ

- 1.3.1 ใช้อาร์เอฟไอดีแท็ก หรือ ทัชซิม ติดต่อกับ เครื่องอ่าน/เขียนอาร์เอฟไอดีได้
- 1.3.2 ส่งข้อความแจ้งเตือนผ่านระบบข้อความสั้นไปยังผู้ประกอบการเมื่อสินค้าหมดได้
- 1.3.3 ออกแบบและเขียนโปรแกรมเติมเงินโดยใช้โปรแกรม Visual C# 2008 ได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1.4.1 มีความรู้ ความเข้าใจเกี่ยวกับเทคโนโลยีอาร์เอฟไอดี และสามารถนำเทคโนโลยีอาร์เอฟไอดีไปประยุกต์ใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1.4.2 มีความรู้ความเข้าใจเกี่ยวกับการส่งข้อความผ่านระบบข้อความสั้น
- 1.4.3 มีความรู้เกี่ยวกับไมโครคอนโทรลเลอร์พีไอซี (PIC) และ เอวีอาร์ (AVR)
- 1.4.4 มีความรู้เกี่ยวกับการเขียนโปรแกรม Visual C# 2008
- 1.4.5 สามารถนำความรู้ที่ได้จากโครงการนี้ไปประยุกต์ใช้งานกับการทำระบบอื่นๆ ได้

1.5 ส่วนที่ บริษัท ทรุ คอปเปอร์เรชั่น ให้มา

- 1.5.1 ทัชซิม (Touch Sim) ซึ่งเป็นซิมรูปแบบใหม่พร้อมชิพอาร์เอฟไอดี
- 1.5.2 อบรมให้ความรู้เกี่ยวกับอาร์เอฟไอดี
- 1.5.3 เงินเบี่ยเลี้ยงสนับสนุน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 กล่าวนำ

ปัจจุบันจะเห็นว่าตู้ขายสินค้าส่วนใหญ่จะยังคงเป็นแบบตู้หยอดเหรียญ ซึ่งการใช้บริการจะต้องมีเหรียญตามจำนวนที่ต้องการ ทำให้บางครั้งผู้ให้บริการเกิดความไม่สะดวก เนื่องจากมีเหรียญไม่เพียงพอ นอกจากนี้เครื่องขายของอัตโนมัติที่มีอยู่ในปัจจุบันยังมีข้อเสียอยู่ตรงที่ เมื่อสินค้าในเครื่องขายของหมดทางผู้ประกอบการจะไม่สามารถทราบได้ในทันที ด้วยเหตุนี้จึงได้มีการจัดทำโครงการที่เกี่ยวกับการทำระบบการซื้อสินค้าที่เครื่องขายของอัตโนมัติโดยใช้เทคโนโลยีอาร์เอฟไอดี (RFID: Radio Frequency Identification) และ การทำระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด

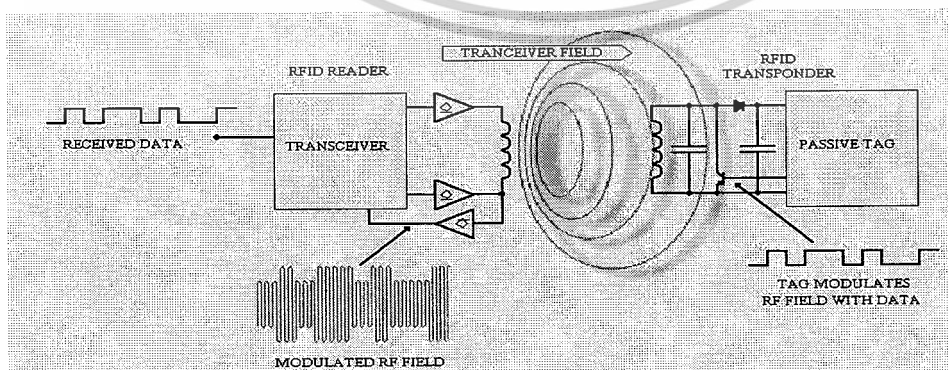
2.2 ประวัติความเป็นมาของ RFID

เทคโนโลยีของ RFID มีการพัฒนาอย่างต่อเนื่องมาตั้งแต่ต้น ค.ศ.1920 โดยที่อุปกรณ์ RFID ที่มีการประดิษฐ์ขึ้นใช้งานเป็นครั้งแรกนั้น เป็นผลงานของ Leon Theremin ซึ่งสร้างให้กับรัฐบาลของประเทศรัสเซียในปี ค.ศ. 1945 ซึ่งอุปกรณ์ที่สร้างขึ้นมาในเวลานั้นทำหน้าที่เป็นเครื่องมือดักจับสัญญาณไม่ได้ทำหน้าที่เป็นตัวระบุเอกลักษณ์อย่างที่ใช้งานกันอยู่ในปัจจุบัน

ปัจจุบันมีการพัฒนาเทคโนโลยีของอุปกรณ์ Tag และ Reader ให้มีประสิทธิภาพมากยิ่งขึ้น โดยการใช้งานและการบำรุงรักษาดีขึ้น สามารถทำงานร่วมกับเทคโนโลยีด้านการสื่อสารข้อมูลและโครงสร้างพื้นฐาน เพื่อการสื่อสารข้อมูลทั้งในปัจจุบันและอนาคต

2.2.1 ความหมายของ RFID

RFID ย่อมาจากคำว่า “Radio Frequency Identification” เป็นลักษณะของการใช้คลื่นความถี่วิทยุเพื่อระบุลักษณะเฉพาะของวัตถุแต่ละชิ้น ปัจจุบันมีการนำ RFID ไปประยุกต์ใช้งานในรูปแบบต่าง ๆ เช่น ระบบขนส่งสินค้า ระบบการชำระเงิน ตลอดจนการนำไปใช้ร่วมกับระบบงานห้องสมุด เป็นต้น ส่วนประกอบหลักของ RFID แสดงดังรูปที่ 2.1



รูปที่ 2.1 ส่วนประกอบหลักของ RFID

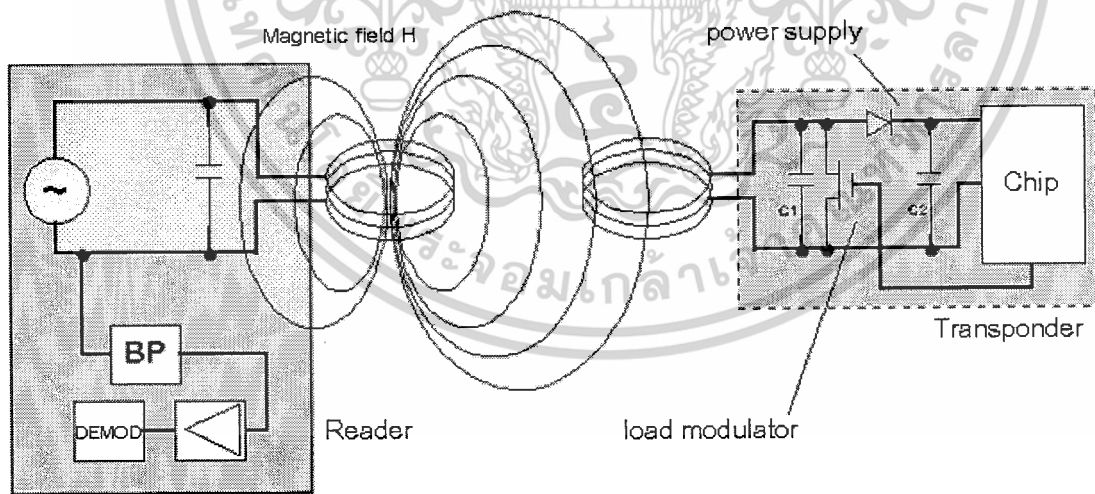
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ส่วนประกอบของระบบ RFID

ระบบ RFID นั้นมีส่วนประกอบขึ้นด้วยกันหลายส่วน ได้แก่ Tag, เครื่องอ่าน Tag (Tag Reader), Middleware และซอฟต์แวร์ที่ใช้ในการพัฒนา

จุดประสงค์ของการใช้ระบบ RFID ก็เพื่อให้เราสามารถรับส่งข้อมูลระหว่างชิ้นส่วนที่มีการเคลื่อนย้ายตำแหน่ง ซึ่งก็คือ Tag ที่จะถูกอ่านโดยเครื่องอ่าน RFID และทำการประมวลผลข้อมูลที่อ่านได้ตามความต้องการใช้งานแล้ว แต่โปรแกรมที่มีอยู่ ข้อมูลที่รับ-ส่งจาก Tag อาจจะเป็นรหัสแสดงเอกลักษณ์หรือข้อมูลระบุตำแหน่ง หรืออาจจะเป็นข้อมูลเฉพาะของผลิตภัณฑ์ เช่น ราคา สี วันจำหน่าย เป็นต้น การประยุกต์ใช้งาน RFID ในการตรวจสอบข้อมูลเริ่มมีใช้กันในช่วง ค.ศ. 1980 และได้รับความสนใจเพิ่มมากขึ้น เนื่องจากจุดเด่นที่สามารถติดตามวัตถุที่มีการโยกย้ายตำแหน่งได้นั้นเอง

ในระบบ RFID ที่ใช้กันทั่วไปนั้น จะติด Tag ซึ่งมีลักษณะเป็นแผ่นวงจรรเล็ก ๆ ไว้กับวัตถุที่ต้องการใช้งานรับ-ส่งข้อมูล ภายใน Tag จะมี Transponder และชิปหน่วยความจำขนาดเล็กที่จะบันทึกรหัสเอกลักษณ์ประจำตัวนั้นไว้ แล้วจะมีเครื่องอ่านสัญญาณ ที่เรียกว่า Interrogator ที่มีเสาอากาศและตัวถอดรหัสอยู่ภายใน เครื่องอ่านสัญญาณนี้จะส่งสัญญาณออกไปยัง RFID Tag ที่อยู่ในบริเวณที่รับคลื่นได้ และสามารถเขียนหรืออ่านข้อมูลลงใน Tag เหล่านั้นได้ เมื่อนำ Tag มาผ่านบริเวณที่มีสัญญาณแม่เหล็กไฟฟ้าจากเครื่องอ่านสัญญาณ เครื่องอ่านจะกระตุ้นให้ Tag ส่งสัญญาณ แล้วตรวจจับสัญญาณจาก Tag จากนั้นก็จะถอดรหัสที่อยู่ภายใน Tag เพื่อนำไปประมวลผลในโปรแกรมที่กำหนดไว้ต่อไป แสดงการใช้งานดังรูปที่ 2.2



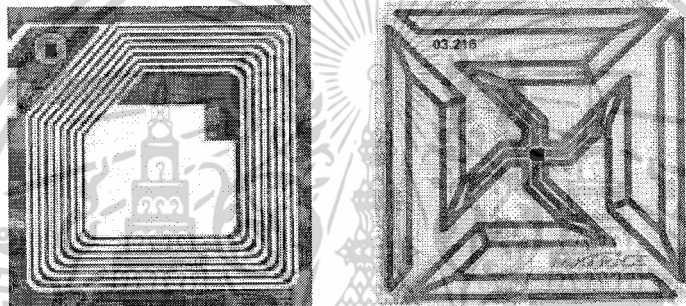
รูปที่ 2.2 การใช้ Tag กับ เครื่องอ่าน Tag

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 ประเภทของ RFID Tag

RFID Tag แบ่งได้เป็นสองประเภท คือ Passive RFID Tag และ Active RFID Tag

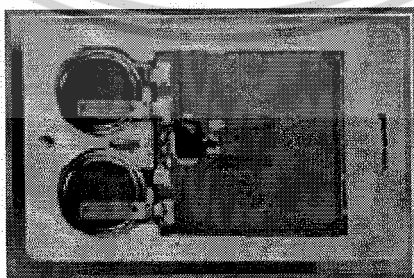
- **Passive RFID Tag** ไม่มีแหล่งกำเนิดไฟฟ้าในตัวเอง กระแสไฟฟ้าที่อยู่ในเสาอากาศของ Tag นั้นเกิดจากสัญญาณวิทยุที่ได้รับมาจากเครื่องส่ง ซึ่งมีกำลังไฟฟ้ามกพอที่จะให้ Tag ใช้ส่งสัญญาณตอบสนองกลับไปยังเครื่องอ่าน RFID ได้ และเนื่องจากการมีพลังงานไฟฟ้าอยู่จำกัด สัญญาณตอบสนองของ Tag ประเภทนี้จึงเป็นข้อมูลสั้นๆ ซึ่งโดยทั่วไปอาจจะเป็นเพียงหมายเลขที่ใช้ระบุเอกลักษณ์ (GUID) เท่านั้น ด้วยเหตุที่ Tag ประเภทนี้ไม่จำเป็นต้องใช้แหล่งพลังงานไฟฟ้า ทำให้มีขนาดเล็กมาก ขนาดที่เล็กที่สุดที่ใช้งานอยู่ในตอนนี้ มีขนาดเพียง 0.4 x 0.4 มิลลิเมตร เท่านั้น ซึ่งบางกว่าแผ่นกระดาษ และบางประเภทก็ไม่อาจมองเห็นได้ด้วยตาเปล่า ในการใช้งานนั้น Passive Tag สามารถรับส่งสัญญาณได้ในระยะตั้งแต่ 10 มิลลิเมตร จนถึง 6 เมตร แสดงดังรูป 2.3



รูปที่ 2.3 RFID Tag แบบ Passive

ภาพ RFID Tag แบบ Passive ที่มีวงจรมินิชิปทำหน้าที่เป็นเสาอากาศรับส่งสัญญาณวิทยุ ซึ่ง Tag นี้จะมีขนาดเล็กและบางมาก สามารถนำไปติดกับสิ่งต่างๆ เพื่อใช้ในการระบุเอกลักษณ์ และเก็บข้อมูลของสิ่งนั้นไว้ได้

- **Active RFID Tag** ต้องใช้แหล่งกำเนิดไฟฟ้าที่ติดตั้งอยู่ภายในเพื่อการทำงาน และสามารถรับส่งสัญญาณที่มีข้อมูลจำนวนมากได้ในระยะทางที่ไกลกว่าแบบ Passive ปัจจุบันนี้ Active Tag ที่มีขนาดเล็กที่สุดมีขนาดประมาณเหรียญบาท สามารถรับส่งข้อมูลได้ในระยะห่างหลายสิบเมตร และมีแบตเตอรี่ที่สามารถใช้งานได้นานหลายปี แสดงดังรูป 2.4

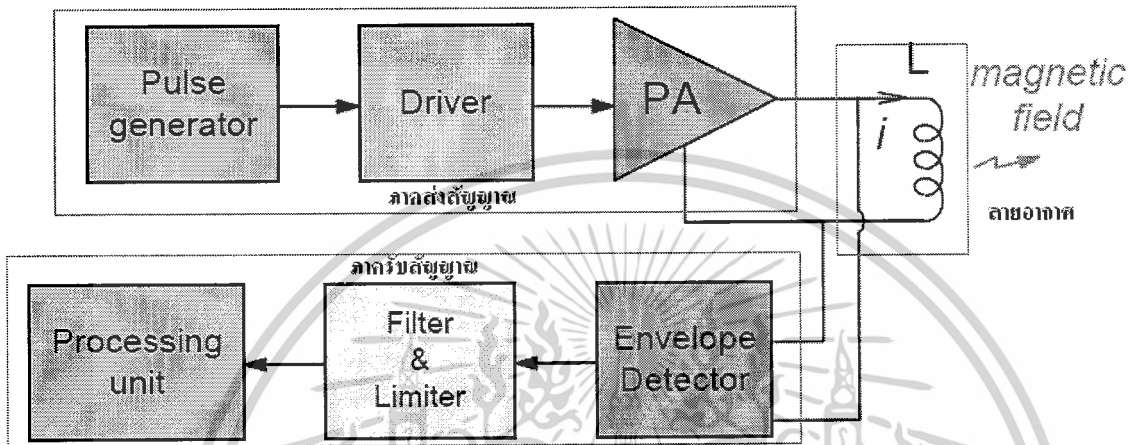


รูปที่ 2.4 RFID Tag แบบ Active

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 เครื่องอ่าน (Reader)

หน้าที่ของเครื่องอ่านก็คือ การเชื่อมต่อเพื่อเขียนหรืออ่านข้อมูลลงในแท็กด้วยความถี่วิทยุ ภายในเครื่องอ่านจะประกอบด้วยเสาอากาศที่ทำจากลวดทองแดงเพื่อใช้รับ-ส่งสัญญาณภาครับและภาคส่ง สัญญาณวิทยุและวงจรควบคุมการอ่านหรือเขียนข้อมูล จำพวกไมโครคอนโทรลเลอร์และส่วนของการติดต่อกับคอมพิวเตอร์ ดังรูปที่ 2.5



รูปที่ 2.5 โครงสร้างภายในของ Reader

โดยทั่วไปเครื่องอ่านจะประกอบด้วยส่วนประกอบหลักดังนี้

- ภาครับและส่งสัญญาณวิทยุ
- ภาคสร้างสัญญาณพาหะ
- ขดลวดที่ทำหน้าที่เป็นสายอากาศ
- ส่วนประมวลผลข้อมูล และภาคติดต่อกับคอมพิวเตอร์

หน่วยประมวลผลข้อมูลที่อยู่ภายในเครื่องอ่านมักใช้เป็นไมโครคอนโทรลเลอร์ ซึ่งอัลกอริทึมที่อยู่ภายในโปรแกรมจะทำหน้าที่ถอดรหัสข้อมูล (Decoding) ที่ได้รับและทำหน้าที่ติดต่อกับคอมพิวเตอร์ ลักษณะขนาดและรูปร่างของเครื่องอ่านจะแตกต่างกันไปตามประเภทของการใช้งาน เช่น แบบมือถือขนาดเล็กหรือติดผนัง จนถึงขนาดใหญ่เท่าประตู เป็นต้น

Reader หรือเครื่องอ่าน ทำหน้าที่ในการส่งคลื่นความถี่วิทยุ เพื่อไปกระตุ้นให้ Tag ทำงานและทำการอ่านข้อมูลเลขรหัสต่าง ๆ ตัวอย่างของเครื่อง RFID Reader แสดงดังรูปที่ 2.6



รูปที่ 2.6 RFID Reader

2.2.5 คุณสมบัติของระบบ RFID

- อ่าน/เขียนโดยไม่ต้องสัมผัส จุดเด่นข้อแรกของระบบอาร์เอฟไอดี คือเครื่องอ่านกับป้ายข้อมูลอิเล็กทรอนิกส์สามารถสื่อสารกันได้โดยไม่ต้องสัมผัสทำให้ไม่เกิดส่วนของการสึกหรอเหมือนการ์ดแถบแม่เหล็ก ทำให้ต้นทุนในการดูแลรักษาต่ำ อายุการใช้งานยาวนานสะดวกรวดเร็วในการใช้งาน

- ทนต่อสภาพแวดล้อมและสิ่งสกปรก ปัญหาที่เป็นอุปสรรคในการอ่าน/เขียนข้อมูลในระบบบ่งชี้อัตโนมัติ ที่แก้ไขลำบากก็คือ สภาพแวดล้อมในการใช้งาน เช่น ในโรงงานอุตสาหกรรมมีทั้งฝุ่นละออง น้ำมัน ระบบบ่งชี้อัตโนมัติที่มีปัญหามากที่สุดคือระบบบาร์โค้ดเพราะถ้าแถบบาร์โค้ดสกปรกหรือฉีกขาดก็ไม่สามารถอ่านข้อมูลได้หรือถ้าหน้าจอของตัวอ่านสกปรกก็มีปัญหาในการอ่านอีกเช่นกัน แต่ด้วยลักษณะเทคโนโลยีของอาร์เอฟไอดี ที่ใช้คลื่นความถี่วิทยุเป็นพาหะนำข้อมูลไป พบว่า ปัญหาดังกล่าวไม่มีผลกระทบต่อระบบอาร์เอฟไอดี เลย 100% ดังนั้นอาร์เอฟไอดี จึงเป็นอุปกรณ์ ระบบบ่งชี้อัตโนมัติ ที่เหมาะสมที่สุดสำหรับการใช้งานในโรงงานอุตสาหกรรม

- สามารถอ่าน/เขียนข้อมูลได้สะดวก มีระบบบ่งชี้อัตโนมัติ น้อยชนิดที่สามารถอ่าน/เขียนข้อมูลได้สะดวกหรือบางระบบต้องใช้เครื่องอ่าน/เขียนแยกกันต่างหาก เช่น บาร์โค้ด ต้องมีเครื่องพิมพ์และเครื่องอ่านแยกกัน สามารถการ์ดต้องนำป้ายข้อมูลอิเล็กทรอนิกส์มาสัมผัสกับวงจรอ่าน/เขียนโดยตรง แต่ระบบอาร์เอฟไอดีตัวอ่านกับตัวเขียนข้อมูล อยู่ในตัวเดียวกันเพียงเปลี่ยนโหมดโดยใช้ซอฟต์แวร์เท่านั้น จึงเหมาะสำหรับงานที่ต้องอ่านและ เปลี่ยนแปลงข้อมูลอยู่ตลอดเวลาเช่น สายการผลิตอัตโนมัติ

- สื่อสารได้ทุกทิศทาง เนื่องจากคุณสมบัติของคลื่นแม่เหล็กไฟฟ้าการอ่าน/เขียนในระบบอาร์เอฟไอดี จึงไม่ต้องคำนึงถึงทิศทางว่าป้ายข้อมูลอิเล็กทรอนิกส์ต้องอยู่ตรงหน้ากับเครื่องอ่านเสมอป้ายข้อมูลอิเล็กทรอนิกส์สามารถอยู่ด้านหลัง ด้านข้างหรือแม้กระทั่งถูกทับอยู่ แต่ถ้าเข้ามาอยู่ในพื้นที่สัญญาณแล้วสามารถอ่าน/เขียนข้อมูลได้ตามปกติ

- ป้ายข้อมูลอิเล็กทรอนิกส์ สามารถนำกลับมาใช้ใหม่ได้ด้วยลักษณะโครงสร้างและความสามารถในการเขียนข้อมูลซ้ำได้ทำให้ป้ายข้อมูลอิเล็กทรอนิกส์สามารถนำกลับมาใช้ในกระบวนการผลิตได้มากกว่า 100,000 ครั้งต่อ 1 ป้ายข้อมูลอิเล็กทรอนิกส์คุณสมบัติข้อนี้เป็นจุดแข็งอีกจุดหนึ่งที่ระบบบ่งชี้อัตโนมัติอื่นไม่สามารถทำได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **ป้ายข้อมูลอิเล็กทรอนิกส์** มีหลากหลายแบบให้ประยุกต์ใช้งาน ป้ายข้อมูลอิเล็กทรอนิกส์ของระบบอาร์เอฟไอดี นั้นถูกออกแบบให้มีรูปร่าง ขนาดโครงสร้างความจุของหน่วยความจำและลักษณะการใช้งานที่แตกต่างกันออกไปเช่น มีลักษณะเป็น สมาร์ทการ์ด กระดุม เหรียญ ทรงสี่เหลี่ยม หรือแม้กระทั่งเป็นแผ่นบาง ๆ เพื่อให้ผู้ใช้งานสามารถเลือกใช้ตามความต้องการ

- **ความสามารถในการทะลุทะลวงของสัญญาณ** คลื่นแม่เหล็กไฟฟ้าสามารถทะลุผ่านวัตถุที่ไม่ใช่โลหะหรือมีโลหะเป็นส่วนผสมอยู่ ได้เช่น พลาสติก ผิวหนัง ไม้ ปูนซีเมนต์ ฯลฯ ดังนั้นป้ายข้อมูลอิเล็กทรอนิกส์จึงสามารถติดตั้งแบบฝัง หรือซ่อนลงไปในพื้นที่วัตถุที่เราต้องการได้ เช่น เราพบเห็นการฉีดอาร์เอฟไอดี ที่มีลักษณะเป็นแท่งแก้วเล็กๆ เข้าไปในตัวสัตว์ การป้ายข้อมูลอิเล็กทรอนิกส์ลงบนพื้นในระบบ AGV (Automatic Guide Vehicle)

- **สื่อสารได้ระยะไกล** ระยะในการอ่าน/เขียนข้อมูลของระบบอาร์เอฟไอดี นั้นทำได้ตั้งแต่ 0-10 เมตร ซึ่งถือว่าไกลที่สุดในบรรดาระบบบ่งชี้อัตโนมัติที่ใช้งานอยู่ในปัจจุบันนี้ ทั้งนี้ระยะในการอ่าน/เขียนข้อมูลขึ้นอยู่กับกำลังส่งของสายอากาศและช่วงความถี่ที่ใช้งาน สำหรับกำลังส่งของสายอากาศนั้นถูกกำหนดโดยกฎหมายของแต่ละประเทศทำให้ อาร์เอฟไอดีที่ผลิตในบางประเทศมีระยะในการอ่าน/เขียนต่างกันทั้งที่ความถี่ใช้งานเท่ากัน

- **หน่วยความจำขนาดใหญ่** หน่วยความจำที่ใช้ในระบบอาร์เอฟไอดี มีตั้งแต่ขนาด 1 บิต (EAS) จนถึงมากกว่า 8 กิโลไบต์หน่วยความจำที่เป็น RAM สามารถเก็บข้อมูลได้มากกว่าหน่วยความจำแบบอื่น ข้อมูลในกระบวนปฏิบัติงานสามารถบันทึกลงในป้ายข้อมูลอิเล็กทรอนิกส์ได้ทั้งกระบวนการ หรือแม้กระทั่งข้อมูลส่วนบุคคลก็สามารถบันทึกลงในป้ายข้อมูลอิเล็กทรอนิกส์ได้

- **อ่าน/เขียนข้อมูลได้ครั้งละมากกว่า 1 ป้ายข้อมูลอิเล็กทรอนิกส์พร้อมกัน** เมื่อป้ายข้อมูลอิเล็กทรอนิกส์เข้ามาอยู่ในพื้นที่สัญญาณมากกว่า 1 ป้ายข้อมูลอิเล็กทรอนิกส์พร้อมกันเครื่องอ่านสามารถอ่านข้อมูลซึ่งมาพร้อมกันได้ทั้งหมดหรือสามารถเลือกอ่านเฉพาะป้ายข้อมูลอิเล็กทรอนิกส์ที่ระบุก็ได้

- **สามารถอ่าน/เขียนข้อมูลขณะวัตถุกำลังเคลื่อนที่** เครื่องอ่านกับป้ายข้อมูลอิเล็กทรอนิกส์สามารถสื่อสารกันได้แม้ขณะฝ่ายใดฝ่ายหนึ่ง กำลังเคลื่อนที่โดยความเร็วของการเคลื่อนที่ขึ้นอยู่กับชนิดของการสื่อสาร หน่วยความจำและปริมาณข้อมูลที่ใช้อ่าน/เขียน

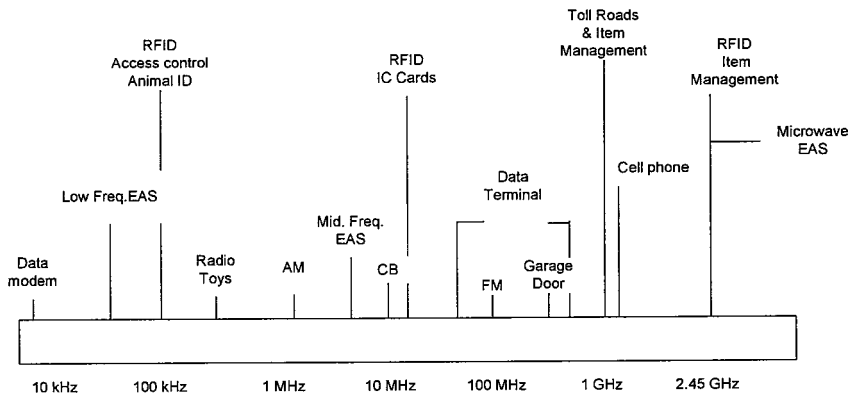
2.2.6 ย่านความถี่สำหรับ RFID

ย่านความถี่สำหรับการใช้งานกัน อาร์เอฟไอดีทั่วโลกอยู่ในย่านความถี่สาธารณะ (Industrial-Scientific-Medical) หรือ ISM Band ซึ่งเป็นย่านความถี่หลายประเทศกำหนดไว้สอดคล้องกันในการอนุญาตให้ใช้งานในเชิงอุตสาหกรรม วิทยาศาสตร์ และการแพทย์ มี 4 ย่านความถี่และสำหรับย่านความถี่ใน ISM Band ที่กำหนดให้ใช้สำหรับ อาร์เอฟไอดีทั้ง 4 ย่านความถี่คือ

- ย่านความถี่ต่ำ (Low Frequency: LF) ต่ำกว่า 150 KHz
- ย่านความถี่สูง (High Frequency: HF) 13.56/27.125 MHz
- ย่านความถี่สูงยิ่ง (Ultra High Frequency: UHF) 433/868/915 MHz
- ย่านความถี่ไมโครเวฟ (Microwave frequency) 2.45/5.8 GHz

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการเรียนการสอน ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ย่านความถี่ ISM Band ที่นำระบบอาร์เอฟไอดีในปัจจุบัน

จากรูปที่ 2.7 ในการใช้งาน 2 ย่านความถี่แรก เหมาะสำหรับการใช้กับงานที่มีระยะการสื่อสารข้อมูลในระยะใกล้ (LF ระยะอ่านประมาณ 10-20 เซนติเมตร และ HF ระยะอ่านประมาณ 15-100 เซนติเมตร) เช่น การตรวจสอบการผ่านเข้าออกพื้นที่หรือรถไฟฟ้า เป็นต้น การเก็บประวัติในสัตว์ ส่วนย่านความถี่ UHF ถูกใช้กับงานที่มีระยะการสื่อสารข้อมูลในระยะไกล (UHF ระยะอ่านประมาณ 1-10 เมตร) เช่น ระบบเก็บค่าบริการทางด่วน ระบบขนส่งสินค้า และในปัจจุบันระบบ อาร์เอฟไอดี กำลังถูกวิจัยและพัฒนาในย่านความถี่ไมโครเวฟที่ความถี่ 2.4 GHz และความถี่ 5.8 GHz เพื่อใช้งานในระบบคลังสินค้าเนื่องจากขนาดของสายอากาศที่เล็กมาก ในด้านของราคาและความเร็วในการสื่อสารข้อมูล เมื่อเทียบกับแล้วอาร์เอฟไอดีซึ่งใช้คลื่นพาหะย่านความถี่สูงเป็นระบบที่มีความเร็วในการส่งข้อมูลสูงและมีราคาสูงด้วยเช่นกัน

2.3 ไมโครคอนโทรลเลอร์พีไอซี (PIC)

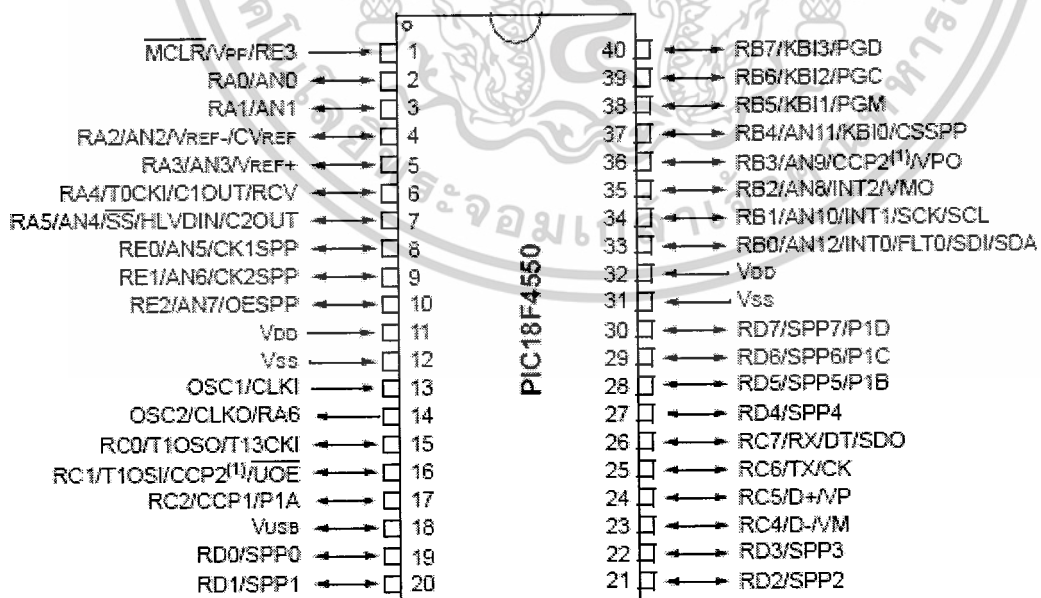
PIC เป็น ไมโครคอนโทรลเลอร์อีกตระกูลหนึ่ง ย่อมาจากคำว่า Peripheral Interface Controller โดยไมโครคอนโทรลเลอร์ตระกูลนี้จะพยายามรวมเอาทุกอย่างเอาไว้ในตัว ไม่ว่าจะเป็น PROGRAM MEMROY, RAM, EEPROM, SERIAL, I2C, PWM, A/D ไม่จำเป็นต้องต่ออุปกรณ์เสริมจากภายนอก ในตัวของ PIC จะมีฟังก์ชันที่ใช้ในการประมวลผล รวมทั้งหน่วยความจำที่ทำให้มันเหมือนกัน CPU ตัวหนึ่ง โดยโครงสร้างของไมโครคอนโทรลเลอร์ PIC เบอร์ 18F4550 ที่ใช้ในโครงงานนี้ แสดงดังรูปที่ 2.8 ซึ่งคุณสมบัติของไมโครคอนโทรลเลอร์ PIC เบอร์ 18F4550 แสดงดังตารางที่ 2.1

ตารางที่ 2.1 คุณสมบัติไมโครคอนโทรลเลอร์ PIC18F4550

คุณสมบัติ	PIC18F4550
Operating Frequency	DC – 48 MHz
Program Memory (Bytes)	32768
Data Memory (Bytes)	2048
Data EEPROM Memory (Bytes)	256
Interrupt Sources	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

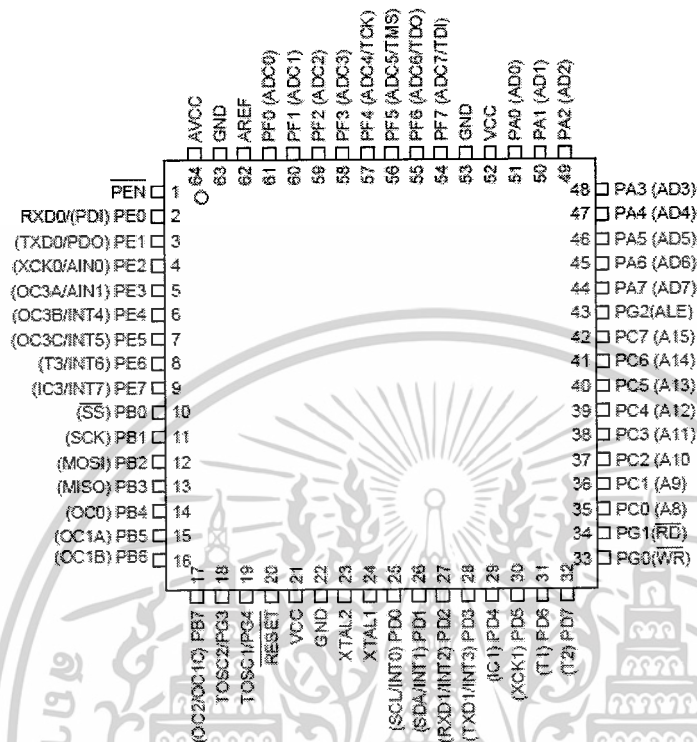
I/O Ports	Ports A, B, C, D, E
Timers	4
Capture/Compare/PWM Modules	1
Enhanced Capture/Compare/PWM Modules	1
Universal Serial Bus (USB) Module	1
Serial Communications	MSSP, Enhanced USART
Streaming Parallel Port (SPP)	Yes
10-bit Analog-to-Digital Module	13 Input Channels
Resets (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT
Programmable High/Low-Voltage Detect	Yes
Programmable Brown-out Reset	Yes
Instruction Set	75 Instructions; 83 with Extended Instruction Set enabled
Packages	40-pin PDIP 44-pin QFN 44-pin TQFP



รูปที่ 2.8 โครงสร้างลักษณะของไมโครคอนโทรลเลอร์ PIC เบอร์ 18F4550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ไมโครคอนโทรลเลอร์เอวีอาร์ (AVR)



รูปที่ 2.9 โครงสร้างลักษณะของไมโครคอนโทรลเลอร์ AVR

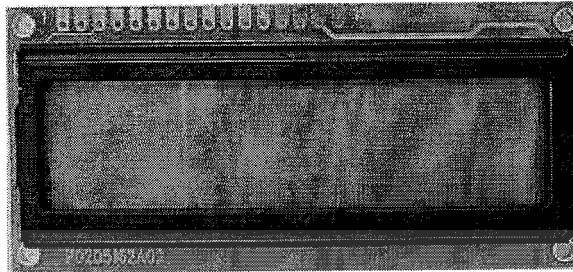
2.4.1 คุณสมบัติของไมโครคอนโทรลเลอร์ AVR

- มีหน่วยความจำแบบ FLASH ขนาด 64 KBytes
- มีหน่วยความจำข้อมูลถาวรแบบ EEPROM ขนาด 2 KByte
- มีจำนวน I/O สูงสุดถึง 53 I/O Pins
- มีวงจรสื่อสาร SPI จำนวน 1 ช่อง, I2C จำนวน 1 ช่อง
- มีวงจรสื่อสาร Programmable Serial USARTS จำนวน 2 ช่อง
- มีระบบการเปลี่ยนสัญญาณ ANALOG TO DIGITAL ขนาด 10 บิต จำนวน 8 ช่อง
- มี Timer/Counters 8 บิต จำนวน 2 ช่อง, Timer/Counter 16 บิต จำนวน 2 ช่อง, Watchdog Timer, Real Time Counter
- มีระบบการกำเนิดความถี่สัญญาณแบบ PWM จำนวน 2 ช่อง
- พอร์ตอินพุตและเอาต์พุตจำนวน 6 พอร์ต ได้แก่ PORT A, PORT B, PORT C, PORT D, PORT E และ PORT F

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 โมดูล (Module) ที่เกี่ยวข้องในการทำโครงการ

2.5.1 โมดูลแอลซีดี (Module LCD)



รูปที่ 2.10 หน้าจอแสดงผล LCD ขนาด 16x2

อุปกรณ์สำหรับแสดงผลในปัจจุบันมีหลายแบบด้วยกัน LCD ก็เป็นอุปกรณ์แสดงผลประเภทหนึ่งที่ได้รับคามนิยมอย่างแพร่หลาย แสดงดังรูปที่ 2.10

2.5.2 ส่วนประกอบหลักของโมดูล LCD

1. ตัวควบคุม (Controller) เป็นอุปกรณ์สำหรับรับข้อมูลที่จะส่งมาจากอุปกรณ์ภายนอกเพื่อควบคุมการทำงานภายในโมดูล LCD เช่น การลบจอภาพ, การแสดงตัวอักษร หรือการเลื่อนเคอร์เซอร์ เป็นต้น
2. ตัวขับ (Driver) เป็นอุปกรณ์รับข้อมูลจากตัวควบคุม (Controller) เพื่อขับให้ตัวแสดงผลแสดงข้อมูลตามที่กำหนด
3. ตัวแสดงผล (Dot Matrix Display) เป็นอุปกรณ์แสดงผลให้สามารถมองเห็นเป็นตัวอักษรหรืออักขระ ซึ่งภายในจุดแสดงผลจะเป็นผลึกเหลวที่สามารถแสดงผลให้เห็น โดยอาศัยการเปิดและปิดตัวเองกับแสงจากภายนอก

ตารางที่ 2.2 รายละเอียดต่างๆของ โมดูล LCD

ขาที่	สัญญาณ	รายละเอียด
1	Vss	0 V Ground
2	Vcc	+ 5V
3	Vee	ใช้ปรับความสว่างของ LCD ถ้าต่อลงดินจะสว่างที่สุด
4	RS	สัญญาณ Register Select ใช้เลือกรีจิสเตอร์ควบคุมหรือหน่วยความจำแสดงผล ถ้าเป็น "0" แสดงว่าต้องการติดต่อกับรีจิสเตอร์ควบคุม ถ้าเป็น "1" แสดงว่าต้องการติดต่อกับรีจิสเตอร์แสดงผล
5	R/W	สัญญาณควบคุมการอ่าน/เขียน ถ้าเป็น "0" " แสดงว่าต้องการเขียนหรือส่งข้อมูล ถ้าเป็น "1" แสดงว่าต้องการอ่านข้อมูลจากโมดูล

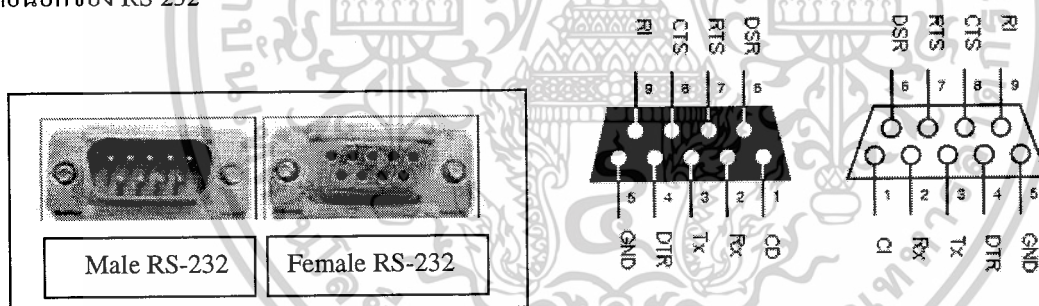
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ห้ามเผยแพร่ไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6	E	Enable – สัญญาณสั่งให้เริ่มต้นการทำงาน สำหรับ การอ่าน/เขียนข้อมูล การรับส่งข้อมูลจะเกิดเมื่อเป็น '1' และขบขาลง
7-10	DB0-DB3	เป็นบัสแบบสองทิศทางใช้สำหรับส่งถ่ายข้อมูลระหว่างซีพียูกับ โมดูลในกรณีทีการทำงานเป็นแบบ 4 บิต บัสนี้ไม่ได้ใช้และควรต่อลงดินด้วย แต่ถ้าเป็น การทำงานแบบ 8 บิต บัสนี้จะเป็น 4 บิตต่ำ ใช้เพื่อการส่งถ่ายข้อมูล
11-14	DB4-DB7	เป็นบัสแบบสองทิศทางใช้สำหรับส่งถ่ายข้อมูลระหว่างซีพียูกับ โมดูลในกรณีทีการทำงานเป็นแบบ 4 บิต จะใช้บัสนี้ส่งถ่ายข้อมูล แต่ถ้าเป็นการทำงานแบบ 8 บิต บัสนี้จะเป็น 4 บิตสูงนอกจากนี้ DB7 ยังใช้เป็นบิตแสดงสถานะ Busy

2.6 พอร์ตอนุกรม RS-232

RS-232 ย่อมาจาก Recommended Standard-232 เป็นมาตรฐานการเชื่อมต่อข้อมูลแบบอนุกรม กำหนดโดย EIA(Electronics Industry Association) หรือสมาคมผู้ประกอบการอุตสาหกรรม อิเล็กทรอนิกส์ของอเมริกา ใช้กับการสื่อสารแบบจุดต่อจุด โดยใช้สายเชื่อมต่อ DB 9 ขา ปกติพอร์ตอนุกรม RS-232 จะสามารถต่อสายได้ยาว 50 ฟุต โดยประมาณขึ้นอยู่กับ ชนิดของ สายสัญญาณ, ระยะทาง, และ ปริมาณสัญญาณรบกวน ดังรูปที่ 2.11 แสดงถึงรายละเอียดของขา RS 232 แบบ 9 ขา และลักษณะภายนอกของ RS 232



รูปที่ 2.11 ลักษณะภายนอกและรายละเอียดขาของ RS-232

พอร์ตอนุกรม RS-232 ที่ใช้อยู่บนเครื่องพีซีโดยทั่วไปนั้นจะมีขาที่ใช้งานอยู่ 9 ขา แต่มีขาที่ทำหน้าที่เป็นอินพุตนั้นมีอยู่ 4 ขา ถ้าตัดขา RX ซึ่งเป็นขาที่รับข้อมูลแล้วจะมีขาที่ใช้งาน 4ขา จะพอดีกับลอจิกอะนาไลเซอร์ (Logic Analyzer) ที่สร้างขึ้นโดยแต่ละ channel จะใช้ตำแหน่งขาต่างๆ ในการรับค่าอินพุตดังนี้คือ แชนแนล 1 ใช้ขา CTS ขา8 channel 2 ใช้ขา DSR ขา6 channel 3 ใช้ขา RI ขา 9 channel 4 ใช้ขา DCD ขา1 และขาที่เหลือต่างๆจะมีหน้าที่ซึ่งมีรายละเอียดในตารางที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 รายละเอียดขาของ RS-232

ขาที่ใช้งาน	ทิศทาง	หน้าที่
1	อินพุต	Data carrier Detect(DCD)
2	อินพุต	Receive Data(RX)
3	เอาต์พุต	Transmit Data(TX)
4	เอาต์พุต	Data terminal Ready(DTR)
5	กราวด์	Signal Ground
6	อินพุต	Data set Ready(DSR)
7	เอาต์พุต	Request to sent(RTS)
8	อินพุต	Clear to Sent(CTS)
9	อินพุต	Ring Indicator(RI)

2.6.1 การเชื่อมต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ด้วยสาย DB9

การทำงานของขาสัญญาณ DB9

TXD เป็นขาที่ใช้ส่งข้อมูล

RXD เป็นขาที่ใช้รับข้อมูล

DTR แสดงสถานะพอร์ตที่เปิดใช้งาน, **DSR** ตรวจสอบว่าพอร์ต ที่ติดต่อกับ เปิดอยู่หรือไม่ เมื่อเปิดพอร์ตอนุกรม ขา DTR จะ “ON” เพื่อให้อุปกรณ์ได้รับทราบว่าการติดต่อดำเนินการในขณะเดียวกันก็จะตรวจสอบขา DSR ว่าอุปกรณ์พร้อมหรือไม่

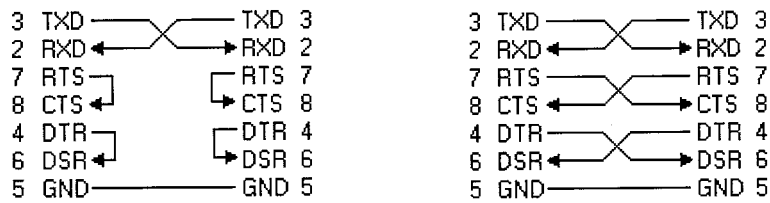
RTS แสดงสถานะพอร์ตที่ต้องการส่งข้อมูล, **CTS** ตรวจสอบว่าพอร์ตที่ติดต่อกับ ต้องการส่งข้อมูลหรือไม่

เมื่อต้องการส่งข้อมูลขา RTS จะ “ON “และจะส่งข้อมูลออกที่ขา TXD เมื่อส่งเสร็จก็จะ OFF ในขณะเดียวกันก็จะตรวจสอบขา CTS ว่าอุปกรณ์ที่ต้องการที่จะส่งข้อมูลหรือไม่

GND ขา ground

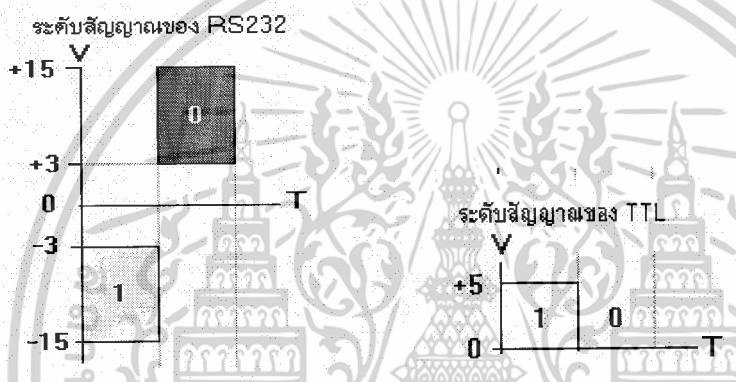
รูปที่ 2.12 เป็นรูปที่แสดงตัวอย่างการเชื่อมต่ออุปกรณ์ภายนอกผ่าน DB 9 ในแบบ Null Modem และการเชื่อมต่อแบบ 3 เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 การเชื่อมต่ออุปกรณ์ภายนอกผ่าน DB 9 แบบ 3 เส้นและการเชื่อมต่อแบบ Null Modem ตามลำดับ

2.6.2 ระดับสัญญาณของ RS232



รูปที่ 2.13 ระดับสัญญาณของ RS232C และระดับสัญญาณของ TTL

สัญญาณรบกวนที่เกิดขึ้นในสายนำสัญญาณมักจะมีแรงดันเป็นบวก เมื่อเทียบกับกราวด์ (Ground) เพื่อป้องกันสัญญาณรบกวนนี้จึงออกแบบแรงดันของโลจิก "1" เป็นลบ คืออยู่ในช่วง -3V ถึง -15V ส่วนแรงดันของโลจิก "0" อยู่ในช่วง +3V ถึง +15V และเหตุที่ระดับสัญญาณของ RS232 อยู่ในช่วง +15V ถึง -15V ก็เพื่อให้ต่อสายสัญญาณไปได้ไกลขึ้น ดังนั้นจึงจำเป็นต้องมีวงจรเปลี่ยนระดับแรงดันของ RS232 มาเป็นระดับแรงดันของ TTL โดยที่ระดับสัญญาณ RS 232 และ ระดับสัญญาณ TTL ดังรูปที่ 2.13

2.6.3 อัตราการส่งข้อมูล (Baud rate)

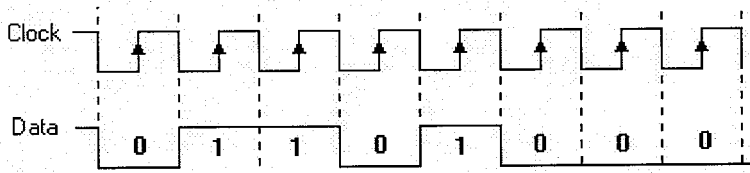
คือความเร็วของการรับ-ส่งข้อมูล เป็นจำนวนบิตต่อวินาที เช่น 300, 1200, 2400, 4800, 9600, 14400, 19200, 38400, 56000 เป็นต้น การเลือกอัตราการส่งข้อมูลขึ้นอยู่กับ ชนิดของสายสัญญาณ, ระยะทางและปริมาณสัญญาณรบกวน

2.6.4 รูปแบบการสื่อสารแบบอนุกรม

มีด้วยกันอยู่ 2 แบบ คือแบบซิงโครนัส (Synchronous) และ แบบอะซิงโครนัส (Asynchronous)

เอกสารนี้เป็นลิขสิทธิ์สงวนลิขสิทธิ์ของ บริษัท อักษรพาณิชย์ จำกัด (มหาชน) การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับส่งข้อมูลจะมีสัญญาณนาฬิกา ซึ่งเป็นตัวกำหนดจังหวะเวลา การส่งข้อมูล ร่วมอยู่ด้วยอีกเส้นหนึ่ง ใช้คู่กับสัญญาณข้อมูล ตัวอย่างเช่น การส่งสัญญาณจากคีย์บอร์ด แสดงดังรูปที่ 2.14

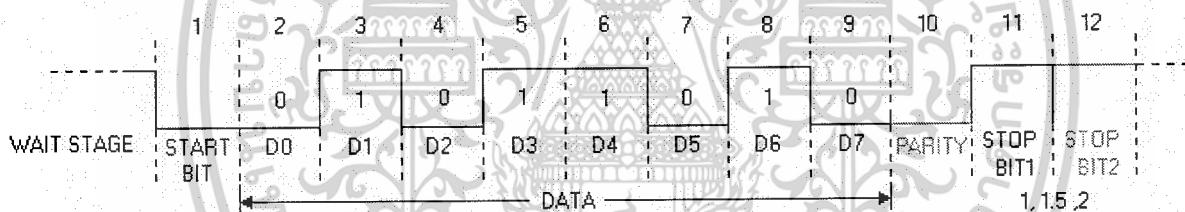


รูปที่ 2.14 การสื่อสารแบบซิงโครนัส (Synchronous)

2. การสื่อสารแบบอะซิงโครนัส (Asynchronous)

การรับส่งข้อมูล โดยที่ไม่จำเป็นต้อง มีสัญญาณนาฬิกา ร่วมด้วย แต่จะใช้ให้ตัวส่ง และตัวรับ มีอัตราส่งข้อมูลที่เท่ากัน แสดงดังรูปที่ 2.15 โดยมีรูปแบบข้อมูลแบบอะซิงโครนัส ประกอบด้วย 4 ส่วนคือ

1. บิตเริ่มต้น (Start bit) มีขนาด 1 บิต
2. บิตข้อมูล (Data) มีขนาด 5,6,7 หรือ 8 บิต
3. บิตตรวจสอบพาริตี (Parity bit) มีขนาด 1 บิตหรือไม่มี
4. บิตหยุด (Stop bit) มีขนาด 1, 1.5, 2 บิต



รูปที่ 2.15 การสื่อสารแบบอะซิงโครนัส (Asynchronous)

การรับและส่งข้อมูลแบบอนุกรมยังแบ่งออกเป็นลักษณะการใช้งานได้ 3 แบบคือ

1. แบบซิมเพลกซ์ (Simplex) เป็นการส่ง หรือรับข้อมูล แบบทิศทางเดียว เท่านั้น
2. แบบฮาล์ฟดูเพลกซ์ (Half Duplex) เป็นการส่งและรับข้อมูลแบบสลับกัน คือเมื่อด้านหนึ่งส่ง อีกด้าน หนึ่ง เป็นฝ่ายรับ สลับกัน ไม่สามารถรับ-ส่งในเวลาเดียวกันได้
3. แบบฟูลดูเพลกซ์ (Full Duplex) สามารถรับ-ส่งข้อมูลในเวลาเดียวกันได้

2.7 ระบบ GSM

GSM ย่อมาจาก Global System for Mobile นั้นเป็นระบบเครือข่ายโทรศัพท์เคลื่อนที่แบบเซลลูลาร์ ชนิดดิจิทัลเซลลูลาร์ (Digital Cellular) ซึ่งโทรศัพท์เคลื่อนที่แบบเซลลูลาร์ หรือ Cellular Network หรือ Cellular System นั้น หมายถึงโทรศัพท์เคลื่อนที่ ที่มีการติดต่อกับเครือข่ายซึ่งจำแนกพื้นที่การใช้งานโดยแบ่งเป็นเซลล์ (cell) ถ้าจะเปรียบเทียบกับลักษณะการใช้งาน กับระบบ อนุาลอกเซลลูลาร์ ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Analog Cellular) แล้วมีข้อดีที่ต่างกันมาก เช่น ความปลอดภัยจากการดักฟัง และด้านการโทรข้ามประเทศ หรือ International Roaming เพราะเนื่องจาก มาตรฐาน โทรศัพท์เคลื่อนที่แบบ อนาล็อก ได้ออกมาตรฐาน จากหลายประเทศ และไม่สามารถใช้งานร่วมกันได้ ทำให้โทรข้ามเครือข่าย ไปยังประเทศอื่นไม่ได้ เป็นต้น

2.7.1 โครงสร้างของระบบเครือข่าย GSM

โครงสร้างของระบบ GSM นั้นประกอบไปด้วยส่วนประกอบที่สำคัญดังนี้

-อุปกรณ์ลูกข่าย (ME)

นับเป็นอุปกรณ์ที่ผู้คนที่ส่วนมากมีความคุ้นเคยด้วยมากที่สุดสำหรับชื่อเรียกเป็นทางการของเครื่องลูกข่าย คือ Mobile Station Terminal Equipment ซึ่งสามารถแบ่งออกได้เป็นหลายประเภทตามกำลังส่งสูงสุดที่เครื่องลูกข่ายสามารถส่งได้

-SIM CARD

SIM เป็นคำย่อจากคำว่า Sub scriber Identity Module มีลักษณะเป็นแผ่นการ์ดขนาดเท่ากับบัตรเครดิต ภายในมีอุปกรณ์ไมโครเซสเซอร์และหน่วยความจำขนาดเล็กฝังอยู่ ในการใช้งานจะต้องเสียบแผ่น SIM ลงในช่อง สำหรับใส่ SIM ภายในเครื่องลูกข่าย เพื่อให้เครื่องลูกข่ายสามารถอ่านหรือเขียนข้อมูลลงใน SIM ได้จึงจะสามารถใช้เครื่องลูกข่ายดังกล่าวได้ภายในหน่วยความจำของ SIM แต่ละใบจะบรรจุข้อมูลจำเพาะประจำตัวผู้ใช้บริการซึ่งเป็นเจ้าของ SIM นั้นๆ รวมถึงข้อมูลส่วนตัว เช่น เลขหมาย พร้อมทั้งชื่อของผู้ที่ต้องติดต่อด้วยประจำ ข้อมูลจำเพาะภายใน SIM แต่ละใบจะถูกใช้สำหรับแสดงตัวต่อระบบเครือข่าย

-สถานีฐาน (BTS : Base Transceiver Station)

ทำหน้าที่เป็นศูนย์กลางในการติดต่อสื่อสารกับเครื่องลูกข่ายและเครือข่ายโทรศัพท์เคลื่อนที่ โดยใช้คลื่นวิทยุสำหรับการติดต่อสื่อสาร โดยปกติแล้ว BTS จะมีตำแหน่งอยู่ที่ภายในกึ่งกลางของเซลล์ในกรณีของเซลล์แบบ Omni-directional

-ส่วนควบคุมสถานีฐาน (Base Station Controller)

ส่วนควบคุมสถานีฐานจะมีการเชื่อมต่อกับสถานีฐาน (BTS) และ ชุมสายโทรศัพท์เคลื่อนที่ (BSC and MSC) เพื่อทำหน้าที่ในการควบคุมสถานีฐาน โดยปกติส่วนควบคุมสถานีฐานสามารถที่จะรองรับ สถานีฐานได้ 10 ถึง 100 สถานี ทำหน้าที่จัดสรรช่องสัญญาณ, วัเคราะห์การเรียกเข้าของเครื่อง โมบายล์ และควบคุมการแฮนด์ออฟระหว่างสถานีฐานกับสถานีฐาน

-ชุมสายโทรศัพท์เคลื่อนที่ (Mobile Switching Center or Mobile Telephone Switching Office)

ทำหน้าที่ในการเชื่อมต่อวงจรระหว่างเครื่องลูกข่ายด้วยกัน เพื่อสร้างวงจรสนทนาและเป็นตัวกลางในการติดต่อสื่อสารระหว่างเครื่องลูกข่ายและอุปกรณ์อื่นๆ ภายในเครือข่ายชุมสายโทรศัพท์

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เคลื่อนที่มีชื่อเรียกว่า Mobile Services Switching Center หรือ MSC นอกจาก MSC จะเชื่อมต่อกับอุปกรณ์ต่างๆ ภายในเครือข่าย แล้วยังจะต้องมีการเชื่อมต่อกับเครือข่ายโทรศัพท์สาธารณะ (PSTN) เครือข่าย ISDN รวมถึงเครือข่ายโทรศัพท์เคลื่อนที่อื่นๆ กรณีที่เครือข่ายมี MSC หลายๆ แห่งก็จะต้องมีการเชื่อมต่อกับ MSC แต่ละแห่งเข้าด้วยกัน

-ฐานข้อมูลผู้ใช้บริการท้องถิ่น (VLR : Visitors Location Register)

ทำหน้าที่เป็นระบบข้อมูลชั่วคราวของชุมสายโทรศัพท์เคลื่อนที่สำหรับเมื่อผู้ใช้โทรศัพท์เคลื่อนที่เข้ามาอยู่ในเขตการให้บริการของ MSC นั้นๆ โดยทั่วไป VLR จะอยู่ภายใน MSC

-ฐานข้อมูลผู้ใช้บริการกลาง (HLR : Home Location Register)

ทำหน้าที่เป็นระบบฐานข้อมูลของชุมสายโทรศัพท์เคลื่อนที่ สำหรับผู้ใช้โทรศัพท์เคลื่อนที่ที่จดทะเบียนในพื้นที่ชุมสายนั้นๆ ทั้งยังเป็น AUC (Authentication Center) ซึ่งทำหน้าที่ตรวจสอบความถูกต้องของเครื่องลูกข่ายว่าเป็นเครื่องโทรศัพท์ผ่านการอนุมัติจากบริษัทหรือไม่

-อุปกรณ์ลูกข่าย (GSM Module)

ทำหน้าที่ในการเชื่อมต่อระบบเครือข่ายเคลื่อนที่ในย่านความถี่ของระบบเครือข่าย GSM เหมือนกับโทรศัพท์เคลื่อนที่ทั่วไป โดยตัว GSM Module นั้นจะมีคุณสมบัติพิเศษกว่าโทรศัพท์เคลื่อนที่ทั่วไปโดยจะทำการสนับสนุนการใช้งานชุดคำสั่งในการควบคุม (AT Command) ได้มากกว่าโทรศัพท์เคลื่อนที่ทั่วไป ซึ่งมีข้อจำกัดในการใช้งานฟังก์ชันต่างๆ ซึ่งควบคุมด้วยชุดคำสั่ง AT Command

2.7.2 AT Command

เป็นชุดคำสั่งมาตรฐานที่ใช้ติดต่อสื่อสารกับโทรศัพท์มือถือ โดยส่วนมากมักใช้ในการสื่อสารกับอุปกรณ์สื่อสารต่างๆ เช่น โมเด็มหรืออุปกรณ์ DTE (Data Terminal Equipment) ในชุดคำสั่งพื้นฐานนั้น บริษัท Hayes ได้เป็นผู้ออกแบบคิดค้นเพื่อใช้กับโมเด็มของตนและต่อมาบริษัทผู้ผลิตมือถือยี่ห้อต่างๆ ได้พัฒนามาใช้กับผลิตภัณฑ์ของตนเป็นเหตุให้คำสั่งพิเศษบางคำสั่งไม่เหมือนกันในผลิตภัณฑ์ยี่ห้ออื่นและความสามารถของโทรศัพท์ในบางรุ่นจะไม่รองรับคำสั่งดังกล่าว เนื่องจากไม่ได้มีวงจรส่วนของโมเด็มบรรจุอยู่ภายใน

2.7.2.1 หลักการรับส่ง SMS ของโทรศัพท์มือถือ

SMS ย่อมาจาก Short Message Service เป็นบริการส่งข้อความสั้นๆ จากโทรศัพท์มือถือต้นทางผ่านชุมสายไปยังโทรศัพท์มือถือปลายทาง โดยสามารถส่งได้สูงสุด 160 ตัวอักษรต่อครั้ง ตามข้อกำหนดมาตรฐานขององค์การ ETSI (European Telecommunications Standards Institute)

การส่งข้อความใน Text Mode นั้นจะเป็นการนำข้อความที่ต้องการส่งมาเข้ารหัสก่อน (โดยตัวเครื่องเอง) แล้วจึงส่งข้อมูลในรูปแบบ PDU Mode อีกครั้งหนึ่ง แต่ในบางเครื่องก็ไม่สนับสนุนการส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบ Text Mode ผ่านทาง AT Command แต่หากเป็น PDU Mode จะสามารถส่งได้ เนื่องจากเครื่องจะไม่ต้องทำอาศัยการแปลงข้อมูลอีกชั้น

โดยที่โทรศัพท์มือถือจะมีรูปแบบในการส่งข้อมูลในรูปแบบ SMS ผ่าน AT-Commands 2 รูปแบบคือ Text Mode และ PDU Mode

1. Text Mode เป็นการส่งข้อมูลในรูปแบบของตัวอักษรได้โดยตรง ซึ่งตัวเครื่องส่วนใหญ่ไม่รองรับการส่งข้อมูลรูปแบบนี้ผ่านทาง AT Commands จึงไม่สามารถใช้งานได้สมบูรณ์ เนื่องจากการส่งข้อความใน Text Mode นั้นจะเป็นการนำข้อความที่ต้องการส่งมาเข้ารหัสก่อน (โดยตัวเครื่องเอง) แล้วจึงส่งข้อมูลในรูปแบบ PDU Mode อีกครั้งหนึ่ง แต่ในโทรศัพท์บางเครื่องก็ไม่สนับสนุนการส่งข้อความแบบ Text Mode ผ่านทาง AT Commands แต่หากส่งข้อความแบบ PDU Mode จะสามารถส่งได้ เนื่องจากโทรศัพท์จะไม่ต้องการแปลงข้อมูลอีกชั้นหนึ่ง

2. PDU Mode (Protocol Data Unit) คือ โหมดการทำงานประเภทหนึ่ง ซึ่งจะทำการแปลงรหัสแอสกี (ASCII) ของตัวอักษรแต่ละตัวให้เป็นรหัส PDU ซึ่งรหัส PDU นั้น สามารถนำมาใช้งานได้กับชุดคำสั่ง AT Commands ในการส่งข้อความ SMS และสามารถใช้ได้กับโทรศัพท์มือถือทุกเครื่องที่รับคำสั่ง AT Command ได้

2.7.2.2 กลุ่มคำสั่งในการส่ง SMS

AT+CMGF เป็นคำสั่งที่ใช้เลือกรูปแบบของการส่งข้อความ ซึ่งมี 2 โหมด คือ SMS PDU Mode กับ SMS Text Mode โดย

AT+CMGF = 1 คือ แสดงข้อความในรูปแบบ TEXT

AT+CMGF = 0 คือ แสดงข้อความในรูปแบบ PDU CODE

AT+CMGS เป็นคำสั่งที่ใช้ส่งข้อความ SMS

ในกรณีที่ใช้ Text Mode นั้นให้ใช้รูปแบบคำสั่งเป็น “AT+CMGS=”+เบอร์ผู้รับ” โดยเบอร์ของผู้รับต้องใส่รหัสประเทศหน้าแทนศูนย์ด้วยเสมอ ซึ่งในกรณีที่เป็นประเทศไทยจะใช้รหัสประเทศเป็น “66” ดังนั้นถ้าต้องการส่งข้อความ SMS ให้กับเบอร์ที่ใช้งานอยู่ในประเทศไทย เช่น 089-1234567 ก็จะต้องกำหนดหมายเลขของเบอร์ผู้รับปลายทางเป็น 6689-1234567 แทน ซึ่งในกรณีนี้จะได้รับรหัสเบอร์ผู้รับข้อความเป็น “+66891234567”

ในกรณีที่ใช้ PDU Mode ในการส่งข้อความสั้นนั้น จะใช้คำสั่ง AT+CMGS โดยรูปแบบคำสั่งและพารามิเตอร์ต่างๆมีดังนี้

AT+CMGS=XX : XX = ความยาวของข้อมูลที่จะส่ง

(โดยที่ความยาวของข้อมูลที่จะส่ง = (Header+PDUCode-3)/2)

>0011000A8180866827520000AA05CB66924A04<Ctrl+z>

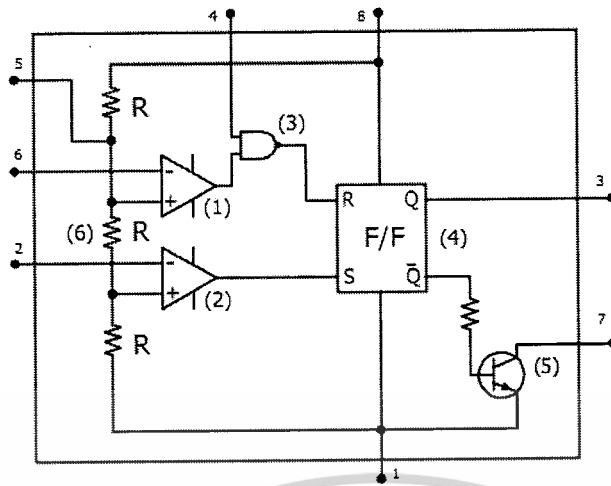
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 00 ความยาวของ SMSC Information โดยที่ 00 หมายถึงให้ใช้ SMSC Information ที่เก็บอยู่ภายในเครื่อง (ปกติแล้วเครื่องที่สามารถส่ง SMS ได้มีข้อมูล SMSC ภายในเครื่องอยู่แล้ว)
- 11 First octet of the SMS-SUBMIT message
- 00 TP-message-Reference โดยที่ 00 คือให้เครื่องตั้งหมายเลขอ้างอิงข้อความขึ้นเอง
- 0A Address-Length คือความยาวของเลขหมายผู้รับ (10 ตัว)
- 81 ใช้แสดงว่าเบอร์โทรศัพท์เคลื่อนที่เป็นแบบภายในประเทศ
- 8086682752 เบอร์โทรศัพท์เคลื่อนที่ปลายทางที่ทำการเข้ารหัสแบบสลับแล้ว โดยหมายเลขที่แท้จริงคือ 08-68867225
- 00 TP-PID (Protocol identifier) เป็น 00
- 00 TP-DCS (Data coding scheme) เป็น 00
- AA TP-Validity-Period โดยที่ AA หมายถึง ช่วงเวลาหมดอายุของข้อความ 4 วัน ถ้าภายในช่วงเวลานี้ยังไม่ถึงปลายทาง ข้อความจะถูกยกเลิกโดยอัตโนมัติ
- 05 TP-User-Data-Length คือความยาวของข้อความก่อนแปลงรหัส นั่นคือ “kmidl” = 5 ตัวอักษร
- CB66924A04 TP-User-Data เป็นข้อมูลในส่วนของข้อความตัวอักษรแบบ 7 บิต นั่นคือคำว่า “kmidl” ที่ผ่านการแปลง (เข้ารหัส) เป็นข้อมูลแบบ 8 บิต แล้ว

2.8 IC 555

ไอซี 555 เป็นไอซีไทเมอร์ที่สามารถนำไปประยุกต์ใช้งานได้อย่างกว้างขวางและเป็นที่ยอมรับใช้งานมากในการสร้างสัญญาณรูปคลื่นแบบต่างๆ เช่นสัญญาณ square wave, สัญญาณ ramp, สัญญาณ pulse และการตั้งเวลา ไอซีเบอร์ 555 เป็นอุปกรณ์วงจรรวมที่มีอุปกรณ์อิเล็กทรอนิกส์อื่นๆ อยู่ภายใน และมีส่วนที่ต้องต่อกับภายนอกเพื่อควบคุมการทำงาน และใช้งานเป็นลักษณะต่างๆ ซึ่งง่ายต่อการออกแบบ และง่ายในการสร้างสัญญาณ pulse ความถี่ต่างๆ อีกทั้งสามารถเข้าใจการทำงานได้ง่าย แสดงรูปโครงสร้างภายในของไอซีเบอร์ 555 ดังรูปที่ 2.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 โครงสร้างภายในไอซี 555

ตารางที่ 2.4 หน้าที่ของแต่ละขาของไอซีเบอร์ 555

ขา	ชื่อ	หน้าที่
1	GND	กราวด์ หรือ คอมมอนส์
2	TR	พัลส์สั้นกระตุ้นทริกเกอร์เพื่อเริ่มนับเวลา
3	Q	ช่วงการนับเวลา เอาต์พุตจะอยู่ที่ $\pm V_{CC}$
4	R	ช่วงเวลานับ อาจหยุดโดยการใช้พัลส์รีเซ็ต
5	CV	แรงดันควบคุมยอมให้เข้าถึงตัวหารแรงดันภายใน ($2/3 V_{CC}$)
6	THR	เทรตโวลต์ที่จุดช่วงเวลานับ
7	DIS	เชื่อมต่อกับคาปาซิเตอร์ตัวหนึ่ง ซึ่งเวลาคายประจุของมันจะมีผลต่อช่วงเวลากการนับ
8	V+, V _{CC}	แรงดันจ่ายไฟบวก ซึ่งต้องอยู่ในช่วง +5 ถึง +15 V

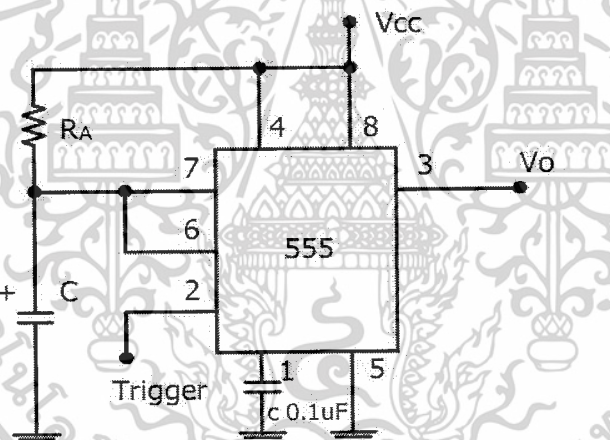
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไอซี 555 มีโหมดการทำงานอยู่ 3 โหมดด้วยกัน คือ

- โมโนสเตเบิล (Mono-stable) ในโหมดนี้ การทำงานของไอซี 555 จะเป็นแบบซิงเกิลช็อต หรือ วันช็อต โดยการสร้างสัญญาณครั้งเดียว ประยุกต์การใช้งานสำหรับการนับเวลา การตรวจสอบพัลส์ สวิตช์สัมผัส ฯลฯ
- อะสเตเบิล (A-stable) ในโหมดนี้ การทำงานจะเป็นออสซิลเลเตอร์ การใช้งาน ได้แก่ ทำไฟกระพริบ กำเนิดพัลส์ กำเนิดเสียง เตือนภัย ฯลฯ
- ไบสเตเบิล (Bi-stable) ในโหมดนี้ ไอซี 555 สามารถทำงานเป็นฟลิปฟล็อป (flip-flop) ถ้าไม่ต่อขา DIS และไม่ใช่คาปาซิเตอร์ ใช้เป็นสวิตช์ bounce-free latched switches เป็นต้น

2.8.1 วงจรโมโนสเตเบิล (Mono-stable Multivibrator)

เป็นวงจรที่สร้างสัญญาณพัลส์ขึ้นมา 1 ลูกหลังจากที่มีการทริก (การกระตุ้น) ให้วงจรเกิดสัญญาณ มีความกว้างของพัลส์ที่สามารถกำหนดได้จากค่า RC มีรูปวงจรแสดงดังรูปที่ 2.17



รูปที่ 2.17 วงจร Mono-stable Multivibrator

หลักการการทำงานของวงจรโมโนสเตเบิล

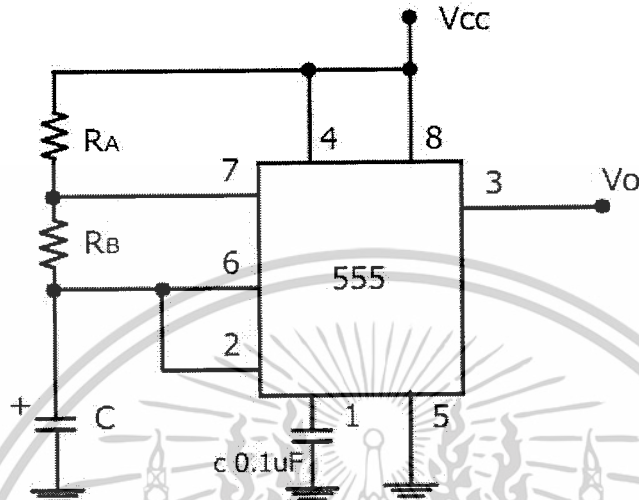
ขณะที่ไม่มีสัญญาณทริกที่ขา 2 $V_{trig} = V_{cc}$ แรงดันที่คร่อม C จะมีค่าเป็น 0 เพราะ V_{cc} จะมีกระแสไหลผ่าน R_A ผ่าน Tr ลงกราวด์ ได้ $V_o = 0$

เมื่อมีสัญญาณทริกที่ขา 2 คือ $V_{trig} = 0$ Tr ภายในจะ “off” V_{cc} ทำการเก็บประจุ (Charge C) จนกระทั่งเกิดแรงดันตกคร่อม C ประมาณ $2V_{cc}/3$ C จะคายประจุผ่าน Tr โดยขณะที่ C คายประจุ output จะเป็นลอจิก 1 ได้ $V_o = V_{cc}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.2 วงจรอะสเตเบิล (A-stable Multivibrator)

เป็นวงจรที่สร้างสัญญาณ Square wave มีความถี่ที่สามารถกำหนดได้ตามต้องการด้วยค่า R_A , R_B และ C แสดงดังรูปวงจรรูปที่ 2.18



รูปที่ 2.18 วงจร A-stable Multivibrator

หลักการทํางานของวงจรอะสเตเบิล

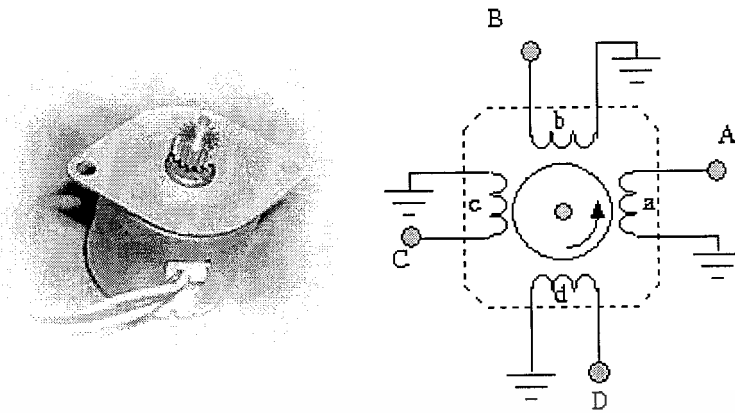
ใช้หลักการเก็บประจุ และคายประจุของคาปาซิเตอร์ โดยจะทำการเก็บประจุผ่าน R_A , R_B และจะคายประจุผ่าน R_B และ T_r ภายในลงกราวด์

- ขณะที่คาปาซิเตอร์เก็บประจุจนถึงระดับแรงดัน $2/3V_{CC}$ ขณะนั้นแรงดันของ $V_o = V_{CC}$
- ขณะที่คาปาซิเตอร์คายประจุออกเกิดแรงดันตกคร่อมช่วงจาก $2/3V_{CC}$ ถึง $V_{CC}/3$ ขณะนั้นแรงดันของ $V_o = 0$ (ศูนย์โวลท์)

2.9 Stepping motor

Stepping motor เป็นมอเตอร์ไฟฟ้าชนิดหนึ่งที่มี input เป็น Binary voltage และมี output เป็นการเคลื่อนที่ในเชิงมุม (หมุน) แกนหมุน (Shaft) เป็น Step Stepping Motor สามารถควบคุมตำแหน่งการหมุนได้ ซึ่งจะมีความละเอียดของมุมในการหมุนและ Step ที่แตกต่างกันออกไป ทั้งนี้ก็ขึ้นอยู่กับ ชนิดของมอเตอร์และลักษณะการส่งสัญญาณไปควบคุมมอเตอร์ ทำให้ Stepping Motor มีความยืดหยุ่นในการนำมาใช้งาน ทำให้สามารถนำมาประยุกต์ใช้งานได้หลายรูปแบบ ไม่ว่าจะเป็นหุ่นยนต์หรือแขนกลต่างๆ เนื่องจากมีความแม่นยำในการควบคุมตำแหน่งสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 โครงสร้างของ Stepping motor

จากรูปที่ 2.19 Stepping motor มีโครงสร้างภายในประกอบไปด้วย ขดลวด stator 4 ขด ล้อมรอบแกนหมุน หลักการทำงาน คือ เมื่อจ่ายกระแสไฟฟ้าให้กับขดลวด Stator Coil a, b, c, d ไม่พร้อมกัน นั่นคือ ถ้าเราจ่ายกระแสให้ a ก่อน โดยไม่จ่ายให้ขดอื่น แล้วตามด้วย b, c และ d เรียงตามลำดับ จะทำให้เกิดสนามแม่เหล็ก หมุนวนในลักษณะทวนเข็มนาฬิกา ซึ่งส่วนของ Rotor ที่เป็นแม่เหล็กถาวรก็จะหมุนตามสนามแม่เหล็กไปด้วย คือ ทวนเข็มนาฬิกา ในทำนองเดียวกันถ้าเราจ่ายกระแสให้ขด a, d, c, b ก็จะทำให้สนามแม่เหล็กในทิศทางตามเข็มนาฬิกา การกำหนดความเร็วของ Stepping motor ทำได้โดยการเปลี่ยนแปลงความเร็วของการเปลี่ยนการจ่ายกระแสจากขดลวดหนึ่งไปยังอีกขดลวดหนึ่งให้เร็วขึ้น

2.9.1 การควบคุมการทำงานของ Stepping motor แบบ 4 เฟส

ในการควบคุมการทำงานของ Stepping motor สามารถแบ่งออกได้ 3 แบบ ดังนี้

- ควบคุมแบบ Full step 1 เฟส หรือแบบเวฟ
- ควบคุมแบบ Full step 2 เฟส
- ควบคุมแบบ Half step

- ควบคุมแบบ Full step 1 เฟส หรือแบบเวฟ

หลักการคือ เริ่มจากจ่ายกระแสให้กับขดลวด stator เฟสที่ 1 จากนั้นกระตุ้นเฟสที่ 2 และ เฟสที่ 3 ไปเรื่อยๆ ตามลำดับ จากนั้นก็วนกลับมาที่ขดลวด stator เฟสที่ 1 อีกครั้งและวนลูปไปเรื่อยๆ ก็จะทำให้ Stepping motor หมุนและในทางกลับกัน ถ้าต้องการให้ Stepping motor หมุนกลับทางก็ต้องกระตุ้นขดลวด stator เฟสที่ 4 เฟสที่ 3 เฟสที่ 2 และเฟสที่ 1 ตามลำดับ สามารถเขียนขั้นตอนการทำงานเป็นตารางออกมาได้ดังตาราง 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.5 ตารางการควบคุมแบบ Full step 1 เฟส

STEP	เฟส 1	เฟส 2	เฟส 3	เฟส 4
Step 1	ON	OFF	OFF	OFF
Step 2	OFF	ON	OFF	OFF
Step 3	OFF	OFF	ON	OFF
Step 4	OFF	OFF	OFF	ON

- ควบคุมแบบ Full step 2 เฟส

ในการควบคุม Stepping motor แบบ 2 เฟสนั้น จะต้องจ่ายกระแสไฟฟ้าเพื่อกระตุ้นขดลวดของมอเตอร์ทีละ 2 เฟส ในเวลาเดียวกันและเรียงกันไปตามลำดับซึ่งได้แสดงดังตารางที่ 2.6 โดย Stepping motor จะหมุนเหมือนกับการควบคุมแบบเวฟ แต่การควบคุมแบบ 2 เฟสจะให้แรงบิดที่สูงกว่าแบบเวฟ

ตารางที่ 2.6 ตารางการควบคุมแบบ Full step 2 เฟส

STEP	เฟส 1	เฟส 2	เฟส 3	เฟส 4
Step 1	ON	ON	OFF	OFF
Step 2	OFF	ON	ON	OFF
Step 3	OFF	OFF	ON	ON
Step 4	ON	OFF	OFF	ON

- ควบคุมแบบ Half wave

การควบคุม Stepping motor แบบครึ่งสเต็ปนั้นทำให้สามารถเพิ่มความละเอียดในการควบคุมการหมุนของ Stepping motor มีความแม่นยำมากขึ้น ซึ่งเป็นการควบคุมแบบผสมระหว่างการควบคุมแบบเวฟและแบบ Full step 2 เฟสเข้าด้วยกัน ลักษณะการจ่ายกระแสไฟ เพื่อกระตุ้นขดลวดจะแสดงดังตารางที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.7 ตารางการควบคุมแบบ Half wave

STEP	เฟส 1	เฟส 2	เฟส 3	เฟส 4
Step 1	ON	OFF	OFF	OFF
Step 2	ON	ON	OFF	OFF
Step 3	OFF	ON	OFF	OFF
Step 4	OFF	ON	ON	OFF



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

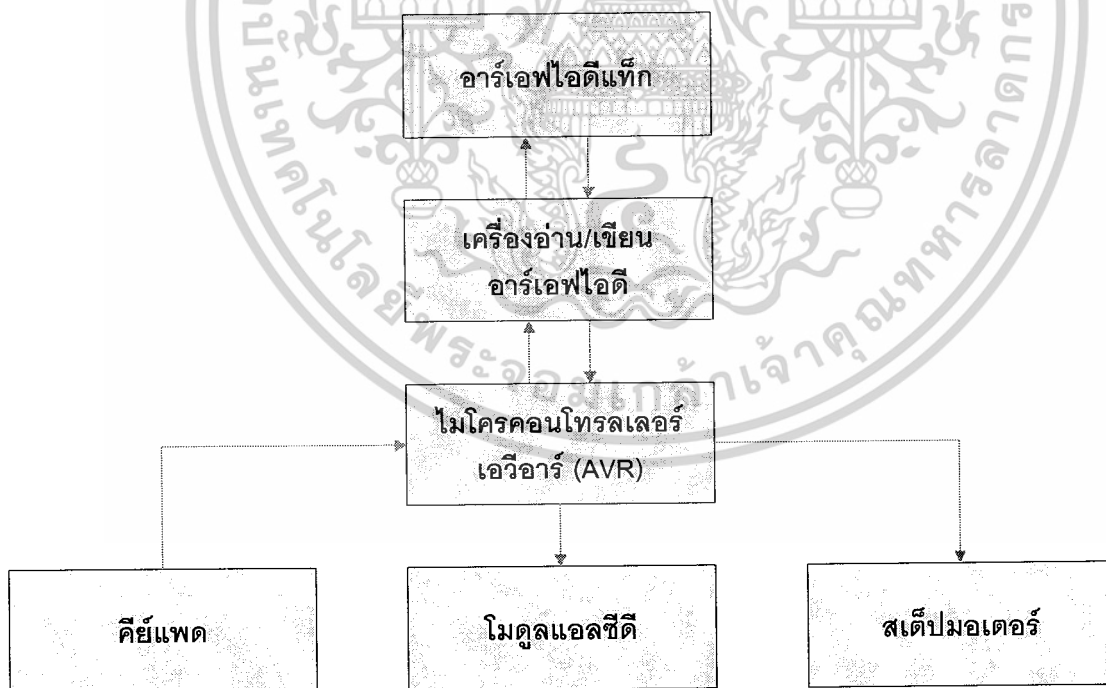
บทที่ 3

การออกแบบและการสร้าง

ในโครงการนี้ได้แบ่งการออกแบบและการสร้างออกเป็น 3 ส่วน ประกอบด้วย ส่วนของระบบการซื้อสินค้า, ระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด และ โปรแกรมเติมเงิน ซึ่งรายละเอียดของการออกแบบและการสร้างจะกล่าวถึงดังต่อไปนี้

3.1 การออกแบบระบบการซื้อสินค้า

ฮาร์ดแวร์ที่ใช้ประกอบด้วย เครื่องอ่าน/เขียนอาร์เอฟไอดี, คีย์แพด, โมดูลแอลซีดี และ สเต็ปมอเตอร์ โดยจะทำการเชื่อมต่อกับไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) ในขั้นตอนแรกผู้ใช้บริการจะทำการเลือกสินค้าจากคีย์แพด เมื่อไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) รับค่าการกดเลือกสินค้าจากคีย์แพดแล้ว ผู้ใช้บริการจะต้องนำอาร์เอฟไอดีแท็กเข้ามาที่เครื่องอ่าน/เขียนอาร์เอฟไอดี เพื่อที่จะให้ไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) ทำการส่งข้อมูลไปยังเครื่องอ่าน/เขียนอาร์เอฟไอดี เพื่อทำการหักเงินภายในบัตร หลังจากทำการซื้อสินค้าเสร็จแล้ว ไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) จะทำการสั่งให้โมดูลแอลซีดีแสดงจำนวนเงินที่เหลือภายในอาร์เอฟไอดีแท็ก พร้อมกับสั่งให้สเต็ปมอเตอร์ปล่อยสินค้า แสดงส่วนประกอบหลักของฮาร์ดแวร์ดังรูปที่ 3.1



รูปที่ 3.1 ส่วนประกอบหลักของฮาร์ดแวร์ในส่วน of ระบบการซื้อสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ส่วนประกอบของเครื่องอ่าน/เขียนอาร์เอฟไอดี

ส่วนประกอบของเครื่องอ่าน/เขียนอาร์เอฟไอดีที่ใช้ในระบบการซื้อสินค้าเป็นของบริษัท StrongLink รุ่น SL015M ดังรูปที่ 3.2 โดยขาโมดูลมีคุณสมบัติดังตารางที่ 3.1

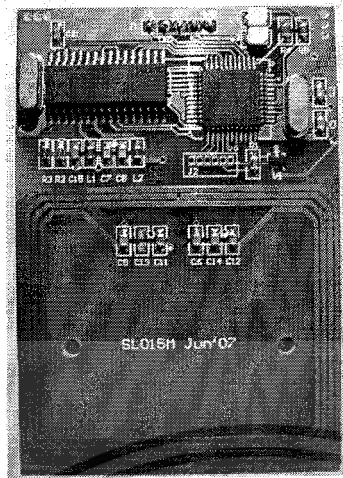
ตารางที่ 3.1 แสดงข้อมูลขาโมดูลอาร์เอฟไอดี

ขา	สัญลักษณ์	ชนิด	รายละเอียด
1	TagSta	Output	แท็กส์ตรวจจับสัญญาณ ระดับสัญญาณต่ำเมื่อมีแท็กเข้ามาอยู่ในระยะ ระดับสัญญาณสูงเมื่อไม่มีแท็กเข้ามาอยู่ในระยะ
2	TXD	Output	พอร์ตเอาต์พุตอนุกรม
3	RXD	Input	พอร์ตอินพุตอนุกรม
4	VCC	PWR	แหล่งจ่ายไฟ
5	GND	PWR	กราวด์

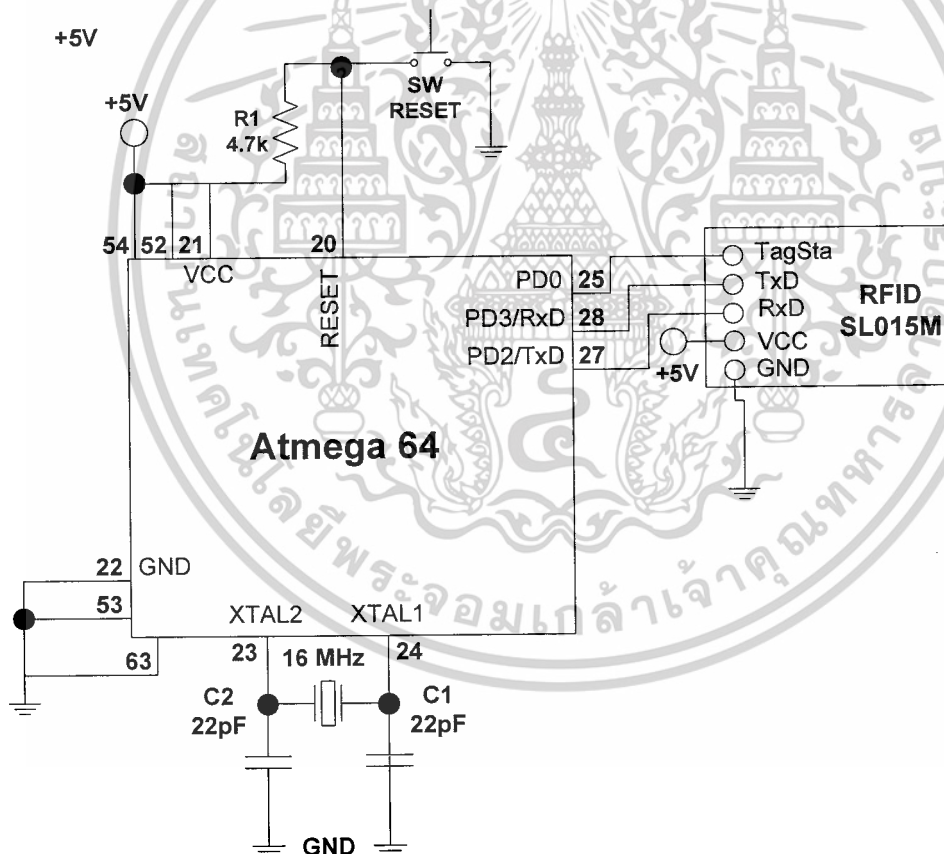
3.2.1 คุณลักษณะหลักของเครื่องอ่าน/เขียนอาร์เอฟไอดี

- สนับสนุนอาร์เอฟไอดีแท็กชนิด Mifare 1K (S50), Mifare 4K (S70), Mifare Pro(x) Mifare Ultralight
- ทำงานที่ความถี่ 13.56 เมกะเฮิร์ต, ISO 14443A
- สามารถตรวจจับอาร์เอฟไอดีแท็กได้อัตโนมัติ
- มีสายอากาศอยู่ภายใน
- เชื่อมต่อแบบยูเออาร์ที (UART) อัตราความเร็วในการรับส่งข้อมูล (Baud Rate) 9600 - 115200 บิตต่อวินาที (Bit/s)
- ทำงานที่ไฟกระแสตรงอยู่ในช่วง 4.5 V ถึง 5.5 V
- ระยะกระทำการ ประมาณ 80 มิลลิเมตร, ขึ้นอยู่กับชนิดของอาร์เอฟไอดีแท็ก
- ทนทานต่ออุณหภูมิระหว่าง -40 °C ถึง +85 °C
- มีขนาด กว้าง*ยาว*สูง 80×55×7 มิลลิเมตร
- มีหลอดไฟแอลอีดีแสดงไฟสีเขียวเมื่อมีอาร์เอฟไอดีแท็กอยู่ในระยะ และไฟสีแดงเมื่อไม่มีอาร์เอฟไอดีแท็ก
- ระดับสัญญาณต่ำเมื่อมีอาร์เอฟไอดีแท็กเข้ามาอยู่ในระยะ และระดับสัญญาณสูงเมื่อไม่มีอาร์เอฟไอดีแท็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 เครื่องอ่าน/เขียนอาร์เอฟไอดี รุ่น SL015M

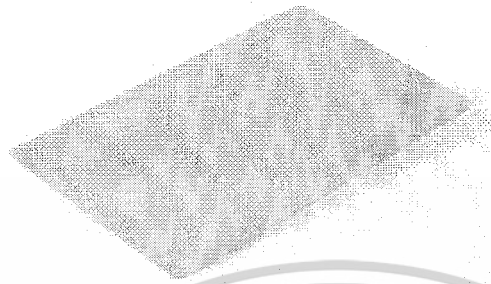


รูปที่ 3.3 การเชื่อมต่อเครื่องอ่าน/เขียนอาร์เอฟไอดี รุ่น SL015M กับไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) เบอร์ ATmega64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนประกอบของอาร์เอฟไอดีแท็ก

ส่วนประกอบของอาร์เอฟไอดีแท็กที่ใช้ คือ Mifare S50 ซึ่งมีหน่วยความจำขนาด 1 กิโลไบต์ แสดงดังรูปที่ 3.4



รูปที่ 3.4 บัตร Mifare S50

3.3.1 คุณลักษณะหลักของอาร์เอฟไอดีแท็ก

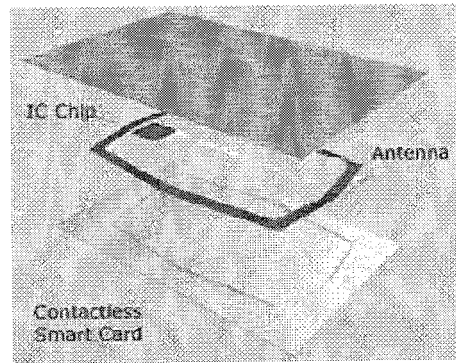
- การรับส่งข้อมูลและพลังงานไฟฟ้าระหว่างเครื่องอ่าน/เขียน กับบัตรเป็นแบบไร้สัมผัส
- ระยะห่างระหว่างเครื่องอ่าน/เขียน มากถึง 10 เซนติเมตร (ขึ้นอยู่กับสายอากาศ)
- ส่งผ่านข้อมูลระหว่างเครื่องอ่าน/เขียน กับบัตรด้วยความเร็วสูงถึง 106 กิโลบิต/วินาที
- ใช้งานความถี่ที่ย่าน 13.56 MHz
- มีการตรวจสอบความถูกต้องของการรับส่งข้อมูลสูงด้วยเทคนิคการทำ 16 bit CRC, parity, Bit Coding และ Bit Counting
- มีระบบป้องกันการคัดลอกข้อมูล (True Anticollision)
- เวลาในการอ่าน/เขียนข้อมูลน้อยกว่า 100 มิลลิวินาที
- หน่วยความจำแบบอีอีพรอม (EEPROM) ขนาด 1 กิโลไบต์ (S50) โดยแบ่งหน่วยความจำเป็น 16 เซกเตอร์
- เก็บข้อมูลได้ถึง 10 ปี และเขียนข้อมูลซ้ำได้ถึง 100,000 ครั้ง
- รองรับการทำงานแบบ Multi Application โดยการแบ่งหน่วยความจำเป็นบล็อกๆ และแต่ละบล็อกมีรหัสผ่านในการเข้าถึงข้อมูลภายใน จึงสามารถใช้บัตรใบเดียวกันหลายงานได้

3.3.2 ส่วนประกอบของบัตร Mifare S50

ส่วนประกอบของบัตรจะแบ่งได้เป็น 3 ส่วน คือ

- IC Chip หน่วยความจำของบัตร
- เสาอากาศ (Antenna) สำหรับส่งข้อมูล
- Card, ส่วนที่บรรจุ IC Chip และ Antenna

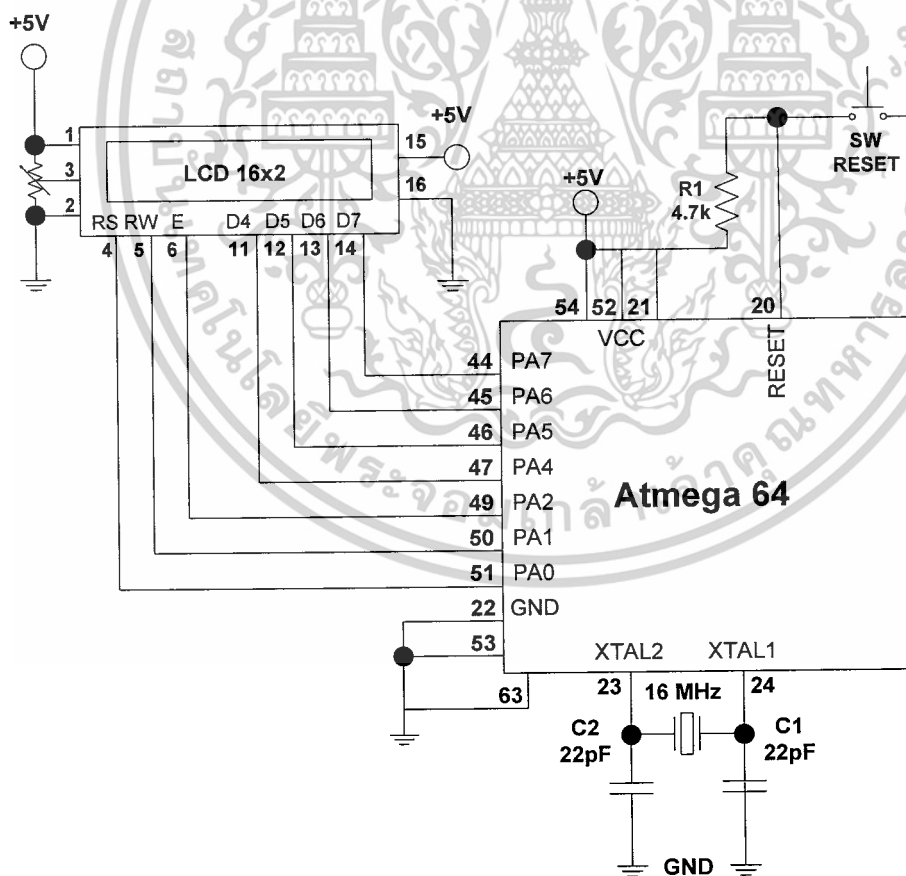
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 ส่วนประกอบของบัตร Mifare S50

3.4 ส่วนประกอบของโมดูลแอลซีดี

ส่วนประกอบของโมดูลแอลซีดีที่ใช้เป็นแบบ 16 ตัวอักษร 2 บรรทัด โดยจะทำการเชื่อมต่อกับพอร์ตเอ (Port A) ของไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) เบอร์ ATmega64 ซึ่งจะใช้การเชื่อมต่อแบบ 4 บิต แสดงดังรูปที่ 3.6



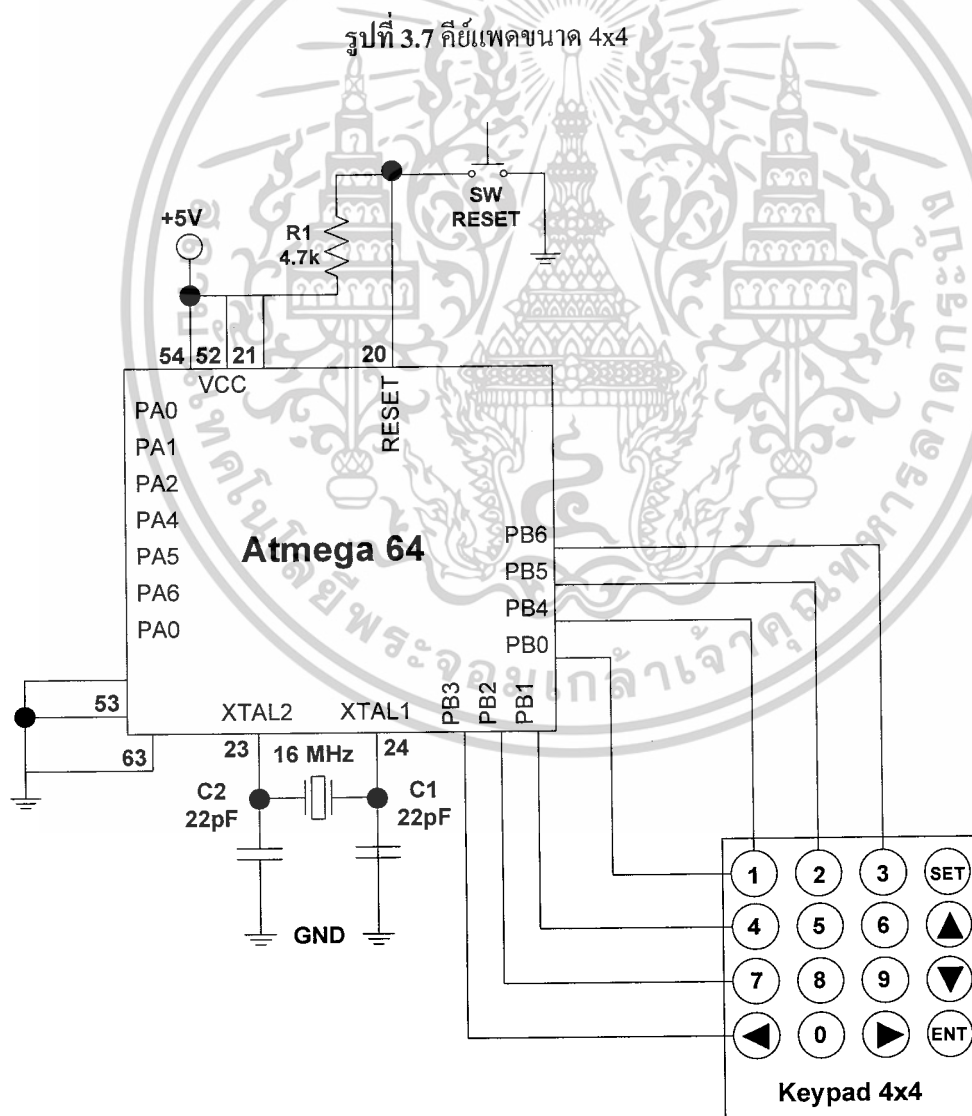
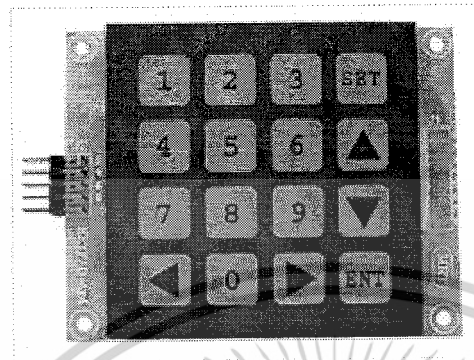
รูปที่ 3.6 การเชื่อมต่อโมดูลแอลซีดีกับพอร์ตเอของไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) เบอร์

ATmega64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 ส่วนประกอบของคีย์แพด

ส่วนประกอบของคีย์แพดที่ใช้เป็นแบบ 4x4 แสดงดังรูปที่ 3.7 โดยจะทำการเชื่อมต่อกับพอร์ตบี (Port B) ของไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) แสดงดังรูปที่ 3.8

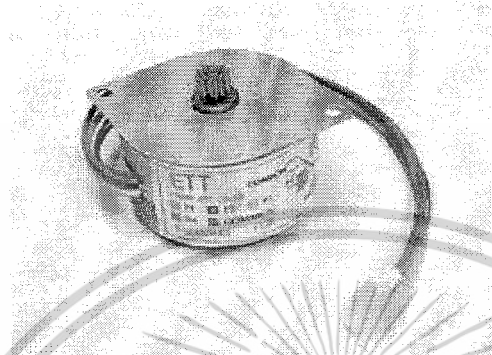


รูปที่ 3.8 การเชื่อมต่อกับคีย์แพดกับพอร์ตบีของไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) เบอร์ ATmega64

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 ส่วนประกอบของสแตมป์มอเตอร์

ส่วนประกอบของสแตมป์มอเตอร์เป็นของบริษัท ETT รุ่น SMJ40-48 แสดงดังรูปที่ 3.9 ซึ่งเป็นมอเตอร์ขนาด 4 เฟสหมุนได้ 7.5 องศาต่อสแตมป์ โดยจะทำการเชื่อมต่อกับพอร์ตอี (Port E) และ พอร์ตเอฟ (Port F) ของไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) แสดงดังรูปที่ 3.9



รูปที่ 3.9 ETT สแตมป์มอเตอร์

3.6.1 การขับสแตมป์มอเตอร์แบบฟลูตสเต็ป 2 เฟส

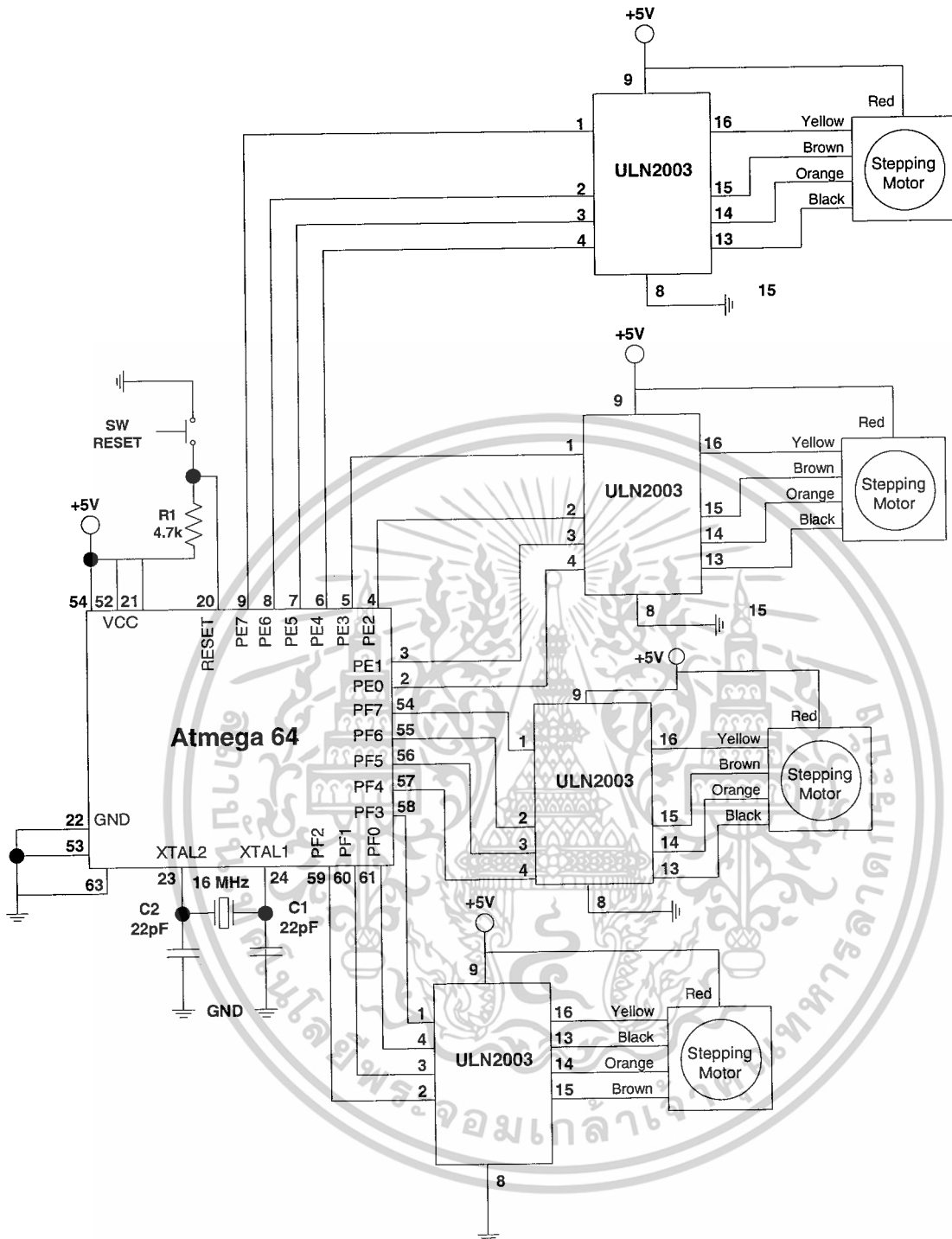
การขับสแตมป์มอเตอร์แบบฟลูตสเต็ปสองเฟส จะเป็นการขับสแตมป์มอเตอร์ที่จ่ายกระแสไฟให้กับขดลวดสองขดพร้อมกัน จะกินกระแสไฟมากกว่าแบบหนึ่งเฟส แต่จะให้แรงบิดที่มากขึ้น ลักษณะการขับสแตมป์มอเตอร์แบบสองเฟส แสดงดังตารางที่ 3.2

ตารางที่ 3.2 การทำงานของขดลวดแบบฟลูตสเต็ป 2 เฟส

สแตมป์ที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	ทำงาน	-	-
2	-	ทำงาน	ทำงาน	-
3	-	-	ทำงาน	ทำงาน
4	ทำงาน	-	-	ทำงาน

การควบคุมสแตมป์มอเตอร์แบบฟลูตสเต็ปสองเฟสให้หมุนไปกลับจะควบคุมผ่านไอซีเบอร์ ULN2003 ซึ่งมีเอาต์พุตแบบคอลเล็กเตอร์เปิด (Open Collector) และสามารถขับกระแสได้สูงสุดถึง 500 มิลลิแอมป์ โดยสามารถนำมาเชื่อมต่อกับไมโครคอนโทรลเลอร์ได้โดยตรง ดังรูปที่ 3.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



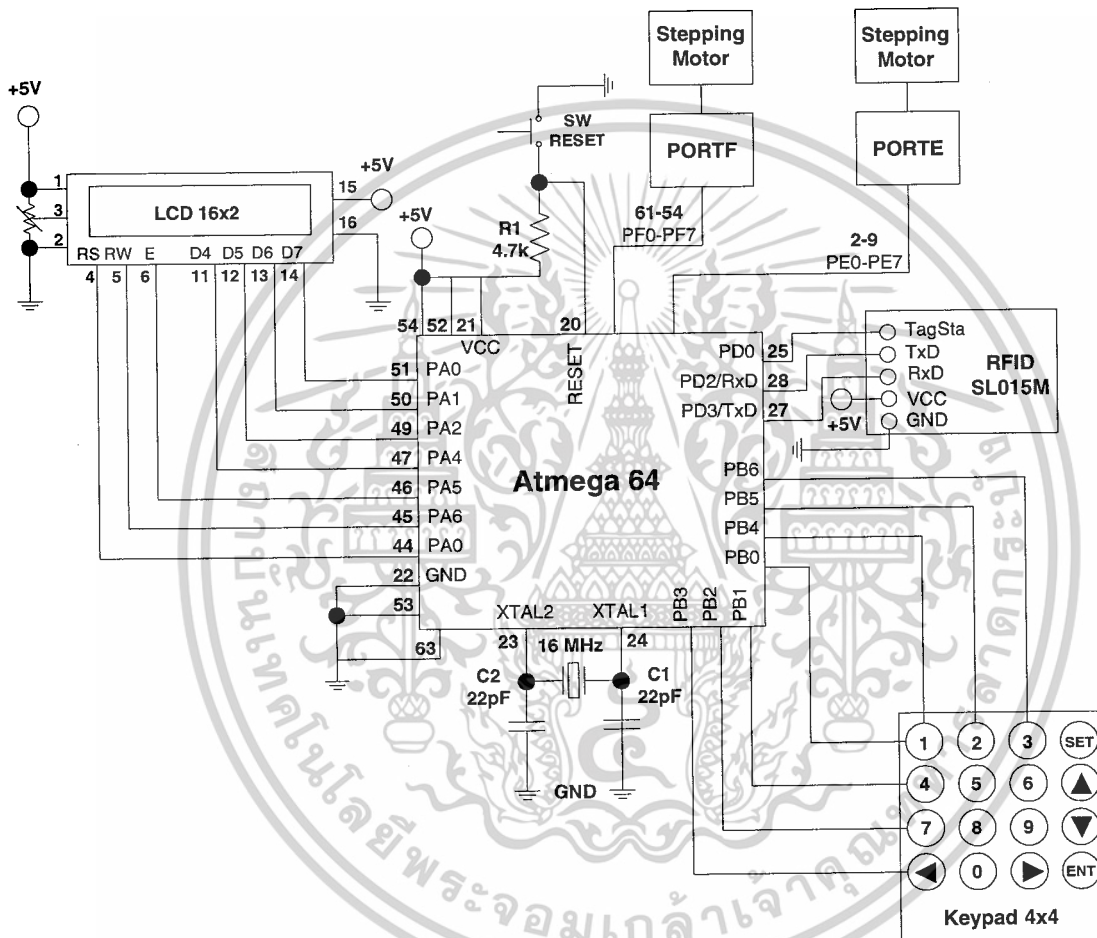
รูปที่ 3.10 การเชื่อมต่อดีปมอเตอร์กับพอร์ตไอและพอร์ตเอฟของไมโครคอนโทรลเลอร์เอวีอาร์ (AVR)

บอร์ด ATmega64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 ภาพรวมของวงจรที่ใช้ในระบบการซื้อสินค้า

เนื่องจากเครื่องอ่าน/เขียนอาร์เอฟไอดี รุ่น SL015M นั้น ต้องการระดับสัญญาณที่เป็น TTL เพราะฉะนั้น การเชื่อมต่อไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) กับเครื่องอ่าน/เขียนอาร์เอฟไอดี จะสามารถต่อกันได้โดยตรงโดยไม่ต้องต่อไอซี MAX232 เพราะสัญญาณที่ส่งออกมาจากไมโครคอนโทรลเลอร์เอวีอาร์ (AVR) จะเป็นระดับสัญญาณ TTL แสดงการเชื่อมต่อดังรูปที่ 3.11

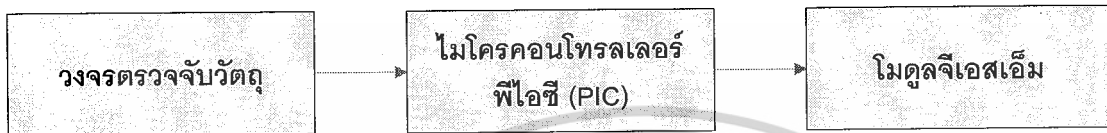


รูปที่ 3.11 ภาพรวมของวงจรที่ใช้ในส่วนของการซื้อสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8 การออกแบบระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด

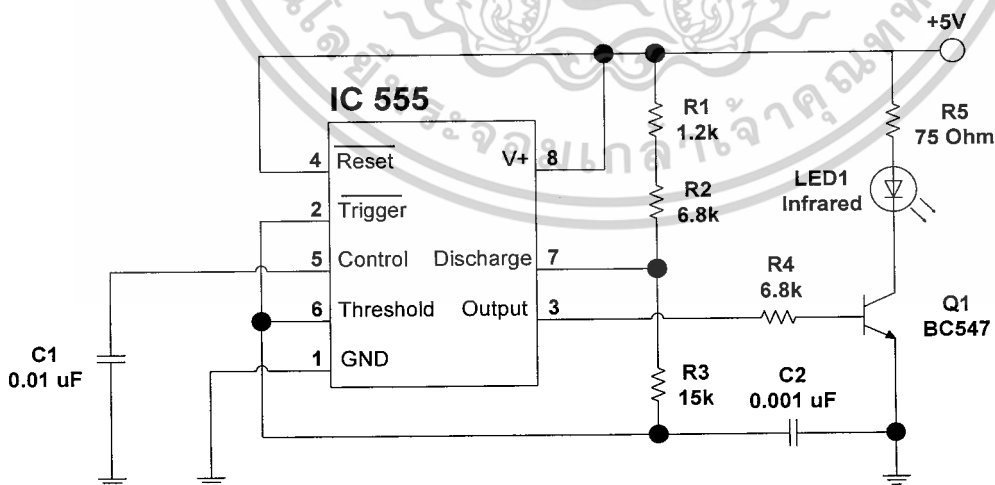
ฮาร์ดแวร์ที่ใช้ประกอบด้วย วงจรตรวจจับวัตถุ, โมดูลจีเอสเอ็ม โดยจะทำการเชื่อมต่อกับ ไมโครคอนโทรลเลอร์พีไอซี (PIC) โดยเมื่อสินค้าที่ตู้ขายของหมดแล้ว ไมโครคอนโทรลเลอร์พีไอซี (PIC) จะทำการส่งข้อมูลซึ่งเป็นชุดคำสั่งเอทีคอมมานด์ (AT Command) ไปยังโมดูลจีเอสเอ็ม เพื่อให้โมดูลจีเอสเอ็มทำการส่งข้อความสั้น (SMS : Short Message Service) แจ้งเตือนไปยังผู้ประกอบการ แสดงดังรูปที่ 3.12



รูปที่ 3.12 ส่วนประกอบหลักของฮาร์ดแวร์ในส่วนของการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด

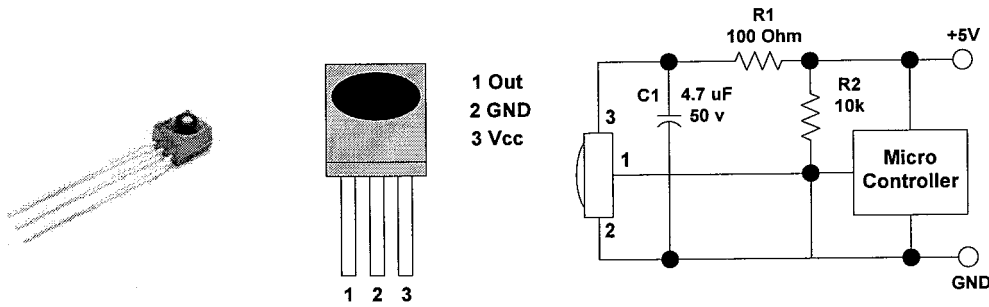
3.9 ส่วนประกอบของวงจรตรวจจับวัตถุ

ในการสร้างวงจรตรวจจับวัตถุ วงจรที่ใช้ในฝั่งตัวส่ง จะใช้ไอซี 555 ทำงานในโหมดอะสเตเบิล มัลติไวเบเรเตอร์ (A-stable Multivibrator) เพื่อสร้างสัญญาณความถี่ 38 กิโลเฮิร์ต (kHz) และส่งสัญญาณที่ได้ให้กับหลอดแอลอีดีอินฟราเรด ส่วนวงจรที่ใช้ในฝั่งรับจะใช้โมดูล TSOP 4838 รับสัญญาณอินฟราเรด 38 กิโลเฮิร์ต โดยถ้าโมดูล TSOP 4838 ไม่สามารถตรวจจับสัญญาณอินฟราเรดความถี่ 38 กิโลเฮิร์ตได้ เอาต์พุตที่ได้จะเป็นลอจิก 1 ในทางกลับกัน ถ้าโมดูลสามารถตรวจจับสัญญาณอินฟราเรดความถี่ 38 กิโลเฮิร์ตได้ เอาต์พุตที่ได้จะเป็นลอจิก 0 วงจรที่ใช้ในฝั่งตัวส่งและตัวรับจะแสดงดังรูปที่ 3.13 และ 3.14 ตามลำดับ



รูปที่ 3.13 วงจรกำเนิดสัญญาณความถี่ 38 กิโลเฮิร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 TSOP 4838 โมดูลรับสัญญาณอินฟราเรด 38 กิโลเฮิร์ต

3.9.1 หลักการสร้างสัญญาณความถี่ 38 กิโลเฮิร์ต

ก่อนที่จะกล่าวถึงหลักการสร้างสัญญาณนั้น ผู้จัดทำขอกล่าวถึงสาเหตุที่ใช้ความถี่ในการรับส่งอินฟราเรด โดยสาเหตุที่ใช้ความถี่ก็เพราะว่ามันสามารถทำให้รับ/ส่งได้ไกลขึ้น และมีผลกระทบต่อแสงน้อยกว่าการรับส่งแบบธรรมดา

การสร้างสัญญาณความถี่ 38 กิโลเฮิร์ตนั้น จะใช้ไอซี 555 ในการสร้างความถี่ โดยการต่อวงจรจะต่อเป็นแบบอะอสเตเบิล มัลติไวเบเรเตอร์ (A-stable Multivibrator) ซึ่งมีตัวต้านทาน (R_1 , R_2 และ R_3) และ ตัวเก็บประจุ (C_2) เป็นตัวกำหนดความถี่ที่เกิดขึ้น โดยคำนวณได้จากสูตร

$$f = \frac{1.44}{(R_A + 2R_B)C}$$

โดยที่ $R_A = R_1 + R_2$, $R_B = R_3$, $C = C_2$

$$f = \frac{1.44}{(R_1 + R_2 + 2R_3) \times C_2}$$

จากวงจรในรูปที่ 3.13 แทนค่า $R_1 = 1.2k$, $R_2 = 6.8k$, $R_3 = 15k$, $C_2 = 1nF$

$$f = \frac{1.44}{(1.2k + 6.8k + 2(15k)) \times 10^{-9}}$$

$$f = \frac{1.44}{38 \times 10^{-6}}$$

$$f \approx 37.89 kHz$$

สัญญาณความถี่ที่สร้างได้จะออกจากขา 3 ของไอซี 555 ส่งต่อให้ขาเบสของทรานซิสเตอร์ Q1 เพื่อทำการขับ (Drive) หลอดแอลอีดีอินฟราเรดของตัวส่ง โดยมี R_4 ทำหน้าที่จำกัดกระแสเบส (I_b) ให้กับขาเบส, R_5 ทำหน้าที่จำกัดกระแสคอลเล็กเตอร์ (I_c) ให้กับหลอดแอลอีดีอินฟราเรดและขาคอลเล็กเตอร์ เพื่อไม่ให้กระแสเกินพิกัดที่จะรับได้ การออกแบบการทำงานของทรานซิสเตอร์จะออกแบบให้ทำงานในเอกซารันเป็นเอกซารันที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ย่านคัตออฟ (Cut off) และอิ่มตัว (Saturation) คือ ทำงานแบบสวิตช์ การกำหนดกระแสเบสและกระแสคอลเล็กเตอร์ มีผลต่อการรับส่งสัญญาณอินฟราเรด ถ้ากำหนดให้กระแสคอลเล็กเตอร์มากก็จะทำให้ความเข้มของสัญญาณอินฟราเรดมากขึ้นสามารถส่งได้ไกลมากขึ้น สาเหตุที่ต้องขับ (Drive) ก่อนนั้น เนื่องจากแรงดันที่ออกมาจากเอาต์พุตของไอซี 555 มีแรงดันที่ต่ำกว่าแรงดัน VCC ถ้าจ่าย 5 โวลต์ ก็จะออกประมาณ 3.3 โวลต์ ดังนั้นจึงอาจทำให้แรงดันและกระแสไม่เพียงพอที่ขับหลอดแอลอีดีอินฟราเรด

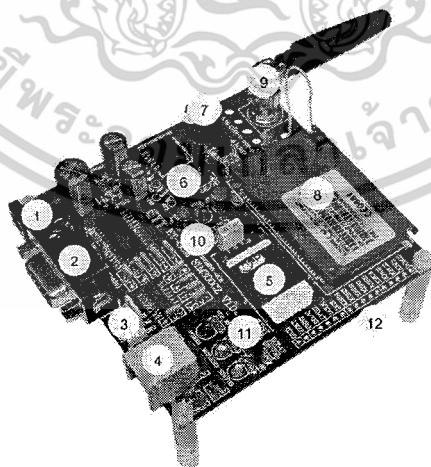
3.9.2 TSOP 4838

เป็นโมดูลที่ใช้ในการรับสัญญาณอินฟราเรดความถี่ 38 กิโลเฮิร์ต ซึ่งโครงสร้างข้างในเป็นไอซีที่มีวงจรมอดูเลชันเพื่อถอดเอาแต่สัญญาณความถี่ที่ต้องการออกมา เอาต์พุตที่ออกมาจากไอซีนี้จะต้องต่อ R พูลอัพด้วย เนื่องจากโครงสร้างด้านในทางเอาต์พุตเป็นแบบคอลเล็กเตอร์เปิด สัญญาณเอาต์พุตที่ออกมาจะอินเวอร์สกับอินพุต คือ เมื่อจับสัญญาณได้เอาต์พุตที่ออกมาจะเป็น 0 ในทางกลับกันถ้าจับสัญญาณไม่ได้เอาต์พุตจะออกเป็น 1

3.10 ส่วนประกอบของโมดูลจีเอสเอ็ม (GSM Module)

โมดูลจีเอสเอ็มที่ใช้ คือ ET-GSM SIM300CZ V1.0 ซึ่ง SIM300CZ เป็นโมดูลสื่อสารระบบ จีเอสเอ็ม/จีพีอาร์เอส ขนาดเล็ก รองรับระบบสื่อสารจีเอสเอ็มความถี่ 900/1800/1900 MHz โดยทำงานผ่านทางพอร์ตสื่อสารอนุกรม อาร์เอส232 ด้วยชุดคำสั่ง AT Command สามารถประยุกต์ใช้งานได้มากมายหลายรูปแบบ ไม่ว่าจะเป็นการรับส่งสัญญาณแบบเสียง, ข้อความ, ข้อมูลและยังรวมถึงการสื่อสารด้วยโปรโตคอล TCP/IP ด้วย

3.10.1 โครงสร้างของบอร์ด ET-GSM SIM300CZ V1



รูปที่ 3.15 โมดูล ET-GSM SIM300CZ V1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.15 หมายเลข 1-11 คือส่วนประกอบของบอร์ด ET-GSM SIM300CZ V1 โดยที่

- หมายเลข 1 เป็น JACK DC-IN แบบมีขั้ว โดยมีด้านนอกเป็นขั้วบวก และด้านในเป็นกราวด์ใช้สำหรับรับแหล่งจ่ายไฟจากภายนอกโดยออกแบบมาให้ใช้กับ แหล่งจ่ายไฟขนาด 5V ขึ้นไป ที่จ่ายกระแสได้ 1A ถึง 3A โดยที่จะมีวงจร Regulate ขนาด 4.2V / 3A สำหรับจ่ายให้กับโมดูล SIM300CZ ซึ่งสามารถใช้กับ Sim Card ของทุกๆเครื่อง่าย และมีวงจร Regulate ขนาด 3.3V / 1A สำหรับจ่ายให้กับวงจรเชื่อมต่อภายนอกโดยไม่ต้องไปดึงไฟจากตัวโมดูลมาใช้ ป้องกันปัญหาโมดูลเสียหายจากวงจรภายนอกดึงกระแสเกินพิกัด และสะดวกต่อการออกแบบวงจรเชื่อมต่อเพิ่มเติม โดยไม่ต้องกังวลว่ากระแสจะไม่พอจ่ายให้กับอุปกรณ์
- หมายเลข 2 เป็น ขั้วต่ออาร์เอส232 (DCE) แบบ DB9 ตัวเมีย สำหรับใช้เชื่อมต่อกับสัญญาณ อาร์เอส 232 (DTE) แบบ DB9 ตัวผู้ จากคอมพิวเตอร์หรืออุปกรณ์ภายนอกอื่นๆ โดยใช้สาย 9 Pin แบบต่อตรง
- หมายเลข 3 เป็น ขั้วต่อ DEBUG ใช้สำหรับพัฒนา และ DEBUG โปรแกรม สำหรับต่อกับ อาร์เอส 232 ในกรณีที่ต้องการพัฒนาโปรแกรมเพิ่มเติมให้กับโมดูล SIM300CZ เอง
- หมายเลข 4 เป็น ขั้วต่อ RJ11 สำหรับใช้เชื่อมต่อกับชุด Handset ในกรณีที่ต้องการใช้งานโมดูล SIM300CZ เพื่อโทรออกและรับสาย โดยสามารถเชื่อมต่อกับ Handset มาตรฐานได้ทั่วไป
- หมายเลข 5 เป็น Socket สำหรับติดตั้ง SIM Card ให้กับโมดูล
- หมายเลข 6 เป็น Switch Push-Button สำหรับใช้ Power-On และ Power-OFF ตัวโมดูล
- หมายเลข 7 เป็น Buzzer สำหรับสร้างเสียงเรียกเข้าในกรณีที่มีการโทรเข้ามายังโมดูล SIM300CZ และสามารถปรับระดับความดังของเสียงได้จากคำสั่ง “AT+CRSL” ได้อีกด้วย
- หมายเลข 8 เป็น จุคอร์รับ โมดูล SIM300CZ พร้อมเสาและสกรูสำหรับยึดโมดูลกับบอร์ด
- หมายเลข 9 เป็น จุดยึด Connector เสาอากาศ จีเอสเอ็มย่านความถี่ 900/1800/1900 MHz รองรับ GPRS Multi-Slot Class10 และ GPRS Mobile Station Class B
- หมายเลข 10 เป็น LED แสดงแหล่งจ่าย VBAT โดยจะติดสว่างเมื่อมีการจ่ายไฟให้บอร์ดแล้ว
- หมายเลข 11 เป็น LED แสดงสถานะของบอร์ด ซึ่งมีด้วยกัน 3 ดวงคือ
 1. POWER สีแดง จะติดสว่าง เมื่อ โมดูลอยู่ในสถานะ Power-ON
 2. NETLIGHT สีเหลือง จะกระพริบเมื่อ โมดูลอยู่ในสถานะ Power-ON
 3. STATUS สีเขียว จะติดสว่างเมื่อ โมดูลอยู่ในสถานะ Power-ON
- หมายเลข 12 เป็น จุดต่อสัญญาณเพิ่มเติมในกรณีที่ต้องการประยุกต์ใช้งาน โมดูลเพิ่มเติม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.10.2 การเชื่อมต่อโมดูลจีเอสเอ็มกับโปรแกรมไฮเปอร์เทอร์มินอล

ในขั้นตอนแรกจะทำการเชื่อมต่อ โมดูลจีเอสเอ็มกับคอมพิวเตอร์โดยใช้โปรแกรมไฮเปอร์เทอร์มินอล (Hyper Terminal) เพื่อทดสอบชุดคำสั่งเอทีคอมมานด์ (AT Command) ในส่วนที่เกี่ยวข้องกับการส่งข้อความสั้น (SMS) แสดงดังรูปที่ 3.16



รูปที่ 3.16 ส่วนการเชื่อมต่อโมดูลจีเอสเอ็มเข้ากับคอมพิวเตอร์

3.10.3 ชุดคำสั่งเอทีคอมมานด์ที่ใช้ในการส่งข้อความสั้น

ในการส่งข้อความสั้น (SMS) นั้น จะใช้การส่งแบบเท็กซ์โหมด (Text Mode) ซึ่งสามารถพิมพ์ตัวอักษรหรือข้อความลงในโปรแกรมไฮเปอร์เทอร์มินอลได้เลย โดยที่โมดูลจีเอสเอ็มจะเก็บตัวอักษรหรือข้อความเป็นรหัสแอสกีไค้ดและนำรหัสแอสกีไค้ดที่ได้ส่งให้กับเลขหมายปลายทาง และเลขหมายปลายทางจะต้องแปลงรหัสแอสกีไค้ดที่ได้มาให้เป็นตัวอักษรเพื่อแสดงผลออกหน้าจอ โทรศัพท์เคลื่อนที่ โดยขั้นตอนการส่งชุดคำสั่งเอทีคอมมานด์ (AT Command) มีดังนี้ แสดงขั้นตอนการส่งข้อความดังรูปที่ 3.17

- 3.9.3.1 ใช้คำสั่ง AT ถ้าคอมพิวเตอร์สามารถติดต่อโมดูลจีเอสเอ็มได้จะตอบ OK กลับมา
- 3.9.3.2 ใช้คำสั่ง AT+CMGF=1 เพื่อเลือกรูปแบบการส่งข้อความแบบ Text Mode
- 3.9.3.3 ใช้คำสั่ง AT+CMGS="+66 แล้วตามด้วยเบอร์ผู้รับ"
- 3.9.3.4 เมื่อมีเครื่องหมาย > ส่งกลับมา ก็ทำการพิมพ์ข้อความต่างๆที่ต้องการส่งออกไป
- 3.9.3.5 ปิดท้ายข้อความด้วยรหัส "Ctrl+Z"

```

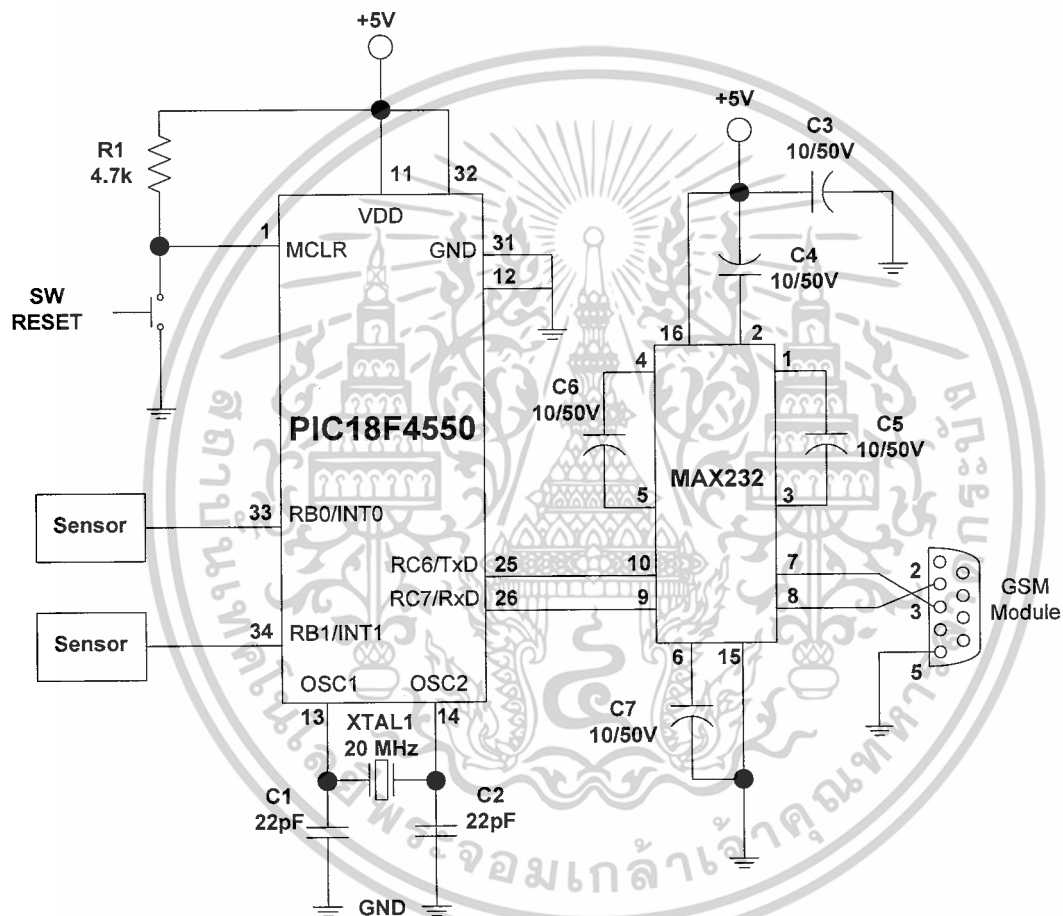
    TEST Sending SMS - HyperTerminal
    File Edit View Call Transfer Help
    [Icons]
    AT
    OK
    AT+CMGF=1
    OK
    AT+CMGS="+66859539175"
    > Test Sending SMS
    +CMGS: 1
    OK
    Connected 00:01:59 Auto detect 4800 8-N-1 SCROLL CMGS [ESC] [ENTER]
  
```

รูปที่ 3.17 ขั้นตอนการส่ง SMS ด้วยชุดคำสั่ง AT Command ผ่านทางโปรแกรม Hyper Terminal

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11 ภาพรวมของวงจรที่ใช้ในระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด

เนื่องจากโมดูลจีเอสเอ็มนั้น ต้องการระดับสัญญาณที่เป็น RS232 แต่สัญญาณที่ส่งออกมาจากไมโครคอนโทรลเลอร์พีไอซี (PIC) เป็นระดับสัญญาณ TTL เพราะฉะนั้นการเชื่อมต่อไมโครคอนโทรลเลอร์พีไอซี (PIC) กับโมดูลจีเอสเอ็มนั้น จะต้องใช้ไอซี MAX232 ต่อระหว่างอุปกรณ์ทั้งสอง โดยที่ไอซี MAX232 จะทำหน้าที่แปลงระดับสัญญาณของ RS-232 มาเป็นระดับ TTL และในทำนองเดียวกันก็แปลงระดับสัญญาณ TTL ไปเป็นระดับสัญญาณ RS-232 แสดงภาพวงจรการเชื่อมต่อดังรูปที่ 3.18

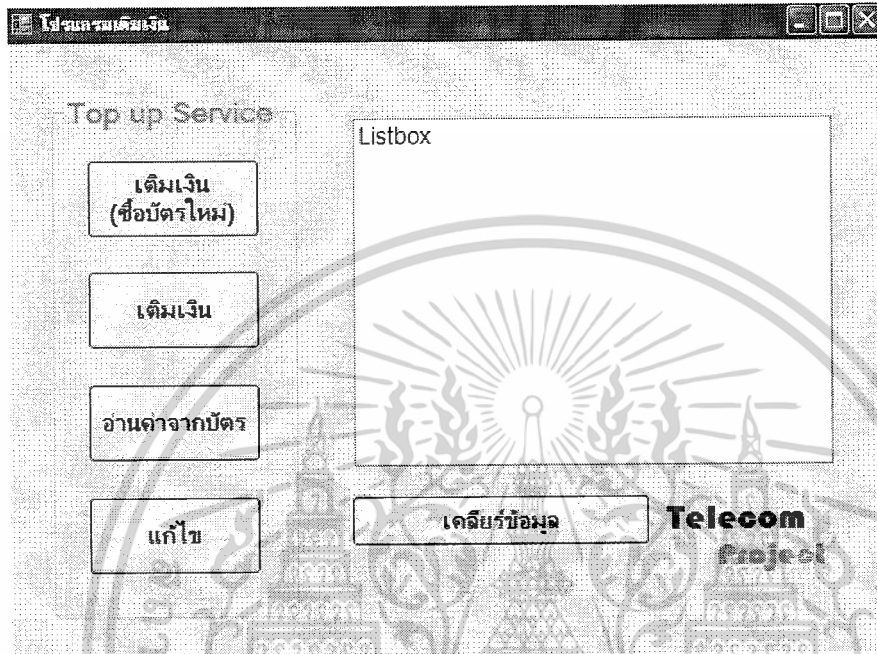


รูปที่ 3.18 ภาพรวมของวงจรที่ใช้ในระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.12 การออกแบบโปรแกรมเติมเงิน

การออกแบบในส่วนนี้จะเป็นการออกแบบโดยใช้ซอฟต์แวร์ โดยจะใช้โปรแกรม Visual C# ซึ่งโปรแกรมที่สร้างขึ้นมา สามารถที่จะเพิ่มเงิน หักเงิน แก้ไข และเคลียร์ค่าภายในบัตร (อาร์เอฟไอดีแท็ก) ได้ตามจำนวนที่ต้องการ แสดงดังรูปที่ 3.19, 3.20 และ 3.21 ตามลำดับ

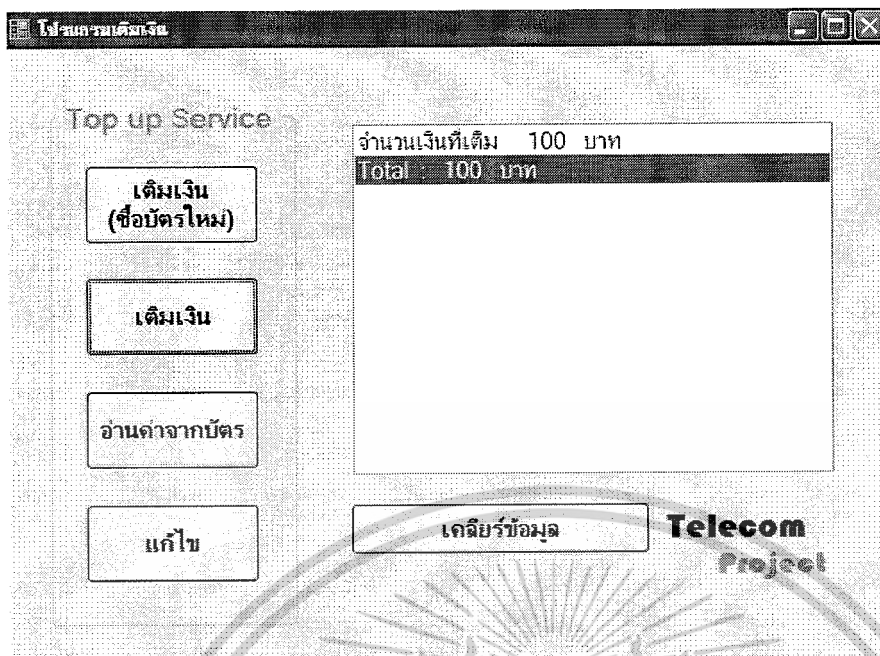


รูปที่ 3.19 ส่วนของโปรแกรมเติมเงินที่ได้ออกแบบ



รูปที่ 3.20 วิธีการเติมเงินใส่บัตร (อาร์เอฟไอดีแท็ก)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.21 ภาพเมื่อมีการเติมเงินใส่ในบัตร (อาร์เอฟไอดีแท็ก) แล้ว

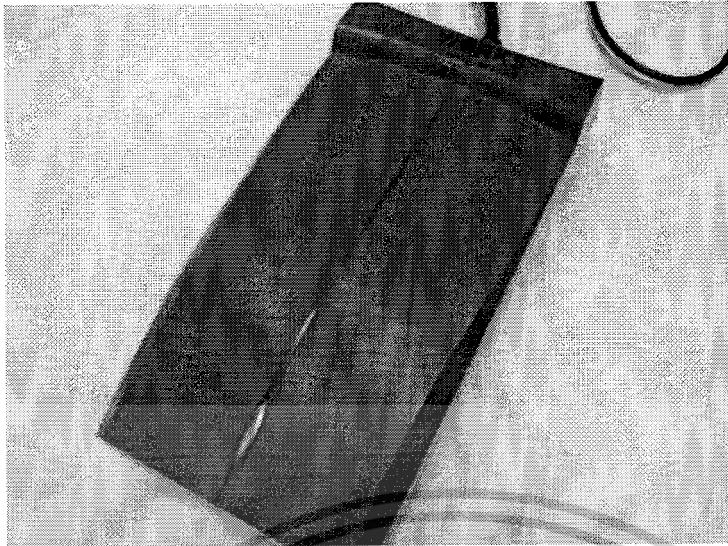
3.12.1 ส่วนประกอบของเครื่องอ่าน/เขียนอาร์เอฟไอดี

ส่วนประกอบของเครื่องอ่าน/เขียนอาร์เอฟไอดีที่ใช้ในโปรแกรมเติมเงิน เป็นของบริษัท IT Works รุ่น RFID-M1 ดังรูป 3.22 คุณสมบัติแสดงดังตารางที่ 3.3

ตารางที่ 3.3 ข้อมูลของเครื่องอ่าน/เขียน RFID-M1

ข้อมูล	รายละเอียด
ส่วนเชื่อมต่อเครื่องคอมพิวเตอร์	USB 1.1
มาตรฐานการทำงาน	ISO14443 Type A
ความถี่คลื่นสัญญาณ	13.56 เมกกะเฮิร์ต
อัตราการส่งข้อมูล	106 กิโลบิตต่อวินาที
มาตรฐานอุปกรณ์ในการบรรจุข้อมูล	Mifare Classic, Mifare UltraLight, Mifare Pro(X) Mifare DESFire และ ISO14443A
การแสดงผล	หลอดไฟแอลอีดีและเสียงเตือน
ความยาวสาย USB	1.2 เมตร
แหล่งจ่ายไฟ	ผ่านทาง USB
กระแสไฟฟ้า	ไม่เกิน 120 มิลลิแอมป์
ระยะทางสูงสุดในการอ่านข้อมูล	10 เซนติเมตร

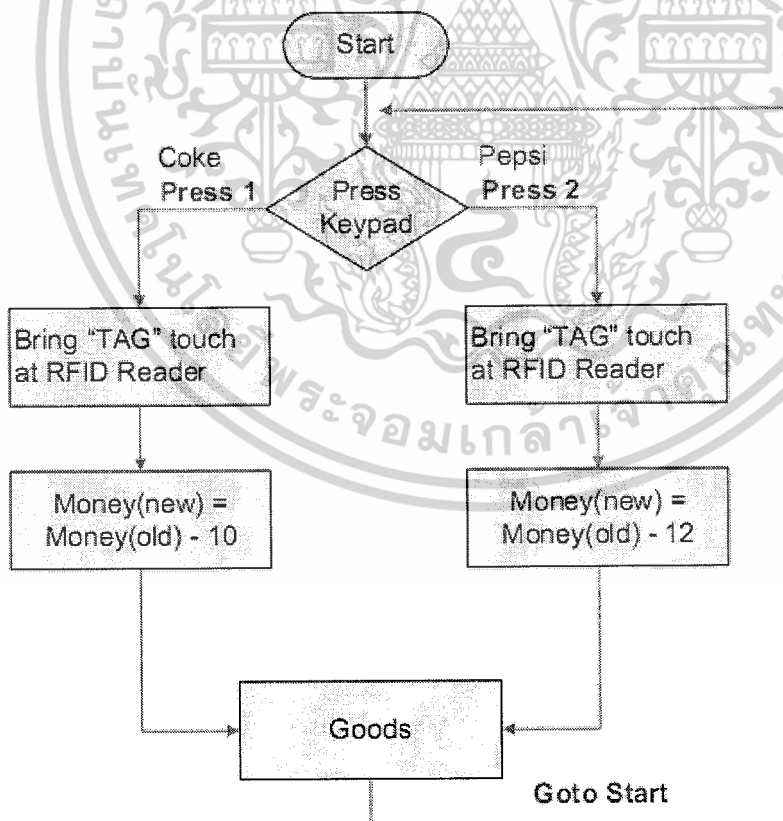
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.22 เครื่องอ่าน/เขียนอาร์เอฟไอดี รุ่น RFID-M1

3.13 โฟลว์ชาร์ตของระบบการซื้อสินค้าที่ผู้ขายของผ่านอาร์เอฟไอดี

3.13.1 โฟลว์ชาร์ตในส่วนของการซื้อสินค้า

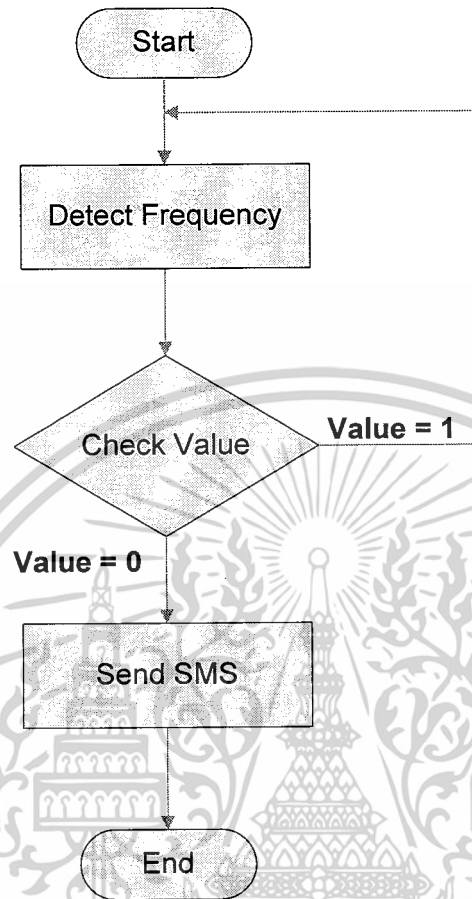


รูปที่ 3.23 โฟลว์ชาร์ตส่วนของการซื้อสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.13.2 โฟลว์ชาร์ตในส่วนของการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด



รูปที่ 3.24 โฟลว์ชาร์ตส่วนของการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

จากการทดลองใช้งานระบบตู้ขายสินค้าที่ใช้เทคโนโลยีของอาร์เอฟไอดี ซึ่งในโครงการนี้จะแบ่งออกเป็น 3 ส่วน ดังนี้

1. ส่วนของระบบการซื้อสินค้าด้วยเทคโนโลยีอาร์เอฟไอดี
2. ส่วนของการแจ้งเตือนผ่านข้อความสั้น (SMS) เมื่อสินค้าภายในตู้ขายของหมด
3. ส่วนของโปรแกรมเติมเงิน

4.1 ผลการทดลองจากการใช้งานระบบการซื้อสินค้าผ่านอาร์เอฟไอดี

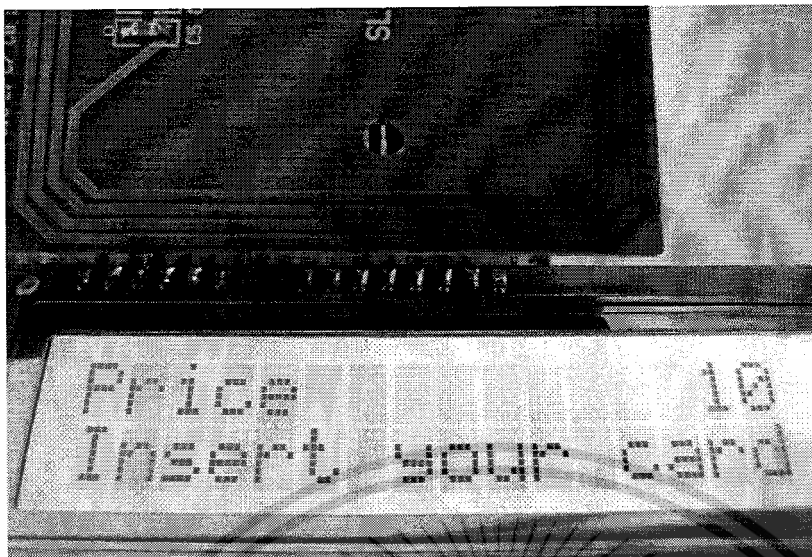
1. เมื่อยังไม่มีการใช้งานระบบ หน้าจอ LCD ที่ตู้จะแสดงข้อความ “KMITL SOFT DRINK SELECTS ITEM” แสดงดังรูปที่ 4.1



รูปที่ 4.1 หน้าจอ LCD เมื่อยังไม่มีการใช้งาน

2. เมื่อมีผู้ใช้งานมาใช้งานและทำการเลือกสินค้าจากตู้ขายสินค้า หน้าจอ LCD จะขึ้นแสดงข้อความบอกราคาของสินค้า และก็จะบอกให้ผู้ให้บริการนำ RFID Tag มาแตะเพื่อทำการหักเงิน โดยในโครงการนี้เราได้นำ Touch Sim เข้ามาใช้แทน RFID Tag ซึ่งในการทดลองนี้จะมีสินค้าอยู่ 2 ชนิด ให้ผู้ให้บริการเลือกซื้อ และสินค้าทั้ง 2 ชนิด ก็จะมีราคาแตกต่างกัน แสดงดังรูปที่ 4.2 และ 4.3 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 เมื่อผู้ใช้บริการเลือกสินค้าชนิดที่ 1



รูปที่ 4.3 เมื่อผู้ใช้บริการเลือกสินค้าชนิดที่ 2

3. จากนั้นผู้ใช้บริการจะต้องนำ Touch Sim มาแตะที่เครื่องอ่าน/เขียนอาร์เอฟไอดี เพื่อให้ระบบทำการหักเงิน และเมื่อทำการหักเงินแล้วก็จะแสดงยอดเงินคงเหลือใน Touch Sim ให้ผู้ใช้บริการได้ทราบ และทำการจ่ายสินค้า แสดงดังรูปที่ 4.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



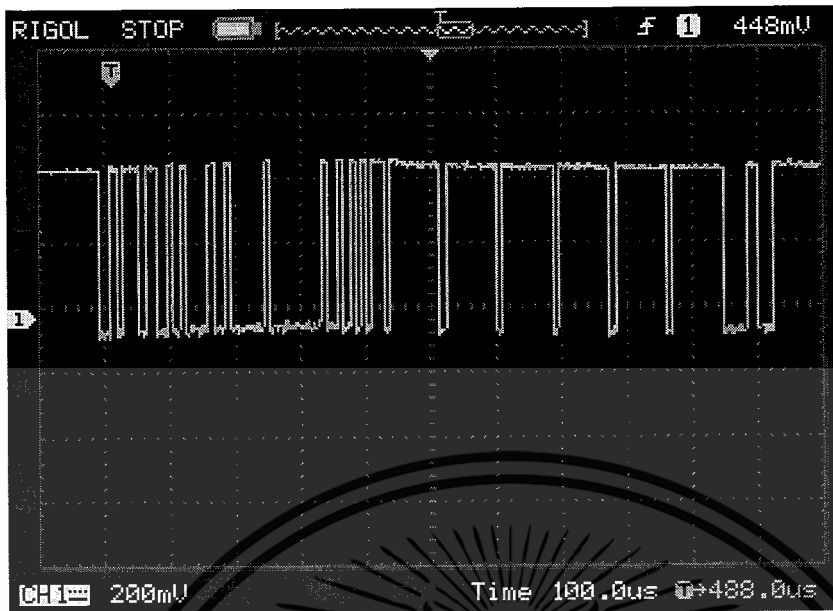
รูปที่ 4.4 รูปเมื่อผู้ใช้บริการนำ Touch Sim มาแตะที่เครื่องอ่าน/เขียนอาร์เอฟไอดีเพื่อทำการหักเงินและจ่ายสินค้า

4. ผลทางสัญญาณไฟฟ้าของ RFID เพื่อดูสัญญาณที่ขา Tx และ Rx ของ RFID Module ซึ่งสัญญาณเป็นดังรูปที่ 4.5 และ 4.6 ตามลำดับ ซึ่งได้จากการวัดของเครื่อง RFID Mifare Read/Write Module SL015M



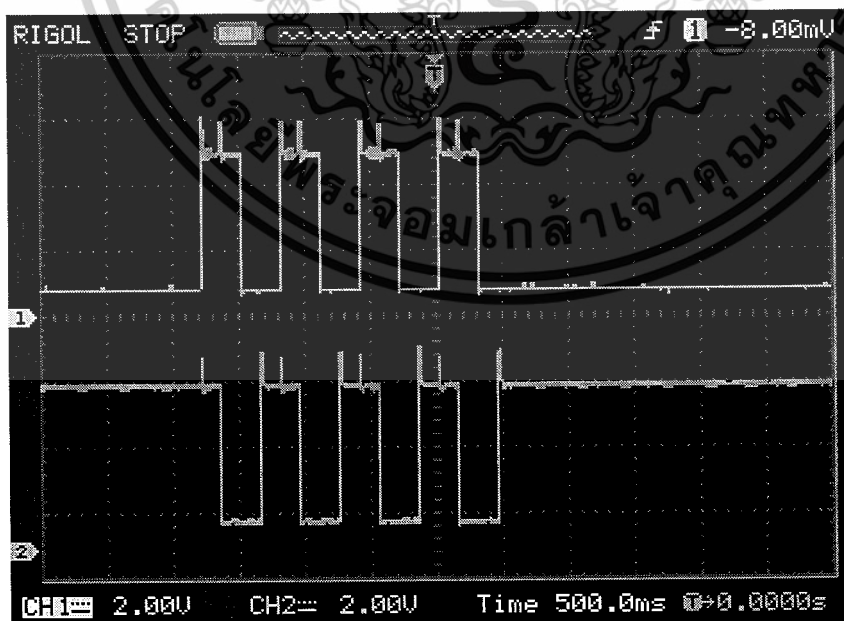
รูปที่ 4.5 สัญญาณที่ขา Tx ของ RFID module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



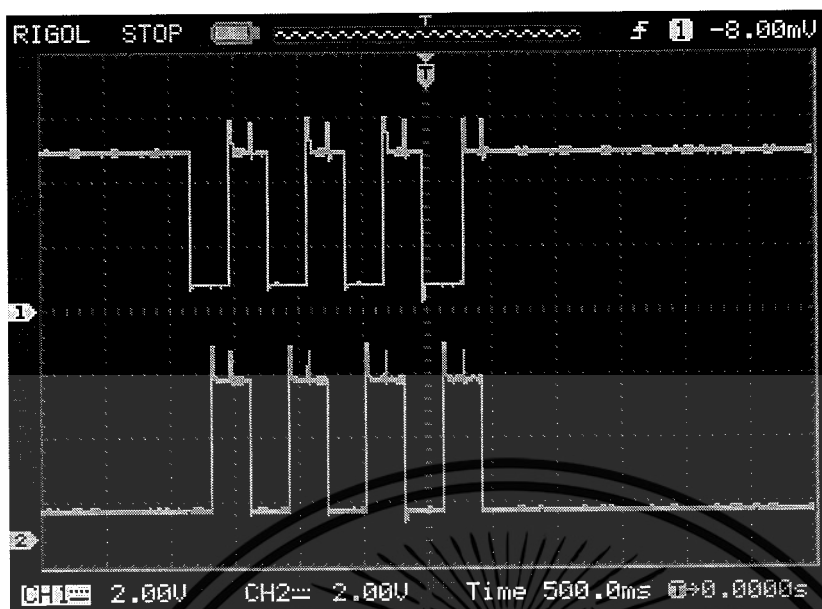
รูปที่ 4.6 สัญญาณที่ขา Rx ของ RFID Module

5. เมื่อทำการหักเงินใน Touch Sim เรียบร้อยแล้วจากนั้น สเต็ปมอเตอร์ของสินค้าที่ได้มีการสั่งซื้อก็จะทำการหมุนเพื่อจ่ายสินค้า โดยในส่วนนี้ทางผู้จัดทำได้เขียน โปรแกรมเพื่อให้สเต็ปมอเตอร์ทำงานแบบ 2 เฟส เก็บภาพสัญญาณที่สเต็ปมอเตอร์ทั้ง 4 เส้น ได้ดังรูปที่ 4.7 และ 4.8 โดยในรูปเป็นการเขียนให้มอเตอร์หมุนแบบตามเข็มนาฬิกา



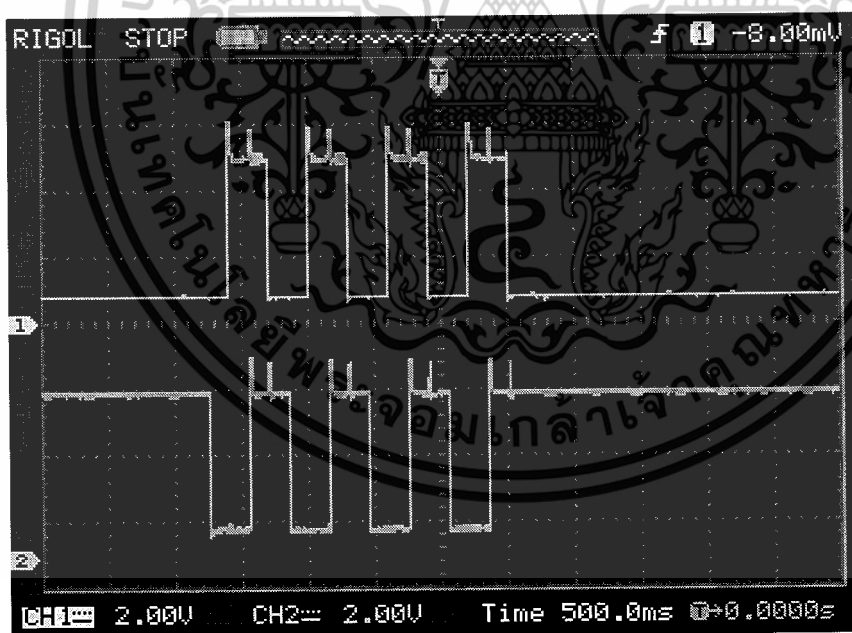
รูปที่ 4.7 สัญญาณที่ขา 16 และ 15 ตามลำดับของ ULN 2003 แบบหมุนตามเข็มนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



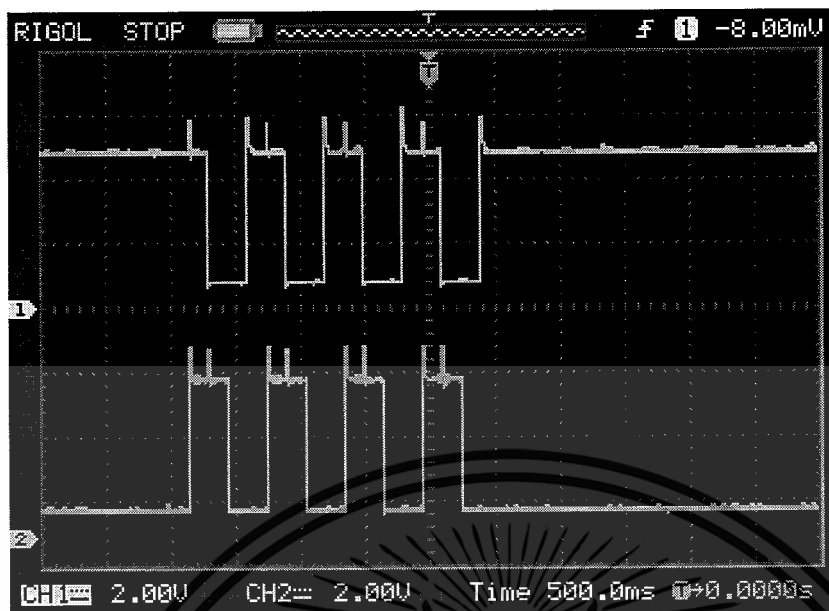
รูปที่ 4.8 สัญญาณที่ขา 14 และ 13 ตามลำดับของ ULN 2003 แบบหมุนตามเข็มนาฬิกา

รูปที่ 4.9 และ 4.10 ตามลำดับ นั้นจะเป็นรูปที่เขียน โปรแกรมเพื่อให้มอเตอร์หมุนทวนเข็มนาฬิกา



รูปที่ 4.9 สัญญาณที่ขา 16 และ 15 ตามลำดับของ ULN 2003 แบบหมุนทวนเข็มนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

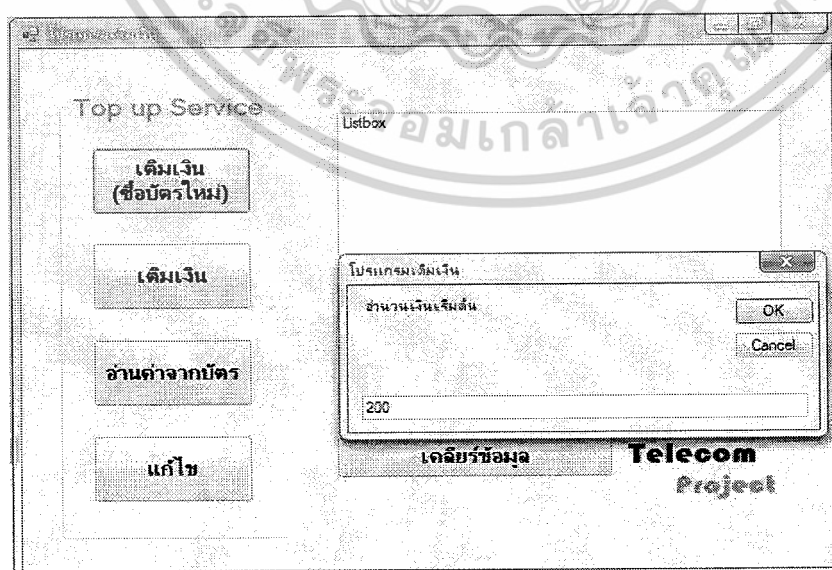


รูปที่ 4.10 สัญญาณที่ขา 14 และ 13 ตามลำดับของ ULN 2003 แบบหมุนทวนเข็มนาฬิกา

4.2 การทดลองใช้งานโปรแกรมเติมเงิน

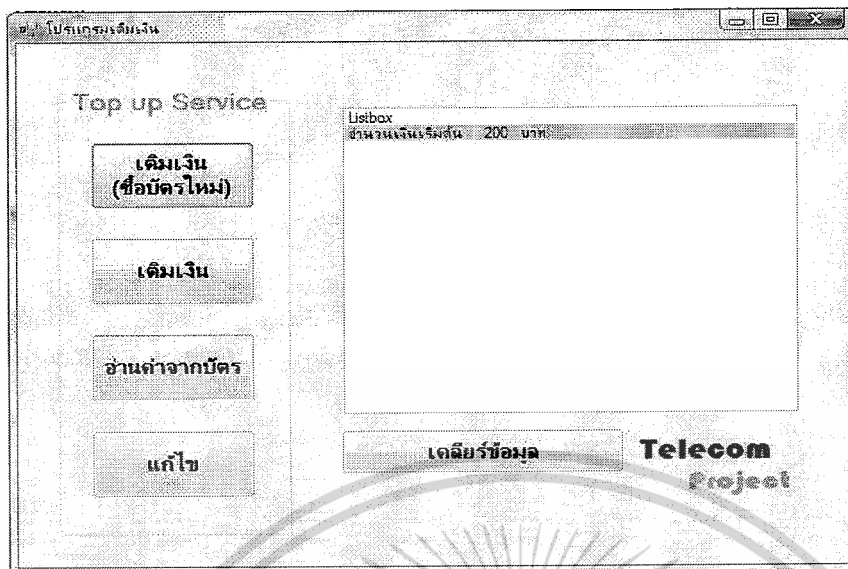
ในโครงการนี้ได้ออกแบบโปรแกรมที่ไว้ใช้สำหรับเติมเงินให้ Tag ซึ่งใช้โปรแกรม Visual C# ในการเขียนโปรแกรม และจากการที่ทดสอบโปรแกรมที่เขียน ได้ดังนี้

1. ทดลองการเติมเงินค่าเริ่มต้นของ Touch Sim แสดงดังรูปที่ 4.11 และ 4.12



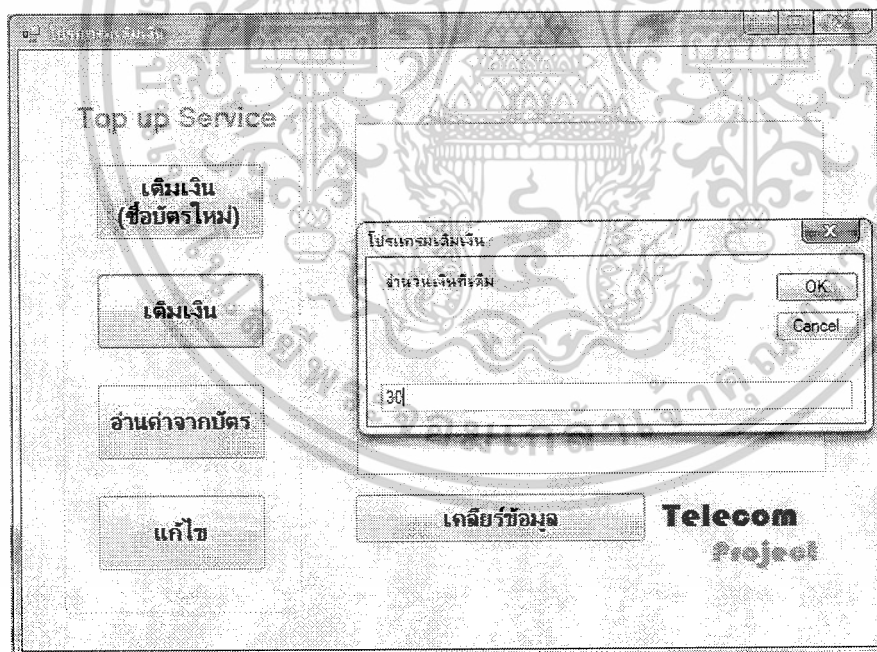
รูปที่ 4.11 ขั้นตอนการเติมเงินเริ่มต้นให้ Touch Sim

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



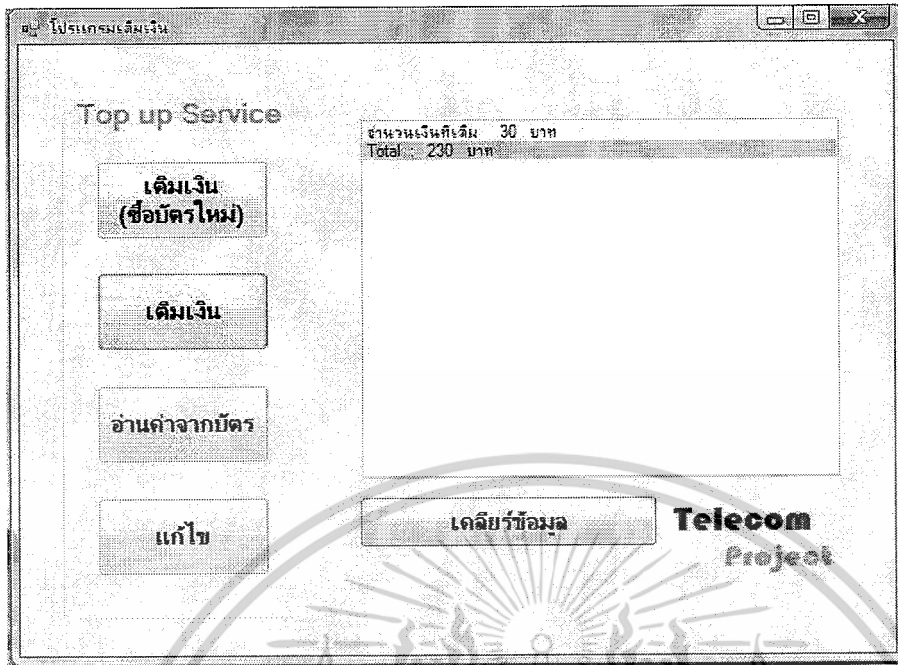
รูปที่ 4.12 ผลที่ได้จากการเติมเงินเริ่มต้น

2. ทดลองเติมเงินใส่ Touch Sim ตัวโปรแกรมก็จะเติมเงินเพิ่มเข้าไปจากจำนวนเงินเดิมที่เหลืออยู่ใน Touch Sim และจะโชว์จำนวนเงินทั้งหมดใน Touch Sim ให้ได้ทราบ แสดงดังรูปที่ 4.13 และ 4.114



รูปที่ 4.13 ขั้นตอนการเติมเงินเพิ่มใส่ Touch Sim

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 ผลลัพธ์จากการเติมเงินเพิ่มใส่ Touch Sim

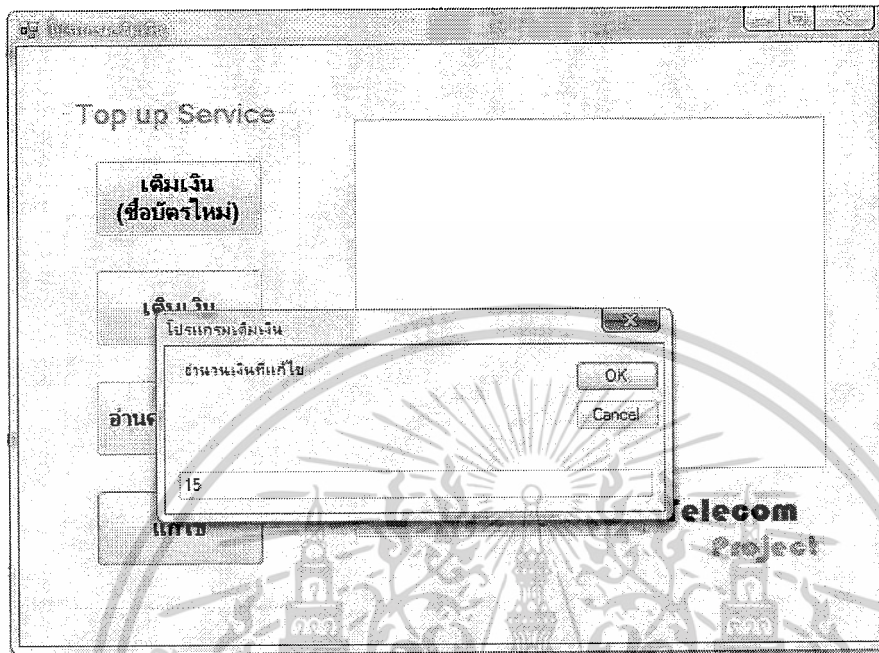
3. ทดลองการอ่านค่าจาก Touch Sim จากโปรแกรม แสดงดังรูปที่ 4.15



รูปที่ 4.15 ผลจากการอ่านค่า Touch Sim

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.ทดลองการแก้ไขข้อมูลใน Touch Sim กรณีที่มีการเติมเงินใส่ Touch Sim เข้าไปมากเกินไป ในที่นี้เราสมมติให้มีการเติมเงินเกินไป 15 บาท แสดงดังรูปที่ 4.16 และ 4.17



รูปที่ 4.16 การแก้ไขข้อมูลใน Touch Sim กรณีที่เติมเงินเกินไป 15 บาท



รูปที่ 4.17 ผลลัพธ์ที่ได้จากการแก้ไขข้อมูล

หมายเหตุ: ส่วนปุ่ม “เคลียร์ข้อมูล” นั้นไว้ใช้สำหรับการเคลียร์หน้าต่างสถานะที่แสดงข้อมูลที่หน้าโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

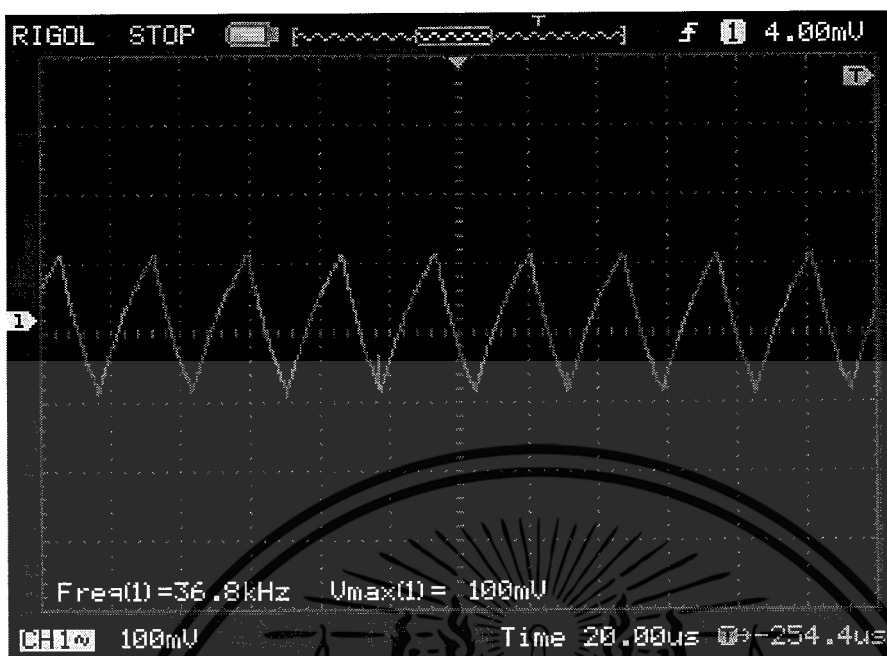
4.3 การทดลองของระบบแจ้งเตือนสินค้าหมดผ่านทางข้อความสั้น (SMS: Short Message Service)

การทำงานของส่วนนี้จะเริ่มจาก IC 555 ทำการสร้างความถี่ที่ 38 kHz แสดงดังรูปที่ 4.18 แล้วก็จะส่งความถี่นี้ไปให้แอลอีดีอินฟราเรด ทำการส่งความถี่นี้ไปยังตัวรับอินฟราเรด โดยเมื่อมีสินค้าอยู่ในตู้ขายสินค้า สินค้าจะบังสัญญาณความถี่นี้ไม่ทำให้สามารถส่งไปถึงยังฝั่งรับได้ สัญญาณที่ไมโครคอนโทรลเลอร์ก็จะมีค่าเป็น 1 แต่ถ้าหากเมื่อสินค้าหมด ก็จะสามารถส่งสัญญาณไปยังตัวรับได้ สัญญาณที่ไมโครคอนโทรลเลอร์จะมีค่าเป็น 0 เมื่อไมโครคอนโทรลเลอร์ได้รับค่า 0 มากก็จะทำการสั่งให้โมดูลจีเอสเอ็ม (GSM module) ทำการส่งข้อความไปยังผู้ประกอบการเพื่อแจ้งให้ทราบว่าสินค้าหมดแล้ว แสดงรูปสัญญาณเมื่อมีสินค้าและไม่มีสินค้านี้ดังรูปที่ 4.19, 4.20 และ 4.21 ตามลำดับ



รูปที่ 4.18 สัญญาณความถี่ที่ 38 kHz จาก IC555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

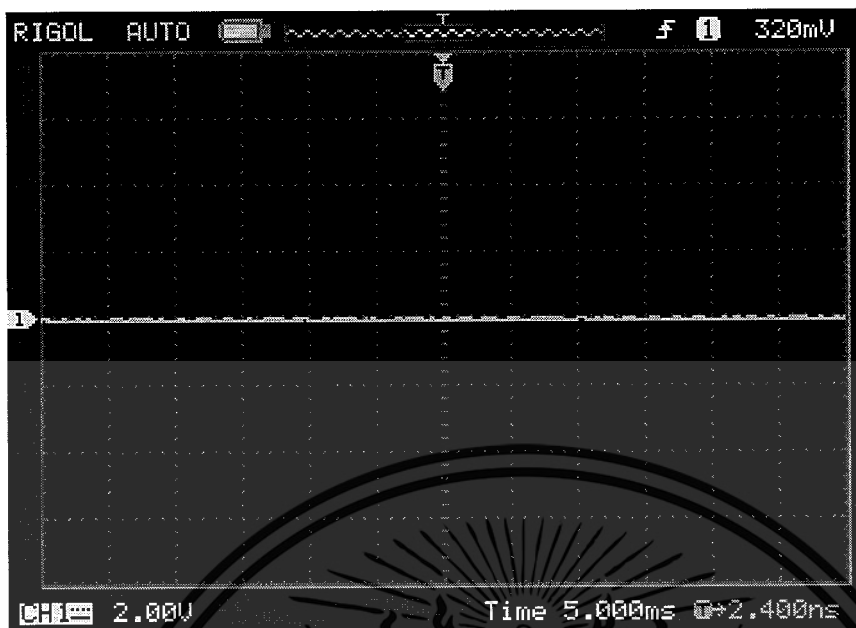


รูปที่ 4.19 สัญญาณการเก็บประจุและคายประจุของคาร์ปาซิเตอร์



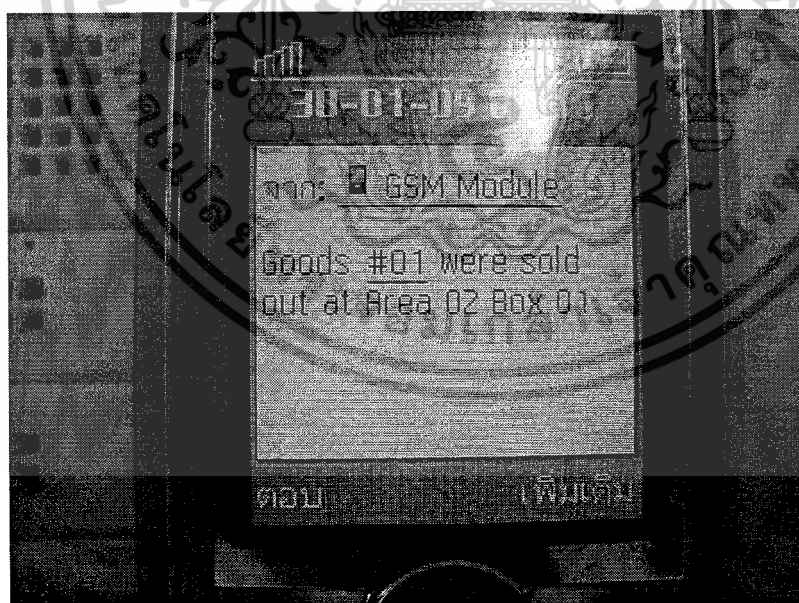
รูปที่ 4.20 สัญญาณเป็น 1 เมื่อยังมีสินค้าคงเหลือภายในตู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



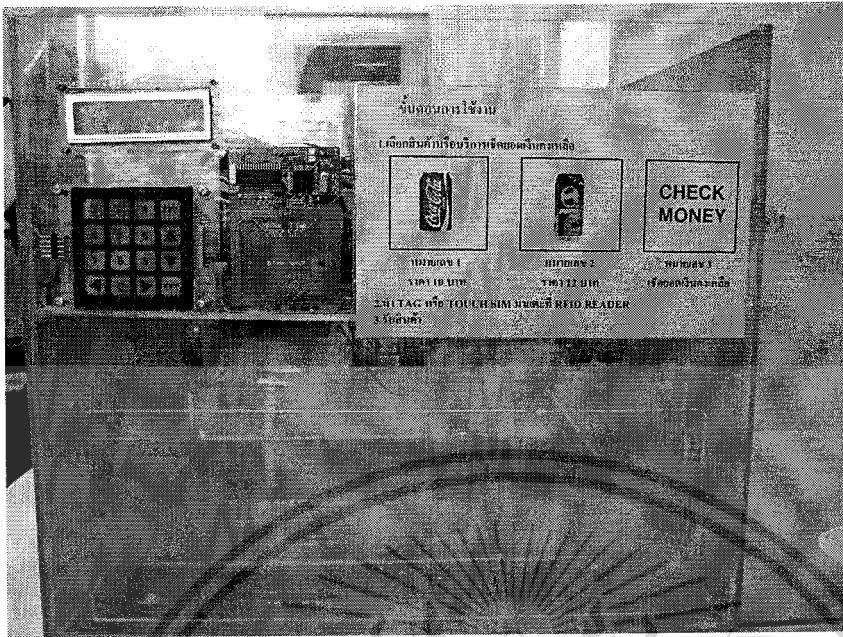
รูปที่ 4.21 สัญญาณเป็น 0 เมื่อสินค้าภายในตู้หมด

เมื่อสินค้าหมด GSM Module จะส่งข้อความสั้นแจ้งเตือนไปยังมือถือผู้ประกอบการ โดยในข้อความจะบอกถึงสินค้าชนิดใดหมด ตู้สินค้าอยู่ในพื้นที่ใด และตู้หมายเลขใด ดังแสดงในรูป 4.22

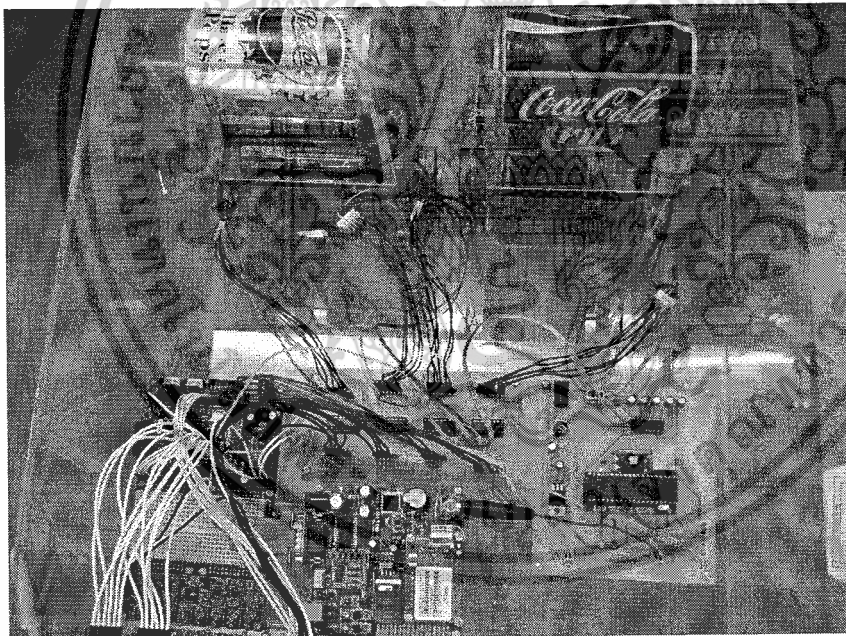


รูปที่ 4.22 ข้อความที่ได้รับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 ตู้ขายของอัตโนมัติ (Vending Machine)



รูปที่ 4.24 ตู้ขายของอัตโนมัติเมื่อมองจากด้านบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุป

5.1 สรุปผลการทดลอง

จากโครงการนี้ได้ทำการศึกษาและทำการทดลองทำระบบการซื้อขายของผ่านตู้ขายสินค้าอัตโนมัติโดยในโครงการนี้ได้แบ่งส่วนของการสร้างออกเป็น 3 ส่วน ประกอบด้วย ระบบการซื้อสินค้าที่เครื่องขายสินค้าอัตโนมัติโดยใช้เทคโนโลยีอาร์เอฟไอดี ระบบการแจ้งเตือนเมื่อสินค้าภายในตู้ขายของหมด และระบบเติมเงิน

ในส่วนของระบบการซื้อสินค้าที่เครื่องขายสินค้าอัตโนมัติโดยใช้เทคโนโลยีอาร์เอฟไอดีนั้น ได้ทำการทดลองการใช้งานของระบบ โดยเริ่มต้นเมื่อผู้ใช้บริการเข้ามายังตู้ขายสินค้า ที่จอ LCD จะแสดงข้อความเพื่อบอกให้ผู้ใช้บริการทำการกดเลือกซื้อสินค้า จากนั้นเมื่อผู้ใช้บริการกดเลือกสินค้าแล้ว ระบบก็จะแจ้งราคาสินค้าให้ทราบว่าเป็นจำนวนเงินเท่าไร และบอกให้ผู้ใช้บริการนำ RFID Tag หรือ Touch Sim มาแตะที่เครื่อง RFID Reader เพื่อทำการหักยอดเงิน เมื่อระบบทำการหักยอดเงินเสร็จเรียบร้อยแล้ว ระบบก็จะแสดงยอดเงินคงเหลือภายใน RFID Tag หรือ Touch Sim ของผู้ใช้บริการให้ได้ทราบ พร้อมทั้งตู้ขายสินค้าก็จะจ่ายสินค้าออกมาให้ เมื่อเสร็จสิ้นระบบก็จะวนกลับไปหน้าจอเริ่มต้นใหม่อีกครั้ง เพื่อรอรับการใช้งานจากผู้ใช้บริการในครั้งต่อไป

ส่วนของระบบการแจ้งเตือนเมื่อสินค้าภายในตู้หมด เราได้ทำการทดสอบ โดย ทำการซื้อสินค้าจากตู้ขายสินค้าจนเมื่อสินค้าภายในตู้ขายหมด จากการทดลองพบว่าในส่วนของเซนเซอร์เมื่อทำการตรวจพบว่าสินค้าภายในตู้ขายสินค้าหมด ก็ได้ทำการส่งให้โมดูลจีเอสเอ็มทำการส่งข้อความไปยังผู้ประกอบการได้ทราบว่าสินค้าประเภทใดหมด และมีที่ตั้งของตู้ขายสินค้าอยู่ที่ไหน สามารถใช้งานได้จริง

และในส่วนของระบบเติมเงิน โดยได้ใช้โปรแกรม Visual C# ในการออกแบบและเขียนโปรแกรมเพื่อใช้ในการเติมเงิน หักเงิน แก่ใจจำนวนเงินภายใน RFID Tag หรือ Touch Sim เมื่อทดลองใช้งานก็สามารถใช้งานโปรแกรมได้ดี สามารถหักเงิน เติมเงิน แก่ใจข้อความ ได้ถูกต้องตามที่ต้องการ

5.2 ปัญหาที่พบ

1. การเชื่อมต่อไมโครคอนโทรลเลอร์กับ RFID Reader นั้นมีความซับซ้อนทำให้เกิดปัญหาในการเขียนโปรแกรมที่จะทำการติดต่อให้ Reader ทำงานได้ตรงตามเป้าหมายได้อย่างถูกต้องและแม่นยำ
2. ระบบการแจ้งเตือนผ่านทาง SMS ต้องใช้เวลานานและมีข้อผิดพลาดในการส่ง SMS ไปยังผู้ประกอบการเมื่อสินค้าหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ได้สำเร็จลุล่วงไปได้ด้วยดี นั้นเนื่องจากข้าพเจ้าได้รับคำแนะนำ และคำปรึกษาที่ดีจาก รศ.ดร. ไกรสิน ส่องวัฒนา และ ผศ.ดร.พิเชฐ ม่วงนวล ซึ่งเป็นอาจารย์ที่ปรึกษา ข้าพเจ้ารู้สึกซาบซึ้งในความอนุเคราะห์จากท่านอาจารย์ทั้ง 2 ท่าน และขอขอบพระคุณเป็นอย่างยิ่ง

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิศวกรรม โทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณเพื่อนๆภาควิชาวิศวกรรม โทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำ ช่วยเหลือ และคอยเป็นกำลังใจให้เสมอมา

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกๆเรื่องทำให้ข้าพเจ้าทำวิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี



คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

ประจัน พลังสันติกุล. 2537. PIC Works Examples and C Source Code. พิมพ์ครั้งที่ 1. กรุงเทพฯ๑

สัจจะ จรัสรุ่งรวิวรร. 2550. คู่มือ Visual C# 2005 ฉบับสมบูรณ์. พิมพ์ครั้งที่ 1. นนทบุรี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// Code Program Vending Machine
// Hardware : LCD, Keypad, Step Motor and Rfid sl015M Module
// Interface with AVR ATmega64
```

```
#include <avr/io.h>
#include <avr/signal.h>
#include <avr/interrupt.h>
#include <avr/wdt.h>
#include <stdio.h>
#include <stdarg.h>
#include <string.h>
#include <stdlib.h>
```

```
#include "mifare.h"
#include "lcd.h" // library
#include "serial.h" // library
#include "printf.h" // library
#include "mydefs.h" // library
#include <util/delay.h> // library
```

```
char ch;
char g_cErr;
char g_cStatus;
int state=0; // Set State equal 0
```

```
uint8_t g_ucCardType;
```

```
uint8_t fs2p_M1[4] = {0x93,0xC6,0x6C,0x39}; // 2 Phase Stepping Motor 1
uint8_t fs2p_M2[4] = {0x93,0xC6,0x6C,0x39}; // 2 Phase Stepping Motor 2
```

```
void init_key (void); // Initial Keypad Function
unsigned char kbd_getc (void); // keypad Function
```

```
//----- Step Motor 1 & 2 Config -----//
```

```
#define STEP_M_POUT PORTF
#define STEP_M_DDR DDRF
#define STEP_M_DDR_OUT ((1<<DDF7) | (1 << DDF6) | (1 << DDF5) | (1 << DDF4)
| (1 << DDF3) | (1 << DDF2) | (1 << DDF1) | (1 << DDF0))
```

```
//-----//
```

```
//----- Step Motor 3 & 4 Config -----//
```

```
#define STEP_M2_POUT PORTE
#define STEP_M2_DDR DDRE
#define STEP_M2_DDR_OUT ((1<<DDE7) | (1 << DDE6) | (1 << DDE5) | (1 <<
DDE4) | (1 << DDE3) | (1 << DDE2) | (1 << DDE1) | (1 << DDE0))
```

```
//-----//
```

```
//----- Keypad Config -----//
```

```
#define KEY_DATA_POUT PORTB
#define KEY_DATA_PIN PINB
#define KEY_DATA_DDR DDRB
#define KEY_DATA_DDR_OUT ((1<<DDB7) | (1 << DDB6) | (1 << DDB5) | (1 <<
DDB4))
```

```
const unsigned char kbd_pad[4][3] = {
    {'1','2','3'},
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```

                                {'7','8','9'},
                                {'*','0','#'}
                                };

//-----//

//----- Initial keypad -----//

void init_key (void)
{
    KEY_DATA_DDR = KEY_DATA_DDR_OUT;
}

//-----//

//----- Scan keypad -----//

unsigned char kbd_getc (void)
{
    // x[110]1111, x[101]1111, x[011]1111

    unsigned char read_col[3] = {0x6F, 0x5F, 0x3F};
    int col, row;

    static char last_key;

    for (col = 0; col < 3; col++)
    {
        KEY_DATA_POUT = read_col[col];
        _delay_ms (10);
        row = KEY_DATA_PIN & 0x0F;

        switch (row) {
            case 0x0E : row = 0;
                        break;
            case 0x0D : row = 1;
                        break;
            case 0x0B : row = 2;
                        break;
            case 0x07 : row = 3;
                        break;

            default : row = 4;
        }

        };

        if (row != 4)
        {
            if (kbd_pad[row][col] == last_key)
                return '\0';

            last_key = kbd_pad[row][col];
            return kbd_pad[row][col];
        }
    }

    last_key = '\0';
    return last_key;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
//-----//

//----- Debug Function -----//

void debug (uint8_t *buf)
{
    int n;
    printf ("\r\n%02X",buf[0]);
    for (n = 1; n < 10;n++)
        printf (" %02X",buf[n]);
}

//-----//

//----- Delay Function -----//

void delaytime (unsigned int k)
{
    for (; k > 0; k--)
        _delay_ms(1);          // Delay 1 ms
}

//-----//

//----- fw step1 Function -----//

void fw_step (uint8_t step, uint8_t d1)
{
    unsigned char n = 0 ;
    while (step--)
    {
        STEP_M_POUT = fs2p_M1[n];
        if (n++ == 3)
            n = 0;
        delaytime(d1);
    }
}

//-----//

//----- fw step2 Function -----//

void fw_step2 (uint8_t step, uint8_t d1)
{
    unsigned char n = 0;
    while (step--)
    {
        STEP_M2_POUT = fs2p_M2[n];
        if (n++ == 3)
            n = 0;
        delaytime(d1);
    }
}

//-----//
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
//----- Check Timeout -----//

uint8_t Check_timeout (void)
{
    uint32_t    timeout = 10;    // Check Time 10 second
    uint8_t     t;
    uint8_t     ret = 0;

    while (timeout>0)
    {
        if(check_mifare())
        {
            t = ret ;
            return 0;
        }
        else
        {
            timeout--;
            delaytime (1000);
        }
    }

    return 1;
}

//-----//

//----- Check Card -----//

void checkCard(void)
{
    uint8_t buf[25];
    uint8_t c;

    if (check_mifare ())
    {
        printf ("\r\ncard Activate\r\n"); // Test Send data
                                           // to Hyper Terminal
        mifare_send_command (SelectCard,3); // Select Mifare Card

        c = get_dat_frommifare (buf); // Receive data From RFID Module
        printf ("\r\nCounter Recived = %d\r\n",c);
        #ifdef PRINT_DEBUG
        debug(buf);
        #endif
        if (buf[1] == 0x08)
            g_ucCardType = buf[8]; // For this project, Mifare 1 K,
                                   // Touch Sim (Mifare Pro Card)
        else
            g_ucCardType = buf[11]; // Another Mifare Card which
                                     // Module can support
        _delay_ms (500); // Delay 500 ms
        switch(g_ucCardType)
        {
            case 1: printf ("\r\nMifare_std_1k"); // Mifare 1 K
                    g_cStatus = discount_coke (); // Goto Decrease
                                                    // Coke Function
                    printf("%ld",g_cStatus) ;
                    if (g_cStatus != 1)
                    {
                        g_cErr = 3;
                    }
        }
    }
}


```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```

state=2;
break;

Card
    case 2: printf ("\r\nMifare_Pro"); // Mifare Pro
           // or Touch Sim
           g_cStatus = discount_coke (); // Goto Decrease
           // Coke Function
           if (g_cStatus != 1)
           {
               g_cErr = 4;
           }
           state=2;
           break;

    default: g_cErr = 3;
             break;
}
}
}

//-----//
//----- check card 1 -----//

void CheckCard1(void)
{
    uint8_t buf[25];
    uint8_t c;

    if (check_mifare ())
    {
        printf ("\r\ncard Activate\r\n");
        mifare_send_command (SelectCard,3); // Select Mifare Card

        c = get_dat_frommifare (buf);
        printf ("\r\nCounter Recived = %d\r\n",c);
#ifdef PRINT_DEBUG
        debug(buf);
#endif

        if (buf[1] == 0x08)
            g_ucCardType = buf[8]; // For this project
        else
            g_ucCardType = buf[11]; // Another Mifare card
        _delay_ms (500);
        switch(g_ucCardType)
        {
            case 1: printf ("\r\nMifare_Std_1K"); // Mifare
                   //Standard 1 K
                   g_cStatus = discount_pepsi (); // Goto Decrease
                   // Pepsi

Function
            if (g_cStatus != 1)
            {
                g_cErr = 3;
            }
            state=4;
            break;

            case 2: printf ("\r\nMifare_Pro"); // Mifare Pro card
                   // or Touch Sim

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปยังผู้อื่นโดยไม่ผ่านการคัด

```

        g_cStatus = discount_pepsi (); // Goto Decrease
        // Pepsi Function
        if (g_cStatus != 1)
        {
            g_cErr = 4;
        }
        state=4;
        break;

    default:    g_cErr = 3;
                break;
}
}
}

```

//-----//

//----- Check Money From Card -----//

```

void CheckMon(void)
{

```

```

    uint8_t buf[25];
    uint8_t c;

```

```

    if (check_mifare ())
    {

```

```

        printf ("\r\ncard Activate\r\n");

```

```

        mifare_send_command (selectCard,3);

```

```

        c = get_dat_frommifare (buf);

```

```

        printf ("\r\nCounter Recived = %d\r\n",c);

```

```

#ifdef PRINT_DEBUG

```

```

    debug(buf);

```

```

#endif

```

```

    if (buf[1] == 0x08)

```

```

        g_ucCardType = buf[8]; // For this project

```

```

    else

```

```

        g_ucCardType = buf[11];

```

```

    _delay_ms (500);

```

```

    switch(g_ucCardType)
    {

```

```

        case 1: printf ("\r\nMifare_std_1k");
                g_cStatus = ReadwriteMifareStd ();

```

```

                if (g_cStatus != 1)

```

```

                {

```

```

                    g_cErr = 3;

```

```

                }

```

```

                state=6;

```

```

                break;

```

```

        case 2: printf ("\r\nMifare_Pro");
                g_cStatus = ReadwriteMifareStd ();
                if (g_cStatus != 1)

```

```

                {

```

```

                    g_cErr = 4;

```

```

                }

```

```

                state=6;

```

```

                break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```

        default:    g_cErr = 3;
                   break;
    }
}

//-----//

//----- Main Function -----//

int main(void)
{
    DDRD  |= (0 << PD0);           // Active input

    STEP_M_DDR = STEP_M_DDR_OUT;   // Set Port For Step motor 1 and 2
    STEP_M2_DDR = STEP_M2_DDR_OUT; // Set Port For Step motor 3 and 4

    ser_init();                    // Init serial port
    init_key();                    // Baudrate 115200
    init_uart1 ();                // Baudrate 115200
    init_lcd ();                  // Initial LCD Function
    sei();

    printf ("%c",'\f');           // Clear screen Hyper Terminal
    printf ("\r\nkmitl test mifare"); // Test Send data to Hyper Terminal

    mifare_send_command (reset,3); // Reset Board
    _delay_ms (100);              // Delay 100 ms

    while (1)
    {
        if(state == 0)
        {
            lcd_gotoxy (1,1);
            lcd_printf ("kmitl soft Drink");
            lcd_gotoxy (1,2);
            lcd_printf ("Select Item      ");

            ch = kbd_getc();
            if (ch == '1')           // Push 1 on Keypad
            {
                lcd_clear(1);
                lcd_clear(2);
                lcd_gotoxy(1,1);
                lcd_printf("Price          10"); // Buy Coke by card
                                                    // or touch sim

                lcd_gotoxy(1,2);
                lcd_printf("Insert your card");
                if(Check_timeout() == 0)
                {
                    state = 1;
                }
            }
            else
            {
                state = 0;
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```

if (ch == '2') // Push 2 on keypad
{
    lcd_clear(1);
    lcd_clear(2);
    lcd_gotoxy(1,1);
    lcd_printf("Price          12");
    lcd_gotoxy(1,2);
    lcd_printf("Insert your card"); // Buy Pepsi by card
                                // or touch sim

    if(Check_timeout() == 0)
    {
        state = 3;
    }
    else
    {
        state = 0;
    }
}

if (ch == '3') // Push 3 on keypad
{
    lcd_clear(1);
    lcd_clear(2);
    lcd_gotoxy(1,1);
    lcd_printf("Check Money "); // Check Money in your
                                // card or touch sim

    lcd_gotoxy(1,2);
    lcd_printf("Insert your card");
    if(Check_timeout() == 0)
    {
        state = 5;
    }
    else
    {
        state = 0;
    }
}

}

if(state == 1)
{
    CheckCard();
    if (state==2)
    {
        while (!card_disconnect()); // wait until card disconnect

        delaytime (1500);
        fw_step2 (16,50); // increasing speed of Step Motor
        state=0;
    }
}

if(state == 3)
{
    CheckCard1();
    if (state==4)
    {
        while (!card_disconnect());

        delaytime (1500);
        fw_step (16,50);
        state=0;
    }
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
if(state == 5)
{
    checkMon();
    if (state==6)
    {
        while (!card_disconnect());
        delaytime (1500);           // Delay 1500 ms or 1.5 s
        state=0;
    }
}
}
}
//-----//
```



```

D:\My_CCS\Test_Led_Sensor\Test_Led_Sensor.c
// Code Sending Alert SMS to Vendor

#include <18F4550.h>
#fuses HS,NOLVP
#fuses NOPROTECT,NOWDT

#define CLK_SPD 20000000 // Clock Speed
#use delay(clock=CLK_SPD)

#use rs232(baud=4800,xmit=PIN_C6,rcv=PIN_C7) // Set Usart

int1 sms1=0;
int1 sms2=0;

int count1=0;
int count2=0;

char phone_num[13] = "+66832539782"; // Vendor Phone Number
char ctrl_z = 26; // Last Command For Sending SMS

void Init_MCU(void); // Initial MCU Function
void Rece_Value1(void); // Receive Value From Sensor 1 Function
void Rece_Value2(void); // Receive Value From Sensor 2 Function
void Blink_BLed(int x); // Blink Blue led Function
void Blink_RLed(int y); // Blink Red led Function
void Blink_SLed(int z); // Led For Starting Program Function
void Test_Message1(void); // SMS 1 Function
void Test_Message2(void); // SMS 2 Function

#INT_EXT1
void Rece_Value1(void) // Interrupts From Portb bit 0
{
    sms1=1;
}

#INT_EXT
void Rece_Value2(void) // Interrupts From Portb bit 1
{
    sms2=1;
}

void Init_MCU(void)
{
    set_tris_b(0x0F); // Set Portb Direction
                    // b0,b1,b2,b3 : Input
                    // b4,b5,b6,b7 : Output
    set_tris_d(0x00); // Set Portd Direction

    enable_interrupts(GLOBAL); // Enable GLOBAL Interrupts
    enable_interrupts(INT_EXT); // Enable External 0 Interrupts
    enable_interrupts(INT_EXT1); // Enable External 1 Interrupts
    ext_int_edge(0,H_TO_L); // High to Low Edge on Ext 0
    ext_int_edge(1,H_TO_L); // High to Low Edge on Ext 1

    output_low(PIN_D4); // Pin d4 is 0
    output_low(PIN_D5); // Pin d5 is 0
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D:\My_CCS\Test_Led_Sensor\Test_Led_Sensor.c

```
void Blink_BLed(int x)
{
    int i;

    for(i=0;i<15;i++)
    {
        output_high(PIN_D4);           // Pin d4 is 1
        delay_ms(x);                  // Delay x ms
        output_low(PIN_D4);           // Pin d4 is 0
        delay_ms(x);
    }
}

void Blink_RLed(int y)
{
    int j;

    for(j=0;j<15;j++)
    {
        output_high(PIN_D5);           // Pin d5 is 1
        delay_ms(y);                  // Delay y ms
        output_low(PIN_D5);           // Pin d5 is 0
        delay_ms(y);
    }
}

void Blink_SLed(int z)
{
    int k;

    for(k=0;k<25;k++)
    {
        output_high(PIN_D4);           // Delay z ms
        delay_ms(z);
        output_low(PIN_D4);
        delay_ms(z);
    }
}

void Test_Message1(void)
{
    printf("at+cmgf=1\r\n");
    printf("at+cmgs=\"%s\"\r\n",phone_num);
    delay_ms(500);
    printf("The Goods 01 were Sold out at Area 2 Box 1\r\n");
    printf("%c\r\n",ctrl_z);
}

void Test_Message2(void)
{
    printf("at+cmgf=1\r\n");
    printf("at+cmgs=\"%s\"\r\n",phone_num);
    delay_ms(500);
    printf("The Goods 02 were Sold out at Area 2 Box 1\r\n");
    printf("%c\r\n",ctrl_z);
}

void main()
{
    Init_MCU();                       // Initial MCU
    Blink_SLed(50);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน 2 ข้อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D:\My_CCS\Test_Led_Sensor\Test_Led_Sensor.c

```
while(true)
{
    while(sms1 == 1)
    {
        sms1=0;
        if(count1 == 3)
        {
            // printf("Count 1 = %d\r\n",count1); // Use when Test with H
            Blink_RLed(50);
            Test_Message1(); // Sending SMS1 To Vendor
            count1=0;
            disable_interrupts(INT_EXT1);
        }
        else
        {
            Blink_RLed(20);
            // printf("Count 1 = %d\r\n",count1); // Use when Test with Hyp
        }
        count1++;
    }
    while(sms2 == 1)
    {
        sms2=0;
        if(count2 == 3)
        {
            // printf("Count 2 = %d\r\n",count2); // Use when Test with Hyp
            Blink_BLed(50);
            Test_Message2(); // Sending SMS2 To Vendor
            count2=0;
            disable_interrupts(INT_EXT);
        }
        else
        {
            Blink_BLed(20);
            // printf("Count 2 = %d\r\n",count2); // Use when Test with Hyp
        }
        count2++;
    }
}
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน³เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RFID MODULE

Mifare Read/Write Module

SL015M-1

User Manual

Version 2.5

Sep 2007

StrongLink

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

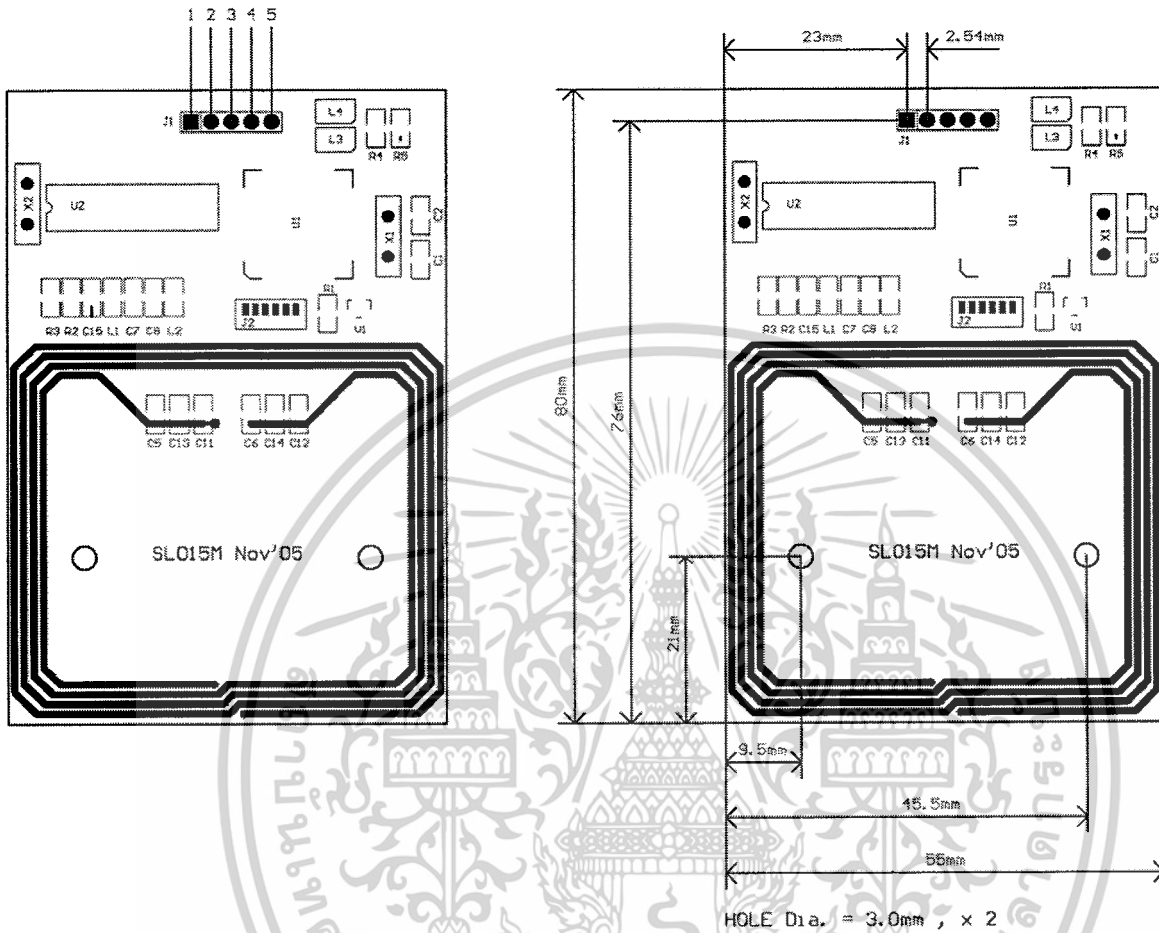
CONTENT

1. MAIN FEATURES	3
2. PINNING INFORMATION	4
3. BAUD RATE SETTING	5
4. COMMUNICATION PROTOCOL	5
4-1. Communication setting	5
4-2. Communication Format	5
4-3. Command Overview	6
4-4. Command List	7
4-4-1. Select Mifare card	7
4-4-2. Login to a sector	7
4-4-3. Read a data block	7
4-4-4. Write a data block	8
4-4-5. Read a value block	8
4-4-6. Initialize a value block	8
4-4-7. Write master key (key A)	9
4-4-8. Increment value	9
4-4-9. Decrement value	9
4-4-10. Copy value	10
4-4-11. Read a data page (UltraLight)	10
4-4-12. Write a data Page (UltraLight)	10
4-4-13. Control Red Led	11
4-4-14. Reset	11

1. MAIN FEATURES

- Tag supported: Mifare 1K, Mifare 4K, Mifare UltraLight
- Auto detecting tag
- Built-in antenna
- UART interface, baud rate 9,600 ~ 115,200 bps
- DC4.5V to DC5.5V VDD operating
- Operating distance: Up to 80mm, depending on tag
- Storage temperature: -40 °C ~ +85 °C
- Operating temperature: -20 °C ~ +70 °C
- Dimension: 80 × 55 × 7 mm
- Two LEDs, green led is auto light when tag in detection range, red led is controlled by host
- The OUT pin is low level indicating tag in detection range, and high level indicating tag out

2. PINNING INFORMATION



PIN	SYMBOL	TYPE	DESCRIPTION
1	TagSta	Output	Tag detect signal low level indicating tag in detection range high level indicating tag out
2	TXD	Output	Serial output port
3	RXD	Input	Serial input port
4	VCC	PWR	Power Supply
5	GND	PWR	Ground

3. BAUD RATE SETTING

R6 & R7 are two 0 ohm resistances assembled on the bottom layer of module, are used for config baud rate as follows sheet

	R6	R7	Baud rate bps
Assembled	no	no	9,600
	yes	no	19,200
	no	yes	57,600
	yes	yes	115,200

4. COMMUNICATION PROTOCOL

4-1. Communication setting

The communication protocol is byte oriented. Both sending and receiving bytes are in hexadecimal format. The communication parameters are as follows,

Baud rate: 9,600 ~ 115,200 bps
 Data: 8 bits
 Stop: 1 bit
 Parity: None
 Flow control: None

4-2. Communication Format

Host to Reader:

Header	Len	Command	Data	Checksum
--------	-----	---------	------	----------

Header: Communication header, 1 byte.
 From host to module: 0xBA.

Len: Byte length counting from Command to Checksum inclusively, 1 byte.

Command: Command, 1 byte.

Data: Data, variable length depends on the command type.

Checksum: XOR result from Header to Data inclusively, 1 byte.

Reader to Host:

Header	Len	Command	Status	Data	Checksum
--------	-----	---------	--------	------	----------

Header: Communication header, 1 byte.
 From module to host: 0xBD.

Len: Byte length counting from Command to Checksum inclusively, 1 byte.

Command: Command, 1 byte.

Status: Command status, 1 byte

Data: Data, variable length depends on the command type.

Checksum: XOR result from Header to Data inclusively, 1 byte.

4-3. Command Overview

Command	Description
0x01	Select Mifare card
0x02	Login to a sector
0x03	Read a data block
0x04	Write a data block
0x05	Read a value block
0x06	Initialize a value block
0x07	Write master key (key A)
0x08	Increment value
0x09	Decrement value
0x0A	Copy value
0x10	Read a data page (UltraLight)
0x11	Write a data page (UltraLight)
0x40	Control the red led
0xFF	Reset

Status Overview

Status	Description
0x00	Operation success
0x01	No tag
0x02	Login success
0x03	Login fail
0x04	Read fail
0x05	Write fail
0x06	Unable to read after write
0x0A	Collision occur
0x0D	Not authenticate
0x0E	Not a value block
0xF0	Checksum error
0xF1	Command code error

4-4. Command List

4-4-1. Select Mifare card

0xBA	Len	0x01	Checksum
------	-----	------	----------

Return:

0xBD	Len	0x01	Status	Serial num	Type	Checksum
------	-----	------	--------	------------	------	----------

Status: 0x00: Operation success

0x01: No tag

0x0A: Collision occur

0xF0: Checksum error

Serial num: Serial number of the card detected if the operation is success, 4 bytes for Mifare Standard & Mifare Pro(X), 7 bytes for Mifare UltraLight & Mifare DesFire

Type: 0x01: Mifare Standard 1K card

0x02: Mifare Pro card

0x03: Mifare UltraLight card

0x04: Mifare Standard 4K card

0x05: Mifare ProX card

0x06: Mifare DesFire card

4-4-2. Login to a sector

0xBA	Len	0x02	Sector	Type	Key	Checksum
------	-----	------	--------	------	-----	----------

Sector: Sector need to login

Type: Key type (0xAA: authenticate with KeyA, 0xBB: authenticate with KeyB)

Key: Authenticate key, 6 bytes

Return:

0xBD	Len	0x02	Status	Checksum
------	-----	------	--------	----------

Status: 0x02: Login success

0x01: No tag

0x03: Login fail

0xF0: Checksum error

4-4-3. Read a data block

0xBA	Len	0x03	Block	Checksum
------	-----	------	-------	----------

Block: The block number to be read, 1 byte

Return:

0xBD	Len	0x03	Status	Data	Checksum
------	-----	------	--------	------	----------

Status: 0x00: Operation success

0x01: No tag

0x04: Read fail

0x0D: Not authenticate

0xF0: Checksum error

Data: Block data returned if operation is success, 16 bytes.

4-4-4. Write a data block

0xBA	Len	0x04	Block	Data	Checksum
------	-----	------	-------	------	----------

Block: The block number to be written, 1 byte.

Data: The data to write, 16 bytes.

Return:

0xBD	Len	0x04	Status	Data	Checksum
------	-----	------	--------	------	----------

Status: 0x00: Operation success

0x01: No tag

0x05: Write fail

0x06: Unable to read after write

0x0D: Not authenticate

0xF0: Checksum error

Data: Block data written if operation is success, 16 bytes.

4-4-5. Read a value block

0xBA	Len	0x05	Block	Checksum
------	-----	------	-------	----------

Block: The block number to be read, 1 byte.

Return:

0xBD	Len	0x05	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation success

0x01: No tag

0x04: Read fail

0x0D: Not authenticate

0x0E: Not a value block

0xF0: Checksum error

Value: Value returned if the operation is success, 4 bytes.

4-4-6. Initialize a value block

0xBA	Len	0x06	Block	Value	Checksum
------	-----	------	-------	-------	----------

Block: The block number to be initialized, 1 byte.

Value: The value to be written, 4 bytes.

Return:

0xBD	Len	0x06	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation success

0x01: No tag

0x05: Write fail

0x06: Unable to read after write

0x0D: Not authenticate

0xF0: Checksum error

Value: Value written if the operation is success, 4 bytes.

4-4-7. Write master key (key A)

0xBA	Len	0x07	Sector	Key	Checksum
------	-----	------	--------	-----	----------

Sector: The sector number to be written, 1 byte.

Key: Authentication key, 6 bytes

Return:

0xBD	Len	0x07	Status	Key	Checksum
------	-----	------	--------	-----	----------

Status: 0x00: Operation success

0x01: No tag

0x05: Write fail

0x0D: Not authenticate

0xF0: Checksum error

Key: Authentication key written if the operation is success, 6 bytes.

4-4-8. Increment value

0xBA	Len	0x08	Block	Value	Checksum
------	-----	------	-------	-------	----------

Block: The block number to be increased, 1 byte.

Value: The value to be increased by, 4 bytes.

Return:

0xBD	Len	0x08	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation success

0x01: No tag

0x05: Write fail

0x06: Unable to read after write

0x0D: Not authenticate

0x0E: Not a value block

0xF0: Checksum error

Value: The value after increment if the operation is success, 4 bytes

4-4-9. Decrement value

0xBA	Len	0x09	Block	Value	Checksum
------	-----	------	-------	-------	----------

Block: The block number to be decreased, 1 byte

Value: The value to be decreased by, 4 bytes

Return:

0xBD	Len	0x09	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation success

0x01: No tag

0x05: Write fail

0x06: Unable to read after write

0x0D: Not authenticate

0x0E: Not a value block

0xF0: Checksum error

Value: The value after decrement if the operation is success, 4 bytes

4-4-10. Copy value

0xBA	Len	0x0A	Source	Destination	Checksum
------	-----	------	--------	-------------	----------

Source: The source block copy from, 1 byte

Destination: The destination copy to, 1 byte

The source and destination must in the same sector

Return:

0xBD	Len	0x0A	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation success

0x01: No tag

0x05: Write fail

0x06: Unable to read after write

0x0D: Not authenticate

0x0E: Not a value block (Source)

0xF0: Checksum error

Value: The value after copy if the operation is success, 4 bytes

4-4-11. Read a data page (UltraLight)

0xBA	Len	0x10	Page	Checksum
------	-----	------	------	----------

Page: The page number to be read, 1 byte

Return:

0xBD	Len	0x10	Status	Data	Checksum
------	-----	------	--------	------	----------

Status: 0x00: Operation success

0x01: No tag

0x04: Read fail

0xF0: Checksum error

Data: Block data returned if operation is success, 4 bytes.

4-4-12. Write a data Page (UltraLight)

0xBA	Len	0x11	Page	Data	Checksum
------	-----	------	------	------	----------

Page: The page number to be written, 1 byte.

Data: The data to write, 4 bytes.

Return:

0xBD	Len	0x11	Status	Data	Checksum
------	-----	------	--------	------	----------

Status: 0x00: Operation success

0x01: No tag

0x05: Write fail

0x06: Unable to read after write

0xF0: Checksum error

Data: page data written if operation is success, 4 bytes.

4-4-13. Control Red Led

0xBA	Len	0x40	Code	Checksum
------	-----	------	------	----------

Code: 0 command red led turn off , other red led turn on, 1 byte

Return:

0xBD	Len	0x40	Status	Checksum
------	-----	------	--------	----------

Status: 0x00: Operation success

0xF0: Checksum error

4-4-14. Reset

0xBA	Len	0xFF	Checksum
------	-----	------	----------

No return

