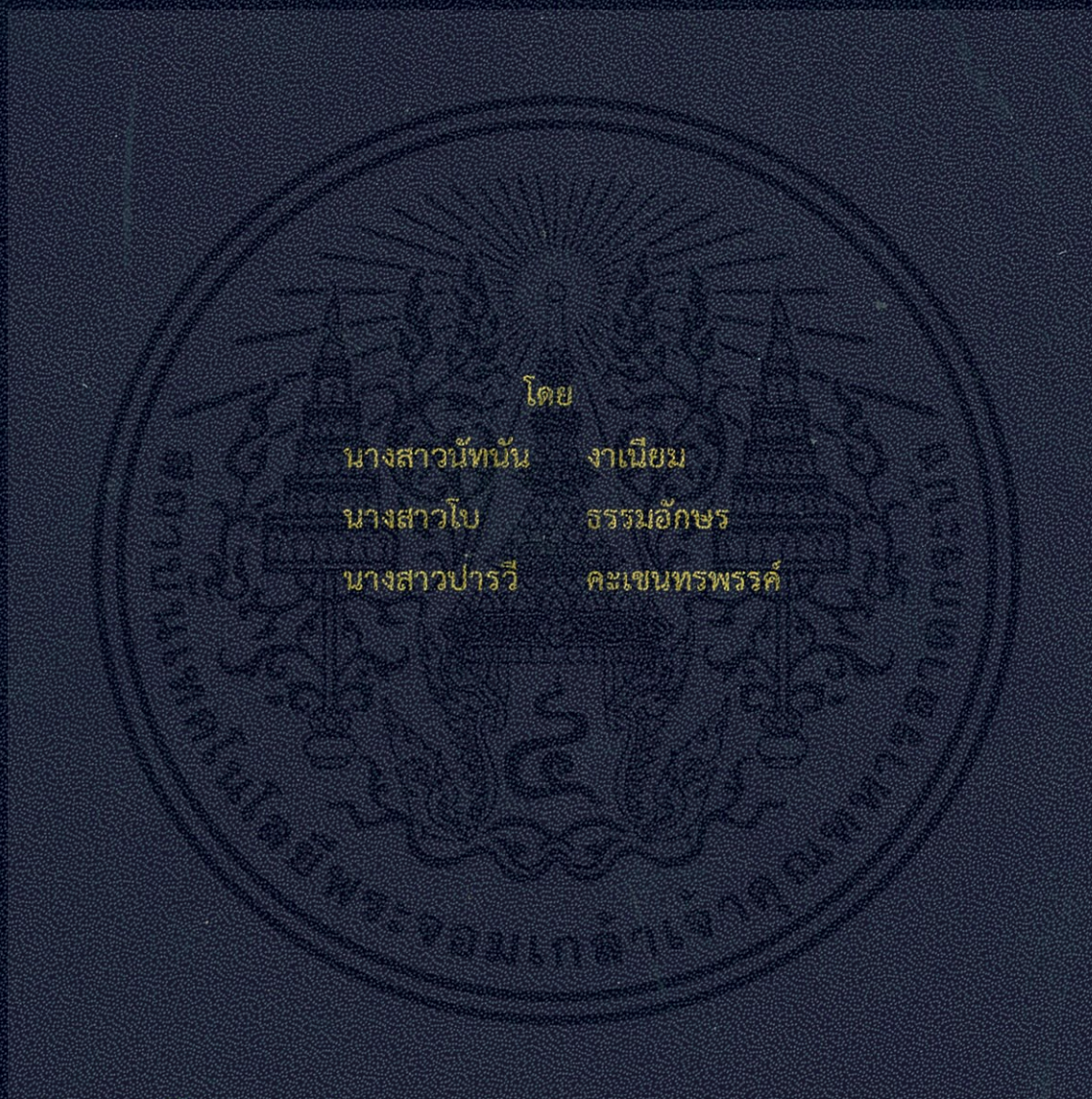


ระบบชำระเงินอัตโนมัติผ่านบัตรสมาร์ทการ์ด
AUTOMATIC PAYMENT BY SMARTCARD



โดย

นางสาวนันทน์	จาเนียม
นางสาวโบ	ธรรมอักษร
นางสาวปารวี	คะเชนทรพรรค

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2556

ระบบชำระเงินอัตโนมัติผ่านบัตรสมาร์ทการ์ด
AUTOMATIC PAYMENT BY SMARTCARD

โดย
นางสาวนันทน์ งามเนียม
นางสาวโบ ธรรมอักษร
นางสาวปารวี คณะเชนทรพรรค์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2556

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบชำระเงินอัตโนมัติผ่านบัตรสมาร์ทการ์ด
AUTOMATIC PAYMENT BY SMARTCARD



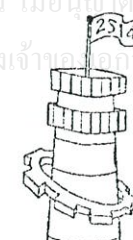
ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2556

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ผ่านการตรวจรูปเล่มแล้ว
(*Dr. S.*)
อาจารย์ที่ปรึกษา
13/3/57



ผ่านการตรวจชิ้นงานแล้ว
(*[Signature]*)
กรรมการผู้ตรวจชิ้นงาน
24/3/57

ปริญญาบัตรปีการศึกษา 2556

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบชำระเงินอัตโนมัติผ่านบัตรสมาร์ทการ์ด

AUTOMATIC PAYMENT BY SMARTCARD

ผู้จัดทำ

- | | | |
|-----------------|-------------|----------|
| 1. นางสาวนันทน์ | งานิยม | 53010835 |
| 2. นางสาวโบ | ธรรมอักษร | 53010890 |
| 3. นางสาวปารวี | คะเชนทรพรรค | 53010983 |


..... อาจารย์ที่ปรึกษา
(ดร. สิริภพ ตู่ประกาย)

..... อาจารย์ที่ปรึกษา
(รศ.ดร. กอบชัย เดชหาญ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้ได้รับความกรุณาจาก ดร. สิริภพ ตู้ประกาย อาจารย์ที่ปรึกษา และ รศ.ดร.กอบชัย เดชหาญ อาจารย์ที่ปรึกษาร่วม ที่ได้ให้คำเสนอ แนวคิด ตลอดจนแก้ไขข้อบกพร่องต่างๆ มาโดยตลอด ผู้ศึกษาขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณคุณพ่อ คุณแม่ และผู้ปกครอง ที่อยู่เบื้องหลังในความสำเร็จที่ได้ให้ความช่วยเหลือ สนับสนุน และให้กำลังใจที่ดีเสมอมา



นางสาวนันทน์ งามเนียม
นางสาวโบ ธรรมอักษร
นางสาวปารวี คะเชนทรพรรค
ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบชำระเงินอัตโนมัติผ่านบัตรสมาร์ทการ์ด
AUTOMATIC PAYMENT BY SMARTCARD

โดย นางสาวนันทน์ งามเนียม 53010835
นางสาวโบ ธรรมอักษร 53010890
นางสาวปารวี คะเชนทรพรรค 53010983

อาจารย์ที่ปรึกษา ดร.สิรภพ ตู้ประกาย
รศ.ดร.กอบชัย เดชหาญ

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการนำเอาเทคโนโลยี RFID มาประยุกต์ใช้ในการชำระเงิน เพื่อให้การชำระเงินมีความสะดวกมากขึ้นคล้ายเป็นบัตรสมาร์ทการ์ด ที่สามารถพกพาสะดวก ใช้งานได้ง่าย และออกแบบระบบการสั่งอาหารที่สามารถสั่งอาหารได้โดยการกดปุ่มที่เชื่อมต่ออยู่กับคอมพิวเตอร์แล้วโปรแกรมที่สร้างขึ้นจะเป็นตัวส่งการไปที่ไมโครคอนโทรลเลอร์ และใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมการทำงานของหน้าจอแสดงผลและตัวอ่าน RFID ที่อยู่ทางด้านผู้ซื้อ โดยข้อมูลผู้ถือบัตรและยอดเงินคงเหลือจะถูกบันทึกลงในฐานข้อมูล

ABSTRACT

This project presents an automatic payment system through RFID card to make payment easier like a smartcard. The program design that can order food by pressing a keyboard then the program sends command to microcontroller and uses microcontroller to control LCD and RFID reader on a customer side. Personal information and balance saves in database.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	III
สารบัญรูป	VI
สารบัญตาราง	VIII
บทที่ 1	1
บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของปริญญานิพนธ์	1
บทที่ 2	3
ทฤษฎีและหลักการที่เกี่ยวข้อง	3
2.1 ความหมายของอาร์เอฟไอดี	3
2.2 ส่วนประกอบของระบบอาร์เอฟไอดี	3
2.2.1 แท็ก	4
2.2.1.1 พาสซีฟอาร์เอฟไอดีแท็ก	5
2.2.1.2 แอคทีฟอาร์เอฟไอดีแท็ก	6
2.2.2 เครื่องอ่าน	6
2.3 หลักการและเทคนิคที่ใช้	8
2.3.1 การเข้ารหัสแบบแมนเชสเตอร์	8
2.3.2 ขั้นตอนการทำงานระหว่างเครื่องอ่านกับแท็ก	9
2.4 คลื่นพาหะในระบบอาร์เอฟไอดี	10
2.5 ตัวอย่างการประยุกต์ใช้งานระบบอาร์เอฟไอดี	11
2.6 การใช้งานโปรแกรมอาดูอิโน้ (Arduino)	11
2.6.1 ขั้นตอนการใช้งานโปรแกรมอาดูอิโน้	12
2.7 การจัดการฐานข้อมูลมายเอสคิวแอลโดยใช้โปรแกรมแอพเซิร์ฟ (AppServ)	16
2.7.1 ขั้นตอนการติดตั้งโปรแกรมแอพเซิร์ฟ	16
2.7.2 การจัดการฐานข้อมูลมายเอสคิวแอล	22

2.8	การใช้งานโปรแกรมวิซวลเบสิก	26
2.8.1	ขั้นตอนการใช้งานโปรแกรมวิซวลเบสิก	26
2.9	การใช้งานโปรแกรมคริสตัลรีพอร์ต	30
2.9.1	ขั้นตอนการใช้งานโปรแกรมคริสตัลรีพอร์ต	30
บทที่ 3	การออกแบบและการจัดทำปฏิญานีพนธ์	36
3.1	การออกแบบ	36
3.1.1	ออกแบบการทำงานโดยรวม	36
3.1.1.1	ออกแบบระบบการสั่งอาหารในอาดูอิน	37
3.1.1.2	ออกแบบระบบเติมเงิน	38
3.1.1.3	ออกแบบการต่อไมโครคอนโทรลเลอร์กับหน้าจอแสดงผล	39
3.1.1.4	ออกแบบการต่อไมโครคอนโทรลเลอร์กับตัวอ่านอาร์เอฟไอดี	40
3.1.1.5	ออกแบบฐานข้อมูลที่ใช้งาน	41
3.2	เครื่องมือที่ใช้ในการทดลอง	41
3.3	การจัดเก็บผลการทดลอง	42
บทที่ 4	ผลการทดลอง	43
4.1	สัญญาณจากวงจรอ่านอาร์เอฟไอดีเมื่อมีการอ่านบัตร	43
4.2	วัดความถี่ของเครื่องอ่านอาร์เอฟไอดี	50
4.3	ผลการแสดงผลบนหน้าจอแสดงผลเทียบกับโปรแกรมสั่งอาหารในคอมพิวเตอร์	51
4.3.1	เมื่อทำการล็อกอินเข้าโปรแกรมในคอมพิวเตอร์	51
4.3.2	เมื่อทำการใส่รหัสล็อกอินเสร็จ	51
4.3.3	หน้าต่างให้เลือกสั่งอาหารหรือเติมเงิน	52
4.3.3.1	เลือกสั่งอาหาร	52
4.3.3.2	เลือกเติมเงิน	57
4.3.3.3	เลือกสมัครสมาชิกใหม่	58
4.3.3.4	เลือกดูข้อมูลลูกค้า	59
4.3.3.5	เลือกดูรายงานการขาย	59
4.4	อุปกรณ์ ฝั่งลูกค้า	60

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5	สรุปผลและข้อเสนอแนะ	61
	5.1 สรุปผล	61
	5.1.1 ออกแบบการทำให้ไมโครคอนโทรลเลอร์เชื่อมต่อกับอุปกรณ์ต่างๆ	61
	5.1.2 การเขียนโปรแกรมสั่งอาหาร	61
	5.1.3 การชำระเงินผ่านบัตรอาร์เอฟไอดี	61
	5.2 ข้อเสนอแนะ	61
บรรณานุกรม		62
ภาคผนวก ก	โค้ดเขียนไมโครคอนโทรลเลอร์	
ภาคผนวก ข	DATASHEET ATMEGA2560	



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า	
1.1	บล็อกไดอะแกรมของระบบ	2
2.1	ภาพรวมของระบบอาร์เอฟไอดี	4
2.2	อาร์เอฟไอดีแท็กในรูปแบบต่างๆ	5
2.3	บล็อกไดอะแกรมของพาสซีฟแท็ก	6
2.4	ตัวอย่าง แอคทิฟแท็กที่มีแบตเตอรี่ลิเทียม 2 ก้อนอยู่ภายนอก โครงสร้าง	6
2.5	โครงสร้างภายในเครื่องอ่าน ภายในเครื่องอ่าน	7
2.6	ตัวอย่างเครื่องอ่านแบบต่างๆ	7
2.7	สัญญาณรูปคลื่นที่เข้ารหัสแบบแมนเชสเตอร์	9
2.8	ความถี่ย่านที่ระบบอาร์เอฟไอดีถูกใช้งาน	10
2.9	ระบบควบคุมการผ่านเข้าออก	11
2.10	ต่อบอร์ดอาคูอิโนเข้ากับคอมพิวเตอร์	12
2.11	หน้าต่างโปรแกรมอาคูอิโน	12
2.12	การกำหนดบอร์ดและพอร์ตการใช้งาน	13
2.13	พื้นที่ที่ใช้สำหรับเขียนข้อมูล	13
2.14	การตรวจสอบความผิดพลาดของข้อมูล	14
2.15	การอัปโหลดข้อมูล	15
2.16	หน้าต่างเมื่ออัปโหลดข้อมูลสำเร็จ	16
2.17	ขั้นตอนการติดตั้งโปรแกรมแอปเซิร์ฟ	17
2.18	รายละเอียดเงื่อนไขการติดตั้ง	17
2.19	เลือกปลายทางการติดตั้งโปรแกรมแอปเซิร์ฟ	18
2.20	เลือกแพ็คเกจคอมโพเนนท์ที่ต้องการติดตั้ง	19
2.21	การกำหนดค่าคอนฟิกค่าอะแพชชีเว็บเซิร์ฟเวอร์	20
2.22	การกำหนดค่าคอนฟิกของมายเอสคิวแอลดาต้าเบส	21
2.23	หน้าขั้นตอนสิ้นสุดการติดตั้งโปรแกรมแอปเซิร์ฟ	21
2.24	หน้าต่างโปรแกรมแอปเซิร์ฟ	22
2.25	หน้าต่างโปรแกรม PHPMYADMIN	22
2.26	การสร้างฐานข้อมูลใหม่ในโปรแกรม PHPMYADMIN	23
2.27	การสร้างตารางใน PHPADMIN	23
2.28	การตั้งค่าตารางใน PHPADMIN	24

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์อื่นใด

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.29	หน้าต่าง PHPADMIN เมื่อสร้างตารางเสร็จแล้ว	24
2.30	ช่องสำหรับบันทึกข้อมูลใน PHPADMIN	25
2.31	หน้าต่าง PHPADMIN เมื่อข้อมูลถูกบันทึกเรียบร้อยแล้ว	25
2.32	โปรแกรมวิซวลเบสิก	26
2.33	การสร้างงานใหม่ในโปรแกรมวิซวลเบสิก	27
2.34	การเลือกภาษาและประเภทในโปรแกรมวิซวลเบสิก	27
2.35	หน้าต่างการทำงานในโปรแกรมวิซวลเบสิก	28
2.36	หน้าต่างสำหรับการเขียนโค้ดในโปรแกรมวิซวลเบสิก	28
2.37	ปุ่มเริ่มต้น (START) ในโปรแกรมวิซวลเบสิก	29
2.38	หน้าต่างหลังจากสั่งเริ่มต้น (START) ในโปรแกรมวิซวลเบสิก	29
2.39	หน้าต่างแสดงการสร้างฐานข้อมูลที่ต้องการทำรายงาน	30
2.40	หน้าต่างแสดงการสร้างตาต้าเซ็ท	31
2.41	หน้าต่างแสดงการกำหนดชนิดตาต้าเซ็ท	31
2.42	หน้าต่างแสดงการตั้งค่าในตาต้าเซ็ท	32
2.43	หน้าต่างแสดงการเลือกฐานข้อมูล	32
2.44	หน้าต่างแสดงการกำหนดชื่อให้รายงาน	33
2.45	หน้าต่างแสดงการเลือกรูปแบบการแสดงผล	34
2.46	หน้าต่างแสดงการกำหนดเลือกฐานข้อมูลที่ต้องการแสดง	34
2.47	หน้าต่างสำหรับสร้างรายงาน	35
2.48	หน้าต่างแสดงผลของรายงาน	35
3.1	ระบบการทำงานโดยรวม	36
3.2	โพล์ชาร์ตของระบบการสั่งอาหาร	37
3.3	โพล์ชาร์ตของระบบเติมเงิน	38
3.4	การต่อไมโครคอนโทรลเลอร์กับหน้าจอแสดงผล	39
3.5	การต่อไมโครคอนโทรลเลอร์กับวงจรรอ่านอาร์เอฟไอดี	40
3.6	การออกแบบฐานข้อมูล	41
4.1	สัญญาณพัลส์ 35 32 35 38 38 42	43
4.2	สัญญาณพัลส์ 34 35 33 34	44
4.3	สัญญาณพัลส์ 33 32 33 39	44
4.4	สัญญาณพัลส์ 33 34 33 39 33	45
4.5	สัญญาณพัลส์ 36 33 37 33 32 33	45
4.6	สัญญาณพัลส์ 39 33 35 32 30 32	46
4.7	สัญญาณพัลส์ 30 32 30 32 30 32	46
4.8	สัญญาณพัลส์ 30 32 30 32 30	47

4.9	สัญญาณพัลส์ 32 30 32 30 32 30 32	47
4.10	สัญญาณพัลส์ 30 32 30 32 30 32	48
4.11	สัญญาณพัลส์ 30 CR	48
4.12	สัญญาณทั้งหมดที่อ่านได้เมื่อมีแท็ก	49
4.13	ค่าแท็กที่อ่านได้จากมอนิเตอร์	50
4.14	ความถี่ของเครื่องอ่านอาร์เอฟไอดี	50
4.15	หน้าต่างโปรแกรมการลือกอิน	51
4.16	หน้าต่างเมื่อการลือกอินสำเร็จ	51
4.17	การเลือกรายการ	52
4.18	การเลือกรายการเพื่อสั่งอาหาร	52
4.19	หน้าจอแสดงผลเมื่อทาบบัตร	53
4.20	หน้าจอแสดงผลเมื่อทาบบัตรเสร็จ	53
4.21	หน้าจอแสดงผลเพื่อให้สั่งอาหาร	53
4.22	หน้าจอแสดงผลเพื่อให้ใส่จำนวน	54
4.23	หน้าจอแสดงผลเมื่อสินค้าคงเหลือไม่พอ	54
4.24	หน้าจอแสดงผลเมื่อสินค้าหมด	55
4.25	หน้าจอแสดงผลเมื่อสิ้นสุดการสั่งอาหาร	55
4.26	หน้าจอแสดงผลเมื่อเงินคงเหลือไม่พอ	56
4.27	หน้าจอแสดงผลเมื่อเลือกสั่งอาหารต่อ	56
4.28	หน้าจอแสดงผลเมื่อเลือกดูรายการอาหารที่สั่ง	57
4.29	หน้าจอแสดงผลเมื่อเลือกเติมเงิน	58
4.30	หน้าจอแสดงผลเมื่อเลือกสมัครสมาชิกใหม่	58
4.31	หน้าจอเมื่อเลือกดูข้อมูลลูกค้า	59
4.32	แผนภูมิแท่งสรุปยอดขาย	59
4.33	ตารางสรุปยอดขาย	60
4.34	เครื่องสั่งอาหารและชำระเงินฝั่งทางลูกค้า	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันนี้ผู้คนมีความเร่งรีบในชีวิตมากขึ้น จึงต้องการการบริการเพิ่มมากขึ้นเพื่อความ สะดวกสบายและความรวดเร็วในการทำสิ่งต่างๆ สมาชิกในกลุ่มจึงได้มีความคิดเห็นตรงกัน ที่จะทำ ระบบเกี่ยวกับการซื้อของอัตโนมัติเพียงแค่กดปุ่มเพื่อสั่งอาหาร แล้วจ่ายเงินผ่านบัตรอาร์เอฟไอดี ซึ่ง เป็นบัตรที่สามารถใช้งานง่ายคล้ายเป็นบัตรสมาร์ทการ์ด และรรับอาหาร โดยไม่ต้องใช้เวลาในการต่อ คิวรอนาน ซึ่งระบบนี้จะเพิ่มความรวดเร็วและความสะดวกสบายในการซื้อของมากขึ้นโดยใช้เวลาเพียง ไม่นาน จะใช้ไมโครคอนโทรลเลอร์ ในการควบคุมการทำงานของหน้าจอแสดงผลและเครื่องอ่านบัตร อาร์เอฟไอดี

1.2 วัตถุประสงค์

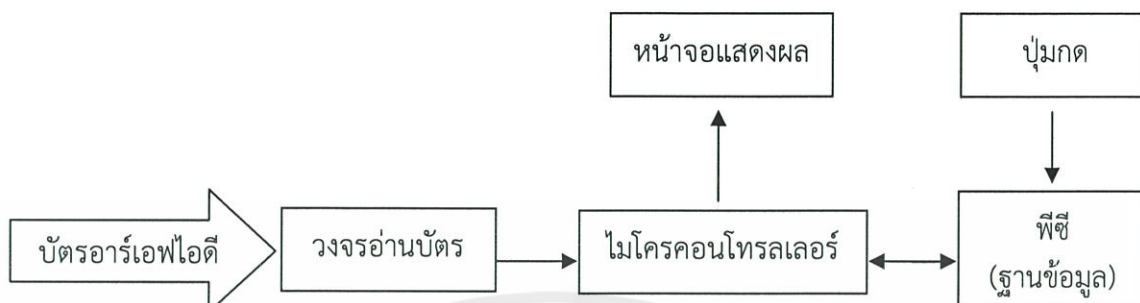
- 1) เพื่อสร้างระบบการสั่งอาหาร
- 2) เพื่อศึกษาการทำงานของ RFID และนำไปประยุกต์ใช้งาน
- 3) เพื่อศึกษาการเชื่อมต่อไมโครคอนโทรลเลอร์กับอุปกรณ์อื่นๆ
- 4) เพื่อศึกษาการเขียนโปรแกรมไมโครคอนโทรลเลอร์

1.3 ขอบเขตของปริญญานิพนธ์

- 1) สามารถนำบัตร RFID มาใช้ในการชำระเงิน
- 2) สามารถคำนวณจำนวนเงินคงเหลือในบัตรจากการใช้จ่าย
- 3) สามารถแสดงผลการใช้งานบนจอแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.1 บล็อกไดอะแกรมรวมของระบบ



รูปที่ 1.1 บล็อกไดอะแกรมของระบบ

การทำงานโดยรวมของระบบชำระเงินอัตโนมัติผ่านบัตรสมาชิกการ์ดหรือบัตรอาร์เอฟไอดีจะตามแสดงแผนผังของระบบในรูปที่ 1.1 ซึ่งจะประกอบด้วยไมโครคอนโทรลเลอร์เป็นตัวหลักที่สำคัญ ซึ่งต่ออยู่กับอุปกรณ์ส่วนอื่นๆ เช่น หน้าจอแสดงผล วงจรรอ่านบัตร และเครื่องคอมพิวเตอร์ที่มีฐานข้อมูล โดยจะรับข้อมูลจากปุ่มกดแล้วแสดงผลทางหน้าจอแสดงผล ผ่านตัวไมโครคอนโทรลเลอร์ และใช้วงจรรอ่านบัตรในการชำระเงินผ่านบัตรอาร์เอฟไอดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 ความหมายของอาร์เอฟไอดี

อาร์เอฟไอดีเป็นเทคโนโลยีใหม่ที่กำลังมีบทบาทและความสำคัญเพิ่มขึ้นอย่างรวดเร็ว การประยุกต์เทคโนโลยีอาร์เอฟไอดีมีรูปแบบหลากหลายด้วยจุดประสงค์ที่แตกต่างกัน แต่อยู่บนหลักการพื้นฐานเดียวกัน นั่นคือการใช้คลื่นความถี่วิทยุเพื่อการระบุอัตลักษณ์ของวัตถุหรือเจ้าของวัตถุที่ติดป้ายอาร์เอฟไอดีแทนการระบุด้วยวิธีการอื่น ซึ่งวิธีการนี้จะช่วยอำนวยความสะดวกและเพิ่มประสิทธิภาพได้ดีกว่า

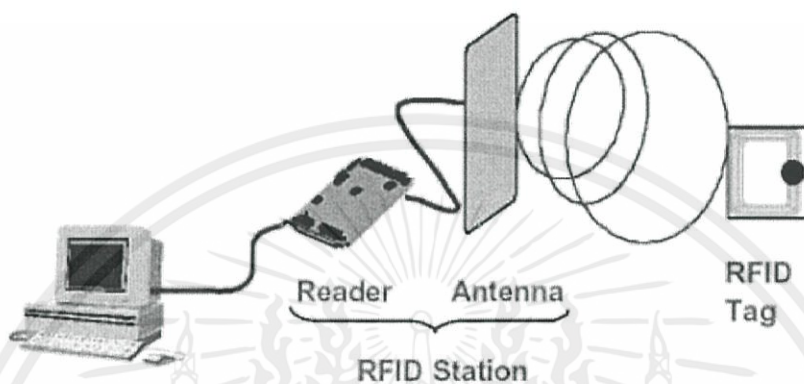
อาร์เอฟไอดีเป็นระบบระบุลักษณะของวัตถุด้วยคลื่นความถี่วิทยุที่ได้ถูกพัฒนามาตั้งแต่ปี ค.ศ. 1980 มีวัตถุประสงค์หลักเพื่อนำไปใช้งานแทนระบบบาร์โค้ด (Barcode) โดยจุดเด่นของอาร์เอฟไอดีอยู่ที่การอ่านข้อมูลจากแท็ก (Tag) ได้หลายๆ แท็กแบบไร้สัมผัสและสามารถอ่านค่าได้แม้ในสภาพที่ทัศนวิสัยไม่ดี ทนต่อความเปียกชื้นแรงสั่นสะเทือน การกระทบกระแทก สามารถอ่านข้อมูลได้ด้วยความเร็วสูง โดยข้อมูลจะถูกเก็บไว้ในไมโครชิปที่อยู่ในแท็กในปัจจุบันได้มีการนำอาร์เอฟไอดีไปประยุกต์ใช้งานในด้านอื่นๆ นอกเหนือจากนำมาใช้แทนระบบบาร์โค้ดแบบเดิมเช่นใช้ในบัตรชนิดต่างๆ เช่น บัตรสำหรับใช้ผ่านเข้าออกสถานที่ต่างๆ บัตรที่จอดรถตามศูนย์การค้าต่างๆ ที่เราอาจพบเห็นอยู่ในรูปของแท็กสินค้า มีขนาดเล็กจนสามารถแทรกลงระหว่างชั้นของเนื้อกระดาษได้ หรือเป็นแคปซูลขนาดเล็กฝังเอาไว้ในตัวสัตว์เพื่อบันทึกประวัติต่างๆ เป็นต้น

2.2 ส่วนประกอบของระบบอาร์เอฟไอดี

ในระบบอาร์เอฟไอดีจะมององค์ประกอบหลักๆอยู่ 2 ส่วนด้วยกัน ส่วนแรก คือ ทรานสปอนเดอร์หรือแท็ก (Transponder/Tag) ที่ใช้ติดกับวัตถุต่างๆ ที่เราต้องการ โดยแท็กที่ว่า จะบันทึกข้อมูลเกี่ยวกับวัตถุชิ้นนั้นๆเอาไว้ ส่วนที่สองก็คือเครื่องสำหรับอ่านเขียนข้อมูลภายในแท็ก (Interrogator/Reader) ด้วยคลื่นความถี่วิทยุ ข้อแตกต่างของระบบ อาร์เอฟไอดีกับระบบบาร์โค้ดหรือระบบรหัสแท่ง คือ ระบบอาร์เอฟไอดีจะใช้คลื่นความถี่ในการอ่านเขียน ส่วนระบบรหัสแท่งจะใช้แสงเลเซอร์ในการอ่าน โดยข้อเสียของระบบบาร์โค้ดคือหลักการอ่านเป็นการใช้แสงในการอ่านแท็กบาร์โค้ด ซึ่งจะต้องอ่านแท็กที่มิฉะนั้นจะปิดหรือต้องอยู่ในเส้นตรงเดียวกับลำแสงที่ยิงจากเครื่องสแกน และอ่านได้ทีละแท็กในระยะใกล้ๆ แต่ระบบอาร์เอฟไอดีจะแตกต่างโดยสามารถ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่เชิงพาณิชย์ การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อ่านแท็กได้ โดยไม่ต้องเห็นแท็ก หรือแท็กนั้นซ่อนอยู่ในวัตถุและไม่จำเป็นต้องอยู่ในเส้นตรงกับคลื่น เพียงอยู่ในบริเวณที่สามารถรับคลื่นวิทยุได้ก็สามารถอ่านข้อมูลได้ ภาพรวมของระบบอาร์เอฟไอดีแสดงดังรูปที่ 2.1

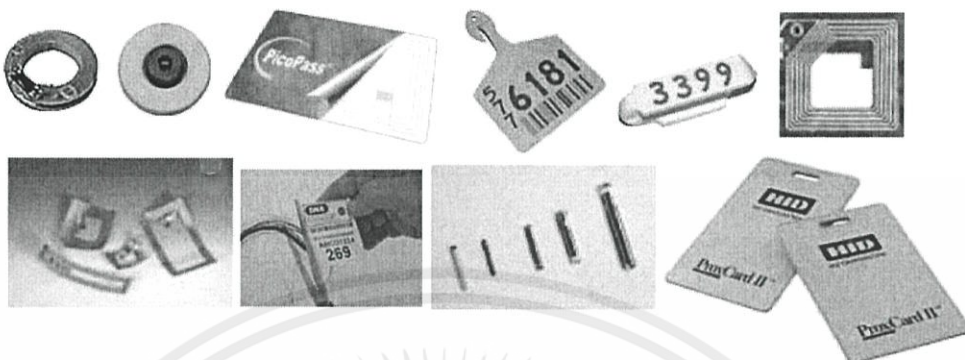


รูปที่ 2.1 ภาพรวมของระบบอาร์เอฟไอดี [1]

2.2.1 แท็ก

โครงสร้างภายในของแท็กจะประกอบด้วย 2 ส่วนใหญ่ๆ ได้แก่ ขดลวดขนาดเล็กซึ่งทำหน้าที่เป็นสายอากาศ สำหรับรับส่งสัญญาณคลื่นความถี่วิทยุ และสร้างพลังงาน ป้อนให้ส่วนของไมโครชิปที่ทำหน้าที่เก็บข้อมูลของวัตถุเช่นรหัสสินค้า โดยทั่วไปตัวแท็กอาจอยู่ในชนิดทั้งเป็นกระดาษ แผ่นฟิล์ม พลาสติก มีขนาดและรูปร่างต่างๆ กันไป ขึ้นอยู่กับวัสดุที่จะนำมาไปติด และมีหลายรูปแบบ เช่น ขนาดเท่าบัตรเครดิต เหรียญ กระดุม ฉลากสินค้า แคปซูล เป็นต้น ดังรูปที่ 2.2 แต่โดยหลักการอาจแบ่งแท็กที่มีการใช้งานกันอยู่ 2 ชนิดใหญ่ๆ แต่ละชนิดก็จะมี ความแตกต่างกันในแง่ของการใช้งานราคา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



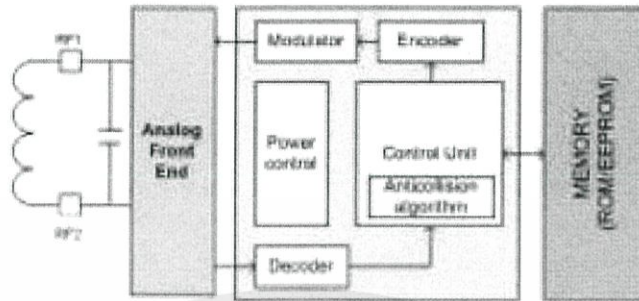
รูปที่ 2.2 อาร์เอฟไอดีแท็กในรูปแบบต่างๆ [1]

2.2.1.1 พาสซีฟอาร์เอฟไอดีแท็ก

แท็กชนิดนี้ไม่ต้องอาศัยแหล่งจ่ายไฟภายนอกใดๆ เพราะภายในแท็กจะมีวงจรถูกฝังไว้เป็นแหล่งจ่ายไฟในตัวอยู่ทำให้การอ่านข้อมูลทำได้ไม่ไกลมากนักระยะอ่านสูงสุดประมาณ 1 เมตร ขึ้นอยู่กับความแรงของเครื่องส่งและคลื่นความถี่วิทยุที่ใช้ ปกติแท็กชนิดนี้มักมีหน่วยความจำขนาดเล็กโดยทั่วไปประมาณ 16 ถึง 1,024 ไบต์ มีขนาดเล็กและน้ำหนักเบา

ไอซีของแท็กชนิดพาสซีฟที่มีการผลิตออกมา จะมีทั้งขนาดและรูปร่างเป็นแท่งหรือแผ่นขนาดเล็กจนแทบไม่สามารถมองเห็นได้ไปจนถึงขนาดใหญ่สะดุดตา ซึ่งต่างก็มีความเหมาะสมกับชนิดงานที่แตกต่างกัน ส่วนโครงสร้างภายในที่เป็นไอซีของแท็กนั้นก็ประกอบด้วย 3 ส่วนหลักๆ ได้แก่ ส่วนควบคุมการทำงานของภาครับ-ส่งสัญญาณวิทยุ ส่วนควบคุมภาคลอจิก ส่วนของหน่วยความจำ ซึ่งอาจจะเป็นแบบรอมหรืออีอีพรอมดังรูปที่ 2.3

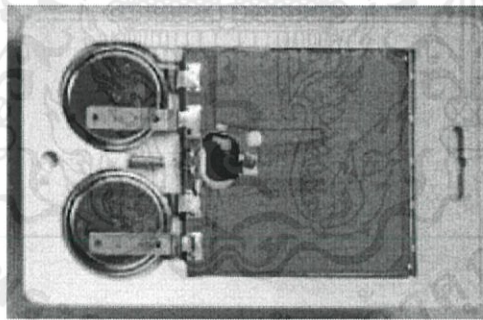
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 บล็อกไดอะแกรมของพาสซีฟแท็ก [1]

2.2.1.2 แอคทีฟอาร์เอฟไอดีแท็ก

แท็กชนิดนี้จะต้องอาศัยแหล่งจ่ายไฟจากแบตเตอรี่ภายนอก เพื่อจ่ายพลังงานให้กับวงจรภายในทำงาน แท็กชนิดนี้มีหน่วยความจำภายในขนาดใหญ่ได้ถึง 1 เมกะไบต์ และสามารถอ่านได้ในระยะไกลสูงสุดประมาณ 10 เมตร แม้ว่าแท็กจะมีข้อดีอยู่หลายข้อแต่ก็มีข้อเสียด้วยเช่นกัน เช่น มีราคาต่อหน่วยแพง มีขนาดค่อนข้างใหญ่ และมีระยะเวลาในการทำงานที่จำกัด ดังรูปที่ 2.4

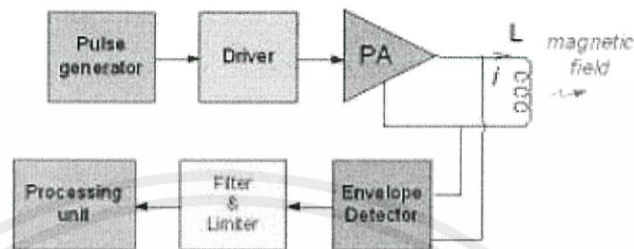


รูปที่ 2.4 ตัวอย่าง แอคทีฟแท็กที่มีแบตเตอรี่ลิเทียม 2 ก้อนอยู่ภายนอก [1]

2.2.2 เครื่องอ่าน

โดยหน้าที่ของเครื่องอ่านก็คือ การเชื่อมต่อเพื่อเขียนหรืออ่านข้อมูลลงในแท็ก ด้วยสัญญาณความถี่วิทยุภายในเครื่องอ่านจะประกอบด้วยเสาอากาศทำจากขดลวดทองแดงเพื่อใช้เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รับส่งสัญญาณภาครับและภาคส่งสัญญาณวิทยุและวงจรควบคุมการอ่าน-เขียนข้อมูล จำพวก ไมโครคอนโทรลเลอร์ และส่วนของการติดต่อกับคอมพิวเตอร์ดังรูปที่ 2.5



รูปที่ 2.5 โครงสร้างภายในเครื่องอ่าน [1]

หน่วยประมวลผลข้อมูลที่อยู่ภายในเครื่องอ่านมักใช้เป็นไมโครคอนโทรลเลอร์ ซึ่ง อัลกอริทึมที่อยู่ภายในโปรแกรมจะทำหน้าที่ถอดรหัสข้อมูลที่ได้รับและทำหน้าที่ติดต่อกับ คอมพิวเตอร์ ลักษณะขนาดและรูปร่างของเครื่องอ่านจะแตกต่างกันไปตามประเภทของการทำงาน เช่น แบบมือถือขนาดเล็กหรือติดผนัง จนถึงขนาดใหญ่เท่าประตู เป็นต้น ดังแสดงในรูปที่ 2.6



รูปที่ 2.6 ตัวอย่างเครื่องอ่านแบบต่างๆ [1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 หลักการและเทคนิคที่ใช้

เทคนิคในการรับส่งข้อมูลระหว่างเครื่องอ่านและแท็กจะใช้หลักการมอดูเลตทางแอมพลิจูด หรือใช้การมอดูเลตทางแอมพลิจูดบวกกับการเข้ารหัสแมนเชสเตอร์ แต่ในปัจจุบันมีแท็กที่ใช้การมอดูเลตแบบอื่นๆ ด้วย เช่น การมอดูเลตแบบอื่นๆ ด้วย เช่น การมอดูเลชันแบบเฟสซีพียอิ่ง ฟรีควเอนซีซีพียอิ่งหรือการใช้การมอดูเลตทางความถี่

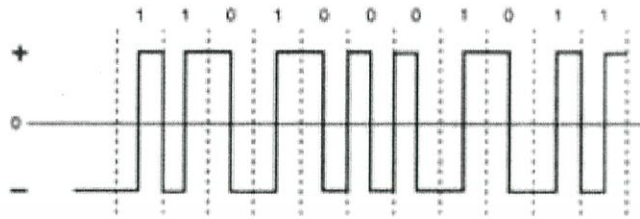
ในการรับส่งข้อมูลหรือสัญญาณวิทยุระหว่างแท็กกับเครื่องอ่านจะทำได้อย่างมีประสิทธิภาพต่อเมื่อสายอากาศมีความยาวที่เหมาะสมกับความถี่พาหะที่ใช้งาน

นอกจากการรับส่งข้อมูลแล้วสายอากาศก็ยังทำหน้าที่เป็นแหล่งจ่ายพลังงานให้กับแท็กด้วย โดยอาศัยหลักการทำงานตามแนวคิดของไมเคิลฟาราเดย์ เรื่องแรงดันเหนี่ยวนำในขดลวดที่เกิดขึ้นจากเส้นแรงแม่เหล็ก ที่มีค่าเปลี่ยนแปลงไปตามเวลา

2.3.1 การเข้ารหัสแบบแมนเชสเตอร์

เป็นการเข้ารหัสข้อมูลดิจิทัลวิธีหนึ่งก่อนที่ข้อมูลซึ่งผ่านการเข้ารหัสแล้วจะถูกส่งไปมอดูเลต เพื่อแก้ปัญหาเกี่ยวกับการชิงโครไนซ์ของข้อมูลเนื่องจากการส่งกระจายสัญญาณตามปกตินั้นหากมีการส่งสัญญาณดิจิทัลในระดับเดียวติดต่อกันเป็นช่วงยาว เช่น ส่งสัญญาณดิจิทัลที่มีค่าลอจิกเป็น 1 ออกไป 20 บิตติดต่อกัน จะทำให้การชิงโครไนซ์ของข้อมูลเกิดการคลาดเคลื่อนและทำให้รับข้อมูลผิดพลาดเพื่อป้องกันปัญหาดังกล่าวจึงจะต้องมีการนำสัญญาณดิจิทัลปกติไปผ่านเข้ารหัสเสียก่อน โดยการเข้ารหัสแบบแมนเชสเตอร์จะเปลี่ยนให้สัญญาณดิจิทัลลอจิก 0 ถูกแทนด้วยการเปลี่ยนค่าจากลอจิก 1 เป็น 0 และสัญญาณดิจิทัลลอจิก 1 แทนด้วยการเปลี่ยนค่าจากลอจิก 0 เป็น 1 ข้อดีของการเข้ารหัสแบบนี้ก็คือ ทำให้การเปลี่ยนระดับของข้อมูลทุกครั้งเป็นไปอย่างแน่นอน หรือเกิดการเข้าจังหวะกันของข้อมูลนั่นเองแต่ว่าการเข้ารหัสแบบนี้ก็มีข้อเสียอยู่กล่าวคือช่วงความถี่ที่ใช้ในการส่งข้อมูลต้องเพิ่มขึ้นเป็น 2 เท่าดังรูปที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 สัญญาณรูปคลื่นที่เข้ารหัสแบบแมนเชสเตอร์ [1]

2.3.2 ขั้นตอนการทำงานระหว่างเครื่องอ่านกับแท็ก

- 1) ตัวเครื่องอ่านจะทำการส่งสัญญาณวิทยุอย่างต่อเนื่องหรือเป็นจังหวะและรอคอยสัญญาณตอบจากตัวแท็ก
- 2) เมื่อแท็กได้รับสัญญาณคลื่นวิทยุที่ส่งมาจากเครื่องอ่านในระดับที่เพียงพอ ก็จะทำเนี่ยวนำเพื่อสร้างพลังงานป้อนให้แท็กทำงานโดยแท็กจะสร้างสัญญาณนาฬิกาเพื่อกระตุ้นให้วงจรภาคดิจิทัลในแท็กทำงาน
- 3) วงจรภาคดิจิทัลจะไปอ่านข้อมูลจากหน่วยความจำภายในและเข้ารหัสข้อมูลแล้วส่งไปยังภาคแอนะล็อกที่ทำหน้าที่มอดูเลตข้อมูล
- 4) ข้อมูลที่ถูกมอดูเลตจะถูกส่งไปยังภาคขดลวดที่ทำหน้าที่เป็นสายอากาศ เพื่อส่งไปยังเครื่องอ่าน
- 5) เครื่องอ่านจะสามารถตรวจจับสัญญาณการเปลี่ยนแปลงของแอมพลิจูดและใช้พีค ดีเทกเตอร์ ในการแปลงสัญญาณข้อมูลที่มีมอดูเลตแล้วจากแท็ก
- 6) เครื่องอ่านจะถอดรหัสข้อมูลและส่งไปยังคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมต่อไป

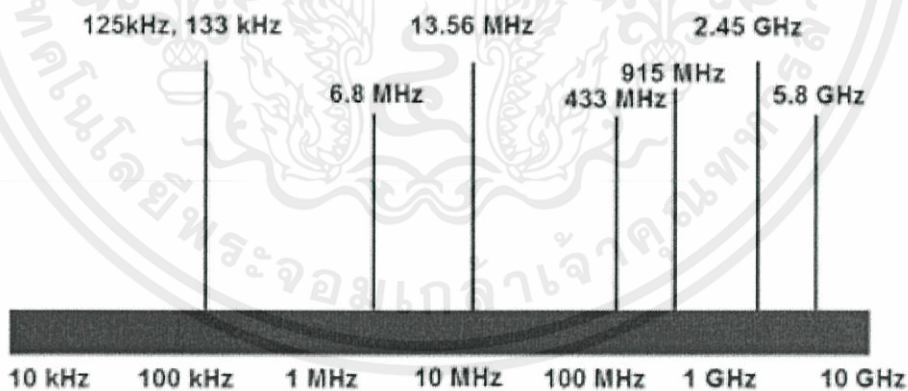
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 คลื่นพาหะในระบบอาร์เอฟไอดี

ในปัจจุบันคลื่นพาหะที่ใช้งานกันในระบบอาร์เอฟไอดีจะอยู่ในย่านที่กำหนดการใช้งานในเชิงอุตสาหกรรม วิทยาศาสตร์ และการแพทย์ สามารถใช้งานได้โดยไม่ตรงกับย่านความถี่ที่ใช้งานในการสื่อสารทั่วไป สำหรับคลื่นพาหะที่ใช้งานในระบบอาร์เอฟไอดีอาจแบ่งออกได้เป็น 3 ย่านความถี่ใช้งานหลัก ได้แก่

- 1) ย่านความถี่ต่ำ (Low Frequency) ต่ำกว่า 150 กิโลเฮิร์ตซ์
- 2) ย่านความถี่สูง (High Frequency) 13.56 เมกะเฮิร์ตซ์
- 3) ย่านความถี่สูงยิ่ง (Ultra High Frequency) 433/868/915 เมกะเฮิร์ตซ์

การใช้งาน 2 ย่านความถี่แรกจะเหมาะสำหรับใช้กับงานที่มีระยะการสื่อสารข้อมูลในระยะใกล้ เช่น การตรวจสอบการผ่านเข้าออกพื้นที่การตรวจหาและเก็บประวัติในสัตว์ ส่วนย่านความถี่สูงยิ่ง จะถูกใช้กับงานที่มีระยะการสื่อสารข้อมูลในระยะไกล เช่น ระบบเก็บค่าบริการทางด่วน และในปัจจุบันระบบอาร์เอฟไอดีกำลังถูกวิจัยและพัฒนาในย่านความถี่ไมโครเวฟที่ความถี่ 2.4 กิกะเฮิร์ตซ์ และความถี่ 5.8 กิกะเฮิร์ตซ์ เพื่อใช้งานที่ต้องการระยะอานที่ไกลกว่า 10 เมตร เป็นต้น ดังรูปที่ 2.8

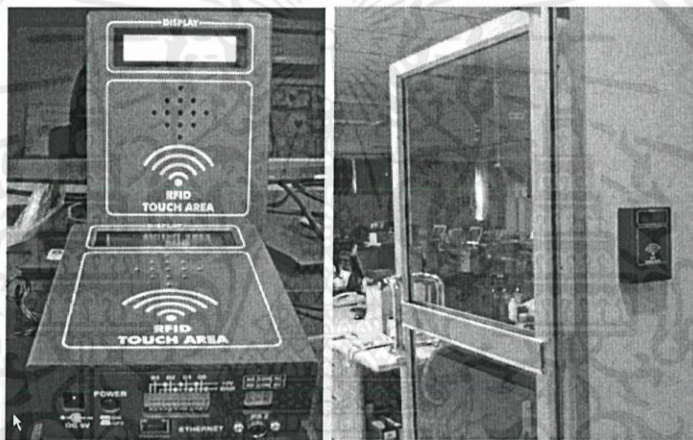


รูปที่ 2.8 ความถี่ย่านที่ระบบอาร์เอฟไอดีถูกใช้งาน [1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 ตัวอย่างการประยุกต์ใช้งานระบบอาร์เอฟไอดี

ประเทศไทยได้มีการนำเทคโนโลยีอาร์เอฟไอดีมาใช้งานมากขึ้นเรื่อยๆ ในหลายรูปแบบ เช่น บัตรทางด่วน บัตรรถโดยสาร บัตรพนักงาน บัตรเงินสด งานห้องสมุด งานปศุสัตว์ และงานระบบขนส่ง เป็นต้น โดยระบบบางระบบก็นำเข้าจากต่างประเทศทั้งระบบเข้ามาติดตั้งและใช้งาน และบางระบบก็เริ่มมีการพัฒนาขึ้นใช้งานเองโดยวิศวกรคนไทย ตัวอย่างดังรูปที่ 2.9 เป็นระบบการควบคุมการเข้าออก ซึ่งโดยทั่วไปจะเป็นการพัฒนาส่วนเชื่อมต่อกับเครื่องอ่าน โดยการใช้หน่วยประมวลผลขนาดเล็กไมโครคอนโทรลเลอร์



รูปที่ 2.9 ระบบควบคุมการผ่านเข้าออก [2]

2.6 การใช้งานโปรแกรมาคูอินโน (Arduino)

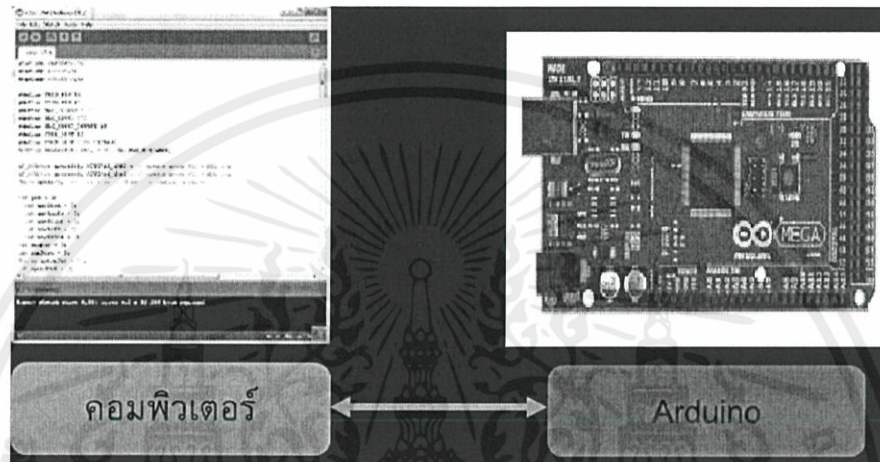
อาคูอินโนคือไมโครคอนโทรลเลอร์ชนิดหนึ่ง ซึ่งสามารถทำเองโดยใช้แบบที่มีการเปิดเผยทั่วไปก็ได้ หรือสามารถซื้อหาได้ง่าย มีราคาถูก มีซอฟต์แวร์ให้ใช้งานฟรี สามารถนำไปใช้งานทั่วไปหรือแบบธุรกิจได้โดยไม่ต้องเสียค่าลิขสิทธิ์ เป็นรูปแบบที่มีข้อมูลมากที่สุดบนอินเทอร์เน็ต มีการพัฒนาง่าย เพราะมีตัวอย่างมากมาย และสามารถใช้อำสั่งเขียนโปรแกรมได้เสมือนโปรแกรมภาษาขั้นสูงทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1 ขั้นตอนการใช้งานโปรแกรมอาดูอิโน้

1) เขียนโปรแกรมบนคอมพิวเตอร์ โดยดาวน์โหลดโปรแกรมอาดูอิโน้ จาก <http://www.arduino.cc/en/main/software>

2) ต่อบอร์ดเข้ากับคอมพิวเตอร์ ดังรูป 2.10



รูปที่ 2.10 ต่อบอร์ดอาดูอิโน้เข้ากับคอมพิวเตอร์ [5]

3) เปิดโปรแกรมอาดูอิโน้ซึ่งจะมีลักษณะดังรูป 2.11

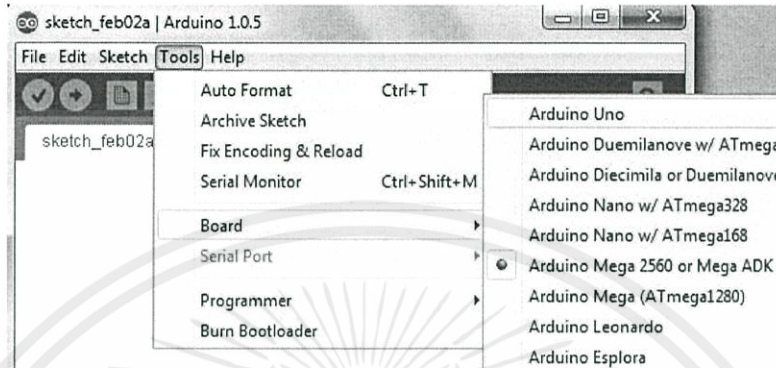


รูปที่ 2.11 หน้าต่างโปรแกรมอาดูอิโน้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้นเพื่อใช้ในการทำโครงงาน ไม่ควรนำออกจำหน่ายโดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

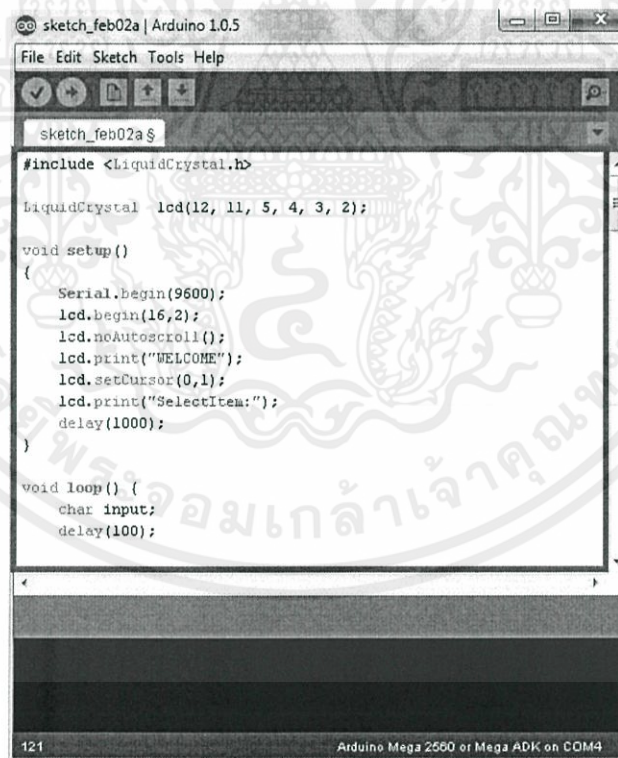
4) กำหนดบอร์ดการใช้งานให้ตรงกับบอร์ดและพอร์ตที่ใช้งาน ตัวอย่างดัง

รูป 2.12



รูปที่ 2.12 การกำหนดบอร์ดและพอร์ตการใช้งาน

5) เขียนข้อมูลลงในพื้นที่ว่าง ดังในรูป 2.13



รูปที่ 2.13 พื้นที่ที่ใช้สำหรับเขียนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) กดปุ่มเพื่อตรวจสอบความถูกต้องและคอมไพล์ (Compile) โค้ดโปรแกรมอาดีโนเพื่อตรวจสอบความผิดพลาดของโค้ด โดยการกดปุ่มดังรูป 2.14

```

sketch_feb02a $
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  Serial.begin(9600);
  lcd.begin(16,2);
  lcd.noAutoscroll();
  lcd.print("WELCOME");
  lcd.setCursor(0,1);
  lcd.print("SelectItem:");
  delay(1000);
}

void loop() {
  char input;
  delay(100);
}

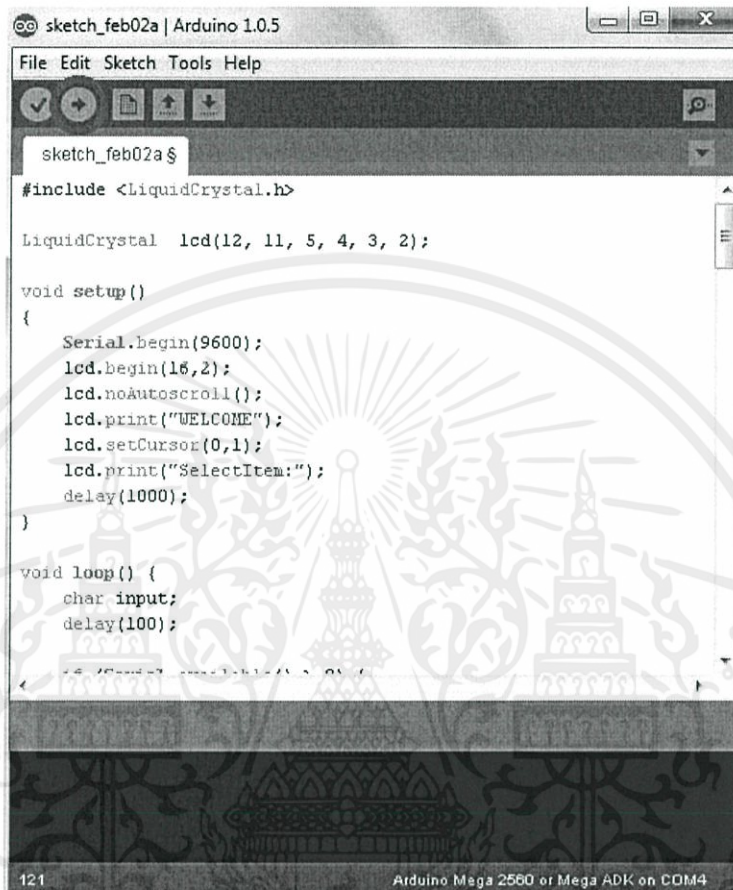
```

121 Arduino Mega 2560 or Mega ADK on COM4

รูปที่ 2.14 การตรวจสอบความผิดพลาดของคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) อัฟโหลดข้อมูล โดยการกดปุ่มดังรูป 2.15



```

sketch_feb02a | Arduino 1.0.5
File Edit Sketch Tools Help
sketch_feb02a $
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  Serial.begin(9600);
  lcd.begin(16,2);
  lcd.noAutoscroll();
  lcd.print("WELCOME");
  lcd.setCursor(0,1);
  lcd.print("SelectItem:");
  delay(1000);
}

void loop() {
  char input;
  delay(100);
}

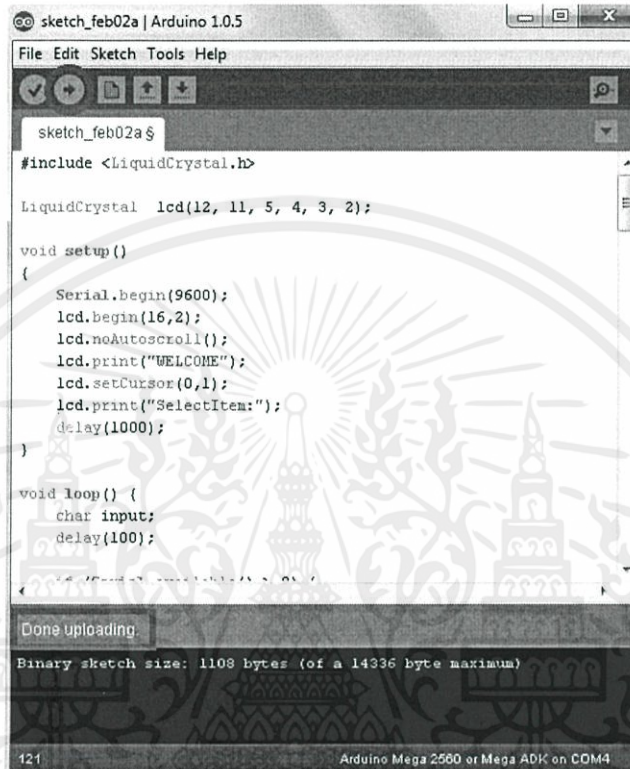
```

121 Arduino Mega 2560 or Mega ADK on COM4

รูปที่ 2.15 การอัฟโหลดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) เมื่ออัปโหลดข้อมูลสำเร็จ จะขึ้นข้อความว่าอัปโหลดสำเร็จ (Done uploading) ดังรูป 2.16



```

sketch_feb02a | Arduino 1.0.5
File Edit Sketch Tools Help
sketch_feb02a $
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  Serial.begin(9600);
  lcd.begin(16,2);
  lcd.noAutoscroll();
  lcd.print("WELCOME");
  lcd.setCursor(0,1);
  lcd.print("SelectItem:");
  delay(1000);
}

void loop() {
  char input;
  delay(100);
}

Done uploading.
Binary sketch size: 1108 bytes (of a 14336 byte maximum)
121 Arduino Mega 2560 or Mega ADK on COM4

```

รูปที่ 2.16 หน้าต่างเมื่ออัปโหลดข้อมูลสำเร็จ

2.7 การจัดการฐานข้อมูลมายเอสคิวแอลโดยใช้โปรแกรมแอปเซิร์ฟ (AppServ)

ดาวน์โหลดโปรแกรมแอปเซิร์ฟที่เว็บไซต์ <http://www.appservnetwork.com> โดยเลือกเวอร์ชันที่ต้องการติดตั้ง

2.7.1 ขั้นตอนการติดตั้งโปรแกรมแอปเซิร์ฟ

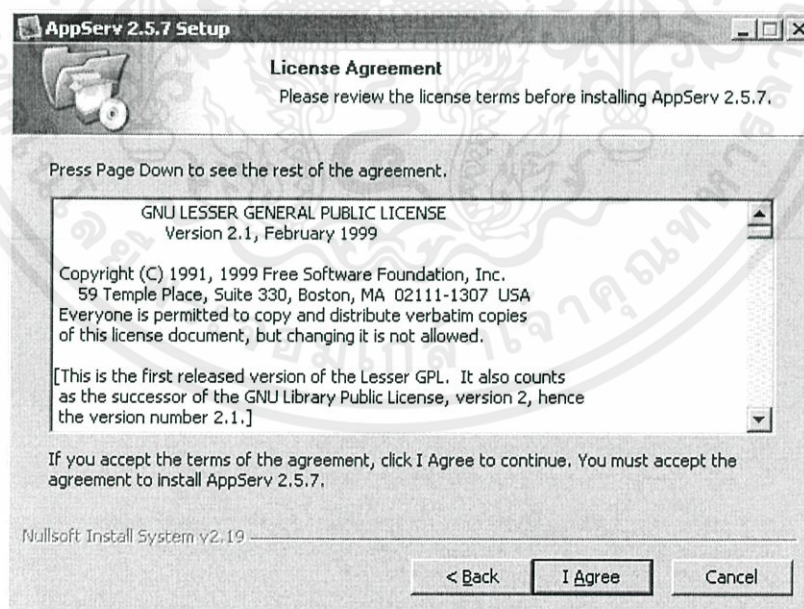
1) ดับเบิลคลิกไฟล์ appserv-win32-x.x.x.exe เพื่อทำการติดตั้ง จะปรากฏหน้าจอตามรูปที่ 2.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 ขั้นตอนการติดตั้งโปรแกรมแอฟเซิร์ฟ [9]

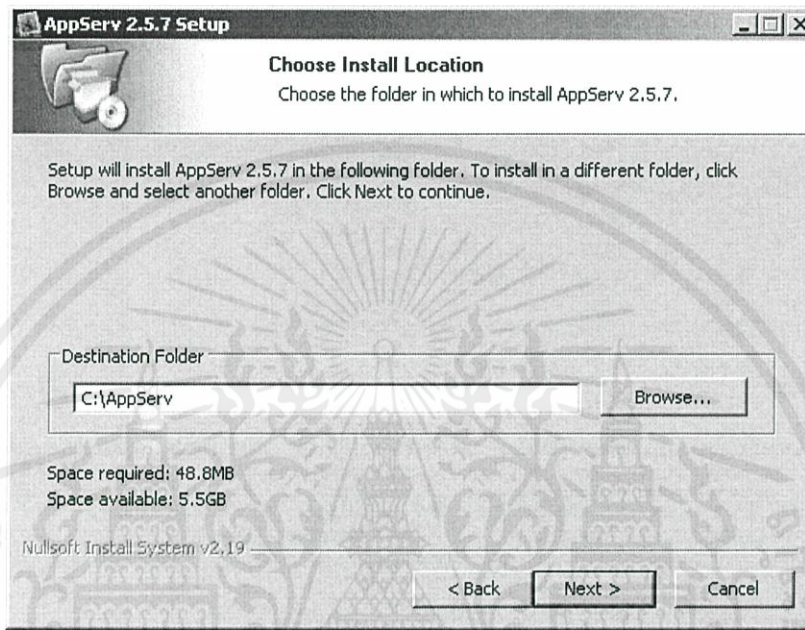
2) หากยอมรับเงื่อนไขให้กดถัดไปเพื่อเข้าสู่การติดตั้งในขั้นต่อไป แต่หากว่าไม่ยอมรับเงื่อนไขให้กดยกเลิก เพื่อออกจากการติดตั้งโปรแกรมแอฟเซิร์ฟดังรูปตัวอย่างที่ 2.18



รูปที่ 2.18 รายละเอียดเงื่อนไขการติดตั้ง [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) เข้าสู่ขั้นตอนการเลือกปลายทางที่ต้องการติดตั้ง โดยค่าเริ่มต้นปลายทางที่ติดตั้งจะเป็น C:\AppServ หากต้องการเปลี่ยนปลายทางที่ติดตั้ง ให้กดเลือกปลายทางที่ต้องการตามรูปที่ 2.19 เมื่อเลือกปลายทางเสร็จสิ้นให้กดปุ่มถัดไปเพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป



รูปที่ 2.19 เลือกปลายทางการติดตั้งโปรแกรมแอฟเซิร์ฟ [9]

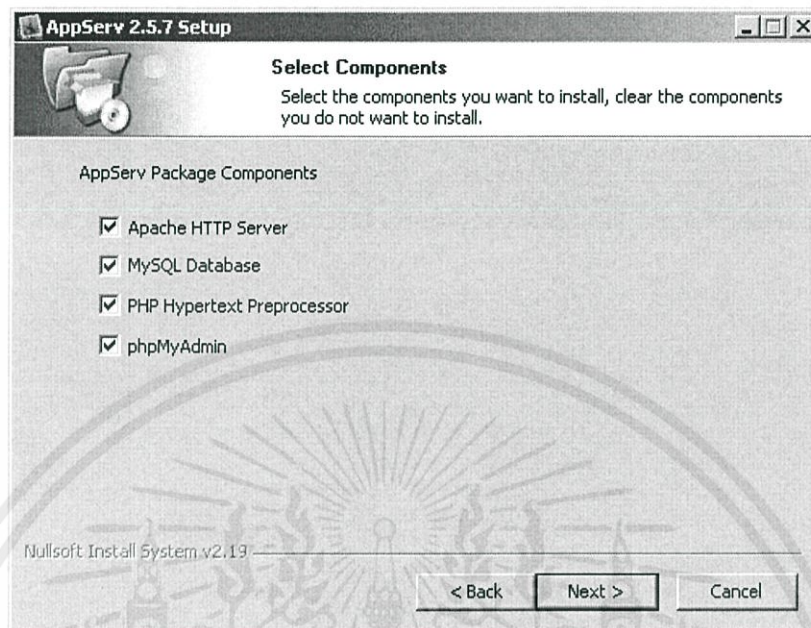
4) เลือกแพ็คเกจคอมพิวเตอร์ที่ต้องการติดตั้ง โดยค่าเริ่มต้นนั้นจะให้เลือกทุกแพ็คเกจแต่หากว่าผู้ใช้งานต้องการเลือกเฉพาะบางแพ็คเกจก็สามารถเลือกตามข้อที่ต้องการออก โดยรายละเอียดแต่ละแพ็คเกจมีดังนี้

- Apache HTTP Server คือ โปรแกรมที่ทำหน้าเป็นเว็บเซิร์ฟเวอร์
- MySQL Database คือ โปรแกรมที่ทำหน้าเป็นดาต้าเบสเซิร์ฟเวอร์
- PHP Hypertext Preprocessor คือ โปรแกรมที่ทำหน้าประมวลผลการทำงานของภาษาพีเอชพี(PHP)
- phpMyAdmin คือ โปรแกรมที่ใช้ในการบริหารจัดการฐานข้อมูลมายเอสคิวแอลผ่านเว็บไซต์

เมื่อทำการเลือกแพ็คเกจตามรูปที่ 2.20 เรียบร้อยแล้วให้กดถัดไปเพื่อเข้า

สู่ขั้นตอนการติดตั้งต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 เลือกแพ็คเกจคอมโพเนนต์ที่ต้องการติดตั้ง [9]

5) กำหนดค่าคอนฟิกของอะแพชีเว็บเซิร์ฟเวอร์มีอยู่ด้วยกันทั้งหมด 3 ส่วนตามรูปที่ 2.21 คือ

Server Name คือช่องสำหรับป้อนชื่อเว็บเซิร์ฟเวอร์ของท่าน เช่น www.appservnetwork.com

Admin Email คือช่องสำหรับป้อนข้อมูล อีเมลผู้ดูแลระบบ เช่น root@appservnetwork.com

HTTP Port คือช่องสำหรับระบุพอร์ตที่จะเรียกใช้งานอะแพชีเว็บเซิร์ฟเวอร์โดยทั่วไปแล้วจะมีค่าหลักคือ 80 หากว่าต้องการหลีกเลี่ยงการใช้พอร์ต 80 ก็สามารรถแก้ไขได้หากมีการเปลี่ยนแปลงพอร์ตการเข้าใช้งานเว็บเซิร์ฟเวอร์แล้ว ทุกครั้งที่เรียกใช้งานเว็บไซต์จำเป็นต้องระบุหมายเลขพอร์ตด้วย เช่น หากเลือกใช้พอร์ต 99 ในการเข้าเว็บไซต์ทุกครั้งต้องใช้ <http://www.appservnetwork.com:99> จึงจะสามารถเข้าใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AppServ 2.5.7 Setup

Apache HTTP Server Information
Please enter your server's information.

Server Name (e.g. www.appservnetwork.com)
localhost

Administrator's Email Address (e.g. webmaster@gmail.com)
root@appservnetwork.com

Apache HTTP Port (Default : 80)
80

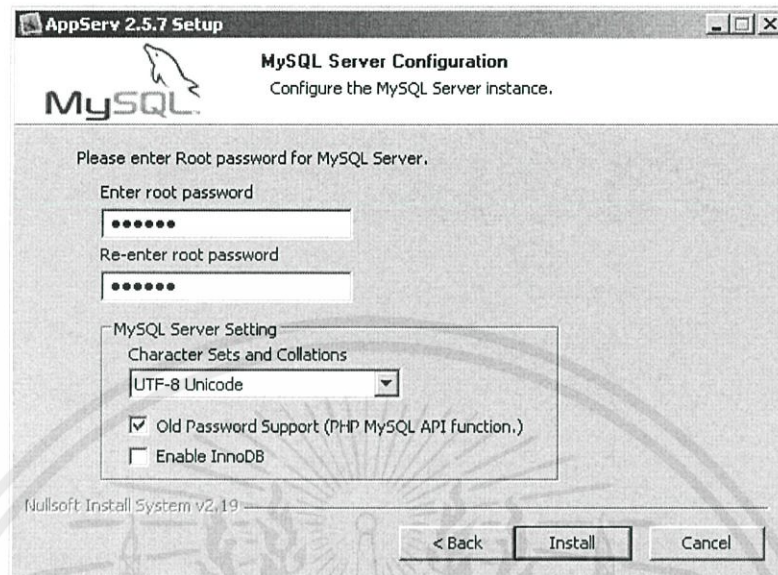
Nullsoft Install System v2.19

< Back Next > Cancel

รูปที่ 2.21 การกำหนดค่าคอนฟิกค่าอะแพชีเว็บเซิร์ฟเวอร์ [9]

- 6) กำหนดค่าคอนฟิกของมายเอสคิวแอลดาต้าเบสมืออยู่ด้วยกันทั้งหมด 3 ส่วน ตามรูปที่ 2.22 คือ
- Root Password คือช่องสำหรับป้อน รหัสผ่านการใช้งานฐานข้อมูลของ Root หรือผู้ดูแลระบบครั้งที่ใช้งานฐานข้อมูลในลักษณะที่เป็นผู้ดูแลระบบ ให้ระบุ user คือ root
 - Character Sets ใช้ในการกำหนดค่าระบบภาษาที่ใช้ในการจัดเก็บฐานข้อมูล, เรียงลำดับฐานข้อมูล, นำเข้าฐานข้อมูล, ส่งออกฐานข้อมูล, ติดต่อฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 การกำหนดค่าคอนฟิกของมายเอสคิวแอลดาต้าเบส [9]

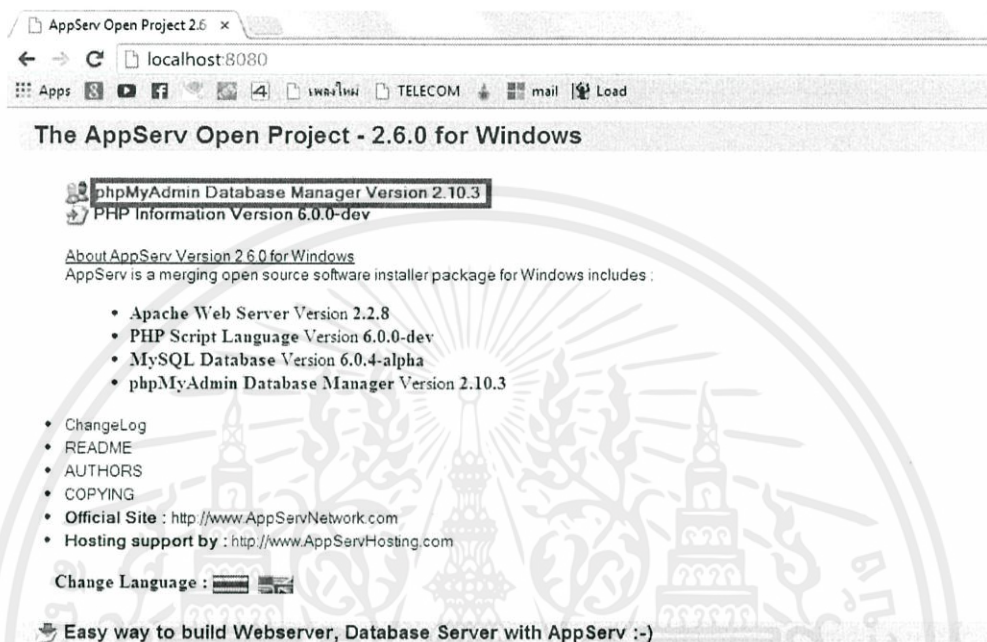
7) สิ้นสุดขั้นตอนการติดตั้งโปรแกรมแอปเซิร์ฟสำหรับขั้นตอนสุดท้ายนี้จะมีให้เลือกว่าต้องการสั่งให้มีการรันอะแพชชีและมายเอสคิวแอลทันทีหรือไม่ จากนั้นกดปุ่มเสร็จสิ้นเพื่อเสร็จสิ้นการติดตั้งโปรแกรมแอปเซิร์ฟดังรูปที่ 2.23



เอกสารนี้เป็นเอกสารที่สงวนรูปที่ 2.23 หน้าขั้นตอนสิ้นสุดการติดตั้งโปรแกรมแอปเซิร์ฟ [9] ใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

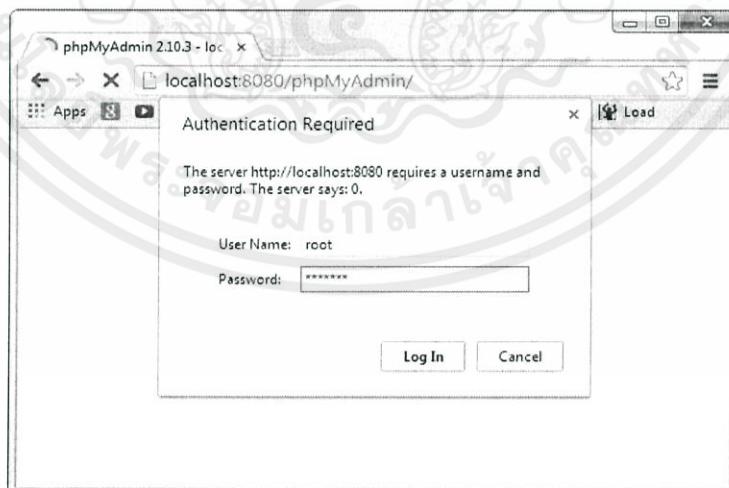
2.7.2 การจัดการฐานข้อมูลมายเอสคิวแอล

1) หน้าต่างโปรแกรมแอฟเซิร์ฟกดเข้า phpMyadmin ดังรูปที่ 2.24



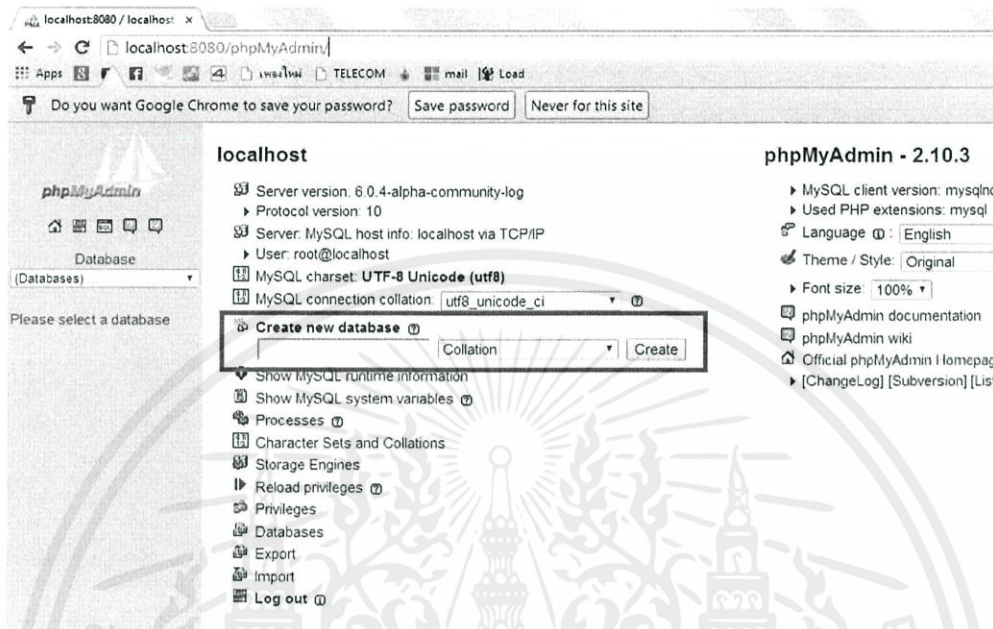
รูปที่ 2.24 หน้าต่างโปรแกรมแอฟเซิร์ฟ

2) หน้าต่างโปรแกรม phpMyadmin กดล็อกอินเข้าไป ดังรูปที่ 2.25



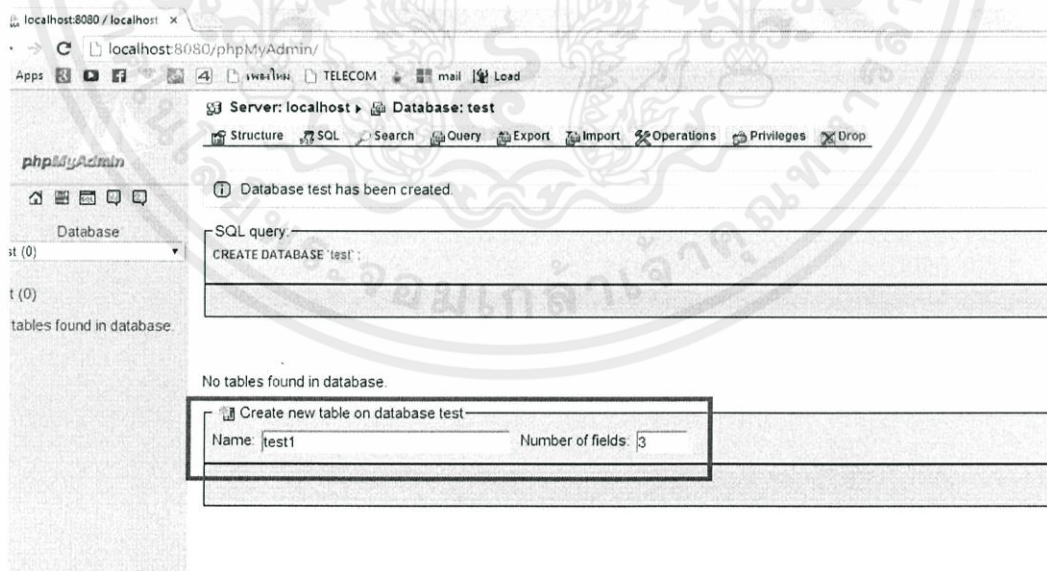
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 2.25 หน้าต่างโปรแกรม phpMyadmin ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) สร้างฐานข้อมูลโดยการเขียนชื่อที่ต้องการแล้วกดปุ่มสร้างดังรูปที่ 2.26



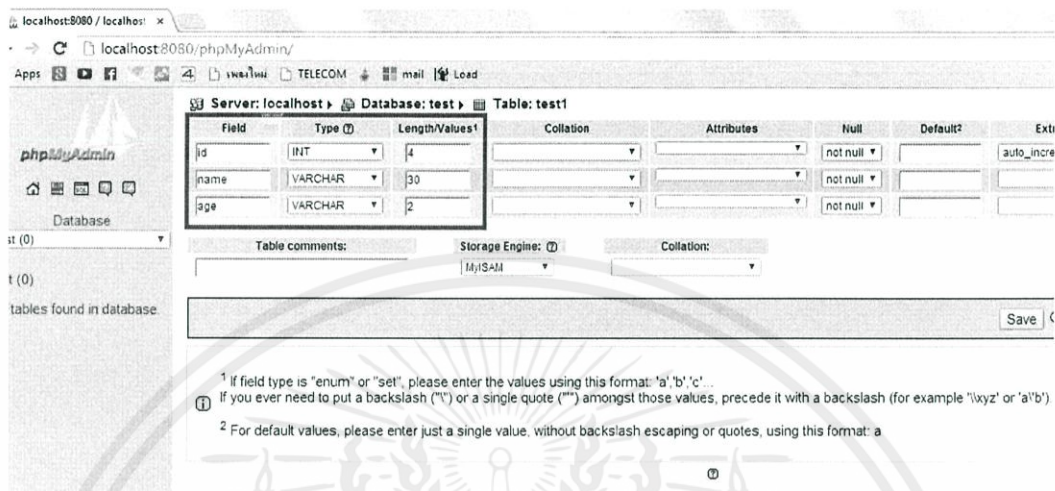
รูปที่ 2.26 การสร้างฐานข้อมูลใหม่ในโปรแกรม phpMyadmin

4) สร้างตารางและจำนวนช่องที่ต้องการดังรูปที่ 2.27



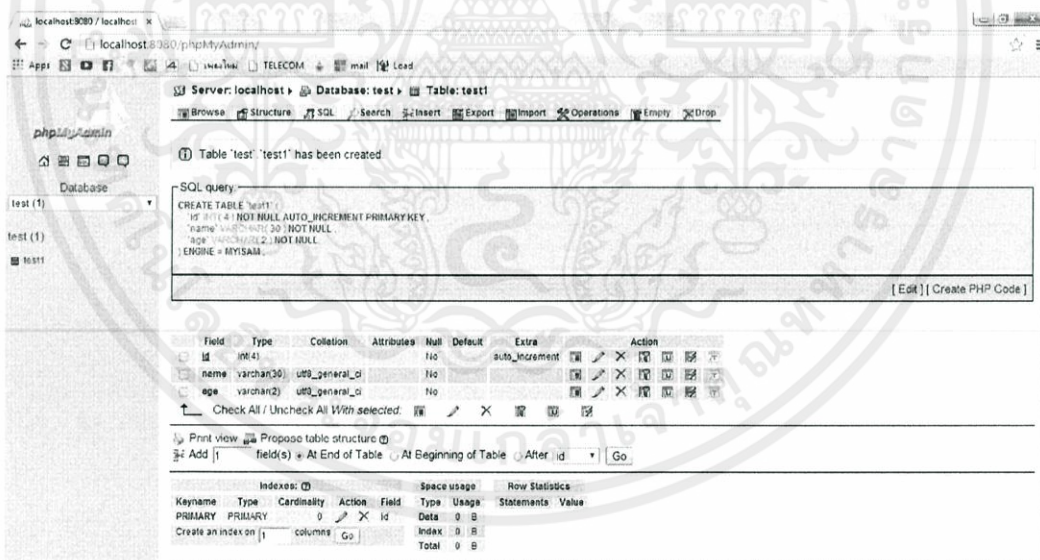
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูอาจารย์ในการใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) ตั้งค่าแต่ละช่องว่างแล้วกดบันทึก ดังรูปที่ 2.28



รูปที่ 2.28 การตั้งค่าตารางใน phpAdmin

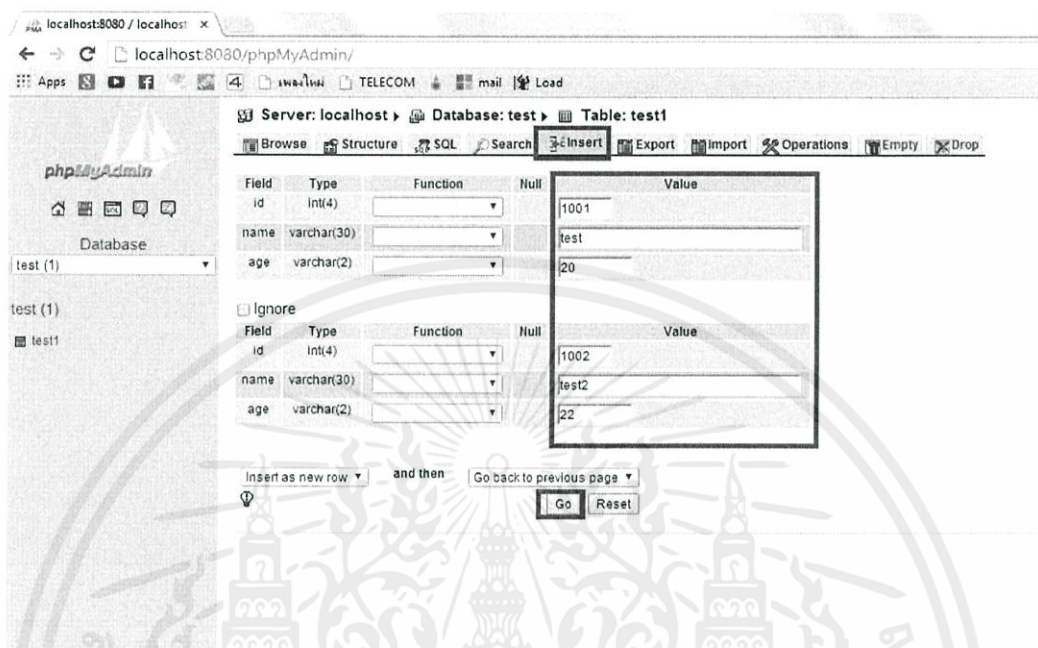
6) หน้าต่างเมื่อสร้างตารางเสร็จแล้ว ดังรูปที่ 2.29



รูปที่ 2.29 หน้าต่าง phpMyadmin เมื่อสร้างตารางเสร็จแล้ว

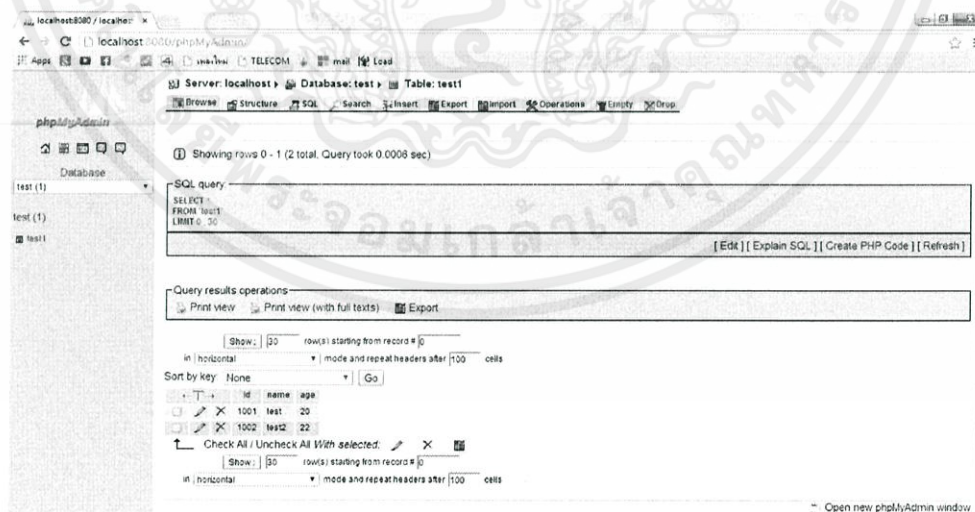
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) เลือกแถบ insert เขียนข้อมูลในช่อง value แล้วกด go ดังรูปที่ 2.30



รูปที่ 2.30 ช่องสำหรับบันทึกข้อมูลใน phpMyadmin

8) หน้าต่าง phpMyadmin เมื่อข้อมูลถูกบันทึกเรียบร้อยแล้ว ดังรูปที่ 2.31



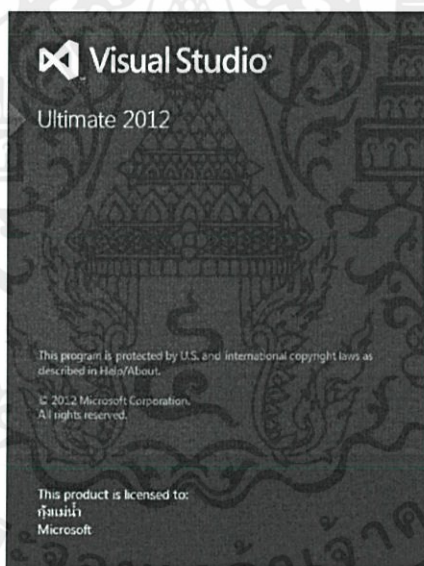
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี 2561
รูปที่ 2.31 หน้าต่าง phpMyadmin เมื่อข้อมูลถูกบันทึกเรียบร้อยแล้ว
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 การใช้งานโปรแกรมวิซวลเบสิก

วิซวลเบสิกเป็นภาษาโปรแกรมภาษาหนึ่ง พัฒนาโดยบริษัทไมโครซอฟท์ บรรจุอยู่ภายใต้ผลิตภัณฑ์ชุดหนึ่งชื่อว่า Microsoft Visual Studio ประกอบไปด้วยภาษาต่างๆ หลายภาษา เช่น Visual Basic, Visual C++, Visual FoxPro, Visual InterDev, Visual J++ และ Visual SourceSafe โดย วัตถุประสงค์คือต้องการให้เป็นภาษาที่ใช้สำหรับพัฒนาโปรแกรมหรือแอปพลิเคชัน ที่ทำงานภายใต้ระบบปฏิบัติการ Windows และ Windows NT การใช้งาน Visual Basic นั้นค่อนข้างง่าย โปรแกรมเมอร์สามารถวาดและวางองค์ประกอบต่างๆ บนหน้าจอเพื่อติดต่อกับผู้ใช้ (User Interface) ได้ตามความต้องการ

2.8.1 ขั้นตอนการใช้งานโปรแกรมวิซวลเบสิก

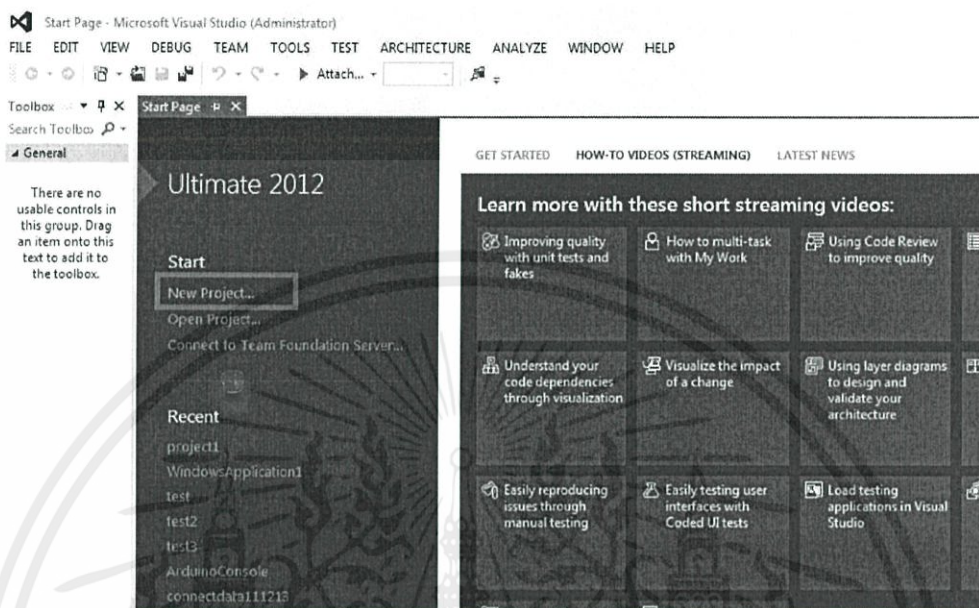
1) เมื่อกดเข้าโปรแกรมวิซวลเบสิกจะแสดงดังรูปที่ 2.32



รูปที่ 2.32 โปรแกรมวิซวลเบสิก

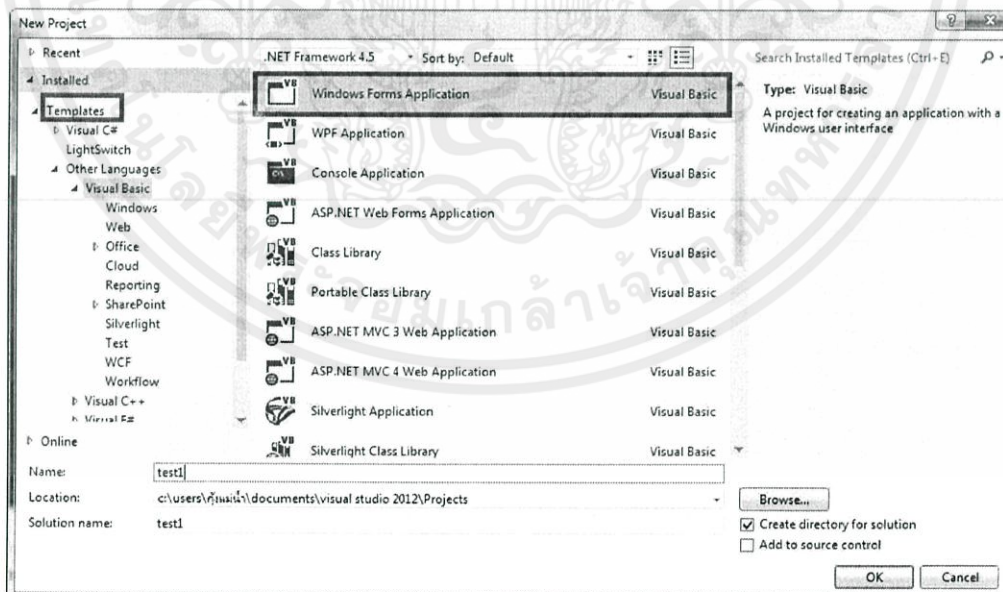
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) สร้างงานใหม่โดยการกดโปรเจกใหม่ (New Project) ดังรูปที่ 2.33



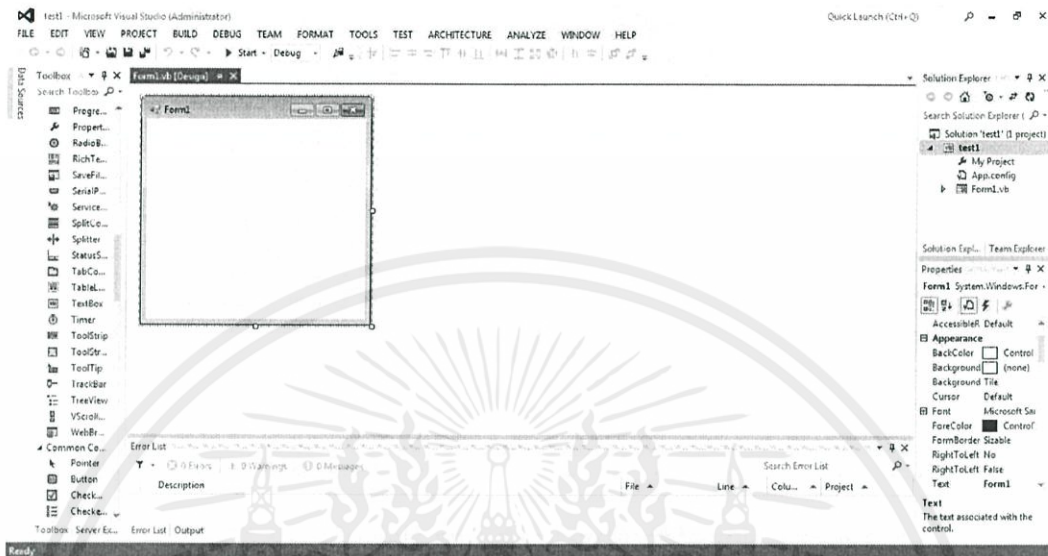
รูปที่ 2.33 การสร้างงานใหม่ในโปรแกรมวิซวลเบสิก

3) เลือกภาษาและประเภทการใช้งานโดยกดเลือกดังรูปที่ 2.34



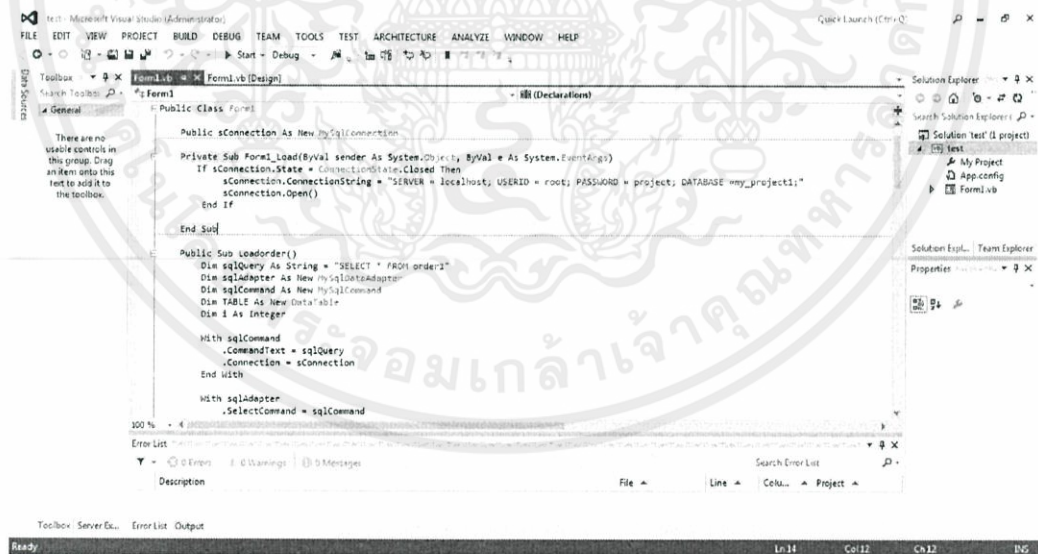
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ 2.34 การเลือกภาษาและประเภทในโปรแกรมวิซวลเบสิก ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) หน้าต่างการทำงานแสดงดังรูปที่ 2.35



รูปที่ 2.35 หน้าต่างการทำงานในโปรแกรมวิซวลเบสิก

5) หน้าต่างสำหรับการเขียนโค้ดแสดงดังรูปที่ 2.36

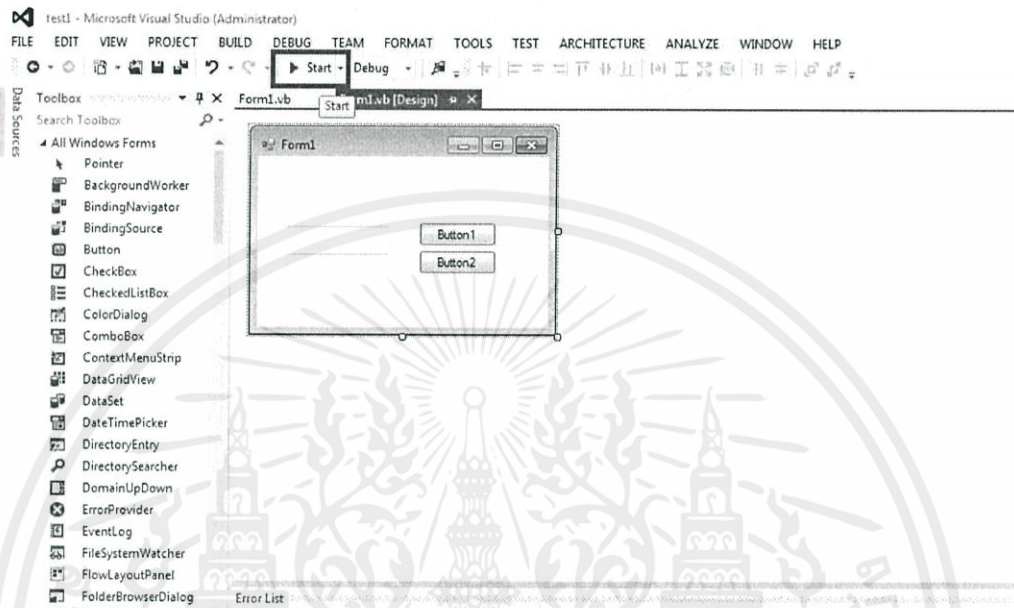


รูปที่ 2.36 หน้าต่างสำหรับการเขียนโค้ดในโปรแกรมวิซวลเบสิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

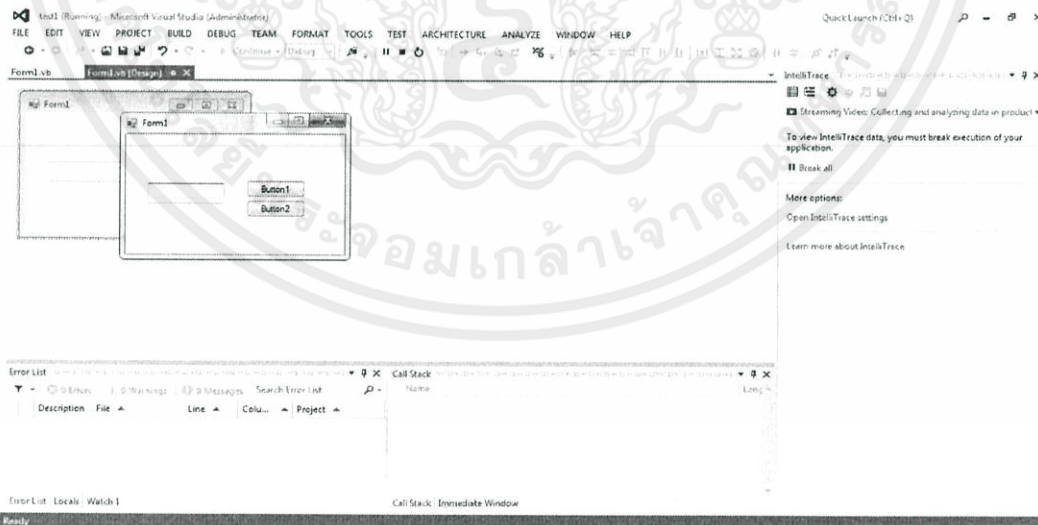
6) เมื่อเขียนโค้ดเสร็จ ให้กดเริ่มต้น (start) เพื่อทดสอบการทำงานแสดงดัง

รูปที่ 2.37



รูปที่ 2.37 ปุ่มเริ่มต้น (start) ในโปรแกรมวิซวลเบสิก

7) เมื่อกด start เสร็จจะขึ้นหน้าต่างการทำงานแสดงดังรูปที่ 2.38



รูปที่ 2.38 หน้าต่างหลังจากสั่งเริ่มต้น (start) ในโปรแกรมวิซวลเบสิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการเชิงนามธรรมเพื่อการศึกษาค้นคว้า โดยผู้ดูแลเนื้อหาใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 การใช้งานโปรแกรมคริสตัลรีพอร์ต

คริสตัลรีพอร์ต คือเครื่องมือที่ใช้ในการออกรายงาน ซึ่งสามารถออกรายงานได้หลากหลายรูปแบบ เช่น แบบรายงานธรรมดา แบบ Cross Tab และแบบอื่นๆ ซึ่งมีเครื่องมือที่ออกแบบมาให้ง่ายต่อการใช้งาน และการติดต่อกับฐานข้อมูลสามารถทำได้หลากหลาย สามารถนำข้อมูลจาก Viewer ของเครื่องมาดูได้ ซึ่งให้ความสามารถที่หลากหลาย และการ View ก็สามารถ View ได้หลากหลาย เช่น การ View ผ่านตัวโปรแกรมเอง การ View ผ่านโปรแกรมที่เป็นโปรแกรมประยุกต์ที่ Software House ต่างๆผลิตขึ้นมา หรือดูบนเว็บ

2.9.1 ขั้นตอนการใช้งานโปรแกรมคริสตัลรีพอร์ต

1) สร้างฐานข้อมูลตามที่ต้องการทำรายงานขึ้นมา ดังรูปที่ 2.39

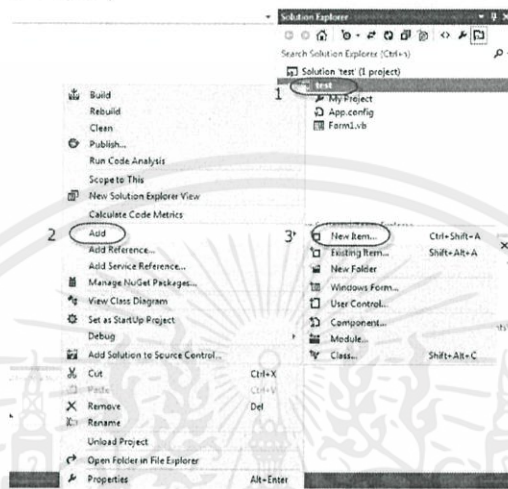
← T →	id_user	name_menu	price	number	amount	date_order	time_order
<input type="checkbox"/>  	1001	Big Burger	50	1	50	2014-03-04	12:31:38
<input type="checkbox"/>  	1001	Big Burger	50	1	50	2014-03-04	12:21:12
<input type="checkbox"/>  	1001	ChickenNuggets	50	3	150	2014-03-05	14:54:32
<input type="checkbox"/>  	1001	FreshSalad	60	2	120	2014-03-05	14:55:46
<input type="checkbox"/>  	1001	FishBurger	30	6	180	2014-03-05	15:17:27
<input type="checkbox"/>  	1001	ChickenNuggets	50	3	150	2014-03-05	15:26:43
<input type="checkbox"/>  	1001	FriedChicken	30	2	60	2014-03-05	15:29:09
<input type="checkbox"/>  	1001	FrenchFries	55	1	55	2014-03-05	15:31:56

รูปที่ 2.39 หน้าต่างแสดงการสร้างฐานข้อมูลที่ต้องการทำรายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) สร้าง dataset โดยคลิกขวาที่ project > Add > New Item ดังรูป

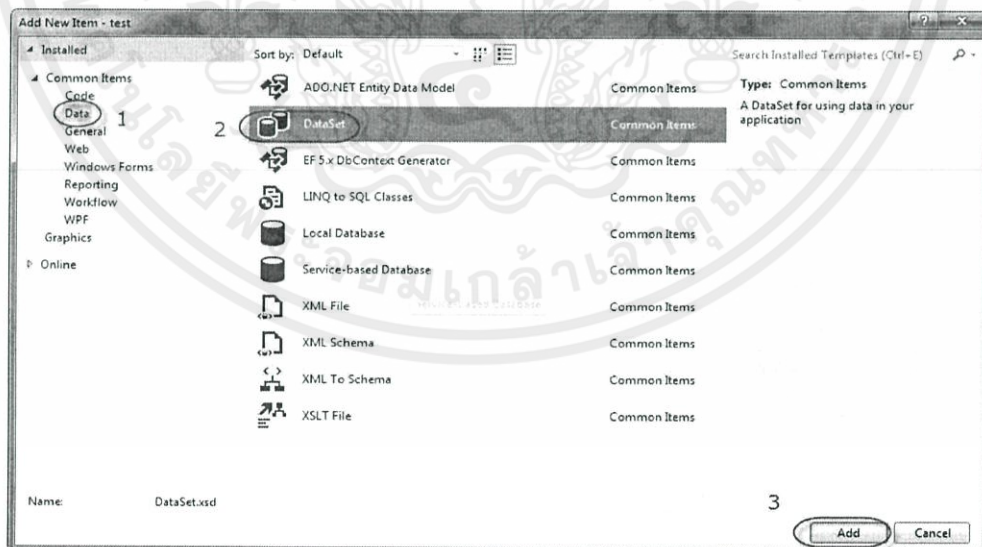
2.40



รูปที่ 2.40 หน้าต่างแสดงการสร้างดาต้าเซ็ท

3) เลือก data > dataset กำหนดชื่อฐานข้อมูล จากนั้นกด Add ดังรูป

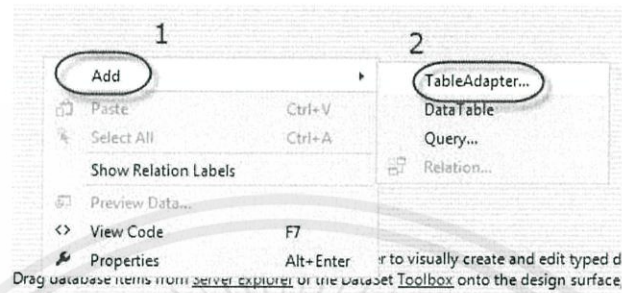
2.41



รูปที่ 2.41 หน้าต่างแสดงการกำหนดชนิดดาต้าเซ็ท

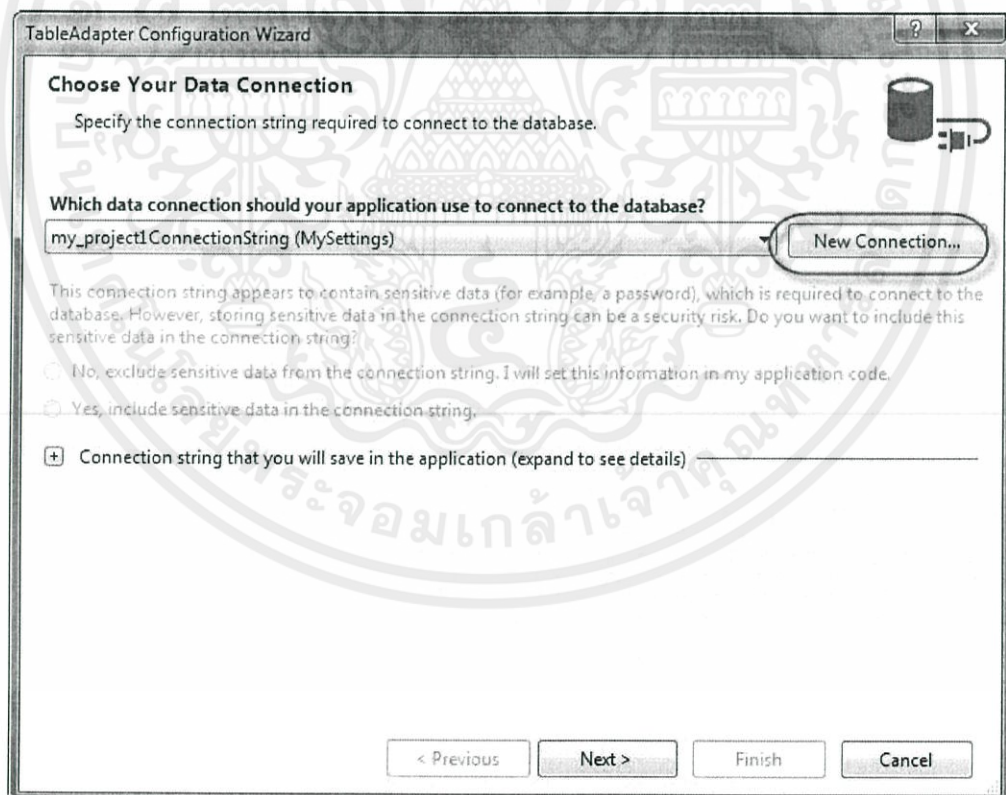
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) โปรแกรมจะทำการสร้างหน้าต่าง dataset ขึ้นมา ให้คลิกขวาในหน้าต่าง แล้วกด Add > TableAdapter ดังรูป 2.42



รูปที่ 2.42 หน้าต่างแสดงการตั้งค่าในดาต้าเซ็ท

5) ทำการเลือกฐานข้อมูลที่ต้องการเชื่อมต่อ จากนั้นเลือก yes และกด Next ดังรูป 2.43

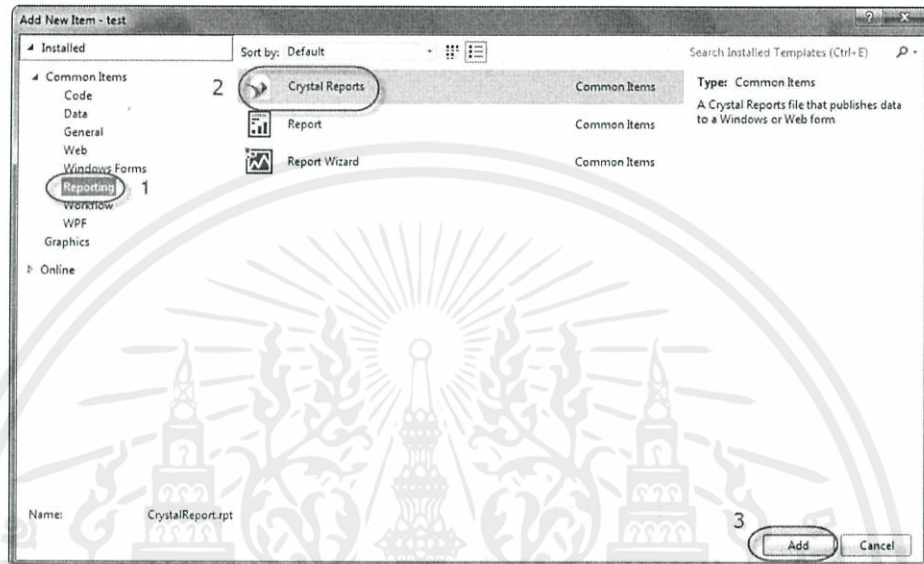


รูปที่ 2.43 หน้าต่างแสดงการเลือกฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนการสอนเท่านั้น ไม่ให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) เลือก reporting > crystal report กำหนดชื่อให้รายงาน จากนั้นกด

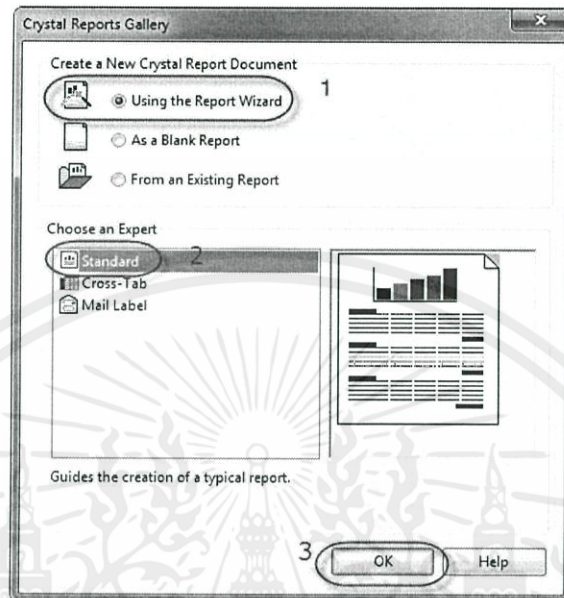
Add ดังรูป 2.44



รูปที่ 2.44 หน้าต่างแสดงการกำหนดชื่อให้รายงาน

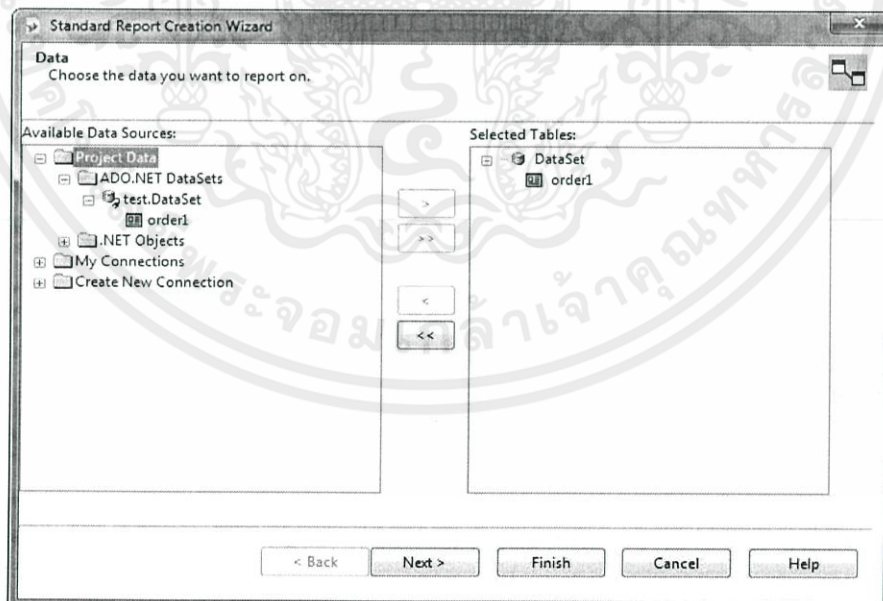
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) เลือกฟอร์มของรายงานตามแบบฟอร์มที่นำมา ดังรูป 2.45



รูปที่ 2.45 หน้าต่างแสดงการเลือกรูปแบบการแสดงผล

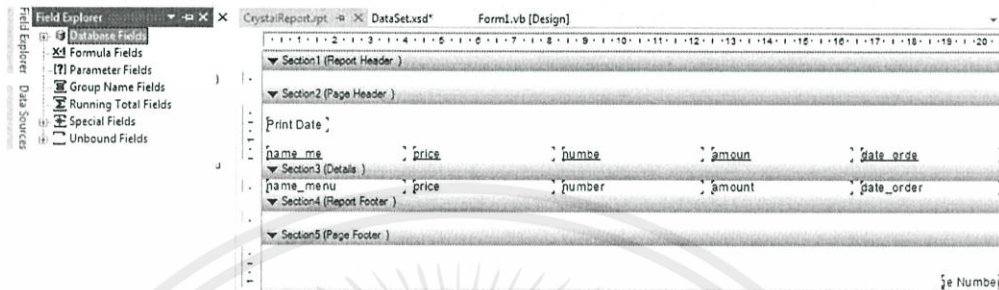
8) เลือกฐานข้อมูลที่ต้องการแสดงในรายงาน ดังรูป 2.46



รูปที่ 2.46 หน้าต่างแสดงการกำหนดเลือกฐานข้อมูลที่ต้องการแสดง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของบริษัท เซ็นทรัลคอมพิวติ้ง จำกัด ผู้ให้บริการด้านไอทีชั้นนำในประเทศไทย
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9) เลือกข้อมูลที่ต้องการให้แสดงในรายงานแล้วจะได้หน้าต่างสำหรับสร้างรายงานขึ้นมา ดังรูป 2.47



รูปที่ 2.47 หน้าต่างสำหรับสร้างรายงาน

10) ทำการรันโปรแกรมก็จะได้รูปแบบของรายงานตามที่กำหนดไว้ ดังรูป 2.48

The screenshot shows the Crystal Reports preview view. At the top, it says 'Save data in the report' and '1 / 1' pages. The date '19/04/2014' is displayed. Below is a table with the following data:

name_menu	price	number	amount
Navy	5,221	12,331	6,429
Aqua	22,856	25,935	29,836
Fuchsia	23,623	27,914	9,580
White	2,626	6,864	9,014
Red	22,301	20,895	18,057
Olive	21,350	21,769	11,296
Blue	17,719	2,721	13,422
Red	18,526	11,839	17,742
Teal	14,780	6,646	12,363
Lime	29,163	11,869	25,207

รูปที่ 2.48 หน้าต่างแสดงผลของรายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

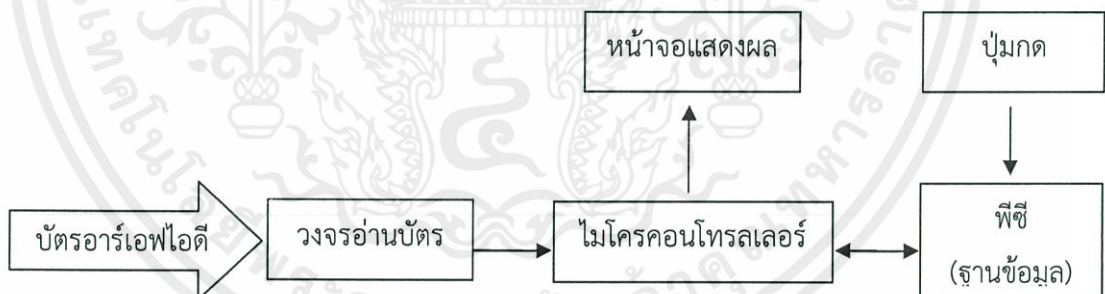
การออกแบบและการจัดทำปฏิญญานิพนธ์

ปฏิญญานิพนธ์นี้เป็นการทำระบบการชำระเงินอัตโนมัติผ่านบัตรสมาร์ทการ์ดโดยใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมอุปกรณ์อื่นๆ

3.1 การออกแบบ

3.1.1 ออกแบบการทำงานโดยรวม

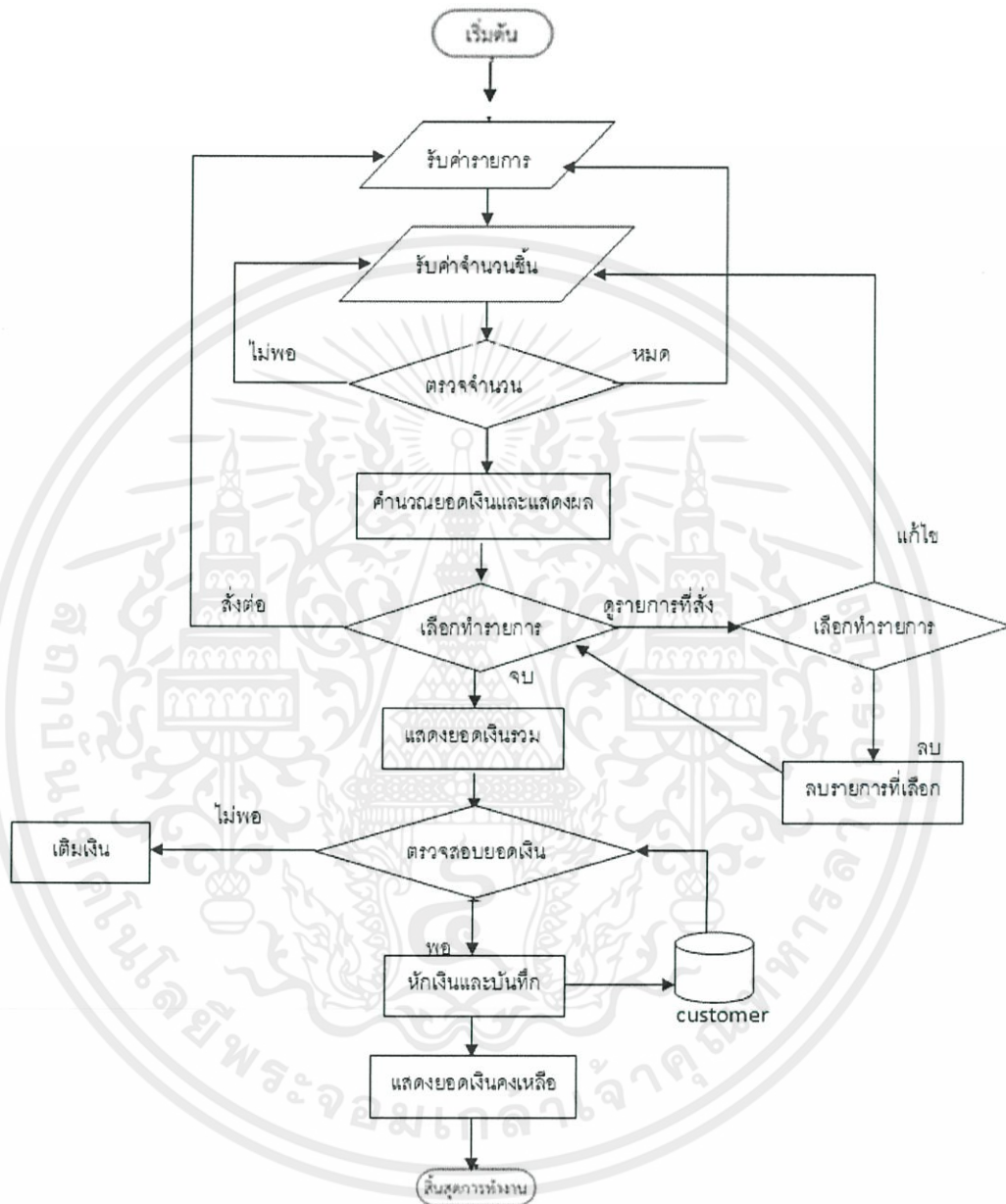
การทำงานโดยรวมของระบบชำระเงินอัตโนมัติผ่านบัตรสมาร์ทการ์ดหรือบัตรอาร์เอฟไอดีจะตามแสดงแผนผังของระบบในรูปที่ 3.1 ซึ่งจะประกอบด้วยไมโครคอนโทรลเลอร์เป็นตัวหลักที่สำคัญ ซึ่งต่ออยู่กับอุปกรณ์ในส่วนอื่นๆ เช่น หน้าจอแสดงผล วงจรอ่านบัตร และเครื่องคอมพิวเตอร์ที่มีฐานข้อมูล โดยจะรับข้อมูลจากปุ่มกดแล้วแสดงผลทางหน้าจอแสดงผล ผ่านตัวไมโครคอนโทรลเลอร์ และใช้วงจรอ่านบัตรในการชำระเงินผ่านบัตรอาร์เอฟไอดี



รูปที่ 3.1 ระบบการทำงานโดยรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.1 ออกแบบระบบการสั่งอาหารในอาตูดิโอโน

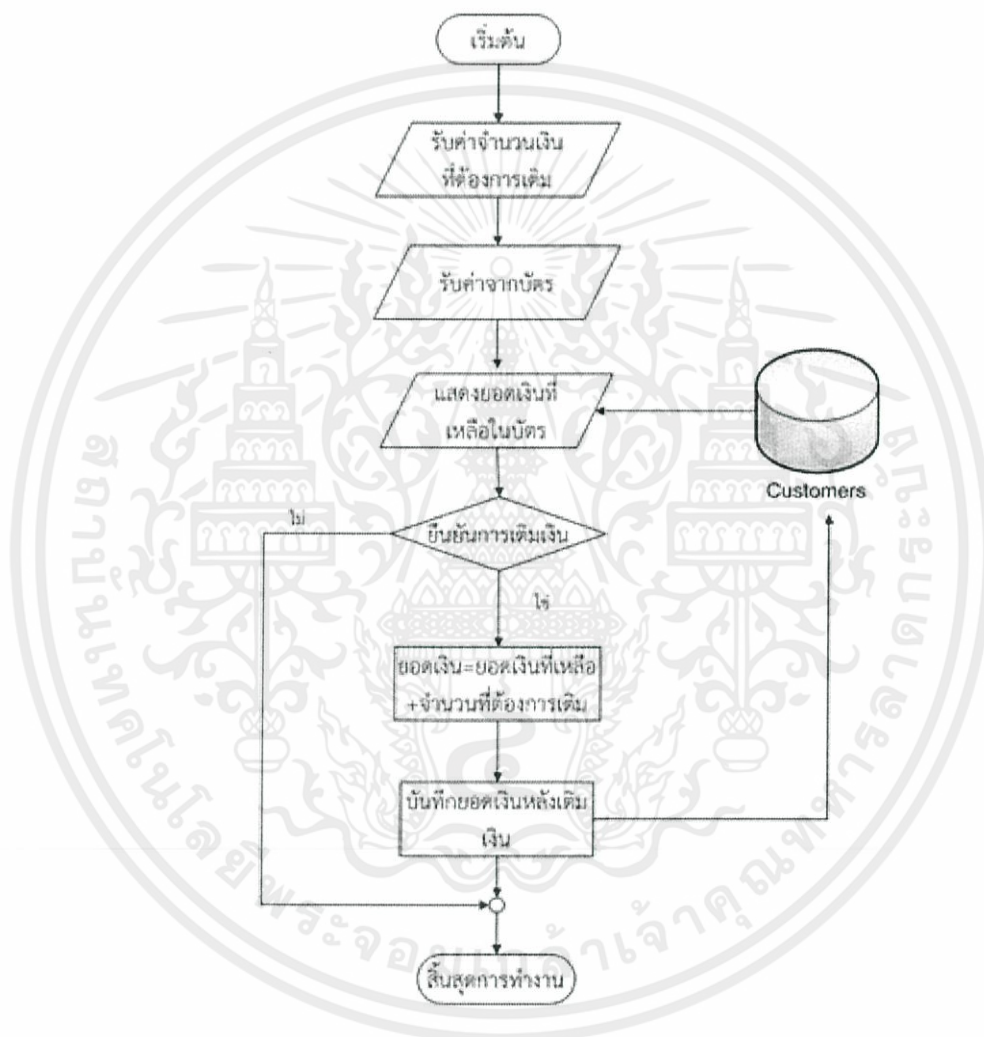


รูปที่ 3.2 โฟลว์ชาร์ตของระบบการสั่งอาหาร

โฟลว์ชาร์ตของระบบการสั่งอาหารแสดงได้ตามรูปที่ 3.2 เมื่อเริ่มต้นการทำงาน จะมีข้อความขึ้นที่หน้าจอแสดงผลว่าให้เลือกรายการ ถ้ากดสั่งอาหาร (order) จะมาต่อที่เอกสารนี้เป็นเอกสารหลังรวมรายการของอาหารที่เลือกแล้ว เมื่อผู้ใช้กดเลือกเมนูอาหารแล้ว จะมีการคำนวณยอดเงินคงเหลือ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าจอให้เลือกอาหาร ถ้าเลือกครบรายการอาหาร จะแสดงรายการอาหารที่เลือก แล้วให้กดจำนวนรายการอาหารที่ต้องการ และแสดงจำนวนเงินทั้งหมดที่ต้องชำระ ซึ่งถ้ายอดเงินไม่พอจะต้องมีการเติมเงินก่อน

3.1.1.2 ออกแบบระบบเติมเงิน



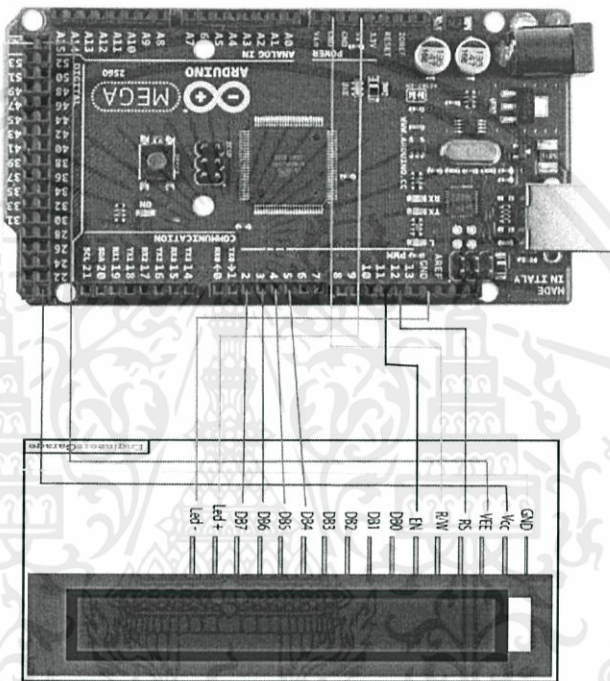
รูปที่ 3.3 โฟลว์ชาร์ตของระบบเติมเงิน

โฟลว์ชาร์ตของระบบการเติมเงินได้ตามรูปที่ 3.3 เมื่อกดเติมเงิน (top-up)

เพื่อเข้าสู่ระบบการทำงานของระบบเติมเงิน จะมีการแสดงที่หน้าจอแสดงผลให้ยืนยันว่าต้องการจะ
เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับวิทยานิพนธ์การศึกษาระดับปริญญาโท สาขาวิชาเทคโนโลยีสารสนเทศ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เติมเงิน เมื่อกดตกลง (enter) จะมีการแสดงผลที่หน้าจอให้ใส่จำนวนเงินที่ต้องการเติม และจะเก็บข้อมูลที่ฐานข้อมูลต่อไป

3.1.1.3 ออกแบบการต่อไมโครคอนโทรลเลอร์กับหน้าจอแสดงผล

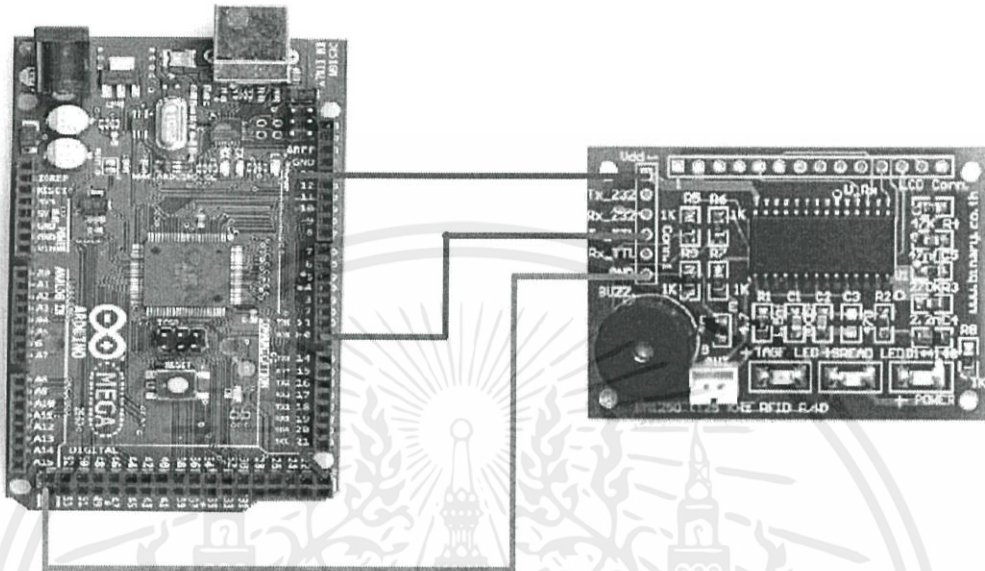


รูปที่ 3.4 การต่อไมโครคอนโทรลเลอร์กับหน้าจอแสดงผล

- 1) ภาควัดจ่ายไฟ ให้ต่อกราวด์และไฟของหน้าจอแสดงผลเข้ากับ กราวด์และไฟของไมโครคอนโทรลเลอร์ตามลำดับ
- 2) ภาควัดสัญญาณ ให้ต่อ 4 เส้นจากหน้าจอแสดงผลขา D4 ถึง D7 ไปที่บอร์ดขา 5 4 3 และ 2 ตามลำดับ
- 3) ไฟแบล็คไลท์ ให้ต่อขา LCD Led+ และ Led- ไปที่บอร์ดขา 13 และ Gnd ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.4 ออกแบบการต่อไมโครคอนโทรลเลอร์กับตัวอ่านอาร์เอฟไอดี

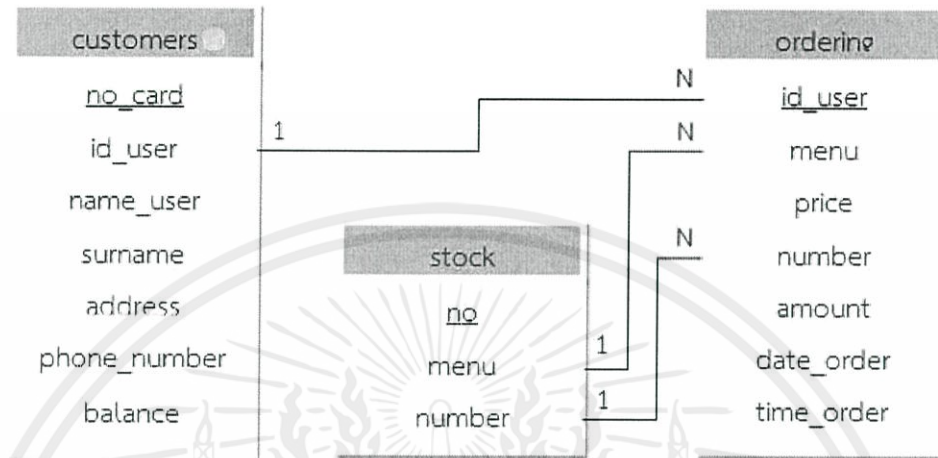


รูปที่ 3.5 การต่อไมโครคอนโทรลเลอร์กับวงจรรอ่านอาร์เอฟไอดี

การต่อไมโครคอนโทรลเลอร์กับวงจรรอ่านอาร์เอฟไอดี แสดงดังรูปที่ 3.5 โดยต่อไฟ กราวน์ และขาสัญญาณขาเข้า (Tx) ของวงจรรอ่านอาร์เอฟไอดีเข้ากับไฟ กราวน์ และสัญญาณขาเข้าของไมโครคอนโทรลเลอร์ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.5 ออกแบบฐานข้อมูลที่ใช้งาน



รูปที่ 3.6 การออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลแสดงดังรูปที่ 3.6 โดยจะมีทั้งหมดสามส่วนคือ ส่วนเก็บข้อมูลลูกค้า ส่วนเก็บจำนวนอาหาร และส่วนที่เก็บข้อมูลรายการอาหาร

3.2 เครื่องมือที่ใช้ในการทดลอง

- บอร์ด ATmega2560
- เครื่องอ่านอาร์เอฟไอดี
- หน้าจอแสดงผล
- ปุ่มกด
- เครื่องออสซิลโลสโคป
- เครื่องวัดสเปกตรัม
- สายปากคีบ
- โปรแกรม Arduino
- โปรแกรม Mysql

เอกสารนี้เป็นเอกสารที่ส่ง-โปรแกรม Visual studio 2012 ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การจัดเก็บผลการทดลอง

จากที่กล่าวไปข้างต้น ผลการทดลองสามารถจัดเก็บได้ด้วยวิธีดังนี้

- 1) สัญญาณจากเครื่องอ่านอาร์เอฟไอดีเมื่อมีการอ่านบัตร
- 2) วัดความถี่ของเครื่องอ่านอาร์เอฟไอดี
- 3) ผลการแสดงผลหน้าจอแสดงผลเทียบกับโปรแกรมสั่งอาหารในคอมพิวเตอร์
- 4) ผลการทดลองในส่วนฐานข้อมูล



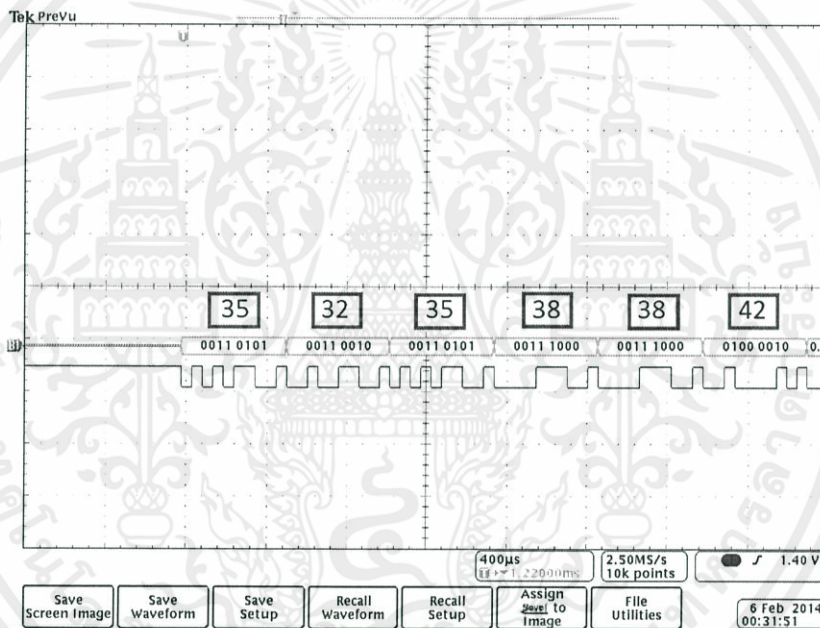
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 สัญญาณจากวงจรอ่านอาร์เอฟไอดีเมื่อมีการอ่านบัตร

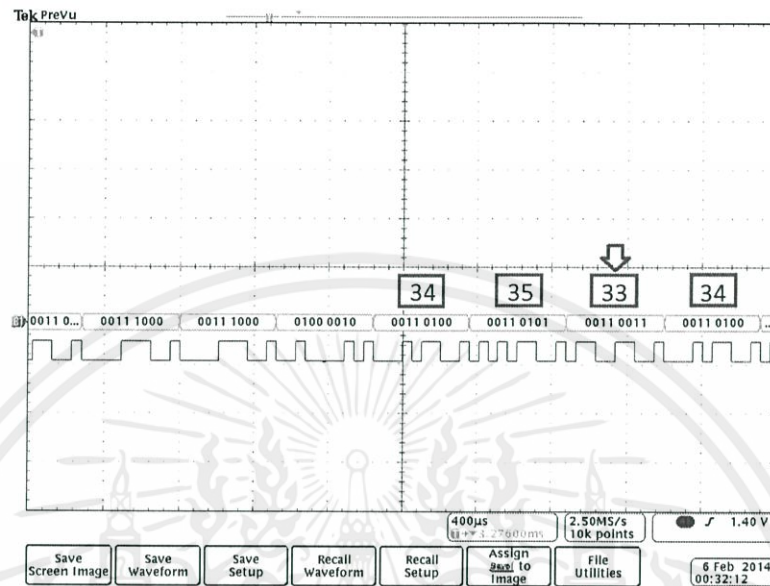
เมื่อมีการอ่านบัตร ใช้เครื่องวัดสัญญาณมาวัดสัญญาณที่ Tx จะเห็นเป็นรูปสัญญาณ โดยเป็นข้อมูลของบัตร จะเห็นข้อมูลเป็นเลข 0 และ 1 เมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 35 32 35 38 38 42 ดังรูปที่ 4.1



รูปที่ 4.1 สัญญาณพัลส์ 35 32 35 38 38 42

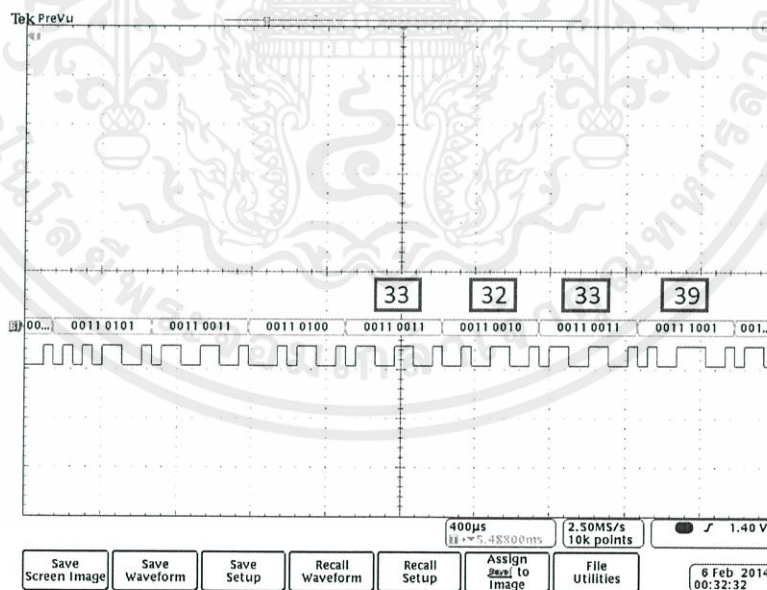
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลต่อไปเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 34 35 33 34 ดังรูปที่ 4.2



รูปที่ 4.2 สัญญาณพัลส์ 34 35 33 34

ข้อมูลต่อไปเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 33 32 33 39 ดังรูปที่ 4.3

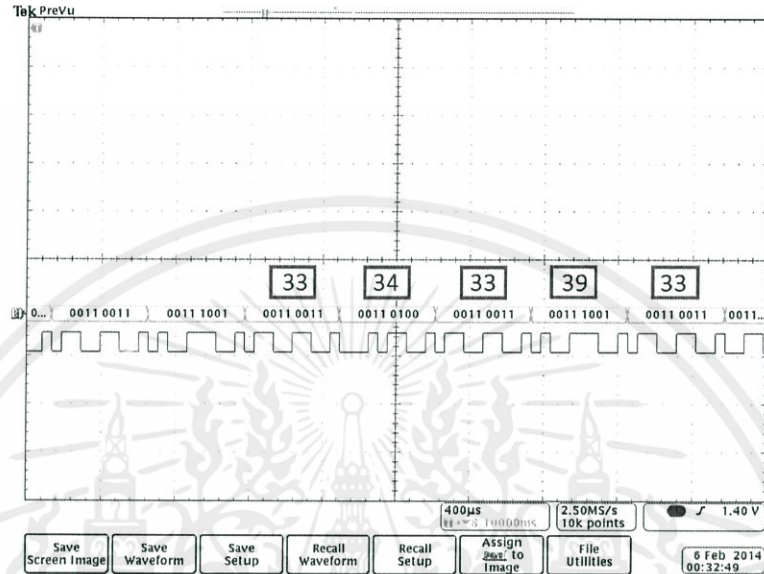


รูปที่ 4.3 สัญญาณพัลส์ 33 32 33 39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลต่อไปเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 33 34 33 39 33 ดังรูปที่

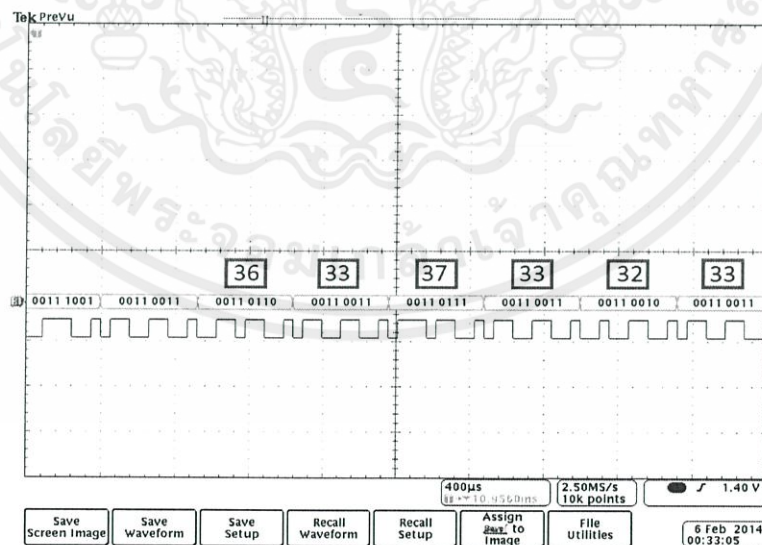
4.4



รูปที่ 4.4 สัญญาณพัลส์ 33 34 33 39 33

ข้อมูลต่อไปเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 36 33 37 33 32 33 ดังรูปที่

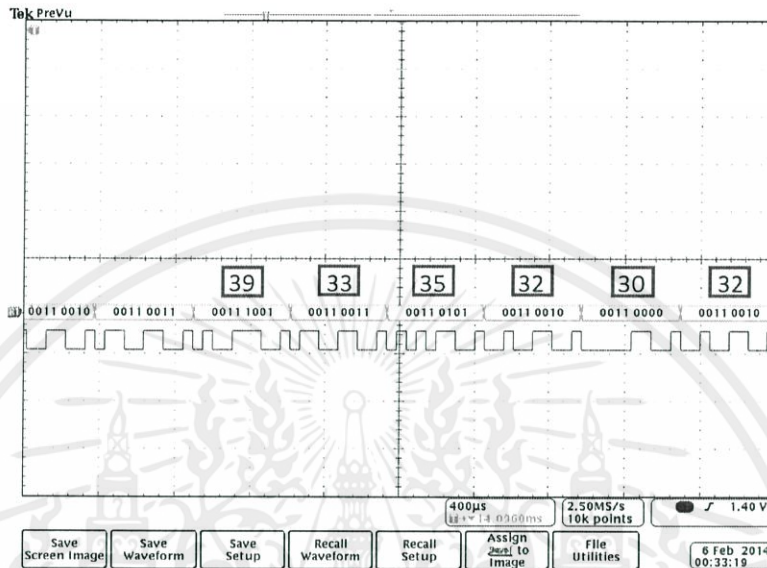
4.5



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.5 สัญญาณพัลส์ 36 33 37 33 32 33 ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลต่อไปเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 39 33 35 32 30 32 ดังรูปที่

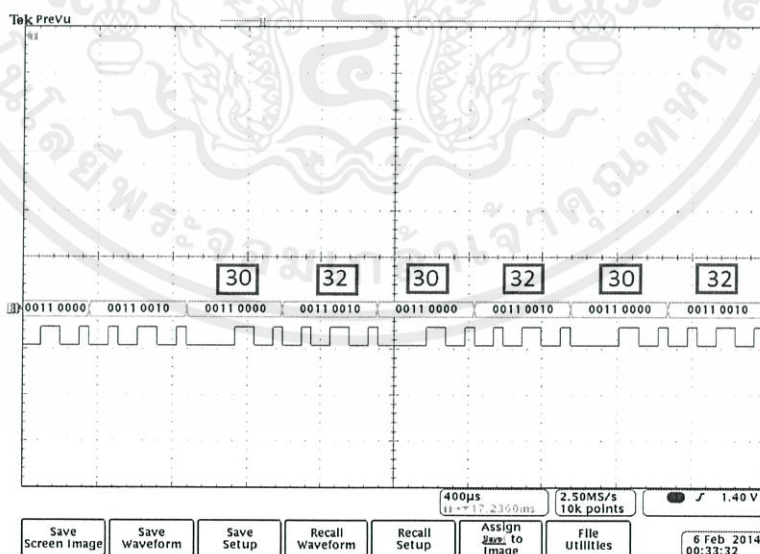
4.6



รูปที่ 4.6 สัญญาณพัลส์ 39 33 35 32 30 32

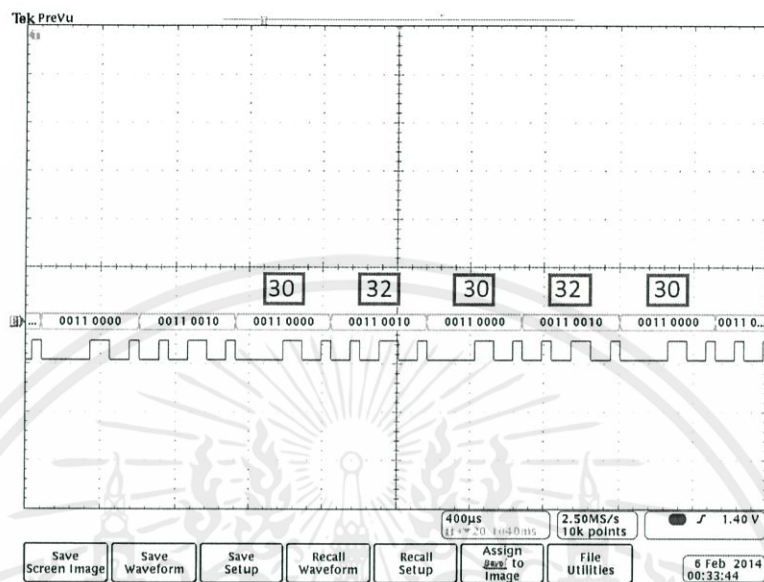
ข้อมูลต่อไปเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 30 32 30 32 30 32 ดังรูปที่

4.7



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.7 สัญญาณพัลส์ 30 32 30 32 30 32 ให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

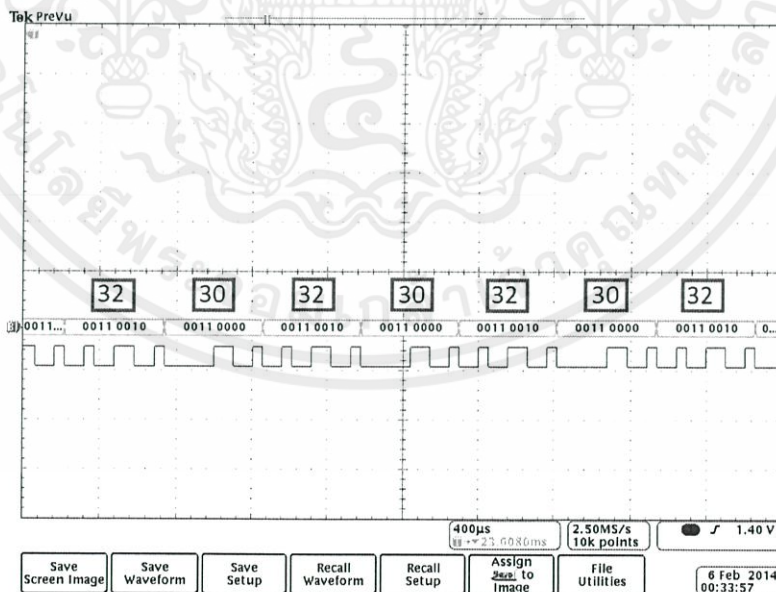
ข้อมูลต่อไปเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 30 32 30 32 30 ดังรูปที่ 4.8



รูปที่ 4.8 สัญญาณพัลส์ 30 32 30 32 30

ข้อมูลต่อไปเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 32 30 32 30 32 30 32 ดังรูป

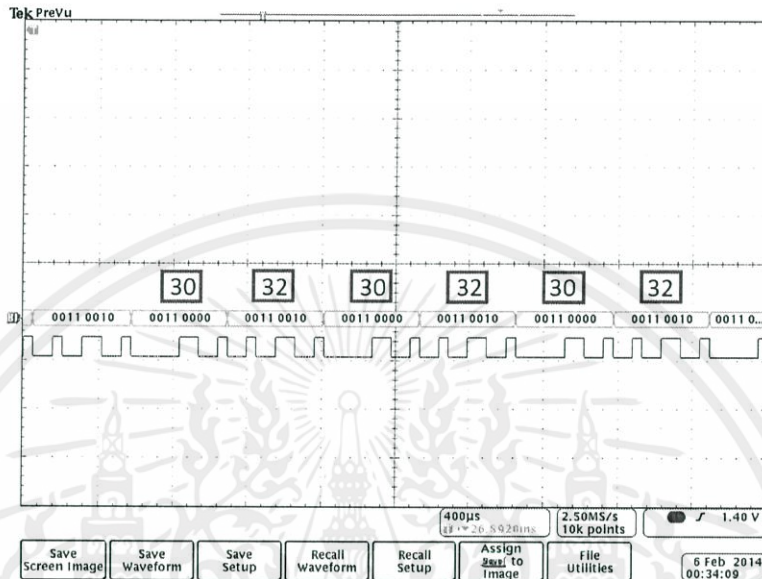
ที่ 4.9



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.9 สัญญาณพัลส์ 32 30 32 30 32 30 32 นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

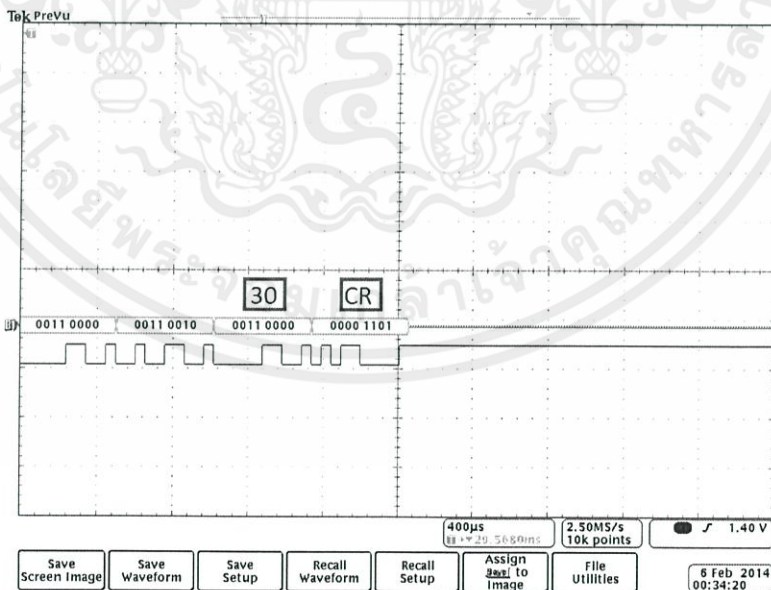
ข้อมูลต่อไปเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 30 32 30 32 30 32 ดังรูปที่

4.10



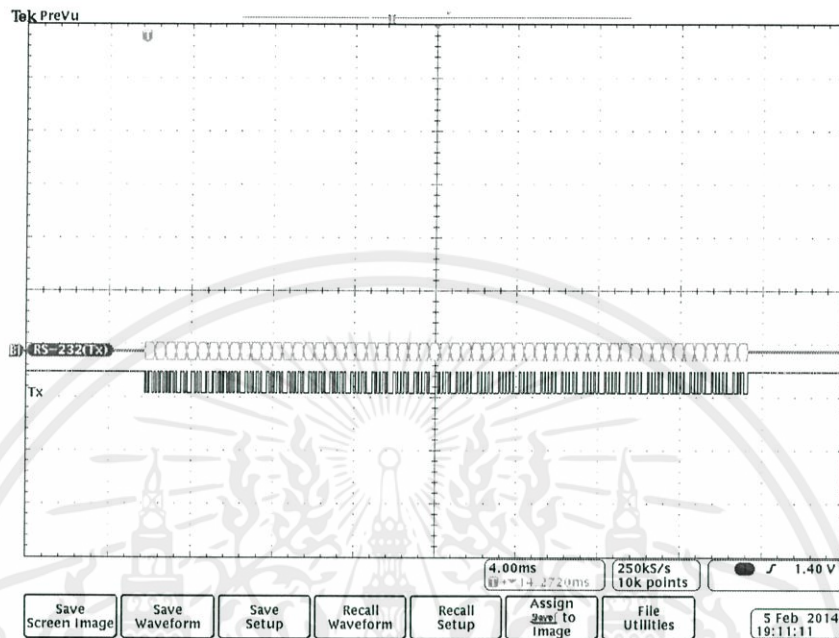
รูปที่ 4.10 สัญญาณพัลส์ 30 32 30 32 30 32

ข้อมูลสุดท้ายเมื่อแปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 30 CR ดังรูปที่ 4.11



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้รูปที่ 4.11 สัญญาณพัลส์ 30 CR กรุณาอย่าให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

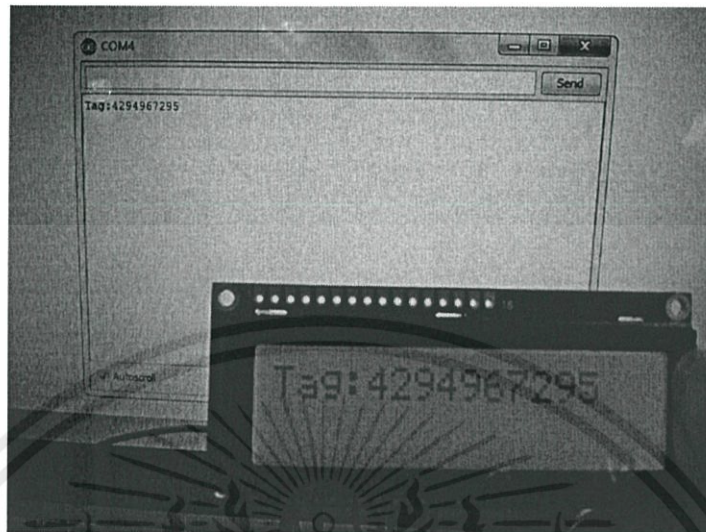
เมื่อดูสัญญาณรวมที่อ่านได้เมื่อมีแท็กในหน้าเดียวกันจะเห็นเป็นดังรูปที่ 4.12



รูปที่ 4.12 สัญญาณทั้งหมดที่อ่านได้เมื่อมีแท็ก

การอ่านข้อมูลออกมานั้น จะต้องเริ่มอ่านจากชุดที่ 10 ในรูปที่ 4.2 เนื่องจากใน 9 ชุดแรกจะเป็นค่าประจำที่มีอดูลจะอ่าน และเริ่มข้อมูลที่ชุด 10 ยกตัวอย่างเช่น อ่านได้อัตราบิตข้อมูล 0 1100 1100 1 จะต้องตัดบิตเริ่มต้นและบิตสุดท้ายซึ่งจะได้เป็น 1100 1100 และอ่านข้อมูลจากขวาไปซ้ายได้ 0011 0011 แปลงเป็นเลขฐานสิบหกจะได้ข้อมูลเป็น 33 ซึ่งสามารถแปลเป็นอักขระได้เลข 3 จากนั้นจะอ่านข้อมูลตัวต่อไปที่ได้ คือ 0 0010 1100 1 ตัดบิตเริ่มต้นและบิตสุดท้าย จะได้เป็น 0010 1100 และกลับบิตจากซ้ายไปขวาเป็น 0011 0100 ซึ่งแปลงเป็นเลขฐานสิบหกได้ค่า 34 แปลงเป็นอักขระได้ 4 จะถอดรหัสได้เป็น 0x34 ซึ่งได้เป็น 4 จากนั้นนำค่าไปตรวจสอบกับข้อมูลที่อ่านได้จากมอนิเตอร์ในคอมพิวเตอร์ จะได้ตรงกับเลขตัวแรกดังรูปที่ 4.13

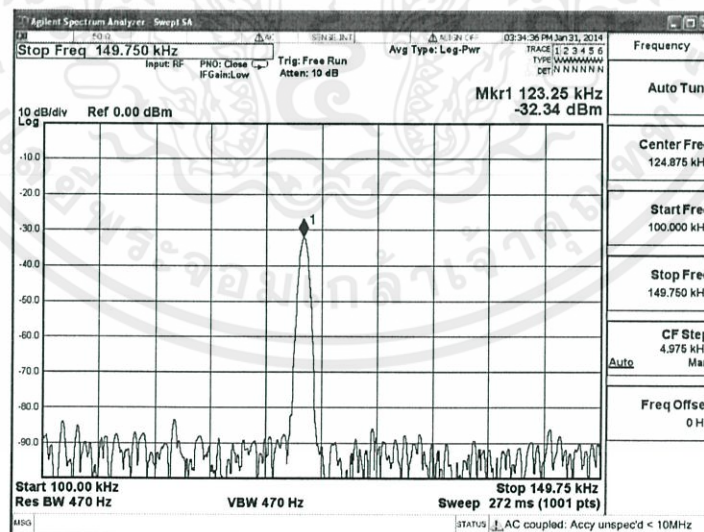
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 ค่าแท็กที่อ่านได้จากมอนิเตอร์

4.2 วัดความถี่ของเครื่องอ่านอาร์เอฟไอดี

เครื่องอ่านอาร์เอฟไอดีที่ใช้คือมอดูลรุ่น BM1250 ซึ่งจะมีความถี่ 125 กิโลเฮิร์ต เมื่อทำการวัดค่าด้วยเครื่องวัดสเปกตรัม จะได้ค่าดังรูปที่ 4.14 ซึ่งมีค่าประมาณ 123.25 กิโลเฮิร์ต ซึ่งใกล้เคียงกับข้อมูลของมอดูล



รูปที่ 4.14 ความถี่ของเครื่องอ่านอาร์เอฟไอดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการแสดงผลบนหน้าจอแสดงผลเทียบกับโปรแกรมสั่งอาหารในคอมพิวเตอร์

ในการสั่งอาหารเมื่อผู้ซื้อกดปุ่มกดเพื่อสั่งอาหาร จะมีการแสดงข้อความทางฝั่งผู้ซื้อ ผ่านหน้าจอแสดงผล ซึ่งเชื่อมโยงไปยังโปรแกรมสั่งอาหารในคอมพิวเตอร์ของผู้ขาย โดยจะแสดงภาพเปรียบเทียบกันระหว่างหน้าจอแสดงผลกับโปรแกรมสั่งอาหาร

4.3.1 เมื่อทำการล็อกอินเข้าโปรแกรมในคอมพิวเตอร์

หน้าจอแสดงผลจะแสดงดังรูปที่ 4.15



รูปที่ 4.15 หน้าต่างโปรแกรมการล็อกอิน

4.3.2 เมื่อทำการใส่รหัสล็อกอินเสร็จ

หน้าจอแสดงผลจะแสดงดังรูปที่ 4.16

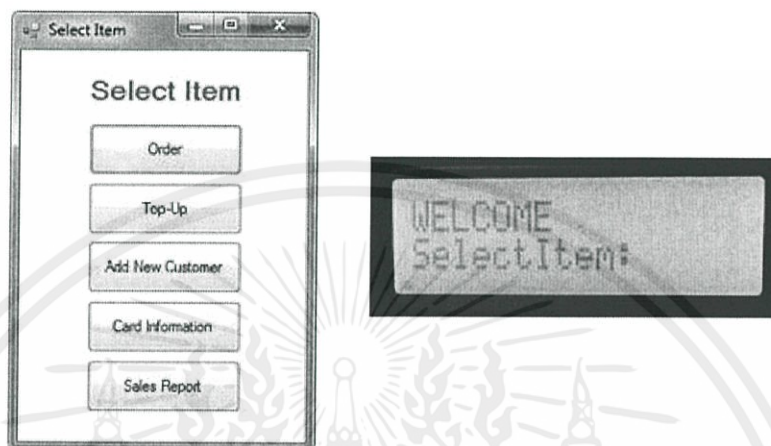


รูปที่ 4.16 หน้าต่างเมื่อการล็อกอินสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 หน้าต่างให้เลือกทำรายการต่างๆ

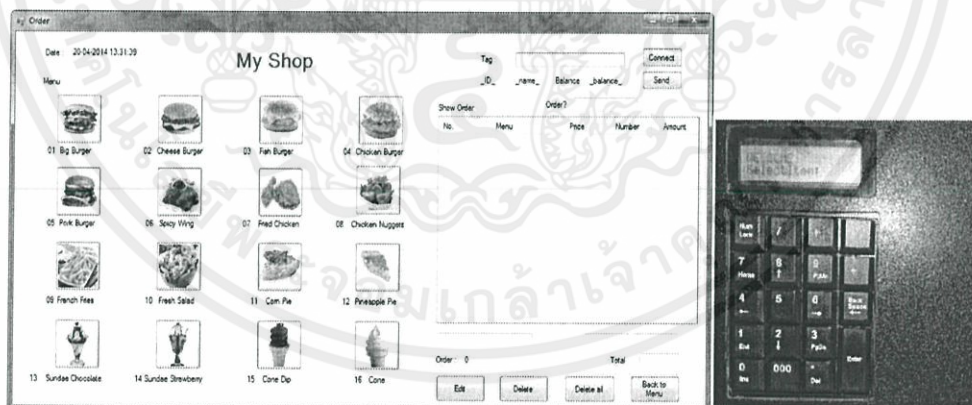
หน้าจอแสดงผลและหน้าต่างโปรแกรมจะแสดงดังรูปที่ 4.17



รูปที่ 4.17 การเลือกรายการ

4.3.3.1 เลือกสั่งอาหาร

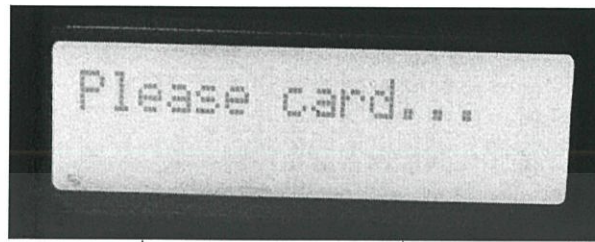
หน้าจอแสดงผลและหน้าต่างโปรแกรมจะแสดงดังรูปที่ 4.18



รูปที่ 4.18 การเลือกรายการเพื่อสั่งอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) เมื่อทาบบัตร หน้าจอแสดงผลจะแสดงดังรูปที่ 4.19



รูปที่ 4.19 หน้าจอแสดงผลเมื่อทาบบัตร

2) จะแสดงหมายเลขบัตร ชื่อผู้ใช้ และยอดเงินคงเหลือ หน้าจอแสดงผลจะแสดงดังรูปที่ 4.20



รูปที่ 4.20 หน้าจอแสดงผลเมื่อทาบบัตรเสร็จ

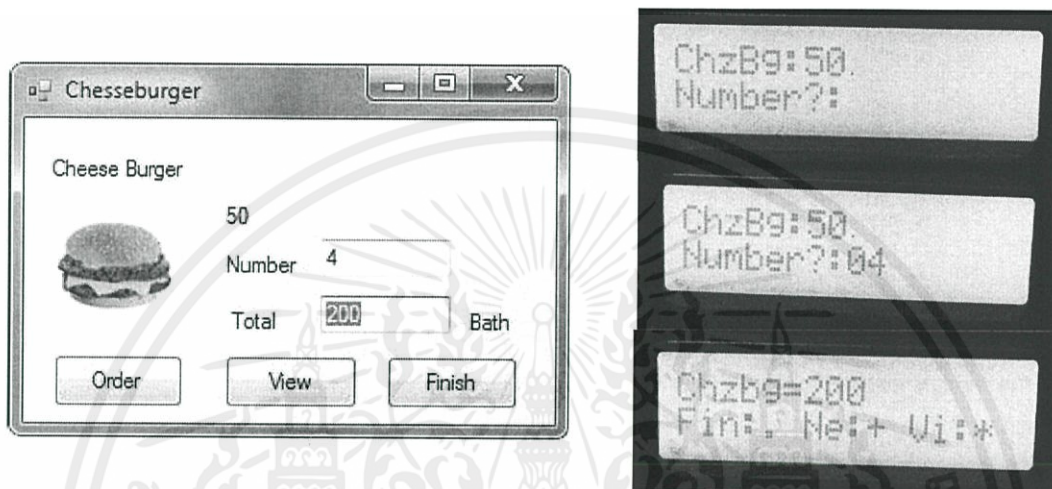
3) หน้าจอแสดงผลจะแสดงให้ป้อนหมายเลขรายการอาหารที่ต้องการสั่ง ดังรูปที่ 4.21



รูปที่ 4.21 หน้าจอแสดงผลเพื่อให้สั่งอาหาร

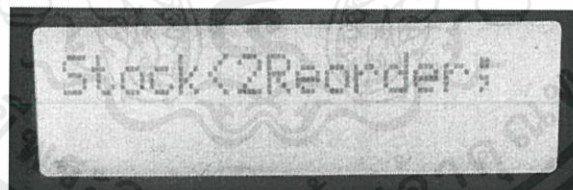
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) หน้าจอแสดงผลจะแสดงชื่อ และราคาของรายการอาหารที่สั่ง พร้อมทั้งให้ระบุจำนวนชิ้นในการสั่ง และฝั่งของพนักงานที่คอมพิวเตอร์จะแสดงหน้าต่างโปรแกรมดังรูป 4.22



รูปที่ 4.22 หน้าจอแสดงผลเพื่อให้ใส่จำนวน

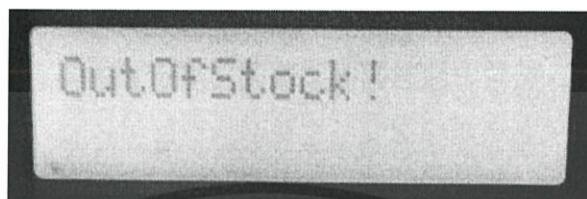
- กรณีที่สินค้าคงเหลือไม่พอกับจำนวนที่สั่ง โปรแกรมจะแสดงยอดสินค้าคงเหลือ และให้ระบุจำนวนชิ้นที่จะสั่งใหม่ ดังรูป 4.23



รูปที่ 4.23 หน้าจอแสดงผลเมื่อสินค้าคงเหลือไม่พอ

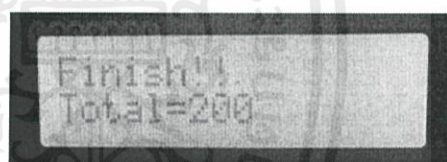
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- กรณีที่ไม่มีสินค้าคงเหลือเลย โปรแกรมจะบอกว่า
สินค้าหมด และให้ทำการเลือกรายการอาหารใหม่ ดังรูป 4.24



รูปที่ 4.24 หน้าจอแสดงผลเมื่อสินค้าหมด

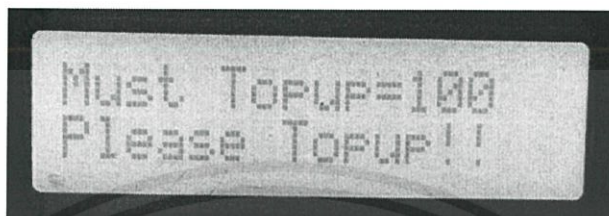
5) เมื่อเลือกสิ้นสุดการสั่งอาหาร โปรแกรมจะทำการคำนวณ
ราคาทั้งหมด และแสดงหน้าการชำระเงิน เพื่อหักยอดชำระกับยอดเงินที่มีอยู่ เป็นอันจบการทำงาน
ดังรูป 4.25



รูปที่ 4.25 หน้าจอแสดงผลเมื่อสิ้นสุดการสั่งอาหาร

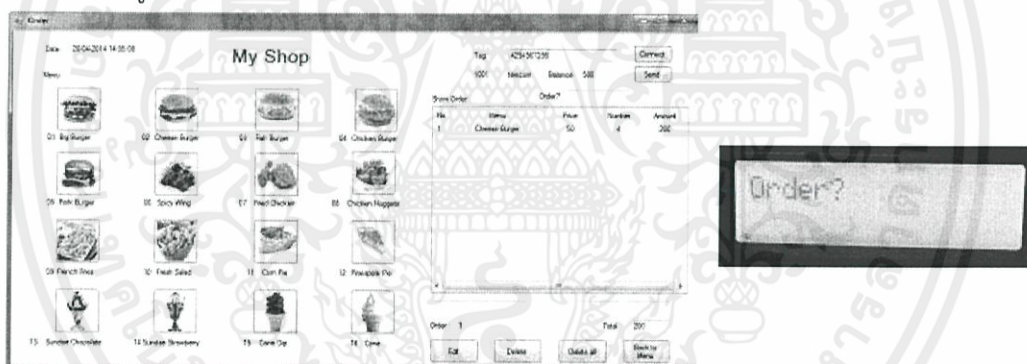
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- กรณีที่ยอดเงินคงเหลือไม่พอให้ชำระเงิน โปรแกรมจะ
คำนวณว่าต้องเติมเงินเพิ่มเท่าไร และไปที่หน้าเติมเงิน ดังรูป 4.26



รูปที่ 4.26 หน้าจอแสดงผลเมื่อเงินคงเหลือไม่พอ

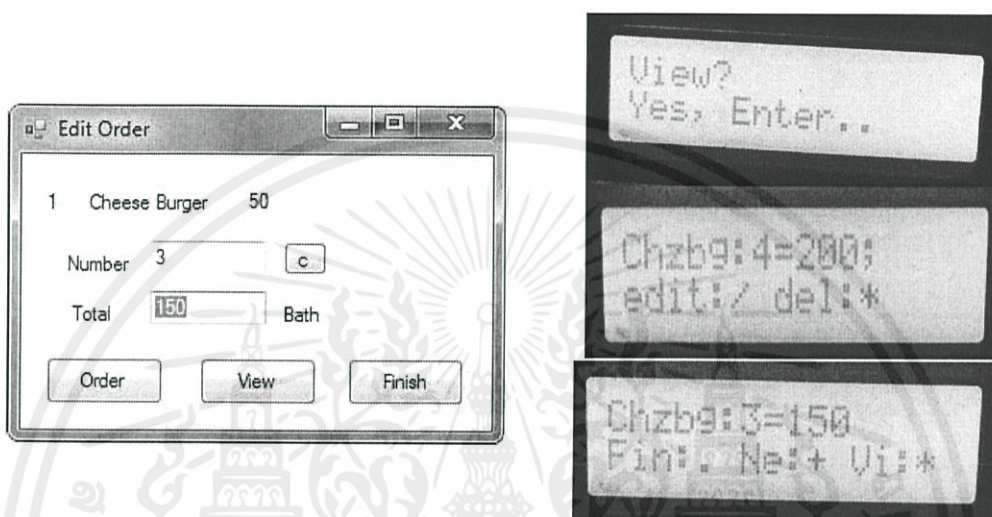
6) เมื่อเลือกสั่งอาหารต่อ โปรแกรมจะให้ป้อนหมายเลขรายการ
อาหารที่ต้องการสั่งอีกครั้ง เพื่อทำการสั่งเพิ่ม ส่วนรายการที่ถูกส่งไปแล้วจะถูกบันทึกไว้ในตาราง
ด้านข้าง ดังรูป 4.27



รูปที่ 4.27 หน้าจอแสดงผลเมื่อเลือกสั่งอาหารต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) เมื่อเลือกดูรายการที่สั่งไปแล้ว โปรแกรมจะสามารถเรียกดูรายการอาหารที่สั่งไว้ และสามารถทำการแก้ไขจำนวนชิ้นที่สั่ง หรือทำการลบรายการที่สั่งไว้แล้วได้ หน้าจอแสดงผลและหน้าต่างโปรแกรมจะแสดงดังรูปที่ 4.28

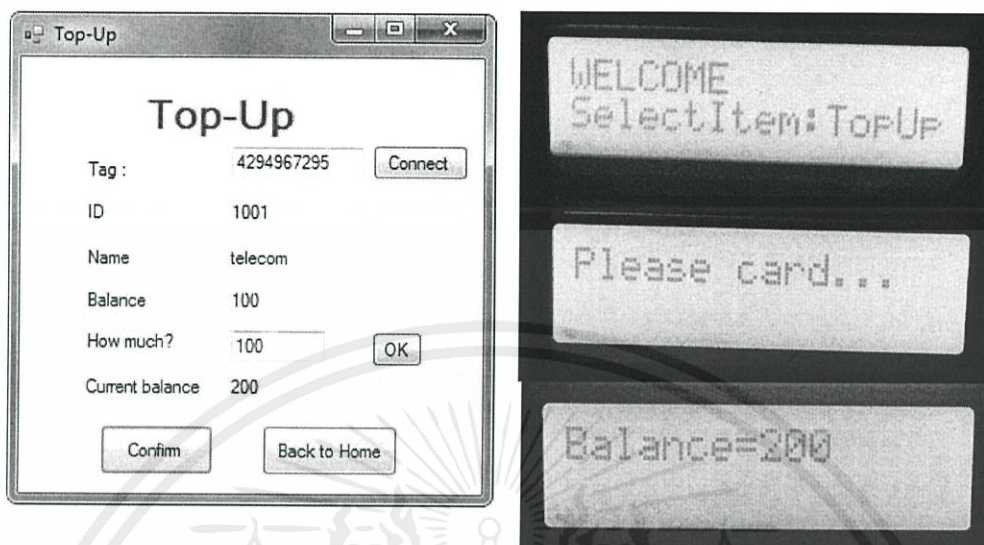


รูปที่ 4.28 หน้าจอแสดงผลเมื่อเลือกดูรายการอาหารที่สั่ง

4.3.3.2 เลือกเติมเงิน

เมื่อเลือกเติมเงิน ระบบจะให้ผู้ใช้ทาบัตร แล้วหน้าจอแสดงผลจะแสดงยอดเงินคงเหลือในบัตร จากนั้นให้ผู้ใช้บอกจำนวนเงินที่ต้องการเติมแก่พนักงาน เพื่อให้พนักงานเติมเงินลงบัตร หน้าจอแสดงผลและหน้าต่างโปรแกรมจะแสดงดังรูปที่ 4.29

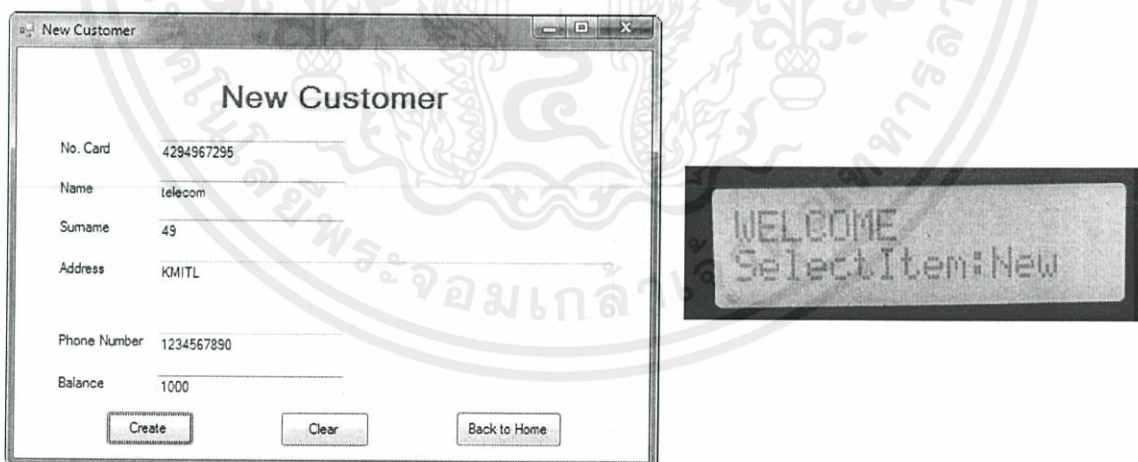
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.29 หน้าจอแสดงผลเมื่อเลือกเติมเงิน

4.3.3.3 เลือกสมัครสมาชิกใหม่

สำหรับลูกค้าใหม่จะต้องทำการสมัครสมาชิก เพื่อกรอกข้อมูลลงในระบบ หน้าจอแสดงผลและหน้าต่างโปรแกรมจะแสดงดังรูปที่ 4.30

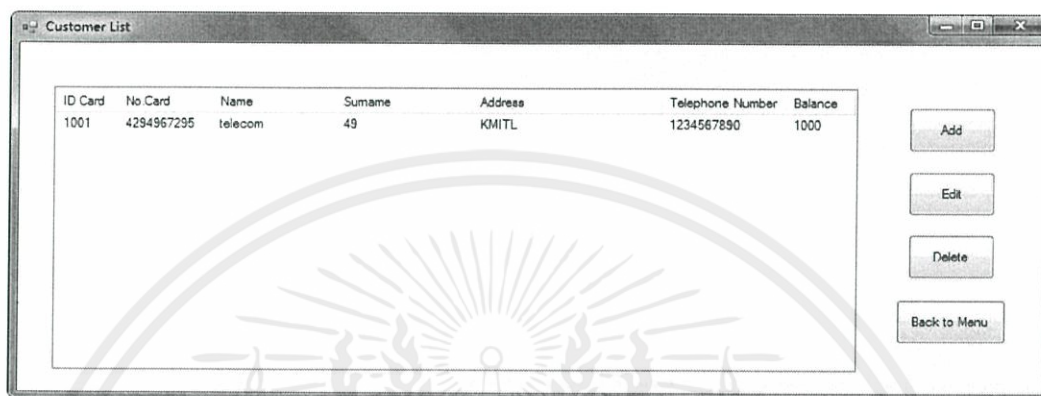


รูปที่ 4.30 หน้าจอแสดงผลเมื่อเลือกสมัครสมาชิกใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3.4 เลือกดูข้อมูลลูกค้า

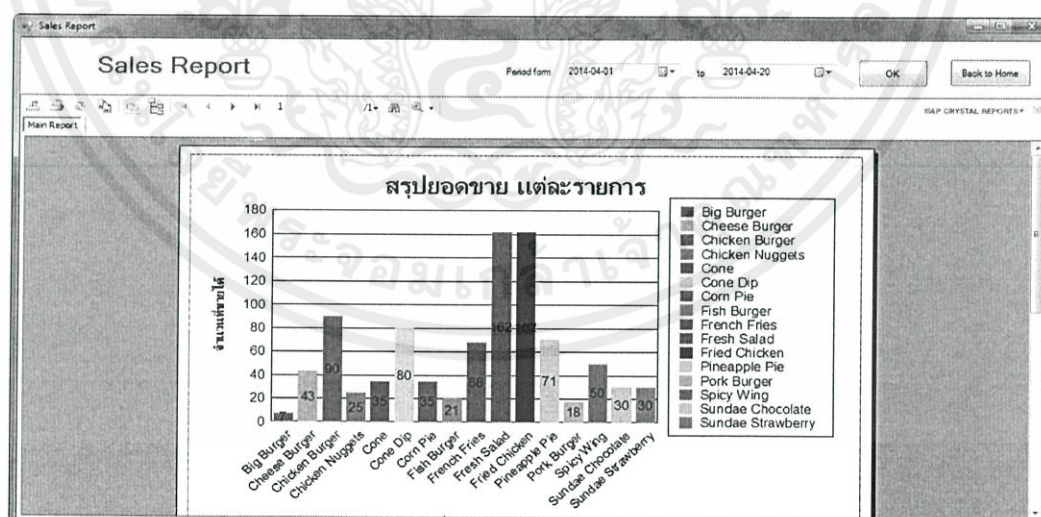
ในหน้านี้จะแสดงข้อมูลของลูกค้าที่สมัครเป็นสมาชิกแล้ว และยังสามารถทำการแก้ไข หรือลบข้อมูลออกได้ ดังรูป 4.31



รูปที่ 4.31 หน้าจอเมื่อเลือกดูข้อมูลลูกค้า

4.3.3.5 เลือกดูรายงานการขาย

สามารถดูสรุปยอดการขายได้ โดยเลือกช่วงวันที่ที่ต้องการ ระบบจะแสดงข้อมูลออกมาเป็นแผนภูมิแท่ง และตารางสรุปข้อมูล แสดงดังรูป 4.32 และรูป 4.33 ตามลำดับ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการรูปที่ 4.32 แผนภูมิแท่งสรุปลยอดขายญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sales Report

Period from 2014-04-01 to 2014-04-20

Total = 29,575.00

20/04/2014

name_menu	price	number	amount
French Fries	55	6	330
Big Burger	50	1	50
Fried Chicken	30	3	90
Fried Chicken	30	3	90
Com Pie	25	5	125
Cone	9	9	81
Cone Dip	15	3	45
Spicy Wing	20	1	20
Com Pie	25	9	225
Sundae Chocolate	29	2	58
Fried Chicken	30	110	3300
Fresh Salad	60	8	480
Cone Dip	15	8	120
Cheese Burger	60	3	180
Sundae Chocolate	29	4	116
Sundae Strawberry	29	3	87
Fresh Salad	60	3	180
Fried Chicken	30	9	270
Fresh Salad	60	6	360
Cheese Burger	60	3	180
Sundae Strawberry	29	3	87
Fried Chicken	30	9	270
Sundae Strawberry	29	3	87
Chicken Nuggets	50	5	250
Sundae Strawberry	29	3	87

รูปที่ 4.33 ตารางสรุปยอดขาย

4.4 อุปกรณ์ฝังลูกค้า

รูปเครื่องที่ใช้งานในการสั่งอาหารและชำระเงินทางฝังของลูกค้า แสดงได้ดังรูปที่ 4.36



รูปที่ 4.34 เครื่องสั่งอาหารและชำระเงินฝังทางลูกค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่หรือแจกจ่ายให้ผู้อื่นโดยไม่ได้รับอนุญาต หากมีข้อผิดพลาดประการใด ขออภัยเป็นอย่างสูง และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

จากการทำปฏิญานิพนธ์เรื่องระบบชำระเงินอัตโนมัติผ่านบัตรสมาร์ตการ์ด ซึ่งได้ทำการศึกษาลักษณะต่างๆ เพื่อนำมาประยุกต์ใช้ในการชำระเงินอัตโนมัติ โดยระบบจะถูกแบ่งออกเป็นส่วนของร้านค้า และส่วนของลูกค้า เมื่อมีลูกค้าต้องการมาซื้อสินค้าก็สามารถดำเนินการได้เอง โดยการป้อนข้อมูลที่ต้องการผ่านคีย์บอร์ดที่เชื่อมต่ออยู่กับไมโครคอนโทรลเลอร์ และดูผลผ่านจอแอลซีดี เมื่อสิ้นสุดการส่งสินค้าลูกค้าจะต้องชำระเงินผ่านบัตรอาร์เอฟไอดีที่ได้สมัครเป็นสมาชิกไว้กับทางร้าน เนื่องจากนี้ในระบบยังประกอบด้วย ระบบเติมเงิน และระบบสมัครสมาชิก โดยข้อมูลต่างๆ จะถูกบันทึกไว้ในฐานข้อมูล

โดยประกอบไปด้วยส่วนหลักๆ 2 ส่วน คือ ส่วนไมโครคอนโทรลเลอร์ กับ ส่วนของเครื่องอ่านอาร์เอฟไอดี ซึ่งทำหน้าที่อ่านข้อมูลจากบัตรอาร์เอฟไอดี ทำให้เห็นถึงปัญหาและอุปสรรคที่เกิดจากสาเหตุซึ่งสามารถแก้ไขปรับปรุงได้ดังแนวทางต่อไปนี้

5.1 สรุปผล

5.1.1 ออกแบบการทำให้ไมโครคอนโทรลเลอร์เชื่อมต่อกับอุปกรณ์ต่างๆ
ทำการเขียนไมโครคอนโทรลเลอร์ให้เชื่อมโยงกับอุปกรณ์ต่างๆ เช่น เครื่องอ่านอาร์เอฟไอดี หน้าจอแสดงผล และคอมพิวเตอร์

5.1.2 การเขียนโปรแกรมสั่งอาหาร
ทำการเขียนโปรแกรมสั่งอาหาร โดยใช้โปรแกรม Visual Studio 2012 เพื่อให้สามารถเชื่อมต่อข้อมูลระหว่างไมโครคอนโทรลเลอร์และฐานข้อมูลได้

5.1.3 การชำระเงินผ่านบัตร RFID
ทำการชำระเงินผ่านบัตร RFID โดยใช้เครื่องอ่านอาร์เอฟไอดีเป็นตัวอ่านข้อมูลจากบัตร แล้วนำไปเทียบในฐานข้อมูล โดยเครื่องอ่านอาร์เอฟไอดีนี้ จะถูกสั่งงานจากไมโครคอนโทรลเลอร์

5.2 ข้อเสนอแนะ

ในการทำปฏิญานิพนธ์นี้ต้องเขียนไมโครคอนโทรลเลอร์ เขียนโปรแกรมภาษาซี และทำการจัดการฐานข้อมูล ซึ่งจำเป็นต้องมีการศึกษาข้อมูลเพิ่มเติมเกี่ยวกับการใช้งานโปรแกรมต่างๆ เอกสารนี้เพื่อให้ผลที่ได้มีความสมบูรณ์มากยิ่งขึ้น เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] Art Chula. "สอนใช้ไมโครคอนโทรลเลอร์ Arduino." <http://osxarduino.blogspot.com>.
- [2] เอ เม้าท์เทน "เริ่มต้นใช้งาน Arduino." <http://www.arduinotronics.com/article>.
- [3] ปิยะ โควินท์ทวีวัฒน์ และคณะ, ระบบป้องกันขโมยด้วยคลื่นความถี่วิทยุ (RFID system), ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (เนคเทค), เพชรเกษมการพิมพ์, 2552.
- [4] Chuck Blog. "Introducing MySQL Connector/Arduino." <http://drcharlesbell.blogspot.com>.
- [5] Pc Man. "เริ่มต้นกับ Arduino." <http://www.logicthai.net>.
- [6] ธนกร ซอไม้ทอง. "อะไรคือVisual Basic." http://www.oocities.org/vb_training/whatis.html.
- [7] TC Admin. "สร้าง Crystal Report บน Visual Studio." <http://www.thaicreate.com/dotnet/crystal-report-vb-net-cs.html>.

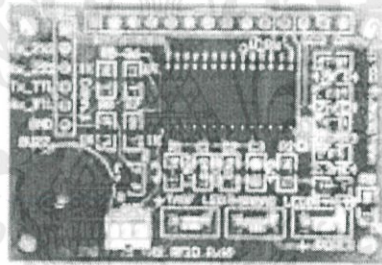
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



DATASHEET

RFID PRODUCTS

BM1250 RFID READER / WRITER MODULE



Binary Co., Ltd.

1351/10 Prachauchien Road, Bangkue District, Bangkok 10500, Thailand
Tel: +66-2-0517320-3, Fax: +66-2-589-7803
Email: sales@binary.co.th
URL: <http://www.binary.co.th>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

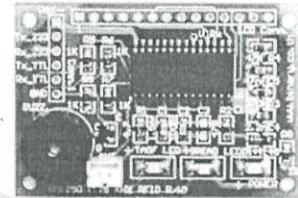
BM1250 RFID READER / WRITER MODULE



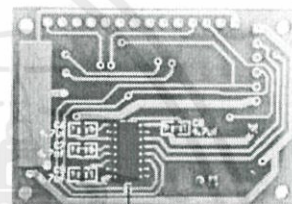
Overviews

Designed for embedding into products manufactured by third-parties, the BM1250 Module is a low cost 125Khz RFID card reader / writer in a printed circuit board form factor

- Ease - of - Usefulness, Rapid Product Development and being Cost Effective
- Automatically demodulate and decode RF signal and make the actual tag data ready. No need to know about complicated RFID subject
- Supports Atmel / Temic T55xx, EM4100 compatible and EM4102(Unique) tags
- Supports Manchester RF/64, Manchester RF/32 modulation
- Read Distance: 5 – 15 cm, typically 8 cm. (varies according to antenna & tags)
- Small Form factor, unit size : 5 cm. x 3.5 cm
- Instant port for 16 x 2 LCD display module, ready for display output reading data from EM4100 compatible and EM4102(Unique) tags
- On board color LED (Red/Green) for Pass indication
- On board Buzzer for Pass indication
- External RFID Coil Antenna (~860 uH)
- UART / TTLInterface up to 115200 bps (default 19,200)
- RS-232 Interface (optional) up to 115200 bps (default 19,200)



Front View



Rear View

RS-232 Interface (optional)

Description

BM1250 is designed to read and write the popular range of Atmel / Temic T55xx, EM4100 compatible and EM4102(Unique) tags cards and transponders available on the general market. EM4100 compatible transponders contain 64 bits of Read Only memory, while T55xx transponders carry 224 bits of Read / Write memory. The most used applications for the 125 KHz RFID systems is in general the person identification, specifically access control, time & attendance systems, and the part identification i.e. the immobilizer systems used in automotive applications.

BM1250 is one of the lowest cost RFID reader / writer (encoder) module with high performance and reliability for embedding to any third-parties' projects when compare to other competitors in the market. The module is designed for easy to use, with serial communication to the host devices either UART/TTL or RS-232 (optional through MAX 232 chip on the PCB's bottom side). By the special function, there is a instant output port which can drive the data output from the EM4100 tag reading directly to 16 x 2 LCD module without any additional components need.

BM1250 module needs a simple external antenna for driving magnetic field to passive RFID transponder. When they are within range of the reader unit they are able to draw sufficient power from the electromagnetic field to power their own internal electronics. Once powered they are able to modulate the incident magnetic field which is detected by the reader. In this way the transponders are able to transmit their data to the reader. The recommended inductance for the antenna is approximately 860 uH, customers can purchase from us or request coil information for in-house producing.

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand
Tel : +66-2-9547520-3, Fax : +66-2-569-7803
Email: sales@binary.co.th
URL: http://www.binary.co.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BM1250 RFID READER / WRITER MODULE



Electrical Specifications

Vdd Supply Voltage	4.75 V to 5.5 V
Digital Inputs to GND	-0.5V to 5.5V
Operating Current	75 mA
Operating Temperature Range	-40 C to 85 C
Storage Temperature Range	-55 C to 100 C

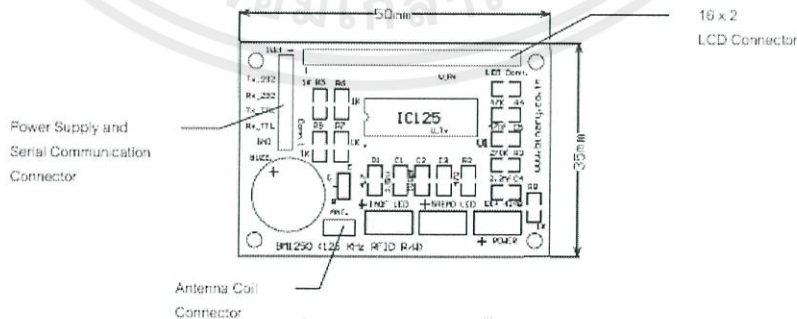
NOTE: (1) Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. Exposure to absolute maximum conditions for extended periods may affect unit reliability.

Connector Pin Description : Interface is via 6 line connection.

Pin	Label	Description
1	+5V	System power, +5V DC input
2	Tx_232	TXD (RS-232)
3	Rx_232	RXD (RS-232)
4	Tx_TTL	UART TXD (TTL levels only)
5	Rx_TTL	UART RXD (TTL levels only)
6	GND	System ground. Connect to power supply's ground (GND) terminal

Board Layout

Board Layout



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาติให้นำไปใช้หรือเผยแพร่การค้ำ

ไม่ว่ากรณีใดๆทั้ง
Binary Co., Ltd.
 1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand
 Tel : +66-2-9517520-3, Fax : +66-2-589-7803
 Email: sales@binary.co.th
 URL: http://www.binary.co.th

BM1250 RFID READER / WRITER MODULE



.Serial Output Format description

When a successful read of a transponder takes place the unit will transmit a string of information. This string of information will vary in length depending on the type of transponder being scanned, and the configuration of the transponder.

EM4100 transponder Output

The output format for a read of an EM4100 transponder is a simple string of 10 ASCII coded Hexidecimal characters followed by the ASCII code \$0D (carriage return) as a string end marker.

For example, when reading a EM4100 compatible card with the Version Number of \$06, and a card data of \$001259E3 the following string is transmitted:

06001259E3 <cr> where <cr> is serial ASCII code \$0D

The card information is transmitted once and a new string will not be sent until the transponder moves outside of the scanning range, and it, or another transponder enters scanning range again.

NOTE: Although EM4100 compatible transponders contain 64bits of information not all the bits are defined for user data. Some data bits are allocated as parity check bits and for other functions.

T55XX transponder Output

The output format for a read of a T55XX transponder is slightly more complicated than an EM4100 as it has more data and various options that alter its behavior. The memory structure of the T55XX rfid tags is as shown in Fig 1. T55XX memory structure.

Block memory description	
Block 0,	32 bit Configuration data
Block 1,	32 bit Used for Byte Track
Block 2,	32 bit R/W Data
Block 3,	32 bit R/W Data
Block 4,	32 bit R/W Data
Block 5,	32 bit R/W Data
Block 6,	32 bit R/W Data
Block 7,	32 bit R/W Data, or Password

Fig 1. T55XX memory structure

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand

Tel : +66-2-9517520-3, Fax : +66-2-589-7803

Email: sales@binary.co.th

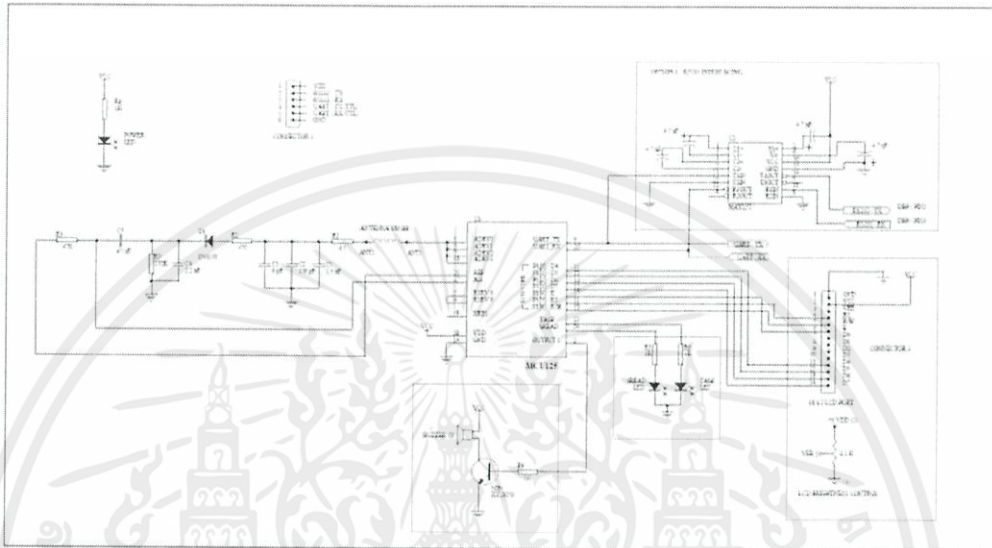
URL: http://www.binary.co.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BM1250 RFID READER / WRITER MODULE



EM1250 Schematic Diagram



Interfacing to 16 x 2 LCD module



Binary Co., Ltd.

135/1/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand
Tel : +66-2-9517520-3, Fax : +66-2-589-7803
Email: sales@binary.co.th
URL: <http://www.binary.co.th>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Communication Protocol

BM1250 can read tag automatically and output tag data without requiring any start read command. It eliminates use of external microcontroller. So far, auto read mode has been used widely in industry. When a tag enters into the RF field, its data is sent automatically through UART / RS232 to the Host / Terminal / PC or MCU, then the host processes the tag data and performs necessary action.

The communication between the host and the module can take place at 9600 bps, 19200 bps, 38400 bps, 57600 bps or 115200 bps N, 8, 1. The Module communicates at 19200 bps, N, 8, 1 as default. Once the baud rate is changed using the CMD_SET_BAUD command, successful communication will only occur with the new baud rate. The host first sends the command and the module executes the operation and replies with a response to the command. The host can analyze the reply to check if the operation was successful or if any error occurred during the operation.

Command Structure (From Host to BM1250)

Following is the UART frame for the commands sent by the host:

Header	Reserved	Length	Command	Data	CSUM
1 Byte	1 Byte	1 Byte	1 Byte	N Byte	1 Byte

1. Header : This is a single byte that indicates the beginning of a frame. This byte should be always 0xFF.
2. Reserved : This byte is reserved for future use and not implemented currently. It needs to be always 0x01.
3. Length : This byte is used to indicate the length of the payload data. This includes the Command and the Data bytes.
4. Command : This byte is used to instruct the module on what operation to perform.
5. Data : These are parameters necessary for the module to execute the command. For example, for a Read command, the data will be the modulation type to be read. For a Write command, this will be the block number and block data to be written to the tag.
6. CSUM : This is the checksum byte. This byte is used on the host as well as the module to check the validity of the packet and to trap any data corruption. This is calculated by adding all the bytes in the packet except the Header byte (0xFF).

Response Structure for Commands (From BM1250 to Host)

Following is the UART frame for the response packets sent by BM1250 module in response to the commands. For most of the commands that does not require parameter from BM1250, there is only one response indicating the command is executed successfully. For a few commands mentioned below, the response comes with the parameters requested and these commands are configuration commands which may not be used by a MCU system.

Binary Co., Ltd.

เอกสารนี้เป็นเอกสารของบริษัท Binary Co., Ltd. 1351/10, Prachachuen Road, Bangsue District, Bangkok 10800, Thailand. ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 Tel : +66-2-9517520-3, Fax : +66-2-589-7803
 Email: sales@binary.co.th
 URL: http://www.binary.co.th
 ไม่ว่ากรณีใดๆก็ตาม กรุณาแจ้งชื่อและที่อยู่ของเอกสารทุกครั้งที่มีการนำไปใช้

BM1250 RFID READER / WRITER MODULE



Header	Reserved	Length	Command	Response	CSUM
1 Byte	1 Byte	1 Byte	1 Byte	N Byte	1 Byte

1. Header : This is a single byte that indicates the beginning of a frame. This byte will be always 0xFF.
2. Reserved : This byte is reserved for future use and not implemented currently. It is 0x01.
3. Length : This byte is used to indicate the length of the payload data.
4. Command : This is the command for which the response is being sent back. The host can use this byte to verify that the received response is for the command it sent.
5. Response : This contains the data which may be a byte that indicates a successful or faulty operation, or the parameters requested.
6. CSUM: This is the checksum byte. This is the sum of all bytes except the Header byte.

When a command is executed successfully the returning response from the BM1250 device is : 0xFF 0x01 0x01 0x99 0x9B.
"0x99" means success operations and the 0x9B is the checksum.

Response Structure upon Tag Read (From BM1250 to Host)

Tag data is passed to the host as converted to ASCII characters. CR and LF will be added to the end of data. For example : If the tag contains data > 9977553311, then

Incoming ASCII Characters >9977553311

(As HEX >0x39 0x39 0x37 0x37 0x35 0x35 0x33 0x33 0x31 0x31 0x0D 0x0A)

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand
Tel : +66-2-9517520-3, Fax : +66-2-589-7803
Email: sales@binary.co.th
URL: <http://www.binary.co.th>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BM1250 RFID READER / WRITER MODULE



UART COMMANDS

BM1250 can be controlled over UART/Serial or there is no necessary for a control mechanism once the Auto Read Mode is setup. In Auto Mode, BM1250 runs automatically, continuously seek for tag, and does not require any command to start read operation. When a tag is read it can output through UART. Alternatively, BM1250 can be controlled by commands explained below.

CMD_READ

This command starts standard read operation in BM1250. The command requires two parameters. One parameter is the Mode (Read Type/Modulation Type) and the other is number of blocks to be read. Response indicating successful operation is expected immediately just after command is sent. If the UART output is enabled, BM1250 will send tag data as soon as the tag enters into the field.

Command CMD_READ	0x10
Mode / Type	1 Byte – Mode or Type of Read 0x01 – Byte Track Mode – Manchester RF/64 0x02 – EM4102 Mode not-decoded(raw) – Manchester RF/64 0x03 – EM4102 Mode – Parity decoded – Manchester RF/64 0x04 – Byte Track Mode – Manchester RF/32 Mode will be set to 0x03 automatically in the BM1250 device if it is different than the above values
Block Number	1 Byte – Number of total blocks to be read For EM4102 Mode block number will be adjusted to 2 automatically in the BM1250 device For Byte Track Mode, this should be between 1 and 7. Value greater than 7 blocks will be adjusted to 7 automatically in the BM1250 device

Example Command:
EM4102 Parity Decoded

```
TX > Read Tag   FF 01 03 10 02 02 18
RX > Success    FF 01 01 99 9B
```

```
0xFF   Header
0x01   Fixed
0x03   Length of payload (Data + Command)
0x10   Read Command
0x02   Mode Byte (EM4102 Parity Decoded) (Popular 125 KHz Read Type)
0x02   Block No
0x18   Checksum
```

เอกสารนี้เป็น **Binary Co., Ltd.** ให้บริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand
Tel: +66-2-9517520-3, Fax: +66-2-589-7803
ไม่ว่ากรณีใดๆ กรุณาติดต่อขอสงวนสิทธิ์ในเอกสาร และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
Email: sales@binary.co.th
URL: http://www.binary.co.th

CMD SET BAUD RATE

Once you have finished setting the new baud rate, please reset module to communicate with the new baud rate.

Command CMD_SET_BAUD_RATE	0x52
Baud Rate	1 Byte – Baud Rate setting Value 0x00 – 9600 0x01 – 19200 (Default) 0x02 – 38400 0x03 – 57600 0x04 – 115200

Example Command:
Set to baud rate 9600

TX > Set Baud FF 01 02 52 00 55
RX > Success FF 01 01 99 9B

CMD STOP_READ

This command stops reading operation. There is no parameter required for this command. Response indicating successful operation is expected immediately just after command is sent.

Command CMD_STOP_READ	0x12
--------------------------	------

Example Command:

TX > Stop Read FF 01 01 12 14
RX > Success FF 01 01 99 9B

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10600, Thailand
 Tel : +66-2-9517520-3, Fax : +66-2-589-7803
 Email: sales@binary.co.th
 URL: http://www.binary.co.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BM1250 RFID READER / WRITER MODULE



CMD_READ_WITH_PASS

This command starts reading operation for a password protected tag. The password will be sent approximately each second to the tag automatically. A tag that is not password protected will also be read. Response indicating successful operation is expected immediately just after command is sent.

Command	0x13
Mode / Type	<p>1 Byte – Mode or Type of Read</p> <p>0x01 – Byte Track Mode – Manchester RF/64 0x02 – EM4102 Mode not-decoded(raw) – Manchester RF/64 0x03 – EM4102 Mode – Parity decoded – Manchester RF/64 0x04 – Byte Track Mode – Manchester RF/32</p> <p>Mode will be set to 0x03 automatically in the BM1250 device if it is different than the above values</p>
Block Number	<p>1 Byte – Number of total blocks to be read</p> <p>For EM4102 Mode block number will be adjusted to 2 automatically in the BM1250 device</p> <p>For Byte Track Mode, this should be between 1 and 7. Value greater than 7 blocks will be adjusted to 7 automatically in the BM1250 device</p>
Password	<p>4 Byte – Password</p> <p>Note that password is the Block7 of the tag data</p>

Example Command:

```
TX > Read Tag w pass      FF 01 07 13 01 02 10 20 30 40 BE
RX > Success              FF 01 01 99 9B
```

```
0xFF  Header
0x01  Fixed
0x07  Length of payload (Data + Command)
0x13  Read with password command
0x01  Mode ( Byte Track mode – Manchester RF/64 is used)
0x02  Block No ( 2 block will be read = 8 byte )
0x10  1st password byte
0x20  2nd password byte
0x30  3rd password byte
0x40  4th password byte
0xBE  Checksum
```

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10900, Thailand
 Tel : +66-2-9517520-3, Fax : +66-2-589-7903
 Email: sales@binary.co.th
 URL: http://www.binary.co.th

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของ Binary Co., Ltd. ห้ามเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ หากมีข้อผิดพลาดหรือข้อสงสัย กรุณาแจ้งให้เราทราบเพื่อปรับปรุงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BM1250 RFID READER / WRITER MODULE



CMD_WRITE

This command writes/programs given data to supported re-writeable tags. Supported rewriteable tags have 4 byte in each block. 4 byte data can be written to allowed tag blocks. Response indicating successful operation is expected just after programming of tag is done. Unlike the other commands' responses, the expected 'success' response can take longer (~500ms) to get. The CMD_WRITE function supports Q5 (T5555), T5557 and other re-writeable T55xx tags.

Please note that, a success response does not guarantee that the given data is written to the tag. A success response guarantee that tag programming functions executed. The tag might not be programmed with the given data. User need to check manually by reading the tag. Write distance, tag shape, size and type are important parameters that affect writing.

Command CMD_WRITE	0x20
Block Number	<p>1 Byte – Block number to be programmed</p> <p>Please make sure you go over Tag Programming section before you program any block of the tag. Incorrect writing may lock the tag and make it useless.</p> <p>0x00 Configuration Block. 0x01 Block 1. Used for Byte Track 0x02 Block 2. Free Block 0x03 Block 3. Free Block 0x04 Block 4. Free Block 0x05 Block 5. Free Block 0x06 Block 6. Free Block 0x07 Block 7. Password Block or Free Block</p>
Data	4 Byte – Data to be programmed to the tag

Response:
 Success: 0xFF 0x01 0x01 0x99 0x9B
 Response may arrive with around 500ms delay

Example Command:

TX > FF 01 06 20 02 20 21 22 23 AF Program Block2 with data: 20 21 22 23
 RX > Success FF 01 01 99 9B

0xFF Header
 0x01 Fixed
 0x06 Length of payload (Data + Command)
 0x20 Write Tag command
 0x02 Block No (Block-2 will be written)
 0x20 1st byte to be programmed
 0x21 2nd byte to be programmed
 0x22 3rd byte to be programmed
 0x23 4th byte to be programmed
 0xAF Checksum

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand
 Tel : +66-2-9517520-3, Fax : +66-2-589-7903
 Email: sales@binary.co.th
 URL: http://www.binary.co.th

เอกสารนี้เป็นเอกสารของบริษัทฯ ใช้เฉพาะเพื่อการสื่อสารเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ หากมีข้อผิดพลาด กรุณาแจ้งให้เราทราบเพื่อปรับปรุงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BM1250 RFID READER / WRITER MODULE



CMD_WRITE_WITH_PASS

This command is another version of CMD_WRITE command. It writes/programs given data to supported re-writeable tags with a password. First password is sent then programming is performed. Unlike the other commands' responses, the expected 'success' response can take longer (~500ms) to get.

Please note that; a success response does not guarantee that the given data is written to the tag. A success response guarantee that tag programming functions executed. The tag might not be programmed with the given data. User need to check manually by reading the tag. Write distance, tag shape, size and type are important parameters that affect writing.

Command	0x23
CMD_WRITE_WITH_PASS	
Block Number	<p>1 Byte – Block number to be programmed Please make sure you go over Tag Programming section before you program any block of the tag. Incorrect writing may lock the tag and make it useless.</p> <p>0x00 Configuration Block, 0x01 Block 1. Used for Byte Track 0x02 Block 2. Free Block 0x03 Block 3. Free Block 0x04 Block 4. Free Block 0x05 Block 5. Free Block 0x06 Block 6. Free Block 0x07 Block 7. Password Block or Free Block</p>
Data	4 Byte – Data to be programmed to the tag
Password	4 Byte – Password Note that password is the Block7 of the tag data.

Response:
Success: 0xFF 0x01 0x01 0x99 0x9B
Response may arrive with around 500ms delay

Example Command : //Program Block2 with data: 20 21 22 23 with password:10 20 30 40

TX > FF 01 0A 23 02 20 21 22 23 10 20 30 40 56
RX > Success FF 01 01 99 9B

0xFF Header
0x01 Fixed
0x0A Length of payload (Data + Command)
0x23 Write with Password command
0x02 Block No (Block-2 will be written)
0x20 1st byte to be programmed
0x21 2nd byte to be programmed
0x22 3rd byte to be programmed
0x23 4th byte to be programmed
0x10 1st password byte
0x20 2nd password byte
0x30 3rd password byte
0x40 4th password byte
0x56 Checksum

เอกสารนี้เป็น **Binary Co., Ltd.** บริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆก็ตาม
1351/10 Prachachuen Road., Bangsue District, Bangkok 10800, Thailand
Tel: +66-2-9517520-3, Fax: +66-2-589-7803
Email: sales@binary.co.th
URL: http://www.binary.co.th

BM1250 RFID READER / WRITER MODULE



CMD_SET_AUTO_MODE

This command enables or disables Auto Read Mode and determines the necessary parameters to be used in Auto Read Mode. The parameters define the Mode (Read/Modulation type) to be used, number of blocks to be read and if it is necessary or not to send password. Auto Read Mode brings advantage to user, so that external control of BM1250 may not be required. Once Auto Read mode command is executed, the parameters are stored in non-volatile memory of the BM1250 device. Therefore it can reliably be used in the field without need of external microcontroller to initiate read operations.

Command CMD_SET_AUTO_MODE	0x57
Auto Mode Enable/Disable	1 Byte – Auto Mode Enable / Disable 0x00 Disables Auto Read Mode 0x01 Enables Auto Read Mode
Mode	1 Byte – Mode or Type of Read 0x01 – Byte Track Mode – Manchester RF/64 0x02 – EM4102 Mode not-decoded(raw) – Manchester RF/64 0x03 – EM4102 Mode – Parity decoded – Manchester RF/64 0x04 – Byte Track Mode – Manchester RF/32 Mode will be set to 0x03 automatically in the BM1250 device if it is different than the above values
Block Number	1 Byte – Number of total blocks to be read For EM4102 Mode block number will be adjusted to 2 automatically in the BM1250 device For Byte Track Mode, this should be between 1 and 7
Password Enable/Disable	1 Byte – Password Enable / Disable 0x00 Password is not send 0x01 Password is sent approximately each second
Password	4 Byte – Password If the password is not going to be used, random numbers can be given Note that Password is the data stored in Block7 of the tag

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand

Tel : +66-2-9517520-3, Fax : +66-2-582-7803

Email: sales@binary.co.th

URL: http://www.binary.co.th

เอกสารนี้เป็นเอกสารของบริษัท Binary Co., Ltd. การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น ขอสงวนสิทธิ์ในเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BM1250 RFID READER / WRITER MODULE



Response:

Success: 0xFF 0x01 0x01 0x99 0x2B

Example Command:

Enable Auto Read Mode – EM4102 Decoded Mode – No password

TX > Set Auto Mode FF 01 09 87 01 03 02 00 10 20 30 40 37

RX > Success FF 01 01 99 2B

0xFF Header
0x01 Fixed
0x09 Length of payload (Data + Command)
0x87 Set Auto Mode Command
0x01 Auto Mode is enabled
0x03 Mode (EM4102 parity decoded)
0x02 Total block no to be read
0x00 Disable Password
0x10 1st password byte even it is not used
0x20 2nd password byte even it is not used
0x30 3rd password byte even it is not used
0x40 4th password byte even it is not used
0x37 Checksum

CMD RESET

This command resets BM1250 device. There is no parameter required for this command. Response indicating successful operation is expected immediately just after command is sent.

Please note that, if a read is stopped by CMD_STOP_READ command, and if Auto Read Mode is enabled, after a reset, automated read operation will start again.

Command	0x51
CMD_STOP_READ	

Example Command:

TX > Reset FF 01 01 51 53

RX > Success FF 01 01 99 2B

0xFF Header
0x01 Fixed
0x01 Length of payload (Data + Command) (Here Data is zero byte in length)
0x51 Reset Command
0x53 Checksum

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand

Tel : +66-2-9517520-3, Fax : +66-2-629-7503

Email: sales@binary.co.th

URL: <http://www.binary.co.th>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CMD_SLEEP

This command sends sleep command to the device to save power. Once module enters into sleep mode, only external hardware reset or Power-On-Reset will wake up the BM1250 device

Command	0x00
CMD_STOP_READ	

Response:
Success: 0xFF 0x01 0x01 0x99 0x6B

Example Command:

```
TX > Sleep      FF 01 01 60 62
RX > Success    FF 01 01 99 6B
```

```
0xFF  Header
0x01  Fixed
0x01  Length of payload (Data + Command) (Here Data is zero byte in length)
0x60  Sleep Command
0x62  Checksum
```

THEORY OF READ OPERATION

There are two options for the read operation according to the modulation and the method used in the tag.

1. Manchester Modulation

BM1250 supports Manchester RF/64 and Manchester RF/32 modulations. The Manchester RF/64 is the most used modulation type in 125 KHz systems and is recommended to be used for new designs. Manchester RF/32 is used in some tags (e.g TK5552) but the read performance is lower than the Manchester RF/64 and not recommended for new designs. For Manchester RF/64, signal representing 1 or 0 is in 612 us width for 125 KHz. For Manchester RF/32, signal representing 1 or 0 is in 256 us width for 125 KHz. EM4102 tags use Manchester RF/64 modulation and EM method and the configuration of these read-only tags is fixed and can't be altered.

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand
Tel : +66-2-9517520-3, Fax : +66-2-589-7803
Email: sales@binary.co.th
URL: http://www.binary.co.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Byte Track Method

Whether the modulation is Manchester RF/32 or Manchester RF/64, when the tag is energized in the magnetic field across the antenna, tag will be sending data consist of 1s and 0s continuously with no break or no sign for the header. To detect the start sequence of the incoming data there are two methods. One is the "byte track" method and the other is "EM" method. If "Byte Track" is used, BM1250 device tracks 4 byte; 0x52 0x58 0x8B 0x45 as default. These settings can be changed. BM1250 device can trace for 1 byte, 2 byte, 3 byte or 4 byte. The most reliable and recommended option is to trace for 4 byte. Note that the traced bytes should not repeat elsewhere in the incoming data stream, otherwise the header/block1 synchronization can be wrong. Please note that, if the tag does not include bytes to be traced, then that tag can not be read. T555x re-writable tags needs to use Byte Track method and so the first block needs to be written/programmed with the bytes to be traced.

Incoming and continuous data stream from the tag:

```
11110000011100100...0101 0010 0101 1000 1000 1011 0100 0101...000111010010011111001
      [ 0x52 ][ 0x58 ][ 0x8B ][ 0x45 ]
```

T555x Tags send data as follows:

(Example is given for 3 blocks is transmitted by the tag. Each block has 4 byte data)

```
[---][Block 1][Block 2][Block 3][Block 1][Block 2][Block 3][Block 1][---]
```

Block1 should be programmed with 0x52 0x58 0x8B 0x45 so that when BM125 catch these bytes, it will be synchronized and assume 0x52 0x58 0x8B 0x45 as the first block.

Notes

- Block 0 For T5555 (Q5), configuration block must contain 0x00, 0x01, 0xF0, 0x0E
For T5567, T5567, configuration block must contain 0x20, 0x14, 0x80, 0xE0
corresponds to Manchester RF/64 - transmit of 7 block (Block1, 7).
Block0 should not be programmed with arbitrary values otherwise the tag may not be accessed again.
- Block 1 This block can be used for "Byte Track" method. What is programmed in this block should be traced by BM1250.
BM1250 device tracks 4 byte; 0x52 0x58 0x8B 0x45 as default
- Block 2-6 General purpose data block. Writing to this block is safe. No care required.
- Block 7 This block can be used as general purpose data block or can be used as password. If password is enabled, access to the tag requires the password that matches block7 content. Once password is enabled and if the block7 data is unknown then it is impossible to access the tag again unless with the correct 4 byte password.

Binary Co., Ltd.

135/110 Prachachuen Road, Bangsue District, Bangkok 10900, Thailand
Tel : +66-2-2517520-3, Fax : +66-2-589-7803
Email: sales@binary.co.th
URL: http://www.binary.co.th

เอกสารนี้เป็นเอกสารของบริษัทฯ อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. EM4102 Mode

Whether the modulation is Manchester RF/32 or Manchester RF/64, when the tag is energized in the magnetic field across the antenna, tag will be sending data consist of 1s and 0s continuously with no break or no sign for the header. To detect the start sequence of the incoming data there is another method called "EM4102" method. This method is used in EM4102 tags and generally uses Manchester RF/64 modulation. In this mode, instead of tracing bytes explained in section 2.1, nine 1s (1111 1111 1) are traced to detect the beginning of the data sequence. Moreover the data written in the tag is created with parity bits so that even a tag has nine 1s, the read operation will not be successful if the parities do not match.

EM4102 tags consist of 2 blocks each have 4 byte. These 8 bytes include nine 1s, actual data that is the 5 byte identification code and the row and column parity bits.

1111100100...1111 1111 1...000111010010011111001
 ; Header] [Parity coded data]

EM4102 tag sends data as follows:

[---][Block 1][Block 2][Block 1][Block 2][---]

Please note that, the actual identification code is 5 Bytes in EM4102 tag. With the nine ones and parity bits the raw data is two blocks each 4 byte. Parity check will be done by BM1250 automatically and if parity bits do not match then there won't be any output and BM1250 keep trying to read. To get the actual 5 byte data, use "EM4102 decoded" option. To get the raw data including nine ones and parity bits use "EM4102 raw data" option.

Disclaimers

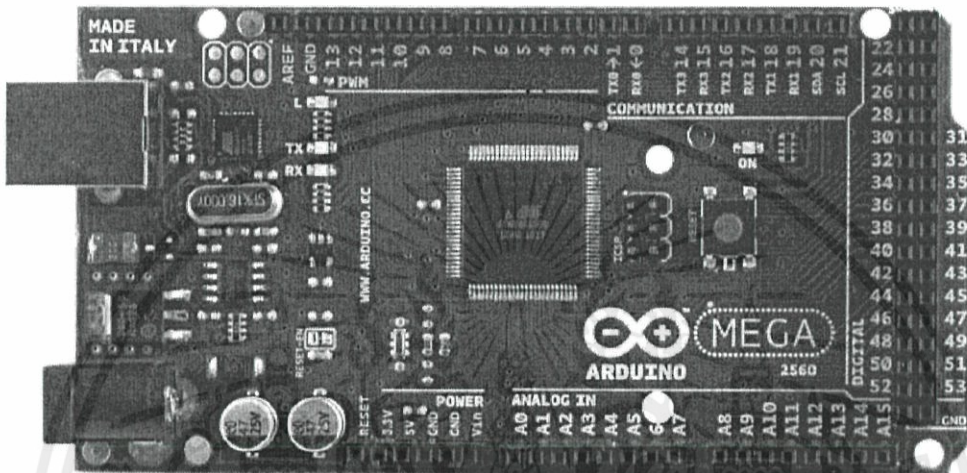
Binary makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Binary shall not be liable to recipients or any third party for any damages, including but not limited to personal injury, property damage, loss of profits, loss of use, interrupt of business or indirect, special incidental or consequential damages, of any kind, in connection with or arising out of the furnishing, performance or use of products supplied, or the technical data herein.

Binary Co., Ltd.

1351/10 Prachachuen Road, Bangsue District, Bangkok 10800, Thailand
 Tel : +66-2-9517520-3, Fax : +66-2-589-7503
 Email: sales@binary.co.th
 URL: <http://www.binary.co.th>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Arduino MEGA 2560



Product Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Index

Technical Specifications

Page 2

How to use Arduino
Programming Environment, Basic Tutorials

Page 6

Terms & Conditions

Page 7

Environmental Policies

Page 7

เอกสารนี้ half sqm of green via Impatto Zero® งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



radiospares

RADIONICS



Technical Specification

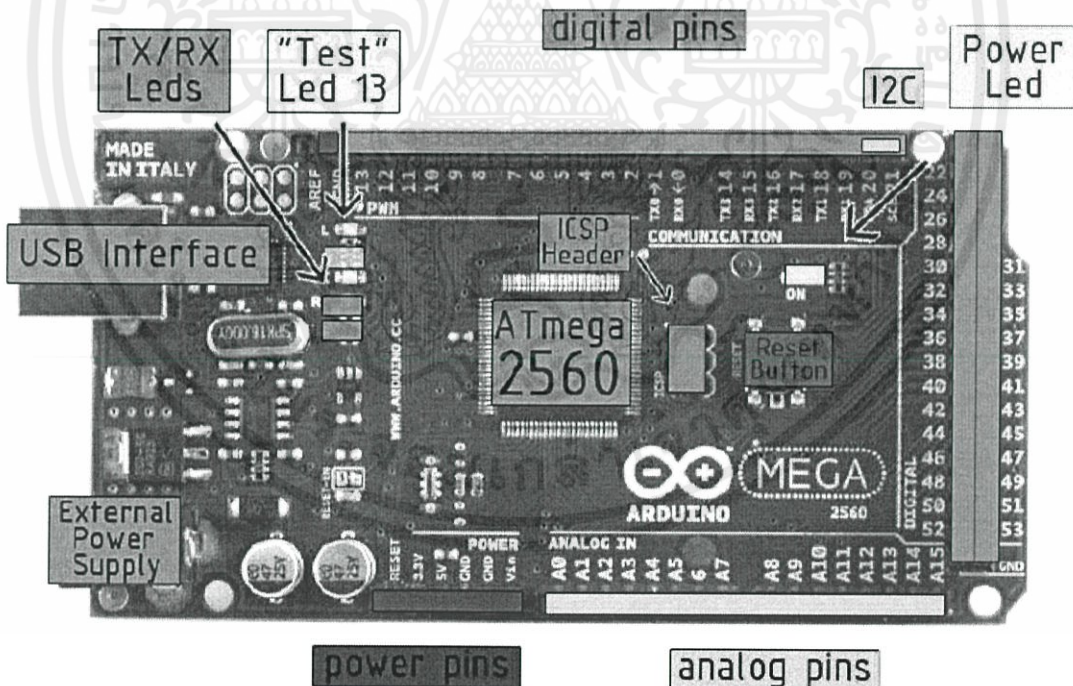


EAGLE files: [arduino-mega2560-reference-design.zip](#) Schematic: [arduino-mega2560-schematic.pdf](#)

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

the board



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆก็ตาม อีกหนึ่งสิ่งที่คุณต้องคำนึงถึงคือ ต้นฉบับเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกฉบับ นำไปใช้



Radiospares RADIONICS



Power

The Arduino Mega2560 can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V. The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND. Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX). Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .
- External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2). These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the `attachInterrupt()` function for details.
- PWM: 0 to 13. Provide 8-bit PWM output with the `analogWrite()` function.
- SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Duemilanove and Diecimila.
LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
I²C: 20 (SDA) and 21 (SCL). Support I²C (TWI) communication using the Wire library (documentation on the Wiring website). Note that these pins are not in the same location as the I²C pins on the Duemilanove.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and `analogReference()` function.

There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with `analogReference()`.
- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.



radiospares

RADIONICS



Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows for serial communication on any of the Mega's digital pins.

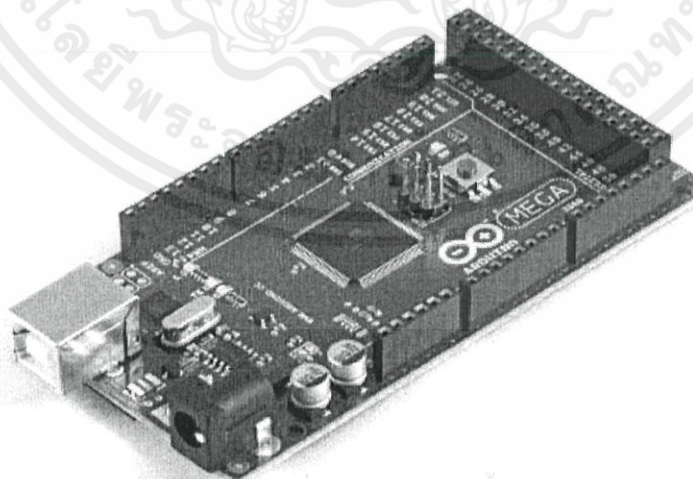
The ATmega2560 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation on the Wiring website for details. To use the SPI communication, please see the ATmega2560 datasheet.

Programming

The Arduino Mega2560 can be programmed with the Arduino software (download). For details, see the reference and tutorials.

The ATmega2560 on the Arduino Mega comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



radiospares

RADIONICS



Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Mega contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

The Arduino Mega has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics and Shield Compatibility

The maximum length and width of the Mega PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega is designed to be compatible with most shields designed for the Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega and Duemilanove / Diecimila. Please note that I²C is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



radiospares

RADIONICS



How to use Arduino



communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP). Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing). Arduino projects can be stand-alone or they can

Arduino is a cross-platform program. You'll have to follow different instructions for your personal OS. Check on the Arduino site for the latest instructions. <http://arduino.cc/en/Guide/HomePage>

Linux Install

Windows Install

Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

File>Sketchbook>
Arduino-0017>Examples>
Digital>Blink

Once you have your skecth you'll see something very close to the screenshot on the right.

In Tools>Board select MEGA

Now you have to go to
Tools>SerialPort
and select the right serial port, the one arduino is attached to.

```
int ledPin = 13; // LED connected to digital pin 13
// The setup() method runs once, when the sketch starts
void setup() {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
}
// the loop() method runs over and over again,
// as long as the Arduino has power
void loop() {
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(1000); // wait for a second
  digitalWrite(ledPin, LOW); // set the LED off
  delay(1000); // wait for a second
}
```



Done compiling.



Upload



TX RX Flashing



Blinking Led!



RADIOSPARES

RADIONICS



Terms & Conditions



1. Warranties

1.1 The producer warrants that its products will conform to the Specifications. This warranty lasts for one (1) years from the date of the sale. The producer shall not be liable for any defects that are caused by neglect, misuse or mistreatment by the Customer, including improper installation or testing, or for any products that have been altered or modified in any way by a Customer. Moreover, The producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.

1.2 If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are determined by the producer not to conform to such warranty. If the producer elects to replace such products, the producer shall have a reasonable time to replacements. Replaced products shall be warranted for a new full warranty period.

1.3 EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE

1.4 Customer agrees that prior to using any systems that include the producer products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical, applications or design advice, quality characterization, reliability data or other services. Customer acknowledges and agrees that providing these services shall not expand or otherwise alter the producer's warranties, as set forth above, and no additional obligations or liabilities shall arise from the producer providing such services.

1.5 The Arduino™ products are not authorized for use in safety-critical applications where a failure of the product would reasonably be expected to cause severe personal injury or death. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino™ products are neither designed nor intended for use in military or aerospace applications or environments and for automotive applications or environment. Customer acknowledges and agrees that any such use of Arduino™ products which is solely at the Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

1.6 Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino™ products in Customer's applications, notwithstanding any applications-related information or support that may be provided by the producer.

2. Indemnification

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third-party losses, damages, liabilities and expenses it incurs to the extent directly caused by: (i) an actual breach by a Customer of the representation and warranties made under this terms and conditions or (ii) the gross negligence or willful misconduct by the Customer.

3. Consequential Damages Waiver

In no event the producer shall be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.

4. Changes to specifications

The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." The producer reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.



Environmental Policies



The producer of Arduino has joined the Impatto Zero® policy of LifeGate.it. For each Arduino board produced is created / looked after half squared Km of Costa Rica's forest's.

เอกสารนี้เป็นเอกสารของบริษัทฯ สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ หากมีข้อสงสัย กรุณาติดต่อฝ่ายขายและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



radiospares

RADIONICS

