ระบบสั่งอาหารไร้สาย
WIRELESS FOOD ORDERING SYSTEM

กิตติภพ ไชยพันธ์
ธนภัทร พลตื้อ
ธนศาสตร์ เขียวมณี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมระบบควบคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

สมพิมพลัลป์ 0112

ระบบสั่งอาหารไร้สาย
WIRELESS FOOD ORDERING SYSTEM

กิตติภพ ไชยพันธ์
ธนภัทร พลตื้อ
ธนศาสตร์ เขียวมณี

ปพ.
ก674ร
2558
144284
09 พย 2559
b.12816516
i.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมระบบควบคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

WIRELESS FOOD ORDERING SYSTEM

KITTIPHOP CHAIYAPHAN

THANAPAT POLTUE

TANASART KIEWMANEE

THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN CONTROL ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2015

ปริญญานิพนธ์ปีการศึกษา 2558

ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบสั่งอาหารไร้สาย

WIRELESS FOOD ORDERING SYSTEM

ผู้จัดทำ นายกิตติภพ ไชยพันธ์ 55010093

นายธนภัทร พลตื้อ 55010501

นายธนศาสตร์ เขียวมณี 55010519

.....อาจารย์ที่ปรึกษา

(รองศาสตราจารย์ ดร.วรพงศ์ ตั้งศรีรัตน์)

ระบบสั่งอาหารไร้สาย

โดย
นายกิตติภพ ไชยพันธ์ 55010093
นายธนภัทร พลตื้อ 55010501
นายธนศาสตร์ เขียวมณี 55010519

อาจารย์ที่ปรึกษา
รองศาสตราจารย์ ดร.วรพงศ์ ตั้งศรีรัตน์

ปีการศึกษา 2558

บทคัดย่อ

ในปัจจุบันร้านอาหารทั่วไปในประเทศไทยโดยส่วนใหญ่แล้ว ยังใช้วิธีการสั่งอาหารแบบยืนต่อแถวเพื่อรอสั่งอาหาร หรือมีพนักงานคอยรับรายการอาหารและแจ้งจำนวนค่าอาหาร ซึ่งอาจจะเกิดปัญหาที่พนักงานบริการรับรายการอาหารผิด หรือเนื่องมาจากความผิดพลาดต่างๆ เราจึงตัดสินใจที่จะจัดทำโครงงานนี้ขึ้นมา เพื่อแก้ไขปัญหาเหล่านี้ โดยออกแบบระบบสั่งอาหารไร้สายที่สามารถปรับแต่งตามความต้องการได้ ลูกค้าสามารถสั่งอาหารได้จากการกดรหัสรายการอาหารบนแป้นปุ่มกดแล้วแสดงออกทางหน้าจอแอลซีดี ต่อมารายการอาหารจะถูกส่งข้อมูลไปยังห้องครัวผ่านระบบไวไฟแล้วรหัสรายการอาหารจะถูกนำไปเทียบกับฐานข้อมูล ในส่วนอินเตอร์เฟสของร้านอาหารถูกออกแบบโดยใช้โปรแกรม Visual Studio เพื่อที่ร้านอาหารจะสามารถเห็นรายการที่ถูกสั่งเข้ามาได้ตลอดเวลา และยังสามารถจัดการรายการอาหารที่มีอยู่หรือดูข้อมูลการเรียกเก็บค่าบริการได้ด้วยตนเอง โครงสร้างของระบบนี้ประกอบด้วยบอร์ดไมโครคอนโทรลเลอร์อาดูโน, จอแอลซีดี, แป้นพิมพ์, และไวไฟโมดูล โดยโครงงานชิ้นนี้สามารถให้ทั้งความสะดวก ประสิทธิภาพ และความถูกต้องต่อร้านอาหาร โดยการลดเวลา ลดความผิดพลาดจากบุคคล รวมทั้งรับความเห็นจากลูกค้าได้ตลอดเวลา

WIRELESS FOOD ORDERING SYSTEM

By

Mr. Kittiphop Chaiyaphan 55010093

Mr. Thanapat Poltue 55010501

Mr. Tanasart Kiewmanee 55010519

Advisor

Assoc. Prof. Dr. Worapong Tangsrirat

Academic Year 2015

ABSTRACT

Nowadays, most of the restaurants still use the waiters to take food orders from customers and prepare itemized checks and take payments. Which will have some mistake by human errors. Therefore, this project has been presented the concept and design a customizable food ordering system. The customers can make an order by keypad with the LCD screen. The order was updated in central database, and subsequently sent to kitchen via Wi-Fi system. The interface system was designed from Visual Studio in order that the restaurant can be seen the new update all the time. In addition, the restaurant can also manage the lists of food and payment information by themselves. The main structure of this system is based on Arduino microcontroller board, LCD module, Keypad module and Wi-Fi module. This project will provide the convenience, efficiency and accuracy for restaurants by saving time, reducing human errors, as well as real-time customer feedback.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดี เพราะได้รับความช่วยเหลือเป็นอย่างดีจาก รองศาสตราจารย์ ดร.วรพงศ์ ตั้งศรีรัตน์ ที่ให้ความเมตตากรุณาและให้คำปรึกษามาโดยตลอด รวมทั้งเอื้อเฟื้อสถานที่ อุปกรณ์ที่จำเป็นต่อการดำเนินงาน โดยผู้จัดทำรู้สึกซาบซึ้งและขอบพระคุณเป็นอย่างสูง

ขอบคุณเพื่อนๆ ทุกคนที่คอยให้กำลังใจ ให้ความช่วยเหลือสนับสนุนอุปกรณ์ที่ขาดเหลือ เตือนสติ และชักชวนทำแต่สิ่งที่ดีมีประโยชน์

สุดท้ายนี้ผู้จัดทำขอกราบขอบพระคุณบิดา มารดา และครอบครัว ผู้ให้การอุปการะเลี้ยงดู ให้การศึกษา และคอยเป็นกำลังใจให้ผู้จัดทำเสมอมา

ผู้จัดทำ
นายกิตติภพ ไชยพันธ์
นายธนภัทร พลตื้อ
นายธนศาสตร์ เขียวมณี

สารบัญ

	หน้า
บทคัดย่อ	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของปริญญานิพนธ์	1
1.2 วัตถุประสงค์ในการทำปริญญานิพนธ์	2
1.3 ขอบเขตของปริญญานิพนธ์	2
1.4 ขั้นตอนการดำเนินปริญญานิพนธ์	3
1.5 ตารางการทำงาน	4
1.6 ประโยชน์ที่คาดว่าจะได้รับ	5
1.7 รายละเอียดภายในปริญญานิพนธ์	5
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	6
2.1 ไมโครคอนโทรลเลอร์ (Microcontroller)	6
2.2 บอร์ดอาดูโน (Arduino Board)	9
2.3 โมดูลแอลซีดี (LCD Module)	11
2.4 โมดูลแป้นพิมพ์ (Keypad Module)	13
2.5 โมดูลไวไฟ (Module Wi-Fi ESP8266E-12F)	14
2.6 โปรโตคอล TCP/IP	16
2.7 วิธีการสื่อสารผ่านรูปแบบ TCP/IP	17
2.8 ส่วนของเครื่อง Server	18
บทที่ 3 การออกแบบระบบสั่งอาหารไร้สาย	23
3.1 การออกแบบระบบ	23
3.2 หลักการทำงานภาคสั่ง	25
3.3 หลักการทำงานภาครับ	26

สารบัญ (ต่อ)

		หน้า
บทที่ 4	**การทดลองและผลลัพธ์**	29
4.1	ผลการทดลองอุปกรณ์ภาครับและภาคส่งสัญญาณไร้สาย	29
4.2	ผลการทดลองการสั่งงานและแสดงผลที่หน้าจอแอลซีดี	32
4.3	ผลการทดลองแสดงข้อมูลรายการอาหารผ่านทางหน้าจอคอมพิวเตอร์	36
4.4	ผลการทดลองการส่งสัญญาณข้อมูล	38
บทที่ 5	**สรุปผลการดำเนินงาน**	39
5.1	สรุปผล	39
5.2	ปัญหาและข้อจำกัดของการดำเนินงาน	39
5.3	ปัญหาที่พบและแนวทางการแก้ไข	39
5.4	แนวทางการพัฒนาต่อไป	41
เอกสารอ้างอิง		42
ภาคผนวก		44
ภาคผนวก ก		45
ภาคผนวก ข		60
ภาคผนวก ค		64
ภาคผนวก ง		66
ภาคผนวก จ		67

สารบัญรูป

รูปที่		หน้า
1.1	ระบบการสั่งอาหารแบบดั้งเดิม	1
1.2	ระบบการสั่งอาหารไร้สาย	2
2.1	ลักษณะของตัวไมโครคอนโทรลเลอร์ Atmega328P	6
2.2	โครงสร้างของตัวไมโครคอนโทรลเลอร์ Atmega328P	7
2.3	ตัวบอร์ด Arduino UNO R3	9
2.4	ส่วนประกอบต่างๆ บนตัวบอร์ด Arduino UNO R3	9
2.5	โมดูลจอแอลซีดี	11
2.6	โมดูล I2C Bus	12
2.7	การเชื่อมต่อระหว่างบอร์ดอาดูโน กับโมดูลแอลซีดี (I2C)	12
2.8	โมดูลแป้นปุ่มกด	13
2.9	การเชื่อมต่อโมดูลแป้นปุ่มกด กับบอร์ดอาดูโน	14
2.10	โมดูล ESP8266 รุ่น ESP8266 E-12F	15
2.11	ลักษณะขาสัญญาณของตัวโมดูล ESP8266 E-12F	15
2.12	การเชื่อมต่อโมดูล ESP8266 E-12F กับบอร์ดอาดูโน	16
2.13	โครงสร้างของโปรโตคอล TCP/IP	17
2.14	โครงสร้างการเชื่อมต่อแบบ TCP	17
2.15	หน้าต่างโปรแกรม Visual Basic	20
2.16	หน้าต่างโปรแกรม phpMyAdmin	22
3.1	หลักการทำงานของระบบสั่งอาหารไร้สาย	23
3.2	ส่วนประกอบของเครื่องรับรายการอาหาร	24
3.3	โฟลว์ชาร์ตการทำงานของระบบสั่งอาหารไร้สาย	24
3.4	โฟลว์ชาร์ตการทำงานของเครื่องสั่งอาหาร	25
3.5	การต่อวงจรของเครื่องสั่งอาหารไร้สาย	26
3.6	โฟลว์ชาร์ตการทำงานของเครื่องรับรายการอาหาร	27
3.7	หน้าจออินเตอร์เฟสของเซิร์ฟเวอร์	28
4.1	เครื่องรับรายการอาหารไร้สายสำหรับโต๊ะ 1 และโต๊ะ 2	29
4.2	วงจรภายในเครื่องรับรายการอาหารไร้สาย	30
4.3	หน้าจอการทำงานของเครื่องรับอาหารไร้สาย แสดงข้อความต้อนรับ	30

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.4	หน้าจอการทำงานของเครื่องรับอาหารไร้สาย พร้อมที่รับออเดอร์จากลูกค้า	31
4.5	หน้าจอแสดงรายการอาหารในส่วนของผู้รับออเดอร์	31
4.6	รายละเอียดของรายการอาหารที่ลูกค้าสั่งจากโต๊ะ 1 จะปรากฏบนหน้าจอ	36
4.7	รายละเอียดของรายการอาหารที่ลูกค้าสั่งจากโต๊ะ 2 จะปรากฏบนหน้าจอ	37

สารบัญตาราง

ตารางที่		หน้า
1.1	ขั้นตอนการดำเนินงาน	4
2.1	การใช้งานขาต่างๆ ของไมโครคอนโทรลเลอร์ Atmega328P	8
2.2	เปรียบเทียบขาสัญญาณระหว่างบอร์ดอาดูโน กับ LCD (I2C)	13
3.1	หน้าที่การทำงานของอุปกรณ์	25
4.1	ขั้นตอนการทำงานของเครื่องรับรายการอาหารที่โต๊ะ 1	32
4.2	ขั้นตอนการทำงานของเครื่องรับรายการอาหารที่โต๊ะ 2	34
4.3	สัญญาณการส่งข้อมูล	38

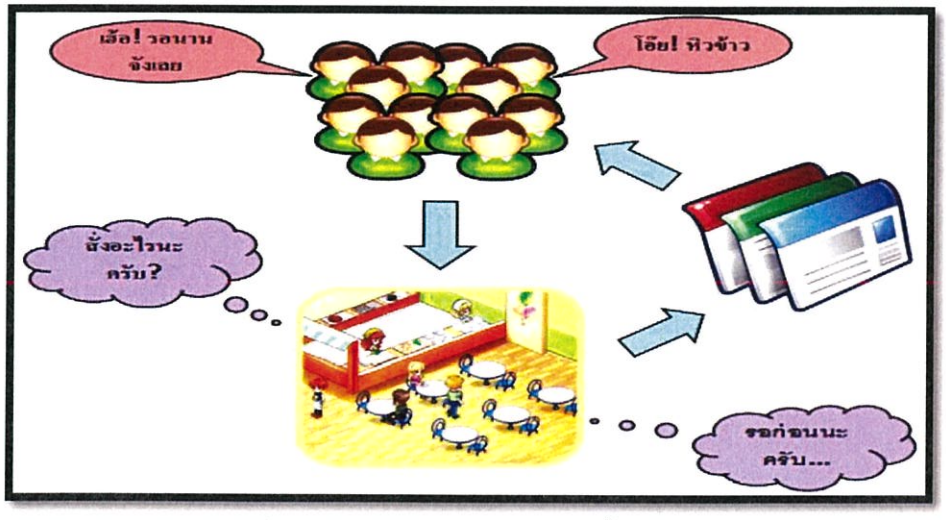
บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ปัจจุบันนี้ธุรกิจร้านอาหารมีจำนวนเพิ่มขึ้นอย่างต่อเนื่อง ก่อให้เกิดการแข่งขันกันทางธุรกิจมากขึ้น ดังนั้นจึงมีการนำเทคนิคต่างๆ มาใช้เพื่อเพิ่มแรงจูงใจให้ลูกค้า อาทิเช่น การทำโปรโมชั่นส่วนลดราคาอาหาร กริยามารยาทของพนักงานบริการ คุณภาพของวัตถุดิบที่ใช้ เป็นต้น นอกจากนี้มีการนำเทคโนโลยีสมัยใหม่เข้ามาช่วยในการสั่งรายการอาหาร แต่ยังเป็นเพียงส่วนน้อย โดยส่วนใหญ่แล้วร้านอาหารทั่วไปยังใช้วิธีการสั่งอาหารแบบยืนต่อแถวเพื่อรอสั่งอาหาร หรือมีพนักงานคอยรับรายการอาหารจากลูกค้า ซึ่งอาจเกิดปัญหา พนักงานบริการรับรายการอาหารผิด อันเนื่องมาจากความผิดพลาดในการสื่อสารกับลูกค้า และกรณีที่ลูกค้ามีจำนวนมากขึ้นจนไม่สามารถสั่งรายการอาหารได้ทันที ต้องรอพนักงานบริการมารับคำสั่งเป็นเวลานานทำให้เกิดความไม่พอใจ เกิดความความยุ่งยากไม่เป็นระเบียบตามมาได้

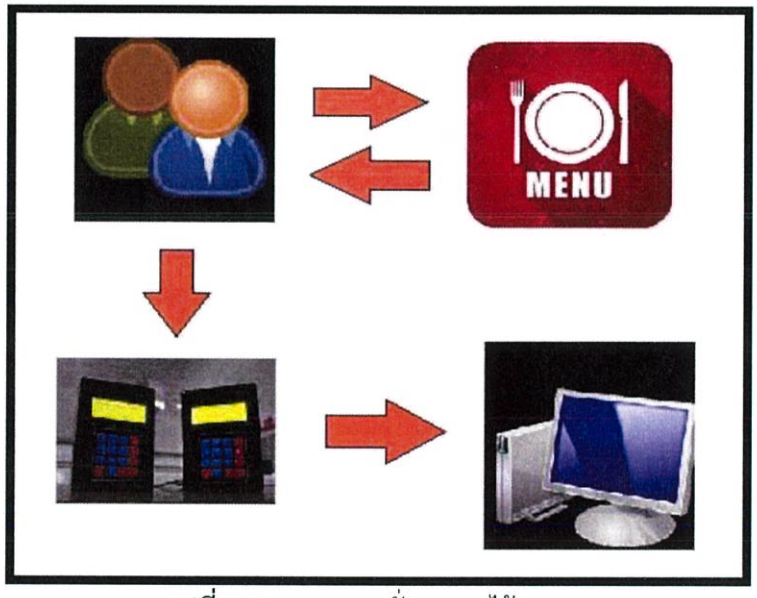
ปริญญานิพนธ์ฉบับนี้ได้นำเสนอระบบการรับส่งรายการอาหารแบบไร้สาย เพื่อเพิ่มความประทับใจในการให้บริการจากทางร้าน ซึ่งลูกค้าสามารถสั่งรายการอาหารได้ทันที และลดความผิดพลาดในการสั่งรายการอาหารระหว่างลูกค้ากับพนักงานบริการ อันเป็นระบบที่ลูกค้าสามารถสั่งอาหารได้ด้วยตนเองจากบนโต๊ะอาหารโดยตรง ไม่ต้องเสียเวลาไปยืนต่อแถวสั่งอาหาร และลูกค้าสามารถที่จะดูรายการ และราคาอาหารที่สั่งรวมไว้ได้ก่อนที่จะทำการยืนยันรายการสั่งอาหาร เพื่อช่วยให้ลูกค้าสามารถตัดสินใจเลือกรายการอาหารที่ต้องการจริง และคำนวณค่าอาหารที่จะต้องจ่ายไว้ล่วงหน้าได้



รูปที่ 1.1 ระบบการสั่งอาหารแบบดั้งเดิม

1.2 วัตถุประสงค์ในการทำปริญญานิพนธ์

1. เพื่อศึกษาการทำงานของตัวไมโครคอนโทรลเลอร์ (Microcontroller Atmega328P)
2. เพื่อศึกษาการรับส่งข้อมูลไร้สายระหว่างโมดูลไวไฟ ESP8266 E-12F กับระบบเซิร์ฟเวอร์
3. เพื่อศึกษาการเขียนโปรแกรม Arduino UNO R3 และโปรแกรม Visual Basic
4. เพื่อศึกษาการสร้างเซิร์ฟเวอร์ และส่วนฐานข้อมูลโดยใช้โปรแกรม Visual Basic, PHP, MySQL
5. เพื่อออกแบบ และสร้างเครื่องสั่งอาหารไร้สายใช้งานภายในร้านอาหาร



รูปที่ 1.2 ระบบการสั่งอาหารไร้สาย

1.3 ขอบเขตของปริญญานิพนธ์

เครื่องรับรายการอาหารจะถูกติดตั้งอยู่บนโต๊ะภายในร้านอาหาร โดยทางลูกค้าสามารถดูรหัสรายการอาหารจากเมนู และสั่งอาหารผ่านทางแป้นพิมพ์ ซึ่งถูกควบคุมโดยไมโครคอนโทรลเลอร์เพื่อส่งรายการอาหารไปยังเครื่องรับรายการอาหารไร้สาย หรือเครื่อง Servers ผ่านทางโมดูลไวไฟ ESP8266 และแสดงรายการอาหารที่ลูกค้าสั่งไว้ให้พนักงานบริการทราบผ่านทางหน้าจอแสดงผลหรือหน้าจอเครื่องคอมพิวเตอร์

สร้างส่วนเก็บรวบรวมข้อมูลรายการอาหารราคาอาหาร และโปรแกรมการคำนวณราคาอาหาร โดยใช้เป็นระบบ Servers ในที่นี้ได้จำลองการสั่งอาหารโดยใช้เครื่องรับรายการอาหารจำนวน 2 เครื่อง และเครื่อง Server จำนวน 1 เครื่อง

1.4 ขั้นตอนการดำเนินปริญญานิพนธ์

1.4.1 การศึกษาข้อมูล

1. ศึกษาหลักการทำงานของไมโครคอนโทรลเลอร์ แป้นพิมพ์ จอแอลซีดี (LCD) และการรับส่งข้อมูลผ่านพอร์ตอนุกรม (Series Port)

2. ศึกษาการเขียนโปรแกรมอาดูโน สั่งงานไมโครคอนโทรลเลอร์

3. ศึกษาการเขียนโปรแกรม Visual Basic, PHP, MySQL สร้างเซิร์ฟเวอร์และฐานข้อมูล

4. ศึกษาการรับส่งข้อมูลผ่านโมดูลไวไฟ (ESP8266 E-12F)

1.4.2 การออกแบบวงจรและเซิร์ฟเวอร์

การออกแบบวงจรของเครื่องสั่งรายการอาหารให้มีความเหมาะสมกับวัตถุประสงค์ในการใช้งาน รวมถึงการออกแบบระบบเซิร์ฟเวอร์ ระบบฐานข้อมูลรายการอาหาร และฟังก์ชันต่างๆ ในการใช้งาน

1.4.3 การสร้างอุปกรณ์

สร้างเครื่องรับส่งรายการอาหารให้มีขนาดเหมาะสมกับวัตถุประสงค์ในการใช้งาน

1.4.4 การสร้างเซิร์ฟเวอร์

การใช้โปรแกรม Visual Basic, PHP, MySQL สร้างเซิร์ฟเวอร์ และฐานข้อมูลขึ้นมาเพื่อใช้รับส่งข้อมูลกับเครื่องสั่งรายการอาหาร

1.4.5 การทดลอง

การทดลองให้เครื่องสั่งรายการอาหารส่งข้อมูลให้กับเครื่องเซิร์ฟเวอร์ จากนั้นตรวจสอบรายการอาหารที่รับเข้ามา

1.4.6 การปรับปรุงแก้ไข

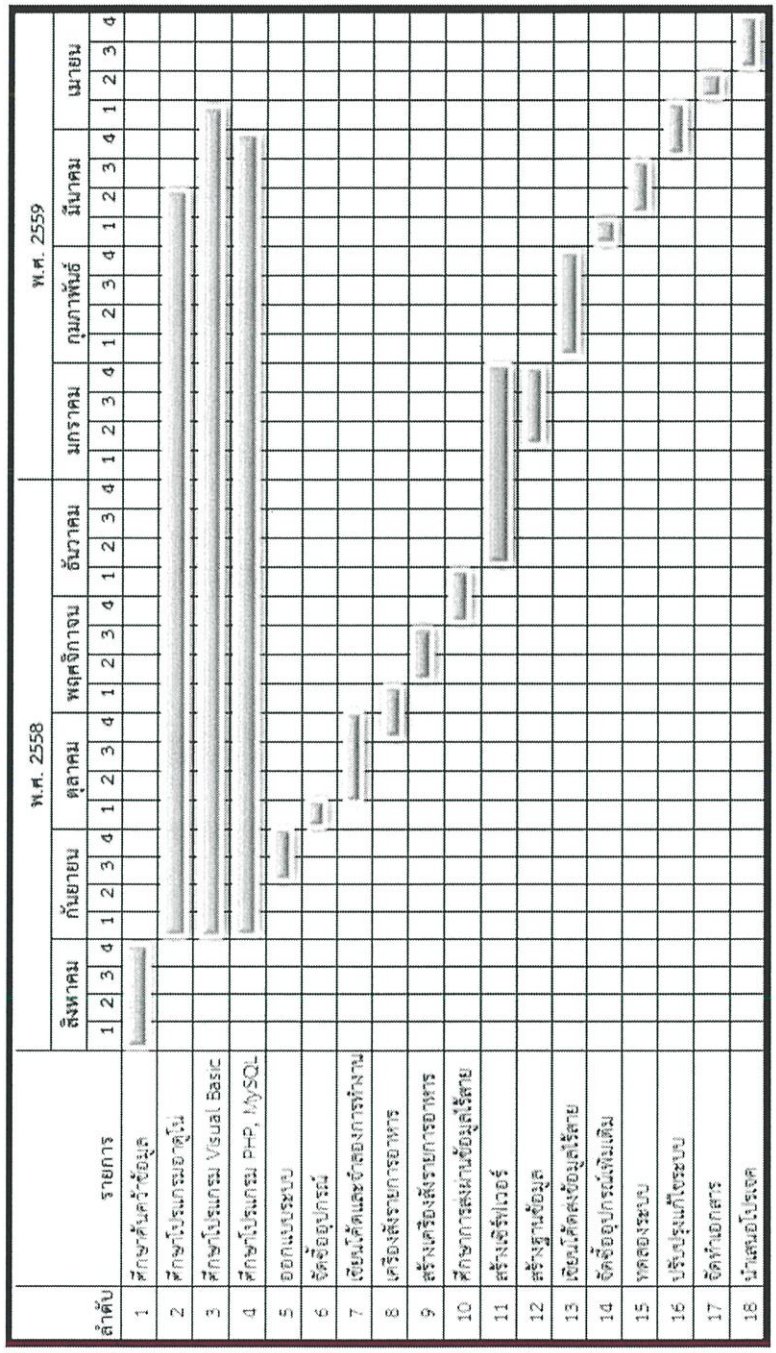
การแก้ไขซอร์สโค้ด (Source Code) โปรแกรมเซิร์ฟเวอร์ให้สามารถรับรายการอาหาร และการคิดราคาค่าอาหารได้อย่างถูกต้อง

1.4.7 การทำรูปเล่มปริญญานิพนธ์

การทำคู่มืออธิบายวิธีการใช้งานเครื่องรับรายการอาหาร พร้อมแนบเมนูรหัสรายการอาหาร

1.5 ตารางการทำงาน

ตารางที่ 1.1 ขั้นตอนการดำเนินงาน



ลำดับ	รายการ	พ.ศ. 2558																พ.ศ. 2559											
		สิงหาคม				กันยายน				ตุลาคม				พฤศจิกายน				ธันวาคม				มกราคม				กุมภาพันธ์			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	ศึกษาค้นคว้าข้อมูล																												
2	ศึกษาโปรแกรมยอดนิยม																												
3	ศึกษาโปรแกรม Visual Basic																												
4	ศึกษาโปรแกรม PHP, MySQL																												
5	ออกแบบระบบ																												
6	จัดซื้ออุปกรณ์																												
7	เขียนโค้ดและจัดลองการทำงาน																												
8	เครื่องสังรายการอาหาร																												
9	สร้างเครื่องสังรายการอาหาร																												
10	ศึกษาการจัดการข้อมูลให้สวย																												
11	สร้างเซิร์ฟเวอร์																												
12	สร้างฐานข้อมูล																												
13	เขียนโค้ดส่งข้อมูลให้สวย																												
14	จัดซื้ออุปกรณ์เพิ่มเติม																												
15	ทดลองระบบ																												
16	ปรับปรุงแก้ไขระบบ																												
17	จัดทำเอกสาร																												
18	นำเสนอโปรแกรม																												

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถนำระบบนี้ไปใช้ในร้านอาหาร เพื่อเพิ่มความสะดวกสบายให้ลูกค้า และลดความผิดพลาดในการสั่งอาหารได้

2. ทำให้มีความรู้ ความเข้าใจ เกี่ยวกับการทำงานของไมโครคอนโทรลเลอร์ และสามารถรับส่งข้อมูลผ่านโมดูลไร้สายได้

3. ทำให้มีความรู้ ความเข้าใจ ทางด้านการเขียนโปรแกรมอาดูโน ในการควบคุม และจำลองการทำงานของไมโครคอนโทรลเลอร์ การเขียนโปรแกรมด้านเซิร์ฟเวอร์ และระบบฐานข้อมูล

1.7 รายละเอียดภายในปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้ แบ่งเนื้อหาออกเป็น 5 บทดังต่อไปนี้

บทที่ 1 บทนำ กล่าวถึงความเป็นมา วัตถุประสงค์ ขอบเขต ขั้นตอนในการดำเนินการ ตารางการทำงาน ประโยชน์ที่คาดว่าจะได้รับและรายละเอียดภายในปริญญานิพนธ์

บทที่ 2 ทฤษฎี และความรู้ที่เกี่ยวข้อง กล่าวถึงทฤษฎี และหลักการต่างๆ ที่นำมาใช้ในการทำปริญญานิพนธ์ซึ่งประกอบด้วย ไมโครคอนโทรลเลอร์ โมดูลต่างๆ ที่ใช้ในการเชื่อมต่อกับไมโครคอนโทรลเลอร์ การสื่อสารไร้สาย การใช้งานโปรแกรมอาดูโน และการใช้โปรแกรม Visual Basic, PHP, MySQL

บทที่ 3 หลักการออกแบบ กล่าวถึงการออกแบบวงจรเครื่องสั่งรายการอาหาร การออกแบบระบบเซิร์ฟเวอร์และฐานข้อมูล

บทที่ 4 การทดลอง กล่าวถึงการทดลองรับส่งสัญญาณไร้สาย และตรวจสอบผลลัพธ์ที่ได้จากการทดลอง

บทที่ 5 บทวิจารณ์ และสรุป กล่าวถึงการสรุปผลการทำปริญญานิพนธ์ ปัญหาที่พบในระหว่างดำเนินงาน การแก้ปัญหา และแนวทางในการพัฒนาต่อไป

บทที่ 2

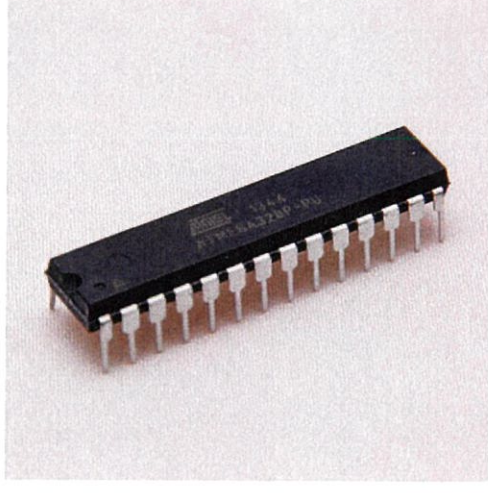
ทฤษฎีที่เกี่ยวข้อง

2.1 ไมโครคอนโทรลเลอร์ตระกูล AVR

ไมโครคอนโทรลเลอร์ตระกูล AVR เป็นไอซีไมโครคอนโทรลเลอร์ของทางบริษัท Atmel มีสถาปัตยกรรมในแบบ RISC (Reduced Instruction Set Computer) มีการรวบรวมเอาหน่วยประมวลผลกลาง (CPU : Central Processing Unit) หน่วยคำนวณทางคณิตศาสตร์และลอจิก (ALU : Arithmatic Logic Unit) วงจรรับสัญญาณอินพุต (Input Port) วงจรส่งสัญญาณเอาต์พุต (Output Port) หน่วยความจำ (ROM, RAM : Memory) วงจรกำเนิดสัญญาณนาฬิกา (Oscillator) ไว้ด้วยกัน โดยใช้สัญญาณนาฬิกาเพียง 1 ลูก ในการปฏิบัติงานใน 1 คำสั่ง ทำให้สามารถนำไปใช้งานแทนวงจรอิเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี ดังนั้นไมโครคอนโทรลเลอร์จึงเป็นอุปกรณ์ที่ใช้ในกาควบคุม โดยสามารถเขียนโปรแกรมเพื่อกำหนดรูปแบบการทำงานได้อย่างอิสระ

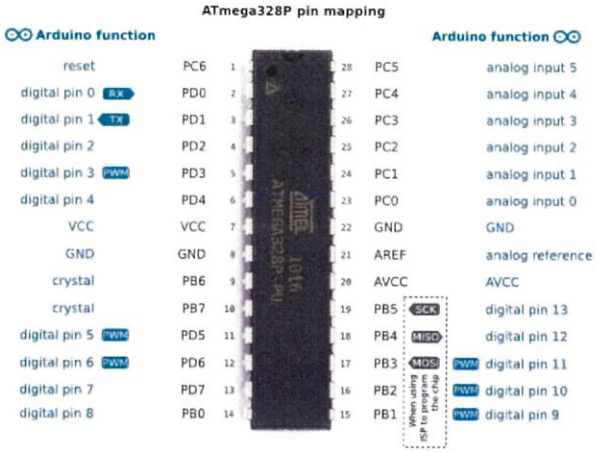
2.1.1 ไมโครคอนโทรลเลอร์ ATmega328P

Atmega328P เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต (Bits) ที่มีสถาปัตยกรรม RISC (Reduce Instruction Set Computer) ทำให้การประมวลผลมีความเร็ว 1 คำสั่ง/1 รอบนาฬิกา หรือหน่วยประมวลผลกลางสามารถประมวลคำสั่งได้ 1 ล้านคำสั่งต่อวินาที (MIPS/MHz)



รูปที่ 2.1 ลักษณะของตัวไมโครคอนโทรลเลอร์ ATmega328P

2.1.2 โครงสร้างของไมโครคอนโทรลเลอร์ ATmega328P



รูปที่ 2.2 โครงสร้างของตัวไมโครคอนโทรลเลอร์ ATmega328P

2.1.3 คุณสมบัติของไมโครคอนโทรลเลอร์ Atmega328P

1. สถาปัตยกรรมภายนอกถูกออกแบบให้ใช้สถาปัตยกรรมชั้นสูงแบบ RISC
2. ชุดคำสั่งควบคุมการทำงานของไมโครคอนโทรลเลอร์ 131 คำสั่ง/1 รอนาฬิกา
3. กลุ่มรีจิสเตอร์ใช้งานทั่วไปขนาด 8 บิต จำนวน 32 ตัว
4. ความเร็วในการประมวลผลมากกว่า 20 ล้านคำสั่ง/วินาที
5. หน่วยความจำแบบ 32 กิโลไบต์ (KByte) สามารถเขียนลบได้ 10,000 ครั้ง
6. หน่วยความจำแบบเอ็ปรอม (EPROM : Erasable Programmable Read-Only Memory) 1 กิโลไบต์ สามารถเขียนและลบโปรแกรมได้ 100,000 ครั้ง
7. หน่วยความจำแบบเอสแรม (SRAM : Static Random Access Memory) 2 กิโลไบต์
8. ความถี่สัญญาณนาฬิการะหว่าง 0-4 เมกะเฮิรตซ์ (MHz)
9. มีการรองรับอุปกรณ์ต่อพ่วง
10. อุปกรณ์สื่อสารข้อมูลแบบอนุกรมแบบ I^2C (Inter-IC Communication) และยูเอสอาร์ต (USART : Universal Synchronous Asynchronous Receiver Transmitter)
11. มีระบบการรีเซ็ต (Reset) แบบอัตโนมัติเมื่อจ่ายกระแสไฟฟ้าเข้าสู่บอร์ดไมโครคอนโทรลเลอร์
12. มีฟังก์ชัน (Function) ตรวจสอบแรงดัน
13. มีระบบการขัดจังหวะทั้งภายใน และภายนอก
14. มีระบบตรวจจับความผิดพลาดของคอมพิวเตอร์
15. มีระบบอนุรักษ์พลังงาน 5 แบบ ได้แก่ Idle, ADC Noise Reduction
16. ไฟเลี้ยงระหว่าง 1.8 -2.5 โวลต์ (V)

ตารางที่ 2.1 การใช้งานขาต่างๆ ของไมโครคอนโทรลเลอร์ Atmega328P

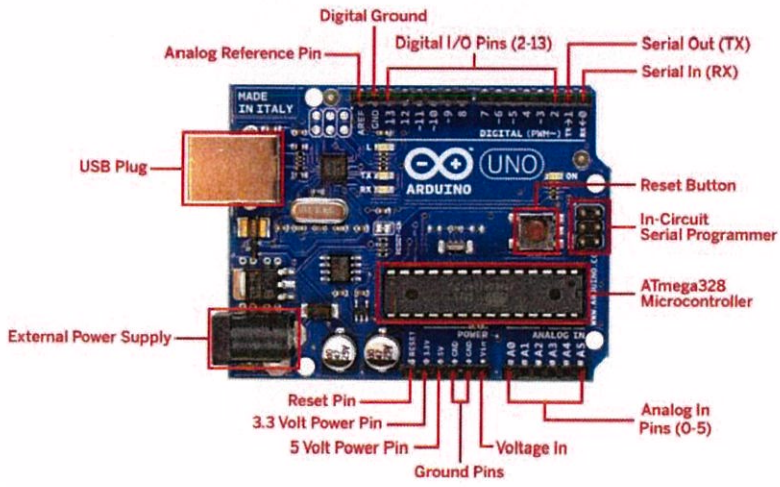
ลำดับขา	ชื่อขา	การใช้งาน
8,22	GND	- ขากราวด์ (Ground) ต่อสายดิน
7	VCC	- ไฟเลี้ยง 1.8-5.5 โวลต์
9,10,14-19	Port B (PB7-PB0) XTAL1/XTAL2/TOSC1/TOSC2	- เป็นพอร์ต 2 ทิศทาง ขนาด 8 บิต โดยสามารถกำหนดให้ขาของแต่ละพอร์ตสามารถตั้งค่าให้ พูลอัพรีซิสเตอร์ (Pull Up Resistor) ได้ (ภายในเป็นอิสระแยกจากกันเพื่อดึงแรงดันของลอจิก (Logic) 1 ให้เท่ากับ 5 โวลต์) - สามารถใช้งานพิเศษตามความต้องการของ ATmega328P โดยขึ้นอยู่กับการตั้งค่าสัญญาณนาฬิกาที่ขา PB6 ที่ใช้เป็นแรงดันและขาอินพุตของวงจรกำเนิดสัญญาณนาฬิกา
1	PC6/RESET	- ขารีเซ็ต
1-6,11-13	Port D (PD0-PD7)	- เป็นพอร์ตสองทิศทางขนาด 8 บิต โดยสามารถกำหนดให้ขาของแต่ละพอร์ตสามารถตั้งค่าให้พูลอัพรีซิสเตอร์ได้ - สามารถใช้งานพิเศษตามความต้องการของ ATmega328P
20	AVCC	-aใช้จ่ายไฟให้กับวงจรแปลงสัญญาณแอนะล็อก (Analog) เป็นดิจิตอล (Digital) มักจะต่อเข้ากับขา VCC
21	AREF	- แรงดันอ้างอิงที่ใช้งานในส่วนของวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิตอลมักต่อกับ VCC
9-10	Port B (PB6-PB7) (TQFP QFN/MLF Package Only)	- ขากำลังงานใช้แปลงสัญญาณแอนะล็อกเป็นดิจิตอล
23-28	Port C (PC0-PC5)	- เป็นพอร์ตสองทิศทางขนาด 8 บิต โดยสามารถกำหนดให้ขาของแต่ละพอร์ตสามารถตั้งค่าให้ Pull Up Resistor ได้ - สามารถใช้งานพิเศษตามความต้องการของ ATmega328P

2.2 บอร์ดอาดูโน (Arduino Board)

บอร์ดอาดูโน เป็นบอร์ดไมโครคอนโทรลเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัวบอร์ดอาดูโนถูกออกแบบมาให้ใช้งานได้ง่าย เหมาะแก่การนำไปเริ่มต้นทำการศึกษา และสามารถนำไปดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด หรือตัวโปรแกรมต่อได้อีกด้วย โดยในการทำโปรเจคนี้ทำการใช้ไมโครคอนโทรลเลอร์เบอร์ ATmega328P ของบริษัท Atmel เป็น MCU (Multipoint Control Unit) ประจำบอร์ด โดย MCU รุ่นนี้มีขาทั้งหมด 28 ขา ดังแสดงในรูปที่ 2.3

รูปที่ 2.3 ตัวบอร์ด Arduino UNO R3

2.2.1 โครงสร้างของบอร์ดอาดูโน



รูปที่ 2.4 ส่วนประกอบต่างๆ บนตัวบอร์ด Arduino UNO R3

2.2.2 คุณสมบัติของบอร์ดอาดูโน

บอร์ดอาดูโนมี ATmega328P เป็น MCU ตระกูล AVR ประจำบอร์ด โดยเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบคริสตอลออสซิเลเตอร์ (Crystal Oscillator) ที่มีค่า 16 เมกะเฮิรตซ์ เพื่อให้สามารถใช้งานพอร์ตสื่อสารอนุกรมได้อย่างลงตัว

1. บอร์ดสามารถเปลี่ยนการติดตั้ง MCU เป็นแบบ 28 ขา หรือเบอร์อื่นในอนุกรมเดียวกันได้ โดยไม่ต้องมีการดัดแปลงใดๆ เช่น ATmega88 เป็นต้น

2. มีหน่วยความจำแบบแฟรช (Flash) 32 กิโลไบต์ โดยแบ่งเป็น Boot Loader 2 กิโลไบต์

3. มีเอ็ปรอม 1 กิโลไบต์/เอสแรม 2 กิโลไบต์

4. มีพอร์ตเป็น 14 ดิจิตอล อินพุต/เอาต์พุต ซึ่งมี 6 ขา สามารถสร้างเป็น PWM (Pulse Width Modulation) เอาต์พุต

5. มีพอร์ต 6 แอนะล็อก อินพุต/เอาต์พุต

6. ไฟกระแสตรง (DC) ขาอินพุต/เอาต์พุต มีค่า 40 มิลลิแอมป์ (mA)

7. ไฟกระแสตรง ขา 3.3 โวลต์ มีค่า 50 มิลลิแอมป์

8. MCU ประจำบอร์ดที่ได้รับการติดตั้ง Boots Loader สามารถอัพโหลดโค้ดให้บอร์ดผ่านทางพอร์ตสื่อสารแบบอนุกรมได้ทันที

9. มีขั้วต่อ USB Interface สื่อสารข้อมูลแบบอนุกรมเข้าคอมพิวเตอร์ได้ทันที

10. มีแอลอีดีสำหรับแสดงสถานะไฟเลี้ยง

11. มีแอลอีดีแสดงสถานการณ์รับส่งข้อมูล

12. ใช้ไฟเลี้ยงประจำบอร์ด 7-12 โวลต์

2.2.3 การใช้งานพอร์ตต่างๆ ของบอร์ดอาดูโน

1. พอร์ต USB (Universal Serial Bus) จะใช้ต่อกับคอมพิวเตอร์เพื่ออัพโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ดอาดูโน

2. ปุ่มรีเซ็ต เป็นปุ่มรีเซ็ต ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่

3. พอร์ต ICSP (In Circuit Serial Programming) ของ Atmega16U2 มีพอร์ตที่ใช้โปรแกรม Visual Com Port บน Atmega16U2

4. พอร์ตอินพุต/เอาต์พุต เป็นดิจิตอล อินพุต/เอาต์พุต ตั้งแต่ขา D0 ถึง D13 โดย ขา 0, 1 เป็นขา Tx, Rx Serial, ขา 3, 5, 6, 9, 10 และ 11 เป็นขา PWM

5. พอร์ต ICSP Atmega328P เป็นพอร์ตที่ใช้โปรแกรม Boot Loader

6. MCU Atmega328P เป็น MCU ที่ใช้บนบอร์ดอาดูโน

7. พอร์ตอินพุต/เอาต์พุต เป็นดิจิตอล อินพุต/เอาต์พุต และเปลี่ยนเป็นช่องรับสัญญาณแอ

นะล็อกตั้งแต่ขา A0-A5

8. พอร์ตพาวเวอร์ (Power Port) ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง +3.3 โวลต์, +5 โวลต์, กราวด์, Vin

9. พาวเวอร์แจ็ค (Power Jack) รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 โวลต์

10. MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328P จะติดต่อกับคอมพิวเตอร์ผ่าน Atmega16U2

2.2.4 จุดเด่นของบอร์ดอาดูโน

1. การใช้งานและเรียนรู้ง่าย เนื่องจากมีการออกคำสั่งต่างๆ ขึ้นมาสนับสนุน การใช้งานด้วยรูปแบบที่ง่ายไม่ซับซ้อน

2. ราคาไม่แพง เนื่องจากมีซอร์สโค้ด และวงจรแจกฟรี สามารถต่อวงจรขึ้นมาใช้เองได้ และมีการเปิดเผยวงจรซอร์สโค้ดทำให้สามารถนำไปพัฒนาต่อยอดได้ดี ทั้งด้านฮาร์ดแวร์และซอฟต์แวร์

3. โปรแกรมที่ใช้พัฒนาของอาดูโน สามารถรองรับการทำงานทั้ง Window Linux และ OSX

4. รูปแบบคำสั่งนั้นง่ายต่อการใช้งาน สามารถนำไปใช้งานจริงกับส่วนที่มีความซับซ้อนมากได้ และยังสามารถสร้างคำสั่งรวมถึงไลบรารี (Library) ใหม่ๆ ขึ้นมาใช้งานเองได้เมื่อมีความชำนาญมากขึ้น

2.3 โมดูลจอแอลซีดี (LCD Module)

โมดูลแอลซีดี รุ่น 2004 LCD (Yellow Screen) ขนาด 20 ตัวอักษร 4 บรรทัด ใช้ไฟขนาด 5 โวลต์ มาพร้อมกับไฟ Blacklight เชื่อมต่อกับบอร์ด Arduino ในรูปแบบการเชื่อมต่อแบบ I2C การเลือกใช้หน้าจอแอลซีดี ขนาด 20 × 4 เนื่องจากสามารถทำการเพิ่มข้อความ หรือข้อมูลในภาคส่งข้อมูลได้เพิ่มขึ้น

รูปที่ 2.5 โมดูลจอแอลซีดี

2.3.1 ส่วนประกอบที่สำคัญ

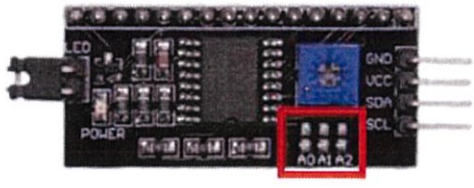
1. ตัวควบคุม (Controller) เป็นอุปกรณ์สำหรับรับข้อมูลที่ส่งมาจากอุปกรณ์ภายนอก เพื่อควบคุมการทำงานภายในโมดูลแอลซีดี เช่น การแสดงตัวอักษร การลบข้อความ ฯลฯ

2. ตัวขับ (Driver) เป็นอุปกรณ์สำหรับรับข้อมูลจากตัวควบคุม เพื่อส่งข้อมูลไปที่ตัวแสดงผล แสดงข้อมูลตามที่กำหนด

3. ตัวแสดงผล (Dot Matrix Display) เป็นอุปกรณ์แสดงผลให้สามารถมองเห็นเป็นตัวอักษร หรืออักขระ ซึ่งภายในชุดแสดงผลให้เห็น โดยอาศัยการเปิดและปิดตัวเองกับแสงจากภายนอก

2.3.2 การเชื่อมต่อโมดูลจอแอลซีดี กับไมโครคอนโทรลเลอร์ผ่านทาง โมดูล I2C Bus

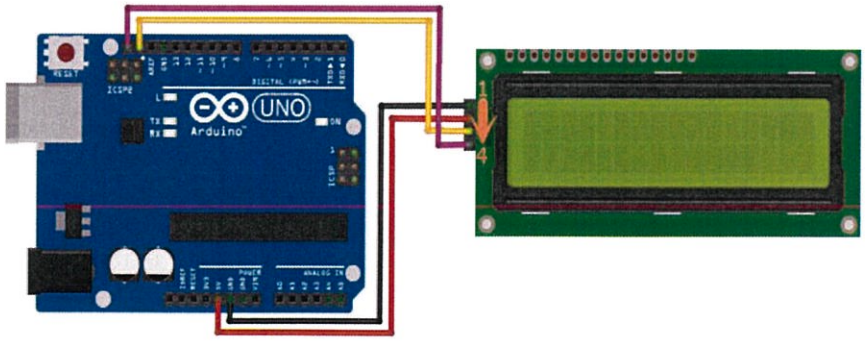
ในการเชื่อมต่อหน้าจอโมดูลแอลซีดี กับไมโครคอนโทรลเลอร์จะใช้รูปแบบการเชื่อมต่อแบบ I2C หรือเรียกอีกอย่างหนึ่งว่าการเชื่อมต่อแบบ Serial โดยโมดูล I2C Bus เป็นโมดูลแปลงขาสัญญาณ Parallel เป็น I2C เพื่อลดจำนวนขาสัญญาณในการใช้งานจาก 16 ขาเหลือเพียง 4 ขา ส่วนใหญ่นำมาใช้งานคู่กับจอ Character LCD เพื่อให้สะดวกต่อการเชื่อมต่อ และสามารถต่อจอแสดงผลหรืออุปกรณ์อื่นได้ในจำนวนที่มากขึ้น สามารถปรับให้รองรับได้ถึง 8 ตำแหน่งแอดเดรส (20-27) เพื่อความยืดหยุ่นในการใช้งาน



รูปที่ 2.6 โมดูล I2C Bus

2.3.3 การเชื่อมต่อบอร์ดอาดูโน กับโมดูลแอลซีดี (I2C)

การเชื่อมต่อระหว่างหน้าจอแอลซีดี และบอร์ดอาดูโนจะใช้สายทั้งหมด 4 เส้น โดยการต่อสายจะแสดงดังตารางที่ 2.2 และการเชื่อมต่อผ่านโมดูล (I2C Bus) ทำให้ลดการใช้สาย



รูปที่ 2.7 การเชื่อมต่อระหว่างบอร์ดอาดูโน กับโมดูลแอลซีดี (I2C)

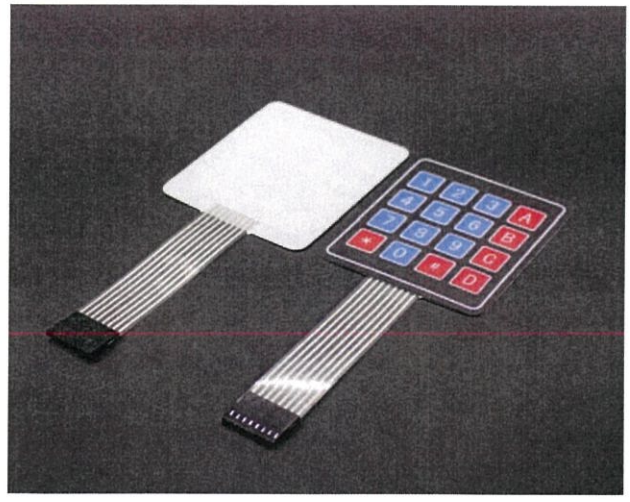
ตารางที่ 2.2 เปรียบเทียบขาสัญญาณระหว่างบอร์ดอาดูโนกับ LCD (I2C)

Arduino UNO R3	LCD (I2C)
GND	GND (Pin 1)
+5 VDC	VCC (Pin 2)
A4 (SDA)	SDA (Pin 3 Serial Data)
A5 (SCL)	SCL (Pin 4 Serial Clock)

1. GND เป็น Ground ใช้ต่อระหว่าง Ground ของระบบ Microcontroller กับ LCD
2. VCC เป็นไฟเลี้ยงวงจรที่ป้อนให้กับ LCD มีขนาด +5VDC
3. SDA (Serial Data) เป็นขาที่ใช้ในการรับส่งข้อมูล
4. SCL (Serial Clock) เป็นขาสัญญาณนาฬิกาในการรับส่งข้อมูล

2.4 โมดูลแป้นปุ่มกด 4 x 4 Matrix Keypad

แป้นปุ่มกดหรือ Keypad หมายถึง อุปกรณ์อ่านค่าอินพุตต่อเข้ากับแป้นปุ่มกด จำนวนของแป้นปุ่มกดจะเท่ากับขาพอร์ตของที่ใช้งาน ซึ่งเป็นการสิ้นเปลืองขาพอร์ตจึงมีการต่อใช้สวิตซ์จำนวนมากนี้ในรูปแบบของสวิตซ์เมตริก (Matrix Switch) จึงทำให้ลดการใช้พอร์ตได้เป็นอย่างดี การต่อใช้งานแป้นปุ่มกดเมตริกซ์สวิตซ์จะต่อใช้งานแบบพลูอัพ ดังนั้นสถานะของสวิตซ์จะมีสถานะเป็น “1” ทั้งหมด เมื่อต้องการอ่านค่าของแป้นปุ่มกด จะต้องกำหนดค่าของหลัก (Column) ให้สถานะเป็น “0” เพราะหลักนั้นจะเป็นการควบคุมสัญญาณ หากสวิตซ์แถวไหน มีการเปลี่ยนแปลง แสดงว่าแถวนั้นมีการกดสวิตซ์

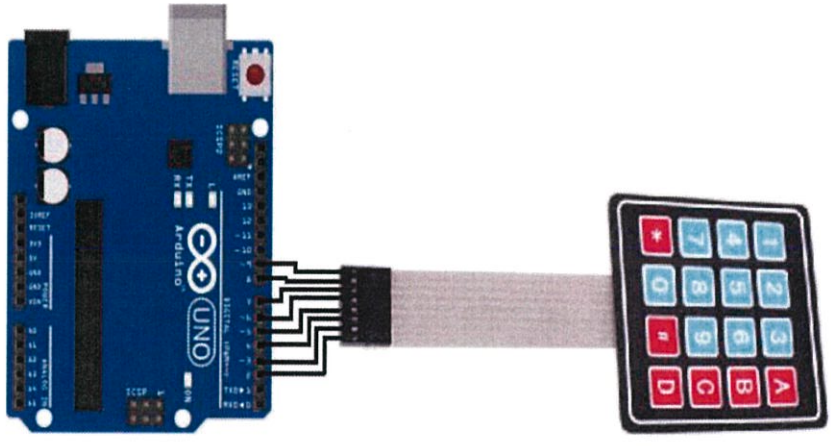


รูปที่ 2.8 โมดูลแป้นปุ่มกด

2.4.1 การใช้งานโมดูลแป้นปุ่มกดขนาด 4 x 4 Matrix Keypad

แป้นปุ่มกด เป็นอุปกรณ์สำหรับรับอินพุตจากผู้ใช้ มีลักษณะเป็นปุ่มกดหลายปุ่ม ถูกจัดเรียงกันในลักษณะเป็นอาร์เรย์ แบ่งเป็นแถวแนวนอน (Row) และแถวแนวตั้ง (Column) เช่น 3 x 4 (12 ปุ่ม) หรือ 4 x 4 (16 ปุ่ม) เป็นต้น แต่ละปุ่มก็จะมีสัญลักษณ์เขียนกำกับไว้ เช่น ตัวเลข 0-9, #, * เป็นต้น โดยปกติถ้าทำการต่อปุ่มกดแยกจำนวน 16 ตัว จะต้องใช้ขาสัญญาณทั้งหมด 16 ขา แต่ถ้าใช้การจัดเรียงแบบ 4 x 4 จะใช้ขาสัญญาณเพียง 8 ขา แต่ต้องมีเทคนิคในการตรวจดูว่า ปุ่มกดใดถูกกดบ้างในขณะนั้น วิธีการนี้เรียกว่า การสแกนปุ่มกด (Key Scan)

แป้นปุ่มกดแบบ 4 x 4 มีสายเชื่อมต่อและคอนเนกเตอร์ (Connector) จำนวน 8 ขา ถ้ามองจากทางด้านหน้าและนับจากซ้ายไปขวา จะเป็นขาหมายเลข 1-8 ตามลำดับ โดยที่ขา 1-4 จะเป็นขาสำหรับแถวแนวนอน (Row) และขา 5-8 จะเป็นขาแนวตั้ง (Column) ในการใช้งานร่วมกับบอร์ดอาดูโนจะต้องทำการต่อตัวต้านทานแบบ (Pull Up) โดยใช้ตัวต้านทานขนาด 1k และ 10k (โอห์ม) ซึ่งใช้ตัวต้านทานต่อเข้าเฉพาะที่ขาในแนวตั้ง (Column) ทั้ง 4 ขา

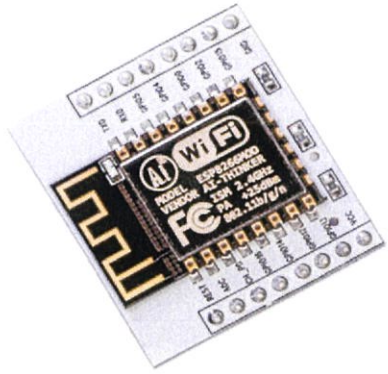


รูปที่ 2.9 การเชื่อมต่อโมดูลแป้นปุ่มกด กับบอร์ดอาดูโน

2.5 โมดูล Wi-Fi ESP8266E-12F

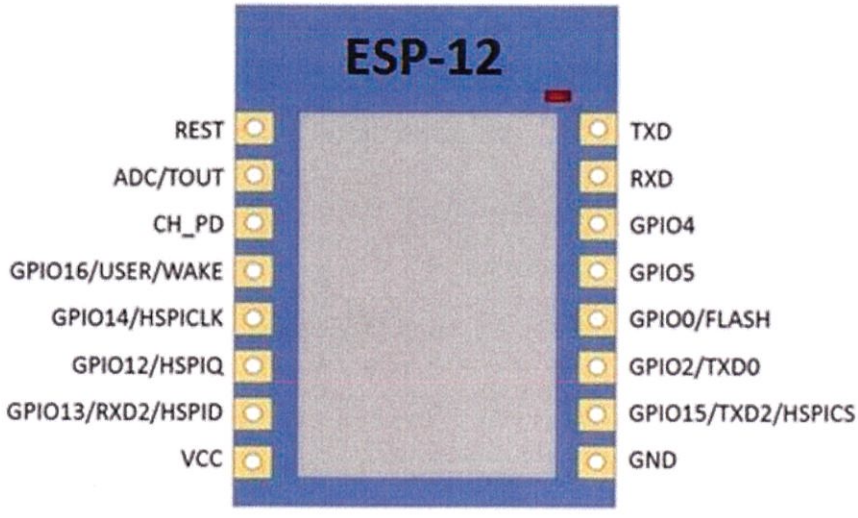
ESP8266E-12F คือ โมดูล Wi-Fi ที่มีความพิเศษตรงที่ตัวมันสามารถโปรแกรมลงไปได้ ทำให้สามารถนำไปใช้งานแทนไมโครคอนโทรลเลอร์ได้เลย และมีพื้นที่โปรแกรมที่มากถึง 4MB ทำให้มีพื้นที่เหลือมากในการเขียนโปรแกรมลงไป โมดูล ESP8266 เป็นชื่อของชิปไอซีบนบอร์ดของโมดูล ซึ่งไอซี ESP8266 ไม่มีพื้นที่โปรแกรม (Flash Memory) ในตัว ทำให้ต้องใช้ไอซีภายนอก (External Flash Memory) ในการเก็บโปรแกรม ที่ใช้การเชื่อมต่อผ่านโปรโตคอล SPI ซึ่งสาเหตุนี้เองทำให้โมดูล ESP8266 มีพื้นที่โปรแกรมมากกว่าไอซีไมโครคอนโทรลเลอร์เบอร์อื่นๆ โมดูล ESP8266 ทำงานที่แรงดันไฟฟ้า 3.3V - 3.6V การนำไปใช้งานร่วมกับเซนเซอร์อื่นๆ ที่ใช้แรงดัน 5V ต้องใช้วงจร

แบ่งแรงดันมาช่วย เพื่อไม่ให้โมดูลพังเสียหาย กระแสที่โมดูลใช้งานสูงสุดคือ 200mA ความถี่คริสตอล 40MHz ทำให้เมื่อนำไปใช้งานอุปกรณ์ที่ทำงานรวดเร็วตามความถี่ เช่น LCD ทำให้การแสดงผลข้อมูลรวดเร็วกว่าไมโครคอนโทรลเลอร์ยอดนิยม Arduino มาก

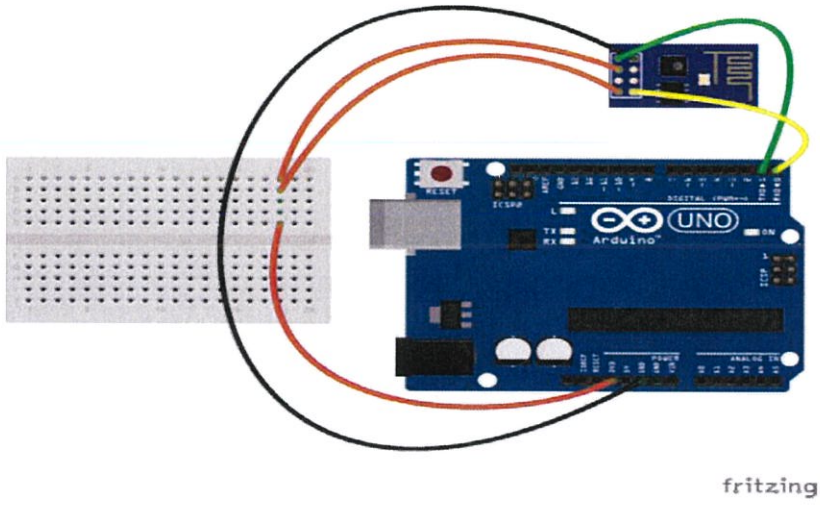


รูปที่ 2.10 โมดูล ESP8266 รุ่น ESP8266 E-12F

สำหรับโมดูล ESP8266 รุ่น E-12F จะมีขาเป็น 16 ขา มีขา GPIO ที่ใช้งานได้ 7 ขา ได้แก่ 2 4 5 12 13 14 16 สามารถใช้งานเป็นดิจิตอลอินพุตเอาต์พุตได้ ส่วนขา Tx Rx เป็นขาสำหรับต่อ Serial Port ขา GPIO0 สำหรับเลือกโหมด GPIO15 ต้องต่อลงกราวด์ไว้เสมอ ขา CH_PD ต่อเข้าไฟ + ขา Reset สามารถปล่อยว่างไว้ได้ เพียงต่อเสาอากาศเปลี่ยนเป็นแบบลายทองแดงบน PCB รุ่นนี้นิยมใช้งานมากในการทดลองหรือพัฒนา เนื่องจากไม่ต้องต่อเสาอากาศเพิ่มขึ้นมา มีความเสถียร และความเร็วในการดำเนินการโปรแกรมเท่ากับ ESP-07



รูปที่ 2.11 ลักษณะขาสัญญาณของตัวโมดูล ESP8266 E-12F



รูปที่ 2.12 การเชื่อมต่อโมดูล ESP8266 E-12F กับบอร์ดอาดูโน

2.6 โปรโตคอล TCP/IP

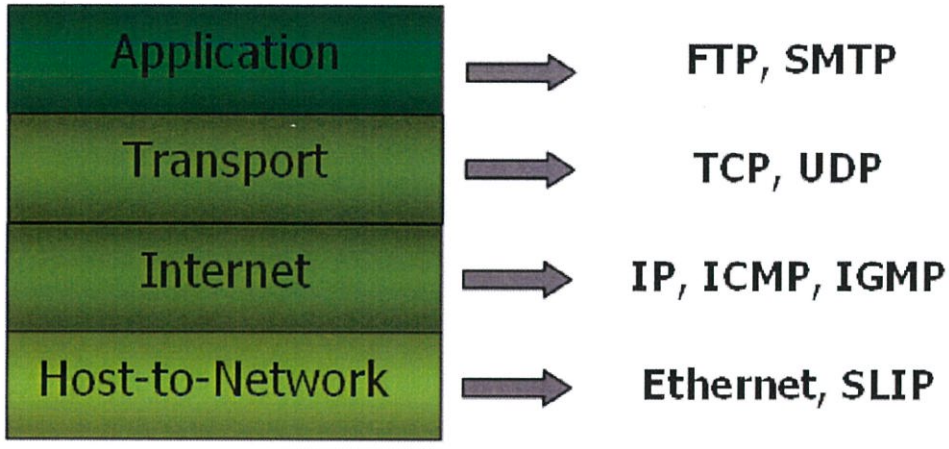
โปรโตคอล TCP/IP เป็นชื่อเรียกของชุดโปรโตคอลที่สำคัญ มีการใช้งานกันอย่างแพร่หลาย ตามการขยายตัวของอินเตอร์เน็ต/อินทราเน็ต ความจริงแล้วโปรโตคอล TCP/IP เป็นกลุ่มของโปรโตคอลหลายตัวที่ประกอบกันเป็นชุดให้ใช้งาน โดยมีคำเต็มว่า Transmission Control Protocol/Internet Protocol ซึ่งจากชื่อเต็มทำให้เห็นว่าอย่างน้อยก็มีโปรโตคอลประกอบกันทำงานร่วมกัน 2 โปรโตคอลคือ TCP และ IP โดย TCP ก็เป็นส่วนหนึ่งของคำว่า TCP/IP เพราะว่า IP หรือ Internet Protocol Suite มี TCP เป็นส่วนหนึ่งในนั้น บ่อยครั้งจึงเรียกกันว่า TCP/IP หรือเรียกว่า TCP อย่างเดียว โดย TCP มีชื่อเต็มๆ ว่า Transmission Control Protocol เป็นการทำงานหนึ่งที่อยู่ใน Transport Layer ของ Network ส่วน IP จะอยู่ใน Internet Layer โดยทั้งสองส่วนมีหน้าที่จัดการการรับส่งข้อมูลระหว่างเครื่อง ข้อมูลที่รับส่งมีความน่าเชื่อถือ ไม่ผิดเพี้ยนเพราะมีการเช็คความถูกต้องของข้อมูล และข้อมูลเรียงเป็นลำดับที่ถูกต้อง

ตัวอย่างของกลุ่มโปรโตคอลในชุดของ TCP/IP ที่พบและใช้งานบ่อยๆ (ส่วนใหญ่จะไม่ใช้งานตรงๆ แต่ใช้งานผ่านแอพพลิเคชั่นต่างๆ หรือใช้งานโดยทางอ้อม) เช่น Internet Protocol (IP), Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP), User Datagram Protocol (UDP), Transport Control Protocol(TCP), Simple Mail Transfer Protocol (SMTP) และ Domain Name System (DNS) เป็นต้น

โปรโตคอลที่มีบทบาทสำคัญในการทำงานในเครือข่ายอินเตอร์เน็ต คือ Internet Protocol (โปรโตคอล IP) เนื่องจากเมื่อโปรโตคอลอื่นๆ ต้องการส่งผ่านข้อมูลข้ามเครือข่ายในอินเตอร์เน็ตนั้น จะต้องอาศัยการผนึกข้อมูล (Encapsulation) ไปกับโปรโตคอล IP ที่มีกลไกการระบุเส้นทาง (Route

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

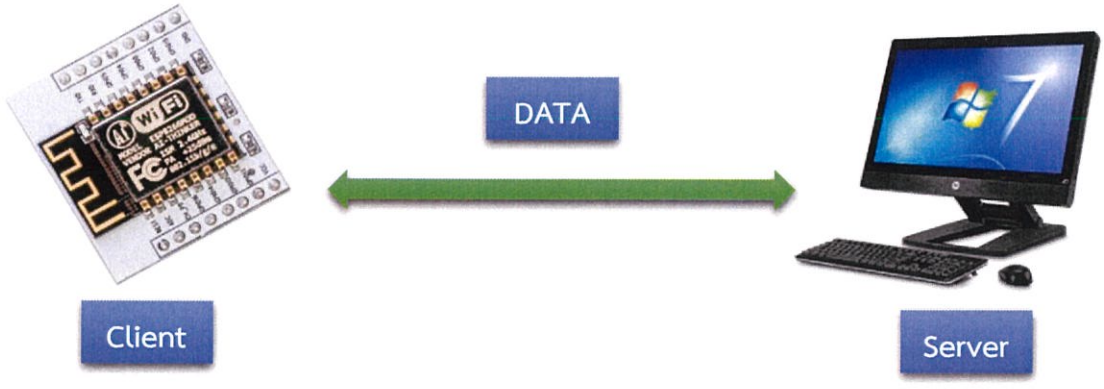
Service) ผ่าน Gateway หรือ Router เพื่อนำข้อมูลไปยังเครือข่ายและเครื่องปลายทางที่ถูกต้อง เนื่องจากกลไกการระบุเส้นทางจะทำงานที่โปโตคอล IP เท่านั้น และด้วยเหตุนี้จึงเรียก IP ว่าเป็นโปรโตคอลที่มีความสามารถระบุเส้นทางการส่งผ่านข้อมูลได้ (Routable)



รูปที่ 2.13 โครงสร้างของโปรโตคอล TCP/

2.7 วิธีการสื่อสารผ่านรูปแบบ TCP/IP

ในการส่งข้อมูลก่อนอื่นต้องเข้าใจว่า TCP นั้นไม่สามารถที่จะส่งข้อมูลไปยังอีกฝั่งได้เลยในทันที เพราะปลายทางก็ต้องอยู่ในสถานะที่พร้อมรับข้อมูลด้วย โดยฝั่งต้นทางจะขอเรียกว่า Client และปลายทางเรียกว่า Server



รูปที่ 2.14 โครงสร้างการเชื่อมต่อแบบ TCP

ดังนั้นก่อนที่จะส่งข้อมูลฝั่ง Server ก็จะต้องอยู่ในสถานะพร้อมรับข้อมูลด้วย (ต่อไปจะขอเรียกว่า "เปิด Server") โดยเปิดทิ้งไว้เพื่อรอให้ฝั่ง Client ส่งข้อมูลมาได้ แต่ถ้าฝั่ง Server ปิดอยู่ก็จะทำให้ Client ส่งข้อมูลมาไม่ได้นั่นเอง

นอกจากการเปิด-ปิด Server แล้ว ยังมีเรื่องของการเชื่อมต่อระหว่าง Client กับ Server อยู่

144284

ด้วย เพราะก่อนที่จะส่งข้อมูลให้ Server นั้น ฝั่ง Client จะทำการเชื่อมต่อกับ Server เสียก่อน (ทำให้รู้ได้ว่าปลายทางมีอยู่จริงหรือไม่)

2.7.1 การส่งข้อมูลแบบ Continuous TCP

การส่งข้อมูลที่ใช้สำหรับเครื่องส่งอาหารไร้สาย จะใช้เป็นแบบ Continuous TCP ให้ทำการ Open ส่วนของ Server ทิ้งไว้ตลอดเวลา แล้วส่งข้อมูลจากฝั่งของ Client ได้เรื่อยๆ จนกว่าจะทำการ Close ในส่วนของตัว Server หรือหมายความว่าให้คงสถานะ Open ค้างไว้ตลอดเวลาที่ใช้งาน แล้วระหว่างนั้นก็สามารถรับส่งข้อมูลได้เรื่อยๆ จนกว่าจะสั่ง Close ซึ่งจะต่างจากแบบการส่งทีละชุดแล้วจบการเชื่อมต่อ ซึ่งแบบการส่งทีละชุดจะทำให้ Server ต้องคอย Open และ Close ไปมา แต่วิธีนี้เปิดทีเดียวแล้วส่งได้ตลอดเวลา และข้อมูลก็ไม่จำเป็นต้องต่อเนื่อง ส่งชุดแรกเสร็จ แล้วอีกซักพักค่อยส่งชุดถัดไปก็ได้

2.8 ส่วนของเครื่อง Server

2.8.1 หน้าที่การทำงานของเครื่อง Server

1. รับข้อมูลรายการอาหารจากลูกค้าแต่ละโต๊ะ ผ่านทางระบบไร้สาย

2. แสดงรายละเอียดข้อมูลรายการอาหารที่ลูกค้าแต่ละโต๊ะสั่ง โดยแสดงรายการอาหารผ่านทางหน้าจอแสดงผลที่สร้างขึ้น

3. ทำการส่งค่าตอบกลับเพื่อทำการยืนยันการส่งข้อมูลรายการอาหารของลูกค้าแต่ละโต๊ะ โดยส่งค่าตอบกลับไปยังที่โต๊ะของลูกค้า

4. สามารถสั่งพิมพ์ใบเสร็จเพื่อใช้ในการเรียกเก็บเงินจากลูกค้าได้

2.8.2 โปรแกรมของเครื่อง Server มีลักษณะการทำงาน ดังนี้

เมื่อโปรแกรมของเครื่อง Server อยู่ในสถานะพร้อมทำงาน โปรแกรมจะทำการรอรับข้อมูลรายการอาหารจากเครื่องสั่งอาหาร โดยทำการส่งข้อมูลผ่านทางโมดูลไร้สายเชื่อมต่อกับส่วนของเครื่อง Server ซึ่งข้อมูลที่ถูกส่งมาจะอยู่ในรูปแบบตัวเลข จากนั้นส่วนของ Server จะนำชุดตัวเลขไปประมวลผล และทำการแสดงผลของข้อมูลที่ได้รับมาผ่านทางหน้าจอแสดงผลที่สร้างขึ้นและทำการตอบกลับเพื่อใช้สำหรับยืนยันว่าเครื่อง Server นั้น ได้รับข้อมูลจากทางลูกค้าเรียบร้อยแล้ว โดยทำการส่งการแจ้งเตือนการยืนยันไปยังโต๊ะของลูกค้าที่ส่งข้อมูลมา

2.8.3 ส่วนโปรมแกรม User Interface

การสร้างหน้าจอ Interface ของเครื่อง Server ถูกสร้างขึ้นโดยโปรแกรม Visual Basic 2010 เพื่อให้สามารถติดต่อกับผู้ใช้งานได้ตามต้องการ

Visual Basic เป็นภาษาคอมพิวเตอร์ (Programming Language) ที่พัฒนาโดยบริษัท

ไมโครซอฟท์ ซึ่งเป็นบริษัทยักษ์ใหญ่ที่สร้างระบบปฏิบัติการ Windows 95/98 และ Windows NT ที่ใช้กันอยู่ในปัจจุบัน โดยตัวภาษาเองมีรากฐานมาจากภาษา Basic ซึ่งย่อมาจาก Beginner's All Purpose Symbolic Instruction ถ้าแปลให้ได้ตามความหมายก็คือ "ชุดคำสั่งหรือภาษาคอมพิวเตอร์สำหรับผู้เริ่มต้น" ภาษา Basic มีจุดเด่นคือ ผู้ที่ไม่มีพื้นฐานเรื่องการเขียนโปรแกรมเลยก็สามารถเรียนรู้และนำไปใช้งานได้อย่างง่ายดายและรวดเร็ว เมื่อเทียบกับการเรียนภาษาคอมพิวเตอร์อื่นๆ เช่น ภาษาซี (C), ปาสคาส (Pascal), ฟอร์แทรน (Fortian) หรือ แอสเซมบลี (Assembler)

ไมโครซอฟท์ที่ได้พัฒนาโปรแกรมภาษา Basic มานานนับสิบปี ตั้งแต่ภาษา MBASIC (Microsoft Basic). BASICA (Basic Advanced) : GWBASIC และ QuickBasic ซึ่งได้ติดตั้งมาพร้อมกับระบบปฏิบัติการ MS DOS ในที่สุดโดยใช้ชื่อว่า QBASIC โดยแต่ละเวอร์ชันที่ออกมานั้นได้มีการพัฒนา และเพิ่มเติมคำสั่งต่างๆ เข้าไปโดยตลอด ในอดีตโปรแกรมภาษาเหล่านี้ล้วนทำงานใน Text Mode คือ เป็นตัวอักษรล้วนๆ ไม่มีภาพกราฟฟิกสวยงามแบบระบบ Windows อย่างในปัจจุบัน จนกระทั่งเมื่อระบบปฏิบัติการ Windows ได้รับความนิยมอย่างสูงและเข้ามาแทนที่ DOS ไมโครซอฟท์ก็เล็งเห็นว่าโปรแกรมภาษาใน Text Mode นั้นคงถึงกาลที่หมดสมัย จึงได้พัฒนาปรับปรุงโปรแกรมภาษา Basic ของตนออกมาใหม่เพื่อสนับสนุนการทำงานในระบบ Windows ทำให้ Visual Basic ถือกำเนิดขึ้นมาตั้งแต่บัดนั้น

Visual Basic เวอร์ชันแรกคือ เวอร์ชัน 1.0 ออกสู่สายตาประชาชนตั้งแต่ปี 1991 โดยในช่วงแรกนั้นยังไม่มีความสามารถต่างจากภาษา GBASIC มากนัก แต่จะเน้นเรื่องเครื่องมือที่ช่วยในการเขียนโปรแกรมวินโดว์ซึ่งปรากฏว่า Visual Basic ได้รับความนิยม และประความสำเร็จเป็นอย่างดี ไมโครซอฟท์จึงพัฒนา Visual Basic ให้ดีขึ้นเรื่อยๆ ทั้งในด้านประสิทธิภาพ ความสามารถ และเครื่องมือต่างๆ เช่น เครื่องมือตรวจสอบแก้ไขโปรแกรม (Debugger) สภาพแวดล้อมของการพัฒนาโปรแกรม การเขียนโปรแกรมแบบหลายวินโดว์ย่อย (MDI) และอื่นๆ อีกมากมาย

ข้อดีของการเขียนโปรแกรมด้วย Visual Basic

สาเหตุที่ Visual Basic เป็นภาษาที่เหมาะสำหรับการเรียนรู้ในการเขียนโปรแกรมนั้นเนื่องจาก Visual Basic มีข้อดีหลายประการคือ

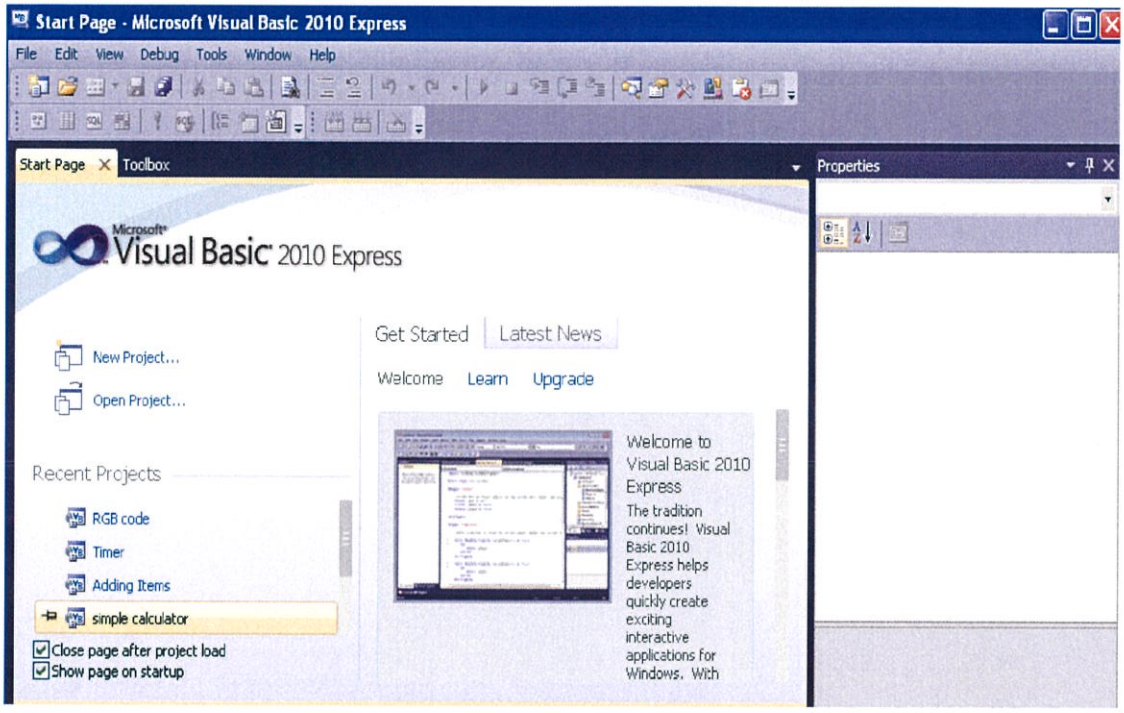
1. ง่ายต่อการเรียนรู้เหมาะสำหรับผู้เริ่มต้นใช้งานโปรแกรม ทั้งในเรื่องไวยากรณ์ของภาษาเอง และเครื่องมือการใช้งานที่เปิดกว้างให้ผู้ที่สนใจเรียนรู้ได้ค้นคว้าอย่างมากมาย

2. ความนิยมของตัวภาษา โดยอาจกล่าวได้ว่าภาษา Basic นั้นเป็นภาษาที่คนเรียนรู้ และใช้งานมากที่สุดในประวัติศาสตร์ของคอมพิวเตอร์

3. การพัฒนาอย่างต่อเนื่อง การปรับปรุงประสิทธิภาพในด้านของตัวภาษา และความเร็วของการประมวลผล และในเรื่องของความสามารถใหม่ๆ เช่น การติดต่อกับระบบฐานข้อมูล รวมถึงการ

เชื่อมต่อกับเครือข่ายอินเตอร์เน็ต

4. ผู้พัฒนาสำคัญของ Visual Basic คือ บริษัทไมโครซอฟท์ซึ่งจัดว่าเป็นยักษ์ใหญ่ของวงการคอมพิวเตอร์ในปัจจุบัน จึงสามารถมั่นใจได้ว่า Visual Basic จะยังมีการพัฒนา ปรับปรุงและคงอยู่ไปอีกนาน



รูปที่ 2.15 หน้าต่างโปรแกรม Visual Basic

2.8.4 ส่วนของระบบฐานข้อมูล

MySQL เป็นโปรแกรมจัดการฐานข้อมูล Relational Database Management System (RDBMS) เป็นฐานข้อมูลที่สามารถจัดเก็บ ค้นหา เรียงข้อมูล และดึงข้อมูล MySQL มีความสามารถให้ผู้ใช้งานเข้าดึงข้อมูลได้หลายๆ คนในเวลาเดียวกันได้และมีการเข้าถึงข้อมูลที่รวดเร็ว มีการกำหนดการเข้าใช้งานของผู้ใช้ในแบบต่างๆ อย่างเหมาะสม ปลอดภัย MySQL ถูกใช้งานเมื่อปี 1996 แต่โปรแกรมนี้พัฒนาตั้งแต่ปี 1979 และชนะรางวัล Linux Journal Reader ‘s Choice Award 3 ปีซ้อน ปัจจุบัน MySQL ได้ใช้งานแพร่หลายโดยเป็นโปรแกรม Open Source License แต่ก็มีแบบ Commercial License ให้ใช้ด้วย โดยคุณสมบัติจะแตกต่างกันออกไป

phpMyAdmin คือ โปรแกรมที่ถูกพัฒนาโดยใช้ภาษา PHP เพื่อใช้ในการบริหารจัดการฐานข้อมูล MySQL แทนการคีย์คำสั่ง เนื่องจากถ้าจะใช้ฐานข้อมูลที่เป็น MySQL บางครั้งจะมีความลำบากและยุ่งยากในการใช้งาน ดังนั้นจึงมีเครื่องมือในการจัดการฐานข้อมูล MySQL ขึ้นมาเพื่อให้สามารถจัดการ ตัว DBMS ที่เป็น MySQL ได้ง่ายและสะดวกยิ่งขึ้น โดย phpMyAdmin ก็ถือเป็นเครื่องมือชนิหนึ่งในการจัดการนั้นเอง phpMyAdmin เป็นส่วนต่อประสานที่สร้างโดยภาษาพีเอชพี ซึ่งใช้จัดการฐานข้อมูล MySQL ผ่านเว็บเบราว์เซอร์ โดยสามารถที่จะทำการสร้างฐานข้อมูลใหม่ หรือ

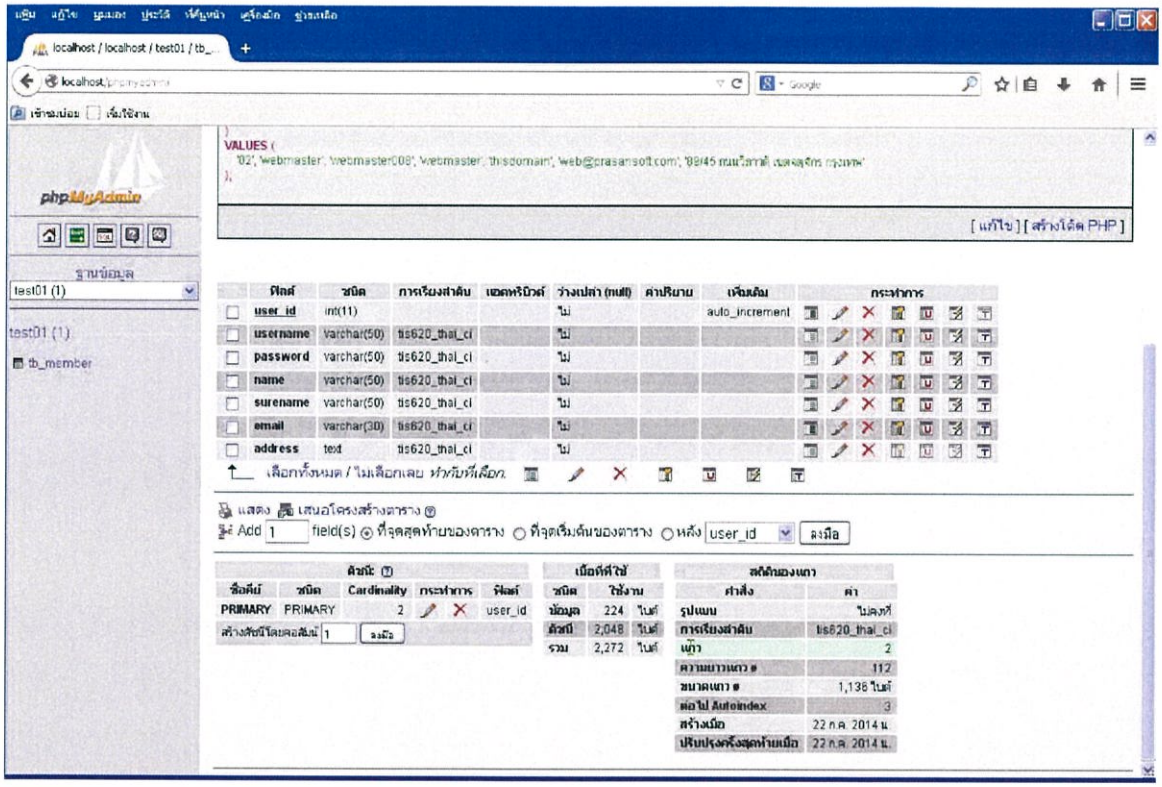
ทำการสร้าง TABLE ใหม่ๆ และยังมี Function ที่ใช้สำหรับการทดสอบการ Query ข้อมูลด้วยภาษา SQL พร้อมกันนั้น ยังสามารถทำการ Insert, Delete, Update หรือแม้กระทั่งใช้ คำสั่งต่างๆ เหมือนกันกับการใช้ภาษา SQL ในการสร้างตารางข้อมูล

phpMuAdmin เป็นโปรแกรมประเภท MySQL Client ตัวหนึ่งที่ใช้ในการจัดการข้อมูล MySQL ผ่าน Web Browser ได้โดยตรง phpMyAdmin ตัวนี้จะทำงานบน Web Server เป็น PHP Application ที่ใช้ควบคุมจัดการ MySQL Server ความสามารถของ phpMyAdmin คือ

1. สร้างและลบ Database
2. สร้างและจัดการ Table เช่น การแทรก Record, การลบ Record, การแก้ไข Record, การลบ Table, และการแก้ไข Field
3. โหลดเท็กซ์ไฟล์เข้าไปเก็บเป็นข้อมูลในตารางได้
4. หาผลสรุป (Query) ด้วยคำสั่ง SQL

PHP เป็นภาษาจำพวก Scripting Language คำสั่งต่างๆ จะเก็บอยู่ในไฟล์ที่เรียกว่าสคริปต์ (Script) และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษาสคริปก็เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมา เพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า Server-Side หรือ HTML-embedded Scripting Language เป็นเครื่องมือที่สำคัญชนิดหนึ่ง ที่ช่วยให้สามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น ถ้าใครรู้จัก Server Side Include (SSI) ก็จะสามารถเข้าใจการทำงานของ PHP ได้ไม่ยาก สมมติว่าต้องการจะแสดงวันเวลาปัจจุบันที่ผู้เข้ามาเยี่ยมชมเว็บไซต์ในขณะนั้น ในตำแหน่งใดตำแหน่งหนึ่งภายในเอกสาร HTML ที่ต้องการ อาจจะใช้คำสั่งในรูปแบบนี้ เช่น อนุมัติก่อนที่จะส่งไปยังผู้อ่านอีกทีหนึ่งอาจจะกล่าวได้ว่า PHP ได้รับการพัฒนาขึ้นมาเพื่อแทนที่ SSI รูปแบบเดิมๆ โดยให้มีความสามารถและมีส่วนเชื่อมต่อกับเครื่องมือชนิดอื่นมากขึ้น เช่น ติดต่อกับคลังข้อมูลหรือ Database เป็นต้น

PHP ได้รับการเผยแพร่เป็นครั้งแรกในปี ค.ศ. 1994 จากนั้นก็มีการพัฒนาต่อมาตามลำดับเป็นเวอร์ชั่น 1 ในปี 1995 เวอร์ชั่น 2 (ตอนนั้นใช้ชื่อว่า PHP/FI) ในช่วงระหว่าง 1995-1997 และเวอร์ชั่น 3 ช่วง 1997 ถึง 1999 จนถึงเวอร์ชั่น 4 ในปัจจุบัน



รูปที่ 2.16 หน้าต่างโปรแกรม phpMyAdmin

Apache คือ Web Server พัฒนามาจาก HTTPD Web Server โดยเจ้า Apache นี้จะทำหน้าที่ในการจัดเก็บ Homepage และส่ง Homepage ไปยัง Browser ที่มีการเรียกเข้ายัง Web Server ที่เก็บ HomePage นั้นอยู่ ซึ่งปัจจุบันจัดได้ว่าเป็น Web Server ที่มีความน่าเชื่อถือมาก เนื่องจากเป็นที่นิยมใช้กันทั่วโลก อีกทั้งอาปาเช่ยังเป็นซอฟต์แวร์แบบโอเพ่นซอร์ส ที่เปิดให้บุคคลทั่วไปสามารถเข้ามาร่วมพัฒนาส่วนต่างๆ ของอาปาเช่ได้ ซึ่งทำให้เกิดเป็นโมดูลที่เกิดประโยชน์มากมาย เช่น mod_perl,mod_python หรือ mod_php และทำงานร่วมกับภาษาอื่นได้ แทนที่จะเป็นเพียงเซิร์ฟเวอร์ที่ให้บริการเพียงแค่ HTML อย่างเดียว โดยสามารถหา Download ได้จาก www.apache.org

นอกจากนี้อาปาเช่เองยังมีความสามารถอื่นๆ ด้วย เช่น การยืนยันตัวบุคคล หรือเพิ่มความปลอดภัยในการสื่อสารผ่านโปรโตคอล https และยังมีโมดูลอื่นๆ ที่ได้รับความนิยมใช้ เช่น mod_vhost ทำให้สามารถสร้างโฮสท์เสมือนภายในเครื่องเดียวกันได้

บทที่ 3
การออกแบบระบบสั่งอาหารไร้สาย

3.1 การออกแบบระบบ

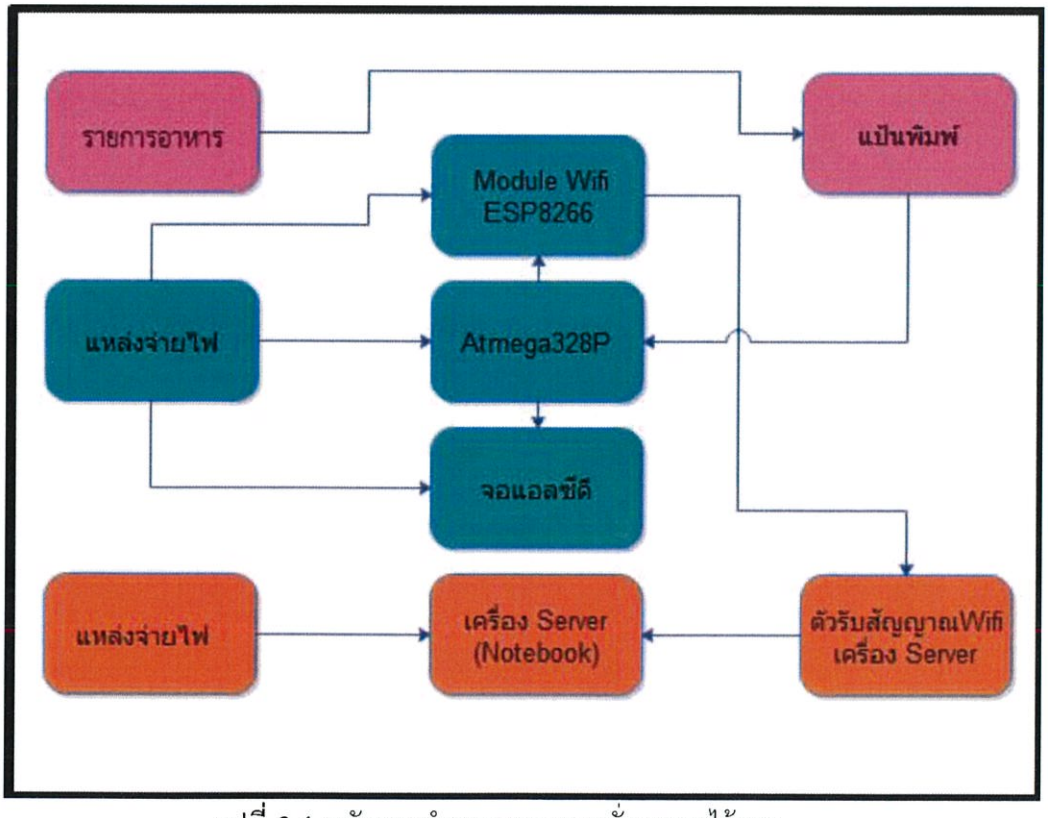
3.1.1 การออกแบบระบบสั่งอาหารไร้สาย

ระบบสั่งอาหารไร้สายที่นำเสนอ มีหลักการออกแบบการทำงานในรูปที่ 3.1 สามารถอธิบายหลักการทำงานได้ดังนี้

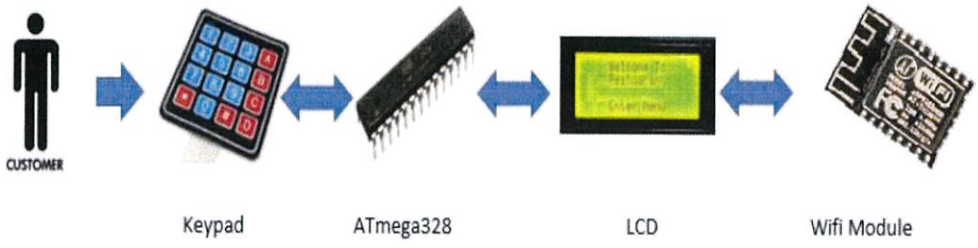
1. ทำการบันทึกรายการ และราคาอาหารในฐานข้อมูลที่เชื่อมต่อกับเซิร์ฟเวอร์ โดยกำหนดรายการอาหารตามหมายเลขที่ตั้งไว้

2. ทำการสั่งอาหารได้โดยการป้อนหมายเลขรายการอาหาร และจำนวนรายการอาหารที่ตัวรับรายการอาหาร จากนั้นข้อมูลจะถูกเก็บไว้ที่ไมโครคอนโทรลเลอร์ เมื่อทำการยืนยันรายการอาหารเรียบร้อยแล้วไมโครคอนโทรลเลอร์จะส่งข้อมูลไปที่ Module ESP8266 แล้ว Module ESP8266 จะส่งต่อข้อมูลแบบไร้สายไปให้ตัวรับสัญญาณที่เครื่องเซิร์ฟเวอร์แล้วนำไปแสดงผลผ่านหน้าจออินเตอร์เฟส

3. เมื่อทำการจ่ายเงินที่ร้าน ทางร้านจะทำการเคลียร์รายการอาหารที่เซิร์ฟเวอร์

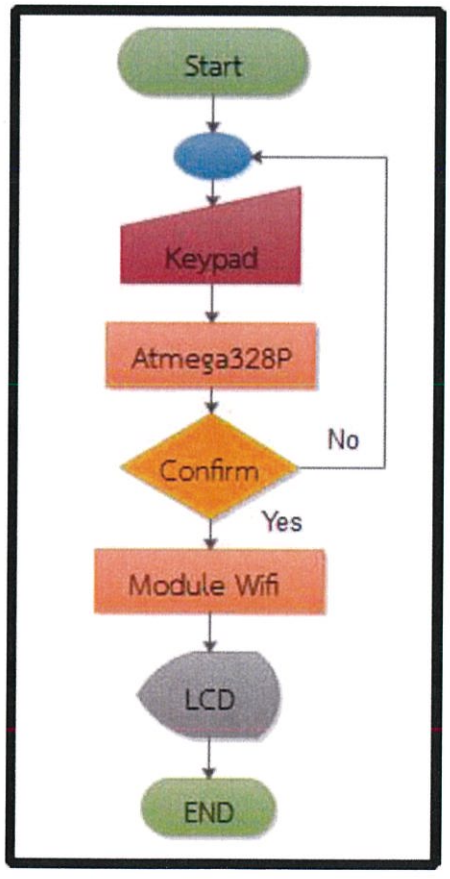


รูปที่ 3.1 หลักการทำงานของระบบสั่งอาหารไร้สาย



รูปที่ 3.2 ส่วนประกอบของเครื่องรับรายการอาหาร

การทำงานของไมโครคอนโทรลเลอร์นั้นมีลำดับขั้นการทำงานดังโฟลว์ชาร์ต (Flowchart) ที่แสดงในรูปที่ 3.3 โดยเริ่มจากการป้อนรหัสรายการอาหารผ่านแป้นพิมพ์ให้ตัวไมโครโทรลเลอร์ Atmega328P ทำการประมวลผล จากนั้นเมื่อกดยืนยันตัวไมโครคอนโทรลเลอร์ Atmega328P จะส่งข้อมูลผ่าน Module Wi-Fi ESP8266 เพื่อแสดงผลทางจอแอลซีดี แต่ถ้ากดปฏิเสธระบบจะย้อนกลับมาเริ่มใหม่

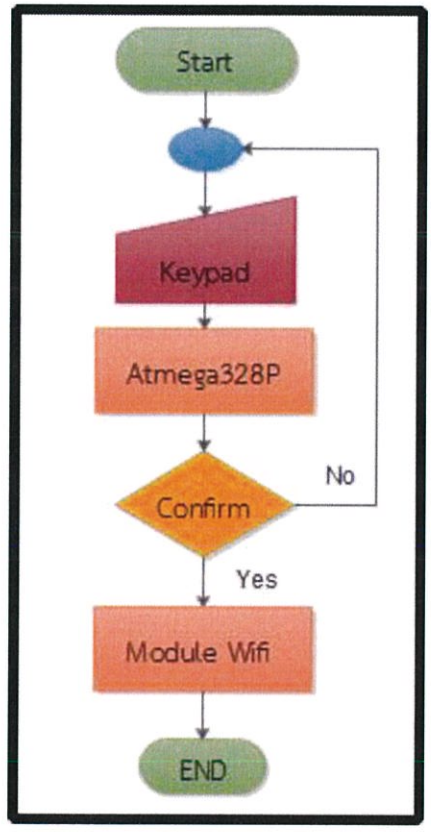


รูปที่ 3.3 โฟลว์ชาร์ตการทำงานของระบบสั่งอาหารไร้สาย

3.2 หลักการทำงานภาคสั่ง

3.2.1 วงจรสั่งอาหาร

การทำงานของไมโครคอนโทรลเลอร์ในส่วนภาคสั่งนั้น มีลำดับขั้นการทำงานดังโฟลว์ชาร์ต (Flowchart) ที่แสดงในรูปที่ 3.4 โดยเริ่มจากการป้อนรหัสรายการอาหารผ่านแป้นพิมพ์ให้ตัวไมโครคอนโทรลเลอร์ Atmega328P จากนั้นเมื่อกดยืนยันตัวไมโครคอนโทรลเลอร์ Atmega328P จะส่งข้อมูลผ่าน Module ESP8266-E12F ไปที่ระบบเซิร์ฟเวอร์ แต่ถ้ากดปฏิเสธระบบจะย้อนกลับมาเริ่มใหม่

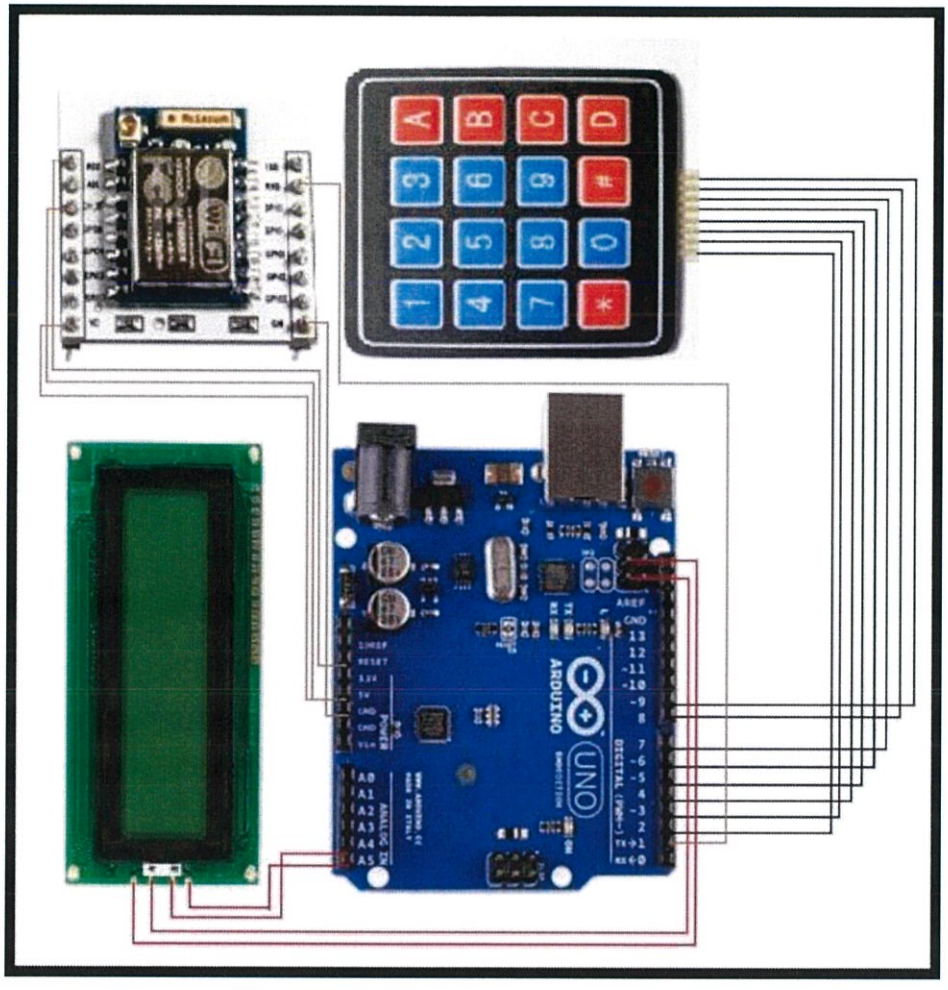


รูปที่ 3.4 โฟลว์ชาร์ตการทำงานของเครื่องสั่งอาหาร

โดยระบบที่นำเสนอจะใช้วงจรการต่ออุปกรณ์ดังรูปที่ 3.5 มีการรับค่าเป็นรหัสของรายการอาหารแล้ว ส่งข้อมูลไปยังส่วนของเซิร์ฟเวอร์ โดยมีอุปกรณ์สำคัญดังตารางที่ 3.1 ดังนี้

ตารางที่ 3.1 หน้าที่การทำงานของอุปกรณ์

อุปกรณ์	หน้าที่การทำงาน
Atmega328P	ควบคุมการทำงานของระบบ
แป้นพิมพ์	รับค่าข้อมูลจากผู้ใช้
จอแอลซีดี	แสดงผลของข้อมูล
Module Wi-Fi	รับ - ส่งข้อมูลแบบไร้สาย

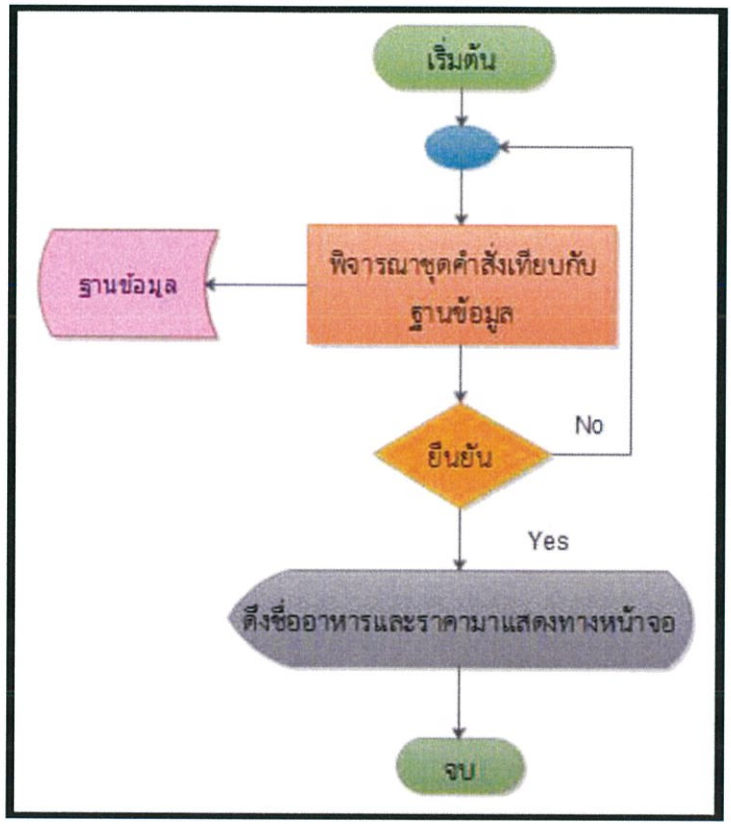


รูปที่ 3.5 การต่อวงจรของเครื่องสั่งอาหารไร้สาย

3.3 หลักการทำงานภาครับ

3.3.1 ระบบเซิร์ฟเวอร์และฐานข้อมูล

การทำงานของเซิร์ฟเวอร์ในส่วนภาครับนั้น มีลำดับขั้นการทำงานดังโฟลว์ชาร์ต(Flowchart) ที่แสดงในรูปที่ 3.5 โดยเริ่มจากโมดูล Wi-Fi ของเครื่องเซิร์ฟเวอร์รับข้อมูลจากเครื่องรับรายการอาหาร จากนั้นประมวลผลเทียบกับฐานข้อมูล ถ้ามีข้อมูลตรงกันกับฐานข้อมูล ระบบจะดึงชื่อรายการอาหาร และราคามาแสดงผลทางจออินเตอร์เฟสของเซิร์ฟเวอร์ แต่ถ้าข้อมูลไม่ตรงกับฐานข้อมูล ระบบจะไม่แสดงรายการอาหารบนหน้าจออินเตอร์เฟส



รูปที่ 3.6 โฟลว์ชาร์ตการทำงานของเครื่องรับรายการอาหาร

3.3.2 หน้าจออินเตอร์เฟสเซิร์ฟเวอร์

ในส่วนหน้าจออินเตอร์เฟสของเซิร์ฟเวอร์ประกอบไปด้วยข้อมูลดังนี้

ข้อมูลการสั่งอาหาร

หน้าต่างนี้จะแสดงจำนวนโต๊ะอาหาร และตารางรายการอาหาร ราคา จำนวน ราคารวม เวลาที่ลูกค้าสั่งเข้ามาแต่ละโต๊ะ

การจัดรายการอาหาร

หน้าต่างนี้จะเป็นการเพิ่มรายการอาหาร และรายละเอียดต่างๆ เข้าไปในฐานข้อมูลเพื่อความสะดวกของเจ้าของร้านในการใช้งาน

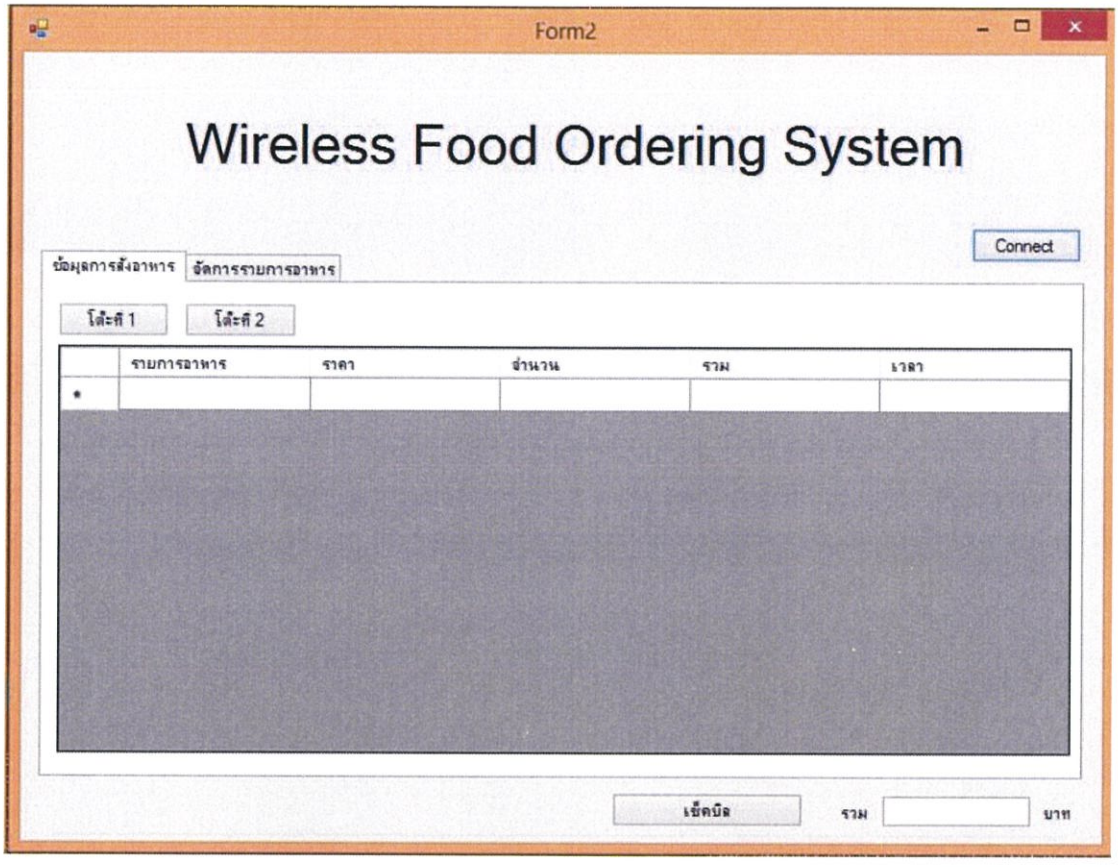
รายละเอียดอื่นๆ

ปุ่ม Connect ปุ่มนี้จะเป็นการกดเพื่อเชื่อมต่อกับฐานข้อมูลของทางร้าน และเชื่อมต่อกับระบบอินเตอร์เน็ตของทางร้านเพื่อให้สามารถส่งรายการอาหารมายังหน้าจอแสดงผลได้

ปุ่มข้อมูลการสั่งอาหาร จะแสดงหมายเลขโต๊ะอาหาร โต๊ะที่ 1 และโต๊ะที่ 2 และประกอบด้วยตารางแสดงรายการอาหารที่แต่ละโต๊ะสั่งเข้ามา พร้อมกับระบุจำนวน ราคาอาหาร และเวลาที่สั่ง

ปุ่มจัดการรายการอาหาร ทางเจ้าของร้านสามารถเข้าฟังก์ชันนี้เพื่อทำการเปลี่ยนแปลง แก้ไข และอัพเดต รายการอาหาร จนไปถึงราคาอาหาร

ปุ่มเช็คบิล เป็นการจำลองขึ้นมาว่าสามารถทำการเช็คบิลได้เมื่อลูกค้าต้องการ และถัดจากปุ่มเช็คบิลจะมีหน้าต่างรวมราคาค่าอาหารของแต่ละโต๊ะ และในฟังก์ชันของปุ่มเช็คบิล เมื่อลูกค้าชำระเงินค่าอาหารเรียบร้อยแล้วก็กดปุ่มนี้เพื่อเป็นการเคลียร์รายการอาหาร ดังแสดงในรูปที่3.7



รูปที่ 3.7 หน้าจออินเตอร์เฟสของเซิร์ฟเวอร์

บทที่ 4

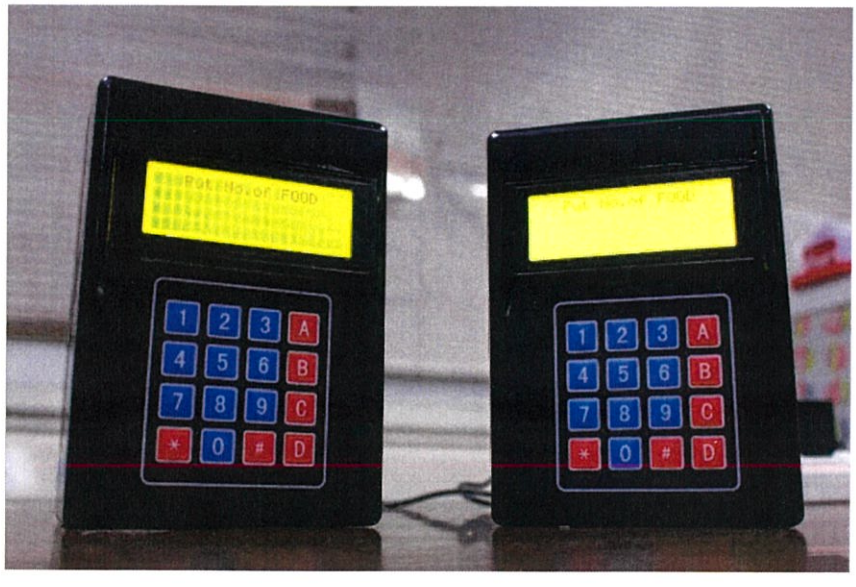
ผลการทดลอง

ผลการทดลองแบ่งออกเป็น 4 ส่วน

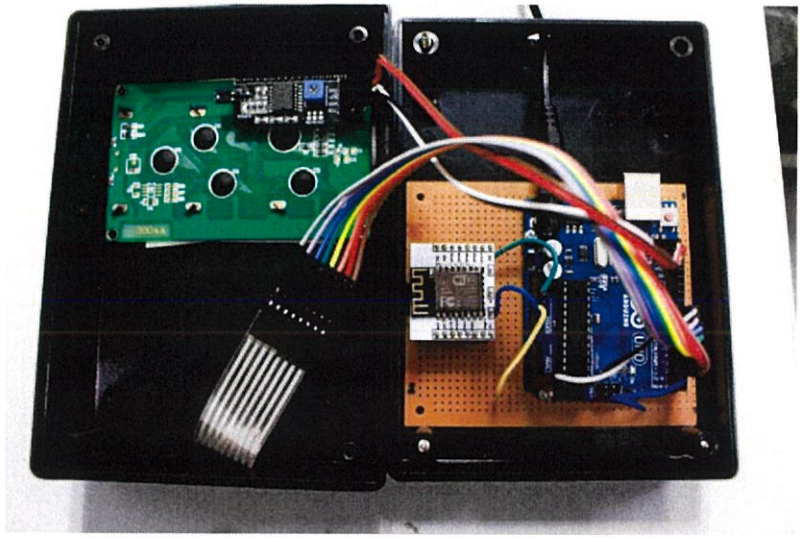
1. ผลการทดลองอุปกรณ์ภาครับ และภาคส่งสัญญาณไร้สาย
2. ตารางแสดงขั้นตอนการทำงานของเครื่องรับรายการอาหารที่โต๊ะ 1 และโต๊ะ 2
3. หน้าจอแสดงข้อมูลรายการอาหารที่สั่งมาจากโต๊ะที่ 1 และโต๊ะ 2
4. สัญญาณการส่งข้อมูล

4.1 ผลการทดลองอุปกรณ์ภาครับและภาคส่งสัญญาณไร้สาย

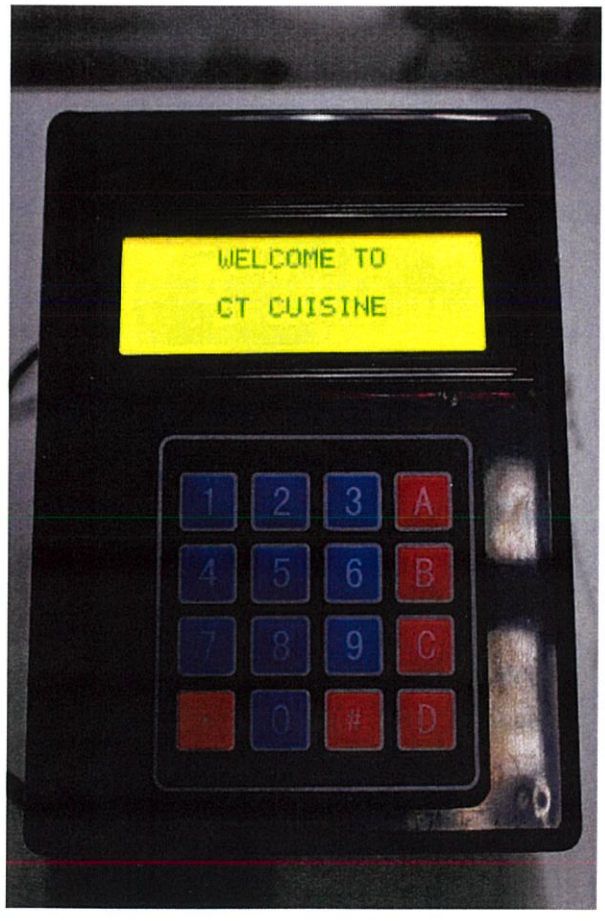
เมื่อเปิดใช้งานอุปกรณ์ภาครับและภาคส่งสัญญาณไร้สาย จะมีการแสดงข้อความต้อนรับ “WELCOME TO CT CU CUISINE” จากนั้นเครื่องรับรายการอาหารไร้สายก็จะแสดงข้อความ “Put No. of FOOD” เพื่อรอรับข้อมูลรายการอาหารจากลูกค้า เมื่อมีการสั่งรหัสรายการอาหารจากเครื่องรับรายการอาหาร ข้อมูลจะถูกส่งผ่านโมดูล Wi-Fi (ESP8266 E-12F) ไปที่เครื่องรับสัญญาณไร้สาย (Computer) หลังจากส่งข้อมูลเรียบร้อยแล้วข้อมูลรายการอาหารที่ลูกค้าสั่งจะไปแสดงที่หน้าจอคอมพิวเตอร์ของทางร้าน ผู้ดูแลรับรายการอาหารก็จะสามารถนำข้อมูลส่งต่อไปยังพ่อครัวต่อไป



รูปที่ 4.1 เครื่องรับรายการอาหารไร้สายสำหรับโต๊ะ 1 และโต๊ะ 2

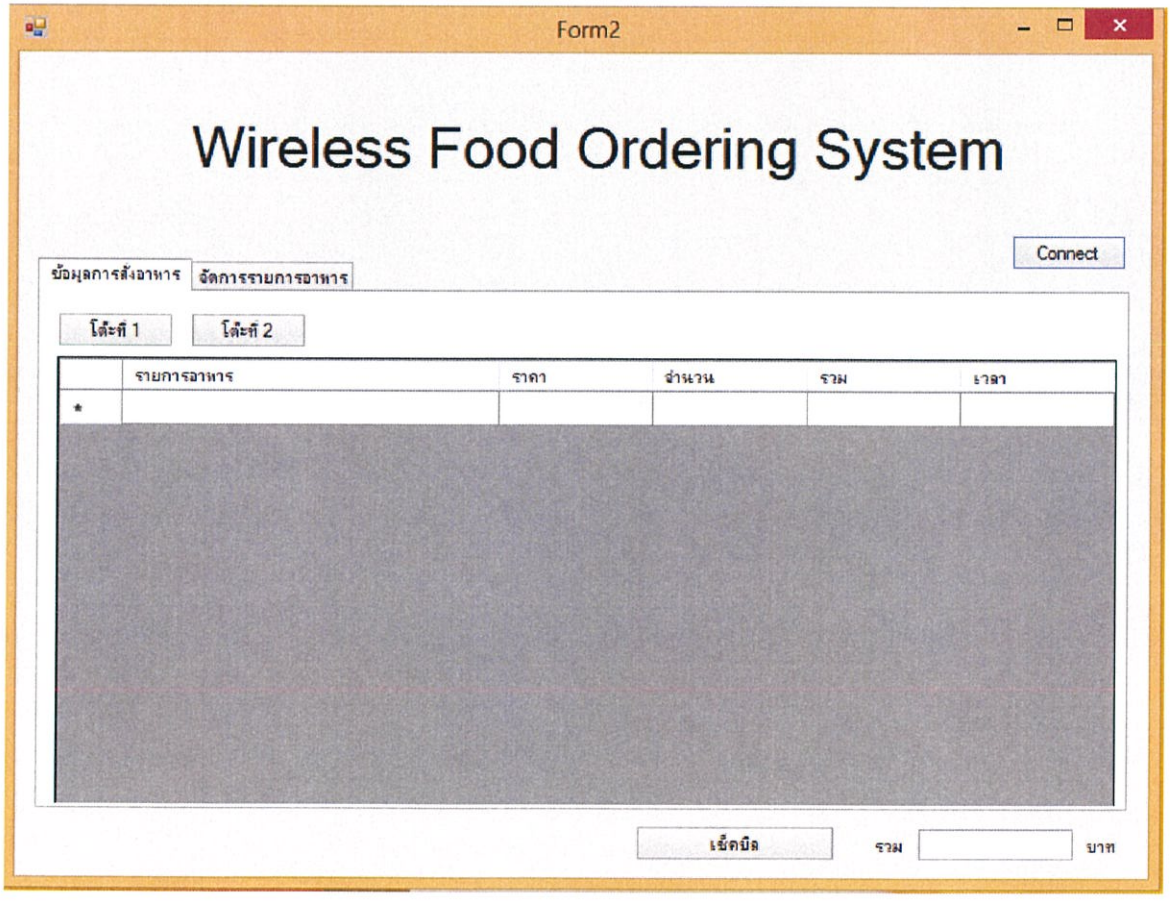


รูปที่ 4.2 วงจรภายในเครื่องรับรายการอาหารไร้สาย



รูปที่ 4.3 หน้าจอการทำงานของเครื่องรับอาหารไร้สาย แสดงข้อความต้อนรับ

รูปที่ 4.4 หน้าจอการทำงานของเครื่องรับอาหารไร้สาย พร้อมที่รับออเดอร์จากลูกค้า

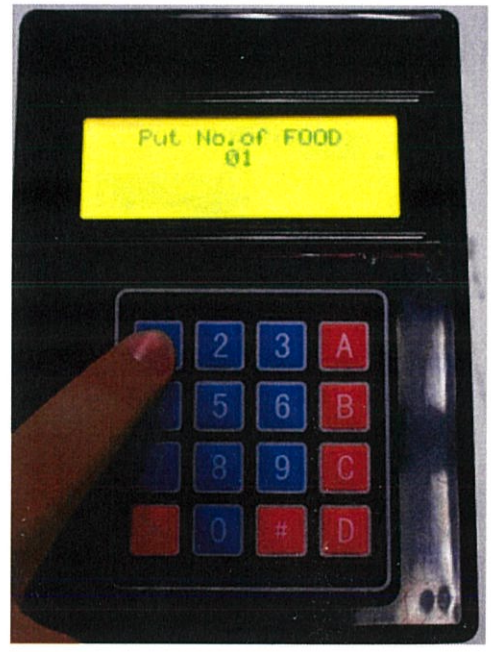


รูปที่ 4.5 หน้าจอแสดงรายการอาหารในส่วนของผู้รับออเดอร์

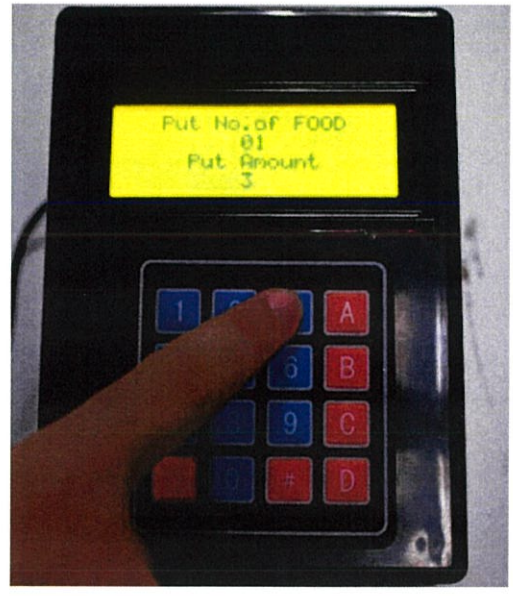
4.2 ผลการทดลองการสั่งงานและแสดงผลที่หน้าจอแอลซีดี

การกดหมายเลขรหัสายการอาหารเพื่อสั่งอาหาร มีขั้นตอนในการใช้งาน และจะมีการแสดงผ่านหน้าจอแอลซีดี ดังนี้

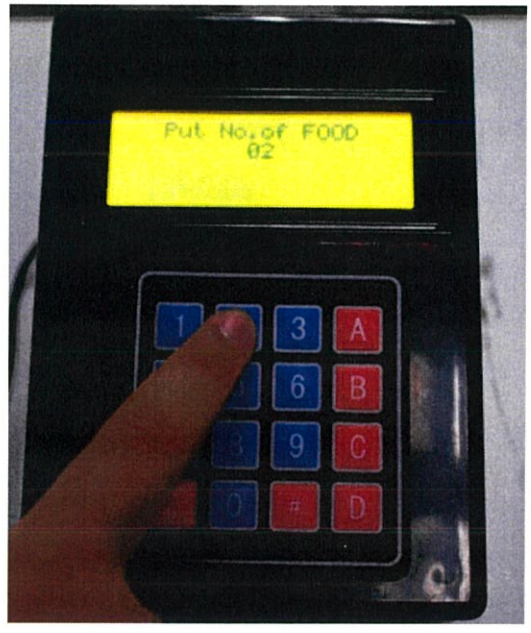
ตารางที่ 4.1 ขั้นตอนการทำงานของเครื่องรับรายการอาหารที่โต๊ะ 1

ขั้นตอนการสั่งรายการอาหาร	ขั้นตอนการทำงานของเครื่องรับรายการอาหาร
- หน้าจอจะขึ้นข้อความ Put No. of FOOD จากนั้นให้ลูกค้าใส่รหัสรายการอาหารที่ต้องการสั่ง - โดยรายการอาหารจะแทนด้วยรหัส 2 หลัก	
- กดตัวอักษร A เพื่อยืนยันรายการอาหารที่สั่ง (และถ้าต้องการเปลี่ยนแปลงรายการก็ทำการกดหมายเลขตัวใดก็ได้อีก 1 ครั้ง ก็จะสามารถใส่รหัสรายการอาหารอันใหม่ได้)	

ตารางที่ 4.1 (ต่อ) ขั้นตอนการทำงานของเครื่องรับรายการอาหารที่โต๊ะ 1

- หน้าจอจะขึ้นข้อความให้ลูกค้าระบุจำนวนที่ต้องการสั่ง (ถ้าต้องการเปลี่ยนแปลงจำนวนที่สั่งให้ทำการกดหมายเลขใดก็ได้จนกว่าหน้าจอจะว่างให้ทำการระบุจำนวนใหม่อีกครั้ง) - จากนั้นกดตัวอักษร A รายการอาหารก็จะถูกส่งไป ยังหน้าจอแสดงผลที่ฝั่งผู้รับออเดอร์	
- หน้าจอจะแสดงข้อความ Send OK แสดงว่ารายการอาหารถูกส่งเรียบร้อยแล้ว	

ตารางที่ 4.2 ขั้นตอนการทำงานของเครื่องรับรายการอาหารที่โต๊ะ 2

ขั้นตอนการสั่งรายการอาหาร	ขั้นตอนการทำงานของเครื่องรับรายการอาหาร
- หน้าจอจะขึ้นข้อความ Put No. of FOOD จากนั้นให้ลูกค้าใส่รหัสรายการอาหารที่ต้องการสั่ง - โดยรายการอาหารจะแทนด้วยรหัส 2 หลัก	
- กดตัวอักษร A เพื่อยืนยันรายการอาหารที่สั่ง (และถ้าต้องการเปลี่ยนแปลงรายการก็ทำการกดหมายเลขตัวใดก็ได้อีก 1 ครั้ง ก็จะสามารถใส่รหัสรายการอาหารอันใหม่ได้)	

ตารางที่ 4.2 (ต่อ) ขั้นตอนการทำงานของเครื่องรับรายการอาหารที่โต๊ะ 2

- หน้าจอจะขึ้นข้อความให้ลูกค้าระบุจำนวนที่ต้องการสั่ง (ถ้าต้องการเปลี่ยนแปลงจำนวนที่สั่งให้ทำการกดหมายเลขใดก็ได้จนกว่าหน้าจอจะว่างให้ทำการระบุจำนวนใหม่อีกครั้ง) - จากนั้นกดตัวอักษร A รายการอาหารก็จะถูกส่งไป ยังหน้าจอแสดงผลที่ฝั่งผู้รับออเดอร์	
- หน้าจอจะแสดงข้อความ Send OK แสดงว่ารายการอาหารถูกส่งเรียบร้อยแล้ว	

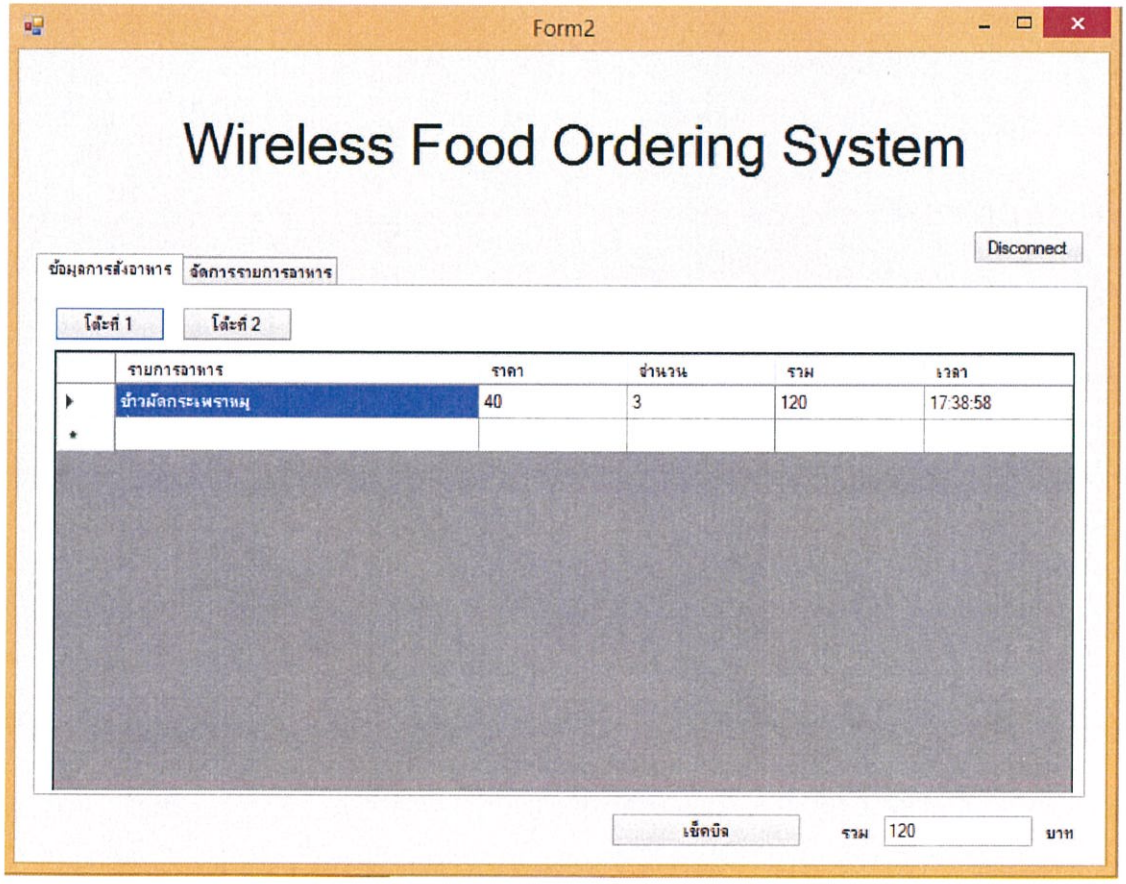
ผลการทดลองจากตารางที่ 4.1 และตารางที่ 4.2

จากการทดลองพบว่า เมื่อทำการส่งข้อมูลรายการอาหารจากเครื่องรับรายการอาหารทั้ง 2 โต๊ะ รายการอาหารที่ลูกค้าสั่งก็จะปรากฏบนหน้าจอแสดงผลที่ฝั่งผู้รอรับออเดอร์ ดังแสดงในรูปที่ 4.6 และรูปที่ 4.7 และสามารถตรวจสอบได้ว่าเป็นรายการอาหารจากทางโต๊ะใด ซึ่งในการทดลองส่วนนี้เป็นการเชื่อมต่อสัญญาณอินเตอร์เน็ตกับเครื่องรับรายการอาหารเข้ากับเร้าเตอร์ที่จัดเตรียมไว้ และทำให้พบว่าในการส่งข้อมูลถ้า ณ เวลานั้นสัญญาณอินเตอร์เน็ตมีสัญญาณดีก็จะส่งข้อมูลได้อย่างรวดเร็วแบบ Real Time แต่ถ้าสัญญาณอินเตอร์เน็ตมีสัญญาณดรอปลงก็จะทำให้การส่งเกิดการดีเลย์เล็กน้อย ซึ่งขึ้นอยู่แล้วแต่ช่วงเวลาที่มีการเข้าเชื่อมต่ออินเตอร์เน็ตจากหลายๆ อุปกรณ์พร้อมๆ กัน

4.3 ผลการทดลองแสดงข้อมูลรายการอาหารผ่านทางหน้าจอคอมพิวเตอร์

4.3.1 การแสดงรายการอาหารที่สั่งจากโต๊ะ 1

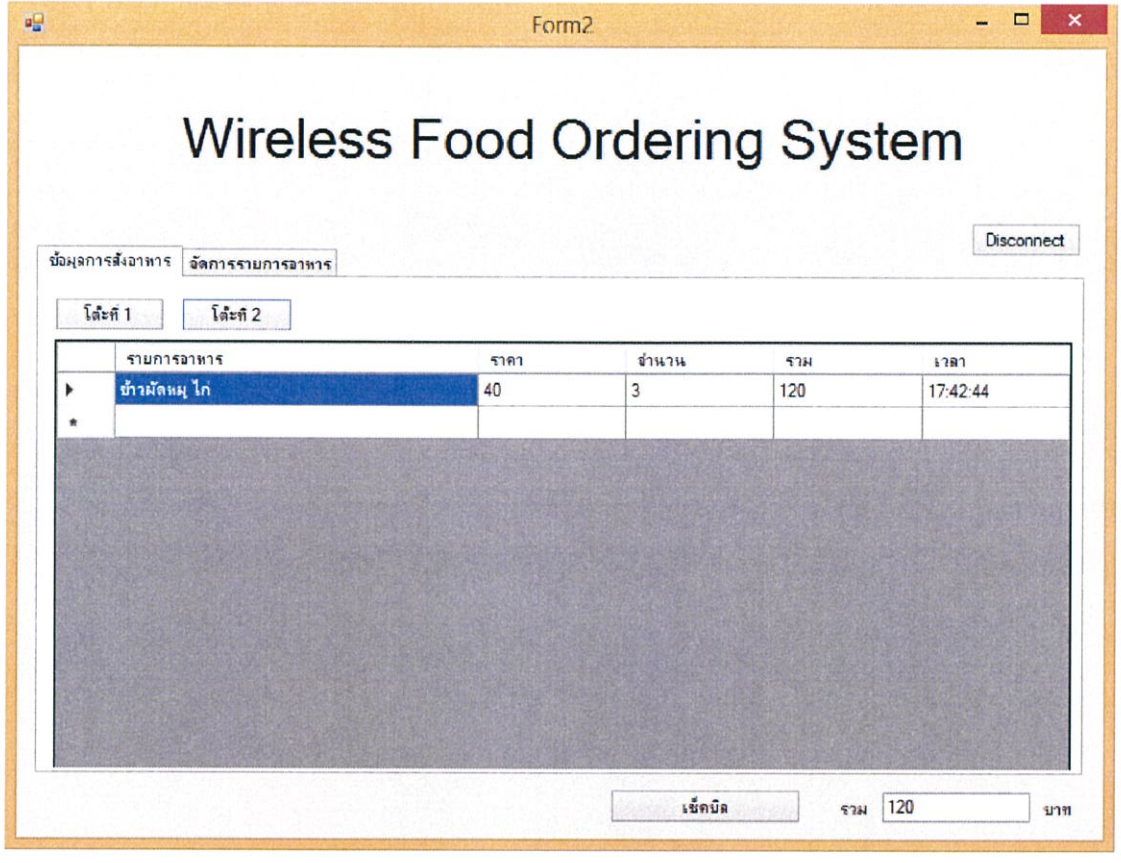
ลูกค้าโต๊ะที่ 1 ทำการสั่งอาหารเสร็จแล้ว รายการอาหารที่ลูกค้าสั่งก็จะปรากฏที่หน้าต่างดังแสดงในรูปที่ 4.6 รายละเอียดก็จะประกอบด้วย ชื่ออาหาร ราคา จำนวนที่สั่ง และเวลาที่สั่ง



รูปที่ 4.6 รายละเอียดของรายการอาหารที่ลูกค้าสั่งจากโต๊ะ 1 จะปรากฏบนหน้าจอ

4.3.2 การแสดงรายการอาหารที่สั่งจากโต๊ะ 2

ลูกค้าโต๊ะที่ 2 ทำการสั่งอาหารเสร็จแล้ว รายการอาหารที่ลูกค้าสั่งก็จะปรากฏที่หน้าต่างดังแสดงในรูปที่ 4.7 รายละเอียดก็จะประกอบด้วย ชื่ออาหาร ราคา จำนวนที่สั่ง และเวลาที่สั่ง



รูปที่ 4.7 รายละเอียดของรายการอาหารที่ลูกค้าสั่งจากโต๊ะ 2 จะปรากฏบนหน้าจอ

ผลการทดลองจากรูปที่ 4.6 และรูปที่ 4.7

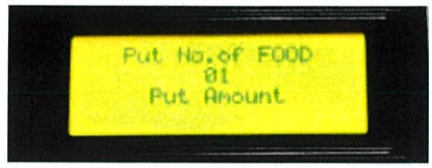
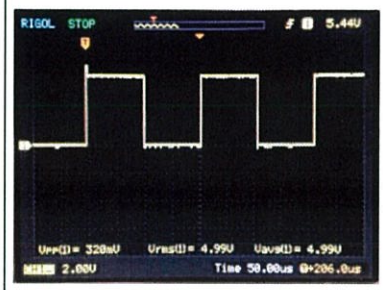
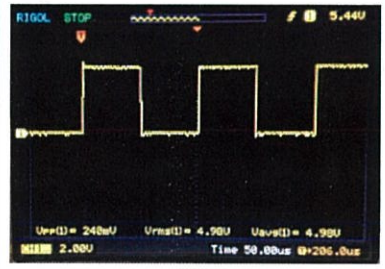
จากการทดลองพบว่า เมื่อผู้รอรับออเดอร์จากลูกค้าทำการกด Connect ที่หน้าจอแสดงผลก็จะทำให้ลูกค้าสามารถทำการสั่งรายการอากรผ่านทางเครื่องรับรายการอาหาร และรายการอาหารที่ลูกค้าสั่งก็จะปรากฏบนหน้าจอแสดงผลดังรูปที่ 4.6 และรูปที่ 4.7 ซึ่งฟังก์ชันต่างๆ บนหน้าจอแสดงผลสามารถใช้งานและตอนสนองต่อความต้องการของผู้ใช้บริการได้เป็นอย่างดี และมีความสะดวกเข้าใจง่าย

4.4 ผลการทดลองการส่งสัญญาณข้อมูล

4.4.1 แสดงสัญญาณในการส่งข้อมูล

สัญญาณในการส่งข้อมูลของเครื่องรับรายการอาหาร โดยกดหมายเลขตามลำดับขั้นตอนการสั่งอาหาร ดังนี้

ตารางที่ 4.3 สัญญาณการส่งข้อมูล

ขั้นตอนการกด	ภาพการกดในแต่ละขั้นตอน	สัญญาณในแต่ละขั้นตอน
กด A เพื่อยืนยันรหัสของรายการอาหาร		
กด A เพื่อทำการส่งรายการอาหร		

ผลการทดลองการส่งสัญญาณ

จากการทดลองโดยใช้เครื่องออสซิลโลสโคป (Oscilloscope) ในการวัดค่าความต่างศักย์ไฟฟ้าเฉลี่ย ของขาไมโครคอนโทรลเลอร์ และสังเกตคลื่นสัญญาณในแต่ละขาว่าเกิดการผิดเพี้ยนหรือไม่ โดยพบว่าค่าความต่างศักย์ไฟฟ้าเฉลี่ย ที่วัดออกมาได้มีค่าใกล้เคียงกัน และคลื่นสัญญาณที่ได้จากการกดแต่ละครั้งจะมีสัญญาณรบกวนเล็กน้อย โดยใช้ช่วงเวลาในการวัดที่แตกต่างกัน

บทที่ 5
สรุปผลการดำเนินงาน

5.1 สรุปผล

ปริญญานิพนธ์นี้ได้ทำการออกแบบและสร้างชุดอุปกรณ์ระบบสั่งอาหารไร้สาย เพื่อความสะดวกต่อลูกค้าที่มารับบริการภายในร้านอาหาร การดำเนินงานของระบบสั่งอาหารไร้สายแบ่งออกได้เป็น 2 ส่วนคือ

1. ส่วนการทำงานของอุปกรณ์ภาคส่งสัญญาณไร้สาย หรือเครื่องรับรายการอาหาร โดยการทำงานเริ่มจากการกดรหัสรายการอาหารทางแป้นพิมพ์จากนั้น รหัสจะถูกประมวลผลด้วยไมโครคอนโทรลเลอร์ Atmega328P พร้อมทั้งมีการแสดงสถานะการสั่งรายการอาหารบนหน้าจอ LCD และข้อมูลรายการอาหารจะถูกส่งด้วย Wi-Fi โมดูลไปยังส่วนของ Server

2. ส่วนการทำงานของอุปกรณ์ภาครับสัญญาณไร้สาย หรือเครื่องคอมพิวเตอร์ โดยเริ่มจากการรับข้อมูลจาก Wi-Fi โมดูลมาทำการประมวลผล และดึงข้อมูลรายการอาหารจากฐานข้อมูลออกมาแสดงผลที่หน้าจออินเตอร์เฟส

5.2 ปัญหาและข้อจำกัดของการดำเนินงาน

1. ปัญหาการอัพโหลดข้อมูลเพื่อเข้าไปยัง Wi-Fi โมดูล เนื่องจากในการอัพโหลดต้องถอดชิบออกก่อนทุกครั้ง จึงเกิดความไม่สะดวกในการอัพโหลด

2. จากการส่งสัญญาณไร้สายของอุปกรณ์ภาคส่งสัญญาณ ในบางครั้งเกิดการรบกวนจากสัญญาณภายนอกขึ้น ทำให้ส่งสัญญาณได้ต่ำกว่าความสามารถที่แท้จริง

3. เนื่องจากบอร์ดอาดูโนที่ใช้มีราคาถูก ดังนั้นจึงมีขาทั้งหมด 28 ขา และมีหน่วยความจำเพียง 32 กิโลไบต์ ทำให้การใช้งานเพื่อนำไปพัฒนาต่อเป็นไปได้ยาก

4. การสื่อสารระหว่างภาคส่งและภาครับยังเป็นการสื่อสารแบบทิศทางเดียว คือ เครื่องรับรายการอาหารส่งข้อมูลไปยังส่วนของ Server แต่ไม่มีการตอบกลับสถานะของข้อมูลที่ได้รับจากการส่ง

5.3 ปัญหาที่พบและแนวทางการแก้ไข

1. ปัญหาการส่งข้อมูล

จากการทดลองการสั่งอาหารผ่านแป้นพิมพ์พบว่าบางครั้งการรับส่งข้อมูลมีความล่าช้าเกิดขึ้นเนื่องจากความแออัดในบางแชนแนล Wi-Fi จนทำให้เครือข่ายอินเทอร์เน็ตมีอาการช้า

วิธีการแก้ไข

สามารถแก้ไขปัญหาดังกล่าวโดยการใช้เร้าเตอร์ Wi-Fi ที่เป็นส่วนตัว ทำให้สามารถส่งข้อมูลได้อย่างรวดเร็วและไร้ปัญหา

2. ปัญหาการดึงข้อมูล

ในการดึงข้อมูลจากฐานข้อมูลมาใช้ในบางเครื่องคอมพิวเตอร์ ไม่สามารถดึงข้อมูลรายการอาหารมาแสดงออกที่หน้าจออินเตอร์เฟสได้ทั้งที่ทำการดึงข้อมูลจากฐานข้อมูลเดียวกัน

วิธีการแก้ไข

พบว่าปัญหาที่เกิดขึ้นเนื่องจากคอมพิวเตอร์บางเครื่องมีการปรับเปลี่ยนพอร์ตข้อมูลจากค่าเริ่มต้น (Port:80) จึงเข้าไปแก้ให้ค่าพอร์ตกลับมาตรงกันเพียงเท่านี้ก็สามารถดึงข้อมูลได้ตามปกติ

3. ปัญหาไลบรารี่ไม่สามารถนำมาใช้งานได้

ปัญหาเกี่ยวกับไลบรารี่ที่พบเจอในการทำเครื่องสั่งอาหารไร้สาย คือไลบรารี่ของตัวโมดูล Wi-Fi (ESP8266 E-12F) ที่ทำการดาวน์โหลดมาจากเว็บไซต์ เมื่อนำมาใช้ทดลองจริงทำการอัพลงในโมดูล (ESP8266 E-12F) แล้วไม่สามารถเชื่อมต่อกับตัวเร้าเตอร์ที่ปล่อยสัญญาณอินเตอร์เน็ตได้ ทำให้ไม่สามารถส่งข้อมูลจากเครื่องสั่งอาหารไปยังเครื่องคอมพิวเตอร์ได้

วิธีการแก้ไข

พบปัญหาดังกล่าวคือ ตัวไลบรารีของตัวโมดูล Wi-Fi (ESP8266 E-12F) จึงทำการเขียนไลบรารี่ขึ้นมาใหม่สำหรับใช้งานกับตัวโมดูล Wi-Fi ซึ่งทำให้ปัญหานี้หมดไป และทำการทดลองได้ตามปกติ

4. ปัญหาการออกแบบหน้าจออินเตอร์เฟส

ในการสร้างแบบฟอร์มหน้าจออินเตอร์เฟสผ่านโปรแกรม Visual Basic โดยเมื่อทดลองRun กราฟฟิกที่ออกแบบมา พบว่าเมื่อขยายหน้าจอให้มีขนาดเต็มหน้าจอ ปุ่ม Button และสเกลต่างๆ ไม่ขยายตามออกมาตามที่ควรจะเป็น

วิธีการแก้ไข

ในการแก้ไขพบว่าต้องดาวน์โหลดไลบรารีเพิ่มเติมเข้ามาในโปรแกรม และต้องเขียนโค้ดเพิ่มเติมให้ปุ่ม Button ต่างๆ ขยายตามเมื่อเปิดหน้าจออินเตอร์เฟสแบบ Maximum Size

5. ปัญหาการสื่อสารแบบทิศทางเดียว

การรับส่งข้อมูลระหว่างเครื่องรับรายการอาหารกับส่วนของ Server เมื่อทำการส่งข้อมูลรายการอาหารแล้วพบว่าส่วนของ Server ไม่มีการตอบกลับเพื่อแสดงสถานะเมื่อได้รับข้อมูลจากเครื่องรับรายการอาหาร และปุ่มยกเลิกรายการอาหารเมื่อลูกค้ากดผิด หรือต้องการเปลี่ยนแปลงรายการที่สั่ง

วิธีการแก้ปัญหา

สร้างเงื่อนไขของการแจ้งเตือนตอบกลับเพื่อให้ลูกค้ารับทราบถึงสถานะของรายการที่สั่ง และเพิ่ม

เงื่อนไขเพื่อให้สามารถกดปุ่มยกเลิกรายการอาหาร โดยการเขียนโปรแกรมเพิ่มเงื่อนไขเข้าไปในส่วนของเครื่องรับรายการอาหาร และส่วนของ Server

5.4 แนวทางในการพัฒนาต่อ

ปริญญานิพนธ์นี้สามารถนำไปพัฒนาต่อได้อย่างหลากหลาย เนื่องจากได้ทำการพัฒนาทั้งฮาร์ดแวร์ และซอฟต์แวร์ควบคุมการทำงานของตัวรับส่งสัญญาณไร้สาย โดยสามารถพัฒนาจากหน้าจอแอลซีดีเป็นหน้าจอทัชสกรีน (Touch Screen) เพื่อสะดวกต่อผู้ใช้งาน หรือสามารถทำการพัฒนาเป็นแอพพลิเคชั่นในโทรศัพท์มือถือ ให้รองรับกับการสั่งรายการอาหารผ่านทางโทรศัพท์มือถือ และสามารถเพิ่มรูปภาพอาหารได้เพื่อเพิ่มความสวยงาม ทำให้ผู้ใช้งานเห็นรูปภาพสีสันของอาหารได้หลากหลายขึ้น เพื่อเพิ่มความสะดวกสบายต่อผู้ใช้งานมากที่สุด

เอกสารอ้างอิง

[1] K. J. Patel, et al., **"PDA-based Wireless Food Ordering System for Hospitality Industry - A Case Study of Box Hill Institute"** in Wireless Telecommunications Symposium 2007, Pomona, CA, 2007, pp. 1-8.

[2] J. Purnama, et al.**"Application of Order Management System in Restaurants"**, Seminar Nasional Aplikasi Teknologi Informasi 2007, Yogyakarta, 16 June 2007 (SNATI 2007) ISSN: 1907-5022.

[3] Apache Software Foundation. Jmeter. http://jakarta.apache.org/jmeter.

[4] S. Misra et al. (eds.), **"Guide to Wireless Sensor Networks, Computer Communications and Networks"**, Springer-Verlag London Limited., 2009.

[5] DfRobot (2015). **LCD Keypad Shield for Arduino**. Retrieved December 5, 2015 fromhttp://www.dfrobot.com/index.php?route=product/product&product_id=51.

[6] K. Kamarudin, et al., **"The Application of Wireless Food Ordering System,"** MASAUM Journal of Computing, vol. 1, pp. 178-184, 2009.

[7] F. M. Cady, **"Microcontrollers and Microcomputers Principles of Software and Hardware Engineering"**, Oxford University Press, Inc., 2009.

[8] M. Firdouse Ali Khan, Swapna,**"Design and Implementation of Ordering System for Restaurants"**, in International Journal of Engineering Research&Technology (IJERT), Vol. 1, Issue 10, December- 2012.

[9] Ayob J., Mohd. HelmyA.Wahab, Khairunnisa K., M. IzwanAyob, M. AfifAyob, M. ErdiAyob, **"The Application of Wireless Food Ordering System"**, in MASAUM Journal of Computing, Volume 1 Issue 2, September 2009.

[10] Holger Karl ,Andreas Willing. **"Protocol and Architectures for Wireless Sensor Networks"**, John Wiley and Sons., 2005.

เอกสารอ้างอิง (ต่อ)

[11] M. Simon, "**Programming Arduino: Getting Started With Sketches**", McGraw Hil., 2012.

[12] M. Margolis, "**Arduino Cookbook**", O'Reilly Media Inc., 2011.

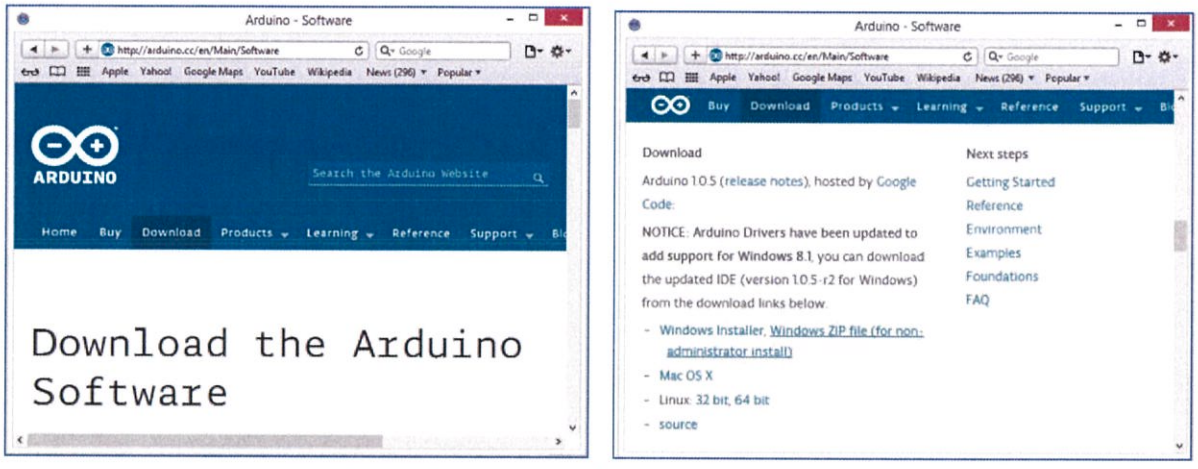
[13] XU Hongzhen, et al,"**Wireless Food Ordering System Based on Webservices**", Second international Conference on Intelligent Computation Technology and Automation, 2009.

ภาคผนวก

ภาคผนวก ก.
คู่มือการติดตั้ง

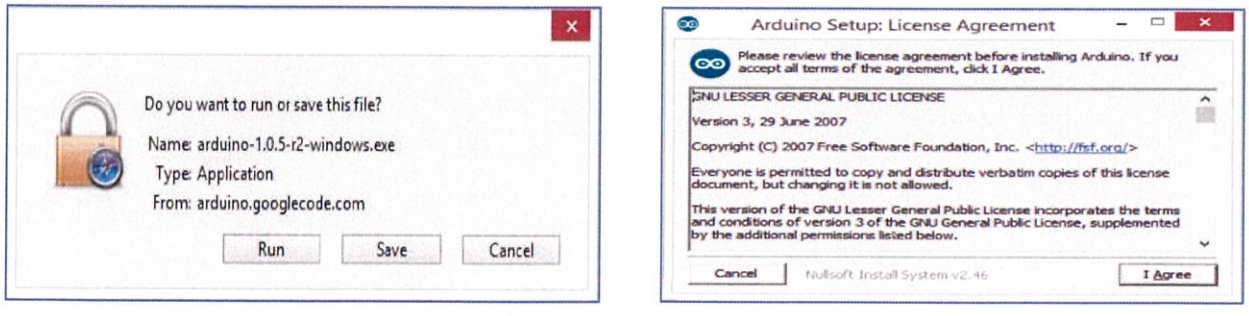
การติดตั้ง โปรแกรม Arduino

1. เปิดเว็บไซต์ http://arduino.cc/en/Main/Software และคลิกที่เมนู Download เลือกรายการ Windows Installer ในกรณีที่เป็น Administrator หรือเลือกรายการ Windows ZIP file เพื่อที่จะดาวน์โหลด zip ไฟล์



รูปที่ ก.1 ขั้นตอนการติดตั้งโปรแกรม Arduino

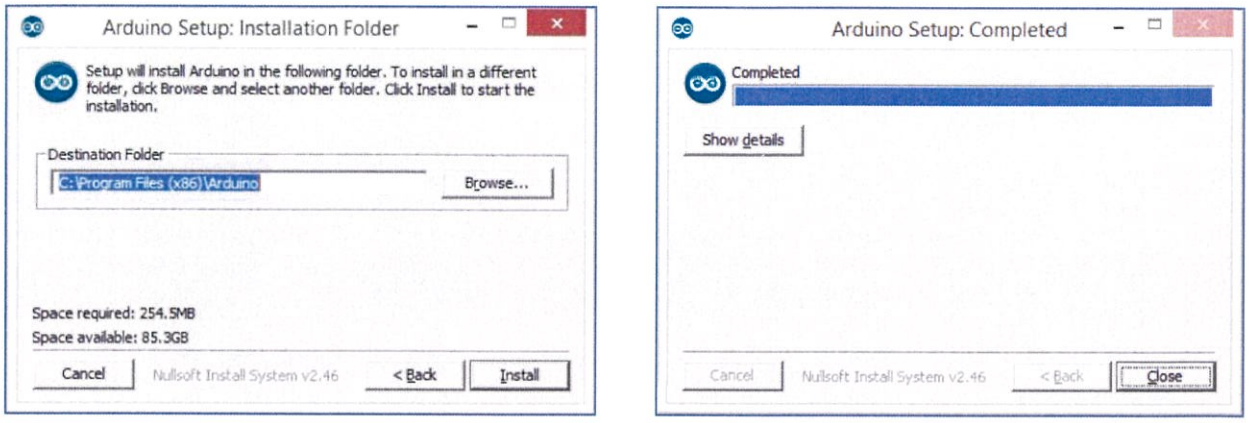
2. ในกรณีที่เลือกรายการ Windows Installer จะทำการดาวน์โหลด EXE ไฟล์ (arduino-1.0.5-r2-windows.exe) และ เมื่อทำการดาวน์โหลดเสร็จ จะขึ้นหน้าจอถามว่าจะ run โปรแกรม หรือ save โปรแกรม ให้เลือก save โปรแกรมเพื่อนำโปรแกรมไปใช้ติดตั้งในภายหลัง หลังจากนั้นให้ทำการดับเบิลคลิกโปรแกรมที่เก็บไว้ จะปรากฏหน้าจอแจ้งข้อมูลเรื่องลิขสิทธิ์ ให้คลิก I Agree



รูปที่ ก.2 ขั้นตอนการติดตั้งโปรแกรม Arduino (ต่อ)

หลังจากนั้นจะปรากฏหน้าจอสอบถามตำแหน่งโฟลเดอร์ที่จะติดตั้งโปรแกรม ค่าปริยายคือ C:\Program Files (x86)\Arduino ในกรณีที่ไม่ต้องการใช้ค่าปริยายให้กดปุ่ม Browse แล้วเลือก

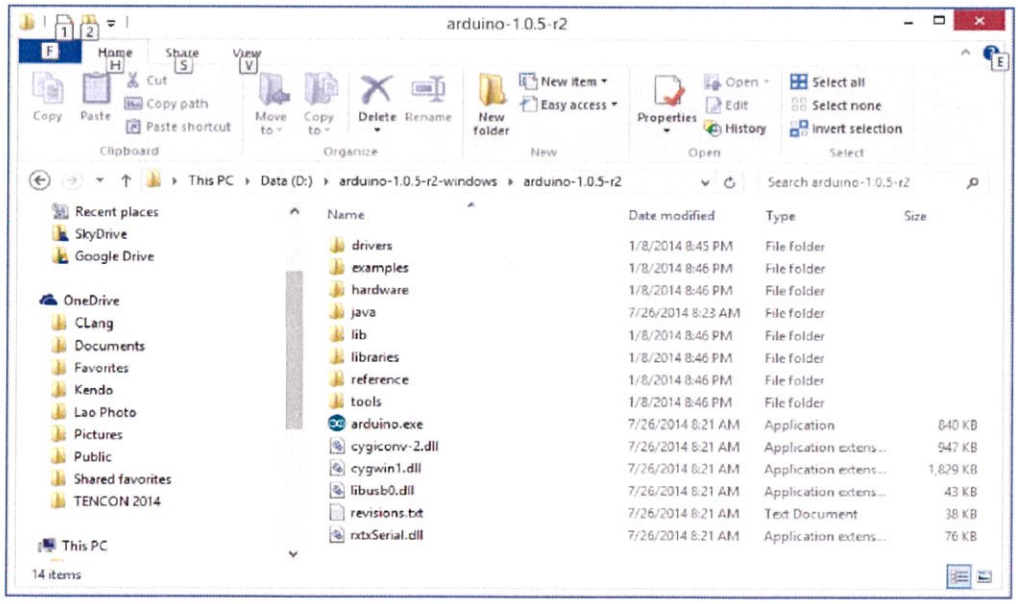
ตำแหน่งโฟลเดอร์ที่ต้องการ แล้วกดปุ่ม Install เมื่อโปรแกรมทำการติดตั้งเสร็จจะปรากฏ shortcut สัญลักษณ์ของ Arduino ที่หน้าจอ Desktop



รูปที่ ก.3 ขั้นตอนการติดตั้งโปรแกรม Arduino (ต่อ)

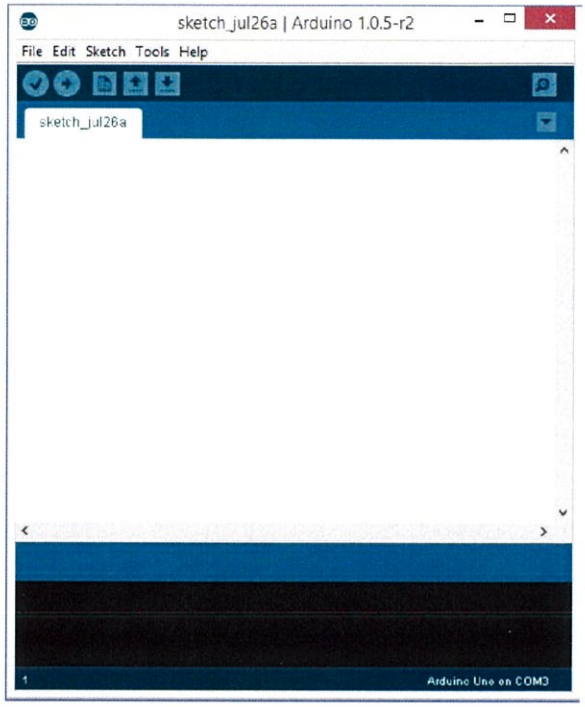
3. ในกรณีที่เลือก ดาวน์โหลด zip ไฟล์ ให้ทำการ unzip ไฟล์ ซึ่งจะปรากฏหน้าจอขึ้นมาถามถึงตำแหน่งโฟลเดอร์ที่จะติดตั้ง (ค่าโดยปริยายคือ Arduino-1.0.5.r2-windows) ให้เลือกตำแหน่งโฟลเดอร์ C:\arduino หรือ D:\arduino หรือใช้ค่าโดย ปริยาย แล้วกดปุ่ม Extract

4. ทำการตรวจสอบโฟลเดอร์ที่ได้ทำการติดตั้งไฟล์ จะปรากฏไฟล์ชื่อ arduino.exe ซึ่งเป็นโปรแกรมที่จะใช้งาน



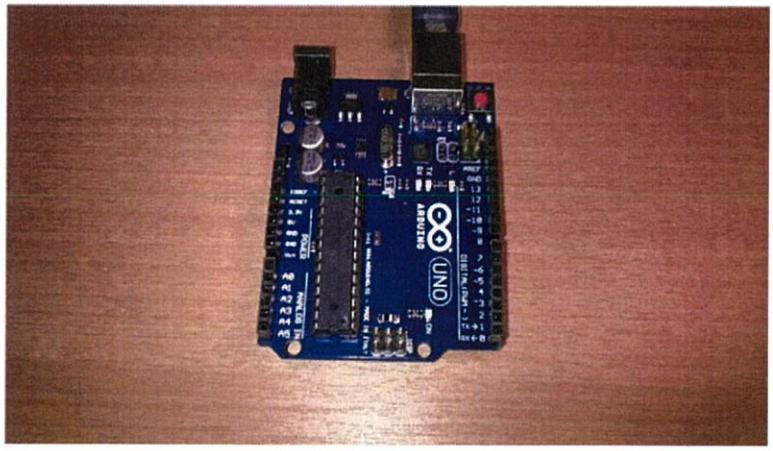
รูปที่ ก.4 ขั้นตอนการติดตั้งโปรแกรม Arduino (ต่อ)

5. ทำการรันโปรแกรม arduino.exe จะปรากฏหน้าจอดังนี้

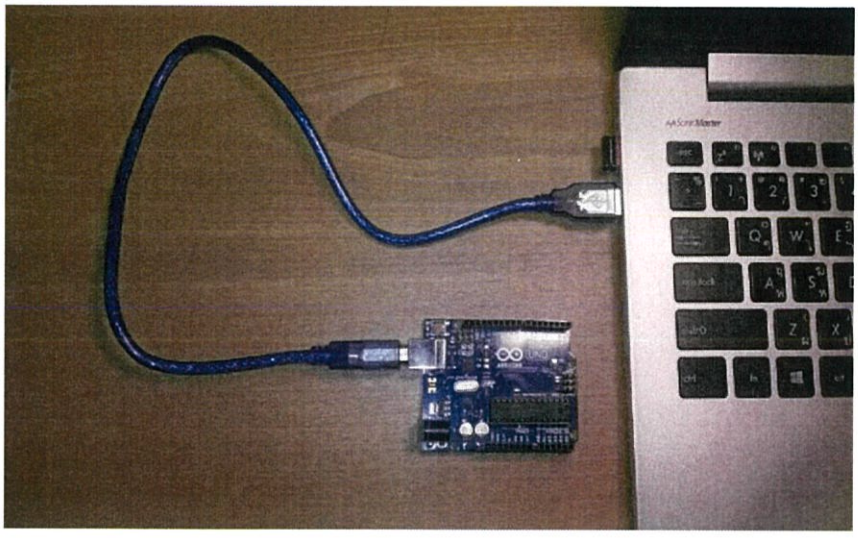


รูปที่ ก.5 ขั้นตอนการติดตั้งโปรแกรม Arduino (ต่อ)

6. ให้ทำการเสียบสาย USB เข้ากับบอร์ด Arduino และคอมพิวเตอร์

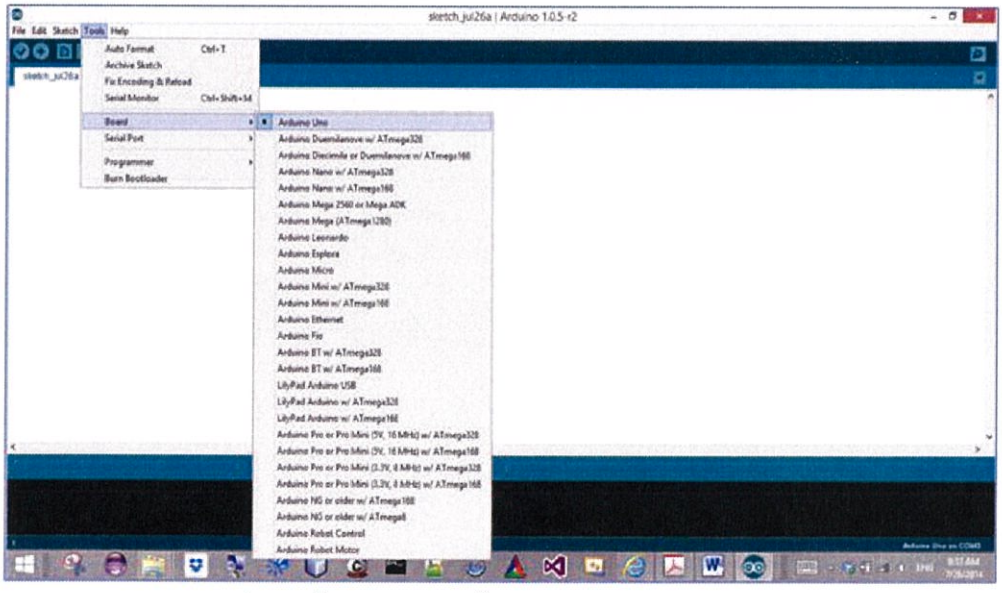


รูปที่ ก.6 ขั้นตอนการติดตั้งโปรแกรม Arduino (ต่อ)

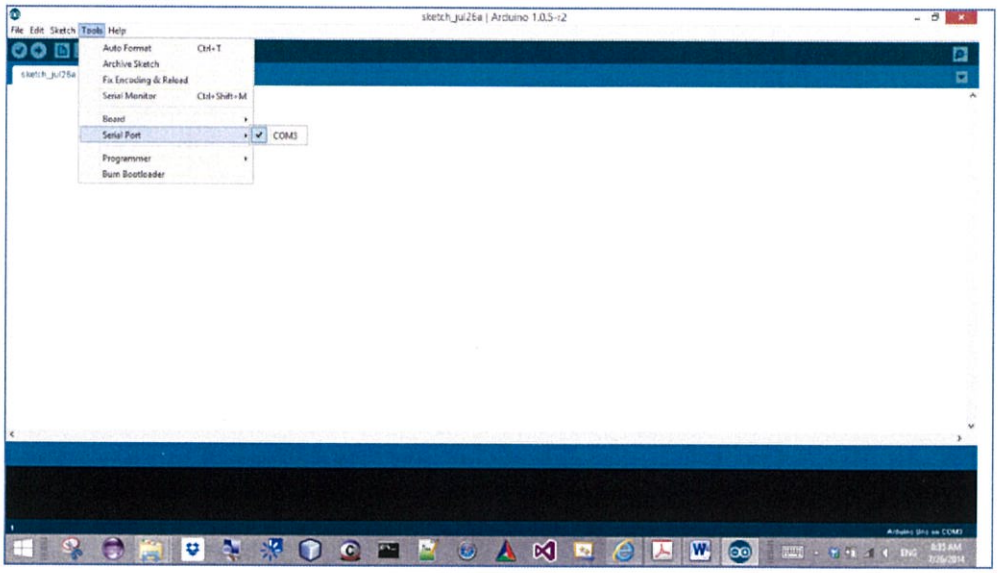


รูปที่ ก.7 ขั้นตอนการติดตั้งโปรแกรม Arduino (ต่อ)

7. ที่หน้าจอเลือกรุ่นของบอร์ด Arduino ตามที่ต้องการโดยการเลือกเมนู Tool -> Board -> ??? สำ หรับการทดลองให้เลือก เมนู Tool -> Board -> Arduino Uno หลังจากนั้นให้ทำ การเซ็ท Serial Port ที่จะใช้ในการติดต่อสื่อสาร โดยการเลือกเมนู Tool -> Serial Port -> ??? ในการทดลอง Com Port จะขึ้นอยู่กับคอมพิวเตอร์ที่ใช้ซึ่งแต่ละเครื่องอาจไม่ตรงกัน



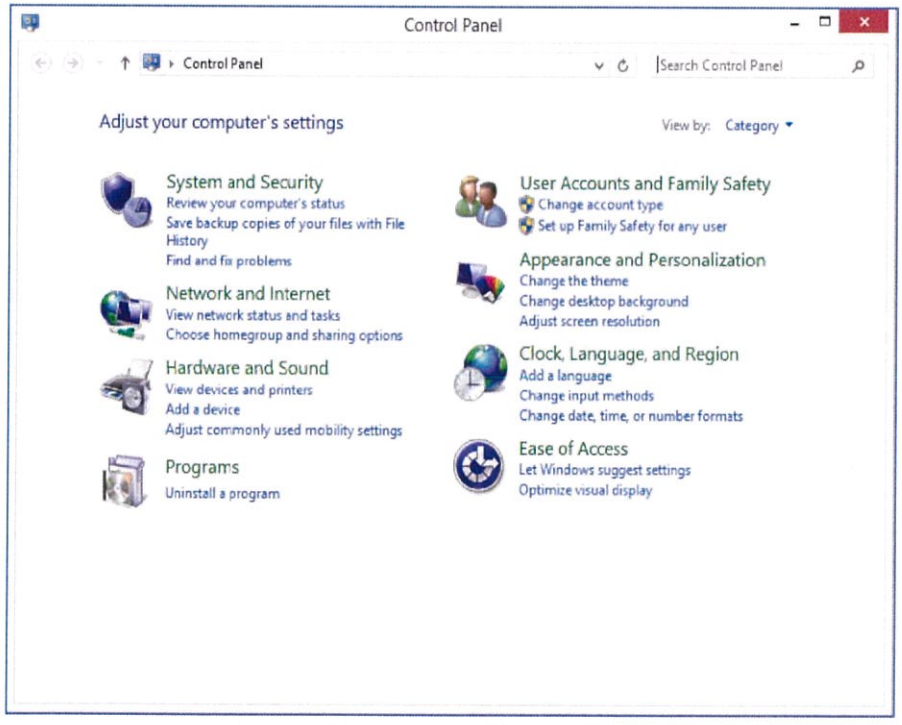
รูปที่ ก.8 ขั้นตอนการติดตั้งโปรแกรม Arduino (ต่อ)



รูปที่ ก.9 ขั้นตอนการติดตั้งโปรแกรม Arduino (ต่อ)

8. ทำการลงไดรว์เวอร์บอร์ด Arduino

a) เลือก Control Panel

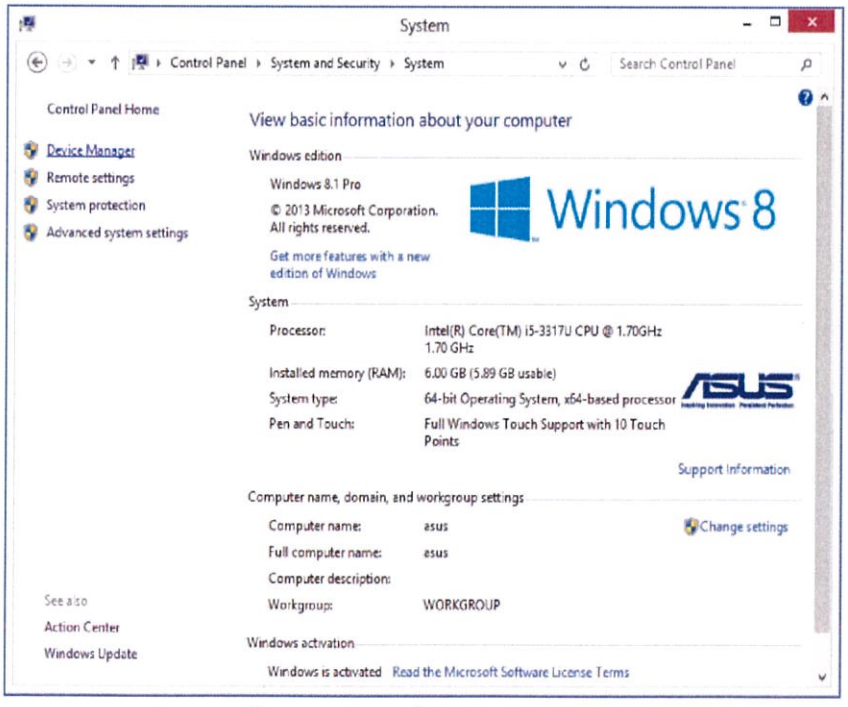


รูปที่ ก.10 ขั้นตอนการติดตั้ง Device Driver

b) เลือก System and Security

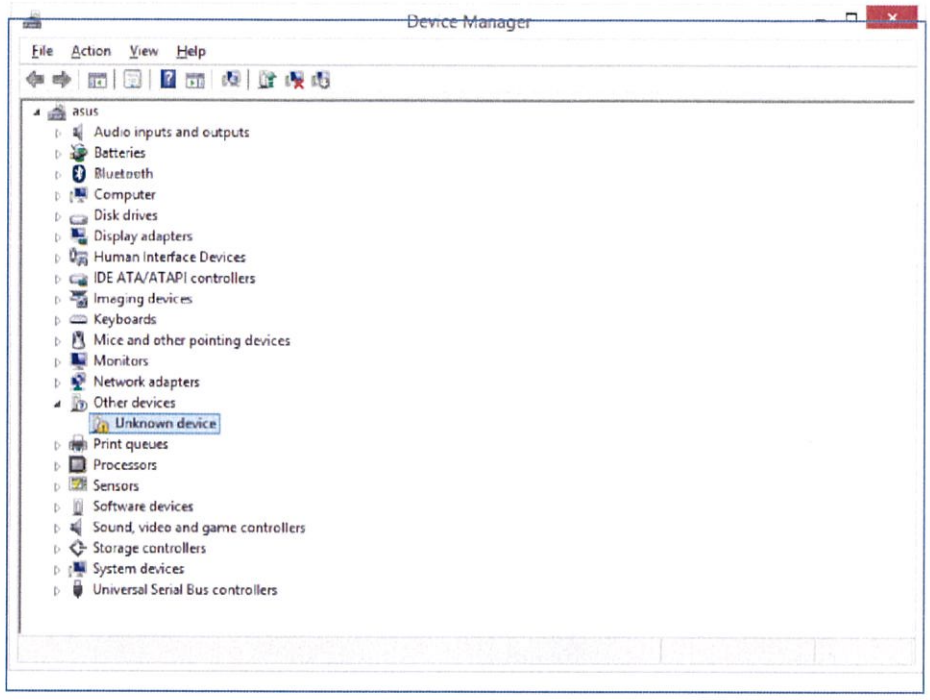
รูปที่ ก.11 ขั้นตอนการติดตั้ง Device Driver (ต่อ)

c) เลือก System



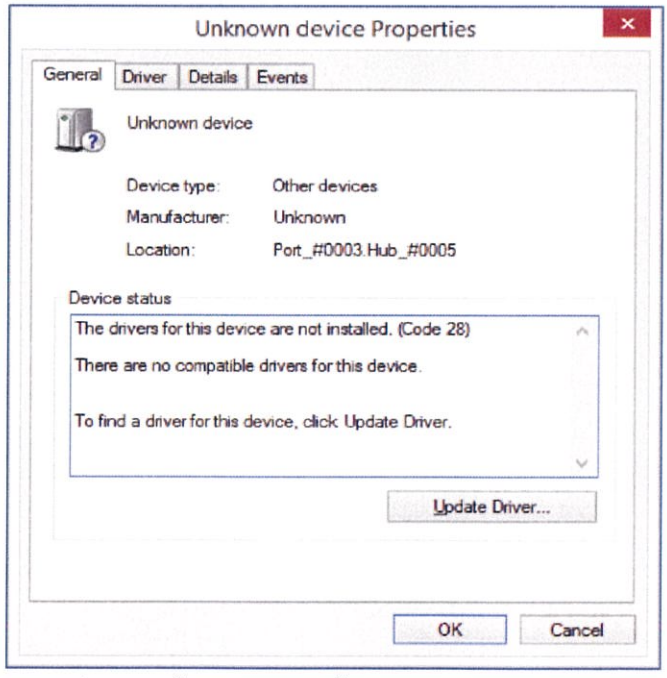
รูปที่ ก.12 ขั้นตอนการติดตั้ง Device Driver (ต่อ)

d) เลือก Device Manager



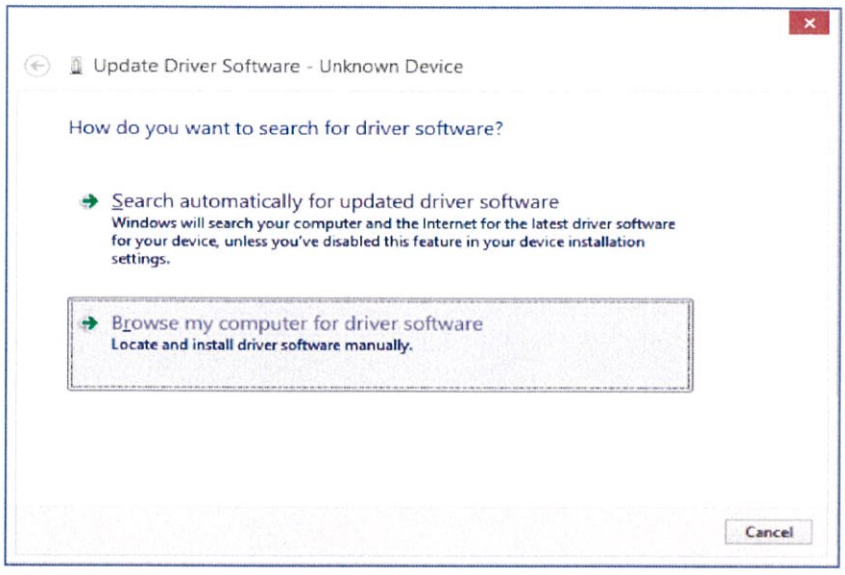
รูปที่ ก.13 ขั้นตอนการติดตั้ง Device Driver (ต่อ)

e) ดับเบิลคลิกที่ Unknown device



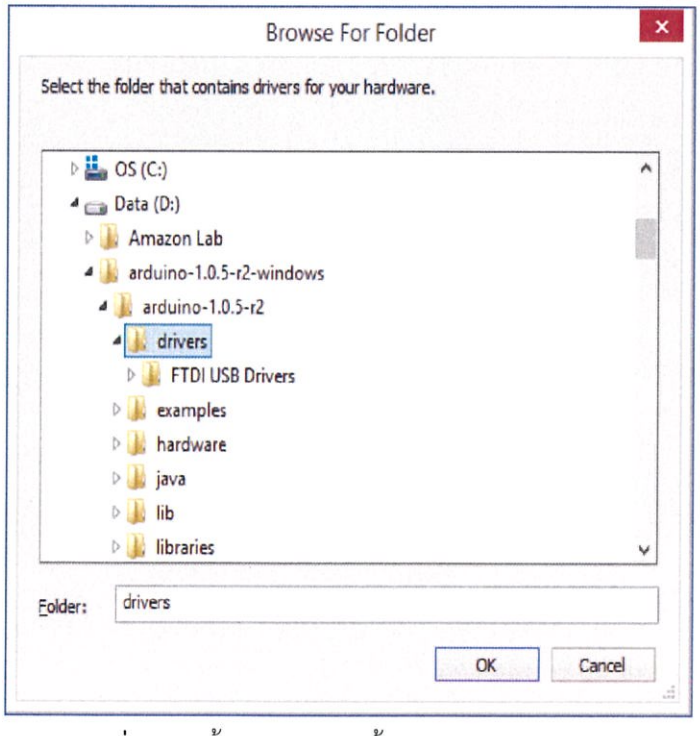
รูปที่ ก.14 ขั้นตอนการติดตั้ง Device Driver (ต่อ)

f) กดปุ่ม Update driver



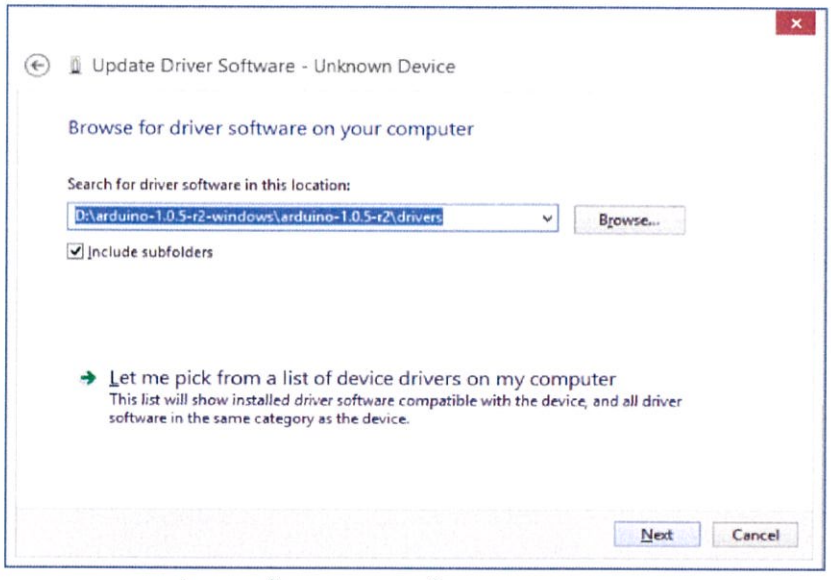
รูปที่ ก.15 ขั้นตอนการติดตั้ง Device Driver (ต่อ)

g) เลือก Browse my computer for driver software



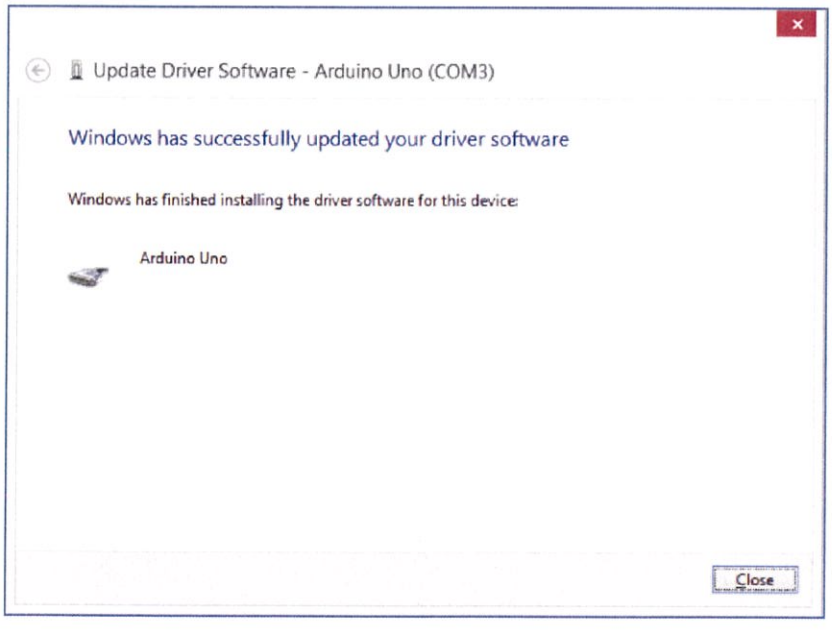
รูปที่ ก.16 ขั้นตอนการติดตั้ง Device Driver (ต่อ)

h) เลือกโฟลเดอร์ drivers จากตำแหน่งโฟลเดอร์ที่ได้ติดตั้งซอฟต์แวร์ Arduino



รูปที่ ก.17 ขั้นตอนการติดตั้ง Device Driver (ต่อ)

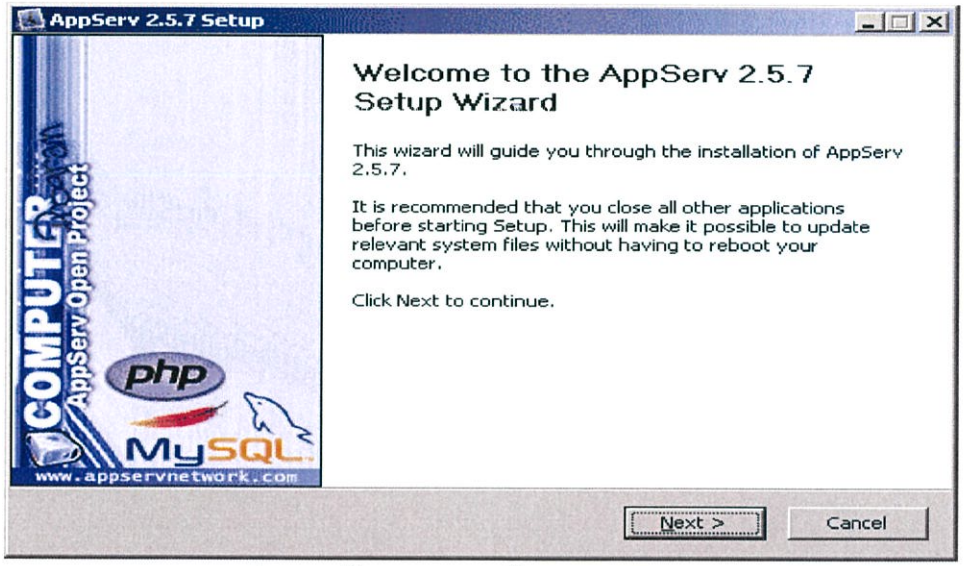
i) เมื่อทำการติดตั้งเรียบร้อยจะปรากฏหน้าจอ แสดงว่าได้ทำการอัพเดทซอฟต์แวร์เรียบร้อย (Windows has successfully updated your driver software)



รูปที่ ก.18 ขั้นตอนการติดตั้ง Device Driver (ต่อ)

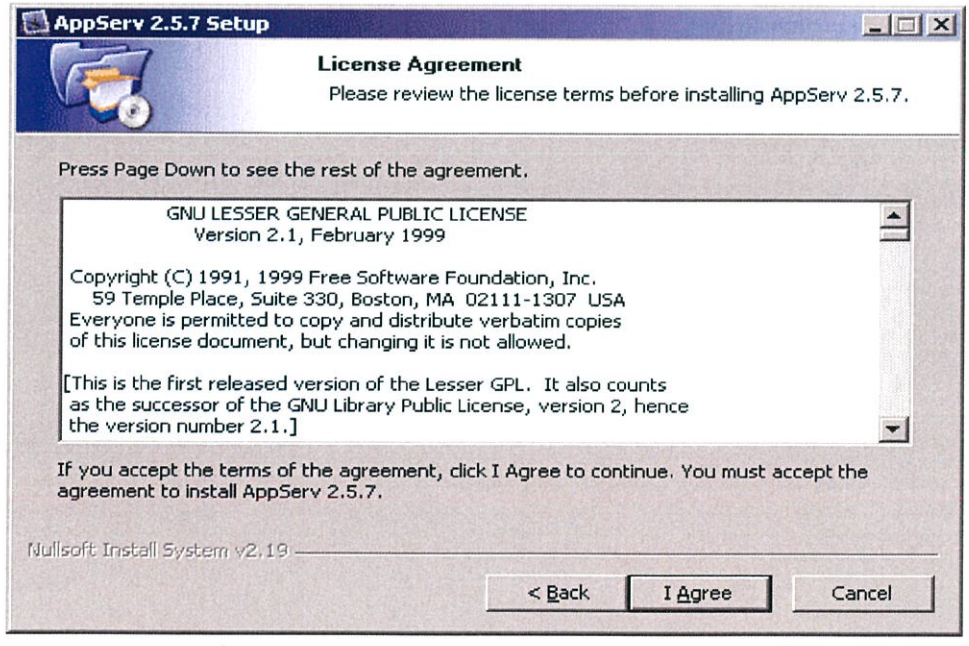
การติดตั้งโปรแกรม AppServ

1. ดับเบิ้ลคลิกไฟล์ appserv-win32-x.x.x.exe เพื่อทำการติดตั้ง จะปรากฏหน้าจอตามรูปที่ 1



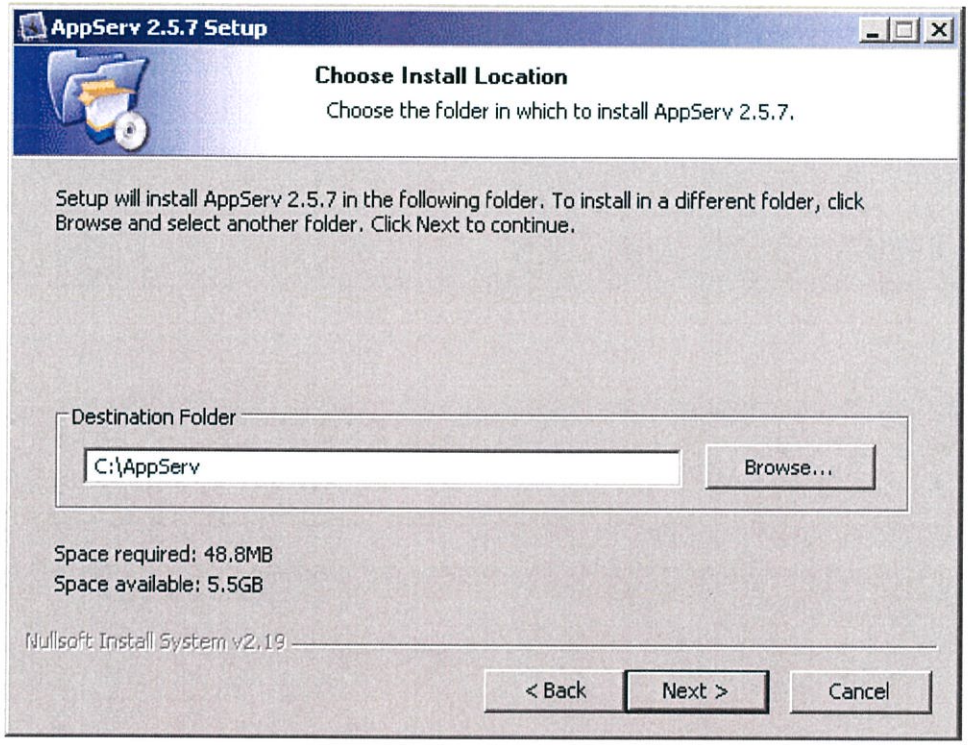
รูปที่ ก.19 ขั้นตอนการติดตั้งโปรแกรม AppServ

2. เข้าสู่ขั้นตอนเงื่อนไขการใช้งานโปรแกรม โดยโปรแกรม AppServ ได้แจกจ่ายในรูปแบบ GNU License หากผู้ติดตั้งอ่านเงื่อนไขต่างๆ เสร็จสิ้นแล้ว หากยอมรับเงื่อนไขให้กด Next เพื่อเข้าสู่การติดตั้งในขั้นต่อไป แต่หากว่าไม่ยอมรับเงื่อนไข ให้กด Cancel เพื่อออกจากการติดตั้งโปรแกรม AppServ ดังรูปตัวอย่างที่ 2



รูปที่ ก.20 แสดงรายละเอียดเงื่อนไขการ GNU License

3. เข้าสู่ขั้นตอนการเลือกปลายทางที่ต้องการติดตั้ง โดยค่าเริ่มต้นปลายทางที่ติดตั้งจะเป็น C:AppServ หากต้องการเปลี่ยนปลายทางที่ติดตั้ง ให้กด Browse แล้วเลือกปลายทางที่ต้องการ ตามรูปที่ 3 เมื่อเลือกปลายทางเสร็จสิ้น ให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป



รูปที่ ก.21 เลือกปลายทางการติดตั้งโปรแกรม AppServ

4. เลือก Package Components ที่ต้องการติดตั้ง โดยค่าเริ่มต้นนั้นจะให้เลือกลงทุก Package แต่หากว่าผู้ใช้งานต้องการเลือกลงเฉพาะบาง Package ก็สามารถเลือกตามข้อที่ต้องการออก โดยรายละเอียดแต่ละ Package มีดังนี้

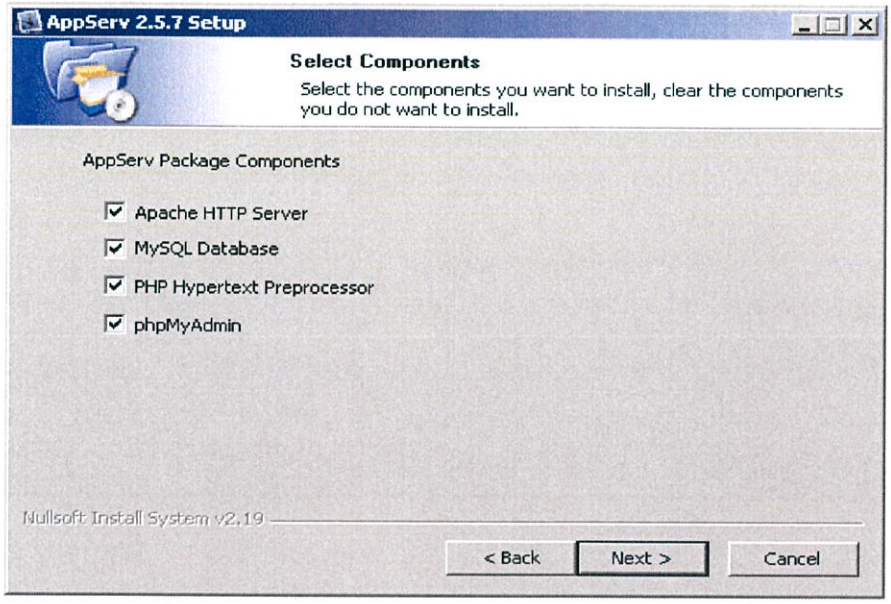
Apache HTTP Server คือ โปรแกรมที่ทำหน้าเป็น Web Server

MySQL Database คือ โปรแกรมที่ทำหน้าเป็น Database Server

PHP Hypertext Preprocessor คือ โปรแกรมที่ทำหน้าประมวลผลการทำงานของภาษา PHP

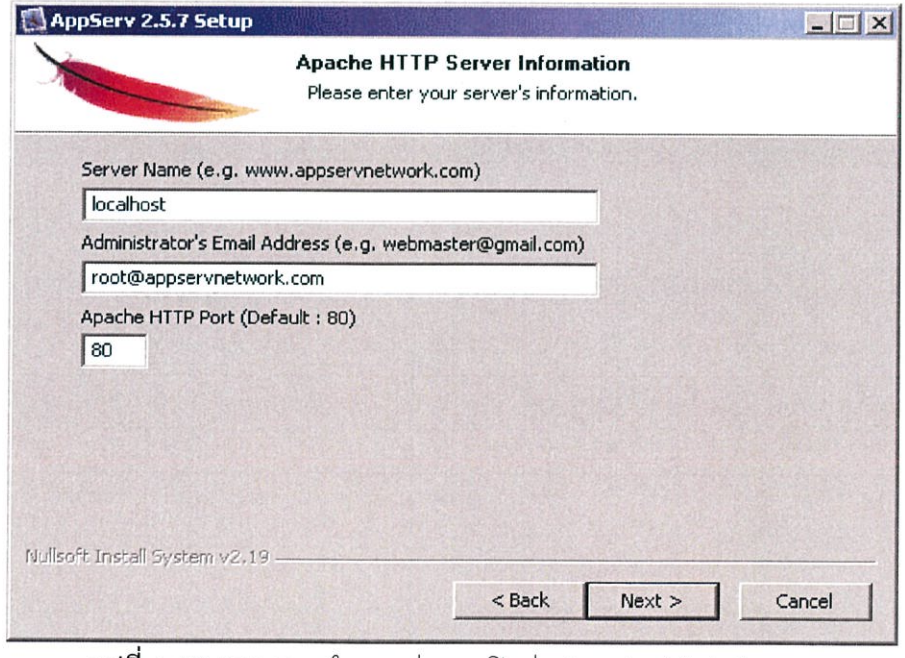
phpMyAdmin คือ โปรแกรมที่ใช้ในการบริหารจัดการฐานข้อมูล MySQL ผ่านเว็บไซต์

เมื่อทำการเลือก Package ตามรูปที่ 4 เรียบร้อยแล้ว ให้กด Next เพื่อเข้าสู่ขั้นตอนการติดตั้งต่อไป



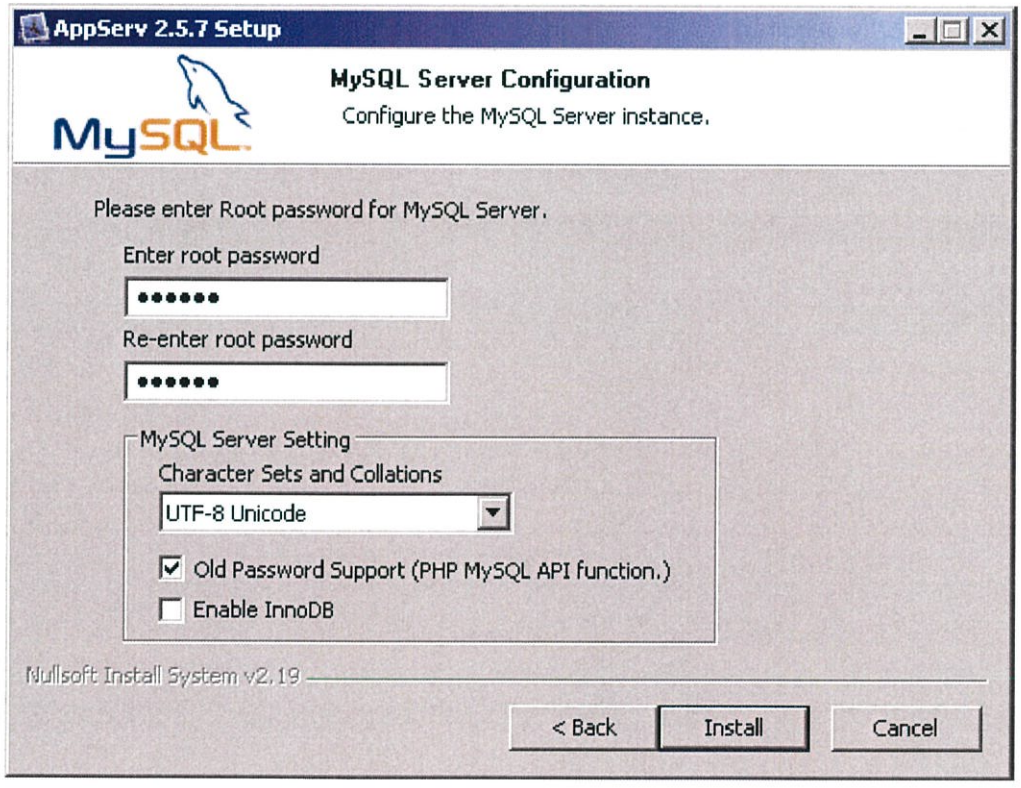
รูปที่ ก.22 เลือก Package Components ที่ต้องการติดตั้ง

5. กำหนดค่าคอนฟิกของ Apache Web Server มีอยู่ด้วยกันทั้งหมด 3 ส่วน ตามรูปที่ 5 คือ Server Name คือช่องสำหรับป้อนข้อมูลชื่อ Web Server ของท่านเช่น www.appservnet.com Admin Email คือช่องสำหรับป้อนข้อมูล อีเมล์ผู้ดูแลระบบ เช่น root@appservnet.com HTTP Port คือช่องสำหรับระบุ Port ที่จะเรียกใช้งาน Apache Web Server โดยทั่วไปแล้ว Protocol HTTP นั้นจะมีค่าหลักคือ 80 หากว่าท่านต้องการหลีกเลี่ยงการใช้ Port 80 ก็สามารถแก้ไขได้ หากมีการเปลี่ยนแปลง Port การเข้าใช้งาน Web Server แล้ว ทุกครั้งที่เรียกใช้งานเว็บไซต์จำเป็นที่ต้องระบุหมายเลข Port ด้วย เช่น หากเลือกใช้ Port 99 ในการเข้าเว็บไซต์ทุกครั้งต้องใช้ http://www.appservnet.com:99 จึงจะสามารถเข้าใช้งานได้



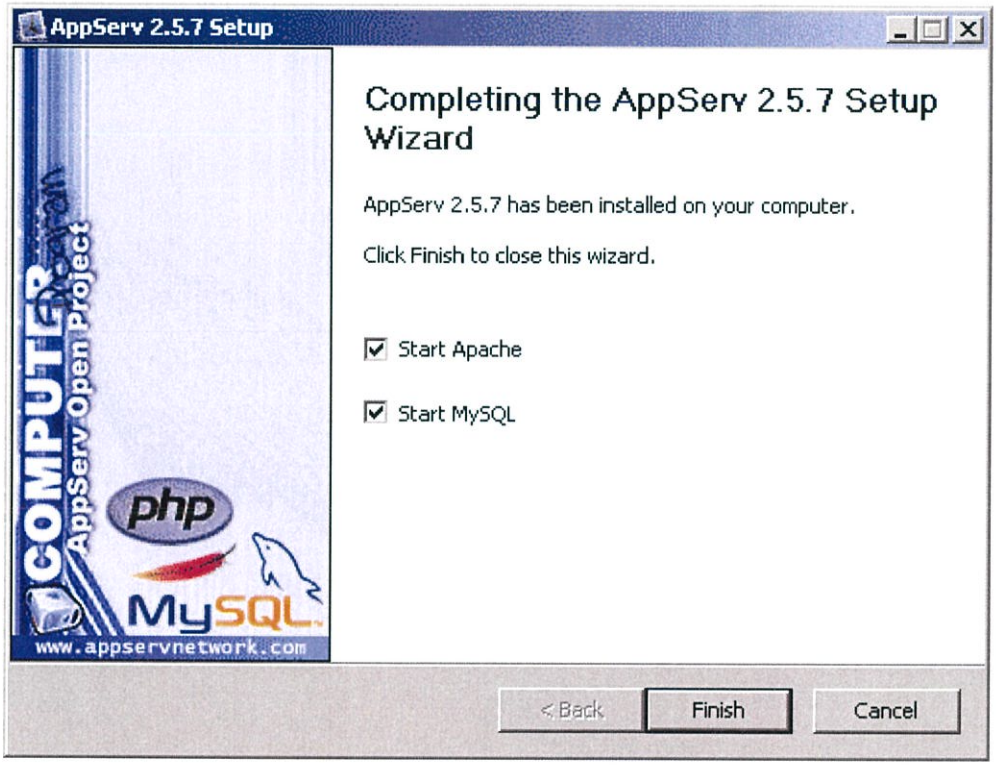
รูปที่ ก.23 แสดงการกำหนดค่าคอนฟิกค่า Apache Web Server

6. กำหนดค่าคอนฟิกของ MySQL Database มีอยู่ด้วยกันทั้งหมด 3 ส่วน ตามรูปที่ 6 คือ Root Password คือช่องสำหรับป้อน รหัสผ่านการเข้าใช้งานฐานข้อมูลของ Root หรือผู้ดูแลระบบทุกครั้งที่เข้าใช้งานฐานข้อมูลในลักษณะที่เป็นผู้ดูแลระบบ ให้ระบุ user คือ root Character Sets ใช้ในการกำหนดค่าระบบภาษาที่ใช้ในการจัดเก็บฐานข้อมูล, เรียงลำดับฐานข้อมูล ,Import ฐานข้อมูล, Export ฐานข้อมูล, ติดต่อฐานข้อมูล Old Password หากท่านมีปัญหาเกี่ยวกับการใช้งาน PHP กับ MySQL API เวอร์ชั่นเก่า โดยเจอ Error Client does not support authentication protocol requested by server; consider upgrading MySQL client ให้เลือกในส่วนของ Old Password เพื่อหลีกเลี่ยงปัญหานี้ Enable MyISAM หากท่านต้องการใช้งานฐานข้อมูลในรูปแบบ MyISAM ให้เลือกในส่วนนี้ด้วย



รูปที่ ก.24 แสดงการกำหนดค่าคอนฟิกของ MySQL Database

7. สิ้นสุดขั้นตอนการติดตั้งโปรแกรม AppServ สำหรับขั้นตอนสุดท้ายนี้จะมีให้เลือกว่าต้องการสั่งให้มีการรัน Apache และ MySQL ทันทีหรือไม่ จากนั้นกดปุ่ม Finish เพื่อเสร็จสิ้นการติดตั้งโปรแกรม AppServ



รูปที่ ก.25 แสดงหน้าจอขั้นตอนสิ้นสุดการติดตั้งโปรแกรม AppServ

การอัพโปรแกรมลงบอร์ดโมดูลไวไฟ ESP8266 E-12F

1.ถอดชิพของตัวไมโครคอนโทรลเลอร์ออกจากบอร์ด Arduino UNO R3

2.ต่อสาย ดังนี้

3. จากนั้นทำการอัพโปรแกรมลงโมดูลไวไฟ ESP8266 ให้เรียบร้อย โมดูลไวไฟพร้อมใช้งาน

ตารางที่ ก.1 การต่อขาของโมดูลไวไฟเพื่อทำการอัพโหลดโปรแกรม

โมดูลไวไฟ ESP8266 E-12F	บอร์ด Arduino UNO R3
TX	TX
RX	RX
GND	GND
VCC	5V
CH_PD	5V
RST	RST
GPIO0	GND

การต่อใช้งานโมดูลไวไฟ ESP8266 E-12F ในวงจรใช้งานจริง

1. ถอดขา GPIO0 ของโมดูลไวไฟ ที่ต่อกับ ขา GND ของบอร์ดอาดูโนออก

2. นำชิพไมโครคอนโทรลเลอร์ที่ถอดออกใส่ลงในบอร์ดอาดูโน

3. นำขา TX ของบอร์ดอาดูโน ต่อกับขา RX ของโมดูลไวไฟ ESP8266 E-12F ดังนี้

ตารางที่ ก.2 การต่อขาโมดูลไวไฟเพื่อใช้งานจริง

โมดูลไวไฟ ESP8266 E-12F	บอร์ด Arduino UNO R3
RX	TX
VCC	5V
CH_PD	5V
GND	GND
RST	RST

ภาคผนวก ข.

คู่มือการใช้งาน

1. เมนูอาหาร

ตารางที่ ข.1 เมนูอาหาร

รหัสรายการอาหาร	รายการอาหาร	ราคา (บาท)
01	ข้าวผัดกระเพราหมู	40
02	ข้าวผัดหมู ไก่	40
03	ข้าวผัดกุ้ง	50
04	ข้าวผัดต้มยำทะเล	50
05	ข้าวผัดหมู	40
06	กระเพราหมูกรอบ	40
07	ผัดคะน้าหมูกรอบ	40
08	ผัดผักบุ้งหมูสับ	40
09	ผัดไทกุ้งสด	45
10	ผัดพริกแกงหมูชิ้น	45
11	ผัดฉ่าปลาแซลมอน	100
12	ยำไข่ปลาคาเวียร์	200
13	ผัดพริกอ่อนหมู	40
14	ข้าวราดหมูผัดซอส	40
15	ข้าวผัดน้ำพริกลงเรือ	45
16	สุกี้หมู	40
17	สุกี้ทะเล	45
18	ราดหน้าหมู	40
19	ไข่เจียวยัดใส้	30
20	ต้มยำหม้อไฟ	150
21	คอหมูย่าง	50
22	ตำไทยไข่เค็ม	30
23	ตำไทยปู	30
24	ตำปูปลาร้า	30

ตารางที่ ข.1 (ต่อ) เมนูอาหาร

25	พิซซ่าหน้าหมู	99
26	ปีกไก่ทอด	50
27	น่องไก่ทอด	50
28	ข้าวผัดน้ำพริกกะปิ	50
29	ข้าวผัดน้ำพริกนรก	50
0	คะน้าปลากระป๋อง	45
31	ยำหมูยอ	50
32	ยำวุ่นเส้น	50
33	กุ้งแช่น้ำปลา	150
34	กุ้งเผา	200
35	ปลาเผา	200
36	ยำถั่วพู	50
37	ห่อหมกมะพร้าวอ่อน	100
38	ข้าวราดหมูน้ำมันหอย	40
39	คะน้าปลาเค็ม	50
40	ผัดเผ็ดหมูป่า	100
41	ไก่ย่างสมุนไพร	80
42	สเต็กหมูพริกไทยดำ	79
43	สเต็กไก่สไปซี่	79
44	สเต็กไก่พริกไทยดำ	79
45	พริกเผาหมู	40
46	ผัดผักบุ้งไฟแดง	50
47	เย็นตาโฟต้มยำ	50
48	ก๋วยเตี๋ยวหมูต้มยำ	45
49	ข้าวหมูน้ำมันหอย	45
50	ข้าวราดผัดผักรวม	40
51	ข้าวหน้าเป็ด	50
52	ข้าวหน้าไก่ทอด	45
53	ข้าวหน้าเนื้อ	60
54	ข้าวหมูอบ	55

ตารางที่ ข.1 (ต่อ) เมนูอาหาร

55	ข้าวไก่อบ	55
56	ข้าวซอย	45
57	ขนมจีนน้ำเงี้ยว	40
58	ขนมจีนน้ำยาป่า	40
59	ขนมจีนน้ำยากะทิ	40
60	ยำขนมจีน	50
61	โค้ก	30
62	น้ำส้ม	30
63	น้ำแดง	30
64	ชานมเย็น	20
65	ชาดำเย็น	20
66	ชามะนาว	20
67	ชาเขียวนม	20
68	ชานมปั่น	25
69	ชาเขียวนมปั่น	25
70	น้ำเก๊กฮวย	20
71	น้ำกระเจี๊ยบ	20
72	น้ำใบบัวบก	20
73	น้ำลำไย	20
74	น้ำเฉาก๊วย	20
75	สละลอยแก้ว	50
76	ข้าวเหนียวมะม่วง	50
77	ผลไม้รวม	40
78	บัวลอยไข่หวาน	30
79	ทับทิมกรอบ	50
80	ลอดช่อง	30

2. วิธีการสั่งอาหาร

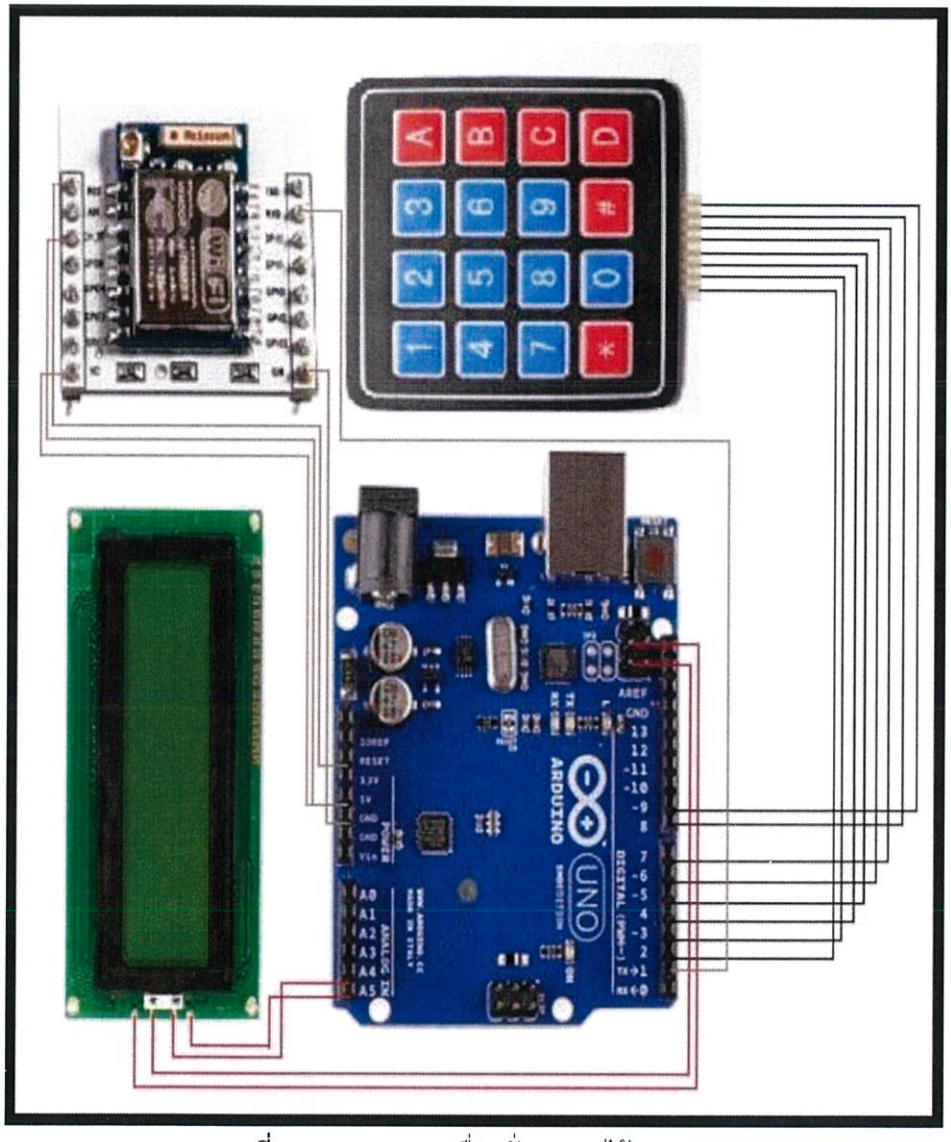
รูปที่ ข.1 แป้นพิมพ์ขนาด 4 X 4

1. เลือกรหัสรายการอาหารที่ต้องการ
2. กดรหัสรายการอาหารลงบนแป้นพิมพ์
3. กด A 1 ครั้งเพื่อยืนยัน
4. กดจำนวนที่ต้องการลงบนแป้นพิมพ์
5. กด A 1 ครั้งเพื่อยืนยัน
6. รอรับรายการอาหารที่ต้องการ

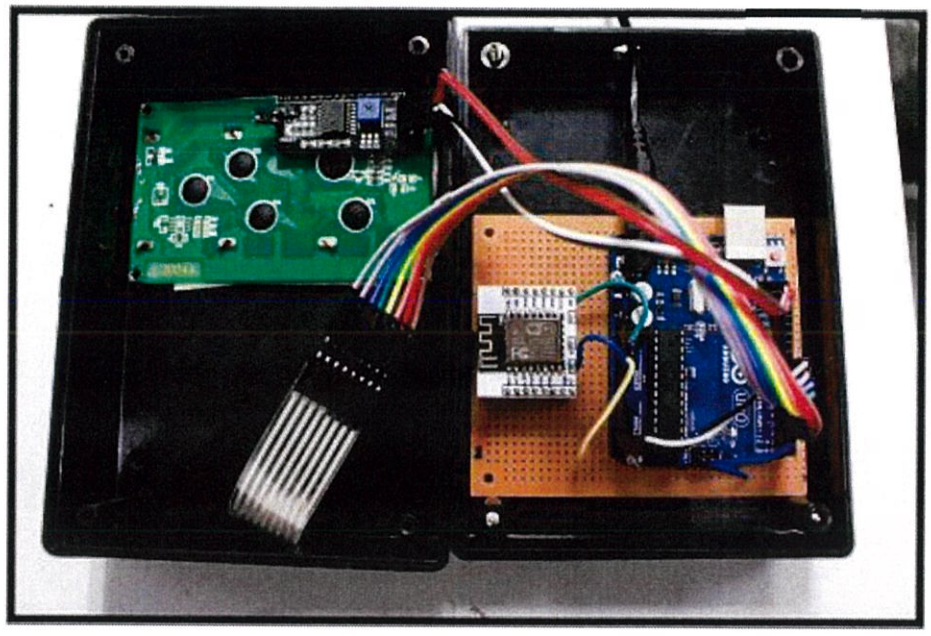
ภาคผนวก ค.

ชิ้นงาน

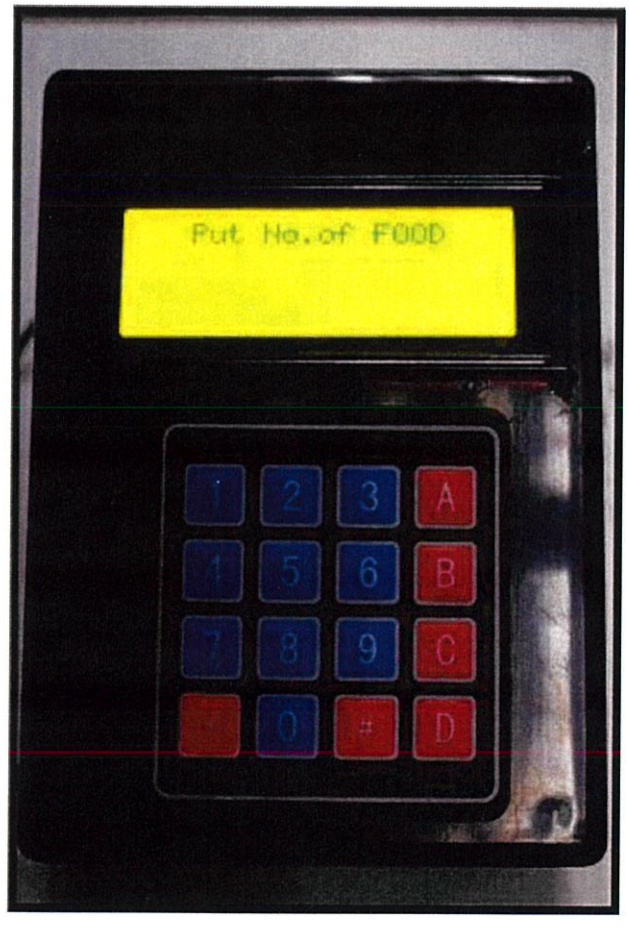
วงจรเครื่องสั่งรายการอาหาร



รูปที่ ค.1 วงจรของเครื่องสั่งอาหารไร้สาย



รูปที่ ค.2 ภายในเครื่องรับรายการอาหาร



รูปที่ ค.3 เครื่องสั่งรายการอาหาร

ภาคผนวก ง.

โปสเตอร์ระบบสั่งอาหารไร้สาย

IC-046

Wireless Food Ordering System

Kittiphop Chaiyaphan, Thanapat Poltue and Tanasart Kiewmanee
Advisor: Assoc.Prof.Dr.Worrapong Tangsrirat
Department of Instrumentation and Control Engineering
E-mail: freedomblood0093@gmail.com,thanapatpol@hotmail.com, tanasartk@gmail.com

Abstract

Nowadays, most of the restaurants still use the waiters to take food orders from customers and prepare itemized checks and take payments. This maybe have some mistakes by human errors. The concept of this project is to design a customizable food ordering system. The customers can make an order through a keypad with the LCD screen. The order was updated in central database and subsequently sent to kitchen via Wi-Fi system. The interface system was designed from Visual Studio program and provided the new update all the time. In addition, the restaurant can also manage the list of food and payment information by themselves. The main structure of this system is based on Arduino microcontroller board, LCD module, keypad module and Wi-Fi module. This project will provide the convenience, efficiency and accuracy for restaurants by saving time, reducing human errors, as well as real-time customer feedback.

Introduction

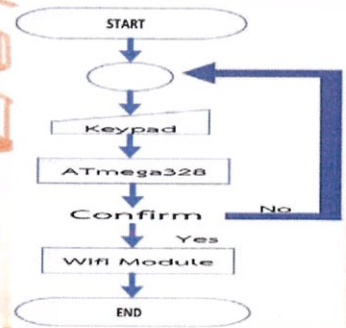
The convergence of wireless technologies can facilitate ubiquitous platform for implementing business applications such as food ordering system.

Without any information and communication technology facilities, food ordering procedures require waiters to note orders from customers, bring orders to the kitchen, write receipts, and deliver the ordered menu. Although such routines look simple, the conventional food ordering system may significantly increase the work load of waiters or even prone to human errors in note making when the number of customers increases during peak hours.

The concept of this project is to design a customizable food ordering system. Hopefully this project will provide convenience and efficiency for restaurants owner by saving time, reducing human errors, as well as real-time customer feedback.

Methodology

Flowchart of wireless food ordering system



Results

Figure1. Customers use Keypad to ordering food

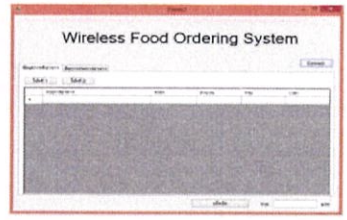


Figure2. User interface for the restaurant, its display new order from customer.

Conclusion

The advantage of this project is that the restaurant owner can manage the food list or update new menu by themselves, Combine with user-friendly interface design and easy to use.

An interesting future improvement might involve mobile devices in the mobile application platform.

References

[1] K. J. Patel, et al., "PDA-based Wireless Food Ordering System for Hospitality Industry - A Case Study of Box Hill Institute," in Wireless Telecommunications Symposium 2007, Pomona, CA, 2007, pp. 1-8.

[2] K. Kamarudin, et al., "The Application of Wireless Food Ordering System," MASAUM Journal of Computing, vol. 1, pp. 178-184, 2009.

รูปที่ ง.1 โปสเตอร์ระบบสั่งอาหารไร้สาย

ภาคผนวก จ.

Source Code

โปรแกรมเครื่องรับรายการอาหาร

ส่วนของจอแอลซีดี คีย์แพท

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,2,1,0,4,5,6,7);
#include <Keypad.h>

const byte ROWS = 4; //four rows
const byte COLS = 4; //three columns
char keys[ROWS][COLS] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};

byte rowPins[ROWS] = {2, 3, 4, 5}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {6, 7, 8, 9}; //connect to the column pinouts of the keypad
Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );


void setup() {
        Serial.begin(9600);
        lcd.begin (20,4);
        lcd.setBacklightPin(3,POSITIVE);
```

```
        lcd.setBacklight(HIGH);
        lcd.setCursor(5,0);
        lcd.print("WELCOME TO");
        lcd.setCursor(5,2);
        lcd.print("CT CUISINE");
        delay(3000);
        lcd.clear();
}
int menu=0;
char key;
String food_id;
String food_num;
String food_table="1";  //ตัวแก้เลขโต๊ะ
int loca1;
int loca2;
void loop() {
        key = keypad.getKey();
        if (key){
        Serial.println(key);
  }
   if(key=='1' || key=='2' || key=='3' || key=='4' || key=='5' || key=='6' || key=='7' ||
key=='8' || key=='9' || key=='0'){
        if(menu==0){
        if(loca1<=1){
        loca1+=1;
        food_id+=String(key);
        lcd.clear();
        }else{
```

```
          loca1=0;
          food_id="";
          lcd.clear();
        }
      }
          if(menu==1){
          if(loca2<=1){
          loca2+=1;
          food_num+=String(key);
          lcd.clear();
        }else{
          loca2=0;
          food_num="";
          lcd.clear();
        }
      }
    }
    if(menu==0 && key=='A'){
menu=1;
lcd.clear();
  }
          if(key=='A' && menu==1 && food_num.toInt()>=1){
lcd.clear();
lcd.setCursor(6,1);
lcd.print("Send OK");
Serial.print("<");
```

```
Serial.print(food_id);
Serial.print(",");
Serial.print(food_num);
Serial.print(",");
Serial.print(food_table);
Serial.println(",>");
delay(1000);
        loca1=0;
        loca2=0;
        food_id="";
        food_num="";
        menu=0;
        lcd.clear();
  }
        if(menu==0 && key=='#'){
loca1=0;
food_id="";
menu=0;
lcd.clear();
  }
        if(key=='#' && menu==1){
loca2=0;
food_num="";
menu=1;
lcd.clear();
  }
        if(key=='C'){
loca1=0;
```

```
loca2=0;
food_id="";
food_num="";
menu=0;
lcd.clear();
  }
        if(menu==0){
main_menu();
}else if(menu==1){
main_menu2();
  }
}
 void main_menu2(){
main_menu();
lcd.setCursor(5,2);
lcd.print("Put Amount");
lcd.setCursor(9,3);
lcd.print(food_num);
}
void main_menu(){
lcd.setCursor(3,0);
lcd.print("Put No.of FOOD");
lcd.setCursor(9,1);
lcd.print(food_id);
}
```

ส่วนของโมดูลไวไฟ ESP8266 E-12F

```
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <ESP8266WiFiMulti.h>
#include <ESP8266HTTPClient.h>
#define USE_SERIAL Serial
ESP8266WiFiMulti WiFiMulti;
HTTPClient http;
String command = "";
String command2="";
char commandStart = '<';
char commandEnd = '>';

String food;
String foodnum;
String table;

void setup() {

    USE_SERIAL.begin(9600);
       // USE_SERIAL.setDebugOutput(true);

USE_SERIAL.println();
USE_SERIAL.println();
USE_SERIAL.println();

       for(uint8_t t = 4; t > 0; t--) {
       USE_SERIAL.printf("[SETUP] WAIT %d...\n", t);
       USE_SERIAL.flush();
       delay(1000);

  }
WiFiMulti.addAP("ECC-208", "control1234");

}
```

```
String getStringPartByNr(String data, char separator, int index)

{

          int stringData = 0;

          String dataPart = "";

          for(int i = 0; i<data.length()-1; i++) {

          if(data[i]==separator) {

          stringData++;

          }else if(stringData==index) {

          dataPart.concat(data[i]);

          }else if(stringData>index) {

          return dataPart;

          break;

      }

    }

    return dataPart;

}

void connect_serv(){

          if((WiFiMulti.run() == WL_CONNECTED)) {

          http.begin("192.168.2.101", 80,

"/food/add.php?f="+food+"&fn="+foodnum+"&t="+table+"");

          int httpCode = http.GET();

          if(httpCode) {

          USE_SERIAL.printf("[HTTP] GET... code: %d\n", httpCode);

          // file found at server

          if(httpCode == 200) {

          http.writeToStream(&USE_SERIAL);

           }
```

```
        } else {
            USE_SERIAL.print("[HTTP] GET... failed, no connection or no HTTP server\n");
        }
    }
}
void handle(String command) {
        Serial.println("Received: '" + command + "'");
        food=getStringPartByNr(command, ',', 0);
        foodnum=getStringPartByNr(command, ',', 1);
        table=getStringPartByNr(command, ',', 2);
        connect_serv();
}
void loop() {
        while (Serial.available() > 0) {
        int input = Serial.read();
        if (input == commandStart) {
        command = "";
} else if (input == commandEnd) {
        handle(command);
        command = "";
} else {
        command += (char)input;
    }
  }
}
```

โปรแกรมส่วนของ Server

```
Imports MySql.Data.MySqlClient
Public Class Form1
Public Shared table_num As Integer = 1
Friend WithEvents Connection As New
MySqlConnection("SERVER=localhost;DATABASE=food;UID=root;PASSWORD=1234;")

    Public Function Disconnect() As Boolean
    Try
    Connection.Close()
    Return True
    Catch ex As MySqlException
    MessageBox.Show(ex.Message)
    End Try
    Return False
End Function

    Public Function Connect() As Boolean
    Try
    Connection.Open()
    Return True
    Catch ex As MySqlException
    MessageBox.Show(ex.Message)
    End Try
    Return False
End Function

    Public Function update_add() As Boolean
    Dim cmd As MySqlCommand = New MySql.Data.MySqlClient.MySqlCommand(
_
    "select food_id,food_name,food_price from food", Connection)
```

```
        Dim adapter2 As New MySqlDataAdapter(cmd)
        Dim Table2 As New DataTable()
        adapter2.Fill(Table2)

        DataGridView2.DataSource = Table2
        Return False

End Function

        Public Function update_table() As Boolean
        Dim cmd As MySqlCommand = New MySql.Data.MySqlClient.MySqlCommand(
_
        "select food_name,food_num,food_price,bill_time,(food_num * food_price)
        AS balance from food_bill inner join food on food_bill.food_id=food.food_id
        where food_bill.bill_check='0' and food_bill.food_table='" +
        table_num.ToString + "'", Connection)

        Dim adapter As New MySqlDataAdapter(cmd)
        Dim Table As New DataTable()

adapter.Fill(Table)
        dataGridView1.DataSource = Table
        'If Table.Rows.Count > 0 Then
        Me.TextBox3.Text = Table.Rows(0)("balance")
        'End If
        'If Table.Rows.Count > 0 Then
        Dim iTotal As Integer
        Dim food_price As Integer
        Dim sum As Integer
        Dim sumall As Integer
        For Each row As DataGridViewRow In Me.dataGridView1.Rows

                sum = 0
                iTotal = 0
```

```
                food_price = 0
                If Table.Rows.Count > 0 Then
                iTotal = iTotal + row.Cells("Column2").Value
                food_price = food_price + row.Cells("Column3").Value
                sum = food_price * iTotal
                sumall += sum

            End If

        Next
            TextBox3.Text = sumall
            Return False

    End Function

        Private Sub conn_disconn_button_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles conn_disconn_button.Click
        If Connection.State = System.Data.ConnectionState.Closed Then
        Connect()
        update_add()
        conn_disconn_button.Text = "Disconnect"
        Elself Connection.State = System.Data.ConnectionState.Open Then
        Disconnect()
        conn_disconn_button.Text = "Connect"

      End If

    End Sub

        Private Sub Add_Student_OR_Subject_Button_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles AddStudentButton.Click
        Dim query As String = ""
        query = ("INSERT INTO food (food_id, food_name, food_price) VALUES('" +
food_id.Text & "','" + food_name.Text & "','" + food_price.Text & "')")
        Dim cmd As New MySqlCommand(query, Connection)
```

```
        '  update_add()
        Try
         'Execute command
        cmd.ExecuteNonQuery()
        ErrorLabel.Text = ""
        '' MessageBox.Show("Successful Adding")
        ' update_add()
        update_add()
        Catch ex As Exception
        ErrorLabel.Text = ex.Message

      End Try

  End Sub

      Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click, Button1.Click
      table_num = 1
      update_table()

  End Sub
      Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click, Button2.Click
      table_num = 2
      update_table()

  End Sub
      Private Sub Button3_Click(sender As Object, e As EventArgs) Handles
Button3.Click
      Dim cmd As MySqlCommand = New MySql.Data.MySqlClient.MySqlCommand(
_
      "update food_bill set bill_check='1' where food_table='" +
table_num.ToString + "'", Connection)
      Dim adapter As New MySqlDataAdapter(cmd)
```

```
        Dim Table As New DataTable()
        adapter.Fill(Table)
        ' dataGridView1.DataSource = Table
        update_table()

    End Sub

        Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles
Timer1.Tick
        If Connection.State = System.Data.ConnectionState.Open Then
        update_table()

      End If

    End Sub

        Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

    End Sub

        Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Dim mResult
        Dim ObjConnection As New SqlConnection()
        mResult = MsgBox("You are really want to delete the slected record?", _
        vbYesNo + vbQuestion, "Removal Confirmation")
        If mResult = vbNo Then

         Exit Sub

      End If

      Try
        Dim cmd As MySqlCommand = New MySql.Data.MySqlClient.MySqlCommand(
_
        "delete from food where food_id='" &
```

```
Me.DataGridView2.CurrentRow.Cells(0).Value & "'", Connection)
            cmd.ExecuteNonQuery()
            update_add()
        Finally
            ObjConnection.Close()
        End Try
    End Sub
End Class
```