ระบบสั่งอาหารผ่านเทคโนโลยี NFC และ QR CODE บนระบบปฏิบัติการแอนดรอยด์และส่งอาหารผ่านหุ่นยนต์อัตโนมัติ

FOOD ORDERING SYSTEM FOR RESTAURANT USING NFC TECHNOLOGY AND QR CODE WITH AUTOMATIC SERVED FOOD VIA ROBOT

โดย

นายนภวิชญ์ คำสมหมาย

นายพิเชฐ แซ่ต่าน

นายศิรวิชญ์ กิตติวิชญกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2556

ระบบสั่งอาหารผ่านเทคโนโลยี NFC และ QR CODE บนระบบปฏิบัติการแอนดรอยด์และส่งอาหารผ่านหุ่นยนต์อัตโนมัติ

FOOD ORDERING SYSTEM FOR RESTAURANT USING NFC TECHNOLOGY AND QR CODE WITH AUTOMATIC SERVED FOOD VIA ROBOT

โดย

นายนภวิชญ์ คำสมหมาย

นายพิเชฐ แซ่ต่าน

นายศิรวิชญ์ กิตติวิชญกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2556

ระบบสั่งอาหารผ่านเทคโนโลยี NFC และ QR CODE บนระบบปฏิบัติการแอนดรอยด์และส่งอาหารผ่านหุ่นยนต์อัตโนมัติ

FOOD ORDERING SYSTEM FOR RESTAURANT USING NFC TECHNOLOGY AND QR CODE WITH AUTOMATIC SERVED FOOD VIA ROBOT

โดย

นายนภวิชญ์	คำสมหมาย	53010799
นายพิเชฐ	แซ่ต่าน	53011125
นายศิรวิชญ์	กิตติวิชญกุล	53011567

อาจารย์ที่ปรึกษา

ผศ.ดร. นภัทร สระเอี่ยม

ผศ.ดร. ธเนศ พัฒนธาดาพงษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2556

ผ่านการตรวจรูปเล่มแล้ว

(.....)

อาจารย์ที่ปรึกษา

17/ม.ค./57

KMITL วิศวกรรมโทรคมนาคม Telecommunications Engineering

ผ่านการตรวจชิ้นงานแล้ว

(.....)

กรรมการผู้ตรวจชิ้นงาน

18/4/57

KMITL วิศวกรรมโทรคมนาคม Telecommunications Engineering

ปริญญานิพนธ์ปีการศึกษา 2556
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง ระบบสั่งอาหารผ่านเทคโนโลยี NFCและ QR CODEบนระบบปฏิบัติการแอนดรอยด์และส่งอาหารผ่านหุ่นยนต์อัตโนมัติ
FOOD ORDERING SYSTEM FOR RESTAURANT USING NFC TECHNOLOGY AND QR CODE WITH AUTOMATIC SERVED FOOD VIA ROBOT

ผู้จัดทำ

1. นายนภวิชญ์	คำสมหมาย	53010799
2. นายพิเชฐ	แซ่ต่าน	53011125
3. นายศิรวิชญ์	กิตติวิชญกุล	53011567

..... อาจารย์ที่ปรึกษา
(ผศ.ดร. นภัทร สระเอี่ยม)

..... อาจารย์ที่ปรึกษาร่วม
(ผศ.ดร.ธเนศ พัฒนธาดาพงษ์)

กิตติกรรมประกาศ

โครงงานนี้สำเร็จลุล่วงลงได้ด้วยความกรุณาจากท่าน ผู้ช่วยศาสตราจารย์ ดร.นภัทร สระเอี่ยม และผู้ช่วยศาสตราจารย์ ดร.ธเนศ พัฒนธาดาพงษ์ ซึ่งเป็นอาจารย์ผู้ควบคุมปริญญานิพนธ์ได้เป็นที่ปรึกษาและแนะนำวิธีแก้ปัญหาต่างๆที่เกิดขึ้นอีกทั้งยังให้ความรู้ในด้านวิชาการและการทำงาน รวมทั้งตักเตือนในสิ่งที่ไม่เหมาะสม คณะผู้จัดทำรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านและขอขอบพระคุณเป็นอย่างสูง

คณะผู้จัดทำโครงงานนี้ขอขอบพระคุณคณาจารย์ สาขาวิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกท่าน ที่ถ่ายทอดความรู้ ความสามารถ ให้คำปรึกษาและแนะนำในสิ่งที่ถูกต้องมาโดยตลาด

คุณค่าและประโยชน์อันพึงมาจากโครงงานฉบับนี้ คณะผู้จัดทำขอมอบแด่ผู้มีพระคุณทุกท่าน

นายนภวิชญ์ คำสมหมาย
นายศิรวิชญ์ กิตติวิชญกุล
นายพิเชฐ แซ่ต่าน

ผู้จัดทำ

ระบบสั่งอาหารผ่านเทคโนโลยี NFC และ QR CODE บนระบบปฏิบัติการแอนดรอยด์และส่งอาหารผ่านหุ่นยนต์อัตโนมัติ
FOOD ORDERING SYSTEM FOR RESTAURANT USING NFC TECHNOLOGY AND QR CODE WITH AUTOMATIC SERVED FOOD VIA ROBOT

โดย นภวิชญ์ คำสมหมาย 53010799
พิเชฐ แซ่ต่าน 53011125
ศิรวิชญ์ กิตติวิชญกุล 53011567

อาจารย์ที่ปรึกษา ผศ.ดร.นภัทร สระเอี่ยม
ผศ.ดร.ธเนศ พัฒนธาดาพงษ์

บทคัดย่อ

โครงงานนี้จะมีการสร้างระบบการสั่งอาหารผ่านแอพพลิเคชั่นบนโทรศัพท์มือถือระบบปฏิบัติการณ์แอนดรอยด์ และยังมีเทคโนโลยี เอ็นเอฟซี ซึ่งเป็นเทคโนโลยีสื่อสารไร้สายระยะสั้น (4 ซม.) ซึ่งรองรับการสื่อสารระหว่างเครื่องมืออิเล็กทรอนิกส์ในระยะใกล้ๆ โดยเอ็นเอฟซี จะใช้คลื่นความถี่ 13.56 MHz และ QR Code ดังนั้น ผู้ดำเนินโครงงาน จึงได้นำเทคโนโลยีทางด้านเอ็นเอฟซีและ QR CODE มาประยุกต์ใช้กับแอพพลิเคชั่น เพื่อสร้างเป็น ระบบสั่งอาหารอัตโนมัติ อีกทั้งยังมีระบบส่งอาหารอัตโนมัติ โดยจะมีหุ่นยนต์อัตโนมัติควบคุมด้วยไมโครคอนโทรลเลอร์ชนิดอาร์ดูโนทำการควบคุมหุ่นยนต์เพื่อไปส่งอาหารได้อย่างถูกต้อง

ABSTRACT

This project create ordering system application on android phone by using "Near field communication (NFC) which is short communication system for communication between smartphone and NFC tag. Wherewith NFC using frequency approximation 13.56 MHz and QR Code. We create sending system food by robot control by AVR microcontroller.

สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	III
สารบัญรูป	VII
สารบัญตาราง	XI
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง	4
2.1 Near Field Communication	4
2.1.1 การประยุกต์ใช้เทคโนโลยี NFC	7
2.1.2 อุปกรณ์ที่รองรับเทคโนโลยี NFC	8
2.2 QR Code	8
2.2.1 ประเภทของ QR Code	9
2.2.2 ประโยชน์ของ QR Code	9
2.3 การพัฒนาแอพพลิเคชั่นที่ทำงานจากระบบปฏิบัติการณ์แอนดรอยด์	10
2.3.1 ชั้นแอพพลิเคชั่น (Application)	10
2.3.2 ชั้นแอพพลิเคชั่นเฟรมเวิร์ค (Application Framework)	11
2.3.3 ชั้นไลบรารี (Library)	12
2.3.4 ชั้น Android Runtime	12
2.3.5 ชั้นลีนุกซ์เคอร์เนล (Linux Kernel)	13

สารบัญ(ต่อ)

	หน้า
2.4 ภาษาจาวา	14
2.4.1 ข้อดีของภาษาจาวา	14
2.5 ภาษา PHP	16
2.6 MySQL	16
2.6.1 สถาปัตยกรรมของ MySQL	17
2.7 วงจร Charger Board	18
2.7.1 วงจร Bridge Rectifier	18
2.7.2 วงจร Voltage Regulators	19
2.7.3 วงจร Voltage Divider: วงจรแบ่งแรงดัน	19
2.8 วงจร Motor Controller Board	20
2.8.1 H-Bridge Circuit	20
2.8.2 การควบคุมความเร็วมอเตอร์	21
2.9 เซ็นเซอร์	21
2.9.1 ความหมายของเซ็นเซอร์	21
2.9.2 คุณสมบัติของเซ็นเซอร์	21
2.10 KEY PAD	23
2.10.1 ความรู้ทั่วไปเกี่ยวกับ Input Logic	23
2.10.2 การต่อวงจรแบบ Switch ขาร่วม	23
2.10.3 การต่อวงจร Switch แบบ Matrix	24
2.11 ไมโครคอนโทรลเลอร์ AVR	26
2.11.1 ประเภทของไมโครคอนโทรลเลอร์ AVR	27
บทที่ 3 การออกแบบและการจัดทำปริญญานิพนธ์	**28**
3.1 การออกแบบ	28
3.1.1 การออกแบบโดยภาพรวม	26
3.1.2 การติดตั้งเครื่องมือจำลองโทรศัพท์ Android และเครื่องมือช่วยเขียนโปรแกรม	29
3.1.3 การออกแบบส่วน Layout ของแอพพลิเคชั่น	30
3.1.4 การออกแบบส่วนไฟล์ควบคุม (.java)	32
3.1.5 การจัดทำ Web Application	35
3.1.6 การเชื่อมต่อ NFC และ QR Code เข้ากับแอพพลิเคชั่น	43
3.1.7 การออกแบบวงจร Charger Board	48

สารบัญ(ต่อ)

	หน้า
3.1.8 วงจร Motor Controller Board	50
3.1.9 วงจร Microcontroller	52
3.2 เครื่องมือที่ใช้ในการทดลอง	56
3.2.1 ส่วนของแอพพลิเคชั่นรับส่งข้อมูลบนมือถือ	56
3.2.2 ส่วนของเซิร์ฟเวอร์	56
3.2.3 ส่วนวงจรของหุ่นยนต์	56
3.2.4 ส่วนโปรแกรมหุ่นยนต์	56
3.2.5 ส่วนอื่นๆ	56
3.3 การจัดเก็บผลการทดลอง	57
3.3.1 การทดสอบแอพพลิเคชั่น	57
3.3.2 การทดสอบหุ่นยนต์	57
บทที่ 4 ผลการทดลอง	**58**
4.1 ผลการทดลองส่วนของ User Interface ในการกรอกข้อมูลลงในฟอร์ม	58
4.2 ผลการทดลองของการติดต่อกับเซิร์ฟเวอร์	60
4.2.1 ผลการทดลองในส่วนการดึงข้อมูลจาก Database โดยใช้ PHP และ JSON	61
4.3 ผลการทดลองส่วนของหน้าจอแสดงผลของพ่อครัวและจัดการฐานข้อมูล	64
4.4 ผลการทดลองส่วนของการติดต่อสื่อสารกับ NFC และ QR-CODE	68
4.5 ผลการทดสอบการชาร์จของค่า Voltage Battery	70
4.6 ผลการทดสอบวงจร Motor Controller Board	72
4.7 ผลการทดสอบการรับค่า ความเข้มสีขาวและสีดำของเซ็นเซอร์อินฟราเรด	76
4.7.1 ผลการทดสอบการจับความเข้มสี สีขาว	76
4.7.2 ผลการทดสอบการจับความเข้มสี สีดำ	77
4.8 ผลการทดสอบการปรับค่าความเร็วของมอเตอร์	78
4.8.1 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 10001	79
4.8.2 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 00011	80
4.8.3 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 00111	81
4.8.4 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 11000	82
4.8.5 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 11100	83
4.8.6 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 00000	84
บทที่ 5 สรุปผลและข้อเสนอแนะ	**85**

สารบัญ(ต่อ)

	หน้า
5.1 สรุปผล	85
5.2 ปัญหาและแนวทางแก้ไข	86
5.3 ข้อเสนอแนะ	87
บรรณานุกรม	88
ภาคผนวก	90

สารบัญรูป

รูปที่	หน้า
2.1 การใช้งานบัตรวีซ่า เพย์เวฟ ที่ไม่ต้องรูด แค่แตะสัมผัส	5
2.2 การใช้มือถือทางานเป็นอาร์เอฟไอดีการ์ด	6
2.3 การสื่อสารในลักษณะเพียร์ทูเพียร์	7
2.4 แสดงลำดับชั้นแอพพลิเคชั่นบนระบบปฏิบัติการ Android	10
2.5 แสดงชั้นแอพพลิเคชั่นเฟรมเวิร์ค (Application Framework)	11
2.6 แสดงชั้นไลบรารี (Library) และ Android Runtime	13
2.7 แสดงชั้นลีนุกซ์เคอร์เนล (Linux Kernel)	13
2.8 แสดง Class ของภาษาจาวา	15
2.9 แสดงการเปลี่ยนสัญญาณไฟสลับเป็นไฟกระแสตรง	18
2.10 แสดงวงจรเรียงกระแสแบบเต็มคลื่น (Full Wave)	19
2.11 แสดง H-Bridge Switching	20
2.12 แสดงตัวอย่างเซ็นเซอร์อินฟราเรด	22
2.13 แสดงตัวอย่างการต่อวงจร Switch แบบขาร่วม	24
2.14 แสดงตัวอย่างการต่อวงจร Switch แบบ Matrix	25
2.15 แสดงตัวอย่างของ Microcontroller AVR ชนิดต่างๆ	27
3.1 Block diagram รวมของระบบ	28
3.2 แสดงโปรแกรม Eclipse พร้อมใช้งานและโทรศัพท์จำลอง	29
3.3 การรันอีมูเลเตอร์จาลองบน Eclipse	30
3.4 หน้าจอแสดงผลของแอพพลิเคชั่น	31
3.5 หน้าจอแสดงผลของแอพพลิเคชั่น(2)	31
3.6 รูปแอพพลิเคชั่น Orderfood	32
3.7 โฟลว์ชาร์ตของแอพพลิเคชั่นรับค่าจาก Database	33
3.8 โฟลว์ชาร์ตของส่วนแอพพลิเคชั่นสั่งอาหาร	34
3.9 แสดงหน้าล็อคอินก่อนจัดการฐานข้อมูล	35
3.10 แสดงตัวแปรต่างๆที่ใช้ในฐานข้อมูล ตารางชื่อ image	36
3.11 แสดงข้อมูลที่อยู่ภายในฐานข้อมูล ตาราง image	36
3.12 แสดงตัวแปรต่างๆที่ใช้ในฐานข้อมูล ตาราง log	37
3.13 แสดงข้อมูลที่อยู่ภายในฐานข้อมูล ตาราง code	37
3.14 แสดงผลการดึงค่ามาแสดงผลยัง Android Client	38

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.15 โฟลว์ชาร์ตของส่วนของการรับค่าจากเซิร์ฟเวอร์มายัง Android Client	39
3.16 แสดงหน้าการกรอกฟอร์มของแอพพลิเคชั่น	40
3.17 แสดงหน้าสรุปการสั่งอาหารของแอพพลิเคชั่น	40
3.18 แสดงหน้าการกรอกฟอร์มของแอพพลิเคชั่น	41
3.19 แสดงหน้าการจัดการโค๊ดส่วนลดราคาอาหาร	41
3.20 โฟลว์ชาร์ตของส่วนของการประมวลผลในเซิร์ฟเวอร์	42
3.21 แสดงหน้าแอพพลิเคชั่น Readnfc และ Tag NFC	43
3.22 โฟลว์ชาร์ตของส่วนการทำงานระหว่าง QR-Code และ Android Client	44
3.23 แสดงหน้าแอพพลิเคชั่น Barcode Scanner	45
3.24 โฟลว์ชาร์ตของส่วนการทำงานระหว่าง QR-Code และ Android Client	46
3.25 โฟลว์ชาร์ตแสดงภาพรวมการทำงานของระบบสั่งอาหารผ่าน Android	47
3.26 แสดง Schematic ของวงจร Charger Board	48
3.27 แสดงการต่อวงจรลงบนบอร์ดทดลอง	48
3.28 แสดงการออกแบบลายวงจร Charger Board	49
3.29 แสดงการต่อวงจร Charger Board ลงบนแผ่น PCB	49
3.30 แสดง Schematic ของวงจร Motor Controller Board	50
3.31 แสดงการออกแบบรายวงจร Motor Controller Board	51
3.32 แสดงการต่อวงจร Motor Controller Board ลงบนแผ่น PCB	51
3.33 แสดง Schematic ของวงจร Microcontroller	53
3.34 แสดง Schematic ของวงจร Microcontroller(2)	54
3.35 แสดง Arduino ATTMEGA 88	55
4.1 หน้า User Interface ส่วนการกรอกฟอร์มของแอพพลิเคชั่น	58
4.2 ผลของการกดปุ่ม Order Food เมื่อท าการส่งข้อมูลเข้าไปยังฐานข้อมูลสำเร็จ	59
4.3 รูปแบบการดึงข้อมูลเข้าสู่โทรศัพท์	60
4.4 ข้อมูลที่จะส่งให้ Android Client	61
4.5 ข้อมูลที่ดึงมาจาก Database แสดงผลแบบ Listview	62
4.6 การแสดงผลเมื่อกดปุ่ม HOW TO ORDER FOOD	63
4.7 การแสดงผลเมื่อกดค้างที่รายการอาหารที่ต้องการ	63
4.8 แสดงรายการอาหารที่ได้ถูกเลือก	64
4.9 แสดงรายการที่สั่งไปทั้งหมดและราคาอาหารที่ต้องจ่ายพร้อมส่วนลด	65

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.10 แสดงรายการที่สั่งไปทั้งหมดและราคาอาหารที่ต้องจ่ายโดยไม่มีส่วนลด	65
4.11 หน้าจอแสดงผลรายการอาหาร ราคาอาหาร ของพ่อครัว	66
4.12 หน้าจอแสดงถึงหน้าการจัดการ Generate Code และ Delete Code	67
4.13 หน้าจอแอพพลิเคชั่นที่ทำการอ่าน QR-CODE และ QR-CODE ที่มี Text = initial	68
4.14 หน้าจอแอพพลิเคชั่น Orderfood และการเชื่อมต่อ Wi-Fi	68
4.15 หน้าจอแอพพลิเคชั่นที่ทำการอ่าน NFC Tag ที่มี Text = password	69
4.16 หน้าจอแอพพลิเคชั่น Orderfood และการเชื่อมต่อ Wi-Fi	69
4.17 แสดงก่อนการชาร์จ ถ่านมีไฟอยู่ที่ 5.3 V	70
4.18 แสดงการชาร์จถ่านมีไฟเพิ่มเป็น 7.2 V โดยมีการสว่างของ LED	70
4.19 ผลการชาร์จไฟ ด้วยการเพิ่มของ Voltage จาก 5.3 V เป็น 9 V ใน 55 นาที	71
4.20 แสดงวงจร Motor Controller Board	72
4.21 แสดงลอจิก 1 (+5V) ที่รับมาจาก Arduino ATTMEGA88	73
4.22 แสดง ค่า V_{BE} ที่ทรานซิสเตอร์ 2N2222	73
4.23 แสดง ค่า V_{BE} ที่ทรานซิสเตอร์ TIP120	74
4.24 แสดง ค่าการไบอัสกลับที่ V_{BE} ของทรานซิสเตอร์ TIP42C	74
4.25 แสดงแรงดันตกคร่อมมอเตอร์ ได้เท่ากับ 8.80 Volt	75
4.26 แสดงการติดสว่างของ LED เมื่อเซ็นเซอร์ทั้ง 5 ตัว มีการจับความเข้มของสีขาวได้	76
4.27 แสดงการดับของ LED เมื่อเซ็นเซอร์ทั้ง 5 ตัว มีการจับความเข้มของสีดำได้	77
4.28 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 10001	79
4.29.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 9 V ทำให้ล้อซ้ายหมุน	79
4.29.ข แสดงมอเตอร์ขวามี ค่า Voltage = 9 V ทำให้ล้อขวาหมุน	79
4.30 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 00011	80
4.31.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 0 V ทำให้ล้อซ้ายหยุดหมุน	80
4.31.ข แสดงมอเตอร์ขวามี ค่า Voltage = 9 V ทำให้ล้อขวาหมุน	80
4.32 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 00111	81
4.33.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 0 V ทำให้ล้อซ้ายหยุดหมุน	81
4.33.ข แสดงมอเตอร์ขวามี ค่า Voltage = 9 V ทำให้ล้อขวาหมุน	81
4.34 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 11000	82
4.35.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 9 V ทำให้ล้อซ้ายหมุน	82
4.35.ข แสดงมอเตอร์ขวามี ค่า Voltage = 0 V ทำให้ล้อขวาหยุดหมุน	82

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.36 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 11100	83
4.37.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 9 V ทำให้ล้อซ้ายหมุน	83
4.37.ข แสดงมอเตอร์ขวามี ค่า Voltage = 0 V ทำให้ล้อขวาหยุดหมุน	83
4.38 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 00000	84
4.39.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 0 V ทำให้ล้อซ้ายหยุดหมุน	84
4.39.ข แสดงมอเตอร์ขวามี ค่า Voltage = 0 V ทำให้ล้อขวาหยุดหมุน	84

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงถึงตารางข้อมูล NFC Platform	4
2.2 แสดงอุปกรณ์ที่รองรับเทคโนโลยี NFC	8
4.1 แสดงสถานการณ์ทำงานของหุ่นยนต์ เมื่อมีการจับค่าความเข้มของสีขาวและสีดำได้	78

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากในปัจจุบันมีผู้ประกอบการร้านอาหาร และร้านอาหารขนาดเล็กไปถึงขนาดใหญ่เกิดขึ้นเป็นจำนวนมาก ส่งผลให้การแข่งขันทางธุรกิจมากขึ้นตามไปด้วย ไม่เพียงแค่รสชาติของอาหารที่ต้องเป็นจุดเด่นของร้านแล้ว ร้านอาหารยังจำเป็นต้องสร้างจุดเด่นให้เหนือกว่าคู่แข่งทางการค้าไม่ว่าจะเป็น ราคาอาหาร บรรยากาศ การบริการ หรือแม้กระทั่งเทคโนโลยีด้านการอำนวยความสะดวกในร้าน เป็นต้น สิ่งเหล่านี้ล้วนเป็นการเพิ่มต้นทุนในการผลิตมากที่มากขึ้นตามไปด้วย

โครงงานนี้จะเป็นส่วนที่จะช่วยแบ่งเบาภาระของร้านอาหาร โดยจะครอบคลุมการบริการ และเป็นเทคโนโลยีในการอำนวยความสะดวกด้วยเช่นกัน ซึ่งจะเป็นอีกหนึ่งทางเลือกในการลดต้นทุนในการผลิตของร้านอาหาร โดยโครงงานนี้จะมีการสร้างระบบการสั่งอาหารผ่านแอพพลิเคชั่นบนโทรศัพท์มือถือระบบปฏิบัติการณ์แอนดรอยด์ และยังมีเทคโนโลยี Near Field Communication (NFC) ซึ่งเป็นเทคโนโลยีสื่อสารไร้สายระยะสั้น (4 cm) ซึ่งรองรับการสื่อสารระหว่างเครื่องมืออิเล็กทรอนิกส์ในระยะใกล้ๆ โดย NFC ใช้คลื่นความถี่ 13.56 MHz บนพื้นฐานมาตรฐาน ISO 14443 ดังนั้น ผู้ดำเนินโครงงาน จึงได้นำเทคโนโลยีทางด้าน NFC มาประยุกต์ใช้กับแอพพลิเคชั่น เพื่อสร้างเป็น ระบบสั่งอาหารอัตโนมัติ อีกทั้งยังมีระบบส่งอาหารอัตโนมัติ โดยจะมีหุ่นยนต์อัตโนมัติควบคุมด้วยไมโครคอนโทรลเลอร์ชนิด AVR ซึ่งเป็นไอซีไมโครคอนโทรลเลอร์ของบริษัท Atmel ทำการควบคุมหุ่นยนต์เพื่อไปส่งอาหารได้อย่างถูกต้อง ซึ่งระบบสั่ง และส่งอาหารอัตโนมัตินี้จะช่วยลดต้นทุนในการผลิตหรือการจ้างพนักงานได้ ซึ่งในร้านอาหารขนาดใหญ่นั้น การจ้างพนักงานบริการเป็นจำนวนมาก เป็นการสร้างต้นทุนที่มากเกินความจำเป็น โดยโครงงานนี้เป็นการทดแทนพนักงานบริการ ได้อย่างมีประสิทธิภาพ

1.2 วัตถุประสงค์

1. เพื่อออกแบบ ระบบส่ง และรับอาหารอัตโนมัติ

2. เพื่อประยุกต์ใช้ เทคโนโลยีทางด้าน NFC และ QR Code กับการสร้างระบบสั่งอาหารอัตโนมัติ

3. เพื่อออกแบบและสร้างแอพพลิเคชั่น บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์เพื่อเลือกเมนูอาหารสำหรับลูกค้า

4. เพื่อออกแบบระบบเซิร์ฟเวอร์และฐานข้อมูล MySQL เพื่อคำนวณค่าอาหารและจัดเก็บข้อมูลลงใน Database พร้อมแสดงผล

5. เพื่อออกแบบ และประกอบวงจร ของหุ่นยนต์ที่สำคัญ อันได้แก่ วงจร Charger Board ซึ่งเป็นส่วนจ่ายไฟเลี้ยงแก่วงจรทั้งหมด วงจร Motor Controller Board ซึ่งเป็นส่วนควบคุมการหมุนของมอเตอร์ และวงจร Microcontroller Board ซึ่งเป็นส่วนในการเขียนคำสั่งควบคุมความเร็ว และทิศทางของหุ่นยนต์

6. เพื่อออกแบบและเขียนโปรแกรม ด้วยภาษา C เพื่อควบคุมเซ็นเซอร์อินฟราเรดและบังคับทิศทางของหุ่นยนต์

7. เพื่อออกแบบโครงสร้างแมคคานิคของหุ่นยนต์ให้สมดุลกับกำลัง และความเร็วของมอเตอร์

8. เพื่อออกแบบ Key pad เพื่อใช้ในการทำเป็นสวิตซ์ เพื่อเลือกเป้าหมายและเส้นทางของหุ่นยนต์

1.3 ขอบเขตของปริญญานิพนธ์

1. ประยุกต์ใช้เทคโนโลยี NFC และ QR CODE บนมือถือระบบปฏิบัติการแอนดรอยด์เพื่อใช้ยืนยันตัวตน เชื่อมต่อ Wi-Fi และเพื่อเข้าใช้งานแอพพลิเคชั่นอัตโนมัติ

2. เขียนแอพพลิเคชั่นเพื่อสั่งอาหาร และส่งข้อมูลอันได้แก่ เลขโต๊ะ เมนูอาหาร และราคา ไปประมวลผลและคำนวณราคาในเซิร์ฟเวอร์ พร้อมแสดงผลได้

3. ใช้คอมพิวเตอร์ในการสร้างระบบเซิร์ฟเวอร์ของร้านอาหาร เพื่อคำนวณราคาอาหารและจัดเก็บข้อมูลลงฐานข้อมูล พร้อมแสดงผลแก่พ่อครัวและลูกค้า

4. สร้างวงจรฮาร์ดแวร์ของหุ่นยนต์ ซึ่งได้แก่ วงจร Charger Board ซึ่งเป็นส่วนจ่ายไฟเลี้ยงแก่วงจรทั้งหมด วงจร Motor Controller Board ซึ่งเป็นส่วนควบคุมทิศทาง การหมุนของมอเตอร์ และวงจร Microcontroller Board ซึ่งเป็นส่วนในการเขียนคำสั่งควบคุมทิศทางของหุ่นยนต์

4. ควบคุมการรับค่าของเซ็นเซอร์อินฟราเรด ในการจับความเข้มของสีขาว และสีดำ และส่งค่าไปประมวลผลที่ Microcontroller

5. ใช้ AVR Microcontroller เบอร์ ATTMEGA88 ในการเขียนโปรแกรมด้วย ภาษา C เพื่อควบคุมการเคลื่อนที่ไปยังจุดหมายของหุ่นยนต์

6. ออกแบบโครงสร้างแมคคานิคของหุ่นยนต์ ให้สมดุลกับกำลังของมอเตอร์ เพื่อสามารถนำอาหารไปส่งลูกค้าได้โดยไม่เสียหาย

7. ออกแบบ Key pad เพื่อใช้ในการทำเป็นสวิตช์เพื่อเลือกเป้าหมาย และเส้นทางของหุ่นยนต์

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

เนื่องจากโครงงานนี้เป็นการพัฒนาซอฟต์แวร์ที่ควบคู่กับฮาร์ดแวร์ที่ใช้งานบนระบบปฏิบัติการแอนดรอยด์ โดยส่วนของซอฟต์แวร์นั้น ประกอบด้วย ส่วนของแอพพลิเคชั่น ส่วนของฐานข้อมูลและส่วนชุดคำสั่งไมโครคอนโทลเลอร์ โดยส่วนซอร์ฟแวร์นั้น มีการสร้างแอพพลิเคชั่นในการสั่งอาหาร และมีการออกแบบ และสร้างฐานข้อมูล เพื่อรองรับการสั่งอาหารจริง อีกทั้งมีการนำเทคโนโลยี Near field Communication (NFC) และ QR-CODE มาใช้ในการยืนยันตัวตนและเชื่อมต่อ Wi-Fi อัตโนมัติ เพื่อเปิดการใช้งานแอพพลิเคชั่นดังกล่าว อีกทั้งยังมีการโปรแกรมคำสั่งไมโครคอนโทลเลอร์เพื่อควบคุมทิศทางของหุ่นยนต์ โดยส่วนฮาร์ดแวร์นั้น มีการออกแบบ และสร้างวงจรการทำงานของหุ่นยนต์ ไม่ว่าจะเป็น ส่วน Charger Board ที่เป็นส่วนการชาร์จไฟ และควบคุมแรงดันไฟเลี้ยงแก่ทุกวงจร วงจร Motor Controller Board ซึ่งเป็นส่วน ควบคุมทิศทางของมอเตอร์ซึ่งจะได้รับ Input เป็น Pulse ในการควบคุมมาจากวงจร Microcontroller Board ดังนั้นจึงต้องศึกษาเกี่ยวกับทฤษฎีพื้นฐานต่างๆให้เข้าใจอย่างถ่องแท้ เพื่อที่จะสามารถนำเอาข้อมูลและความรู้ที่ศึกษานั้นมาใช้ในการพัฒนาโครงงานได้อย่างถูกต้อง

2.1 Near field Communication

Near Field Communication (NFC) เป็นการผสมผสานกันระหว่างเทคโนโลยีการส่งข้อมูลแบบไร้สาย และเทคโนโลยีการเชื่อมต่อกันระหว่างอุปกรณ์ เทคโนโลยีไร้สายที่นำมาใช้ใน NFC คือเทคโนโลยีอาร์เอฟไอดี โดยใช้คลื่นความถี่ 13.56 MHz. ซึ่งเทคโนโลยี NFC จะรองรับกับเทคโนโลยีอาร์เอฟไอดีตามรายละเอียดด้านล่าง

ตารางที่ 2.1 แสดงถึงตารางข้อมูล NFC Platform

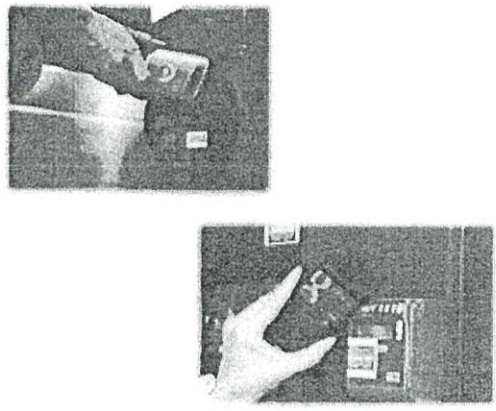
NFC Forum Platform	RFID Compatible
NFC Forum Type 1 tag	Innovision Topaz
NFC Forum Type 2 tag	Mifare Ultralight Mifare Ultralight C
NFC Forum Type 3 tag	Sony Felica
NFC Forum Type 4 tag	DESfire SmartMX

เทคโนโลยีเอ็นเอฟซีนี้ช่วยให้การสื่อสารระหว่างอุปกรณ์สามารถทำได้ง่ายขึ้น ต่างจากเทคโนโลยีไร้สายประเภทอื่น ได้แก่ ไวไฟ (Wi-Fi) หรือ บลูทูธ (Bluetooth) ที่ต้องมีตั้งค่าต่าง ๆ ก่อนการใช้งานแต่เทคโนโลยีเอ็นเอฟซีเพียงแค่นำอุปกรณ์มือถือ เช่น โทรศัพท์ ไปใกล้กับเครื่องอ่านหรืออาร์เอฟไอดีการ์ด ก็สามารถที่จะทำการส่งข้อมูลระหว่างกันได้เลย โดยไม่จำเป็นต้องมีการตั้งค่าใดๆ ก่อนการใช้งาน การใช้งานเทคโนโลยีเอ็นเอฟซีมีได้ 3 ลักษณะ

1. ทำงานเป็นอาร์เอฟไอดีการ์ด (RFID Tag) ในอุปกรณ์มือถือ หมายความว่าอุปกรณ์มือถือตามมาตรฐานเอ็นเอฟซี จะทำตัวเป็นบัตรในรูปแบบใดก็ได้ตามมาตรฐาน ISO 14443 เพื่อใช้ในงานทางธุรกรรม โทรศัพท์มือถือที่มีเครื่องอ่านเอ็นเอฟซี ฝังอยู่สามารถทำงานเป็นการ์ดอาร์เอฟไอดีได้ ซึ่งต่างจากเครื่องอ่านอาร์เอฟไอดีในปัจจุบันที่ทำหน้าที่เป็นเครื่องอ่านเพียงอย่างเดียว

รูปที่ 2.1 การใช้งานบัตรวีซ่า เพย์เวฟ ที่ไม่ต้องรูด แค่แตะสัมผัส

การทำงานในลักษณะนี้ส่วนใหญ่จะใช้ในแอพพลิเคชั่นในเรื่องการเงิน เช่น การจ่ายเงินชำระค่าผ่านทาง ต่าง ๆ เพียงแค่นำโทรศัพท์มือถือไปใกล้กับเครื่องอ่านอาร์เอฟไอดีที่ติดตั้งไว้ที่จุดชำระเงิน ก็สามารถทำการชำระเงินได้ แทนการชำระเงินด้วยบัตรอาร์เอฟไอดีหรือเงินสด นอกจากนั้นเราสามารถขยายการใช้งานอุปกรณ์เอ็นเอฟซีชิ้นเดียวเป็นบัตรหลายใบได้ เช่น เป็นบัตรเครดิต บัตรโดยสารรถไฟฟ้า บัตรเงินสด บัตรสะสมแต้ม เป็นต้น ทำให้เมื่อเอ็นเอฟซีเป็นที่นิยมและผู้ให้บริการบัตรต่างๆทำแอพพลิเคชั่น สำหรับบริการของตนลงบนอุปกรณ์ เราจะได้ไม่ต้องพกบัตรหลายใบ



รูปที่ 2.2 การใช้มือถือทำงานเป็นอาร์เอฟไอดีการ์ด

2. ทำงานเป็นเครื่องอ่านอาร์เอฟไอดี อุปกรณ์มือถือ เช่น โทรศัพท์มือถือที่มีเครื่องอ่านเอ็นเอฟซี ฝังอยู่ สามารถทำงานเป็นเครื่องอ่านอาร์เอฟไอดี ดังรูปที่ 2.8 เมื่อต้องการอ่านข้อมูลจากอาร์เอฟไอดีการ์ด ในโหมดนี้อุปกรณ์เอ็นเอฟซี สามารถทำตัวเสมือนเป็นเครื่องอ่านเขียนแท็ก โดยจะสามารถอ่านข้อมูลจากแท็ก ที่ติดอยู่ในจุดให้บริการข้อมูล โดยที่จุดให้บริการข้อมูลจะมีอาร์เอฟไอดีสติกเกอร์ (RFID Sticker) เมื่อต้องการอ่านข้อมูลจากอาร์เอฟไอดีสติกเกอร์เพียงแค่นำโทรศัพท์มือถือที่มีเครื่องอ่านไปอ่านอาร์เอฟไอดีสติกเกอร์ที่จุดให้บริการ ข้อมูลข้างในก็จะปรากฏขึ้นมาบนโทรศัพท์มือถือ

3. การสื่อสารในลักษณะเพียร์ทูเพียร์ เครื่องอ่านเอ็นเอฟซี สองเครื่องสามารถที่จะติดต่อ สื่อสารกันโดยตรงได้ ดังรูปที่ 2.9 โหมดนี้จะทำการแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์เอ็นเอฟซี ด้วยกัน คล้ายกับการที่มือถือมีบลูทูธ แล้วทำการแลกเปลี่ยนข้อมูลโดยการจับคู่ (Pair) เข้าด้วยกันแล้วแลกเปลี่ยนข้อมูล เช่น นามบัตร รูปถ่าย แฟ้มข้อมูลอื่นๆ แต่ว่าสำหรับเอ็นเอฟซีแล้ว ไม่ต้องมีการจับคู่เหมือนบลูทูธ เพียงแค่เลือกข้อมูลที่ต้องการแลกเปลี่ยนแล้วนำอุปกรณ์เอ็นเอฟซี ที่รองรับโหมดนี้มาแตะกัน ข้อมูลก็จะทำการโอนถ่ายกันระหว่างเครื่อง เพราะรัศมีทำการของเอ็นเอฟซี อยู่ในระดับน้อยกว่า 10 ซ.ม. ซึ่งต่างจากบลูทูธ ซึ่งออกแบบไว้ให้สื่อสารข้อมูลในระยะหลายเมตร การแลกเปลี่ยนข้อมูลทำได้ผ่านโปรโตคอลทีซีพี/ไอพี (TCP/IP) นอกจากแลกเปลี่ยนข้อมูลแล้วยังสามารถใช้ทำการเข้าจังหวะ ข้อมูลกับอุปกรณ์อื่นๆได้ด้วย ตัวอย่างเช่น โทรศัพท์มือถือสองเครื่องที่มีฟังก์ชันเอ็นเอฟซี สามารถที่จะส่งข้อมูลให้แก่กันได้โดยตรง เพียงนำโทรศัพท์ทั้งสองเครื่อง เข้ามาใกล้กันในระยะที่เครื่องอ่านที่อยู่ในโทรศัพท์ทั้งสองสามารถอ่านกันได้ก็สามารถที่จะสั่งข้อมูลถึงกันได้ โดยไม่ต้องผ่านเครือข่ายของโทรศัพท์มือถือ ไม่ว่าจะเป็นจีพีอาร์เอส (GPRS) หรือเอดจ์ (EDGE) เป็นต้น

รูปที่ 2.3 การสื่อสารในลักษณะเพียร์ทูเพียร์

ถึงแม้ว่าเทคโนโลยีเอ็นเอฟซี จะได้รับการกล่าวถึงเป็นอย่างมากในปัจจุบัน แต่อุปกรณ์มือถือที่รองรับเทคโนโลยีเอ็นเอฟซี ยังมีอยู่อย่างจำกัด ดังนั้นในช่วงการเปลี่ยนผ่านจึงมีการพัฒนาเทคโนโลยีต่าง ๆ ขึ้นมาเพื่อรองรับกับเทคโนโลยีเอ็นเอฟซี ที่กำลังจะเกิดขึ้น ตัวอย่างเช่น เทคโนโลยีซิมการ์ด + เสาอากาศ เป็นการต่อเสาอากาศเพิ่มจากซิมการ์ดของโทรศัพท์มือถือ ซึ่งเทคโนโลยีเหล่านี้ใช้อาร์เอฟไอดีคลื่นความถี่ 13.56 เมกะเฮิรตซ์ เหมือนกับเทคโนโลยีเอ็นเอฟซี เทคโนโลยีนี้มีการใช้อย่างแพร่หลายในประเทศจีน

ก่อนการใช้งานเทคโนโลยี NFC เพียงแค่นำอุปกรณ์มือถือ เช่น โทรศัพท์ไปใกล้กับเครื่องอ่านหรืออาร์เอฟไอดีการ์ด (RFID tag) ก็สามารถที่จะทำการส่งข้อมูลระหว่างกันได้เลย โดยไม่จำเป็นต้องมีการตั้งค่าใดใดก่อนการใช้งาน

2.1.1 การประยุกต์ใช้เทคโนโลยี NFC

2.1.1.1 ทำงานเป็นอาร์เอฟไอดีการ์ด (RFID Tag) อุปกรณ์มือถือ เช่น โทรศัพท์มือถือที่มีเครื่องอ่าน NFC ฝังอยู่สามารถทำงานเป็น RFID Tag ได้ ซึ่งต่างจากเครื่องอ่านอาร์เอฟไอดีในปัจจุบันที่ทำหน้าที่เป็นเครื่องอ่านเพียงอย่างเดียว การทำงานในลักษณะนี้ส่วนใหญ่จะใช้ใน Application ในเรื่องการเงิน เช่น การจ่ายเงินชำระค่าผ่านทาง การจ่ายเงินตาม POST ต่าง ๆ เพียงแค่นำโทรศัพท์มือถือไปใกล้กับเครื่องอ่านอาร์เอฟไอดี ที่ติดตั้งไว้ที่จุดชำระเงิน ก็สามารถทำการชำระเงินได้ แทนการชำระเงินด้วยบัตรอาร์เอฟไอดีหรือเงินสด

2.1.1.2 ทำงานเป็นเครื่องอ่านอาร์เอฟไอดีกับอุปกรณ์มือถือ เช่น โทรศัพท์มือถือที่มีเครื่องอ่าน NFC ฝังอยู่ สามารถทำงานเป็นเครื่องอ่านอาร์เอฟไอดี เมื่อต้องการอ่านข้อมูลจากอาร์เอฟไอดีการ์ด (RFID Tag) เช่น การใช้งานในลักษณะ Smart Posterเป็นต้น

โดยที่โปสเตอร์จะมีอาร์เอฟไอดีสติกเกอร์ (RFID Sticker) เมื่อต้องการอ่านข้อมูลจาก RFID Sticker เพียงแค่นำโทรศัพท์มือถือที่มีเครื่องอ่านไปอ่านอาร์เอฟไอดีสติกเกอร์บนโปสเตอร์ ข้อมูลในโปสเตอร์ก็จะปรากฏขึ้นมาบนโทรศัพท์มือถือ

2.1.1.3 การสื่อสารในลักษณะ Pier to Pier (P2P) เครื่องอ่าน NFC สองเครื่องสามารถที่จะติดต่อสื่อสารกันโดยตรงได้เมื่อต้องการส่งข้อมูล ตัวอย่างเช่น โทรศัพท์มือถือสองเครื่องที่มีฟังก์ชัน NFC สามารถที่จะส่งข้อมูลให้แก่กันได้โดยตรง เพียงนำโทรศัพท์ทั้งสองเครื่องเข้ามาใกล้กันในระยะที่เครื่องอ่านที่อยู่ในโทรศัพท์ทั้งสองสามารถอ่านกันได้ ก็สามารถที่จะส่งข้อมูลถึงกันได้ โดยไม่ต้องผ่านเครือข่ายของโทรศัพท์มือถือไม่ว่าจะเป็น GPRS หรือ EDGE เป็นต้น

ถึงแม้ว่าเทคโนโลยี NFC จะได้รับการกล่าวถึงเป็นอย่างมากในปัจจุบัน แต่อุปกรณ์มือถือที่รองรับเทคโนโลยี NFC ยังมีอยู่อย่างจำกัด ดังนั้นในช่วงการเปลี่ยนผ่านจึงมีการพัฒนาเทคโนโลยีต่าง ๆ ขึ้นมาเพื่อรองรับกับเทคโนโลยี NFC ที่กำลังจะเกิดขึ้นตัวอย่างเช่น เทคโนโลยี SIM Card + เสาอากาศ เป็นการต่อเสาอากาศเพิ่มจาก SIM Card ของโทรศัพท์มือถือ

2.1.2 อุปกรณ์ที่รองรับเทคโนโลยีNFC

อุปกรณ์ที่รองรับเทคโนโลยี NFC ได้แก่มือถือค่ายหลักๆ ทั้ง Nokia และ Samsung เป็นต้น ยกตัวอย่างเช่น

ตารางที่ 2.2 แสดงอุปกรณ์ที่รองรับเทคโนโลยี NFC

Nokia	Samsung
Nokia Lumia 620, Nokia 700 ,Nokia 720	Samsung SGH-X700 NFC ,Samsung D500E
Nokia 701 ,Nokia 603 ,Nokia 6212	Samsung Galaxy Ace 2 i8160
Nokia 6131 ,Nokia 3220 + NFC Shell	Samsung Galaxy Nexus
	Samsung Galaxy SIII
	Samsung Note II , Samsung Note III

2.2 QR Code

QR Code คือ รหัสสองมิติ (two-dimensional code) ที่สามารถบันทึกข้อมูล เช่น ตัวหนังสือ ตัวเลข ภาพต่างๆ โดยรหัสถูกจัดไว้ในรูปแบบสี่เหลี่ยมจัตุรัส QR Code สามารถอ่านข้อมูลได้รวดเร็วกว่าและใช้งานง่ายกว่ารหัสแท่ง (Bar Code) ที่ใช้อยู่ในปัจจุบัน การเก็บข้อมูลก็สามารถเก็บข้อมูลได้มากกว่า Bar Code ในขณะที่ Bar Code (one-dimensional bar code) สามารถเก็บข้อมูลได้ประมาณ 20 อักขระ QR Code สามารถเก็บข้อมูลตัวเลข ตัวอักษร (ในรูป ASCII) 2,953 binary bytes เก็บอักขระ Kanji ได้สูงสุด 1,817 อักขระรหัสแท่ง (Bar Code) โดยมากใช้ในวงจำกัดของงาน Logistics สำหรับการคำนวณ การสำรวจสินค้า เป็นต้น แต่

QR Code นำมาใช้ในวงกว้าง นำมาใช้ในการสื่อสารข้อมูล การตลาด การประชาสัมพันธ์ และการทำธุรกิจ นามบัตร (card making)

บริษัท Denso Wave ของญี่ปุ่น ได้ประดิษฐ์ QR Code ขึ้นมาใช้ใน ปี ค.ศ. 1994 และได้ขึ้นทะเบียนลิขสิทธิ์ชื่อ QR Code และเป็นรหัสมาตรฐานสากลของ ISO ในปี ค.ศ. 2000 เปิดให้ใช้ทั้งประเทศญี่ปุ่นและทั่วโลกและมีการนำมาใช้ในวงการต่างๆ เช่น วารสารธุรกิจ ขายสินค้า หรือขนส่ง เพื่อการโฆษณา ประชาสัมพันธ์ และการสื่อสารในระบบออนไลน์

โทรศัพท์มือถือที่จะสามารถอ่าน QR Code ต้องมีคุณสมบัติ ดังนี้

1. มีกล้องถ่ายรูปได้
2. มีโปรแกรมอ่าน QR Code (โปรแกรมอ่านรหัส QR สามารถโหลดมาติดตั้งได้ฟรี เช่น The Kaywa Reader, QuickMark)
3. ใช้อินเทอร์เน็ต
4. โทรศัพท์จะต้องมีระบบปฏิบัติการ Window mobile หรือ Symbian Version 6 ขึ้นไป หรือระบบ Java แต่ถ้าเป็นโทรศัพท์รุ่นใหม่ เช่น IPhone สามารถใช้ได้เลย

2.2.1 ประเภทของ QR Code

1. แบบ Real-Time ก็คือ แค่ใช้กล้องในโทรศัพท์มือถือส่อง QR Code โปรแกรมก็จะแปลงสัญลักษณ์นั้นเป็นข้อมูลให้ได้ทันที
2. แบบ Snapshot/Capture คือ ต้องเปิดโปรแกรมแล้วถ่ายภาพจากกล้องมือถือ โดยถ่ายภาพ QR Code ก่อน แล้วจึงประมวล Code นั้นออกมา สัญลักษณ์สี่เหลี่ยมเหล่านี้เริ่มเห็นแพร่หลายในบ้านเรามากขึ้นเรื่อยๆ ไม่ว่าจะเป็นจากหนังสือพิมพ์ จากนิตยสาร หรือเว็บไซต์ต่าง

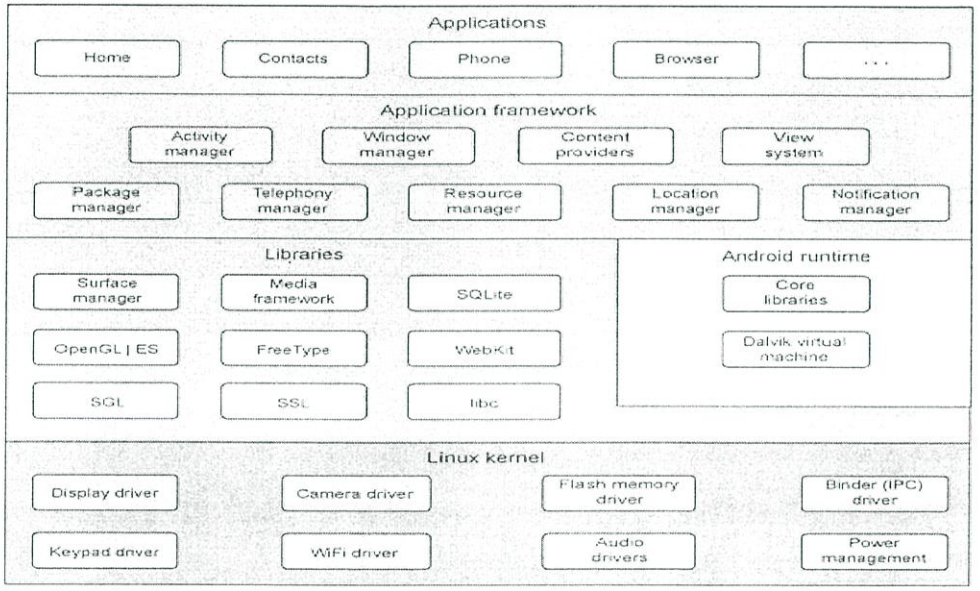
2.2.2 ประโยชน์ของ QR Code

เราสามารถนำ QR Code มาประยุกต์ใช้ได้หลากหลายรูปแบบ เช่น แสดง URL ของเว็บไซต์,ข้อความ,เบอร์โทรศัพท์ และข้อมูลที่เป็นตัวอักษร ได้อีกมากมาย ปัจจุบัน QR Code ถูกนำไปใช้ในหลายๆ ด้าน เพราะสะดวกรวดเร็วอีกทั้งทุกวันนี้ผู้คนส่วนใหญ่จะมีมือถือกันทุกคนและมือถือ ก็มีกล้องเกือบทุกรุ่นเช่นกัน ประโยชน์ที่เห็นได้ชัดที่สุดของ QR Code คือการแสดง URL ของเว็บไซต์ เพราะ URL โดยปกติ แล้วจะจดจำยากเนื่องจากมีความยาว และบางทีก็จะซับซ้อนมาก แต่ด้วย QR Code เราเพียงแค่ยกมือถือมาสแกน QR Code ที่เราพบเห็นตามผลิตภัณฑ์ต่างๆ, นามบัตร, นิตยสาร ฯลฯ มือถือจะลิ้งค์เข้าเว็บเพจที่ QR Code นั้นๆ บันทึกข้อมูลอยู่โดยอัตโนมัติ AIS เป็นผู้ให้บริการโทรศัพท์มือถือที่นำ เทคโนโลยี QR Code มาใช้ เพื่อให้ผู้ใช้บริการสามารถเข้าถึงข้อมูลสำคัญได้อย่างรวดเร็ว อีกทั้งยังสามารถสร้าง QR Code ไว้ใช้งานส่วนตัวได้อย่างง่ายๆ เช่นกัน

2.3 การพัฒนาแอพพลิเคชั่นที่ทำงานจากระบบปฏิบัติการณ์แอนดรอยด์

แอนดรอยด์เป็นซอฟต์แวร์ที่มีโครงสร้างแบบเรียงทับซ้อนหรือแบบสแต็ก (Stack) ซึ่งรวมเอาระบบปฏิบัติการ (Operating System), มิดเดิลแวร์ (Middleware) และแอพพลิเคชันที่สำคัญเข้าไว้ด้วยกัน เพื่อใช้สำหรับทำงานบนอุปกรณ์พกพาเคลื่อนที่ (Mobile Devices) เช่น โทรศัพท์มือถือ เป็นต้น

การทำงานของแอนดรอยด์มีพื้นฐานอยู่บนระบบลีนุกซ์ เคอร์เนล (Linux Kernel) ซึ่งใช้ Android SDK (Software Development Kit) เป็นเครื่องมือสำหรับการพัฒนาแอพพลิเคชันบนระบบปฏิบัติการ Android และใช้ภาษา Java ในการพัฒนา สถาปัตยกรรมของแอนดรอยด์ (Android Architecture) นั้นถูกแบ่งออกเป็นลำดับชั้น ออกเป็น 4 ชั้นหลัก



รูปที่ 2.4 แสดงลำดับชั้นแอพพลิเคชั่นบนระบบปฏิบัติการ Android

2.3.1 ชั้นแอพพลิเคชั่น (Application)

ชั้นนี้จะเป็นชั้นที่อยู่บนสุดของโครงสร้างสถาปัตยกรรม Android ซึ่งเป็นส่วนของแอพพลิเคชั่นที่พัฒนาขึ้นมาใช้งาน เช่น แอพพลิเคชั่น รับ / ส่งอีเมล์ , SMS , ปฏิทิน , แผนที่ , เว็บบราวเซอร์, รายชื่อผู้ติดต่อ เป็นต้น ซึ่งแอพพลิเคชั่นจะอยู่ในรูปแบบของไฟล์ .apk โดยทั่วไปแล้วจะอยู่ในไดเร็คทอรี data/app

2.3.2 ชั้นแอพพลิเคชั่นเฟรมเวิร์ค (Application Framework)

ในชั้นนี้จะอนุญาตให้นักพัฒนาสามารถเข้าเรียกใช้งาน โดยผ่าน API (Application Programming Interface) ซึ่ง Android ได้ออกแบบไว้เพื่อลดความซ้ำซ้อนในการใช้งาน application component โดยในชั้นนี้ประกอบด้วยแอพพลิเคชั่นเฟรมเวิร์คดังนี้

2.3.2.1 View System เป็นส่วนที่ใช้ในการควบคุมการทำงานสำหรับการสร้างแอพพลิเคชั่น เช่น lists , grids , text boxes , buttons และ embeddable web browser

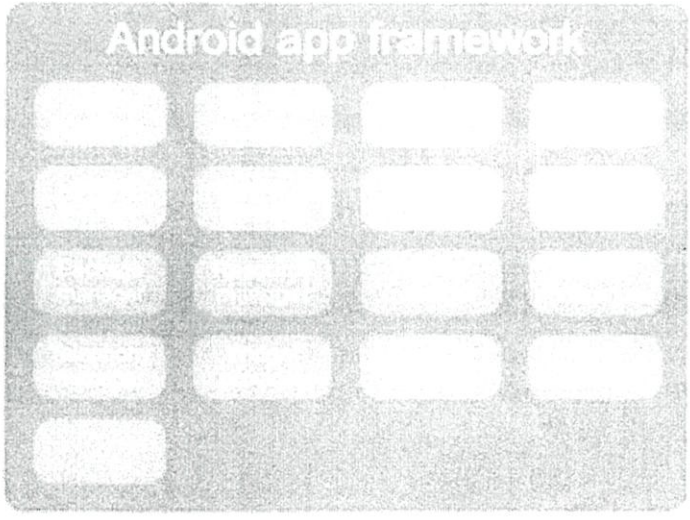
2.3.2.2 Location Manager เป็นส่วนที่จัดการเกี่ยวกับค่าตำแหน่งของเครื่องอุปกรณ์พกพาเคลื่อนที่

2.3.2.3 Content Provider เป็นส่วนที่ใช้ควบคุมการเข้าถึงข้อมูลที่มีการใช้งานร่วมกัน (Share data) ระหว่างแอพพลิเคชั่นที่แตกต่างกัน เช่น ข้อมูลผู้ติดต่อ (Contact)

2.3.2.4 Resource Manager เป็นส่วนที่จัดการข้อมูลต่างๆ ที่ไม่ใช่ส่วนของโค้ดโปรแกรม เช่น รูปภาพ, localized strings, layout ซึ่งจะอยู่ในไดเร็คทอรี res/

2.3.2.5 Notification Manager เป็นส่วนที่ควบคุมอีเว้นต์ (Event) ต่างๆ ที่แสดงบนแถบสถานะ (Status bar) เช่น ในกรณีที่ได้รับข้อความหรือสายที่ไม่ได้รับและการแจ้งเตือนอื่นๆ เป็นต้น

2.3.2.6 Activity Manager เป็นส่วนควบคุม Life Cycle ของแอพพลิเคชั่น



รูปที่ 2.5 แสดงชั้นแอพพลิเคชั่นเฟรมเวิร์ค (Application Framework)

2.3.3 ชั้นไลบรารี (Library)

Android ได้รวบรวมกลุ่มของไลบรารีต่างๆ ที่สำคัญและมีความจำเป็นเอาไว้มากมาย เพื่ออำนวยความสะดวกให้กับนักพัฒนาและง่ายต่อการพัฒนาโปรแกรม โดยตัวอย่างของไลบรารีที่สำคัญเช่น

2.3.3.1 System C library เป็นกลุ่มของไลบรารีมาตรฐานที่อยู่บนพื้นฐานของภาษา C library สำหรับ embedded system ที่มีพื้นฐานมาจาก Linux

2.3.3.2 Media Libraries เป็นกลุ่มการทำงานมัลติมีเดีย เช่น MPEG4 , H.264 , MP3 , AAC , AMR , JPG , และ PNG

2.3.3.3 Surface Manager เป็นกลุ่มการจัดการรูปแบบหน้าจอ การวาดหน้าจอ

2.3.3.4 2D / 3D library เป็นกลุ่มของกราฟิกแบบ 2 มิติ หรือ SGL (Scalable Graphics Library) และแบบ 3 มิติ หรือ OpenGL

2.3.3.5 FreeType เป็นกลุ่มของบิตแมป (Bitmap) และเวคเตอร์ (Vector) สำหรับการเรนเดอร์ (Render) ภาพ

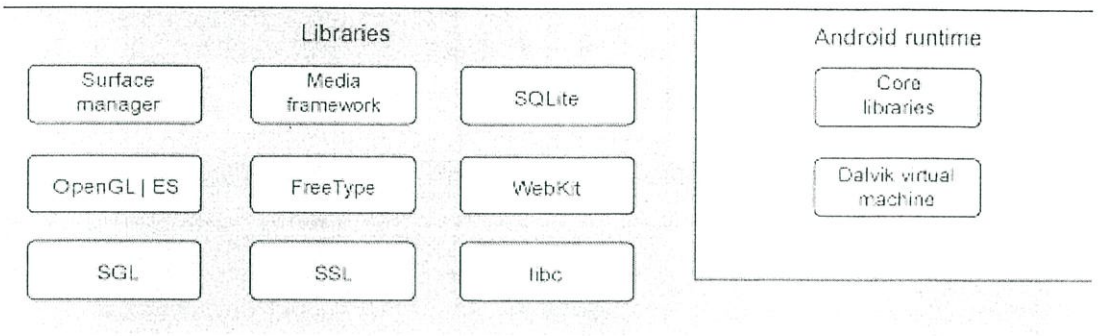
2.3.3.6 SQLite เป็นกลุ่มของฐานข้อมูล โดยนักพัฒนาสามารถใช้ฐานข้อมูลนี้เก็บข้อมูลแอพพลิเคชั่นต่างๆ ได้

2.3.3.7 Browser Engine เป็นกลุ่มของการแสดงผลบนเว็บบราวเซอร์ โดยอยู่บนพื้นฐานของ Webkit ซึ่งจะมีลักษณะคล้ายกับ Google Chrome

2.3.4 Android Runtime

เป็นชั้นย่อยที่อยู่ในชั้นไลบรารี ซึ่งจะประกอบด้วย 2 ส่วนหลักคืออีกส่วนหนึ่งที่ทำงานอยู่บนเคอร์เนลของลินุกซ์ก็คือ Android Runtime ซึ่งประกอบด้วย Core Library สำหรับภาษาจาวา และ Dalvik Virtual Machine ซึ่งก็คือ Java Virtual Machine ในแบบของแอนดรอยด์เอง ซึ่งถูกออกแบบมาสำหรับอุปกรณ์ที่ใช้ไฟฟ้าจากแบตเตอรี่และมีหน่วยความจำจำกัด ในแอนดรอยด์นั้นแต่ละแอพพลิเคชั่นจะรันอยู่ในโพรเซสของตัวเอง และมี Dalvik VM ของตัวเองอยู่ด้วย ดังนั้นโค้ยดของแต่ละแอพพลิเคชั่นจึงรันอยู่ใน VM ที่แยกจากกัน

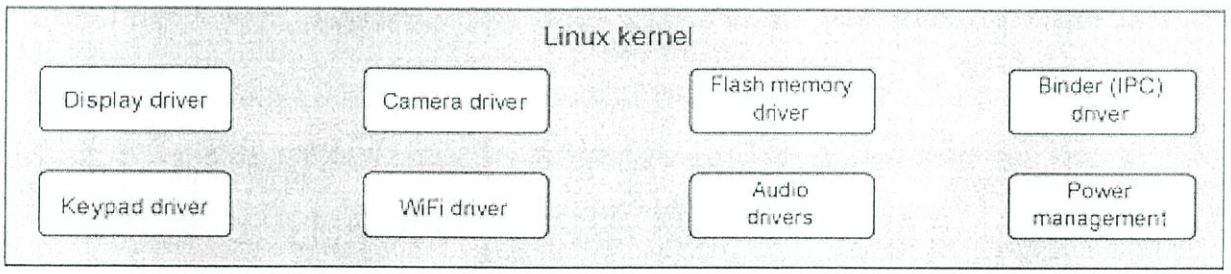
สำหรับ Core Library ซึ่งเป็น Java Library นั้นส่วนใหญ่จะเหมือนกับใน Java Standard Edition (Java SE) ที่เราใช้พัฒนาจาวาแอพพลิเคชั่นบนพีซี แต่บางไลบรารีที่มีใน Java SE จะไม่มีในแอนดรอยด์และบางไลบรารีถึงแม้จะมีในแอนดรอยด์แต่ก็มีรูปแบบการใช้งานที่แตกต่างออกไป



รูปที่ 2.6 แสดง ชั้นไลบรารี (Library) และ Android Runtime

2.3.5 ชั้นลีนุกซ์เคอร์เนล (Linux Kernel)

ระบบ Android นั้นถูกสร้างบนพื้นฐานของระบบปฏิบัติการ Linux โดยในชั้นนี้จะมีฟังก์ชันการทำงานหลายๆ ส่วน แต่โดยส่วนมากแล้วจะเกี่ยวข้องกับฮาร์ดแวร์โดยตรง เช่น การจัดการหน่วยความจำ (Memory Management) การจัดการโพรเซส (Process Management) การเชื่อมต่อเครือข่าย (Networking) เป็นต้น



รูปที่ 2.7 แสดง ชั้นลีนุกซ์เคอร์เนล (Linux Kernel)

2.4 ภาษาจาวา

ในโลกของการเขียนโปรแกรม มีภาษามากมายให้เราเลือกศึกษาและนำมาใช้ โดยแต่ละภาษานั้นก็มีจุดเด่นจุดด้อยที่แตกต่างกันไป และภาษาหนึ่งที่จัดว่าได้รับนิยมอย่างมาก นั่นก็คือภาษาจาวา (Java) ซึ่งนับว่าเป็นภาษาที่ค่อนข้างนิยมในยุคนี้ ภาษาจาวานั้นพัฒนาโดย Sun Microsystems ในปี 1991 ด้วยแนวคิดที่ต้องการให้ภาษานี้ใช้งานง่าย สะดวกต่อการพัฒนา และสามารถใช้บนอุปกรณ์ใดก็ได้โดยไม่ยึดติดกับแพลตฟอร์มใดแพลตฟอร์มหนึ่ง ทำให้ภาษานี้กลายเป็นภาษาที่มีความโดดเด่นและถูกใช้งานอย่างแพร่หลาย ไม่ว่าจะเป็นการพัฒนาโปรแกรมอรรถประโยชน์ต่างๆ หรือแม้กระทั่งโปรแกรมขนาดใหญ่ รวมไปถึงในอุปกรณ์ระบบสมองกลฝังตัวต่างๆ (Embedded System) เช่นโทรศัพท์มือถือ หรือคอมพิวเตอร์พกพาต่างๆ ก็มีการใช้ภาษาจาวาในการพัฒนาแอพพลิเคชั่นให้เห็นกันทั่วไป

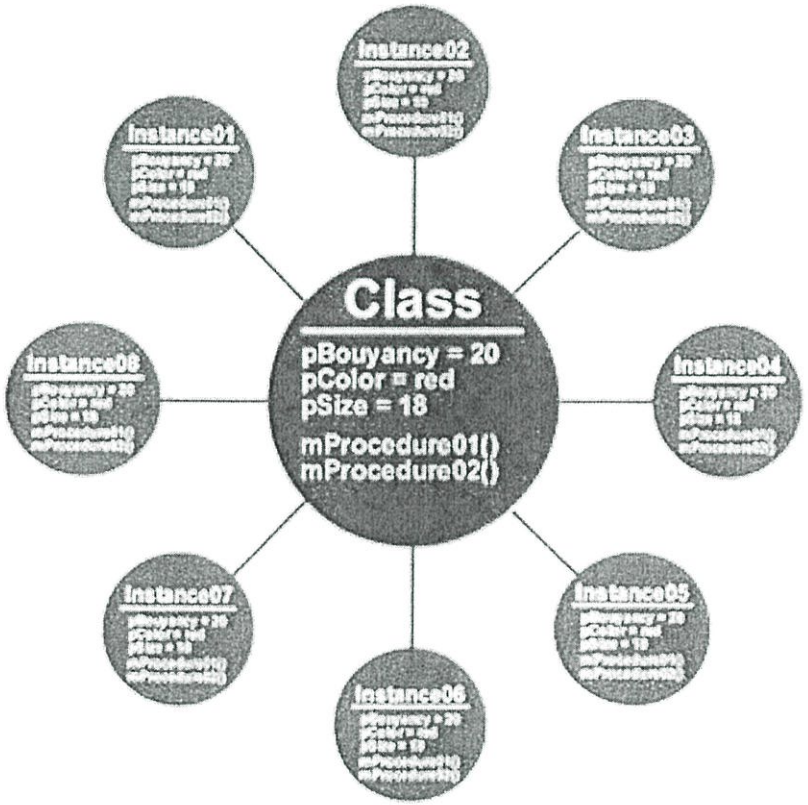
ภาษาจาวา ถูกคิดค้นโดย James Gosling และคณะ จากบริษัท Sun Microsystems โดยมีวัตถุประสงค์เดิม คือ ใช้สำหรับการเขียนโปรแกรมเพื่อฝังตัวในอุปกรณ์อิเล็กทรอนิกส์ ผลที่ได้ คือ ภาษาสำหรับเขียนโปรแกรม ซึ่งเป็นลักษณะของโปรแกรมเชิงวัตถุ (Object-Oriented Programming) ซึ่งสามารถใช้งานบนเว็บได้ด้วย โดยเทคโนโลยีจาวานั้น มีองค์ประกอบที่สำคัญอยู่สองอย่างก็คือ ภาษาจาวา ซึ่งเป็นภาษาที่ใช้ในการเขียนและพัฒนาโปรแกรม และอีกอย่างก็คือ Java platform คือแพลตฟอร์มหรือสภาพแวดล้อมที่ใช้ในการรันโปรแกรมจาวา โดยโปรแกรมจาวาจะทำงานบน Java platform เท่านั้น Java platform จะประกอบไปด้วยสองอย่าง คือ Java VM (JVM) และ Runtime library โปรแกรมจาวาที่เราเขียนขึ้นจะทำงานบนแพลตฟอร์มใดก็ได้ที่มี Java platform ทำงานอยู่

2.4.1 ข้อดีของภาษาจาวา

2.4.1.1 ภาษา Java เป็นภาษาโปรแกรมที่ง่ายในการเรียนรู้ ภาษา Java มีคุณลักษณะต่างๆ ดังนี้ เช่น เชื่อมต่อข้ามแพล็ตฟอร์ม (Platforms) ต่างๆ ได้ สามารถเขียนโปรแกรมแบบ OOP (Object-Oriented Programming) ได้ง่ายมาก โปรแกรมมีขนาดเล็ก และมีวิธีการเขียนไม่ยุ่งยากซับซ้อน ดังนั้นโปรแกรมที่เขียนด้วยภาษา Java จึงคอมไพล์ได้ง่ายตลอดจนตรวจสอบหาข้อผิดพลาดโปรแกรมได้ง่ายด้วย ภาษา java เป็นภาษาที่ทำความเข้าใจได้ง่ายมาก มีขนาดเล็กและยากที่จะเกิดข้อผิดพลาด เขียนคำสั่งได้ง่าย มีประสิทธิภาพในการทำงานและมีความยืดหยุ่นสูง

2.4.1.2 ภาษา Java เป็นการเขียนโปรแกรมเชิงวัตถุ OOP (Object-Oriented Programming) การเขียนโปรแกรมเชิงวัตถุ เป็นเทคนิคการเขียนโปรแกรมให้มีลักษณะเป็นโมดูล (Module) แบ่งโปรแกรมเป็นส่วนๆ ตามสภาวะแวดล้อมการทำงานของโปรแกรมซึ่งเรียกว่า Method โดยทุก Method ก็คือ ระเบียบวิธี หรือการทำงานอย่างใดอย่างหนึ่ง โดยจะถูกรวบรวมอยู่ในคลาส ซึ่งหลักการเขียนโปรแกรมเชิงวัตถุจะมององค์ประกอบของโปรแกรมต่างๆเป็นคลาสหรือวัตถุ เรียกว่า Object ตัวอย่าง เช่น วัตถุที่มองเห็นได้ เช่น รถ สินค้า หรือ วัตถุที่ไม่

สามารถมองเห็นได้ เช่น เหตุการณ์ต่างๆ ข้อมูลต่างๆของ Object จะถูกซ่อนไว้คลาสเรียกว่า Data Encapsulation ซึ่งมีประโยชน์ในการแก้ไขข้อมูลหรือ Method ใดๆ ที่อยู่ในคลาส โดยไม่ส่งผลกระทบต่อการทำงานหรือเรียกใช้งานของ Object นั้น นอกจากนั้น Java ยังมีคุณสมบัติการสืบทอด (Inheritance) เพื่อส่งผ่านและถ่ายทอดลักษณะต่างๆ ของคลาสแม่ไปยังคลาสลูก ทำให้เขียนโปรแกรมได้ง่ายขึ้น และมีโครงสร้างการทำงานที่เข้าใจง่ายและมีความสัมพันธ์กัน



รูปที่ 2.8 แสดง Class ของภาษาจาวา

2.4.1.3 ภาษา Java เป็นอิสระต่อแพล็ตฟอร์ม (Java is Platform-Independent) Java เป็นอิสระต่อแพล็ตฟอร์ม ทั้งระดับซอร์ซโค้ด (Source Code) และไบนารีโค้ด (Binary Code) ช่วยให้สามารถเคลื่อนย้ายโปรแกรมจากระบบคอมพิวเตอร์หนึ่งไปยังระบบคอมพิวเตอร์อื่นได้อย่างง่ายดาย

2.4.1.4 ภาษา Java มีระบบการทำงานและมีระบบความปลอดภัยที่ดี Java จะคำสั่งต่างๆที่เป็นส่วนประกอบของ Java API โดยมีการรวบรวมเป็นคลาสต่างๆไว้มากมาย ช่วยอำนวยความสะดวกในการเขียนโปรแกรม

2.5 ภาษา PHP

PHP คือภาษา script อย่างหนึ่งที่เรียกว่า server-side script ซึ่งจะทำงาน ในฝั่ง server แล้วส่งการแสดงผลมายัง browser ของตัว client และนอกจากนี้ มันยังเป็น script ที่ embed บน HTML อีกด้วย

PHP เป็นภาษาจำพวก scripting language คำสั่งต่างๆจะเก็บอยู่ในไฟล์ที่เรียกว่า สคริปต์ (script) และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง โดยมีรากฐานโครงสร้างคำสั่งมาจากภาษา ภาษาซี ภาษาจาวา และ ภาษาเพิร์ล

ลักษณะของ PHP ที่แตกต่างจากภาษาแบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมา เพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า server-side script หรือ HTML-embedded scripting language เป็นเครื่องมือที่สำคัญชนิดหนึ่งที่ช่วยให้เราสามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น

โดยสรุปได้ว่า PHP ได้รับการพัฒนาขึ้นมา เพื่อแทนที่ SSI รูปแบบเดิมๆ โดยให้มีความสามารถ และมีส่วนเชื่อมต่อกับเครื่องมือชนิดอื่นมากขึ้น เช่น ติดต่อกับคลังข้อมูลหรือ database เป็นต้น

PHP ได้รับการเผยแพร่เป็นครั้งแรกในปีค.ศ.1994 จากนั้นก็มีการพัฒนาต่อมาตามลำดับ เป็นเวอร์ชั่น 1 ในปี 1995 เวอร์ชั่น 2 (ตอนนั้นใช้ชื่อว่า PHP/FI) ในช่วงระหว่าง 1995-1997 และเวอร์ชั่น 3 ช่วง 1997 ถึง 1999 เวอร์ชั่น 4 ช่วง 1999 ถึง 2006 และในปัจจุบัน เวอร์ชั่น 5.2.6

PHP เป็นผลงานที่เติบโตมาจากกลุ่มของนักพัฒนาในเชิงเปิดเผยรหัสต้นฉบับหรือ OpenSource ดังนั้น PHP จึงมีการพัฒนาไปอย่างรวดเร็ว และแพร่หลายโดยเฉพาะอย่างยิ่งเมื่อใช้ร่วมกับ Apache Web Server ระบบปฏิบัติอย่างเช่น Linux หรือ FreeBSD เป็นต้น ในปัจจุบัน PHP สามารถใช้ร่วมกับ Web Server หลายๆตัวบนระบบปฏิบัติการอย่างเช่น Windows XP/7/8 เป็นต้น

2.6 MySQL

MySQL จัดเป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS : Relational Database Management System) ซึ่งเป็นที่นิยมใช้กันมากในปัจจุบัน โดยเฉพาะอย่างยิ่งในโลกของ Internet เนื่องจาก

- MySQL เป็นฟรีแวร์ทางด้านฐานข้อมูลที่มีประสิทธิภาพสูง
- นักพัฒนาฐานข้อมูลที่เคยใช้ MySQL ต่างยอมรับในความรวดเร็ว การรองรับจำนวนผู้ใช้ และขนาดของข้อมูลจำนวนมหาศาล

- สนับสนุนการใช้งานบนระบบปฏิบัติการมากมาย เช่น UNIX OS/2 MAC OS Windows

- สามารถใช้งานร่วมกับ Web Development Platform เช่น C , C++ , Java , Perl , PHP , Python , TCL , หรือ ASP

- ได้รับความนิยมอย่างมากในปัจจุบัน และมีแนวโน้มสูงขึ้นเรื่อยๆ ในอนาคต

MySQL จัดเป็นซอฟท์แวร์ประเภท Open Source Software สามารถ Download ซอร์สโค้ดต้นฉบับได้จากอินเตอร์เน็ตโดยไม่เสียค่าใช้จ่ายใดๆ การแก้ไขสามารถทำได้ตามต้องการ MySQL ยึดถือสิทธิบัตรตาม GPL (GNU General Public License) ซึ่งเป็นข้อกำหนดของซอฟต์แวร์ประเภทนี้ โดยจะเป็นการชี้แจงว่าสิ่งใดทำได้ หรือทำไม่ได้ในกรณีต่างๆ สามารถหาข้อมูลเพิ่มเติมได้จากเว็บไซท์ www.gnu.org

2.6.1 สถาปัตยกรรมของ MySQL

โครงสร้างการทำงานของ MySQL เป็นลักษณะการทำงานแบบ client / server ซึ่งประกอบด้วย 2 ส่วนหลักๆคือ ส่วนของผู้ให้บริการ (Server) และ ส่วนของผู้ใช้บริการ (Client) โดยในแต่ละส่วนก็จะมีโปรแกรมสำหรับการทำงานตามหน้าที่ของตน

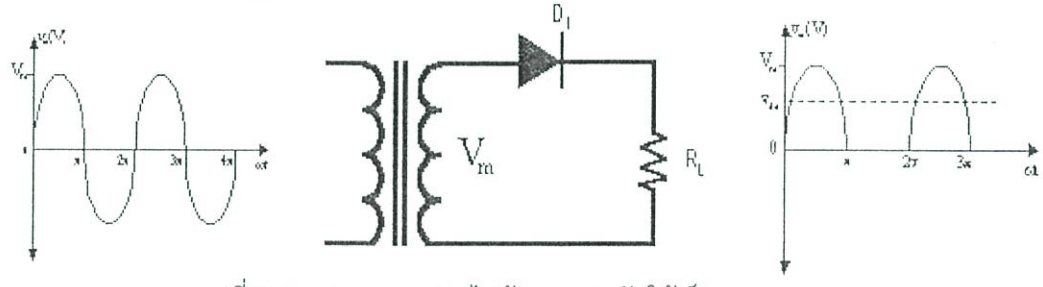
ส่วนของผู้ให้บริการ (Server) เป็นส่วนที่ทำหน้าที่บริหารจัดการระบบฐานข้อมูล ก็คือตัว MySQL server นั่นเอง และเป็นที่จัดเก็บข้อมูลทั้งหมด

ส่วนของผู้ใช้บริการ (Client) คือผู้ใช้นั่นเอง โปรแกรมใช้งานในส่วนนี้ได้แก่ MySQL client , Access , Web development platform ต่างๆ เช่น Java , Perl , PHP , ASP

2.7 วงจร Charger Board

วงจร Charger Board เป็นวงจรที่เป็นแหล่งกำเนิดไฟแก่หุ่นยนต์ในทุกส่วนไม่ว่าจะเป็นแหล่งกำเนิดไฟเพื่อทำให้มอเตอร์หมุนด้วยความเร็วคงที่ วงจรนี้ใช้ไฟเลี้ยงแก่วงจร +9 V ซึ่งใช้ แบตเตอรี่จำนวน 6 ก้อน และยังสามารถชาร์จไฟแก่วงจรโดยใช้ Adapter จากไฟ 220 V เป็น +9 V ในวงจร Charger Board มีวงจรทาง Electronic ที่สำคัญได้แก่

2.7.1 วงจร Bridge Rectifier: วงจรเรียงกระแสหรือเรียกอีกอย่างหนึ่งว่าเรคติไฟเออร์คือวงจรไฟฟ้าที่มีคุณสมบัติในการแปลงสัญญาณกระแสสลับให้กลายเป็นสัญญาณไฟฟ้ากระแสตรงหรือมีคุณสมบัติยอมให้ไฟฟ้าไหลผ่าน ไปในทิศทางใดทิศทางหนึ่ง อุปกรณ์ที่นิยมใช้ในการแปลงสัญญาณได้แก่ ไดโอด

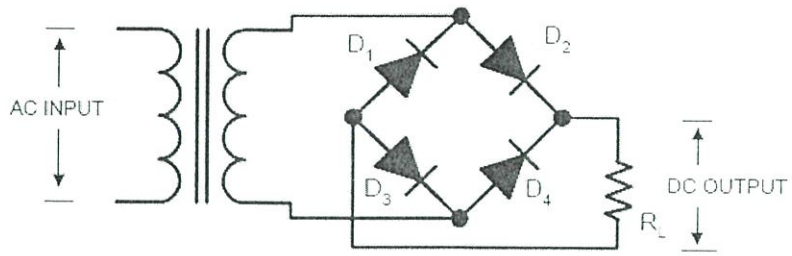


รูปที่ 2.9 แสดง การแปลงไฟฟ้ากระแสสลับให้เป็นกระแสตรง

การแปลงไฟฟ้ากระแสสลับให้เป็นกระแสตรงโดยหลักๆ แล้วจะมี 3 รูปแบบ ได้แก่

- Half Wave Rectifier หรือ แบบครึ่งคลื่น
- Full Wave Rectifier หรือ แบบเต็มคลื่น ใช้หม้อแปลงแบบ Center Tab โดยกำหนดจุดแท็บกลางของหม้อแปลงเป็นหลัก
- Bridge Rectifier เป็นแบบเต็มคลื่นอีกรูปแบบหนึ่งเช่นกัน แต่ไม่ใช้หม้อแปลงแบบ Center Tab

ในโครงงานนี้จะใช้การแปลงไฟฟ้าจากวงจรเรียงกระแสแบบเต็มคลื่น (Full Wave) คือวงจรเรียงกระแสแบบบริดจ์ แรงดันไฟสลับจะต่อเข้าที่ สองมุมของวงจรบริดจ์และเอาต์พุตจะถูกนำออกที่สองมุมที่เหลือ ดังในรูป



รูปที่ 2.10 แสดงวงจรเรียงกระแสแบบเต็มคลื่น (Full Wave)

ในแต่ละครึ่งไซเคิลของวงจรอินพุตสมมุติว่าเมื่อขั้ว A ของขดทุติยภูมิมีค่าเป็นบวก และขั้ว B มีค่า เป็นลบจึงเหมือนกับครึ่งไซเคิลลบถูกป้อนเข้าทางขดปฐมภูมิของหม้อแปลง ไดโอด D2 และ D3 จะอยู่ในลักษณะไบอัสตรงดังนั้นกระแสจึงไหลครบวงจรจากขั้ว A ผ่านไดโอด D2 ความต้านทานโหลดและไดโอด D3 แล้วกลับเข้าสู่ขั้ว B ของหม้อแปลง ดังรูปที่ 1.8 ก. และเมื่อแรงดันไฟสลับเปลี่ยนข้างมาเป็นขั้วบวก ที่ขั้ว B และเป็นลบที่ขั้ว A การนำกระแสของไดโอดจะเปลี่ยนไปโดยเริ่มจากจุด B ของขดทุติยภูมิ ผ่าน D4 ความต้านทานโหลด และ D1 กลับเข้าขั้ว A ของหม้อแปลง ทิศทางแรงดันตกคร่อมโหลดจะมีภูมิ ผ่าน D4 ความต้านทานโหลด และ D1 กลับเข้าขั้ว A ของหม้อแปลง ทิศทางแรงดันตกคร่อมโหลดจะมีทิศทางเดียวกับตอนแรกคือ มีขั้วบวกอยู่ทางด้านบน ดังนั้นการนำกระแสไดโอดจะเกิดสลับกันทีละสองตัว D2 กับ D3 และ D1 กับ D4

2.7.2 วงจร Voltage Regulators

วงจรคงค่าแรงดันมีหน้าที่รักษาระดับแรงดันเอาต์พุตให้มีค่าคงที่ไม่เปลี่ยนแปลงตามกระแสที่จ่ายให้กับโหลดและแรงดันอินพุตโดยทั่วไป

2.7.3 วงจร Voltage Divider: วงจรแบ่งแรงดัน

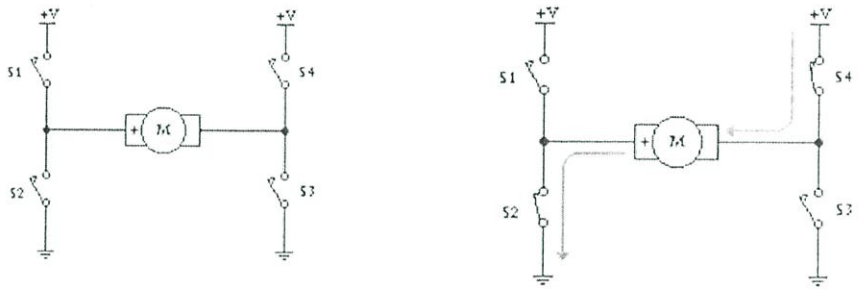
วงจรแบ่งแรงดันไฟฟ้าเป็นวงจรที่ทำหน้าที่แบ่งแรงดันไฟฟ้าออกเป็นระดับต่าง ๆ ตามความต้องการ วงจรมีลักษณะเป็นวงจรแบบอนุกรมสามารถแบ่งออกเป็น 2 ชนิดด้วยกันคือวงจร แบ่งแรงดันไฟฟ้าแบบไม่มีภาระไฟฟ้า (Unloaded Voltage Divider)และวงจรแบ่งแรงดันไฟฟ้าแบบมีภาระไฟฟ้า (Loaded Voltage Divider)

2.8 วงจร Motor Controller Board

เป็นวงจรควบคุมการหมุนของมอเตอร์ โดยมอเตอร์ซ้ายและมอเตอร์ขวาจะเป็นอิสระต่อกัน ซึ่งในวงจร Motor Controller Board นี้ จะมีวงจรที่ควบคุมล้อซ้าย และวงจรที่ควบคุมล้อขวาที่แยกจากกันพร้อมมีไฟเลี้ยงวงจรมาจากวงจร Charger Board ซึ่งวงจรนี้จะทำงานก็ต่อเพื่อมีการกดปุ่มสวิตช์จากวงจร Charger Board

2.8.1 H-Bridge Circuit

หลักการของวงจรนั้นจะประกอบไปด้วยสวิทช์ 4 ตัว นั่นก็คือ S1, S2, S3 และ S4 นั่นเอง ซึ่งในรูปตัวอย่างจะใช้ดีซีมอเตอร์ (DC-Motor) เป็นโหลด (Load) ของวงจร



รูปที่ 2.11 แสดง H-Bridge Switching

ในสภาวะเริ่มต้นสวิตช์ทุกตัวปิด (Off) อยู่จะไม่มีอะไรเกิดขึ้นทั้งสิ้น เพราะไม่มีกระแสไฟฟ้าไหลเข้าสู่มอเตอร์ และเมื่อเราทำการเปิด (On) สวิตช์ S1 และ S3 พร้อมกัน จะเป็นการเชื่อมวงจรทำให้มีกระแสไฟฟ้าไหลผ่านมอเตอร์ จากขั้วบวกของมอเตอร์ไปยังขั้วลบของมอเตอร์จึงทำให้มอเตอร์สามารหมุนได้ในทิศทางฟอร์เวิร์ด (Forward) (จะหมุนแบบตามเข็มนาฬิกาหรือทวนเข็มนาฬิกานั้น ขึ้นอยู่กับลักษณะของ การพันขดลวดภายในมอเตอร์) และในทางกลับกันถ้าหากเราทำการเปิด (On) สวิตช์ S2 และ S4 พร้อมกัน ก็จะเป็นการเชื่อมวงจร และทำให้เกิดกระแสไฟฟ้าไหลผ่านมอเตอร์ จากขั้วลบของมอเตอร์ไปยังขั้วบวกของมอเตอร์ จึงทำให้มอเตอร์สามารถหมุนได้และเป็นการหมุนในทิศทางรีเวิร์ส (Reverse) (กลับทิศทางกับกรณีแรก)

จากทฤษฎีข้างต้นจึงสามารถนำไปใช้ในการในวงจรควบคุมมอเตอร์เพื่อควบคุมทิศทางการหมุนของมอเตอร์ โดยผู้จัดทำได้เลือกมอสเฟต (Mosfet) มาเป็นสวิตช์ในการควบคุมทิศทางการหมุนของมอเตอร์

2.8.2 การควบคุมความเร็วมอเตอร์

การควบคุมความเร็วของมอเตอร์สามารถทำได้โดยการควบคุมระดับแรงดันตกคร่อมตัวมอเตอร์และวิธีการที่นิยมใช้จะมีสองวิธีดังนี้

2.8.2.1 การปรับระดับแรงดันโดยใช้หลักการเปลี่ยนค่าความต้านทานของวงจร ซึ่งทำให้แรงดันตกคร่อมเปลี่ยนไปด้วย แต่จะเห็นว่ามีวิธีนี้เป็นการสร้างความสูญเสียเนื่องจากกำลังไฟฟ้าตกคร่อมตัวต้านทาน (พลังงานไฟฟ้าเปลี่ยนเป็นพลังงานความร้อน)

2.8.2.2 การปรับระดับแรงดันโดนการเปิดปิดแหล่งจ่ายไฟ โดยระดับแรงดันเฉลี่ยที่ได้จะขึ้นกับช่วงเปิดและช่วงปิด เราเรียกวิธีการนี้ว่า Pulse width modulate และเรียกอัตราส่วนของช่วงเปิดและช่วงปิดว่าดิวตี้ไซเคิล (Duty cycle) วิธีการนี้มี ข้อเสียคือ หากเราเลือกความถี่ของพัลส์ไม่เหมาะสมอาจจะทำให้มอเตอร์สั่นได้

2.9 เซ็นเซอร์

2.9.1 ความหมายของเซ็นเซอร์

เซ็นเซอร์ เป็นอุปกรณ์ที่ใช้ตรวจจับสภาวะใดๆเช่น อุณหภูมิ สี แสง หรือวัตถุต่างๆโดยอาศัยหลักการที่แตกต่างกันไปตามแต่ชนิด เช่น เซ็นเซอร์สีขาวดำ โดยอาศัยหลักการสะท้อนแสงของสีขาวและดำ ทางฟิสิกส์แล้วจะเห็นว่าสีขาวมีอัตราการสะท้อนแสงมากกว่าสีดำ เราจึงสามารถนำแสงสะท้อนมาเปรียบเทียบได้ โดยใช้ตัวเซ็นเซอร์คือ อุปกรณ์จำพวกโฟโต้ เช่น โฟโต้ไดโอด โฟโต้ทรานซิสเตอร์ LDR เป็นต้น ซึ่งมีความไวต่อแสงมาก โดยส่วนใหญ่จะแสดงผลเอาต์พุตในรูปของความต้านทานที่เปลี่ยนไปตามสภาวะของตัวเซ็นเซอร์นั้นๆ

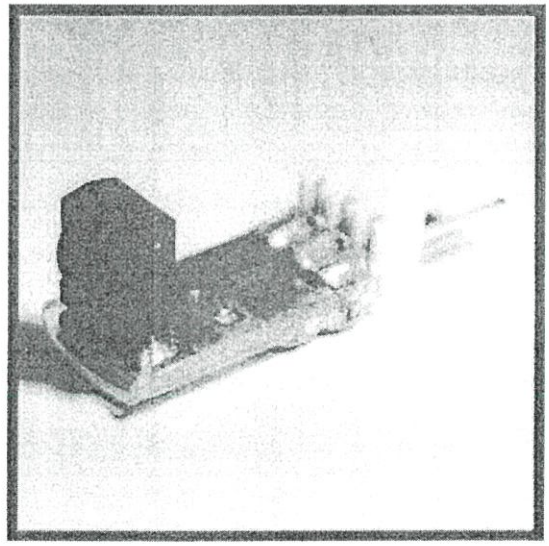
2.9.2 คุณสมบัติของเซ็นเซอร์

เซ็นเซอร์คืออะไร ในที่นี้ เซนเซอร์เป็นตัวที่ใช้ตรวจจับสภาวะใด ๆ เช่น อุณหภูมิ สี แสง หรือ วัตถุ ต่าง ๆ โดยอาศัยหลักการที่แตกต่างกันไปแต่ละตัว เพื่อเปลี่ยนจากคุณสมบัติของฟิสิกส์ มาเป็นคุณสมบัติทางไฟฟ้า เช่น ที่ใช้งานกันใน Robot คือ เซ็นเซอร์ สีขาวดำ โดยอาศัยหลักการสะท้อนแสงของสีขาวและสีดำ ทางฟิสิกแล้วจะเห็นว่าสีขาวมีอัตราการสะท้อนแสงมากกว่าสีดำ เราจึงสามารถนำแสงสะท้อนมาเปรียบเทียบได้โดยใช้ตัวเซ็นเซอร์คือ อุปกรณ์จำพวก โฟโต้ เช่น โฟโต้ไดโอด โฟโต้ทรานซิสเตอร์ LDR เป็นต้น ซึ่งจะมีความไวต่อแสงมากตัวเซ็นเซอร์ส่วนใหญ่เมื่อแสดงผลเอาพุต จะแสดงผลในรูปความต้านทานที่เปลี่ยนไปตามสภาวะของตัวเซนเซ็อร์นั้น ๆ ในปัจจุบัน ในวงการเซ็นเซอร์ ได้พัฒนาไปมาก มีเซ็นเซอร์ให้เราได้เลือกใช้มากมาย มีวงจรที่ง่ายขึ้น มีความแน่นอน สูง จึงทำให้เราสามารถมีตัวเลือกในการใช้งานมากขึ้น ซึ่งในโครงงานนี้เลือกใช้ อินฟราเรดเซ็นเซอร์ (Infrared Sensor)

2.9.2.1 อินฟราเรดเซ็นเซอร์ (Infrared Sensor)

อินฟราเรดเป็นแสงที่มีความยาวคลื่นอยู่ในช่วงประมาณ 780-3000 นาโนเมตร ซึ่งเป็นแสงที่ไม่สามารถมองเห็นได้ด้วยตาเปล่า ซึ่งเป็นที่นิยมที่จะนำมาใช้ในการสื่อสาร หรือตรวจจับสิ่งของต่างๆ เพราะปัญหาการรบกวนของสัญญาณจากแสงอื่นๆมีน้อย อีกทั้งการสร้างวงจรที่ใช้ในระบบอินฟราเรดไม่มีความซับซ้อนมากนัก และความน่าเชื่อถือของสัญญาณที่ส่งก็มีความเชื่อถือที่สูงในการนำไปใช้งาน

โดยในระบบอินฟราเรดจะต้องมีเครื่องส่ง และเครื่องรับ ซึ่งการสร้างเครื่องส่งนั้นก็เพียงแค่ให้มีการส่งแสงออกมาในช่วงความถี่ที่สูงกว่าความถี่ทั่วๆไปของแสงธรรมดา คือต้องมากกว่า 20 kHz โดยจะใช้ IR LED เป็นตัวขับแสงอินฟราเรด ส่วนการสร้างเครื่องรับนั้นจะใช้ โฟโต้ไดโอด หรือโฟโต้ทรานซิสเตอร์ เป็นตัวรับแสงโดยที่ทั้งเครื่องรับและเครื่องส่งจะต้องมีความถี่เท่ากัน สำหรับโครงการนี้จะใช้คุณสมบัติของตัวโฟโต้ไดโอด ที่เมื่อมีแสงมาตกกระทบแล้วจะทำให้ปริมาณของกระแสที่วิ่งผ่านโฟโต้ไดโอดมีค่าเพิ่มขึ้นตามปริมาณความเข้มของแสงทำให้ Voltage ที่ตกคร่อมโฟโต้ไดโอดมีค่ามากตามไปด้วย และใช้ในการตรวจจับหาแหล่งที่มาของแสงเพื่อที่จะใช้บอกทิศทางต่อไป



รูปที่ 2.12 แสดงตัวอย่างเซ็นเซอร์อินฟราเรด

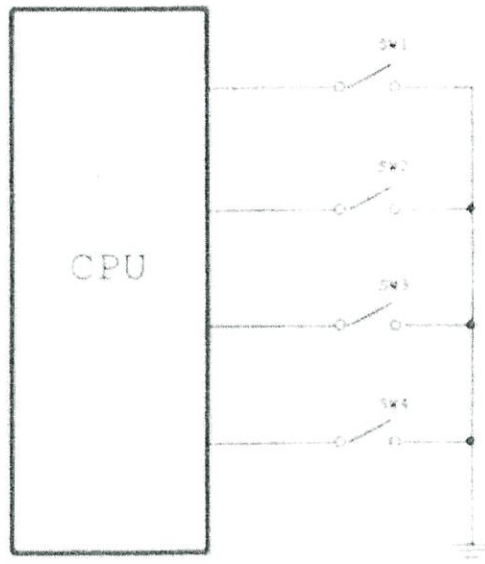
2.10 KEY PAD

2.10 ความรู้ทั่วไปเกี่ยวกับ Input Logic

ในงานทางด้าน Microcontroller อุปกรณ์ Input เป็นสิ่งสำคัญ เช่น Switch, Keyboard อุปกรณ์เหล่านี้เป็นส่วนสำคัญสำหรับติดต่อระหว่างระบบ Microcontroller กับผู้ใช้ การเชื่อมต่อ Keyboard กับ Microcontroller นั้นสามารถทำได้หลายวิธี ทั้งนี้ขึ้นอยู่กับการออกแบบฟังก์ชั่นการทำงานของบอร์ดที่ผู้ออกแบบกำหนดขึ้น ในงานบางประเภทอาจใช้จำนวน Switch เพียง 1-2 ตัวเท่านั้น แต่สามารถออกแบบและกำหนดลำดับขั้นตอนของหน้าที่การทำงานให้สามารถทำงานได้มากมายหลายหน้าที่ โดยไม่จำเป็นต้องใช้จำนวน Switch หลายตัวให้สิ้นเปลือง เช่น Switch สำหรับใช้ในการตั้งเวลาของนาฬิกาแบบดิจิตอล เป็นต้น ซึ่งในงานบางประเภทนั้นก็อาจมีการออกแบบวงจรสำหรับเชื่อมต่อกับ Switch และ Keyboard ขึ้นเป็นพิเศษโดยเฉพาะ โดยการใช้ไอซีสำเร็จรูปสำหรับทำหน้าที่ตรวจสอบการทำงานของ Switch ต่างๆ แล้วจึงส่งค่าที่ได้ให้กับ CPU เพื่อทำการประมวลผลและตัดสินใจในภายหลังก็ได้ แต่อย่างไรก็ตามสำหรับผู้ใช้งานไมโครคอนโทรลเลอร์ส่วนมากแล้วมักไม่นิยมใช้วิธีการดังกล่าวเพราะเป็นการสิ้นเปลืองและเพิ่มต้นทุนค่าใช้จ่ายโดยไม่จำเป็น แต่จะหันมาใช้วิธีการให้ CPU ทำหน้าที่คอยตรวจสอบ Switch ต่างๆ เองโดยตรง ซึ่งเมื่อพิจารณาตามหลักการโดยรวมแล้วจะเห็นว่ามีวิธีการอยู่ด้วยกัน 2 แบบ คือ

2.10.1 การต่อวงจรแบบ Switch ขาร่วม

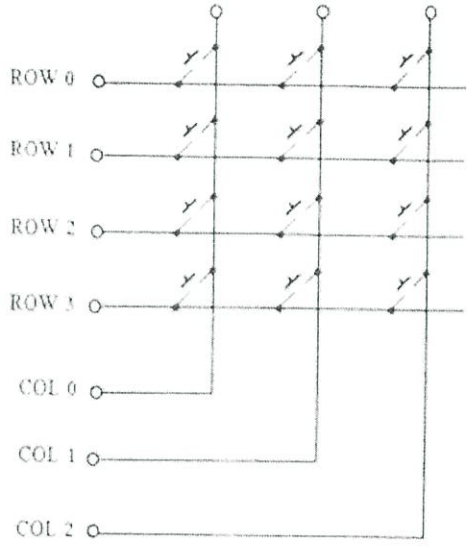
โดยในวิธีนี้จะเหมาะสำหรับงานที่มีความต้องการใช้ Switch จำนวนไม่มากนัก หรือในงานที่ต้องการแยกสัญญาณ Input ของแต่ละจุดออกจากกันโดยสิ้นเชิง ซึ่งในวิธีการแบบนี้จะต้องใช้จำนวนบิตของ Input มากเท่ากับจำนวน Switch ที่ต้องการจะใช้ในระบบ แต่ อย่างไรก็ตามวิธีนี้มีความสะดวกและง่ายในการออกแบบวงจรและเขียนโปรแกรมเพื่อตรวจสอบหาตำแหน่งของ Switch ที่ถูกกดได้ง่ายกว่าวิธีการอื่นๆ เพราะการทำงานของ Switch แต่ละตัวจะแยกออกจากกันอย่างอิสระ แต่วิธีการนี้จะมีข้อเสียคือความสิ้นเปลืองจำนวนพอร์ตมากขึ้นเมื่อต้องใช้กับระบบที่มีความจำเป็นต้องใช้ Input มากๆ ซึ่งวิธีการนี้จะต่อวงจรโดยนำหน้าสัมผัสของ Switch ทุกตัวมารวมเข้าด้วยกันเป็น Common ไว้ สำหรับหน้าสัมผัสอีกด้านหนึ่งของ Switch แต่ละตัวก็จะต่อเข้ากับ Port-Input ของระบบโดยตรง ซึ่งส่วนมากแล้ว Port-Input นี้จะทำการคงสถานะของสัญญาณที่แน่นอนไว้ก่อนเสมอ โดยการต่อตัวต้านทานเข้ากับทุกบิตของ Port-Input ขึ้นแหล่งจ่ายไว้ (Pull-Up Resistor) เพื่อให้ค่าสถานะของโลจิกของ Port-Input มีค่าเป็น “1” อยู่ตลอดเวลาในขณะที่ยังไม่มีการกด Switch ตัวใด สำหรับขาร่วมของ Switch ก็จะต่อลง Ground ไว้ ดังนั้นเมื่อมีการกด Switch ตัวใดก็จะทำให้ตำแหน่งของบิต Input นั้นมีค่าเป็น “0”



รูปที่ 2.13 แสดงตัวอย่างการต่อวงจร Switch แบบขาร่วม

2.10.2 การต่อวงจร Switch แบบ Matrix

โดยวิธีการนี้เหมาะกับระบบที่มีความจำเป็นต้องใช้งาน Switch มากๆ เช่น วงจรแป้นพิมพ์ Keyboard ที่ใช้สำหรับป้อนค่าตัวเลข ตัวอักษร และข้อความต่างๆ ซึ่งจะเกิดความไม่สะดวกสำหรับผู้ใช้งานเป็นอย่างมาก ถ้าออกแบบให้มีจำนวน Switch Keyboard น้อยๆ เพราะจะยากแก่การค้นหาตำแหน่งของตัวเลข ตัวอักษรที่อยู่ซ้อนกันอยู่ในคีย์เดียวหลายๆ ชั้น โดยวิธีการนี้จะต้องใช้พอร์ต 2 ส่วน คือ พอร์ตสำหรับอ่านค่าสถานะของ Key-Switch จากทางแถว (Row) และพอร์ตสำหรับทำหน้าที่ส่งค่าออกไป Scankey ในแต่ละหลัก (Column) ของวงจร โดยจำนวนของ Switch จะขึ้นอยู่กับขนาดของแถวและหลักที่ใช้ เช่น ถ้าเป็นขนาด 4x4 ก็จะได้ทั้งหมด 16 ตำแหน่ง เป็นต้น



รูปที่ 2.14 แสดงตัวอย่างการต่อวงจร Switch แบบ Matrix

โดยวิธีการ Scankey แบบ Matrix นี้จะทำทีละหลัก (Column) โดยเริ่มจากหลักแรกไปหาหลักสุดท้ายตามลำดับ สำหรับลักษณะของการต่อวงจรโดยทั่วไปของวิธีการนี้จะนิยมคงสถานะของสัญญาณด้านที่เป็น Input ให้มีค่าเป็น "1" รอไว้ก่อนเสมอโดยการต่อตัวต้านทาน Pull-Up เข้ากับ Port-Input รอไว้ก่อน โดยในการ Scankey จะทำทางด้านหลัก (Column) โดยส่งค่าออกไปทางด้าน Port-Input ให้มีค่าเป็น "0" ครั้งละ 1 บิต แล้วก็อ่านค่าจาก Port-Input เข้ามาตรวจสอบว่าทุกบิตยังคงเป็น "1" อยู่หรือไม่ ซึ่งถ้าพบว่ามีบิตใดเป็น "0" (Column Active = "0") ก็สามารถทราบได้ทันทีว่ามีการกดคีย์ขึ้นที่ตำแหน่ง Row และ Column นั้นๆ แต่ถ้าทุกบิตยังคงมีค่าเป็น "1" ก็ให้เปลี่ยนการ Scan ไปยัง Column ถัดไปอีกโดยทำยยยเหมือนกันกับ Column แรกจนครบทุก Column

2.11 ไมโครคอนโทรลเลอร์ AVR

ไมโครคอนโทรลเลอร์ AVR เป็นไอซีไมโครคอนโทรลเลอร์ของบริษัท Atmel มีสถาปัตยกรรมภายในเป็นแบบ RISC (Reduced Instruction Set Computer) โดยใช้สัญญาณนาฬิกาเพียง 1 ลูกในการปฏิบัติงานใน 1 คำสั่ง โดยจะประกอบด้วยหน่วยความจำโปรแกรมภายในที่เป็นแบบแฟลช โปรแกรมข้อมูลได้แบบ In-System Programmable และในบางเบอร์ยังสามารถมีการกำหนดตำแหน่งของหน่วยความจำที่สร้างเป็นบูตโหลดเดอร์ (เขียนโปรแกรมเพื่อติดต่อกับ PC หรือไอซีตัวอื่นๆ และยังสามารถโปรแกรมให้กับตัวเองได้) มีขนาดของหน่วยความจำตามเบอร์ของไอซีแต่ละตัว ยกตัวอย่างคุณสมบัติเบื้องต้นของไอซีเบอร์ Atmega8A ได้ดังต่อไปนี้

- เป็นไอซีขนาด 8-bit ใช้พลังงานต่ำ
- มีโครงสร้างภายในแบบ RISC
- มีคำสั่งควบคุมการทำงานไมโครคอนโทรลเลอร์ 130 คำสั่ง คำสั่งส่วนมากจะทำสำเร็จในรอบสัญญาณนาฬิกาเดี่ยว
- มีจำนวนรีจิสเตอร์ทั่วไปขนาด 32 x 8
- มีหน่วยความจำโปรแกรมภายในแบบ Flash ขนาด 8K Bytes มีการโปรแกรมได้แบบ In-System Self-programmable
- มีหน่วยความจำภายในแบบ EEPROM ขนาด 512 Bytes
- มีหน่วยความจำภายในแบบ SRAM ขนาด 1K Byte
- เขียน / ลบ ได้ถึง 10,000 ครั้ง สำหรับหน่วยความจำแบบ Flash และ 100,000 สำหรับหน่วยความจำแบบ EEPROM
- กำหนดการ Boot Code Section ในตำแหน่งต่างๆ และ Lock Bits ได้ (ทำ Boot Loadder)
- Programming Lock for Software Security ป้องกันข้อมูล
- Timer/Counters ขนาด 8-bit 2 ตัว และมี Separate Prescaler โหมด Compare อีก 1 ตัว
- 16-bit Timer/Counter with Separate Prescaler, Compare Mode , and Capture
- มี PWM 3 ช่อง
- มีการติดต่อแบบ Master/Slave SPI Serial Interface
- ใช้งาน RC Oscillator ภายในไอซี และภายนอกไอซีได้
- ทำงานที่แรงดัน 2.7 - 5.5V for ATmega8A
- ทำงานที่ความถี่ 0 - 16 MHz for ATmega8A

2.11.1 ไมโครคอนโทรลเลอร์ AVR ได้แบ่งออกเป็น 4 กลุ่มใหญ่ๆ ดังนี้

2.11.1.1 TinyAVR — ATtiny series เช่นเบอร์ Tiny13 , Tiny2313

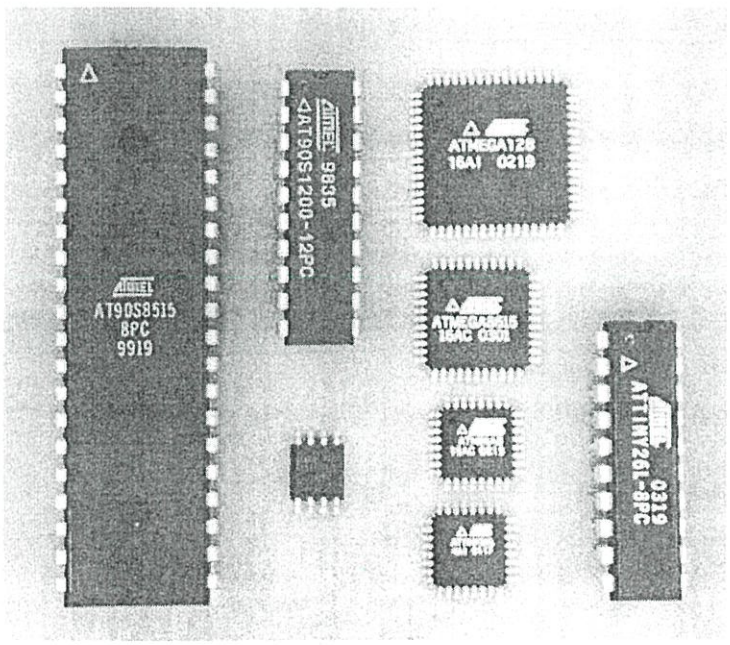
2.11.1.2 MegaAVR — ATmega series เช่นเบอร์ ATmega8 ,ATmega16, ATmega32, ATmega64

- มีหน่วยความจำโปรแกรมขนาด 4 – 256 kB
- มีจำนวนขาใช้งาน 28–100 ขา
- มีชุดคำสั่งที่สามารถจัดการกับหน่วยความจำที่มีขนาดใหญ่มากขึ้น
- มีส่วนของอุปกรณ์เสริมมากในตัวไอซี

2.11.1.3 XMEGA ATxmega series เช่นเบอร์ ATxmega64A1, ATxmega128A1

- มีหน่วยความจำโปรแกรมขนาด 16–384 kB
- มีจำนวนขาใช้งาน 44–64–100 ขา
- มีส่วนของอุปกรณ์เสริมมากในตัวไอซีโดยการใช้งานกับ Digital-to-Analog Converter (DACs) และยังสามารถเขียนรหัสเฉพาะ โดยเข้ากันได้กับไฟล์แบบ AES และ DES (เช่นใช้สำหรับการกำหนดรหัสส่วนตัว)

2.11.1.4 Application specific AVR เป็นไอซีที่สร้างเพื่อใช้งานเฉพาะ เช่น CAN AVR (ATmega64C1), LCD AVR (ATmega3290P/V), USB AVR (AT90USB1287)



รูปที่ 2.15 แสดงตัวอย่างของ Microcontroller AVR ชนิดต่างๆ

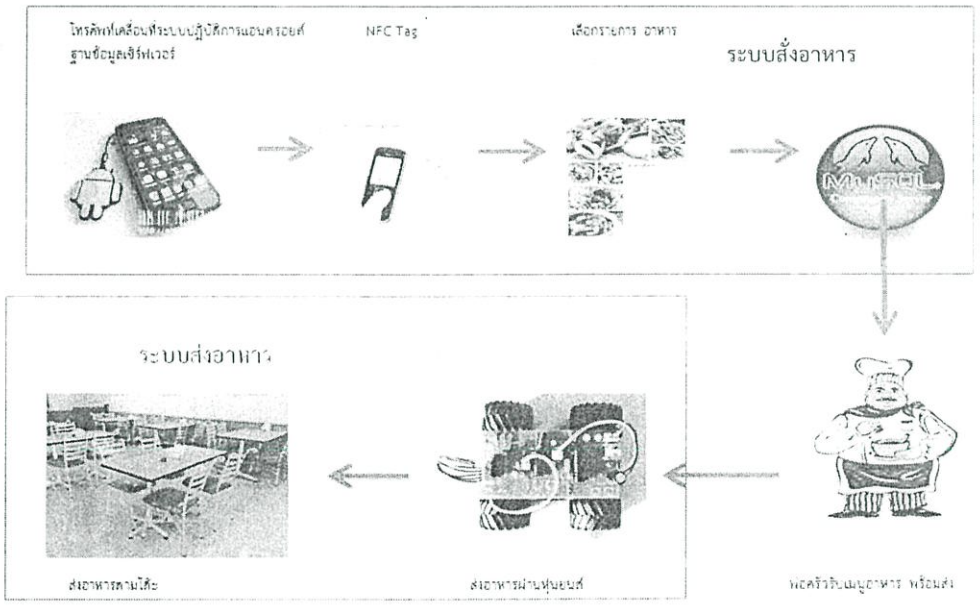
บทที่ 3

การออกแบบและการจัดทำปริญญานิพนธ์

3.1 การออกแบบ

3.1.1 การออกแบบโดยภาพรวม

ปริญญานิพนธ์นี้เป็นการออกแบบระบบสั่งและส่งอาหารโดยอัตโนมัติ เมื่อลูกค้าเข้ามายังร้านอาหารจะทำการแท็กข้อมูลของลูกค้าผ่านเทคโนโลยี NFC (Near Field Communication) พร้อมทั้งสั่งอาหารผ่านแอพพลิเคชั่นบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์จากนั้นข้อมูลอันได้แก่ เมนูอาหาร หมายเลขโต๊ะ คอมเม้นต่างๆจะถูกส่งไปที่เซิร์ฟเวอร์ พร้อมแสดงผลที่พ่อครัว จากนั้นเมื่อพ่อครัวประกอบอาหารเสร็จเรียบร้อย จะฝากอาหารของแต่ละโต๊ะไปกับหุ่นยนต์อัตโนมัติ และทำการส่งอาหารไปยังโต๊ะอาหารได้อย่างถูกต้อง



รูปที่ 3.1 Block diagram รวมของระบบ

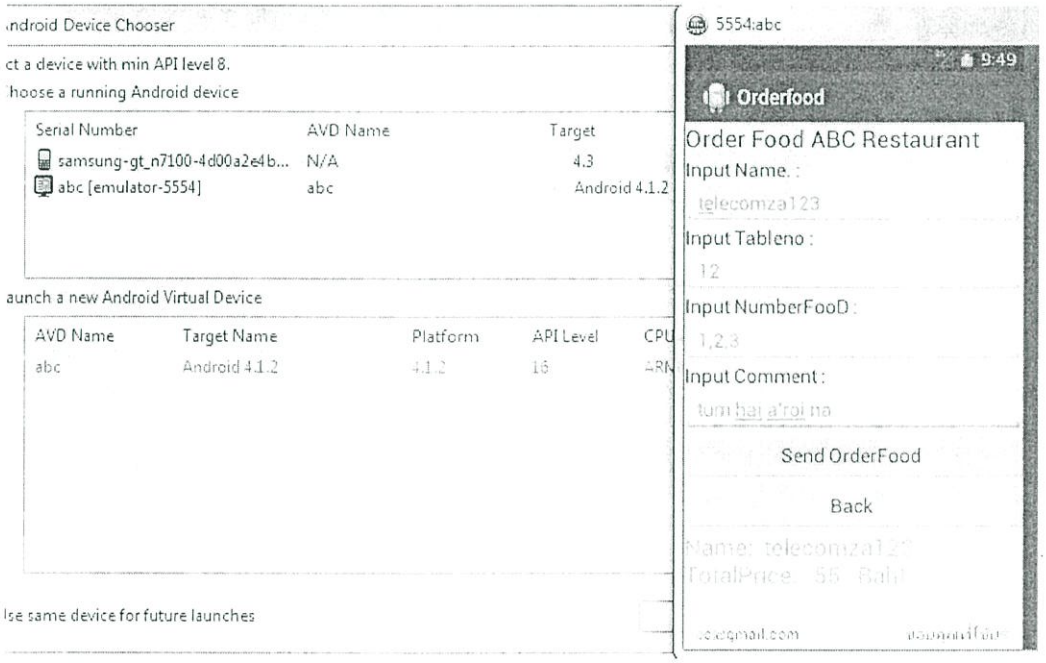
3.1.2 การติดตั้งเครื่องมือจำลองโทรศัพท์ Android และ เครื่องมือช่วยเขียนโปรแกรม

ติดตั้งโดย ADT Bundle ซึ่งเป็นแพ็คเกจที่รวม 3 โปรแกรมเข้าไว้ด้วยกัน (Eclipse , Android SDK และ ADT) โดยติดตั้ง ADT Bundle แพ็คเกจเดียว จะทำการติดตั้งทั้ง 3 โปรแกรมให้พร้อมกัน โดยจะได้รูปแบบของโปรแกรม Eclipse ที่พร้อมใช้เขียนโปรแกรมและจำลองโทรศัพท์ เมื่อติดตั้ง ครบทั้ง 3 โปรแกรมแล้ว ดังรูปที่ 3.2

รูปที่ 3.2 แสดงโปรแกรม Eclipse พร้อมใช้งานและโทรศัพท์จำลอง

3.1.2.1 การใช้งานอีมูเลเตอร์จำลองโทรศัพท์บน Eclipse

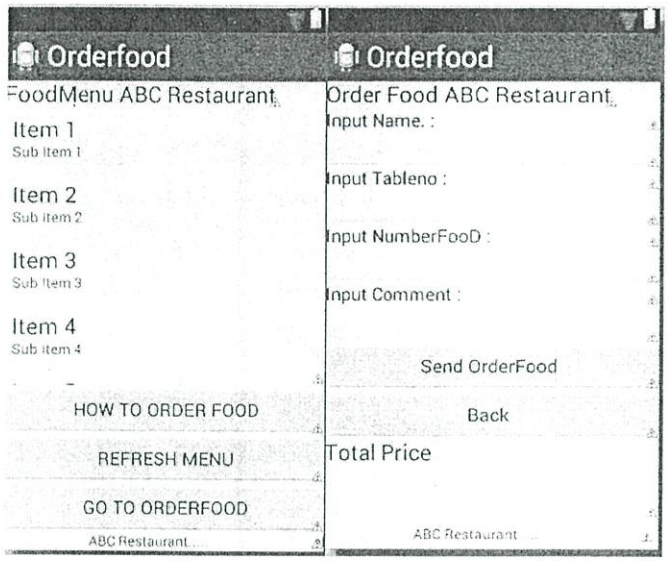
หากมีการเขียน Application เสร็จสมบูรณ์และจะทำการรัน Application บนคอมพิวเตอร์ผ่านอีมูเลเตอร์จำลอง (AVD) โดยคลิกเมนู Windows > AVD Manager แล้วตั้งค่าอุปกรณ์ที่จะจำลอง > คลิกปุ่ม Run ดังรูปที่ 3.3



รูปที่ 3.3 การรันอีมูเลเตอร์จำลองบน Eclipse

3.1.3 การออกแบบส่วน Layout ของแอพพลิเคชั่น

ในการออกแบบไฟล์หน้าจอ Layout ในการทำงานของ Android จะแยกการทำงานเป็นไฟล์ควบคุม (.java) และไฟล์หน้าจอ (.xml) โดยใน Layout (.xml) เป็นเสมือนหน้าแสดงผลออกจอ และรับข้อมูลจาก Activity ผู้ใช้ทำงานต่อไป โดยในโครงงานนี้จะออกแบบหน้าจอ ประกอบไปด้วย Textview Edittext และ Button โดยมีรูปแบบการจัดเรียงข้อมูลแบบ Listview ดังรูปที่ 3.4 และ 3.5

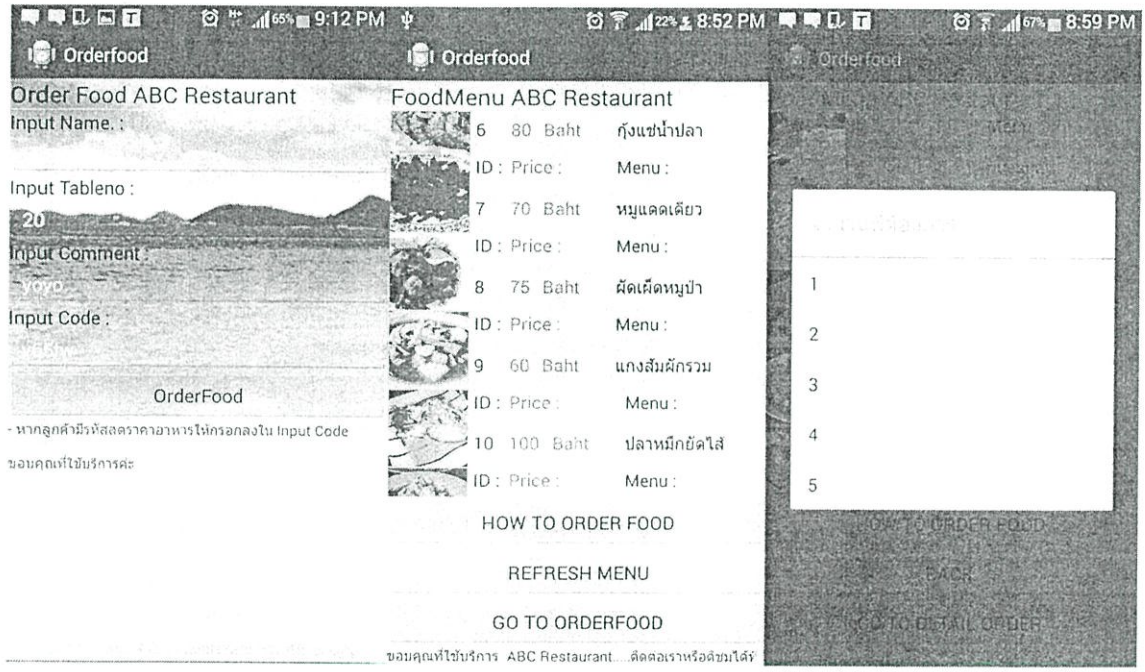


รูปที่ 3.4 หน้าจอแสดงผลของแอพพลิเคชั่น

รูปที่ 3.5 หน้าจอแสดงผลของแอพพลิเคชั่น(2)

3.1.4 การออกแบบส่วนไฟล์ควบคุม (.java)

ในส่วนไฟล์ควบคุมจะมีหน้าที่เป็นโปรแกรมเชื่อมต่อเพื่อการอ่านค่าหรือส่งค่าไปที่องค์ประกอบหน้าจอที่อยู่บนหน้าจอทุกอย่าง โดยในโครงงานนี้แอพพลิเคชั่นที่หนึ่งจะทำการแสดงข้อมูลเป็นแบบ Listview และสามารถกดปุ่ม Button เพื่อไปยังส่วนการสั่งอาหารหรือจะทำการตรวจสอบวิธีการสั่งอาหาร ส่วนแอพพลิเคชั่น BarcodeScanner ทำหน้าอ่านข้อมูลใน QR Code และ readnfc ทำหน้าที่อ่านข้อมูลใน NFC Tag

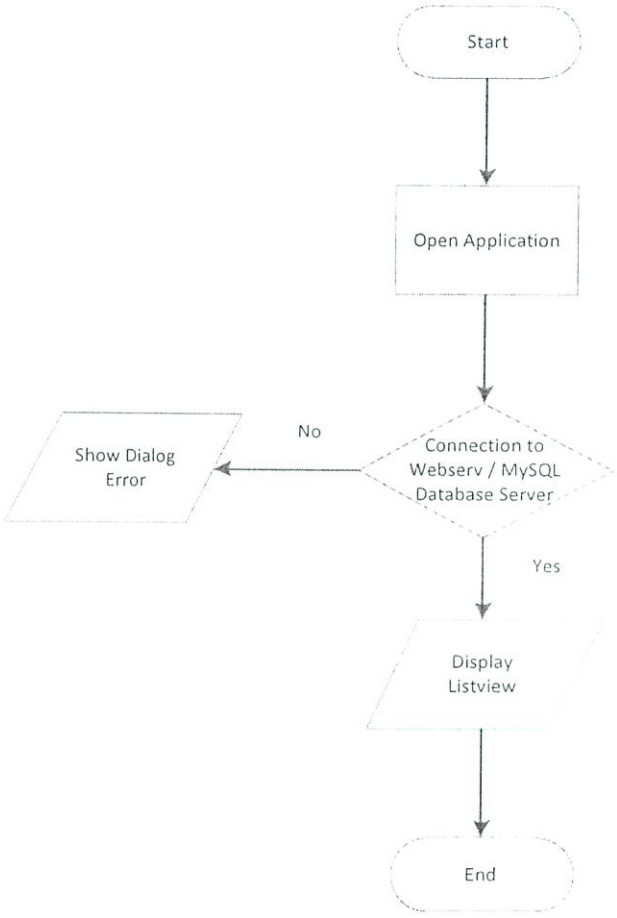


รูปที่ 3.6 รูปแอพพลิเคชั่น Order Food

3.1.4.1 โฟลว์ชาร์ตการทำงานของแอพพลิเคชั่น

1) โฟลว์ชาร์ตของแอพพลิเคชั่นรับค่าจาก Database

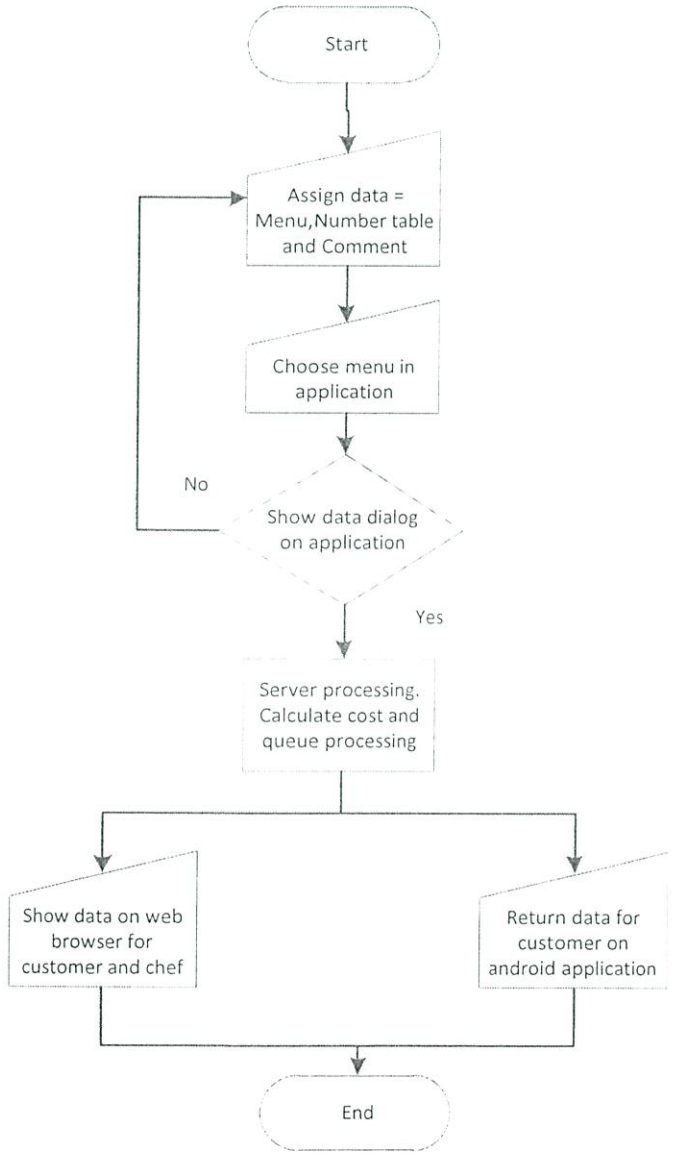
เมื่อเริ่มเปิดแอพพลิเคชั่น Android Client จะทำการติดต่อกับเซิร์ฟเวอร์โดยจะทำการดึงข้อมูลจากฐานข้อมูลที่ได้สร้างไว้มาแสดงผลบน Android Client โดยถ้าติดต่อกับเซิร์ฟเวอร์ไม่สำเร็จจะมี Dialog error ขึ้นมาแสดงผลยัง Android Client และถ้าติดต่อกับเซิร์ฟเวอร์สำเร็จข้อมูลจะถูกแสดงผลยัง Android Client แบบ Listview



รูปที่ 3.7 โฟลว์ชาร์ตของแอพพลิเคชั่นรับค่าจาก Database

2) โฟลว์ชาร์ตของแอพพลิเคชั่นส่วนการสั่งอาหาร

เมื่อทำการเข้าสู่ส่วนการกรอกฟอร์มการสั่งอาหาร ผู้ใช้งานจะทำการกรอกข้อมูลตามฟอร์มที่กำหนดไว้ จากนั้นทำการกดปุ่ม Order Food จากข้อมูลที่จะถูกเก็บไว้ จากนั้นทำการเลือกรายการอาหารตามจำนวนที่ต้องการ ข้อมูลจะถูกส่งไปยังเซิร์ฟเวอร์และแสดงผลยังหน้าจอของพ่อครัว ส่วนใน Android Client จะแสดงค่าอาหารทั้งหมดที่ได้สั่งไว้พร้อมกับส่วนลด (ถ้ามี) กลับมาแสดงผลในหน้า Detail

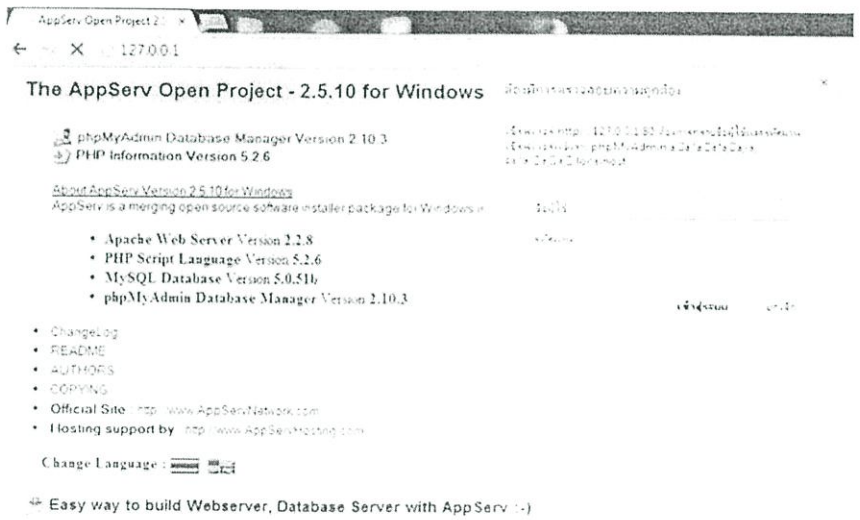


รูปที่ 3.8 โฟลว์ชาร์ตของส่วนแอพพลิเคชั่นการสั่งอาหาร

3.1.5 การจัดทำ Web Application

3.1.5.1 การจัดเตรียม Web Server

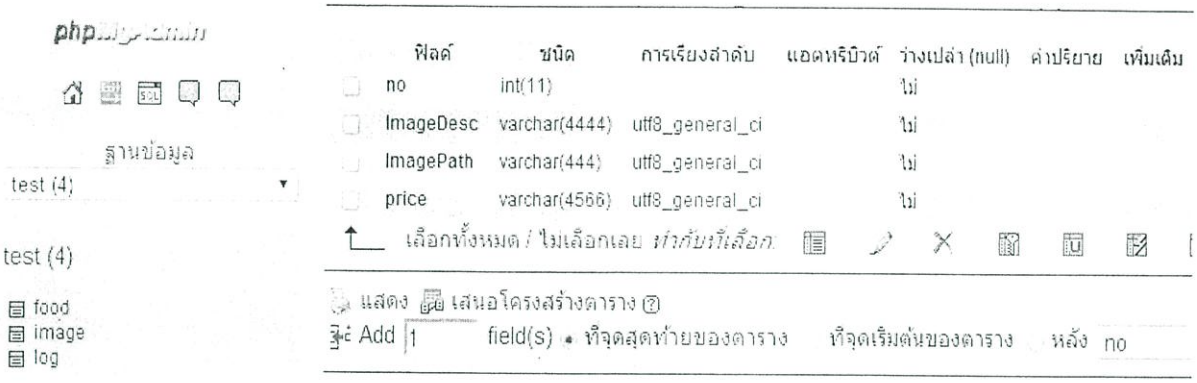
การจัดทำ Web Application ที่ใช้ในการส่งข้อมูลไปยัง Android Client จะใช้ภาษา PHP ในการเขียน ซึ่งจะถูกประมวลผลโดยคอมพิวเตอร์ที่ติดตั้งเป็นแบบ Web Server โดยใช้โปรแกรม AppServ เป็นตัวจัดการฐานข้อมูล ซึ่งใน AppServ จะรวบรวม Open Source Software ต่างๆประกอบไปด้วย MySQL , phpMyAdmin , Apache เป็นต้น



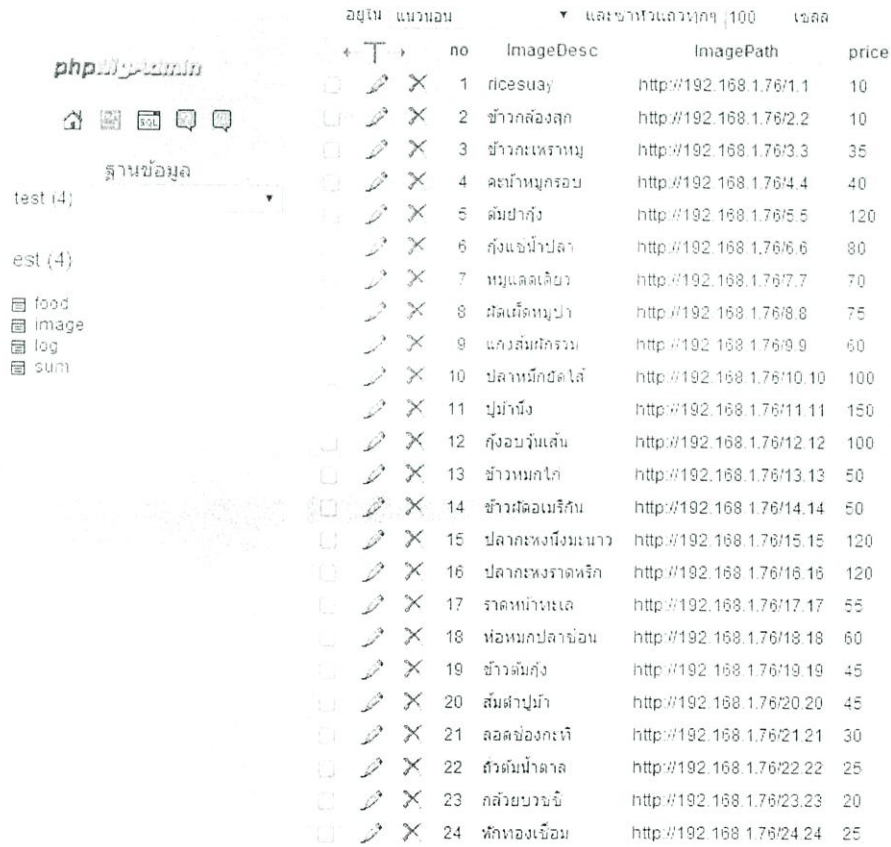
รูปที่ 3.9 แสดงหน้าล็อคอินก่อนจัดการฐานข้อมูล

3.1.5.2 การจัดทำระบบฐานข้อมูลที่ใช้เก็บข้อมูล

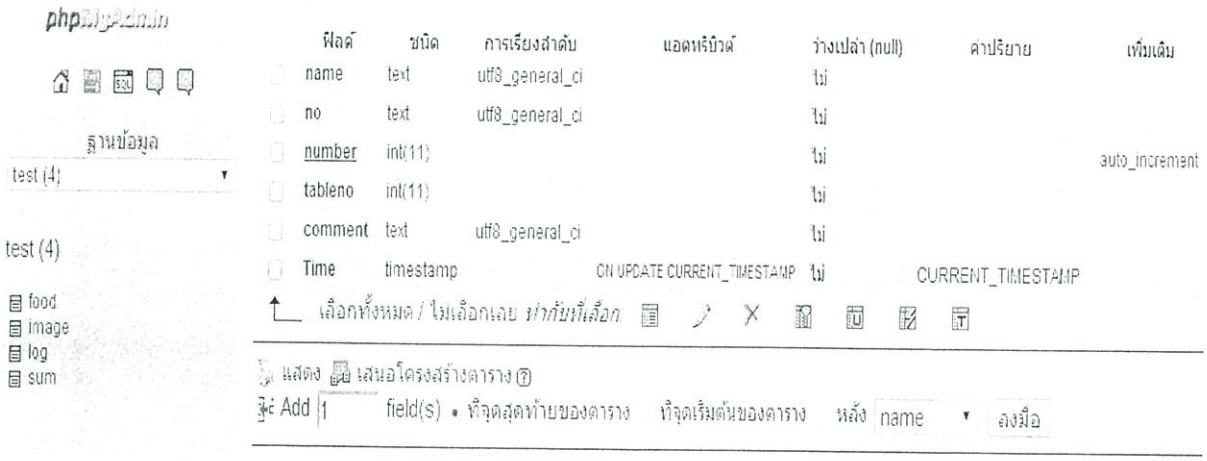
ทำการสร้างฐานข้อมูลชื่อ test ตารางเก็บข้อมูลอาหารชื่อ image ตารางเก็บรายการอาหารและข้อเสนอแนะของผู้สั่งชื่อ log โดยตัวแปรแต่ละคอลัมน์ของตาราง image คือ No ImageDesc ImagePath และ Price ดังรูป 3.10 รายละเอียดข้อมูล ดังรูป 3.11 ส่วนตัวแปรแต่ละคอลัมน์ของตาราง log คือ name no tableno number comment และ time ดังรูป 3.12 ส่วนตัวแปรแต่ล่ะคอลัมน์ของตาราง code คือ code_gen no time ดังรูป 3.13



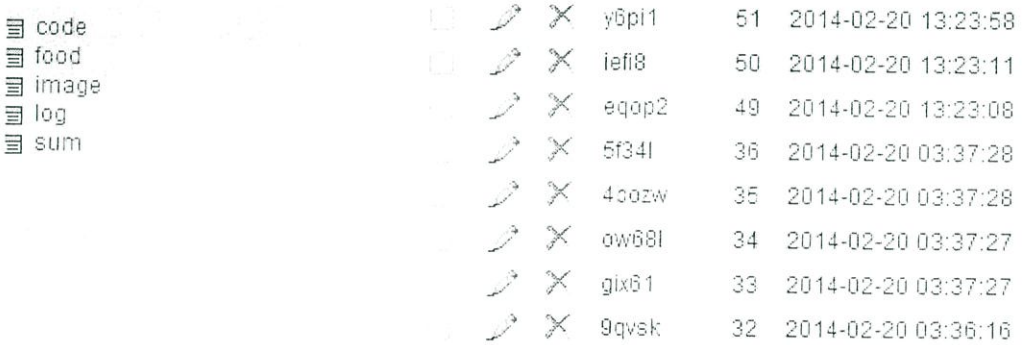
รูปที่ 3.10 แสดงตัวแปรต่างๆที่ใช้ในฐานข้อมูล ตารางชื่อ image



รูปที่ 3.11 แสดงข้อมูลที่อยู่ภายในฐานข้อมูล ตาราง image



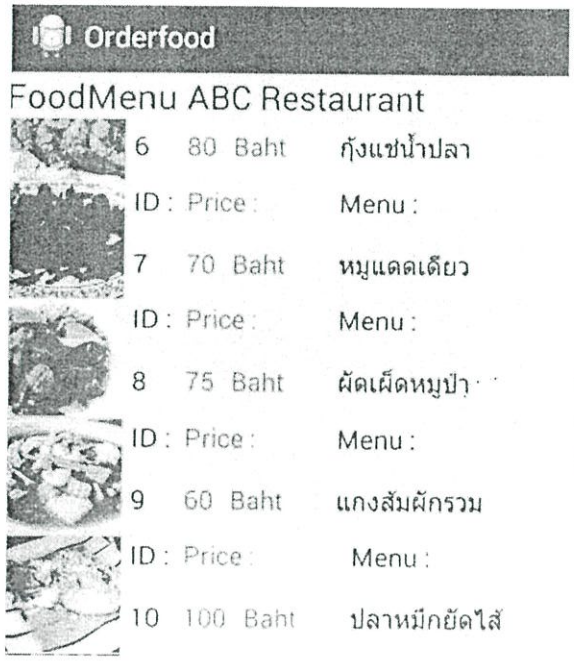
รูปที่ 3.12 แสดงตัวแปรต่างๆที่ใช้ในฐานข้อมูล ตาราง log



รูปที่ 3.13 แสดงข้อมูลที่อยู่ภายในฐานข้อมูล ตาราง code

3.1.5.3 การรับข้อมูลจากเซิร์ฟเวอร์มายัง Android Client

เมื่อ Android Client ทำการติดต่อกับเซิร์ฟเวอร์ Web Application จะทำหน้าที่ประมวลผลให้กับ Server ผ่านไฟล์ getJSON.php โดยจะทำการดึงค่าจากฐานข้อมูลและส่งไปยัง Android Client เพื่อแสดงผล ดังรูปที่ 3.14

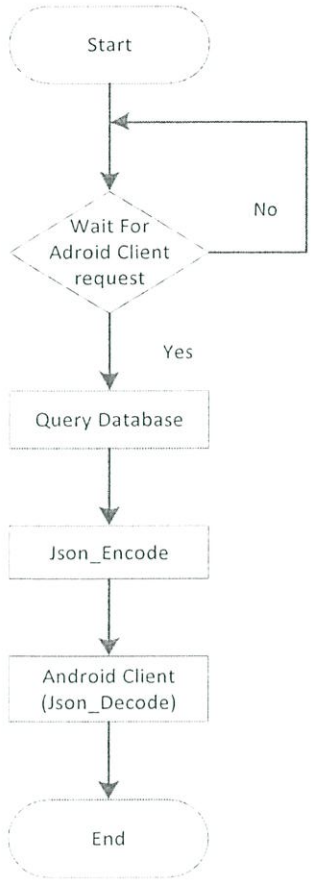


6	กุ้งแช่น้ำปลา	http://192.168.1.76/6.6	80
7	หมูแดดเดียว	http://192.168.1.76/7.7	70
8	ผัดเผ็ดหมูป่า	http://192.168.1.76/8.8	75
9	แกงส้มผักรวม	http://192.168.1.76/9.9	60
10	ปลาหมึกยัดไส้	http://192.168.1.76/10.10	100
11	ปูม้านึ่ง	http://192.168.1.76/11.11	150
12	กุ้งอบวุ้นเส้น	http://192.168.1.76/12.12	100

รูปที่ 3.14 แสดงผลการดึงค่ามาแสดงผลยัง Android Client

1) โฟลว์ชาร์ตของส่วนของการรับค่าจากเซิร์ฟเวอร์มายัง Android Client

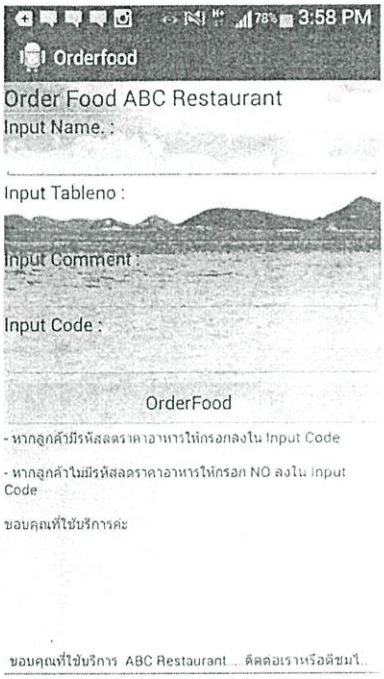
เมื่อ Android Client ทำการร้องขอไปยัง Server โดยจะทำการเลือกข้อมูลที่จะเข้ามายังตัว Client เมื่อทำการร้องขอสำเร็จ Web Application จะทำการแปลงข้อมูลให้อยู่ใน JSON Form และจะส่งค่าไปยัง Android Client โดยใน Android จะมี Activity ที่ทำการ Encode JSON แล้วแสดงผลบนโทรศัพท์



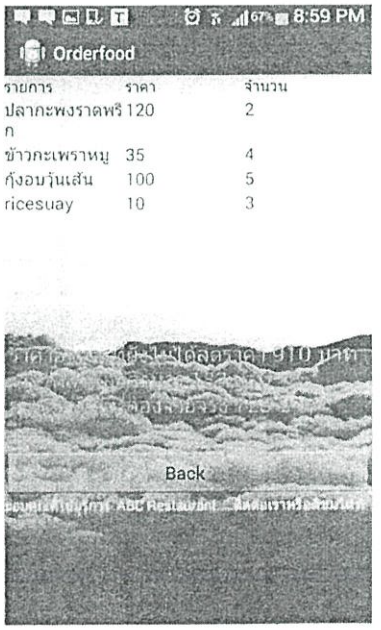
รูปที่ 3.15 โฟลว์ชาร์ตของส่วนของการรับค่าจากเซิร์ฟเวอร์มายัง Android Client

3.1.5.4 การส่งข้อมูลจาก Android บน Form ไปบันทึกที่ Server Database

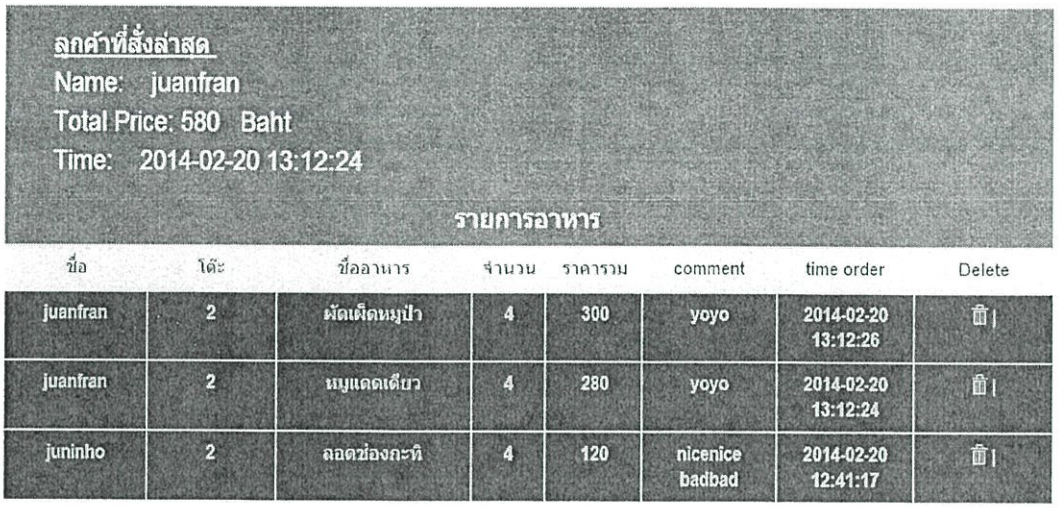
เมื่อ Android Client ทำการติดต่อกับเซิร์ฟเวอร์ ผู้ใช้ทำการกรอกฟอร์มตามที่กำหนดไว้และกด Order Food เพื่อตรวจสอบรายการอาหารและสั่งอาหาร หลังจากนั้นจะสามารถเช็คว่าได้รับส่วนลดเท่าไร และสั่งอาหารชนิดใดไปบ้างในหน้าถัดไป



รูปที่ 3.16 แสดงหน้าการกรอกฟอร์มของแอพพลิเคชั่น



รูปที่ 3.17 แสดงหน้าสรุปการสั่งอาหารของแอพพลิเคชั่น

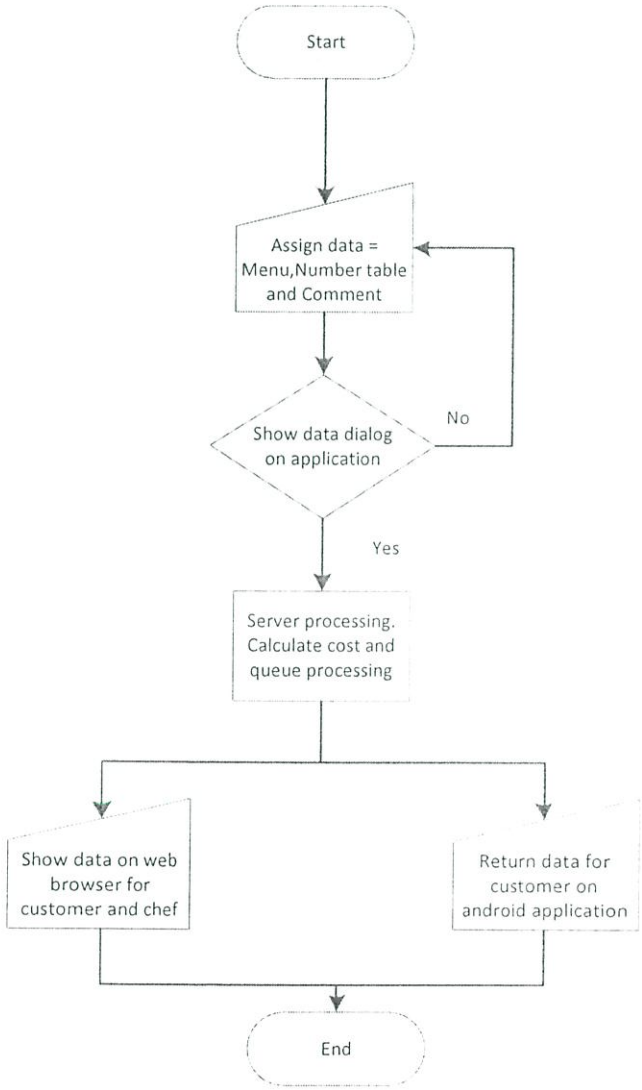


รูปที่ 3.18 แสดงหน้าการกรอกฟอร์มของแอพพลิเคชั่น

รูปที่ 3.19 แสดงหน้าการจัดการโค๊ดส่วนลดราคาอาหาร

1) โฟลว์ชาร์ตของส่วนของการส่งข้อมูลไปยังเซิร์ฟเวอร์

เมื่อ Android Client ทำการร้องขอไปยัง Server โดยจะทำการเลือกข้อมูลที่จะเข้ามายังตัว Client เมื่อทำการร้องขอสำเร็จ Web Application จะทำการแปลงข้อมูลให้อยู่ใน Json Form และจะส่งค่าไปยัง Android Client โดยใน Android จะมี Activity ที่ทำการ Encode Json แล้วแสดงผลบนโทรศัพท์

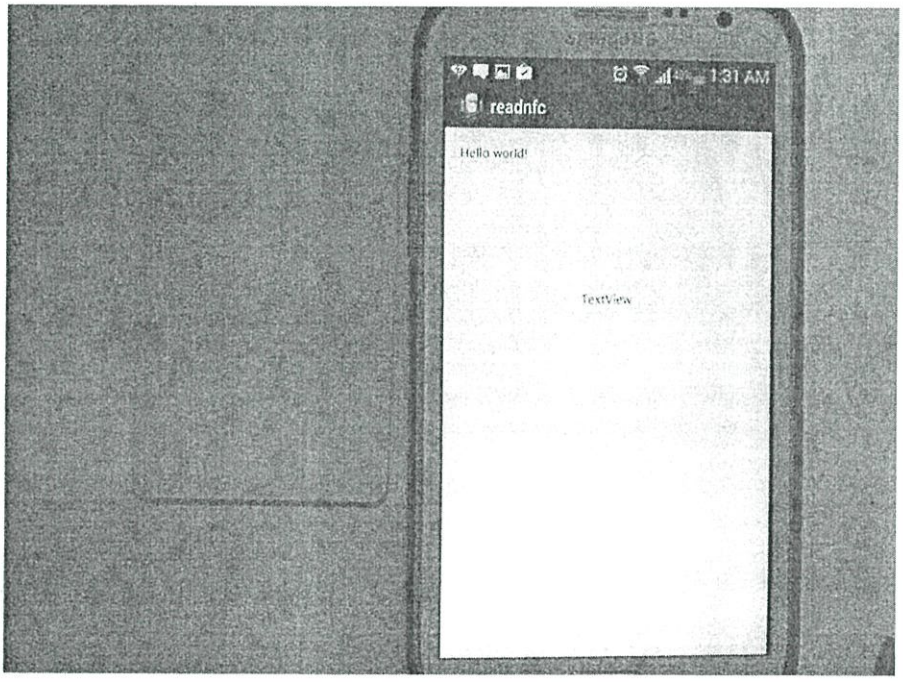


รูปที่ 3.20 โฟลว์ชาร์ตของส่วนของการประมวลผลในเซิร์ฟเวอร์

3.1.6 การเชื่อมต่อ NFC และ QR Code เข้ากับแอพพลิเคชั่น

3.1.6.1 การเชื่อมต่อ NFC Tag กับ Android Client

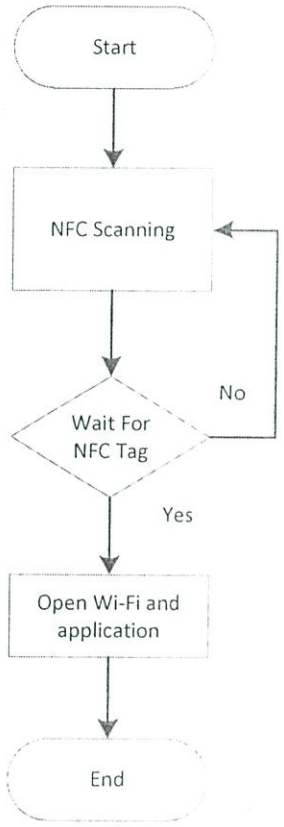
เมื่อนำ NFC Tag ที่ได้ทำการ Write ข้อมูลไว้ โดย NFC Tag จะมีข้อมูลเก็บไว้ มาเชื่อมต่อกับ Android Client เพื่อนำข้อมูลใน Tag มาใช้ในการออกคำสั่งเชื่อมต่อเครือข่ายไร้สาย (Wi-Fi) และทำการเปิดแอพพลิเคชั่นสั่งอาหาร (Order Food) อัตโนมัติ โดยแอพพลิเคชั่นที่มีหน้าที่อ่าน Tag ชื่อ "Readnfc" ดังรูปที่ 3.21



รูปที่ 3.21 แสดงหน้าแอพพลิเคชั่น Readnfc และ Tag NFC

1) โฟลว์ชาร์ตของส่วนการสื่อสารระหว่าง NFC Tag และ Android Client

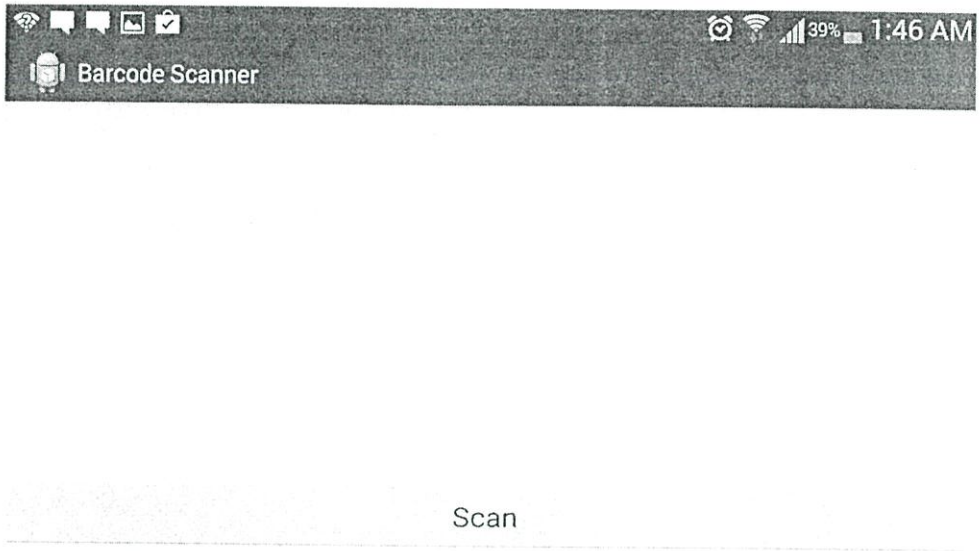
เมื่อ Android Client ทำการเปิดแอพพลิเคชั่น Readnfc จะทำการสแกน NFC Tag เพื่อนำข้อมูลที่บันทึกใน NFC Tag มาใช้ในการสั่งงานเปิดเครือข่ายไร้สายที่มีรหัสผ่านได้อัตโนมัติ และเปิดแอพพลิเคชั่นสั่งอาหารทันที



รูปที่ 3.22 โฟลว์ชาร์ตของส่วนการทำงานระหว่าง NFC-Tag และ Android Client

3.1.6.2 การเชื่อมต่อ QR-Code กับ Android Client

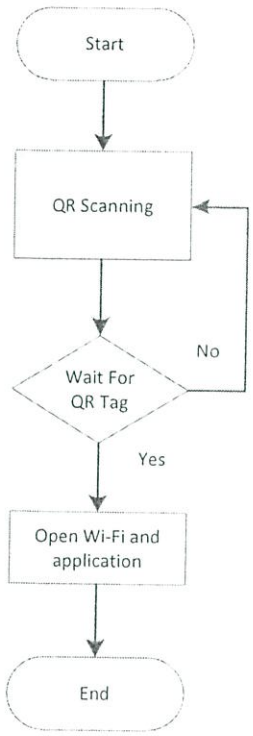
เมื่อนำ QR-Code ที่ได้ทำการ Write ข้อมูลไว้ โดย QR-Code จะมีข้อมูลเก็บไว้ มาเชื่อมต่อกับ Android Client เพื่อนำข้อมูลใน Tag มาใช้ในการออกคำสั่งเชื่อมต่อเครือข่ายไร้สาย (Wi-Fi) และทำการเปิดแอพพลิเคชั่นสั่งอาหาร (Order Food) อัตโนมัติ โดยแอพพลิเคชั่นที่มีหน้าที่อ่าน QR-Code ใช้ชื่อว่า “BarcodeScanner” ดังรูปที่ 3.23



รูปที่ 3.23 แสดงหน้าแอพพลิเคชั่น Barcode Scanner

1) โฟลว์ชาร์ตของส่วนการสื่อสารระหว่าง QR Code และ Android Client

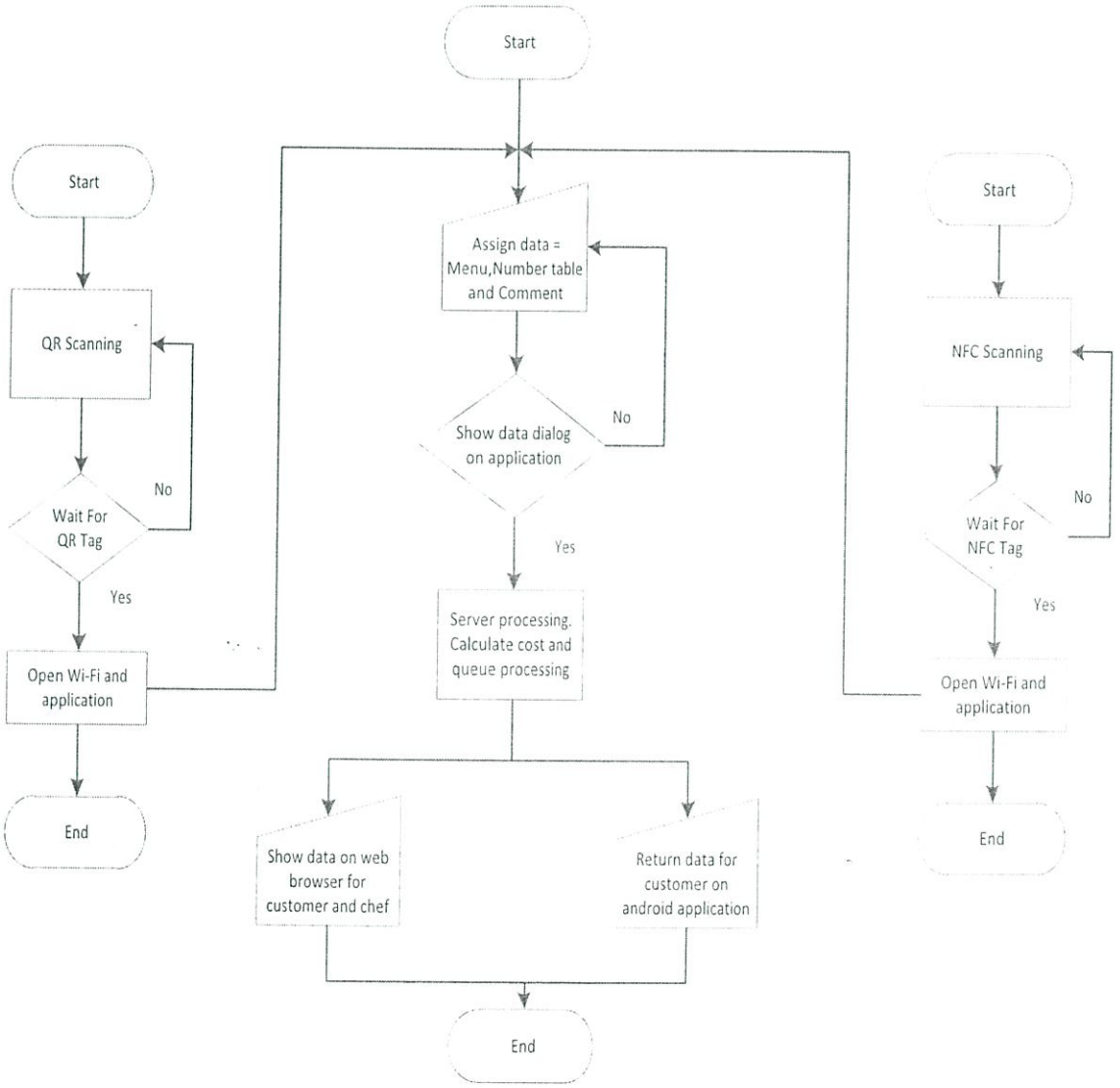
เมื่อ Android Client ทำการเปิดแอพพลิเคชั่น Barcode Scanner จะทำการสแกน Barcode เพื่อนำข้อมูลที่บันทึกใน QR Code มาใช้ในการสั่งงานเปิดเครือข่ายไร้สายที่มีรหัสผ่านได้อัตโนมัติ และเปิดแอพพลิเคชั่นสั่งอาหารทันที



รูปที่ 3.24 โฟลว์ชาร์ตของส่วนการทำงานระหว่าง QR Code และ Android Client

3.1.5.3 โฟลว์ชาร์ตแสดงภาพรวมการทำงานของระบบสั่งอาหารผ่าน Android

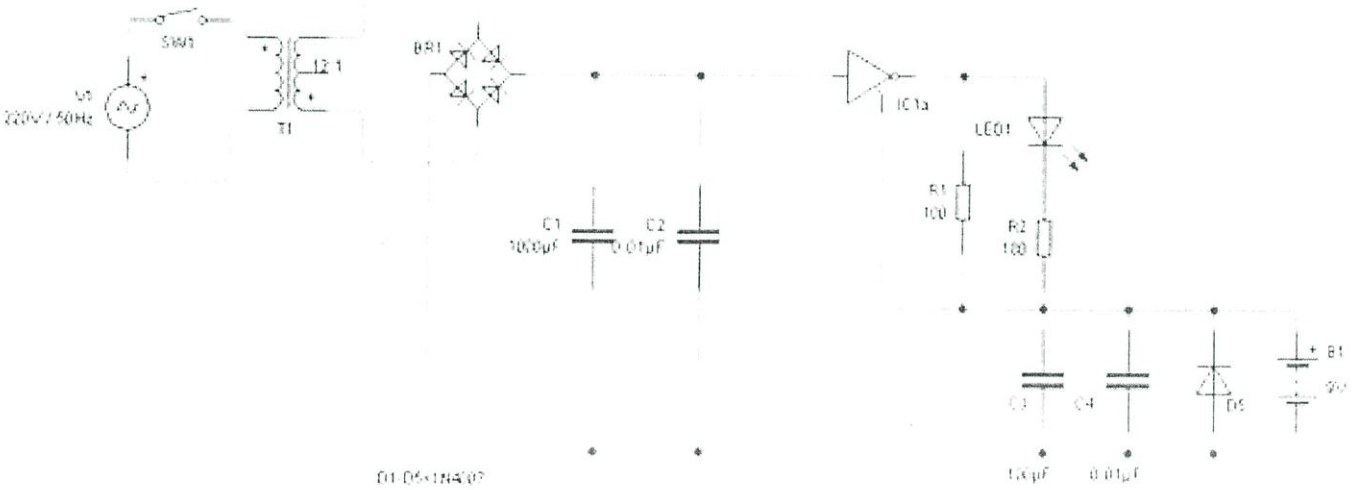
ในส่วนที่เป็นโฟลว์ชาร์ตสีแดง (ด้านซ้ายของโฟลว์ชาร์ตรวม) จะเป็นส่วนของการอ่าน Tag NFC ในส่วนที่เป็นโฟลว์ชาร์ตสีเขียว (ด้านขวาของโฟลว์ชาร์ตรวม) จะเป็นส่วนของการอ่าน QR-Code และในส่วนที่เป็นโฟลว์ชาร์ตสีฟ้า (ตรงกลางของโฟลว์ชาร์ตรวม) จะเป็นส่วนของการทำงานของแอพพลิเคชั่นสั่งอาหาร (Order Food)



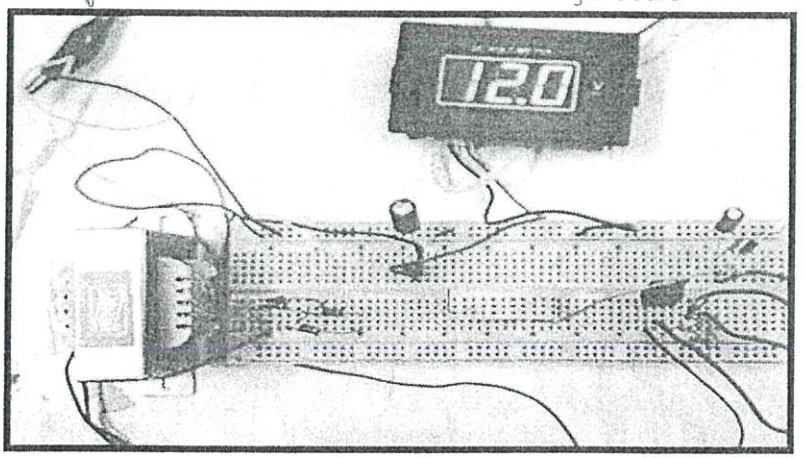
รูปที่ 3.25 โฟลว์ชาร์ตแสดงภาพรวมการทำงานของระบบสั่งอาหารผ่าน Android

3.1.7 การออกแบบวงจร Charger Board

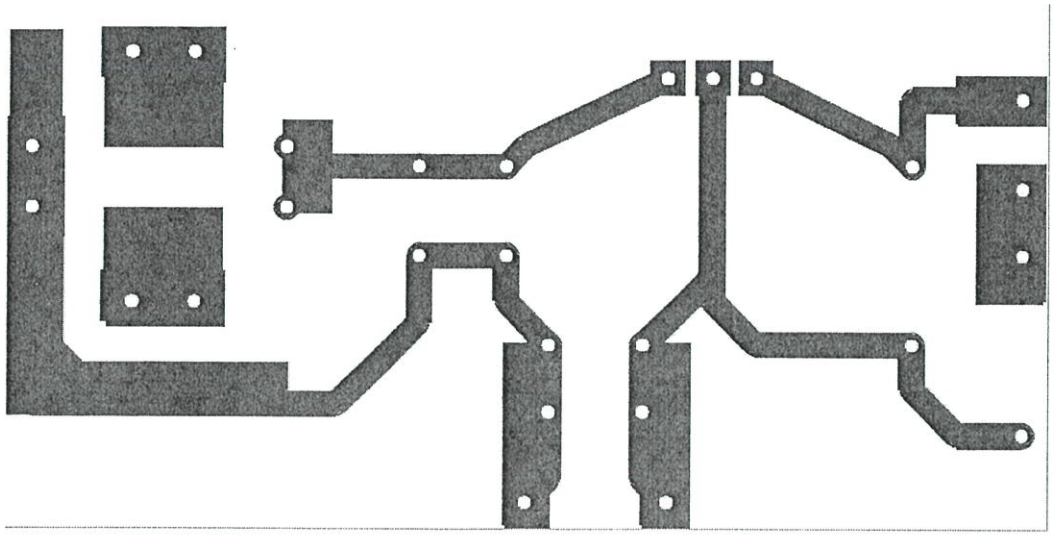
หน้าที่: เป็นสวิตส์ เปิด/ปิด การทำงานของหุ่นยนต์ และจ่ายไฟเลี้ยงแก่วงจรทั้งหมด พร้อมทั้งสามารถชาร์ต Battery 9 V แก่ Battery ได้ โดยการทำงาน สามารถใช้งานได้จริงโดยการทำงานของวงจรนี้คือนำไฟ 220 VAC ผ่านวงจร Bridge rectifier เพื่อปรับระดับสัญญาณ ผ่าน IC 7809 ผ่านการชาร์จและดิสชาร์จประจุ ของตัวเก็บประจุ แล้วแปลงเป็น 9 VDC ซึ่งเมื่อ Battery เข้าที่ชาร์จแล้ว LED จะทำงานโดยจะติดสว่างซึ่ง การทดลองนี้ได้ทดลองกับเครื่องทดลองในห้องทดลอง โดยไฟที่ออกมาที่ Load เราจะใช้เครื่องมิเตอร์วัดค่าซึ่งค่าที่ออกมาก็อยู่ประมาณที่ 9-10 VDC



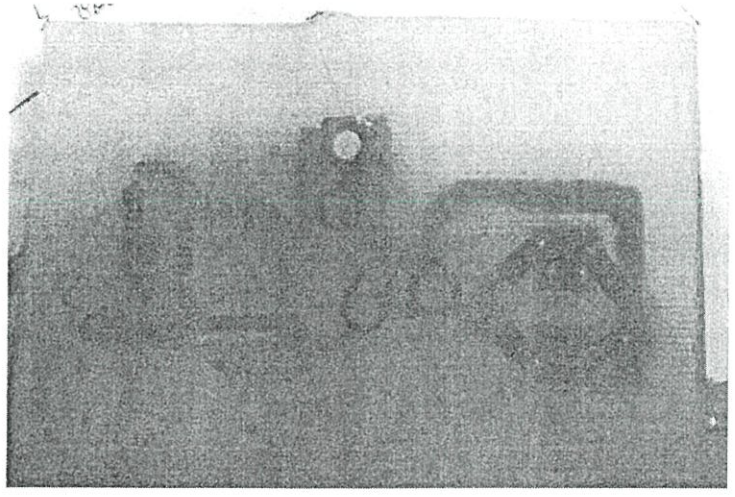
รูปที่ 3.26 แสดง Schematic ของวงจร Charger Board



รูปที่ 3.27 แสดงการต่อวงจรลงบนบอร์ดทดลอง



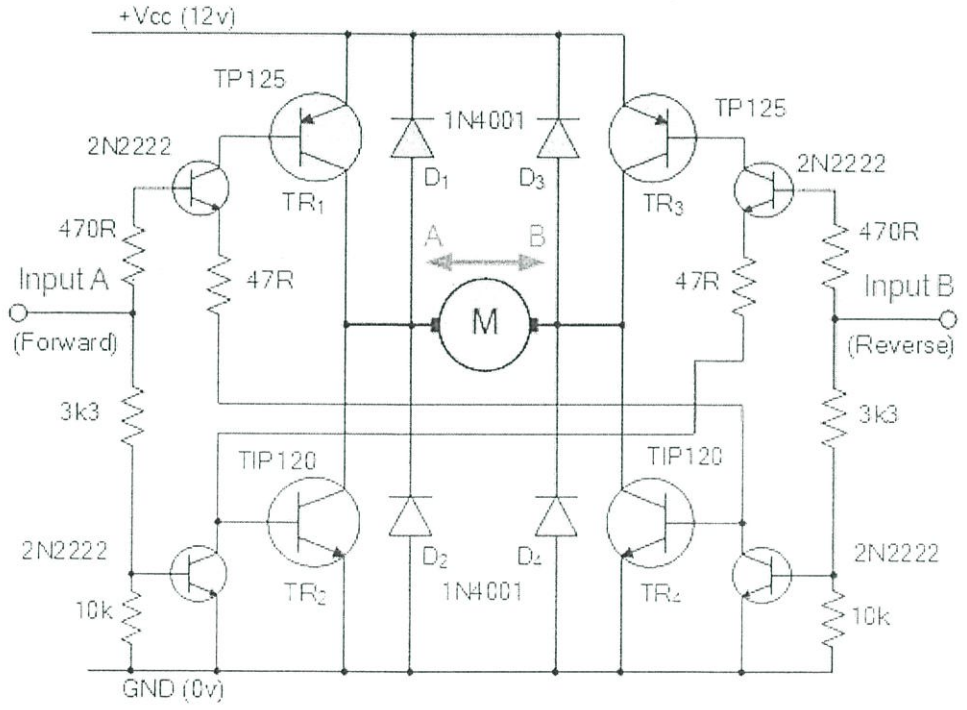
รูปที่ 3.28 แสดงการออกแบบลายวงจร Charger Board



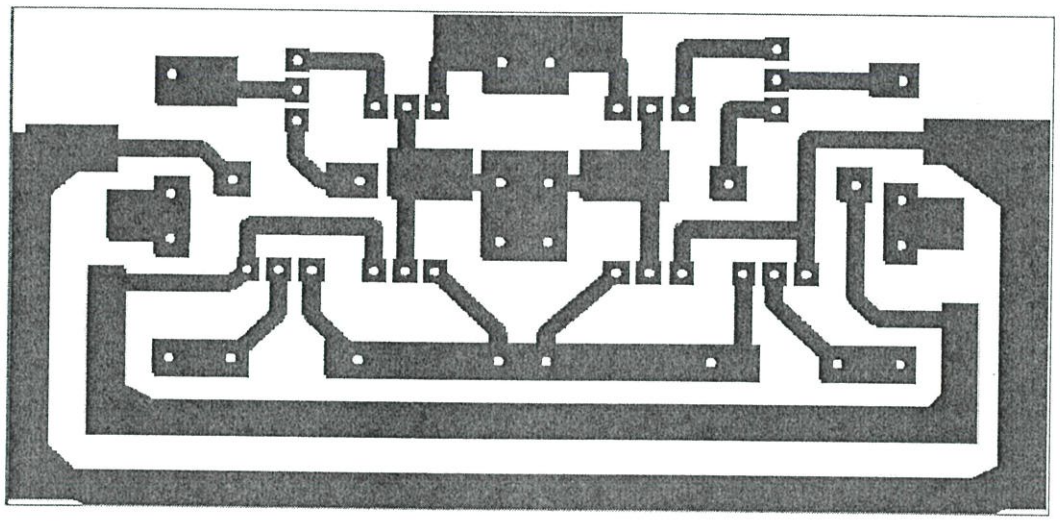
รูปที่ 3.29 แสดงการต่อวงจร Charger Board ลงบนแผ่น PCB

3.1.8 วงจร Motor Controller Board

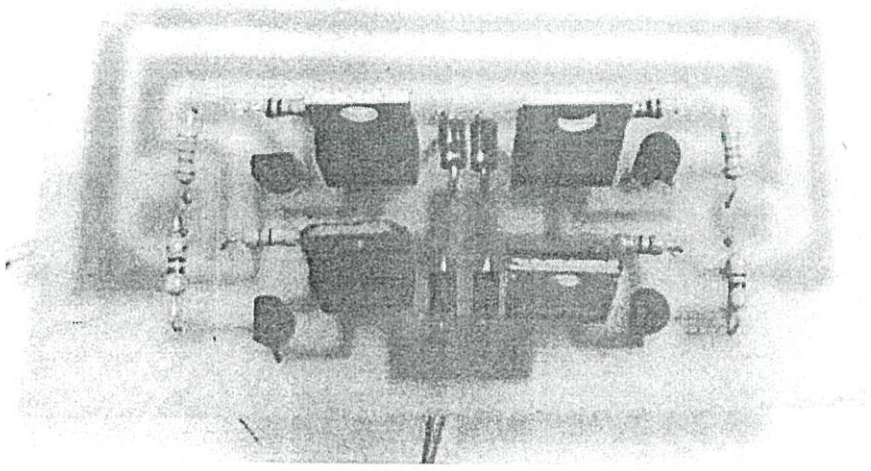
หน้าที่: ควบคุมการหมุนของมอเตอร์ ทั้งล้อซ้ายและขวา โดยจะมีวงจร H-bridge เป็นตัวขับมอเตอร์ หลักการของวงจรนั้นจะประกอบไปด้วยสวิตช์ 4 ตัว นั่นก็คือ S1 , S2, S3 และ S4 มีกระแสไฟฟ้าไหลเข้าสู่มอเตอร์ และเมื่อเราทำการเปิด (On) สวิตช์ S1 และ S3 พร้อมกัน จะเป็นการเชื่อมวงจรทำให้มีกระแสไฟฟ้าไหลผ่านมอเตอร์ จากขั้วบวกของมอเตอร์ไปยังขั้วลบของมอเตอร์จึงทำให้มอเตอร์สามารถหมุนได้ในทิศทางฟอร์เวิร์ด (เคลื่อนที่ไปข้างหน้า)



รูปที่ 3.30 แสดง Schematic ของวงจร Motor Controller Board



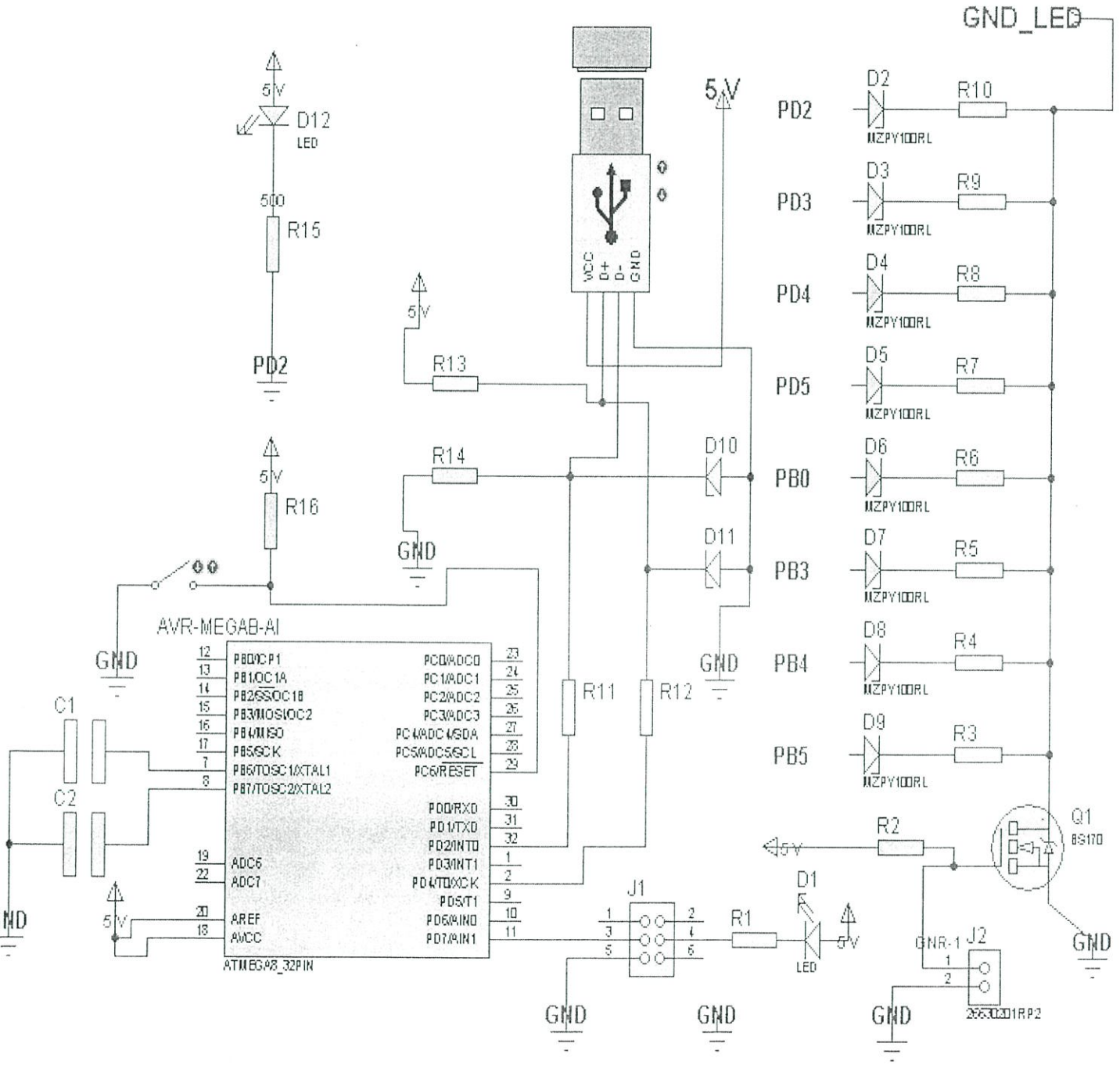
รูปที่ 3.31 แสดงการออกแบบรายวงจร Motor Controller Board



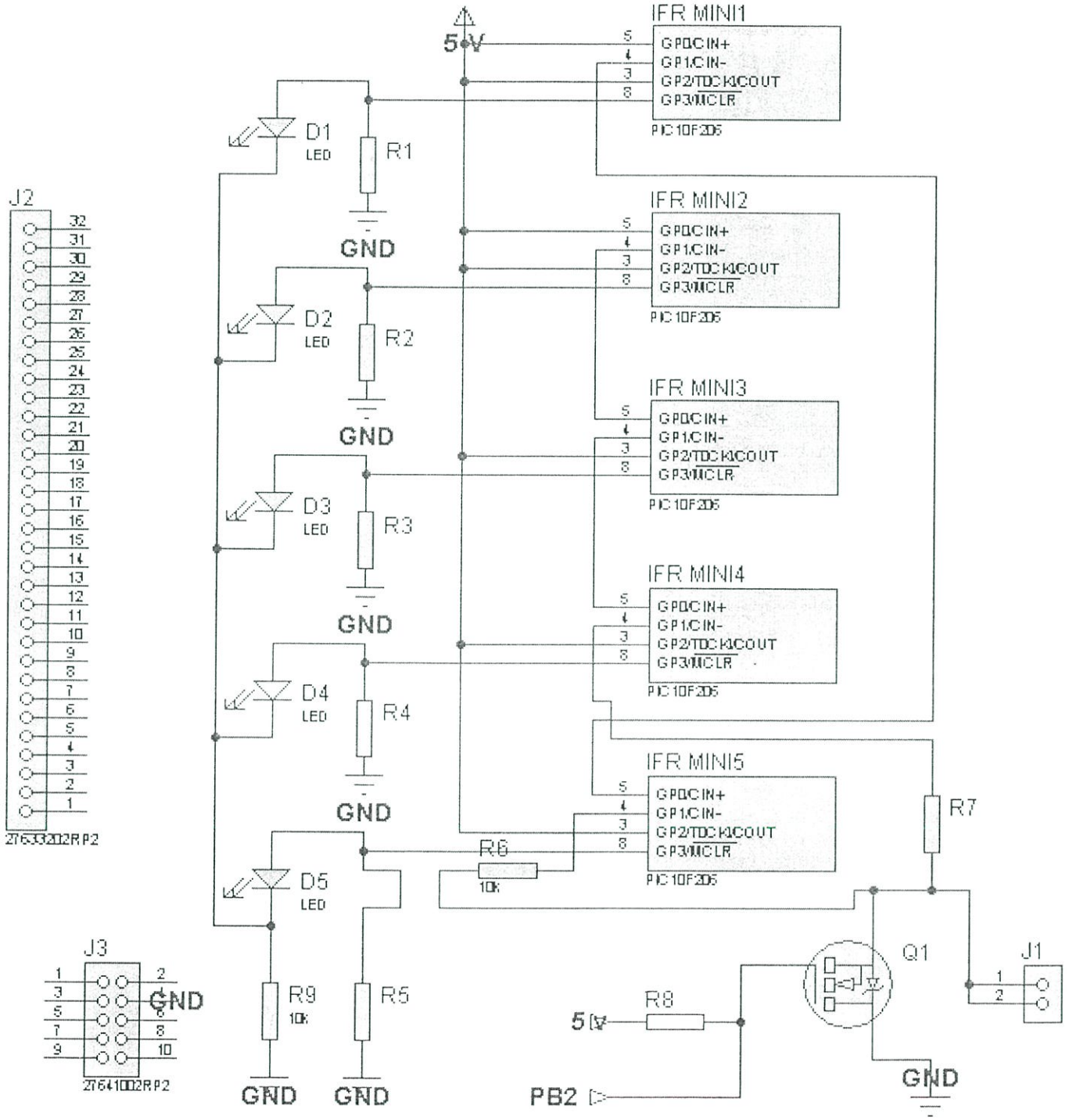
รูปที่ 3.32 แสดงการต่อวงจร Motor Controller Board ลงบนแผ่น PCB

3.1.9 วงจร Microcontroller

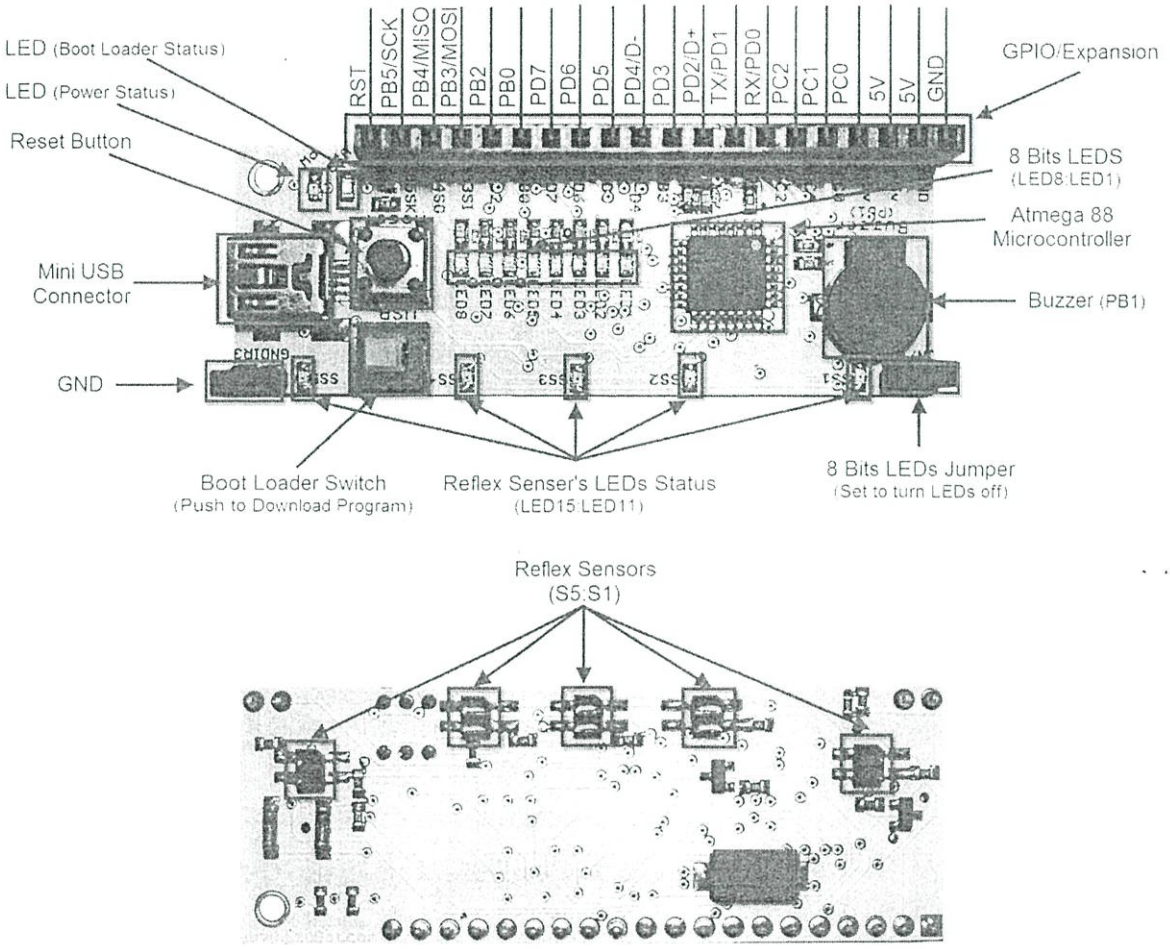
หน้าที่: เป็นตัวโปรแกรมคำสั่ง ด้วยภาษา C++ หรือ Assembly ในโปรแกรม AVR studio 4 และ Arduino IDE และทำการโปรแกรมคำสั่ง ผ่านพอร์ต USB เพื่อควบคุม การเคลื่อนที่ของหุ่นยนต์ ให้สามารถไปยังเป้าหมายได้ถูกต้อง โดยจะประกอบไปด้วยเซ็นเซอร์อินฟราเรดทั้งหมด 5 ตัว โดย Arduino ATTMEGA88 จะทำหน้าที่ในการรับอินพุตเป็นค่าลอจิก จากเซ็นเซอร์อินฟราเรด หากเซ็นเซอร์ตัวใดจับสีดำได้ จะส่ง Output เป็นลอจิก 1 หากเซ็นเซอร์ตัวใดจับสีขาวได้ จะส่ง Output เป็นลอจิก 0 เพื่อนำ Output ที่ได้ไปประมวลผลในการเลือกเส้นทางของหุ่นยนต์ต่อไป



รูปที่ 3.33 แสดง Schematic ของวงจร Motor Controller Board



รูปที่ 3.34 แสดง Schematic ของวงจร Motor Controller Board (2)



รูปที่ 3.35 แสดง Arduino ATTMEGA 88

3.2 เครื่องมือที่ใช้ในการทดลอง

ในการออกแบบโดยรวมจะใช้เครื่องมือเป็นโปรแกรม ซอฟต์แวร์ เป็นส่วนใหญ่ โดยจะแบ่งโปรแกรมซอฟต์แวร์ที่ใช้ในการออกแบบในส่วนต่างๆแยกได้ดังนี้

3.2.1 ส่วนของแอพพลิเคชั่นรับส่งข้อมูลบนมือถือ
ซอร์ฟแวร์ที่ใช้ในการออกแบบมีดังนี้

- Eclipse
- Android Development Tools Plugin for eclipse (ADT)
- Software Development Kit (SDK)
- Java Development Kit (JDK)

3.2.2 ส่วนของเซิร์ฟเวอร์

- Notepad ++
- AppServe Open Project

3.2.3 ส่วนวงจรของหุ่นยนต์

- Power Supply
- Signal Generator
- Oscilloscope

3.2.4 ส่วนโปรแกรมหุ่นยนต์

- AVR Studio4
- Arduino IDE
- DEV C++

3.2.5 ส่วนอื่นๆ

- โทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์
- NFC Tag
- QR Code
- Computer
- USB Connecter

3.3 การจัดเก็บผลการทดลอง

3.3.1 การทดสอบแอพพลิเคชั่น

3.3.1.1 ทำการเชื่อมต่ออินเตอร์เน็ตบนโทรศัพท์มือถือที่จะใช้ทดสอบ

3.3.1.2 ฝั่งเซิร์ฟเวอร์เชื่อมต่ออินเตอร์เน็ต

3.3.1.3 เปิดแอพพลิเคชั่นในการเลือกเมนูอาหารบนโทรศัพท์มือถือ

3.3.1.4 ทำการดึงค่าเมนูอาหารจาก Server

3.3.1.5 ทำการเลือกเมนูอาหาร และส่งเมนูที่เลือก ไปประมวลผลยัง Server

3.3.1.6 ทำการทดสอบการทำงานของ QR Code และ NFC Tag

3.3.2 การทดสอบหุ่นยนต์

3.3.2.1 ทดสอบการชาร์จ Battery ของหุ่นยนต์เข้า Battery ผ่านไฟบ้าน 220 VAC พร้อมโชว์ค่า Voltage ที่เหลือของ Battery บนจอ LCD

3.3.2.2 ทดสอบการหมุนของมอเตอร์หุ่นยนต์เมื่อมีการกด Switch ในวงจร Charger Board

3.3.2.3 ทดสอบการรับค่าความเข้มของสีขาวและสีดำ จากเซ็นเซอร์อินฟราเรด

3.3.2.4 ทดสอบการปรับความความเร็วมอเตอร์ จากการรับอินพุตเป็น ความเข้มสีดำ และสีขาวจากเซ็นเซอร์อินฟราเรด เพื่อให้มอเตอร์คงสภาวะการเดินทางและเดินบนเส้นสีดำได้

3.3.2.5 ทดสอบการวิ่งของหุ่นยนต์ให้ไปยังเป้าหมายได้ถูกต้อง โดยควบคุมจากเซ็นเซอร์อินฟราเรด ในการรับค่าความเข้มของสีดำและสีขาว

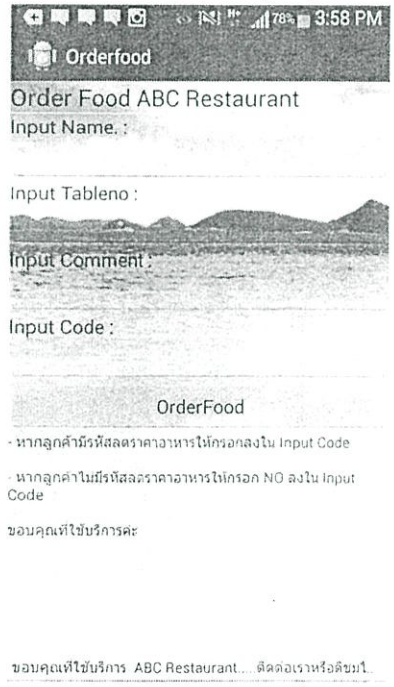
บทที่ 4

ผลการทดลอง

ในส่วนของผลการทดลองจะประกอบไปด้วย ผลการทดลองของการติดต่อกับเซิร์ฟเวอร์ ผลการทดลองส่วนของ User Interface ในการกรอกข้อมูลลงในฟอร์ม ผลการทดลองส่วนของหน้าจอแสดงผลของพ่อครัวและจัดการฐานข้อมูล ผลการทดลองส่วนของการติดต่อสื่อสารกับ NFC และ QR Code ผลการทดสอบวงจร Motor Controller Board ผลการทดสอบการรับค่าความเข้มสีขาวและสีดำของเซ็นเซอร์อินฟราเรด และผลการทดสอบการปรับค่าความเร็วของมอเตอร์

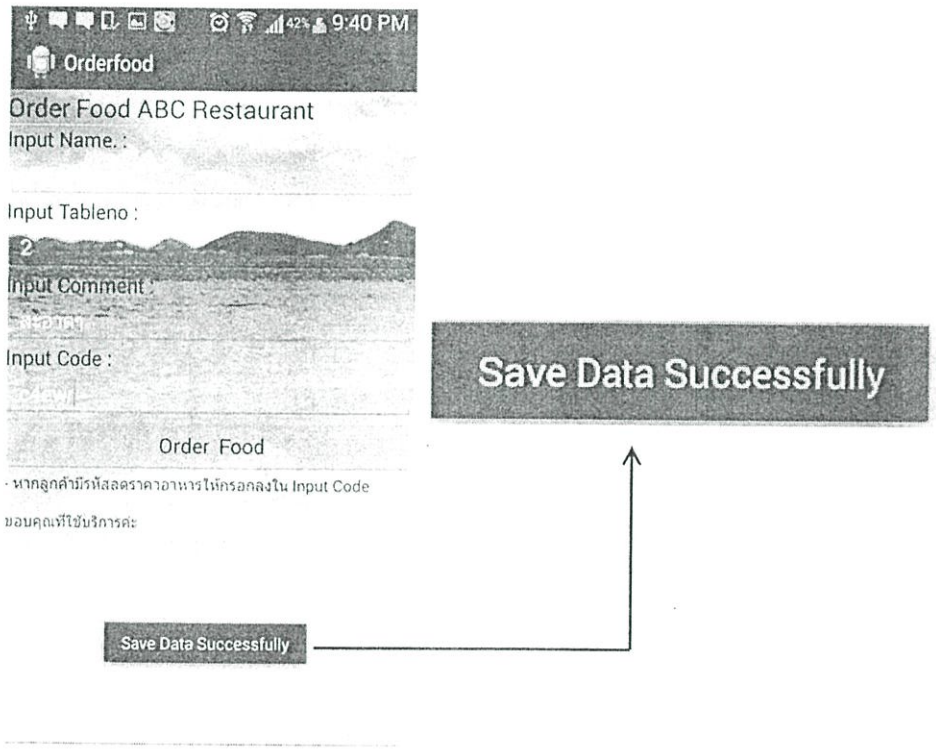
4.1 ผลการทดลองส่วนของ User Interface ในการกรอกข้อมูลลงในฟอร์ม

ในส่วนนี้ได้ทำการสร้าง User Interface ของแอพพลิเคชั่นขึ้นมาโดยแสดงผล ดังรูปที่ 4.1 และจะทำการทดลองกรอกข้อมูลลงไปในตามหัวข้อในแต่ละฟอร์ม จะมีการกรอกข้อมูลเลขโต๊ะ ชื่อผู้สั่ง คำแนะนำสำหรับทางร้าน และรหัสสำหรับลดราคาอาหาร(ถ้ามี)



รูปที่ 4.1 หน้า User Interface ส่วนการกรอกฟอร์มของแอพพลิเคชั่น

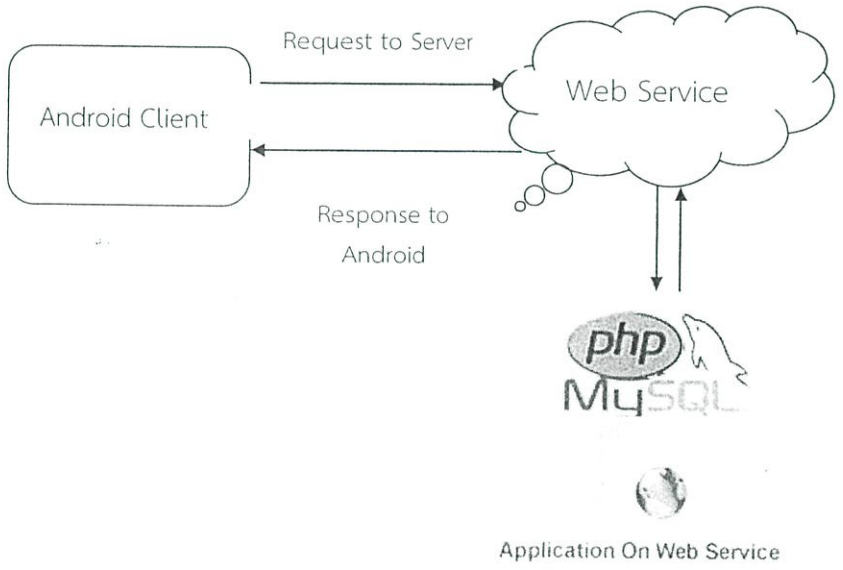
เมื่อทำการกรอกข้อมูลลงไปที่ได้กำหนดไว้ครบและทำการกดปุ่ม “Order Food” แอพพลิเคชั่นทำการบันทึกข้อมูลเรียบร้อยแล้วจะมี Dialog แสดงว่า “Save Data Successfully” จากนั้นแอพพลิเคชั่นจะ เปลี่ยนไปยังหน้ารายการอาหาร ดังรูปที่ 4.2



รูปที่ 4.2 ผลของการกดปุ่ม Order Food เมื่อทำการส่งข้อมูลเข้าไปยังฐานข้อมูลสำเร็จ

4.2 ผลการทดลองของการติดต่อกับเซิร์ฟเวอร์

การทดลองส่วนนี้จะทำการทดสอบการดึงข้อมูลจากเซิร์ฟเวอร์เข้ามายังโทรศัพท์มือถือแสดงผลเป็น ดังรูปที่ 4.3 และการทดสอบการอ่านข้อมูลที่ถูกเลือกของหน้า User Interface(UI) ที่เป็นแบบ Listview



รูปที่ 4.3 รูปแบบการดึงข้อมูลเข้าสู่โทรศัพท์

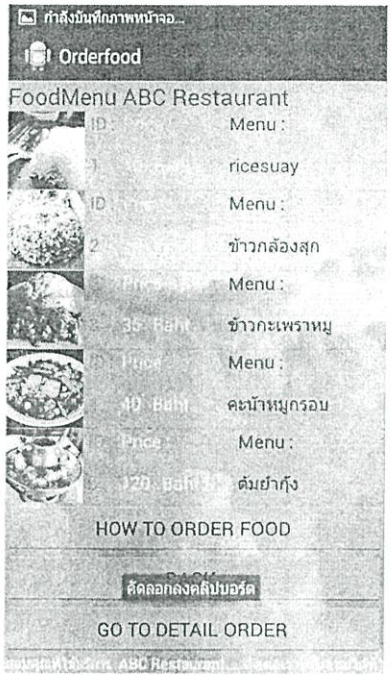
4.2.1 ผลการทดลองในส่วนการดึงข้อมูลจาก Database โดยใช้ PHP และ JSON

ในการทดลองนี้จะทำการรับข้อมูลจากเซิร์ฟเวอร์ โดยมี Database ที่สร้างจากโปรแกรมสำเร็จรูป Appserv โดย Android Client จะร้องขอในการดึงข้อมูลกับ Web Service ต่อจากนั้นจะมีการตอบสนองจาก Web Service เพื่อส่งค่าให้ Android Client โดย Web Service จะมีส่วนประกอบคือ PHP Script และ Database MySQL โดยในรูปที่ 4.4 จะแสดงข้อมูลที่จะส่งให้ Android Client

no	ImageDesc	ImagePath	price
1	ricesuay	http://192.168.1.76/1.1	10
2	ข้าวกล่องสุก	http://192.168.1.76/2.2	10
3	ข้าวกะเหรา่หมู	http://192.168.1.76/3.3	35
4	คะน้าหมูกรอบ	http://192.168.1.76/4.4	40
5	ต้มยำกุ้ง	http://192.168.1.76/5.5	120
6	กุ้งแช่น้ำปลา	http://192.168.1.76/6.6	80
7	หมูแดดเดียว	http://192.168.1.76/7.7	70
8	ผัดเผ็ดหมูป่า	http://192.168.1.76/8.8	75
9	แกงส้มผักรวม	http://192.168.1.76/9.9	60
10	ปลาหมึกยัดไส้	http://192.168.1.76/10.10	100
11	ปูม้านึง	http://192.168.1.76/11.11	150
12	กุ้งอบวุ้นเส้น	http://192.168.1.76/12.12	100
13	ข้าวหมกไก่	http://192.168.1.76/13.13	50
14	ข้าวผัดอเมริกัน	http://192.168.1.76/14.14	50
15	ปลากะพงนึงมะนาว	http://192.168.1.76/15.15	120
16	ปลากะพงราดพริก	http://192.168.1.76/16.16	120
17	ราดหน้าทะเล	http://192.168.1.76/17.17	55
18	ห่อหมกปลาย่อน	http://192.168.1.76/18.18	60
19	ข้าวต้มกุ้ง	http://192.168.1.76/19.19	45
20	ส้มตำปูม้า	http://192.168.1.76/20.20	45
21	ลอดช่องกะทิ	http://192.168.1.76/21.21	30
22	ถั่วต้มน้ำตาล	http://192.168.1.76/22.22	25
23	กล้วยบวชชี	http://192.168.1.76/23.23	20
24	ฟักทองเชื่อม	http://192.168.1.76/24.24	25

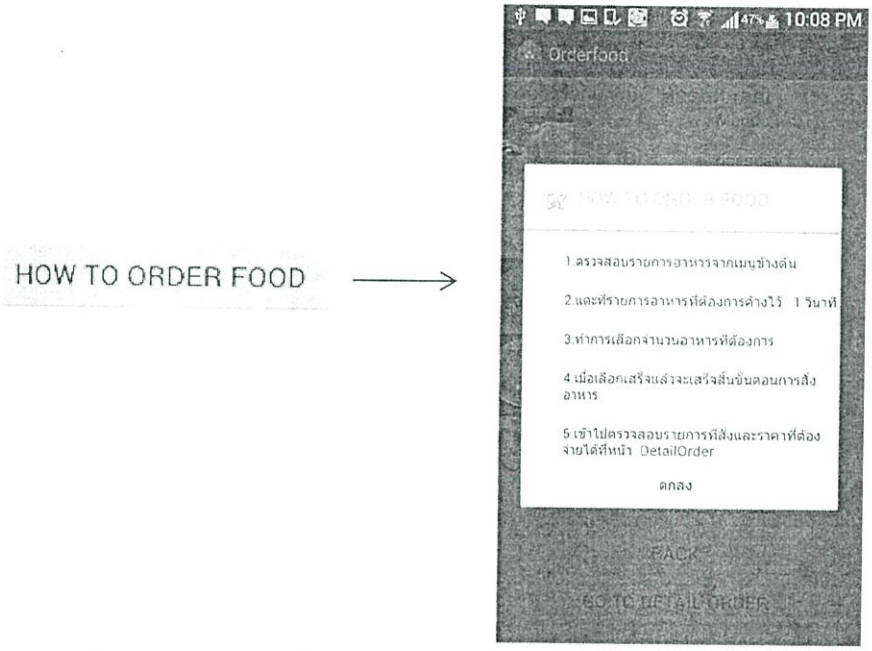
รูปที่ 4.4 ข้อมูลที่จะส่งให้ Android Client

ต่อมาเมื่อเราทำการเปิดแอพพลิเคชั่นที่ได้สร้างขึ้นเพื่อดึงข้อมูลจากฐานข้อมูล ก็ถือว่าเป็นการร้องขอการส่งข้อมูลจาก Web Service เมื่อทำการดึงค่าจาก Database ถ้าติดต่อ Server สำเร็จ และไม่มีปัญหาระหว่างการรับข้อมูลจะได้ข้อมูลที่ถูกดึงมาจากฐานข้อมูลและถูกจัดเรียงให้อยู่ในรูปแบบ Listview ดังรูป 4.5



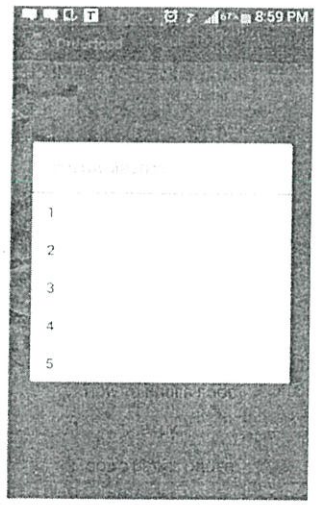
รูปที่ 4.5 ข้อมูลที่ดึงมาจาก Database แสดงผลแบบ Listview

เมื่อผู้ใช้งานต้องการอ่านวิธีการสั่งอาหารผ่านแอพพลิเคชั่น ขั้นตอน และข้อกำหนดในการกรอกฟอร์มสั่งอาหาร ให้ทำการกดที่ปุ่ม “HOW TO ORDER FOOD” เมื่อผู้ใช้งานทำการกดปุ่มจะมี Dialog Box แสดงถึงวิธีการสั่งอาหารและข้อกำหนดต่างๆ ดังรูปที่ 4.6



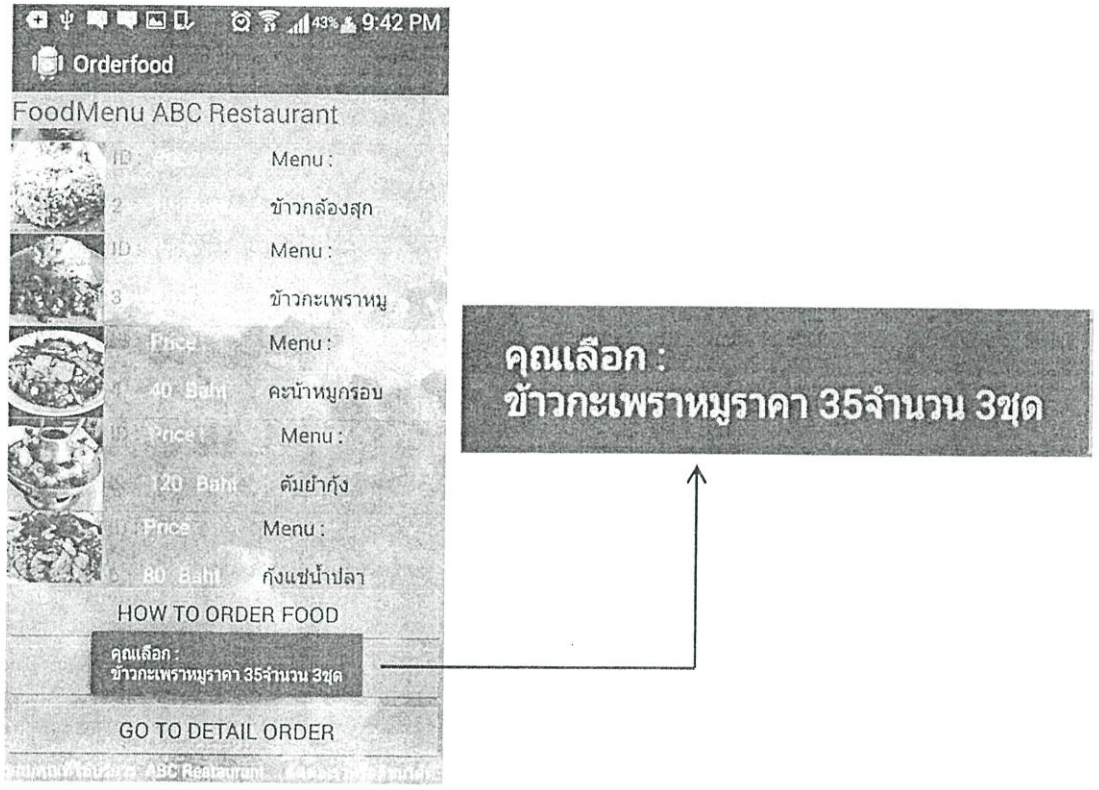
รูปที่ 4.6 การแสดงผลเมื่อกดปุ่ม HOW TO ORDER FOOD

เมื่อผู้ใช้งานต้องการสั่งอาหาร ผู้ใช้งานต้องกดค้างที่รายการอาหารที่ต้องการสั่งอาหารจะมี Dialog แสดง จำนวนรายการเราต้องการสั่ง ดังรูปที่ 4.7



รูปที่ 4.7 การแสดงผลเมื่อกดค้างที่รายการอาหารที่ต้องการ

ต่อมาเมื่อผู้ใช้ทำการเลือกจำนวนอาหารแล้วจะมี Dialog แสดงถึงรายการอาหารที่เราได้ทำการเลือกและจำนวนอาหารที่ต้องการ ดังรูปที่ 4.9



รูปที่ 4.8 แสดงรายการอาหารที่ได้ถูกเลือก

ต่อมาเมื่อทำการเลือกรายการอาหารที่ต้องการสั่งเรียบร้อยแล้วสามารถกดปุ่ม "GO TO DETAIL ORDER" เพื่อดูรายการอาหารที่ผู้ใช้งานสั่งไปทั้งหมดและ ราคาอาหารทั้งหมดพร้อมกับส่วนลดราคาอาหาร(ถ้ามี) ดังรูปที่ 4.9

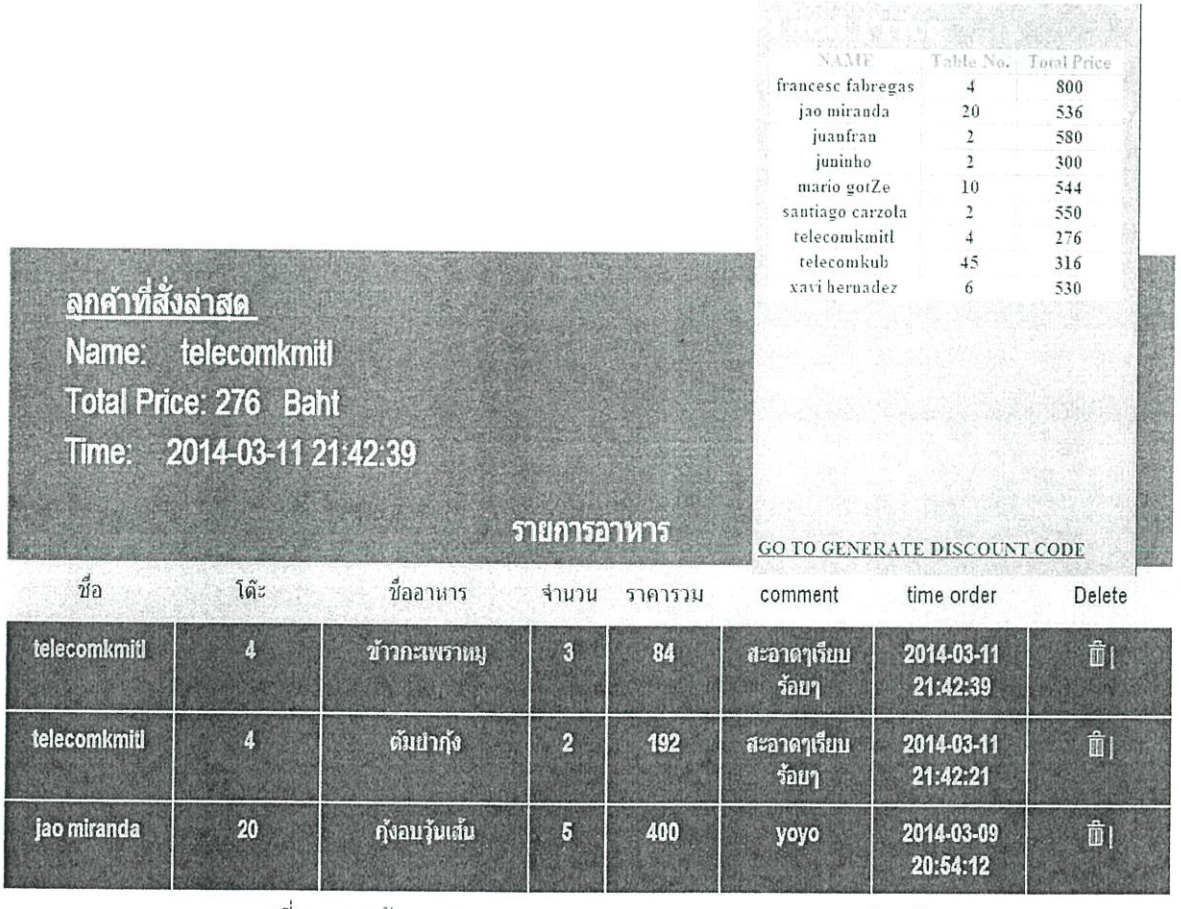
รูปที่ 4.9 แสดงรายการที่สั่งไปทั้งหมดและราคาอาหารที่ต้องจ่ายพร้อมส่วนลด

หากผู้ใช้งานใส่รหัสส่วนลดอาหารผิดหรือไม่มีรหัสส่วนลดอาหาร ผู้ใช้งานจะต้องจ่ายราคาอาหารตามจริง โดยจะไม่ได้ส่วนลดใดๆจากทางร้านดังรูปที่ 4.10

รูปที่ 4.10 แสดงรายการที่สั่งไปทั้งหมดและราคาอาหารที่ต้องจ่ายโดยไม่มีส่วนลด

4.3 ผลการทดลองส่วนของหน้าจอแสดงผลของพ่อครัวและจัดการฐานข้อมูล

ในส่วนนี้ได้ทำการนำข้อมูลที่ได้รับจากผู้ใช้งานมาแสดงผลเป็นเว็บแอพพลิเคชั่นซึ่งแสดงให้พ่อครัวได้ทราบถึงอาหารที่ต้องทำ ราคาอาหารรวมทั้งหมดของแต่ละผู้คนที่สั่งอาหารเป็นเท่าใด โดยสามารถทำการลบรายการอาหารที่พ่อครัวทำเสร็จแล้วได้ และสามารถจัดการ Generate Code หรือ Delete Code ที่ใช้สำหรับเป็นส่วนลดอาหารได้ โดยเมื่อทำการ Generate Code ข้อมูล Code จะถูกเก็บไว้ยังฐานข้อมูล และเมื่อทำการสั่งอาหารโดยใช้รหัสส่วนลดอาหาร รหัสใดถูกใช้งานแล้วระบบจะทำการลบโค้ดนั้นโดยอัตโนมัติ โดยรูปที่ 4.11 จะเป็นการแสดงถึงผลลัพธ์ที่ออกมาหลังจากการส่งข้อมูลเข้าไปยังฐานข้อมูลเรียบร้อยแล้วแสดงผลยังหน้าเว็บแอพพลิเคชั่น ส่วนรูปที่ 4.12 จะแสดงถึงหน้าการจัดการ Generate Code และ Delete Code ผ่านหน้าเว็บแอพพลิเคชั่นได้



ชื่อ	โต๊ะ	ชื่ออาหาร	จำนวน	ราคารวม	comment	time order	Delete
telecomkmitl	4	ข้าวกะเพราหมู	3	84	สะอาดๆเรียบร้อยๆ	2014-03-11 21:42:39	
telecomkmitl	4	ต้มยำกุ้ง	2	192	สะอาดๆเรียบร้อยๆ	2014-03-11 21:42:21	
jao miranda	20	กุ้งอบวุ้นเส้น	5	400	yoyo	2014-03-09 20:54:12	

รูปที่ 4.11 หน้าจอแสดงผลรายการอาหาร ราคาอาหาร ของพ่อครัว

code_gen	no	TIME
y6pi1	51	2014-02-20 13:23:58
iefi8	50	2014-02-20 13:23:11
eqop2	49	2014-02-20 13:23:08
5f34l	36	2014-02-20 03:37:28
4bozw	35	2014-02-20 03:37:28

Generate Code

BACK TO TOTAL PRICE

MANAGE DISCOUNT CODE

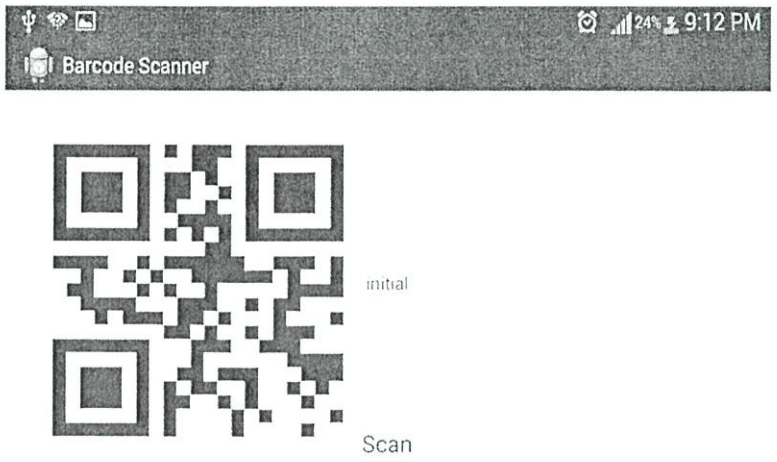
ONLINE DISCOUNT CODE

ชื่อ	TIME	Delete
y6pi1	2014-02-20 13:23:58	
iefi8	2014-02-20 13:23:11	
eqop2	2014-02-20 13:23:08	
5f34l	2014-02-20 03:37:28	
4bozw	2014-02-20 03:37:28	

รูปที่ 4.12 หน้าจอแสดงถึงหน้าการจัดการ Generate Code และ Delete Code

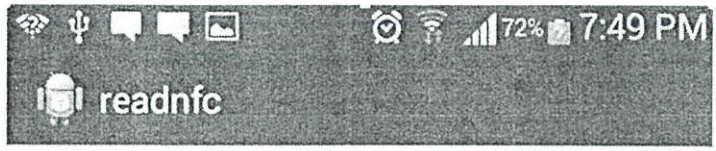
4.4 ผลการทดลองส่วนของการติดต่อสื่อสารกับ NFC และ QR Code

ในส่วนนี้ได้ทำการนำ NFC Tag และ QR Code มาเชื่อมต่อกับแอพพลิเคชั่น Order food เพื่อทำการเชื่อมต่อเครือข่ายไร้สาย (Wi-Fi) ของทางร้านอาหารซึ่งมีรหัสเชื่อมต่อที่จะต้องใช้คำสั่งเปิดจาก NFC Tag และ QR Code โดยใน NFC Tag จะใช้คำว่า password ส่วน QR Code จะใช้คำว่า initial เป็นตัวสั่งการให้ Android Client เชื่อมต่อเครือข่ายอัตโนมัติและทำการ Intent Application อัตโนมัติ



รูปที่ 4.13 หน้าจอแอพพลิเคชั่นที่ทำการอ่าน QR Code และ QR Code ที่มี Text = initial

รูปที่ 4.14 หน้าจอแอพพลิเคชั่น Order Food และการเชื่อมต่อ Wi-Fi



Hello world!

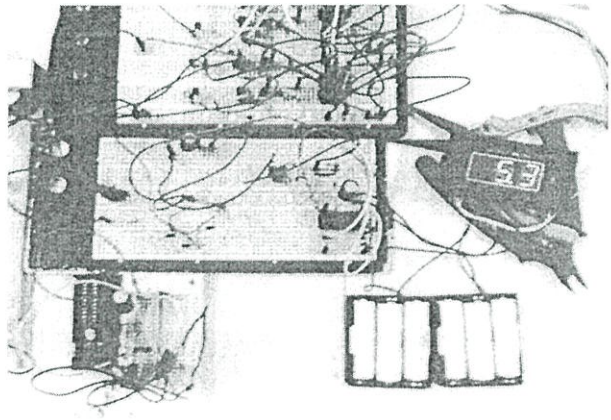
password

รูปที่ 4.15 หน้าจอแอพพลิเคชั่นที่ทำการอ่าน NFC Tag ที่มี Text = password

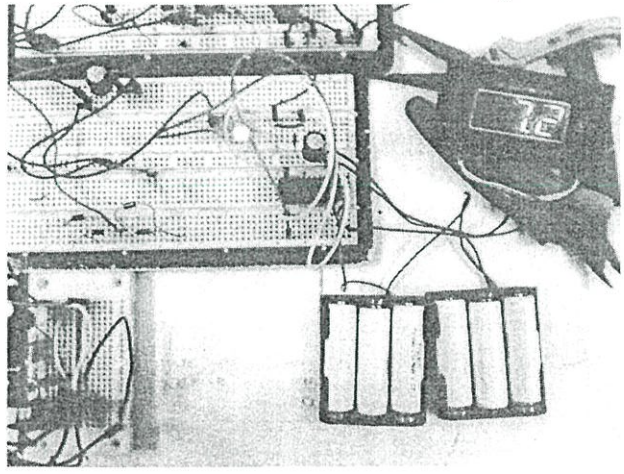
รูปที่ 4.16 หน้าจอแอพพลิเคชั่น Order Food และการเชื่อมต่อ Wi-Fi

4.5 ผลการทดสอบการชาร์จของค่า Voltage Battery

ในส่วนที่ได้มีการออกแบบวงจรการชาร์จไฟของแบตเตอรี่ในหุ่นยนต์ การทำวงจรชาร์จ Battery โดยจ่ายกระแสคงที่ด้วยไอซี 7809 ในขณะที่ไฟหมด ผลการทดลองสามารถใช้งานได้จริงโดย การทำงานของวงจรนี้คือนำไฟ 220 VAC เข้าแล้วแปลงเป็นไป 9 VDC ซึ่งเมื่อ Battery ที่ชาร์จแล้ว LED จะทำงานโดยจะสว่างซึ่งการทดลองนี้ได้ทดลองกับเครื่องออสซิลโลสโคปในห้องทดลอง โดยไฟที่ ออกมาที่ Load เราจะใช้เครื่องมิเตอร์วัดค่าซึ่งค่าที่ออกมาก็อยู่ประมาณที่ 9-11 VDC ตามรูปที่ 4.18



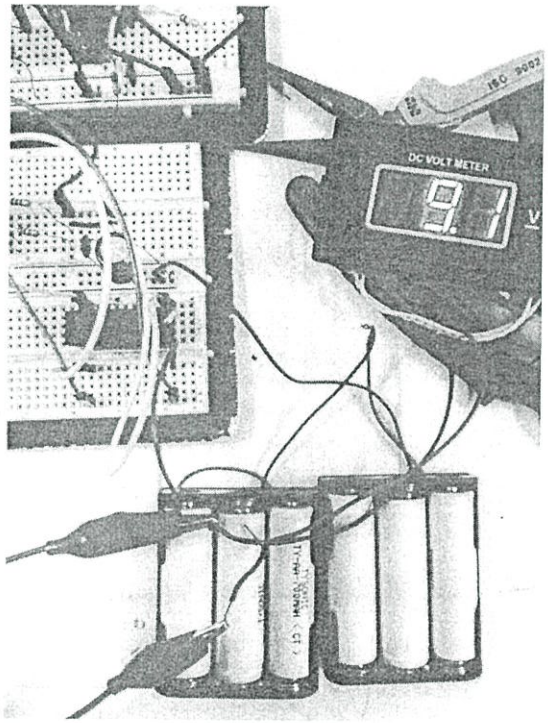
รูปที่ 4.17 แสดงก่อนการชาร์จ ถ่านมีไฟอยู่ที่ 5.3 V



รูปที่ 4.18 แสดงการชาร์จถ่านมีไฟเพิ่มเป็น 7.2 V โดยมีการสว่างของ LED

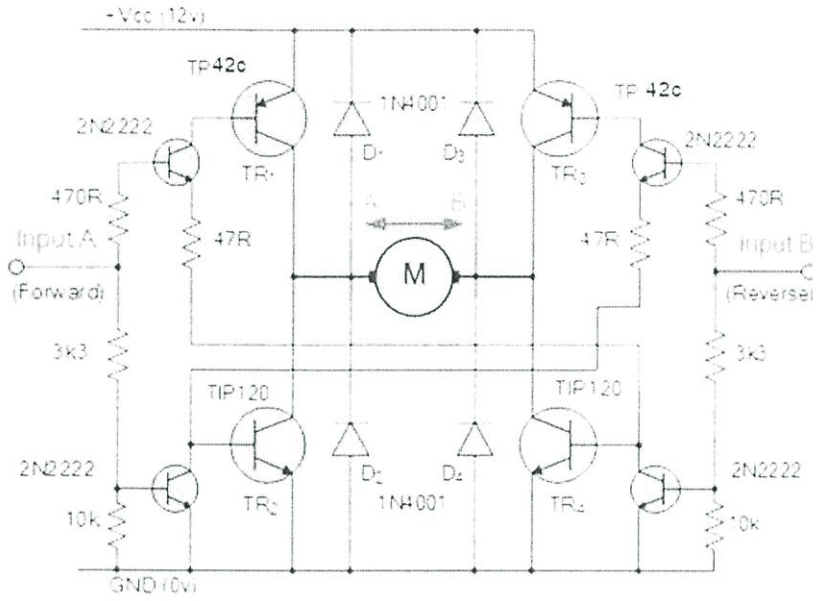
พบว่าอัตราการชาร์จไฟขึ้นกับค่า Capacitor ในวงจร โดยจะทำหน้าที่เป็นอุปกรณ์ที่ใช้ในการเก็บประจุ (Charge) และสามารถคายประจุ (Discharge)

โดยวงจรนี้มีอัตราการชาร์จประจุอยู่ที่ 55 นาที ในการชาร์ตไฟจาก 5.3 V เป็น 9 V ดังรูปที่ 4.19



รูปที่ 4.19 ผลการชาร์จไฟ ด้วยการเพิ่มของ Voltage จาก 5.3 V เป็น 9 V ใน 55 นาที

4.6 ผลการทดสอบวงจร Motor Controller Board



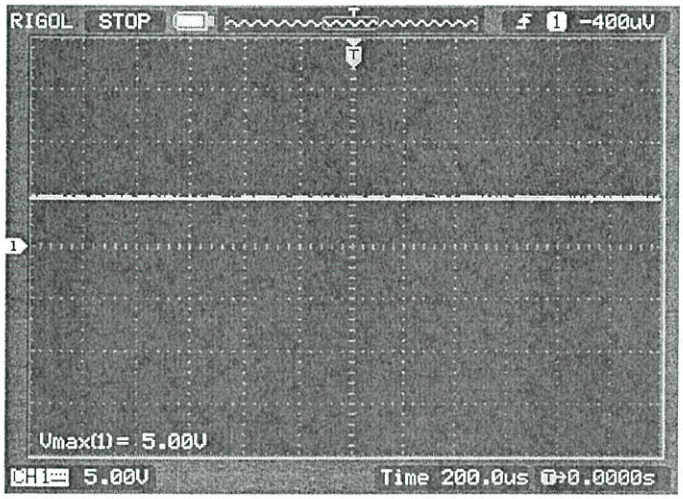
รูปที่ 4.20 แสดงวงจร Motor Controller Board

หน้าที่: ควบคุมการหมุนของมอเตอร์ ทั้งล้อซ้ายและขวา โดยจะมีวงจร H-bridge เป็นตัวขับมอเตอร์ หลักการของวงจรนั้นจะประกอบไปด้วยสวิทช์ 4 ตัว นั่นก็คือ S1, S2, S3 และ S4 มีกระแสไฟฟ้าไหลเข้าสู่มอเตอร์ และเมื่อเราทำการเปิด (On) สวิตช์ S1 และ S3 พร้อมกัน จะเป็นการเชื่อมวงจรทำให้มีกระแสไฟฟ้าไหลผ่านมอเตอร์ จากขั้วบวกของมอเตอร์ไปยังขั้วลบของมอเตอร์จึงทำให้มอเตอร์สามารถหมุนได้ในทิศทางฟอร์เวิร์ด (Forward)

ดังนั้น หลักการทำงานคือ ต้องมีแรงดันไฟขนาด 5 V มาไบอัสที่ Input (Input A) เพื่อนำค่าแรงดันที่ได้มาไบอัส ทรานซิสเตอร์ 2N222 และ ทรานซิสเตอร์ TIP42C ตามลำดับ เพื่อทำให้มีกระแสไหลผ่านมอเตอร์ และไหลผ่านไปที่ ทรานซิสเตอร์ TIP120 และ ทรานซิสเตอร์ 2N222 และลง Ground ตามลำดับ

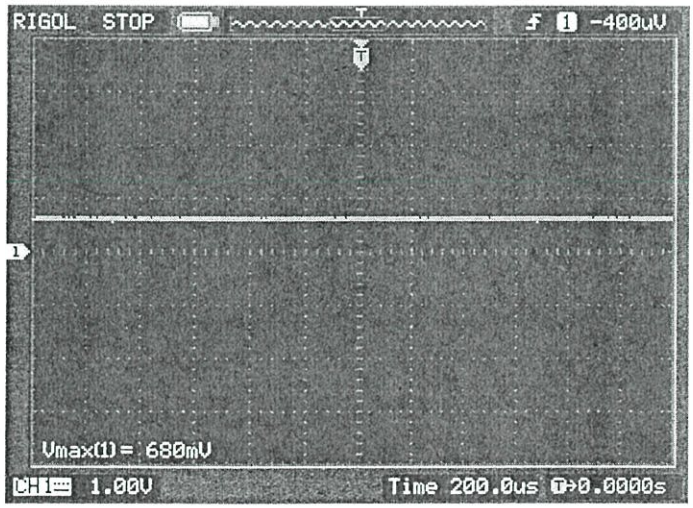
เพราะฉะนั้น แรงดัน 5 V จะได้รับมาจากวงจร Microcontroller (Arduino ATTMEGA88) จะสามารถทำให้มอเตอร์หมุน โดยใช้หลักการไบอัสของทรานซิสเตอร์

เมื่อมีการรับลอจิก 1 มาจากวงจร Microcontroller (Arduino ATTMEGA88) เป็นแรงดันไฟขนาด 5 V เข้าที่อินพุต (Input A) ของวงจร Motor Controller Board ที่ขาเบสของทรานซิสเตอร์ 2N2222 ดังรูปที่ 4.21



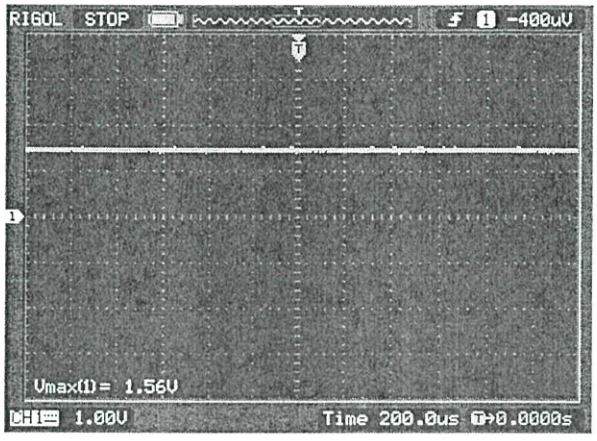
รูปที่ 4.21 แสดงลอจิก 1 (+5V) ที่รับมาจาก Arduino ATTMEGA88

โดยจะวัดค่า V_{BE} ที่ทรานซิสเตอร์ 2N2222 ได้เท่ากับ 0.68 V ดังรูปที่ 4.22



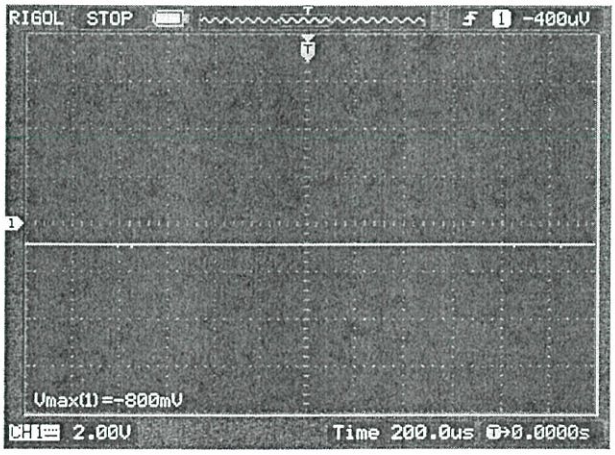
รูปที่ 4.22 แสดง ค่า V_{BE} ที่ทรานซิสเตอร์ 2N2222

จะเกิดกระแส Ie ที่ขาอีมิตเตอร์ของ ทรานซิสเตอร์ 2N2222 ไปยังขาเบสของทรานซิสเตอร์TIP120 และทำการไบอัส โดยวัดค่าของ V_{BE} ของ ทรานซิสเตอร์ TIP120 = 1.56 V ดังรูปที่ 4.23



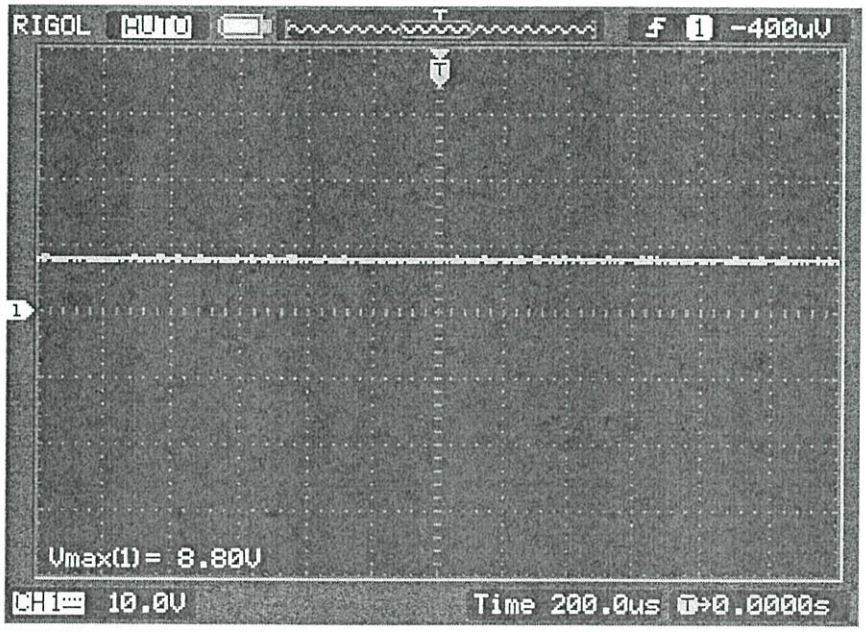
รูปที่ 4.23 แสดง ค่า V_{BE} ที่ทรานซิสเตอร์ TIP120

กระแสไฟไหลผ่านคอลเลคเตอร์ ของทรานซิสเตอร์ 2N2222 ที่ขาเบสของทรานซิสเตอร์ TIP42C เกิดการไบอัสกลับ V_{BE} =- 0.8 V ดังรูปที่ 4.24



รูปที่ 4.24 แสดง ค่าการไบอัสกลับที่ V_{BE} ของทรานซิสเตอร์ TIP42C

เมื่อทรานซิสเตอร์ TIP42C และ ทรานซิสเตอร์ TIP120 ทำงาน จะเกิดกระแสไหลผ่านมอเตอร์ โดยทำการวัดแรงดันตกคร่อมได้เท่ากับ 8.80 V โดยนำแรงดันนี้ไปขับมอเตอร์ต่อไป ดังรูปที่ 4.25

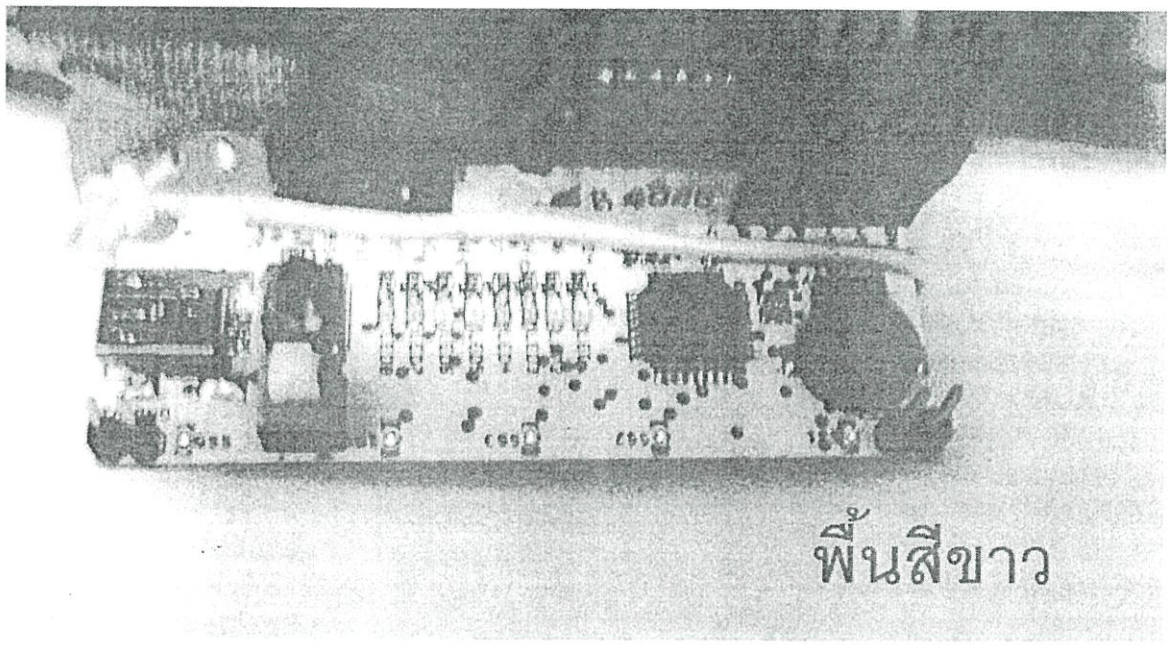


รูปที่ 4.25 แสดงแรงดันตกคร่อมมอเตอร์ ได้เท่ากับ 8.80 V

4.7 ผลการทดสอบการรับค่า ความเข้มสีขาวและสีดำ ของเซ็นเซอร์อินฟราเรด

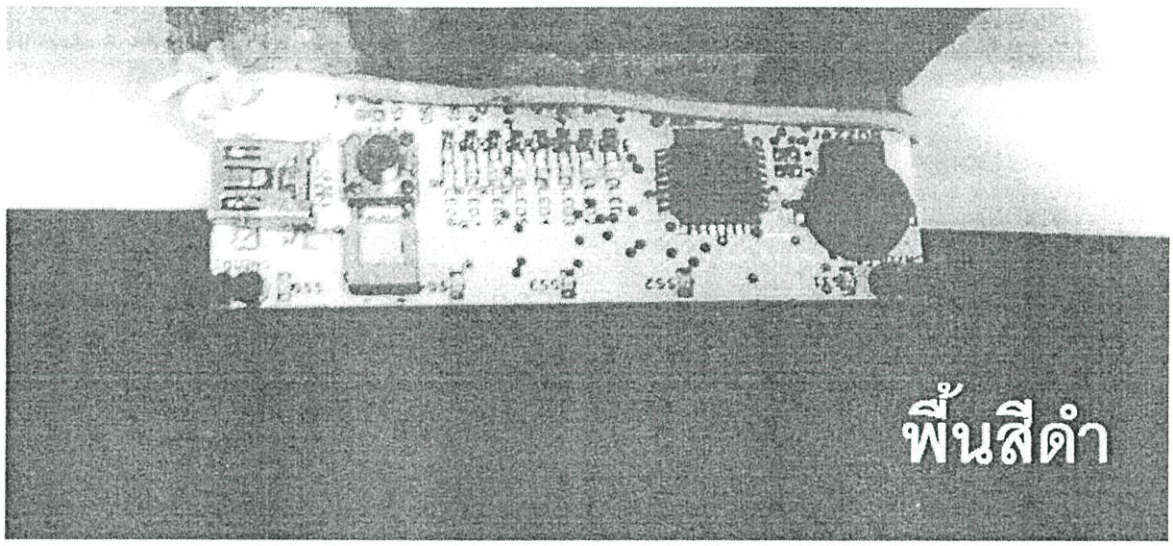
Arduino ประกอบไปด้วยเซ็นเซอร์อินฟราเรดทั้งหมด 5 ตัว โดย Arduino ATTMEGA88 จะทำหน้าที่ในการรับอินพุตเป็นค่าลอจิก จากเซ็นเซอร์อินฟราเรด หากเซ็นเซอร์ตัวใดจับสีดำได้ จะส่ง Output เป็นลอจิก 1 หากเซ็นเซอร์ตัวใดจับสีขาวได้ จะส่ง Output เป็นลอจิก 0 เพื่อนำ Output ที่ได้ไปประมวลผลในการเลือกเส้นทางของหุ่นยนต์ต่อไป

4.7.1 ผลการทดสอบการจับความเข้มสี สีขาว จะทำให้ LED ที่เซ็นเซอร์ส่องสว่าง พร้อมมีการส่งค่าลอจิก 0 ไปประมวลผลการทำงานยัง Arduino



รูปที่ 4.26 แสดงการติดสว่างของ LED เมื่อเซ็นเซอร์ทั้ง 5 ตัว มีการจับความเข้มของสีขาวได้

4.7.2 ผลการทดสอบการจับความเข้มสี สีดำ จะทำให้ LED ที่เซ็นเซอร์ดับ พร้อมมีการส่งค่าลอจิก 1 ไปประมวลผลการทำงานยัง Arduino



รูปที่ 4.27 แสดงการดับของ LED เมื่อเซ็นเซอร์ทั้ง 5 ตัว มีการจับความเข้มของสีดำได้

4.8 ผลการทดสอบการปรับค่าความเร็วของมอเตอร์

การปรับความเร็วมอเตอร์มีส่วนสำคัญมากในการเดินตามเส้นทางของหุ่นยนต์ หลักการปรับความเร็วของหุ่นยนต์เดินตามเส้นนั้นสามารถดูได้ตามตารางข้างล่าง

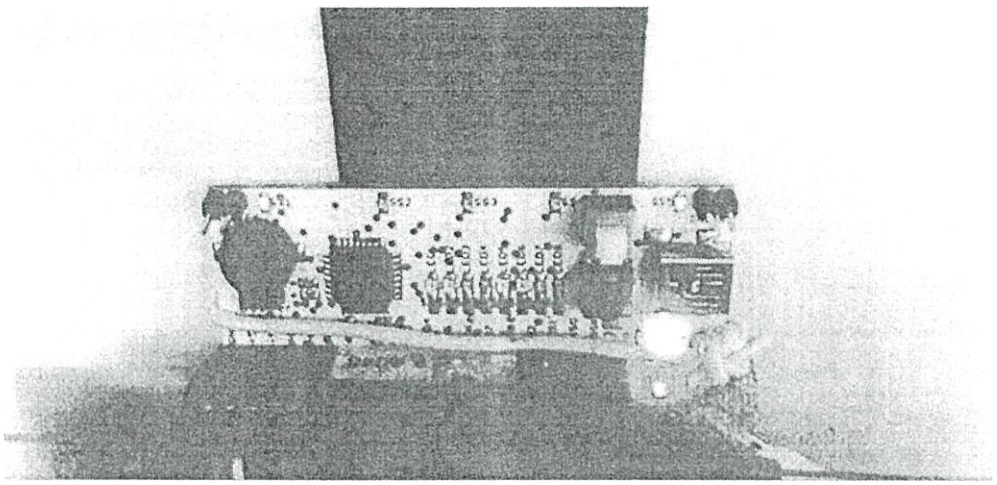
เลข 1 คือ มีการส่งลอจิก 1 ของเซ็นเซอร์ตัวนั้น นั่นคือการจับโดนความเข้มสีขาว

เลข 0 คือ มีการส่งลอจิก 0 ของเซ็นเซอร์ตัวนั้น นั่นคือการจับโดนความเข้มสีดำ

ตารางที่ 4.1 แสดงสถานะการทำงานของหุ่นยนต์ เมื่อมีการจับค่าความเข้มของสีขาว และสีดำได้

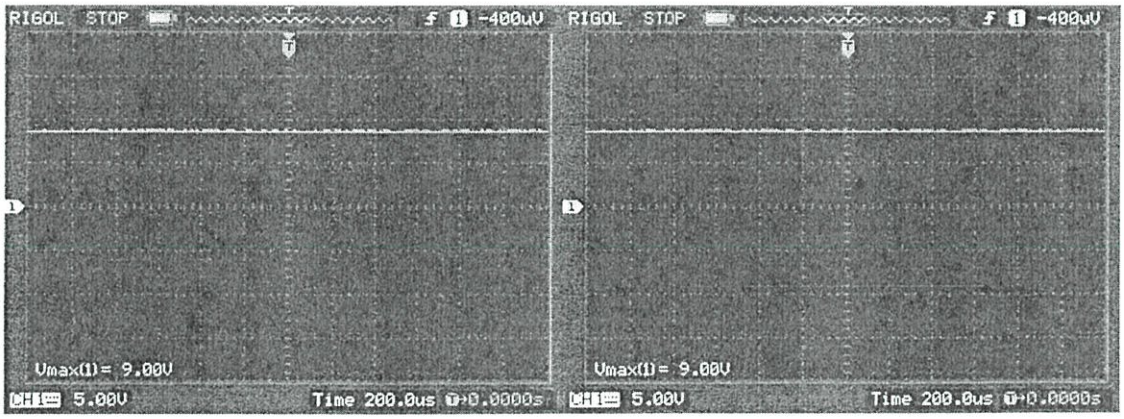
ตำแหน่งเซ็นเซอร์					การทำงาน
1	2	3	4	5	
1	0	0	0	1	หุ่นยนต์เดินตรงไปข้างหน้า
0	0	0	1	1	ล้อซ้ายหยุด ล้อขวาหมุน (เลี้ยวซ้าย) เพื่อปรับระดับให้อยู่ในเส้นทางเหมือนเดิม
0	0	1	1	1	ล้อซ้ายหยุด ล้อขวาหมุน (เลี้ยวซ้าย) เพื่อปรับระดับให้อยู่ในเส้นทางเหมือนเดิม
1	1	0	0	0	ล้อซ้ายหมุน ล้อขวาหยุด (เลี้ยวขวา) เพื่อปรับระดับให้อยู่ในเส้นทางเหมือนเดิม
1	1	1	0	0	ล้อซ้ายหมุน ล้อขวาหยุด (เลี้ยวขวา) เพื่อปรับระดับให้อยู่ในเส้นทางเหมือนเดิม
0	0	0	0	0	หุ่นยนต์หยุด เพื่อพิจารณาการทำงานในลำดับถัดไป

4.8.1 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 10001



รูปที่ 4.28 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 10001

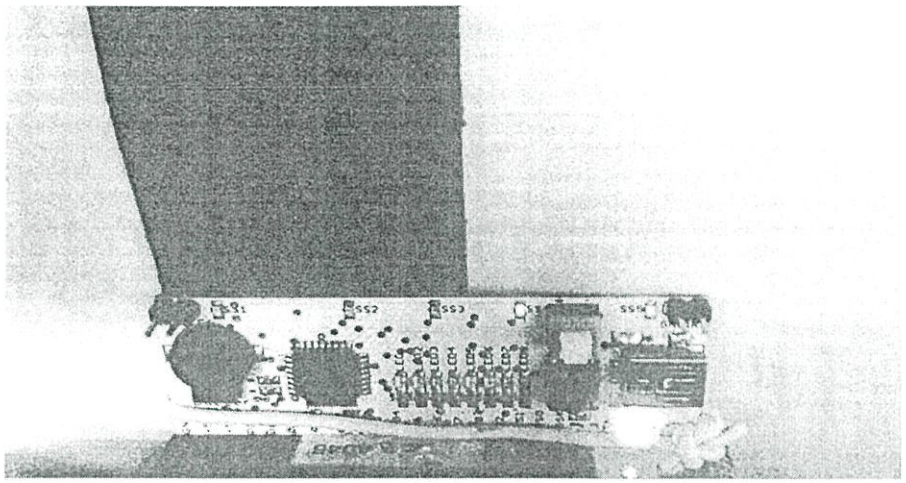
สถานะการทำงานคือ หุ่นยนต์เดินตรงไปข้างหน้า



ก ข

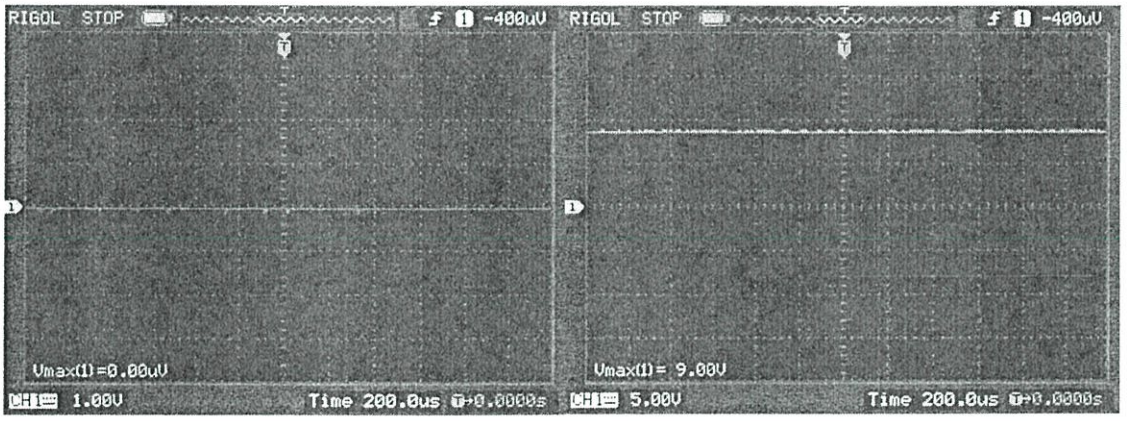
รูปที่ 4.29.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 9 V ทำให้ล้อหมุน
รูปที่ 4.29.ข แสดงมอเตอร์ขวามี ค่า Voltage = 9 V ทำให้ล้อขวาหมุน

4.8.2 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 00011



รูปที่ 4.30 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 00011

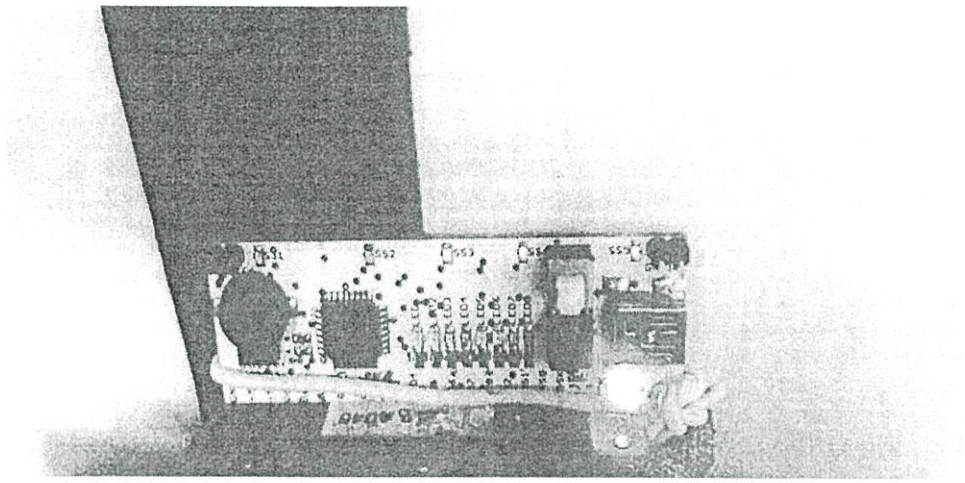
สถานะการทำงานคือ ล้อซ้ายหยุด ล้อขวาหมุน (เลี้ยวซ้าย) เพื่อปรับระดับให้อยู่ในเส้นทางเหมือนเดิม



ก ข

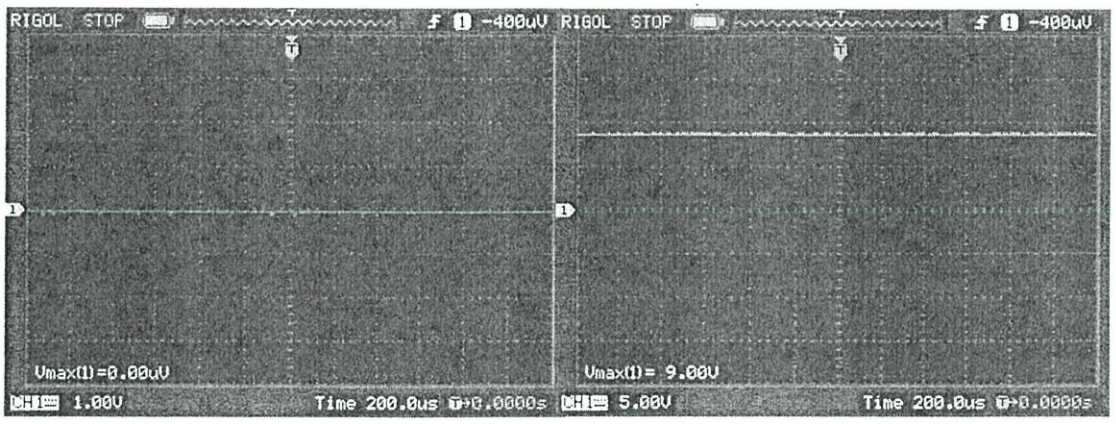
รูปที่ 4.31.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 0 V ทำให้ล้อซ้ายหยุดหมุน
รูปที่ 4.31.ข แสดงมอเตอร์ขวามี ค่า Voltage = 9 V ทำให้ล้อขวาหมุน

4.8.3 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 00111



รูปที่ 4.32 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 00111

สถานะการทำงานคือ คือล้อซ้ายหยุด ล้อขวาหมุน (เลี้ยวซ้าย) เพื่อปรับระดับให้อยู่ในเส้นทางเหมือนเดิม

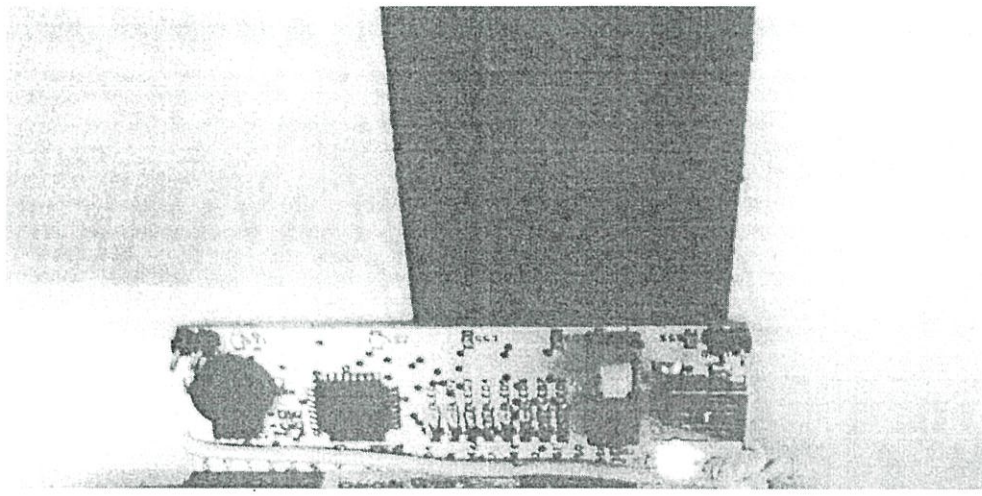


ก ข

รูปที่ 4.33.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 0 V ทำให้ล้อซ้ายหยุดหมุน

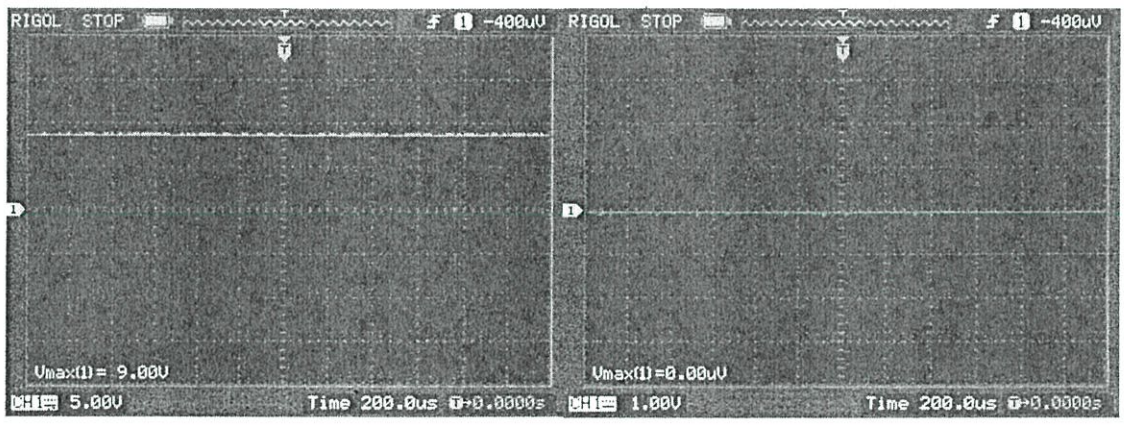
รูปที่ 4.33.ข แสดงมอเตอร์ขวามี ค่า Voltage = 9 V ทำให้ล้อขวาหมุน

4.8.4 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 11000



รูปที่ 4.34 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 11000

สถานะการทำงานคือ คือล้อซ้ายหมุน ล้อขวาหยุด (เลี้ยวขวา) เพื่อปรับระดับให้อยู่ในเส้นทางเหมือนเดิม

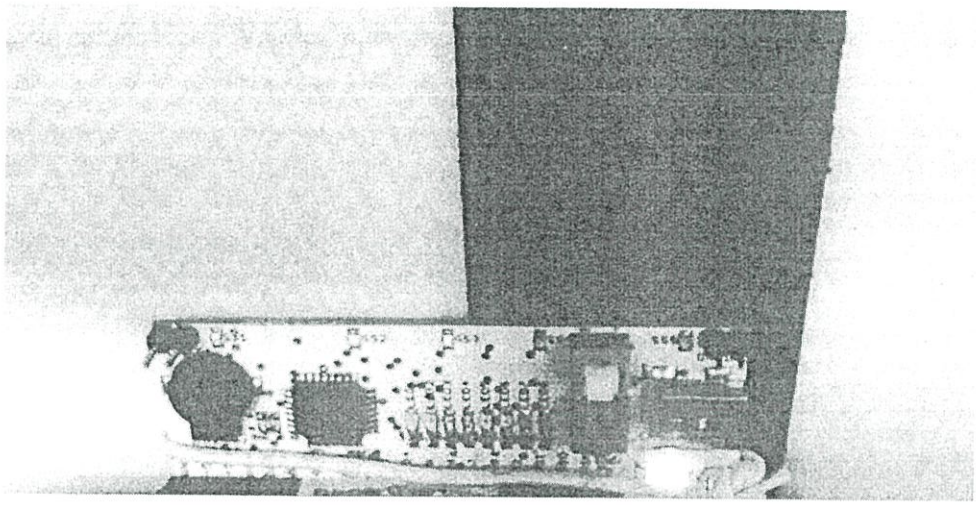


ก ข

รูปที่ 4.35.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 9 V ทำให้ล้อซ้ายหมุน

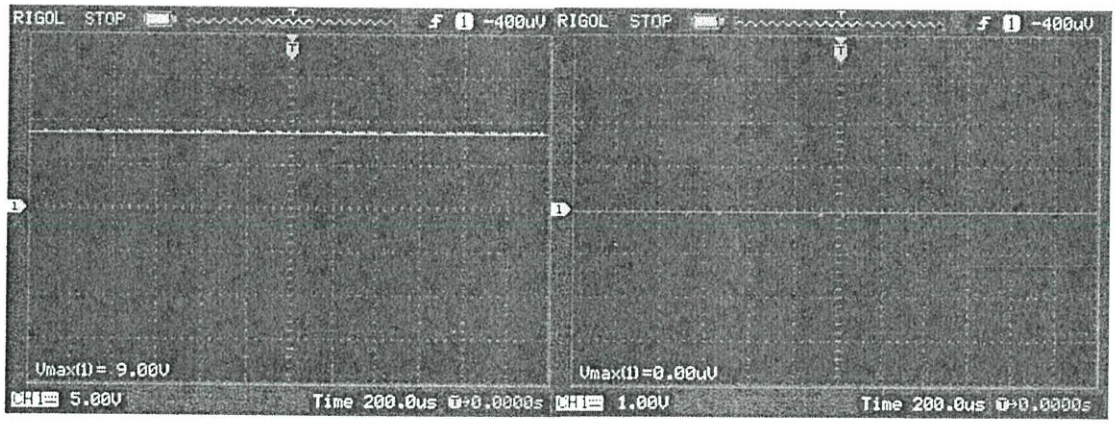
รูปที่ 4.35.ข แสดงมอเตอร์ขวามี ค่า Voltage = 0 V ทำให้ล้อขวาหยุดหมุน

4.8.5 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 11100



รูปที่ 4.36 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 11100

สถานะการทำงานคือ ล้อซ้ายหมุน ล้อขวาหยุด (เลี้ยวขวา) เพื่อปรับระดับให้อยู่ในเส้นทางเหมือนเดิม

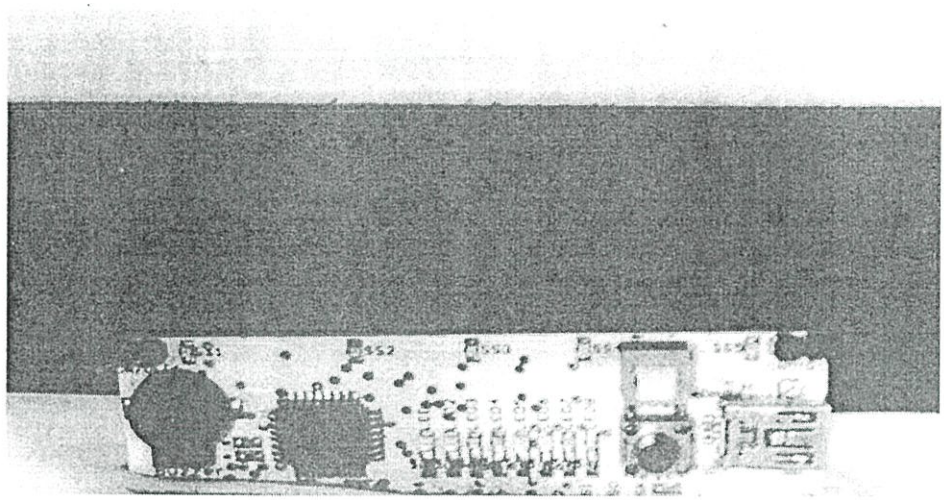


ก ข

รูปที่ 4.37.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 9 V ทำให้ล้อซ้ายหมุน

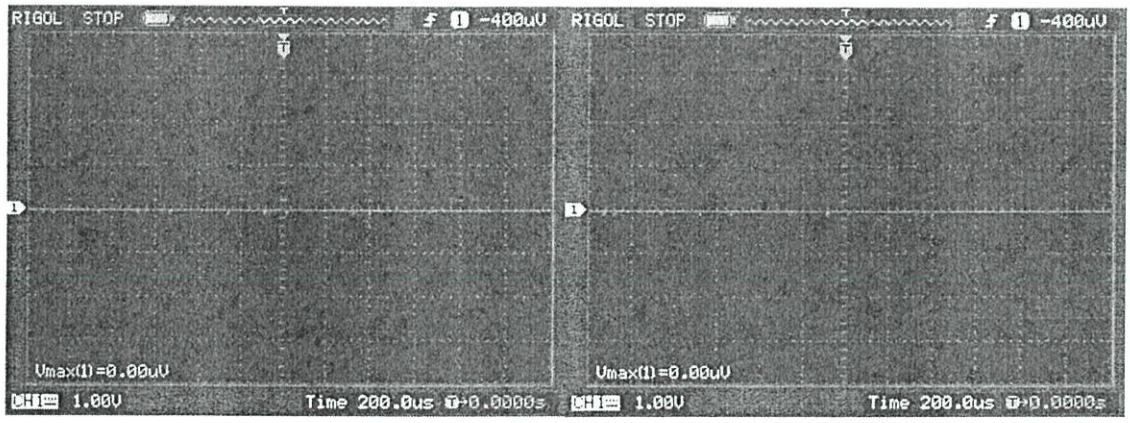
รูปที่ 4.37.ข แสดงมอเตอร์ขวามี ค่า Voltage = 0 V ทำให้ล้อขวาหยุดหมุน

4.8.6 ผลการทดสอบเมื่อเซ็นเซอร์มีการรับค่าของเซ็นเซอร์เป็น 00000



รูปที่ 4.38 แสดงการทำงานของ LED เมื่อมีการรับค่าเซ็นเซอร์เป็น 00000

สถานะการทำงานคือ หุ่นยนต์หยุด เพื่อพิจารณาการทำงานในลำดับถัดไป



ก ข

รูปที่ 4.39.ก แสดงมอเตอร์ซ้ายมี ค่า Voltage = 0 V ทำให้ล้อซ้ายหยุดหมุน
รูปที่ 4.39.ข แสดงมอเตอร์ขวามี ค่า Voltage = 0 V ทำให้ล้อขวาหยุดหมุน

บทที่ 5
สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

โครงงานนี้จะเป็นส่วนที่จะช่วยแบ่งเบาภาระของร้านอาหาร โดยจะครอบคลุมการบริการและ เป็นเทคโนโลยีในการอำนวยความสะดวกด้วยเช่นกัน ซึ่งจะเป็นอีกหนึ่งทางเลือกในการลดต้นทุนในการผลิตของร้านอาหาร โดยโครงงานนี้จะมีการสร้างระบบการสั่งอาหารผ่านแอพพลิเคชั่นบนโทรศัพท์มือถือระบบปฏิบัติการณ์แอนดรอยด์ และยังมีเทคโนโลยี Near Field Communication (NFC) และ QR-Code อีกทั้งยังมีระบบส่งอาหารอัตโนมัติ โดยจะมีหุ่นยนต์อัตโนมัติควบคุมด้วยไมโครคอนโทรลเลอร์ AVR โครงงานนี้มีขั้นตอนและส่วนการทางานที่สำคัญหลาย ส่วนดังนี้

ส่วนของแอพพลิเคชั่น Readnfc และ Barcodescanner ทำหน้าที่อ่านแท็ก NFC และ QR-Code เพื่อทำการเปลี่ยนแอพพลิเคชั่นอัตโนมัติและเชื่อมต่อ Wireless Network ของทางร้านอาหารได้ โดยต้องทำการอ่านแท็กจึงจะสามารถเชื่อมต่อได้เนื่องจากมีรหัสผ่าน

ส่วนของเซิร์ฟเวอร์ มีหน้าที่เก็บรายการอาหารสำหรับร้านอาหาร โดยการเพิ่มหรือลดรายการอาหาร เพิ่มหรือลดราคาอาหารในร้าน มีการจัดการกับรหัสที่ใช้เป็นส่วนลดราคาอาหารได้และทำการแปลงข้อมูลให้เป็นข้อมูลแบบ Json

ส่วนของการแสดงผลแบบ Listview เป็นส่วนของแอพพลิเคชั่นที่ได้ทำการสร้างขึ้น แอพพลิเคชั่นนี้จะทำหน้าที่ในการแปลงข้อมูล Json ให้เป็น String พร้อมกับ Image และทำการเลือกข้อมูลในตาราง Database มาแสดงผลแบบ Listview บนหน้าจอโทรศัพท์มือถือ ระบบปฏิบัติการแอนดรอยด์

ส่วนของแอพพลิเคชั่น Order Food ผู้ใช้งานจะทำการกรอกข้อมูลตามฟอร์ม หลังจากนั้นลูกค้าจะทำการเลือกรายการอาหารโดยเลือกแตะค้างไว้ที่รายการอาหารที่ต้องการสั่งจากนั้นเลือกจำนวนที่ต้องการจาก Dialog Box เมื่อทำการเลือกรายการอาหารแล้วจะมีส่วนการคำนวณราคาอาหารและส่วนลดอาหารในหน้า Detail Food

ส่วนวงจรการชาร์จไฟของหุ่นยนต์ ซึ่งสามารถชาร์จ Voltage ของวงจรและยังเป็นวงจรที่เป็นแหล่งกำเนิดไฟแก่หุ่นยนต์ในทุกๆส่วน

ส่วนของวงจรควบคุมมอเตอร์ ซึ่งเป็นวงจรควบคุมการหมุนของมอเตอร์ โดยมอเตอร์ซ้ายและมอเตอร์ขวาจะเป็นอิสระต่อกัน ซึ่งในวงจรนี้ จะได้รับ Input มาจากวงจรไมโครคอนโทลเลอร์ เพื่อนำมาขับมอเตอร์ ทั้งล้อซ้าย และล้อขวา ตามค่าลอจิกที่ได้รับ

ส่วนของวงจรไมโครคอนโทลเลอร์ (Arduino) ประกอบไปด้วยเซ็นเซอร์อินฟราเรดทั้งหมด 5 ตัว โดย Arduino ATTMEGA88 จะทำหน้าที่ในการรับอินพุตเป็นค่าลอจิกจากเซ็นเซอร์อินฟราเรด เพื่อนำ Input ที่ได้ไปประมวลผลในการเลือกเส้นทางของหุ่นยนต์ต่อไป

5.2 ปัญหาและแนวทางแก้ไข

- การเปิดใช้งานแอพพลิเคชั่นนั้นจำเป็นต้องมีการเชื่อมต่อเครือข่ายภายในร้านซึ่งเข้าใช้งานโดยผ่าน Tag NFC หรือ QR Code โดย Smart Phone ที่จะสั่งอาหารต้องสามารถอ่าน Tag NFC หรือ Scan QR Code ได้ จึงจะสามารถสั่งอาหารได้

- การสร้างแอพพลิเคชั่นนั้นรองรับเพียงโทรศัพท์มือถือบางรุ่นเท่านั้น นั่นคือ Android รุ่น Gingerbread (2.3) ขึ้นไปเท่านั้น ดังนั้นควรตรวจสอบรุ่นของโทรศัพท์มือถือก่อนการติดตั้ง หากโทรศัพท์มือถือเป็นรุ่นต่ำกว่า Gingerbread (2.3) เราสามารถทาการอัพเกรดรุ่นของโทรศัพท์มือถือให้เป็นรุ่น Gingerbread (2.3) ขึ้นไป ได้ตามศูนย์โทรศัพท์มือถือทั่วไป

- ในส่วนของวงจรชาร์จไฟ บางครั้งไม่สามารถชาร์จจนเต็มได้อันเนื่องมาจากประสิทธิภาพของ Battery ชาร์จที่ใช้ ควรเปลี่ยนเป็น Battery ที่มีประสิทธิภาพและกาลังสูงเพื่อประสิทธิภาพการทางานที่ดี

- การทำงานของวงจรเนื่องจากมีวงจรจำนวนมาก และมีมอเตอร์ที่มีกำลัง และการทดรอบที่สูง จึงส่งผลทาให้ ถ่านมีการลดลงของค่า Voltage อย่างรวดเร็ว ควรเปลี่ยนเป็น Battery ชาร์จที่มีประสิทธิภาพและกำลังที่สูง เพื่อประสิทธิภาพการทำงานที่ดี

- การรับค่าของเซ็นเซอร์อินฟราเรด ของวงจร Microcontroller นั้นอาจมีความผิดพลาดได้จาก การตั้งค่าความเข้มแสง จากในส่วนของโปรแกรม หากจะนำไปใช้ในสถานที่ที่มีความสว่าง ที่แตกต่างจากที่ทดลองมากจนเกินไป ควรทำการปรับค่าให้ถูกต้องก่อนการใช้งาน เพื่อป้องกันความผิดพลาดในการตรวจจับความเข้มสี ของเซ็นเซอร์อินฟราเรดที่จะเกิดขึ้น

- การรับค่าของเซ็นเซอร์อินฟราเรด ของวงจร Microcontroller ขึ้นอยู่กับ แบตเตอรี่ที่เหลืออยู่ หากแบตเตอรี่น้อยเกินค่าไฟเลี้ยง ของวงจร Microcontroller ที่ประมาณ 5 V อาจทำให้วงจร Microcontroller และเซ็นเซอร์อินฟราเรด อาจเกิดความผิดพลาดได้ ดังนั้นควรเช็คแบตเตอรี่ทุกครั้งก่อนการใช้งานว่าจะสามารถใช้ได้นานเพียงใด

5.3 ข้อเสนอแนะ

แอพพลิเคชั่นบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ยังมีการประมวลผลที่ช้าเวลาทำการเชื่อมต่อกับเซิร์ฟเวอร์ภายนอก ทำให้บางเวลาขณะทำการส่งข้อมูลหรือรับข้อมูลจากเซิร์ฟเวอร์แอพพลิเคชั่นอาจเกิด Error ได้ ดังนั้นผู้พัฒนา Android ควรแก้ไขปัญหาการประมวลผลของ Android ให้รวดเร็วขึ้น

หุ่นยนต์สามรถเลือกโต๊ะได้เพียง 3 โต๊ะ เนื่องจากถูกจำกัดด้วยหน่วยความจำ และ พอร์ตการใช้งานของวงจร Microcontroller หากต้องการใช้ในระบบจริง ในร้านอาหารที่มีโต๊ะจำนวนมากๆ ก็ให้เลือกใช้ Microcontroller เบอร์อื่นที่ไม่ถูกจำกัดด้วยจำนวนพอร์ตการใช้งานและหน่วยความจำ อีกทั้งหุ่นยนต์ยังเป็นขนาดทดสอบ ที่ยังไม่สามารถรับน้ำหนักได้มากๆ หากจะนำไปใช้จริง ควรใช้มอเตอร์ที่มีคุณภาพและกำลังสูงๆ เพื่อที่จะสามารถรับน้ำหนักอาหารให้เพิ่มมากขึ้น

บรรณานุกรม

[1] "สถาปัตยกรรมของแอนดรอยด์."
http://kadroidz.blogspot.com/2012/03/android-architecture.html
[2] "ลาดับชั้น Android app framework."
http://kadroidz.blogspot.com/2012/03/android-architecture.html
[3] "ลาดับชั้น Libraries และ Android runtime."
http://kadroidz.blogspot.com/2012/03/android-architecture.html
[4] "แสดงลาดับชั้น Libraries และ Android runtime."
http://kadroidz.blogspot.com/2012/03/android-architecture.html
[5] " Class ของภาษาจาวา."
http://kadroidz.blogspot.com/2012/03/android-architecture.html
[6] "การเปลี่ยนสัญญาณไฟสลับเป็นไฟกระแสตรง."
http://www.neutron.rmutphysics.com/physicsboard/forum/index.php?topic=689.0
[7] "วงจรเรียงกระแสแบบเต็มคลื่นชนิดบริดจ์."
http://www.neutron.rmutphysics.com/physicsboard/forum/index.php?topic=689.0
[8] "วงจรเรียงกระแสแบบเต็มคลื่น ชนิดบริดจ์."
http://www.neutron.rmutphysics.com/physicsboard/forum/index.php?topic=689.0
[9] "การทางานของวงจรเรียงกระแสเต็มคลื่นแบบบริดจ์เมื่ออินพุตเป็นซีกบวก."
http://www.neutron.rmutphysics.com/physicsboard/forum/index.php?topic=689.0
[10] "วงจรแบ่งแรงดันไฟฟ้าแบบไม่มีภาระไฟฟ้า."
http://www.neutron.rmutphysics.com/physicsboard/forum/index.php?topic=689.0
[11] "แสดงH-Bridge Switching."
http://www.neutron.rmutphysics.com/physicsboard/forum/index.php?topic=689.0
[12] "แสดงวงจร Voltage Regulators."
http://www.neutron.rmutphysics.com/physicsboard/forum/index.php?topic=689.0
[13] "แสดงการพิจารณา load regulation."
http://www.neutron.rmutphysics.com/physicsboard/forum/index.php?topic=689.0
[14] "วงจรภายในของมอเตอร์กระแสตรง."
http://npt.ac.th/wp-content/uploads/2012/08/page444.html
[15] "อินพุตและเอาต์พุตของโมเดลทางคณิตศาสตร์ของมอเตอร์."

http://npt.ac.th/wp-content/uploads/2012/08/page444.html
[16] "โมเดลของดีซีมอเตอร์แบบฟิลด์แยกกระตุ้น."
http://npt.ac.th/wp-content/uploads/2012/08/page444.html
[17] "ถึงแรงบิดต่างๆที่เกิดขึ้นต่อโหลดของมอเตอร์." http://npt.ac.th/wp-content/uploads/2012/08/page444.html
[18] "บล็อกไดอะแกรมของดีซีมอเตอร์โมเดล."
http://npt.ac.th/wp-content/uploads/2012/08/page444.html
[19]"สัญญาณ PWM ซึ่งแสดงค่า duty cycles ที่ต่างๆกัน."
http://wara.com/article-794.html
[20]กอบเกียรติ สระอุบล. พัฒนา App Android. พิมพ์ครั้งที่ 1. กรุงเทพฯ : โรงพิมพ์มีเดีย เนทเวิร์ค, 2556.
[21]ผศ.ดร.จีรสุดา โกษียาภรณ์. *วิศวกรรมอิเล็กทรอนิกส์*. พิมพ์ครั้งที่ 1. กรุงเทพฯ : โรงพิมพ์ห้างหุ้นส่วนจากัด วี.เจ. พริ้นติ้ง, 2551.
[22]กิตติ ภักดีวัฒนะกุล. *คัมภีร์ PHP*. พิมพ์ครั้งที่ 6. กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์ จากัด, 2547

ภาคผนวก

Features

- High performance, low power Atmel® AVR® 8-bit microcontroller
- Advanced RISC architecture
  - 131 powerful instructions – most single clock cycle execution
  - 32 x 8 general purpose working registers
  - Fully static operation
  - Up to 20 MIPS throughput at 20MHz
  - On-chip 2-cycle multiplier
- High endurance non-volatile memory segments
  - 4/8/16 Kbytes of in-system self-programmable flash program memory
  - 256/512/512 bytes EEPROM
  - 512/1K/1Kbytes internal SRAM
  - Write/erase cyles: 10,000 flash/100,000 EEPROM
  - Data retention: 20 years at 85°C/100 years at 25°C(1)
  - Optional boot code section with independent lock bits
    - In-system programming by on-chip boot program
    - True read-while-write operation
  - Programming lock for software security
- QTouch® library support
  - Capacitive touch buttons, sliders and wheels
  - QTouch and QMatrix acquisition
  - Up to 64 sense channels
- Peripheral features
  - Two 8-bit timer/counters with separate prescaler and compare mode
  - One 16-bit timer/counter with separate prescaler, compare mode, and capture mode
  - Real time counter with separate oscillator
  - Six PWM channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package
  - 6-channel 10-bit ADC in PDIP Package
  - Programmable serial USART
  - Master/slave SPI serial interface
  - Byte-oriented 2-wire serial interface (Philips I^2C compatible)
  - Programmable watchdog timer with separate on-chip oscillator
  - On-chip analog comparator
  - Interrupt and wake-up on pin change
- Special microcontroller features
  - DebugWIRE on-chip debug system
  - Power-on reset and programmable brown-out detection
  - Internal calibrated oscillator
  - External and internal interrupt sources
  - Five sleep modes: Idle, ADC noise reduction, power-save, power-down, and standby
- I/O and packages
  - 23 programmable I/O lines
  - 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF
- Operating voltage:
  - 1.8V - 5.5V for Atmel ATmega48V/88V/168V
  - 2.7V - 5.5V for Atmel ATmega48/88/168
- Temperature range:
  - -40°C to 85°C
- Speed grade:
  - ATmega48V/88V/168V: 0 - 4MHz @ 1.8V - 5.5V, 0 - 10MHz @ 2.7V - 5.5V
  - ATmega48/88/168: 0 - 10MHz @ 2.7V - 5.5V, 0 - 20MHz @ 4.5V - 5.5V
- Low power consumption
  - Active mode:
    - 250µA at 1MHz, 1.8V
    - 15µA at 32kHz, 1.8V (including oscillator)
  - Power-down mode:
    - 0.1µA at 1.8V

Note: 1. See "Data retention" on page 8 for details.

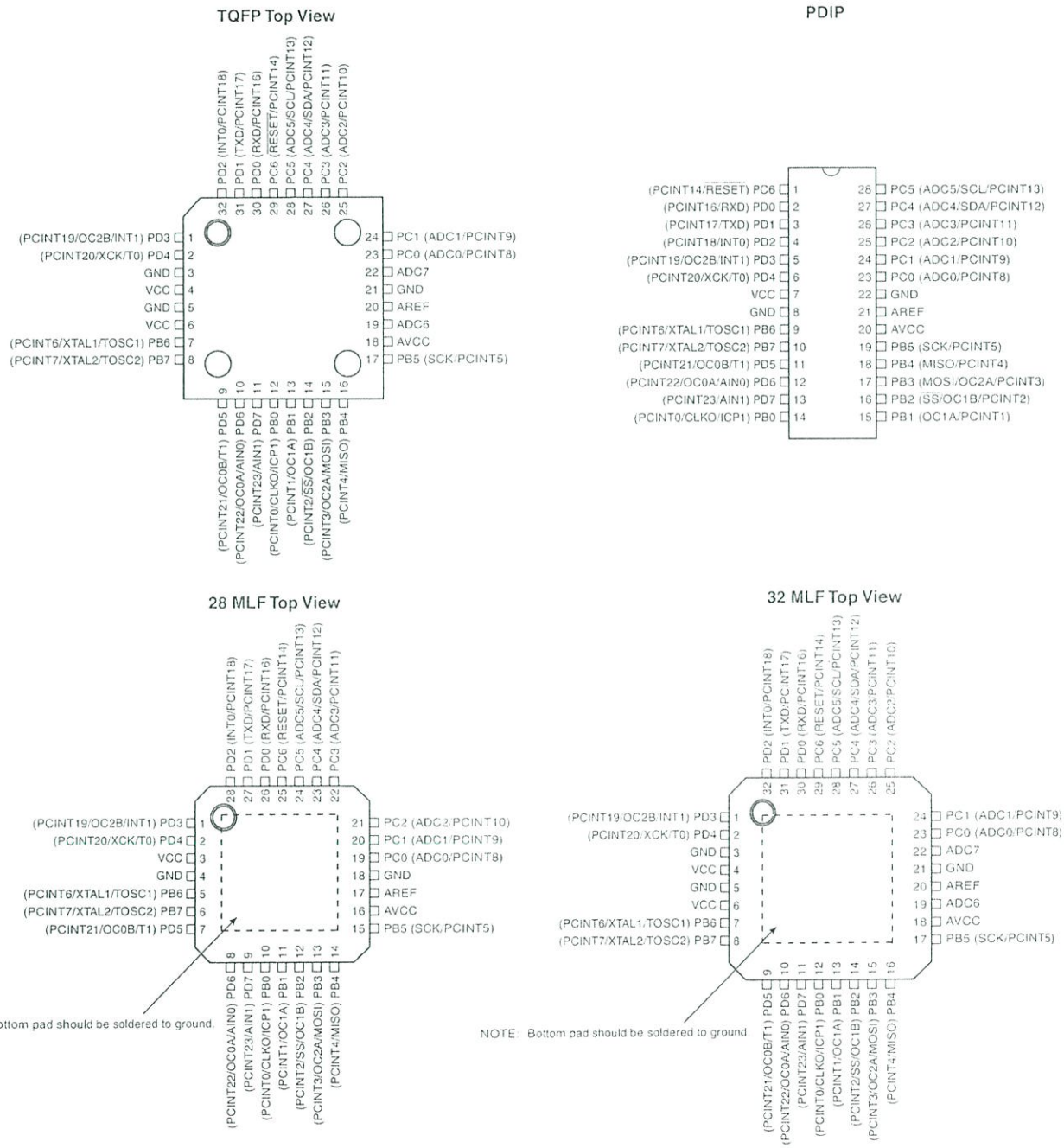
8-bit Atmel Microcontroller with 4/8/16K Bytes In-System Programmable Flash

ATmega48/V
ATmega88/V
ATmega168/V

Rev. 2545T–AVR–05/11

1. Pin configurations

Figure 1-1. Pinout Atmel ATmega48/88/168.



1.1 Pin descriptions

1.1.1 VCC

Digital supply voltage.

1.1.2 GND

Ground.

1.1.3 Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7..6 is used as TOSC2..1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of Port B are elaborated in "Alternate functions of port B" on page 78 and "System clock and clock options" on page 27.

1.1.4 Port C (PC5:0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5..0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

1.1.5 PC6/$\overline{\text{RESET}}$

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in Table 29-3 on page 307. Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated in "Alternate functions of port C" on page 81.

1.1.6 Port D (PD7:0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up

resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

The various special features of Port D are elaborated in "Alternate functions of port D" on page 84.

1.1.7 AV_{CC}

AV_{CC} is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should be externally connected to V_{CC}, even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter. Note that PC6..4 use digital supply voltage, V_{CC}.

1.1.8 AREF

AREF is the analog reference pin for the A/D Converter.

1.1.9 ADC7:6 (TQFP and QFN/MLF package only)

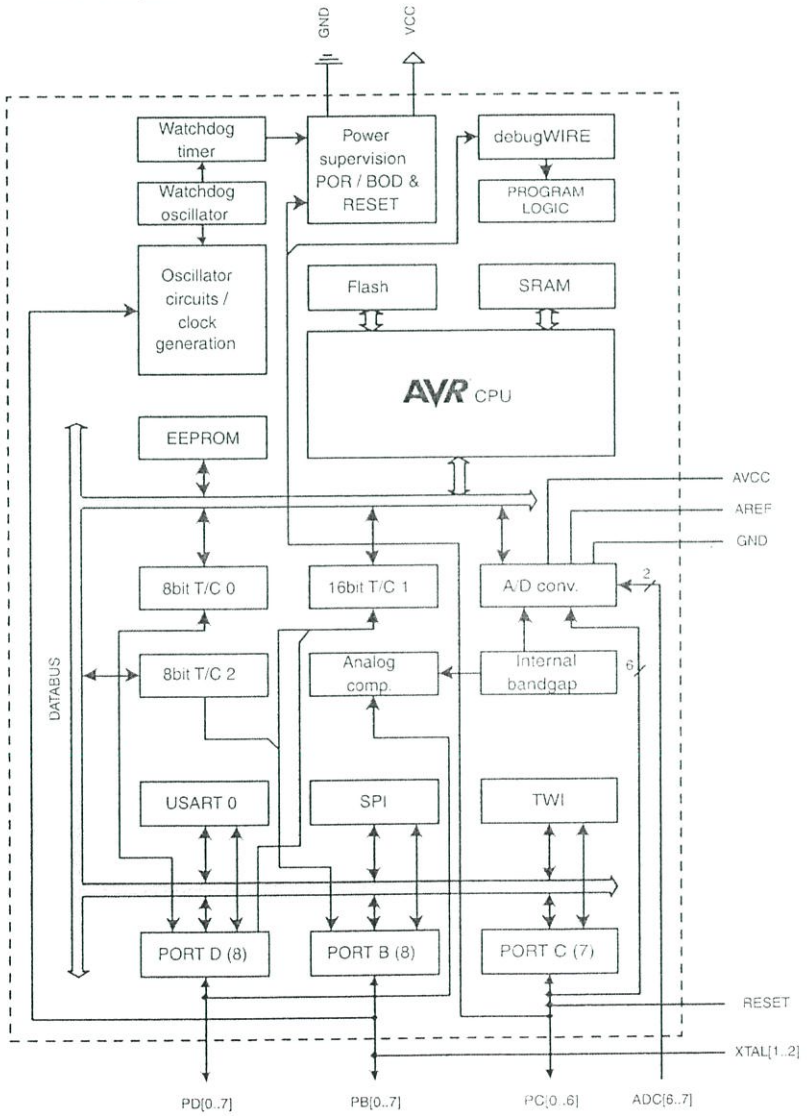
In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

2. Overview

The Atmel ATmega48/88/168 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega48/88/168 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

2.1 Block diagram

Figure 2-1. Block diagram.



The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting

architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The Atmel ATmega48/88/168 provides the following features: 4K/8K/16K bytes of In-System Programmable Flash with Read-While-Write capabilities, 256/512/512 bytes EEPROM, 512/1K/1K bytes SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte-oriented 2-wire Serial Interface, an SPI serial port, a 6-channel 10-bit ADC (8 channels in TQFP and QFN/MLF packages), a programmable Watchdog Timer with internal Oscillator, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, USART, 2-wire Serial Interface, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption.

Atmel offers the QTouch Library for embedding capacitive touch buttons, sliders and wheels functionality into AVR microcontrollers. The patented charge-transfer signal acquisition offers robust sensing and includes fully debounced reporting of touch keys and includes Adjacent Key Suppression® (AKS®) technology for unambigiuous detection of key events. The easy-to-use QTouch Suite toolchain allows you to explore, develop and debug your own touch applications.

The device is manufactured using the Atmel high density non-volatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip Boot program running on the AVR core. The Boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega48/88/168 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATmega48/88/168 AVR is supported with a full suite of program and system development tools including: C Compilers, Macro Assemblers, Program Debugger/Simulators, In-Circuit Emulators, and Evaluation kits.

2.2 Comparison between Atmel ATmega48, Atmel ATmega88, and Atmel ATmega168

The ATmega48, ATmega88 and ATmega168 differ only in memory sizes, boot loader support, and interrupt vector sizes. Table 2-1 summarizes the different memory and interrupt vector sizes for the three devices.

Table 2-1. Memory size summary.

Device	Flash	EEPROM	RAM	Interrupt vector size
ATmega48	4Kbytes	256Bytes	512Bytes	1 instruction word/vector
ATmega88	8Kbytes	512Bytes	1Kbytes	1 instruction word/vector
ATmega168	16Kbytes	512Bytes	1Kbytes	2 instruction words/vector

ATmega88 and ATmega168 support a real Read-While-Write Self-Programming mechanism. There is a separate Boot Loader Section, and the SPM instruction can only execute from there. In ATmega48, there is no Read-While-Write support and no separate Boot Loader Section. The SPM instruction can execute from the entire Flash.

3. Resources

A comprehensive set of development tools, application notes and datasheets are available for download on http://www.atmel.com/avr.

4. Data retention

Reliability Qualification results show that the projected data retention failure rate is much less than 1 PPM over 20 years at 85°C or 100 years at 25°C.

5. About code examples

This documentation contains simple code examples that briefly show how to use various parts of the device. These code examples assume that the part specific header file is included before compilation. Be aware that not all C compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C compiler documentation for more details.

For I/O Registers located in extended I/O map, "IN", "OUT", "SBIS", "SBIC", "CBI", and "SBI" instructions must be replaced with instructions that allow access to extended I/O. Typically "LDS" and "STS" combined with "SBRS", "SBRC", "SBR", and "CBR".

6. Capacitive touch sensing

The Atmel QTouch Library provides a simple to use solution to realize touch sensitive interfaces on most Atmel AVR microcontrollers. The QTouch Library includes support for the QTouch and QMatrix acquisition methods.

Touch sensing can be added to any application by linking the appropriate Atmel QTouch Library for the AVR Microcontroller. This is done by using a simple set of APIs to define the touch channels and sensors, and then calling the touch sensing API's to retrieve the channel information and determine the touch sensor states.

The QTouch Library is FREE and downloadable from the Atmel website at the following location: www.atmel.com/qtouchlibrary. For implementation details and other information, refer to the Atmel QTouch Library User Guide - also available for download from the Atmel website.