

ระบบควบคุมไฟฟ้าผ่านแอปพลิเคชันบนแอนดรอยด์
ANDROID-BASED ELECTRIC CONTROL SYSTEM FOR HOUSEHOLD

นายขวัญรัช ขวัญอ่อน

รหัส 52371177

นายอาวุธพงษ์ จินดาวัตนชัย

รหัส 52371191

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2555

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 20 ก.ค. 2558
เลขทะเบียน..... 1682689/1
เลขเรียกหนังสือ..... 2/6
มหาวิทยาลัยนเรศวร 2268 5

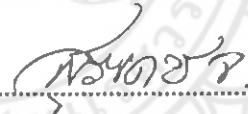
2555




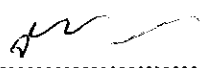
ใบรับรองปริญญาโท

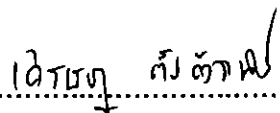
ชื่อหัวข้อโครงการ ระบบควบคุมไฟฟ้าบ้านผ่านแอปพลิเคชันบนแอนดรอยด์
ผู้ดำเนินโครงการ นายขวัญรัช ขวัญอ่อน รหัส 52371177
นายจรรพษ์ จินดาวัฒน์ชัย รหัส 52371191
ที่ปรึกษาโครงการ ดร.สุรเดช จิตประไพกุลศาล
สาขาวิชา วิศวกรรมคอมพิวเตอร์
ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา 2555

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี อนุมัติให้ปริญญาโทฉบับนี้เป็นส่วนหนึ่ง
ของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์


.....ที่ปรึกษาโครงการ
(ดร.สุรเดช จิตประไพกุลศาล)


.....กรรมการ
(ดร.พงศ์พันธ์ กิจสมาโชชิน)


.....กรรมการ
(อาจารย์ จิราพร พุกสุข)


.....กรรมการ
(อาจารย์ เสรีฐา คิ่งคำวานิช)

ชื่อหัวข้อโครงการ	ระบบควบคุมไฟบ้านผ่านแอปพลิเคชันบนแอนดรอยด์	
ผู้ดำเนินโครงการ	นายขวัญวัช ขวัญอ่อน	รหัส 52371177
	นายจารุพงษ์ จินดาวัฒนชัย	รหัส 52371191
ที่ปรึกษาโครงการ	ดร.สุรเดช จิตประไพกุลศาล	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์	
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์	
ปีการศึกษา	2555	

บทคัดย่อ

ในปัจจุบันการใช้ชีวิตในสังคมเมืองนั้นต้องดำเนินไปอย่างรวดเร็ว ต้องรีบไปทำงาน รีบออกไปนอกสถานที่ต่าง ๆ หรือรีบทำกิจกรรมต่างๆ อาจทำให้บางครั้งผู้คนจำนวนมากลืมปิดไฟในบ้านหรืออุปกรณ์เครื่องใช้ไฟฟ้าต่างๆทิ้งไว้ ผู้จัดทำจึงจัดทำโครงการระบบควบคุมไฟในบ้านผ่านแอปพลิเคชันบนแอนดรอยด์ โดยการนำอุปกรณ์อัจฉริยะ (Smart Device) เป็นตัวช่วยในการควบคุมระบบไฟฟ้าภายในบ้าน ซึ่งทำเป็นแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์โดยควบคุมผ่านเครือข่ายทางอินเทอร์เน็ต โดยสามารถควบคุมและจัดการระบบไฟฟ้าภายในบ้านผ่านทางแอปพลิเคชันได้ ถึงแม้ว่าจะไม่อยู่ภายในบ้านก็ตาม

Project title Android-based electric control system for household

Name Mr. Kwantawat Kwanon ID. 52371177

 Mr. Jarupong Jindawattanachai ID. 52371191

Project advisor DR. Suradet Jitprapaikulsarn

Major Computer Engineering

Department Electrical and Computer Engineering

Academic year 2555

.....

Abstract

In today's social life in the city to take action quickly. Rush to work Rushed out of the places or hurry make various activities sometimes may lots of people forget the switch lights in the house equipment or electrical appliances various left. Contributors to the project Android-based electric control system for household. By the Smart Device helps to control electrical systems in the house. Which make application on the Android operating system by controlling over the network of Internet. You can control and manage electrical system in your home via application although it not within the home

กิตติกรรมประกาศ

โครงการ “ระบบควบคุมไฟบ้านผ่านแอปพลิเคชันบนแอนดรอยด์” จะไม่สามารถสำเร็จได้ถ้าไม่ได้รับความช่วยเหลือจาก คร.สุรเดช จิตประไพกุลศาล อาจารย์ที่ปรึกษาโครงการนี้ ที่ให้ความกรุณาแนะนำวิธีในการทำงานให้เข้าใจถึงการศึกษาอย่างเป็นระบบขั้นตอน อีกทั้งสละเวลาเพื่อตรวจสอบการทำงานและชี้แนะแนวทางแก้ไขในทุกขั้นตอนตลอดการทำงานโครงการ และสุดท้ายนี้ขอขอบพระคุณอาจารย์ทุกท่านและเพื่อนๆ ทุกคนที่ไม่ได้เอ่ยนามที่คอยให้ความช่วยเหลือและคำแนะนำต่างๆ จนได้โครงการนี้สำเร็จได้ด้วยดี

คณะผู้ดำเนินโครงการวิศวกรรม

นาย ขวัญวัช ขวัญอ่อน

นาย จารุพงษ์ จินควัฒนชัย

มีนาคม 2555



สารบัญ

หน้า

ใบรับรองปริญญาโท.....	ก
บทคัดย่อภาษาไทย.....	ข
บทคัดย่อภาษาอังกฤษ.....	ค
กิตติกรรมประกาศ.....	ง
สารบัญ.....	จ
สารบัญ(ต่อ).....	ฉ
สารบัญรูป.....	ช
สารบัญรูป(ต่อ).....	ซ
สารบัญตาราง.....	ฎ
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.4 ขอบเขตการทำโครงการ.....	2
1.5 ขั้นตอนการดำเนินงาน.....	2
1.6 แผนการดำเนินงาน.....	3
1.7 รายละเอียดงบประมาณตลอด.....	3
บทที่ 2 หลักการและทฤษฎี	
2.1 ภาษา Java.....	4
2.2 ภาษา C#.....	5
2.3 ภาษา XML.....	7
2.4 ภาษา PHP.....	8
2.5 ภาษา C.....	10
2.6 ภาษา SQL.....	11
2.7 แอนครอชต์.....	12
2.8 เครื่องบริการ(server).....	16
2.9 อัลกอริทึม SHA-1.....	17

สารบัญ(ต่อ)

	หน้า
2.10 อุปกรณ์ฮาร์ดแวร์.....	18
บทที่ 3 วิธีดำเนินโครงการ	
3.1 การวิเคราะห์ระบบ.....	28
3.2 การออกแบบ.....	29
3.2 หลักการทำงาน.....	36
บทที่ 4 ผลการทดสอบและวิเคราะห์	
4.1 ผลการทดลองการล็อกอินเข้าไปยังหน้าควบคุมบนแอนดรอยด์.....	37
4.2 ผลการทดลองจากการสั่งแอนดรอยด์เพื่อติดต่อกับเว็บเซอร์วิส.....	39
4.3 ผลการทดลองจากการติดต่อจาก โปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์ไปยังบอร์ดET-BASE AVR EASY8841.....	42
4.4 ผลการทดลองจากส่วนที่ติดต่อกับฮาร์ดแวร์กับเว็บเซอร์วิส.....	46
4.5 ผลการทดลองจากการควบคุมจากแอปพลิเคชันบนแอนดรอยด์ไปยังฮาร์ดแวร์.....	48
4.6 ผลการทดลองจากการควบคุมจาก โปรแกรมที่ติดต่อกับฮาร์ดแวร์เพื่อ ไปแสดงผลที่แอปพลิเคชันบนแอนดรอยด์.....	50
บทที่ 5 บทสรุปและข้อเสนอแนะ	
5.1 สรุปผล.....	53
5.2 ปัญหาและอุปสรรค.....	55
5.3 แนวทางพัฒนาต่อ.....	55
5.4 ข้อเสนอแนะ.....	55
5.4 ความรู้ที่จำเป็นต่อการพัฒนา.....	55
เอกสารอ้างอิง.....	56
ภาคผนวก	
ภาคผนวก ก.....	57
ภาคผนวก ข.....	61
ภาคผนวก ค.....	65
ประวัติผู้ดำเนินโครงการ.....	70

สารบัญรูป

รูปที่	หน้า
2.1 โครงสร้างของแอนครอยด์	12
2.2 แอนครอยด์ SDK เมื่อทำการ Run Program.....	15
2.3 ET-BASE AVR EASY88.....	18
2.4 โครงสร้างของบอร์ด ET-BASE AVR EASY88.....	20
2.5 วงจรต่อลำโพงผ่านทรานซิสเตอร์ BC337.....	21
2.6 ขั้วต่อสัญญาณจาก PD[0..7].....	21
2.7 วงจร LED ใช้แสดงสถานะของขาสัญญาณ PB[5].....	22
2.8 แสดงขั้วต่อ AVRISP.....	22
2.9 วงจร ไอซีเบอร์ 74HC595.....	23
2.10 ขาสัญญาณ Output จาก 74HC595.....	23
2.11 ขาสัญญาณจาก PB[0..5].....	24
2.12 วงจรแสดงสถานะของขาสัญญาณ PB[1].....	24
2.13 ขาสัญญาณจาก PC[0..5].....	25
2.14 ขาสัญญาณ RS232.....	25
2.15 วงจรการต่อสายของ RS232 สำหรับใช้งานกับบอร์ด.....	26
2.16 วงจรสวิตช์ BL(Bootloader).....	26
3.1 สถาปัตยกรรมของระบบ.....	28
3.2 การออกแบบหน้าจอล็อกอิน.....	29
3.3 การออกแบบหน้าควบคุม.....	30
3.4 การออกแบบหน้าโปรแกรมส่วนที่ติดต่อฮาร์ดแวร์.....	31
3.5 แผนภาพแสดงการทำงานในส่วนของบอร์ดET-BASE AVR EASY88.....	32
3.6 แผนภาพแสดงการทำงานในส่วนของโปรแกรมที่ติดต่อกับฮาร์ดแวร์.....	33
3.7 แผนภาพแสดงการทำงานในส่วนของแอปพลิเคชันบนแอนครอยด์.....	34
3.8 แสดงการทำงานของหน้าจอล็อกอิน.....	35
3.9 การทำงานของแอปพลิเคชันบนแอนครอยด์.....	36
3.10 การทำงานของโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์.....	36

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.1	หน้าแรกของแอปพลิเคชันบนแอนดรอยด์.....37
4.2	กรณีกดล็อกอินโดยไม่ใส่ Username หรือ Password.....38
4.3	กรณีสื่อ Username หรือ Password ผิด.....38
4.4	ส่วนของหน้าควบคุม.....39
4.5	เมื่อทำการกดเปิดสวิตช์.....39
4.6	แอนดรอยด์ทำการติดต่อเว็บเซิร์ฟเวอร์.....40
4.7	เว็บเซิร์ฟเวอร์เปลี่ยนค่าในฐานข้อมูล.....40
4.8	XML อ่านข้อมูลจากฐานข้อมูล.....40
4.9	เมื่อกดปิดสวิตช์.....41
4.10	แอนดรอยด์ทำการติดต่อเว็บเซิร์ฟเวอร์.....41
4.11	เว็บเซิร์ฟเวอร์เปลี่ยนค่าในฐานข้อมูล.....41
4.12	XML อ่านข้อมูลจากฐานข้อมูล.....42
4.13	การตรวจสอบสาย RS232.....42
4.14	กรณีที่ไม่สามารถเชื่อมต่อบอร์ดได้.....43
4.15	กรณีที่เชื่อมต่อบอร์ดได้.....43
4.16	กรณีที่กดปุ่ม Reset บนบอร์ด.....44
4.17	ส่งค่าตัวเลข 1, 3, 5, 7 และแสดงสถานะที่โปรแกรม.....44
4.18	ผลจากการส่งค่า 1, 3, 5, 7 ที่โปรแกรม.....45
4.19	ส่งค่าตัวเลข 2, 4, 6, 8 และแสดงสถานะที่โปรแกรม.....45
4.20	ผลจากการส่งค่า 2, 4, 6, 8 ที่โปรแกรม.....46
4.21	ส่งค่า 1 เข้าไปเพื่อไปติดต่อกับเว็บเซิร์ฟเวอร์.....46
4.22	ฐานข้อมูลที่ V1 เปลี่ยนจาก 0 เป็น 1.....47
4.23	XML อ่านค่าจากฐานข้อมูล.....47
4.24	getdata_relay.php อ่านค่าจากฐานข้อมูล.....47
4.25	กดเปิดที่สวิตช์บนแอปพลิเคชัน.....48
4.26	แอนดรอยด์ติดต่อกับทางเว็บเซิร์ฟเวอร์.....48
4.27	เว็บเซิร์ฟเวอร์เปลี่ยนข้อมูลในฐานข้อมูลที่ V1 จาก 0 เป็น 1.....49
4.28	XML อ่านค่าจากฐานข้อมูล.....49

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.29 โปรแกรมอ่านค่าจากเว็บเซอร์วิสที่เป็น XML.....	49
4.30 ตั้งรีเลย์ควงที่ 1 ไฟติด.....	50
4.31 พิมพ์ 3 แล้วกดส่งที่โปรแกรม.....	50
4.32 โปรแกรมมาติดต่อเว็บเซอร์วิส.....	51
4.33 เว็บเซอร์วิสเปลี่ยนข้อมูลจากฐานข้อมูลที่ V2 จาก 0 เป็น 1.....	51
4.34 getdata_relay.php เปลี่ยนข้อมูลจากฐานข้อมูลที่ V2 จาก 0 เป็น 1.....	51
4.35 แอนครอยด์ทำการเปลี่ยนสวิตซ์จากปิดเป็นเปิด	52
ก-1 เข้าไปดาวน์โหลด โปรแกรม.....	57
ก-2 แดกไฟล์ไว้ที่ C:\eclipse	57
ก-3 กรณีที่ขึ้น Error.....	57
ก-4 ดาวน์โหลด Java Development JDK	58
ก-5 โหลดที่เป็น JDK เลือกให้ถูกต้องกับ OS.....	58
ก-6 ทำการดับเบิ้ลคลิกเพื่อติดตั้ง.....	58
ก-7 คลิกเลือกที่ Next	59
ก-8 ทำคลิกเลือกที่ Next	59
ก-9 รอจนกว่าจะทำการติดตั้งเสร็จ.....	59
ก-10 คลิกทำคลิกเลือกที่ Next	60
ก-11 กำลังติดตั้ง Eclipse ให้รอจนเสร็จสิ้น.....	60
ก-12 การติดตั้งเสร็จสิ้น.....	60
ข-1 ดาวน์โหลด Android SDK.....	61
ข-2 แดกไฟล์ไว้ที่ C:\sdk.....	61
ข-3 เปิดโปรแกรม Eclipse ขึ้นมา.....	61
ข-4 Install ADT Plug-in.....	62
ข-5 รอค้นหา.....	62
ข-6 โหลดตัว ADT Plug-in.....	63
ข-7 Eclipse จะทำการ Reboot.....	63
ข-8 ให้เลือกที่ Windows --> Android SDK Manager	63
ข-9 เลือกเวอร์ชันของแอนครอยด์ที่ต้องการพัฒนา.....	64
ข-10 เลือกที่ Accept All แล้วกดที่ Install Accepted.....	64

สารบัญรูป(ต่อ)

รูปที่	หน้า
ข-11 รองนควาน์โหลดเสร็จ.....	64
ค-1 เลือกภาษา English กดตกลง.....	65
ค-2 กดที่ Next.....	65
ค-3 กดที่ I Agree.....	66
ค-4 เลือก Folder ที่จะติดตั้งแล้วกด Next.....	66
ค-5 ทำการคลิกที่ Install.....	66
ค-6 จากนั้นกด Finish เป็นอันว่าลงเสร็จสิ้น.....	67
ค-7 ดับเบิ้ลคลิก ไฟล์ที่ควาน์โหลดมา.....	67
ค-8 เลือก I accept the terms of the license agreement แล้วกด Next	68
ค-9 เลือกโฟลเดอร์ที่จะทำการติดตั้ง แล้วกด Next.....	68
ค-10 กดที่ Next	68
ค-11 กดที่ Finish เป็นอันเสร็จสิ้น.....	69

สารบัญตาราง

ตารางที่	หน้า
1.1 แผนการดำเนินงาน.....	3
5.1 แสดงการทำงานเมื่อกดสวิตช์เปิด.....	53
5.2 แสดงการทำงานเมื่อกดสวิตช์ปิด.....	54
5.3 แสดงการทำงานเมื่อส่งค่าผ่านโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์.....	54
5.4 แสดงปัญหาที่พบและวิธีการแก้ไข.....	55



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

ในปัจจุบันการใช้ชีวิตต้องดำเนินไปอย่างรวดเร็ว การไปทำงาน การรีบไปนอกสถานที่ การออกไปนอกบ้านอย่างเร่งรีบ หรือแม้แต่การออกไปทำงานนอกสถานที่ ทำให้บางคนนั้นลืมที่จะปิดไฟในบ้านในสถานที่ทำงานทำให้เกิดการสิ้นเปลืองพลังงานหรืออาจจะก่อให้เกิดความเสียหายตามมา เนื่องจากในปัจจุบันการนำได้มีการนำอุปกรณ์อัจฉริยะมาใช้งานมากขึ้นแล้วเป็นที่นิยมทั่วไป สามารถใช้งานในระบบอินเทอร์เน็ตได้ ยังมีแอปพลิเคชันที่สามารถอำนวยความสะดวกแก่ผู้ใช้มากมายนอกจากนั้นยังสามารถพัฒนาได้ด้วยตนเอง

ในปัจจุบัน ได้เห็นความสำคัญของอุปกรณ์อัจฉริยะจึงได้พัฒนาโครงการในรูปแบบของแอปพลิเคชัน โดยเกิดแนวคิดที่จะควบคุมไฟฟ้าภายในบ้านผ่านแอปพลิเคชันของอุปกรณ์อัจฉริยะขึ้นเพื่อที่จะอำนวยความสะดวกให้กับผู้ใช้ โดยการทำงานนั้นสามารถตรวจสอบได้ว่า มีดวงไฟดวงไหนเปิดค้างไว้อย่างชัดเจนว่ามีการเปิดใช้ไฟฟ้าตรงจุดไหนบ้าง และต้องการจะปิด - เปิดไฟฟ้าตรงจุดไหนภายในบ้านได้ ก็สามารถแก้ไขได้อย่างรวดเร็ว

ในการสร้างระบบควบคุมไฟฟ้าภายในบ้านได้ศึกษาการทำงานของอุปกรณ์อิเล็กทรอนิกส์ที่ใช้ตรวจับการปิด - เปิดอุปกรณ์ไฟฟ้า อุปกรณ์อิเล็กทรอนิกส์ที่ใช้ปิด - เปิดอุปกรณ์ไฟฟ้าอัตโนมัติ, อุปกรณ์ที่ใช้เป็นตัวส่งสัญญาณให้กับโทรศัพท์เพื่อแสดงถึงสถานะการทำงานในขณะนั้น และการเชื่อมต่อผ่านทางอินเทอร์เน็ตซึ่งได้มีอยู่ในอุปกรณ์อัจฉริยะ

1.2 วัตถุประสงค์ของโครงการ

1.2.1 สามารถที่จะควบคุมและจัดการระบบไฟฟ้าภายในบ้าน โดยผ่านทางอุปกรณ์อัจฉริยะที่ใช้ระบบปฏิบัติการแอนดรอยด์

1.2.2 เพื่อพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์

1.2.3 เพื่อจำลองการจัดการระบบไฟฟ้าภายในบ้านผ่านทางอุปกรณ์อัจฉริยะที่ระบบปฏิบัติการแอนดรอยด์

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1.3.1 สามารถสั่งควบคุมไฟผ่านระบบควบคุมไฟบ้านผ่านแอปพลิเคชันบนแอนดรอยด์ได้

1.3.2 สามารถควบคุมเครื่องใช้ไฟฟ้าภายในบ้านได้ถึงแม้ไม่ได้อยู่ภายในบ้าน

1.3.3 สามารถเพิ่มความมั่นใจและความปลอดภัยในบ้านได้มากขึ้นผ่านทางอุปกรณ์อัจฉริยะที่ใช้ระบบปฏิบัติการแอนดรอยด์โดยสถานะตรวจสอบสถานะ

1.3.4 สามารถใช้อุปกรณ์อัจฉริยะที่ใช้ระบบปฏิบัติการแอนดรอยด์ให้เกิดประโยชน์

1.4 ขอบเขตการทำโครงการ

1.4.1 สามารถจำลองการควบคุมการเปิด-ปิดของอุปกรณ์ไฟฟ้าภายในบ้านในระยะไกล โดยทางแอปพลิเคชันบนอุปกรณ์อัจฉริยะที่ใช้ระบบปฏิบัติการแอนดรอยด์เวอร์ชัน 4.0.3 ผ่านทางอินเทอร์เน็ต

1.4.2 สามารถแสดงสถานะผ่านทางอุปกรณ์อัจฉริยะที่ใช้ระบบปฏิบัติการแอนดรอยด์เวอร์ชัน 4.0.3 เมื่อมีการเปิดหรือปิดไฟผ่านทางโปรแกรม

1.4.3 ศึกษาเกี่ยวกับทฤษฎีและหลักการในสิ่งต่างๆ

1.5 ขั้นตอนการดำเนินงาน

1.5.1 ศึกษาเกี่ยวกับทฤษฎีและหลักการในสิ่งต่างๆ

1.5.2 ออกแบบและพัฒนาโปรแกรม

1.5.3 ทดสอบโปรแกรม

1.5.4 ทำการปรับปรุงและแก้ไขโปรแกรม

1.5.5 วิเคราะห์การทดสอบพร้อมทั้งสรุปผล

1.5.6 จัดทำเป็นรูปเล่ม

1.6 แผนการดำเนินงาน

ตารางที่ 1.1 แผนการดำเนินงาน

กิจกรรม	ปี 2555						ปี 2556					
	ก.ค	ส.ค	ก.ย	ต.ค	พ.ย	ธ.ค	ม.ค	ก.พ	มี.ค	เม.ย	พ.ค	
1. ศึกษาและหาข้อมูลเกี่ยวกับ ทฤษฎีต่างๆที่เกี่ยวข้อง	↔											
2. เตรียมอุปกรณ์และโปรแกรม สำหรับดำเนินงาน	↔	↔										
3. ออกแบบวงจรด้านฮาร์ดแวร์			↔	↔								
4. ทดสอบฮาร์ดแวร์					↔	↔						
5. ออกแบบและเขียนโปรแกรม ทางด้านซอฟต์แวร์							↔	↔				
6. ทดสอบซอฟต์แวร์									↔	↔		
7. วิเคราะห์การทดลองและสรุป การทดลอง											↔	↔
8. จัดทำรูปเล่มรายงาน												↔

1.7 รายละเอียดงบประมาณตลอดโครงการ

1. ค่าเอกสาร	500	บาท
2. อุปกรณ์และเครื่องมือ	500	บาท
3. ค่าใช้จ่ายอื่นๆเช่น ค่าจัดทำรูปเล่มรายงาน	1,000	บาท
รวมเป็นเงิน	2,000	บาท

บทที่ 2

หลักการและทฤษฎีเบื้องต้น

2.1 ภาษา Java

Java หรือ Java programming language คือภาษาโปรแกรมเชิงวัตถุ (Object-Oriented Programming) พัฒนาโดย เจมส์ กอสลิง (James Gosling) และวิศวกรคนอื่นๆที่บริษัทซันไมโครซิสเต็มส์ ภาษานี้มีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัส (C++) โดยรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาอ็อบเจกต์ซีฟซี (Objective-C) แต่เดิมภาษานี้เรียกว่า ภาษาโอ๊ก (Oak) ซึ่งตั้งชื่อตามต้นไม้โอ๊กใกล้ที่ทำงานของ เจมส์ กอสลิง (James Gosling) แล้วภายหลังจึงเปลี่ยนไปใช้ชื่อ "Java" ซึ่งเป็นชื่อ "Coffee" แทนจุดเด่นของภาษา Java อยู่ที่ผู้เขียน โปรแกรมสามารถใช้หลักการของภาษาโปรแกรมเชิงวัตถุพัฒนา (Object-Oriented Programming) มาพัฒนาโปรแกรมของคนด้วย Java ได้

ข้อดีของ ภาษา Java

ผู้จัดทำโครงการานได้เลือกใช้ภาษา Java เนื่องจากลักษณะภาษามีข้อดีดังต่อไปนี้

- ภาษา Java เป็นภาษาที่สนับสนุนการเขียน โปรแกรมเชิงวัตถุแบบสมบูรณ์ ซึ่งเหมาะสำหรับพัฒนาระบบที่มีความซับซ้อนการพัฒนาโปรแกรมแบบวัตถุจะช่วยให้เราสามารถใช้คำหรือชื่อ ต่าง ๆ ที่มีอยู่ในระบบงานนั้นมาใช้ในการออกแบบ โปรแกรมได้ ทำให้เข้าใจได้ง่ายขึ้น
- โปรแกรมที่เขียนขึ้น โดยใช้ภาษา Java จะมีความสามารถทำงานได้ในระบบปฏิบัติการที่แตกต่างกัน ไม่จำเป็นต้องดัดแปลงแก้ไขโปรแกรม เช่น หากเขียน โปรแกรมบนเครื่อง Sun โปรแกรมนั้นก็สามารถูก compile และ run บนเครื่องพีซีธรรมดาได้
- ภาษา Java มีการตรวจสอบข้อผิดพลาดทั้งตอน compile time และ runtime ทำให้ลดข้อผิดพลาดที่อาจเกิดขึ้นใน โปรแกรม และช่วยให้ debug โปรแกรมได้ง่าย
- ภาษา Java มีความซับซ้อนน้อยกว่าภาษา C++ เมื่อเปรียบเทียบ code ของโปรแกรมที่เขียนขึ้น โดยภาษา Java กับ C++ พบว่า โปรแกรมที่เขียน โดยภาษา Java จะมีจำนวน code น้อยกว่าโปรแกรมที่เขียน โดยภาษา C++ ทำให้ใช้งานได้ง่ายกว่าและลดความผิดพลาดได้มากขึ้น
- ภาษา Java ถูกออกแบบมาให้มีความปลอดภัยสูงตั้งแต่แรก ทำให้โปรแกรมที่เขียนขึ้นด้วย Java มีความปลอดภัยมากกว่าโปรแกรมที่เขียนขึ้นด้วยภาษาอื่นเพราะ Java มี security ทั้ง low level และ high level ได้แก่ electronic signature, public and private key management

- มี IDE, application server, และ library ต่าง ๆ มากมายสำหรับงานที่เราสามารถใช้งานได้โดยไม่ต้องเสียค่าใช้จ่าย ทำให้เราสามารถลดค่าใช้จ่ายที่ต้องเสียไปกับการซื้อ tool และซอฟต์แวร์ต่างๆ

2.2 ภาษา C#

ภาษา C# เป็นภาษาโปรแกรมเชิงวัตถุทำงานบน .NET Framework พัฒนาโดยบริษัท ไมโครซอฟท์ และมี Anders Hejlsberg เป็นหัวหน้าโครงการ โดยมีรากฐานมาจากภาษา C++ และภาษาอื่นๆ (โดยเฉพาะภาษา Delphi และ Java) โดยปัจจุบันภาษา C# เป็นภาษามาตรฐานรองรับโดย ECMA

ซึ่งในปัจจุบัน ได้พัฒนาและปรับปรุงรูปแบบของภาษา C# อยู่ตลอดเวลาโดยทาง Microsoft ได้นำภาษา C# ไปอยู่ในชุดพัฒนาซอฟต์แวร์อย่าง Visual Studio ซึ่งทำให้เป็นที่นิยมเพิ่มมากขึ้น ไมโครซอฟท์ส่งมาตรฐานภาษา C# ให้กับ ECMA และได้รับการยอมรับเป็นมาตรฐาน ECMA ในเดือนธันวาคม ค.ศ.2001 ในชื่อว่า ECMA-334 C# Language Specification ในค.ศ.2003 ภาษา C# ได้รับการยอมรับเป็นมาตรฐาน ISO (ISO/IEC 23270) มาตรฐาน ISO/IEC 23270:2003 ระบุรูปแบบและกำหนดการแปล โปรแกรมที่เขียนด้วยภาษา C# โดยตัวมาตรฐานได้ระบุ

- รูปแบบการนำเสนอ
- ไวยากรณ์
- กฎการศึกษาสำหรับแปล โปรแกรมภาษา C#
- ข้อห้าม และข้อจำกัด ของเครื่องมือที่สร้างตามข้อกำหนดของ C# ISO/IEC
- กลไกในการแปลง โปรแกรมภาษา C# เพื่อใช้ในระบบประมวลผลข้อมูล
- กลไกในการเรียกให้โปรแกรมภาษา C# ทำงาน เพื่อใช้ในระบบประมวลผลข้อมูล
- กลไกในการแปลงข้อมูลเข้าเพื่อใช้กับโปรแกรมภาษา C#
- กลไกในการแปลงข้อมูลออกจากถูกประมวลผลโดยโปรแกรมภาษา C#

นอกจากนี้ตัวมาตรฐานไม่ได้กล่าวถึง โครงสร้างข้อมูล (Data Structure) และตัว Library กลางของ .NET Framework

ข้อดีของภาษา C#

เป็นรูปแบบของภาษาที่ทำงานเป็นลำดับ (Sequential) และต้องผ่านการแปลงไฟล์ให้อยู่ในรูปแบบพร้อมใช้งานหรือ EXE file (ที่เรียกว่าการ Compile) เช่นเดียวกับภาษา C โดย C# เป็นภาษาที่ถูกผลักดันโดยบริษัท ไมโครซอฟท์ให้ออกมายิ่งใหญ่โดยมีจุดเด่นดังนี้

- เขียนคล่อง C# มีเครื่องมือในการช่วยเขียนอย่าง Visual Studio ที่มีฟีเจอร์ในการไฮไลต์โค้ด และส่วนช่วยในการเขียนโปรแกรมที่สะดวกมาก
- เขียนง่ายจุดเด่นของตัวแปรและอ็อบเจกต์บน C# เค้นชัดในเรื่องของการจัดการคุณสมบัติ (Properties) และการตั้งค่าเริ่มต้นที่ช่วยให้สามารถพัฒนาระบบได้สะดวกรวดเร็วขึ้น
- อ่านง่ายการจัดระเบียบโดยตัว Visual Studio เมื่อทำการเขียนโค้ด ไปซัฟฟักหนึ่ง จะเริ่มรู้สึกคล้ายตากับย่อหน้าแต่ Visual Studio โดยปกติเมื่อจะจัดการกับย่อหน้าทั้งหมดเหล่านั้นให้อยู่ในระเบียบสะอาดตาที่สุดเลยทีเดียวนะ
- เป็นอ็อบเจกต์ (Object) เนื่องจากภาษา C# นั้นมีแม่แบบมาจากภาษา Java ซึ่งจุดเด่นคือการทำทุกสิ่งให้เป็นวัตถุ (Object) ทำให้สามารถเขียนและพัฒนาได้ง่าย
- ประสิทธิภาพสูง C# เป็นภาษาที่พัฒนาขึ้นมาภายใต้ .NET Framework ซึ่งสามารถดึงเอาความสามารถของเทคโนโลยีบน .NET ออกมาใช้ได้อย่างเต็มประสิทธิภาพ
- สามารถทำงานระดับลึกภาษา C# สามารถทำงานกับหน่วยความจำรวมได้ถึงระบบคอมพิวเตอร์ในระดับลึก โดยผ่านพอยเตอร์ (Pointer) หรือทำงานกับ โปรโตคอล TCP/IP ที่ต่ำกว่าระดับ 4 ได้
- เน้นที่ XML ภาษา C# ออกแบบมาเพื่อใช้งานร่วมกับ XML ได้อย่างราบรื่นที่สุดด้วยความช่วยเหลือของ .NET Framework
- คุณสมบัติ Generic type ช่วยยืดหยุ่นในการประกาศตัวแปร
- เทคโนโลยี LINQ ช่วยในการทำงานกับฐานข้อมูลได้อย่างมีประสิทธิภาพ

2.3 ภาษา XML

XML ย่อมาจาก Extensible Markup Language คือภาษาหนึ่งที่ใช้ในการแสดงผลข้อมูล ซึ่งภาษาที่ใช้กำหนดรูปแบบของคำสั่งภาษา HTML หรือที่เรียกว่า Meta Data ซึ่งจะใช้สำหรับกำหนดรูปแบบของคำสั่ง Markup ต่างๆถ้าเปรียบเทียบกับภาษา HTML จะแตกต่างกันที่ HTML ถูกออกแบบมาเพื่อการแสดงผลอย่างเฉียวเท่านั้น เช่น ให้แสดงผลตัวเล็ก ตัวหนา ตัวเอียง เหมือนที่คุณเคยเห็นในเว็บเพจทั่วไปแต่ภาษา XML นั้นถูกออกแบบมาเพื่อเก็บข้อมูลโดยทั้งข้อมูลและโครงสร้างของข้อมูลนั้นๆไว้ด้วยกัน ส่วนการแสดงผลก็จะใช้ภาษาเฉพาะซึ่งก็คือ XSL (Extensible Stylesheet Language)

ภาษา XML มีโครงสร้างที่ประกอบด้วยแท็กเปิด และแท็กปิด เช่นเดียวกับภาษา HTML แต่ภาษา XML สามารถสร้างแท็กรวมทั้งกำหนดโครงสร้างของข้อมูลได้เอง ซึ่งความสามารถครั้งนี้ตัวภาษาทำไม่ได้เพราะภาษา HTML ถูกกำหนดแท็กตายตัวโดย W3C หรือ World Wide Web Consortium อาจกล่าวได้ว่า XML เป็นส่วนเสริมของ HTML เพราะตัว XML ไม่สามารถแสดงผลได้ในตัวของมันเอง หากต้องการแสดงผลที่ถูกต้อง จะต้องมีการใช้ร่วมกับภาษาอื่น เช่น HTML, JSP, PHP, ASP หรือภาษาอื่น ๆ ที่สนับสนุน XML จะมีนามสกุลเป็น .XML สามารถสร้างขึ้นจากโปรแกรมประเภท Text Editor ใดก็ได้ เช่น Notepad, Editplus, DreamWeaver, MS Word เป็นต้น

สิ่งที่ถือได้ว่าเป็นเสน่ห์ของภาษา XML นั้นจะเป็นความสะดวกในการจัดการด้านระบบการติดต่อกับผู้ใช้จาก โครงสร้างของข้อมูลเราสามารถนำข้อมูลจากแหล่งมาแสดงผลและประมวลผลร่วมกันได้ ไม่ว่าจะเป็นข้อมูลลูกค้า รายการสั่งซื้อ ผลการวิจัย รายการรับชำระเงินข้อมูลเวชระเบียน รายการสินค้าหรือข้อมูลสารสนเทศอื่น ๆ ก็สามารถแปลงให้เป็น XML ได้ และในส่วนของข้อมูลสามารถปรับให้เป็นภาษา HTML ได้ สำหรับประโยชน์ในการใช้งานนั้นเราจะสามารถนำมาใช้สำหรับการเข้าถึงระบบข้อมูลขนาดใหญ่ใช้กับระบบเครือข่ายในองค์กร หรืออินเทอร์เน็ต เพื่อดูข้อมูลหรือเรียกใช้ข้อมูลที่ให้การแสดงผลทางหน้าจอที่รวดเร็ว

ข้อดีของภาษา XML

- ดูเอกสารได้ง่าย สะดวก และได้ผลดีเหมือน HTML
- สนับสนุนการประยุกต์ใช้กับงานต่าง ๆ และสนับสนุน โปรแกรมประยุกต์ต่าง ๆ
- เขียนง่าย
- อ่านได้ด้วยมนุษย์ โดยไม่ต้องอาศัย โปรแกรมหรือเครื่องมือช่วยแปล
- การเขียน XML ทำได้ด้วยการใช้ Text Editor ทั่วไปและไม่ต้องการเครื่องมือที่ซับซ้อน
- ใช้เป็นตัวควบคุมข้อมูล (Meta data) จึงเป็นแนวทางในการขนส่งข้อมูล และสร้างการเชื่อมโยงระหว่างแอปพลิเคชันได้ง่าย

- สนับสนุน UNICODE ทำให้ใช้ได้กับหลากหลายภาษา และผสมกันได้หลากหลายภาษา
- คึงเอกสาร XML มาใช้งานได้ง่าย และใช้ร่วมกับโปรแกรมประยุกต์อื่นได้ง่าย เช่น โปรแกรม DB2, Oracle, SAP เป็นต้น
- นำไปประยุกต์ใช้ในการดำเนินกิจกรรมบนเครือข่ายได้มาก เช่น E-Business, EDI, E-Commerce, การจัดการ Supply chain / Demand chain management, การดำเนินการแบบอินเทอร์เน็ต และ เว็บแอปพลิเคชัน

2.4 ภาษา PHP

PHP เป็นภาษาจําพวก scripting language คำสั่งต่างๆจะเก็บอยู่ในไฟล์ที่เรียกว่าสคริปต์ (script) และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษาสคริปต์ก็เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมา เพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า Server-side หรือ HTML-embedded scripting language เป็นเครื่องมือที่สำคัญชนิดหนึ่ง ที่ช่วยให้เราสามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น อาจจะกล่าวได้ว่า PHP ได้รับการพัฒนาขึ้นมาเพื่อแทนที่ SSI รูปแบบเดิมๆ โดยให้มีความสามารถ และมีส่วนเชื่อมต่อกับเครื่องมือชนิดอื่นมากขึ้น เช่น ติดต่อกับคลังข้อมูลหรือ database เป็นต้น

PHP ได้รับการเผยแพร่เป็นครั้งแรกในปีค.ศ.1994 จากนั้นก็มีการพัฒนาต่อมาตามลำดับเป็นเวอร์ชัน 1 ในปี 1995 เวอร์ชัน 2 (ตอนนั้นใช้ชื่อว่า PHP/FI) ในช่วงระหว่าง 1995-1997 และเวอร์ชัน 3 ช่วง 1997 ถึง 1999 จนถึงเวอร์ชัน 4 ในปัจจุบัน

PHP เป็นผลงานที่เคิบโตมาจากกลุ่มของนักพัฒนาในเชิงเปิดเผยรหัสต้นฉบับ หรือ OpenSource ดังนั้น PHP จึงมีการพัฒนาไปอย่างรวดเร็ว และแพร่หลายโดยเฉพาะอย่างยิ่งเมื่อใช้ร่วมกับ Apache Webserver ระบบปฏิบัติการอย่างเช่น Linux หรือ FreeBSD เป็นต้น ในปัจจุบัน PHP สามารถใช้ร่วมกับเว็บเซิร์ฟเวอร์หลายๆตัวบนระบบปฏิบัติการอย่างเช่น Windows 95/98/NT เป็นต้น

ข้อดีของภาษา PHP

- เป็นของฟรีว่ากันว่าสุดยอดของ เว็บเซิร์ฟเวอร์ในฝันของผู้ใช้ที่รู้จักคุณค่าของเงินก็คือระบบปฏิบัติการ Linux โปรแกรมเว็บ Apache โปรแกรมฐานข้อมูล MySQL และ Server Site Script อย่าง PHP เพราะทุกอย่างฟรีหมด
- มีความเร็ว อะไรที่เกิมาที่หลังย่อมได้เปรียบ ถ้าพูดนี้ดูเหมือนจะเป็นจริงเสมอ เพราะ PHP นำเอาข้อดีของทั้ง C, Perl และ Java มาผนวกเข้าด้วยกัน ทำให้ทำงานได้รวดเร็วกว่า CGI หรือแม้แต่ ASP และมีประสิทธิภาพ โดยเฉพาะเมื่อใช้กับ Apache Server เพราะไม่ต้องใช้โปรแกรมจากภายนอก
- Open Source การพัฒนาของโปรแกรมไม่ได้ยึดติดกับบุคคลหรือกลุ่มคนเล็กๆ แต่เปิดโอกาสให้โปรแกรมเมอร์ทั่วไปได้เข้ามาช่วยกันพัฒนา ทำให้มีคนใช้งานจำนวนมากและพัฒนาได้เร็วขึ้น
- Crossable Platform ใช้ได้กับหลายๆระบบปฏิบัติการไม่ว่าบน Windows, Unix, Linux หรืออื่นๆ โดยแทบจะไม่ต้องเปลี่ยนแปลงโค้ดคำสั่งเลย
- เรียนรู้ง่าย เนื่องจาก PHP ผังเข้าไปใน HTML และใช้โครงสร้างและไวยากรณ์ภาษาต่างๆ
- ใช้ร่วมกับ XML ได้ทันที
- ใช้ร่วมกับ Database ได้เกือบทุกยี่ห้อ ดังกล่าวไปแล้วข้างต้น
- ใช้กับระบบเพิ่มข้อมูลได้
- ใช้ร่วมกับข้อมูลตัวอักษรได้อย่างมีประสิทธิภาพ
- ใช้กับโครงสร้างข้อมูลได้ทั้งแบบ Scalar, Array, Associative array
- ใช้กับการประมวลผลภาพได้

2.5 ภาษา C

ภาษา C เป็นภาษาคอมพิวเตอร์ที่ถูกค้นคิดขึ้น โดย Denis Ritchie ในปี ค.ศ. 1970 โดยใช้ระบบปฏิบัติการของยูนิกซ์ (UNIX) นับจากนั้นมาก็ได้รับความนิยมเพิ่มขึ้นจนถึงปัจจุบัน ภาษา C สามารถติดต่อในระดับฮาร์ดแวร์ได้ดีกว่าภาษาระดับสูงอื่น ๆ ไม่ว่าจะเป็นภาษาเบสิกฟอร์แทรน ขณะเดียวกันก็มีคุณสมบัติของภาษาระดับสูงอยู่ด้วย ด้วยเหตุผลดังกล่าวจึงจัดได้ว่าภาษา C เป็นภาษาระดับกลาง (Middle-level language) ภาษา C เป็นภาษาคอมพิวเตอร์ชนิดคอมไพล์ (compiled Language) ซึ่งมีคอมไพเลอร์ (Compiler) ทำหน้าที่ในการคอมไพล์ (Compile) หรือแปลงคำสั่งทั้งหมดใน โปรแกรมให้เป็นภาษาเครื่อง (Machine Language) เพื่อให้เครื่องคอมพิวเตอร์นำคำสั่งเหล่านั้นไปทำงานต่อไป

ภาษา C มีสิ่งอำนวยความสะดวกสำหรับการเขียน โปรแกรมเชิงโครงสร้าง และสามารถกำหนดขอบข่ายตัวแปรและเรียกซ้ำ เช่นเดียวกับภาษาโปรแกรมเชิงเงื่อนไขส่วนใหญ่ในสายตระกูลภาษาอัลกอล ในขณะที่ระบบชนิดตัวแปรแบบอพลวัตช่วยป้องกันการดำเนินการที่ไม่ได้ตั้งใจ รหัสที่ทำงานได้ทั้งหมดในภาษาซีถูกบรรจุอยู่ในฟังก์ชันพารามิเตอร์ของฟังก์ชันส่งผ่านด้วยค่าของตัวแปรเสมอ ส่วนการส่งผ่านด้วยการอ้างอิงจะถูกจำลองขึ้น โดยการส่งผ่านค่าพอยเตอร์ ชนิดข้อมูลรวมแบบแตกต่าง (struct) ช่วยให้สมาชิกข้อมูลที่เกี่ยวข้องกันสามารถรวมกันและจัดการได้ในหน่วยเดียว รหัสต้นฉบับของภาษาซีเป็นรูปแบบอิสระซึ่งใช้อัฒภาค (;) เป็นตัวจบคำสั่ง (มิใช่ตัวแบ่ง)

ข้อดีของภาษา C

- เป็นภาษาที่มีลักษณะเป็น โครงสร้างจึงสามารถเขียน โปรแกรมได้ง่าย โปรแกรมที่เขียนขึ้นจะทำงาน ได้อย่างมีประสิทธิภาพสูง ทำงานได้รวดเร็วกว่าภาษาอื่นๆ
- สิ่งงานอุปกรณ์คอมพิวเตอร์ได้เกือบทุกส่วนของฮาร์ดแวร์ซึ่งภาษาระดับสูงภาษาอื่นทำงานได้น้อยกว่า
- คอมไพเลอร์ภาษา C ทุก โปรแกรมในท้องตลาดจะทำงานอ้างอิงมาตรฐาน(ANSI = American National Standards institute) เกือบทั้งหมดทำให้ โปรแกรมที่เขียนขึ้นด้วยภาษา C สามารถนำไปใช้กับคอมพิวเตอร์ได้ทุกรุ่นที่มาตรฐาน ANSI รับรอง
- ภาษา C สามารถนำไปใช้ได้บนเครื่องทุก Platform ไม่ว่าจะเป็น Intel PC ที่วิ่ง Windows 95 หรือ Windows NT หรือ แม้แต่ Linux ทั้งเครื่อง Macintosh และ เครื่องเวอร์คสเตชัน คลอคอนแมนเฟรม เนื่องจากมี compiler ของภาษา C อยู่ทั่วไป
- ภาษา C เป็นภาษาที่ง่าย ๆ คือมีแต่ข้อกำหนดในการใช้งาน หรือ syntax แต่ไม่มีฟังก์ชันสำเร็จรูป (Built-in Function) ใดๆ ดังนั้นหากผู้ใช้ต้องการทำอะไรก็ตาม ต้องเขียนทุกอย่างขึ้นเอง หรือ อาจเรียก Library Functions มาใช้งาน โดย ฟังก์ชันที่เป็นงานที่ใช้บ่อยๆ จะถูกรวบรวมไว้ใน Library Functions เช่น การจัดการข้อความ การดำเนินการเกี่ยวกับ Input/Output (I/O) การจองหน่วยความจำ

(Memory Allocation) แต่ฟังก์ชันที่วิไลศมาหรา จะไม่มีใน Standard Library เช่น ฟังก์ชันที่จัดการ Graphics ทั้งนี้จะขึ้นกับระบบที่ใช้ (เช่น เป็นระบบ UNIX หรือ Windows 95) และ สิ่งแวดล้อมในการทำงาน (เช่น GUI เป็น X-Windows หรือ Direct X) การทำเช่นนี้จะทำให้ภาษา C เป็นภาษาที่เคลื่อนย้ายได้ง่าย (portable)

2.6 ภาษา SQL

SQL ย่อมาจาก Structured Query Language เป็นภาษาที่ใช้ในการจัดการข้อมูลของฐานข้อมูลเชิงสัมพันธ์ผู้คิดค้น SQL เป็นรายแรกคือ บริษัทไอบีเอ็มหลังจากนั้นผู้ผลิตซอฟต์แวร์ด้านระบบจัดการฐานข้อมูลเชิงสัมพันธ์ได้พัฒนาระบบที่สนับสนุน SQL มากขึ้น จนเป็นที่นิยมใช้กันอย่างแพร่หลายในปัจจุบัน โดยผู้ผลิตแต่ละรายก็พยายามที่จะพัฒนาระบบจัดการฐานข้อมูลของตนให้มีลักษณะเด่นเฉพาะขึ้นมาทำให้รูปแบบการใช้คำสั่ง SQL มีรูปแบบที่แตกต่างกันไปบ้าง เช่น ORACLE ACCESS SQL Base ของ Sybase INGRES หรือ SQL Server ของ Microsoft เป็นต้น ในขณะที่ American National Standards Institute (ANSI) ได้กำหนดรูปแบบมาตรฐานของ SQL ขึ้น ซึ่งเป็นมาตรฐานคำสั่ง SQL ตาม ANSI-86 ที่ใช้เป็นมาตรฐานขั้นต่ำในการอ้างอิง อย่างไรก็ตามรูปแบบมาตรฐาน SQL ตาม ANSI-86 มีข้อจำกัดในการใช้คำสั่ง SQL เช่นกันเมื่อเปรียบเทียบกับ SQL ของระบบจัดการฐานข้อมูลผู้ผลิตบางรายได้ทำการปรับปรุงพัฒนาให้เป็นประโยชน์และง่ายสำหรับผู้ใช้อยู่ตลอดเวลา

ข้อดีของภาษา SQL

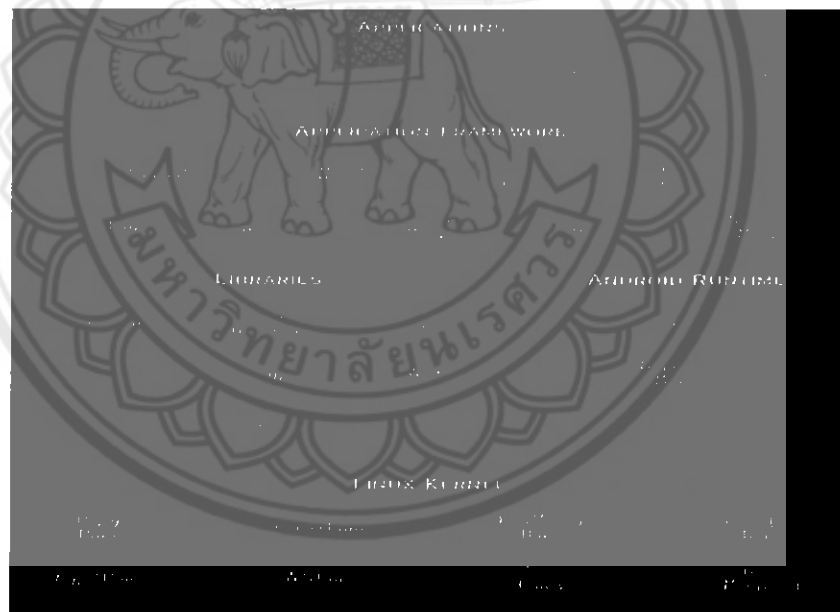
- สร้างฐานข้อมูลและ ตาราง
- สนับสนุนการจัดการฐานข้อมูล ซึ่งประกอบด้วย การเพิ่ม การปรับปรุง และการลบข้อมูล
- สนับสนุนการเรียกใช้หรือ ค้นหาข้อมูล
- SQL เป็นภาษาฐานข้อมูล ที่สามารถใช้ในเรื่องของการนิยามข้อมูล การเรียกใช้ หรือการควบคุมคำสั่งเหล่านี้จะช่วยประหยัดเวลาในการพัฒนาระบบงาน
- สามารถนำไปใช้ในส่วนของการสร้างฟอร์ม (Form) การทำรายงาน (Report) ของระบบงานต่าง ๆ ได้รวดเร็ว

2.7 แอนดรอยด์

เป็นชื่อเรียกชุดซอฟต์แวร์ หรือแพลตฟอร์ม (Platform) สำหรับอุปกรณ์อิเล็กทรอนิกส์ ที่มีหน่วยประมวลผลเป็นส่วนประกอบ เช่น คอมพิวเตอร์ โทรศัพท์ (Telephone) โทรศัพท์เคลื่อนที่ (Cell phone) อุปกรณ์เล่นอินเทอร์เน็ตขนาดพกพา (MID) เป็นต้น แอนดรอยด์นั้นถือกำเนิดอย่างเป็นทางการในวันที่ 5 พฤศจิกายน 2550 โดยบริษัท กูเกิล (google)

จุดประสงค์ของแอนดรอยด์นั้นมีจุดเริ่มต้นมาจากบริษัท Android Inc. ที่ได้นำเอา ระบบปฏิบัติการลินุกซ์ (Linux) ซึ่งนิยมนำไปใช้งานกับเครื่องบริการ (Server) เป็นหลัก นำมาลดทอนขนาดตัว (แต่ไม่ลดทอนความสามารถ) เพื่อให้เหมาะสมแก่การนำไปติดตั้งบนอุปกรณ์พกพาที่มีขนาดพื้นที่จัดเก็บข้อมูลที่จำกัด โดยคาดว่าแอนดรอยด์นั้นจะเป็นเสมือนหุ่นยนต์ตัวน้อยๆ ที่คอยช่วยเหลืออำนวยความสะดวกแก่ผู้ที่พกพามันไปในทุกที่ทุกเวลา

2.7.1 โครงสร้างของแอนดรอยด์



รูปที่ 2.1 : โครงสร้างของแอนดรอยด์

ที่มา <http://www.sourcecode.in.th/articles.php?id=71>

จากโครงสร้างของระบบปฏิบัติการแอนดรอยด์จะเห็นได้ว่า มีการแบ่งออกมาเป็นส่วนๆ ที่มีความเกี่ยวเนื่องกัน โดยส่วนบนสุดจะเป็นส่วนที่ผู้ใช้งานทำการติดต่อโดยตรงซึ่งก็คือ ส่วนของ แอปพลิเคชัน จากนั้นก็จะลำดับลงมาเป็นองค์ประกอบอื่นๆตามลำดับ และสุดท้ายจะเป็นส่วนที่ติดต่อกับอุปกรณ์โดยผ่านทาง Linux Kernel โครงสร้างของแอนดรอยด์ สามารถอธิบายเป็นส่วนๆ ได้ดังนี้

แอปพลิเคชัน หรือส่วนของโปรแกรมที่มีมากับระบบปฏิบัติการ หรือเป็นกลุ่มของโปรแกรมที่ผู้ใช้งานได้ทำการติดตั้งไว้ โดยผู้ใช้งานสามารถเรียกใช้โปรแกรมต่างๆ ได้โดยตรง ซึ่งการทำงานของแต่ละโปรแกรมจะเป็นไปตามที่ผู้พัฒนาโปรแกรมได้ออกแบบและเขียนโค้ดโปรแกรมเอาไว้

แอปพลิเคชัน Framework เป็นส่วนที่มีการพัฒนาขึ้นเพื่อให้นักพัฒนาสามารถพัฒนาโปรแกรมได้สะดวก และมีประสิทธิภาพมากยิ่งขึ้น โดยนักพัฒนาไม่จำเป็นต้องพัฒนาในส่วนที่มีความยุ่งยากมากๆ เพียงแค่ทำการศึกษาดังวิธีการเรียกใช้งาน แอปพลิเคชัน Framework ในส่วนที่ต้องการใช้งาน แล้วนำมาใช้งาน ซึ่งมีหลายกลุ่มด้วยกัน ตัวอย่างเช่น

- Activities Manager เป็นกลุ่มของชุดคำสั่งที่จัดการเกี่ยวกับวงจรการทำงานของหน้าต่างโปรแกรม (Activity)
- Content Providers เป็นกลุ่มของชุดคำสั่งที่ใช้ในการเข้าถึงข้อมูลของโปรแกรมอื่น และสามารถแบ่งปันข้อมูลให้โปรแกรมอื่นเข้าถึงได้
- View System เป็นกลุ่มของชุดคำสั่งที่เกี่ยวกับการจัดการ โครงสร้างของหน้าจอที่แสดงผลในส่วนที่ติดต่อกับผู้ใช้งาน (User Interface)
- Telephony Manager เป็นกลุ่มของชุดคำสั่งที่ใช้ในการเข้าถึงข้อมูลด้านโทรศัพท์ เช่นหมายเลขโทรศัพท์ เป็นต้น
- Resource Manager เป็นกลุ่มของชุดคำสั่งในการเข้าถึงข้อมูลที่เป็น ข้อความ, รูปภาพ
- Location Manager เป็นกลุ่มของชุดคำสั่งที่เกี่ยวกับตำแหน่งทางภูมิศาสตร์ ที่ระบบปฏิบัติการได้รับค่าจากอุปกรณ์
- Notification Manager เป็นกลุ่มของชุดคำสั่งที่จะถูกเรียกใช้เมื่อ โปรแกรม ต้องการแสดงผลให้กับผู้ใช้งาน ผ่านทางแถบสถานะ (Status Bar) ของหน้าจอ

Libraries เป็นส่วนของชุดคำสั่งที่พัฒนาด้วย C/C++ โดยแบ่งชุดคำสั่งออกเป็นกลุ่มตามวัตถุประสงค์ของการใช้งาน เช่น Surface Manage จัดการเกี่ยวกับการแสดงผล Media Framework จัดการเกี่ยวกับการการแสดงผลและเสียง Open GL | ES และ SGL จัดการเกี่ยวกับภาพ 3 มิติ และ 2 มิติ

แอนดรอยด์ Runtime จะมี DAVIK Virtual Machine ที่ถูกออกแบบมา เพื่อให้ทำงานบนอุปกรณ์ที่มี หน่วยความจำ (Memory) หน่วยประมวลผลกลาง (CPU) และพลังงาน (Battery) ที่จำกัด ซึ่งการทำงานของ DAVIK Virtual Machine จะทำการแปลงไฟล์ที่ต้องการทำงาน ไปเป็นไฟล์ .DEX ก่อนการทำงาน เหตุผลก็เพื่อให้มีประสิทธิภาพเพิ่มขึ้นเมื่อใช้งานกับ หน่วยประมวลผลกลาง

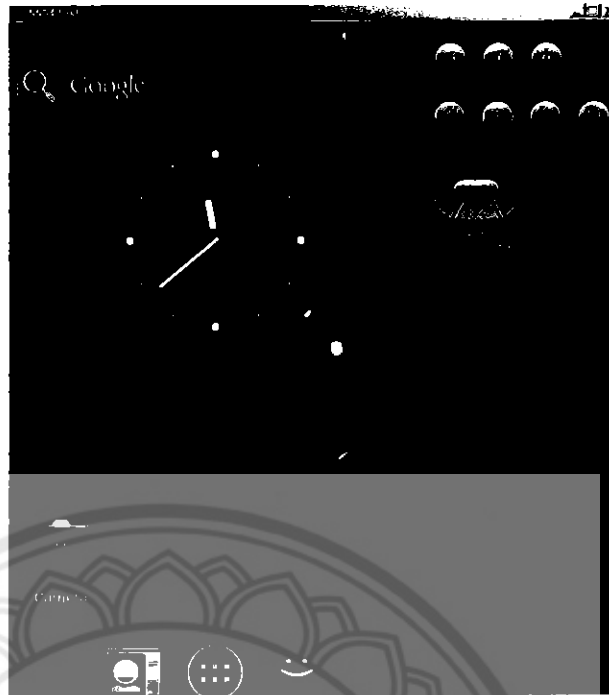
ที่มีความเร็วไม่มาก ส่วนต่อมาคือ Core Libraries ที่เป็นส่วนรวบรวมคำสั่งและชุดคำสั่งสำคัญ โดยถูกเขียนด้วยภาษาจาวา (Java Language)

Linux Kernel เป็นส่วนที่ทำหน้าที่หัวใจสำคัญในการจัดการกับบริการหลักของระบบปฏิบัติการ เช่น เรื่องหน่วยความจำ พลังงาน ติดต่อกับอุปกรณ์ต่างๆ ความปลอดภัย เครือข่าย โดยแอนดรอยด์ได้นำเอาส่วนนี้มาจากระบบปฏิบัติการลินุกซ์ รุ่น 2.6 (Linux 2.6. Kernel) ซึ่งได้มีการออกแบบมาเป็นอย่างดี

2.7.2 แอนดรอยด์ SDK (Android Software Development Kit)

นอกเหนือจากการใช้งานชุดพัฒนาโปรแกรมของระบบปฏิบัติการนั้นๆ ก็เป็นขั้นตอนหนึ่งของของการพัฒนาโปรแกรม แน่นอนว่าสำหรับแอนดรอยด์เองก็มีชุดพัฒนาโปรแกรมให้นักพัฒนาโปรแกรมได้ใช้งานกัน โดยชื่อของชุดพัฒนาโปรแกรมของแอนดรอยด์มีชื่อว่า แอนดรอยด์ SDK โดย SDK ย่อมาจาก Software Development Kit ซึ่งเป็นชุดโปรแกรมที่ทาง Google พัฒนาออกมาเพื่อแจกจ่ายให้นักพัฒนาแอปพลิเคชัน หรือผู้สนใจทั่วไปดาวน์โหลดไปใช้งานกันได้ โดยไม่มีค่าใช้จ่ายใดๆ (ฟรีสำหรับการใช้งานโปรแกรม) ซึ่งนี่ก็เป็นหนึ่งในปัจจัยที่ทำให้แอปพลิเคชันบนแอนดรอยด์นั้นเพิ่มขึ้นอย่างรวดเร็ว

ในชุด SDK นั้นจะมีโปรแกรมและ Libraries ต่างๆ ที่จำเป็นต่อการพัฒนาแอปพลิเคชันบนแอนดรอยด์ อย่างเช่น Emulator ซึ่งทำให้ผู้พัฒนาโปรแกรมสามารถสร้างแอปพลิเคชัน และนำมาทดลองใช้งานบน Emulator ก่อน โดยมีสถานะแวดล้อมเหมือนกับการนำไปใช้งานโทรศัพท์มือถือที่ใช้ระบบปฏิบัติการแอนดรอยด์จริงๆ เลย โดยสามารถดาวน์โหลดแอนดรอยด์ SDK มีให้เลือกทั้งบน Windows, Mac และ Linux



รูปที่ 2.2 : แอนดรอยด์ SDK เมื่อทำการ Run Program

ข้อเด่นของแอนดรอยด์

เนื่องจากระบบปฏิบัติการแอนดรอยด์มีการเจริญเติบโตอย่างรวดเร็ว และมีส่วนแบ่งตลาดของอุปกรณ์ด้านนี้ขึ้นทุกขณะทำให้กลุ่มผู้ใช้งาน และกลุ่มนักพัฒนาโปรแกรมหรือผู้ที่เกี่ยวข้องกับด้านนี้จึงหันมาให้ความสำคัญกับระบบปฏิบัติการแอนดรอยด์เพิ่มมากขึ้น

เมื่อมองในด้านของกลุ่มผลิตภัณฑ์ บริษัทที่มีการพัฒนาผลิตภัณฑ์รุ่นใหม่ ได้มีการนำเอาระบบปฏิบัติการแอนดรอยด์ไปใช้ในสินค้าของตนเอง พร้อมทั้งยังมีการปรับแต่งให้ระบบปฏิบัติการมีความสามารถ การจัดวาง โปรแกรม และลูกเล่นใหม่ๆ ที่แตกต่างจากคู่แข่งในท้องตลาด โดยเฉพาะอย่างยิ่ง กลุ่มสินค้าที่เป็น มือถือรุ่นใหม่ (Smart Phone) และอุปกรณ์จอสัมผัส (Touch Screen) โดยมีคุณลักษณะแตกต่างกันไป เช่นขนาดหน้าจอ ระบบโทรศัพท์ ความเร็วของหน่วยประมวลผล ปริมาณ หน่วยความจำ แม้กระทั่งอุปกรณ์ตรวจจับต่างๆ (Sensor)

หากมองในด้านของการพัฒนาโปรแกรม ทางบริษัท กูเกิล ได้มีการพัฒนาแอปพลิเคชัน Framework ไว้สำหรับนักพัฒนาใช้งาน ได้อย่างสะดวก และไม่เกิดปัญหาเมื่อนำชุดโปรแกรมที่พัฒนาขึ้นมา ไปใช้กับอุปกรณ์ที่มีคุณลักษณะต่างกัน เช่นขนาดจออุปกรณ์ ไม่เท่ากัน ก็ยังสามารถใช้งานโปรแกรมได้เหมือนกัน เป็นต้น

2.8 เครื่องบริการ(server)

เครื่องบริการ คือ เครื่องคอมพิวเตอร์หรือระบบปฏิบัติการหรือโปรแกรมคอมพิวเตอร์ ที่ทำหน้าที่ให้บริการอย่างใดอย่างหนึ่งหรือหลายอย่าง แก่เครื่องคอมพิวเตอร์หรือโปรแกรมคอมพิวเตอร์ ที่เป็นลูกข่ายโดยปรกติแล้ว โปรแกรมคอมพิวเตอร์ที่เป็นเครื่องบริการจะทำงานบนระบบปฏิบัติการ อาจจะเป็น Linux หรือ Windows หรือ Unix ก็ได้ ดังนั้นคำว่าเครื่องบริการจึงมิได้หมายถึงคอมพิวเตอร์ เพียงอย่างเดียวแต่ยังหมายถึงระบบปฏิบัติการคอมพิวเตอร์ หรือ โปรแกรมคอมพิวเตอร์อีกด้วย ยกตัวอย่างเป็นกลุ่มๆ ได้ดังต่อไปนี้

- Web Server คือ โปรแกรมที่ทำหน้าที่ให้บริการเว็บ เช่น Apache Web Server
- Mail Server คือ โปรแกรมที่ทำหน้าที่ให้บริการอีเมล เช่น Postfix, Qmail, courier
- DNS Server คือ โปรแกรมที่ทำหน้าที่ให้บริการ โดเมนเนม เช่น Bind9
- Database Server คือ โปรแกรมที่ทำหน้าที่ให้บริการฐานข้อมูล เช่น Mysql, DB2

2.8.1 ประเภทของเครื่องบริการ

ประเภทของเครื่องบริการ โดยปกติจะแบ่งได้เป็น 4 ประเภท คือ File Server , Print Server , Database Server , Application Server การแบ่งออกเป็น 4 ประเภทนั้น แบ่งตามลักษณะการใช้งาน คือ เก็บ-บริการ ไฟล์ บริการ/บริหาร งานพิมพ์ เก็บและบริการฐานข้อมูล และบริการ/บริหาร ซอฟต์แวร์ประยุกต์ ส่วน Mail Server, Internet Server หรือประเภทอื่น ๆ ที่มีการเรียกชื่อนั้น เกิดจากการนำเอาเซิร์ฟเวอร์มากกว่า 2 ประเภทมารวมกันในตัวเดียว

- 1) File Server เป็นเครื่องบริการที่มีหน้าที่จัดเก็บไฟล์ โดยการจัดเก็บไฟล์จะทำเสมือนเป็น ฮาร์ดดิสก์รวมศูนย์ (Centerized disk storage) เสมือนว่าผู้ใช้งานทุกคนมีที่เก็บข้อมูลอยู่ที่เดียว เพราะควบคุม-บริหารง่าย การสำรองข้อมูล การ Restore ง่าย ข้อมูลดังกล่าวสามารถ Shared ให้กับ Client ได้ โดยส่วนมากข้อมูลที่อยู่ใน file Server คือ โปรแกรมและข้อมูล (Personal Data File) โดยปกติแล้วเครื่องบริการ ไม่มีหน้าที่ต้องประมวลข้อมูลเหล่านี้ เป็นเพียงแหล่งเก็บข้อมูล กล่าวง่าย ๆ ก็คือทำหน้าที่เหมือน Input/Output สำหรับไฟล์ การทำงานของเครื่องบริการที่เป็น File Server นั้นในทางเทคนิคแล้วยังไม่เรียกว่าเป็น "Client and Server" เพราะไม่มีการแบ่งโหลดการทำงานระหว่าง Client and Server แต่หน้าที่ที่ File Server จะต้องจัดการคือ มี NOS (Network Operating System) ที่ดูแลเกี่ยวกับการ "เข้าถึง"ไฟล์ ต้องมีกระบวนการ "ล็อก" ไว้ ไม่ให้เกิดความซ้ำซ้อนในการแก้ไขไฟล์ เช่น ขณะที่ผู้ใช้งานคนที่ 1 เปิด ไฟล์ A และกำลังแก้ไข (edit) อยู่ ผู้ใช้งานคนที่สองจะเปิดไฟล์ A เพื่อแก้ไขไม่ได้ (แต่เปิดเพื่ออ่านได้อย่างเดียว) แต่ถ้าหากข้อมูลนั้นเป็นฐานข้อมูลแทนที่

ไฟล์หรือฐานข้อมูลทั้งฐานข้อมูลจะถูกถือกระบวนการ ล็อกก็อาจจะเกิดเฉพาะ Record (Row) นี้เป็นหน้าที่ของ NOS และแอปพลิเคชันที่ใช้งาน

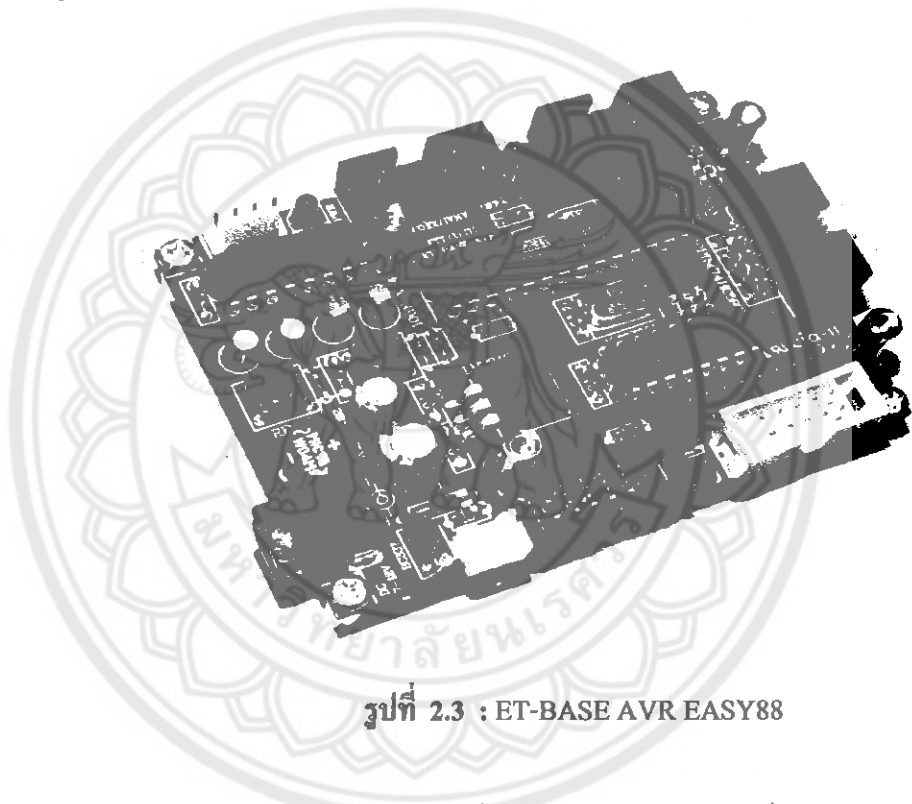
- 2) **Print server** เหตุผลที่ต้องมี Print Server ก็คือ เพื่อแบ่งให้เครื่องพิมพ์ราคาแพงบางรุ่นที่ ออกแบบมาสำหรับการทำงานหลายๆ เช่น HP Laser 5000 พิมพ์ได้ 10 – 24 แผ่นต่อหน้าที่ พรินเตอร์ประเภทนี้ ความสามารถในการทำงานสูง ถ้าหากซื้อมาเพื่อใช้งานเพียงคนเดียว แต่ละวันพิมพ์ 50 แผ่น ก็ไม่คุ้มค่า ดังนั้นจึงต้องมีกระบวนการจัดการแบ่งปันเครื่องพิมพ์ ดังกล่าวให้กับผู้ใช้ทุกๆ คนในสำนักงาน หน้าที่ในการแบ่งปัน ก็ประกอบด้วย การจัดคิว ใ้เครื่องพิมพ์ก่อน การจัดการเรื่องไฟล์ Spooling เป็นของเครื่องบริการที่มีชื่อว่า Print Server โดยส่วนใหญ่ในองค์กร น้อยองค์กรที่จะซื้อเครื่องบริการมาเพื่อใช้สำหรับเป็น Print Server โดยเฉพาะ แต่จะใช้วิธีเอาเครื่องบริการที่ซื้อมาเพื่อเป็น File Server, Data Base Server ทำเป็น Print sever ไปด้วย
- 3) **Database Server** หมายถึง เครื่องบริการที่มีไว้เพื่อรันระบบที่เป็นฐานข้อมูล DBMS (DataBase Management System) เช่น SQL , Informix เป็นต้น โดยภายในเครื่องบริการที่มี ฐานข้อมูลและตัวจัดการฐานข้อมูล ตัวจัดการฐานข้อมูลในที่นี้หมายถึง มีการแบ่งปัน การประมวลผล โดยผ่านทาง Client
- 4) **Application Server** คือ เครื่องบริการที่รัน โปรแกรมประยุกต์ได้ด้วย โดยการทำงาน สอดคล้องกับ Client เช่น Mail Server (รัน MS Exchange Server) Proxy Server (รัน Proxy Server) หรือ
- 5) **Web Server** (Run Web Server เช่น Xitami , Apache)

2.9 อัลกอริทึม SHA-1

ในการทำงานของอัลกอริทึม SHA-1 จะแบ่งข้อความต้นฉบับขนาดใดๆออกเป็นกลุ่มบิต หลายๆกลุ่มบิตที่มีขนาด 512 บิตเช่นเดียวกัน ในส่วนของการเสริมเต็มเพื่อให้มีจำนวนบิตของ ข้อความต้นฉบับ โดยรวมเป็นจำนวนเท่าของ 512 จะมีวิธีการและขั้นตอนเหมือนกับอัลกอริทึม MD5 ทุกประการ เนื่องจากอัลกอริทึม SHA-1 จะสร้างค่าแฮชผลลัพธ์ขนาด 160 บิต ตัวแปรขนาด 32 บิตที่ใช้ในการคำนวณจึงมี 5 ตัว แปร คือ a, b, c, d และ f โดยในตอนเริ่มต้นของการคำนวณ จะ มีการสร้างตัวแปร A, B, C, D และ E ขนาด 32 บิตขึ้น และมีการกำหนดค่าเบื้องต้น ในรูปของ เลขฐานสิบหกอัลกอริทึม SHA-1 จะประกอบไปด้วยรอบการทำงาน 4 รอบ แต่ละรอบจะมีการ คำเนิการที่แตกต่างกัน 20 ครั้ง กล่าวคือ ฟังก์ชันที่ไม่เป็นเชิงเส้นของตัวแปร 3 ตัวที่ใช้ในการ คำนวณหาค่าแฮชจะมีอยู่ 4 รูปแบบคือ $f(X, Y, Z) = (X \text{ AND } Y) \text{ OR } ((\text{NOT } X) \text{ AND } Z)$, สำหรับ $t = 0$ ถึง 19

แผนผังการดำเนินการ 1 รอบการทำงานของอัลกอริทึม SHA เมื่อดำเนินการครบ 20 รอบการทำงานแล้ว ค่าที่อยู่ในตัวแปร a, b, c, d และ e จะถูกบวกรวมกับค่าที่เก็บไว้ในตัวแปร A, B, C, D และ E ตามลำดับ จากนั้นอัลกอริทึม SHA-1 จะเริ่มการคำนวณค่าแฮชสำหรับกลุ่มบิตนำเข้าขนาด 512 บิตกลุ่มที่สองต่อไป ขั้นตอนการทำงานจะดำเนินไปในลักษณะนี้จนกระทั่งครบทุกกลุ่มบิตนำเข้า ค่าที่เก็บไว้ในตัวแปร A, B, C, D และ E ที่ได้เป็นครั้งสุดท้ายจะนำมาต่อรวมกันเพื่อสร้างค่าแฮชผลลัพธ์ขนาด 160 บิต

2.10 อุปกรณ์ฮาร์ดแวร์



รูปที่ 2.3 : ET-BASE AVR EASY88

ET-BASE AVR EASY88 เป็นบอร์ดไมโครคอนโทรลเลอร์ในตระกูล AVR โดยบอร์ดเลือกใช้ไมโครคอนโทรลเลอร์ตระกูล AVR เบอร์ ATMEGA88 ของ ATMEL เป็น MCU ประจำบอร์ด โดย MCU รุ่นนี้จะบรรจุอยู่ในตัวถังแบบ 28 Pin DIP โดย MCU ตัวนี้จะมีจุดเด่นคือ เป็นไมโครคอนโทรลเลอร์ขนาดเล็กแต่เพียงพร้อมไปด้วยทรัพยากรพื้นฐานต่างๆอย่างครบถ้วนเหมาะแก่การใช้ในการศึกษาเรียนรู้สำหรับผู้เริ่มต้น และยังสามารถนำไปประยุกต์ใช้งานต่างๆได้โดยง่ายซึ่ง MCU สามารถทำงานได้ด้วยความถี่สูงสุด 20MHz ที่ 1 Clock / Machine Cycle นอกจากนี้แล้วยังมีความพร้อมด้วยอุปกรณ์พื้นฐานต่างๆที่จำเป็นต่อการใช้งานไม่ว่าจะเป็นหน่วยความจำสำหรับเก็บข้อมูลแบบ EEPROM ขนาด 512 Byte และหน่วยความจำใช้งานแบบ SRAM อีก 1 K Byte ส่วนในด้านของอุปกรณ์ Peripheral นั้นก็นับว่าครบถ้วนเหมาะแก่การนำไป

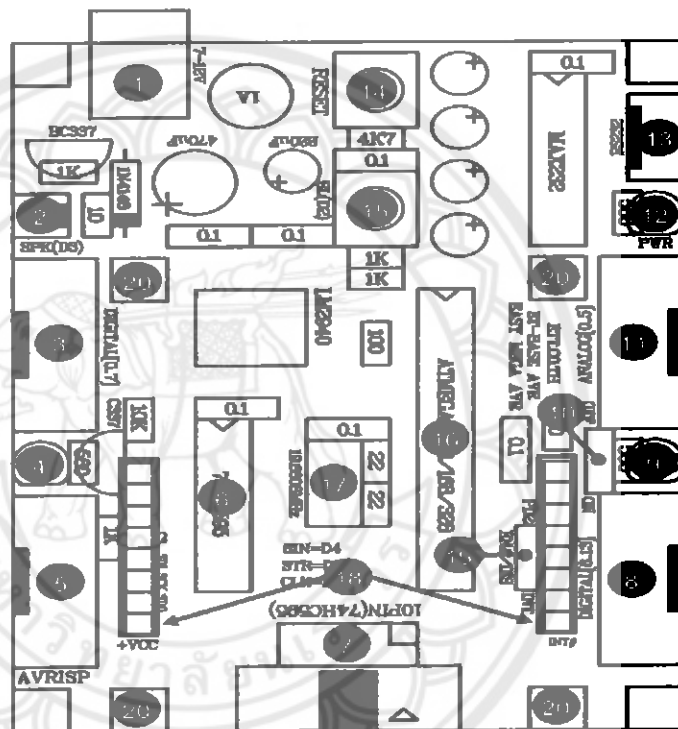
ประยุกต์ใช้งานเกี่ยวกับการควบคุมและประมวลผลต่างๆได้เป็นอย่างดี โดยจะมีทั้งระบบฮาร์ดแวร์ของ SPI, UART, I2C, Watchdog, Timer/Counter, PWM และ ADC ฯลฯ

2.10.1 คุณสมบัติของบอร์ด

- เลือกใช้ MCU ตระกูล AVR เบอร์ NATMEGA88 ของ ATMEL เป็น MCU ประจำบอร์ด โดยเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบ Crystal Oscillator ค่า 19.6608 MHz เพื่อให้สามารถใช้งานพอร์ตสื่อสารอนุกรม RS232 ได้อย่างลงตัว
- สามารถเปลี่ยนไปติดตั้งใช้งาน MCU ตระกูล AVR ขนาด 28 PIN เบอร์อื่นๆในอนุกรมเดียวกันได้เช่น ATMEGA8, ATMEGA48, ATMEGA168 และ ATMEGA328 เป็นต้น โดย MCU เหล่านี้จะมีตำแหน่งขาที่เข้ากันได้สามารถติดตั้งใช้งานในบอร์ดได้ทันทีโดยไม่ต้องดัดแปลงแก้ไขวงจร
- มีหน่วยความจำ 8 KByte Flash / 512 Byte EEPROM / 1024 Byte SRAM
- มีพอร์ต I/O ขนาด 20 บิต จำนวน 3 พอร์ต PB (6บิต) PC (6บิต) PD (8Bit) มีวงจรสื่อสารอนุกรม UART จำนวน 1 พอร์ต มีวงจรสื่อสาร SPI จำนวน 1 พอร์ต มีวงจรสื่อสาร I2C จำนวน 1 พอร์ต มีวงจร Timer/Counter ขนาด 16 บิต 1 ชุด และ Timer/Counter ขนาด 8 บิต 2 ชุด มีวงจร ADC ขนาด 10 บิต จำนวน 6 ช่อง
- MCU ประจำบอร์ดได้รับการติดตั้ง Bootloader สำหรับใช้ Upload Code ให้บอร์ดผ่านทางพอร์ตสื่อสารอนุกรม RS232 ได้ทันที โดยไม่ต้องใช้เครื่องโปรแกรมภายนอก สามารถใช้การพัฒนาโปรแกรมได้ทั้ง ภาษา Assembly ภาษา C ภาษา Basic และ ภาษา C++ ของ Arduino ได้ทันที
- มีขั้วต่อสัญญาณ I/O แบบ TTL แบบ Header 2x5 จำนวน 3 ชุด (PB PC และ PD)
- มีสวิตช์ RESET พร้อมวงจร External Reset แบบ RC Reset ภายในบอร์ด
- มีสวิตช์ BL(PD2) สำหรับใช้รีเซ็ตบอร์ดเข้าทำงานใน Bootloader และใช้ทดสอบ Input
- มีขั้วต่อ RS232 สำหรับใช้งาน และ Upload Code ด้วย Bootloader ผ่าน RS232
- มีขั้วต่อ AVRISP แบบ 10 PIN IDE มาตรฐาน AVRISP สำหรับใช้โปรแกรม Code ให้บอร์ด ในกรณีที่ไม่ต้องการใช้การ Load Code ผ่านทาง RS232 ของ Bootloader
- มีวงจรขยาย Output ด้วย 74HC595 พร้อมขั้วต่อสัญญาณ Output แบบ IDE 10 Pin จำนวน 1 ชุด
- มี LED แสดงสถานะแหล่งจ่าย Power และ 2 LED สำหรับใช้ทดสอบการทำงานของบอร์ด

- ใช้กับแหล่งจ่ายไฟ 7 ถึง VAC/DC พร้อมวงจร Bridge Rectifier และ Regulate 1A (Low Drop)
- มีฐานสำหรับใช้ติดตั้งบอร์ดทดลอง I/O ขนาดมาตรฐาน ET-MINI I/O SET เพื่อใช้งานร่วมกับบอร์ดทดลองต่างๆ เช่น DC Motor, Stepping Motor, I2C RTC, I2C I/O, I2C EEPROM, SPI ฯลฯ
- ขนาด PCB Size เล็กเพียง 8 x 6 Cm.

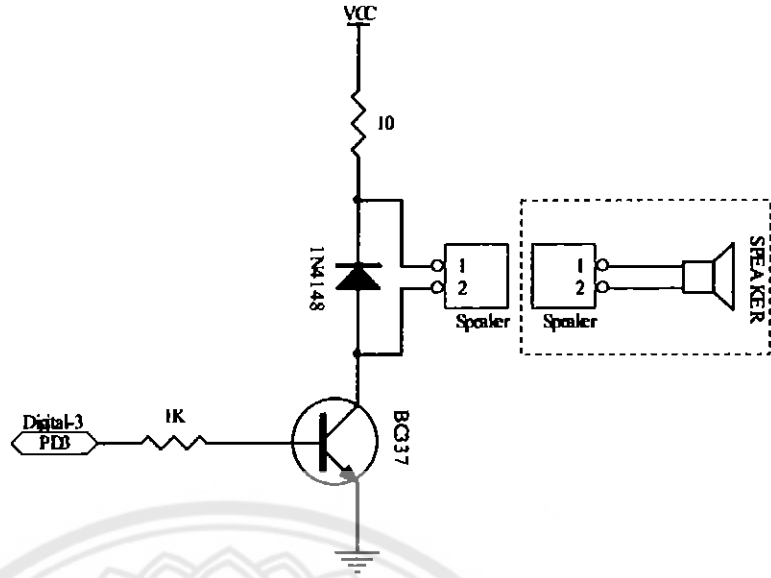
2.10.2 โครงสร้างบอร์ด ET-BASE AVR EASY88



รูปที่ 2.4 : โครงสร้างของบอร์ด ET-BASE AVR EASY88

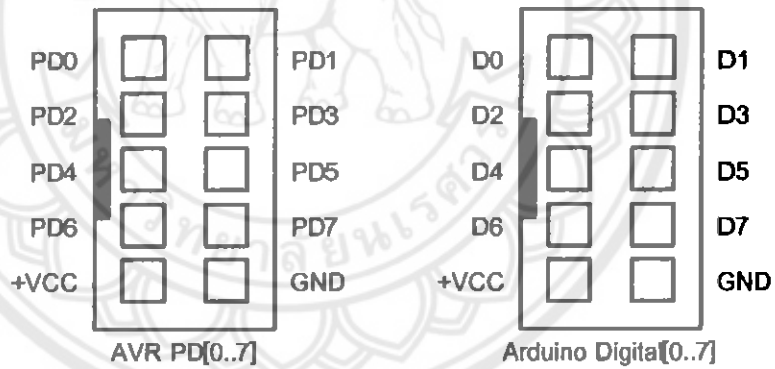
หมายเลข 1 คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงวงจรของบอร์ด ใช้กับแหล่งจ่าย 7-10VAC/DC

หมายเลข 2 คือ เป็นขั้วต่อสำหรับใช้ต่อกับลำโพงซึ่งถูก Drive ผ่านทรานซิสเตอร์ BC337



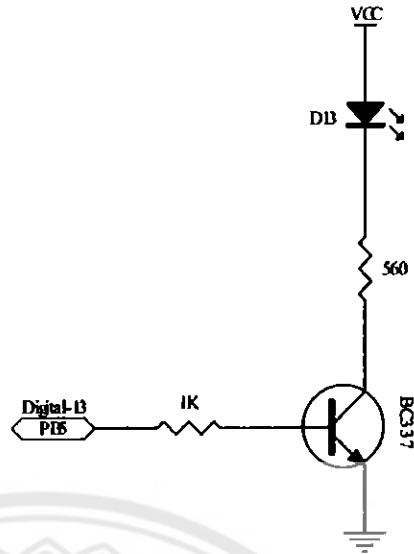
รูปที่ 2.5 : วงจรต่อลำโพง ผ่านทรานซิสเตอร์ BC337

หมายเลข 3 คือ เป็นขั้วต่อสัญญาณจาก PD[0..7] ซึ่งในกรณีใช้การพัฒนาโปรแกรมด้วย Arduino จะเป็นขาสัญญาณของ Digital[0..7]



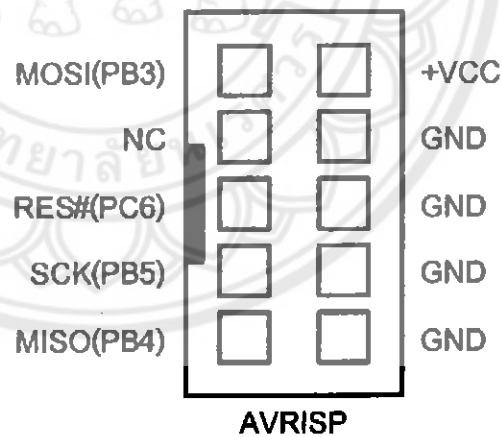
รูปที่ 2.6 : ขั้วต่อสัญญาณจาก PD[0..7]

หมายเลข 4 คือ เป็น LED ใช้แสดงสถานะของขาสัญญาณ PB[5] หรือ Digital[13] ของ Arduino ซึ่ง LED นี้จะถูก Drive ผ่านทรานซิสเตอร์ BC337



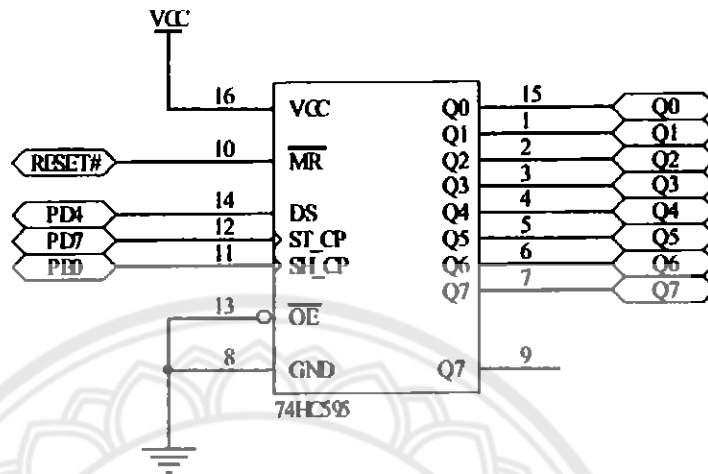
รูปที่ 2.7 : วงจร LED ใช้แสดงสถานะของขาสัญญาณ PB[5]

หมายเลข 5 คือ เป็นขั้วต่อ AVRISP ใช้สำหรับ Download Code ให้กับ MCU ในกรณีที่ใช้การพัฒนาโปรแกรมของบอร์ดเป็นแบบ MCU ของ AVR ตามปกติโดยไม่ผ่านระบบ Bootloader โดยขั้วต่อ AVRISP นี้จะสามารถใช้งานได้กับเครื่องโปรแกรมทุกรุ่นที่รองรับการใช้งานกับ ATMEGA88 และใช้ขั้วต่อ ตรงตามมาตรฐาน AVRISP



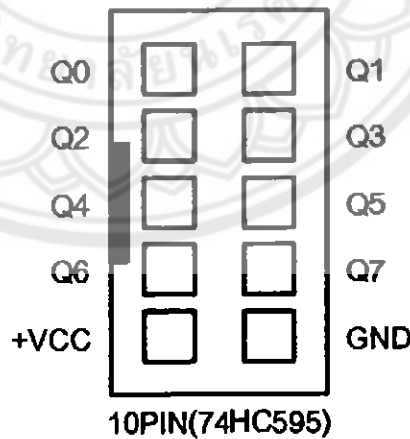
รูปที่ 2.8 : แสดงขั้วต่อ AVRISP

หมายเลข 6 คือ เป็นไอซีเบอร์ 74HC595 ซึ่งใช้ขยาย Output Port ขนาด 8 บิต โดยมีการต่อวงจรร่วมกับ MCU ที่ใช้ในบอร์ดโดย Output ของ 74HC595 นี้สามารถนำไปประยุกต์ใช้งานเพื่อทำหน้าที่เป็น Output ทั่วๆ ไป หรือใช้สำหรับเชื่อมต่อกับ Character LCD ในแบบ 4 Bit Mode



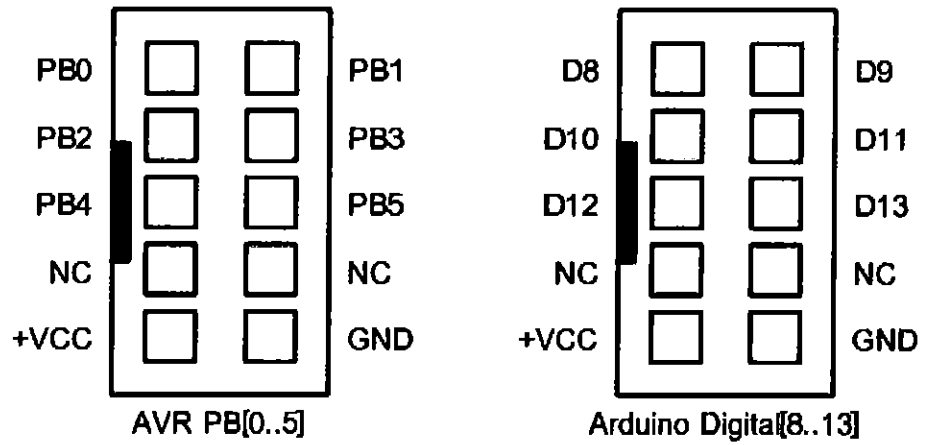
รูปที่ 2.9 : วงจรไอซีเบอร์ 74HC595

หมายเลข 7 คือ เป็นขั้วต่อสัญญาณ Output จาก 74HC595 ซึ่งมีขนาด 8 บิตก็คือ Q[0..7] โดยมี การจัดเรียงขาสัญญาณ



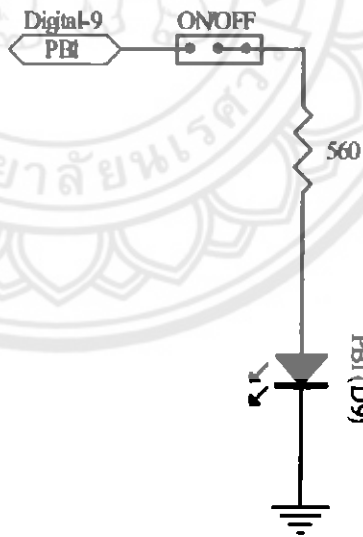
รูปที่ 2.10 : ขาสัญญาณ Output จาก 74HC595

หมายเลข 8 คือ เป็นขั้วต่อสัญญาณจาก PB[0..5] ซึ่งในกรณีใช้การพัฒนาโปรแกรมด้วย Arduino จะเป็นขาสัญญาณของ Digital[8..13]



รูปที่ 2.11 : ขาสัญญาณจาก PB[0..5]

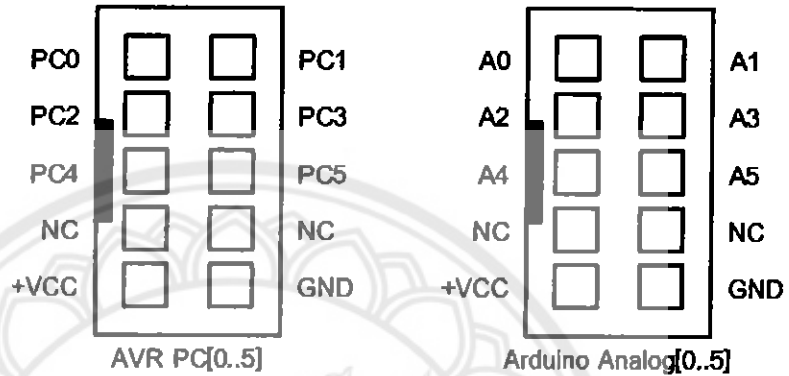
หมายเลข 9 คือ เป็น LED ใช้แสดงสถานะของขาสัญญาณ PB[1] หรือ Digital[9] ของ Arduino ซึ่ง LED นี้จะถูกต่อวงจรแบบ Source Current จากขาสัญญาณของ MCU โดยมี Jumper เป็นตัวตัด ต่อสัญญาณระหว่าง PB[1] กับ LED ซึ่ง LED นี้สามารถแสดงผลได้ 2 แบบ คือ ใช้ทดสอบการแสดงผลแบบ ON/OFF เมื่อกำหนดขาสัญญาณ PB[1] เป็นแบบ Digital Output และ ใช้ทดสอบการแสดงผลแบบ Dimmer เมื่อกำหนดขาสัญญาณ PB[1] เป็นแบบ Output PWM



รูปที่ 2.12 : วงจรแสดงสถานะของขาสัญญาณ PB[1]

หมายเลข 10 คือ เป็น Jumper สำหรับใช้ในการตัดต่อสัญญาณ PB[1] กับ LED โดยเมื่อเลือกไว้ด้าน ON จะเป็นการต่อสัญญาณ PB[1] เข้ากับ LED แต่เมื่อเลือก OFF จะเป็นการตัดการเชื่อมต่อ ของ PB[1] ออกจากวงจรแสดงผลของ LED

หมายเลข 11 คือ เป็นขั้วต่อสัญญาณจาก PC[0..5] ซึ่งในกรณีใช้การพัฒนาโปรแกรมด้วย Arduino จะเป็นขาสัญญาณของ Analog[0..5]

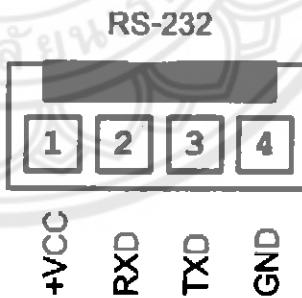


รูปที่ 2.13 : ขาสัญญาณจาก PC[0..5]

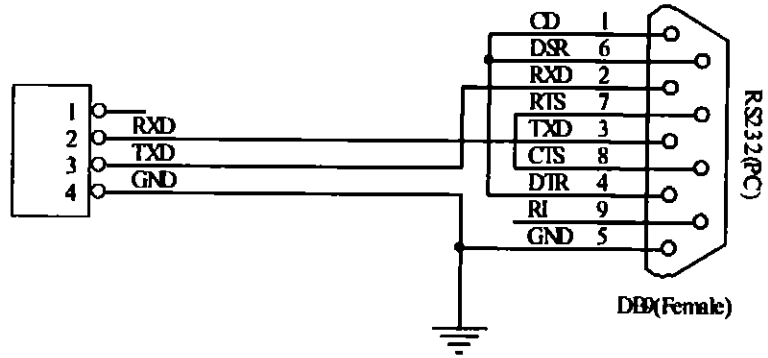
หมายเลข 12 คือ เป็น LED Power ใช้แสดงสถานะของแหล่งจ่ายไฟ +5VDC

หมายเลข 13 คือ ขั้วต่อ RS232 สำหรับใช้งานทั่วไป และ Upload Code ให้กับ MCU ผ่าน

ระบบ Bootloader



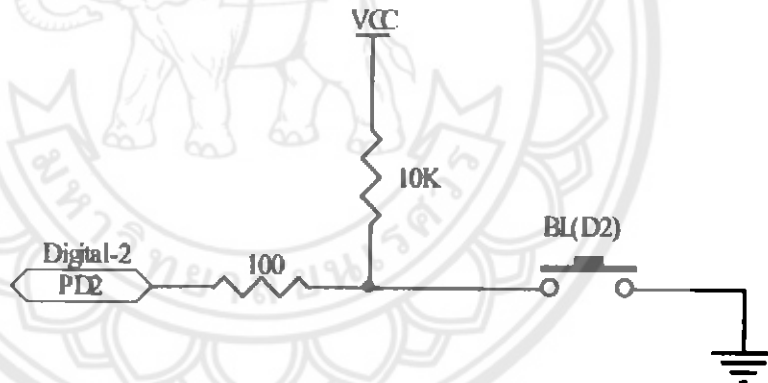
รูปที่ 2.14 : ขาสัญญาณ RS232



รูปที่ 2.15 : วงจรการต่อสายของ RS232 สำหรับใช้งานกับบอร์ด

หมายเลข 14 คือ สวิตช์ RESET ใช้สำหรับรีเซ็ตการทำงานของ MCU

หมายเลข 15 คือ สวิตช์ BL (Bootloader) โดยต่อผ่านขาสัญญาณ PD[2] ใช้สำหรับสร้างสัญญาณ Logic LOW ให้กับขาสัญญาณ PD[2] เพื่อทดสอบการรับค่า Input รวมทั้งการสร้างสัญญาณ Trigger Interrupt ของ INT0 รวมทั้งการใช้สั่งให้ MCU เข้าทำงานใน Bootloader โดยใช้ร่วมกับสวิตช์ RESET



รูปที่ 2.16 : วงจร สวิตช์ BL(Bootloader)

หมายเลข 16 คือ เป็น MCU ประจำบอร์ด ซึ่งสามารถใช้ได้กับ AVR ขนาด 28ขาได้หลายเบอร์ เช่น ATMEGA8, ATMEGA88, ATMEGA168 และ ATMEGA328

หมายเลข 17 คือ เป็น Crystal Oscillator ค่าความถี่ 19.6608 MHz

หมายเลข 18 คือ เป็น Header สำหรับรองรับการเชื่อมต่อสัญญาณกับบอร์ด ET-MINI ENC28J60 ของ บริษัท อีทีที จำกัด สำหรับใช้พัฒนาโปรแกรมใช้งานกับระบบ Ethernet LAN

หมายเลข 19 คือ เป็น Jumper สำหรับใช้ตัดต่อขาสัญญาณของ PD2(INT0) ที่เชื่อมต่อระหว่าง PD2(INT0) ของบอร์ด ET-BASE AVR EASY88 กับ INT ของบอร์ด ET-MINI ENC28J60 ซึ่ง ถ้าเลือกไว้ด้าน ENA หมายถึง Enable ซึ่งจะเป็นการเชื่อมต่อขา INT จาก ENC28J60 เข้ากับขา PD2 หรือ INT0 ของ ATMEGA88 แต่เมื่อเลือกไว้ทางด้าน DIS จะหมายถึง Disable ซึ่งเป็นการตัดการเชื่อมต่อขา INT ของ ENC28J60 ออกจากขา PD2(INT0) ของ ATMEGA88 ซึ่งตามปรกติควรเลือกไว้ที่ด้าน DIS เสมอ

หมายเลข 20 คือ เป็น ตำแหน่งฐานรองสำหรับชิคบอร์ดทดลองขนาดเล็กของ บริษัท อีทีที จำกัด ที่มีขนาดมาตรฐาน ในขนาด MINI I/O Size ซึ่งผู้ใช้สามารถนำบอร์ดชุด ET-MINI I/O ต่างๆ มาต่อทดลองร่วมกับบอร์ด ET-BASE AVR EASY88 ได้ทันที



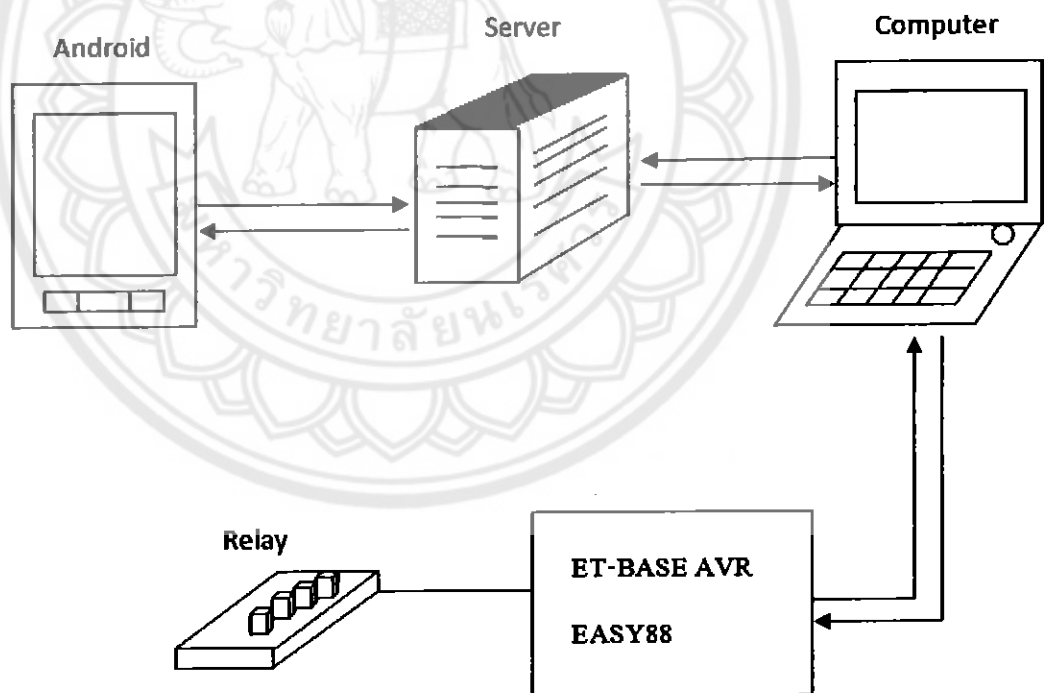
บทที่ 3

วิธีดำเนินงานโครงการ

ในการดำเนินงานสร้างระบบควบคุมไฟฟ้าผ่านแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ ได้แยกระบบการทำงานออกเป็น 3 ส่วนหลักๆ คือ การวิเคราะห์ระบบ การออกแบบและหลักการทำงาน

3.1 การวิเคราะห์ระบบ

เป็นระบบการควบคุมการ ปิด-เปิด ไฟสามารถสั่งได้ทางแอปพลิเคชันบนแอนดรอยด์ทำได้ โดยเมื่อ ต้องมีเครื่องคอมพิวเตอร์หนึ่งเครื่องเพื่อเปิด โปรแกรมที่ติดต่อกับฮาร์ดแวร์ และแอนดรอยด์ก็จะทำการติดต่อเครื่องบริการเพื่อให้โปรแกรมที่ติดต่อกับฮาร์ดแวร์มาติดต่อเครื่องบริการ แล้วอ่านค่าจากเครื่องบริการแล้วส่งการฮาร์ดแวร์ โดยระบบมีสถาปัตยกรรมดังรูป 3.1



รูปที่ 3.1 : สถาปัตยกรรมของระบบ

3.2 การออกแบบ

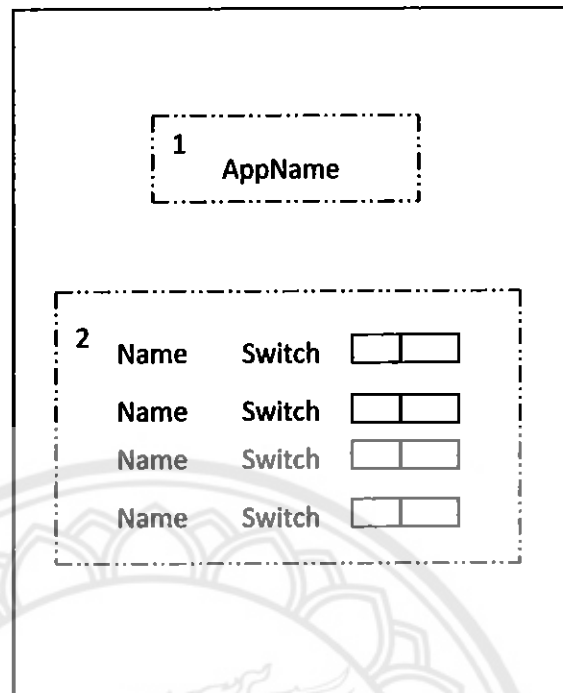
3.2.1 การออกแบบหน้าแอปพลิเคชันบนแอนดรอยด์

การออกแบบหน้าแอปพลิเคชันแอนดรอยด์ ได้แบ่งออกเป็น 2 หน้าการออกแบบ โดยหน้าแรกของแอปพลิเคชันเป็นหน้าล็อกอิน ดังรูปที่ 3.2 และหน้าที่สองเป็นหน้าควบคุมผ่านแอนดรอยด์ดังรูปที่ 3.3 ส่วนหน้าที่เป็น โปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์ดังรูปที่ 3.4 ตามลำดับ

รูปที่ 3.2 : การออกแบบหน้าล็อกอิน

หน้าแรกหรือหน้าล็อกอินแอปพลิเคชันแอนดรอยด์ จะแบ่งเป็น 3 ส่วนหลักๆดังนี้

1. ชื่อแอปพลิเคชัน: เป็นพื้นที่สำหรับใส่ชื่อหรือข้อความสำคัญสำหรับแอปพลิเคชัน
2. ชื่อผู้ใช้และรหัสผ่าน: ช่องกรอก Username และ Password สำหรับเข้าหน้าควบคุมต่อไป
3. ปุ่ม Login และ Cancel: เป็นปุ่ม Login เพื่อเข้าสู่หน้าควบคุม และปุ่ม Cancel เพื่อออกจากแอปพลิเคชัน



รูปที่ 3.3: การออกแบบหน้าควบคุม

หน้าที่สองหรือหน้าควบคุม จะแบ่งเป็น 2 ส่วนหลักๆดังนี้

1. ชื่อแอปพลิเคชัน: เป็นพื้นที่ใช้ใส่ชื่อหรือข้อความสำคัญสำหรับแอปพลิเคชัน
2. ส่วนควบคุม: เป็นส่วนสำหรับสั่งการเปิดหรือปิดไฟโดยใช้สวิตช์ควบคุม

3.2.2 การออกแบบหน้าโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์

The diagram shows a software interface with four main sections:

- 1 ชื่อโปรแกรม**: A text input field for the program name.
- 2 ส่วนแสดงผล**: A large rectangular area for displaying the results of the hardware connection.
- 3 ส่วนส่งข้อมูล**: A section for sending data, containing a text input field, radio buttons for 'Text' and 'Hex', and buttons for 'ส่ง' (Send) and 'ล้าง' (Clear).
- 4 เปิดพอร์ท**: A section for opening the port, containing five buttons labeled 'a', 'b', 'c', 'd', and 'e', and a 'เปิดพอร์ท' (Open Port) button.

รูปที่ 3.4: การออกแบบหน้าโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์

หน้าโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์แบ่งเป็น 4 ส่วนหลักๆ ได้ดังนี้

1. ชื่อ โปรแกรม เป็นส่วนที่ใช้เป็นชื่อ โปรแกรมหรือข้อความที่เกี่ยวข้อง
2. ส่วนแสดงผล เป็นส่วนที่แสดงผลการติดต่อติดต่อกับฮาร์ดแวร์
3. ส่วนส่งข้อมูล เป็นส่วน ใช้งานที่ใช้สำหรับเปิดพอร์ทเพื่อติดต่อกับบอร์ด

ET-BASE AVR EASY 88 เป็นส่วนที่ใช้ส่งข้อมูลไปที่ บอร์ด ET-BASE AVR EASY 88

โดย a, b, c, d, e เป็นส่วนที่ตั้งค่า Com Serial Port

a คือ ส่วนที่สำหรับตั้งค่า Com Port

b คือ ส่วนที่สำหรับตั้งค่า Baud Rate

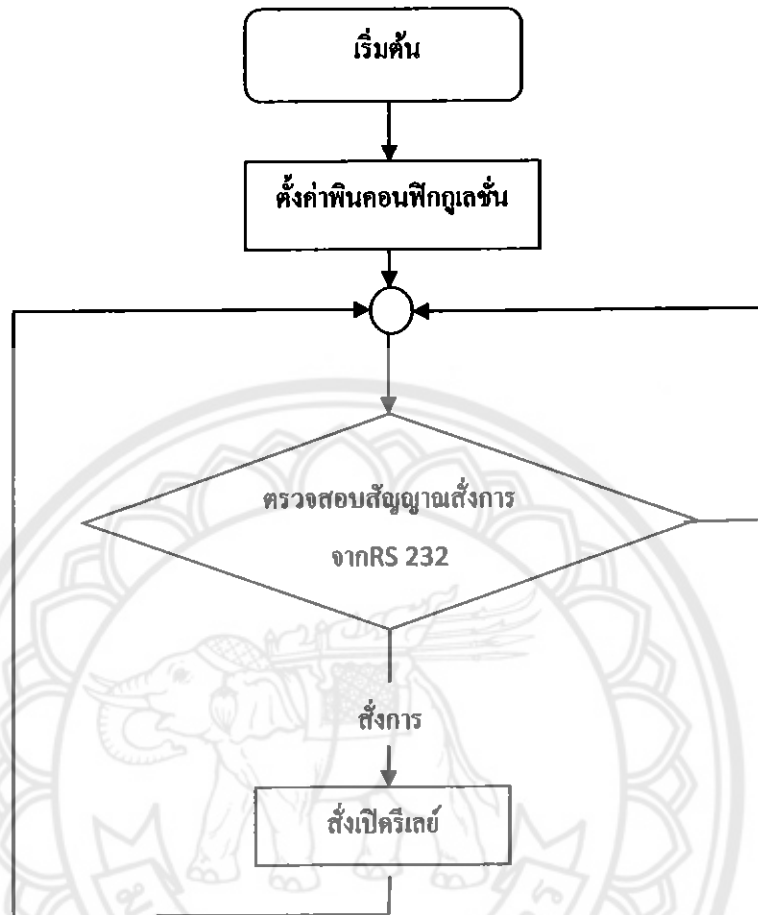
c คือ ส่วนที่สำหรับตั้งค่า Parity

d คือ ส่วนที่สำหรับตั้งค่า Data Bits

e คือ ส่วนที่สำหรับตั้งค่า Stop Bits

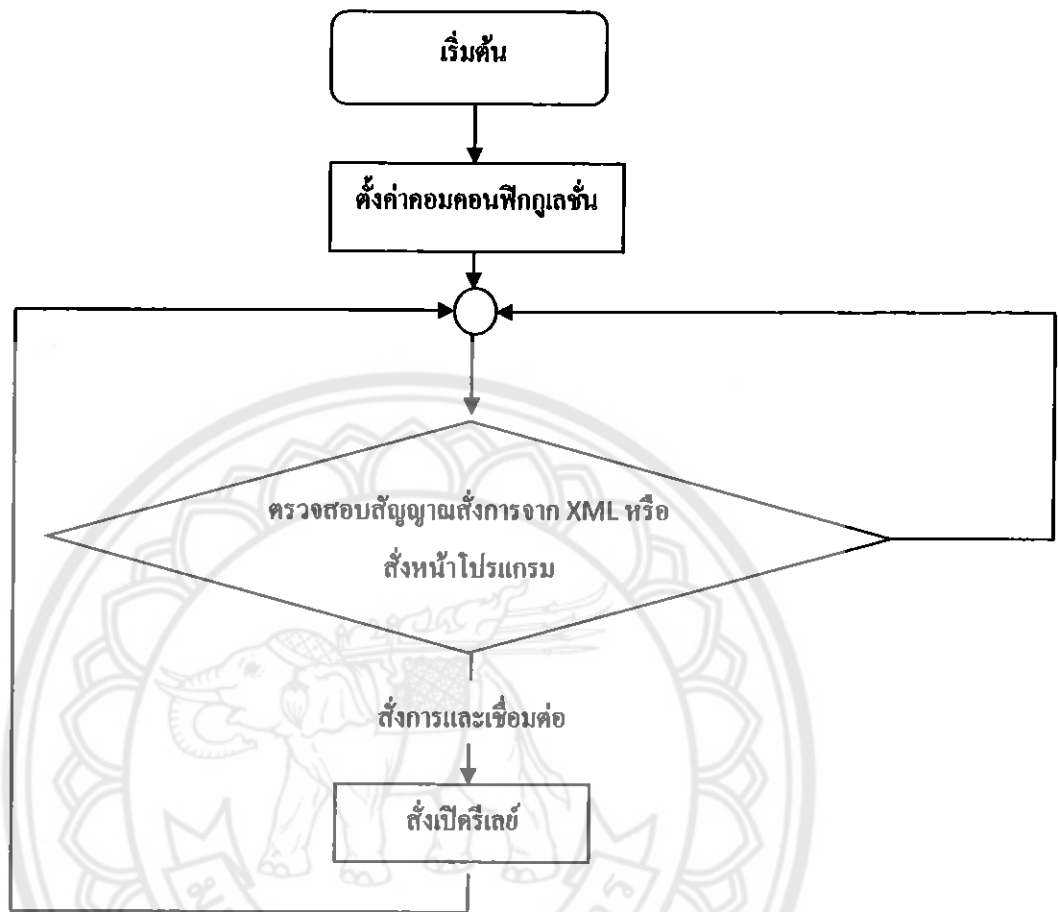
4. ส่วนแสดงสถานะ เป็นส่วนที่ใช้แสดงสถานะของอุปกรณ์ไฟฟ้า

3.2.3 แผนภาพการออกแบบการทำงานของบอร์ด ET-BASE AVR EASY88



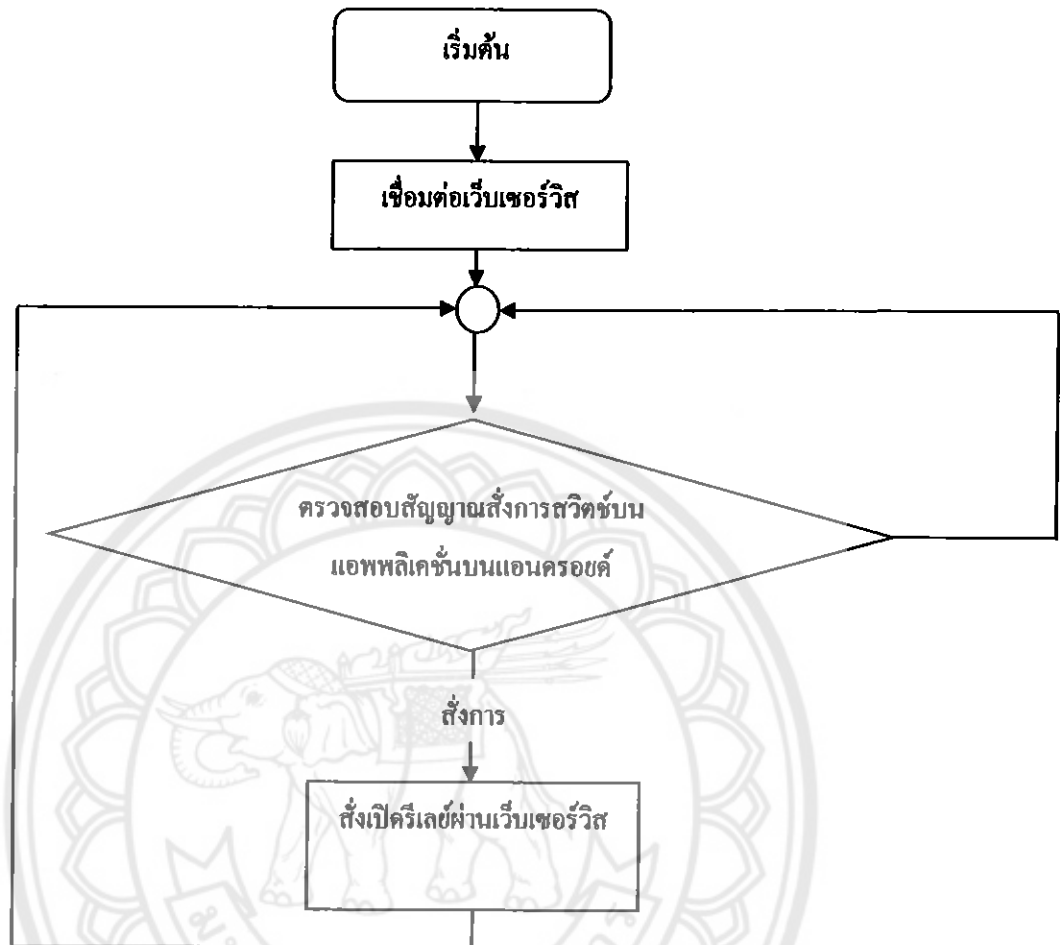
รูปที่ 3.5 : แผนภาพแสดงการทำงานของบอร์ด ET-BASE AVR EASY88

3.2.4 แผนภาพการออกแบบการทำงานของโปรแกรมที่ติดต่อกับฮาร์ดแวร์



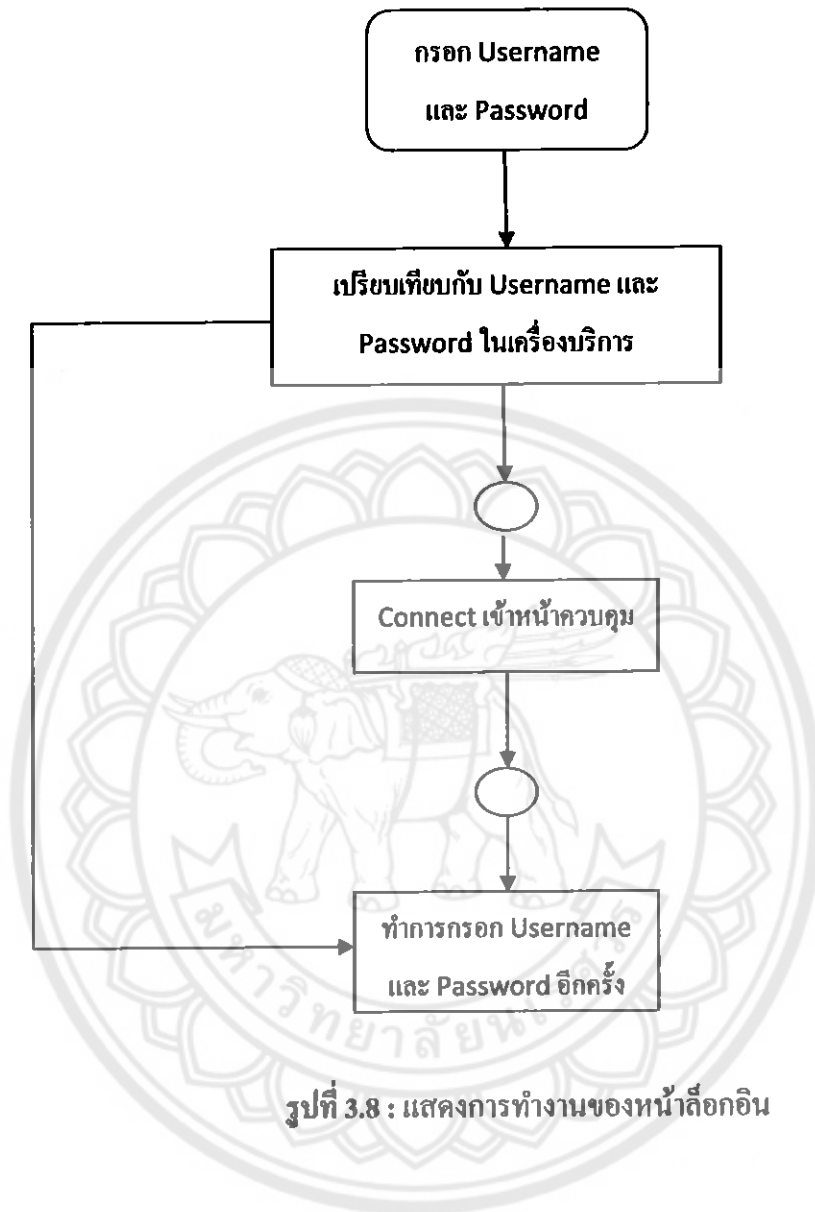
รูปที่ 3.6 : แผนภาพแสดงการทำงานของโปรแกรมที่ติดต่อกับฮาร์ดแวร์

3.2.5 แผนภาพการออกแบบการทำงานของคำสั่งผ่านแอปพลิเคชันบนแอนดรอยด์



รูปที่ 3.7 : แผนภาพแสดงการทำงานในส่วนของแอปพลิเคชันบนแอนดรอยด์

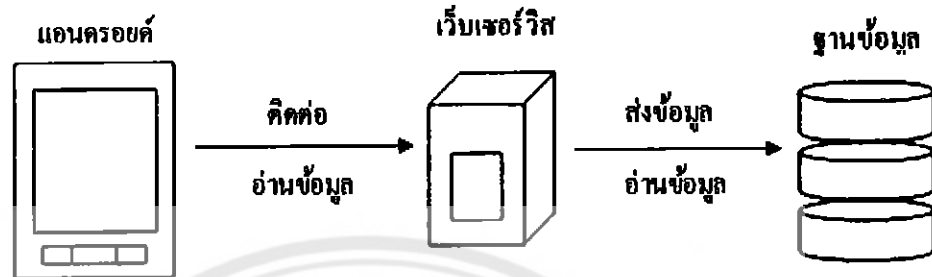
3.2.6 แผนภาพการออกแบบการทำงานของระบบของหน้าล็อกอิน



3.3 หลักการทำงาน

3.3.1 หลักการทำงานของแอปพลิเคชันบนแอนดรอยด์

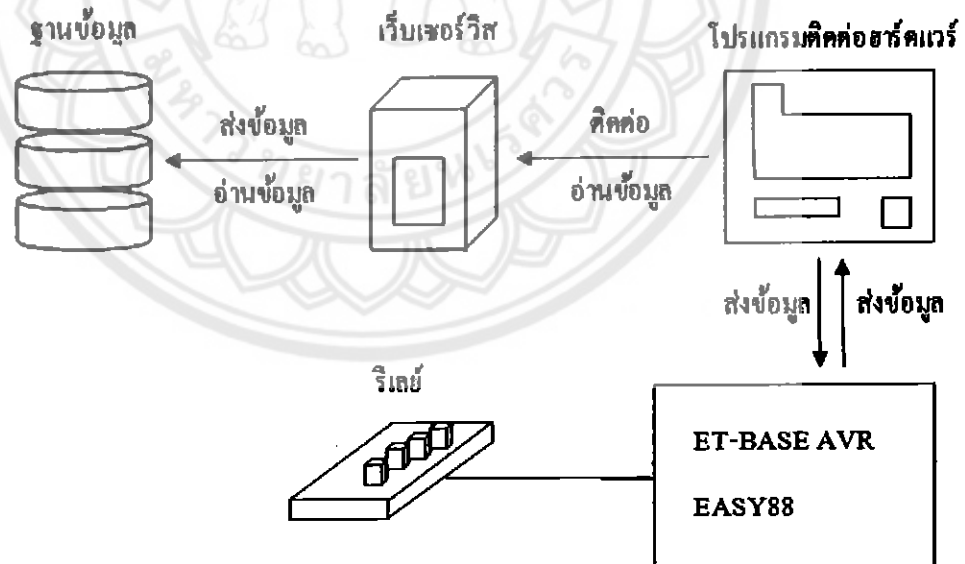
หลักการทำงานของส่วนแอปพลิเคชันบนแอนดรอยด์คือ จะทำการติดต่อเว็บเซิร์ฟเวอร์เพื่อเปลี่ยนแปลงค่าเมื่อเลื่อนสวิตช์ทุกครั้งที่เลื่อนปุ่มปิดหรือเปิด และทำการอ่านค่าจากเว็บเซิร์ฟเวอร์ อยู่นิ่งตลอดเวลาเพื่อทำการเปลี่ยนสถานะแล้วมาแสดงที่แอปพลิเคชันดังรูป 3.8



รูปที่ 3.9 : การทำงานของแอปพลิเคชันบนแอนดรอยด์

3.1.2 หลักการทำงานของโปรแกรมที่ติดต่อกับฮาร์ดแวร์

หลักการทำงานของโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์ คือ ตัวโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์มีหน้าที่ติดต่อกับเว็บเซิร์ฟเวอร์ และอ่านค่าจากเว็บเซิร์ฟเวอร์ก่อนที่จะส่งค่าให้บอร์ด ET-BASE AVR EASY 88 และให้แสดงออกไปยังบอร์ดรีเลย์



รูปที่ 3.10 : การทำงานของโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์

บทที่ 4

ผลการทดลองและวิเคราะห์

เมื่อพัฒนาแอปพลิเคชันและโปรแกรมส่วนที่ติดต่อฮาร์ดแวร์เสร็จสมบูรณ์ ต่อไปจะเป็นการทดสอบแอปพลิเคชันและโปรแกรมส่วนที่ติดต่อฮาร์ดแวร์ ว่าเป็นไปตามเงื่อนไขที่กำหนดไว้หรือไม่ โดยการทดลองโปรแกรมนั้นแบ่งออกเป็น 6 ส่วนใหญ่ดังนี้

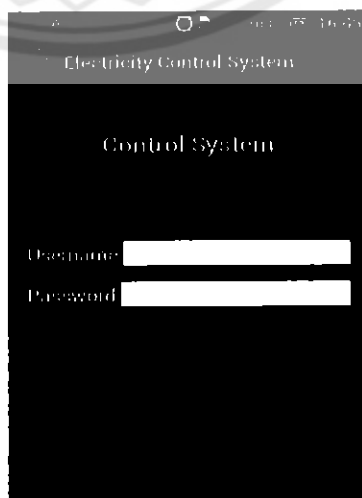
1. ผลการทดลองการล็อกอินเข้าไปยังหน้าควบคุมบนแอนดรอยด์
2. ผลการทดลองจากการสั่งแอนดรอยด์เพื่อติดต่อกับเว็บเซิร์ฟเวอร์
3. ผลการทดลองจากการติดต่อจาก โปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์ไปยังบอร์ด ET-BASE AVR EASY88
4. ผลการทดลองจากส่วนที่ติดต่อกับฮาร์ดแวร์กับเว็บเซิร์ฟเวอร์
5. ผลการทดลองจากการควบคุมจากแอปพลิเคชันบนแอนดรอยด์ไปยังฮาร์ดแวร์
6. ผลการทดลองจากการควบคุมจาก โปรแกรมที่ติดต่อฮาร์ดแวร์เพื่อไปแสดงผลที่แอปพลิเคชันบนแอนดรอยด์

4.1 ผลการทดลองการล็อกอินเข้าไปยังหน้าควบคุมบนแอนดรอยด์

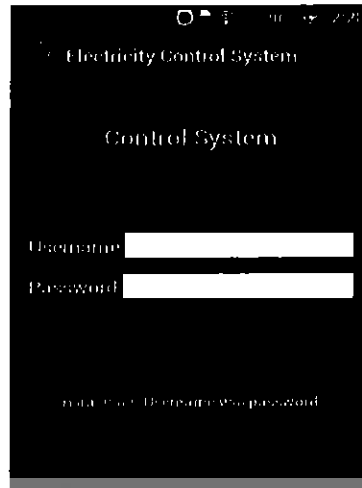
ในส่วนของแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์นั้น ได้ทำขึ้นมา 2 ส่วน โดยส่วนแรกเป็นหน้า Login เพื่อความปลอดภัยในระดับหนึ่ง และอีกส่วนหนึ่งเป็นหน้าตาส่วนของการควบคุม ดังนี้

4.1.1 ส่วนของหน้าล็อกอิน

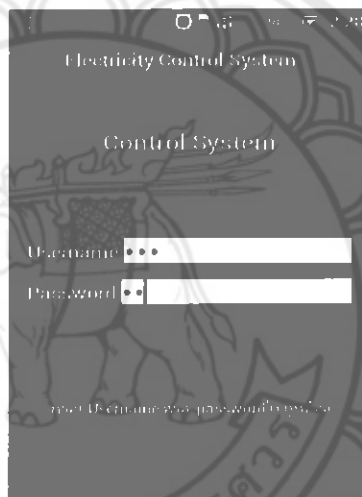
หน้าล็อกอินเมื่อเปิดแอปพลิเคชันที่ได้ทำการสร้างขึ้นมา ดังรูป 4.1 ถ้าไม่ได้กรอก Username และ Password จะขึ้นดังรูป 4.2 ถ้ากรอก Username หรือ Password ผิดจะขึ้นดังรูป 4.3



รูปที่ 4.1 : หน้าแรกของแอปพลิเคชันบนแอนดรอยด์



รูปที่ 4.2 : กรณีกดล็อกอินโดยไม่ได้ Username และ Password



รูปที่ 4.3 : กรณีใส่ Username หรือ Password ผิด

4.1.2 ส่วนของหน้าควบคุม

เมื่อใส่ Username และ Password ถูกต้องก็จะเข้าสู่หน้าควบคุมโดยควบคุมการเปิดปิดไฟ โดยในสวิตช์แทนปุ่มกด โดยมีหน้าตาดังรูป 4.4



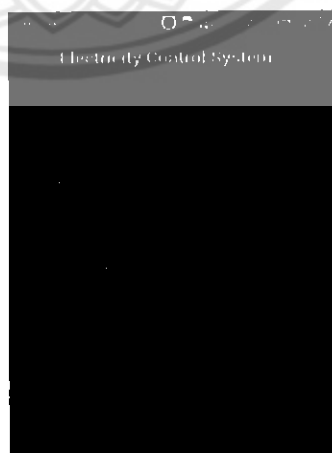
รูปที่ 4.4 : ส่วนของหน้าควบคุม

4.2 ผลการทดลองจากการสั่งแอนดรอยด์เพื่อติดต่อกับเว็บเซอร์วิส

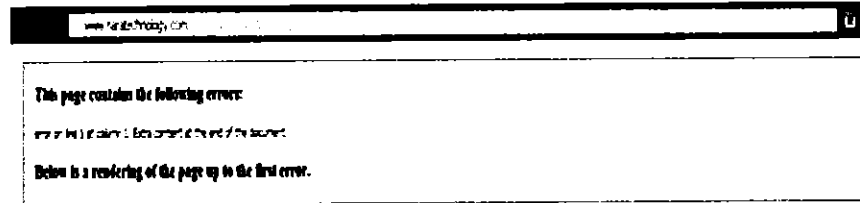
ในส่วนของแอปพลิเคชันบนแอนดรอยด์ เพื่อทำหน้าที่ติดต่อกับเว็บเซอร์วิสและทำหน้าที่เปลี่ยนแปลงข้อมูลในฐานข้อมูลแล้วให้ไฟล์ XML อ่านค่าจากฐานข้อมูล(มีค่าเป็น 0 และ 1)

4.2.1 เมื่อกดเปิดที่สวิตช์

เมื่อทำการสั่ง โดยกดเปิดสวิตช์แอนดรอยด์ดังรูป 4.5 ต่อมาแอนดรอยด์ทำการติดต่อเว็บเซอร์วิส โดยแสดงให้เห็นในผังรูปแบบเว็บไซต์ดังรูป 4.6 เว็บเซอร์วิสเข้าไปเปลี่ยนแปลงข้อมูลในฐานข้อมูล V1 เป็น 1 ดังรูป 4.7 จากนั้นเว็บเซอร์วิสที่เป็น XML อ่านข้อมูลจากฐานข้อมูลดังรูป 4.8



รูปที่ 4.5 : เมื่อทำการกดเปิดสวิตช์



รูปที่ 4.6 : แอนครอยด์ทำการติดต่อเว็บเซอร์วิส

Show : Start row: 0 Number of rows: 30 Headers every 100 rows

Options
 Edit Copy Delete 00001; 1; 0; 0; 0
 Check All / Uncheck All With selected: Change Delete Export

รูปที่ 4.7: เว็บเซอร์วิสเปลี่ยนค่าในฐานข้อมูล

This XML file does not appear to

```

  <NewDataSet>
    <tbl_status>
      <id>00001 </id>
      <r1>1</r1>
      <r2>0</r2>
      <r3>0</r3>
      <r4>0</r4>
    </tbl_status>
  </NewDataSet>
  
```

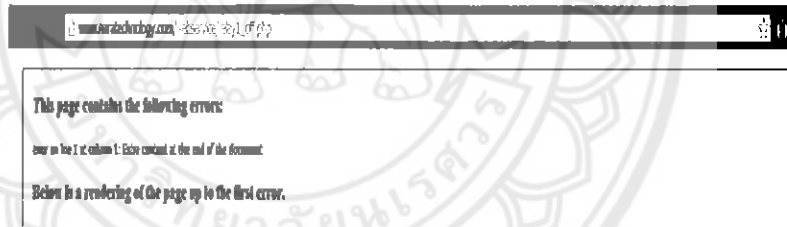
รูปที่ 4.8 : XML อ่านข้อมูลจากฐานข้อมูล

4.2.2 เมื่อกดปิดที่สวิตช์

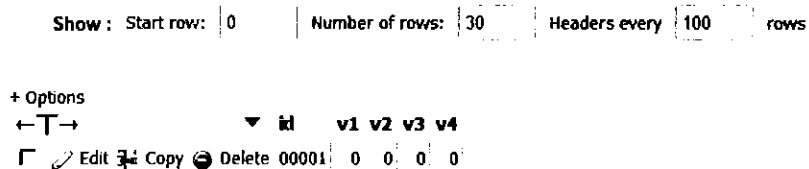
เมื่อทำการสั่งโดยกดปิดสวิตช์ดังรูป 4.9 ต่อมาแอนครอยด์ทำการติดต่อเว็บเซอร์วิส โดยแสดงรูปแบบเว็บไซต์ดังรูป 4.10 เว็บเซอร์วิสเข้าไปเปลี่ยนแปลงข้อมูลในฐานข้อมูล V1 เป็น 0 ดังรูป 4.11 จากนั้นเว็บเซอร์วิสที่เป็น XML อ่านข้อมูลจากฐานข้อมูลดังรูป 4.12



รูปที่ 4.9 : เมื่อกดปิดสวิตช์



รูปที่ 4.10 : แอนครอยด์ทำการติดต่อเว็บเซอร์วิส



รูปที่ 4.11: เว็บเซอร์วิสเปลี่ยนค่าในฐานข้อมูล

This XML file does not appear

```

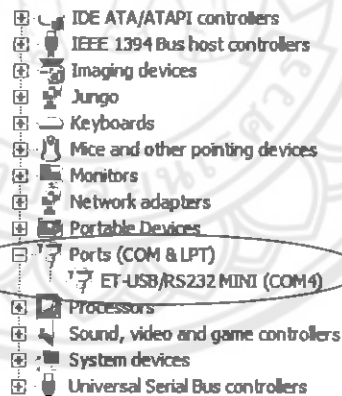
▼ <NewDataSet>
  ▼ <tbl_status>
    <id>00001</id>
    <r1>0</r1>
    <r2>0</r2>
    <r3>0</r3>
    <r4>0</r4>
  </tbl_status>
</NewDataSet>

```

รูปที่ 4.12 : XML อ่านข้อมูลจากฐานข้อมูล

4.3 ผลการทดลองจากการติดต่อจากโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์ไปยังบอร์ด ET-BASE AVR EASY88

ในการควบคุมฮาร์ดแวร์นั้น การส่งข้อมูลเพื่อติดต่อกับบอร์ด ET-BASE AVR EASY88 ต้องทำการเชื่อมต่อผ่านสาย RS232 โดยสามารถเช็คว่าสาย RS232 ใช้งานได้หรือไม่นั้นบน Windows สามารถตรวจสอบได้โดยเข้าไปที่ Device Manager ถ้าสาย RS232 ใช้งานได้ก็จะขึ้นตรง Ports (COM & LPT) ดังรูป 4.13

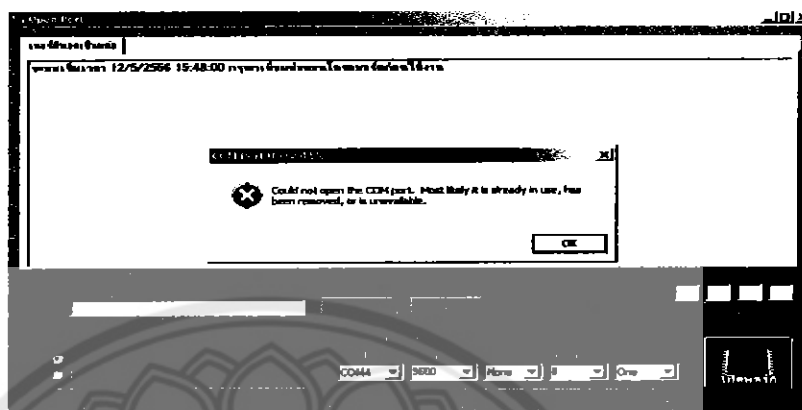


รูปที่ 4.13 : การตรวจสอบสาย RS232

ในการทดลองว่า โปรแกรมที่ติดต่อกับฮาร์ดแวร์ที่พัฒนาขึ้นมาสามารถใช้งานได้ มีวิธีการตรวจสอบเป็นขั้นตอนดังนี้

4.3.1 ทดสอบการติดต่อกับบอร์ด ET-BASE AVR EASY88

ในการทดสอบการติดต่อกับบอร์ด ET-BASE AVR EASY88 โดยคลิกที่ปุ่มเปิดพอร์ตในโปรแกรมแต่ถ้าในกรณีไม่สามารถติดต่อกับบอร์ดได้ก็จะขึ้นดังรูป 4.14

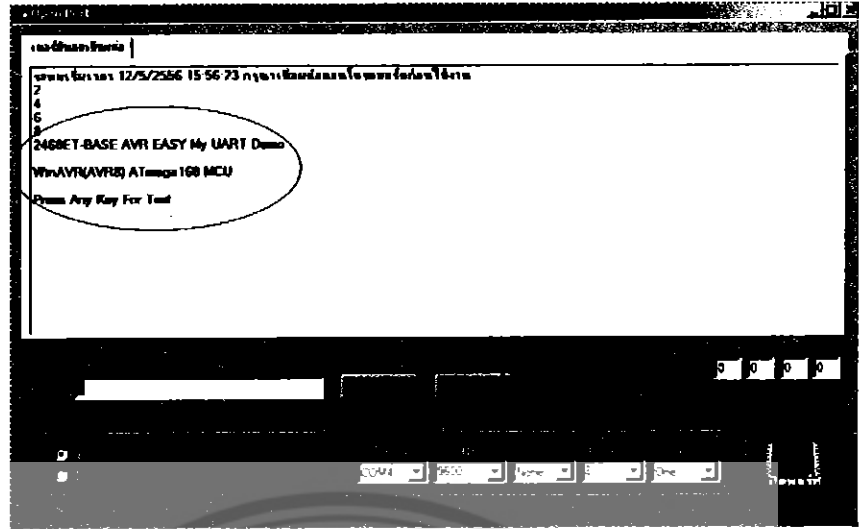


รูปที่ 4.14 : กรณีที่ไม่สามารถเชื่อมต่อบอร์ดได้

ในกรณีที่สามารถเชื่อมต่อได้ก็จะแสดงข้อความออกมาดังรูป 4.15 คือ 2, 4, 6, 8 เพื่อสั่งให้ไฟดับ หรือทดสอบการเชื่อมต่อได้อีกวิธีคือการกดที่ปุ่ม Reset ของบอร์ด ET-BASE AVR EASY88 ส่งข้อความ Echo กลับมา ดังรูป 4.16



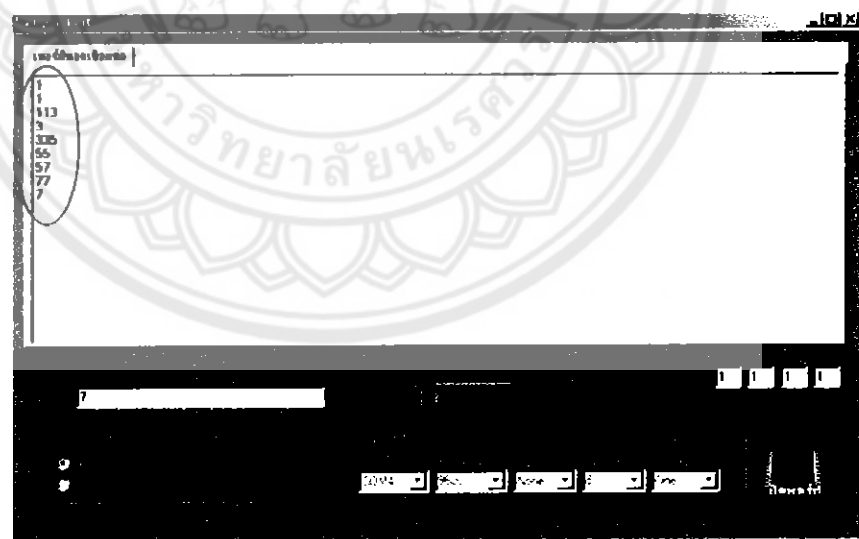
รูปที่ 4.15 : กรณีที่เชื่อมต่อบอร์ดได้



รูปที่ 4.16 : กรณีที่กดปุ่ม Reset บนบอร์ด

4.3.2 ทดสอบการสั่งเปิด-ปิดไฟผ่านโปรแกรม

ในการทดสอบการเปิด-ปิดไฟผ่านโปรแกรมได้ตั้งค่าไว้ คือ ทำการพิมพ์ 1, 3, 5, 7 แล้วกดส่ง โปรแกรมก็จะแสดงสถานะดังรูปที่ 4.17 และทำให้ไฟที่บอร์ดรีเลย์ดวงที่ 1, 2, 3, 4 ติดดังรูปที่ 4.18 แต่ถ้าพิมพ์เลข 2, 4, 6, 8 แล้วกดส่ง โปรแกรมก็จะแสดงสถานะดังรูป 4.19 และทำให้ไฟที่บอร์ด รีเลย์ดวงที่ 1, 2, 3, 4 คับดังรูปที่ 4.20



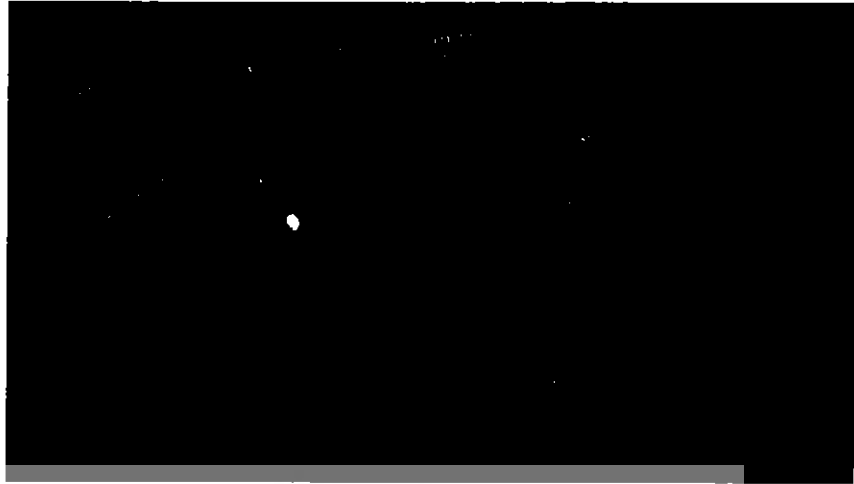
รูปที่ 4.17 : ส่งค่าตัวเลข 1, 3, 5, 7 และแสดงสถานะที่โปรแกรม



รูปที่ 4.18 : ผลจากการส่งค่า 1, 3, 5, 7 ที่โปรแกรม



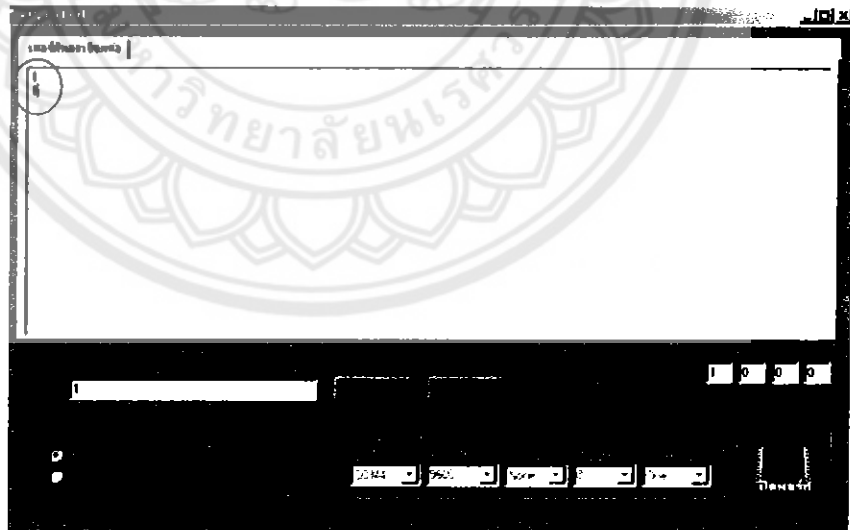
รูปที่ 4.19 : ส่งค่าตัวเลข 2, 4, 6, 8 และแสดงสถานะที่โปรแกรม



รูปที่ 4.20 : ผลจากการส่งค่า 2, 4, 6, 8 ที่โปรแกรม

4.4 ผลการทดลองจากส่วนที่ติดต่อกับฮาร์ดแวร์กับเว็บเซอร์วิส

ในขั้นตอนการทดสอบนั้นพิมพ์ค่า 1 แล้วกดส่งที่โปรแกรมดังรูป 4.21 แล้วโปรแกรมทำการไปติดต่อกับเว็บเซอร์วิสและทำให้ฐานข้อมูลที่ V1 เปลี่ยนจาก 0 เป็น 1 ดังรูป 4.22 ไฟล์เว็บเซอร์วิสที่เป็น XML จึงอ่านค่าจากฐานข้อมูลแล้วเปลี่ยนค่าดังรูป 4.23 ไฟล์เว็บเซอร์วิส `getdata_relay.php` ก็ทำการอ่านจากฐานข้อมูลแล้วเปลี่ยนค่าดังรูป 4.24



รูปที่ 4.21 : ส่งค่า 1 เข้าไปเพื่อไปติดต่อกับเว็บเซอร์วิส

Show : Start row: 0 | Number of rows: 30 | Headers every 100 | rows

+ Options

← T →

▼ id v1 v2 v3 v4

⌂ Edit Copy Delete 00001 | 1 | 0 | 0 | 0

รูปที่ 4.22 : ฐานข้อมูลที่ V1 เปลี่ยนจาก 0 เป็น 1

This XML file does not appear

```

▼ <NewDataSet>
  ▼ <tbl_status>
    <id>00001</id>
    <r1>1</r1>
    <r2>0</r2>
    <r3>0</r3>
    <r4>0</r4>
  </tbl_status>
</NewDataSet>

```

รูปที่ 4.23 : XML อ่านค่าจากฐานข้อมูล

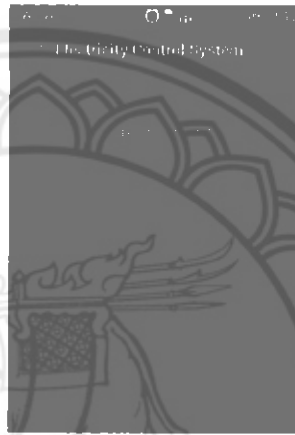
www.naratechnology.com/webservice/getdata_relay.php

```
{"id":"00001","v1":"1","v2":"0","v3":"0","v4":"0"}
```

รูปที่ 4.24 : getdata_relay.php อ่านค่าจากฐานข้อมูล

4.5 ผลการทดลองจากการควบคุมจากแอปพลิเคชันบนแอนดรอยด์ไปยังฮาร์ดแวร์

ในการเชื่อมต่อกันระหว่างแอปพลิเคชันบนแอนดรอยด์กับฮาร์ดแวร์ทดสอบการกดเปิดที่สวิทช์ บนแอปพลิเคชันดังรูป 4.25 ทำให้แอนดรอยด์ติดต่อกับทางเว็บเซอร์วิสในรูปแบบของเว็บดังรูป 2.26 ดังรูป เว็บเซอร์วิสจึงเข้าไปเปลี่ยนข้อมูลในฐานข้อมูลที่ V1 จาก 0 เป็น 1 ดังรูป 4.27 ไฟล์เว็บเซอร์วิสที่เป็น XML จึงเปลี่ยนข้อมูลดังรูป 4.28 โปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์ทำการอ่านค่าจากไฟล์ XML แล้วมาแสดงในโปรแกรมก่อนที่จะส่งให้บอร์ด ET-BASE AVR EASY88 ดังรูป 4.29 เมื่อ ET-BASE AVR EASY88 รับค่าจาก โปรแกรมจึงตรวจสอบค่าแล้วจึงสั่งไฟที่บอร์ดรีเลย์ดวงที่ 1 ดังรูป 4.30



รูปที่ 4.25 : กดเปิดที่สวิทช์บนแอปพลิเคชัน

www.naratechnology.com/webservice/relay1_on.php

This page contains the following errors:

error on line 1 at column 1: Extra content at the end of the document

Below is a rendering of the page up to the first error.

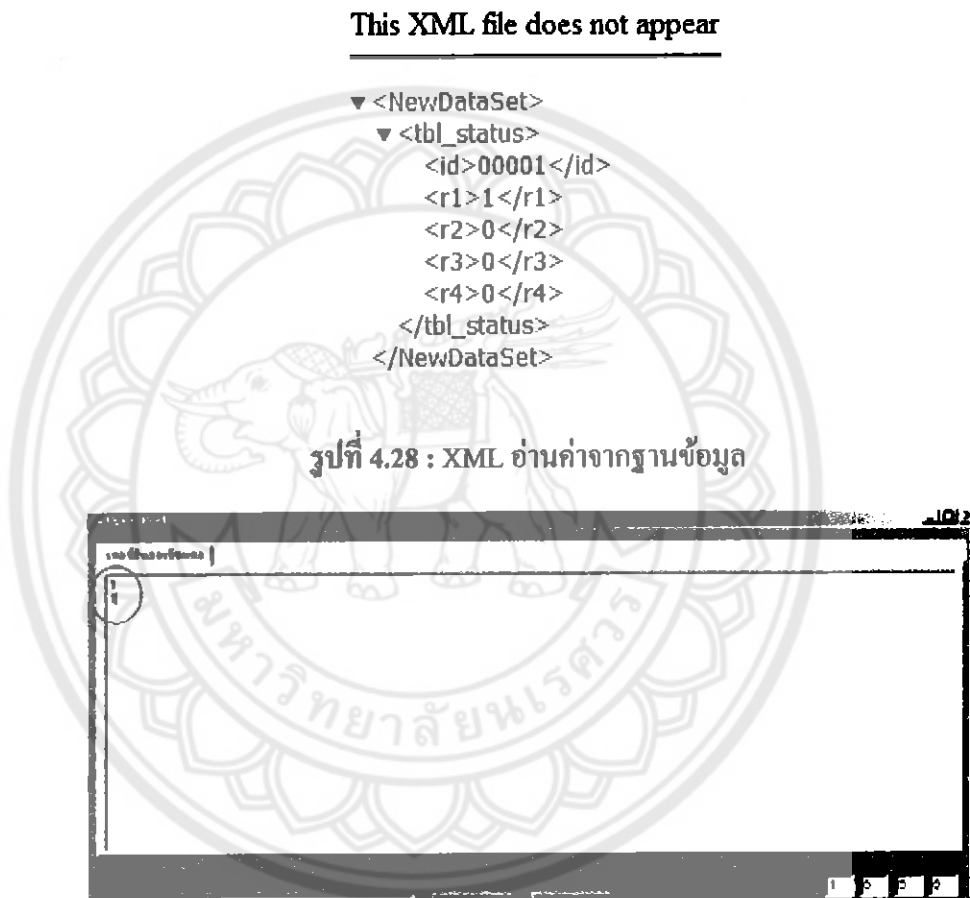
รูปที่ 4.26 : แอนดรอยด์ติดต่อกับทางเว็บเซอร์วิส

Show : Start row: 0 | Number of rows: 30 | Headers every 100 | rows

+ Options

	id	v1	v2	v3	v4
← T →	▼				
⌂ Edit Copy Delete	00001	1	0	0	0

รูปที่ 4.27 : เว็บเซอวิสเซิลเปลี่ยนข้อมูลในฐานข้อมูลที่ V1 จาก 0 เป็น 1



รูปที่ 4.28 : XML อ่านค่าจากฐานข้อมูล



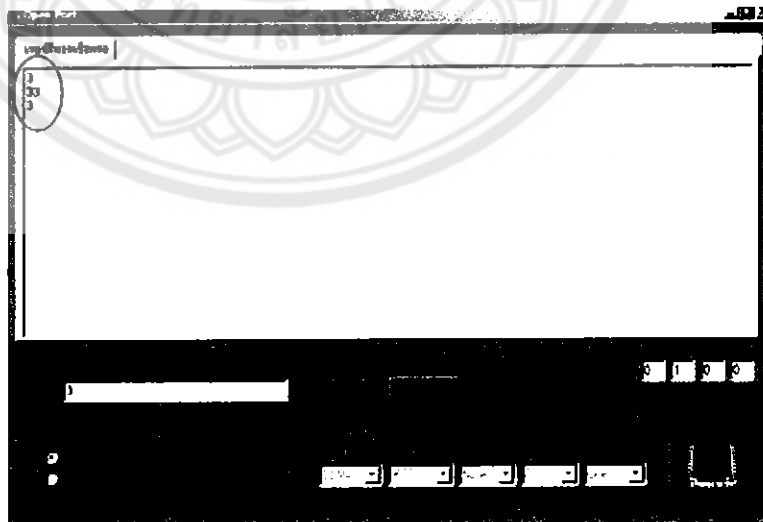
รูปที่ 4.29 : โปรแกรมอ่านค่าจากเว็บเซอวิสเซิลที่เป็น XML



รูปที่ 4.30 : ตั้งรีเลย์ดวงที่ 1 ไฟติด

4.6 ผลการทดลองจากการควบคุมจากโปรแกรมที่ติดต่อฮาร์ดแวร์เพื่อไปแสดงผลที่ แอปพลิเคชันบนแอนดรอยด์

ในการทดลองการควบคุมจาก โปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์ สามารถแสดงผลบน แอปพลิเคชันบนแอนดรอยด์ได้ โดยทดลองพิมพ์ 3 แล้วกดส่งที่โปรแกรมดังรูป 4.31 แล้ว โปรแกรมจะเข้าไปติดต่อเว็บเซอร์วิส โดยจะแสดงในเห็นดังรูป 4.32 แล้วเว็บเซอร์วิสก็เข้าไปเปลี่ยน ข้อมูลจากฐานข้อมูลที่ V2 จาก 0 เป็น 1 ดังรูป 4.33 ไฟล์ `getdata_relay.php` ก็อ่านค่าจากฐานข้อมูล แล้วเปลี่ยนค่า v2 จาก 0 เป็น 1 ดังรูป 4.34 จากนั้นแอปพลิเคชันที่ทำการอ่านค่าจากไฟล์ `getdata_relay.php` อยู่ทุกๆ 1 วินาทีอยู่แล้วนั้นก็เปลี่ยนสวิตซ์จากปิดเป็นเปิดดังรูป 4.35



รูปที่ 4.31 : พิมพ์ 3 แล้วกดส่งที่โปรแกรม

www.naratechnology.com/webservice/relay2_on.php

This page contains the following errors:

error on line 1 at column 1: Extra content at the end of the document

Below is a rendering of the page up to the first error.

รูปที่ 4.32 : โปรแกรมมาคิดต่อเว็บเซอร์วิส



รูปที่ 4.33: เว็บเซอร์วิสเปลี่ยนข้อมูลจากฐานข้อมูลที่ V2 จาก 0 เป็น 1

www.naratechnology.com/webservice/getdata_relay.php

```
{"id":"00001","v1":"0","v2":"1","v3":"0","v4":"0"}
```

รูปที่ 4.34 : getdata_relay.php เปลี่ยนข้อมูลจากฐานข้อมูลที่ V2 จาก 0 เป็น 1



รูปที่ 4.35 : แอนดรอยด์ทำการเปลี่ยนสวิตซ์จากปิดเป็นเปิด



บทที่ 5

บทสรุปและข้อเสนอแนะ

โครงการนี้มีวัตถุประสงค์เพื่อที่จะทำการจำลองการควบคุมและการจัดการระบบไฟฟ้าผ่านทางอุปกรณ์อัจฉริยะที่ใช้ระบบปฏิบัติการแอนดรอยด์ เพื่อที่จะสามารถตรวจสอบและสั่งควบคุมไฟฟ้าบ้านได้ถึงไม่ได้อยู่ในตู้บ้าน ภาษาที่นำมาใช้ในการพัฒนาในส่วนของการติดต่อกับฮาร์ดแวร์คือภาษา C# เนื่องจากเป็นภาษาที่ง่ายต่อการสร้างหน้าของโปรแกรมและมีคู่มือการใช้มากมายในปัจจุบัน และใช้ PHP และ XML ในการสื่อสารข้อมูลเพราะการอ่านค่าใน XML นั้นสามารถทำได้ง่าย

การออกแบบในโครงการนั้นได้ออกแบบระบบโดยให้ควบคุมไฟผ่านทางแอปพลิเคชันบนแอนดรอยด์ โดยมีเซิร์ฟเวอร์คอยให้เปลี่ยนแปลงข้อมูลเพื่อที่จะให้คอมพิวเตอร์ที่มีโปรแกรมที่ติดต่อกับฮาร์ดแวร์อ่านข้อมูลจากเครื่องบริการแล้วส่งค่าให้กับบอร์ด ET-BASE AVR EASY88 ก่อนที่จะสั่งให้บอร์ดรีเลย์ทำงาน

5.1 สรุปผล

จากการจัดทำโครงการระบบควบคุมไฟฟ้าภายในบ้านผ่านแอปพลิเคชันบนแอนดรอยด์นั้นผลที่ได้นั้นสามารถเขียนโปรแกรมที่สามารถสั่งเปิด-ปิด ไฟบ้าน ผ่านทางแอปพลิเคชันบนแอนดรอยด์ได้จริง ตัวโปรแกรมที่พัฒนาขึ้นมาเพื่อทำการสั่งเปิดปิดไฟฟ้าภายในบ้านนั้นมี 2 ส่วนดังนี้

5.1.1 ผลการทดลองจากการสั่งผ่านแอปพลิเคชันบนแอนดรอยด์

ตารางที่ 5.1 : แสดงการทำงานเมื่อกดที่สวิตช์เปิด

สวิตช์	เชื่อมต่อ	ค่าในไฟล์ XML	โปรแกรมอ่านค่าจากไฟล์ XML แดว	สถานะที่บอร์ดรีเลย์
LED 1 เปิด	relay1_on.php	1	r1	ไฟติด
LED 2 เปิด	relay2_on.php	1	r2	ไฟติด
LED 3 เปิด	relay3_on.php	1	r3	ไฟติด
LED 4 เปิด	relay4_on.php	1	r4	ไฟติด

ตารางที่ 5.2 : แสดงการทำงานเมื่อกดที่สวิตช์ปิด

สวิตช์	เชื่อมต่อ	ค่าในไฟล์ XML	โปรแกรมอ่านค่า จากไฟล์ XML แถว	สถานะที่บอร์ดรีเลย์
LED 1 ปิด	relay1_off.php	0	r1	ไฟดับ
LED 2 ปิด	relay2_off.php	0	r2	ไฟดับ
LED 3 ปิด	relay3_off.php	0	r3	ไฟดับ
LED 4 ปิด	relay4_off.php	0	r4	ไฟดับ

5.1.2 ผลการทดลองจากการสั่งผ่านโปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์

ตารางที่ 5.3 : แสดงการทำงานเมื่อส่งค่าผ่าน โปรแกรมส่วนที่ติดต่อกับฮาร์ดแวร์

ค่าที่ส่งไปในโปรแกรม	เชื่อมต่อ	ค่าในไฟล์ getdata_relay	สถานะในแอปพลิเคชัน	สถานะที่บอร์ดรีเลย์
1	relay1_on.php	1	สวิตช์ LED 1 เลื่อนเปิด	ไฟติด
2	relay1_off.php	0	สวิตช์ LED 1 เลื่อนปิด	ไฟดับ
3	relay2_on.php	1	สวิตช์ LED 2 เลื่อนเปิด	ไฟติด
4	relay2_off.php	0	สวิตช์ LED 2 เลื่อนปิด	ไฟดับ
5	relay3_off.php	1	สวิตช์ LED 3 เลื่อนเปิด	ไฟติด
6	relay3_off.php	0	สวิตช์ LED 3 เลื่อนปิด	ไฟดับ
7	relay4_off.php	1	สวิตช์ LED 4 เลื่อนเปิด	ไฟติด
8	relay4_off.php	0	สวิตช์ LED 4 เลื่อนปิด	ไฟดับ

5.2 ปัญหาและอุปสรรคที่พบ

ตารางที่ 5.4 : แสดงปัญหาที่พบและวิธีการแก้ไข

ปัญหา	วิธีการแก้ไข
1.เกิด bug ที่ปุ่มสวิทช์ใน Eclipse 64 bit	ลงโปรแกรม Eclipse แบบ 32 แทน
2.แอปพลิเคชันค้างเมื่อทำการอ่านข้อมูลที่เซิร์ฟเวอร์	ตั้งค่าเวลาให้อ่านข้อมูลจากเซิร์ฟเวอร์ให้มากขึ้น
3.เกิด ERROR เวลาปรับแต่ง Activity ในแอนดรอยด์ 4.0.3	ตั้งค่าในไฟล์ AndroidManifest โดยเปลี่ยน <uses-sdk android:minSdkVersion="8" />
4.แอปพลิเคชันที่พัฒนาไม่สามารถเชื่อมต่ออินเทอร์เน็ตได้	เพิ่มโค้ด <uses-permission android:name="android.permission.INTERNET"/> ในไฟล์ AndroidManifest
5.ไม่สามารถเบิร์นโปรแกรมลงบอร์ด ET-BASE AVR EASY 88 ผ่านทาง RS232 ได้	ใช้ ET-AVR PROG MINI ในการเบิร์นแทนการเบิร์นผ่านทาง RS 232

5.3 แนวทางการพัฒนาต่อ

- พัฒนาโดยให้บอร์ด ET-BASE AVR EASY88 เป็น Client โดยสามารถอ่านข้อมูลจากเครื่องบริการได้โดยตรง
- ทำอุปกรณ์ฮาร์ดแวร์ที่เชื่อมต่อกับบอร์ด ET-BASE AVR EASY88 ให้สามารถเชื่อมต่อกับอุปกรณ์ที่ใช้ไฟ 220 v จริงได้

5.4 ข้อเสนอแนะ

- สามารถล็อกอินผ่านเครื่องบริการได้เพื่อในกรณีที่มีผู้ใช้หลายคน
- พัฒนาให้สามารถตรวจสอบได้ในกรณีที่ส่งควบคุมแล้วอุปกรณ์นั้นเสีย
- สามารถตั้งเวลาควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้าได้

5.5 ความรู้ที่จำเป็นต่อการพัฒนา

- การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์
- การเขียนโปรแกรมบนระบบปฏิบัติการแอนดรอยด์
- การเขียนโปรแกรมภาษา C# เพื่อติดต่อกับฮาร์ดแวร์
- การออกแบบฐานข้อมูล
- การเข้ารหัสและการถอดรหัส

เอกสารอ้างอิง

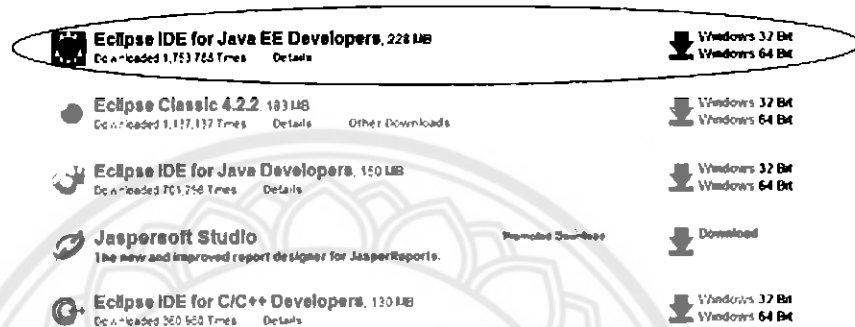
- [1] Code ฟังก์ชัน request web service. <http://stackoverflow.com/questions/16481567/cannot-change-android-switch-style-when-using-holo-everywhere>
- [2] Code ฟังก์ชัน Switch. <http://stackoverflow.com/questions/16481567/cannot-change-android-switch-style-when-using-holo-everywhere>
- [3] ประวัติของภาษา java. <http://javaprogram.freetzi.com/webpages/aboutjava.php>
- [4] คู่มือของบอร์ดไมโครคอนโทรลเลอร์. <http://www.etteam.com/>
- [5] ประจัน พลังตันติสุข.การประยุกต์ใช้งานภาษา C สำหรับไมโครคอนโทรลเลอร์ AVR .
<http://www.appsofttech.com>



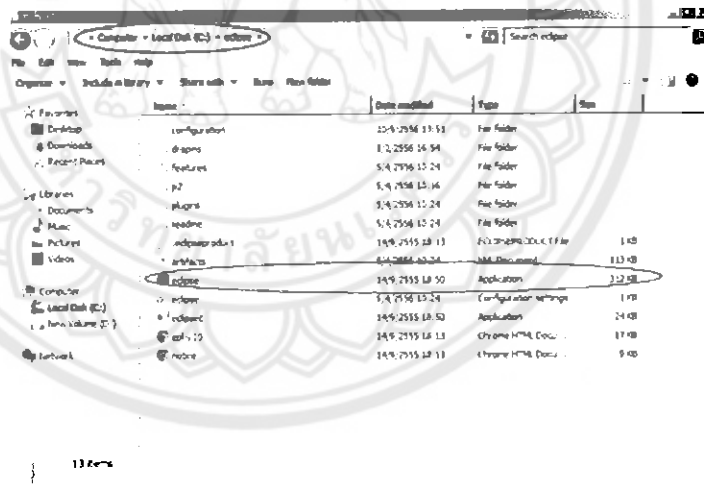
ภาคผนวก ก

การติดตั้งโปรแกรม Eclipse

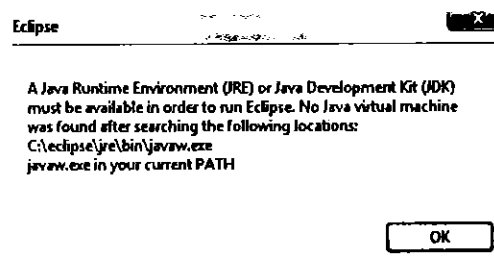
- 1 เข้าไปดาวน์โหลดโปรแกรมที่ <http://www.eclipse.org/downloads/> โดยเลือกดาวน์โหลด
 ด้งรูปควรเลือกในระบบ 32 bit



- 2 เมื่อทำการดาวน์โหลดเสร็จสิ้นให้ทำการแตกไฟล์ไว้ที่ C:\eclipse จากนั้นให้เปิด
 โปรแกรม Eclipse โดยการคลิกที่ eclipse.exe



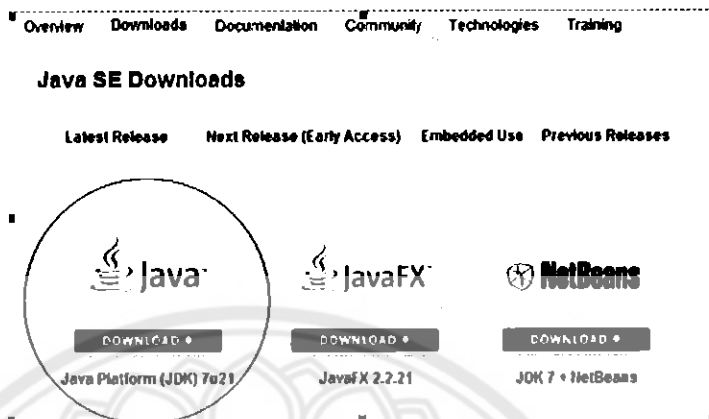
- 3 ถ้าในกรณีที่ขึ้น Error ดังภาพแสดงว่าไม่ได้ทำการติดตั้ง Java Development JDK



4 เข้าไปดาวน์โหลด Java Development JDK ได้ที่

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

โดยเลือกไปที่ Java Platform (JDK) 7u21



5 แล้วเลือกดาวน์โหลดที่เป็น JDK เลือกให้ถูกต้องกับ OS ที่ต้องการติดตั้ง

Java SE Development Kit 7u21
 You must accept the Oracle Binary Code License Agreement for Java SE to download this software.
 Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.

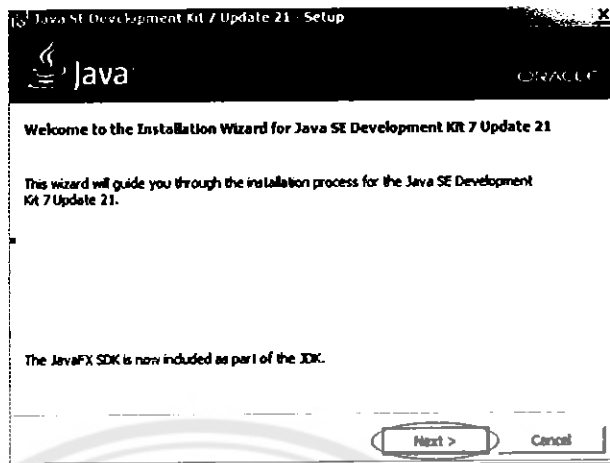
Product / File Description	File Size	Download
Linux ARM v5.7 Soft Float ABI	65.09 MB	jdk-7u21-linux-arm-sfp.tar.gz
Linux x86	80.35 MB	jdk-7u21-linux-i586.rpm
Linux x86	93.05 MB	jdk-7u21-linux-i586.tar.gz
Linux x64	81.43 MB	jdk-7u21-linux-x64.rpm
Linux x64	91.81 MB	jdk-7u21-linux-x64.tar.gz
Mac OS X x64	144.18 MB	jdk-7u21-macosx-x64.dmg
Solaris x86 (SvR4 package)	135.84 MB	jdk-7u21-solaris-i586.tar.Z
Solaris x86	92.08 MB	jdk-7u21-solaris-i586.tar.gz
Solaris x64 (SvR4 package)	22.67 MB	jdk-7u21-solaris-x64.tar.Z
Solaris x64	15.02 MB	jdk-7u21-solaris-x64.tar.gz
Solaris SPARC (SvR4 package)	136.09 MB	jdk-7u21-solaris-sparc.tar.Z
Solaris SPARC	95.44 MB	jdk-7u21-solaris-sparc.tar.gz
Solaris SPARC 64-bit (SvR4 package)	22.97 MB	jdk-7u21-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	17.58 MB	jdk-7u21-solaris-sparcv9.tar.gz
Windows x86	88.98 MB	jdk-7u21-windows-i586.exe
Windows x64	90.57 MB	jdk-7u21-windows-x64.exe

6 ทำการดับเบิลคลิกเพื่อติดตั้ง

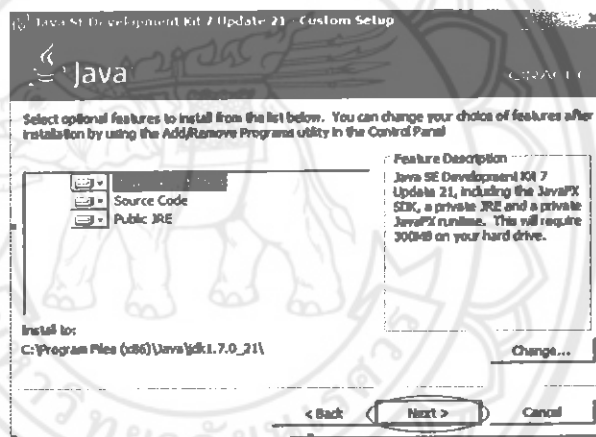


jdk-7u21-windows-i586

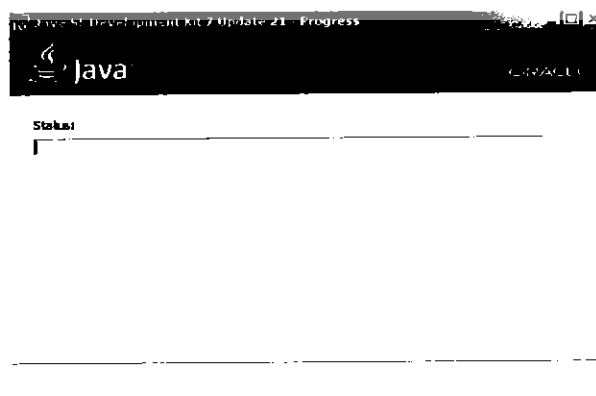
7 คลิกเลือกที่ Next



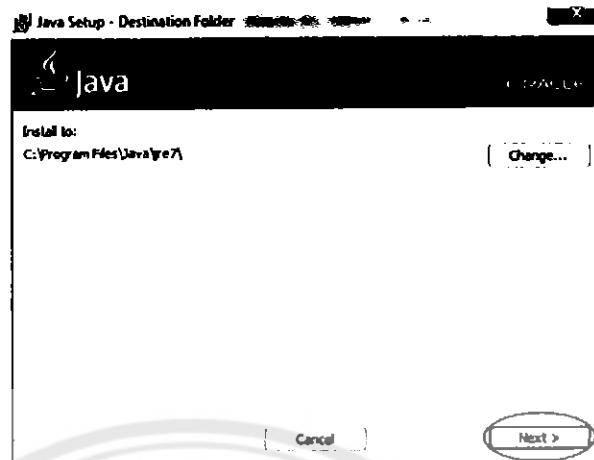
8 คลิกเลือกที่ Next



9 รอนกว่าจะทำการติดตั้งเสร็จ



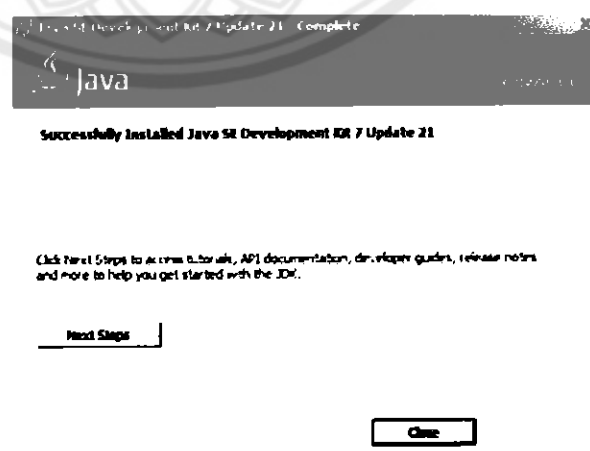
10 คลิกเลือกที่ Next



11 กำลังติดตั้ง Eclipse ให้ร้อนเสร็จสิ้น



12 การติดตั้งเสร็จสิ้น



ภาคผนวก ข

การติดตั้งแอนดรอยด์โปรแกรม Eclipse

เนื่องจากเป็นส่วนเพิ่มเติมเพราะในโปรแกรม Eclipse ไม่สามารถเขียนโปรแกรมบนแอนดรอยด์ตรงได้จึงต้องทำการติดตั้งเพิ่มเติมดังนี้

1. เข้าไปดาวน์โหลด Android SDK ที่ <http://developer.android.com/sdk/index.html>

Get the Android SDK

Before installing the Android SDK, you must agree to the following terms and conditions.

1. Introduction

1.1 The Android Software Development Kit (referred to in this License Agreement as the "SDK" and specifically including the Android system files, packaged APIs, and Google APIs add-ons) is licensed to you subject to the terms of this License Agreement. This License Agreement forms a legally binding contract between you and Google in relation to your use of the SDK.

1.2 "Android" means the Android software stack for devices, as made available under the Android Open Source Project, which is located at the following URL: <http://source.android.com/>, as updated from time to time.

1.3 "Google" means Google Inc., a Delaware corporation with its principal place of business at 1600 Amphitheatre Parkway, Mountain View, CA 94043, United States.

2. Accepting this License Agreement

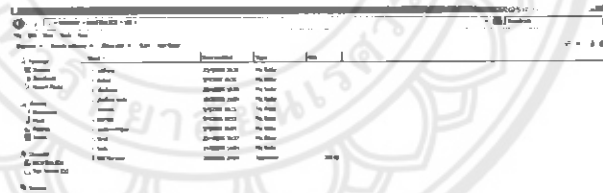
2.1 In order to use the SDK you must first agree to this License Agreement. You may not use the SDK if you do not accept this License Agreement.

I have read and agree with the above terms and conditions.

6 32 bit 64 bit

Install the Android SDK

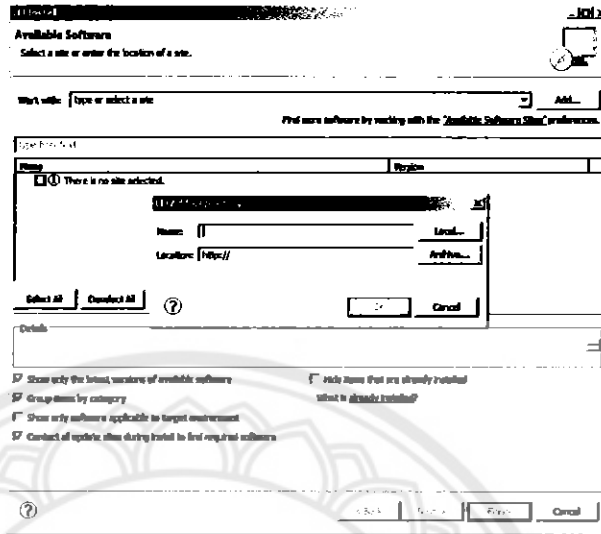
2. เมื่อดาวน์โหลดเรียบร้อยทำการแตกไฟล์ไว้ที่ C:\sdk



3. ทำการเปิดโปรแกรม Eclipse ขึ้นมาเพื่อเข้าไปตั้งค่าต่อไป

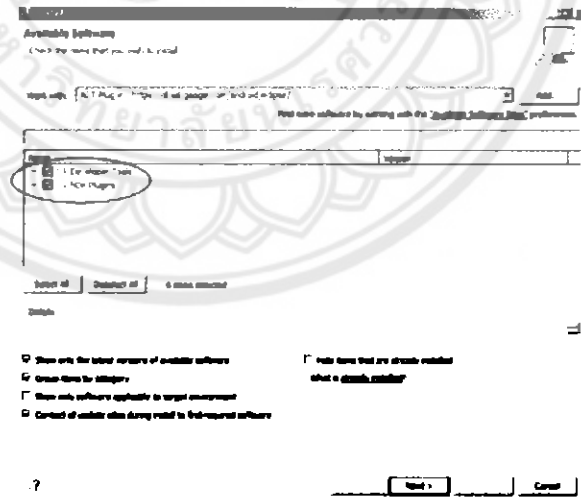


- 4 เข้าไป Install ADT Plug-in โดยให้ทำการเลือกที่ Menu Help --> Install New Software แล้วคลิกที่ปุ่ม Add จะขึ้นหน้าต่างดังรูป



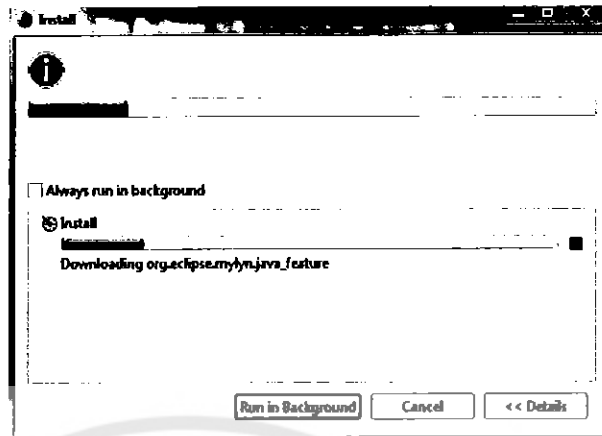
- 5 เข้าไปแล้วพิมพ์ดังนี้ - Name : ADT-Plug-in
- Location : <https://dl-ssl.google.com/android/eclipse/>

- 6 รอดค้นหา เมื่อหาเจอจะขึ้น Delevoper Tools ให้ทำการทำเครื่องหมายถูกข้างหน้าแล้วกด Next

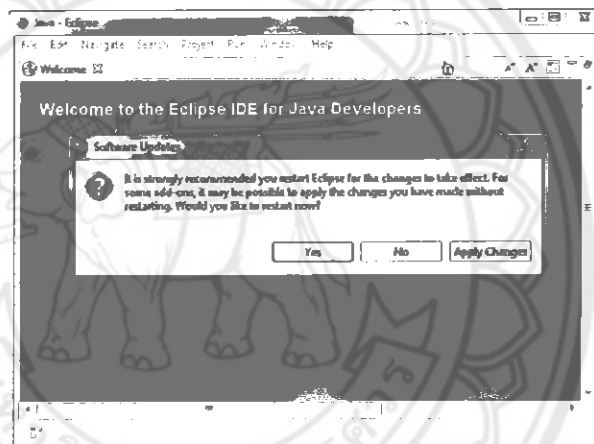


- 7 โปรแกรมจะขึ้นหน้าต่างต่อไปให้ทำการกด Next อีกครั้ง แล้วให้อ่านข้อตกลงให้ เลือกที่ I accept จากนั้นกด Finish เป็นอันเสร็จ

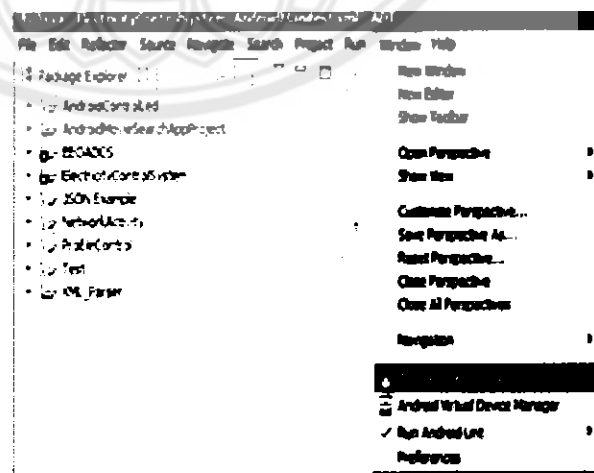
8 จากนั้นจะมีการโหลดตัว ADT Plug-in ให้รอกจนกว่าจะติดตั้งเสร็จ



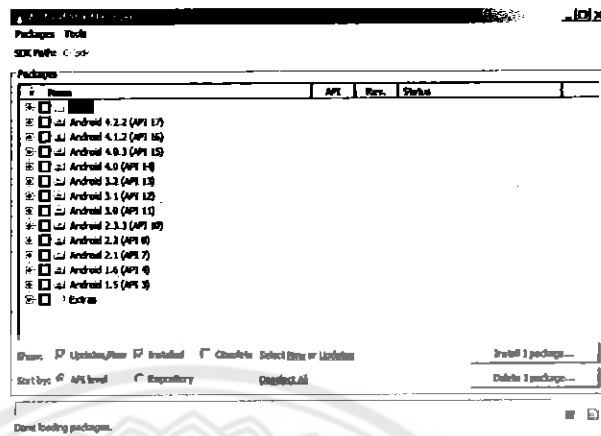
9 จากนั้นโปรแกรม Eclipse จะทำการ Reboot ถือเป็นการเสร็จสมบูรณ์



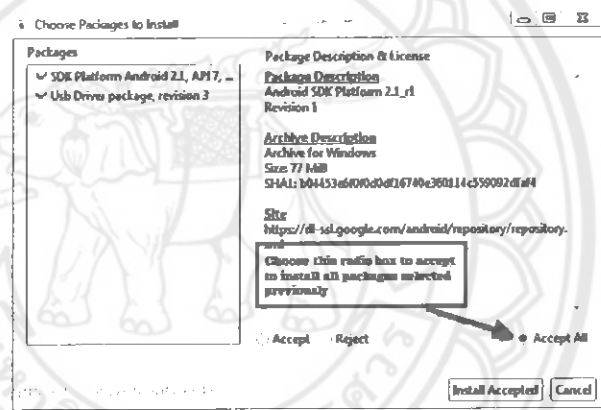
10 จากนั้นเมื่อเข้าโปรแกรม Eclipse ให้เลือกที่ Windows --> Android SDK Manager



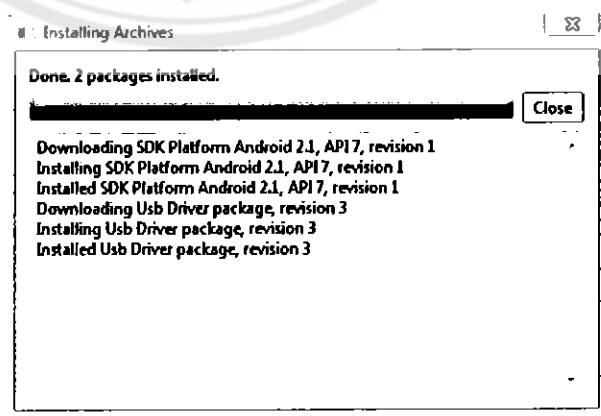
- 11 จากนั้นให้ทำการเลือกเวอร์ชันของแอนดรอยด์ที่ต้องการพัฒนาเมื่อทำการเลือก Package ที่เราต้องการจะพัฒนาเสร็จแล้วคลิกปุ่ม Install Selected



- 12 จากนั้นทำการกดเลือกที่ Accept All แล้วกดที่ Install Accepted



- 13 จากนั้นรอมความ โหลดเสร็จจะขึ้นหน้าต่างนี้ขึ้นมาให้ปิดแล้ว Reboot โปรแกรม Eclipse หนึ่งครั้ง



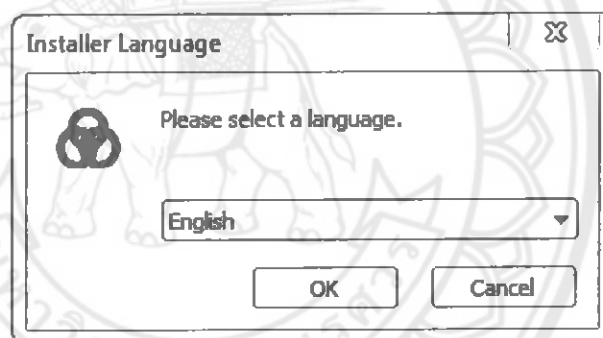
ภาคผนวก ก

การติดตั้งโปรแกรม AVR Studio

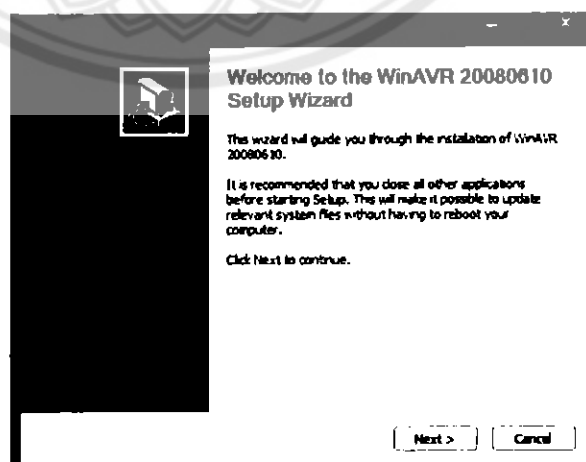
AVR Studio เป็น โปรแกรมที่ใช้สำหรับเพื่อควบคุมไมโครคอนโทรลเลอร์ใช้ของ Atmega ดังนั้น โปรแกรมที่เราต้องใช้ร่วมกันก็คือ WinAVR

WinAVR เป็น โปรแกรม Compiler จะมีหน้าที่ช่วย Compiler code ที่เราเขียนด้วย โปรแกรม AVR Studio ขั้นตอนในการลงโปรแกรมมีดังต่อไปนี้

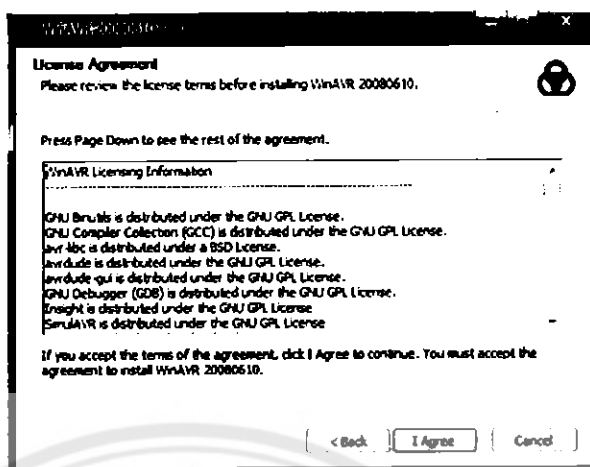
- 1 คิวโหลดโปรแกรม WinAVR มาก่อนเนื่องจากเป็น โปรแกรมที่เป็น Freeware จึง สามารถโหลดได้ที่ <http://sourceforge.net/projects/winavr/files/WinAVR/20100110/>
- 2 เมื่อคิวโหลดโปรแกรมมาแล้วก็ทำการลงโปรแกรม โดยดับเบิลคลิก โปรแกรมที่ได้คิวโหลดมา และจะปรากฏหน้าต่างดังรูปเลือกภาษา English กดตกลง



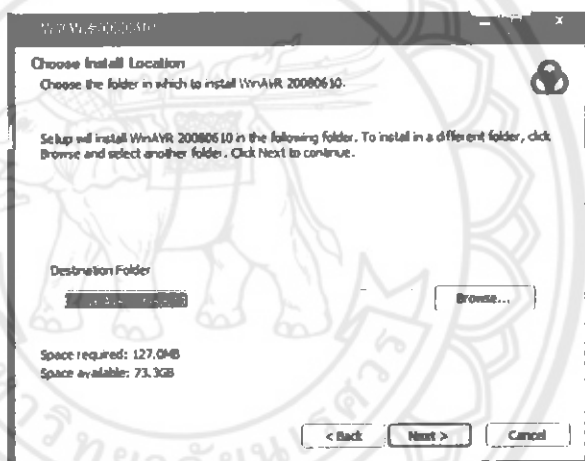
- 3 กดที่ Next



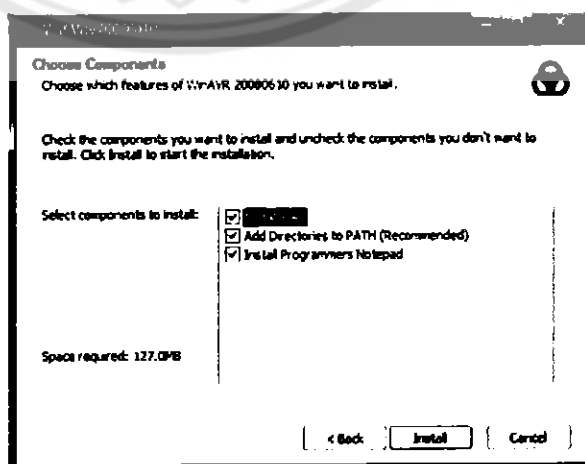
4 กดที่ I Agree



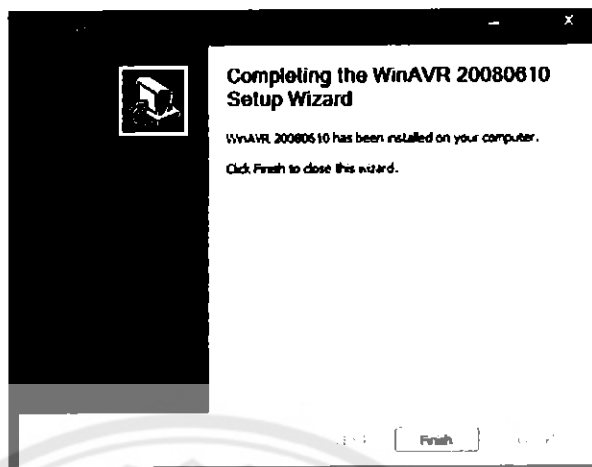
5 เลือก Folder ที่จะติดตั้งแล้วกด Next



6 ทำการคลิกที่ Install



7 จากนั้นกด Finish เป็นอันว่าลงเสร็จสิ้น

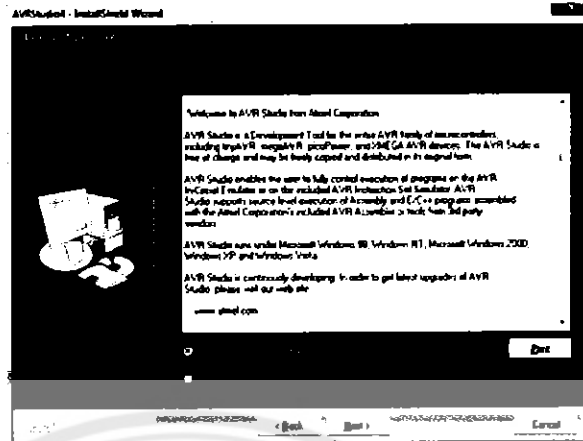


8 จากนั้นในทำการลงโปรแกรม AVR Studio โดยสามารถทำได้โดยการดาวน์โหลดจาก http://www.atmel.com/forms/software_download.asp?category_id=163&family_id=607&subfamily_id=760&fn=dl_AvrStudio4Setup.exe

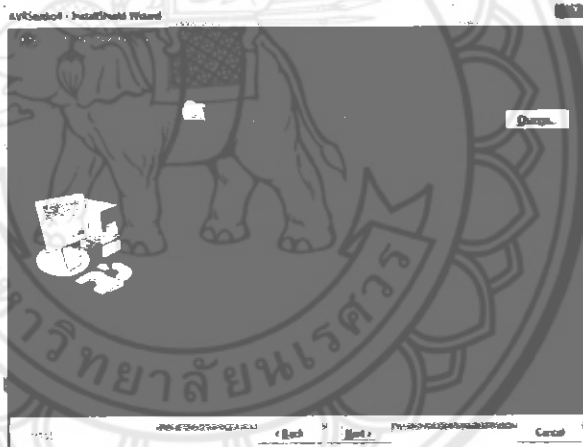
9 เมื่อดาวน์โหลดมาก็ดับเบิลคลิกไฟล์ที่ดาวน์โหลดมา และจะปรากฏหน้าต่างขึ้นมา ก็จะปรากฏหน้าต่างดังรูปแล้วก็กด Next



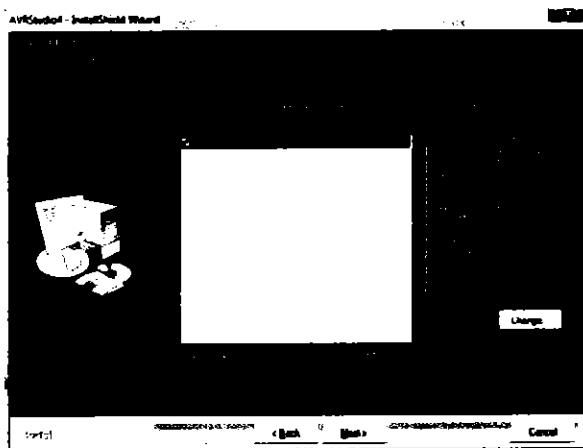
10 เลือก I accept the terms of the license agreement แล้วกด Next



11 เลือกไฟล์เครื่องที่จะทำการติดตั้ง แล้วกด Next



12 กดที่ Next



13 กดที่ Finish เป็นอันเสร็จสิ้น

