ระบบควบคุมอุณหภูมิและความชื้นผ่านระบบปฏิบัติการแอนดรอยด์
TEMPERATURE AND HUMIDITY CONTROL SYSTEM VIA ANDROID APPLICATION

โดย

นายจตุพล สุขช่วย

นายจิรกิตต์ ยอดแก้ว

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557

ระบบควบคุมอุณหภูมิและความชื้นผ่านระบบปฏิบัติการแอนดรอยด์

TEMPERATURE AND HUMIDITY CONTROL SYSTEM VIA ANDROID APPLICATION

โดย

นายจตุพล สุขช่วย 54010168

นายจิรกิตต์ ยอดแก้ว 54010198

อาจารย์ที่ปรึกษา

ผศ.ดร.สมเกียรติ ฤกษ์วรัญญู

รศ.ดร.ยุทธพงษ์ รังสรรค์เสรี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557

2514 ผ่านการตรวจรูปเล่มแล้ว
(..............................)
อาจารย์ที่ปรึกษา
12./.5./.58
KMITL วิศวกรรมโทรคมนาคม Telecommunications Engineering

2514 ผ่านการตรวจชิ้นงานแล้ว
(..............................)
กรรมการผู้ตรวจชิ้นงาน
11./.5./.58
KMITL วิศวกรรมโทรคมนาคม

ปริญญานิพนธ์ปีการศึกษา 2557

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบควบคุมอุณหภูมิและความชื้นผ่านระบบปฏิบัติการแอนดรอยด์

TEMPERATURE AND HUMIDITY CONTROL SYSTEM VIA ANDROID APPLICATION

ผู้จัดทำ

1. นายจตุพล สุขช่วย 54010168
2. นายจิรกิตต์ ยอดแก้ว 54010198

.... **อาจารย์ที่ปรึกษา**

(ผศ.ดร.สมเกียรติ ฤกษ์วรัญญู)

.... **อาจารย์ที่ปรึกษาร่วม**

(รศ.ดร. ยุทธพงษ์ รังสรรค์เสรี)

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้ประสบความสำเร็จด้วยดี จากความความช่วยเหลือและแนะนำจากหลายๆบุคคล โดยเฉพาะ อาจารย์ที่ปรึกษาสำหรับโครงงานนี้ คือ ผศ.ดร.สมเกียรติ ฤกษ์วรัญญู ซึ่งอาจารย์ได้ให้คำปรึกษา คำแนะนำและความช่วยเหลือในการทำโครงงานนี้ตลอดการทำงาน รวมทั้งรุ่นพี่และเพื่อนที่คอยแนะนำและช่วยเหลือตลอดการทำงาน

นายจตุพล สุขช่วย

นายจิรกิตต์ ยอดแก้ว

ผู้จัดทำ

ระบบควบคุมอุณหภูมิและความชื้นผ่านระบบปฏิบัติการแอนดรอยด์
TEMPERATURE AND HUMIDITY CONTROL SYSTEM VIA ANDROID APPLICATION

โดย นายจตุพล สุขช่วย 54010168
นายจิรกิตต์ ยอดแก้ว 54010198

อาจารย์ที่ปรึกษา ผศ.ดร.สมเกียรติ ฤกษ์วรัญญู
อาจารย์ที่ปรึกษาร่วม รศ.ดร.ยุทธพงษ์ รังสรรค์เสรี

บทคัดย่อ

โครงงานนี้เป็นโครงงานที่สร้างและออกแบบแอพพลิเคชั่นที่ใช้บนระบบปฏิบัติการแอนดรอยด์เพื่อใช้ในการควบคุมวงจรไฟฟ้า เพื่อที่จะควบคุมอุณหภูมิในพื้นที่ปิดนั้นๆให้ไม่เกินค่าที่กำหนดไว้ โดยวงจรไฟฟ้าจะมีเซนเซอร์เพื่อวัดค่าอุณหภูมิและค่าความชื้นในพื้นที่ปิด ระบบทำการส่งค่าข้อมูลเข้าไปเก็บในฐานข้อมูลแสดงผลค่าอุณหภูมิและค่าความชื้นในพื้นที่ปิดนั้นๆ โดยแอพพลิเคชั่นที่ออกแบบจะทำหน้าที่ควบคุมอุปกรณ์ไฟฟ้าที่เชื่อมต่อกับวงจรไฟฟ้า

ABSTRACT

The thesis is design and creating a system and application on Android operating system that use to control the temperature in the closed area according to need. By circuits have a sensors to measure the temperature and humidity values in the closed area. The system sends data to store in the database and interpolate data. Display the values of temperature and humidity in the closed area. Application was designed to control electrical equipment.

สารบัญ

		หน้า
กิตติกรรมประกาศ		I
บทคัดย่อ		II
สารบัญ		III
สารบัญรูป		VII
สารบัญตาราง		IX
บทที่ 1	**บทนำ**	**1**
	1.1 ความเป็นมาและความสำคัญของปัญหา	1
	1.2 วัตถุประสงค์	1
	1.3 ขอบเขตของโครงงาน	1
บทที่ 2	**ทฤษฎีและหลักการที่เกี่ยวข้อง**	**3**
	2.1 แอนดรอยด์ (ANDROID)	3
	2.1.1 เวอร์ชั่นของแอนดรอยด์	3
	2.1.2 คุณสมบัติและความสามารถของแอนดรอยด์	5
	2.1.3 สถาปัตยกรรมของแอนดรอยด์ (ANDROID ARCHITECTURE)	6
	2.1.3.1 ชั้นแอพพลิเคชั่น (APPLICATION)	7
	2.1.3.2 ชั้นแอพพลิเคชั่นเฟรมเวิร์ค (APPLICATION FRAMEWORK)	7
	2.1.3.3 ชั้นไลบรารี (LIBRARY)	8
	2.1.3.4 ชั้นลินุกซ์เคอร์เนล (LINUX KERNEL)	9
	2.1.4 ส่วนประกอบของแอพพลิเคชั่น (APPLICATION COMPONENT)	9
	2.1.5 วัฏจักรการทำงานของแอพพลิเคชั่น (APPLICATION LIFE CYCLE)	9
	2.1.5.1 สถานะทำงานของแอพพลิเคชั่น	11
	2.2 การประมาณค่าในช่วง (INTERPOLATION)	13

สารบัญ(ต่อ)

		หน้า
	2.2.1 การประมาณค่าในช่วงแบบเชิงเส้น (LINEAR INTERPOLATION)	13
	2.3 อุณหภูมิและความชื้นสัมพัทธ์	15
	2.3.1 อุณหภูมิ	15
	2.3.2 ความชื้นสัมพัทธ์	16
	2.4 หลักการทำงานของเซนเซอร์ และไมโครคอนโทรลเลอร์	16
	2.4.1 เซนเซอร์ตรวจวัดอุณหภูมิ (TEMPERATURE SENSOR)	16
	2.4.2 เซนเซอร์ตรวจวัดความชื้น	18
	2.4.3 หลักการทำงานของ ARDUINO UNO R3	19
	2.5 WIRELESS LAN (WLAN)	20
	2.5.1 มาตรฐาน WIRELESS LAN IEEE 802.11	21
	2.6 ประเภทของโรงเรือน	22
	2.6.1 โรงเรือนระบบปิด (EVAPORATIVE COOLING SYSTEM EVAP)	22
	2.6.1.1 ระบบ EVAPORATIVE COOLING SYSTEM	23
	2.6.2. โรงเรือนระบบเปิด (OPEN HOUSE)	23
บทที่ 3	**การออกแบบและการจัดทำโครงงาน**	**24**
	3.1 การออกแบบ	24
	3.1.1 การออกแบบการทำงานของไมโครคอนโทรลเลอร์	25
	3.1.1.1 ควบคุมการทำงานของพัดลมตามค่าอุณหภูมิที่ตั้งไว้	26
	3.1.1.2 ควบคุมการทำงานของพัดลมโดยผ่าแอพพลิเคชั่นที่ใช้ระบบปฎิบัติ การแอนดรอยด์	27
	3.1.1.3 ไมโครคอนโทรลเลอร์ส่งค่าของอุณหภูมิและความชื้นสัมพัทธ์ไปยังฐานข้อมูลที่เซิร์ฟเวอร์	27
	3.1.2 การออกแบบแอพพลิเคชั่น	28

3.1.2.1 หน้าล็อคอินสำหรับผู้ใช้งานเพื่อที่จะเข้าไปควบคุมการเปิดปิดของพัดลม 30
3.1.2.2 ออกแบบหน้าแสดงผลค่าข้อมูล 32
3.1.3 การออกแบบฐานข้อมูล (DATABASE) 33
3.1.4 การออกแบบโปรแกรมที่ใช้แสดงผลข้อมูลจากฐานข้อมูล 33
3.2 เครื่องมือที่ใช้ในการทดลอง 33
3.2.1 อุปกรณ์ด้านฮาร์ดแวร์ 33
3.2.2 อุปกรณ์ด้านซอร์ฟแวร์ 34
3.3 การจัดเก็บผลการทดลอง 35
3.3.1 การทำงานของแอพพลิเคชั่น 35
3.3.2 การควบคุมไมโครคอนโทรลเลอร์ผ่านแอพพลิเคชั่น 35
3.3.3 เก็บผลการทดลองจากการเชื่อมต่อ WI-FI MODULE กับเซิร์ฟเวอร์ 35
3.3.4 เก็บผลการทดลองการวัดค่าอุณหภูมิ 35
3.3.5 การวัดสัญญาณที่เซนเซอร์ส่งค่าไปยังไมโครคอนโทรลเลอร์ 35
3.3.6 การแสดงผลค่าอุณหภูมิและความชื้นสัมพัทธ์จากฐานข้อมูล 35
บทที่ 4 ผลการทดลอง **36**
4.1 ผลการทดสอบการทำงานของแอพพลิเคชั่น 36
4.2 การควบคุมไมโครคอนโทรลเลอร์ผ่านแอพพลิเคชั่น 39
4.3 ผลการทดลองจากการเชื่อมต่อ WI-FI MODULE กับเซิร์ฟเวอร์ 41
4.4 ผลการทดลองการวัดค่าอุณหภูมิ 42
4.5 การวัดสัญญาณที่เซนเซอร์ส่งค่าไปยังไมโครคอนโทรลเลอร์ 43
4.6 การแสดงผลค่าอุณหภูมิและความชื้นสัมพัทธ์จากฐานข้อมูล 44
บทที่ 5 สรุปผลและข้อเสนอแนะ **45**
5.1 สรุปผล 45
5.2 ข้อเสนอแนะ 45
บรรณานุกรม 46
ภาคผนวก ก. ข้อมูลเซนเซอร์ 47

สารบัญ(ต่อ)

หน้า

ภาคผนวก ข. ข้อมูลไมโครคอนโทรลเลอร์ 58

สารบัญรูป

รูปที่		หน้า
1.1	บล็อกไดอะแกรมการทำงานของระบบ	2
2.1	สัญลักษณ์ของระบบปฏิบัติการแอนดรอยด์	4
2.2	ส่วนประกอบภายในชั้นต่างๆของสถาปัตยกรรมของแอนดรอยด์	7
2.3	วัฏจักรการทำงานของแอพพลิเคชั่น	11
2.4	กราฟความสัมพันธ์ระหว่างข้อมูล 2 ค่า	14
2.5	เซนเซอร์ตรวจวัดอุณหภูมิ	17
2.6	โรงเรือนแบบปิด	23
3.1	บล็อกไดอะแกรมของระบบ	24
3.2	แผนผังการทำงานของไมโครคอนโทรลเลอร์	25
3.3	วงจรรวมของระบบ	26
3.4	วงจรรีเลย์	27
3.5	หน้าเมนูหลักของแอพพลิเคชั่น	28
3.6	สถานะการเชื่อมต่อเครือข่ายของโทรศัพท์มือถือ	29
3.7	แผนผังการทำงานการตรวจสอบสถานะการเชื่อมต่อ	29
3.8	ไปยังหน้าล็อกอินของแอพพลิเคชั่น	30
3.9	แอพพลิเคชั่นแจ้งเตือนเมื่อไม่พบข้อมูล	30
3.10	ผังการทำงานหน้าล็อกอินของแอพพลิเคชั่น	31
3.11	ล็อกอินและไปยังหน้าควบคุมของแอพพลิเคชั่น	32
3.12	แอพพลิเคชั่นเปิดไปยังหน้า DATA	32
3.13	รูปแบบของโปรแกรมที่ใช้แสดงผลข้อมูล	33
4.1	หน้าแรกของแอพพลิเคชั่น	36
4.2	แอพพลิเคชั่นที่เชื่อมต่อกับเครือข่ายแล้ว	37
4.3	ผู้ใช้งานเข้าสู่ระบบ	37
4.4	ข้อมูลในตารางฐานข้อมูล	38

สารบัญรูป(ต่อ)

หน้า

4.5 แอพพลิเคชั่นขึ้นข้อความแสดง LOGIN SUCCESS 38

4.6 แอพพลิเคชั่นขึ้นข้อความแสดง LOGIN ERROR 39

4.7 แอพพลิเคชั่นหน้าควบคุมการทำงาน 39

4.8 ค่าคำสั่งที่ถูกส่งมาเก็บในฐานข้อมูล 40

4.9 พัดลมที่ทำงานจากการสั่งงานผ่านแอพพลิเคชั่น 40

4.10 เช็คสถานะการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับเซิร์ฟเวอร์ 41

4.11 ค่าอุณหภูมิและความชื้นสัมพัทธ์ในฐานข้อมูล 41

4.12 โปรแกรมคอมพิวเตอร์ที่ใช้แสดงค่าจากเซนเซอร์ 42

4.13 สัญญาณที่วัดจากขาของเซนเซอร์ 42

4.14 โปรแกรมแสดงผลข้อมูล แสดงกราฟจากข้อมูลในฐานข้อมูล 44

สารบัญตาราง

ตารางที่		หน้า
2.1	เวอร์ชั่น วันที่เปิดตัว โค้ด API LEVEL ของแอนดรอยด์เวอร์ชั่นต่างๆ	4
2.2	ชั้น 4 ชั้นหลักของสถาปัตยกรรมของแอนดรอยด์	6
2.3	การเปรียบเทียบอุณหภูมิในหน่วยต่าง ๆ ใช้น้ำเป็นสสารมาตรฐาน	16
2.4	คุณสมบัติของมาตรฐาน IEEE ชนิดต่างๆ	22
4.1	ค่าอุณหภูมิจากเซนเซอร์และจากมิเตอร์วัดอุณหภูมิ	41

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในระบบพื้นที่ปิดเป็นพื้นที่ที่ต้องควบคุมปัจจัยต่างๆ เช่น อุณหภูมิ ความชื้นสัมพัทธ์ ความดันบรรยากาศ เป็นต้น ในปัจจุบันมีการสร้างพื้นที่ระบบปิดเพื่อนำไปใช้งานในด้านต่างๆมากมาย ทั้งในด้านการเกษตรกรรม เป็นการสร้างโรงเรือนในระบบปิด เพื่อต้องการควบคุมปัจจัยต่างๆที่มีผลต่อการทำเกษตรกรรมและส่งผลต่อผลผลิต หรือมีการสร้างพื้นที่ระบบปิดของห้องชุมสายโทรศัทพ์ เพราะอุปกรณ์ด้านการสื่อสาร เมื่อทำงานจะเกิดความร้อนและเมื่อเกิดความร้อนจะทำให้เกิดสัญญาณรบกวนในระบบสื่อสาร ทำให้ต้องมีระบบที่คอยควบคุมอุณหภูมิให้อยู่ในระดับปกติเพื่อไม่ให้ระบบต่างๆเกิดความเสียหาย จากปัญหาดังกล่าวจึงสร้างระบบที่สามารถควบคุมปัจจัยต่างๆในพื้นที่ปิดได้

1.2 วัตถุประสงค์

1) เพื่อสร้างระบบในการติดตาม สังเกตการณ์ ควบคุมค่าอุณหภูมิและความชื้นสัมพัทธ์ภายในพื้นที่ปิด

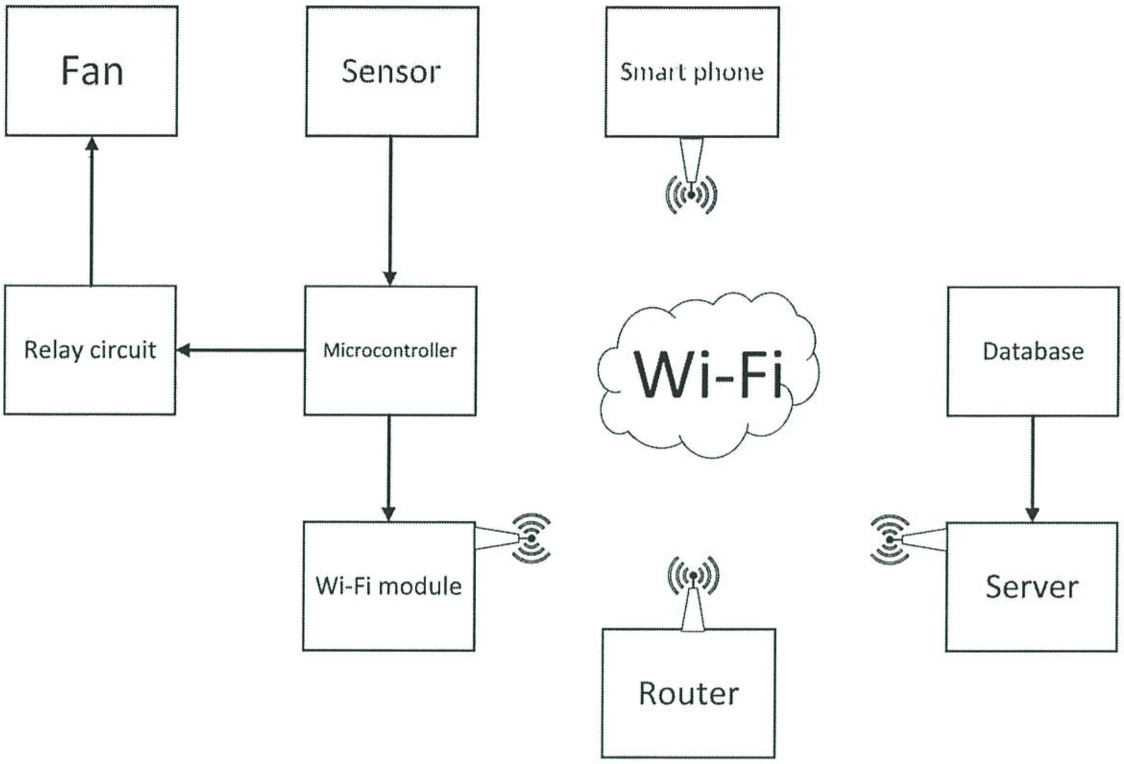
2) จัดเก็บค่าข้อมูลอุณหภูมิและความชื้นสัมพัทธ์ลงในระบบฐานข้อมูล

3) สร้างแอพพลิเคชั่นในการควบคุมการทำงานของระบบ

4) การประมาณค่าอุณหภูมิและความชื้นในบริเวณพื้นที่ปิดนั้น

1.3 ขอบเขตของปริญญานิพนธ์

สร้างระบบประมวลผลเพื่อวัดค่าอุณหภูมิและค่าความชื้นสัมพัทธ์ในพื้นที่ปิดเพื่อสังเกตการณ์และบันทึกค่าข้อมูลลงในระบบฐานข้อมูลผ่านระบบเครือข่าย โดยสร้างแอพพลิเคชั่นที่ใช้งานบนระบบปฏิบัติการแอนดรอยด์เพื่อใช้เชื่อมต่อกับระบบประมวลผลและสร้างวงจรไฟฟ้าเพื่อควบคุมการทำงานอุปกรณ์ไฟฟ้า ซึ่งการทำงานโดยคร่าวๆของระบบสามารถแสดงเป็นบล็อกไดอะแกรมดังรูปที่ 1.1 จากรูปที่ 1.1 ระบบจะมีไมโครคอนโทรลเลอร์ซึ่งต่อกับเซนเซอร์สำหรับวัดค่าอุณหภูมิและความชื้นสัมพัทธ์และเชื่อมต่อเครือข่ายไร้สายผ่าน Wi-Fi module โดยที่มีเราเตอร์ทำหน้าที่สร้างสัญญาณเครือข่ายไร้สาย ที่ฝั่งเซิร์ฟเวอร์จะมีฐานข้อมูลเพื่อใช้สำหรับเก็บค่าข้อมูลที่ส่งมาจากไมโครคอนโทรลเลอร์



รูปที่ 1.1 : บล็อกไดอะแกรมการทำงานของระบบ

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 แอนดรอยด์ (Android)

แอนดรอยด์ คือ ระบบปฏิบัติการ (Operating System) สำหรับอุปกรณ์พกพา เช่น โทรศัพท์มือถือ คอมพิวเตอร์แท็บแล็ต เป็นต้น แอนดรอยด์เป็นซอฟต์แวร์ที่มีโครงสร้างแบบเรียงทับซ้อนหรือแบบสแต็ก ซึ่งรวมเอาระบบปฏิบัติการ มิดเดิลแวร์ และแอพพลิเคชั่นที่สำคัญเข้าไว้ด้วยกัน เพื่อใช้สำหรับอุปกรณ์พกพา โดยสัญลักษณ์ของระบบปฏิบัติการแอนดรอยด์สามารถดูได้จากรูปที่ 2.1

แอนดรอยด์ถูกพัฒนามาจากระบบปฏิบัติการลินุกซ์ (Linux) การทำงานของแอนดรอยด์มีพื้นฐานอยู่บนระบบลีนุกซ์ เคอร์เนล (Linux Kernel) ซึ่งใช้ Android SDK (Software Development Kit) เป็นเครื่องมือสำหรับการพัฒนาแอพพลิเคชั่นบนระบบปฏิบัติการแอนดรอยด์ โดยใช้ภาษาจาวา (Java) ในการพัฒนา โดยการเข้าถึงความสามารถต่างๆของแอนดรอยด์จะกระทำผ่าน Java Library ที่กูเกิลจัดเตรียมไว้ให้ใน Android SDK

แอนดรอยด์ถูกพัฒนาโดยบริษัท Android Inc. ซึ่งก่อตั้งในปี ค.ศ.2003 โดย Andy Rubin และ Rich Miner ต่อมาในปี ค.ศ.2005 บริษัทกูเกิล (Google) ได้ซื้อกิจการของบริษัทนี้ และได้ร่วมมือกับกลุ่มบริษัททางด้านฮาร์ดแวร์ ซอฟต์แวร์และการสื่อสาร จัดตั้งเป็นองค์กรความร่วมมือที่มีชื่อว่า Open Handset Alliance โดยมีวัตถุประสงค์ในการสร้างแพลตฟอร์ม (Platform) สำหรับอุปกรณ์พกพาที่มีพื้นฐานอยู่บนมาตรฐานเปิด (Open Standard) หรือโอเพ่นซอร์ส (Open Source) โดยมีลิขสิทธิ์ตาม Apache Version 2 license

ซึ่งหลักลิขสิทธิ์ของ Apache จะอนุญาตให้ผู้พัฒนาสามารถนำโค้ดที่มีอยู่นำไปพัฒนาต่อได้ ทั้งในส่วนของการค้าหรือซอฟต์แวร์กรรมสิทธิ์และแบบใช้ฟรีแวร์

รูปที่ 2.1 สัญลักษณ์ของระบบปฏิบัติการแอนดรอยด์

2.1.1 เวอร์ชั่นของแอนดรอยด์

แอนดรอยด์ผ่านการพัฒนามาอย่างต่อเนื่องดังที่แสดงในตาราง ซึ่งประกอบด้วย เวอร์ชั่น วันที่เปิดตัว ชื่อรหัสหรือโค้ดเนม (Codename) และ API Level ของแอนดรอยด์เวอร์ชั่นต่างๆ

ตารางที่ 2.1 เวอร์ชั่น วันที่เปิดตัว โค้ด API Level ของแอนดรอยด์เวอร์ชั่นต่างๆ

เวอร์ชั่น	วันที่เปิดตัว	โค้ดเนม	API Level
1.0	23 กันยายน 2008	-	1
1.1	9 กุมภาพันธ์ 2009	-	2
1.5	30 เมษายน 2009	Cupcake	3
1.6	15 กันยายน 2009	Donut	4
2.0	26 ตุลาคม 2009	Éclair	5
2.01	3 ธันวาคม 2009	Éclair	6
2.1	12 มกราคม 2010	Éclair	7
2.2 - 2.2.3	20 พฤษภาคม 2010	Froyo	8

2.3 - 2.3.2	6 ธันวาคม 2010	Gingerbread	9
2.3.3 - 2.3.7	9 กุมภาพันธ์ 2011	Gingerbread	10
3.0	22 กุมภาพันธ์ 2011	Honeycomb	11
3.1	10 พฤษภาคม 2011	Honeycomb	12
3.2	15 กรกฎาคม 2011	Honeycomb	13
4.0 - 4.0.2	19 ตุลาคม 2011	Ice Cream Sandwich	14
4.0.3 - 4.0.4	16 ธันวาคม 2011	Ice Cream Sandwich	15
4.1 - 4.1.2	27 มิถุนายน 2012	Jelly Bean	16
4.2 - 4.2.2	13 พฤศจิกายน 2012	Jelly Bean	17
4.3	24 กรกฎาคม 2013	Jelly Bean	18
4.4	31 ตุลาคม 2013	Kitkat	19

2.1.2 คุณสมบัติและความสามารถของแอนดรอยด์

1) การเชื่อมต่อ แอนดรอยด์สนับสนุนเทคโนโลยีการเชื่อมต่อ ได้แก่ GSM, EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC และ WiMAX

2) Messaging สนับสนุน SMS, MMS, Threaded Text Messaging และ Cloud To Device Messaging Framework (C2DM)

3) การจัดเก็บข้อมูล แอนดรอยด์มี SQLite ซึ่งเป็นฐานข้อมูลเชิงสัมพันธ์ขนาดเล็กที่มีประสิทธิภาพสูง สำหรับใช้จัดเก็บข้อมูล

4) เว็บบราวเซอร์ แอนดรอยด์ติดตั้งมาพร้อมกับโปรแกรมเว็บบราวเซอร์ที่พัฒนาบนเอนจิ้น WebKit และใช้จาวาสคลิปต์เอนจิ้น V8 ของเว็บบราวเซอร์ Google Chrome

5) มีเดีย (Media) สนับสนุนเสียง วีดิโอ และรูปภาพในฟอร์แมตต่างๆ เช่น MPEG4, MP3, AAC, AMR, JPG และ PNG

6) สตรีมมิง (Streaming) สนับสนุน RTP/RTSP streaming และ HTML progressive download

7) สนับสนุนจาวา การพัฒนาแอพพลิเคชั่นบนแอนดรอยด์จะใช้ภาษาจาวา

8) มัลติทัช (Multi-touch) รองรับการใช้นิ้วมือแตะหน้าจอเพื่อสั่งงานได้มากกว่า1จุดพร้อมกัน

9) มัลติทาสกิง (Multi-tasking) คือความสามารถในการรันแอพพลิเคชั่นหลายแอพพลิเคชั่นพร้อมกัน

10) Tethering (หรือ Mobile Hotspot) คือความสามารถในการแชร์อินเตอร์เน็ตผ่านอุปกรณ์แอนดรอยด์

11) สนับสนุนฮาร์ดแวร์เสริมอื่นๆ เช่น กล้องถ่ายรูป, GPS

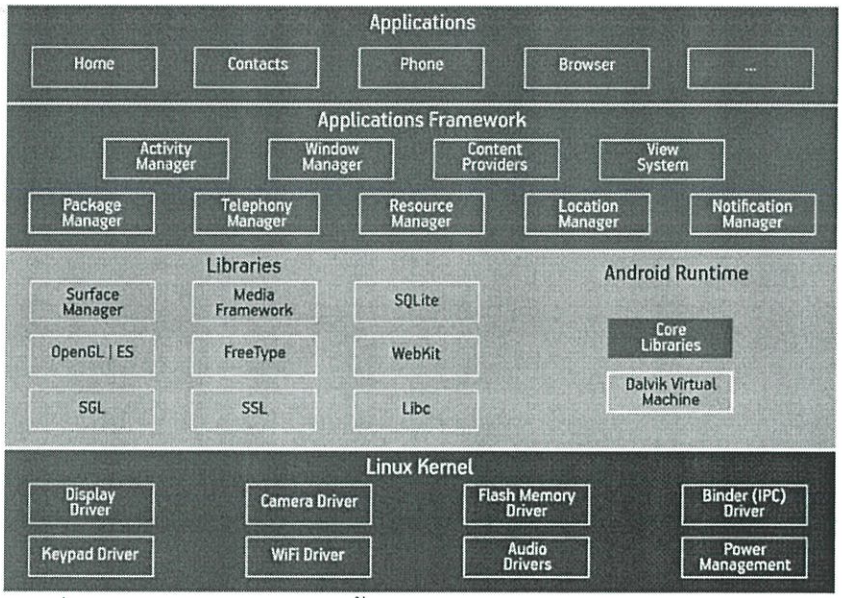
12) สนับสนุนหลายภาษา

2.1.3 สถาปัตยกรรมของแอนดรอยด์ (Android Architecture)

สถาปัตยกรรมของการออกแบบแอนดรอยด์นั้นจะถูกแบ่งออกปีนลำดับชั้น หรือ Layer โดยที่แต่ละชั้นจะเรียกใช้บริการจากระดับชั้นที่อยู่ด้านล่างของตัวเอง ซึ่งสามารถแบ่งออกเป็น 4 ชั้นหลักคือ ชั้นลินุกซ์เคอร์เนล (Linux Kernel) ชั้นไลบรารี (Library) ชั้นแอพพลิเคชั่นเฟรมเวิร์ค (Application Framework) และชั้นแอพพลิเคชั่น (Application) ดังแสดงในรูปที่ 2.2

ตารางที่ 2.2 ชั้น 4 ชั้นหลักของสถาปัตยกรรมของแอนดรอยด์

Application
Application Framework
Library
Linux Kernel



รูปที่ 2.2 ส่วนประกอบภายในชั้นต่างๆของสถาปัตยกรรมของแอนดรอยด์

2.1.3.1 ชั้นแอพพลิเคชั่น (Application)

ชั้นนี้เป็นชั้นบนสุดของโครงสร้างแอนดรอยด์ซึ่งเป็นส่วนของแอพพลิเคชั่นที่พัฒนาขึ้นมาใช้งาน เช่น อีเมลล์ เว็บบราวเซอร์ แอพพลิเคชั่นโทรศัพท์ เป็นต้น

2.1.3.2 ชั้นแอพพลิเคชั่นเฟรมเวิร์ค (Application Framework)

ถัดจากชั้น Native Libraries และ Android Runtime เป็นส่วนของเฟรมเวิร์คใช้พัฒนาแอพพลิเคชั่น ซึ่งประกอบด้วยคอมโพเนนต์พื้นฐานต่างๆที่ใช้ในการสร้างแอพพลิเคชั่น คอมโพเนนต์เหล่านี้จะติดตั้งมากับแอนดรอยด์ซึ่งส่วนสำคัญใน Application Framework มีดังนี้

1) Active Manager เป็นส่วนควบคุม Life Cycle ของแอพพลิเคชั่น

2) View System เป็นส่วนควบคุมการทำงานสำหรับการสร้างแอพพลิเคชั่น เช่น lists, grids, text, buttons เป็นต้น

3) Location Manager เป็นส่วนดูแลค่าตำแหน่งของเครื่องอุกปรณ์พกพาเคลื่อนที่

4) Content Provider เป็นส่วนควบคุมการเข้าถึงของข้อมูลที่มีการใช้งานร่วมกันระหว่างแอพพลิเคชั่นที่แตกต่างกันเช่น ข้อมูลผู้ใช้

5) Resource Manager เป็นส่วนการดูแลการเข้าใช้ข้อมูลต่างๆ ที่ไม่ใช้โค้ด

6) Notification Manager เป็นส่วนควบคุมอีเวนต์ (Event) ต่างๆ ที่แสดงบนแถบสถานะ (Status bar)

2.1.3.3 ชั้นไลบรารี (Library)

ไลบรารีของแอนดรอยด์ ซึ่งทั้งหมดถูกเขียนด้วยภาษา C หรือ C++ และถูกคอมไพล์มาสำหรับฮาร์ดแวร์ของอุปกรณ์แต่ละรุ่น ดังนี้

1) System C library เป็นกลุ่มของไลบรารีมาตรฐานที่อยู่บนพื้นฐานของภาษา C ไลบรารี (libc)

2) Media libraries เป็นกลุ่มการทำงานมัลติมีเดีย เช่น เสียง วิดีโอ รูปภาพต่างๆ เช่นไฟล์สกุล MPEG4, MP3, JPG และ PNG เป็นต้น

3) Surface Manager เป็นไลบรารีที่จัดการส่วนแสดงผลของหน้าจอ การวาดหน้าจอ

4) SQLite คือ Database Engine ที่มีประสิทธิภาพละมีขนาดเล็ก เพื่อให้จัดเก็บข้อมูลของแอพพลิเคชั่นไว้ในรูปของฐานข้อมูลเชิงสัมพันธ์ (Relational Database)

5) WebKit คือไลบรารีที่ใช้แสดงเนื้อหาเว็บเพจ ซึ่งมีลักษณะแบบเดียวกับ Google Chrome, Safari โดยในชั้นไลบรารีนี้แอนดรอยด์จะถูกแบ่งเป็นชั้นย่อยที่เรียกว่า Android Runtime ซึ่งประกอบด้วย 2 ส่วนหลักคือ Dalvik VM และ Core Java Library ส่วนDalvik VM (Virtual Machine) ส่วนนี้ถูกเขียนด้วยภาษาจาวา เพื่อใช้เฉพาะการใช้งานอุปกรณ์พกพาเคลื่อนที่ มักถูกออกแบบมาสำหรับอุปกรณ์ที่ใช้ไฟฟ้าจากแบตเตอรี่และมีหน่วยความจำจำกัด และส่วน Core Java Library ส่วนนี้เป็นไลบรารีมาตรฐาน เป็นจาวาไลบรารี ซึ่งส่วนใหญ่จะเหมือนกับ Java Standard Edition (Java SE) ซึ่งใช้พัฒนาแอพพลิเคชั่นบนพีซี

2.1.3.4 ชั้นลินุกซ์เคอร์เนล (Linux Kernel)

ระบบแอนดรอยด์อยู่บนพื้นฐานของระบบปฏิบัติการลินุกซ์ โดยชั้นลินุกซ์เคอร์เนลมีฟังก์ชันการทำงานหลายๆส่วน ซึ่งแต่ละส่วนถูกพัฒนาขึ้นด้วยภาษา C เช่น การจัดการหน่วยความจำ การจัดการโพรเซส การเชื่อมต่อเครือข่าย เป็นต้น ส่วนของลินุกซ์เคอร์เนลนี้จะทำหน้าที่เป็น Hardware Abstraction Layer หมายถึงเป็นตัวกลางระหว่างฮาร์ดแวร์กับส่วนของซอฟต์แวร์ที่อยู่ถัดขึ้นไป ทั้งนี้ผู้พัฒนาจะไม่มีสิทธิ์เข้าถึงส่วนนี้โดยตรง

2.1.4 ส่วนประกอบของแอพพลิเคชั่น (Application Component)

ซึ่ง Application Component ของแอนดรอยด์สามารถแบ่งออกเป็น 4 ส่วน คือ Activity, Service, Content Provider และ Broadcast Receiver

1) Activity คือ หน้าจอที่ติดต่อกับผู้ใช้ ซึ่งในแต่ละแอพพลิเคชั่นอาจมีมากกว่า 1 หน้าจอ หรือ 1 Activity ซึ่งแต่ละActivity จะทำหน้าที่เก็บสถานการณ์ใช้งานในส่วนต่างๆ

2) Service คือ งานหรือบริการต่างๆที่ทำงานอยู่เบื้องหลัง เช่น Service ที่เปิดดนตรีอยู่ ขณะที่ผู้ใช้กำลังใช้งานแอพพลิเคชั่นอื่นไปด้วย

3) Broadcast and Intent Receiver คือ การตอบสนองซึ่งโดยปกติ Broadcast Receiver จะเป็นการตอบสนองต่อการเกิดอีเว้นต์ของระบบในวงกว้าง เช่น การเตือนว่าแบตเตอรี่ใกล้จะหมด ส่วน Intent Receiver เป็นส่วนที่ทำให้แอพพลิเคชั่นอื่นๆเข้าถึงการทำงานของ Activity และ Service

4) Content Provider คือส่วนของการให้บริการข้อมูลสำหรับแต่ละแอพพลิเคชั่น ทั้งนี้ข้อมูลสามารถเก็บอยู่ในรูปแบบของระบบไฟล์หรือฐานข้อมูลก็ได้

2.1.5 วัฏจักรการทำงานของแอพพลิเคชั่น (Application Life Cycle)

แอนดรอยด์จะจัดการหน้าจอหรือแอคทิวิตีต่างๆในแบบสแตค (Stack) คือแอคทิวิตีที่พึ่งรันขึ้นมาจะถูกวางไว้ด้านบนสุดของสแตค และกลายเป็นแอคทิวิตีที่กำลังทำงาน (Running Activity) ส่วนแอคทิวิตีที่ทำงานมาก่อนจะอยู่ถัดลงไปตามลำดับและเมื่อแอคทิวิตีที่เพิ่งจะรันล่าสุดจบการทำงาน แอคทิวิตีในลำดับก่อนหน้า ก็จะกลับมาอยู่บนสุดตามเดิม โดยปกติแอพพลิเคชั่นจะทำงานแยกกันในแต่ละโพรเซส ในแต่ละโพรเซสอาจจะมีแอคทิวิตีที่ทำงานอยู่

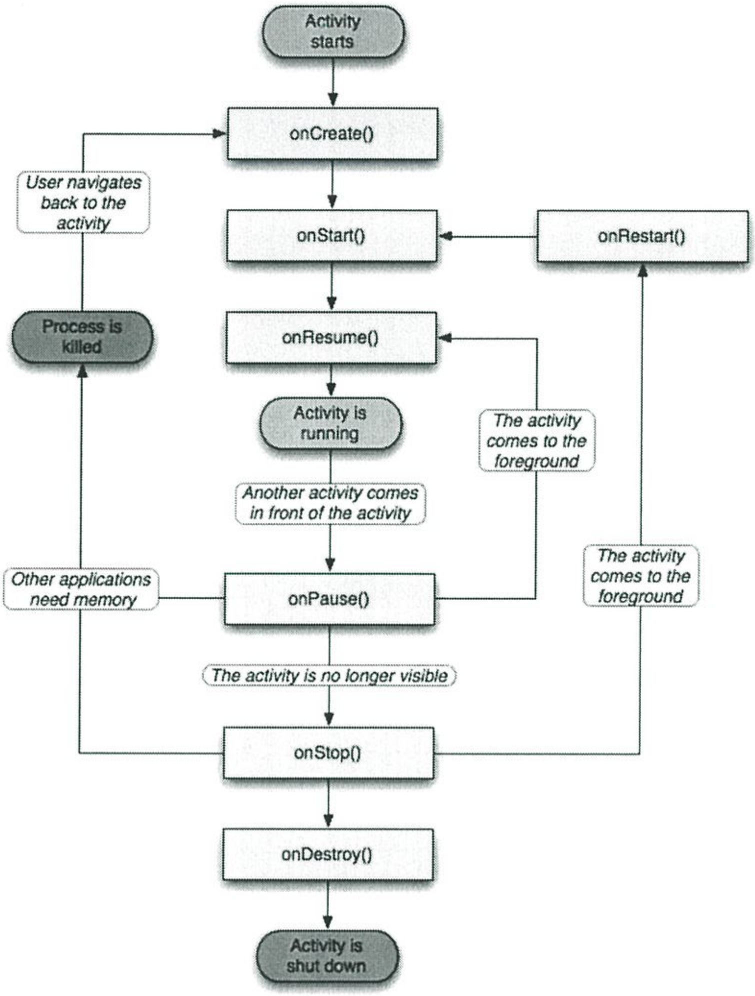
มากกว่า 1 แอคทิวิตี ซึ่งในการเริ่มทำงานของแอคทิวิตี จะเริ่มด้วย startActivity() สำหรับ แอคทิวิตีแบบซิงโครนัส (Synchronous) และแอคทิวิตีแบบอะซิงโครนัส (Asynchronous) จะเริ่มด้วย startSubActivity() โดยแต่ละแอคทิวิตีจะมีวัฏจักรชีวิต (Life cycle) ที่แยกจากกันโดยชัดเจนใน Life cycle ของแอคทิวิตีตั้งแต่เริ่มจนกระทั่งจบการทำงาน แอคทิวิตีอาจมีการเปลี่ยนแปลงสถานะดังนี้

1) ถ้าแอคทิวิตีถูกแสดงออกมาเบื้องหน้า (บนสุดของสแตค) แอคทิวิตีจะอยู่ในสถานะ Runningหรือ Active

2) ถ้าแอคทิวิตีไม่ได้อยู่เบื้องหน้า แต่ยังคงแสดงออกมา แอคทิวิตีจะอยู่ในสถานะ Paused ซึ่งสถานะดังกล่าว แอคทิวิตียังคงทำงานได้ตามปกติ แต่หากหน่วยความจำเหลือน้อย โพรเซสของแอคทิวิตีอาจถูกทำลายทิ้งไป

3) ถ้าแอคทิวิตีถูกแอคทิวิตีอื่นบังหมด จะอยู่ในสถานะ Stopped ซึ่งยังคงเก็บข้อมูลต่างๆไว้ แต่จะไม่แสดงออกผลมา และโพรเซสอาจถูกทำลายทิ้งหากระบบต้องการใช้หน่วยความจำ

4) ถ้าแอคทิวิตีอยู่ในสถานะ Paused หรือ Stopped ระบบอาจจะดึงแอคทิวิตีออกจากหน่วยความจำโดยให้แอคทิวิตีจบการทำงาน ซึ่งเมื่อแอคทิวิตีกลับมาแสดงผลอีกครั้ง จะต้องไปเริ่มต้นใหม่พร้อมทั้งเรียกคืนข้อมูลต่างๆกลับมาเอง



รูปที่ 2.3 วัฏจักรการทำงานของแอพพลิเคชั่น

2.1.5.1 สถานะทำงานของแอพพลิเคชั่น

1) จากรูปที่ 2.3 จะเริ่มต้นที่ onCreate (Bundle) จะถูกเรียกเมื่อแอคทิวิตีเริ่มต้นทำงาน ในกรณีที่มีการเรียกใช้งานเมธอด (Method) นี้ Android Framework จะนำBundle object ไปบันทึกไว้ในแอคทิวิตี ก่อนที่แอคทิวิตีจะทำงาน จากนั้นจะตามด้วยฟังก์ชัน onStart()

2) onStart() จะถูกเรียกเมื่อแอคทิวิตีกำลังจะแสดงผลออกไปให้ผู้ใช้เห็น จากนั้นสถานะจะถูกย้ายไปเป็นสถานะ onResume แต่ถ้าแอคทิวิตีนั้นไม่สามารถทำงานได้ สถานะจะถูกย้ายไปเป็นสถานะ onStop()

3) onRestart() จะถูกเรียกเมื่อแอคทิวิตีกำลังจะกลับมาแสดงผลให้ผู้ใช้เห็นอีกครั้ง ซึ่งจะตามด้วยสถานะ onStart()

4) onResume() จะถูกเรียกเมื่อแอคทิวิตีเริ่มตอบโต้กับผู้ใช้ ตรงจุดนี้แอคทิวิตีจะอยู่บนสุดของสแตคและได้รับอินพุตต่างๆจากผู้ใช้

5) onPause() จะถูกเรียกเมื่อแอคทิวิตีกำลังจะเปลี่ยนไปเป็นการทำงานเบื้องหลัง (Background) เนื่องจากจะมีแอคทิวิตีอื่นมาทำงานเบื้องหน้าแทน

6) onStop() จะถูกเรียกใช้เมื่อผู้ใช้งานไม่ต้องการใช้งานแอคทิวิตีนั้นๆ ในช่วงระยะเวลาหนึ่งๆซึ่งอาจเป็นแอคทิวิตีเก่าที่ถูกดึงกลับมาใช้หรือแอคทิวิตีใหม่ และจะตามด้วยสถานะ onRestart() เมื่อต้องการกลับมาทำงานที่แอคทิวิตีนั้นๆ แต่ในกรณีที่ระบบเหลือหน่วยความจำไม่พอ จะทำลายโพรเซสของแอคทิวิตีนั้นๆไปเลยโดยอาจไม่เรียกมายังเมธอด onStop

7) onDestroy จะถูกเรียกเป็นเมธอดสุดท้ายก่อนที่แอคทิวิตีจะปิดการทำงานไป ซึ่งอาจเป็นเพราะแอคทิวิตีกำลังจะจบการทำงาน หรือระบบขอทำลายแอคทิวิตีเพื่อนนำหน่วยความจำไปใช้

8) ช่วงเวลาทั้งหมด (Entire Lifetime) เกิดขึ้นระหว่างเมธอด onCreat ครั้งแรกไปจนถึงการเรียกเมธอด onDestroy

9) ช่วงเวลาที่แอคทิวิตีแสดงผลออกมาให้เห็น (Visible Lifetime) เกิดขึ้นระหว่างการเรียกเมธอด onStart ไปจนถึงการเรียกเมธอด onStop ช่วงเวลานี้ผู้ใช้จะเห็นแอคทิวิตีบนหน้าจอ แม้อาจจะไม่ได้อยู่เบื้องหน้าและตอบโต้กับผู้ใช้ได้ก็ตาม สองเมธอดนี้คือจุดเริ่มต้นและจุดสิ้นสุดที่จะควบคุมทรัพยากรต่างๆที่เกี่ยวข้องกับการแสดงผลของแอคทิวิตี

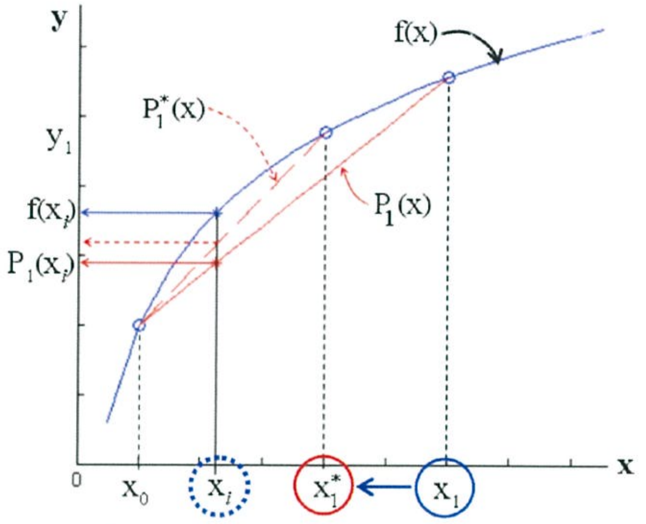
10) ช่วงเวลาที่แอคทิวิตีอยู่เบื้องหน้า (Foreground Lifetime) เกิดขั้นระหว่างการเรียกเมธอด onResume ไปจนถึงการเรียกเมธอด onPause ในช่วงเวลานี้แอคทิวิตีจะอยู่เบื้องหน้าและตอบโต้กับผู้ใช้ได้ แอคทิวิตีหนึ่งๆอาจมีการสลับจากเบื้องหน้าและเบื้องหลังมาเบื้องหน้าบ่อยๆ

2.2 การประมาณค่าในช่วง (Interpolation)

เมื่อพิจารณาปัญหาทางวิศวกรรม รวมถึงการทดลองทางวิทยาศาสตร์ต่างๆ จะพบว่า ข้อมูลที่ได้โดย จะเป็นจุดไม่ต่อเนื่อง (discrete points) จึงมีความต้องการที่จะวาดเส้นกราฟที่ต่อเนื่องผ่านจุดเหล่านั้น (to fit a curve over those data points) เพื่อเป็นการอธิบายข้อมูล หรือต้องการทราบค่าที่อยู่ระหว่างจุดของข้อมูลที่มี ในทางวิศวกรรมจะอาศัยเทคนิคที่เรียกว่า curve fitting ซึ่งสามารถกระทำได้ 2 แนวทาง วิธีแรกใช้การ Interpolation วิธีนี้จะลากเส้นกราฟ (ซึ่งก็คือ ฟังก์ชัน) ผ่านจุดทุกจุดของข้อมูลที่ เทคนิคนี้เหมาะสำหรับจุดข้อมูลที่เราทราบ หรือแน่ใจว่ามีความถูกต้องสูง เช่น ข้อมูลที่ได้จากการคำนวณที่ซับซ้อน หรือได้การทำการทดลองตามวิธีมาตรฐานด้วยเครื่องมือวัดที่มีความถูกต้องสูง ซึ่งมีค่าใช้จ่ายในการวัดสูงด้วย ซึ่งเป็นสาเหตุหนึ่ง ที่ไม่สามารถวัดออกมาได้หลายๆ จุด และวิธีที่สองคือ วิธี Regression

2.2.1 การประมาณค่าในช่วงแบบเชิงเส้น (Linear Interpolation)

จากกราฟรูปที่ 2.4 มีความสัมพันธ์ระหว่าง 2 ปริมาณ (x กับ y) ตามธรรมชาติ ที่เราต้องการทราบกำหนดให้เป็น ฟังก์ชัน f(x) แต่เนื่องจากข้อมูลจากการศึกษา ทดลองมีจำกัด อาจทราบข้อมูลเพียง 2 จุดเท่านั้น คือ จุด (x_0, y_0) กับจุด $(x_1\ , y_1)$ จากความรู้ทางคณิตศาสตร์ สามารถสร้างสมการเส้นตรงที่ unique ผ่านจุดทั้งสองนี้ สมการเส้นตรงที่หาได้ กำหนดให้เป็นสมการ y = P_1 (x) ที่นิยามในช่วง x*$[x_0\ , x_1]$ หากต้องการทราบค่าของฟังก์ชัน f(x_i) ที่ตำแหน่ง x ใดๆ กำหนดให้เป็น xi ดังรูปที่ 2.4 ในที่นี่จะพบว่าจุดต่างๆ บนแกน x เรียงตัวกันดังนี้ คือ x_0< x_i < *1x < x_1 เนื่องจากไม่ทราบ f(x) ในธรรมชาติ เพียงแค่มีจุดข้อมูล 2 จุด และสมการเส้นตรงที่ผ่าน 2 จุดนั้น จึงทำการประมาณฟังก์ชันจริงในธรรมชาติ ด้วยฟังก์ชันหรือสมการเส้นตรงที่สร้างขึ้น f(x), y, P_1 (x) ฉะนั้นค่าของฟังก์ชันที่สนใจ f(x_i) ก็จะประมาณได้ด้วย P_1 (x_i)



รูปที่ 2.4 กราฟความสัมพันธ์ระหว่างข้อมูล 2 ค่า

จากรูปที่ 2.4 จะสนใจค่าของฟังก์ชันที่ตำแหน่ง x_i ซึ่งอยู่ระหว่าง x_0 กับ x_1 เมื่อคำนวณหาสมการเส้นตรง ตามสมการจะได้สมการ P_1 (x) ค่าฟังก์ชันในตำแหน่งที่สนใจจะมีค่า f(x_i), y, P_1 (x_i) หากกำหนดช่วงข้อมูลที่ศึกษาให้แคบลง จาก x_1 เป็น *1 x เมื่อคำนวณหาสมการเส้นตรงในช่วง x*[x_0 , *1 x] จะได้สมการ *P_1 (x) และได้ค่าฟังก์ชันเท่ากับ f(xi) y *P_1 (x_i) ซึ่งจะเข้าใกล้ค่าจริง f(x_i) มากกว่า จะเห็นได้ว่าช่วงข้อมูล (interval) ที่มีความยาวต่างกัน ก็จะมีผลต่อความถูกต้องในการคำนวณหรือประมาณค่าฟังก์ชันที่ตำแหน่ง x_i แตกต่างกันไปด้วย เมื่อทราบค่า (x_1,y_1) และ (x_2 , y_2) จากจุดที่สนใจ แล้ว สามารถประมาณค่าของ y สำหรับ x เมื่อ x_1 <= x <= x_2 จากสมการที่ 1

$$y = \frac{y_1+(x-x_1)(y_2-y_1)}{(x_2-x_1)} \tag{1}$$

2.3 อุณหภูมิและความชื้นสัมพัทธ์ (Temperature and relative humidity)

2.3.1 อุณหภูมิ

1) อุณหภูมิ หมายถึง “ปริมาณของระดับความร้อน หรือความเย็นใด ๆ สามารถวัดระดับอุณหภูมิได้จาก เครื่องมือวัดที่เรียกว่า “เทอร์โมมิเตอร์ (Thermometer)” ในการวัดอุณหภูมิในสมัยแรก ๆ นั้นทำได้ยาก เพราะว่ายังไม่มีมาตรฐานในการกำหนดสเกลอุณหภูมิ ให้เป็นที่แน่นอน การตั้งระดับสเกลอุณหภูมิในยุคแรกนั้นผู้ทดลองจะเป็นผู้ตั้งเอง ค่าที่ได้จึงไม่เหมือนกับผู้ทดลองคนอื่น และเป็นที่เข้าใจยาก เมื่อทดลองทำซ้ำหลาย ๆ ครั้งความคลาดเคลื่อนจึงมีมาก ในเวลาต่อมาได้มีการกำหนดการวัดอุณหภูมิให้เป็นสเกลมาตรฐาน โดยมีการกำหนดให้จุด ๆ หนึ่งเป็นจุดเริ่มนับอุณหภูมิ เป็น “จุดที่แน่นอน” จุดนี้จะเปลี่ยนไปตามชนิดของสสาร

2) จุดที่แน่นอน (Fixed point) จุดนี้เป็นอุณหภูมิเฉพาะของแต่ละสสาร เป็น “จุดร่วม 3 จุด” ที่นำมาใช้ในวิชาเทอร์โมไดนามิกส์สมัยใหม่ ในน้ำจุดร่วม 3 จุดจะอยู่ที่ 273.16 K (0°C) คือจุดเยือกแข็งของน้ำ

3) จุดร่วม 3 จุด (Triple point) หมายถึงสภาวะเฟสสมดุลที่กล่าวถึงในวิชาเทอร์โมไดนามิกส์ ณ จุดนี้สถานะทั้งสามจะร่วมจุดเดียวกัน คือ สถานะของแข็ง ของเหลว และไอ ของสสาร ในจุดนี้ไม่มีการเปลี่ยนสถานะ มีความดัน และอุณหภูมิไม่เปลี่ยนแปลง นายเอนเดอร์ส เซลเซียส (Anders Celsius) (พ.ศ. 2244-2287) เป็นนักดาราศาสตร์ และนักประดิษฐ์ ชาวสวีเดน เป็นคนคิดค้นหน่วยอุณหภูมิองศาเซลเซียส (°C) โดยตั้งให้น้ำเป็นมาตรฐานในการกำหนดสเกลอุณหภูมิ กำหนดให้จุดเยือกแข็งของน้ำ เท่ากับ 0°C (ถ้าในทางเทอร์โมไดนามิกส์ คือจุดร่วม 3 จุด) จุดเดือดของน้ำ เท่ากับ 100°C นายการ์เบรียล ฟาห์เรนไฮท์ (Gabriel Fahrenheit) (พ.ศ. 2229-2279) เป็นนักฟิสิกส์ ชาวเยอรมัน เป็นคนคิดค้นหน่วยอุณหภูมิองศาฟาห์เรนไฮท์ (°F) โดยกำหนดให้จุดเยือกแข็งของน้ำ อยู่ที่ 32°F และจุดเดือดของน้ำ 212°F วิลเลี่ยม จอห์น แม็คเควน แรงคิน (William John MacQuorn Rankine) (พ.ศ. 2363-2415) เป็นนักฟิสิกส์ ชาวสก็อตแลนด์ เป็นคนคิดค้นหน่วยอุณหภูมิองศาแรงคิน (°R) เป็นหน่วยอุณหภูมิองศาสัมบูรณ์ที่จะนำไปเทียบกับอุณหภูมิองศาฟาห์เรนไฮท์ โดยจุด 0°R อยู่ที่ 459.67°F จุดเยือกแข็งของน้ำ เท่ากับ 491.67°R วิลเลียม ทอมสัน เคลวิน (William Thomson Kelvin) (พ.ศ. 2367-2450) เป็นนักฟิสิกส์ ชาวอังกฤษ เป็นผู้คิดค้นหน่วยอุณหภูมิเคลวิน (K : ไม่มีเครื่องหมายองศา) เป็นหน่วยอุณหภูมิที่ใช้ใน

การศึกษาในวิชาเทอร์โมไดนามิกส์ โดยกำหนดให้ศูนย์องศาสัมบูรณ์ (0K) เป็นจุดเริ่มต้น โดยไม่ได้เทียบกับสสารใด ๆ อุณหภูมิของห้องอยู่ที่ 300K

ตารางที่ 2.2 การเปรียบเทียบอุณหภูมิในหน่วยต่าง ๆ ใช้น้ำเป็นสสารมาตรฐาน

	องศาเซลเซียส (°C)	องศาฟาห์เรนไฮท์ (°F)	องศาแรงคิน (°R)	องศาเคลวิน (K)
จุดศูนย์องศาสัมบูรณ์	-273.16	-459.67	0	0
จุดเยือกแข็ง	0	32	491.67	273.16
จุดเดือด	100	212	672	373.14

2.3.2 ความชื้นสัมพัทธ์

ความชื้นสัมพัทธ์ (relative humidity) หมายถึงค่าเปรียบเทียบระหว่างมวลของไอน้ำที่มีอยู่จริงในอากาศขณะนั้น กับมวลของไอน้ำในอากาศอิ่มตัวที่อุณหภูมิและปริมาตรเดียวกัน คิดเป็นเปอร์เซนต์ดังสมการที่ 2 และสมการที่ 3

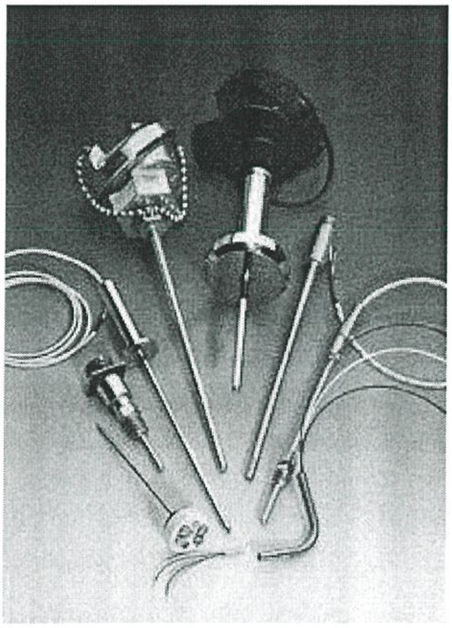
$$\text{ความชื้นสัมพัทธ์ } (\%) = \frac{\text{ความชื้นสัมบูรณ์} \left(\frac{g}{cm3}\right) \text{ x } 100}{\text{ปริมาณไอน้ำอิ่มตัว} \left(\frac{g}{cm3}\right)} \quad (2)$$

$$\text{ความชื้นสัมพัทธ์ } (\%) = \frac{\text{มวลไอน้ำที่มีอยู่จริงในอากาศขณะนั้น} \left(\frac{g}{cm3}\right) \text{ x } 100}{\text{มวลไอน้ำในอากาศอิ่มตัวที่อุณหภูมิและปริมาตรเดียวกัน}} \quad (3)$$

2.4 หลักการทำงานของเซนเซอร์ และไมโครคอนโทรล

2.4.1 เซนเซอร์ตรวจวัดอุณหภูมิ (Temperature Sensor)

การตรวจวัดอุณหภูมิใช้รูปแบบการเปลี่ยนแปลงระดับแรงดันไฟฟ้าจากสัญญาณอนาล็อกไปสู่ สัญญาณดิจิตอล โดยสัมพันธ์กับอุณหภูมิ มีรูปแบบใหญ่ ๆ ของ เซนเซอร์ อยู่ด้วยกัน 3 รูปแบบดังรูปที่ 2.5 คือ Thermocouple คือ อุปกรณ์วัดอุณหภูมิโดยใช้หลักการเปลี่ยนแปลงอุณหภูมิเป็นแรงเคลื่อนไฟฟ้า ทำมาจากโลหะตัวนำที่ต่างชนิดกัน 2 ตัว มาเชื่อมต่อปลายทั้งสองเข้าด้วยกัน ที่ปลายด้านหนึ่ง เรียกว่า "จุดอุณหภูมิ" ส่วนปลายอีกด้านหนึ่งปล่อยเปิดไว้ เรียกว่า "จุดอ้างอิง" หากที่จุดวัดอุณหภูมิและจุดอ้างอิงมีอุณหภูมิต่างกันก็จะทำให้มีการนำกระแสในวงจร Thermocouple Resistance Temperature Detector (RTD) คือ ตัวเซ็นเซอร์อุณหภูมิที่ใช้หลักการเปลี่ยนแปลงค่าความต้านทานของโลหะ ซึ่งค่าความต้านทานดังกล่าวจะมีค่าเพิ่มตามอุณหภูมิ ความต้านทานของโลหะที่เพิ่มขึ้นนี้ เรียกว่า “สัมประสิทธิ์การเปลี่ยนแปลงอุณหภูมิแบบบวก” นิยมนำไปใช้ในการวัดอุณหภูมิในช่วง -270 to 850 °C วัสดุที่นำมาใช้จะเป็นโลหะที่มีความต้านทานจำเพาะต่ำ เช่น แพลทินัม, ทังสเตนและนิกเกิล



รูปที่ 2.5 เซนเซอร์ตรวจวัดอุณหภูมิ

Thermistor เป็นอุปกรณ์ความต้านทานชนิดที่สามารถเปลี่ยนค่าความต้านทานเมื่อได้รับความร้อน โดยที่ค่าความต้านทานจะเปลี่ยนแปลงแบบไม่เป็นเชิงเส้น กับอุณหภูมิ แบ่งเป็น 2 ลักษณะคือ Positive Temperature Comitial (PTC) เป็นชนิดที่ปกติจะมีค่าความต้านทานต่ำ เมื่อได้รับความร้อนจะทำให้มีค่าความต้านทานสูงขึ้นตามลำดับอุณหภูมิ นำไปใช้ตรวจสอบระดับความร้อน หรือทำให้เกิดความร้อนขึ้นเพื่อควบคุมการจ่ายแรงดันไฟฟ้าให้กับขดลวด เช่น วงจรล้างสนามแม่เหล็กอัตโนมัติของเครื่องรับโทรทัศน์สี (Degaussing coil) เป็นต้น Negative Temperature Comitial (NTC) เป็นชนิดที่ปกติจะมีความต้านทานสูงเมื่อได้รับความร้อน ค่าความต้านทานจะต่ำลง ใช้งานด้านการตรวจสอบความร้อนเพื่อควบคุมระดับการทำงาน เช่น ในวงจรขยายเสียงที่ดีใช้ตรวจจับความร้อนที่เกิดจากการทำงานแล้วป้อนกลับไปลดการทำงานของวงจรให้น้อยลง เพื่ออุปกรณ์หลักจะไม่เกิดความร้อนมากจนเกินไปวงจรการต่ออุปกรณ์ การนำไปใช้งานการตรวจวัดอุณหภูมิใช้รูปแบบการเปลี่ยนแปลงระดับแรงดันไฟฟ้าจากสัญญาณอนาล็อกไปสู่สัญญาณดิจิตอล

2.4.2 เซนเซอร์ตรวจวัดความชื้น

เซนเซอร์วัดค่าความชื้น (Humidity Sensor) ที่ใช้กันอยู่ในอุตสาหกรรมมีอยู่ด้วยกัน 3 ชนิด คือ Capacitive, Thermal Conductivity และ Resistive

1) เซนเซอร์วัดความชื้นแบบคาปาซิตีฟ (Capacitive Humidity Sensor) เซนเซอร์วัดค่าความชื้นแบบนี้วัดค่าความชื้นสัมพัทธ์ (Relative Humidity) มีรูปร่างหน้าตาดังแสดงในรูปที่ 1 มีการใช้งานกันอย่างกว้างขวางในอุตสาหกรรม เชิงพาณิชย์ งานวิจัยหรือทดลองทางฟิสิกส์ เซนเซอร์แบบนี้มีโครงสร้างที่ประกอบไปด้วยชั้นฐานแผ่นฟิล์มบางที่ทาจากโพลีเมอร์ หรือเมทัลออกไซด์ (Metal Oxide) ถูกวางอยู่ระหว่างอิเล็กโตรดทั้งสอง โดยพื้นผิวของฟิล์มบางดังกล่าวถูกเคลือบด้วยอิเล็กโตรดโลหะแบบมีรูพรุนเพื่อป้องกันฝุ่นละอองและปัญหาจากแสงแดดเซนเซอร์แบบคาปาซิตีฟสามารถตรวจจับความชื้นสัมพัทธ์ในสภาพแวดล้อมได้เกือบจะเป็นเชิงเส้น หรือมีการตอบสนองได้อย่างเป็นสัดส่วนที่ดีนั้นเอง โดยเมื่อค่าความชื้นสัมพัทธ์เปลี่ยนไป 1 เปอร์เซ็นต์ ค่าความจุไฟฟ้า (Capacitive) ก็จะเปลี่ยนไป 0.2 ถึง 0.5 pFเซนเซอร์แบบคาปาซิตีฟถูกกำหนดให้มีคุณลักษณะเฉพาะคือค่าสัมประสิทธิ์อุณหภูมิต่ำจึงทาให้ทางานได้ดี แม้ อุณหภูมิสูงถึง 200°Cการกลับสู่สภาวะเดิมจากสภาวะการควบแน่น และยังทนต่อไอระเหยของสารเคมีอีกด้วย

2) เซนเซอร์ความชื้นแบบรีซีสตีฟ (Resistive Humidity Sensor) เซนเซอร์ความชื้นแบบความต้านทานนี้จะวัดการเปลี่ยนแปลงอิมพีแดนซ์ไฟฟ้าของตัวกลางดูดความชื้น (Hygroscopic Medium) โครงสร้างของเซนเซอร์ Resistive ประกอบด้วยอิเล็กโตรดโลหะ 2 ส่วนวางอยู่บนฐานด้วยเทคนิคการวาง แบบโฟโตรีซีส (Photoresist) อิเล็กโตรดอาจมีขดลวดพันรอบ Wire-wound Electrodes ใช้แกนเป็นพลาสติกหรือแท่งแก้วทรงกระบอกในส่วนของฐานนั้นถูกเคลือบด้วยเกลือ (Salt) หรือโพลีเมอร์ (Conductive Polymer) การทำงานของเซนเซอร์ก็คือดูดซับไอน้ำและไอออนที่แตกตัว เป็นผลให้ค่าความนำไฟฟ้าของตัวกลางเพิ่มขึ้น เมื่อความต้านทานเปลี่ยนตามการเปลี่ยนของความชื้นเป็นผลให้เกิดกระแสไฟฟ้าไหลในวงจรวัด กระแสงฟ้านี้จะถูกแปลงเป็นสัญญาณแรงดันกระแสตรงเพื่อการส่งผ่านไปยังวงจรขยายย่านวัด วงจรขยายแรงดัน วงจรปรับเชิงเส้นและวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอลต่อไป ซึ่งเมื่อผ่านกระบวนการต่าง ๆ แล้วจะทำให้เซนเซอร์และ Resistive มีการตอบสนองต่ออุณหภูมิได้ดี

3) เซนเซอร์ความชื้นแบบ Thermal Conductivity เซนเซอร์แบบนี้เป็นชนิดเดียวที่วัดค่าความชื้นสมบูรณ์ โดยอาศัยการคำนวณความแตกต่างระหว่างค่าการนำความร้อนของอากาศแห้ง(Thermal Conductivity) กับการนำความร้อนของอากาศที่มีไอน้ำอยู่ โดยเมื่ออากาศหรือก๊าซแห้ง มันจะมีความสามารถที่จะรับความจุความร้อนสูงกว่า เซนเซอร์แบบ Thermal Conductivity หรือเราอาจะเรียก เซนเซอร์ความชื้นสมบูรณ์ (Absolute Humidity Sensor) ประกอบด้วยเทอร์มิสเตอร์ 2 ตัว ต่ออยู่ในวงจรบริดจ์โดยเทอร์มิสเตอร์ตัวหนึ่งบรรจุอยู่ในแคปซูลที่มีก๊าซไนโตรเจน และเทอร์มิสเตอร์อีกตัวหนึ่งถูกวางอยู่ในบรรยากาศ เซนเซอร์แบบ Thermal Conductivity มีความทนทานสูงและทำงานในสภาพแวดล้อมที่มีอุณหภูมิสูงถึง 300°C และยังทนต่อไอระเหยสารเคมีได้เป็นอย่างดีจากคุณสมบัติที่ดีของวัสดุโครงสร้างเครื่องที่ไม่มีปฏิกิริยาทางสารเคมี

2.4.3 หลักการทำงานของ Arduino Uno R3

Arduino Uno R3 บอร์ดไมโครคอนโทรลเลอร์แบบ Open-source บนแพลตฟอร์ม Arduino จากผู้ผลิต arduino.cc ประเทศอิตาลี ออกแบบมาให้ใช้งานได้ง่าย ใช้ชิพ ATmega328 รันที่ความถี่ 16 MHz หน่วยความจำแฟลช 32 KB แรม 2 KB บอร์ดใช้ไฟเลี้ยง 7 ถึง 12 V มีระดับแรงดันไฟฟ้าในการทำงานและขาสัญญาณอยู่ที่ 5 V (TTL) มี Digital Input / Output 14 ขา (เป็น PWM ได้ 6 ขา) มี Analog Input 6 ขา Serial UART 1 ชุด I2C 1 ชุด SPI 1 ชุด เขียนโปรแกรมบนซอฟท์แวร์

2.4.4 โปรแกรมสำหรับใช้งานกับไมโครคอนโทรลเลอร์

โปรแกรมที่ใช้งานกับไมโครคอนโทรลเลอร์รุ่น Arduino Uno R3 จะใช้โปรแกรม Arduino IDE เขียนโปรแกรมบนเครื่องคอมพิวเตอร์เพื่อนำไปใช้งานกับไมโครคอนโทรลเลอร์ โดยจะใช้ภาษาซีในการเขียนโปรแกรม โดยโปรแกรม Arduino จะจัดแบ่งรูปแบบโครงสร้างของการเขียนโปรแกรมเป็นส่วนย่อยๆหลายๆส่วน โดยเรียกแต่ละส่วนว่า "ฟังก์ชั่น" และเมื่อนำฟังก์ชันมารวมเข้าด้วยกัน ก็จะเรียกว่า"โปรแกรม" โดยโครงสร้างการเขียนโปรแกรมของ Arduino ทุกๆโปรแกรมจะประกอบไปด้วยฟังก์ชั่นจำนวนเท่าใดก็ได้ แต่อย่างน้อยสุดต้องมี 2 ฟังก์ชั่นคือ setup() และ loop()

1) Setup() : เป็นฟังก์ชั่นบังคับที่ต้องกำหนดให้มีทุกๆโปรแกรม แม้ว่าบางโปรแกรมจะไม่ต้องการใช้งานแต่ก็ยังจำเป็นต้องประกาศไว้เสมอ เพียงแต่ไม่ต้องเขียนคำสั่งใดๆไว้หลังวงเล็บปีกกา { } ที่ใช้เป็นตัวกำหนดขอบเขตของฟังก์ชั่น โดยฟังก์ชันนี้จะใช้สำหรับบรรจุคำสั่งที่ต้องการให้โปรแกรมทำงานเพียงรอบเดียว ตอนเริ่มต้นทำงานของโปรแกรมครั้งแรกเท่านั้น ซึ่งได้แก่คำสั่งเกี่ยวกับการ Setup ค่าการทำงานต่างๆ เช่นการกำหนดหน้าที่ของการใช้งานของ PinMode และค่า Baudrate สำหรับการใช้งานสื่อสารพอร์ตอนุกรม เป็นต้น

2) Loop() : เป็นส่วนฟังก์ชั่นบังคับที่ต้องกำหนดให้มีในทุกโปรแกรม เช่นเดียวกันกับฟังก์ชั่น Setup() โดยฟังก์ชั่น Loop() นี้จะใช้ในการบรรจุคำสั่งที่ต้องการให้โปรแกรมทำงานเป็นวนรอบซ้ำๆ

2.5 WIRELESS LAN (WLAN)

Wireless LAN หรือ Wi-Fi คือระบบสื่อสารข้อมูลที่ส่งสัญญาณผ่านอากาศโดยใช้คลื่นวิทยุหรือคลื่นแม่เหล็กไฟฟ้า โดยที่ไม่ต้องใช้สายสัญญาณในการนำสัญญาณ ทำให้เครือข่าย สะดวกต่อการติดตั้งและใช้ จะใช้คลื่นวิทยุ (Radio Frequency) เป็นสัญญาณ และใช้อากาศเป็นตัวนำสัญญาณการเชื่อมต่อหรือสื่อสารกัน การเชื่อมต่อแลนไร้สายมีทั้งแบบเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์ด้วยกัน และเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์ผ่านอุปกรณ์กระจายสัญญาณ (Access Point) เคลื่อนย้าย และความง่ายต่อการขยายเครือข่าย

2.5.1 มาตรฐาน Wireless LAN IEEE 802.11

สถาบันวิศวกรรมไฟฟ้าและอิเลคทรอนิกส์ (Institute of Electrical and Electronic Engineers) ได้กำหนดมาตรฐานแรกของระบบเครือข่ายไร้สาย โดยใช้ชื่อ IEEE802.11 แล้วมีการพัฒนาอย่างต่อเนื่องและขยายออกไปเป็นมาตรฐานต่าง ๆ ดังนี้

1) มาตรฐาน IEEE 802.11a

เป็นมาตราฐานเครือข่ายไวร์เลสแลนที่มีผู้ใช้มากที่สุด เพราะใช้ความถี่ย่าน 2.4 กิกะเฮิรตซ์ ซึ่งเป็นความถี่ที่ทุกประเทศเปิดให้ใช้โดยอิสระ และการที่ถูกประกาศเป็นมาตราฐานตั้งแต่ปี ค.ศ.1999 จึงมีผู้รู้จักมากกว่ามาตราฐานอื่น ๆ มีความเร็วในการทำงาน 4 ระดับ คือ 1, 2, 5.5 และ 11 เมกะบิตต่อวินาที เครือข่ายที่ให้บริการอินเทอร์เน็ตไร้สายแบบ Hot spot ส่วนใหญ่จะใช้มาตราฐาน แต่มีความเร็วต่ำที่สุดในบรรดามาตราฐานต่าง ๆ และมีปัญหาสัญญาณรบกวนค่อนข้างสูง เนื่องจากช่วงความถี่ย่าน 2.4 กิกะเฮิรตซ์นี้มีอุปกรณ์ใช้งานอยู่มาก ไม่ว่าจะเป็นโทรศัพท์ไร้สาย เตาไมโครเวฟและบลูทูธ เป็นต้น

2) มาตรฐาน IEEE 802.11b

เป็นมาตราฐานที่เปิดตัวมาพร้อมกับมาตราฐาน 802.11b ตั้งแต่ปี ค.ศ. 1999 แต่ไม่ค่อยเป็นที่นิยม เนื่องจากใช้ความถี่ย่าน 5 กิกะเฮิรตซ์ ซึ่งเป็นย่านที่ไม่ได้เปิดให้ใช้อย่างเสรีในทุก ๆ ประเทศ ในประเทศไทยย่านความถี่นี้ก็ไม่เปิดให้ใช้เช่นกัน อุปกรณ์เครือข่ายมาตราฐานนี้จึงไม่มีจำหน่ายในเมืองไทย ข้อดีของมาตราฐานนี้ก็คือ มีความเร็วในการทำงานสูง 54 เมกะบิต แต่ก็มีข้อเสียคือ ใช้งานได้ไกลสุด 50 เมตร และนอกจากนี้ยังไม่สามารถทำงานร่วมกับอุปกรณ์ในมาตราฐาน 802.11b และ 802.11g ได้เนื่องจากความถี่ต่างกัน

3) มาตรฐาน IEEE 802.11g

เป็นมาตราฐานเครือข่ายไวร์เลสแลน ประกาศใช้มาตราฐานเมื่อปี ค.ศ. 2003 มีข้อดีกว่ามาตราฐานอื่น คือ มีความเร็วในการทำงานสูงถึง 54 เมกะบิตและมีระยะทำงานไกลสุดเท่ากับมาตราฐาน 802.11b นอกจากนี้ยังใช้ย่านความถี่ 2.4 กิกะเฮิรตซ์ซึ่งเป็นช่วงความถี่ที่ใช้งานได้เสรีและสามารถใช้งานกับมาตราฐาน 802.11b

4) มาตรฐาน IEEE 802.11n

เสร็จสมบูรณ์ในปี ค.ศ. 2009 ทำงานบนย่านความถี่ 2.4 และ 5 GHz โดยที่สามารถให้อัตราการส่งถ่ายข้อมูลสูงสุดถึง 300 Mbps มีความสามารถในการส่งคลื่นสัญญาณ ได้ระยะประมาณ 70 เมตรในโครงสร้างปิด และ 250 เมตรในที่โล่งแจ้ง เพิ่มความสามารถในการกัน

สัญญาณกวนจากอุปกรณ์อื่นๆ ที่ใช้ความถี่ 2.4GHz เหมือนกัน และสามารถรองรับอุปกรณ์มาตรฐาน IEEE 802.11b และ IEEE 802.11g ได้

5) มาตรฐาน IEEE 802.11ac

802.11ac เป็นมาตรฐานใหม่ของระบบเครือข่ายไร้สาย (WiFi) ที่กำลังจะเกิดขึ้นเร็วๆนี้ โดยการมอดูเลต(Modulation) แบบใหม่ ปรับช่องสัญญาณให้มีขนาดใหญ่ขึ้นและรองรับการส่งข้อมูลพร้อมกันถึง 8 ชุด ยิ่งเสามากก็ยิ่งแรงขึ้นและยังมีการเพิ่ม bandwidth ขึ้นมาเป็น 80 กับ 160 MHz อีก ส่งผลให้ 802.11ac ทำความเร็วสูงสุดได้ที่ 6.93 Gb/s แต่จะต้องอยู่บนเงื่อนไขที่ใช้อุปกรณ์แบบสุดๆ ด้วยเช่นกัน กล่าวคือ AP ตัวส่งสัญญาณจะต้องมี 8 เสาและตัวรับสัญญาณจะต้องมี 2 เสา 4 ชุด ซึ่งในการใช้งานทั่วไปนั้นจะมีเป็นแบบตัวรับและตัวส่ง 2 เสาเป็นส่วนใหญ่ ทำให้ความเร็ว 802.11ac วิ่งได้ไม่เต็มที่

ตารางที่ 2.3 คุณสมบัติของมาตรฐาน IEEE ชนิดต่างๆ

มาตรฐาน	ความถี่ที่ใช้ (GHz)	Bandwidth (MHz)	อัตราการส่งข้อมูลสูงสุด (Mbps)	Modulation
IEEE 802.11	2.4	20	2	DSSS,FHSS
IEEE 802.11a	5	20	54	OFDM
IEEE 802.11b	2.4	20	11	DSSS
IEEE 802.11g	2.4	20	54	OFDM,DSSS
IEEE 802.11n	5	20/40	600	OFDM
IEEE 802.11ac	5	80/160	6930	256-QAM,OFDM

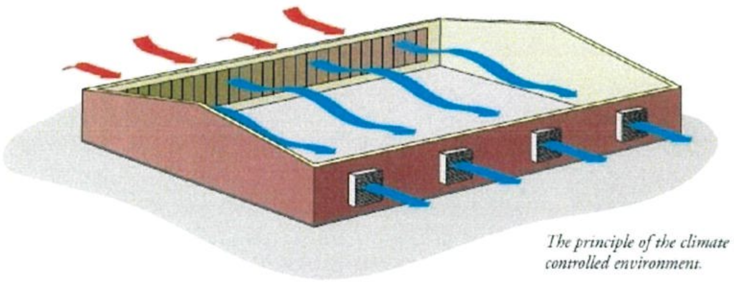
2.6 ประเภทของโรงเรือน

2.6.1โรงเรือนระบบปิด (evaporative cooling system Evap)

หมายถึง โรงเรือนที่ควบคุมสภาพแวดล้อมภายในแสดงดังรูปที่ 2.6 เหมาะสมสำหรับการเลี้ยงไก่ ผนังโรงเรือนจะปิดทึบ และบังคับอากาศที่จะผ่านเข้าโรงเรือนให้ผ่านแผ่นทำความเย็นที่มีน้ำไหล ผ่านทำให้อุณหภูมิของร่างกายลดต่ำลงและมีระบบดูดอากาศออกภายนอกโรงเรือน

2.6.1.1 ระบบ Evaporative Cooling System (EVAP)

เป็นระบบทำความเย็นที่ทันสมัย เหมาะสำหรับ การปรับอากาศในพื้นที่โล่ง อาคารขนาดใหญ่ที่มีแหล่งกำเนิดความร้อนสูง สถานที่ที่มีคนเป็นจำนวนมาก ศูนย์สรรพสินค้า ศูนย์กีฬาในร่ม งานแสดงสินค้า นิทรรศการ รวมทั้งอุตสาหกรรมการเกษตร เช่น โรงเรือนปศุสัตว์ และโรงเรือนเพาะชำ ระบบEVAP เป็นระบบที่ใช้อากาศบริสุทธิ์ 100% และไม่มีละอองน้ำปะปน จึงมีความปลอดภัยต่อสุขอนามัย และสามารถควบคุมอุณหภูมิและความชื้นในพื้นที่ที่ต้องการได้ดี



รูปที่ 2.6 โรงเรือนแบบปิด

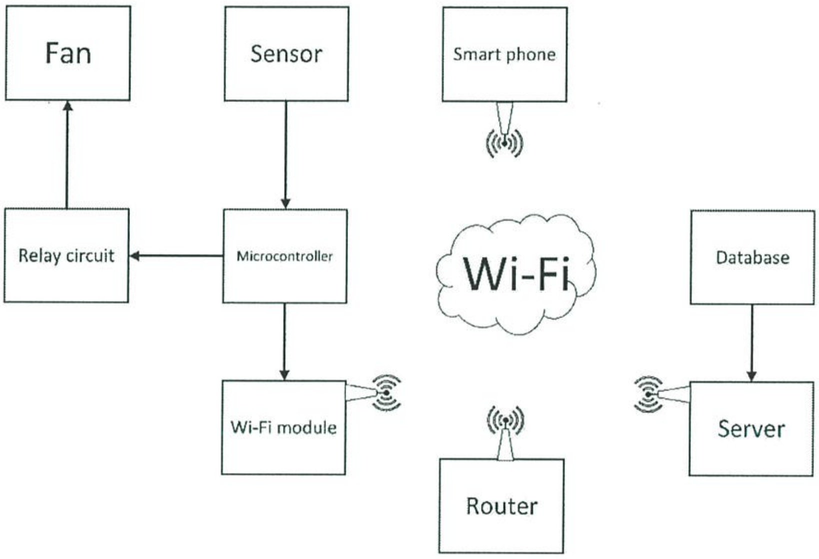
2.6.2. โรงเรือนระบบเปิด (open house)

หมายถึง โรงเรือนที่อากาศเข้าออกในโรงเรือนได้ สภาพแวดล้อมในโรงเรือนจะเปลี่ยนแปลงตามสภาพภูมิอากาศภายนอก

บทที่ 3
การออกแบบและการจัดทำปริญญานิพนธ์

3.1 การออกแบบ

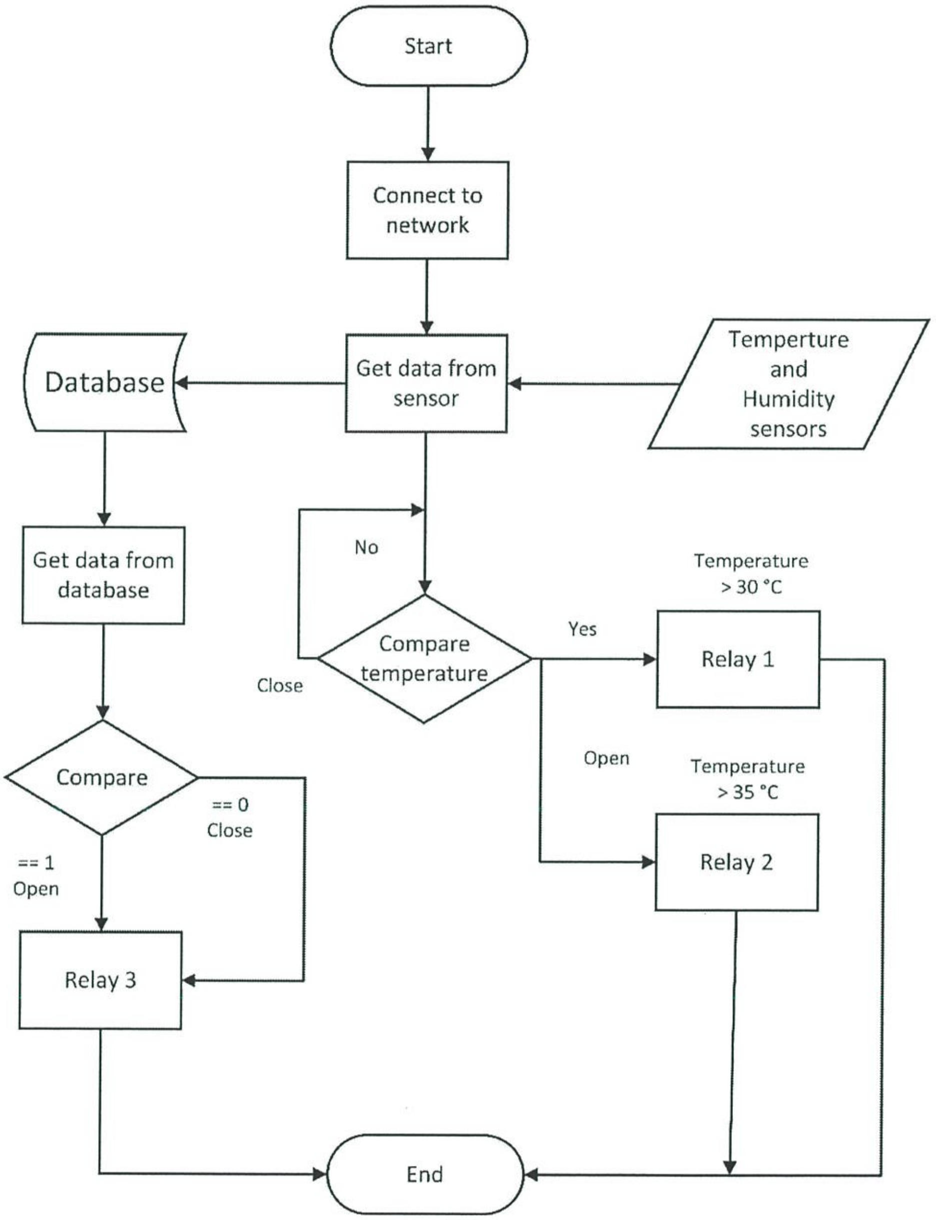
การออกแบบและการจัดทำปริญญานิพนธ์ จะเริ่มต้นด้วยการวางแผนการจัดทำโครงงาน ออกแบบการทำงานของระบบดังรูปที่ 3.1



รูปที่ 3.1 บล็อกไดอะแกรมของระบบ

โดยการทำงานของระบบจะแบ่งการออกแบบเป็นสามส่วน คือ ส่วนไมโครคอนโทรลเลอร์ ส่วนแอพพลิเคชั่นและส่วนของฐานข้อมูล จากรูปที่ 3.1 สามารถอธิบายการทำงานของระบบ เริ่มจากฝั่งของไมโครคอลโทรลเลอร์มีเซนเซอร์สำหรับวัดอุณหภูมิและความชื้นสัมพัทธ์ต่ออยู่กับไมโครคอนโทรลเลอร์ มีวงจรรีเลย์ต่ออยู่เพื่อควบคุมพัดลมและต่อกับ Wi-Fi module เพื่อใช้การติดต่อสื่อสารแบบไร้สายกับเซิร์ฟเวอร์ โดยมีเราเตอร์เป็นตัวปล่อยสัญญาณ Wi-Fi ซึ่งที่เซิร์ฟเวอร์จะมีระบบฐานข้อมูลซึ่งไว้ใช้เก็บข้อมูลต่างๆและมีแอพพลิเคชั่นที่ใช้บนระบบปฏิบัติการแอนดรอยด์ออกแบบให้สามารถติดต่อกับฐานข้อมูลและไมโครคอนโทรลเลอร์ได้โดยผ่านเครือข่ายไร้สาย

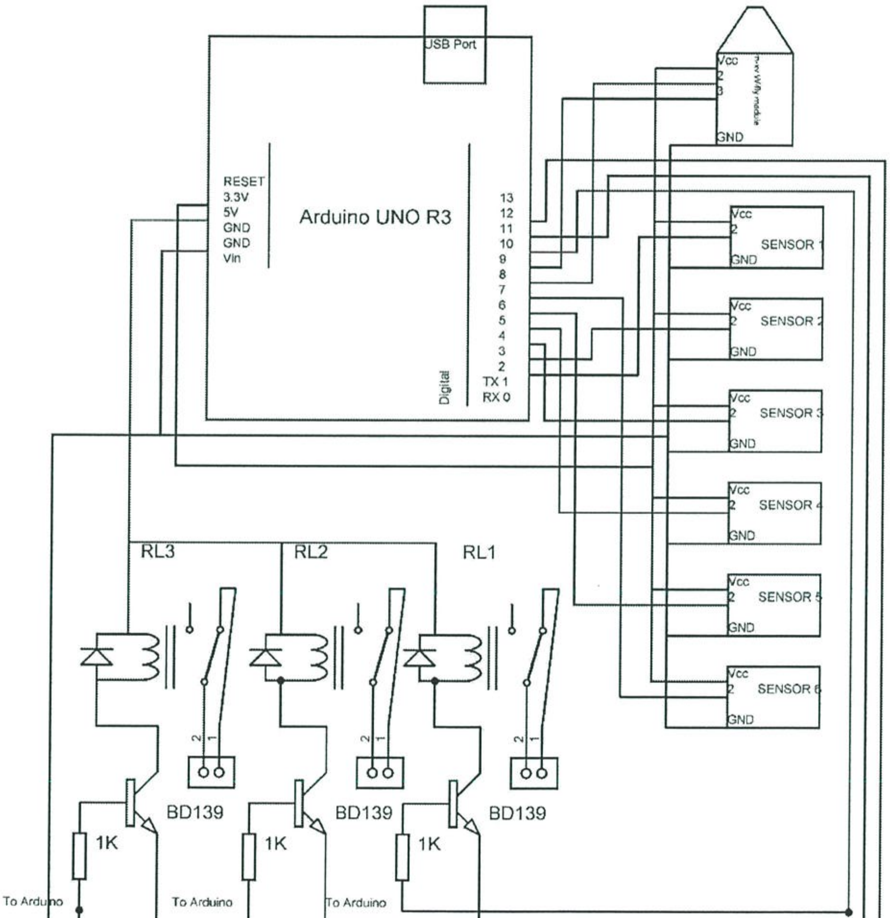
3.1.1 การออกแบบการทำงานของไมโครคอนโทรลเลอร์



รูปที่ 3.2 แผนผังการทำงานของไมโครคอนโทรลเลอร์

การออกแบบการทำงานให้กับไมโครคอนโทรลเลอร์ดังรูปที่ 3.2 ซึ่งแสดงกระบวนการทำงานของไมโครคอนโทรลเลอร์ ได้มีการออกแบบฟังก์ชั่นสำหรับการทำงานดังนี้ ซึ่งวงจรทั้งหมด

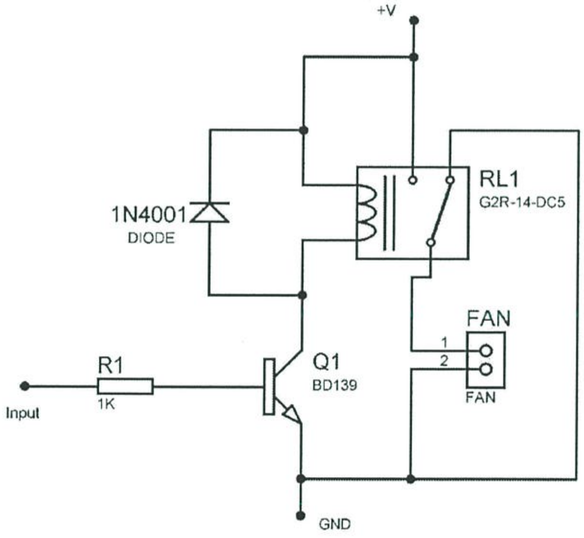
ของระบบ ที่มีไมโครคอนโทรลเลอร์ WiFi-module เซนเซอร์และวงจรรีเลย์แสดงได้ดังรูปที่ 3.3 ซึ่งจากรูป WiFi midule วงจรรีเลย์และเซนเซอร์ จะถูกต่อเขากับขาดิจิตอลของไมโครคอนโทรลเลอร์ โดย WiFi module และเซนเซอร์จะใช้ไฟเลี้ยงจากไมโครคอนโทรลเลอร์ 3.3 โวลต์ และวงจรรีเลย์จะใช้ไฟเลี้ยงจากไมโครคอนโทรลเลอร์ 5 โวลต์



รูปที่ 3.3 วงจรรวมของระบบ

3.1.1.1 การควบคุมการทำงานของพัดลมตามค่าอุณหภูมิที่ตั้งไว้

ฟังก์ชันนี้จะมีการควบคุมการทำงานพัดลมที่ต่ออยู่กับวงจรรีเลย์ ดังรูปที่ 3.4 ซึ่งต่ออยู่กับพัดลม การทำงานของพัดลมว่าจะเปิดหรือปิดจะขึ้นอยู่กับการตั้งค่าของอุณหภูมิให้กับไมโครคอนโทรลเลอร์ โดยจะอ้างอิงอุณหภูมิที่กำหนดตามค่าอุณหภูมิที่เหมาะสมสำหรับโรงเรือนประเภทต่างๆ ในการทดลองนี้จะกำหนดอุณหภูมิที่เหมาะสมที่ไม่ต้องการให้เกินอยู่ที่ 30 ถึง 35องศาเซลเซียส โดยกำหนดอุณหภูมิที่ต้องการ Temp1 = 30 องศาเซลเซียส และ Temp2 = 35 องศาเซลเซียส ให้กับไมโครคอนโทรลเลอร์ โดยเมื่อถึงค่าอุณหภูมิที่ตั้งค่าไว้ ไมโครคอนโทรลเลอร์จะไปสั่งให้วงจรรีเลย์ทำงานและเปิดพัดลม แต่ในทางกลับกัน ถ้าอุณหภูมิไม่ถึงค่าที่กำหนดไว้พัดลมก็จะไม่ถูกสั่งให้ทำงาน ซึ่งค่าอุณหภูมิจะได้ค่าจากเซนเซอร์วัดอุณหภูมิและความชื้นสัมพัทธ์ ซึ่งต่ออยู่กับไมโครคอนโทรลเลอร์



รูปที่ 3.4 วงจรรีเลย์

3.1.1.2 ควบคุมการทำงานของพัดลมโดยผ่านแอพพลิเคชั่นที่ใช้ระบบปฏิบัติการแอนดรอยด์

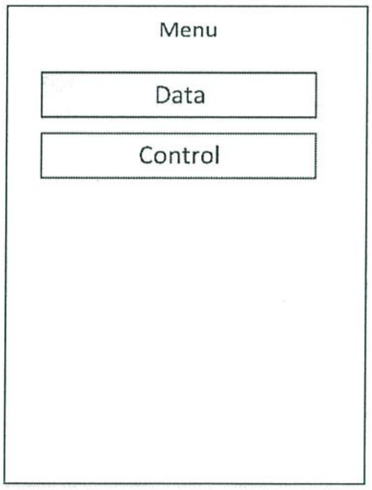
ฟังก์ชันนี้จะควบคุมการทำงานของพัดลมที่ต่ออยู่กับวงจรรีเลย์ดังรูปที่ 3.4 ซึ่งต่ออยู่กับพัดลม ซึ่งจะใช้แอพพลิเคชั่นที่ออกแบบให้สามารถควบคุมการทำงานของไมโครคอนโทรลเลอร์ผ่านเครือข่ายไร้สาย โดยไมโครคอนโทรลเลอร์จะไปตรวจสอบค่าในฐานข้อมูลที่เซิร์ฟเวอร์ที่ถูกส่งมาจากโทรศัพท์มือถือผ่านแอพพลิเคชั่น

3.1.1.3 ไมโครคอนโทรลเลอร์ส่งค่าของอุณหภูมิและความชื้นสัมพัทธ์ไปยังฐานข้อมูลที่เซิร์ฟเวอร์

ฟังก์ชันนี้ไมโครคอนโทรลเลอร์จะรับค่าจากเซนเซอร์ที่วัดค่าอุณหภูมิและความชื้นสัมพัทธ์และส่งไปยังฐานข้อมูลที่เซิร์ฟเวอร์ ผ่านเครือข่ายไร้สาย

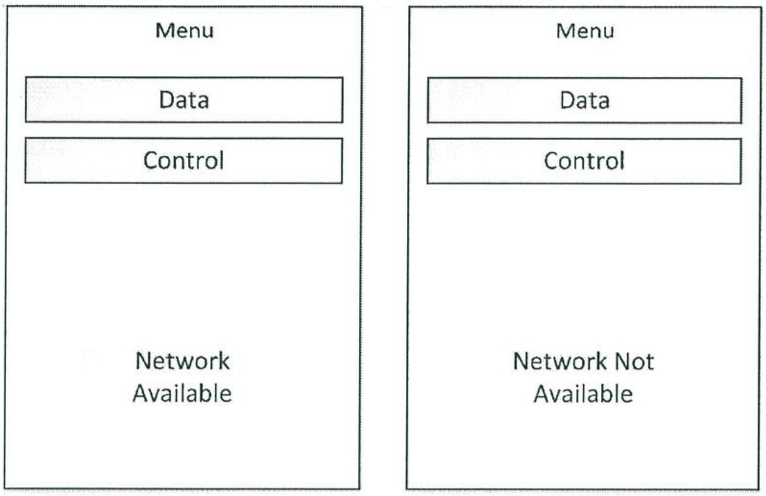
3.1.2 การออกแบบแอพพลิเคชั่น

การออกแบบการทำงานให้กับแอพพลิเคชั่นเพื่อให้สามารถทำงานร่วมกับไมโครคอนโทรลเลอร์และฐานข้อมูล โดยจะออกแบบหน้าจอแอพพลิเคชั่น ให้ผู้ใช้งานสามารถใช้งานได้สะดวก หน้าของแอพพลิเคชั่นสามารถแสดงได้ดังนี้



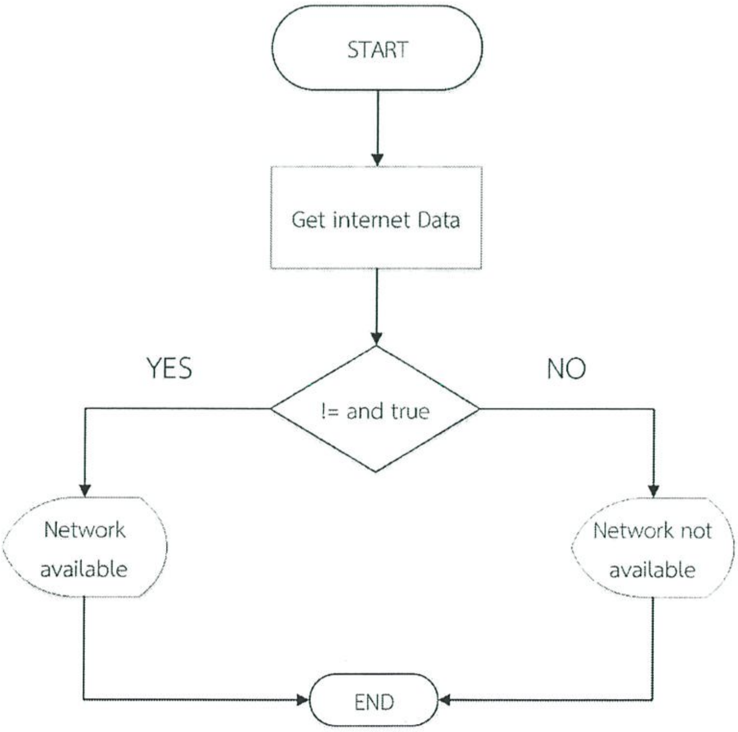
รูปที่ 3.5 หน้าเมนูหลักของแอพพลิเคชั่น

เมื่อผู้ใช้งานติดตั้งแอพพลิเคชั่น และกดใช้งาน ผู้ใช้งานจะเจอกับหน้าแรกของแอพพลิเคชั่น ที่จะให้ผู้ใช้ทำการเลือก โดยมีปุ่ม Data เพื่อดูข้อมูลที่เก็บไว้ในฐานข้อมูลที่เซิร์ฟเวอร์ และมีปุ่ม Control ไว้ใช้เพื่อกดไปยังหน้าล็อกอิน เพื่อสำหรับควบคุมจงจร เปิดปิดพัดลม ดังรูปที่ 3.5 แอพพลิเคชั่นจะทำการเช็คสถานะของเครือข่ายที่โทรศัพท์ต่ออยู่ หากโทรศัพท์เชื่อมต่อเครือข่ายอยู่จะขึ้นสถานะ Network Available แต่ถ้าหากไม่ได้เชื่อมต่อจะขึ้นสถานะ Network Not Available ดังรูปที่ 3.6



รูปที่ 3.6 สถานะการเชื่อมต่อเครือข่ายของโทรศัพท์มือถือ

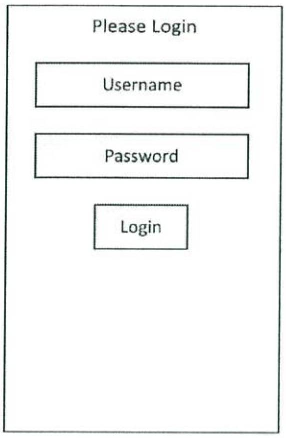
จากการทำงานของแอพพลิเคชั่นสามารถแสดงแผนผังการทำงานได้ดังรูปที่ 3.7



รูปที่ 3.7 แผนผังการทำงานการตรวจสอบสถานะการเชื่อมต่อ

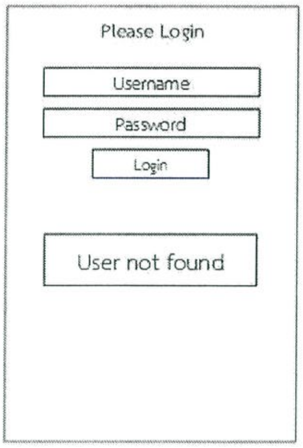
สามารถแบ่งฟังก์ชันการทำงานได้ดังนี้

3.1.2.1 หน้าล็อกอินสำหรับผู้ใช้งานเพื่อที่จะเข้าไปควบคุมการเปิดปิดของพัดลม

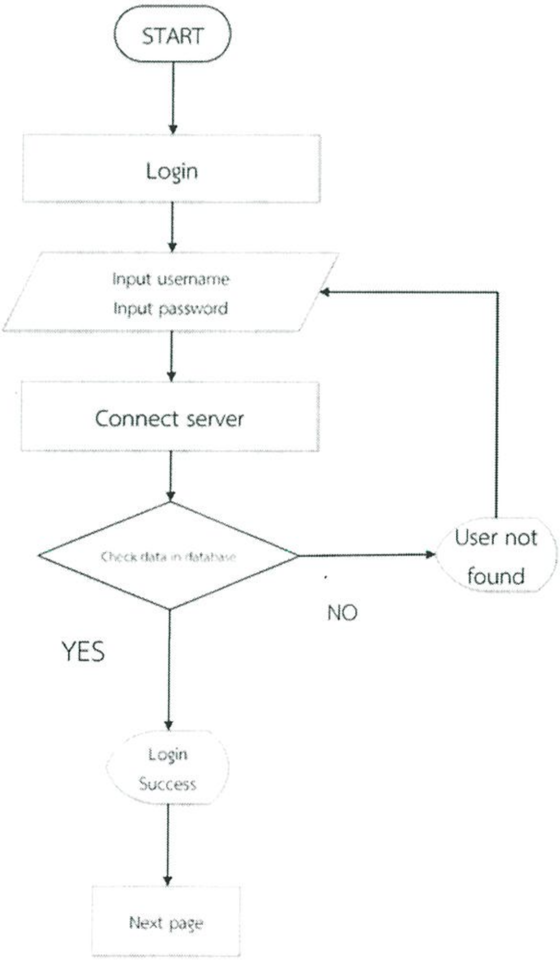


รูปที่ 3.8 หน้าสำหรับเข้าสู่ระบบ (Login) ของแอพพลิเคชั่น

เมื่อผู้ใช้เปิดใช้งานแอพพลิเคชั่นแล้ว จะเปิดกับหน้าเมนู เพื่อให้เลือกการทำงาน เมื่อผู้ใช้เลือกที่ปุ่ม Control ดังรูปที่ 3.5 แอพพลิเคชั่นจะไปเปิดหน้าล็อกอินดังรูปที่ 3.8 เพื่อให้ผู้ใช้ป้อนชื่อผู้ใช้ (Username) และรหัสผ่าน (Password) จากรูปที่ 3.8 ผู้ใช้จะต้องป้อนชื่อผู้ใช้ในช่อง Username และรหัสผ่านในช่อง Password จากนั้นกดปุ่ม Login แอพพลิเคชั่นจะรับค่าที่ผู้ใช้งานป้อนเข้ามา แล้วนำไปตรวจสอบกับชื่อผู้ใช้งานและรหัสผ่านที่เก็บอยู่ในฐานข้อมูลที่เซิร์ฟเวอร์ ถ้าหากตรวจไม่พบข้อมูลที่ตรงกับข้อมูลในฐานข้อมูลแอพพลิเคชั่นจะแจ้งเตือนว่า User not found ดังรูปที่ 3.9 จากรูปที่ 3.9 เมื่อผู้ใช้งานป้อนชื่อผู้ใช้งานและรหัสผ่าน หากข้อมูลที่ป้อนไม่ตรงกับข้อมูลในฐานข้อมูล แอพพลิเคชั่นจะขึ้นแจ้งเตือน ซึ่งสามารถแสดงผังการทำงานได้ดังรูปที่ 3.10 จากรูปที่ 3.10 ชื่อผู้ใช้และรหัสผ่านที่ถูกป้อนจากผู้ใช้จะถูกตรวจสอบกับฐานข้อมูลที่เซิร์ฟเวอร์และหากถูกต้องก็จะสามารถเข้าสู่ระบบได้ แต่หากไม่ถูกต้องจะต้องทำการป้อนข้อมูลใหม่

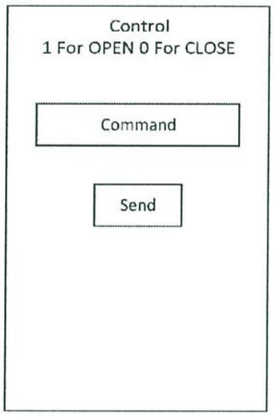


รูปที่ 3.9 แอพพลิเคชั่นแจ้งเตือนเมื่อไม่พบข้อมูล



รูปที่ 3.10 แผนผังการทำงานหน้าล็อกอินของแอพพลิเคชั่น

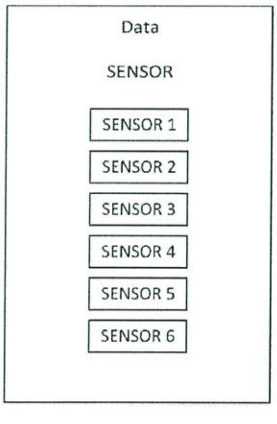
หากชื่อผู้ใช้งานและรหัสผ่านที่ผู้ใช้งานป้อนตรงกับข้อมูลที่มีอยู่ในฐานข้อมูลแอพพลิเคชั่น จะเปิดหน้าเพื่อใช้สำหรับควบคุมการเปิดปิดพัดลม ดังรูปที่ 3.11



รูปที่ 3.11 ล็อกอินและไปยังหน้าควบคุมของแอพพลิเคชั่น

3.1.2.2 ออกแบบหน้าแสดงผลค่าข้อมูล

เมื่อผู้ใช้งานเลือกปุ่ม Data ที่หน้าเมนูหลักของแอพพลิเคชั่น แอพพลิเคชั่นจะเปิดไปยังหน้าที่ให้ผู้ใช้เลือกดูข้อมูล ดังรูปที่ 3.12



รูปที่ 3.12 แอพพลิเคชั่นเปิดไปยังหน้าData

จากรูปที่ 3.12 เมื่ออยู่ในหน้านี้ผู้ใช้งานจะสามารถเลือกดูข้อมูลที่ไมโครคอนโทรลเลอร์ส่งค่าจากเซนเซอร์วัดอุณหภูมิและความชื้นสัมพัทธ์เข้ามาเก็บไว้ในฐานข้อมูลที่เซิร์ฟเวอร์

3.1.3 การออกแบบฐานข้อมูล (Database)

ที่เซิร์ฟเวอร์จะมีฐานข้อมูลเพื่อใช้เก็บข้อมูลที่จะถูกส่งมาจากแอพพลิเคชั่นบนโทรศัพท์มือถือและค่าข้อมูลจากไมโครคอนโทรลเลอร์ โดยจะสร้างฐานข้อมูลมาหนึ่งฐานข้อมูลและสร้างตาราง(Table)เพื่อใช้เก็บข้อมูล มีรายละเอียดตารางของฐานข้อมูลที่ต้องสร้างดังนี้

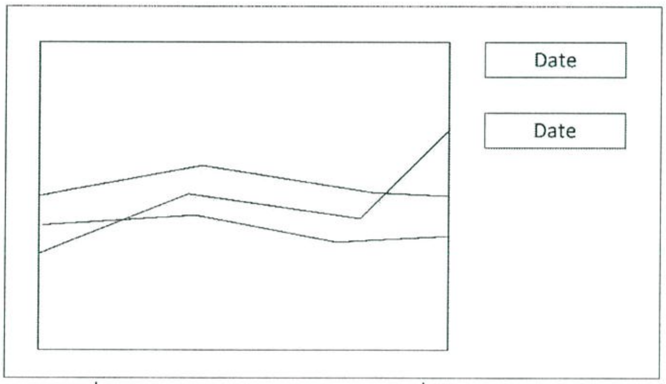
1) ตารางเพื่อใช้เก็บชื่อผู้ใช้และรหัสผ่าน

2) ตารางเพื่อใช้เก็บคำสั่งที่ส่งมาจากแอพพลิเคชั่นบนโทรศัพท์มือถือ

3) ตารางเพื่อใช้รับค่าข้อมูลที่ส่งมาจากไมโครคอนโทรลเลอร์

3.1.4 การออกแบบโปรแกรมที่ใช้แสดงผลข้อมูลจากฐานข้อมูล

ออกแบบโปรแกรมโดยใช้โปรแกรม visual studio โดยออกแบบให้โปรแกรมเชื่อมต่อกับฐานข้อมูลภายในเครื่องเซิร์ฟเวอร์ แล้วให้นำค่าอุณหภูมิและความชื้นสัมพัทธ์มาพล็อตกราฟเทียบกับเวลา โดยออกแบบให้ผู้ใช้งานสามารถเลือกช่วงวันที่ที่ต้องการจะดูข้อมูล รูปแบบของโปรแกรมแสดงดังรูปที่ 3.13 จากรูปที่ 3.13 จะมีส่วนที่แสดงกราฟโดยที่นำข้อมูลจากฐานข้อมูลมาแสดงผล และจะมีส่วนปุ่มสำหรับให้ผู้ใช้งานเลือกช่วงวันที่ต้องการ



รูปที่ 3.13 รูปแบบของโปรแกรมที่ใช้แสดงผลข้อมูล

3.2 เครื่องมือที่ใช้ในการทดลอง

เครื่องมือและอุปกรณ์ที่ใช้ในการพัฒนาแอพพลิเคชั่นและการจัดการฐานข้อมูล มีดังนี้

3.2.1 อุปกรณ์ด้านฮาร์ดแวร์

1) เครื่องคอมพิวเตอร์ (Computer)

เพื่อใช้เป็นเครื่องมือสำหรับเขียนโปรแกรมและใช้เป็นเซิร์ฟเวอร์และติดตั้งระบบฐานข้อมูล

2) โทรศัพท์มือถือที่ใช้ระบบปฏิบัติการแอนดรอยด์

เพื่อใช้เป็นเครื่องมือสำหรับทดสอบการทำงานของแอพพลิเคชั่นและส่งคำสั่งไปเก็บที่ฐานข้อมูล

3) เราเตอร์

เพื่อใช้เป็นตัวปล่อยสัญญาณ Wi-Fi และเชื่อมต่อกับเซิร์ฟเวอร์ เพื่อให้ไมโครคอนโทรลเลอร์และโทรศัพท์มือถือเชื่อมต่อ

4) ไมโครคอนโทรลเลอร์

เพื่อใช้เป็นตัวควบคุมการทำงานของวงจรรีเลย์

5) WiFi module

ใช้ต่อกับไมโครคอนโทรลเลอร์เพื่อให้ต่อกับเครือข่าย Wi-Fi

6) พัดลม

7) วงจรรีเลย์

เพื่อใช้ควบคุมการทำงานของพัดลมโดยควบคุมการทำงานด้วยไมโครคอนโทรลเลอร์

3.2.2 อุปกรณ์ด้านซอร์ฟแวร์

โปรแกรมที่ต้องใช้ในการพัฒนาแอพพลิเคชั่นและเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์

1) JDK (Java Development Kit)

2) Eclipse

3) Android SDK

4) Android Development Tools (ADT) plug-in

5) mysql

6) Appserv

7) Arduino

8) Visual studio

3.3 การจัดเก็บผลการทดลอง

ในการจัดเก็บผลการทดลองเพื่อทดสอบการทำงานของแอพพลิเคชั่นจะใช้เครื่องคอมพิวเตอร์และโทรศัพท์มือถือที่ใช้ระบบปฏิบัติการแอนดรอยด์ในการทดสอบการทำงานของแอพพลิเคชั่น

3.3.1 การทำงานของแอพพลิเคชั่น

การทดสอบการทำงานของแอพพลิเคชั่น จะนำแอพพลิเคชั่นไปติดตั้งบนโทรศัพท์มือถือที่ใช้ระบบปฏิบัติการแอนดรอยด์ และทำการเปิดแอพพลิเคชั่น ทดสอบและ ใช้งานแล้วบันทึกหน้าจอขณะที่แอพพลิเคชั่นกำลังทำงานเพื่อเก็บผลการทดสอบ

3.3.2 การควบคุมไมโครคอนโทรลเลอร์ผ่านแอพพลิเคชั่น

เปิดใช้งานแอพพลิเคชั่นและส่งค่าคำสั่งผ่านเครือข่ายไปยังฐานข้อมูลและให้ไมโครคอนโทรลเลอร์รับค่ามาจากฐานข้อมูลแล้วควบคุมการทำงานวงจรรีเลย์

3.3.3 เก็บผลการทดลองจากการเชื่อมต่อ Wi-Fi module กับเซิร์ฟเวอร์

ทำการตรวจสอบสถานะการเชื่อมต่อของ Wi-Fi module กับเซิร์ฟเวอร์โดยการร้องขอแพคเก็ต (ping) จากเซิร์ฟเวอร์ไปยัง และการส่งค่าอุณหภูมิและความชื้นสัมพัทธ์จากไมโครคอนโทรลเลอร์ไปยังฐานข้อมูล

3.3.4 เก็บผลการทดลองการวัดค่าอุณหภูมิ

ใช้เซนเซอร์ที่ต่ออยู่ไมโครคอนโทรลเลอร์วัดค่าอุณหภูมิภายในห้องและวัดค่าอุณหภูมิจริงจากเทอร์โมมิเตอร์และเปรียบเทียบผล

3.3.5 การวัดสัญญาณที่เซนเซอร์ส่งค่าไปยังไมโครคอนโทรลเลอร์

วัดสัญญาณที่ขาของเซนเซอร์ที่ทำหน้าที่ส่งข้อมูลเข้าไปยังไมโครคอนโทรลเลอร์

3.3.6 การแสดงผลค่าอุณหภูมิและความชื้นสัมพัทธ์จากฐานข้อมูล

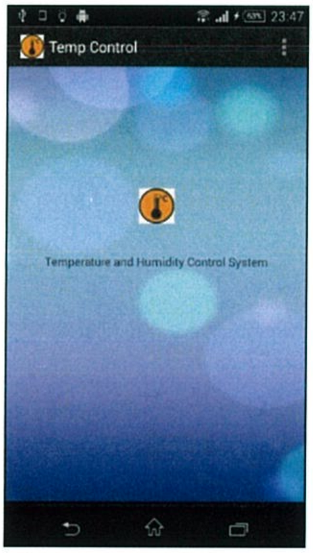
เมื่อข้อมูลถูกส่งมายังฐานข้อมูลจากไมโครคอนโทรลเลอร์ จะใช้โปรแกรมเพื่อใช้แสดงผลข้อมูลที่เครื่องเซิร์ฟเวอร์

บทที่ 4

ผลการทดลอง

4.1 การทำงานของแอพพลิเคชั่น

เมื่อติดตั้งแอพพลิเคชั่น บนโทรศัพท์มือถือที่ใช้ระบบปฏิบัติการณ์แอนดรอยด์แล้ว ทดสอบการทำงานของแอพลิเคชั่นโดยเริ่มเปิดแอพพลิเคชั่น จะเข้าสู่หน้าแรก ดังรูปที่ 4.1 ซึ่งเป็นหน้าแรกของแอพพลิเคชั่นเมื่อเริ่มต้นเปิดใช้งาน

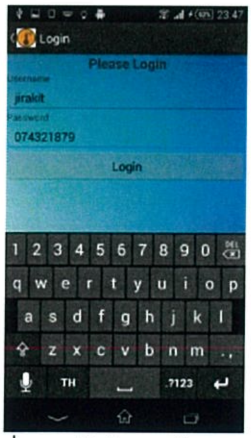


รูปที่ 4.1 หน้าแรกของแอพพลิเคชั่น

เมื่อเริ่มการทำงานแอพพลิเคชั่นจะตรวจสอบสถานะการเชื่อมต่อกับเครือข่ายของโทรศัพท์มือถือ หากโทรศัพท์มือถือมีการเชื่อมต่อกับเครือข่ายแล้วจะแสดงข้อความว่า Network Available ดังรูปที่ 4.2 ซึ่งหมายความว่า ขณะนั้นโทรศัพท์มือถือเชื่อมต่อกับเครือข่ายและสามารถใช้งานแอพพลิเคชั่นต่อไปได้ แต่หากแอพพลิเคชั่นขึ้นข้อความว่า Network not Availableจะหมายถึงโทรศัพท์มือถือยังไม่ได้มีการเชื่อมต่อกับเครือข่าย ผู้ใช้งานจะต้องตั้งค่าการเชื่อมต่อของโทรศัพท์มือถือก่อน

รูปที่ 4.2 แอพพลิเคชั่นที่เชื่อมต่อกับเครือข่ายแล้ว

เมื่อตรวจสอบสถานะการเชื่อมต่อกับเครือข่ายแล้ว ผู้ใช้งานจะต้อง เข้าสู่ระบบโดยการ ล็อคอิน (Login) ดังรูปที่ 4.3 เพื่อเข้าสู่หน้าถัดไปของแอพพลิเคชั่น โดยจากรูปที่ 4.3 ผู้ใช้งานจะต้องพิมพ์ชื่อผู้ใช้งาน (Username) และรหัสผ่าน (Password) ซึ่งทั้งชื่อผู้ใช้งานและรหัสผ่าน จะถูกเก็บไว้ในฐานข้อมูลที่เซิร์ฟเวอร์ ดังรูปที่ 4.4 ซึ่งมีชื่อผู้ใช้และรหัสผ่านเก็บไว้ในฐานข้อมูลเมื่อพิมพ์ชื่อผู้ใช้กับรหัสผ่าน แอพพลิเคชั่นจะตรวจสอบข้อมูลโดยการเชื่อมต่อไปยังฐานข้อมูลที่เซิร์ฟเวอร์ผ่านเครือข่าย หากข้อมูลตรงตามข้อมูลในฐานข้อมูล แอพพลิเคชั่นจะให้เข้าไปยังหน้าถัดไปได้และจะขึ้นข้อความแสดง Login Success ดังรูปที่ 4.5 ซึ่งแอพพลิเคชั่นขึ้นข้อความแสดง



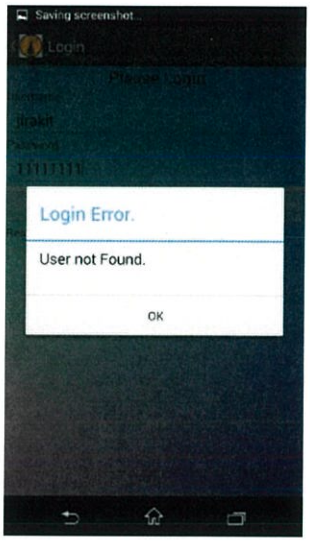
รูปที่ 4.3 ผู้ใช้งานเข้าสู่ระบบ

←T→	id	username	password
	1	jirakit	074321879
	2	yod	12345678
	3	jatupol	12345678
	4	kaew	11111111

รูปที่ 4.4 : ข้อมูลในตารางฐานข้อมูล

หากชื่อผู้ใช้และรหัสผ่านที่พิมพ์ เมื่อแอพพลิเคชั่นตรวจสอบกับฐานข้อมูลแล้วไม่พบชื่อผู้ใช้และรหัสผ่านดังกล่าวในฐานข้อมูล จะไม่สามารถไปยังหน้าถัดไปของแอพพลิชั่นได้และจะขึ้นข้อความเตือน Login Error ดังรูปที่ 4.6 ซึ่งแอพพลิเคชั่นแสดงข้อความเตือน

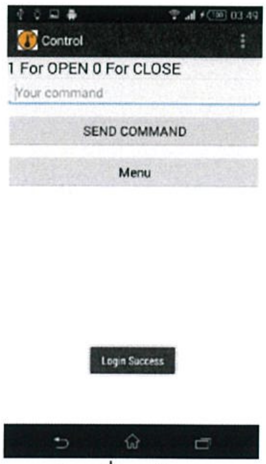
รูปที่ 4.5 แอพพลิเคชั่นขึ้นข้อความแสดง Login Success



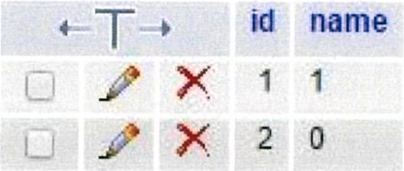
รูปที่ 4.6 แอพพลิเคชั่นขึ้นข้อความแสดง Login Error

4.2 การควบคุมไมโครคอนโทรลเลอร์ผ่านแอพพลิเคชั่น

เมื่อผู้ใช้ล็อกอินโดยป้อนค่าชื่อผู้ใช้และรหัสผ่าน แอพพลิเคชั่นจะเปิดไปยังหน้าควบคุมดังรูปที่ 4.7 เพื่อให้ผู้ใช้ส่งค่าคำสั่งในการเปิด ปิดพัดลม ค่าคำสั่งจะถูกส่งไปเก็บยังฐานข้อมูลดังรูปที่ 4.8 ซึ่งเก็บค่าคำสั่งไว้ในฐานข้อมูล เมื่อไมโครคอนโทรลเลอร์ รับค่าคำสั่งจากฐานข้อมูล หากรับค่าข้อมูลเป็น “1” จะสั่งให้วงจรรีเลย์เปิดให้พัดลมทำงานหากรับค่าเป็น “0” จะสั่งให้ปิดการทำงาน ดังรูปที่ 4.9 ซึ่งแสดงภาพที่พัดลมกำลังหมุนเมื่อสั่งให้พัดลมเปิดผ่านแอพพลิเคชั่น

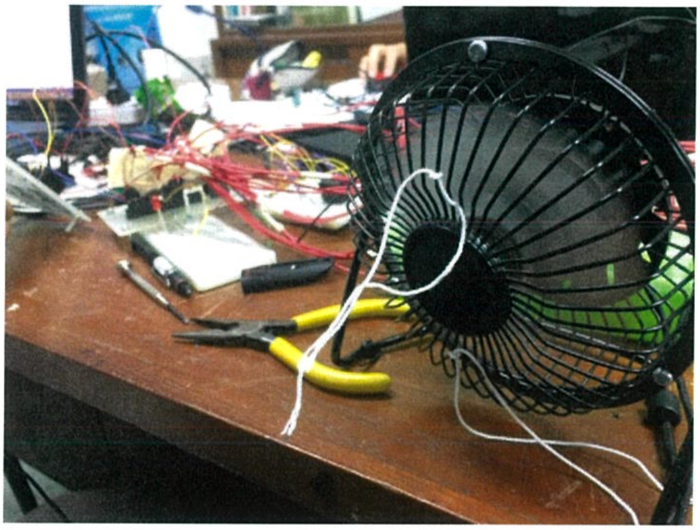


รูปที่ 4.7 แอพพลิเคชั่นหน้าควบคุมการทำงาน



←T→			id	name
			1	1
			2	0

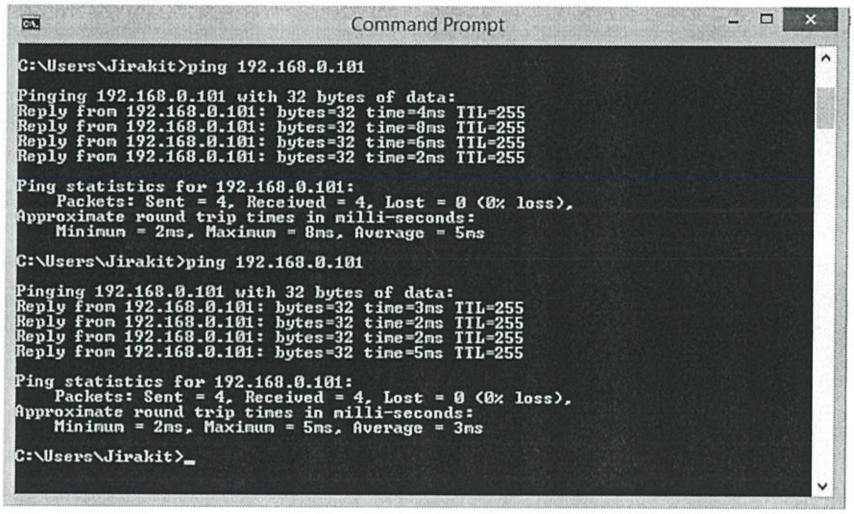
รูปที่ 4.8 ค่าคำสั่งที่ถูกส่งมาเก็บในฐานข้อมูล



รูปที่ 4.9 พัดลมที่ทำงานจากการสั่งงานผ่านแอพพลิเคชั่น

4.3 ผลการทดลองจากการเชื่อมต่อ Wi-Fi module กับเซิร์ฟเวอร์

เมื่อทำการเชื่อมต่อไมโครคอนโทรลเลอร์ด้วย Wi-Fi module บนเครือข่ายเพื่อเชื่อมต่อเข้ากับเซิร์ฟเวอร์ แล้วทำการทดสอบเรียกร้องแพคเก็ต (ping) จากเซิร์ฟเวอร์ไปยังไมโครคอนโทรลเลอร์ดังรูปที่ 4.10 ซึ่งโปรแกรมบนเครื่องคอมพิวเตอร์ แสดงสถานะการเชื่อมต่อเพื่อเช็คสถานะการเชื่อมต่อ



รูปที่ 4.10 การเช็คสถานะการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับเซิร์ฟเวอร์

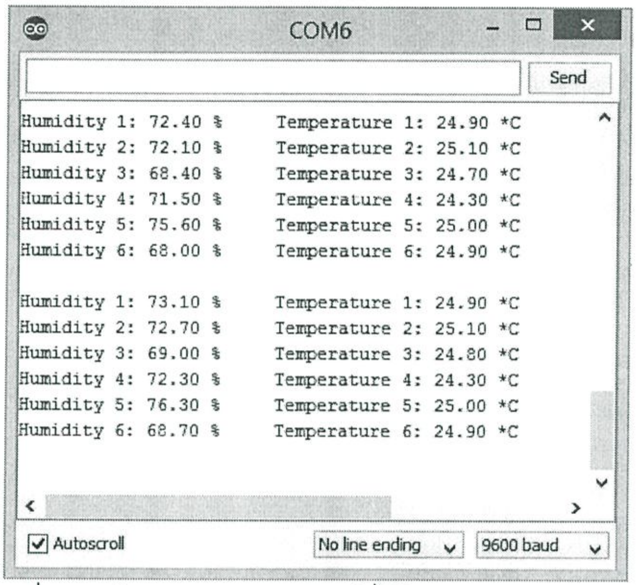
หลังจากนั้นจะทำการส่งค่าข้อมูลค่าอุณหภูมิและค่าความชื่นสัมพัทธ์จากเซนเซอร์ที่ไมโครคอนโทรลเลอร์แล้วส่งค่าไปเก็บยังฐานข้อมูลดังรูปที่ 4.11 ซึ่งแสดงค่าอุณหภูมิและความชื้นสัมพัทธ์ในตารางของฐานข้อมูล

id	humidity1	temperature1	createat	humidity2	temperature2	humidity3	temperature3	humidity4	temperature4
1	59.60	25.10	2015-04-17 05:57:46	64.70	25.20	60.00	25.00	67.20	25.00
2	59.50	25.10	2015-04-17 05:57:51	64.70	25.10	60.00	25.00	67.20	25.00
3	59.60	25.00	2015-04-17 05:57:56	64.80	25.10	60.00	25.00	67.30	25.00
4	59.60	25.00	2015-04-17 05:58:01	65.00	25.10	60.10	25.00	67.60	25.00
5	60.00	25.00	2015-04-17 05:58:06	65.20	25.10	60.40	25.00	67.90	25.00
6	60.20	25.00	2015-04-17 05:58:11	65.50	25.10	60.60	25.00	68.10	25.00
7	60.40	25.00	2015-04-17 05:58:16	65.70	25.10	60.80	25.00	68.40	25.00
8	60.50	25.00	2015-04-17 05:58:21	65.90	25.10	61.00	25.10	68.60	25.00
9	60.70	25.00	2015-04-17 05:58:25	66.00	25.10	61.10	25.00	68.80	25.00
10	60.90	25.00	2015-04-17 05:58:30	66.20	25.10	61.30	25.00	68.90	25.00
11	61.10	25.00	2015-04-17 05:58:35	66.40	25.10	61.60	25.00	69.20	25.00
12	61.20	25.00	2015-04-17 05:58:40	66.50	25.10	61.60	25.00	69.20	25.00
13	61.20	25.00	2015-04-17 05:58:45	66.50	25.10	61.60	25.00	69.30	25.00
14	61.40	25.10	2015-04-17 05:58:50	66.60	25.10	61.80	25.00	69.40	25.00

รูปที่ 4.11 ค่าอุณหภูมิและความชื้นสัมพัทธ์ในฐานข้อมูล

4.4 ผลการทดลองการวัดค่าอุณหภูมิ

วัดค่าอุณหภูมิจากเซนเซอร์ โดยวัดค่าเทียบกับมิเตอร์วัดอุณหภูมิ เปรียบเทียบค่าที่ได้และคำนวณหาค่าเปอร์เซ็นความผิดพลาดของเซนเซอร์ และทดสอบการทำงานของเซนเซอร์ โดยให้ส่งค่าข้อมูล และแสดงผลผ่านโปรแกรมคอมพิวเตอร์ ดังรูปที่ 4.12 ซึ่งโปรแกรมบนคอมพิวเตอร์จะแสดงค่าที่ส่งมาจากไมโครคอนโทรเลอร์เพื่อแสดงผล



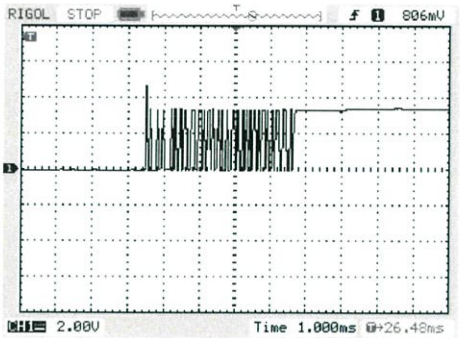
รูปที่ 4.12 โปรแกรมคอมพิวเตอร์ที่ใช้แสดงค่าจากเซนเซอร์

ตารางที่ 4.1 ตารางค่าอุณหภูมิจากเซนเซอร์และจากมิเตอร์วัดอุณหภูมิ (ค่าจริง)

วัดค่าอุณหภูมิภายในห้องขนาด 7x3 เมตร (องศาเซลเซียส)											
ค่าจริง	เซนเซอร์ 1	ค่าจริง	เซนเซอร์ 2	ค่าจริง	เซนเซอร์ 3	ค่าจริง	เซนเซอร์ 4	ค่าจริง	เซนเซอร์ 5	ค่าจริง	เซนเซอร์ 6
24	24.3	24.8	24.6	24.9	24.2	24.8	24.8	24.2	24.4	25	24.8
24	24.3	24.2	24.6	24.6	24.1	23.8	24.8	23.2	24.5	25.4	25.1
24	24.2	22.9	24.3	24.7	24.1	24	24.5	24.3	24.1	25.3	25.1
23.4	24.1	23.4	23.9	24.9	24.3	24.8	24.3	24.4	24.3	24.1	25.2
23.9	23.8	24.1	23.6	25	24.5	25	24.4	24.4	24.4	25	24.9
24.6	23.7	24.3	23.7	24.4	24.7	25.1	24.6	24.7	24.5	25.1	24.9
24.8	23.9	24.1	23.8	23.8	24.6	24.2	24.7	24.7	24.7	25.2	25
24.8	24.3	23.3	23.8	24.2	24.1	24.1	24.6	24.4	24.7	25.4	25.1
24	24.5	23.7	23.6	24.9	24	23.9	24.5	24	24.6	25.5	25.3
23.6	24.2	24.5	23.4	24.8	24.2	24.8	24.3	24.2	24.4	25.4	25.4
24.11	24.13	23.93	23.93	24.62	24.28	24.45	24.55	24.25	24.46	25.14	25.08
ค่าเปอร์เซ็นความผิดพลาด											
0.08 %		0 %		1.38 %		0.4 %		0.85 %		0.24 %	

4.5 การวัดสัญญาณที่เซนเซอร์ส่งค่าไปยังไมโครคอนโทรลเลอร์

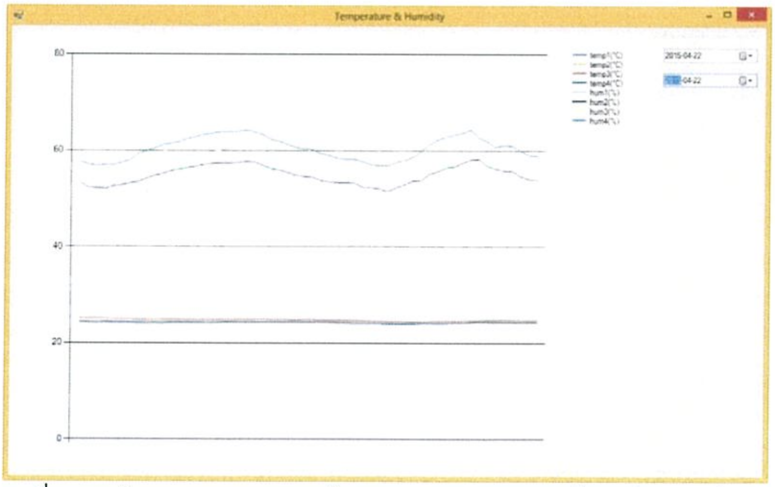
สัญญาณที่วัดได้จากขาของเซนเซอร์ที่ต่อกับขาดิจิตอลของไมโครคอลโทรลเลอร์ วัดสัญญาณขณะที่เซนเซอร์ส่งค่าไปยังไมโครคอลโทรลเลอร์ โดยวัดสัญญาณได้ดังรูปที่ 4.13 โดยจากรูป สัญญาณที่ได้มีขนาดประมาณ 4.8 โวลต์ มีคาบโดยประมาณ 20 มิลลิวินาที เป็นสัญญาณดิจิตอล



รูปที่ 4.13 สัญญาณที่วัดจากขาของเซนเซอร์

4.6 การแสดงผลค่าอุณหภูมิและความชื้นสัมพัทธ์จากฐานข้อมูล

เมื่อเปิดโปรแกรมที่ใช้สำหรับแสดงผลข้อมูล เมื่อเลือกช่วงวันที่ที่ต้องการดูข้อมูล โปรแกรมจะแสดงกราฟดังรูปที่ 4.14 จากรูปโดยกราฟเส้นแต่ละสีจะแสดงค่าความชื้นสัมพัทธ์และแสดงค่าอุณหภูมิที่รับค่ามาจากเซนเซอร์ในแกน y โดยเทียบกับเวลาในแกน x



รูปที่ 4.14 โปรแกรมแสดงผลข้อมูล แสดงกราฟจากข้อมูลในฐานข้อมูล

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

จากผลการทดลองการทำงานของแอพพลิเคชั่นพบว่า แอพพลิเคชั่นสามารถเชื่อมต่อกับฐานข้อมูลที่เซิร์ฟเวอร์ได้ โดยผ่านเครือข่าย แอพพลิเคชั่นสามารถรับ-ส่งข้อมูลกับฐานข้อมูลที่เซิร์ฟเวอร์ได้ ส่วนของไมโครคอนโทรลเลอร์สามารถรับส่งข้อมูลกับฐานข้อมูลได้โดยการส่งค่าไปเก็บไว้ในฐานข้อมูลได้และรับค่าจากฐานข้อมูลได้สามารถควบคุมการเปิดปิดของพัดลม ไมโครคอนโทรลเลอร์สามารควบคุมวงจรรีเลย์ได้ตามคำสั่งและทำงานได้เมื่อมีค่าอุณหภูมิที่เกินค่าที่ได้กำหนดค่าไว้ในการทดลอง

5.2 ข้อเสนอแนะ

ในการควบคุมอุณหภูมิและความชื้นภายสัมพัทธ์ในห้องปิดหรือโรงเรือน ประสิทธิภาพในการควบคุมขึ้นอยู่กับสภาพแวดล้อมอื่นๆและปัจจัยหลายๆอย่าง ประสิทธิภาพในการควบคุมอุณหภูมิและความชื้นสัมพัทธ์จึงไม่ได้มีประสิทธิภาพดีและแม่นยำเท่าที่ควรเนื่องจากปัจจัยเหล่านี้ส่งผลกระทบต่อการควบคุม

บรรณานุกรม

[1] พร้อมเลิศ หล่อวิจิตร. คู่มือเขียนแอพ Android ฉบับสมบูรณ์. พิมพ์ครั้งที่1. กรุงเทพฯ. โปรวิชั่น. 2556.

[2] จักรชัย โสอินทร์. Basic Android App Development. พิมพ์ครั้งที่1. นนทบุรี. ไอดีซีฯ. 2554.

[3] อรพิน ประวัติบริสุทธิ์. คู่มือเขียนโปรแกรมด้วยภาษา Java ฉบับสมบูรณ์. พิมพ์ครั้งที่1. กรุงเทพฯ. โปรวิชั่น. 2556.

[4] กอบเกียรติ สระอุบล. พัฒนา App Android. พิมพ์ครั้งที่1. กรุงเทพฯ. มีเดีย เนทเวิร์ค. 2556

[5] Department of Computer Science University of Maryland. "Interpolation." www.cs.umd.edu/~djacobs/CMSC427/Interpolation.pdf.

[6] Michigan Technological University. "Linear Interpolation." www.me.mtu.edu/~jstallen/courses/.../thermo.../Linear_Interpolation.pdf.

[7] มหาลัยวิทยาลัยมหิดล. "อุณหภูมิ" http://www.il.mahidol.ac.th/emedia/apchemistry2/gases/web/link/temperature.htm

[8] บริษัทธนพลวานิช จำกัด."ความชื้น" http://www.tanapolvanich.com/drybag/html/Relative_Humidity_th.htm

ภาคผนวก ก

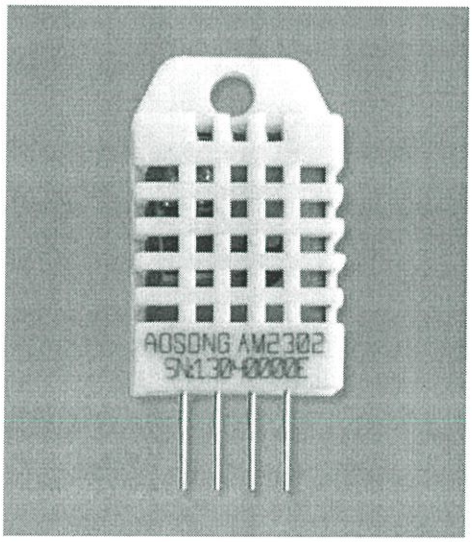
ข้อมูลเซนเซอร์

参考資料

AOSONG

Temperature and humidity module

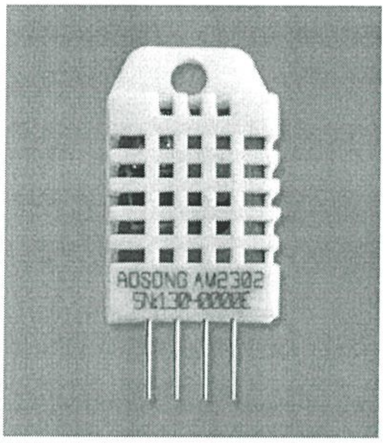
AM2302 Product Manual



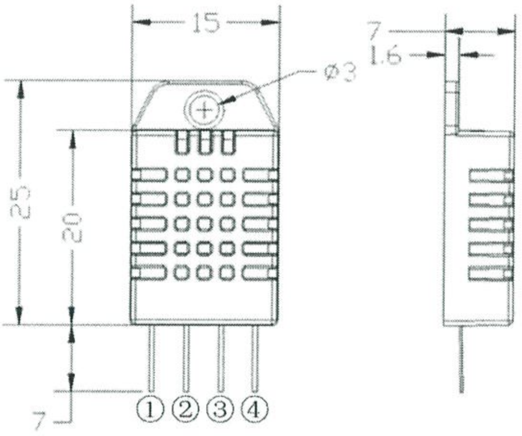
www.aosong.com

1、Product Overview

AM2302 capacitive humidity sensing digital temperature and humidity module is one that contains the compound has been calibrated digital signal output of the temperature and humidity sensors. Application of a dedicated digital modules collection technology and the temperature and humidity sensing technology, to ensure that the product has high reliability and excellent long−term stability. The sensor includes a capacitive sensor wet components and a high−precision temperature measurement devices, and connected with a high−performance 8−bit microcontroller. The product has excellent quality, fast response, strong anti−jamming capability, and high cost. Each sensor is extremely accurate humidity calibration chamber calibration. The form of procedures, the calibration coefficients stored in the microcontroller, the sensor within the processing of the heartbeat to call these calibration coefficients. Standard single−bus interface, system integration quick and easy. Small size, low power consumption, signal transmission distance up to 20 meters, making it the best choice of all kinds of applications and even the most demanding applications. Products for the 3−lead (single−bus interface) connection convenience. Special packages according to user needs.



Physical map



Dimensions (unit: mm)

2、Applications

HVAC, dehumidifier, testing and inspection equipment, consumer goods, automotive, automatic control, data loggers, home appliances, humidity regulator, medical, weather stations, and other humidity measurement and control and so on.

3、Features

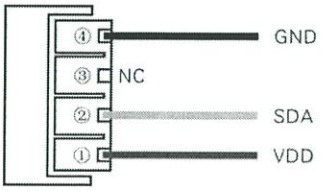
Ultra−low power, the transmission distance, fully automated calibration, the use of capacitive humidity sensor, completely interchangeable, standard digital single−bus output, excellent long−term stability, high accuracy temperature measurement devices.

4、The definition of single-bus interface

4.1 AM2302 Pin assignments

Table 1: AM2302 Pin assignments

Pin	Name	Description
①	VDD	Power (3.3V−5.5V)
②	SDA	Serial data, bidirectional port
③	NC	Empty
④	GND	Ground



PIC1: AM2302 Pin Assignment

4.2 Power supply pins (VDD GND)

AM2302 supply voltage range 3.3V − 5.5V, recommended supply voltage is 5V.

4.3 Serial data (SDA)

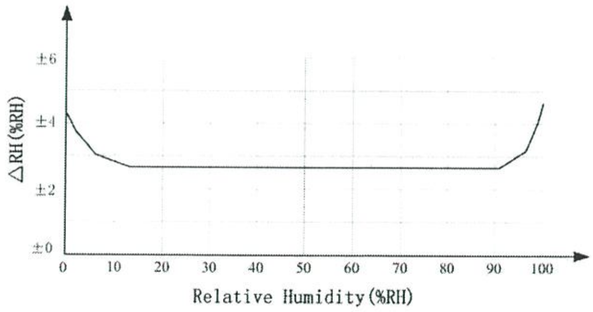
SDA pin is tri structure for reading, writing sensor data. Specific communication timing, see the detailed description of the communication protocol.

5、Sensor performance

5.1 Relative humidity

Table 2: AM2302 Relative humidity performance table

Parameter	Condition	min	typ	max	Unit
Resolution			0.1		%RH
Range		0		99.9	%RH
Accuracy [1]	25℃		±2		%RH
Repeatability			±0.3		%RH
Exchange		Completely interchangeable			
Response [2]	1/e(63%)		<5		S
Sluggish			<0.3		%RH
Drift [3]	Typical		<0.5		%RH/yr

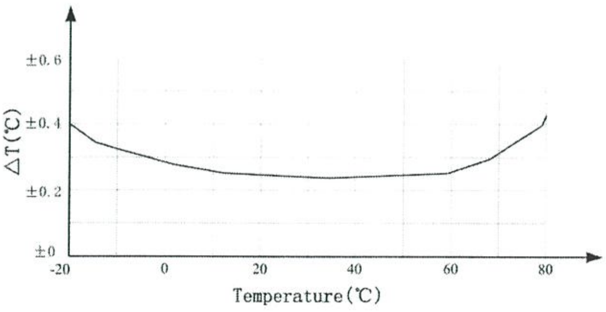


Pic2: At25℃ The error of relative humidity

5.2 Temperature

Table 3: AM2302 Relative temperature performance

Parameter	Condition	min	typ	max	Unit
Resolutio			0.1		℃
n			16		bit
Accuracy			±0.5	±1	℃
Range		−40		80	℃
Repeat			±0.2		℃
Exchange		Completely interchangeable			
Response	1/e(63%)		<10		S
Drift			±0.3		℃/yr



Pic3: The maximum temperature error

6、Electrical Characteristics

Electrical characteristics, such as energy consumption, high, low, input, output voltage, depending on the power supply. Table 4 details the electrical characteristics of the AM2302, if not identified, said supply voltage of 5V. To get the best results with the sensor, please design strictly in accordance with the conditions of design in Table 4.

Table 4: AM2302 DC Characteristics

Parameter	Condition	min	typ	max	Unit
Voltage		3.3	5	5.5	V
Power consumption [4]	Dormancy	10	15		μA
	Measuring		500		μA
	Average		300		μA
Low level output voltage	I_{OL} [5]	0		300	mV
High output voltage	Rp<25 kΩ	90%		100%	VDD
Low input voltage	Decline	0		30%	VDD
Input High Voltage	Rise	70%		100%	VDD
Rpu[6]	VDD = 5V VIN = VSS	30	45	60	kΩ
Output current	turn on		8		mA
	turn off	10	20		μA
Sampling period		2			S

[1] the accuracy of the factory inspection, the sensor 25°C and 5V, the accuracy specification of test conditions, it does not include hysteresis and nonlinearity, and is only suitable for non−condensing environment.

[2] to achieve an order of 63% of the time required under the conditions of 25℃ and 1m / s airflow.

[3] in the volatile organic compounds, the values may be higher. See the manual application to store information.

[4] this value at VDD = 5.0V when the temperature is 25℃, 2S / time, under the conditions of the average.

[5] low output current.

[6] that the pull−up resistor.

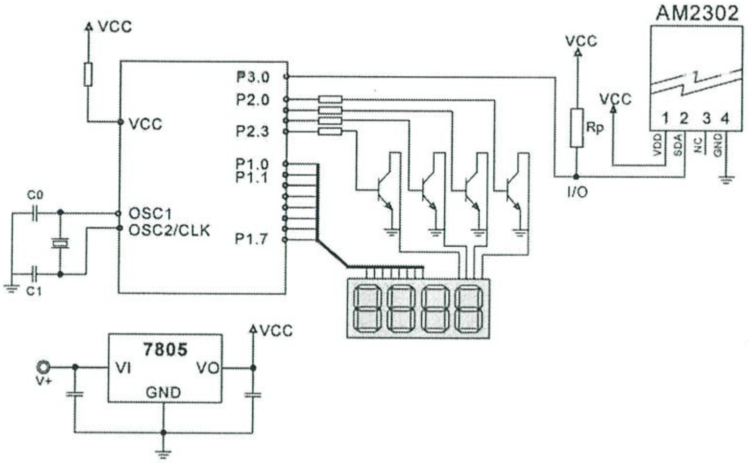
7、Single−bus communication (ONE−WIRE)

7.1 Typical circuits for single bus

Microprocessor and AM2302 connection typical application circuit is shown in Figure 4. Single bus communication mode, pull the SDA microprocessor I / O port is connected.

Special instructions of the single−bus communication:

1. Typical application circuit recommended in the short cable length of 30 meters on the 5.1K pull−up resistor pullup resistor according to the actual situation of lower than 30 m.
2. With 3.3V supply voltage, cable length shall not be greater than 100cm. Otherwise, the line voltage drop will lead to the sensor power supply, resulting in measurement error.
3. Read the sensor minimum time interval for the 2S; read interval is less than 2S, may cause the temperature and humidity are not allowed or communication is unsuccessful, etc..
4. Temperature and humidity values are each read out the results of the last measurement For real−time data that need continuous read twice, we recommend repeatedly to read sensors, and each read sensor interval is greater than 2 seconds to obtain accuratethe data.



Pic4: AM2302 Typical circuits for single bus

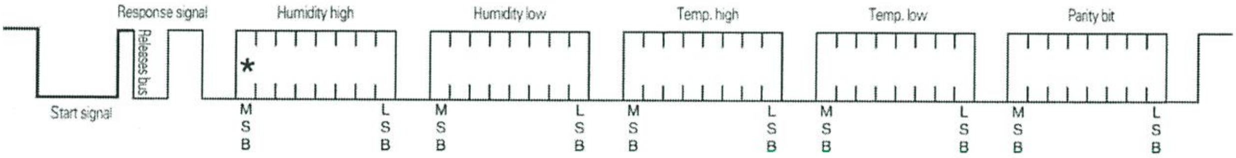
7.2、Single-bus communication protocol

◎Single bus Description

AM2302 device uses a simplified single-bus communication. Single bus that only one data line, data exchange system, controlled by the data line to complete. Equipment (microprocessor) through an open-drain or tri-state port connected to the data line to allow the device does not send data to release the bus, while other devices use the bus; single bus usually require an external about 5.1kΩ pull-up resistor, so when the bus is idle, its status is high. Because they are the master-slave structure, only the host calls the sensor, the sensor will answer, so the hosts to access the sensor must strictly follow the sequence of single bus, if there is a sequence of confusion, the sensor will not respond to the host.

◎Single bus to send data definition

SDA For communication and synchronization between the microprocessor and the AM2302, single-bus data format, a transmission of 40 data, the high first-out. Specific communication timing shown in Figure 5, the communication format is depicted in Table 5.



Pic5: AM2302 Single-bus communication protocol

Table 5：AM2302 Communication format specifier

Name	Single-bus format definition
Start signal	Microprocessor data bus (SDA) to bring down a period of time (at least 800μ s) [1] notify the sensor to prepare the data.
Response signal	Sensor data bus (SDA) is pulled down to 80μ s, followed by high-80μ s response to host the start signal.
Data format	Host the start signal is received, the sensor one-time string from the data bus (SDA) 40 data, the high first-out.
Humidity	Humidity resolution of 16Bit, the previous high; humidity sensor string value is 10 times the actual humidity values.
Temp.	Temperature resolution of 16Bit, the previous high; temperature sensor string value is 10 times the actual temperature value; The temperature is the highest bit (Bit15) is equal to 1 indicates a negative temperature, the temperature is the highest bit (Bit15) is equal to 0 indicates a positive temperature; Temperature in addition to the most significant bit (Bit14 ~ bit 0) temperature values.
Parity bit	Parity bit = humidity high + humidity low + temperature high + temperature low

◎**Single-bus data calculation example**

Example 1：40 Data received：

0000 0010	1001 0010	0000 0001	0000 1101	1010 0010
High humidity 8	Low humidity 8	High temp. 8	Low temp. 8	Parity bit

Calculate：

0000 0010+1001 0010 +0000 0001+0000 1101= 1010 0010（Parity bit）

Received data is correct：

humidity：0000 0010　1001 0010 = 0292H (Hexadecimal)= 2×256 + 9×16 + 2 = 658

=> Humidity = 65.8%RH

Temp.：0000 0001　0000 1101 = 10DH(Hexadecimal) = 1×256 + 0×16 + 13 = 269

=> Temp.= 26.9℃

◎**Special Instructions**：

When the temperature is below 0 ℃, the highest position of the temperature data.

Example：　−10.1 ℃ Expressed as 1 000 0000 0110 0101

Temp.：0000 0000 0110 0101 = 0065H(Hexadecimal)=　6×16 +5 = 101

=> Temp. ＝ −10.1℃

Example 2：40 received data：

0000 0010	1001 0010	0000 0001	0000 1101	1011 0010
High humidity 8	Low humidity 8	High temp. 8	Low temp. 8	Parity bit

Calculate：

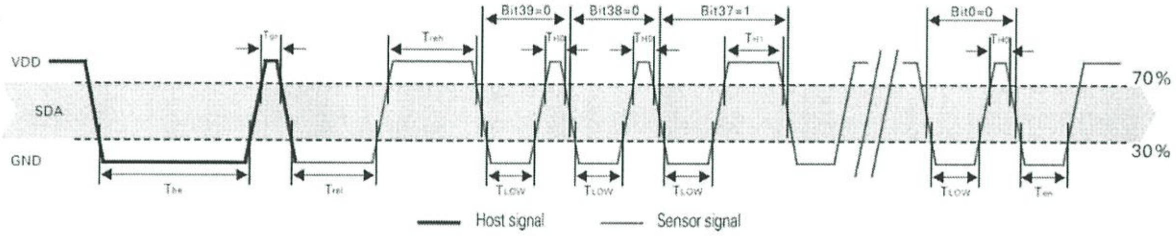
0000 0010+1001 0010 +0000 0001+0000 1101= 1010 0010 ≠ 1011 0010 （Validation error）

The received data is not correct, give up, to re-receive data.

7.3 Single—bus communication timing

User host (MCU) to send a start signal (data bus SDA line low for at least 800μ s) after AM2302 from Sleep mode conversion to high—speed mode.The host began to signal the end of the AM2302 send a response signal sent from the data bus SDA serial 40Bit's data, sends the byte high; data sent is followed by: Humidity high、Humidity low、Temperature high、Temperature low、Parity bit, Send data to the end of trigger information collection, the collection end of the sensor is automatically transferred to the sleep mode, the advent until the next communication.

Detailed timing signal characteristics in Table 6, Single—bus communication timing diagram Pic 6:



Pic 6: AM2302 Single—bus communication timing

Note: the temperature and humidity data read by the host from the AM2302 is always the last measured value, such as the two measurement interval is very long, continuous read twice to the second value of real—time temperature and humidity values, while two readtake minimum time interval be 2S.

Table 6: Single bus signal characteristics

Symbol	Parameter	min	typ	max	Unit
T_{be}	Host the start signal down time	0.8	1	20	mS
T_{go}	Bus master has released time	20	30	200	μS
T_{rel}	Response to low time	75	80	85	μS
T_{reh}	In response to high time	75	80	85	μS
T_{LOW}	Signal "0", "1" low time	48	50	55	μS
T_{H0}	Signal "0" high time	22	26	30	μS
T_{H1}	Signal "1" high time	68	70	75	μS
T_{en}	Sensor to release the bus time	45	50	55	μS

Note: To ensure the accurate communication of the sensor, the read signal, in strict accordance with the design parameters and timing in Table 6 and Figure 6.

7.4 Peripherals read step example

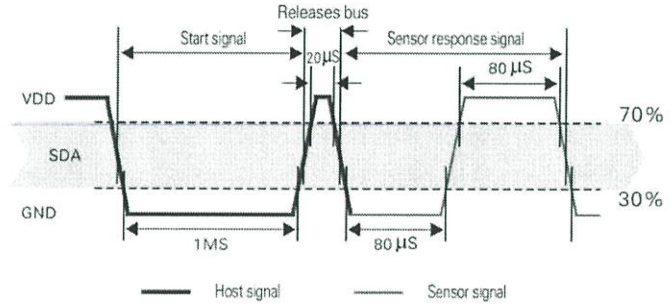
Communication between the host and the sensor can read data through the following three steps to complete.

Step 1

AM2302 have to wait for the power (on AM2302 power 2S crossed the unstable state, the device can not send any instructions to read during this period), the test environment temperature and humidity data, and record data, since the sensor into a sleep state automatically. AM2302 The SDA data line from the previous pull—up resistor pulled up is always high, the AM2302 the SDA pin is in input state, the time detection of external signal.

Step 2

Microprocessor I/O set to output, while output low, and low hold time can not be less than 800us, typical values are down 1MS, then the microprocessor I/O is set to input state, the release of the bus, due to the pull−up resistor, the microprocessor I/O AM2302 the SDA data line also will be high, the bus master has released the AM2302 send a response signal, that is, the output 80 microseconds low as the response signal, tightthen output high of 80 microseconds notice peripheral is ready to receive data signal transmission as shown to Pic7 :



Pic7: Single bus decomposition of the timing diagram

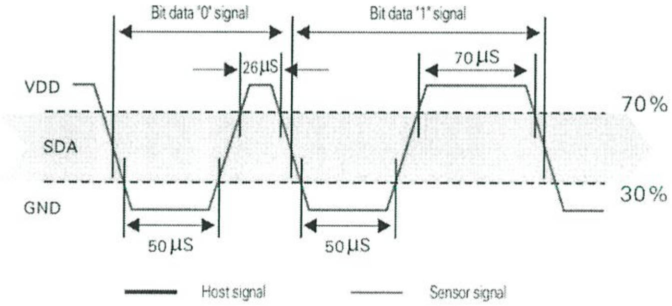
Step 3

AM2302 sending the response, followed by the data bus SDA continuous serial output 40 data, the microprocessor receives 40 data I/O level changes.

Bit data "0" format: 26−28 microseconds 50 microseconds low plus high;

Bit data "1" format: the high level of low plus, 50 microseconds to 70 microseconds;

Bit data "0" bit data "1" format signal shown to pic 8:

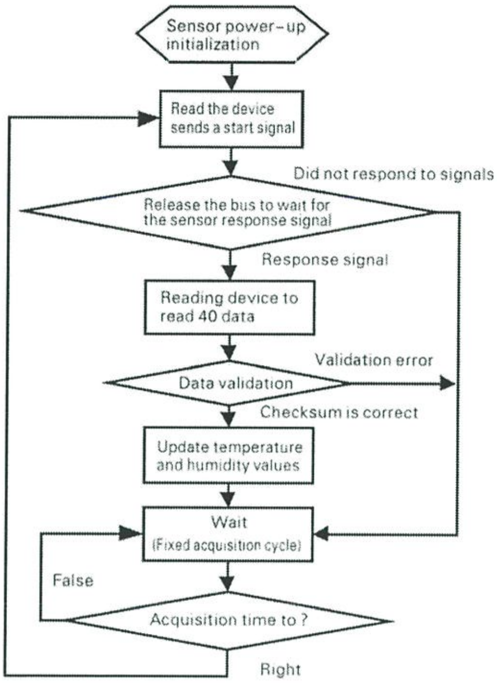


Pic 8: The single bus break down the timing diagram

AM2302 data bus SDA output 40 data continue to output the low 50 microseconds into the input state, followed by pull−up resistor goes high. AM2302 internal re−test environmental temperature and humidity data, and record the data, the end of the test records, the microcontroller automatically into hibernation. Microcontroller only after receipt of the start signal of the host wake−up sensor, into the working state.

7.5 Peripheral to read flow chart

AM2302 sensor read single bus flow chart diagram shown in Figure 9, we also provide the C51 read the code examples, customers need to download, please visit our website (www.aosong.com) related to downloadthis manual does not provide the code description.



Pic9: Single-bus to read the flow chart

8、Application of information

1. Work and storage conditions

Outside the sensor the proposed scope of work may lead to temporary drift of the signal up to 300% RH. Return to normal working conditions, sensor calibration status will slowly toward recovery. To speed up the recovery process may refer to "resume processing". Prolonged use of non-normal operating conditions, will accelerate the aging of the product.

Avoid placing the components on the long-term condensation and dry environment, as well as the following environment.

A, salt spray

B, acidic or oxidizing gases such as sulfur dioxide, hydrochloric acid

Recommended storage environment

Temperature: 10 ~ 40 ℃ Humidity: 60% RH or less

2. The impact of exposure to chemicals

The capacitive humidity sensor has a layer by chemical vapor interference, the proliferation of chemicals in the sensing layer may lead to drift and decreased sensitivity of the measured values. In a pure environment, contaminants will slowly be released. Resume processing as described below will accelerate this process. The high concentration of chemical pollution (such as ethanol) will lead to the complete damage of the sensitive layer of the sensor.

3. The temperature influence

Relative humidity of the gas to a large extent dependent on temperature. Therefore, in the measurement of humidity,

should be to ensure that the work of the humidity sensor at the same temperature. With the release of heat of electronic components share a printed circuit board, the installation should be as far as possible the sensor away from the electronic components and mounted below the heat source, while maintaining good ventilation of the enclosure. To reduce the thermal conductivity sensor and printed circuit board copper plating should be the smallest possible, and leaving a gap between the two.

4. Light impact

Prolonged exposure to sunlight or strong ultraviolet radiation, and degrade performance.

5. Resume processing

Placed under extreme working conditions or chemical vapor sensor, which allows it to return to the status of calibration by the following handler. Maintain two hours in the humidity conditions of 45℃ and <10% RH (dry); followed by 20−30℃ and> 70% RH humidity conditions to maintain more than five hours.

6. Wiring precautions

The quality of the signal wire will affect the quality of the voltage output, it is recommended to use high quality shielded cable.

7. Welding information

Manual welding, in the maximum temperature of 300℃ under the conditions of contact time shall be less than 3 seconds.

8. Product upgrades

Details, please the consultation Aosong electronics department.

9、The license agreement

Without the prior written permission of the copyright holder, shall not in any form or by any means, electronic or mechanical (including photocopying), copy any part of this manual, nor shall its contents be communicated to a third party. The contents are subject to change without notice.

The Company and third parties have ownership of the software, the user may use only signed a contract or software license.

10、Warnings and personal injury

This product is not applied to the safety or emergency stop devices, as well as the failure of the product may result in injury to any other application, unless a particular purpose or use authorized. Installation, handling, use or maintenance of the product refer to product data sheets and application notes. Failure to comply with this recommendation may result in death and serious personal injury. The Company will bear all damages resulting personal injury or death, and waive any claims that the resulting subsidiary company managers and employees and agents, distributors, etc. that may arise, including: a variety of costs, compensation costs, attorneys' fees, and so on.

11、Quality Assurance

The company and its direct purchaser of the product quality guarantee period of three months (from the date of delivery). Publishes the technical specifications of the product data sheet shall prevail. Within the warranty period, the product was confirmed that the quality is really defective, the company will provide free repair or replacement. The user must satisfy the following conditions:

① The product is found defective within 14 days written notice to the Company;

② The product shall be paid by mail back to the company;

③ The product should be within the warranty period.

The Company is only responsible for those used in the occasion of the technical condition of the product defective product. Without any guarantee, warranty or written statement of its products used in special applications. Company for its products applied to the reliability of the product or circuit does not make any commitment.

ภาคผนวก ข

ข้อมูลไมโครคอนโทรลเลอร์

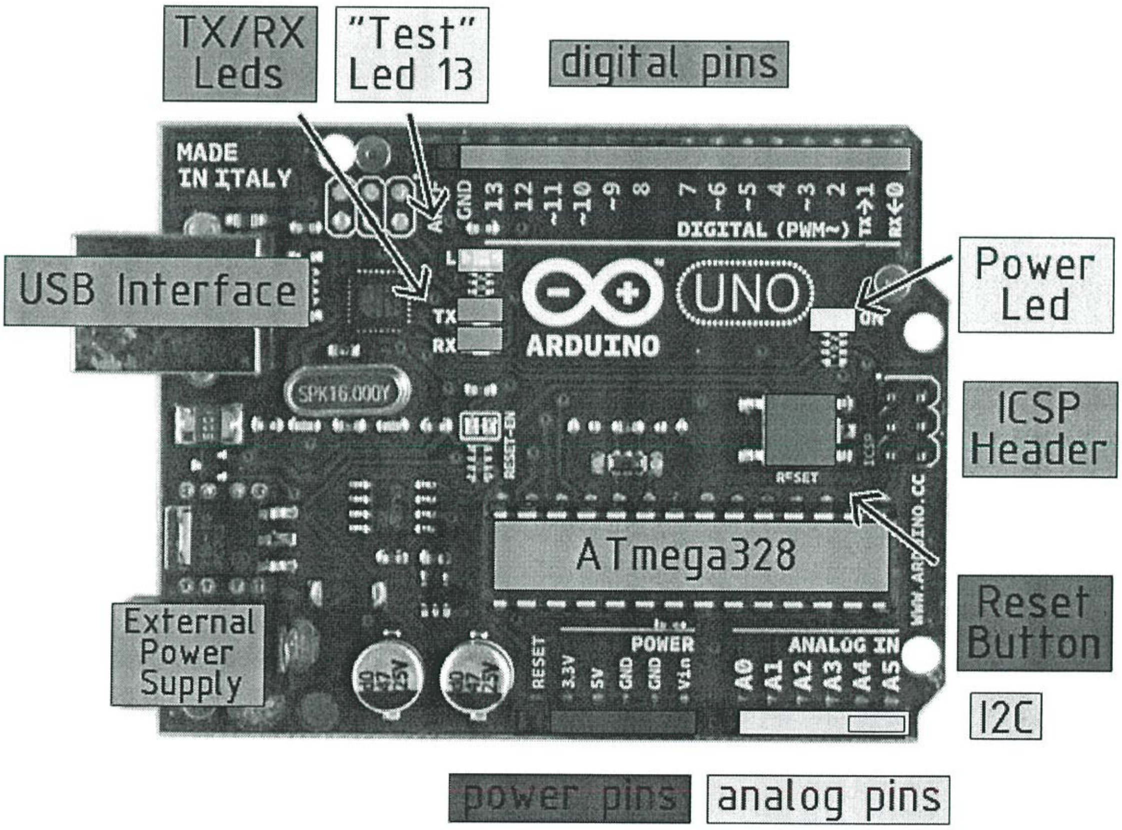
Technical Specification

EAGLE files: arduino-duemilanove-uno-design.zip Schematic: arduino-uno-schematic.pdf

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

the board



RS
radiospares
RADIONICS
ALLIED ELECTRONICS

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

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

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

The power pins are as follows:

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

Memory

The Atmega328 has 32 KB of flash memory for storing code (of which 0,5 KB is used for the bootloader); It has also 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. TThese pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the analogWrite() function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.

- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. Additionally, some pins have specialized functionality:

- **I^2C: 4 (SDA) and 5 (SCL).** Support I^2C (TWI) communication using the Wire library.

There are a couple of other pins on the board:

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

See also the mapping between Arduino pins and Atmega328 ports.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega8U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '8U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an *.inf file is required..

The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

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

The ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. To use the SPI communication, please see the ATmega328 datasheet.

Programming

The Arduino Uno can be programmed with the Arduino software (download). Select "Arduino Uno w/ ATmega328" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the reference and tutorials.

The ATmega328 on the Arduino Uno comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

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

The ATmega8U2 firmware source code is available . The ATmega8U2 is loaded with a DFU bootloader, which can be activated by connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader).

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

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

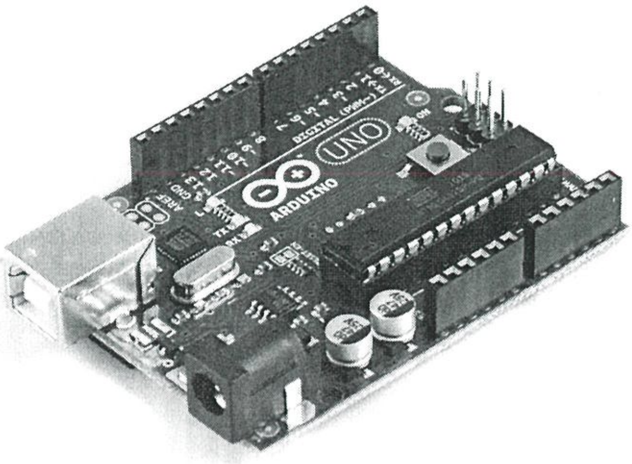
The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread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

USB Overcurrent Protection

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

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.



How to use Arduino

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

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

Linux Install **Windows Install** **Mac Install**

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

Blink led

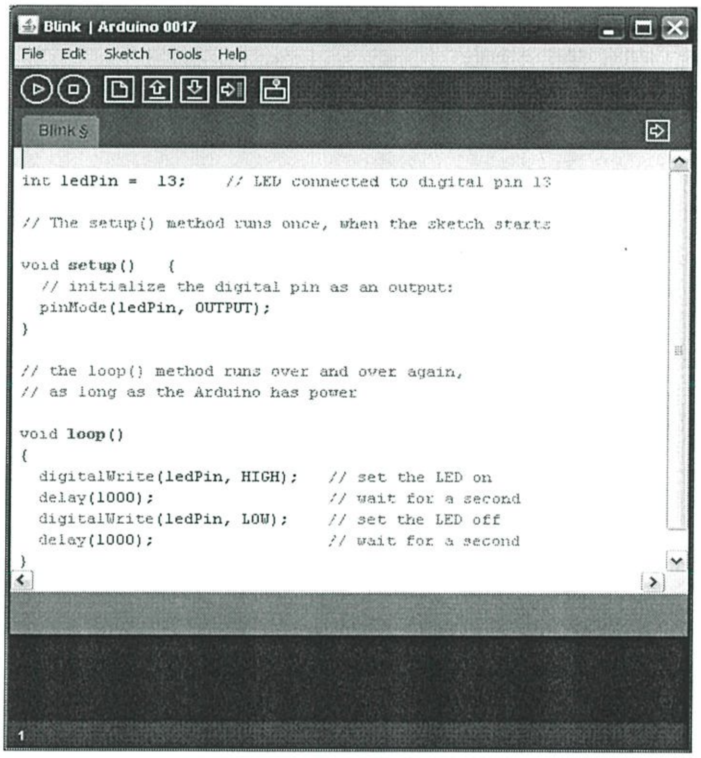
Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

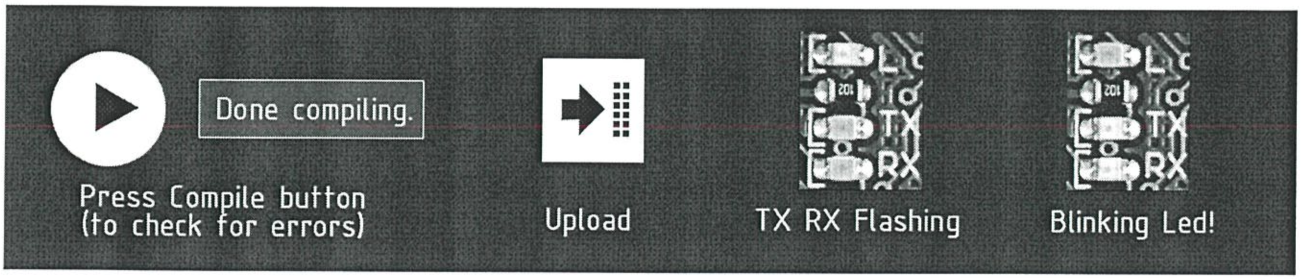
**File>Sketchbook>
Arduino-0017>Examples>
Digital>Blink**

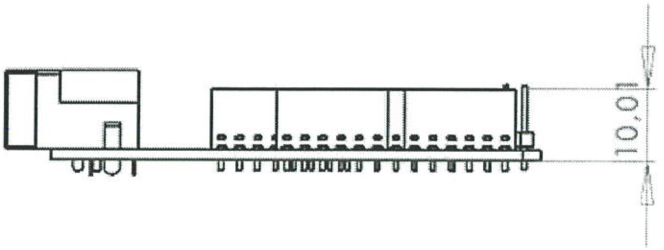
Once you have your skecth you'll see something very close to the screenshot on the right.

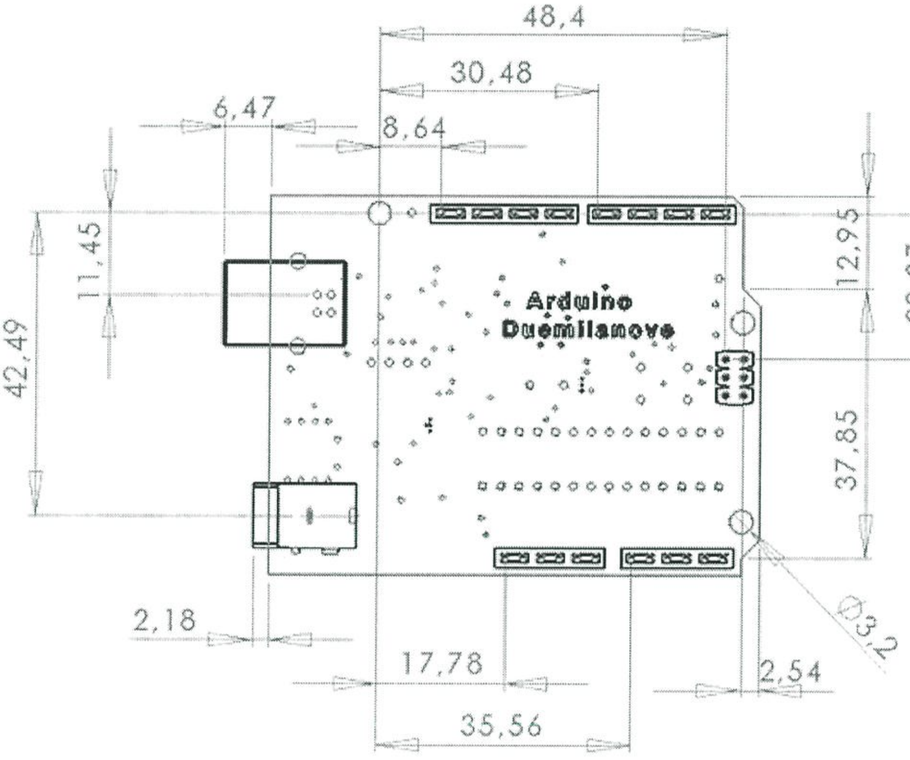
In **Tools>Board** select

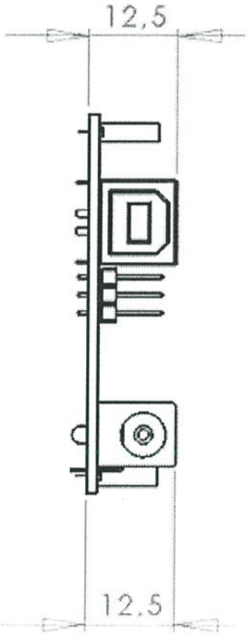
Now you have to go to
Tools>SerialPort
and select the right serial port, the one arduino is attached to.

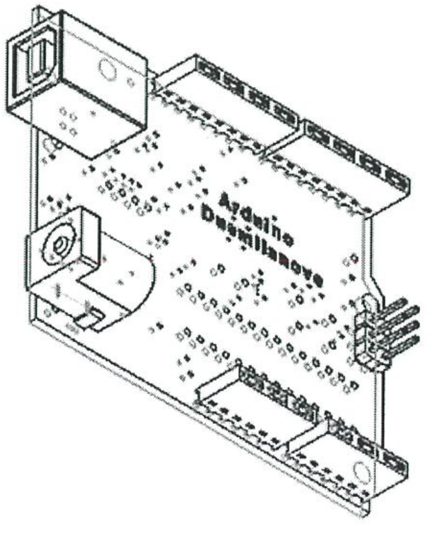












RS radiospares RADIONICS ALLIED ELECTRONICS

Terms & Conditions

1. Warranties

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

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

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

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

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

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

2. Indemnification

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

3. Consequential Damages Waiver

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

4. Changes to specifications

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

Enviromental Policies

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

Features

- Drop in Wi-Fi solution for existing systems currently using 802.15.4 modules
- Based on Roving Networks' robust RN-171 Wi-Fi module
- Based on pseudo-standard footprint
- Onboard TCP/IP stack provides internet access to every node
- No custom profiles needed
- Ultra-low power: 4uA sleep, 40mA Rx, 180 mA Tx at 10dBm
- Configurable transmit power: 0dBm to +12dBm
- Hardware interface: TTL UART
- Data rate: 464Kbps using hardware flow control
- Through hole board simplifies system integration
- 8 general purpose digital I/O
- 3 analog sensor interfaces.
- Real-time clock for wakeup and time stamping
- Complete TCP/IP networking stack
- Wi-Fi Alliance certified for WEP, WPA and WPA2-PSK
- WPS push button for easy configuration
- FCC / CE/ ICS certified and RoHS compliant.

Applications

- Industrial metering
- HVAC control
- Room temperature sensors
- Pump configuration and control
- Telemetry
- PV/Solar controllers
- Robotics

Description

The RN-XV is a 802.11 b/g solution especially designed for customer who want to migrate their existing 802.15.4 architecture to a more standard TCP/IP based platform without having to redesign their existing hardware.

Based on a pseudo standard footprint often found in embedded applications, the RN-XV module from Roving Networks allows for Wi-Fi connectivity using 802.11 b/g standards in legacy and existing designs that may have been based upon 802.15.4 standard.

The RN-XV module is based upon Roving Networks' robust RN-171 Wi-Fi module and incorporates 802.11 b/g radio, 32 bit SPARC processor, TCP/TP stack, real-time clock, crypto accelerator, power management unit and analog sensor interface.

The module offers additional functionality through its onboard programmable GPIOs (10) and ADCs (8). The ADCs provide 14-bit resolution while the GPIOs can be configured to provide standard functionality or status signaling to a host microcontroller to reduce the need for serial polling between the Wi-Fi module and host microcontroller.

The module is pre-loaded with firmware to simplify integration and minimize development time of your application. In the simplest configuration, the hardware only requires four connections (PWR, TX, RX and GND) to create a wireless data connection.

Overview

- Host Data Rate up to 460Kbps over UART
- Intelligent built-in power management with programmable wakeup events (timers and I/O)
- Real time clock for time stamping, auto-sleep and auto-wakeup modes
- Configuration over WiFi or UART using simple ASCII commands
- Over the air firmware upgrade via FTP
- Secure WiFi authentication: WEP, WPA-PSK (TKIP), WPA2-PSK
- Built in networking applications DHCP, DNS, ARP, ICMP UDP, Telnet, FTP, HTML client
- 802.11 b/g power save and roaming functions
- Configurable transmit power: -2dBm to 12dBm
- WPS push button mode for easy and secure wireless setup
- Built-in HTML web client to send GPIO, UART and sensor data to remote web servers

Environmental Conditions

Parameter	Value
Temperature Range (Operating)	-40 °C - 85 °C
Temperature Range (Storage)	-40°C - 85 °C
Relative Humidity (Operating)	≤90%
Relative Humidity (Storage)	≤90%

Electrical Characteristics

Supply Voltage	Min	Typ.	Max.	Unit
Input Power	3.0	3.3	3.7	V
Power consumption				
Sleep		4		uA
Standby (doze)		15		mA
Connected (idle, RX)		40		mA
Connected (TX)		180 at 10dBm		mA

Analog Sensor Inputs

Parameter	Value
Sense 0,1,2,3 wakeup detect threshold	500mV
AD sense 0-4 measurement range	0-400mV
Precision	14 bits = 12uV
Accuracy	5% un-calibrated, .01% calibrated
Minimum conversion time	35uS (5kHz over Wi-Fi)
Sensor Power (pin 33) output resistance 3.3V	10 ohms, max current = 50mA

Radio Characteristics

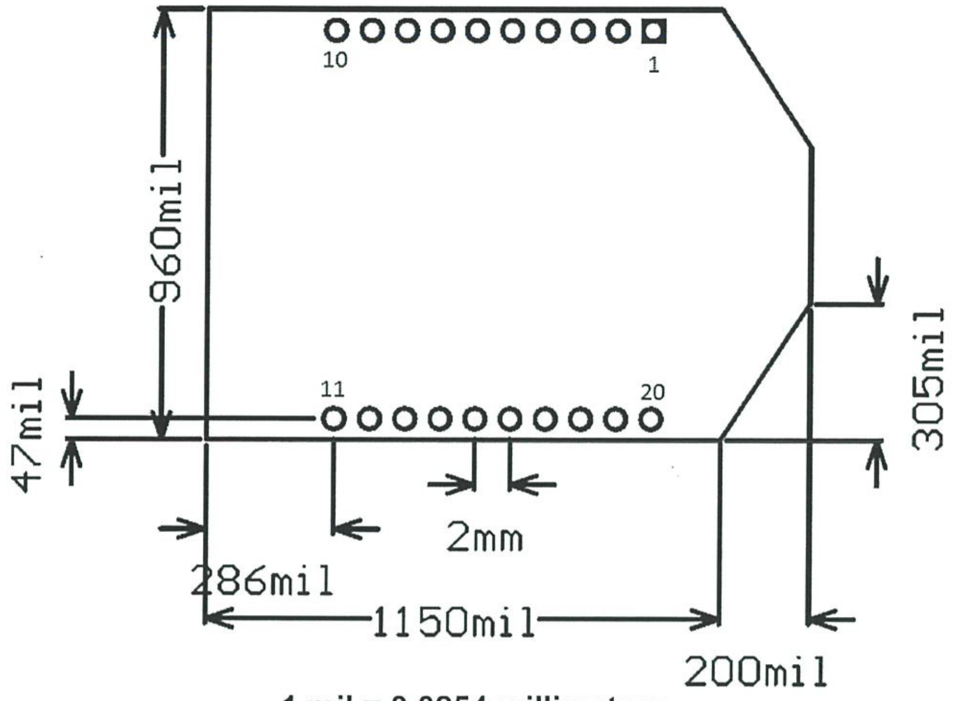
Parameter	Specifications
Frequency	2402 ~ 2480MHz
Modulation	802.11b compatibility : DSSS(CCK-11, CCK-5.5, DQPSK-2, DBPSK-1) 802.11g : OFDM (default)
Channel intervals	5MHz
Channels	1 - 14
Transmission rate (over the air)	1 – 11Mbps for 802.11b / 6 – 54Mbps for 802.11g
Receive sensitivity	-83dBm typ.
Output level (Class1)	0dBm to +12dBm (software configurable)

Transmit Power

Output Power	802.11 b (2Mbps) Current in mA*	802.11 g (24Mbps) Current in mA*
0	120	135
2	130	150
4	170	190
6	175	200
8	180	210
10	185	225
12	190	240

*** Measured at 3.3VDC input. The power consumption is the average power, active during actual power consumption**

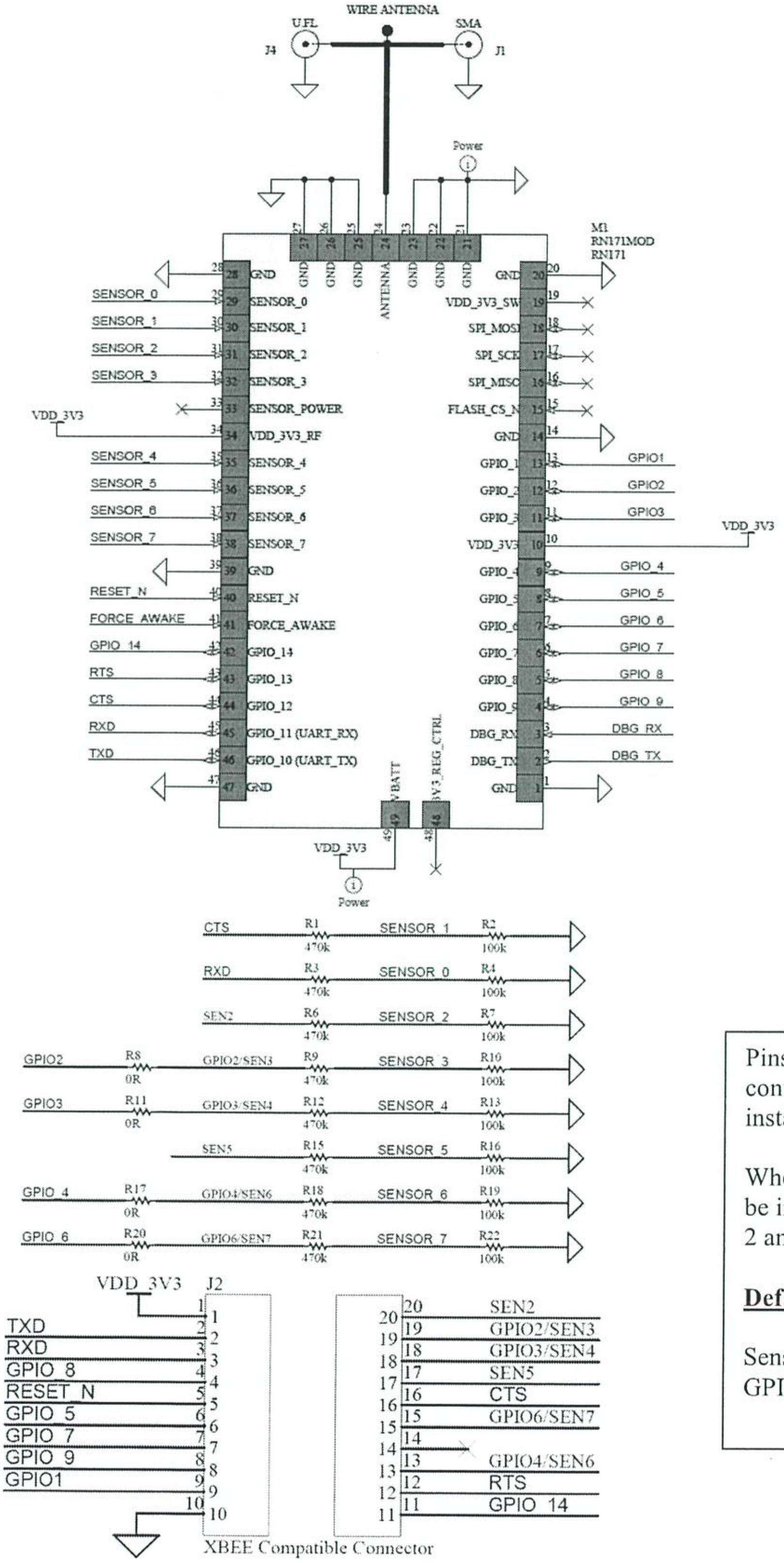
Physical Dimensions and pin out table



1 mil = 0.0254 millimeters

Pad Number	Signal Name	Description	Optional Function	Direction
1	VDD_3V3	3.3V regulated power input to the module		POWER
2	UART_TX	UART TX, 8mA drive, 3.3V tolerant		OUT →
3	UART_RX	UART RX, 3.3V tolerant		IN ←
4	GPIO 8	GPIO, 24mA drive, 3.3V tolerant		IN / OUT
5	RESET	**Optional** Module Reset Signal (active low), 100k Pull up, apply pulse of at least 160us, 3.3V Tolerant		INPUT
6	GPIO 5	GPIO, 24mA drive, 3.3V tolerant	Data TX/RX	OUT
7	GPIO 7	GPIO, 24mA drive, 3.3V tolerant		IN / OUT
8	GPIO 9	Enable Adhoc mode, Restore factory defaults, 8mA drive, 3.3V tolerant		IN / OUT
9	GPIO 1	GPIO, 8mA drive, 3.3V tolerant		IN / OUT
10	GND	Ground		GND
11	GPIO 14	GPIO, 8mA drive, 3.3V tolerant		IN / OUT
12	UART_RTS	UART RTS flow control, 8mA drive, 3.3V tolerant		OUT →
13	GPIO 4/SEN 6	GPIO, 24mA drive, 3.3V tolerant/ADC input , (**3.3V tolerant). ** Defaults to GPIO 4	Association Status	IN / OUT
14	Not Used			No Connect
15	GPIO 6/SEN 7	GPIO, 24mA drive, 3.3V tolerant/ADC input , (**3.3V tolerant). ** Defaults to GPIO 6	Connection Status	POWER
16	UART_CTS	UART CTS flow control, 3.3V tolerant		IN ←
17	SENSOR 5	Sensor Interface, Analog input to module, (**3.3V tolerant**)		INPUT
18	GPIO 3/SEN 4	GPIO, 8mA drive, 3.3V tolerant/ADC input (**3.3V tolerant**). Defaults to GPIO 3		IN / OUT
19	GPIO 2/SEN 3	GPIO, 8mA drive, 3.3V tolerant/ADC input (**3.3V tolerant**). Defaults to SEN 3		IN / OUT
20	SEN 2	Sensor Interface, Analog input to module, **3.3V tolerant**		INPUT

Module Schematic



Pins 13, 15, 18 and 19 on the RN-XV connector can be configured as GPIOs or as sensor inputs depending on the installed resistors.

When configured as GPIOs, ONLY Column 1 resistors MUST be installed. When configured as sensor inputs, ONLY Column 2 and Column 3 resistors MUST be stuffed.

Default configuration:

Sensor inputs – Pins 19 and 20
GPIO – Pins 13, 15 and 18

Design Concerns

1. **Powering the RN-XV module:** Apply ONLY 3.3V±10% regulated power to pin 1 (VDD) and pin 10 (Ground). The module does not have any voltage regulator on board and hence MUST be powered from a regulated 3.3V power supply.

2. **Antenna:** The RN-XV ships with a wire antenna. Custom antenna configurations such as chip, U.FL. and SMA connector are available with extended lead times

Ordering Information

Part Number	Description
RN-XV-W	Standard configuration, industrial Temperature (- 40 to + 85 C) 802.15.4 replacement solution with ¼ inch wire antenna
RN-XV-U	Custom configuration, industrial Temperature (- 40 to + 85 C) 802.15.4 replacement solution with U.FL. connector
RN-XV-S	Custom configuration, industrial Temperature (- 40 to + 85 C) 802.15.4 replacement solution with SMA connector
RN-XV-C	Custom configuration, industrial Temperature (- 40 to + 85 C) 802.15.4 replacement solution with chip antenna

Copyright © 2011 Roving Networks. All rights reserved.

Roving Networks reserves the right to make corrections, modifications, and other changes to its products, documentation and services at any time. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

Roving Networks assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using Roving Networks components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

Roving Networks products are not authorized for use in safety-critical applications (such as life support) where a failure of the Roving Networks product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use.

All other trademarks are property of their respective owners.