เซ็นเซอร์ตรวจวัดค่าสภาพแวดล้อมในโรงเรือนกล้วยไม้

โดย

นางสาวนารัตน์ แนบชัยภูมิ รหัสนักศึกษา B5600250

นางสาวกรรณิการ์ รุ่งอรุณ รหัสนักศึกษา B5606535

นางสาวสุชาดา กัณหา รหัสนักศึกษา B5612567

รายงานนี้เป็นส่วนหนึ่งของการศึกษารายวิชา 527499 โครงงานวิศวกรรมโทรคมนาคม

หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม

หลักสูตรวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

ประจำภาคการศึกษาที่1 ปีการศึกษา 2559

เซ็นเซอร์ตรวจวัดค่าสภาพแวดล้อมในโรงเรือนกล้วยไม้

คณะกรรมการสอบโครงงาน

(ผู้ช่วยศาสตราจารย์ เรืออากาศเอก ดร.ประโยชน์ คำสวัสดิ์)

อาจารย์ที่ปรึกษาโครงงาน

(รองศาสตราจารย์ ดร.มนต์ทิพย์ภา อุฑารสกุล)

กรรมการ

(อาจารย์ ดร.เศรษฐวิทย์ ภูฉายา)

กรรมการ

มหาวิทยาลัยเทคโนโลยีสุรนารี

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับรายงานโครงงานฉบับนี้ เป็นส่วนหนึ่งของการศึกษาระดับปริญญาตรี สาขาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์

บทคัดย่อ

การดำรงชีวิตและการเจริญเติบโตของกล้วยไม้จะต้องได้รับอิทธิพลจากหลายปัจจัยได้แก่ น้ำ อุณหภูมิ ความชื้น คาร์บอนไดออกไซด์ และความเข้มแสงแดด สภาพแวดล้อมที่ต้นไม้เจริญเติบโตก็เป็นตัว แปรที่สำคัญเช่นกัน ความชื้นและอุณหภูมิ เป็นตัวแปรสำคัญที่มีผลต่อการเจริญเติบโตของกล้วยไม้และถ้าเกิดมีความไม่เหมาะสมอาจส่งผลให้การเจริญเติบโตของกล้วยไม้มีปัญหา ในทีนี้จึงมีการใช้น้ำเข้ามาช่วยใน การรักษา อุณหภูมิและความชื้น ดังนั้น จึงคิดใช้เซ็นเซอร์มาช่วยควบคุมการเปิด–ปิดน้ำเพื่อใช้รักษาค่าสภาพแวดล้อมในโรงเรือนกล้วยไม้โดยสั่งการผ่านสมาร์ทโฟนที่สามารถทำงานบนระบบปฏิบัติการแอนดรอยด์ เพื่อควบคุมเวลาในการรดน้ำกล้วยไม้ในโรงเรือน โดยขั้นตอนแรกจะติดตั้งแอปพลิเคชั่นลงบนสมาร์ทโฟน ที่ทา งานบนระบบปฏิบัติการแอนดรอยด์หลังจากนั้น คำสั่งจะรับค่าจากเซ็นเซอร์วัดค่าความชื้นและอุณหภูมิ (AMI 1001) และเซ็นเซอร์วัดความเข้มของแสง (ZX-02F Light) เมื่อมีการตรวจจับได้ค่าได้ ก็จะทำการส่ง ข้อมูลไปยังโยโย่บอร์ด แล้วจะมีการรับค่าผ่าน Bluetooth แล้วทา แสดงผลหน้าจอแอปพลิเคชั่น เพื่อให้ผู้ใช้งานทำการเปิด-ปิด โซลินอยด์วาล์วและปั๊มน้ำตามต้องการ และใช้คำสั่งเดิมที่เปลี่ยนค่าให้หยุดเมื่อ ตรวจพบว่าความชื้นและอุณหภูมิเข้าสู่ค่าที่เหมาะสม

มหาวิทยาลัยเทคโนโลยีสุรนารี

กิตติกรรมประกาศ

จากการที่คณะจัดทำรายงานได้รับมอบหมายให้ทำโครงงานเรื่องการตรวจวัดค่าสภาพแวดล้อมในโรงเรือนกล้วยไม้ส่งผลให้คณะจัดทำรายงานได้รับความรู้และประสบการณ์ต่างๆ เกี่ยวกับการเขียนโปรแกรมด้วยโปรแกรมแอนดรอยด์สตูดิโอ (Android Studio 1.0) เป็นอย่างมาก บัดนี้โครงงานดังกล่าวพร้อมทั้งรายงานได้สำเร็จลงแล้ว ทั้งนี้ด้วยความร่วมมือและสนับสนุนจากบุคคลต่างๆ ดังนี้

1. ผศ. รอ. ดร. ประโยชน์ คำสวัสดิ์ (อาจารย์ที่ปรึกษาโครงงาน)
2. นายเกริกฤทธิ์ ศรีเคน (นักศึกษาปริญญาโท สาขาวิชาวิศวกรรมโทรคมนาคม)

ข้าพเจ้าใคร่ขอขอบพระคุณผู้ที่มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูลและเป็นที่ปรึกษาในการทำรายงานฉบับนี้จนเสร็จสมบูรณ์ตลอดจนให้การดูแลและให้ความเข้าใจเกี่ยวกับพื้นฐานการใช้งานโปรแกรม ซึ่งข้าพเจ้าขอขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ด้วย

นางสาวนวรัตน์ แนบชัยภูมิ
นางสาวกรรณิการ์ รุ่งอรุณ
นางสาวสุชาดา กัณหา

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ

หน้า

สารบัญ
สารบัญภาพ
สารบัญตาราง

บทที่1 บทนำ

1.1 บทนำ
1.2 ความเป็นมาและความสำคัญของปัญหา 2
1.3 วัตถุประสงค์ 2
1.4 ขอบเขตของโครงงาน 3
1..5 ประโยชน์ที่ได้รับ 3

บทที่2 ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 ระบบสมองกลฝังตัว (Embedded System) 4
2.2 ระบบปฏิบัติการแอนดรอยด์ (Android Operating System) 5
2.3 โยโย่ บอร์ด (IOIO Board) 7
2.3.1 ภาษา JAVA 10
2.4 เซ็นเซอร์สำหรับตรวจวัดทางการเกษตร
2.4.1 AMT 1001 เซ็นเซอร์อุณหภูมิและความชื้น
(Temperature And Humidty Sensors Analog Voltage Output) 11
2.4.2 เซ็นเซอร์ตรวจวัดความเข้มแสง (ZX-02F Light Sensor) 13
2.5 โมดูลรีเลย์ 2 ช่อง (2 channel Relay Module) 14
2.6 โซลินอยด์วาล์ว (Solenoid Valve) 15

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ (ต่อ)

	หน้า
บทที่ 3 การออกแบบโครงงาน	
3.1 บทนำ	16
3.2 การออกแบบฮาร์ดแวร์	18
3.2.1 การออกแบบตู้ควบคุมขนาดเล็ก	24
3.2.2 การออกแบบตู้จำลอง	29
3.3 การออกแบบซอฟต์แวร์	30
3.3.1 การออกแบบหน้าแอปพลิเคชั่น	30
3.3.2 การรันโค้ดคำสั่งการอ่านค่า Sensor (ความชื้น อุณหภูมิ และความเข้มแสง)	31
3.3.3 การเขียนโค้ดคำสั่งการเปิด-ปิด Relay	33
3.3.4 การเขียนคำสั่งการส่งค่าข้อมูลขึ้น Thingspeak	34
บทที่ 4 ผลการทดลองและการวิเคราะห์ผลการทดลอง	
4.1 วิธีการทดสอบ	
4.1.1 การทดลองที่ 1	35
4.1.2 การทดลองที่ 2	39
บทที่ 5 สรุปผลโครงงานและข้อเสนอแนะ	
5.1 สรุปผลโครงงาน	40
5.2 ปัญหาที่พบ	40
5.3 ข้อเสนอแนะ	40
เอกสารอ้างอิง	41

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูป

ภาพที่ **หน้า**

2.1 ระบบสมองกลฝังตัว 4

2.2 Crashes by OS Version Normalized (12/1 – 12/15) 7

2.3 การเขียนโปรแกรมเชื่อมต่อกับแอนดรอยด์ 8

2.4 การสั่งงานบอร์ด IOIO 9

2.5 IOIO Board 9

2.6 เซ็นเซอร์ AMT 1001 13

2.7 ZX-02F Light Sensor เซ็นเซอร์ตรวจวัดความเข้มแสง 13

2.8 โมดูลรีเลย์ 2 ช่อง 14

2.9 โซลินอยด์วาล์ว 15

3.1 รูปแบบการทำงานของชุดทดลอง 16

3.2 หน้าแอปพลิเคชั่นการเชื่อมต่อบอร์ดโยโย่กับสมาร์ทโฟน 23

3.2.1 กล่องควบคุมระบบขนาดเล็ก 24

3.2.1-1 อุปกรณ์การเชื่อมต่อภายในกล่องควบคุมขนาดเล็ก 25

3.2.2 การออกแบบตู้จำลอง 29

3.3.2 การแสดงค่าความชื้น อุณหภูมิและความเข้มแสง 32

3.3.3 แสดงการสั่งเปิด-ปิดรีเลย์บอร์ด 2 ช่อง 33

4.1 ตู้ออกแบบควบคุมขนาดเล็กขณะทำการเก็บค่า 38

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญตาราง

ตาราง **หน้า**

ตารางที่ 4.1.2 การทดลองใช้คำสั่ง ON/OFF ผ่านแอปพลิเคชั่นเพื่อเปิด-ปิด โซลินอยด์วาล์วและปั๊มน้ำจำลองผ่านทาง Bluetooth 39

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 1
บทนำ

1.1 บทนำ

โลกของการติดต่อสื่อสารในปัจจุบันได้มีการพัฒนาที่ก้าวหน้าเป็นอย่างมาก โดยเฉพาะการสื่อสารแบบไร้สาย ที่ได้มีการพัฒนาความเร็วในการรับส่งข้อมูลที่สูงขึ้น จากเดิมที่มีการส่งได้เพียงข้อความสั้น (SMS :Short Message Service) และ MMS(Multimedia Messaging Service) ปัจจุบันสามารถทำการโทรศัพท์แบบเห็นหน้าคู่สนทนากันได้ (Video Call) แต่ต้องผ่านทางระบบของวายฟาย Wi-Fi (Wireless Fidelity) หรือ ระบบ 3G (Third Generation Of Mobile Telephone) ซึ่งสำหรับประเทศไทยแล้ว อุปกรณ์มือถือ และอุปกรณ์พกพา ส่วนมากในตลาดจะรองรับระบบการรับส่งข้อมูลความเร็วสูงเป็นที่เรียบร้อยแล้ว และได้รับความนิยมจากผู้ใช้งานเป็นจำนวนมาก โดยอุปกรณ์ส่วนใหญ่ที่มีอยู่ในท้องตลาด จะมีระบบปฏิบัติการเป็นของตัวเอง ที่ไม่เหมือนกับระบบปฏิบัติการที่อยู่บนคอมพิวเตอร์ส่วนบุคคล (PC :Personal Computer) ส่งผลให้แนวทางในการพัฒนาโปรแกรม เพื่อนำไปใช้งานบนอุปกรณ์เหล่านั้นยุ่งยาก และหลากหลายขึ้น ระบบปฏิบัติการบนอุปกรณ์ดังกล่าว มีอยู่หลายตัวกันเช่น Android, IOS, Windows Phone, BlackBerry, Symbian, WebOS, MeeGo และ QNX เป็นต้น โดยลักษณะของระบบปฏิบัติการข้างต้นส่วนมากจะเป็นประเภทไม่เปิดเผยซอฟต์แวร์ต้นฉบับ (Closed Source) ซึ่งหมายความว่าระบบปฏิบัติการดังกล่าว ไม่สามารถนำมาศึกษา ดัดแปลงการทำงานของระบบปฏิบัติการเพื่อนำไปใช้งานตามที่ต้องการได้ ทำให้เกิดความไม่สะดวกในการพัฒนา และการพัฒนาจะถูกกำหนดทิศทางโดยบริษัทเจ้าของลิขสิทธิ์

แอนดรอยด์ (Android) คือระบบปฏิบัติการแบบเปิดเผยซอฟต์แวร์ต้นฉบับ (Open Source) โดยบริษัท กูเกิ้ล (Google Inc.) ที่ได้รับความนิยมเป็นอย่างสูง เนื่องจากอุปกรณ์ที่ใช้ระบบปฏิบัติการแอนดรอยด์ มีจำนวนมาก อุปกรณ์มีหลากหลายระดับ หลายราคา รวมทั้งสามารถทำงานบนอุปกรณ์ที่มีขนาดหน้าจอ และความละเอียดแตกต่างกันได้ ทำให้ผู้บริโภคสามารถเลือกได้ตามต้องการ และหากมองในทิศทางสำหรับนักพัฒนาโปรแกรม (Programmer) แล้วนั้น การพัฒนาโปรแกรมเพื่อใช้งานบนระบบปฏิบัติการแอนดรอยด์ ไม่ใช่เรื่องที่ยาก เพราะมีข้อมูลในการพัฒนารวมทั้ง Android SDK (Software Development Kit) เตรียมไว้ให้กับนักพัฒนาได้เรียนรู้ และเมื่อนักพัฒนาต้องการจะเผยแพร่หรือจำหน่ายโปรแกรมที่พัฒนาแล้วเสร็จ แอนดรอยด์ก็ยังมีตลาดในการเผยแพร่โปรแกรม ผ่าน Android Market แต่หากจะกล่าวถึงโครงสร้างภาษาที่ใช้ในการพัฒนานั้น สำหรับ Android SDK จะยึดโครงสร้างของภาษาจาวา (Java language) ในการเขียน

โปรแกรม เพราะโปรแกรมที่พัฒนามาได้จะต้องทำงานอยู่ภายใต้ Dalvik Virtual Machine เช่นเดียวกับโปรแกรมจาวา ที่ต้องทำงานอยู่ภายใต้ Java Virtual Machine (Virtual Machine เปรียบได้กับสภาพแวดล้อมที่โปรแกรมทำงานอยู่)

นอกจากนั้นแล้ว แอนดรอยด์ ยังมีโปรแกรมแกรมที่เปิดเผยซอร์ฟแวร์ต้นฉบับ (Open Source) เป็นจำนวนมาก ทำให้นักพัฒนาที่สนใจ สามารถนำซอร์ฟแวร์ต้นฉบับ มาศึกษาได้อย่างไม่ยาก ประกอบกับความนิยมของแอนดรอยด์ได้เพิ่มขึ้นอย่างมากใน โดยดูได้จากส่วนแบ่งการตลาด

1.2 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันนั้นเทคโนโลยีนั้นได้มีการพัฒนาอย่างต่อเนื่องและรวดเร็วอย่างก้าวกระโดด โดยหนึ่งในเทคโนโลยีที่ได้เกิดขึ้นมานั้นคือ สมาร์ทโฟนและบลูทูธ ด้วยความอัจฉริยะและความสามารถต่างๆที่มีมากมายโดยรอบด้านของสมาร์ทโฟนและบลูทูธนี้เอง มันจึงกระจายสู่ผู้ใช้ทุกเพศทุกวัยอย่างรวดเร็ว เรียกได้ว่า แทบจะทุกคนเลยจะต้องมีสมาร์ทโฟนหรือบลูทูธติดตัว

สำหรับผู้ที่ทำงานในฟาร์มเป็นประจำนั้นผู้จัดทำคิดว่าควรที่จะนำเทคโนโลยีสมัยใหม่ คือ สมาร์ทโฟนหรือแทบเล็ตมาใช้ในการดูแลฟาร์มเพื่อให้เกิดความสะดวกสบาย ง่ายต่อการควบคุม เพราะจากปัญหาที่ผู้จัดทำประสบก็คือ เวลาที่จะรดน้ำต้นไม้นั้นผู้ใช้จะเสียเวลาในการเดินไปเปิด-ปิดปั๊มน้ำ เพื่อเป็นการที่จะลดเวลาที่เสียไปตรงนั้น ผู้จัดทำจึงได้มีความคิดริเริ่มที่จะพัฒนาแอพพลิเคชั่นบนสมาร์ทโฟน โดยเน้นไปที่ระบบปฏิบัติการแอนดรอยด์เพราะเปิดกว้างสำหรับคนที่ต้องการพัฒนาแอพพลิเคชั่น และมีผู้ใช้ที่มีจำนวนมาก และมีแนวโน้มเพิ่มขึ้นเรื่อยๆ

มหาวิทยาลัยเทคโนโลยีสุรนารี

1.3 วัตถุประสงค์

1. ศึกษาการควบคุมการเปิด/ปิดน้ำผ่านแอปพลิเคชั่นบนสมาร์ทโฟน (Smart Phone) ที่ทำงานบนระบบปฏิบัติการแอนดรอยด์ด้วยภาษาจาวา
2. เพื่อศึกษาการใช้ IOIO Broad และการเขียนแอนดรอยด์โดยใช้ Android Studio
3. ออกแบบชุดควบคุมการเปิด/ปิดน้ำด้วย IOIO Broad, บลูทูธดองเกิล, เซ็นเซอร์ AMT 1001 และโซลินอยด์วาล์ว
4. สร้างชุดควบคุมการเปิด/ปิดน้ำด้วย IOIO Broad, บลูทูธดองเกิล, เซ็นเซอร์ AMT 1001, โซลินอยด์วาล์วและเขียนโปรแกรมที่ใช้ควบคุม IOIO Broad
5. ทดสอบและวิเคราะห์การทำงานของอุปกรณ์การเปิด - ปิดน้ำ

1.4 ขอบเขตของโครงงาน

1. จำลองการรดน้ำต้นไม้ในแบบจำลอง
2. ทดสอบเก็บข้อมูลความชื้น อุณหภูมิและแสงภายในระยะเวลา 48 ชั่วโมง
3. ควบคุมการเปิด/ปิดน้ำ จากสมาร์ทโฟนที่ทำงานบนระบบปฏิบัติการแอนดรอยด์ โดยใช้โยโย่บอร์ดควบคุม เพื่อทำหน้าที่เป็นสวิตซ์ในการควบคุมโซลินอยด์วาล์วและปั๊มน้ำ

1.5 ประโยชน์ที่ได้รับ

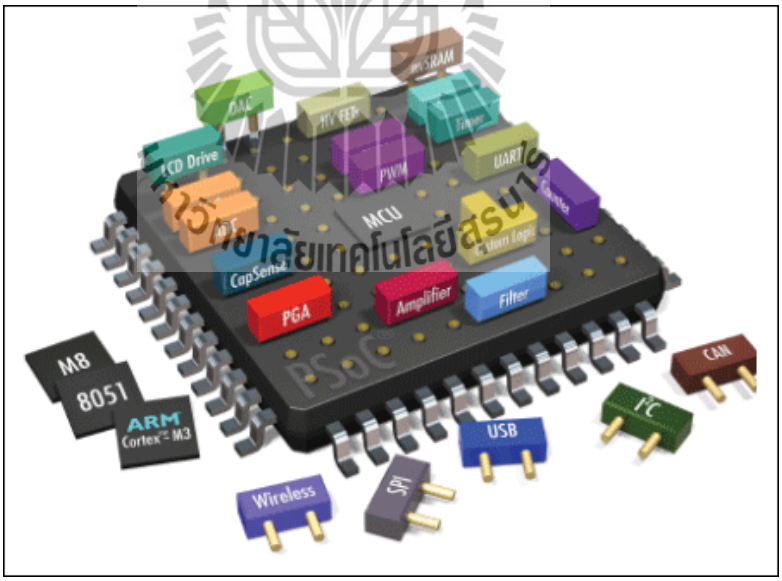
1. ความรู้เกี่ยวกับการใช้ IOIO Board
2. ความรู้เกี่ยวกับการใช้แอปพลิเคชั่นที่ทำงานบนระบบปฏิบัติการแอนดรอยด์
3. เพิ่มความสะดวกสบายแก่ผู้ที่รดน้ำต้นไม้
4. เข้าใจหลักการของโซลินอยด์วาล์วที่ใช้ในการเปิด –ปิดปั๊มน้ำ

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 ระบบสมองกลฝังตัว (Embedded System)

ระบบสมองกลฝังตัว หมายถึง ระบบคอมพิวเตอร์ขนาดจิ๋วที่ใช้อุปกรณ์ควบคุม ได้แก่ ไมโครคอนโทรลเลอร์ และไมโครโพรเซสเซอร์ โดยนำไปฝังไว้ในอุปกรณ์เครื่องมือเครื่องใช้ไฟฟ้าและเครื่องเล่นอิเล็กทรอนิกส์ต่างๆ ทั้งนี้เพื่อเพิ่ม ความฉลาดและความสามารถต่างๆให้กับอุปกรณ์เหล่านั้น ระบบสมองกลฝังตัวแม้ไม่ใช่เครื่องคอมพิวเตอร์ แต่ก็มีระบบคอมพิวเตอร์อยู่ภายในอาจจะเป็นเพียงไมโครโพรเซสเซอร์ (Microprocessor) หรือ ชิป (Chip) ธรรมดาหรือโพรเซสเซอร์ (Processor) ที่ประกอบด้วยชิป ที่มีวงจรซับซ้อน โดยจะมีหลักการทำงาน คือมีสัญญาณข้อมูลเข้า (Input) จากอุปกรณ์เซ็นเซอร์ (Sensor) เข้าสู่ระบบและมีสัญญาณผลลัพธ์ (Output) ของระบบไปควบคุมบังคับ (Actuator) สวิตซ์ของเครื่องควบคุมต่างๆ เช่นสวิตซ์เครื่องจักรหรือวาล์วควบคุมทิศทางการไหลของท่อต่างๆ ดังแสดงในรูปที่ 2.1



รูปที่ 2.1 ระบบสมองกลฝังตัว (Embeded System) [1]

ระบบปฏิบัติการสำหรับสมองกลฝังตัว

ในการพัฒนาระบบสมองกลฝังตัวอาจจะมีการใช้ระบบปฏิบัติการเป็นแกนหลักในการพัฒนาหรือไม่มีการใช้ในการพัฒนาก็ได้ ระบบปฏิบัติการสำหรับระบบสมองกลฝังตัวมีหลายประเภทมากตั้งแต่ RTOS , UcOS-II จนไปถึงระบบปฏิบัติการที่มีขนาดใหญ่ขึ้นมาเช่น Linux, Windows CE จนถึงระบบปฏิบัติการสมัยใหม่ที่มีการพัฒนา เช่น MeeGo Android

ภาษาที่ใช้ในการพัฒนาของระบบสมองกลฝังตัว

ในปัจจุบันมีภาษาโปรแกรมต่างๆมากมายที่ใช้ในการพัฒนาระบบสมองกลฝังตัวเช่นภาษา Assembly ภาษา C ,C++ หรือภาษาระดับสูงที่ถูกนำมาใช้ในการพัฒนาระบบสมองกลฝังตัวที่มีระบบปฏิบัติการเช่น JAVA หรือ Python โดยผู้ใช้สามารถเลือกใช้ภาษาในการพัฒนาระบบสมองกลฝังตัวได้ตามความเหมาะสม

2.2 ระบบปฏิบัติการแอนดรอยด์ (Android Operating System)

แอนดรอยด์ (Android) เป็นระบบปฏิบัติการที่มีพื้นฐานอยู่บนลินุกซ์ ถูกออกแบบมาสำหรับอุปกรณ์ที่ใช้จอสัมผัส เช่นสมาร์ทโฟน และแท็บเล็ตคอมพิวเตอร์ ถูกคิดค้นและพัฒนาโดยบริษัทแอนดรอยด์ (Android, Inc.) ซึ่งต่อมากูเกิลได้ทำการซื้อต่อบริษัทในปี พ.ศ. 2548 แอนดรอยด์ถูกเปิดตัวเมื่อ ปี พ.ศ. 2550 พร้อมกับการก่อตั้งโอเพนแฮนด์เซตอัลไลแอนซ์ ซึ่งเป็นกลุ่มของบริษัทผลิตฮาร์ดแวร์, ซอฟต์แวร์ และการสื่อสารคมนาคม ที่ร่วมมือกันสร้างมาตรฐานเปิด สำหรับอุปกรณ์พกพาโดยสมาร์ทโฟนที่ใช้ระบบปฏิบัติการแอนดรอยด์เครื่องแรกของโลกคือ เอชทีซี ดรีม วางจำหน่ายเมื่อปี พ.ศ. 2551

แอนดรอยด์ (Android) คือระบบปฏิบัติการแบบเปิดเผยซอร์ฟแวร์ต้นฉบับ (Open Source) โดยบริษัท กูเกิ้ล (Google Inc.) ที่ได้รับความนิยมเป็นอย่างสูง เนื่องจากอุปกรณ์ที่ใช้ระบบปฏิบัติการแอนดรอยด์ มีจำนวนมาก อุปกรณ์มีหลากหลายระดับ หลายราคา รวมทั้งสามารถทำงานบนอุปกรณ์ที่มีขนาดหน้าจอ และความละเอียดแตกต่างกันได้ ทำให้ผู้บริโภคสามารถเลือกได้ตามต้องการ

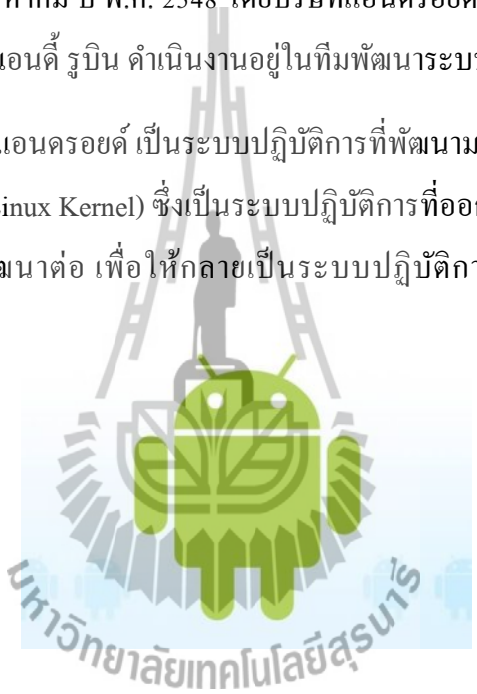
บริษัทวิจัย Kantar เผยแพร่ผลสำรวจสะท้อนยอดขายอุปกรณ์พกพาใน 9 ตลาดใหญ่ทั่วโลก ซึ่งผลสำรวจก็ยังคงเป็น Android ที่ครองอันดับหนึ่งทุกพื้นที่เหมือนเดิม และต้องปรบมือให้ Windows Phone ที่มียอดสูงขึ้นในตลาดโมบาย

บริษัทวิจัยการตลาด Kantar ทำการสำรวจตลาดหลัก 9 แห่งของโลก ได้แก่ ออสเตรเลีย, จีน, ฝรั่งเศส, เยอรมนี, อิตาลี, ญี่ปุ่น, สเปน, อเมริกา และอังกฤษ ซึ่งผลที่ได้ก็แสดงให้เห็นว่า Android ครองส่วนแบ่งไป 64.2% ของยอดขายอุปกรณ์พกพาในไตรมาสแรกของปีนี้

ประวัติความเป็นมา

เริ่มต้นระบบปฏิบัติการแอนดรอยด์ ถูกพัฒนามาจากบริษัทแอนดรอยด์ (Android Inc.) เมื่อปี พ.ศ. 2546 โดยมีนาย แอนดี้ รูบิน (Andy Rubin) ผู้ให้กำเนิดระบบปฏิบัติการนี้ และถูกบริษัท กูเกิ้ล ซื้อกิจการเมื่อเดือนสิงหาคม ปี พ.ศ. 2548 โดยบริษัทแอนดรอยด์ ได้กลายเป็นมาบริษัทลูก ของบริษัทกูเกิ้ล และยังมีนาย แอนดี้ รูบิน ดำเนินงานอยู่ในทีมพัฒนาระบบปฏิบัติการต่อไป

ระบบปฏิบัติการแอนดรอยด์ เป็นระบบปฏิบัติการที่พัฒนามาจากการนำเอา แกนกลางของระบบปฏิบัติการลินุกซ์ (Linux Kernel) ซึ่งเป็นระบบปฏิบัติการที่ออกแบบมาเพื่อทำงานเป็นเครื่องให้บริการ (Server) มาพัฒนาต่อ เพื่อให้กลายเป็นระบบปฏิบัติการบนอุปกรณ์พกพา (Mobile Operating System)



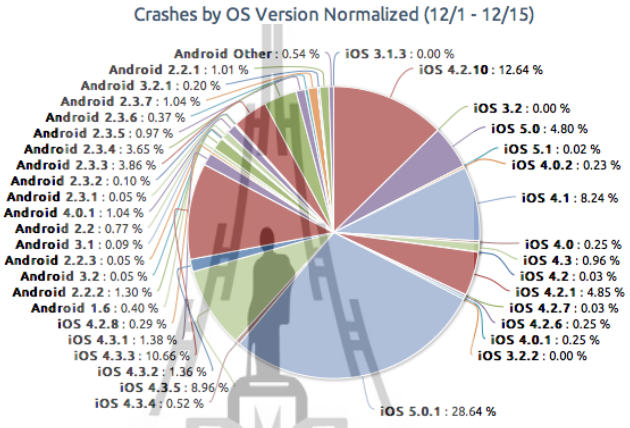
ข้อเด่นของแอนดรอยด์

เนื่องจากระบบปฏิบัติการแอนดรอยด์มีการเจริญเติบโตอย่างรวดเร็ว และมีส่วนแบ่งตลาดของอุปกรณ์ด้านนี้ ขึ้นทุกขณะ ทำให้กลุ่มผู้ใช้งาน และกลุ่มนักพัฒนาโปรแกรม ให้ความสำคัญกับระบบปฏิบัติการแอนดรอยด์เพิ่มมากขึ้น

เมื่อมองในด้านของกลุ่มผลิตภัณฑ์ บริษัทที่มีการพัฒนาผลิตภัณฑ์รุ่นใหม่ ได้มีการนำเอาระบบปฏิบัติการแอนดรอยด์ไปใช้ในสินค้าของตนเอง พร้อมทั้งยังมีการปรับแต่งให้ระบบปฏิบัติการมีความสามารถ การจัดวาง โปรแกรม และลูกเล่นใหม่ๆ ที่แตกต่างจากคู่แข่งในท้องตลาด โดยเฉพาะอย่างยิ่ง กลุ่มสินค้าที่เป็น มือถือรุ่นใหม่ (Smart Phone) และอุปกรณ์จอสัมผัส

(Touch Screen) โดยมีคุณลักษณะแตกต่างกันไป เช่นขนาดหน้าจอ ระบบโทรศัพท์ ความเร็วของหน่วยประมวลผล ปริมาณหน่วยความจำ แม้กระทั่งอุปกรณ์ตรวจจับต่างๆ (Sensor)

หากมองในด้านของการพัฒนาโปรแกรม ทางบริษัทกูเกิ้ล ได้มีการพัฒนา Application Framework ไว้สำหรับนักพัฒนาใช้งาน ได้อย่างสะดวก และไม่เกิดปัญหาเมื่อนำชุดโปรแกรมที่พัฒนาขึ้นมา ไปใช้กับอุปกรณ์ที่มีคุณลักษณะต่างกัน เช่นขนาดจออุปกรณ์ ไม่เท่ากัน ก็ยังสามารถใช้งานโปรแกรมได้เหมือนกัน เป็นต้น ดังแสดงในรูปที่ 2.2



รูปที่ 2.2 Crashes by OS Version Normalized (12/1 – 12/15) [2]

มหาวิทยาลัยเทคโนโลยีสุรนารี

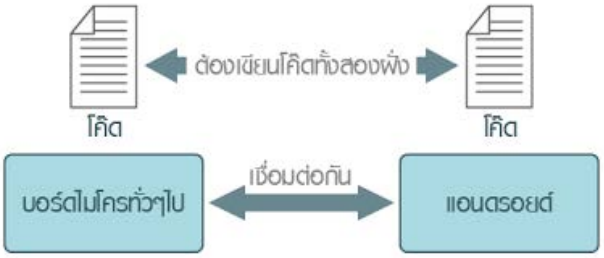
2.3 โยโย่บอร์ด (IOIO Board)

บอร์ด IOIO คืออะไร

บอร์ด IOIO (โยโย) เป็นบอร์ดไมโครคอนโทรลเลอร์ตัวหนึ่งที่ถูกสร้างขึ้นโดยฝีมือของ YTAI Ben-Tsvi (ชื่ออ่านว่า อีทาย) ซึ่งเป็นวิศวกรชาวอิสราเอลของบริษัท Google นั่นเอง

สำหรับบอร์ด IOIO นั่นเป็นบอร์ดไมโครคอนโทรลเลอร์ที่เดิมทีเกิดมาเพื่อเชื่อมต่อกับแอนดรอยด์โดยเฉพาะ โดยต่างจากไมโครคอนโทรลเลอร์ตัวอื่นๆ เพราะปกติแล้วการเขียนโปรแกรมเชื่อมต่อกับแอนดรอยด์ ไม่ว่าจะใช้บอร์ดไมโครคอนโทรลเลอร์ตัวใดก็ตาม จะต้องเขียนโปรแกรม

ให้กับไมโครคอนโทรลเลอร์และต้องเขียนแอปพลิเคชั่นบนแอนดรอยด์ เพื่อให้สามารถเชื่อมต่อและส่งข้อมูลระหว่างกันได้ ดังรูปที่ 2.3



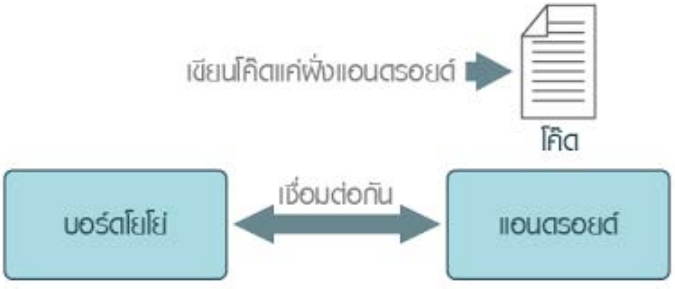
รูปที่ 2.3 การเขียนโปรแกรมเชื่อมต่อกับแอนดรอยด์

บอร์ดไมโครทั่วๆไปก็รวมไปถึง Arduino ด้วยเช่นกัน ผู้ที่หลงเข้ามาอ่านหลายๆคนชอบเข้าใจกันว่าบอร์ดไมโครที่เชื่อมต่อกับแอนดรอยด์ได้นั้น จะมีแค่บอร์ด IOIO และ Arduino เท่านั้น

ซึ่งจริงๆแล้วไม่ใช่เลย บอร์ดไมโครทั่วไปก็ทำได้ ทั้งนี้ขึ้นอยู่กับวิธีในการเชื่อมต่อ ถ้าจะง่ายสุดก็เป็นบลูทูธ จึงเป็นความเข้าใจแบบผิดๆว่าต้อง Arduino หรือ IOIO เท่านั้น บอร์ดทั่วๆไปนี้จะหมายถึงบอร์ดที่ไม่ได้เกิดมาเพื่อแอนดรอยด์โดยตรง

บอร์ด IOIO จะแตกต่างจากบอร์ดทั่วๆไปตรงจุดนี้นั่นเอง เพราะเกิดมาเพื่อเชื่อมต่อและถูกสั่งงานจากแอนดรอยด์ ไม่สามารถทำงานได้ด้วยตัวเอง ต้องรอคำสั่งจากแอนดรอยด์เท่านั้น เนื่องจากการที่เกิดมาเพื่อแอนดรอยด์ ผู้พัฒนาจึงทำให้ผู้ใช้งานไม่จำเป็นต้องเขียนโค๊ดให้กับบอร์ด IOIO เลย เพราะจะมีโค๊ดใส่มาในบอร์ดให้พร้อมไว้เรียบร้อยแล้ว (หรือที่เรียกกันว่าเฟิร์มแวร์นั่นเอง)

ดังนั้นผู้ใช้งานจึงเขียนโค๊ดแค่ฝั่งแอนดรอยด์เท่านั้น โดยผู้ผลิตจะมีไลบรารีของบอร์ด IOIO ให้ใช้ในโค๊ดฝั่งแอนดรอยด์เลย ดังนั้นจึงสามารถสั่งงานบอร์ด IOIO ด้วยคำสั่งในแอปพลิเคชั่น ดังรูปที่ 2.4

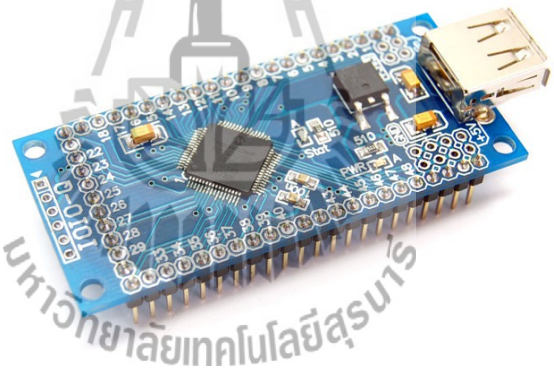


รูปที่ 2.4 การสั่งงานบอร์ด IOIO

ด้วยจุดดีข้อนี้ จึงทำให้ง่ายต่อการนำไปใช้งาน ไม่ต้องวุ่นวายกับการเชื่อมต่อ เพราะเฟิร์มแวร์ทำให้หมด และจะทำงานทันทีที่เชื่อมต่อกับแอนดรอยด์ แต่นั้นก็กลายเป็นข้อเสียอีกอย่างหนึ่งเช่นกัน เพราะว่าจะไม่สามารถทำงานด้วยตัวเองได้ (Standalone) เนื่องด้วยวิธีการทำงานของบอร์ด IOIO นั่นเอง ที่ทำงานแบบ Realtime คือต้องรอแอนดรอยด์สั่งงานทุกครั้ง ไม่ได้รับโค๊ดทั้งหมดจากแอนดรอยด์แล้วมาทำงานเองทั้งหมด แต่จะรอคำสั่งจากแอนดรอยด์แล้วทำคำสั่งนั้นๆทีละคำสั่งเรื่อยๆ เมื่อทำคำสั่งนั้นๆเสร็จแล้ว แอนดรอยด์ก็จะสั่งให้ทำคำสั่งต่อไปเรื่อยๆ จึงสรุปได้ง่ายๆว่าบอร์ด IOIO เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ใช้สำหรับเชื่อมต่อกับแอนดรอยด์แล้วทำงานตามคำสั่งในแอนดรอยด์ สำหรับคำสั่งที่จะสั่งงานผู้ใช้ก็ต้องเขียนขึ้นมาเป็นแอพพลิเคชั่นแทน

<u>คุณสมบัติของบอร์ด IOIO</u>

บอร์ด IOIO สร้างขึ้นโดยใช้ชิป PIC ตระกูล PIC24FJ โดยตัวบอร์ดจะมีขา I/O ให้ใช้งานได้มากถึง 48 ขา (เต็มที่เลยทีเดียว) แต่ละขามีคุณสมบัติพิเศษด้วย เช่น Analog, I2C, หรือ UART เป็นต้น ซึ่งก็เหมือนบอร์ดไมโครคอนโทรลเลอร์ทั่วๆไปนั่นเอง เพียงแต่ว่าเขียนโค้ดที่เป็นเฟิร์มแวร์ใส่ลงในชิปให้แล้ว แสดง



มหาวิทยาลัยเทคโนโลยีสุรนารี

2.3.1 ภาษา JAVA

Java เป็นภาษาโปรแกรมที่ใช้ในการเขียนคำสั่งสั่งงานคอมพิวเตอร์ ซึ่งพัฒนาขึ้นโดยบริษัท ซันไมโครซิสเต็มส์ จำกัด (Sun Microsystems Inc.) ในปี ค.ศ. 1991 เป็นส่วนหนึ่งของโครงการวิจัยเพื่อพัฒนาซอฟต์แวร์ สำหรับอุปกรณ์อิเล็กทรอนิคส์ต่างๆ เช่น โทรทัศน์ โทรศัพท์มือถือ โดยมีเป้าหมายการทำงานเชื่อมต่อกับอุปกรณ์ฮาร์ดแวร์ต่างๆได้อย่างกว้างขวางและมีประสิทธิภาพ ใช้เวลาน้อย รวดเร็วในการพัฒนาโปรแกรม และสามารถเชื่อมต่อไปยังแพล็ตฟอร์ม (Platform) อื่นๆได้ง่าย

Java เป็นภาษาสำหรับเขียนโปรแกรมภาษาหนึ่งที่มีลักษณะสนับสนุนการเขียนโปรแกรมเชิงวัตถุ (OOP : Object-Oriented Programming) ที่ชัดเจน โปรแกรมต่าง ๆ ถูกสร้างภายในคลาส (Class) โปรแกรมเหล่านั้นเรียกว่า Method หรือ Behavior โดยปกติจะเรียกแต่ละ Class ว่า Object โดยแต่ละ Object มีพฤติกรรมมากมาย โปรแกรมที่สมบูรณ์จะเกิดจากหลาย Object หรือหลาย Class มารวมกัน โดยแต่ละ Class จะมี Method หรือ Behavior แตกต่างกันไป

ข้อดีของ ภาษา Java

1. ภาษา Java เป็นภาษาที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุแบบสมบูรณ์ ซึ่งเหมาะสำหรับพัฒนาระบบที่มีความซับซ้อน การพัฒนาโปรแกรมแบบวัตถุจะช่วยให้เราสามารถใช้คำหรือชื่อต่างๆ ที่มีอยู่ในระบบงานนั้นมาใช้ในการออกแบบโปรแกรมได้ ทำให้เข้าใจได้ง่ายขึ้น

2. โปรแกรมที่เขียนขึ้นโดยใช้ภาษา Java จะมีความสามารถทำงานได้ในระบบปฏิบัติการที่แตกต่างกัน ไม่จำเป็นต้องดัดแปลงแก้ไขโปรแกรม เช่น หากเขียนโปรแกรมบนเครื่อง Sun โปรแกรมนั้นก็สามารถถูก Compile และ Run บนเครื่องพีซีธรรมดาได้

3.ภาษา Java มีการตรวจสอบข้อผิดพลาดทั้งตอน Compile Time และ Runtime ทำให้ลดข้อผิดพลาดที่อาจเกิดขึ้นในโปรแกรม และช่วยให้ Debug โปรแกรมได้ง่าย

4. ภาษา Java มีความซับซ้อนน้อยกว่าภาษา C++ เมื่อเปรียบเทียบ Code ของโปรแกรมที่เขียนขึ้นโดยภาษา Java กับ C++ พบว่า โปรแกรมที่เขียนโดยภาษา Java จะมีจำนวน Code น้อยกว่าโปรแกรมที่เขียนโดยภาษา C++ ทำให้ใช้งานได้ง่ายกว่าและลดความผิดพลาดได้มากขึ้น

5. ภาษา Java ถูกออกแบบมาให้มีความปลอดภัยสูงตั้งแต่แรก ทำให้โปรแกรมที่เขียนขึ้นด้วยจาวามีความปลอดภัยมากกว่าโปรแกรมที่เขียนขึ้นด้วยภาษาอื่น เพราะ Java มี Security ทั้ง Low Level และ High Level ได้แก่ Electronic Signature, Public And Private Key Management, Access Control และ Certificates

6.มี IDE, Application Server, และ Library ต่างๆ มากมายสำหรับจาวาที่เราสามารถใช้งานได้โดยไม่ต้องเสียค่าใช้จ่าย ทำให้เราลดค่าใช้จ่ายที่ต้องเสียไปกับการซื้อ Tool และ S/W ต่าง ๆ

ข้อเสียของ ภาษา Java

1.ทำงานได้ช้ากว่า Native Code (โปรแกรมที่ Compile ให้อยู่ในรูปของภาษาเครื่อง) หรือโปรแกรมที่เขียนขึ้นด้วยภาษาอื่น อย่างเช่น C หรือ C++ ทั้งนี้ก็เพราะว่าโปรแกรมที่เขียนขึ้นด้วยภาษา Java จะถูกแปลงเป็นภาษากลางก่อน แล้วเมื่อโปรแกรมทำงานคำสั่งของภาษากลางนี้จะถูกเปลี่ยนเป็นภาษาเครื่องอีก ทีหนึ่ง ทีล่ะคำสั่ง (หรือกลุ่มของคำสั่ง) ณ Runtime ทำให้ทำงานช้ากว่า Native Code ซึ่งอยู่ในรูปของภาษาเครื่องแล้วตั้งแต่ Compile โปรแกรมที่ต้องการความเร็วในการทำงานจึงไม่นิยมเขียนด้วยจาวา

2.Tool ที่มีในการใช้พัฒนาโปรแกรมจาวามักไม่ค่อยเก่ง ทำให้หลายอย่างโปรแกรมเมอร์จะต้องเป็นคนทำเอง ทำให้ต้องเสียเวลาทำงานในส่วนที่ Tool ทำไม่ได้ ถ้าเราดู Tool ของ MS จะใช้งานได้ง่ายกว่า และพัฒนาได้เร็วกว่า (แต่เราต้องซื้อ Tool ของ MS และก็ต้องรันบน Platform ของ MS)

3. โปรแกรมไม่อาจรันได้ด้วยตนเอง ต้องอาศัย Java Runtime

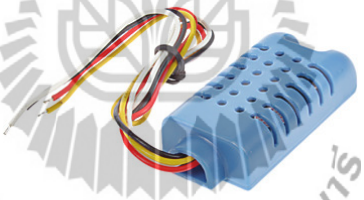
4. ความเร็วในการทำงาน ยังไม่เท่า C/C++

5. ไม่อาจติดต่ออุปกรณ์ได้เองโดยตรง

2.4 เซ็นเซอร์สำหรับตรวจวัดทางการเกษตร

2.4.1 AMT 1001 เซ็นเซอร์อุณหภูมิและความชื้น (Temperature And Humidity Sensors Analog Voltage Output)

AMT 1001 Temperature And Humidity Sensors Analog Voltage Output (เซ็นเซอร์อุณหภูมิและเซ็นเซอร์ความชื้น) AMT 1001 ให้สัญญาณค่าอุณหภูมิและความชื้นออกมาเป็นค่าแบบอะนาล็อก และให้แรงดัน ดังรูปที่ 2.6



มหาวิทยาลัยเทคโนโลยีสุรนารี

[4]

<u>รายละเอียดของเซ็นเซอร์ AMT 1001</u>

1. พลังงานต่ำ
2. ปริมาณขนาดเล็ก
3. การชดเชยอุณหภูมิ
4. ชิปเดียวมาตรฐานเป็นเส้นตรงเอาท์พุท
5. ความน่าเชื่อถือสูง
6. ใช้งานง่าย
7. ราคาไม่แพง

<u>คุณสมบัติของเซ็นเซอร์ AMT 1001</u>

1.การจ่ายแรงดัน (Vin): DC5V ± 5%

2.การบริโภคในปัจจุบัน: 2mA (สูงสุด 5mA)

3.ช่วงอุณหภูมิในการทำงาน: 0 ถึง 60 ° C

4.ช่วงความชื้น: 95% RH (ควบแน่น)

5.ความชื้นช่วงการวัด 20 ถึง 95% RH

6.อุณหภูมิการจัดเก็บ ช่วง: 0 ถึง 60 ° C

7.จัดเก็บช่วงความชื้น: 95% RH (ควบแน่น)

8.แม่นยำในการวัดความชื้น: ± 5% RH (เงื่อนไข: at25° C)

9.มาตรฐานแรงดัน output ความชื้น: (เงื่อนไข: at25 ° C , Vin = 5.0V)

10.ลักษณะการตรวจสอบอุณหภูมิ: ± 5% R (25 ° C) = 10Kω ± 5%

11.อ้างอิงอุณหภูมิ (11) (อ้างอิง): ± 5% RH (Vin = 5.00V DC, 40-80% RH, 25 ° C เป็นช่วงของการอ้างอิง , 0-50 ° C)

12.แรงดันพึ่งพา (12) (อ้างอิง): ± 5%

13.RH (at25 ° C 40-80% RHDC5V มาตรฐาน 4.75-5 .25 V ช่วง)

ตาราง แสดงความสัมพันธ์ระหว่างความชื้นสัมพัทธ์และแรงดันเอาท์พุต ดังตารางที่ 2.4.1

Relative humidity(%RH)	20	30	40	50	60	70	80
Output voltage(V)	0.6	0.9	1.2	1.5	1.8	2.1	2.4

ตารางที่ 2.4.1 แสดงความสัมพันธ์ระหว่างความชื้นสัมพัทธ์และแรงดันเอาท์พุต [5]

มหาวิทยาลัยเทคโนโลยีสุรนารี

2.4.2 เซ็นเซอร์ตรวจวัดความเข้มแสง (ZX-02F Light Sensor)

ในงานที่ต้องการวัดและเปรียบเทียบความเข้มแสงหรือตรวจจับการเปลี่ยนแปลงของปริมาณแสงนั้น จำเป็นต้องมีอุปกรณ์หรือเซ็นเซอร์ที่ใช้สำหรับวัตถุประสงค์ดังกล่าว อาจจะเลือกใช้ตัวต้านทานไวแสง (LDR: Light Dependent Resistor) ที่นำมาต่อกับตัวต้านทาน เพื่อใช้เป็นวงจร Voltage Divider แล้วใช้ ADC (Analog-To-Digital Converter) ภายนอกหรือภายในไมโครคอนโทรลเลอร์ (MCU) วัดระดับแรงดันของสัญญาณที่ได้ หรือจะเลือกใช้ไอซีสำหรับวัดความเข้มแสงโดยเฉพาะ ซึ่งจะได้เอาต์พุตแบบอะนาล็อกหรือดิจิทัลแล้วแต่อุปกรณ์

บทความนี้จะกล่าวถึง ZX-02F Light Sensor ตัวตรวจจับที่ใช้คือ ตัวต้านทานแปรค่าตามแสงหรือ LDR ซึ่งจะมีค่าความต้านทานลดลงเมื่อมีแสงตกกระทบ จัดวงจรเป็นแบบวงจรแบ่งแรงดัน ถ้า LDR ได้รับแสงน้อย ค่าความต้านทานจะสูง แรงดันตกคร่อมตัวมันจะมาก ดังนั้นแรงดันที่ (Vout) จะน้อย แต่เมื่อ LDR ได้รับแสงมาก ค่าความต้านทานจะน้อย ค่าแรงดันตกคร่อม LDR จะน้อยไปด้วย ทำให้แรงดันตกคร่อมขา (Vout) มีมาก ค่าที่ได้นี้สามารถนำไปป้อนให้กับอินพุต Sensor ของ I-BOX และอินพุตอะนาล็อกของ Stamp-BOX เพื่อนำค่าไปประมวลต่อไป แสดงในรูปที่ 2.7

ปที่ 2.7 ZX-02F Light Sensor เซ็นเซอร์ตรวจวัดความเข้มแสง [6]

2.5 โมดูลรีเลย์ 2 ช่อง (2 Channel Relay Module)

โมดูลรีเลย์ 2 ช่อง 5V (2 Channel Relay Module) เป็นโมดูลที่ใช้ควบคุมโหลดได้ทั้งแรงดันไฟฟ้า DC และ AC ซึ่งโหลดสูงสุด (Maximum Load) คือ AC 250V/10A, DC 30V/10A โดยใช้สัญญาณในการควบคุมการทำงานด้วยสัญญาณโลจิก TTL ทำงานด้วยสัญญาณแบบ Active Low, กระแสขับรีเลย์ (Drive Current) 15-20 mA., มีการออกแบบให้เป็น Isolate ด้วย Optocoupler, มี LED แสดงสถานะ Relay สามารถนำไปประยุกต์ใช้งาน PLC Control, บ้านอัจฉริยะ, ใช้ในโรงงานอุตสาหกรรม หรืองานอื่นๆ ขึ้นอยู่กับการเขียนโปรแกรมและการต่อใช้งานภายนอก สามารถ

เชื่อมต่อใช้งานกับบอร์ด Rasberry Pi, Arduino, ARM, MCS-51, AVR, PIC, 8051, DSP, MSP430, TTL logic แสดงในรูปที่ 2.8

<u>สเปคของโมดูลรีเลย์ 2 ช่อง</u>

1.ไฟเลี้ยงโมดูลรีเลย์ VCC = 5VDC.

2.ควบคุมโหลดได้ทั้งแรงดันไฟฟ้า AC ได้สูงสุด 250VAC 10A หรือ แรงดันไฟฟ้า DC ได้สูงสุด 30VDC 10A (Maximum Load)

3.ระดับสัญญาณอินพุทควบคุมแบบ TTL ทำงานด้วยสัญญาณแบบ Active Low

4.กระแสขับรีเลย์ (Drive Current) 15-20mA.

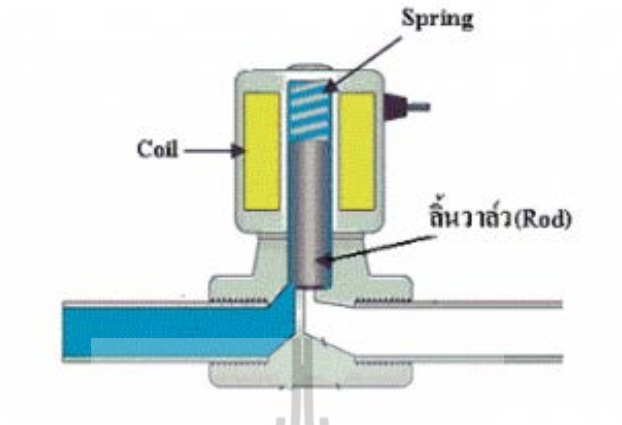
5.มีการออกแบบให้เป็น Isolate ด้วย Optocoupler

6.มี LED แสดงสถานะ Relay

7.โมดูลขนาด 3.85cm.(กว้าง) x 5.05cm.(ยาว) x 1.85cm.(สูง)

รูปที่ 2.8 โมดูลรีเลย์ 2 ช่อง [7]

2.6 โซลินอยด์วาล์ว (Solenoid Valve)



<u>หลักการทำงานของโซลินอยด์วาล์ว</u> (Solenoid valve)

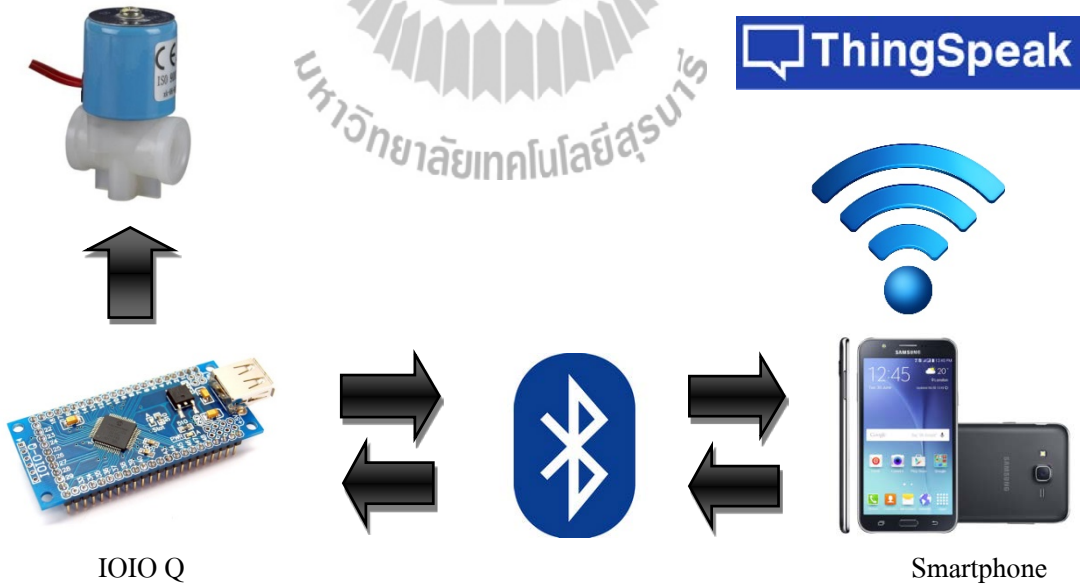
โซลินอยด์ (Solenoid) เป็นอุปกรณ์แม่เหล็กไฟฟ้าชนิดหนึ่ง ที่มีหลักการทำงานคล้ายกับรีเลย์ (Relay) ภายในโครงสร้างของโซลินอยด์จะประกอบด้วยขดลวดที่พันอยู่รอบแท่งเหล็กที่ภายในประกอบด้วยแม่เหล็กชุดบนกับชุดล่าง เมื่อมีกระแสไฟฟ้าไหลผ่านขดลวดที่พันรอบแท่งเหล็ก ทำให้แท่งเหล็กชุดล่างมีอำนาจแม่เหล็กดึงแท่งเหล็กชุดบนลงมาสัมผัสกันทำให้ครบวงจรทำงาน เมื่อวงจรถูกตัดกระแสไฟฟ้าทำให้แท่งเหล็กส่วนล่างหมดอำนาจแม่เหล็ก สปริงก็จะดันแท่งเหล็กส่วนบนกลับสู่ตำแหน่งปกติ จากหลักการดังกล่าวของโซลินอยด์ก็จะนำมาใช้ในการเลื่อนลิ้นวาล์วของระบบนิวแมติกส์ การปิด/เปิดการจ่ายน้ำหรือของเหลวอื่นๆ โครงสร้างของ Solenoid โดยทั่วไปแบ่งออกเป็น 2 ชนิดคือ เลื่อนวาล์วด้วยโซลินอยด์วาล์วกลับด้วยสปริง (Single Solenoid Valve) และเลื่อนวาล์วด้วยโซลินอยด์วาล์วกลับด้วยโซลินอยด์วาล์ว (Double Solenoid Valve) ในที่นี้ใช้แบบ เลื่อนวาล์วด้วยโซลินอยด์วาล์วกลับด้วยสปริง (Single Solenoid Valve) ดังรูปที่ 2.9

รูปที่ 2.9 โซลินอยด์วาล์ว [8]

บทที่ 3
การออกแบบโครงงาน

3.1 บทนำ

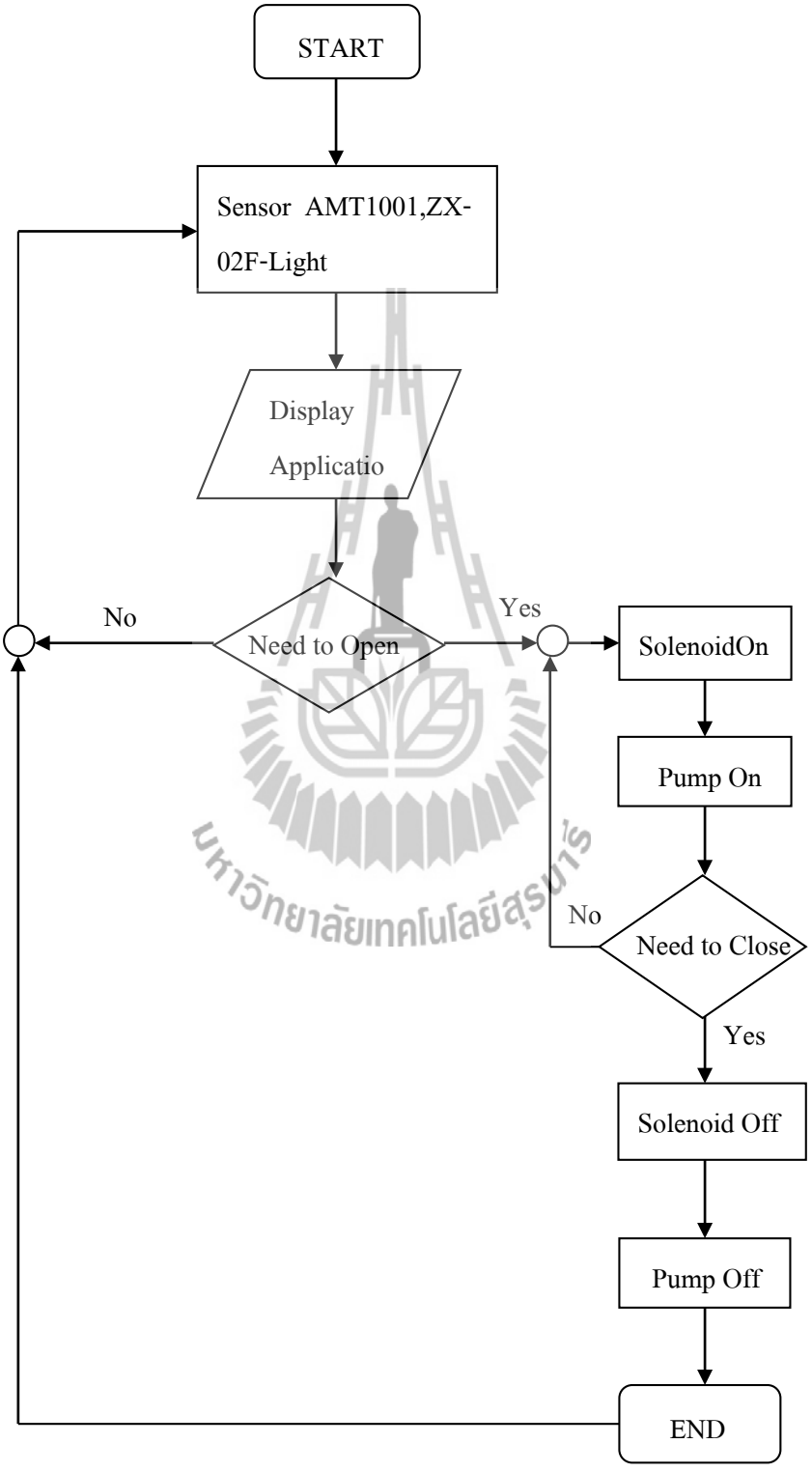
การทำงานเกี่ยวกับการควบคุมความชื้น อุณหภูมิและแสงในโรงเรือนกล้วยไม้มีความสำคัญเป็นอย่างมาก เพราะบางครั้งการให้น้ำกล้วยไม้โดยไม่ทราบค่าเหล่านั้น อาจจะมีผลต่อการเจริญเติบโตของต้นกล้วยไม้ได้ จากปัญหาข้างต้นทางคณะผู้จัดทำโครงงานจึงสร้างระบบเปิด – ปิดน้ำ เพื่อช่วยควบคุมความชื้น และตรวจสอบค่าสภาพอุณหภูมิและแสง โดยใช้บอร์ดโยโย่สร้างระบบนี้ขึ้นโดยผ่านสมาร์ทโฟนที่ทำงานบนระบบปฏิบัติการแอนดรอยด์ โดยติดตั้งแอปพลิเคชั่นบนสมาร์ทโฟนเพื่อควบคุมการเปิด/ปิดน้ำและแสดงผลค่าต่างๆที่ได้รับจากบอร์ดโยโย่แอปพลิเคชั่นใช้ช่องทางการสื่อสารผ่านทางบลูทูธ (Bluetooth) โดยเชื่อมต่อกับแลนไร้สาย (wireless Lan) เพื่อส่งข้อมูลผ่านแอปพลิเคชั่นไว้สำหรับเก็บค่าสภาพแวดล้อม เมื่อบอร์ดโยโย่ได้รับข้อมูลแล้วจะทำการเก็บข้อมูลไว้บน Thingspeak และเมื่อมีการสั่งคำสั่งเปิด – ปิดน้ำผ่านแอปพลิเคชั่น จะทำให้เกิดสัญญาณค่าลอจิกที่ขาดิจิตอล1 เพื่อควบคุมหน้าสัมผัสของรีเลย์ เมื่อหน้าสัมผัสของรีเลย์ปิดทำให้โซลินอยด์วาล์วทำงาน แสดงในรูปที่ 3.1



รูปที่ 3.1 รูปแบบการทำงานของชุดทดลอง

มหาวิทยาลัยเทคโนโลยีสุรนารี

แผนภาพการทำงานของเซ็นเซอร์ตรวจวัดอุณหภูมิและความชื้นในฟาร์มกล้วยไม้



มหาวิทยาลัยเทคโนโลยีสุรนารี

อุปกรณ์ที่ใช้ในโครงงาน มีดังนี้

1. โมดูล IOIO Q
2. โมดูล IOIO Activity Board
3. โมดูล IOIO Bluetooth Dongle
4. ZX-02F Light Sensor เซนเซอร์ตรวจวัดความเข้มแสง
5. AMT 1001 เซนเซอร์อุณหภูมิและความชื้น
6. โมดูล Breadboard MB-102
7. สายไฟ Jumper Wire 20 cm 40 pin Female-male
8. สายไฟ Jumper Wire 20 cm 40 pin Male-male
9. โมดูล Relay 2 channel
10. กล่องกันน้ำ
11.แบบจำลอง

3.2 การออกแบบฮาร์ดแวร์

การเชื่อมต่อ IOIO Board กับระบบ Android

ทำการ Download Library IOIO จาก https://github.com/ytai/ioio/wiki/Downloads

อุปกรณ์ที่ใช้ในการเชื่อมต่อ มีดังนี้

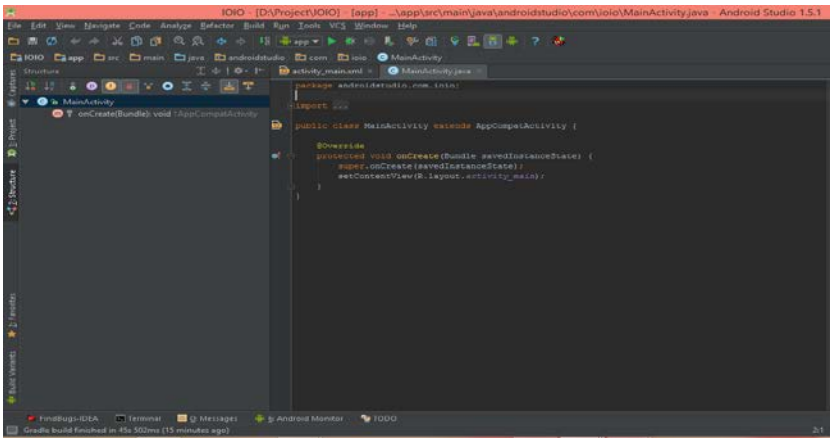
1.Bluetooth Dongle ทำการเสียบเข้า IOIO แล้วเปิด Bluetooth ใน Smart Phone เพื่อค้นหา IOIO จากนั้น ใส่ password 4545 เป็นการเชื่อมต่อ

2. สาย USB

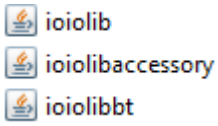
จากนั้นนำอุปการณ์ดังกล่าวมาเชื่อมกัน จำเป็นต้องติดตั้ง โปรแกรมดังนี้

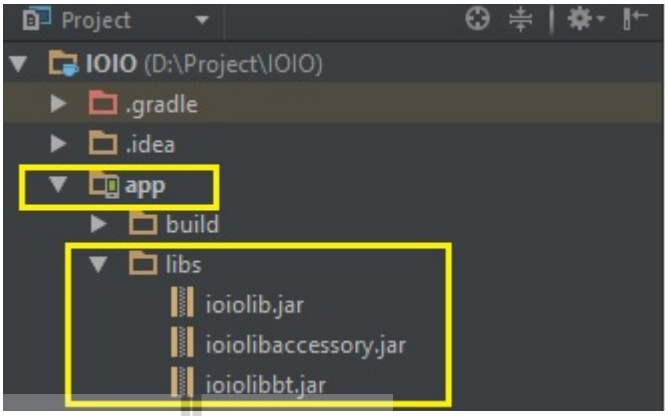
- Android studio 1.5.1
- Java SDK

จากนั้นทำการเปิด Android studio เปิด Project : File→New→New Project

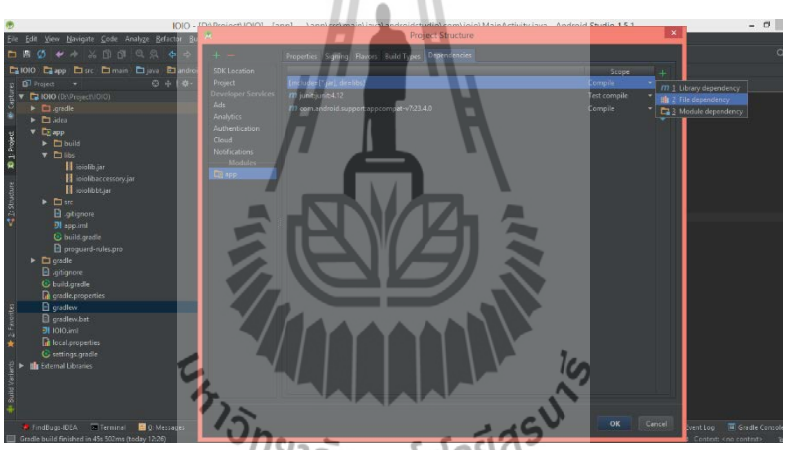


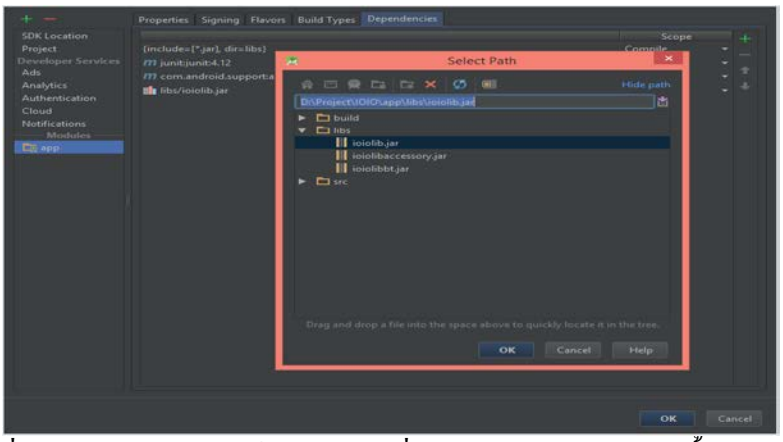
Copy Library IOIO แล้วทำการ Paste ลงบน Lib





File ⟶ Project Structure ⟶ app เลือก Dependencies เพิ่ม File Library IOIO ทีละไฟล์ จนครบ 3 ไฟล์ กด OK





เมื่อทำการเพิ่ม Library IOIO จะใช้การเขียนคำสั่งลงใน MainActivity ดังนี้

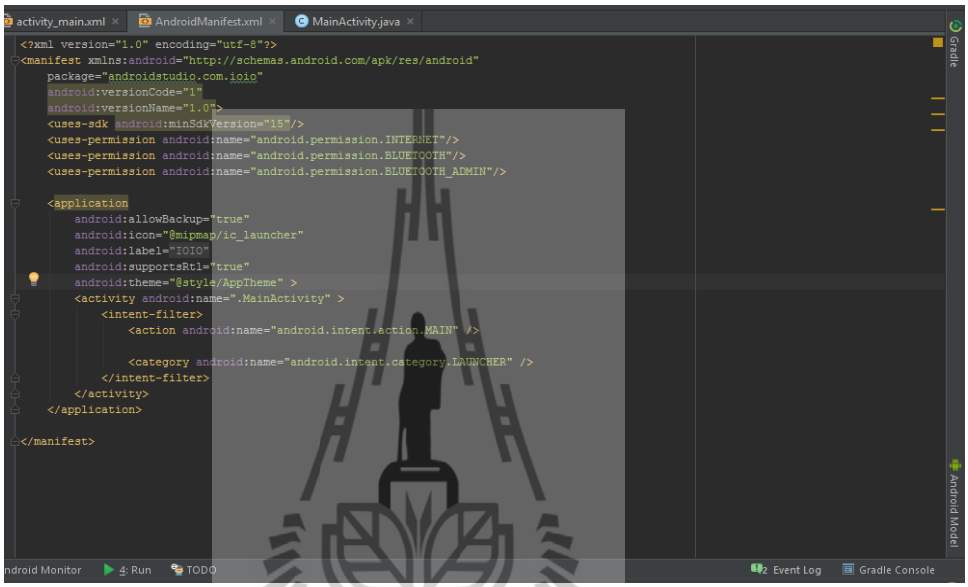
มหาวิทยาลัยเทคโนโลยีสุรนารี

```
package androidstudio.com.ioio;


import android.os.Bundle;
import android.widget.Toast;
import ioio.lib.api.exception.ConnectionLostException;
import ioio.lib.util.BaseIOIOLooper;
import ioio.lib.util.IOIOLooper;
import ioio.lib.util.android.IOIOActivity;


public class MainActivity  extends IOIOActivity {


   @Override
   public void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
   }
class Looper extends BaseIOIOLooper {
      protected void setup() throws ConnectionLostException
            , InterruptedException {


         runOnUiThread(new Runnable() {
            public void run() {
               Toast.makeText(getApplicationContext(),
                     "Connected!", Toast.LENGTH_SHORT).show();
            }
         });
      }
```

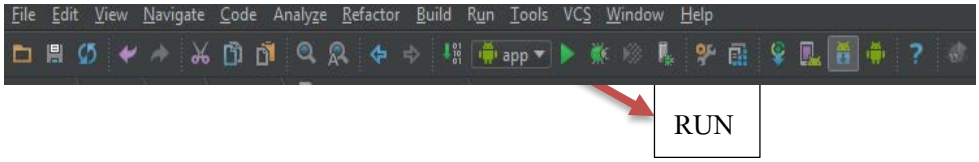
มหาวิทยาลัยเทคโนโลยีสุรนารี

บอร์ด IOIO นั้นจำเป็นจะต้องเข้าใช้งาน Internet และ Bluetooth ซึ่ง Bluetooth ใช้สำหรับเชื่อมต่ออุปกรณ์แอนดรอยด์กับบอร์ด IOIO ผ่าน Bluetooth และ Internet

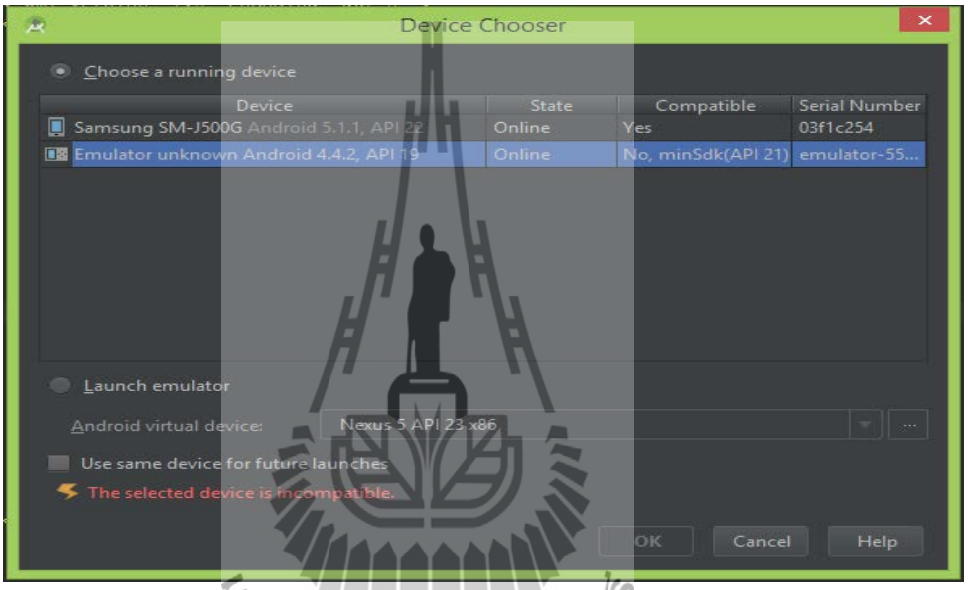
```
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.BLUETOOTH"/>
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
```



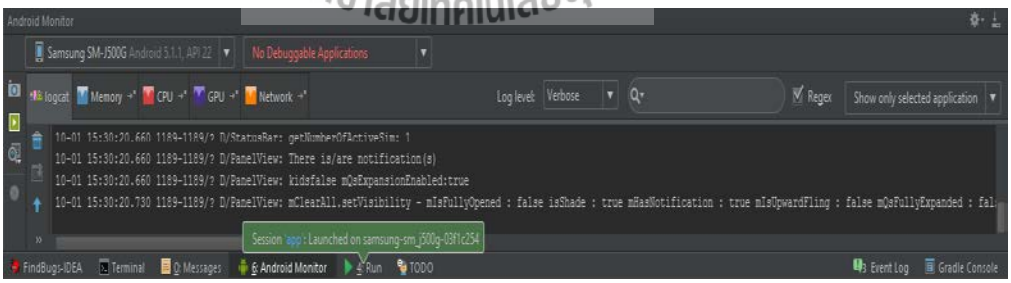
กดปุ่ม Run หรือ Shift + F10 เพื่อทำการติดตั้งแอปพลิเคชันลงบนเครื่อง ทำการเชื่อมสาย USB กับคอมพิวเตอร์และ Smart Phone โดยทำการเปิด Bluetooth เพื่อเชื่อมต่อ



ก่อนทำการติดตั้งจะมีหน้าต่างแสดงขึ้นมาถามว่าต้องการติดตั้งแอปพลิเคชันที่สร้างลงเครื่องไหน ให้เลือกเครื่องที่ต้องการติดตั้งแล้วกดปุ่ม OK



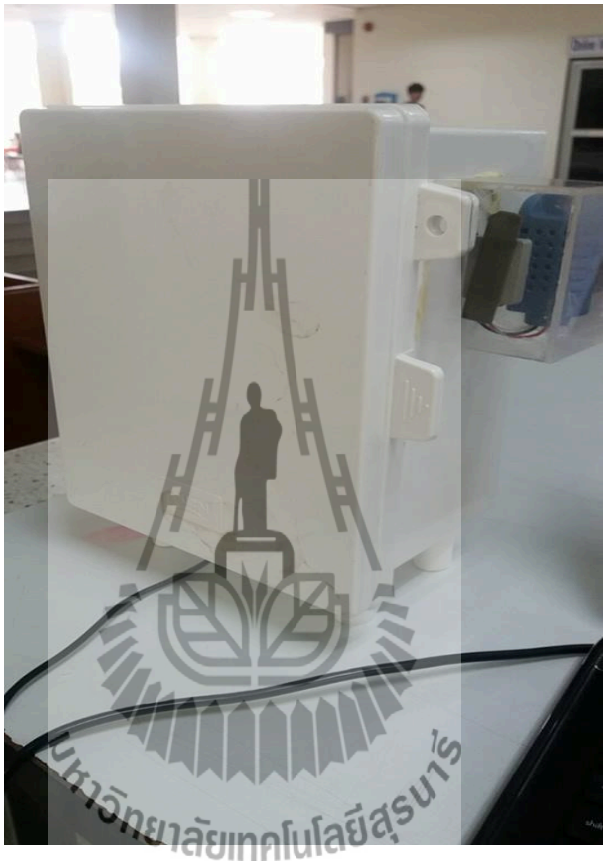
รอ Android Studio ติดตั้งลงบนเครื่องจนเสร็จ แอปพลิเคชันบนเครื่องก็จะเปิดขึ้นมา ดังรูปที่ 3.2



รูปที่ 3.2 หน้าแอปพลิชั่นแสดงการเชื่อมต่อบอร์ดโยโย่กับสมาร์ทโฟน

3.2.1 การออกแบบตู้ควบคุมขนาดเล็ก

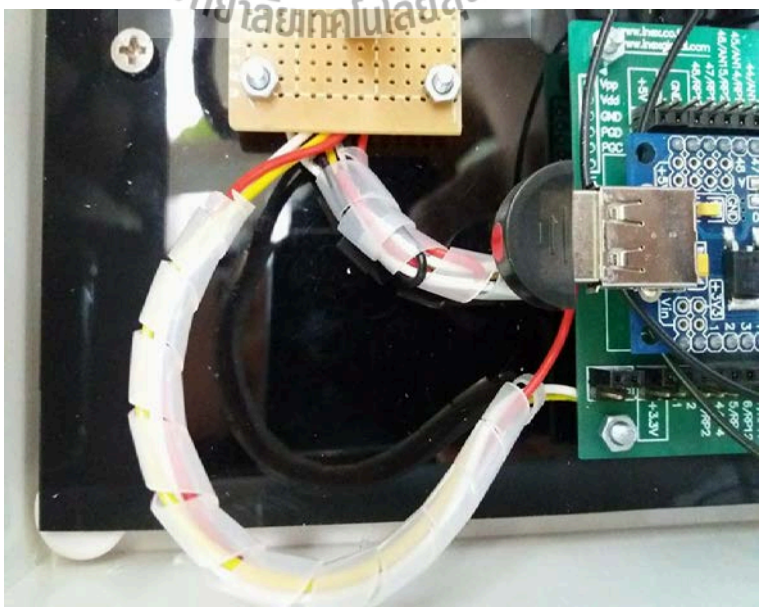
ส่วนควบคุม ประกอบเซนเซอร์อ่านค่าความชื้น อุณหภูมิและแสงรวมทั้งโมดูลรีเลย์เข้ากับบอร์ดโยโย่ จากนั้นจะแสดงผลที่แอปพลิเคชั่นโดยทำการติดตั้งลงบนสมาร์ทโฟน พร้อมทั้งสามารถสั่งงานระบบเปิด – ปิด ปั๊มน้ำตามคำสั่งโปรแกรม แสดงในรูปที่ 3.2.1 และรูปที่ 3.2.1-1

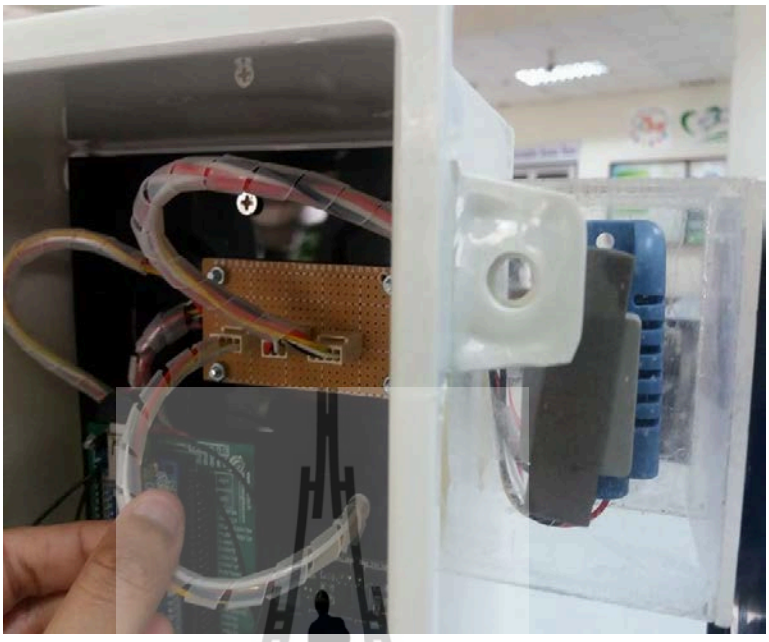


รูปที่ 3.2.1 กล่องควบคุมระบบขนาดเล็ก

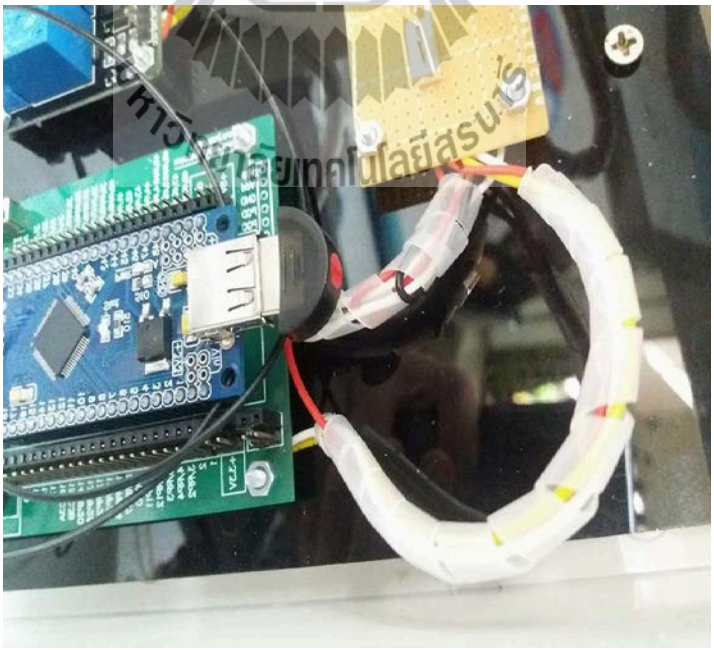
รูปที่ 3.2.1-1 อุปกรณ์การเชื่อมต่อภายในกล่องควบคุมขนาดเล็ก

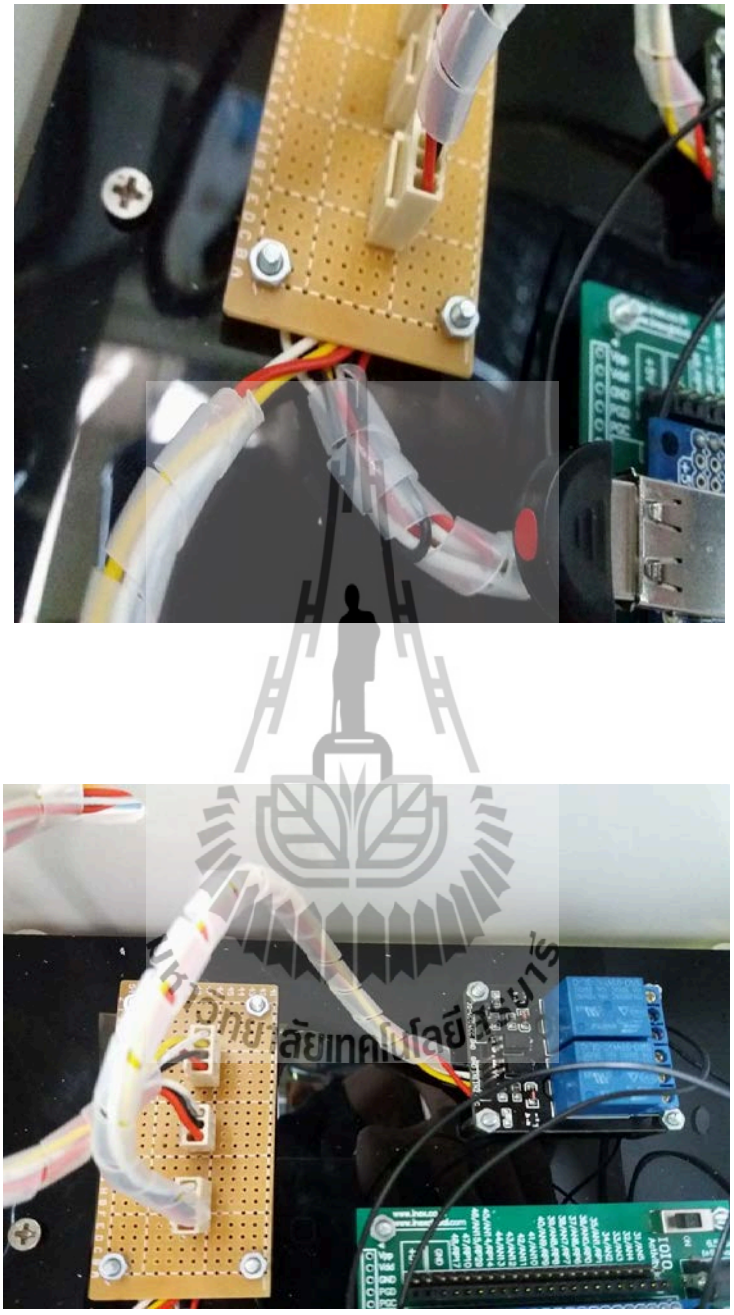
การเชื่อมต่อบอร์ดโยโย่กับบอร์ดรีเลย์ 2 ช่อง



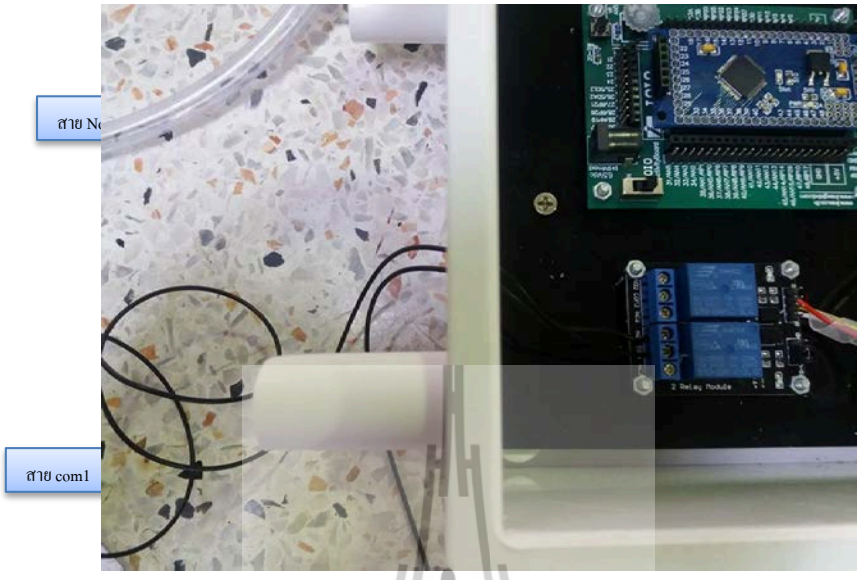


ต่อเซนเซอร์ AMT 1001 และ เซนเซอร์ ZX-02F Light Sensor กับบอร์ดโยโย





แสดงการต่อโมดูลนับเวลา (Relay) กับบอร์ดโย่โย



ทำการเชื่อสาย Com 1 ไปยังอะแดปเตอร์ และเชื่อมสาย No1 ยังขั้วลบของโซลินอยด์วาล์ว และปั๊มน้ำ แต่เนื่องจากทั้งสองสายไม่มีขั้วสามารถสลับสายในการเชื่อมต่อได้

การต่อโมดูลนับเวลา (Relay) กับโซลินอยด์วาล์วและปั๊มน้ำ เมื่อต่อโมดูลรีเลย์กับโซลินอยด์วาล์วได้แล้ว จะเหลือขั้วบวกของโซลินอยด์วาล์วกับปั๊มน้ำให้ทำการเชื่อมต่อกันเป็นสายเดียว แล้วต่อเข้ากับขั้วบวกของอะแดปเตอร์

.

3.2.2 การออกแบบตู้จำลอง

การออกแบบตู้จำลอง ใช้กล่องพลาสติกใส 2 กล่อง กล่องเล็กมีด้านกว้างยาว 14 เซ็นติเมตร และมีด้านยาวยาว 22 เซ็นติเมตร โดยภายในกล่องจะใช้บรรจุ โซลินอยด์วาล์ว และปั๊มน้ำขนาด 12 VDC ซึ่งต่อเข้ากับโซลินอยด์วาล์ว และทำการเจาะรูที่กล่องพลาสติกใส เพื่อใช้ต่อสายไฟเข้ากับโซลินอยด์วาล์ว และให้สายยางรอดผ่านรูเจาะไปยังอ่างใส่น้ำ และอีกด้านหนึ่งสายยางจะต่ออยู่กับหัวพ่นหมอกที่อยู่ในกล่องพลาสติกใสอีกกล่องหนึ่ง ซึ่งมีด้านกว้างยาว 26 เซ็นติเมตร และมีด้านยาวยาว 37 เซ็นติเมตร ซึ่งภายในกล่องนี้จะบรรจุต้นไม้ขนาดเล็กและหัวพ่นหมอก เพื่อแสดงให้เห็นว่าสามารถสั่งเปิด-ปิดน้ำผ่านแอปพลิเคชั่นได้จริง ดังรูปที่ 3.2.2

รูปที่ 3.2.2 การออกแบบตู้จำลอง

3.3 การออกแบบซอฟต์แวร์

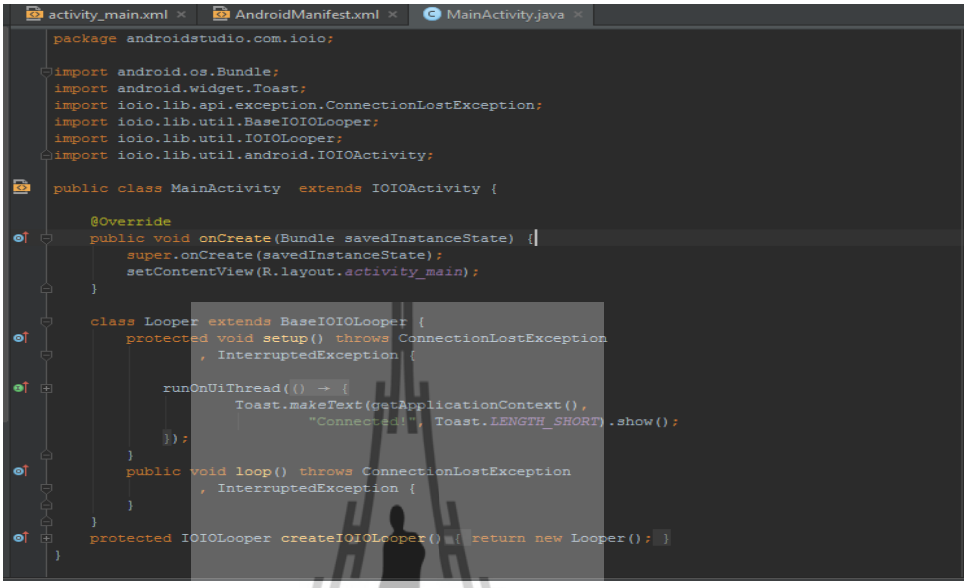
3.3.1 การออกแบบหน้าแอปพลิเคชั่น

หน้าแอปพลิเคชั่น จะประกอบด้วย

1. แถบค่าความชื้น
2. แถบอ่านค่าอุณหภูมิ
3. แถบอ่านค่าความเข้มแสง
4. ปุ่ม OFF/ON สำหรับเปิด – ปิดโซลินอยวาล์วและปั๊มน้ำ

5. ปุ่มแสดงตำแหน่งการติดตั้งเซ็นเซอร์

3.3.2 การรันโค้ดคำสั่งการอ่านค่า Sensor (ความชื้น อุณหภูมิและความเข้มแสง) ดังรูปที่ 3.3.2



รูปที่3.3.2หน้าต่างโปรแกรม Android studio สำหรับเขียนคำสั่งโค้ด Java

การเขียนโค้ดภาษา JAVA เพื่อควบคุมการอ่านค่าข้อมูล Sensor

```
import android.view.View;
import android.widget.TextView;
import ioio.lib.api.AnalogInput;
import ioio.lib.api.DigitalOutput;
```

การประกาศตัวแปร เพื่อทำการเก็บค่าลงตัวแปรที่กำหนด

```
TextView txtHum, txtTem, txtLight;
public double rldr=0;
```

การสร้างที่ตัวแปรเพื่อเก็บค่าที่อ่านได้

```
txtHum = (TextView) findViewById(R.id.txtHum);
txtTem = (TextView) findViewById(R.id.txtTem);
txtLight = (TextView) findViewById(R.id.txtLight);
```

โค้ดประกาศตัวแปร ในการใช้พอร์ต

```
AnalogInput ain, ain1;
AnalogInput ain2;
```

โค้ดกำหนดสัญญาณอินพุตและสัญญาณเอาต์พุตให้กับพอร์ต

```
ain=ioio_.openAnalogInput(32);  // hum
ain1=ioio_.openAnalogInput(33);  // tem
ain2 = ioio_.openAnalogInput(34); //ligh
```

โค้ดในการอ่านค่า จะได้หน้าแอปพลิเคชั่น ดังรูปที่ 3.2.2.1

```
txtVoltage.setText("Voltage" + String.format("%.2f",ain.getVoltage())+"V");
txtHum.setText("Humidity : " + String.format("%.2f", ain.getVoltage() * 100 / 3) + " %");
txtTem.setText("Temperature : " + String.format("%.2f", ain1.getVoltage() * 80 / 0.8) + " C");
rldr = (10 * (3.3 - ain2.getVoltage())) / ain2.getVoltage();
txtLight.setText("Light : " + String.format("%.2f", 500 / rldr) + "LUX");
```

รูปที่ 3.2.2.1 การแสดงค่าความชื้น อุณหภูมิและความเข้มแสง

3.3.3 การเขียนโค้ดคำสั่งการเปิด - ปิด Relay

โค้ดจาวาที่ใช้ในการควบคุม

```
myToggleButton1 = (ToggleButton) findViewById(R.id.toggleButton);
```

โค้ดประกาศตัวแปรในการเก็บค่าตัวแปร

```
private ToggleButton myToggleButton1;
```

โค้ดในการประกาศการใช้พอร์ต

```
private DigitalOutput myOutput1;
```

โค้ดแสดงแสดงการใช้งานพอร์ต

```
myOutput1= ioio_.openDigitalOutput(1,true);
```

โค้ดแสดงการควบคุมการเปิด – ปิด รีเลย์

```
myOutput1.write(!myToggleButton1.isChecked());
```

รูปที่3.3.3. แสดงการสั่งเปิด-ปิด รีเลย์บอร์ด 2 ช่อง

3.3.4 การเขียนคำสั่งการส่งค่าข้อมูลขึ้น Thingspeak

```
///send to thingspeak
                String res ="";
 String[] params =
{"api_key","NWFB4VLBR4ON5IY6","field1",Double.toString(ain1.getVoltage() * 80 /
0.8),"field2",Double.toString(ain.getVoltage() * 100 / 3),"field3",Double.toString(500 /
rldr)};
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

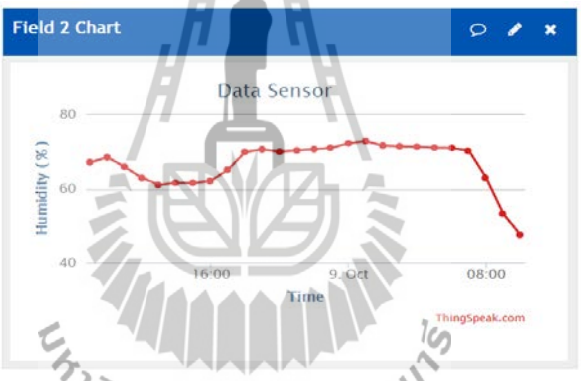
บทที่ 4
ผลการทดลองและการวิเคราะห์ผลการทดลอง

4.1 วิธีการทดสอบ

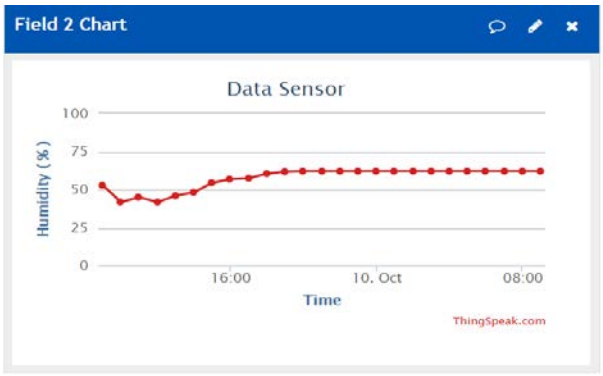
1. ทำการติดตั้งชุดอุปกรณ์ให้อยู่ในสถานะพร้อมใช้งาน
2. ทำการเก็บค่าอุณหภูมิ ความชื้น และความเข้มแสง
3. ทำการเชื่อมอินเตอร์เน็ต เพื่ออัพข้อมูลขึ้น Thingspeak

4.1.1 การทดลองที่ 1

ตารางที่ 4.1.1 การเก็บค่าข้อมูล ความชื้น อุณหภูมิและแสง โดยส่งข้อมูลเซนเซอร์ขึ้น Thingspeak (วันที่ 8 ตุลาคม 2559– 10 ตุลาคม 2559)

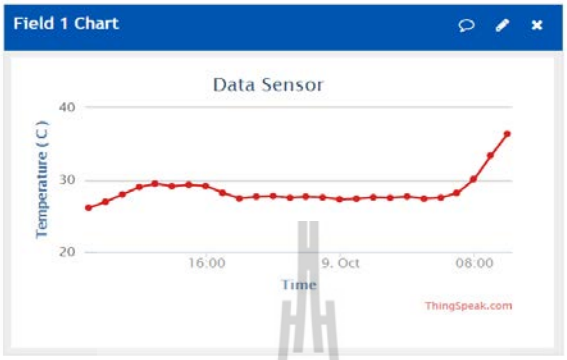


จากกราฟความชื้นนี้จะพบว่าวันที่ 1 ช่วงเวลา 09.00 -10.00 จะมีความชื้นสูงเกิน 60 เปอร์เซ็นต์และจะค่อยๆลดลงจนมีความชื้นต่ำสุด ช่วงเวลา 12.00 ที่ 60 เปอร์เซ็นต์ และเพิ่มขึ้นเกิน 60 เปอร์เซ็นต์อีกครั้งในช่วง 16.00 เป็นต้นไป จนถึงช่วง 08.00 ของวันที่ 2 (9 ตุลาคม 2559)

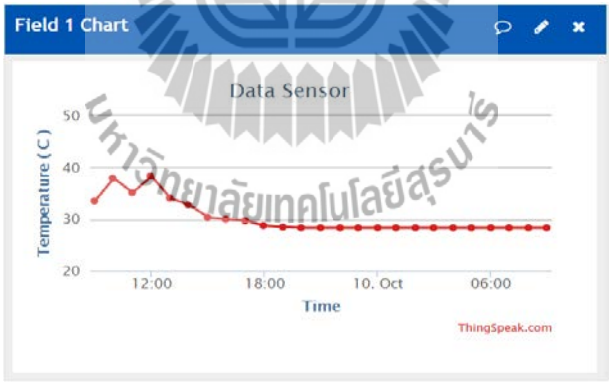


ช่วงกราฟแสดงความชื้นวันที่ 2 จะพบว่าช่วง 09.00- 12.00 ค่าความชื้นจะลดต่ำลงสุดในช่วง 12.00 ที่ ความชื้น 40 เปอร์เซ็นต์ และค่าความชื้นจะค่อยๆเพิ่มในช่วงเวลา 13.00 น และจะ

เริ่มคงที่ที่ความชื้นเกิน 60 เปอร์เซ็นต์ในเวลา 20.00 เป็นต้นไป เนื่องจากมีสภาพอากาศมีฝนตกในช่วงบ่าย จึงทำให้ช่วงเย็น กลางคืนทั้งคืนและเวลาเช้า ของสองวันนี้มีความชื้นเกิน 60 เปอร์เซ็นต์ขึ้นไป



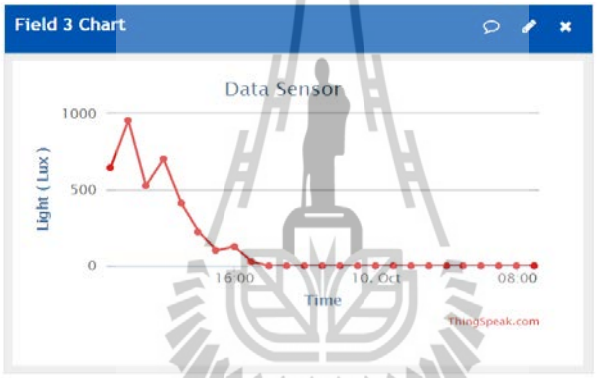
จากกราฟแสดงอุณหภูมิของวันที่ 1 พบว่า ในช่วงเวลา 09.00 จะเริ่มมีอุณหภูมิสูงขึ้นไม่เกิน 30 องศาเซลเซียส และเริ่มคงที่ในช่วงเวลาตั้งแต่12.00 – 16.00 และจะเริ่มลดลงต่ำกว่า 30 องศาเซลเซียสในช่วง 17.00 เป็นต้นไปหลังจากนั้นจะเริ่มคงที่ และเพิ่มขึ้นอีกครั้งในช่วง 07.00 ของวันที่ 2 (9 ตุลาคม 2559)



จากกราฟของวันที่ 2 ช่วงเวลา 09.00- 14.00 อุณหภูมิจะสูงเกิน 30 องศาเซลเซียส ในช่วง 15.00 – 17.00 อุณหภูมิจะคงที่ระยะหนึ่งที่ 30 องศาเซลเซียส _และลดต่ำลงน้อยกว่า 30 องศาเซลเซียสอยู่ในช่วง18.00 เป็นต้นไป

จะเห็นว่าทั้งสองกราฟจะมีช่วงอุณหภูมิที่สอดคล้องกัน เนื่องจากมีสภาพอากาศฝนตกในช่วงบ่ายและเย็นเกิดขึ้นทั้งสองวันที่ทำการเก็บค่า ส่งผลให้ได้กราฟดังสองรูปนี้ขึ้นมา

กราฟแสดงค่าความแข้มของแสง จะพบว่า ช่วง 09.00 จะมีความเข้มแสงเพิ่มขึ้นไม่เกิน500 ลักซ์ จนถึงเวลา 12.00 ก็จะตกลงมาเรื่อยๆและเริ่มคงที่ในช่วง 19.00 – 07.00 ซึ่งเป็นช่วงต่ำสุดของกราฟเนื่องจากเป็นเวลากลางคืน และจะเพิ่มขึ้นในช่วง 08.00 ของวันที่ 2 (9 ตุลาคม 2559)



จากกราฟแสดงความเข้มของแสงวันที่ 2 จะพบว่า ช่วง 09.00 – 12.00 จะมีความเข้มของแสงค่อนข้างสูงเกิน 500 ลักซ์ เนื่องจากสภาพอากาศช่วงนั้นค่อนข้างแจ่มใส ไม่มีเมฆ และแสงแดดแรง ส่งผลให้ค่าความเข้มของแสงสูงตามมา ช่วง 13.00 ความเข้มแสงจะค่อยๆลดลงเรื่อยจนถึงช่วง 19.00 น และเริ่มคงที่ในจุดที่แสงต่ำสุดของกราฟที่ประมาณ 0 ลักซ์ เป็นต้นไป

รูปที่ 4.1 ตู้ออกแบบควบคุมขนาดเล็กขณะทำการเก็บค่า

4.1.2 การทดลองที่ 2

<u>ตารางที่ 4.1.2</u> การทดลองใช้คำสั่ง ON/OFF ผ่านแอปพลิเคชั่นเพื่อเปิด -ปิด โซลินอยด์วาล์วและปั๊มน้ำจำลองผ่านทาง Bluetooth

ผลการแสดงเปิด – ปิด ตามคำสั่ง บนแอปพลิเคชั่นมือถือแอนดรอยด์

คำสั้ง	ผลลัพธ์
ON	โซลินอยด์และปั๊มน้ำทำงาน
OFF	โซลินอยด์และปั๊มน้ำหยุดทำงาน

จากการทดลองการเปิด–ปิดคำสั่งคำสั่ง ON/OFF ผ่านแอปพลิเคชั่น จะพบว่าสามารถสั่งการเปิด–ปิด โซลินอยด์วาล์วและปั๊มน้ำได้ตามคำสั่ง

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 5
สรุปผลโครงงานและข้อเสนอแนะ

5.1 สรุปผลโครงงาน

อุปกรณ์รดน้ำกล้วยไม้ควบคุมด้วยสมาร์ทโฟนด้วยการสื่อสารผ่านบลูทูธ สามารถส่งคำสั่งเปิด/ปิดน้ำโดยการส่งคำสั่งผ่านสมาร์ทโฟน สามารถส่งข้อมูลสถานะต่างๆของอุปกรณ์มาแสดงที่หน้าโปรแกรมประยุกต์ได้สามารถเปิดรดน้ำกล้วยไม้และสามารถเก็บข้อมูลสถิติ โดยผ่าน Internet โดยใช้ Thingspeak แสดงกราฟได้

5.2 ปัญหาที่พบ

1.เมื่อมีการสั่งค่าเซ็นเซอร์ขึ้น internet บอร์ดจะเกิดการทำงานช้า ทำให้หน้าแอฟพลิเคชั่นเกิดการทำงานช้า

5.3 ข้อเสนอแนะ

1.ให้ตัดการเชื่อมอินเตอร์เน็ตในกรณีต้องการแค่ดูสภาพแวดล้อมภายในโรงเรือน

มหาวิทยาลัยเทคโนโลยีสุรนารี

เอกสารอ้างอิง

[1] ระบบสมองกลฝังตัว (Embeded System) สืบค้นจาก
http://www.smeclabs.com/img/embedded/2.gif

[2] Crashes by OS Version Normalized (12/1 – 12/15) สืบค้นจาก
https://panomkorn.wordpress.com/2013/07/26

[3] IOIO Board สืบค้นจาก
http://inex.co.th/shop/media/catalog/product/cache/1/image/9df78eab33525d08d6e5fb8d27136e95/i/o/ioio_q.jpg

[4] เซ็นเซอร์ AMT 1001 สืบค้นจาก
http://cu.lnwfile.com/_/cu/_raw/2p/zb/qc.jpg

[5] ตารางแสดงความสัมพันธ์ระหว่างความชื้นสัมพัทธ์และแรงดันเอาท์พุต สืบค้นจาก
http://www.myarduino.net/product/382/amt1001-temperature-and-humidity-sensors-analog-voltage-output

[6] ZX-02F Light Sensor เซ็นเซอร์ตรวจวัดความเข้มแสง สืบค้นจาก
http://inex.co.th/shop/media/catalog/product/cache/1/small_image/9df78eab33525d08d6e5fb8d27136e95/z/x/zx_02.jpg

[7] โมดูลรีเลย์ 2 ช่อง สืบค้นจาก
https://ae01.alicdn.com/kf/HTB1GPthJXXXXXazXFXXq6xXFXXX0/High-Quality-New-2-Channel-5V-font-b-Relay-b-font-interface-board-controlled-directly-by.jpg

[8] โซลินอยด์วาล์ว สืบค้นจาก
http://h.lnwfile.com/_/h/_raw/pz/6a/ls.jpg

มหาวิทยาลัยเทคโนโลยีสุรนารี

มหาวิทยาลัยเทคโนโลยีสุรนารี