โครงงาน

ระบบตรวจสอบสถานะภายในบ้าน (Smart Home)

นายฐากร	สุขะสิษฐ์	B5926902
นายปิยะราช	วงชะศรี	B6028828
นางสาวอารียา	ซื่อตรง	B5928848

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์ สำนักวิชาวิศวกรรมมศาสตร์

มหาวิทยาลัยเทคโนโลยีสุรนารี

ปีการศึกษา 2561

หัวข้อโครงงาน : ระบบตรวจสอบสถานะภายในบ้าน (Smart Home)

ผู้เสนอโครงงาน : นางสาวอารียา ซื่อตรง รหัสนักศึกษา B6028848
นายปิยะราช วงชะศรี รหัสนักศึกษา B6020028
นายฐากร สุขะสิษฐ์ รหัสนักศึกษา B5826902

อาจารย์ที่ปรึกษา : อ.ดร.ธีทัต ดลวิชัย

ปีการศึกษา : 2561

บทคัดย่อ

โครงงานนี้จัดทำขึ้นเพื่อศึกษาและออกแบบในสร้างระบบตรวจสอบสถานะภายในบ้าน เพื่อใช้ในการตรวจเช็คและใช้ตรวจสอบสถานะการทำงานของระบบต่าง ๆ ที่สามารถทำงานปกติและไม่ทำงานหรือระบบขัดข้อง เพื่อลดปัญหาที่ตามมาและลดอัตรายจากการทำงานผิดพลาดของระบบที่เสี่ยงต่อภัยต่าง ๆ ภายในบ้าน คณะผู้จัดทำได้ศึกษาหาข้อมูลและสร้างสร้างระบบตรวจสอบสถานะภายในบ้าน ซึ่งสามารถใช้ตรวจสอบสถานะการทำงานของระบบภายในบ้านได้โดยการตรวจเช็คผ่านระบบเครือข่ายและใช้แอพพลิเคชั่นแสดงสถานะการทำงาน ส่วนประกอบของวงจรและอุปกรณ์อิเล็กทรอนิกส์ที่ใช้ในการทำระบบ เริ่มจากการการเขียนโปรแกรมด้านระบบการควบคุมผ่านแอพพลิเคชั่น โดยประยุกต์ใช้โปรแกรม Arduino ร่วมกับบอร์ด NodeMCU ESP8266 (ESP-12E) ให้สามารถทำงานผ่านระบบเครือข่ายได้โดยใช้ Relay Module ควบคุมการแสดงสถานะทำงานของระบบซึ่งในการออกแบบระบบเป็นการจำลองการตรวจสอบการทำงานและแสดงสถานะทำงานของระบบต่าง ๆ ภายในบ้าน ลักษณะการทำงานของระบบคือ ทำการติดต่อสื่อสารกับบอร์ด NodeMCU ESP8266 (ESP-12E) ผ่านระบบเครือข่ายและใช้แอพพลิเคชั่นควบคุมการทำงานของระบบไมโครคอนโทรลเลอร์และวงจรอิเล็กทรอนิกส์ที่เกี่ยวข้อง โดยเป้าหมายที่ตั้งไว้คือ ระบบตรวจสอบสถานะภายในบ้านสามารถตรวจสอบและแสดงสถานะการทำงานของระบบต่าง ๆ ได้ถูกต้องแม่นยำ

การศึกษาและออกแบบระบบตรวจสอบสถานะระบบต่าง ๆ ภายในบ้านสามารถตรวจสอบสถานะการทำงานของระบบต่าง ๆ ภายในบ้านได้ถูกต้องแม่นยำและมีประสิทธิภาพ

สำนักวิทยบริการและเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีสุรนารี

กิตติกรรมประกาศ

โครงงานนี้สำเร็จลุล่วงตามวัตถุประสงค์ทุกประการคณะผู้จัดทำโครงงานขอขอบพระคุณบุคคลต่าง ๆ ที่ให้คำปรึกษา แนะนำ และช่วยเหลืออย่างดียิ่ง ทั้งด้านวิชาการ และด้านการดำเนินโครงงานดังนี้อาจารย์ ดร.ธีทัต ดลวิชัย อาจารย์ที่ปรึกษาโครงงานที่ให้ความรู้และคำปรึกษาในการทำโครงงานรวมทั้งให้คำแนะนำในการดำเนินชีวิตด้วยความเมตตาเสมอ ตลอดจนเป็นแบบอย่างที่ดีแก่ผู้จัดทำโครงงาน ในด้านการดำเนินชีวิต และการอุทิศตนต่อคนรุ่นหลังโดยไม่เห็นแก่ความเหน็ดเหนื่อย ซึ่งผู้จัดทำโครงงานจะจดจำช่วงเวลาดังกล่าวไว้เป็นคติสอนใจที่ดีงามตลอดไปคณาจารย์ทุกท่านที่ประสิทธิ์ประสาทวิชาความรู้ให้แก่ผู้จัดทำโครงงานด้วยความเมตตากรุณาทั้งใน อดีตจนถึงปัจจุบัน วิศวกรและเจ้าหน้าที่ประจำศูนย์เครื่องมือวิทยาศาสตร์และเทคโนโลยีทุกท่านที่อำนวยความ สะดวกในการใช้เครื่องมือและอุปกรณ์ต่าง ๆ ตลอดจนให้คำปรึกษาที่เป็นประโยชน์ต่อโครงงานท้ายที่สุดขอกราบขอบพระคุณ บิดา มารดา รวมถึงครอบครัวที่ให้การอบรมเลี้ยงดูและให้โอกาสทางด้านการศึกษาเป็นอย่างดีมาโดตลอดจนทำ ให้ผู้จัดทำ โครงงานประสบความสำเร็จในชีวิตเรื่อยมา

คณะผู้จัดทำ

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ

หน้า

เกี่ยวกับโครงงาน .....ก

กิตติกรรมประกาศ .....ข

บทคัดย่อ.....ค

สารบัญ.....ง

สารบัญรูป.....ฉ

สารบัญาราง.....ช

บทที่

1 **บทนำ**.....1

1.1 ความเป็นมาและความสำคัญของปัญหา.....1

1.2 วัตถุประสงค์ของโครงงาน.....2

1.3 ขอบเขตของโครงงาน.....2

1.4 ประโยชน์ที่คาดว่าจะได้รับ.....2

1.5 คำนิยามศัพท์.....2

2 **เอกสารและโครงงานที่เกี่ยวข้อง**.....3

2.1 Arduino.....5

2.2 โพรโทบอร์ด.....5

2.3 NodeMCU ESP8266.....5

2.4 รีเลย์**(Relay)** 5

2.5 มอเตอร์.....5

2.6 งานโครงงานเกี่ยวข้อง.....5

3 วิธีการจัดทำโครงงาน.....19

3.1วัสดุและอุปกรณ์.....19

สาขาวิชาวิศวกรรมแมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ (ต่อ)

หน้า

3.2 วิธีจัดทำโครงงาน……..……....20

4 ผลการศึกษา……..……...……29

4.1 ผลการทดลอง……...…29

4.2 ลักษณะการทำงานของชุดควบคุม……..……...……...30

5 สรุปผลและข้อเสนอแนะ…… ……..…….31

5.1 สรุปผลการดำเนินโครงงาน……...…….…….……...…….31

5.2ข้อเสนอแนะในการดำเนินโครงงาน……...……..…..…31

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูป

รูปที่ **หน้า**

2.1 บอร์ด Arduino ต่อกับ LED……8

2.2 บอร์ด Arduino ต่อกับบอร์ด XBee Shield……9

2.3 รูปแบบการเขียนโปรแกรมบน Arduino……10

2.4 เลือกรุ่นบอร์ด Arduino ที่ต้องการ upload ……21

2.5 เลือกหมายเลข Comport ของบอร์ด……21

2.6 กดปุ่ม Verify เพื่อตรวจสอบความถูกต้อง……22

2.7 Ultrasonic Sensor และSensor บอกระยะ……22

2.8 ส่วนประกอบของ Layout & Pin out Arduino Board ……23

2.9 แผ่นโพรโทบอร์ด……23

2.10 อักษรกำกับประจำหลัก……24

2.11 การต่อวงจรวงจรภายในแผ่นโพรโทบอร์ด……24

2.12 การต่อแหล่งจ่ายไฟกับแผ่นโพรโทบอร์ด……27

2.13 การต่อแหล่งจ่ายฟกับแผ่นโพรโทบอร์ดทั้ง 2 ด้าน……28

2.14 การต่ออุปกรณ์กับแผ่นโพรโทบอร์ด 1……29

2.15 การต่ออุปกรณ์กับแผ่นโพรโทบอร์ด ……29

2.16 บอร์ด NodeMCU ESP8266 (ESP-12E)……29

2.17 รีเลย์กำลัง……29

2.18 เซอร์โวมอเตอร์ MG90S-180……29

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูป(ต่อ)

รูปที่ **หน้า**

3.1 ออกแบบโครงสร้างชิ้นงานด้วยมือ(แบบร่าง)……..….19

3.2 ออกแบบโครงสร้างชิ้นงานด้วยคอมพิวเตอร์……20

3.3 ออกแบบโครงสร้างชิ้นงานด้วยคอมพิวเตอร์……..20

3.4 ออกแบบวงจรของระบบด้วยคอมพิวเตอร์……..……..……21

3.5 ออกแบบการเชื่อมต่อระบบด้วยคอมพิวเตอร์……..……...…..21

3.6 บอร์ด NodeMCU ESP8266 (ESP-12E)……..……..……...…..22

3.7 สายไฟ……..……..……..22

3.8 มอเตอร์ Servo mg90s……..……...23

3.9 พัดลม DC 5V ……..……23

3.10 8-Channel-5V Relay-Module……..…..…….24

3.11 แสดงการต่ออุปกรณ์เข้ากับบอร์ด NodeMCU ESP8266 (ESP-12E)……..……24

3.12 แสดงการตรวจสอบสถานนการทำงานของวงจร……..……..25

3.13 แสดงการต่ออุปกรณ์ลงกล่อง……25

3.14 การต่ออุปกรณ์กับแผ่นโพรโทบอร์ด 1……...……..…26

3.15 การต่ออุปกรณ์กับแผ่นโพรโทบอร์ด ……..…26

3.16 ผังการทำงานระบบตรวจสอบสถานะภายในบ้าน……...……28

4.1 แสดงผลสถานะการทำงานของระบบ คือ ปิดการทำงานทุกระบบ…….……..….29

4.2 แสดงสถานะเปิดการทำงาน 1 ระบบ ……..…..…30

4.3 แสดงลักษณะชุดควบคุมระบบตรวจสอบสถานะภายในบ้าน……...……...…..30

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญตาราง

ตารางที่ **หน้า**

3.1 วัสดุและอุปกรณ์ที่ใช้ในการจัดทำโครงงาน.....18

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในยุคปัจจุบันเทคโนโลยีจะทำให้อุปกรณ์ต่าง ๆ สื่อสารและทำงานกันเองได้อย่างอัตโนมัติ เทคโนโลยีที่ช่วยเหลือ อำนวยความสะดวก หยิบจับ คำนวณ ประมวลผล เทคโนโลยีถูกนำมาพัฒนาต่อยอดเพื่อลดบทบาทของมนุษย์ และเพิ่มศักยภาพของมนุษย์ในการใช้ความคิดเพื่อข้ามขีดจำกัดสร้างสรรค์พัฒนาสิ่งใหม่ ๆ เช่น เราสามารถเปิด-ปิด หรือสั่งงานอื่น ๆ กับเครื่องใช้ไฟฟ้าในบ้านตัวเองผ่านแอพลิเคชั่นโดยไม่ต้องเดินไปกดสวิตช์ หรือตัวอย่างที่ถูกนำมาใช้งานจริงเทคโนโลยีซิมูเลชั่น Simulation จำลองสถานการณ์เพื่อฝึก วางแผนสถานการณ์โดยที่ไม่ต้องอุปกรณ์จริง หรือเป็นทดสอบและทดลองวงจรไฟฟ้าและอิเล็กทรอนิกส์และใช้เป็นสื่อการเรียนรู้แบบ Interactive เป็นต้น การทำงานของอุปกรณ์บางอย่างมีขีดจำกัดในการใช้งานทำให้เกิดปัญหาต่าง ๆ ตามมาอีกหลายอย่าง

จากข้อมูลที่ได้ศึกษาและปัญหาที่พบในการใช้ระบบต่าง ๆ ภายในบ้านคณะผู้จัดทำจึงมีแนวคิดที่จะแก้ไขปัญหาที่เกิดขึ้น โดยให้ความสำคัญในเรื่องการตรวจสอบการทำงานของระบบต่าง ๆ ในบ้าน เช่น ระบบไฟฟ้า ระบบกล้องวงจรปิด ระบบน้ำ เป็นต้น ซึ่งเป็นระบบที่สำคัญจึงได้ศึกษาและหาข้อมูลเพื่อสร้างระบบตรวจสอบสถานะภายในบ้านโดยการควบคุมแอพพลิเคชั่นผ่านระบบเครือข่ายเพื่อแก้ไขปัญหาการทำงานขอระบบต่าง ๆ ภายในบ้านให้สามารถทำงานได้อย่างมีประสิทธิภาพและสามารถนำมาประยุกต์ใช้ในด้านอื่น ๆ ต่อไป

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อสร้างระบบตรวจสอบสถานะภายในบ้าน

1.2.2 เพื่อศึกษาแอพพลิเคชั่น Blynk ควบคุมการทำงานผ่านระบบเครือข่าย WiFi

1.2.3 เพื่อประยุกต์ใช้งานโปรแกรม Arduino และบอร์ด NodeMCU ESP8266 (ESP-12E)

1.2.4 เพื่อพัฒนาผลงานที่ประดิษฐ์ขึ้นให้เข้าสู่ความเป็นมาตรฐาน สามารถนำไปใช้งานได้อย่างมีคุณภาพ

1.3 ขอบเขตของโครงงาน

1.3.1 สามารถตรวจสอบสถานะการทำงานของระบบต่าง ๆ ภายในบ้านโดยการเปิดหรือปิดระบบต่าง ๆ ด้วยแอพพลิเคชั่นตามที่กำหนดไว้ได้

1.3.2 สามารถสั่งเปิดหรือปิดอุปกรณ์ต่าง ๆ ที่กำหนดไว้ได้

1.3.3 สามารถควบคุมการเปิดหรือปิดผ่านระบบ WiFi ได้

1.3.4 สามารถควบคุมการทำงานของระบบด้วยมือถือผ่านแอพพลิเคชั่น Blynk

1.3.5 สามารถแสดงสถานะการทำงานด้วยแอพพลิเคชั่น Blynk บนมือถือ

1.3.6 ใช้โปรแกรม Arduino ควบคุมการทำงานของ NodeMCU ESP8266 (ESP-12E)

1.3.7 ใช้ 8 Channel Relay 5V

1.3.8 ใช้มอเตอร์ SERVO 5V ในการเปิดหรือปิดประตู่รั้ว

1.3.9 ใช้แหล่งจ่ายไฟ 220 V , 5V และ 3V

1.3.10 ทดสอบการทางานพร้อมทั้งแก้ไขปรับปรุงข้อผิดพลาดที่อาจจะเกิดจากอุปกรณ์ในระบบว่าจะทำงานเต็มที่หรือไม่

1.4 ประโยชน์ที่คาดว่าจะได้รับ

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์

1.4.1 ได้ระบบตรวจสอบสถานะภายในบ้าน

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

1.4.2 ได้ระบบตรวจสอบสถานการณ์ทำงานของระบบภายในบ้านควบคุมด้วยแอพพลิเคชั่นที่มีประสิทธิภาพและความพึงพอใจ

1.4.3 ได้ฝึกทักษะการเขียนโปรแกรม Arduino สั่งเปิดหรือปิดอุปกรณ์ต่าง ๆ ผ่านมือถือได้

1.4.4 ได้ศึกษาเกี่ยวกับการทำงานของระบบต่าง ๆ เพื่อนำไปปรับใช้ในชีวิตประจำวันได้

1.5 คำนิยามศัพท์

1.5.1 ระบบอัตโนมัติ คือ ระบบใด ๆ หรือกลไกที่สามารถเริ่มทำงานได้ด้วยตัวเอง โดยทำงานตามโปรแกรมที่วางไว้ เช่น ระบบรดน้ำอัตโนมัติ ระบบตอบรับโทรศัพท์ระบบอัตโนมัติอาจเป็นการใช้กลไกทางคอมพิวเตอร์หรืออุปกรณ์อิเล็กทรอนิกส์ที่จะควบคุมการทำงานถูกต้องต่อเมื่อมีการวางแผนหรือโปรแกรมโดยมนุษย์ทั้งสิ้น

บทที่ 2

เอกสารและโครงงานที่เกี่ยวข้อง

โครงงานได้ทำการศึกษาแนวความคิดเอกสารและงานวิจัยที่เกี่ยวข้องดังต่อไปนี้

2.1 Arduino

2.2 โพรโทบอร์ด

2.3 NodeMCU ESP8266

2.4 รีเลย์(Relay)

2.5 มอเตอร์

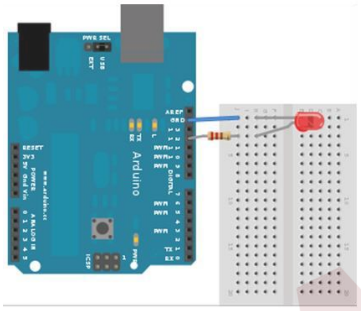
2.6 งานโครงงานเกี่ยวข้อง

2.1 Arduino

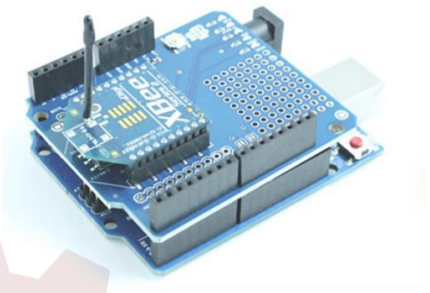
Arduino คืออะไร Arduino อ่านว่า (อา-ดู-อิ-โน่ หรือ อาดุยโน่) เป็นบอร์ดไมโครคอนโทรลเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัว บอร์ด Arduino ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

ความง่ายของบอร์ด Arduino ในการต่ออุปกรณ์เสริมต่าง ๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิกส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขา I/O ของบอร์ด (ดูตัวอย่างรูปที่ 1) หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม (Arduino Shield) ประเภทต่าง ๆ (ดูตัวอย่างรูปที่ 2) เช่น Arduino XBee Shield, Arduino Music Shield, Arduino Relay Shield, Arduino GPRS Shield เป็นต้น มาเสียบกับบอร์ดบนบอร์ด Arduino แล้วเขียนโปรแกรมพัฒนาต่อได้



รูปที่ 2.1 บอร์ด Arduino ต่อกับ LED



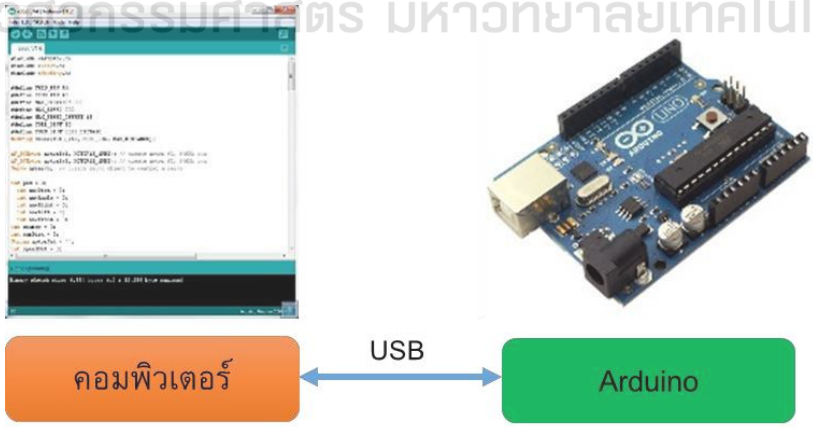
รูปที่ 2.2 บอร์ด Arduino ต่อกับบอร์ด XBee Shield

2.1.2 จุดเด่นที่ทำให้บอร์ด Arduino เป็นที่นิยม

- ง่ายต่อการพัฒนามีรูปแบบคาสั่งพื้นฐาน ไม่ซับซ้อนเหมาะสำหรับผู้เริ่มต้น
- มี Arduino Community กลุ่มคนที่ร่วมกันพัฒนาที่แข็งแรง
- Open Hardware ทำให้ผู้ใช้ สามารถนำบอร์ดไปต่อยอดใช้งานได้หลายด้าน
- ราคาไม่แพง
- Cross Platform สามารถพัฒนาโปรแกรมบน OS ใดก็ได้

2.1.3 รูปแบบการเขียนโปรแกรมบน Arduino

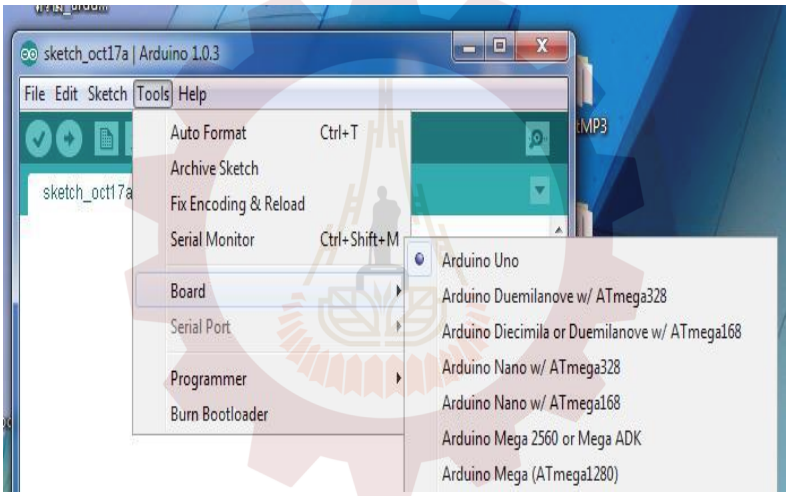
สำนักวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่ 2.3 รูปแบบการเขียนโปรแกรมบน Arduino

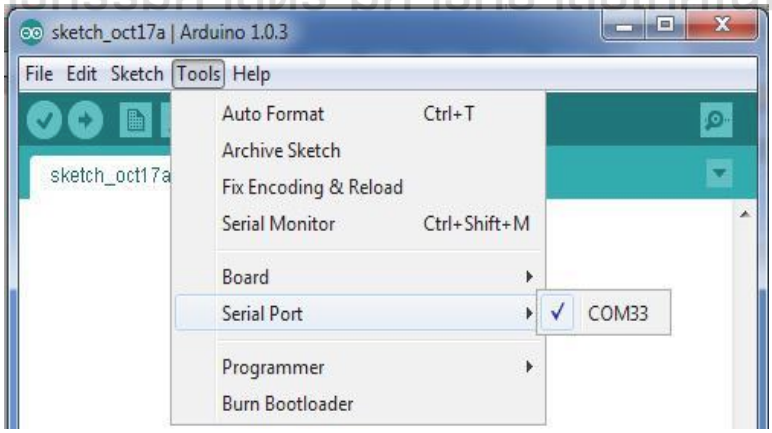
1) เขียนโปรแกรมบนคอมพิวเตอร์ ผ่านทางโปรแกรม ArduinoIDE ซึ่งสามารถดาวน์โหลดได้จาก Arduino.cc/en/main/software

2) หลังจากที่เขียนโคด้ โปรแกรมเรียบร้อยแล้วให้ผู้ทำงานเลือกรุ่นบอร์ด Arduino ที่ใช้และหมายเลข Com port



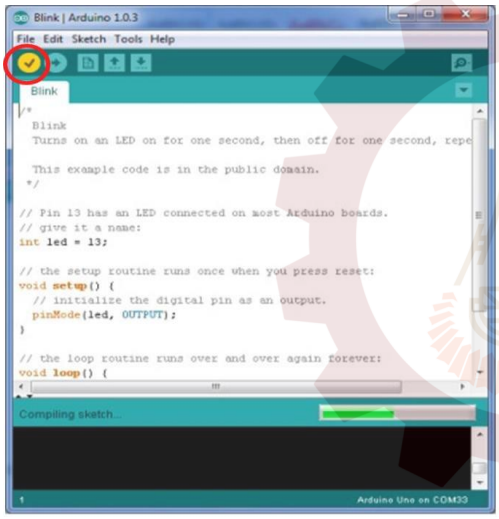
รูปที่ 2.4 เลือกรุ่นบอร์ด Arduino ที่ต้องการ upload

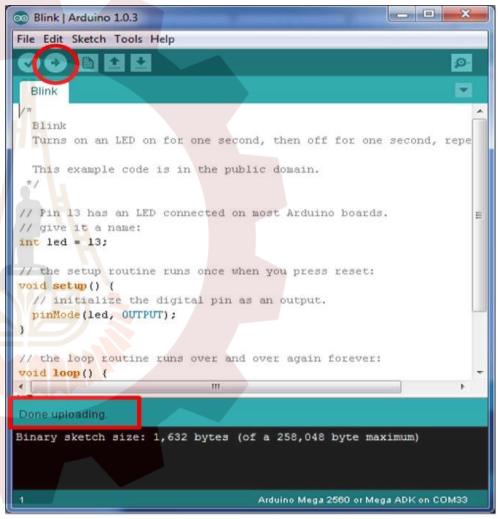
สาขาวิชาวิศวกรรมเครื่องกล
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่ 2.5 เลือกหมายเลข Comport ของบอร์ด

3) กดปุ่มVerify เพื่อตรวจสอบความถูกต้องและทำการ Compile โค้ดโปรแกรม จากนั้นกดปุ่ม Upload โค้ดโปรแกรมไปยังบอร์ด Arduino ผ่านทางสาย USB เมื่ออับโหลดเรียบร้อยแล้วจะแสดงข้อความแถบข้างล่าง “Done uploading” และบอร์ดจะเริ่มทางานตามที่เขียนโปรแกรมไว้ได้ทันที

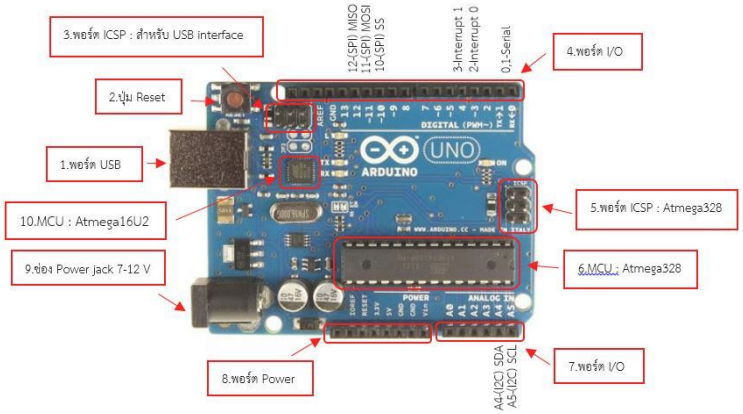




รูปที่ 2.6 กดปุ่ม Verify เพื่อตรวจสอบความถูกต้อง **รูปที่ 2.7** Upload โค้ดโปรแกรม

สถาบันวิทยบริการ มหาวิทยาลัยเทคโนโลยีสุรนารี

2.1.4 Layout & Pin out Arduino Board (Model: Arduino UNO R3)



รูปที่ 2.8 ส่วนประกอบของ Layout & Pin out Arduino Board

1. USB Port : ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด

2. Reset Button : เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทางานใหม่

3. ICSP Port ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2

4. I/O Port : Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้บาง Pin จะทำหน้าที่อื่น ๆ เพิ่มเติมด้วย เช่น Pin0, 1 เป็นขา Tx, Rx Serial, Pin3, 5, 6, 9, 10 และ 11 เป็นขา PWM

5. ICSP Port : Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader

6. MCU : Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino

7. I/O Port : นอกจากจะเป็น Digital I/O แล้วเปลี่ยนเป็นช่องรับสัญญาณอนาล็อกตั้งแต่ขา A0-A5

8. Power Port : ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอกประกอบด้วยขาไฟเลี้ยง +3.3 V, +5V, GND, Vin

9. Power Jack : รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V

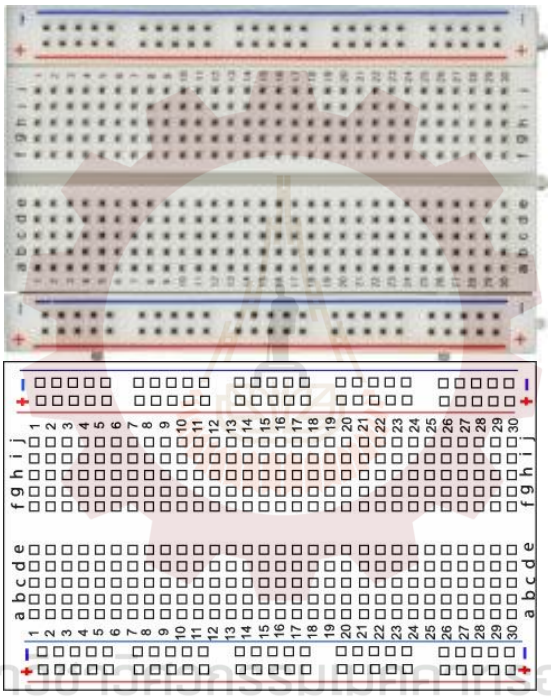
10. MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่าน Atmega16U2

2.2 โพรโทบอร์ด

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

โพรโทบอร์ด (อังกฤษ: Protoboard) หรือเบรดบอร์ด (อังกฤษ: Breadboard) เป็ นบอร์ดที่ใช้ทดลองวงจรอิเล็กทรอนิกส์ ลกั ษณะเป็ นแผ่นพลาสติกหนาสีขาว บนแผ่นมีรูเรียงกันจานวนมาก ภายในรูมีตำนำไฟฟ้ำซึ่งเชื่อมต่อกันในรูปแบบที่มีการกำหนดไว้ เวลาทดลองก็เสียบขาของอุปกรณ์อิเล็กทรอนิกส์ลงไปให้ตัวนำภายในเชื่อมวงจรถึงกันและอาจใช้สายไฟเสียบลงรูเพื่อเชื่อมวงจรไฟฟ้าได้เช่นกันข้อดีของโพรโทบอร์ดคือ ไม่ต้องออกแบบแผงวงจรและไม่ต้องบัดกรี แต่มีข้อเสียคือใช้ทดลองวงจรที่ทางานที่ความถี่สูง ๆ ไม่ได้ เนื่องมีปัญหาเรื่องสัญญาณรบกวนในวงจร

2.5.1 การใช้งานแผ่นโพรโทบอร์ดแผ่นโพรโทบอร์ดจะเป็นแผ่นที่จะใช้ในการทดลองเกี่ยวกับวงจรอิเล็กทรอนิกส์ทั่วไป โดยที่แผ่นโพรโทบอร์ดจะมีช่องสาหรับใช้เสียบขาอุปกรณ์ได้ซึ่งการทดลองการต่อวงจรอิเล็กทรอนิกส์ที่ต่อกับแผ่นโพรโทบอร์ดจะทาให้อุปกรณ์ไม่เสียหาย

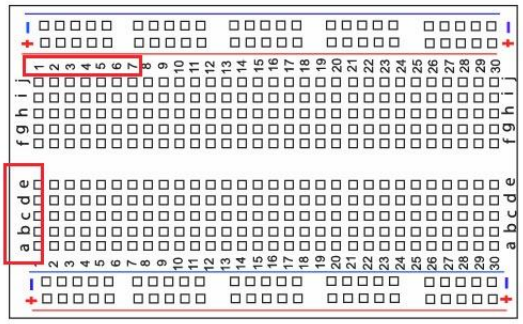


สาขาวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 2.9 แผ่นโพรโทบอร์ด

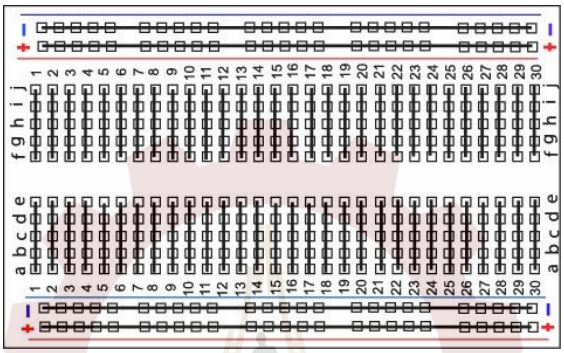
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

แผ่นโพรโทบอร์ด (บางรุ่น) จะมีตัวเลขและตัวอักษรภาษาอังกฤษ เขียนกำกับไว้ เช่น ช่องที่ a1 ช่องที่ g1 เป็ นต้น



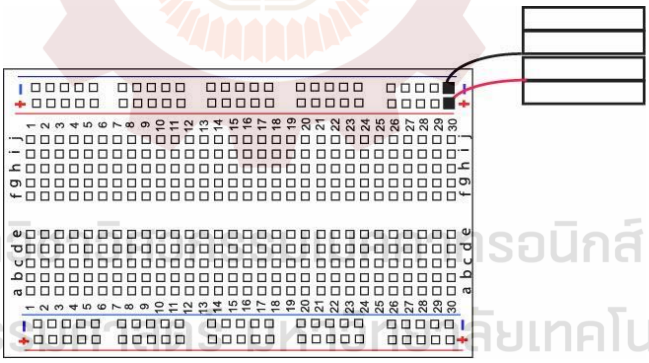
รูปที่ 2.10 อักษรกำกับประจำหลัก

การต่อวงจรภายในแผ่นโพรโทบอร์ดจะเป็นดังรูปกล่าวคือ ด้านบนและด้านล่างจะใช้สำหรับต่อไปเลี้ยงวงจร ซึ่งวงจรด้านล่างจะต่อถึงกันทั้งแถว ส่วนการต่อวงจรด้านใน จะต่อถึงกันตามแนวตั้งดังรูปที่ 2.15



รูปที่ 2.11 การต่อวงจรวงจรภายในแผ่นโพรโทบอร์ด

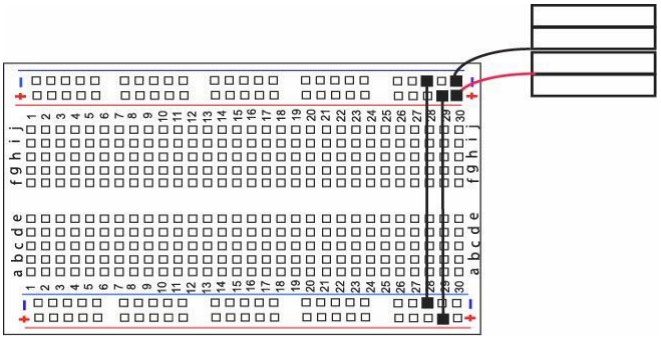
การต่อไฟเลี้ยงวงจรสามารถต่อไฟ + และ - เข้ากับแผ่นโพรโทบอร์ดดังรูป



สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

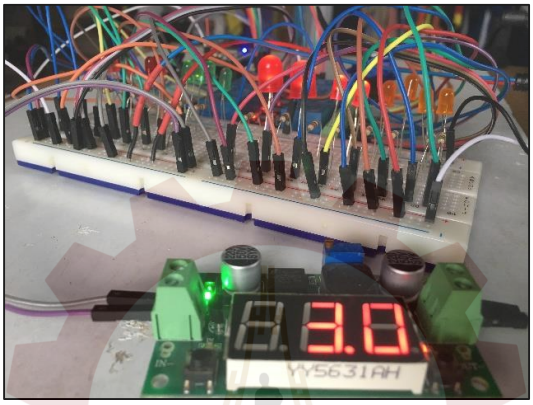
รูปที่ 2.12 การต่อแหล่งจ่ายไฟกับแผ่นโพรโทบอร์ด

และถ้าต้องการการต่อไฟเลี้ยงทั้ง 2 ข้างก็สามารถใช้สายโทรศัพท์ต่อเข้าด้วยกันทั้ง 2 ด้าน

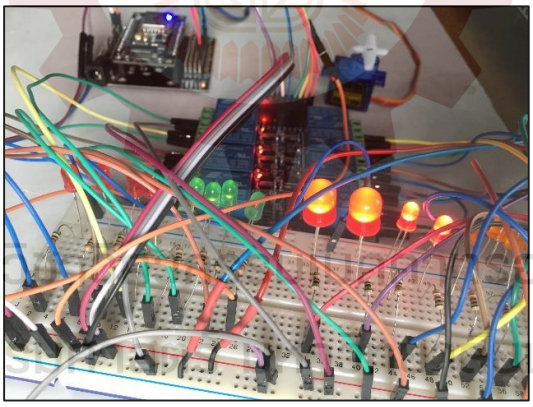


รูปที่ 2.13 การต่อแหล่งจ่ายฟกับแผ่นโพรโทบอร์ดทั้ง 2 ด้าน

ตัวอย่างการต่อวงจรโดยใช้แผ่นโพรโทบอร์ด



รูปที่ 2.14 การต่ออุปกรณ์กับแผ่นโพรโทบอร์ด 1



รูปที่ 2.15 การต่ออุปกรณ์กับแผ่นโพรโทบอร์ด 2

สาขาวิชาวิศวกรรมไฟฟ้าและอิเล็กทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

2.3 NodeMCU ESP8266

บอร์ด NodeMCU/ESP8266 รองรับการเชื่อมต่อเครือข่ายมาตรฐาน IEEE 802.11b/g/n ให้สามารถเชื่อมต่อกับเครือข่าย WiFi ได้เพื่อให้บอร์ดเชื่อมต่อกับอินเทอร์เน็ตต่อไปส่วนที่สองเป็นการทำให้บอร์ด NodeMCU/ESP8266 เชื่อมต่อกับเครือข่าย WiFi แบบ SmartConfig ข้อดีของโหมด SmartConfig คือผู้ใช้งานบอร์ดไม่ต้องระบุ SSID (ชื่อของเครือข่าย WiFi) และPassword ลงในโปรแกรมเหมือนกับในการทดลองส่วนแรก เพราะในบอร์ด NodeMCU/ESP8266 จะทำหน้าที่

เป็น Access Point ให้เราสามารถตั้งค่าเครือข่ายให้กับบอร์ด NodeMCU/ESP8266 โดยตรงผ่านเบราว์เซอร์อุปกรณ์สื่อสารไร้สายที่มีโมดูล WiFi หลังเสร็จสิ้นการตั้งค่าบอร์ด NodeMCU/ESP8266 จะทำการ Restart เราสามารถตั้งค่าได้ล่วงหน้าว่าหลัง Restart แล้วจะให้บอร์ดกลับสู่โหมดทำงานตามปกติ หรืออยู่โหมด SmartConfig หรือ จะทำงานแบบปกติได้อีกด้วย โดยการกดปุ่มที่ได้ตั้งค่าไว้ในตัวโปรแกรมเพื่อเลือกโหมดการทำงานนั้นเอง

คุณสมบัติด้าน WiFi ของ NodeMCU/ESP8266 เนื่องจาก NodeMCU/ESP8266 สายอากาศในตัว จึงทำให้สามารถเชื่อมต่อ WiFi ได้โดยไม่ต้องต่ออุปกรณ์หรือสายอากาศเพิ่มเติม และคุณสมบัติทางเทคนิคที่สำคัญของโมดูล WiFi บน NodeMCU/ESP8266 มีดังนี้

- รองรับมาตรฐาน IEEE802.11 b/g/n

- รองรับการทำงานแบบ WiFi Direct (P2P) และ SoftAP

- TCP/IP Stack

- วงจรสวิตช์ TR, สายอากาศที่มี Balun, LNA หรือวงจรขยายสัญญาณรบกวนต่ำ, วงจรขยายกำลังและแมตชิ่งเน็ตเวิร์กเพื่อการรับคลื่นวิทยุอย่างมีประสิทธิภาพ

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

- วงจรเฟสล็อกลูปและวงจรควบคุมสัญญาณออสซิลเลเตอร์แบบดิจิตอล (DCXO) เพื่อช่วยบริหารจัดการด้านความถี่

- วงจรบริหารกำลังไฟฟ้าและวงจรควบคุมไฟเลี้ยงคงที่ เพื่อช่วยให้วงจร WiFi ใช้กำลังได้อย่างเหมาะสม โดยปกติต้องการกระแสไฟฟ้า 0.9 mA ในขณะแสตนด์บาย, 135-215 mA ขณะส่งข้อมูล, 60 mA เมื่อรับข้อมูล, 1 mA ในโหมดประหยัดพลังงาน และ 0.5μA ในขณะปิด

- กำลังส่ง +19.5dBm เมื่อทำงานในโหมด 802.11b

ฟังก์ชั่นของโปรแกรม serial.begin

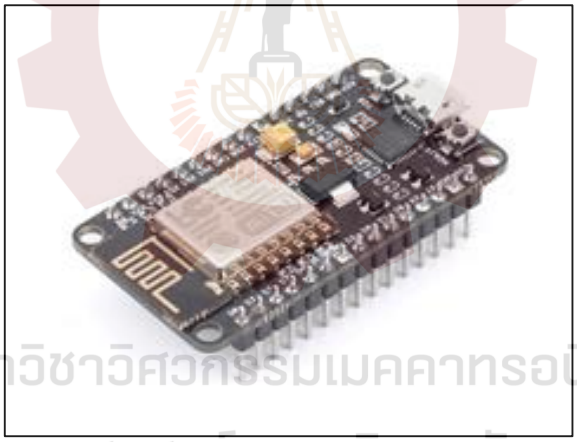
serial.begin เป็นฟังก์ชั่นที่ใช้ในการตั้งค่าความเร็วในการส่งข้อมูลของพอร์ต serial ที่เชื่อมต่อระหว่างบอร์ดกับคอมพิวเตอร์

Syntax:
serial.begin(speed)
Parameter:
speed: **ความเร็วในการรับส่งข้อมูล หน่วยเป็นบิตต่อวินาที** (baud)

- serial.print

serial.print **เป็นฟังก์ชั่นเพื่อให้บอร์ดส่งค่าข้อมูลออกไปพิมพ์ทางพอร์ต** serial

Syntax:
serial.print(val)
serial.print(val, format)



สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 2.16 บอร์ด NodeMCU ESP8266 (ESP-12E)

2.4 รีเลย์ (Relay)

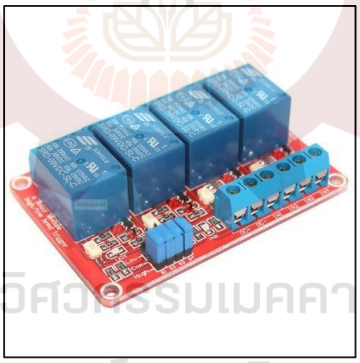
รีเลย์ (Relay) คือ อุปกรณ์อิเล็กทรอนิกส์ที่ทำหน้าที่เป็นสวิตช์ ตัด-ต่อวงจร โดยใช้แม่เหล็กไฟฟ้าและการที่จะให้ทำงานก็ต้องจ่ายไฟให้เป็นไปตามที่กำหนด เมื่อจ่ายไฟให้กับตัวรีเลย์ รีเลย์จะทำให้หน้าสัมผัส ติดกันกลายเป็นวงจรปิด และตรงข้ามทันทีที่ไม่ได้จ่ายไฟให้รีเลย์ก็จะกลายเป็นวงจรเปิดไฟ เมื่อมีการป้อนให้กับตัว รีเลยก็จะเป็นไฟที่มาจากเพาเวอร์ของเครื่อง ดังนั้นทันทีที่เปิดเครื่องก็จะทำให้รีเลย์ทำงาน

2.3.1 ประเภทของรีเลย์

เป็นอุปกรณ์ทำหน้าที่เป็นสวิตช์ มีหลักการทำงานคล้ายกับขดลวดแม่เหล็กไฟฟ้าหรือโซลินอยด์ (Solenoid) รีเลย์ใช้ในการควบคุมวงจรไฟฟ้าได้อย่างหลากหลาย รีเลย์เป็นสวิตช์ควบคุมที่ทำงานด้วยไฟฟ้า แบ่งออกตามลักษณะการใช้งานได้เป็น 2 ประเภทคือ

– รีเลย์กาลัง (Power relay) หรือมักเรียกกันว่า คอนแทกเตอร์ (Contactor or Magnetic contactor) ใช้ในการควบคุมไฟฟ้ากาลงั มีขนาดใหญ่กว่า รีเลย์ธรรมดา

– รีเลย์ควบคุม (Control relay) มีขนาดเล็กกำลังไฟฟ้าต่ำใช้ในวงจรควบคุมทั่วไปที่มีกำลังไฟฟ้าไม่มากนักหรือเพื่อการควบคุมรีเลย์หรือคอนแทกเตอร์ขนาดใหญ่ รีเลย์ควบคุมบางทีเรียกกันง่าย ๆ ว่า"รีเลย์"



สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 2.17 รีเลย์กำลัง

2.3.2 ชนิดของรีเลย์

การแบ่งชนิดของรีเลย์สามารถแบ่งได้ 11 แบบ คือ ชนิดของรีเลย์แบ่งตามลักษณะของคอยล์หรือแบ่งตามลกั ษณะการใช้งาน (Application) ได้แกรีเลย์ดังต่อไปนี้

- รีเลย์กระแส(Current relay) คือ รีเลย์ที่ทำงานโดยใช้กระแส มีทั้งชนิดกระแสขาด (Undercurrent) และกระแสเกิน (Over current)

- รีเลย์แรงดัน(Voltage relay)คือ รีเลย์ ที่ทำงานโดยใช้แรงดันมีทั้งชนิดแรงดันขาดและแรงดันเกิน (Over voltage)

- รีเลย์ช่วย(Auxiliary relay) คือ รีเลย์เวลาใช้งานต้องประกอบเข้ากับรเลย์ชนิดอื่นจึงจะทำงานได้

- รีเลย์กำลัง(Power relay) คือ รีเลย์ที่รวมเอาคุณสมบัติของรีเลย์กระแส และรีเลย์แรงดันเข้าด้วยกัน

- รีเลย์เวลา(Time relay)คือ รีเลย์ทำงานโดยมีเวลาเข้ามาเกี่ยวข้องด้วยย ซึ่งมด้วยกัน 4 แบบ คือ

- รีเลย์กระแสเกินชนิดเวลาผกผันกับกระแส (Inverse time over current relay) คือ รีเลย์ที่มีเวลาทางานเป็นส่วนกลับกับกระแส

- รีเลย์กระแสเกินชนิดทางานทันที (Instantaneous over current relay) คือ รีเลย์ที่ทำงานทันทีทันใด เมื่อมีกระแสไหลผ่านเกินกว่าที่กำหนดที่ตั้งไว้

- รีเลย์แบบดิฟฟินิตไทม์เล็ก (Definite time lag relay) คือ รีเลย์ที่มีเวลาการทำงานไม่ขึ้นอยู่กับความมากน้อยของกระแสหรือค่าไฟฟ้าอื่น ๆ ที่ทำให้เกิดงานขึ้น

- รีเลย์แบบอินเวอสดิฟฟินิตมินิมมั่มไทม์เล็ก (Inverse definite time lag relay) คือ รีเลย์ที่ทำงาน โดยรวมเอาคุณสมบัติของเวลาผกผันกับกระแส (Inverse time) และแบบดิฟฟินิตไทม์แล็ก (Definite time lag relay) เข้าด้วยกัน

- รีเลย์กระแสต่าง (Differential relay) คือ รีเลย์ที่ทำงานโดยอาศัยผลต่างของกระแส

- รีเลย์มีทิศ (Directional relay) คือ รีเลย์ที่ทำงานเมื่อมีกระแสไหลผิดทิศทางมีแบบรีเลย์กำลังมีทิศ (Directional power relay) และรีเลย์กระแสมีทิศ (Directional current relay)

สำนักวิทยาศาสตร์และนวัตกรรมดิจิทัล
สำนักวิทยาศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

- รีเลย์ระยะทาง (Distance relay) คือ รีเลยระยะทางมีแบบต่าง ๆ ดังนี้

- รีแอกแตนซ์รีเลย์ (Reactance relay)
- อิมพีแดนซ์รีเลย์ (Impedance relay)
- โมห์รีเลย์ (Mho relay)
- โอห์มรีเลย์ (Ohm relay)
- โพลาไรซ์โมห์รีเลย์ (Polaized mho relay)
- ออฟเซทโมห์รีเลย์ (Off set mho relay)

- รีเลย์อุณหภูมิ (Temperature relay) คือ รีเลย์ที่ทำงานตามอุณหภูมิที่ตั้งไว้

- รีเลย์ความถี่ (Frequency relay) คือ รีเลย์ที่ท างาน เมื่อความถี่ของระบบต่ากว่าหรือมากกว่าที่ตั้ง

- บูคโฮลซ์รีเลย์ (Buchholz's relay) คือ รีเลย์ที่ทำงานด้วยก๊าซใช้กับหม้อแปลงที่แช่อยู่ในน้ำมัน เมื่อเกิดฟอลต์ขึ้นภายในหม้อแปลง จะทำให้น้ำมันแตกตัวและเกิดก๊าซขึ้นภายในไปดันหน้าสัมผัสให้รีเลย์ทำงาน

2.5 มอเตอร์ (Motor)

เป็นเซอร์โวมอเตอร์ขนาดเล็กที่เหมาะกับงานหุ่นยนต์ ควบคุมการทำงานด้วยสัญญาณ PWM สามารถสั่งงานผ่านไมโครคอนโทรลเลอร์ เพื่อควบคุมตำแหน่งได้ง่าย

คุณสมบัติ

- ขนาด 23.2x12.5x22 mm.
- น้ำหนัก 14 กรัม
- เฟืองโลหะ **5** ชิ้น
- **ความเร็ว** 0.12 **วินาที**/60 **องศา ที่แรงดัน** 4.8v **และ** 0.10 **วินาที** / 60 **องศา ที่แรงดัน** 6v
- **แรงบิด** 1.5kg.cm **ที่แรงดัน** 4.8V **และ** 1.8 kg.cm **ที่แรงดัน** 6V
- **ทำงานที่แรงดัน** 4.8 **ถึง** 6v
- **ควบคุมการทำงานด้วยความกว้างของสัญญาณพัลส์ช่วงบวก** 1 **ถึง** 2 **มิลลิวินาที**
- **ระยะการเคลื่อนที่** 180 **องศา**
- **ความยาวสาย** 24 **ซม.**

รูปที่ 2.18 เซอร์โวมอเตอร์ MG90S-180

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

2.6 งานวิจัยที่เกี่ยวข้อง

สรกฤชและพิมลลักษณ์ (2551) การพัฒนาระบบควบคุมไฟฟ้าโดยใช้ Zigbee วงจรสำหรับรับ-ส่งขอ้ มูล ประมวลผลข้อมูลที่ได้รับและควบคุมการเปิ ด-ปิ ดไฟฟ้าของอุปกรณ์ไฟฟ้าตามที่ส่วนของการประมวลผลสั่ง โดยใช้ Zigbee ไมโครคอนโทรลเลอร์และ Relay, Web Application ส าหรับใช้งานระบบผ่านทางอินเทอร์เน็ต โปรแกรมสำหรับควบคุมวงจรในโปรแกรมส าหรับควบคุมการเปิด-ปิดไฟฟ้าบนโทรศพท์มือถือจากโครงการที่พัฒนาขึ้นนี้ จะช่วยเพิ่มความสะดวกสบายแก่ผู้ใช้งานได้อย่างมาก เนื่องจากผู้ใช้งานมีวิธีในการเลือกใช้งานระบบได้ถึง 2 ช่องทาง ไม่ว่าจะทางอินเทอร์เน็ตหรือทางโทรศัพท์มือถือ และผู้ใช้งานยังสามารถควบคุมการเปิด-ปิ ดไฟฟ้าได้จากระยะไกลด้วย นอกจากนี้ยังมีผลช่วยให้ลดอัตราการเกิดอัคคีภัยได้อีกด้วยดังนั้น โครงการการพัฒนาระบบควบคุมไฟฟ้าโดยใช้ Zigbee นี้จึงเหมาะแก่การนาไปประยุกต์ใช้ในชีวิตประจำวันหรือนำไปพัฒนาต่อเพื่อเพิ่ม ประโยชน์ และประสิทธิให้มีมากยิ่งขึ้น

ชาริณีและณัฐการ (2553) ศึกษาการเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์โดยใช้ภาษา C/C++ ศึกษาการเขียนภาษา HTML และการสร้าง Dynamic HTML ศึกษาการสร้างเว็บไซต์ที่มีการตอบสนองแบบ Real-time ด้วยเทคโนโลยี AJAX ศึกษาการใช้โปรแกรมประยุกต์ที่ช่วยในการออกแบบและตกแต่งเว็บไซต์เช่น Adobe Dreamweaver, Adobe Photoshop ศึกษาและออกแบบฮาร์ดแวร์ที่ใช้ในการควบคุมอุปกรณ์ผ่านระบบเน็ตเวิร์ก ศึกษาการออกแบบวงจรอิเล็กทรอนิกส์ โดยระบบที่กล่าวมาจะใช้งานหรือควบคุมทั้งหมดผ่านหน้าเวบ็ เพจ โดยใช้ไมโครคอนโทรลเลอร์รับค่าจากการสั่งผ่านหน้าเว็บเพจ ส่งไปยังเครื่องใชไฟฟ้าที่เราตอ้ งการจะควบคุม และยังสามารถตรวจสอบสถานะการท างานของเครื่องใชไฟฟ้าเหล่านั้นได้ โดยในโครงงานนี้เน้นไปที่ควบคุมเครื่องใช้ไฟฟ้าประเภทเครื่องปรับอากาศและหลอดไฟเป็นหลัก โดยระบบการงานหลัก ๆ มีระบบควบคุมเครื่องปรับอากาศ ระบบควบคุมระบบแสงสว่างประธานและคณะ (2555) ระบบควบคุมไฟฟ้าไร้สายผ่านเว็บบราวเซอร์ สร้างระบบควบคุมเครื่องใชไฟฟ้าไร้สายผ่านเว็บบราวเซอร์ เพิ่มความสะดวกสบายให้กับผู้ใช้งาน สามารถควบคุมการปิด-เปิดอุปกรณ์ไฟฟ้าผ่านเวบ็ บราวเซอร์ สามารถสร้างเว็บบราวเซอร์ที่มีระบบรักษาความปลอดภัยด้วยระบบ Login ในส่วนของการใชง้ านระบบควบคุมไฟฟ้า

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

ไร้สายผ่านเว็บบราวเซอร์ จะเริ่มจากการเข้าสู่หน้า Web Browser ของระบบท าการ Login เข้าระบบโดย จะมี 2 สถานะของผู้ใช้คือ ผู้ควบคุมและผู้เฝ้าดูเท่านั้นเมื่อทำการ Login เสร็จเรียบร้อยก็จะเข้าสู่หน้า ควบคุมจะมี 3 หน้าจำนวน 3 Stationโดยแต่ละ Station จะมีปุ่มสสำหรับควบคุมเครื่องใช้ไฟฟ้าได้ จำนวน 8 เครื่องใช้ไฟฟ้า โดยจะมีปุ่มOpen ปุ่ม Close ปุ่ม Open All ปุ่ม Close All ปุ่ม Refresh ปุ่ม Save Chang ช่อสำหรับตั้งเวลาเปิด-และช่องสาหรับเลือกการควบคุมแบบกาหนดเองแบบอัตโนมัติ โดยผู้ใช้ สามารถควบคุมเครื่องใช้ไฟฟ้าตามแต่ดุลยพินิจในของผู้ใช้แต่ละคน

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 3

วิธีการจัดทำโครงงาน

ในการสร้างชุดโมเดลระบบตรวจสอบสถานะภายในบ้านมีขั้นตอนและกรรมวิธีที่ทำให้โครงงานนี้สำเร็จลงได้มีดังนี้

3.1 วัสดุและอุปกรณ์ที่ใช้ในการจัดทำโครงงานได้แก่

ลำดับ	วัสดุอุปกรณ์	จำนวน
1	บอร์ด NodeMCU ESP8266 (ESP-12E)	1 ชุด
2	Relay 8 CH	2 ตัว
3	Led	50 หลอด
4	Switch Power	10 ตัว
5	Computer	1 เครื่อง
6	SERVO Motor	2 ตัว
7	โทรศัพท์มือถือ	1 เครื่อง
8	สว่านมือ , ชุดดอกสว่านเจาะเหล็ก	1 ชุด
9	ไม้ขนาด 30x30 cm หนา 0.5 mm	4 แผ่น
10	แผ่นอะคลิลิค 30x30 cm หนา 0.5 mm	4 แผ่น
11	กล่องอเนกประสงค์	6 แผ่น

ตารางที่ 3.1 วัสดุและอุปกรณ์ที่ใช้ในการจัดทำโครงงาน

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

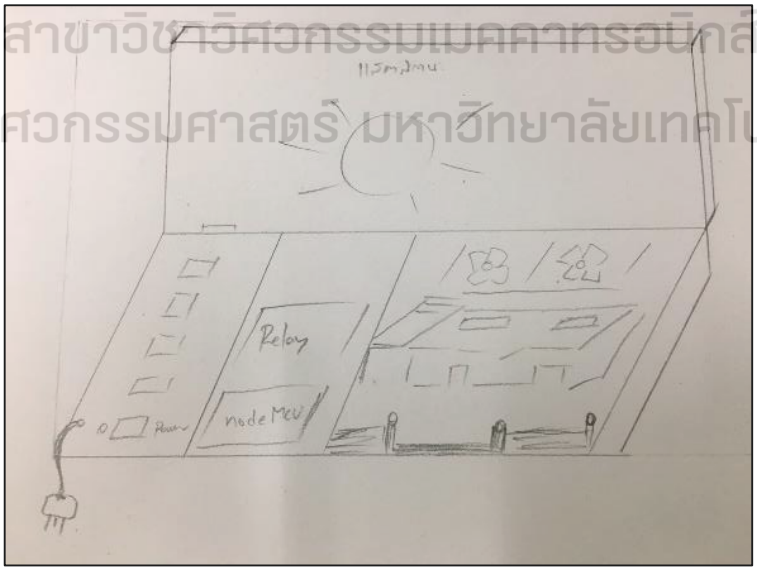
3.2 วิธีการจัดทำโครงงาน

- ศึกษาข้อมูลและกระบวนการควบคุมระบบแบบอัตโนมัติที่ใช้ตัวควบคุมผ่านเครือข่ายโดยใช้แอพพลิเคชั่นตามหัวข้อโครงงานที่ได้รับจากอาจารย์ที่ปรึกษา
- ศึกษาการควบคุมระบบการทำงานและการใช้งานโปรแกรม Arduino รวมทั้งการใช้งานบอร์ด NodeMCU ESP8266 (ESP-12E) และวิธีการติดตั้ง
- ออกแบบระบบเบื้องต้นและอุปกรณ์ที่เกี่ยวข้องพร้อมทั้งหาขนาดและขอใบเสนอราคาจากผู้จำหน่ายของอุปกรณ์แต่ละชนิดเพื่อใช้ในการออกแบบ
- ออกแบบโครงสร้างชิ้นงาน ด้วยโปรแกรม Microsoft Visio และAdobe Photoshop CC ในการสร้างต้นแบบดังรูปที่ 3.3-3.5 นำเสนออาจารย์ที่ปรึกษาโครงงาน

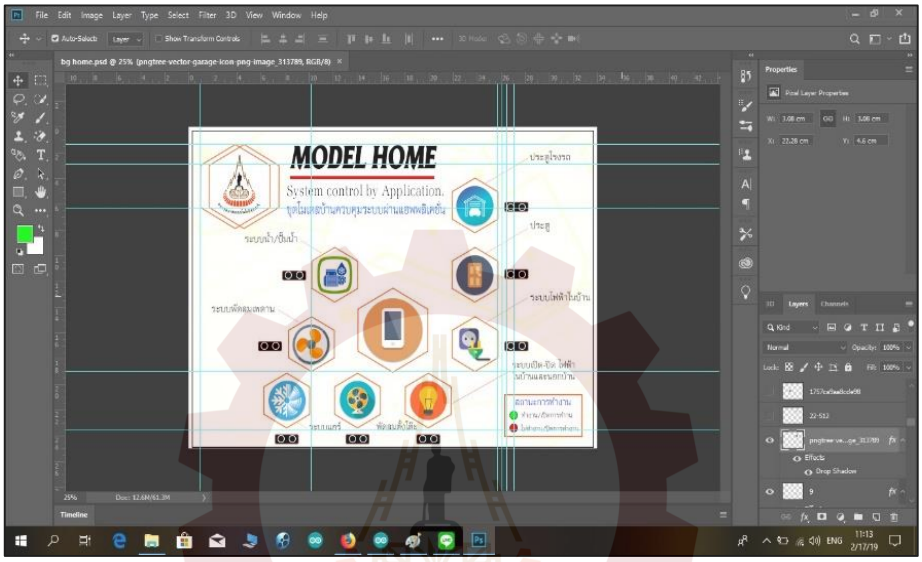
โดยทำการออกแบบโครงสร้างและมีขั้นตอนการดำเนินงานดังนี้

ในการสร้างขั้นตอนออกแบบโครงสร้างชิ้นงานด้วยการร่างรูปแบบชิ้นงานคร่าว ๆ แบบที่ได้จะยึดการนำอุปกรณ์ที่ใช้ในการสร้างชิ้นงานส่วนต่าง ๆ

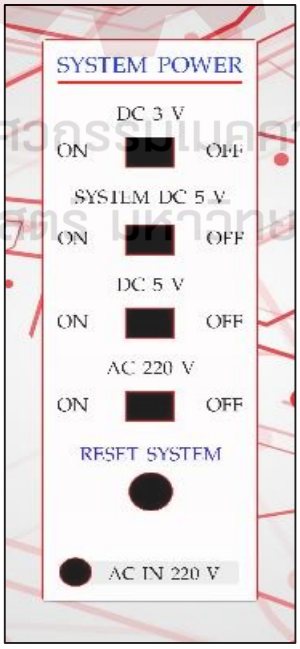
สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่ 3.1 ออกแบบโครงสร้างชิ้นงานด้วยมือ(แบบร่าง)

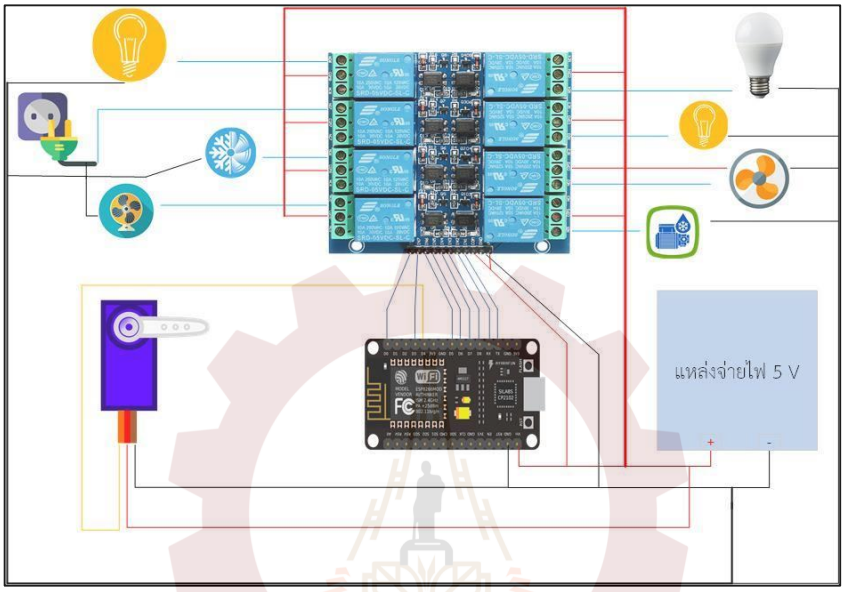


รูปที่ 3.2 ออกแบบโครงสร้างชิ้นงานด้วยคอมพิวเตอร์

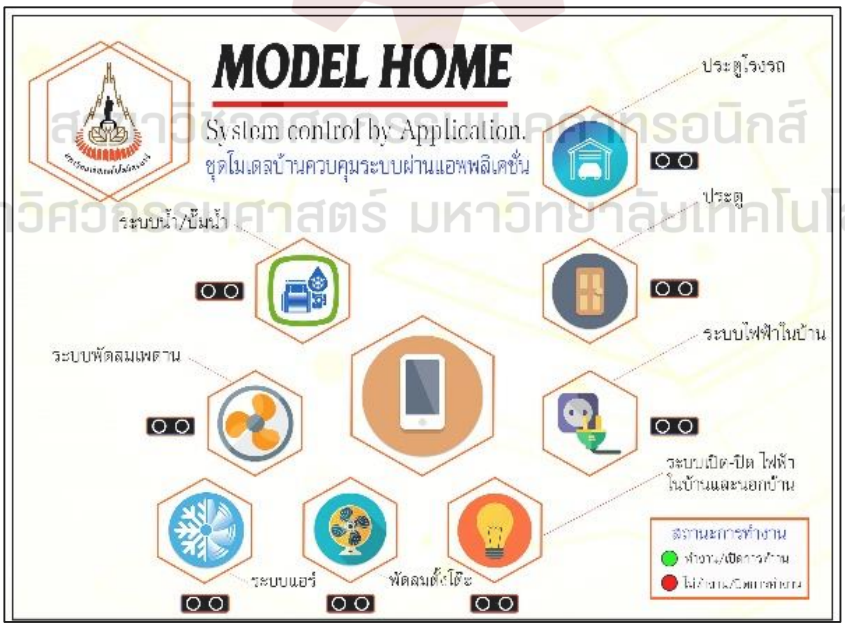


สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 3.3 ออกแบบโครงสร้างชิ้นงานด้วยคอมพิวเตอร์



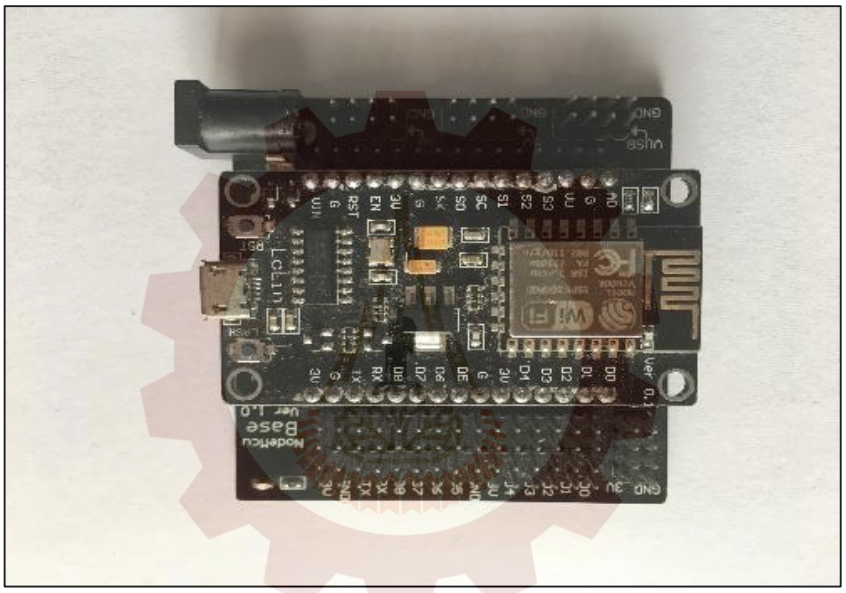
รูปที่ 3.4 ออกแบบวงจรของระบบด้วยคอมพิวเตอร์



รูปที่ 3.5 ออกแบบการเชื่อมต่อระบบด้วยคอมพิวเตอร์

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

– สั่งซื้ออุปกรณ์และเมื่อได้รับสินค้าจึงนำมาประกอบอุปกรณ์ต่าง ๆ เข้าด้วยกันตามรูปแบบที่ได้ออกแบบไว้



รูปที่ 3.6 บอร์ด NodeMCU ESP8266 (ESP-12E)

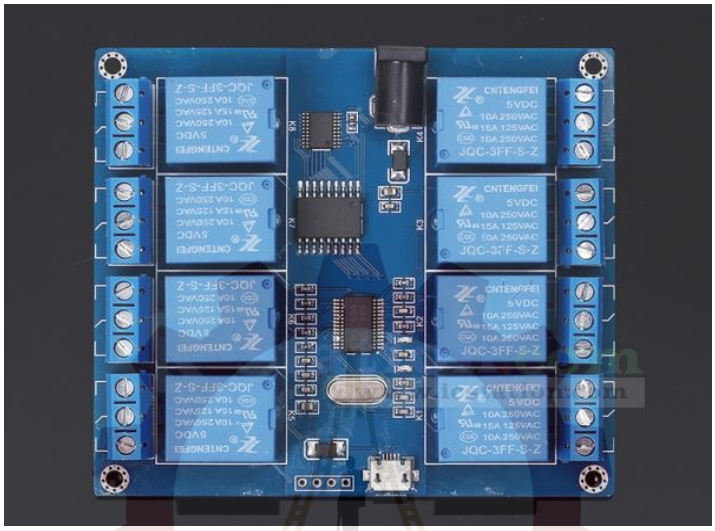
สาขาวิชาวิศวกรรมไฟฟ้าและอิเล็กทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 3.7 สายไฟ

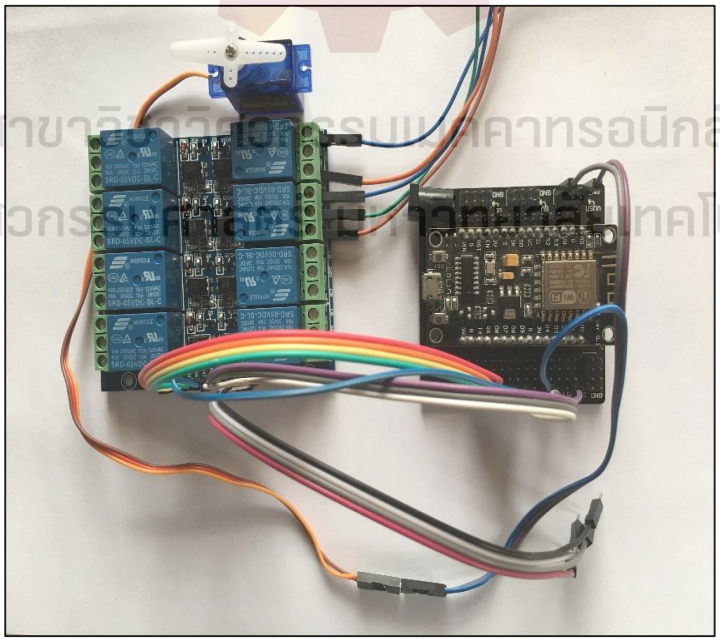
รูปที่ 3.8 มอเตอร์ Servo mg90s

สาขาวิชาวิศวกรรมไฟฟ้าและอิเล็กทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 3.9 พัดลม DC 5V

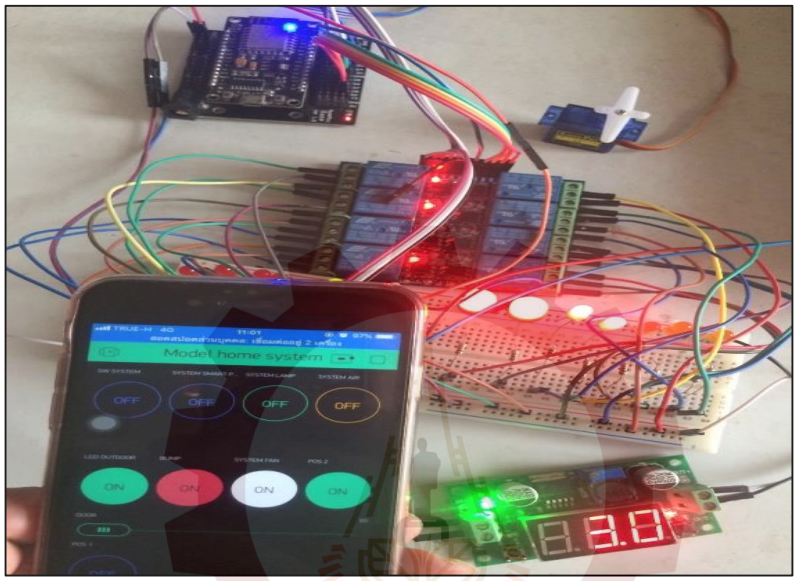


รูปที่ 3.10 8-Channel-5V Relay-Module

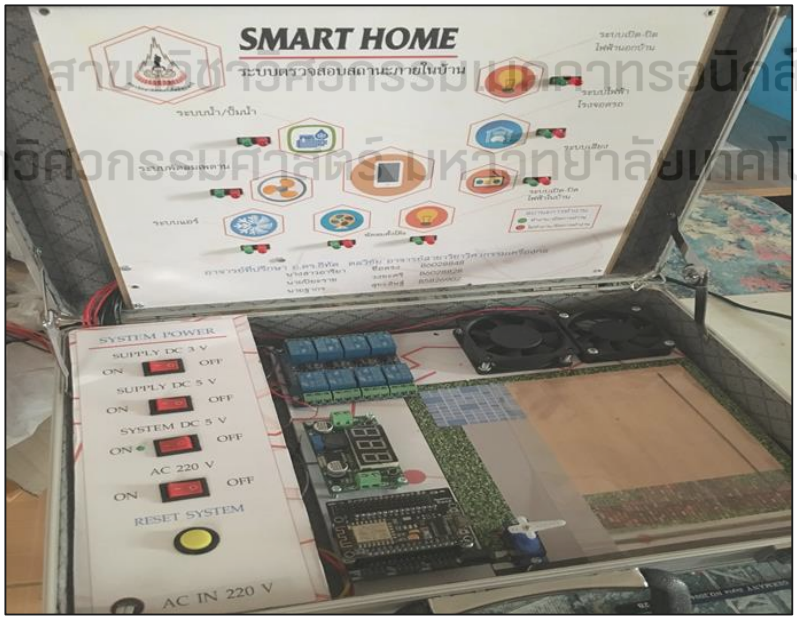


รูปที่ 3.11 แสดงการต่ออุปกรณ์เข้ากับบอร์ด NodeMCU ESP8266 (ESP-12E)

สาขาวิชาวิศวกรรมไฟฟ้าคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

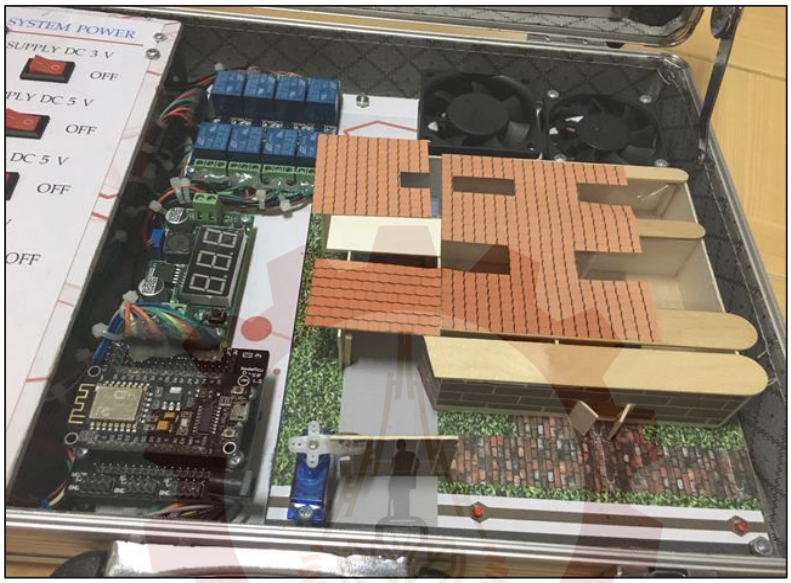


รูปที่ 3.12 แสดงการตรวจสอบสถานะการทำงานของวงจร

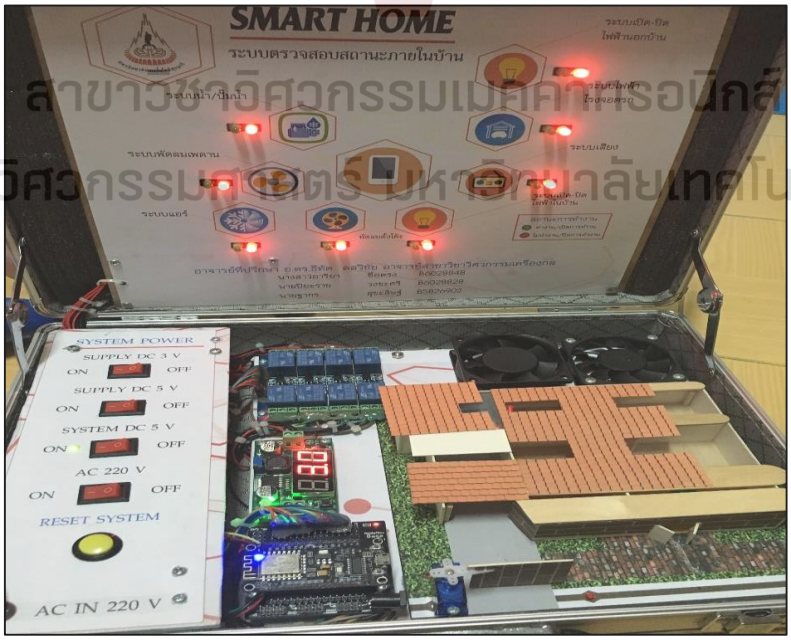


สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 3.13 แสดงการต่ออุปกรณ์ลงกล่อง



รูปที่ 3.14 แสดงการต่ออุปกรณ์ลงกล่องสมบูรณ์



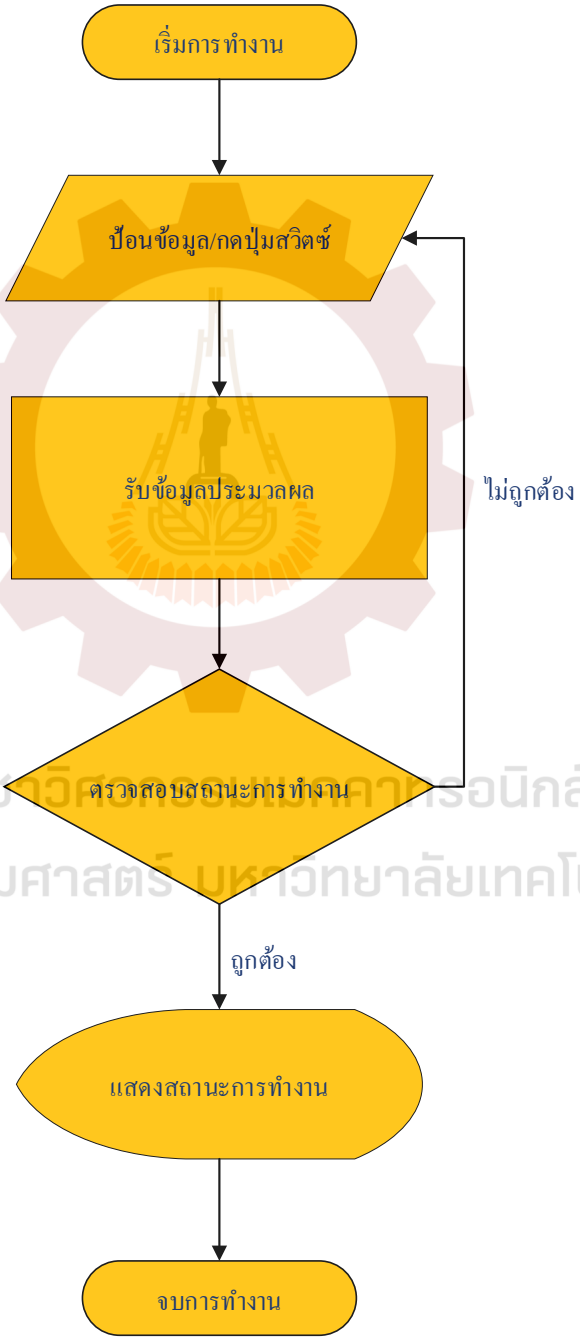
สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 3.15 แสดงการทดสอบระบบการแสดงสถานะการทำงาน

- ออกแบบระบบการควบคุมการทำงานของระบบด้วยแอพพลิเคชั่นที่ใช้ตัวควบคุมบอร์ด NodeMCU ESP8266 (ESP-12E) ในเขียนโปรแกรม Ardunio แล้วจึงทำการ upload เข้าบอร์ด NodeMCU ESP8266 (ESP-12E)พร้อมทั้งทดสอบและเก็บผลการทดสอบโปรแกรมการควบคุมต่าง ๆ
- จัดทำรายงานข้อสรุปโครงงานเพื่อนำเสนอต่อไป

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

แผนผังแสดงการทำงานของระบบ



สาขาวิชาวิศวกรรมไฟฟ้าและอิเล็กทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 3.16 ผังการทำงานระบบตรวจสอบสถานะภายในบ้าน

บทที่ 4

ผลการศึกษา

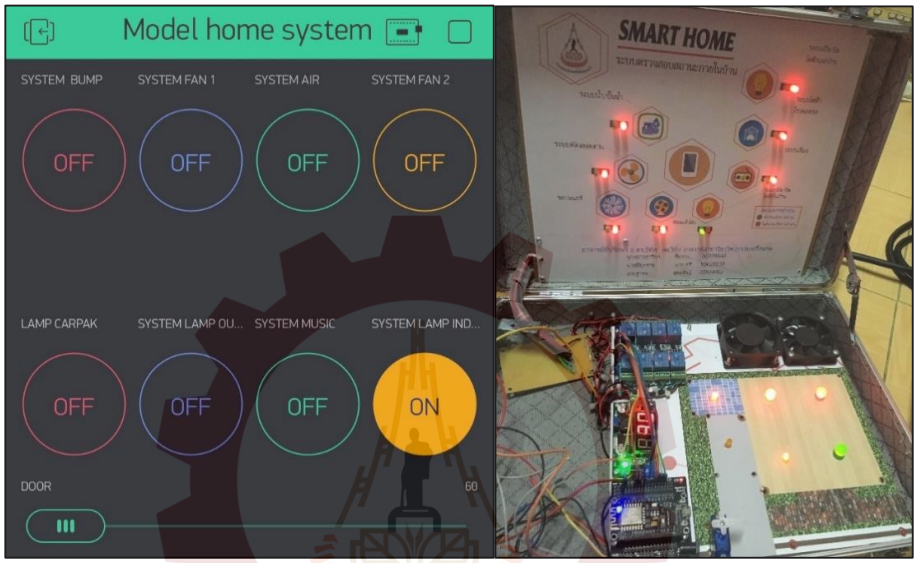
4.1 ผลการทดลองระบบตรวจสอบสถานะภายในบ้าน

แสดงผลผ่านหน้าจอสมาร์ทโฟนและชุดโมเดลของระบบจะแสดงสถานะการทำงานของระบบภายในบ้านว่ามีการทำงานปกติหรือไม่ หากมีการทำงานผิดปกติหรือขัดข้องสามารถตรวจเช็คและตรวจสอบสถานะการทำงานจากสมาร์ทโฟนที่ใช้ระบบเครือข่ายผ่านแอพพลิเคชั่น Blynk ดังแสดงที่หน้าจอสมาร์ทโฟนในรูปที่ 4.1 แสดงสถานะทำงานของระบบ คือ ปิดการทำงาน

รูปที่ 4.1 แสดงผลสถานะการทำงานของระบบ คือ ปิดการทำงานทุกระบบ

จากรูปที่ 4.1 แสดงผลการทดลองสถานะทำงานของระบบตรวจสอบสถานะ คือ ปิดการทำงานทุกระบบภายในบ้าน มีการแสดงสถานะการทำงานที่หน้าจอของระบบและที่หน้าจอสมาร์ทโฟนและหน้าจอของระบบตรวจสอบสถานะภายในบ้าน โดยสถานะที่ระบบทำงานจะแสดงไฟสีเขียวสว่างและสถานะที่ระบบไม่ทำงานคือไฟสีแดงสว่าง

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

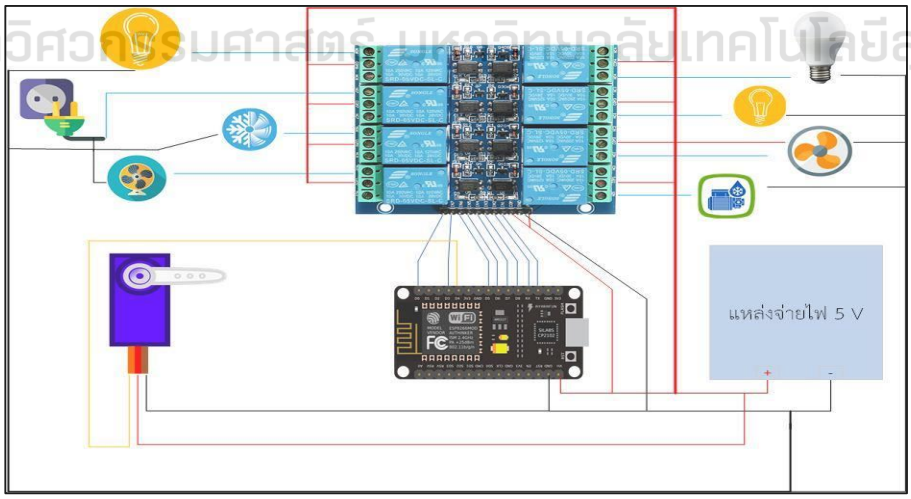


รูปที่ 4.2 แสดงสถานะเปิดการทำงาน 1 ระบบ

จากรูปที่ 4.2 แสดงสถานะทำงานของระบบตรวจสอบสถานะ คือ เปิดการทำงาน 1 ระบบ มีการแสดงสถานะการทำงานของระบบที่หน้าจอของระบบและที่หน้าจอสมาร์ทโฟน

4.2 ลักษณะการทำงานของชุดควบคุม(ต่อชุดควบคุมของระบบ)

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่ 4.3 แสดงลักษณะชุดควบคุมระบบตรวจสอบสถานะภายในบ้าน

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1. สรุปผลการดำเนินโครงงาน

จากการจัดทำโครงงานพบว่า ระบบตรวจสอบสถานะการทำงานภายในบ้าน สามารถนำมาทดสอบหาประสิทธิภาพการทำงานของระบบ ผลปรากฏว่าเป็นไปตามวัตถุประสงค์ที่ตั้งเป้าหมายไว้ โดยการนำไปทดลองใช้จริงกับระบบภายในบ้านซึ่งผลที่ได้ มีการแสดงผลผ่านหน้าจอของระบบและหน้าจอสมาร์ทโฟนซึ่งระบบสามารถทำงานได้ถูกต้องแม่นยำและมีประสิทธิภาพสามารถนำไปพัฒนาต่อยอด แก้ไขปัญหาในชีวิตประจำวันได้และยังนำไปประยุกต์ใช้ในด้านอื่น ๆ ต่อไป

5.2. ข้อเสนอแนะในการดำเนินโครงงาน

- ขั้นตอนการออกแบบ ควรออกแบบตามอุปกรณ์ที่แน่นอนไม่ควรประมาณการและนำมาใช้ในการออกแบบ เพื่อป้องกันข้อผิดพลาดของระยะในการทดสอบ

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์

- ควรวิเคราะห์ปัจจัยต่าง ๆ ที่มีผลต่อการทำงานของระบบ เช่น อุปกรณ์ ระบบเครือข่าย

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

- การออกแบบระบบคอนโทรล โปรแกรมคำสั่ง ไม่ควรให้เกิดการทำงานซ้ำซ้อนเพราะอาจทำให้ระบบเกิดข้อผิดพลาด

- ควรค้นคว้าข้อมูลการจัดทำโครงงานในรูปแบบระบบและการเขียนโปรแกรมควบคุมที่หลากหลายเพื่อใช้ในการประกอบการตัดสินใจเลือกอุปกรณ์

ภาคผนวก

คู่มือการใช้งาน Ardunio UNO เชื่อมต่อกับ Ardunio Microcontroller

PID Controller Design,

[1] PID Controller Design, http://ctms.engin.umich.edu/CTMS/index.php?example=Introduction&section=ControlPID, Last modified: Unknown, Access time: October 13, 2013.

[2] PID Controller, https://sites.google.com/site/suppapongclub/ceaa-luk-rabbpid /, Last modified: Un-known, Access time: October 13, 2013.

[3] ARDUINO, http://www.thainetbeans.com/arduino/start/start.php/, Last modified: Unknown, Access time: October 13, 2013.

[4] ARDUINO, http://arduino.cc//, Last modified: Unknown, Access time: November 2, 2013.

[5] How to use GY80 Arduino ADXL345 Accelerometer, http://blog.oscarliang.net/use-gy80-arduino-adxl345-accelerometer/, Last modified: Unknown, Access time: September 15, 2013.

[6] Self-Balancing Robot, http://www.instructables.com/id/Self-Balancing-Robot/?ALLSTEPS/, Last modified: Unknown, Access time: December 1,2013

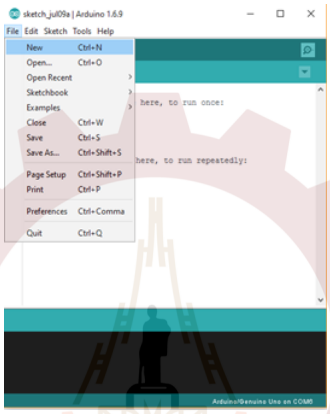
[7] Kalman filtering , http://blog.tkjelectronics.dk/2011/06/guide-gyro-and-accelerometer-kalman-filtering-with-the-arduino/ , modified: Unknown, Access time: June 14th, 2011.

[8] The Balancing Robot , http://blog.tkjelectronics.dk/2012/03/the-balancing-robot/ , modified: Un-known, Access time: March 4th, 2012

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

การทดลองที่ 1 การเชื่อมต่อ NodeMCU/ESP8266 กับเครือข่าย WiFi

1.1 เปิดโปรแกรม Arduino IDE แล้วไปที่เมนู File -> New File



1.2 เขียนโค้ดลงไปในโปรแกรม ดังนี้

```
#include <ESP8266WiFi.h>
const char* ssid     = "Your WiFi SSID";
const char* password = "Your Password";

void setup()
{
  Serial.begin(115200);
  Serial.println("Starting...");

  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(250);
    Serial.print(".");
  }
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
void loop()
{
}
```

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

คำอธิบายโปรแกรม

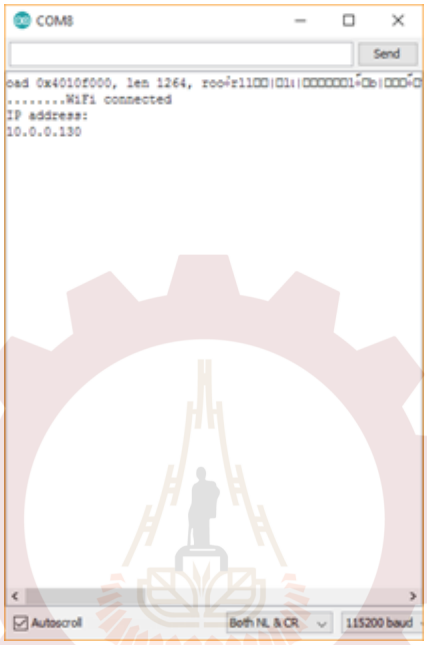
การกำหนด #include เป็นการประกาศเรียกใช้ไลบรารี่ WiFi ของ ESP8266 (ESP8266WiFi.h) แล้วกำหนดค่า SSID และรหัสผ่านของเครือข่าย WiFi ที่ต้องการเชื่อมต่อเก็บไว้ที่ตัวแปร ssid และ password ตามลำดับ

ส่วนภายในฟังก์ชั่น setup() เริ่มจากตั้งค่าความเร็วเริ่มต้นของ Serial ที่ 115,200 bits/s เมื่อตั้งความเร็วสายแล้ว สั่งพิมพ์คำว่า Startingออกที่หน้าจอคอมพิวเตอร์ และเริ่มทำการเชื่อมต่อกับเครือข่าย WiFi ด้วยคำสั่ง WiFi.begin จากนั้นคอยวนตรวจสอบว่าเชื่อมต่อสำเร็จหรือไม่ด้วยคำสั่ง WiFi.status หากยังไม่เชื่อมต่อให้หน่วงเวลา 250 ms และแสดงจุดออกมาที่หน้าจอคอมพิวเตอร์ หากมีการเชื่อมต่อเรียบร้อยแล้วให้แสดงคำว่า WiFi connectedแสดงหมายเลขIP Addressของ NodeMCU/ESP82266 ที่ได้รับมาโดยใช้คำสั่ง WiFi.localIP

1.3 คลิกที่ปุ่ม เพื่อคอมไพล์โค้ด แล้วทำการบันทึกไฟล์เป็น“Connect_Ethernet” ไว้บน Desktop หากไม่มีข้อผิดพลาด ทำการอัพโหลดโดยคลิกปุ่ม เมื่อเสร็จเรียบร้อยแล้วจะปรากฏข้อความ “Done uploading” ขึ้นดังภาพ

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

1.4 คลิกที่ปุ่ม (Serial monitor) และทำการกดสวิตช์ค้างไว้ สังเกตผลลัพธ์บนจอดังภาพ



การทดลองที่ 2 การเชื่อมต่อ NodeMCU/ESP8266 กับเครือข่าย WiFi ในโหมด SmartConfig

ในการทดลองนี้ ผู้ศึกษาจะได้เรียกใช้ไลบรารี่ WiFiManager เพื่อเชื่อมต่อบอร์ด NodeMCU/ESP8266 ในโหมด SmartConfig หรือโหมด Access Point โดยการทำงานของไลบรารี่มีดังนี้

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

I) เมื่อบอร์ดเริ่มต้นทำงาน จะเข้าสู่โหมดปกติหรือ Station และจะพยายามเชื่อมต่อกับเครือข่าย WiFi ที่บันทึกไว้เดิม

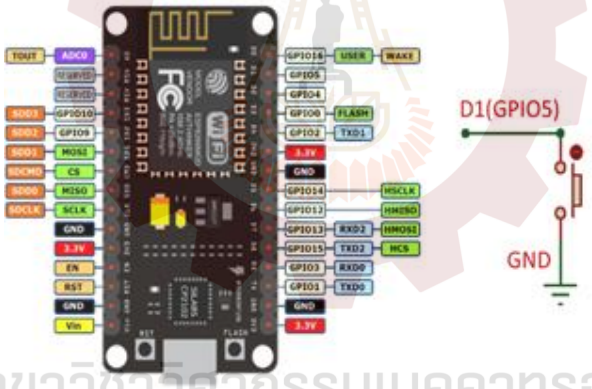
II) หากเชื่อมต่อไม่สำเร็จ บอร์ดจะเปลี่ยนเข้าสู่โหมด SmartConfig หรือ Access Point โดยมี Web Server ที่มี Default IP 192.168.4.1 และ DNS Server ที่จะนำทราฟฟิกทั้งหมดของอุปกรณ์ที่มาต่อกับบอร์ดไปยัง Default Webserver

III) เมื่อนำอุปกรณ์สื่อสารที่มี WiFi Module มาเชื่อมต่อกับบอร์ดในโหมด SmartConfig เมื่อเปิดเว็บเบราว์เซอร์จะปรากฏ Captive Portal ให้ใส่ SSID และ Password ของเครือข่าย WiFi ที่ต้องการตั้งให้บอร์ดเชื่อมต่อ และทำการบันทึกเอาไว้

IV) บอร์ดจะ Reboot และเข้าสู่โหมด Station ตามปกติ วนกลับไปสู่ข้อ I) เราจะผสมผสานความรู้ที่ได้จาก Module NETPIE on NodeMCU/ESP8266 Lab 3 การติดต่อ Digital Input โดยออกแบบให้มีปุ่มกดเพื่อบังคับให้บอร์ดเข้าสู่โหมด SmartConfig หลังจากเริ่มทำงานทันที

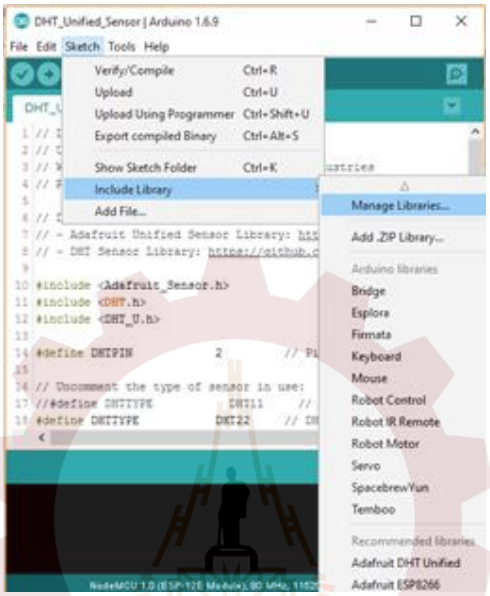
ขั้นตอนการทดลอง

2.1 ต่อสวิตช์ดังภาพ สวิตช์ตัวนี้จะถูกโปรแกรมให้เป็นปุ่มเลือกโหมดของบอร์ดดังที่กล่าวมาข้างต้น

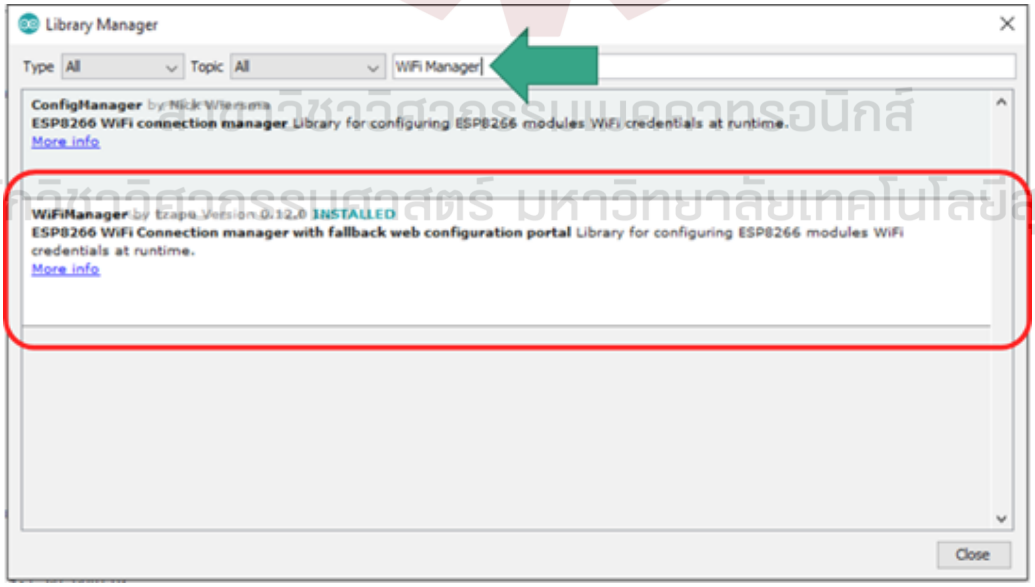


สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร มหาวิทยาลัยเทคโนโลยีสุรนารี

2.2 เพิ่มไลบรารี่ WiFi Manager ใน Arduiono IDE โดยไปที่เมนู Sketch -> Include Library -> Manage Libraries



2.3 พิมพ์ WiFi Manager ในช่องค้นหาดังรูปข้างล่าง เมื่อผลการค้นปรากฏ ให้คลิกที่ WiFiManager จากนั้นคลิก Install.



สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

2.4 เขียนโค้ดลงในโปรแกรมดังนี้

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Servo.h>
Servo servo;

char auth[] = "0cf65e892af14aa5afb01acbbccfa415";
char ssid[] = "iPhone"; //Enter your WIFI Name
char pass[] = "Tangmo3108";//Enter your WIFI Password

const int Relay8  = D0;
const int ledRed  = D2;
const int Relay1  = D3;

const int Relay3  = D5;
const int Relay4  = D6;
const int Relay5  = D7;
const int Relay6  = D8;
const int Relay7  = D10;

const int button  = D1;
bool buttonStatus = 0;
bool buttonTemp = 0;

WidgetLED ledBlynk(V10);
```

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

```
BlynkTimer timer;

BLYNK_CONNECTED() {
  Blynk.syncAll();
}

void checkStatusButton()
{
  buttonStatus = digitalRead(button);

  if (buttonStatus != buttonTemp)
  {
    if (buttonStatus == HIGH) {
      digitalWrite(ledRed, HIGH);
      Serial.println("LED Turned ON");
      Blynk.virtualWrite(V11, 1);
      ledBlynk.on();
    }
    else {
      digitalWrite(ledRed, LOW);
      Serial.println("LED Turned OFF");
      Blynk.virtualWrite(V11, 0);
      ledBlynk.off();
    }
    buttonTemp = buttonStatus;
  }
}
```

สำนักวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

```
BLYNK_WRITE(V11)
{
 if (param.asInt() == 0)
 {
  digitalWrite(ledRed, LOW);
  ledBlynk.off();
 }
 if (param.asInt() == 1)
 {
  digitalWrite(ledRed, HIGH);
  ledBlynk.on();
 }
}

BLYNK_WRITE(V3)
{
 servo.write(param.asInt());
}
void setup()
{
 Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);

 // Debug console
 pinMode(ledRed,OUTPUT);
 pinMode(D0,OUTPUT);
 pinMode(D3,OUTPUT);
```

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

```
pinMode(D5,OUTPUT);
pinMode(D6,OUTPUT);
pinMode(D7,OUTPUT);
pinMode(D9,OUTPUT);
pinMode(D10,OUTPUT);



  //set default status
digitalWrite(ledRed,LOW);
digitalWrite(D0,HIGH);
digitalWrite(D3,HIGH);

digitalWrite(D5,HIGH);
digitalWrite(D6,HIGH);
digitalWrite(D7,HIGH);
digitalWrite(D9,HIGH);
digitalWrite(D10,HIGH);
 //button
pinMode(button, INPUT);


//start blynk
Blynk.begin(auth, ssid, pass);


  //set servo
servo.attach(2);
}
```

สำนักวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

```
void loop() {


  Blynk.run();
}
```

คำอธิบายโปรแกรม

การกำหนด #include เป็นส่วนการประกาศเรียกใช้ไลบรารี่ WiFi ของ ESP8266 (ESP8266WiFi.h), ไลบรารี่ที่เกี่ยวข้องกับการจัดการ DNS Server (DNSServer.h) , ไลบรารี่ที่เกี่ยวข้องกับการจัดการ Web Server บน ESP8266 (ESP8266WebServer.h) และไลบรารี่ที่เกี่ยวข้องกับการจัดการ WiFi เพื่อให้ ESP8266 ส่งสัญญาณเป็น Access point (WiFiManager.h)

การกำหนด #define เริ่มจากการกำหนดให้ขา 16 ของ ESP8266 เป็นขา D0 บน NodeMCU และกำหนดให้เรียกขา D0 ของ NodeMCU ซึ่งต่อกับ LED บนบอร์ดว่า ledPin ต่อมากำหนดให้ขา 5 ของ ESP8266 เป็นขา D1 แล้วให้เรียกขา D1 ว่า ConfigWiFi_Pin เนื่องจากเราใช้ขานี้ต่อกับสวิตช์กดเลือกโหมดในข้อที่ 2.1 ในโหมด SmartConfig ซึ่งบอร์ดทำงานเป็น WiFi Access Point จึงกำหนดชื่อให้เป็น ESP8266 Config AP

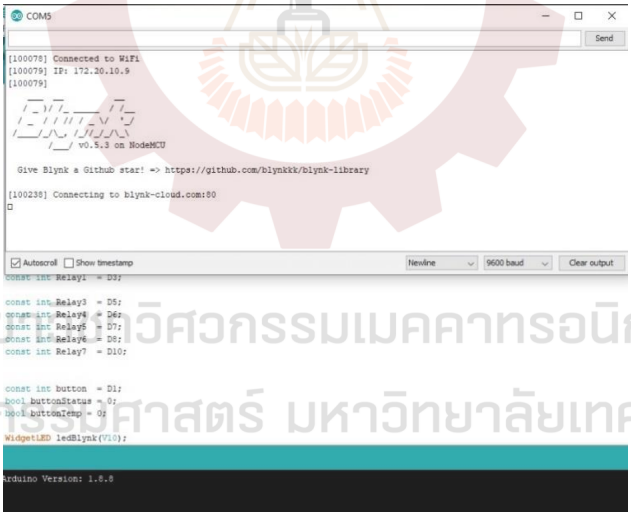
สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

ส่วนภายในฟังก์ชั่น setup() กำหนดให้ขา ledPin ทำงานในโหมด Output และขา ConfigWiFi_Pin ทำงานในโหมด Input Pullup ตั้งค่าความเร็วของพอร์ต Serial ที่ 115,200 บิตต่อวินาที และสั่งให้เขียนลอจิก Low ไปยังขา ledPin เพื่อเปิดหลอด LED ซึ่งเป็นตัวบ่งชี้ว่าบอร์ดกำลังอยู่ในช่วงตั้งค่าและติดต่อเครือข่าย WiFi จากนั้นประกาศ Object WiFiManager เพื่อเริ่มต้นเรียกใช้ไลบรารี่ WiFi Manager แล้วอ่านค่าลอจิกจากขา ConfigWiFi_Pin หากมีการกด (ลอจิก Low) ให้รีเซ็ตค่าเครือข่ายที่เคยบันทึกไว้ เป็นการบังคับให้บอร์ดเข้าสู่โหมดการทำงานแบบ SmartConfig หากไม่มีการกดปุ่มบอร์ดจะเริ่มต้นจากโหมด Station ตามปกติตามลำดับขั้นตอนการทำงานที่บรรยายไว้ในตอนต้น หลังจากตั้งค่าและบันทึกเรียบร้อย บอร์ดจะพยายามเชื่อมต่อ จึงมีคำสั่งวนลูปเพื่อตรวจสอบสถานะการเชื่อมต่อด้วยคำสั่ง WiFi.status หากยังไม่เชื่อมต่อให้หน่วงเวลา 250 ms และแสดงจุดออกมาที่หน้าจอคอมพิวเตอร์ หากมีการเชื่อมต่อเรียบร้อยแล้ว ให้แสดงคำว่า WiFi connected และแสดงหมายเลข IP Address ของ NodeMCU/ESP82266 ที่ได้รับมา โดยคำสั่ง WiFi.localIP และ

ส่งลอจิก High ไปยังขา ledPin เพื่อปิดหลอด LED เป็นการบ่งชี้ว่าขั้นตอนการเชื่อมต่อสิ้นสุดและสำเร็จ

ส่วนภายในฟังก์ชั่น loop() เริ่มจากส่งลอจิก HIGH ไปยังขา ledPin เพื่อปิดหลอดไฟ LED แล้วหน่วงเวลา 2 วินาที จากนั้นส่งลอจิก LOW ไปยังขา ledPin เพื่อเปิดหลอดไฟ LED แล้วหน่วงเวลาอีก 2 วินาที จากนั้นกลับไปยังคำสั่งแรกในฟังก์ชั่น วนลูปซ้ำเช่นนี้ไปเรื่อยๆ

2.5 คลิกปุ่ม เพื่อคอมไพล์โค้ด แล้วทำการบันทึกไฟล์เป็น “Relay” ไว้บน Desktop (ดังภาพตัวอย่างด้านล่าง เมื่อคอมไพล์เสร็จสิ้นและไม่มีข้อผิดพลาด ทำการอัพโหลดโดยคลิกปุ่ม

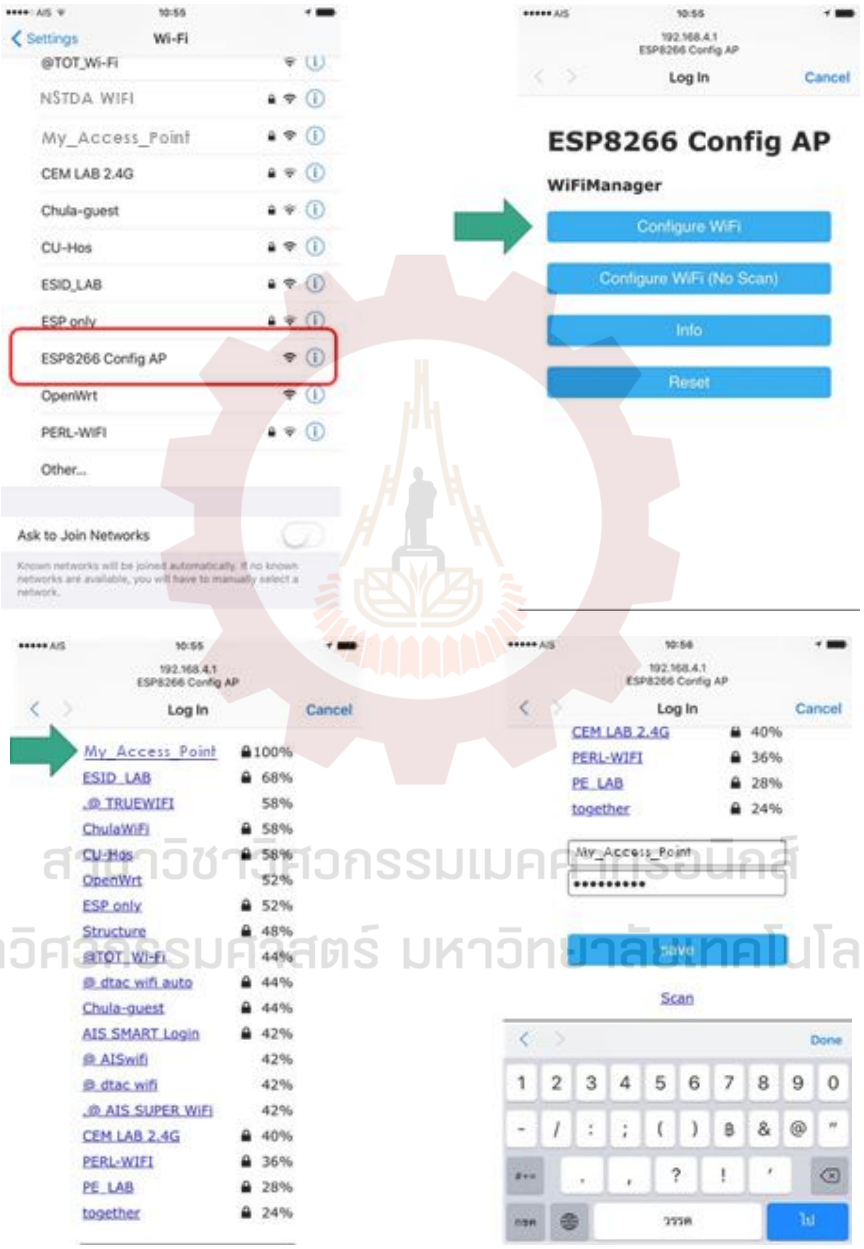


สาขาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

2.6 ถอดสาย USB ออก แล้วกดสวิตช์ที่ต่อกับขา D1 ค้างไว้ และเสียบสาย USB กลับเข้าไปใหม่อีกครั้ง จะเห็นหลอดไฟ LED บนบอร์ดติดค้าง (หลังจากนี้สามารถปล่อยปุ่มสวิตช์ได้)

2.7 นำ Smart phone มาตั้งค่าเครือข่าย WiFi โดยเลือกเครือข่ายชื่อ “ESP8266 Config AP” เพื่อเชื่อมต่อกับบอร์ด NodeMCU/ESP8266

2.8 เปิดเว็บเบราว์เซอร์บน Smart Phone ในหน้า Portal ให้เลือกเครือข่าย WiFi ที่ต้องการให้บอร์ด NodeMCU/ESP8266 เชื่อมต่อด้วย โดยในรูปตัวอย่าง เลือกเครือข่ายที่มี SSID ชื่อ “Anoney Potter WiFi” จากนั้นกดปุ่ม Save บอร์ด NodeMCU/ESP8266



2.9 บอร์ด NodeMCU/ESP8266 จะ Reboot ตัวเองและเชื่อมต่อกับเครือข่าย WiFi ที่ผู้ศึกษาเลือก เมื่อเชื่อมต่อสำเร็จ หลอดไฟ LED จะดับไปชั่วครู่ และกลับมากระพริบ

ที่มาhttps://netpie.gitbooks.io/nodemcu-esp8266-on-netpie/content/lab-6.html

สำนักวิชาวิศวกรรมเมคคาทรอนิคส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

โปรแกรมที่ใช้ควบคุมระบบตรวจสอบสถานะภายในบ้าน

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

โปรแกรมที่ใช้ควบคุมระบบตรวจสอบสถานะภายในบ้าน
เขียนได้ดังนี้

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Servo.h>
Servo servo;

char auth[] = "0cf65e892af14aa5afb01acbbccfa415";
char ssid[] = "iPhone"; //Enter your WIFI Name
char pass[] = "Tangmo3108";//Enter your WIFI Password

const int Relay8  = D0;
const int ledRed  = D2;
const int Relay1  = D3;
const int Relay3  = D5;
const int Relay4  = D6;
const int Relay5  = D7;
const int Relay6  = D8;
const int Relay7  = D10;

const int button  = D1;
bool buttonStatus = 0;
bool buttonTemp = 0;

WidgetLED ledBlynk(V10);
```

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

```
BlynkTimer timer;


BLYNK_CONNECTED() {
  Blynk.syncAll();
}


void checkStatusButton()
{
  buttonStatus = digitalRead(button);


  if (buttonStatus != buttonTemp)
  {
    if (buttonStatus == HIGH) {
      digitalWrite(ledRed, HIGH);
      Serial.println("LED Turned ON");
      Blynk.virtualWrite(V11, 1);
      ledBlynk.on();
    }
    else {
      digitalWrite(ledRed, LOW);
      Serial.println("LED Turned OFF");
      Blynk.virtualWrite(V11, 0);
      ledBlynk.off();
    }
    buttonTemp = buttonStatus;
  }
}
```

สำนักวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

```
BLYNK_WRITE(V11)
{
 if (param.asInt() == 0)
 {
  digitalWrite(ledRed, LOW);
  ledBlynk.off();
 }
 if (param.asInt() == 1)
 {
  digitalWrite(ledRed, HIGH);
  ledBlynk.on();
 }
}

BLYNK_WRITE(V3)
{
 servo.write(param.asInt());
}
void setup()
{
 Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);

 // Debug console
 pinMode(ledRed,OUTPUT);
 pinMode(D0,OUTPUT);
 pinMode(D3,OUTPUT);
```

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

```
 pinMode(D5,OUTPUT);
 pinMode(D6,OUTPUT);
 pinMode(D7,OUTPUT);
 pinMode(D9,OUTPUT);
 pinMode(D10,OUTPUT);

   //set default status
 digitalWrite(ledRed,LOW);
 digitalWrite(D0,HIGH);
 digitalWrite(D3,HIGH);
 digitalWrite(D5,HIGH);
 digitalWrite(D6,HIGH);
 digitalWrite(D7,HIGH);
 digitalWrite(D9,HIGH);
 digitalWrite(D10,HIGH);

  //button
 pinMode(button, INPUT);
 //start blynk
 Blynk.begin(auth, ssid, pass);

   //set servo
 servo.attach(2);
}
void loop() {

 Blynk.run();
}
```

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

คู่มือการเปิดใช้งาน Blynk

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

SMART HOME

คู่มือการใช้งาน
ระบบตรวจสอบสถานะภายในบ้าน

Blynk

เริ่มต้นใช้งาน Blynk

การเชื่อมต่อเข้ากับ Blynk server สามารถใช้ได้ทั้ง WiFi และเครือข่ายมือถือ
โดยสามารถ Download application นี้ได้ฟรีทั้งระบบ IOS และ Android

1. ติดตั้งแอฟ Blynk ลงในสมาร์ตโฟน

Android OS

IOS

2. ลงทะเบียนใช้งาน Blynk จาก APP ที่ติดตั้งในมือถือ เปิด Blynk APP ขึ้นมา
แล้วคลิกที่ "Create New Account"แล้วใส่อีเมล์และรหัสผ่านที่ต้องการ
โดยอีเมล์ที่กรอก "ต้องใช้งานได้จริงๆ" เพราะระบบจะส่งรหัส TOKEN ไปให้ตามอีเมล์

สร้างผู้ใช้งานใหม่

ระบุที่อยู่อีเมล์และรหัสผ่าน

เลือกตั้งค่า Server

ระบุ IP Server และพอร์ต

สร้างโปรเจคใหม่

สร้างโปรเจค

THEME ที่ดีไซน์ขึ้นของ APP
ถ้าใครชอบสีแนวออกสว่างสีขาวก็สามารถเลือกเป็น "LIGHT"
ส่วนผู้เขียนชอบพื้นเลือกโทนสีดำจึงเลือกเป็น "DARK" จากนั้นคลิกที่ปุ่ม

SMART HOME

คู่มือการใช้งาน
ระบบตรวจสอบสถานะภายในบ้าน

สถาบันวิชาวิศวกรรมเมคคาทรอนิคส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

Blynk

เริ่มต้นใช้งาน APP

Create New account เพื่อสร้างการเชื่อมต่อกับ application กับ Email

connect เข้ากับ server ของ Blynk ของเราเองได้ โดยเลือก Custom
และใส่ IP Address ของ Server เราเอง โดยกดที่รูป Problems Signing In
แล้วเลื่อน scroll จาก Blynk ไป Custom

Hardware ให้เลือก ESP8266

AUTH TOKEN

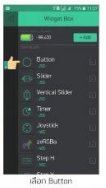
ทุก ๆ ครั้งที่เริ่มสร้างโปรเจคใหม่
AUTH TOKEN จะถูกเปลี่ยนใหม่เสมอ

สร้าง project ของเราด้วย Blynk ให้กดที่สัญลักษณ์หกเหลี่ยมมุมขวาบนเพื่อตั้งค่า

สร้าง WIDGET ปุ่มควบคุมขึ้นมา ให้คลิ๊กที่พื้นที่ว่างในโปรเจคตามรูป
แล้วคลิ๊กที่สัญลักษณ์รูป +

คลิ๊กพื้นที่ว่าง

คลิ๊กเครื่องหมายบวก


เลือก Button

คลิ๊กที่ปุ่ม OFF

เลือก Button (200 energy) แล้วคลิ๊กที่ปุ่มเพื่อไปหน้าตั้งค่า

ตั้งค่า Output เป็น GP4

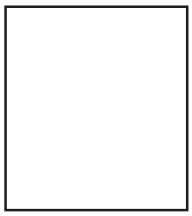
ตั้งค่า Output เป็น GP4

เริ่มใช้งาน

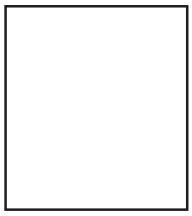
ควบคุมการ ปิด เปิด

เริ่มการทำงานของ APP กัน ให้คลิ๊กที่ปุ่มสามเหลี่ยม
ท่านจะเห็นได้ว่ารูปสามเหลี่ยมจะเปลี่ยนเป็นรูปสี่เหลี่ยมแทน แล้วที่ปุ่ม
Button จะเปลี่ยนจากคำว่า gp4 เป็น OFF แทน ท่านลองกดปุ่มดู

ข้อมูลผู้จัดทำ



ชื่อ-สกุล อายุ ปี
ที่อยู่
เบอร์โทร
อีเมลล์



ชื่อ-สกุล อายุ ปี
ที่อยู่
เบอร์โทร
อีเมลล์

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

ชื่อ-สกุล อายุ ปี
ที่อยู่
เบอร์โทร
อีเมลล์

บรรณานุกรม

Ryu, K., and Oh, Y. (2011). Balance Control ofBall-Beam System Using Redundant Manipulator,paper presented in *IEEE International Conference on Mechatronics 2011*, Istanbul, Turkey.Nganga-Kouya, D. and Okou, F.A. (2011). A New Adaptive State Feedback Controller for theBall and Beam System, presented in *Canadian Conference Electrical and Computer Engineering*2011, Niagara Falls, Canada.

Dynamic Studies**. IEEE Transactions on Industrial Applications**, Vol. 51, No. 1, January 2015.

Dao Thanh Liem, Dinh Quang Thuong and Kyoung Kwan Ahn (2015). A torque estimator using online tuning grey fuzzy PID for applications to torque-sensor less control of DC motors. **Elsevier Ltd**. Mechatronics 26. February 2015

Osman Kalender and Yavuz Ege (2007). A PIC Microcontroller Based Electromagnetic Stirrer. **IEEE Transactions on Magnetics**, Vol. 43, No. 9, September 2007

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี