รายงานสหกิจศึกษาฉบับสมบูรณ์

ชุดตรวจสอบและรับส่งค่า รถขนส่งของอัตโนมัติ

Detect and transfer value for Fast automatic mobile robot

นายธรรมรัตน์ ปงรังษี

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

T148491

รายงานสหกิจศึกษาฉบับสมบูรณ์

ชุดตรวจสอบและรับส่งค่า รถขนส่งของอัตโนมัติ
Detect and transfer value for Fast automatic mobile robot

นายธรรมรัตน์ ปงรังษี

สหม.
เลขทะเบียน 148491
วัน.เดือน.ปี 30 มิ.ย. 2560

.b 12871072
.i

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานสหกิจศึกษานี้จะสำเร็จสมบูรณ์ตามเป้าหมายเพราะได้รับคำแนะนำและความช่วยเหลือจากอาจารย์ที่ปรึกษาของโครงการสหกิจ อาจารย์ที่ปรึกษาของทางภาควิชาวิศวกรรมอิเล็กทรอนิกส์ และทีมวิศวกรของทางบริษัท เดลต้า อิเลคทรอนิคส์(ประเทศไทย) จำกัด(มหาชน) แผนก AMT ที่คอยช่วยเหลือ ให้คำปรึกษาในการแก้ไขปัญหาต่างๆที่เกิดขึ้นในระหว่างการทำโครงงาน นอกจากนี้ทางบริษัทยังให้การดูแลนักศึกษาอย่างดี คอยแนะนำแนวทางการทำงาน มอบความรู้และประสบการณ์ ที่จะประยุกต์ใช้ในชีวิตและการทำงานในอนาคต โดยเป็นระยะเวลาในการปฏิบัติงานสหกิจทั้งหมด 4 เดือน ซึ่งถ้าขาดบุคคลเหล่านี้โครงงานสหกิจศึกษานี้มิอาจสำเร็จได้ จึงขอขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

นายธรรมรัตน์ ปงรังษี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงงานสหกิจศึกษา ชุดตรวจสอบและรับส่งค่า รถขนส่งของอัตโนมัติ
ชื่อ-สกุล นักศึกษา นายธรรมรัตน์ ปงรังษี
คณะ วิศวกรรมศาสตร์ **ภาควิชา** วิศวกรรมอิเล็กทรอนิกส์
ชื่อ-สกุล อาจารย์นิเทศ อาจารย์ เฉลิมพันธ์ หวังวิวัฒนา และ อาจารย์ชินภัทร นันทจิวากรชัย
ชื่อ-สกุล ผู้นิเทศงาน นาย สรายุทธ แช่มชื่น
ชื่อสถานประกอบการ บริษัท เดลต้า อีเลคทรอนิคส์(ประเทศไทย) จำกัด(มหาชน)

บทคัดย่อ

บริษัท เดลต้า อีเลคทรอนิคส์(ประเทศไทย) จำกัด(มหาชน) เป็นบริษัทรับจ้างผลิต Power Supply และ ROBOT technology บริษัทอุตสาหกรรมต่างๆ ซึ่งเป็นอุตสาหกรรมที่มีการเจริญเติบโตทางด้านธุรกิจเป็นอย่างมากในขณะนี้ ทางกลุ่มบริษัทเหล่านั้น ต้องการที่จะให้ผลิตภัณฑ์ของทางบริษัท เดลต้า มีมาตรฐาน และความน่าเชื่อถือมากยิ่งขึ้น บริษัทเดลต้าจึงนำ หุ่นยนต์รูปแบบต่างๆ มาใช้งานในการผลิต และการขนส่ง เพื่อยกระดับมาตรฐานที่มีอยู่ให้สูงยิ่งขึ้นและเป็นการลดต้นทุนในการผลิตเพื่อให้ลูกค้าพอใจยิ่งขึ้น

จากที่กล่าวมานั้น หุ่นยนต์ส่งของที่มีอยู่ต้องติดตั้งอุปกรณ์ในการนำทางเพิ่ม คือ แทบแม่เหล็ก ซึ่งสิ่งนี้มีความนูน จึงทำให้เกิดอุบัติเหตุของพนักงานในบริษัท ทำให้ของเสียหาย จึงได้นำอุปกรณ์นำทางเสริมคือ Gyro sensor ควบคู่กับโปรแกรม PLC มาใช้ในการนำทางแทน ซึ่งจะมีการตรวจสอบเป็นระยะด้วยจุดสี และคำนวณระยะห่างระหว่างสิ่งรอบข้างด้วย Ultrasonic sensor

คำสำคัญ : แทบแม่เหล็ก การนำทางด้วย PLC, Gyro sensor, RGB sensor และ Ultrasonic sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Research Title: Detect and Transfer Value for Fast automatic mobile robot
Student intern name: Mr.THAMMARAT PONGRANGSI
Faculty: Engineering **Department:** Electronics engineering
Advisor name: Mr. Chaloemphan Wangwiwattana and Mr.Chinnapat Nantajiwakornchai
Mentor name: Mr. Sarayut Chaemchuen
Company: DELTA ELECTRONICS (THAILAND) PCL.

ABSTRACT

Delta Electronics (Thailand) PCL is manufacturer of power supplies and creating a robot technology. Customers of Delta want to a high standard products of Delta. So that we use robot in manufacturing and transportation, to raise standards and reduce coasts to satisfy therefore need.

From the above robotic vehicles on Delta are required a bar of magnetic for navigation which has bulging is a cause of accident and damage on products. Then we changed a bar of magnetic to PLC, Gyro sensor which would be reviewed periodically by color and spacing.

Keywords: magnetic bar, PLC, Gyro sensor, RGB sensor, Ultrasonic sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

กิตติกรรมประกาศ I

บทคัดย่อ II

ABSTRACT III

สารบัญ IV

สารบัญตาราง VI

สารบัญภาพ VII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญ 1

1.2 วัตถุประสงค์ 1

1.3 ขอบเขตการวิจัย 1

1.4 วิธีการดำเนินงาน 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 2

บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 3

2.1 Arduino UNO R3 3

2.3 RGB color sensor 5

2.4 Ultrasonic sensor 7

2.5 TTL และ RS232 11

2.6 ESP8266-01 Wireless module 13

2.7 TCP (transmission control protocol) 15

2.8 UDP (User Datagram Protocol) 16

2.9 Visual C# 16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 วิธีดำเนินการ 20

3.1 วิธีดำเนินการทำโครงงาน 20

3.2 ตรวจสอบตำแหน่งที่ติดตั้งเซนเซอร์ 21

3.3 การเก็บค่าเซนเซอร์ 22

3.4 คำสั่งการส่งข้อมูลผ่าน Wireless module 24

3.5 ออกแบบ Schematic และ PCB 25

บทที่ 4 การทดลองและผลการทดลอง 27

4.1 รายละเอียดการดำเนินการทดลอง 27

4.2 ผลการทดลอง 30

4.3 ทดสอบและเก็บผลการทดลอง 37

บทที่ 5 สรุปผลการดำเนินงาน และข้อเสนอแนะ 44

เอกสารอ้างอิง 46

ภาคผนวก 47

ประวัติผู้เขียน 69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 ตารางแสดงข้อมูลจำเพาะของบอร์ด Arduino Uno....4
ตารางที่ 4.1 ตารางแสดงผลระยะ Ultrasonic Sensor....30
ตารางที่ 4.2 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 1....31
ตารางที่ 4.3 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 2....31
ตารางที่ 4.4 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 3....31
ตารางที่ 4.5 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 4 31
ตารางที่ 4.6 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 5....31
ตารางที่ 4.7 แสดงผลจากความสัมพันธ์ระหว่างระยะที่วัดได้กับอุณหภูมิของ Ultrasonic Sensor 36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

หน้า

ภาพที่ 2.1 บอร์ด Arduino Uno....2
ภาพที่ 2.2 Color detect....5
ภาพที่ 2.3 TCS230 IC....5
ภาพที่ 2.4 ตำแหน่งขา TCS230 IC....5
ภาพที่ 2.5 การทำงานของ TCS230 IC....6
ภาพที่ 2.6 การอ่านค่าสีของ TCS230 IC....6
ภาพที่ 2.7 การควบคุมขนาดสัญญาณของ TCS230 IC....6
ภาพที่ 2.8 อัลตร้าโซนิคเซ็นเซอร์ วงจรส่งผ่าน/รับ....8
ภาพที่ 2.9 อัลตร้าโซนิคเซ็นเซอร์ , วงจรเวลาคงที่....9
ภาพที่ 2.10 คลื่นอัลตร้าโซนิคที่อุณหภูมิต่างกัน....9
ภาพที่ 2.11 แสดงการสะท้อนของเสียง....10
ภาพที่ 2.12 Distance detect....10
ภาพที่ 2.13 สัญญาณการส่งคลื่นเสียงของ Ultrasonic sensor....11
ภาพที่ 2.14 ระดับของสัญญาณชนิด TTL....11
ภาพที่ 2.15 ระดับของสัญญาณชนิด RS232....12
ภาพที่ 2.16 การใช้งาน MAX232 CPE IC....12
ภาพที่ 2.17 การต่อการใช้งาน MAX232 CPE IC....13
ภาพที่ 2.18 ESP8266-01....13
ภาพที่ 2.19 ตัวอย่างการใช้งาน ESP8266-01....14
ภาพที่ 2.20 การส่งข้อมูลแบบ TCP....15
ภาพที่ 2.21 การส่งข้อมูลแบบ UDP....16
ภาพที่ 2.22 โปรแกรม Visual Studio 2013....17
ภาพที่ 2.23 การสร้างโปรเจค Visual studio....18
ภาพที่ 2.24 Form Program....18
ภาพที่ 2.25 ส่วนประกอบโปรแกรม....19
ภาพที่ 3.1 รถ AVG....21
ภาพที่ 3.2 ตำแหน่ง RGB Sensor....21
ภาพที่ 3.3 ตำแหน่ง Ultrasonic Sensor ซ้ายและขวา....22
ภาพที่ 3.4 คำสั่งอ่านค่าเซ็นเซอร์....22
ภาพที่ 3.5 คำสั่งตรวจสอบสี....23
ภาพที่ 3.6 คำสั่งส่งข้อมูลผ่าน Wireless....24
ภาพที่ 3.7 Schematic hardware design....25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 3.8 PCB hardware design 26
ภาพที่ 4.1 บอร์ด Arduino Uno R3 and ESP8266-01 27
ภาพที่ 4.2 RGB Sensor 27
ภาพที่ 4.3 Ultrasonic Sensor 28
ภาพที่ 4.4 TT Program 28
ภาพที่ 4.5 SP8266-01 ที่ IP Address 192.168.172.111 32
ภาพที่ 4.6 ESP8266-01 ที่ IP Address 192.168.172.100 32
ภาพที่ 4.7 รูปแสดง Flow Chart การทำงาน ESP8266-01 33
ภาพที่ 4.8 รูปแสดง Flow Chart การทำงาน Ultrasonic Sensor 34
ภาพที่ 4.9 รูปแสดง Flow Chart การทำงาน RGB Sensor 35
ภาพที่ 4.10 ผลการวัดสีฟ้า และระยะซ้ายขวาที่ 15 cm มีไฟส่องที่วัตถุ 37
ภาพที่ 4.11 ผลการวัดสีแดง และระยะซ้ายขวาที่ 30 cm ไม่มีไฟส่องที่วัตถุ 38
ภาพที่ 4.12 ผลการวัดสีเขียว และระยะซ้ายขวาที่ 100 cm ไม่มีไฟส่องที่วัตถุ 39
ภาพที่ 4.13 IP Address : 192.168.172.150 AP : Delta-Guest ส่งข้อความจาก TT Program 40
ภาพที่ 4.14 IP Address : 192.168.172.150 AP : Delta-Guest ส่งข้อความจาก Monitor IDE 41
ภาพที่ 4.15 IP Address : 192.168.172.200 AP : Delta-Guest ส่งข้อความจาก TT Program 42
ภาพที่ 4.16 IP Address : 192.168.172. AP : Delta-Guest ส่งข้อความจาก Monitor 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

บริษัท เดลต้า อีเลคทรอนิคส์(ประเทศไทย) จำกัด(มหาชน) เป็นบริษัทรับจ้างผลิต Power Supply และ ROBOT technology ต่างๆ โดยในกระบวนการผลิตมีการขนส่งมากมาย เช่นการส่งอุปกรณ์อิเล็กทรอนิกส์ การส่งเครื่องมือ หรือการส่งเอกสารต่างๆ เพื่อลดความเสียหายต่ออุปกรณ์ และลดเวลาการเคลื่อนย้าย จึงมี หุ่นยนต์ในการขนส่งสิ่งเหล่านี้แทนการเคลื่อนย้ายโดยคนที่ก่อให้เกิดความเสียหายมากมาย

ซึ่งในปัจจุบันทางแผนก AMT ทำรถหุ่นยนต์ขึ้นโดยไม่ใช้แถบสี และแถบแม่เหล็กในการนำทางแต่ใช้แผนที่จำลองแทนซึ่งจะช่วยลดปัญหาอุบัติเหตุที่จะเกิดจากรอยนูนของแถบแม่เหล็ก และการเปลี่ยนสีของแถบสี เพราะสิ่งเหล่านี้จะก่อให้เกิดปัญหาเช่นพนักงานสะดุดล้มก่อให้เกิดความเสียหาย และลดต้นทุนในการติดตั้งแถบแม่เหล็กและแถบสี

1.2 วัตถุประสงค์

1.2.1 เพื่อลดต้นทุนการติดตั้งอุปกรณ์ในการใช้นำทาง

1.2.2 เพื่อเพิ่มความสามารถในการหลบสิ่งกีดขว้าง

1.2.3 เพื่อลดอุบัติเหตุจากอุปกรณ์นำทาง

1.3 ขอบเขตการวิจัย

1.3.1 สามารถตรวจสอบสิ่งกีดขวางที่อยู่รอบตัวได้

1.3.2 มีการตรวจสอบความถูกต้องในการเคลื่อนย้าย

1.3.3 สามารถวัดระยะห่างระหว่างเซนเซอร์กับวัตถุได้ในช่วง 3 เซนติเมตรถึง 300 เซนติเมตร

1.3.4 สามารถตรวจสอบสีได้ 3 สี คือ เขียว แดง น้ำเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 วิธีการดำเนินงาน

ขั้นตอนการดำเนินงานแบ่งเป็น 5 ขั้นตอน

1.4.1 ทดสอบระยะการวัดของ Ultrasonic sensor

1.4.2 ทดสอบระยะการตรวจสีของ RGB Sensor

1.4.3 เชื่อมต่อ Arduino ESP8266-01 กับโปรแกรม TT ผ่าน ไวไฟ

1.4.4 สร้างแผนผังโค้ด และเขียน โค้ดบนบอร์ด Arduino ESP8266-01

1.4.5 Test และ Calibration

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 จะสามารถลดจำนวนความเสียหายของชิ้นงานที่เกิดจากสิ่งกีดขวาง

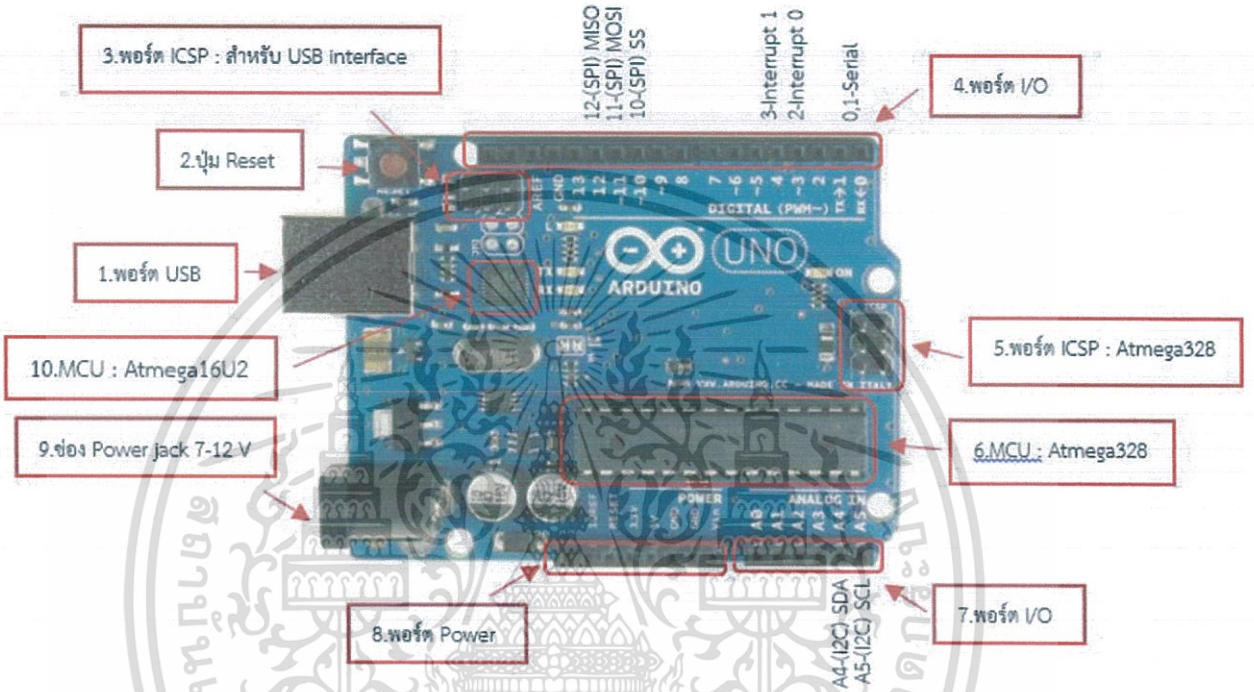
1.5.2 รู้จักการสังเกต ข้อมูลผลการทดสอบ เพื่อวิเคราะห์สาเหตุของปัญหาที่เกิดขึ้น ในแต่ละ Test และแก้ไขปัญหานั้น

1.5.3 ฝึกการเขียนโปรแกรม Arduino และ Visual C#

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 Arduino UNO R3



ภาพที่ 2.1 บอร์ด Arduino Uno

Arduino Uno R3 เป็นบอร์ด Arduino ที่ได้รับความนิยมมากที่สุด เนื่องจากราคาไม่แพง ส่วนใหญ่โปรเจคและ Library ต่างๆ ที่พัฒนาขึ้นมา Support จะอ้างอิงกับบอร์ดนี้เป็นหลัก เนื่องจากเป็นขนาดที่เหมาะสำหรับการเริ่มต้นเรียนรู้ Arduino และมี Shields ให้เลือกใช้งานได้มากกว่าบอร์ด Arduino รุ่นอื่นๆที่ออกแบบมาเฉพาะมากกว่า โดยบอร์ด Arduino Uno ได้มีการพัฒนาเรื่อยมา ตั้งแต่ R2 R3 และรุ่นย่อยที่เปลี่ยนชิปไอซีเป็นแบบ SMD และข้อดีอีกอย่างคือ กรณีที่ MCU เสีย ผู้ใช้งานสามารถซื้อมาเปลี่ยนเองได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจำเพาะ

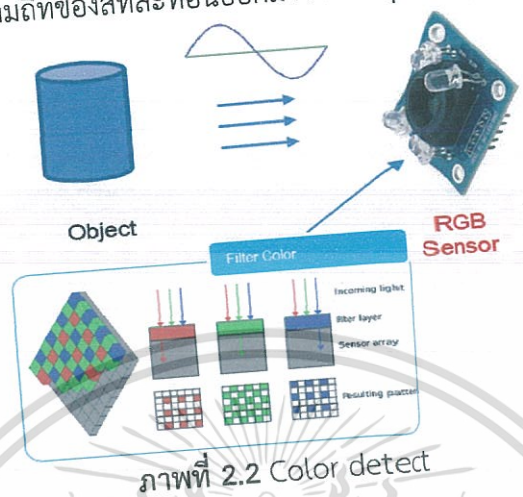
ตารางที่ 2.1 ตารางแสดงข้อมูลจำเพาะของบอร์ด Arduino Uno R3

ชิปไอซีไมโครคอนโทรเลอร์	ATmega328
ใช้แรงดันไฟฟ้า	5V
รองรับการจ่ายแรงดันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงดันไฟฟ้า (ที่จำกัด)	6 – 20V
พอร์ต Digital I/O	14 พอร์ต (มี 6 พอร์ต PWM output)
พอร์ต Analog Input	6 พอร์ต
กระแสไฟที่จ่ายได้ในแต่ละพอร์ต	40mA
กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	50mA
พื้นที่โปรแกรมภายใน	32KB พื้นที่โปรแกรม, 500B ใช้โดย Bootloader
พื้นที่แรม	2KB
พื้นที่หน่วยความจำถาวร (EEPROM)	1KB
ความถี่คริสตัล	16MHz
ขนาด	68.6x53.4 mm
น้ำหนัก	25 กรัม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 RGB color sensor

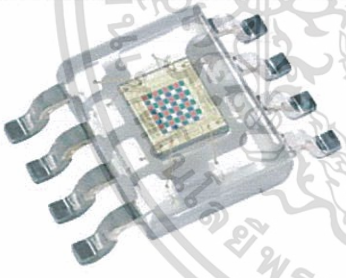
โมดูลวัดค่าสี RGB color sensor คือ เซนเซอร์แยกสีของวัตถุ หรือสิ่งของต่างๆ โดยใช้ชิป TCS230 ในการแยกสีตามความถี่ที่ของสีที่สะท้อนออกมาจากวัตถุที่วัด



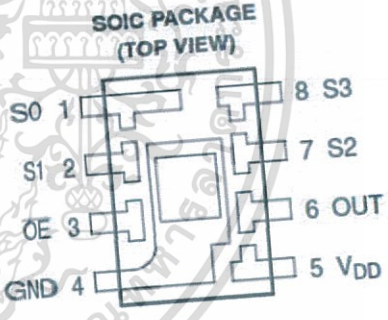
ภาพที่ 2.2 Color detect

2.2.1 TCS230

2.2.1.1 TCS230 IC



ภาพที่ 2.3 TCS230 IC

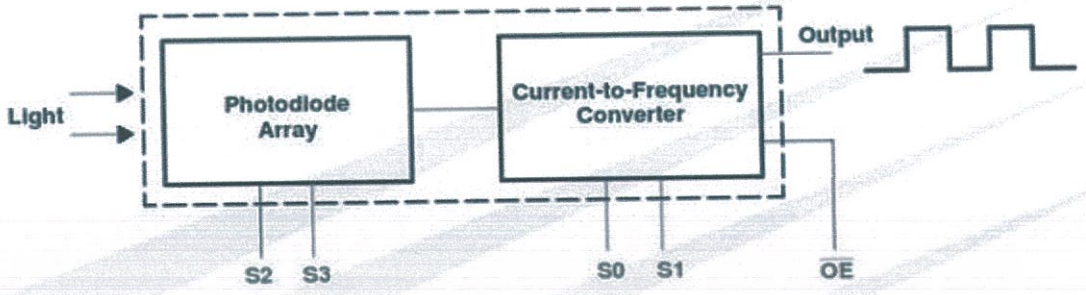


ภาพที่ 2.4 ตำแหน่งขา TCS230 IC

TCS230 IC เป็นเซนเซอร์ตรวจวัดสี โดยการแยกความถี่ของแสงที่สะท้อนมากจากวัตถุ โดยใช้ photodiodes โดย TCS230 IC สามารถแปลงค่าสีแดง สีเขียว สีฟ้า มาเป็นความถี่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.2 หลักการทำงานของ TCS230 IC



ภาพที่ 2.5 การทำงานของ TCS230 IC

- ใน IC จะมี photodiodes ขนาด 8x8 ตัว โดยมีสีน้ำเงิน 16 ตัว สีเขียว 16 ตัว สีแดง 16 ตัว และสีขาว 16 ตัว

- โดยจะให้สัญญาณเอาต์พุตออกมาทางขา s2 และ s3 เช่น ถ้า s2 และ s3 ให้สัญญาณ 0 หรือ L ออกมา แปลว่า อ่านได้ค่าแสงสีแดง แล้ว ถ้า s2 และ s3 ให้ค่าออกมาเป็น L และ H แสดงว่าเป็นสีน้ำเงิน

S2	S3	PHOTODIODE TYPE
L	L	Red
L	H	Blue
H	L	Clear (no filter)
H	H	Green

ภาพที่ 2.6 การอ่านค่าสีของ TCS230 IC

- เราสามรถควบคุมขนาดของสัญญาณเอาต์พุตนี้ได้โดยควบคุมที่ขา s0 และ s1 โดยมี 4 ระดับ คือ ถ้า s0 กับ s1 เป็น L L หรือ 0 0 แปลว่าปิดการทำงาน เช่นกัน ถ้า s0 กับ s1 เป็น H L แปลว่า ขยายสัญญาณที่ 20 %

S0	S1	OUTPUT FREQUENCY SCALING (f_o)
L	L	Power down
L	H	2%
H	L	20%
H	H	100%

ภาพที่ 2.7 การควบคุมขนาดสัญญาณของ TCS230 IC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 Ultrasonic sensor

Ultrasonic หมายถึง คลื่นเสียงที่มีความถี่สูงเกินกว่าที่หูมนุษย์จะได้ยิน โดยทั่วไปแล้วหูของมนุษย์โดยเฉลี่ยจะได้ยินเสียงสูงถึงเพียงแค่ประมาณ 15 KHz เท่านั้น แต่พวกที่อายุยังน้อย ๆ อาจจะได้ยินเสียงที่มีความถี่สูงกว่านี้ได้ ดังนั้นโดยปกติแล้วคำว่าอัลตร้าโซนิคจึงมักจะหมายถึงคลื่นเสียงที่มีความถี่สูงกว่า 20 KHz ขึ้นไป จะสูงขึ้นจนถึงเท่าใดไม่ได้ระบุจำกัดเอาไว้

สาเหตุที่มีการนำเอาคลื่นย่านอัลตร้าโซนิคมาใช้ก็เพราะว่าเป็นคลื่นทีมีทิศทางทำให้เราสามารถเล็งคลื่นเสียงไปยังเป้าหมายที่ต้องการได้โดยเจาะจง เรื่องนี้เป็นคุณสมบัติของคลื่นอย่างหนึ่ง ยิ่งคลื่นมีความถี่สูงขึ้นความยาวคลื่นก็จะยิ่งสั้นลง ถ้าความยาวคลื่นยาวกว่าช่องเปิด (ที่ให้เสียงนั้นออกมา)ของตัวกำเนิดเสียงความถี่นั้นเช่น คลื่นความถี่ 300 Hz ในอากาศจะมีความยาวถึงประมาณ 1 เมตรเศษ ๆ ซึ่งจะยาวกว่าช่องที่ให้คลื่นเสียงออกมาจากตัวกำเนิดเสียงโดยทั่วไปมากมายคลื่นจะหักเบนที่ขอบด้านนอกของตัวกำเนิดเสียงทำให้เกิดการกระจายทิศทางคลื่นแต่ถ้าความถี่สูงขึ้นมาอยู่ในย่านอัลตร้าโซนิค อย่างเช่น 40 KHz จะมีความยาวคลื่นในอากาศเพียงประมาณ 8 มม. เท่านั้นซึ่งเล็กกว่ารูเปิดของตัวที่ให้กำเนิดเสียงความถี่นี้มากคลื่นเสียงจะไม่มีการเลี้ยวเบนที่ขอบจึงพุ่งออกมาเป็นลำแคบ ๆ หรือที่เราเรียกว่า “มีทิศทาง”

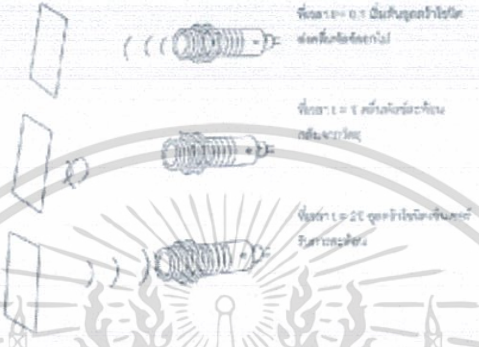
การมีทิศทางของคลื่นเสียงย่านอัลตร้าโซนิคทำให้เรานำไปใช้งานได้หลายอย่าง เช่น นำไปใช้ในเครื่องควบคุมระยะไกล (Ultrasonic remote control) เครื่องล้างอุปกรณ์ (Ultrasonic cleaner) โดยให้น้ำสั่นที่ความถี่สูง เครื่องวัดความหนาของวัตถุโดยสังเกตระยะเวลาที่คลื่นสะท้อนกลับมา เครื่องวัดความลึกและทำแผนที่ใต้ท้องทะเล ใช้ในเครื่องหาตำแหน่งอวัยวะบางส่วนในร่างกาย ใช้ทดสอบการรั่วไหลของท่อ เป็นต้น โดยความถี่ที่ใช้ขึ้นอยู่กับการใช้งาน เช่น คลื่นเสียงต้องเดินทางผ่านอากาศแล้วความถี่ที่ใช้ก็มักจะจำกัดอยู่เพียงไม่เกิน 50 KHz เพราะที่ความถี่สูงขึ้นกว่านี้อากาศจะดูดกลืนคลื่นเสียงเพิ่มขึ้นมาก ทำให้ระดับความแรงของคลื่นเสียงที่ระยะห่างออกไปลดลงอย่างรวดเร็ว ส่วนการใช้งานด้านการแพทย์ซึ่งต้องการรัศมีทำการสั้น ๆ ก็อาจใช้ความถี่ในช่วง 1 MHz ถึง 10 MHz ขณะที่ความถี่เป็น GHz (109 Hz) ก็มีใช้กันในหลายๆ การใช้งานที่ตัวกลางที่คลื่นเสียงเดินทางผ่านไม่ใช่อากาศ

Ultrasonic sensor คือ อัลตราโซนิกเซ็นเซอร์ส่งสัญญาณพัลส์ของพลังงานซึ่งเป็นการเดินทางของความเร็วเสียง การลดทอนของพลังงานที่ถูกสะท้อนกลับมาจากวัตถุเสียงนี้เป็นการสะท้อนกลับจากวัตถุแล้วเดินทางกลับไปยังเซ็นเซอร์ โดยการตรวจจับระยะเวลาที่ใช้ในการเดินทางไปกลับของเสียงเมื่อมีการตกกระทบจากวัตถุแล้วนำมาคำนวณเป็นระยะทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรส่งผ่าน / รับ

สำหรับการทำงานเป็นวงจรของอัลตร้าโซนิคเซ็นเซอร์ จะส่งผ่านคลื่นพัลซ์เสียงที่ช่วงเวลาสม่ำเสมอหรือช่วงเวลาที่เปลี่ยนแปลง คลื่นเสียงที่ปล่อยออกไปจะถูกสะท้อนได้โดยวัตถุที่เหมาะสม โดยเซ็นเซอร์และระบบการทำงานจะรับการสะท้อนของคลื่นเสียงที่สะท้อนกลับมา (ดังแสดงในรูป)ความกว้างของคลื่นพัลซ์ของเสียงอยู่ในช่วง 2.-200 ไมโครเซท

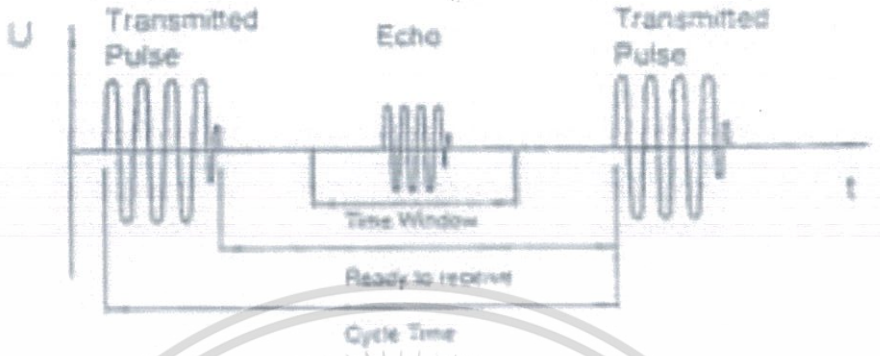


ภาพที่ 2.8 อัลตร้าโซนิคเซ็นเซอร์ วงจรส่งผ่าน/รับ

เวลาในการเดินทางของคลื่นพัลซ์ของคลื่นเสียงเป็นการวัดระยะห่างจากวัตถุ ทั้งนี้ขึ้นอยู่กับชนิดของเซ็นเซอร์ ระยะห่างนี้นำไปแสดงในรูปของสัญญาอนาล็อก (Analog Signal) (เช่น 0-20 mA) สัญญาณลอจิก (Logic Signal) (เช่น สัญญาณลอจิก 8 bit) ตลอดทั้ง ซีเรียลอินเตอร์เฟส (Serial Interface) (RS232) หรือการเปรียบเทียบกับค่าอ้างอิงในรูปของสวิสซ์พัลซ์ที่เรียกว่า ไทม์เฟรม (Time Frame) เนื่องจากขบวนการดำเนินไปตามเวลาที่คลื่นสะท้อนเดินทาง ไม่ใช่เป็นไปตามความเข้มของคลื่นสะท้อน จึงจัดได้ว่าอัลตร้าโซนิคเซ็นเซอร์ มีข้อดีเหนือกว่าเซ็นเซอร์แบบออปติคอล (Optical Sensor) เวลาที่คลื่นสะท้อนการเดินทางจะ ทำให้ขบวนการดำเนินโดยไม่ขึ้นกับความเข้มของคลื่นสะท้อน ตราบเท่าที่วัตถุยังคงสะท้อนคลื่นที่สามารถตรวจจับได้ออกมา ดังนั้นคุณลักษณะการสวิทซ์ไม่เปลี่ยนไป แม้ในสภาวะที่การสะท้อนเป็นไปอย่างไม่ดีคลื่นสะท้อนที่อ่อนจะมีผลต่อความถูกต้องในการตรวจจับวัตถุ ซึ่งอาจทำให้ไม่สามารถทำการตรวจจับวัตถุได้เลย ความเร็วที่เปลี่ยนไปของคลื่นพัลซ์ของเสียง มีผลกระทบต่อพิสัย การทำงานของสวิทช์ (ระยะทาง) โดยตรงเซ็นเซอร์ทำงานด้วยวงจรเวลาที่คงที่ (เช่น t = 20 ms) จะส่งคลื่นเสียงออกมาอย่างสม่ำเสมอ (ดังแสดงในรูป)ดังนั้นวงจรเวลาจะเป็นตัวกำหนดช่วงและวงจรการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

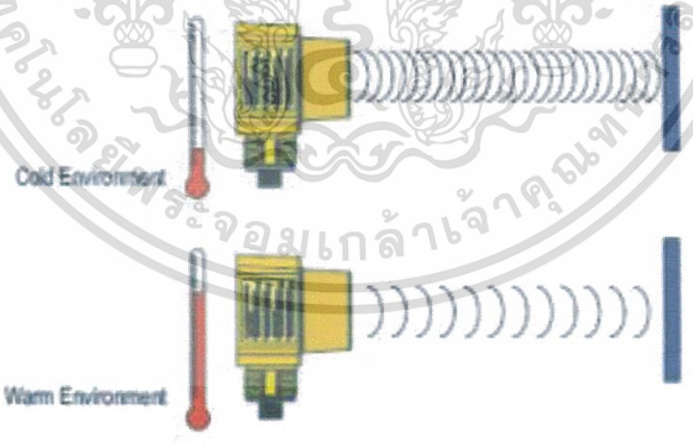
ทำงานของสวิทซ์ของเซ็นเซอร์



ภาพที่ 2.9 อัลตร้าโซนิคเซ็นเซอร์ , วงจรเวลาคงที่

ผลกระทบของอุณหภูมิ(Temperature Effect)

ความไวของเสียงขึ้นอยู่กับแรงดัน และ อุณหภูมิของก๊าซที่เสียงเดินทางผ่าน ในการประยุกต์ใช้อัลตร้าโซนิคส่วนใหญ่องค์ประกอบอื่นๆ และรงดันของก๊าซจะถูกกำหนดให้มีความสัมพันธ์กัน ในขณะที่อุณหภูมิไม้ได้ถูกกำหนดไว้ โดยความไวของเสียงจะเพิ่มขึ้น 1 % ต่ออุณหภูมิที่เพิ่มขึ้น 10° F (6° C)

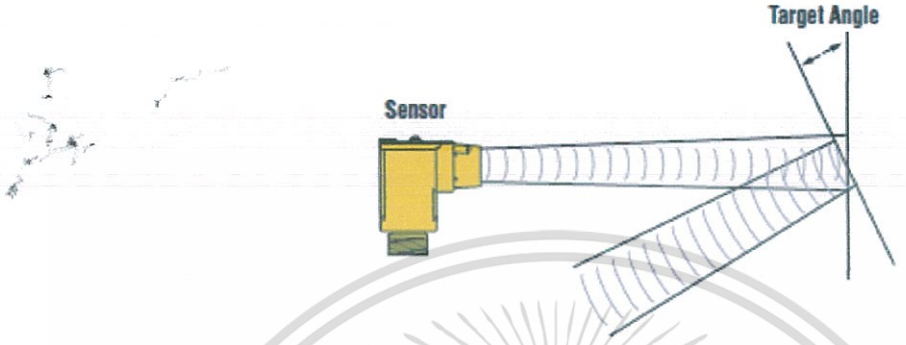


ภาพที่ 2.10 คลื่นอัลตร้าโซนิคที่อุณหภูมิต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมของวัตถุ(Target Angle)

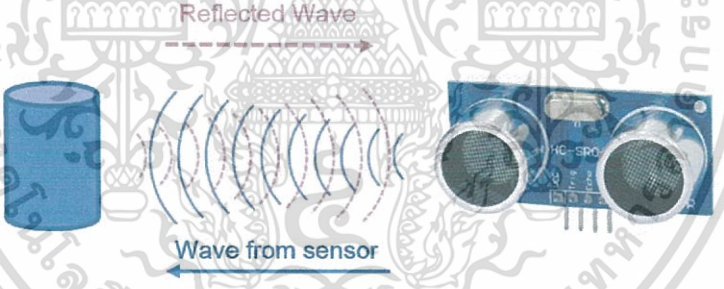
วัตถุที่มีลักษณะแบนที่ตั้งกับแกนของลำแสงจะสะท้อนพลังงานเสียงไปยังเซ็นเซอร์ได้มากที่สุด ดังนั้นถ้ามุมของวัตถุเพิ่มมากขึ้น พลังงานโดยรวมจะส่งกลับไปยังเซ็นเซอร์ได้น้อยลง สำหรับอัลตร้าโซนิคส่วนใหญ่มุมของวัตถุควรจะน้อยกว่า หรือเท่ากับ 10 องศา



ภาพที่ 2.11 แสดงการสะท้อนของเสียง

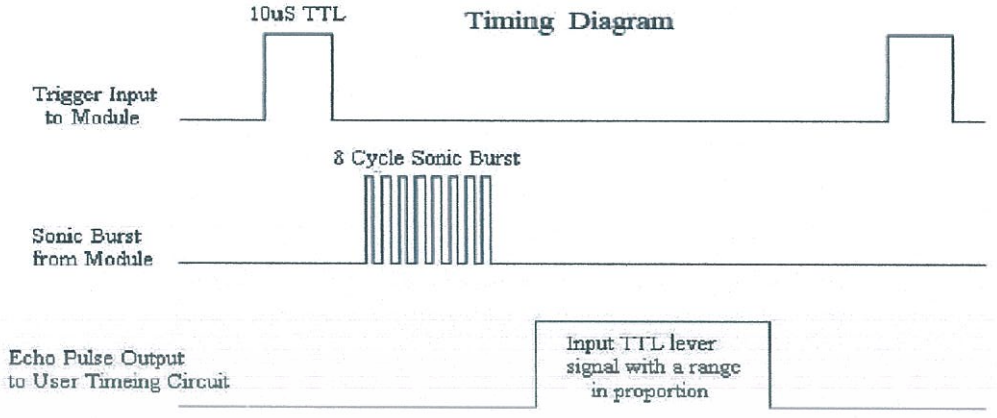
กระแสอากาศ(Air Currents)

กระแสอากาศที่เนื่องมาจากลม, พัดลม, อุปกรณ์นิวแมติกหรือแหล่งอื่นๆสามารถรบกวนเส้นทางของพลังงานเสียงได้ ดังนั้นเซ็นเซอร์อาจไม่สามารถตรวจจับวัตถุในสภาพแวดล้อมแบบนี้ได้



ภาพที่ 2.12 Distance detect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.13 สัญญาณการส่งคลื่นเสียงของ Ultrasonic sensor

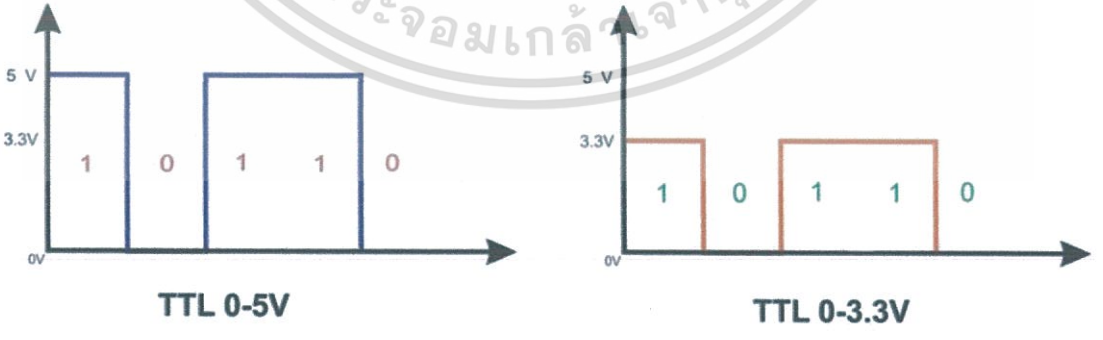
ตัวเซนเซอร์จะเริ่มต้นทำงานโดยการส่งสัญญาณเริ่มต้นยาว 10 ไมโครวินาที ไปสั่งแหล่งกำเนิดเสียง จากนั้นแหล่งกำเนิดเสียงจะส่งคลื่นเสียงที่ความถี่ 40 KHz ออกไป 8 พัลส์ แล้วรอรับเสียงที่สะท้อนกลับมา โดยตัวเซนเซอร์จะมีมุมวัดได้ที่จำกัดอยู่ที่ 15 องศา และวัดได้ช่วง 2 ซม. ถึง 400 ซม. โดยระยะทางคำนวณได้จาก

ระยะทาง = (ความยาวของสัญญาณสะท้อน x 340(m/s)) / 2

2.4 TTL และ RS232

2.4.1 TTL (Transistor-Transistor Logic)

TTL เป็นระดับสัญญาณที่ถูกกำกำหนดขึ้นเพื่อใช้ระหว่าง Transistor กับ Transistor ภายในวงจรรวม(IC) ดังนั้น TTL จะใช้ระดับสัญญาณอยู่ที่ 0-5 Volt แต่ในปัจจุบันมีอุปกรณ์หลากหลายที่สามารถทำงานในช่วง 0-3.3 Volt (เรียกแรงดันระดับนี้ว่า LVTTL) ซึ่งผู้ใช้ควรตรวจสอบจาก datasheet ของอุปกรณ์ที่ใช้เพื่อป้องกันอุปกรณ์เสียหาย



ภาพที่ 2.14 ระดับของสัญญาณชนิด TTL

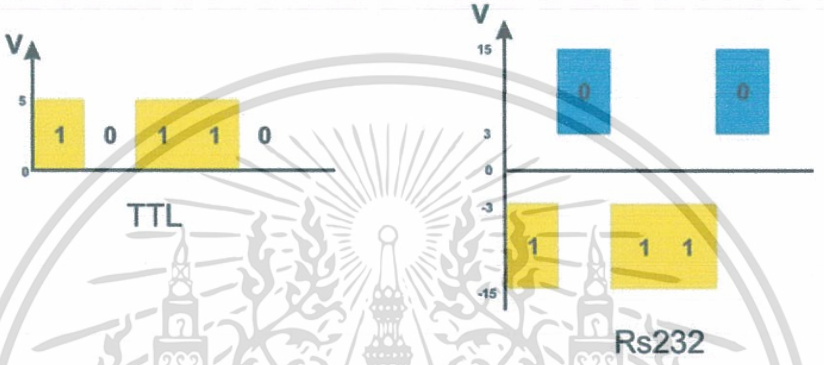
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 RS232 (Recommended Standard 232)

RS232 คือ มาตรฐานการเชื่อมต่อข้อมูลแบบ Serial ใช้เพื่อเพิ่มระยะทางในการส่งข้อมูลแบบ Serial ให้สามารถส่งได้ไกลขึ้น โดยมีการเปลี่ยนแปลงระดับของสัญญาณ จากเดิมที่จะอยู่ในช่วง 0-5 Volt หรือ 0-3.3 Volt เป็นช่วง -15 ถึง +15 Volt โดยมีรายละเอียดดังนี้

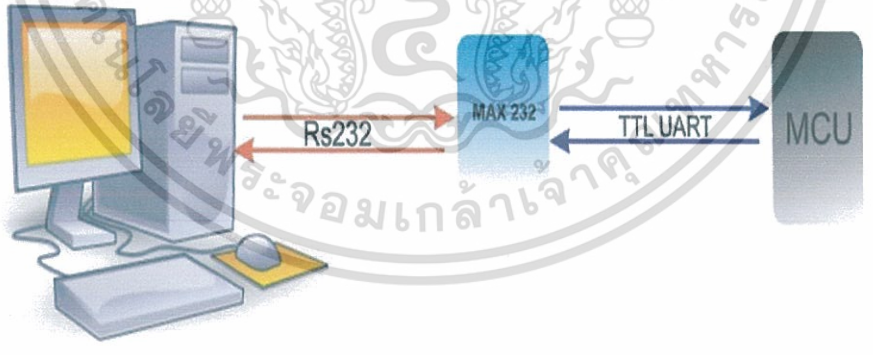
Logic 0 ของ RS232 จะอยู่ในช่วง 3 ถึง 15 Volt

Logic 1 ของ RS232 จะอยู่ในช่วง -3 ถึง -15 Volt



ภาพที่ 2.15 ระดับของสัญญาณชนิด RS232

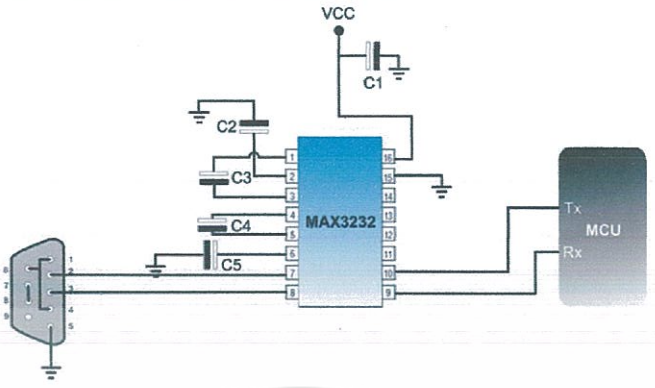
2.4.2 MAX232 CPE IC คือ IC ที่ใช้ในการเปลี่ยนระดับของสัญญาณระหว่าง สัญญาณ TTL เป็น RS232



ภาพที่ 2.16 การใช้งาน MAX232 CPE IC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2.1 วิธีต่อการใช้งาน MAX232CPE



Vcc = 3.3V

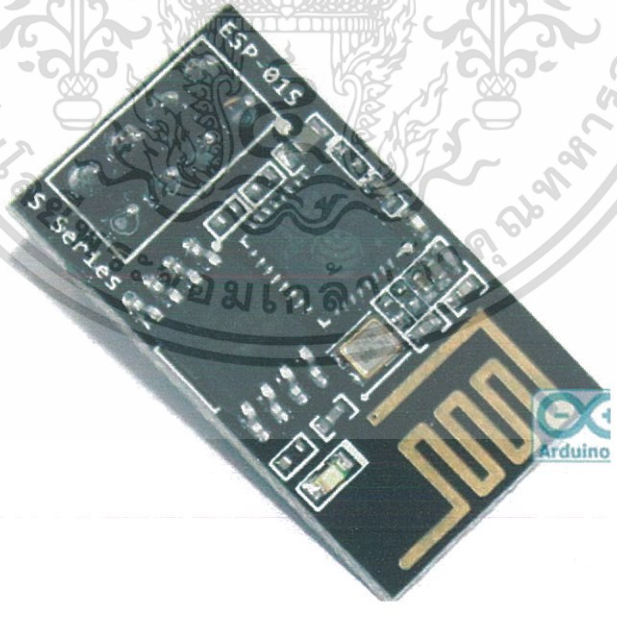
C1-C5 = 0.1uF

ภาพที่ 2.17 การต่อการใช้งาน MAX232CPE IC

2.5 ESP8266-01 Wireless module

2.5.1 ESP8266

ESP8266-01 เป็นโมดูลไวไฟ และเป็นไมโครคอลโทรลเลอร์ โดยในตัวมีความจำมากถึง 4 MB และมีขนาดเล็กใช้พลังงานต่ำ รองรับการทำงานได้หลายรูปแบบ ใช้การเชื่อมต่อแบบ Serial UART ที่ไฟ 3.3 Volt ใช้คำสั่ง AT Command หรือ Library JSON ของ Arduino IDE ในการเขียนโปรแกรม



ภาพที่ 2.18 ESP8266-01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สเปค:

802.11 b/g/n

WIFI Direct (P2P), SOFT-AP

Integrated TR switch, balun, LNA, Power amplifier and matching network

Integrated PLLs, regulators, DCXO and power management units

+19.5dBm output power in 802.1b mode

Power down leakage current of <10 uA

Integrated low power 32-bit CPU could be used as application processor

SDIO 1.1/2.0, SPI, UART

STBC, 1×1 MIMO, 2×1 MIMO

A-MPDU & A-MSDU aggregation & 0.4ms guard interval

Integrated TCP/IP protocol stack

Flash Memory Size: 1MB (8Mbit)

Wi-Fi security modes: WPA, WPA2

Regular operation current draw: ~70mA

2.5.2 การเชื่อมต่อ ESP8266-01

ESP8266-01 มีความสามรถในการส่งข้อมูลผ่านทาง server ได้ อีกทั้งยังสามารถเปลี่ยนให้ตัวเองเป็น client, server และclientกับserver ในเวลาเดียวกัน โดยการส่งข้อมูลจะมีการเชื่อมต่อ client ไปที่ module แล้วขอข้อมูลโดยสร้างโปรแกรมจากภาษา C#

ภาพที่ 2.19 ตัวอย่างการใช้งาน ESP8266-01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 TCP (transmission control protocol)

TCP/IP (Transmission Control Protocol/Internet Protocol) เป็นชุดของโปรโตคอลที่ถูกใช้ในการสื่อสารผ่านเครือข่ายอินเทอร์เน็ต โดยมีวัตถุประสงค์เพื่อให้สามารถใช้สื่อสารจากต้นทางข้ามเครือข่ายไปยังปลายทางได้ และสามารถหาเส้นทางที่จะส่งข้อมูลไปได้เองโดยอัตโนมัติ ถึงแม้ว่าในระหว่างทางอาจจะผ่านเครือข่ายที่มีปัญหา โปรโตคอลก็ยังคงหาเส้นทางอื่นในการส่งผ่านข้อมูลไปให้ถึงปลายทางได้

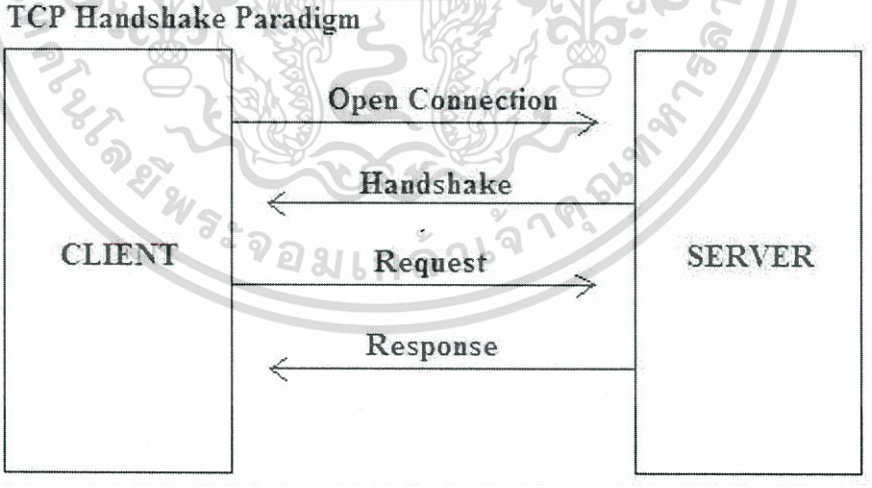
TCP/IP แบ่งออกเป็น 4 เลเยอร์

- Link Layer - เลเยอร์นี้มีหน้าที่ควบคุมการรับส่งข้อมูลในระดับฮาร์ดแวร์ของเครือข่าย รับผิดชอบการรับส่งข้อมูลในระดับกายภาพ จนถึงการแปลความจะสัญญาณไฟฟ้าเป็นข้อมูลทางคอมพิวเตอร์

- Network Layer - ทำหน้าที่รับข้อมูลจากชั้น Transport Layer และค้นหาและเลือกเส้นทาง ระหว่างผู้รับและผู้ส่ง เทียบได้กับ Network Layer ของ OSI Model โปรโตคอลในเลเยอร์นี้ได้แก่ IP, ICMP, IGMP

- Transport Layer - รับผิดชอบการรับส่งข้อมูลระหว่างปลายด้านส่งและด้านรับข้อมูล และส่งข้อมูลขึ้นไปให้ Application Layer นำไปใช้งาน ต่อ เทียบได้กับ Session Layer และ Transport Layer ของ OSI Model

- Application Layer - เป็นเลเยอร์ที่แอพพลิเคชั่นเรียกโปรโตคอลระดับล่างๆลงไป เพื่อให้บริการต่างๆ เช่น FTP, SMTP, Telnet, HTTP, POP

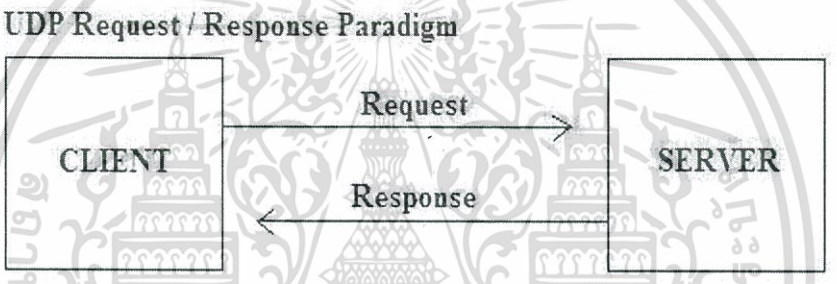


ภาพที่ 2.20 การส่งข้อมูลแบบ TCP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 UDP (User Datagram Protocol)

UDP (User Datagram Protocol) เป็นโปรโตคอลที่อยู่ใน Transport Layer เมื่อเทียบกับโมเดล OSI โดยการส่งข้อมูลของ UDP นั้นจะเป็นการส่งครั้งละ 1 ชุดข้อมูล เรียกว่า UDP datagram ซึ่งจะไม่มีความสัมพันธ์กันระหว่างดาต้าแกรมและจะไม่มีกลไกการตรวจสอบความสำเร็จในการรับส่งข้อมูล กลไกการตรวจสอบโดย checksum ของ UDP นั้นเพื่อเป็นการป้องกันข้อมูลที่อาจจะถูกแก้ไข หรือมีความผิดพลาดระหว่างการส่ง และหากเกิดเหตุการณ์ดังกล่าว ปลายทางจะได้รู้ว่ามีข้อผิดพลาดเกิดขึ้น แต่มันจะเป็นการตรวจสอบเพียงฝ่ายเดียวเท่านั้น โดยในข้อกำหนดของ UDP หากพบว่า Checksum Error ก็ให้ผู้รับปลายทางทำการทิ้งข้อมูลนั้น แต่จะไม่มีการแจ้งกลับไปยังผู้ส่งแต่อย่างใด การรับส่งข้อมูลแต่ละครั้งหากเกิดข้อผิดพลาดในระดับ IP เช่น ส่งไม่ถึง, หมดเวลา ผู้ส่งจะได้รับ Error Message จากระดับ IP เป็น ICMP Error Message แต่เมื่อข้อมูลส่งถึงปลายทางถูกต้อง แต่เกิดข้อผิดพลาดในส่วนของ UDP เองจะไม่มีการยืนยัน หรือแจ้งให้ผู้ส่งทราบแต่อย่างใด



ภาพที่ 2.21 การส่งข้อมูลแบบ UDP

2.8 Visual C#

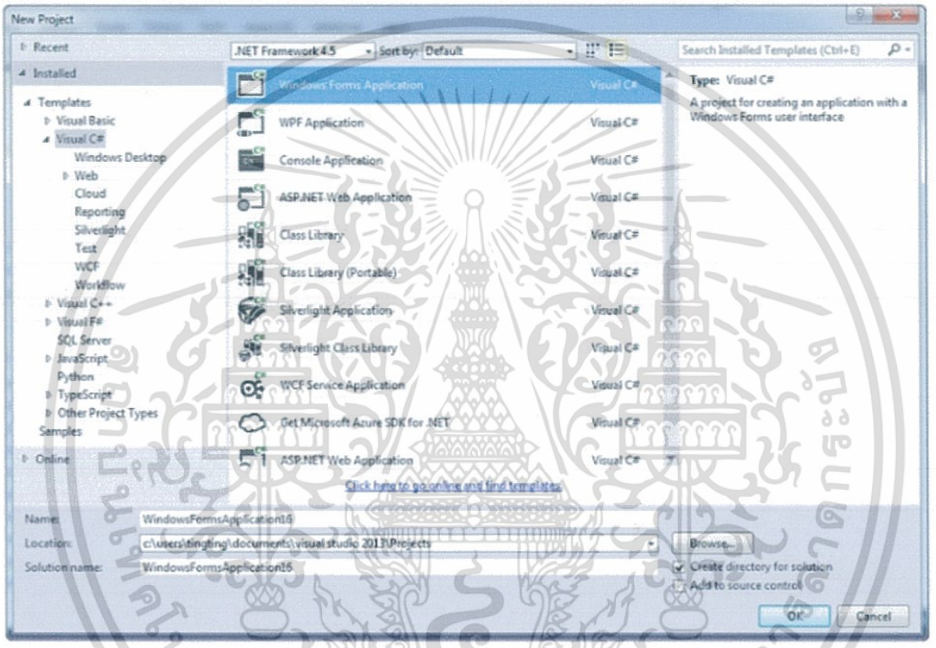
ภาษา C# (C# Programming Language) จะเขียนว่า C Sharp (ซี-ชาร์พ) ภาษา C, Java, C++ เป็นที่มาของ C# (ตัว C เป็นตัวบ่งบอกให้รู้ว่า C# มีต้นกำเนิดมาจาก C นั่นเอง) เครื่องหมาย # ดังที่เคยพูดคือ เป็นสิ่งที่แสดงถึงความก้าวหน้ากว่า C++ ไปอีกระดับนึง ถ้าจะให้พูดกันเข้าใจง่ายๆอีกก็คือ C# ได้รวบรวมข้อดีของภาษาต่างๆเช่น Java, Delphi, C++ เข้าไว้ด้วยกัน มีความเรียบง่าย อีกทั้งยังมีเครื่องมือดีๆ อย่าง Visual C# 2008 ของทางไมโครซอฟท์อีกด้วย จึงลดความยุ่งยากในการโปรแกรมได้มาก สามารถพัฒนาโปรแกรมระดับสูงได้ด้วย ** ภาษา C , C++, C# จัดเป็นภาษาระดับกลาง กล่าวคือได้รวมเอาข้อดีจาก ภาษาระดับต่ำ และสูงเข้าไว้ด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.8.1 Visual studio

Visual Studio เป็นชุดโปรแกรมที่นำไปใช้เขียนโปรแกรมคอมพิวเตอร์ด้วยภาษาต่างๆ เช่น ภาษา C, ภาษา C++ และ ภาษา C# เป็นต้น เพื่อสร้างโปรแกรมคอมพิวเตอร์ให้ตรงตามวัตถุประสงค์ที่ต้องการ โดยโปรแกรมนี้ได้รวบรวมเครื่องมือต่างๆที่ใช้สำหรับเขียนโปรแกรมคอมพิวเตอร์เข้าไว้ด้วยกัน เพื่อที่จะคอยอำนวยความสะดวกให้กับผู้ใช้งาน ปัจจุบันบริษัทไมโครซอฟท์ได้เปิดให้นักพัฒนาและบุคคลทั่วไปดาวน์โหลดโปรแกรม Visual Studio 2013 ไปใช้สำหรับการพัฒนาโปรแกรมบนแพลทฟอร์มต่างๆ ได้ฟรี ซึ่งสามารถดาวน์โหลดได้ฟรีจากเว็บไซต์ www.visualstudio.com



ภาพที่ 2.22 โปรแกรม Visual Studio 2013

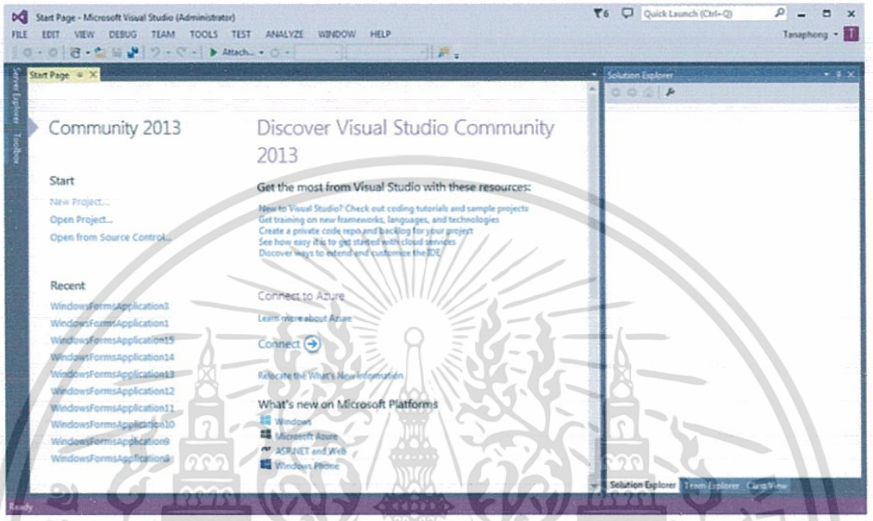
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

148491

2.8.2การเริ่มต้นใช้งานโปรแกรม Visual Studio

2.8.2.1การสร้าง Project

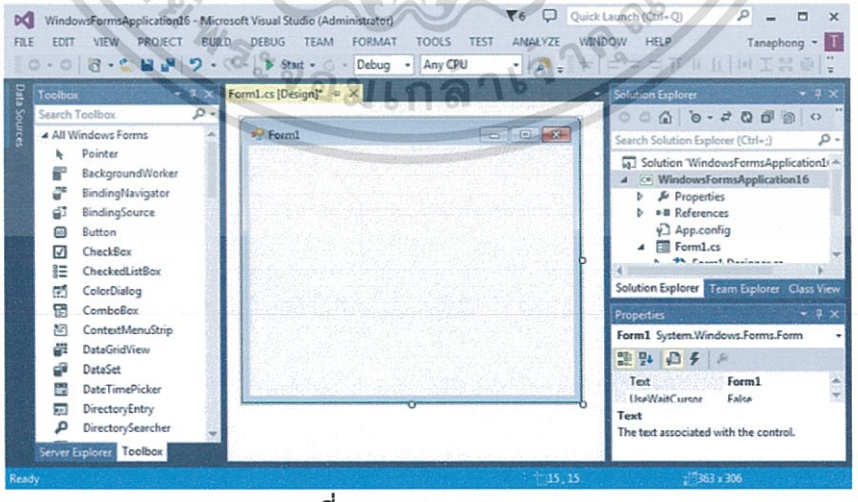
การเขียนโปรแกรมด้วย Visual Studio 2013 จะต้องเป็นการเขียนโปรแกรมใน Project ซึ่งเราสามารถสร้าง Project ใหม่ได้ โดยคลิกที่เมนู File > New > Project จากนั้นจะพบกับหน้าต่าง ดังภาพ



ภาพที่ 2.23 การสร้างโปรเจค Visual studio

ในเมนูแถบด้านซ้าย ให้เลือกที่ Visual C# เพื่อกำหนดลักษณะของ Project ว่าเป็นการเขียนโปรแกรมด้วยภาษา C# ส่วนรายการตรงกลาง ให้เลือกที่ Windows Form Application เพื่อกำหนดว่าเป็นการสร้างโปรแกรมที่รันบนระบบปฏิบัติการ Windows จากนั้นกดปุ่ม OK

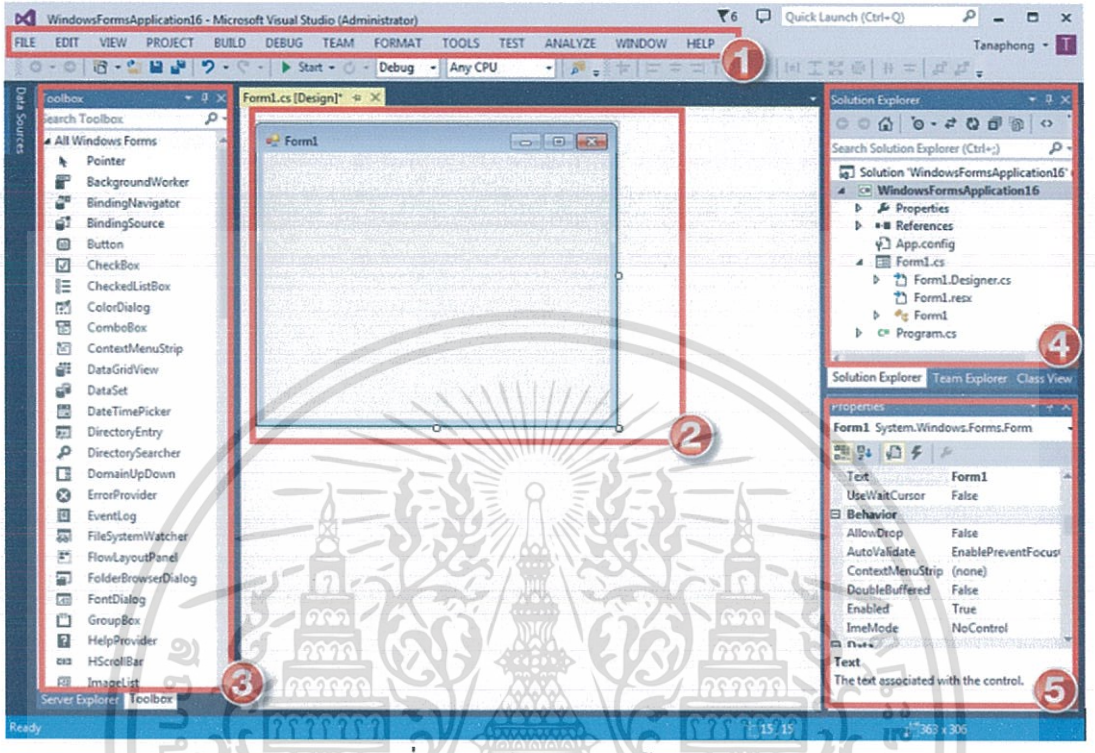
จะพบกับหน้าจอของ Windows Form Application เปล่าๆ ขึ้นมา ซึ่งเราสามารถเริ่มเขียนโปรแกรมได้ทันที ดังภาพ 2.23



ภาพที่ 2.24 Form Program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.2.2ส่วนต่างๆ ของโปรแกรม Visual Studio



ภาพที่ 2.25 ส่วนประกอบโปรแกรม

1. แถบเมนู (Menu bar) เป็นแถบแสดงรายการคำสั่งต่างๆ ของโปรแกรม เช่น การสร้างโปรเจคใหม่ การตั้งค่าโปนแกรม หรือการเรียกใช้เครื่องมืออื่นๆ เป็นต้น
2. พื้นที่สำหรับเขียนโปรแกรม (Workspace) เป็นพื้นที่สำหรับการออกแบบและเขียนคำสั่งต่างๆ ให้โปรแกรม
3. กล่องเครื่องมือ (Toolbox) เป็นกลุ่มของเครื่องมือต่างๆ สำหรับใช้ในการออกแบบหน้าจอของโปรแกรม เช่น Button, Listbox, Checkbox, Radio button Label เป็นต้น
4. ส่วนควบคุมการทำงาน (Solution Explorer) เป็นส่วนสำหรับแสดงโปรเจคหรือไฟล์ที่กำลังเปิดอยู่
5. หน้าต่างแสดงคุณสมบัติของเครื่องมือ (Properties) เป็นส่วนที่กำหนดคุณสมบัติต่างๆ ให้กับเครื่องมือแต่ละตัว เช่น การกำหนดสี กำหนดขนาดตัวอักษร และอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีดำเนินการ

3.1 วิธีดำเนินการทำโครงงาน

3.1.1 กำหนดหัวข้อที่จะทำการศึกษา โดยศึกษาจากการทำงานของบริษัทและนำปัญหาที่พบเจอหรือสิ่งที่สามารถนำพัฒนาได้มาแก้ไข

3.1.2 ศึกษาข้อมูลต่างๆ ที่เกี่ยวข้อง

3.1.2.1 ศึกษาการทำงานของหุ่นยนต์

3.1.2.2 ศึกษาหลักการทำงานของ Ultrasonic sensor และ RGB Sensor

3.1.2.3 ศึกษาบอร์ด Arduino ต่างๆ เพื่อเลือกใช้บอร์ดที่เหมาะสม

3.1.2.4 ศึกษาหลักการส่งข้อมูลระหว่างชุดคอนโทรล

3.1.2.5 ศึกษาโปรแกรม Visual C#

3.1.2.6 ศึกษาการใช้ Wireless module (ESP8266-01)

3.1.2.7 ศึกษาการส่งข้อมูลแบบ RS232

3.1.3 เขียนโปรแกรมการทำงาน โดยใช้โปรแกรม Arduino

3.1.3.1 เขียนโค้ดการส่งข้อมูลส่งไปยัง PLC และ Wireless module

3.1.3.2 เขียนโค้ด Wireless module ส่งไปยัง PC โดยใช้ Visual C#

3.1.3.3 เขียนโค้ดเช็คระยะห่างโดย Ultrasonic sensor

3.1.3.4 เขียนโค้ดเช็คสีโดย RGB Sensor

3.1.3.5 เขียนโค้ดการส่งข้อมูลแบบ Serial communication

3.1.4 ทดสอบการทำงาน และเก็บข้อมูลผลการทดลอง

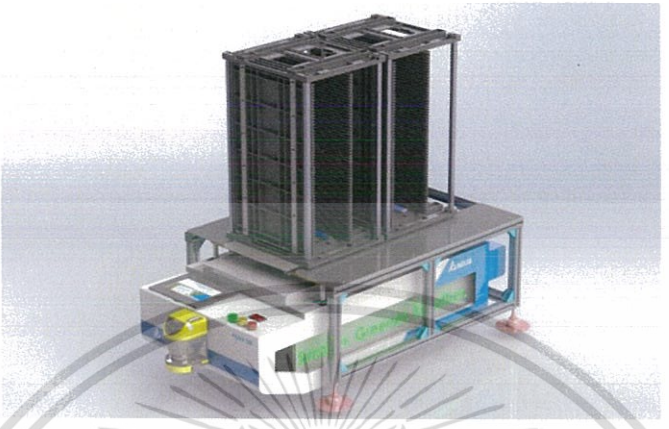
3.1.5 ตรวจสอบ และแก้ไขโค้ด

3.1.6 สรุปผล

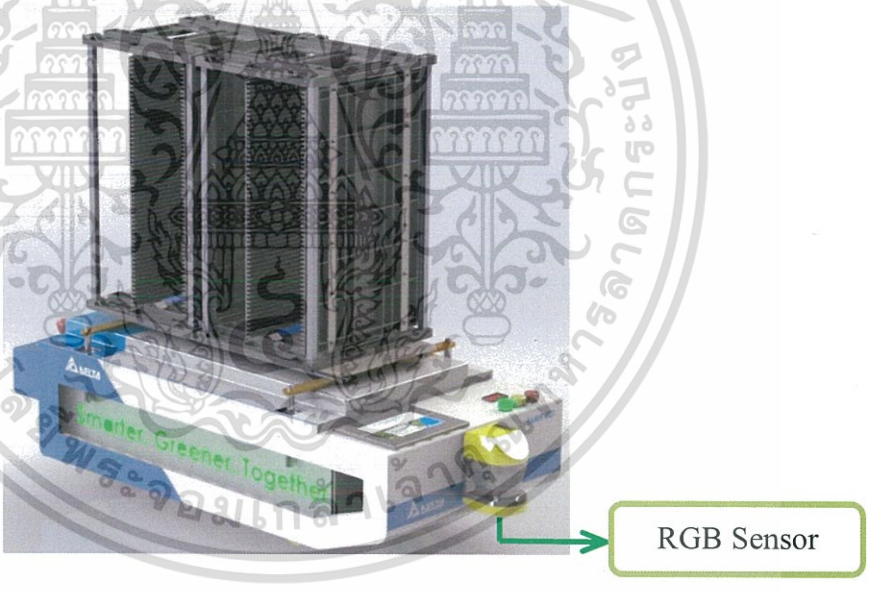
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ตรวจสอบตำแหน่งที่ติดตั้งเซนเซอร์

รถขนส่งของ Fast Automatic Mobile Robot หรือ รถ AVG

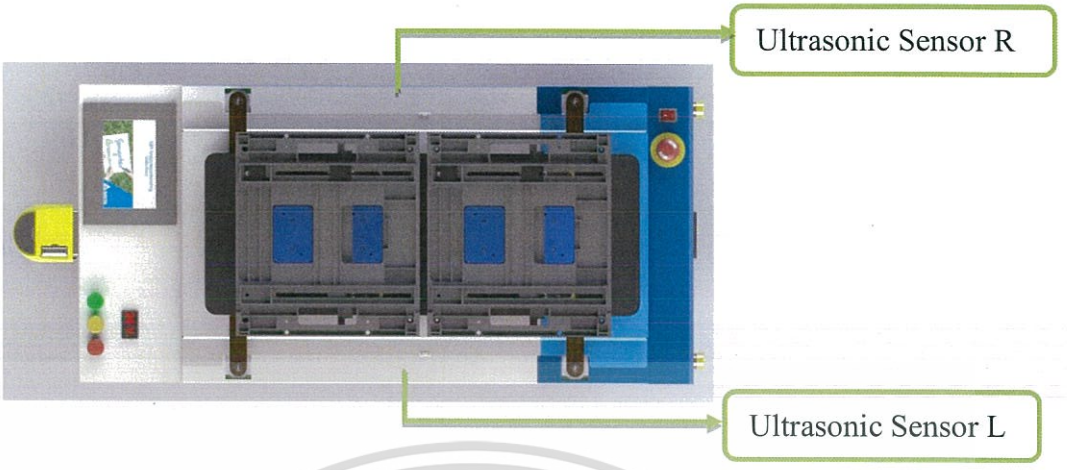


ภาพที่ 3.1 รถ AVG



ภาพที่ 3.2 ตำแหน่ง RGB Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.3 ตำแหน่ง Ultrasonic Sensor ซ้ายและขวา

3.3 การเก็บค่าเซนเซอร์

3.3.1 ติดตัวเซ็นเซอร์เข้า Arduino UNO R3

3.3.2 เขียนโค้ดการอ่านค่าจากเซ็นเซอร์ Ultrasonic sensor

```
void Mesure()
{
  digitalWrite(trigPinL, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPinL, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPinL, LOW);
  delayMicroseconds(10);
  durationL = pulseIn(echoPinL, HIGH);
  digitalWrite(trigPinR, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPinR, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPinR, LOW);
  durationR = pulseIn(echoPinR, HIGH);
  distanceL = durationL/28.84/2.00;  // 340m/s = 29.4Micro:
  distanceR = durationR/28.84/2.00;  // 347m/s = 28.8Micro:
}
```

ภาพที่ 3.4 คำสั่งอ่านค่าเซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 เขียนโค้ดการ RGB sensor เพื่อเช็คความถูกต้องในการเคลื่อนที่ของหุ่นยนต์

```
void color()
{
  digitalWrite(s2, LOW);
  digitalWrite(s3, LOW);
  delayMicroseconds(15);
  red = pulseIn(out, HIGH , 1000);
  digitalWrite(s3, HIGH);
  delayMicroseconds(15);
  blue = pulseIn(out, HIGH , 1000);
  digitalWrite(s2, HIGH);
  delayMicroseconds(15);
  green = pulseIn(out, HIGH , 1000);
  Blue = map(blue, 300, 3, 0, 255);
  Green = map(green, 350, 3, 0, 255);
  Red = map(red, 400, 3, 0, 255);
  if(Blue<=0)
  { Blue = 0;}
  if(Green<=0)
  { Green = 0;}
  if(Red<=0)
  { Red = 0;}
  if(Red>180 && Blue >200 && Green >200)
  { Red = 0; Blue = 0 ; Green = 0;}
}
```

ภาพที่ 3.5 คำสั่งตรวจสอบสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 คำสั่งการส่งข้อมูลผ่าน Wireless module

3.4.1 เขียนโค้ด ส่งคำสั่งการตรวจสอบต่างๆ ผ่าน Wireless module

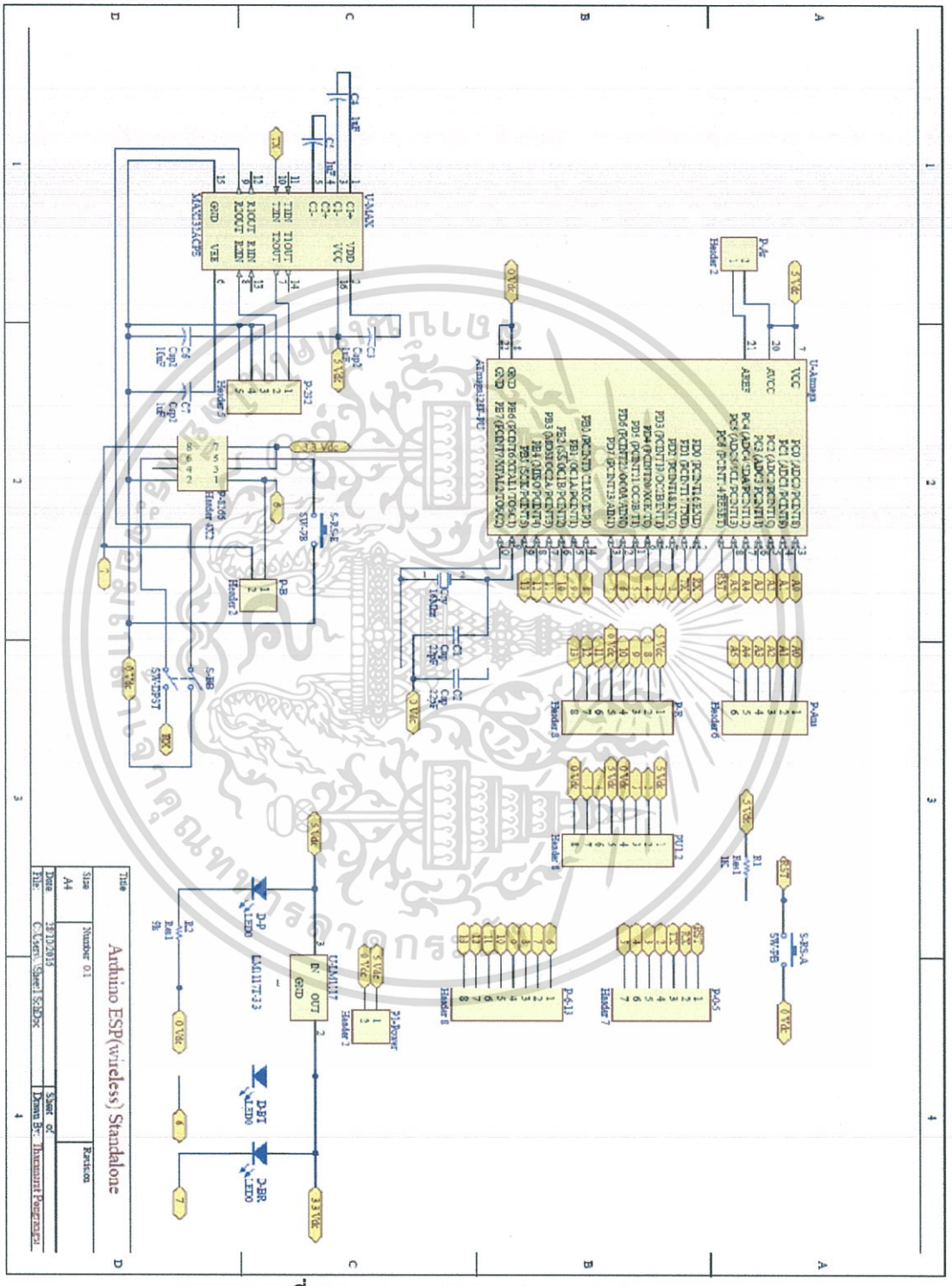
```
void ESP()
{
  String s;
  String ss;
  x=0;
  while (mySerial.available())
  {
    delayMicroseconds(1000);
    uint8_t t = mySerial.read();
    char c = t;
    s += c;
    ss += c;
    x = 1;
  }
  r = 0;
  String a,aa;
  if(x == 1)
  {
    Serial.print(s);     // รับข้อความ จาก esp แล้วส่งไป PLC
    x = 0;
  }
  while(Serial.available())
  {
    delayMicroseconds(1000);
    uint8_t q = Serial.read();
    char w = q;
    if(r == 0)
    {
      aa += "";
    }
    a += w;
    aa += w;
    r = 1;
  }
  if(r == 1)
  {
    mySerial.print(a);     // รับข้อความ จาก esp แล้วส่งไป PLC
    r = 0;
  }
}
```

ภาพที่ 3.6 คำสั่งส่งข้อมูลผ่าน Wireless

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

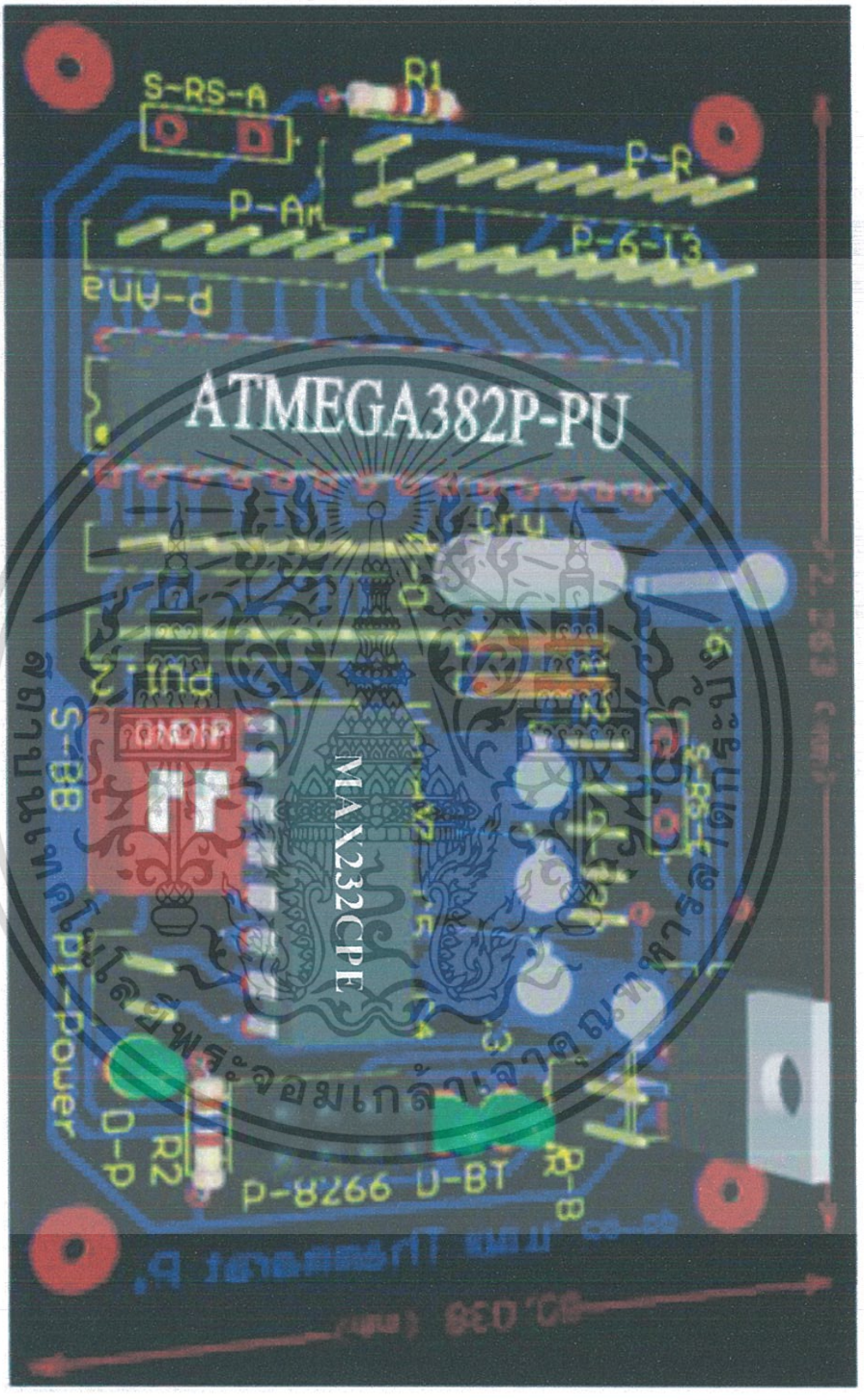
3.5 ออกแบบ Schematic และ PCB

c 3.5.1 Schematic design



ภาพที่ 3.7 Schematic hardware design

3.5.2 PCB design



ภาพที่ 3.8 PCB hardware design

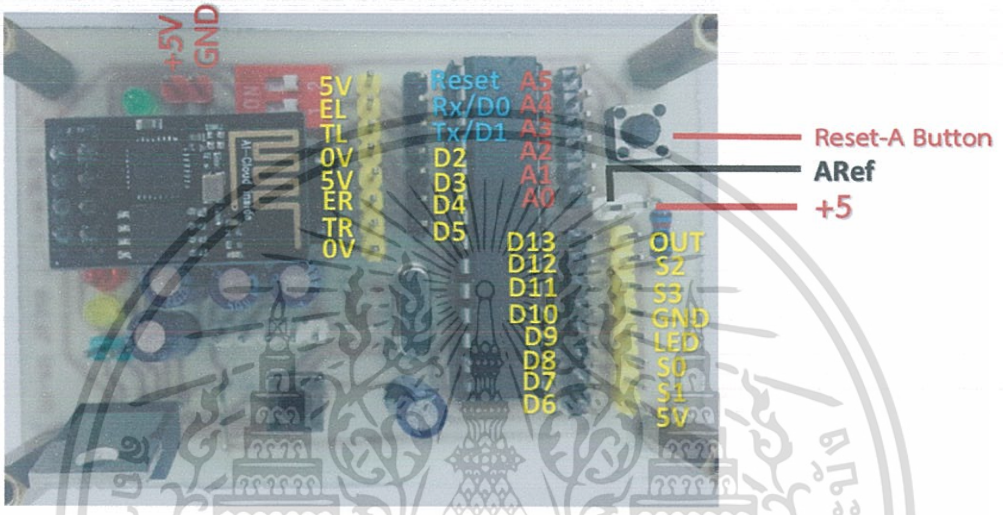
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 รายละเอียดการดำเนินการทดลอง

4.1.1อุปกรณ์การทดลอง

1. บอร์ด Arduino ESP8266-01



ภาพที่ 4.1 บอร์ด Arduino Uno R3 and ESP8266-01

2. RGB Sensor

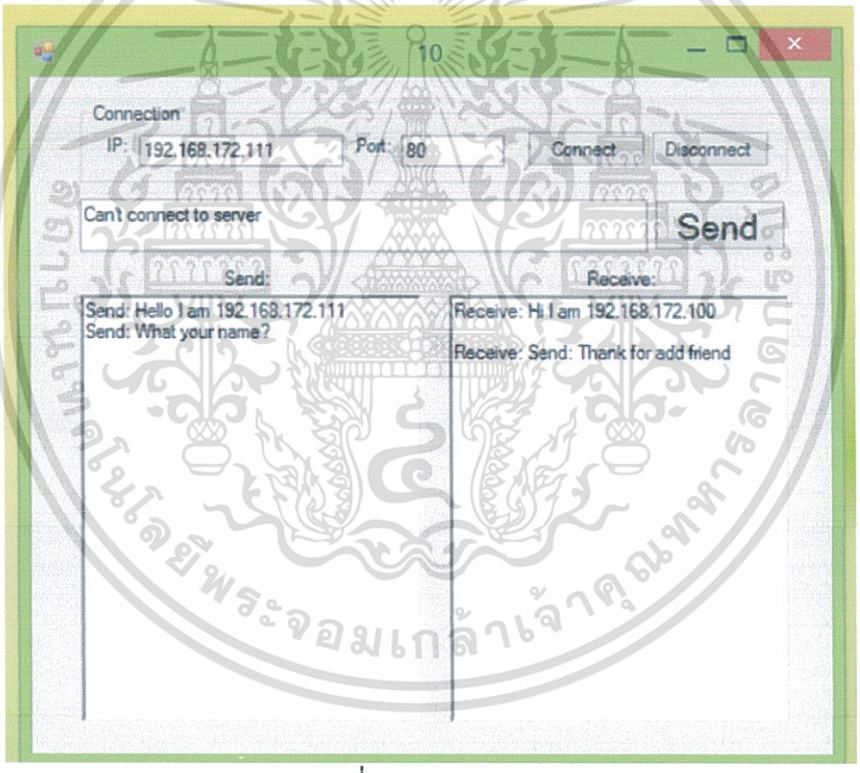
ภาพที่ 4.2 RGB Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.Ultrasonic Sensor

ภาพที่ 4.3 Ultrasonic Sensor

4. โปรแกรมเชื่อมต่อ ESP8266-01 (TT)



ภาพที่ 4.4 TT Program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ขั้นตอนการทดลอง

ตอนที่ 1: ทดสอบระยะการวัดของ Ultrasonic sensor

วัดเพื่อหาค่าระยะต่ำสุดและมากสุดที่โมดูลจะตรวจสอบได้

ขั้นตอนการปฏิบัติการ

1. เชื่อมต่อโมดูล Ultrasonic Sensor กับบอร์ด Arduino ESP8266-01 ตามที่กำหนด Pin ไว้ในโค้ด
2. นำวัตถุผิวเรียบมาทดสอบระยะใกล้ไกล ทำการบันทึกผล
3. ทำทั้งหมด 3 ครั้ง เพื่อเฉลี่ยค่าระยะต่ำสุดและมากที่สุดที่วัดได้

ตอนที่ 2: ทดสอบระยะการตรวจสีของ RGB Sensor

เพื่อหาความสัมพันธ์ ของระยะในการตรวจสีและความถูกต้องของการตรวจสี

ขั้นตอนการปฏิบัติการ

1. เชื่อมต่อโมดูล RGB Sensor กับบอร์ด Arduino ESP8266-01 ตามที่กำหนด Pin ไว้ในโค้ด
2. ในกระดาษสีน้ำเงินมาตรวจสอบที่ระยะ 1 cm, 5 cm, 10cm, 12cm และ 15cm
3. ทำทั้งหมด 5 ครั้ง เพื่อตรวจสอบความถูกต้องในการตรวจจับสี

ตอนที่ 3: เชื่อมต่อ Arduino ESP8266-01 กับโปรแกรม TT ผ่าน ไวไฟ

เพื่อทดสอบการรับส่งข้อมูลรับแบบข้อความผ่านไวไฟจากเครื่องคอมพิวเตอร์ ไป Arduino ESP8266-01 ผ่านไวไฟ

ขั้นตอนการปฏิบัติการ

1. ลงโปรแกรมให้ ESP8266-01 แล้วทำการรีเซต 1 รอบโดยกำหนด Access point และ IP Address ในโค้ดที่เขียน
2. นำเครื่องคอมพิวเตอร์เชื่อมต่อไวไฟ Access point เดียวกันกลับ ESP8266-01
3. ใช้โปรแกรม TT ในการเชื่อมต่อโดยการใส่ IP Address ที่กำหนดให้ ESP8266-01 ในโค้ด กด connect เพื่อเชื่อมต่อ
4. ทดสอบส่งข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตอนที่ 4: สร้างแผนผังโค้ด และเขียน โค้ดบนบอร์ด Arduino ESP8266-01
เขียนโครงสร้างการทำงานและลำดับขั้นในการทำงานของบอร์ด Arduino ESP8266-01
ขั้นตอนการปฏิบัติการ

1. เขียน application การทำงานของบอร์ด
2. สร้างแผนผัง Flow chart
3. เขียนโค้ดตามลักษณะของ Flow chart และ Application ที่วางไว้

ตอนที่ 5: Test และ Calibration
เพื่อให้เซนเซอร์ วัดค่าได้ถูกต้องมากที่สุด
ขั้นตอนการปฏิบัติการ

1. เชื่อมต่อ Arduino ESP8266-01 กับ RGB Sensor และ Ultrasonic Sensor
2. ทำการปรับเปลี่ยนสภาพอากาศในการทดลองจากอุณหภูมิที่ 25 เป็น 30 องศาเซลเซียส
3. ทำการแก้ไขโค้ด และทำการทดสอบ จนกว่าจะเซนเซอร์จะตรวจสอบได้แม่นยำที่สุด

4.2 ผลการทดลอง

4.2.1 ผลการทดลองตอนที่ 1: ทดสอบระยะการวัดของ Ultrasonic sensor

ตารางที่ 4.1 ตารางแสดงผลระยะ Ultrasonic Sensor

ระยะจริง	1 cm	2 cm	3 cm	4 cm	5 cm	10 cm	15 cm	30 cm
วัดได้	Error	Error	3.18 cm	4.07 cm	5.21 cm	10.01 cm	14.97 cm	29.52 cm

ระยะจริง	50 cm	100 cm	150 cm	200 cm	250 cm	300 cm	350 cm	400 cm
วัดได้	51.19 cm	101.22 cm	149.27 cm	197.57 cm	248.17 cm	297.32 cm	Error	Error

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 ผลการทดลองตอนที่ 2: ทดสอบระยะการตรวจสีของ RGB Sensor

ตารางที่ 4.2 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 1

ระยะตรวจจับ	1 cm	5 cm	10 cm	12 cm	15 cm
ผลที่ได้	ถูกต้อง	ถูกต้อง	ถูกต้อง	ไม่ถูกต้อง	ไม่ถูกต้อง

ตารางที่ 4.3 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 2

ระยะตรวจจับ	1 cm	5 cm	10 cm	12 cm	15 cm
ผลที่ได้	ถูกต้อง	ถูกต้อง	ถูกต้อง	ไม่ถูกต้อง	ไม่ถูกต้อง

ตารางที่ 4.4 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 3

ระยะตรวจจับ	1 cm	5 cm	10 cm	12 cm	15 cm
ผลที่ได้	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ไม่ถูกต้อง

ตารางที่ 4.5 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 4

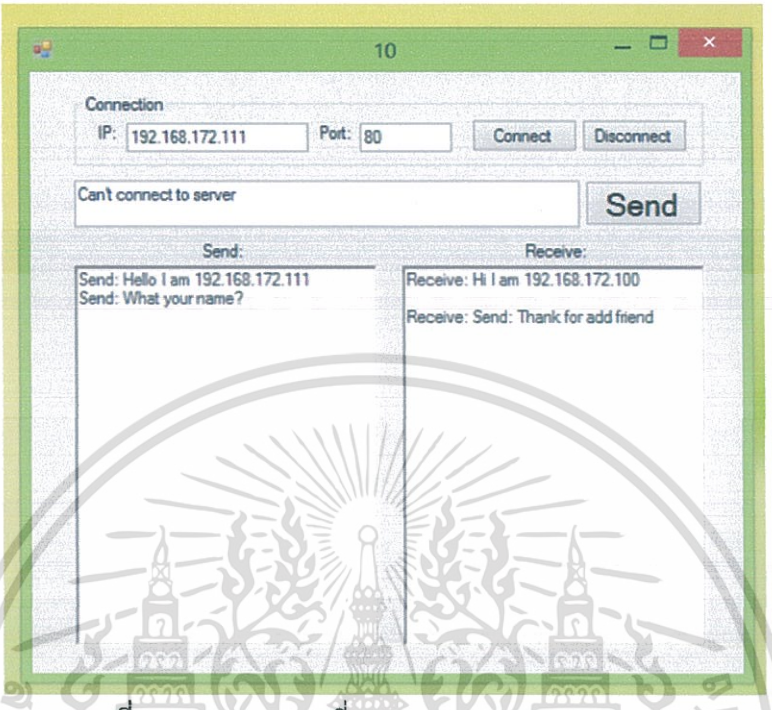
ระยะตรวจจับ	1 cm	5 cm	10 cm	12 cm	15 cm
ผลที่ได้	ถูกต้อง	ถูกต้อง	ถูกต้อง	ไม่ถูกต้อง	ไม่ถูกต้อง

ตารางที่ 4.6 ตารางแสดงผลระยะ RGB Sensor ตรวจจับสีฟ้า ครั้งที่ 5

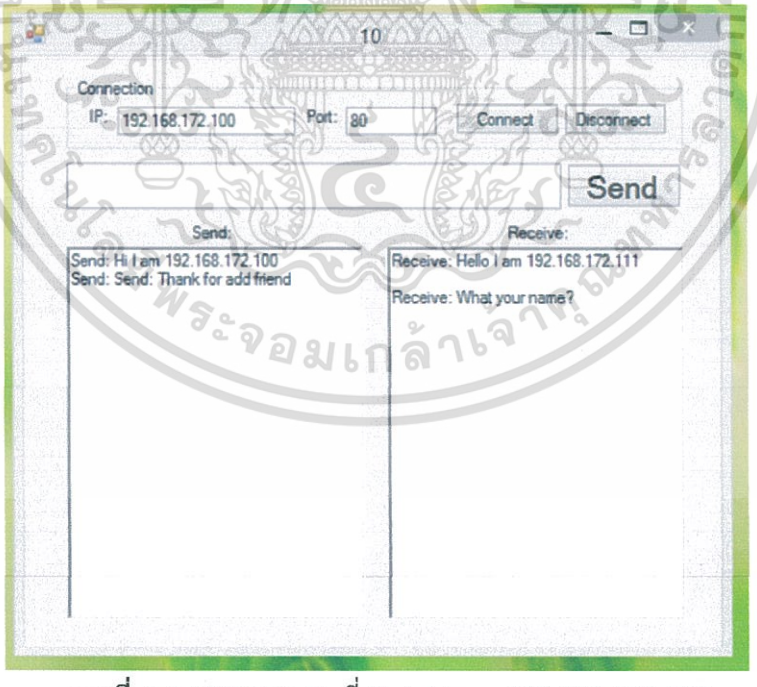
ระยะตรวจจับ	1 cm	5 cm	10 cm	12 cm	15 cm
ผลที่ได้	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ไม่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 ผลการทดลองตอนที่ 3: เชื่อมต่อ Arduino ESP8266-01 กับโปรแกรม TT ผ่าน ไวไฟ



ภาพที่ 4.5 SP8266-01 ที่ IP Address 192.168.172.111

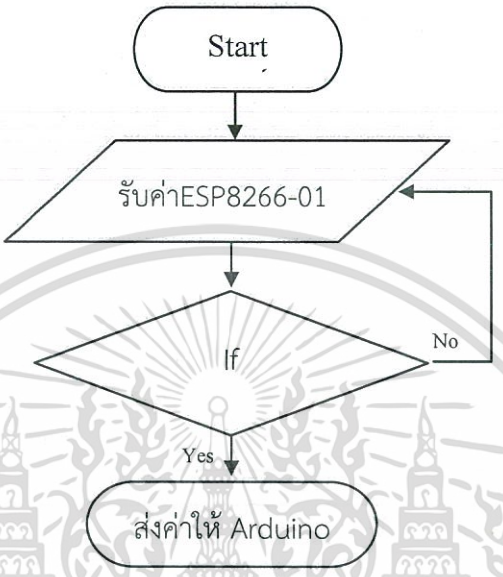


ภาพที่ 4.6 ESP8266-01 ที่ IP Address 192.168.172.100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 ผลการทดลองตอนที่ 4: สร้างแผนผังโค้ดและเขียนโค้ดบนบอร์ดArduinoESP8266-01

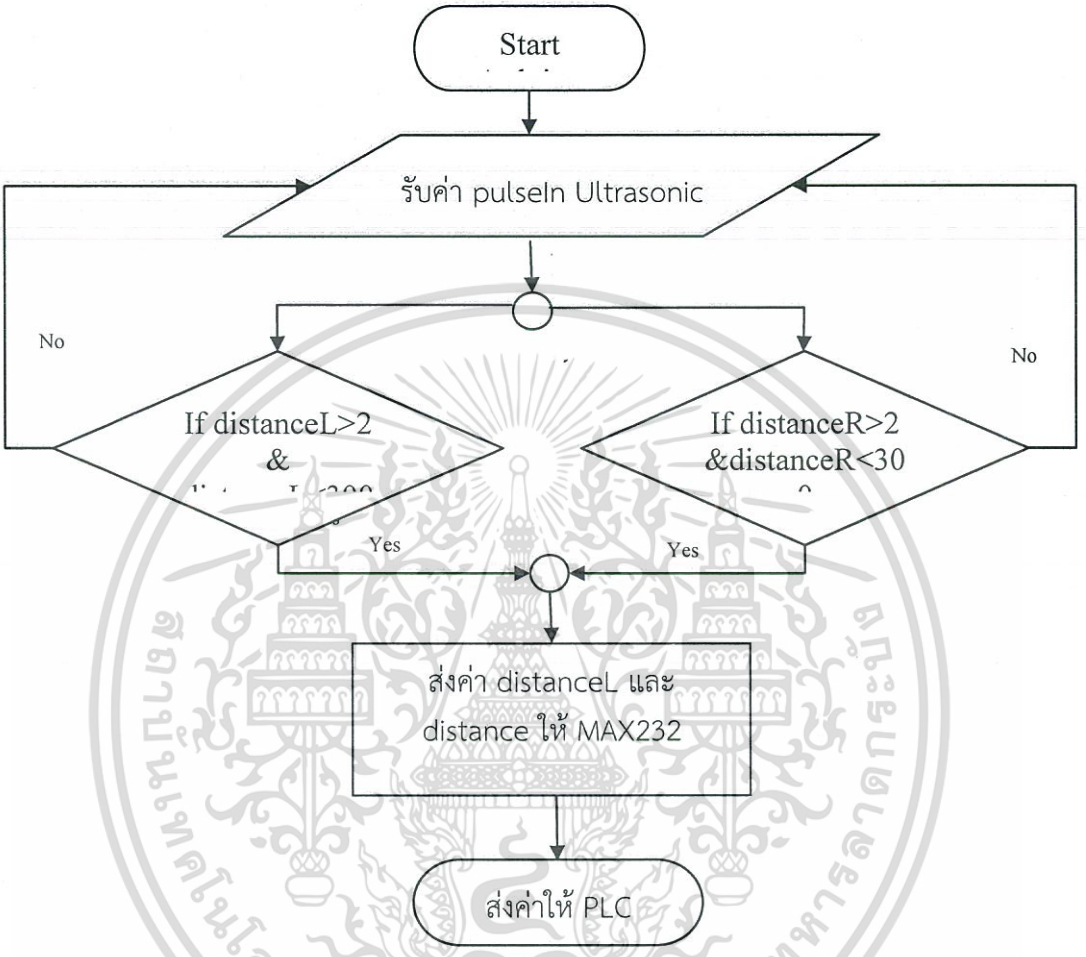
4.2.4.1 เซนเซอร์ ESP8266



ภาพที่ 4.7 รูปแสดง Flow Chart การทำงาน ESP8266-01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

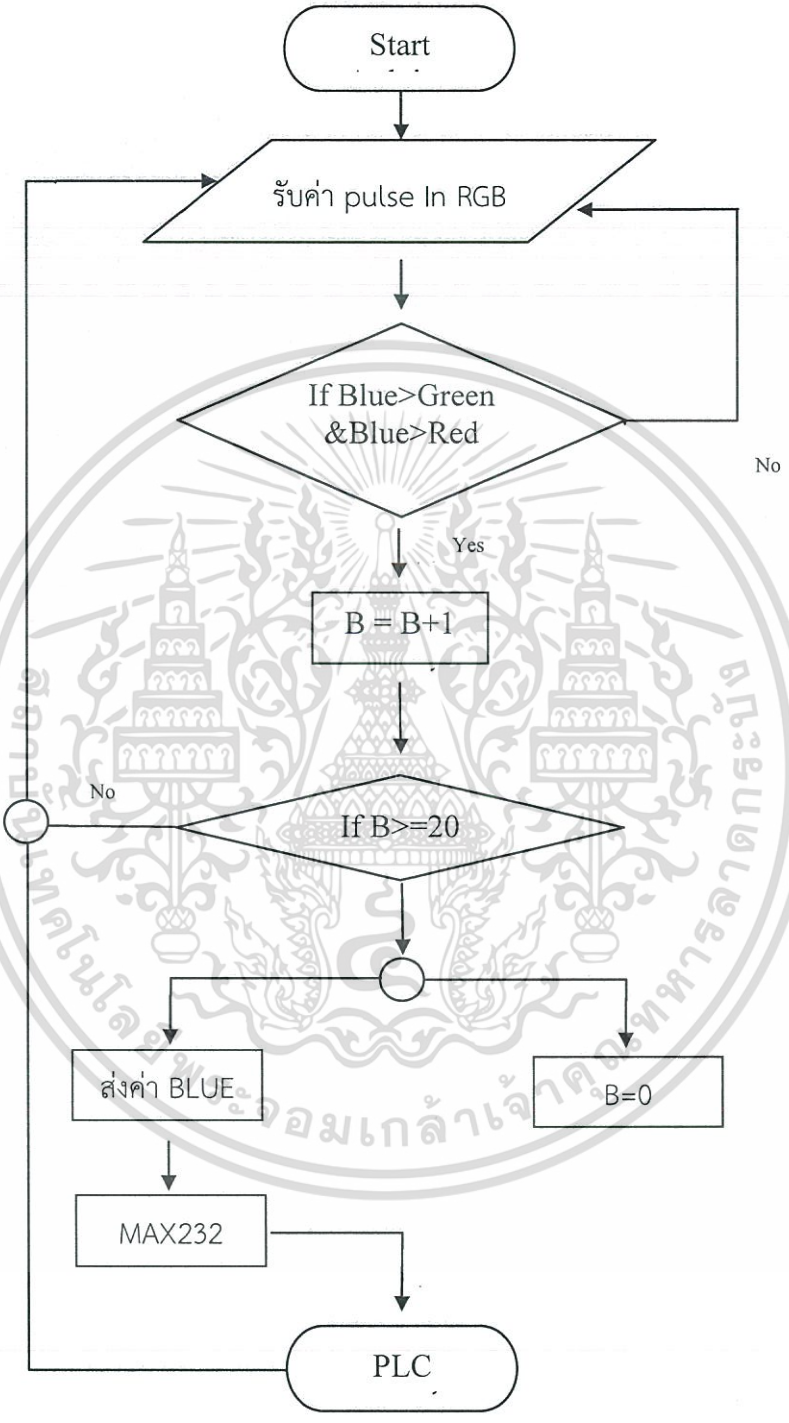
4.2.4.2เซนเซอร์ Ultrasonic



ภาพที่ 4.8 รูปแสดง Flow Chart การทำงาน Ultrasonic Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4.3เซนเซอร์ RGB



ภาพที่ 4.9 รูปแสดง Flow Chart การทำงาน RGB Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.5 ผลการทดลองตอนที่ 5: Test และ Calibration

ตารางที่ 4.7 แสดงผลจากความสัมพันธ์ระหว่างระยะที่วัดได้กับอุณหภูมิของ Ultrasonic Sensor

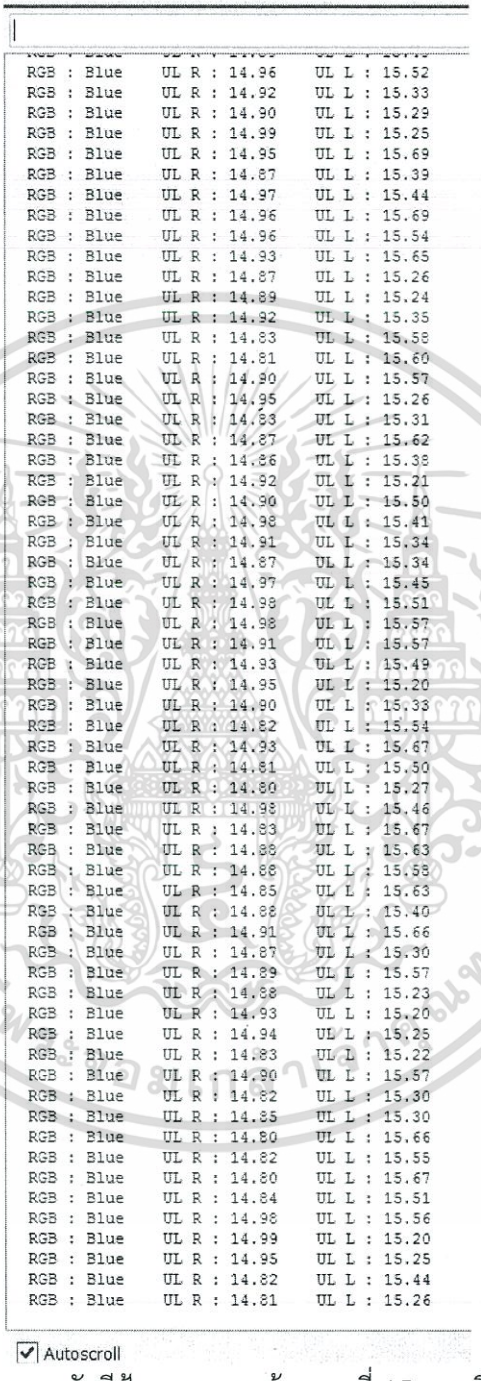
เนื่องจาก Ultrasonic sensor เป็นเซนเซอร์ที่ใช้คลื่นเสียงในการตรวจสอบเป็นหลัก ซึ่งอุณหภูมิจะส่งผลต่อความเร็วของคลื่นเสียงที่เซนเซอร์ปล่อยออกมา จะทำให้มีค่าความคลาดเคลื่อนได้ โดยการทดลองจะเปรียบเทียบอุณหภูมิที่ 25 องศาเซลเซียส และ 30 องศาเซลเซียส

อุณหภูมิ องศาเซลเซียส	ระยะจริง (cm)	ระยะที่วัดได้ (cm)
25	5	5.21
25	10	10.01
25	50	51.19
25	100	101.22
25	200	197.57
30	5	5.32
30	10	10.25
30	50	52.31
30	100	103.27
30	200	201.72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ทดสอบและเก็บผลการทดลอง

ในส่วนนี้จะเก็บผลการทดลองโดยกำหนดค่ามาตรฐานในการวัด ระยะในการวัด และอุณหภูมิในการวัด และสีที่ใช้



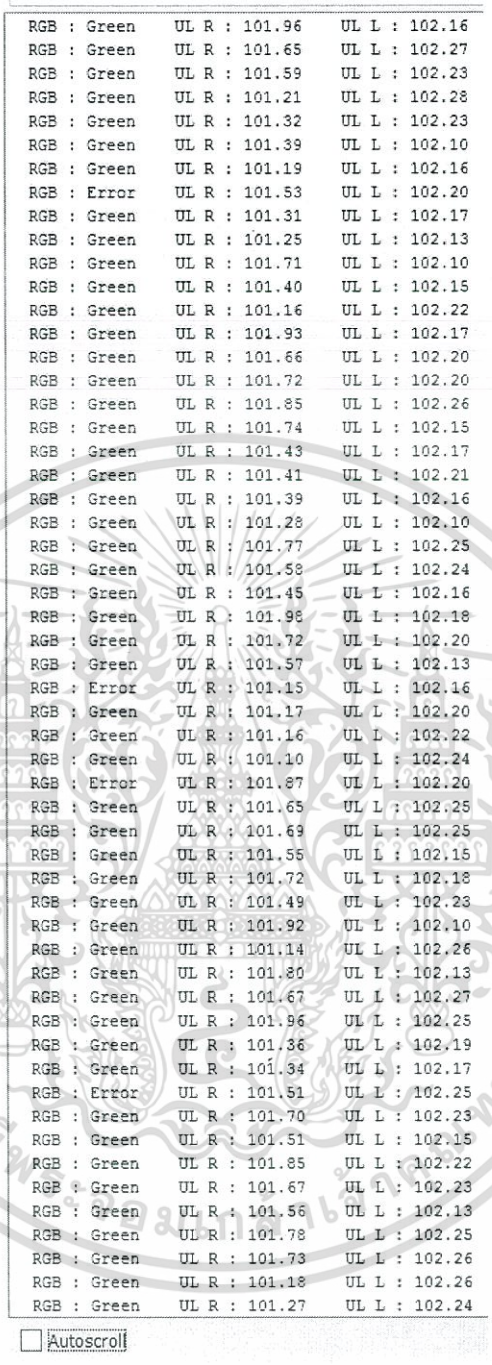
ภาพที่ 4.10 ผลการวัดสีฟ้า และระยะซ้ายขวาที่ 15 cm มีไฟส่องที่วัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
RGB : Red    UL R : 29.19    UL L : 30.67
RGB : Error    UL R : 29.35    UL L : 30.61
RGB : Red    UL R : 29.30    UL L : 30.64
RGB : Red    UL R : 29.34    UL L : 30.45
RGB : Red    UL R : 29.49    UL L : 30.50
RGB : Red    UL R : 29.67    UL L : 30.46
RGB : Red    UL R : 29.73    UL L : 30.54
RGB : Red    UL R : 29.64    UL L : 30.30
RGB : Red    UL R : 29.69    UL L : 30.40
RGB : Red    UL R : 29.56    UL L : 30.52
RGB : Red    UL R : 29.62    UL L : 30.33
RGB : Red    UL R : 29.80    UL L : 30.29
RGB : Red    UL R : 29.69    UL L : 30.25
RGB : Red    UL R : 29.55    UL L : 30.69
RGB : Red    UL R : 29.97    UL L : 30.39
RGB : Red    UL R : 29.57    UL L : 30.44
RGB : Red    UL R : 29.76    UL L : 30.69
RGB : Red    UL R : 29.36    UL L : 30.54
RGB : Red    UL R : 29.13    UL L : 30.65
RGB : Red    UL R : 29.47    UL L : 30.26
RGB : Red    UL R : 29.69    UL L : 30.24
RGB : Red    UL R : 29.32    UL L : 30.35
RGB : Error    UL R : 29.63    UL L : 30.58
RGB : Red    UL R : 29.41    UL L : 30.60
RGB : Red    UL R : 29.70    UL L : 30.57
RGB : Red    UL R : 29.35    UL L : 30.26
RGB : Error    UL R : 29.63    UL L : 30.31
RGB : Red    UL R : 29.97    UL L : 30.62
RGB : Red    UL R : 29.36    UL L : 30.38
RGB : Red    UL R : 29.92    UL L : 30.21
RGB : Red    UL R : 29.30    UL L : 30.50
RGB : Red    UL R : 29.68    UL L : 30.41
RGB : Red    UL R : 29.61    UL L : 30.34
RGB : Red    UL R : 29.77    UL L : 30.34
RGB : Red    UL R : 29.67    UL L : 30.45
RGB : Red    UL R : 29.58    UL L : 30.51
RGB : Red    UL R : 29.88    UL L : 30.57
RGB : Red    UL R : 29.31    UL L : 30.57
RGB : Red    UL R : 29.73    UL L : 30.49
RGB : Red    UL R : 29.55    UL L : 30.20
RGB : Red    UL R : 29.60    UL L : 30.33
RGB : Red    UL R : 29.62    UL L : 30.54
RGB : Red    UL R : 29.53    UL L : 30.67
RGB : Red    UL R : 29.11    UL L : 30.50
RGB : Red    UL R : 29.30    UL L : 30.27
RGB : Red    UL R : 29.98    UL L : 30.46
RGB : Red    UL R : 29.13    UL L : 30.67
RGB : Error    UL R : 29.88    UL L : 30.63
RGB : Red    UL R : 29.68    UL L : 30.58
RGB : Red    UL R : 29.35    UL L : 30.63
RGB : Red    UL R : 29.88    UL L : 30.40
RGB : Error    UL R : 29.81    UL L : 30.66
RGB : Red    UL R : 29.97    UL L : 30.30
RGB : Red    UL R : 29.29    UL L : 30.57
RGB : Red    UL R : 29.38    UL L : 30.23
RGB : Red    UL R : 29.33    UL L : 30.20
RGB : Red    UL R : 29.24    UL L : 30.25
RGB : Red    UL R : 29.33    UL L : 30.22
RGB : Red    UL R : 29.50    UL L : 30.57
RGB : Red    UL R : 29.62    UL L : 30.30
RGB : Red    UL R : 29.25    UL L : 30.30
RGB : Red    UL R : 29.10    UL L : 30.66
RGB : Red    UL R : 29.22    UL L : 30.55
```

ภาพที่ 4.11 ผลการวัดสีแดง และระยะซ้ายขวาที่ 30 cm ไม่มีไฟส่องที่วัตถุ

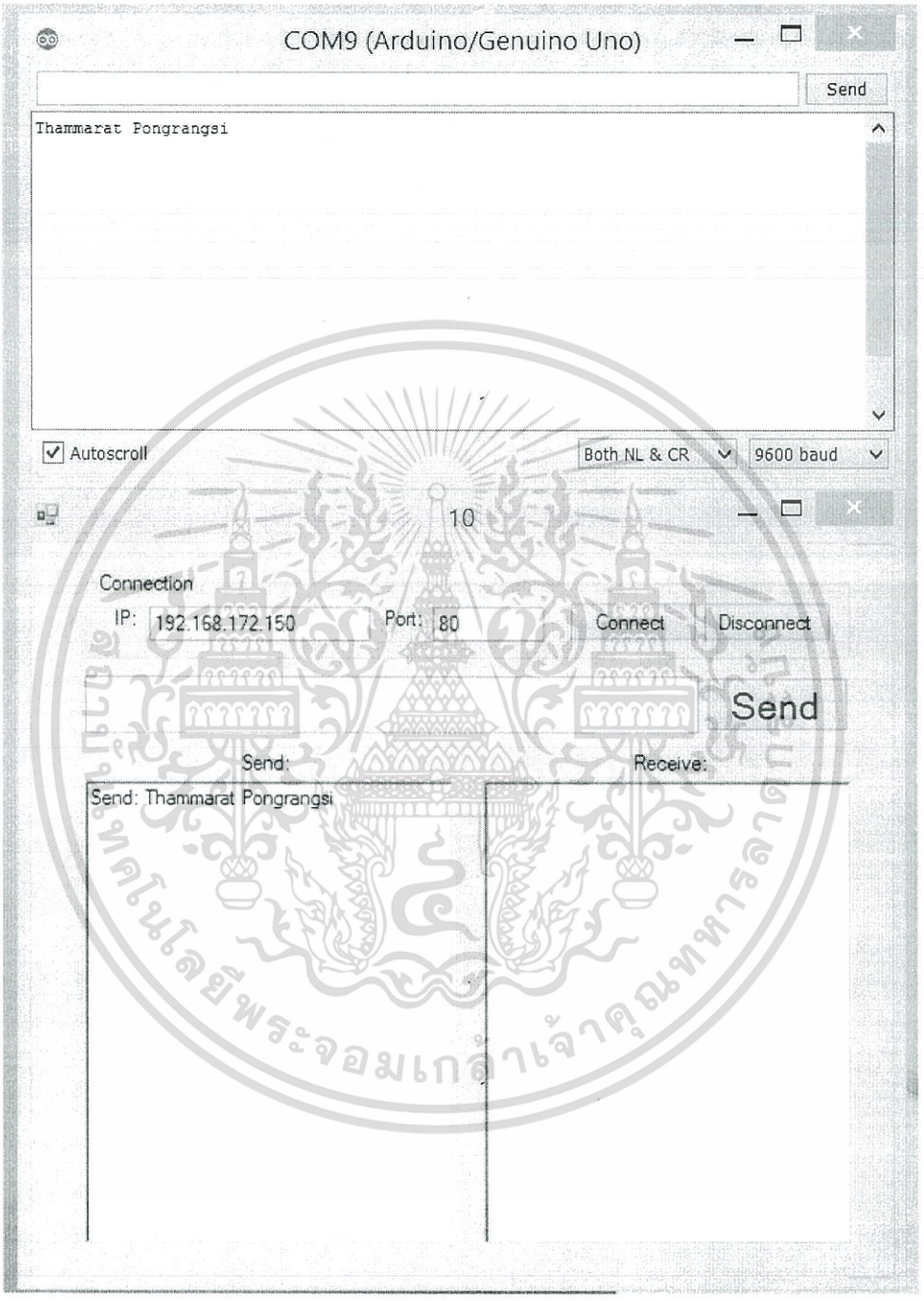
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.12 ผลการวัดสีเขียว และระยะซ้ายขวาที่ 100 cm ไม่มีไฟส่องที่วัตถุ

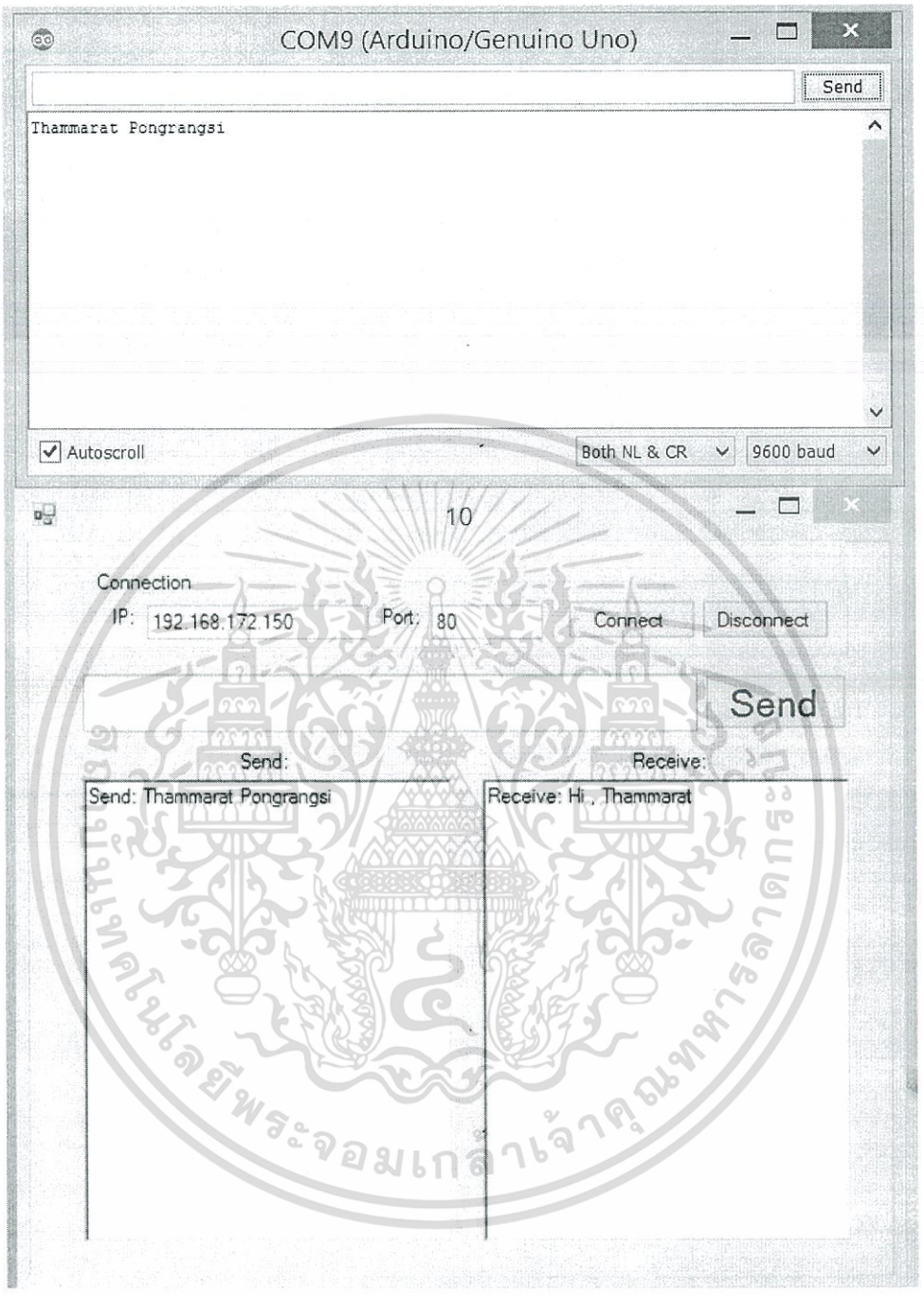
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนนี้จะเก็บผลการทดลองการเชื่อมต่อโมดูลไวไฟ กับ คอมพิวเตอร์ โดยโปรแกรม TT ซึ่งจะกำหนดระยะการเชื่อมต่อ IP Address และ Access Point ของ บริษัท



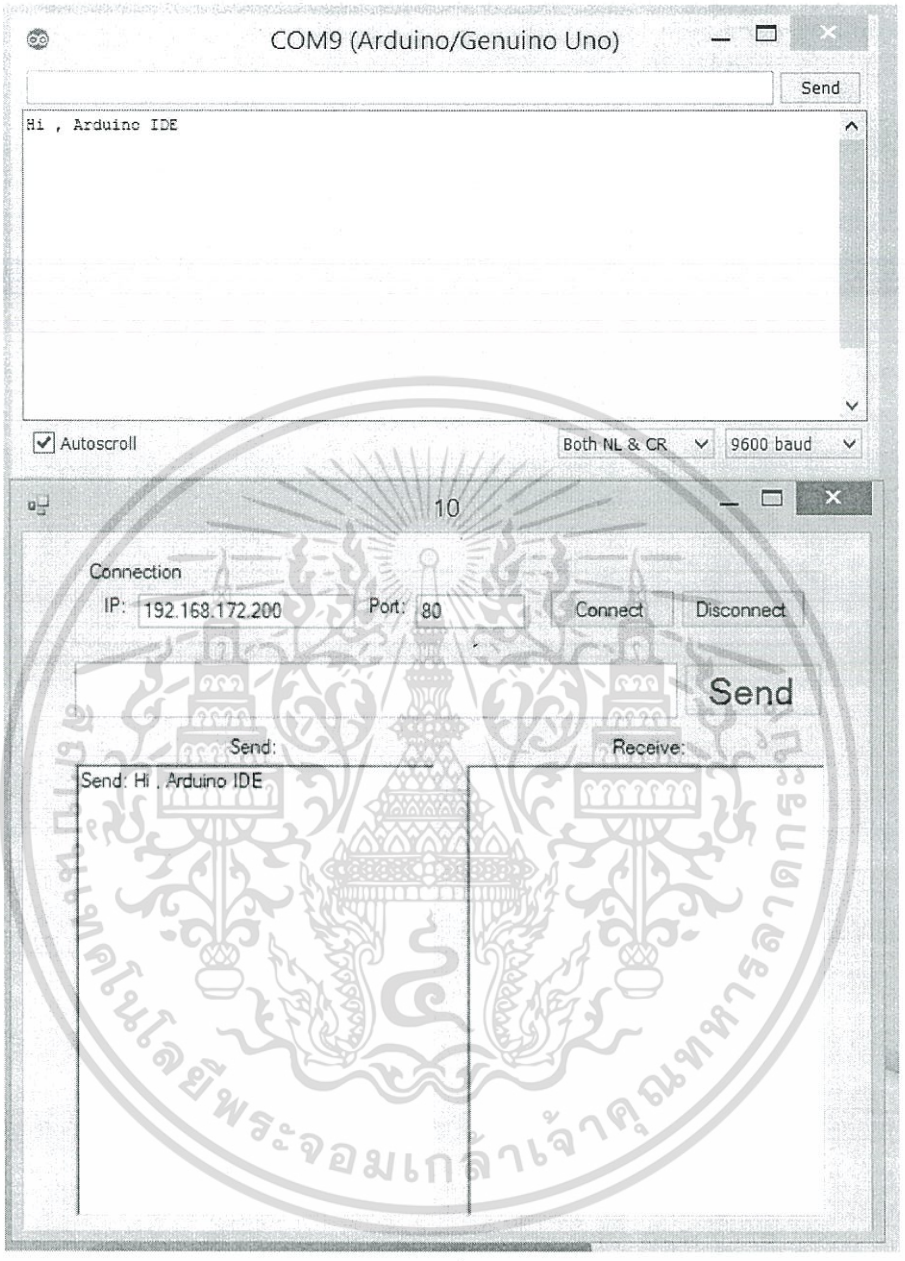
ภาพที่ 4.13 IP Address : 192.168.172.150 AP : Delta-Guest ส่งข้อความจาก TT Program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



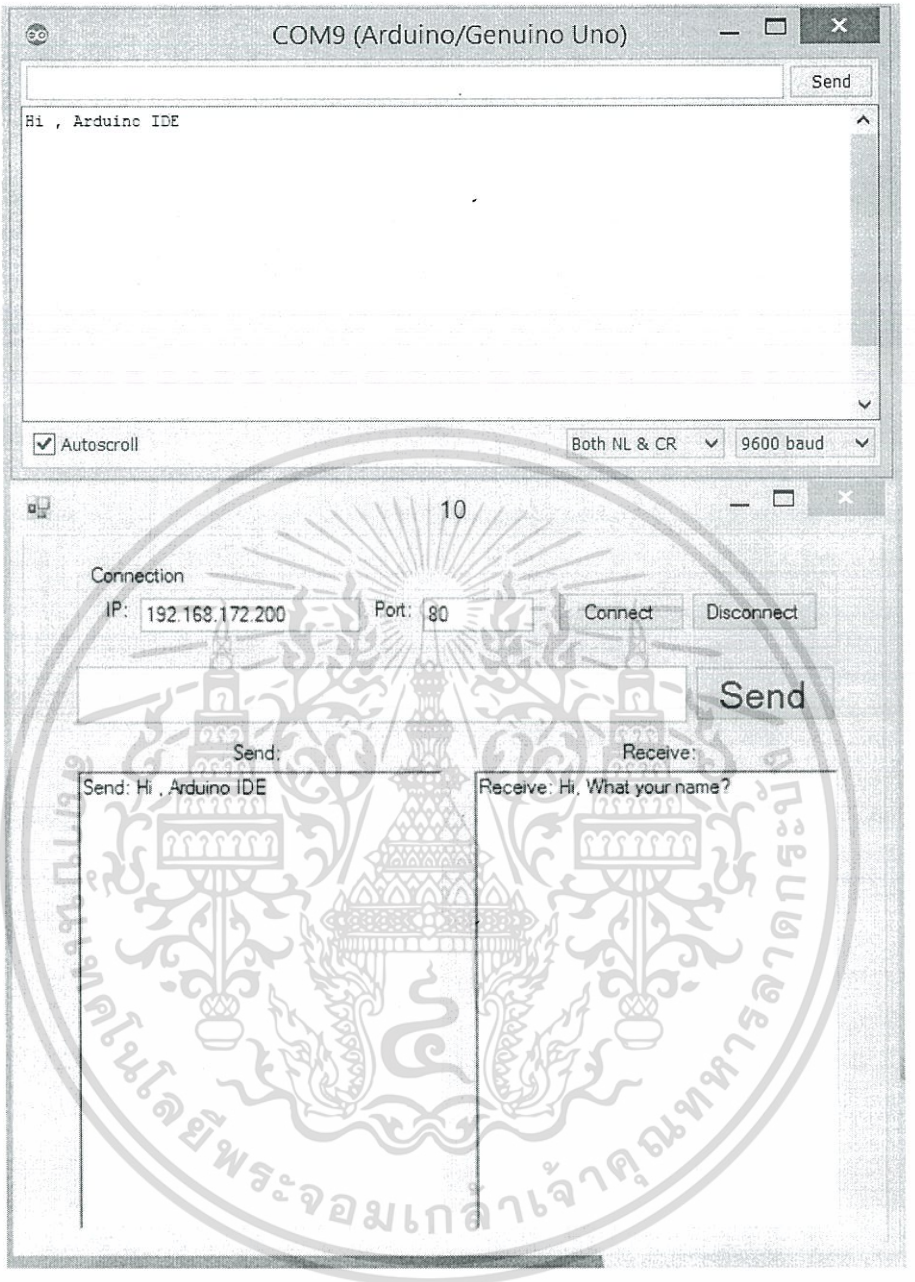
ภาพที่ 4.14 IP Address : 192.168.172.150 AP : Delta-Guest ส่งข้อความจาก Monitor IDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.15 IP Address : 192.168.172.200 AP : Delta-Guest ส่งข้อความจาก TT Program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.16 IP Address : 192.168.172. AP : Delta-Guest ส่งข้อความจาก Monitor I

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการดำเนินงาน และข้อเสนอแนะ

5.1 วิจารณ์ผลการทดลอง

จากการสังเกตและผลการทดลองทำให้พบว่าตัวเซนเซอร์ RGB และ Ultrasonic มีค่าความคลาดเคลื่อนบางและมีเอาท์พุทที่ไม่คงที่ อีกทั้งโปรเจคนี้ยังเป็นเพียงแค่จุดเริ่มต้นเพราะยังต้องนำไปประยุกต์อีกเพื่อให้สามารถเพิ่มประสิทธิ์ภาพและลดค่าความคลาดเคลื่อน

5.2 สรุปผลของการทดลอง

จากการผลการทดลองจะพบว่าที่เซนเซอร์ต่างๆมีความคลาดเคลื่อนบ้างเล็กน้อยซึ่งเกิดจากสภาพแวดล้อมเช่น RGB Sensor ยกตัวอย่างจากภาพที่ 4.8 และ 4.9 มี ค่า Error ในผล เนื่องจากมีคนเดินผ่านแล้วเงาตกที่วัตถุทำให้ค่าของสีที่เซนเซอร์ตรวจจับผิดเพี้ยนแต่ยังสามารถแก้ไขได้โดยการใช้หลอดไฟ LED ส่องที่วัตถุดังภาพที่ 4.7 ไม่มีค่า Error ที่เซนเซอร์ Ultrasonic มีค่าความคลาดเคลื่อนตามอุณหภูมิเนื่องจากมีผลต่อความเร็วเสียงที่เซนเซอร์ปล่อยยกตัวอย่างจากตารางที่ 4.7 มีการสวิงของค่าเอาท์พุทแต่อยู่ในขอบเขตที่พอรับได้เนื่องจากตัวเซนเซอร์จะทำงานอยู่ในห้องที่ควบคุมอุณหภูมิไว้ที่ 25-26 องศาเซลเซียส ซึ่งเปลี่ยนแปลงน้อยมากเมื่อเทียบกับการทดลองที่เปลี่ยนจาก 25 ไป 30 องศาเซลเซียส

5.2.1 สรุปผลการทดลองตอนที่ 1: ทดสอบระยะการวัดของ Ultrasonic sensor

จากหัวข้อการทดลองนี้ได้ทำการทดสอบการหาระยะจากการปล่อยคลื่นเสียงของ Ultrasonic sensor ซึ่งจากตารางที่ 4.1 ผลที่วัดได้มีค่าความคลาดเคลื่อนเล็กน้อย แต่เซนเซอร์จะไม่สามารถวัดได้ต่ำกว่า 3 เซนติเมตร และมากกว่า 300 เซนติเมตรเนื่องจากตัวเซนเซอร์มีสเปคในการวัดระยะที่ 2-300 เซนติเมตร

5.2.2 สรุปผลการทดลองตอนที่ 2: ทดสอบระยะการตรวจสีของ RGB Sensor

จากหัวข้อการทดลองนี้ได้ทำการทดสอบการวัดสีโดยเปลี่ยนระยะการตรวจที่ 1, 5, 10, 12 และ 15 เซนติเมตร จากตารางที่ 4.2, 4.3, 4.4, 4.5 และ 4.6 จะเห็นได้ว่าเซนเซอร์ RGB มีระยะการตรวจจับในช่วงที่ดีที่ 1 ถึง 10 เซนติเมตร ในช่วงที่ 12 เซนติเมตรขึ้นไป จะมีการตรวจจับผิดพลาดเนื่องจากเซนเซอร์จะใช้การสะท้อนของแสงที่ตกกระทบวัตถุมาหาความถี่ของคลื่นของแสงที่สีต่างๆ ถ้าวัตถุที่ทำการตรวจจับมีระยะห่างมากเกินไปแสงที่สะท้อนมากจากวัตถุจะถูกแสงสีอื่นๆรบกวนจนไม่สามารถหาค่าได้

5.2.3 สรุปผลการทดลองตอนที่3:เชื่อมต่อArduino ESP8266-01กับโปรแกรม TT ผ่าน ไวไฟ

จากหัวข้อการทดลองนี้ได้ทำการทดสอบการเชื่อมต่อ Arduino ESP8266-01 ผ่านไวไฟที่บริษัทโดยกำหนดค่า AP และ IP address ก่อนแล้ว โดยทำการเชื่อมต่อจาก computer ซึ่งใช้โปรแกรมชื่อ TT ที่ทำการเขียนขึ้นด้วย Visual C# จากการทดสอบเชื่อมต่อตัว Arduino ESP8266-01 สามารถเชื่อมต่อและส่งข้อความหากันได้ยกตัวอย่างจากภาพที่ 4.5 และภาพที่ 4.6 ที่ IP Address ต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.4 สรุปผลการทดลองตอนที่ 4: สร้างแผนผังโค้ด และเขียน โค้ดบนบอร์ด Arduino ESP8266-01

จากการทดลองนี้มีเพื่อทดลองการใช้งานบอร์ด Arduino ที่ทำงานร่วมกับเซนเซอร์ เนื่องจากผู้ทดลองไม่เคยใช้เซนเซอร์ RGB และ Ultrasonic มาก่อนจึงได้เริ่มทำการศึกษาการใช้งานโดยการเริ่มศึกษาการทำงานของเซนเซอร์ทั้งสองตัวตามตัวอย่างการใช้งานของเซนเซอร์นั้นๆ

ซึ่งลักษณะของการทำงานของ Flow Chart นี้ก็คือ จะทำการรับค่าสัญญาณพัลส์เข้ามาแล้วหาความถี่ของสัญญาณนั้นแล้วนำมาเปรียบเทียบตามดาต้าชีทของเซนเซอร์แล้วส่งค่าออกทางช่องสัญญาณ RX และ TX หัวข้อการทดลองนี้มีจุดมุ่งหมายที่จะศึกษาการทำงานของบอร์ด Arduino เพื่อที่จะสามารถออกแบบ Flow Chart สำหรับการเขียน code ภาษา C

5.2.5 สรุปผลการทดลองตอนที่ 5: Test และ Calibration

สำหรับการทดลองหัวข้อนี้สามารถสรุปผลได้ว่าหลังจากทำการต่อเซนเซอร์และทำการวัดค่าจริงค่าที่ได้มีความคลาดเคลื่อนจึงได้ทำการ calibrate ภายในของ code บางส่วน เพื่อให้ตรวจจับค่าให้ใกล้ค่าจริงมากที่สุด ยกตัวอย่างเช่นเซนเซอร์ Ultrasonic ตามสมการ

ระยะทาง = (Duration(uS)/2) * (Velocity(cm/uS))

จะได้ว่า distance = (duration*347)/(2*10^6) ที่อุณหภูมิ 25 องศาเซลเซียส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. ESP8266-01 Wireless Module
แหล่งที่มา : http://www.thaieasyelec.com/article-wiki/embedded-electronics-application/getting-started-with-esp8266-nodemcu.html (วันที่ค้นข้อมูล: 15 สิงหาคม 2559)
2. RGB Sensor
แหล่งที่มา : https://www.arduinoall.com/article (วันที่ค้นข้อมูล: 6 สิงหาคม 2559)
3. RS232
แหล่งที่มา : http://www.thaieasyelec.com/article-wiki/basic-electronics/uart-ttl-rs232-max232-max3232.html (วันที่ค้นข้อมูล: 7 สิงหาคม 2559)
4. Ultrasonic Sensor
แหล่งที่มา : http://www.elec2you.com/article/22/การใช้งาน-ultrasonic-hc-sr04
(วันที่ค้นข้อมูล: 18 สิงหาคม 2559)
5. Ultrasonic
แหล่งที่มา : http://www.compomax.co.th/product/ultrasonic-theory/
(วันที่ค้นข้อมูล: 14 สิงหาคม 2559)
6. Visual C#
แหล่งที่มา : http://www.vcharkarn.com/blog/43670 (วันที่ค้นข้อมูล: 3 กันยายน 2559)
7. สอนใช้งาน Ultrasonic Module HC-SR04 กับ arduino
แหล่งที่มา : http://www.myarduino.net/article/38/สอนใช้งาน-ultrasonic-hc-sr04-กับ-arduino-2 (วันที่ค้นข้อมูล: 18 สิงหาคม 2559)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
ซอร์สโค้ดสำหรับโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1 สำหรับ ESP8266-01

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#define SERVER_PORT 80                    //กำหนด Port ใช้งาน


IPAddress ip(192,168,172,99);             //เอาไว้ FIX ค่า IP
IPAddress gateway(192,168,168,254);       //ตั้งค่า IP Gateway
IPAddress subnet(255,255,248,0);          //ตั้งค่า Subnet
const char* ssid = "Delta-Guest";         //กำหนด SSID
const char* password = "";                //กำหนด Password

WiFiServer server(SERVER_PORT);           //สร้าง object server และกำหนด port
int x,tt ;
String re;
void setup()
{
   Serial.begin(9600);                 //ไม่ควรเปลี่ยน  //ส่งออก ไป Software Serial Arduino
   WiFi.mode(WIFI_STA);
   //Serial.println("");
   //Serial.println("");
   WiFi.begin(ssid, password);          //เชื่อมต่อกับ AP
   WiFi.config(ip,gateway,subnet);      //FIX ค่า IP Gateway Subnet
    while (WiFi.status() != WL_CONNECTED)  //รอการเชื่อมต่อ
   {
     delay(500);
     //Serial.print(".");
   }
   //Serial.println("WiFi connected");  //แสดงข้อความเชื่อมต่อสำเร็จ
   //Serial.println("IP address: ");
   Serial.println(WiFi.localIP());          //แสดงหมายเลข IP
   server.begin();                          //เริ่มต้นทำงาน TCP Server
   //Serial.println("Server started");     //แสดงข้อความ server เริ่มทำงาน
   x = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    tt = 0;
}

void loop()
{
  WiFiClient client = server.available();     //รอรับ การเชื่อมต่อจาก Client
  if (client)                                 //ตรวจเช็คว่ามี Client เชื่อมต่อเข้ามาหรือไม่
  {
    Serial.println("new client");             //แสดงข้อความว่ามี Client เชื่อมต่อเข้ามา
    while(1)
    {
     x = 0;
     String s;
     while(client.available())        //ตรวจเช็ตว่ามี Data ส่งมาจาก Client  หรือไม่
     {
       uint8_t t = client.read();    //อ่าน Data จาก Buffer
       char c = t;                   //เปลี่ยน char ไปเป็น string ส่งทาง serial.print
       s += c;
       x = 1;
     }
       if(x == 1)
       {
         Serial.println(s);         //ส่งค่าออกทาง tx ให้ Softwareserial ของ Arduino
         x = 0;
       }
     if(server.hasClient())         //ตรวจเช็คว่ายังมี  Client เชื่อมต่ออยู่หรือไม่
     {
      return;                       //ถ้าไม่มีให้ออกจาก ลูป ไปเริ่มต้นรอรับ Client ใหม่
     }
     /*********************************************************************************/



     String gg;
     String gt;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      int gy = 0;
      while(Serial.available()>0)     // รับค่าจาก ขา rx แล้วส่งให้ client ส่งไปยัง Visual C#
      {
        char xy = Serial.read();
        delayMicroseconds(1500);
        gg += xy;
        gy = 1;
      }
      if(gy == 1)
      {
       delayMicroseconds(10);
       client.print(gg);
       gy = 0;
      }
      Serial.flush();
      /*************************************************************************************/
    }
  }
}
```

1.2 สำหรับ Arduino

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(6, 7);          // RX, TX
int x,r;

#define echoPinL 2                    // L 3
#define trigPinL 3                   // L 2
#define echoPinR 4                    // R
#define trigPinR 5                   // R
int maximum, minimum ;
float durationL, distanceL;
float durationR, distanceR;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
const int s0 = 9;
const int s1 = 8;
const int s2 = 12;
const int s3 = 11;
const int out = 13;
const int nLED = 10;
int red,green,blue,Red,Green,Blue,B,G,R ;

void setup()
{
  Serial.begin(9600);
  pinMode(s0, OUTPUT);
  pinMode(s1, OUTPUT);
  pinMode(s2, OUTPUT);
  pinMode(s3, OUTPUT);
  pinMode(nLED, OUTPUT);
  pinMode(out, INPUT);
  pinMode(trigPinL, OUTPUT);
  pinMode(echoPinL, INPUT);
  pinMode(trigPinR, OUTPUT);
  pinMode(echoPinR, INPUT);
  digitalWrite(nLED,1);
  digitalWrite(s0, 1);
  digitalWrite(s1, 1);
  maximum = 300;                    // ultrasonic วัดไกลกว่านี้ไม่ได้ จะมีค่า error สูง
  minimum = 2;                      // ultrasonic วัดได้ต่ำสุด 2 cm
  red = 0;
  green = 0;
  blue = 0;
  Red = 0;
  Green = 0;
  Blue = 0;
  B=0;
  G=0;
  R=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  x = 0;
  mySerial.begin(9600);
}

void loop()
{
  color();
  Ifcolor();
  Mesure();
  IfUltra();
  ESP();
}

void ESP()
{
  String s;
  String ss;
  x=0;
  while (mySerial.available())
  {
    delayMicroseconds(1000);
    uint8_t t = mySerial.read();
    char c = t;
    if(x==0)
    {
      ss += "";
    }
    s += c;
    ss += c;
    x = 1;
  }
  r = 0;
  String a,aa;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if(x == 1)
  {
    Serial.print(s);    // รับข้อความ จาก esp แล้วส่งไป PLC
    x = 0;
  }
  while(Serial.available())
  {
    delayMicroseconds(1000);
    uint8_t q = Serial.read();
    char w = q;
    if(r == 0)
    {
      aa += "";
    }
    a += w;
    aa += w;
    r = 1;
  }
  if(r == 1)
  {
    mySerial.print(a);    // รับข้อความ จาก esp แล้วส่งไป PLC
    r = 0;
  }
}

void color()
{
  digitalWrite(s2, LOW);
  digitalWrite(s3, LOW);
  delayMicroseconds(15);
  red = pulseIn(out, HIGH , 1000);
  digitalWrite(s3, HIGH);
  delayMicroseconds(15);
  blue = pulseIn(out, HIGH , 1000);
  digitalWrite(s2, HIGH);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  delayMicroseconds(15);
  green = pulseIn(out, HIGH , 1000);
  Blue = map(blue, 300, 3, 0, 255);
  Green = map(green, 350, 3, 0, 255);
  Red = map(red, 400, 3, 0, 255);
  if(Blue<=0)
  {
    Blue = 0;
  }
  if(Green<=0)
  {
    Green = 0;
  }
  if(Red<=0)
  {
    Red = 0;
  }
  if(Red>180 && Blue >200 && Green >200)
  {
    Red = 0;
    Blue = 0 ;
    Green = 0;
  }
}

void Ifcolor()
{
 if (red < blue && red < green && red <50 && green > 100 && blue >100)
  {
   Serial.println(" -  Red ");
  }
if (Blue > Red && Blue > Green && red-blue > 80 && green-blue > 80)
 {
   B = B+1;
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  else if (Green > Red && Green > Blue )
  {
   Serial.println("  Green ");
  }
  if (B>=20)
  {
    Serial.print("Blue");
    delay(10);
    B=0;
  }
}

void Mesure()
{
  digitalWrite(trigPinL, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPinL, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPinL, LOW);
  delayMicroseconds(10);
  durationL = pulseIn(echoPinL, HIGH);
  digitalWrite(trigPinR, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPinR, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPinR, LOW);
  durationR = pulseIn(echoPinR, HIGH);
  distanceL = durationL/28.84/2.00;  // 340m/s = 29.4Microsecond/cm at 20 Celsius
  distanceR = durationR/28.84/2.00;  // 347m/s = 28.8Microsecond/cm at 25 Celsius
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void IfUltra()
{
  if(distanceL <= minimum || distanceL >= maximum)
  {
    Serial.print("EL");
  }
  else
  {
    Serial.print(distanceL);
  }
  if (distanceR >= maximum || distanceR <= minimum)
  {
    Serial.print("ER");
  }
   else
  {
   Serial.print(distanceR);
  }
}
```

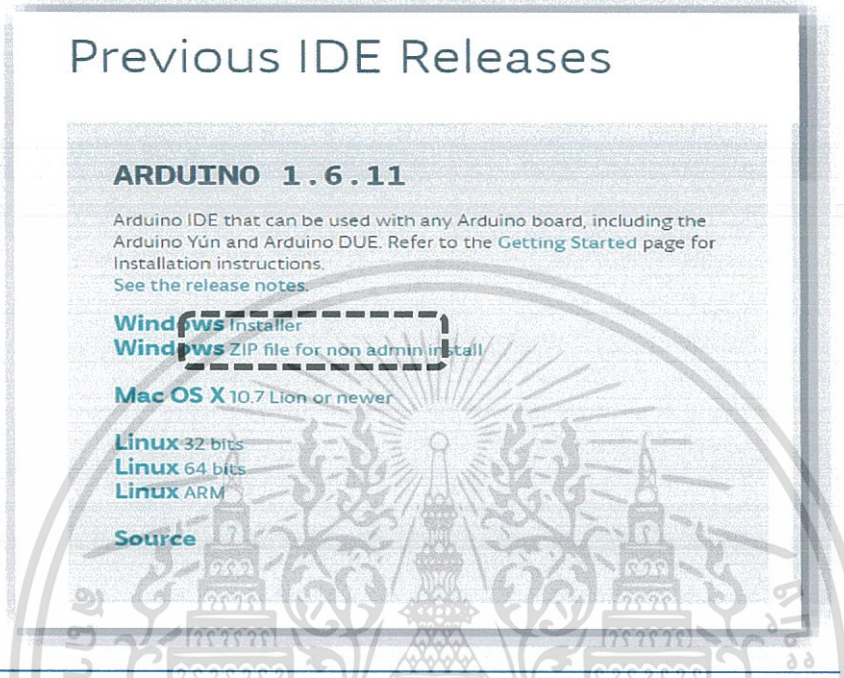
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
คู่มือวิธีการลงโปรแกรม Arduino ESP-8266

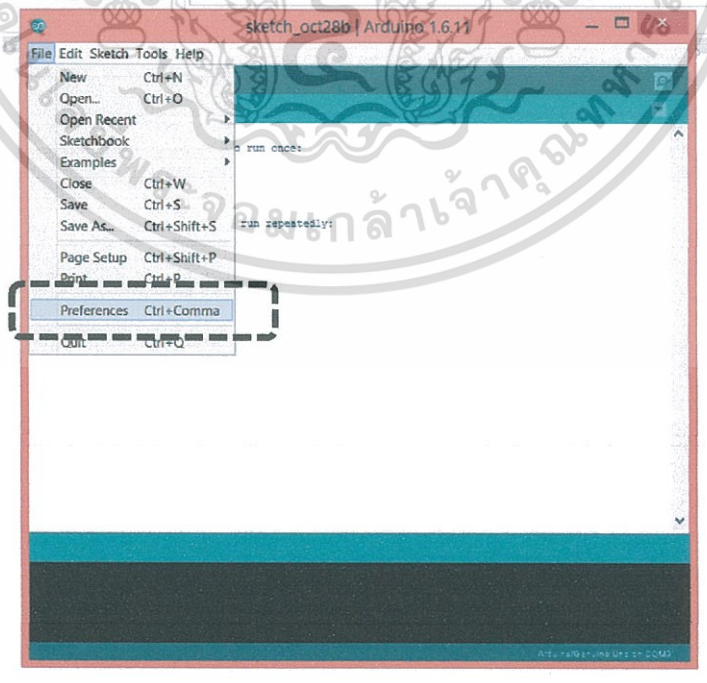
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการ Burn Program Arduino ESP-8266

1. ติดตั้ง โปรแกรม Arduino IDE 1.6.11 จากเว็ป
   https://www.arduino.cc/en/Main/OldSoftwareReleases#previous

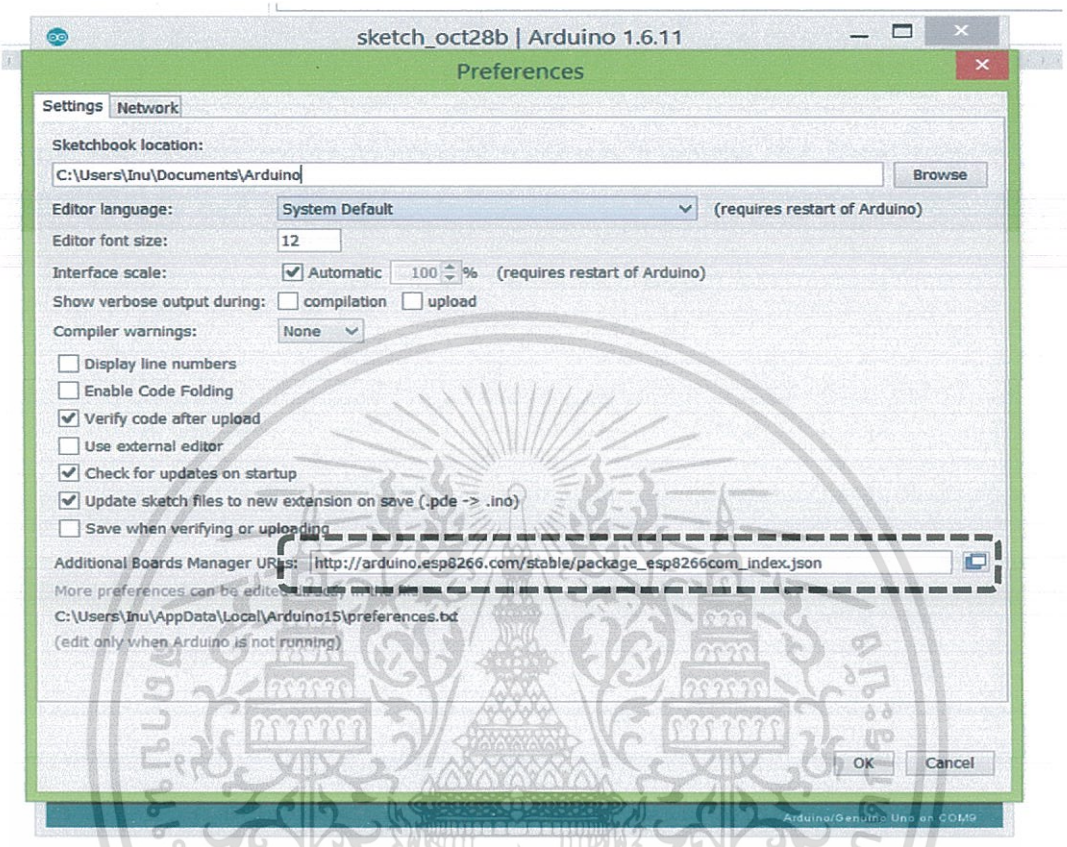


2. เมื่อทำการติดตั้ง Arduino IDE เรียบร้อยแล้ว ให้เปิด Arduino IDE ขึ้นมา ไปที่ Menu File >> Preferences

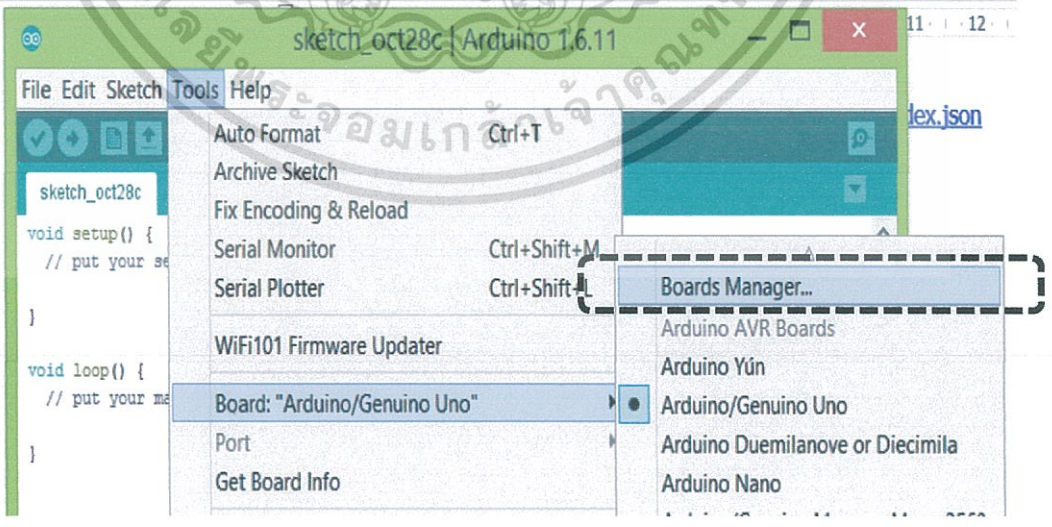


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ใส่ URL >> ลงใน Addition Board Manager URLs: ดังนี้
   http://arduino.esp8266.com/stable/package_esp8266com_index.json

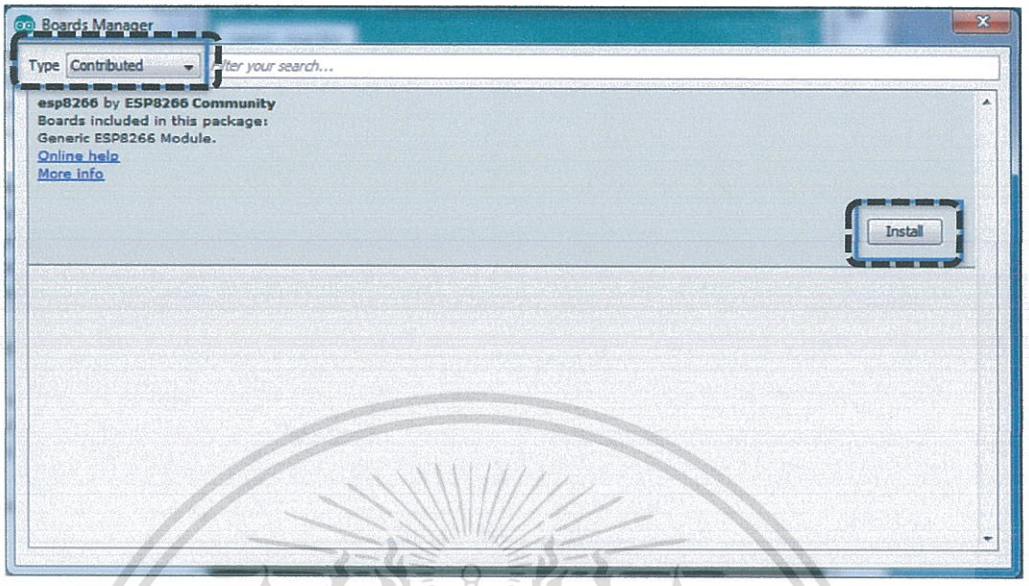


4. แล้วกด OK ไปที่ Menu Tools >> Board:"xxxxxx" >> Board Manager...

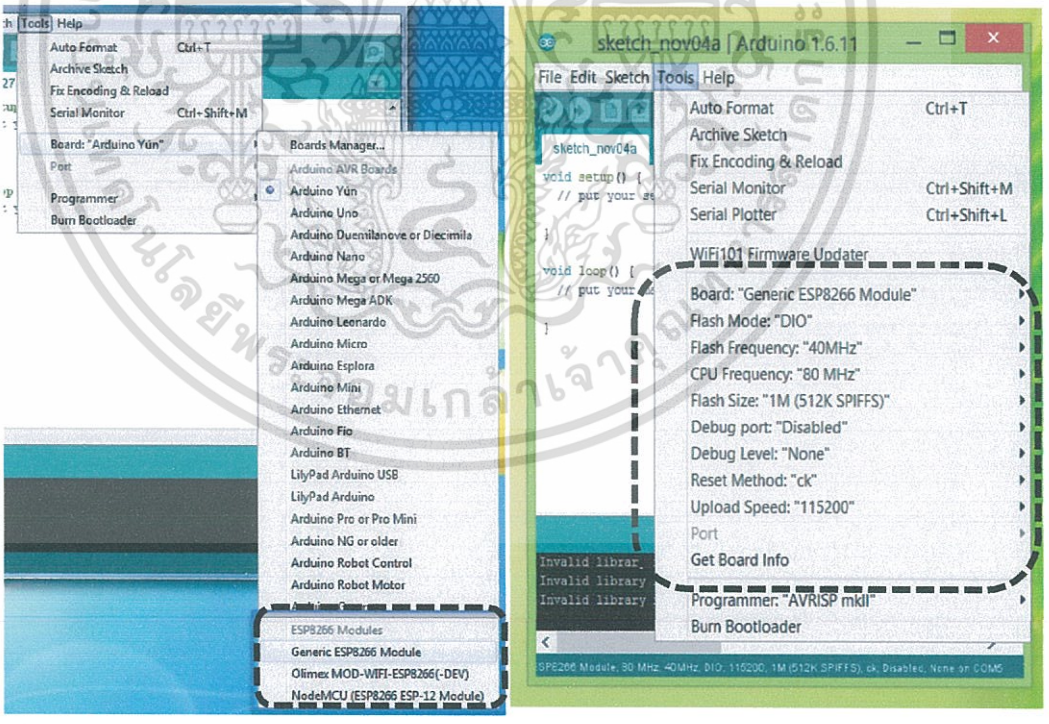


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เลือก Type เป็น Contributed แล้ว Search ไปที่ ESP8266 และกด Install รอจนติดตั้งเสร็จ

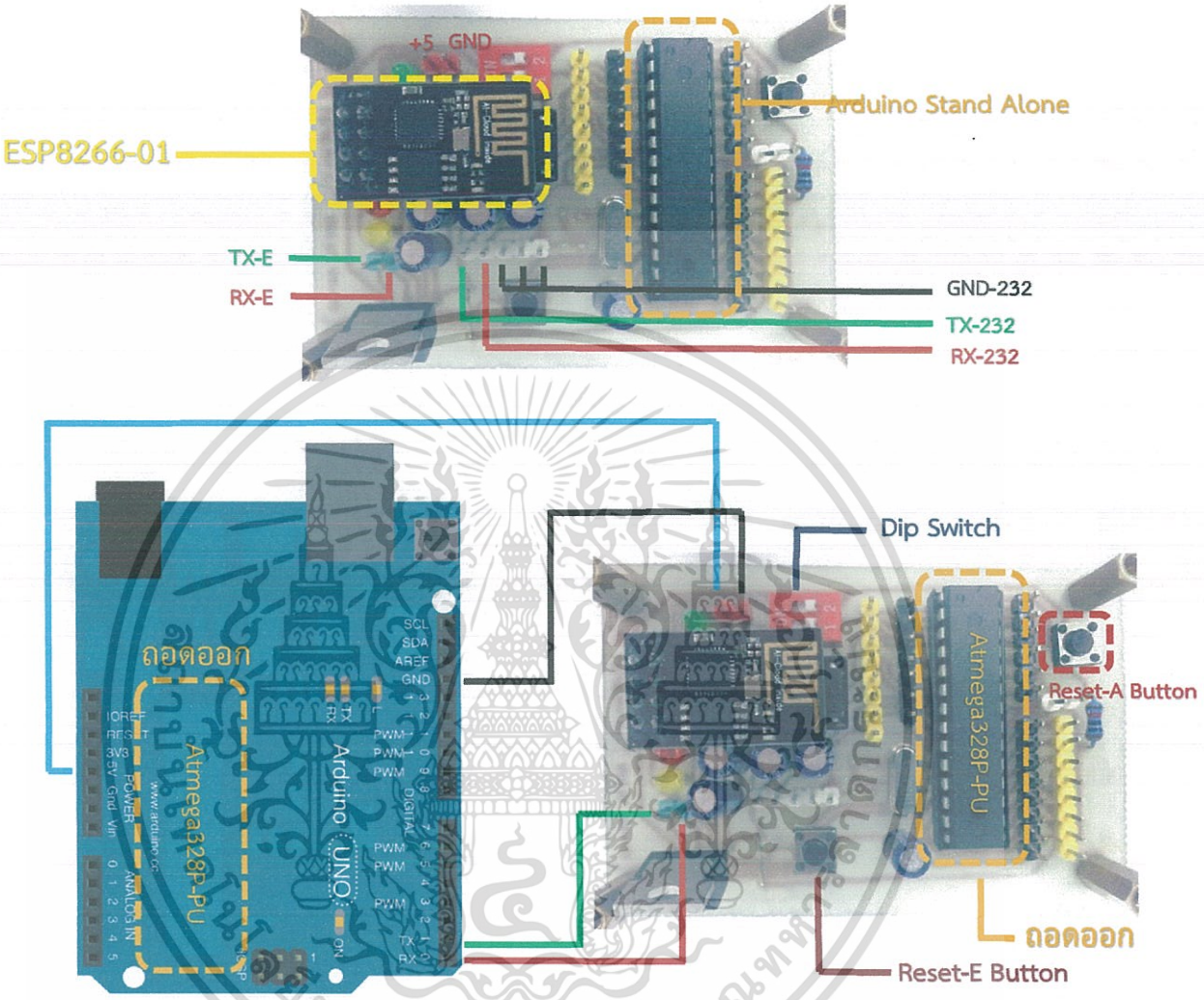


6. เมื่อติดตั้ง ESP8266 เสร็จเรียบร้อยแล้ว ให้ปิดโปรแกรม Arduino IDE ก่อน แล้วจึงเปิดขึ้นมาใหม่ จากนั้นจะพบ board ESP ในโปรแกรม โดยเลือก Generic ESP8266 Module



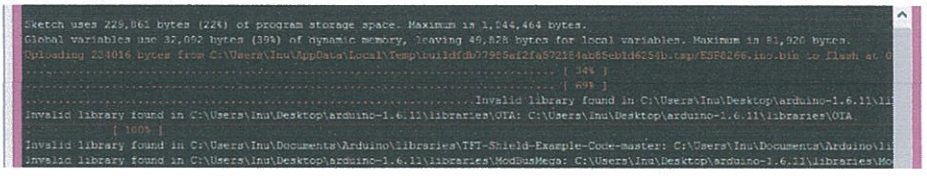
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. การเชื่อมต่อการลง Program ESP8266



8. - จากนั้นต่อวงจรดังข้อ 7 แล้วเลือก Board >> Generic ESP8266 Module
- ถอด IC Atmega328P-Pu ของบอร์ด Arduino Uno ออก และ (กดปุ่ม Reset-A Button ค้างไว้จนกว่าจะลงโปรแกรมเสร็จ หรือ ถอด Arduino ESP-8266 ออก)

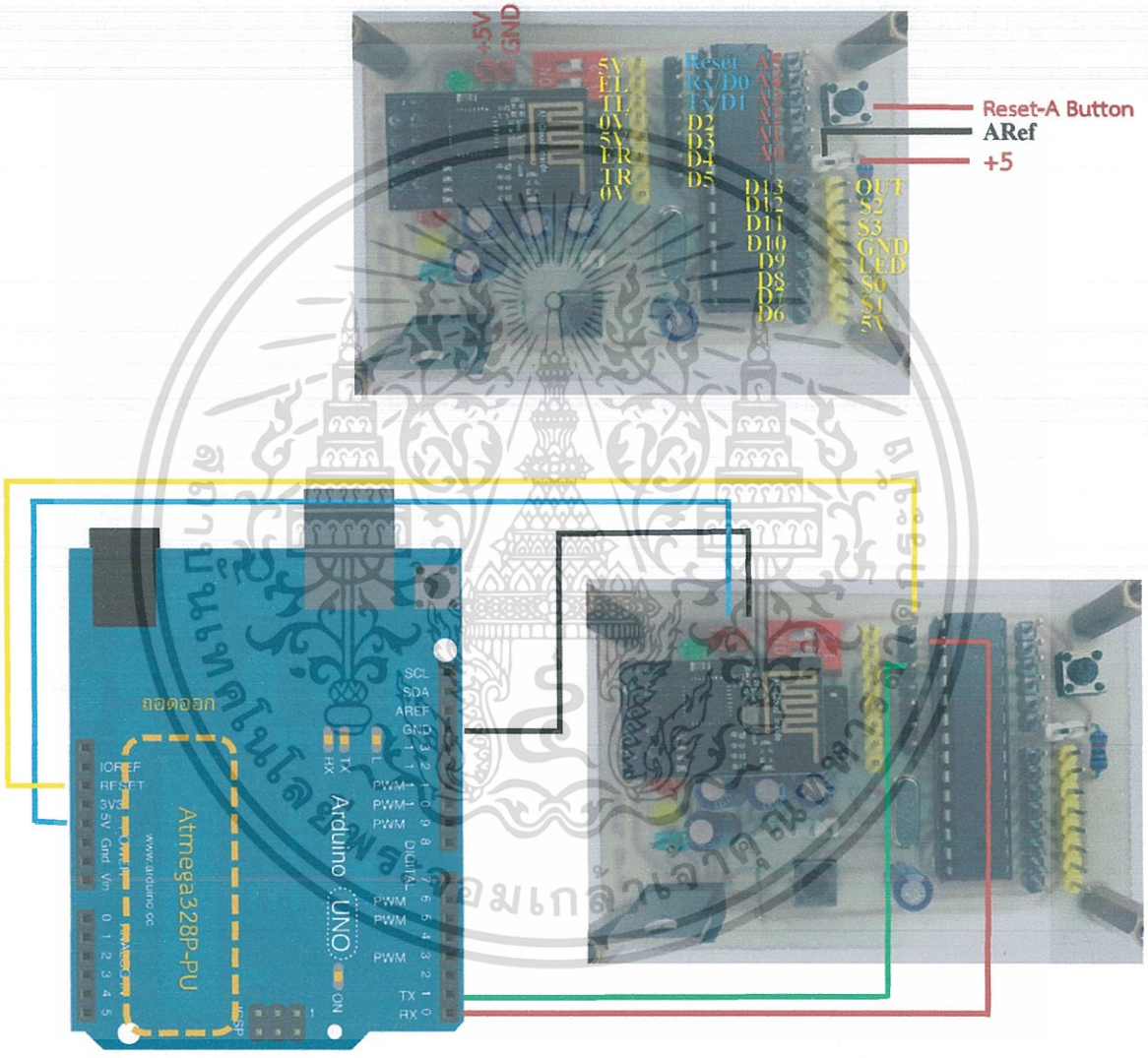
TX-Arduino	=> TX-E	RX-Arduino	=> RX-E
GND-Arduino	=> GND	5V-Arduino	=> +5V



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เลื่อน Dip Switch ช่อง 2 ไป ON จากนั้น กด Reset-E Button 1 ครั้ง แล้วกด Upload
- เมื่อลงโปรแกรมเสร็จ เลื่อน Dip Switch ช่อง 2 ไป OFF แล้วกด Reset-E Button 1 ครั้ง

9. ลงโปรแกรม Arduino Uno Stand Alone



- เชื่อม ต่อวงจร ดังรูป โดยถอด IC Atmega328P-PU บนบอร์ด Arduino Uno

TX-Arduino => TX-Arduino Stand Alone

RX-Arduino => RX- Arduino Stand Alone

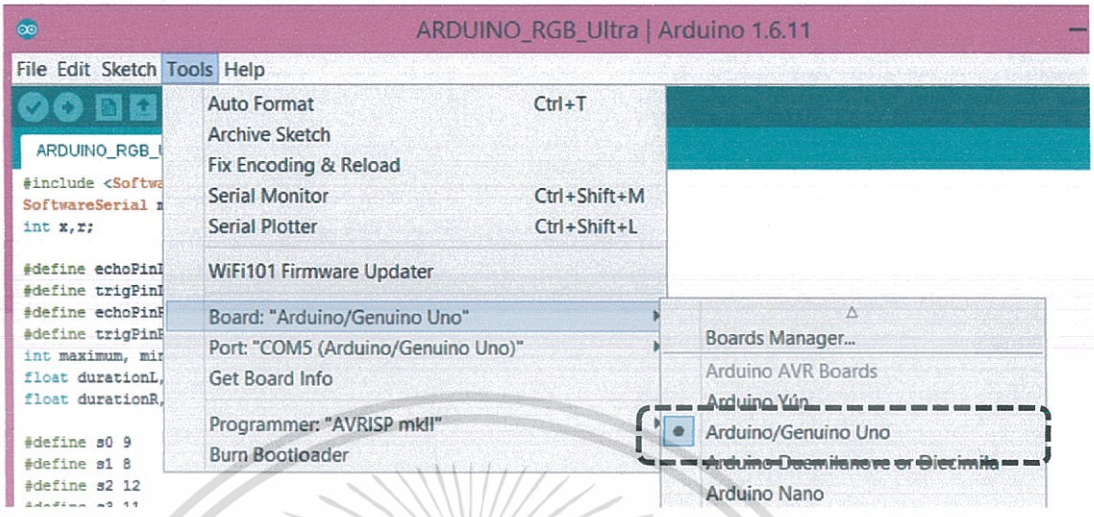
Reset-Arduino => Reset- Arduino Stand Alone

GND-Arduino => GND 5V-Arduino => +5V

- ไปที่โปรแกรม Arduino IDE 1.6.11 ไปที่ Tools >> Board >> Arduino/Genuino Uno

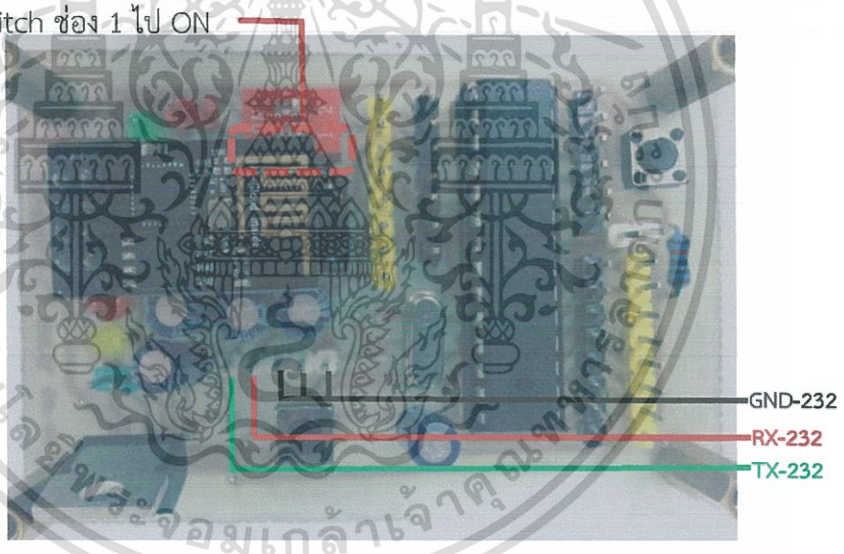
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เลือก Port แล้ว กด Upload โปรแกรม



10. การใช้งาน RS232 ของ Arduino ESP Stand Alone

- เลื่อน Dip Switch ช่อง 1 ไป ON



เชื่อมต่อสายสัญญาณ

RX- 232 => TX-Device-EXT.

TX- 232 => RX-Device-EXT.

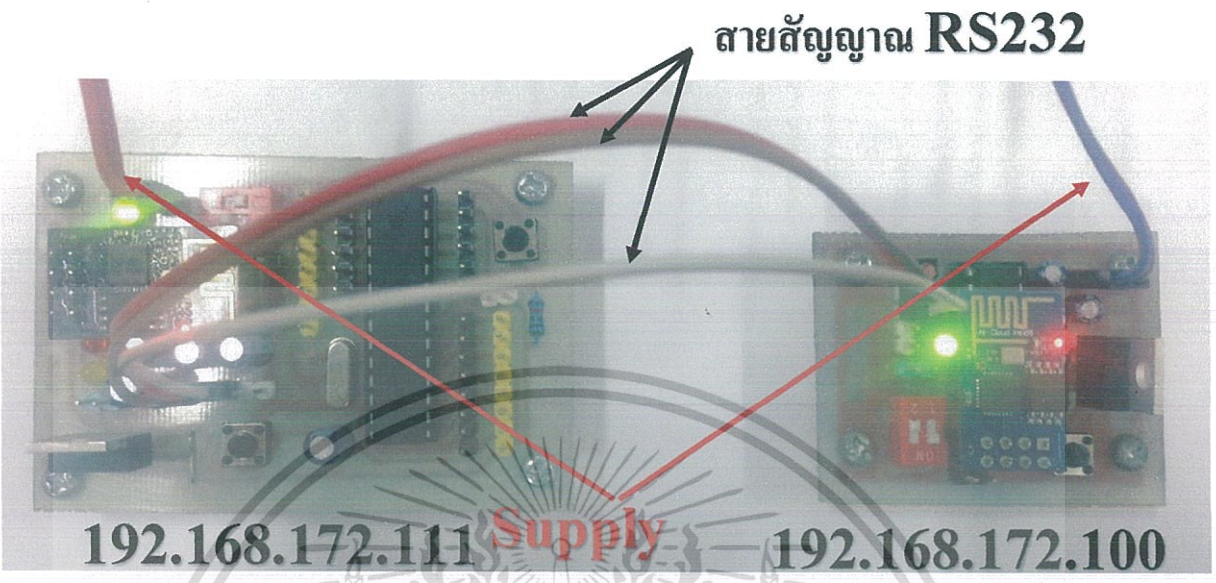
GND-232 => GND-Device-EXT.

- Burn Program = ESP8266.ino ใส่ ESP8266 บนบอร์ด Arduino ESP-8266

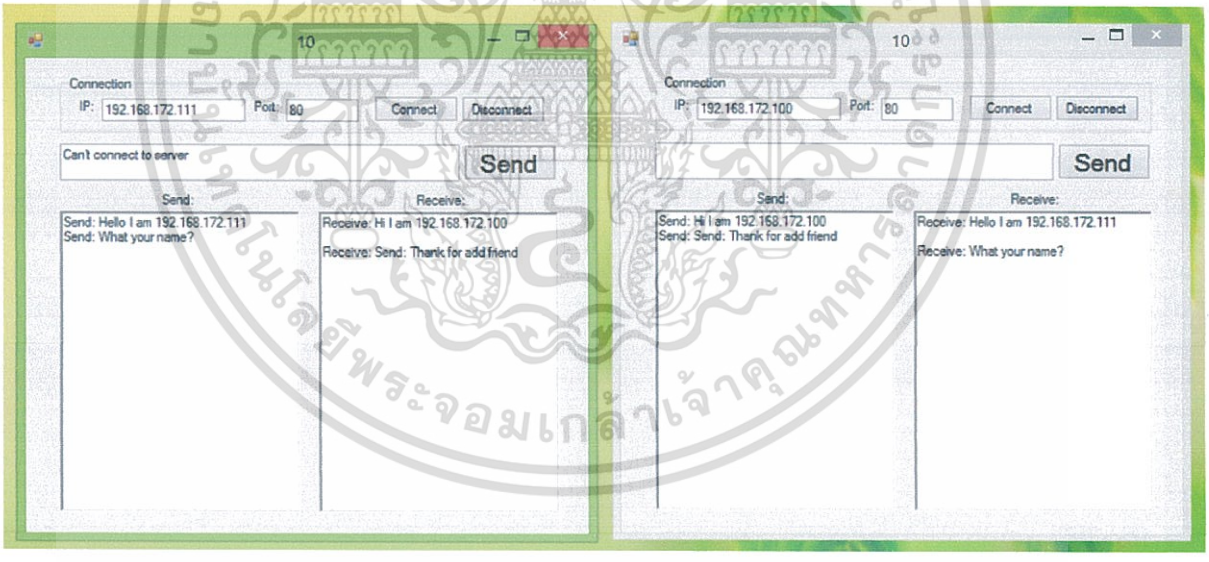
- Burn Program = ARDUINO_RGB_Ultra.ino ใส่ Arduino Stand Alone บนบอร์ด Arduino ESP-8266

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการเชื่อมต่อ ผ่าน RS232

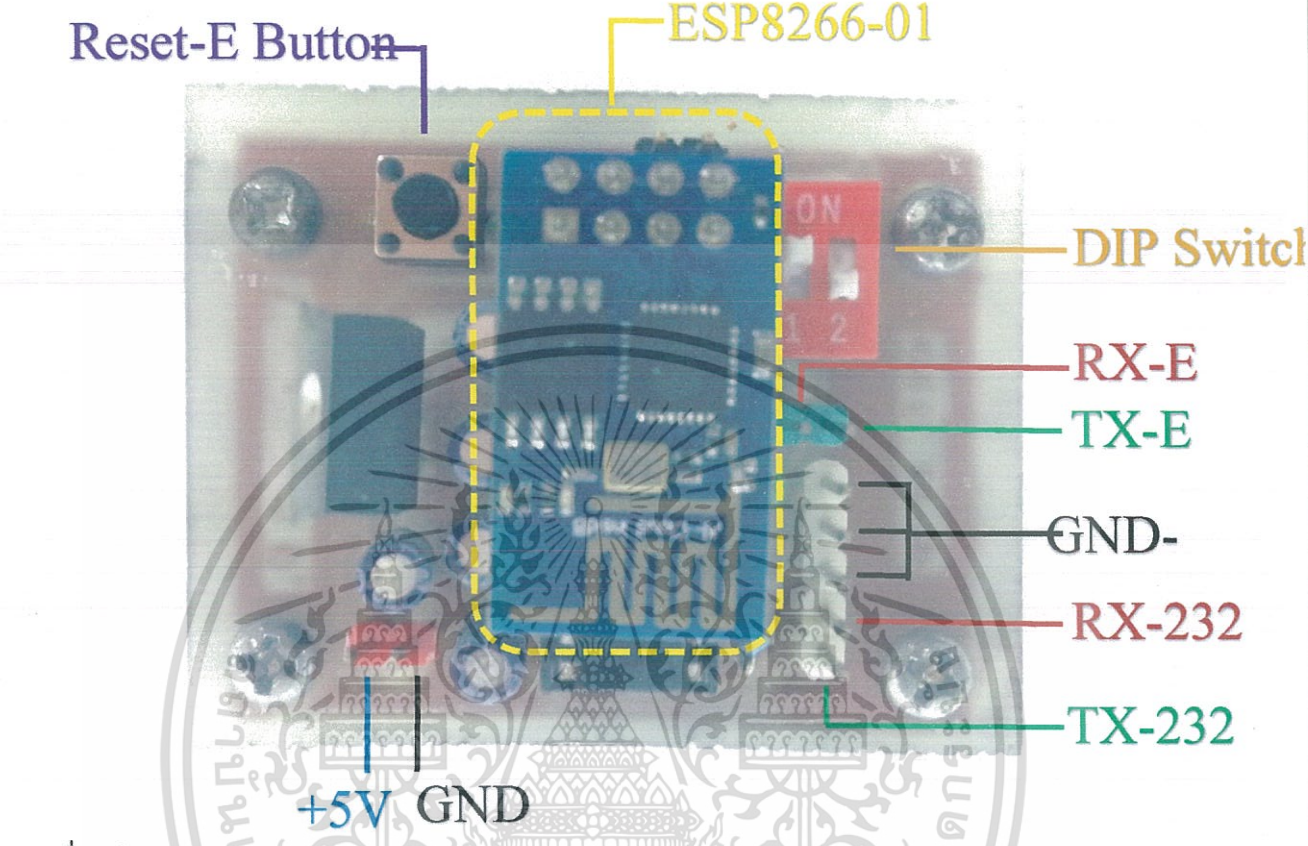


ผลการทดลองส่งข้อความ

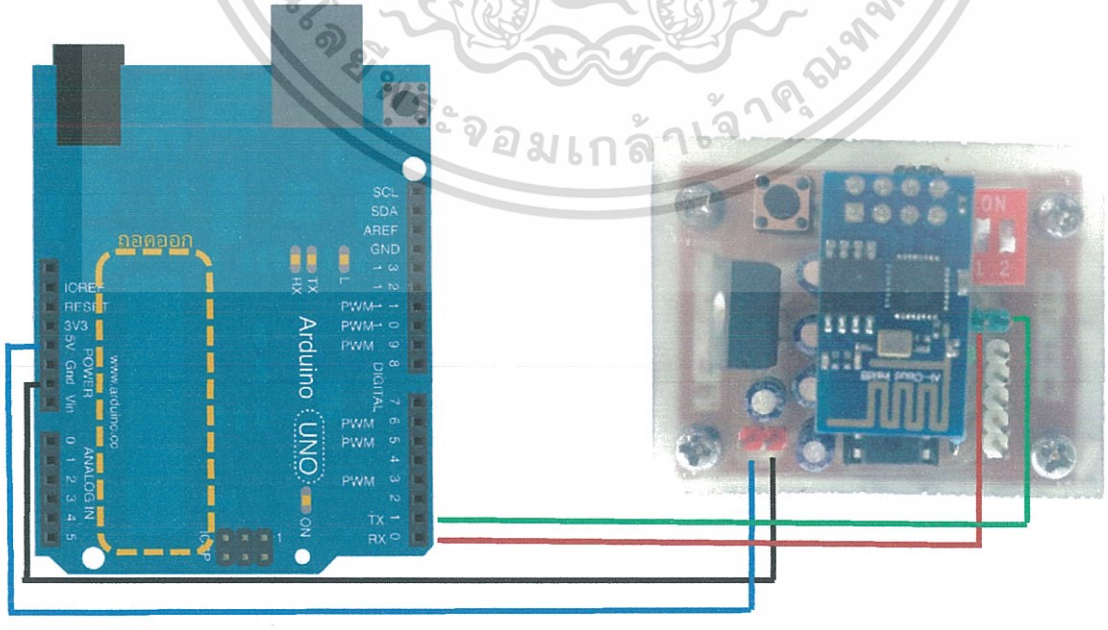


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ESP-8266-RS232



การเชื่อมต่อ และการ Burn



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นต่อวงจรดังข้อ 7 แล้วเลือก Board >> Generic ESP8266 Module เลือก port
เชื่อมต่อวงจรดังรูป เลื่อน Dip Switch ช่อง 2 ไป ON กด Reset-E Button 1 ครั้ง แล้วกด Burn
เมื่อลงโปรแกรมเสร็จ เลื่อน Dip Switch ช่อง 2 ไป OFF กด Reset-E Button 1 ครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ – สกุล นายธรรมรัตน์ ปงรังษี
Mr.THAMMARAT PONGRANGSI
วัน เดือน ปี เกิด 26 ธันวาคม พ.ศ.2537
ที่อยู่ปัจจุบัน 99/200 ตำบลเสม็ด อำเภอเมืองชลบุรี จังหวัดชลบุรี 20000
E-mail popeye261237@gmail.com
ประวัติการศึกษา กำลังศึกษาระดับปริญญาตรี หลักสูตรวิศวกรรมศาสตรบัณฑิต (วศ.บ.)
สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประสบการณ์การทำงาน พ.ศ.2559 ฝึกงาน บริษัท มิตซูบิชิ เฮฟวี่ อินดัสตรี่ส์-มหาจักร แอร์ คอนดิชั่นเนอร์ส จำกัด 220 นิคมอุตสาหกรรมลาดกระบัง เขตประกอบการเสรี 3 ซอยฉลองกรุง 31 แขวงลำปลาทิว เขตลาดกระบัง กรุงเทพมหานคร 10520

พ.ศ.2559 ฝึกงานสหกิจ บริษัท เดลต้า อีเลคทรอนิคส์(ประเทศไทย) จำกัด(มหาชน) เลขที่ 909 ซอย 9 หมู่ 4 นิคมอุตสาหกรรมบางปู ถ.พัฒนา 1 ตำบลแพรกษา อำเภอเมือง จังหวัดสมุทรปราการ 10280

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้