รายงานสหกิจศึกษาฉบับสมบูรณ์

อุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตของ PLC ระยะไกลด้วยความถี่ 2.4 GHz
PLC Wireless Remote I/O with 2.4 GHz

นายชนสรณ์ สุกใส

ภาควิชาวิศวกรรมการวัดและควบคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2561

รายงานสหกิจศึกษาฉบับสมบูรณ์

อุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตของ PLC ระยะไกลด้วยความถี่ 2.4 GHz

PLC Wireless Remote I/O with 2.4 GHz

นายชนสรณ์ สุกใส

ภาควิชาวิศวกรรมการวัดและควบคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงงานสหกิจศึกษา อุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตของ PLC ระยะไกลด้วยความถี่ 2.4 GHz
ชื่อ-สกุล นักศึกษา นายชนสรณ์ สุกใส
คณะ วิศวกรรมศาสตร์ **ภาควิชา** วิศวกรรมการวัดและควบคุม
ชื่อ-สกุล อาจารย์นิเทศ รศ.ดร.เกษตร์ ศิริสันติสัมฤทธิ์
ชื่อ-สกุล ผู้นิเทศงาน นายสุรพรรณ วงศ์สวัสดิชาติ
ชื่อสถานประกอบการ บริษัท อุตสาหกรรมทำเครื่องแก้วไทย จำกัด (มหาชน)

บทคัดย่อ

โครงงานสหกิจศึกษานี้นำเสนออุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตของ PLC ระยะไกลด้วยความถี่ 2.4 GHz วัตถุประสงค์ในการจัดทำเนื่องจากทางแผนกวิศวกรรมหน้ารางวางแผนจะสร้างฐานข้อมูล โดยเก็บข้อมูลอินพุต/เอาต์พุตของเครื่องจักรต่างๆที่อยู่ในความรับผิดชอบของแผนกฯไปวิเคราะห์ เพื่อเพิ่มประสิทธิภาพในการทำงานของเครื่องจักรดีขึ้น ดังนั้นทางบริษัทฯจึงมอบหมายให้ผู้วิจัยสร้างเครื่องต้นแบบของอุปกรณ์ที่จะใช้รับส่งข้อมูลอินพุต/เอาต์พุตของเครื่องจักร และสามารถรับส่งข้อมูลบัตรพนักงานซ่อมบำรุงเพื่อให้สามารถระบุตัวพนักงานที่ซ่อมหรือแก้ไขเครื่องจักรได้

เครื่องต้นแบบประกอบไปด้วย 1) อาดุยโน่ 2 ตัว ซึ่งใช้เป็นไมโครคอนโทรลเลอร์ 2) อาร์เอฟไอดี อาร์ซี 522 เป็นโมดูลอ่านข้อมูลบัตรพนักงาน 3) เอ็นอาร์เอฟ 24 แอล 01 จำนวน 2 ตัว ซึ่งใช้เป็นโมดูลที่ใช้รับส่งข้อมูลอินพุต/เอาต์พุตระยะไกลแบบไร้สายระหว่างอาดุยโน่ นอกจากนี้ยังมีการใช้โปรแกรมต่างๆ ดังนี้ 1) โปรแกรม Arduino IDE ใช้ในการเขียนให้อาดุยโน่ ใช้ในการรับและส่งข้อมูลต่างๆ 2) โปรแกรม Proteus-8 ใช้ในการทดสอบวงจรอินพุต/เอาต์พุตของตัวอุปกรณ์ 3) โปรแกรม PCBwizard ใช้ในการใช้ในการออกแบบวงจรอินพุต/เอาต์พุตของตัวอุปกรณ์ หลังจากนั้นนำไปทดลองและใช้รับส่งข้อมูลอินพุต/เอาต์พุตผลที่ได้คือสามารถรับส่งข้อมูลได้จริง

คำสำคัญ: อาดุยโน่, เอ็นอาร์เอฟ 24 แอล 01, อาร์เอฟไอดี อาร์ซี 522

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Co-operative Title: PLC Wireless Remote I/O with 2.4 GHz

Student Intern Name: Mr. Chanasorn Suksai

Faculty: Engineering **Department:** Instrumentation and Control Engineering

Advisor Name: Assoc. Prof. Dr. Kaset Sirisantisamrid

Mentor Name: Mr. Suraphan Wongsawatdichart

Company: Thai Glass Industries Public Company Limited

ABSTRACT

This cooperative project presents PLC Wireless Remote I/O with 2.4 GHz. The purpose of this project is due to the department of cool end process has plan to manage database by receiving input and output data of machines in department to analyze and increase the performance of machine. Thereby, the company requires to produce a prototype device for receiving input and output data of machine and the name of maintenance engineer to identify name who was performed to repair the machine.

The prototype device consists of 1) 2 units of Arduino, they are as microcontroller 2) RFID RC522 for reading employee card to identify the maintenance's name 3) 2 units of nRF24L01, they are a module of transmission wireless data I/O to transmit the data between two Arduino. Moreover, it needs many programs consist of 1) Arduino IDE program to write the program to Arduino 2) Proteus 8 program for testing the I/O circuit of device 3) PCBwizard program that used for designing the circuit of device. On experiment, the device can send and receive data between two Arduino.

Keyword: Arduino, nRF24L01, RFID RC522

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานฉบับนี้ได้รับการสนับสนุนจาก บริษัท อุตสาหกรรมทำเครื่องแก้วไทย จำกัด (มหาชน) เพื่อสร้างโอกาสที่ดีให้กับนักศึกษาในโครงการสหกิจศึกษา ในเรื่องของการเรียนรู้ และฝึกการทำงานจากการทำงานจริง ขอขอบพระคุณบุคลากรทุกท่านในบริษัท อุตสาหกรรมทำเครื่องแก้วไทย จำกัด (มหาชน) อาทิ เช่น คุณสุรพรรณ วงศ์สวัสดิชาติ ที่ให้โอกาสนักศึกษาได้เข้ามาสร้างประสบการณ์การทำงานจริง และพี่ ๆ ในแผนกวิศวกรรมหน้ารางทุกคนที่คอยให้คำแนะนำ และถ่ายทอดวิชาความรู้รวมถึงประสบการณ์การในการทำงานใหม่ ๆ ให้อยู่เสมอ ซึ่งเป็นสิ่งที่อยู่นอกเหนือจากการศึกษาในห้องเรียนหรือจากในหนังสือเล่มใด

สุดท้ายนี้ ขอขอบพระคุณคณาจารย์ในโครงการสหกิจศึกษา รศ. ดร. เกษตร์ ศิริสันติสัมฤทธิ์ ที่เปิดโอกาสให้นักศึกษาได้เรียนรู้การทำงานจากการทำงานจริง รวมถึงคอยให้คำปรึกษาในปัญหาต่างๆ ที่พบในขณะปฏิบัติงานในโครงการสหกิจศึกษา ตลอดจนช่วยอ่านและแก้ไขรายงานฉบับนี้จนกระทั่งเสร็จสมบูรณ์

นายชนสรณ์ สุกใส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทที่ ..... หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VI

สารบัญรูป VII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญ 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตของโครงงาน 2

1.4 วิธีดำเนินการทำโครงงาน 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 2

บทที่ 2 วัสดุอุปกรณ์ และโปรแกรมที่เกี่ยวข้อง 3

2.1 กล่าวนำ 3

2.2 โมดูลสื่อสารไร้สาย รุ่น nRF24L01 3

2.3 Arduino 4

2.4 RFID 7

2.5 โปรแกรม Proteus 8 professional 8

2.6 โปรแกรม PCBWizard 9

บทที่ 3 วิธีการดำเนินงาน 10

3.1 กล่าวนำ 10

3.2 วัสดุอุปกรณ์ 10

3.2.1 Arduino Nano 10

3.2.2 nRF24L01 11

3.2.3 RFID-RC522 13

3.3 วงจรอินพุต เอาต์พุต 14

3.3.1 Slave 14

3.3.2 Master 16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

บทที่ ..... **หน้า**

3.4 โปรแกรม 17

3.4.1 Slave 17

3.4.2 Master 20

3.5 ประกอบอุปกรณ์ 23

3.5.1 Slave 23

3.5.2 Master 24

บทที่ 4 ผลการดำเนินงาน 26

4.1 กล่าวนำ 26

4.2 การทดสอบ 26

4.2.1 ทดสอบดิจิตอลอินพุตจาก Slave ส่งไปยัง Master 26

4.2.2 ทดสอบดิจิตอลเอาต์พุตของ Slave ที่รับมาจาก Master 29

4.2.3 ทดสอบอนาล็อกอินพุตจาก Slave ส่งไปยัง Master 30

4.2.4 ทดสอบส่งข้อมูล RFID จาก Slave ส่งไปยัง Master 30

4.3 ผลการทดสอบ 31

บทที่ 5 สรุปผลและข้อเสนอแนะ 32

5.1 สรุปผล 32

5.2 ปัญหาและวิธีการแก้ไขปัญหา 32

5.2.1 ปัญหาที่พบ 32

5.2.2 วิธีการแก้ไขปัญหา 32

5.3 ข้อเสนอแนะ 32

เอกสารอ้างอิง 33

ภาคผนวก 34

ภาคผนวก ก. 35

ภาคผนวก ข. 40

ประวัติผู้เขียน 45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

1.1 แสดงแผนการดำเนินงาน 2

4.1 ผลการทดสอบ 31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	โมดูล nRF24L01	3
2.2	บอร์ด Arduino	4
2.3	โปรแกรม Arduino IDE	5
2.4	บอร์ด Arduino Nano	6
2.5	RFID Module (RC522)	8
2.6	โปรแกรม proteus 8 professional	9
2.7	โปรแกรม PCBWizard	9
3.1	บอร์ด Arduino Nano	10
3.2	Pinout ของ Arduino Nano	11
3.3	โมดูล nRF24L01	11
3.4	การ Wiring nRF24L01 กับ Arduino Nano	12
3.5	โมดูล RFID RC522	13
3.6	การ Wiring RFID RC522 กับ Arduino Nano	13
3.7	ภาพรวมของอุปกรณ์	14
3.8	วงจร Digital Input ฝั่ง Slave	14
3.9	วงจร Digital Output ฝั่ง Slave	15
3.10	วงจร Analog Input ฝั่ง Slave	15
3.11	วงจร Digital Output ฝั่ง Master	16
3.12	วงจร Digital Input ฝั่ง Master	16
3.13	วงจร Analog Output ฝั่ง Master	17
3.14	โปรแกรมที่ใช้อ่านของฝั่ง Slave	18
3.15	โปรแกรมที่ใช้เขียนของฝั่ง Slave	19
3.16	โปรแกรมที่ใช้แสดงอินพุต/เอาต์พุตของฝั่ง Slave	20
3.17	โปรแกรมที่ใช้อ่านของฝั่ง Master	21
3.18	โปรแกรมที่ใช้เขียนของฝั่ง Master	22
3.19	โปรแกรมที่ใช้แสดงอินพุต/เอาต์พุตของฝั่ง Master	23
3.20	อุปกรณ์ฝั่ง Slave ที่ออกแบบโดยใช้โปรแกรม PCBWizard	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.21 อุปกรณ์ฝั่ง Slave 24
3.22 อุปกรณ์ฝั่ง Master ที่ออกแบบโดยใช้โปรแกรม PCBWizard 25
3.23 อุปกรณ์ฝั่ง Master 25
4.1 การรับส่งอินพุต/เอาต์พุต และข้อมูลRFID 26
4.2 เครื่องฉีดน้ำยาเคลือบผิวพิเศษ 27
4.3 ขณะทำการติดตั้งอุปกรณ์ 27
4.4 จอแสดงผลของอุปกรณ์ฝั่ง Slave 28
4.5 จอแสดงผลของอุปกรณ์ฝั่ง Slave (ต่อ) 28
4.6 จอแสดงผลของอุปกรณ์ฝั่ง Master 28
4.7 จอแสดงผลของอุปกรณ์ฝั่ง Master (ต่อ) 29
4.8 กราฟอินพุต/เอาต์พุตของเครื่องจักรที่ได้จากจอแสดงผลของอุปกรณ์ฝั่ง Master 29
4.9 จอแสดงผลของอุปกรณ์ฝั่ง Slave และ Master 30
4.10 แรงดันไฟฟ้าที่วัดได้จากอนาล็อกเอาต์พุตของอุปกรณ์ฝั่ง Master 30
4.11 บัตรพนักงานและอุปกรณ์ฝั่ง Slave 31
4.12 จอแสดงผลของอุปกรณ์ฝั่ง Slave และ Master 31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

บริษัท อุตสาหกรรมทำเครื่องแก้วไทย จำกัด (มหาชน) มีการดำเนินการทางธุรกิจด้านบรรจุภัณฑ์แก้วในไทย ปัจจุบันบริษัทฯต้องการทราบข้อมูลการทำงานของเครื่องจักรที่อยู่ในความรับผิดชอบของแผนกวิศวกรรมหน้าราง ซึ่งประกอบด้วยเครื่องจักรที่ทำหน้าที่ตรวจสอบคุณภาพ และบรรจุหีบห่อ เพื่อจะได้ทราบประสิทธิผลโดยรวมของเครื่องจักร และเพื่อนำมาใช้ปรับปรุงกระบวนการผลิตให้มีประสิทธิภาพมากยิ่งขึ้น ซึ่งเดิมหากเครื่องจักรมีปัญหา หรือขัดข้อง จะมีพนักงานลงพื้นที่เพื่อไปซ่อม จากนั้นจะลงบันทึกไว้ว่าเครื่องจักรมีปัญหาอย่างไร แก้ไขด้วยวิธีใด ในบางครั้งข้อมูลจากการบันทึกของพนักงานไม่ครบถ้วน ทำให้ไม่สามารถวิเคราะห์ประสิทธิผลโดยรวมของเครื่องจักรได้อย่างมีประสิทธิภาพ จึงมีแนวคิดที่จะสร้างฐานข้อมูลสำหรับเก็บข้อมูลอินพุต/เอาต์พุตของเครื่องจักร เพื่อที่จะได้ข้อมูลที่ถูกต้อง อีกทั้งยังสามารถใช้ข้อมูลช่วยในการตัดสินใจว่าควรแก้ไขหรือพัฒนาเครื่องจักรได้อย่างมีประสิทธิภาพ อาทิเช่น เมื่อทางแผนกมีข้อมูลว่าเครื่องจักรส่วนที่ทำหน้าที่ตรวจสอบคุณภาพมีการหยุดการทำงานบ่อย ๆ ทางแผนกก็จะส่งช่างไปปรับปรุงแก้ไข ณ จุดๆนั้นได้ทันท่วงที และใช้ข้อมูลนั้นในการพัฒนาให้เครื่องจักรดียิ่งขึ้น ดังนั้นทางบริษัทฯจึงมอบหมายให้ผู้วิจัยสร้างต้นแบบของอุปกรณ์ที่จะใช้รับส่งอินพุต/เอาต์พุตของเครื่องจักรขึ้น โดยอุปกรณ์ชิ้นนี้สามารถรับส่งอินพุต/เอาต์พุตของเครื่องจักร นอกจากนี้ยังสามารถรับส่งข้อมูลบัตรพนักงาน เพื่อให้สามารถระบุตัวพนักงานที่ซ่อม หรือแก้ไขเครื่องจักรได้

อย่างไรก็ตาม ข้อจำกัดแรกในการรับส่งข้อมูลเหล่านี้คือ หากทางบริษัทจะเปลี่ยน PLC จากของเดิมที่ติดอยู่กับเครื่องจักรให้สามารถรับส่งอินพุต/เอาต์พุตได้จะเป็นเรื่องที่ยุ่งยาก เนื่องจากต้องมีการหยุดการทำงานของเครื่องจักรทำให้เสียผลผลิตในช่วงเวลานั้น และอาจส่งผลกระทบต่อการส่งมอบสินค้าให้แก่ลูกค้า อีกข้อจำกัดหนึ่งก็คือถ้าหากทางแผนกใช้สายส่งสัญญาณเพื่อส่งข้อมูล จะมีต้นทุนทั้งการติดตั้งสาย ค่าเดินสาย การบำรุงรักษา และยังยากต่อการใช้งานหากต้องนำไปใช้กับเครื่องจักรอื่น ๆ และเนื่องด้วยอุปกรณ์ที่สามารถรับส่งข้อมูลแบบไร้สายที่วางขายทั่วไปนั้นมีราคาสูงมาก อีกทั้งการใช้งานยังไม่สามารถตอบโจทย์ที่ทางแผนกต้องการ และไม่สามารถพัฒนาตัวอุปกรณ์ต่อได้เพราะมีลิขสิทธิ์ในการพัฒนาตัวอุปกรณ์ ดังนั้นทางบริษัทฯจึงมอบหมายให้ผู้วิจัยจัดทำอุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตของ PLC ระยะไกลด้วยความถี่ 2.4 GHz ขึ้น

1.2 วัตถุประสงค์ของโครงงาน

1. ศึกษาการรับส่งข้อมูลอินพุต/เอาต์พุตระยะไกลแบบไร้สาย
2. เพื่อสร้างอุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตและ RFID ให้กับ PLC แบบไร้สาย
3. เพื่อนำอุปกรณ์ไปปรับใช้กับเครื่องจักรในโรงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน

1. ออกแบบวงจรอินพุต/เอาต์พุตของอุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตและ RFID ให้กับ PLC แบบไร้สาย โดยใช้โปรแกรม Proteus 8, PCBWizard

2. ออกแบบโปรแกรมเพื่อรับส่งข้อมูลอินพุต/เอาต์พุตและข้อมูลบัตรพนักงานแบบไร้สายโดยใช้โปรแกรม Arduino IDE

1.4 วิธีดำเนินการทำโครงงาน

ตารางที่ 1.1 แผนการดำเนินงาน

แผนการดำเนินงาน	เดือน สิงหาคม				เดือน กันยายน				เดือน ตุลาคม				เดือน พฤศจิกายน			
ศึกษาโรงงานและวางแผนโครงงาน																
ศึกษาการใช้งานโปรแกรม Solidworks																
ศึกษาการใช้งานโปรแกรม PLC Omron																
ศึกษาการใช้งานโปรแกรม Arduino																
ศึกษาการใช้งานโมดูลส่งสัญญาณ																
เขียนโปรแกรม และออกแบบอุปกรณ์																
ทดลอง ปรับปรุง และประกอบอุปกรณ์																
จัดทำรูปเล่มโครงงานและนำเสนอ																
นำเสนอผลงาน																

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถนำอุปกรณ์ไปใช้รับส่งข้อมูลของเครื่องจักรต่าง ๆ ในโรงงานได้จริง

2. สามารถนำข้อมูลจากอุปกรณ์รับส่งข้อมูลไปวิเคราะห์ และปรับปรุงประสิทธิภาพของเครื่องจักรในโรงงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

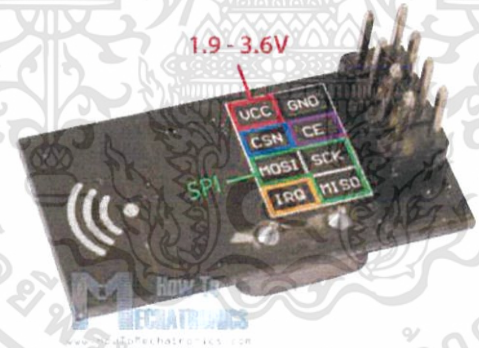
บทที่ 2
วัสดุอุปกรณ์ และโปรแกรมที่เกี่ยวข้อง

2.1 กล่าวนำ

บทที่ 2 จะกล่าวถึงที่มา ข้อมูล และคุณสมบัติต่าง ๆ ของวัสดุอุปกรณ์ และโปรแกรมที่ใช้ในโครงงานนี้ ได้แก่ โมดูลสื่อสารไร้สาย รุ่น nRF24L01, Arduino Nano, RFID-RC522, โปรแกรม Arduino IDE, โปรแกรม Proteus 8 Professional และโปรแกรม PCBWizard

2.2 โมดูลสื่อสารไร้สาย รุ่น nRF24L01 [1]

โมดูล NRF24L01 เป็นโมดูลสื่อสารไร้สายที่สามารถเขียนโปรแกรมให้เป็นได้ทั้งตัวรับ และตัวส่ง สามารถใช้กับ Arduino ได้หลายๆ ตัวพร้อมกัน ใช้งานบนความถี่ย่าน 2.4 GHz มีขนาดเล็กสะดวกในใช้งาน สามารถประยุกต์ใช้งานได้หลายๆอย่าง เช่น ใช้เป็นอุปกรณ์ส่งข้อมูลของเซนเซอร์อัตโนมัติสำหรับควบคุมอุณหภูมิ ความชื้น การแจ้งเตือนต่าง ๆ ควบคุมและติดตามหุ่นยนต์ Robot Control and Monitoring ได้ในระยะ 15 ถึง 500 เมตร เหมาะสำหรับงานที่มีเซนเซอร์หลายจุดและไม่สะดวกเดินสายไปจุดต่าง ๆ โมดูลนี้ใช้ชิฟ nRF24L01+ ทำงานด้วยความเร็วสูง High-speed SPI Interface ใช้พลังงานต่ำ รองรับการทำงานร่วมกับ Arduino และมีเสาอากาศมาให้ในตัว



รูปที่ 2.1 โมดูล nRF24L01

ตัวอย่างคุณสมบัติของโมดูล nRF24L01

- ใช้ย่านความถี่มาตรฐาน 2.4 GHz
- สามารถเลือกอัตราการสื่อสารได้ 3 ย่าน คือ 250 kbps, 1 Mbps และ 2 Mbps
- ใช้พลังงาน 13.5 mA ในการรับข้อมูลที่อัตราการสื่อสาร 2 Mbps
- ในสภาวะ Standby ใช้พลังงานเพียง 26 μA (ไมโครแอมป์)
- ทำงานที่ย่านแรงดัน 1.9 – 3.6 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

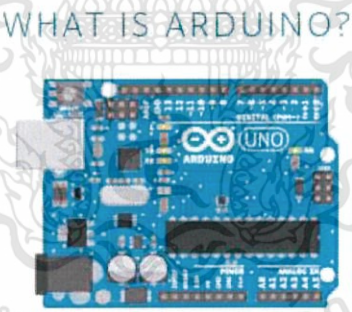
- ใช้โปรโตคอลการรับ-ส่งข้อมูล 2 ทางแบบ Enhanced ShockBurst™
- ใช้มาตรฐาน MultiCeiver™ ทำให้สามารถรับข้อมูลจากตัวส่งพร้อมกันทีเดียวสูงสุด 6 ชุดในหนึ่งความถี่

2.3 Arduino [2][3][4]

Arduino อ่านว่า (อาดูอิโน่ หรือ อาดุยโน่) เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัว บอร์ด Arduino ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลงเพิ่มเติมพัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย

จุดเด่นที่ทำให้บอร์ด Arduino เป็นที่นิยม

- ง่ายต่อการพัฒนา มีรูปแบบคำสั่งพื้นฐาน ไม่ซับซ้อนเหมาะสำหรับผู้เริ่มต้น
- มี Arduino Community กลุ่มคนที่ร่วมกันพัฒนาที่แข็งแรง
- Open Hardware ทำให้ผู้ใช้สามารถนำบอร์ดไปต่อยอดใช้งานได้หลายด้าน
- ราคาไม่แพง
- Cross Platform สามารถพัฒนาโปรแกรมบน OS ใดก็ได้



รูปที่ 2.2 บอร์ด Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเขียนโปรแกรมบน Arduino

เขียนโปรแกรมบนคอมพิวเตอร์ผ่านทางโปรแกรม Arduino IDE ซึ่งสามารถดาวน์โหลดได้จาก https://www.arduino.cc/en/main/software

แนวคิดการใช้งานโปรแกรม Arduino IDE

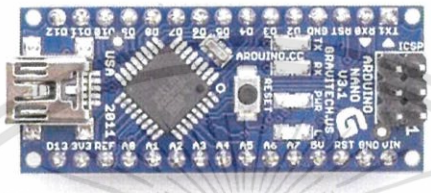
1. เขียนโปรแกรมด้วยภาษา C/C++ สำหรับ Arduino
2. คอมไพล์หรือแปลโปรแกรมภาษา C/C++ ให้เป็นภาษาสำหรับไมโครคอนโทรลเลอร์และบันทึกเป็น Intel Hex File
3. อัปโหลด Intel Hex File ลงบนไมโครคอนโทรลเลอร์ซึ่งอยู่บนบอร์ด Arduino ผ่านสาย USB หรือผ่าน Programmer

รูปที่ 2.3 โปรแกรม Arduino IDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Arduino Nano

บอร์ด Arduino Nano ออกแบบมาให้มีขนาดเล็ก และใช้กับงานทั่วไป ใช้ชิปไอซีไมโครคอนโทรเลอร์เบอร์ ATmega168 หรือเบอร์ ATmega328 (รุ่น 2.3 กับ 3) โปรแกรมผ่านโปรโตคอล UART มีชิป USB to UART มาให้ใช้ Mini USB เชื่อมต่อกับคอมพิวเตอร์ มีพอร์ตดิจิตอลอินพุตเอาต์พุต 14 พอร์ต มีพอร์ตอนาล็อกอินพุต 8 พอร์ต บนบอร์ดยังมีเรกกูเลเตอร์ สามารถจ่ายไฟได้ตั้งแต่ 7–12V เพื่อให้บอร์ดทำงานได้ (จ่ายไฟที่ขา VIN) กรณีมีแหล่งจ่ายไฟ 5V อยู่แล้วก็จ่ายเข้าได้เลยที่ขา 5V



รูปที่ 2.4 บอร์ด Arduino Nano

ข้อมูลจำเพาะ

ชิปไอซีไมโครคอนโทรเลอร์	ATmega168 หรือ ATmega328
ใช้แรงดันไฟฟ้า	5 V
รองรับการจ่ายแรงดันไฟฟ้า (ที่แนะนำ)	7 – 12 V
รองรับการจ่ายแรงดันไฟฟ้า (ที่จำกัด)	6 – 20 V
พอร์ต Digital I/O	14 พอร์ต (มี 6 พอร์ต PWM output)
พอร์ต Analog Input	6 พอร์ต
กระแสไฟที่จ่ายได้ในแต่ละพอร์ต	40 mA
กระแสไปที่จ่ายได้ในพอร์ต 3.3V	50 mA
พื้นที่โปรแกรมภายใน	16 KB หรือ 32 KB พื้นที่โปรแกรม, 500 B ใช้โดย Bootloader
พื้นที่แรม	1 หรือ 2 KB
พื้นที่หน่วยความจำถาวร (EEPROM)	512 B หรือ 1 KB
ความถี่คริสตัล	16 MHz
ขนาด	45x18 mm
น้ำหนัก	5 กรัม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 RFID [5][6]

RFID ย่อมาจากคำว่า Radio Frequency Identification เป็นระบบที่ได้ถูกพัฒนามาตั้งแต่ปี ค.ศ. 1980 โดยจุดเด่นของ RFID คือความสามารถในการอ่านข้อมูลของฉลากได้โดยที่ไม่ต้องมีการสัมผัส สามารถอ่านค่าได้แม่นยำแม้ในสภาพที่ทัศนวิสัยไม่ดี ทนต่อความเปียกชื้น แรงสั่นสะเทือน การกระทบกระแทก และสามารถอ่านข้อมูลได้ด้วยความเร็วสูง

ปัจจุบันมีการนำ RFID มาใช้งานกันในงานหลายงาน ไม่ว่าจะเป็นในบัตรชนิดต่างๆ เช่น บัตรประจำตัวประชาชน บัตร ATM บัตรเข้าออกสำนักงาน หรือ ในอาคารที่พัก บัตรจอดรถ ฉลากของสินค้า หรือแม้แต่ใช้ฝัง RFID ลงในตัวสัตว์เพื่อบันทึกประวัติ เป็นต้น

องค์ประกอบหลักของ RFID ประกอบด้วย 2 ส่วน

1. Transponder ตัวจัดเก็บและส่งข้อมูล ซึ่งมาจากคำว่า Transmitter ผสมกับคำว่า Responder ที่อยู่ในรูปแบบของ ฉลาก หรือ ป้าย โดยฉลากนี้จะทำ การบันทึกข้อมูลเกี่ยวกับวัตถุชิ้นนั้นเอาไว้ โดยเราอาจเรียกทั่ว ไปว่า "Tag" (แท็ก) ซึ่งแท็กจะทำหน้าที่ส่งสัญญาณ หรือ ข้อมูลที่บันทึกอยู่ในแท็กตอนสนองไปที่ตัวอ่านข้อมูล

2. Reader อุปกรณ์สำหรับอ่าน หรือ เขียนข้อมูลภายในแท็ก

หลักการทำงานเบื้องต้นของระบบ RFID

การสื่อสารระหว่างแท็กและตัวอ่านข้อมูลจะเป็นการสื่อสารกันโดยอาศัยช่องความถี่วิทยุผ่านอากาศ ซึ่งสัญญาณนี้ผ่านได้ทั้งโลหะและอโลหะ แต่ Tags ไม่สามารถติดต่อกับเครื่องอ่านให้อ่านได้โดยตรง เมื่อเครื่องอ่านส่งข้อมูลผ่านความถี่วิทยุ แสดงถึงความต้องการข้อมูลที่ถูกระบุไว้จากป้าย ป้ายจะตอบข้อมูลกลับ และเครื่องอ่านจะส่งข้อมูลต่อไปยังส่วนประมวลผลหลักของคอมพิวเตอร์ โดยเครื่องอ่านจะติดต่อสื่อสารกับคอมพิวเตอร์โดยผ่านสายเครือข่าย Lan (Loco Area Network) หรือ ส่งผ่านทางความถี่วิทยุจากทั้งอุปกรณ์มีสายและอุปกรณ์ไร้สาย

RFID-RC522

RFID RC522 Read/Write Module (SPI) เป็น RFID ทำงานที่ย่านความถี่ HF 13.56 MHz แบบ Mifare® (ISO-14443) ในชุดมาพร้อมกับ Tag Card และ Tag Key สามารถใช้งานได้ทั้งเขียนและอ่าน RFID tag ได้ ทำงานที่แรงดันไฟขนาด 3.3 โวลต์ เชื่อมต่อกับไมโครคอนโทรเลอร์ด้วย SPI Interface การใช้งาน RFID RC522 Read/Write Module (SPI) สามารถใช้งานร่วมกับ Arduino ได้ มีไลบารีสำเร็จรูปสำหรับนักพัฒนาอยู่บน GitHub แล้วสามารถดาวน์โหลดมาใช้งานได้ทันที

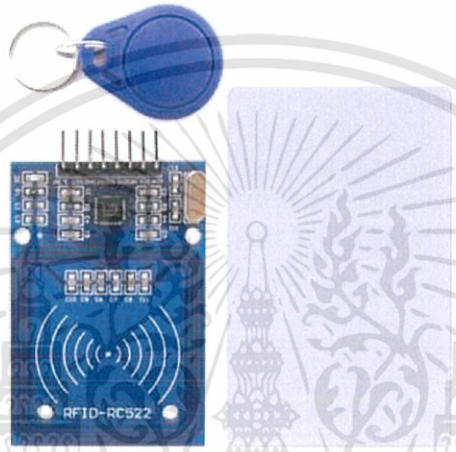
RFID ที่นิยมใช้กันมี 3 ย่านความถี่ ซึ่งก็มีข้อดีข้อเสียแตกต่างกัน จึงต้องเลือกใช้ให้เหมาะกับงาน

- ความถี่ต่ำ หรือ LF ความถี่ 125 kHz มีระยะการอ่านที่ใกล้ ส่งข้อมูลได้ช้า แต่ทะลุทลวงสิ่งกีดขวางได้ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ความถี่สูง หรือ HF ความถี่ 13.5 MHz มีระยะการอ่านที่ไกลกว่า LF ส่งข้อมูลได้ไกลกว่า สามารถทะลุทลวงสิ่งกีดขวางได้ดี ความฎีนี้จึงเป็นที่นิยมใช้งานกันมากที่สุด
- ความถี่สูงยิ่ง หรือ UHF ความถี่ในช่วง 860 to 960 MHz ส่งข้อมูลได้รวดเร็วและได้ไกล หลายเมตร ข้อด้อยคือทะลุทลวงสิ่งกีดขวางได้ไม่ค่อยดี

การเลือกใช้แทกและเครื่องอ่าน จะต้องเป็นความถี่เดียวกันถึงจะสามารถอ่านข้อมูลร่วมกันได้ ในโมดูล RFID Module นี้เลือกใช้ RFID ย่านความถี่ HF 13.5MHz ซึ่งเป็นที่นิยมใช้ในงานทั่วมากที่สุด มีราคาไม่แพงมีประสิทธิภาพการทำงานดี



รูปที่ 2.5 RFID Module (RC522)

2.5 โปรแกรม Proteus 8 professional [7][8]

โปรแกรม Proteus หรือ Proteus VSM (Virtual System Modeling) เป็นโปรแกรมที่พัฒนาขึ้นโดยบริษัทแล็บเซ็นเตอร์อิเล็กทรอนิกส์ จำกัด (Labcenter Electronics Ltd.) ที่ประเทศอังกฤษ โปรแกรม Proteus มีชื่อเต็มว่า Labcenter Electronics Proteus ซึ่งภายในโปรแกรมจะประกอบด้วยส่วนประกอบหลัก 2 ส่วน คือ ISIS และ ARES โปรแกรม Proteus จะมีอยู่หลายเวอร์ชันให้เลือกใช้งาน ซึ่งเวอร์ชันในปัจจุบัน คือ เวอร์ชัน 8 และถ้าต้องการศึกษาหาข้อมูลเกี่ยวกับเวอร์ชันนี้ ก็สามารถเยี่ยมชมได้ที่เว็บไซต์ http://www.labcenter.co.uk

ความสามารถในการทำงานของโปรแกรม Proteus ก็คือ สามารถจำลองการทำงานของวงจรอิเล็กทรอนิกส์ ได้หลากหลายรูปแบบ ไม่ว่าจะเป็นแบบอนาล็อกและแบบดิจิตอล หรือทั้งแบบอนาล็อกและดิจิตอลผสมกัน นอกจากนี้ Proteus ยังสามารถออกแบบลายวงจรพิมพ์ (PCB) ได้อีกด้วย จุดเด่นของโปรแกรม Proteus คือ การจำลองการทำงานของวงจรอิเล็กทรอนิกส์ ที่ใช้ไมโครคอนโทรลเลอร์ตระกูลต่าง ๆ ไม่ว่าจะเป็น PIC, MCS-51, AVR และ ARM เป็นต้น ทำให้นักเขียนโปรแกรมหรือโปรแกรมเมอร์สามารถตรวจสอบได้ว่าโปรแกรม หรือซอสโค้ด (Source Code) ที่เขียนขึ้นมานั้น สามารถสนับสนุนกับวงจรฮาร์ดแวร์ที่ต่อได้หรือไม่ ถ้าโปรแกรม (Source Code) ที่เขียนขึ้น ไม่สนับสนุนกับวงจรฮาร์ดแวร์ที่ต่อโปรแกรมเมอร์ก็จะทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาโปรแกรม (Source Code) ที่เขียนขึ้นใหม่ หรือปรับปรุงวงจรฮาร์ดแวร์ใน Proteus จนกว่าโปรแกรมที่เขียนขึ้นและฮาร์ดแวร์ที่ต่อสามารถสนับสนุนซึ่งกันและกัน ทำให้การสร้างโครงงานต่างๆ สามารถประหยัดเวลาและค่าใช้จ่ายเป็นอย่างมาก เพราะในอดีตการเขียนโปรแกรมขึ้นมานั้น จะต้องต่อวงจรจริงเพื่อทดสอบ ทำให้เสียเวลาและค่าใช้จ่ายมาก ในกรณีที่วงจรฮาร์ดแวร์และโปรแกรมที่เขียนขึ้นไม่สนับสนุนซึ่งกันและกัน

รูปที่ 2.6 โปรแกรม proteus 8 professional

2.6 โปรแกรม PCBWizard [9]

PCB Wizard คือซอฟต์แวร์สำเร็จรูปเปี่ยมประสิทธิภาพ ที่ได้รับการออกแบบมาสำหรับแผ่นวงจรพิมพ์ (PCB) ชนิดหน้าเดียวและสองหน้า การออกแบบและสร้างแผ่น PCB การสร้างลายปริ้นต์ทองแดงทำได้ง่ายๆ โดยใช้โปรแกรม PCB Wizard ทำการวาดภาพลายเส้น ด้วยการเลือกวางตำแหน่งอุปกรณ์ Component แล้วลากลายเส้น Track เพื่อเชื่อมต่อลายวงจรอุปกรณ์อิเล็กทรอนิกส์

ซึ่งมีซอฟต์แวร์ประกอบด้วยเครื่องมือหลากหลายที่ครอบคลุมทุกขั้นตอนการผลิต PCB แบบดั้งเดิม ซึ่งได้แก่ การวาดแผนผัง, การจับภาพวงจร, การเปลี่ยนส่วนประกอบ, การเดินวงจรอัตโนมัติ, การรายงานรายการวัสดุ และการสร้างไฟล์สำหรับการผลิต. นอกจากนี้ PCB Wizard ยังมีคุณสมบัติใหม่ๆ อันชาญฉลาดที่กำจัดความยากในการเรียนรู้ที่มักมากับซอฟต์แวร์สำเร็จรูป PCB

รูปที่ 2.7 โปรแกรม PCBWizard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

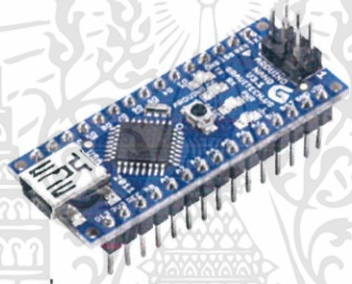
บทที่ 3
วิธีการดำเนินงาน

3.1 กล่าวนำ

ในการออกแบบและสร้างอุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตและ RFID ให้กับ PLC แบบไร้สาย จำเป็นต้องศึกษาวัสดุอุปกรณ์ วงจรอินพุต/เอาต์พุต และโปรแกรมต่างๆ เพื่อที่จะนำมาใช้ดำเนินงาน เช่น วัสดุอุปกรณ์ที่เราจะนำมาใช้มีความเหมาะสมกับลักษณะการใช้งานหรือไม่ วงจรที่ใช้ควรเป็นอย่างไร โปรแกรมที่ออกแบบมาสามารถใช้งานได้ครอบคลุมหรือไม่

3.2 วัสดุอุปกรณ์

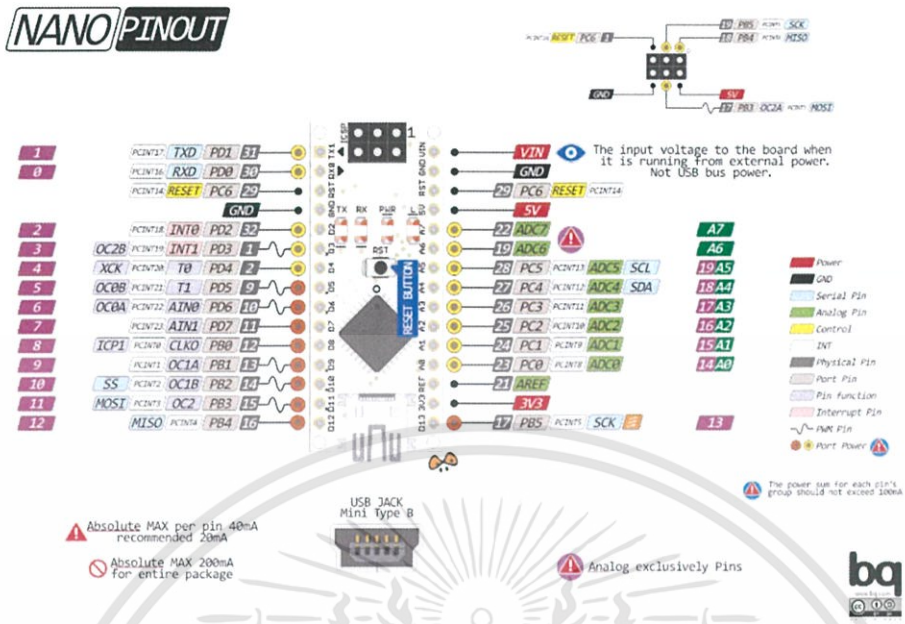
3.2.1 Arduino Nano จำนวน 2 ตัว



รูปที่ 3.1 บอร์ด Arduino Nano

เหตุผลที่เลือกใช้ Arduino Nano เนื่องจาก เป็นรุ่นที่มีขนาดเล็กเหมาะสำหรับการนำไปใช้กับตัวอุปกรณ์ที่จะสร้างขึ้น และมีจำนวนพอร์ตอินพุต/เอาต์พุตให้ใช้เยอะถึง 20 พอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 Pinout ของ Arduino Nano

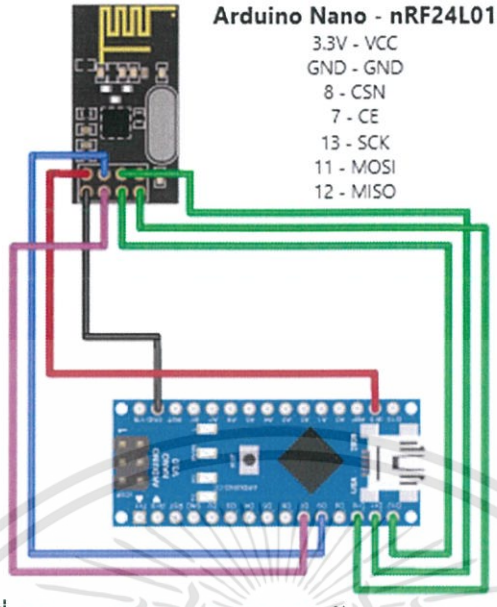
จากรูปที่ 3.2 เนื่องจาก Arduino แต่ละชนิดมีจำนวน Pinout ไม่เท่ากันและข้อมูลการใช้งานแต่ละอันต่างกัน ผู้ศึกษาจึงต้องทำความเข้าใจเกี่ยวกับการใช้งาน Pinout เพื่อที่จะนำไปใช้กับโมดูลอื่นๆ ที่ประกอบอยู่ในตัวอุปกรณ์

3.2.2 nRF24L01 จำนวน 2 ตัว

รูปที่ 3.3 โมดูล nRF24L01

อุปกรณ์สื่อสารไร้สายมีให้เลือกหลากหลายชนิด สาเหตุที่เลือกใช้ nRF24L01 เนื่องจากเป็นโมดูลไร้สายที่สามารถรับส่งได้ในตัวเดียวกัน สามารถส่งข้อมูลจากหลาย ๆ จุดมาหาตัวรับตัวเดียว ใช้งานบนย่านความถี่ 2.4 GHz ใช้งานไม่ยากมาก และราคาไม่สูงมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



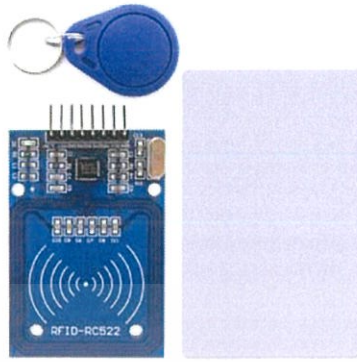
รูปที่ 3.4 การ Wiring nRF24L01 กับ Arduino Nano

จากรูปที่ 3.4 คือการ Wiring nRF24L01 กับ Arduino Nano โดยหน้าที่ของ Pinout ของ nRF24L01 แต่ละตัวมีดังนี้

ชื่อขา	ชื่อเต็มของขา	หน้าที่
Ground	Ground	เชื่อมต่อกับกราวนด์ของระบบ
Vcc	Power	เชื่อมต่อแหล่งจ่ายไฟของโมดูล (3.3V)
CE	Chip Enable	ใช้เพื่อเปิดใช้งานการสื่อสาร SPI
CSN	Ship Select Not	ขานี้จะต้องเป็น 1 เสมอ มิฉะนั้นจะปิดการใช้งาน SPI
SCK	Serial Clock	สร้างสัญญาณนาฬิกาให้กับ nRF24L01
MOSI	Master Out Slave In	สำหรับส่งข้อมูลออกไปผ่าน nRF24L01
MISO	Master In Slave Out	รับข้อมูลเข้ามาจาก nRF24L01
IRQ	Interrupt	เป็นขาที่ Active Low และใช้เมื่อต้องมี Interrupt เท่านั้น

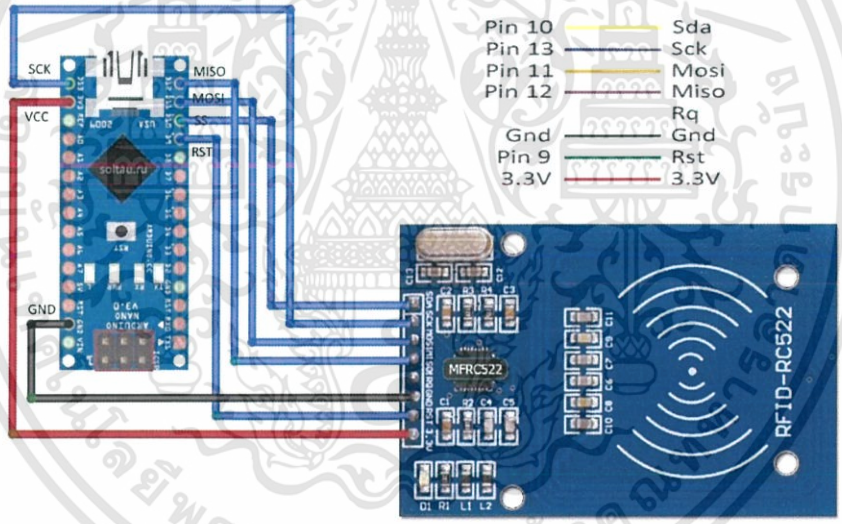
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 RFID-RC522 จำนวน 1 ตัว



รูปที่ 3.5 โมดูล RFID RC522

เนื่องจาก มีความจำเป็นที่จะต้องการระบุตัวพนักงานที่เข้ามาแก้ไข หรือ ซ่อมเครื่องจักร ซึ่งโดยปกติบัตรพนักงานที่ใช้อยู่ปัจจุบันเป็นแท็กที่สามารถใช้ร่วมกับโมดูลนี้อยู่แล้วจึงเป็นเหตุผลให้เลือกใช้โมดูลนี้



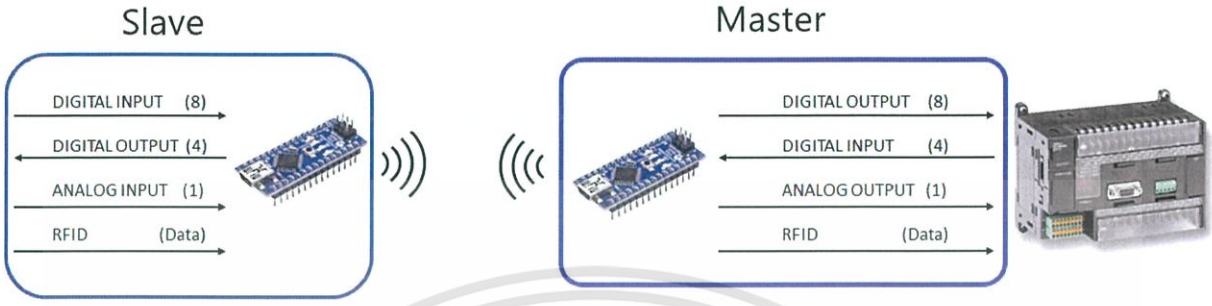
รูปที่ 3.6 การ Wiring RFID RC522 กับ Arduino Nano

จากรูปที่ 3.6 การ Wiring โมดูล RFID RC522 กับ Arduino Nano โดยหน้าที่ของ Pinout แต่ละอันมีหน้าที่เหมือนกับโมดูล nRF24L01 เนื่องจากเป็นการสื่อสารส่งข้อมูลแบบ SPI เหมือนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 วงจรอินพุต เอาต์พุต

โดยหลัก ๆ อุปกรณ์จะแบ่งเป็น 2 ฝั่ง คือ Master และ Slave

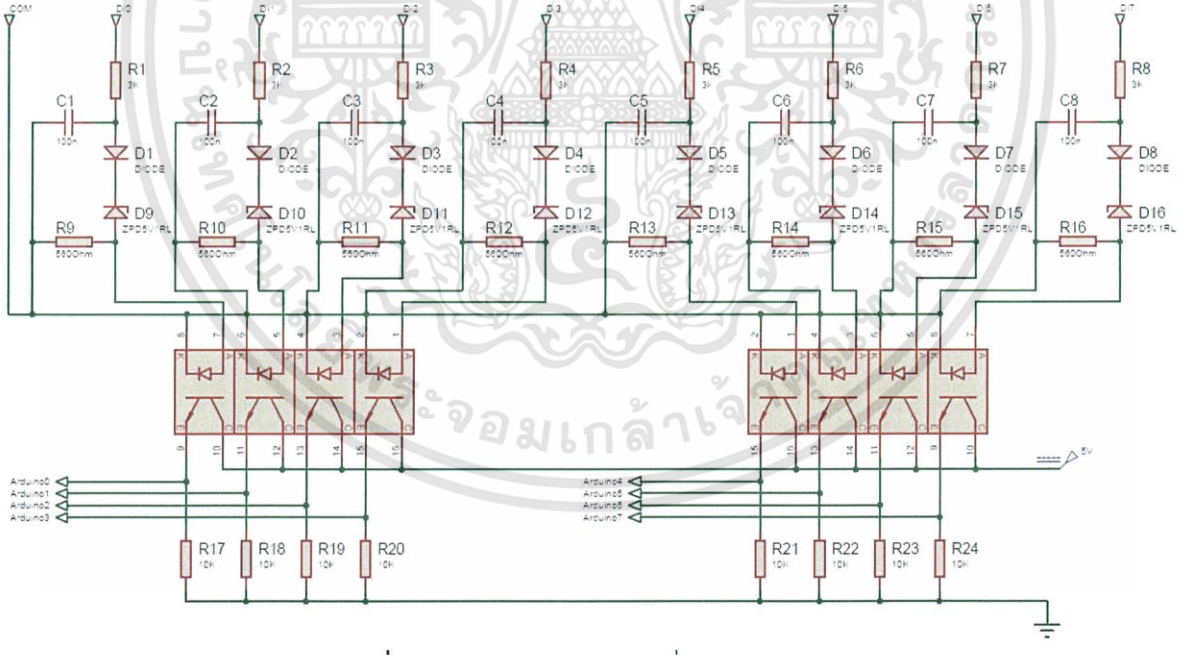


รูปที่ 3.7 ภาพรวมของอุปกรณ์

3.3.1 Slave

3.3.1.1 Digital Input

เป็นวงจรที่รับดิจิตอลอินพุตจากภายนอก เช่น เครื่องจักร ,PLCที่ควบคุมเครื่องจักร เป็นต้น จากนั้นจะส่งอินพุตนี้เข้าไปยัง Arduino

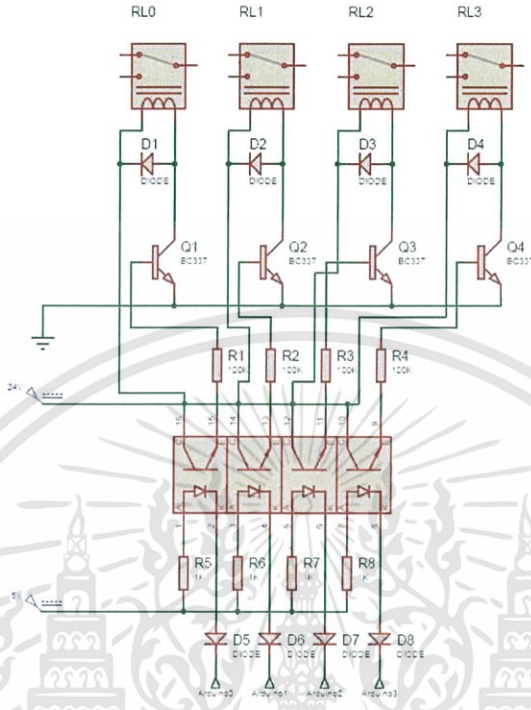


รูปที่ 3.8 วงจร Digital Input ฝั่ง Slave

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1.2 Digital Output (Relays Output)

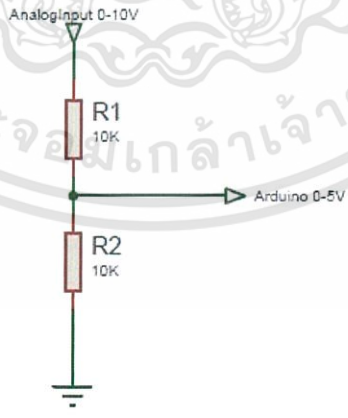
เป็นวงจรดิจิตอลเอาต์พุตที่รับสถานะมาจาก Arduino และส่งต่อไปสั่งการ Relays Output



รูปที่ 3.9 วงจร Digital Output ฝั่ง Slave

3.3.1.3 Analog Input

เป็นวงจรแบ่งแรงดันซึ่งสามารถรับค่าได้ 0-10 V จากนั้นจะแบ่งแรงดันให้เหลือ 0-5 V และส่งต่อไปยัง Arduino



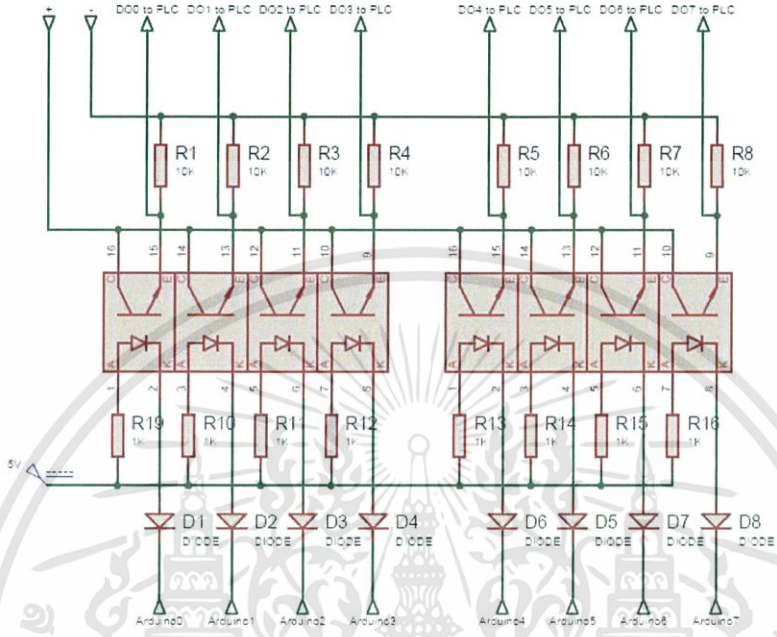
รูปที่ 3.10 วงจร Analog Input ฝั่ง Slave

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 Master

3.3.2.1 Digital Output

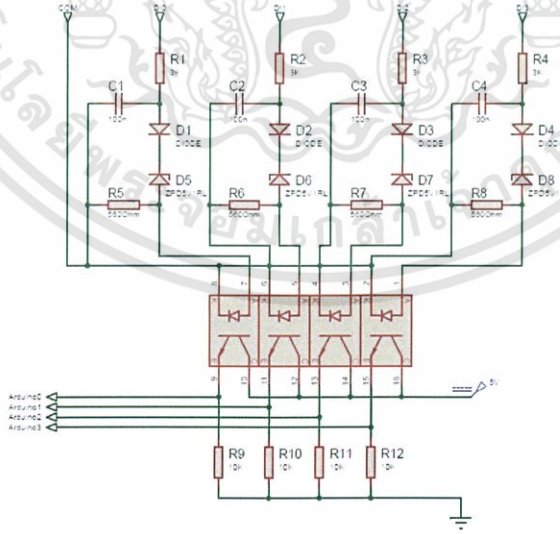
เป็นวงจรที่รับดิจิตอลเอาต์พุตซึ่งรับอินพุตมาจากฝั่ง Slave จากนั้นจะส่งเอาต์พุตนี้ไปยัง PLC



รูปที่ 3.11 วงจร Digital Output ฝั่ง Master

3.3.2.2 Digital Input

เป็นวงจรที่รับดิจิตอลอินพุตจาก PLC จากนั้นจะส่งอินพุตนี้เข้าไปยัง Arduino

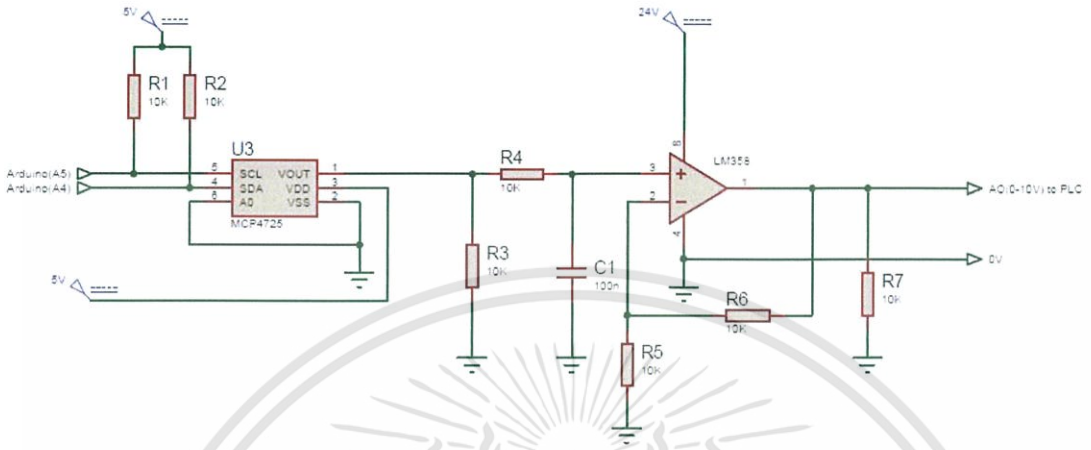


รูปที่ 3.12 วงจร Digital Input ฝั่ง Master

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2.3 Analog Output

เป็นวงจรที่รับอนาล็อกอินพุตจากฝั่ง Slave จากนั้นวงจรจะแปลงอินพุตจาก Arduino ออกมาเป็น 0-10 V



รูปที่ 3.13 วงจร Analog Output ฝั่ง Master

3.4 โปรแกรม

ในส่วนการเขียนโปรแกรมจะเป็นการเขียนให้มีการรับส่งอินพุต/เอาต์พุตและข้อมูลของบัตรพนักงานไปมาระหว่างฝั่งSlaveและฝั่งMaster

3.4.1 Slave

3.4.1.1 Read

ส่วนนี้จะเป็นการเขียนโปรแกรมให้เริ่มรับอินพุตมาจากฝั่ง Master แล้วแสดงออกเป็นรีเลย์เอาต์พุตที่ฝั่ง Slave ซึ่งหากไม่ได้รับอินพุตจากฝั่ง Master ภายใน 200 millisecond โปรแกรมจะแสดงให้เห็นว่าไม่ได้รับข้อมูลจากฝั่ง Master

จากโปรแกรมหากอินพุตที่ได้รับจากฝั่ง Master มีค่าเป็น LOW Arduino ฝั่ง Slave จะส่ง HIGH ไปให้วงจรรีเลย์เอาต์พุต ทำให้รีเลย์เอาต์พุตไม่ทำงาน แต่ถ้าอินพุตจากฝั่ง Master มีค่าเป็น HIGH Arduino ฝั่ง Slave จะส่ง LOW ไปให้วงจรรีเลย์เอาต์พุต ทำให้ตัวรีเลย์เอาต์พุตทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
radio.startListening();
unsigned long started_waiting_at = millis();
while(!radio.available()) {
  if (millis() - started_waiting_at > 200 ) {
    Serial.println("No response received - timeout!");
    return;
  }
}
radio.read(&Data2,sizeof(Data2));
for(int a=0;a<4;a++) {
  if(Data2[a] == 0) {
    digitalWrite(outputPins[a], HIGH);
  }
  else if (Data2[a] == 1) {
    digitalWrite(outputPins[a], LOW);
  }
}
```

รูปที่ 3.14 โปรแกรมที่ใช้อ่านของฝั่ง Slave

3.4.1.2 Write

ส่วนนี้จะเป็นการเขียนโปรแกรมให้อ่าน แสดง และส่งข้อมูลบัตรไปยังฝั่ง Master ซึ่งข้อมูลที่อ่านและส่งไป คือ เลขบัตร นอกจากนี้จะมีโปรแกรมที่ใช้อ่านค่าอนาล็อกอินพุตจากวงจรอนาล็อกอินพุตและส่งค่าอินพุตไปให้ฝั่ง Master อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
radio.stopListening();
for (int a=0;a<8;a++) {
  Data[a] = digitalRead(inputPins[a]);
}
if (rfid.isCard()) {
  if (rfid.readCardSerial()) {
    if (rfid.serNum[0] != Data[8]
    && rfid.serNum[1] != Data[9]
    && rfid.serNum[2] != Data[10]
    && rfid.serNum[3] != Data[11]
    && rfid.serNum[4] != Data[12]) {
      Data[8] = rfid.serNum[0];
      Data[9] = rfid.serNum[1];
      Data[10] = rfid.serNum[2];
      Data[11] = rfid.serNum[3];
      Data[12] = rfid.serNum[4];
      Serial.print("Cardnumber : ");
      for (int a=0;a<5;a++) {
        Serial.print(rfid.serNum[a],DEC);
        Serial.print(" ");
      }
      Serial.println(" ");
      previousMillis = currentMillis;
    }
  }
}
rfid.halt();
if (currentMillis - previousMillis >= interval) {
  for (int a=8;a<13;a++) {
    Data[a] = 0;
  }
}
Data[13] = analogRead(inputPins[8]);
radio.write(&Data,sizeof(Data));
```

รูปที่ 3.15 โปรแกรมที่ใช้เขียนของฝั่ง Slave

3.4.1.3 Show

ส่วนนี้จะเป็นการเขียนโปรแกรมให้แสดงข้อมูลอินพุต/เอาต์พุตของฝั่ง Slave ทุก ๆ 1 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (currentMillis - pre >= in) {
  for (int a=0;a<8;a++) {
    if (Data[a] == 1) {
      Serial.print("DI");
      Serial.print(a);
      Serial.println(" = HIGH");
    }
    else if (Data[a] == 0) {
      Serial.print("DI");
      Serial.print(a);
      Serial.println(" = LOW");
    }
  }
  for(int a=0;a<4;a++) {
    if(Data2[a] == 0) {
      Serial.print("DO");
      Serial.print(a);
      Serial.println(" = LOW");
    }
    else if(Data2[a] == 1) {
      Serial.print("DO");
      Serial.print(a);
      Serial.println(" = HIGH");
    }
  }
  Serial.print("AI = ");
  Serial.print(Data[13]);
  Serial.println("\n");
  pre = currentMillis;
 }
}
```

รูปที่ 3.16 โปรแกรมที่ใช้แสดงอินพุต/เอาต์พุตของฝั่ง Slave

3.4.2 Master

3.4.2.1 Read

ส่วนนี้จะเป็นการเขียนโปรแกรมให้รับทั้งดิจิตอลอินพุต อนาล็อกอินพุต และข้อมูลของบัตรพนักงานจากฝั่ง Slave

จากโปรแกรมหากดิจิตอลอินพุตที่ได้รับจากฝั่ง Slave มีค่าเป็น LOW Arduino ฝั่ง Master จะส่ง HIGH ไปให้วงจรดิจิตอลเอาต์พุต ทำให้ดิจิตอลเอาต์พุตไม่ทำงาน แต่ถ้าอินพุตจากฝั่ง Slave มีค่าเป็น HIGH Arduino ฝั่ง Master จะส่ง LOW ไปให้วงจรดิจิตอลเอาต์พุต ทำให้ตัวดิจิตอลเอาต์พุตทำงาน ส่วนของอนาล็อกอินพุตเมื่อรับค่ามาแล้ว Arduino ฝั่ง Master จะส่งค่าไปยังวงจรอนาล็อกเอาต์พุตต่อไป ส่วนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัตรพนักงาน เมื่อได้ข้อมูลบัตรพนักงานจากฝั่ง Slave ทาง Arduino ฝั่ง Master จะแสดงข้อมูลบัตรพนักงานใน Serial Monitor

```
if(radio.available()) {
  while(radio.available()) {
    radio.read(&Data, sizeof(Data));
    for(int a=0;a<8;a++) {
      if(Data[a] == 0) {
        digitalWrite(outputPins[a], HIGH);
      }
      else if (Data[a] == 1) {
        digitalWrite(outputPins[a], LOW);
      }
    }
    if(Data[8] != RFID[0]
    &&Data[9] != RFID[1]
    &&Data[10] != RFID[2]
    &&Data[11] != RFID[3]
    &&Data[12] != RFID[4]) {
      RFID[0] = Data[8];
      RFID[1] = Data[9];
      RFID[2] = Data[10];
      RFID[3] = Data[11];
      RFID[4] = Data[12];
      if(RFID[0] != 0
      &&RFID[1] != 0
      &&RFID[2] != 0
      &&RFID[3] != 0
      &&RFID[4] != 0) {
        Serial.print("Cardnumber : ");
        for(int i=0;i<5;i++){
          Serial.print(RFID[i]);
          Serial.print(" ");
        }
        Serial.println(" ");
      }
    }
    x = Data[13];
    y = map(x,0,1023,0,4095);
    dac.setVoltage(y,false);
  }
```

รูปที่ 3.17 โปรแกรมที่ใช้อ่านของฝั่ง Master

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2.2 Write

ส่วนนี้จะเป็นการเขียนโปรแกรมให้อ่าน และส่งดิจิตอลอินพุตไปยังฝั่ง Slave นอกจากนี้ถ้าหากไม่ได้รับค่าจากฝั่ง Slave โปรแกรมมจะสั่งให้สถานะของวงจรเอาต์พุตเป็น LOW

```
    radio.stopListening();
    for(int a=0;a<4;a++) {
      Data2[a] = digitalRead(inputPins[a]);
    }
   radio.write(&Data2,sizeof(Data2));
    radio.startListening();
  }
 else if(!radio.available()) {
    for(int a=0;a<8;a++) {
      digitalWrite(outputPins[a], HIGH);
    }
 }
```

รูปที่ 3.18 โปรแกรมที่ใช้เขียนของฝั่ง Master

3.4.2.3 Show

ส่วนนี้จะเป็นการเขียนโปรแกรมให้แสดงข้อมูลอินพุต/เอาต์พุตของฝั่ง Slave ทุก ๆ 1 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(currentMillis - pre >= in) {
  for(int a=0;a<8;a++) {
    if(Data[a] == 0) {
      Serial.print("DO");
      Serial.print(a);
      Serial.println(" = LOW");
     }
    else if(Data[a] == 1) {
      Serial.print("DO");
      Serial.print(a);
      Serial.println(" = HIGH");
     }
   }
  for(int a=0;a<4;a++) {
    if(Data2[a] == 1) {
      Serial.print("DI");
      Serial.print(a);
      Serial.println(" = HIGH");
     }
    else if (Data2[a] == 0) {
      Serial.print("DI");
      Serial.print(a);
      Serial.println(" = LOW");
     }
   }
  Serial.print("AO = ");
  Serial.print(y);
  Serial.println("\n");
  pre = currentMillis;
 }
}
```

รูปที่ 3.19 โปรแกรมที่ใช้แสดงอินพุต/เอาต์พุตของฝั่ง Master

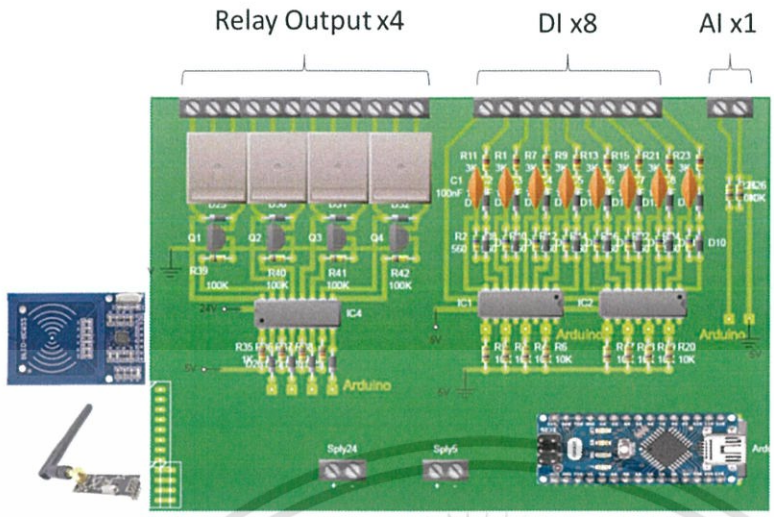
3.5 ประกอบอุปกรณ์

3.5.1 Slave

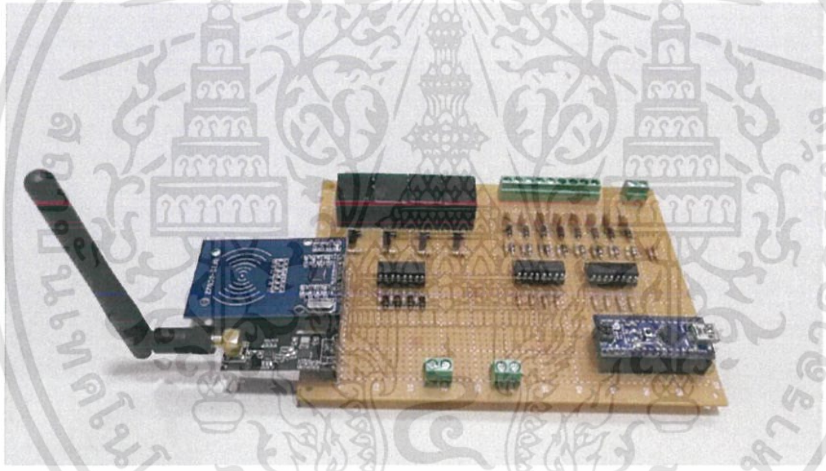
จากรูปที่ 3.20 คือ อุปกรณ์ฝั่ง Slave ซึ่งออกแบบโดยใช้โปรแกรม PCBWizard เพื่อจะได้เห็นภาพได้ชัดเจนยิ่งขึ้นว่าควรจัดวางวงจรแต่ละวงจรทั้งหมดอย่างไรให้เหมาะกับขนาดของอุปกรณ์

จากรูปที่ 3.21 คืออุปกรณ์ฝั่ง Slave หลังจากประกอบส่วนประกอบต่าง ๆ เข้าด้วยกันแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 อุปกรณ์ฝั่ง Slave ที่ออกแบบโดยใช้โปรแกรม PCBWizard



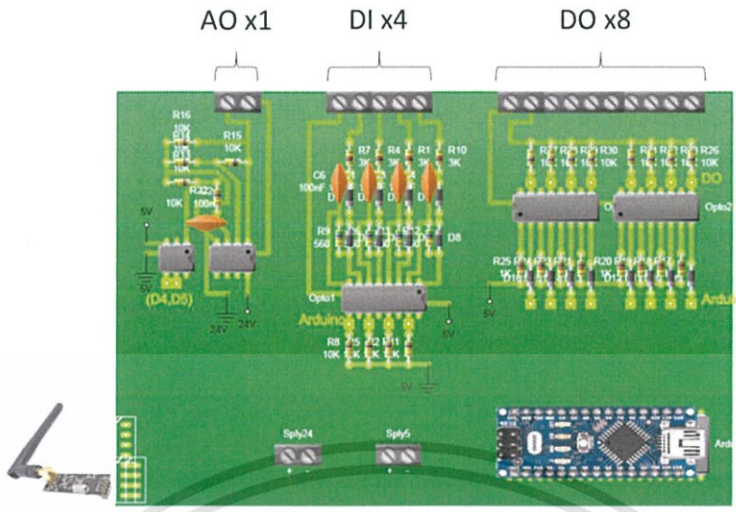
รูปที่ 3.21 อุปกรณ์ฝั่ง Slave

3.5.2 Master

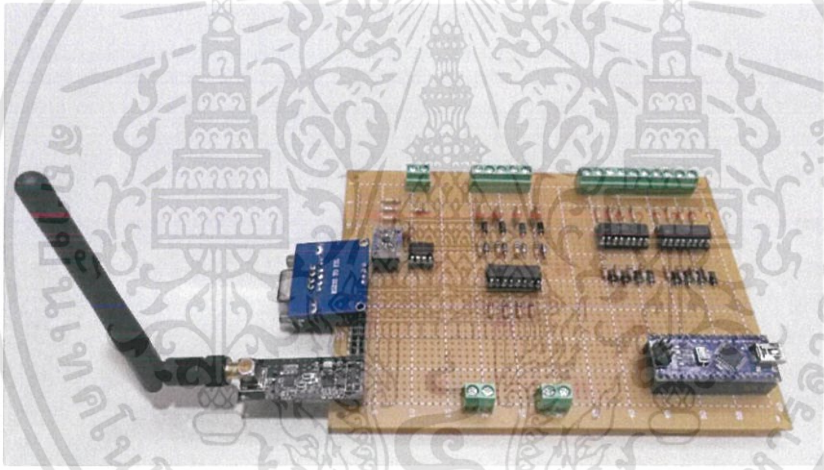
จากรูปที่ 3.22 คืออุปกรณ์ฝั่ง Master ซึ่งออกแบบโดยใช้โปรแกรม PCBWizard เพื่อจะได้เห็นภาพได้ชัดเจนยิ่งขึ้นว่าควรจัดวางวงจรแต่ละวงจรทั้งหมดอย่างไรให้เหมาะกับขนาดของอุปกรณ์

จากรูปที่ 3.23 คือ อุปกรณ์ฝั่ง Master หลังจากประกอบส่วนประกอบต่าง ๆ เข้าด้วยกันแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.22 อุปกรณ์ฝั่ง Master ที่ออกแบบโดยใช้โปรแกรม PCBWizard



รูปที่ 3.23 อุปกรณ์ฝั่ง Master

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

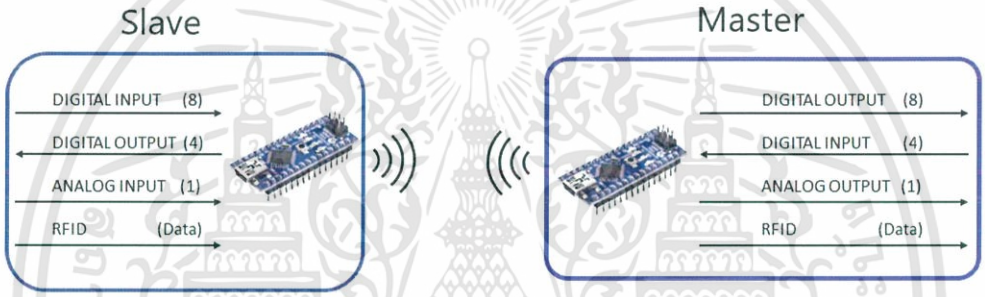
บทที่ 4
ผลการดำเนินงาน

4.1 กล่าวนำ

สำหรับเนื้อหาในบทนี้จะกล่าวถึงขั้นตอนในการทดสอบและผลการทดสอบ อุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตของ PLC ระยะไกลด้วยความถี่ 2.4 GHz กับเครื่องจักรภายในโรงงาน ซึ่งเครื่องจักรที่ผู้จัดทำได้ทำการทดสอบคือ เครื่องฉีดน้ำยาเคลือบผิวพิเศษ (Cold End Spray)

4.2 การทดสอบ

การทดสอบจะแบ่งเป็น 4 การทดสอบด้วยกัน



รูปที่ 4.1 การรับส่งอินพุต/เอาต์พุต และข้อมูลRFID

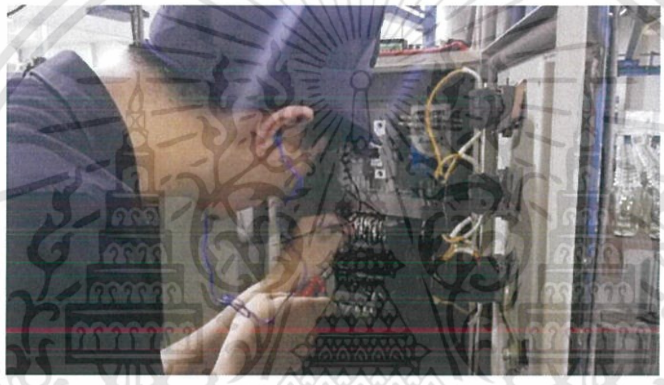
4.2.1 ทดสอบดิจิตอลอินพุตจาก Slave ส่งไปยัง Master

ทดสอบโดยใช้อุปกรณ์ฝั่ง Slave รับอินพุตจากเครื่องฉีดน้ำยาเคลือบผิวพิเศษ จากนั้นแสดงผลให้เห็นที่อุปกรณ์ฝั่ง Master

Cold End Spray เป็นเครื่องจักรที่คอยฉีดน้ำยาเคลือบผิวภายนอก เพื่อป้องกันรอยถลอกขณะที่ขวดแก้วอยู่ในระบบลำเลียง ซึ่งการทำงานของ Cold End Spray หลัก ๆ จะประกอบด้วยอินพุต คือ Photoelectric Sensor ทำหน้าที่นับแถวของขวด, Proximity Sensor 2 ตัว ทำหน้าที่ตรวจตำแหน่งของ Spray ที่ปลายทาง เอาต์พุตคือ Motor ทำหน้าที่พา Spray เคลื่อนที่จากฝั่งหนึ่งไปยังอีกฝั่ง, Spray ทำหน้าที่ฉีดน้ำยาเคลือบผิวพิเศษ, Cylinder ทำหน้าที่บังคับรางที่ Spray เคลื่อนที่ไปมา หลักการทำงานคือ เมื่อ Photoelectric Sensor ตรวจพบช่องว่างระหว่างแถวของขวดก็จะสั่งให้ Motor เคลื่อนที่ไปตามราง และ Spray จะฉีดน้ำยาไปจนสุดรางและ ณ ปลายราง Proximity Sensor จะตรวจพบว่า Spray ได้เคลื่อนที่มาถึงปลายทางแล้วก็จะสั่งให้ Motor และ Spray หยุดทำงานและสั่งให้ Cylinder ผลักรางเป็นอีกทิศทางหนึ่ง จากนั้นเมื่อ Photoelectric Sensor ตรวจพบช่องว่างระหว่างแถวของขวดอีกครั้งก็จะทำงานวนลูปเดิมไปเรื่อย ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 4.2 เครื่องฉีดน้ำยาเคลือบผิวพิเศษ



ภาพที่ 4.3 ขณะทำการติดตั้งอุปกรณ์

โดยผู้จัดทำได้นำอุปกรณ์ฝั่ง Slave ไปติดตั้งกับ PLC ที่ควบคุมเครื่องฉีดน้ำยาเคลือบผิวพิเศษ ซึ่งอินพุต/เอาต์พุตของเครื่องจักรจะเป็นอินพุตของอุปกรณ์ฝั่ง Slave จากนั้นจะส่งไปเป็นเอาต์พุตของอุปกรณ์ฝั่ง Master ซึ่งตั้งอยู่ห่างกันประมาณ 10 เมตร

จากการทดสอบดิจิตอลอินพุตของอุปกรณ์ฝั่ง Slave มีดังนี้

DI 0 = Photoelectric Sensor DI 1 = Motor DI 2 = Spray

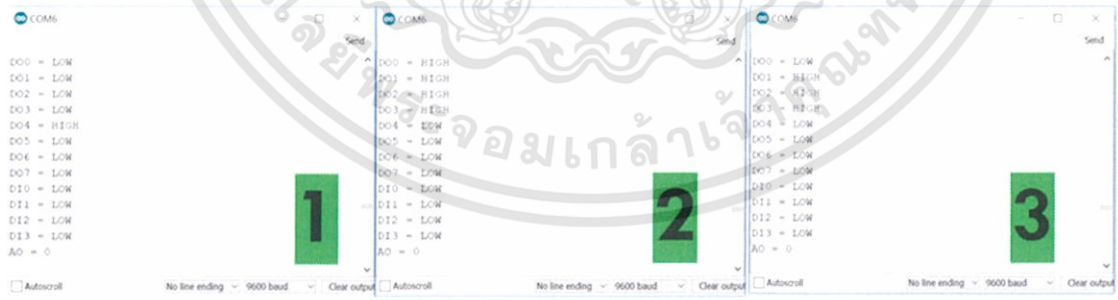
DI 3 = Cylinder1 DI 4 = Cylinder2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.4 จอแสดงผลของอุปกรณ์ฝั่ง Slave

รูปที่ 4.5 จอแสดงผลของอุปกรณ์ฝั่ง Slave (ต่อ)

จากรูปที่ 4.4 และ 4.5 จอแสดงผลได้แสดงให้เห็นว่าอุปกรณ์ได้ส่งอินพุต/เอาต์พุตของเครื่องฉีดน้ำยาเคลือบผิวพิเศษ ไปยังอุปกรณ์ฝั่ง Master อย่างไรบ้างโดยเลข 1-6 เรียงลำดับตามช่วงเวลาการทำงานของเครื่องจักร

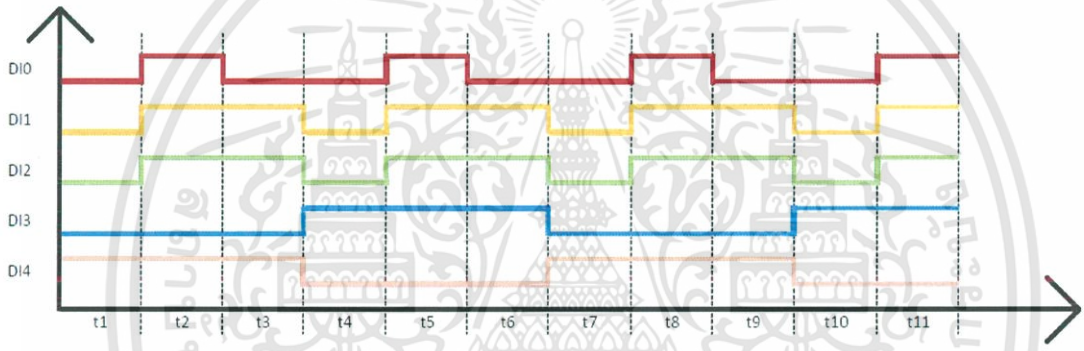


รูปที่ 4.6 จอแสดงผลของอุปกรณ์ฝั่ง Master

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.7 จอแสดงผลของอุปกรณ์ฝั่ง Master (ต่อ)

จากรูปที่ 4.6 และ 4.7 จอแสดงผลจะแสดงอินพุต/เอาต์พุตของเครื่องฉีดน้ำยาเคลือบผิวพิเศษ ที่ได้รับมาจากอุปกรณ์ฝั่ง Slave ว่าได้รับอะไรมาบ้าง



รูปที่ 4.8 กราฟอินพุต/เอาต์พุตของเครื่องจักรที่ได้จากจอแสดงผลของอุปกรณ์ฝั่ง Master

จากรูปที่ 4.8 เป็นการนำอินพุต/เอาต์พุตของเครื่องฉีดน้ำยาเคลือบผิวพิเศษที่อุปกรณ์ฝั่ง Master รับมาแปลงเป็นกราฟเพื่อจะได้เข้าใจง่ายขึ้น โดยแกน X จะเป็นช่วงเวลา ส่วนแกน Y จะเป็นอินพุต/เอาต์พุตของเครื่องจักร ซึ่งหากสังเกตจากกราฟอินพุต/เอาต์พุตของเครื่องฉีดน้ำยาเคลือบผิวพิเศษ ที่อุปกรณ์ฝั่ง Master ได้รับจะสอดคล้องกับหลักการทำงานที่กล่าวมาข้างต้น

4.2.2 ทดสอบดิจิตอลเอาต์พุตของ Slave ที่รับมาจาก Master

ทดสอบโดยจ่ายไฟให้วงจรดิจิตอลอินพุตของอุปกรณ์ฝั่ง Master เพื่อบังคับการทำงานของ Relays Output ที่อุปกรณ์ฝั่ง Slave

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.9 จอแสดงผลของอุปกรณ์ฝั่ง Slave และ Master

จากรูปที่ 4.9 เป็นการส่งสัญญาณอินพุตจากอุปกรณ์ฝั่ง Master ไปควบคุมเอาต์พุตฝั่ง Slave (Relays Output)

4.2.3 ทดสอบอนาล็อกอินพุตจาก Slave ส่งไปยัง Master

ทดสอบโดยจ่ายแรงดันไฟฟ้าให้วงจรอนาล็อกอินพุตที่อุปกรณ์ฝั่ง Slave จากนั้นวัดแรงดันไฟฟ้าที่วงจรอนาล็อกเอาต์พุตของอุปกรณ์ฝั่ง Master

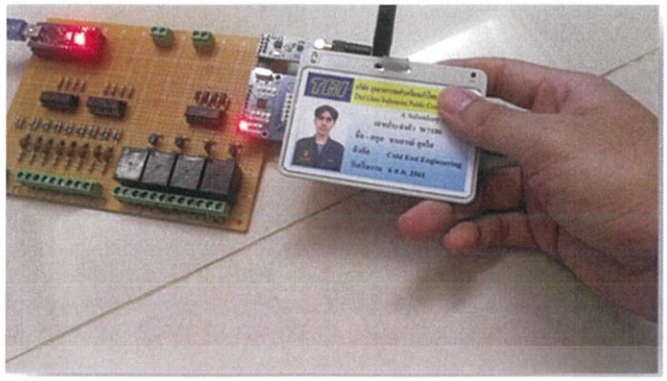
รูปที่ 4.10 แรงดันไฟฟ้าที่วัดได้จากอนาล็อกเอาต์พุตของอุปกรณ์ฝั่ง Master

จากรูปที่ 4.10 แรงดันไฟฟ้าที่วัดได้จากอนาล็อกเอาต์พุตของอุปกรณ์ฝั่ง Master มีค่าเท่ากับ 9.85 V จากการจ่ายแรงดันไฟฟ้าที่อนาล็อกอินพุตของอุปกรณ์ฝั่ง Slave 10 V

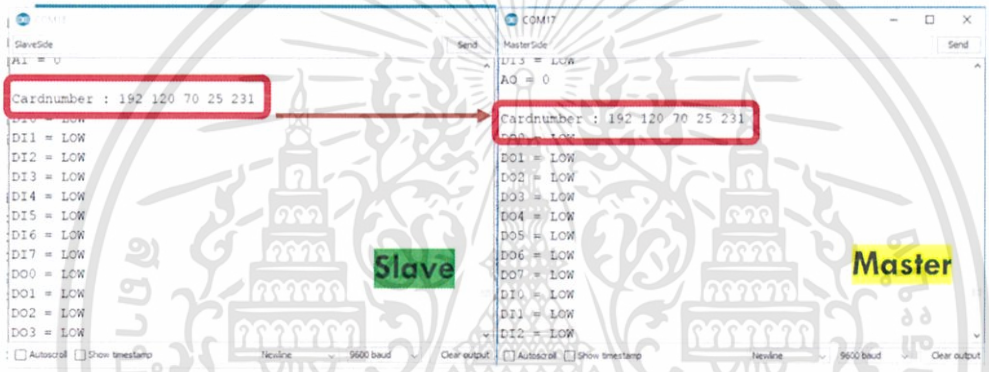
4.2.4 ทดสอบส่งข้อมูล RFID จาก Slave ส่งไปยัง Master

ทดสอบโดยการสแกนบัตรพนักงานที่อุปกรณ์ฝั่ง Slave จากนั้นแสดงผลให้เห็นที่อุปกรณ์ฝั่ง Master

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 บัตรพนักงานและอุปกรณ์ฝั่ง Slave



รูปที่ 4.12 จอแสดงผลของอุปกรณ์ฝั่ง Slave และ Master

จากรูปที่ 4.12 เมื่อทำการสแกนบัตรพนักงาน ที่จอแสดงผลอุปกรณ์ฝั่ง Slave จะแสดงให้เห็นว่าส่งข้อมูลบัตรพนักงานอะไรไป ส่วนที่จอแสดงผลอุปกรณ์ฝั่ง Master จะแสดงให้เห็นว่าได้รับข้อมูลบัตรพนักงานอะไรมา

4.3 ผลการทดสอบ

ตารางที่ 4.1 ผลการทดสอบ

การทดสอบ	ได้	ไม่ได้
ทดสอบดิจิตอลอินพุตจาก Slave ส่งไปยัง Master	✓	
ทดสอบดิจิตอลเอาต์พุตของ Slave ที่รับมาจาก Master	✓	
ทดสอบอนาล็อกอินพุตจาก Slave ส่งไปยัง Master	✓	
ทดสอบส่งข้อมูล RFID จาก Slave ส่งไปยัง Master	✓	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

จากการทดสอบอุปกรณ์รับส่งข้อมูลอินพุต/เอาต์พุตของ PLC ระยะไกลด้วยความถี่ 2.4 GHz ทั้ง 4 การทดลองดังนี้

1. ทดสอบดิจิตอลอินพุตจาก Slave ส่งไปยัง Master
2. ทดสอบดิจิตอลเอาต์พุตของ Slave ที่รับมาจาก Master
3. ทดสอบอนาล็อกอินพุตจาก Slave ส่งไปยัง Master
4. ทดสอบส่งข้อมูล RFID จาก Slave ส่งไปยัง Master

ความสัมพันธ์ของค่าที่อุปกรณ์รับส่งจากจอแสดงผล และการทำงานเครื่องจักร ทำให้เห็นได้ว่าอุปกรณ์สามารถรับส่งค่าอินพุต/เอาต์พุตของเครื่องจักร และข้อมูล RFID ได้จริง

5.2 ปัญหาและวิธีการแก้ไขปัญหา

5.2.1 ปัญหาที่พบ

1. ระยะทางในการรับส่งระหว่าง Master และ Slave ไม่สามารถส่งได้ไกล เนื่องจากในพื้นที่มีเครื่องจักรเป็นจำนวนมาก
2. ยิ่งระยะห่างระหว่าง Master และ Slave เพิ่มมากขึ้น จะทำให้การรับส่งมีความคลาดเคลื่อนมากขึ้น

5.2.2 วิธีการแก้ไขปัญหา

1. การติดตั้งอุปกรณ์อย่างมีประสิทธิภาพ เช่น ติดตั้งให้สูงกว่าเครื่องจักร จะช่วยให้สามารถการรับส่งดียิ่งขึ้น
2. ปรับปรุงแก้ไขเรื่องเสาส่งสัญญาณเพื่อเพิ่มระยะทางในการรับส่ง

5.3 ข้อเสนอแนะ

เนื่องจากอุปกรณ์ต้องใช้โปรแกรม Arduino IDE เพื่อดูจอแสดงผลว่ากำลังรับส่งข้อมูลใดอยู่หรือไม่ ทำให้ไม่สะดวกในการใช้งานมากนัก จึงควรพัฒนาให้ตัวอุปกรณ์สามารถแสดงผลได้ในตัวเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] **nRF24L01** "Arduino nRF24L01 ส่งข้อมูลแบบหลายตัว"
แหล่งที่มา : http://www.semi-journal.com/arduino-nrf24l01/.2561

[2] **Arduino** "Arduino คืออะไร?"
แหล่งที่มา : https://www.thaieasyelec.com/article-wiki/basic-electronics/.2561

[3] **Arduino** "โปรแกรม Arduino IDE เบื้องต้น"
แหล่งที่มา : https://poundxi.com/.2561

[4] **Arduino** "Arduino ชนิดต่างๆ"
แหล่งที่มา : https://www.ioxhop.com/article/2/arduino/.2561

[5] **RFID** "RFID คืออะไร"
แหล่งที่มา : http://www.smartiden.com/index.php?lay=show&ac=article&Id
=539310565&Ntype=12/.2561

[6] **RFID** "RFID Card Read/Write Module Kit (RC522)"
แหล่งที่มา : https://www.thaieasyelec.com/products/rfid-nfc/hf-13-56-mhz/rfid-card-
read-write-module-kit-rc-522-detail.html .2561

[7] **Proteus** "ทำความรู้จักกับโปรแกรม Proteus"
แหล่งที่มา : http://www.shadowwares.com/Tutorials/Proteus/Introduction.html.2561

[8] **Proteus** "เริ่มต้นใช้งาน Proteus"
แหล่งที่มา : http://know2learning.blogspot.com/2014/06/proteus.html.2561

[9] **PCBWizard** "PCBWizard"
แหล่งที่มา : https://sites.google.com/site/pspicesbt/bi-ngan-pcb-wizard.2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

โปรแกรมของอุปกรณ์ฝั่ง Slave

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include "Arduino.h"
#include <SPI.h>
#include <RF24.h>
#include <RFID.h>

#define SS_PIN 10
#define RST_PIN 9
RFID rfid(SS_PIN, RST_PIN);

int inputPins[] = {2,3,6,A0,A1,A2,A3,A4,A7};      //A7
analogIn
int outputPins[] = {4,5,A5,A6};
int Data[14];
int Data2[4];
unsigned long previousMillis = 0,pre = 0;        // will
store last time LED was updated
const long interval = 3000,in = 1000;           //
interval at which to blink (milliseconds)

RF24 radio(7, 8); // CE, CSN
const byte addresses[][6] = {"00001", "00002"};

void setup() {
 Serial.begin(9600);
 Serial.println("THIS IS THE SLAVE CODE");
 SPI.begin();
 rfid.init();
 for (int a=0;a<9;a++) {
  pinMode(inputPins[a],INPUT);
 }
 for (int b=0;b<4;b++) {
  pinMode(outputPins[b],OUTPUT);
 }
 radio.begin();
 radio.openWritingPipe(addresses[1]); // 00001
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  radio.openReadingPipe(1, addresses[0]); // 00002
  radio.setPALevel(RF24_PA_MIN);
  radio.setDataRate(RF24_2MBPS);
  radio.setChannel(124);
}
void loop() {
  unsigned long currentMillis = millis();

  radio.stopListening();
  for (int a=0;a<8;a++) {
    Data[a] = digitalRead(inputPins[a]);
  }
  if (rfid.isCard()) {
    if (rfid.readCardSerial()) {
      if (rfid.serNum[0] != Data[8]
      && rfid.serNum[1] != Data[9]
      && rfid.serNum[2] != Data[10]
      && rfid.serNum[3] != Data[11]
      && rfid.serNum[4] != Data[12]) {
        Data[8] = rfid.serNum[0];
        Data[9] = rfid.serNum[1];
        Data[10] = rfid.serNum[2];
        Data[11] = rfid.serNum[3];
        Data[12] = rfid.serNum[4];
        Serial.print("Cardnumber : ");
        for (int a=0;a<5;a++) {
          Serial.print(rfid.serNum[a],DEC);
          Serial.print(" ");
        }
        Serial.println(" ");
        previousMillis = currentMillis;
      }
    }
  }
  rfid.halt();
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (currentMillis - previousMillis >= interval) {
  for (int a=8;a<13;a++) {
    Data[a] = 0;
  }
}
Data[13] = analogRead(inputPins[8]);
radio.write(&Data,sizeof(Data));

radio.startListening();
unsigned long started_waiting_at = millis();
while(!radio.available()) {
  if (millis() - started_waiting_at > 200 ) {
    Serial.println("No response received - timeout!");
    return;
  }
}
radio.read(&Data2,sizeof(Data2));
for(int a=0;a<4;a++) {
  if(Data2[a] == 0) {
    digitalWrite(outputPins[a], HIGH);
  }
  else if (Data2[a] == 1) {
    digitalWrite(outputPins[a], LOW);
  }
}

if (currentMillis - pre >= in) {
  for (int a=0;a<8;a++) {
    if (Data[a] == 1) {
      Serial.print("DI");
      Serial.print(a);
      Serial.println(" = HIGH");
    }
    else if (Data[a] == 0) {
      Serial.print("DI");
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Serial.print(a);
            Serial.println(" = LOW");
          }
        }
        for(int a=0;a<4;a++) {
          if(Data2[a] == 0) {
            Serial.print("DO");
            Serial.print(a);
            Serial.println(" = LOW");
          }
          else if(Data2[a] == 1) {
            Serial.print("DO");
            Serial.print(a);
            Serial.println(" = HIGH");
          }
        }
        Serial.print("AI = ");
        Serial.print(Data[13]);
        Serial.println("\n");
        pre = currentMillis;
      }
    }
```

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

โปรแกรมของอุปกรณ์ฝั่ง Master

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <SPI.h>
#include <RF24.h>
#include <Wire.h>
#include <Adafruit_MCP4725.h>

Adafruit_MCP4725 dac;

int inputPins[] = {9,5,A6,A7};
int outputPins[] = {2,3,6,A0,A1,A2,A3,4};
int Data[14];
int Data2[4];
int RFID[5];
int x,y;
unsigned long pre = 0;        // will store last time LED
was updated
const long in = 1000;           // interval at which to
blink (milliseconds)

RF24 radio(7, 8); // CE, CSN
const byte addresses[][6] = {"00001", "00002"};

void setup() {
 Serial.begin(9600);
 Serial.println("THIS IS THE MASTER CODE");
 dac.begin(0x60);
 for (int a=0;a<8;a++) {
  pinMode(outputPins[a],OUTPUT);
 }
 for (int b=0;b<4;b++) {
  pinMode(inputPins[b],INPUT);
 }
 radio.begin();
 radio.openWritingPipe(addresses[0]); // 00002
 radio.openReadingPipe(1, addresses[1]); // 00001
 radio.setPALevel(RF24_PA_MIN);
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  radio.setDataRate(RF24_2MBPS);
  radio.setChannel(124);
  radio.startListening();
}
void loop() {
  unsigned long currentMillis = millis();

  if(radio.available()) {
    while(radio.available()) {
      radio.read(&Data, sizeof(Data));
      for(int a=0;a<8;a++) {
        if(Data[a] == 0) {
          digitalWrite(outputPins[a], HIGH);
        }
        else if (Data[a] == 1) {
          digitalWrite(outputPins[a], LOW);
        }
      }
      if(Data[8] != RFID[0]
      &&Data[9] != RFID[1]
      &&Data[10] != RFID[2]
      &&Data[11] != RFID[3]
      &&Data[12] != RFID[4]) {
        RFID[0] = Data[8];
        RFID[1] = Data[9];
        RFID[2] = Data[10];
        RFID[3] = Data[11];
        RFID[4] = Data[12];
        if(RFID[0] != 0
        &&RFID[1] != 0
        &&RFID[2] != 0
        &&RFID[3] != 0
        &&RFID[4] != 0) {
          Serial.print("Cardnumber : ");
          for(int i=0;i<5;i++){
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
          Serial.print(RFID[i]);
          Serial.print(" ");
        }
        Serial.println(" ");
      }
    }
    x = Data[13];
    y = map(x,0,1023,0,4095);
    dac.setVoltage(y,false);
  }
  radio.stopListening();
  for(int a=0;a<4;a++) {
    Data2[a] = digitalRead(inputPins[a]);
  }
  radio.write(&Data2,sizeof(Data2));
  radio.startListening();
}
else if(!radio.available()) {
  for(int a=0;a<8;a++) {
    digitalWrite(outputPins[a], HIGH);
  }
}

if(currentMillis - pre >= in) {
  for(int a=0;a<8;a++) {
    if(Data[a] == 0) {
      Serial.print("DO");
      Serial.print(a);
      Serial.println(" = LOW");
    }
    else if(Data[a] == 1) {
      Serial.print("DO");
      Serial.print(a);
      Serial.println(" = HIGH");
    }
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    for(int a=0;a<4;a++) {
      if(Data2[a] == 1) {
        Serial.print("DI");
        Serial.print(a);
        Serial.println(" = HIGH");
      }
      else if (Data2[a] == 0) {
        Serial.print("DI");
        Serial.print(a);
        Serial.println(" = LOW");
      }
    }
    Serial.print("AO = ");
    Serial.print(y);
    Serial.println("\n");
    pre = currentMillis;
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ - นามสกุล	นายชนสรณ์ สุกใส
วัน เดือน ปีเกิด	17 มีนาคม 2540
ที่อยู่	25/93 หมู่ที่ 5 ต.เขารูปช้าง อ.เมืองสงขลา จ.สงขลา 90000
E-mail	bank.chanasorn@gmail.com
โทรศัพท์	085-896-4429

ประวัติการศึกษา

พ.ศ. 2552 – 2557	ระดับมัธยมศึกษา โรงเรียนมหาวชิราวุธ จังหวัดสงขลา
พ.ศ. 2558 – ปัจจุบัน	วิศวกรรมศาสตร์บัณฑิต หลักสูตรวิศวกรรมการวัดคุม ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประสบการณ์ทำงาน

- นักศึกษาฝึกงาน แผนกวิศวกรรมเครื่องมือวัด บริษัท อุตสาหกรรมทำเครื่องแก้วไทย จำกัด (มหาชน)
- นักศึกษาโครงการสหกิจศึกษา แผนกวิศวกรรมหน้าราง บริษัท อุตสาหกรรมทำเครื่องแก้วไทย จำกัด (มหาชน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้