รายงานสหกิจศึกษาฉบับสมบูรณ์

การแก้ปัญหาสกรูติดค้างในแผงหน้าปัดรถยนต์โดยเครื่องจ่ายสกรู

Improvement of Bonus Part at Instrument Panel by Screw Feeding Machine

นายพชร พัชรมงคล

ภาควิชาวิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2562

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงงานสหกิจศึกษา การแก้ปัญหาสกรูติดค้างในแผงหน้าปัดรถยนต์โดยเครื่องจ่ายสกรู
ชื่อ-สกุล นักศึกษา นายพชร พัชรมงคล
คณะ วิศวกรรมศาสตร์ **ภาควิชา** วิศวกรรมเครื่องกล สาขาวิศวกรรมเครื่องกล
ชื่อ-สกุล อาจารย์นิเทศ ผศ.ดร.ณัฐวุฒิ เดไปวา
ชื่อ-สกุล ผู้นิเทศงาน นายทศศาสตร์ นากาย
ชื่อสถานประกอบการ บริษัท ฟอร์ด มอเตอร์ คัมปะนี (ประเทศไทย) จำกัด

บทคัดย่อ

จากการปฏิบัติงานสหกิจศึกษา ผู้ศึกษาได้เรียนรู้ระบบการทำงาน ขั้นตอนการดำเนินการผลิตตลอดจนกระบวนการแก้ปัญหาภายในสายการผลิตรถกระบะฟอร์ด ผู้ศึกษาได้ศึกษาหลักการ Six-Sigma ซึ่งเป็นหลักมาตรฐานว่าด้วยการนำความรู้ทางสถิติมาประยุกต์ใช้ในการแก้ปัญหาภายในโรงงาน และได้นำหลักการนี้มาแก้ปัญหาชิ้นส่วนติดค้างภายในแผงหน้าปัดรถยนต์ ซึ่งจะทำให้เกิดเสียงกระเทาะซึ่งสร้างความรำคาญขณะการใช้งาน หลังจากการวิเคราะห์ปัญหาแล้ว พบว่าสาเหตุหลักของปัญหานี้มาจากการที่ไม่สามารถควบคุมปริมาณการหยิบสกรูได้ ส่งผลให้สกรูส่วนเกินตกลงไปติดค้างในแผงหน้าปัดระหว่างการประกอบ ผู้ศึกษาจึงได้คิดประดิษฐ์เครื่องจ่ายสกรูโดยใช้โปรแกรม Catia ในการออกแบบชิ้นส่วนของเครื่อง และใช้เครื่องพิมพ์ 3 มิติในการสร้างชิ้นงาน จากนั้นผู้ศึกษาได้ใช้แผงควบคุม Arduino และภาษาซีในการควบคุมวงจรการทำงานของตัวเครื่อง หลังจากการใช้งานเครื่องจ่ายสกรูที่สายการผลิตแล้วแล้วพบว่าปัญหาชิ้นส่วนติดค้างภายในแผงหน้าปัดลดลง 79.2 เปอร์เซ็นต์

คำสำคัญ : เครื่องจ่ายสกรู,Six-Sigma, Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Cooperative Title : Improvement of bonus part at instrument panel assembly by screw feeding machine

Student intern name : Pachara Patcharamongkol

Faculty : Engineering **Department** : Mechanical Engineering

Advisor name : Asst. Prof. Dr. Nutthawut Depaiwa

Mentor Name : Mr.Tosasart Nakay

Company : Ford Motor Company (Thailand) Ltd.

ABSTRACT

From the cooperative education operation, I have learned the working system, production process along with the problem-solving process at the Ford pickup truck production line. I studied Six-Sigma principles, which are standard principles for applying statistical knowledge to solve problems in the factory. I used this principle to solve the problem of bonus part inside the instrument panel. This issue will cause annoying rattle noise during the use. After analyzing the problem, I found that the main cause is the inability to control the amount of screw picking resulting in excess screws falling into the instrument panel during the assembly process. I, therefore, invented a screw feeding machine using the Catia program to design the mechanical parts and use 3D printers to create workpieces. Then, I used the Arduino board and the C language to control the operation of the machine. After using the screw feeding machine, it was found that the problem of the bonus part inside the instrument panel was reduced by 79.2 percent.

Keyword : Screw feeding machine,Six-Sigma, Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานเรื่อง “การแก้ปัญหาสกรูติดค้างในแผงหน้าปัดรถยนต์โดยเครื่องจ่ายสกรู” สามารถสำเร็จลุล่วงไปได้ด้วยดีด้วยความช่วยเหลือและคำแนะนำจากอาจารย์และบุคลากรหลายท่าน ผู้ศึกษาจึงขอขอบพระคุณอาจารย์และบุคลากรดังนี้

ผศ.ดร.ณัฐวุฒิ เดไปวา อาจารย์ที่ปรึกษา ผู้ซึ่งคอยให้คำปรึกทางด้านวิชาการที่เกี่ยวข้องกับการทำโครงงาน ตลอดจนคำแนะนำเกี่ยวกับการทำสหกิจศึกษา

นายสืบสกุล แสงเฟื่อง ผู้จัดการแผนกวิศวกรรมการผลิต และนายทศศาสตร์ นากาย ผู้นิเทศงานในการปฏิบัติสหกิจ ซึ่งคอยให้ความรู้ และคำชี้แนะเกี่ยวกับกระบวนการประกอบรถกระบะฟอร์ด ทั้งยังแก้ไขปัญหาระหว่างการปฏิบัติสหกิจ ทำให้โครงงานสหกิจศึกษาสามารถลุล่วงสำเร็จไปได้ด้วยดี

ขอขอบคุณ ฟอร์ด ไทยแลนด์ แมนูเฟคเจอริ่ง (เอฟทีเอ็ม) ที่ให้โอกาสได้เข้าไปปฏิบัติสหกิจศึกษา พร้อมอำนวยความสะดวกตลอดระยะเวลาการดำเนินการโครงงาน

พชร พัชรมงคล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญรูป VI

บทที่ 1 บทนำ 1

1.1 รายระเอียดเกี่ยวกับองค์กร 1

1.2 ที่มาและความสำคัญ 1

1.3 วัตถุประสงค์ของการวิจัย 2

1.4 ขอบเขตของการวิจัย 2

1.5 วิธีดำเนินการวิจัย 2

1.6 ประโยชน์ที่คาดว่าจะได้รับ 3

บทที่ 2 แนวคิดและข้อมูลที่เกี่ยวข้อง 4

2.1 หลักการ Six-Sigma 4

2.1.1 ประวัติและที่มาของ Sig Sigma 4

2.1.2 ขั้นตอนการทำงานของ Six Sigma 4

2.2 Arduino 8

2.2.1 ข้อมูลทั่วไปของ Arduino Uno R3 10

2.2.2 ส่วนประกอบของบอร์ด Arduino Uno R3 11

2.2.3 การเขียนโปรแกรม 11

2.3 Catia 13

บทที่ 3 วิธีดำเนินการวิจัย 15

3.1 ศึกษาปัญหาที่เกิดขึ้นภายในโรงประกอบ 15

3.2 หาสาเหตุของปัญหาเสียงกระเทาะในแผงหน้าปัด 16

3.3 ทดสอบความสามารถการวัดของผู้ตรวจคุณภาพรถ 16

3.2.4 ศึกษาขั้นตอนการประกอบแผงหน้ารถ 18

3.4 หาสาเหตุของการมีสกรูติดอยู่ในแผงหน้าปัด(Bonus Part) 19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

3.4.1 สมมติฐานที่ 1 การหยิบสรูเกินกว่าที่จำเป็น 19
3.4.2 สมมติฐานที่ 2 เบ้ายึดที่ปืนขันแน่นยึดสกรูไม่แน่น 21
3.5 การหาแนวทางแก้ปัญหาการหยิบสกรูเกิน 22
3.6 การออกแบบเครื่องจ่ายสกรู 22
3.6.1 ส่วนฐาน 23
3.6.2 แผ่นรองรับสกรูและที่บรรจุสกรู 23
3.6.3 แผ่นรองรับสกรูด้านนอก 25
3.6.4 อุปกรณ์จ่ายสกรู 27
3.7 การต่อวงจรควบคุม 29
3.7.1 IR Infrared Obstacle Avoidance Sensor Module 29
3.7.2 เซอร์โวมอเตอร์ S90G 30
3.7.3 มอเตอร์ไฟฟ้ากระแสตรง และแผงควบคุม L298N 32
3.7.4 สวิทช์ปรับปริมานสกรู 33
3.7.5 หลอดไฟ LED 34
3.8 เขียนโปรแกรม Aruidno IDE 36
3.9 ทดสอบการทำงานของเครื่อง 37
3.10 บันทึกผล 37
บทที่ 4 ผลการดำเนินงาน 38
4.1 การเปลี่ยนขั้นตอนการทำงาน 38
4.2 ข้อมูลรายละเอียดของเครื่องจ่ายสกรู 39
4.3 ผลการทำงานของเครื่องจ่ายสกรู 39
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ 41
5.1 สรุปผลการวิจัย 41
5.2 ข้อเสนอแนะและปัญหาที่เกิดขึ้น 41
เอกสารอ้างอิง 43
ภาคผนวก 44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

1.1 รถกระบะฟอร์ด 1

1.2 กราฟแสดงปัญหาเสียงกระเทาะภายในรถกระบะฟอร์ดเนื่องจากชิ้นส่วนติดค้างในแผงหน้าปัด..... 2

2.1 แผนภาพก้างปลา..... 5

2.2 แผนภาพ X-Y กำหนดความสำคัญของตัวแปร 7

2.3 การทดสอบสมมติฐานด้วยซอฟต์แวร์ Minitap 8

2.4 บอร์ด Arduino ต่อกับ LED และบอร์ด XBee Shield..... 9

2.5 บอร์ด Arduino Uno R3..... 10

2.6 ชิ้นส่วนของบอร์ด Arduino Uno R3..... 11

2.7 ฟังค์ชั่นการใช้งาน Catia 14

2.8 การใช้งาน Catia ในอุตสาหกรรมรถยนต์..... 14

3.1 พารีโตแสดงปัญหาเสียงกระเทาะทั้งหมด..... 15

3.2 ผังพาเรโตแสดงสาเหตุของปัญหาเสียงกระเทาะที่แผงหน้าปัด..... 16

3.3 สนามทดสอบเสียงกระเทาะ 17

3.4 การทำ Measurement System Analysis(MSA)..... 17

3.5 แผงหน้าปัดรถ(Instrument Panel) IP skin(ซ้าย) และCross Car Beam(ขวา)..... 18

3.6 แสดงการประกอบแผงหน้าปัด 1-2..... 18

3.7 แสดงการประกอบแผงหน้าปัด 3-4..... 19

3.8 แผนภาพก้างปลาแสดงสาเหตุของปัญหาสกรูติดค้างในแผงหน้าปัดรถ 19

3.9 กราฟแสดงปริมานการหยิบสกรู..... 20

3.10 กราฟแสดงการทดสอบสมมติฐาน..... 20

3.11 การทดสอบการยึดแน่นของเบ้าปีนขันแน่น..... 21

3.12 การทดสอบสมมติฐานเบ้ายึดสกรู..... 21

3.13 เครื่องจ่ายสกรูทั่วไป..... 22

3.14 การออกแบบเครื่องนับสกรูอัตโนมัติ 22

3.15 การออกแบบชิ้นส่วนฐาน..... 23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

3.16 การเทียบขนาดมอเตอร์กับชิ้นส่วนฐาน 23
3.17 แผ่นรองรับสกรู 24
3.18 รูที่ใช้ประกอบกับแกนมอเตอร์ไฟฟ้า 24
3.19 กลไกการจ่ายสกรู 25
3.20 แขนจับยึด 25
3.21 แผ่นรองรับสกรูด้านนอก 26
3.22 การรองรับสกรู 26
3.23 การประกอบแผ่นรองรับสกรูด้านนอกเข้ากับฐานของเครื่อง 27
3.24 ชิ้นส่วนย่อยของอุปกรณ์จ่ายสกรู 27
3.25 อุปกรณ์จ่ายสกรู 28
3.26 การประกอบชิ้นส่วนต่างๆเข้าด้วยกัน 28
3.27 เสร็จสิ้นการประกอบและออกแบบเครื่องจ่ายสกรู 29
3.28 โมดูลเซ็นเซอร์แสงสำหรับตรวจจับวัตถุกีดขวาง(ซ้าย) และเซนเซอร์รูปตัวยู(ขวา) 29
3.29 การต่อเซนเซอร์อินฟราเรดเข้ากับบอร์ด Arduino 30
3.30 แสดงไดอะแกรมการทำงานของเซอร์โวมอเตอร์ 31
3.31 การต่อวงจรเซอร์โวมอเตอร์ S90G เข้ากับบอร์ด Arduino 31
3.32 มอเตอร์ไฟฟ้ากระแสตรง(ซ้าย) และแผงควบคุม L298N (ขวา) 32
3.33 แสดงรายละเอียดของบอร์ด L298N 32
3.34 แสดงการต่อใช้งานอุปกรณ์ 33
3.35 การต่อสวิทช์แบบ Pull-Up และ Pull-Down 34
3.36 การต่อหลอดไฟ LED เข้ากับบอร์ด Arduino 35
3.37 แผงวงจรเครื่องจ่ายสกรู 35
3.38 แสดงผังการทำงานของเครื่องจ่ายสกรู 36
3.39 ชิ้นงานหลังจากประกอบแล้ว 36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

3.40 การติดตั้งเครื่องจ่ายสกรูภายในสายการผลิต....37
4.1 แสดงการหยิบสกรูจากกล่องข้างสายการประกอบ....38
4.2 แสดงการรับสกรูจากเครื่องจ่ายสกรู....39
4.3 แสดงปริมาณการหยิบสกรูก่อนและหลังใช้เครื่องจ่ายสกรู....40
4.4 การเพิ่มความแม่นยำในการหยิบสกรูหลังใช้เครื่องจ่ายสกรู....40
5.1 แสดงปัญหาชิ้นส่วนติดค้างในแผงหน้าปัดทั้งหมดตั้งแต่เดือนมิถุนายน-พฤษจิกายน....41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 รายระเอียดเกี่ยวกับองค์กร

บริษัท ฟอร์ด มอเตอร์ คัมปะนี (ประเทศไทย) จำกัด ตั้งโรงงานผลิตรถยนต์ชื่อ ฟอร์ด ไทยแลนด์ แมนูเฟคเจอริ่ง (เอฟทีเอ็ม) ตั้งอยู่ที่ 500/103 หมู่ 3 นิคมอุตสาหกรรมดับบลิวเอชเอ อีสเทิร์นซีบอร์ด ตำบลตาสิทธิ์ อำเภอปลวกแดง จังหวัดระยอง 21140 ก่อตั้งอย่างเป็นทางการเมื่อวันที่ 3 พฤกษภาคม 2555 โดยบริษัท ฟอร์ด มอเตอร์ คัมปะนี ได้ลงทุนกว่า 1.4 หมื่นล้านบาทในฐานะที่ประเทศไทยเป็นฐานการผลิตสำคัญของฟอร์ดและศูนย์กลางการส่งออกไปทั่วภูมิภาค ซึ่งในปี 2559 บริษัทฟอร์ด มอเตอร์ คัมปะนี ได้เพิ่มการลงทุนจำนวน 6,269 ล้านบาท ในการขยายกำลังการผลิตรถกระบะฟอร์ด เรนเจอร์ ที่โรงงานเอฟทีเอ็ม เพื่อรองรับกับความต้องการ ของผู้บริโภคที่เพิ่มสูงขึ้นทั่วเอเชียแปซิฟิก

โรงงานเอฟทีเอ็มมีพื้นที่ขนาด 200,000 ตารางเมตร มีศักยภาพในการรองรับการจ้างงานถึง 11,000 ตำแหน่ง แบ่งเป็นการจ้างงานโดยตรงกับฟอร์ด 2,200 ตำแหน่ง และการจ้างงานผ่านเครือข่ายผู้ผลิตชิ้นส่วนและผู้จำหน่าย 8,800 ตำแหน่ง ฟอร์ด เป็นโรงงานประกอบรถยนต์ที่ทันสมัยและรองรับการผลิตได้หลายประเภท โรงงานแห่งนี้ได้รับการบูรณาการอย่างรอบด้านเพื่อรองรับการปั๊มขึ้นรูปชิ้นส่วนประกอบตัวถัง พ่นสี และการประกอบรถยนต์ ซึ่งโมเดลที่ทำการผลิตอยู่ ณ ปัจจุบันคือ ฟอร์ด เจนเจอร์ โมเดลของปี 2019 ได้แก่ Ford Ranger new model 2019 และ Ford Ranger Raptor

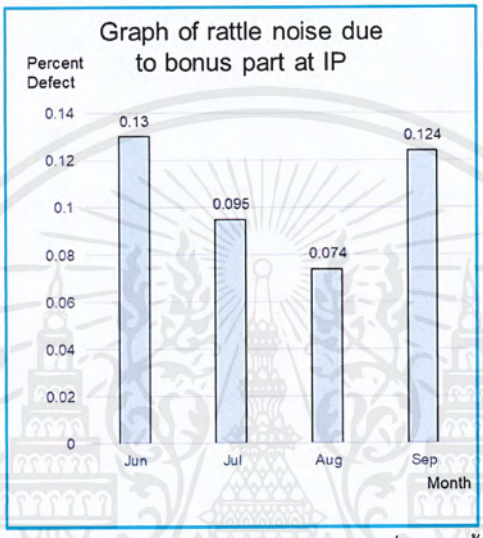
รูปที่ 1.1 รถกระบะฟอร์ด

1.2 ที่มาและความสำคัญ

ในการผลิตรถกระบะจากเดือนมิถุนายน 2019 ถึง เดือนกันยายน 2019 เกิดปัญหาเสียงกระเทาะภายในรถยนต์ราว 0.562 เปอร์เซ็นต์ โดยสาเหตุของปัญหามีหลายประการ ทั้งนี้ผู้ศึกษาได้เลือกแก้ไขปัญหาบริเวณแผงหน้าปัดรถเนื่องจากเป็นบริเวณที่เกิดปัญหาบ่อยครั้ง อีกทั้งบริเวณนี้ยังตรวจสอบและแก้ไขปัญหาได้ยาก โดยจำเป็นต้องให้ผู้ตรวจสอบนั่งภายในห้องโดยสารของรถ และขับรถนั้นบนพื้นถนนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขรุขระ โดยหากรถคันนั้นเกิดปัญหา ก็จะมีเสียงผิดปกติเกิดขึ้นเนื่องจากการกระทบกันระหว่างสกรูและชิ้นส่วนข้างเคียง จากนั้นรถจะถูกนำมาวินิจฉัยโดยละเอียดอีกครั้ง เพื่อหาตำแหน่งของสกรูที่เกินมา และนำสกรูออก โดยมีค่าใช้จ่ายประมาณ 6000 บาทต่อคัน ผู้ศึกษาจึงทำการวิเคราะห์และพบว่าสาเหตุที่สกรูติดค้างภายในแผงหน้าปัดรถส่วนใหญ่มาจากการที่ผู้ประกอบงานมีแนวโน้มที่จะหยิบสกรูมากเกินความจำเป็น ซึ่งเพิ่มโอกาสที่สกรูจะตกไปในแผงหน้าปัด ผู้ศึกษาจึงออกแบบเครื่องจ่ายสกรูอัตโนมัติขึ้น เพื่อควบคุมจำนวนสกรูให้มีปริมาณตรงตามที่ใช้ในการประกอบ ลดปัญหาสกรูตกลงไปในแผงหน้าปัด



รูปที่ 1.2 กราฟแสดงปัญหาเสียงกระเทาะภายในรถกระบะฟอร์ดเนื่องจากชิ้นส่วนติดค้างในแผงหน้าปัด

1.3 วัตถุประสงค์ของการวิจัย

1.2.1 แก้ไขปัญหาเสียงกระเทาะในแผงควบคุมหน้าอันเนื่องมาจากการประกอบชิ้นส่วนเกิน

1.2.2 สร้างมาตรฐานการหยิบ-จ่ายสกรูที่สามารถควบคุมและตรวจสอบได้ในสายการผลิตรถยนต์

1.4 ขอบเขตของการวิจัย

1.3.1 วิเคราะห์และแก้ไขปัญหาเสียงกระเทาะ เนื่องจากมีสกรูติดค้างในแผงหน้าปัดรถ

1.3.2 แก้ปัญหาที่เกิดขึ้นโดยใช้โปรแกรม Catia และ แผงควบคุม Arduino ในการออกแบบและสร้างเครื่องจ่ายสกรูอัตโนมัติ

1.5 วิธีดำเนินการวิจัย

1.5.1 ศึกษาปัญหาที่เกิดขึ้นภายในโรงประกอบ

1.5.2 หาสาเหตุของปัญหาเสียงกระเทาะในแผงหน้าปัด

1.5.3 ทดสอบความสามารถการวัดของผู้ตรวจคุณภาพรถ

1.5.4 หาสาเหตุของการมีสกรูติดอยู่ในแผงหน้าปัด(Bonus Part)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5.5 การหาแนวทางแก้ปัญหาการหยิบสกรูเกิน

1.5.6 การออกแบบเครื่องจ่ายสกรู

1.5.7 การต่อวงจรควบคุม

1.5.8 เขียนโปรแกรม Aruidno IDE

1.5.9 ทดสอบการทำงานของเครื่อง

1.5.10 บันทึกผล

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 เรียนรู้กระบวนการผลิตรถกระบะ และการแก้ปัญหาในสายการผลิต

1.5.2 ได้เครื่องนับสกรูเพื่อใช้แก้ปัญหาสกรูติดค้างภายในแผงหน้าปัดรถ

1.5.3 เรียนรู้การใช้งานแผงควบคุมArduino และอุปกรณ์อิเล็กทรอนิกส์อื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
แนวคิด และข้อมูลที่เกี่ยวข้อง

2.1. หลักการ Six-Sigma

2.1.1 ประวัติและที่มาของ Six Sigma

Sig-Sigma เป็นระบบการจัดการคุณภาพในองค์กรที่มีคุณภาพมากระบบหนึ่ง ที่ยอมให้มีของเสียในระบบได้เพียง 3.4 ชิ้น ต่อการผลิตสินค้าล้านชิ้น เป็นระบบที่เน้นความร่วมมือของทุกคนในองค์กรโดยไล่มาตั้งแต่ผู้บริหารไปจนถึงบุคลากรในองค์กรทุกๆคน เน้นการลดความผิดพลาด ลดความสูญเปล่า ลดการแก้ไขตัวชิ้นงานและสอนให้พนักงานในองค์กรดำเนินธุรกิจอย่างมีหลักการ เป็นระบบที่จะไม่พยายามแก้ไขปัญหา แต่เน้นการกำจัดปัญหา หรือที่เรียกว่า เป็นการรวมกันระหว่าง “อานุภาพแห่งคน (Power of people)” และ“อานุภาพแห่งกระบวนการ (Process Power)” มีเป้าหมายสูงสุดเพื่อสร้างรายได้ ลดค่าใช้จ่ายของบริษัทและนำไปสู่การสร้างความพึงพอใจให้แก่ลูกค้า

Six Sigma เป็นการประยุกต์ความรู้ทางด้านสถิติภายใต้สมมุติฐานที่ว่า 1) ทุกสิ่งทุกอย่าง คือ กระบวนการ 2) กระบวนการทุกอย่างมีความแปรปรวนตลอดเวลา 3) การนำข้อมูลมาวิเคราะห์ ทำเพื่อให้เข้าใจความแปรปรวนและนำไปสู่การพัฒนาและปรับปรุงกระบวนการให้ดียิ่งขึ้น

Sigma : S ในทางสถิติใช้แทนระดับความผันแปรของกระบวนการ Six Sigma มีการนำความรู้ทางสถิติมาใช้ ซึ่งความหมายเชิงทฤษฎี Six Sigma คือความพยายามลดความผันแปรของกระบวนการโดยบีบให้ความผันแปรทั้งหมดอยู่ภายใต้ขีดจำกัดของข้อกำหนดด้านคุณภาพ โดยสมมุติให้ปรากฏการณ์ที่เกิดในระบบนั้นมีการแจกแจงแบบปกติ (Normal Distribution) หรือการกระจายเป็นรูประฆังคว่ำทั้งหมด โดยค่าเฉลี่ยที่จุดกึ่งกลางของการกระจายตัวนั้นคือ ค่าที่ต้องการ ส่วน ซิกมา คือ หนึ่งช่วงของความเบี่ยงเบนมาตรฐาน ที่วัดจากจุดกึ่งกลางดังกล่าว ซึ่งถ้าตัว Six sigma มีค่าสูงหรือมีความผันแปรมากขึ้นเท่าไรก็เปรียบเสมือนมีการทำข้อผิดพลาดน้อยลงเท่านั้น ซึ่งโอกาสที่จะเกิดข้อผิดพลาดตัวนี้เรียกว่า DPMO (Defects Per Million Opportunities)

2.1.2 ขั้นตอนการทำงานของ Six Sigma

Six Sigma Project เป็นโครงการย่อยที่จัดตั้งขึ้นเพื่อแก้ปัญหา โดยต้องพิจารณาระดับปัญหาในปัจจุบันและศักยภาพที่จะเกิดประโยชน์จากการแก้ไขปัญหา ซึ่งทุกโครงการจะต้องดำเนินการผ่านขั้นตอนของ Six Sigma หรือที่เรียกว่า D-M-A-I-C มีขั้นตอนดังนี้

2.1.2.1 Define

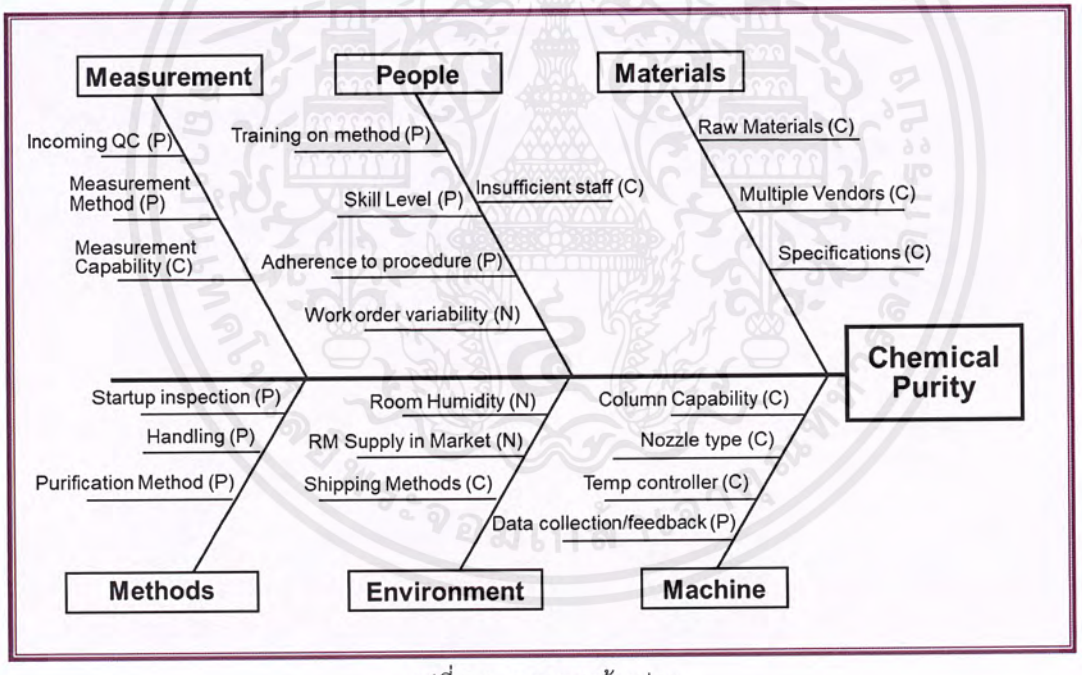
การกำหนดหัวข้อปัญหาที่มีผลกระทบสูง สาเหตุของปัญหารวมทั้งตัวแปรทั้งหมดที่เกี่ยวข้องกับปัญหา ถือเป็นส่วนที่ท้าทายและยากที่สุดสำหรับการทำโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนภาพก้างปลา ใช้ในการระดมความคิดเพื่อหาสาเหตุทั้งหมด ที่คาดว่ามีความเป็นไปได้ที่จะทำให้เกิดปัญหา โดยมีการแจกแจงสาเหตุเหล่านี้เป็น 6 ประเภท คือ คน เครื่องจักร วิธีการ วัสดุ การวัด และสิ่งแวดล้อม

2.1.2.2 Measure

การวัดความสามารถของกระบวนการและเครื่องมือที่ใช้ในการแก้ไขปัญหา รวมถึงการกรองหาสาเหตุของปัญหาต่อจากขั้น Define โดยใช้แผนภาพก้างปลา ในแผนภาพนี้มีการจัดประเภทของสาเหตุทั้งหมด โดย C : Controllable คือ ตัวแปรสามารถควบคุมหรือเปลี่ยนแปลงได้ เช่น ความเร็วรอบ อุณหภูมิ อัตราการป้อน P : Procedural คือตัวแปรที่ได้รับผลกระทบผ่านกระบวนการมาตรฐานที่สร้างขึ้นเพื่อสร้างความสมบูรณ์ของขั้นตอน เช่น การจัดเรียงชิ้นส่วน และ N : Noise คือตัวแปรที่ควบคุมไม่ได้ ควบคุมได้ยาก หรือมีราคาที่สูงเกินไป หลังจากจัดประเภทแล้วจะตัดตัวแปล N ออกและเลือกศึกษาเฉพาะตัวแปร C และ P



รูปที่ 2.1 แผนภาพก้างปลา

MSA (Measure System Analysis) คือ การตรวจสอบความสามารถและประสิทธิภาพของเครื่องมือวัดและผู้วัด ข้อผิดพลาดจากระบบการวัดอาจส่งผลกระทบอย่างมากในการวิเคราะห์ข้อมูล โดยธรรมชาติทุกกระบวนการจะมีความผันแปรนี้แฝงอยู่ แหล่งความผันแปรเชื่อมโยงไปถึงลักษณะทางกลศาตร์ของกระบวนการกับวัตถุดิบและเสถียรภาพของคนทำงาน ความผันแปรนี้จะวิเคราะห์ด้วยกระบวนการ MSA ซึ่งเป็นขั้นตอนสำคัญในการประยุกต์ใช้การควบคุมคุณภาพในทุกๆส่วน ความผันแปรแยกเป็นสองประเภท ประเภทแรก คือ ความแม่นยำ แบ่งเป็น Repeatability คือความสามารถในการวัดซ้ำของคนวัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรืออุปกรณ์วัด และReproducibility ความสามารถในการวัดซ้ำระหว่างคนวัดหรือระหว่างเครื่องมือวัด ส่วนประเภทที่สอง คือ ความเที่ยงตรง แบ่งเป็น Stability ความเที่ยงตรงเมื่อเวลาผ่านไป, Linearity ความเที่ยงตรงเมื่อเปลี่ยนย่านวัด, Resolution ความละเอียดเครื่องมือวัดและ Bias ค่าที่ต่างไปจากค่าจริง

GRR repeatability คือการตรวจสอบการวัดด้วยข้อสงสัยว่าระบบการวัดจะให้ผลวัดเดียวกันหรือไม่ เมื่อคนวัดคนเดิมกับชิ้นงานเดิมโดยที่คนวัดเหมือน "คนตาบอด" ไม่ทราบว่าวัดชิ้นงานเดิมหลายครั้ง คนวัดจะมีความเสถียรในการวัดชิ้นงานเดิมหรือไม่เมื่อใช้เครื่องมือวัดเดิม โดย Repeatability จะแสดงถึงความผันแปรของอุปกรณ์การวัดด้วยเพราะว่าหลายครั้งผลสะท้อนกลับไปเรื่องการออกแบบอุปกรณ์หรือเงื่อนไขการวัด

GRR Reproducibility คือการตรวจสอบการวัดด้วยข้อสงสัยว่าระบบการวัดจะให้ผลวัดเดียวกันหรือไม่เมื่อมีสองหรือสามคนวัดชิ้นงานเดิมโดยที่คนวัดเหมือน "คนตาบอด" ไม่ทราบว่าวัดชิ้นงานเดิมหลายครั้ง ความสามารถในการวัดซ้ำระหว่างคนวัดเป็นอีกมิติของแหล่งความผันแปรของกระบวนการวัด Reproducibility ยังแสดงถึงความผันแปรของคนวัดด้วยเพราะว่านัยสำคัญส่วนใหญ่สะท้อนไปที่ความเสถียรของคนวัดที่อยู่ในระบบการวัด หนทางแก้ไขของ Reproducibility ส่วนใหญ่แล้วจะเป็นการฝึกอบรมคนวัด

การศึกษา Gage R&R ปกติทำขึ้นจากสองส่วน : ความผันแปรจากเครื่องมือวัดและคนวัด สองส่วนนี้ส่งให้เรามุ่งไปสนใจการแก้ไขให้ถูกจุดในการซ่อมหรือปรับปรุงเครื่องมือวัดให้ดีขึ้น(EV-ความผันแปรจากเครื่องมือวัด) หรือฝึกอบรมคนวัดให้ใช้เครื่องมือวัดได้อย่างถูกต้องหรือให้มีความเสถียรยิ่งขึ้น(AV-ความผันแปรจากคนวัด) ค่าพิกัด (USL-LSL) หรือค่าความผันแปรกระบวนการ(6 เท่าของค่าส่วนเบี่ยงเบนมาตรฐานที่มาจากข้อมูล SPC ที่นิ่งแล้ว) หรือความผันแปรที่ได้จากการศึกษา(6 เท่าของค่าส่วนเบี่ยงเบนมาตรฐานของข้อมูลที่ได้จากการศึกษาระบบการวัด) ทั้งสองค่าสามารถใช้ในการคำนวณสัดส่วนที่เกิดจากความผิดพลาดจากระบบการวัด ข้อแนะนำสำหรับค่าที่ยอมรับได้ของ Repeatability และ Reproducibility (% R&R) ความผิดพลาดต้องน้อยกว่า 10% , ความผิดพลาดของค่าวัดที่ยอมรับได้คือ 10%-30% เกณฑ์การยอมรับระบบการวัดจะให้ความสำคัญกับต้นทุนของการใช้เครื่องมือวัดเป็นสำคัญ ต้นทุนการซ่อมแซมหรือการปรับปรุงความสามารถกระบวนการที่เป็นอยู่ ถ้าหากความผิดพลาดของค่าวัดมากกว่า 30% ระบบการวัดต้องการการปรับปรุง (ขึ้นอยู่กับความสามารถกระบวนการด้วย) ทุกๆการดำเนินการเพื่อระบุปัญหาและทำให้ถูกต้อง

แผนภาพ X-Y ในรูปที่ 2.4 เป็นอีกหนึ่งเครื่องมือที่ใช้ในการจัดลำดับความสำคัญของปัญหา โดยมีการให้คะแนน ระหว่าง 1-9 กับสาเหตุของปัญหา(X) และปัญหา(Y) หลังจากนั้นหาผลรวมของผลคูณระหว่างสาเหตุกับปัญหาเพื่อจัดอันดับความสำคัญ และเลือกแก้สาเหตุที่มีคะแนนสูงที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XY Diagram

Demo | Delete | View Summary

Process:	laminating
Date:	5/8/2006

Ranking Scale:
1- none, 3- marginal, 9- highest

	Output Variables (Ys) / Description	1 broken	2 foreign material	3 smears	4 thickness	5 unbonded area	6	7	8	9	10	
	Weight	10	6	8	7	9						
	Input Variables (Xs)											Ranking
1	pressure	9			3	1						120
2	temperature	9			9	1						162
3	time	3			9	1						102
4	clean room practices		3	9								90
5	clean room cleanliness		9	3								78
6	washer		9	9								126
7	material properties		3	3	1	9						130
8	robot handling	9	1	3								120
9	human handling	9	3	3		3						159
10												

รูปที่ 2.2 แผนภาพ X-Y กำหนดความสำคัญของตัวแปร

Failure Modes and Effects Analysis (FMEA) คือเทคนิคที่นิยมใช้กันเป็นส่วนใหญ่ในการวิเคราะห์ความเสี่ยง การวิเคราะห์ความเสี่ยงเป็นกิจกรรมของมนุษย์โดยปกติทั่วไป FMEA เป็นการวิเคราะห์ความเสี่ยงที่มีประสิทธิภาพในการออกแบบและการผลิต ใช้ตรวจสอบการออกแบบและกระบวนการผลิตและระบุโอกาสสำหรับขอบเขตและของเสียที่สามารถเห็นผลลัพธ์ที่จะทำให้ลูกค้าไม่พึงพอใจ โดยสามารถวัดระดับความเสี่ยงได้จากตัวเลข RPN

สมการ RPN = S x O x D

เมื่อ S คือ ความรุนแรงผลกระทบที่เกิดความล้มเหลว (Severity)

O คือ โอกาสที่จะเกิดขึ้นจากสาเหตุบ่อยครั้ง (Occurrence)

D คือ ความสามารถตรวจจับและป้องกันไม่ให้เกิดความล้มเหลว (Detection)

2.1.2.3 Analyze

การวิเคราะห์สาเหตุของปัญหาหลัก ซึ่งเป็นการวิเคราะห์ในเชิงสถิติเพื่อระบุสาเหตุหลักที่ส่งผลโดยตรงต่อปัญหานั้น ซึ่งเรียกสาเหตุหลักนี้ว่า KPIV (Key Process Input Variable) ซึ่งต้องสามารถระบุให้ชัดเจนว่า อะไรคือ KPIV ของปัญหาและต้องสามารถเชื่อมโยงกับ ตัวหลักของกระบวนการ หรือที่เรียกว่า KPOV (Key Process Output Variable) ให้ได้ หลักการสถิติที่ใช้ในการวิเคราะห์ ได้แก่ การตรวจสอบสมมติฐาน (Hypothesis Testing) ผังการกระจาย (Scattering Diagram) การวิเคราะห์การถดถอย (Regression Analysis) เป็นต้น

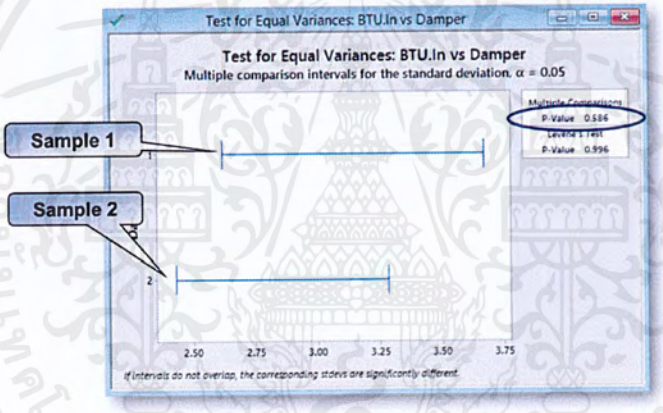
การทดสอบสมมติฐาน (Hypothesis Testing) เป็นส่วนหนึ่งของสถิติเชิงอนุมาน (Statistical Inference) ซึ่งเป็นการทดสอบเกี่ยวกับตัวแปรที่ไม่ทราบค่า โดยการทดสอบว่าตัวแปร(X)แต่ละตัวที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขึ้นมาในขั้นตอน Define มีผลกระทบต่อปัญหาจริงๆ โดยสุ่มตัวอย่างจากประชากรแล้ว อาศัยการแจกแจงของตัวสถิติ สร้างสถิติทดสอบเกี่ยวกับพารามิเตอร์ฐานๆ

สมมติฐานที่จะทดสอบเรียกว่า สมมติฐานหลัก (Null Hypothesis) โดยทั่วไปคือสมมติฐานที่กล่าวว่าตัวแปรที่เลือกไม่มีความเกี่ยวข้องกับปัญหาที่เกิดขึ้น เขียนแทนด้วย H0 ส่วนสมมติฐานที่แย้งกับสมมติฐานหลัก และนำมาพิจารณาในการทดสอบด้วยเรียกว่าสมมติฐานแย้งหรือสมมติฐานรอง (Alternative Hypothesis) ซึ่งกล่าวว่าแปรที่เลือกมีความเกี่ยวข้องกับปัญหาที่เกิดขึ้นหรือก็คือสาเหตุของปัญหา แทนด้วย H1

ทั้งนี้สามารถใช้ซอฟต์แวร์ Minitap เพื่อวิคราะห์สมมติฐานได้โดยสังเกตได้ที่ค่า P-Value หากมีค่ามากกว่า 0.05 เราจะยอมรับสมมติฐาน H0 ที่ว่าตัวแปรทีเลือกไม่เกี่ยวข้องกับปัญหาที่เกิดขึ้น แต่ถ้า P-Value มีค่าน้อยกว่า 0.05 เราจะยอมรับสมมติฐาน H1 ที่ว่าตัวแปรที่เลือกเกี่ยวข้องหรือเป็นสาเหตุของปัญหาที่เกิดขึ้น



รูปที่ 2.3 การทดสอบสมมติฐานด้วยซอฟต์แวร์ Minitap

2.1.2.4 Improve

การปรับปรุงโดยเน้นแก้ไขที่ตัวแปร(X) ซึ่งเป็นสาเหตุหลักที่ก่อให้เกิดปัญหาที่ได้มาจากขั้นตอน Analyse โดยต้องตรวจสอบและบริหารอย่างรอบคอบ และต้องวิเคราะห์ปัญหาอย่างระมัดระวังเพื่อป้องกันความผิดพลาดที่อาจเกิดขึ้น

2.1.2.5 Control

การควบคุมกระบวนการที่มีผลกระทบและพัฒนากระบวนการติดตามเพื่อรักษากระบวนการที่ได้เปลี่ยนแปลงไว้ให้คงอยู่ โดยทั่วไปจะใช้การเพิ่มขั้นตอนการทำงานลงไปในกระบวนการประกอบ รวมถึงการสร้างมาตรฐานต่างๆขึ้นมาเพื่อตรวจสอบผลการดำเนินงาน

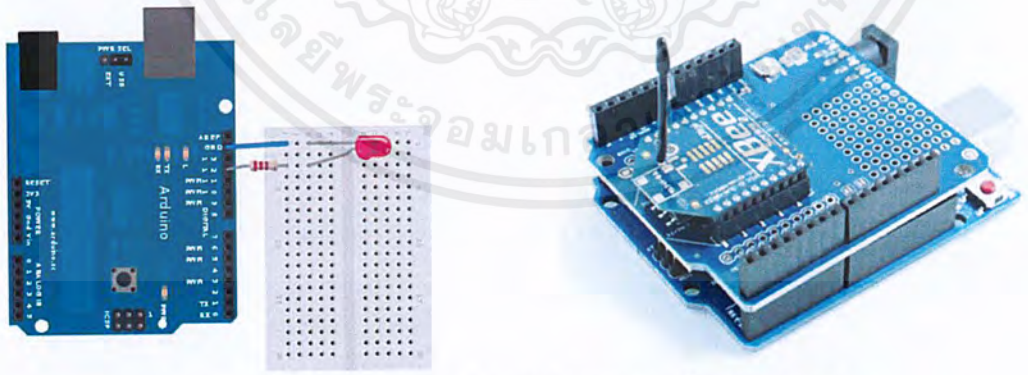
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 Arduino

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

Arduino คือ โครงการที่นำชิปไอซีไมโครคอนโทรลเลอร์ตระกูลต่างๆ มาใช้ร่วมกันในภาษา C ซึ่งภาษา C นี้เป็นลักษณะเฉพาะ คือมีการเขียนไลบารี่ของ Arduino ขึ้นมาเพื่อให้การสั่งงานไมโครคอนโทรลเลอร์ที่แตกต่างกัน สามารถใช้งานโค้ดตัวเดียวกันได้ โดยตัวโครงการได้สร้างบอร์ดทดลองมาหลายรูปแบบ เพื่อใช้งานกับ IDE ของตนเอง สาเหตุหลักที่ทำให้ Arduino เป็นนิยมมาก เป็นเพราะซอฟแวร์ที่ใช้งานร่วมกันสามารถโหลดได้ฟรี และตัวบอร์ดทดลองยังถูกแจกแปลน ทำให้ผู้ผลิตจีนนำไปผลิตและขายออกตลาดมาในราคาที่จับต้องได้

Auduino นั้นได้ใช้ชิป AVR เป็นหลักใน Auduino แทบรุ่น สาเหตุมาจากไมโครคอนโทรเลอร์ของตะกูล AVR นั้นมีความทันสมัย ในชิปในบางตัวสามารถเชื่อมต่อผ่าน USB ได้โดยตรง สามารถใช้กับคอมพิวเตอร์สมัยใหม่ได้เป็นอย่างดี และในไมโครคอนโทรเลอร์ตะกูล AVR ยังมีส่วนของโปรแกรมพิเศษที่เรียกว่า Bootloader อยู่ในระดับล่างกว่าส่วนโปรแกรมปกติ ซึ่งจะเป็นส่วนโปรแกรมที่จะถูกเรียกขึ้นมาก่อนการเรียกโปรแกรมปกติ ทำให้สามารถเขียนสั่งให้ทำงานใดๆก็ได้ ก่อนการเรียกโปรแกรมปกติ ทำให้ Arduino นั้นอาศัยส่วนโปรแกรมแกรมพิเศษนี้ในการทำให้ชิปสามารถโปรแกรมผ่านพอร์ตอนุกรมชนิด UART ได้ จึงทำให้การเขียนโปรแกรมลงไปในชิปใช้เพียง USB to UART ก็เพียงพอแล้ว แต่การโปรแกรมด้วยการใช้โปรโตคอล UART ก็มีข้อเสียตรงที่ต้องใช้เวลาในการบูตเข้าโปรแกรมปกติประมาณ 1 – 2 วินาที



รูปที่ 2.4 บอร์ด Arduino ต่อกับ LED และบอร์ด XBee Shield

บอร์ด Arduino นั้นมีด้วยกันหลักๆ 9 บอร์ด คือ 1. Arduino Uno 2. Arduino Duo 3. Arduino Leonardo 4. Arduino MEGA ADK 5. Arduino Mega 2560 6. Arduino Micro 7. Arduino Nano 8. Arduino Mini 9. Arduino Pro Mini ซึ่งบอร์ดแต่ละชนิดนั้นจะมีขนาดและข้อจำกัดการใช้งานที่เหมาะสมแตกต่างกัน ทั้งนี้ผู้ศึกษาได้เลือกใช้บอร์ด Arduino Uno R3 เนื่องจากเป็นบอร์ดที่ใช้งานได้หลากหลาย

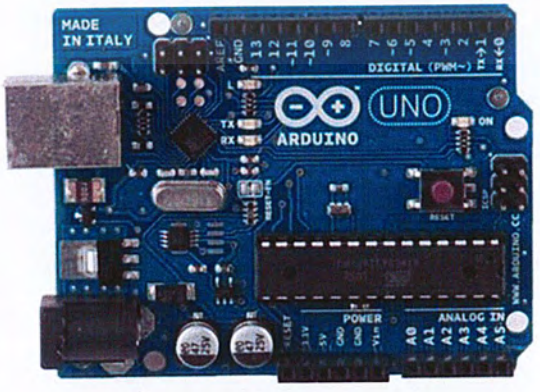
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เหมาะสมกับการนำมาควบคุมเซ็นเซอร์ และมอเตอร์ไฟฟ้า พร้อมทั้งมีราคาที่เหมาะสมสำหรับการเริ่มศึกษา

2.2.1 ข้อมูลทั่วไปของ Arduino Uno R3

คำว่า Uno เป็นภาษาอิตาลี ซึ่งแปลว่าหนึ่ง เป็นบอร์ด Arduino รุ่นแรกที่ออกมา มีขนาดประมาณ 68.6x53.4mm เป็นบอร์ดมาตรฐานที่นิยมใช้งานมากที่สุด เนื่องจากเป็นขนาดที่เหมาะสำหรับการเริ่มต้นเรียนรู้ Arduino และมี Shields ให้เลือกใช้งานได้มากกว่าบอร์ด Arduino รุ่นอื่นๆที่ออกแบบมาเฉพาะมากกว่า โดยบอร์ด Arduino Uno ได้มีการพัฒนาเรื่อยมา ตั้งแต่ R2 R3 และรุ่นย่อยที่เปลี่ยนชิปไอซีเป็นแบบ SMD โดยมีข้อมูลจำเพาะดังนี้

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 Booloader
พื้นที่แรม 2KB
พื้นที่หน่วยความจำถาวร (EEPROM) 1KB
ความถี่คริสตัล 16MHz
ขนาด 68.6x53.4 mm
น้ำหนัก 25 กรัม

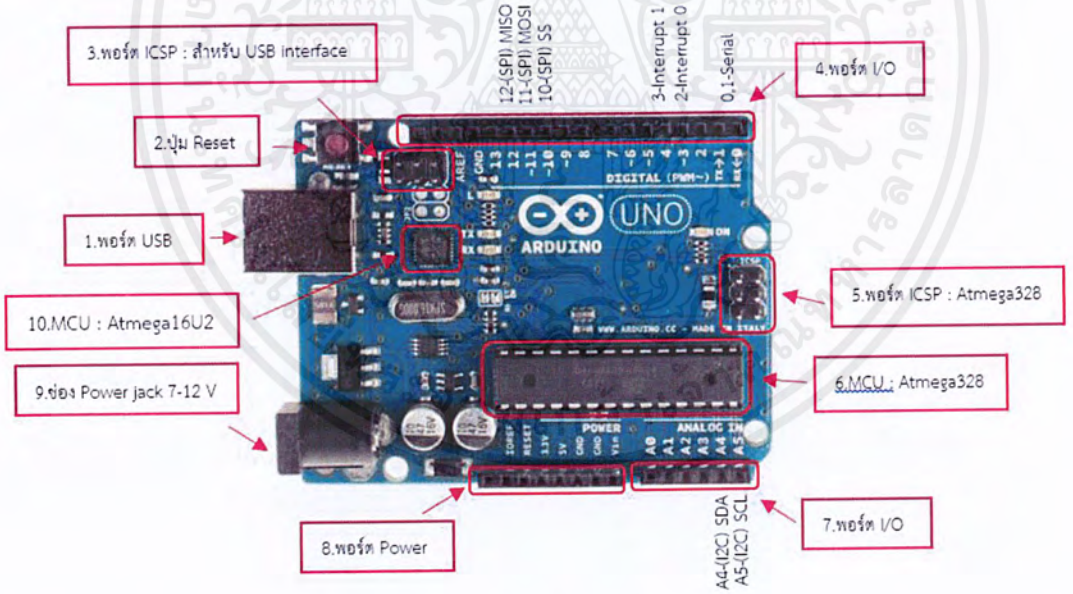


รูปที่ 2.5 บอร์ด Arduino Uno R3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ส่วนประกอบของบอร์ด Arduino Uno R3

1. USB Port: ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
2. Reset Button: เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่
3. ICSP Port ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2
4. I/OPort:Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin0,1 เป็นขา Tx,Rx Serial, Pin3,5,6,9,10 และ 11 เป็นขา PWM
5. ICSP Port: Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
6. MCU: Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino
7. I/OPort: นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็น ช่องรับสัญญาณอนาล็อก ตั้งแต่ขา A0-A5
8. Power Port: ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง +3.3 V, +5V, GND, Vin
9. Power Jack: รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V
10. MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่าน Atmega16U2



รูปที่ 2.6 ชิ้นส่วนของบอร์ด Arduino Uno R3

2.2.3 การเขียนโปรแกรม

ในการเขียนโปรแกรมนั้นจะใช้ภาษาซีเขียน โดยภาษาซีของ Arduino จะจัดรูปแบบโครงสร้างของการเขียนโปรแกรมออกเป็นส่วนย่อยๆหลายๆส่วน โดยเรียกแต่ละส่วนว่า ฟังก์ชั่น และ เมื่อนำฟังก์ชั่น มารวมเข้าด้วยกัน ก็จะเรียกว่าโปรแกรม โดยโครงสร้างการเขียนโปรแกรมของ Arduino นั้น ทุกๆโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะต้องประกอบไปด้วยฟังก์ชั่นจำนวนเท่าใดก็ได้ แต่อย่างน้อยที่สุดต้องมีฟังก์ชั่น จำนวน 2 ฟังก์ชั่น คือ setup() และ loop() ดังนี้

```
#include<servo.h>              //เรียกไลบรารี่ ชื่อ servo.h เข้ามาใช้ในโปรแกรม
int Servo1=9;                  //กำหนดให้ Servo1 แทน Pin Digital-9
Servo myservo;                 //สร้าง object ชื่อ myservo เพื่อควบคุม Servo
void setup()
{
myservo.attach(Serrvo1);       //กำหนดให้ใช้ขา Digital-9 สร้างสัญญาณควบคุม Servo
}
void loop()
{
myservo.write(180);            //กำหนดค่าตำแหน่งให้กับ Servo = 180 องศา
}
```

จะได้เห็นได้โครงสร้างพื้นฐานของภาษาซีที่ใช้กับ Arduino นั้นจะประกอบไปด้วย 3 ส่วนใหญ่ๆ ด้วยกัน คือ 1. Header ในส่วนนี้จะมีหรือไม่มีก็ได้ ถ้ามีต้องกำหนดไว้ในส่วนเริ่มต้นของโปรแกรม ซึ่งส่วนของ Header ได้แก่ ส่วนที่เป็น Compiler Directive ต่างๆรวมไปถึงส่วนของการประกาศตัวแปร และค่าคงที่ต่างๆที่จะใช้ในโปรแกรม 2.setup() ในส่วนนี้เป็นฟังก์ชั่นบังคับที่ต้องกำหนดให้มีในทุกๆโปรแกรม ถึงแม้ว่าในบางโปรแกรมจะไม่ต้องการใช้งานก็ยังจำเป็นต้องประกาศไว้ด้วยเสมอ เพียงแต่ไม่ต้องเขียนคำสั่งใดๆไว้ในระหว่างวงเล็บปีกกา {} ที่ใช้เป็นตัวกำหนดของเขตของฟังก์ชั่น โดยฟังก์ชั่นนี้จะใช้สำหรับบรรจุคำสั่งในส่วนที่ต้องกาให้โปรแกรมทำงานเพียงรอบเดียวตอนเริ่มต้นทำงานของโปรแกรมครั้งแรกเท่านั้น ซึ่งได้แก่คำสั่งเกี่ยวกับการ Setup ค่าการทำงานต่างๆ เช่น การกำหนดหน้าที่การใช้งานของ PinMode และการกำหนดค่า Baudrate สำหรับใช้งานพอร์ตสื่อสารอนุกรม เป็นต้น
3.loop() เป็นส่วนฟังก์ชั่นบังคับที่ต้องกำหนดให้มีในทุกๆโปรแกรมเช่นเดียวกันกับฟังก์ชั่น setup() โดยฟังก์ชั่น loop() นี้จะใช้บรรจุคำสั่งที่ต้องการให้โปรแกรมทำงานเป็นวงรอลซ้ำๆกันไปไม่รู้จบ ซึ่งถ้าเปรียบเทียบกับรูปแบบของ ANSI-C ส่วนนี้ก็คือ ฟังก์ชั่น main() นั่นเอง

เมื่อพบคำสั่ง #include ตัวแปลภาษาของ Arduino จะไปค้นหาไฟล์ที่ระบุไว้ในเครื่องหมาย <> หลังคำสั่ง #include จากตำแหน่ง Directory ที่เก็บไฟล์ Library ของโปรแกรม Arduino ไว้ ซึ่งแน่นอนว่าส่วนของ Header จะนับรวมไปถึง คำสั่งส่วนที่ใช้ประกาศสร้าง ตัวแปร(Variable Declaration)และค่าคงที่(Constant Declaration) รวมทั้ง ฟังก์ชั่นต่างๆ (Function Declaration) ด้วย

สำหรับส่วนที่สำคัญที่สุดและขาดไม่ได้ คือ ฟังก์ชั่น setup() และ ฟังก์ชั่น loop() ซึ่งฟังก์ชั่น ทั้ง 2 ส่วนนี้มีรูปแบบโครงสร้างที่เหมือนกัน แต่ถูกกำหนดด้วยชื่อฟังก์ชั่นเป็นการเฉพาะ คือ setup() และ loop()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย setup() จะเขียนไว้ก่อน loop() ซึ่งทัง 2 ฟังก์ชั่นนี้ มีขอบเขต เริ่มต้นและสิ้นสุด อยู่ภายใต้เครื่องหมาย {}

```
void setup()
{
คำสั่งต่างๆ ที่ต้องการเขียนไว้ภายใต้ฟังก์ชั่น setup()
}
```

หน้าที่ของฟังก์ชั่น setup() ใน Arduino คือ ใช้ทำหน้าที่เป็นส่วนของโปรแกรมย่อยสำหรับใช้บรรจุคำสั่งต่างๆที่ใช้สำหรับกำหนดกาทำงานของระบบ หรือ กำหนดคุณสมบัติการทำงานให้กับอุปกรณ์ต่างๆซึ่งคำสั่งทั้งหมดที่บรรจุไว้ภายใต้ฟังก์ชั่นของ Setup() นี้ จะถูกเรียกขึ้นมาทำงานเพียงรอบเดียวคือตอนเริ่มต้นการทำงานของโปรแกรม โดยคำสั่งที่นิยมบรรจุไว้ในฟังก์ชั่นส่วนนี้ ได้แก้ คำสั่งสำหรับกำหนดโหมดการทำงานของ Digital Pin หรือ คำสั่งสำหรับ กำหนดคุณสมบัติของพอร์ตสื่อสารอนุกรม เป็นต้น

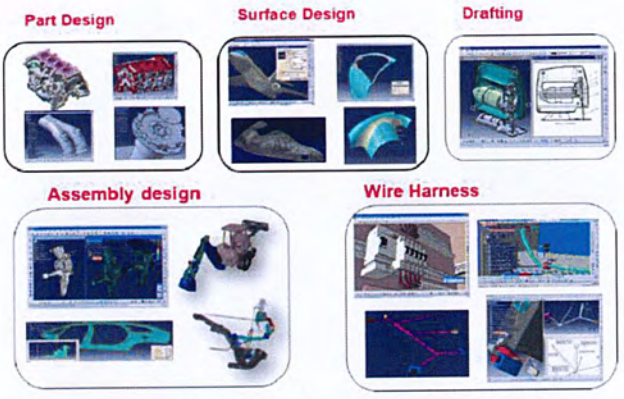
```
void loop()
{
คำสั่งต่างๆที่ต้องการให้ทำงานภายใต้ฟังก์ชั่น loop()
}
```

หน้าที่ของฟังก์ชั่น loop() ใน Arduino คือใช้ทำหน้าที่เป็นส่วนของโปรแกรมหลัก สำหรับใช้บรรจุคำสั่งควบคุมการทำงานต่างๆของโปรแกรม ที่ต้องการใช้โปรแกรมทำงาน โดยคำสั่งที่บรรจุไว้ในฟังก์ชั่นนี้จะถูกเรียกขึ้นมาทำงานซ้ำๆกันตามลำดับและเงื่อนไขที่กำหนดไว้

2.3 Catia

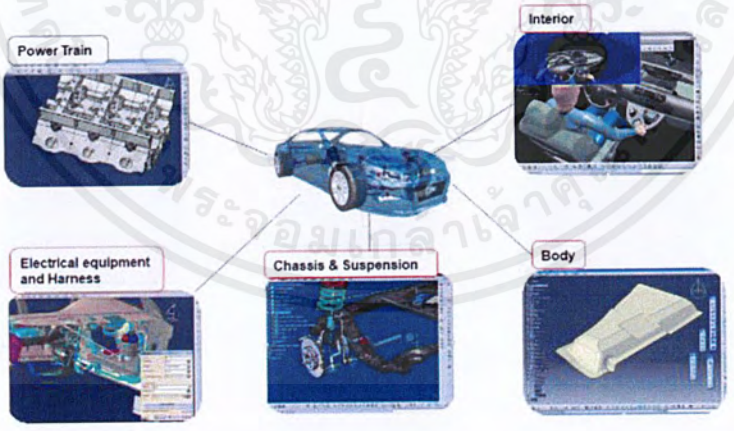
Catia คือ มาตราฐานซอฟท์แวร์ออกแบบงานสามมิติระดับสูง ที่ใช้ในอุตสาหกรรมยานยนต์ อิเล็กทรอนิคส์ เครื่องจักร และการบิน ฯลฯ และถือเป็นหนึ่งในโปรแกรมมาตราฐานในการออกแบบผลิตภัณฑ์ และอุสาหกรรมระดับโลก ด้วยความสามารถที่เป็นที่ยอมรับในการออกแบบ, ลดเวลาการทำงาน , ลดความผิดพลาดต่างๆ ตั้งแต่เริ่มต้นการออกแบบ, ออกแบบรูปทรงได้หลากหลาย อีกทั้งยังช่วยลดค่าใช้จ่ายในการแก้ปัญหาเรื่องของการออกแบบที่ผิดพลาด เนื่องจากการทำงานในรูปแบบที่หลากหลายจากผลิตภัณฑ์ หรือขั้นตอนการทำงานที่ต่างกันภายในและนอกองค์กร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ฟังค์ชั่นการใช้งาน Catia

CATIA V5 ประกอบด้วยฟังก์ชั่นการทำงานหลักๆ เบื้องต้นอยู่ 4 ส่วน คือ งานออกแบบชิ้นส่วนเป็น Solid (Part Design) การออบแบบผิวชิ้นงาน (Generative Shape Design) การประกอบชิ้นส่วน (Assembly) การเขียนแบบ (Drafting) โปรแกรม CATIA V5 จะสามารถช่วยตอบโจทย์ในการพัฒนาผลิตภัณฑ์ในทุกๆส่วนของขั้นตอนการผลิต ไม่ว่าจะเป็นผู้ผลิตรถยนต์ และอากาศยานชั้นนำ (OEM) ระดับโลกจนถึง Suppliers ที่มีขนาดเล็กๆ โดยสามารถออกแบบและเขียนชิ้นงานได้ทุกส่วนของผลิตภัณฑ์ และโครงสร้างประกอบอื่นๆ อาทิเช่น เครื่องยนต์ วงจรและสายไฟฟ้า อีกทั้งยังสามารถเพิ่มขยายขีดความสามารถไปในเรื่องของ การวิเคราะห์ การสร้างชิ้นงานจาก Scan Data (Reversed Engineering) และงานโปรแกรมของเครื่องจักร



รูปที่ 2.8 การใช้งาน Catia ในอุตสาหกรรมรถยนต์

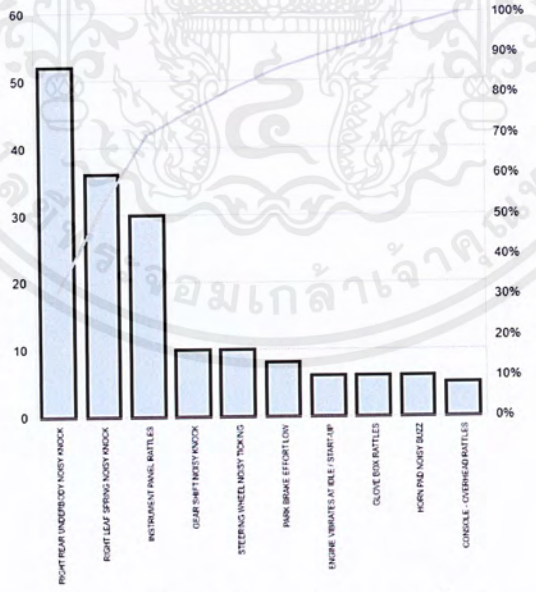
วิศวกรเครื่องกล สามารถดึงความสามารถทางวิศ กรรมมาใช้ได้อย่างมีปะสิทธิภาพ โดยไม่มีข้อจำกัดทางซอฟท์แวร์ ซึ่งสามารถทำงานโดยใช้ CATIA V5 ตั้งแต่กระบวนการออกแบบชิ้นงานในขั้นตอนแรก ไปจนถึงการขึ้นรูป 3 มิติ,การประกอบ,การเขียน 2D, การวิเคราะห์ความแข็งแรง และ การสร้าง ผลิตภัณฑ์ต้นแบบ (Prototype)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีดำเนินการวิจัย

3.1 ศึกษาปัญหาที่เกิดขึ้นภายในโรงประกอบ

ปัญหาภายในโรงประกอบนั้นมีหลากหลาย แต่ส่วนมากนั้นได้มีวิธีการแก้ไขแล้ว ผู้ศึกษาจึงสนใจปัญหาเสียงกระเทาะภายในตัวรถ เนื่องจากเป็นปัญหาที่มีมานานประกอบกับไม่มีวิธีแก้ไขที่เหมาะสม ทั้งยังสร้างความเสียหายให้แก่บริษัทเป็นอย่างมาก โดยเมื่อรถเกิดปัญหาขึ้น โรงงานจำเป็นต้องนำรถคันนั้นมาตรวจสอบเพื่อหาสาเหตุทำให้เสียยอดการผลิต นอกจากนี้ยังเป็นการเพิ่มภาระงานให้แก่แผนกซ่อมบำรุง โดยหากปัญหานั้นเกี่ยวข้องกับชิ้นส่วนเกินติดค้างในแผงหน้าปัดรถ ก็จะมีค่าใช้จ่ายประมาณ 6000 บาทต่อคัน ผู้ศึกษาจึงเก็บรวบรวมข้อมูลปัญหาเสียงกระเทาะภายในรถยนต์ทั้งหมดมาจากแผนกทดสอบเสียง พบว่าจากเดือนมิถุนายน ถึงเดือนกันยายน ปี 2019 ปัญหาที่พบบ่อยที่สุดเป็นอันดับแรก คือ ปัญหาเสียงกะเทาะที่ตัวถังคิดเป็น 18.51 เปอร์เซนต์ของปัญหาเสียงกระเทาะทั้งหมด มีสาเหตุมาจาก การเชื่อมถังรถไม่สนิท อันดับต่อมาคือ ปัญหาเสียงกระเทาะที่แหนบรองน้ำหนักและกันสะเทือนของรถคิดเป็น 14.81 เปอร์เซนต์ของปัญหาเสียงกระเทาะทั้งหมด มีสาเหตุมาจากการที่ไม่สามารถควบคุมปริมาณจารบีที่ใช้ทาได้ ซึ่งทั้ง 2 ปัญหานั้นได้มีแนวทางการแก้ไขที่เหมาะสมแล้ว ผู้ศึกษาจึงเลือกปัญหาที่เกิดขึ้นบ่อยเป็นอันดับสาม คือ ปัญหาเสียงกะเทาะภายในแผงหน้าปัดรถคิดเป็น 13.87 เปอร์เซนต์ของปัญหาเสียงกระเทาะทั้งหมด

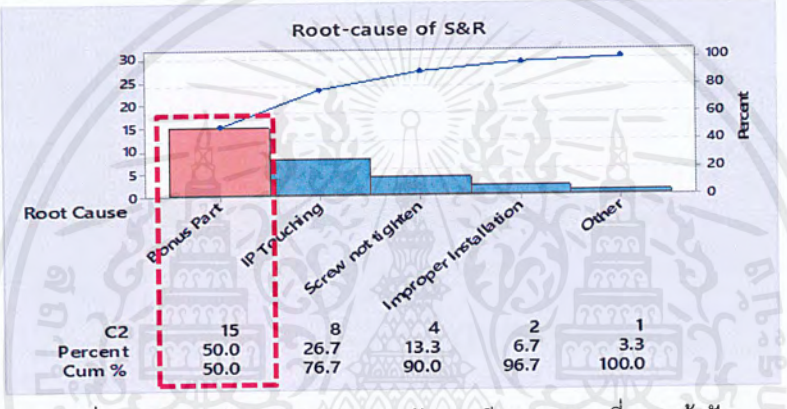


รูปที่ 3.1 พาริโตแสดงปัญหาเสียงกระเทาะทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 หาสาเหตุของปัญหาเสียงกระเทาะในแผงหน้าปัด

จากการศึกษาและเก็บข้อมูลนั้น พบว่าปัญหาเสียงกระเทาะที่แผงหน้าปัดเกิดขึ้นได้หลายสาเหตุ ดังนี้ การมีชิ้นส่วนเกินไปติดอยู่ในแผงหน้าปัด การกระทบกันของชิ้นส่วนภายในแผงหน้าปัดเนื่องมาจากขนาดไม่ตรงมาตรฐาน การขันสกรูตามจุดยึดต่างๆไม่แน่นหนา และการติดตั้งชิ้นส่วนภายในแผงหน้ารถไม่เป็นไปตามมาตรฐาน จากข้อมูลพบว่าสาเหตุหลักกว่า 50 เปอร์เซนต์นั้นเกิดขึ้นจาก การมีชิ้นส่วนเกินไปติดอยู่ในแผงหน้าปัด(Bonus Part) และชิ้นส่วนเหล่านั้นส่วนมากคือสกรูขนาดเล็ก ผู้ศึกษาจึงเลือกเจาะจงไปที่การป้องกันละแก้ไขปัญหานี้ ซึ่งจากข้อมูลทางสถิติพบว่าจะช่วยลดปัญหาเสียงกระเทาะที่หน้าปัดได้ถึง 50 เปอร์เซนต์



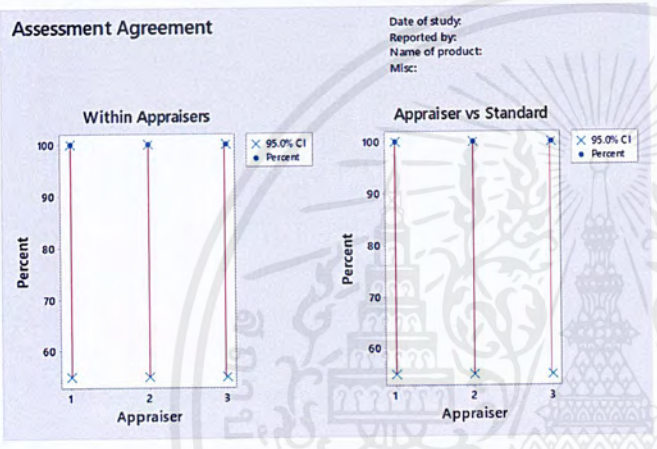
รูปที่ 3.2 ผังพาเรโตแสดงสาเหตุของปัญหาเสียงกระเทาะที่แผงหน้าปัด

3.3 ทดสอบความสามารถการวัดของผู้ตรวจคุณภาพรถ

ในการตรวจปัญหาเสียงกระเทาะในรถยนต์นั้น ทำได้โดยการนำรถมาวิ่งในลู่พิเศษที่มีพื้นผิวขรุขระแตกต่างกัน โดยแต่ละพื้นผิวนั้นจะสร้างแรงสั่นสะเทือนในหลายช่วงความถี่ เพื่อใช้ตรวจหาปัญหาเสียงกระเทาะที่เกิดขึ้นได้ในความถี่นั้นๆ โดยผู้ตรวจคุณภาพจะนั่งไปในรถเพื่อคอยฟังเสียงผิดปกติ ทั้งนี้เพื่อยืนยันความน่าเชื่อถือของข้อมูลรายงานปัญหาข้างต้น ผู้ศึกษาจึงได้ทดสอบความสามารถในการตรวจ และฟังเสียงของผู้ตรวจคุณภาพ โดยให้ผู้ตรวจคุณภาพตรวจรถที่มีความผิดปกติของเสียงที่บริเวณต่างกันจำนวน 3 คัน และรถที่ปกติจำนวน 2 คัน เพื่อทดสอบว่าผู้ตรวจคุณภาพสามารถแยกได้ระหว่างรถปกติและรถผิดปกติ ทั้งยังต้องสามารถระบุได้ว่าเป็นเสียงผิดปกติจากบริเวณใดของรถ จากผลการสุ่มทดสอบผู้ตรวจคุณภาพ 3 คน พบว่า ทั้ง 3 คน มีความสามารถในการตรวจจับปัญหาเสียงผิดปกติในรถยนต์ได้แม่นยำ 100 เปอร์เซนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.3 สนามทดสอบเสียงกระเทาะ



Within Appraisers

Assessment Agreement

Appraiser	# Inspected	# Matched	Percent	95% CI
1	5	5	100.00	(54.93, 100.00)
2	5	5	100.00	(54.93, 100.00)
3	5	5	100.00	(54.93, 100.00)

 # Matched: Appraiser agrees with him/herself across trials.

Each Appraiser vs Standard

Assessment Agreement

Appraiser	# Inspected	# Matched	Percent	95% CI
1	5	5	100.00	(54.93, 100.00)
2	5	5	100.00	(54.93, 100.00)
3	5	5	100.00	(54.93, 100.00)

 # Matched: Appraiser's assessment across trials agrees with the known standard.

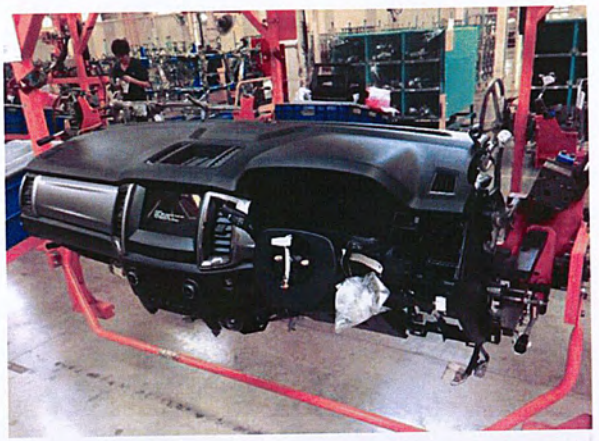
รูปที่ 3.4 การทำ Measurement System Analysis(MSA)

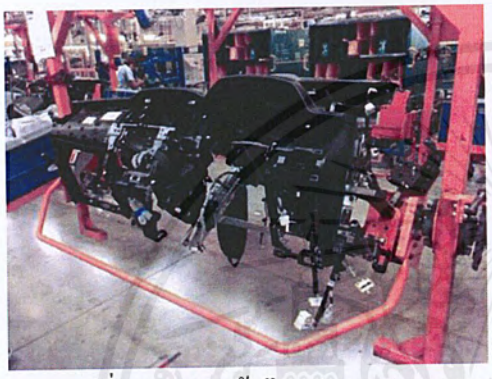
3.2.4 ศึกษาขั้นตอนการประกอบแผงหน้ารถ

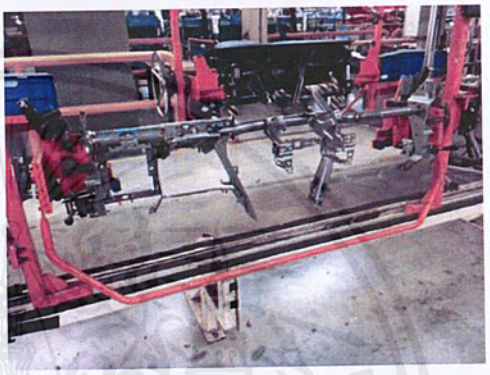
แผงหน้าปัดรถ (Instrument Panel) คือ ชุดอุปกรณ์ที่ใช้ในการระบบต่างๆของรถยนต์ โดยติดตั้งบริเวณด้านหน้าคนขับ ภายในจะประกอบด้วยแผงวงจรไฟฟ้าและอิเล็กทรอนิกส์หลายชนิด โดยมีชิ้นส่วนสำคัญขนาดใหญ่อยู่ 2 ส่วนคือ Cross Car Beam และ IP Skin ดังแสดงในรูปที่ 3.5

การประกอบแผงหน้าปัดรถนั้นมีหลายขั้นตอน โดยจากการเก็บข้อมูลพบว่าจุดประกอบที่มีความเสี่ยงสูงที่สกรูจะตกลงไปในแผงหน้าปัดมากที่สุดคือ สถานีที่ 8(TIP08 XS) ซึ่งเป็นการประกอบ Module และ Adaptor A/C เข้ากับ IP skin ซึ่งที่จุดประกอบนี้มีการใช้สกรูขนาดเล็กจำนวน 7 ตัวเพื่อประกอบรถทุกรุ่น ทั้งยังเป็นจุดประกอบที่แผงหน้าปัดถูกคว่ำลงทำให้มีรูและช่องต่างๆจำนวนมาก เพิ่มความเสี่ยงที่สกรูจะเข้าไปติดได้ โดยมีขั้นตอนการประกอบดังรูปที่ 3.6 และ 3.7

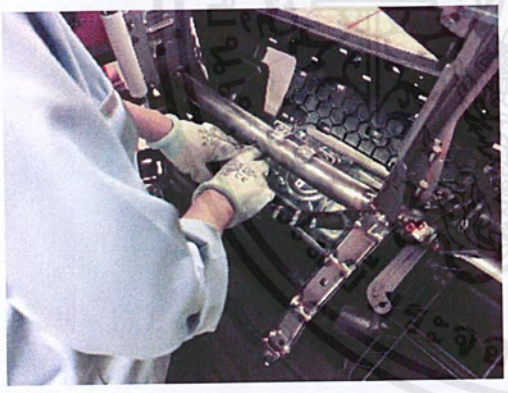
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้







รูปที่ 3.5 แผงหน้าปัดรถ(Instrument Panel) IP skin(ซ้าย) และCross Car Beam(ขวา)



1.Wiring

2.Install module

รูปที่ 3.6 แสดงการประกอบแผงหน้าปัด 1-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

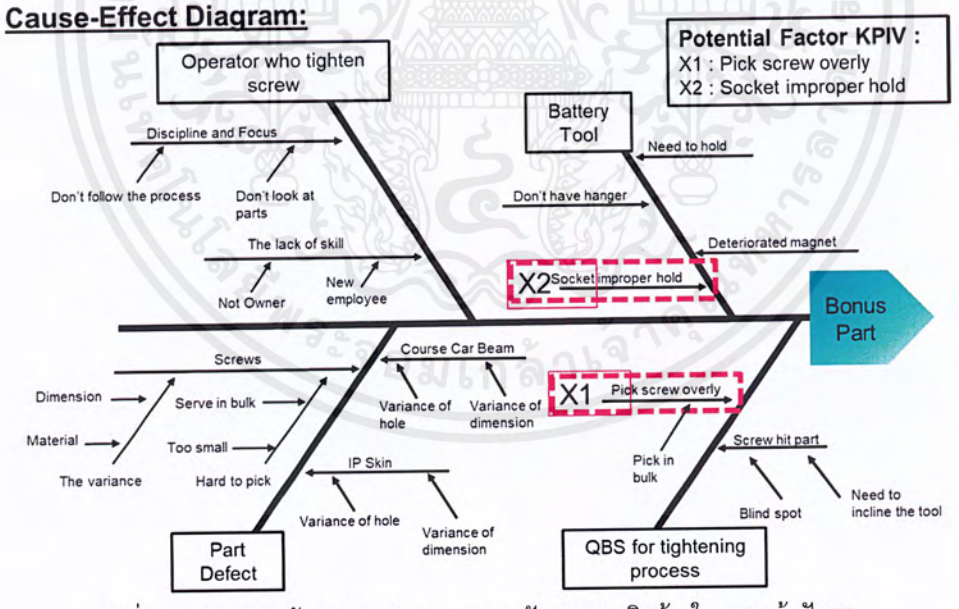
3.Pick screws

4.Tighten screw at IP skin

รูปที่ 3.7 แสดงการประกอบแผงหน้าปัด 3-4

3.4 หาสาเหตุของการมีสกรูติดอยู่ในแผงหน้าปัด(Bonus Part)

จากการเก็บข้อมูลโดยสอบถามผู้ประกอบงานและปรึกษาวิศวกรผู้คุมงาน คาดการณ์ว่า สาเหตุที่สกรูติดค้างในแผงหน้าปัด ส่วนใหญ่มาจาก 2 สาเหตุ คือ 1. การหยิบสรูเกินกว่าที่จำเป็น 2. เบ้ายึดที่ปืนขันแน่นยึดสกรูไม่แน่น จากนั้นผู้ศึกษาได้ทำการทดสอบสมมติฐานทางสถิติ เพื่อยืนยันสมมติฐานข้างต้น



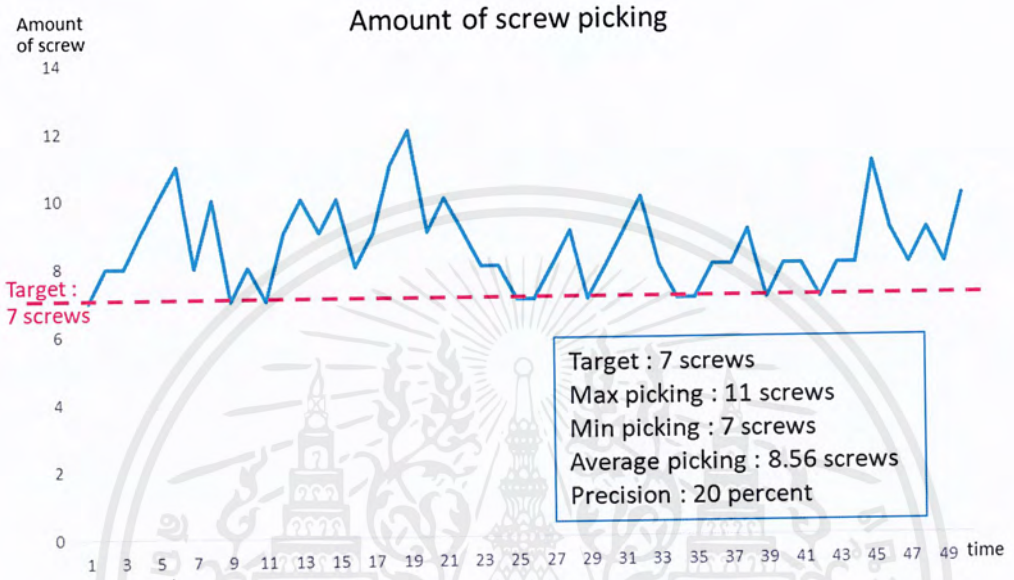
รูปที่ 3.8 แผนภาพก้างปลาแสดงสาเหตุของปัญหาสกรูติดค้างในแผงหน้าปัดรถ

3.4.1 สมมติฐานที่ 1 การหยิบสรูเกินกว่าที่จำเป็น

ผู้ศึกษาได้ทำการทดลองโดยการให้ผู้ประกอบงานหยิบสกรูตามปกติและเก็บบันทึกผล 50 ครั้ง พบว่า ผู้ประกอบงานมีโอกาสที่จะหยิบสกรูมากเกินกว่าที่จำเป็นต้องใช้ถึง 80 เปอร์เซ็น ซึ่งต้องมีการหยิบสกรู 7 ตัว มีการหยิบเฉลี่ยที่ 8.56 ตัว มากที่สุดคือ 11 ตัว และน้อยที่สุดคือ 7 ตัว โดยในการประกอบ ผู้

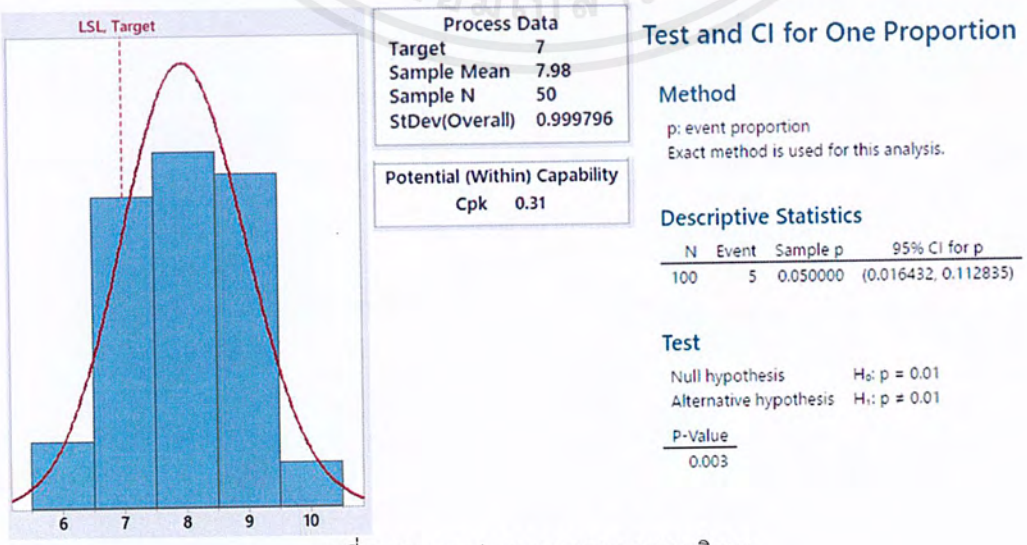
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประกอบจำเป็นต้องใช้มือหนึ่งข้างเพื่อถือสกรู ในขณะที่มืออีกข้างใช้จับอุปกรณ์ขันแน่น ซึ่งปริมาณสกรูที่มากเกินมานี้สร้างความลำบากในการถือด้วยมือข้างเดียวทำให้มีโอกาสที่สกรูจะตกหล่นได้ ประกอบกับเสียงเครื่องจักรในสายการประกอบที่ดังรบกวน และการประกอบงานอย่างเร่งรีบเพื่อให้ทันต่อปริมาณการผลิต ทำให้ผู้ประกอบงานไม่มีเวลาที่จะสังเกตุเห็นในกรณีที่สกรูตกลงไปในแผงหน้าปัด



รูปที่ 3.9 กราฟแสดงปริมานการหยิบสกรู

นอกจากนี้ผู้ศึกษายังทำการทดสอบสมมติฐานเพิ่มเติม โดยให้ผู้ประกอบหยิบสกรู 100 ครั้งให้ครบ 7 ตัวพอดีพบว่าอัตราการทำสกรูตกอยู่ที่ 1 เปอร์เซ็นต์ ในขณะที่การหยิบสกรูตามปกตินั้นมีอัตราการทำสกรูตกมากถึง 5 เปอร์เซนต์ โดยวัดค่า P-Value ได้ 0.003 ซึ่งชี้ชัดว่าการหยิบสกรูเกินนั้นส่งผลให้ทำสกรูตกอย่างมีนัยยะสำคัญ

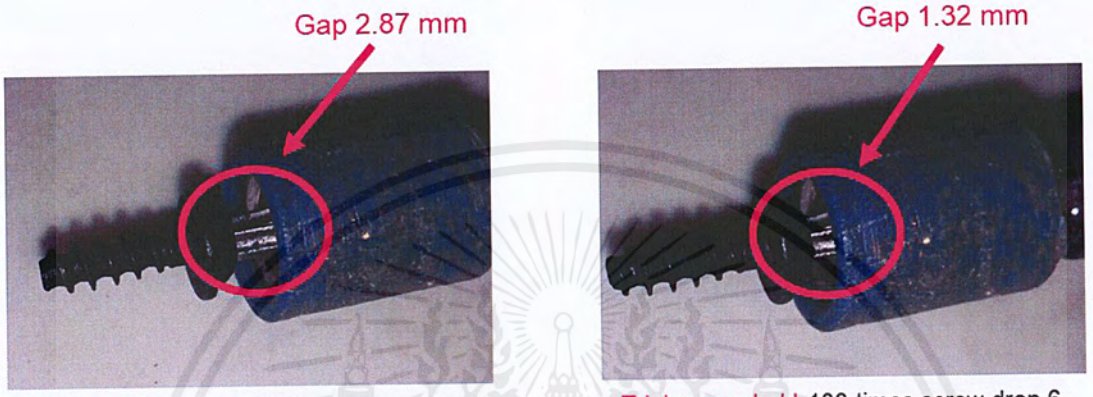


รูปที่ 3.10 กราฟแสดงการทดสอบสมมติฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2 สมมติฐานที่ 2 เบ้ายึดที่ปืนขันแน่นยึดสกรูไม่แน่น

ผู้ศึกษาได้ทำการทดลองโดยขั้นแน่น 100 ครั้ง แบ่งเป็นการขั้นแน่นแบบเบ้ายึดสกรูไม่แน่นโดยมีช่องว่าง 2.87 mm พบว่าสกรูตก 7 เปอร์เซนต์ และการขันแน่นแบบเบ้ายึดสกรูแน่น โดยมีช่องว่าง 1.32 mm พบว่าสกรูตก 6 เปอร์เซ็น และจากการทดสอบสมมติฐานได้ผล P-Value มีค่า 0.444 ซึ่งสามารถสรุปได้ว่าการยึดของเบ้ายึดที่ปืนขันแน่นไม่มีผลต่อการตกหล่นของสกรู ผู้ศึกษาจึงตัดสมมติฐานนี้ออก



รูปที่ 3.11 การทดสอบการยึดแน่นของเบ้าปืนขันแน่น

Test and CI for One Proportion

Method

p: event proportion
Exact method is used for this analysis.

Descriptive Statistics

N	Event	Sample p	95% Upper Bound for p
100	6	0.060000	0.114985

Test

Null hypothesis	H_0: p = 0.07
Alternative hypothesis	H_1: p < 0.07

P-Value
0.444

รูปที่ 3.12 การทดสอบสมมติฐานเบ้ายึดสกรู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดสอบสมมติฐานทั้งสอง ผู้ศึกษาสามารถสรุปได้ว่า การหยิบสกรูเกินเป็นสาเหตุหลักของการทำสกรูติดอยู่ในแผงหน้าปัด

3.5 การหาแนวทางแก้ปัญหาการหยิบสกรูเกิน

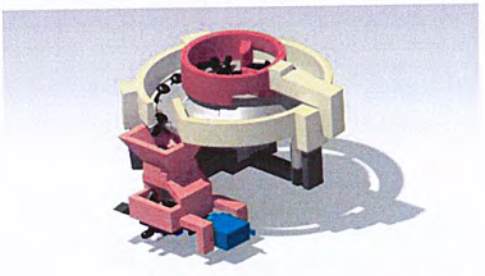
ในการแก้ไขการหยิบสกรูเกินนั้น ผู้ศึกษาได้คิดใช้เครื่องนับและจ่ายสกรูมาติดตั้งที่ข้างสายการประกอบ เพื่อควบคุมปริมาณสกรูให้เป็นไปตามมาตรฐานการประกอบ

รูปที่ 3.13 เครื่องจ่ายสกรูทั่วไป

โดยเครื่องนับสกรูทั่วไปมีข้อเสียหลักอยู่ 3 ประการ คือ 1. มีราคาสูง โดยพบว่า มีราคาตั้งแต่ 50000 บาท ขึ้นไปจนถึง 1000000 บาท ซึ่งไม่คุ้มค่าต่อการลงทุน 2. มีเสียงที่ดังรับกวนการทำงาน เนื่องจากตัวเครื่องมีขนาดใหญ่และใช้การสั่นสกรูจำนวนมาก เพื่อจ่ายสกรูผ่านไปยังทางลำเลียง ทำให้มีเสียงดังมากกว่า 80 เดซิเบลซึ่งถือว่าเกินมาตรฐานของฟอร์ด 3. มีขนาดที่ใหญ่ ทำให้ไม่เหมาะสมต่อใช้นำไปใช้งานในสายการประกอบ ผู้ศึกษาจึงคิดประดิษฐ์เครื่องนับและจ่ายสกรูที่ไม่มีข้อเสียเหล่านี้

3.6 การออกแบบเครื่องจ่ายสกรู

ผู้ศึกษาได้นำแนวคิดจากเครื่องนับสกรูทั่วไปมาประยุกต์ใช้ในการออกแบบ โดยออกแบบชิ้นส่วนแต่ละชิ้นโดยใช้ Catia หลังจากนั้นจึงใช้ 3D printer ในการขึ้นรูปชิ้นส่วนแต่ละชิ้น จากนั้นนำมาประกอบเข้าด้วยกันกับส่วนของแผงวงจร Arduino



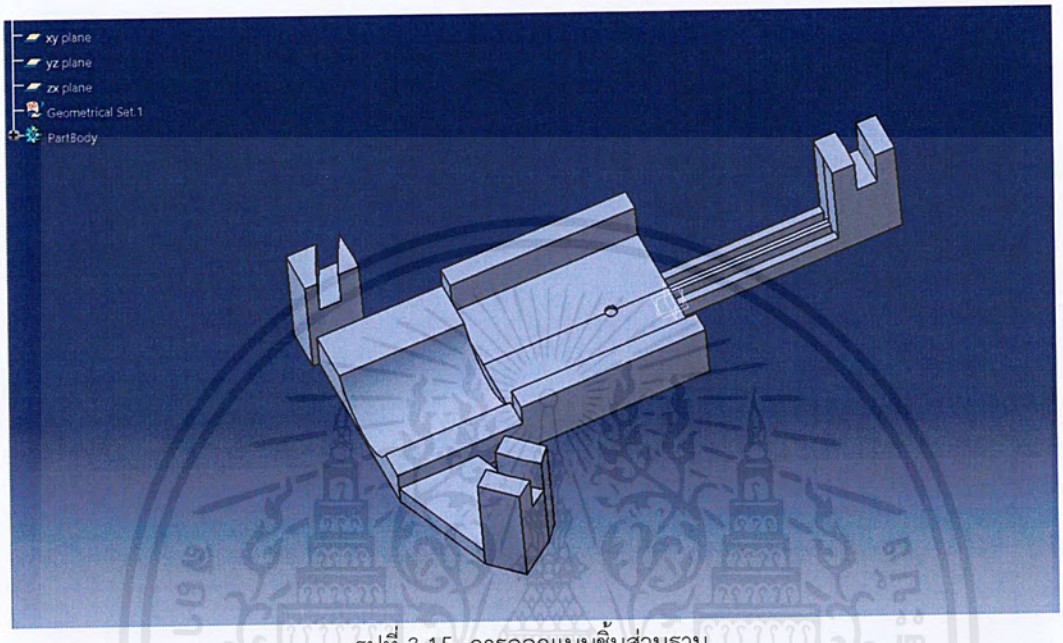
รูปที่ 3.14 การออกแบบเครื่องนับสกรูอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

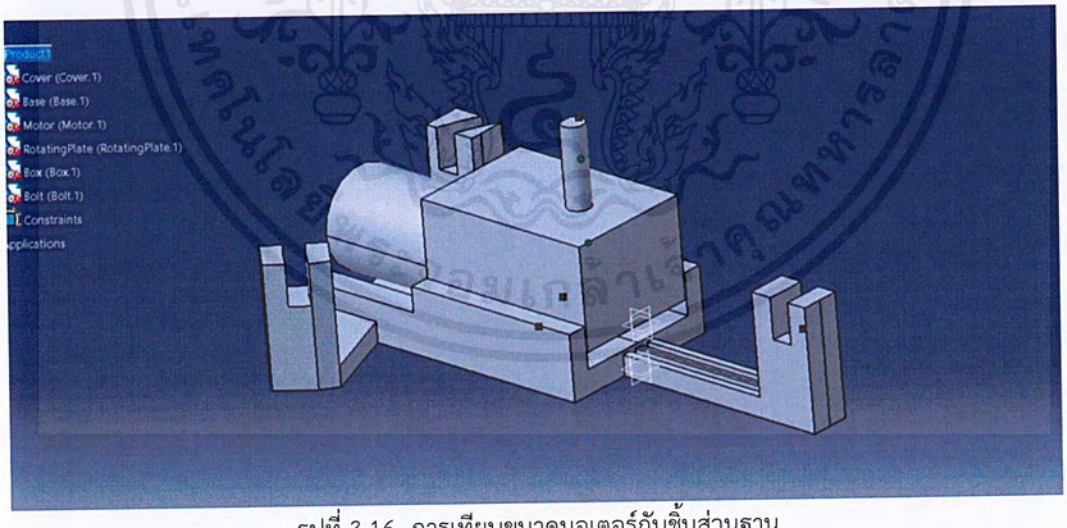
มีรายระเอียดในการออกแบบชิ้นส่วนดังนี้

3.6.1 ส่วนฐาน

มีการออกแบบให้มีช่องว่างเพื่อใช้ในการรองรับมอเตอร์ไฟฟ้ากระแสตรง รวมทั้งโครงสร้างอื่นๆของตัวเครื่อง โดยมีการวัดและจำลองขนาดของมอเตอร์ไฟฟ้าเพื่อนำมาเทียบเคียงกับแบบ



รูปที่ 3.15 การออกแบบชิ้นส่วนฐาน



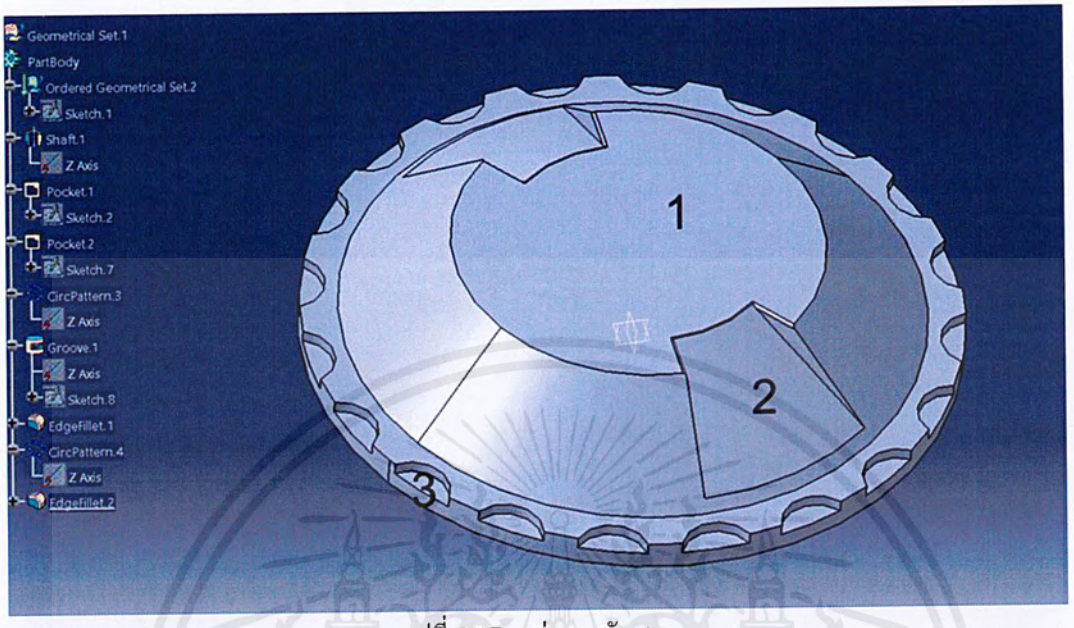
รูปที่ 3.16 การเทียบขนาดมอเตอร์กับชิ้นส่วนฐาน

3.6.2 แผ่นรองรับสกรูและที่บรรจุสกรู

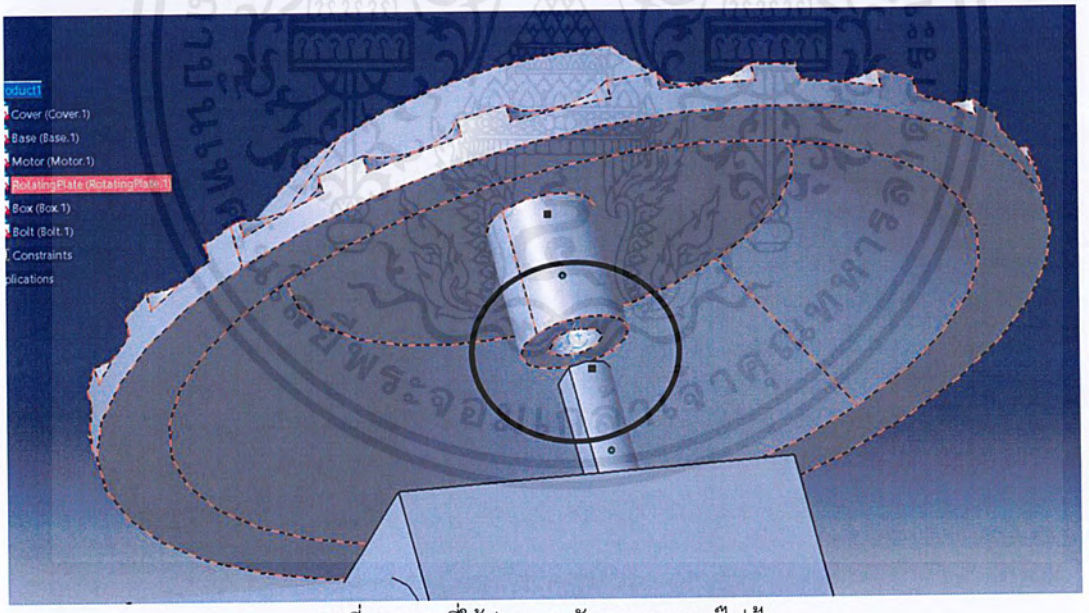
มีการออกแบบให้เป็นวงกลมเพื่อให้สะดวกต่อการหมุน โดยมีพื้นที่บริเวณด้านบน(1)ใช้ในการเก็บสกรูโดยมีพื้นที่ขนาด 2400 mmsq และพื้นที่ที่ลาดเอียงลง(2)มาเพื่อให้สกรูไหลตก และพื้นที่ที่เป็นช่องเพื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รองรับส่วนหัวของสกรู(3) จำนวน 20 ช่อง โดยบริเวณด้านล่างของตัวแผ่นจะมีรูเพื่อใช้ประกอบเข้ากับแกนของมอเตอร์ไฟฟ้า



รูปที่ 3.17 แผ่นรองรับสกรู

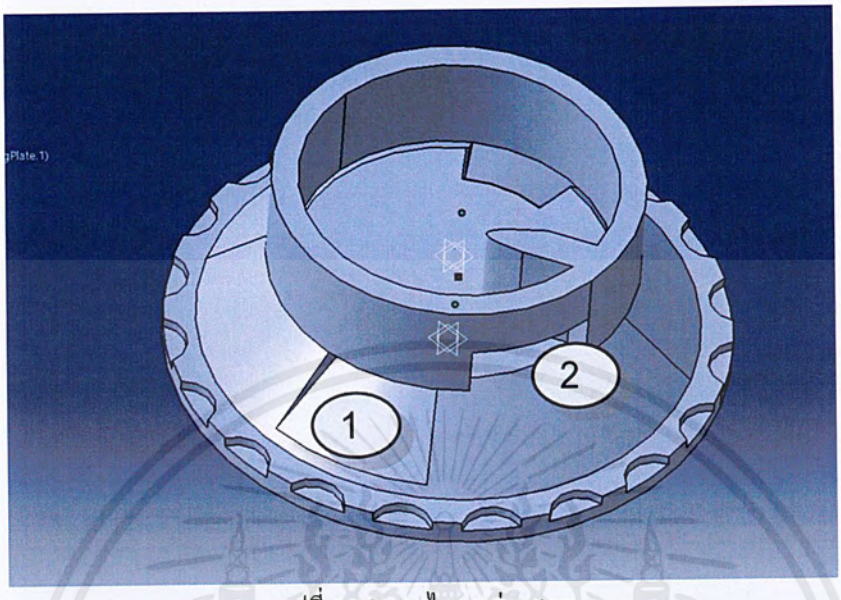


รูปที่ 3.18 รูที่ใช้ประกอบกับแกนมอเตอร์ไฟฟ้า

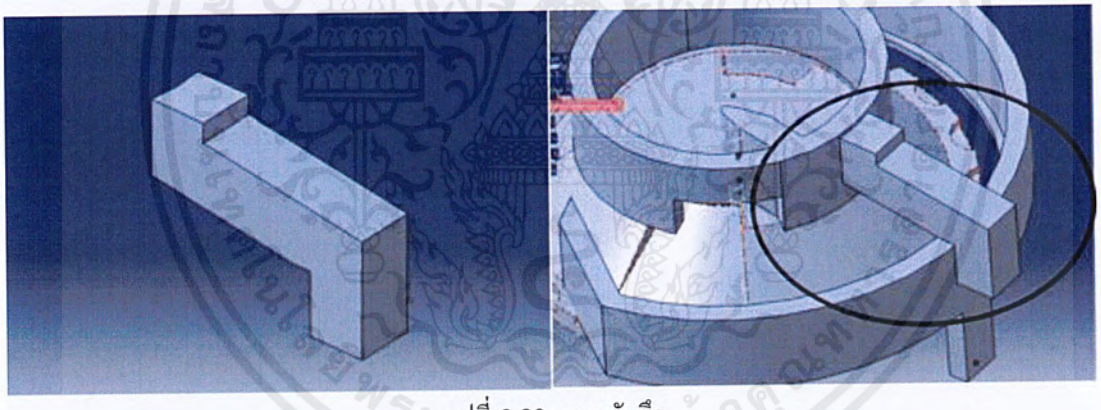
ตัวเครื่องถูกออกแบบให้จ่ายสกรูเมื่อพื้นที่ที่ลาดเอียงหมุนมาเจอกันกับ ส่วนที่เป็นช่องของบริเวณที่เก็บสกรูเท่านั้น (บริเวณ 1 และ 2 อยู่ในตำแหน่งเดียวกัน) เพื่อควบคุมปริมาณการจ่ายสกรูไม่ให้มากจนเกินไปอันจะทำให้การทำงานติดขัด และก่อให้เกิดความผิดพลาดในการนับ โดย ชิ้นส่วนจะถูกยึดเข้าด้วยกันด้วยแขนจับยึดดังรูปที่ 3.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวเครื่องมีปริมาตรสำหรับบรรจุสกรูทั้งหมด 2400*5 = 12000 mmcu โดยสกรู 1 ตัว มีปริมาตรประมาณ 306 mmcu ดังนั้น เครื่องสามารถบรรจุสกรูได้ 12000/306 = 39 ตัว



รูปที่ 3.19 กลไกการจ่ายสกรู

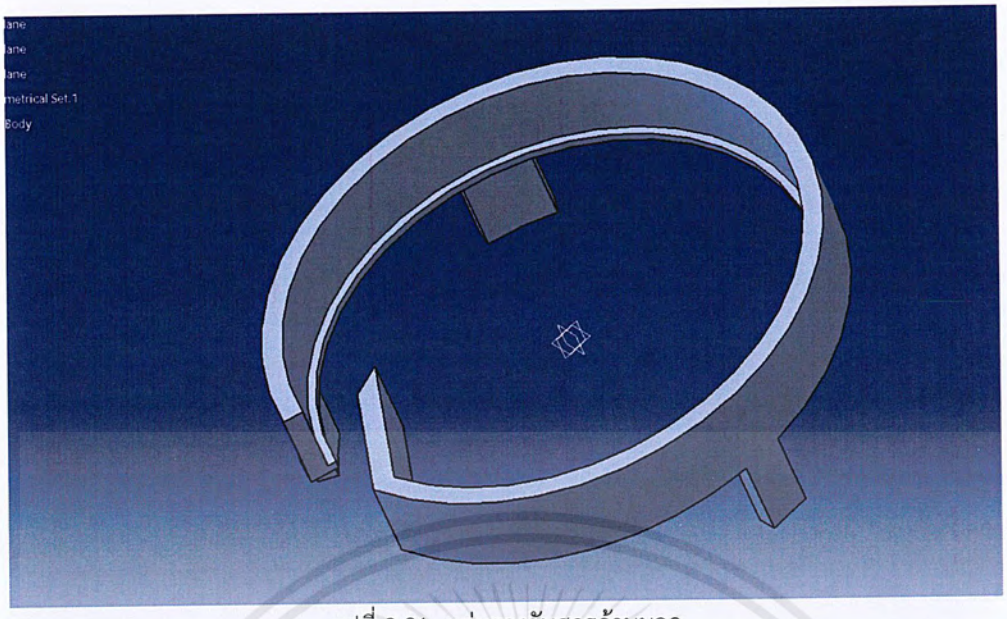


รูปที่ 3.20 แขนจับยึด

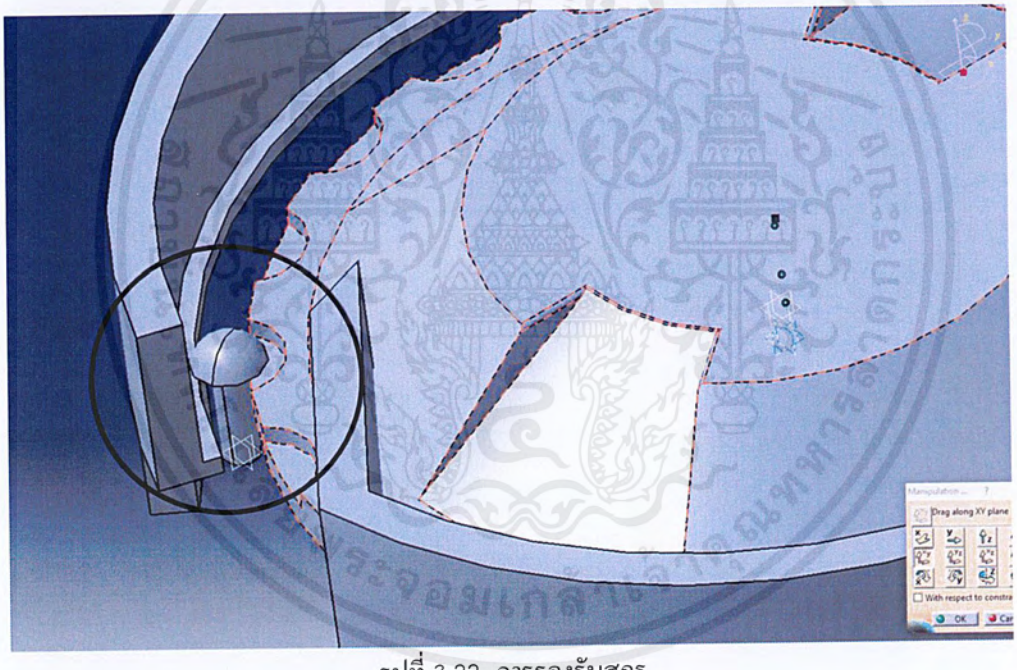
3.6.3 แผ่นรองรับสกรูด้านนอก

ออกแบบให้เป็นวงกลมและมีขนาดสอดคล้องกับแผ่นรองรับสกรู ทำหน้าที่ป้องกันสกรูตกออกจากเครื่อง และทำหน้าที่รองรับสกรู โดยแผ่นรองรับสกรูด้านนอกนี้จะมีช่องว่าง 5 mm และพื้นที่รองรับหัวสกรูเพื่อให้สกรูวางตัวในลักษณะดังรูปที่ 3.22 ขาทั้ง 3 ขาที่ยื่นออกมามีหน้าที่ประกอบกับตัวฐานเพื่อให้มีความแข็งแรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

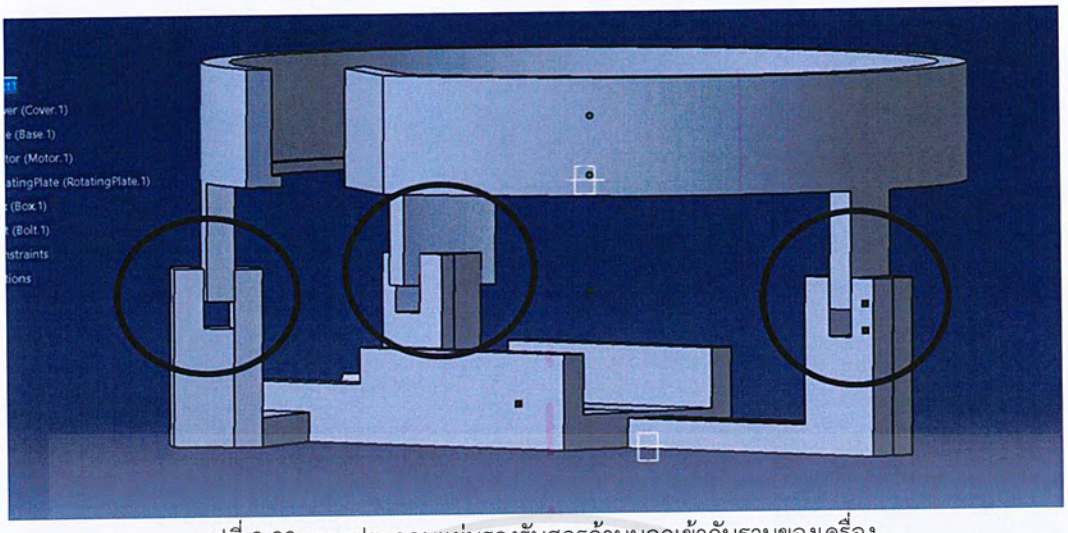


รูปที่ 3.21 แผ่นรองรับสกรูด้านนอก



รูปที่ 3.22 การรองรับสกรู

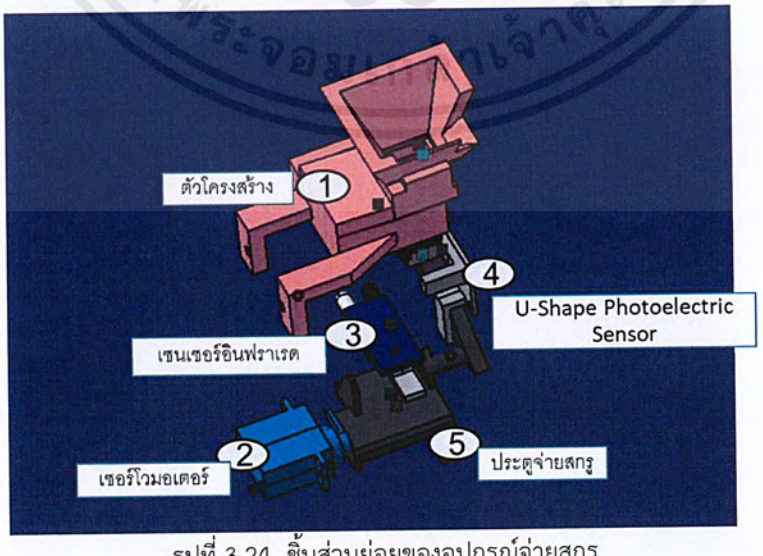
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.23 การประกอบแผ่นรองรับสกรูด้านนอกเข้ากับฐานของเครื่อง

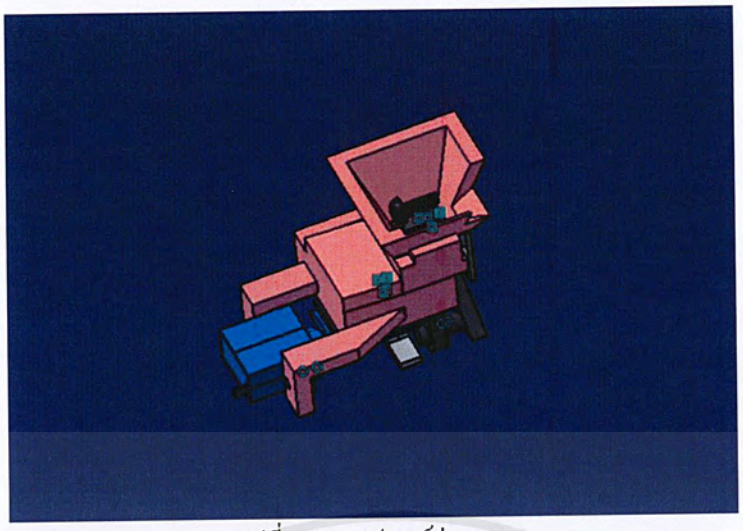
3.6.4 อุปกรณ์จ่ายสกรู

ชุดอุปกรณ์นี้ทำหน้าที่ในการรับสกรูที่ตกลงมาจากแผ่นหมุนด้านบน โดยสกรูจะตกผ่านเซนเซอร์และถูกนับที่บริเวณนี้ ในชิ้นส่วนนี้ประกอบด้วยชิ้นส่วนย่อยๆ 5 ชิ้นดังแสดงในรูปที่ 3.23 คือ 1. ตัวโครงสร้าง ทำหน้าที่เป็นแกนหลักในการยึดชิ้นส่วนอื่นๆเข้าด้วยกันด้วยช่อง และจุดยึดต่างๆ 2. เซอร์โวเซนเซอร์ ทำหน้าที่เปิดประตูเพื่อจ่ายสกรูออกไป 3. เซนเซอร์อินฟราเรด ตัวเซนเซอร์จะหันหน้าออกจากเครื่องเพื่อตรวจจับในกรณีที่มีวัตถุมาอยู่ในบริเวณใกล้เคียงและส่งคำสั่งไปเปิดประตูจ่ายสกรู 4. เซนเซอร์นับหรือ U-Shape Photoelectric Sensor จะถูกติดตั้งโดยสอดผ่านช่องว่างด้านหลังของตัวโครงสร้าง ทำหน้าที่ตรวจจับ และนับสกรูที่ตกผ่านลงมา 5. ประตูจ่ายสกรู จะถูกติดตั้งอยู่ล่างสุดของอุปกรณ์นี้ทำหน้าที่รองรับและจัดเก็บสกรูที่ถูกนับแล้วเปิดออกเพื่อจ่ายสกรู โดยชิ้นส่วนและอุปกรณ์ทั้งหมดจะถูกยึดติดเข้าด้วยกันดังรูปที่ 3.24



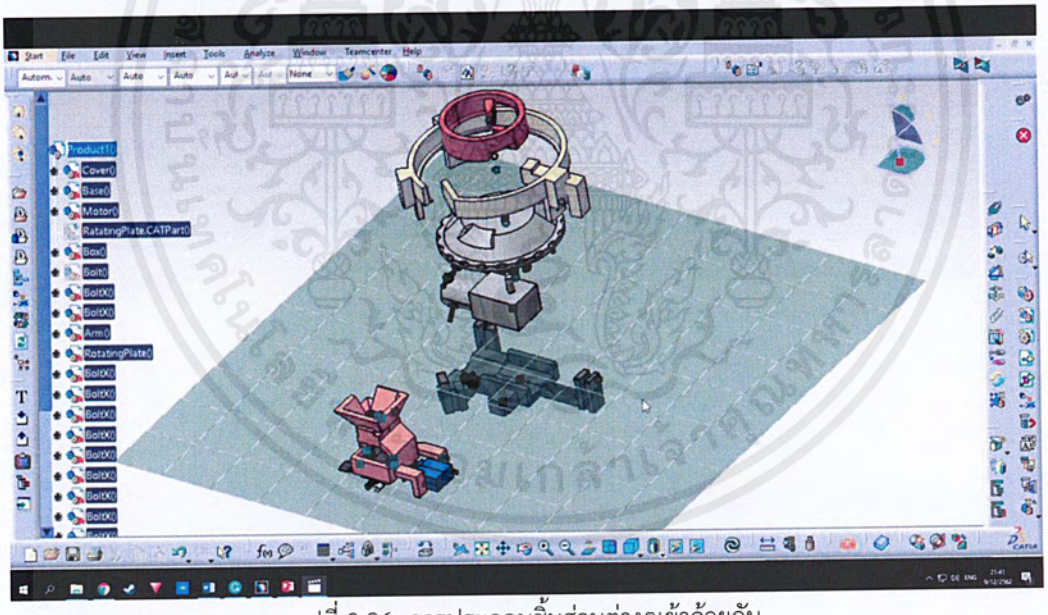
รูปที่ 3.24 ชิ้นส่วนย่อยของอุปกรณ์จ่ายสกรู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



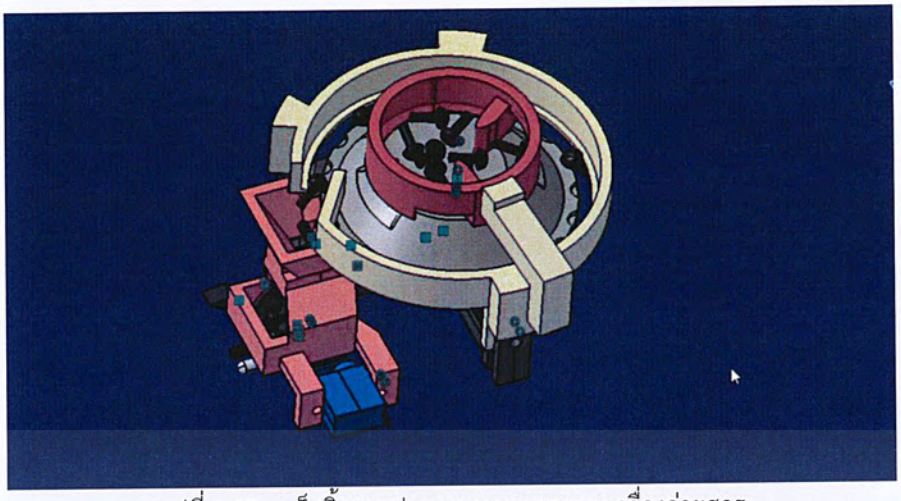
รูปที่ 3.25 อุปกรณ์จ่ายสกรู

หลังจากนั้นจะนำชิ้นส่วนที่ออกแบบแล้วทั้งหมดมาประกอบเข้าด้วยกัน เพื่อตรวจสอบความเข้ากันได้ของช่องว่างและข้อต่อต่างๆ พร้อมทั้งเช็คความถูกต้องเรียบร้อย จากนั้นจึงสร้างชิ้นงานโดยใช้เครื่องพิมพ์ 3 มิติ



รูปที่ 3.26 การประกอบชิ้นส่วนต่างๆเข้าด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

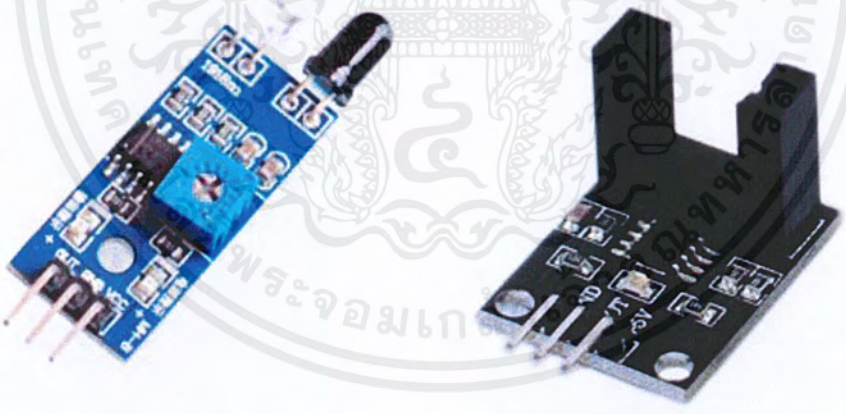


รูปที่ 3.27 เสร็จสิ้นการประกอบและออกแบบเครื่องจ่ายสกรู

3.7 การต่อวงจรควบคุม

วงจรที่ใช้ควบคุมเครื่องจ่ายสกรูนั้น ประกอบด้วยชิ้นส่วนอิเล็กทรอนิกส์หลายชิ้น ซึ่งได้มีการแจกแจงรายละเอียดข้อมูลพื้นฐาน หลักการทำงาน พร้อมทั้งขั้นตอนการต่อเข้ากับบอร์ด Arduino Uno R3 ดังนี้

3.7.1 IR Infrared Obstacle Avoidance Sensor Module



รูปที่ 3.28 โมดูลเซ็นเซอร์แสงสำหรับตรวจจับวัตถุกีดขวาง(ซ้าย) และเซนเซอร์รูปตัวยู(ขวา)

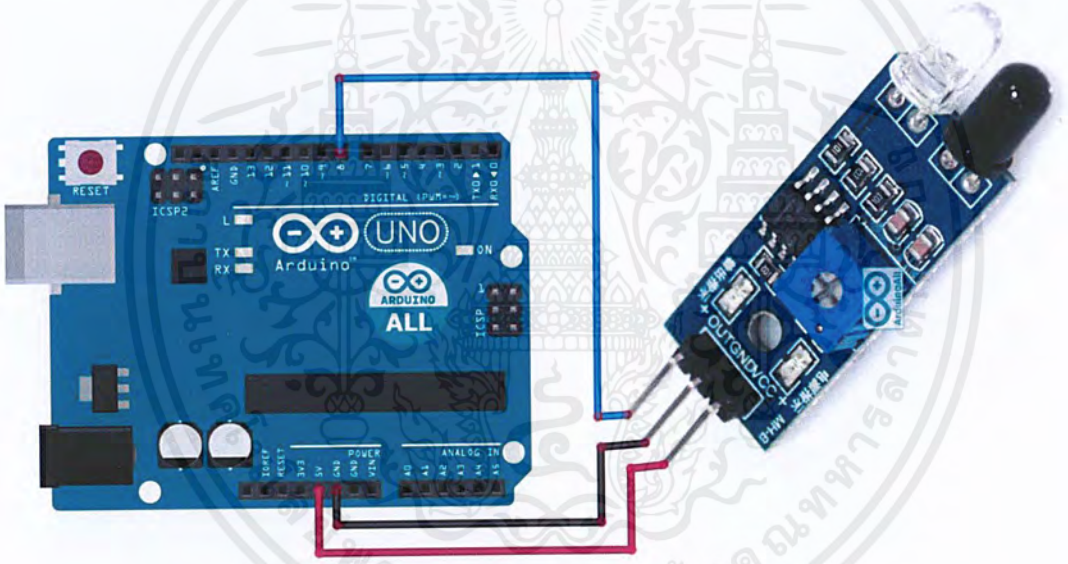
ผู้ศึกษาได้ใช้เซนเซอร์ตรวจจับวัตถุกีดขวางในรูปที่ 3.27 เพื่อตรวจจับวัตถุที่อยู่ในระยะ(มือของผู้ประกอบงาน) และเซ็นเซอร์รูปตัวยูเพื่อนับสกรูที่ตกผ่านโดยเซนเซอร์เหล่านี้ จะมีตัวรับและตัวส่ง infrared ในตัว ตัวสัญญาณ(สีขาว) infrared จะส่งสัญญาณออกมา และเมื่อมีวัตถุมาบัง คลื่นสัญญาณ infrared ที่ถูกสั่งออกมาจะสะท้องกลับไปเข้าตัวรับสัญญาณ (สีดำ) สามารถนำมาใช้ตรวจจับวัตถุที่อยู่ตรงหน้าได้ และสามารถปรับความไว ระยะการตรวจจับ ใกล้หรือไกลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายตัวเซ็นเซอร์แบบนี้จะมีตัวส่ง Emitter และ ตัวรับ Receiver ติดตั้งภายในตัวเดียวกัน ทำให้ไม่จำเป็นต้องเดินสายไฟทั้งสองฝั่ง เหมือนแบบ Opposed Mode ทำให้การติดตั้งใช้งานได้ง่ายกว่า แต่อย่างไรก็ตามจำเป็นต้องติดตั้งตัวแผ่นสะท้อนหรือ Reflector ไว้ตรงข้ามกับตัวเซ็นเซอร์เอง โดยโฟโต้เซ็นเซอร์แบบที่ใช้แผ่นสะท้อนแบบนี้จะเหมาะสำหรับชิ้นงานที่มีลักษณะทึบแสงไม่เป็นมันวาว เนื่องจากอาจทำให้ตัวเซ็นเซอร์เข้าใจผิดว่าเป็นตัวแผ่นสะท้อน และ ทำให้ทำงานผิดพลาดได้

ในสภาวะการทำงานปกติตัวรับ Receiver จะสามารถรับสัญญาณแสงจากตัวส่ง Emitter ได้ตลอดเวลา เนื่องจากลำแสงจะสะท้อนกับแผ่นสะท้อน Reflector อยู่ตลอดเวลา จะแสดงค่า เป็น 0 แต่เมื่อวัตถุ หรือ ชิ้นงานผ่านเข้ามาที่หน้าเซ็นเซอร์ จะเกิดการขวางลำแสงที่ส่งจากตัวส่ง Emitter ไปยังแผ่นสะท้อน จึงทำให้ตัวรับ Receiver ไม่สามารถรับลำแสงที่จะสะท้อนกลับมาได้ จะแสดงค่า เป็น 1

โดยเซนเซอร์ทั้งสองสามารถต่อเข้ากับบอร์ด Arduinoได้ในลักษณะเดียวกันดังนี้ นำขา A5 ต่อกับขาdigital ใดก็ได้ จากนั้นนำขา 5V ต่อกับ VCC และนำขา GND ต่อกับ GND ดังรูปที่ 3.29



รูปที่ 3.29 การต่อเซนเซอร์อินฟราเรดเข้ากับบอร์ด Arduino

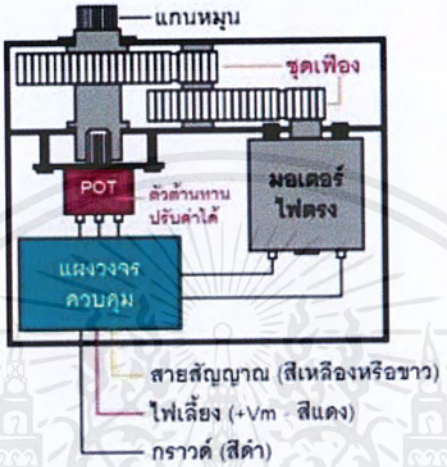
3.7.2 เซอร์โวมอเตอร์ S90G

Servo Motor เป็นอุปกรณ์ที่สามารถควบคุมเครื่องจักรกล หรือระบบการทำงานนั้นๆ ให้เป็นไปตามความต้องการ เช่น ควบคุมความเร็ว ควบคุมแรงบิดและควบคุมแรงตำแหน่ง โดยให้ผลลัพธ์ตามความต้องการที่มีความแม่นยำสูง ซึ่งมอเตอร์ทั่วไปไม่สามารถ ควบคุมในลักษณะงานเบื้องต้นได้

โดยเวอร์โวมอเตอร์จะประกอบไปด้วย Controller มีหน้าที่รับคำสั่งจากผู้ใช้งานแล้วควมคุมเซอร์โวมอเตอร์ แต่ Controller ไม่สามรถสั่งการควบคุมไปที่เซอร์โวเมอเตอร์ ได้โดยตรงเนื่องจาก ตัว Servo Driver จะเป็น ตัวกำหนดค่าตัวแปรหรือพารามิเตอร์ต่างๆ ให้กับตัวมอเตอร์ เพื่อให้ทำงานได้อย่างถูกต้องและแม่นยำ เพราะฉะนั้นเมื่อใช้เซอร์โวมอเตอร์ ก็จะต้องมี Servo Driver เสมอ

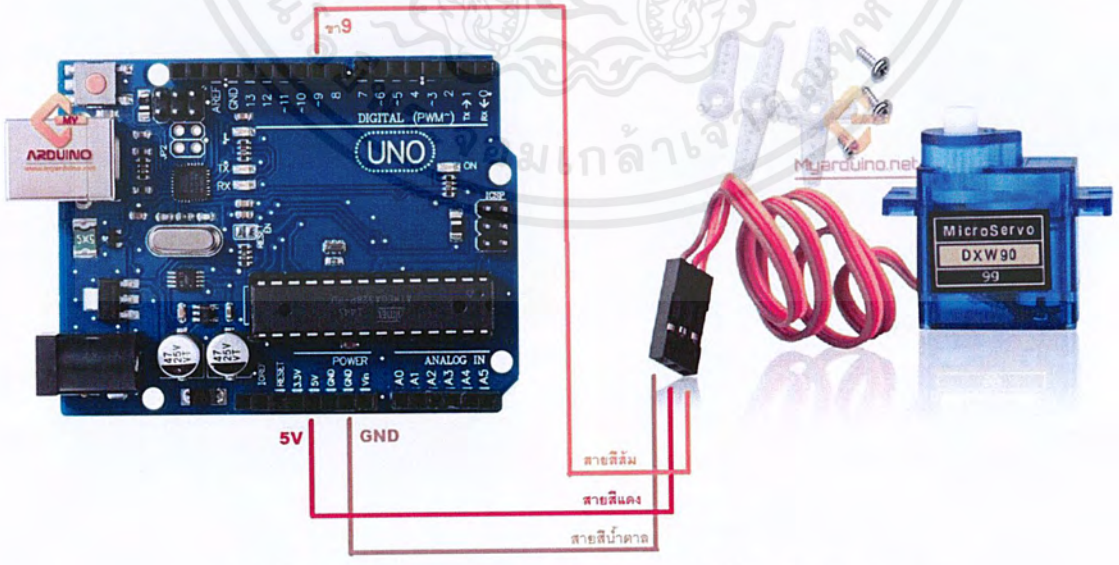
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายในเซอร์โวมอเตอร์ประกอบด้วย มอเตอร์ไฟตรงขนาดเล็ก,ชุดเฟืองทด,แผงวงจรควบคุม และตัวต้านทานปรับค่าได้(POT : Potentiometer) โดยแผงวงจรควบคุมจะมีวงจรป้อนกลับ เพื่อให้เซอร์โวมอเตอร์รับรู้ตำแหน่งของตัวเองได้ โดยผู้ใช้งานเพียงส่งสัญญาณพัลส์ออกไปควบคุมเท่านั้น ดังแสดงไดอะแกรมการทำงานของเซอร์โวมอเตอร์ในรูปที่ 3.30 แกนของมอเตอร์ไฟตรงจะต่อเข้ากับ ชุดเฟืองเพื่อลดความเร็วรอบลงส่งผลให้แรงบิดที่แกนหมุนมากขึ้น



รูปที่ 3.30 แสดงไดอะแกรมการทำงานของเซอร์โวมอเตอร์

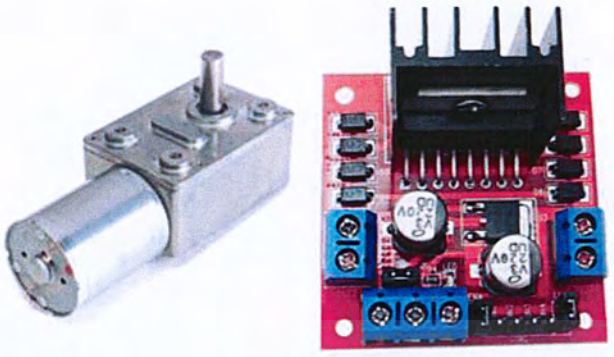
การต่อเข้ากับ Arduino Uno R3 นั้นสามารถทำได้ดังนี้ สายสีน้ำตาล ต่อกับ GND จากนั้นนำสายสีแดงต่อกับ 5V และสายสีส้มต่อกับ ขา digital ใดก็ได้ ดังรูปที่ 3.31



รูปที่ 3.31 การต่อวงจรเซอร์โวมอเตอร์ S90G เข้ากับบอร์ด Arduino

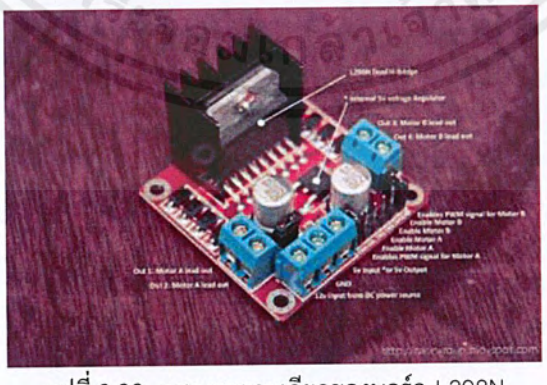
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.3 มอเตอร์ไฟฟ้ากระแสตรง และแผงควบคุม L298N



รูปที่ 3.32 มอเตอร์ไฟฟ้ากระแสตรง(ซ้าย) และแผงควบคุม L298N (ขวา)

L298N เป็นชุดขับมอเตอร์ชนิด H-Bridge ซึ่งส่วนใหญ่จะถูกนำไปใช้ในการควบคุมทิศทางและความเร็วของมอเตอร์ ซึ่งสามารถควบคุมมอเตอร์ได้ทั้งหมด 2 ตัว โดยมีหลักการทำงาน คือ วงจร H-Bridge ของ L298N จะขับกระแสเข้ามอเตอร์ ตามขั้วที่กำหนดด้วยลอจิกเพื่อควบคุมทิศทาง ส่วนความเร็วของมอเตอร์นั้นจะถูกควบคุมด้วย สัญญาณ PWM : Pulse Width Modulation หมายถึง การควบคุมช่วงจังหวะการทำงานของอิเล็กตรอน ลองจินตนาการถึงแปรงขดลวดในมอเตอร์เป็นระหัดวิดน้ำและอิเล็กตรอนเป็นน้ำที่ตกลงมาจากระหัดวิดน้ำ ค่าแรงดันไฟฟ้าก็คล้ายกับกระแสน้ำที่ไหลผ่านระหัดวิดน้ำด้วยความเร็วคงที่ ยิ่งกระแสน้ำไหลเร็วเท่าไรก็จะหมายความว่าแรงดันไฟฟ้ายิ่งสูงขึ้น แต่มอเตอร์มีอัตราความเร็วคงที่และสามารถเสียหายได้หากมีแรงดันไฟฟ้าสูงไหลผ่านหรือหยุดทันทีเพื่อที่จะหยุดมอเตอร์ ดังนั้น PWM คล้ายกับการควบคุมระหัดวิดน้ำให้ตักน้ำในจังหวะคงที่ที่กระแสน้ำคงที่ ยิ่งระหัดวิดน้ำหมุนเร็วเท่าไรช่วงของ pulse ก็จะยาวขึ้น ในทางกลับกันถ้าระหัดวิดน้ำหมุนช้าช่วงของ pulse จะสั้นลง ดังนั้นเพื่อยืดอายุการใช้งานของมอเตอร์จึงควรที่จะควบคุมมอเตอร์ด้วย PWM



รูปที่ 3.33 แสดงรายละเอียดของบอร์ด L298N

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลทางเทคนิค

Dual H bridge Drive Chip : L298N

แรงดันสัญญาณลอจิก : 5V Drive voltage: 5V-35V

กระแสของสัญญาณลอจิก : 0-36mA

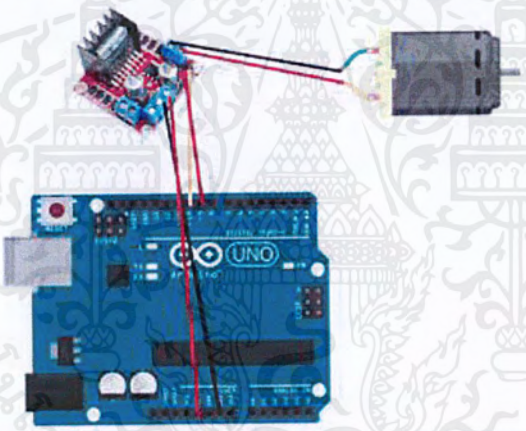
กระแสขับมอเตอร์ : สูงสุดที่ 2A (เมื่อใช้มอเตอร์เดียว)

กำลังไฟฟ้าสูงสุด : 25W

ขนาด : 43 x 43 x 26 มิลลิเมตร

น้ำหนัก : 26 กรัม

ในการต่อกับ Arduino นั้น ขา IN1,IN2,IN3 และ IN4 นั้น สามารถต่อกับพอร์ต Digital ใดๆก็ได้ เนื่องจาก 4 ขานี้ จะใช้ในการควบคุมสัญญาณลอจิกบอกทิศทางให้กับมอเตอร์ ส่วน ENA และ ENB นั้น จำเป็นที่จะต้องต่อกับพอร์ต Digital ที่รองรับ PWM เนื่องจากจะต้องใช้สัญญาณ PWM ในการควบคุมความเร็วของมอเตอร์ และใช้แหล่งจ่าย 12V 2A ในการขับเคลื่อนมอเตอร์ไฟฟ้ากระแสตรง



รูปที่ 3.34 แสดงการต่อใช้งานอุปกรณ์

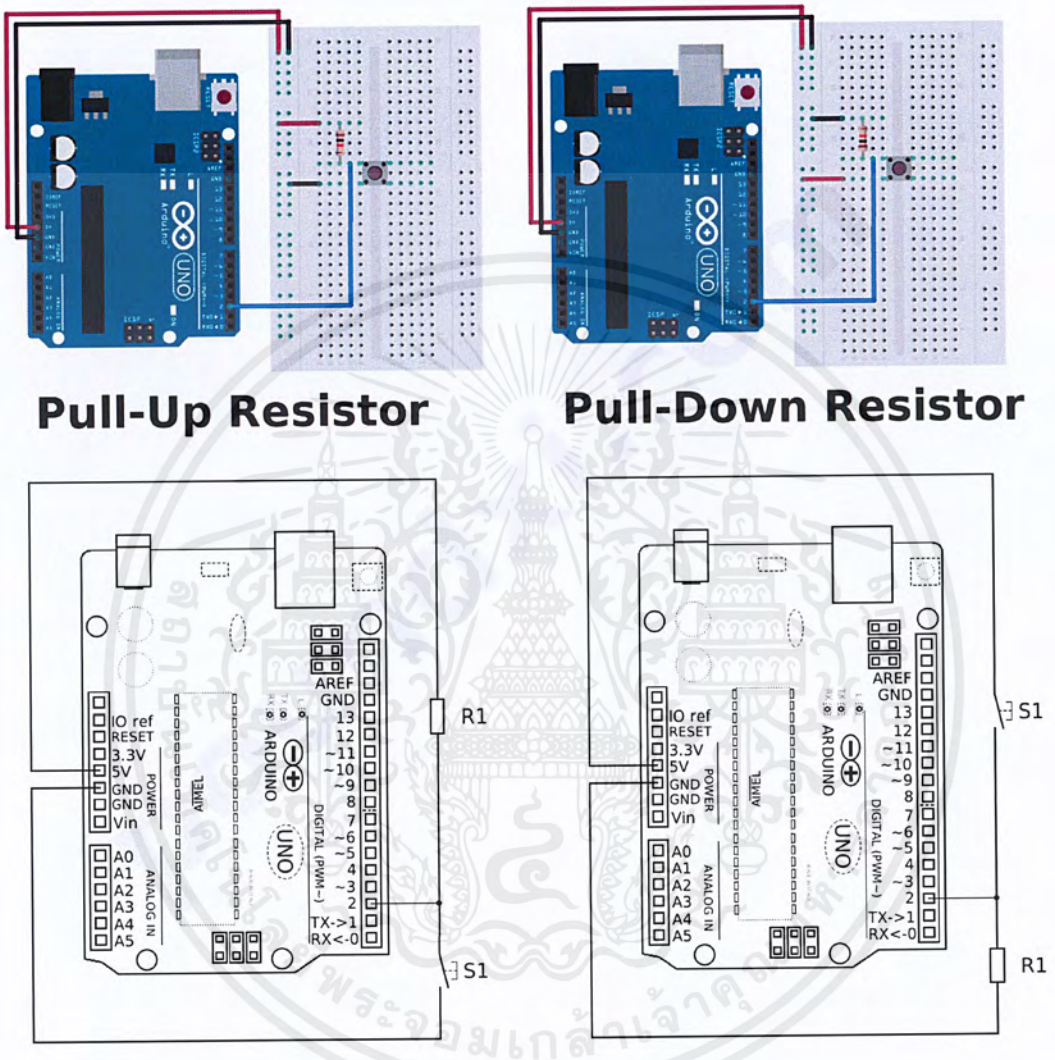
3.7.4 สวิทช์ปรับปริมานสกรู

เราจะต่อสวิตช์เพื่อใช้งานกับ Arduino โดยใช้การต่อวงจรที่เรียกว่า Pull-Up Resistor หรือ Pull-Down Resistor อย่างใดก็ได้ เพื่อป้องกันไม่ให้ขา Input อยู่ในสถานะ Floating จนอ่านค่าจากขา Input มาใช้งานไม่ได้ โดย Floating เป็นสถานะที่ขา Input ของ MCU ไม่ได้ต่ออยู่กับ VCC หรือ GND อย่างใดอย่างหนึ่งเลย ทำให้เมื่ออ่านค่าจากขา Input นั้น จะได้ค่าออกมาในลักษณะสุ่มระหว่าง LOW และ HIGH แบบไม่สามารถคาดเดาได้

โดยความแตกต่างในด้านการใช้งานของทั้ง 2 วงจรนี้ คือ Pull-Up Resistor เมื่อไม่ได้กดปุ่มจะอ่านค่าได้เป็น HIGH แต่เมื่อกดปุ่มจะอ่านค่าได้เป็น LOW ส่วน Pull-Down Resistor นั้นเมื่อไม่ได้กดปุ่ม จะอ่านค่าได้เป็น LOW แต่เมื่อกดปุ่มจะอ่านค่าได้เป็น HIGH ตรงจุดนี้สามารถเลือกใช้ได้ตามความเหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ว่าอยากได้สถานะตอนกดปุ่มเป็น HIGH หรือเป็น LOW ในที่นี้ผู้ศึกษาเลือกใช้การต่อแบบ Pull-Up resistor และเนื่องจากสวิทช์มีความต้านทานกระแสสูงสุด 1.2 mA และ บอร์ด Arduino จ่ายไฟ 5 V จากกฎของโอห์ม ควรใช้ตัวต้านทานมากกว่า 4167 Ω เพื่อความปลอดภัยผู้ศึกษาจึงใช้ตัวต้านทานขนาด 4700 Ω



รูปที่ 3.35 การต่อสวิทช์แบบ Pull-Up และ Pull-Down

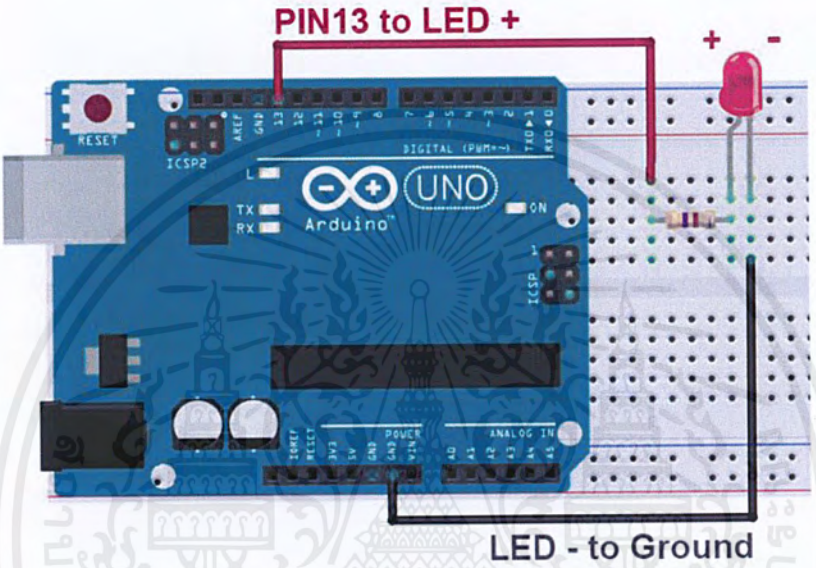
3.7.5 หลอดไฟ LED

หลอดไฟ LED หรือไดโอดเปล่งแสง (light-emitting diode) เป็นอุปกรณ์สารกึ่งตัวนำอย่างหนึ่ง จัดอยู่ในจำพวกไดโอด ที่สามารถเปล่งแสงในช่วงสเปกตรัมแคบ เมื่อถูกไบอัสทางไฟฟ้าในทิศทางไปข้างหน้า (ไบอัสตรง) ปรากฏการณ์นี้อยู่ในรูปของ electroluminescence สีของแสงที่เปล่งออกมานั้นขึ้นอยู่กับองค์ประกอบทางเคมีของวัสดุกึ่งตัวนำที่ใช้ และเปล่งแสงได้ใกล้ช่วงอัลตราไวโอเลต ช่วงแสงที่มองเห็น และช่วงอินฟราเรด ผู้พัฒนาไดโอดเปล่งแสงขึ้นเป็นคนแรก คือ นิก โฮโลยัก (Nick Holonyak Jr.) เกิด ค.ศ.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

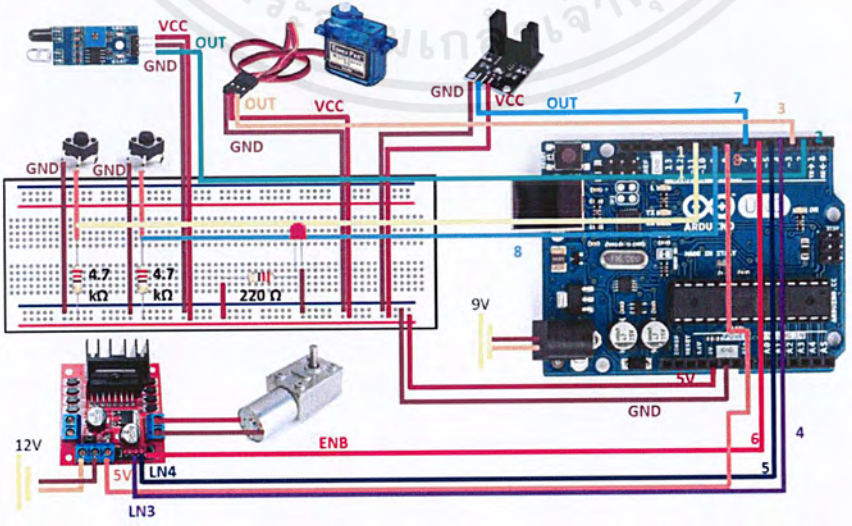
1928 แห่งบริษัทเจเนรัล อิเล็กทริก (General Electric Company) โดยได้พัฒนาไดโอดเปล่งแสงในช่วงแสงที่มองเห็น และสามารถใช้งานได้ในเชิงปฏิบัติเป็นครั้งแรก เมื่อ ค.ศ. 1962

สำหรับการต่อหลอดไฟ LED ขนาดเข้ากับ บอร์ด Aruidon Uno R3 นั้น ทำได้โดยต่อขาบวกเข้ากับแหล่งจ่ายไฟจากขา digital ขาใดก็ได้ และต่อขาลบเข้ากับ GND เนื่องจากหลอดไฟ LED ที่ใช้นั้นทนความต่างศักย์ได้ 3 V และใช้กระแส 15 mA และบอร์ด Arduino จ่ายไฟ 5 V จากกฎของโอห์มควรใช้ตัวต้านทานอย่างน้อย 200 Ω ผู้ศึกษาจึงเลือกใช้ตัวต้านทานขนาด 220 Ω ดังแสดงในรูปที่ 3.37



รูปที่ 3.36 การต่อหลอดไฟ LED เข้ากับบอร์ด Arduino

หลังจากเสร็จสิ้นการต่อวงจรของอุปกรณ์ทั้งหมดแล้ว ก็นำมาต่อเข้าด้วยกับกับ Bread Board ได้ดังรูปที่ 3.37

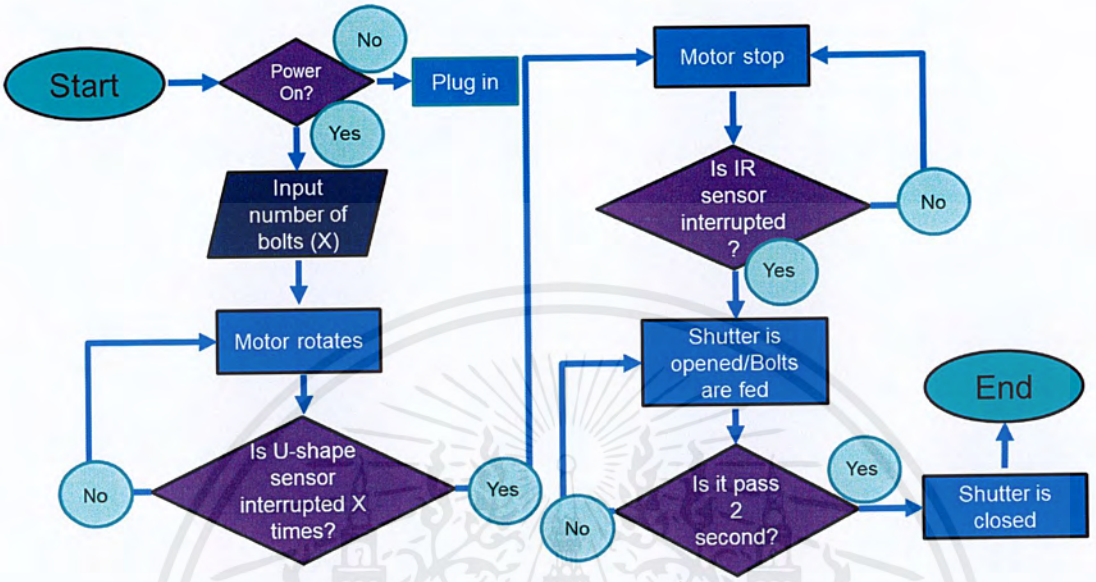


รูปที่ 3.37 แผงวงจรเครื่องจ่ายสกรู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8 เขียนโปรแกรม Aruidno IDE

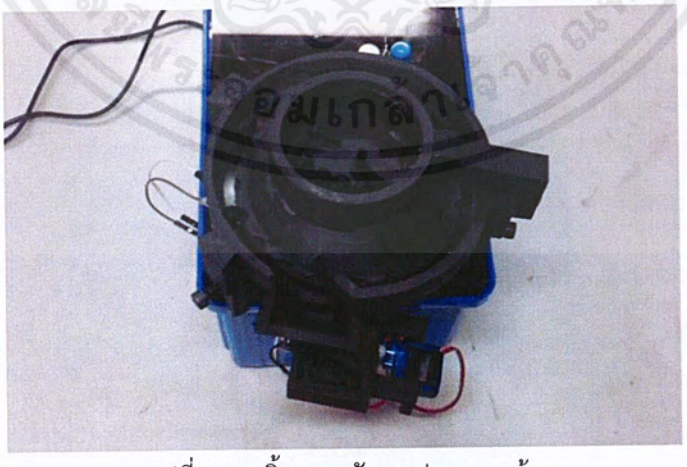
เขียนโปรแปรมด้วยพื้นฐานโครงสร้างภาษาซี ในซอฟต์แวร์ Arduino IDE เพื่อควบคุมการทำงานของบอร์ด Arduino Uno R3 และอุปกรณ์อิเล็กทรอนิกส์อื่นๆ ซึ่งมีผังการทำงานดังรูปที่ 3.38



รูปที่ 3.38 แสดงผังการทำงานของเครื่องจ่ายสกรู

3.9 ประกอบเครื่อง

หลังจากได้จากรับชิ้นส่วนจากเครื่องพิมพ์ 3 มิติแล้ว ก็นำชิ้นส่วนมาประกอบ ยึดเข้าด้วยกันโดยใช้สกรูยึดตามจุดต่างๆ พร้อมทั้งประกอบแผงวงจรอิเล็กทรอนิกส์ มอเตอร์ และเซ็นเซอร์ เข้ากับตัวเครื่องตามแบบที่วางไว้



รูปที่ 3.39 ชิ้นงานหลังจากประกอบแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9 ทดสอบการทำงานของเครื่อง

นำเครื่องจ่ายสกรูอัตโนมัติไปใช้งานจริงที่สถานีประกอบแผงหน้าปัดรถ TIP08XS ตามรูปที่ 3.41

รูปที่ 3.40 การติดตั้งเครื่องจ่ายสกรูภายในสายการผลิต

3.10 บันทึกผล

เก็บข้อมูลปัญหาเสียงกระเทาะที่บริเวณแผงหน้าปัดรถ 2 เดือน(ตุลาคม-พฤษจิกายน) หลังติดตั้งเครื่องจ่ายสกรูที่สถานี TIP08XS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการดำเนินงาน

4.1 การเปลี่ยนขั้นตอนการทำงาน

จากเดิมเป็นการหยิบสกรูจากกล่องข้างสายการผลิตซึ่งมีความผิดพลาดสูง ทั้งยังเป็นการเสียเวลาเนื่องจากขั้นตอนการขันแน่นสกรูนี้เป็นขั้นตอนสุดท้ายจึงทำให้มีการประกอบที่ปลายสถานีเนื่องจากผู้ประกอบงานต้องเดินย้อนกลับมาหยิบสกรูที่กล่องในสถานี และนับสกรูเองอย่างคร่าวๆ ซึ่งใช้เวลาประมาณ 4.87 วินาที

รูปที่ 4.1 แสดงการหยิบสกรูจากกล่องข้างสายการประกอบ

หลังการติดตั้งเครื่องจ่ายสกรูผู้ประกอบงานเพียงต้องหันหลังกลับมารับสกรูจากเครื่อง เนื่องจากเครื่องจ่ายสกรูนั้นถูกติดตั้งไว้ที่ปลายสถานี ทำให้ใช้เวลาเพียง 3.78 วินาที โดยใช้เวลาน้อยกว่าเดิม 22.38 เปอร์เซ็นต์ แต่ทั้งนี้เนื่องจากเครื่องจ่ายสกรู สามารถบรรจุสกรูได้ประมาณ 39 ตัว ซึ่งเพียงพอต่อการประกอบรถ 5 คัน ทำให้ผู้ประกอบงานต้อง ใช้เวลาประมาณ 5.42 วินาที ในการเติมสกรูใส่เครื่องทุกๆการประกอบรถ 5 คัน หลังจากรวมเวลาการบรรจุสกรูนี้เข้าไปแล้ว พบว่าใช้เวลาเฉลี่ย 4.86 วินาที ซึ่งถือว่ามีเวลาใกล้เคียงกับขั้นตอนเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.2 แสดงการรับสกรูจากเครื่องจ่ายสกรู

4.2 ข้อมูลรายละเอียดของเครื่องจ่ายสกรู

ตัวเครื่องรวมฐาน มีขนาดกว้าง 14.57 cm ยาว 26.37 cm สูง 16.73 cm มีมวลทั้งหมด 1.28 kg

มอเตอร์ไฟฟ้ามีความเร็วการหมุน 5.87 rps ใช้ไฟ 12V 2A

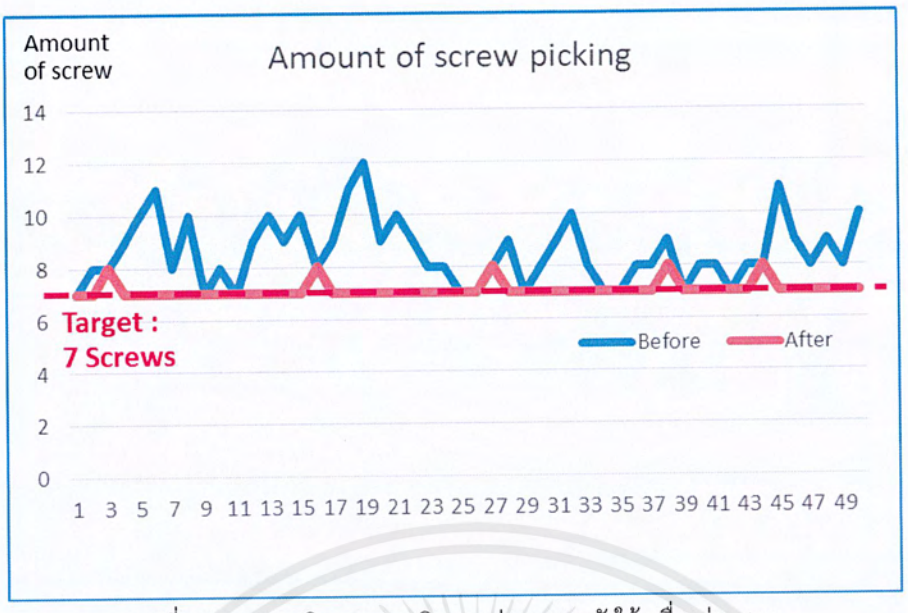
สามารถบรรจุสกรูได้สูงสุด 39 ตัว สามารถจ่ายสกรูได้ด้วยความเร็วสูงสุด 3.4 ตัว/วินาที

ใช้แหล่งจ่ายไฟฟ้า 2 แหล่ง โดยใช้ไฟจริง 8.72V 1.13A จ่ายให้กับมอเตอร์ไฟฟ้า และ ไฟ 7.89 V 1.78 A จ่ายให้กับบอร์ด Arduino ใช้กำลังไฟรวม 23.9 watt

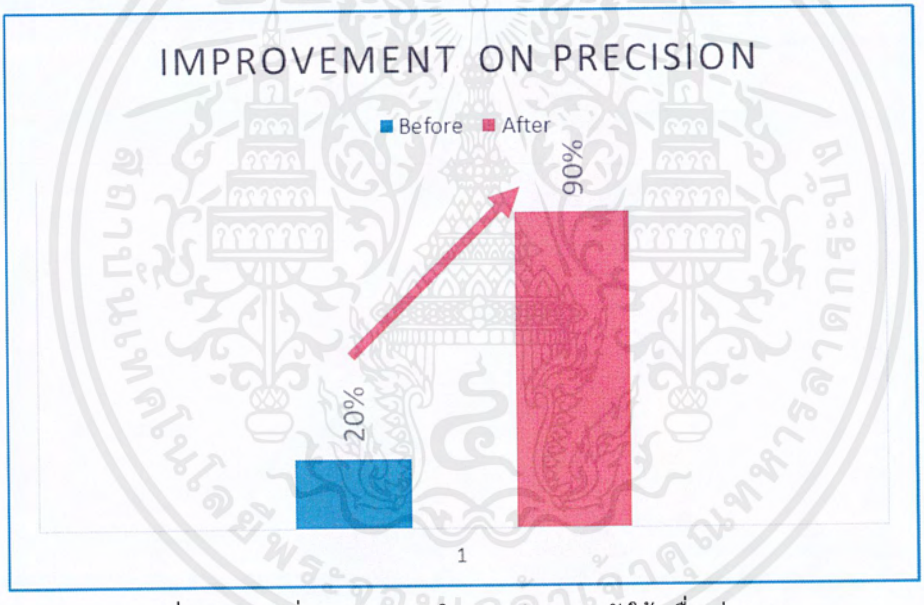
4.3 ผลการทำงานของเครื่องจ่ายสกรู

ก่อนติดตั้งเครื่องจ่ายสกรูอัตโนมัติจะสังเกตได้ว่าจากการหยิบสกรู 50 ครั้ง ปริมาณการหยิบมีการขยับขึ้นลง ไม่สามารถควบคุมได้ โดยมีค่าเฉลี่ยการหยิบสกรูที่ 8.56 ตัว หรือมีการหยิบมากกว่ามาตรฐานประมาณ 22.29 เปอร์เซ็น และยังมีความแม่นยำในการหยิบเพียง 20เปอร์เซ็นต์ หลังการติดตั้งเครื่องจ่ายสกรูอัตโนมัติจะสังเกตได้ว่า สามารถควบคุมปริมาณการหยิบสกรูได้ดีขึ้นได้ โดยมีค่าเฉลี่ยการหยิบสกรูที่ 7.1 ตัว หรือมีการหยิบมากกว่ามาตรฐานประมาณ 1.43 เปอร์เซ็น และมีความแม่นยำในการหยิบเพิ่มขึ้นเป็น 90 เปอร์เซ็นต์ ดังแสดงในรูปที่ 4.1 และ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงปริมาณการหยิบสกรูก่อนและหลังใช้เครื่องจ่ายสกรู



รูปที่ 4.4 การเพิ่มความแม่นยำในการหยิบสกรูหลังใช้เครื่องจ่ายสกรู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

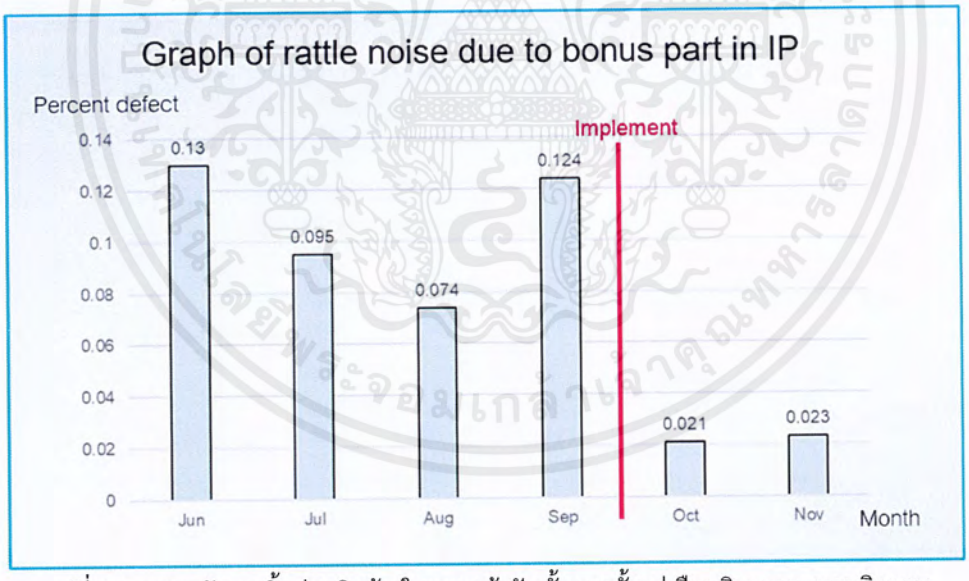
5.1 สรุปผลการวิจัย

ในช่วงการทำสหกิจศึกษา 6 เดือน เริ่มตั้งแต่เดือนมิถุนายน –พฤษจิกายน นั้นพบปัญหาเสียงกระเทาะทั้งหมด 6.68 ครั้งต่อรถ 1000 คัน หรือคิดเป็น 0.668 เปอร์เซ็นต์ ซึ่งปัญหาเหล่านี้นั้นเกิดจากหลายสาเหตุ ซึ่งหนึ่งในสาเหตุหลักคือ ชิ้นส่วนติดค้างอยู่ในแผงหน้าปัดรถซึ่งคิดเป็น 10.68 เปอร์เซ็นต์ ของสาเหตุทั้งหมด โดยสรุปผลได้ดังนี้

ก่อนใช้งานเครื่องจ่ายสกรู ช่วง 4 เดือนแรก(มิถุนายน-กันยายน) มีปัญหาสกรูติดค้างอยู่ในแผงหน้าปัด 0.078 เปอร์เซ็นต์ซึ่งคิดเป็น 13.87 เปอร์เซ็นต์ของปัญหาเสียงกระเทาะทั้งหมด

หลังใช้งานเครื่องจ่ายสกรู ช่วง 2 เดือนหลัง(ตุลาคม-พฤษจิกายน) มีปัญหาสกรูติดค้างอยู่ในแผงหน้าปัด 0.022 เปอร์เซ็นต์ ซึ่งคิดเป็น 6.29 เปอร์เซ็นต์ของปัญหาเสียงกระเทาะทั้งหมด

ดังนั้น สามารถสรุปได้ว่าหลังจากการติดตั้งและใช้งานเครื่องจ่ายสกรูที่สถานี TIP08XS พบว่าปัญหาสกรูติดค้างภายในแผงหน้าปัดรถลดลงจาก 0.106 เปอร์เซ็นต์ เป็น 0.022 เปอร์เซ็นต์ คือลดลงจากเดิม 79.2 เปอร์เซ็นต์



รูปที่ 5.1 แสดงปัญหาชิ้นส่วนติดค้างในแผงหน้าปัดทั้งหมดตั้งแต่เดือนมิถุนายน-พฤษจิกายน

5.2 ข้อเสนอแนะและปัญหาที่เกิดขึ้น

ในการทำโครงการเพื่อแก้ปัญหาส่วนใหญ่ในโรงงานผลิตและประกอบรถยนต์นั้น ส่วนมากจะใช้หลักการ Six-Sigma ซึ่งเป็นการนำข้อมูลปัญหามาวิเคราะห์โดยใช้ความรู้ทางสถิติเพื่อหาสาเหตุและแนวทางการแก้ไขปัญหา จึงจำเป็นอย่างมาก ที่ผู้ศึกษาจำเป็นต้องมีความรู้ทางสถิติในระดับพอสมควร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการใช้งานแผงวงจร Arduino นั้น ควรศึกษาข้อมูลของบอร์ดแต่ละชนิดเพื่อทำความเข้าใจในพื้นฐาน จุดเด่นและจุดด้อยของบอร์ดนั้นๆ เพื่อการเลือกใช้งานที่เหมาะสม นอกจากนี้ยังควรออกแบบ และวางแผนอย่างรอบคอบ ในกรณีที่จำเป็นต้องใช้อุปกรณ์ทางไฟฟ้า และอิเล็กทรอนิกส์อื่นๆ ร่วมกับบอร์ด Arduino ตลอดจนศึกษาการเขียนโปรแกรมชุดคำสั่งโดยใช้ภาษาซี และการใช้งานไลบรารี่ที่เหมาะสม เพื่อป้องกันปัญหาอื่นๆที่ตามมา เช่น ปัญหาแหล่งจ่ายไฟไม่เพียงพอ ปัญหาอุปกรณ์ทำงานร่วมกันไม่ได้ ปัญหาไม่สามารถอัพโหลดชุดคำสั่งรวมทั้งปัญหาโปรแกรมและอุปกรณ์ทางไฟฟ้าทำงานขัดข้อง ทั้งนี้ควรมีความรู้ทางไฟฟ้าพื้นฐานเพื่อใช้ในการออกแบบและต่อวงจรด้วย

ในการสร้างเครื่องนับสกรูนั้นมีส่วนที่สำคัญที่สุด 3 ส่วนคือ ส่วนแรก ส่วนที่ใช้บรรจุสกรู จำเป็นต้องออกแบบให้สามารถบรรจุสกรูได้มาก ทั้งยังต้องสามรถควบคุมปริมาณการจ่ายสกรูได้ เพื่อป้องกันปัญหาการจ่ายสกรูมากเกินไปทำให้เกิดปัญหาการติดขัดกันของสกรู ส่วนที่สอง ส่วนที่ใช้ลำเลียงสกรู จำเป็นต้องออกแบบให้พื้นผิวมีแรงเสียดทานน้อยที่สุดเพื่อป้องกันปัญหาการติดขัดกันระหว่างสกรูและพื้นผิวของเครื่อง และควรมีความแข็งแรงมากกว่าสกรูที่ใช้เนื่องจากมอเตอร์หมุนด้วยแรงบิดที่มาก ทำให้สกรูขูดขีดพื้นผิวของตัวเครื่อง ส่งผลให้พื้นผิวของเครื่องเสียหายอันจะนำไปสู่ปัญหาที่กล่าวมาก่อนหน้านี้ นอกจากนี้ควรออกแบบให้สามารถคัดกรองสกรูให้ผ่านไปได้ทีละตัวเพื่อความสะดวกในการนับ และส่วนสุดท้าย ส่วนการนับ ควรออกแบบให้มีพื้นที่กว้างพอให้สกรูไหลผ่าน หรือตกผ่าน ควรเลือกใช้ระบบการนับ และเซ็นเซอร์ที่มีความเหมาะสมกับวัตถุที่ต้องการนับและระบบการลำเลียง ทั้งนี้ควรตั้งความหน่วงของเซ็นเซอร์ให้เหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] ผศ.ดร.วรพงศ์ ตั้งศรีรัตน์. เซ็นเซอร์และทรานสดิวเซอร์ : ทฤษฎีและการประยุกต์ใช้ในระบบการวัด และระบบควบคุม. กรุงเทพมหานคร : โรงพิมพ์ซีเอ็ดยูเคชั่น. 2548.

[2] จีราวุธ วารินทร์. Arduino Uno พื้นฐานสำหรับงาน IOT. กรุงเทพมหานคร : โรงพิมพ์รีไววา. 2561.

[3] My arduino. สอนใช้งาน Arduino เริ่มต้นติดตั้งโปรแกรม Arduino IDE. สืบค้นเมื่อ 17 กรกฎาคม 2562. จากเว็ปไซต์ https://www.myarduino.net/. 2561.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โค้ดโปรแกรม Arduino IDE

```
#include <Servo.h>                    //เพิ่ม Servo เข้าไปจากไลบรารี่
int IRsensor = 2;                     //ประกาศตัวแปร IRsensor ไว้ที่พอร์ตหมายเลข 2
Servo servo;                          // ประกาศตัวแปร servo ควบคุม servo motor
int val=0;                            // ประกาศคัวแปร val มีค่า 0
int powerservo = 3;                   // ประกาศตัวแปร powerservo ไว้ที่พอร์ต
                                          หมายเลข 3
int MotorPin3 =4 ;                    // ประกาศตัวแปรMotorPin3ไว้ที่พอร์ตหมายเลข4
int MotorPin4 = 5;                    // ประกาศตัวแปรMotorPin4ไว้ที่พอร์ตหมายเลข5
int Speed = 6;                        // ประกาศตัวแปร Speed ไว้ที่พอร์ตหมายเลข 6
int countsensor = 7;                  // ประกาศตัวแปรcountsensorไว้ที่พอร์ต
                                          หมายเลข 7
int counter = 0;                      // ประกาศตัวแปร counter ไว้ที่พอร์ตหมายเลข 0
int Power = 8;                        // ประกาศตัวแปร Power ไว้ที่พอร์ตหมายเลข 8
int switchup = 9;                     // ประกาศตัวแปร switchupไว้ที่พอร์ตหมายเลข 9
int switchdown = 11;                  // ประกาศตัวแปร switchdown ไว้ที่พอร์ต
                                          หมายเลข 11
int statedown = 0;                    // ประกาศตัวแปร statedown มีค่า 0
int Open = 0;                         // ประกาศตัวแปร open มีค่า 0
int stateup = 0;                      // ประกาศตัวแปร stateup มีค่า 0
int x= 7;                             // ประกาศตัวแปร x มีค่า 7 (ปริมาณสกรูเริ่มต้นมี
                                          ค่า 7)
int light =12;                        // ประกาศตัวแปร light ไว้ที่พอร์ตหมายเลข 12
                                          (ควบคุมไฟ LED)
void setup() {                        // สร้างฟังก์ชัน void setup
 servo.attach(3);                     // อินพุตเซอร์โวมอเตอร์
  pinMode(IRsensor,INPUT);            //กำหนดตัวแปรIRsensor ให้รอรับค่า/กระแส
  pinMode(powerservo,OUTPUT);         //กำหนดตัวแปรpowerservo ให้ส่งค่า/กระแส
  pinMode(MotorPin3,OUTPUT);          //กำหนดตัวแปรMotorPin3 ให้ส่งค่า/กระแส
  pinMode(MotorPin4,OUTPUT);          //กำหนดตัวแปรMotorPin4 ให้ส่งค่า/กระแส
  pinMode(Speed,OUTPUT);              //กำหนดตัวแปรSpeed ให้ส่งค่า/กระแส
  pinMode(countsensor,INPUT);         //กำหนดตัวแปรcountsensor ให้รอรับค่า/กระแส
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  pinMode(Power,OUTPUT);                    //กำหนดตัวแปรPower ให้ส่งค่า/กระแส
  pinMode(switchup,INPUT);                  //กำหนดตัวแปรswitchup ให้รอรับค่า/กระแส
  pinMode(light,OUTPUT);                    //กำหนดตัวแปรlight ให้ส่งค่า/กระแส
}
void loop() {                               //สร้างฟังก์ชัน void loop
val = digitalRead(IRsensor);                //ให้เก็บค่าที่อ่านได้จากเซนเซอร์IR ไว้ที่ตัวแปร val
servo.write(100);                           //ปรับมุมเริ่มต้นของเซอร์โวมอเตอร์ไปที่ 100 องศา
int iscount = digitalRead(countsensor);     //ประกาศตัวแปร iscount ให้เก็บค่าที่อ่านได้จาก
                                                เซนเซอร์รูปตัวยู

stateup = digitalRead(switchup);            //ให้เก็บค่าสวิทช์สีแดงไว้ที่ตัวแปร stateup
statedown  = digitalRead(switchdown);       //ให้เก็บค่าสวิทช์สีน้ำเงินไว้ที่ตัวแปร statedown
  if (Open == 0)                            //ประกาศฟังก์ชั่น if ให้เริ่มทำงานเมื่อตัวแปร open
                                                มีค่า 0

  {
    digitalWrite(MotorPin3, HIGH);          //ตั้งค่าให้มอเตอร์กระแสตรงหมุนทวนเข็ม
    digitalWrite(MotorPin4, LOW);           //ตั้งค่าให้มอเตอร์กระแสตรงหมุนทวนเข็ม
    digitalWrite(Power, HIGH);              //ตั้งให้จ่ายกระแสให้มอเตอร์กระแสตรง
    analogWrite(Speed, 80);                 //ตั้งให้จ่ายกระแส 80 จาก 255 คิดเป็น 31.37
                                                เปอร์เซ็นต์มีความเร็วรอบ 5.87 rps

    if (iscount ==1)                        //ประกาศฟังก์ชั่น if ให้ทำงานในเงื่อนไขตัวแปร
                                                iscount มีค่าเท่ากับ 1(สกรูตกผ่าน
                                                เซ็นเซอร์รูปตัวยู)

      {
        counter++;                          //ให้ตัวแปร counter บวกขึ้น 1 (นับจำนวนสกรู)
        digitalWrite(MotorPin3, LOW);       //ตั้งค่าให้มอเตอร์กระแสตรงหมุนตามเข็ม
        digitalWrite(MotorPin4, HIGH);      //ตั้งค่าให้มอเตอร์กระแสตรงหมุนตามเข็ม
        analogWrite(Speed, 255);            //ตั้งให้จ่ายกระแส 255 จาก 255 คิดเป็น 100
                                                เปอร์เซ็นต์มีความเร็วรอบ 21.23 rps

        digitalWrite(Power, HIGH);          //ตั้งให้จ่ายกระแสให้มอเตอร์กระแสตรง
        delay(80);                          //ตั้งดีเลย์ของเซ็นเซอร์รูปตัวยูและการหมุนตามเข็ม
                                                ไว้ที่ 80 ms

        digitalWrite(MotorPin3, HIGH);      //ตั้งค่าให้มอเตอร์กระแสตรงหมุนทวนเข็ม
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
digitalWrite(MotorPin4, LOW);          //ตั้งค่าให้มอเตอร์กระแสตรงหมุนทวนเข็ม
digitalWrite(Power, HIGH);             //ตั้งให้จ่ายกระแสให้มอเตอร์กระแสตรง
analogWrite(Speed, 80);                //ตั้งให้จ่ายกระแส 80 จาก 255 คิดเป็น 31.37
                                              เปอร์เซ็นต์มีความเร็วรอบ 5.87 rps
if (counter >= x)                      //ประกาศฟังก์ชั่น if ให้ทำงานในเงื่อนไขตัวแปร
                                              counter มีค่ามากกว่าหรือเท่ากับ x(นับ
                                              เสร็จ)
  {
    digitalWrite(MotorPin3, LOW);      //ตั้งค่าให้มอเตอร์กระแสตรงหมุนตามเข็ม
    digitalWrite(MotorPin4, HIGH);     //ตั้งค่าให้มอเตอร์กระแสตรงหมุนตามเข็ม
    analogWrite(Speed, 255);           //ตั้งให้จ่ายกระแส 255 จาก 255 คิดเป็น 100
                                              เปอร์เซ็นต์มีความเร็วรอบ 21.23 rps
    digitalWrite(Power, HIGH);         //ตั้งให้จ่ายกระแสให้มอเตอร์กระแสตรง
    delay(100);                        //ให้ชุดคำสั่งหมุนมอเตอร์ตามเข็มทำงานค้าง100
                                              ms
    digitalWrite(Speed, LOW);          //ตั้งค่าให้มอเตอร์กระแสตรงหยุดทำงาน
    digitalWrite(Power, LOW);          //ตั้งค่าให้มอเตอร์กระแสตรงหยุดทำงาน
    digitalWrite(light, HIGH);         //ตั้งค่าให้ไฟ LED ติด
    Open = 1;                          //ให้ตัวแปร open มีค่า 1 (หลุดออกจากลูป)
  }
}
 else if(stateup == HIGH)              //ประกาศให้ทำงานเมื่อมีการกดสวิทช์สีแดง
{
    x++;                               //ตัวแปร x เพิ่มขึ้น 1(เพิ่มจำนวนสกรูที่ต้องการ)
    digitalWrite(light, HIGH);         //ไฟ LED ติด
    delay(300);                        //ให้ไฟ LED ติดเป็นเวลา 300 ms
    digitalWrite(light, LOW);          //ไฟ LED ดับ
}
  else if(statedown == HIGH and x > 0) //ประกาศให้ทำงานเมื่อมีการกดสวิทช์สีน้ำเงิน
{
    x--;                               //ตัวแปร x ลดลง 1(ลดจำนวนสกรูที่ต้องการ)
    digitalWrite(light, HIGH);         //ไฟ LED ติด
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            delay(300);                          //ให้ไฟ LED ติดเป็นเวลา 300 ms
            digitalWrite(light, LOW);            //ไฟ LED ดับ
        }
    }
    else if (val == 0 and Open == 1 )            //ประกาศให้ทำงานเมื่อการนับเสร็จสิ้น และ
                                                          เซ็นเซอร์IR จับสัญญานได้
        {       delay(500);                      //ดีเลย์ 500 ms
                digitalWrite(light, LOW);        //ให้ไฟ LED ดับ
                analogWrite(powerservo, HIGH);   //จ่ายไฟให้เซอร์โวมอเตอร์
                servo.write(50);                 //ให้เซอร์โวมอเตอร์หมุนไปที่มุม 50 องศา
                delay(2000);                     //ให้เซอร์โวมอเตอร์ค้างที่ตำแหน่งนั้น 2000 ms
                Open = 0;                        //ให้ตัวแปร open มีค่า 0(เริ่มทำงานใหม่)
                counter = 0;                     //ให้ตัวแปล counter มีค่า 0(เริ่มการนับใหม่)
        }
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้