

**ระบบสะสมแต้ม**

**นิติกานต์ เพ็ชแหวน**

**โครงงานนี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิทยาศาสตรบัณฑิต**

**สาขาเทคโนโลยีสารสนเทศ คณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา**

**ปีการศึกษา 2563**

**ลิขสิทธิ์ของคณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา**



**Collecting points system**

**Nitikarn Pechwan**

**A PROJECT SUMMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE BACHELOR OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY**

**FACULTY OF INFORMATICS, BURAPHA UNIVERSITY**

**ACADEMIC YEAR 2020**

**COPYRIGHT OF FACULTY OF INFORMATICS, BURAPHA UNIVERSITY**

**ใบรับรอง**

|  |  |
| --- | --- |
| **หัวข้อรายงาน** | ระบบสะสมแต้ม |
| **นิสิต** | นิติกานต์ เพ็ชแหวน |
| **รหัสประจำตัว** | 60160040 |
| **อาจารย์ที่ปรึกษาโครงงาน** | อาจารย์ ดร.ประจักษ์ จิตเงินมะดัน |
| **ระดับการศึกษา** | วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศ |
| **คณะ** | วิทยาการสารสนเทศ มหาวิทยาลัยบูรพา |
| **ปีการศึกษา** | 2563 |

# บทคัดย่อ

การสูญหายของแต้มสะสมที่เกิดจากการลืมบัตรสะสมหรือบัตรสะสมเสียหายนั้น เป็นปัญหาที่ต้องเริ่มสะสมแต้มใหม่ตั้งแต่ต้น ซึ่งการเริ่มสะสมแต้มใหม่เป็นปัญหาที่เกิดขึ้นได้บ่อยครั้ง หรือระบบสะสมแต้มที่มีอยู่ตัวเลือกวิธีการสะสมมีอย่างจำกัด จึงอาจไม่ใช่วิธีการเก็บสะสมแต้มที่ผู้ประกอบการต้องการจริง

ดังนั้น ผู้พัฒนาจึงศึกษาและทำการพัฒนาระบบสะสมแต้มแทนการใช้บัตรสะสมที่มีการแก้ไขข้อมูลสูญหายโดยการเก็บแต้มสะสมของผู้ใช้ไว้ที่ระบบฐานข้อมูล ทั้งยังมีวิธีการเก็บแต้มสะสมแบบกำหนดแต้มหรือราคาสินค้า เพื่อเพิ่มความหลากหลายของวิธีการสะสม ซึ่งช่วยให้ผู้ประกอบการสามารถสร้างวิธีการเก็บสะสมแต้มได้ตามที่ตัวเองต้องการ

ผลลัพธ์การทำงานของระบบสะสมแต้มได้สำเร็จไปตามวัตถุประสงค์ที่ตั้งไว้ โดยระบบสะสมแต้มนั้นสามารถป้องกันการลืมหรือบัตรเสียหายโดยเก็บแต้มสะสมไว้ที่ฐานข้อมูล และผู้ประกอบการยังสามารถสร้างวิธีการเก็บข้อมูลได้ตามแบบที่ตัวเองต้องการ นอกจากนี้ยังสามมารถนำข้อมูลที่เก็บบันทึกไว้มาวิเคราะห์ถึงพฤติกรรมการเก็บสะสมแต้มของสมาชิกได้ เพื่อนำไปสร้างโปรโมชันที่น่าสนใจให้แก่สมาชิกได้

|  |  |
| --- | --- |
| **Project Title** | Collecting points system |
| **Student** | Nitikarn Pechwan |
| **Student ID** | 60160040 |
| **Advisor** | Prajaks Jitngernmadan, PhD. |
| **Level of Study** | Bachelor of Science in Information Technology |
| **Faculty** | Faculty of Informatics, Burapha University |
| **Year** | 2020 |

# Abstract

Loss of reward points caused by forgetting or damaged reward card, It is a problem of having to start accumulating points again from the beginning. Which starting to accumulate points again is a frequent problem, Existing point accumulation system the choice of collection methods is limited, Therefore it may not be an option that entrepreneurs requirement.

Therefore, developers have studied and developed a system for collecting points instead of using reward cards, that prevents data loss by storing the user's accumulated points in the database, there is also a way to collect reward points by specifying points or specifying the price, to increase the variety of collection methods. This allows entrepreneurs to create a way to collect points as they requirement.

The result of the collecting points system has succeeded in accordance with the objectives set. The collecting points system can fix forgotten or damaged cards by storing points in the database. And entrepreneurs can also create a method for collecting data as they want. In addition, the recorded data can be used to analyze the behavior of members collecting points. To create interesting promotions for members

# กิตติกรรมประกาศ

โครงงาน ระบบสะสมแต้ม สามารถสำเร็จลุล่วงไปได้ด้วยดี เนื่องด้วยคำแนะนำและการให้คำปรึกษาจากที่เกี่ยวข้องดังนี้

ขอกราบขอบพระคุณ อาจารย์ ดร. ประจักษ์ จิตเงินมะดัน อาจารย์ที่ปรึกษาการทำโครงงานในครั้งนี้ ที่ช่วยให้ความรู้และช่วยสอนการทำงานที่นำมาพัฒนา ให้คำแนะนำแก้ไขข้อบกพร่องต่าง ๆ และข้อเสนอ แนะ ในการทำโครงงานเล่มนี้ ด้วยความเอาใจใส่อย่างดียิ่งจนทำให้โครงงานสำเร็จลุล่วงไปด้วยดี

ขอกราบขอบพระคุณอาจารย์ทุกท่านในคณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา ที่ให้ความรู้และคำปรึกษาที่ดีโดยตลอด ทำให้สามารถนำความรู้ต่าง ๆ มาประยุกต์ให้เข้ากับโครงงาน

สุดท้ายขอกราบขอบพระคุณบิดา มารดา รุ่นพี่ และเพื่อนทุกคนสำหรับคำแนะนำ ความช่วยเหลือในการปฏิบัติงานและกำลังใจที่มีให้เสมอมา

ผู้จัดทำหวังเป็นอย่างยิ่งว่าโครงงานเล่มนี้จะเป็นประโยชน์ต่อผู้ที่สนใจและผู้ที่จะนำไปพัฒนาต่อเป็นอย่างมาก และใคร่ขอขอบคุณผู้ที่มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูลและเป็นที่ปรึกษาในการทำโครงงานเล่มนี้จนเสร็จสมบูรณ์ ซึ่งผู้จัดทำขอขอบพระคุณเป็นอย่างยิ่งไว้ ณ ที่นี่

นิติกานต์ เพ็ชแหวน

มีนาคม 2564

# สารบัญ

[บทคัดย่อ ก](#_Toc75876841)

[Abstract ข](#_Toc75876842)

[กิตติกรรมประกาศ ค](#_Toc75876843)

[สารบัญ ง](#_Toc75876844)

[สารบัญรูปภาพ ฉ](#_Toc75876845)

[สารบัญตาราง ช](#_Toc75876846)

[บทที่ 1 บทนำ 1](#_Toc75876847)

[1.1 ที่มาของโครงงาน 1](#_Toc75876848)

[1.2 วัตถุประสงค์ของโครงงาน 1](#_Toc75876849)

[1.3 ขอบเขตของโครงงาน 2](#_Toc75876850)

[1.4 เครื่องมือที่ใช้ในการพัฒนา 2](#_Toc75876851)

[1.5 ขั้นตอนการดำเนินงาน 3](#_Toc75876852)

[1.6 แผนการดำเนินงาน 4](#_Toc75876853)

[1.7 ประโยชน์ที่คาดว่าจะได้รับ 4](#_Toc75876854)

[บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 5](#_Toc75876855)

[2.1 ทฤษฎีที่เกี่ยวข้อง 5](#_Toc75876856)

[1. เทคโนโลยี AI (Artificial Intelligence: AI) [1] 5](#_Toc75876857)

[2. Adafruit PCA9685 Python Library [2] 5](#_Toc75876858)

[2.2 งานที่เกี่ยวข้อง 7](#_Toc75876859)

[บทที่ 3 วิธีการดำเนินโครงงาน 9](#_Toc75876860)

[3.1 ศึกษาปัญหาและความเป็นไปได้ 9](#_Toc75876861)

[3.2 วิเคราะห์และออกแบบแบบจำลอง 9](#_Toc75876862)

[3.2.1 Workflow Diagram 10](#_Toc75876863)

[3.2.2 Use Case Diagram 11](#_Toc75876864)

[3.2.3 Use Case Description 12](#_Toc75876865)

[3.2.4 Activity Diagram 15](#_Toc75876866)

[บทที่ 4 ผลการดำเนินโครงงาน 21](#_Toc75876867)

[4.1 การพัฒนาระบบ 21](#_Toc75876868)

[4.2 การทดสอบและเปรียบเทียบ 23](#_Toc75876869)

[บทที่ 5 สรุปผลการดำเนินโครงงาน 24](#_Toc75876870)

[5.1 สรุปผลการดำเนินงาน 24](#_Toc75876871)

[5.2 ปัญหา อุปสรรค และ ข้อจำกัด 24](#_Toc75876872)

[5.3 ข้อเสนอแนะและงานในอนาคต 24](#_Toc75876873)

[บรรณานุกรม 25](#_Toc75876874)

[ภาคผนวก 26](#_Toc75876875)

[ภาคผนวก ก การหมุนของ Servo Motor แต่ละส่วน 27](#_Toc75876876)

[ภาคผนวก ข การทำงานของโปรแกรม 28](#_Toc75876877)

[ภาคผนวก ค โปรแกรม 34](#_Toc75876878)

[ประวัติผู้จัดทำโครงงาน 47](#_Toc75876879)

# สารบัญรูปภาพ

[ภาพที่ 1 OpenCV 7](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681239)

[ภาพที่ 2 Dlib Library 7](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681240)

[ภาพที่ 3 PCA9685 7](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681241)

[ภาพที่ 4 Ultrasonic Sensor 8](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681242)

[ภาพที่ 5 Workflow Diagram ระบบหุ่นยนต์ป้อนอาหาร 12](#_Toc77681243)

[ภาพที่ 6 Use Case Diagram ระบบหุ่นยนต์ป้อนอาหาร 13](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681244)

[ภาพที่ 7 Activity Diagram ระบบหุ่นยนต์ป้อนอาหาร 17](#_Toc77681245)

[ภาพที่ 8 วงจรควบคุมการทำงานของระบบ 18](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681246)

[ภาพที่ 9 รูปแบบการทำงานภาพรวมของระบบ Artificial Intelligence (AI) 19](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681247)

[ภาพที่ 10 รูปแบบการทำงานของ module servo 19](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681248)

[ภาพที่ 11 Ultrasonic Sensor คำนวณหาระยะห่าง 20](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681249)

[ภาพที่ 12 ควบคุม Servo Motor ผ่าน Adafruit PCA9685 21](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681250)

[ภาพที่ 13 ผู้ป่วยหรือผู้สูงอายุอ้าปากเพื่อรับทานอาหาร 23](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681251)

[ภาพที่ 14 รูปแขนกลป้อนอาหาร 23](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681252)

[ภาพที่ 15 รูป Ultrasonic Sensor บนอุปกรณ์แขนกลป้อนอการ 24](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681253)

[ภาพที่ 16 รูปการทำงานของแขนกลป้อนอาหาร 24](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681254)

[ภาพที่ 17 ตำแหน่งหมายเลขของ Servo Motor 29](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681255)

[ภาพที่ 18 อธิบายการหมุนของ Servo Motor หมายเลข 0 กับ หมายเลข 4 29](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681256)

[ภาพที่ 19 อธิบายการหมุนของ Servo Motor หมายเลข 1 กับ หมายเลข 2 29](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681257)

[ภาพที่ 20 อธิบายการหมุนของ Servo Motor หมายเลข 3 29](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681258)

[ภาพที่ 21 อธิบายการทำงานของกล้องตรวจระยะอ้าปากเพื่อรับทานอาหาร 30](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681259)

[ภาพที่ 22 แขนกลเซ็ตมุมก่อนทำงานเพื่อป้องกันปัญหาข้อผิดพลาด 30](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681260)

[ภาพที่ 23 แขนกลทำงานฟังก์ชันเตรียมก่อนตักอาหาร 31](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681261)

[ภาพที่ 24 แขนกลทำงานฟังก์ชันตักอาหาร 32](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681262)

[ภาพที่ 25 สาธิตการทำงานหมุนมาทางผู้รับประทานอาหาร 32](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681263)

[ภาพที่ 26 แขนกลหันทางผู้รับประทานอาหาร 33](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681264)

[ภาพที่ 27 แขนกลเพิ่มองศาจนสุดที่มุม 150 องศา 34](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681265)

[ภาพที่ 28 แขนกลเพิ่มองศาเพื่อขยับเข้าใกล้ผู้รับประทานอาหาร 34](file:///D:\Nitikarn\Flutter\collecting_points\Senior_Project.docx#_Toc77681266)

# สารบัญตาราง

[ตารางที่ 1 ขั้นตอนการดำเนินงาน 4](#_Toc73661414)

[ตารางที่ 2 Use Case Description เริ่มการทำงาน 12](#_Toc73661415)

[ตารางที่ 3 Use Case Description การทำงานของแขนกล 13](#_Toc73661416)

[ตารางที่ 4 Use Case Description จัดการข้อมูลพนักงาน 14](#_Toc73661417)

[ตารางที่ 5 ผลการทดสอบภาพรวม 23](#_Toc73661418)

[ตารางที่ 6 ผลการทดสอบและเปรียบเทียบการทำงาน 23](#_Toc73661419)

[ตารางที่ 7 องศาของ Servo Motor ของแต่ละส่วนที่กำหนดก่อนทำงาน 29](#_Toc73661420)

[ตารางที่ 8 องศาของ Servo Motor ของฟังก์ชันเตรียมก่อนตักอาหาร 29](#_Toc73661421)

[ตารางที่ 9 องศาของ Servo Motor ของฟังก์ชันเตรียมก่อนตักอาหาร 30](#_Toc73661422)

[ตารางที่ 10 องศาของ Servo Motor ของฟังก์ชันตักอาหารหันมาทางผู้รับประทานอาหาร 31](#_Toc73661423)

[ตารางที่ 11 องศาของ Servo Motor ของฟังก์ชันแขนกลหันทางผู้รับประทานอาหาร 31](#_Toc73661424)

[ตารางที่ 12 องศาของ Servo Motor ของภาพที่ 27 32](#_Toc73661425)

[ตารางที่ 13 องศาของ Servo Motor ของภาพที่ 28 33](#_Toc73661426)

[ตารางที่ 14 Start.py 34](#_Toc73661427)

[ตารางที่ 15 Servo.py 36](#_Toc73661428)

[ตารางที่ 16 Ultrasonict.py 43](#_Toc73661429)

# บทที่ 1 บทนำ

ในส่วนนี้เป็นการอธิบายรายละเอียดและความเป็นมาของโครงงาน ซึ่งเป็นโครงงานเรื่อง ระบบสะสมแต้ม โดยมีหัวข้อสำคัญตามลำดับต่อไปนี้

* 1. ที่มาของโครงงาน
  2. วัตถุประสงค์ของโครงงาน
  3. ขอบเขตของโครงงาน
  4. เครื่องมือที่ใช้ในการพัฒนา
  5. ขั้นตอนในการดำเนินโครงงาน
  6. แผนการดำเนินโครงงาน
  7. ประโยชน์ที่คาดว่าจะได้รับ

## 1.1 ที่มาของโครงงาน

ปัจจุบัน การสะสมแต้มของผู้ประกอบการส่วนใหญ่จะใช้วิธีการสะสมแต้มเป็นบัตรสะสมแต้ม ซึ่งได้เกิดปัญหาด้านการเก็บรักษา หากบัตรสะสมแต้มเปียกน้ำ สูญหาย หรือข้อมูลแต้มสะสมเลือนรางจนไม่สามารถยืนยันแต้มสะสมได้ ทำให้ผู้สะสมจะต้องเริ่มต้นสะสมแต้มใหม่อีกครั้ง หรือผู้ประกอบการที่ใช้ระบบสะสมแต้มที่มีอยู่ วิธีการสะสมแต้มนั้นมีตัวเลือกอย่างจำกัด ซึ่งอาจไม่ตรงความต้องการของผู้ประกอบการ

จากการสำรวจประชากรเทศบาลเมืองแสนสุขพบว่า ผู้ที่ใช้งานโทรศัพท์มือถือมี 82.4% ที่ใช้งานระบบปฏิบัติการ Android

ผู้พัฒนาได้เห็นถึงปัญหาตรงส่วนนี้จึงทำการพัฒนาระบบสะสมแต้มที่แก้ไขปัญหาข้างตน โดยให้ผู้สะสมแต้มใช้ระบบสะสมแต้มแทนการใช้บัตรสะสม ซึ่งเป็นการจัดเก็บแต้มสะสมไว้ที่ระบบฐานข้อมูล เพื่อที่ป้องกันการสูญหายของข้อมูลแต้มสะสม ทั้งยังเลือกวิธีการเก็บแต้มสะสมแบบกำหนดแต้มหรือจะเป็นการกำหนดราคาสินค้าก็ได้ เพื่อเพิ่มความหลากหลายของวิธีการสะสมแต้ม ซึ่งช่วยให้ผู้ประกอบการสามารถสร้างวิธีการเก็บสะสมแต้มได้ตามที่ต้องการ สามารถตรวจสอบแต้มได้ทั้งผู้ประกอบการและสมาชิก ดังนั้นผู้พัฒนาจึงทำการพัฒนาโครงการตัวนี้ขึ้น เพื่อแก้ปัญหาในการสะสมแต้มเหล่านั้น

อ้างอิงจากการสำรวจมาแล้ว, ระบบที่มีอยู่, วิธีแก้ปัญหา โดยเทคโนโลยีอะไร

## 1.2 วัตถุประสงค์ของโครงงาน

1. พัฒนาระบบสะสมแต้ม
2. ศึกษาพฤติกรรม การจัดโปรโมชันของผู้ประกอบการ
3. ศึกษาพฤติกรรม การสะสมแต้มและแลกรางวัลของลูกค้าหรือสมาชิก
4. ศึกษาและเรียนรู้วิธีการใช้การสร้างแอปพลิเคชัน
5. ศึกษาและเรียนรู้วิธีการใช้การระบบฐานข้อมูล
6. ผู้ประกอบการสามารถเลือกจัดโปรโมชันได้ตามต้องการ

## 1.3 ขอบเขตของโครงงาน

การพัฒนาระบบจะมุ่งเน้นไปที่การเรียนรู้การใช้งานโปแกรมสร้างแอปพลิเคชัน และวิธีการสร้างระบบฐานข้อมูล โดยกลุ่มเป้าหมายที่เราสนใจจะมี 2 กลุ่ม คือ

1. ผู้ประกอบการ
   * เข้าสู้ระบบ
   * สมัครใช้งานระบบ
   * สามารถสร้างองค์กร
   * เพิ่มสมาชิกเข้าองค์กร
   * ลงทะเบียนผู้ใช้ให้สมาชิก
   * แก้ไขแต้มของสมาชิก
   * สร้างรางวัล
   * สร้างวิธีการรับแต้ม
   * สร้างชื่อหน่วยของแต้มสะสม
   * แลกรางวัลให้สมาชิก
   * ดูรายชื่อของสมาชิก
   * ดูคะแนนสะสมของสมาชิก
2. สมาชิก
   * เข้าสู้ระบบ
   * สมัครใช้งานระบบ
   * ตรวจสอบคะแนนสะสม
   * ตรวจสอบรางวัล

1.4 เครื่องมือที่ใช้ในการพัฒนา

1. เครื่องมือด้าน Hardware
   1. คอมพิวเตอร์
   2. โทรศัพท์มือถือ android
2. เครื่องมือด้าน Software
3. โปรแกรมสำหรับจัดทำเอกสาร ได้แก่ Microsoft Word 2016 \*
4. ระบบปฏิบัติการ ได้แก่ Windows 10
5. โปรแกรมที่เขียนโค้ด (IDE) Android Studio
6. ภาษาโปแกรม Dart
7. เฟรมเวิร์ก Flutter
8. โปรแกรมเว็บเบราว์เซอร์ Google Chrome \*

## 1.5 ขั้นตอนการดำเนินงาน

เก็บ requirement ศึกษาข้อมูล ศึกษาเทคโนโลยี วางแผน สร้าง ทดสอบ เขียนเล่ม

1. สำรวจความต้องการ
2. ศึกษาข้อมูล

* ศึกษาหลักการและทฤษฎี

1. ศึกษาเทคโนโลยี

* เรียนรู้การใช้งานเทคโนโลยี

1. วางแผนการดำเนินงาน
   * กำหนดปัญหา
   * ออกแบบ Use Case Diagram
   * ออกแบบโครงสร้างของซอฟต์แวร์
   * ออกแบบส่วนติดต่อผู้ใช้ของซอฟต์แวร์
2. สร้างระบบ
3. ทดสอบระบบ
4. ทำเล่มโครงงาน
   * จัดทำเล่มโครงงานบทที่ 1 - 5
5. วางแผนการดำเนินงาน
   1. กำหนดปัญหา
   2. ศึกษาข้อมูลในการทำระบบ
6. ศึกษาวิเคราะห์ระบบงาน
   1. วิเคราะห์ความต้องการของระบบงาน
   2. จัดลำดับความสำคัญของปัญหา
   3. รวบรวมข้อมูล
7. การออกแบบระบบ
   1. ออกแบบรูปแบบการทำงานโปรแกรมของระบบ
   2. ออกแบบ Workflow Diagram
   3. ออกแบบโครงสร้างของซอฟต์แวร์
8. จัดทำโปรแกรม
   1. พัฒนาปรับปรุงตัวต้นแบบหุ่นยนต์แขนกลป้อนอาหารโดยใช้ AI
   2. เขียนโปรแกรมในแต่ละส่วนของการทำงาน
   3. ทดสอบโปรแกรมและแก้ไขเพิ่มเติมโปรแกรมให้สมบูรณ์
9. การทำเล่มโครงงาน
   * จัดทำเล่มโครงงานตั้งแต่บทที่ 1-5

## 1.6 แผนการดำเนินงาน

ให้อธิบายถึงการทำงานในแต่ละขั้นตอน โดยมีการอ้างอิงไปที่ตาราง ซึ่งชื่อของตารางจะต้องอยู่มุมบนด้านซ้ายมือเท่านั้น เช่น ตารางที่ 1 โดยใช้ใช้การใส่ Caption อัตโนมัติในการใส่ชื่อตาราง

ตารางที่ 1 ขั้นตอนการดำเนินงาน

| ขั้นตอนการดำเนินงาน | ระยะเวลาในการดำเนินงาน | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| เดือนที่ 1 | เดือนที่ 2 | เดือนที่ 3 | เดือนที่ 4 | เดือนที่ 5 |
| 1. สำรวจความต้องการ |  |  |  |  |  |
| 1. ศึกษาข้อมูล |  |  |  |  |  |
| 1. ศึกษาเทคโนโลยี |  |  |  |  |  |
| 1. ออกแบบ Use Case Diagram |  |  |  |  |  |
| 1. ออกแบบโครงสร้างของซอฟต์แวร์ |  |  |  |  |  |
| 1. ออกแบบส่วนติดต่อผู้ใช้ของซอฟต์แวร์ |  |  |  |  |  |
| 1. สร้างระบบ |  |  |  |  |  |
| 1. ทดสอบและแก้ไขโปรแกรมให้เสร็จสมบูรณ์ |  |  |  |  |  |
| 1. การจัดทำเอกสาร |  |  |  |  |  |
| 1. นำเสนอโครงงาน |  |  |  |  |  |

## 1.7 ประโยชน์ที่คาดว่าจะได้รับ

1. นำไปใช้แทนการสะสมแบบเดิมที่มีอยู่ เช่น การเช็คชื่อ บัญชีออมเงิน
2. ลดขยะจากการนำวัสดุมาทำบัตรสะสม
3. ผู้ประการสามารถสร้างโปรโมชันที่น่าดึงดูด
4. สมาชิกมีความสะดวกสบายในการสะสมแต้ม

# บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการทำโครงงาน และงานวิจัยที่ได้ศึกษาเพื่อช่วยในการทำโครงงาน ซึ่งผู้จัดทำเป็นต้องศึกษา วิเคราะห์ และเก็บรวบรวมข้อมูลความรู้ต่าง ๆ เพื่อใช้ในการออกแบบระบบ เพื่อใช้เป็นแนวทางในการจัดทำโครงงาน โดยมีรายละเอียด ดังต่อไปนี้

## ทฤษฎีที่เกี่ยวข้อง

UX UI Design,

ส่วนผสมการตลาด 4P



ภาพที่ 1 4P

### 1. ส่วนผสมการตลาด 4P

ส่วนผสมการตลาด 4P (Marketing Mix) เป็นทฤษฎีที่ทางการตลาดที่ได้รับความนิยมและเป็นพื้นฐานที่สุด โดยแบ่งการวิเคราะห์องค์ประกอบเป็น 4 ส่วน ได้แก่

1. **ผลิตภัณฑ์** (Product) สินค้าที่ผลิตออกมาต้องมีคุณภาพ สามารถใช้งานตามมารฐาน พร้อมกับต้องวิเคราะห์ว่ากลุ่มลูกค้าที่จะเข้ามาใช้ผลิตภัณฑ์หรือบริการเป็นคนกลุ่มไหน ชื่นชอบและสนใจผลิตภัณฑ์แบบใด พร้อมกับพยายามสื่อสารจุดขาย ข้อดี และประโยชน์ของผลิตภัณฑ์ ที่กลุ่มผู้บริโภคจะได้รับหรือผลิตภัณฑ์นี้จะสามารถเข้าไปแก้ปัญหาให้กับผู้บริโภคได้ ออกไปให้ตรงจุดที่สุดนั่นเอง
2. **ราคา** (Price) การตั้งราคา ซึ่งเป็นหัวใจสำคัญสำหรับการทำการค้าเช่นกัน การตั้งราคาให้เหมาะสมกับสินค้าและกลุ่มผู้บริโภคก็นับว่าเป็นเรื่องที่ต้องใช้การวิเคราะห์อย่างเหมาะสม
3. **ช่องทางจัดจำหน่าย** (Place) วางขายที่ไหน วางขายผ่านหน้าร้านเพื่อให้ลูกค้าเขามาเลือกและสัมผัสสินค้าจริงได้ก่อนตัดสินใจซื้อ ฝากขายผ่านหน้าร้านตัวแทนเพื่อกระจายช่องทางขายสินค้าให้กว้างขึ้น หรือขายผ่านแพลตฟอร์มออนไลน์ พร้อมกับส่งสินค้าไปยังผู้บริโภคผ่านบริษัทที่ให้บริการด้านขนส่งที่ได้มาตรฐาน
4. **การส่งเสริมการขาย** (Promotion) ไม่ใช่การลดราคาสินค้าเพียงอย่างเดียว แต่รวมถึงการจัดกิจกรรมทางการตลาดที่ช่วยส่งเสริมการขาย เช่น การทำใบปลิว ทำโฆษณาสินค้า การประชาสัมพันธ์ผ่านช่องทางต่าง ๆ เพื่อเพิ่มโอกาสประชาสัมพันธ์สินค้าให้เป็นที่รู้จักในวงกว้างขึ้นและดึงดูดลูกค้าให้เข้ามาเลือกซื้อสินค้าภายในร้านของคุณได้อย่างเห็นผล

<https://fillgoods.co/online-biz/shop-orders-build-business-with-4p-make-more-sale/>

### 2. OpenCV (Open source Computer Vision: OpenCV) [2]

OpenCV เป็นไลบรารีฟังก์ชันการเขียนโปรแกรม (Library of Programming Functions) โดย ส่วนใหญ่จะเน้นไปที่การแสดงผลด้วยคอมพิวเตอร์แบบเรียลไทม์ (Real-Time Computer Vision) OpenCV เป็นไลบรารีแบบข้ามแพลตฟอร์ม (Cross-Platform) และใช้งานได้ฟรีภายใต้ลิขสิทธิ์ของ BSD License (Berkeley Software Distribution: BSD License)

OpenCV ยังสนับสนุนเฟรมเวิร์กการเรียนรู้เชิงลึก (Deep Learning Frameworks) ได้แก่ TensorFlow, Torch/PyTorch และ Caffe

ภาพที่ 2 OpenCV

### 3. Dlib Library [3]

Dlib คือ C++ library ซึ่งมี Machine learning algorithms และ เครื่องมือต่าง ๆ ในการสร้าง โปรแกรมที่ซับซ้อน อย่างระบบการจดจำใบหน้าที่มีโมดูลไว้ให้ใช้เช่นเดียวกัน

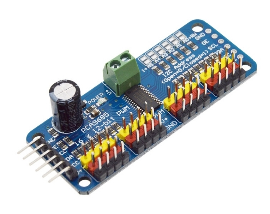
Dlib เป็น Library ที่เปิดให้ผู้พัฒนาโปรแกรมได้ใช้โดยไม่เสียค่าใช้จ่าย (Open-Source Licensing)



ภาพที่ 3 Dlib Library

### 4. Adafruit PCA9685 [4]

Adafruit PCA9685 เป็นไลบรารีฟังก์ชันการเขียนโปรแรกม (Library of Programming Functions) โดย ใช้ควบคุมการทำงานของ Servo Motor พร้อมกันหลายตัว โดยที่ใช้ร่วมกับอุปกรณ์ PCA9685 จะมีช่องสำหรับการต่อ Servo Motor ได้ถึง 16 ช่องในการควบคุมการทำงาน



ภาพที่ 4 PCA9685

### 5. ทฤษฎีคลื่นเสียง [5]

 การนำหลักทฤษฎีทางวิทยาศาสตร์ คลื่นเสียง มาประยุกต์ใช้งานกับอุปกรณ์ Ultrasonic Sensor การทำงานโดย ส่งคลื่นความถี่เสียงย่านอัลตร้าโซนิค โดยส่งคลื่นออกมากระทบกับวัตถุ หรือ ชิ้นงานที่ตรวจจับ แล้ว สะท้อนคลื่นเสียงกลับมาที่ตัวเซนเซอร์ ทำให้สามารถเช็คระยะตรวจจับของระยะห่าง

ภาพที่ 5 Ultrasonic Sensor

## 2.2 งานที่เกี่ยวข้อง

1. จากการศึกษาวิจัย Self-feeding apparatus [6] เป็นอุปกรณ์ให้อาหารด้วยตนเอง สำหรับผู้พิการ และผู้สูงอายุซึ่งประกอบด้วยตัวฐานที่ยึดสำหรับภาชนะบรรจุอาหารและเชื่อมต่อกับ ฐานแบบ Pivotally การดักอาหารจากภาชนะใช้ระบบสำหรับเคลื่อนย้ายสิ่งของที่ตักออกจาก ตำแหน่งที่ตักอาหารจากภาชนะและตำแหน่งการป้อนอาหารที่ความสูงที่เลือกจากฐานที่สอดคล้องกับ ปากของผู้ใช้ ระบบนี้รวมถึงส่วนแขนแรกที่ยื่นออกมาจากฐาน และ แขนที่สองโดยจัดให้อยู่ในตำแหน่ง อุปกรณ์นี้มีระบบการเขียนโปรแรกม และ การควบคุมที่มีการทำงานด้วยบอร์ดไมโครคอนโทรลเลอร์

ข้อดีและข้อเสีย

ข้อดี

* + - อุปกรณ์สำหรับป้อนอาหารด้วยตนเองถูกออกแบบมาให้ใช้จาน และ ช้อนส้อมได้ทุกแบบ
    - โปรแกรมการเคลื่อนไหวแบบอัตโนมัติหรือแบบ servo เขียนโปรแกรมมาไว้แล้ว

ข้อเสีย

* + - การใช้งานตัวเครื่องดื่มยังไม่เหมาะสมกับการใช้งาน
    - ตัวเครื่องยังไม่สามารถขยับแขนไปหาผู้ใช้งานได้พอดี

2.จากศึกษาโครงงาน หุ่นยนต์แขนกลป้อนอาหารอัตโนมัติโดยใช้ Artificial Intelligence (iFeedingBot) [7] ที่นำมาต่อยอดนั้น หุ่นยนต์แขนกลป้อนอาหาร สามารถตรวจจับรับรู้ ผู้ใช้งานมีความต้องการรับประทานอาหาร โดยการนำกล้องมาใช้ในการจับใบหน้าการอ้าปาก เมื่อระยะของริมฝีปากบน กับ ริมฝีปากล่างห่างกัน ถึงระยะที่กำหนด หุ่นยนต์แขนกลจึงทำงานตักอาหาร ให้กับผู้ใช้งาน ส่วนโปรแกรมที่ควบคุมข้อต่อ หุ่นยนต์แขนกลป้อมอาหาร เป็นรูปแบบกำหนดองศาที่ตายตัวไว้ ซึ่งยังไม่สามารถควบคุมมุมของแขนกลให้สัมพันธ์ กับ ระยะห่างของ ผู้ใช้งานได้ ไม่ดีเท่า ที่ควร

ข้อดีและข้อเสีย

ข้อดี

* + - อุปกรณ์มีกล้องเข้ามาช่วยในการทำงาน ตรวจจับภาพ ริมฝีปาก
    - โปรแกรมสามารถตรวจจับรับรู้ ระยะการอ้าปาก เพื่อเริ่มทำงานของหุ่นยนต์แขนกล ทำงานตักอาหาร

ข้อเสีย

* + - โปรแกรมควบคุมข้อต่อ หุ่นยนต์แขนกลป้อมอาหาร ยังเป็นรูปแบบกำหนดองศาที่ตายตัว ซึ่งไม่สามารถควบคุมมุมของแขนกลให้สัมพันธ์ กับระยะห่างของผู้ใช้งานได้ ไม่ดีเท่า ที่ควร
    - Servo Motor ของแขนกลมีความเร็วในการทำงานที่ค่อนข้างช้า ทำให้ใช้เวลานานในการตักอาหาร มาให้ผู้ใช้งาน

# บทที่ 3 วิธีการดำเนินโครงงาน

ในส่วนของวิธีการดำเนินโครงงาน ได้มีการวางแผนศึกษากระบวนการวิธีการ เพื่อดำเนินการให้เป็นไปอย่างถูกต้องและมีประสิทธิภาพสูงที่สุด ซึ่งมีรายละเอียด ดังนี้

* 1. ศึกษาปัญหาและความเป็นไปได้
  2. วิเคราะห์และออกแบบแบบจำลอง
  3. การออกแบบเว็บไซด์และการใช้งาน
  4. หลักการวัดระยะห่างและทำมุมองศาของแขนกล

## 3.1 ศึกษาปัญหาและความเป็นไปได้

จากการศึกษาพบว่าใน ผู้ป่วยที่พิการทางแขนนั้น มีความลำบากในการใช้ชีวิตประจำวันมากกว่าคนธรรมดาทั่วไป โครงงานนี้จึงเจาะจงลงไปในเรื่องของการช่วยเหลือในการรับประทานอาหาร ซึ่งเทคโนโลยีในสมัยนี้ช่วยได้คือ แขนกลป้อนอาหาร และ รถเข็นสำหรับผู้พิการ

ซึ่งในปัจจุบันเทคโนโลยีทั้ง 2 ตัวนี้ มีราคาที่ค่อนข้างสูง เนื่องจากมีฟังก์ชันการทำงานที่หลากหลาย ที่ตอบโจทย์ได้เกือบทุกกรณี ดังนนั้นเราจึงทำการสร้างแขนกลขึ้นมาโดยคำนึงถึงค่าใช้จ่ายที่ไม่แพง แต่ก็ยังสามารถทำการป้อนอาหารได้ตรงตามวัตถุประสงค์เช่นเดิม แม้ว่าฟังก์ชันการทำงานจะไม่ตอบโจทย์ได้เกือบทุกกรณีก็ตาม แต่ก็สามารถเข้าถึงคนที่มีรายได้น้อยได้

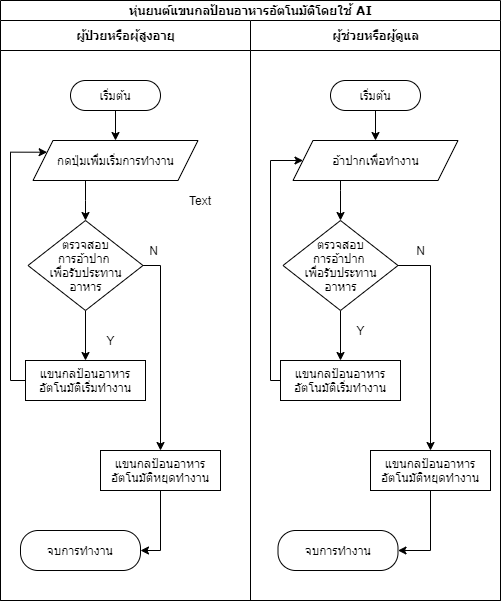
ในอนาคตอาจจะมีการพัฒนาแขนกล หรือรถเข็นสำหรับผู้พิการได้ดีกว่านี้ หรือไม่ก็มีสิ่งที่ช่วยอำนวยความสะดวกได้มากกว่านี้ เช่น หุ่นยนต์ผู้ดูแล หรือ แขนกลที่สั่งการด้วยสมองโดยตรงก็เป็นได้

## 3.2 วิเคราะห์และออกแบบแบบจำลอง

ในส่วนการออกแบบนั้น ขึ้นอยู่กับงานที่นิสิตเลือกทำ โดยจะยกตัวอย่างของการพัฒนาระบบ ดังนี้

### 3.2.1 Workflow Diagram

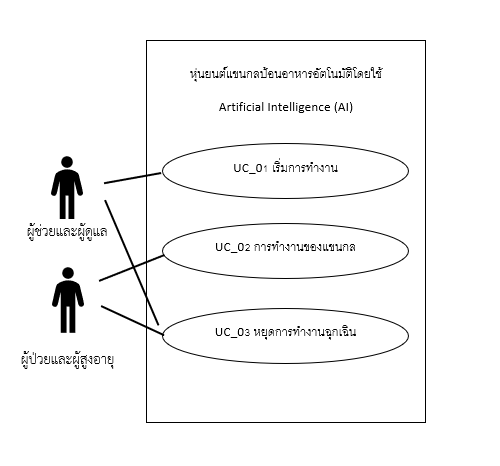
ผังงาน (Flowchart) คือ รูปภาพ (Image) หรือสัญลักษณ์ (Symbol) ที่ใช้เขียนแทนขั้นตอน คำอธิบาย ข้อความ หรือคำพูด ที่ใช้ในอัลกอริทึม (Algorithm)



ภาพที่ 6 Workflow Diagram ระบบหุ่นยนต์ป้อนอาหาร

### 3.2.2 Use Case Diagram

Use Case Diagram คือแผนภาพที่แสดงการทำงานของผู้ใช้ระบบ (User) และความสัมพันธ์กับระบบย่อย (Sub systems) ภายในระบบใหญ่ ในการเขียน Use Case Diagram ผู้ใช้ระบบ (User) จะถูกกำหนดว่าให้เป็น Actor และ ระบบย่อย (Sub systems) คือ Use Case จุดประสงค์หลักของการเขียน Use Case Diagram เพื่อเล่าเรื่องราวทั้งหมดของระบบว่ามีการทำงานอะไรบ้าง เป็นการดึง Requirement หรือเรื่องราวต่าง ๆ ของระบบจากผู้ใช้งาน ซึ่งถือว่าเป็นจุดเริ่มต้นในการวิเคราะห์และออกแบบระบบ สัญลักษณ์ที่ใช้ใน Use Case Diagram จะใช้สัญลักษณ์รูปคนแทน Actor ใช้สัญลักษณ์วงรีแทน Use Case และใช้เส้นตรงในการเชื่อม Actor กับ Use Case เพื่อแสดงการใช้งานของ Use Case ของ Actor นอกจากนั้น Use Case ทุก ๆ ตัวจะต้องอยู่ภายในสี่เหลี่ยมเดียวกันซึ่งมีชื่อของระบบระบุอยู่ด้วย



ภาพที่ 7 Use Case Diagram ระบบหุ่นยนต์ป้อนอาหาร

### 3.2.3 Use Case Description

Use Case Description คือ การอธิบาย functional requirement แบบละเอียด แนะนำรูปแบบการอธิบายแบบ ‘Use Case Specification’ หรือเรียกว่า ‘Use Case Description’ เนื่องจากมีการแบ่งประเด็นเป็นหัวข้อ ๆ ชัดเจน ทำให้เราไม่ต้องการอธิบายเป็นข้อความยาวเหยียดเป็นเรียงความ ที่อ่านและจับประเด็นยาก

ตารางที่ 2 Use Case Description เริ่มการทำงาน

|  |  |  |
| --- | --- | --- |
| Use Case ID : | UC\_01 | |
| Use Case Name : | เริ่มการทำงาน | |
| Brief Description : | เริ่มโปรแกรมเพื่อเริ่มการทำงานทั้งหมด | |
| Triggering Event : | เมื่อต้องการใช้งานหุ่นยนต์แขนกลป้อนอาหารอัตโนมัติ | |
| Actors : | ผู้ช่วย/ผู้ดูแลผู้ป่วยและผู้สูงอายุ | |
| Related Use Case : | ป้อนอาหารอัตโนมัติโดยใช้ Artificial Intelligence (AI) | |
| Preconditions : | ต้องมีผู้ช่วย/ผู้ดูแลผู้ป่วยและผู้สูงอายุ คอยดูแลในเบื้องต้น | |
| Postconditions : | ผู้ช่วยหรือผู้ดูแลต้องคอยดูแลผู้ป่วยและผู้สูงอายุอย่างใกล้ชิด | |
| Flow of Events | Actor | System |
|  | 1. เปิดใช้งานเครื่อง | 2. ระบบทำการเตรียมเครื่องและประมวลผลโปรแกรม  3. ระบบตรวจสอบการตรวจจับรูปแบบปาก |
| Exceptions : | - หากไม่กดปุ่มเริ่มการทำงาน ระบบก็จะไม่ทำงานตามคำสั่งทั้งหมด | |

ตารางที่ 3 Use Case Description การทำงานของแขนกล

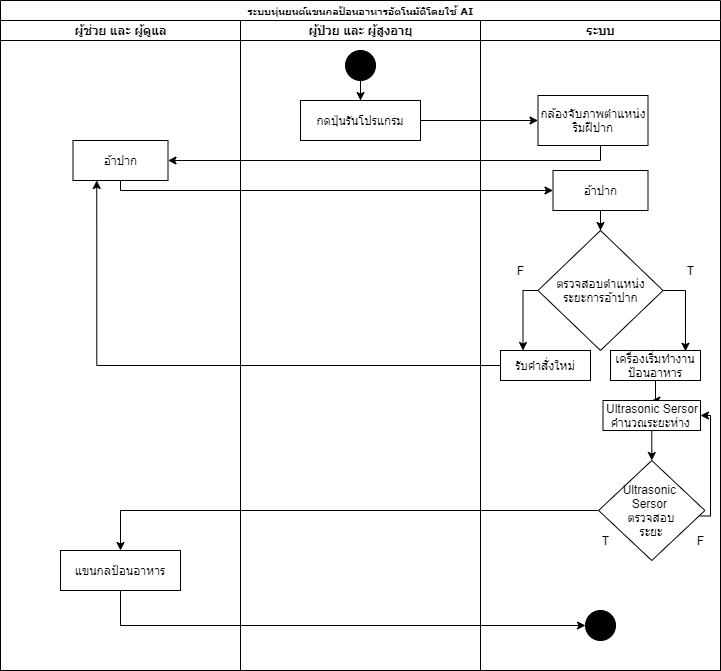
|  |  |  |
| --- | --- | --- |
| Use Case ID : | UC\_02 | |
| Use Case Name : | การทำงานของแขนกล | |
| Brief Description : | เพื่อตรวจสอบว่าผู้ป่วยหรือผู้สูงอายุ กำลังอ้าปาก | |
| Triggering Event : | ผู้ใช้งานกำลังอ้าปาก | |
| Actors : | ผู้ใช้งาน/ผู้ป่วย/ผู้สูงอายุ | |
| Related Use Case : | การทำงานของแขนกล | |
| Preconditions : | สามารถทำได้เปิดโปรแกรมเริ่มการใช้งาน | |
| Postconditions : | ผู้ช่วยหรือผู้ดูแลต้องคอยดูแลผู้ป่วยและผู้สูงอายุอย่างใกล้ชิด | |
| Flow of Events | Actor | System |
|  | 2. ผู้ใช้งานอ้าปากเพื่อเตรียมรับประทานอาหาร  7. ผู้ใช้งานรับประทานอาหารที่ตักมา | 1. ระบบเปิดกล้องตรวจจับการอ้าของริมฝีปากของผู้ใช้งาน  3. แขนกลเริ่มทำงานตักอาหาร  4. แขนกลหมุนหันมาทางผู้ป่วย  5. ระบบจะใช้ Ultrasonic Sensor เพื่อวัดระยะห่างกับผู้ใช้งาน  6. แขนกลทำมุมองศา สัมพันธ์กับระยะห่างที่คำนวณ  8. แขนกลหมุนกลับไปยังตำแหน่งเดิม เพื่อรอตรวจจับอ้าของริมฝีปากใหม่ |
| Exceptions : | - หากไม่ได้เริ่มโปรแกรม ระบบก็จะไม่ทำงานตามคำสั่งทั้งหมด | |

ตารางที่ 4 Use Case Description จัดการข้อมูลพนักงาน

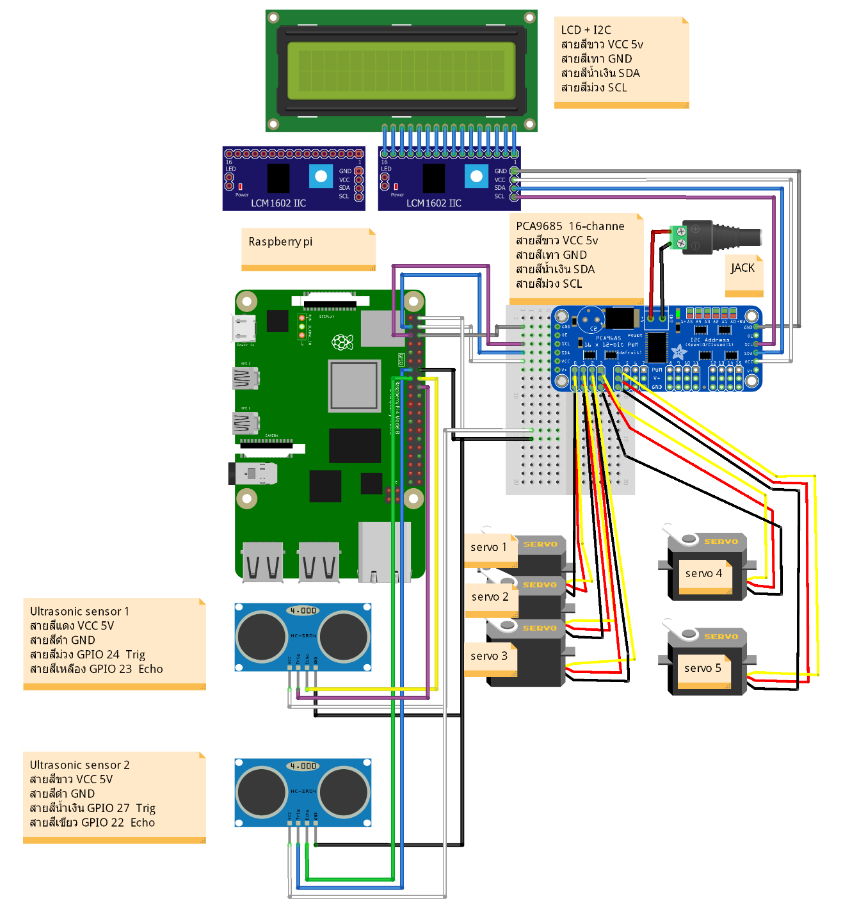
|  |  |  |
| --- | --- | --- |
| Use Case ID : | UC\_03 | |
| Use Case Name : | หยุดการทำงานฉุกเฉิน | |
| Brief Description : | เพื่อช่วยเหลือผู้ป่วยและผู้สูงอายุในการรับประทานอาหารเมื่อเครื่องมีปัญหาหรือทำงานผิดปกติ | |
| Triggering Event : | เมื่อต้องการหยุดการทำงานระบบฉุกเฉินเมื่อเกิดเหตุระบบขัดข้อง | |
| Actors : | ผู้ช่วย/ผู้ดูแลผู้ป่วยและผู้สูงอายุ | |
| Related Use Case : | ป้อนอาหารอัตโนมัติโดยใช้ Artificial Intelligence (AI) | |
| Preconditions : | ต้องมีผู้ช่วย/ผู้ดูแลผู้ป่วยและผู้สูงอายุ คอยดูแลในเบื้องต้น | |
| Postconditions : | ผู้ช่วยหรือผู้ดูแลต้องคอยดูแลผู้ป่วยและผู้สูงอายุอย่างใกล้ชิด | |
| Flow of Events | Actor | System |
|  | 1. กดปุ่มหยุดการทำงานฉุกเฉิน | 2. ระบบตัดการทำงานของระบบทั้งหมด  3. ระบบหยุดการสั่งงานทั้งหมด |
| Exceptions : | - เมื่อกดปุ่มหยุดการทำงานฉุกเฉินแล้วให้ทำการตรวจเช็คเครื่อง แล้วควร Reset เครื่องใหม่ก่อนเริ่มใช้งานต่อ | |

### 3.2.4 Activity Diagram

Activity Diagram หรือแผนภาพกิจกรรม คือ ใช้อธิบายกิจกรรมที่เกิดขึ้นในลักษณะกระแสการไหลของการทำงาน (Workflow) จะมีลักษณะเดียวกับ Flowchart (แสดงขั้นตอนการทำงานของระบบ) โดยขั้นตอนในการทำงานแต่ละขั้นจะเรียกว่า Activity



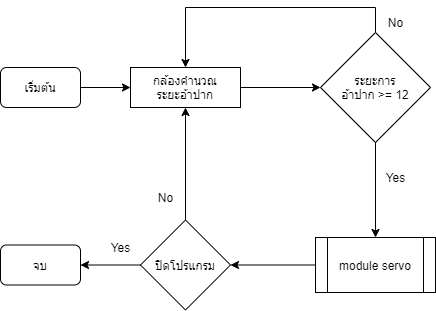
ภาพที่ 8 Activity Diagram ระบบหุ่นยนต์ป้อนอาหาร

3.3 วงจรควบคุมการทำงานของระบบ

ภาพที่ 9 วงจรควบคุมการทำงานของระบบ

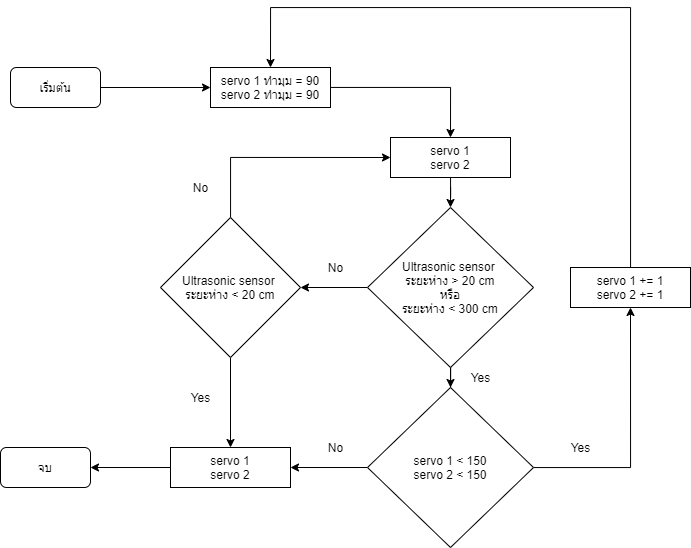
วงจรควบคุมการทำงานทั้งหมดของระบบซึ่งเป็นภาพรวมในการควบคุมหุ่นยนต์แขนกลป้อนอาหารอัตโนมัติ

3.4 รูปแบบการทำงานโปรแกรมของระบบ Artificial Intelligence (AI)

 3.4.1 รูปแบบการทำงานภาพรวมของระบบ Artificial Intelligence (AI)

ภาพที่ 10 รูปแบบการทำงานภาพรวมของระบบ Artificial Intelligence (AI)

3.4.2 รูปแบบการทำงานของ module servo



ภาพที่ 11 รูปแบบการทำงานของ module servo

3.5 หลักการวัดระยะห่างและทำมุมองศาของแขนกล

การนำหลักทฤษฎีทางวิทยาศาสตร์มาประยุกต์ใช้เพื่อให้แขนกลสามารถวัดระยะห่างและการแปรสัญญาณความถี่เพื่อมาช่วยการทำนวณองศาที่แน่นอนขึ้น

3.5.1 หลักการวัดระยะห่างโดยใช้ Ultrasonic Sensor

โดยใช้ Ultrasonic Sensor มาช่วยคำนวณหาระยะห่างกับผู้ป่วย เพื่อนำค่าที่ได้ส่งไปให้ Servo Motor ทำมุมองศาที่ถูกต้องที่สุด



ภาพที่ 12 Ultrasonic Sensor คำนวณหาระยะห่าง

สมมการที่ใช้หาระยะห่างคือ

ความเร็ว = ระยะทาง / เวลา

ความเร็วเสียงมีค่าประมาณ 340 เมตร / วินาที และให้ระยะทางเท่ากับ 2D หารด้วยเวลา เมื่อเราถอดสมการออกมา จะได้ในรูป

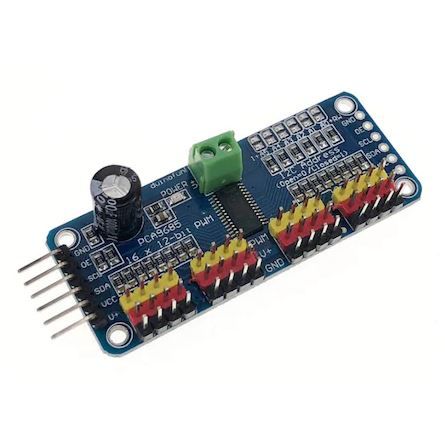
ความเร็ว =

340 =

D = 170 x เวลา

3.5.2 หลักการแปรสัญญาณความถี่ของ PCA9685 เพื่อควบคุมองศา

โดยโครงสร้างของแขนกลที่พัฒนาได้นำ PCA9685 หรือก็คือ PWM LED controller ที่ช่วยในการควบคุมการทำงานของ Servo Motor ได้หลายตัวพร้อมกัน โดยทางเราได้ใช้ library ที่ชื่อว่า Adafruit PCA9685 ซี่งเป็น library ที่พัฒนามาเพื่อรองรับการทำงานบนภาษา python เพื่อควบคุมการทำงาน PCA9685 16-Channel ในการแปรสัญญาณความถี่ของ servo motor



ภาพที่ 13 ควบคุม Servo Motor ผ่าน Adafruit PCA9685

สัญญาณความถี่ของ library ที่แปรออกมาจะอยู่ในช่วง 100 Hz ถึง 700 Hz โดยที่ความถี่ 100 Hz จะมีค่ามุมอยู่ที่ 0 องศา ส่วน 700 Hz จะมีค่ามุมประมาณ 252 องศา

0 องศา = 100 Hz

15 องศา = 142 Hz

30 องศา = 183 Hz

45 องศา = 225 Hz

90 องศา = 350 Hz

135 องศา = 475 Hz

180 องศา = 600 Hz

โดยที่มุม 1 องศาจะมีค่าประมาณที่ = 2.77 Hz

ซึ่งจะมีความยากลำบากอย่างมากในการกำหนดมุม ทางเราซึ่งได้เขียนสมการฟังก์ชันใหม่เพื่อให้สามารถนำค่าองศาไปใช้ในการควบคุมได้เลย

สมการของฟังก์ชันที่ใช้คือ

องศา = C

ความถี่ = 2.77 x C

ความถี่ = ความถี่ + 100

ความถี่ = int(องศา)

ตัวอย่างเช่น มุมที่ต้อง = 70 องศา

degree = 70

Hz = 2.77 x 70

Hz = 193.9 + 100

Hz = int (293.9)

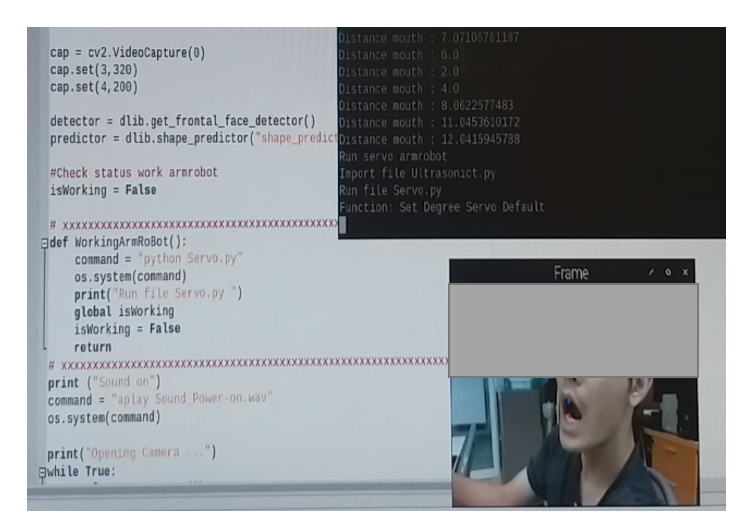
Hz = 293

หลังจากที่แทนค่าองศาของมุม ก็จะสามารถนำไปแปรสัญญาณความถี่เพื่อควบคุม servo motor ได้เลย

# บทที่ 4 ผลการดำเนินโครงงาน

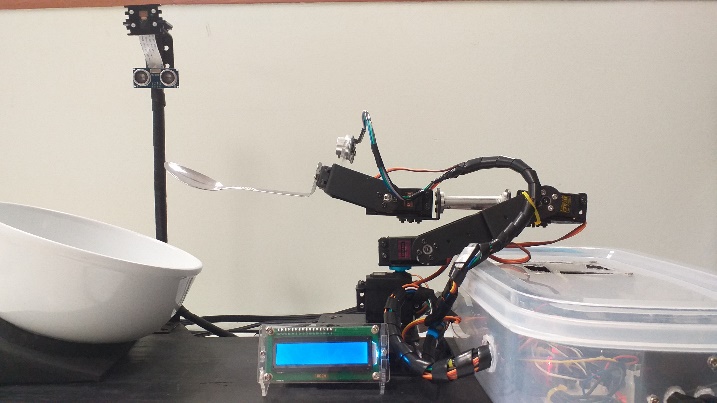
การพัฒนาต่อยอดหุ่นยนต์แขนกลป้อนอาหารอัตโนมัติโดยใช้ Artificial Intelligence (AI) โดยนำ Ultrasonic Sensor เข้ามาช่วยเพื่อให้การทำงานของหุ่นยนต์แขนกลให้มีประสิทธิภาพความสมบูรณ์มากยิ่งขึ้น

## 4.1 การพัฒนาระบบ

เป็นส่วนของการทำงานตรวจจับการอ้าปากแล้วระบบจึง เริ่มกระบวนการป้อนอาหาร โดยกล้องจับจุดริมฝีปากระหว่างริมฝีปากบน กับ ริมฝีปากล่าง ว่าระยะอ้าปากพร้อมรับประทานอาหารแล้วหรือยัง

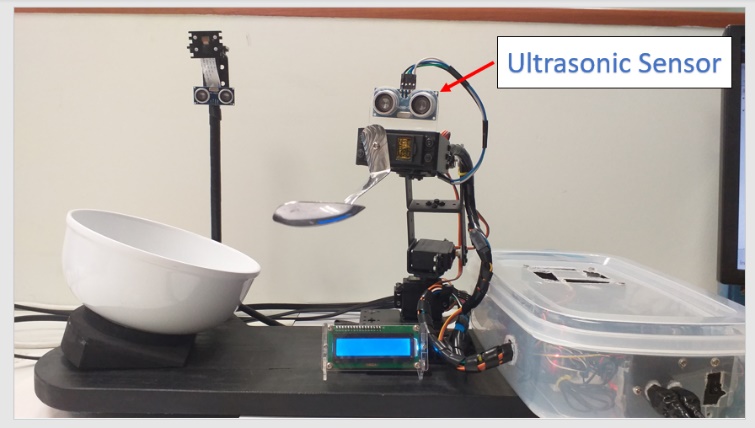
ภาพที่ 14 ผู้ป่วยหรือผู้สูงอายุอ้าปากเพื่อรับทานอาหาร

เมื่อกล้องทำงานตรวจจับการอ้าปากได้ระยะห่างริมฝีปากบน กับ ริมฝีปากล่าง ที่กำหนดไว้ก็จะเริ่มการทำงานของหุ่นยนต์แขนกลในส่วนของตักอาหาร

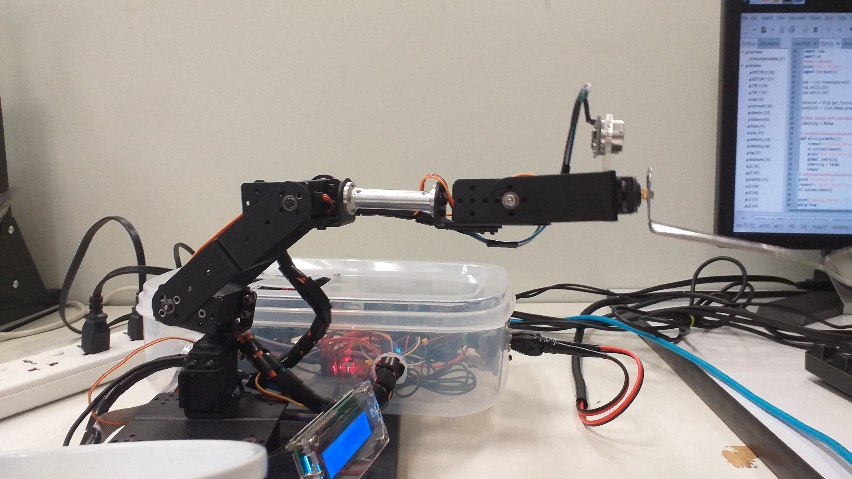


ภาพที่ 15 รูปแขนกลป้อนอาหาร

เมื่อตักอาหารเสร็จ แขนกลก็จะหมุนมาทางผู้ใช้งาน จากนั้น Ultrasonic Sensor ก็จะทำงานในส่วนวัดระยะห่างกับผู้ใช้



ภาพที่ 16 รูป Ultrasonic Sensor บนอุปกรณ์แขนกลป้อนอการ

 หุ่นยนต์แขนกลจะนำค่าจาก Ultrasonic Sensor ที่ได้มาประมวลผล เพื่อเพิ่มองศาของ servo motor แต่ละส่วนเพื่อให้ได้ระยะห่างที่พอดีกับผู้ใช้งาน

ภาพที่ 17 รูปการทำงานของแขนกลป้อนอาหาร

## 4.2 การทดสอบและเปรียบเทียบ

ตารางที่ 5 ผลการทดสอบภาพรวม

|  |  |  |
| --- | --- | --- |
| รายการทดสอบ | จำนวนการทดสอบ | คิดเป็นเปอร์เซ็นต์ความถูกต้อง |
| การทดสอบอ้างปากจับจุดริมฝีปาก | 100 ครั้ง | 100 % |
| การทดสอบการทำงานของ Servo Motor | 100 ครั้ง | 88 % |
| การทดสอบใช้งาน Ultrasonic Sensor | 100 ครั้ง | 100 % |
| การทดสอบการทำงานของโปรแกรม | 100 ครั้ง | 98 % |
| การทดสอบจากผู้ใช้งาน การทำงานภาพรวมของเครื่อง | 100 ครั้ง | 89 % |

ตารางที่ 6 ผลการทดสอบและเปรียบเทียบการทำงาน

|  |  |  |
| --- | --- | --- |
| รายการ | เวอร์ชันก่อนหน้านี้ | เวอร์ชันหลังจากพัฒนา |
| การทำงาน | แขนกลทำงานที่ลำดับ ซึ่งต้องรอแต่ส่วนทำงานให้สำเร็จก่อนจึงจะทำงาน ส่วนถัดไป ซึ่งเวลาการทำงานภาพรวม ค่อยข้างนาน | แขนกลแต่ละส่วนทำงานได้พร้อมกันมากขึ้น โดยภาพรวมจะมีเวลาการทำงานที่ เร็วขึ้นมาก |
| ความสามารถของระบบ | มีกล้องจับภาพริมฝีปากเพื่อช่วยในการทำงานของระบบแขนกล | มีกล้องจับภาพริมฝีปาก นำ Ultrasonic Sensor มาช่วยในการคำนวณมุม และ องศาของแขนกล |
| เวลาการทำงาน | เวลาการทำงาน ในส่วนขั้นตอน ตักข้าว จนถึง ขั้นตอนป้อนอาหาร ใช้เวลา 3 นาที | เวลาการทำงาน ในส่วนขั้นตอน ตักข้าว จนถึง ขั้นตอนป้อนอาหาร ใช้เวลา 1 นาที |

# บทที่ 5 สรุปผลการดำเนินโครงงาน

## 5.1 สรุปผลการดำเนินงาน

ตัวแขนกลนั้นสามารถขยับและป้อนอาหารให้แก่ผู้ที่พิการทางแขนได้ ผู้ที่พิการทางแขนสามารถรับประทานอาหารเองได้โดยไม่จำเป็นต้องขยับแขนของตนเอง แขนกลนั้นสามารถตักอาหารในภาชนะ สามารถวัดระยะห่างกับผู้พิการทางแขนกับแขนกลได้ และทำการป้อมอาหาร โดยที่แขนกลทำมุมไปให้ผู้ใช้ได้อย่างแม่นยำ ซ้อนไม่เข้าไปในปากผู้พิการทางแขนมากหรือน้อยเกินไป

## 5.2 ปัญหา อุปสรรค และ ข้อจำกัด

* 1. โครงสร้างแขนกลนั้นมีน้ำหนัก และร่วมไปถึงของส่วน Servo Motor ที่มากไปจึงทำให้องศาของมุม ไม่ตรงเท่าที่ควร
  2. ตัวแขนยังมีปัญหาการสั่น ซึ่งส่วนนึงมาจากโครงสร้างที่มีน้ำหนักทำให้ servo motor ทำงานหนัก
  3. การทำงานของโปรแกรมในส่วนของกล้องจับใบหน้า มีกระบวนการประมวลผล เยอะ ทำให้บอร์ดมีการประมวลผลมาก ส่งผลให้กล้องที่จับใบหน้านั้นเกิดการแสดง Frame rate ที่ต่ำ ทำให้เกิดภาพค้าง หรือ ภาพกระตุก
  4. การจ่ายไฟยังไม่คงที่พอ เนื่องจากสายไฟหลวมบางจุด ซึ่งทำให้ตัวแขนกลมี เกิดอาการผิดปกติ เช่น การทำงานของ servo motor ผิดเพี้ยน ultrasonic sonic ไม่ทำงาน หรือ ค่าที่ได้ ระยะห่าง ไม่ตรงเท่าที่ควร

## 5.3 ข้อเสนอแนะและงานในอนาคต

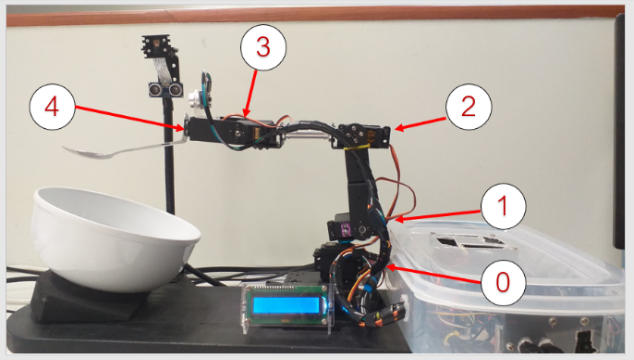
1. การพัฒนาระบบแขนกลให้มีประสิทธิภาพเพิ่มขึ้น
2. พัฒนาโครงสร้างของแขนกลให้แข็งแรง และ มีน้ำหนักโครงสร้างเบาลง
3. พัฒนามือจับเพื่อให้สามารถไปประยุกต์ใช้กับงานด้านอื่นได้

# บรรณานุกรม

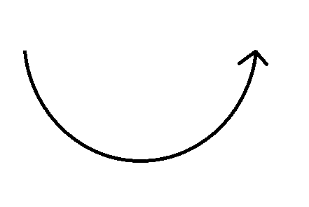
|  |  |
| --- | --- |
| [1] | nessessence, “ปัญญาประดิษฐ์ (AI : Artificial Intelligence) คืออะไร ???” [ออนไลน์] เข้าถึงจาก https://www.thaiprogrammer.org/2018/12/whatisai/ สืบค้นเมื่อ 30 ธันวาคม 2563 |
| [2] | Nuttakan Chuntra, “OpenCV คืออะไร?” [ออนไลน์] เข้าถึงจากhttps://medium.com/@nut.ch40/opencv-คืออะไร-8771e2a4c414 สืบค้นเมื่อ 12 ธันวาคม 2563 |
| [3] | Adrian Rosebrock, “Facial landmarks with dlib, OpenCV, and Python” [ออนไลน์] เข้าถึงจาก <https://www.pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-python/> สืบค้นเมื่อ 12 ธันวาคม 2563 |
| [4] | Maker Tutor, “Easy way!! Control Multiple Servo with Raspberry Pi” [ออนไลน์] เข้าถึงจาก <https://www.youtube.com/watch?v=9jcEwn7GzNs> สืบค้นเมื่อ 12 ธันวาคม 2563 |
| [5] | Gaven MacDonald, “Ultrasonic Sensor with the Raspberry Pi” [ออนไลน์] เข้าถึงจาก <https://www.youtube.com/watch?v=9jcEwn7GzNs> สืบค้นเมื่อ 12 ธันวาคม 2563 |
| [6] | GUGLIELMELLI, Eugenio, et al, “Self-feeding apparatus” [ออนไลน์] เข้าถึงจาก  <https://patents.google.com/patent/US8240967B2/en> สบค้นเมื่อ 2 มกราคม 2563 |  |
| [7] | เศรษฐพงศ์ ผาละพรม. (2562). หุ่นยนต์แขนกลป้อนอาหารอัตโนมัติโดยใช้ Artificial Intelligence (iFeedingBot) (วิทยานิพนธ์ปริญญาบัณฑิต). ชลบุรี: มหาวิทยาลัยบูรพา |
|  |  |
|  |  |
|  |  |

# ภาคผนวก

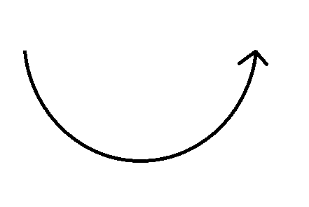
## ภาคผนวก ก การหมุนของ Servo Motor แต่ละส่วน

Servo Motor แต่ละส่วนจะมีการหมุนที่ไม่เหมือนกัน หมายเลขของ Servo Motor ที่กำหนดดังรูปต่อไปนี้

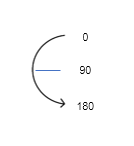
ภาพที่ 18 ตำแหน่งหมายเลขของ Servo Motor

Servo Motor หมายเลข 0 กับ หมายเลข 4 จะหมุนจากทางซ้าย มุม 0 องศา ไปทางขวา สุดที่มุม 180 องศา

ภาพที่ 19 อธิบายการหมุนของ Servo Motor หมายเลข 0 กับ หมายเลข 4

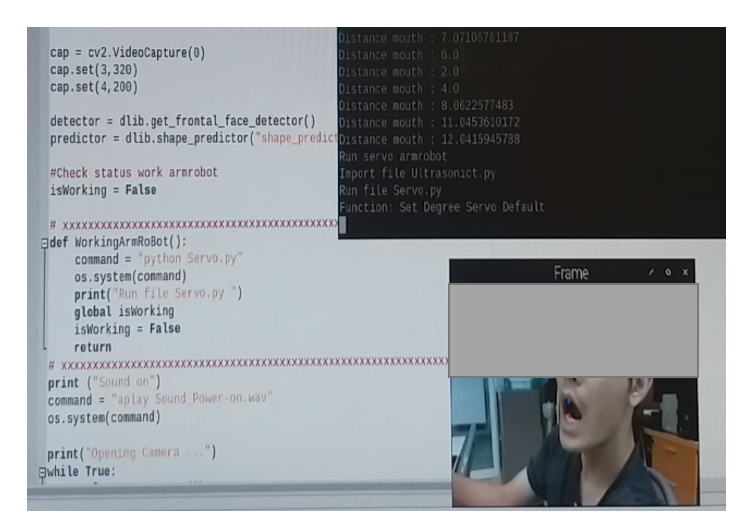
Servo Motor หมายเลข 1 กับ หมายเลข 2 จะหมุนจากทางขวา มุม 0 องศา ไปทางขวา สุดที่มุม 180 องศา

ภาพที่ 20 อธิบายการหมุนของ Servo Motor หมายเลข 1 กับ หมายเลข 2

Servo Motor หมายเลข 3 จุดหมุนเริ่มต้นจะอยู่ที่มุม 90 องศา หมุนลงมุมสุดที่ 180 องศา หมุนขึ้นมุมสุดที่ 0 องศา

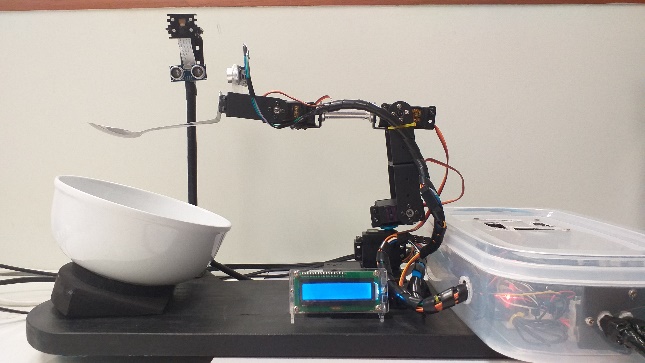
ภาพที่ 21 อธิบายการหมุนของ Servo Motor หมายเลข 3

## ภาคผนวก ข การทำงานของโปรแกรม

1. เมื่อรันโปรแกรม โปรแกรมก็จะเริ่มทำงานในส่วนของกล้องเพื่อจับจุดริมฝีปากบน กับริมฝีปากล่าง ตรวจระยะการอ้าปากว่าถึงระยะที่กำหนดหรือยัง เมื่อถึงจุดที่กำหนดระบบก็จะเริ่มทำงานส่วน แขนกลเพื่อตักอาหาร

ภาพที่ 22 อธิบายการทำงานของกล้องตรวจระยะอ้าปากเพื่อรับทานอาหาร

1. ก่อนทำงานจะกำหนดมุม Servo Motor เริ่มต้นทุกครั้งก่อนทำงาน เพื่อป้องกันปัญหาข้อผิดพลาด



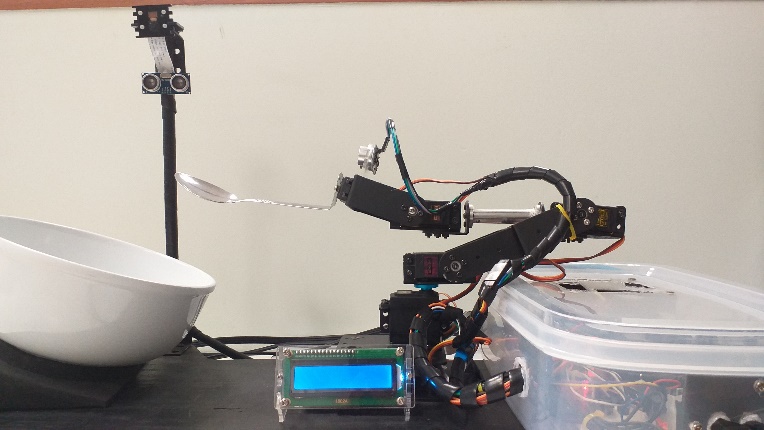
ภาพที่ 23 แขนกลเซ็ตมุมก่อนทำงานเพื่อป้องกันปัญหาข้อผิดพลาด

โดยมุมของ Servo Motor ของแต่ละส่วนจะกำหนดค่าองศานั้นนี้

ตารางที่ 7 องศาของ Servo Motor ของแต่ละส่วนที่กำหนดก่อนทำงาน

|  |  |
| --- | --- |
| Servo Motor | องศา |
| หมายเลข 0 | 10 |
| หมายเลข 1 | 90 |
| หมายเลข 2 | 90 |
| หมายเลข 3 | 85 |
| หมายเลข 4 | 90 |

1. เมื่อผ่านการทำงานขั้นตอนกำหนดมุมองศาก่อนเริ่มทำงานเสร็จแล้ว ก็จะเริ่มทำงาน ตักอาหารโดย แขนจะทำงานฟังก์ชัน เตรียมก่อนตักอาหาร ดังรูปต่อไปนี้



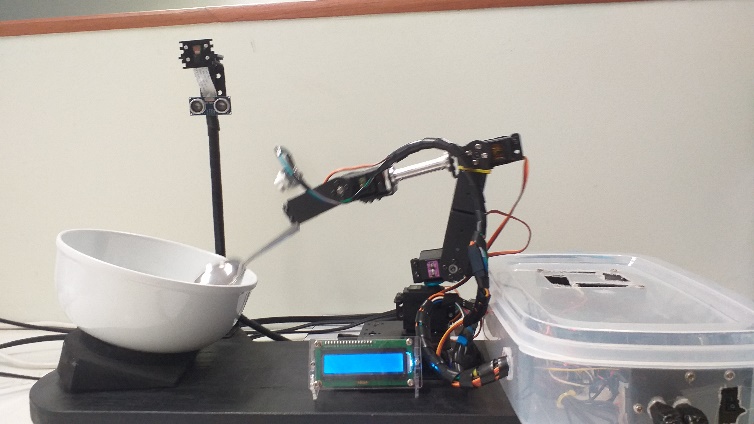
ภาพที่ 24 แขนกลทำงานฟังก์ชันเตรียมก่อนตักอาหาร

โดยมุมของ Servo Motor ของแต่ละส่วนจะกำหนดค่าองศานั้นนี้

ตารางที่ 8 องศาของ Servo Motor ของฟังก์ชันเตรียมก่อนตักอาหาร

|  |  |
| --- | --- |
| Servo Motor | องศา |
| หมายเลข 0 | 10 |
| หมายเลข 1 | 20 |
| หมายเลข 2 | 0 |
| หมายเลข 3 | 85 |
| หมายเลข 4 | 0 |

1. ขั้นตอนกำหนดการทำงานฟังก์ชัน ตักอาหาร ดังรูปต่อไปนี้

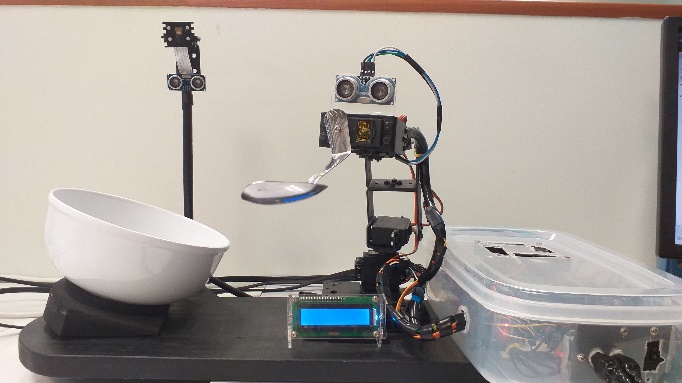


ภาพที่ 25 แขนกลทำงานฟังก์ชันตักอาหาร

โดยมุมของ Servo Motor ของแต่ละส่วนจะกำหนดค่าองศานั้นนี้

ตารางที่ 9 องศาของ Servo Motor ของฟังก์ชันเตรียมก่อนตักอาหาร

|  |  |
| --- | --- |
| Servo Motor | องศา |
| หมายเลข 0 | 10 |
| หมายเลข 1 | 90 |
| หมายเลข 2 | 50 |
| หมายเลข 3 | 140 |
| หมายเลข 4 | 90 |

1. เมื่อตักอาหารขึ้นมาแขนกลจะหมุนจากภาชนะ มาทางผู้รับประทานอาหาร ดังรูปสาธิตการตักอาหารต่อไปนี้

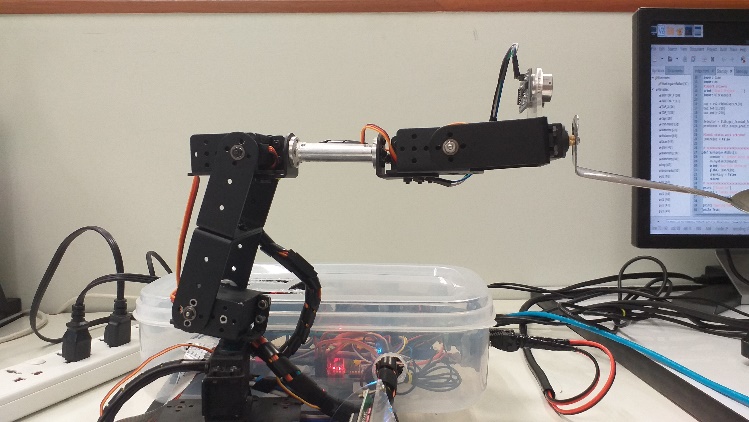
โดยมุมของ Servo Motor ของแต่ละส่วนจะกำหนดค่าองศานั้นนี้

ภาพที่ 26 สาธิตการทำงานหมุนมาทางผู้รับประทานอาหาร

ตารางที่ 10 องศาของ Servo Motor ของฟังก์ชันตักอาหารหันมาทางผู้รับประทานอาหาร

|  |  |
| --- | --- |
| Servo Motor | องศา |
| หมายเลข 0 | 80 |
| หมายเลข 1 | 90 |
| หมายเลข 2 | 90 |
| หมายเลข 3 | 100 |
| หมายเลข 4 | 90 |

1. หลังจากแขนกลหันทางผู้รับประทานอาหาร Servo Motor จะกำหนดมุมองศาในแต่ละส่วน ดังใน ตารางที่ 11



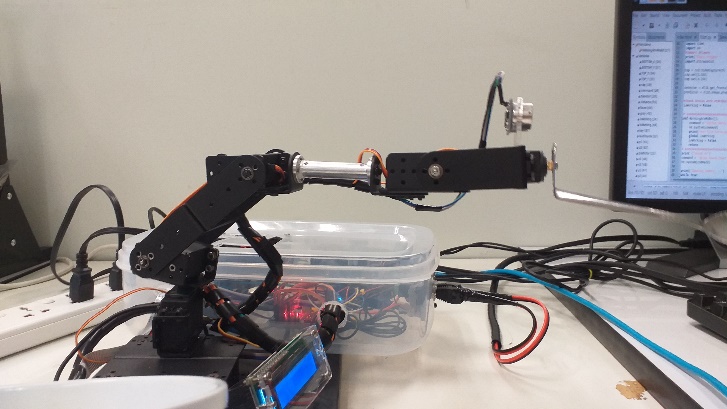
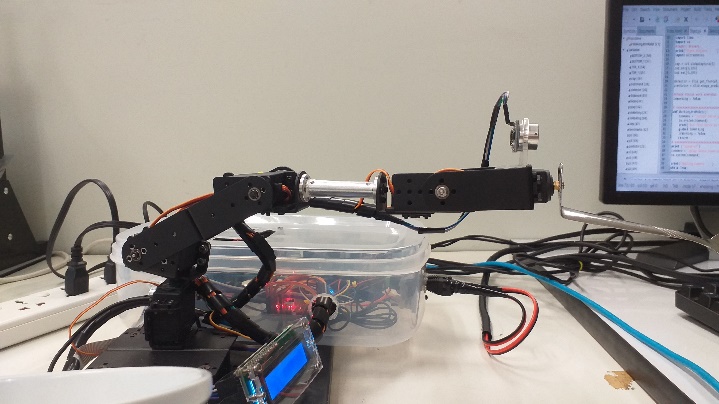
ภาพที่ 27 แขนกลหันทางผู้รับประทานอาหาร

โดยมุมของ Servo Motor ของแต่ละส่วนจะกำหนดค่าองศานั้นนี้

ตารางที่ 11 องศาของ Servo Motor ของฟังก์ชันแขนกลหันทางผู้รับประทานอาหาร

|  |  |
| --- | --- |
| Servo Motor | องศา |
| หมายเลข 0 | 90 |
| หมายเลข 1 | 90 |
| หมายเลข 2 | 90 |
| หมายเลข 3 | 100 |
| หมายเลข 4 | 90 |

1. จากนั้นโปรแกรมจะทำงานในส่วน Ultrasonic Sensor เพื่อคำนวณระยะห่างกับผู้รับประทาน แขนกลจะเพิ่มองศาของ Servo Motor จนสุดค่าของ Servo Motor หมายเลข 1 กับ 2 สุดที่มุม 150 องศา



ภาพที่ 28 แขนกลเพิ่มองศาจนสุดที่มุม 150 องศา

ภาพที่ 29 แขนกลเพิ่มองศาเพื่อขยับเข้าใกล้ผู้รับประทานอาหาร

โดยมุมของ Servo Motor ของแต่ละส่วนจะกำหนดค่าองศานั้นนี้

|  |  |
| --- | --- |
| Servo Motor | องศา |
| หมายเลข 0 | 90 |
| หมายเลข 1 | 120 |
| หมายเลข 2 | 120 |
| หมายเลข 3 | 100 |
| หมายเลข 4 | 90 |

ตารางที่ 12 องศาของ Servo Motor ของภาพที่ 27

ตารางที่ 13 องศาของ Servo Motor ของภาพที่ 28

|  |  |
| --- | --- |
| Servo Motor | องศา |
| หมายเลข 0 | 90 |
| หมายเลข 1 | 150 |
| หมายเลข 2 | 150 |
| หมายเลข 3 | 100 |
| หมายเลข 4 | 90 |

## ภาคผนวก ค โปรแกรม

การทำงานของโปรแกรมต่างละไฟล์ระเริ่มที่ไฟล์ Start.py เป็นไฟล์แรกที่จะทำงานก่อนแล้วจึงจะไปเรียกใช้ต่าง ๆ เช่น ไฟล์ Ultrasonict.py ทำหน้าที่คำนวณระยะห่างโดยใช้คลื่นเสียง ส่วนไฟล์ Servo.py ทำงานส่วนควบคุม Servo Motor แต่ละส่วนให้สัมพันธ์กัน

ตารางที่ 14 Start.py

|  |
| --- |
| from \_\_future\_\_ import division  from threading import Thread  import RPi.GPIO as GPIO #นำเข้าเพื่อใช้งาน GPIO  import multiprocessing as mp #นำเข้าเพื่อให้โปรแกรมทำงานหลายงานได้พร้อมกัน  import Adafruit\_PCA9685 #นำเข้าเพื่อเรียกไลบรารีควบคุมการทำงาน Servo Motor 16 ช่อง  import cv2 #นำเข้าเพื่อการแสดงผลแบบเรียลไทม์  import numpy as np #นำเข้าเพื่อใช้คำนวณทางคณิตศาสตร์ในการจัดการกับอาเรย์หลายมิติ  import dlib #นำเข้าเพื่อเรียกใช้การรู้จำตำแหน่งจุดของใบหน้า  import math #นำเข้าเพื่อใช้คำนวณ  import time #นำเข้าเพื่อคิดเวลา  import os #นำเข้าเพื่อใช้คำสั่งผ่านทาง command line  print("Start Project ....")  import Ultrasonict #นำเข้าเพื่อคำนวณระยะจากคลื่นเสียง |
| cap = cv2.VideoCapture(0) #หน้าต่างไว้แสดงผลแบบเรียลไทม์  cap.set(3,320) #กำหนดขนาดเฟรมความกว้าง  cap.set(4,200) #กำหนดขนาดเฟรมควมาสูง  detector = dlib.get\_frontal\_face\_detector()  predictor = dlib.shape\_predictor("shape\_predictor\_68\_face\_landmarks.dat")  isWorking = False #Check status work armrobot  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def WorkingArmRoBot():  command = "python Servo.py" #ใช้คำสั่งเขียน command line เพื่อสั่งทำงานไฟล์ Servo.py  os.system(command)  print("Run file Servo.py ")  global isWorking  isWorking = False  return  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  print ("Sound on")  command = "aplay Sound\_Power-on.wav" #ใช้คำสั่งเขียน command line เพื่อเปิดไฟล์เสียง  os.system(command)  print("Opening Camera ...")  while True:  \_, frame = cap.read()  gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)  faces = detector(gray)  for face in faces:  x1 = face.left()  y1 = face.top()  x2 = face.right()  y2 = face.bottom()  landmarks = predictor(gray, face)  TOP\_X = landmarks.part(62).x #สร้างตัวแปรเพื่อใช้กำหนดจุดที่ริมฝีปากบน กำหนดเป็นแกน x  TOP\_Y = landmarks.part(62).y #สร้างตัวแปรเพื่อใช้กำหนดจุดที่ริมฝีปากบน กำหนดเป็นแกน y  BOTTOM\_X = landmarks.part(66).x #สร้างตัวแปรเพื่อใช้กำหนดจุดที่ริมฝีปากล่าง กำหนดเป็นแกน x  BOTTOM\_Y = landmarks.part(66).y #สร้างตัวแปรเพื่อใช้กำหนดจุดที่ริมฝีปากล่าง กำหนดเป็นแกน y  cv2.circle(frame, (TOP\_X, TOP\_Y), 2, (255,0,0), -1)  cv2.circle(frame, (BOTTOM\_X, BOTTOM\_Y), 2, (0,0,255), -1)  p1 = [TOP\_X,TOP\_Y] #สร้างตัวแปรเพื่อเรียกใช้ตำแหน่งจุดริมฝีปากบน  p2 = [BOTTOM\_X,BOTTOM\_Y] #สร้างตัวแปรเพื่อเรียกใช้ตำแหน่งจุดริมฝีปากล่าง  distance = math.sqrt( ((p1[0]-p2[0])\*\*2)+((p1[1]-p2[1])\*\*2) )  if isWorking == False :  if distance >= 12 : #ถ้าระยะห่างของริมฝีปากบน กับ ล่างห่างเกิน 12 cm  print "Distance mouth :",distance  isWorking = True  print "Run servo armrobot"  Thread(target=WorkingArmRoBot).start() #เรียกใช้ฟังก์ชัน แล้วสั่งไฟล์รันไฟล์ Servo.py  else:  print "Distance mouth :",distance  else: #ถ้าสถานะ isWorking เป็น True จะไม่แสดงอะไรเลย    cv2.imshow("Frame", frame)  key = cv2.waitKey(1)  if key == 27:  break |

ตารางที่ 15 Servo.py

|  |
| --- |
| from \_\_future\_\_ import division  import time #นำเข้าเพื่อคิดเวลา  import Adafruit\_PCA9685 #นำเข้าเพื่อเรียกไลบรารีควบคุมการทำงาน Servo Motor 16 ช่อง  import multiprocessing as mp #นำเข้าเพื่อให้โปรแกรมทำงานหลายงานได้พร้อมกัน  import Ultrasonict Ultrasonict #นำเข้าเพื่อคำนวณระยะจากคลื่นเสียง  import os #นำเข้าเพื่อใช้คำสั่งผ่านทาง command line |
| pwm = Adafruit\_PCA9685.PCA9685()  print("Run file Servo.py")  def set\_servo\_pulse(channel, pulse):  pulse\_length = 1000000 # 1,000,000 us per second  pulse\_length //= 60 # 60 Hz  print('{0}us per period'.format(pulse\_length))  pulse\_length //= 4096 # 12 bits of resolution  print('{0}us per bit'.format(pulse\_length))  pulse \*= 1000  pulse //= pulse\_length  pwm.set\_pwm(channel, 0, pulse) #ควบคุมการทำงานของ Servo Motor หมายเลข 0  pwm.set\_pwm(channel, 1, pulse) #ควบคุมการทำงานของ Servo Motor หมายเลข 1  pwm.set\_pwm(channel, 2, pulse) #ควบคุมการทำงานของ Servo Motor หมายเลข 2  pwm.set\_pwm(channel, 3, pulse) #ควบคุมการทำงานของ Servo Motor หมายเลข 3  pwm.set\_pwm(channel, 4, pulse) #ควบคุมการทำงานของ Servo Motor หมายเลข 4  pwm.set\_pwm\_freq(60)  ''' convert pulse to degree  variable a Is Channel  variable b To 0, don't need to pay attention  variable c Is Degree  # servo\_min = 100 Hz is 0 degree  # servo\_Almostmax = 600 Hz is 180 degree  # servo\_max = 700 Hz is 252 degree  # 0 degree = 100 Hz  # 15 degree = 142 Hz  # 30 degree = 183 Hz  # 45 degree = 225 Hz  # 90 degree = 350 Hz  # 135 degree = 475 Hz  # 180 degree = 600 Hz  # 1 degree = 2.77 Hz  '''  def calDeg(a,b,c): #เป็นฟังก์ชันเมื่อมีค่ามุมขององศา ส่งมาให้แปลงค่าเป็น Hz  re\_deg = c  degree = 2.77\*c  degree = degree+100  degree = int(degree)  pwm.set\_pwm(a,b,degree)  return re\_deg  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def default(): #เซ็ตมุมแต่ละส่วนก่อนทำงานเพื่อป้องกันปัญหาข้อผิดพลาดก่อนหมุม  calDeg(0, 0, 10)  calDeg(1, 1, 90)  calDeg(2, 2, 90)  calDeg(3, 3, 85)  calDeg(4, 4, 90)  print("Function: Set Degree Servo Default")  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def default\_low(): #ฟังก์ชันหลักเตรียมพร้อมก่อนตักเริ่มตักอาหาร  calDeg(0, 0, 10)  calDeg(1, 1, 90)  calDeg(2, 2, 90)  calDeg(3, 3, 85)  calDeg(4, 4, 90)  def default\_low\_channel1(): #servo channel 1  for i in range(90, 20, -1):  calDeg(1, 1, i)  time.sleep(0.01)  def default\_low\_channel2(): #servo channel 2  for i in range(90, 0, -1):  calDeg(2, 2, i)  time.sleep(0.01)  if(\_\_name\_\_=='\_\_main\_\_'):  p2 = mp.Process(target=default\_low\_channel2)  p1 = mp.Process(target=default\_low\_channel1)  p2.start()  p1.start()  time.sleep(0.5)  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def default\_rice(): #ฟังก์ชันควบคุมการหมุมการทำงานให้แขนกลตักอาหาร  def default\_rice\_channel1(): #servo channel 1  for i in range(20, 45, 1):  calDeg(1, 1, i)  time.sleep(0.02)  def default\_rice\_channel2(): #servo channel 2  for i in range(10, 50, 1):  calDeg(2, 2, i)  time.sleep(0.02)  #print("deg servo 2 "+str(i))  def default\_rice\_channel3(): #servo channel 3  for i in range(60, 90, 1):  calDeg(3, 3, i)  time.sleep(0.02)  print ("Function: default rice")  calDeg(0, 0, 10)  calDeg(1, 1, 20)  calDeg(2, 2, 10)  calDeg(3, 3, 60)  calDeg(4, 4, 0)  default\_rice\_channel2()  default\_rice\_channel1()  default\_rice\_channel3()  time.sleep(0.5)  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def scoop\_rice(): #ฟังก์ชันควบคุมการหมุมแขนกลตักอาหาร  def scoop\_rice\_channel1(): #servo channel 1  for i in range(45, 90, 1):  calDeg(1, 1, i)  time.sleep(0.02)    def scoop\_rice\_channel3(): #servo channel 3  for i in range(90, 140, 1):  calDeg(3, 3, i)  time.sleep(0.02)    def scoop\_rice\_channel4(): #servo channel 4  for i in range(0, 90, 1):  calDeg(4, 4, i)  time.sleep(0.01)  print ("Function: scoop rice")  calDeg(0, 0, 10)  calDeg(1, 1, 45)  calDeg(2, 2, 50)  calDeg(3, 3, 90)  calDeg(4, 4, 0)    if(\_\_name\_\_=='\_\_main\_\_'):  p3 = mp.Process(target=scoop\_rice\_channel3)  p1 = mp.Process(target=scoop\_rice\_channel1)  p3.start()  p1.start()  time.sleep(0.5)  scoop\_rice\_channel4()  time.sleep(0.5)  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def scoop\_up(): #ฟังก์ชันควบคุมการหมุมแขนกลตักอาหารขึ้นมา  def scoop\_up\_channel3(): #servo channel 3  for i in range(141, 100, -1):  calDeg(3, 3, i)  time.sleep(0.05)  def scoop\_up\_channel2(): #servo channel 2  for i in range(50, 90, 1):  calDeg(2, 2, i)  time.sleep(0.03)  #print("deg servo 2 "+str(i))  print ("Function: scoop up")  calDeg(0, 0, 10)  calDeg(1, 1, 90)  calDeg(2, 2, 50)  calDeg(3, 3, 140)  calDeg(4, 4, 90)    scoop\_up\_channel3()  time.sleep(0.05)  scoop\_up\_channel2()  time.sleep(0.05)  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def turn\_corner\_forward(): #ฟังก์ชันควบคุมการหมุมจากภาชนะมาทางผู้รับประทาน  def turn\_corner\_forward\_channel0(): #servo channel 0  for i in range(10, 90, 1):  calDeg(0, 0, i)  time.sleep(0.03)    print ("Function: turn corner forward")  calDeg(0, 0, 10)  calDeg(1, 1, 90)  calDeg(2, 2, 90)  calDeg(3, 3, 100)  calDeg(4, 4, 90)  turn\_corner\_forward\_channel0()  time.sleep(0.05)  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def arm2fit(): #ฟังก์ชันเซ็ตค่าแขนกลก่อนหมุนแขนกลเข้าใกล้ผู้รับประทาน  calDeg(0, 0, 90)  calDeg(1, 1, 90)  calDeg(2, 2, 90)  calDeg(3, 3, 100)  calDeg(4, 4, 90)  print 'Function: arm to fit'  time.sleep(0.05)  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def turn\_cornerback(s1, s2): #ฟังก์ชันควบคุมการหมุมแขนกลเพิ่มองศาให้เข้าใกล้ผู้รับประทาน  print 'Function: turn cornerback'  print ("Value servo 1 = " + str(s1)) #แสดงค่า หมุมองศา ปัจจุบันของ Servo Motor หมายเลข 1  print ("Value servo 2 = " + str(s2)) #แสดงค่า หมุมองศา ปัจจุบันของ Servo Motor หมายเลข 0  def turn\_cornerback\_channel1(): #servo channel 1  for i in range(s1, 90, -1):  calDeg(1, 0, i)  print("servo 1 deg: "+str(i))  time.sleep(0.03)  def turn\_cornerback\_channel2(): #servo channel 2  for i in range(s2, 90, -1):  calDeg(2, 0, i)  print("servo 2 deg: "+str(i))  time.sleep(0.03)    calDeg(3, 3, 100)  calDeg(4, 4, 90)  if(\_\_name\_\_=='\_\_main\_\_'):  p2 = mp.Process(target=turn\_cornerback\_channel2)  p1 = mp.Process(target=turn\_cornerback\_channel1)  p2.start()  p1.start()  time.sleep(0.05)  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  def turn\_back(): #ฟังก์ชันหมุนแขนกลจากด้านตรงผู้รับประทานอาหารให้หมุมกับไปทางภาชนะ  def turn\_back\_channel0(): #servo channel 0  for i in range(90, 10, -1):  calDeg(0, 0, i)  time.sleep(0.01)  print ("Function: turn back")  calDeg(0, 0, 10)  calDeg(1, 1, 90)  calDeg(2, 2, 90)  calDeg(3, 3, 100)  calDeg(4, 4, 90)  turn\_back\_channel0()  time.sleep(0.05)  # xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  default()  time.sleep(1)  default\_low()  time.sleep(0.5)  time.sleep(0.1)  default\_rice()  time.sleep(0.1)  scoop\_rice()  time.sleep(0.1)  scoop\_up()  time.sleep(0.1)  turn\_corner\_forward()  time.sleep(0.1)  arm2fit()  time.sleep(1)  servo1 = calDeg(1, 1, 90)  servo2 = calDeg(2, 2, 90)  time.sleep(0.5)  print("Ultrasonic sensor")  arm\_dis = Ultrasonict.check\_extra()  while arm\_dis >= 20:  arm\_dis = Ultrasonict.check\_extra()  print(arm\_dis)  if servo1 < 150:  servo1 += 1  calDeg(1, 1, servo1)  print("servo 1 deg: " + str(servo1))  elif servo1 == 180:  calDeg(1, 1, 180)    if servo2 < 150:  servo2 += 1  calDeg(2, 2, servo2)  print("servo 2 deg: " + str(servo2))  elif servo2 == 180:  calDeg(2, 2, 180)  print("======================")  time.sleep(0.1)  arm\_dis = Ultrasonict.check\_extra()  while arm\_dis < 20:  arm\_dis = Ultrasonict.check\_extra()  for i in range(3):  print('========== OK ==========')    time.sleep(5)  turn\_cornerback(servo1, servo2)  time.sleep(0.5)  turn\_back()  time.sleep(0.5)  print ("Sound off")  command = "aplay Sound\_EndDetect.wav"  os.system(command) |

ตารางที่ 16 Ultrasonict.py

|  |
| --- |
| import RPi.GPIO as GPIO  import time  import lcdlib  import drivers |
| # Ultrasonict Canera  TRIG\_1 = 24 #Purple  ECHO\_1 = 23 #Yellow  # Ultrasonict Arm Robot  TRIG\_2 = 27 #Blue  ECHO\_2 = 22 #Green  GPIO.setmode(GPIO.BCM) # BCM is Number GPIO  GPIO.setwarnings(False)  lcd = lcdlib.lcd()  print("Import file Ultrasonict.py")  try:  #Camera  distance\_Camera = 0  sum\_Camera = 0    def Camera():  print("distance measurement in progress 1 is Camera")  sum\_Camera = 0  for i in range(5):  print "Ultrasonic 1 is Camera i = ",i+1  GPIO.setup(TRIG\_1, GPIO.OUT)  GPIO.setup(ECHO\_1, GPIO.IN)  GPIO.output(TRIG\_1,False)    time.sleep(0.2) # Origin is 0.2  GPIO.output(TRIG\_1, True)  time.sleep(0.00001)  GPIO.output(TRIG\_1, False)    while GPIO.input(ECHO\_1) == 0:  pulse\_start = time.time()  while GPIO.input(ECHO\_1) == 1:  pulse\_end = time.time()    pulse\_duration = pulse\_end-pulse\_start  distance\_Camera = pulse\_duration\*17000  distance\_Camera = round(distance\_Camera,2)  sum\_Camera += distance\_Camera  print "Ultrasonic 1 is Camera distance:",distance\_Camera,"cm"  print 'Ultrasonic 1 is Camera sum = ', sum\_Camera  print('')  time.sleep(0.01)    # Outside the loop  print ("---------------")  print "Ultrasonic 1 is Camera sum = ", sum\_Camera  print "Ultrasonic 1 is Camera num = ", i+1  average\_Camera = round( sum\_Camera / (i + 1) , 2 )  print "Ultrasonic 1 is average =", average\_Camera  print ("---------------")  dist = str(average\_Camera)  lcd.lcd\_display\_string("Distance Camera",1)  lcd.lcd\_display\_string(">>> "+dist+" cm",2)  print "Ultrasonic 1 is Camera Distance is >>> "+dist+" cm"  time.sleep(0.1)    lcd.lcd\_clear()  print("\n \n")  return average\_Camera    # ==============================================================  #Arm  distance\_Arm = 0  sum\_Arm = 0    def Arm():  print("distance measurement in progress 2 is Arm")  sum\_Arm = 0  for i in range(5):  #print "Ultrasonict 2 is i = ",i+1  GPIO.setup(TRIG\_2, GPIO.OUT)  GPIO.setup(ECHO\_2, GPIO.IN)  GPIO.output(TRIG\_2,False)    time.sleep(0.2) # Origin is 0.2  GPIO.output(TRIG\_2, True)  time.sleep(0.00001)  GPIO.output(TRIG\_2, False)    while GPIO.input(ECHO\_2) == 0:  pulse\_start = time.time()  while GPIO.input(ECHO\_2) == 1:  pulse\_end = time.time()    pulse\_duration = pulse\_end-pulse\_start  distance\_Arm = pulse\_duration\*17000  distance\_Arm = round(distance\_Arm,2)  sum\_Arm += distance\_Arm  print "Ultrasonict 2 is distance:",distance\_Arm,"cm"  print 'Ultrasonict 2 is sum = ', sum\_Arm  print('')  time.sleep(0.01)    # Outside the loop  print ("---------------")  print "Ultrasonict 2 is sum = ", sum\_Arm  print "Ultrasonict 2 is num = ", i+1  average\_Arm = round( sum\_Arm / (i + 1) , 2 )  print "Ultrasonict 2 is average =", average\_Arm  #print ("---------------")  dist = str(average\_Arm)  lcd.lcd\_display\_string("Distance Arm",1)  lcd.lcd\_display\_string(">>> "+dist+" cm",2)  #print "Ultrasonic 2 is Camera Distance is >>> "+dist+" cm"  time.sleep(0.1)    lcd.lcd\_clear()  print("\n \n")  #return average\_Arm    # ==============================================================  def extra\_arm():  GPIO.setup(TRIG\_2, GPIO.OUT)  GPIO.setup(ECHO\_2, GPIO.IN)  GPIO.output(TRIG\_2,False)    time.sleep(0.1) # Origin is 0.2  GPIO.output(TRIG\_2, True)  time.sleep(0.00001)  GPIO.output(TRIG\_2, False)    while GPIO.input(ECHO\_2) == 0:  pulse\_start = time.time()  while GPIO.input(ECHO\_2) == 1:  pulse\_end = time.time()    pulse\_duration = pulse\_end-pulse\_start  distance\_extra = pulse\_duration\*17000  distance\_extra = round(distance\_extra,2)    lcd.lcd\_clear()  return distance\_extra    def check\_extra():  a = extra\_arm()  print(a)  while a<=2 or a>=500:  a = extra\_arm()  print(a)  return a  # ==============================================================    #Arm()  #Camera()  #check\_extra()  except KeyboardInterrupt:  GPIO.cleanup()  print("Cleaning up!")  lcd.lcd\_clear() |

# ประวัติผู้จัดทำโครงงาน

|  |  |
| --- | --- |
| **ชื่อ-สกุล** | **นายนิติกานต์ เพ็ชแหวน** |
| **วัน เดือน ปี เกิด** | **5 กันยายน พ.ศ. 2541** |
| **ที่อยู่ปัจจุบัน** | **เลขที่ 154 หมู่ 2 ตำบล วังทอง**  **อำเภอ วังสมบูรณ์ จังหวัด สระแก้ว 27250** |
| **อีเมล** | **nitikran12@gmail.com** |
| **ประวัติการศึกษา** |  |
| **ระดับปริญญาตรี** | **คณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา** |
| **ระดับมัธยมศึกษาตอนปลาย** | **โรงเรียนวังสมบูรณ์วิทยาคม** |
| **ระดับมัธยมศึกษาตอนต้น** | **โรงเรียนวังสมบูรณ์วิทยาคม** |
| **ความสามารถพิเศษ** | **ไม่มี** |
| **เกียรติประวัติและผลงาน** | **ไม่มี** |