



EE03-แบบรายงานความก้าวหน้า

นักศึกษา ☒ ปกติ ☐ สมทบ

รายงานความก้าวหน้าโครงการวิศวกรรมไฟฟ้า

หลักสูตรสาขาวิชาวิศวกรรมไฟฟ้า

รหัสโครงการ : S1-665-CS2

ชื่อโครงการ

รถไร้ขยะเคลื่อนที่ตามผู้ใช้งาน

Autonomous Trash-Collecting Robot Following User

ผู้จัดทำ

นายชุตีพงศ์ แก้วตั้ง

รหัสนักศึกษา 163404130039 โทร. 0925297634

นายพงศกร สมพรหม

รหัสนักศึกษา 163404130051 โทร. 0925148528

ความเห็นอาจารย์ที่ปรึกษา ได้ตรวจโครงการแล้ว เห็นควร ☐ อนุมัติให้สอบได้ ☐ ปรับปรุงเนื้อหาหรือรูปแบบใหม่ ลงชื่อ (.....)	ความเห็นอาจารย์ที่ปรึกษาหลัก ได้ตรวจโครงการแล้ว เห็นควร ☒ อนุมัติให้สอบได้ ☐ ปรับปรุงเนื้อหาหรือรูปแบบใหม่ ลงชื่อ (ผู้ช่วยศาสตราจารย์ นายพิทักษ์ สถิตวรรธนะ)
--	---

สาขาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีราชมงคลศรีวิชัย

บทที่ 1

บทนำ

ในบทนี้จะกล่าวถึงความสำคัญและที่มาของหัวข้อโครงการรวมถึงวัตถุประสงค์ ประโยชน์ที่คาดว่าจะได้รับ ขอบเขตการทำงาน ขั้นตอนและวิธีการดำเนินการโครงการและแผนการดำเนินงานดังต่อไปนี้

1.1 ความสำคัญและที่มาของหัวข้อโครงการ

เนื่องด้วยการดำเนินชีวิตต่างๆของมนุษย์ในปัจจุบัน จำเป็นต้องพึ่งสิ่งอำนวยความสะดวกและยังจำเป็นต้องพึ่งเทคโนโลยีต่างๆ ในการดำเนินชีวิตหรือทำกิจกรรม ผู้จัดทำได้เล็งเห็นว่าหุ่นยนต์ เป็นนวัตกรรมหนึ่งที่น่าสนใจและยังมีบทบาทต่อชีวิตของมนุษย์มากมาย เช่น การอำนวยความสะดวกในด้านต่างๆ การช่วยทดแทนแรงงานมนุษย์ ช่วยลดความเสี่ยงต่อการเกิดอันตรายจากการทำงาน ลดเวลาในการทำงาน ด้วยเหตุนี้ทางผู้จัดทำจึงมีความสนใจในการศึกษาค้นคว้า และพัฒนารถเก็บขยะเคลื่อนที่ตามผู้ใช้งาน ซึ่งเป็นอุปกรณ์หนึ่งที่จะช่วยในการอำนวยความสะดวกต่อการทำงานๆของมนุษย์ในปัจจุบันได้

ในปัจจุบันเวลาทำงานในการเก็บขยะการที่ต้องเดินจนถึงขยะอาจเป็นปัญหาต่อการทำงานและอาจทำให้เกิดความวุ่นวายต่างๆ ซึ่งรถเก็บขยะเดินตามเคลื่อนที่อัตโนมัติที่ผู้จัดทำได้จัดทำขึ้นนี้ผู้จัดทำหวังว่าจะช่วยอำนวยความสะดวกให้กับผู้ใช้ ทั้งยังช่วยในการลดการใช้แรงงานของมนุษย์ ลดเวลาในการทำงาน ช่วยลดอุบัติเหตุต่างๆ ที่เกิดจากการทำงาน และยังสามารถนำไปประยุกต์ใช้เป็นอุปกรณ์อำนวยความสะดวกอื่นๆ ในชีวิตประจำวันของมนุษย์ได้

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อศึกษาความรู้ด้านการออกแบบระบบรถไร้สายเคลื่อนที่ตามผู้ใช้งาน
- 1.2.2 เพื่อสร้างเครื่องมือที่ช่วยทุ่นแรงในการทำงาน
- 1.2.3 เพื่อได้สร้างสิ่งประดิษฐ์ที่ใช้งานได้จริง
- 1.2.4 เพื่อลดระยะเวลาการทำงาน
- 1.2.5 เพื่อเพิ่มประสิทธิภาพในการทำงาน

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1.3.1 ได้ความรู้เกี่ยวกับการออกแบบระบบของรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน
- 1.3.2 ได้มีเครื่องมือในการทุ่มแรงในการทำงาน
- 1.3.3 ได้รถสิ่งประดิษฐ์ที่สามารถใช้งานได้จริง
- 1.3.4 ได้เครื่องมือลดเวลาในการทำงาน
- 1.3.5 ได้รถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานที่เพิ่มประสิทธิภาพในการทำงาน

1.4 ขอบเขตของโครงการ

- 1.4.1 ออกแบบและสร้างรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานจำนวน 1 คัน และรองรับน้ำหนักบรรทุกไม่น้อยกว่า 50 กิโลกรัม
- 1.4.2 มีแอปพลิเคชันควบคุมบังคับการเคลื่อนที่ได้
- 1.4.3 สามารถแสดงผลค่าระดับแรงดันแบตเตอรี่,แสดงระยะทาง และน้ำหนักบรรทุกได้
- 1.4.4 รถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานได้อัตโนมัติ
- 1.4.5 สามารถใช้งานได้ต่อเนื่องไม่น้อยกว่า 30 นาทีต่อการชาร์จ 1 ครั้ง

1.5 ขั้นตอนและวิธีการดำเนินการโครงการ

- 1.5.1 ค้นคว้าและศึกษาเกี่ยวกับงานวิจัยจากฐานข้อมูลด้านวิศวกรรมที่เกี่ยวข้อง
- 1.5.2 ทำการสังเคราะห์ข้อมูลเพื่อเป็นแนวทางในการทำโครงการ
- 1.5.3 ศึกษาการทำงานและทดลองหัดเขียนโปรแกรม Arduino IDE
- 1.5.4 ทำการออกแบบรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน โดยโปรแกรม Solid works
- 1.5.5 ดำเนินการสร้างรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน
- 1.5.6 ทดลองและบันทึกผลการทดลอง
- 1.5.7 จัดทำบทความและนำเสนองานประชุมวิชาการ
- 1.5.8 จัดทำรูปเล่มปริญญานิพนธ์
- 1.5.9 สอบป้องกันปริญญานิพนธ์
- 1.5.10 จัดแสดงผลงานโครงการ

บทที่ 2

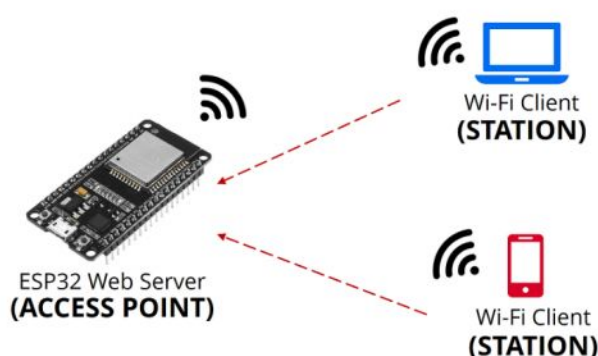
ทบทวนวรรณกรรมและทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงการทบทวนวรรณกรรมที่เกี่ยวข้อง ดังต่อไปนี้

2.1 ทบทวนวรรณกรรมที่เกี่ยวข้อง

หุ่นยนต์เคลื่อนที่อัตโนมัติโดยการใช้กล้องระบุตำแหน่ง, 2564 บทความนี้นำเสนอการระบุตำแหน่ง ซึ่งเป็นวิธีที่เชื่อถือได้ในการระบุตำแหน่งปัจจุบันของหุ่นยนต์โดยใช้การประมวลผลภาพซึ่งสามารถปรับปรุงการหลีกเลี่ยงการชนของสิ่งกีดขวางได้อย่างมีประสิทธิภาพด้วยเซ็นเซอร์ตรวจจับแสงและวัดระยะ (LiDAR) สำหรับหุ่นยนต์เคลื่อนที่อัตโนมัติ หุ่นยนต์พิจารณาห้าประการประการแรก เพื่อลดปัญหาที่เกิดจากการเลี้ยวเมื่อหุ่นยนต์ต้องการเปลี่ยนทิศทางการเคลื่อนที่ ได้มีการออกแบบหุ่นยนต์เคลื่อนที่โดยให้หุ่นยนต์สามารถเปลี่ยนทิศทางโดยไม่ต้องดึงเลี้ยวการเคลื่อนที่ ประการที่สอง มีการใช้เซ็นเซอร์ตรวจจับแสงและวัดระยะในการตรวจจับวัตถุที่อยู่ด้านหน้าหุ่นยนต์เคลื่อนที่ เพื่อให้หุ่นยนต์มีความสามารถในการหลีกเลี่ยงสิ่งกีดขวางได้ ประการที่สาม เสนอให้มีการวางแผนเส้นทางโดยใช้อัลกอริทึมในการเดินทางหลบหลีกสิ่งกีดขวางจากท่าเริ่มต้นไปยังปลายทางในสภาพแวดล้อมที่ไม่รู้จักได้ ประการที่สี่เพื่อปรับปรุงประสิทธิภาพการเคลื่อนที่ระบบสามารถประมวลผลแบบทันที โดยใช้การสื่อสารแบบไร้สายมีความยืดหยุ่นหุ่นยนต์สามารถเคลื่อนที่ได้อย่างอิสระมากกว่าใช้สาย สุดท้ายการปรับปรุงที่เป็นไปได้คือการเพิ่มกล้องมากกว่าหนึ่งตัว ซึ่งจะสามารถครอบคลุมการเคลื่อนที่ของหุ่นยนต์ได้กว้างมากขึ้น

ESP32 (AP Mode) สำหรับเว็บเซิร์ฟเวอร์ เมื่อดำเนินการ ESP32 เป็นจุดเชื่อมต่อ สร้างเครือข่าย Wi-Fi ของตัวเองและอุปกรณ์ Wi-Fi บริเวณใกล้เคียงสามารถเชื่อมต่อกับเครือข่ายนั้นได้เช่น สมาร์ทโฟนหรือคอมพิวเตอร์



ภาพที่ 2-1 ESP32 (AP Mode)

เซิร์ฟเวอร์ (Server) คือคอมพิวเตอร์หรือระบบคอมพิวเตอร์ที่ให้บริการ และจัดการทรัพยากรต่างๆ ให้กับคอมพิวเตอร์ หรืออุปกรณ์อื่นๆ ในเครือข่าย เซิร์ฟเวอร์สามารถให้บริการหลายประเภท เช่น การเก็บข้อมูล การประมวลผล การส่งข้อมูล และการจัดการทรัพยากรต่างๆ

การใช้งาน: ให้บริการเว็บไซต์และแอปพลิเคชันเว็บ ซึ่งเว็บเซิร์ฟเวอร์จะรับคำขอจากผู้ใช้ผ่านเว็บเบราว์เซอร์ และส่งข้อมูลกลับไปยังผู้ใช้ในรูปแบบของหน้าเว็บ โดยการเชื่อมต่อไวไฟ

UDP (User Datagram Protocol) เป็นโปรโตคอลหลักตัวหนึ่งในชุดโปรโตคอลอินเทอร์เน็ต ซึ่งถูกออกแบบมาเพื่อการส่งข้อมูลที่รวดเร็ว เช่น การเล่นเกมออนไลน์แบบสตรีมมิ่ง หรือการค้นหา DNS

ในการส่งข้อมูลผ่าน UDP คอมพิวเตอร์จะส่งข้อมูลขนาดเล็กที่เรียกว่า “ดาต้าแกรม” (datagram) ข้ามเครือข่ายไปยังปลายทาง โดยที่ UDP จะไม่รับประกันความน่าเชื่อถือหรือเรียงลำดับของดาต้าแกรม ซึ่งอาจทำให้แพ็คเกจสูญหายระหว่างทางได้ — จุดอ่อนตรงนี้สามารถถูกนำไปใช้ในการโจมตีแบบ DDoS (Distributed Denial-of-Service) ได้เช่นกัน

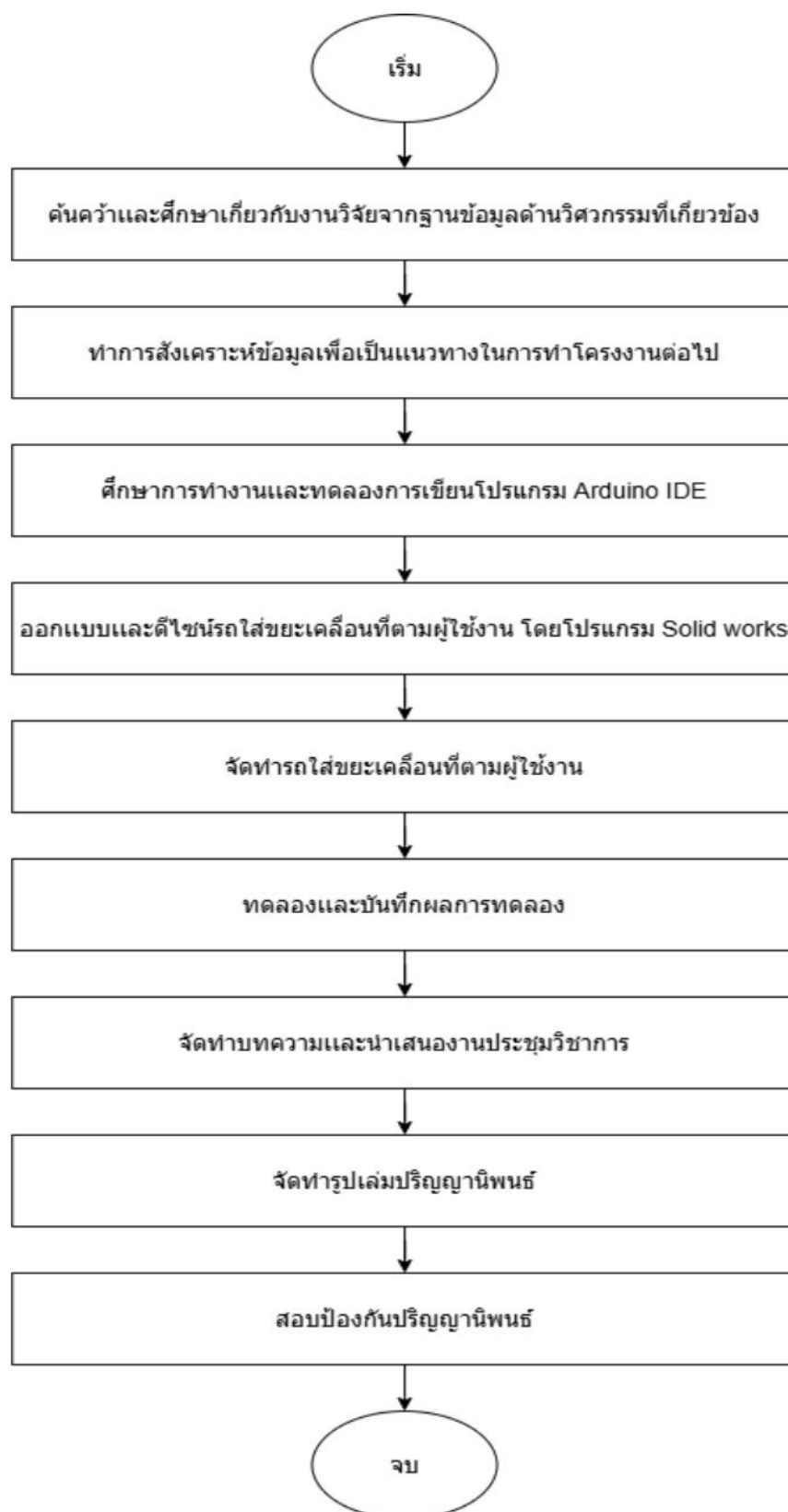
บทที่ 3

ขั้นตอนการดำเนินงาน

ในบทนี้จะกล่าวถึง การออกแบบรถไถขยะเคลื่อนที่ตามผู้ใช้งาน โดยใช้โปรแกรม Solid works หลักการเลือกอุปกรณ์ที่ใช้สำหรับการสร้างรถไถขยะเคลื่อนที่ตามผู้ใช้งาน การสร้างและประกอบโครงงาน วงจรการทำงานของโครงงาน ขั้นตอนการทำงานของรถไถขยะเคลื่อนที่ตามผู้ใช้งาน วิธีการทดลอง รวมไปถึงการสร้างรถไถขยะเคลื่อนที่ตามผู้ใช้งานเพื่อให้เป็นไปตามขอบเขตของโครงงานดังต่อไปนี้

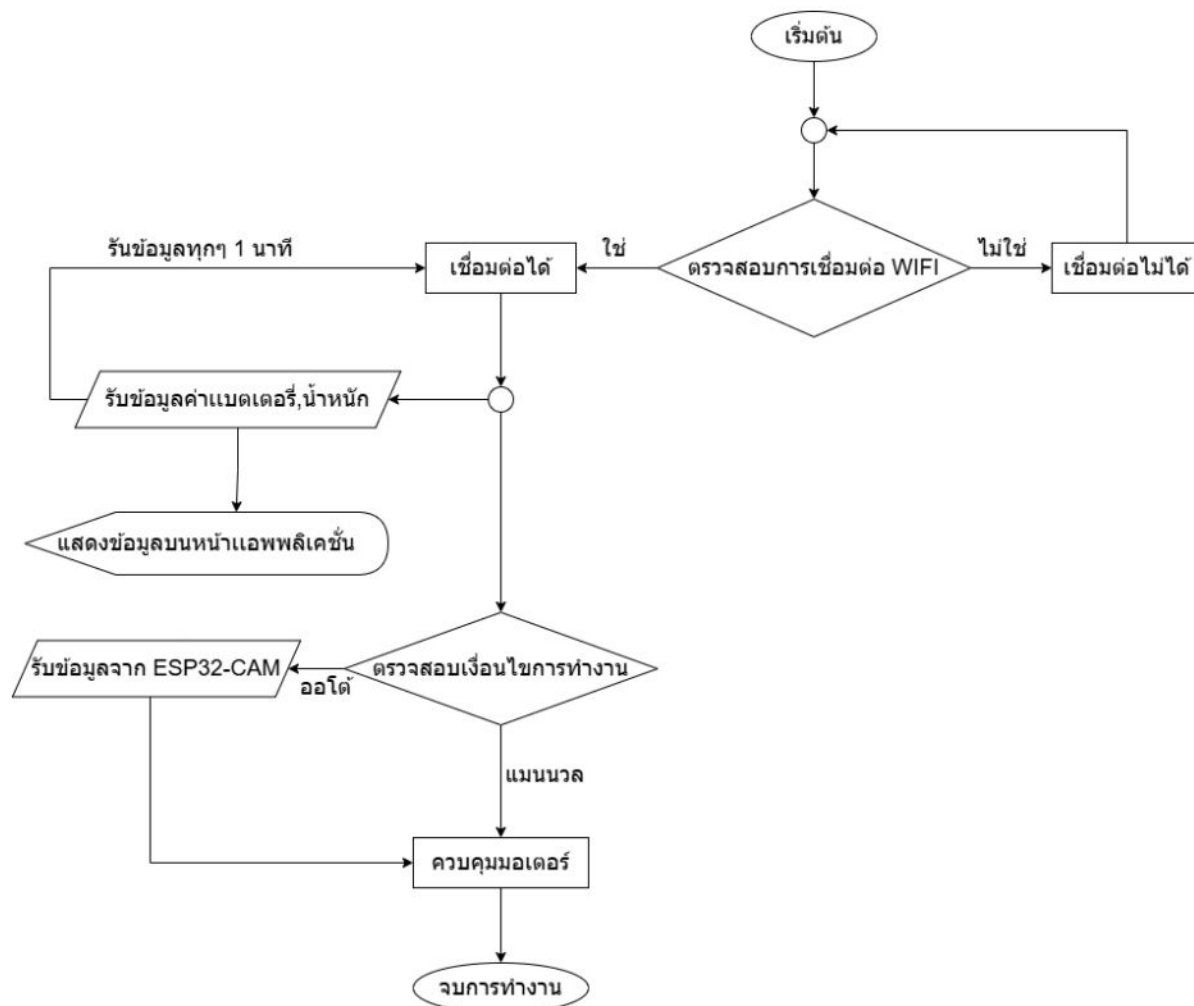
ในภาพที่ 3-1 สามารถอธิบายขั้นตอนการดำเนินโครงงานได้ดังนี้

- ขั้นตอนที่ 1 ค้นคว้าและศึกษาเกี่ยวกับงานวิจัยจากฐานข้อมูลด้านวิศวกรรมที่เกี่ยวข้อง
 - ขั้นตอนที่ 2 ทำการสังเคราะห์ข้อมูลเพื่อเป็นแนวทางในการทำโครงงานต่อไป
 - ขั้นตอนที่ 3 ศึกษาการทำงานและทดลองการเขียน โปรแกรม Arduino IDE
 - ขั้นตอนที่ 4 ออกแบบและดีไซน์รถไถขยะเคลื่อนที่ตามผู้ใช้งาน โดยโปรแกรม Solid works
 - ขั้นตอนที่ 5 จัดทำรถไถขยะเคลื่อนที่ตามผู้ใช้งาน
 - ขั้นตอนที่ 6 ทดลองและบันทึกผลการทดลอง
 - ขั้นตอนที่ 7 จัดทำบทความและนำเสนองานประชุมวิชาการ
 - ขั้นตอนที่ 8 จัดทำรูปเล่มปริญญานิพนธ์
 - ขั้นตอนที่ 9 สอบป้องกันปริญญานิพนธ์
- การดำเนินโครงงานสามารถเขียนแผนผังขั้นตอนการทำงานได้ดังแสดงในภาพที่ 3-1



ภาพที่ 3-1 แผนผังขั้นตอนการดำเนินโครงการ

3.1 ขั้นตอนการทำงานของรถไร้ขยะเคลื่อนที่ตามผู้ใช้งาน

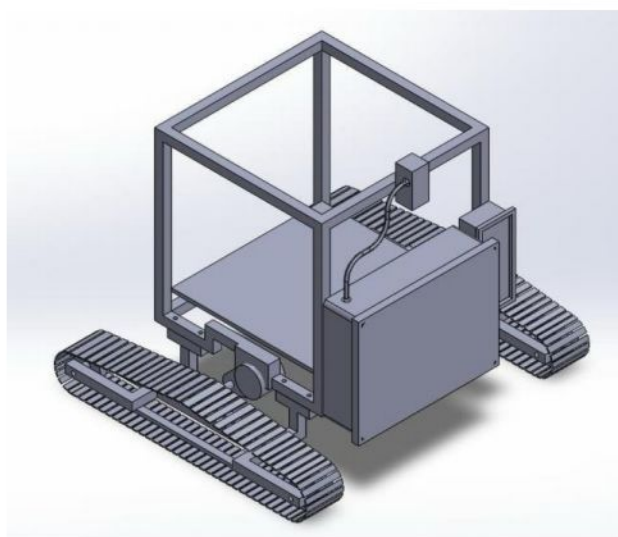


ภาพที่ 3-2 หลักการทำงานของรถไร้ขยะเคลื่อนที่ตามผู้ใช้งาน

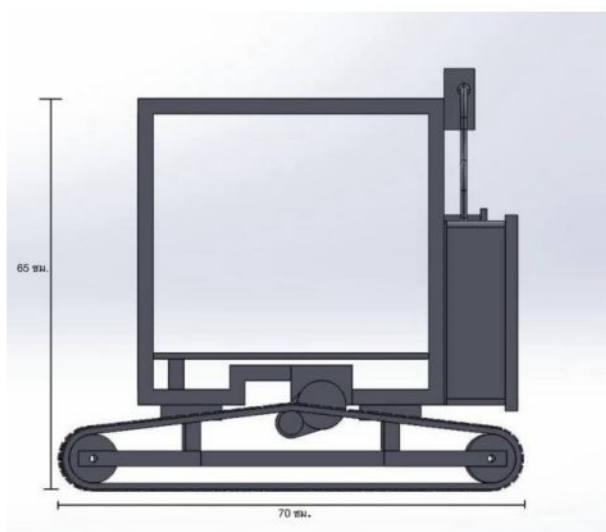
3.2 ออกแบบรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน

3.2.1 การออกแบบโครงสร้างชิ้นงาน

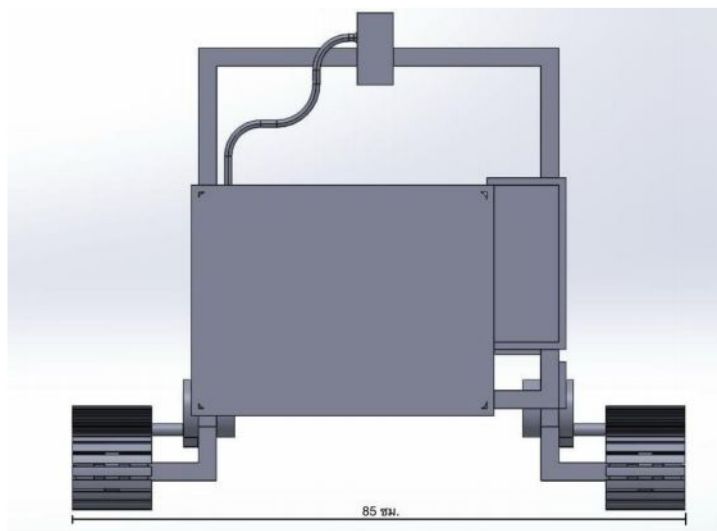
การออกแบบโครงสร้างรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน จะใช้โปรแกรม Solid works ในการออกแบบเพื่อที่จะนำไปเป็นแบบในการประยุกต์และจัดทำอุปกรณ์ใช้งานจริง ซึ่งผู้จัดทำได้ออกแบบซึ่งถอดแบบโครงสร้างมาจากถังขยะเทศบาลขนาด 60 ลิตร



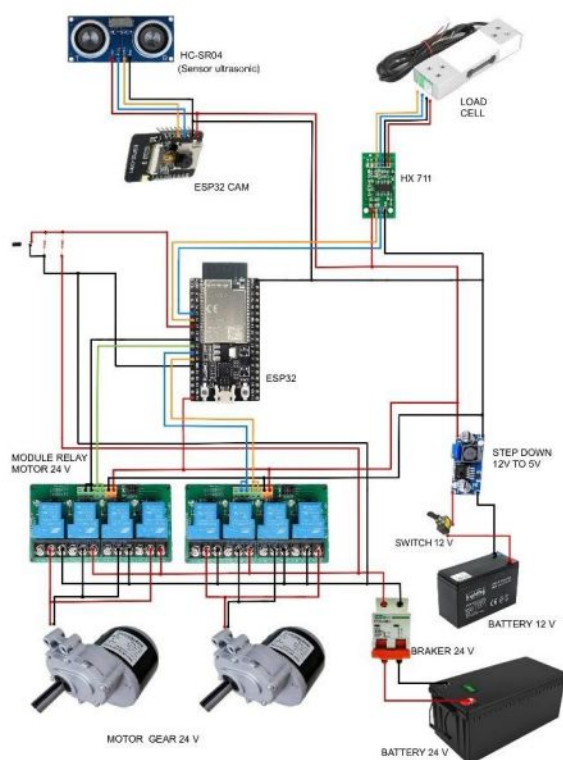
ภาพที่ 3-3 แบบจำลองรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน



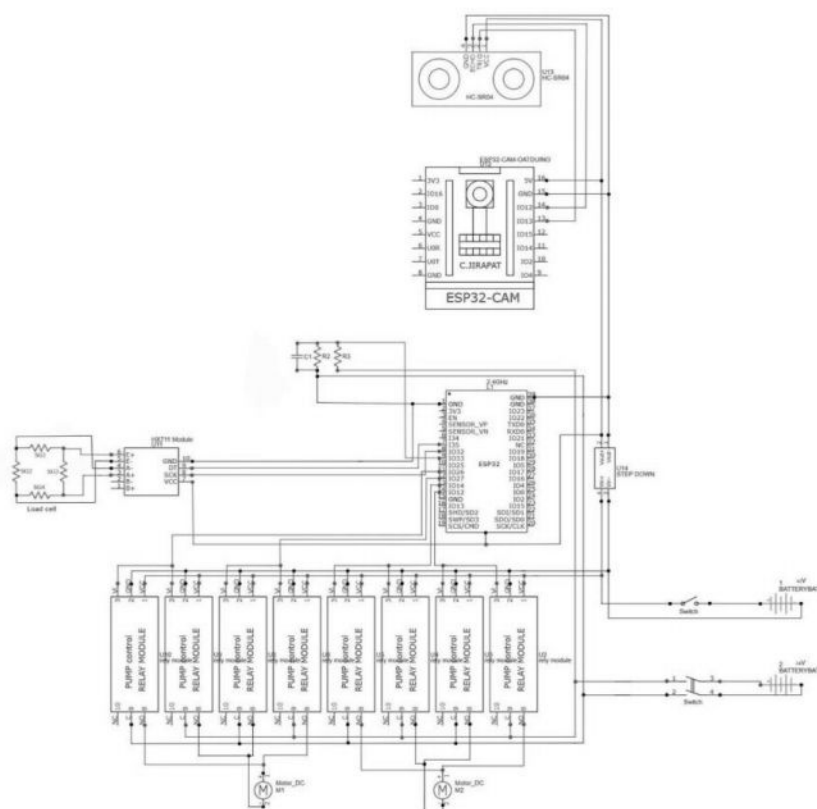
ภาพที่ 3-4 แบบจำลองด้านข้างรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน



ภาพที่ 3-5 แบบจำลองด้านหน้ารถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน



ภาพที่ 3-6 วงจรรวมของรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน



ภาพที่ 3-7 วงจรไฟฟ้า

3.3 หลักการเลือกอุปกรณ์ที่ใช้สำหรับการสร้างรถไร้สายระยะเคลื่อนที่ตามผู้ใช้งาน

3.3.1 ESP32

ESP32 เป็นชิปไมโครคอนโทรลเลอร์ที่มาพร้อม WiFi มาตรฐาน 802.11 b/g/n และบลูทูธเวอร์ชัน 4.2 เป็นรุ่นต่อ ยอดความล้ำเร้าของ ESP8266 โดยในรุ่นนี้ได้ออกมาแก้ไขข้อเสียของ ESP8266 ทั้งหมดโดย CPU ใช้สถาปัตยกรรม Tensilica LX6 จำนวน 2 คอร์ สัญญาณนาฬิกา 240MHz สามารถแยกการทำงานระหว่างโปรแกรมจัดการ WiFi และแอปพลิเคชันออกจากกันได้ ทำให้มีเสถียรภาพเพิ่มขึ้นมาก มีแรม 520KB มาในตัว นอกจากนี้ยังมี GPIO เพิ่มขึ้นมาก และมีช่อง ADC เพิ่มขึ้นเป็น 12 ช่อง จากเดิม ESP8266 มีเพียงช่องเดียว ใช้แรงดันไฟฟ้า 3.3V ในโหมด Sleep ใช้กระแสไฟฟ้าเพียง 2.5uA ผลิตโดยบริษัท Espressif จากประเทศจีน โดยตัวไอซี ESP32 มีสเปคโดยละเอียด ดังนี้

- ชิพใช้สถาปัตยกรรม Tensilica LX6 แบบ 2 แกนสมอง สัญญาณนาฬิกา 240MHz
- มีแรมในตัว 512KB

- รองรับการเชื่อมต่อรวมภายนอกสูงสุด 16MB

- มาพร้อมกับ WiFi มาตรฐาน 802.11 b/g/n รองรับการใช้งานทั้งในโหมด Station Soft AP และ Wi-Fi direct

- มีบลูทูธในตัว รองรับการใช้งานในโหมด 2.0 และโหมด 4.0 BLE

- ใช้แรงดันไฟฟ้าในการทำงาน 2.6V ถึง 3V

- ทำงานได้ที่อุณหภูมิ -40°C ถึง 125°C

ทำหน้าที่ เป็นตัวสื่อสารกับเว็บแอปผ่านทาง WIFI รับข้อมูลในการควบคุมการเปิด-ปิด Relay จากแอปพลิเคชัน ประมวลผลข้อมูลจากโมดูล HX711, การวัดแบตเตอรี่เพื่อส่งข้อมูลไปยังเว็บแอป และประมวลผลข้อมูลจาก ESP32-CAM เพื่อใช้ในการควบคุมเปิด-ปิด Relay



ภาพที่ 3-8 ESP32

(ที่มา : <https://www.artronshop.co.th/article/51/esp32->)

3.3.2 Relay 5V

โมดูลรีเลย์ ควบคุมเปิด-ปิดจ่ายไฟให้อุปกรณ์อิเล็กทรอนิกส์ พร้อม LED แสดงสถานะการทำงาน ใช้ไฟเลี้ยง 5V โดยสามารถเลือกโหมดทำงานแบบ Active High / Active Low โดยการสลับจัมเปอร์ ใช้ได้ทั้งไฟ DC และ AC ทนกระแสได้สูงสุด 30A

ทำหน้าที่ ควบคุมทิศทางของมอเตอร์ โดยใช้ Relay 5V 4 Channel ต่อมอเตอร์ 1 ตัว



ภาพที่ 3-9 Relay 5V

(ที่มา : <https://www.analogread.com/product/1915/4-channel-relay-5v-high-and-low-trigger-240v-30a-module>)

3.3.3 ESP32-CAM

ESP32-CAM เป็นโมดูลที่คุณสามารถใช้กับโครงการต่างๆมากมายและกับ Arduino เป็นโมดูลที่สมบูรณ์พร้อมด้วยไมโครคอนโทรลเลอร์ในตัวซึ่งสามารถทำให้ทำงานได้อย่างอิสระ นอกเหนือจากการเชื่อมต่อ WiFi + Bluetooth แล้วโมดูลนี้ยังมีกล้องวิดีโอในตัวและช่องเสียบ microSD สำหรับจัดเก็บข้อมูล

- ไมโครคอนโทรลเลอร์ ESP32: กับเขา CPU 32 บิตที่มีสองคอร์ชิปนี้มีความเร็วสัญญาณนาฬิกาสูงถึง 160 MHz นอกจากนี้ยังมาพร้อมกับ SRAM ขนาด 520 KB และ PSRAM ภายนอก 4 MB.
- เชื่อมต่อไร้สาย: ESP32-CAM มีทั้งสองอย่าง ไร้ไฟ 802.11b/g/n/c/I ในขณะที่ บลูทูธ 4.2 พร้อม BLE (บลูทูธพลังงานต่ำ) สิ่งนี้ทำให้คุณสามารถรวมเข้ากับเครือข่ายท้องถิ่นและควบคุมอุปกรณ์ไร้สายอื่นๆ ได้อย่างง่ายดาย

- กล้อง OV2640: มันคือ กล้อง 2 ล้านพิกเซล สามารถถ่ายภาพความละเอียดสูงสุด 1622x1200 พิกเซล ทำงานในรูปแบบ JPEG ซึ่งทำให้ง่ายต่อการบีบอัดภาพเพื่อส่ง
- พื้นที่เก็บข้อมูลเพิ่มเติม: ช่องเสียบไมโคร SD สามารถรองรับพื้นที่จัดเก็บข้อมูลได้สูงสุดถึง 4 GB ทำให้คุณสามารถจัดเก็บรูปภาพ วิดีโอ หรือข้อมูลอื่น ๆ ที่คุณต้องการบันทึกไว้ในเครื่องได้
- เสาอากาศ WiFi: แม้ว่า ESP32-CAM จะมี เสาอากาศ WiFi ในตัวคุณยังมีตัวเลือกในการเชื่อมต่อเสาอากาศ WiFi ภายนอกโดยใช้ตัวเชื่อมต่อ UFL ที่ปรับปรุงการรับสัญญาณในสภาพแวดล้อมที่มีความครอบคลุมต่ำ
ทำหน้าที่ ตรวจจับวัตถุที่กำหนดตามเงื่อนไขเพื่อส่งข้อมูลไปยัง ESP32 ใช้ในการควบคุม Relay



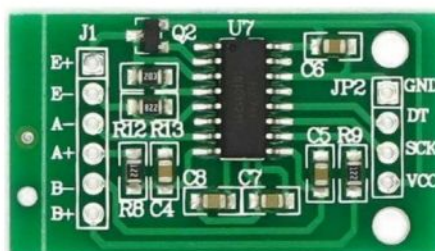
ภาพที่ 3-10 ESP32-CAM

(ที่มา : <https://th.hwlibre.com/>)

3.3.4 HX711

โมดูลขยายสัญญาณ HX711 คือโมดูลขยายสัญญาณจากโหลดเซลล์สำหรับส่งให้ Arduino เป็นสัญญาณแบบดิจิทัล 24-bit เป็นบอร์ดภาคขยายสัญญาณจากโหลดเซลล์ มีช่องอินพุตสำหรับกับ ต่อกับโหลดเซลล์ได้โดยตรง ใช้ไฟเลี้ยง 2.6-5.5 โวลต์

ทำหน้าที่ ขยายสัญญาณจากโหลดเซลล์สำหรับส่งข้อมูลให้กับ ESP32 ในการส่งข้อมูลไปยังเว็บแอปเพื่อแสดงผล



ภาพที่ 3-11 HX711

(ที่มา : <https://eresearch.rbru.ac.th/pdf-uploads/thesis-1807-file06-2020-09-29-11-12-28.pdf>)

3.3.5 Load cell

โหลดเซลล์ คืออุปกรณ์ที่ใช้ในการเปลี่ยนจากแรงหรือน้ำหนักที่กระทำต่อตัวโหลดเซลล์เป็นสัญญาณทางไฟฟ้า ซึ่งเราสามารถนำสัญญาณทางไฟฟ้านี้ไปจ่ายเข้าจอแสดงผล เพื่อแสดงค่าเป็นน้ำหนักหรือแรงที่กระทำ

โหลดเซลล์ทำมาจาก Strain Gauge ที่จัดเรียงวงจรในรูปแบบวงจรวิทสโตน บริดจ์ (Wheatstone Bridge) ซึ่งสามารถแปลงค่าแรงกด หรือแรงดึง ให้เป็นสัญญาณไฟฟ้าได้

ทำหน้าที่ ใช้การเปลี่ยนน้ำหนักที่กระทำต่อตัวโหลดเซลล์เป็นสัญญาณทางไฟฟ้าจากนั้นใช้ HX711 ในการขยายสัญญาณส่งไปยัง ESP32



ภาพที่ 3-12 Load cell

(ที่มา : <https://www.tic.co.th/th/news/detail/499/2>)

3.3.6 มอเตอร์เกียร์ 24V (Gear Motor)

คือ อุปกรณ์ระบบส่งกำลังที่ประกอบไปด้วยมอเตอร์ไฟฟ้าและชุดเกียร์ทดรอบ ทำหน้าที่แปลงพลังงานไฟฟ้าให้เป็นพลังงานกล นิยมใช้ในงานอุตสาหกรรมต่างๆ เช่น สายพานลำเลียงสินค้า เครื่องบรรจุภัณฑ์ ประตูอัตโนมัติ เครื่องจักรกลการเกษตร ระบบรางเลื่อน

หลักการทำงานของมอเตอร์เกียร์ อาศัยมอเตอร์ไฟฟ้าแปลงพลังงานไฟฟ้าเป็นพลังงานกล เพลาของมอเตอร์ไฟฟ้าหมุนส่งแรงบิดไปยังชุดเกียร์ทดรอบ ชุดเกียร์ทดรอบลดรอบการหมุนและเพิ่มแรงบิดของเพลาเอาต์พุต

ประเภทของมอเตอร์เกียร์ จำแนกตามประเภทของมอเตอร์ไฟฟ้า ชุดเกียร์ทดรอบ และโครงสร้าง

ข้อดี ของมอเตอร์เกียร์ คือ ควบคุมรอบและแรงบิดได้แม่นยำ แรงบิดสูง ทนทาน ใช้งานง่าย

ข้อเสีย ของมอเตอร์เกียร์ คือ ขนาดใหญ่ มีเสียงดัง ราคาแพง
ทำหน้าที่ ใช้ในการเคลื่อนที่ไปยังทิศทางตามเงื่อนไข



ภาพที่ 3-13 มอเตอร์เกียร์ 24V (Gear Motor)

(ที่มา : <https://highcontrol.co.th/>)

3.3.7 DC Step Down

วงจรลดแรงดันแบบ Step-Down หรือเรียกอีกแบบว่า Buck Converter (บัคคอนเวอร์เตอร์) ใช้ลดแรงดันจากแรงดันสูงให้ต่ำลง ใช้หลักการสวิตชิง-ตัวเหนี่ยวนำ(L) จึงทำให้มีความร้อนและความสูญเสียกำลังไฟน้อย ไม่เหมือนกับการลดแรงดันโดยใช้ IC ตระกูล 78xx / 317 ทั่วไปที่ใช้หลักการลดทอนทำให้เกิดความร้อนสูง วงจรบัคคอนเวอร์เตอร์เมื่อลดแรงดันลงแล้วจะได้กระแส Output เพิ่มขึ้น

ทำหน้าที่ ลดแรงดันจากแหล่งจ่ายให้เหลือ 5V เพื่อเป็นไฟเลี้ยงให้บอร์ดESP32,ESP32-CAM และ Relay



ภาพที่ 3-14 DC Step Down

(ที่มา : <https://www.arduitronics.com/product/570/dc-to-dc-step-down-lm2596-lm2596s-module-3a>)

3.3.8 Ultrasonic ranging module HC-SR04

โมดูลอัลตราโซนิกนี้เป็นอุปกรณ์ใช้วัดระยะทางโดยไม่ต้องมีการสัมผัสกับตำแหน่งที่ต้องการวัด วัดได้ตั้งแต่ 2 cm ถึง 400 cm โดยส่งสัญญาณอัลตราโซนิกความถี่ 40 kHz ไปที่วัตถุที่ต้องการวัดและรับสัญญาณที่สะท้อนกลับมา พร้อมทั้งจับเวลาเพื่อนำมาใช้ในการคำนวณระยะทาง

ทำหน้าที่ เมื่อใช้การเดินแบบอัตโนมัติจะจับระยะเพื่อให้หุ่นยนต์หยุดเดินเมื่อเข้าใกล้ผู้ใช้งาน



ภาพที่ 3-15 Ultrasonic ranging module HC-SR04

(ที่มา : <https://www.arduitronics.com/product/20/ultrasonic-sensor-module-hc-sr04-5v>)

3.3.9 แบตเตอรี่ลิเทียมไอออน (Lithium-Ion Battery)

คือแบตเตอรี่แบบชาร์จได้ที่อาศัยการเคลื่อนย้ายของไอออนลิเทียมระหว่างขั้วแคโทด (ขั้วบวก) และแอโนด (ขั้วลบ) เพื่อกักเก็บและจ่ายพลังงาน ด้วยจุดเด่นของลิเทียมที่มีน้ำหนักโมเลกุลน้อยและมีขนาดเล็ก เมื่อนำมาใช้ในแบตเตอรี่จึงทำให้มีความหนาแน่นของพลังงานสูง สามารถกักเก็บพลังงานได้มากกว่าแบตเตอรี่ชนิดอื่นๆ เมื่อเทียบกับน้ำหนักและปริมาตรเท่ากัน

ทำหน้าที่ เป็นแหล่งจ่ายให้กับวงจร โดยแบตเตอรี่ 24V เป็นแหล่งจ่ายให้กับมอเตอร์ 24V และแบตเตอรี่ 12V เป็นแหล่งจ่ายให้กับบอร์ดควบคุมโดยใช้ DC Step Down ในการลดแรงดันให้เหมาะสม



ภาพที่ 3-16 แบตเตอรี่ลิเทียมไอออน (Lithium-Ion Battery)

(ที่มา : <https://www.gmsolar.com/lithium-battery-comprehensive-guide/>)

แบตเตอรี่ (24V 40Ah) โดยใช้การคำนวณจากวัตต์(W) แรงดัน(V) ของมอเตอร์และนาฬิกาใช้งานที่ไม่น้อยกว่า(30 นาที)
ดังสมการ

$$\begin{aligned} A &= (W \times 2) \times 0.8 \times T \\ &= (355 \times 2) \times 0.8 \times 30 \\ mA &= 17040 \approx 17 A \end{aligned}$$

เมื่อ

A คือ กระแสไฟฟ้า

W คือ กำลังไฟฟ้า

0.8 คือ ประสิทธิภาพการจ่ายของกระแสไฟฟ้า

T คือ เวลา

หมายเหตุ เนื่องจากการคำนวณมอเตอร์ 355 W 2 ตัว ซึ่งต้องการใช้งานไม่น้อยกว่า 30 นาที จึงได้ความจุแบตเตอรี่ที่ต้องการคือ 40 A

3.3.10 สายไฟ THW(F)

คือ สายไฟฟ้าที่มีตัวนำไฟฟ้าเป็นทองแดง มีลักษณะเป็น เส้นฝอยขนาดเล็ก สายคอนโทลจึงเป็นสายไฟที่มีความอ่อนตัวสูง ช่วยให้การจัดสายให้เป็นระเบียบได้โดยง่าย ส่วนใหญ่จึงมักจะนิยมนำมาใช้เป็นสายไฟสำหรับเชื่อมต่อในตู้คอนโทลที่มีพื้นที่อย่างจำกัด หรือ ตู้สวิตช์บอร์ด หรือนำมาใช้เชื่อมต่อในงานอุตสาหกรรม



ภาพที่ 3-17 สายไฟ THW(F)

(ที่มา : <https://www.enscigroup.com/bangkok-cable-vs-f-thw-f-2/>)

การคำนวณขนาดสายไฟ

ดังสมการ

$$A = \frac{2 * I * L * p}{V_{drop}}$$

$$A = \frac{2 * 29.62 * 1.6 * 0.0175}{0.72}$$

$$A = \frac{1.65872}{0.72} \approx 2.30 \text{ mm}^2$$

A คือ ขนาดสายไฟ

I คือ กระแสไฟฟ้า

หากระแสไฟฟ้า จากสมการ

$$I = \frac{P}{V}$$

L คือ ระยะกระแสไฟ (ไป-กลับ)

p คือ ค่าความต้านทานจำเพาะของทองแดง

V_{drop} คือ แรงดันตกคร่อมที่ขอมรับได้ (3% ของพิกัดแรงดันแบตเตอรี่)

ดังนั้น จึงเลือกใช้สาย THW(F) ขนาดมาตรฐาน 2.5 mm²

3.3.11 เหล็กชุบกลวไนซ์ (Hot-Dip Galvanized Steel)

คือ เหล็กที่นำไปผ่านกระบวนการชุบกลวไนซ์ โดยนำเหล็กรูปพรรณไปชุบกลวไนซ์ หรือชุบสังกะสี ซึ่งเป็นกระบวนการเพื่อให้เหล็กป้องกันสนิมได้ดีขึ้น



ภาพที่ 3-18 เหล็กกล่องชุบกลวไนซ์ (Hot-Dip Galvanized Steel)

(ที่มา : <https://www.knbwatsadu.com>)

3.3.12 เสื้อเซฟตี้แถบสะท้อนแสง

คือ เสื้อที่ใช้ในการปฏิบัติงานเลือกใช้สีแดง เนื่องจากเป็นสีที่กล้องสามารถตรวจจับได้
แม่นยำเพราะเป็นสีที่เด่นชัด



ภาพที่ 3-19 เสื้อเซฟตี้แถบสะท้อนแสง

(ที่มา : <https://www.lazada.co.th/products/-i5251245570-s22305315806.html?>)

3.4 โปรแกรมการทำงานของรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งาน

3.4.1 ชุดการเขียนโปรแกรมการรับข้อมูลจากเว็บแอปที่ส่งผ่านเว็บ ใช้ในการควบคุมเปิด-ปิดรีเลย์เป็นการควบคุมแบบแมนนวล

```
<div class="control-wrapper">
  <div class="control-row">
    <button onclick="sendControlCommand('forward')" class="btn btn-forward">↑</button>
  </div>
  <div class="control-row">
    <button onclick="sendControlCommand('left')" class="btn btn-left">←</button>
    <button onclick="sendControlCommand('stop')" class="btn btn-stop">STOP</button>
    <button onclick="sendControlCommand('right')" class="btn btn-right">→</button>
  </div>
  <div class="control-row">
    <button onclick="sendControlCommand('backward')" class="btn btn-backward">↓</button>
  </div>
</div>
```

ภาพที่ 3-20 ชุดการเขียนโปรแกรมการรับข้อมูลจากเว็บแอปที่ส่งผ่านเว็บ

3.4.2 ชุดคำสั่งที่ใช้ข้อมูลที่รับมาทำตามเงื่อนไขที่กำหนด ใช้งานการเปิด-ปิดรีเลย์

```
void handleForward() {
  moveForward();
  lastCommand = "เดินหน้า";
  server.sendHeader("Location", "/");
  server.send(303);
}

void handleBackward() {
  moveBackward();
  lastCommand = "ถอยหลัง";
  server.sendHeader("Location", "/");
  server.send(303);
}

void handleLeft() {
  turnLeft();
  lastCommand = "เลี้ยวซ้าย";
  server.sendHeader("Location", "/");
  server.send(303);
}

void handleRight() {
  turnRight();
  lastCommand = "เลี้ยวขวา";
  server.sendHeader("Location", "/");
  server.send(303);
}

void handleStop() {
  stopMotors();
  lastCommand = "หยุด";
  server.sendHeader("Location", "/");
  server.send(303);
}
```

ภาพที่ 3-21 ชุดคำสั่งที่ใช้ข้อมูลที่รับมาทำตามเงื่อนไขที่กำหนด

3.4.3 ชุดคำสั่งการวัดแบตเตอรี่โดยใช้ ESP32 เนื่องจากแบตเตอรี่มีแรงดันสูงสุด 24V ซึ่ง ESP32 สามารถรองรับสูงสุด 3.3V จึงต้องลดแรงดันโดยใช้ตัวต้านทาน $R1 = 100k\Omega$ และ $R2 = 15k\Omega$

สูตร

$$V_{out} = V_{battery} * \frac{R2}{(R1+R2)}$$

เมื่อ

$V_{battery}$ = แรงดันแบตเตอรี่จริง (สูงสุด 24V)

V_{out} = แรงดันที่ ADC อ่านได้ (ต้องไม่เกิน 3.3V)

$R1$ = $100k\Omega$

$R2$ = $15k\Omega$

แทนค่า

$$\begin{aligned} V_{out} &= 24 * \frac{15k\Omega}{(100k\Omega + 15k\Omega)} \\ &= 24V \times 0.1304 \approx 3.13V \end{aligned}$$

กำหนดขาพินที่ใช้วัดแบตเตอรี่และค่าของตัวต้านทานที่ใช้ในสมการ,ค่า ADC อยู่ระหว่าง 0 –4095 (0V-3.3V)

```
#define BATTERY_PIN 34
#define R1 100000.0
#define R2 15000.0
#define ADC_MAX 4095.0
#define VOLTAGE_REF 3.3
```

ภาพที่ 3-22 กำหนดขาพินและค่าของสมการ

อ่านค่า ADC และคำนวณแรงดันจริง (อ่านค่า ADC 10 ครั้ง แล้วเฉลี่ย) แล้วนำมาคำนวณมาเป็นค่าจริง

สูตร

$$V_{\text{measured}} = \frac{\text{ค่าADC}}{4095} \cdot 3.3V$$

จากนั้นนำมาแปลงแรงดันเป็นเปอร์เซ็นต์แบตเตอรี่

```
float readBatteryVoltage() {
    const int samples = 10;
    long total = 0;
    for (int i = 0; i < samples; i++) {
        total += analogRead(BATTERY_PIN);
        delay(5);
    }
    float avgAdc = total / float(samples);
    float measuredVoltage = (avgAdc / ADC_MAX) * VOLTAGE_REF;
    return measuredVoltage / (R2 / (R1 + R2));
}

int getBatteryPercentage(float voltage) {
    if (voltage >= 24.0) return 100;
    if (voltage <= 5.0) return 0;
    return (int)(((voltage - 5.0) / (24.0 - 5.0)) * 100);
}
```

ภาพที่ 3-23 อ่านค่า ADC และคำนวณแรงดันจริง

3.4.4 ชุดคำสั่งการชั่งน้ำหนักโดยใช้ Load cell และ HX711 ชุดคำสั่งนี้มีการกำหนดค่าที่ใช้เชื่อมต่อกับ HX711 การกำหนดค่าเริ่มต้นให้เป็น 0
ค่า "44888.f" เป็นค่าคาลิเบรท ซึ่งต้องกำหนดจากการทดลองจริง
อ่านค่าน้ำหนัก 10 ครั้ง แล้วเฉลี่ยเพื่อให้แม่นยำขึ้น

```

scale.begin(DOUT, CLK);
Serial.println("Taring the scale...");
delay(2000);
scale.tare();
Serial.println("Scale tared.");
scale.set_scale(44888.f);
float weight = scale.is_ready() ? scale.get_units(10) : -1;

```

ภาพที่ 3-24 ชุดคำสั่งการชั่งน้ำหนัก

3.4.5 ชุดคำสั่งการส่งค่าน้ำหนักและแบตเตอรี่ไปแสดงหน้าเว็บแอป

```

void handleData() {
  String json = "{\"weight\":\"" + String(weight) +
    "\",\"batteryVoltage\":\"" + String(batteryVoltage) +
    "\",\"batteryPercent\":\"" + String(batteryPercent) + "\"}";
  server.send(200, "application/json", json);
}

```

ภาพที่ 3-25 การส่งข้อมูลแบบ JSON จาก ESP32 ไปยังเว็บ

```

<div class="data-panel">
  <div class="data-item">
    <span class="data-label">สถานะ UDP:</span>
    <span class="data-value" id="udpStatus">-</span>
  </div>
  <div class="data-item">
    <span class="data-label">น้ำหนัก:</span>
    <span class="data-value" id="weightValue">0.0 kg</span>
  </div>
  <div class="data-item">
    <span class="data-label">แบตเตอรี่:</span>
    <span class="data-value" id="batteryValue">0% (0.0V)</span>
  </div>
</div>
</body>
</html>
)=====;

```

ภาพที่ 3-26 การแสดงค่าสถานะ UDP, ค่าน้ำหนักและค่าแบตเตอรี่

3.4.6 การตรวจจับตำแหน่งสีของ ESP32-CAM ใช้ esp_camera_fb_get ดึงภาพจากกล้อง และวนลูปเช็คค่าพิกเซลทุกจุด แยกค่าสี RGB ออกจากค่าข้อมูล RGB565 ถ้าพบ สีแดงเด่นชัด ($r > 150$ และ $g, b < 100$) → บันทึกตำแหน่ง X

```
void detectColor() {
    camera_fb_t *fb = esp_camera_fb_get();
    if (!fb) {
        Serial.println("Camera failed");
        return;
    }

    int redThreshold = 150;
    int sumX = 0, count = 0;

    for (int y = 0; y < fb->height; y++) {
        for (int x = 0; x < fb->width; x++) {
            int pixelIndex = (y * fb->width + x) * 2;
            uint8_t r = (fb->buf[pixelIndex] & 0xF8);
            uint8_t g = ((fb->buf[pixelIndex] & 0x07) << 5) | ((fb->buf[pixelIndex + 1] & 0xE0) >> 3);
            uint8_t b = (fb->buf[pixelIndex + 1] & 0x1F) << 3;

            if (r > redThreshold && g < 100 && b < 100) {
                sumX += x;
                count++;
            }
        }
    }
}
```

ภาพที่ 3-27 การตรวจจับตำแหน่งสีของ ESP32-CAM

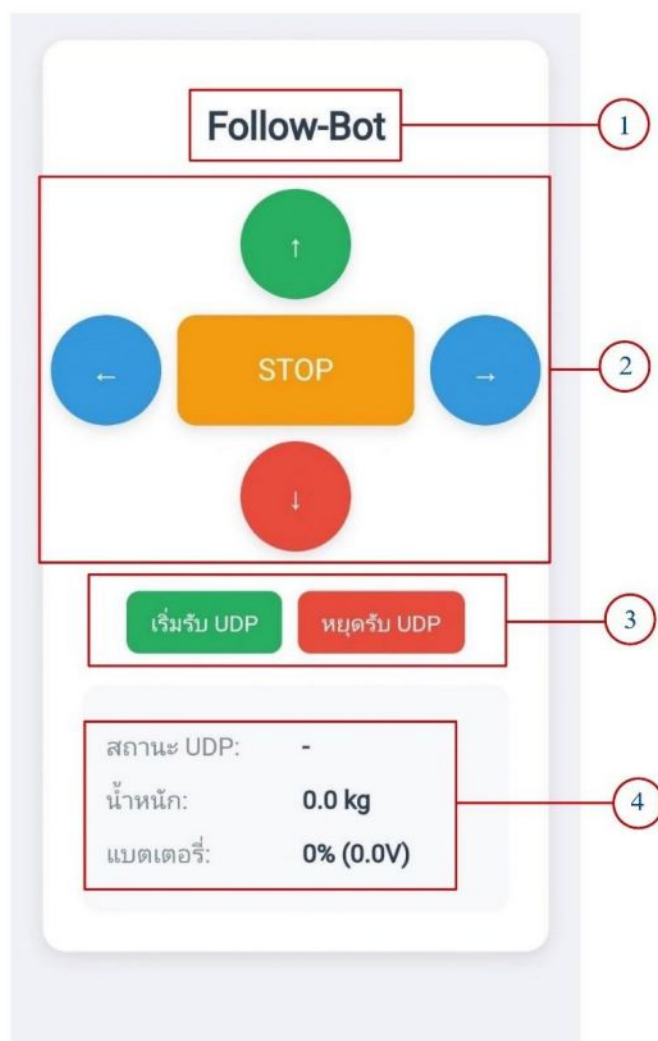
3.4.7 ตำแหน่งวัตถุสีแดง $\text{ซ้าย} = \text{avgX}$ น้อยกว่าหนึ่งในสามของความกว้าง, $\text{ขวา} = \text{avgX}$ มากกว่าสองในสามของความกว้าง, $\text{ตรงกลาง} = \text{อยู่ระหว่างนั้น}$

```
esp_camera_fb_return(fb);
if (count > 50) {
    int avgX = sumX / count;
    if (avgX < fb->width / 3) {
        dataToSend.direction = 1;
    } else if (avgX > (fb->width * 2 / 3)) {
        dataToSend.direction = 3;
    } else {
        dataToSend.direction = 2;
    }
} else {
    dataToSend.direction = 0;
}
}
```

ภาพที่ 3-28 ตำแหน่งวัตถุสีแดง

3.5 เว็บไซต์

3.5.1 หน้าเว็บไซต์



ภาพที่ 3-29 หน้าเว็บไซต์

1.ชื่อเว็บไซต์

2.ปุ่มการควบคุมแบบแมนนวล

3.ปุ่มเริ่ม/หยุดการทำงานแบบอัตโนมัติ

4.แสดงสถานะเริ่ม/หยุดการทำงานแบบอัตโนมัติ,ค่าน้ำหนักและค่าแบตเตอรี่

3.6 หลักการทำงานของรถไร้ขยะเคลื่อนที่ตามผู้ใช้งาน

หลักการทำงานของรถไร้ขยะเคลื่อนที่ตามผู้ใช้งาน เมื่อเปิดสวิทช์ที่ตัวรถและเชื่อมต่อ WIFI ของบอร์ด ESP32 ที่ตัวรถไร้ขยะเคลื่อนที่ตามผู้ใช้งานกับโทรศัพท์แล้วตัวรถจะสามารถเริ่มการทำงานได้ทันทีโดยจะมีค่าน้ำหนักและแบตเตอรี่ก็จะแสดงขึ้นบนหน้า WEB APP ทุกๆ 1 นาที และจะสามารถควบคุมตัวรถได้สองฟังก์ชัน คือ

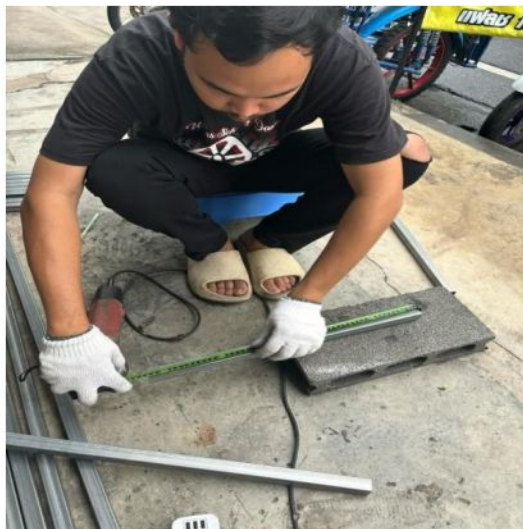
การควบคุมการเคลื่อนที่แบบแมนนวล โดยการควบคุมเคลื่อนที่แบบแมนนวลจะควบคุมผ่านการทำงานทางโทรศัพท์ผ่าน WEB APP โดยจะมีฟังก์ชันการคอนโทรลภายใน WEB APP ซึ่งจะมีปุ่มกดเคลื่อนที่ไปหน้า,เคลื่อนที่ไปด้านซ้าย,เคลื่อนที่ไปด้านขวา,เคลื่อนที่ถอยหลังและหยุด

การเคลื่อนที่แบบอัตโนมัติจะทำงานผ่าน WEB APP เช่นกันโดยเมื่อจะมีปุ่มเริ่มซึ่งจะเป็นการเริ่มการทำงานการเคลื่อนที่แบบอัตโนมัติ เมื่อกดปุ่มหยุดจะเป็นการควบคุมแบบแมนนวล

3.7 การสร้างและประกอบรถไร้ขยะเคลื่อนที่ตามผู้ใช้งาน

3.7.1 การสร้างโครงสร้างรถไร้ขยะตามผู้ใช้งาน

1) นำโครงสร้างที่ได้ออกแบบไว้ในโปรแกรม Solid works ซึ่งอ้างอิงจากถังขยะขนาด 60 ลิตร โดยได้ออกแบบจากการปรึกษาจากพี่สาวที่เรียนจบวิศวกรรมโยธาซึ่งได้คำแนะนำมาว่าการออกแบบโครงสร้างตัวรถที่ใช้เหล็กกล่อง และจะออกเพื่อให้เป็นรูปทรงสี่เหลี่ยม ให้ใช้เหล็กที่มีความหนา 1.4 mm ซึ่งการตัดแบ่งเหล็กเส้นเพื่อนำมาประกอบเป็นรูปทรงสี่เหลี่ยมในความยาวขนาดไม่เกิน 1 m. จะทำให้เหล็กรับแรงได้ดี เหล็กกล่องหนา 1.4 mm ความยาวไม่เกิน 1 m จะสามารถรับแรงกดได้มากกว่า 150 Kg ซึ่งเพียงพอต่อความต้องการ ทั้งชุดโครงสร้างตัวถังรับน้ำหนักและชุดระบบขับเคลื่อน หลังจากการออกแบบจึงได้นำมาถ่ายขนาดลงในชิ้นงานขึ้นจริง โดยเริ่มจากการถ่ายขนาดของโครงสร้าง โดยใช้เหล็กกล่องกัลวาไนซ์ ขนาด 1 นิ้ว*1 นิ้ว หนา 1.4 mm มาตัดแบ่งเป็นท่อนละ 50 cm นวน 12 ท่อน



ภาพที่ 3-30 วัดขนาดเหล็กกล่อง 1 นิ้ว*1 นิ้ว ขนาด 50 cm จำนวน 12 ท่อน

2) นำเหล็กกล่องที่วัดขนาดไว้มาตัดแบ่งเพื่อทำการเชื่อมขึ้นรูปเป็นกล่อง 4 เหลี่ยมจัตุรัสตามที่ได้ทำการออกแบบโครงสร้างไว้



ภาพที่ 3-31 ทำการตัดเหล็กโครงสร้างที่ได้วัดขนาดไว้

3) หลังจากนำเหล็กกล่องที่ได้ตัดแบ่งขนาดไว้ก่อนหน้ามาเชื่อมขึ้นรูปเป็นโครงสร้างตัวถังของตัวรถตามที่ออกแบบไว้ก่อนหน้านี้จะได้



ภาพที่ 3-32 นำเหล็กที่ตัดไว้มาเชื่อมจะได้โครงสร้างตัวถังของตัวรถ

4) สร้างชุดขับเคลื่อนซึ่งเป็นชุดขับเคลื่อนโดยใช้ล้อดินตะขาบตามที่ออกแบบไว้ โดยชุดขับเคลื่อนจะใช้เหล็กกล่องกัลวาไนซ์ขนาด 1 นิ้ว*1 นิ้ว มาตัดและเชื่อมตามแบบ ซึ่งจะออกแบบให้กว้างกว่าตัวรถ โดยมีขนาด ความยาวของล้อ 70 ซม



ภาพที่ 3-33 โครงสร้างของชุดขับเคลื่อนโดยใช้ดินตะขาบ

5) กลึงชุดโรลเลอร์และชุดเฟืองล้อที่ใช้ขับเคลื่อน และสั่งโซ่ซึ่งใช้เป็นสายพานในการขับเคลื่อนของชุดล้อดินตะขาบ



ภาพที่ 3-34 กลึงชุดโรลเลอร์และชุดเฟืองล้อขับเคลื่อน



ภาพที่ 3-35 โซ่สร้างซึ่งใช้เป็นสายพานขับเคลื่อนชุดล้อ



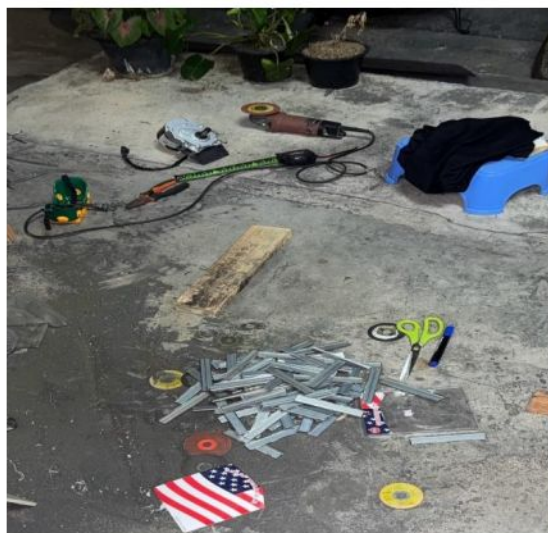
ภาพที่ 3-36 ชุดขับเคลื่อนที่ลองนำมาประกอบกัน

6) นำโครงสร้างชุดขับเคลื่อนมาประกอบกับโครงสร้างตัวถังเพื่อหาตำแหน่งการวางมอเตอร์เพื่อใช้เป็นตัวส่งกำลังในการขับเคลื่อนตัวรถ



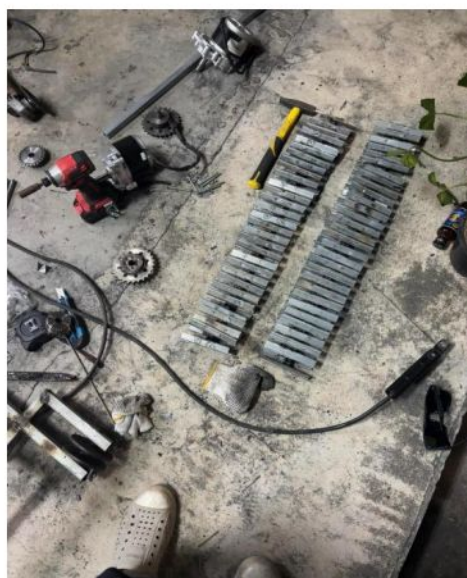
ภาพที่ 3-37 หาตำแหน่งวางมอเตอร์ซึ่งใช้เป็นตัวส่งกำลังขับเคลื่อน

7) สร้างไบแทรคซึ่งเป็นจุดที่ใช้เป็นหน้าสัมผัสของชุดล้อที่ใช้ในการขับเคลื่อนของล้อดินตะขาบ โดยการนำเหล็กกล่องขนาด 3 หุน มาผ่าแบ่งเป็นแผ่นมีขนาดความยาว 10.5 ซม



ภาพที่ 3-38 ชุดไบแทรคที่ได้จากการผ่าเหล็กกล่องขนาด 3 หุน

8) นำไบแทรคมาเชื่อมติดกับโซ่สร้างจะได้ความกว้างของล้อดินตะขาบ ข้างละ 10.5 ซมและยาว 72 ซม



ภาพที่ 3-39 ชุดไบแทรคที่นำมาเชื่อมติดกับโซ่สร้างจะได้ความกว้าง 10.5 ซม ยาว 72 ซม

9) นำชุดขับเคลื่อนกับตัวโครงสร้างมาประกอบกันเพื่อทดลองการขับเคลื่อนชุดล้อตีนตะขาบที่ได้ทำการสร้างขึ้น



ภาพที่ 3-40 ประกอบชุดขับเคลื่อนกับชุด โครงสร้างเข้าด้วยกันเพื่อทดลองการทำงานของชุดล้อ



ภาพที่ 3-41 ทดสอบการรับโหลดจริงของชุด โครงสร้างและชุดขับเคลื่อน

10) เริ่มประกอบชุดวงจรคอนโทรลของตัวรถและทดลองต่อใช้งานวงจร



ภาพที่ 3-42 ประกอบอุปกรณ์ต่างๆของวงจรคอนโทรล

11) นำชุดอุปกรณ์ของวงจรคอนโทรลขึ้นไปติดตั้งบนตัวรถ



ภาพที่ 3-43 นำตู้ที่ใช้ติดตั้งอุปกรณ์คอนโทรลติดตั้งกับตัวรถ

บทที่ 4

ผลการทดลองและวิจารณ์

สำหรับในบทนี้จะกล่าวถึงการทดลองและผลการทดลองใช้งานของชุดทดลองรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานที่ได้ออกแบบสร้างขึ้น ว่าสามารถทำงานได้ตามวัตถุประสงค์ที่กำหนดไว้หรือไม่ ซึ่งจะทำให้การทดลองใช้งานรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานและ โปรแกรมที่เขียนขึ้นมา

4.1 วิธีการทดสอบ

- 4.1.1 ทดสอบการควบคุมรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานแบบแมนนวล
- 4.1.2 ทดลองระยะสูงสุดที่ ESP32-CAM สามารถตรวจจับสีแดงที่ตัวผู้ใช้งานได้
- 4.1.3 ทดสอบการควบคุมรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานแบบอัตโนมัติ
- 4.1.4 ทดสอบน้ำหนักที่ตัวรถสามารถวัดค่าน้ำหนักได้
- 4.1.5 ทดสอบการวัดแบตเตอรี่
- 4.1.6 ทดสอบกระแสของมอเตอร์ขณะเคลื่อนที่แบบมีโหลด

4.2 ผลการทดสอบ

จากวิธีการทดสอบที่ 1-4 เป็นการทดสอบระบบเคลื่อนที่ของชุดทดลองรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานโดยมีการทดสอบระยะ แบบดังนี้

4.2.1 ทดสอบการควบคุมรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานแบบแมนนวล

- ✓ คือ สำเร็จ
- ✗ คือ ไม่สำเร็จ

จำนวนครั้งที่ ทดสอบ	1	2	3	4	5
เคลื่อนที่ไปข้างหน้า	✓	✓	✓	✓	✓
เคลื่อนที่ไปด้านซ้าย	✓	✓	✓	✓	✓
เคลื่อนที่ไปด้านขวา	✓	✓	✓	✓	✓
เคลื่อนที่ถอยหลัง	✓	✓	✓	✓	✓

ตารางที่ 4-1 ทดสอบการควบคุมรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานแบบแมนนวล

4.2.2 ระยะสูงสุดที่ ESP32-CAM สามารถตรวจจับสีแดงที่ตัวผู้ใช้งานได้

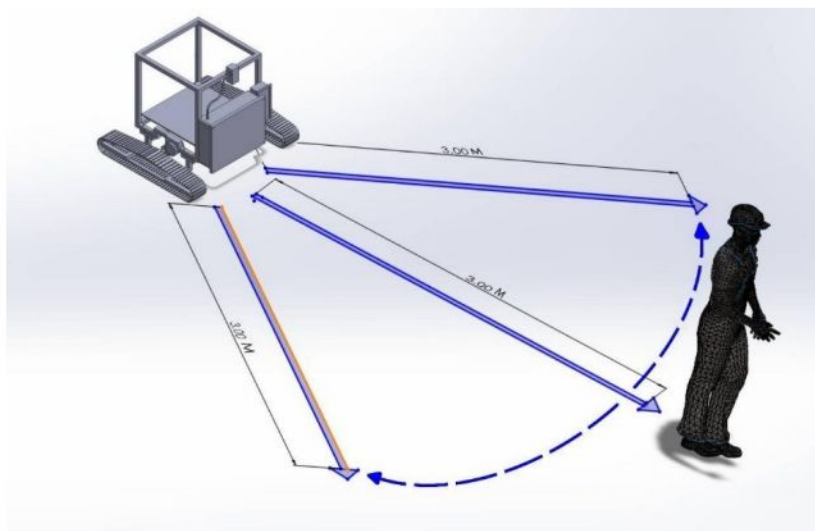
✓ คือ ESP32-CAM สามารถตรวจจับได้

✗ คือ ESP32-CAM ไม่สามารถตรวจจับได้

จำนวนครั้งที่ทดสอบ	1	2	3	4	5
การทดสอบระยะ 1 เมตร					
เคลื่อนที่ไปด้านหน้า	✓	✓	✓	✓	✓
เลี้ยวไปด้านซ้าย	✓	✓	✓	✓	✓
เลี้ยวไปด้านขวา	✓	✓	✓	✓	✓
การทดสอบระยะ 2 เมตร					
เคลื่อนที่ไปด้านหน้า	✓	✓	✓	✓	✓
เลี้ยวไปด้านซ้าย	✓	✓	✓	✓	✓
เลี้ยวไปด้านขวา	✓	✓	✓	✓	✓
การทดสอบระยะ 3 เมตร					
เคลื่อนที่ไปด้านหน้า	✓	✓	✓	✓	✓
เลี้ยวไปด้านซ้าย	✓	✓	✗	✓	✓
เลี้ยวไปด้านขวา	✓	✓	✓	✓	✗
การทดสอบระยะ 4 เมตร					
เคลื่อนที่ไปด้านหน้า	✓	✓	✓	✗	✓
เลี้ยวไปด้านซ้าย	✗	✓	✗	✓	✗
เลี้ยวไปด้านขวา	✓	✗	✓	✓	✗

หมายเหตุ การทดสอบขึ้นอยู่กับสภาพแวดล้อมและความเข้มข้นของแสงและความเร็วในการเดินของผู้ใช้งาน

ตารางที่ 4-2 ระยะสูงสุดที่ ESP32-CAM สามารถตรวจจับสีแดงที่ตัวผู้ใช้งานได้



ภาพที่ 4-1 ระยะสูงสุดที่ ESP32-CAM สามารถตรวจจับสีแดงที่ตัวผู้ใช้งานได้



ภาพที่ 4-2 การเคลื่อนที่ไปด้านซ้ายแบบอัตโนมัติ



ภาพที่ 4-3 การเคลื่อนที่ไปด้านขวาแบบอัตราโนมิติ



ภาพที่ 4-4 การเคลื่อนที่ไปด้านหน้าแบบอัตราโนมิติ

4.2.3 ทดสอบการควบคุมรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานแบบอัตโนมัติ

✓ คือ สำเร็จ

✗ คือ ไม่สำเร็จ

จำนวนครั้งที่ทดสอบ	1	2	3	4	5
เคลื่อนที่ไปข้างหน้า	✓	✓	✓	✓	✓
เคลื่อนที่ไปด้านซ้าย	✓	✓	✓	✓	✓
เคลื่อนที่ไปด้านขวา	✓	✓	✓	✓	✓

ตารางที่ 4-3 ทดสอบการควบคุมรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานแบบอัตโนมัติ

4.2.4 ทดสอบความแม่นยำของน้ำหนักที่ตัวรถสามารถวัดค่าได้

ชั่งน้ำหนักโดยใช้น้ำหนักของผู้จัดทำในการทดสอบ

จำนวนครั้งที่ทดสอบ	1	2	3
ตาชั่งดิจิตอล (กิโลกรัม)	71.90	71.90	71.90
ตาชั่งของตัวรถ (กิโลกรัม)	71.43	71.44	71.28

หมายเหตุ : การทดสอบมีความผิดพลาดไม่เกิน $\pm 0.7\%$

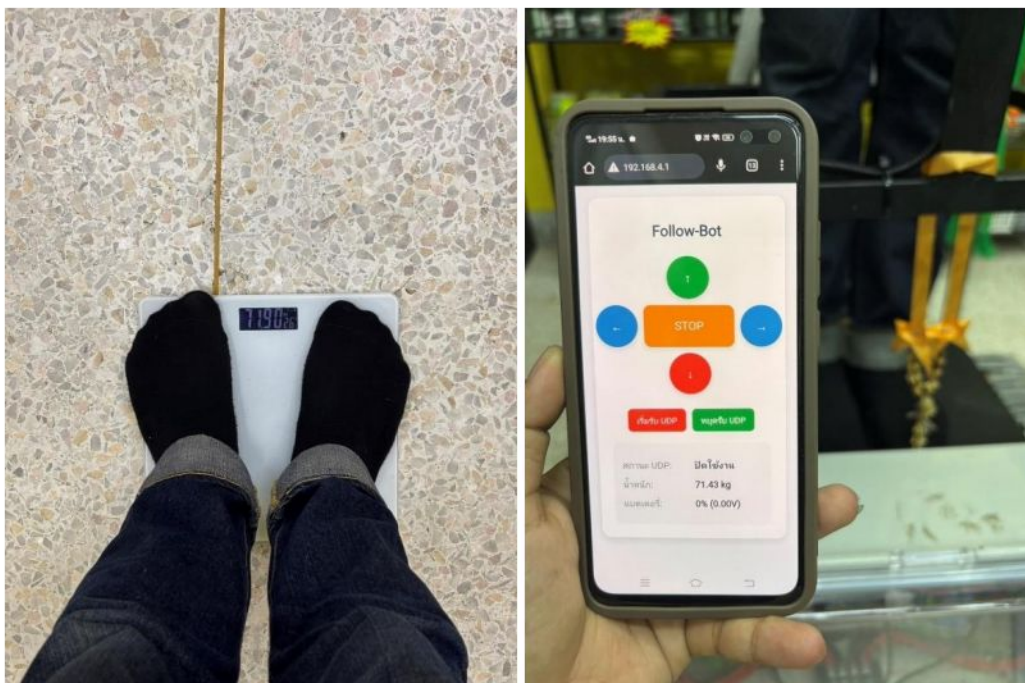
ตารางที่ 4-4 ทดสอบความแม่นยำของน้ำหนักที่ตัวรถสามารถวัดค่าได้

โดยใช้สูตร

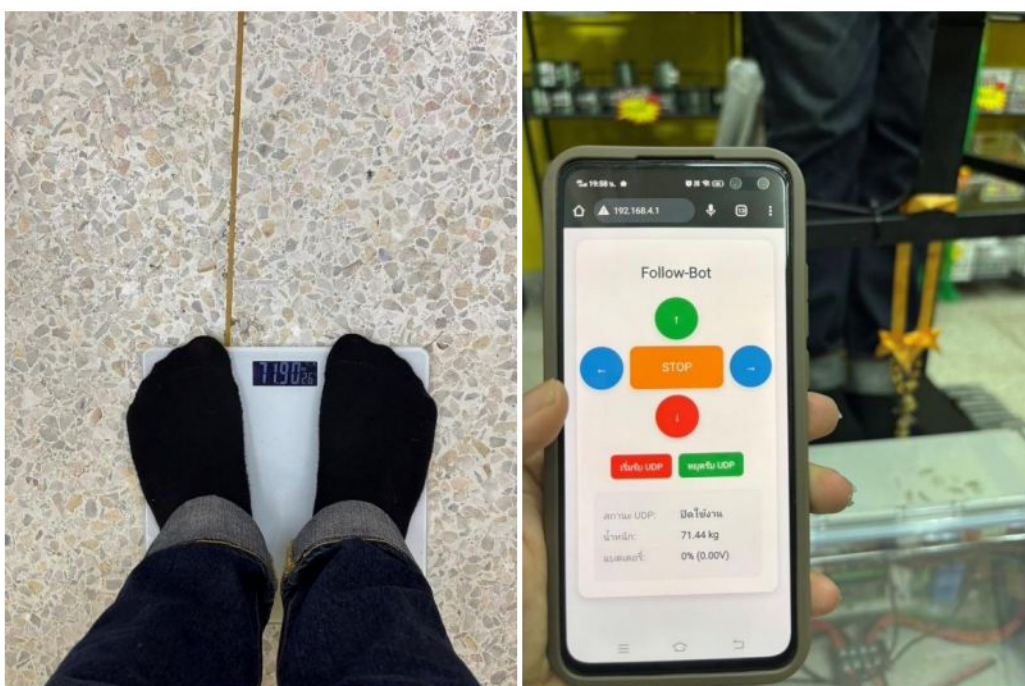
$$\% \text{ error} = \frac{\text{Exact Value} - \text{Approximate Value}}{\text{Exact Value}} * 100$$

Exact Value = ค่าที่แน่นอน

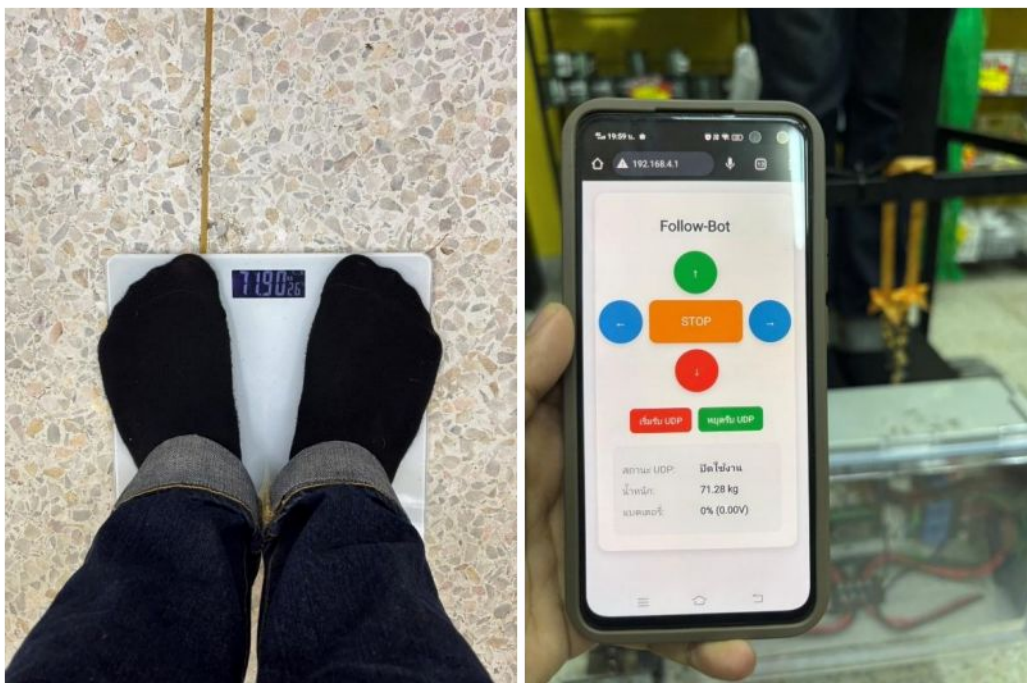
Approximate Value = ค่าที่ชั่งตัวรถ



ภาพที่ 4-5 การทดสอบชั่งน้ำหนักครั้งที่ 1



ภาพที่ 4-6 การทดสอบชั่งน้ำหนักครั้งที่ 2



ภาพที่ 4-7 การทดสอบชั่งน้ำหนักครั้งที่ 3

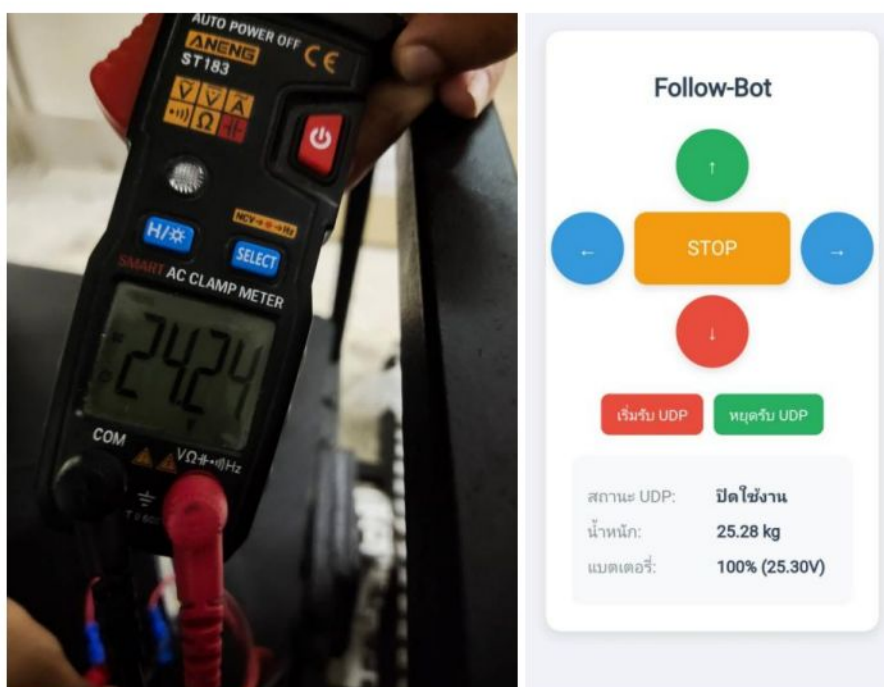
4.2.5 ทดสอบการวัดแบตเตอรี่

วัดแบตเตอรี่เทียบกับการวัดแบตเตอรี่โดยใช้มัลติมิเตอร์

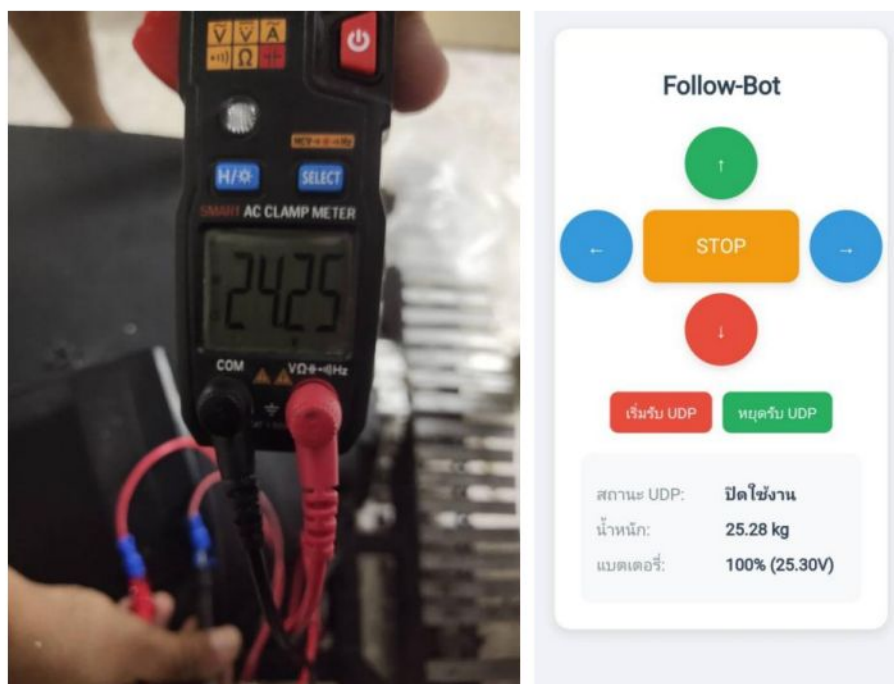
จำนวนครั้งที่ทดสอบ	1	2	3
วัดโดยมัลติมิเตอร์(โวลต์)	24.24	24.25	24.24
วัดโดยเว็บแอป(โวลต์)	25.30	25.30	25.30

หมายเหตุ : การทดสอบมีความผิดพลาดไม่เกิน $\pm 4\%$

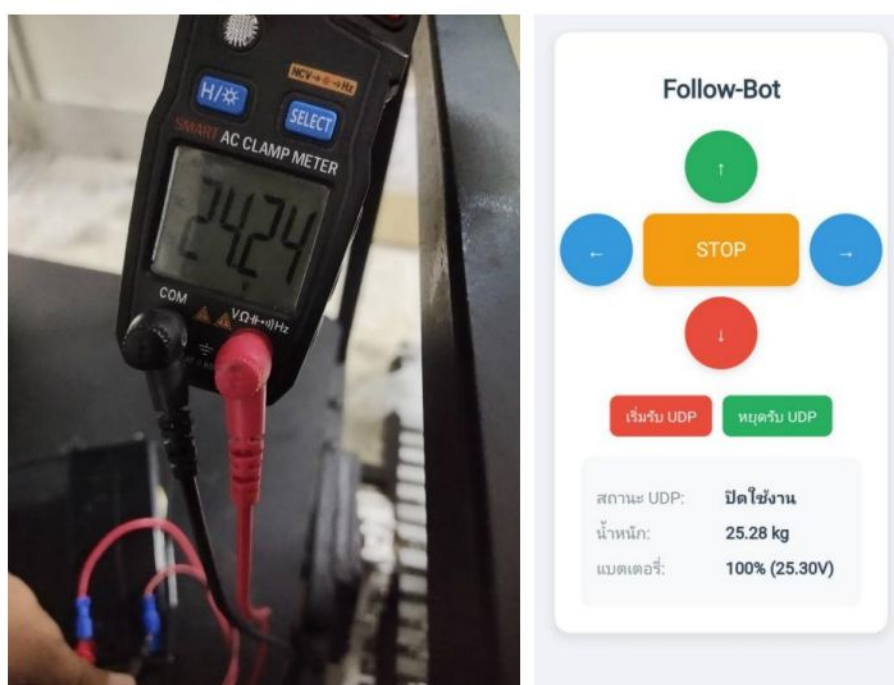
ตารางที่ 4-5 ทดสอบการวัดแบตเตอรี่



ภาพที่ 4-8 ทดสอบการวัดแบตเตอรี่ครั้งที่ 1



ภาพที่ 4-9 ทดสอบการวัดแบตเตอรี่ครั้งที่ 2



ภาพที่ 4-10 ทดสอบการวัดแบตเตอรี่ครั้งที่ 3

4.2.6 ทดสอบกระแสของมอเตอร์ขณะทำงาน

กระแสเคลื่อนที่ขณะไม่มีโหลด	กระแสเคลื่อนที่ขณะมีโหลด
7.50 A	14.81 A

ตารางที่ 4-6 คสอบกระแสของมอเตอร์ขณะทำงาน



ภาพที่ 4-11 ทดสอบกระแสขณะไม่มีโหลด



ภาพที่4-12 ทดสอบกระแสขณะมีโหลด

4.3 สรุปผลการทดลอง

จากการทดสอบวัดระยะสูงสุดที่ ESP32-CAM สามารถตรวจจับสีแดงที่ตัวผู้ใช้งานได้ มีระยะสูงสุดอยู่ที่ 3 เมตร

จากการทดสอบการควบคุมรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานแบบแมนนวล สามารถควบคุมรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานได้

จากการทดสอบการควบคุมรถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานแบบอัตโนมัติ รถไถ่ขยะเคลื่อนที่ตามผู้ใช้งานสามารถเคลื่อนที่ตามได้

จากการทดสอบน้ำหนักที่สามารถบรรทุกได้ สามารถบรรทุกน้ำหนักได้ไม่น้อยกว่า 50 กิโลกรัมตามขอบเขตที่วางไว้

จากการทดสอบวัดแบตเตอรี่ สามารถวัดแบตเตอรี่ได้

เอกสารอ้างอิงหรือบรรณานุกรม

[1] มทร.พระนคร.หุ่นยนต์เคลื่อนที่อัตโนมัติโดยการใช้กล้องระบุตำแหน่ง

จาก <http://journal.rmutp.ac.th>

[2] The Study of UAV-Photogrammetric Accuracy for Mapping Based on Variation of Ground Control Points

จาก https://www.researchgate.net/publication/325620020_Assessment_of_UAV-photogrammetric_mapping_accuracy_based_on_variation_of_ground_control_points

[[5] What is UDP?

จาก <https://thaiconfig.com/network/udp->

%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/

3] How to Set an ESP32 Access Point (AP) for Web Server

จาก <https://randomnerdtutorials.com/esp32-access-point-ap-web-server/>

[4] Web Server คืออะไร

จาก <https://www.pp->

ontime.co.th/%E0%B8%A3%E0%B8%B2%E0%B8%A2%E0%B8%A5%E0%B8%B0%E0%B9%80%E0%B8%AD%E0%B8%B5%E0%B8%A2%E0%B8%94/Web_Und_Server_Und__Und_%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3

งบประมาณ

ลำดับ	อุปกรณ์	จำนวน	ราคา
1	โครงสร้าง	1	5,900
2	ESP32	1	265
3	ESP32-CAM	1	300
4	HX711	1	70
5	Load cell	1	790
6	Motor	1	1,900
7	Step Down DC to DC	1	170
8	Battery 24V	1	1,540
9	กล่องกันน้ำ	1	750
10	Relay 5V 4CH	2	550
14	อื่นๆ	1	1200
รวม			13,987

หมายเหตุ : ราคาสินค้าอาจจะมีการเปลี่ยนแปลงได้ในอนาคตขึ้นอยู่กับความเหมาะสมของชิ้นงานและความเหมาะสมด้านการใช้งานที่ถูกต้อง*

ลงชื่อ..... (นักศึกษา)

(นายชุตินพงศ์ แก้วตั้ง)

ลงชื่อ..... (นักศึกษา)

(นายพงศกร สมพรหม)

ลงชื่อ..... (อาจารย์ที่ปรึกษา)

(ผู้ช่วยศาสตราจารย์ นายพิทักษ์ สติสุวรรณนะ)