การพัฒนาหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง

GESTURE BASED CONTROL OF MOBILE ROBOT NAVIGATION

นายพีระวุธ จันทนา รหัส 57363290

นายวิศรุต ธรรมบรรเทิง รหัส 57363474

นายศุภกิตต์ ยุติธรรมวรวาท รหัส 57363528

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2560

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อโครงงาน การพัฒนาหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง

ผู้ดำเนินโครงงาน นายพีระวุธ จันทนา รหัส 57363290

นายวิศรุต ธรรมบรรเทิง รหัส 57363474

นายศุภกิตต์ ยุติธรรมวรวาท รหัส 57363528

ที่ปรึกษาโครงงาน ผู้ช่วยศาสตราจารย์ ดร.พนัส นัถฤทธิ์

สาขาวิชา วิศวกรรมไฟฟ้า

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2560

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมไฟฟ้า

.....ที่ปรึกษาโครงงาน

(ผู้ช่วยศาสตราจารย์ ดร.พนัส นัถฤทธิ์)

.....กรรมการ

(รองศาสตราจารย์ ดร.สุชาติ แย้มเม่น)

.....กรรมการ

(รองศาสตราจารย์ ดร.สุวิทย์ กิระวิทยา)

หัวข้อโครงงาน	การพัฒนาหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง		
ผู้ดำเนินโครงงาน	นายพีระวุธ	จันทนา	รหัส 57363290
	นายวิศรุต	ธรรมบรรเทิง	รหัส 57363474
	นายศุภกิตต์	ยุติธรรมวรวาท	รหัส 57363528
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.พนัส นัถฤทธิ์		
สาขาวิชา	วิศวกรรมไฟฟ้า		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2560		

.....

บทคัดย่อ

โครงงานนี้เป็นการศึกษาและพัฒนาหุ่นยนต์สั่งงานด้วยท่าทาง ซึ่งพัฒนาขึ้นเพื่อให้เกิดการทำงานร่วมกันระหว่างหุ่นยนต์กับมนุษย์และยังสามารถนำไปพัฒนาต่อให้เป็นหุ่นยนต์สำหรับรองรับการใช้งานด้านบริการประเภทต่างๆ โดยหุ่นยนต์จะเคลื่อนที่จากการสั่งงานด้วยท่าทางโดยใช้กล้องอินฟราเรดตรวจจับได้ภาพเป็นโครงกระดูกและนำมาประมวลผลต่อที่เครื่องคอมพิวเตอร์ จากนั้นเครื่องคอมพิวเตอร์จะส่งคำสั่งไปยังไมโครคอนโทรลเลอร์เพื่อควบคุมทิศทางและความเร็วในการหมุนของมอเตอร์ตามที่ได้กำหนดเงื่อนไขไว้ โดยมีเซนเซอร์อัลทราโซนิกทำหน้าที่ตรวจจับวัตถุ หากพบสิ่งกีดขวางในระยะที่กำหนดหุ่นยนต์จะหยุดการเคลื่อนที่เพื่อป้องกันไม่ให้หุ่นยนต์เกิดความเสียหาย ผลการทดสอบระบบหุ่นยนต์ที่พัฒนาขึ้นแสดงให้เห็นว่าหุ่นยนต์สามารถทำงานตามคำสั่งได้อย่างปลอดภัยและมีประสิทธิภาพ

Project Title	Gesture based Control of Mobile Robot Navigation		
Name	Mr. Peerawut	Juntana	ID. 57363290
	Mr. Witsarut	Thambantheng	ID. 57363474
	Mr. Supakit	Yuttithamworawat	ID. 57363528
Project Advisor	Assistant Professor Panus Nattharith, Ph.D.		
Major	Electrical Engineering.		
Department	Electrical and Computer Engineering.		
Academic year	2017		

.....

Abstract

This project is to develop a gesture based control robot, for working with humans. The developed robot can be used in various types of services. The robot moves by gesture based control using infrared camera to detect skeletal image. The computer is also used to process, such data and sends command to the microcontroller to control the motor speed and rotation in the specified conditions. The ultrasonic sensors are used to detects objects, if any obstacle is encountered within the specified range, the robot then stops the movement to prevent itself from damaging. The experiment shows that the developed robot can operate efficiently, as it can successfully complete the tasks in a robust, safe, and speedy manner.

กิตติกรรมประกาศ

โครงงานวิศวกรรมฉบับนี้ สำเร็จลุล่วงไปได้ด้วยดีเนื่องจากความอนุเคราะห์จากอาจารย์ที่ปรึกษาโครงงานนี้คือ ผู้ช่วยศาสตราจารย์ ดร.พนัส นัถฤทธิ์ ซึ่งเป็นผู้ให้ความคิดริเริ่มในการพัฒนาโครงงานอีกทั้งยังเสียสละเวลาเพื่อช่วยเหลือและให้คำปรึกษา ในการทำวิทยานิพนธ์มาโดยตลอด

ขอขอบคุณ ว่าที่ร้อยตรีธานี โกสุม และ นายเผ่าพัฒน์ แสงอบ ที่คอยช่วยอำนวยความสะดวกในด้านอุปกรณ์เครื่องมือช่างและสถานที่ทำโครงงาน

ในโอกาสนี้ผู้จัดทำโครงงาน จึงขอขอบคุณทุกท่านที่ได้มีส่วนรวมให้โครงงานนี้ประสบความสำเร็จไปได้ด้วยดี สุดท้ายนี้ขอขอบคุณอาจารย์ทุกท่าน ที่คอยสั่งสอนให้ความรู้จนกระทั่งคณะผู้จัดทำสามารถสำเร็จการศึกษาไปได้ด้วยดี

ผู้ดำเนินโครงงาน

นายพีระวุธ จันทนา

นายวิศรุต ธรรมบรรเทิง

นายศุภกิตต์ ยุติธรรมวรวาท

สารบัญ

หน้า

ใบรับรองปริญญานิพนธ์ ..... ก
บทคัดย่อภาษาไทย ..... ข
บทคัดย่อภาษาอังกฤษ ..... ค
กิตติกรรมประกาศ ..... ง
สารบัญ ..... จ
สารบัญตาราง ..... ฌ
สารบัญรูป ..... ญ

บทที่ 1 บทนำ 1
1.1 ที่มาและความสำคัญของโครงงาน 1
1.2 จุดประสงค์ของโครงงาน 1
1.3 ขอบเขตของโครงงาน 2
1.4 ขั้นตอนการดำเนินงาน 2
1.5 แผนการดำเนินงาน 2
1.6 ผลที่คาดว่าจะได้รับ 3
1.7 งบประมาณของโครงงาน 3

บทที่ 2 หลักการและทฤษฎี 4
2.1 ทฤษฎีพื้นฐานและหลักการทำงานของหุ่นยนต์เคลื่อนที่ 4
2.2 การเคลื่อนที่โดยใช้ล้อประเภท Omnidirectional 5
2.3 ทฤษฎีและหลักการทำงานของ Kinect 8
2.4 ระบบประมวลผลหลัก (Host computer) 10
2.5 หลักการควบคุมการทำงานของมอเตอร์ไฟฟ้า 11
2.5.1 การควบคุมความเร็วของมอเตอร์โดยวิธีการมอดูเลตทางความกว้างของพัลส์ ... 12
2.5.2 การควบคุมความเร็วของมอเตอร์โดยการป้อนกลับสัญญาณ 13

สารบัญ (ต่อ)

หน้า

2.6 ทฤษฎีของวงจรไดร์เวอร์มอเตอร์ 13
2.7 ไมโครคอนโทรลเลอร์ (Microcontroller) 15
2.8 หลักการทำงานของเซนเซอร์อัลทราโซนิก (Ultrasonic Sensors) 16
2.9 อุปกรณ์ไฟฟ้าสำรอง (Uninterruptible Power Supply) 17
บทที่ 3 วิธีการดำเนินโครงงาน 19
3.1 การออกแบบภาพรวมการทำงานของระบบ 19
3.2 การออกแบบโครงสร้างของหุ่นยนต์และการติดตั้งอุปกรณ์ 20
3.3 การออกแบบการเชื่อมต่ออุปกรณ์ 22
3.4 การออกแบบและพัฒนาวิธีการเลือกรับคำสั่งของหุ่นยนต์ 27
3.4.1 สั่งงานด้วยการกดปุ่มบน Joystick 28
3.4.2 สั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์ (Kinect on Robot) 28
3.4.3 สั่งงานระยะไกลด้วยท่าทาง (Kinect on Remote PC) 28
3.5 การออกแบบและพัฒนาการเปลี่ยนสถานะของหุ่นยนต์ 28
3.5.1 สถานะรอ (Standby) 29
3.5.2 สถานะรับคำสั่ง (Controlled) 29
3.5.3 สถานะติดตาม (Follow) 30
3.6 การศึกษาและพัฒนาโปรแกรมการตัดสินท่าทาง 30
3.6.1 สั่งงานให้หุ่นยนต์เข้าสู่สถานะรับคำสั่ง 31
3.6.2 สั่งงานให้หุ่นยนต์เข้าสู่สถานะติดตาม 31
3.6.3 สั่งงานให้หุ่นยนต์เข้าสู่สถานะรอ 32
3.6.4 สั่งงานให้หุ่นยนต์เดินหน้า 32
3.6.5 สั่งงานให้หุ่นยนต์ถอยหลัง 33
3.6.6 สั่งงานให้หุ่นยนต์ไปทางซ้าย 33
3.6.7 สั่งงานให้หุ่นยนต์ไปทางขวา 34
3.6.8 สั่งงานให้หุ่นยนต์หมุนซ้าย 34

สารบัญ (ต่อ)

หน้า

3.6.9 สั่งงานให้หุ่นยนต์หมุนขวา 34
3.6.10 สั่งงานให้หุ่นยนต์หยุดอยู่กับที่ 35
3.7 การศึกษาและพัฒนาโปรแกรมการเคลื่อนที่ของหุ่นยนต์ 37
3.7.1 หุ่นยนต์สามารถเคลื่อนที่ด้วยความเร็วคงที่ได้ 38
3.7.2 หุ่นยนต์สามารถเคลื่อนที่ไปทุกทิศทางโดยที่ด้านหน้าของหุ่นยนต์ไม่เปลี่ยนทิศ 39
3.8 การศึกษาและพัฒนาโปรแกรมการตรวจจับสิ่งกีดขวาง 39
3.9 การทดสอบการทำงานของหุ่นยนต์ 40
3.10 อัลกอริทึมการทำงานของหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง 40
3.10.1 อัลกอริทึมการทำงานของโปรแกรมสำหรับส่วนที่ใช้คอมพิวเตอร์ควบคุม 40
3.10.2 อัลกอริทึมการทำงานของโปรแกรมสำหรับส่วนไมโครคอนโทรลเลอร์ 41
3.11 การใช้งานหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง 45
3.11.1 การเริ่มต้นใช้งานหุ่นยนต์ 45
3.11.2 หน้าจอแสดงผลของโปรแกรม 46
3.11.3 การกำหนดช่องทางการรับส่งข้อมูลผ่าน Wi-Fi 48
3.11.4 การสั่งงานหุ่นยนต์ด้วยวิธีการตามที่ผู้ใช้งานเลือกใช้ 49

บทที่ 4 การทดสอบการทำงานของระบบที่พัฒนาขึ้น 54
4.1 การทดสอบความถูกต้องของการสั่งงานด้วยท่าทาง 54
4.2 การทดสอบความแม่นยำของการตรวจจับระยะของเซนเซอร์อัลทราโซนิก 58
4.2.1 ระยะที่เซนเซอร์อัลทราโซนิกสามารถวัดได้ (Sensor Range) 58
4.2.2 ขนาดของลำคลื่นของเซนเซอร์อัลทราโซนิก (Sensor Beam) 65
4.3 การทดสอบการเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนด 68

บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ 71
5.1 สรุปผลการดำเนินงาน 71
5.1.1 ความถูกต้องในการตัดสินท่าทางการสั่งงาน 71

สารบัญ (ต่อ)

หน้า

5.1.2 ระยะที่เซนเซอร์อัลทราโซนิกวัดได้และลำคลื่นของเซนเซอร์อัลทราโซนิก 71
5.1.3 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนด 72
5.2 ปัญหาและอุปสรรค 72
5.3 แนวทางในการพัฒนา 72

เอกสารอ้างอิง 74
ภาคผนวก ก รายละเอียดของอุปกรณ์ 75
ภาคผนวก ข การเชื่อมต่ออุปกรณ์ทั้งหมด 78
ภาคผนวก ค โปรแกรมการทำงานของหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง 80
ประวัติผู้ดำเนินโครงงาน 82

สารบัญตาราง

ตารางที่ หน้า

1.1 ตารางแสดงระยะเวลาการดำเนินงาน 2

2.1 ข้อมูลจำเพาะของ Kinect Xbox ONE 8

2.2 ข้อมูลจำเพาะทางเทคนิคของ Mini PC 11

2.3 การควบคุมการทำงานของไดร์เวอร์มอเตอร์ 15

3.1 พินของบอร์ด Arduino Due ที่ใช้งานเพื่อส่งค่า 23

3.2 พินของบอร์ด Arduino Due ที่ใช้งานเพื่อรับค่า 23

3.3 การเคลื่อนที่ของหุ่นยนต์เพื่อรักษามุมระหว่างตัวหุ่นยนต์กับผู้สั่งงาน 30

3.4 การเคลื่อนที่ของหุ่นยนต์เพื่อรักษาระยะห่างระหว่างตัวหุ่นยนต์กับผู้สั่งงาน 30

3.5 ตัวแปรค่าพิกัดของจุดที่นำมาใช้เพื่อตรวจสอบเงื่อนไขท่าทาง 36

3.6 เงื่อนไขท่าทางคำสั่งสำหรับเปลี่ยนสถานะการทำงาน 36

3.7 เงื่อนไขท่าทางคำสั่งสำหรับควบคุมการเคลื่อนที่ 37

3.8 การตอบสนองของบอร์ด Arduino Due ต่อรหัสที่ได้รับจากคอมพิวเตอร์ 38

3.9 การตอบสนองของบอร์ด Arduino Due ต่อรหัสการเคลื่อนที่ 38

3.10 ความสอดคล้องระหว่างการเคลื่อนที่ของหุ่นยนต์กับทิศทางการหมุนของมอเตอร์ 39

3.11 เงื่อนไขการตัดสินใจของหุ่นยนต์เมื่อตรวจพบสิ่งกีดขวาง 39

3.12 การใช้งาน Joystick เพื่อสั่งงานหุ่นยนต์ 50

4.1 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงมากและอยู่ในระยะใกล้ 54

4.2 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงมากและอยู่ในระยะปานกลาง 55

4.3 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงมากและอยู่ในระยะไกล 56

4.4 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงปกติและอยู่ในระยะใกล้ 56

4.5 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงปกติและอยู่ในระยะปานกลาง 57

4.6 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงปกติและอยู่ในระยะไกล 58

4.7 ความแม่นยำของการวัดระยะของเซนเซอร์อัลทราโซนิกตัวที่ 1 60

4.8 ความแม่นยำของการวัดระยะของเซนเซอร์อัลทราโซนิกตัวที่ 2 60

4.9 ความแม่นยำของการวัดระยะของเซนเซอร์อัลทราโซนิกตัวที่ 3 61

4.10 ความแม่นยำของการวัดระยะของเซนเซอร์อัลทราโซนิกตัวที่ 4 61

สารบัญรูป

รูปที่ หน้า

2.1 ล้อประเภท Omni-directional 5

2.2 ทิศทางการเคลื่อนที่แบบรวมเวกเตอร์ 6

2.3 Kinect Xbox ONE 8

2.4 แผนภาพการติดต่อระหว่าง Kinect กับแอปพลิเคชัน 9

2.5 ส่วนประกอบของ Software Development Kit (SDK) 9

2.6 จุดข้อต่อทั้ง 25 จุดของร่างกายที่ Kinect สามารถระบุพิกัดได้ 10

2.7 คอมพิวเตอร์ส่วนบุคคลขนาดเล็ก 10

2.8 มอเตอร์ไฟฟ้ากระแสตรง 11

2.9 Duty Cycle 12

2.10 การทำงานของเอนโค้ดเดอร์ 13

2.11 หลักการทำงานของวงจร H-Bridge สำหรับควบคุมการหมุนของ DC Motor 14

2.12 บอร์ดขับมอเตอร์ดีซีแบบ H-Bridge 14

2.13 Arduino Due 15

2.14 Arduino Uno R3 15

2.15 Joystick PS2 wireless 16

2.16 เซนเซอร์อัลทราโซนิก 16

2.17 อุปกรณ์ไฟฟ้าสำรองที่เลือกใช้ 18

3.1 การทำงานโดยรวมของระบบทั้งหมด 19

3.2 โครงสร้างของหุ่นยนต์ซึ่งได้ออกแบบไว้ 20

3.3 ตำแหน่งของอุปกรณ์ต่างๆ ที่ติดตั้งลงบนโครงสร้างของหุ่นยนต์ 20

3.4 ตำแหน่งของเซนเซอร์อัลทราโซนิกทั้งสี่ตัว 21

3.5 ตำแหน่งของมอเตอร์และล้อทั้งสี่ 21

3.6 การเชื่อมต่ออุปกรณ์กลุ่มที่ 1 22

3.7 การเชื่อมต่ออุปกรณ์กลุ่มที่ 2 24

3.8 บอร์ดไดร์เวอร์มอเตอร์ 25

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.9 การเชื่อมต่ออุปกรณ์กลุ่มที่ 3 25
3.10 อุปกรณ์เซนเซอร์อัลทราโซนิก 26
3.11 การเชื่อมต่ออุปกรณ์กลุ่มที่ 4 26
3.12 ส่วนต่อประสานกับผู้ใช้ (User Interface, UI) 27
3.13 หน้าตาของหุ่นยนต์เมื่ออยู่ในสถานะรอ 29
3.14 หน้าตาของหุ่นยนต์เมื่ออยู่ในสถานะรับคำสั่ง 29
3.15 หน้าตาของหุ่นยนต์เมื่ออยู่ในสถานะติดตาม 30
3.16 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์เข้าสู่สถานะรับคำสั่ง 31
3.17 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์เข้าสู่สถานะติดตาม 31
3.18 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์กลับเข้าสู่สถานะรอ 32
3.19 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์เดินหน้า 32
3.20 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์ถอยหลัง 33
3.21 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์ไปทางซ้าย 33
3.22 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์ไปทางขวา 34
3.23 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์หมุนซ้าย 34
3.24 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์หมุนขวา 35
3.25 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์หยุดเคลื่อนที่ 35
3.26 การทำงานของโปรแกรมที่ใช้คอมพิวเตอร์ควบคุม 40
3.27 การทำงานของโปรแกรมสำหรับส่วนไมโครคอนโทรลเลอร์ 41
3.28 การทำงานของโปรแกรมหลักสำหรับบอร์ด Arduino Due 42
3.29 ฟังก์ชันที่ใช้ตรวจสอบระยะห่างของสิ่งกีดขวาง 44
3.30 ขั้นตอนในการเริ่มต้นใช้งานหุ่นยนต์ 45
3.31 หน้าจอแสดงผลของโปรแกรม 46
3.32 ส่วนต่าง ๆ ของหน้าจอแสดงผล 47
3.33 การกำหนดช่องทางการรับส่งข้อมูลผ่าน Wi-Fi 48

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.34 ตัวเลือกที่เพิ่มขึ้นหลังจากกำหนดช่องทางการรับส่งข้อมูลผ่าน Wi-Fi 48

3.35 หน้าจอของโปรแกรมสำหรับสั่งงานด้วย Joystick 49

3.36 รูปภาพประกอบตารางที่ 3.12 การใช้งาน Joystick เพื่อสั่งงานหุ่นยนต์ 50

3.37 หน้าจอของโปรแกรมสำหรับสั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์ 51

3.38 หน้าจอของโปรแกรมที่เปิดกับคอมพิวเตอร์บนตัวหุ่นยนต์ ซึ่งพร้อมรับคำสั่งผ่าน Wi-Fi 52

3.39 หน้าจอของโปรแกรมที่เปิดกับคอมพิวเตอร์ภายนอก ซึ่งพร้อมส่งคำสั่งผ่าน Wi-Fi 53

4.1 แผนภูมิแสดงค่าร้อยละความถูกต้องเปรียบเทียบระหว่างกรณีที่มีแสงมาก 59

4.2 การทดลองหาระยะที่เซนเซอร์อัลทราโซนิกสามารถวัดได้ 60

4.3 ระยะทางที่เซนเซอร์อัลทราโซนิกตัวที่ 1 วัดได้เปรียบเทียบกับระยะทางจริง 62

4.4 ระยะทางที่เซนเซอร์อัลทราโซนิกตัวที่ 2 วัดได้เปรียบเทียบกับระยะทางจริง 62

4.5 ระยะทางที่เซนเซอร์อัลทราโซนิกตัวที่ 3 วัดได้เปรียบเทียบกับระยะทางจริง 62

4.6 ระยะทางที่เซนเซอร์อัลทราโซนิกตัวที่ 4 วัดได้เปรียบเทียบกับระยะทางจริง 63

4.7 กราฟแสดงค่าร้อยละความคลาดเคลื่อนที่ระยะทางต่างๆ เมื่อวัดระยะทางด้วย 63

4.8 กราฟแสดงค่าร้อยละความคลาดเคลื่อนที่ระยะทางต่างๆ เมื่อวัดระยะทางด้วย 64

4.9 กราฟแสดงค่าร้อยละความคลาดเคลื่อนที่ระยะทางต่างๆ เมื่อวัดระยะทางด้วย 64

4.10 กราฟแสดงค่าร้อยละความคลาดเคลื่อนที่ระยะทางต่างๆ เมื่อวัดระยะทางด้วย 65

4.11 การทดลองหาขนาดของลำคลื่นของเซนเซอร์อัลทราโซนิก 66

4.12 ลักษณะบีมของเซนเซอร์อัลทราโซนิกตัวที่ 1 66

4.13 ลักษณะบีมของเซนเซอร์อัลทราโซนิกตัวที่ 2 66

4.14 ลักษณะบีมของเซนเซอร์อัลทราโซนิกตัวที่ 3 67

4.15 ลักษณะบีมของเซนเซอร์อัลทราโซนิกตัวที่ 4 67

4.16 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดรูปแบบที่ 1 68

4.17 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดรูปแบบที่ 2 69

4.18 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดรูปแบบที่ 3 69

4.19 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดรูปแบบที่ 4 70

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ปัจจุบันวิวัฒนาการทางเทคโนโลยีมีความก้าวหน้าอย่างไม่มีที่สิ้นสุด รัฐบาลจึงมีวิสัยทัศน์เชิงนโยบาย “ประเทศไทย 4.0” ซึ่งเป็นนโยบายหนึ่งในการวางรากฐานการพัฒนาประเทศในระยะยาว ที่ต้องการปรับเปลี่ยนโครงสร้างเศรษฐกิจไปสู่ “เศรษฐกิจที่ขับเคลื่อนด้วยนวัตกรรม” มีฐานความคิดหลักคือ เปลี่ยนจากการผลิตสินค้า “โภคภัณฑ์” ไปสู่สินค้าเชิง “นวัตกรรม” เปลี่ยนจากการขับเคลื่อนประเทศด้วยภาคอุตสาหกรรม ไปสู่การขับเคลื่อนด้วยเทคโนโลยี ความคิดสร้างสรรค์ และนวัตกรรม เปลี่ยนจากการเน้นภาคการผลิตสินค้า ไปสู่การเน้นภาคบริการมากขึ้น งานบางประเภทที่สามารถใช้หุ่นยนต์ทำงานทดแทนได้ โดยควบคุมการทำงานด้วยระบบคอมพิวเตอร์ จะช่วยลดภาระการทำงานที่ไม่จำเป็น ทำให้ผู้ปฏิบัติงานสามารถนำเวลาไปพัฒนางานด้านอื่นๆ ได้ จึงปฏิเสธไม่ได้ว่าความก้าวหน้าทางเทคโนโลยีนั้นนอกจากจะช่วยอำนวยความสะดวกสบายให้กับผู้คนเป็นจำนวนมากแล้ว ยังมีผลต่อการขับเคลื่อนประเทศด้วย

ทั้งนี้เทคโนโลยีหุ่นยนต์ในภาคบริการที่สามารถตอบโต้กับผู้ใช้งานตามความต้องการของผู้ใช้ได้ยังคงมีไม่มากนัก คณะผู้จัดทำจึงได้พัฒนาหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง ซึ่งสามารถนำไปประยุกต์ใช้งานได้หลายด้าน เช่น ใช้เป็นหุ่นยนต์ต้อนรับภายในพิพิธภัณฑ์ ในงานนิทรรศการ และอาจใช้เป็นหุ่นยนต์บริการภายในห้องสมุด โรงพยาบาล และห้างสรรพสินค้า ซึ่งจะช่วยสร้างภาพลักษณ์องค์กรให้ดูทันสมัยยิ่งขึ้นไปด้วย

1.2 จุดประสงค์ของโครงงาน

1.2.1 เพื่อพัฒนาหุ่นยนต์เคลื่อนที่ที่รับคำสั่งจากท่าทางที่กำหนด

1.2.2 เพื่อศึกษาการตรวจจับการเคลื่อนไหวของร่างกาย โดยวิธีการประมวลผลภาพ

1.2.3 เพื่อศึกษาการควบคุมการเคลื่อนที่ของหุ่นยนต์โดยล้อประเภท Omni–directional

1.2.4 เพื่อศึกษาการใช้งานเซนเซอร์อัลทราโซนิก (Ultrasonic sensor)

1.3 ขอบเขตของโครงงาน

1.3.1 หุ่นยนต์สามารถเคลื่อนที่ในพื้นราบในสภาวะปกติเท่านั้น

1.3.2 หุ่นยนต์มีการเคลื่อนที่ 7 รูปแบบ สั่งการด้วยท่าทาง 10 ท่าทาง

1.3.3 รับคำสั่งท่าทางจากบุคคลเพียงคนเดียว ซึ่งยืนอยู่ด้านหน้าในระยะที่กล้องสามารถจับได้

1.3.4 หุ่นยนต์สามารถติดตามผู้สั่งงานและหยุดเมื่อพบสิ่งกีดขวางได้

1.4 ขั้นตอนการดำเนินงาน

1.4.1 การออกแบบภาพรวมการทำงานของระบบ

1.4.2 การออกแบบโครงสร้างของหุ่นยนต์และการติดตั้งอุปกรณ์

1.4.3 การออกแบบการเชื่อมต่ออุปกรณ์

1.4.4 การออกแบบและพัฒนาวิธีการเลือกรับคำสั่งของหุ่นยนต์

1.4.5 การออกแบบและพัฒนาการเปลี่ยนสถานะของหุ่นยนต์

1.4.6 การศึกษาและพัฒนาโปรแกรมการตัดสินท่าทาง

1.4.7 การศึกษาและพัฒนาโปรแกรมการเคลื่อนที่ของหุ่นยนต์

1.4.8 การศึกษาและพัฒนาโปรแกรมการตรวจจับสิ่งกีดขวาง

1.4.9 การทดสอบการทำงานของหุ่นยนต์

1.5 แผนการดำเนินงาน

ตารางที่ 1.1 ตารางแสดงระยะเวลาการดำเนินงาน

รายการ	พ.ศ.2560					พ.ศ.2561			
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.
1.5.1 การออกแบบภาพรวมการทำงานของระบบ									
1.5.2 การออกแบบโครงสร้างของหุ่นยนต์และการติดตั้งอุปกรณ์									
1.5.3 การออกแบบการเชื่อมต่ออุปกรณ์									

ตารางที่ 1.1 (ต่อ) ตารางแสดงระยะเวลาการดำเนินงาน

รายการ	พ.ศ.2560					พ.ศ.2561			
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.
1.5.4 การออกแบบและพัฒนาวิธีการเลือกรับคำสั่งของหุ่นยนต์									
1.5.5 การออกแบบและพัฒนาการเปลี่ยนสถานะของหุ่นยนต์									
1.5.6 การศึกษาและพัฒนาโปรแกรมการตัดสินท่าทาง									
1.5.7 การศึกษาและพัฒนาโปรแกรมการเคลื่อนที่ของหุ่นยนต์									
1.5.8 การศึกษาและพัฒนาโปรแกรมการตรวจจับสิ่งกีดขวาง									
1.5.9 การทดสอบการทำงานของหุ่นยนต์									
1.5.10 ปรับปรุงการทำงานโดยรวมของหุ่นยนต์									
1.5.11 รวบรวมผลของการทำงานและจัดทำเอกสาร									

1.6 ผลที่คาดว่าจะได้รับ

1.6.1 ได้หุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง ซึ่งสามารถนำไปใช้ในงานบริการประเภทต่างๆ ได้

1.7 งบประมาณของโครงงาน

1.7.1 ค่าอุปกรณ์สำหรับทดสอบและประดิษฐ์ชิ้นงาน	135,000 บาท
1.7.2 ค่าเอกสารเล่มปริญญานิพนธ์	3,000 บาท
รวมเป็นเงินทั้งสิ้น	138,000 บาท

บทที่ 2
หลักการและทฤษฎี

ในบทนี้จะกล่าวถึงหลักการและทฤษฎีต่างๆ ที่เกี่ยวข้อง ประกอบด้วยทฤษฎีและหลักการทำงานของ Kinect หลักการควบคุมการทำงานของมอเตอร์ไฟฟ้า ทฤษฎีของวงจรไดร์เวอร์มอเตอร์ หลักการทำงานของเซนเซอร์อัลทราโซนิก ทฤษฎีพื้นฐานและหลักการทำงานของหุ่นยนต์ที่สั่งการด้วยท่าทางของผู้ใช้งาน หลังจากนั้นระบบจะส่งคำสั่งไปที่ไมโครคอนโทรลเลอร์ เพื่อสั่งการให้มอเตอร์หมุนตามทิศทางที่กำหนดไว้ในโปรแกรมต่อไป

2.1 ทฤษฎีพื้นฐานและหลักการทำงานของหุ่นยนต์เคลื่อนที่

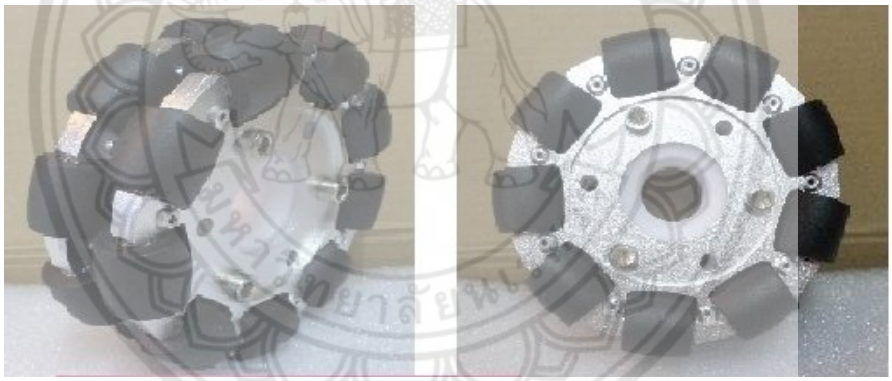
ในสมัยก่อนหุ่นยนต์เป็นเพียงจินตนาการของมนุษย์ ที่ต้องการให้เข้ามาช่วยในการผ่อนแรงจากงานที่ทำ หรือช่วยปฏิบัติงานที่ยากลำบากเกินขอบเขตความสามารถ และจากจินตนาการได้กลายเป็นแรงบันดาลใจให้มนุษย์คิดประดิษฐ์สร้างสรรค์หุ่นยนต์ขึ้นมา จนกลายเป็นหุ่นยนต์หรือ โรบอท (Robot) ในปัจจุบัน คำว่า Robot มาจากคำว่า Robota ในภาษาเช็ก ซึ่งแปลโดยตรงว่า การทำงานเสมือนทาส ถือกำเนิดขึ้นจากละครเวทีเรื่อง Rossum's Universal Robots (Karel Čapek. Rossum's Universal Robots (R.U.R.).) ซึ่งเป็นบทเนื้อหาของละครเวทีมีความเกี่ยวพันกับจินตนาการของมนุษย์ ในการใฝ่หาสิ่งที่มาช่วยในการปฏิบัติงาน การประดิษฐ์คิดค้นสร้างหุ่นยนต์จึงถือกำเนิดขึ้นเพื่อเป็นเสมือนทาสคอยรับใช้มนุษย์ การใช้ชีวิตร่วมกันระหว่างหุ่นยนต์และมนุษย์ดำเนินต่อไปจนกระทั่งหุ่นยนต์เกิดมีความคิดเช่นเดียวกับมนุษย์ การถูกกดขี่ข่มเหงเช่นทาสจากมนุษย์ทำให้หุ่นยนต์เกิดการต่อต้านไม่ยอมเป็นเบี้ยล่างอีก ซึ่งละครเวทีเรื่องนี้โด่งดังมากจนทำให้คำว่า Robot เป็นที่รู้จักทั่วโลก

ส่วนประกอบของหุ่นยนต์ประกอบด้วยอุปกรณ์และชิ้นส่วนต่างๆ มากมายซึ่งอุปกรณ์แต่ละชนิดจะมีหน้าที่แตกต่างกันตามลักษณะและวัตถุประสงค์ของการใช้งาน การเลือกใช้จึงจำเป็นต้องอาศัยความรู้ความเข้าใจรวมถึงความเหมาะสม เพื่อให้หุ่นยนต์สามารถทำงานได้อย่างมีประสิทธิภาพ รวดเร็ว คงทน และประหยัดพลังงาน หุ่นยนต์จะแบ่งส่วนประกอบใหญ่ๆ เป็น 4 ส่วนได้แก่ อุปกรณ์ทางกล (Mechanic devices) คือ ชิ้นส่วนกลไกต่างๆ ของหุ่นยนต์ เช่น โครงสร้าง เป็นต้น อุปกรณ์ขับเร้า (actuator) คืออุปกรณ์ที่สามารถเปลี่ยนแปลงพลังงานไฟฟ้าที่ป้อนเข้าให้กลายเป็นการกระจัดการเคลื่อนที่ หรือแรง เช่น มอเตอร์ไฟฟ้า อุปกรณ์ไฟฟ้า (electronic devices) คืออุปกรณ์ที่ใช้

สัญญาณทางระบบไฟฟ้า เช่น เซนเซอร์และวงจรขับต่างๆ อุปกรณ์ควบคุม (controller) คือสมองกลที่ควบคุมการทำงานของหุ่นยนต์ เช่น สมองกลที่ประดิษฐ์จากอุปกรณ์อิเล็กทรอนิกส์ คอมพิวเตอร์ชนิดแผงวงจรสำเร็จรูป

2.2 การเคลื่อนที่โดยใช้ล้อประเภท Omnidirectional

การเคลื่อนที่โดยใช้ล้อ เหมาะสำหรับหุ่นยนต์ทั่วไปที่ใช้งานบนพื้นราบ ซึ่งมีข้อดีคือหุ่นยนต์สามารถเคลื่อนที่ได้อย่างรวดเร็ว ควบคุมง่าย ดังนั้น ส่วนใหญ่หุ่นยนต์จึงถูกสร้างให้เคลื่อนที่โดยล้อ แต่ข้อจำกัดของการเคลื่อนที่ในลักษณะนี้คือ หุ่นยนต์จะเคลื่อนที่ได้ยากในพื้นที่ต่างระดับและในพื้นที่ที่ขรุขระ โดยในโครงงานนี้จะใช้ล้อประเภท Omni-directional ดังแสดงในรูปที่ 2.1 คือมีสามแกนอิสระประกอบด้วยทิศทางการหมุนตามแนวการวางของล้อ การหมุนของลูกกลิ้งที่ติดอยู่โดยรอบๆ และการหมุนที่จุดสัมผัสระหว่างลูกกลิ้งกับพื้น สำหรับสาเหตุที่เลือกใช้ล้อประเภทนี้เนื่องจากจะทำให้หุ่นยนต์เคลื่อนที่โดยที่ทิศทางของหน้าหุ่นไม่เปลี่ยน

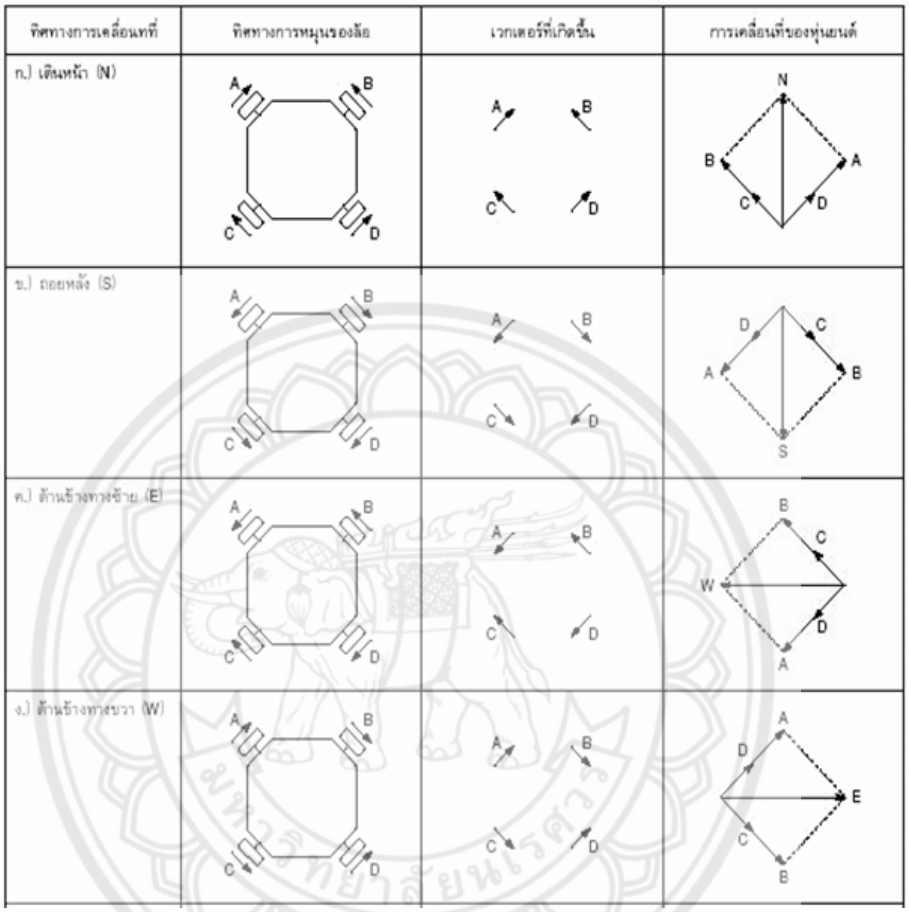


รูปที่ 2.1 ล้อประเภท Omni-directional

(ที่มา: http://www.sangtawan.org/product_detail.asp?product_id=1035&lng=th)

ในโครงงานนี้ออกแบบให้หุ่นยนต์เคลื่อนที่ด้วยล้อ 4 ล้อ การควบคุมการเคลื่อนที่ของล้อประเภทนี้จะเป็นการรวมเวกเตอร์ โดยอาศัยการหมุนของมอเตอร์กระแสตรงจำนวน 4 ชุด เพื่อกำหนดทิศทางการเคลื่อนที่ของหุ่นยนต์ ซึ่งทิศทางของการเคลื่อนที่จะขึ้นอยู่กับผลรวมของเวกเตอร์ความเร็วและทิศทางของมอเตอร์ในแต่ละตัว ยกตัวอย่างเช่น เมื่อต้องการเคลื่อนที่ไปข้างหน้า (ทิศเหนือ) จะต้องสั่งมอเตอร์ทั้ง 4 หมุนพร้อมๆ กันในทิศทางดังรูปที่ 2.2 (ก) กรณีนี้จะเกิดแรงกระทำต่อล้อทั้ง 4 มีทิศทางแสดงได้เป็นเวกเตอร์ 4 ตัว ในที่นี้เวกเตอร์ทั้ง 4 มีขนาดเท่ากัน เมื่อนำเวกเตอร์ทั้ง 4 มารวมกันก็จะเกิดแรงลัพธ์ขึ้น แรงลัพธ์ที่เกิดจะแสดงทิศทางที่หุ่นยนต์เคลื่อนที่ไป เมื่อต้องการให้

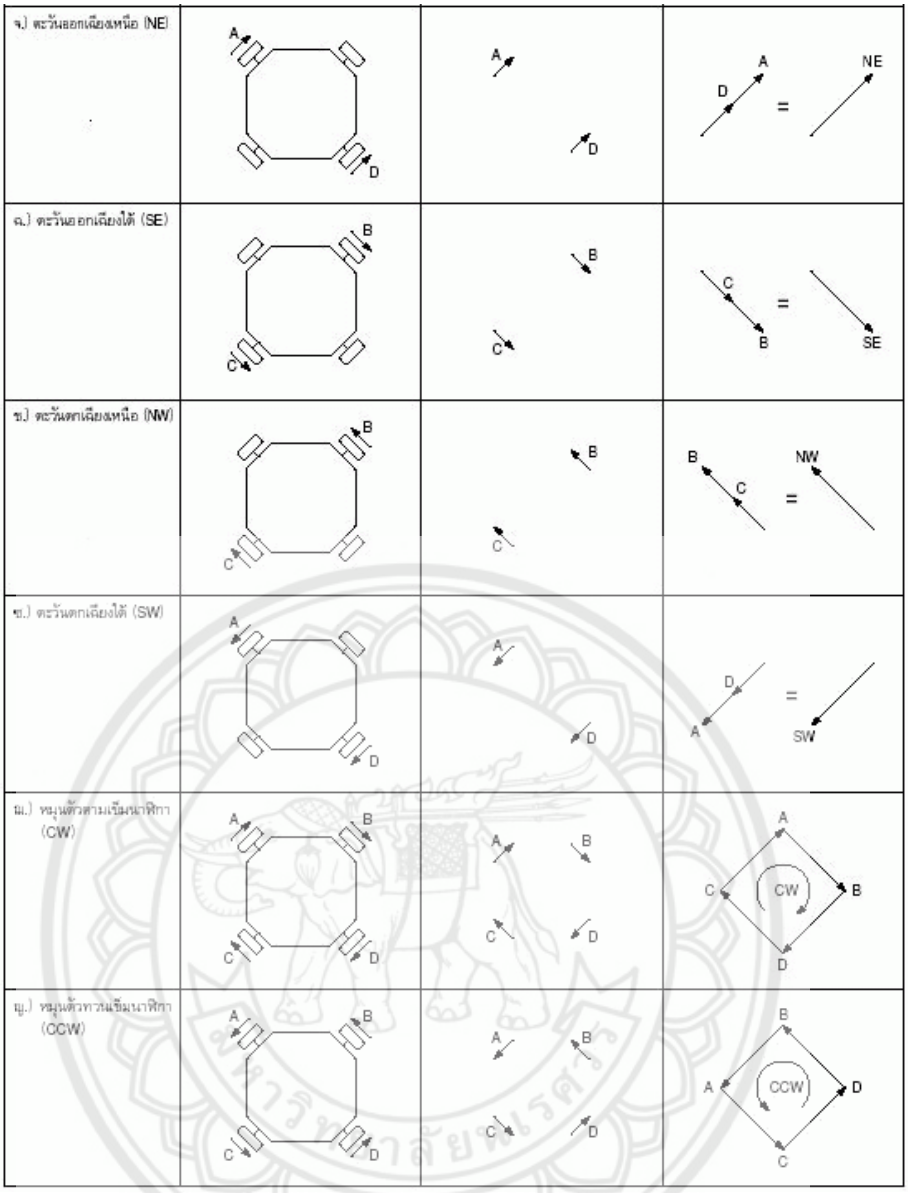
หุ่นยนต์เคลื่อนที่ไปในทิศทางอื่นๆ อีก 3 ทิศ (ถอยหลัง,ด้านข้างทางซ้าย,ด้านข้างทางขวา) ก็สามารถทำได้โดยการสั่งให้มอเตอร์หมุนล้อไปในทิศทางต่างๆ ดังที่แสดงในรูปที่ 2.2 (ข), 2.2 (ค) และ 2.2 (ง) ตามลำดับ



รูปที่ 2.2 ทิศทางการเคลื่อนที่แบบรวมเวกเตอร์

(ที่มา: http://www.nawattakam.com/talk/index.php?topic=368.0)

สำหรับการเคลื่อนที่ในแนวทแยงอีก 4 ทิศที่เหลือสามารถทำได้โดยอาศัยหลักการรวมแรงทางเวกเตอร์ เช่นต้องการให้หุ่นยนต์เคลื่อนที่ในแนวทแยงมุมในทิศตะวันออกเฉียงเหนือ ก็สั่งให้มอเตอร์หมุนเฉพาะล้อ A กับ D ในทิศทางดังรูปที่ 2.2 (จ) ส่วนมอเตอร์ B กับ C ไม่ต้องทำงาน ในกรณีนี้จะเกิดขึ้นเฉพาะเวกเตอร์ A กับ D ในทิศทางเดียวกัน เมื่อนำมารวมกันจะเกิดแรงลัพธ์ทำให้หุ่นยนต์เคลื่อนที่ไปในทิศตะวันออกเฉียงเหนือ ส่วนในแนวทแยงมุมอื่นๆ สามารถทำได้โดยการสั่งให้มอเตอร์หมุนในทิศต่างๆ ดังรูปที่ 2.2 (ฉ), 2.2 (ช) และ 2.2 (ซ) ตามลำดับ นอกเหนือจากการเคลื่อนที่ทั้ง 8 ทิศยังมีการเคลื่อนที่ได้อีก 2 แบบที่หุ่นยนต์สามารถทำได้คือ การหมุนรอบตัวเองในทิศทางการหมุนรอบเข็มนาฬิกาและทวนเข็มนาฬิกา ซึ่งอาศัยการรวมแรงทางเวกเตอร์อีกเช่นกัน โดยสั่งให้มอเตอร์หมุนทั้ง 4 ล้อในทิศทางดังรูปที่ 2.2 (ฌ) และ 2.2 (ญ) ตามลำดับ



รูปที่ 2.2 (ต่อ) ทิศทางการเคลื่อนที่แบบรวมเวกเตอร์

(ที่มา: http://www.nawattakam.com/talk/index.php?topic=368.0)

การเคลื่อนที่ของหุ่นยนต์ในลักษณะนี้อาจมีปัญหาบ้าง เนื่องจากในความเป็นจริงล้อทั้ง 4 ของหุ่นยนต์จะสร้างแรงแต่ละแรงออกมาไม่เท่ากัน ทำให้หุ่นยนต์เคลื่อนที่เบนจากทิศทางเดิมที่ต้องการไปบ้างเล็กน้อย ในโครงงานนี้การควบคุมหุ่นยนต์จะใช้อุปกรณ์ที่มีชื่อว่า “Kinect” เป็นตัวรับข้อมูลแล้วส่งไปที่ระบบประมวลผลหลักหรือคอมพิวเตอร์ส่วนบุคคลขนาดเล็ก (Mini Personal Computer) หลังจากนั้นระบบประมวลผลหลักจะส่งค่าไปที่ไมโครคอนโทรลเลอร์ เพื่อสั่งการให้มอเตอร์หมุนตามที่กำหนดไว้ในโปรแกรมต่อไป

2.3 ทฤษฎีและหลักการทำงานของ Kinect

Kinect เป็นอุปกรณ์รับค่าการเคลื่อนไหวท่าทางของมนุษย์ ที่ผลิตโดย Microsoft เพื่อนำมาใช้คู่กับเครื่องเล่นวิดีโอเกมส์ Xbox และระบบปฏิบัติการ Microsoft Windows ซึ่งจะช่วยให้ผู้ใช้สามารถควบคุมและโต้ตอบกับเครื่องเล่นวีดีโอเกมส์และคอมพิวเตอร์ ผ่านทาง Natural User Interface (NUI) ที่ใช้ท่าทางและคำสั่งเสียงโดยไม่จำเป็นต้องมีอุปกรณ์ควบคุม เช่น จอยสติก (Joystick) แต่ตัวผู้สั่งงานจะต้องเป็นจอยคอนโทรลเลอร์ (Joy Controller) แทนภาพของอุปกรณ์ Kinect Xbox ONE แสดงดังรูปที่ 2.3

รูปที่ 2.3 Kinect Xbox ONE

(ที่มา: https://www.xbox.com/en-US/xbox-one/accessories/kinect)

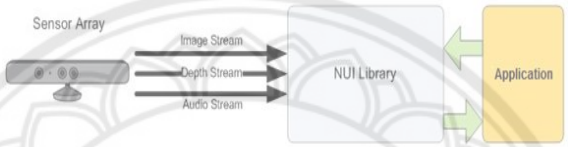
ในตารางที่ 2.1 แสดงข้อมูลจำเพาะของ Kinect Xbox ONE ซึ่งมีคุณสมบัติ คือ อุปกรณ์ Kinect สามารถตรวจจับโครงกระดูกได้ถึง 6 ชุด และข้อต่อ 25 จุด ต่อคนหนึ่งคน ทำให้การตรวจจับส่วนต่างๆ ของร่างกายมีความละเอียดยิ่งขึ้น เซนเซอร์ตรวจจับความลึกและการตรวจจับในที่มืดมีความละเอียด 512 x 424 พิกเซล เซนเซอร์นี้จะทำให้ภาพดูมีมิติสามารถตรวจจับได้แม้จะอยู่ในที่มืด กล้อง RGB คุณภาพระดับ 1080p ความละเอียด 1920 x 1080 พิกเซล 30 เฟรมต่อวินาที ทำให้การแสดงผล และนำภาพไปใช้ร่วมกับโปรแกรมมีประสิทธิภาพอย่างมาก ไมโครโฟนอาร์เรย์มีไมโครโฟนสี่ตัว เพื่อบันทึกเสียง หาตำแหน่งของแหล่งกำเนิดเสียง

ตารางที่ 2.1 ข้อมูลจำเพาะของ Kinect Xbox ONE

Kinect Xbox ONE	ข้อมูลจำเพาะ
ระยะการตรวจจับ	ใกล้สุด 50 cm ไกลสุด 450 cm
มุมมองภาพ	แนวตั้ง 60 องศา แนวนอน 70 องศา
การเชื่อมต่อคอมพิวเตอร์	USB 3.0
เซนเซอร์ตรวจจับความลึก	512 x 424 พิกเซล
กล้อง	RGB 1920 x 1080 พิกเซล
ระบบปฏิบัติการ	Windows 8
สัญญาณเสียง	ไมโครโฟนอาร์เรย์ 4 ตัว ความถี่ 48 kHz

(ที่มา: http://zugara.com/how-does-the-kinect-2-compare-to-the-kinect-1)

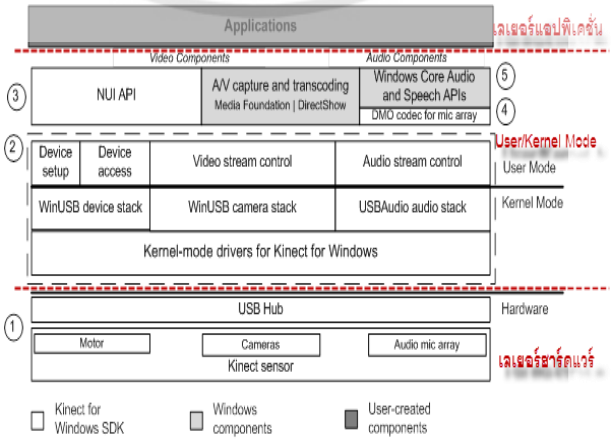
เซนเซอร์ Kinect จะทำงานคล้ายกับ User Interface (UI) ที่เชื่อมต่อระหว่างผู้ใช้งานกับโปรแกรม ซึ่งจะมีทั้งส่วนที่นำเข้าข้อมูลหรือส่วนสั่งงาน (Input) และส่วนที่ใช้แสดงผลหรือส่วนที่รอคำสั่งจากผู้ใช้งาน (Output) อาทิเช่น แป้นพิมพ์ เมาส์ จะจัดเป็นอินเตอร์เฟสชนิดสัมผัส (Touched User Interface) ส่วน Kinect นั้นเป็นอินเตอร์เฟสชนิดไม่สัมผัส (Touch-less User Interface) หรืออินเตอร์เฟสชนิดเสมือนจริง (Natural User Interface) คือใช้การเคลื่อนไหวของร่างกายมนุษย์เป็นตัวสั่งงาน รูปที่ 2.4 แสดงแผนภาพการเชื่อมต่อระหว่าง Kinect กับแอปพลิเคชัน จากรูปจะเห็นว่าเซนเซอร์ Kinect มีลักษณะเซนเซอร์เป็นแบบอาร์เรย์ (มีเซนเซอร์หลายๆ ตัว) โดย Kinect จะติดต่อกับแอปพลิเคชันผ่าน NUI Library (เป็นไลบรารีที่เชื่อมต่อระหว่างเซนเซอร์ Kinect กับแอปพลิเคชัน)



รูปที่ 2.4 แผนภาพการติดต่อระหว่าง Kinect กับแอปพลิเคชัน

(ที่มา: https://kinectasia.wordpress.com/tag/nui/)

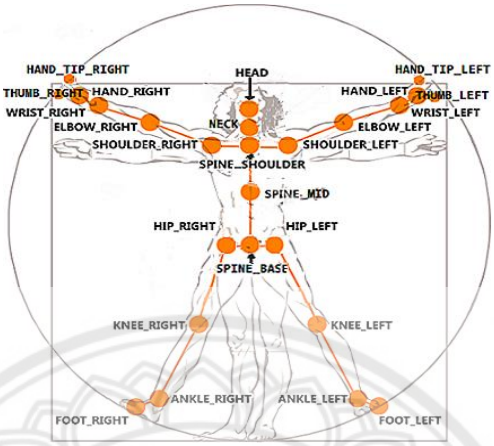
รูปที่ 2.5 แสดงส่วนประกอบต่างๆ ของ Software Development Kit (SDK) คือเครื่องมือที่ใช้สำหรับพัฒนาโปรแกรมหรือแอปพลิเคชัน ซึ่งในชุด SDK จะมีโปรแกรมและไลบรารีต่างๆ ที่จำเป็นต่อการพัฒนาแอปพลิเคชั่น ในส่วนบนสุดของรูปที่ 2.5 จะเป็นเลเยอร์ของแอปพลิเคชั่น ซึ่งเป็นส่วนที่ผู้ใช้งานจะนำไปพัฒนาโปรแกรมต่อไป ในเลเยอร์ต่อมาคือเลเยอร์ API ประกอบไปด้วย API ต่างๆ รองลงมาจะเป็นเลเยอร์ User/Kernel Mode และเลเยอร์สุดท้ายคือเลเยอร์ฮาร์ดแวร์



รูปที่ 2.5 ส่วนประกอบของ Software Development Kit (SDK)

(ที่มา: https://kinectasia.wordpress.com/tag/nui/)

ส่วนสำคัญของ Kinect คือ NUI API เป็นส่วนเชื่อมต่อระหว่าง Kinect Sensor (RGB Camera, IR Camera, Microphones) กับคอมพิวเตอร์ หน้าที่สำคัญคือการนำข้อมูล (รูปภาพ ความตื้นลึก) ไปใช้ในการประมวลผล เพื่อนำมาใช้งานในรูปแบบ Skeleton Tracking ดังแสดงในรูปที่ 2.6



รูปที่ 2.6 จุดข้อต่อทั้ง 25 จุดของร่างกายที่ Kinect สามารถระบุพิกัดได้

(ที่มา: http://www.bryancook.net/2014/03/drawing-kinect-v2-body-joints.html)

2.4 ระบบประมวลผลหลัก (Host computer)

ในโครงงานนี้ได้นำเอาเครื่องคอมพิวเตอร์ส่วนบุคคลขนาดเล็ก (Mini Personal Computer) ดังแสดงในรูปที่ 2.7 มาใช้สำหรับรับข้อมูลจากอุปกรณ์ Kinect (ส่วนควบคุมระดับบน) และสื่อสารกับบอร์ดไมโครคอนโทรลเลอร์ (ส่วนควบคุมระดับล่าง) เหตุผลที่เลือกใช้คอมพิวเตอร์ตัวนี้เพราะอุปกรณ์มีขนาดเล็กสามารถวางบนโครงสร้างหุ่นยนต์ได้ อีกทั้งมีระดับความต้องการของระบบที่รองรับกับอุปกรณ์ Kinect ข้อมูลจำเพาะทางเทคนิคของคอมพิวเตอร์ส่วนบุคคลขนาดเล็ก (Mini Personal Computer) สามารถแสดงได้ดังตารางที่ 2.2 หลักการทำงานของคอมพิวเตอร์ส่วนบุคคลขนาดเล็ก (Mini Personal Computer) คือรับค่าจากอุปกรณ์ Kinect แล้วนำไปประมวลผลส่งคำสั่งให้กับอุปกรณ์ไมโครคอนโทรลเลอร์ (ส่วนควบคุมระดับล่าง) เพื่อควบคุมการทำงานของมอเตอร์ต่อไป

รูปที่ 2.7 คอมพิวเตอร์ส่วนบุคคลขนาดเล็ก

(ที่มา: http://www.thailand.intel.com/content/www/th/th/products/boards-kits/nuc/mini-pcs/nuc7i7bnhxg.html)

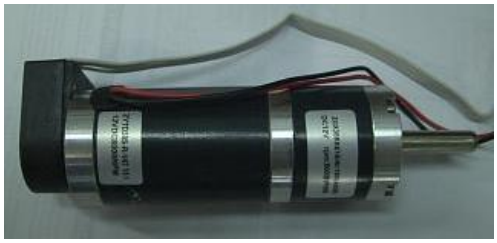
ตารางที่ 2.2 ข้อมูลจำเพาะทางเทคนิคของ Mini PC

Device Name	INTEL NUC 7 ENTHUSIAST MINI PC – NUC7I7BNHXG
DC Input Voltage Supported	12-19 VDC
Processor Included	Intel core i7-7567U Processor
Operating System	Windows 10 , 64-bit
Memory	8GB DDR4-2400
Graphics Output	HDMI 2.0a , USB-C (DP1.2)
USB Ports	6 ports (USB 2.0 , 3.0)

(ที่มา: https://www.thailand.intel.com/content/www/th/th/products/boards-kits/nuc/mini-pcs/nuc7i7bnhxg.html)

2.5 หลักการควบคุมการทำงานของมอเตอร์ไฟฟ้า

มอเตอร์ไฟฟ้ากระแสตรง คืออุปกรณ์ไฟฟ้าที่แปลงพลังงานไฟฟ้าเป็นพลังงานกล โดยรูปที่ 2.8 แสดงภาพมอเตอร์ไฟฟ้ากระแสตรงที่นำมาใช้ในโครงงานนี้โดยมีคุณสมบัติเป็นมอเตอร์ DC 12 V มีเอนโค้ดเดอร์ (Encoder) ในตัว ความละเอียดของเกียร์ทด 1 ต่อ 16 รอบ การทำงานของมอเตอร์ไฟฟ้ากระแสตรง เมื่อมีกระแสไฟฟ้าผ่านเข้าไปยังขดลวด สนามแม่เหล็กจะทำให้เกิดแรงแม่เหล็ก ซึ่งมีสัดส่วนของแรงขึ้นกับกระแสแรงของสนามแม่เหล็ก โดยแรงจะเกิดขึ้นเป็นมุมฉากกับกระแสและสนามแม่เหล็ก ขณะที่ทิศทางของแรงตรงกันข้าม ถ้าหากกระแสของสนามแม่เหล็กไหลย้อนกลับ จะทำให้เกิดการเปลี่ยนแปลงของกระแส และสนามแม่เหล็ก เป็นผลทำให้ทิศทางของแรงเปลี่ยนไป ด้วยคุณสมบัตินี้ทำให้มอเตอร์กระแสตรงกลับทิศการหมุนได้ สนามแม่เหล็กของมอเตอร์ส่วนหนึ่งเกิดขึ้นจากแม่เหล็กถาวรซึ่งจะถูกยึดติดกับแผ่นเหล็ก หรือเหล็กกล้า โดยปกติส่วนนี้จะเป็นส่วนที่ยึดอยู่กับที่ และขดลวดเหนี่ยวนำจะพันอยู่กับส่วนที่เป็นแกนหมุนของมอเตอร์ มอเตอร์ไฟฟ้าสามารถขับเคลื่อนโดยแหล่งจ่ายไฟฟ้ากระแสตรง เช่น จากแบตเตอรี่ เป็นต้น

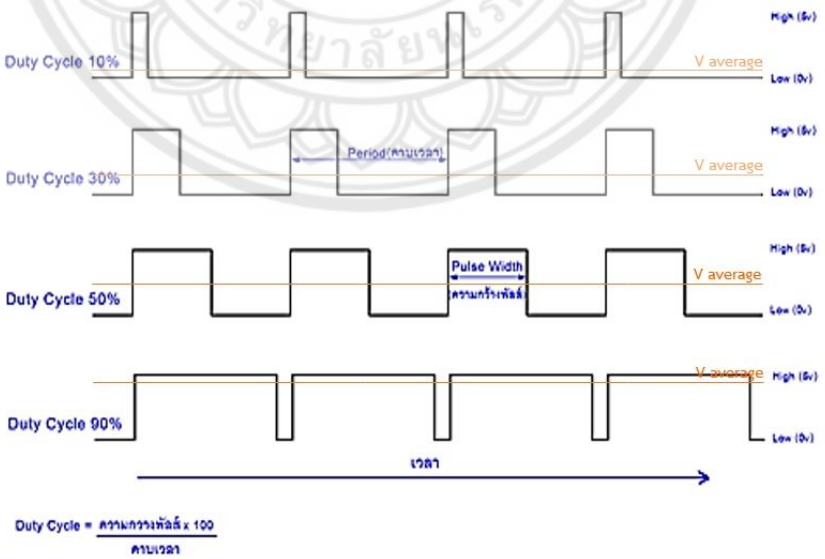


รูปที่ 2.8 มอเตอร์ไฟฟ้ากระแสตรง

(ที่มา: http://www.sangtawan.org/product_detail.asp?product_id=959&lng=th)

2.5.1 การควบคุมความเร็วของมอเตอร์โดยวิธีการมอดูเลตทางความกว้างของพัลส์

การมอดูเลตทางความกว้างของพัลส์ (Pulse Width Modulation : PWM) คือวิธีที่ใช้ในการควบคุมวงจรและเขียนค่าแบบแอนะล็อก (Analog) ด้วยพอร์ตดิจิทัล (Digital) โดยปกติแล้วพอร์ตดิจิทัลจะสามารถแบ่งได้แค่ 2 สถานะ คือ HIGH (5V) กับ LOW (0V) เท่านั้น ซึ่งการใช้วิธีการมอดูเลตทางความกว้างของพัลส์ จะทำให้พอร์ตดิจิทัลสามารถเขียนค่าได้มากกว่า HIGH หรือ LOW โดยทำให้สามารถเขียนค่าเป็นแบบแอนะล็อกได้ โดยใช้วิธีการปรับสถานะของสัญญาณลอจิก HIGH/LOW สลับกันไปมาด้วยคาบเวลาหนึ่งๆ โดยค่าที่ได้นั้นจะขึ้นอยู่กับสัดส่วนเวลาของสัญญาณในช่วงเวลาที่มีสถานะเป็น HIGH กับช่วงเวลาที่มีสถานะเป็น LOW ช่วงเวลาทั้งหมดที่สัญญาณมีสถานะเป็น HIGH นั้นจะเรียกว่าความกว้างพัลส์ (Pulse Width) โดยสัญญาณพัลส์เมื่อเทียบร้อยละของช่วงเวลาที่เป็น HIGH กับร้อยละของคาบเวลา (Period) ของพัลส์ลูกนั้นๆ จะเรียกว่า Duty Cycle ดังแสดงในรูปที่ 2.9 จะเห็นว่าที่ Duty Cycle เท่ากับ 50% ความกว้างพัลส์ (Pulse Width) จะเป็นครึ่งหนึ่งเมื่อเทียบกับร้อยละของคาบเวลา (Period) ยกตัวอย่างเช่นสมมุติให้คาบเวลามีค่าเท่ากับ 2 วินาที และความกว้างพัลส์มีค่าเท่ากับ 1 วินาที หมายความว่าใน 2 วินาที มีสถานะ HIGH 1 วินาที และสถานะ LOW 1 วินาที เมื่อนำมาคำนวณหาค่า Duty Cycle จะได้ (1/2)x100 = 50% ซึ่งค่า Duty Cycle จะส่งผลต่อความเร็วของมอเตอร์ โดยกระแสไฟฟ้าที่จ่ายออกไปจะเหลือแค่ 50% ของทั้งหมด เมื่อนำไปควบคุมมอเตอร์จะทำให้มอเตอร์หมุนด้วยความเร็ว 50% ของความเร็วสูงสุด

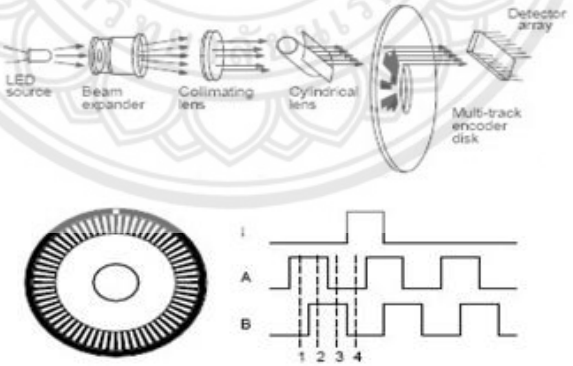


รูปที่ 2.9 Duty Cycle

(ที่มา: http://naringroup.blogspot.com/2016/03/arduino-pwm.html)

2.5.2 การควบคุมความเร็วของมอเตอร์โดยการป้อนกลับสัญญาณ

เอนโค้ดเดอร์ คือ เซนเซอร์ชนิดหนึ่งที่ทำหน้าที่ในการเข้ารหัส จากระยะทางจากการหมุนรอบตัวเอง และแปลงออกมาเป็นรหัสในรูปแบบของสัญญาณไฟฟ้าโดยสามารถนำเอารหัสเหล่านี้มาแปลงกลับ เพื่อหาค่าต่างๆ ที่ต้องการได้ ไม่ว่าจะเป็นระยะทางการหมุน องศาการเคลื่อนที่หรือความเร็วรอบ ซึ่งในโครงงานนี้นำเอนโค้ดเดอร์มาใช้เพื่อวัดความเร็วรอบ โดยหลักการทำงานของเอนโค้ดเดอร์มีลักษณะเป็นแผ่นกลมมีแกนอยู่ตรงกลาง ที่แผ่นกลมจะมีช่องเล็กๆ ที่แสงสามารถส่องผ่านได้ เรียกช่องนี้ว่าช่อง slit ซึ่งที่ด้านหนึ่งของแผ่นกลม จะมีตัวส่งสัญญาณแสงอินฟราเรด ไปยังตัวรับสัญญาณแสงอินฟราเรด ซึ่งจะอยู่ด้านตรงกันข้าม เมื่อแกนหมุนทำให้แผ่นกลมตัดลำแสงอินฟราเรด ดังนั้น ชุดรับแสงอินฟราเรด จึงมีแสงมากระทบเป็นช่วงๆ ตามจังหวะที่แสงผ่านช่อง slit ทำให้ได้สัญญาณ output ออกมา จำนวน pulse ที่ได้ออกมา จะเป็นตัวที่ชี้ว่ามอเตอร์หมุนไปกี่องศาหรือกี่รอบ ซึ่งสามารถคำนวณได้จากสมการ จำนวนรอบที่มอเตอร์หมุนเท่ากับจำนวน pulse โดยที่ค่าความละเอียดของเอนโค้ดเดอร์ใช้หน่วยเป็น pulse / round หรือ ppr เช่น เอนโค้ดเดอร์ความละเอียด 1000 ppr หมายความว่า motor เมื่อหมุนไป 1 รอบจะมี pulse ออกมา 1000 pulse เป็นต้น การทำงานของเอนโค้ดเดอร์แสดงดังรูปที่ 2.10 ซึ่งในโครงงานนี้ใช้เอนโค้ดเดอร์สำหรับเก็บค่าการหมุนของมอเตอร์และนำค่าที่ได้ป้อนกลับไปยังไมโครคอนโทรลเลอร์เพื่อควบคุมความเร็วของการเคลื่อนที่ของหุ่นยนต์ต่อไป



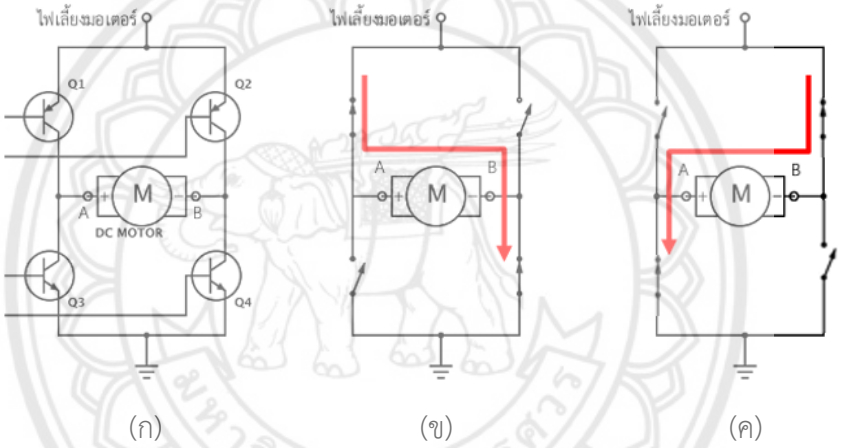
รูปที่ 2.10 การทำงานของเอนโค้ดเดอร์

(ที่มา: http://bmesensor.blogspot.com/p/potentiometer-or-potentiometric.html)

2.6 ทฤษฎีของวงจรไดร์เวอร์มอเตอร์

เนื่องจากมอเตอร์ไฟฟ้ากระแสตรงต้องใช้กระแสไฟฟ้าสูงในการทำงาน ซึ่งไมโครคอนโทรลเลอร์จะไม่สามารถจ่ายกระแสไฟฟ้าโดยตรงให้กับมอเตอร์ได้ จึงต้องมีชุดขับกระแสเพื่อทำให้มอเตอร์

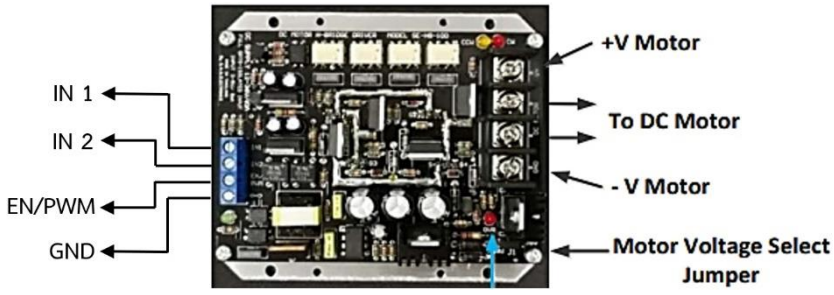
ทำงานได้เป็นปกติ ชุดขับกระแสดีซีมอเตอร์หรือวงจรไดร์เวอร์มอเตอร์ ที่ใช้ในโครงงานนี้จะเป็นแบบ H-Bridge ซึ่งวงจรประกอบด้วยทรานซิสเตอร์ (Transistor) จำนวน 4 ชุด (Q1-Q4) ทำหน้าที่เป็นสวิตช์เปิด-ปิด โดยต่อกับ DC Motor ดังรูป 2.11 (ก) ซึ่งสามารถควบคุมการทิศทางการไหลของกระแสได้ เมื่อส่งสัญญาณควบคุมให้ทรานซิสเตอร์ Q1 และ Q4 ทำงาน และปิดการทำงานของทรานซิสเตอร์ Q2 และ Q3 กระแสจะไหลจากจุด A ไปจุด B ดังรูป 2.11 (ข) จึงทำให้มอเตอร์เริ่มหมุน เมื่อส่งสัญญาณควบคุมให้ทรานซิสเตอร์ Q2 และ Q3 ทำงาน และปิดการทำงานของทรานซิสเตอร์ Q1 และ Q4 กระแสจะไหลจากจุด B ไปจุด A ดังรูป 2.11 (ค) เป็นผลให้มอเตอร์หมุนกลับทิศทาง โดยทิศทางการหมุนของมอเตอร์จะเป็นไปตามตารางที่ 2.3 ซึ่งแสดงการควบคุมการทำงานของไดร์เวอร์มอเตอร์



รูปที่ 2.11 หลักการทำงานของวงจร H-Bridge สำหรับควบคุมการหมุนของ DC Motor

(ที่มา: http://aimagin.com/blog/motor/?lang=th)

สำหรับวงจรไดร์เวอร์มอเตอร์ที่ใช้ในโครงงานนี้คือ “DC Motor H-Bridge Driver (SE-HB-100)” ดังแสดงในรูปที่ 2.12



รูปที่ 2.12 บอร์ดขับมอเตอร์ดีซีแบบ H-Bridge

(ที่มา: http://www.es.co.th/detail.asp?prod=089000021)

ตารางที่ 2.3 การควบคุมการทำงานของไดร์เวอร์มอเตอร์

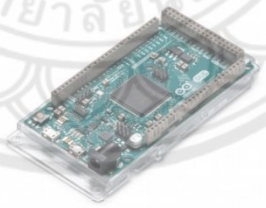
EN / PWM	IN 1	IN 2	การทำงานของมอเตอร์
0V / PWM	X	X	หยุดเมื่อหมดแรงเฉื่อย
5V / PWM	1	0	หมุนตามเข็ม
5V / PWM	0	1	หมุนทวนเข็ม
5V /PWM	0	0	หยุดการหมุนทันที

(ที่มา: http://www.es.co.th/detail.asp?prod=089000021)

2.7 ไมโครคอนโทรลเลอร์ (Microcontroller)

ในโครงงานนี้ได้เลือกใช้ไมโครคอนโทรลเลอร์ตระกูล Arduino เนื่องจากใช้งานง่าย มีองค์ประกอบเหมาะสมสำหรับใช้ในโครงงานการโปรแกรมข้อมูลเป็นแบบ open source ซึ่งสามารถเขียนโปรแกรมและนำโปรแกรมที่มีผู้ทำไว้แล้วมาปรับเปลี่ยนประยุกต์ได้ตามต้องการ อีกทั้งโปรแกรมยังง่ายต่อการพัฒนา เนื่องจากคำสั่งที่ใช้เขียนโปรแกรมเป็นภาษาชั้นสูง มีคำสั่งเขียนโปรแกรมและตัวอย่างต่างๆ รองรับ สามารถเลือกใช้ได้ตามต้องการ รองรับการทำงานได้ทั้งระบบปฏิบัติการ Windows Linux และ Mac OS X

สำหรับไมโครคอนโทรลเลอร์ที่เลือกใช้คือ “Arduino Due” ดังแสดงในรูปที่ 2.13 และ “Arduino Uno R3” ดังแสดงในรูปที่ 2.14



รูปที่ 2.13 Arduino Due

(ที่มา: https://store.arduino.cc/usa/arduino-due)

รูปที่ 2.14 Arduino Uno R3

(ที่มา: https://store.arduino.cc/usa/arduino-uno-rev3)

เหตุผลที่ต้องใช้ไมโครคอนโทรลเลอร์ถึง 2 ตัว เนื่องจากโครงงานนี้กำหนดให้ผู้ใช้งานสามารถเลือกสั่งงานหุ่นยนต์จาก Joystick ได้ โดยเลือกใช้อุปกรณ์ Joystick PS2 ชนิด Wireless ดังแสดงในรูปที่ 2.15 ซึ่งไม่รองรับการใช้งานร่วมกับบอร์ด Arduino Due จึงใช้บอร์ด Arduino Uno R3 เพื่อรับค่าคำสั่งจากการกดปุ่มบน Joystick จากนั้นส่งค่าไปยังบอร์ด Arduino Due ผ่านทางพอร์ตอนุกรม

สำหรับบอร์ด Arduino Due จะมีหน้าที่ในการรับค่าจากคอมพิวเตอร์ มอเตอร์เอนโค้ดเดอร์ และเซนเซอร์อัลทราโซนิก จากนั้นสั่งงานไดร์เวอร์มอเตอร์เพื่อควบคุมการเคลื่อนที่ของหุ่นยนต์ต่อไป

รูปที่ 2.15 Joystick PS2 wireless

(ที่มา: http://www.myarduino.net/p/172)

2.8 หลักการทำงานของเซนเซอร์อัลทราโซนิก (Ultrasonic Sensors)

เซนเซอร์อัลทราโซนิก คืออุปกรณ์ที่ใช้วัดระยะห่างของวัตถุ ดังแสดงในรูปที่ 2.16 ซึ่งเซนเซอร์ชนิดนี้มีความแม่นยำในการตรวจจับ แม้ในสภาวะที่ไม่เอื้ออำนวย เช่น วัตถุมีฝุ่นผง กรณีวัตถุมีความโปร่งใส โปร่งแสง มีความแวววาวก็สามารถตรวจวัดระยะได้ อีกทั้งยังเหมาะสำหรับการตรวจจับของเหลวและวัตถุที่เป็นเม็ดได้เป็นอย่างดี ซึ่งเซนเซอร์นี้จะอาศัยคลื่นที่เรียกว่า “คลื่นอัลทราโซนิก”

รูปที่ 2.16 เซนเซอร์อัลทราโซนิก

(ที่มา: https://www.robot-electronics.co.uk/htm/srf05tech.htm)

คลื่นอัลทราโซนิกคือคลื่นเสียงที่มีความถี่สูงเกิน 20,000 Hz จนหูมนุษย์ไม่สามารถได้ยินได้ คลื่นเสียงที่มนุษย์ได้ยินจะอยู่ในช่วงความถี่ระหว่าง 20 -20,000 เฮิรตซ์ (hertz) แต่คลื่นอัลทราโซนิกจะมีความถี่ตั้งแต่ 20 kHz ขึ้นไป ความถี่นี้จะมีสัตว์บางประเภทที่สามารถได้ยินได้ เช่น ค้างคาวและ

โลมา เนื่องจากค้างคาวมีดวงตาที่เล็กและออกหากินในเวลากลางคืน ทำให้ค้างคาวต้องมีสิ่งที่มาทดแทนคือคลื่นอัลทราโซนิก ซึ่งค้างคาวจะใช้คลื่นความถี่อัลทราโซนิกเพื่อระบุตำแหน่ง รูปร่าง ทิศทาง ของวัตถุที่ขวางเส้นทางการเดินทาง ด้วยหลักการสะท้อนของคลื่นเสียงที่ว่ามุมตกกระทบเท่ากับมุมสะท้อน คลื่นเสียงเคลื่อนที่ในอากาศด้วยความเร็ว 343 เมตรต่อวินาที ที่อุณหภูมิ 25℃ คลื่นย่านอัลทราโซนิกเป็นคลื่นที่มีทิศทางจึงสามารถเล็งคลื่นไปที่ตำแหน่งหรือเป้าหมายที่ต้องการได้อย่างแม่นยำ

หลักการคำนวณระยะห่างระหว่างอุปกรณ์เซนเซอร์อัลทราโซนิกกับวัตถุ จะเป็นไปตามสมการการเคลื่อนที่ในแนวเส้นตรง ดังสมการต่อไปนี้

$$S = VT \tag{2.1}$$

S = ระยะทาง (เมตร, m)

V = ความเร็วของคลื่นเสียง (343 เมตรต่อวินาที, m/s)

T = ระยะเวลาที่คลื่นเสียงเดินทางทั้งหมด (วินาที, s)

ภายในอุปกรณ์เซนเซอร์อัลทราโซนิกจะทำงานโดยส่งสัญญาณนาฬิกา (ตัวจับเวลา) ไปที่ตัวควบคุมการทำงานเพื่อควบคุมการแปลงสัญญาณ แล้วจึงส่งสัญญาณไปที่ตัวรับคลื่นและตัวส่งคลื่นอัลทราโซนิก ตัวส่งคลื่นจะสร้างคลื่นเสียงอัลทราโซนิกจากสัญญาณไฟฟ้าแล้วส่งคลื่นเสียงความถี่สูงออกไปเป็นแนวตรง เมื่อคลื่นเสียงไปกระทบกับวัตถุ คลื่นเสียงจะสะท้อนกลับมายังตัวรับ ตัวรับจะแปลงคลื่นเสียงอัลทราโซนิกนั้นเป็นสัญญาณไฟฟ้าแล้วส่งต่อให้กับตัวประมวลผล ตัวประมวลผลจะคำนวณค่าระยะห่างจากระยะทางที่คลื่นเสียงเดินทางไปและกลับ และส่งค่าที่คำนวณได้ให้กับตัวส่งสัญญาณเอาต์พุต สัญญาณที่ได้จะถูกส่งให้กับอุปกรณ์ที่เชื่อมต่อกับอุปกรณ์เซนเซอร์อัลทราโซนิกต่อไป

2.9 อุปกรณ์ไฟฟ้าสำรอง (Uninterruptible Power Supply)

อุปกรณ์ไฟฟ้าสำรอง คือ อุปกรณ์ไฟฟ้าชนิดหนึ่งที่สามารถทำการจ่ายพลังงานไฟฟ้าให้กับอุปกรณ์อิเล็กทรอนิกส์ได้อย่างต่อเนื่องแม้ในเวลาที่เกิดไฟดับหรือเกิดปัญหาแรงดันไฟฟ้าผันผวนผิดปกติโดยอุปกรณ์ไฟฟ้าสำรองจะทำการปรับระดับแรงดันไฟฟ้าให้คงที่อยู่ในระดับที่ปลอดภัยต่ออุปกรณ์ไฟฟ้าและอุปกรณ์อิเล็กทรอนิกส์มีหน้าที่หลัก คือ ป้องกันความเสียหายที่สามารถเกิดขึ้นกับ

อุปกรณ์ไฟฟ้าและอุปกรณ์อิเล็กทรอนิกส์ (โดยเฉพาะคอมพิวเตอร์และอุปกรณ์เชื่อมต่อ) อันมีสาเหตุจากความผิดปกติของพลังงานไฟฟ้า เช่น ไฟตก ไฟดับ ไฟกระชาก และไฟเกิน เป็นต้น รวมถึงมีหน้าที่ในการจ่ายพลังงานไฟฟ้าสำรองจากแบตเตอรี่ให้แก่อุปกรณ์ไฟฟ้าหรืออุปกรณ์อิเล็กทรอนิกส์เมื่อเกิดปัญหาทางไฟฟ้าซึ่งส่วนประกอบหลักของอุปกรณ์ไฟฟ้าสำรอง ประกอบไปด้วย เครื่องประจุแบตเตอรี่ (Charger) หรือ เครื่องแปลงกระแสไฟฟ้าจาก AC เป็น DC (Rectifier) ทำหน้าที่รับกระแสไฟฟ้า AC จากระบบจ่ายไฟ แปลงกระแสไฟฟ้า DC จากนั้นประจุเก็บไว้ในแบตเตอรี่ เครื่องแปลงกระแสไฟฟ้า (Inverter) ทำหน้าที่รับกระแสไฟฟ้า DC จากเครื่องแปลงกระแสไฟฟ้า AC เป็น DC หรือแบตเตอรี่และแปลงเป็นกระแสไฟฟ้า AC สำหรับใช้กับอุปกรณ์ไฟฟ้าและอุปกรณ์อิเล็กทรอนิกส์ แบตเตอรี่ (Battery) ทำหน้าที่เก็บพลังงานไฟฟ้าสำรองไว้ใช้ในกรณีเกิดปัญหาทางไฟฟ้า โดยจะจ่ายกระแสไฟฟ้า DC ให้กับเครื่องแปลงกระแสไฟฟ้าในกรณีที่ไม่สามารถรับกระแสไฟฟ้า AC จากระบบจ่ายไฟได้

โดยทั่วไปแล้วเมื่ออุปกรณ์ไฟฟ้าสำรองรับพลังงานไฟฟ้าเข้ามา ไม่ว่าคุณภาพไฟฟ้าจะเป็นอย่างไรก็จะสามารถจ่ายพลังงานไฟฟ้าให้กับอุปกรณ์ไฟฟ้าได้เป็นปกติ รวมถึงทำการจ่ายพลังงานไฟฟ้าสำรองที่เก็บไว้ในแบตเตอรี่ให้กับอุปกรณ์ไฟฟ้า ซึ่งหลักการของอุปกรณ์ไฟฟ้าสำรอง ก็คือใช้วิธีการแปลงไฟฟ้าจากกระแสสลับ (AC) เป็นไฟฟ้ากระแสตรง (DC) แล้วเก็บสำรองไว้ในแบตเตอรี่ส่วนหนึ่ง เมื่อเกิดปัญหาทางไฟฟ้า (เช่น ไฟดับ หรือ หรือคุณภาพไฟฟ้าผิดปกติ เป็นต้น) อุปกรณ์ไฟฟ้าไม่สามารถใช้พลังงานไฟฟ้าที่รับมาได้ UPS ก็จะเปลี่ยนไฟฟ้ากระแสตรง (DC) จากแบตเตอรี่ให้กลายเป็นไฟฟ้ากระแสสลับ (AC) แล้วจึงจะจ่ายพลังงานไฟฟ้าให้กับอุปกรณ์ไฟฟ้าตามปกติ

อุปกรณ์ไฟฟ้าสำรองที่เลือกใช้แสดงดังรูปที่ 2.17

รูปที่ 2.17 อุปกรณ์ไฟฟ้าสำรองที่เลือกใช้

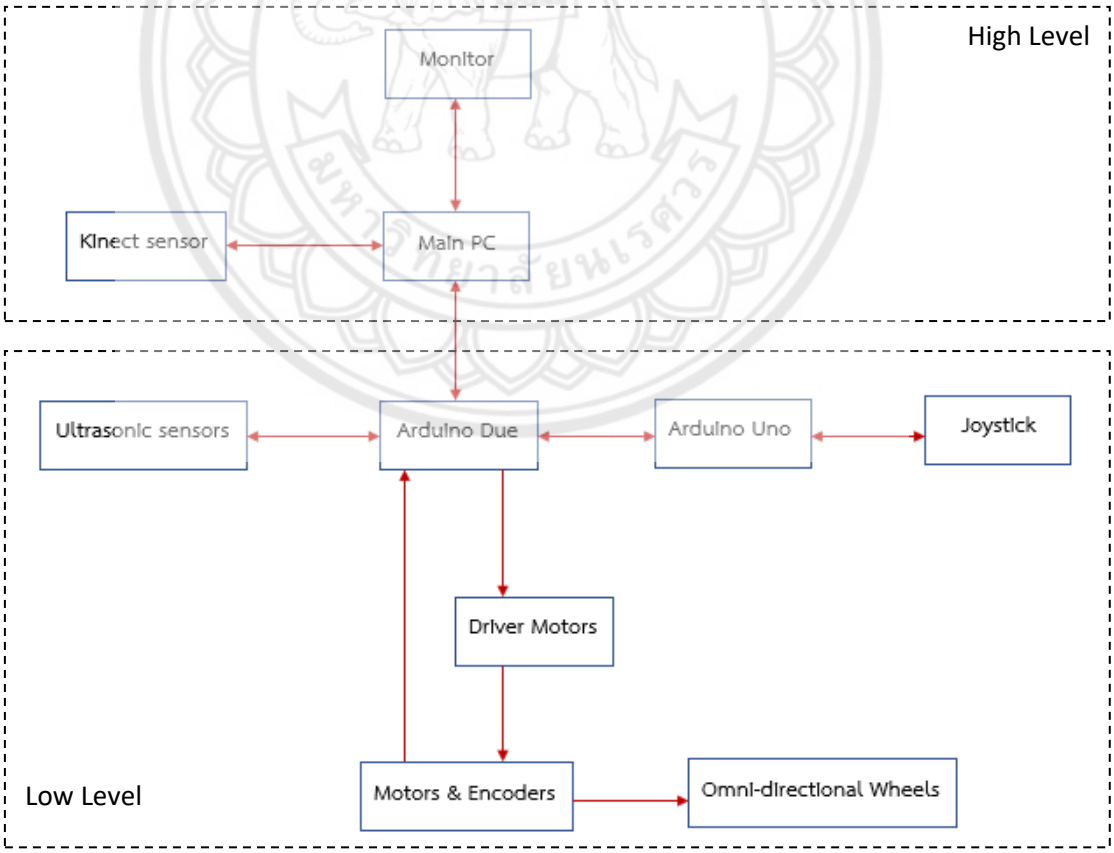
(ที่มา: https://www.advice.co.th/product/ups/etech/900va-etech-eazy-2-by-zircon)

บทที่ 3
วิธีการดำเนินโครงงาน

เนื้อหาในบทนี้ประกอบด้วยการออกแบบหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง คุณสมบัติและความสามารถของหุ่นยนต์ รวมทั้งหลักการทำงานของหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง

3.1 การออกแบบภาพรวมการทำงานของระบบ

กำหนดให้การทำงานของหุ่นยนต์สามารถถูกแบ่งออกเป็น 2 ส่วนหลักๆ คือ ส่วนแรก เป็นส่วนควบคุมระดับบน มีคอมพิวเตอร์ส่วนบุคคลขนาดเล็กนำเข้าข้อมูลจากอุปกรณ์ Kinect มาเพื่อประมวลผลเป็นหลัก และส่วนที่สอง เป็นส่วนควบคุมระดับล่าง มีไมโครคอนโทรลเลอร์ Arduino Due คอยควบคุมการเคลื่อนที่ของหุ่นยนต์เป็นหลัก รับข้อมูลคำสั่งจากคอมพิวเตอร์ส่วนบุคคลขนาดเล็ก มีการทำงานดังแสดงในรูปที่ 3.1



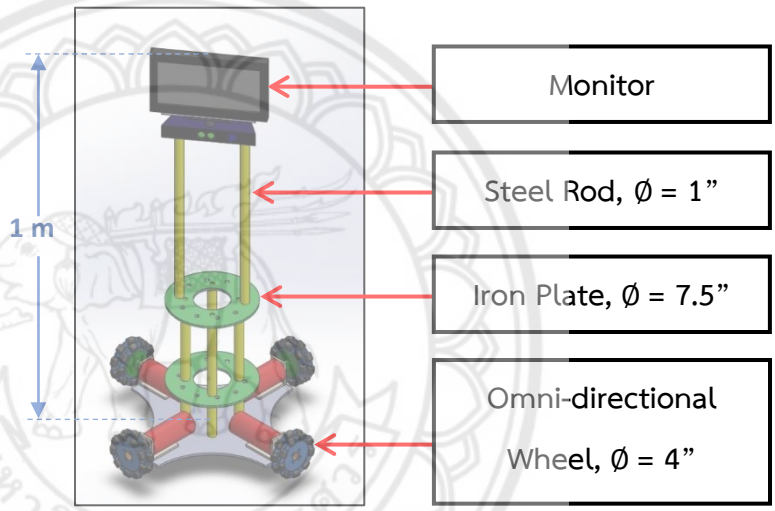
รูปที่ 3.1 การทำงานโดยรวมของระบบทั้งหมด

ผู้จัดทำเลือกใช้โปรแกรม Visual Studio ในการพัฒนาโปรแกรมซึ่งใช้กับการควบคุมระดับบน และใช้โปรแกรม Arduino IDE ในการพัฒนาโปรแกรมซึ่งใช้กับการควบคุมระดับล่าง

กำหนดให้ผู้ใช้งานสามารถเลือกวิธีการสั่งงานหุ่นยนต์ได้ 3 ทางเลือก ได้แก่ สั่งงานด้วยการกดปุ่มบน Joystick สั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์ และสั่งงานระยะไกลด้วยท่าทาง ซึ่งจะได้มีการกล่าวถึงโดยละเอียดในหัวข้อที่ 3.4

3.2 การออกแบบโครงสร้างของหุ่นยนต์และการติดตั้งอุปกรณ์

โครงสร้างของหุ่นยนต์ซึ่งได้ออกแบบไว้สามารถแสดงได้ดังรูปที่ 3.2

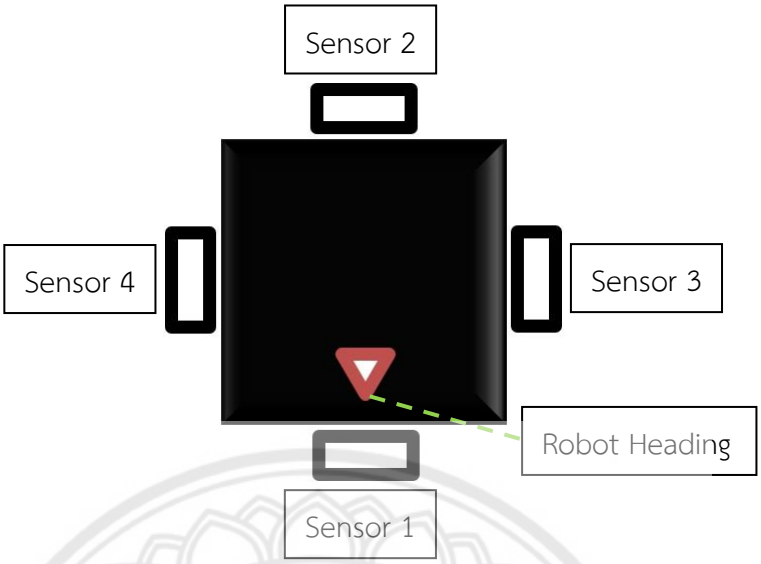


รูปที่ 3.2 โครงสร้างของหุ่นยนต์ซึ่งได้ออกแบบไว้

โดยเมื่อทำการจัดสร้างและติดตั้งอุปกรณ์ลงบนโครงสร้างหลักแล้วสามารถแสดงได้ดังรูปที่ 3.3

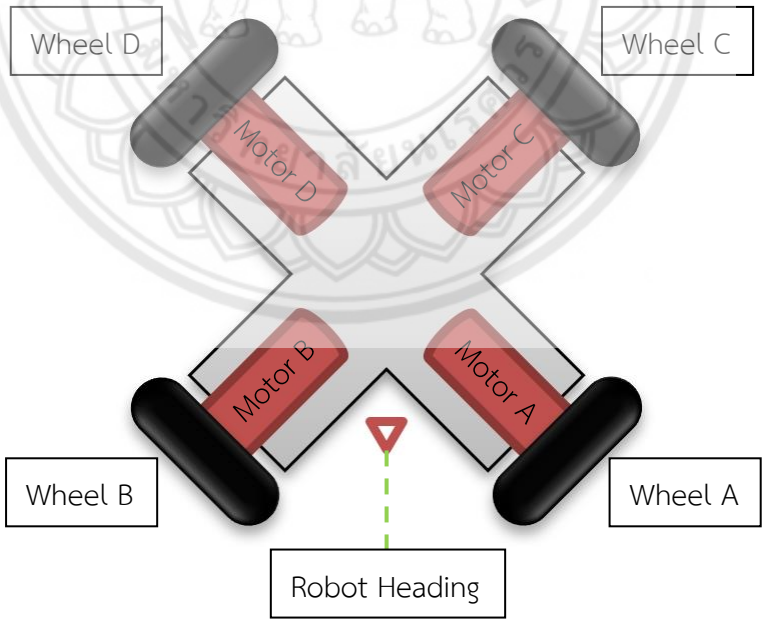
รูปที่ 3.3 ตำแหน่งของอุปกรณ์ต่างๆ ที่ติดตั้งลงบนโครงสร้างของหุ่นยนต์

กำหนดชื่อเซนเซอร์อัลทราโซนิกซึ่งติดตั้งลงบนตัวหุ่นยนต์ในตำแหน่งต่างๆ ดังรูปที่ **3.4**



รูปที่ 3.4 ตำแหน่งของเซนเซอร์อัลทราโซนิกทั้งสี่ตัว

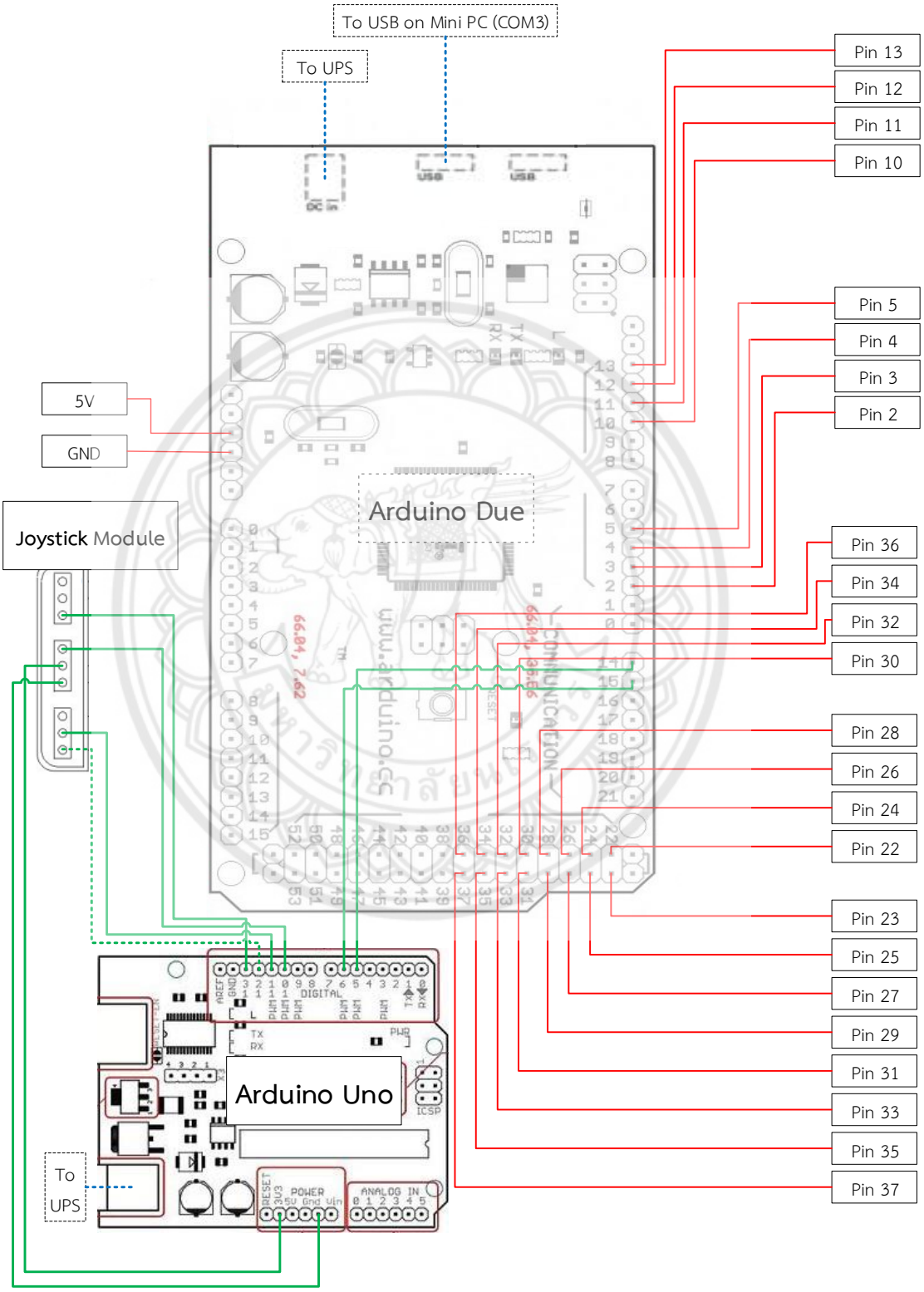
กำหนดชื่อล้อและมอเตอร์ซึ่งติดตั้งลงบนตัวหุ่นยนต์ในตำแหน่งต่างๆ ดังรูปที่ **3.5**



รูปที่ 3.5 ตำแหน่งของมอเตอร์และล้อทั้งสี่

3.3 การออกแบบการเชื่อมต่ออุปกรณ์

แบ่งกลุ่มอุปกรณ์เพื่อแสดงการเชื่อมต่อทั้งหมด ออกเป็น 4 กลุ่ม การเชื่อมต่ออุปกรณ์กลุ่มที่ 1 แสดงดังรูปที่ 3.6 พิจารณาตารางที่ 3.1 และ 3.2 ประกอบ



รูปที่ 3.6 การเชื่อมต่ออุปกรณ์กลุ่มที่ 1

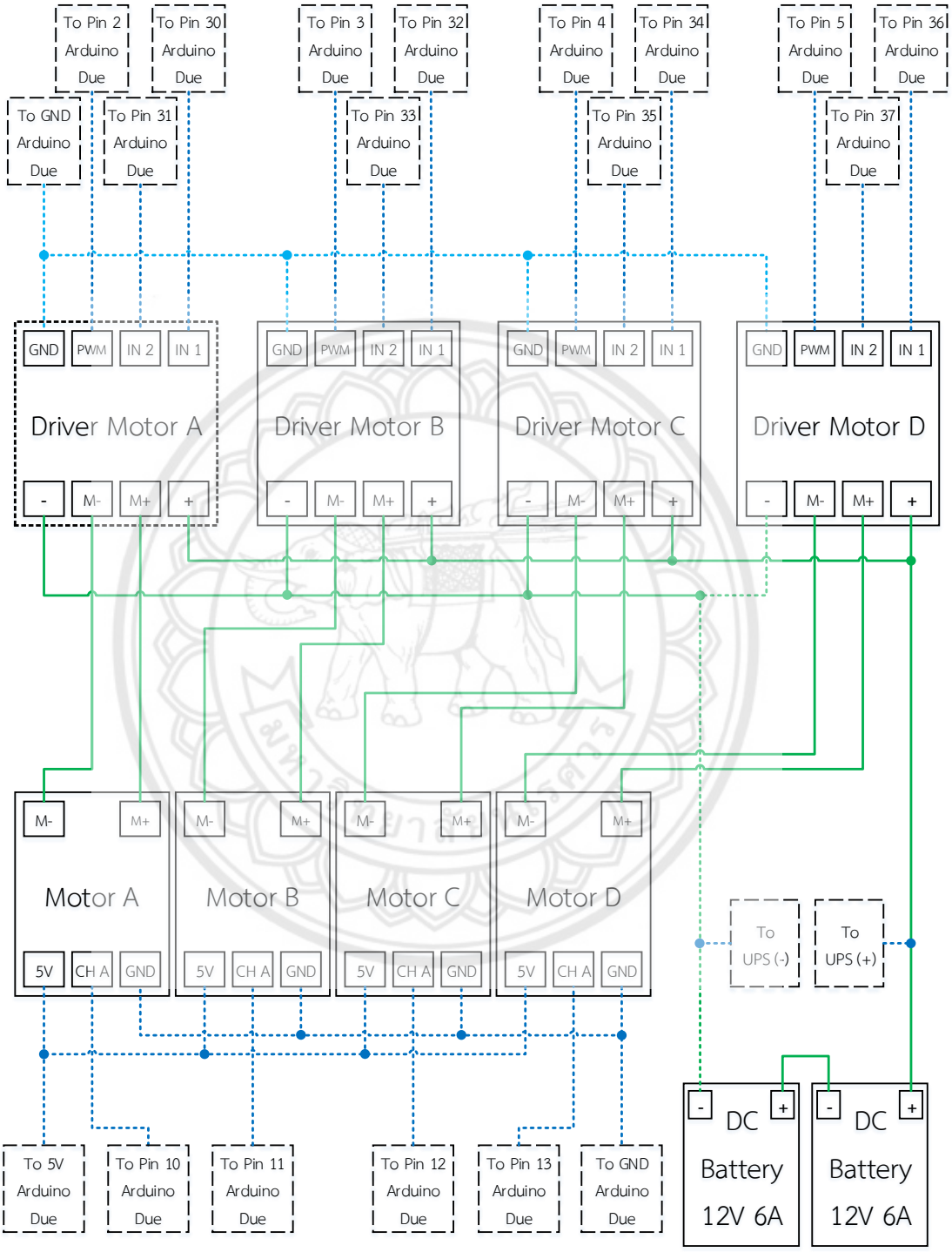
ตารางที่ 3.1 พินของบอร์ด Arduino Due ที่ใช้งานเพื่อส่งค่า

หมายเลขพิน	การใช้งาน
2	ส่งสัญญาณ PWM ไปยังไดร์เวอร์มอเตอร์ A
3	ส่งสัญญาณ PWM ไปยังไดร์เวอร์มอเตอร์ B
4	ส่งสัญญาณ PWM ไปยังไดร์เวอร์มอเตอร์ C
5	ส่งสัญญาณ PWM ไปยังไดร์เวอร์มอเตอร์ D
14	ส่งการสื่อสารอนุกรมไปยังพินที่ 5 ของบอร์ด Arduino Uno
23	ส่งค่าไปยังเซนเซอร์อัลทราโซนิกตัวหน้า (Sensor 1) เพื่อให้ส่งคลื่นออกไป
25	ส่งค่าไปยังเซนเซอร์อัลทราโซนิกตัวหลัง (Sensor 2) เพื่อให้ส่งคลื่นออกไป
27	ส่งค่าไปยังเซนเซอร์อัลทราโซนิกตัวซ้าย (Sensor 3) เพื่อให้ส่งคลื่นออกไป
29	ส่งค่าไปยังเซนเซอร์อัลทราโซนิกตัวขวา (Sensor 4) เพื่อให้ส่งคลื่นออกไป
30	ส่งค่าไปยังบอร์ดไดร์เวอร์มอเตอร์เพื่อกำหนดทิศการหมุนของมอเตอร์ A
31	ส่งค่าไปยังบอร์ดไดร์เวอร์มอเตอร์เพื่อกำหนดทิศการหมุนของมอเตอร์ A
32	ส่งค่าไปยังบอร์ดไดร์เวอร์มอเตอร์เพื่อกำหนดทิศการหมุนของมอเตอร์ B
33	ส่งค่าไปยังบอร์ดไดร์เวอร์มอเตอร์เพื่อกำหนดทิศการหมุนของมอเตอร์ B
34	ส่งค่าไปยังบอร์ดไดร์เวอร์มอเตอร์เพื่อกำหนดทิศการหมุนของมอเตอร์ C
35	ส่งค่าไปยังบอร์ดไดร์เวอร์มอเตอร์เพื่อกำหนดทิศการหมุนของมอเตอร์ C
36	ส่งค่าไปยังบอร์ดไดร์เวอร์มอเตอร์เพื่อกำหนดทิศการหมุนของมอเตอร์ D
37	ส่งค่าไปยังบอร์ดไดร์เวอร์มอเตอร์เพื่อกำหนดทิศการหมุนของมอเตอร์ D

ตารางที่ 3.2 พินของบอร์ด Arduino Due ที่ใช้งานเพื่อรับค่า

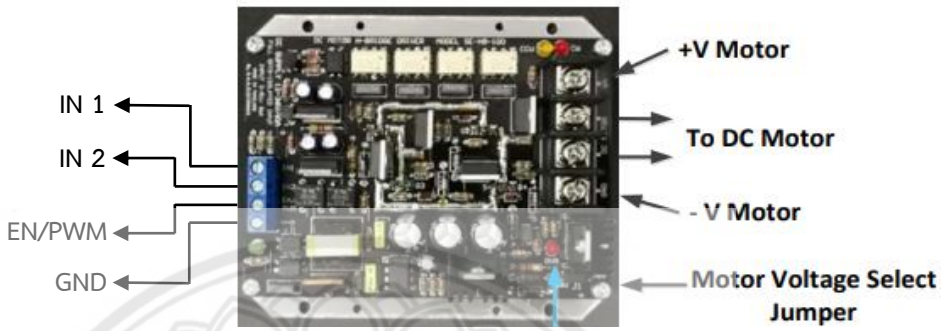
หมายเลขพิน	การใช้งาน
10	รับสัญญาณเอ็นโค้ดเดอร์จากมอเตอร์ A
11	รับสัญญาณเอ็นโค้ดเดอร์จากมอเตอร์ B
12	รับสัญญาณเอ็นโค้ดเดอร์จากมอเตอร์ C
13	รับสัญญาณเอ็นโค้ดเดอร์จากมอเตอร์ D
15	รับการสื่อสารอนุกรมจากพินที่ 6 ของบอร์ด Arduino Uno
22	รับค่าจากเซนเซอร์อัลทราโซนิกตัวหน้า (Sensor 1)
24	รับค่าจากเซนเซอร์อัลทราโซนิกตัวหลัง (Sensor 2)
26	รับค่าจากเซนเซอร์อัลทราโซนิกตัวซ้าย (Sensor 3)
28	รับค่าจากเซนเซอร์อัลทราโซนิกตัวขวา (Sensor 4)

การเชื่อมต่ออุปกรณ์กลุ่มที่ 2 แสดงดังรูปที่ 3.7 บอร์ดไดร์เวอร์มอเตอร์จำนวน 4 บอร์ดจะใช้ควบคุมการทำงานของล้อ 4 ล้อ คือ ล้อ A ล้อ B ล้อ C และ ล้อ D



รูปที่ 3.7 การเชื่อมต่ออุปกรณ์กลุ่มที่ 2

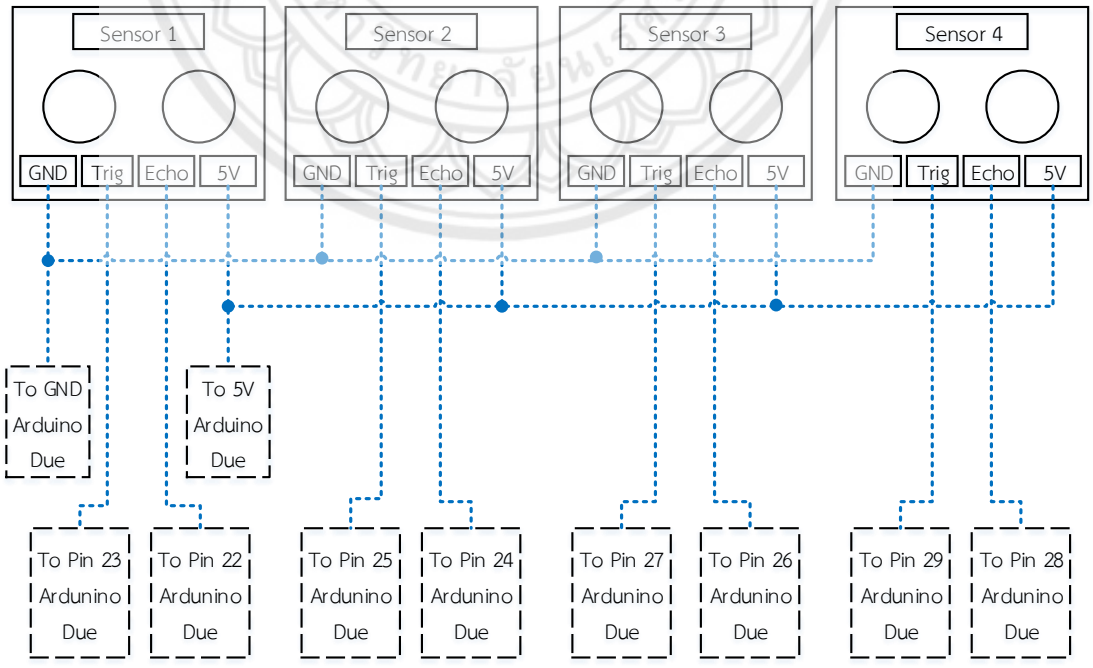
พินที่ไดร์เวอร์มอเตอร์ใช้รับค่ามีรายละเอียดดังแสดงในรูปที่ 3.8 นั่นคือ พิน IN 1 และ IN 2 ใช้รับค่าลอจิกจากบอร์ด Arduino Due เพื่อสั่งให้มอเตอร์หมุนตามทิศทางที่กำหนดโดยจะส่งเป็น 2 สถานะ คือ หากให้ลอจิก 1 ที่ IN 1 ลอจิก 0 ที่ IN 2 มอเตอร์จะหมุนไปทิศหนึ่ง หากให้ลอจิก 0 ที่ IN 1 ลอจิก 1 ที่ IN 2 มอเตอร์ก็จะหมุนกลับทิศ



รูปที่ 3.8 บอร์ดไดร์เวอร์มอเตอร์

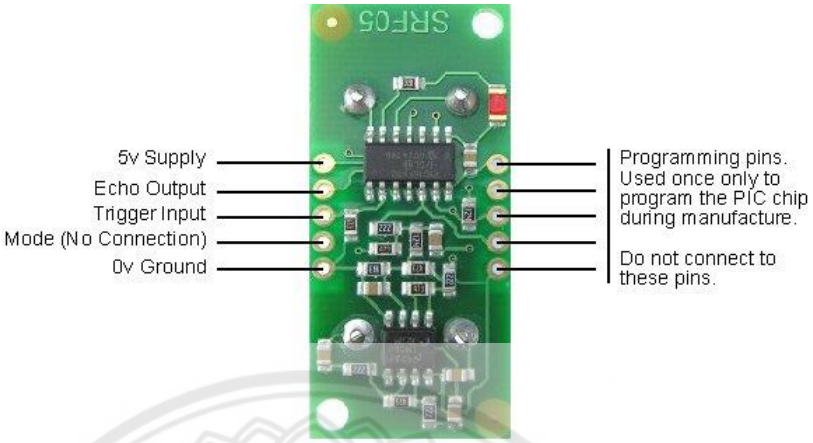
(ที่มา: http://www.es.co.th/detail.asp?prod=089000021)

โครงงานนี้ใช้เซนเซอร์อัลทราโซนิกจำนวน 4 ตัว ติดตั้งทั้ง 4 ด้านบนตัวหุ่นยนต์ คือ ด้านหน้าตัวด้านหลัง ตัวด้านซ้าย และตัวด้านขวา การเชื่อมต่ออุปกรณ์กลุ่มที่ 3 แสดงดังรูปที่ 3.9



รูปที่ 3.9 การเชื่อมต่ออุปกรณ์กลุ่มที่ 3

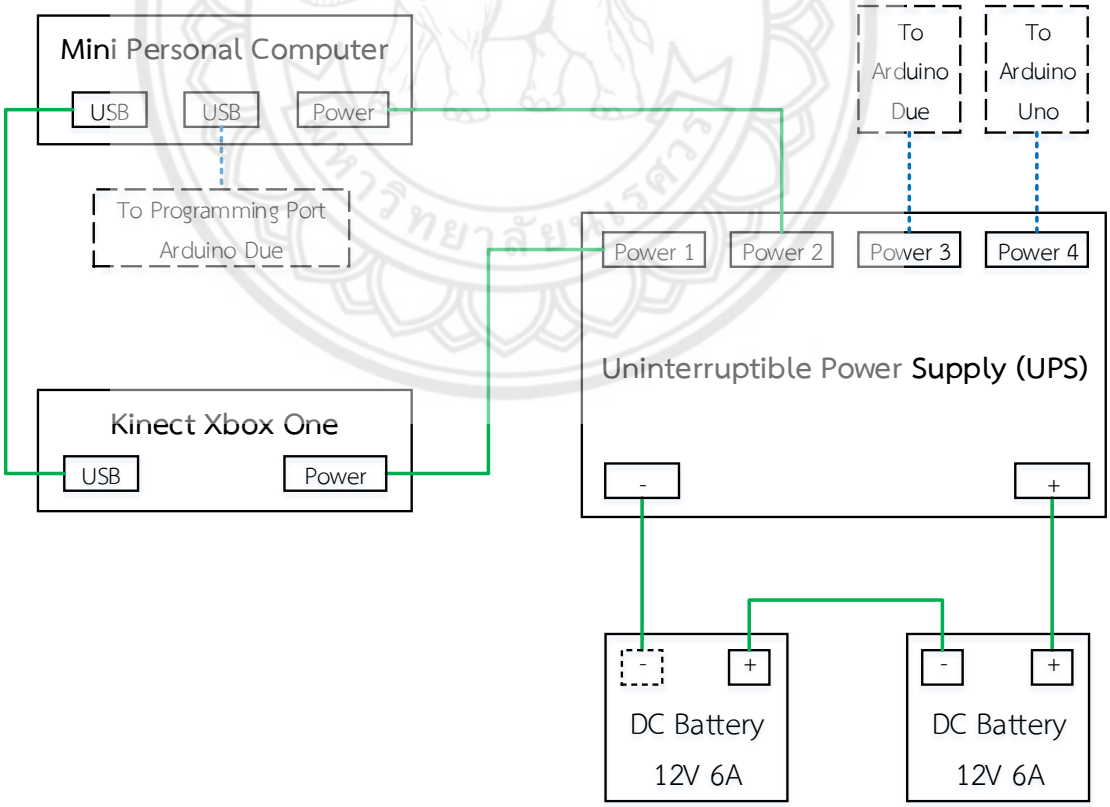
พินการใช้งานต่างๆ ของอุปกรณ์เซนเซอร์อัลทราโซนิกที่เลือกใช้แสดงดังรูปที่ 3.10



รูปที่ 3.10 อุปกรณ์เซนเซอร์อัลทราโซนิก

(ที่มา: https://www.robot-electronics.co.uk/htm/srf05tech.htm)

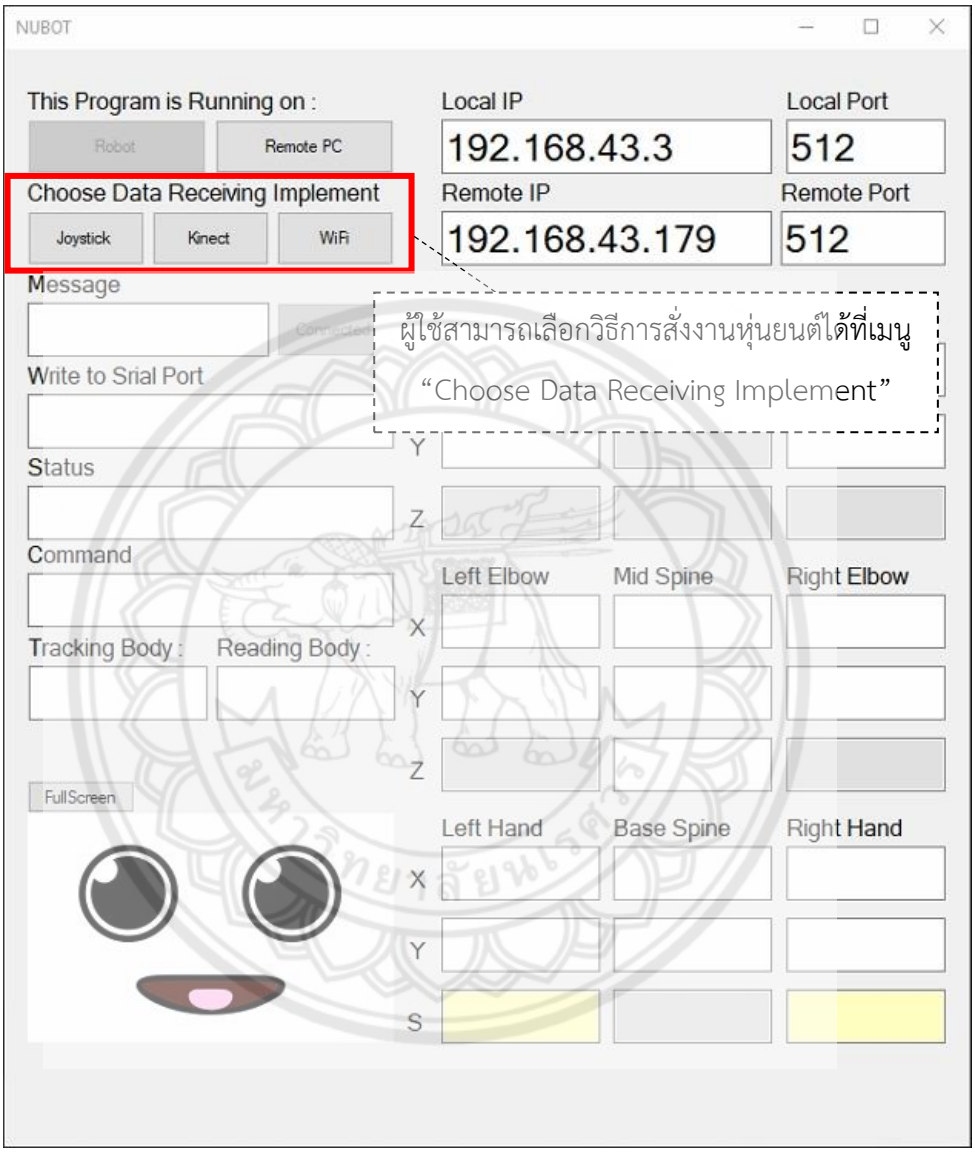
การเชื่อมต่ออุปกรณ์กลุ่มที่ 4 แสดงดังรูปที่ 3.11



รูปที่ 3.11 การเชื่อมต่ออุปกรณ์กลุ่มที่ 4

3.4 การออกแบบและพัฒนาวิธีการเลือกรับคำสั่งของหุ่นยนต์

กำหนดให้ผู้ใช้งานสามารถเลือกวิธีการสั่งงานหุ่นยนต์ได้จากส่วนต่อประสานกับผู้ใช้ (User Interface, UI) ซึ่งผู้จัดทำได้พัฒนาขึ้น ดังแสดงในรูปที่ 3.12



รูปที่ 3.12 ส่วนต่อประสานกับผู้ใช้ (User Interface, UI)

เมื่อผู้ใช้งานเลือกวิธีการสั่งงานหุ่นยนต์จากส่วนต่อประสานกับผู้ใช้ จะมีการส่งรหัส ASCII ผ่านพอร์ตอนุกรมจากคอมพิวเตอร์บนตัวหุ่นยนต์ไปยังบอร์ด Arduino Due โดยที่ หากเลือกสั่งงานด้วยการกดปุ่มบน Joystick จะส่ง “O” และ หากเลือกสั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์ หรือ สั่งงานระยะไกลด้วยท่าทาง จะส่ง “K”

ทางเลือกในการสั่งงานหุ่นยนต์มีทั้งหมด 3 ทางเลือก ดังนี้

3.4.1 สั่งงานด้วยการกดปุ่มบน Joystick

ใช้ Joystick แบบไร้สาย โดยเชื่อมต่ออุปกรณ์ฝั่งรับกับบอร์ด Arduino Uno บอร์ด Arduino Uno ทำหน้าที่ประมวลผลรหัสของการกดปุ่มบน Joystick ที่ได้รับการส่งข้อมูลเข้ามา จากนั้นส่งข้อมูลรหัสการเคลื่อนที่ตามที่ได้เขียนโปรแกรมไว้ต่อไปยังบอร์ด Arduino Due ผ่านการสื่อสารอนุกรมเพื่อควบคุมการเคลื่อนที่ของหุ่นยนต์ต่อไป

3.4.2 สั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์ (Kinect on Robot)

ผู้ใช้งานแสดงท่าทางให้ตรงตามเงื่อนไขเพื่อสั่งให้หุ่นยนต์เคลื่อนที่ตามการสั่งงาน โดยเครื่องคอมพิวเตอร์ส่วนบุคคลขนาดเล็กบนตัวหุ่นยนต์ซึ่งเชื่อมต่อกับเซนเซอร์ Kinect จะทำการประมวลผลภาพให้ได้เป็นค่าพิกัดข้อต่อของร่างกายซึ่งเป็นพิกัดฉาก แล้วจึงประมวลผลคำสั่งจากข้อมูลที่ได้ แล้วส่งข้อมูลรหัสการเคลื่อนที่ไปที่บอร์ด Arduino Due ผ่านการสื่อสารอนุกรมเพื่อควบคุมการเคลื่อนที่ของหุ่นยนต์ต่อไป

3.4.3 สั่งงานระยะไกลด้วยท่าทาง (Kinect on Remote PC)

มีการประมวลผลเช่นเดียวกันกับการสั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์ แต่จะใช้คอมพิวเตอร์ส่วนบุคคลและเซนเซอร์ Kinect ที่ตั้งอยู่ภายนอกตัวหุ่นยนต์ คอมพิวเตอร์ภายนอกจะส่งข้อมูลรหัสการเคลื่อนที่ไปยังคอมพิวเตอร์บนตัวหุ่นยนต์ผ่านการเชื่อมต่อ Wi-Fi (Wireless Fidelity) จากนั้นคอมพิวเตอร์บนตัวหุ่นยนต์จะส่งข้อมูลรหัสการเคลื่อนที่ต่อไปที่บอร์ด Arduino Due ผ่านการสื่อสารอนุกรมเพื่อควบคุมการเคลื่อนที่ของหุ่นยนต์ต่อไป

สาเหตุที่คณะผู้จัดทำตัดสินใจเพิ่มทางเลือกในการใช้งานแบบสั่งงานระยะไกลด้วยท่าทาง เนื่องมาจากการสั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์ มีข้อจำกัดในการใช้งาน คือ เมื่อหุ่นยนต์เริ่มเคลื่อนที่ เซนเซอร์ Kinect มักเกิดการสั่นสะเทือน ส่งผลให้ในบางช่วงไม่สามารถประมวลผลภาพหาค่าพิกัดข้อต่อของร่างกายได้ การใช้งานแบบสั่งงานระยะไกลด้วยท่าทางแสดงให้เห็นว่าเมื่อไม่มีการสั่นสะเทือนของเซนเซอร์ Kinect หุ่นยนต์จะสามารถทำงานได้อย่างต่อเนื่อง

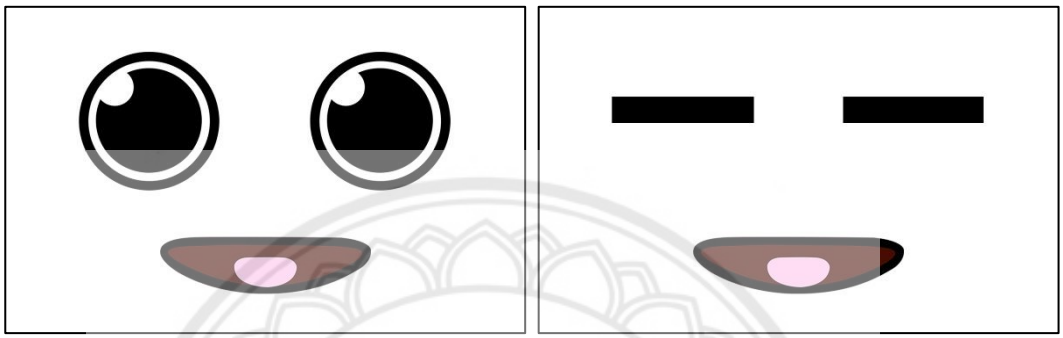
3.5 การออกแบบและพัฒนาการเปลี่ยนสถานะของหุ่นยนต์

กำหนดให้การสั่งงานด้วยท่าทางมีสถานะการทำงาน 3 สถานะ หุ่นยนต์จะมีการกระพริบตาและเปลี่ยนสีหน้าไปตามสถานะการทำงาน

3.5.1 สถานะรอ (Standby)

เป็นสถานะเริ่มต้นเมื่อหุ่นยนต์เริ่มทำงาน โดยในสถานะนี้หุ่นยนต์จะตรวจหาภายในกรอบภาพที่เซนเซอร์ Kinect สามารถตรวจวัดได้ ว่ามีบุคคลใดหรือไม่ที่ทำท่าทางสอดคล้องกับเงื่อนไขที่เป็นการสั่งให้เข้าสู่สถานะรับคำสั่ง

หน้าตาของหุ่นยนต์เมื่ออยู่ในสถานะรอแสดงดังรูปที่ 3.13

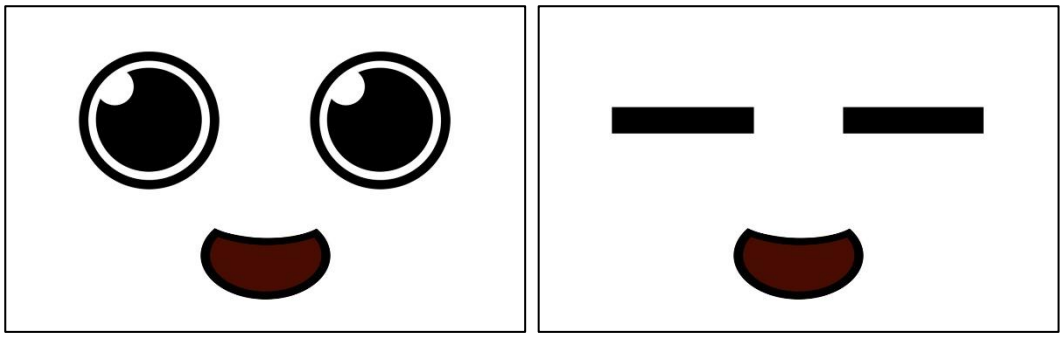


รูปที่ 3.13 หน้าตาของหุ่นยนต์เมื่ออยู่ในสถานะรอ

3.5.2 สถานะรับคำสั่ง (Controlled)

บุคคลแรกที่หุ่นยนต์พบว่าทำท่าทางตรงตามเงื่อนไขที่เป็นการสั่งให้เข้าสู่สถานะรับคำสั่งจะถูกกำหนดให้เป็น “ผู้สั่งงาน” จนกว่าจะมีการเปลี่ยนกลับไปสู่สถานะรอ มีเพียงผู้สั่งงานเท่านั้นที่สามารถสั่งให้หุ่นยนต์เคลื่อนที่ตามการสั่งงาน กลับสู่สถานะรอ หรือเข้าสู่สถานะติดตามได้ โดยการทำท่าทางให้ตรงตามเงื่อนไข เพื่อลดความเป็นไปได้ในการสั่งให้หุ่นยนต์เข้าสู่สถานะติดตามโดยที่ไม่ได้ตั้งใจ จึงกำหนดให้การเข้าสู่สถานะติดตามจำเป็นต้องเข้าสู่สถานะรับคำสั่งเสียก่อน

หน้าตาของหุ่นยนต์เมื่ออยู่ในสถานะรับคำสั่งแสดงดังรูปที่ 3.14

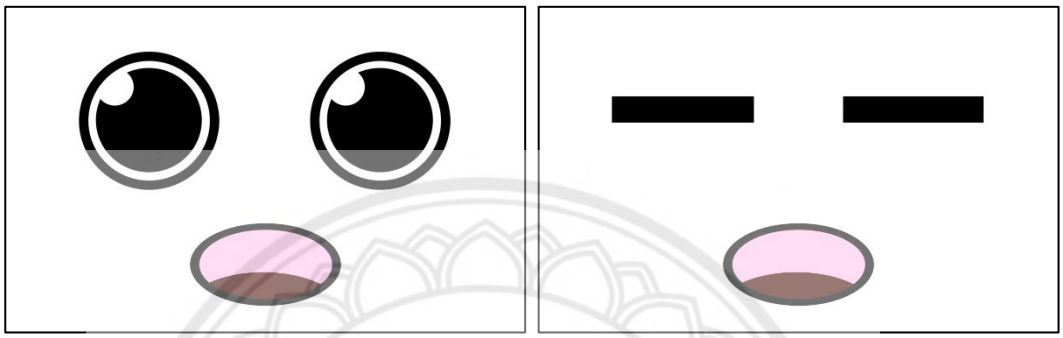


รูปที่ 3.14 หน้าตาของหุ่นยนต์เมื่ออยู่ในสถานะรับคำสั่ง

3.5.3 สถานะติดตาม (Follow)

เมื่อเข้าสู่สถานะติดตาม หุ่นยนต์พยายามรักษามุมและระยะห่างระหว่างตัวหุ่นยนต์กับผู้สั่งงาน โดยจะตรวจสอบเงื่อนไขมุมก่อนจากนั้นจึงตรวจสอบเงื่อนไขระยะห่าง ซึ่งจะมีการส่งรหัสการเคลื่อนที่ไปยังบอร์ดไมโครคอนโทรลเลอร์ ดังแสดงในตารางที่ 3.3 และ 3.4 ตามลำดับ

หน้าตาของหุ่นยนต์เมื่ออยู่ในสถานะติดตามแสดงดังรูปที่ 3.15



รูปที่ 3.15 หน้าตาของหุ่นยนต์เมื่ออยู่ในสถานะติดตาม

ตารางที่ 3.3 การเคลื่อนที่ของหุ่นยนต์เพื่อรักษามุมระหว่างตัวหุ่นยนต์กับผู้สั่งงาน

ตำแหน่งของผู้สั่งงาน	การเคลื่อนที่	ASCII ที่ส่ง
เยื้องไปทางซ้ายของหุ่นยนต์มากกว่า 15 องศา	เคลื่อนที่ไปทางซ้ายด้วยความเร็วต่ำ	“A”
เยื้องไปทางขวาของหุ่นยนต์มากกว่า 15 องศา	เคลื่อนที่ไปทางขวาด้วยความเร็วต่ำ	“D”

ตารางที่ 3.4 การเคลื่อนที่ของหุ่นยนต์เพื่อรักษาระยะห่างระหว่างตัวหุ่นยนต์กับผู้สั่งงาน

ระยะห่างระหว่างผู้สั่งงานกับหุ่นยนต์	การเคลื่อนที่	ASCII ที่ส่ง
น้อยกว่า 1.5 เมตร	ถอยหลังด้วยความเร็วต่ำ	“X”
มากกว่าหรือเท่ากับ 1.5 เมตร แต่น้อยกว่า 2 เมตร	ไม่เคลื่อนที่	“S”
มากกว่าหรือเท่ากับ 2 เมตร แต่น้อยกว่า 3 เมตร	เดินหน้าด้วยความเร็วต่ำ	“W”
มากกว่า 3 เมตร	เดินหน้าด้วยความเร็วสูง	“Y”

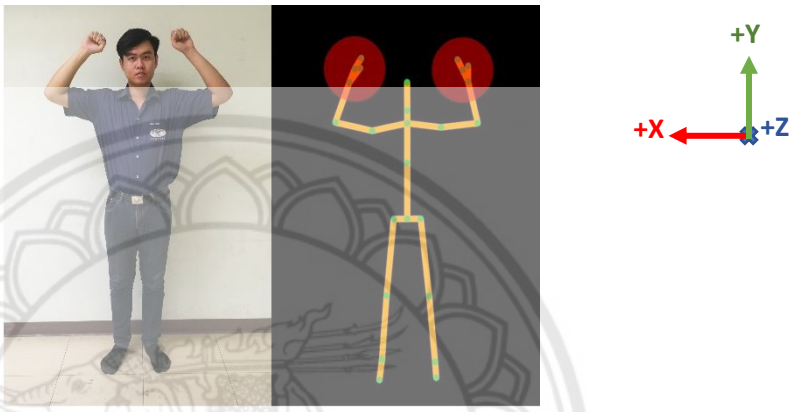
3.6 การศึกษาและพัฒนาโปรแกรมการตัดสินท่าทาง

กำหนดให้ในสถานะรับคำสั่งหุ่นยนต์สามารถรับการสั่งงานด้วยท่าทางได้ทั้งหมด 10 คำสั่ง แบ่งเป็นคำสั่งเปลี่ยนสถานะการทำงาน 3 คำสั่ง และ คำสั่งควบคุมการเคลื่อนที่ 7 คำสั่ง ดังต่อไปนี้

หมายเหตุ : สัญลักษณ์วงกลมแสดงถึงลักษณะของมือ โดยที่ สีแดงหมายถึงกำมือ สีเขียวหมายถึงแบมือ สีน้ำเงินหมายถึงชูสองนิ้ว หากไม่มีสัญลักษณ์วงกลมแสดงว่าทำมือเช่นไรก็ได้

3.6.1 สั่งงานให้หุ่นยนต์เข้าสู่สถานะรับคำสั่ง

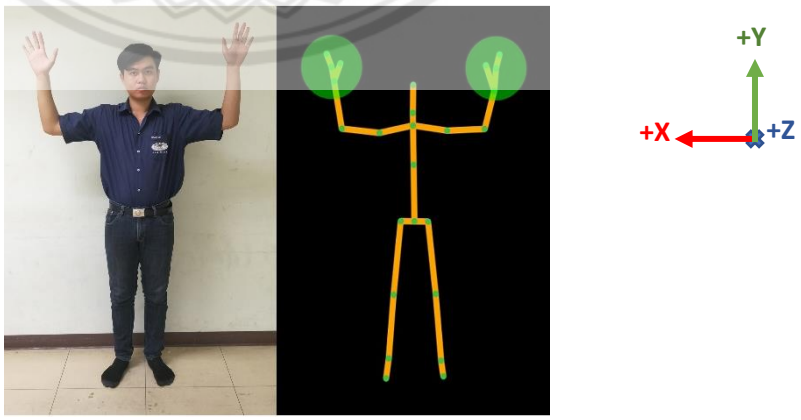
กำหนดให้กำมือทั้งสองข้างชูขึ้นเหนือไหล่หุบเข้าหาศีรษะ ดังรูปที่ 3.16 หุ่นยนต์จะเข้าสู่สถานะรับคำสั่ง



รูปที่ 3.16 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์เข้าสู่สถานะรับคำสั่ง

3.6.2 สั่งงานให้หุ่นยนต์เข้าสู่สถานะติดตาม

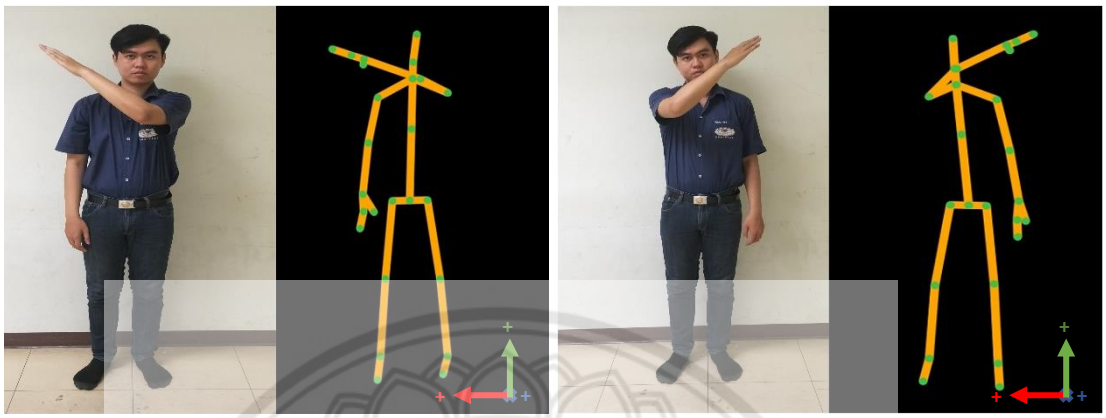
กำหนดให้แบมือทั้งสองข้างชูขึ้นเหนือไหล่กางแขนออก ดังรูปที่ 3.17 หุ่นยนต์จะเข้าสู่สถานะติดตาม



รูปที่ 3.17 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์เข้าสู่สถานะติดตาม

3.6.3 สั่งงานให้หุ่นยนต์เข้าสู่สถานะรอ

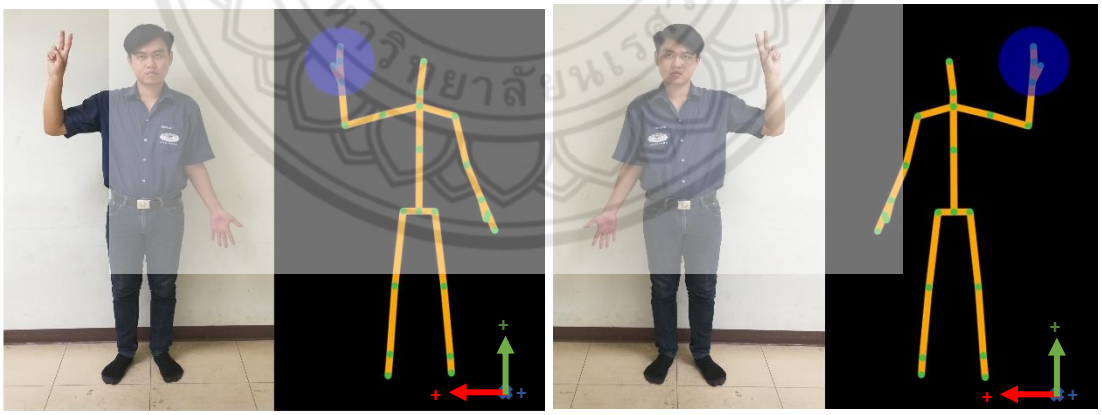
กำหนดให้ยกมือขวาขึ้นเหนือไหล่ซ้าย หรือ ยกมือซ้ายยกขึ้นเหนือไหล่ขวา ดังรูปที่ 3.18 หุ่นยนต์จะกลับเข้าสู่สถานะรอ



รูปที่ 3.18 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์กลับเข้าสู่สถานะรอ

3.6.4 สั่งงานให้หุ่นยนต์เดินหน้า

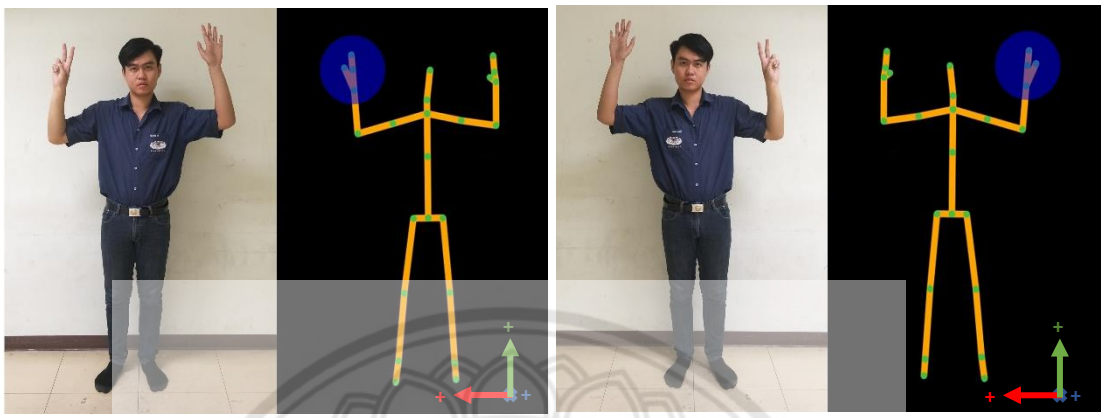
กำหนดให้ใช้มือข้างใดข้างหนึ่งชูสองนิ้วเหนือไหล่ส่วนมืออีกข้างวางต่ำกว่าสะโพก ดังรูปที่ 3.19 หุ่นยนต์จะเคลื่อนที่ไปข้างหน้า



รูปที่ 3.19 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์เดินหน้า

3.6.5 สั่งงานให้หุ่นยนต์ถอยหลัง

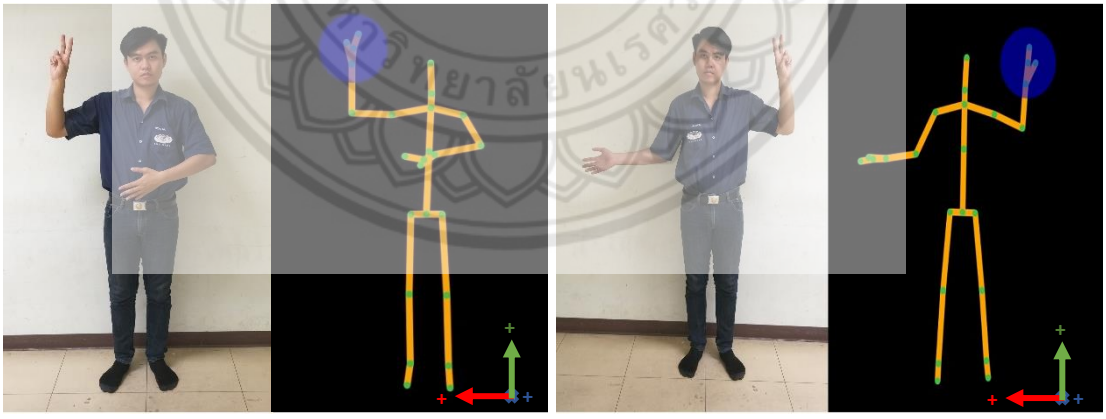
กำหนดให้ชูมือทั้งสองข้างเหนือไหล่ มือข้างใดข้างหนึ่งชูสองนิ้ว ดังรูปที่ 3.20 หุ่นยนต์จะเคลื่อนที่ถอยหลัง



รูปที่ 3.20 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์ถอยหลัง

3.6.6 สั่งงานให้หุ่นยนต์ไปทางซ้าย

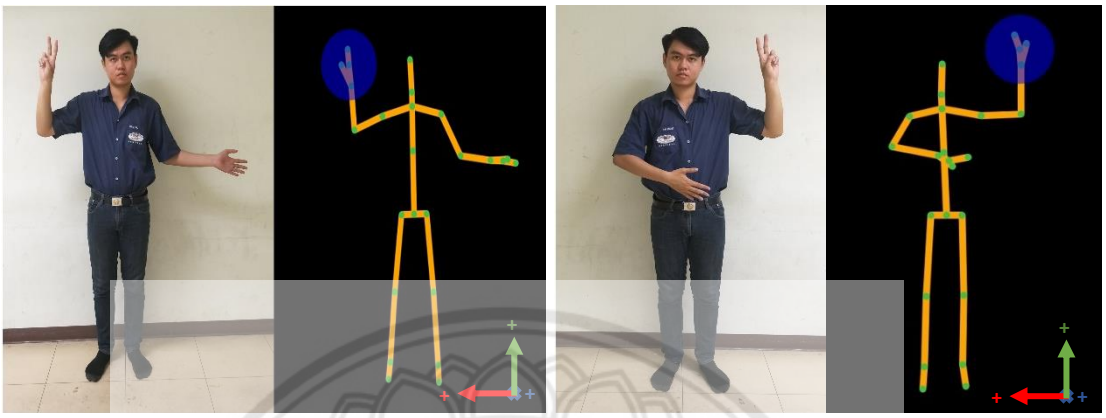
กำหนดให้ใช้มือข้างใดข้างหนึ่งชูสองนิ้วเหนือไหล่ส่วนมืออีกข้างอยู่ต่ำกว่าไหล่แต่สูงกว่าสะโพก วางมือไปในทิศที่เป็นด้านซ้ายของหุ่นยนต์ ดังรูปที่ 3.21 หุ่นยนต์จะเคลื่อนที่ไปทางซ้าย



รูปที่ 3.21 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์ไปทางซ้าย

3.6.7 สั่งงานให้หุ่นยนต์ไปทางขวา

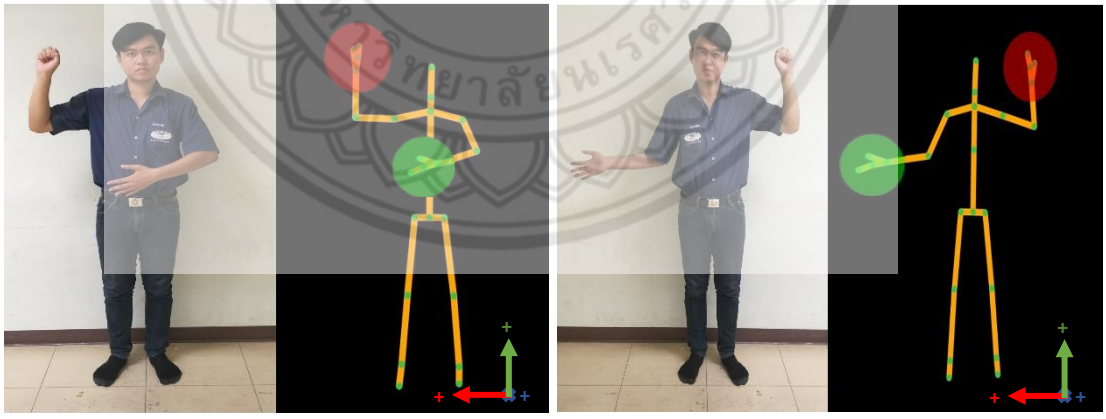
กำหนดให้ใช้มือข้างใดข้างหนึ่งชูสองนิ้วเหนือไหล่ส่วนมืออีกข้างอยู่ต่ำกว่าไหล่แต่สูงกว่าสะโพก วางมือไปในทิศที่เป็นด้านขวาของหุ่นยนต์ ดังรูปที่ 3.22 หุ่นเคลื่อนที่ไปทางด้านขวา



รูปที่ 3.22 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์ไปทางขวา

3.6.8 สั่งงานให้หุ่นยนต์หมุนซ้าย

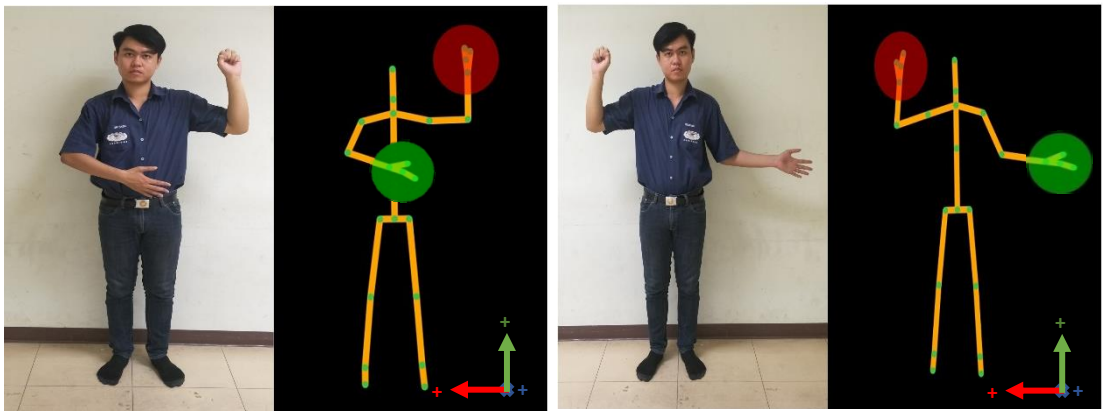
กำหนดให้กำมือข้างใดข้างหนึ่งชูขึ้นเหนือไหล่ส่วนมืออีกข้างอยู่ต่ำกว่าไหล่แต่สูงกว่าสะโพก วางมือไปในทิศที่เป็นด้านซ้ายของหุ่นยนต์ ดังรูปที่ 3.23 หุ่นยนต์จะหมุนตัวไปทางซ้าย



รูปที่ 3.23 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์หมุนซ้าย

3.6.9 สั่งงานให้หุ่นยนต์หมุนขวา

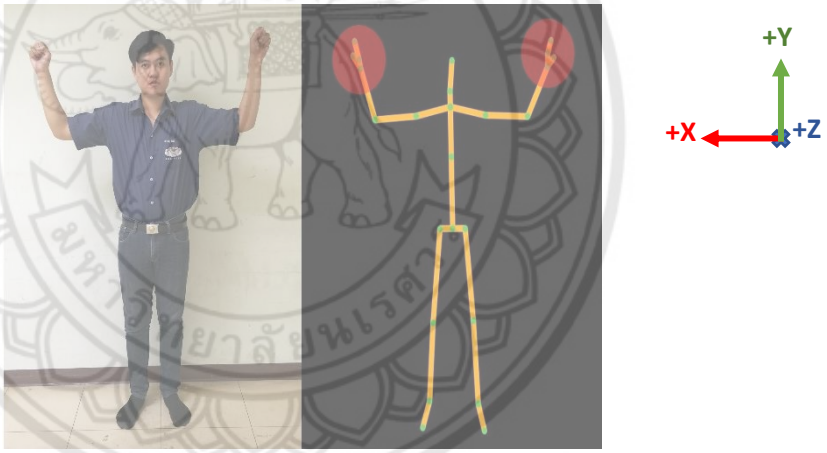
กำหนดให้กำมือข้างใดข้างหนึ่งชูขึ้นเหนือไหล่ส่วนมืออีกข้างอยู่ต่ำกว่าไหล่แต่สูงกว่าสะโพก วางมือไปในทิศที่เป็นด้านขวาของหุ่นยนต์ ดังรูปที่ 3.24 หุ่นยนต์จะหมุนตัวไปทางขวา



รูปที่ 3.24 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์หมุนขวา

3.6.10 สั่งงานให้หุ่นยนต์หยุดอยู่กับที่

กำหนดให้กำมือทั้งสองข้างชูขึ้นเหนือไหล่กางแขนออก ดังรูปที่ 3.25 หุ่นยนต์จะหยุดเคลื่อนที่



รูปที่ 3.25 ท่าทางที่ใช้สั่งงานให้หุ่นยนต์หยุดเคลื่อนที่

คำสั่งสำหรับควบคุมการเคลื่อนที่เป็นการสั่งให้เคลื่อนที่ด้วยความเร็วสูงทั้งหมดยกเว้นการสั่งให้หมุนจะหมุนด้วยความเร็วต่ำ เว้นแต่จะอยู่ในสถานะติดตามดังที่แสดงเงื่อนไขไว้ข้างต้น

จำนวนจุดต่างๆ ของร่างกายที่สามารถระบุพิกัดได้จากการใช้เซนเซอร์ Kinect คือ 25 จุด ดังที่ได้กล่าวไว้ในบทที่ 2 สำหรับระบบที่ได้จัดทำขึ้นนี้มีการนำพิกัดมาเปรียบเทียบเงื่อนไขเพียง 8 จุด คือ ไหล่ซ้าย ไหล่ขวา ข้อศอกซ้าย ข้อศอกขวา มือซ้าย มือขวา กลางลำตัว และปลายกระดูกสันหลัง กำหนดตัวแปรค่าพิกัดของจุดต่างๆ ดังตารางที่ 3.5

ตารางที่ 3.5 ตัวแปรค่าพิกัดของจุดที่นำมาใช้เพื่อตรวจสอบเงื่อนไขท่าทาง

จุดที่พิจารณา	พิกัด	ตัวแปร	จุดที่พิจารณา	พิกัด	ตัวแปร
ไหล่ซ้าย	X	LS_X	ไหล่ขวา	X	RS_X
	Y	LS_Y		Y	RS_Y
	Z	LS_Z		Z	RS_Z
ศอกซ้าย	X	LE_X	ศอกขวา	X	RE_X
	Y	LE_Y		Y	RE_Y
	Z	LE_Z		Z	RE_Z
มือซ้าย	X	LH_X	มือขวา	X	RH_X
	Y	LH_Y		Y	RH_Y
	Z	LH_Z		Z	RH_Z
กลางลำตัว	X	MS_X	ปลายกระดูกสันหลัง	X	BS_X
	Y	MS_Y		Y	BS_Y
	Z	MS_Z		Z	BS_Z

นอกจากนี้ยังมีการกำหนดตัวแปร LH_State เป็นสถานะของมือซ้าย และ RH_State เป็นสถานะของมือขวา ซึ่งมีได้ทั้งหมด 3 สถานะ คือ O (Open) หมายถึง แบมือ C (Closed) หมายถึง กำมือ และ L (Lasso) หมายถึง ชูสองนิ้ว

การตรวจสอบท่าทางใช้เงื่อนไขดังแสดงในตารางที่ 3.6 และ 3.7 รหัสสถานะจะใช้เพียงเพื่อกำหนดรูปแบบการทำงานของหุ่นยนต์ที่การควบคุมส่วนบนเท่านั้น แต่รหัสการเคลื่อนที่จะถูกส่งต่อไปยังการควบคุมส่วนล่าง

ตารางที่ 3.6 เงื่อนไขท่าทางคำสั่งสำหรับเปลี่ยนสถานะการทำงาน

คำสั่ง	**เงื่อนไข**	**รหัสสถานะ**
ให้รับคำสั่ง	(LH_State == 'C') && (RH_State == 'C') && (LH_X > LE_X) && (LH_Y > LS_Y) && (RH_X < RE_X) && (RH_Y > LS_Y)	'C'
ให้ติดตาม	(LH_State == 'O') && (RH_State == 'O') && (LH_X < LE_X) && (LH_Y > LS_Y) && (RH_X > RE_X) && (RH_Y > LS_Y)	'F'
ให้รอ	**(** (LH_X > MS_X) && (LH_Y > LS_Y) **)** || **(** (RH_X < MS_X) && (RH_Y > LS_Y) **)**	'S'

ตารางที่ 3.7 เงื่อนไขท่าทางคำสั่งสำหรับควบคุมการเคลื่อนที่

คำสั่ง	เงื่อนไข	รหัสการเคลื่อนที่
ให้เดินหน้า	((LH_State == 'L') && (LH_Y > LS_Y) && (RH_Y < BS_Y)) || ((RH_State == 'L') && (RH_Y > RS_Y) && (LH_Y < BS_Y))	“Y”
ให้ถอยหลัง	((LH_State == 'L') && (LH_Y > LS_Y) && (RH_Y > RS_Y)) || ((LH_State == 'L') && (LH_Y > LS_Y) && (LH_Y > LS_Y))	“N”
ให้ไปทางซ้าย	((LH_State == 'L') && (LH_Y > LS_Y) && (RH_X > RE_X)) || ((RH_State == 'L') && (RH_Y > RS_Y) && (LH_X > LE_X))	“G”
ให้ไปทางขวา	((LH_State == 'L') && (LH_Y > LS_Y) && (RH_X < RE_X)) || ((RH_State == 'L') && (RH_Y > RS_Y) && (LH_X < LE_X))	“J”
ให้หมุนซ้าย	((LH_State == 'C') && (RH_State == 'O') && (LH_Y > LS_Y) && (RH_Y < RE_Y) && (RH_X > RE_X)) || ((RH_State == 'C') && (LH_State == 'O') && (RH_Y > RS_Y) && (LH_Y < LE_Y) && (LH_X > LE_X))	“Q”
ให้หมุนขวา	((LH_State == 'C') && (RH_State == 'O') && (LH_Y > LS_Y) && (RH_Y < RE_Y) && (RH_X < RE_X)) || ((RH_State == 'C') && (LH_State == 'O') && (LH_Y > LS_Y) && (RH_Y < RE_Y) && (LH_X < LE_X))	“E”
ให้หยุด	(LH_State == 'C') && (RH_State == 'C') && (LH_X < LE_X) && (LH_Y > LS_Y) && (RH_X > RE_X) && (RH_Y > LS_Y)	“S”

3.7 การศึกษาและพัฒนาโปรแกรมการเคลื่อนที่ของหุ่นยนต์

สำหรับการตอบสนองของบอร์ดไมโครคอนโทรลเลอร์จะมีการตรวจสอบรหัส ASCII ที่ได้รับจากคอมพิวเตอร์บนตัวหุ่นยนต์ก่อนเสมอและตอบสนองดังตารางที่ 3.8 เพื่อเลือกว่าจะนำข้อมูลจากพอร์ตอนุกรมพอร์ตใดมากำหนดการเคลื่อนที่ โดยที่การตอบสนองต่อรหัสการเคลื่อนที่ไม่ว่าจะเป็นการรับจากคอมพิวเตอร์บนตัวหุ่นยนต์ หรือจาก Joystick สามารถแสดงได้ดังตารางที่ 3.9

ตารางที่ 3.8 การตอบสนองของบอร์ด Arduino Due ต่อรหัสที่ได้รับจากคอมพิวเตอร์

รหัสที่ได้รับ	การตอบสนอง
“K”	เลือกรับคำสั่งงานจากคอมพิวเตอร์บนตัวหุ่นยนต์จนกว่าจะได้รับรหัส “I”
“I”	เลือกรับคำสั่งงานจาก Joystick จนกว่าจะได้รับรหัส “K”

ตารางที่ 3.9 การตอบสนองของบอร์ด Arduino Due ต่อรหัสการเคลื่อนที่

รหัสที่ได้รับ	การตอบสนอง	
	การกำหนดความเร็วในการเคลื่อนที่	รูปแบบการเคลื่อนที่
“W”	ความเร็วต่ำ	เดินหน้า
“X”		ถอยหลัง
“A”		ไปทางซ้าย
“D”		ไปทางขวา
“Q”		หมุนซ้าย
“E”		หมุนขวา
“S”	-	หยุดอยู่กับที่
“Y”	ความเร็วสูง	เดินหน้า
“N”		ถอยหลัง
“G”		ไปทางซ้าย
“J”		ไปทางขวา

3.7.1 หุ่นยนต์สามารถเคลื่อนที่ด้วยความเร็วคงที่ได้

การความคุมความเร็วในการเคลื่อนที่ของหุ่นยนต์ทำได้โดยวิธีการแปรค่าความกว้างพัลส์ของสัญญาณที่ป้อนให้กับบอร์ดไดร์เวอร์มอเตอร์ กำหนดให้แปรผันตามสัญญาณที่ป้อนกลับจากเอนโค้ดเดอร์ซึ่งสามารถนำมาคำนวณหาความเร็วจริงในการหมุนของมอเตอร์ ณ ช่วงเวลาใดๆ หากความเร็วจริงต่ำกว่าความเร็วที่ต้องการให้ค่าความกว้างพัลส์ของสัญญาณที่ป้อนให้กับบอร์ดไดร์เวอร์มอเตอร์เพิ่มขึ้น หากความเร็วจริงสูงกว่าความเร็วที่ต้องการให้กำหนดค่าความกว้างพัลส์ของสัญญาณที่ป้อนให้กับบอร์ดไดร์เวอร์มอเตอร์ลดลง ทั้งนี้การรักษาความเร็วของมอเตอร์แต่ละตัวให้คงที่นั้นก็เพื่อเป็นการกำหนดเวกเตอร์ความเร็วให้สอดคล้องกับการเคลื่อนที่ของหุ่นยนต์ซึ่งคำนวณได้จากการรวมเวกเตอร์ความเร็วของการหมุนของมอเตอร์ทั้งสี่ตัว กำหนดให้ ที่ความเร็วต่ำ มอเตอร์หมุนด้วยความเร็ว 30 รอบต่อนาที และที่ความเร็วสูง มอเตอร์หมุนด้วยความเร็ว 60 รอบต่อนาที

3.7.2 หุ่นยนต์สามารถเคลื่อนที่ไปทุกทิศทางโดยที่ด้านหน้าของหุ่นยนต์ไม่เปลี่ยนทิศ

การเคลื่อนที่ของหุ่นยนต์เกิดจากการรวมเวกเตอร์ความเร็วของล้อทั้งสี่ซึ่งเป็นล้อชนิด Omni-Directional ทิศทางการเคลื่อนที่ของหุ่นยนต์สอดคล้องกับทิศทางการหมุนของมอเตอร์ดังที่แสดงด้วยตารางที่ 3.10

ตารางที่ 3.10 ความสอดคล้องระหว่างการเคลื่อนที่ของหุ่นยนต์กับทิศทางการหมุนของมอเตอร์

รูปแบบการเคลื่อนที่	ทิศทางการหมุนของมอเตอร์ (เมื่อมองออกจากแกนหมุนไปยังล้อ)			
	มอเตอร์ A	มอเตอร์ B	มอเตอร์ C	มอเตอร์ D
เดินหน้า	CW	CCW	CW	CCW
ถอยหลัง	CCW	CW	CCW	CW
ไปทางซ้าย	CCW	CCW	CW	CW
ไปทางขวา	CW	CW	CCW	CCW
หมุนซ้าย	CCW	CCW	CCW	CCW
หมุนขวา	CW	CW	CW	CW

หมายเหตุ : เมื่อ CW หมายถึง ตามเข็มนาฬิกา และ CCW หมายถึง ทวนเข็มนาฬิกา

3.8 การศึกษาและพัฒนาโปรแกรมการตรวจจับสิ่งกีดขวาง

กำหนดให้หุ่นยนต์สามารถลดความเร็วในการเคลื่อนที่และหยุดเมื่อตรวจพบสิ่งกีดขวาง โดยใช้เซนเซอร์อัลทราโซนิกในการตรวจวัดระยะห่างระหว่างหุ่นยนต์กับสิ่งกีดขวางที่พบ

หุ่นยนต์จะถูกสั่งให้เคลื่อนที่ด้วยความเร็วสูงหรือเคลื่อนที่ด้วยความเร็วต่ำก็ได้ขึ้นอยู่กับคำสั่งท่าทางในสถานะรับคำสั่ง เงื่อนไขในสถานะติดตาม หรือคำสั่งจากการสั่งงานด้วย Joystick กำหนดให้ความเร็วที่ถูกสั่งงานตามเงื่อนไขข้างต้น คือ “ความเร็วที่ถูกกำหนด” หากตรวจพบว่าไม่มีสิ่งกีดขวางอยู่ในระยะ ต่ำกว่า 0.8 เมตร ในทิศทางที่กำลังเคลื่อนที่ไปหุ่นยนต์จะเคลื่อนที่ด้วยความเร็วที่ถูกกำหนด เงื่อนไขการตัดสินใจของหุ่นยนต์เมื่อตรวจพบสิ่งกีดขวาง แสดงดังตารางที่ 3.11

ตารางที่ 3.11 เงื่อนไขการตัดสินใจของหุ่นยนต์เมื่อตรวจพบสิ่งกีดขวาง

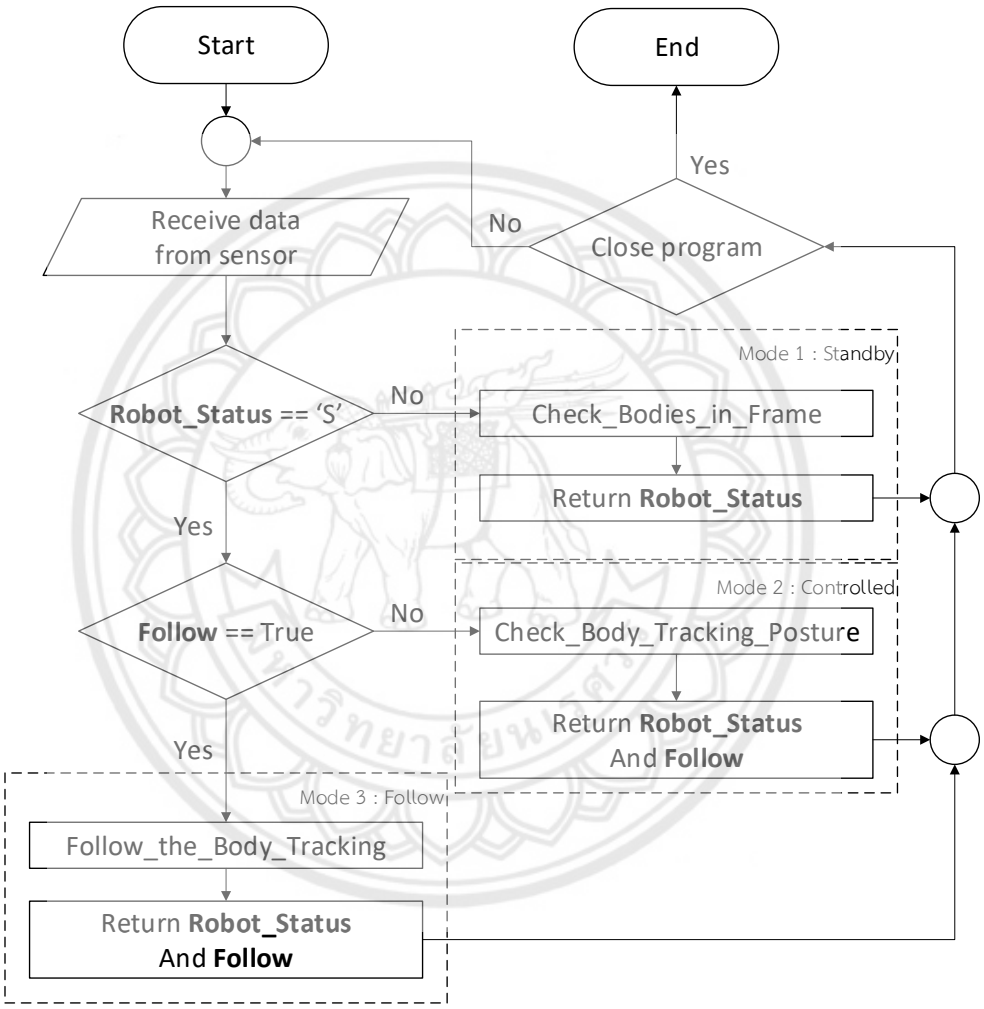
ระยะห่างระหว่างหุ่นยนต์กับสิ่งกีดขวางที่พบ	การตอบสนอง
มากกว่า 0.8 เมตร	เคลื่อนที่ต่อไปด้วยความเร็วที่ถูกกำหนด
มากกว่า 0.4 เมตร แต่น้อยกว่า 0.8 เมตร	เคลื่อนที่ด้วยความเร็วต่ำเสมอ
น้อยกว่า 0.4 เมตร	ไม่เคลื่อนที่

3.9 การทดสอบการทำงานของหุ่นยนต์

การทดสอบการทำงานของหุ่นยนต์จะแสดงโดยละเอียดในบทที่ 4

3.10 อัลกอริทึมการทำงานของหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง

3.10.1 อัลกอริทึมการทำงานของโปรแกรมสำหรับส่วนที่ใช้คอมพิวเตอร์ควบคุม

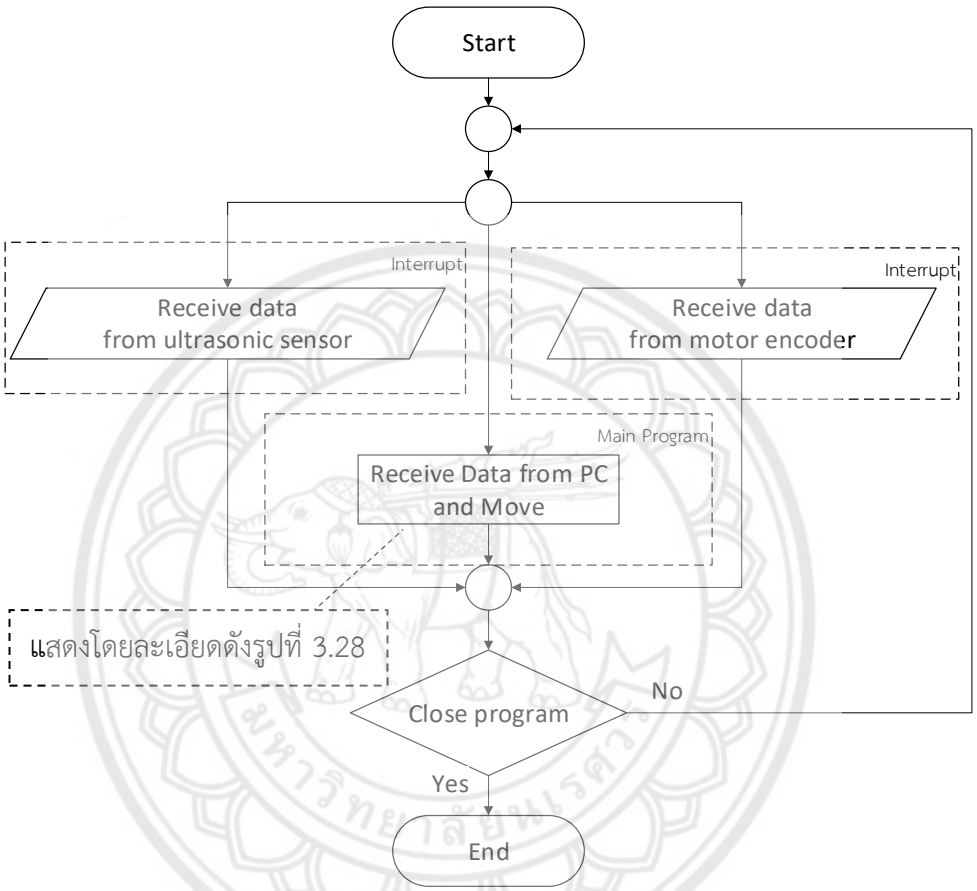


รูปที่ 3.26 การทำงานของโปรแกรมที่ใช้คอมพิวเตอร์ควบคุม

การทำงานของโปรแกรมในส่วนที่ใช้คอมพิวเตอร์ควบคุมจะมีการแบ่งสถานะการทำงานออกเป็น 3 สถานะ สถานะแรก คือ สถานะรอ เป็นการรอจนกว่าจะมีคนมาสั่งให้เริ่มรับคำสั่ง โดยต้องทำท่าทางตรงตามคำสั่งที่ให้เข้าสู่สถานะรับคำสั่งซึ่งก็คือ สถานะที่สอง โดยในสถานะรับคำสั่งหุ่นยนต์จะปฏิบัติตามคำสั่งจากท่าทางของผู้ใช้งาน สถานะที่สาม คือ สถานะติดตาม เมื่อผู้ใช้งานสั่งให้หุ่นยนต์เข้าสู่สถานะติดตาม หุ่นยนต์จะเคลื่อนที่ตามผู้ใช้งานโดยจะรักษาระยะห่างให้คงที่ มี

เงื่อนไขดังที่กล่าวไปแล้วในหัวข้อที่ 3.5 ภายในการทำงานแต่ละสถานะโปรแกรมในส่วนการควบคุมระดับสูงจะตัดสินรูปแบบการเคลื่อนที่แล้วส่งคำสั่งไปยังไมโครคอนโทรลเลอร์ผ่านทางพอร์ตอนุกรม

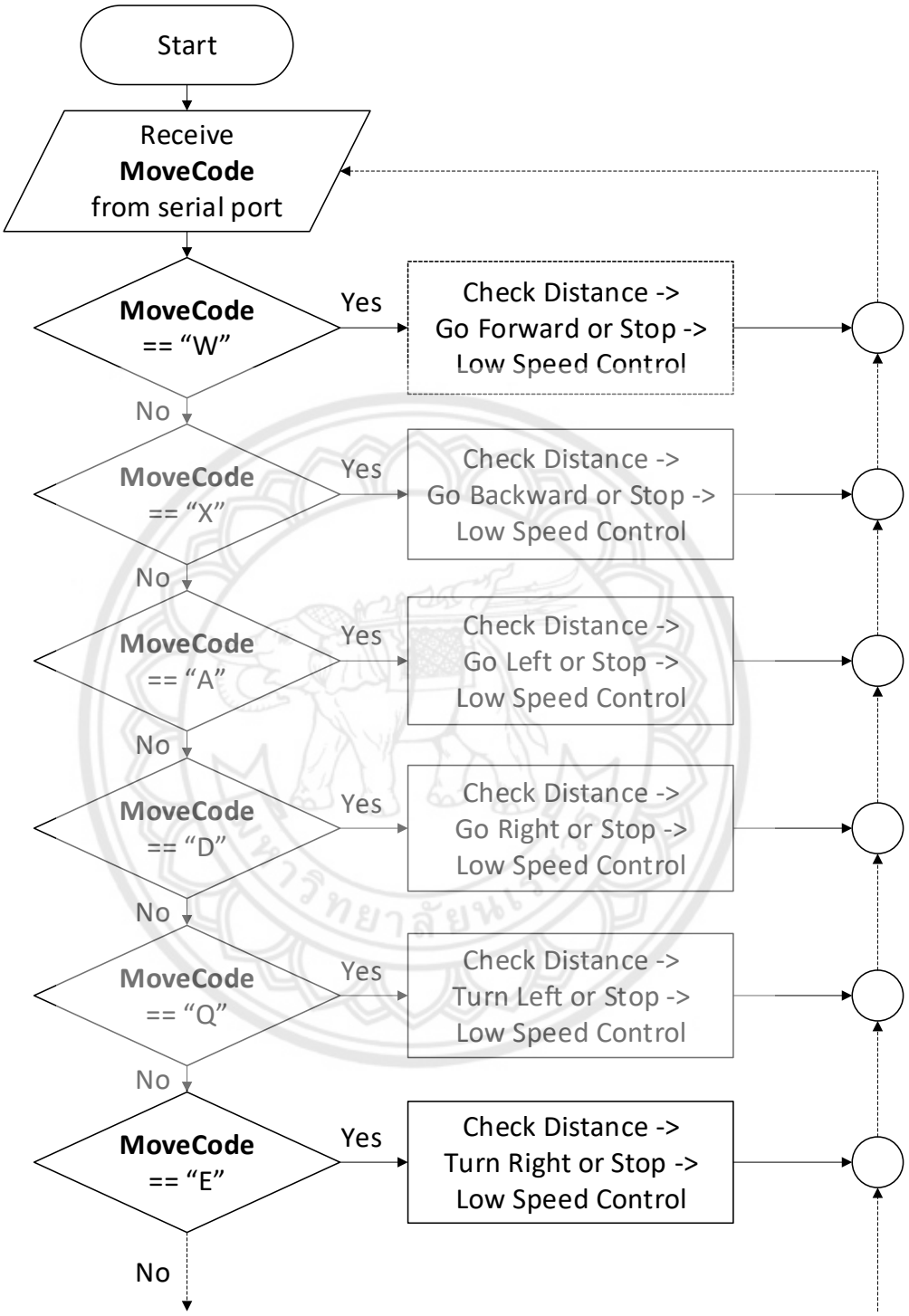
3.10.2 อัลกอริทึมการทำงานของโปรแกรมสำหรับส่วนไมโครคอนโทรลเลอร์



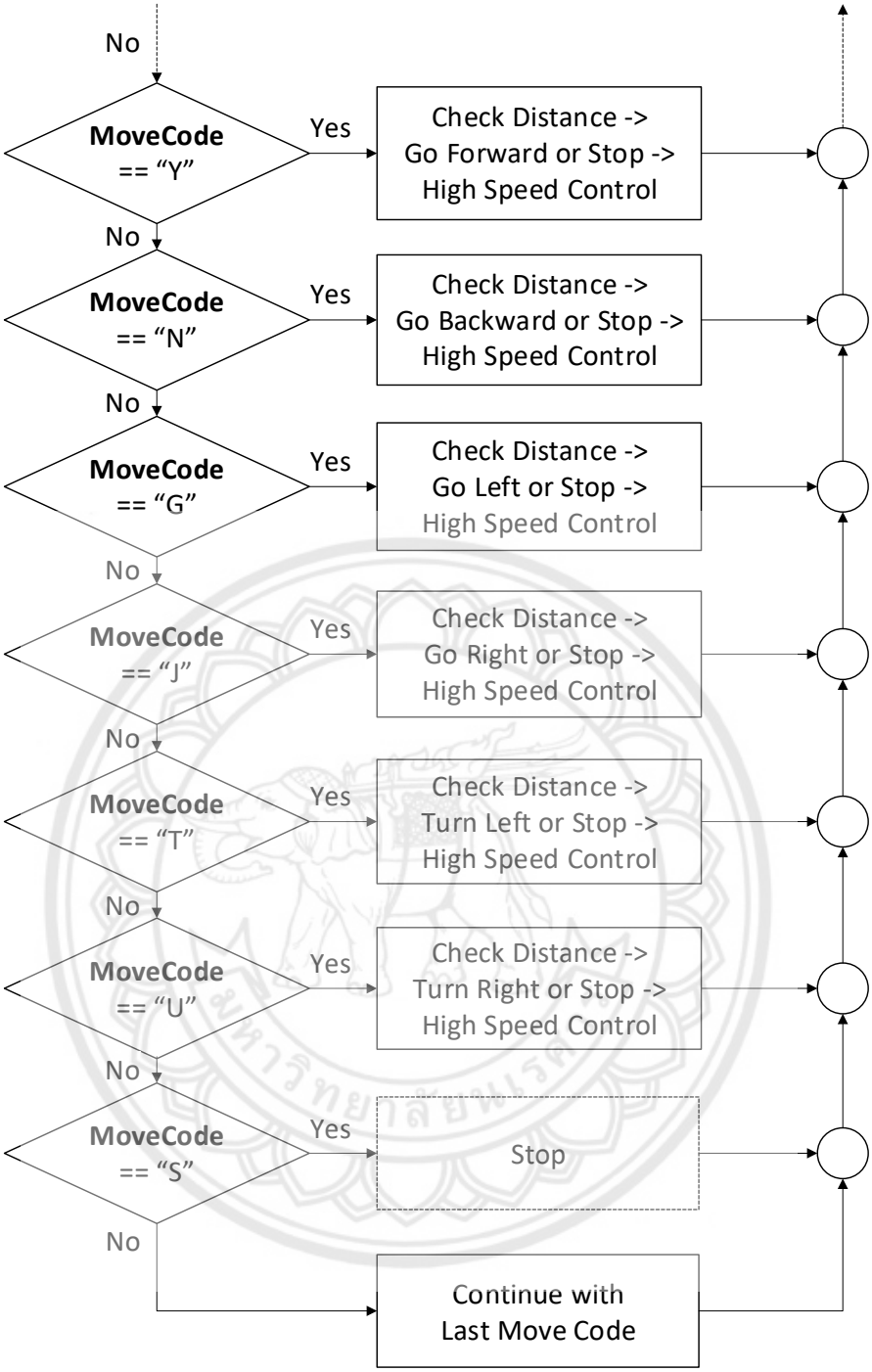
รูปที่ 3.27 การทำงานของโปรแกรมสำหรับส่วนไมโครคอนโทรลเลอร์

ในส่วนของไมโครคอนโทรลเลอร์จะมีการรับค่าทั้งจากเซนเซอร์อัลทราโซนิก เอ็นโค้ดเดอร์และ คอมพิวเตอร์ส่วนบุคคลขนาดเล็ก ซึ่งต้องมีการทำงานแยกจากกันโดยที่การรับค่าคำสั่งจากคอมพิวเตอร์ส่วนบุคคลขนาดเล็กจะอยู่ในส่วนของโปรแกรมหลัก ส่วนการรับค่าจากเซนเซอร์อัลทราโซนิกและเอ็นโค้ดเดอร์จำเป็นต้องอยู่ในส่วนที่เป็นการอินเตอร์รัพท์ซึ่งแยกอิสระจากกัน โดยค่าที่ได้จากเซนเซอร์อัลทราโซนิกและเอ็นโค้ดเดอร์จะถูกส่งต่อผ่านค่าของตัวแปรที่จะถูกนำไปใช้ในโปรแกรมหลัก

การทำงานของโปรแกรมหลักแสดงดังรูปที่ 3.28



รูปที่ 3.28 การทำงานของโปรแกรมหลักสำหรับบอร์ด Arduino Due



รูปที่ 3.28 (ต่อ) การทำงานของโปรแกรมหลักสำหรับบอร์ด Arduino Due

เมื่อบอร์ด Arduino Due ได้รับรหัสการเคลื่อนที่มาจากคอมพิวเตอร์ โปรแกรมจะทำฟังก์ชันการเคลื่อนที่ตามรหัสที่ได้รับ โดยในทุกฟังก์ชันการเคลื่อนที่จะตรวจสอบเงื่อนไขระยะที่ได้จากเซนเซอร์อัลทราโซนิกก่อนเพื่อพิจารณาว่าจะเคลื่อนที่หรือหยุด แล้วส่งสัญญาณไปที่พินควบคุมทิศการหมุนของไดร์เวอร์มอเตอร์ จากนั้นทำฟังก์ชันสำหรับควบคุมความเร็ว โดยเพิ่มค่าหรือลดค่าความกว้างพัลส์ของสัญญาณเปิดปิดที่ส่งไปยังไดร์เวอร์มอเตอร์ตามเงื่อนไขดังที่ได้กล่าวไว้ในหัวข้อที่ 3.7.1

ภายในฟังก์ชันต่างๆ ของการเคลื่อนที่จะต้องมีฟังก์ชันที่ใช้ตรวจสอบเงื่อนไขระยะห่างของสิ่งกีดขวางในทิศทางที่หุ่นยนต์กำลังจะเคลื่อนที่ไปอยู่ภายในฟังก์ชันการเคลื่อนที่ หากตรวจพบว่ามีสิ่งกีดขวางอยู่ในระยะใกล้เกินกว่าที่กำหนดไว้ได้แก่ ระยะ 0.8 เมตร จะเคลื่อนที่ด้วยความเร็วต่ำ และ ระยะ 0.4 เมตร จะหยุดเคลื่อนที่ กำหนดตัวแปรของระยะห่างในทิศต่างๆ ดังนี้

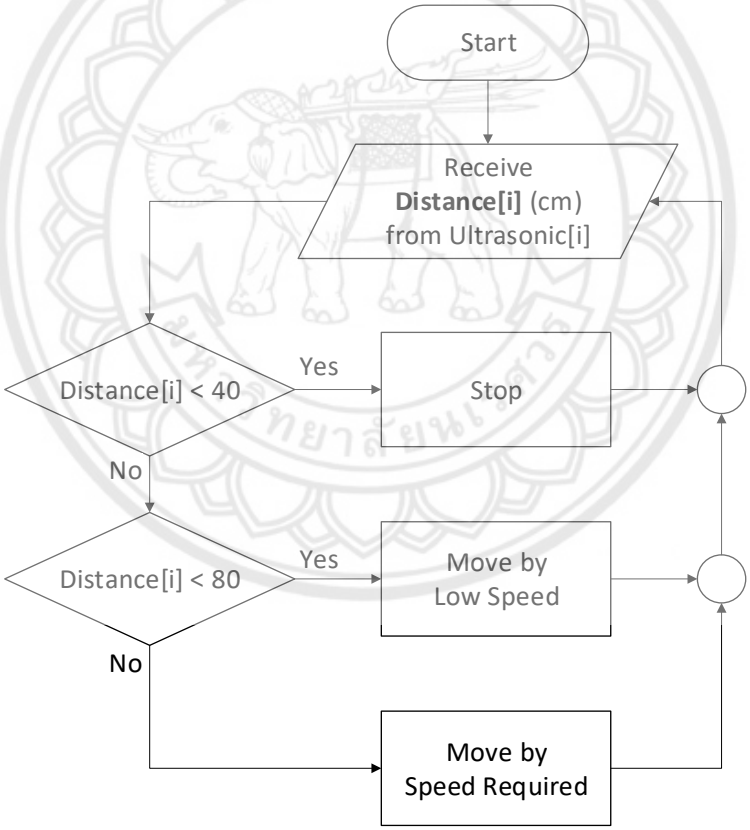
Distance[1] คือ ระยะห่างซึ่งตรวจวัดได้จากเซนเซอร์อัลทราโซนิกที่ติดตั้งไว้ด้านหน้า (Sensor 1)

Distance[2] คือ ระยะห่างซึ่งตรวจวัดได้จากเซนเซอร์อัลทราโซนิกที่ติดตั้งไว้ด้านหลัง (Sensor 2)

Distance[3] คือ ระยะห่างซึ่งตรวจวัดได้จากเซนเซอร์อัลทราโซนิกที่ติดตั้งไว้ด้านซ้าย (Sensor 3)

Distance[4] คือ ระยะห่างซึ่งตรวจวัดได้จากเซนเซอร์อัลทราโซนิกที่ติดตั้งไว้ด้านขวา (Sensor 4)

การทำงานของฟังก์ชันตรวจสอบระยะห่างของสิ่งกีดขวางในทิศทางที่กำลังจะเคลื่อนที่ไปแสดงได้ดังรูปที่ 3.29



รูปที่ 3.29 ฟังก์ชันที่ใช้ตรวจสอบระยะห่างของสิ่งกีดขวาง

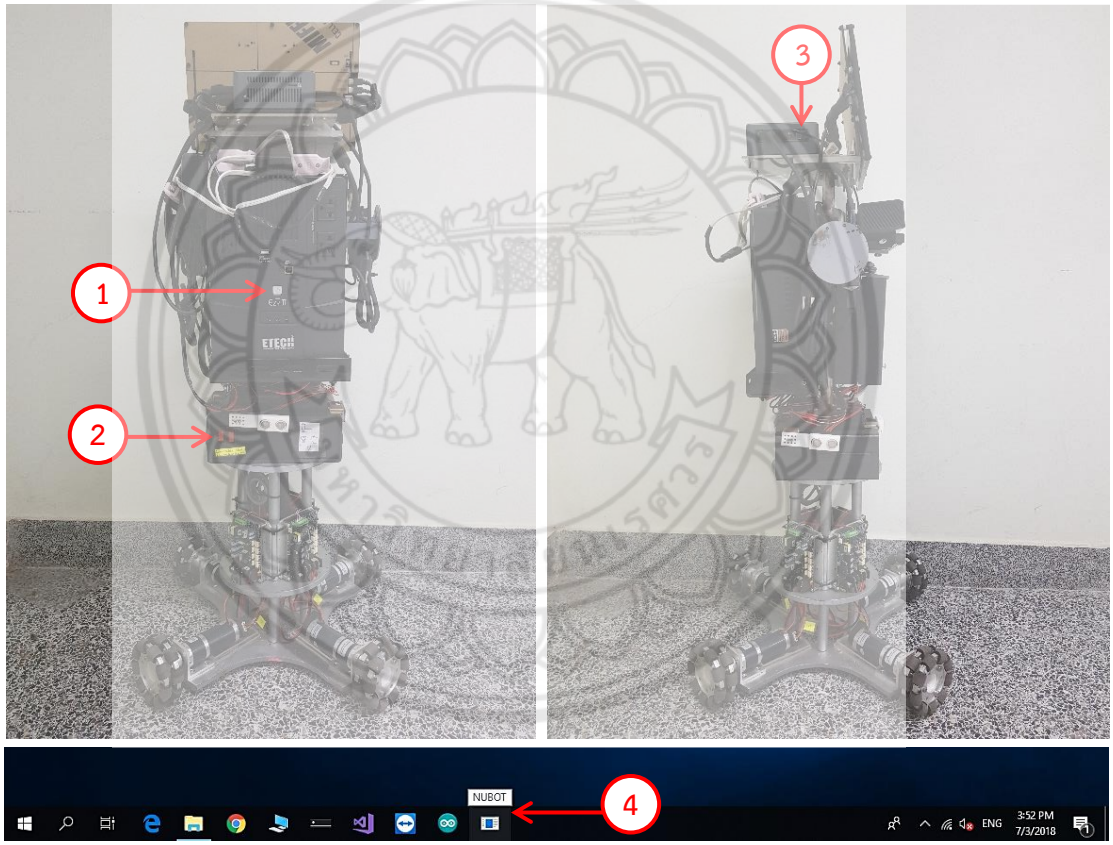
จากการออกแบบและพัฒนาระบบทั้งหมดที่กล่าวมาข้างต้น การสั่งงานด้วยท่าทาง การตรวจจับระยะของเซนเซอร์อัลทราโซนิก และการเคลื่อนที่ของหุ่นยนต์ จะถูกนำไปทดสอบในบทถัดไป

3.11 การใช้งานหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง

3.11.1 การเริ่มต้นใช้งานหุ่นยนต์

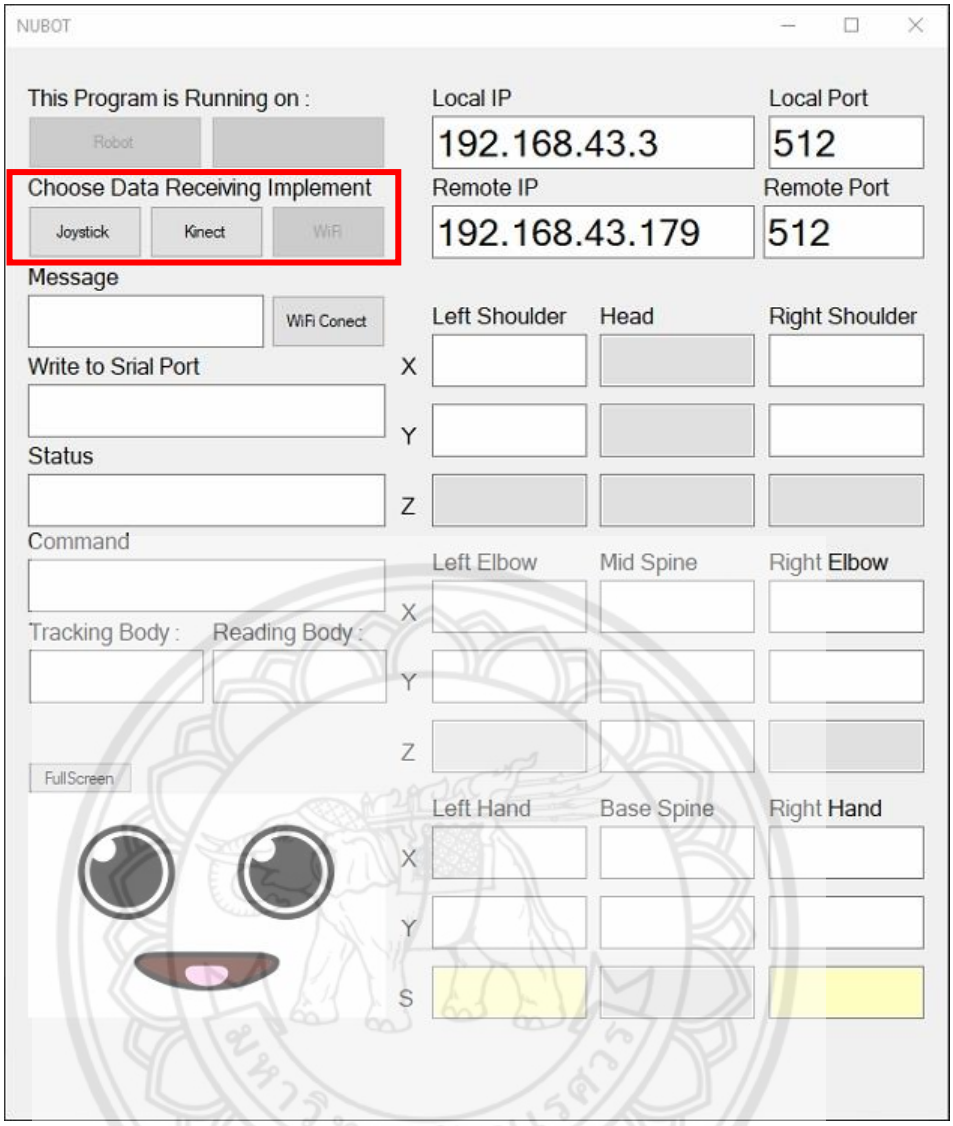
ขั้นตอนในการเริ่มต้นใช้งานหุ่นยนต์ ดูรูปที่ 3.30 มีรายละเอียดดังนี้ คือ

1) กดสวิตช์เปิดที่ UPS เพื่อเปิดใช้งานแหล่งจ่ายไฟฟ้า
2) สับสวิตช์ทางด้านซ้ายขึ้นเพื่อเชื่อมต่อ UPS และ DC Battery 12 V เข้าด้วยกัน และสับสวิตช์ทางด้านขวาขึ้นเพื่อเปิดใช้งานวงจรขับมอเตอร์
3) กดสวิตช์เปิดเครื่องคอมพิวเตอร์
4) เปิดโปรแกรม NUBOT.exe



รูปที่ 3.30 ขั้นตอนในการเริ่มต้นใช้งานหุ่นยนต์

หลังจากเปิดโปรแกรมขึ้นมาแล้วจะพบว่าหน้าจอแสดงผลของโปรแกรมเป็นดังรูปที่ 3.31 ผู้ใช้สามารถเลือกวิธีการสั่งงานหุ่นยนต์ได้ที่เมนู “Choose Data Receiving Implement” ดังที่แสดงในกรอบสี่เหลี่ยมสีแดง ซึ่งหากยังไม่ได้กำหนดช่องทางการรับส่งข้อมูลระหว่างคอมพิวเตอร์ผ่าน Wi-Fi ก็จะยังไม่สามารถเลือกใช้วิธีการสั่งงานระยะไกลด้วยท่าทางผ่าน Wi-Fi ได้



รูปที่ 3.31 หน้าจอแสดงผลของโปรแกรม

3.11.2 หน้าจอแสดงผลของโปรแกรม

สำหรับการสั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์ และสั่งงานระยะไกลด้วยท่าทาง ส่วนต่างๆ ของหน้าจอแสดงผลมีนัยสำคัญดังนี้ (ดูรูปที่ 3.32 ประกอบ)

1) เครื่องคอมพิวเตอร์ที่โปรแกรมกำลังทำงานอยู่บนหุ่นยนต์หรือภายนอก
2) ทางเลือกในการสั่งงานหุ่นยนต์
3) ข้อมูลที่ส่งหรือรับผ่าน Wi-Fi
4) ข้อมูลที่คอมพิวเตอร์บนตัวหุ่นยนต์ส่งไปยังบอร์ด Arduino Due
5) สถานะของหุ่นยนต์ภายใต้การสั่งงานด้วยท่าทาง
6) คำสั่งควบคุมการเคลื่อนที่ที่ถูกตีความจากท่าทาง

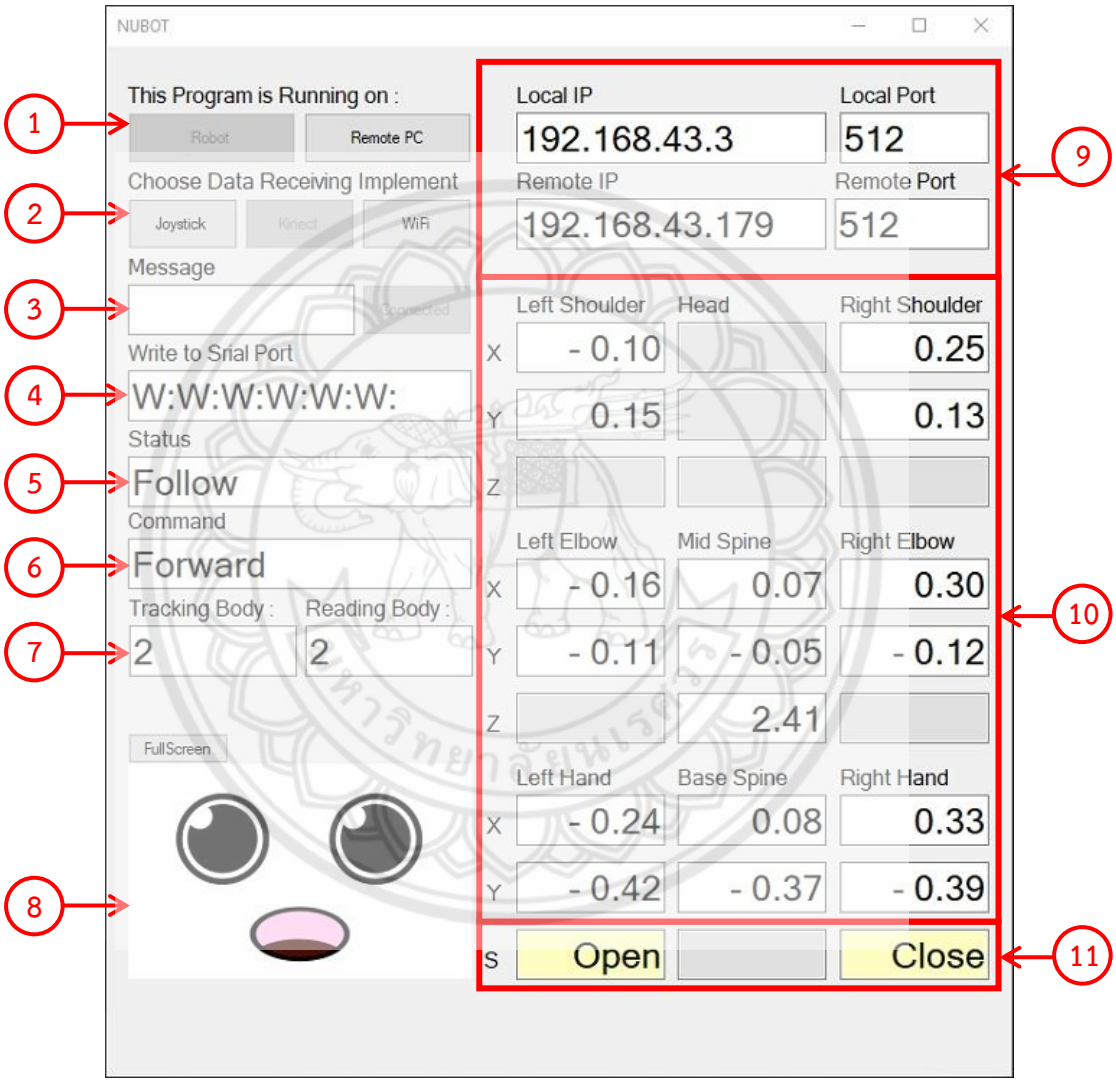
7) หมายเลขบุคคลที่หุ่นยนต์กำลังวิเคราะห์ท่าทางการสั่งงาน

8) สีหน้าของหุ่นยนต์ซึ่งจะเปลี่ยนไปตามสถานการณ์ทำงาน

9) หมายเลข IP Address ของเครื่องคอมพิวเตอร์ และพอร์ตสำหรับเชื่อมต่อ Wi-Fi

10) พิกัดฉาก ของจุด/ข้อต่อ ของผู้สั่งงาน ที่โปรแกรมใช้เพื่อตัดสินท่าทาง

11) สถานะของมือซ้ายและมือขวาของผู้สั่งงาน



รูปที่ 3.32 ส่วนต่างๆ ของหน้าจอแสดงผล

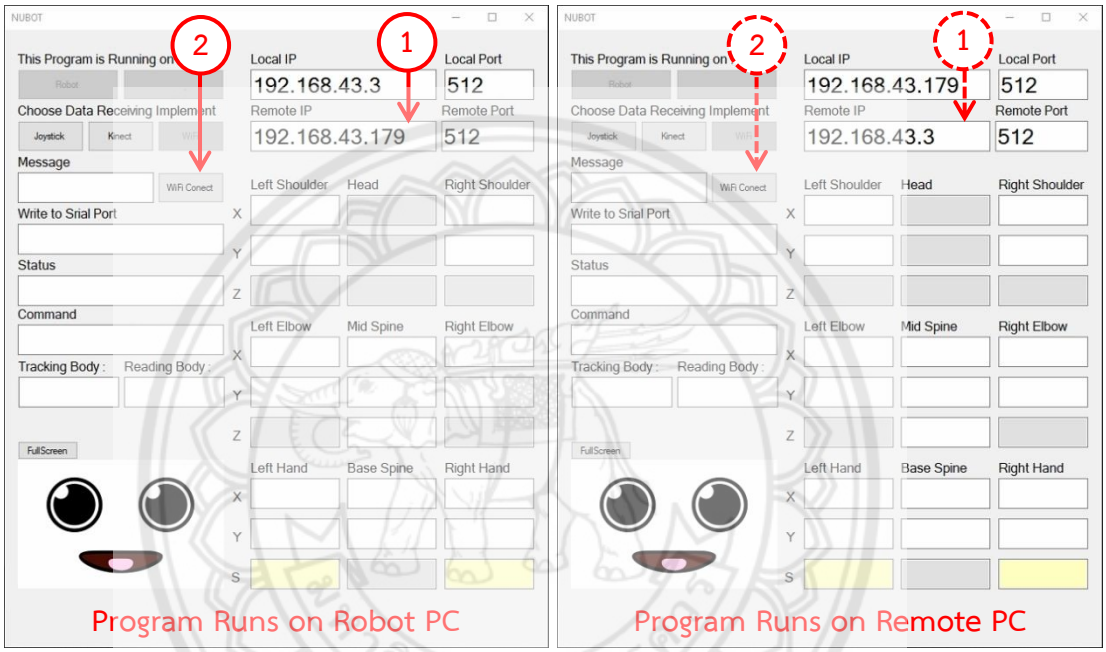
โปรแกรมถูกเขียนขึ้นให้สามารถใช้งานได้ทั้งกับคอมพิวเตอร์บนตัวหุ่นยนต์ และคอมพิวเตอร์ภายนอกซึ่งใช้ในกรณีที่ต้องการสั่งงานระยะไกล สถานะเริ่มต้นของโปรแกรมถูกกำหนดให้เป็นการใช้งานกับคอมพิวเตอร์บนตัวหุ่นยนต์ หากเป็นการใช้งานกับคอมพิวเตอร์ภายนอกผู้ใช้ต้องทำการกำหนดช่องทางการรับส่งข้อมูลระหว่างคอมพิวเตอร์ทั้งสองเครื่องผ่าน Wi-Fi เสียก่อน

3.11.3 การกำหนดช่องทางการรับส่งข้อมูลผ่าน Wi-Fi

ขั้นตอนการกำหนดช่องทางการรับส่งข้อมูลผ่าน Wi-Fi ดูรูปที่ 3.33 มีรายละเอียดดังนี้

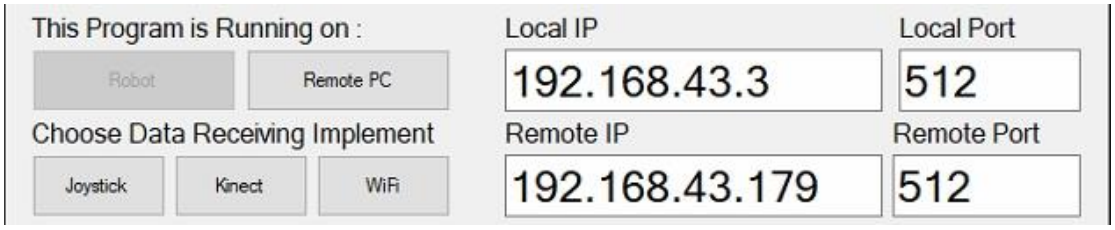
1) เปลี่ยนหมายเลขในช่อง “Remote IP” ให้เป็นหมายเลข IP Address ของเครื่องคอมพิวเตอร์อีกเครื่องหนึ่งที่ต้องการเชื่อมต่อ สามารถดูได้จากช่อง “Local IP” ซึ่งจะแสดงหมายเลข IP Address ของเครื่องคอมพิวเตอร์ที่โปรแกรมนั้นถูกเปิดขึ้น

2) คลิกที่ปุ่ม “WiFi Connect”



รูปที่ 3.33 การกำหนดช่องทางการรับส่งข้อมูลผ่าน Wi-Fi

หลังจากกำหนดช่องทางการรับส่งข้อมูลแล้ว จะพบว่าที่เมนู “Choose Data Receiving Implement” สามารถคลิกที่ปุ่ม “WiFi” เพื่อเลือกใช้วิธีการสั่งงานระยะไกลด้วยท่าทางผ่าน Wi-Fi ได้ รวมทั้ง ที่เมนู “This Program is Running On :” ก็สามารถคลิกที่ปุ่ม “Remote PC” เพื่อเป็นการระบุว่าโปรแกรมนี้กำลังทำงานอยู่บนคอมพิวเตอร์ภายนอกได้แล้วเช่นกัน ดังรูปที่ 3.34

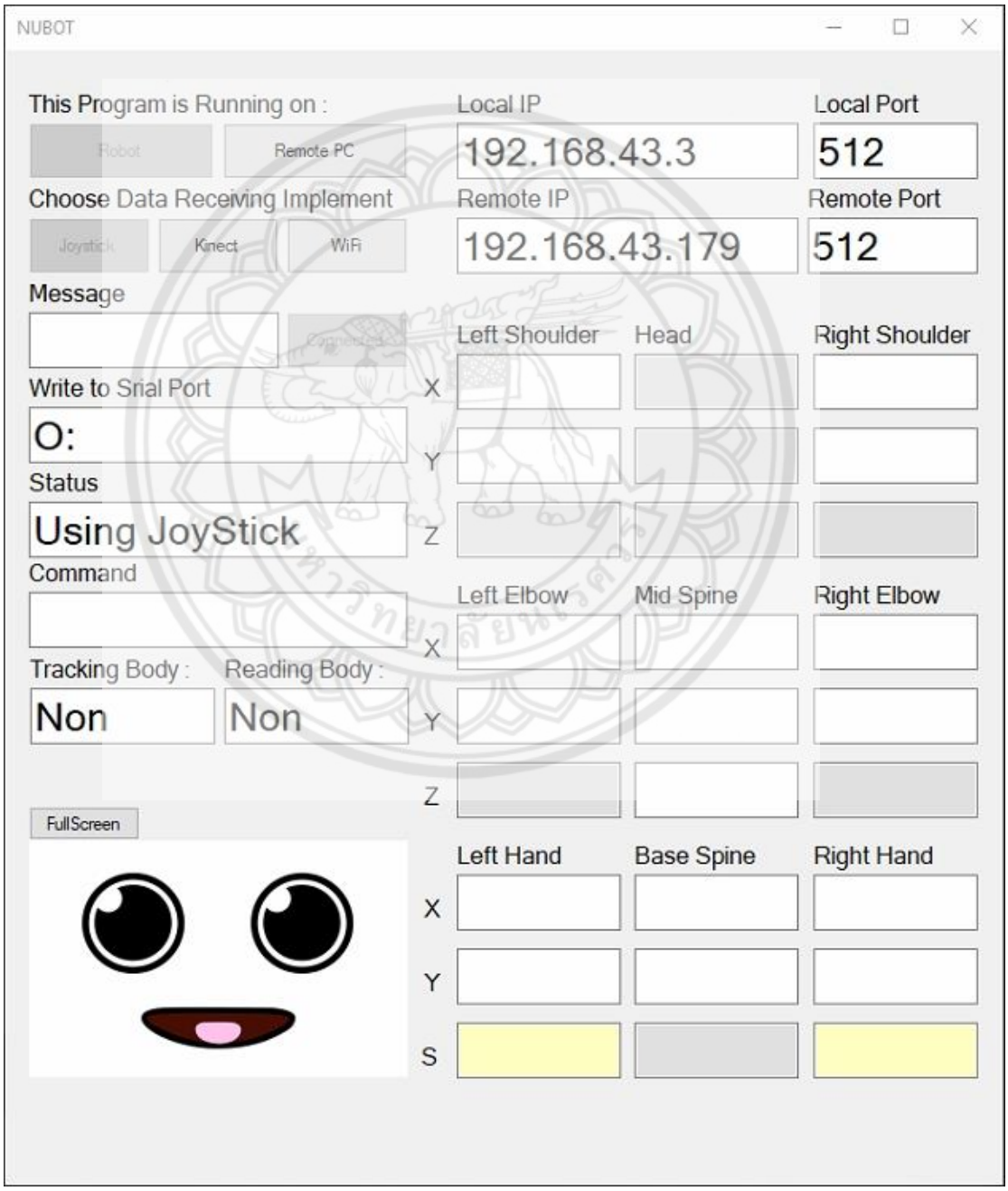


รูปที่ 3.34 ตัวเลือกที่เพิ่มขึ้นหลังจากกำหนดช่องทางการรับส่งข้อมูลผ่าน Wi-Fi

3.11.4 การสั่งงานหุ่นยนต์ด้วยวิธีการตามที่ผู้ใช้งานเลือกใช้

3.11.4.1 สั่งงานด้วยการกดปุ่มบน Joystick

คลิกที่ปุ่ม “Joystick” ที่เมนู “Choose Data Receiving Implement” หุ่นยนต์จะแสดงสีหน้าแบบเต็มจอ สามารถทำให้ย่อกลับได้โดยคลิกที่แถบด้านบน เมื่อคลิกที่แถบดังกล่าวแล้ว หน้าจอแสดงผลจะเป็นดังรูปที่ 3.35 การใช้งาน Joystick เพื่อสั่งงานหุ่นยนต์ กำหนดให้ผู้ใช้มีทางเลือกในการกดปุ่มเพื่อสั่งงานได้ถึงสองรูปแบบ แสดงดังตารางที่ 3.12 ดูรูปที่ 3.36 ประกอบ



รูปที่ 3.35 หน้าจอของโปรแกรมสำหรับสั่งงานด้วย Joystick

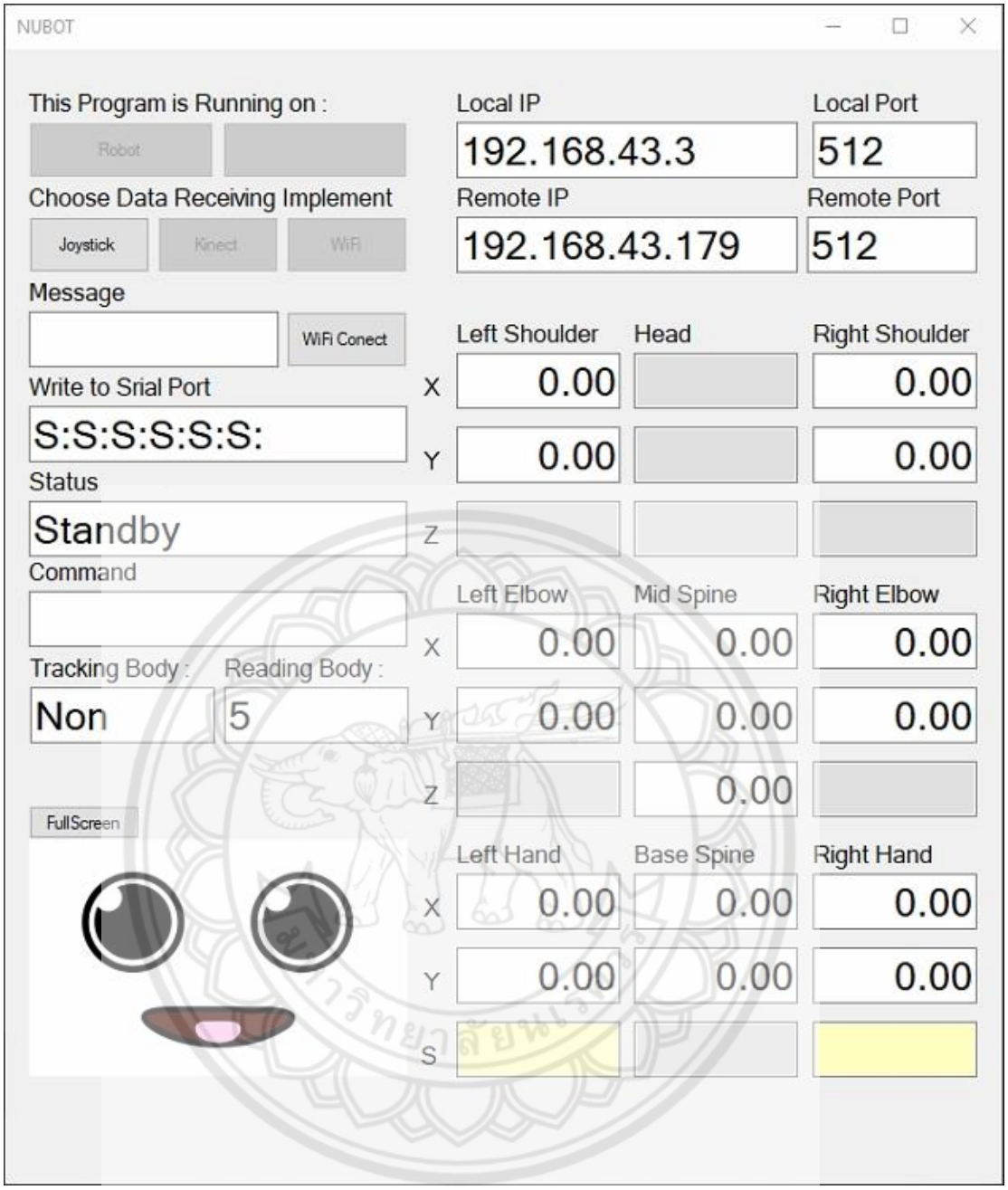
ตารางที่ 3.12 การใช้งาน Joystick เพื่อสั่งงานหุ่นยนต์

รูปแบบการเคลื่อนที่		ทางเลือกที่ 1	ทางเลือกที่ 2
ความเร็วสูง	เดินหน้า	กดปุ่ม Δ	กดปุ่ม L1 หรือ R1 พร้อมกับปุ่ม ^
	ถอยหลัง	กดปุ่ม X	กดปุ่ม L1 หรือ R1 พร้อมกับปุ่ม v
	ไปทางซ้าย	กดปุ่ม □	กดปุ่ม L1 หรือ R1 พร้อมกับปุ่ม <
	ไปทางขวา	กดปุ่ม O	กดปุ่ม L1 หรือ R1 พร้อมกับปุ่ม >
ความเร็วต่ำ	เดินหน้า	กดปุ่ม ^	กดปุ่ม L1 หรือ R1 พร้อมกับปุ่ม Δ
	ถอยหลัง	กดปุ่ม v	กดปุ่ม L1 หรือ R1 พร้อมกับปุ่ม X
	ไปทางซ้าย	กดปุ่ม <	กดปุ่ม L1 หรือ R1 พร้อมกับปุ่ม □
	ไปทางขวา	กดปุ่ม >	กดปุ่ม L1 หรือ R1 พร้อมกับปุ่ม O
	หมุนซ้าย	กดปุ่ม L2	
	หมุนขวา	กดปุ่ม R2	
หยุด		ไม่กดปุ่มใด ๆ	

รูปที่ 3.36 รูปภาพประกอบตารางที่ 3.12 การใช้งาน Joystick เพื่อสั่งงานหุ่นยนต์

3.11.4.2 สั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์ (Kinect On Robot)

คลิกที่ปุ่ม “Kinect” ที่เมนู “Choose Data Receiving Implement” หุ่นยนต์จะแสดงสีหน้าแบบเต็มจอ สามารถทำให้ย่อกลับได้โดยคลิกที่แถบด้านบน เมื่อคลิกที่แถบดังกล่าวแล้ว หน้าจอแสดงผลจะเป็นดังรูปที่ 3.37 ภายใต้การสั่งงานด้วยท่าทางหุ่นยนต์มีสถานะการทำงาน 3 สถานะ มีรายละเอียดดังที่กล่าวไปแล้วในหัวข้อที่ 3.6



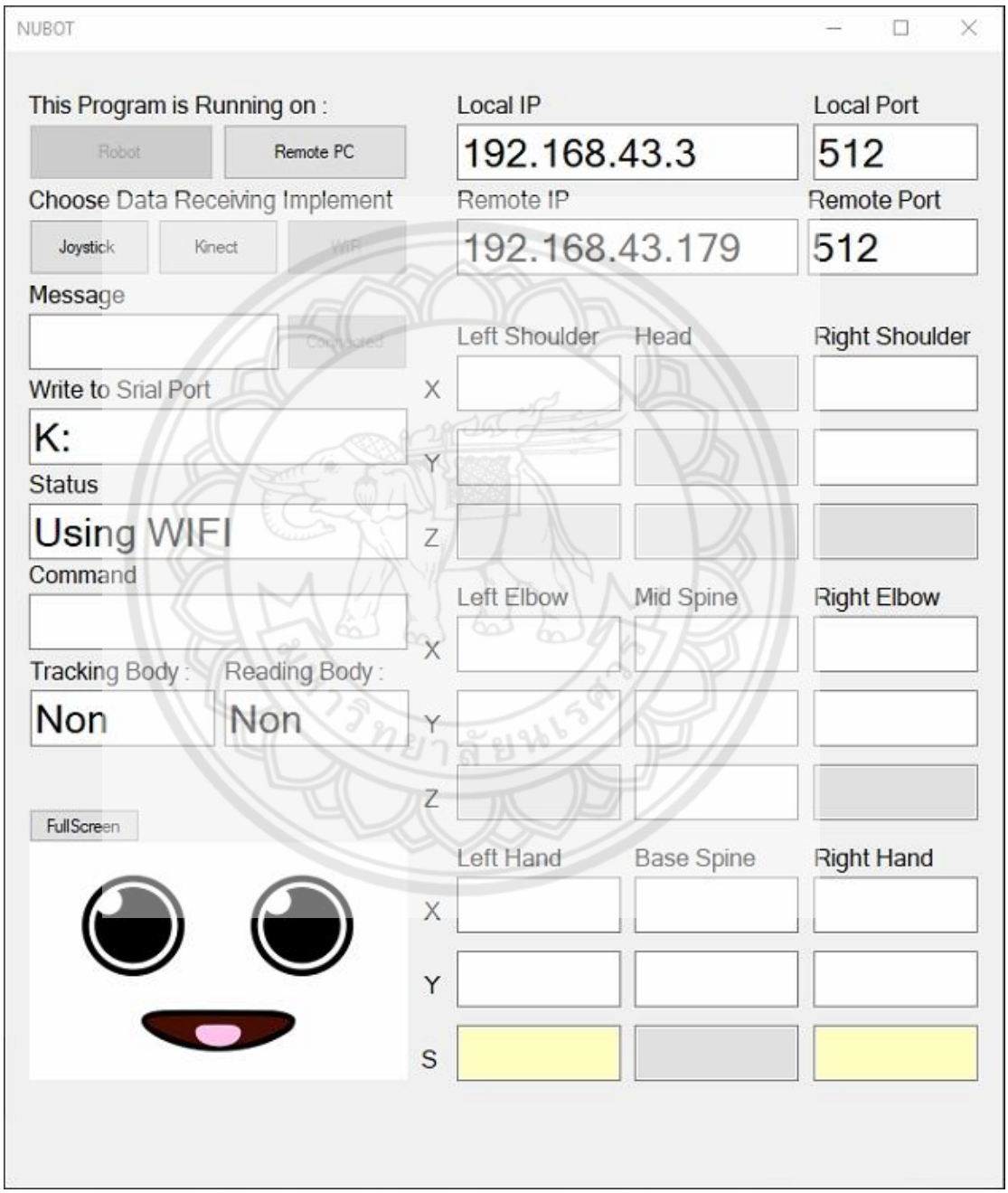
รูปที่ 3.37 หน้าจอของโปรแกรมสำหรับสั่งงานด้วยท่าทางผ่าน Kinect บนตัวหุ่นยนต์

3.11.4.3 สั่งงานระยะไกลด้วยท่าทาง (Kinect On Remote PC)

มีการทำงานของโปรแกรมสองส่วน คือ โปรแกรมที่เปิดกับคอมพิวเตอร์บนตัวหุ่นยนต์ และโปรแกรมที่เปิดกับคอมพิวเตอร์ภายนอกซึ่งใช้ในการสั่งงานระยะไกล ผู้ใช้ต้องทำการกำหนดช่องทางการรับส่งข้อมูลระหว่างคอมพิวเตอร์ทั้งสองเครื่องผ่าน Wi-Fi เสียก่อน เมื่อกำหนดช่องทางการรับส่งข้อมูลเรียบร้อยแล้วให้ทำดังนี้

ก. **โปรแกรมที่เปิดกับคอมพิวเตอร์บนตัวหุ่นยนต์**

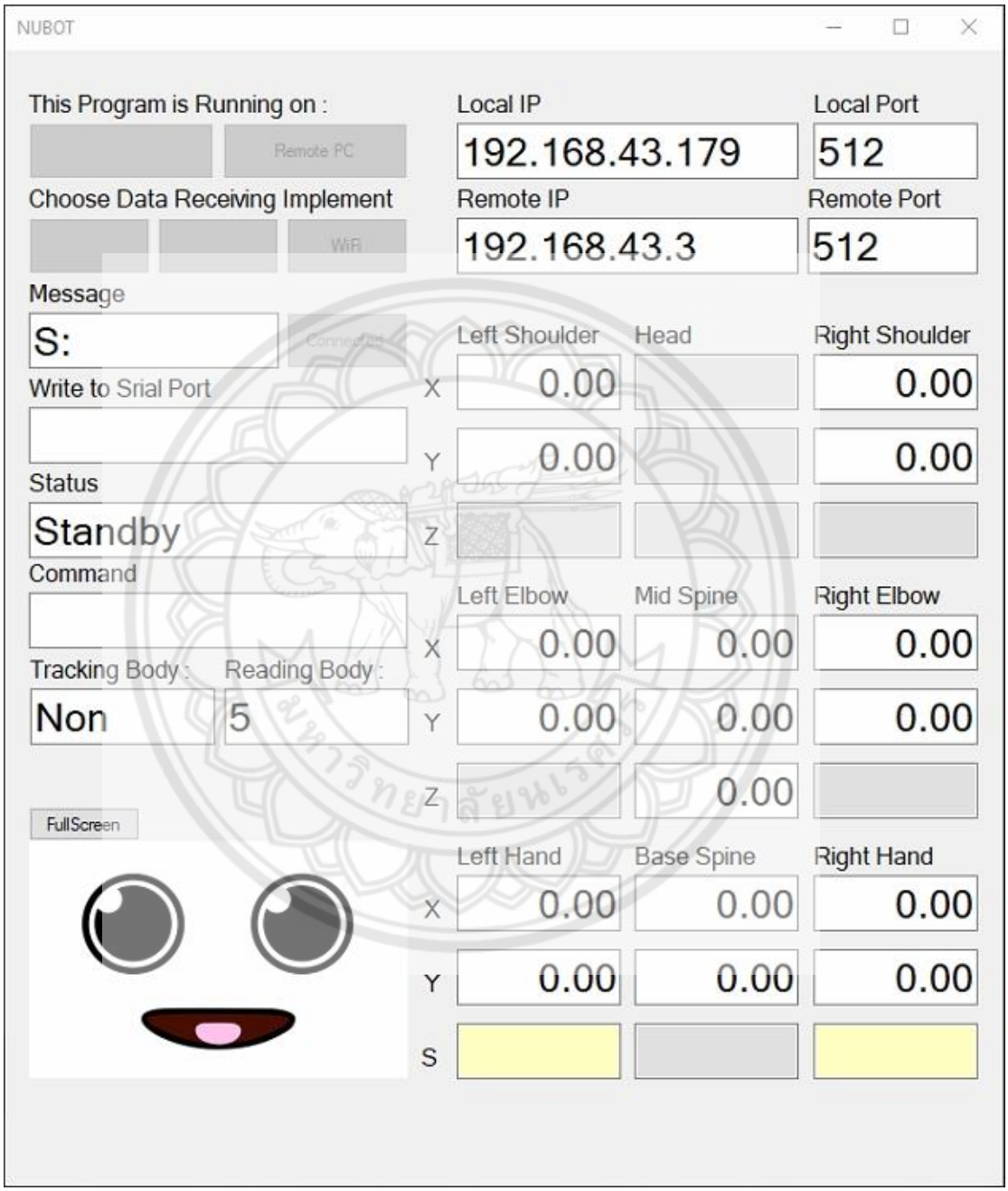
คลิกที่ปุ่ม “WiFi” ที่เมนู “Choose Data Receiving Implement” หุ่นยนต์จะแสดงสีหน้าแบบเต็มจอ สามารถทำให้ย่อกลับได้โดยคลิกที่แถบด้านบน เมื่อคลิกที่แถบดังกล่าวแล้ว หน้าจอแสดงผลจะเป็นดังรูปที่ 3.38



รูปที่ 3.38 หน้าจอของโปรแกรมที่เปิดกับคอมพิวเตอร์บนตัวหุ่นยนต์ ซึ่งพร้อมรับคำสั่งผ่าน Wi-Fi

ข. **โปรแกรมที่เปิดกับคอมพิวเตอร์ภายนอก**

คลิกที่ปุ่ม “Remote PC” ที่เมนู “This Program is Running On :” หน้าจอแสดงผลจะเป็นดังรูปที่ 3.39



รูปที่ 3.39 หน้าจอของโปรแกรมที่เปิดกับคอมพิวเตอร์ภายนอก ซึ่งพร้อมส่งคำสั่งผ่าน Wi-Fi

บทที่ 4
การทดสอบการทำงานของระบบที่พัฒนาขึ้น

ในบทนี้จะกล่าวถึงผลการทดสอบการเคลื่อนที่ของหุ่นยนต์ และความแม่นยำของการตรวจจับระยะของเซนเซอร์อัลทราโซนิก รวมทั้งผลทดสอบความถูกต้องของการรับคำสั่งจากอุปกรณ์ Kinect โดยมีรายละเอียดเป็น ดังนี้

4.1 การทดสอบความถูกต้องของการสั่งงานด้วยท่าทาง

ในการทดลองนี้กำหนดให้มีเงื่อนไขของการสั่งงานด้วยท่าทาง 2 เงื่อนไขหลักด้วยกัน คือ เงื่อนไขเรื่องแสงสว่าง (แสงมากและแสงปกติ) และ เงื่อนไขเรื่องระยะห่าง (ระยะใกล้ คือ 1.5 เมตร ระยะปานกลาง คือ 2.5 เมตร และระยะไกล คือ 4 เมตร) กำหนดให้ผู้สั่งงานทำท่าทางตามเงื่อนไขที่กล่าวไว้ในหัวข้อที่ 3.6 ทำการบันทึกผลลัพธ์ของการตัดสินท่าทางของระบบว่าถูกต้องหรือไม่ โดยมีการทำซ้ำทั้งหมด 10 ครั้ง ผลการทดลองแสดงดังตารางที่ 4.1 ถึง 4.6

ตารางที่ 4.1 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงมากและอยู่ในระยะใกล้

พฤติกรรม	ครั้งที่										ร้อยละความถูกต้อง
	1	2	3	4	5	6	7	8	9	10	
รับคำสั่ง	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ติดตาม	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ถอยหลัง (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ถอยหลัง (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100

ตารางที่ 4.1 (ต่อ) ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงมากและอยู่ในระยะใกล้

พฤติกรรม	ครั้งที่										ร้อยละความถูกต้อง
	1	2	3	4	5	6	7	8	9	10	
หมุนซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หยุด	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ค่าเฉลี่ยของร้อยละความถูกต้อง											100

ตารางที่ 4.2 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงมากและอยู่ในระยะปานกลาง

พฤติกรรม	ครั้งที่										ร้อยละความถูกต้อง
	1	2	3	4	5	6	7	8	9	10	
รับคำสั่ง	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ติดตาม	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ถอยหลัง (ท่าที่1)	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	90
ถอยหลัง (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนขวา (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หยุด	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	90
ค่าเฉลี่ยของร้อยละความถูกต้อง											98.94

ตารางที่ 4.3 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงมากและอยู่ในระยะไกล

พฤติกรรม	ครั้งที่										ร้อยละ
	1	2	3	4	5	6	7	8	9	10	ความถูกต้อง
รับคำสั่ง	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ติดตาม	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่1)	✗	✗	✓	✓	✓	✓	✗	✓	✓	✗	60
เดินหน้า (ท่าที่2)	✗	✓	✗	✗	✓	✓	✗	✓	✗	✓	50
ถอยหลัง (ท่าที่1)	✓	✗	✓	✗	✓	✓	✗	✗	✓	✗	50
ถอยหลัง (ท่าที่2)	✗	✓	✓	✗	✗	✓	✓	✓	✗	✓	60
ไปทางซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✗	✓	✗	✗	70
ไปทางซ้าย (ท่าที่2)	✓	✓	✗	✗	✓	✓	✗	✓	✓	✓	70
ไปทางขวา (ท่าที่1)	✓	✓	✗	✓	✓	✓	✓	✗	✗	✓	70
ไปทางขวา (ท่าที่2)	✗	✓	✓	✗	✓	✗	✗	✓	✓	✓	60
หมุนซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนซ้าย (ท่าที่2)	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	90
หมุนขวา (ท่าที่1)	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	90
หมุนขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หยุด	✓	✓	✗	✓	✓	✓	✓	✓	✗	✗	70
ค่าเฉลี่ยของร้อยละความถูกต้อง											78.82

ตารางที่ 4.4 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงปกติและอยู่ในระยะใกล้

พฤติกรรม	ครั้งที่										ร้อยละ
	1	2	3	4	5	6	7	8	9	10	ความถูกต้อง
รับคำสั่ง	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ติดตาม	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100

ตารางที่ 4.4 (ต่อ) ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงปกติและอยู่ในระยะใกล้

พฤติกรรม	ครั้งที่										ร้อยละความถูกต้อง
	1	2	3	4	5	6	7	8	9	10	
ถอยหลัง (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ถอยหลัง (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนขวา (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หยุด	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ค่าเฉลี่ยของร้อยละความถูกต้อง											100

ตารางที่ 4.5 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงปกติและอยู่ในระยะปานกลาง

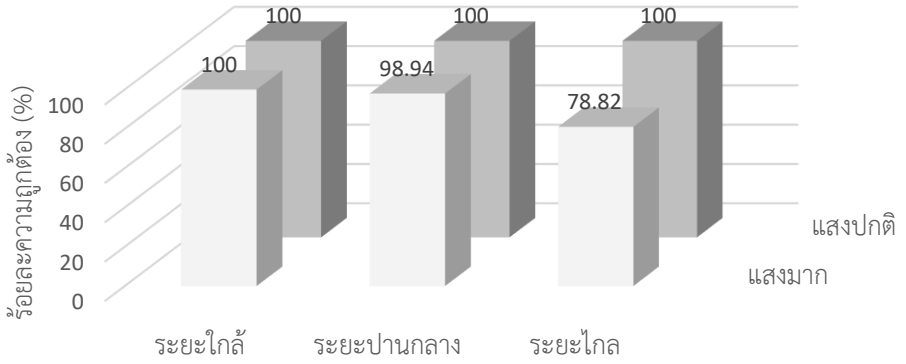
พฤติกรรม	ครั้งที่										ร้อยละความถูกต้อง
	1	2	3	4	5	6	7	8	9	10	
รับคำสั่ง	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ติดตาม	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ถอยหลัง (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ถอยหลัง (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100

ตารางที่ 4.5 (ต่อ) ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีมีแสงปกติและระยะปานกลาง

พฤติกรรม	ครั้งที่										ร้อยละความถูกต้อง
	1	2	3	4	5	6	7	8	9	10	
หมุนซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนขวา (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หยุด	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ค่าเฉลี่ยของร้อยละความถูกต้อง											100

ตารางที่ 4.6 ความถูกต้องในการตัดสินท่าทางการสั่งงานกรณีที่มีแสงปกติและอยู่ในระยะไกล

พฤติกรรม	ครั้งที่										ร้อยละความถูกต้อง
	1	2	3	4	5	6	7	8	9	10	
รับคำสั่ง	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ติดตาม	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
รอ (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
เดินหน้า (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ถอยหลัง (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ถอยหลัง (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ไปทางขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนซ้าย (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนซ้าย (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนขวา (ท่าที่1)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หมุนขวา (ท่าที่2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
หยุด	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
ค่าเฉลี่ยของร้อยละความถูกต้อง											100



รูปที่ 4.1 แผนภูมิแสดงค่าร้อยละความถูกต้องเปรียบเทียบระหว่างกรณีที่มีแสงมากกับกรณีที่มีแสงปกติที่ระยะทางต่างกัน

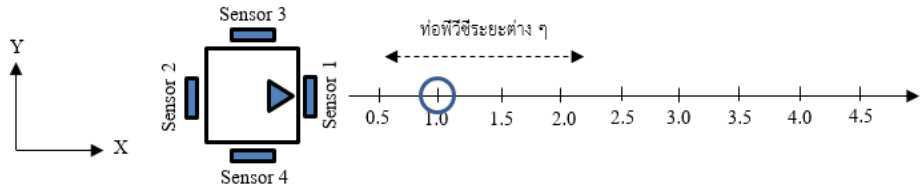
พิจารณาจากค่าร้อยละความถูกต้องจะพบว่า ในกรณีที่มีแสงปกติระบบสามารถตัดสินท่าทางการสั่งงานได้ถูกต้องทั้งหมด ขณะที่ในกรณีที่มีแสงมากค่าร้อยละของความถูกต้องในการตัดสินท่าทางการสั่งงานจะมีค่าน้อยกว่า ดังนั้น จะเห็นว่าปัจจัยหลักที่ส่งผลต่อการตัดสินท่าทางการสั่งงาน คือ แสง เนื่องจากอุปกรณ์ Kinect ใช้เซนเซอร์อินฟราเรดในการตรวจวัดระยะจึงทำงานได้ดีในที่ที่มีแสงปกติ นอกจากปัจจัยเรื่องแสงแล้ว ค่าร้อยละของความถูกต้องยังแสดงให้เห็นว่าระยะทางก็มีผลต่อการตัดสินท่าทางการสั่งงาน ระยะห่างที่มากเกินไปส่งผลต่อความสามารถในการวัดระยะทำให้ไม่สามารถประมวลภาพเป็นลักษณะของโครงกระดูกที่ชัดเจนจึงมีโอกาสตัดสินท่าทางผิดพลาดได้มากขึ้น

4.2 การทดสอบความแม่นยำของการตรวจจับระยะของเซนเซอร์อัลทราโซนิก

ในการทดสอบการทำงานของเซนเซอร์อัลทราโซนิกได้เลือกใช้วัสดุทดสอบเป็นท่อพีวีซีขนาดเส้นผ่านศูนย์กลาง 10 เซนติเมตร สูง 70 เซนติเมตรโดยนำมาวางที่ระยะต่างๆ ดังรายละเอียดต่อไปนี้

4.2.1 ระยะที่เซนเซอร์อัลทราโซนิกสามารถวัดได้ (Sensor Range)

การทดลองต่อไปนี้ดำเนินการโดยกำหนดระยะทดสอบขึ้นแบ่งเป็นช่วง ช่วงละ 0.5 เมตร โดยวางวัสดุทดสอบตรงตำแหน่งที่กำหนด จากนั้นใช้เซนเซอร์อัลทราโซนิกซึ่งติดตั้งบนตัวหุ่นยนต์วัดระยะทางแล้วบันทึกค่าที่ได้ โดยในแต่ละระยะจะทำการวัดค่าทั้งหมด 8 ครั้ง แล้วนำค่าที่วัดได้ทั้งหมดมาหาค่าความคลาดเคลื่อนเฉลี่ย จากนั้นจึงทำการทดลองกับเซนเซอร์อัลทราโซนิกจนครบ 4 ตัว รูปที่ 4.2 แสดงวิธีการทดสอบการตรวจจับระยะของเซนเซอร์อัลทราโซนิก ผลการทดลองแสดงได้ดังตารางที่ 4.7 - 4.10



รูปที่ 4.2 การทดลองหาระยะที่เซนเซอร์อัลทราโซนิกสามารถวัดได้

ตารางที่ 4.7 ความแม่นยำของการวัดระยะของเซนเซอร์อัลทราโซนิกตัวที่ 1

ระยะจริง (เมตร)	ระยะที่วัดได้ (เมตร)									ร้อยละความคลาดเคลื่อน
	1	2	3	4	5	6	7	8	ค่าเฉลี่ย	
0.5	0.49	0.49	0.49	0.49	0.49	0.49	0.48	0.49	0.489	-2.25
1.0	0.98	0.98	0.98	1.00	0.98	0.99	0.98	0.97	0.983	-1.75
1.5	1.69	0.70	1.48	1.39	1.47	1.48	1.46	1.29	1.370	-8.67
2.0	1.00	1.98	1.94	1.99	1.94	1.96	1.99	1.98	1.848	-7.63
2.5	2.42	2.48	2.27	2.29	1.35	2.49	2.50	2.48	2.285	-8.60
3.0	2.17	2.97	2.99	2.78	2.99	3.16	5.26	3.15	3.184	6.12
3.5	3.49	3.63	3.49	3.49	5.06	5.25	5.25	5.25	4.364	24.68
4.0	3.97	5.26	5.25	5.07	5.26	5.06	5.26	5.26	5.049	26.22
4.5	5.04	5.26	5.26	5.25	5.26	5.26	5.26	5.26	5.231	16.25

ตารางที่ 4.8 ความแม่นยำของการวัดระยะของเซนเซอร์อัลทราโซนิกตัวที่ 2

ระยะจริง (เมตร)	ระยะที่วัดได้ (เมตร)									ร้อยละความคลาดเคลื่อน
	1	2	3	4	5	6	7	8	ค่าเฉลี่ย	
0.5	0.50	0.51	0.50	0.51	0.51	0.50	0.51	0.49	0.504	0.75
1.0	1.00	0.99	1.00	1.00	0.99	0.99	1.00	1.00	0.996	-0.38
1.5	0.71	1.49	1.49	1.50	1.47	1.51	1.50	1.49	1.395	-7.00
2.0	1.99	1.91	2.00	2.03	2.02	1.99	2.91	1.03	1.985	-0.75
2.5	2.50	2.71	2.65	2.52	2.53	1.31	2.54	2.79	2.444	-2.25
3.0	3.02	3.04	3.16	3.30	2.54	3.05	3.17	3.00	3.035	1.17
3.5	5.24	3.49	3.50	5.24	5.25	3.74	3.64	5.25	4.419	26.25
4.0	5.25	5.24	5.24	5.25	5.25	5.24	5.24	5.25	5.245	31.13
4.5	5.26	5.25	5.26	5.25	5.25	5.26	5.25	5.25	5.254	16.75

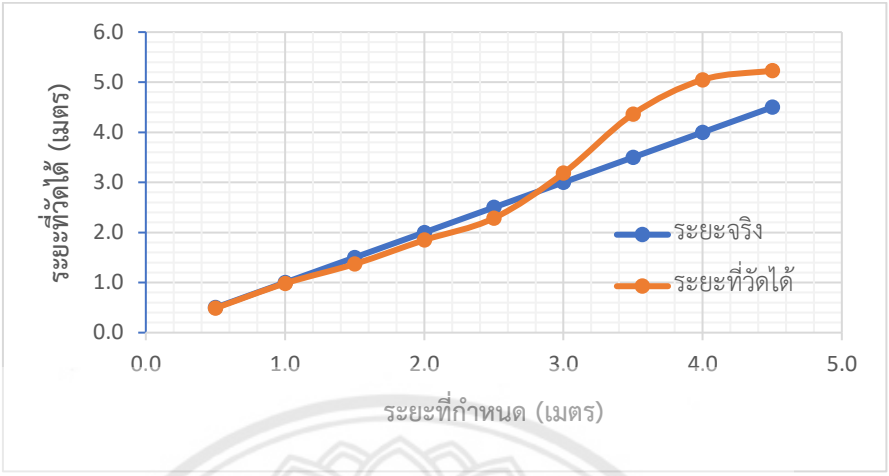
ตารางที่ 4.9 ความแม่นยำของการวัดระยะของเซนเซอร์อัลทราโซนิกตัวที่ 3

ระยะจริง (เมตร)	ระยะที่วัดได้ (เมตร)									ร้อยละความคลาดเคลื่อน
	1	2	3	4	5	6	7	8	ค่าเฉลี่ย	
0.5	0.50	0.50	0.49	0.50	0.50	0.50	0.50	0.50	0.499	-0.25
1.0	1.01	1.00	0.99	0.98	0.99	1.01	0.99	0.99	0.995	-0.50
1.5	1.51	1.49	1.48	1.48	1.49	1.50	1.22	0.48	1.331	-11.25
2.0	1.13	2.01	2.00	1.99	2.00	1.94	1.99	1.95	1.876	-6.19
2.5	2.51	2.50	1.23	2.52	2.65	2.54	2.51	2.65	2.389	-4.45
3.0	3.15	3.17	3.01	5.27	2.99	3.01	3.02	2.99	3.326	10.88
3.5	3.48	3.64	5.27	5.27	3.62	5.27	3.47	5.28	4.413	26.07
4.0	4.01	5.27	5.27	4.00	5.28	5.27	5.27	5.27	4.955	23.88
4.5	5.27	5.28	5.28	5.28	5.28	5.28	5.28	5.27	5.278	17.28

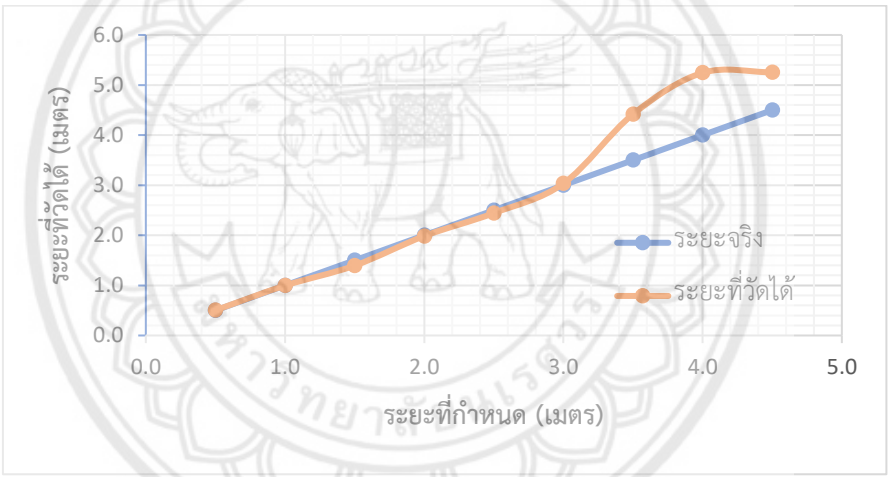
ตารางที่ 4.10 ความแม่นยำของการวัดระยะของเซนเซอร์อัลทราโซนิก**ตัวที่** 4

ระยะจริง (เมตร)	ระยะที่วัดได้ (เมตร)									ร้อยละความคลาดเคลื่อน
	1	2	3	4	5	6	7	8	ค่าเฉลี่ย	
0.5	0.50	0.50	0.50	0.49	0.50	0.49	0.50	0.50	0.498	-0.50
1.0	1.00	1.00	0.99	0.99	1.00	0.99	0.99	0.99	0.994	-0.62
1.5	1.50	1.48	1.49	1.48	0.85	1.49	0.92	1.49	1.338	-10.83
2.0	2.01	1.99	1.19	1.16	1.99	1.99	2.00	2.01	1.793	-10.38
2.5	2.55	2.50	2.51	2.50	2.56	2.28	2.94	2.55	2.549	1.95
3.0	3.04	3.05	3.01	3.02	3.00	3.35	5.28	3.03	3.348	11.58
3.5	3.51	5.28	5.28	5.28	5.28	5.28	3.02	5.28	4.776	36.46
4.0	5.28	5.29	5.29	5.28	5.28	5.29	5.28	5.29	5.285	32.13
4.5	5.29	5.29	5.29	5.29	5.29	5.29	5.28	5.29	5.289	17.53

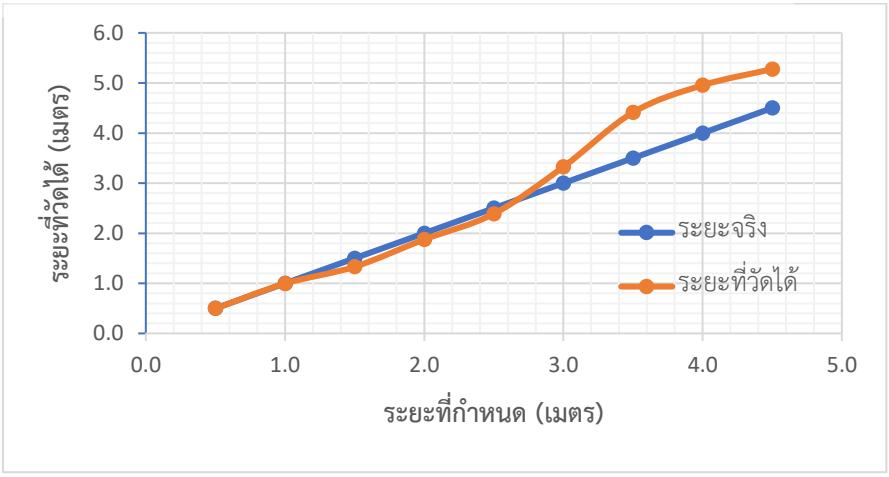
รูปที่ 4.3 – 4.6 แสดงการเปรียบเทียบระหว่างระยะที่เซนเซอร์อัลทราโซนิกวัดได้กับระยะทางจริง



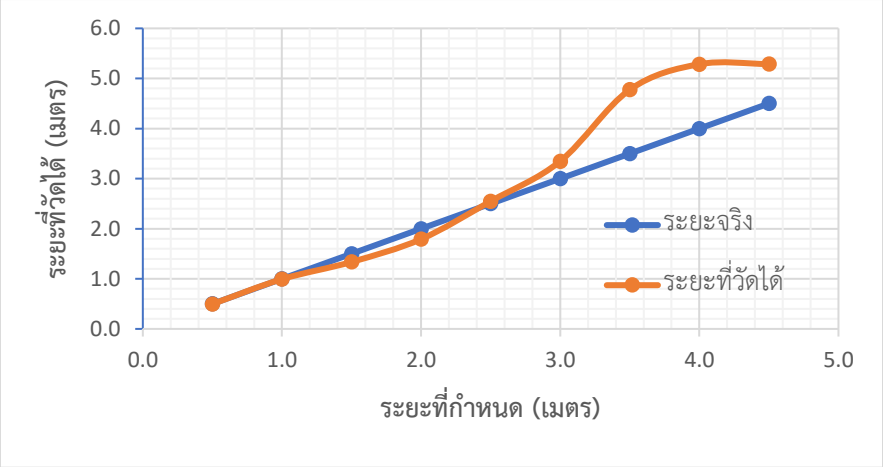
รูปที่ 4.3 ระยะทางที่เซนเซอร์อัลทราโซนิกตัวที่ 1 วัดได้เปรียบเทียบกับระยะทางจริง



รูปที่ 4.4 ระยะทางที่เซนเซอร์อัลทราโซนิกตัวที่ 2 วัดได้เปรียบเทียบกับระยะทางจริง

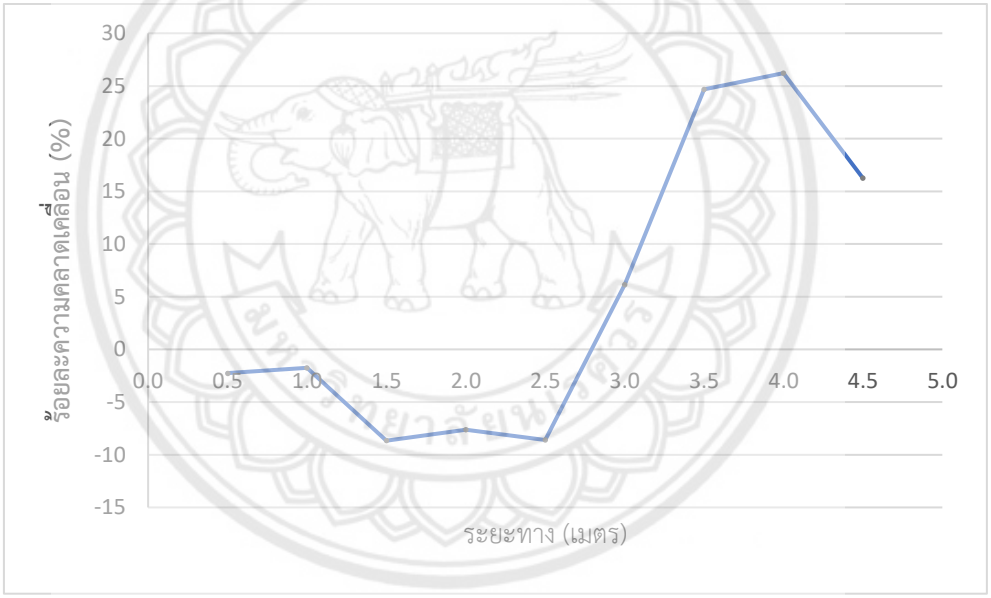


รูปที่ 4.5 ระยะทางที่เซนเซอร์อัลทราโซนิกตัวที่ 3 วัดได้เปรียบเทียบกับระยะทางจริง

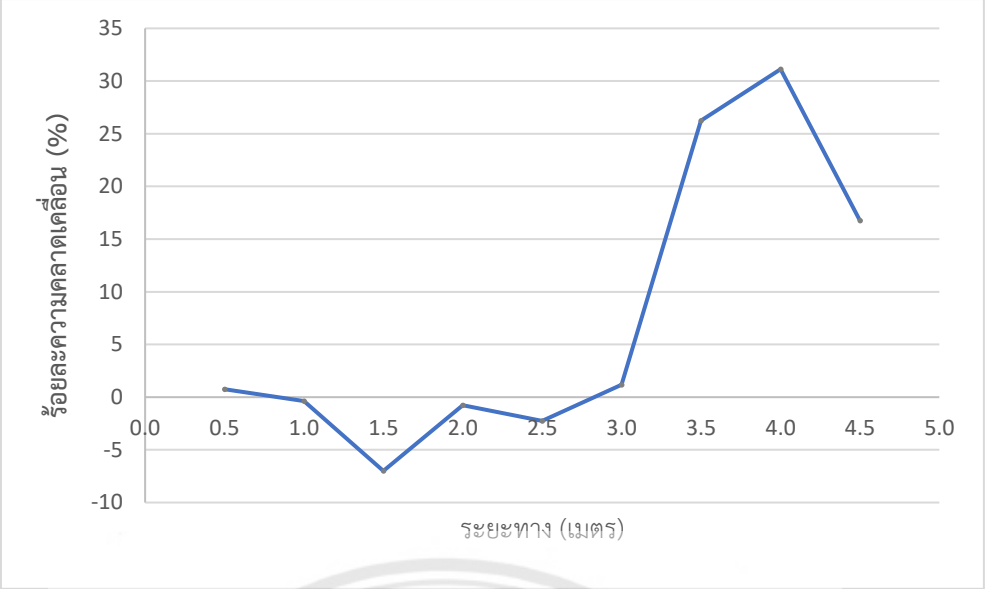


รูปที่ 4.6 ระยะทางที่เซนเซอร์อัลทราโซนิกตัวที่ 4 วัดได้เปรียบเทียบกับระยะทางจริง

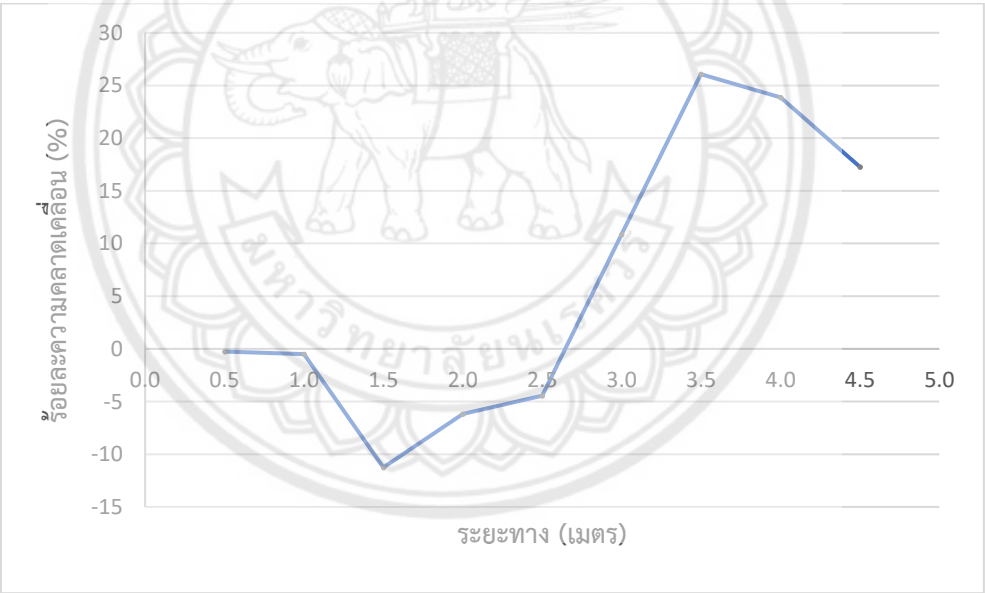
รูปที่ 4.7 – 4.10แสดงค่าร้อยละความคลาดเคลื่อนที่ระยะทางต่าง ๆ



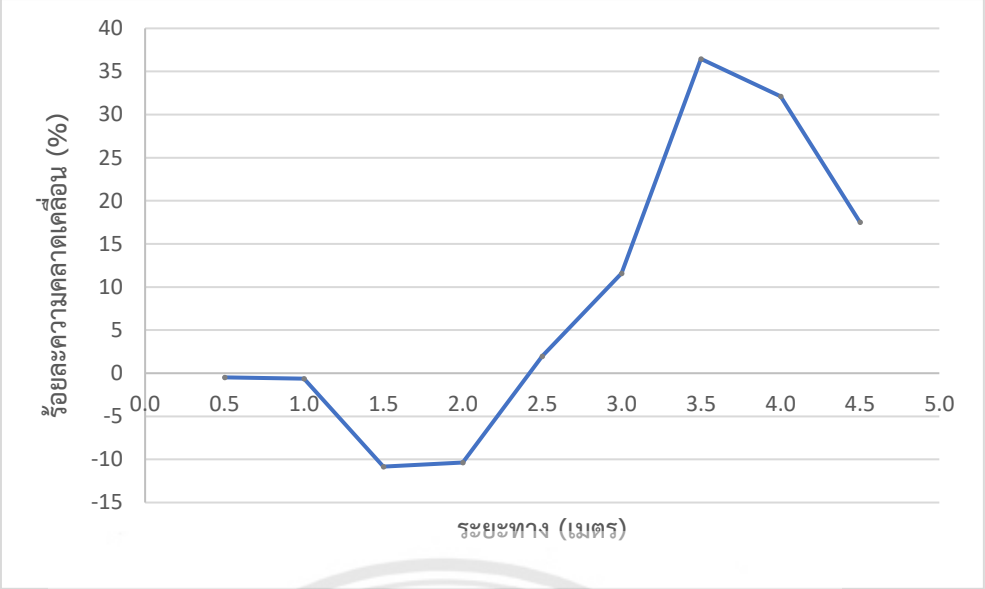
รูปที่ 4.7 กราฟแสดงค่าร้อยละความคลาดเคลื่อนที่ระยะทางต่างๆ เมื่อวัดระยะทางด้วยเซนเซอร์อัลทราโซนิกตัวที่ 1



รูปที่ 4.8 กราฟแสดงค่าร้อยละความคลาดเคลื่อนที่ระยะทางต่างๆ เมื่อวัดระยะทางด้วยเซนเซอร์อัลทราโซนิกตัวที่ 2



รูปที่ 4.9 กราฟแสดงค่าร้อยละความคลาดเคลื่อนที่ระยะทางต่างๆ เมื่อวัดระยะทางด้วยเซนเซอร์อัลทราโซนิกตัวที่ 3

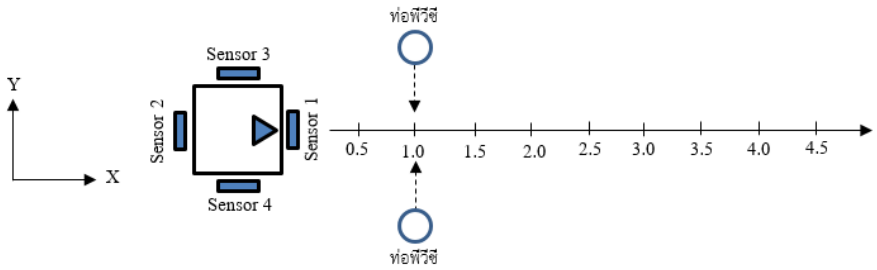


รูปที่ 4.10 กราฟแสดงค่าร้อยละความคลาดเคลื่อนที่ระยะทางต่างๆ เมื่อวัดระยะทางด้วยเซนเซอร์อัลทราโซนิกตัวที่ 4

จากผลการทดลองหาระยะที่เซนเซอร์อัลทราโซนิกสามารถวัดได้ พบว่า ร้อยละของความคลาดเคลื่อนของเซนเซอร์อัลทราโซนิกทั้ง 4 ตัว มีค่าเพิ่มขึ้นมากเมื่อใช้วัดระยะทางที่มากกว่า 3 เมตรขึ้นไป ทั้งนี้สาเหตุอาจเกิดขึ้นมาจากสัญญาณรบกวน และชนิดของวัตถุที่ใช้ ดังนั้นจึงสามารถสรุปได้ว่าระยะหวังผลของเซนเซอร์อัลทราโซนิกที่ใช้ คือ 3 เมตร

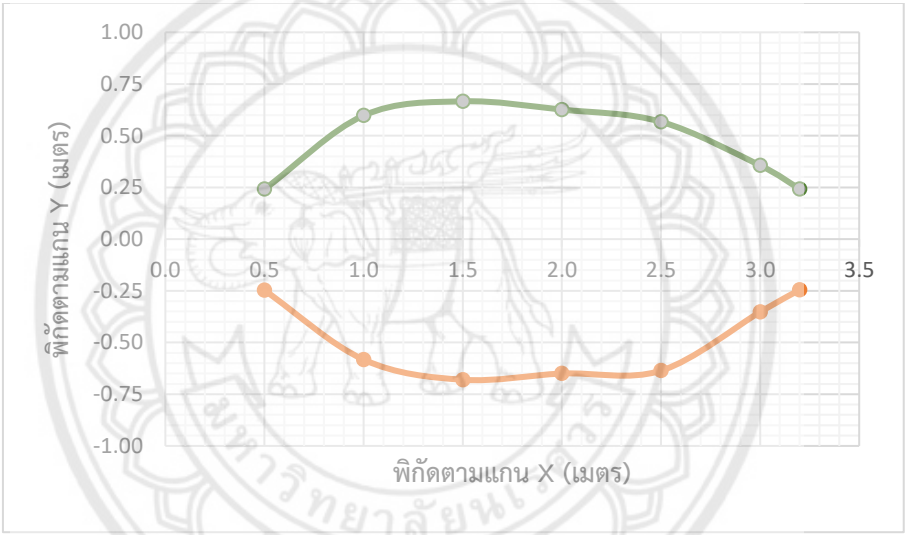
4.2.2 ขนาดของลำคลื่นของเซนเซอร์อัลทราโซนิก (Sensor Beam)

การทดลองนี้เริ่มจากการกำหนดพิกัดฉากลงบนพื้นที่ที่ทำการทดลอง จัดวางเซนเซอร์อัลทราโซนิกตัวที่กำลังดำเนินการทดลองให้ตั้งอยู่ที่พิกัด (0,0) หันไปทางทิศบนแนวแกน X ที่เป็นค่าบวกและตั้งฉากกับแกน Y จากนั้นกำหนดระยะตามแกน X ที่จะใช้ทดสอบซึ่งในที่นี้คือ เริ่มจาก X = 0.5 เมตร และเพิ่มขึ้นทีละ 0.5 เมตร ไม่เกินระยะสูงสุดที่เซนเซอร์อัลทราโซนิกตรวจจับได้ จากนั้นใช้วัสดุทดสอบเลื่อนเข้าตามแกน Y (ที่ระยะ X ที่ทดสอบ) เมื่อเลื่อนเข้ามาจนเซนเซอร์อัลทราโซนิกสามารถตรวจจับวัตถุได้จึงทำการบันทึกค่าพิกัดไว้ จากนั้นจึงทำซ้ำไปเรื่อยๆ จนสุดระยะโดยทำการทดลองซ้ำ 4 ครั้งแล้วนำค่าตามแกน Y ของแต่ละช่วงแกน X ที่กำหนดมาหาค่าเฉลี่ย จากนั้นจึงนำค่ามาวาดกราฟ แล้วจึงทำการทดลองกับเซนเซอร์อัลทราโซนิกจนครบ 4 ตัว รูปที่ 4.11 แสดงการทดลองเพื่อหาขนาดของลำคลื่นของเซนเซอร์อัลทราโซนิก

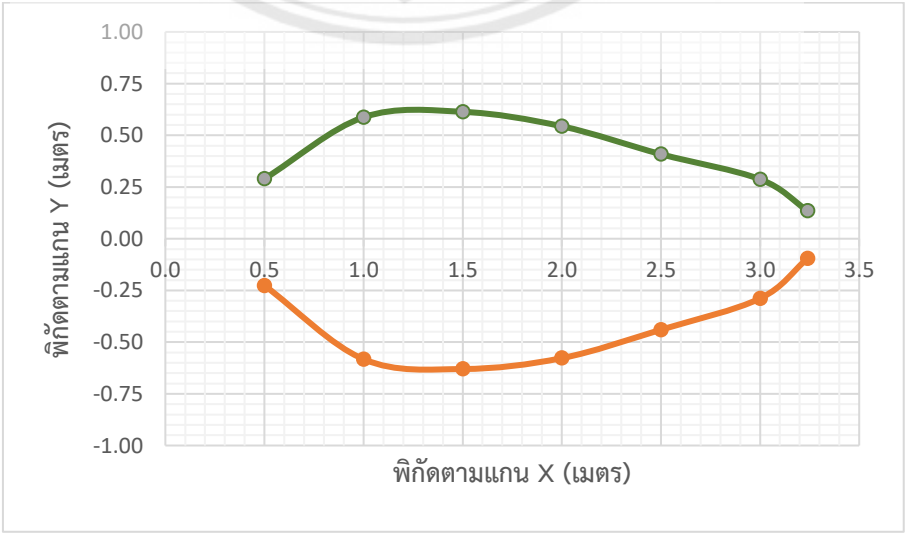


รูปที่ 4.11 การทดลองหาขนาดของลำคลื่นของเซนเซอร์อัลทราโซนิก

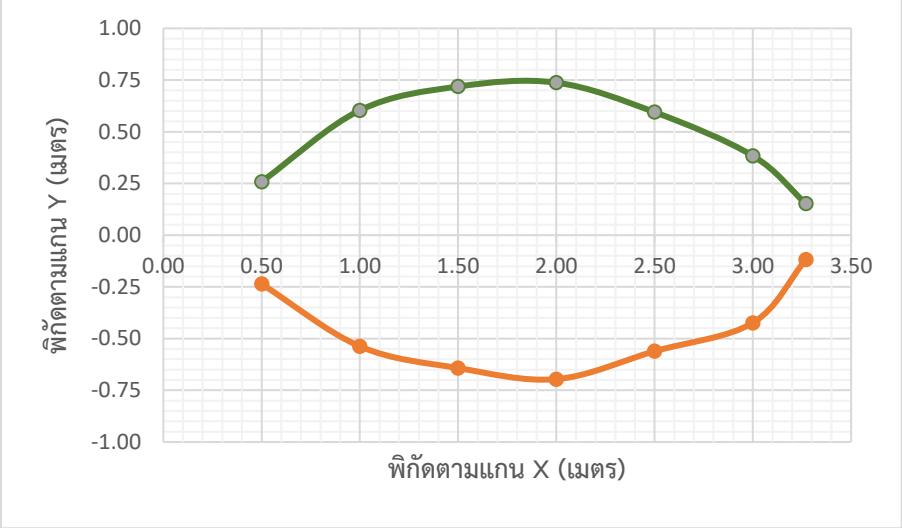
จากการทดลองสามารถแสดงขนาดของลำคลื่นของเซนเซอร์อัลทราโซนิกแต่ละตัวได้ดังรูปที่ 4.12 – 4.15



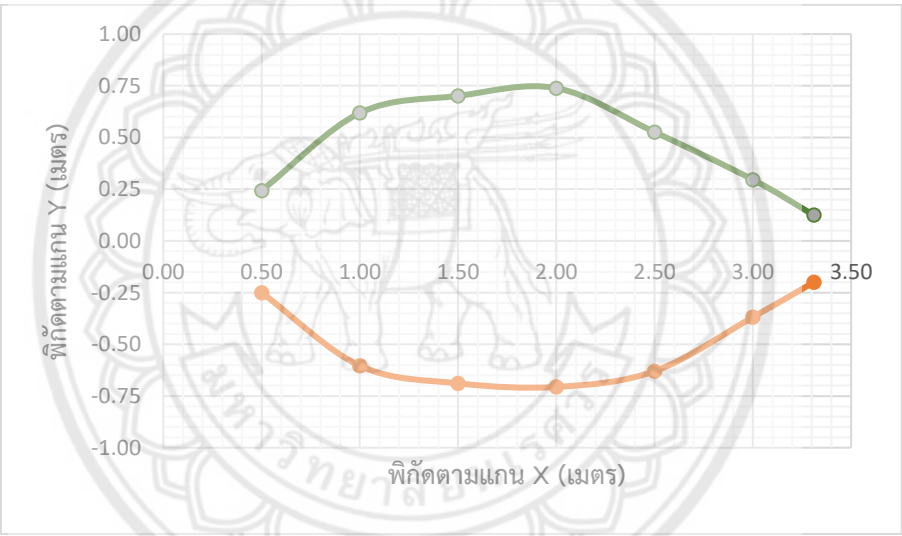
รูปที่ 4.12 ลักษณะบีมของเซนเซอร์อัลทราโซนิกตัวที่ 1



รูปที่ 4.13 ลักษณะบีมของเซนเซอร์อัลทราโซนิกตัวที่ 2



รูปที่ 4.14 ลักษณะบีมของเซนเซอร์อัลทราโซนิกตัวที่ 3



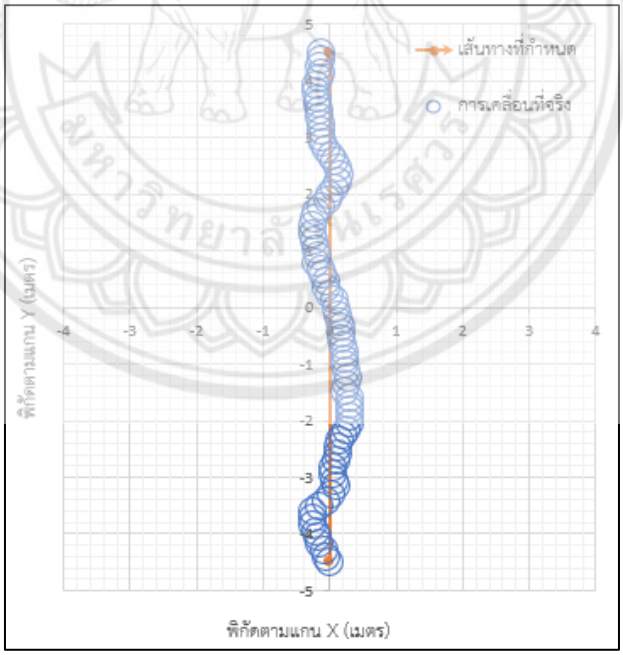
รูปที่ 4.15 ลักษณะบีมของเซนเซอร์อัลทราโซนิกตัวที่ 4

จากผลการทดลองหาขนาดของลำคลื่นของเซนเซอร์อัลทราโซนิกทั้ง 4 ตัว พบว่า ขนาดของลำคลื่นของเซนเซอร์อัลทราโซนิกแต่ละตัวมีลักษณะแตกต่างกันเล็กน้อยค่าที่วัดได้จะไม่แน่นอน แต่ทั้งหมดล้วนมีลักษณะคล้ายวงรี นอกจากนั้นยังพบว่าบางตำแหน่งโดยเฉพาะที่ระยะ 3.5 เมตรจะหวังผลลัพธ์ที่แน่นอนไม่ได้ โดยปัจจัยที่ส่งผลต่อการทดสอบระยะ คือ ชนิดและระนาบพื้นผิวของวัสดุที่ใช้ทดสอบ ส่งผลให้ระยะที่เซนเซอร์อัลทราโซนิกวัดได้เกิดความคลาดเคลื่อน

4.3 การทดสอบการเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนด

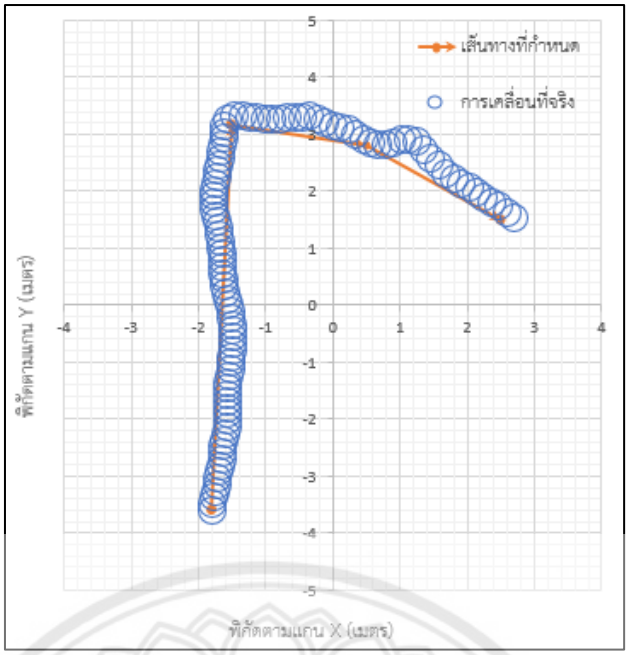
การทดสอบการเคลื่อนที่ของหุ่นยนต์ดำเนินการโดยกำหนดเส้นทางการเคลื่อนที่ของหุ่นยนต์ขึ้นมาทั้งหมด 4 รูปแบบ โดยกำหนดพิกัดฉาก (x,y) ลงบนพื้นที่ที่ทำการทดลอง แล้วจึงกำหนดจุดผ่านของการเคลื่อนที่ตามที่ต้องการลงบนพื้นที่ทดลองซึ่งการทดสอบทำบนพื้นกระเบื้องยางที่มีลักษณะผิวค่อนข้างมัน ส่งผลให้อาจเกิดการลื่นไถลของล้อหุ่นยนต์ได้ขณะดำเนินการทดลอง จากนั้นให้ผู้สั่งงานบังคับให้หุ่นยนต์เคลื่อนที่จากจุดหนึ่งไปยังจุดถัดไปตามที่ได้กำหนดไว้ เมื่อเสร็จสิ้นการเคลื่อนที่จึงบันทึกพิกัดตามที่หุ่นยนต์เคลื่อนที่จริงทำการทดลอง 2 ครั้ง จากนั้นนำค่าเฉลี่ยพิกัดที่บันทึกได้ทั้งหมดมาพล็อตแสดงเป็นเส้นทางการเคลื่อนที่ ดังแสดงในรูปที่ 4.16 – 4.19

เส้นทางที่ 1 เริ่มต้นโดยการกำหนดให้หุ่นยนต์เคลื่อนที่จากพิกัด (0,-4.5) ไปยังจุดหมายปลายทางที่พิกัด (0,4.5) โดยหุ่นยนต์สามารถเคลื่อนที่ไปยังจุดหมายปลายทางได้สำเร็จอย่างไรก็ตามเส้นทางการเคลื่อนที่อาจไม่เรียบมีการส่ายไปมาทั้งนี้เนื่องจากลักษณะของพื้นที่ทำการทดลองซึ่งเป็นกระเบื้องยางที่มีผิวค่อนข้างมันและการออกตัวในช่วงต้นของหุ่นยนต์มีการกระตุก



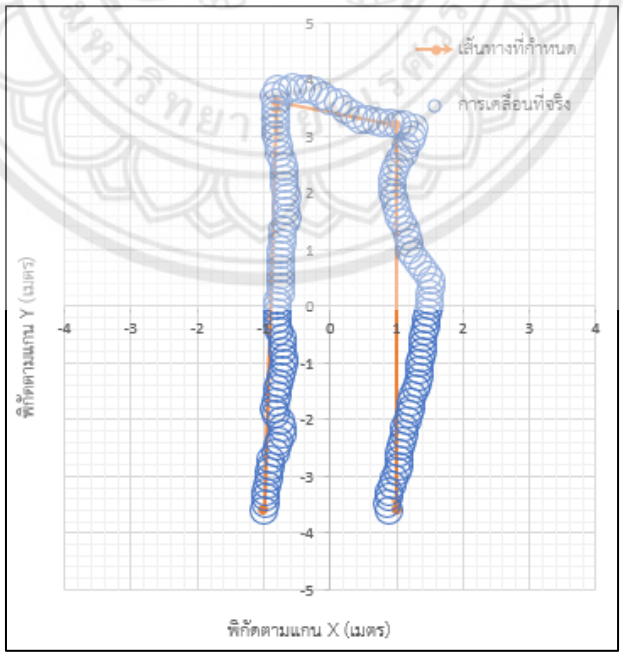
รูปที่ 4.16 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดรูปแบบที่ 1

เส้นทางที่ 2 เริ่มต้นโดยการกำหนดให้หุ่นยนต์เคลื่อนที่จากพิกัด (-1.8,-3.6) ไปยังจุดหมายปลายทางที่พิกัด (2.5,1.5) โดยหุ่นยนต์สามารถเคลื่อนที่ไปยังจุดหมายปลายทางได้สำเร็จอย่างไรก็ตามเส้นทางการเคลื่อนที่มีการส่ายในตอนหมุนตัวเกิดจากสาเหตุเดียวกันกับเส้นทางก่อนหน้า



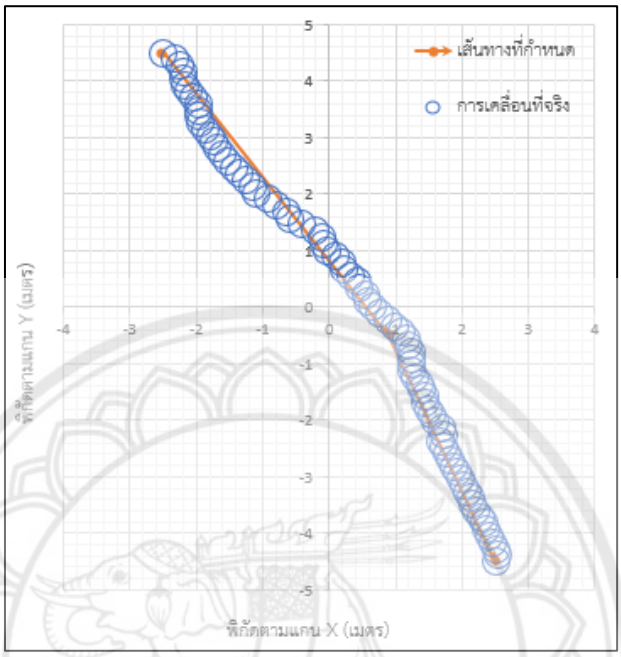
รูปที่ 4.17 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดรูปแบบที่ 2

เส้นทางที่ 3 เริ่มต้นโดยการกำหนดให้หุ่นยนต์เคลื่อนที่จากพิกัด (-1.0,-3.6) ไปยังจุดหมายปลายทางที่พิกัด (1.0,-3.6) โดยหุ่นยนต์สามารถเคลื่อนที่ไปยังจุดหมายปลายทางได้สำเร็จอย่างไรก็ตามเส้นทางการเคลื่อนที่มีการส่ายไปมาในช่วงของการกลับตัวเกิดจากสาเหตุเดียวกันกับเส้นทางที่ 1



รูปที่ 4.18 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดรูปแบบที่ 3

เส้นทางที่ 4 เริ่มต้นโดยการกำหนดให้หุ่นยนต์เคลื่อนที่จากพิกัด (2.5,-4.5) ไปยังจุดหมายปลายทางที่พิกัด (-2.5,4.5) โดยหุ่นยนต์สามารถเคลื่อนที่ไปยังจุดหมายปลายทางได้สำเร็จอย่างไรก็ตามเส้นทางการเคลื่อนที่มีการส่ายในช่วงแรกทั้งนี้เกิดจากสาเหตุเดียวกันกับเส้นทางอื่นๆ



รูปที่ 4.19 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดรูปแบบที่ 4

จากผลการทดลองการเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดทั้ง 4 เส้นทางพบว่า หุ่นยนต์สามารถเคลื่อนที่ตามเส้นทางที่กำหนดได้ โดยจะเห็นได้ว่าหุ่นยนต์มีแนวโน้มในการเคลื่อนที่ดีพอสมควร ความคลาดเคลื่อนในบางช่วงของการเคลื่อนที่นั้น มีปัจจัยทั้งที่เกิดจากผู้สั่งงาน การออกตัวในช่วงต้น สภาพของพื้นที่ลื่นทำให้ล้อเกิดการลื่นไถล และหุ่นยนต์เกิดการเหวี่ยงตัวทำให้เสียสมดุลในขณะเคลื่อนที่

จากผลการทดลองข้างต้นแสดงให้เห็นว่า หุ่นยนต์สามารถเคลื่อนที่ในพื้นราบ มีการเคลื่อนที่ 7 รูปแบบ สั่งการได้ด้วยท่าทาง 10 ท่าทาง รับคำสั่งท่าทางจากบุคคลซึ่งยืนอยู่ด้านหน้าในระยะที่กล้องสามารถจับได้ รวมทั้งสามารถติดตามผู้สั่งงานและตรวจจับสิ่งกีดขวางได้

บทที่ 5
สรุปผลการดำเนินงานและข้อเสนอแนะ

จากการจัดทำโครงงานการพัฒนาหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง คณะผู้จัดทำได้เรียนรู้จากปัญหาที่เกิดขึ้นระหว่างการดำเนินงานและมีการปรับปรุงพัฒนาต่อจนแล้วเสร็จในที่สุด ในบทนี้จะเป็นการสรุปผลการดำเนินงานและให้ข้อเสนอแนะเพื่อเป็นแนวทางในการพัฒนาต่อไป

5.1 สรุปผลการดำเนินงาน

หลังจากดำเนินงานโครงงานการพัฒนาหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทางจนแล้วเสร็จ จึงได้มีการทดสอบการทำงานในส่วนต่างๆ ของหุ่นยนต์ และสามารถสรุปผลการทดลองออกมาได้ดังนี้

5.1.1 ความถูกต้องในการตัดสินท่าทางการสั่งงาน

จากผลการทดลองพบว่า ในกรณีที่มีแสงปกติระบบสามารถตัดสินท่าทางการสั่งงานได้ถูกต้องทั้งหมดซึ่งมีค่าร้อยละของความถูกต้องอยู่ที่ร้อยละ 100 ขณะที่ในกรณีที่มีแสงมากค่าร้อยละของความถูกต้องในการตัดสินท่าทางการสั่งงานจะมีค่าร้อยละ 98.94 ที่ระยะปานกลางและร้อยละ 78.82ที่ระยะไกลซึ่งน้อยกว่าในกรณีที่มีแสงปกติ ดังนั้น จะเห็นว่าปัจจัยหลักที่ส่งผลต่อการตัดสินท่าทางการสั่งงาน คือ แสง เนื่องจากอุปกรณ์ Kinect ใช้เซนเซอร์อินฟราเรดในการตรวจวัดระยะจึงทำงานได้ดีในที่ที่มีแสงปกติ นอกจากปัจจัยเรื่องแสงแล้ว ค่าร้อยละของความถูกต้องยังแสดงให้เห็นว่าระยะทางก็มีผลต่อการตัดสินท่าทางการสั่งงาน ระยะห่างที่มากเกินไปส่งผลต่อความสามารถในการวัดระยะทำให้ไม่สามารถประมวลภาพเป็นลักษณะโครงกระดูกได้ชัดเจนจึงมีโอกาสตัดสินท่าทางผิดพลาดได้มากขึ้น

5.1.2 ระยะที่เซนเซอร์อัลทราโซนิกวัดได้และลำคลื่นของเซนเซอร์อัลทราโซนิก

จากผลการทดลองหาระยะที่เซนเซอร์อัลทราโซนิกสามารถวัดได้ พบว่า ร้อยละของความคลาดเคลื่อนของเซนเซอร์อัลทราโซนิกทั้ง 4 ตัว มีค่าเพิ่มขึ้นมากเมื่อใช้วัดระยะทางที่มากกว่า 3 เมตรขึ้นไปโดยมีค่าร้อยละของความคลาดเคลื่อนอยู่ที่ร้อยละ 24.68 ร้อยละ26.25 ร้อยละ26.06 และร้อยละ 36.46 ดังนั้น สามารถสรุปได้ว่าระยะหวังผลของเซนเซอร์อัลทราโซนิกที่ใช้ คือ 3 เมตร

สำหรับลำคลื่นของเซนเซอร์อัลทราโซนิกทั้ง 4 ตัวนั้น พบว่า ลำคลื่นของเซนเซอร์อัลทราโซนิกแต่ละตัวมีลักษณะแตกต่างกันเล็กน้อย แต่ทั้งหมดล้วนมีลักษณะคล้ายวงรี นอกจากนั้นยังพบว่าบางตำแหน่งโดยเฉพาะที่ระยะ 3.5 เมตรจะหวังผลลัพธ์ที่แน่นอนไม่ได้ ปัจจัยอื่นที่ส่งผลให้ระยะที่เซนเซอร์อัลทราโซนิกวัดได้เกิดความคลาดเคลื่อน คือ ชนิดและระนาบพื้นผิวของวัสดุที่ตรวจพบ

5.1.3 การเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนด

จากผลการทดลองการเคลื่อนที่ของหุ่นยนต์ตามเส้นทางที่กำหนดทั้ง 4 เส้นทางพบว่าหุ่นยนต์สามารถเคลื่อนที่ตามเส้นทางที่กำหนดได้ จะเห็นได้ว่าหุ่นยนต์มีการเคลื่อนที่ได้ดีพอสมควร ความคลาดเคลื่อนในบางช่วงของเส้นทางมีปัจจัยทั้งที่เกิดจากผู้สั่งงาน การออกตัวในช่วงต้น สภาพของพื้นที่ลื่นทำให้ล้อเกิดการลื่นไถล รวมทั้งหุ่นยนต์เกิดการเหวี่ยงตัวทำให้เสียสมดุลในขณะเคลื่อนที่

การทำโครงงานนี้หุ่นยนต์สามารถเคลื่อนที่ได้รอบทิศทางตามการสั่งงานจากท่าทางที่กำหนดและหยุดเคลื่อนที่เมื่อตรวจพบสิ่งกีดขวางหรือสั่งให้หยุด ซึ่งทั้งหมดนี้เป็นไปตามวัตถุประสงค์ที่ตั้งไว้

5.2 ปัญหาและอุปสรรค

5.2.1 ต้องเคลื่อนย้ายหุ่นยนต์และเปลี่ยนสถานที่ในการทำงานบ่อยครั้งเนื่องจากการทำหุ่นยนต์ต้องมีการบูรณาการหลายส่วนเข้าด้วยกันและต้องการพื้นที่สำหรับทำการทดสอบการเคลื่อนที่ของหุ่นยนต์

5.2.2 ระบบปฏิบัติการวินโดวส์ (Windows) มีการอัพเดทบ่อยครั้งส่งผลให้การทำงานล่าช้าและข้อมูลบางส่วนสูญหายเมื่ออัพเดทไม่สมบูรณ์

5.2.3 เมื่อหุ่นยนต์เคลื่อนที่เซนเซอร์ Kinect เกิดการสั่นส่งผลให้การตรวจจับ (Track) โครงกระดูกหลุด จึงไม่สามารถสั่งงานหุ่นยนต์ต่อได้

5.3 แนวทางในการพัฒนา

5.3.1 การรับมือกับปัญหา Kinect สั่นโดยอาจแก้ไขด้าน Software คือ การเขียนคำสั่ง ปิดการใช้งาน Accelerometer ภายในตัว Kinect หรือแก้ไขด้าน Hardware คือ เปลี่ยนไปใช้อุปกรณ์ Kinect รุ่นที่ไม่มี Accelerometer

5.3.2 จากการทำงานที่มีประสิทธิภาพของหุ่นยนต์ที่พัฒนาขึ้น หุ่นยนต์มีการเคลื่อนที่ตามการสั่งการด้วยท่าทางซึ่งถูกปกป้องด้วยระบบจากเซนเซอร์อัลทราโซนิกจึงควรมีการพัฒนาต่อให้หุ่นยนต์สามารถทำงานได้เองโดยอัตโนมัติ เพื่อนำไปประยุกต์ใช้ในงานบริการด้านต่างๆ เช่น งานค้นหา นำเที่ยว หรืองานต้อนรับ เป็นต้น

เอกสารอ้างอิง

โรงเรียนปักธงชัยประชานิรมิตร. **ประวัติของหุ่นยนต์.** สืบค้นเมื่อ 22 มีนาคม 2561, จาก https://sites.google.com/site/robotprograming/prawati-khxng-hun-ynt

Karel Čapek. **Rossum's Universal Robots (R.U.R.).** สืบค้นเมื่อ 22 มีนาคม 2561, จาก https://en.wikipedia.org/wiki/R.U.R.

Purinatth. **ส่วนประกอบของหุ่นยนต์ (Robot).** สืบค้นเมื่อ 22 มีนาคม 2561, จาก http://engineerknowledge.blogspot.com/2011/09/robot.html

Project Lab. **Mecanum & Omni wheel.** สืบค้นเมื่อ 22 มีนาคม 2561, จาก http://projectlab.co.th/2017/06/04/mecanum-omni-wheel/

Thanawat Raibroycharoen. **เจาะลึกโครงสร้างของ Kinect สำหรับ Windows.** สืบค้นเมื่อ 22 มีนาคม 2561, จาก https://kinectasia.wordpress.com/tag/nui/

Computer Engineering Chiang Mai University. **เทคนิค Pulse Width Modulation (PWM).** สืบค้นเมื่อ 22 มีนาคม 2561 จาก https://learninginventions.org/?page_id=2371

สุชิน เสือช้อย. **อุปกรณ์ป้อนกลับ (Feedback Device).** สืบค้นเมื่อ 22 มีนาคม 2561, จาก www.9engineer.com/index.php?m=article&a=print&article_id=2176

Chaiyaporn Silawatchananai. **การควบคุมมอเตอร์ไฟฟ้ากระแสตรง.** สืบค้นเมื่อ 22 มีนาคม 2561, จาก http://aimagin.com/blog/motor/?lang=th

Thaieasyelec. **บทความ Arduino คืออะไร ตอนที่2 มาทำความรู้จักกับ Arduino รุ่นต่างๆกัน**. สืบค้นเมื่อ 22 มีนาคม 2561, จาก https://www.thaieasyelec.com/article-wiki/basic-electronics/บทความเริ่มต้นใช้งาน-arduino-เลือกใช้บอร์ด-arduino-ตามความเหมาะสมกับการใช้งาน.html

Supremelines. **อัลตราโซนิคเซนเซอร์ (ULTRASONIC SENSORS).** สืบค้นเมื่อ 22 มีนาคม 2561, จาก http://www.supremelines.co.th/สาระน่ารู้/2090-อัลตราโซนิค-เซนเซอร์-ultrasonic-sensors.html

Stem Education. **หุ่นยนต์บังคับไร้สายใช้ Arduino ควบคุมด้วย PS2 Wireless Controller.** สืบค้นเมื่อ 22 มีนาคม 2561, จาก http://www.se-edstemeducation.com/หุ่นยนต์บังคับไร้สายใช้-arduino-ควบคุมด้วย-ps2-wireless-controller/

ภาคผนวก ก
รายละเอียดของอุปกรณ์

บอร์ดไมโครคอนโทรลเลอร์ “Arduino Due” มีรายละเอียดดังต่อไปนี้

Microcontroller	AT91SAM3X8E
Operating Voltage	3.3 V
Recommended Input Voltage	7-12 V
Min-Max Input Voltage	6-20 V
Digital I/O Pins	54 of which 12 provide PWM output
Analog Input Pins	12
Analog Outputs Pins	2
Total DC Output Current on all I/O lines	130 mA
DC Current for 3.3V Pin	800 mA
DC Current for 5V Pin	800 mA
Flash Memory	512 KB for the user applications
SRAM	96 KB (two banks: 64KB and 32KB)
Clock Speed	84 MHz

บอร์ดไมโครคอนโทรลเลอร์ “Arduino Uno R3” มีรายละเอียดดังต่อไปนี้

Microcontroller	ATmega328P
Operating Voltage	5 V
Recommended Input Voltage	7-12 V
Min-Max Input Voltage	6-20 V
Digital I/O Pins	14 of which 6 provide PWM output
Analog Input Pins	6
Total DC Output Current on all I/O lines	130 mA
DC Current for 3.3V Pin	50 mA
DC Current per I/O Pin	20 mA
Flash Memory	32 KB which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz
LED_BUILTIN	13

เซนเซอร์อัลทราโซนิก “SRF05 - Ultra - Sonic” มีรายละเอียดดังต่อไปนี้

Voltage	5V
Current	30mA ~ 50mA (Max)
Frequency	40 kHz
Max Range	3 m
Min Range	3 cm
Sensitivity	Detect at 3 cm
Input Trigger	10 us (Min) TTL level pulse
Echo Pulse	Positive TTL level signal, width proportional to range.
Size	43mm x 20mm x 17mm (height)

บอร์ดไดร์เวอร์มอเตอร์ “DC Motor H-Bridge Driver (SE-HB-100)” มีรายละเอียดดังต่อไปนี้

Output : Single motor driver

- Motor DC Supply 12-36 V / 80 A (max)
- All N-Channel Power MOSFET H-Bridge Driver With ultra-fast reverse recovery inverse parallel protection diodes

Input :

- Full Opto-isolated ground-loop input interface signals
- Control Voltage input : 3-5 V / 8 mA (min)

Drive Mode : independently with :

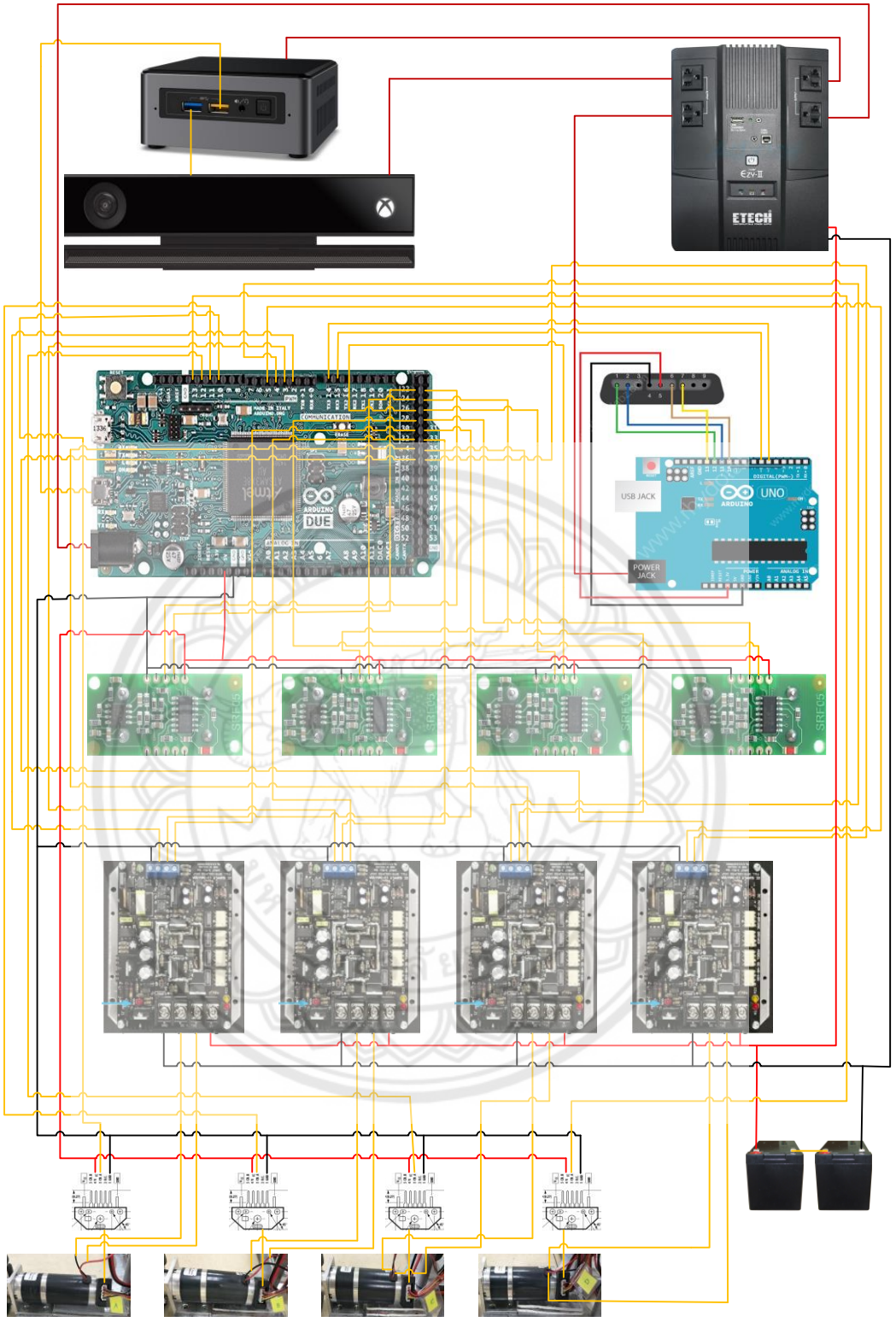
- Start – Stop Control
- Direction Control
- Speed Control (PWM Driver)

PWM Duty cycle Range : 0 - 100 %

PWM Frequency : 400 Hz – 25 kHz

Board built-in Transient voltage protection up to 100 V circuit

ภาคผนวก ข
การเชื่อมต่ออุปกรณ์ทั้งหมด


ETECH
ARDUINO
DUE
USB JACK
POWER
JACK
UNO
ARDUINO
SRF05

ภาคผนวก ค
โปรแกรมการทำงานของหุ่นยนต์เคลื่อนที่สั่งงานด้วยท่าทาง

โปรแกรมซึ่งใช้กับการควบคุมระดับบน (รันบน PC)

ดาวน์โหลดได้ที่ https://drive.google.com/file/d/1H-AQZTVjXTgYyplXMrnKqJwWFeZdNVyO/view?usp=sharing

โปรแกรมซึ่งใช้กับการควบคุมระดับล่าง (รันบนบอร์ด Arduino Due)

ดาวน์โหลดได้ที่ https://drive.google.com/file/d/1m9WXoVK1FL0npzUxnCfmnRoj7AWxnJol/view?usp=sharing

โปรแกรมซึ่งใช้สื่อสารกับโมดูล Joystick (รันบนบอร์ด Arduino Uno)

ดาวน์โหลดได้ที่ https://drive.google.com/open?id=1J2azAF2DuVeyUZ8J7LiY2uAKMxRfjhHj