

การควบคุมการเดินของหุ่นยนต์แมลงหกขาโดยใช้ระบบสมองกลฝังตัวแบบเอสโอพีซี

โดย

นางสาวศิริินภา เสงส์ขวอน 630910489

นายรัชชัย อางปึกษา 630910658

นายญาณภัทร เล็กทิมทอง 630910687

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์และระบบคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม

มหาวิทยาลัยศิลปากร

ปีการศึกษา 2566

การควบคุมการเดินของหุ่นยนต์แมลงหกขาโดยใช้ระบบสมองกลฝังตัวแบบเอสโอพีซี

โดย

นางสาวศิริินภา เสงส์ขวอน 630910489

นายรัชชัย อางปึกษา 630910658

นายญาณภัทร เล็กทิมทอง 630910687

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์และระบบคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม

มหาวิทยาลัยศิลปากร

ปีการศึกษา 2566

Locomotion Control of a Hexapod Robot

Using SoPC-based Embedded System

By

Sirinapha Hengsangvon

Thawatchai Ardpaksa

Yanaphat lektimthong

Project Submitted in Partial Fulfillment of the Requirements for the

Degree of Bachelor of Engineering

(Electronic and Computer System Engineering)

Department of Electrical Engineering

Faculty of Engineering and Industrial Technology

Silpakorn University

2023

การควบคุมการเคลื่อนที่ของหุ่นยนต์หกขาโดยใช้ระบบสมองกลฝังตัวแบบ SoPC

นางสาวศิริรภา

เฮงสังข์วอน 630910489

นายรัชชัย

อาจปึกษา 630910658

นายญาณภัทร

เล็กทิมทอง 630910687

ปริญญานิพนธ์นี้ได้รับอนุมัติเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์และระบบคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยศิลปากร

ปีการศึกษา 2566

คณะกรรมการสอบปริญญานิพนธ์

..... ประธานกรรมการ

(ผศ.ดร.จิรภัฏฐ์ เหมือนชู)

..... กรรมการ

(รศ.ดร.ชูเกียรติ สอดศรี)

..... กรรมการและอาจารย์ที่ปรึกษาปริญญานิพนธ์

(ผศ.ดร.ยุทธนา เจวจินดา)

ลิขสิทธิ์ของภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม

มหาวิทยาลัยศิลปากร

หัวข้อปริญญานิพนธ์	การควบคุมการเคลื่อนที่ของหุ่นยนต์หกขาโดยใช้ระบบสมองกลฝังตัวแบบ SoPC	
โดย	นางสาวศิริินภา	เฮงสังข์วอน 630910489
	นายธวัชชัย	อาจปึกษา 630910658
	นายญาณภัทร	เล็กทิมทอง 630910687
อาจารย์ที่ปรึกษา	อาจารย์ ดร.ยุทธนา	เจวจินดา
ชื่อปริญญา	วิศวกรรมศาสตรบัณฑิต (วิศวกรรมอิเล็กทรอนิกส์และระบบคอมพิวเตอร์)	

บทคัดย่อ

หุ่นยนต์หกขา คือ หุ่นยนต์ที่มีการออกแบบให้มีลักษณะคล้ายแมลงหกขา ทำทางการเคลื่อนที่ของหุ่นยนต์หกขาจึงเลียนแบบการเคลื่อนที่ของแมลงหกขา ซึ่งสามารถมีรูปแบบการเคลื่อนที่ที่หลากหลายมาก การออกแบบทำทางการเคลื่อนที่ของหุ่นยนต์หกขา และการควบคุมการเคลื่อนที่ของหุ่นยนต์หกขา จึงเป็นปัญหาสำคัญทางด้าน หุ่นยนต์เคลื่อนที่ที่ใช้ขา ดังนั้นโครงงานนี้นำเสนอการออกแบบทำเดินพื้นฐานของหุ่นยนต์หกขา การพัฒนาระบบควบคุมแบบฝังตัวแบบกระจายพัฒนาซอฟต์แวร์ฝังตัวสำหรับควบคุมการเคลื่อนที่ให้เป็นไปตามทำเดินที่ได้ออกแบบไว้ ซึ่งทำเดินพื้นฐานของหุ่นยนต์หกขาจะมีทำเดินแบบคลีน ทำเดินแบบกระเพื่อม และทำเดินแบบก้าวสามขา โดยแต่ละทำเดินสามารถกำหนดทิศทางการเคลื่อนที่จะมีการเดินไปข้างหน้า เดินถอยหลัง เดินไปทางซ้าย เดิน ไปทางขวา และเดินแนวทะแยงอีก 4 ทิศ อีกทั้งยังสามารถปรับความเร็ว และความสูงในการก้าวขาได้ รวมถึงการเก็บข้อมูลจากเซนเซอร์ต่าง ๆ ที่วัดได้ขณะเดิน เพื่อนำไปประกอบในการพัฒนาทำทางอื่น ๆ ต่อไป จากการทดลองพบว่าหุ่นยนต์สามารถเดินได้ตามรูปแบบทำเดินที่ออกแบบ และสามารถอ่านค่าจากเซนเซอร์ได้โดยมีความคลาดเคลื่อนเล็กน้อย

คำสำคัญ : หุ่นยนต์หกขา

Title	Locomotion Control of a Hexapod Robot Using SoPC-based Embedded System		
By	Sirinapha	Hengsangvon	630910489
	Thawatchai	Ardpaksa	630910658
	Yanaphat	lektimthong	630910687
Advisor	Lect. Dr. Yutana	Jewajinda	
Degree	Bachelor of Engineering (Electronic and Computer System Engineering)		

Abstract

Hexapod robot is a kind of mobile robots that imitates insect with six legs because of that there are so many locomotion patterns of the robots. The design and control of locomotion patterns are crucial to the hexapod robot and are also foundation for other leg-type mobile robots. Therefore, this project presents the design of locomotion patterns, an embedded system for distributed control, and embedded software for controlling movement according to design locomotion patterns. The basic locomotion patterns of the hexapod are wave, Ripple and Tripod Gaits for which each type is capable of moving forward, backward, left, right, and oblique directions. Moreover, the control system can adjust speed and level of each gait pattern. Also, collecting sensor data while the hexapod is moving in order to use to develop other locomotion patterns. From experiment results, the robot can imitate the movement similar to the designed pattern of locomotion and collect sensor data with small error margin.

Keywords: Hexapod robot

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จได้ด้วยความช่วยเหลืออย่างดียิ่งจากอาจารย์ ดร.ยุทธนา เจวจินดา อาจารย์ที่ปรึกษา ซึ่งได้สละเวลาให้คำแนะนำคอยให้ความรู้เสนอแนะข้อคิดเห็น และแนวทางในการทำปริญญานิพนธ์ให้สำเร็จตามวัตถุประสงค์ที่ได้ตั้งไว้

ขอขอบคุณคณาจารย์ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยศิลปากร วิทยาเขตพระราชวังสนามจันทร์ ผู้ประสิทธิประสาทวิชาความรู้ อันมีค่ายิ่ง และคำแนะนำต่าง ๆ แก่ผู้วิจัย และบุคลากรประจำภาควิชาวิศวกรรมไฟฟ้าทุกท่านที่คอยช่วยเหลือผู้วิจัยมาโดยตลอด

กราบขอบพระคุณ บิดา มารดา และผู้มีพระคุณทุกท่าน ที่ให้การส่งเสริมสนับสนุนการศึกษา แก่ผู้วิจัยอย่างดียิ่ง และขอบคุณพี่ ๆ เพื่อน ๆ น้อง ๆ ทุกคน ที่ให้คำปรึกษาในการทำงานเอกสาร งานวิจัย และความช่วยเหลือด้านต่าง ๆ เป็นอย่างดียิ่งตลอดมา

ท้ายสุดนี้ผู้วิจัยหวังเป็นอย่างยิ่งว่างานวิจัยนี้จะเป็นประโยชน์ต่อผู้ที่สนใจหรือผู้ที่เกี่ยวข้องทั่วไปและหากงานวิจัยนี้มีข้อผิดพลาดประการใด ทางผู้วิจัยต้องขออภัยมา ณ ที่นี้

ผู้จัดทำ

นางสาวศิริินภา	เฮงสังข์วอน
นายธวัชชัย	อาจปักษา
นายญาณภัทร	เล็กทิมทอง

สารบัญ

หน้า

บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ช
สารบัญรูป	ณ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ขอบเขตของโครงการ	2
1.4 แผนการดำเนินงาน	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	4
ทฤษฎีพื้นฐานที่เกี่ยวข้อง	4
2.1.1 ทฤษฎีจลนศาสตร์ของหุ่นยนต์	4
2.1.2 จลนศาสตร์ของหุ่นยนต์หกขา	4
2.1.3 ลักษณะการเดินของหุ่นยนต์หกขา	8
2.2 การควบคุมมอเตอร์ Dynamixel AX-12A โดยใช้บอร์ด OpenCM 9.04	15
2.2.1 มอเตอร์ Dynamixel AX-12A	16
2.2.2 บอร์ดพัฒนา OpenCM9.04	18
2.2.3 OpenCM 485 Expansion Board	20
2.2.4 ชุดพัฒนาสมองกลฝังตัวแบบ SoPC	22
2.2.5 Accelerometers & Gyroscope GY-521	26
บทที่ 3 ภาพรวมของโครงการ	27
3.1 ภาพรวมของโครงการ	27
3.2 ระบบควบคุมแบบกระจายสำหรับหุ่นยนต์หกขา	28

สารบัญ (ต่อ)

หน้า

3.2	สิ่งที่ดำเนินการไปแล้ว.....	29
-----	-----------------------------	----

สารบัญรูป

หน้า

รูปที่ 2.1 กรอบอ้างอิงของขาหนึ่งข้าง	4
รูปที่ 2.2 แบบจำลองของขาหนึ่งข้าง	5
รูปที่ 2.3 ตำแหน่งของขาในระนาบ X-Y	5
รูปที่ 2.4 ตำแหน่งของขาในระนาบ Y-Z	6
รูปที่ 2.5 มุม และด้านของสามเหลี่ยมใด ๆ	6
รูปที่ 2.6 การวางตำแหน่งขากับตัวหุ่นยนต์	8
รูปที่ 2.7 ลำดับการก้าวขาของท่าเดินแบบคลีนเทียบกับเวลา	9
รูปที่ 2.8 ลำดับการก้าวขาของท่าเดินแบบกระเพื่อมเทียบกับเวลา	11
รูปที่ 2.9 ลำดับการก้าวขาของท่าเดินแบบ Tripod เทียบกับเวลา	13
รูปที่ 2.10 มอเตอร์ Dynamixel AX-12A	15
รูปที่ 2.11 ข้อมูลจำเพาะของ Dynamixel AX-12A	15
รูปที่ 2.12 การส่ง และรับข้อมูลระหว่าง Main controller และมอเตอร์ Dynamixel	17
รูปที่ 2.13 การส่ง และรับข้อมูลระหว่าง Main controller กับมอเตอร์ Dynamixel หลายตัว ..	17
รูปที่ 2.14 บอร์ดพัฒนา OpenCM 9.04	18
รูปที่ 2.15 OpenCM 485 EXP	20
รูปที่ 2.16 The Zybo Z7	22
รูปที่ 2.17 Zynq AP SoC architecture	24
รูปที่ 2.18 ZYBO Z7 development board	24
รูปที่ 2.19 Accelerometers & Gyroscope GY-521	26
รูปที่ 2.20 โครงสร้างของ Accelerometers & Gyroscope GY-521	26
รูปที่ 3.1 ภาพรวมการทำงาน.....	27
รูปที่ 3.2 สถาปัตยกรรมของระบบ.....	28
รูปที่ 3.3 รูปสิ่งที่ดำเนินการไปแล้ว	29
รูปที่ 3.4 รูปสิ่งที่ดำเนินการไปแล้ว	30

สารบัญรูป (ต่อ)

หน้า

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

ในปัจจุบันหุ่นยนต์ได้เข้ามามีบทบาทกับชีวิตประจำวันของมนุษย์มากขึ้น เป็นเทคโนโลยีที่ได้รับการพัฒนาอย่างต่อเนื่อง ทำให้ความสามารถของหุ่นยนต์ถูกพัฒนาขึ้นอย่างรวดเร็ว สามารถทำงานต่าง ๆ ที่มนุษย์ไม่สามารถทำได้เป็นจำนวนมาก ส่งผลให้มีการนำหุ่นยนต์เข้ามาใช้งานในด้านต่าง ๆ มากมายไม่ว่าจะเป็น ด้านอุตสาหกรรม ด้านการขนส่ง และใช้งานในครัวเรือน เป็นต้น โดยการสร้างหุ่นยนต์นั้นจะสร้างตามลักษณะที่ต้องการใช้งาน หุ่นยนต์หลากหลายประเภทมีลักษณะแตกต่างกันไป โดยหุ่นยนต์แต่ละประเภทจะมีโครงสร้าง และการเคลื่อนไหวที่ถูกออกแบบไว้ ซึ่งมีระบบสมองกลคอยควบคุมการทำงานของตัวขับเคลื่อนเช่น มอเตอร์ เป็นต้น โครงงานนี้มุ่งเน้นความสนใจปัญหาการควบคุมการเคลื่อนที่ของหุ่นยนต์ที่ใช้ขา ซึ่งหุ่นยนต์หกขานับเป็นพื้นฐานของหุ่นยนต์ที่ใช้ขาในการเคลื่อนที่ ดังนั้นโครงงานนี้จึงศึกษาการควบคุม และปัญหาการเคลื่อนที่ของหุ่นยนต์หกขา

หุ่นยนต์ได้มีการพัฒนาระบบการเคลื่อนที่ให้มีได้หลากหลายรูปแบบเพื่อเลียนแบบสิ่งมีชีวิตเช่น การเคลื่อนที่แบบใช้ขา แบบเลื้อย แบบกลิ้ง แบบกระโดด ซึ่งรูปแบบการเคลื่อนที่ของแต่ละแบบจะมีข้อดี และข้อเสียแตกต่างกันออกไป โดยการเคลื่อนที่ที่นิยมใช้กันมากที่สุดคือ การใช้ล้อเป็นหลักซึ่งสามารถทำได้โดยง่าย สะดวก และเคลื่อนที่ได้เร็ว แต่มีข้อจำกัดในเรื่องของความสามารถในการเคลื่อนที่ซึ่งเป็นอุปสรรคต่อการเคลื่อนที่ไปในบริเวณที่ยากต่อการเข้าถึง ซึ่งหุ่นยนต์หกขาสามารถเคลื่อนที่ไปในพื้นที่ที่มีลักษณะพื้นผิวที่ขรุขระไม่ราบเรียบหรือในบริเวณที่เข้าถึงยากได้ ดังนั้นการจัดทำโครงงานนี้จึงเป็นการพัฒนารูปแบบการเคลื่อนที่โดยใช้ขาเพื่อให้เห็นถึงรูปแบบในการเดินในท่าทางต่าง ๆ ที่สามารถลอกเลียนแบบได้จากแมลง เช่น มด ตัวงู และแมงมุม เป็นต้น

ดังนั้นโครงงานนี้จึงเน้นการพัฒนาการควบคุมการเคลื่อนที่ของหุ่นยนต์หกขาให้เคลื่อนที่ตามรูปแบบการเคลื่อนที่ โดยใช้ระบบควบคุมฝังตัวแบบ SoPC และซอฟต์แวร์ฝังตัวสำหรับกำหนดรูปแบบการเคลื่อนที่ของหุ่นยนต์หกขา

1.2 วัตถุประสงค์ของโครงการ





- 1.2.1 เพื่อศึกษาการเคลื่อนที่ของหุ่นยนต์ที่มีลักษณะคล้ายการเคลื่อนที่ของแมลงหทขา
- 1.2.2 เพื่อศึกษาการใช้งานระบบสมองกลฝังตัวแบบ SoPC
- 1.2.3 เพื่อออกแบบ และสร้างหุ่นยนต์เคลื่อนที่หทขา
- 1.2.4 ศึกษาการออกแบบ และสร้างระบบควบคุมฝังตัวแบบ SoPC สำหรับควบคุมมอเตอร์เพื่อขับเคลื่อนหุ่นยนต์
- 1.2.5 พัฒนาซอฟต์แวร์ฝังตัวสำหรับกำหนดรูปแบบการเคลื่อนที่ของหุ่นยนต์

1.3 ขอบเขตของโครงการ

- 1.3.1 สร้างแบบจำลองของหุ่นยนต์เคลื่อนที่หทขาในคอมพิวเตอร์
- 1.3.2 ออกแบบ และสร้างหุ่นยนต์เคลื่อนที่หทขา
- 1.3.3 พัฒนาระบบสมองกลฝังตัวแบบ SoPC สำหรับควบคุมการเคลื่อนที่ของหุ่นยนต์หทขา
- 1.3.4 พัฒนาซอฟต์แวร์ฝังตัวเพื่อกำหนดรูปแบบการเคลื่อนที่ และการตัดสินใจรูปแบบการเคลื่อนที่ให้สามารถเคลื่อนที่ด้วยท่าเดินที่มีความยืดหยุ่น

1.4 แผนการดำเนินงาน

ตารางที่ 1.1 แผนการดำเนินงาน

ขั้นตอนการดำเนินงาน	ปี พ.ศ. 2559					ปี พ.ศ. 2560					
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.
1. ศึกษาข้อมูลเกี่ยวกับทฤษฎีของหุ่นยนต์หทขา											
2. ศึกษาออกแบบลักษณะโครงสร้าง และเงื่อนไขการทำงานของหุ่นยนต์											
3. ศึกษาวิธีการควบคุมมอเตอร์											
4. ทำการสร้างแบบจำลองของหุ่น											

5. ศึกษาออกแบบทำเดินใน pybullet และทดลองใช้กับหุ่น จริง											
7. ศึกษาการพัฒนาโดยใช้ชุด พัฒนา SoPC											
8. ออกแบบระบบควบคุมโดย ใช้ชุดพัฒนา SoPC ในการ ควบคุมหุ่นยนต์											
9. ทำการทดลองทำเดิน วิเคราะห์ผล รวบรวมข้อมูลและ จัดทำรายงาน											

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1.5.1 ได้ความรู้ด้านการพัฒนาหุ่นยนต์หกขา
- 1.5.2 เป็นแนวทางในการพัฒนาหุ่นยนต์หกขาต่อไปในอนาคต
- 1.5.3 ได้นำความรู้จากการศึกษาในรายวิชาที่เรียนมาประยุกต์ใช้งานจริง

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีพื้นฐานที่เกี่ยวข้อง

2.1.1 ทฤษฎีจลนศาสตร์ของหุ่นยนต์หกลขา

จลนศาสตร์หรือที่เรียกว่า Kinematics เป็นการศึกษาการเคลื่อนที่ที่เกี่ยวข้องกับตำแหน่ง ความเร็ว และความเร่งของวัตถุ จลนศาสตร์สำหรับหุ่นยนต์นั้นมีการพิจารณาอยู่ 2 แบบด้วยกัน

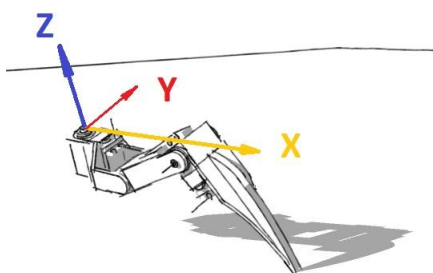
1) Forward Kinematics

คือการวิเคราะห์การเคลื่อนที่ และทิศทางของตัวทำงานส่วนปลาย เมื่อทราบค่ามุมของข้อต่อของหุ่นยนต์ ซึ่งเป็นการคำนวณตามลำดับจากข้อต่อไปยังส่วนปลาย

2) Inverse Kinematics

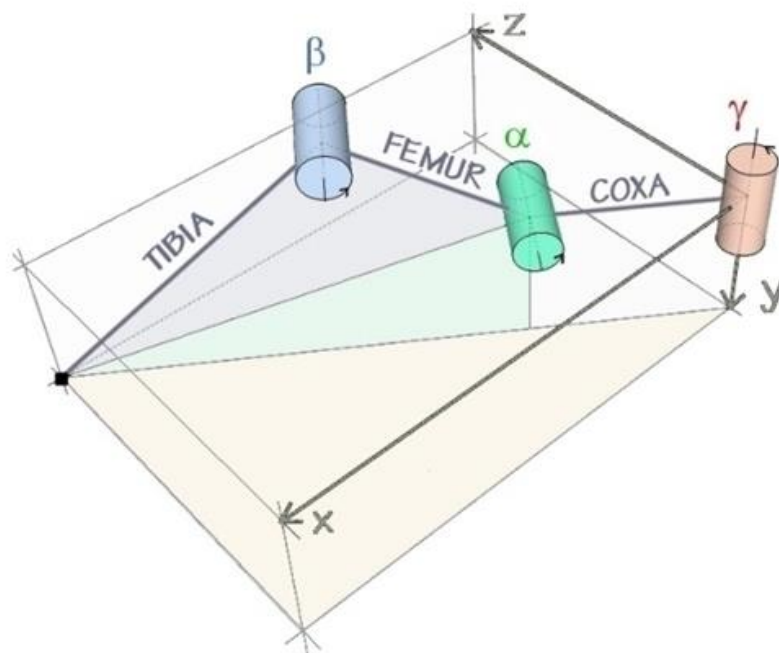
คือการวิเคราะห์หามุมของข้อต่อต่าง ๆ ของหุ่นยนต์ เมื่อรู้ค่าตำแหน่งและทิศทางของตัวทำงานส่วนปลาย (ตำแหน่งเป้าหมาย) ซึ่งเป็นการคำนวณย้อนกลับจากตัวทำงานส่วนปลายมายังข้อต่อต่าง ๆ

ในการควบคุมการเคลื่อนที่ของหุ่นยนต์หกลขาจะเน้นไปที่การควบคุมการเคลื่อนไหวกว้างของขาเป็นหลัก โดยที่แต่ละขาจะมีลักษณะที่เหมือน ๆ กัน ดังนั้นในการวิเคราะห์ จะวิเคราะห์จากขา 1 ข้าง ดังนี้



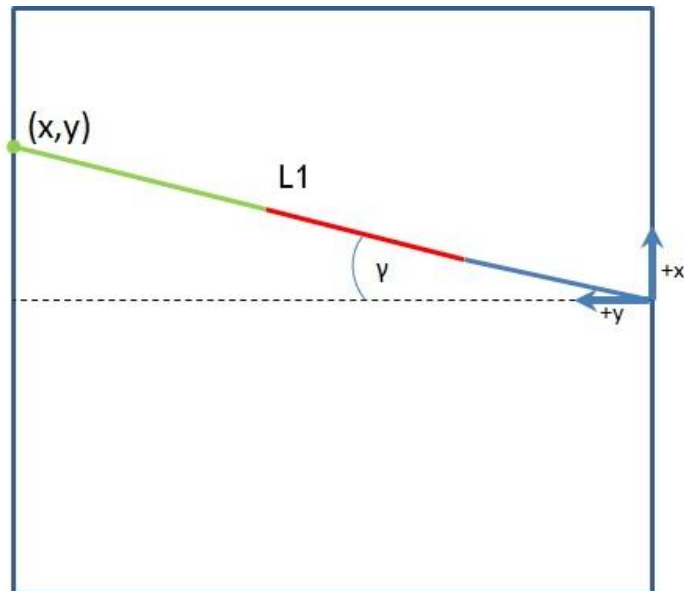
รูปที่ 2.1 กรอบอ้างอิงของขาหนึ่งข้าง [1]

กำหนดให้จุดอ้างอิงอยู่ที่ต้นขา และมีจุดปลายขาอยู่ที่พิกัด (X Y Z) โดยขาจะประกอบไปด้วยท่อนขา (Link) 3 ท่อน และข้อต่อ (Joint) 3 ข้อต่อดังรูป



รูปที่ 2.2 แบบจำลองของขาหนึ่งข้าง [2]

ซึ่งปัญหาที่ต้องการแก้คือ การหาค่ามุมของข้อต่อทั้งสาม ได้แก่ α β γ โดยในการแก้ปัญหจะเป็นการมองในรูปแบบ 2 มิติดังนี้

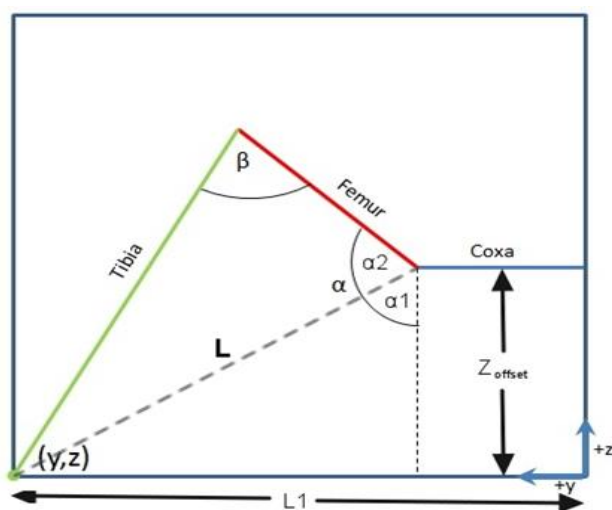


รูปที่ 2.3 ตำแหน่งของขาในระนาบ X-Y [1]

จากรูปทำให้ได้สมการสำหรับการหาค่ามุม γ ดังนี้

$$\frac{x}{y} = \tan(\gamma) \quad (2.1)$$

$$\gamma = \tan^{-1}\left(\frac{x}{y}\right) \quad (2.2)$$



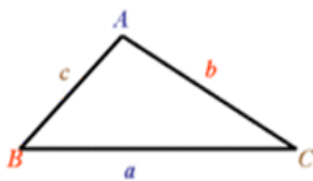
รูปที่ 2.4 ตำแหน่งของขาในระนาบ Y-Z [1]

จากรูปทำให้ได้สมการสำหรับการหาค่ามุม ทำให้ได้สมการสำหรับการหาค่ามุม α_1 α_2 และ β

โดยสามารถหาค่า α_1 ได้จากการหาความยาว L ดังนี้

$$\alpha_1 = \cos^{-1} \left(\frac{Z_{offset}}{L} \right) \quad (2.3)$$

ในส่วนของมุม α_2 กับ β สามารถใช้กฎของ Cosine เข้ามาช่วยในการคำนวณหามุมนั้น ๆ ได้ โดยกฎของ Cosine มีดังนี้



รูปที่ 2.5 มุมและด้านของสามเหลี่ยมใด ๆ [1]

$$a^2 = b^2 + c^2 - 2bc \cos A \quad (2.4)$$

$$b^2 = a^2 + c^2 - 2ac \cos B \quad (2.5)$$

$$c^2 = b^2 + a^2 - 2ab \cos C \quad (2.6)$$

จากสมการที่ (2.4) ถึง (2.6) เมื่อทราบค่าความยาวด้านทั้งสามของรูปสามเหลี่ยมใด ๆ แล้ว ก็จะสามารถหามุมภายในทั้ง 3 มุมได้ดังนี้

$$\cos A = \frac{b^2 + c^2 - a^2}{2bc} \quad (2.7)$$

$$\cos B = \frac{a^2 + c^2 - b^2}{2ac} \quad (2.8)$$

$$\cos C = \frac{a^2 + b^2 - c^2}{2ab} \quad (2.9)$$

ดังนั้นจากความสัมพันธ์นี้ สามารถหามุม α_2 ได้จาก

$$Tibia^2 = Femur^2 + L^2 - 2(Femur)(L) \cos(\alpha_2) \quad (2.10)$$

$$\alpha_2 = \cos^{-1} \frac{Tibia^2 - Femur^2 - L^2}{-2(Femur)(L)} \quad (2.11)$$

ทำให้ได้มุม α ดังนี้

$$\alpha = \alpha_1 + \alpha_2 \quad (2.12)$$

$$\alpha = \cos^{-1} \left(\frac{Z_{offset}}{L} \right) + \cos^{-1} \frac{Tibia^2 - Femur^2 - L^2}{-2(Femur)(L)} \quad (2.13)$$

และมุม β ดังนี้

$$L^2 = Tibia^2 + Femur^2 - 2(Tibia)(Femur) \cos(\beta) \quad (2.14)$$

$$\beta = \cos^{-1} \frac{L^2 - Tibia^2 - Femur^2}{-2(Tibia)(Femur)} \quad (2.15)$$

ดังนั้นจากสมการที่ (2.12) (2.13) และ (2.15) เมื่อทราบตำแหน่งของจุดปลายขา และขนาดของขาในส่วนต่าง ๆ ซึ่งมีค่าคงที่แล้ว ก็จะสามารถหาค่ามุมของแต่ละข้อต่อได้ โดยในการควบคุมอาจเป็นการกำหนดตำแหน่งของจุดปลายขาที่ต้องการ และนำตำแหน่งนั้นไปคำนวณหามุมของแต่ละข้อต่อ ซึ่งมุมที่ได้นี้จะถูกกำหนดให้กับมอเตอร์แต่ละตัวต่อไป

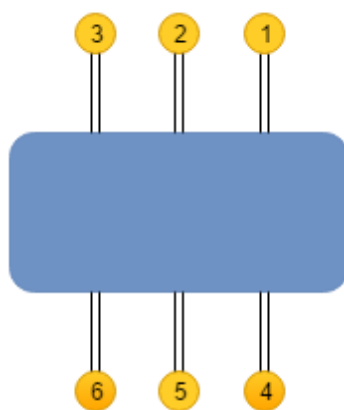
2.1.2 ลักษณะการเดินของหุ่นยนต์หกขา

ท่าทางการเดินของหุ่นยนต์หกขานั้น ค่อนข้างมีความหลากหลาย โดยท่าเดินอาจไม่คงที่ก็ได้ เช่นระยะในการก้าวขา ความสูงในการยกขา และความเร็วในการเดิน เป็นต้น ซึ่งขึ้นอยู่กับทิศทางที่ต้องการเคลื่อนที่ไป และวิธีการควบคุม โดยหลักการสำคัญในการควบคุมการเคลื่อนที่ของหุ่นยนต์หกขา คือ ความสัมพันธ์ของขาแต่ละขา ซึ่งแต่ละขาจะต้องเคลื่อนไหวสอดคล้องกัน และสัมพันธ์กับเวลาด้วย

ลักษณะการเดินจะแบ่งตามความสัมพันธ์ในการย่างก้าวของขาทั้งหกขา ซึ่งจะมีท่าเดินพื้นฐานที่มักจะนิยมใช้กับหุ่นยนต์หกขาดังนี้

- 1) ท่าเดินแบบคลื่น (Wave Gait)
- 2) ท่าเดินแบบกระเพื่อม (Ripple Gait)
- 3) ท่าเดินแบบก้ำสามขา (Tripod Gait)

โดยทั้ง 3 ท่านี้เป็นท่าพื้นฐานอาจมีท่าทางการเดินในลักษณะอื่น ๆ อีก ท่าเดินแต่ละท่าจะมีลำดับการก้าวขาแตกต่างกันไป ซึ่งแสดงได้ดังต่อไปนี้ โดยกำหนดให้การวางตำแหน่งขา และตัวหุ่นเป็นดังรูป



รูปที่ 2.6 การวางตำแหน่งขากับตัวหุ่นยนต์



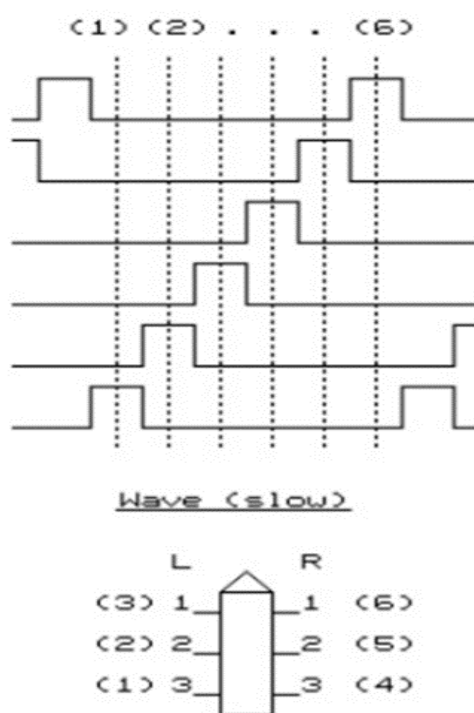
แสดงถึงปลายเท้าที่สัมผัสพื้น



แสดงถึงปลายเท้าที่ไม่สัมผัสพื้น

1) ท่าเดินแบบคลื่น (Wave Gait)

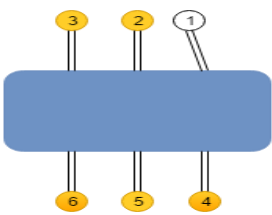
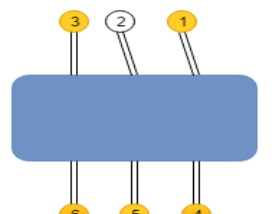
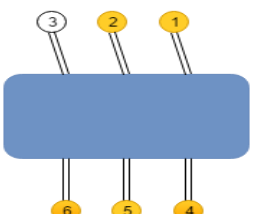
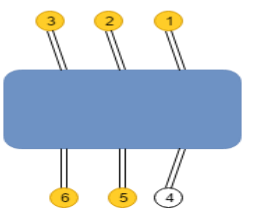
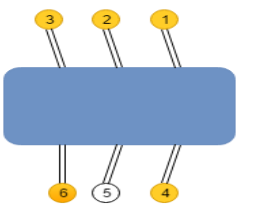
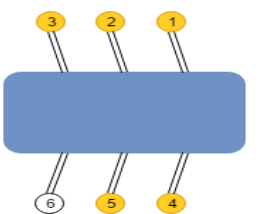
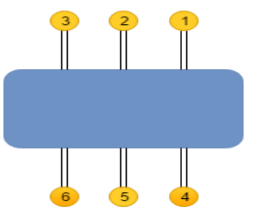
เป็นท่าเดินตามแบบฉบับของหุ่นยนต์หกที่เลียนแบบมาจากแมลงตามธรรมชาติ ท่าเดินแบบคลื่นนั้น หุ่นยนต์จะก้าวขาทีละขาเริ่มจากขาหลังสุดมายังหน้า โดยจะก้าวขาในแต่ละฝั่งให้ครบก่อน จากนั้นหุ่นยนต์จึงจะเริ่มก้าวขาอีกฝั่งของตัวหุ่นยนต์ ดังนั้นในแต่ละรอบของการก้าวขา หุ่นยนต์จะเหลือนขาขึ้นอยู่บนพื้นถึง 5 ขา ทำให้การทรงตัวของหุ่นขณะเดินด้วยท่าเดินแบบคลื่นสามารถทรงตัวได้ค่อนข้างดี โดยลำดับของการก้าวขาเมื่อเทียบกับเวลาแสดงได้ดังรูป



รูปที่ 2.7 ลำดับการก้าวขาของท่าเดินแบบคลื่นเทียบกับเวลา [3]

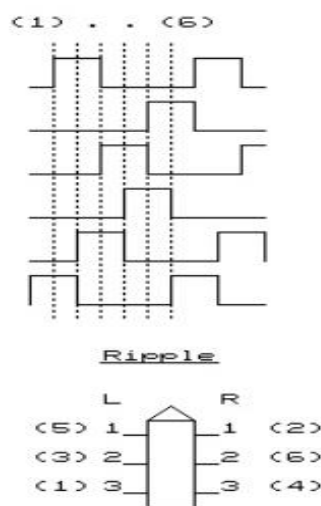
ตารางที่ 2.1 ลำดับการก้าวขาของท่า Wave Gait

ลำดับที่	รูปแสดงการก้าวขา	คำอธิบาย
1		ท่าเริ่มต้น

ลำดับที่	รูปแสดงการก้าวขา	คำอธิบาย
2		ขา 1 ยก และเลื่อนไปข้างหน้า
3		ขา 1 ลงพื้น ขา 2 ยกขึ้น และเลื่อนไปข้างหน้า
4		ขา 2 ลงพื้น ขา 3 ยกขึ้น และเลื่อนไปข้างหน้า
5		ขา 3 ลงพื้น ขา 4 ยกขึ้น และเลื่อนไปข้างหน้า
6		ขา 4 ลงพื้น ขา 5 ยกขึ้น และเลื่อนไปข้างหน้า
7		ขา 5 ลงพื้น ขา 6 ยกขึ้น และเลื่อนไปข้างหน้า
8		ขา 6 ลงพื้น ขา 1,2,3,4,5,6 เลื่อนไปข้างหลังจากนั้นกลับไปทำซ้ำลำดับที่ 1 เรื่อย ๆ

2) ทำเดินแบบกระเพื่อม (Ripple Gait)

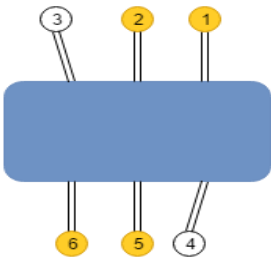
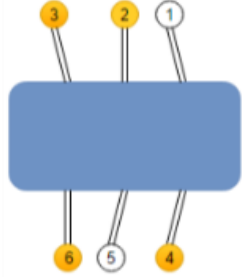
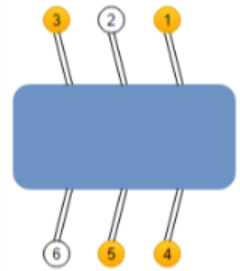
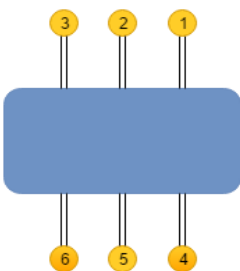
เป็นท่าเดินที่ดูค่อนข้างซับซ้อน โดยเมื่อพิจารณาจากรูปที่ 8 พบว่าท่าเดินแบบกระเพื่อมนั้นจะคล้าย ๆ กับท่าเดินแบบคลีน โดยลำดับการก้าวขาในแต่ละฝั่งยังคงเหมือนกับท่าเดินแบบคลีนคือ ก้าวจากขาหลังมายังขาหน้า สิ่งที่ต่างกันคือ ท่าเดินแบบกระเพื่อมนั้น ขาด้านหนึ่งจะเริ่มก้าวก่อนที่จะขาอีกด้านจะก้าวได้ครบทุกขา นอกจากนี้ช่วงเวลาที่แต่ละขายืนอยู่บนพื้นนั้นจะสั้นกว่าท่าเดินแบบคลีนด้วย ส่งผลให้หุ่นยนต์จะก้าวขาครั้งละ 2 ขา โดย 2 ขานี้จะก้าวเหลื่อมกัน ดังนั้นท่าเดินแบบกระเพื่อมจะมีความเร็วกว่าแบบคลีน และยังคงทรงตัวได้ดีเนื่องจากขณะหนึ่งจะมีขาอยู่บนพื้น 4 ขา และสลับกันไปเรื่อย ๆ

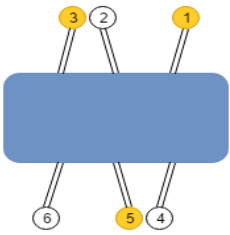
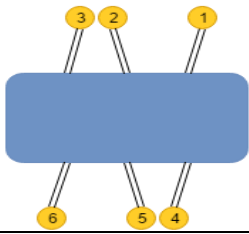
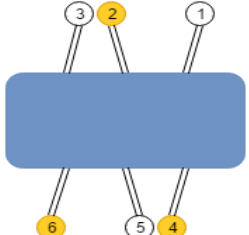
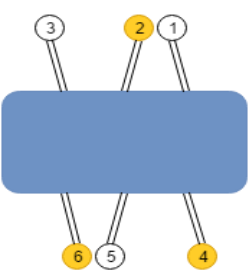
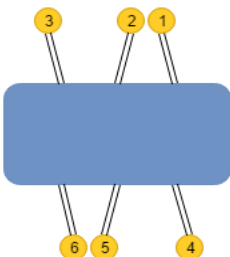
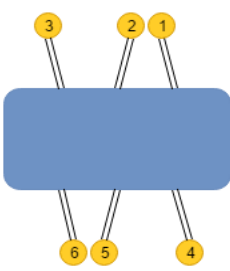


รูปที่ 2.8 ลำดับการก้าวขาของท่าเดินแบบกระเพื่อมเทียบกับเวลา [3]

ตารางที่ 2.2 ลำดับการก้าวขาของท่า Ripple Gait

ลำดับที่	รูปแสดงการก้าวขา	คำอธิบาย
1		ท่าเริ่มต้น
ลำดับที่	รูปแสดงการก้าวขา	คำอธิบาย

2		ขา 3,4 ยกขึ้น และเลื่อนไปข้างหน้า
3		ขา 3,4 ลงพื้น ขา 1,5 ยกขึ้น และเลื่อนไปข้างหน้า
4		ขา 1,5 ลงพื้น ขา 2,6 ยกขึ้น และเลื่อนไปข้างหน้า
5		ขา 2,6 ลงพื้น ขา 1,2,3,4,5,6 ยกขึ้น และเลื่อนไปข้างหลัง

2		ขา 2, 4, 6 เลื่อนไปข้างหน้า โดยที่ขา 1, 3, 5 ที่ แตะอยู่ที่พื้นเลื่อนไปข้างหลัง
3		ขา 2, 4, 6 ลงแตะพื้น
4		ขา 1,3,5 ยกขึ้นจากพื้น
5		ขา 1, 3, 5 เลื่อนไปข้างหน้า โดยที่ขา 2, 4, 6 เลื่อนไปข้างหลัง
6		ขา 1, 3, 5 ลงแตะพื้น ^{ที่} ที่แตะอยู่ที่พื้น
7		ขา 2, 4, 6 ยกขึ้นจากพื้นจากนั้น ^{ที่} กลับไปทำซ้ำ เรื่อย ๆ

2.2 การควบคุมมอเตอร์ Dynamixel AX-12A โดยใช้บอร์ด OpenCM 9.04

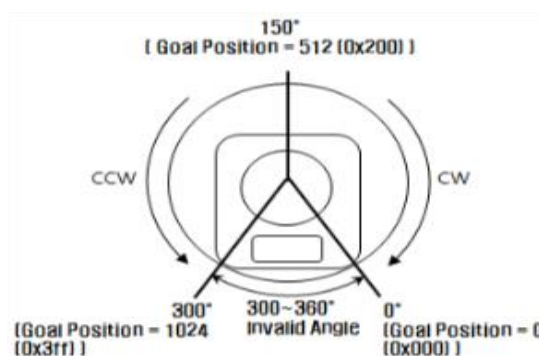
ส่วนประกอบสำคัญของหุ่นยนต์หกขา คือ ตัวขับเคลื่อนที่จะเป็นข้อต่อของขาแต่ละขา ซึ่งจะขับเคลื่อนให้ขาเคลื่อนไหวได้ ในโครงการนี้จะเลือกใช้มอเตอร์ Dynamixel AX-12A ซึ่งแต่ละขาจะประกอบด้วยมอเตอร์ 3 ตัว โดยในการควบคุมมอเตอร์จะใช้บอร์ด OpenCM 9.04 ทำหน้าที่รับค่ามุมและความเร็วของแต่ละข้อต่อ จากนั้นจึงจะส่งสัญญาณควบคุมไปยังมอเตอร์ โดยใช้ OpenCM 485 Expansion Board เป็นส่วนต่อขยายเพื่อให้สามารถจ่ายกระแสขับเคลื่อนมอเตอร์หลาย ๆ ตัวพร้อมกันได้

2.2.1 มอเตอร์ Dynamixel AX-12A

เป็นดิจิทัลเซอร์โวมอเตอร์ มีวงจรในการควบคุมการทำงานของมอเตอร์กระแสดตรงที่มีเกียร์ทดในตัว และมีความแม่นยำมีระบบบัสข้อมูลที่สามารถเชื่อมต่อรวมกันได้ ทั้งหมดนี้รวมอยู่ในแพ็คเกจเดียว ซึ่งแม้จะมีขนาดเล็กแต่ก็ยังให้แรงบิดที่สูงได้ ตัวมอเตอร์ทำมาจากวัสดุคุณภาพสูงเพื่อให้มีความยืดหยุ่น และมีความแข็งแรงทน นอกจากนี้ยังมีความสามารถในการตรวจจับการเปลี่ยนแปลงของอุณหภูมิภายใน หรือระดับแรงดันไฟที่จ่ายมาได้ ซึ่งจะช่วยป้องกันการเสียหายได้



รูปที่ 2.10 มอเตอร์ Dynamixel AX-12A [7]

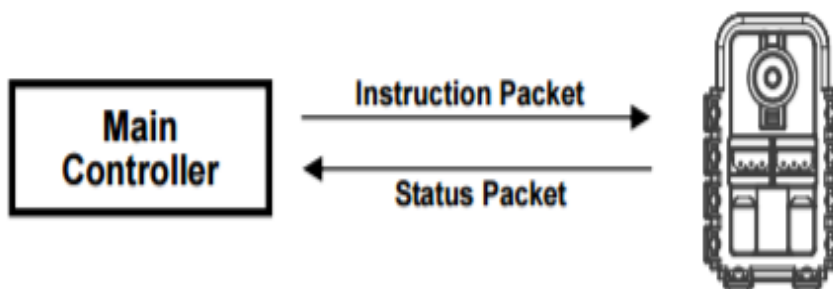


รูปที่ 2.11 ข้อมูลจำเพาะของ Dynamixel AX-12A [7]

ตารางที่ 2.4 แสดงข้อมูลจำเพาะของมอเตอร์ Dynamixel AX-12A

Weight	53.5g (AX-12/AX-12+), 54.6g (AX-12A)
Dimension	32mm * 50mm * 40mm
Resolution	0.29°
Gear Reduction Ratio	254 : 1
Stall Torque	1.5N.m (at 12.0V, 1.5A)
No load speed	59rpm (at 12V)
Running Degree	0° ~ 300° or Endless Turn
Running Temperature	-5°C ~ +70°C
Voltage	9 ~ 12V (Recommended Voltage 11.1V)
Command Signal	Digital Packet
Protocol Type	Half duplex Asynchronous Serial Communication
Link (Physical)	TTL Level Multi Drop (daisy chain type Connector)
ID	254 ID (0~253)
Communication Speed	7343bps ~ 1 Mbps
Feedback	Position, Temperature, Load, Input Voltage, etc.
Material	Engineering Plastic

การติดต่อสื่อสารกับมอเตอร์ Dynamixel AX-12A

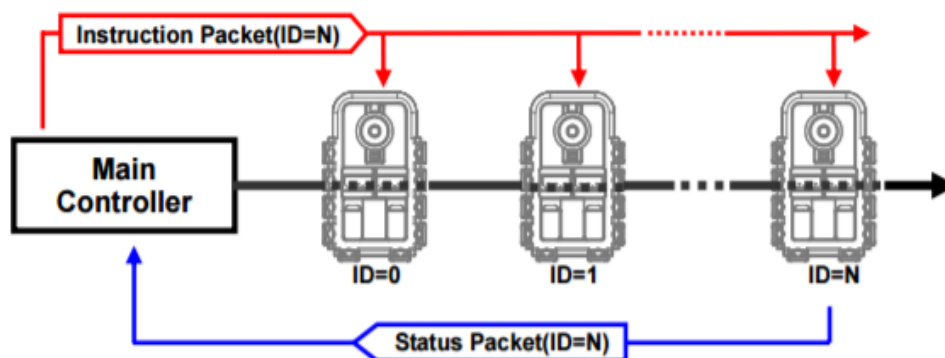


รูปที่ 2.12 การส่ง และรับข้อมูลระหว่าง Main Controller และมอเตอร์ Dynamixel [7]

คอนโทรลเลอร์จะสื่อสารกับมอเตอร์ Dynamixel โดยการส่ง และรับแพ็คเกจข้อมูล ซึ่งแพ็คเกจข้อมูลจะแบ่งออกเป็น 2 ชนิดคือ

- 1) “Instruction Packet” คือ ข้อมูลที่ส่งจากคอนโทรลเลอร์ไปยังมอเตอร์
- 2) “Status Packet” คือ ข้อมูลที่ส่งจากมอเตอร์ไปยังคอนโทรลเลอร์

โดยในการใช้งานคอนโทรลเลอร์สามารถเชื่อมต่อกับมอเตอร์ได้หลายตัวบนบัสข้อมูลเดียวกัน ซึ่งการส่ง Instruction Packet จะส่งไปพร้อมกับ ID และจะมีเพียงมอเตอร์ที่มี ID ตรงกันเท่านั้นที่จะตอบสนองโดยการส่ง Status Packet กลับมา

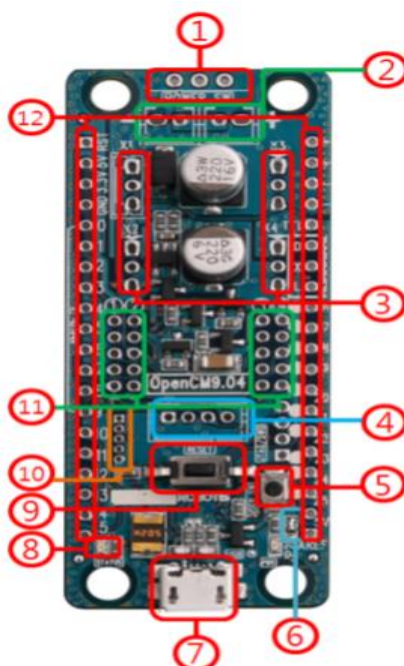


รูปที่ 2.13 การส่ง และรับข้อมูลระหว่าง Main Controller กับมอเตอร์ Dynamixel หลายตัว [7]

ถ้าหากมีมอเตอร์ที่มี ID ซ้ำกันจะทำให้เกิดการชนของข้อมูล และส่งผลให้การสื่อสารมีข้อผิดพลาด ดังนั้นในการใช้งานจริงจะต้องกำหนด ID ของมอเตอร์แต่ละตัวให้แตกต่างกัน

2.2.2 บอร์ดพัฒนา OpenCM9.04

เป็นบอร์ดพัฒนาระบบฝังตัวที่ใช้ไมโครคอนโทรลเลอร์ตระกูล ARM บนสถาปัตยกรรมแบบ 32 บิต บอร์ดพัฒนา OpenCM9.04 เป็นแพลตฟอร์มที่เปิดเผยแพร่ซอร์สโค้ดทั้งฮาร์ดแวร์ และซอฟต์แวร์ โดยมี เครื่องมือสำหรับการพัฒนาซอฟต์แวร์ชื่อ OpenCM IDE ซึ่งมีรูปแบบการพัฒนาคล้ายกับ Arduino พร้อมทั้งยังมีไลบรารีสำหรับการเขียนโปรแกรมใช้งานมอเตอร์ Dynamixel ทำให้ง่าย และ สะดวกต่อการพัฒนาส่วนประกอบ และรายละเอียดของบอร์ดพัฒนา OpenCM 9.04



รูปที่ 2.14 บอร์ดพัฒนา OpenCM 9.04 [8]

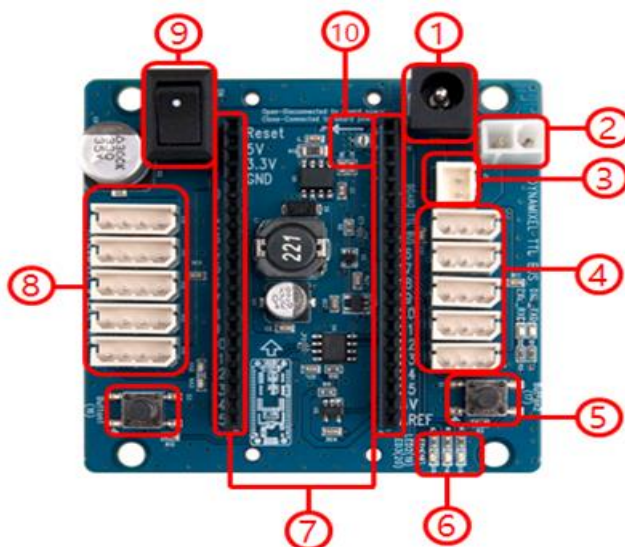
ตารางที่ 2.5 ข้อมูลของบอร์ดพัฒนา OpenCM 9.04

หมายเลข	ชื่อ	หน้าที่การทำงาน
1	Power Switch	สวิตช์เปิด/ปิดการเชื่อมต่อกับแบตเตอรี่รี
2	Battery socket	ช่องสำหรับต่อแบตเตอรี่ Li-Ion (3.7V หรือ 7.4V เมื่อต่อทั้งสองช่อง)
3	DYNAMIXEL TTL BUS	พอร์ตสำหรับมอเตอร์ TTL-based Dynamixels
4	USART PIN	พอร์ตเชื่อมต่อเพื่อติดต่อสื่อสารแบบ UART กับอุปกรณ์อื่น ๆ เช่น BT-110, BT-210
5	User Switch	ปุ่มกดที่สามารถโปรแกรมการทำงานได้
6	Analog Reference Selection Jumper	สำหรับเลือกแรงดันอ้างอิงสำหรับสัญญาณ Analog
7	Micro-B USB	ใช้เชื่อมต่อกับคอมพิวเตอร์เพื่อสื่อสาร หรือดาวน์โหลดโปรแกรม รวมทั้งรับไฟเลี้ยง
8	Status LED: LED	LED สำหรับการทดสอบการทำงานของโปรแกรม (เชื่อมต่อกับขา D16)
9	Reset Switch	สำหรับรีเซ็ตการทำงาน
10	JTAG/SWD 4 PIN	พอร์ตสำหรับการโปรแกรมแบบ JTAG
11	External Sensor PIN	พอร์ตสำหรับ เชื่อมต่อกับเซนเซอร์ของ Robotis
12	2.54 mm GPIO Header	เป็นพอร์ตอินพุต/เอาต์พุต สำหรับเชื่อมต่อกับอุปกรณ์ภายนอก

2.2.3 OpenCM 485 Expansion Board

ในการควบคุมมอเตอร์ Dynamixel จะใช้บอร์ดส่วนขยาย OpenCM 485 สำหรับต่อขยายกับ OpenCM9.04 เพื่อให้สามารถขับมอเตอร์ Dynamixel ได้มากขึ้น

ส่วนประกอบ และรายละเอียดของบอร์ดส่วนขยาย OpenCM 485 EXP



รูปที่ 2.15 OpenCM 485 EXP [9]

ตารางที่ 2.6 ส่วนประกอบของบอร์ดส่วนขยาย OpenCM 485 EXP

หมายเลข	ชื่อ	หน้าที่การทำงาน
1	SMPS DC Connector	สำหรับต่อ DC Adapter เพื่อจ่ายไฟให้กับ OpenCM 485
2	DXL Pro Power Connector	สำหรับต่อไฟเลี้ยงให้กับ Dynamixel Pro (24V)
3	Li-Po battery Connector	สำหรับต่อแบตเตอรี่ Li-Po 11.1V
4	Dynamixel TTL 3-Pin Bus	พอร์ตสำหรับมอเตอร์ TTL-based Dynamixels
5	User Button	ปุ่มกดที่สามารถโปรแกรมการทำงานได้
6	User LED	LED แสดงสถานะ สามารถโปรแกรมการทำงานได้

ตารางที่ 2.6 (ต่อ) ส่วนประกอบของบอร์ดส่วนขยาย OpenCM 485 EXP

7	I/O header	สำหรับติดตั้ง OpenCM9.04 (2.54mm pitch)
8	Dynamixel 485 4-Pin Bus	พอร์ตสำหรับมอเตอร์ Dynamixel TTL Bus แบบ 4 ขา
9	Power Switch	สวิตช์เปิด/ปิด
10	JP1 Jumper	Jumper สำหรับเลือกการจ่ายไฟเลี้ยงจาก OpenCM 485 ไปยัง OpenCM9.04

ตารางที่ 2.7 ข้อมูลจำเพาะของบอร์ดส่วนขยาย OpenCM 485

Size	68 mm X 66.5 mm x 16 mm
Weight	32 g
Input Voltage	5~30V
PowerSMPS	LIPO, DXL PRO 24V
Power	Switch1
DYNAMIXEL	Port4Pin x 5, 3Pin x 5
Buttons	2
LED	5

2.2.4 ชุดพัฒนาสมองกลฝังตัวแบบ SoPC

System on Programmable Chip (SoPC) คือ การพัฒนาทั้งระบบให้รวมอยู่ในชิปเดียว เหมือนกับ System on Chip (SoC) โดยจุดที่แตกต่างคือ SoC จะออกแบบระบบอยู่บนฮาร์ดแวร์ที่ตายตัว เมื่อออกแบบฮาร์ดแวร์เสร็จแล้วจะแก้ไขไม่ได้ ในขณะที่ SoPC เป็นออกแบบระบบอยู่บนฮาร์ดแวร์ที่สามารถโปรแกรมวงจรได้เช่น FPGA ทำให้การพัฒนาสามารถพัฒนาทั้งฮาร์ดแวร์ และซอฟต์แวร์ไปพร้อมกันได้

โดยในโครงการนี้ได้เลือกใช้บอร์ดพัฒนา ZyboZ7 ซึ่งเป็นบอร์ดเริ่มต้นสำหรับการพัฒนาระบบที่เป็น SoPC



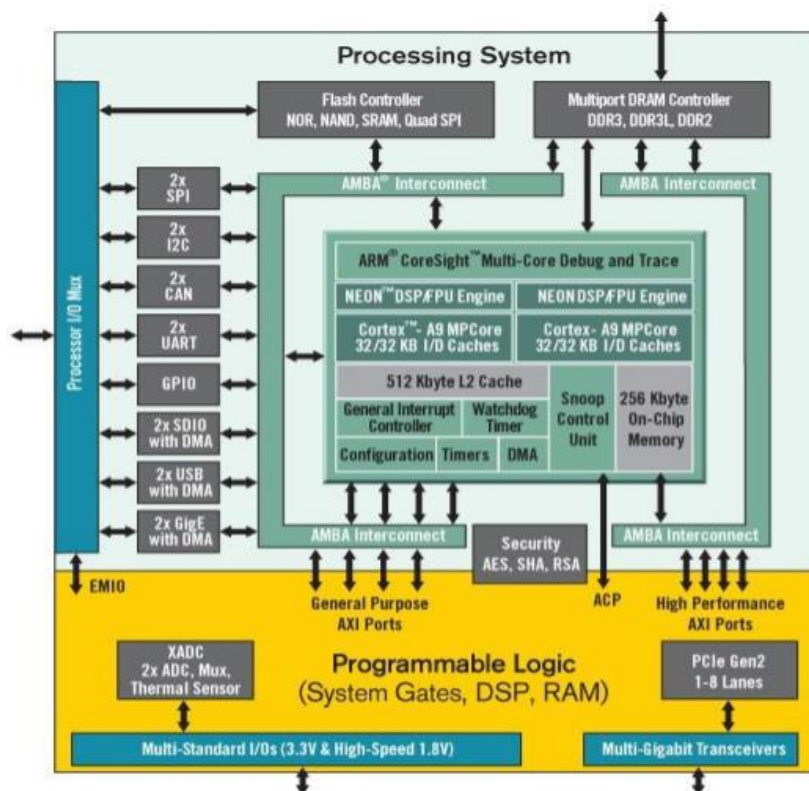
รูปที่ 2.16 Zybo Z7 [11]

Zybo Z7 เป็นบอร์ดสำหรับการพัฒนาซอฟต์แวร์และวงจรดิจิทัลที่มาพร้อมคุณสมบัติหลากหลายและพร้อมใช้งานที่สร้างขึ้นในรอบวัสดุครอบครองครอบคลุมของครอบครัวรอบตระกูล Xilinx Zynq-7000 ซึ่งมีโครงสร้าง Xilinx All Programmable System-on-Chip (AP SoC) ที่รวมโปรเซสเซอร์ ARM Cortex-A9 สองคอร์ร่วมกับ FPGA ตระกูล 7 ของ Xilinx ที่มีตอกวงจรอักขระที่สามารถโปรแกรมได้ (FPGA) อย่างใกล้ชิดกัน Zybo Z7 ต่อบน Zynq ด้วยชุดของอุปกรณ์มัลติมีเดียและการเชื่อมต่อที่หลากหลายเพื่อสร้างคอมพิวเตอร์บอร์ดเดียวที่แข็งแกร่ง แม้จะยังไม่คิดถึงความยืดหยุ่นและความทรงพลังที่ได้จาก FPGA

ชุดคุณสมบัติของ Zybo Z7 สำหรับการแสดงภาพรวมถึงคุณสมบัติที่สามารถแสดงวิดีโอได้ เช่น สามารถเชื่อมต่อกับกล้อง Pcam ที่เข้ากับ MIPI CSI-2, มีอินพุต HDMI, อินพุต HDMI และแอดพุต HDMI, รวมถึงความแข็งแรงของ DDR3L สูง โดยเลือกมาเพื่อให้เป็นทางเลือกที่สามารถเอาไปใช้ในการใช้งานแสดงภาพขั้นสูงที่รู้จักกันสำหรับ Xilinx FPGAs การเชื่อมต่ออุปกรณ์เสริมเป็นเรื่องง่ายด้วยช่องเชื่อมต่อ Pmod ของ Zybo Z7 ซึ่งช่วยให้เข้าถึงบอร์ดอุปกรณ์เสริม Pmod กว่า 70 รายการของ Digilent รวมถึงควบคุมมอเตอร์ เซนเซอร์ จอแสดงผล และอื่น ๆ อีกมากมายได้อย่างง่ายดายและสะดวก

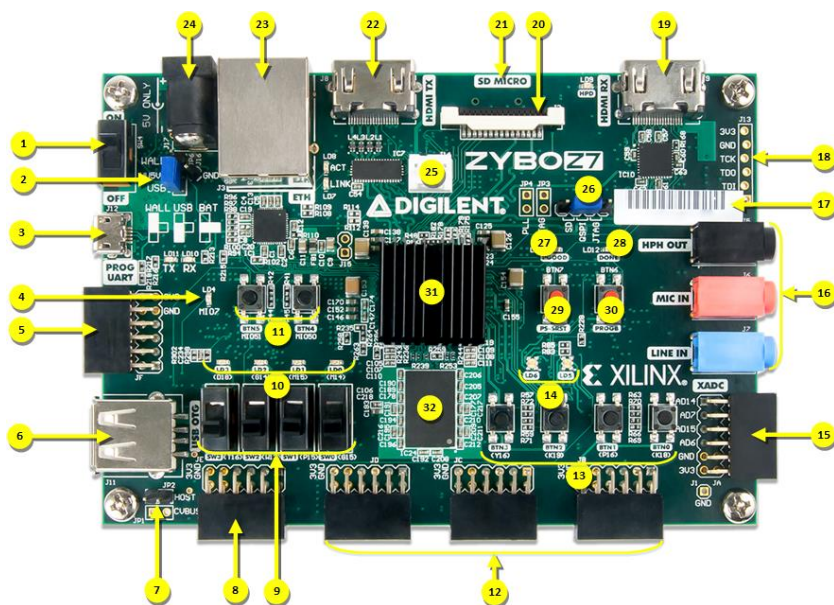
ตารางที่ 2.8 แสดงข้อมูลจำเพาะของ ZYBO Zynq-7000 development board และข้อมูลจำเพาะของ Zynq-7010 AP SoC

ข้อมูลจำเพาะของ ZYBO Zynq 7000 development board	ข้อมูลจำเพาะของ Zynq-7010 AP SoC
ZYNQ XC7Z010-1CLG400C	650 MHz dual-core Cortex-A9
512MB x32 DDR3 w/ 1050Mbps bandwidth	Processor DDR3 memory controller with 8 DMA
16-bits per pixel VGA source port	High-bandwidth peripheral
Trimode (1Gbit/100Mbit/10Mbit) Ethernet PHY	controllers: 1G Ethernet, USB 2.0, SDIO
MicroSD slot (supports Linux file system)	Low-bandwidth peripheral controller: SPI, UART, CAN, I2C
OTG USB 2.0 PHY (supports host and device)	Reprogrammable logic equivalent to Artix-7 FPGA
External EEPROM (programmed with 48-bit globally unique EUI-48/64™ compatible identifier)	4,400 logic slices, each with four 6-input LUTs and 8 flip-flops
Audio codec with headphone out, microphone and line in jacks	240 KB of fast block RAM
128Mb Serial Flash w/ QSPI interface	Two clock management tiles, each with a phase-locked loop (PLL) and mixed mode clock manager (MMCM)
On-board JTAG programming and UART to USB converter	80 DSP slices
GPIO: 6 pushbuttons, 4 slide switches, 5 LEDs	Internal clock speeds exceeding 450MHz
Six Pmod ports (1 processor-dedicated, 1 dual analog/digital, 3 high-speed differential, 1 logicdedicated)	On-chip analog-to-digital converter (XADC)



รูปที่ 2.17 Zynq AP SoC Architecture [12]

ส่วนประกอบ และรายละเอียดของ ZYBO Z7 Zynq-7000 Development Board



รูปที่ 2.18 ZYBO Z7 Development Board [11]

ตารางที่ 2.9 แสดงส่วนประกอบบนบอร์ด ZYBO Z7

Callout	Description	Callout	Description	Callout	Description
1	Power Switch	12	High-speed Pmod ports *	23	Ethernet port
2	Power select jumper	13	User buttons	24	External power supply connector
3	USB JTAG/UART port	14	User RGB LEDs *	25	Fan connector (5V, three-wire) *
4	MIO User LED	15	XADC Pmod port	26	Programming mode select jumper
5	MIO Pmod port	16	Audio codec ports	27	Power supply good LED
6	USB 2.0 Host/OTG port	17	Unique MAC address label	28	FPGA programming done LED
7	USB Host power enable jumper	18	External JTAG port	29	Processor reset button
8	Standard Pmod port	19	HDMI input port	30	FPGA clear configuration button
9	User switches	20	Pcam MIPI CSI-2 port	31	Zynq-7000
10	User LEDs	21	microSD connector (other side)	32	DDR3L Memory
11	MIO User buttons	22	HDMI output port	* denotes difference between Z7-10 and Z7-20	

2.2.5 Accelerometers & Gyroscope GY-521



รูปที่ 2.19 Accelerometers & Gyroscope GY-521 [14]

GY-521 เป็นโมดูล Accelerometers & Gyroscope ซึ่งสามารถทำงานได้ทั้ง 2 อย่างในเวลาเดียวกันใช้ในการตรวจสอบทิศทางการเคลื่อนที่ และสามารถใช้ในการตรวจสอบความเร็วในการเปลี่ยนแปลงทิศทางของแกน XYZ ได้ ยกตัวอย่าง ถ้าวัตถุเกิดการเคลื่อนที่หรือเอียง Output ของ Accelerometer จะบอกค่าของการเอียงว่าสถานะปัจจุบันค่าของ XYZ อยู่ที่เท่าไร แต่ Gyroscope จะวัดค่าได้ตอนที่กำลังเอียงหรือตอนกำลังเคลื่อนไหวเท่านั้นเมื่อวัตถุหยุดนิ่งค่าของ Gyroscope จะวัดไม่ได้เพราะไม่มีการเคลื่อนไหว

ตารางที่ 2.10 แสดงคุณสมบัติของ Accelerometers & Gyroscope GY-521

คุณสมบัติ	คำอธิบาย
ใช้ไฟเลี้ยง	+3.3 ถึง +5 V
ชื่อชิป	MPU6050
การเชื่อมต่อ	เชื่อมต่อผ่านบัส I2C
อุณหภูมิที่รองรับ	-40 to +85 °C
ระดับที่ทดสอบการกระแทก	1.8 เมตร
ขนาด	16 mm * 20 mm

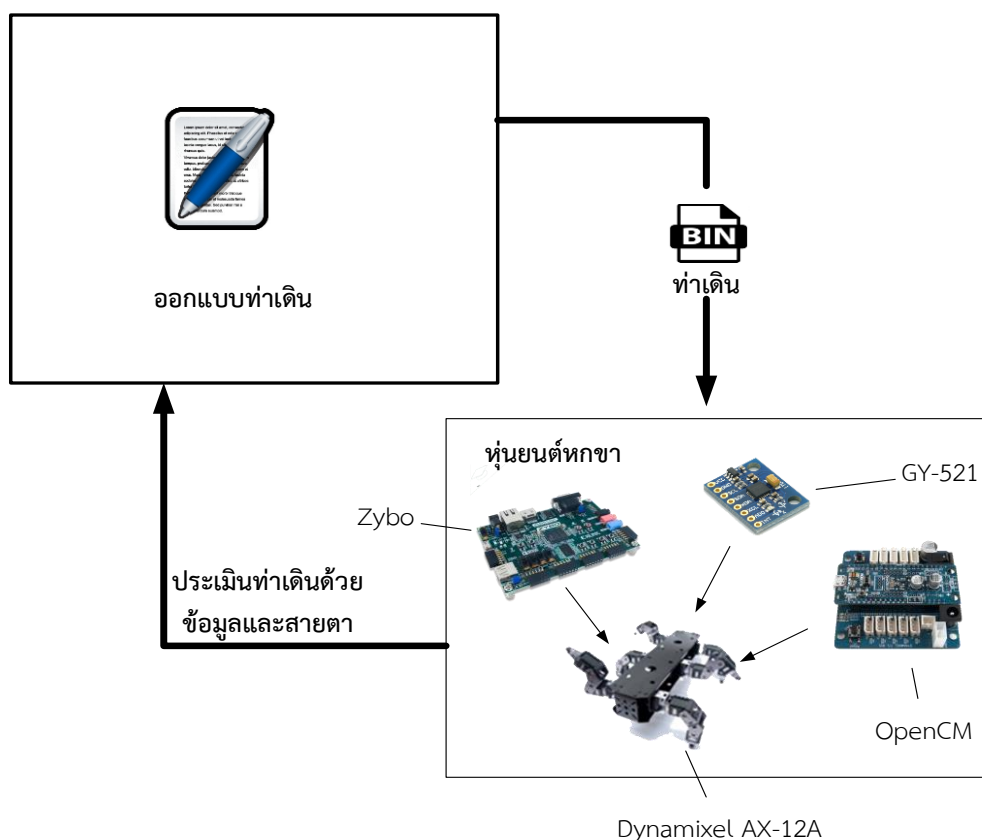


รูปที่ 2.20 โครงสร้างของ Accelerometers & Gyroscope GY-521 [14]

บทที่ 3

รายละเอียดโครงงาน

3.1 ภาพรวมของโครงงาน



รูปที่ 3.1 ภาพรวมการทำงาน

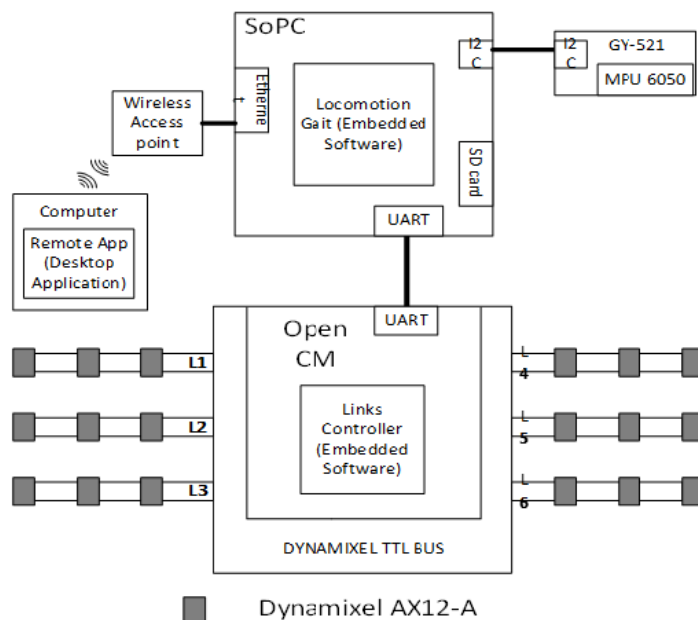
โครงงานนี้จะมุ่งเน้นความสนใจไปที่การควบคุมรูปแบบท่าทางการเดินของหุ่นยนต์หกขา ซึ่งในการทำงานเบื้องต้นจะเริ่มออกแบบท่าเดิน จากนั้นก็นำรูปแบบท่าเดินนั้นไปทดลองกับหุ่นยนต์จริง เพื่อประเมินท่าทางการเดินในสภาพแวดล้อมจริง และนำจุดบกพร่องไปพัฒนาท่าเดินให้ดีขึ้นต่อไป

การออกแบบท่าเดินในซอฟต์แวร์นั้นจะเป็นการออกแบบเส้นทาง(trajecory) โดยจะต้องกำหนดตำแหน่งของจุดปลายที่จะให้ขาก้าวไปวางที่ตำแหน่งนั้น และกำหนดเวกเตอร์ทิศทางที่จะให้ขาลากไปกับพื้น จากนั้นในการจำลองก็จะสร้างเส้นทางของจุดปลายขาจากตำแหน่ง และเวกเตอร์ที่กำหนดนี้ และใช้หลักการจลนศาสตร์ผกผันเพื่อหามุมของข้อต่อแต่ละอัน แล้วประเมินว่าค่ามุมที่ได้ สามารถนำไปใช้ได้หรือไม่

การนำท่าเดินที่ออกแบบไปใช้ในหุ่นจริง โดยมีซอฟต์แวร์ฝังตัวที่ทำงานอยู่บนบอร์ดพัฒนา SoPC ทำหน้าที่ตัดสินใจเลือกท่าเดินที่ออกแบบไว้ แล้วควบคุมให้ขาก้าวอย่างตามรูปแบบพื้นฐานที่ออกแบบไว้ รวมถึงควบคุมให้ท่าเดินมีความยืดหยุ่นมากขึ้น และควบคุมจังหวะในการก้าวของแต่ละขาให้สัมพันธ์กันด้วย โดยจะควบคุมจังหวะการส่งค่ามุม และความเร็วไปยัง OpenCM 9.04 เพื่อนำไปควบคุมมอเตอร์ต่อไป

การประเมินท่าเดินจะประเมินด้วยสายตา ประกอบกับข้อมูลที่ได้จากเซนเซอร์ IMU เพื่อการหมุนเอียงของตัวหุ่นยนต์หกขา โดยจะเปรียบเทียบท่าเดินที่หลากหลายได้ โดยสามารถกำหนดความเร็ว ขนาดความสูงของการย่างก้าว และรูปแบบการเดินได้

3.2 ระบบควบคุมแบบกระจายสำหรับหุ่นยนต์หกขา



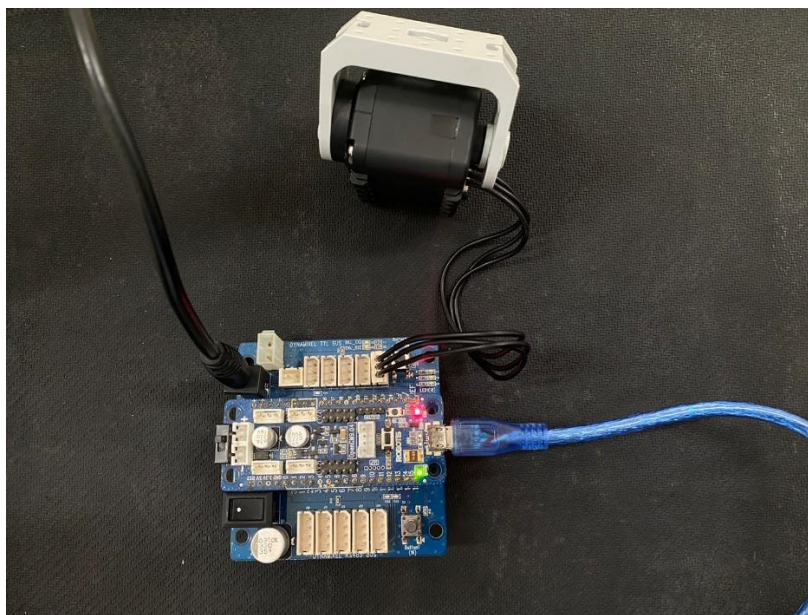
ที่ 3.2 สถาปัตยกรรมของระบบ

ตัวหุ่นยนต์หกขาจะประกอบด้วยบอร์ดพัฒนา ZYBO Z7 เป็นชุดพัฒนาสมองกลฝังตัวแบบ SoPC ภายในจะมีซอฟต์แวร์ฝังตัวที่พัฒนาขึ้นเพื่อประมวลผลท่าเดินที่ยืดหยุ่น เช่นในแง่ของความเร็ว ทิศทางในการเดิน รูปแบบการเดิน และความสูงในการย่างก้าว จากนั้นคำนวณค่ามุมและความเร็วของแต่ละข้อต่อของขาแต่ละข้าง ณ ช่วงเวลาต่าง ๆ จากนั้นก็จะส่งค่ามุม และความเร็วไปยัง OpenCM 9.04 ตามจังหวะเวลา โดยการส่งข้อมูลจะส่งผ่านพอร์ตสื่อสารมาตรฐาน UART

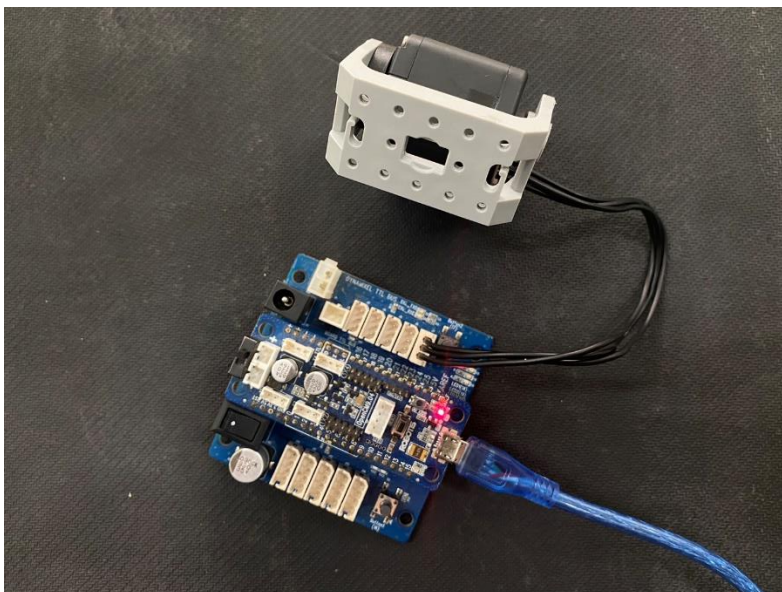
ส่วนของ OpenCM 9.04 ซึ่งเป็นบอร์ดพัฒนาไมโครคอนโทรลเลอร์ตระกูล ARM ที่ออกแบบมาเพื่อให้เหมาะกับการทำงานร่วมกับมอเตอร์ Dynamixel อยู่แล้ว โดย OpenCM 9.04 จะทำหน้าที่รับค่ามุมและความเร็วของมอเตอร์แต่ละตัว และส่งไปควบคุมมอเตอร์ตัวนั้น ๆ ซึ่งในการทำงานจะใช้ร่วมกับบอร์ดส่วนขยาย OpenCM 485 EXP เพื่อให้สามารถขับมอเตอร์หลาย ๆ ตัวได้นั่นเอง

ในโครงงานนี้จะเป็นหุ่นยนต์หกขา โดยแต่ละขาจะเป็นขาแบบสามแกน (3-DOF) หมายความว่าจะมีจุดหมุน 3 จุดด้วย ทำให้ต้องใช้มอเตอร์ 3 ตัวต่อขาแต่ละข้าง ซึ่งมีทั้งหมด 6 ขา ทำให้การควบคุมจะต้องควบคุมมอเตอร์ 18 ตัว ในโครงงานนี้เลือกใช้มอเตอร์ Dynamixel รุ่น AX-12A ซึ่งมีข้อดีคือ สามารถควบคุมการทำงานแบบดิจิทัลได้ โดยการส่งข้อมูลเป็นค่ามุม และความเร็วไป ทำให้การควบคุมทำได้ง่ายขึ้น

3.3 สิ่งที่ต้องดำเนินการไปแล้ว



รูปที่ 3.3 รูปสิ่งที่ดำเนินการไปแล้ว



รูปที่ 3.4 รูปสิ่งที่ดำเนินการไปแล้ว

บรรณานุกรม

- [1] Oscar, “INVERSE KINEMATICS BASICS TUTORIAL,” [ออนไลน์]. Available:
<https://oscarliang.com/inverse-kinematics-and-trigonometry-basics/>.
- [2] Oscar, “INVERSE KINEMATICS FOR HEXAPOD AND QUADRUPEL ROBOTS,” [ออนไลน์]. Available:
<https://oscarliang.com/inverse-kinematics-implementation-hexapod-robots/>.
- [3] Robotis, “Dynamixel AX-12A,” [ออนไลน์]. Available: http://www.hizook.com/files/users/3/AX-12_Robotis_Dynamixel_Servo_UserGuide.pdf.
- [4] Robotis, “OpenCM 9.04,” [ออนไลน์]. Available:
<http://support.robotis.com/en/product/controller/opencm9.04.htm>.
- [5] Robotis, “OpenCM 485 Expansion Board,” [ออนไลน์]. Available: <http://www.robotis.us/opencm-485-expansion-board/>.
- [6] ดร.ยุทธนา เจวจินดา, “การออกแบบระบบผสมผสานระหว่างฮาร์ดแวร์และซอฟต์แวร์,” ใน เอกสารประกอบการสอนรายวิชา 618455 การออกแบบและสร้างระบบดิจิทัล, 2558.
- [7] Digilent, “Zybo Z7 “[ออนไลน์]. Available: <https://digilent.com/reference/programmable-logic/zybo-z7/start>
- [8] Xilinx, “Zynq-7000,” [ออนไลน์]. Available: <http://www.xilinx.com/products/silicon-devices/soc/zynq-7000.html>.
- [9] “Accelerometer&Gyroscope GY-521” [ออนไลน์]. Available:
www.thaieasyelec.com/downloads/ESEN247/GY521_USG.pdf
<https://digilent.com/reference/programmable-logic/zybo-z7/reference-manual>