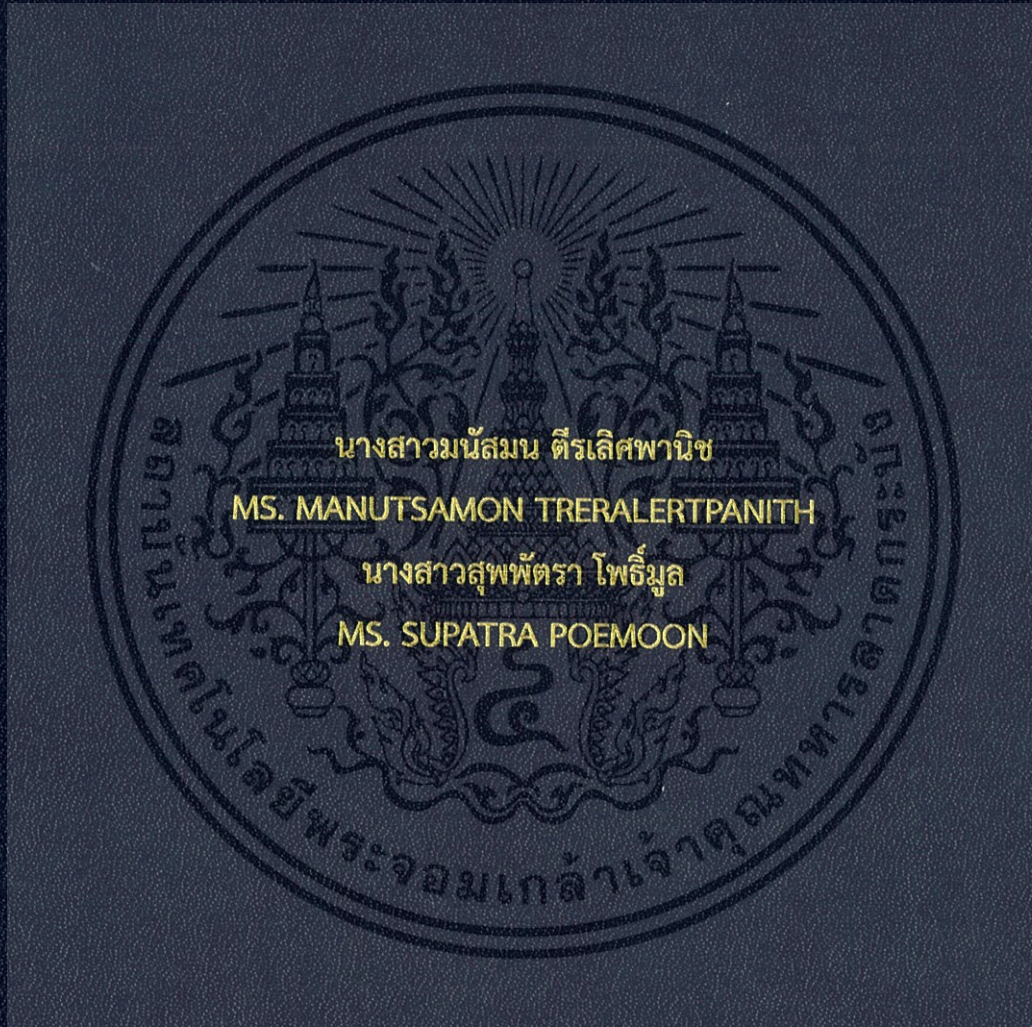


การออกแบบและพัฒนาต้นแบบขาเทียม
ควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ

DESIGN AND DEVELOPMENT OF THE EMG CONTROLLED
ABOVE KNEE PROSTHESIS PROTOTYPE FOR AMPUTEES



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2561

การออกแบบและพัฒนาต้นแบบขาเทียม
ควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ

DESIGN AND DEVELOPMENT OF THE EMG CONTROLLED
ABOVE KNEE PROSTHESIS PROTOTYPE FOR AMPUTEES



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DESIGN AND DEVELOPMENT OF THE EMG CONTROLLED ABOVE KNEE PROSTHESIS PROTOTYPE FOR AMPUTEES



A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INDUSTRIAL ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2018

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญาโท

หัวข้อปริญญาโท

การออกแบบและพัฒนาต้นแบบขาเทียมควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ
DESIGN AND DEVELOPMENT OF THE EMG CONTROLLED ABOVE
KNEE PROSTHESIS PROTOTYPE FOR AMPUTEES

นักศึกษา

นางสาวมนัสมน ตีรเลิศพานิช รหัสประจำตัว 58011005
นางสาวสุพัตรา โพธิ์มูล รหัสประจำตัว 58011356

หลักสูตร

วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมอุตสาหการ

อาจารย์ผู้ควบคุมปริญญาโท



(ผศ.ดร.มนัสชนก จงประสิทธิ์พร)



(ผศ.ดร.อุดม จันทร์จรัสสุข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ การออกแบบและพัฒนาต้นแบบขาเทียมควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ
นักศึกษา นางสาวมนัสมน ตีรเลิศพานิช
นางสาวสุพัตรา โพธิ์มูล
หลักสูตร วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมอุตสาหการ
อาจารย์ผู้ควบคุมปริญญานิพนธ์ ผศ.ดร.มนัสชนก จงประสิทธิ์พร
ผศ.ดร.อุดม จันทร์จรัสสุข

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาขาเทียมต้นแบบประเภทเหนือเข่า (Above Knee Prosthesis) ให้มีระบบควบคุมการเคลื่อนไหวที่ใช้สัญญาณควบคุมจากคลื่นไฟฟ้ากล้ามเนื้อ (EMG) เนื่องจากในประเทศไทยยังต้องนำเข้าจากต่างประเทศซึ่งมีราคาสูง รวมถึงเทคโนโลยีที่ใช้ควบคุมการทำงานของขาเทียมมักควบคุมโดยใช้การเคลื่อนไหวของร่างกายเท่านั้น จึงได้มีการออกแบบระบบควบคุมการเคลื่อนไหวโดยคำนึงถึงราคาต้นทุนเป็นสำคัญ ด้วยการนำสัญญาณคลื่นไฟฟ้าจากกล้ามเนื้อขาท่อนบนทั้ง มัด 4 ได้แก่ Biceps femoris, Semitendinosus, Vastus medialis และ Vastus lateralis ซึ่งวัดได้จากตัววัดสัญญาณ (Surface Electrode) มาเข้าวงจรทางไฟฟ้าเพื่อแปลงสัญญาณและจำลองรูปแบบลักษณะคลื่นไฟฟ้ากล้ามเนื้อจากการสร้างโมเดลส่วนการรับรู้ของเครื่อง (Machine Learning) โดยใช้โปรแกรม Matlab ซึ่งแบบจำลองที่เหมาะสมได้แก่ แบบจำลองของ Levenberg-Marquardt โดยใช้จำนวนชั้นซ่อนตัว 3 ชั้นและให้ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยน้อยที่สุดคือ 0.4441 หลังจากนั้นจะส่งไปยังระบบควบคุมขาเทียมเพื่อประมวลผลด้วยโปรแกรม Arduino ให้ขาเทียมเคลื่อนไหวได้ตามความต้องการของผู้ใช้งาน

ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title Design and Development of the EMG Controlled Above Knee Prosthesis Prototype for Amputees

Student Ms. Manutsamon Treralertpanith
Ms. Supatra Poemoon

Degree Bachelor of Engineering in Industrial Engineering
King Mongkut's Institute of Technology Ladkrabang

Academic Year 2018

Thesis Advisor Asst.Prof.Dr. Manutchanok Jongprasithporn
Asst.Prof.Dr. Udom Janjarassuk

ABSTRACT

The objective of this research is to develop an above knee prosthesis to provide a motion controlled system by using electromyography (EMG). The EMG controlled above knee prosthesis in Thailand must be imported from abroad, resulting in a high price. This research studied a motion controlled system in consideration of low-cost constraint by designing the controlled system to handle the movement of prosthetic legs. The electromyography (EMG) signals were measured during gait cycle to control prosthesis at various walking levels. The electrical circuit converted the EMG signals from four groups of upper leg muscles including Biceps femoris, Semitendinosus, Vastus medialis and Vastus lateralis, simulating the model of EMG pattern by machine learning in Matlab. The best model is Levenberg-Marquardt with 3 hidden layers which provided the mean squared error approximately 0.4441. The predicted knee angle from this model was sent to the prosthetic control system to process with PID in Arduino program to move knee angle of prosthesis in the position where people needs.

กิตติกรรมประกาศ

ปริญญาานิพนธ์เรื่อง การออกแบบและพัฒนาอุปกรณ์ต้นแบบเพื่อใช้วัดพิสัยการเคลื่อนไหวสามารถสำเร็จลุล่วงไปได้ด้วยดี กลุ่มผู้วิจัยขอกราบขอบพระคุณบุคคลทุกคนที่มีส่วนเกี่ยวข้องส่งผลให้ปริญญาานิพนธ์ฉบับนี้เสร็จสมบูรณ์

ผศ.ดร.มนัสชนก จงประสิทธิ์พร และ ผศ.ดร.อุดม จันท์จรัสสุข อาจารย์ที่ปรึกษาปริญญาานิพนธ์ กลุ่มผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงสำหรับการให้โอกาสในการศึกษาปริญญาานิพนธ์ฉบับนี้ รวมทั้งความรู้ คำแนะนำ ตลอดจนตรวจสอบและแก้ไขข้อบกพร่องต่าง ๆ เป็นอย่างดีตลอดจนกระทั่งปริญญาานิพนธ์สามารถสำเร็จลุล่วงด้วยดี

โรงเรียนกายอุปกรณ์สิรินธร คณะแพทยศาสตร์ศิริราชพยาบาล มหาวิทยาลัยมหิดล (SSPO) กลุ่มผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูง สำหรับการให้คำแนะนำ ความช่วยเหลือทางด้านอุปกรณ์ฯ เทียม รวมถึงความรู้ในด้านสรีระวิทยาต่าง ๆ

นายกำธร สุขพิมาย และ นายสิทธิชัย บุญกิจ กลุ่มผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงสำหรับคำแนะนำ รวมทั้งการเอื้อเฟื้ออุปกรณ์ในการจัดทำปริญญาานิพนธ์

รศ.ดร.สกันธ์ คล่องบุญจิต กรรมการที่ปรึกษาปริญญาานิพนธ์ กลุ่มผู้วิจัยขอขอบพระคุณเป็นอย่างสูง ที่คอยให้คำแนะนำในการปรับปรุงแก้ไขที่ตีมาโดยตลอด

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กลุ่มผู้วิจัยขอขอบพระคุณเป็นอย่างสูง ที่มอบโอกาสในการจัดทำปริญญาานิพนธ์ฉบับนี้ รวมถึงงบประมาณที่สนับสนุนในการดำเนินการ

คณาจารย์ในสาขาวิชาวิศวกรรมอุตสาหกรรม ที่ช่วยให้ข้อเสนอแนะในการพัฒนาปริญญาานิพนธ์นี้ให้ดียิ่งๆ ขึ้นไปในอนาคต

สุดท้ายปริญญาานิพนธ์นี้จะสำเร็จไม่ได้ ถ้าไม่ได้บิดา มารดา ที่คอยให้ความรัก ความห่วงใย กำลังใจในการทำงาน และสมาชิกในกลุ่มที่คอยให้ความร่วมมือเป็นอย่างดีในการจัดทำปริญญาานิพนธ์

นางสาวมนัสมน ตีร์เลิศพานิช

นางสาวสุพัตรา โพธิ์มูล

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ช
สารบัญรูป	ซ
บทที่ 1 บทนำ	
1.1 ความเป็นมาของปริญญานิพนธ์.....	1
1.2 วัตถุประสงค์	3
1.3 ขอบเขตของปริญญานิพนธ์	3
1.4 ประโยชน์ที่คาดว่าจะได้รับ	4
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	
2.1 คลื่นไฟฟ้ากล้ามเนื้อที่สัมพันธ์กับการเดิน	5
2.1.1 คลื่นไฟฟ้ากล้ามเนื้อ.....	5
2.1.2 สรีระของกล้ามเนื้อ.....	7
2.1.3 วงจรของการเดิน (Gait cycle)	8
2.2 วิวัฒนาการของขาเทียม.....	12
2.3 ไมโครคอนโทรลเลอร์และระบบควบคุม.....	14
2.3.1 ไมโครคอนโทรลเลอร์ (Microcontroller)	14
2.3.2 บอร์ด Arduino.....	15

สารบัญ (ต่อ)

	หน้า
2.3.3 ระบบควบคุมพีไอดี (PID Controller)	16
2.4 โครงข่ายประสาทเทียม.....	19
2.4.1 Machine Learning (ส่วนการเรียนรู้ของเครื่อง)	19
2.4.2 โครงข่ายประสาทเทียม.....	20
2.5 งานวิจัยที่เกี่ยวข้อง.....	25
บทที่ 3 วิธีการดำเนินงาน	
3.1 การสร้างขาเทียมต้นแบบและออกแบบวงจรสำหรับเก็บข้อมูล.....	31
3.1.1 การสร้างขาเทียมต้นแบบ.....	31
3.1.2 วิธีคำนวณหาค่า R และระยะยืดและหดของมอเตอร์ ซึ่งสัมพันธ์กับมุมงอข้อเข่า....	34
3.1.3 การออกแบบวงจรสำหรับรับค่ามุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อ.....	36
3.1.4 การทดลองเก็บข้อมูลค่ามุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อที่เกิดขึ้นในขณะเดิน....	37
3.2 การเขียนคำสั่งเพื่อให้เกิดการเคลื่อนไหวของขาเทียมต้นแบบ.....	39
3.2.1 การสร้างแบบจำลองโครงข่ายประสาทเทียมเพื่อทำนายลักษณะการเดิน.....	39
3.2.2 การเขียนคำสั่งเพื่อควบคุมการเคลื่อนไหวของขาเทียมต้นแบบ.....	46
บทที่ 4 ผลการดำเนินงาน	
4.1 มุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อที่เกิดขึ้นในขณะเดิน.....	48
4.2 แบบจำลองที่ใช้สำหรับทำนายค่ามุมงอข้อเข่า.....	50
บทที่ 5 สรุปผลและข้อเสนอแนะ	
5.1 สรุปผล	56
5.2 ข้อจำกัด	56

สารบัญ (ต่อ)

	หน้า
5.3 ข้อเสนอแนะ.....	57
5.4 งานในอนาคต	57
หนังสืออ้างอิง.....	58
ภาคผนวก.....	ผ1



สารบัญตาราง

	หน้า
ตารางที่ 2.1 การทำงานของข้อต่อต่าง ๆ และกล้ามเนื้อในขณะเดิน	12
ตารางที่ 2.2 ผลกระทบจากการปรับแต่งค่าพารามิเตอร์ของตัวควบคุมพีไอดี	18
ตารางที่ 2.3 ความสัมพันธ์ของค่าพารามิเตอร์ของตัวควบคุมพีไอดี	19
ตารางที่ 3.1 ความสัมพันธ์ระหว่างระยะยึดหดของมอเตอร์ (d) มุมงอข้อเข่า (θ) และค่า R ที่วัดได้จากเซนเซอร์	35
ตารางที่ 4.1 ค่าความคลาดเคลื่อนกำลังสอง (MSE) ในแบบจำลองประเภทต่าง ๆ	50
ตารางที่ 4.2 ค่าความคลาดเคลื่อนกำลังสอง (MSE) ในแบบจำลองที่มีจำนวนชั้นซ่อนต่าง ๆ	52
ตารางที่ 4.3 แบบจำลองโครงข่ายประสาทเทียมที่เหมาะสม	54

สารบัญรูป

หน้า

รูปที่ 2.1 การปรับสัญญาณ EMG จาก Raw EMG signal มาเป็น Integrated EMG signal.....	6
รูปที่ 2.2 ลักษณะของอิเล็กโทรดที่ใช้บันทึกสัญญาณ EMG ชนิด Needle electrode และ Surface electrode.....	6
รูปที่ 2.3 กล้ามเนื้อส่วนสะโพกและขา.....	8
รูปที่ 2.4 วงจรการเดิน (Gait cycle).....	9
รูปที่ 2.5 ประเภทของขาเทียม.....	13
รูปที่ 2.6 โครงสร้างระบบประสาทในสมอง.....	21
รูปที่ 2.7 กระบวนการทำงานของโครงข่ายใยประสาทประคิฐฐึโนโหนดยอย.....	23
รูปที่ 2.8 โครงข่ายใยประสาทเทียมแบบแพร์กลับ.....	24
รูปที่ 2.9 มุมงอข้อเข่าที่เกิดขึ้นในแต่ละช่วงของวงจรการเดิน.....	26
รูปที่ 2.10 การทำงานของกล้ามเนื้อ Vastus Lateralis, Biceps femoris และ Semitendinosus.....	27
รูปที่ 2.11 การทดลองเดินโดยใช้ขาเทียมแบบควบคุมโดยใช้คลื่นไฟฟ้ากล้ามเนื้อ.....	28
รูปที่ 2.12 แผนภาพการทำงานของขาเทียมควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ.....	28
รูปที่ 3.1 แผนผังขั้นตอนการดำเนินงาน.....	30
รูปที่ 3.2 ลักษณะของ Linear Actuator Motor.....	31
รูปที่ 3.3 ลักษณะของไมโครคอนโทรลเลอร์ (Microcontroller) Arduino Nano.....	32
รูปที่ 3.4 ลักษณะของชุดขับเคลื่อนมอเตอร์ L298N Dual H-Bridge.....	32
รูปที่ 3.5 ลักษณะของอิเล็กโทรดแบบสัมผัสผิวหนัง (Surface Electrode)	33
รูปที่ 3.6 ลักษณะของขาเทียมต้นแบบ	33
รูปที่ 3.7 จุดหมุนและมุมต่างที่สัมพันธ์กับมุมงอข้อเข่า (Knee angle)	34
รูปที่ 3.8 วงจรของอิเล็กโทรด 1 ช่องสัญญาณ.....	36
รูปที่ 3.9 วงจรสำหรับรับค่ามุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อ	36
รูปที่ 3.10 โปรแกรม PLX-DAQ สำหรับรับข้อมูลและนำมาจัดเก็บในโปรแกรมไมโครซอฟท์เอ็กเซล.....	37
รูปที่ 3.11 การทดลองเดินบนลู่วิ่งเพื่อเก็บข้อมูลค่ามุมงอข้อเข่าและคลื่นไฟฟ้า.....	38
รูปที่ 3.12 ตัวอย่างข้อมูลที่เก็บได้แสดงในโปรแกรมไมโครซอฟท์เอ็กเซล.....	38

๗

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 3.13	โครงข่ายประสาทเทียมซึ่งประกอบไปด้วย ชั้นข้อมูลป้อนเข้า ชั้นซ่อน และชั้นผลลัพธ์.....	40
รูปที่ 3.14	กระบวนการสอน (Training) ในโครงข่าย	42
รูปที่ 3.15	หน้าต่างสำหรับนำเข้าข้อมูลค่าคลื่นไฟฟ้ากล้ามเนื้อและค่ามุมมองข้อเข่า	43
รูปที่ 3.16	Two-layer feed-forward network ในฟังก์ชัน Neural Network Fitting Application.....	43
รูปที่ 3.17	หน้าต่างแสดงการกำหนดค่าร้อยละการถ่วงน้ำหนัก	44
รูปที่ 3.18	หน้าต่างแสดงการกำหนดจำนวน Hidden Layer	44
รูปที่ 3.19	หน้าต่างแสดงการเลือกประเภทต้นแบบทางคณิตศาสตร์	45
รูปที่ 3.20	หน้าต่างแสดงการสอนของโปรแกรม	45
รูปที่ 3.21	ค่าผลลัพธ์มุมมองข้อเข่าที่เกิดจากการสอนของโปรแกรมเปรียบเทียบกับค่ามุมเข่าจริง	46
รูปที่ 3.22	แผนผังการควบคุมการเคลื่อนไหวของขาเทียม	47
รูปที่ 4.1	มุมมองข้อเข่าในวงจรถ่ายภาพ 1 รอบ (ก), คลื่นไฟฟ้ากล้ามเนื้อจากกล้ามเนื้อ Biceps femoris (ข), Semitendinosus (ค), Vastus medialis (ง) และ Vastus lateralis (จ)	49
รูปที่ 4.2	ผลการเปรียบเทียบระหว่างมุมมองข้อเข่าจริงกับมุมมองข้อเข่าจากแบบจำลอง Levenberg- Marquardt (ก), Bayesian Regularization (ข) และ Scaled Conjugate Gradient (ค)	51
รูปที่ 4.3	ผลการเปรียบเทียบระหว่างมุมมองข้อเข่าจริงกับมุมมองข้อเข่าจากแบบจำลอง Levenberg- Marquardt ที่มีชั้นซ่อน 3 ชั้น (ก), 4 ชั้น (ข) และ 5 ชั้น (ค)	53
รูปที่ 4.4	การเปรียบเทียบมุมมองข้อเข่าที่ได้จากแบบจำลอง Levenberg-Marquardt ที่มีชั้นซ่อน 3 ชั้น แบบจำลอง Bayesian Regularization ที่มีชั้นซ่อน 4 ชั้น และแบบจำลอง Levenberg- Marquardt ที่มีจำนวนชั้นซ่อน 4 ชั้น.....	55
รูปที่ 4.5	การเปรียบเทียบระหว่างค่ามุมมองข้อเข่าจริงกับค่ามุมเข่าที่ได้จากแบบจำลอง	55
รูปที่ 4.6	แบบจำลองโครงข่ายประสาทเทียม Levenberg-Marquardt ซึ่งมีจำนวนชั้นซ่อน 3 ชั้น.....	57
รูปที่ 4.7	การเคลื่อนไหวของขาเทียมต้นแบบซึ่งควบคุมด้วยระบบควบคุม PID ตามมุมมองข้อเข่า ในช่วงต่าง ๆ.....	58

ณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ปริญญาานิพนธ์ฉบับนี้เป็นการออกแบบและพัฒนาต้นแบบขาเทียมควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ โดยมีความเป็นมาและความสำคัญของการจัดทำปริญญาานิพนธ์ วัตถุประสงค์ และขอบเขตของงานวิจัย เป็นดังนี้

1.1 ความเป็นมาและความสำคัญ

ในปัจจุบัน ประเทศไทยพบจำนวนผู้พิการขาขาดเป็นจำนวนมาก โดยผู้พิการขาขาดคือบุคคลซึ่งมีความบกพร่องทางร่างกายจากการสูญเสียขา ทั้งที่มีสาเหตุมาจากความพิการแต่กำเนิดหรือความพิการที่เกิดขึ้นในภายหลัง เช่น อุบัติเหตุการจราจร โรคเกี่ยวกับหลอดเลือด เป็นต้น จากผลการสำรวจข้อมูลความพิการของสำนักงานสถิติแห่งชาติ ล่าสุดเมื่อวันที่ 31 มีนาคม 2561 พบว่าในประเทศไทยมีผู้พิการที่ได้รับการขึ้นทะเบียนและมีบัตรประจำตัวผู้พิการทั้งสิ้น 1,916,828 คน คิดเป็นร้อยละ 2.89 ของประชากรทั้งประเทศ เป็นเพศชาย 1,006,657 คน และเพศหญิง 910,171 คน โดยในจำนวนนี้มีผู้พิการขาขาดประมาณ 50,000 ราย และยังพบว่าผู้พิการขาขาดเกือบ 20,000 คน ที่ยังเข้าไม่ถึงบริการขาเทียม [1][2]

ในเชิงสังคมศาสตร์ การสูญเสียอวัยวะหรือการสูญเสียการทำหน้าที่ของอวัยวะต่าง ๆ จะทำให้บุคคลเปลี่ยนเอกลักษณ์ไปและถูกเรียกว่าผู้ทุพพลภาพหรือผู้พิการ ส่งผลต่อสภาพทางกาย จิตใจ และสังคม โดยผู้พิการแต่ละคนมักมีปฏิริยาตอบสนองทางด้านอารมณ์ และพฤติกรรมแตกต่างกัน เนื่องจากผู้พิการต้องใช้กำลังกาย กำลังใจ และความพยายามเพื่อปรับตัวให้สามารถช่วยเหลือตนเองในการทำกิจกรรมประจำวัน เพื่อให้สามารถดำรงชีวิตอยู่ในสังคมไม่ต่างจากคนปกติได้อย่างปกติสุข [3]

ทั้งนี้การช่วยเหลือผู้พิการขาขาดให้สามารถดำเนินชีวิตได้อย่างปกติทางหนึ่ง คือ การจัดหาขาเทียมที่มีประสิทธิภาพ เพื่อช่วยให้ผู้พิการขาขาดสามารถฟื้นฟูสมรรถภาพ ช่วยเหลือตัวเองได้และมีคุณภาพชีวิตที่ดี อย่างไรก็ตาม จากข้อมูลของกระทรวงสาธารณสุขพบว่าสภาวะความพิการขาขาดมีแนวโน้มเพิ่มสูงขึ้น โดยมีผู้พิการที่ต้องถูกตัดขาเพิ่มขึ้นประมาณปีละ 3,500 คน แต่การเข้าถึงบริการขาเทียมของผู้พิการกลับยังมีข้อจำกัดในหลายปัจจัย โดยเฉพาะอย่างยิ่งคือข้อจำกัดทางด้านราคา เนื่องจากในปัจจุบันขาเทียมบางชนิดที่มีคุณภาพดีต้องนำเข้าจากต่างประเทศ รวมถึงในกระบวนการผลิตขาเทียม

ในประเทศไทยยังต้องมีการนำเข้าวัสดุส่วนประกอบจากต่างประเทศซึ่งมีราคาสูง ในขณะที่ผู้พิการส่วนใหญ่เป็นผู้ที่มีรายได้น้อย ไม่สามารถซื้อขาเทียมที่มีคุณภาพจากต่างประเทศเหล่านี้ได้ [2]

จากผลสำรวจราคาต้นทุนขาเทียมประเภทต่าง ๆ ของสมาคมคนพิการทางการเคลื่อนไหวสากล [4] [5] ประกอบกับข้อมูลจากบริษัทที่จัดจำหน่ายและนำเข้าขาเทียมโดยเฉพาะ แสดงให้เห็นว่าขาเทียมเหนือเข่าจากต่างประเทศมีราคาสูงแตกต่างกันไปตามลักษณะการทำงาน ซึ่งสามารถแบ่งประเภทของขาเทียมออกเป็น 2 ประเภทหลัก คือ ขาเทียมแกนนอก (Exoskeleton Prosthesis) และขาเทียมแกนใน (Endoskeleton Prosthesis) โดยขาเทียมแกนนอกมีระบบการทำงานแบบควบคุมโดยใช้การเคลื่อนไหวของร่างกาย (Body Controlled) เท่านั้น มีราคาข้างละประมาณ 20,000 บาท ส่วนขาเทียมแกนในสามารถแบ่งย่อยตามลักษณะการทำงานออกเป็น 3 กลุ่มย่อย ได้แก่ ขาเทียมแกนในแบบควบคุมโดยใช้การเคลื่อนไหวของร่างกาย (Body Controlled) ราคาข้างละประมาณ 150,000 บาท ขาเทียมแกนในแบบควบคุมโดยใช้พลังงานภายนอก (External-Power Controlled) โดยหากเป็นระบบไฮดรอลิกส์ซึ่งต้องอาศัยการเคลื่อนไหวของร่างกายร่วมด้วย มีราคาข้างละประมาณ 200,000 บาท หากเป็นระบบไมโครโพรเซสเซอร์สำหรับรับสัญญาณไฟฟ้าจากกล้ามเนื้อ มีราคาข้างละประมาณ 2,300,000 บาทขึ้นไป ซึ่งเป็นราคาที่ค่อนข้างสูงเมื่อเปรียบเทียบกับรายได้ของผู้พิการส่วนใหญ่ในประเทศไทย และถึงแม้ว่าจะมีการผลิตขาเทียมเพื่อผู้ยากไร้โดยหน่วยงานต่าง ๆ เช่น มูลนิธิขาเทียมในสมเด็จพระศรีนครินทราบรมราชชนนีและชมรมนักพัฒนาอุตสาหกรรมไทย ซึ่งมีราคาข้างละประมาณ 1,000 – 2,000 บาท แต่ขาเทียมที่ผลิตได้ส่วนใหญ่ยังมีคุณภาพน้อยกว่าที่นำเข้าจากต่างประเทศ รวมถึงเทคโนโลยีที่ใช้ควบคุมการทำงานของขาเทียม ซึ่งส่วนใหญ่ขาเทียมที่ผลิตมาสำหรับผู้ยากไร้มักเป็นขาเทียมแบบควบคุมโดยใช้การเคลื่อนไหวของร่างกายเท่านั้น ทำให้ผู้พิการมีความยากลำบากในการเดิน

ขาเทียมประเภทเหนือเข่า (Above Knee Prosthesis) ที่นำเข้าจากต่างประเทศ เป็นขาเทียมคุณภาพดีที่สามารถงอที่ข้อเข่าได้ ทำให้ลักษณะการเดินมีความนุ่มนวลและมีสภาพเหมือนบุคคลทั่วไปมากขึ้น เช่น ขาเทียมแบบซีเลก (C-Leg Prosthesis) เป็นขาเทียมแบบควบคุมโดยใช้พลังงานภายนอกโดยอาศัยระบบไมโครโพรเซสเซอร์ มีข้อต่อสองจุด คือ บริเวณเข่าและข้อเท้า โดยข้อต่อบริเวณข้อเท้าสามารถเคลื่อนที่ได้เล็กน้อย และข้อต่อบริเวณเข่าจะเป็นส่วนเคลื่อนที่หลัก โดยจะมีการประมวลผลคลื่นไฟฟ้าที่วัดได้จากกล้ามเนื้อขาท่อนบนเพื่อควบคุมเสถียรภาพในการเดิน [6] ซึ่งการเดินเป็นกระบวนการที่อาศัยความสัมพันธ์อันซับซ้อนของข้อสะโพก ข้อเข่า ข้อเท้าและเท้า โดยใช้ขาทั้งสองข้างเคลื่อนไหวสลับกันเพื่อทำให้ร่างกายเคลื่อนที่ไปทางด้านหน้าและพยุ่งน้ำหนักของร่างกาย โดยวงจรการเดิน 1 รอบนั้นจะสังเกตได้จากการที่เท้าสัมผัสพื้น (Stance Phase) ยกขึ้นลอยจากพื้น (Swing Phase)

จนกระทั่งเวียนกลับมาสัมผัสพื้นใหม่อีกครั้ง ขาเทียมประเภทนี้จึงสามารถงอและเหยียดตรงได้ถูกจังหวะ โดยสามารถวิเคราะห์ลักษณะการเดิน (Gait Analysis) ของแต่ละบุคคลและปรับแต่งลักษณะจังหวะการเดินให้เหมาะสมกับผู้พิการแต่ละคนได้ ทำให้ผู้พิการสามารถเดินได้เหมือนคนทั่วไปและดำเนินชีวิตได้อย่างปกติ [7]

จากข้อมูลข้างต้นผู้จัดทำได้เล็งเห็นถึงความสำคัญของขาเทียมสำหรับผู้พิการที่ถูกตัดขา รวมถึงต้องการลดช่องว่างระหว่างการเข้าถึงบริการขาเทียมกับผู้พิการที่มีรายได้น้อย จึงได้ทำการศึกษาพัฒนาขาเทียมประเภทเหนือเข่า (Above Knee Prosthesis) ให้มีระบบควบคุมการเคลื่อนไหวและมีประสิทธิภาพในราคาต้นทุนที่ต่ำที่สุด ซึ่งจะช่วยให้ผู้พิการสามารถเข้าถึงบริการขาเทียมได้โดยไม่มีข้อจำกัดทางด้านราคา และช่วยให้ผู้พิการขาขาดมีความสะดวกสบายในการเดินมากขึ้น โดยจะใช้สัญญาณคลื่นไฟฟ้ากล้ามเนื้อ หรือที่เรียกว่า Electromyography (EMG) ในการควบคุมการเคลื่อนไหวของขาเทียมในระดับการเดินต่าง ๆ โดยนำสัญญาณคลื่นไฟฟ้าจากกล้ามเนื้อขาที่นอนบนที่ได้จากตัววัดสัญญาณมาเข้าวงจรทางไฟฟ้าเพื่อแปลงสัญญาณและส่งไปยังระบบควบคุมขาเทียมให้เคลื่อนไหวได้ตามความต้องการของผู้ใช้งาน

1.2 วัตถุประสงค์

1. เพื่อพัฒนาขาเทียมต้นแบบประเภทเหนือเข่า (Above Knee Prosthesis) ให้มีระบบควบคุมการเคลื่อนไหวที่ใช้สัญญาณควบคุมจากคลื่นไฟฟ้ากล้ามเนื้อ
2. เพื่อพัฒนาขาเทียมสำหรับผู้พิการขาขาดให้มีประสิทธิภาพในราคาต้นทุนที่ต่ำที่สุดเพื่อให้ผู้พิการที่มีรายได้น้อยสามารถเข้าถึงบริการขาเทียมได้
3. เพื่อบูรณาการองค์ความรู้ที่ประกอบไปด้วย การเขียนโปรแกรม, การพัฒนาระบบควบคุม, ชีวกลศาสตร์ (Biomechanics), ปัจจัยมนุษย์ (Human Factors) และส่วนการเรียนรู้ของเครื่อง (Machine Learning) ในการพัฒนาขาเทียมและระบบควบคุม

1.3 ขอบเขตของปริญญานิพนธ์

1. ควบคุมการเคลื่อนไหวในขาเทียมโดยรับคลื่นไฟฟ้ากล้ามเนื้อจากกล้ามเนื้อ Biceps Femoris, Semitendinosus, Vartus Lateralis และ Vartus Medialis
2. สร้างโปรแกรมในการควบคุมการเคลื่อนไหวในขาเทียมประเภทเหนือเข่า (Above Knee Prosthesis) โดยใช้ไมโครคอนโทรลเลอร์ และประมวลผลโดยโปรแกรม Arduino

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ขาเทียมได้รับการพัฒนาองค์ความรู้ในการสร้างระบบควบคุมของขาเทียมต้นแบบประเภทเหนือเข้าโดยใช้สัญญาณควบคุมจากคลื่นไฟฟ้ากล้ามเนื้อ
2. เป็นประโยชน์กับผู้พิการที่สามารถควบคุมการเคลื่อนไหวของขาเทียมด้วยตนเองได้
3. ได้เรียนรู้ลักษณะของคลื่นไฟฟ้ากล้ามเนื้อที่เกิดขึ้นในระหว่างการเดิน
4. ได้เรียนรู้หลักการเขียนโปรแกรมและกระบวนการทำงานของโปรแกรม Arduino และโปรแกรม MATLAB



บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

ปริญญาบัตรฉบับนี้เป็นการออกแบบและพัฒนาต้นแบบขาเทียมควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ โดยทำการศึกษาและออกแบบต้นแบบขาเทียม รวมถึงเขียนโปรแกรมในการรับสัญญาณคลื่นไฟฟ้ากล้ามเนื้อ เพื่อใช้ในการควบคุมการเคลื่อนไหวในขาเทียม โดยทฤษฎีที่เกี่ยวข้องที่นำมาใช้กับปริญญาบัตร มีดังต่อไปนี้

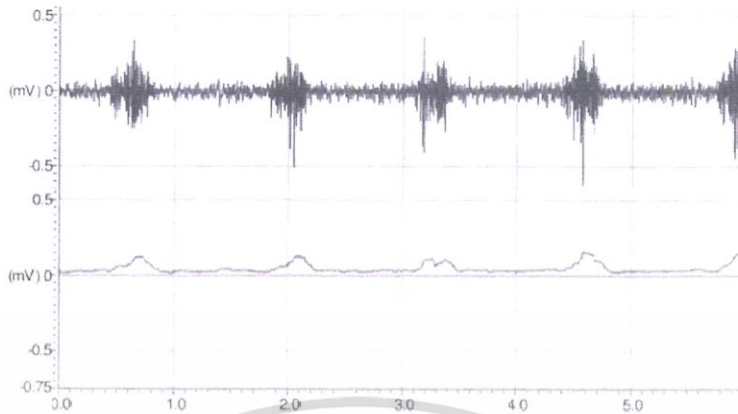
1. คลื่นไฟฟ้ากล้ามเนื้อที่สัมพันธ์กับการเดิน
2. วิวัฒนาการของขาเทียม
3. ไมโครคอนโทรลเลอร์และระบบควบคุม
4. โครงข่ายประสาทเทียม
5. งานวิจัยที่เกี่ยวข้อง

2.1 คลื่นไฟฟ้ากล้ามเนื้อที่สัมพันธ์กับการเดิน

2.1.1 คลื่นไฟฟ้ากล้ามเนื้อ

คลื่นไฟฟ้ากล้ามเนื้อ หมายถึง ระดับความตึงตัวของกล้ามเนื้อที่เกิดจากการกระตุ้นอย่างรุนแรง กล้ามเนื้อประกอบไปด้วยเส้นใยกล้ามเนื้อที่มีลักษณะเหมือนเส้นด้ายจำนวนมากหลายเส้น เมื่อเส้นใยกล้ามเนื้อ เกิดอาการเกร็ง จะทำให้เกิดกลไกการทำงานของเซลล์ประสาท (Motor Neuron) โดยจะส่งกระแสประสาท ไปยังเส้นใยกล้ามเนื้อนั้น ๆ ในทุกช่วงเวลา ถึงแม้ว่ากล้ามเนื้ออยู่ในขณะพักตัว การกระตุ้นที่เกิดขึ้นที่กล้ามเนื้อ นั้น จะถูกกระตุ้นจนกระทั่งเกิดการตอบสนองของกล้ามเนื้อแต่ละส่วนรวมเข้าด้วยกัน เพื่อทำให้เกิดการ ตอบสนองของกล้ามเนื้อทั้งหมด [8]

การเปลี่ยนแปลงในการส่งกระแสไฟฟ้ามาที่ผิวหนัง สามารถวัดได้ด้วยเครื่องวัดคลื่นไฟฟ้ากล้ามเนื้อที่ เรียกว่า Electromyography (EMG) ซึ่งเป็นเครื่องมือวัดที่มีความไวต่อการเปลี่ยนแปลงของกล้ามเนื้อใน ขณะที่กล้ามเนื้อเกิดอาการเกร็ง โดยลักษณะของคลื่นไฟฟ้ากล้ามเนื้อเป็นดังรูปที่ 2.1



รูปที่ 2.1 การปรับสัญญาณ EMG จาก Raw EMG signal มาเป็น Integrated EMG Signal [9]

เครื่องวัดคลื่นไฟฟ้ากล้ามเนื้อ (Electromyography หรือ Electromyogram) คือ เครื่องมือที่ใช้สำหรับบันทึกสัญญาณไฟฟ้าที่เกิดจากการทำงานของการหดตัวของกล้ามเนื้อ (Motor Unit) ซึ่งถูกควบคุมโดยเซลล์ประสาทสั่งการ (Motor Neuron) ที่อยู่บริเวณไขสันหลัง

เครื่องมือและอุปกรณ์สำหรับตรวจวัดคลื่นไฟฟ้ากล้ามเนื้อ ประกอบด้วยตัวเครื่องและอิเล็กโทรดบันทึกสัญญาณเข้า ตัวเครื่องมักมีอุปกรณ์หรือวงจรไฟฟ้าที่ทำหน้าที่กรองขยายสัญญาณและส่วนแสดงผล

อิเล็กโทรด หรือขั้วไฟฟ้าที่ใช้สำหรับตรวจวัดคลื่นไฟฟ้ากล้ามเนื้อ สามารถแบ่งได้เป็น 2 ชนิด คือ ขั้วไฟฟ้าสำหรับกระตุ้น และขั้วไฟฟ้าสำหรับบันทึกสัญญาณไฟฟ้า ขั้วไฟฟ้าสำหรับกระตุ้นมักเป็นตัวกระตุ้นขนาดเล็ก และเป็นชนิด 2 ขั้ว (Bipolar Electrode) โดยจะยึดติดกับส่วนอิเล็กโทรด ในขณะที่ขั้วไฟฟ้าสำหรับบันทึกสัญญาณไฟฟ้านั้น สามารถแบ่งออกเป็น 2 ชนิดย่อย ได้แก่ อิเล็กโทรดแบบเข็ม (Needle Electrode) มักใช้แทงเข้าไปในกล้ามเนื้อเพื่อบันทึกสัญญาณหรือศักย์ไฟฟ้าที่ใยกล้ามเนื้อโดยตรง มักใช้สำหรับวินิจฉัยโรคทางคลินิก และอิเล็กโทรดแบบวางที่ผิวหนัง (Surface Electrode) เป็นแผ่นขั้วไฟฟ้าที่วางบนผิวหนังบริเวณที่วัด มักนิยมใช้เพื่อบันทึกสัญญาณไฟฟ้ากล้ามเนื้อ เพื่อแสดงถึงความหนักเบาของการหดตัวของกล้ามเนื้อ



รูปที่ 2.2 ลักษณะของอิเล็กโทรดที่ใช้บันทึกสัญญาณ EMG
ชนิด Needle Electrode (เข็ม) และ Surface Electrode (ขวา) [10]

ส่วนขยายสัญญาณ (Amplifier) เนื่องจากสัญญาณไฟฟ้าจากเส้นประสาทและกล้ามเนื้อมีปริมาณน้อยและขนาดไม่มาก ดังนั้นที่ตัวเครื่องมักมีส่วนวงจรอิเล็กทรอนิกส์ เพื่อทำหน้าที่ขยายสัญญาณไฟฟ้าที่บันทึกได้ให้มีขนาดโตพอสำหรับการพิจารณา ส่วนที่ทำหน้าที่ขยายสัญญาณที่ดีควรมีกำลังขยายที่สูงพอและสม่ำเสมอตลอดช่วงศักย์ไฟฟ้าในย่านที่ตรวจวัด และสามารถกรองสัญญาณหรือตัดสัญญาณไฟฟ้ารบกวนที่ไม่ได้ใช้ออก

ส่วนแสดงและบันทึกผล เนื่องจากสัญญาณไฟฟ้าจากเส้นประสาทและกล้ามเนื้อมีความถี่ที่ตอบสนองค่อนข้างสูง มักจะแสดงผ่านจอภาพออสซิลโลสโคป (ตอบสนองต่อความถี่ได้สูง) และผ่านสัญญาณเสียง ทำให้การแปลผลมีความเที่ยงตรงมากขึ้นและลดความเมื่อยล้าจากคู่อสัญญาณไฟฟ้าบนจอออสซิลโลสโคปด้วยสายตานอกจากนั้น ในปัจจุบันยังสามารถบันทึกผลออกมาเป็นไฟล์ข้อมูล รูปภาพและค่าเฉลี่ยของขนาดของคลื่นไฟฟ้าได้อีกด้วย [11]

2.1.2 สรีระของกล้ามเนื้อ

การเคลื่อนไหวของมนุษย์เกิดจากการหดตัวของกล้ามเนื้อบริเวณแขน ขาและลำตัว โดยจะทำงานร่วมกับระบบโครงกระดูกและระบบประสาท ในการเดินของมนุษย์นั้นเกิดจากการทำงานของกล้ามเนื้อบริเวณขา (Muscles of the Lower Extremities) ซึ่งสามารถจำแนกออกเป็นกล้ามเนื้อของสะโพก (Muscle of the Gluteal Region) กล้ามเนื้อของต้นขา (Muscle of the Thigh) กล้ามเนื้อของปลายขา (Muscle of the Leg) และกล้ามเนื้อของข้อเท้า (Muscle of the Foot) โดยในปริณิญาฉบับนี้จะกล่าวเฉพาะกล้ามเนื้อของต้นขา (Muscle of the Thigh) [12]

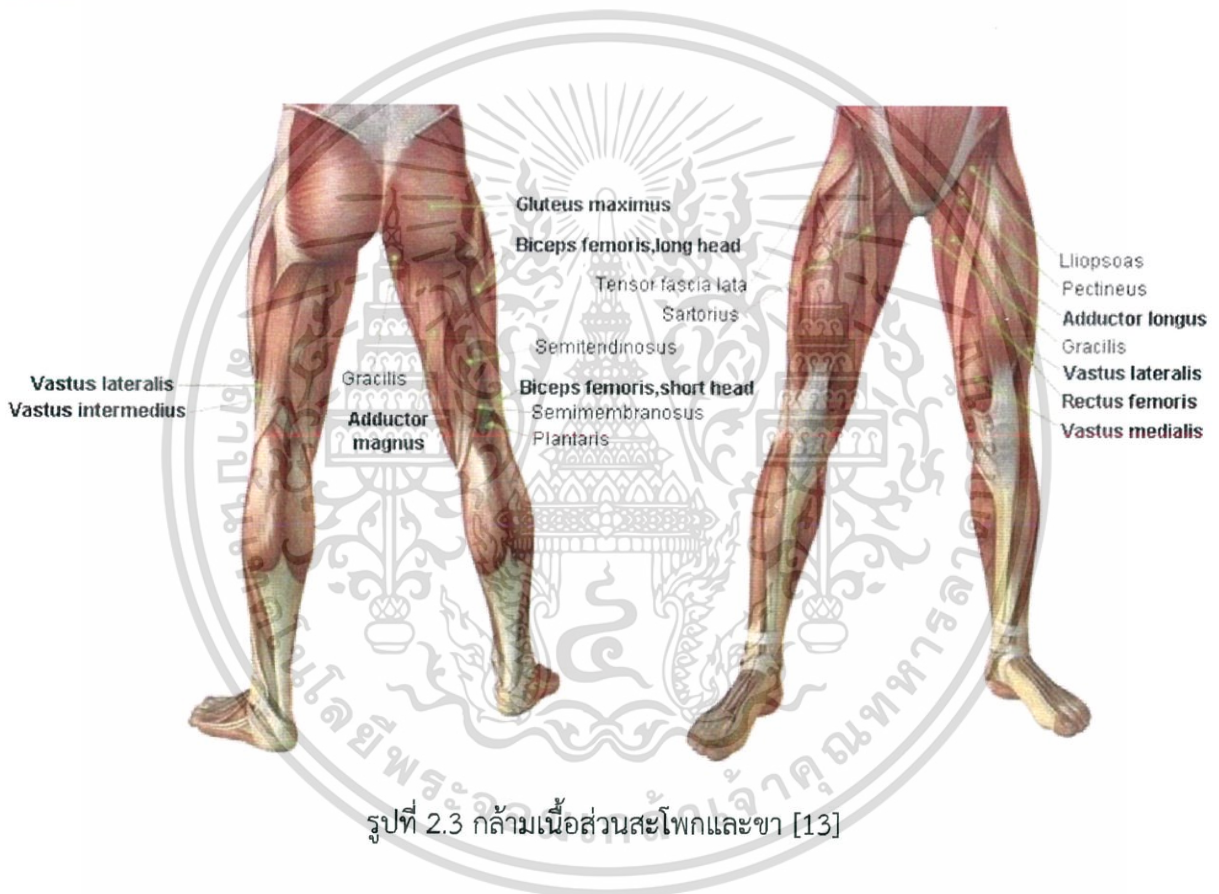
กล้ามเนื้อของต้นขาจะแบ่งออกเป็นกลุ่มหลัก ดังต่อไปนี้ คือ กล้ามเนื้อทางด้านหน้าของต้นขา มีหน้าที่เหยียดปลายขา กล้ามเนื้อด้านในของต้นขา มีหน้าที่หุบต้นขา และกล้ามเนื้อด้านหลังของต้นขา มีหน้าที่งอปลายขา กล้ามเนื้อทางด้านหน้าของต้นขา (Anterior Femoral Muscles) มีดังนี้

1) Quadriceps Femoris มี 4 มัดคือ Rectus Femoris, Vastus Lateralis, Vastus Medialis และ Vastus intermedius โดยกล้ามเนื้อทั้ง 4 มัดนี้รวมกันเป็น Tendon ทอดข้ามไปข้างหน้าหุ้มเข้าไปเกาะที่ Tuberosity ของกระดูก Tibia Tendon ของกล้ามเนื้อมัดนี้เป็นที่ที่กระดูก Patella เกิดขึ้น ซึ่งทำหน้าที่เหยียด (Extend) ปลายขา นอกจากนี้ กล้ามเนื้อ Rectus Femoris ยังช่วยงอ (Flex) ต้นขาอีกด้วย

2) Sartorius เป็นกล้ามเนื้อแผ่นยาว แคบ ตั้งต้นจาก Anterior Superior Iliac Spine ผ่านเหยียดจากด้านนอก และมี Insertion ที่ส่วนบนตอนกลางของ Tibia ครอบคลุมอยู่ที่เกาะของกล้ามเนื้อ Gracillis หน้าที่คือ งอต้นขา กางต้นขา และหมุนต้นขาไปข้างๆ รวมถึงงอและหมุนปลายขาเข้าข้างใน

กล้ามเนื้อด้านในของต้นขา (Medial Femoral Muscles) มีอยู่ 4 มัดคือ Gracillis, Adductor Longus, Adductor Brevis และ Adductor Magnus โดยมีหน้าที่คือ กล้ามเนื้อ Gracillis ทำหน้าที่หุบต้นขาและงอปลายขา ส่วนกล้ามเนื้ออีก 3 มัดทำหน้าที่ Adduct, Flex และ Rotate ต้นขาออกข้างนอก

กล้ามเนื้อด้านหลังของต้นขา (Posterior Femoral Muscles) หรือเรียกว่า Hamstring Muscles มีอยู่ 3 มัด คือ Biceps Femoris, Semitendinosus และ Semimembranosus ซึ่งมีหน้าที่คือ งอปลายขาขึ้นมาและเหยียดต้นขา เมื่ออหิวเข้ากล้ามเนื้อ Semitendinosus และ Semimembranosus จะหมุนปลายขาเข้าด้านใน

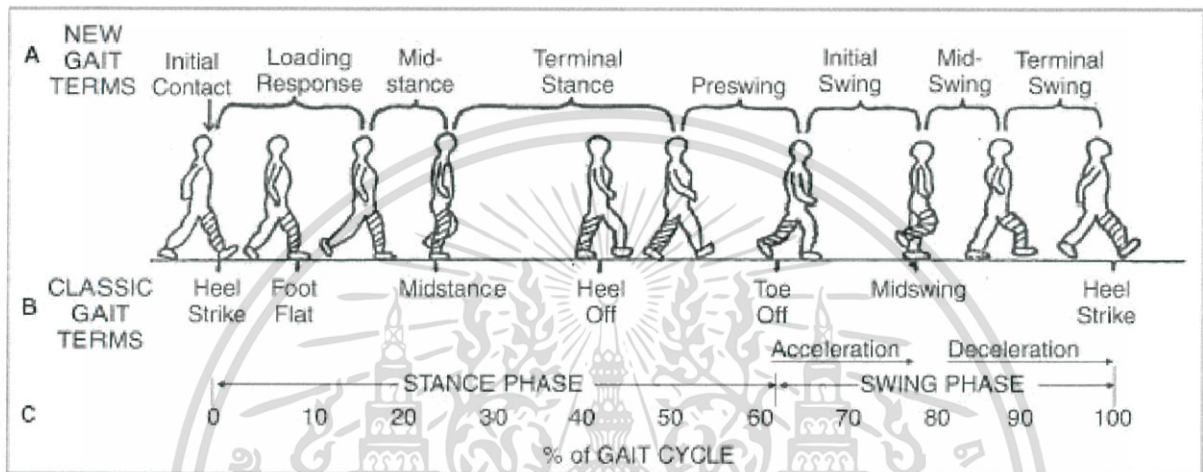


รูปที่ 2.3 กล้ามเนื้อส่วนสะโพกและขา [13]

2.1.3 วงจรของการเดิน (Gait Cycle)

การเดินเป็นการเคลื่อนไหวที่สำคัญในการกิจวัตรประจำวันของมนุษย์ ซึ่งนับว่าเป็นการเคลื่อนไหวที่ซับซ้อนเนื่องจากอาศัยการทำงานร่วมกันระหว่างระบบต่าง ๆ ในร่างกาย เช่น ระบบกระดูกและกล้ามเนื้อ ระบบประสาท ระบบการไหลเวียนโลหิต เป็นต้น โดยรูปแบบการเดินของแต่ละบุคคลจะมีความแตกต่างกันไปตามปัจจัยต่าง ๆ ที่เกี่ยวข้อง เช่น อายุ เพศ สถานการณ์ที่เผชิญอยู่ หรือแม้กระทั่งความผิดปกติของระบบร่างกายที่เกี่ยวข้องกับการเดิน [14]

รูปแบบการเคลื่อนไหวของการเดิน เป็นการเคลื่อนไหวที่มีลักษณะการเคลื่อนที่อย่างต่อเนื่อง โดยใช้ขาทั้งสองข้างเคลื่อนไหวสลับกันอย่างเป็นจังหวะเพื่อทำให้ร่างกายเกิดการเคลื่อนที่ไปข้างหน้า และช่วยพยุงน้ำหนักของร่างกาย โดยวงจรของการเดิน 1 รอบนั้น จะสังเกตได้จากการที่เท้าสัมผัสพื้น (Stance Phase) ยกปลายขึ้นจากพื้น (Swing Phase) และวนกลับมาสัมผัสพื้นใหม่อีกครั้งหนึ่ง ดังแสดงในรูปที่ 2.4



รูปที่ 2.4 วงจรการเดิน (Gait Cycle) [15]

การแบ่งลักษณะของวงจรการเดิน (Gait Cycle) โดยอ้างอิงจากหลักการของ Rancho Los Amigos และ Traditional Gait Cycle สามารถแบ่งออกเป็น 2 ช่วง ดังต่อไปนี้

1) Stance Phase เป็นช่วงที่เท้าสัมผัสพื้นและมีการรับน้ำหนักตัวของร่างกาย โดยกล้ามเนื้อที่ทำงานส่วนใหญ่เป็นกล้ามเนื้อในกลุ่มเหยียด (Extensor Group) โดยสามารถแบ่งการทำงานออกเป็น 5 ช่วงย่อย ได้ดังต่อไปนี้

1.1 Heel Strike เป็นระยะแรกของ Stance Phase ซึ่งเป็นระยะที่เท้าเริ่มสัมผัสพื้น โดยใช้ส้นเท้าเป็นจุดสัมผัสแรก ข้อสะโพกจะอยู่ในท่างอและค่อยๆเหยียดออกจนเข้าสู่ระยะ Foot Flat โดยที่ข้อเข่าจะยังคงอยู่ในลักษณะเหยียด และข้อเท้าจะอยู่ในลักษณะกระดูกขึ้น อย่างไรก็ตาม ระยะนี้สามารถเรียกอีกชื่อหนึ่งว่า Initial Contact ซึ่งกล้ามเนื้อที่มีการทำงานในระยะ Heel Strike นี้ คือ

- กล้ามเนื้อ Quadriceps ทำหน้าที่ให้เข่าเหยียด
- กล้ามเนื้อ Gluteu Maximus ทำหน้าที่เหยียดสะโพกเพื่อให้ส้นเท้ากดกับพื้น
- กล้ามเนื้อ Hamstrings ทำหน้าที่ช่วยเหยียดสะโพกเพื่อให้ส้นเท้ากดกับพื้น
- กล้ามเนื้อกลุ่มกระดูกข้อเท้าขึ้น ทำให้เท้ากระดูกพ่นพื้น

1.2 Foot Flat เป็นระยะของการเดินที่เท้าแตะพื้น ซึ่งเกิดขึ้นทันทีที่ส้นเท้าสัมผัสกับพื้น โดยระยะนี้เท้าจะวางราบกับพื้น ในช่วงนี้ปลายเท้าจะค่อยๆวางลงสัมผัสกับพื้นจนกระทั่งวางเต็มพื้นที่ฝ่าเท้า ข้อสะโพกยังคงเหยียดออก ในขณะที่ข้อเข่าจะเริ่มงอเล็กน้อยเพื่อเตรียมรับน้ำหนักตัว ส่วนข้อเท้าจะค่อยๆวางลงกับพื้น และกระดูกปลายขาเริ่มที่จะเคลื่อนที่มาจากด้านหน้า ร่วมกับการเคลื่อนลำตัวมาทางด้านหน้า ระยะ Foot Flat เป็นช่วงแรกของวงจรการเดินที่พบว่าเท้าทั้ง 2 ข้างสัมผัสพื้นพร้อมกัน (Double Support) โดยระยะนี้ จะมีการทำงานของกล้ามเนื้อข้อศอกเดียวกันกับระยะ Heel Strike ยกเว้นเพียงแต่ข้อเท้าอยู่ในท่ากระดูกง 15 องศา

1.3 Mid Stance เป็นระยะของการเดินที่มีการเคลื่อนไหวของลำตัวมาด้านหน้ามากขึ้น ทำให้น้ำหนักตัวตกลงในแนวศูนย์กลางมวลของร่างกาย ผ่านหลัง ข้อสะโพก ข้อเข่าและข้อเท้า เป็นระยะที่ขาข้างที่เท้าวางราบกับพื้นรับน้ำหนักตัวทั้งหมด ส่วนขาข้างที่อยู่ทางด้านตรงข้ามจะลอยพ้นจากพื้น ในระยะนี้มุมของข้อสะโพกจะเท่ากับ 0 องศา ข้อเข่าจะงอเล็กน้อยประมาณ 10-20 องศา ส่วนข้อเท้าจะกระดูกงขึ้นประมาณ 2 องศา โดยกล้ามเนื้อที่มีการทำงานในระยะนี้คือ

- กล้ามเนื้อ Iliopsoas ทำหน้าที่ให้ข้อสะโพกงอ เพื่อป้องกันไม่ให้เกิด Hyperextension
- กล้ามเนื้อ Gluteus Medius , Gluteus Minimus และ Tensor Fascia Lata ทำหน้าที่ให้กระดูกเชิงกรานอยู่ในระดับ ป้องกันไม่ให้เอียงไปด้านใดด้านหนึ่ง และไม่ให้ร่างกายเสียสมดุลง่าย
- กล้ามเนื้อ Quadriceps ทำหน้าที่ไม่ให้เข่างอพับลงมา
- กล้ามเนื้อกระดูกเท้า ทำหน้าที่ให้ฝ่าเท้าแนบกับพื้น

1.4 Heel Off เป็นระยะที่จะเริ่มเตรียมพร้อมเข้าสู่ช่วง Stance Phase โดยมีการยกส้นเท้าสูงขึ้นจากพื้น ซึ่งลำตัวจะเอนไปข้างหน้ามากขึ้น ในระยะนี้ข้อสะโพกจะเริ่มเหยียดมากขึ้น ส่วนข้อเข่าจะเริ่มงอ และข้อเท้าจะเริ่มงอขึ้นจนกระทั่งเข้าสู่ช่วง Toes Off โดยระยะ Heel Off เป็นระยะสุดท้ายที่มีการพยุ่งร่างกายโดยขาเพียงข้างเดียว กล้ามเนื้อที่ทำงานในระยะนี้ มีดังนี้

- กล้ามเนื้อ Iliopsoas ทำหน้าที่ป้องกันไม่ข้อสะโพกเกิด Hyperextension
- กล้ามเนื้อ Hamstring ทำหน้าที่งอเข่า
- กล้ามเนื้อกระดูกเท้าลง ทำหน้าที่กระดูกข้อเท้าลง

1.5 Toes Off เป็นระยะที่ปลายเท้าดันให้ร่างกายเคลื่อนที่ไปข้างหน้าและเพื่อให้ขาลอยพ้นจากพื้น โดยข้อสะโพกและข้อเข่าจะเริ่มงอมากขึ้น ส่วนข้อเท้าจะเริ่มกดปลายเท้าลง กล้ามเนื้อที่พบการทำงานในระยะนี้ คือ

- กล้ามเนื้อ Iliopsoas ทำหน้าที่งอข้อสะโพกเพื่อเหวี่ยงขาไปข้างหน้า
- กล้ามเนื้อ Quadriceps ทำหน้าที่ป้องกันการงอเข่าเร็วเกินไป

- กล้ามเนื้อกระดูกเท้าลง ทำหน้าที่ดันให้ขาเหวี่ยงไปข้างหน้า

2) Swing Phase เป็นช่วงที่ขาถอยพ้นจากพื้นและแกว่งไปทางด้านหน้า มักพบการทำงานของกล้ามเนื้อในกลุ่มของกล้ามเนื้อขา สามารถแบ่งออกเป็น 3 ระยะย่อยได้ดังนี้

2.1 Acceleration Phase คือระยะเร่ง เป็นระยะที่เท้ายกขึ้นพ้นจากพื้นโดยอาศัยแรงดันจากช่วง Toes Off ของ Stance Phase โดยข้อสะโพกและข้อเข่าจะเริ่มงอเพิ่มขึ้น ซึ่งเข่าจะเหวี่ยงไปข้างหน้าด้วยความเร็วจากแรงเฉื่อยที่มาร่ง ส่วนข้อเท้าจะอยู่ในลักษณะกระดูกเพิ่มมากขึ้น สามารถเรียกอีกชื่อหนึ่งว่าระยะ Initial Swing โดยชุดกล้ามเนื้อที่ทำงานในระยะนี้ คือ

- กล้ามเนื้อ Iliopsoas ทำหน้าที่เหวี่ยงและงอข้อสะโพกไปข้างหน้า
- กล้ามเนื้อ Quadriceps ทำหน้าที่ป้องกันการงอเข่ามากเกินไป
- กล้ามเนื้อกระดูกเท้าขึ้นทำให้ข้อเท้ากระดูกขึ้นเพื่อไม่ให้ปลายเท้าแตะพื้นกันล้ม

2.2 Mid Swing คือระยะแกว่งขา เป็นระยะที่ขาถูกดึงให้งอและลอยมาอยู่ใต้ลำตัว โดยกล้ามเนื้อที่มีการทำงานเป็นกล้ามเนื้อชุดเดียวกับระยะ Acceleration

2.3 Deceleration Phase คือระยะลดอัตราเร่ง เป็นระยะการเดินที่ขาที่ถูกเหวี่ยงไปข้างหน้าจากแรงกล้ามเนื้อที่กล่าวมาข้างต้นถูกชะลอให้ช้าลงเพื่อเข้าสู่ Stance Phase อีกครั้งเป็นวงจรการเดินใหม่ต่อไป เรียกระยะนี้ชื่อหนึ่งว่า Terminal Swing

ในแต่ละช่วงของการเดิน สามารถสรุปการทำงานของข้อต่าง ๆ และกล้ามเนื้อ ได้ดังตารางที่ 2.1 ซึ่งจะแสดงค่ามุมของสะโพก (Hip) ข้อเข่า (Knee) และข้อเท้า (Ankle) ต่าง ๆ โดยมุมที่ข้อสะโพกเป็นค่ามุมที่ได้จากการเทียบกับแนวแกนตั้ง ส่วนมุมข้อเข่าเป็นมุมระหว่างขาท่อนบนและท่อนล่าง และมุมของข้อเท้าเป็นมุมที่เปลี่ยนแปลงไปจากมุมปกติหรือมุมตั้งฉากระหว่างขาท่อนล่างและฝ่าเท้า จะเห็นได้ว่าค่ามุมข้อสะโพก (Hip Angle) มีมุมงอ (Flex) ตั้งแต่ 25 องศา ถึง -25 องศา เมื่อเทียบกับแนวตั้ง โดยแบ่งเป็น 2 ช่วง ช่วงแรกคือช่วงที่ขาท่อนบน อยู่หน้าลำตัว (ระยะ Heel Strike, Foot Flat, Mid Stance และ Mid Swing, Deceleration) ในช่วงนี้ค่ามุมจะเป็นบวก ส่วนช่วงที่สอง คือ ช่วงที่ขาท่อนบนอยู่ข้างหลังลำตัว (ระยะ Heel off เมื่องอไปข้างหลังเกิน 10 องศา, Toe Off และ Acceleration) ค่ามุมข้อเข่า (Knee Angle) จะมีมุมงอตั้งแต่ 0 องศา (Fully Extend) จนถึง 65 องศา และค่ามุมข้อเท้า (Ankle Angle) มีค่ามุมปกติ มุมที่ข้อเท้ากางออก (Plantar Flex) และมุมทึงข้อเท้า (Dorsiflex)

ตารางที่ 2.1 การทำงานของข้อต่าง ๆ และกล้ามเนื้อในขณะเดิน [16]

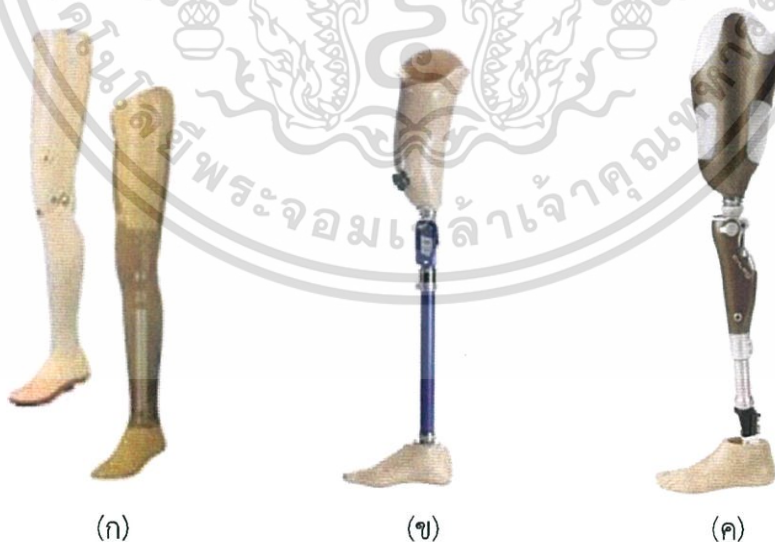
วงจรการเดิน	มุมสะโพก (องศา)	มุมข้อเข่า (องศา)	มุมข้อเท้า (องศา)	กล้ามเนื้อที่ใช้ในขณะเดิน
Heel Strike	งอ 25 องศา	เหยียดตรง	อยู่แนวเดียวกับ ข้อเข่า	Gluteus Maximus, Hamstrings
Foot Flat	งอ 23 องศา	งอ 20 องศา	กระดกลง 15 องศา	Gluteus Maximus, Hamstrings
Mid Stance	งอ 10 องศา	งอ 10 องศา	กระดกขึ้น 3 องศา	Iliopsoas, Gluteus Minimus, Quadriceps
Heel Off	เหยียดเกิน 10 องศา	งอ 2 องศา	กระดกขึ้น 15 องศา	Iliopsoas, Hamstrings
Toes Off	งอ 10 องศา	งอ 40 องศา	กระดกลง 20 องศา	Iliopsoas, Quadriceps
Acceleration	งอ 5 องศา	งอ 65 องศา	อยู่แนวเดียวกับ ข้อเข่า	Iliopsoas, Quadriceps
Mid Swing	งอ 25 องศา	งอ 65 องศา	อยู่แนวเดียวกับ ข้อเข่า	Iliopsoas
Deceleration	งอ 25 องศา	เหยียดตรง	อยู่แนวเดียวกับ ข้อเข่า	Gluteus maximus, Hamstrings

2.2 วิวัฒนาการของขาเทียม

ขาเทียม (Prosthetic Leg) คือ ภายอุปกรณ์ชนิดหนึ่ง ซึ่งมีกลไกและการออกแบบให้มีลักษณะคล้ายกับขาจริง เพื่อคอยช่วยเหลือประคับประคองการทรงตัว การเดิน และการเคลื่อนไหวของผู้ป่วยที่ต้องเสียขาไปจากอุบัติเหตุหรือการเจ็บป่วยต่าง ๆ โดยขาเทียมอาจทำมาจากวัสดุที่แตกต่างหลากหลาย บ้างก็ถูกผลิตให้มีลักษณะคล้ายกับขาจริง โดยทั้งหมดล้วนมีเป้าหมายเพื่อให้ผู้ป่วยสามารถดำเนินชีวิตและทำกิจวัตรประจำวันด้วยตนเองต่อไปได้แม้หลังการสูญเสีย

ประเภทของขาเทียมออกเป็น 2 ประเภทหลัก คือ ขาเทียมแกนนอก (Exoskeleton Prosthesis) และขาเทียมแกนใน (Endoskeleton Prosthesis) โดยขาเทียมแกนนอกมีระบบการทำงานแบบควบคุมโดย

ใช้การเคลื่อนไหวของร่างกาย (Body Controlled) เป็นระบบขาเทียมใช้ไม้ หรือโฟมอัดแน่นเป็นแกนขา และหุ้มด้วยพลาสติกเรซิน เพื่อป้องกันการสึกกร่อนและมีความสวยงามคล้ายขาจริง น้ำหนักค่อนข้างมาก แต่มีราคาถูก [3] ส่วนขาเทียมแกนใน สามารถแบ่งย่อยตามลักษณะการทำงานออกเป็น 3 กลุ่มย่อย ได้แก่ ขาเทียมแกนในแบบควบคุมโดยใช้การเคลื่อนไหวของร่างกาย (Body Controlled) เป็นขาเทียมใช้แกนขาเป็นโลหะหรือพลาสติกเป็นแกน มีน้ำหนักเบา และมีราคาสูงกว่าขาเทียมแกนนอก ขาเทียมแกนในแบบควบคุมโดยใช้พลังงานภายนอก (External-Power Controlled) โดยแบ่งย่อยได้อีกเป็นขาเทียมที่ควบคุมด้วยระบบความดันน้ำหรือน้ำมัน (Fluid หรือ Hydraulic Controlled) ซึ่งต้องอาศัยการเคลื่อนไหวของร่างกายร่วมด้วย จะช่วยให้ผู้ป่วยเดินหรือเคลื่อนไหวขาได้เร็วขึ้น และขาเทียมที่ควบคุมด้วยระบบไมโครโพรเซสเซอร์ (Microprocessor) ซึ่งเป็นอุปกรณ์ประมวลผลอิเล็กทรอนิกส์ขนาดเล็ก ที่ใช้เซ็นเซอร์ควบคุมปรับการเคลื่อนไหวของข้อต่อ เพื่อให้ผู้ป่วยสามารถเดินได้คล่องตัวขึ้น และมีความเสี่ยงจากการพลาดหกล้มลดลง เช่น ขาเทียมซี-เล็ก มีชุดควบคุมอิเล็กทรอนิกส์ มีวงจรมิโครโพรเซสเซอร์เป็นหัวใจสำคัญของระบบ ทำหน้าที่รับและประมวลผลสัญญาณจากชุดเซ็นเซอร์และควบคุมการเคลื่อนไหวของข้อเท้าแบบเรียลไทม์ มีชุดเซ็นเซอร์ข้อเท้า ตรวจสอบการเคลื่อนไหว และความเร็วข้อเท้าก่อนส่งข้อมูลให้ไมโครโพรเซสเซอร์ประมวลผลและสั่งการ มีระบบไฮดรอลิกควบคุมโดยไมโครโพรเซสเซอร์ ปรับระดับแรงต้านในการเหยียดและงอขาให้เหมาะสมในทุกช่วงจังหวะการเดิน มีน้ำหนักเบา และช่วยให้สามารถทำกิจกรรมได้ใกล้เคียงกับคนปกติมากที่สุด แต่มีราคาสูงมาก [6]



รูปที่ 2.5 ประเภทของขาเทียม (ก) ขาเทียมแกนนอก (ข) ขาเทียมแกนในแบบควบคุมโดยใช้การเคลื่อนไหวของร่างกาย (ค) ขาเทียมแกนในควบคุมด้วยระบบไมโครโพรเซสเซอร์ [17]

2.3 ไมโครคอนโทรลเลอร์และระบบควบคุม

2.3.1 ไมโครคอนโทรลเลอร์ (Microcontroller)

ไมโครคอนโทรลเลอร์ (Microcontroller) คือ ชิปปูอุปกรณ์อิเล็กทรอนิกส์ประมวลผลอย่างหนึ่ง ทำหน้าที่ประมวลผลตามโปรแกรมหรือชุดคำสั่ง โครงสร้างภายในเป็นวงจรรวมขนาดใหญ่ ประกอบไปด้วย หน่วยคำนวณทางคณิตศาสตร์และลอจิก บัสต่างๆ พอร์ต รีจิสเตอร์ หน่วยความจำ วงจรนับและวงจรจับเวลารวมกันอยู่ภายในชิป ไมโครคอนโทรลเลอร์ถูกออกแบบมาเพื่อใช้ในงาน ควบคุมสามารถติดต่อกับอุปกรณ์ อินพุตและเอาต์พุตได้ สะดวกใช้งานง่าย สามารถทำงานได้โดยใช้ชิปเดียว สามารถเข้าถึงข้อมูลระดับบิตได้ [18] โปรแกรมควบคุมไมโครคอนโทรลเลอร์ มีภาษาที่ใช้ระดับสูง หลายภาษา ทำให้ง่ายต่อการศึกษาเรียนรู้ ออกแบบวงจรพัฒนาระบบ ข้อแตกต่างระหว่างไมโครโปรเซสเซอร์กับไมโครคอนโทรลเลอร์คือ ไมโครคอนโทรลเลอร์ สามารถติดต่อกับอุปกรณ์อินพุตและเอาต์พุตได้สะดวกกว่า สามารถเข้าถึงข้อมูลระดับบิตและมีภาษาควบคุมระดับสูงหลายภาษาทำให้งานต่อการศึกษาเรียนรู้ โดยไมโครคอนโทรลเลอร์มีข้อดีเหมาะสมต่อการใช้งานควบคุมหลายประการ เช่น

- (1) ชิปป็อบและระบบที่ได้มีขนาดเล็ก
- (2) ระบบที่ได้มีราคาถูกกว่าการใช้ชิปไมโครโปรเซสเซอร์
- (3) วงจรที่ได้จะมีความซับซ้อนน้อย ช่วยลดข้อผิดพลาดที่อาจเกิดขึ้นได้ในการต่อวงจร
- (4) มีคุณสมบัติเพิ่มเติมสำหรับงานควบคุมโดยเฉพาะซึ่งใช้งานได้ง่าย
- (5) ช่วยลดระยะเวลาในการพัฒนาระบบได้
- (6) ไมโครคอนโทรลเลอร์มีหลายประเภทด้วยกัน ซึ่งแต่ละประเภทก็จะมีโครงสร้างภายในและ

ความสามารถในการทำงานที่แตกต่างกัน ทำให้สามารถเลือกใช้กับงานได้อย่างเหมาะสม โครงสร้างเบื้องต้นของไมโครคอนโทรลเลอร์ ประกอบไปด้วยส่วนประกอบต่าง ๆ ดังนี้

1) หน่วยประมวลผลกลางหรือซีพียู (CPU: Central Processing Unit) ทำหน้าที่ประมวลผลข้อมูลตามคำสั่งโปรแกรม ส่วนรีจิสเตอร์จะเก็บข้อมูลและกำหนดการทำงานของไมโครคอนโทรลเลอร์ การทำงานของซีพียูมี 2 จังหวะคือ เฟตช์ (Fetch) และ เอ็กซีคิวต์ (Executed) เริ่มจากการเฟตช์ คือการอ่านคำสั่งแล้วทำการถอดรหัสคำสั่งเป็นภาษาเครื่อง จากนั้นจังหวะเอ็กซีคิวต์ กระทำตามคำสั่งโปรแกรมจนเสร็จ

2) หน่วยความจำโปรแกรม (Program Memory) ทำหน้าที่เก็บโปรแกรมเพื่อส่งให้กับซีพียู ทำการประมวลผล ที่นิยมจะมี 3 แบบคือ

2.1 แบบอีพรอม (EPROM: Erasable Programmable Read Only Memory) มี 2 ประเภทคือ ประเภทที่โปรแกรมได้ครั้งเดียวและโปรแกรมได้หลายครั้ง การลบต้องใช้แสงอุลตราไวโอเล็ต ถ้าเป็นประเภทโปรแกรมได้ครั้งเดียวจะไม่สามารถลบและโปรแกรมใหม่ได้

2.2 แบบอีอีพรอม (EEPROM: Electrically Erasable Programmable Read Only Memory) เป็นหน่วยความจำที่ลบและเขียนใหม่ได้ด้วยไฟฟ้า ปัจจุบันไม่นิยมเพราะมีราคาแพง

2.3 แบบแฟลช (Flash) หน่วยความจำชนิดนี้สามารถลบและเขียนใหม่ได้ด้วยสัญญาณไฟฟ้าเป็นที่นิยมมากเพราะราคาไม่แพงขนาดของหน่วยความจำโปรแกรมขึ้นอยู่กับตระกูลและเบอร์ของไมโครคอนโทรลเลอร์ว่ากี่บิตและมีขนาดความจุเท่าไร เช่น ใน PIC16F877 มีหน่วยความจำโปรแกรม 8192 x14 บิต

3) หน่วยความจำข้อมูลแรม (Data Memory) ทำหน้าที่เก็บข้อมูลจากการประมวลผลของซีพียู

4) หน่วยความจำข้อมูล EEPROM เป็นหน่วยความจำที่มีไมโครคอนโทรลเลอร์บางรุ่นทำหน้าที่เก็บข้อมูลไว้แม้ไม่มีไฟเลี้ยง การอ่านและเขียนจะใช้สัญญาณไฟฟ้า

5) รีจิสเตอร์พอร์ต (Register Space) เป็นหน่วยความจำพิเศษสามารถอ่านเขียนได้ตลอดเวลา ทำหน้าที่เก็บข้อมูลในการทำงานของไมโครคอนโทรลเลอร์ แสดงสถานะของการทำงานกำหนดการทำงานของไมโครคอนโทรลเลอร์ เป็นขาพอร์ตติดต่อสื่อสารข้อมูลที่รับเข้ามาทางขาอินพุตและส่งข้อมูล

6) วงจรสัญญาณนาฬิกา ทำหน้าที่ควบคุมจังหวะการทำงานของไมโครคอนโทรลเลอร์ [19]

2.3.2 บอร์ด Arduino

Arduino เป็นบอร์ดไมโครคอนโทรลเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลด้าน Hardware และ Software ตัว บอร์ด Arduino ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ความง่ายของบอร์ด Arduino ในการต่ออุปกรณ์เสริมต่าง ๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิกส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขา I/O ของบอร์ด หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม (Arduino Shield) ประเภทต่างๆ เช่น Arduino XBee Shield, Arduino Music Shield, Arduino Relay Shield, Arduino Wireless Shield, Arduino GPRS Shield เป็นต้น มาเสียบกับบอร์ดบนบอร์ด Arduino แล้วเขียนโปรแกรมพัฒนาต่อได้เลย [18]

จุดเด่นที่ทำให้บอร์ด Arduino เป็นที่นิยมคือ ง่ายต่อการพัฒนา มีรูปแบบคำสั่งพื้นฐาน ไม่ซับซ้อนมี Arduino Community กลุ่มคนที่ร่วมกันพัฒนาที่แข็งแกร่ง Open Hardware ทำให้ผู้ใช้สามารถนำบอร์ดไปต่อยอดใช้งานได้หลายด้านราคาไม่แพง Cross Platform สามารถพัฒนาโปรแกรมบน OS ใดก็ได้

ขั้นตอนการเขียนโปรแกรมบน Arduino มีดังนี้

1) เขียนโปรแกรมบนคอมพิวเตอร์ ผ่านทางโปรแกรม ArduinoIDE

2) หลังจากที่เราเขียนโค้ดโปรแกรมเรียบร้อยแล้ว ให้ผู้ใช้งานเลือกรุ่นบอร์ด Arduino ที่ใช้และหมายเลข Com port

3) กดปุ่ม Verify เพื่อตรวจสอบความถูกต้องและ Compile โค้ดโปรแกรม จากนั้นกดปุ่ม Upload โค้ด โปรแกรมไปยังบอร์ด Arduino ผ่านทางสาย USB เมื่ออัปโหลดเรียบร้อยแล้วจะแสดงข้อความแถบข้างล่าง “Done uploading” และบอร์ดจะเริ่มทำงานตามที่เขียนโปรแกรม

2.3.3 ระบบควบคุมพีไอดี (PID Controller)

ระบบควบคุมพีไอดี (PID Controller) เป็นระบบควบคุมแบบป้อนกลับที่ใช้กันอย่างกว้างขวางในภาคอุตสาหกรรม ค่าที่ใช้ในการคำนวณคือความผิดพลาดที่เกิดจากความแตกต่างระหว่างผลตอบสนอง และสัญญาณอ้างอิงหรือจุดปรับตั้งของระบบวงปิด (ระบบป้อนกลับ) ตัวควบคุมดังกล่าวจะลดความผิดพลาดให้เหลือน้อยที่สุดด้วยการปรับเปลี่ยนสัญญาณควบคุม (สัญญาณขาเข้า) ของกระบวนการ อัตราขยายของตัวควบคุมพีไอดีที่ใช้งานจะขึ้นอยู่กับคุณลักษณะของกระบวนการ [20]

ระบบควบคุมพีไอดีประกอบด้วยเทอมของหน่วยสัดส่วน (Proportional) หน่วยปริพันธ์ (Integral) และหน่วยอนุพันธ์ (Derivative) ของความผิดพลาด หน่วยทั้งสามนี้จะดำเนินการทางคณิตศาสตร์กับความผิดพลาดแบบแยกส่วนกันและนำผลการคำนวณที่ได้มารวมเข้าด้วยกัน เพื่อเปลี่ยนแปลงสัญญาณควบคุมในโดเมนเวลาอย่างเหมาะสม ทำให้เกิดกลไกการควบคุมอัตโนมัติ ซึ่งกลไกปฏิบัติการทางคณิตศาสตร์ในระบบควบคุมพีไอดี เป็นไปดังสมการที่ 2.1

$$MV(t) = P_{out} + I_{out} + D_{out} \quad (2.1)$$

เมื่อ P_{out} , I_{out} และ D_{out} เป็นผลของสัญญาณขาออกจากระบบควบคุมพีไอดีจากแต่ละเทอม ซึ่งนิยามตามรายละเอียดดังต่อไปนี้

1) หน่วยสัดส่วน (Proportional)

หน่วยสัดส่วน (Proportional) หรืออัตราขยาย จะได้รับความผิดพลาดระหว่างสัญญาณอ้างอิงและผลตอบสนอง โดยการตอบสนองของสัดส่วนสามารถทำได้โดยการคูณค่าความผิดพลาดด้วยอัตราขยายสัดส่วน (K_p) ซึ่งหน่วยสัดส่วนแสดงดังสมการที่ 2.2

$$P_{out} = K_p e(t) \quad (2.2)$$

- เมื่อ P_{out} คือ สัญญาณขาออกของหน่วยสัดส่วน
 K_p คือ อัตราขยายสัดส่วน ซึ่งเป็นตัวแปรปรับค่าได้
 e คือ ค่าความผิดพลาด
 t คือ เวลา

ผลอัตราขยายสัดส่วนที่สูงจะทำให้ค่าความผิดพลาดเปลี่ยนแปลงมาก ซึ่งหากมีค่าสูงเกินไปอาจส่งผลให้ระบบไม่เสถียร แต่ในทางตรงกันข้ามผลอัตราขยายสัดส่วนที่ต่ำ ระบบควบคุมจะมีผลตอบสนองต่อกระบวนการน้อยตามไปด้วย

2) หน่วยปริพันธ์ (Integral)

หน่วยปริพันธ์ (Integral) เป็นสัดส่วนของขนาดความผิดพลาดและระยะเวลาของความผิดพลาด โดยผลรวมของความผิดพลาดในทุกช่วงเวลาหรือปริพันธ์ของความผิดพลาด จะทำให้เกิดความผิดพลาดสะสมที่ควรจะเป็นในก่อนหน้า โดยความผิดพลาดสะสมจะถูกคูณโดยอัตราขยายปริพันธ์ (K_i) ซึ่งอัตราขยายปริพันธ์จะเป็นตัวกำหนดขนาดของผลของหน่วยปริพันธ์ ดังแสดงในสมการที่ 2.3

$$I_{out} = K_i \int_0^t e(\tau) d\tau \quad (2.3)$$

- เมื่อ I_{out} คือ สัญญาณขาออกของหน่วยปริพันธ์
 K_i คือ อัตราขยายปริพันธ์ ซึ่งเป็นตัวแปรปรับค่าได้
 e คือ ค่าความผิดพลาด
 t คือ เวลา
 τ คือ ตัวปริพันธ์หุน

หน่วยปริพันธ์ เมื่อรวมกับหน่วยสัดส่วน) จะเร่งกระบวนการให้เข้าสู่จุดที่ต้องการและขจัดความผิดพลาดที่เหลืออยู่ที่เกิดจากการใช้เพียงหน่วยสัดส่วน แต่อย่างไรก็ตาม หน่วยปริพันธ์เป็นการตอบสนองต่อความผิดพลาดสะสมในอดีต จึงสามารถทำให้เกิดการพุ่งเกิน (Overshoot) ได้

3) หน่วยอนุพันธ์ (Derivative)

หน่วยอนุพันธ์ (Derivative) คือ อัตราการเปลี่ยนแปลงของความผิดพลาดจากกระบวนการ ซึ่งคำนวณหาจากความชันของความผิดพลาดในทุกช่วงเวลา และคูณด้วยอัตราขยายอนุพันธ์ (K_d) ซึ่งขนาดของผลของหน่วยอนุพันธ์จะขึ้นกับอัตราขยายอนุพันธ์ (K_d) ดังสมการที่ 2.4

$$D_{out} = K_d \frac{d}{dt} e(t) \quad (2.4)$$

เมื่อ D_{out} คือ สัญญาณขาออกของหน่วยอนุพันธ์

K_d คือ อัตราขยายอนุพันธ์ ซึ่งเป็นตัวแปรปรับค่าได้

e คือ ค่าความผิดพลาด

t คือ เวลา

หน่วยอนุพันธ์จะชะลออัตราการเปลี่ยนแปลงของสัญญาณขาออกของระบบควบคุมแล ซึ่งจะช่วยให้ระบบควบคุมเข้าสู่จุดที่ต้องการ ดังนั้นหน่วยอนุพันธ์จะใช้ในการลดขนาดของค่าพุ่งเกินที่เกิดจากหน่วยปริพันธ์และทำให้เสถียรภาพของการรวมกันของระบบควบคุมดีขึ้น แต่อย่างไรก็ตามอนุพันธ์ของสัญญาณรบกวนที่ถูกขยายในระบบควบคุมจะมีความไวต่อการรบกวนและสามารถทำให้กระบวนการเกิดความไม่เสถียรได้หากสัญญาณรบกวนและอัตราขยายอนุพันธ์มีขนาดใหญ่เพียงพอ

ผลรวมของหน่วยสัดส่วน (Proportional) หน่วยปริพันธ์ (Integral) และหน่วยอนุพันธ์ (Derivative) จะนำมารวมกันเป็นสัญญาณขาออกของการควบคุมแบบพีไอดี โดยกำหนดให้ $u(t)$ เป็นสัญญาณขาออก ซึ่งสมการสุดท้ายของระบบควบคุมเป็นไปดังสมการที่ 2.5 ดังนี้

$$u(t) = MV(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad (2.5)$$

การปรับแต่งค่าพารามิเตอร์ของระบบควบคุมพีไอดี ผลกระทบจากการปรับแต่งค่าพารามิเตอร์ของตัวควบคุมพีไอดีที่มีต่อผลตอบสนองของระบบในโดเมนเวลา เป็นไปดังตารางที่ 2.2

ตารางที่ 2.2 ผลกระทบจากการปรับแต่งค่าพารามิเตอร์ของตัวควบคุมพีไอดี

ตัวแปร	เวลาขึ้น (Rise time)	การพุ่งเกิน (Overshoot)	เวลาสู่สมดุล (Settling time)	ค่าความผิดพลาดในสถานะคงตัว (Steady-state error)
K_p	ลดลง	เพิ่มขึ้น	เปลี่ยนแปลงเล็กน้อย	ลดลง
K_i	ลดลง	เพิ่มขึ้น	เพิ่มขึ้น	ลดลงอย่างมีนัยสำคัญ
K_d	เปลี่ยนแปลงเล็กน้อย	ลดลง	ลดลง	ไม่มีผลกระทบ

วิธีการที่ใช้อย่างแพร่หลายในการปรับแต่ง คือ วิธี Ziegler–Nichols method โดยขั้นแรกให้กำหนดค่า K_p และ K_d เป็นศูนย์ และเพิ่มอัตราขยายสัดส่วนจนมีค่าสูงที่สุด (K_u) เมื่อเริ่มเกิดการแกว่ง จึงนำค่าอัตราขยายสัดส่วนสูงสุด (K_u) ที่ได้ และคาบการแกว่ง (T_u) มาคำนวณหาค่าตัวแปรที่เหลือดังตารางที่ 2.3

ตารางที่ 2.3 ความสัมพันธ์ของค่าพารามิเตอร์ของตัวควบคุมพีไอดี

รูปแบบการควบคุม	K_p	K_i	K_d
P	$0.5K_u$	-	-
PI	$0.45K_u$	$0.54K_u/T_u$	-
PID	$0.6K_u$	$1.2K_u/T_u$	$3K_uT_u/40$

2.4 โครงข่ายประสาทเทียม

2.4.1 Machine Learning (ส่วนการเรียนรู้ของเครื่อง)

Machine Learning (ส่วนการเรียนรู้ของเครื่อง) หมายถึง ศาสตร์แขนงหนึ่งที่ทำให้คอมพิวเตอร์มีความสามารถในการเรียนรู้ด้วยตนเอง เมื่อมีข้อมูลเข้าสามารถทำนายหรือตัดสินใจได้โดยปราศจากการทำงานตามลำดับคำสั่งโปรแกรม หรือกล่าวอีกนัยหนึ่งคือสามารถคิดได้ด้วยตนเองนั่นเอง เป็นการรวมของศาสตร์หลายแขนงไม่ว่าจะเป็น วิทยาการคอมพิวเตอร์ วิศวกรรม และโดยเฉพาะอย่างยิ่งเกี่ยวกับสถิติ นอกจากนี้ยังเชื่อมโยงกับองค์ความรู้ปลายทางที่ต้องการนำไปประยุกต์ใช้ เช่น ชีววิทยา เคมี หลักการตลาด เป็นต้น

Machine Learning จะเรียนรู้จากสิ่งที่ส่งเข้าไปกระตุ้น จากนั้นจะจดจำเอาไว้เป็นมันสมอง ส่งผลลัพธ์ออกมาเป็นตัวเลข หรือ รหัส (Code) ที่ส่งต่อไปเพื่อแสดงผล หรือให้ปัญญาประดิษฐ์ (Artificial Intelligence : AI) นำไปแสดงการกระทำการเรียนรู้เอง สามารถเอาไปใช้งานได้หลายรูปแบบ ต้องอาศัยกลไกที่เป็นโปรแกรม หรือเรียกว่า Algorithm ที่มีความหลากหลาย

รูปแบบของ Machine Learning สามารถจำแนกตามหน้าที่และขอบเขตปัญหา ได้ดังนี้

1) การเรียนรู้แบบมีผู้สอน (Supervised Learning) เป็นการเรียนรู้ของเครื่องซึ่งสร้างฟังก์ชันจากข้อมูลสอน (Training Data) ข้อมูลสอนประกอบด้วยข้อมูลเข้า (Input) และผลที่ต้องการ (Output) ผลจากการเรียนรู้จะเป็นฟังก์ชันที่อาจจะให้ค่าต่อเนื่องที่เรียกวิธีการว่า การถดถอย (Regression) หรือใช้ทำนายประเภทของข้อมูล ซึ่งเรียกว่า การแบ่งประเภท (Classification) ภารกิจของเครื่องเรียนรู้แบบมีผู้สอนคือการทำนายค่าของฟังก์ชันจากข้อมูลเข้าที่ถูกต้องโดยใช้ตัวอย่างสอนจำนวนน้อย โดยเครื่องเรียนรู้จะต้องวางนัยทั่วไปจากข้อมูลที่มีอยู่ไปยังกรณีที่ไม่เคยพบอย่างมีเหตุผล ตัวอย่างเช่น การเรียนรู้เพื่อรู้จำลายมือ

2) การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) เป็นการเรียนรู้ของเครื่อง โดยการสร้างโมเดลที่เหมาะสมกับข้อมูล การเรียนรู้แบบนี้แตกต่างจากการเรียนรู้แบบมีผู้สอน คือ จะไม่มีการระบุผลลัพธ์ที่ต้องการหรือประเภทไว้ก่อน การเรียนรู้แบบนี้จะพิจารณาข้อมูลเป็นเซตของตัวแปรสุ่ม แล้วจึงสร้างโมเดลความหนาแน่นร่วมของชุดข้อมูล สามารถนำไปใช้ร่วมกับการอนุมานแบบเบย์ เพื่อหาความน่าจะเป็นแบบมีเงื่อนไขของตัวแปรสุ่มโดยกำหนดตัวแปรที่เกี่ยวข้องให้ นอกจากนี้ยังสามารถนำไปใช้ในการบีบอัดข้อมูล ซึ่งโดยพื้นฐานแล้ว ขั้นตอนวิธีการบีบอัดข้อมูลจะขึ้นอยู่กับ การแจกแจงความน่าจะเป็นของข้อมูลไม่อย่างชัดเจนก็โดยปริยาย

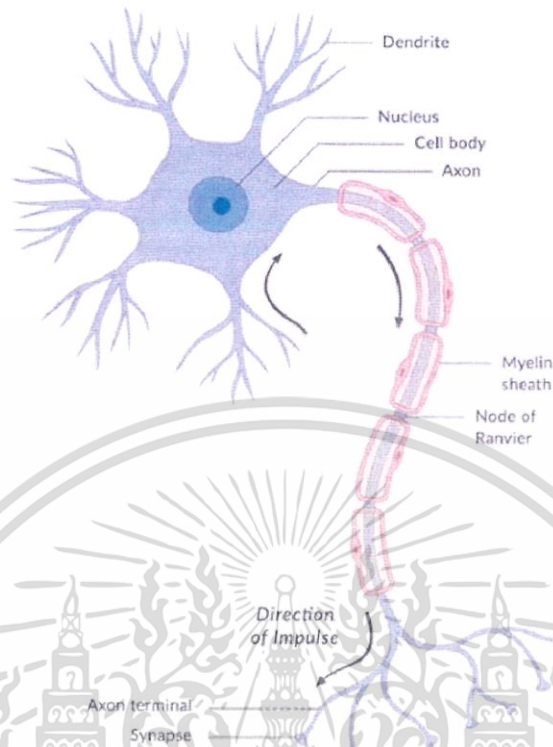
3) การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning) เป็นการเรียนรู้ของเครื่องที่มีแนวความคิดคือ จะเรียนรู้จากสิ่งแวดล้อมรอบตัว เรียนรู้จากธรรมชาติรอบตัวที่มีอยู่ในชีวิตจริง นำมาดัดแปลงให้กับคอมพิวเตอร์ ตัวอย่างเช่น การเล่นเกมหมากรุก จะต้องมีการทำนายล่วงหน้าว่าจะสามารถเกิดอะไรขึ้นได้ ซึ่งการเดินแต่ละครั้งอาจจะไม่เป็นผลดีต่อครั้งนั้นแต่อาจมีผลดีในครั้งต่อ ๆ ไปก็ได้ หรือ เกมส OX

2.4.2 โครงข่ายประสาทเทียม

แบบจำลองโครงข่ายประสาทเทียม ประกอบด้วย ระบบการคำนวณแบบไม่เป็นเชิงเส้นซึ่งมีวิธีการดำเนินระบบเป็นแบบขนานและมีรูปแบบการเรียนรู้แบบโครงข่ายใยประสาทชีวภาพ [21] โดยประกอบไปด้วย นิวรอล (โหนดหรือหน่วยประมวลผล) ซึ่งรวมกันอยู่เป็นชั้น ๆ สามารถรับข้อมูลเข้าได้หลายค่าและคำนวณผล โดยจะให้ผลลัพธ์ค่าเดียวหรือหลายค่าก็ได้ ซึ่งการคำนวณในระบบประกอบไปด้วยฟังก์ชันการรวมและฟังก์ชันการคูณ โดยมีความสามารถในการเรียนรู้จากตัวอย่างหลาย ๆ ตัวอย่าง ซึ่งโครงข่ายประสาทเทียมนี้ จะหาแนวทางการแก้ปัญหา โดยเปรียบเทียบผลลัพธ์ที่คลาดเคลื่อนและปรับเปลี่ยนวิธีการประมวลผลเพื่อให้ผลลัพธ์ที่ถูกต้องที่สุด [22]

2.4.2.1 แนวคิดพื้นฐานของโครงข่ายประสาทเทียม

โครงข่ายประสาทเทียม คล้ายกับระบบประสาทของมนุษย์ โดยรับสัญญาณข้อมูลที่ส่งเข้ามาและจะถูกกระตุ้น ในแต่ละเซลล์ จะประกอบไปด้วยปลายในการรับกระแสประสาท เรียกว่า "เดนไดรต์" (Dendrite) ซึ่งเป็นข้อมูลป้อนเข้า และปลายในการส่งกระแสประสาท เรียกว่า "แอกซอน" (Axon) ดังรูปที่ 2.6 ซึ่งทำหน้าที่เสมือนข้อมูลที่ส่งออกมาของเซลล์ เซลล์เหล่านี้ทำงานด้วยปฏิกิริยาไฟฟ้าเคมี เมื่อมีการกระตุ้นด้วยสิ่งเร้าภายนอก หรือกระตุ้นด้วยเซลล์ด้วยกัน กระแสประสาทจะวิ่งผ่านเดนไดรต์เข้าสู่นิวเคลียส ซึ่งจะเป็นตัวตัดสินใจว่าต้องกระตุ้นเซลล์อื่น ๆ ต่อหรือไม่ ถ้ากระแสประสาทแรงพอ นิวเคลียสจะกระตุ้นเซลล์อื่น ๆ ต่อไปผ่านทางแอกซอน โดยผลการกระตุ้นด้วยสิ่งเร้าที่เหมือนกัน จะให้ผลลัพธ์สุดท้ายเป็นค่าที่ค่อนข้างแน่นอน



รูปที่ 2.6 โครงสร้างระบบประสาทในสมอง [21]

โครงข่ายใยประสาทเทียมเป็นปัญญาประดิษฐ์ (Artificial Intelligence) ที่สร้างเลียนแบบระบบประสาทของมนุษย์ [21] โดยระบบการทำงานต่าง ๆ ของโครงข่ายใยประสาทประดิษฐ์มีลักษณะคล้ายกับระบบประสาทของมนุษย์ คือ มีการเรียนรู้จากประสบการณ์ที่ได้รับ และสามารถให้คำตอบได้แม้แต่ข้อมูลที่ป้อนเข้าจะผิดพลาด หรือไม่สมบูรณ์ โดยหาวิธีการแก้ปัญหาจากประสบการณ์การเรียนรู้ที่ผ่านมา และสามารถที่จะพัฒนาให้ไปเป็นโปรแกรมที่สามารถโต้ตอบกับมนุษย์ได้

2.4.2.2 การสร้างแบบจำลองโครงข่ายใยประสาทประดิษฐ์

โครงข่ายใยประสาทประดิษฐ์ คือ เครื่องจักรการเรียนรู้ ตั้งอยู่บนพื้นฐานความคิดเกี่ยวกับการปรับปรุงตัวแปรควบคุมต่าง ๆ ภายในด้วยตัวมันเอง โดยมีองค์ประกอบภายในระบบซึ่งประกอบไปด้วย 5 องค์ประกอบหลัก คือ หน่วยการเรียนรู้ โครงข่ายใยประสาท แผนการเรียนรู้ กระบวนการเรียนรู้ และกระบวนการวิเคราะห์ สามารถจำแนกส่วนประกอบของโครงข่ายใยประสาทประดิษฐ์ เป็น 3 ชั้น คือ 1) การออกแบบ 2) การสร้างแบบจำลอง และ 3) การทดสอบและหาผลลัพธ์ [23]

2.4.2.3 ข้อมูลป้อนเข้า

โครงข่ายประสาทเทียม ประกอบไปด้วยตัวแปรอิสระ หรือ ข้อมูลป้อนเข้า และตัวแปรตาม หรือ ผลลัพธ์ โดยหลักการเลือกตัวแปรที่ใช้ในโครงข่ายที่เกี่ยวข้องจะมี 2 แบบ [24] คือ วิธีแรก ข้อมูลจะต้องถูกแปลงรูปให้อยู่ในรูปที่เหมาะสม และวิธีที่สอง คือการเลือกข้อมูลโดยใช้พื้นฐานระหว่าง Predictiveness และ Covariance โดยปกติแล้ว ตัวแปรอิสระที่ถูกเลือกจะมีความสามารถในการทำนายผลหากตัวแปรที่เลือกมีความสัมพันธ์กันในทางตรงกันข้าม หากตัวแปรอิสระ 2 ตัวมีความสัมพันธ์ต่อกันจะทำให้แบบจำลองมีความอ่อนไหว (Sensitive) และเกิดปัญหาที่เรียกว่า Over Fitting และ Limit Generalization ด้วยเหตุผลนี้ การเลือกข้อมูลจะต้องเลือกเฉพาะตัวแปรอิสระที่มีความสามารถทำนายผลเพื่อให้ได้ผลลัพธ์หรือตัวแปรตาม โดยตัวแปรอิสระที่เลือกมานั้นจะต้องไม่มีความสัมพันธ์กัน แต่อย่างไรก็ดี ก็ขึ้นอยู่กับรูปแบบของโครงข่ายที่ใช้และเพื่อที่จะลดจำนวนตัวอย่างที่ใช้ในการสอนและเวลาที่ใช้ในการเรียนรู้ ก็ควรจะต้องการคัดเลือกข้อมูลป้อนเข้าให้เหมาะสม เพราะการคัดเลือกข้อมูลเป็นปัจจัยที่สำคัญในการสร้างแบบจำลอง

2.4.2.4 ชั้นซ่อน (Hidden Layer)

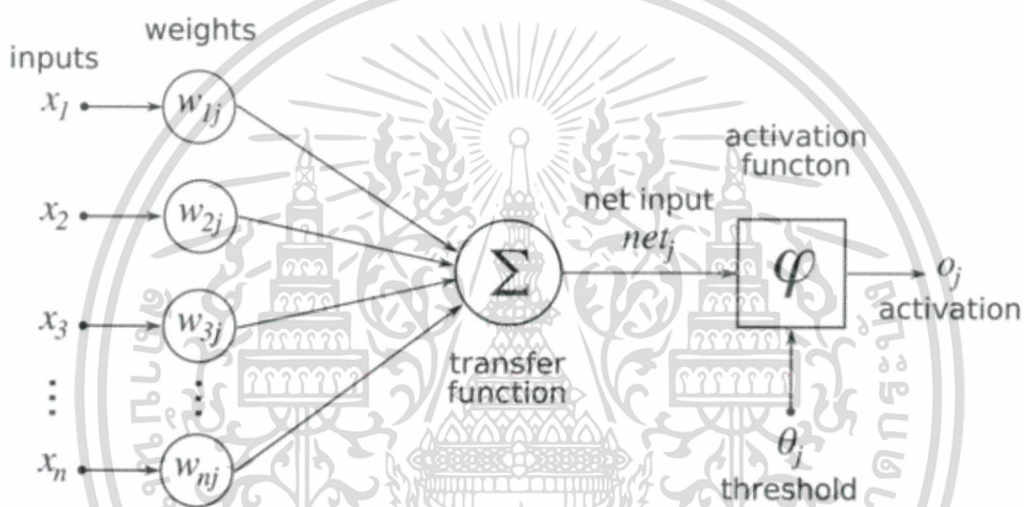
ชั้นซ่อน (Hidden Layer) เป็นชั้นประมวลผลที่อยู่ระหว่างชั้นข้อมูลป้อนเข้าและชั้นแสดงผลลัพธ์ โดยปกติแล้วชั้นซ่อนอาจมีมากกว่า 1 ชั้น โดยโครงข่ายจะสามารถประมวลหาฟังก์ชันที่เหมาะสมจากปัญหาที่ซับซ้อนได้หากมีชั้นซ่อนที่มากพอ [21] ข้อมูลที่ได้จากชั้นซ่อนจะได้เป็นตัวแปรใหม่ที่จะถูกส่งต่อให้กับชั้นแสดงผลลัพธ์ หรือชั้นตัวแปรตาม ถ้าโครงข่ายแบบแพร่กลับ (Backpropagation) มีชั้นซ่อนที่น้อยเกินไปแล้วจะทำให้โครงข่ายไม่สามารถที่จะหาทางแก้ปัญหาได้ แต่ถึงอย่างไรก็ตามถ้าเกิดโครงข่ายมีชั้นซ่อนที่มากจนเกินไป จะทำให้โครงข่ายมีระยะเวลาในการเรียนรู้นาน และไม่ช่วยให้โครงข่ายมีประสิทธิภาพมากขึ้น นอกจากนี้การที่มีโหนดในแต่ละชั้นที่มากเกินไป จะทำให้โครงข่ายไม่สามารถที่จะหาจุดสิ้นสุดได้ ดังนั้น การที่จะทำให้โครงข่ายเกิดประสิทธิภาพสูงสุด ต้องกำหนดให้มีโหนดในชั้นซ่อนอยู่ให้น้อยเท่าที่จะเป็นไปได้ [24]

2.4.2.5 ค่าถ่วงน้ำหนักและไบแอส (Weights and Biases)

ค่าถ่วงน้ำหนักถูกแทนด้วยตัวเลขเพื่อแสดงถึงความแรงในการเชื่อมต่อของโหนดแต่ละโหนดที่ถูกเชื่อมต่อเข้าด้วยกัน ซึ่งผลรวมของค่าถ่วงน้ำหนักที่ป้อนเข้าจะไปปรับปรุงการประมวลผลในแต่ละโหนด ค่าถ่วงน้ำหนักคือค่าความแรงสัมพัทธ์ในทางคณิตศาสตร์ของการเชื่อมต่อ ซึ่งส่งผลต่อการส่งผ่านข้อมูลจากชั้นหนึ่งไปยังชั้นต่อไป โดยปกติค่าถ่วงน้ำหนักจะถูกกำหนดและเริ่มป้อนเข้าสู่โครงข่ายในขั้นตอนการเรียนรู้ ซึ่งต้องมีหลักการในการกำหนดค่าเพื่อที่จะให้โครงข่ายสามารถแก้โจทย์ปัญหาและลดเวลาการเรียนรู้ได้ สำหรับโครงข่ายใด ๆ ค่าถ่วงน้ำหนักจะมีค่าเท่ากับผลคูณของจำนวนโหนดของทุก ๆ การเชื่อมต่อ และค่าของไบแอสจะเท่ากับผลรวมของจำนวนโหนดของทุก ๆ การเชื่อมต่อ [23]

2.4.2.6 ฟังก์ชันการรวมและฟังก์ชันการแปลงค่า (Summation and Transfer Function)

ฟังก์ชันการรวม คือ ฟังก์ชันการหาผลเฉลี่ยค่าถ่วงน้ำหนักของทุก ๆ โหนดที่เชื่อมต่อกัน โดยมีขั้นตอนคือ นำค่าของข้อมูลป้อนเข้า (Input) ในแต่ละโหนดคูณกับค่าถ่วงน้ำหนักของแต่ละโหนดและรวมผลลัพธ์ของทุก ๆ โหนดเข้าด้วยกัน ดังรูปที่ 2.7 ส่วนฟังก์ชันการแปลงค่า คือ ความสัมพันธ์ระหว่างระดับการกระตุ้นภายในโหนด (N) กับผลลัพธ์ที่ได้ (Output) โดยจะอยู่ในรูปของฟังก์ชันซิกมอยด์ $f(N)$ โดยมีข้อแม้ว่าจะต้องมีความต่อเนื่องและค่าของฟังก์ชันซิกมอยด์จะต้องเพิ่มขึ้นเมื่อ N เพิ่มขึ้น



รูปที่ 2.7 กระบวนการทำงานของโครงข่ายประสาทประดิษฐ์ในโหนดย่อย [23]

2.4.2.7 กระบวนการสอนหรือการเรียนรู้ (Training or Learning)

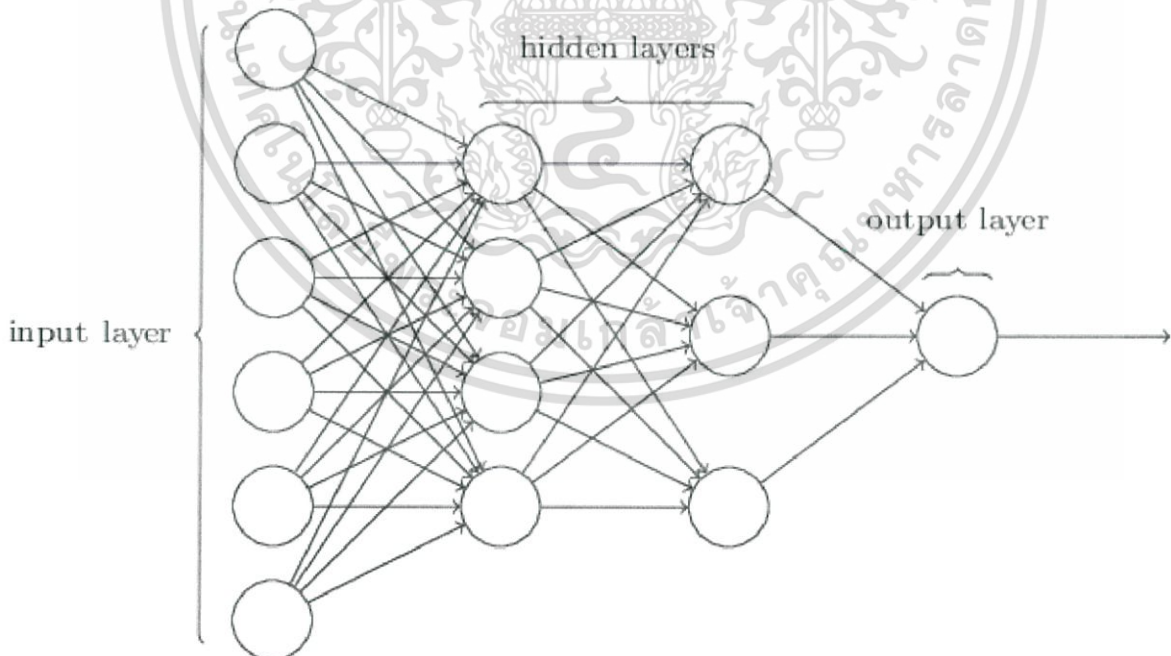
กระบวนการเรียนรู้เป็นกระบวนการหนึ่งในโครงข่ายที่เรียนรู้จากความผิดพลาดโดยมีหลักการ 3 ข้อ คือ 1) คำนวณหาคำตอบ 2) ตรวจสอบคำตอบว่าถูกต้องหรือไม่ 3) ปรับแก้ค่าถ่วงน้ำหนักแล้วคำนวณใหม่อีกครั้ง กระบวนการสอน คือ การนำข้อผิดพลาดจากการคำนวณครั้งก่อนมาปรับแก้ค่าถ่วงน้ำหนักให้กับการสอนในรอบต่อไป ทำให้คำตอบที่ได้มีความถูกต้องสูงขึ้น

ในขั้นตอนการเรียนรู้จะเป็นกระบวนการที่เกี่ยวข้องกับการปรับค่าน้ำหนักของแต่ละโหนด โดยอาศัยจากประสบการณ์การเรียนรู้ของโครงข่าย การเรียนรู้ของโครงข่ายในแต่ละรอบจะนำผลลัพธ์ที่ประมวลได้มาเปรียบเทียบกับค่าจริงที่ได้จากการทดลอง ซึ่งอาจจะมีคลาดเคลื่อน หลังจากนั้นโครงข่ายจะย้อนกลับไปเรียนรู้ในรอบต่อไปพร้อมกับการปรับแก้ค่าถ่วงน้ำหนัก เพื่อให้การประมวลผลรอบต่อไปมีความแม่นยำมากขึ้น

โดยการปรับแก้ อาจจะเป็นการปรับขึ้นหรือปรับลงก็ได้ การวัดผลว่าโครงข่ายสามารถเรียนรู้ได้ดีเพียงใดจะดูจากตัวชี้วัดต่าง ๆ เช่น ค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (Mean Square Error) ในชั้นแสดงผลลัพธ์ [24]

2.4.2.8 การเรียนรู้แบบแพร่กลับ (Back-Propagation)

ในช่วงปี 1950 Rosenblatt ได้สร้างโครงข่ายชั้นเดียวแบบง่าย ๆ ขึ้นมาโดยมีชื่อเรียกว่า Perceptron หลังจากนั้นต่อมา Widrow และ Hoff ได้สร้างอัลกอริทึมขึ้นมาใหม่ที่สามารถอธิบายได้ด้วยกฎของเดลตา (Delta rule) โดยใช้การเรียนรู้ข้อมูลที่มีการปรับค่าถ่วงน้ำหนักซึ่งมีชื่อเรียกว่า Widrow-Hoff rule ต่อมาในปี 1986 Rumelhart, Hilton และ Williams ได้ศึกษาค้นคว้าเพิ่มเติมจนเกิดกระบวนการเรียนรู้แบบแพร่กลับ (Back-Propagation) หรือเรียกว่า Generalized Delta Rule (GDR) การเรียนรู้แบบแพร่กลับนี้ เป็นที่ยอมรับกันอย่างแพร่หลายโดยเป็นการเรียนรู้แบบมีครูสอน และมีระบบการเชื่อมโยงแบบเคลื่อนไปข้างหน้าหลายชั้น (Multilayer Feed Forward) ดังรูปที่ 2.8 สำหรับวิธีการเรียนรู้แบบแพร่กลับ จะมีการปรับค่าถ่วงน้ำหนัก (w) ในทุก ๆ รอบการเรียนรู้เพื่อให้เกิดค่าความผิดพลาดที่น้อยที่สุด โดยจะเริ่มต้นปรับแก้ค่าถ่วงน้ำหนักตั้งแต่ชั้น Output หลังจากนั้นทำการปรับย้อนกลับมาที่ชั้นซ่อนชั้นสุดท้าย และชั้นซ่อนชั้นต่อ ๆ มา จนกระทั่งถึงชั้นซ่อนชั้นแรก



รูปที่ 2.8 โครงข่ายใยประสาทเทียมแบบแพร่กลับ [21]

2.4.2.9 ชุดข้อมูล (Samples)

ชุดข้อมูล คือ ข้อมูลที่ทราบตัวแปรต้นและตัวแปรตามเพื่อนำมาใช้สอนโครงข่าย โดยแหล่งที่มาของชุดข้อมูลแบ่งออกได้เป็น 3 ลักษณะ คือ 1) แบบสอบถาม 2) ข้อมูลทางสถิติ และ 3) จากการทดลอง โดยชุดข้อมูลดังกล่าวจะนำมาแบ่งออกเป็น 2 ชุด คือ ชุดการสอน (Training Set) และ ชุดทดสอบ (Test Set)

2.4.2.10 การทดสอบโครงข่าย (Testing)

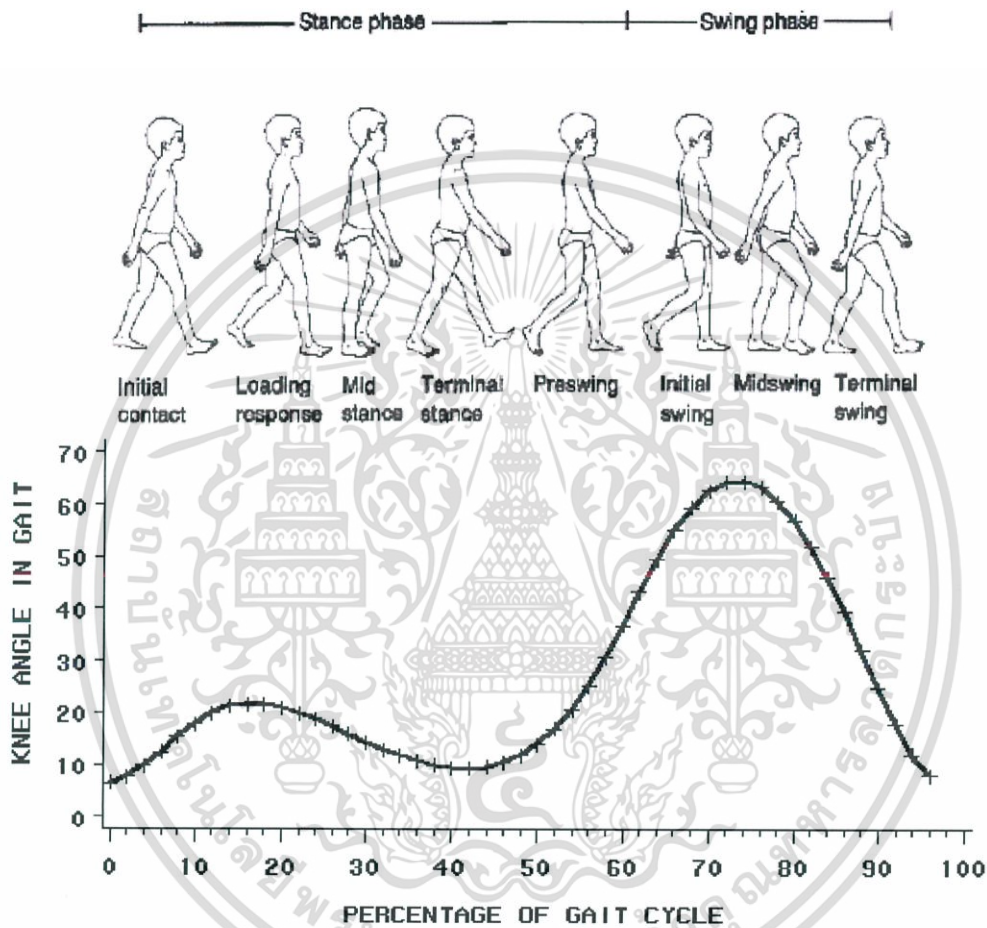
การทดสอบโครงข่ายเป็นการทดสอบว่าโครงข่ายสามารถที่จะเรียนรู้จากชุดการสอน (Training Set) ได้ดีเพียงไร โดยใช้ชุดข้อมูลที่ไม่เคยใช้สำหรับการสอนมาทดสอบ เรียกว่า ชุดทดสอบ (Test Set) ซึ่งโครงข่ายที่สามารถให้ผลลัพธ์ที่แม่นยำได้เมื่อใช้ชุดทดสอบมาทดสอบ จะเป็นโครงข่ายที่น่าเชื่อถือในการทดสอบ

โครงข่ายสามารถแบ่งออกได้เป็น 2 ลักษณะ คือ 1) แบ่งข้อมูลออกเป็น 2 ชุด โดยชุดแรกไว้สำหรับสอนโครงข่ายให้จดจำรูปแบบของข้อมูล และข้อมูลชุดที่สองไว้สำหรับทดสอบโครงข่าย โดยผลลัพธ์ที่แตกต่างระหว่างค่าจริง และค่าที่ได้จากชุดทดสอบจะถูกคำนวณออกมาเป็นค่าผิดพลาดของระบบ (System Error) ซึ่งค่าผิดพลาดของระบบที่น้อย จะแสดงถึงความสามารถในการทำนายที่สูง 2) ใช้ชุดข้อมูลทั้งหมดเป็นทั้งชุดการสอน และชุดทดสอบ โดยนำชุดข้อมูลทั้งหมดมาสอนโครงข่ายก่อน และหลังจากนั้นนำชุดข้อมูลชุดเดิมมาทดสอบโครงข่าย [24]

2.5 งานวิจัยที่เกี่ยวข้อง

ในการจัดทำวิทยานิพนธ์ฉบับนี้ ทางคณะผู้จัดทำได้ใช้ข้อมูลโดยอ้างอิงจากงานวิจัยต่าง ๆ ที่เกี่ยวข้องกับการจัดทำวิทยานิพนธ์ เพื่อให้สามารถเข้าใจงานที่จะต้องจัดทำ และทำให้การจัดทำวิทยานิพนธ์ครั้งนี้มีความถูกต้องมากที่สุด โดยจากการศึกษางานวิจัยที่เกี่ยวข้องทั้งหมดทำให้ทราบได้ว่าการเดินอาศัยหลักการทำงานระหว่างระบบประสาทร่วมกับระบบโครงร่างกระดูกและกล้ามเนื้อ โดยมนุษย์แต่ละคนมักมีกระบวนการในการเคลื่อนไหวร่างกายไปในทิศทางต่าง ๆ และมีการรักษาสมดุลของร่างกายในขณะที่คล้ายคลึงกัน แต่อย่างไรก็ตาม ด้วยรูปแบบโครงสร้างร่างกายที่แตกต่างกัน ทำให้ลักษณะการเดินของแต่ละบุคคลมักมีลักษณะที่เฉพาะตัว [25] โดยการเคลื่อนไหวที่เกิดขึ้นนี้ ถูกควบคุมด้วยการหดตัวและคลายตัวของกล้ามเนื้อ ซึ่งจะทำให้เกิดคลื่นไฟฟ้ากล้ามเนื้อ (Electromyography) ขึ้นในระหว่างการเดิน [26] ซึ่งภายหลังจากการเกิดความพิการและการสวมใส่ขาเทียมในผู้พิการขาขาด จะส่งผลให้กลไกและรูปแบบการเดิน รวมถึงลักษณะคลื่นไฟฟ้ากล้ามเนื้อเปลี่ยนไป โดยผู้พิการจะมีกระบวนการในการชดเชยความสูญเสียการเคลื่อนไหวที่

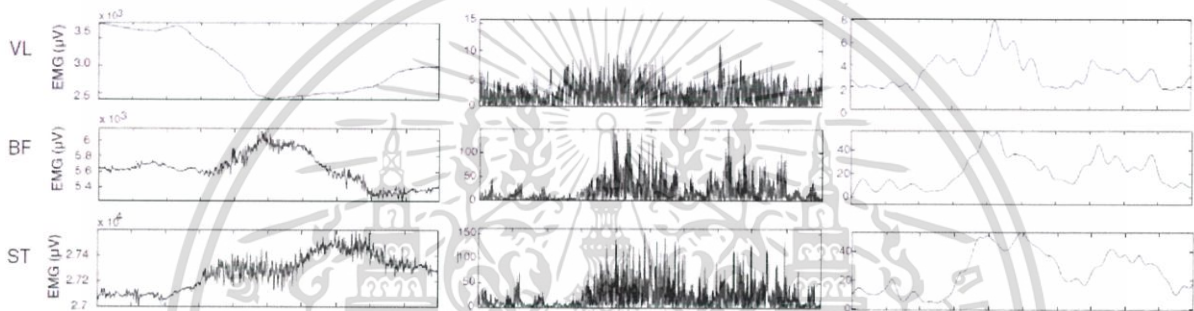
เกิดขึ้น และปรับเปลี่ยนรูปแบบการเดินให้สอดคล้องไปกับการเคลื่อนที่ได้ [27] โดยมุมมองข้อเข่าที่เกิดขึ้นในแต่ละช่วงของวงจรการเดินเป็นดังรูปที่ 2.9



รูปที่ 2.9 มุมงอข้อเข่าที่เกิดขึ้นในแต่ละช่วงของวงจรการเดิน [28]

ลักษณะของคลื่นไฟฟ้ากล้ามเนื้อแต่ละส่วนมีความแตกต่างกันในแต่ละช่วงของวงจรการเดิน โดยในคนปกติกล้ามเนื้อในกลุ่ม Vasti ได้แก่ Vastus Lateralis และ Vastus Medialis เป็นกล้ามเนื้อที่มีการทำงานมากที่สุดในช่วง mid swing (90% ของวงจรการเดิน) ในขณะที่กล้ามเนื้อในกลุ่ม Hamstring ได้แก่ Semitendinosus, Semimembranosus และ Biceps Femoris มีการทำงานมากที่สุดในช่วงระหว่าง Mid Swing ตอนปลายถึงช่วง Terminal Swing (75% ของวงจรการเดิน) ดังแสดงในรูปที่ 2.10 ซึ่งคลื่นไฟฟ้ากล้ามเนื้อที่เกิดขึ้นนี้สามารถใช้ในการระบุความแตกต่างระหว่างการเดินในบุคคลปกติและบุคคลที่มีความ

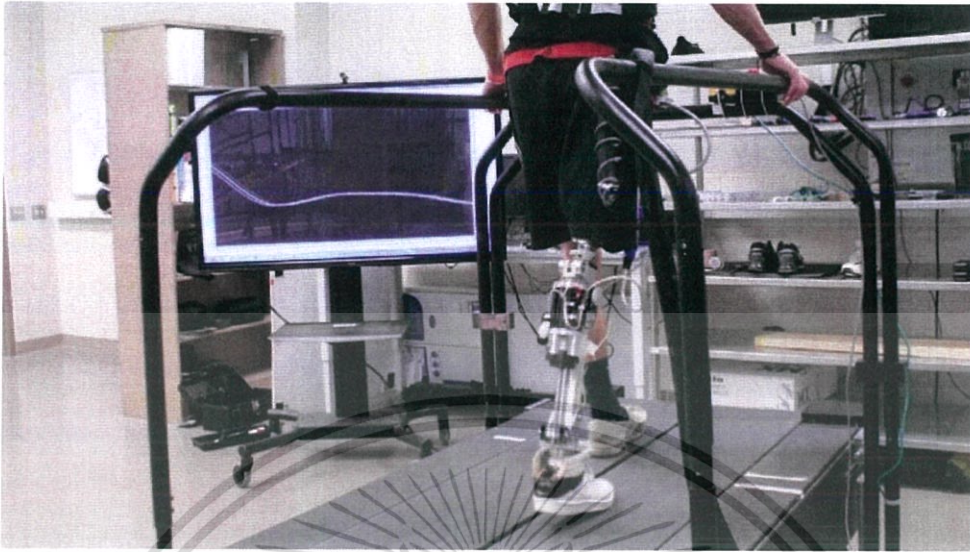
ทพพลาฟได้ [26] โดยกล้ามเนื้อในผู้พิการจะมีการชดเชยความสูญเสีย โดยการเพิ่มการทำงานของกล้ามเนื้อเหนือเข้าเพื่อเพิ่มความสมดุลในการทรงตัว ซึ่งในแต่ละคนจะมีการชดเชยในกล้ามเนื้อแต่ละส่วนที่ต่างกันออกไป ทำให้การทำงานของกล้ามเนื้อ Biceps Femoris ระหว่างวงจรการเดินมีความแตกต่างกันออกไปในผู้พิการแต่ละคน รวมถึงแตกต่างจากบุคคลทั่วไปเช่นกัน [29] นอกจากนี้ในผู้พิการที่ใช้ขาเทียมประเภทเหนือเข้า จะมีการทำงานของกล้ามเนื้อ Vastus Medialis ในช่วง Initial Contact น้อยกว่าในบุคคลปกติ ซึ่งส่งผลให้ไม่สามารถเหยียดขาออกไปได้เท่ากับการเดินแบบปกติ ในขณะที่กล้ามเนื้อ Biceps Femoris ในผู้พิการจะถูกกระตุ้นให้มีการทำงานสูงกว่าบุคคลทั่วไปในระหว่างครึ่งแรกของช่วง Stance Phase [27]



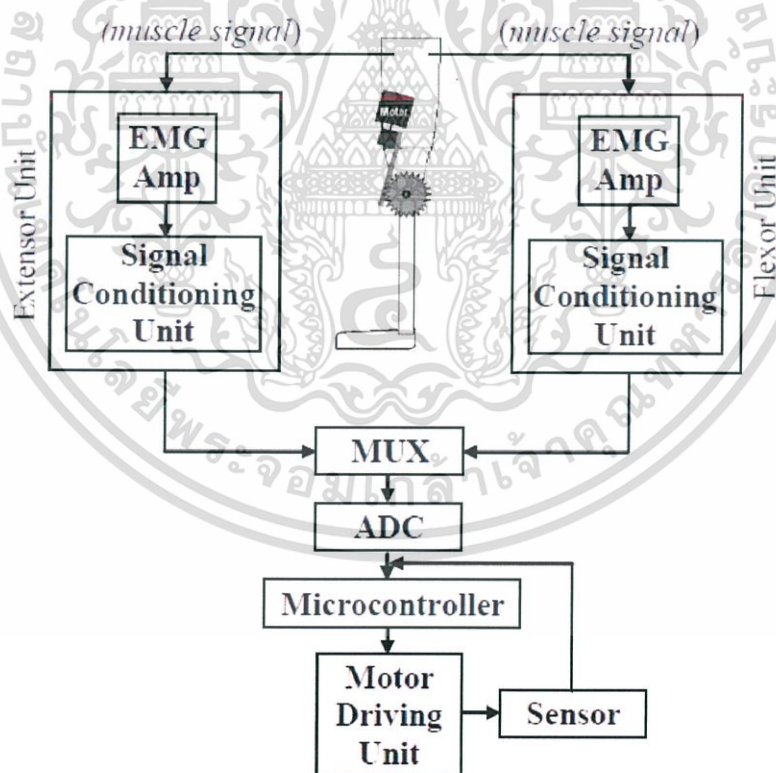
รูปที่ 2.10 การทำงานของกล้ามเนื้อ Vastus Lateralis, Biceps Femoris และ Semitendinosus [30]

เมื่อคลื่นไฟฟ้ากล้ามเนื้อที่เกิดขึ้นในกล้ามเนื้อต่าง ๆ ระหว่างวงจรการเดินมีความแตกต่างกัน ทำให้สามารถนำมาใช้ในการออกแบบขาเทียมสำหรับผู้พิการได้ เนื่องจากคลื่นไฟฟ้ากล้ามเนื้อมีความสัมพันธ์โดยตรงกับปริมาณการเคลื่อนไหวของร่างกาย ดังนั้นจึงนำมาใช้ในการควบคุมความหน่วงของข้อเข้าเทียม โดยระบบที่ควบคุมขาเทียมจะทำการระบุว่าคลื่นไฟฟ้าที่เกิดขึ้นทำให้เกิดการเคลื่อนไหวของข้อเข้าในช่วงใดของวงจรการเดิน [31] [32]

ขาเทียมที่ควบคุมการเคลื่อนไหวโดยใช้คลื่นไฟฟ้ากล้ามเนื้อ ถูกออกแบบมาเพื่อช่วยให้ผู้พิการมีความสะดวกสบายในการเดินมากขึ้น [33] ดังแสดงในรูปที่ 2.11 โดยมีหลักการทำงาน คือ การควบคุมการเคลื่อนไหวของขาเทียมซึ่งใช้สัญญาณมาจากกล้ามเนื้อหรือคลื่นไฟฟ้ากล้ามเนื้อมาใช้ควบคุม โดยตัววัดสัญญาณกล้ามเนื้อที่ใช้คือ อิเล็กโทรด ซึ่งมีขนาดเล็กพกพาสะดวก และติดตั้งใช้งานง่าย โดยนำสัญญาณที่ได้จากอิเล็กโทรดนั้นไปเข้าวงจรทางไฟฟ้าเพื่อขยาย และกรองสัญญาณ เพื่อส่งไปยังระบบควบคุมเป็นตัวประมวลผลและสั่งงาน ทำให้ขาเทียมสามารถเคลื่อนไหวได้ตามความต้องการของผู้ใช้งาน วงจรการทำงาน of ขาเทียมควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ ดังแสดงในรูปที่ 2.12



รูปที่ 2.11 การทดลองเดินโดยใช้ขาเทียมแบบควบคุมโดยใช้คลื่นไฟฟ้ากล้ามเนื้อ [33]



รูปที่ 2.12 แผนภาพการทำงานของขาเทียมควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ [31]

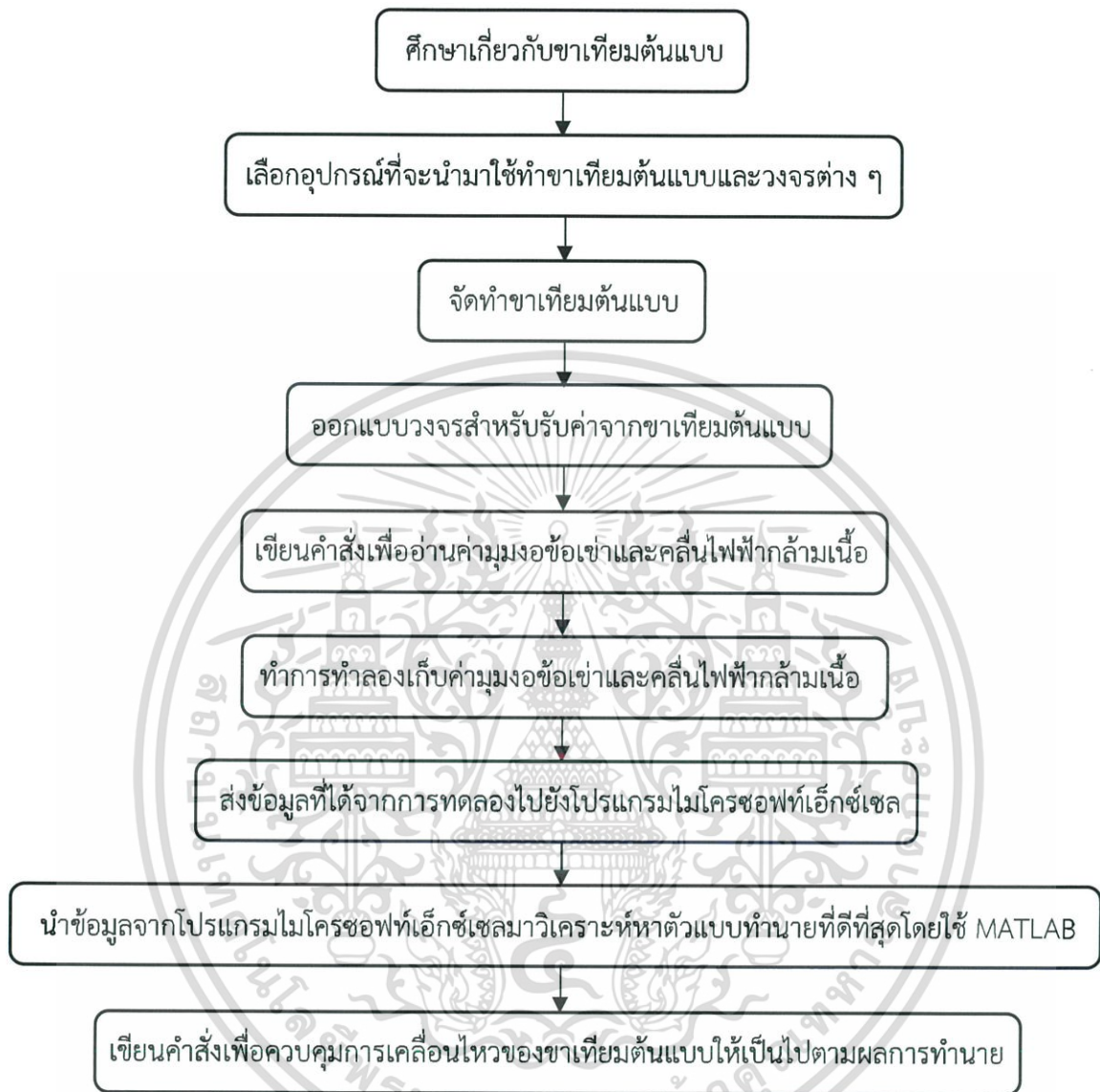
บทที่ 3

วิธีดำเนินงานวิจัย

ในการดำเนินงานจัดทำปฏิญานิพนธ์เรื่องการออกแบบและพัฒนาต้นแบบขาเทียมควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ จากการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง ทำให้ทีมผู้วิจัยได้รับทราบถึงข้อมูลต่าง ๆ ที่มีประโยชน์ และสามารถนำมาประยุกต์ใช้ได้กับการจัดทำปฏิญานิพนธ์ครั้งนี้ โดยภายหลังจากการศึกษานั้น ทางทีมผู้วิจัยสามารถแบ่งขั้นตอนการดำเนินงานวิจัยออกเป็น 2 ส่วน ดังนี้

1. การสร้างขาเทียมต้นแบบและออกแบบวงจรสำหรับเก็บข้อมูล
 - 1.1 การสร้างขาเทียมต้นแบบ
 - 1.2 วิธีคำนวณหาค่า R และระยะยืดและหดของมอเตอร์ ซึ่งสัมพันธ์กับมุมงอข้อเข่า
 - 1.3 การออกแบบวงจรสำหรับรับค่ามุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อ
 - 1.4 การทดลองเก็บข้อมูลค่ามุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อที่เกิดขึ้นในขณะเดิน
2. การเขียนคำสั่งเพื่อให้เกิดการเคลื่อนไหวของขาเทียมต้นแบบ
 - 2.1 การสร้างแบบจำลองโครงข่ายประสาทเทียมเพื่อทำนายลักษณะการเดิน
 - 2.2 การเขียนคำสั่งเพื่อควบคุมการเคลื่อนไหวของขาเทียมต้นแบบ

ในการจัดทำปฏิญานิพนธ์ครั้งนี้ ทางทีมผู้วิจัยได้มีการดำเนินงานในหลายขั้นตอนเริ่มตั้งแต่ขั้นตอนของการศึกษาในหัวข้อต่าง ๆ ที่เกี่ยวข้องกับการปฏิบัติงาน รวมไปถึงการลงมือปฏิบัติงานจริง ทั้งในเรื่องของการออกแบบขาเทียมต้นแบบ การเขียนคำสั่งเพื่อใช้ในการรับค่ามุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อ รวมถึงการออกแบบตัวแบบสำหรับทำนายลักษณะการเดิน และการเขียนคำสั่งเพื่อใช้สำหรับควบคุมการเคลื่อนไหวของขาเทียมต้นแบบให้เป็นไปตามผลการทำนาย ซึ่งสามารถสรุปขั้นตอนการทำงาน ได้ดังรูปที่ 3.1



รูปที่ 3.1 แผนผังขั้นตอนการดำเนินงาน

3.1 การสร้างขาเทียมต้นแบบและออกแบบวงจรสำหรับเก็บข้อมูล

3.1.1 การสร้างขาเทียมต้นแบบ

ในการสร้างขาเทียมต้นแบบต้องอาศัยส่วนประกอบต่าง ๆ เพื่อให้ต้นแบบสามารถทำงานได้ตามความต้องการ โดยในการออกแบบขาเทียมต้นแบบนั้นมีส่วนประกอบ ดังนี้

1) Linear Actuator Motor

Linear Actuator Motor เป็นอุปกรณ์เคลื่อนที่เชิงเส้นขับเคลื่อนด้วยไฟฟ้า ดังรูปที่ 3.2 ซึ่งมีทำงานแบบเคลื่อนที่ในแนวเส้นตรง โดยมีคุณสมบัติดังนี้

- Stroke: 50 มิลลิเมตร (mm)
- ความเร็ว: 24 มิลลิเมตรต่อวินาที (mm/s)
- แรงดันไฟฟ้า: ไฟฟ้ากระแสตรง (DC) 12 โวลต์
- โหลดสูงสุด: 350 นิวตัน (N)
- Duty Cycle: 25 เปอร์เซ็นต์ (%)

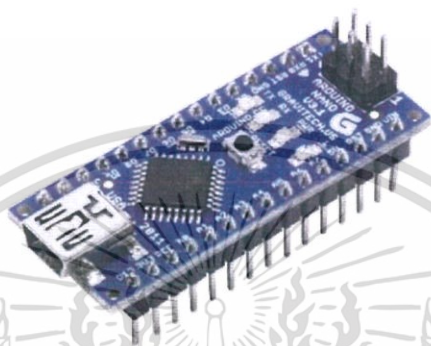


รูปที่ 3.2 ลักษณะของ Linear Actuator Motor

2) ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ (Microcontroller) เป็นอุปกรณ์ควบคุมขนาดเล็ก ทำหน้าที่ประมวลผลตามโปรแกรมหรือชุดคำสั่ง ซึ่งบรรจุความสามารถที่คล้ายคลึงกับระบบคอมพิวเตอร์ โดยในไมโครคอนโทรลเลอร์ได้รวมเอาซีพียู หน่วยความจำ และพอร์ต ซึ่งเป็นส่วนประกอบหลักสำคัญของระบบคอมพิวเตอร์เข้าไว้ด้วยกัน ผ่านการออกแบบวงจรให้เหมาะกับงานต่าง ๆ และยังสามารถโปรแกรมคำสั่งเพื่อควบคุมขาารับเข้า (Input)

และส่งออก (Output) เพื่อสั่งงานให้ไปควบคุมอุปกรณ์ต่าง ๆ ซึ่งบอร์ดที่นำมาใช้คือ บอร์ด Arduino Nano ใช้ชิปไอซีไมโครคอนโทรลเลอร์เบอร์ MEGA328P มีชิป USB to UART และใช้ Mini USB เชื่อมต่อกับคอมพิวเตอร์ ดังรูปที่ 3.3



รูปที่ 3.3 ลักษณะของไมโครคอนโทรลเลอร์ (Microcontroller) Arduino Nano

3) ชุดขับมอเตอร์ L298N Dual H-Bridge Motor Controller

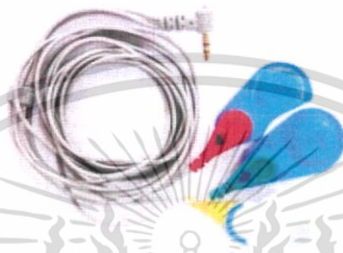
L298N เป็นชุดขับมอเตอร์ชนิด H-Bridge ซึ่งส่วนใหญ่จะถูกนำไปใช้ในการควบคุมทิศทาง และความเร็วของมอเตอร์ ซึ่งสามารถควบคุมมอเตอร์ได้ทั้งหมด 2 ช่อง (Channel) หลักการทำงานคือ วงจร H-Bridge ของ L298N จะขับกระแสเข้ามอเตอร์ตามขั้วที่กำหนดด้วยลอจิกเพื่อควบคุมทิศทาง ส่วนความเร็วของมอเตอร์นั้นจะถูกควบคุมด้วย สัญญาณพัลส์ (Pulse Width Modulation: PWM) ซึ่งมีขนาด: 43 x 43 x 26 มิลลิเมตร มีแหล่งจ่ายไฟในตัวสามารถจ่ายไฟออกจากช่อง 5 โวลต์ เพื่อจ่ายให้บอร์ด Arduino ได้เมื่อต่อไฟเลี้ยงเข้าที่ช่อง 12 โวลต์ ดังรูปที่ 3.4



รูปที่ 3.4 ลักษณะของชุดขับมอเตอร์ L298N Dual H-Bridge

4) อิเล็กโทรดแบบสัมผัสผิวหนัง (Surface Electrode)

อิเล็กโทรดแบบสัมผัสผิวหนัง (Surface Electrode) เป็นแผ่นขั้วไฟฟ้าที่วางบนผิวหนังบริเวณที่วัด ใช้สำหรับตรวจวัดและบันทึกคลื่นไฟฟ้ากล้ามเนื้อ เพื่อแสดงถึงความหนักเบาของการหดตัวของกล้ามเนื้อ



รูปที่ 3.5 ลักษณะของอิเล็กโทรดแบบสัมผัสผิวหนัง (Surface Electrode)

หลังจากนำส่วนประกอบต่าง ๆ ที่กล่าวมาข้างต้นมาประกอบกับขาเทียม จะได้ขาเทียมต้นแบบตามลักษณะที่ต้องการ โดยขาเทียมนี้สามารถเคลื่อนที่ในแนวแกนเดียวและมีข้อเข้าเป็นจุดหมุน ทำให้สามารถเหยียดและงอข้อเข้าได้ ซึ่งจะมีลักษณะดังรูปที่ 3.6



รูปที่ 3.6 ลักษณะของขาเทียมต้นแบบ

3.1.2 วิธีคำนวณค่า R และระยะยึดและหดของมอเตอร์ ซึ่งสัมพันธ์กับมุมงอข้อเข่า (Knee angle)

การคำนวณหาความสัมพันธ์ระหว่างมุมงอข้อเข่า ระยะยึดหดของมอเตอร์ และค่า R ที่วัดได้จากเซนเซอร์ สามารถเขียนแผนภาพแสดงจุดหมุนและแนวแกนต่างๆได้ดังรูปที่ 3.7 โดยสมการที่ใช้ในการคำนวณหาค่ามุมต่างๆที่มีผลต่อมุมงอข้อเข่าของขาเทียม (θ_1') เป็นดังนี้

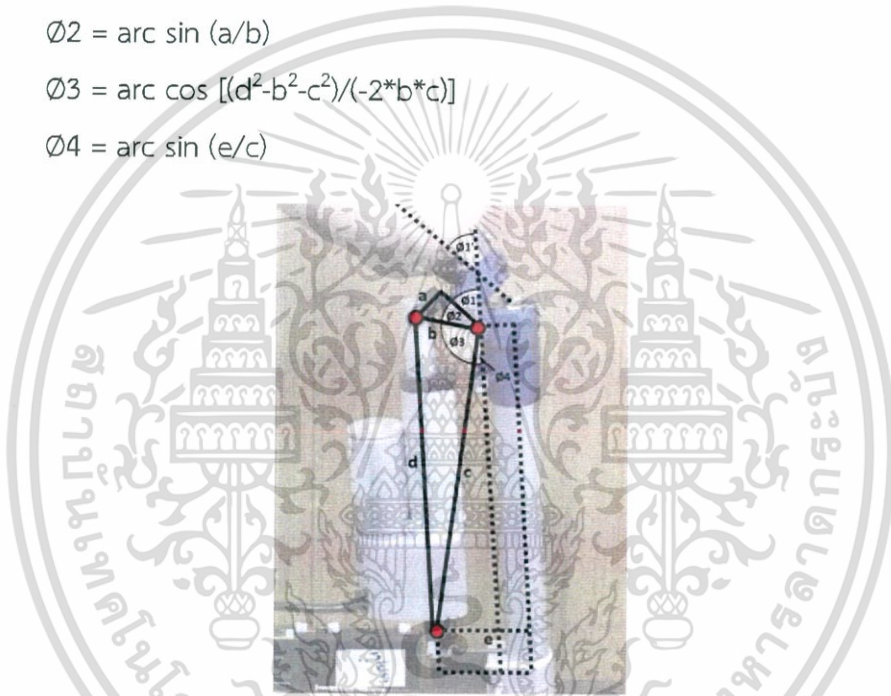
$$\theta_1' = \theta_1 \quad (3.1)$$

$$\theta_1 = 180^\circ - (\theta_2 + \theta_3 + \theta_4) \quad (3.2)$$

$$\theta_2 = \arcsin (a/b) \quad (3.3)$$

$$\theta_3 = \arcsin [(d^2 - b^2 - c^2)/(-2*b*c)] \quad (3.4)$$

$$\theta_4 = \arcsin (e/c) \quad (3.5)$$



รูปที่ 3.7 จุดหมุนและมุมต่างๆที่สัมพันธ์กับมุมงอข้อเข่า (Knee angle)

ความสัมพันธ์ระหว่างระยะยึดหดของมอเตอร์ (d) มุมงอข้อเข่า (θ_1') และค่า R ที่วัดได้จากเซนเซอร์ภายในมอเตอร์ เป็นการแปรผันเชิงเส้น โดยสามารถคำนวณค่า R ได้โดยใช้สมการเส้นตรง ดังสมการที่ 3.6

$$y=43.2653061x-419.714286 \quad (3.6)$$

โดยที่ y คือ ค่า R และ x คือ ระยะยึดหดของมอเตอร์

จากสมการแสดงความสัมพันธ์ข้างต้น สามารถนำมาคำนวณหาความสัมพันธ์ระหว่างระยะยึดหดของมอเตอร์ (d) มุมงอข้อเข่า (θ_1') และค่า R ได้ดังตารางที่ 3.1

ตารางที่ 3.1 ความสัมพันธ์ระหว่างระยะยึดหดของมอเตอร์ (d) มุมงอข้อเข่า (θ) และค่า R ที่วัดได้จากเซนเซอร์

ระยะยึดหดของ มอเตอร์ (d)	มุมงอข้อเข่า (θ)	ค่า R ที่วัดได้ จากเซนเซอร์	ระยะยึดหดของ มอเตอร์ (d)	มุมงอข้อเข่า (θ)	ค่า R ที่วัดได้ จากเซนเซอร์
18.9	91.51	398	21.4	52.90	506
19.0	89.75	402	21.5	51.42	510
19.1	88.03	407	21.6	49.93	515
19.2	86.34	411	21.7	48.44	519
19.3	84.67	415	21.8	46.95	523
19.4	83.04	420	21.9	45.44	528
19.5	81.43	424	22.0	43.93	532
19.6	79.84	428	22.1	42.40	536
19.7	78.27	433	22.2	40.87	541
19.8	76.71	437	22.3	39.32	545
19.9	75.17	441	22.4	37.76	549
20.0	73.64	446	22.5	36.19	554
20.1	72.13	450	22.6	34.59	558
20.2	70.62	454	22.7	32.98	562
20.3	69.12	459	22.8	31.35	567
20.4	67.63	463	22.9	29.70	571
20.5	66.15	467	23.0	28.02	575
20.6	64.67	472	23.1	26.32	580
20.7	63.20	476	23.2	24.59	584
20.8	61.72	480	23.3	22.83	588
20.9	60.25	485	23.4	21.04	593
21.0	58.78	489	23.5	19.20	597
21.1	57.31	493	23.6	17.33	601
21.2	55.84	498	23.7	15.41	606
21.3	54.37	502	23.8	13.44	610

3.1.3 การออกแบบวงจรสำหรับวัดคลื่นไฟฟ้ากล้ามเนื้อและค่ามูมงข้อเข่า

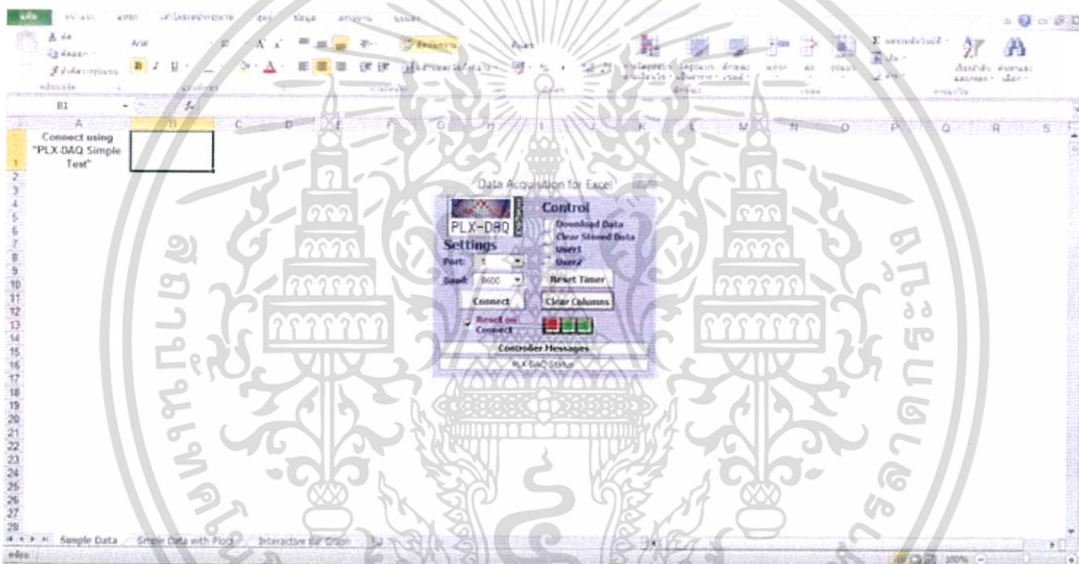
ทีมผู้วิจัยได้สร้างวงจรสำหรับวัดคลื่นไฟฟ้ากล้ามเนื้อและค่ามูมงข้อเข่าในระหว่างการเดินขึ้นมาเอง ภายในวงจรประกอบไปด้วย อิเล็กทรอนิกส์สำหรับวัดคลื่นไฟฟ้ากล้ามเนื้อ ใช้ทั้งหมด 4 ช่องสัญญาณในการวัดคลื่นไฟฟ้ากล้ามเนื้อจากกล้ามเนื้อทั้ง 4 ชุด วงจรภายในอิเล็กทรอนิกส์ 1 ช่องสัญญาณแสดงดังรูปที่ 3.8 Encoder สำหรับวัดค่ามูมงข้อเข่า ซึ่งเชื่อมต่อเข้ากับบอร์ด Arduino ที่ทำหน้าที่ประมวลผลข้อมูล โดยมีการเชื่อมต่อกับคอมพิวเตอร์เพื่อรับส่งข้อมูลค่าคลื่นไฟฟ้ากล้ามเนื้อและค่ามูมงข้อเข่า ซึ่งวงจรดังกล่าวแสดงได้ดังรูปที่ 3.9



รูปที่ 3.8 วงจรของอิเล็กทรอนิกส์ 1 ช่องสัญญาณ

รูปที่ 3.9 วงจรสำหรับรับค่ามูมงข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อ

ทีมผู้วิจัยมีการศึกษาและเขียนคำสั่งเพื่อให้อุปกรณ์สามารถสื่อสารกันเองภายในองค์ประกอบย่อย รวมถึงทำให้ข้อมูลที่ออกมาจากอุปกรณ์สามารถเก็บได้ในคอมพิวเตอร์ภายใต้ไมโครซอฟท์เอ็กเซล (Microsoft Excel) ซึ่งจะมีการรับค่าและนำค่าที่ได้จากอุปกรณ์ไปใช้ผ่านโปรแกรมไมโครซอฟท์เอ็กเซลโดยใช้โปรแกรม Parallax Microcontroller Data Acquisition (PLX-DAQ) ซึ่งข้อมูลที่ได้จะแสดงในโปรแกรมไมโครซอฟท์เอ็กเซล (Microsoft Excel) ภายใต้นามสกุล .csv ซึ่งค่าที่ได้จะเป็นค่าคลื่นไฟฟ้ากล้ามเนื้อเกี่ยวกับค่า มุมของข้อเข่า โดยจะมีลักษณะโปรแกรมดังรูปที่ 3.10 ซึ่งจะนำไปใช้ในการทำนายลักษณะการเดินโดยใช้ Machine Learning ต่อไป



รูปที่ 3.10 โปรแกรม PLX-DAQ สำหรับรับข้อมูลและนำมาจัดเก็บในโปรแกรมไมโครซอฟท์เอ็กเซล

3.1.4 การทดลองเก็บข้อมูลคลื่นไฟฟ้ากล้ามเนื้อและค่ามุมข้อเข่าที่เกิดขึ้นในขณะเดิน

การทดลองเก็บข้อมูลคลื่นไฟฟ้ากล้ามเนื้อและค่ามุมข้อเข่า สามารถทำได้โดยนำวงจรที่สร้างขึ้นเอง ดังที่กล่าวไปในหัวข้อที่ 3.1.3 มาใช้ในการเก็บข้อมูลคลื่นไฟฟ้ากล้ามเนื้อและค่ามุมข้อเข่าที่เกิดขึ้นจริงในขณะเดิน โดยการทดลองนี้จะวัดคลื่นไฟฟ้ากล้ามเนื้อจากกล้ามเนื้อทั้ง 4 มัด ได้แก่ Biceps Femoris, Semitendinosus, Vastus Medialis และ Vastus Lateralis จากขาที่อ่อนบนของคนปกติที่มีสุขภาพดี ค่าคลื่นไฟฟ้ากล้ามเนื้อจะถูกกำหนดให้เป็นข้อมูลนำเข้าหรือตัวแปรอิสระ และมุมข้อเข่าจะถูกกำหนดให้เป็นข้อมูลผลลัพธ์หรือตัวแปรตาม ซึ่งจะใช้โปรแกรม PLX-DAQ สำหรับรับค่าต่าง ๆ ดังกล่าว โดยทำการทดลอง

เดินบนลู่วิ่งด้วยความเร็ว 2 กิโลเมตรต่อชั่วโมง จำนวน 100 ก้าว ทำการทดลองซ้ำ 10 ครั้ง ความถี่ในการเก็บข้อมูลเท่ากับ 60 Hz หรือ 60 ครั้งต่อวินาที และค่าความเร็วในการรับ-ส่งข้อมูลเท่ากับ 38400 บิตต่อวินาที ดังแสดงในรูปที่ 3.11 และนำข้อมูลไปวิเคราะห์เพื่อทำนายลักษณะการเดินของแต่ละบุคคลต่อไป ลักษณะข้อมูลที่เก็บได้แสดงดังรูปที่ 3.12



รูปที่ 3.11 การทดลองเดินบนลู่วิ่งเพื่อเก็บข้อมูลค่ามุมข้อเข่าและคลื่นไฟฟ้า

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1																
2	13:52:38	0.00	6	27	0	0										
3	13:52:38	0.00	11	26	0	0										
4	13:52:38	0.00	11	27	0	0										
5	13:52:38	0.00	6	27	0	0										
6	13:52:38	0.00	5	26	0	0										
7	13:52:38	0.00	5	27	0	0										
8	13:52:38	0.00	4	27	0	0										
9	13:52:38	0.00	3	27	0	0										
10	13:52:38	0.00	3	26	0	0										
11	13:52:38	0.00	3	27	0	0										
12	13:52:38	0.00	2	27	0	0										
13	13:52:38	0.00	3	27	0	0										
14	13:52:38	0.00	2	26	0	0										
15	13:52:38	0.00	3	27	0	0										
16	13:52:38	0.00	1	27	0	0										
17	13:52:38	0.00	4	27	0	0										
18	13:52:38	0.00	6	27	0	0										
19	13:52:38	0.00	5	27	0	0										
20	13:52:39	0.00	2	27	0	0										
21	13:52:39	0.00	6	27	0	0										
22	13:52:39	0.00	9	27	0	0										

รูปที่ 3.12 ตัวอย่างข้อมูลที่เก็บได้แสดงในโปรแกรมไมโครซอฟท์เอ็กเซล

3.2 การเขียนคำสั่งเพื่อให้เกิดการเคลื่อนไหวก่อนขาเทียมต้นแบบ

3.2.1 การสร้างแบบจำลองโครงข่ายประสาทเทียมเพื่อทำนายลักษณะการเดิน

ภายหลังจากการออกแบบต้นแบบขาเทียมและเขียนโปรแกรมเพื่อรับค่าข้อมูลของคลื่นไฟฟ้ากล้ามเนื้อและมุมงอข้อเข่าที่เกิดขึ้นในระหว่างการเดิน ทางทีมผู้วิจัยได้ทำการนำข้อมูลมาวิเคราะห์เพื่อหาแบบจำลองโครงข่ายประสาทเทียมสำหรับทำนายลักษณะการเคลื่อนไหวก่อนขาที่ดีที่สุด โดยใช้ Machine Learning ในโปรแกรม MATLAB

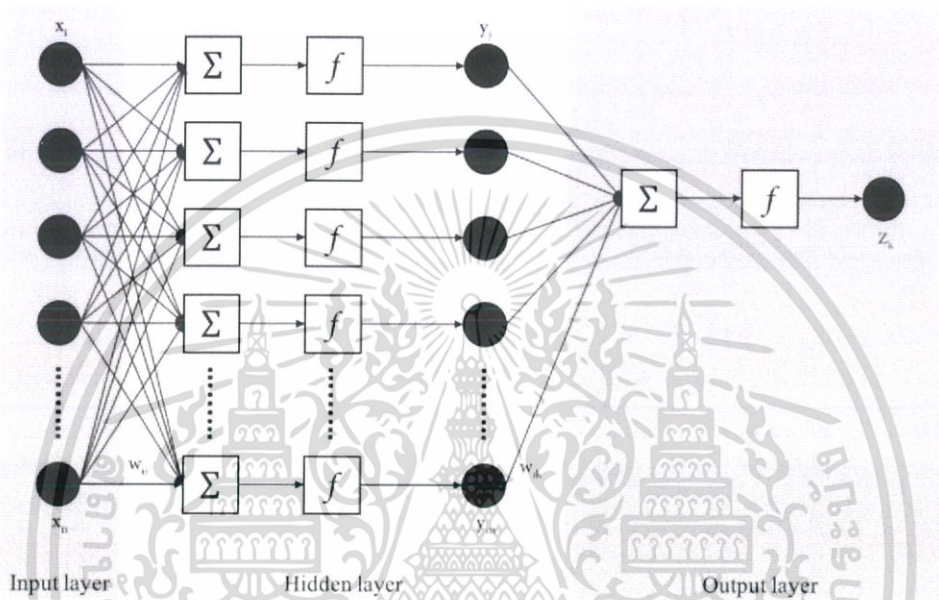
การสร้างแบบจำลองทำนายขนาดมุมงอข้อเข่าที่เกิดขึ้นในระหว่างการเดินโดยใช้โครงข่ายประสาทประดิษฐ์ ประกอบด้วยขั้นตอนใหญ่ ๆ 4 ขั้นตอน คือ การเตรียมข้อมูล การออกแบบโครงข่าย การสอนโครงข่าย และการตรวจสอบความถูกต้องของโครงข่าย โดยเตรียมข้อมูลจากการทดลองเดิน ซึ่งจะกำหนดให้ค่ามุมงอข้อเข่าเป็นตัวแปรตาม และค่าคลื่นไฟฟ้ากล้ามเนื้อจากกล้ามเนื้อทั้ง 4 มัดเป็นตัวแปรอิสระ โดยเรียงข้อมูลตามลำดับเวลา หลังจากนั้นจะแบ่งชุดข้อมูลออกเป็น 2 ชุด คือข้อมูลชุดสอน (Training Set) และข้อมูลชุดทดสอบ (Test Set) และทำการออกแบบโครงข่ายโดยกำหนดให้เป็นแบบไปข้างหน้าหลายชั้น (MLFF) ซึ่งเป็นระบบแบบมีครูสอนโดยจะใช้วิธีการสอนแบบแพร่กระจายความผิดพลาดกลับ (Error Back – Propagation Algorithm) จากนั้นนำข้อมูลชุดสอน (Training Set) ให้โครงข่ายได้เรียนรู้ โครงข่ายจะประมวลผลจนได้คำตอบชุดหนึ่ง สำหรับคำตอบที่โครงข่ายสามารถคำนวณออกมาได้นั้น จะถูกนำมาหาค่าความคลาดเคลื่อนกำลังสอง ถ้ายังมีความผิดพลาดสูงอยู่ ระบบจะย้อนกลับไปปรับเปลี่ยนค่าถ่วงน้ำหนัก และทำการสอนต่อไปจนกว่าค่าความผิดพลาดระหว่างคำตอบที่ได้จากโครงข่าย และคำตอบจริงจะมีค่าน้อยในระดับที่ยอมรับได้ซึ่งในขณะเดียวกันนั้น ชุดทดสอบ (Test Set) จะทำการทดสอบค่าความผิดพลาดของโครงข่ายไปพร้อม ๆ กัน เมื่อค่าความผิดพลาดจากชุดทดสอบมีค่าน้อยในระดับที่ยอมรับได้ จึงจะหยุดทำการปรับสอนและได้โครงข่ายที่เหมาะสมสำหรับใช้งาน ซึ่งรายละเอียดของขั้นตอนการสร้างโครงข่ายประสาทประดิษฐ์ในงานวิจัยครั้งนี้ มีดังต่อไปนี้

1) การเตรียมข้อมูล

ในการศึกษาครั้งนี้ จะนำข้อมูลคลื่นไฟฟ้ากล้ามเนื้อและมุมงอข้อเข่าที่เกิดขึ้นในระหว่างการเดิน ซึ่งเก็บค่าได้จากการทดลองดังที่กล่าวไว้ในหัวข้อ 3.1.4 โดยกำหนดให้ค่ามุมงอข้อเข่าเป็นตัวแปรตาม และคลื่นไฟฟ้ากล้ามเนื้อจากกล้ามเนื้อ 4 มัด เป็นตัวแปรอิสระ โดยจะคัดเลือกชุดข้อมูลในช่วงที่มีความเสถียรเข้าสู่โครงข่ายต่อไป

2) การออกแบบโครงข่าย

ระบบโครงข่ายประสาทเทียมจะประกอบไปด้วยชั้นข้อมูลป้อนเข้า (Input Layer) ชั้นซ่อน Hidden Layer) และชั้นผลลัพธ์ (Output Layer) ดังรูปที่ 3.13



รูปที่ 3.13 โครงข่ายประสาทเทียมซึ่งประกอบไปด้วย ชั้นข้อมูลป้อนเข้า ชั้นซ่อน และชั้นผลลัพธ์ [23]

โดยการออกแบบโครงข่ายจะพิจารณาจากข้อมูลป้อนเข้าโครงข่าย จำนวนชั้นซ่อน ประเภทแบบจำลองที่เลือกใช้ และผลลัพธ์เป็นหลัก ซึ่งจะต้องทำการทดสอบเพื่อหาค่าที่เหมาะสมกับโครงข่ายมากที่สุด โดยมีรายละเอียด ดังนี้

- ข้อมูลป้อนเข้าโครงข่าย

ข้อมูลป้อนเข้าต้องพิจารณาในช่วงข้อมูลที่มีความเสถียร เนื่องจากส่งผลต่อความแม่นยำในกระบวนการสอน โดยข้อมูลป้อนเข้าจะถูกแบ่งออกเป็น 2 กลุ่มได้แก่ ข้อมูลชุดสอน (Training Set) และข้อมูลชุดทดสอบ (Testing Set) ซึ่งต้องกำหนดร้อยละการถ่วงน้ำหนักให้กับข้อมูลแต่ละชุด โดยในข้อมูลชุดสอนจะกำหนดให้ค่าถ่วงน้ำหนักของการสอน (Training) เป็น 70% สำหรับข้อมูลชุดทดสอบจะกำหนดให้ค่าถ่วงน้ำหนักเป็น 30% แบ่งเป็นการตรวจสอบ (Validation) และการทดสอบ (Testing) อย่างละ 15%

- จำนวนชั้นซ่อน (Hidden Layer)

ในการทดสอบหาจำนวนชั้นซ่อนที่มีความเหมาะสมนี้ ผู้จัดทำปริญญาโทจะทำการศึกษาเพื่อเปรียบเทียบระหว่างโครงข่ายประสาทเทียมที่มีชั้นซ่อน 3 ชั้น 4 ชั้น และ 5 ชั้น ตามลำดับ

- ประเภทแบบจำลอง

ทำการเปรียบเทียบแบบจำลองโครงข่ายประสาทเทียมที่เกิดขึ้นจากการเลือกใช้แบบจำลองทางคณิตศาสตร์ทั้ง 3 ประเภท ได้แก่ Levenberg-Marquardt, Bayesian Regularization และ Scaled Conjugate Gradient

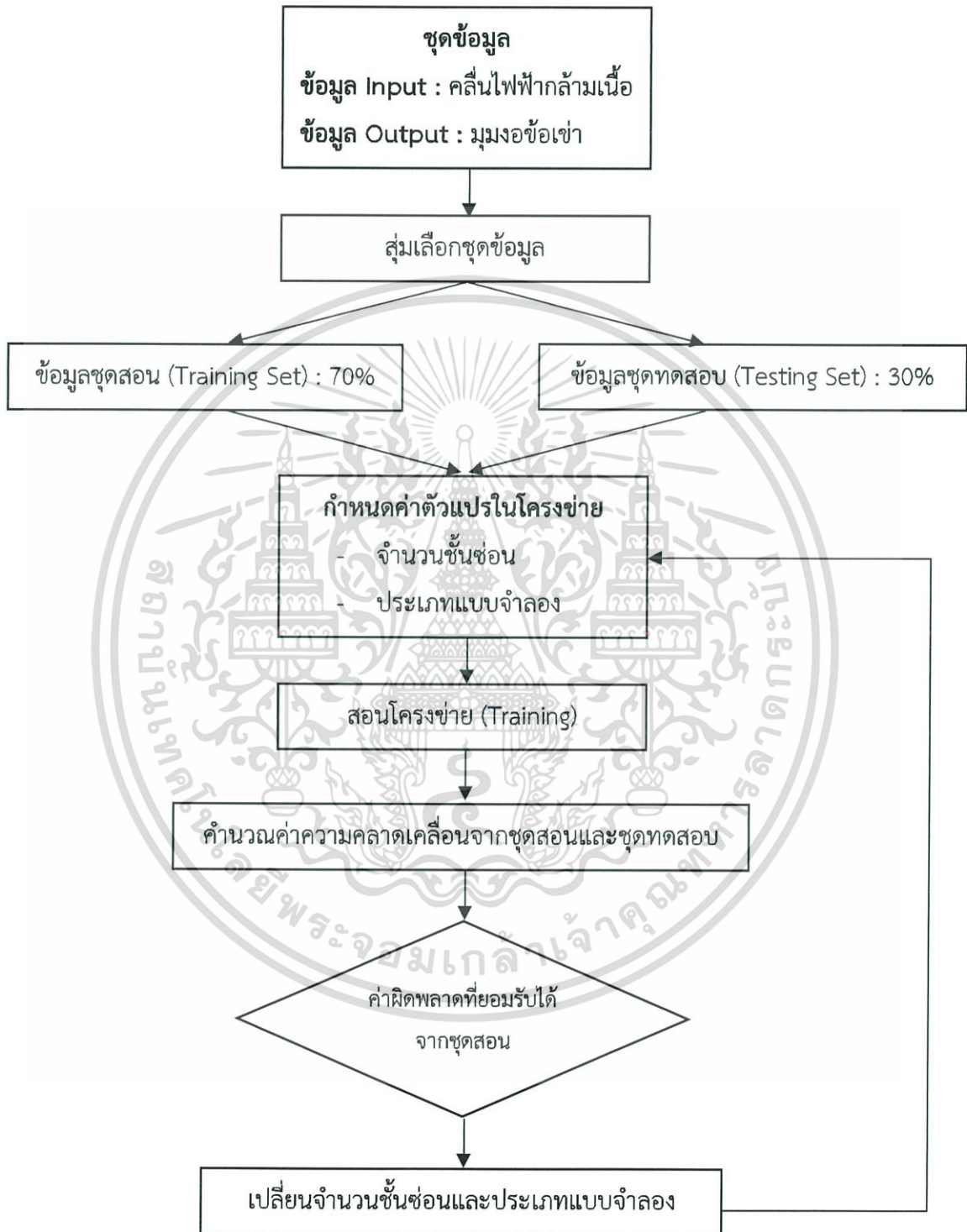
3) การสอนโครงข่าย

ในขั้นตอนการสอนโครงข่ายนี้จะใช้วิธีการแพร่กระจายความผิดพลาดกลับ (Error Back-Propagation Algorithm) โดยแบบจำลองและจำนวนชั้นซ่อนที่เลือกใช้เป็นดังที่กล่าวไปในหัวข้อข้างต้น ซึ่งรายละเอียดกระบวนการสอนเป็นไปดังแผนภาพในรูปที่ 3.14

4) การตรวจสอบความถูกต้องของโครงข่าย

การประเมินผลจากการทำนายโดยใช้โครงข่ายประสาทประดิษฐ์ สามารถใช้สถิติในการทดสอบความถูกต้อง ซึ่งสถิติที่ใช้ในการวัดความผิดพลาดระหว่างผลการทำนายและข้อมูลจริงจากชุดตรวจสอบความถูกต้อง (Validate Set) คือ ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE)

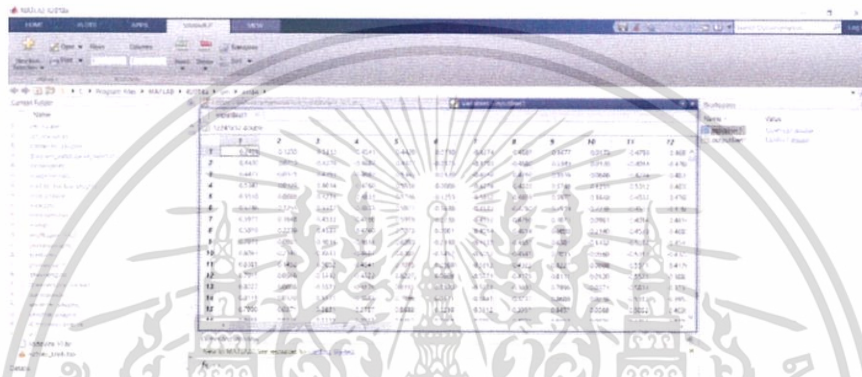
โดยเกณฑ์การพิจารณายอมรับโครงข่าย คือ ต้องเป็นโครงข่ายที่มีค่าความคลาดเคลื่อนกำลังสอง (MSE) เนื่องจากยิ่งความคลาดเคลื่อนน้อย แสดงว่าโครงข่ายมีความแม่นยำมาก



รูปที่ 3.14 กระบวนการสอน (Training) ในโครงข่าย

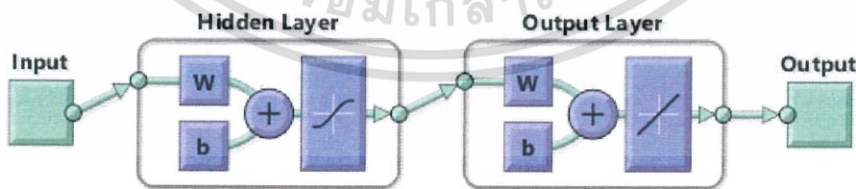
ขั้นตอนในการสร้างแบบจำลองสำหรับทำนายลักษณะการเดินโดยใช้ Machine Learning ในโปรแกรม MATLAB เป็นดังนี้

1) Import ค่าคลื่นไฟฟ้ากล้ามเนื้อและมุมมองข้อเข่าจากโปรแกรมไมโครซอฟต์เอ็กเซลมาใส่ในโปรแกรม MATLAB โดยเลือกประเภทของชุดข้อมูลเป็นแบบ Numeric Matrix โดยหน้าต่างนำเข้าข้อมูลเป็นดังรูปที่ 3.15



รูปที่ 3.15 หน้าต่างสำหรับนำเข้าข้อมูลค่าคลื่นไฟฟ้ากล้ามเนื้อและค่ามุมมองข้อเข่า

2) เปิดใช้ฟังก์ชัน Neural Network Fitting Application เพื่อใช้สร้างตัวแบบทางคณิตศาสตร์สำหรับทำนายค่ามุมมองข้อเข่าที่จะเกิดขึ้นจากคลื่นไฟฟ้ากล้ามเนื้อทั้ง 4 มัด โดยเลือกใช้ Two-Layer Feed-Forward Network ซึ่งมีชั้นซ่อน (Hidden Layer) ดังรูปที่ 3.16



รูปที่ 3.16 Two-Layer Feed-Forward Network ในฟังก์ชัน Neural Network Fitting Application

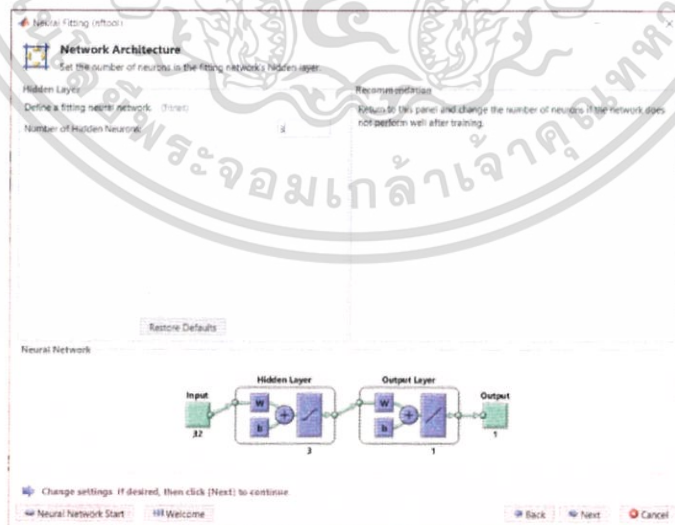
3) ใส่ค่า Inputs และ Targets โดยค่า Inputs คือ คลื่นไฟฟ้ากล้ามเนื้อ และ Targets คือ ค่ามุมมองข้อเข่าที่เกิดขึ้นจริงในระหว่างการเดิน

4) กำหนดค่าร้อยละสำหรับถ่วงน้ำหนักการสอน (Training) การตรวจสอบ (Validation) และการทดสอบ (Testing) โดยเลือกกำหนดให้เป็น 70%, 15% และ 15% ตามลำดับ ดังรูปที่ 3.17



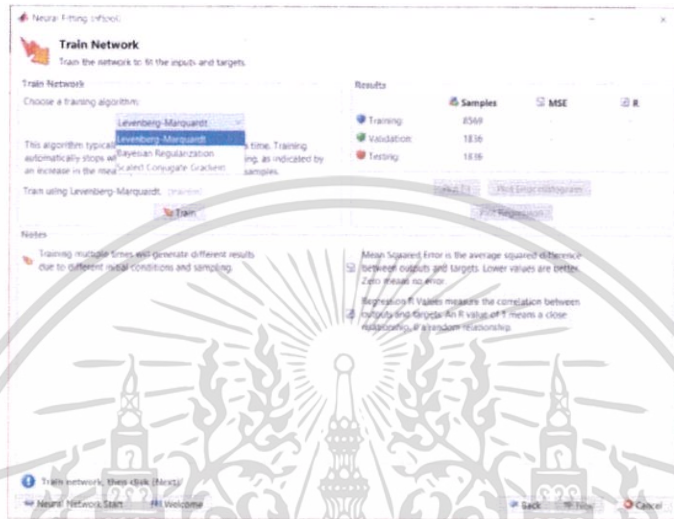
รูปที่ 3.17 หน้าต่างแสดงการกำหนดค่าร้อยละการถ่วงน้ำหนัก

5) กำหนดจำนวน Hidden Layer ที่ต้องการใช้ในการวิเคราะห์ข้อมูล โดยในการศึกษาครั้งนี้เลือกใช้ชั้นซ่อนจำนวน 3, 4 และ 5 ชั้นในการทดสอบ ดังรูปที่ 3.18



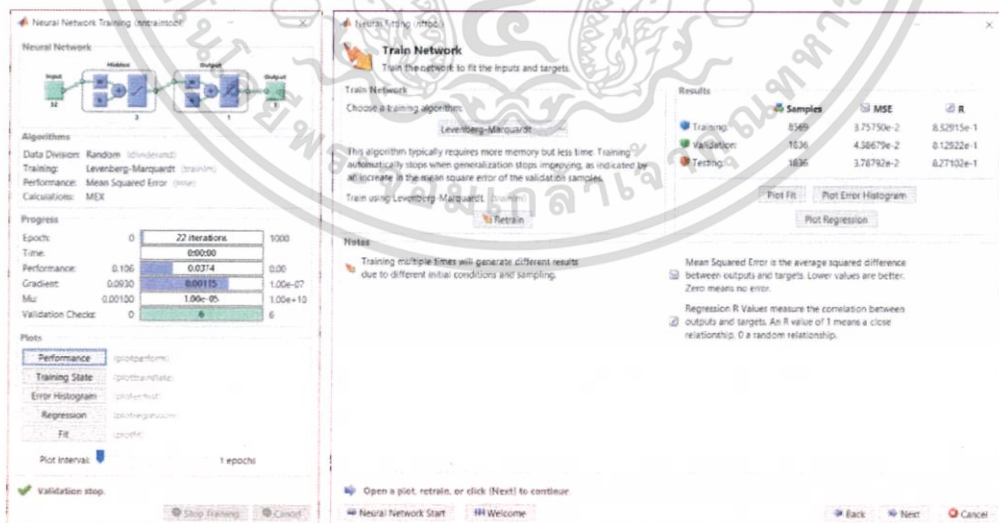
รูปที่ 3.18 หน้าต่างแสดงการกำหนดจำนวน Hidden Layer

6) เลือกประเภทต้นแบบทางคณิตศาสตร์ที่ต้องการใช้ในการทดสอบ ได้แก่ Levenberg-Marquardt, Bayesian Regularization และ Scaled Conjugate Gradient ดังรูปที่ 3.19



รูปที่ 3.19 หน้าต่างแสดงการเลือกประเภทต้นแบบทางคณิตศาสตร์

7) ทำการสอน (Train) เพื่อสร้างตัวแบบสำหรับการทำนาย โดยจะได้ค่าความคลาดเคลื่อนกำลังสอง (MSE) ของการสอนในแต่ละครั้งเพื่อใช้ในการตัดสินใจเลือกแบบจำลองที่ดีที่สุด ดังรูปที่ 3.20

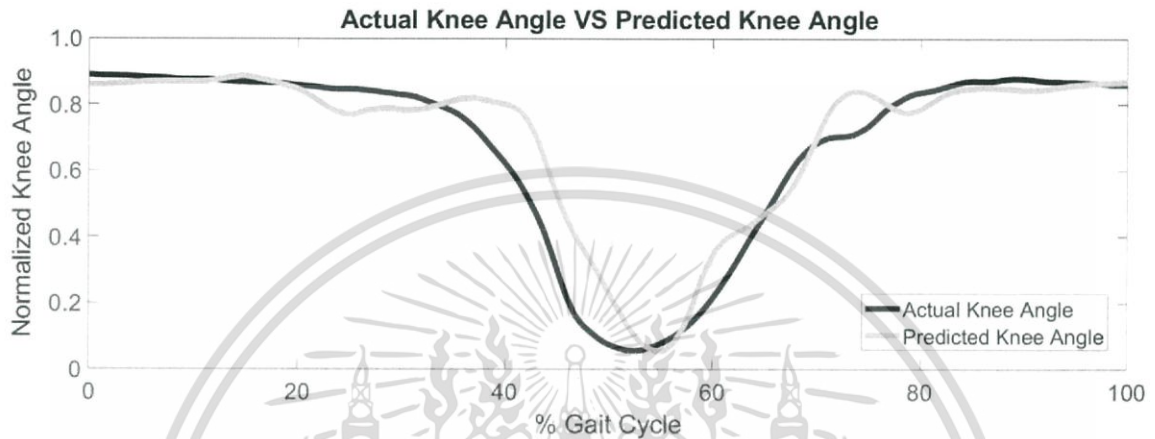


รูปที่ 3.20 หน้าต่างแสดงการสอนของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) บันทึกค่ามุมงอข้อเข่าจากการทำนาย และนำมาเปรียบเทียบกับค่ามุมงอข้อเข่าที่เกิดขึ้นจริง ดังรูป

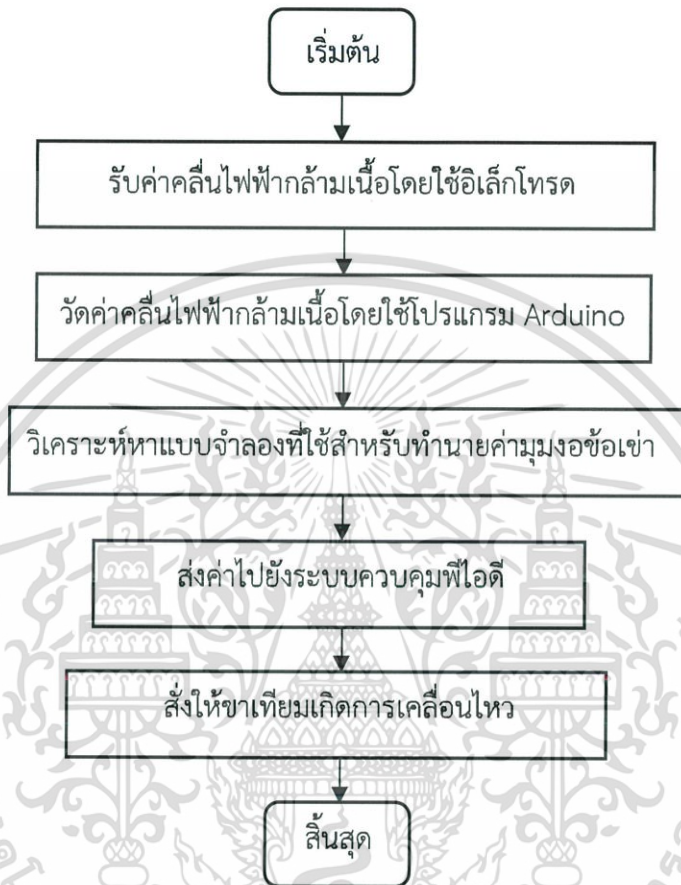
ที่ 3.21



รูปที่ 3.21 ค่าผลลัพธ์มุมงอข้อเข่าที่เกิดจากการสอนของโปรแกรม (เส้นสีเทา) เปรียบเทียบกับค่ามุมงอข้อเข่าจริง (เส้นสีดำ)

3.2.2 การเขียนคำสั่งเพื่อควบคุมการเคลื่อนไหวของขาเทียมต้นแบบ

ภายหลังจากการสร้างแบบจำลองโครงข่ายประสาทเทียม นำผลลัพธ์ที่ได้จากการทำนายใน MATLAB มาควบคุมการเคลื่อนไหวของขาเทียมต้นแบบ โดยเขียนคำสั่งเพื่อประมวลผลในโปรแกรม Arduino ให้สอดคล้องกับแบบจำลองโครงข่ายประสาทเทียมที่ได้ และใช้ระบบควบคุม PID ในการควบคุมการเคลื่อนไหว โดยแผนผังขั้นตอนการประมวลผลเพื่อควบคุมการเคลื่อนไหวของขาเทียมต้นแบบ เป็นดังรูปที่ 3.22 ดังนี้



รูปที่ 3.22 แผนผังการควบคุมการเคลื่อนไหวของขาเทียม

บทที่ 4

ผลการดำเนินงาน

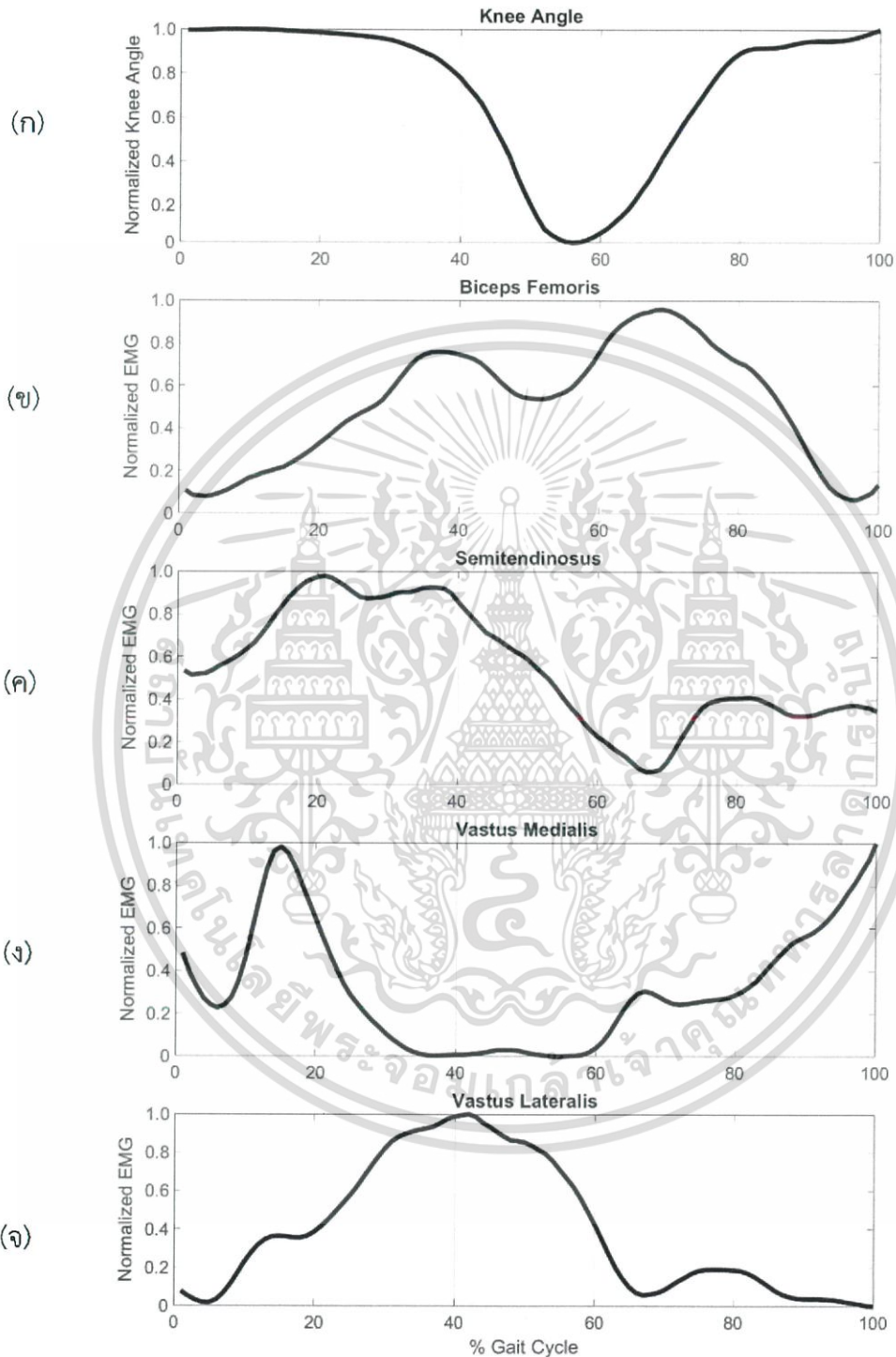
เมื่อจัดทำข่าเทียบต้นแบบรวมถึงเขียนโปรแกรมในการรับค่าข้อมูลและประมวลผลเพื่อควบคุมการเคลื่อนไหวของขาเทียบต้นแบบแล้ว ทางทีมผู้วิจัยได้ทำการทดสอบผลจากการดำเนินงาน โดยแบ่งขั้นตอนทดสอบการดำเนินงานวิจัยเป็น 2 ส่วน ดังนี้

- 4.1 มุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อที่เกิดขึ้นในขณะเดิน
- 4.2 แบบจำลองที่ใช้สำหรับทำนายค่ามุมงอข้อเข่า

4.1 มุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อที่เกิดขึ้นในขณะเดิน

จากการทดลองเพื่อเก็บค่ามุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อที่วัดได้จากกล้ามเนื้อขาท่อนบน ซึ่งได้แก่ Biceps Femoris, Semitendinosus, Vastus Medialis และ Vastus Lateralis ที่เกิดขึ้นในระหว่างการเดิน ลักษณะของมุมงอข้อเข่าและคลื่นไฟฟ้ากล้ามเนื้อในระหว่างวงจรการเดิน 1 รอบ เป็นดังรูปที่ 4.1

จากการศึกษาเปรียบเทียบกับวงจรการเดิน (Gait Cycle) พบว่ากล้ามเนื้อ Biceps Femoris มีการทำงานที่ร้อยละ 30 ถึง 90 โดยจะมีการทำงานมากที่สุดในช่วงร้อยละ 70 ของวงจรการเดินตรงกับช่วงที่เริ่มเข้าสู่ระยะ Swing Phase ในขณะที่กล้ามเนื้อ Semitendinosus มีการทำงานมากในช่วงร้อยละ 20 ถึง 40 ของวงจรการเดินซึ่งตรงกับช่วง Mid Stance ไปจนถึงช่วง Heel Off ของระยะ Stance Phase ส่วนกล้ามเนื้อ Vastus Medialis จะเริ่มทำงานในช่วง Terminal Swing หรือช่วงร้อยละ 80 ถึง 90 ของวงจรการเดิน และมีการทำงานมากที่สุดในช่วงประมาณร้อยละ 10 ของวงจรการเดิน นอกจากนี้ กล้ามเนื้อ Vastus Lateralis จะมีการทำงานในช่วงร้อยละ 20 ถึง 60 ของวงจรการเดิน ซึ่งตรงกับช่วง Mid Stance ไปจนกระทั่งช่วง Toe Off ซึ่งเป็นระยะสิ้นสุด Stance Phase จะเห็นได้ว่าผลการทดลองที่ได้มีค่าคลื่นไฟฟ้ากล้ามเนื้อของกล้ามเนื้อทั้ง 4 มัดที่ใกล้เคียงกับงานวิจัยที่ผ่านมา ๆ มาดังที่ได้กล่าวไปในหัวข้อที่ 2.5



รูปที่ 4.1 มุมงอข้อเข่าในวงจรการเดิน 1 รอบ (ก) คลื่นไฟฟ้ากล้ามเนื้อจากกล้ามเนื้อ Biceps Femoris (ข) Semitendinosus (ค) Vastus Medialis (ง) และ Vastus Lateralis (จ)

4.2 แบบจำลองโครงข่ายประสาทเทียมสำหรับทำนายขนาดมูกของข้อเข่าที่เหมาะสม

จากการสร้างแบบจำลองโครงข่ายประสาทเทียมสำหรับการทำนายขนาดมูกของข้อเข่าที่จะเกิดขึ้นในระหว่างการเดิน ผลการทำนายโดยการเปรียบเทียบมูกของข้อเข่าจริงกับมูกของข้อเข่าที่ได้จากแบบจำลองและจำนวนชั้นซ่อนต่าง ๆ เป็นดังนี้

4.2.1 การเปรียบเทียบมูกของข้อเข่าจากแบบจำลองแต่ละประเภทเพื่อหาแบบจำลองที่เหมาะสมที่สุด

แบบจำลองที่ใช้ในการทดสอบ คือ Levenberg-Marquardt, Bayesian Regularization และ Scaled Conjugate Gradient โดยกำหนดให้ค่าเริ่มต้นของชั้นซ่อนเป็น 3 ชั้น และทำการสอนซ้ำ 3 ครั้ง พบว่ามูกของข้อเข่าที่ได้จากแบบจำลอง Levenberg-Marquardt ให้ค่าความคลาดเคลื่อนที่น้อยที่สุดเมื่อเปรียบเทียบกับแบบจำลองอื่น ดังแสดงในตารางที่ 4.1 โดยเมื่อสังเกตจากลักษณะกราฟของค่ามูกของข้อเข่า จะเห็นได้ว่ามูกของข้อเข่าจากแบบจำลอง Levenberg-Marquardt ในทุกการทำซ้ำ มีลักษณะที่ใกล้เคียงกับค่ามูกของข้อเข่าจริงมากที่สุด แสดงดังรูปที่ 4.2

ตารางที่ 4.1 ค่าความคลาดเคลื่อนกำลังสอง (MSE) ในแบบจำลองประเภทต่าง ๆ

แบบจำลอง	โครงสร้าง	ค่าความคลาดเคลื่อนกำลังสอง (MSE)			
		ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	เฉลี่ย
Levenberg-Marquardt	32-3-1	0.4478	0.4351	0.4494	0.4441
Bayesian Regularization	32-3-1	0.5295	0.5138	0.5155	0.5196
Scaled Conjugate Gradient	32-3-1	0.5224	0.5249	0.5301	0.5258

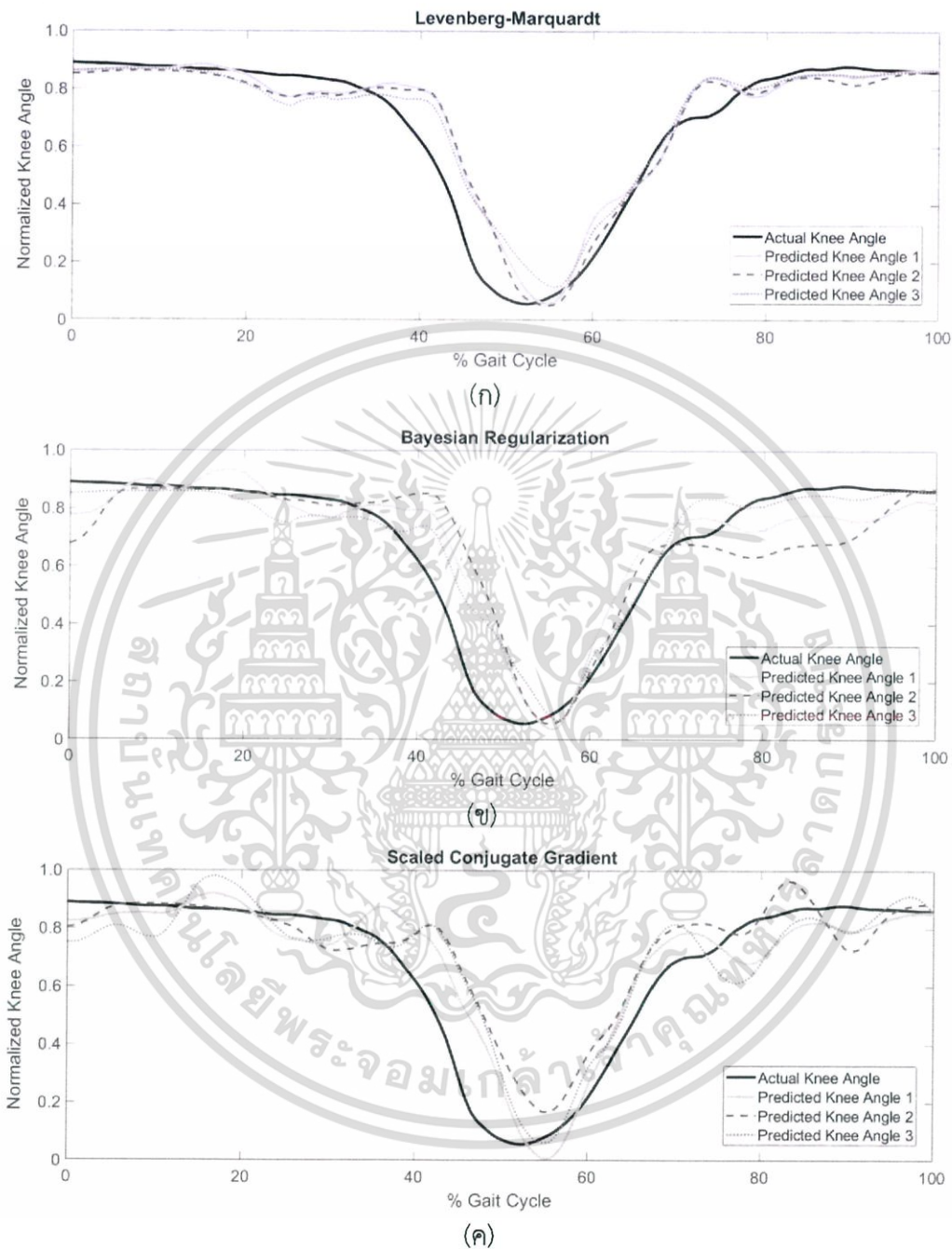
ค่าความคลาดเคลื่อนกำลังสอง (MSE) ของแบบจำลอง สามารถคำนวณได้จากสมการที่ 4.1

$$MSE = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n} \quad (4.1)$$

เมื่อ y_i คือ ค่ามูกข้อเข่าจริง

\hat{y}_i คือ ค่ามูกข้อเข่าที่ได้จากแบบจำลอง

n คือ จำนวนข้อมูล



รูปที่ 4.2 ผลการเปรียบเทียบระหว่างมุมมองข้อเข่าจริงกับมุมมองข้อเข่าจากแบบจำลอง Levenberg-Marquardt (ก) Bayesian Regularization (ข) และ Scaled Conjugate Gradient (ค)

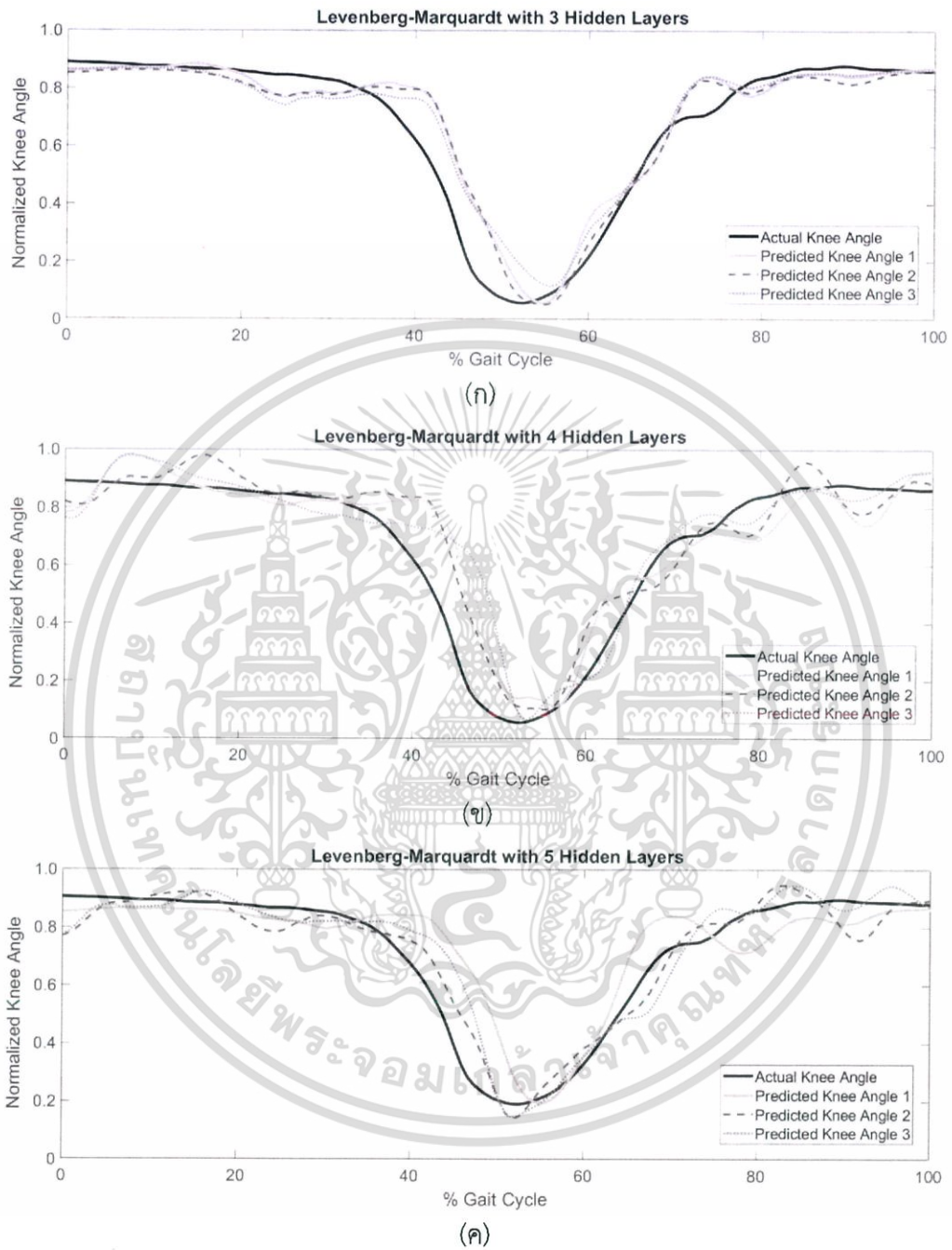
4.2.2 การเปรียบเทียบมุมมองข้อเข้าจากแบบจำลองที่มีจำนวนชั้นซ่อนต่าง ๆ เพื่อหาแบบจำลองที่เหมาะสมที่สุด

จากผลการเปรียบเทียบประเภทแบบจำลองในหัวข้อที่ 4.2.1 ได้แบบจำลองที่เหมาะสมที่สุดคือ Levenberg-Marquardt จากนั้นจึงได้เปรียบเทียบจำนวนชั้นซ่อนภายในแบบจำลองนี้ โดยทำการทดสอบด้วยชั้นซ่อนจำนวน 3, 4 และ 5 ชั้น ตามลำดับ และทำการทดสอบซ้ำ 3 ครั้ง พบว่า พบว่าค่าความคลาดเคลื่อนกำลังสอง (MSE) ของแบบจำลอง Levenberg-Marquardt ที่มีชั้นซ่อนจำนวน 3 ชั้น มีค่าน้อยกว่าแบบจำลองที่มีชั้นซ่อน 4 และ 5 ชั้น แสดงดังตารางที่ 4.2 สอดคล้องกับลักษณะกราฟของมุมมองข้อเข้าจากแบบจำลองที่มีชั้นซ่อนจำนวนต่าง ๆ โดยจะสังเกตได้ว่ามุมมองข้อเข้าจากแบบจำลองที่มี 3 ชั้นซ่อนมีลักษณะกราฟที่ใกล้เคียงกับค่ามุมมองข้อเข้าจริงมากที่สุด แสดงดังรูปที่ 4.3

ตารางที่ 4.2 ค่าความคลาดเคลื่อนกำลังสอง (MSE) ในแบบจำลองที่มีจำนวนชั้นซ่อนต่าง ๆ

แบบจำลอง	โครงสร้าง	จำนวนชั้นซ่อน (Hidden Layer)	ค่าความคลาดเคลื่อนกำลังสอง (MSE)			
			ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	เฉลี่ย
Levenberg-Marquardt	32-3-1	3	0.4478	0.4351	0.4494	0.4441
	32-4-1	4	0.4876	0.4782	0.4739	0.4799
	32-5-1	5	0.4763	0.4594	0.4611	0.4656

อย่างไรก็ตาม เพื่อให้เกิดความถูกต้องในการหาแบบจำลองที่เหมาะสมมากยิ่งขึ้น ผู้จัดทำปริญญาานิพนธ์จึงได้ทำการเปรียบเทียบแบบจำลอง Bayesian Regularization และ Scaled Conjugate Gradient ที่มีชั้นซ่อน 3, 4 และ 5 ชั้นด้วย เพื่อตรวจสอบให้แน่ชัดว่าแบบจำลองที่ได้มาเป็นโครงข่ายที่ดีที่สุด โดยสามารถสรุปผลลัพธ์การวิเคราะห์ข้อมูลค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (MSE) สำหรับการทดสอบแต่ละแบบจำลอง ดังตารางที่ 4.3



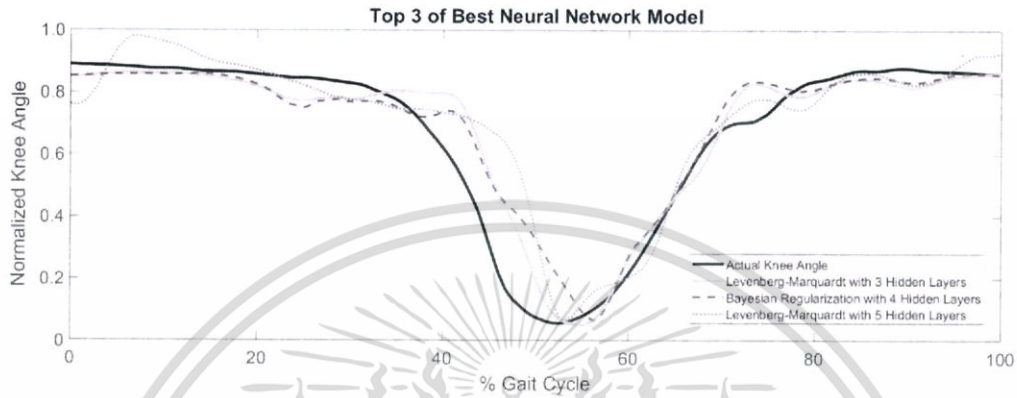
รูปที่ 4.3 ผลการเปรียบเทียบระหว่างมุมข้อเข่าจริงกับมุมข้อเข่าจากแบบจำลอง Levenberg-Marquardt ที่มีชั้นซ่อน 3 ชั้น (ก) 4 ชั้น (ข) และ 5 ชั้น (ค)

ตารางที่ 4.3 แบบจำลองโครงข่ายประสาทเทียมที่เหมาะสม

แบบจำลอง	โครงสร้าง	จำนวนชั้นซ่อน (Hidden Layer)	ค่าความคลาดเคลื่อนกำลัง สองเฉลี่ย (MSE)
Levenberg Marquardt	32-3-1	3	0.4441
	32-4-1	4	0.4799
	32-5-1	5	0.4656
Bayesian Regularization	32-3-2	3	0.5196
	32-4-2	4	0.4500
	32-5-2	5	0.5199
Scaled Conjugate Gradient	32-3-3	3	0.5258
	32-4-3	4	0.5295
	32-5-3	5	0.5375

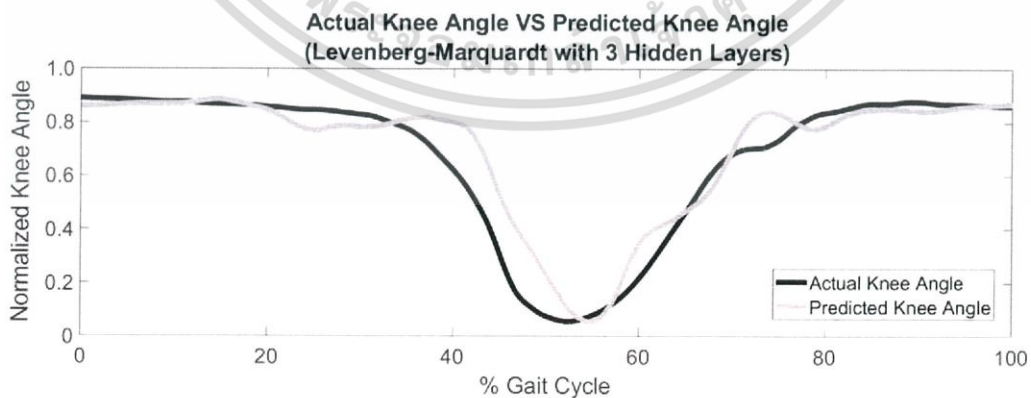
จากตารางที่ 4.3 พบว่าแบบจำลองโครงข่ายประสาทเทียมที่มีค่าความคลาดเคลื่อนกำลังสองเฉลี่ย น้อยที่สุด ได้แก่ แบบจำลอง Levenberg-Marquardt ซึ่งมีจำนวนชั้นซ่อน (Hidden Layer) 3 ชั้น นอกจากนี้ เมื่อเปรียบเทียบแบบจำลอง Levenberg-Marquardt ซึ่งมีจำนวนชั้นซ่อน 4 และ 5 ชั้น พบว่ามีค่าความคลาดเคลื่อนกำลังสองเฉลี่ยใกล้เคียงกันและอยู่ในช่วงที่สามารถยอมรับได้ โดยเมื่อพิจารณาแบบจำลอง Bayesian Regularization ที่มีชั้นซ่อน 4 ชั้น จะเห็นได้ว่าแบบจำลองดังกล่าวมีค่าความคลาดเคลื่อนกำลังสองน้อยกว่าแบบจำลอง Levenberg-Marquardt ซึ่งมีจำนวนชั้นซ่อน 4 ชั้น และ 5 ชั้น รวมถึงมีค่าใกล้เคียงกับแบบจำลอง Levenberg-Marquardt ที่มีจำนวนชั้นซ่อน 3 ชั้นด้วยเช่นกัน ในขณะที่ค่าความคลาดเคลื่อนกำลังสองของแบบจำลอง Bayesian Regularization ที่มีชั้นซ่อน 3 ชั้น และ 5 ชั้น ไม่อยู่ในช่วงที่สามารถยอมรับได้ เช่นเดียวกับแบบจำลอง Scaled Conjugate Gradient ซึ่งมีชั้นซ่อน 3, 4 และ 5 ชั้น โดยผลการเปรียบเทียบค่ามุมมองข้อเข้าที่ได้จากแบบจำลองที่มีค่าความคลาดเคลื่อนกำลังสองเฉลี่ยน้อยที่สุด 3 อันดับแรก เรียงจากน้อยไปมาก ได้แก่ แบบจำลอง Levenberg-Marquardt ที่มีจำนวนชั้นซ่อน 3 ชั้น แบบจำลอง Bayesian

Regularization ที่มีชั้นซ่อน 4 ชั้น และแบบจำลอง Levenberg-Marquardt ที่มีจำนวนชั้นซ่อน 4 ชั้น ตามลำดับ เป็นไปดังรูปที่ 4.4



รูปที่ 4.4 การเปรียบเทียบมุมมองข้อเข่าที่ได้จากแบบจำลอง Levenberg-Marquardt ที่มีชั้นซ่อน 3 ชั้น แบบจำลอง Bayesian Regularization ที่มีชั้นซ่อน 4 ชั้น และแบบจำลอง Levenberg-Marquardt ที่มีจำนวนชั้นซ่อน 4 ชั้น

จากผลการเปรียบเทียบ สามารถสรุปได้ว่าแบบจำลองที่เหมาะสมกับตัวแปรอิสระทั้ง 32 ตัว (ค่าคลื่นไฟฟ้ากล้ามเนื้อจากกล้ามเนื้อ 4 มัดซึ่งเก็บข้อมูลย้อนหลัง 8 ค่า) และตัวแปรตาม 1 ตัว (ค่ามุมมองข้อเข่า) คือ แบบจำลอง Levenberg-Marquardt โดยมีจำนวนชั้นซ่อน (Hidden Layer) 3 ชั้น ซึ่งให้ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยน้อยที่สุดคือ 0.4441 โดยผลการเปรียบเทียบระหว่างค่ามุมมองข้อเข่าจริง (เส้นสีดำ) กับค่ามุมมองข้อเข่าที่ได้จากแบบจำลอง (เส้นสีเทา) แสดงดังรูปที่ 4.5



รูปที่ 4.5 การเปรียบเทียบระหว่างค่ามุมมองข้อเข่าจริงกับค่ามุมมองข้อเข่าที่ได้จากแบบจำลอง

4.2.3 การนำแบบจำลองมาใช้ควบคุมการเคลื่อนไหวของขาเทียม

ภายหลังจากการออกแบบแบบจำลองโครงข่ายประสาทเทียมที่เหมาะสมที่สุดสำหรับการทำนายขนาดมุมงอข้อเข่าที่จะเกิดขึ้นในระหว่างการเดิน โครงสร้างแบบจำลองโครงข่ายประสาทเทียม Levenberg-Marquardt ซึ่งมีชั้นซ่อน 3 ชั้น ที่ได้จากโปรแกรม MATLAB ดังรูปที่ 4.6 ประกอบไปด้วยสมการสำหรับการคำนวณค่าพารามิเตอร์ต่าง ๆ ดังนี้

1) การคำนวณในชั้นข้อมูลป้อนเข้า (Input)

$$Xp1 = \text{mapminmax_apply}(X\{1,ts\},x1_step1) \quad (4.1)$$

โดยที่ function $y = \text{mapminmax_apply}(x,settings)$ (4.2)

$$y = \text{bsxfun}(@minus,x,settings.xoffset) \quad (4.3)$$

$$y = \text{bsxfun}(@times,y,settings.gain) \quad (4.4)$$

$$y = \text{bsxfun}(@plus,y,settings.ymin) \quad (4.5)$$

2) การคำนวณในชั้นที่ 1 (Layer 1)

$$a1 = \text{tansig_apply}(\text{repmat}(b1,1,Q) + IW1_1*Xp1) \quad (4.6)$$

$$l = IW1_1*Xp1 \quad (4.7)$$

โดยที่ function $a = \text{tansig_apply}(n,~)$ (4.8)

$$a = 2 ./ (1 + \exp(-2*n)) - 1 \quad (4.9)$$

3) การคำนวณในชั้นที่ 2 (Layer 2)

$$a2 = \text{repmat}(b2,1,Q) + LW2_1*a1 \quad (4.10)$$

$$L=LW2_1*a1 \quad (4.11)$$

4) การคำนวณในชั้นข้อมูลนำออก (Output)

$$Y\{1,ts\} = \text{mapminmax_reverse}(a2,y1_step1)$$

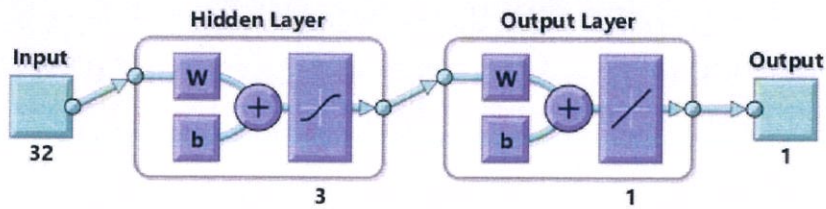
$$Y\{1,ts\} = Y\{1,ts\}'$$

โดยที่ function $x = \text{mapminmax_reverse}(y,settings)$ (4.12)

$$x = \text{bsxfun}(@minus,y,settings.ymin) \quad (4.13)$$

$$x = \text{bsxfun}(@rdivide,x,settings.gain) \quad (4.14)$$

$$x = \text{bsxfun}(@plus,x,settings.xoffset); \quad (4.15)$$



รูปที่ 4.6 แบบจำลองโครงข่ายประสาทเทียม Levenberg-Marquardt ซึ่งมีจำนวนชั้นซ่อน 3 ชั้น

ผู้จัดทำปริญญาพนธ์จึงได้นำสมการจากแบบจำลองโครงข่ายประสาทเทียมดังกล่าว มาประยุกต์ให้สามารถประมวลผลโดยใช้โปรแกรม Arduino ให้สอดคล้องกันได้ โดยสามารถเขียนแปลงสมการในแบบจำลองโครงข่ายประสาทเทียมมาเป็นคำสั่งได้ ดังนี้

```

for (j = 0; j < 32; j++)
    xp1[j] = ((X[j]-xoffsetinput[j])*xgain[j])-1;
for (i = 0; i < 3; i++)
{
    matrix1[i]=0;
    for (j = 0; j < 32; j++)
    {
        matrix1[i] += |W1_1[i][j]*xp1[j];
    }
}
for (i = 0; i < 3; i++)
{
    a1[i] = {2/(1+(exp(-2*(b1[i]+matrix1[i]))))-1};
}
for (i = 0; i < 1; i++)
{
    matrix2[i]=0;

```

```

for (j = 0; j < 3; j++)
{
    matrix2[i] += LW2_1[j] * a1[j];
}
}
a2 = b2 + matrix2[0];
Y = ((a2-yminoutput)/ygain)+xoffsetoutput;

```

ภายหลังจากทางทีมผู้วิจัยได้แบบจำลองโครงข่ายประสาทเทียมสำหรับการทำนายขนาดมุมงอข้อเข่าที่จะเกิดขึ้นในระหว่างการเดินที่เหมาะสมที่สุด และเขียนคำสั่งในโปรแกรม Arduino โดยใช้ระบบควบคุม PID ในการควบคุมการเคลื่อนไหวของขาเทียมต้นแบบให้สอดคล้องกับแบบจำลองโครงข่ายประสาทเทียมแล้ว จึงได้มีการนำขาเทียมต้นแบบมาทดลองใช้งานโดยรับคลื่นไฟฟ้ากล้ามเนื้อเนื่องจากคนปกติ พบว่า สามารถนำคลื่นไฟฟ้ากล้ามเนื้อจากกล้ามเนื้อขาท่อนบน ซึ่งประกอบไปด้วยกล้ามเนื้อ Biceps Femoris, Semitendinosus, Vastus Medialis และ Vastus Lateralis มาควบคุมการเคลื่อนไหวของขาเทียมต้นแบบได้ แต่อย่างไรก็ตามการเคลื่อนไหวของขาเทียมต้นแบบยังคงมีความล่าช้า เนื่องจากมีข้อจำกัดทางเทคนิคของอุปกรณ์ที่ใช้ในการสร้างขาเทียมต้นแบบ ซึ่งส่งผลทำให้การทำงานของขาเทียมต้นแบบขณะเคลื่อนไหวเกิดความคลาดเคลื่อนได้ โดยการเคลื่อนไหวของขาเทียมต้นแบบสามารถแสดงได้ดังรูปที่ 4.7



รูปที่ 4.7 การเคลื่อนไหวของขาเทียมต้นแบบซึ่งควบคุมด้วยระบบควบคุม PID ตามมุมงอข้อเข่าในช่วงต่าง ๆ

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ปริญญานิพนธ์ฉบับนี้จัดทำขึ้นเพื่อออกแบบและพัฒนาต้นแบบขาเทียมควบคุมด้วยคลื่นไฟฟ้ากล้ามเนื้อ รวมถึงยังมีการพัฒนาขาเทียมสำหรับผู้พิการให้มีประสิทธิภาพในราคาต้นทุนที่ต่ำที่สุดเพื่อให้ผู้พิการที่มีรายได้น้อยสามารถเข้าถึงบริการขาเทียมได้ เนื่องจากปัจจุบันประเทศไทยยังนำเข้าจากต่างประเทศซึ่งมีราคาสูง รวมถึงเทคโนโลยีที่ใช้ควบคุมการทำงานของขาเทียมมักควบคุมโดยใช้การเคลื่อนไหวของร่างกายเท่านั้น ทำให้ผู้พิการมีความยากลำบากในการเดิน จึงเห็นถึงความสำคัญของขาเทียมสำหรับผู้พิการที่ถูกตัดขา รวมถึงต้องการลดช่องว่างระหว่างการเข้าถึงบริการขาเทียมกับผู้พิการที่มีรายได้น้อยจึงได้พัฒนาขาเทียมต้นแบบขึ้น โดยใช้คลื่นไฟฟ้ากล้ามเนื้อ (EMG) จากกล้ามเนื้อขาที่นอนบนมาควบคุมการเคลื่อนไหวของขาเทียม มีการเลือกอุปกรณ์ที่จะนำมาใช้ทำขาเทียมต้นแบบและวงจรสำหรับเก็บข้อมูลคลื่นไฟฟ้ากล้ามเนื้อและมุงงอข้อเข่าระหว่างการเดิน จากนั้นนำข้อมูลมาวิเคราะห์หาแบบจำลองที่ใช้สำหรับทำนายค่ามุงงอข้อเข่าที่ดีที่สุด โดยใช้ MATLAB แบบจำลองที่เหมาะสมได้แก่ แบบจำลองของ Levenberg-Marquardt โดยใช้จำนวนชั้นซ่อนตัว 3 ชั้น และนำมาเขียนคำสั่งเพื่อควบคุมการเคลื่อนไหวของขาเทียมต้นแบบให้เป็นไปตามผลการทำนาย ซึ่งพบว่า รูปแบบของคลื่นไฟฟ้ากล้ามเนื้อที่วัดได้สามารถนำมาใช้ควบคุมการเคลื่อนไหวของขาเทียมได้ ถึงแม้จะมีความล่าช้ากว่าการเคลื่อนไหวจริงอยู่เล็กน้อย

5.2 ข้อจำกัด

จากการเคลื่อนไหวของขาเทียมต้นแบบ พบว่ายังคงมีความล่าช้ากว่าการเคลื่อนไหวจริงเล็กน้อย เนื่องจากมีข้อจำกัดทางเทคนิคของอุปกรณ์ที่ใช้ในการสร้างขาเทียมต้นแบบและรับค่าคลื่นไฟฟ้ากล้ามเนื้อ เช่น ความเร็วของ Linear Actuator Motor รวมถึงอิเล็กทรอนิกส์แบบแผ่นซึ่งยังไม่สามารถอ่านคลื่นไฟฟ้ากล้ามเนื้อในช่วงที่สัญญาณเบาได้ ซึ่งส่งผลทำให้การทำงานของขาเทียมต้นแบบขณะเคลื่อนไหวเกิดความคลาดเคลื่อนได้ อย่างไรก็ตาม จากผลการเปรียบเทียบมุงงอข้อเข่าที่ได้จากแบบจำลองกับมุงงอข้อเข่าจริงซึ่งยังมีความคลาดเคลื่อนอยู่เล็กน้อย อาจเป็นผลมาจากข้อจำกัดทางด้านโครงสร้างของแบบจำลองโครงข่ายประสาทเทียมที่ใช้สอน เนื่องจากการเลือกใช้จำนวนชั้นซ่อนที่ไม่ซับซ้อนเพื่อลดระยะเวลาในกระบวนการสอน

5.3 ข้อเสนอแนะ

ปรับปรุงชุดวงจรไฟฟ้าที่ใช้ในการประมวลผลข้อมูลให้มีความทนทานและปลอดภัย เพื่อให้สามารถทำงานได้ในระยะยาวและสามารถนำข้อมูลมาวิเคราะห์ได้อย่างง่ายดาย รวมถึงขนาดของกล่องที่ใช้บรรจุชุดวงจรไฟฟ้าสามารถทำให้มีขนาดเล็กลงกว่านี้ได้ ซึ่งจะทำให้มีน้ำหนักเบา พกพาง่ายและมีความสะดวกสบายในการใช้งานเพิ่มมากขึ้น นอกจากนี้การเลือกใช้อิเล็กทรอนิกส์แบบสัมผัสมีหวังควรเลือกประเภทที่มีความแม่นยำในการรับข้อมูล เนื่องจากรูปแบบคลื่นสัญญาณกล้ำเนื้อมีส่วนสำคัญอย่างมากต่อการนำมาประมวลผล และในกระบวนการสอน (Training) ของส่วนของการเรียนรู้ของเครื่อง (Machine Learning) ควรเพิ่มขนาดของข้อมูลที่ป้อนเข้าไป หรือเพิ่มจำนวนชั้นซ่อนให้มีความมากขึ้น เพื่อให้ค่ามุมข้อเหวี่ยงที่ได้จากการประมวลผลในแบบจำลองใกล้เคียงกับค่ามุมข้อเหวี่ยงจริงมากที่สุด

5.4 งานในอนาคต

ในอนาคต อาจมีการปรับเปลี่ยน Linear Actuator Motor ที่ใช้ควบคุมการเคลื่อนไหวให้มีความเร็วมากขึ้น เพื่อให้สามารถตอบสนองต่อการเคลื่อนไหวได้รวดเร็วและเป็นธรรมชาติตามความต้องการของผู้ใช้งาน โดยอาจต้องมีการออกแบบชุดวงจรไฟฟ้าให้มีน้ำหนักเบาและมีขนาดกะทัดรัด ง่ายต่อการพกพา เพื่อให้ผู้ใช้สามารถสวมใส่ขาเทียมได้อย่างสะดวกสบายมากขึ้น รวมถึงต้องมีการออกแบบชุดวงจรให้มีความปลอดภัยต่อการใช้งานมากขึ้น นอกจากนี้ อาจต้องมีการพัฒนาในส่วนของกระบวนการเรียนรู้ในแบบจำลองโครงข่ายประสาทเทียม เพื่อให้สามารถทำนายลักษณะการเคลื่อนไหวได้อย่างแม่นยำมากยิ่งขึ้น โดยควรต้องมีการกรองและขยายสัญญาณข้อมูลผลลัพธ์ก่อนนำไปควบคุมขาเทียมต้นแบบ อย่างไรก็ตาม ในปริญญาโทฉบับนี้เป็นเพียงการศึกษาในขั้นเริ่มต้น โดยให้ความสำคัญกับการเคลื่อนที่ของข้อเหวี่ยงในแนวแกนเดียวเพียงเท่านั้น ซึ่งในอนาคตควรต้องศึกษาเพิ่มเติมทั้งในส่วนของการเคลื่อนที่ของข้อเหวี่ยงและข้อเท้า เพื่อให้ผู้ใช้สามารถหมุนข้อเหวี่ยงหรือระดกข้อเท้าเพื่อให้สามารถเคลื่อนที่ได้อย่างเป็นธรรมชาติมากขึ้น

เอกสารอ้างอิง

- [1] สำนักงานสถิติแห่งชาติ, “สำรวจความพิการ.” [Online]. Available: <http://service.nso.go.th/nso/web/survey/surpop2-5-4.html>.
- [2] กระทรวงสาธารณสุขศูนย์สิรินธรเพื่อการฟื้นฟูสมรรถภาพทางการแพทย์แห่งชาติ กรมการแพทย์, “คุณภาพชีวิตคนพิการขาขาด,” 2557.
- [3] ทิพาภรณ์ เยสุรธรรม, ปรีชา อุปโยคินเทอดชัย ชีวะเกตุ, “คนพิการขาขาดไทย : กระบวนการตีตราและการปรับตัว,” vol. 22, no. 2, pp. 51–57, 2012.
- [4] ชมรมนักพัฒนาอุตสาหกรรมไทย, “ข้อเปรียบเทียบของขาเทียมที่ประดิษฐ์คิดค้น,” 2555. [Online]. Available: <http://www.thailegs.com/l5compai.htm>.
- [5] พิพิธภักดีจักรพงษ์, “เทคโนโลยีขาเทียมและสถานการณ์การวิจัยพัฒนาขาเทียมในประเทศไทย,” 2551.
- [6] Ottobock, “International C-Leg Studies International C-Leg Studies.”
- [7] R. J. M. N Mohd Nor, Nur Khalidah Zakaria, “Analysis of EMG Signals during Walking of Healthy Children,” vol. 76, no. Iris, pp. 316–322, 2015.
- [8] “คลื่นไฟฟ้ากล้ามเนื้อ,” มหาวิทยาลัยบูรพา.
- [9] N. Kunju, N. Kumar, D. Pankaj, A. Dhawan, and A. Kumar, “EMG Signal Analysis for Identifying Walking Patterns of Normal Healthy Individuals,” no. March, pp. 118–122, 2009.
- [10] สมชาย รัตนทองคำ, “การตรวจประสาท - กล้ามเนื้อด้วยไฟฟ้า,” 2537.
- [11] สมชาย รัตนทองคำ, คู่มือการกระตุ้นด้วยกระแสไฟฟ้าความถี่ต่ำ: ปฏิบัติการและการประยุกต์ใช้ทางคลินิก. ขอนแก่น, 2537.
- [12] กาญจนกุลภาณุพงศ์, “การศึกษาทางชีวกลศาสตร์ของแรงในกล้ามเนื้อขาของผู้ป่วยโรคอัมพาตครึ่งซีก,” 2554.
- [13] “Muscle of the Thigh.” [Online]. Available: <https://basicmedicalkey.com/gluteal-region-and-posterior-thigh/>.

- [14] C. Kongkhiaw, "Video System for Dynamic Motion Analysis of Human Gait," no. 1, 2010.
- [15] J. Perry, "Gait analysis: normal and pathological function," pp. 3–7, 1992.
- [16] สมชาย ทองประเสริฐ, "การฟื้นฟูผู้ป่วยที่มีปัญหาการเดินที่ผิดปกติ," pp. 1–7, 2549.
- [17] "ประเภทของขาเทียม." [Online]. Available: <http://artitayacenter.blogspot.com/2013/02/d.html>.
- [18] C. Engineering, "Introduction to Microcontrollers," 2007.
- [19] ชัยพร ใจแก้ว, อภิรักษ์ จันทรสร้าง, "ไมโครคอนโทรลเลอร์เบื้องต้น," no. 01204223.
- [20] D. Honeywell, "PID Control," 2000.
- [21] R. P. Lippmann, "An Introduction ' to Computing with Neural Nets."
- [22] M. Nielsen, "Neural Networks and Deep Learning."
- [23] A. M. Elazouni, "Unsupervised-Learning Neural Networks," vol. 132, no. 12, pp. 1242–1254, 2007.
- [24] D. Kriesel, "Neural Networks," pp. 3895–3902, 2006.
- [25] R. A. Bogey and L. A. Barnes, "Computer Algorithms to Characterize Individual Subject EMG Profiles During Gait," vol. 73, no. September, 1992.
- [26] J. Perry, "The Contribution of Dynamic Electromyography to Gait Analysis," pp. 33–48, 2005.
- [27] E. Isakov, H. Burger, J. Krajnik, M. Gregoric, and C. Marincek, "KNEE MUSCLE ACTIVITY DURING AMBULATION OF TRANS-TIBIAL AMPUTEES," no. 2, pp. 196–199, 2001.
- [28] R. J. M.N Mohd Nor, Nur Khalidah Zakaria, "Analysis of EMG Signals during Walking of Healthy Children," vol. 76, no. Iris, pp. 316–322, 2015.
- [29] E. Isakov, O. Keren, and N. Benjuya, "Trans-tibial amputee gait : time-distance parameters and EMG activity," pp. 216–220, 2000.
- [30] E. C. Wentink, E. C. Prinsen, J. S. Rietman, and P. H. Veltink, "Comparison of muscle activity patterns of transfemoral amputees and control subjects during walking," 2013.

- [31] T. Latif, C. M. Ellahi, and T. A. Choudhury, "Design of a cost-effective EMG driven bionic leg Design of a Cost effective EMG driven Bionic Leg," no. January, 2008.
- [32] S. Huang and D. P. Ferris, "Muscle activation patterns during walking from transtibial amputees recorded within the residual limb-prosthetic interface," pp. 1-16, 2012.
- [33] L. Resnik, H. H. Huang, A. Winslow, D. L. Crouch, F. Zhang, and N. Wolk, "Evaluation of EMG pattern recognition for upper limb prosthesis control : a case study in comparison with direct myoelectric control," pp. 1-13, 2018.



ภาคผนวก

คำสั่งสำหรับควบคุมการเคลื่อนไหวของขาเทียม

```
// Part1 PID Variable
#include <PID_v1.h>
#define PIN_INPUT A0 //sensor input feedback
#define PIN_OUTPUT1 8 //in3
#define PIN_OUTPUT2 9 //in4
double Input, Output;
bool extend;
double Kp=100000, Ki=1000, Kd=2000; //Specify the links and initial tuning
parameters
float last_error = 0;
float error = 0;
float changeError = 0;
float totalError = 0;
float pidTerm = 0;

// Part2 Machine Learning Variable
float Y;
float xp1[32];
float a1[3];
float a2;
int i;
int j;
int h;
int k;

// EMG
float SensorValue1 = analogRead(A4);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

float SensorValue2 = analogRead(A5);
float SensorValue3 = analogRead(A6);
float SensorValue4 = analogRead(A7);
float X[32];
float sum1;
float sum2;
float sum3;
float sum4;
float mean1;
float mean2;
float mean3;
float mean4;
int count = 0;
float SensorShift1;
float SensorShift2;
float SensorShift3;
float SensorShift4;
float SensorMax1 = 0;
float SensorMax2 = 0;
float SensorMax3 = 0;
float SensorMax4 = 0;
float SensorMin1 = 0;
float SensorMin2 = 0;
float SensorMin3 = 0;
float SensorMin4 = 0;
float SensorNorm1;
float SensorNorm2;
float SensorNorm3;
float SensorNorm4;

```

```
// Input 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

float xoffsetinput [32] = { -1, -1, -0.894528965546468, -0.505259873486217,
                             -1, -1, -0.894528965546468, -0.505259873486217,
                             -1, -1, -0.894528965546468, -0.505259873486217,
                             -1, -1, -0.894528965546468, -0.505259873486217,
                             -1, -1, -0.894528965546468, -0.505259873486217,
                             -1, -1, -0.894528965546468, -0.505259873486217,
                             -1, -1, -0.894528965546468, -0.505259873486217,
                             -1, -1, -0.894528965546468, -0.505259873486217};

float xgain [32] = { 1.01983853306182, 1.01778086218158, 1.05567137603679,
                    1.32867422777168, 1.01983853306182, 1.01778086218158,
                    1.05567137603679, 1.32867422777168, 1.01983853306182,
                    1.01778086218158, 1.05567137603679, 1.32867422777168,
                    1.01983853306182, 1.01778086218158, 1.05567137603679,
                    1.32867422777168, 1.01983853306182, 1.01778086218158,
                    1.05567137603679, 1.32867422777168, 1.01983853306182,
                    1.01778086218158, 1.05567137603679, 1.32867422777168,
                    1.01983853306182, 1.01778086218158, 1.05567137603679,
                    1.32867422777168, 1.01983853306182, 1.01778086218158,
                    1.05567137603679, 1.32867422777168 };

```

// Layer 1

```

float b1[3] = { -5.5842682391693720945, 0.90720871834365479636,
               -1.0198167105711350988 };

```

```

float IW1_1[3][32] = { {4.4812081988614140826, -4.2800209811303258078,
                       -2.4549132717652759617, -0.064781680401894276011,
                       -0.34639419808429799952, 1.5184620410217815056,
                       0.092881415885237739172, -1.353146232513937397,
                       -2.5081710998187798367, 1.8425633160412129286,
                       0.7590312668211753877, -0.49193046603780293813,
                       -2.8338862497805328289, 0.89602289827772796382,
                       2.1835449384015679364, -2.957721283722011929,

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-0.60161030000774440296, 1.1125571912972485045,
1.8958978781724227236, -0.394437118995989211,
0.59391252565495489257, -1.5436067771157329087,
2.664039996060633797, -1.3168392149291692217,
3.7537345317107679854, -0.40435050797186300464,
-1.5479686734263018444, 1.1102167030604164211,
6.5445311474068867952, 0.066999887936905311303,
-0.7369407931681797308, 1.6636844981352714168 },
{1.6048937603231727689, -4.4559165665959907798,
4.0754245951597170006, 3.1880894743759906618,
1.1258547202635267315, 0.3227747040321308214,
-0.34399829523498670625, -0.47486244061608989808,
-0.2112379066004891337, 1.8055023734939170676,
0.81899496910780444914, -1.0333090319538722479,
-1.9855928618787275219, 1.6174899838127549945,
1.9398336902232440337, 0.27182712505260264724,
3.031892364912264437, 1.158303699749061666,
1.2217870974810671747, -1.1766177459454842946,
-3.3776879712073837325, -0.45155162034180756914,
1.4290301788891546231, 2.6157252093331138099,
-1.7795501915527509595, 0.37962453890569086701,
-1.1402605752906900261, 2.3991043957752231641,
10.89229769342481724, -1.9581617955316650281,
-7.4147216981787416046, -4.9403463405571041989},
{-6.8988938930402934346, 3.3088808538785645652,
3.7140932560402015739, 2.0166625328476288637,
3.8863155346929767475, -0.81485978619934629208,
-0.72557330746553772016, -1.0878097633441994319,
1.3780688846647826118, -0.52372590297630272627,
0.46918213871538894111, -0.28290318542273684699,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

3.346537214850447306, -0.72638623837030524832,
0.033116261502615068624, 0.27116029011319192632,
-0.0055643531940012119774, -0.1762551035864867166,
-0.49002178821245234719, -0.38614144885627099679,
-1.6018480311875902267, 0.03465057054066861264,
-1.4100585454069194391, 0.30121695877443915856,
-5.5649867541373723157, 2.4631749086302172635,
-0.17904172887916303125, 1.4794475413092151861,
6.6112282119234224709, -6.5597775002443317049,
-0.55903613531571660911, 0.11476621779137442803 } };

```

```
// Layer 2
```

```
float b2 = 0.29704351736527023764;
```

```
float LW2_1[3] = { 0.24118559227733699091, -0.17601921338594528343,
-0.51116869345137705416};
```

```
// Output 1
```

```
int yminoutput = -1;
```

```
float ygain = 1.40094263518568;
```

```
int xoffsetoutput = -1;
```

```
float matrix1 [3];
```

```
float matrix2 [1];
```

```
// Calculate Angle
```

```
float Angle;
```

```
float Dist;
```

```
float Sensor;
```

```
void setup() {
```

```
    Serial.begin(9600);
```

```
//initialize the variables we're linked to
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 5
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Input = analogRead(PIN_INPUT);
pinMode(A0, INPUT);
pinMode(PIN_OUTPUT1, OUTPUT);
pinMode(PIN_OUTPUT2, OUTPUT);

//turn the PID on
myPID.SetMode(AUTOMATIC);
}
void loop() {
//Part1 Machine Learning
// Recieved EMG input
float SensorValue1 = analogRead(A4);
float SensorValue2 = analogRead(A5);
float SensorValue3 = analogRead(A6);
float SensorValue4 = analogRead(A7);

//Shift sensorValue to zero by minus mean
    count++;
    sum1 += SensorValue1;
    sum2 += SensorValue2;
    sum3 += SensorValue3;
    sum4 += SensorValue4;
    mean1 = (sum1)/count;
    mean2 = (sum2)/count;
    mean3 = (sum3)/count;
    mean4 = (sum4)/count;
    SensorShift1 = SensorValue1 - mean1;
    SensorShift2 = SensorValue2 - mean2;
    SensorShift3 = SensorValue3 - mean3;
    SensorShift4 = SensorValue4 - mean4;

```

```

//Normalize Value
if (SensorShift1 > SensorMax1){
    SensorMax1 = SensorShift1;
}
if (SensorShift1 < SensorMin1){
    SensorMin1 = SensorShift1;
}
if (SensorShift2 > SensorMax2){
    SensorMax2 = SensorShift2;
}
if (SensorShift2 < SensorMin2){
    SensorMin2 = SensorShift2;
}
if (SensorShift3 > SensorMax3){
    SensorMax3 = SensorShift3;
}
if (SensorShift3 < SensorMin3){
    SensorMin3 = SensorShift3;
}
if (SensorShift4 > SensorMax4){
    SensorMax4 = SensorShift4;
}
if (SensorShift4 < SensorMin4){
    SensorMin4 = SensorShift4;
}
if ( abs(SensorMax1)>abs(SensorMin1)){
    SensorNorm1 = SensorShift1/abs (SensorMax1);
}
else if ( abs(SensorMax1)<abs(SensorMin1)){
    SensorNorm1 = SensorShift1/abs (SensorMin1);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if ( abs(SensorMax2)>abs(SensorMin2)){
    SensorNorm2 = SensorShift2/abs (SensorMax2);
}
else if ( abs(SensorMax2)<abs(SensorMin2)){
    SensorNorm2 = SensorShift2/abs (SensorMin2);
}
if ( abs(SensorMax3)>abs(SensorMin3)){
    SensorNorm3 = SensorShift3/abs (SensorMax3);
}
else if ( abs(SensorMax3)<abs(SensorMin3)){
    SensorNorm3 = SensorShift3/abs (SensorMin3);
}
if ( abs(SensorMax4)>abs(SensorMin4)){
    SensorNorm4 = SensorShift4/abs (SensorMax4);
}
else if ( abs(SensorMax4)<abs(SensorMin4)){
    SensorNorm4 = SensorShift4/abs (SensorMin4);
}

//Keep sensornorm in array[32]
for (i=0; i<28; i++)
{
    X[i] = X[i+4];
}

X [28] = SensorNorm1;
X [29] = SensorNorm2;
X [30] = SensorNorm3;
X [31] = SensorNorm4;

```

// model machine learning

```
for (j = 0; j < 32; j++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

xp1[j] =((X[j]-xoffsetinput[j])*xgain[j])-1;
for (i = 0; i < 3; i++)
{
matrix1[i]=0;
for (j = 0; j < 32; j++)
{
matrix1[i] += IW1_1[i][j]*xp1[j];
}
}
for (i = 0; i < 3; i++)
{
a1[i] = {2/(1+(exp(-2*(b1[i]+matrix1[i]))))-1};
}
for (i = 0; i < 1; i++)
{
matrix2[i]=0;
for (j = 0; j < 3; j++)
{
matrix2[i] += LW2_1[j] * a1[j];
}
}
a2 = b2 + matrix2[0];
Y =((a2-yminoutput)/ygain)+xoffsetoutput;

```

```

// calculate Y into angle/distance/sensor : MaxEncoder = 0.14, MinEncoder = -
51.19, MaxModel = 0.27, MinModel = -0.74

```

```

if ( Y > 0.27 ){
Y = 0.27;
}
else if ( Y < -0.74 ){
Y = -0.74;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
Angle = (0.14-(((0.27-Y)*51.33)/1.01))*(-1);
Dist = (-0.06276328*Angle)+24.6260639;
Sensor = (43.2653061*Dist)-419.714286;

```

```
//Part2 PID (Prosthesis Movement)
```

```

Input = analogRead(PIN_INPUT);
Serial.println(Sensor-50);
Serial.print("\t");
//PIDcalculation(); // find PID value
//void PIDcalculation()
{
error = (Sensor-50) - Input;
changeError = error - last_error; // derivative term
totalError += error; //accumalate errors to find integral term
pidTerm = (Kp * error) + (Ki * totalError) + (Kd * changeError); //total gain
pidTerm = constrain(pidTerm, 398, 610); //constraining to appropriate value
last_error = error;
}
if (Input < pidTerm){
digitalWrite(PIN_OUTPUT1,LOW);
digitalWrite(PIN_OUTPUT2,HIGH);
extend = true;
}
else if (Input > pidTerm){
digitalWrite(PIN_OUTPUT1, HIGH);
digitalWrite(PIN_OUTPUT2, LOW);
extend = false;
}
delay(200);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งสำหรับรับค่าคลื่นไฟฟ้ากล้ามเนื้อและมุมข้อเข่า

```
// Encoder Variable
#define outputA A3
#define outputB A2
#define outputZ A1
int counter = 0;
int aState;
int aLastState;
int timer1_counter;
int in_z;

void setup() {
// interrupt
// initialize timer1
noInterrupts(); // disable all interrupts
TCCR1A = 0;
TCCR1B = 0;
timer1_counter = 64494; // preload timer 65536-16MHz/256/60Hz
pinMode (outputA,INPUT);
pinMode (outputB,INPUT);
pinMode (outputZ,INPUT);
Serial.begin (38400);
// Reads the initial state of the outputA
aLastState = digitalRead(outputA);
TCNT1 = timer1_counter; // preload timer
TCCR1B |= (1 << CS12); // 256 prescaler (CS12=1)
TIMSK1 |= (1 << TOIE1); // enable timer overflow interrupt
interrupts(); // enable all interrupts
}
```

```

ISR(TIMER1_OVF_vect)                                // interrupt service routine
{
  // EMG Input
  int SensorValue1 = analogRead(A4);
  int SensorValue2 = analogRead(A5);
  int SensorValue3 = analogRead(A6);
  int SensorValue4 = analogRead(A7);

  TCNT1 = timer1_counter;                          // preload timer
  digitalWrite(LED_BUILTIN, digitalRead(LED_BUILTIN) ^ 1);

  // print the results to the Serial Monitor
  Serial.println(String(SensorValue1)+' '+String(SensorValue2)+' '+String(SensorValue3)+' '+
    String(SensorValue4)+' '+String(counter*0.072));
}

void loop() {
  aState = digitalRead(outputA);                    // Reads the "current" state of the outputA
  in_z = digitalRead(outputZ);
  // If the previous and the current state of the outputA are different, that means a
  Pulse has occurred
  if (aState != aLastState){
  // If the outputB state is different to the outputA state, that means the encoder is
  rotating clockwise
    if (digitalRead(outputB) != aState) {
      counter ++;
    } else {
      counter --;
    }
  }
  aLastState = aState;
}

```

```

// Updates the previous state of the outputA with the current state
    if(in_z == 0)
    {
        counter = 0;
    }
}

```

Model of machine learning

//ค่าคงที่ใน Neural network

//Input

```

x1_step1.xoffset = [ -1, -1, -0.894528965546468, -0.505259873486217,
                    -1, -1, -0.894528965546468, -0.505259873486217,
                    -1, -1, -0.894528965546468, -0.505259873486217,
                    -1, -1, -0.894528965546468, -0.505259873486217,
                    -1, -1, -0.894528965546468, -0.505259873486217,
                    -1, -1, -0.894528965546468, -0.505259873486217,
                    -1, -1, -0.894528965546468, -0.505259873486217,
                    -1, -1, -0.894528965546468, -0.505259873486217 ];
x1_step1.gain = [ 1.01983853306182, 1.01778086218158, 1.05567137603679,
                 1.32867422777168, 1.01983853306182, 1.01778086218158,
                 1.05567137603679, 1.32867422777168, 1.01983853306182,
                 1.01778086218158, 1.05567137603679, 1.32867422777168,
                 1.01983853306182, 1.01778086218158, 1.05567137603679,
                 1.32867422777168, 1.01983853306182, 1.01778086218158,
                 1.05567137603679, 1.32867422777168, 1.01983853306182,
                 1.01778086218158, 1.05567137603679, 1.32867422777168,
                 1.01983853306182, 1.01778086218158, 1.05567137603679,
                 1.32867422777168, 1.01983853306182, 1.01778086218158,
                 1.05567137603679, 1.32867422777168 ];
x1_step1.ymin = -1;

```

//Layer 1

b1 = [-5.5842682391693720945, 0.90720871834365479636,
-1.0198167105711350988] ;

W1_1 = [[4.4812081988614140826, -4.2800209811303258078,
-2.4549132717652759617, -0.064781680401894276011,
-0.34639419808429799952, 1.5184620410217815056,
0.092881415885237739172, -1.353146232513937397,
-2.5081710998187798367, 1.8425633160412129286,
0.7590312668211753877, -0.49193046603780293813,
-2.8338862497805328289, 0.89602289827772796382,
2.1835449384015679364, -2.957721283722011929,
-0.60161030000774440296, 1.1125571912972485045,
1.8958978781724227236, -0.394437118995989211,
0.59391252565495489257, -1.5436067771157329087,
2.664039996060633797, -1.3168392149291692217,
3.7537345317107679854, -0.40435050797186300464,
-1.5479686734263018444, 1.1102167030604164211,
6.5445311474068867952, 0.066999887936905311303,
-0.7369407931681797308, 1.6636844981352714168],
[1.6048937603231727689, -4.4559165665959907798,
4.0754245951597170006, 3.1880894743759906618,
1.1258547202635267315, 0.3227747040321308214,
-0.34399829523498670625, -0.47486244061608989808,
-0.2112379066004891337, 1.8055023734939170676,
0.81899496910780444914, -1.0333090319538722479,
-1.9855928618787275219, 1.6174899838127549945,
1.9398336902232440337, 0.27182712505260264724,
3.031892364912264437, 1.158303699749061666,
1.2217870974810671747, -1.1766177459454842946,
-3.3776879712073837325, -0.45155162034180756914,
1.4290301788891546231, 2.6157252093331138099,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-1.7795501915527509595, 0.37962453890569086701,  
-1.1402605752906900261, 2.3991043957752231641,  
10.89229769342481724, -1.9581617955316650281,  
-7.4147216981787416046, -4.9403463405571041989],  
[-6.8988938930402934346, 3.3088808538785645652,  
3.7140932560402015739, 2.0166625328476288637,  
3.8863155346929767475, -0.81485978619934629208,  
-0.72557330746553772016, -1.0878097633441994319,  
1.3780688846647826118, -0.52372590297630272627,  
0.46918213871538894111, -0.28290318542273684699,  
3.346537214850447306, -0.72638623837030524832,  
0.033116261502615068624, 0.27116029011319192632,  
-0.0055643531940012119774, -0.1762551035864867166,  
-0.49002178821245234719, -0.38614144885627099679,  
-1.6018480311875902267, 0.03465057054066861264,  
-1.4100585454069194391, 0.30121695877443915856,  
-5.5649867541373723157, 2.4631749086302172635,  
-0.17904172887916303125, 1.4794475413092151861,  
6.6112282119234224709, -6.5597775002443317049,  
-0.55903613531571660911, 0.11476621779137442803]] ;
```

```
//Layer 2
```

```
b2 = 0.29704351736527023764;
```

```
LW2_1= [ 0.24118559227733699091, -0.17601921338594528343,  
-0.51116869345137705416 ];
```

```
// Output 1
```

```
y1_step1.ymin = -1;
```

```
y1_step1.gain = 1.40094263518568;
```

```
y1_step1.xoffset = -1;
```

```

//แบบจำลอง
//การจัดรูปแบบของ Input
isCellX = iscell(X);
if ~isCellX
    X = {X};
end

//ขนาดของตัวแปร
TS = size(X,2); // timesteps
if ~isempty(X)
    Q = size(X{1},1); // samples/series
else
    Q = 0;
end

//การจัดรูปแบบของ Output
Y = cell(1,TS);

//Time loop
for ts=1:TS
    // Input 1
    X{1,ts} = X{1,ts}; // tranpose input
    Xp1 = mapminmax_apply(X{1,ts},x1_step1)
    // โปรแกรมรับค่า Input เข้ามา ซึ่งถูกกำหนดเป็นค่า X
    // คำนวณหาค่า Xp1 จากสมการ  $xp1 = (X - xoffsetinput) * xgain - 1$ 

    // Layer 1
    a1 = tansig_apply(repmat(b1,1,Q) + IW1_1*Xp1);
    l = IW1_1*Xp1;
    // คำนวณหาค่า a1 จากสมการ  $a1 = 2 / (1 + \exp(-2 * (b1 + IW1_1 * xp1))) - 1$ 

```

```

// Layer 2
a2 = repmat(b2,1,Q) + LW2_1*a1;
L=LW2_1*a1;
// คำนวณหาค่า a2 จากสมการ a2 = b2 + LW2_1 * a1

// Output 1
Y{1,ts} = mapminmax_reverse(a2,y1_step1);
Y{1,ts} = Y{1,ts}';
// คำนวณหาค่า Y จากสมการ Y = ((a2 - yminoutput) / ygain) + xoffsetoutput
end

// Final Delay States
Xf = cell(1,0);
Af = cell(2,0);

// การจัดรูปแบบของ Output
if ~isCellX
    Y = cell2mat(Y);
end
end

// Functions
// Map Minimum and Maximum Input Processing Function
function y = mapminmax_apply(x,settings)
y = bsxfun(@minus,x,settings.xoffset);
y = bsxfun(@times,y,settings.gain);
y = bsxfun(@plus,y,settings.ymin);
end

// Sigmoid Symmetric Transfer Function
function a = tansig_apply(n,~)

```

```
a = 2 ./ (1 + exp(-2*n)) - 1;
```

```
end
```

```
// Map Minimum and Maximum Output Reverse-Processing Function
```

```
function x = mapminmax_reverse(y,settings)
```

```
x = bsxfun(@minus,y,settings.ymin);
```

```
x = bsxfun(@rdivide,x,settings.gain);
```

```
x = bsxfun(@plus,x,settings.xoffset);
```

```
end
```

