

บทคัดย่อ

ในกีฬาทุกประเภทการฝึกนั้นเป็นสิ่งที่สำคัญมากเพื่อผลลัพธ์ในการเล่น ซึ่งผู้จัดทำได้เล็งเห็นการประยุกต์ใช้เทคโนโลยีกับการฝึกฝนกีฬาประเภทที่ใช้มือ โครงการนี้จึงมีวัตถุประสงค์เพื่อจัดทำอุปกรณ์เพื่อฝึกฝนการจัดท่าทางของมือและเพิ่มทักษะ TGMD-2 เพื่อผลลัพธ์ของการเล่นและแข่งขันกีฬาประเภทใช้มือ โดยการทำอุปกรณ์ที่ช่วยในการจัดท่าทางของมือผู้ใช้ ซึ่งจะสามารถตรวจจับการเคลื่อนที่ของมือผู้ใช้งานได้ โดยสามารถตรวจจับได้โดยตรวจการเคลื่อนที่ทั้งหมด 4 แบบ ได้แก่การเคลื่อนที่เชิงมุม 3 แบบ คือ หันมือขึ้น-ลง หันมือไปทางซ้าย-ขวา และหมุนรอบมือ และการเคลื่อนที่เชิงเส้น 1 แบบคือการเคลื่อนของยกมือ ขึ้นบน-ล่าง โดยเทียบกับพื้นโลก โดยใช้เซ็นเซอร์วัดระยะทาง เซ็นเซอร์วัดความเร่ง และเซ็นเซอร์เข็มทิศวัดค่าการเคลื่อนที่แล้วนำมาคำนวณและประมวลผลในไมโครคอนโทรลเลอร์แล้วจึงแจ้งเตือนทิศการเคลื่อนที่ต่างๆโดยใช้ไฟ LED แจ้งเตือนเมื่อเกิดการเคลื่อนที่ในแกนนั้นๆ และอุปกรณ์สามารถจดจำท่าทางที่ต้องการฝึกได้ จากผลการดำเนินการพบว่าอุปกรณ์นั้นสามารถช่วยเพิ่มประสิทธิภาพการเล่นของผู้ใช้งานได้มากกว่าการฝึกฝนแบบธรรมดาที่ไม่มีเครื่องมือช่วย

คำสำคัญ: ฝึกฝนการจัดท่าทางมือ, ทักษะ TGMD-2, ตรวจจับการเคลื่อนที่

Abstract

In every sport, training is especially important for results. We saw the application of technology to the practice of sports that use hands. The objective of this project is to create equipment to practice hand postures and increase TGMD-2 skills for the results of playing and competing in hand sports. By making a device that helps with the user's hand posture. which can detect the movement of the user's hand by detecting 4 types of movement, including 3 types of angular movement, namely turning the hand up-down Turning your hand left-right, spinning around the hand and one type of linear movement is the movement of the hand up-down relative to the earth. Using a distance, sensor, Accelerometer sensor and the compass sensor measure the movement value calculates and processes it in the microcontroller and then notifies various movement directions using LED lights to notify when movement in each axis occurs. And the device can remember the postures you want to practice. From the results, the equipment was able to increase the user's playing efficiency more than ordinary practice without any equipment.

Keywords: hand gesture training, TGMD-2 skills, motion detection

บทคัดย่อ.....	ก
Abstract.....	ก
สารบัญ.....	ข
1. บทนำ	1
1.1 ที่มาและความสำคัญของโครงงาน	1
1.2 วัตถุประสงค์ของโครงงาน.....	2
1.3 ขอบเขตของโครงงาน	2
1.4 ผลลัพธ์ที่คาดหวังจากโครงงาน	2
1.5 ขั้นตอนการดำเนินงาน.....	3
2. หลักการและทฤษฎีที่เกี่ยวข้อง.....	3
2.1 อุปกรณ์.....	3
2.1.1 บอร์ด Seeed Studio XIAO nRF52840.....	4
2.1.2 VL53L1X (Distance Sensor).....	5
2.1.3 GY-271 (Compass Sensor)	4
2.1.4 LSM6DS3 (Accelerometer).....	5
2.2 สมการและทฤษฎีที่เกี่ยวข้อง.....	6
2.2.1 การคำนวณการเคลื่อนที่เชิงมุมแบบ Roll	8
2.2.2 การคำนวณการเคลื่อนที่เชิงมุมแบบ Pitch	8
2.2.3 มุมอาซิมุท.....	
3. ผลลัพธ์จากการดำเนินการเบื้องต้น	9
3.1 โครงสร้างและอุปกรณ์ที่ใช้.....	9
3.1.1 Block diagram.....	9
3.1.2 อุปกรณ์ที่ใช้.....	10
3.2 ข้อมูลเซ็นเซอร์.....	11
3.3 การทำงานของอุปกรณ์.....	12
3.4 วงจรของอุปกรณ์.....	15
3.5 การใช้งานของอุปกรณ์.....	16
3.5.1 การใช้งานโหมดฮิสเตอร์.....	18
3.5.2 การใช้งานโหมดจดจำ.....	19
3.6 การทดสอบการใช้งานของอุปกรณ์.....	22
4. บทสรุป	25
4.1 สรุปผลการดำเนินการ.....	25
4.2 ปัญหา อุปสรรค และแนวทางแก้ไข	25

5. กิตติกรรมประกาศ.....	26
6. เอกสารอ้างอิง.....	27
7. ภาคผนวก.....	28
ภาคผนวก ก.....	28
ภาคผนวก ข.....	29

1. บทนำ

1.1 ที่มาและความสำคัญของโครงการ

ในกีฬาทุกประเภทการฝึกเป็นสิ่งที่สำคัญมากเพื่อผลลัพธ์ในการแข่งขัน ในโลกทุกวันนี้เทคโนโลยีได้มีส่วนมาช่วยในการแก้ปัญหาต่างๆในโลกมนุษย์ การฝึกกีฬาก็เช่นกัน โดยทางผู้จัดทำได้เล็งเห็นความสำคัญในการใช้เทคโนโลยีมาช่วยเสริมประสิทธิภาพในการฝึกกีฬาของนักกีฬาเพื่อผลลัพธ์ที่ดีขึ้น โดยทางผู้จัดทำมีเป้าหมายในการพัฒนาอุปกรณ์เพื่อสำหรับฝึกการจัดท่าทางของมือและแขนในกีฬาประเภทที่ต้องใช้มือในการเล่นหรือพัฒนาทักษะทางกีฬาที่ชื่อว่า TGMD-2 (Throwing, Catching and Grasping) [4] อาทิเช่น ในกีฬาเทเบิลเทนนิสที่ต้องมีการฝึกจับไม้และการวางท่าทางของมือในการเตรียมความพร้อมสำหรับตีลูกบอล โดยจากผลการศึกษาในบทความ [1] บอกว่าการประสานระหว่างตาและการตั้งท่ามือมีผลต่อความแม่นยำการเสิร์ฟแบบ back spin 25.3% ในบทความ[2] เกี่ยวกับความเร็วของลูกเทนนิสหลังจากการกระทบกับไม้แรกเกิดเมื่อใช้ตำแหน่งต่างๆในการตี เราสามารถใช้เทคโนโลยีนี้ในการฝึกซ้อมเพื่อให้ นักกีฬาสามารถจดจำการจับไม้และวางตำแหน่งของแขนและมือ เพื่อให้การตีลูกเทนนิสมีประสิทธิภาพมากที่สุด โดยการจับไม้เพื่อให้ตีโดนตรงจุดศูนย์กลางของไม้ไม้แรกเกิดและการจัดท่าทางในการตีโดย Racquet trajectory angle, Racquet-face impact angle และ Racquet rotation angle เป็น 16.5, 85.5 และ 0.8 องศาตามลำดับจะทำให้ลูกเทนนิสนั้นมีความเร็วมากที่สุดที่ 136 km/h อีกตัวอย่างจากบทความ [3] ซึ่งศึกษาเกี่ยวกับตัวแปรที่ส่งผลกระทบต่อการเล่นในกีฬาฟุตซอลโดยมีมุมในการมุมในการจับและตำแหน่งในการจับที่ส่งผลกระทบต่อการเล่นนั้น ซึ่งการวางท่ามือเพื่อให้อยู่ในมุมที่เหมาะสมนั้นมีผลอย่างมากต่อระยะทางที่แขนจะพุ่งไปได้ โดยในบทความพบว่าเมื่อมุมในการปล่อยแขนออกจากมือที่ค่า 45.6 องศา จะทำให้แขนพุ่งไปได้ไกลที่สุดที่ 90.33 เมตรซึ่งทางผู้จัดทำคาดว่าการศึกษาการฝึกการจับมือให้อยู่ในมุมที่เหมาะสมจะช่วยส่งเสริมประสิทธิภาพในการเล่นได้อย่างมาก

จากปัญหาที่กล่าวมาข้างต้นทางผู้จัดทำจึงได้จัดทำโครงการนี้ขึ้นมาเพื่อช่วยให้เพิ่มประสิทธิภาพในการเล่นกีฬาและผลลัพธ์ที่ดีในการแข่งขัน โดยโครงการนี้นั้นมีเป้าหมายที่จะพัฒนาระบบในการช่วยในการตั้งท่ามือและแขนเพื่อให้ นักกีฬาตั้งท่าแขนและมืออย่างเหมาะสมโดยเพื่อให้ร่างกายเกิด muscle memory และนักกีฬาสามารถตั้งท่าแขนและมือได้อย่างเหมาะสมโดยอัตโนมัติ โดยอุปกรณ์ที่พัฒนาขึ้นมาจะประกอบด้วย microcontroller สำหรับคำนวณและประมวลผล โดยทางผู้จัดทำจะใช้บอร์ด Seeed Studio XIAO nRF52840 และประกอบด้วยเซนเซอร์ต่างๆที่จับการเคลื่อนไหวที่ในระนาบและแกนที่สำคัญ โดยมี Accelerometer ที่ติดตั้งมากับบอร์ดวัดความเร่งของตัวอุปกรณ์ และนำมาใช้สำหรับคำนวณเพื่อวัดการเคลื่อนไหวที่เชิงมุม(Yaw, Pitch และ Roll), distance sensor เพื่อวัดระยะห่างของตัวอุปกรณ์กับตัวผู้ใช้, GY-271 เพื่อใช้ในการกำหนดทิศและแกนของอุปกรณ์ ในการสร้างอุปกรณ์ของโครงการนี้ขึ้นมาในขนาดที่ถือด้วยมือได้ โดยจะบรรจุอยู่ในรูปแบบทรงกล่องสี่เหลี่ยมโดยวัสดุที่ใช้บรรจุนั้นจะเป็นพลาสติกเพื่อให้สะดวกในการติดกับมือโดยในขั้นตอนการใช้งานจริงผู้ใช้งานจะต้องทรง

ตัวให้อยู่หนึ่งเพื่อฝึกฝนท่าทางที่ต้องการฝึก เมื่อเกิดการขยับตัวอุปกรณ์นั้นจะส่งสัญญาณโดยให้ไฟ LED กระพริบเพื่อแจ้งเตือนให้ผู้ใช้งานให้กลับอยู่ในท่าทางที่ต้องการฝึกฝน

1.2 วัตถุประสงค์ของโครงการ

ในการจัดทำโครงการขึ้นมานั้น มีวัตถุประสงค์เพื่อการฝึกฝนในการซ้อมกีฬาประเภทต่างๆที่ใช้มือในการเล่นกีฬาและการฝึกฝนทักษะทาง TGMD-2 (Throwing, Catching and Grasping) โดยใช้อุปกรณ์ที่พัฒนามานี้สามารถตรวจจับการเคลื่อนไหวของตัวอุปกรณ์ในระนาบและแกนที่สำคัญเพื่อการฝึกฝนการตั้งท่าที่เหมาะสมในการฝึกกีฬานั้นๆ

1.3 ขอบเขตของโครงการ

โครงการนี้จะพัฒนาอุปกรณ์ Stationary box โดยต้นแบบมีข้อกำหนด (Specification) ดังนี้

1. ใช้หลักการในการตรวจจับการเปลี่ยนแปลงของระยะห่างระหว่างอุปกรณ์กับวัตถุในแกน X (Ventral และ Dorsal)
2. ตรวจจับการเปลี่ยนแปลงของระยะห่างระหว่างอุปกรณ์กับวัตถุในแกน Z (Cranial และ Caudal)
3. ตรวจจับการเปลี่ยนแปลงการหมุนของวัตถุแบบ Pitching
4. ตรวจจับการเปลี่ยนแปลงการหมุนของวัตถุแบบ Rolling
5. ตรวจจับการเปลี่ยนแปลงการหมุนของวัตถุแบบ Yawing
6. วัดผลโดยการทดลองใช้อุปกรณ์ในรูปแบบท่าทางต่างๆโดยจะทำวัดผลว่าผู้ทดลองใช้นั้น ก่อนใช้และหลังใช้อุปกรณ์มีผลลัพธ์ในการเล่นกีฬาที่ดีขึ้นหรือไม่

1.4 ผลลัพธ์ที่คาดหวังจากโครงการ

1. อุปกรณ์ที่สามารถตรวจจับการเคลื่อนที่ของตัวกล่องเองได้ทุกแนวในการเคลื่อนที่ที่เป็นระนาบที่สำคัญ
2. อุปกรณ์สามารถพัฒนาทักษะด้าน TGMD-2 ของผู้ใช้งานได้
3. ระบบของอุปกรณ์คำนวณค่าแกนต่างๆและค่าอื่นๆที่วัดได้จากเซ็นเซอร์ได้อย่างแม่นยำ
4. อุปกรณ์สามารถตั้งค่าให้สามารถจับอยู่ในท่าทางที่ผู้ใช้ต้องการได้

1.5 ขั้นตอนการดำเนินงาน

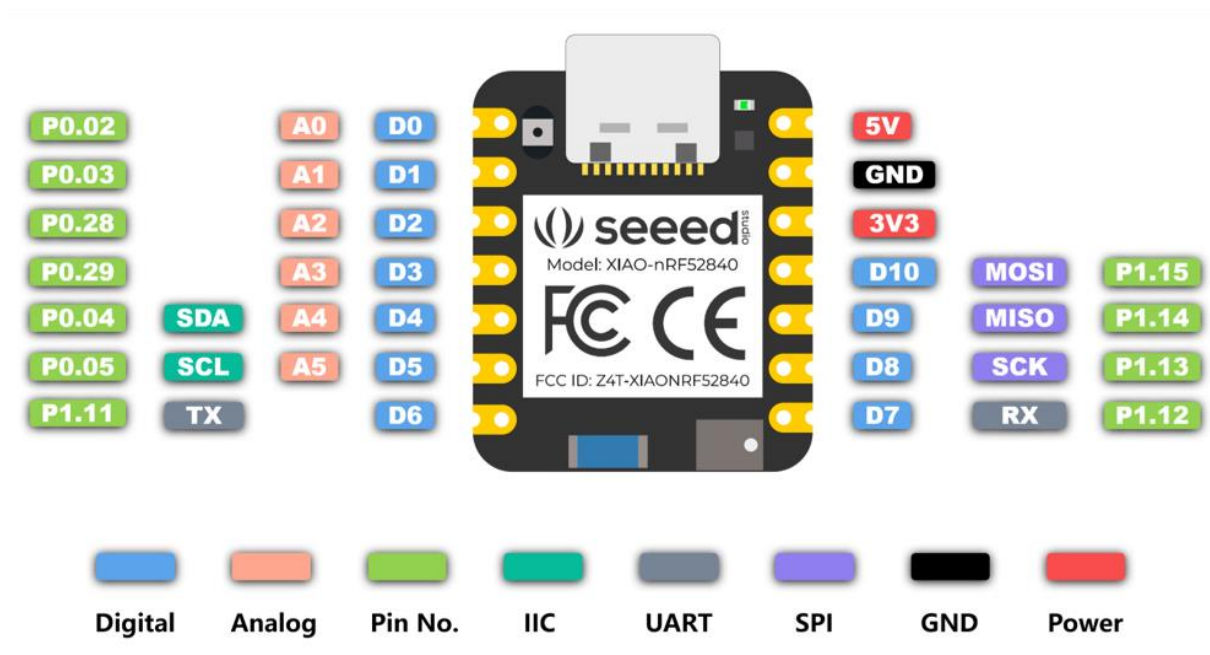
- 1.ศึกษาข้อมูลเกี่ยวกับทักษะของกีฬาประเภทใช้มือและทฤษฎีที่เกี่ยวข้อง
- 2.ออกแบบอุปกรณ์ วงจรของอุปกรณ์ และศึกษาเซนเซอร์ที่ใช้ในตัวเองาน
- 3.ออกแบบระบบการทำงานของอุปกรณ์ และศึกษาการเขียนภาษา C++
- 4.ทดสอบโปรแกรมของอุปกรณ์
- 5.ประกอบอุปกรณ์และวงจร
- 6.ทดสอบประสิทธิภาพการใช้งานของอุปกรณ์ในการฝึกท่าทาง

2. หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 อุปกรณ์

2.1.1 บอร์ด Seeed Studio XIAO nRF52840

บอร์ด Seeed Studio XIAO nRF52840 นั้นเป็นบอร์ดที่มี Nordic nRF52840 MCU เป็นหน่วยไมโครคอนโทรลเลอร์และเป็นบอร์ดที่มีขนาดเล็กที่ 21 x 17.5 ตารางมิลลิเมตร จึงสามารถนำใช้ทำอุปกรณ์สวมใส่กับ Internet of Things โดย pin ของบอร์ดจะมี PWM Pins 11 Pin และ Analog Pins 6 Pin ตัวบอร์ดสามารถใช้งานในระบบ Bluetooth 5.0 และ BLE function เพื่อลดการใช้พลังงานของบอร์ด โดยตัวบอร์ดนั้นมีเซนเซอร์ติดมากับบอร์ด นั่นคือ Accelerometer 6-axis Inertial Measurement Unit(IMU) ซึ่ง จะทำการวัดความเร่งของแกนทั้ง 6 แกนนั่นคือ X, Y, Z, Yaw, Pitch และ Roll และมี LED ที่ติดมาให้กับตัวบอร์ด ซึ่งในบอร์ดนี้สามารถใช้ภาษาคอมพิวเตอร์ในการเขียนคำสั่งได้ทั้งหมด 3 ภาษานั้นคือ Arduino MicroPython และ CircuitPython โดยบอร์ด Seeed Studio XIAO nRF52840 มีไลบรารีโดยเฉพาะของบอร์ดซึ่งผู้พัฒนาได้จัดทำขึ้นมาเพื่อให้ผู้ใช้งานนั้นคือ Seed nRF52 Boards สำหรับใช้งาน Low Energy Cost Function และ Bluetooth function และ Seeed nRF52 mbed-enabled Boards สำหรับการใช้งาน IMU&PDM advanced function สำหรับการใช้งาน Sensor ในตัวของบอร์ด



รูปที่ 1 Pins ของบอร์ด Seeed Studio XIAO nRF52840

2.1.2 VL53L1X (Distance Sensor)

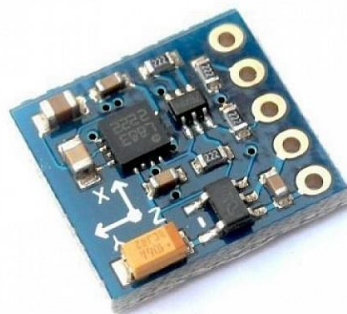
เซ็นเซอร์วัดระยะทางแบบอินฟราเรดเป็นเซ็นเซอร์ประเภทหนึ่งของเซ็นเซอร์วัดระยะทางโดยมีหลักการทำงานโดยส่งคลื่นอินฟราเรดซึ่งอยู่ในความถี่ที่มนุษย์ไม่สามารถเห็นได้จาก LED ไปยังวัตถุที่ต้องการวัดระยะหลังจากอินฟราเรดนั้นส่งไปยังวัตถุที่ต้องการวัดหลังจากนั้นอินฟราเรดจะถูกสะท้อนกลับมายังที่ตัวของเซ็นเซอร์และจะถูกตรวจจับแล้วนำไปคำนวณระยะทางโดยคำนวณจากความเร็วของสัญญาณอินฟราเรดกับระยะเวลาไป-กลับของสัญญาณอินฟราเรด ซึ่งเซ็นเซอร์วัดระยะทางแบบอินฟราเรดที่ผู้จัดทำเลือกใช้คือรุ่น VL53L1X ดังในรูปที่ 3 ซึ่งสามารถวัดระยะทางได้สูงสุด 4 เมตร และมีระยะมุมความกว้าง Field of view ในการวัดระยะทาง 27 องศา



รูปที่ 2 Infrared Distance Sensor

2.1.3 GY-271(Compass sensor)

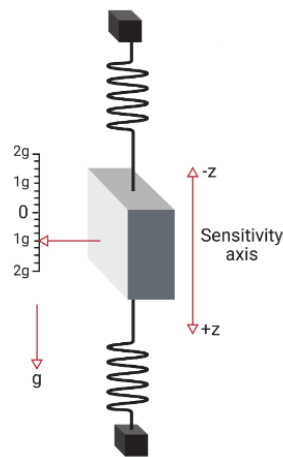
GY-271 นั้นเป็นเซ็นเซอร์ความเข้มแม่เหล็กจากสนามแม่เหล็กของโลก โดยเซ็นเซอร์ประเภทนี้จะถูกใช้งานอย่างมากในการทำหุ่นยนต์เนื่องด้วยเหตุผลการหาทิศทาง โดยเซ็นเซอร์เข็มทิศใช้หลักการ Hall Effect ในการวัดค่าสนามแม่เหล็กของโลกหลังจากนั้นค่าของสนามแม่เหล็กโลกจะถูกเปลี่ยนเป็นสัญญาณไฟฟ้าซึ่งจะถูกส่งไปประมวลผลกับไมโครโปรเซสเซอร์โดยจะคำนวณเทียบกับค่าสนามแม่เหล็กของขั้วโลกเหนือและส่งค่าเอาต์พุตออกมาเป็นค่ามุมอาซิมุท¹ และมุมของเข็มทิศที่เทียบกับทิศเหนือ



รูปที่ 3 GY-271 Compass Sensor

2.1.4 LSM6DS3(Accelerometer)

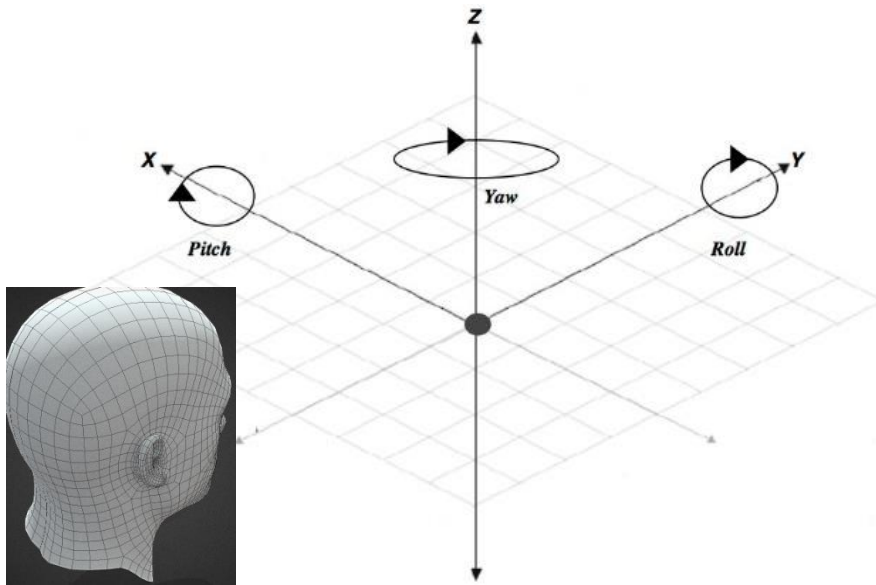
Accelerometer รุ่น LSM6DS3 ซึ่งเป็นรุ่นที่ผู้จัดทำนำมาใช้งาน เซ็นเซอร์วัดความเร่งเป็นเซ็นเซอร์ที่ใช้สำหรับการวัดความเร่งของวัตถุ โดยหลักการทำงานของเซ็นเซอร์รุ่น LSM6DS3 นั้นจะ MEMS Accelerometers โดยมีหลักการทำงานโดยตรวจจับแรงที่กระทำกับมวลในอุปกรณ์ที่เกิดการเปลี่ยนแปลงและวัดความเร่งในแกนดังกล่าว เช่น ตัวอย่างดังภาพที่ 4 ที่วัดความเร่งในแกน Z ซึ่งเซ็นเซอร์รุ่น LSM6DS3 นั้นสามารถวัดความเร่งทั้งหมด 3 แกน คือแกน X,Y,Z และยังมี Gyroscope ที่วัดอัตราเร็วเชิงมุมในแกนการเคลื่อนที่เชิงมุม 3 แกน



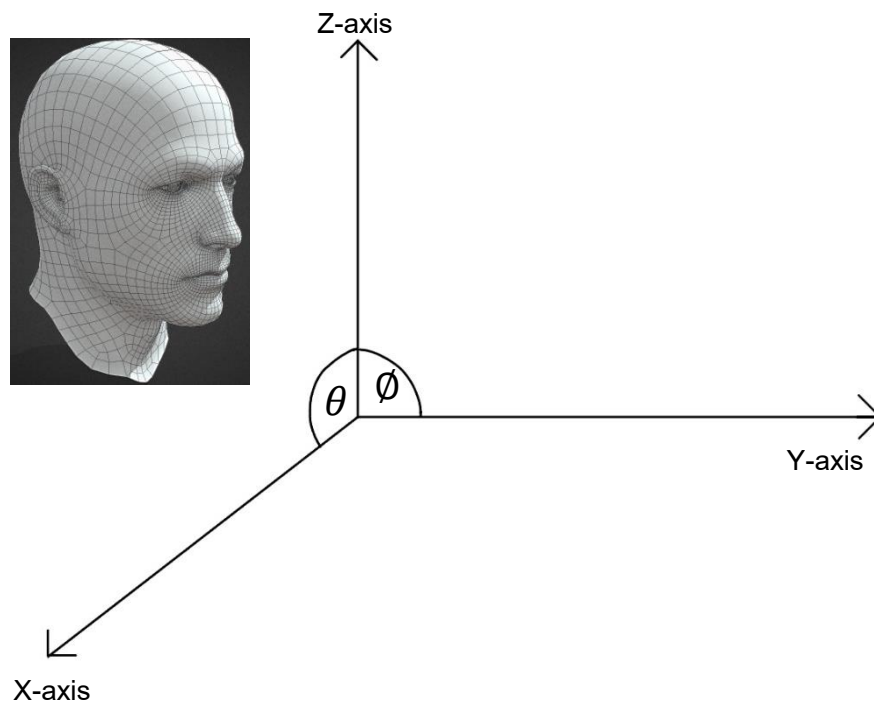
รูปที่ 4 การวัดความเร่งของ Accelerometer

2.2 การคำนวณและทฤษฎี

โดยในส่วนของเนื้อหานี้จะเป็นการอธิบายการคำนวณการเคลื่อนที่เชิงมุม 2 แกน คือ การเคลื่อนที่เชิงมุมแบบ Pitch และ Roll ดังภาพที่ 5 ซึ่งเป็นการเคลื่อนที่ที่ไม่สามารถใช้เซ็นเซอร์วัดได้โดยตรงดังนั้นผู้จัดทำจึงใช้ค่าความเร่งจาก Accelerometer นำมาคำนวณมุมของการเคลื่อนที่เชิงมุมแบบ Pitch และ Roll โดย มุมของการเคลื่อนที่แบบ Pitch คือ ϕ ซึ่งเป็นมุมระหว่างแกน Y และแกน Z ดังภาพที่ 6 และมุมของการเคลื่อนที่แบบ Roll คือ θ ซึ่งเป็นมุมระหว่างแกน X และแกน Y ดังภาพที่ 7



รูปที่ 5 การเคลื่อนที่เชิงมุมแบบ Pitch ,Roll และ Yaw เทียบกับทิศการมองของมนุษย์



รูปที่ 6 มุมของการเคลื่อนที่แบบ Pitch (ϕ) มุมของการเคลื่อนที่แบบ Roll (θ) เทียบกับทิศการมองของมนุษย์

2.2.1 การคำนวณการเคลื่อนที่เชิงมุมแบบ Roll

โดยการคำนวณการเคลื่อนที่เชิงมุมแบบ Roll ซึ่งในตัวขึ้นงานนั้นจะออกแบบให้ตรวจจับการเคลื่อนที่โดยการเปลี่ยนแปลงของมุมซึ่งสมการในการคำนวณจะกำหนดให้ θ คือ มุมการเคลื่อนที่แบบ Roll A_x คือ ค่าความเร่งในแกน X A_y คือ ค่าความเร่งในแกน Y A_z คือ ค่าความเร่งในแกน Z ซึ่งการคำนวณมุมเคลื่อนที่จะคำนวณได้โดยมีสมการดังนี้

$$\theta = \arctan \left(\frac{A_y}{A_z} \right)$$

สมการที่ 1

2.2.2 การคำนวณการเคลื่อนที่เชิงมุมแบบ Pitch

โดยการคำนวณการเคลื่อนที่เชิงมุมแบบ Roll ซึ่งในตัวขึ้นงานนั้นจะออกแบบให้ตรวจจับการเคลื่อนที่โดยการเปลี่ยนแปลงของมุมซึ่งสมการในการคำนวณจะกำหนดให้ ϕ คือ มุมการเคลื่อนที่แบบ Pitch A_x คือ ค่าความเร่งในแกน X A_y คือ ค่าความเร่งในแกน Y A_z คือ ค่าความเร่งในแกน Z

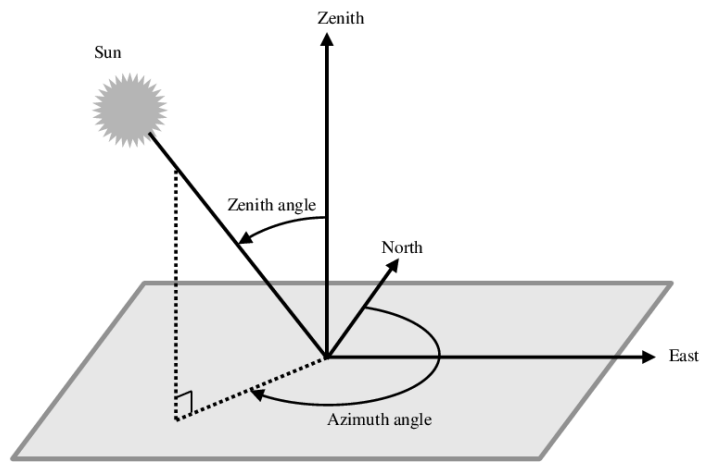
ซึ่งการคำนวณมุมเคลื่อนที่จะคำนวณได้โดยมีสมการดังนี้

$$\phi = \arctan \left(\frac{A_x}{\sqrt{A_y^2 + A_z^2}} \right)$$

สมการที่ 2

2.2.3 มุมอาชิมุต

มุมอาชิมุต¹ คือมุมที่ใช้บอกตำแหน่งของพิกัดบนท้องฟ้าซึ่งแสดงในรูปที่ 8 โดยจะวัดในแนวราบซึ่งวัดจากทิศเหนือไปยังจุดที่ต้องการวัดโดยหมุนตามเข็มนาฬิกา ซึ่งจะมีค่า 0 ถึง 360 องศา



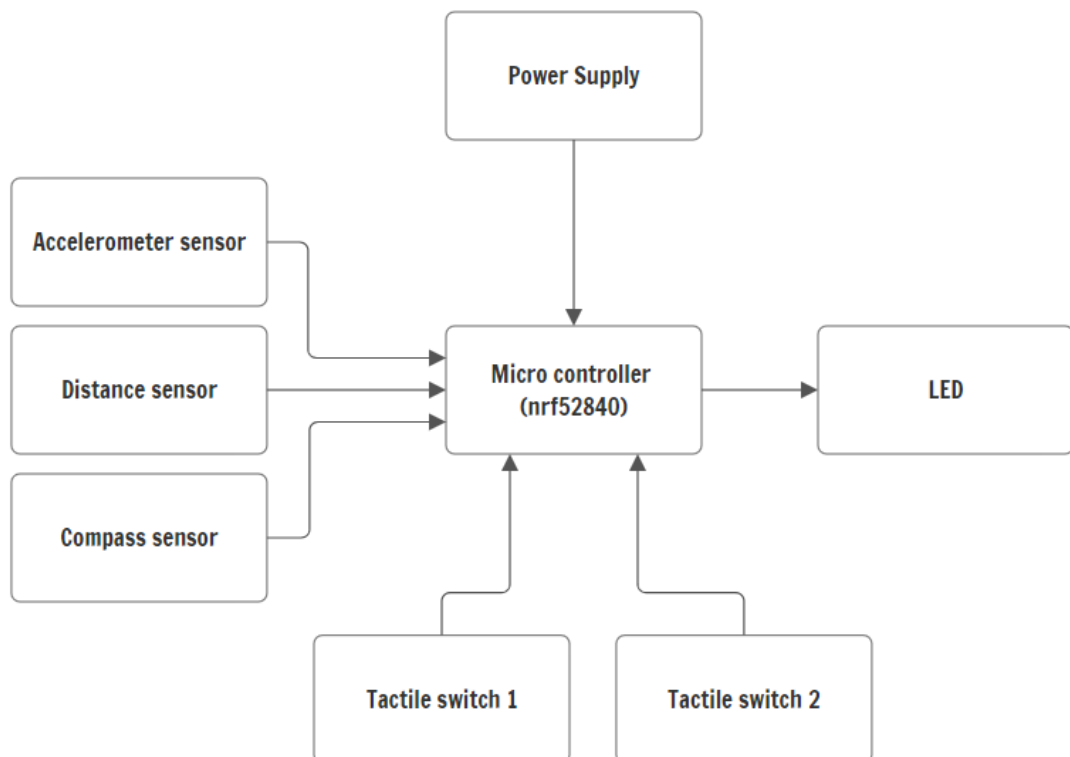
รูปที่ 7 มุมอาซิมุท

3. ผลลัพธ์ของโครงการและการอภิปรายผล

3.1 โครงสร้างและอุปกรณ์ที่ใช้

3.1.1 Block diagram

โดยอุปกรณ์จะมี Block diagram ดังนี้



รูปที่ 8 Block diagram ของอุปกรณ์

3.1.2 อุปกรณ์ที่ใช้

อุปกรณ์	จำนวน
Seeed Studio XIAO nRF52840 Sense	1
Tactile Switch	2
หลอดไฟ LED	4
Power supply (3.7 V 1200 mAh)	1
Distance Sensor VL53L1X	1
Compass Sensor HMC5883L	1
Accelerometer LSM6DS3	1

ตารางที่ 1 อุปกรณ์ที่ใช้

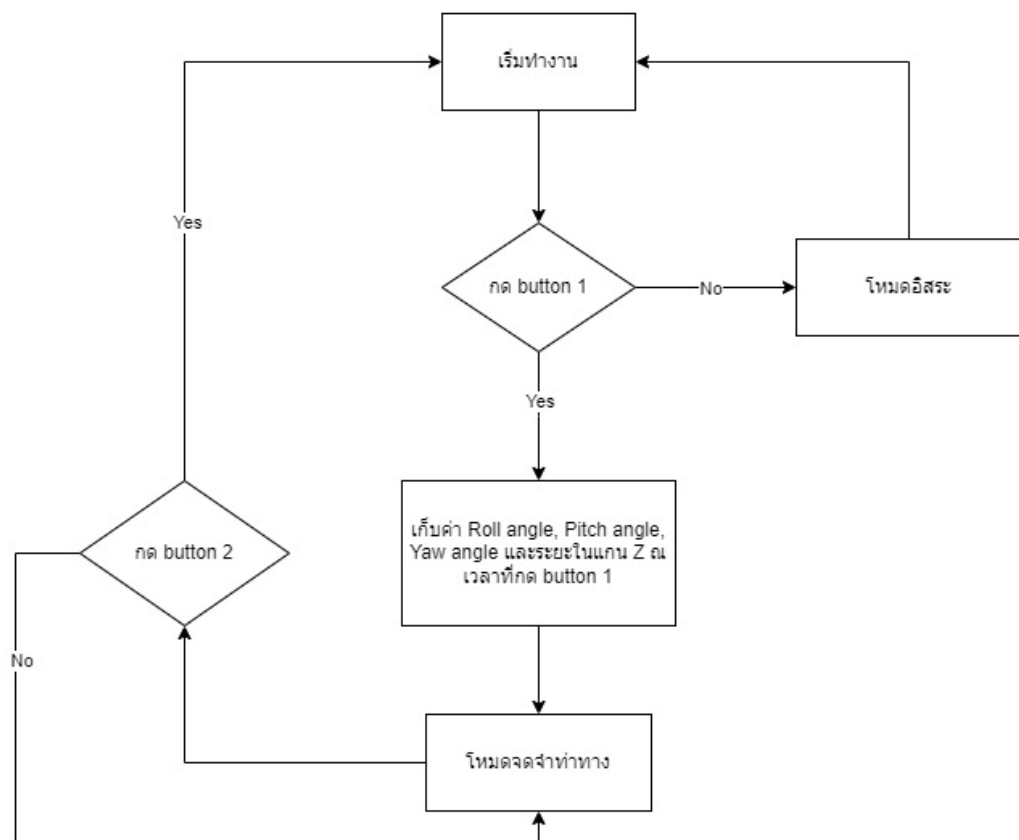
3.2 ข้อมูลเซนเซอร์

เซนเซอร์	หลักการทำงาน	ข้อมูลที่เซ็นเซอร์วัด	Specification
LSM6DS3 (Accelerometer Sensor)	หลักการทำงานโดยตรวจจับแรง ที่กระทำกับมวลในอุปกรณ์ที่เกิด การเปลี่ยนแปลงและวัด ความเร่งในแกนดังกล่าว	ข้อมูลจากเซ็นเซอร์ Accelerometer นั้นจะ เป็นค่าความเร่งในแต่ละ แกนในเชิงเส้นโดยค่าที่ แสดงผลออกมาจะเป็น ค่าที่เทียบกับค่าความเร่ง โน้มถ่วงของโลก	Measurement range $\pm 2 - 16 \text{ g}$
VL53L1X (Distance Sensor)	ส่งคลื่นอินฟราเรด ไปยังวัตถุที่ ต้องการวัดระยะหลังจากอิน ฟราเรดนั้นส่งไปยังวัตถุที่ต้องการ วัดหลังจากนั้นอินฟราเรดจะถูก สะท้อนกลับมายังที่ตัวของ เซ็นเซอร์และจะถูกตรวจจับแล้ว นำไปคำนวณระยะทางโดย คำนวณจากความเร็วของ สัญญาณอินฟราเรดกับระยะเวลา ไป-กลับของสัญญาณอินฟราเรด	ข้อมูลจากเซ็นเซอร์ Distance sensor คือ ระยะความห่างของ เซนเซอร์กับจุดอ้างอิงโดย สามารถวัดระยะทางได้ มากที่สุดคือ 360 เซนติเมตร	Distance Mode range Short mode (135 cm) Medium mode (290 cm) Short mode (360 cm) Timing budget 33 ms Ranging error $\pm 20 \text{ mm}$
HMC5883L (Compass Sensor)	วัดค่าสนามแม่เหล็กของโลกหลัง จากนั้นค่าของสนามแม่เหล็กโลก จะถูกเปลี่ยนเป็นสัญญาณไฟฟ้า ซึ่งจะถูกส่งไปประมวลผลกับไม โครโพรเซสเซอร์โดยจะคำนวณ เทียบกับค่าสนามแม่เหล็กของ ขั้วโลกเหนือ	ข้อมูลจากเซ็นเซอร์ Compass sensor คือมุม มุมอาซิมุท ¹ หรือมุมใน แนวราบ	Measuring range $\pm 1.3 - 8 \text{ Gauss}$

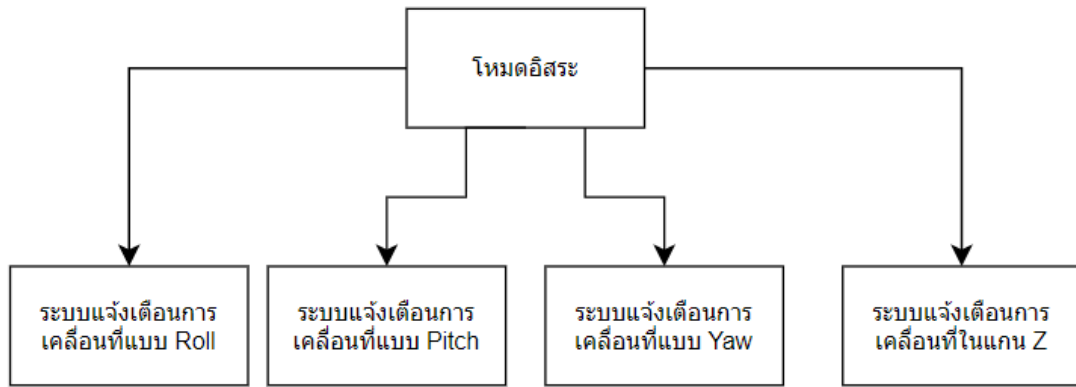
ตารางที่ 2 ข้อมูลเซนเซอร์

3.3 การทำงานของอุปกรณ์

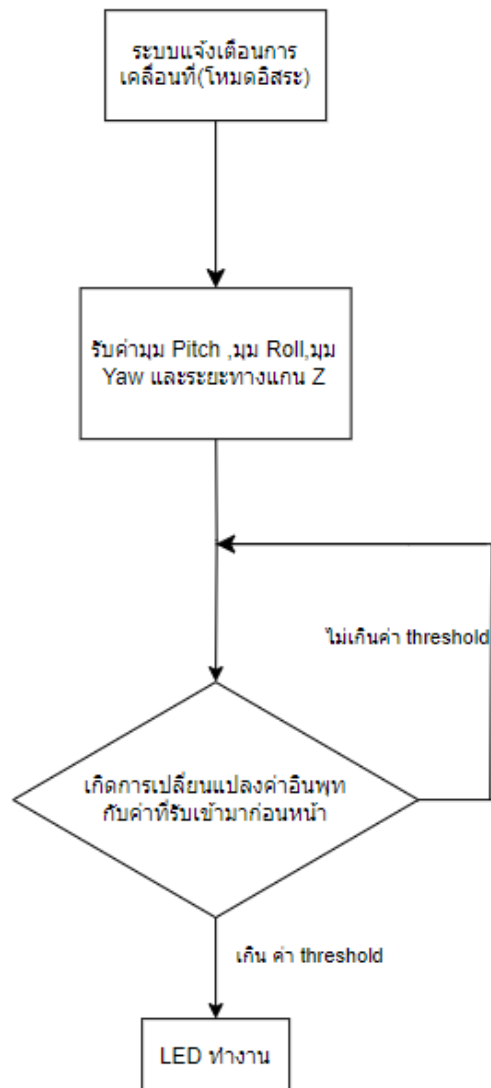
ในการทำอุปกรณ์นี้นั้นจะใช้เซ็นเซอร์ทั้งหมด 3 ตัวคือ LSM6DS3 (Accelerometer), GY-271 (Compass sensor) และ VL53L1X (Distance sensor) โดยรูปต่อไปนี้เป็น Algorithm การประมวลผลของไมโครคอนโทรลเลอร์กับเซ็นเซอร์และอุปกรณ์ต่างๆ โดย มุม Pitch ใช้ค่าความเร่งจาก LSM6DS3 คำนวณมาจากสมการที่ 2 มุม Roll ใช้ค่าความเร่งจาก LSM6DS3 คำนวณมาจากสมการที่ 1 มุม Yaw คำนวณมาจากค่ามุมอาซิมุทจาก GY-271 และค่าระยะในแกน Z วัดจาก VL53L1X



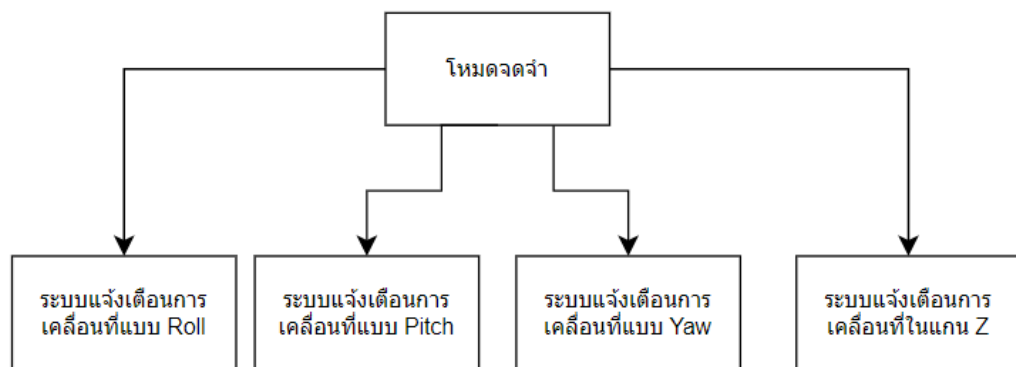
รูปที่ 9 flowchart การทำงานของอุปกรณ์



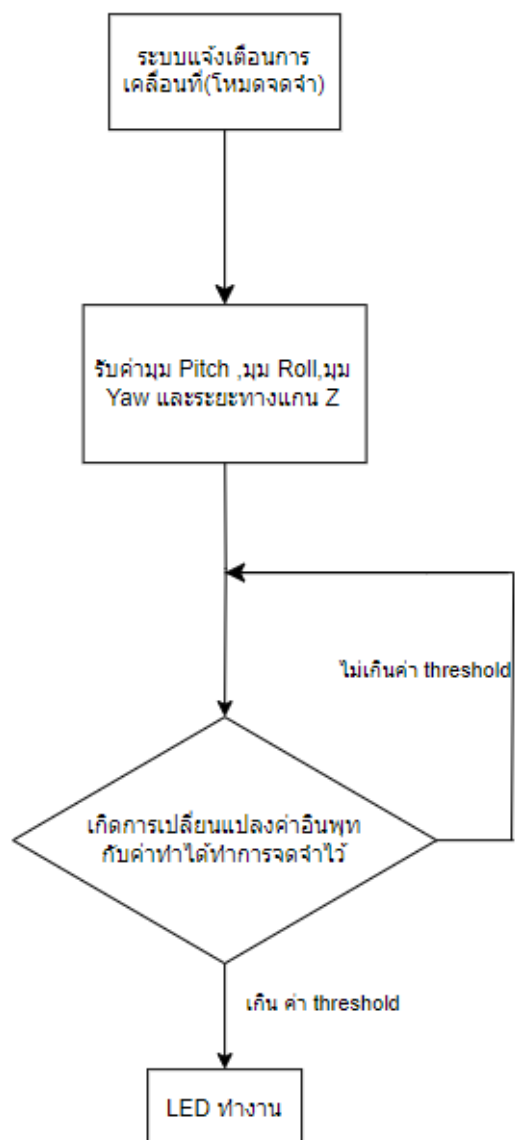
รูปที่ 10 flowchart การทำงานของอุปกรณ์



รูปที่ 11 flowchart การทำงานของอุปกรณ์

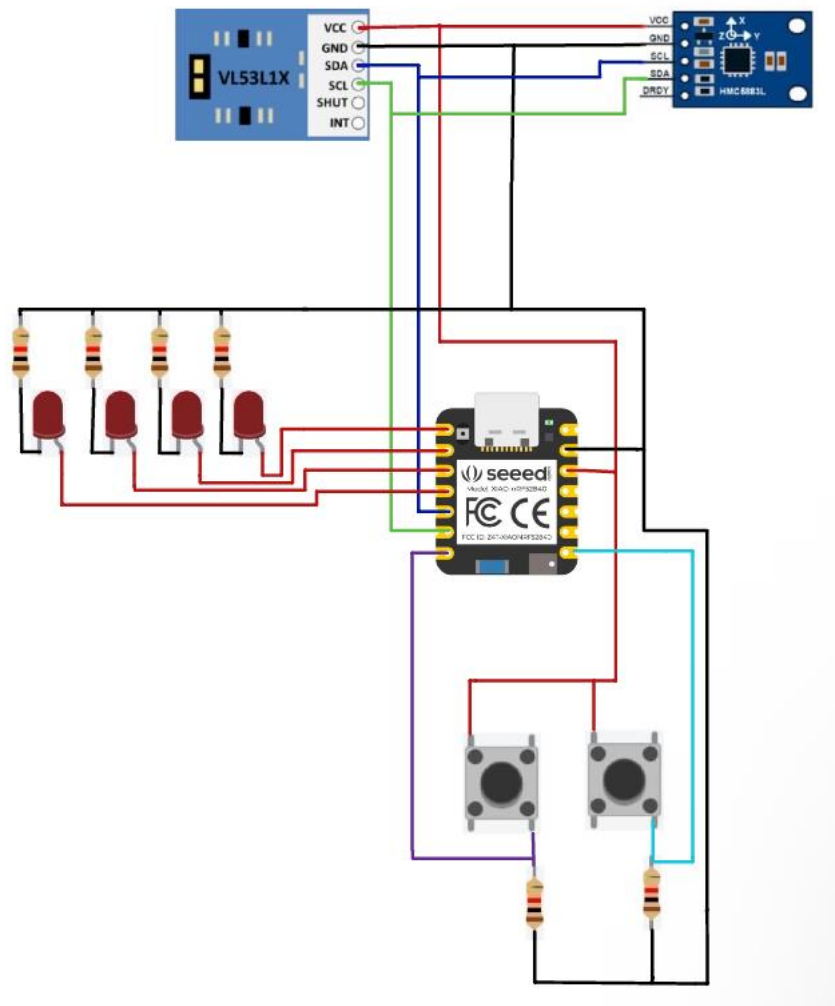


รูปที่ 12 flowchart การทำงานของอุปกรณ์

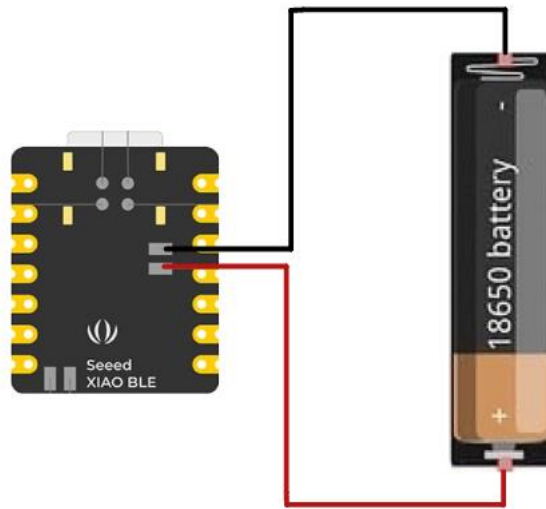


รูปที่ 13 flowchart การทำงานของอุปกรณ์

3.4 วงจรของอุปกรณ์



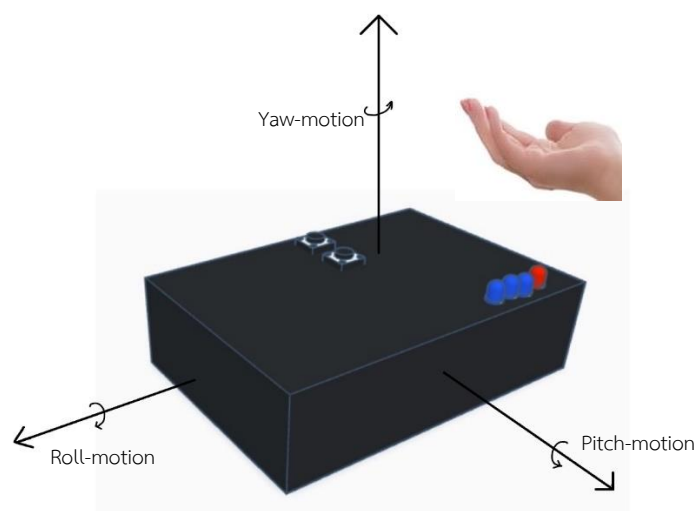
รูปที่ 14 Schematic การเชื่อมต่อไมโครคอนโทรลเลอร์และอุปกรณ์ต่างๆ



รูปที่ 15 Schematic การเชื่อมต่อไมโครคอนโทรลเลอร์และอุปกรณ์ต่างๆ

3.5 การใช้งานของอุปกรณ์

โดยอุปกรณ์จริงจะมีรูปร่างดังรูปที่ 17 และรูปที่ 18 กับรูปที่ 19 คืออุปกรณ์จริง ซึ่งเมื่อเริ่มใช้งาน ผู้ใช้งานจะต้องนำอุปกรณ์ไปติดไว้กับมือข้างที่ต้องการฝึก ซึ่งอุปกรณ์นั้นจะมีอยู่ 2 โหมดจะกล่าวในหัวข้อต่อ 3.4.1 และ 3.4.2



รูปที่ 16 แบบจำลองอุปกรณ์เทียบทิศทางการกับมือ

โดยจากรูปที่ 17 ซึ่งการตรวจจับการเคลื่อนที่ของอุปกรณ์นั้น มุม Pitch คือ หันมือและอุปกรณ์ ขึ้น-ลง มุม Roll คือ หันมือและอุปกรณ์ ไปทางซ้าย-ขวา และ มุม Yaw คือ หมุนและอุปกรณ์ รอบมือตัวเอง และ แบบคือการเคลื่อนของยกมือและอุปกรณ์ ขึ้นบน-ล่าง ซึ่งการแจ้งเตือนเมื่อเกิดการเคลื่อนที่ซึ่งมีค่า threshold ที่บ่งบอกถึงการขยับของมือนัดังนี้

1. การเคลื่อนที่แบบ Pitch โดยมี threshold คือ 3 องศา
2. การเคลื่อนที่แบบ Roll โดยมี threshold คือ 3 องศา
3. การเคลื่อนที่แบบ Yaw โดยมี threshold คือ 3 องศา
4. การเคลื่อนที่ของอุปกรณ์ ขึ้นบนล่าง โดยมี threshold คือ 40 มิลลิเมตร



รูปที่ 17 อุปกรณ์จริง



รูปที่ 18 อุปกรณ์จริง

3.5.1 การใช้งานโหมดอิสระ

ในการใช้งานโหมดอิสระนั้นจะมีขั้นตอนดังต่อไปนี้

1. เมื่อเปิดใช้งานอุปกรณ์ขึ้นมาอุปกรณ์จะอยู่ในโหมดอิสระ
2. ในโหมดอิสระนั้นเมื่อมือและอุปกรณ์นั้นอยู่นิ่งและคงตัวจะไม่เกิดการแจ้งเตือนของไฟ LED ดังรูปที่ 19



รูปที่ 19 เริ่มต้นโหมดอิสระ

3. เมื่อเกิดการขยับของมือและอุปกรณ์จะทำให้เกิดการแจ้งเตือนของไฟ LED ดังรูปที่ 20



รูปที่ 20 การทำงานของอุปกรณ์เมื่อเกิดการขยับ

4. เมื่อเวลาผ่านไปและมือกับอุปกรณ์นั้นกลับมาคงตัวแล้วไฟ LED จะดับลงดังรูปที่ 21 และจะแจ้งเตือนอีกครั้งจนกว่าจะเกิดการขยับครั้งต่อไป



รูปที่ 21 การทำงานของอุปกรณ์เมื่ออยู่นิ่งหลังจากเกิดการขยับ

3.5.2 การใช้งานโหมดจดจำ

ในการใช้งานโหมดจดจำนั้นจะมีขั้นตอนดังต่อไปนี้

1. ในการเริ่มใช้งานโหมดจดจำนั้นผู้ใช้งานจะต้องอยู่ในท่าที่ต้องการจะฝึก หลังจากนั้นทำการกดปุ่ม ดังรูปที่ 22 แล้วอุปกรณ์จะทำการจดจำท่าทาง ณ ขณะนั้นไว้



รูปที่ 22 การกดปุ่มใช้งานโหมดจดจำ

2. เมื่อเกิดการขยับของมือและอุปกรณ์ซึ่งออกจากท่าทางที่ต้องการฝึกจะทำให้เกิดการแจ้งเตือนไปยัง LED ดังรูปที่ 23



รูปที่ 23 การทำงานของอุปกรณ์เมื่อเกิดการขยับ

3. หลังจากผู้ใช้กลับมาอยู่ในท่าทางที่ต้องการฝึกที่ได้จดจำไว้ไฟ LED แฉ่งเตือนจะดับลงดังในรูปที่ 24



รูปที่ 24 การทำงานของอุปกรณ์เมื่ออยู่นิ่งในท่าที่ตั้งค่าไว้ตอนกดปุ่มจดจำหลังจากเกิดการขยับ

4. เมื่อผู้ใช้งานอยากกลับมาใช้โหมดอิสระสามารถกดปุ่มโหมดอิสระได้ดังในรูปที่ 25



รูปที่ 25 การกดปุ่มใช้งานโหมดจดจำเพื่อกลับไปโหมดอิสระ

3.6 การทดสอบการใช้งานของอุปกรณ์

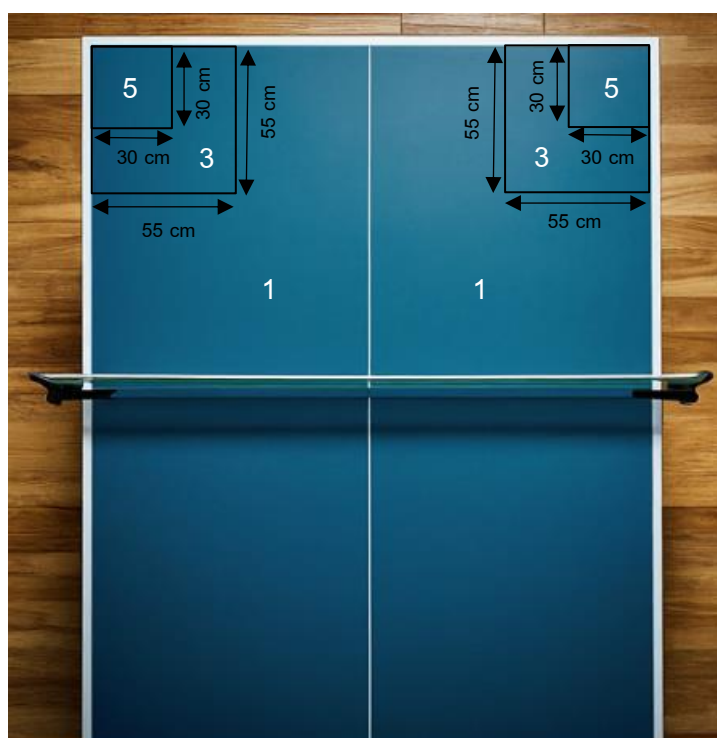
โดยการทดสอบอุปกรณ์นั้นมีวัตถุประสงค์ดังนี้

1. วัดค่าความแม่นยำของการตีลูกเทเบิลเทนนิสก่อนใช้งานอุปกรณ์
2. วัดค่าความแม่นยำของการตีลูกเทเบิลเทนนิสหลังใช้งานอุปกรณ์
3. เปรียบเทียบความแม่นยำในการตีลูกเทเบิลเทนนิสก่อนและหลังใช้งานอุปกรณ์

ซึ่งวิธีการทดสอบนั้นอ้างอิงมาจากบทความ [5] โดยมีวิธีการทดสอบอุปกรณ์ดังนี้ ซึ่งจะแบ่งการทดลองเป็นกลุ่มควบคุมและกลุ่มทดลองโดยกลุ่มผู้ทดลองมีจำนวนกลุ่มละ 1 คน

1. ให้ผู้ทดสอบของแต่ละกลุ่มทดลองตีลูกเทเบิลเทนนิสลงโต๊ะโดยทำ backhand serve โดยเสิร์ฟเป็นจำนวน 30 ครั้ง แล้วทำการบันทึกคะแนน
2. หลังจากทำการทดสอบรอบแรกในแต่ละกลุ่มแล้ว จะให้เวลาฝึกเป็น 20 นาที ซึ่งกลุ่มควบคุมจะให้การฝึกแบบธรรมดา ส่วนกลุ่มทดลองจะใช้อุปกรณ์ช่วยในการฝึก
3. หลังจากทำการฝึกเสร็จแล้วจึงทำการทดลองแบบขั้นตอนที่ 1 อีกรอบ
4. ทำการเปรียบเทียบคะแนนก่อนและหลังใช้อุปกรณ์ และกลุ่มผู้ใช้งานเพื่อดูประสิทธิภาพของอุปกรณ์

โดยการทดลองมีการคิดคะแนนดังรูปที่ 26



รูปที่ 26

โดยมีการนับคะแนนดังภาพโดยวัดจากที่บริเวณที่ลูกปิงปองตกกระทบ โดยเริ่มนับจากขอบมุมของโต๊ะปิงปองโดยบริเวณ 30 x 30 ซม. จะมีคะแนนเท่ากับ 5 คะแนน บริเวณ 55 x 55 ซม. จะมีคะแนนเป็น 3 คะแนน และบริเวณที่เหลือมีคะแนนเป็น 1 ในกรณีลูกปิงปองติดเน็ตหรือตีออกจะคิดคะแนนเป็น 0

ผลการทดลอง

ผลการทดลองกลุ่มควบคุม

รอบ	คะแนนก่อนฝึก	คะแนนหลังฝึก
1	79	82
2	78	80
3	79	83
4	77	82
5	80	84
6	81	84
7	79	83

ตารางที่ 3 คะแนนการทดลองของกลุ่มควบคุม

ผลการทดลองกลุ่มทดลอง

รอบ	คะแนนก่อนฝึก	คะแนนหลังฝึก
1	63	71
2	70	73
3	71	75
4	70	74
5	72	79
6	71	77
7	75	81

ตารางที่ 4 คะแนนการทดลองของกลุ่มทดลอง

3.7 การวิเคราะห์การทดสอบ

จากการทดลองในหัวข้อที่ 3.4 ผู้จัดทำโครงการได้รวบรวมผลการทดลองความแม่นยำในการตีลูก backhand serve โดยมีการวิเคราะห์ข้อมูลจากการทดลองได้ดังนี้

กลุ่ม	ค่าเฉลี่ยและส่วนเบี่ยงเบน มาตรฐานความแม่นยำในการ เสิร์ฟลูกแบ็คแฮนด์ก่อนการฝึก (คะแนน)	ค่าเฉลี่ยและส่วนเบี่ยงเบน มาตรฐานความแม่นยำในการ เสิร์ฟลูกแบ็คแฮนด์หลังการฝึก (คะแนน)
กลุ่มควบคุม	79±1.29	82.79± 1.39
กลุ่มทดลอง	70.27±3.63	75.18±3.49

ตารางที่ 5 ค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานความแม่นยำในการเสิร์ฟลูกแบ็คแฮนด์

จากตารางที่ 4 จะพบว่ากลุ่มควบคุมนั้นมีความแม่นยำในการเสิร์ฟลูกแบ็คแฮนด์เพิ่มขึ้น คะแนน และกลุ่มทดลองนั้นมีความแม่นยำในการเสิร์ฟลูกแบ็คแฮนด์เพิ่มขึ้น คะแนน ซึ่งจะนำข้อมูลที่ได้มาหาอัตราร้อยละการพัฒนาได้ดังตารางต่อไปนี้

กลุ่ม	ร้อยละการพัฒนา(%)
กลุ่มควบคุม	4.79
กลุ่มทดลอง	6.98

ตารางที่ 6 ร้อยละการพัฒนา

จากตารางที่ 5 จะพบว่าค่าร้อยละของการพัฒนาในกลุ่มทดลองนั้นมีค่าร้อยละ 6.98 ซึ่งมีค่ามากกว่ากลุ่มควบคุมที่มีค่าร้อยละการพัฒนาที่ 4.79

4. บทสรุป

4.1 สรุปผลการดำเนินการ

จากการดำเนินการทั้งหมดที่ผ่านมา จะได้ผลการดำเนินงานเป็นอุปกรณ์ที่สามารถตรวจจับการเคลื่อนที่ของมือผู้ใช้งานอุปกรณ์ได้ โดยการนำค่าข้อมูลต่างๆที่ได้รับจากเซ็นเซอร์ทั้ง 3 เซ็นเซอร์ โดยค่าความเร่งจาก LSMD6DS3 (Accelerometer sensor) นำมาคำนวณคิดค่ามุมการเคลื่อนที่ Pitch กับ Roll ค่ามุมอาซิมุทของ HMC5883L (Compass sensor) นำมาใช้คำนวณการเคลื่อนที่เชิงมุมแบบ Yaw และค่าระยะทางจาก VL53L1X ซึ่งวัดระยะของมือและอุปกรณ์เทียบกับพื้นนำประมวลผลโดยโปรแกรมของไมโครคอนโทรลเลอร์ ซึ่งอุปกรณ์สามารถระบุแกนหรือรูปแบบการเคลื่อนที่โดยการแจ้งเตือนด้วยหลอดไฟ LED ทั้งหมด 4 หลอดโดยแต่ละหลอดแจ้งเตือนการเคลื่อนที่เชิงมุม Pitch, Roll, Yaw และการเคลื่อนที่ในแนวแกน Z ตามลำดับ ซึ่งจะการแจ้งเตือนดังกล่าวจะช่วยให้ผู้ใช้งานนั้นสามารถรับรู้ในขณะฝึกฝนท่าทางในการฝึกกีฬาที่ใช้มือ โดยผู้ใช้งานสามารถรับรู้ได้ทันทีว่ามือของตนอยู่ในท่าทางที่ถูกต้องหรือไม่ การใช้งานอุปกรณ์นี้จะช่วยให้ผู้ใช้งานมีการพัฒนาและช่วยปรับปรุงท่าทางในการเล่นกีฬาประเภทใช้มือต่างๆได้อย่างมีประสิทธิภาพมากขึ้น เนื่องจากผู้ใช้งานนั้นสามารถรับรู้และแก้ไขความผิดพลาดได้ทันทีในช่วงเวลาที่ฝึกซ้อม ซึ่งเป็นประโยชน์และช่วยพัฒนาผู้ใช้งานได้มากกว่าการฝึกธรรมดาที่ไม่มีการช่วยเหลือจากอุปกรณ์ช่วยหรือโค้ชได้

โดยในอนาคตอุปกรณ์นี้อาจนำไปใช้เพื่อประยุกต์ในการฝึกกีฬาประเภทใช้มือได้มากขึ้น อาทิ เช่น นำไปประยุกต์ใช้กับเทคโนโลยีตรวจจับการเคลื่อนไหวในภาพเพื่อช่วยตรวจจับท่าทางได้มากขึ้นไปถึงท่าทางการยืน การทำท่าทางของแขน และความเร็วที่ใช้เพื่อให้ผู้ใช้งานนั้นมีประสิทธิภาพในการฝึกมากขึ้นกว่าเดิม หรือการเพิ่มรูปแบบการทำงานของอุปกรณ์ให้สามารถอนุญาตให้มีการเคลื่อนที่ในแกนที่ต้องการได้ ซึ่งจะช่วยให้ผู้ใช้งานสามารถฝึกการเคลื่อนที่ของมือในกีฬาประเภทมือได้ เช่น การฝึกการตีลูกปิงปองโดยใช้กลองนี้โดยมีทั้งการฝึกท่าทางและสามารถใช้กลองอุปกรณ์ฝึกการตีได้ โดยมีการควบคุมแรงและความเร็วเพื่อให้ผู้ใช้งานสามารถฝึกฝนการตีลูกเทเบิลเทนนิสได้และมีประสิทธิภาพในการเล่นกีฬามากกว่าเดิม

4.2 ปัญหา อุปสรรค และแนวทางแก้ไข

ปัญหาที่พบคือ การตรวจจับการเคลื่อนที่เชิงเส้นในแกน x และ y ซึ่งในช่วงต้นได้มีการออกแบบโดยใช้ Accelerometer sensor ซึ่งค่าที่วัดได้จาก Accelerometer นั้นมีค่าที่อ้างอิงกับความเร่งโน้มถ่วงของโลกซึ่งค่าที่อ่านได้จาก Serial monitor ใน Arduino IDE จะมีค่าเป็น 1 โดยเทียบเคียงกับค่าเท่ากับ 9.8 m/s^2 ดังนั้นในการวัดค่าความเร่งในแกน x และ y นั้นจะสามารถวัดค่าออกมาได้เป็นค่าที่น้อยมากเนื่องจากการเคลื่อนไหวของมือนั้นมีความเร่งที่น้อยทำให้ค่าที่อ่านนั้นมีความกว้างทำให้ไม่สามารถตรวจจับการเคลื่อนที่เชิงเส้นได้อย่างแม่นยำ

และการใช้เซ็นเซอร์วัดระยะทางนั้นไม่สามารถทำได้ด้วยเนื่องจากถ้าต้องใช้เซ็นเซอร์วัดระยะทาง อุปกรณ์นั้นจำเป็นต้องถูกใช้ในระบบปิดเพื่อให้สามารถมีจุดอ้างอิงเพื่อวัดระยะทางได้ แต่ทางผู้จัดทำสนใจให้สามารถใช้ได้ในทุกแบบโดยไม่จำเป็นต้องมีจุดอ้างอิง

5. กิตติกรรมประกาศ

ขอขอบคุณอาจารย์ที่ปรึกษา อ.ดร.ศวิษฐ์ ณ สงขลา ตั้งแต่การพิจารณานิสิตเพื่อเป็นอาจารย์ที่ปรึกษาของวิชาโครงงานวิศวกรรมไฟฟ้า ตลอดจนการที่คอยแนะแนววิธีการดำเนินงาน การของโครงงาน การเขียนโปรแกรม การเขียนรายงาน สอนความรู้ทางอิเล็กทรอนิกส์ แนะนำอุปกรณ์สำหรับโครงงาน และแนะนำการประยุกต์ใช้ ตลอดจนการสละเวลาเพื่อช่วยเหลือหาแนวทางแก้ไขที่นิสิตได้พบระหว่างการดำเนินงาน ขอแสดงความขอบคุณเป็นอย่างยิ่งเนื่องจากถ้าไม่ได้รับการแนะนำและช่วยเหลือจากท่านอาจารย์โครงงานนี้จะไม่สามารถบรรลุผลได้เลย นอกขอขอบคุณเพื่อนนิสิตที่เข้าร่วมเป็นผู้ทดลองของอุปกรณ์นี้

6. เอกสารอ้างอิง

- [1] Jeki Haryanto and Frizki Amra, The relationship of concentration and eye-hand coordination with accuracy of backhand backspin serve in table tennis, 2020
- [2] Masahiro Ikenaga, Nobue Okuma, Hiroki Nishiyama, Shinichiro Chiba, Katsutoshi Nishino, Go Omori and Hiroyuki Nunome, Influence of Ball Impact Location on Racket Kinematics, Forearm Muscle Activation and Shot Accuracy during the Fore Groundstrokes in Tennis, 2020
- [3] P. C. Thotawaththa and A. W. S. Chandana, Sagittal Plane Release Parameters of the Javelin Throwing: A Review, 2021
- [4] Ben J. Pullen, Jon L. Oliver, Rhodri S. Lloyd and Camilla J. Knight, Assessing Athletic Motor Skill Competencies in Youths: A Narrative Review of Movement Competency Screens, 2021
- [4] Miss Pornpun Rakprapakarn, Effects of augmented quiet eye training on shot accuracy of forehand topspin drive in table tennis, 2017

7. ภาคผนวก

7.1 ภาคผนวก ก.

Seeed Studio XIAO nRF52840 Datasheet: https://files.seeedstudio.com/wiki/XIAO-BLE/Nano_BLE_MCU-nRF52840_PS_v1.1.pdf

Infrared Distance Sensor Datasheet: <https://www.alldatasheet.com/datasheet-pdf/pdf/1131878/STMICROELECTRONICS/VL53L1X.html>

GY-271 Compass Sensor Datasheet :<https://handsontec.com/dataspecs/sensor/GY-271%20HMC5883L.pdf>

รูปที่ 1 https://wiki.seeedstudio.com/XIAO_BLE/

รูปที่ 2 <https://www.st.com/en/imaging-and-photonics-solutions/vl53l1x.html>

รูปที่ 3 <https://www.flyrobo.in/gy-271-qmc5883l-3-axis-electronic-compass-module-magnetic-field-sensor>

รูปที่ 4 https://personal.utdallas.edu/~tres/integ/sen5/display9_23.html

รูปที่ 5 https://www.researchgate.net/figure/Pitch-Roll-and-Yaw-Against-the-XYZ-Axis-own-figure_fig1_336969692 และ <https://sketchfab.com/3d-models/human-head-f46d952886ae4a8c8851341b810bba43>

รูปที่ 6 <https://sketchfab.com/3d-models/human-head-f46d952886ae4a8c8851341b810bba43>

รูปที่ 7 https://www.researchgate.net/figure/Representation-of-azimuth-and-zenith-angles_fig1_299413323

รูปที่ 26 <https://gemini.google.com/app/f762cecc42c07b45>

7.2 ภาคผนวก ข.



รูปที่ 27 การใช้งานของอุปกรณ์



รูปที่ 28 การใช้งานของอุปกรณ์



รูปที่ 29 การใช้งานของอุปกรณ์

```

1  #include <Wire.h>
2  #include <QMC5883LCompass.h>
3  #include "LSM6DS3.h"
4  #include <VL53L1X.h>
5
6  LSM6DS3 myIMU(I2C_MODE, 0x6A);
7  QMC5883LCompass compass;
8  VL53L1X sensor;
9
10 // Define the states of the state machine
11 enum State {
12     STATE_IDLE,
13     STATE_BUTTON_PRESSED,
14 };
15
16 // Define the initial state
17 State currentState = STATE_IDLE;
18
19 // Define pin numbers
20 const int buttonPin1 = 7 ;
21 const int buttonPin2 = 8 ;
22
23 // Define LED
24 const int LED_PIN0 = 0;
25 const int LED_PIN1 = 1;
26 const int LED_PIN2 = 2;
27 const int LED_PIN3 = 3;
28
29
30 // Variables to store button state
31 bool buttonState1 = false;
32 bool lastButtonState1 = false;
33 bool buttonState2 = false;

```

รูปที่ 30 โค้ดคำสั่งของอุปกรณ์

```

34 bool lastButtonState2 = false;
35
36 // Variable for Pitch and Roll Calculate
37 float X = 0;
38 float Y = 0;
39 float Z = 0;
40 float Roll = 0;
41 float Pitch = 0;
42 float theta = 0;
43 float different_Roll = 0;
44 float different_Pitch = 0;
45 float different_Roll_remem = 0;
46 float different_Pitch_remem = 0;
47 float Roll_remem = 0;
48 float Pitch_remem = 0;
49 float previous_Roll;
50 float previous_Pitch;
51 // Variable for Yaw Calculate
52 int Yaw;
53 int Yaw_remem = 0;
54 int previous_Yaw;
55 int different_Yaw = 0;
56 int different_Yaw_remem = 0;
57
58 // Variable for Z axis
59 float different_range = 0;
60 float previous_Z_axis_range;
61 float Z_axis_range = 0;
62 float different_range_remem = 0;
63 float Z_axis_range_remem = 0;
64
65 void setup() {
66     pinMode(buttonPin1, INPUT_PULLUP); // Set button pin as input with internal pull-up resistor

```

รูปที่ 31 โค้ดคำสั่งของอุปกรณ์

```

67  pinMode(buttonPin2, INPUT_PULLUP);
68  Serial.begin(9600); // Initialize serial communication
69  compass.init();
70
71  Wire.begin();
72  Wire.setClock(400000); // Use 400 kHz I2C
73
74  while (!Serial);
75  | //Call .begin() to configure the IMUs
76  if (myIMU.begin() != 0) {
77  |   Serial.println("Device error");
78  } else {
79  |   Serial.println("Device OK!");
80  }
81  pinMode(LED_PIN0, OUTPUT);
82  pinMode(LED_PIN1, OUTPUT);
83  pinMode(LED_PIN2, OUTPUT);
84  pinMode(LED_PIN3, OUTPUT);
85  }
86
87  void loop() {
88  | // Read the button state
89  | buttonState1 = digitalRead(buttonPin1);
90  | buttonState2 = digitalRead(buttonPin2);
91  | //Roll Pitch data
92  | X = myIMU.readFloatAccelX();
93  | Y = myIMU.readFloatAccelY();
94  | Z = myIMU.readFloatAccelZ();
95  | Pitch = atan2(-X, sqrt(Y*Y + Z*Z)) * 180/3.14;
96  | Roll = atan2(Y, Z) * 180/3.14;
97
98  | //compass data
99  | compass.read();

```

รูปที่ 32 โค้ดคำสั่งของอุปกรณ์

```

100 Yaw = compass.getAzimuth();
101
102 //distance data
103 sensor.read();
104 Z_axis_range = sensor.ranging_data.range_mm;
105
106 switch(currentState) {
107     case STATE_IDLE:
108         if (buttonState1 == HIGH && lastButtonState1 == LOW) {
109             currentState = STATE_BUTTON_PRESSED;
110             Yaw_remem = Yaw;
111             Roll_remem = Roll;
112             Pitch_remem = Pitch;
113             Z_axis_range_remem = Z_axis_range;
114         }
115
116         different_Yaw = abs(Yaw - previous_Yaw);
117         different_Roll = abs(Roll - previous_Roll);
118         different_Pitch = abs(Pitch - previous_Pitch);
119         different_range = abs(Z_axis_range - previous_Z_axis_range);
120
121         if (different_Pitch > 3) {
122             digitalWrite(LED_PIN0, HIGH);
123         } else {
124             digitalWrite(LED_PIN0, LOW);
125         }
126
127         if (different_Roll > 3) {
128             digitalWrite(LED_PIN1, HIGH);
129         } else {
130             digitalWrite(LED_PIN1, LOW);
131         }
132

```

รูปที่ 33 โค้ดคำสั่งของอุปกรณ์

```

133     if (different_Yaw > 3) {
134         digitalWrite(LED_PIN2, HIGH);
135     } else {
136         digitalWrite(LED_PIN2, LOW);
137     }
138
139     if (different_range > 40) {
140         digitalWrite(LED_PIN2, HIGH);
141     } else {
142         digitalWrite(LED_PIN2, LOW);
143     }
144
145     Serial.println("Free");
146     Serial.println(different_Pitch);
147     Serial.println(different_Roll);
148     Serial.println(different_Yaw);
149     Serial.println(different_range);
150     previous_Yaw = Yaw;
151     previous_Roll = Roll;
152     previous_Pitch = Pitch;
153     previous_Z_axis_range = Z_axis_range;
154     delay(500);
155     break;
156
157 case STATE_BUTTON_PRESSED:
158     if (buttonState2 == LOW && lastButtonState2 == HIGH) {
159         currentState = STATE_IDLE;
160     }
161
162     different_Yaw_remem = abs(Yaw - Yaw_remem);
163     different_Roll_remem = abs(Roll - Roll_remem);
164     different_Pitch_remem = abs(Pitch - Pitch_remem);
165     different range remem = abs(Z axis range - Z axis range remem);

```

รูปที่ 34 โค้ดคำสั่งของอุปกรณ์

```

166
167     if (abs(different_Pitch_remem) > 3) {
168         digitalWrite(LED_PIN0, HIGH);
169     } else {
170         digitalWrite(LED_PIN0, LOW);
171     }
172
173     if (abs(different_Roll_remem) > 3) {
174         digitalWrite(LED_PIN1, HIGH);
175     } else {
176         digitalWrite(LED_PIN1, LOW);
177     }
178
179     if (abs(different_Yaw_remem) > 3) {
180         digitalWrite(LED_PIN2, HIGH);
181     } else {
182         digitalWrite(LED_PIN2, LOW);
183     }
184
185     if (abs(different_range_remem) > 40) {
186         digitalWrite(LED_PIN3, HIGH);
187     } else {
188         digitalWrite(LED_PIN3, LOW);
189     }
190
191     Serial.println("Remem");
192     Serial.println(different_Pitch_remem);
193     Serial.println(different_Roll_remem);
194     Serial.println(different_Yaw_remem);
195     Serial.println(different_range_remem);
196     break;
197 }
198

```

รูปที่ 35 โค้ดคำสั่งของอุปกรณ์

```

199 // Store the current button state for comparison in the next loop iteration
200 lastButtonState1 = buttonState1;
201 lastButtonState2 = buttonState2;
202 }
203

```

รูปที่ 36 โค้ดคำสั่งของอุปกรณ์