

เซนเซอร์วิเคราะห์ระยะทาง
GYRO SENSOR PROCESSING ACCELERATION



ชาคร พรหมสาขา ณ สกลนคร
ณัฐนันท์ ทรัพย์อรุณ
สุภัทรา พรหมสุรักษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมระบบควบคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2557

เซนเซอร์วิเคราะห์ระยะทาง

GYRO SENSOR PROCESSING ACCELERATION



ชาคร พรหมสาขา ณ สกลนคร
ณัฐนันท์ ทรัพย์อรุณ
สุภัทรา พรหมสุรักษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ปีการศึกษา 2557
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแบบลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GYRO SENSOR PROCESSING ACCELERATION



THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN CONTROL ENGINEERING
FACULTY OF ENGINEERING

เอกสารนี้เป็นเอกสารของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกหรือเผยแพร่
ACADEMIC YEAR 2014

ปริญญาานิพนธ์ปีการศึกษา 2557

ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เซนเซอร์วิเคราะห์ระยะทาง

GYRO SENSOR PROCESSING ACCELERATION

ผู้จัดทำ นายชาคร พรหมสาขา ณ สกลนคร 54010312

นายณัฐนันท์ ทรัพย์อริญ 54010410

นางสาวสุภัทรา พรหมสุรักษ์ 54011418


.....อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์)


.....อาจารย์ที่ปรึกษา
(รองศาสตราจารย์ ดร. เกียรติศักดิ์ คมวัชระ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์วัดระยะทาง

โดย

นายชาคร พรหมสาขา ณ สกลนคร 54010312

นายณัฐนันท์ ทรัพย์อรัญ 54010410

นางสาวสุภัทรา พรหมสุรักษ์ 54011418

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์

รองศาสตราจารย์ ดร. เกียรติศักดิ์ คมวัชระ

ปีการศึกษา 2557

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอทฤษฎีและการออกแบบอุปกรณ์ทดลองระบุตำแหน่ง โดยโครงสร้างของอุปกรณ์ระบุตำแหน่งในแนวแกน 3 มิติประกอบด้วย ส่วนชุดโครงสร้าง อุปกรณ์วัดระยะทาง เซนเซอร์ และเซอร์โวมอเตอร์ จุดมุ่งหมายของโครงการนี้คือ ตรวจสอบการเคลื่อนที่ของวัตถุในแนวแกน 3 มิติ และสามารถตรวจสอบสถานะของวัตถุต่างๆ ว่ามีสถานะในปัจจุบันเป็นอย่างไร อีกทั้งยังสามารถคำนวณระยะทางของวัตถุได้ว่าเคลื่อนที่ไปเท่าไร

ขั้นตอนดำเนินการเริ่มจากการศึกษาตัวเซนเซอร์วัดระยะทาง คุณสมบัติต่างๆ และชุดอุปกรณ์อิเล็กทรอนิกส์ที่จำเป็นทั้งหมด ประกอบด้วย ไมโครคอนโทรลเลอร์ ใจโรเซนเซอร์ และ เซอร์โวมอเตอร์ เพื่อนำอุปกรณ์มาประกอบเป็นชุดทดลองหาความเร่งของวัตถุ เพื่อนำมาพิจารณาค่าระยะทางการเคลื่อนที่ แล้วจึงเขียนโปรแกรมคอมพิวเตอร์เพื่อให้ตัวเซนเซอร์แสดงค่าที่ต้องการ และ เซอร์โวมอเตอร์เป็นไปตามการทดลอง จากการทดลองพบว่าการจำลองอุปกรณ์วัดระยะทางสามารถแสดงค่าออกมาที่มีความผิดพลาดอยู่ในระดับที่ยอมรับได้

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GYRO SENSOR PROCESSING ACCELERATION

By

Chakorn Promsaka Na Sakolnakorn 54010312

Nattant Sap-Aran 54010410

Supattra Promsurak 54011418

Advisors

Asst.Prof. Sumit Panaaudomsap

Assoc.Prof.Dr. Kiatisuk Komwatchara

Academic Year 2014

ABSTRACT

This thesis presents theory and design of Gyro Sensor processing acceleration. Sensor composes of hardware, software, sensor and servo motor. The purpose of this thesis to monitor the movement of objects in the 3D axis, and it can check the status of object in the current environment moreover. it can also calculate the distance of the objects.

The procedures compose learning the gyro-85 and assembling the necessary electronic circuits, including microcontroller board (Arduino nano v.4.0), gyro sensor (gyro-85) and servo motor (MG90s) in order to attributed to determine the acceleration of the object for considering the range of motion, and then write a computer program to display the desired value, and servo motor work as an experiment. It was found that simulation of the distance measurement can show up with the error in the acceptable level.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญาานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงไปได้ด้วยดีเพราะได้รับความช่วยเหลือเป็นอย่างดีจาก ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์ ที่ได้กรุณาให้คำปรึกษาและคำแนะนำที่ดีมาโดยตลอดตั้งแต่ต้น คอยติดตามความคืบหน้าของผลงาน รวมทั้งเอื้อเฟื้ออุปกรณที่จำเป็นและความช่วยเหลืออื่นๆ ที่เป็นประโยชน์ต่อโครงการ ผู้จัดทำรู้สึกซาบซึ้งและขอกราบขอบพระคุณอย่างสูง

ขอบคุณเพื่อนๆ ในกลุ่มงานทุกคนที่ให้ความร่วมมือในการทำงาน ช่วยเหลือซึ่งกันและกันสนับสนุนอุปกรณที่ขาดเหลือกระตุ้นเตือนรวมทั้งคอยถามไถ่ความคืบหน้าของโครงการอยู่เสมอ

สุดท้ายนี้ผู้จัดทำขอกราบขอบพระคุณบิดา มารดา และครอบครัวที่คอยเป็นกำลังใจที่ดีตลอดมา รวมถึงการสนับสนุนในเรื่องของงบประมาณที่ขาดเหลือ ตลอดจนเป็นแรงบันดาลใจที่ดีที่สุดที่ทำให้โครงการนี้สำเร็จสมบูรณ์ลงได้

ผู้จัดทำ

นายชาคร พรหมสาขา ณ สกลนคร

นายณัฐนันท์ ทรัพย์อรัญ

นางสาวสุภัทรา พรหมสุรกิจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของการศึกษาโครงการ	1
1.4 วิธีการดำเนินงาน	1
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
1.6 รายละเอียดปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง	3
2.1 หลักการทำงานของเครื่องมือ	3
2.1.1 Accelerometer	3
2.1.2 Gyroscope	6
2.1.3 Magnetometer	6
2.1.4 Kalman Filter	7
2.2 การส่งข้อมูล	10
2.2.1 การเชื่อมต่ออุปกรณ์แบบ I ² C	10
2.3 การอินเตอร์รัพท์	12
บทที่ 3 การออกแบบและการเก็บค่าของอุปกรณ์	16
3.1 Sensor Gyro-85	16
3.2 ไมโครคอนโทรลเลอร์	19
3.3 Servo Motor	21
3.4 การประมวลผลข้อมูล	25
บทที่ 4 ผลการทดลอง	28

เอกสารนี้เป็นเอกสารที่ 4.1 ที่การทดลองหาค่าระยะทางการเคลื่อนที่ตามแนวแกน Y อนุภาคให้นำไปใช้ประโยชน์ด้านการศึกษา
ไม่ว่ากรณีใดๆทั้งสิ้น 4.1.1 ระยะทาง 50 เซนติเมตร และต้องอ้างอิงถึงเข้าของเอกสารทุกครั้งที่มีกา 29 ไปใช้

สารบัญ(ต่อ)

	หน้า
4.1.1.1 การทดลองชุดที่ 1	29
4.1.1.2 การทดลองชุดที่ 2	31
4.1.2 ระยะทาง 80 เซนติเมตร	34
4.1.2.1 การทดลองชุดที่ 1	34
4.1.2.2 การทดลองชุดที่ 2	36
4.2 การทดลองหาค่าระยะการเคลื่อนที่ตามแนวแกน X	39
4.2.1 ระยะทาง 50 เซนติเมตร	39
4.2.1.1 การทดลองชุดที่ 1	39
4.2.1.2 การทดลองชุดที่ 2	40
4.2.2 ระยะทาง 80 เซนติเมตร	41
4.2.2.1 การทดลองที่ 1	42
4.2.2.2 การทดลองที่ 2	43
4.3 การทดลองหาค่า ROLL PITCH	45
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	47
5.1 บทสรุปการดำเนินงาน	47
5.2 ปัญหาในการดำเนินงาน	48
5.3 แนวทางในการพัฒนาโครงการ	49
เอกสารอ้างอิง	50
ภาคผนวก	51
ภาคผนวก ก โปรแกรมอ่านค่า Sensor	52
ภาคผนวก ข เอกสารคู่มืออุปกรณ์อิเล็กทรอนิกส์	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า	
2.1	ตัววัดความเร่งเมื่อปราศจากแรงกระทำจากภายนอก	4
2.2	ตัววัดความเร่งเมื่อเคลื่อนที่ไปทางซ้ายด้วยความเร่ง 1g	4
2.3	ผลจากแรงโน้มถ่วงโลกที่มีต่อตัววัดความเร่ง	5
2.4	ตัววัดความเร่งเมื่อหมุนไป 45 องศา ทิศตามเข็มนาฬิกา	5
2.5	มุม Roll, Pitch และ Yaw	6
2.6	จำลองแกนการหมุนของเครื่องบิน	8
2.7	จำลองแกนการหมุนแบบแยกทั้ง 3 แกน	9
2.8	การสื่อสารแบบ SPI	11
2.9	การสื่อสารแบบ I ² C	12
3.1	ตัววัดความเร่ง ADXL345	16
3.2	วัดสนามแม่เหล็ก MC5883L	17
3.3	วัดค่าความเร็วเชิงมุม ITG3205	17
3.4	Arduino Nano V.4	19
3.5	กราฟความเร็วกับเวลา	25
3.6	กราฟระหว่างความเร็วและเวลา สำหรับความเร่งคงตัว	26
4.1	อุปกรณ์ที่ใช้ในการทดลอง	28
4.2	ความเร่งในแนวแกน Y	29
4.3	ความเร็วในแนวแกน Y ที่เกิดจากการอินทิเกรตความเร่ง	30
4.4	ระยะการเคลื่อนที่ในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร็ว	30
4.5	ค่าความเร่งในแนวแกน Xg Yg Zg (m/s ²)	30
4.6	ความเร่งในแนวแกน Y	31
4.7	ความเร็วในแนวแกน Y ที่เกิดจากการอินทิเกรตความเร่ง	31
4.8	ระยะการเคลื่อนที่ในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร็ว	32
4.9	ค่าความเร่งในแนวแกน Xg Yg Zg (m/s ²)	32
4.10	ความเร่งในแนวแกน Y	34
4.11	ความเร็วในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร่ง	34
4.12	ระยะการเคลื่อนที่ในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร็ว	35
4.13	ค่าความเร่งในแนวแกน Xg Yg Zg (m/s ²)	35
4.14	ค่าความเร่งในแนวแกน Y	36
4.15	ค่าความเร็วในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร่ง	36

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นกรณีที่มีเหตุผลเบ็ดเตล็ด และต้องขออนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.16 ค่าระยะการเคลื่อนที่ในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร็ว	37
4.17 ค่าความเร่งในแนวแกน Xg Yg Zg (m/s^2)	37
4.18 ความเร่งในแนวแกน X	39
4.19 ความเร็วในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร่ง	39
4.20 ระยะการเคลื่อนที่ในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร็ว	39
4.21 ความเร่งในแนวแกนการเคลื่อนที่ด้าน X (Xg Yg Zg)	40
4.22 ความเร่งในแนวแกน X	40
4.23 ความเร็วในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร่ง	40
4.24 ระยะการเคลื่อนที่ในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร็ว	41
4.25 ความเร่งในแนวแกนการเคลื่อนที่ด้าน X (Xg Yg Zg)	41
4.26 ค่าความเร่งในแนวแกน Xg	42
4.27 ค่าความเร็วในแนวแกน Xg ที่เกิดจากการอินทิเกรตค่าความเร่ง	42
4.28 ระยะการเคลื่อนที่ในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร็ว	43
4.29 ความเร่งในแนวแกนการเคลื่อนที่ด้าน X (Xg Yg Zg)	43
4.30 ค่าความเร่งในแนวแกน Xg	43
4.31 ค่าความเร็วในแนวแกน Xg ที่เกิดจากการอินทิเกรตค่าความเร่ง	44
4.32 ระยะการเคลื่อนที่ในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร็ว	44
4.33 ความเร่งในแนวแกนการเคลื่อนที่ด้าน X (Xg Yg Zg)	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
4.1 ค่าระยะการเคลื่อนเทียบค่าจากค่าจริงกับค่าจากการทดลอง	33
4.2 ค่าระยะการเคลื่อนเทียบค่าจากค่าจริงกับค่าจากการทดลอง	38
4.3 ค่าระยะการเคลื่อนเทียบค่าจากค่าจริงกับค่าจากการทดลอง	42
4.4 ค่าระยะการเคลื่อนเทียบค่าจากค่าจริงกับค่าจากการทดลอง	45
4.5 ผลการทดลองการเก็บค่ามุม PITCH	45
4.6 ผลการทดลองการเก็บค่ามุม ROLL	46



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

ในปัจจุบันระบบการทรงตัวมีความจำเป็นมากในการควบคุมหุ่นยนต์ การอ้างอิงตำแหน่ง และสภาพการทรงตัวของรถ จึงได้มีการคิดค้นอุปกรณ์ตรวจสอบการเคลื่อนที่ของวัตถุในแนวแกน 3 มิติ ขึ้นมา เพื่อศึกษาการเคลื่อนที่ของวัตถุในแนวแกน 3 มิติ และสามารถตรวจสอบสถานะของวัตถุต่างๆ ว่ามีสถานะในปัจจุบันเป็นอย่างไร อีกทั้งยังสามารถคำนวณระยะทางของวัตถุได้ว่าเคลื่อนที่ไปเท่าไร โดยการศึกษาจากการเขียนโปรแกรมและนำค่าที่ได้มาพิจารณาการเคลื่อนที่ของวัตถุ

ระบบระบุตำแหน่งทิศทางการเคลื่อนที่นับเป็นส่วนสำคัญในงานที่เกี่ยวข้องกับวัตถุ ที่มีการเคลื่อนที่ในงานด้านต่างๆ อาทิเช่น เครื่องมือทางการแพทย์ เทคโนโลยีการกีฬา รวมไปถึงหุ่นยนต์ โดยมีการศึกษา และพัฒนาวิธีการระบุตำแหน่ง และทิศทางขึ้นมากมายแตกต่างกันไปตามลักษณะของงานที่นำไปใช้

1.2 วัตถุประสงค์ของโครงการ

1. ศึกษาค่าความเร่ง ความเร็ว และระยะทาง แล้วปรับให้เกิดความเหมาะสม
2. ศึกษาการเคลื่อนที่เป็นแนวเส้นตรงของวัตถุในแนวแกน 3 มิติ
3. ศึกษาการเขียนโปรแกรมเพื่อคำนวณค่าเพื่อการทรงตัวของวัตถุ
4. ศึกษาเซนเซอร์ที่ใช้ในการตรวจสอบการเคลื่อนที่ของวัตถุในแนวแกน 3 มิติ

1.3 ขอบเขตของการศึกษาโครงการ

1. ศึกษาการเขียนโปรแกรมเพื่อใช้ในการตรวจสอบการเคลื่อนที่ของวัตถุในแนวแกน 3 มิติ
2. ออกแบบและทดลองอุปกรณ์ตรวจสอบการเคลื่อนที่ของวัตถุในแนวแกน 3 มิติ เพื่อตรวจจับความเร็ว ความเร่ง และระยะทาง
3. ศึกษาและออกแบบอุปกรณ์ให้มีค่าที่มีค่าความผิดพลาดที่น้อยที่สุด

1.4 วิธีดำเนินการโครงการ

1. ศึกษาทฤษฎีที่เกี่ยวข้อง

1.1 ศึกษาหลักการทำงาน และการใช้งานตัววัดความเร่งและใจโรเซนเซอร์ รวมถึงการรับส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์กับเซนเซอร์ และไมโครคอนโทรลเลอร์กับคอมพิวเตอร์

1.2 ศึกษาการเขียนโปรแกรมเพื่อวิเคราะห์ค่าความเร่ง

1.3 ศึกษาการสมการการประมวลผล และแสดงผลข้อมูล

1.4 ศึกษาวิธีการลดทอนความผิดพลาด และการปรับปรุงสัญญาณจากเซนเซอร์เพื่อให้มี

เสถียรภาพ ความถูกต้องแม่นยำมากยิ่งขึ้น

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การออกแบบและสร้าง

2.1 การออกแบบการทดลองและทดสอบวิเคราะห์ค่า

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. นำความรู้ในเรื่องของไมโครคอนโทรลเลอร์มาประยุกต์ใช้งาน ในการออกแบบอุปกรณ์ตรวจสอบการเคลื่อนที่ของวัตถุในแนวแกน 3 มิติ
2. นำความรู้ในการเขียนโปรแกรมมาประยุกต์ใช้งาน ในการออกแบบอุปกรณ์ตรวจสอบการเคลื่อนที่ของวัตถุในแนวแกน 3 มิติ
3. ได้ศึกษาเรื่องการอ้างอิงตำแหน่งต่างๆ ของวัตถุในสถานะปัจจุบัน (Inertial Measurement Unit) เพื่อ นำไปใช้หรือพัฒนาไปยังอุปกรณ์ต่างๆ
4. นำความรู้ในเรื่องของเซนเซอร์มาประยุกต์ใช้งาน ในการออกแบบอุปกรณ์ตรวจสอบการเคลื่อนที่ของวัตถุในแนวแกน 3 มิติ

1.6 รายละเอียดปริญญานิพนธ์

ส่วนการทดลองจะแบ่งเป็น 2 ส่วน ได้แก่ ส่วนการออกแบบการทดลองให้มีค่าความเร็วคงที่เพื่อนำมาค่าระยะทางการเคลื่อนที่ โดยแสดงผลในรูปของกราฟความสัมพันธ์ระหว่างค่าความเร่ง ความเร็ว และระยะทางกับเวลา อีกส่วนก็คือ ส่วนของโปรแกรมในการหาค่าตามการทดลอง ใช้โปรแกรมเฉพาะของ Arduino ในการเขียนโค้ดและแสดงผลค่าตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและความรู้ที่เกี่ยวข้อง

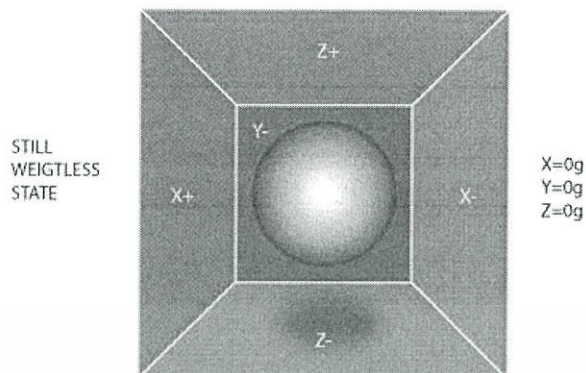
Sensor อุปกรณ์ตรวจจับสัญญาณหรือปริมาณทางฟิสิกส์ต่างๆ เช่น อุณหภูมิ เสียง แสง การสัมผัส มีการนำระบบเซนเซอร์มาใช้ในอุปกรณ์ประจำวัน เช่น ปุ่มกดลิฟต์ที่ไวต่อการสัมผัส (เซนเซอร์สัมผัส) และคอมไฟที่สลับหรือสว่างขึ้นโดยการสัมผัสที่ฐาน นอกจากนี้ยังมีการใช้งานเซนเซอร์ที่นับไม่ถ้วนที่คนส่วนใหญ่ไม่ได้รับรู้ การใช้งานรวมถึงรถยนต์, เครื่องไฟฟ้า, การบินและอวกาศ, การแพทย์, การผลิต และหุ่นยนต์ เซนเซอร์เป็นอุปกรณ์อย่างหนึ่งที่ตอบสนองต่อปริมาณที่นำเข้าไป โดยการสร้างเอาต์พุตที่เกี่ยวข้องกับการทำงาน ที่มักจะอยู่ในรูปของสัญญาณไฟฟ้าหรือสัญญาณแสง ความไวของเซนเซอร์แสดงให้เห็นด้วยปริมาณการเปลี่ยนแปลงของเอาต์พุตที่จะเปลี่ยนแปลงเมื่อปริมาณที่กำลังทำการวัด เปลี่ยนแปลงไป

เซนเซอร์วัดค่าความเร่ง ความเร็วเชิงมุม และค่าสนามแม่เหล็ก เพื่อนำไปหาทิศทางลักษณะการวางตัวของ Rigid Body ต่างๆ การหามุมต่างๆ เช่น Roll Pitch Yaw ของเครื่องบินหรือวัตถุที่ต้องการ

2.1 หลักการทำงานของเครื่องมือ

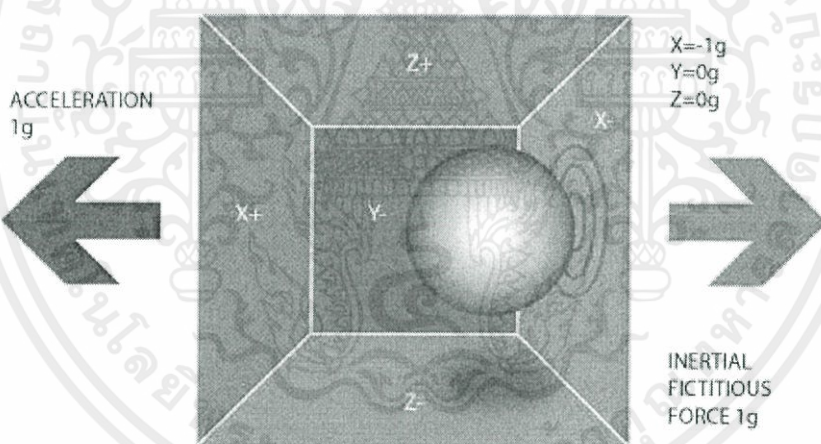
2.1.1 Accelerometer เป็นเซนเซอร์ที่ใช้สำหรับวัดค่าความเร่งใน 3 แกน สามารถทำงานได้เร็ว มีความละเอียดและแม่นยำสูง ค่าที่ได้จากเซนเซอร์นี้สามารถนำไปอินทิเกรตเชิงเวลาเพื่อหาค่าความเร็วและระยะทางได้ แต่เนื่องจากเมื่ออินทิเกรตค่าแล้วอาจเกิดค่าความผิดพลาดสะสม จึงต้องมีการปรับค่าให้เกิดความเหมาะสม ค่าที่ได้สามารถนำมาหาค่ามุม Roll และ Pitch จากค่าความเร่งเนื่องจากแรงโน้มถ่วงโลกได้และการสั่นสะเทือนของวัตถุ บนพื้นฐานของกฎข้อที่สองของนิวตัน (Newton's Second Law; Law of Motion) นั่นคือ ค่าความเร่งที่วัดได้นั้นมาจากการวัดแรงที่กระทำต่อเซนเซอร์ และเป็นผลให้เกิดความเร่ง ซึ่งหลักการทำงานนั้นให้จินตนาการถึงกล่องลูกบาศก์ที่อยู่ในบรรจุลูกบอล และผนังลูกบาศก์แต่ละด้านมีความไวต่อแรงภายนอกที่มากระทำ โดยขณะกล่องใบดังกล่าวตั้งอยู่ในสถานที่ๆ ไม่มีแรงใดๆ กระทำต่อลูกบอล รวมถึงแรงโน้มถ่วงโลกด้วย ลูกบอลจะลอยอยู่บริเวณกึ่งกลางของกล่อง แสดงดังรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 ตั้ววัดความเร่งเมื่อปราศจากแรงกระทำจากภายนอก

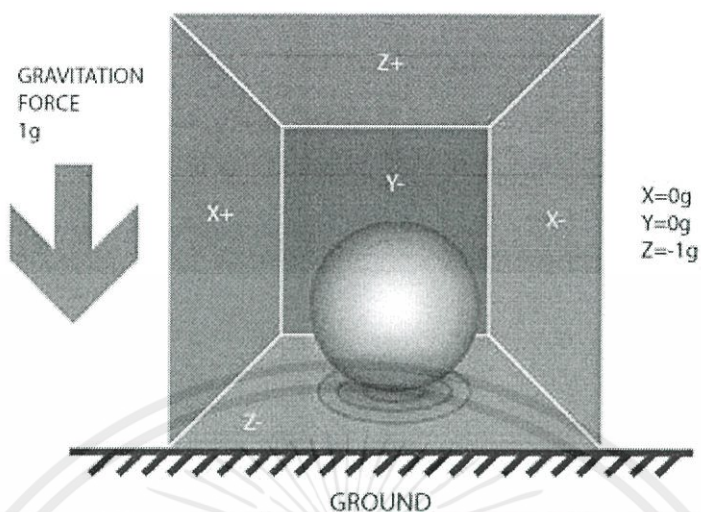
เมื่อเคลื่อนที่ไปทางซ้ายด้วยความเร่ง $1g$ (9.81) ลูกบอลจะเคลื่อนที่ไปชนผนังลูกบาศก์ด้าน X- แรงที่ลูกบอลกระทำต่อผนังลูกบาศก์นี้ คือ ค่าความเร่งที่เซนเซอร์วัดได้นั้นเองโดยมีค่าเท่ากับ $-1g$ ในแนวแกน X แสดงดังรูปที่ 2.2



รูปที่ 2.2 ตั้ววัดความเร่งเมื่อเคลื่อนที่ไปทางซ้ายด้วยความเร่ง $1g$

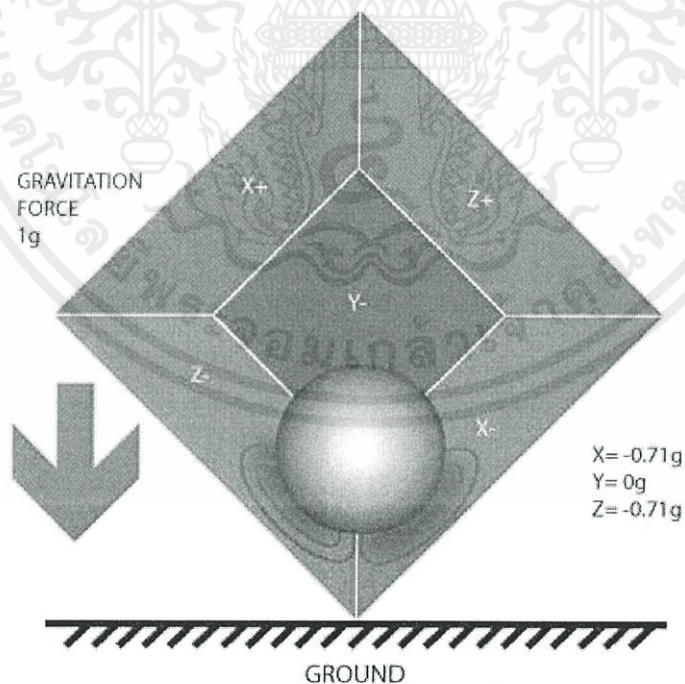
เนื่องจากตั้ววัดความเร่งวัดแรงที่กระทำต่อเซนเซอร์ ค่าเอาต์พุตจึงมีผลของแรงโน้มถ่วงโลกประกอบอยู่ด้วยเสมอ นั่นคือแม้ขณะตั้ววัดความเร่งไม่มีการเคลื่อนที่ค่าเอาต์พุตที่วัดได้จะมีค่าเท่ากับ $-1g$ ในแนวแกน Z- แสดงดังรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ผลจากแรงโน้มถ่วงโลกที่มีต่อตัววัดความเร่ง

และด้วยคุณสมบัตินี้เอง จึงทำให้เราสามารถคำนวณมุมเอียงของเซ็นเซอร์เทียบกับแต่ละแนวแกนได้ เช่น เมื่อหมุนกล่องลูกบาศก์ไปในทิศตามเข็มนาฬิกาเป็นมุม 45 องศา ลูกบอลจะก่อให้เกิดแรงกระทำต่อผนังลูกบาศก์ด้าน Z- และ X- ซึ่งค่าความเร่งที่อ่านได้จากเซ็นเซอร์จะมีค่าเท่ากับ g แสดงดังรูปที่ 2.4

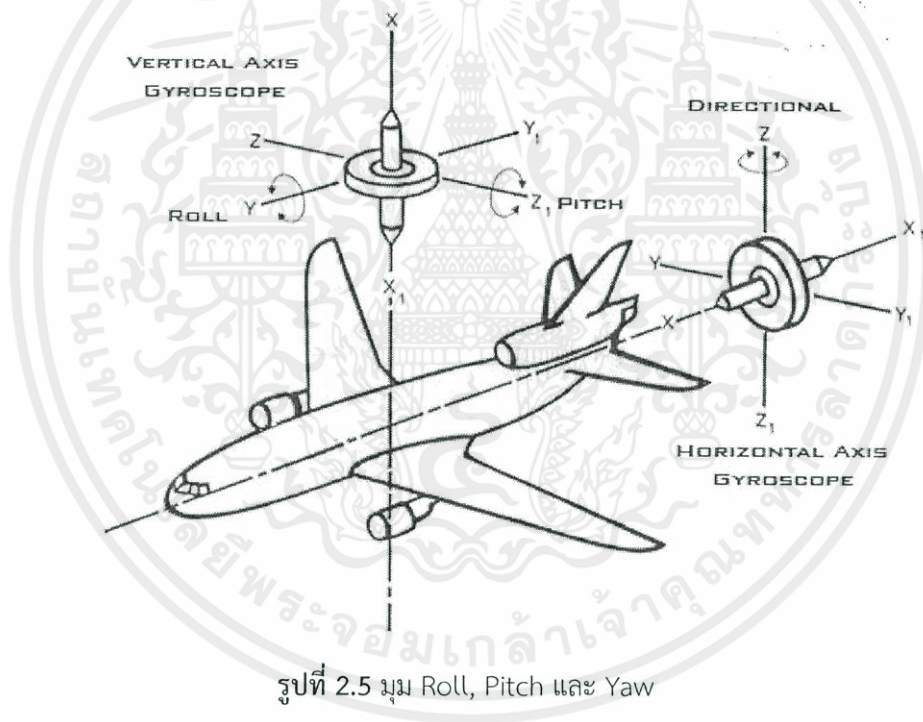


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งยังขอสงวนสิทธิ์ในข้อมูลและข้อมูลอ้างอิงต่างๆที่ปรากฏในเอกสารฉบับนี้ที่มีการนำไปใช้

รูปที่ 2.4 ตัววัดความเร่งเมื่อหมุนไป 45 องศา ทิศตามเข็มนาฬิกา

อย่างไรก็ตาม ตัววัดความเร่งเพียงอย่างเดียวไม่อาจให้ข้อมูลที่ถูกต้องแม่นยำเพียงพอ เนื่องจากผลของการเคลื่อนที่จะทำให้เกิดความเร่งซึ่งทำให้เกิดความผิดพลาดได้ จึงจำเป็นต้องมีการใช้งานร่วมกับเซ็นเซอร์อีกชนิด นั่นคือ ไจโรสโคป

2.1.2 Gyroscope เป็นเซ็นเซอร์ที่ใช้วัดค่าความเร็วเชิงมุมใน 3 แกน ค่าที่ได้จากเซ็นเซอร์ตัวนี้สามารถนำมาหาอินทิเกรตเพื่อหาตำแหน่งของวัตถุในแกน โดยการอินทิเกรตจะทำให้เกิดความผิดพลาดสะสม ไจโรสโคป (Gyroscope) เซ็นเซอร์ที่ใช้วัดความเร็วเชิงมุมในการเคลื่อนที่ของวัตถุรอบแกนหนึ่งๆ ในหน่วยรอบต่อนาที (RPM) หรือองศาต่อวินาที ($^{\circ}/s$) ซึ่งมุมที่หมุนรอบแกน X, Y และ Z เรียกว่า Roll, Pitch และ Yaw แสดงดังรูปที่ 2.5 ตามลำดับ ไจโรสโคปสามารถแบ่งประเภทตามหลักการทำงานได้เป็น 2 ประเภทหลักๆ คือ ไจโรสโคปที่อาศัยหลักการทางกล (Mechanical Gyroscope) และไจโรสโคปที่อาศัยหลักการของแสง (Optical Gyroscope) ซึ่งมีความแม่นยำสูง แต่มีขนาดใหญ่ และราคาแพงมาก จึงไม่เหมาะสมต่อการนำมาใช้งาน ดังนั้นในการทดลองจึงใช้ไจโรสโคปที่อาศัยหลักการทางกล



2.1.3 Magnetometer เป็นเซ็นเซอร์ที่อ่านค่าสนามแม่เหล็กใน 3 แกน มีประสิทธิภาพมากแต่จะเกิดสิ่งรบกวนได้ง่าย สามารถนำมาอ้างอิงตำแหน่งของวัตถุให้แม่นยำมากขึ้นกว่าวัตถุนั้นกำลังอ้างอิงทิศใด

2.1.4 Kalman Filter คือ สูตรทางคณิตศาสตร์ใช้สำหรับหาค่าประมาณที่ดีที่สุดของสถานะของระบบโดยนำข้อมูลเกี่ยวกับความไม่แน่นอน ความคลาดเคลื่อนของเซ็นเซอร์ด้านการค้า (Measurement Noise) มาประกอบการพิจารณาบนพื้นฐานของความน่าจะเป็นในลักษณะที่เกื้อกูล

กันอย่างดีที่สุด เนื่องจากค่าที่ได้จากเซนเซอร์แต่ละตัวอาจมีค่าเกิดความไม่แม่นยำ ซึ่งเครื่องมือที่ใช้ในการกรอง (Filter) เพื่อหาข้อมูลค่าที่ดีที่สุด การใช้ Kalman Filter เริ่มต้นด้วยสมมติฐานว่าเราสามารถเขียนแบบจำลองทางคณิตศาสตร์ของระบบในรูปของ Discrete-Time System Model ด้วยสมการ

State equation

$$\dot{x} = Ax + Bu + Gw$$

Measurement equation

$$y = Cx + Du + Hw + v$$

ค่าที่ทำการคาดการณ์ (Predict)

$$\hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1} + B_k u_k$$

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k$$

ค่าที่ปรับปรุง (Update)

$$\tilde{y}_k = z_k - H_k \hat{x}_{k|k-1}$$

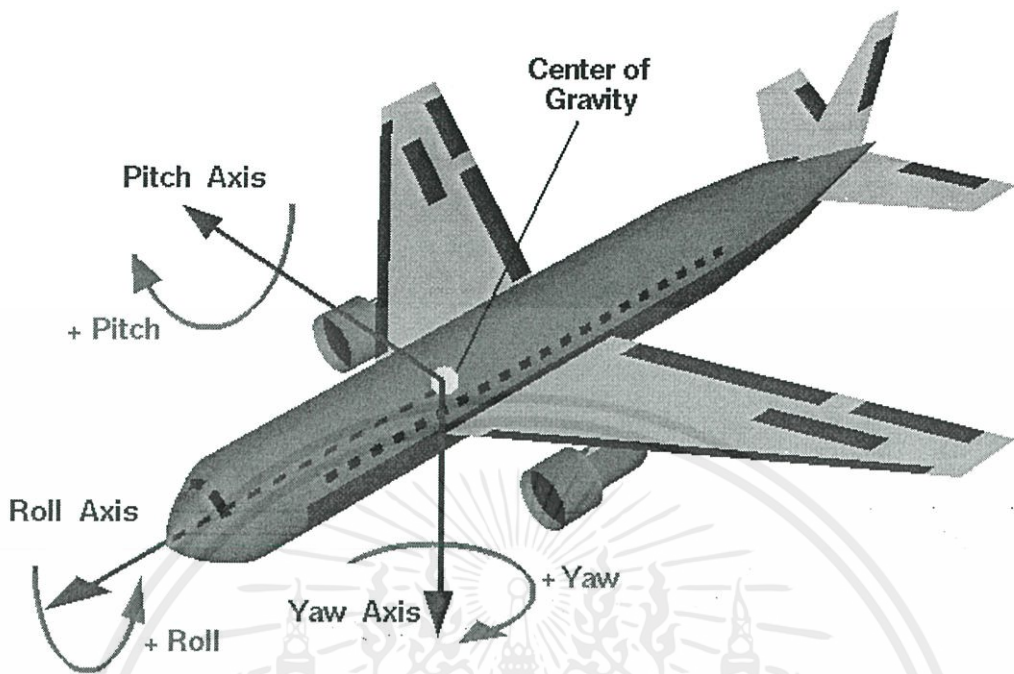
$$S_k = H_k P_{k|k-1} H_k^T + R_k$$

$$K_k = P_{k|k-1} H_k^T S_k^{-1}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



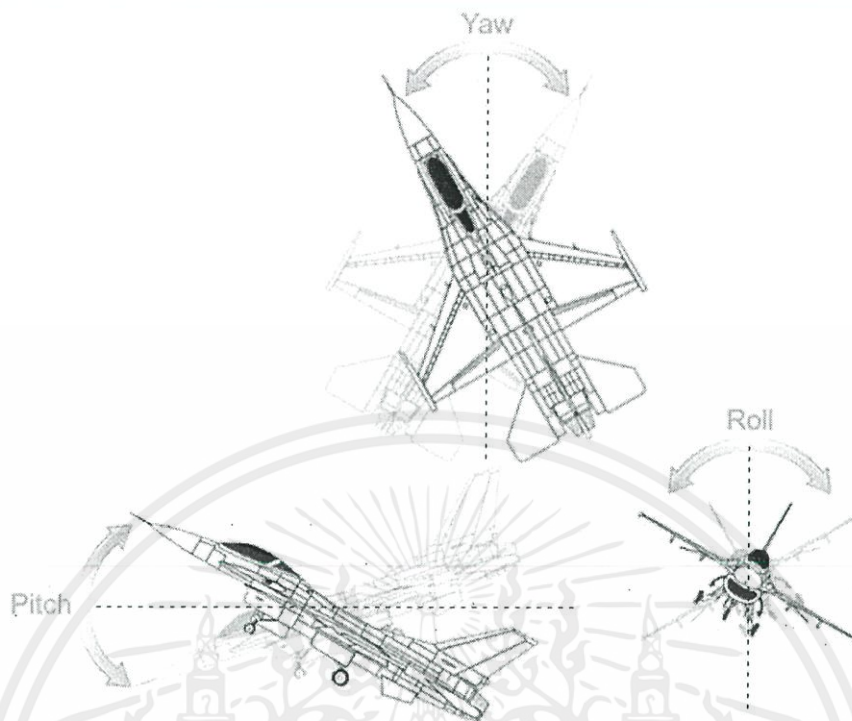
รูปที่ 2.6 จำลองแกนการหมุนของเครื่องบิน

Pitch หรือ Lateral Axis คือ เส้นที่ลากจากปลายปีกข้างหนึ่งไปยังอีกข้างหนึ่ง เมื่อเคลื่อนที่รอบแกนนี้จะทำให้เครื่อง Nose Up หรือ Nose Down

Roll หรือ Longitudinal Axis คือ แกนที่ลากจากหัวเครื่องไปยังท้ายเครื่อง แกนนี้เป็นที่ทำให้เครื่องบินเอียงซ้ายหรือขวา

Yaw หรือ Vertical เป็นแกนที่ลากในแนวตั้งผ่านจุด CG (Center of Gravity) ของเครื่องบินในแนวตั้งฉาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 จำลองแกนการหมุนแบบแยกทั้ง 3 แกน

คำนวณค่ามุม Roll และ Pitch จากเซนเซอร์วัดความเร่ง (Accelerometer)

เนื่องจากข้อจำกัดของ Accelerometer ค่า Yaw ที่วัดได้จะมีความคลาดเคลื่อนสูงจนนำมาใช้ไม่ได้ โดยทั่วไปถ้าใช้แค่ Accelerometer อย่างเดียวมุม Yaw จะไม่ถูกนำมาใช้ ดังนั้นจะพูดถึงแค่ Roll และ Pitch เท่านั้น วิธีคำนวณ Roll และ Pitch นั้นจะขึ้นอยู่กับชนิดของ Accelerometer โดยทั่วไปค่าความเร่งที่วัดได้ในแต่ละแกน (X_g , Y_g , Z_g) จะมีสัญญาณรบกวนผสมอยู่ จึงควรทำการลดปริมาณสัญญาณรบกวนนี้ออกไปก่อนบ้าง โดยใช้

Low-Pass Filter หลังจากที่ได้ความเร่งที่ถูก Filter ของทั้ง 3 แกนมา (fX_g , fY_g , fZ_g) เราสามารถคำนวณหา Pitch และ Roll ได้ดังนี้

$$pitch = \arctan\left(\frac{fY_g}{\sqrt{fX_g^2 + fZ_g^2}}\right)$$

$$roll = \arctan\left(\frac{-fX_g}{fZ_g}\right)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การส่งข้อมูล

2.2.1 การเชื่อมต่ออุปกรณ์แบบ I²C

I²C ย่อมาจาก Inter-IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซี โดยบัส I²C ด้วยจุดมุ่งหมายหลักคือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อ ส่งงาน และ ควบคุมภายใต้สายสัญญาณเพียง 2 เส้น เส้นหนึ่งคือ สายสัญญาณนาฬิกา Serial Clock Line (SCL) ที่ใช้กำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์บนบัส I²C ทำได้ง่ายมาก เพียงต่อสายส่งข้อมูล Serial Data Line (SDA) และสายสัญญาณนาฬิกาของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไป ส่วนการกำหนดแอดเดรสหรือตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสถานะลอจิกที่ขาแอดเดรสของอุปกรณ์แต่ละตัว สายข้อมูลบนบัส I²C มีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรม หรือ SDA (Serial Data Line) ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่า สายสัญญาณนาฬิกาอนุกรม หรือ SCL (Serial Clock Line) ในการอธิบายต่อไปนี้จะเรียกสายสัญญาณทั้งสองว่า SDA และ SCL

หลักการของบัส I²C

บัส I²C ประกอบด้วยสายสัญญาณ 2 เส้นคือ SDA และ SCL อุปกรณ์ที่ต่อพ่วงบนบัสสามารถมีได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูปแบบของการติดต่อบนบัส เพื่อให้ผู้ใช้งานทราบว่าขณะนี้อุปกรณ์ใดติดต่อกันอยู่ และอุปกรณ์ใดเป็นตัวรับหรือส่ง

ในอุปกรณ์บนบัส I²C สามารถเป็นได้ทั้งตัวรับและส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัส I²C ที่ทำหน้าที่เป็นตัวส่งอย่างเดียว

อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส I²C เรียกว่า มาสเตอร์ (Master)

อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส I²C เรียกว่า สเลฟ (Slave)

ข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัส I²C คือ

1. การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น

2. ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ อย่าให้เกิดความเปลี่ยนแปลงเด็ดขาด มิฉะนั้นสัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาณควบคุมแทน

สถานะที่เกิดขึ้นบนบัส I²C มีด้วยกัน 5 สถานะ ดังนี้

1. บัสว่าง (Bus Not Busy) สถานะนี้เกิดขึ้นเมื่อสถานะลอจิกบนสาย SDA และ SCL มีลอจิกสูงทั้งคู่ นั่นหมายความว่า การถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้

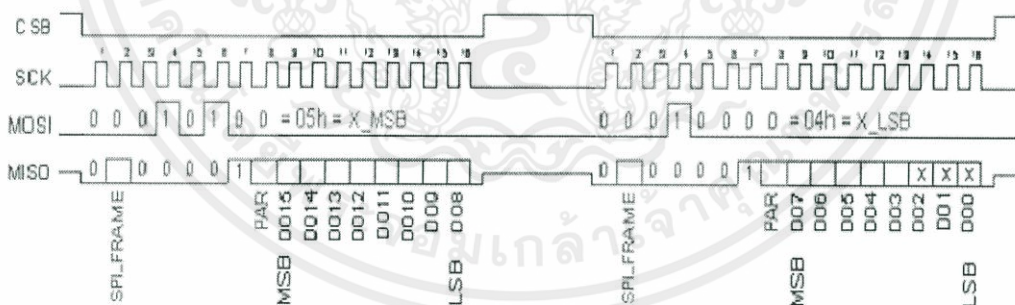
2. เริ่มต้นการถ่ายทอดข้อมูล (Start Data Transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสถานะนี้ว่า สถานะเริ่มต้นเอกสารนี้ (START) การที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ข้อมูลดำรงอยู่บนบัส (Data Valid) สถานะนี้เกิดขึ้นถัดจากสถานะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA ก็คือข้อมูลที่ทำการถ่ายทอด เมื่อสาย SCL มีลอจิกสูง

สถานะที่สาย SDA ต้องคงที่ เพื่อให้อุปกรณ์รับข้อมูลในจังหวะนั้นว่า เป็น "0" หรือ "1" ข้อมูลอาจเกิดความเปลี่ยนแปลงได้ในขณะที่สาย SCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ขา SDA ต้องคงที่ตลอดช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะลอจิกในขณะที่สาย SCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุด หรือสภาวะเริ่มต้นก็ได้ ทำให้ข้อมูลที่ทำการถ่ายทอดเกิดความผิดพลาดเกิดขึ้น

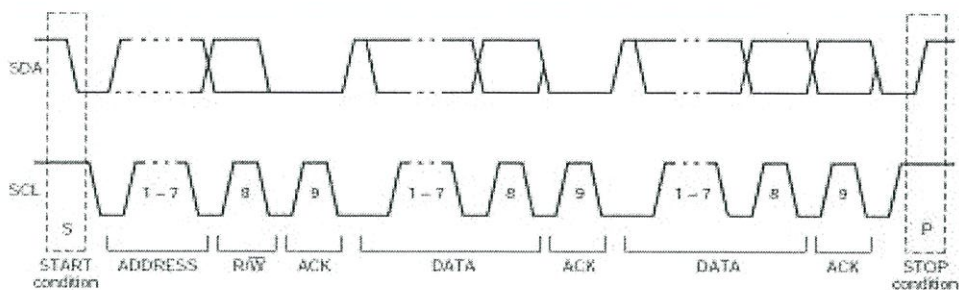
4. รับรู้ข้อมูล (Acknowledge) เกิดขึ้นหลังจากการถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับ เกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิตเรียกว่า บิตรับรู้ (Acknowledge Bit) มีสถานะเป็นลอจิกสูง หลังการส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกา อุปกรณ์สเลฟที่ถูกอ้างอิงในการติดต่อ หรือกำลังติดต่ออยู่ในขณะนั้นก็จะกำเนิดบิตรับรู้ที่มีสถานะลอจิกต่ำเพื่อตอบสนองให้ทราบว่า ได้รับข้อมูลเรียบร้อยแล้ว

5. หยุดการถ่ายทอดข้อมูล (Stop Data Transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูง ในขณะที่สาย SCL มีสถานะลอจิกสูงเรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะหยุด (STOP)



รูปที่ 2.8 การสื่อสารแบบ SPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



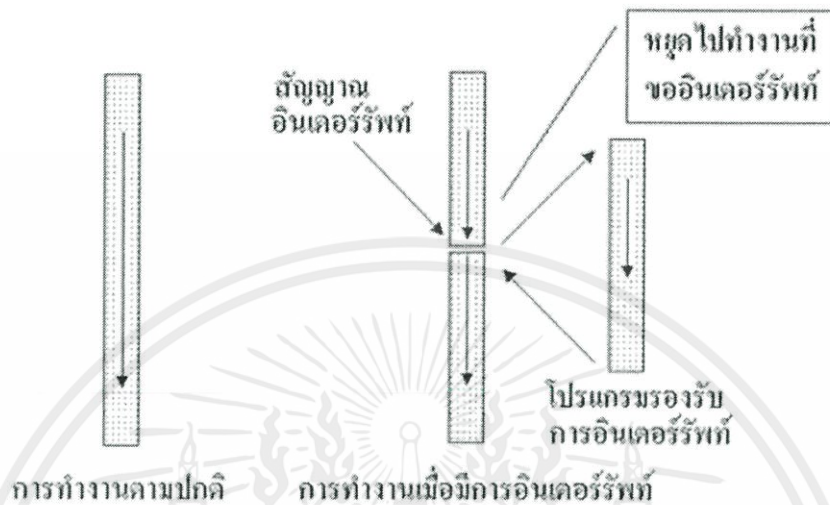
รูปที่ 2.9 การสื่อสารแบบ I²C

2.3 การอินเตอร์รัพท์

การอินเตอร์รัพท์ หรือการขัดจังหวะ คือ การสั่งให้หน่วยประมวลผลหยุดการทำงานชั่วคราว แล้วกระโดดไปทำงานบางอย่างเพื่อตอบสนองการขัดจังหวะนั้น ตัวอย่างของการขัดจังหวะ เช่น อุปกรณ์บางชิ้นได้รับข้อมูล หรือข้อมูลได้รับเขียนเก็บลงในฮาร์ดดิสก์เรียบร้อยแล้ว เป็นต้น เมื่อหน่วยประมวลผลตอบสนองการขัดจังหวะเรียบร้อยแล้ว ก็จะคืนสู่สถานะเดิมและกลับไปประมวลผลงานเก่าที่ประมวลผลค้างไว้เสมือนไม่มีอะไรเกิดขึ้น การขัดจังหวะนี้มีสองประเภทคือ ซอฟต์แวร์อินเตอร์รัพท์ และฮาร์ดแวร์อินเตอร์รัพท์ เรานิยมใช้ซอฟต์แวร์อินเตอร์รัพท์ในการเรียกใช้การบริการต่างๆ ของระบบ ส่วนฮาร์ดแวร์อินเตอร์รัพท์จะนิยมใช้ในการแจ้งการเปลี่ยนสถานะของอุปกรณ์อินพุตเอาต์พุตต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอินเทอร์รัพท์



ประเภทของการอินเทอร์รัพท์

การอินเทอร์รัพท์สามารถแบ่งได้เป็น 2 ประเภทใหญ่ๆ ดังนี้คือ

- Non-maskable Interrupt (NMI) คือ การอินเทอร์รัพท์ที่ซีพียูไม่สามารถปฏิเสธได้ หากมีการอินเทอร์รัพท์ประเภทนี้มา ซีพียูต้องหยุดการทำงานของโปรแกรมที่กำลังทำงานในขณะนั้นโดยไม่มีข้อแม้ และเปลี่ยนไปทำงานให้การอุปการณที่มีการส่งอินเทอร์รัพท์เข้ามา
- Maskable Interrupt (INT) คือ การอินเทอร์รัพท์ที่ซีพียูสามารถปฏิเสธได้ โดยปกติซีพียูจะมีการกำหนดว่าในขณะนั้น ซีพียูจะอยู่ในภาวะ Disable Interrupt (DI) หรือ Enable (EI) ถ้าหากว่าในขณะที่มีการอินเทอร์รัพท์เข้ามานั้น ซีพียูอยู่ในภาวะ Disable Interrupt แล้วการอินเทอร์รัพท์นั้นก็ได้รับการปฏิเสธไป แต่ถ้าเป็น Enable Interrupt การอินเทอร์รัพท์นั้นก็ได้รับการสนองตอบ นอกจากนี้การอินเทอร์รัพท์ยังมีการจัดลำดับความสำคัญของแต่ละอินเทอร์รัพท์ที่เข้ามาอีกด้วย ดังจะกล่าวในหัวข้อถัดไปนี้

การควบคุมอินเทอร์รัพท์

การควบคุมการอินเทอร์รัพท์คือ การควบคุมซีพียูว่าจะให้ตอบรับอินเทอร์รัพท์หรือไม่ เนื่องจากในบางขณะที่ซีพียูกำลังทำงานที่ต่อเนื่องอยู่ เช่น โปรแกรมนาฬิกา ซึ่งหากมีการอินเทอร์รัพท์เข้ามาก็จะก่อให้เกิดความผิดพลาดของนาฬิกา ดังนั้นจึงต้องมีการบังคับไม่ให้มีการสนองตอบต่อการร้องขออินเทอร์รัพท์เข้ามา เมื่อโปรแกรมนาฬิกาเสร็จแล้ว จึงอนุญาตให้มีการตอบสนองได้

คำสั่งที่ใช้ในการเซตหรือรีเซตแฟล็กอินเตอร์รัพท์นั้นเรียกว่า คำสั่งอินเตอร์รัพท์ ซึ่งได้แก่

- DI ซึ่งย่อมาจาก Disable Interrupt เป็นการเซตให้ซีพียูอยู่ในสถานะไม่ตอบสนอง
- EI ซึ่งย่อมาจาก Enable Interrupt ใช้เพื่อรีเซตแฟล็กให้อยู่ในสถานะตอบสนอง

Board	TimerOne	TimerThree
	PWM Pins	PWM Pins
Arduino Leonardo	9, 10, 11	5

Timer0 - An 8 Bit Timer Used by Arduino Functions Delay(), Millis() and Micros().

Timer1 - A 16 Bit Timer Used by The Servo() Library

Timer2 - An 8 Bit Timer Used by The Tone() Library

Timer0

- ใช้เป็น Timer/counter ขนาด 8-Bit (00-FFh)
- รีจิสเตอร์ Timer0 สามารถเขียนอ่านได้
- มีตัวหาร (Prescaler) ขนาด 8-Bit คือสามารถเลือกอัตราส่วนเป็น 2, 4, 8, 16, 32, 64, 128 และ 256 ตัวเพิ่มค่า (Postscaler) เวลาสำหรับ `watchdog` สามารถเลือกอัตราส่วนเป็น 1, 2, 4, 8, 16, 32, 64 และ 128 โดยสามารถเลือกใช้งานได้อย่างใดอย่างหนึ่ง
- สามารถเลือกแหล่งสัญญาณนาฬิกาจากภายใน หรือภายนอกผ่านขา RA4/T0CKI
- มี Interrupt เมื่อการนับเปลี่ยนจาก FFh ไปเป็น 00h (Overflow)
- เมื่อใช้แหล่งสัญญาณนาฬิกาจากภายนอก สามารถเลือกได้ว่าจะนับที่ขอบขาขึ้นหรือขาลง

Timer1

- ใช้เป็น Timer/Counter ขนาด 16-Bit (0000-c)
- รีจิสเตอร์ Timer1 สามารถเขียนอ่านได้
- มีตัวหาร (Prescaler) ขนาด 8-Bit คือ สามารถเลือกอัตราส่วนเป็น 1, 2, 4, 8
- สามารถเลือกแหล่งสัญญาณนาฬิกาจากภายใน หรือภายนอก
- เมื่อทำงานในโหมด Counter จะรับสัญญาณจากภายนอกผ่านทางขา RB6/T1CKI

สามารถให้ Synchronous กับสัญญาณนาฬิกา ภายในได้ และถ้าเป็นแบบ Asynchronous Timer1 ยังคงทำงานอยู่ ถึงแม้และอยู่ในโหมด Sleep

- มี Interrupt เมื่อการนับเปลี่ยนจาก FFFFh ไปเป็น 0000h (Overflow) หาก Enable Interrupt ไว้ เมื่ออยู่ในโหมด Sleep จะทำให้ ไมโครคอนโทรลเลอร์ Wake-up
- มีวงจรออสซิลเลเตอร์พลังงานต่ำสามารถผลิตความถี่ได้สูงสุด 200KHz เมื่อต่อ XTAL ที่ขา RB7/TIOSI, RB6/TIOS
- สามารถใช้โมดูล CCP(Capture/Compare/PWM) ได้ 2 โหมดคือ Capture/Compare เมื่อ Timer1 นับจนกระทั่งมีค่าเท่ากับ ใน CCP1 จะทำให้เกิดการ Interrupt
- CCPR1 มีขนาด 16 บิตประกอบด้วย CCPR1H และ CCPR1L

Timer2

- ใช้เป็น Timer ขนาด 8-Bit (00-FFh) นับได้ 256 ค่า
- รีจิสเตอร์ Timer2 สามารถเขียนอ่านได้
- สามารถใช้สัญญาณนาฬิกาได้จากเฉพาะภายในเท่านั้น
- มี Interrupt เมื่อการนับเปลี่ยนจาก FFh ไปเป็น 00h (Overflow)
- มีทั้งตัวลดทอนการส่งออก (Postscaler) มีค่า 1-16 ,ตัวหารความออสซิลเลเตอร์ (Prescaler) สามารถเลือกอัตราส่วนเป็น 1, 4, 16 และรีจิสเตอร์คาบเวลา (Period Register)
- สามารถใช้โมดูล CCP(Capture/Compare/PWM) ได้ 1 โหมดคือ PWM ในการควบคุม Duty Cycle ของสัญญาณ PWM ออกทางขา RB3/CCP1
- สัญญาณเอาต์พุตของ Timer 2 ยังถูกใช้ใน โมดูลสื่อสาร ข้อมูลอนุกรม แบบซิงโครนัส (Synchronous Serial Port Module :SSP)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

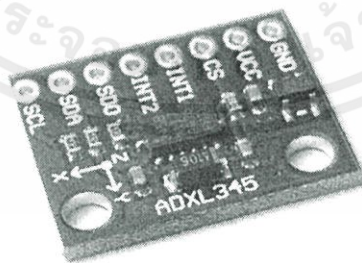
การออกแบบและการเก็บค่าของอุปกรณ์

จากการศึกษาชนิดและการทำงานของเซนเซอร์และตัว Microcontroller แล้วนำอุปกรณ์ที่ทำการต่อวงจรมาประกอบเข้ากับกล่องที่มีทรงเป็นรูปสี่เหลี่ยม เพื่อในขณะที่ยกกล่องหาค่าของมุม ความเร่ง และการตรวจสอบทิศทางในการเคลื่อนที่ในแกน X Y Z เป็นไปตามจุดประสงค์ของเนื้อเรื่องโปรเจกคือ ต้องการนำอุปกรณ์ดังกล่าวไปตรวจสอบสภาวะปัจจุบันและวัดค่าการทรงตัวของหุ่นยนต์หรืออุปกรณ์ต่างๆ ตามคุณสมบัติของเซนเซอร์ Gyro-85

3.1 Sensor Gyro-85

เซนเซอร์ GY-85 ซึ่งเป็นโมดูล Accelerometers, Gyroscope, Compass ซึ่งสามารถทำงานได้ทั้ง 3 อย่างในเวลาเดียวกัน ใช้ในการตรวจสอบทิศทางการเคลื่อนที่ และสามารถใช้ในการตรวจสอบความเร็วในการเปลี่ยนแปลงทิศทางของแกน X Y Z ได้ ถ้าวัตถุเกิดการเคลื่อนที่หรือเอียง Output ของ Accelerometer จะบอกค่าของการเอียงว่าสถานะ ปัจจุบันค่าของ X Y Z ที่เท่าไร แต่ Gyroscope จะวัดค่าได้ตอนที่กำลังเอียงหรือตอนที่กำลังเคลื่อนไหวเท่านั้น เมื่อวัตถุหยุดนิ่งค่าของ Gyroscope จะวัดไม่ได้เพราะไม่มีการเคลื่อนไหว และสามารถเป็นเข็มทิศได้ในเวลาเดียวกัน บนโมดูลประกอบด้วยชิป ADXL345, HMC5883L, ITG3205 ส่งข้อมูลผ่าน Bus I²C ใช้ในการหาค่าของการเคลื่อนไหวทั้ง 3 แกน (Axis) และทิศทางของการเคลื่อนที่

ADXL345 Accelerometer เป็นเซนเซอร์ที่ขนาดเล็กและบาง เป็นซีพียูที่ใช้มีพลังงานต่ำ แต่ประสิทธิภาพสูง ทำหน้าที่วัดความเร่งออกมาให้ทั้ง 3 แกน คือ X, Y, Z สามารถวัดค่าความเร่งได้ถึง $\pm 16g$ และมีความละเอียดถึง 13 Bit



รูปที่ 3.1 ตัววัดความเร่ง ADXL345

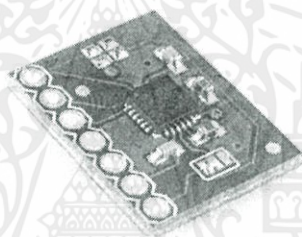
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HMC5883L Magnetometer ทำหน้าที่วัดสนามแม่เหล็กของโลกเพื่อใช้ในการหาทิศทาง (ทิศเหนือได้ออกตก) ดังนั้นจึงเรียกกันโดยทั่วไปว่า Digital Compass หรือเข็มทิศดิจิทัล



รูปที่ 3.2 วัดสนามแม่เหล็ก HMC5883L

ITG3205 Gyroscope เป็นเซนเซอร์ที่ใช้วัดค่าความเร็วเชิงมุมใน 3 แกน



รูปที่ 3.3 วัดค่าความเร็วเชิงมุม ITG3205

General Specifications

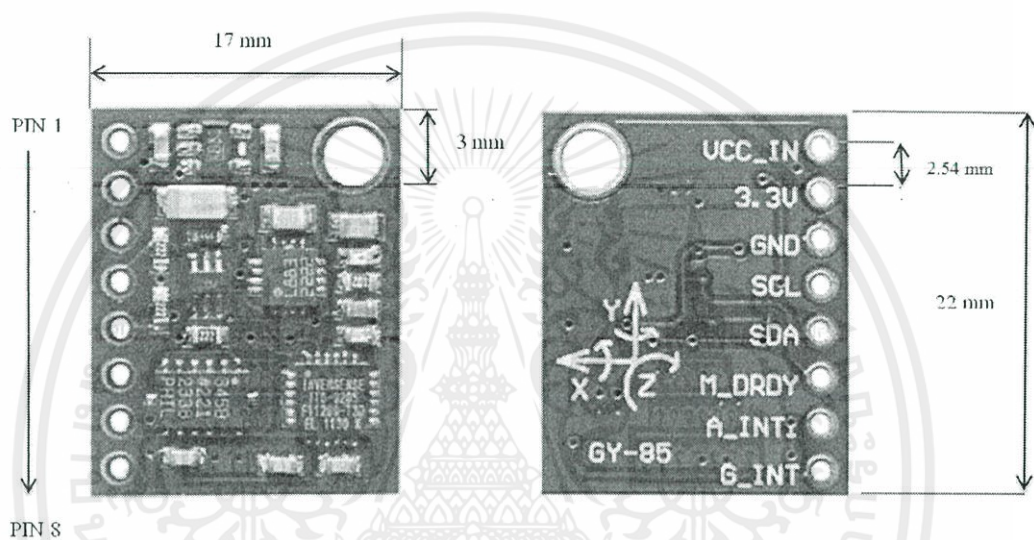
- เชื่อมต่อผ่านบัส I²C
- ใช้ไฟ +3.3 ถึง 5 VDC
- ใช้ชิป ADXL345, HMC5883L และ ITG3205

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณลักษณะ (Specification)

- อุณหภูมิที่รองรับ -40 to +85 °C
- รองรับแรงดัน 3.3 – 5 V
- ทดสอบการตกกระแทกที่ 1.8 เมตร
- ขนาด: 17 mm * 22 mm

โครงสร้าง (Dimension) ของเซนเซอร์

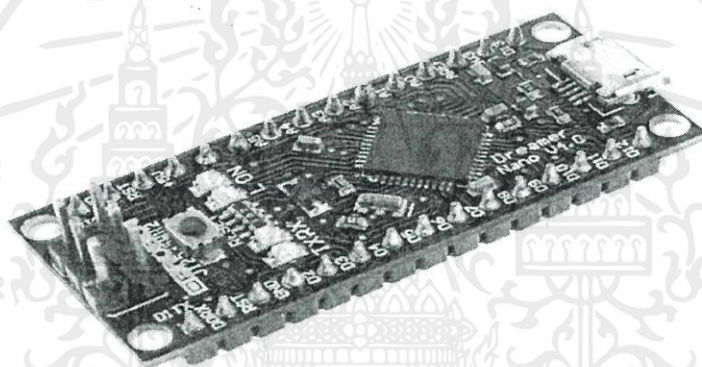


ขาที่	ชื่อ	คำอธิบาย
1	VCC_IN	ขารับไฟ +5 โวลต์ไปที่ Regulate 3.3 โวลต์
2	3.3V	ขาไฟ 3.3 โวลต์
3	GND	กราวด์
4	SCL	ขาสัญญาณนาฬิกา บนบัส I2C
5	SDA	ขาสัญญาณข้อมูล บนบัส I2C
6	M_DRDY	Interrupt ของ HMC5883L
7	A_INT1	Interrupt ของ ADXL345B
8	G_INT	Interrupt ของ ITG3205

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

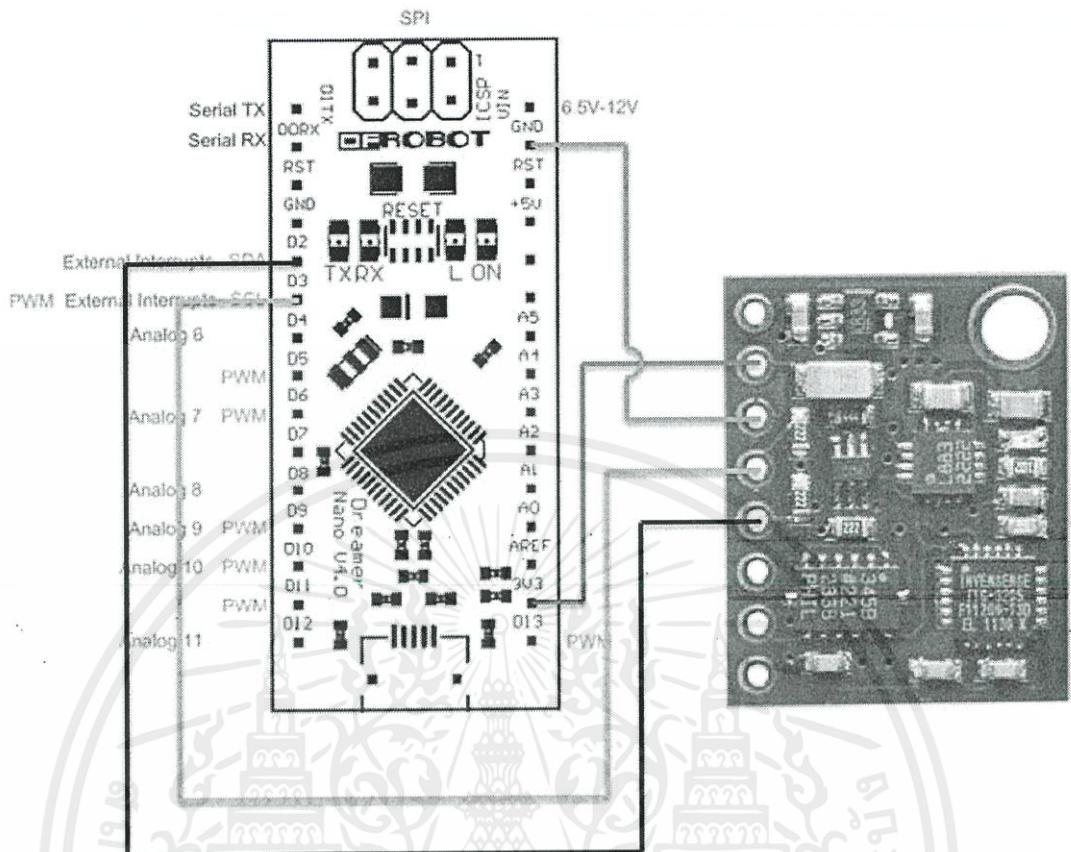
3.2 ไมโครคอนโทรลเลอร์

Arduino คือ แพลตฟอร์มที่มีการพัฒนาแบบ Open Source มีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ผู้ใช้งานสามารถดัดแปลงเพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด และ โปรแกรมได้ รวมถึงสามารถเผยแพร่ และจำหน่ายได้อีกด้วย ตัวบอร์ดถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้น มีลักษณะโครงสร้างภาษาคคล้ายๆ กับภาษา C ซึ่งเป็นภาษาที่ได้รับความนิยมในการเขียนซึ่งเป็นการง่ายต่อผู้ที่เคยใช้ภาษา C อยู่แล้ว นอกจากนี้ยังสามารถดาวน์โหลดไลบรารีต่างๆ เพิ่มเติมได้ ทำให้มีความสะดวกในการเขียนโปรแกรม โดยใช้งานร่วมกับ Arduino Nano V.4 ซึ่งเป็นบอร์ดไมโครคอนโทรลเลอร์ตระกูล AVR (ATmega32u4) มีความถี่คริสตอล 16MHz และมีพอร์ตแรงดันไฟฟ้าเอาต์พุต 3.3 V สำหรับจ่ายแรงดันไฟฟ้าให้กับเซนเซอร์ โดยบอร์ดไมโครคอนโทรลเลอร์ แสดงดังรูปที่ 3.4



รูปที่ 3.4 Arduino Nano V.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



การเชื่อมต่อ

GY-85	Arduino Nano V.4
3.3V	3.3V
GND	GND
SDA	D2
SCL	D3

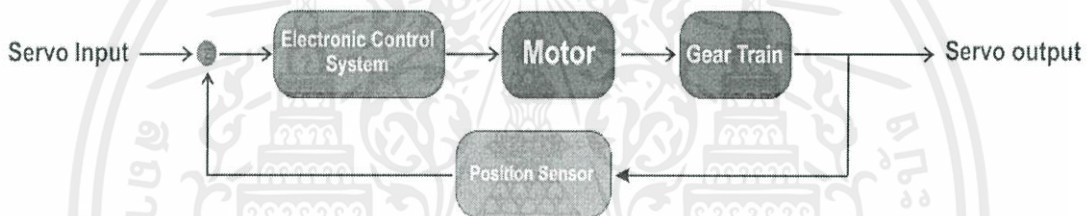
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 Servo Motor

Servo เป็นคำศัพท์ที่ใช้กันทั่วไปในระบบควบคุมอัตโนมัติ มาจากภาษาละตินคำว่า Servus หมายถึง “ทาส” (Slave) ในเชิงความหมายของ Servo Motor ก็คือ Motor ที่เราสามารถสั่งงานหรือตั้งค่าแล้วสามารถควบคุมเครื่องจักรกล หรือระบบการทำงานนั้นๆ ให้เป็นไปตามความต้องการ เช่น ควบคุมความเร็ว (Speed), ควบคุมแรงบิด (Torque), ควบคุมแรง ตำแหน่ง (Position) โดยให้ผลลัพธ์ตามความต้องการที่มีความแม่นยำสูง ตัว Motor จะหมุนไปยังตำแหน่งองศาที่เราสั่งได้เองอย่างถูกต้อง โดยใช้การควบคุมแบบป้อนกลับ (Feedback Control) Servo Motor เป็นอุปกรณ์ที่สามารถควบคุมเครื่องจักรกล หรือระบบการทำงาน

Feedback Control คือ ระบบควบคุมที่มีการวัดค่าเอาต์พุตของระบบนำมาเปรียบเทียบกับค่าอินพุตเพื่อควบคุมและปรับแต่งให้ค่าเอาต์พุตของระบบให้มีค่า เท่ากับ หรือ ใกล้เคียงกับค่าอินพุต

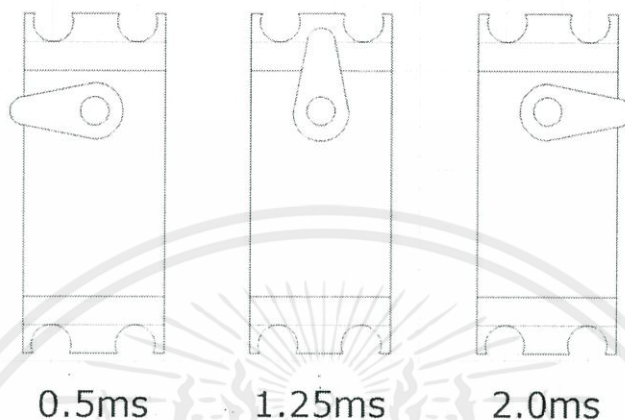
Servo Motor Block Diagram



หลักการทำงานของ MG 60 Servo Motor

เมื่อจ่ายสัญญาณพัลส์เข้ามายัง Servo Motor ส่วนวงจรควบคุม (Electronic Control System) ภายใน Servo จะทำการอ่านและประมวลผลค่าความกว้างของสัญญาณพัลส์ที่ส่งเข้ามาเพื่อแปลค่าเป็นตำแหน่งองศาที่ต้องการให้ Motor หมุนเคลื่อนที่ไปยังตำแหน่งนั้น แล้วส่งคำสั่งไปทำการควบคุมให้ Motor หมุนไปยังตำแหน่งที่ต้องการ โดยมี Position Sensor เป็นตัวเซนเซอร์คอยวัดค่ามุมที่ Motor กำลังหมุนเป็น Feedback กลับมาให้วงจรควบคุมเปรียบเทียบกับค่าอินพุตเพื่อควบคุมให้ได้ตำแหน่งที่ต้องการอย่างถูกต้องแม่นยำ

หากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 0.5 ms ตัว Servo Motor จะหมุนไปทางด้านซ้ายจนสุด ในทางกลับกันหากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 2 ms ตัว Servo Motor จะหมุนไปยังตำแหน่งขวาสุด แต่หากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 1.25 ms ตัว Servo Motor ก็จะหมุนมาอยู่ที่ตำแหน่งตรงกลางพอดี

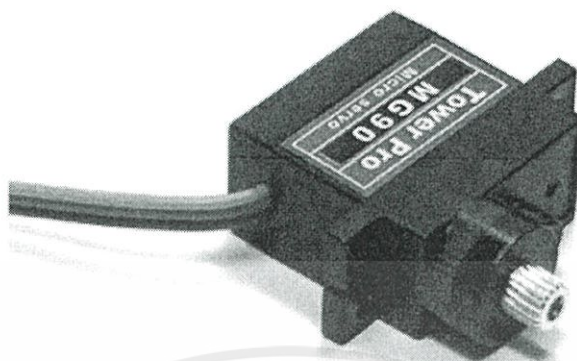


ดังนั้นสามารถกำหนดองศาการหมุนของ Servo Motor ได้โดยการเทียบค่า เช่น Servo Motor สามารถหมุนได้ 180 องศา โดยที่ 0 องศาใช้ความกว้างพัลส์เท่ากับ 1000 us ที่ 180 องศาความกว้างพัลส์เท่ากับ 2000 us เพราะฉะนั้นค่าที่เปลี่ยนไป 1 องศาจะใช้ความกว้างพัลส์ต่างกัน $(2000-1000)/180$ เท่ากับ 5.55 us

จากการหาค่าความกว้างพัลส์ที่มุม 1 องศาข้างต้น หากต้องกำหนดให้ Servo Motor หมุนไปที่มุม 45 องศาจะหาค่าพัลส์ที่ต้องการได้จาก 5.55×45 เท่ากับ 249.75 us แต่ที่มุม 0 องศาเราเริ่มที่ความกว้างพัลส์ 1ms หรือ 1000 us เพราะฉะนั้นความกว้างพัลส์ที่ใช้กำหนดให้ Servo Motor หมุนไปที่ 45 องศา คือ $1000 + 249.75$ เท่ากับประมาณ 1250 us

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

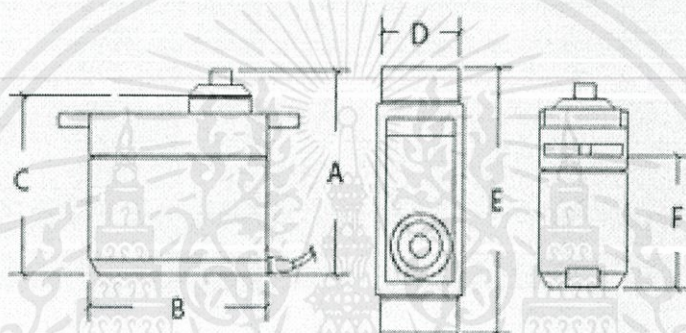
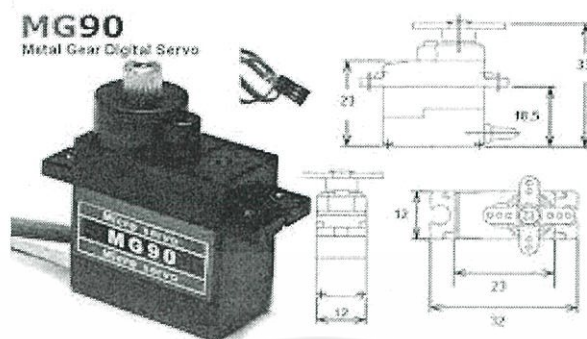
Servo-MG90S



สายสีส้ม คือ สายส่งสัญญาณพัลส์ควบคุม (3-5V)
 สายสีแดง คือ ไฟเลี้ยง (4.8-6V)
 สายสีน้ำตาล คือ กราวด์

NO.	MG90S
Size mm	22.8*12.2*28.5
Weight g	13.4g
Stall torque	1.8kg/cm(4.8v) 2.2kg/cm(6.0v)
Operating speed	0.10sec/60degree(4.8v) 0.08sec/60degree(6.0v)
Operating voltage	4.8V
Dead band width	5 μ s
Remark	สำหรับการใช้งาน Analog Metal Servo ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ศึกษาเท่านั้น ไม่สามารถนำออกจำหน่ายหรือทำซ้ำโดยไม่ได้รับอนุญาต
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Weight (g)	13.4
Torque (kg)	1.8
Speed(Sec/60deg)	0.1
A (mm)	33
B (mm)	23
C (mm)	29
D (mm)	12
E (mm)	32
F (mm)	16

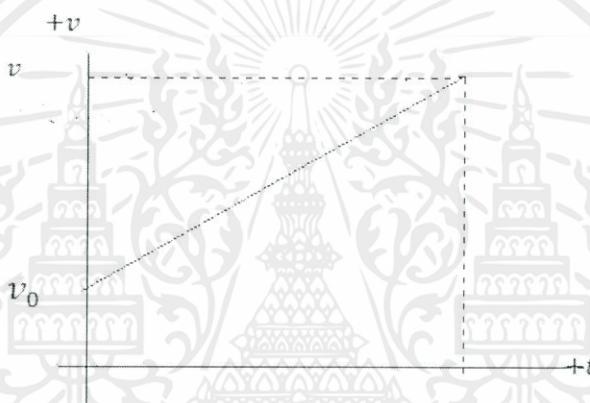
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การประมวลผลข้อมูล

ภายหลังจากการสอบเทียบเซนเซอร์แล้ว ยังต้องมีการคำนวณเพิ่มเติมเพื่อให้ข้อมูลอยู่ในรูปแบบที่ระบบต้องการคือ ระยะทาง และมุม โดยเริ่มศึกษาการประมวลผลจากค่าความเร่งเพื่อหาค่าความเร็ว แล้วจึงนำค่าความเร็วไปหาระยะทาง

การเคลื่อนที่ในกรณีความเร่งเป็นค่าคงตัว

กรณีที่ความเร่งมีค่าคงตัว (Constant Acceleration) นั่นคือ ความเร็วมีการเปลี่ยนแปลงแบบสม่ำเสมอ กราฟของความเร็วที่เพิ่มขึ้นอย่างสม่ำเสมออาจเป็นดังรูป ซึ่งกราฟความเร็วกับ เวลาเป็นกราฟเส้นตรง ความชันที่ทุกจุดบนเส้นตรงคือ ความชันของเส้นตรงนั่นเอง



รูปที่ 3.5 กราฟความเร็วกับเวลา

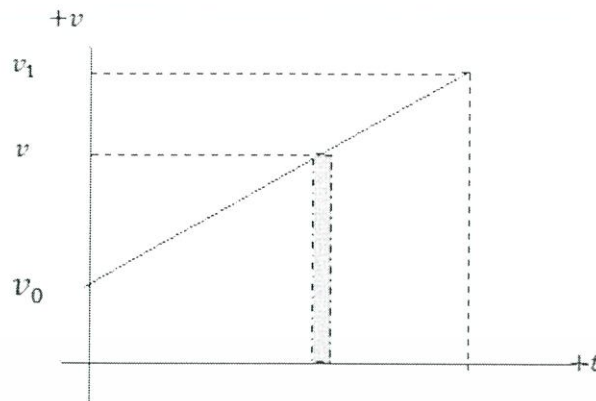
ความชันของเส้นตรงคือ $\frac{v-v_0}{t-0}$ ให้เท่ากับ a_0 ซึ่งเป็นค่าคงตัว

$$\text{ดังนั้น} \quad v-v_0 = a_0 t$$

$$\text{หรือ} \quad v = v_0 + a_0 t$$

ความหมายในสมการนี้คือ v เป็นความเร็วที่เวลา t , v_0 เป็นความเร็วที่เวลา $t = 0$ หรือความเร็วต้น, a_0 เป็นความเร่งค่าหนึ่ง และ t เป็นเวลา ซึ่งจากสมการข้างต้นก็จะเห็นได้ว่า ความเร็วมีค่าเพิ่มขึ้นอย่างสม่ำเสมอ สมการนี้จะใช้ได้ทั้งค่า t ที่เป็น + และ - (ค่า - t หมายถึงเวลาก่อนที่จะเริ่มนับว่าศูนย์) และยังใช้ได้ทั้งกรณีที่ a_0 เป็นบวกและลบ (ถ้า a_0 มีค่าเป็นลบ หมายถึงความเร่งไปทางทิศ -X และมีผลให้ความเร็วลดลงอย่างสม่ำเสมอโดยที่ v อาจจะมีค่าเป็นบวกอยู่)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการเชิงไม่เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 กราฟระหว่างความเร็วและเวลาสำหรับความเร่งคงตัว

การหาระยะทางที่วัตถุเคลื่อนที่ได้ อาจหาได้โดยใช้กราฟดังรูปที่ 3.6 ถ้าใช้ช่วงเวลา Δt เป็นช่วงเวลาสั้นๆ ขณะที่วัตถุมีความเร็ว v ระยะทางที่วัตถุเคลื่อนที่ได้ในช่วงเวลา Δt ก็คือ $v\Delta t$ ซึ่งเป็นช่วงเวลาสั้นๆ เปลี่ยนแปลงไม่มาก ดังนั้นการรวม Δt ทั้งหมดโดยเริ่มจาก $t = 0$ จนถึง $t = t_1$ จะได้ระยะทางของการเคลื่อนที่ทั้งหมด และเป็นระยะทางที่ถูกต้องเมื่อ Δt เข้าใกล้ศูนย์ นั่นคือการอินทิเกรต (Integrate) ของ $v\Delta t$ จาก $t = 0$ จนถึง $t = t_1$ จะให้ระยะทางทั้งหมดสมมติว่าเป็น $x_1 - x_0$

โดยสัญลักษณ์คือ

$$x_1 - x_0 = \int_0^{t_1} v dt$$

ตามรูป $v\Delta t$ เป็นพื้นที่ใต้กราฟที่สูง v และกว้าง Δt การรวมพื้นที่ทั้งหมดจะได้พื้นที่ $v\Delta t$ ใต้กราฟทั้งหมดเป็นรูปสี่เหลี่ยมคางหมูที่มีด้านขนานสูง v_0 กับ v_1 มีความกว้าง $t - 0$ ซึ่งมีค่าเท่ากับ $\frac{1}{2}(v_0 + v_1)(t - 0)$ นั่นคือ $x_1 - x = \frac{(v_0 + v_1)}{2} \Delta t$ และเมื่อแทนค่า $v = v_0 + at$ จะได้ $x_1 - x = v_0 t + \frac{1}{2} at^2$ ขณะที่ $v = v_0 + a_0 t$ ดังนั้น $t = \frac{v - v_0}{a}$ แทนค่าใน $x_1 - x = \frac{(v_0 + v_1)}{2}$ จะได้ $v^2 - v_0^2 = 2a_0(x - x_0)$ ดังนั้นสูตรจะเป็นดังนี้

$$s = \left(\frac{u + v}{2} \right) t$$

$$v = u + at$$

$$s = ut + \frac{1}{2} at^2$$

$$v^2 = u^2 + 2as$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ สูตรเหล่านี้ใช้ได้เฉพาะกรณีที่ a เป็นค่าคงตัวเท่านั้น อย่างไรก็ตามถึงแม้จะนำเอาเอกสารทุกครั้งที่มีการนำไปใช้

ภายหลังจากการสอบเทียบเซนเซอร์แล้ว ยังต้องมีการคำนวณเพิ่มเติมเพื่อให้ข้อมูลอยู่ในรูปแบบที่ระบบต้องการคือ ระยะทาง และมุม โดยระยะทางสามารถหาได้จาก

$$s = s_0 + \int v \, dt$$

$$v = v_0 + \int a \, dt$$

เมื่อ s = ระยะทางที่เวลา t

v = ความเร็วที่เวลา t

a = ความเร่งที่ได้จากตัววัดความเร่ง

s_0 = ระยะทางที่เวลา $t - 1$

v_0 = ความเร็วที่เวลา $t - 1$

dt = ระยะเวลาที่โปรแกรมใช้ในการอ่านค่าจากเซนเซอร์

ส่วนมุม Roll และมุม Pitch เมื่อผ่าน Complementary คำนวณได้จาก

$$\text{Roll} = \alpha \cdot \text{RollGyro} + (1 - \alpha) \text{RollAcc}$$

$$\text{Pitch} = \alpha \cdot \text{PitchGyro} + (1 - \alpha) \text{PitchAcc}$$

$$\text{RollAcc} = \tan^{-1} \left(\frac{-Yg}{\sqrt{Xg^2 + Zg^2}} \right)$$

$$\text{PitchAcc} = \tan^{-1} \left(\frac{Xg}{Zg} \right)$$

เมื่อ Xg = ความเร่งที่ได้จากตัววัดความเร่งในแกน X

Yg = ความเร่งที่ได้จากตัววัดความเร่งในแกน Y

Zg = ความเร่งที่ได้จากตัววัดความเร่งในแกน Z

RollAcc = มุม Roll ที่ได้จากตัววัดความเร่ง

PitchAcc = มุม Pitch ที่ได้จากตัววัดความเร่ง

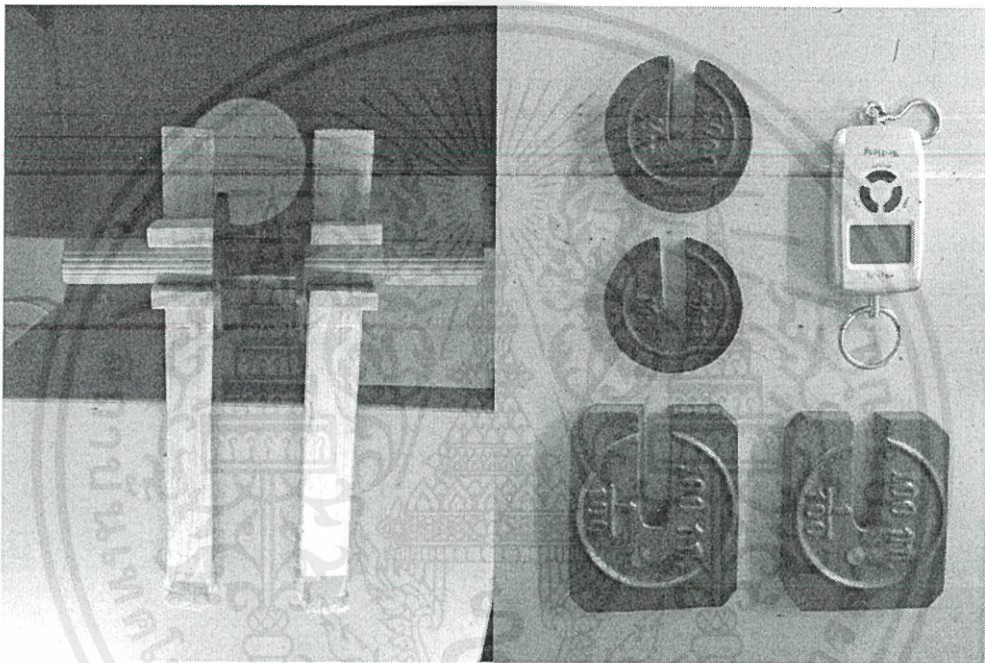
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 การทดลองหาค่าระยะการเคลื่อนที่ตามแนวแกน Y

ในบทนี้จะกล่าวถึงการทดลองเก็บค่าความเร่งจากเซนเซอร์ โดยให้อุปกรณ์ทำการเคลื่อนที่ด้วยความเร่งคงที่ เพื่อนำค่าจากเซนเซอร์มาพิจารณาค่าความเร็วและระยะทาง โดยการสร้างแบบการทดลองลักษณะคล้ายรถกมาเพื่อทำการให้อุปกรณ์เคลื่อนที่ด้วยความเร่งคงที่ ดังรูปที่ 4.1

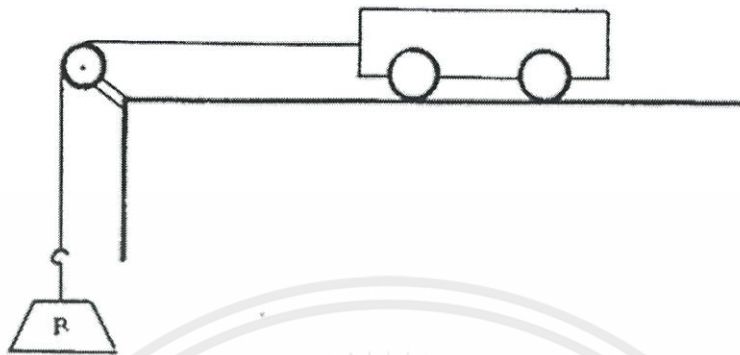


รูปที่ 4.1 อุปกรณ์ที่ใช้ในการทดลอง

โดยกำหนดระยะทางการทดลองเป็น 50 เซนติเมตร และ 80 เซนติเมตร แขนงน้ำหนักที่ถ่วงขนาด 250 กรัม หลังจากการออกแบบโครงสร้างของระบบ จึงทำการทดสอบด้วยการอ่านค่าที่ได้จากเซนเซอร์ โดยข้อมูลที่วัดได้จากระบบมี 3 ข้อมูล ประกอบด้วย ระยะทาง มุม Roll และมุม Pitch โดยระยะทางได้จากการอินทิเกรตค่าความเร่งที่วัดได้จากตัววัดความเร่ง ส่วนมุม Roll และมุม Pitch ได้จากการชดเชยมุมที่ได้จากไจโรสโคปด้วยตัววัดความเร่ง และแม้ว่าไจโรสโคปจะสามารถวัดค่ามุม Yaw ได้ แต่เนื่องจากตัววัดความเร่งไม่สามารถวัดมุมดังกล่าวได้ เนื่องจากผลของแรงโน้มถ่วง

โลกทำให้ไม่มีค่าที่จะนำมาชดเชยปัญหาการ Drift ของไจโรสโคป จึงไม่นำมาใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

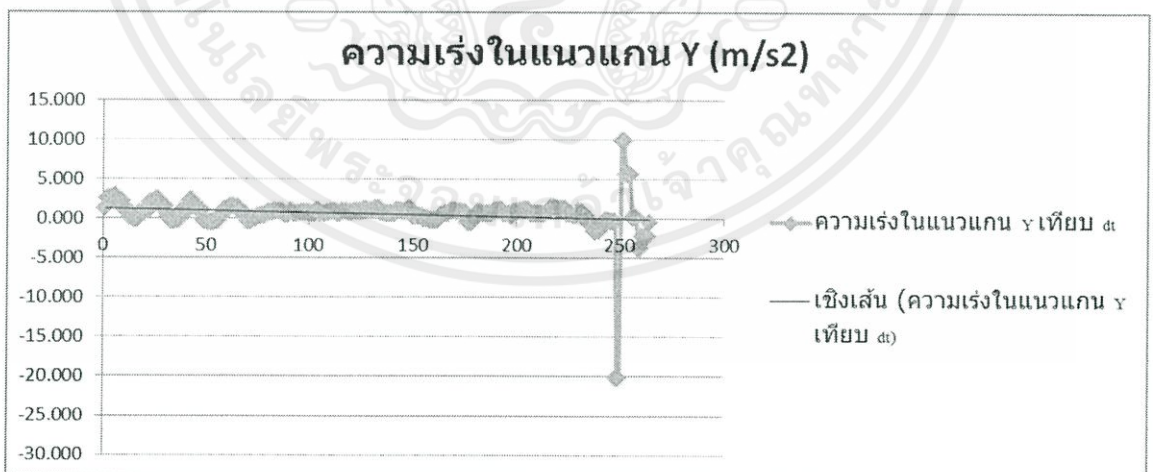


รูปที่ 4.2 การทดลองหาค่าระยะการเคลื่อนที่ตามแนวแกน Y

วางเซนเซอร์ในแนวระนาบ โดยการเก็บค่าจาก 2 มิติ เก็บค่าความเร่งในแนวการเคลื่อนที่แกน Y โดยที่ค่าความเร่งในแนวแกน Z มีค่าเป็นบวก 9.8 m/s^2 จากค่าดิบที่เก็บมาพบว่าความเร่งช่วง 0-250 มีค่าความเร่งคงที่เริ่มจากการปล่อยวัตถุให้เคลื่อนที่เป็นแนวเส้นตรงตามระยะทางที่กำหนด เมื่อถึงช่วงที่ 250-260 เป็นช่วงที่วัตถุหยุดนิ่งกะทันหันด้วยความเร่งที่ติดลบและค่อยๆ หดลงอย่างกะทันหัน ตามรูปที่ 4.2 พิจารณากราฟเป็นกราฟเส้นตรงความเร่งคงที่

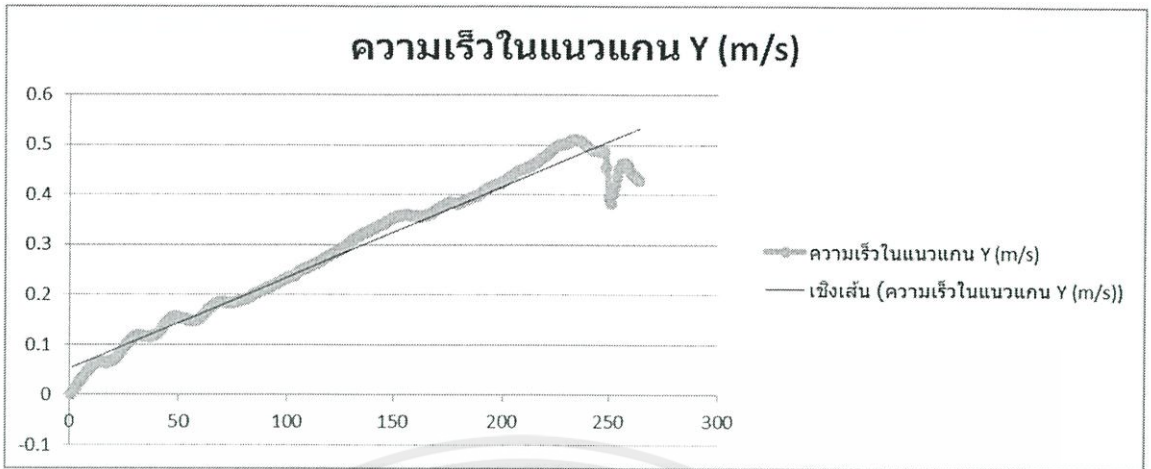
4.1.1 ระยะทาง 50 เซนติเมตร โดยแสดงค่ากราฟ 2 ชุดการทดลองเปรียบเทียบค่าต่างๆ

4.1.1.1 การทดลองชุดที่ 1

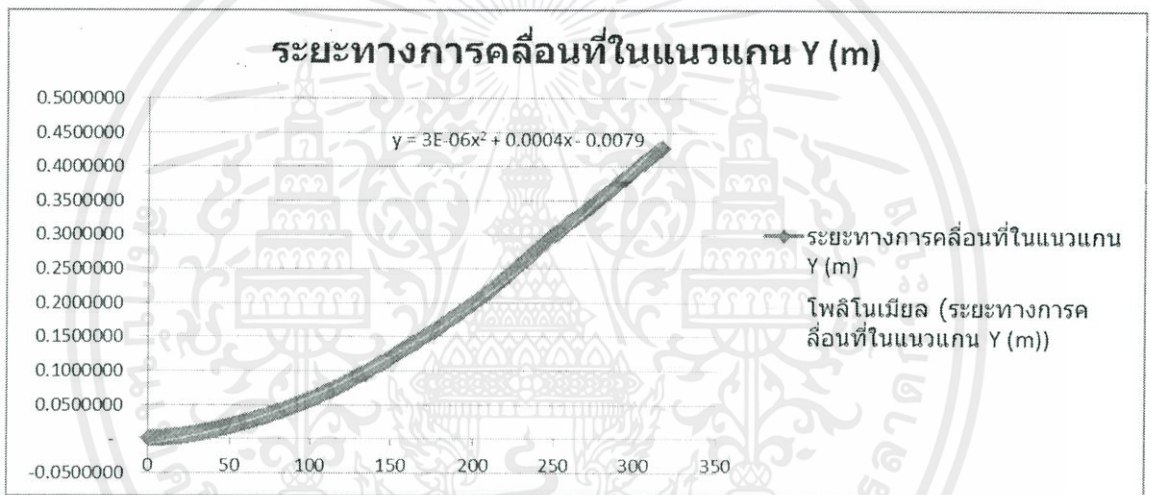


รูปที่ 4.3 ความเร่งในแนวแกน Y

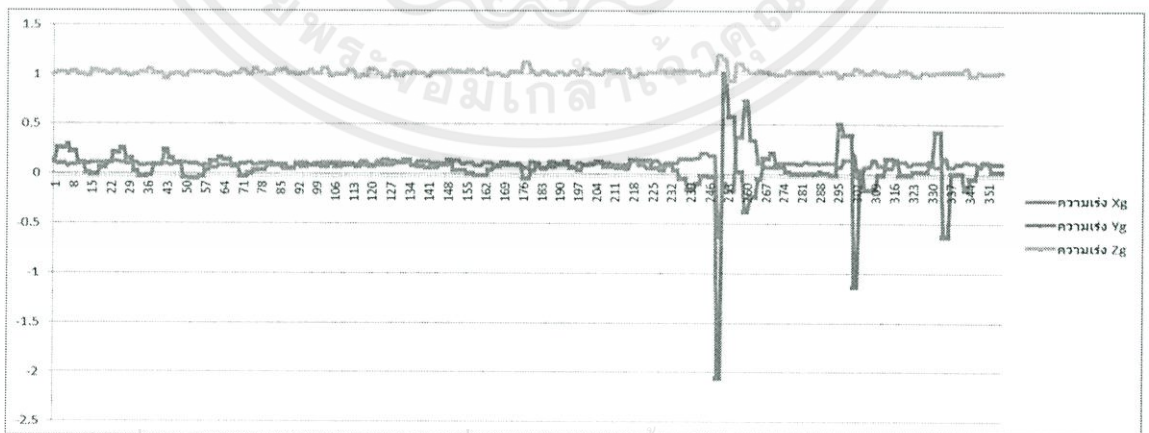
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 ความเร็วในแนวแกน Y ที่เกิดจากการอินทิเกรตความเร่ง



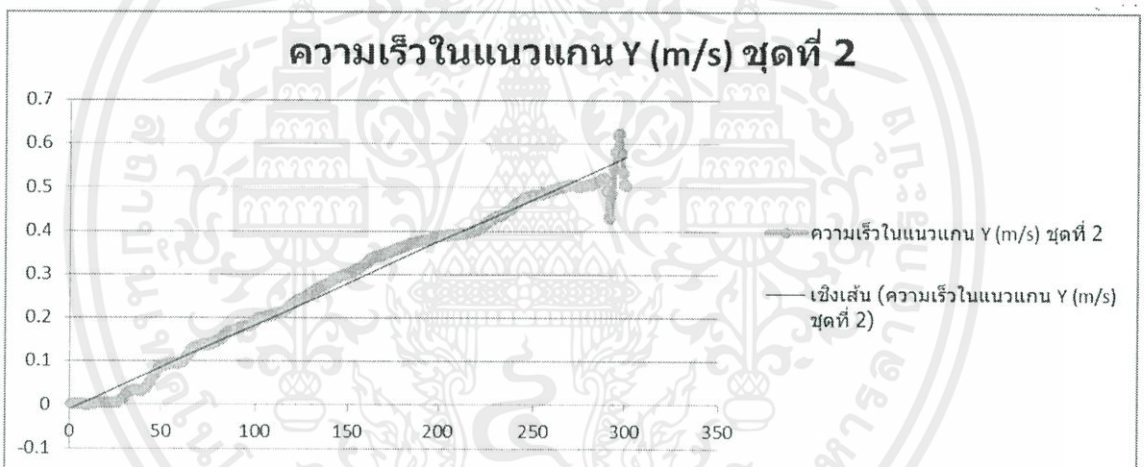
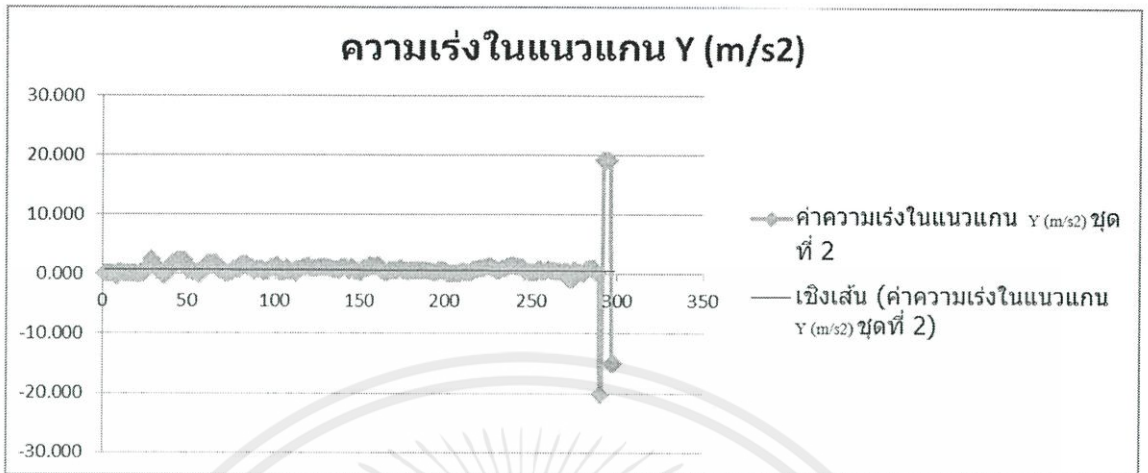
รูปที่ 4.5 ระยะการเคลื่อนที่ในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร่ง



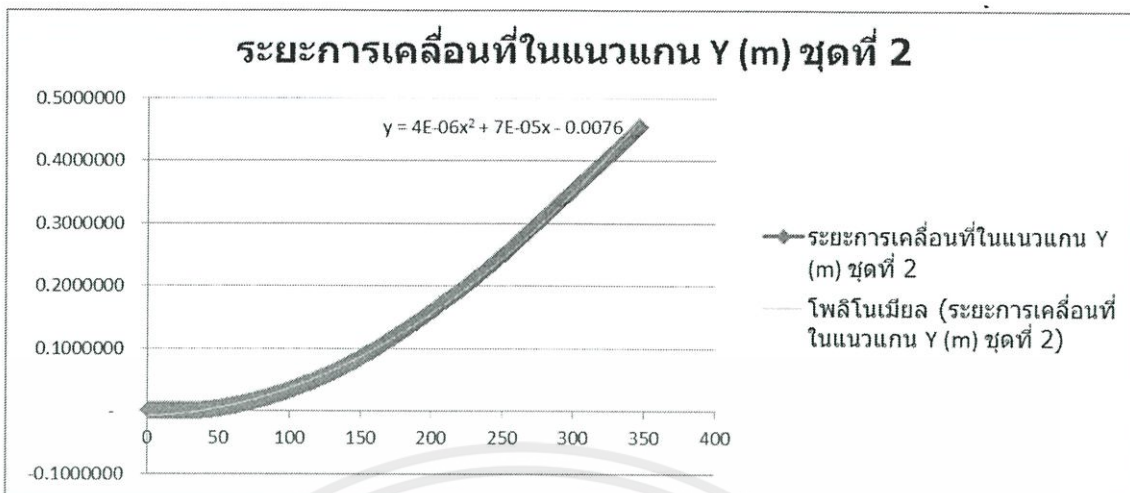
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 4.6 ค่าความเร่งในแนวแกน Xg Yg Zg (m/s²)
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่แบบลงเนื้อหา และต้องอ้างอิงถึงชื่อของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1.2 การทดลองชุดที่ 2



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 ระยะการเคลื่อนที่ในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร็ว



รูปที่ 4.10 ค่าความเร่งในแนวแกน Xg Yg Zg (m/s²)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มต้นทำการทดสอบระบบโดยการเคลื่อนที่เซนเซอร์เป็นระยะทาง 50 ซม. จากนั้นทำการเก็บข้อมูล และนำมาวาดกราฟแสดงความสัมพันธ์ระหว่างข้อมูลกับเวลา โดยโปรแกรมใช้เวลาในการอ่านค่าจากเซนเซอร์ 1 วินาที สามารถตรวจจับการเปลี่ยนแปลงของความเร่งได้ 250-300 ข้อมูล จากนั้นข้อมูลที่เซนเซอร์วัดได้จะผ่านสมการการอินทิเกรต แล้วจึงแสดงผลออกมาเป็นความเร็ว และระยะทาง นอกจากนี้ยังพบว่าความผิดพลาดในการวัดระยะทางของระบบมีความสัมพันธ์กับแรงที่กระทำให้เซนเซอร์เคลื่อนที่ โดยเมื่อควบคุมแรงที่กระทำ จะทำให้ระยะทางที่เซนเซอร์วัดได้มีค่าประมาณ 49.03 ซม. ซึ่งมีความผิดพลาด 1.94%

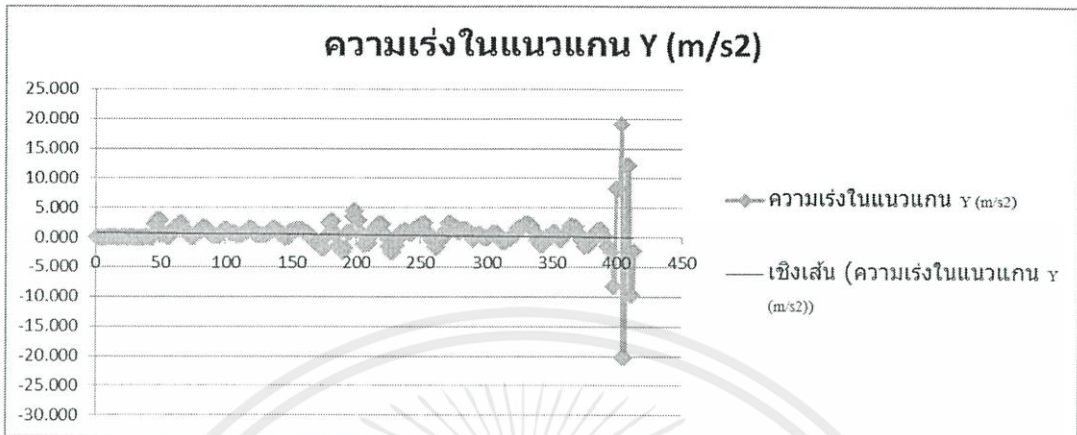
ตารางที่ 4.1 ค่าระยะการเคลื่อนที่เทียบค่าจากค่าจริงกับค่าจากการทดลอง

ระยะทางที่ได้จากเครื่องมือวัด	ครั้งที่	ระยะทางที่เซนเซอร์วัดได้	ค่าเฉลี่ย	ความผิดพลาด
50 cm	1	47.90 cm	49.03 cm	1.94%
	2	48.42 cm		
	3	52.06 cm		
	4	47.72 cm		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ระยะทาง 80 เซนติเมตร โดยแสดงค่ากราฟ 2 ชุดการทดลองเปรียบเทียบค่าต่างๆ

4.1.2.1 การทดลองชุดที่ 1

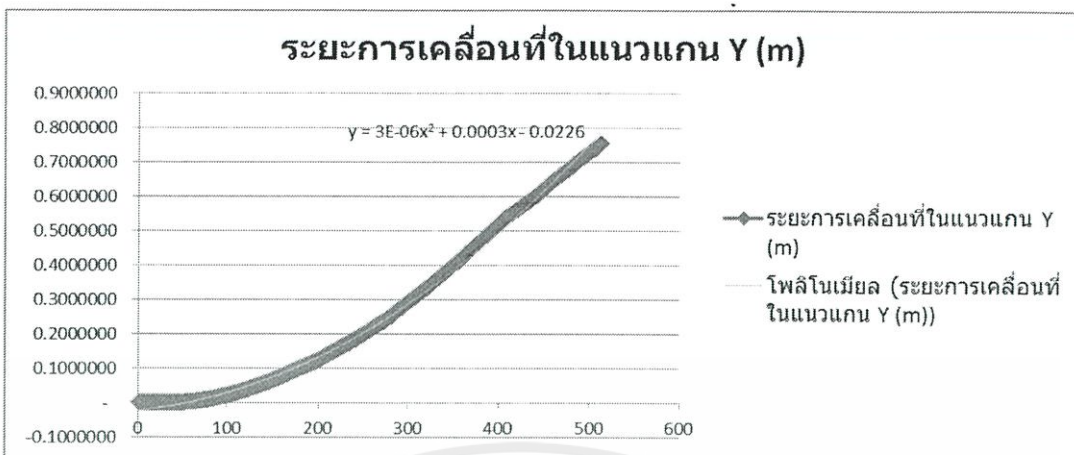


รูปที่ 4.11 ความเร่งในแนวแกน Y

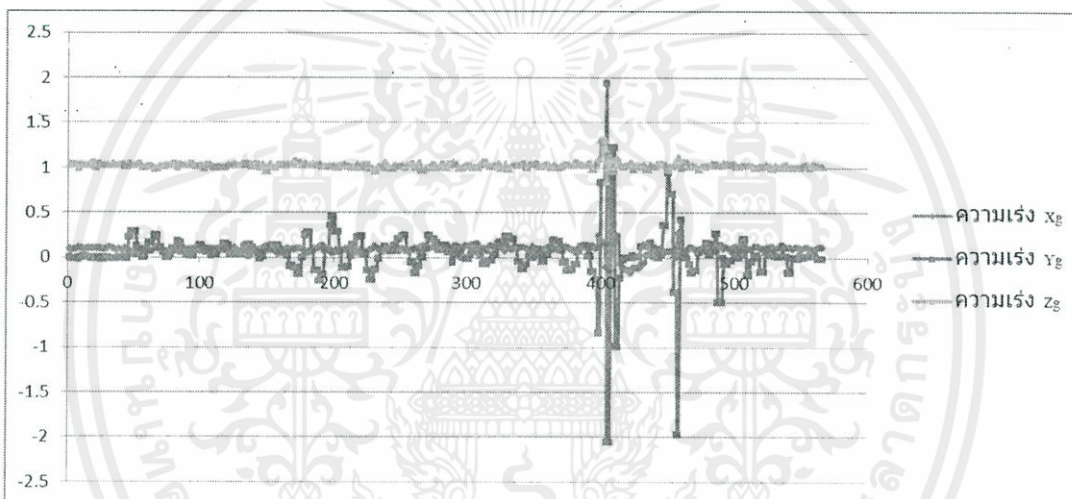


รูปที่ 4.12 ความเร็วในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



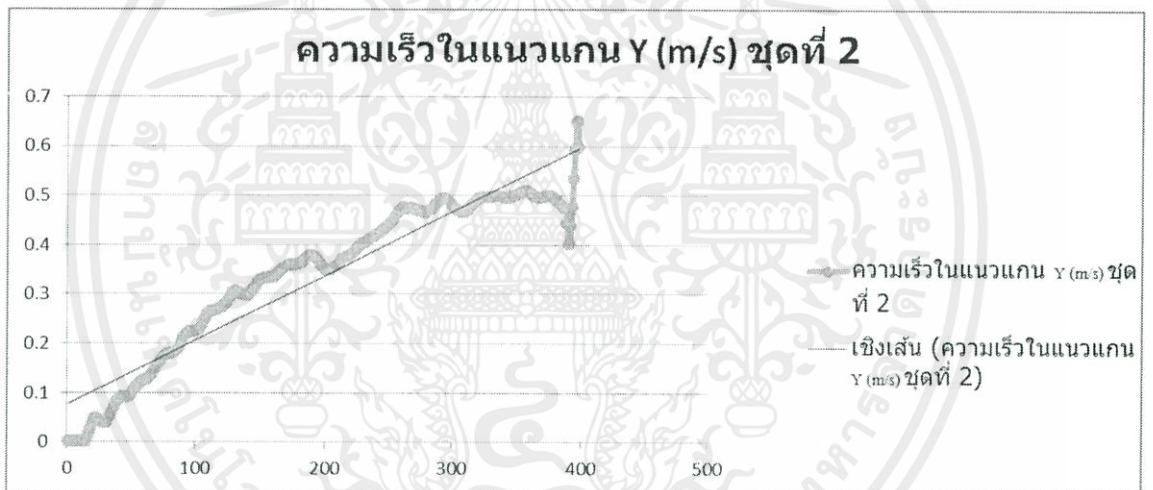
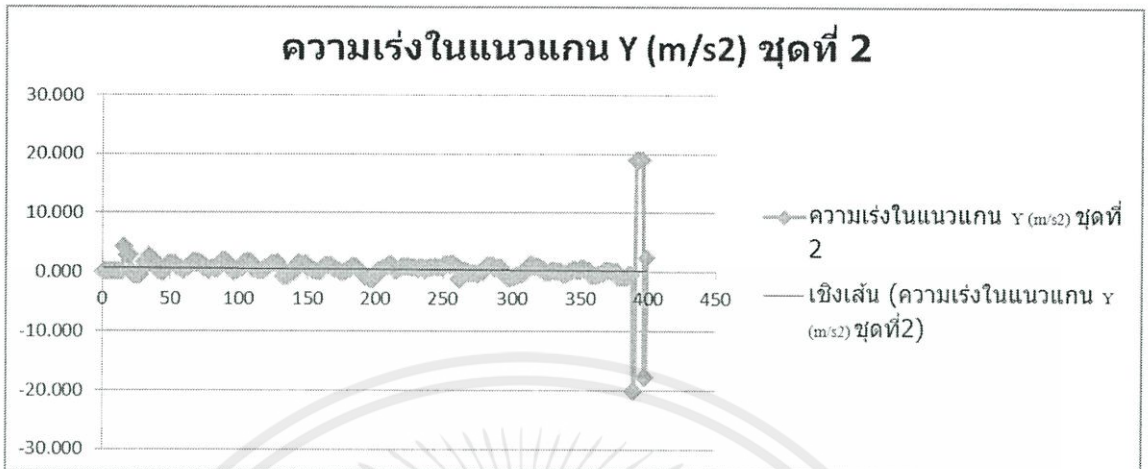
รูปที่ 4.13 ระยะการเคลื่อนที่ในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร็ว



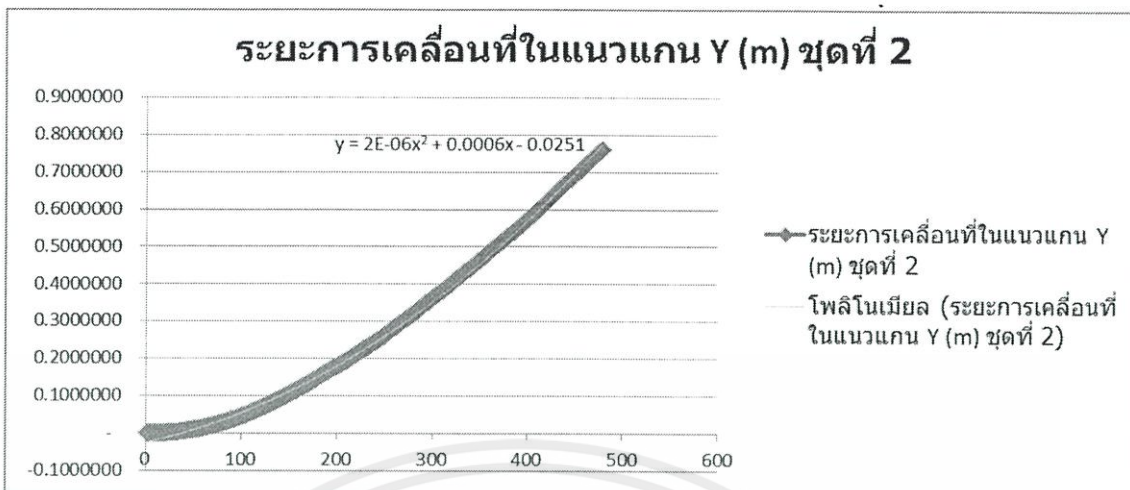
รูปที่ 4.14 ค่าความเร่งในแนวแกน Xg Yg Zg (m/s²)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

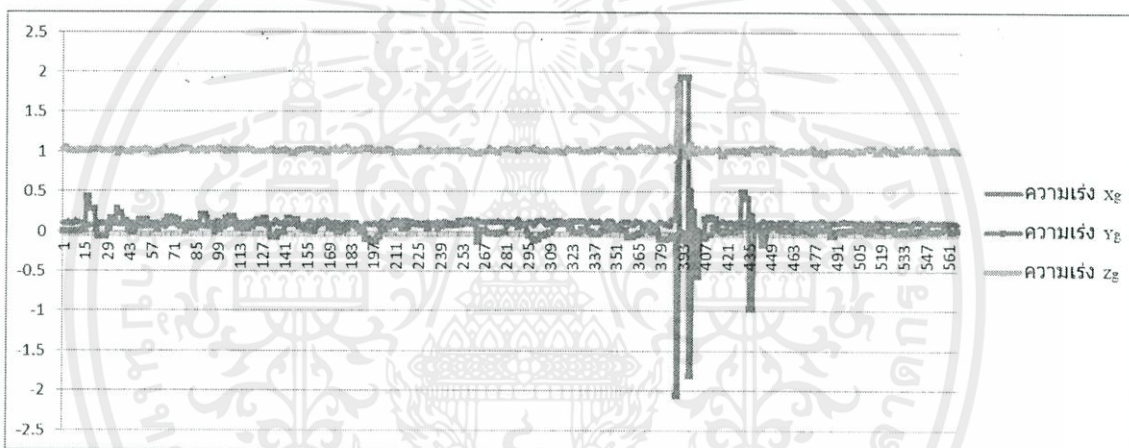
4.1.2.2 การทดลองชุดที่ 2



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.17 ค่าระยะการเคลื่อนที่ในแนวแกน Y ที่เกิดจากการอินทิเกรตค่าความเร็ว



รูปที่ 4.18 ค่าความเร่งในแนวแกน Xg Yg Zg (m/s²)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มต้นทำการทดสอบระบบโดยการเคลื่อนที่เซนเซอร์เป็นระยะทาง 80 ซม. จากนั้นทำการเก็บข้อมูล และนำมาวาดกราฟแสดงความสัมพันธ์ระหว่างข้อมูลกับเวลา โดยโปรแกรมใช้เวลาในการอ่านค่าจากเซนเซอร์ 1 วินาที สามารถตรวจจับการเปลี่ยนแปลงของความเร่งได้ 250-300 ข้อมูล จากนั้นข้อมูลที่เซนเซอร์วัดได้จะผ่านสมการการอินทิเกรต แล้วจึงแสดงผลออกมาเป็นความเร็ว และระยะทาง นอกจากนี้ยังพบว่าความผิดพลาดในการวัดระยะทางของระบบมีความสัมพันธ์กับแรงที่กระทำทำให้เซนเซอร์เคลื่อนที่ โดยเมื่อควบคุมแรงที่กระทำ จะทำให้ระยะทางที่เซนเซอร์วัดได้มีค่าประมาณ 74.21 ซม. ซึ่งมีความผิดพลาด 7.24%

ตารางที่ 4. 2 ค่าระยะการเคลื่อนที่เทียบค่าจากค่าจริงกับค่าจากการทดลอง

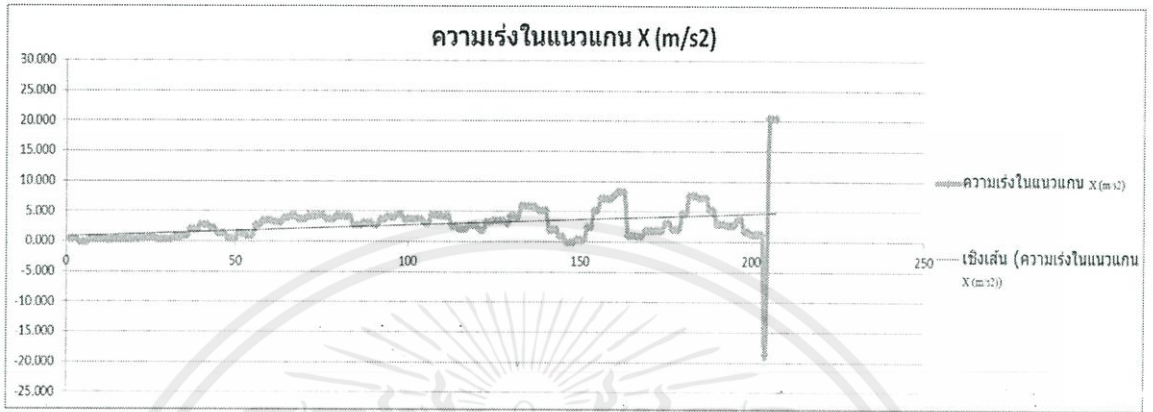
ระยะทางที่ได้จากเครื่องมือวัด	ครั้งที่	ระยะทางที่เซนเซอร์วัดได้	ค่าเฉลี่ย	ความผิดพลาด
80 cm	1	73.91 cm	74.21 cm	7.24%
	2	74.5 cm		
	3	73.21 cm		
	4	75.25 cm		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองหาค่าระยะการเคลื่อนที่ตามแนวแกน X

4.2.1 ระยะทาง 50 เซนติเมตร

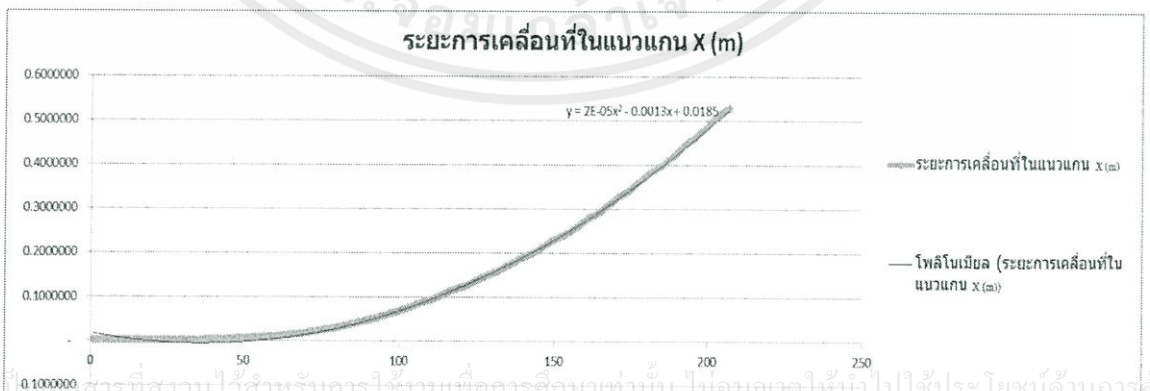
4.2.1.1 การทดลองชุดที่ 1



รูปที่ 4.19 ความเร่งในแนวแกน X



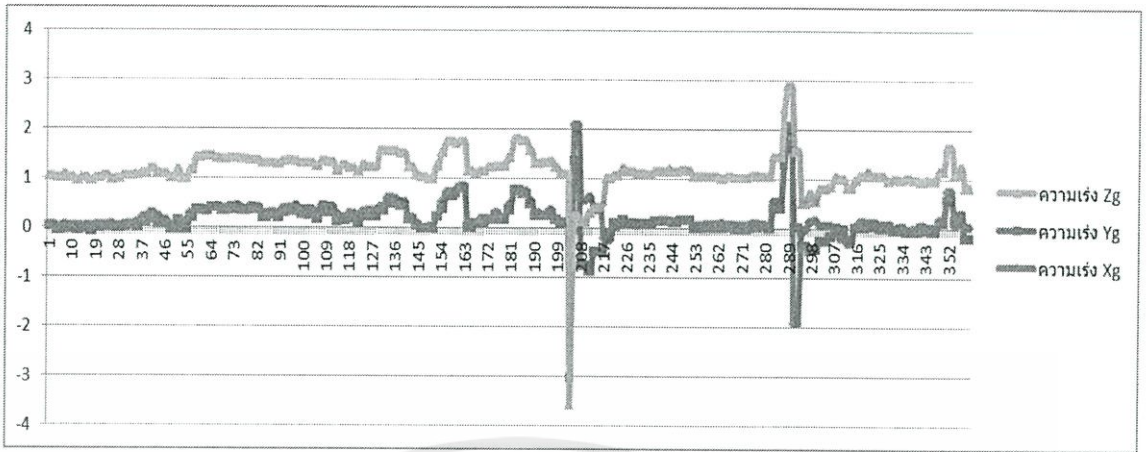
รูปที่ 4.20 ความเร็วในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร่ง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น ผู้ใช้นั้นเป็นผู้รับผิดชอบ และต้องรับผิดชอบหากเอกสารหรือข้อมูลนี้มีการนำไปใช้

รูปที่ 4.21 ระยะการเคลื่อนที่ในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร็ว

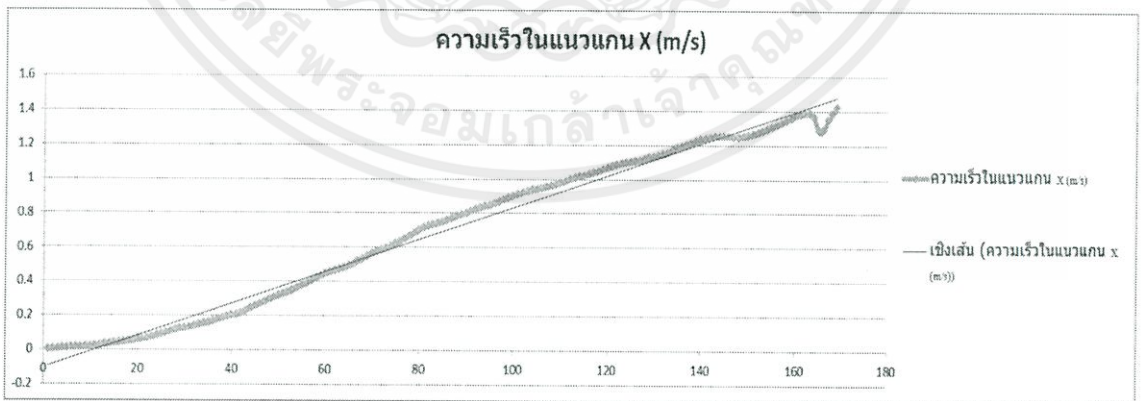


รูปที่ 4.22 ความเร่งในแนวแกนการเคลื่อนที่ด้าน X (Xg Yg Zg)

4.2.1.2 การทดลองชุดที่ 2

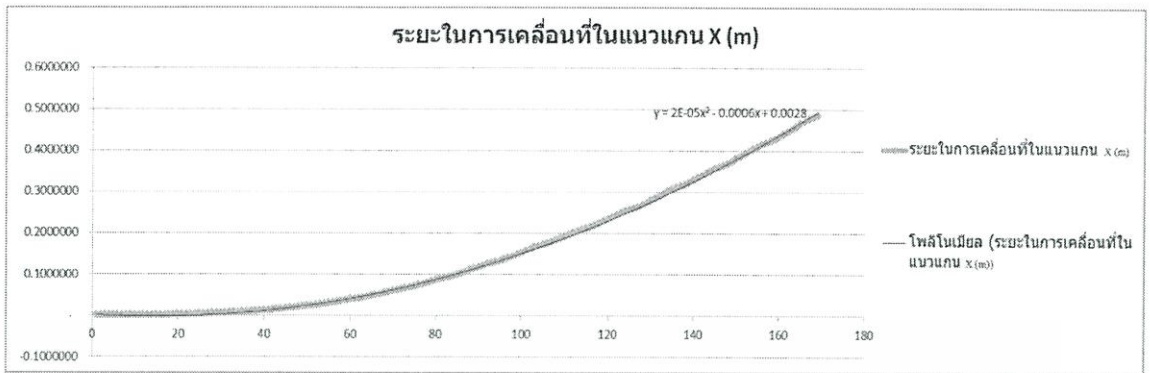


รูปที่ 4.23 ความเร่งในแนวแกน X

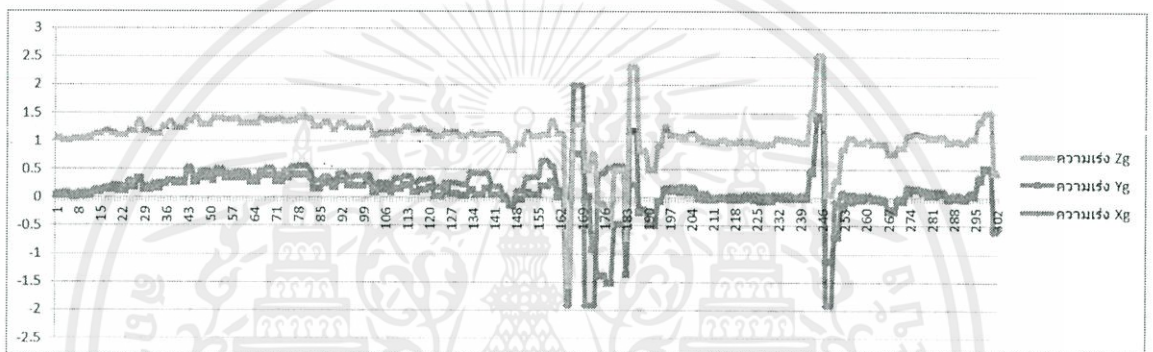


รูปที่ 4.24 ความเร็วในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.25 ระยะการเคลื่อนที่ในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร็ว



รูปที่ 4.26 ความเร่งในแนวแกนการเคลื่อนที่ด้าน X (Xg Yg Zg)

เริ่มต้นทำการทดสอบระบบโดยการเคลื่อนที่เซนเซอร์เป็นระยะทาง 50 ซม. จากนั้นทำการเก็บข้อมูล และนำมาวาดกราฟแสดงความสัมพันธ์ระหว่างข้อมูลกับเวลา โดยโปรแกรมใช้เวลาในการอ่านค่าจากเซนเซอร์ 1 วินาที สามารถตรวจจับการเปลี่ยนแปลงของความเร่งได้ 250-300 ข้อมูล จากนั้นข้อมูลที่เซนเซอร์วัดได้จะผ่านสมการการอินทิเกรต แล้วจึงแสดงผลออกมาเป็นความเร็ว และระยะทาง นอกจากนี้ยังพบว่าความผิดพลาดในการวัดระยะทางของระบบมีความสัมพันธ์กับแรงที่กระทำทำให้เซนเซอร์เคลื่อนที่ โดยเมื่อควบคุมแรงที่กระทำ จะทำให้ระยะทางที่เซนเซอร์วัดได้มีค่าประมาณ 48.625 ซม. ซึ่งมีความผิดพลาด 2.75%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ค่าระยะการเคลื่อนเทียบค่าจากค่าจริงกับค่าจากการทดลอง

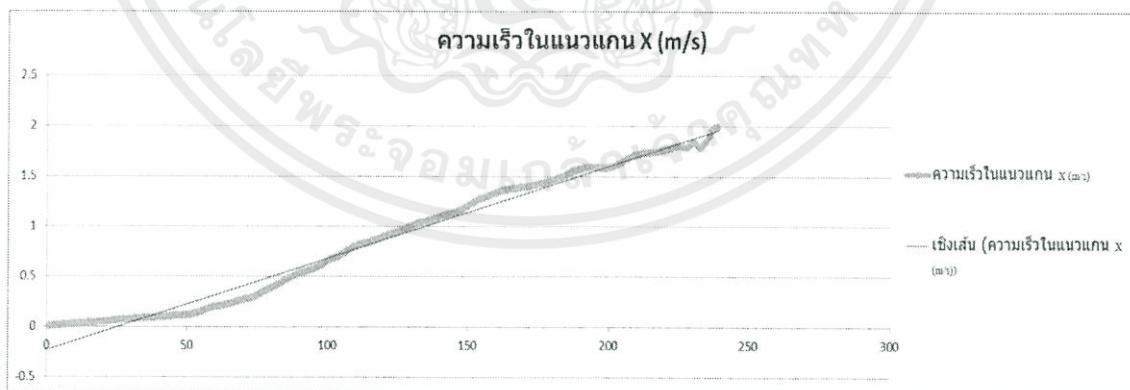
ระยะทางที่ได้จากเครื่องมือวัด	ครั้งที่	ระยะทางที่เซนเซอร์วัดได้	ค่าเฉลี่ย	ความผิดพลาด
50 cm	1	46.53 cm	48.625 cm	2.75%
	2	50.35 cm		
	3	48.95 cm		
	4	48.67 cm		

4.2.2 ระยะทาง 80 เซนติเมตร

4.2.2.1 การทดลองชุดที่ 1

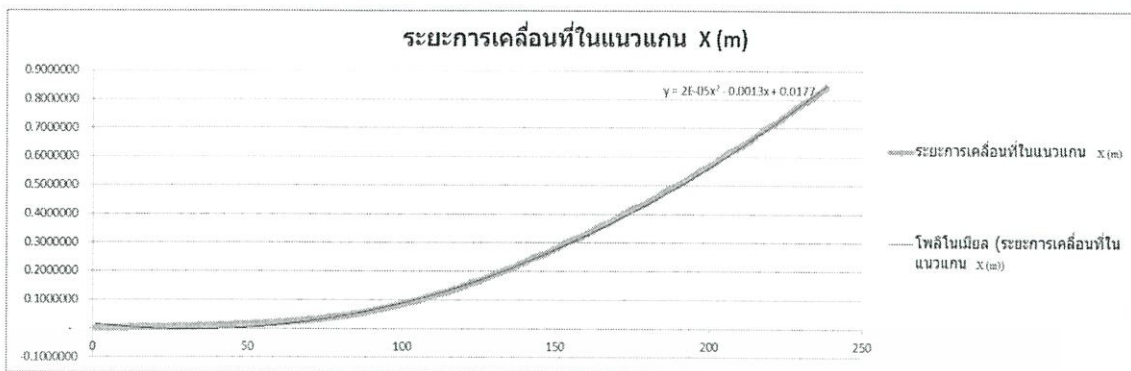


รูปที่ 4.27 ค่าความเร่งในแนวแกน Xg

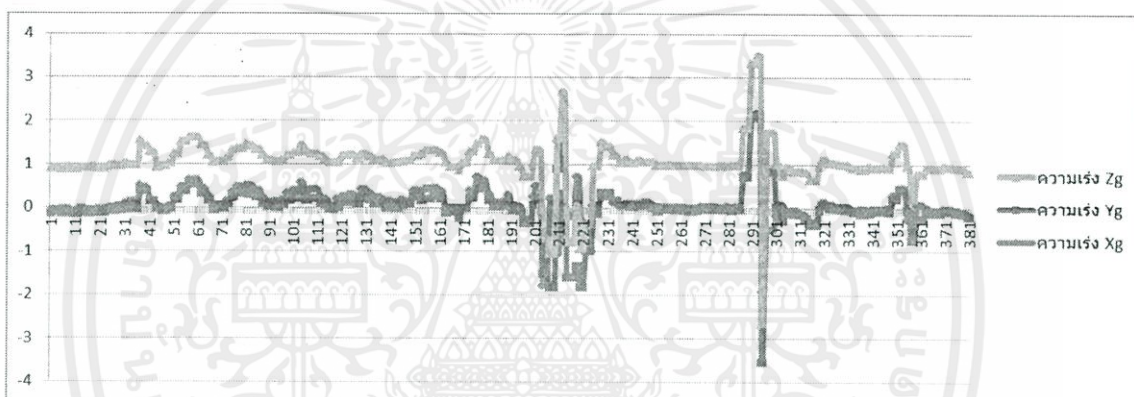


รูปที่ 4.28 ค่าความเร็วในแนวแกน Xg ที่เกิดจากการอินทิเกรตค่าความเร่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.29 ระยะการเคลื่อนที่ในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร็ว



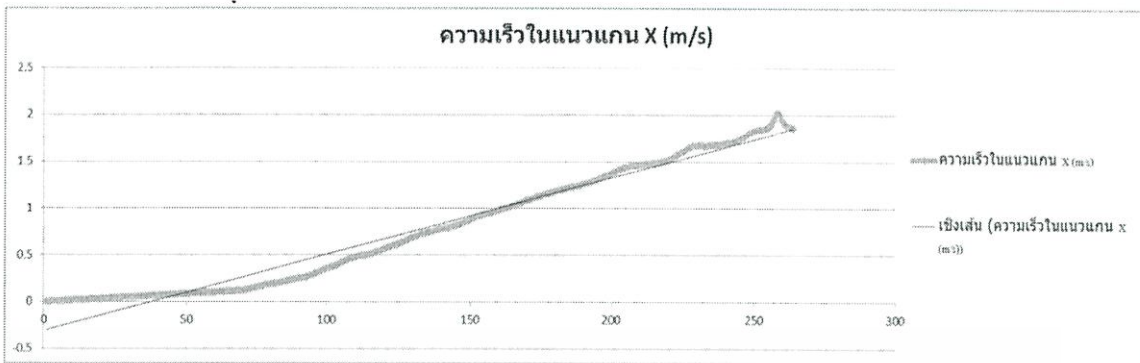
รูปที่ 4.30 ความเร่งในแนวแกนการเคลื่อนที่ด้าน X (Xg Yg Zg)

4.2.2.2 การทดลองที่ 2

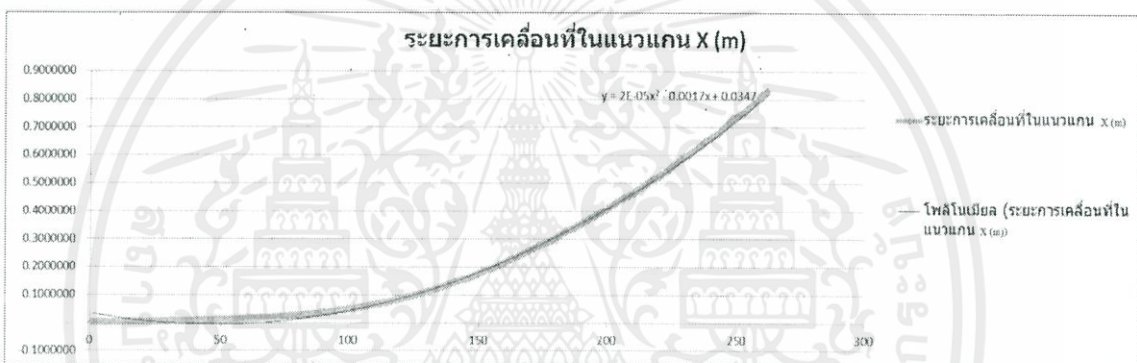


รูปที่ 4.31 ค่าความเร่งในแนวแกน Xg

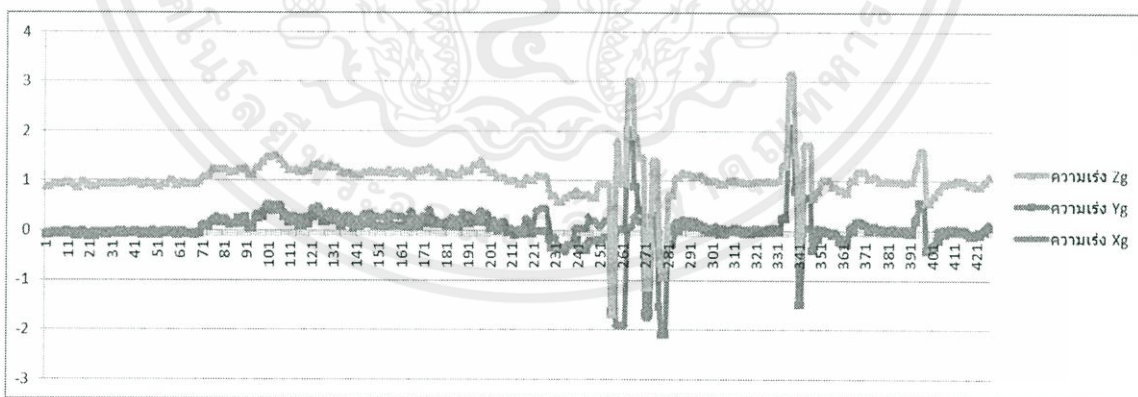
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.32 ค่าความเร็วในแนวแกน Xg ที่เกิดจากการอินทิเกรตค่าความเร่ง



รูปที่ 4.33 ระยะการเคลื่อนที่ในแนวแกน X ที่เกิดจากการอินทิเกรตค่าความเร็ว



รูปที่ 4.34 ความเร่งในแนวแกนการเคลื่อนที่ด้าน X (Xg Yg Zg)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มต้นทำการทดสอบระบบโดยการเคลื่อนที่เซนเซอร์เป็นระยะทาง 80 ซม. จากนั้นทำการเก็บข้อมูล และนำมาวาดกราฟแสดงความสัมพันธ์ระหว่างข้อมูลกับเวลา โดยโปรแกรมใช้เวลาในการอ่านค่าจากเซนเซอร์ 1 วินาที สามารถตรวจจับการเปลี่ยนแปลงของความเร่งได้ 250-300 ข้อมูล จากนั้นข้อมูลที่เซนเซอร์วัดได้จะผ่านสมการการอินทิเกรต แล้วจึงแสดงผลออกมาเป็นความเร็ว และระยะทาง นอกจากนี้ยังพบว่าความผิดพลาดในการวัดระยะทางของระบบมีความสัมพันธ์กับแรงที่กระทำทำให้เซนเซอร์เคลื่อนที่ โดยเมื่อควบคุมแรงที่กระทำ จะทำให้ระยะทางที่เซนเซอร์วัดได้มีค่าประมาณ 77.355 ซม. ซึ่งมีความผิดพลาด 3.3 %

ตารางที่ 4.4 ค่าระยะการเคลื่อนเทียบค่าจากค่าจริงกับค่าจากการทดลอง

ระยะทางที่ได้จากเครื่องมือวัด	ครั้งที่	ระยะทางที่เซนเซอร์วัดได้	ค่าเฉลี่ย	ความผิดพลาด
80 cm	1	78.95 cm	77.355 cm	3.3%
	2	76.50 cm		
	3	76.42 cm		
	4	77.55 cm		

4.3 การทดลองหาค่า ROLL PITCH

ตารางที่ 4.3.1 ผลการทดลองการเก็บค่ามุม PITCH

มุม	ครั้งที่	มุมที่เซนเซอร์วัดได้	ความผิดพลาด
90 °	1	88.52 °	1.48°
	2	89.58 °	0.48°
	3	89.23 °	0.77°
	4	88.91 °	1.09°
0	1	1.05°	1.05°
	2	1.2°	1.2°
	3	0.8°	0.8°
	4	1.7°	1.7°

ตารางที่ 4.6 ผลการทดลองการเก็บค่ามุม roll

มุม	ครั้งที่	มุมที่เซนเซอร์วัดได้	ความผิดพลาด
90 °	1	92.7 °	2.7°
	2	91 °	1°
	3	91.4°	1.4°
	4	90.5 °	0.5°
0	1	0.1°	0.1°
	2	0.13°	0.13°
	3	0.95°	0.95°
	4	0.65°	0.65°

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 บทสรุปการดำเนินงาน

การสร้างระบบระบุตำแหน่งและทิศทางเคลื่อนที่ด้วยตัววัดความเร่งและไจโรสโคป จำเป็นต้องอาศัยความรู้ และทักษะเฉพาะทางที่หลากหลาย เพื่อพัฒนาให้ระบบมีความถูกต้องเหมาะสมสำหรับการนำไปใช้งาน เนื่องจากความสามารถของเซนเซอร์ที่มีขายทั่วไปตามท้องตลาดมีจำกัด การที่ระบบจะสามารถระบุตำแหน่งการเคลื่อนที่ได้อย่างแม่นยำนั้น จึงต้องอาศัยกระบวนการในการประมวลผล และการจัดการข้อมูลที่มีประสิทธิภาพ ในการคัดกรองสัญญาณรบกวน และค่าความผิดพลาดออกจากสัญญาณที่ต้องการ ซึ่งมีเทคนิคและวิธีการที่หลากหลายแตกต่างกันไป การที่จะศึกษาให้เข้าใจจนสามารถนำมาปรับประยุกต์ใช้ได้จึงใช้ระยะเวลาานาน ซึ่งจากการทดลองพบว่า ข้อมูลที่ได้จากระบบยังมีความผิดพลาดในการวัดระยะทางอยู่มาก ส่วนมุม Roll และมุม Pitch ที่วัดได้มีความผิดพลาดอยู่ในช่วง 0 - 2 องศา นั่นคือจำเป็นต้องมีการศึกษาพัฒนาระบบต่อไป โดยการเพิ่มเซนเซอร์ หรืออุปกรณ์ชนิดอื่น ร่วมกับวิธีการกรองสัญญาณรบกวนอื่นๆ รวมทั้งเพิ่มความเร็วของโปรแกรมในการอ่าน และคำนวณค่าจากเซนเซอร์ โดยการเขียนโปรแกรมให้อยู่ในลักษณะ Interrupt หรือเปลี่ยนตัวประมวลผลข้อมูล เป็นต้น ซึ่งจะสามารถช่วยให้ข้อมูลจากระบบมีความถูกต้องแม่นยำเหมาะสมสำหรับการนำไปใช้งานมากยิ่งขึ้น

แม้ว่าการใช้อุปกรณ์ที่รวมเซนเซอร์ทั้งสองชนิดนี้แบบสำเร็จรูปที่มีขายอยู่แล้ว จะช่วยลดระยะเวลาในการดำเนินการได้ แต่ก็มีข้อจำกัดในการเข้าถึงการทำงานของเซนเซอร์ ดังนั้นจึงเลือกการสร้างระบบในการรวมข้อมูลจากตัววัดความเร่งและไจโรสโคปเอง ซึ่งมีความยืดหยุ่น ผู้ใช้สามารถเลือกเซนเซอร์แต่ละชนิดที่มีคุณสมบัติตามที่ต้องการได้ เช่น บริษัทที่ผลิต จำนวนแกนวัด ความไวของเซนเซอร์ เป็นต้น และยังสามารถพลิกแพลงประยุกต์ใช้งานได้หลากหลาย เนื่องจากสามารถเข้าถึงการทำงานของเซนเซอร์ได้โดยตรง และการออกแบบโปรแกรมเอง ทำให้ง่ายต่อการพัฒนาเพิ่มเติมในอนาคต แต่ต้องใช้เวลาในการศึกษาและพัฒนาแต่ละส่วนของระบบ จึงจะมีความถูกต้องและนำไปใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ปัญหาที่พบในการดำเนินงาน

1. เนื่องจากผู้จัดทำไม่มีความชำนาญในการเขียนโปรแกรม และการทำระบบระบุตำแหน่ง และทิศทางการเคลื่อนที่จำเป็นต้องอาศัยความรู้ และทักษะเฉพาะทางค่อนข้างหลากหลาย ทำให้ใช้เวลานานในการค้นคว้าข้อมูลและแก้ไขปัญหาต่างๆ นอกจากนี้ความสามารถของระบบยังไม่มี ความถูกต้องแม่นยำเพียงพอ
2. ความผิดพลาดของข้อมูลที่ได้จากระบบ มีสาเหตุมาจาก Sampling Time ในการอ่านค่าที่ได้จากเซนเซอร์ยังมีความเร็วไม่เพียงพอ และความสามารถของสมการคณิตศาสตร์ที่ใช้ในการลดความผิดพลาดจากการอินทิเกรตความเร่งเพื่อคำนวณระยะทางยังต้องได้รับการพัฒนาหรือปรับเปลี่ยนให้ดีขึ้น
3. ลักษณะของโปรเจกต์เมื่อได้ค่าความเร่งออกมาแล้วไม่เห็นภาพการเคลื่อนที่ที่ชัดเจนนัก ทำให้เกิดความสับสนในการออกแบบการทดลองและคำนวณค่าระยะทางการเคลื่อนที่
4. การจำลองการทดลองเพื่อหาค่าระยะทางและแขนงน้ำหนักให้เกิดการเคลื่อนที่คงที่ ทำให้เกิดความคลาดเคลื่อนจากการที่วัตถุเคลื่อนที่เร็วมาก จนไม่สามารถเก็บค่าที่แสดงผลจริงได้ อาจเป็นเพียงค่าประมาณ
5. สภาพสถานที่ต่างๆ อาจมีค่าความคาดเคลื่อนอยู่ เช่น ระนาบของโต๊ะที่ไม่เรียบ ความสั่นสะเทือนก็สามารถทำให้ค่าที่วัดผลจากไจโรสโคปเปลี่ยนแปลง
6. สถานที่ดำเนินงานไม่พร้อมต่อการปฏิบัติงานทำให้งานล่าช้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางในการพัฒนาโครงการงาน

1. ออกแบบโปรแกรมให้ทำงานในลักษณะ Interrupt เพื่อลดเวลาที่โปรแกรมใช้ในการคำนวณ ศึกษาวิธีการลดความผิดพลาดอันเกิดจากสัญญาณรบกวน ความผิดพลาดที่เกิดจากการอินทิเกรตที่มีประสิทธิภาพมากขึ้น เช่น การเพิ่มเซนเซอร์ชนิดอื่นๆ การใช้สมการคณิตศาสตร์ที่มีความซับซ้อนมากขึ้น
2. ใช้โปรแกรม PROCESSING เข้ามาช่วยในการมองเห็นภาพการเคลื่อนที่ของเซนเซอร์และค่าที่ออกมาได้มากขึ้น
3. ปรับเปลี่ยนค่าน้ำหนักที่ใช้ถ่วงวัสดุ และหาอุปกรณ์ช่วยลดแรงกระแทกเมื่อวัตถุเคลื่อนที่มายังตำแหน่งที่ต้องการ เนื่องจากเมื่อวัตถุหยุดนิ่งเกิดจากตั้งกลับเล็กน้อยทำให้ค่าแกว่ง
4. มีการนำ SERVO เข้ามาช่วยในการทดลองหาค่า PITCH and ROLL เพื่อให้อุปกรณ์หมุนตามระยะและองศาที่ต้องการเพื่อดูค่าความผิดพลาด
5. เนื่องจากเรื่องที่ศึกษาเฉพาะตัวเซนเซอร์ไจโรสโคปเพียงตัวเดียว หากนำไปพัฒนาต่ออาจเสริม เอนโคเดอร์ หรือตัววัดค่าความเร่งชนิดอื่นมาช่วยให้เกิดความผิดพลาดน้อยที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] <http://www.instructables.com/id/Accelerometer-Gyro-Tutorial/>
- [2] <http://www.arduitronics.com/article/arduino-and-motor-control-part-4-servo-motor>
- [3] <http://playground.arduino.cc/code/timer1>
- [4] <http://www.arduino.cc/en/Main/Products>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

โปรแกรมอ่านค่า Sensor

Accelerometer

```

#include <TimerOne.h>
#include <Wire.h>
#include <ADXL345.h>
// #include <Servo.h>
// Servo servo1;
/*
*/
const float alpha = 0.5;
double fXg = 0;
double fYg = 0;
double fZg = 0;
double pitch, roll, Xg, Yg, Zg;
ADXL345 acc;
void setup()
{
    acc.begin();
    Serial.begin(115200);
    delay(100);
    Timer1.initialize(4000); //us
    Timer1.attachInterrupt( timerIsr );
}
void loop()
{
    /*servo1.attach(9);
    servo1.write(90);
    delay(1000);
    servo1.write(0);
    delay(1000);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

servo1.detach();*/
    acc.read(&Xg, &Yg, &Zg);
    //Low Pass Filter
    fXg = Xg * alpha + (fXg * (1.0 - alpha));
    fYg = Yg * alpha + (fYg * (1.0 - alpha));
    fZg = Zg * alpha + (fZg * (1.0 - alpha));
    //Roll & Pitch Equations
    roll = (atan2(-fYg, fZg)*180.0)/M_PI;
    pitch = (atan2(fXg, sqrt(fYg*fYg + fZg*fZg))*180.0)/M_PI;
}
void timerIsr()
{
    // Toggle LED
    digitalWrite( 13, digitalRead( 13 ) ^ 1 );
    unsigned long time=millis();
    Serial.print(time);
    Serial.print("\t");
    Serial.print(pitch);
    Serial.print("\t");
    Serial.print(roll);
    Serial.print("\t");
    Serial.print(Xg);
    Serial.print("\t");
    Serial.print(Yg);
    Serial.print("\t");
    Serial.println(Zg);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Gyroscope

```
#include <Wire.h>
```

```
#include <ITG3205.h>
```

```
ITG3205 itg3205;
```

```
void setup()
```

```
{
```

```
  Wire.begin();
```

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

```
  delay(100);
```

```
  itg3205.itg3205initGyro();
```

```
  delay(100);
```

```
  itg3205.itg3205CalGyro();
```

```
  delay(100);
```

```
}
```

```
void loop()
```

```
{
```

```
  itg3205.itg3205ReadGyro();
```

```
  Serial.print("Gyro(degree/s): ");
```

```
  Serial.print(itg3205.itg3205GyroX()); // Data to Degree conversation
```

```
  Serial.print(", ");
```

```
  Serial.print(itg3205.itg3205GyroY());
```

```
  Serial.print(", ");
```

```
  Serial.println(itg3205.itg3205GyroZ());
```

```
  Serial.print("Temperature: ");
```

```
  Serial.println(itg3205.itg3205Temp());
```

```
  delay(100);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Servo moter

```

#include <Servo.h>
Servo myservo
int pos = 0;
void setup()
{
    myservo.attach(9);
    Serial.begin(9600);
}
void loop()
{
    //for(pos = 0; pos <= 180; pos += 1) // goes from 0 degrees to 180 degrees
    //{
        // in steps of 1 degree
        myservo.write(48);
        Serial.print(millis());
        Serial.print("\t"); // waits 15ms for the servo to reach the position
        Serial.println(myservo.read());
    //}
    //for(pos = 180; pos>=0; pos-=1) // goes from 180 degrees to 0 degrees
    //{
        delay(2000);
        myservo.write(146); // tell servo to go to position in variable 'pos'
        delay(2000); // waits 15ms for the servo to reach the position
    //}
    //myservo.detach();
}
// servo.read()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเร็วของมอเตอร์

```

#include <Wire.h>
#include <ADXL345.h>

const float alpha = 0.5;
double fXg = 0;
double fYg = 0;
double fZg = 0;
double pitch, roll, Xg, Yg, Zg;

ADXL345 acc;

void setup()
{
  acc.begin();
  Serial.begin(115200);
  Serial.println("Connect");
  delay(1000);
  Serial.println("Start");
}

void loop()
{
  unsigned long time = millis(); // จับเวลาก่อนอ่าน
  acc.read(&Xg, &Yg, &Zg);

  //Low Pass Filter
  fXg = Xg * alpha + (fXg * (1.0 - alpha));
  fYg = Yg * alpha + (fYg * (1.0 - alpha));
  fZg = Zg * alpha + (fZg * (1.0 - alpha));

  //Roll & Pitch Equations
  roll = (atan2(-fYg, fZg) * 180.0) / M_PI;
  pitch = (atan2(fXg, sqrt(fYg * fYg + fZg * fZg)) * 180.0) / M_PI;
  unsigned long t_read = millis() - time; // // เวลาปัจจุบัน - ก่อนอ่าน
  float f = 1 / t_read; // หาก f จาก f=1/t
  Serial.print("t(ms)=");

```

เอกสารนี้เป็นทรัพย์สินของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดก็ตาม หากมีข้อผิดพลาดประการใด ขออภัยและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Serial.print(t_read);  
Serial.print("\tf(Hz)=");  
Serial.println(f);  
}
```

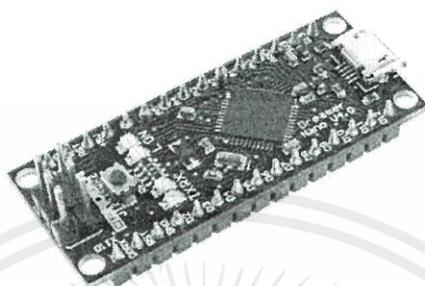


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

เอกสารคู่มืออุปกรณ์อิเล็กทรอนิกส์

Microcontroller arduino nano v.4



The Dreamer Nano V4.0 is a surface mount breadboard embedded version of the ATMEGA 32U4 with integrated Micro USB. It has everything that Leonardo has (electrically). Physically, it is just missing power jack to save space.

Using the ATmega32U4 as its sole microcontroller allows it to be cheaper and simpler. Also, because the 32U4 is handling the USB directly, code libraries are available which allow the board to emulate a computer keyboard, mouse, and more using the USB-HID protocol!

The Dreamer Nano v4.0 is compatible with most Nano shield in the market. It's suitable for projects that require a compact size controller system.

Specification

- Microcontroller: ATmega32u4
- Operating Voltage: 5V
- Input Voltage (recommended): 6.5-12v (VIN) / 5v (Micro USB)
- Input Voltage (limits): 6-12V
- Digital I/O Pins: 20
- PWM Channels: 7
- Analog Input Channels: 12
- DC Current per I/O Pin: 40 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB (ATmega32u4) of which 4 KB used by bootloader
- SRAM: 2.5 KB (ATmega32u4)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในห้องเรียนเพื่อการศึกษาดูเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆก็ตาม หากมีข้อสงสัยหรือต้องการข้อมูลเพิ่มเติม กรุณาติดต่ออาจารย์ผู้สอนหรือเจ้าหน้าที่ที่เกี่ยวข้อง

- EEPROM:1 KB (ATmega32u4)
- Clock Speed:16 MHz
- Compatible with most of Nano shield
- Size: 45x20x20mm

Input and Output

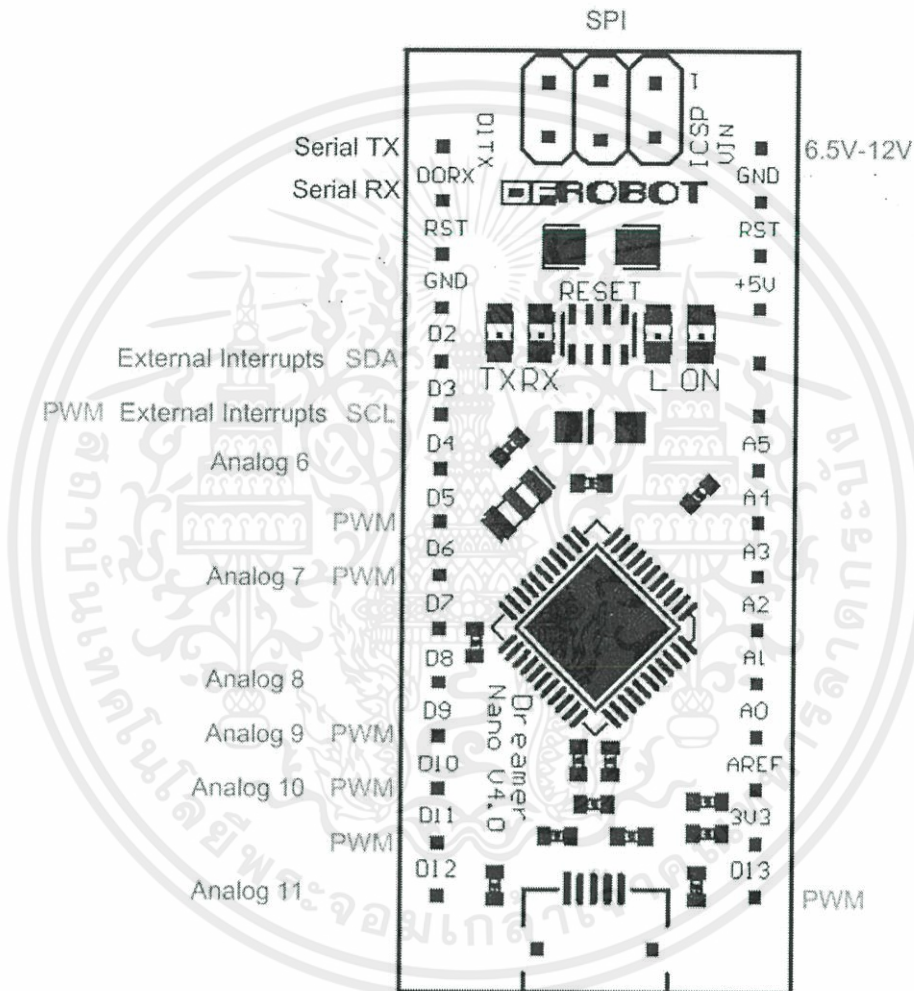
Each of the 20 digital i/o pins on the Uno can be used as an input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data using the ATmega32U4 hardware serial capability. Note that on the Leonardo, the `Serial` class refers to USB (CDC) communication; for TTL serial on pins 0 and 1, use the `Serial1` class.
- TWI: 2 (SDA) and 3 (SCL). Support TWI communication using the `Wire` library.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the `attachInterrupt()` function for details.
- PWM: 3, 5, 6, 9, 10, 11, and 13. Provide 8-bit PWM output with the `analogWrite()` function.
- SPI: on the ICSP header. These pins support SPI communication using the `SPI` library. Note that the SPI pins are not connected to any of the digital I/O pins as they are on the Uno, They are only available on the ICSP connector. This means that if you have a shield that uses SPI, but does NOT have a 6-pin ICSP connector that connects to the Leonardo's 6-pin ICSP header, the shield will not work.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- Analog Inputs: A0-A5, A6 - A11 (on digital pins 4, 6, 8, 9, 10, and 12). The Leonardo has 12 analog inputs, labeled A0 through A11, all of which can also be used as digital i/o. Pins A0-A5 appear in the same locations as on the Uno; inputs A6-A11 are on digital i/o pins 4, 6, 8, 9, 10, and 12 respectively. Each analog input provide 10 bits of resolution (i.e. 1024 different values). By default the analog

inputs measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function.

There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with `analogReference()`.
- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.



Programming

The Dreamer Nano can be programmed with the official Arduino software (download). Select "Arduino Leonardo" from the Tools > Board menu (according to the microcontroller on your board).

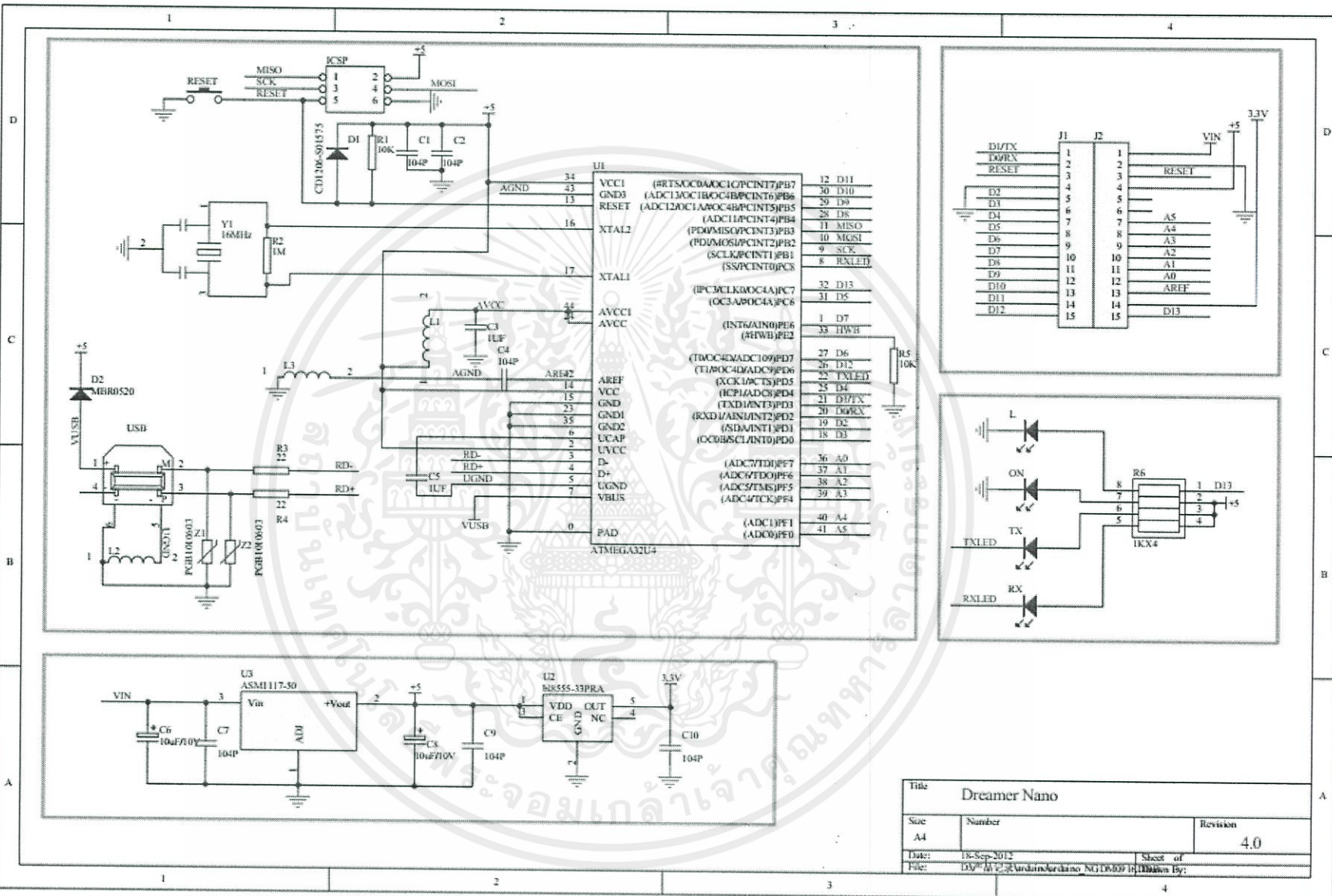
The ATmega32U4 on the Dreamer Nano comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

programmer. It communicates using the AVR109 protocol. You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title		
Dreamer Nano		
Size	Number	Revision
A4		4.0
Date:	18-Sep-2012	Sheet of
File:	D:\P\AV2\Arduino\Arduino_NG\DM0918	Drawn By:

ITG3200: Triple-Axis Digital-Output Gyro Breakout

Features

The ITG-3200 triple-axis MEMS gyroscope includes a wide range of features:

- Digital-output X-, Y-, and Z-Axis angular rate sensors (gyros) on one integrated circuit with a sensitivity of 14.375 LSBs per °/sec and a full-scale range of $\pm 2000^\circ/\text{sec}$
- Three integrated 16-bit ADCs provide simultaneous sampling of gyros while requiring no external multiplexer
- Enhanced bias and sensitivity temperature stability reduces the need for user calibration
- Low frequency noise lower than previous generation devices, simplifying application development and making for more-responsive motion processing
- Digitally-programmable low-pass filter
- Low 6.5mA operating current consumption for long battery life
- Wide VDD supply voltage range of 2.1V to 3.6V
- Flexible VLOGIC reference voltage allows for I2C interface voltages from 1.71V to VDD
- Standby current: 5 μA
- Smallest and thinnest package for portable devices (4x4x0.9mm QFN)
- No high pass filter needed
- Turn on time: 50ms
- Digital-output temperature sensor
- Factory calibrated scale factor
- 10,000 g shock tolerant
- Fast Mode I2C (400kHz) serial interface
- On-chip timing generator clock frequency is accurate to +/-2% over full temperature range
- Optional external clock inputs of 32.768kHz or 19.2MHz to synchronize with system clock
- MEMS structure hermetically sealed and bonded at wafer level
- RoHS and Green compliant

Applications

- Motion-enabled game controllers

- Motion-based portable gaming
- Motion-based 3D mice and 3D remote controls
- “No Touch” UI
- Health and sports monitoring

ADXL345

Description: This new version adds 2 standoff holes as well as an extra decoupling capacitor. The ADXL345 is a small, thin, low power, 3-axis MEMS accelerometer with high resolution (13-bit) measurement at up to +16 g. Digital output data is formatted as 16-bit twos complement and is accessible through either a SPI (3- or 4-wire) or I2C digital interface.

The ADXL345 is well suited to measure the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion or shock. Its high resolution (4 mg/LSB) enables measurement of inclination changes less than 1.0 degrees.

Several special sensing functions are provided. Activity and inactivity sensing detect the presence or lack of motion and if the acceleration on any axis exceeds a user-set level. Tap sensing detects single and double taps. Free-fall sensing detects if the device is falling. These functions can be mapped to one of two interrupt output pins. An integrated, patent pending 32-level first in, first out (FIFO) buffer can be used to store data to minimize host processor intervention. Low power modes enable intelligent motion-based power management with threshold sensing and active acceleration measurement at extremely low power dissipation.

Features and Benefits

- Ultralow power: as low as 40 μA in measurement mode and 0.1 μA in standby mode at $V_S = 2.5\text{ V}$ (typical)
- Power consumption scales automatically with bandwidth
- User-selectable resolution
- Fixed 10-bit resolution

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Full resolution, where resolution increases with g range, up to 13-bit resolution at ± 16 g (maintaining 4 mg/LSB scale factor in all g ranges)
- Embedded, patent pending FIFO technology minimizes host processor load
- Tap/double tap detection
- Activity/inactivity monitoring
- Please see data sheet for additional features.

ADXLS345-EP SUPPORTS DEFENSE AND AREOSPACE APPLICATIONS (AQEC)

- Temperature range:
-55°C to +105°C
- Controlled manufacturing baseline
- One assembly/test site
- One fabrication site
- Enhanced product change notification
- Qualification data available on request

HMC5883L Triple Axis Magnetometer Breakout

This is a breakout board for The Honeywell HMC5883L . The Honeywell HMC5883L is a multi-chip module designed for low-field magnetic sensing with a digital interface for applications such as low-cost compassing and magnetometry. The HMC5883L includes our state-of-the-art, high-resolution HMC118X series magneto-resistive sensors plus an ASIC containing amplification, automatic degaussing strap drivers, offset cancellation, and a 12-bit ADC that enables 1° to 2° compass heading accuracy. The I2C serial bus allows for easy interface. Applications for the HMC5883L include Mobile Phones, Netbooks, Consumer Electronics, Auto Navigation Systems, and Personal Navigation Devices.

The HMC5883L utilizes Honeywell's Anisotropic Magneto-resistive (AMR) technology that provides advantages over other magnetic sensor technologies. These anisotropic, directional sensors feature precision in-axis sensitivity and linearity. These sensors' solid-state construction with very low cross-axis sensitivity is designed to measure both the direction and the magnitude of Earth's magnetic fields, from milli-gauss to 8

gauss. Honeywell's Magnetic Sensors are among the most sensitive and reliable low-field sensors in the industry.

Applications

Mobile Phones, Netbooks

Consumer Electronics

Auto Navigation Systems

Personal Navigation Devices

Features

The Honeywell HMC5883L includes a wide range of features:

3-Axis Magnetoresistive Sensors and ASIC in a 3.0x3.0x0.9mm LCC Surface Mount Package

12-Bit ADC Coupled with Low Noise AMR Sensors Achieves 2 milli-gauss Field Resolution in ± 8 Gauss Fields

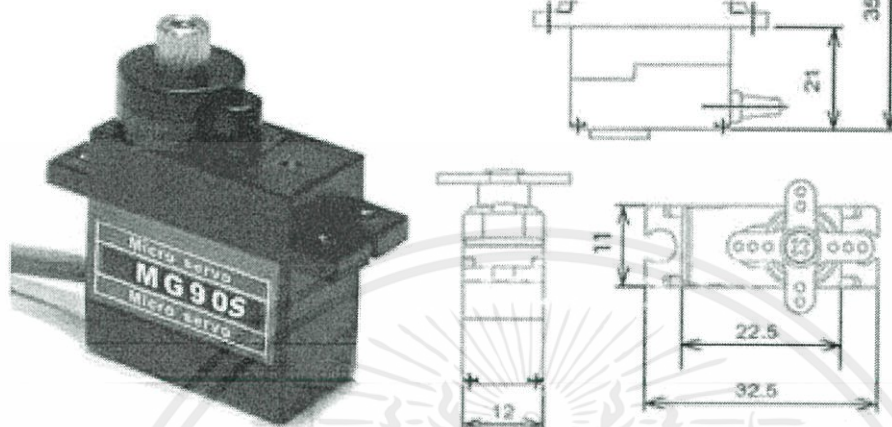
Low Voltage Operations (2.16 to 3.6V) and Low Power Consumption (100 μ A) I2C Digital Interface

Wide Magnetic Field Range (± 8 Oe)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MG90S

Metal Gear Servo



MG90S servo, Metal gear with one bearing

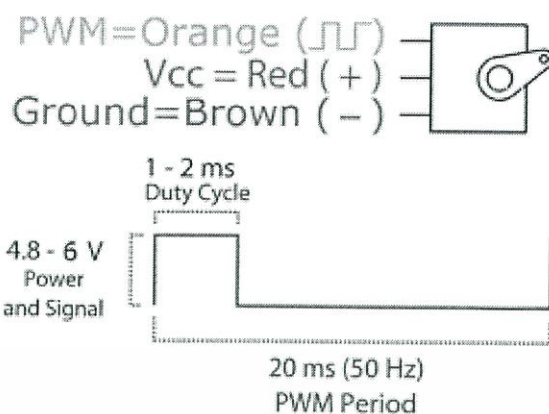
Tiny and lightweight with high output power, this tiny servo is perfect for RC Airplane, Helicopter, Quadcopter or Robot. This servo has *metal gears* for added strength and durability.

Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but *smaller*. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.

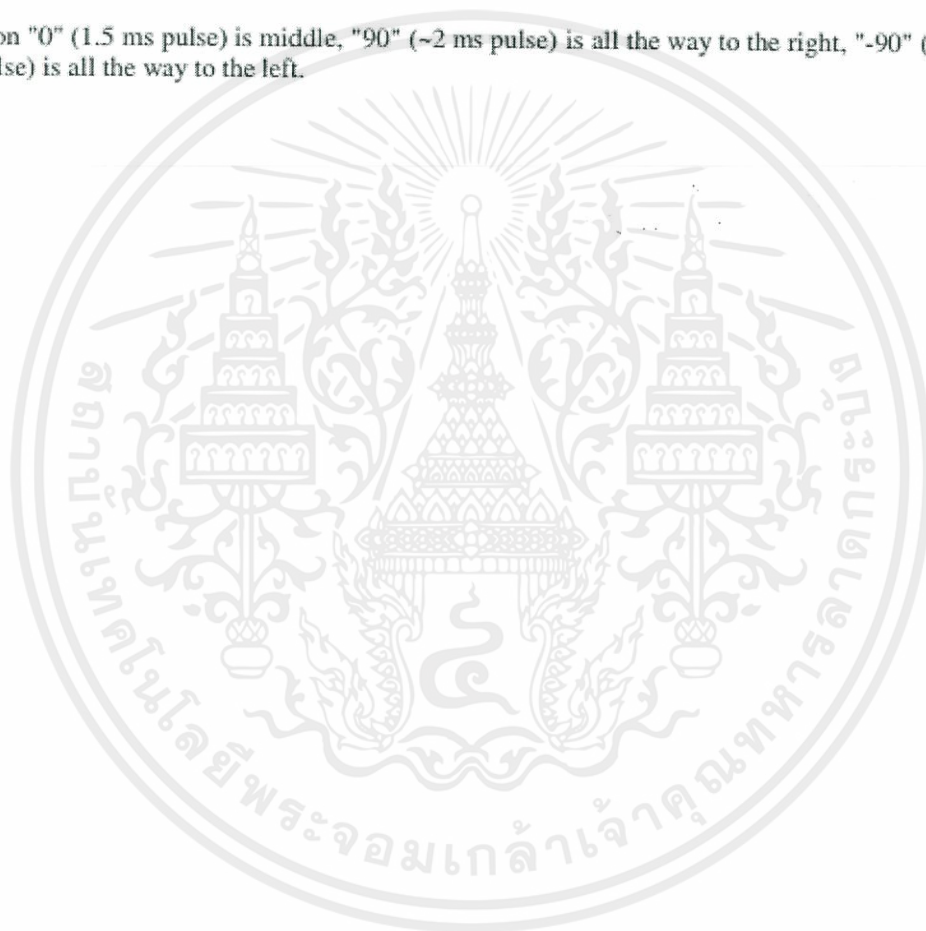
Specifications

- Weight: 13.4 g
- Dimension: 22.5 x 12 x 35.5 mm approx.
- Stall torque: 1.8 kgf·cm (4.8V), 2.2 kgf·cm (6 V)
- Operating speed: 0.1 s/60 degree (4.8 V), 0.08 s/60 degree (6 V)
- Operating voltage: 4.8 V - 6.0 V
- Dead band width: 5 μ s

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Position "0" (1.5 ms pulse) is middle, "90" (~2 ms pulse) is all the way to the right, "-90" (~1 ms pulse) is all the way to the left.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้