

ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี
BEAM BALANCING SYSTEM USING PID CONTROLLER



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2562

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BEAM BALANCING SYSTEM USING PID CONTROLLER



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

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2019

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี

โดย

นายพุทธรักษ์ นามสุขี 59010995

นายภพกร บุญจิตร 59011018

อาจารย์ที่ปรึกษา

ศาสตราจารย์ ดร.วันชัย รุ่งรุจา

รองศาสตราจารย์ ดร.วรรณดี เพชรมณีล้ำค่า

ปีการศึกษา 2652

บทคัดย่อ

ปฏิญานิพนธ์ฉบับนี้มีวัตถุประสงค์เพื่อออกแบบและสร้างระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี การเคลื่อนที่ของคานสมดุลอาศัยแรงยกของใบพัดที่ถูกควบคุมด้วยความเร็วของมอเตอร์ไร้แปรงถ่าน การประมวลผลการทำงานของระบบทั้งหมดใช้ไมโครคอนโทรลเลอร์ Arduino Uno ระบบคานสมดุลสามารถกำหนดตำแหน่งมุมเอียงของคานได้ โดยการป้อนคำสั่งจากภายนอก ยิ่งไปกว่านั้นพารามิเตอร์ของตัวควบคุมแบบพีไอดีสามารถปรับค่าได้โดยการปรับจูนปุ่มที่แผงควบคุมภายนอก ระบบที่ออกแบบนี้สามารถทำงานได้ตามความต้องการ นอกจากนี้ระบบยังเหมาะแก่การใช้ในการศึกษาการทำงานของระบบควบคุมพีไอดี รวมถึงการพัฒนาโดรนในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต่ออ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BEAM BALANCING SYSTEM USING PID CONTROLLER

By

Mr.Phuttarak Namsukhee 59010995

Mr.Phopprakorn Boonjit 59011018

Advisors

Prof.Dr.Vanchai Riewruja

Assoc.Prof.Dr.Wandee Petchmaneelumka

Academic Year 2019

ABSTRACT

The purpose of this thesis is to design and implement a beam balancing system using PID controller. Lifting force of propeller controlled speed by brushless motor is useful for control the balance beam. Microcontroller, Arduino Uno, is used for processing the whole operation of the system. The proposed system can set the desired angle of the balanced beam by applying the external command. Moreover, the parameters of PID controller can be adjusted by tuning the buttons on the front panel. The system designed can operate as needed. In addition, this system is suitable for study of the PID control system including further development of drone.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต่อ||อ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้ประสบผลสำเร็จไปได้ด้วยดี ทั้งนี้เนื่องจากได้รับความช่วยเหลือและคำแนะนำของอาจารย์ที่ปรึกษา รวมถึงคณาจารย์ในหลักสูตรวิศวกรรมระบบควบคุม ภาควิชาวิศวกรรมการวัดและควบคุม คณะผู้จัดทำขอขอบพระคุณในความกรุณาจากอาจารย์ทุกท่านที่คอยช่วยเหลือในการทำโครงการครั้งนี้ และขอขอบคุณเพื่อนๆ ที่คอยให้คำปรึกษาและคำแนะนำ จึงทำให้โครงการนี้สำเร็จลุล่วงไปได้ด้วยดี ดังนั้น คณะผู้จัดทำจึงขอขอบพระคุณในความช่วยเหลือของทุกๆ ท่าน มา ณ ที่นี้

สุดท้ายนี้ขอกราบขอบพระคุณบิดา มารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้และคอยให้ความกรุณามาโดยตลอด

คณะผู้จัดทำ
พุทธรักษ์ นามสุขี
ภพกร บุญจิตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และตัดอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VII
สารบัญตาราง	X
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์	1
1.3 ขอบเขตของโครงการ	2
1.4 ขั้นตอนการดำเนินงาน	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
1.6 รายละเอียดของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	4
2.1 หลักการควบคุมแบบ PID	4
2.2 แรงยก	6
2.3 หลักการทำงานของ Complementary filter	8
2.4 โมดูล GY-521	9
2.4.1 Accelerometer	9
2.4.2 Gyroscope	9
2.5 มอเตอร์ไร้แปรงถ่าน	9
2.6 โมดูลแสดงผล LCD	10
2.6.1 Graphic LCD	10
2.6.2 Character LCD	10
2.6.2.1 การเชื่อมต่อแบบขนาน	10

สารบัญ(ต่อ)

	หน้า
2.6.2.2 การเชื่อมต่อแบบอนุกรม	10
2.7 เพาเวอร์ซัพพลาย	11
2.8 Arduino UNO	11
2.9 โปรแกรม Arduino IDE	11
บทที่ 3 วิธีการดำเนินงาน	12
3.1 ขั้นตอนการดำเนินงาน	12
3.2 อุปกรณ์ที่เกี่ยวข้อง	12
3.2.1 โมดูล GY-521	12
3.2.2 Electronic speed controller	13
3.2.3 มอเตอร์ไร้แปรงถ่าน	13
3.2.4 เพาเวอร์ซัพพลาย	14
3.2.5 โมดูลแสดงผล LCD	14
3.2.6 บอร์ด Arduino UNO	15
3.3 ซอฟต์แวร์ที่เกี่ยวข้อง	15
3.3.1 โปรแกรม Arduino IDE	15
3.4 การออกแบบและการวางแผนการทำงาน	16
3.4.1 การออกแบบและการวางแผนทางด้านฮาร์ดแวร์	16
3.4.2 การออกแบบและการวางแผนทางด้านซอฟต์แวร์	16
3.5 วิธีการดำเนินงาน	16
3.5.1 ออกแบบและสร้างโครงสร้างของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี	16
3.5.2 การทดสอบเซนเซอร์	17
3.5.2.1 โปรแกรมที่ใช้ในการทดสอบ โมดูล GY-521	17
3.5.3 การทดสอบมอเตอร์และ Electronic speed controller	18
3.5.3.1 โปรแกรมที่ใช้ทดสอบชุดการทำงานของมอเตอร์	18
3.5.4 เขียนโปรแกรมที่ใช้ในการทำงานของระบบ	19
3.5.5 ประกอบกล่องควบคุมสำหรับระบบ	23
3.5.6 การติดตั้งอุปกรณ์ที่ใช้ในการทำงาน	23

สารบัญ(ต่อ)

	หน้า
บทที่ 4 ผลการทดลอง	25
4.1 ผลการทดสอบอุปกรณ์	25
4.1.1 ผลการทดสอบการทำงานของโมดูล GY-521	25
4.1.2 ผลการทดสอบการทำงานของชุดมอเตอร์และ Electronic speed controller	33
4.2 ผลการทดสอบการทำงานของระบบ	34
4.2.1 ผลการทดสอบการทำงานของระบบด้วยการปรับค่า Kp	34
4.2.2 ผลการทดสอบการทำงานของระบบด้วยการปรับค่า Kd	36
4.2.3 ผลการทดสอบการทำงานของระบบด้วยการปรับค่า Ki	38
4.2.1 ผลการทดสอบการทำงานด้วยการปรับค่าเป้าหมาย	39
4.3 ผลการประดิษฐ์ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี	42
บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ	43
5.1 สรุปผลการดำเนินงาน	43
5.2 ปัญหาและอุปสรรคในการดำเนินงาน	43
5.3 แนวทางการแก้ไขปัญหา	44
5.4 ข้อเสนอแนะ	44
เอกสารอ้างอิง	45
ภาคผนวก	46
ภาคผนวก ก โมดูล GY-521	47
ภาคผนวก ข Electronic speed controller	48
ภาคผนวก ค มอเตอร์ไร้แปรงถ่าน	49
ภาคผนวก ง จอแสดงผล LCD	51
ภาคผนวก จ Arduino UNO	53
ภาคผนวก ฉ โปรแกรมการทำงานของระบบคานสมดุล	54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 แผนภาพบล็อกไดอะแกรมของการควบคุมแบบ พีไอดี	4
2.2 แรงกระทำที่เกิดขึ้นในภาคตัดขวางของปีก (1)	7
2.3 แรงกระทำที่เกิดขึ้นในภาคตัดขวางของปีก (2)	7
2.4 หลักการทำงานของ Complementary filter	8
3.1 โมดูล GY-521	12
3.2 Electronic speed controller	13
3.3 มอเตอร์ไร้แปรงถ่าน	13
3.4 เพาเวอร์ซัพพลาย	14
3.5 อุปกรณ์แสดงผล LCD	14
3.6 อุปกรณ์ Arduino UNO	15
3.7 โปรแกรม Arduino IDE	15
3.8 โครงสร้างของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี (1)	16
3.9 โครงสร้างของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี (2)	17
3.10 โปรแกรมที่ใช้ในการทดสอบ โมดูล GY-521	17
3.11 การทดสอบโมดูล GY-521	18
3.12 โปรแกรมที่ใช้ในการทดสอบการทำงานของมอเตอร์และ Electronic speed controller	18
3.13 ทดสอบการทำงานและประสิทธิภาพของ มอเตอร์	19
3.14 หน้าต่างโปรแกรม Arduino IDE (1)	19
3.15 หน้าต่างโปรแกรม Arduino IDE (2)	20
3.16 หน้าต่างโปรแกรม Arduino IDE (3)	20
3.17 หน้าต่างโปรแกรม Arduino IDE (4)	21
3.18 หน้าต่างโปรแกรม Arduino IDE (5)	21
3.19 หน้าต่างโปรแกรม Arduino IDE (6)	22
3.20 หน้าต่างโปรแกรม Arduino IDE (7)	22
3.21 ลักษณะภายนอกของกล่องควบคุม	23
3.22 ลักษณะภายในของกล่องควบคุม	23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และตัด viii อ่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.23 อุปกรณ์ที่ใช้ในระบบการทำงาน (1)	24
3.24 อุปกรณ์ที่ใช้ในระบบการทำงาน (2)	24
4.1 ข้อมูลจาก โมดูล GY-521	25
4.2 มุมของคานขณะ 0 องศา	26
4.3 มุมที่ได้จากเซนเซอร์ขณะ 0 องศา	26
4.4 มุมของคานขณะ -10 องศา	27
4.5 มุมที่ได้จากเซนเซอร์ขณะ -10 องศา	27
4.6 มุมของคานขณะ -20 องศา	28
4.7 มุมที่ได้จากเซนเซอร์ขณะ -20 องศา	28
4.8 มุมของคานขณะ -30 องศา	29
4.9 มุมที่ได้จากเซนเซอร์ขณะ -30 องศา	29
4.10 มุมของคานขณะ 10 องศา	30
4.11 มุมที่ได้จากเซนเซอร์ขณะ 10 องศา	30
4.12 มุมของคานขณะ 20 องศา	31
4.13 มุมที่ได้จากเซนเซอร์ขณะ 20 องศา	31
4.14 มุมของคานขณะ 30 องศา	32
4.15 มุมที่ได้จากเซนเซอร์ขณะ 30 องศา	32
4.16 กราฟแสดงการทำงานของระบบเมื่อเปิดอุปกรณ์	34
4.17 วิธีการในการปรับค่า K_p ให้กับระบบ	34
4.18 กราฟแสดงการทำงานของระบบเมื่อปรับค่า $K_p = 0.5$	35
4.19 กราฟแสดงการทำงานของระบบเมื่อปรับค่า $K_p = 1.0$	35
4.20 วิธีการในการปรับค่า K_d ให้กับระบบ	36
4.21 กราฟแสดงการทำงานของระบบเมื่อปรับค่า $K_d = 0.5$	36
4.22 กราฟแสดงการทำงานของระบบเมื่อปรับค่า $K_d = 1.0$	37
4.23 กราฟแสดงการทำงานของระบบเมื่อเกิดค่าพุ่งเกิน	37
4.24 วิธีการในการปรับค่า K_i ให้กับระบบ	38

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.25 กราฟแสดงระบบการทำงานของอุปกรณ์ขณะเกิดออฟเซต	38
4.26 กราฟแสดงการทำงานของระบบเมื่อปรับค่า $K_i = 0.3$	39
4.27 กราฟแสดงการทำงานของระบบเมื่อค่าเป้าหมาย เปลี่ยนจาก 0 องศา ถึง -15 องศา	39
4.28 กราฟแสดงการทำงานของระบบเมื่อค่าเป้าหมาย เปลี่ยนจาก -15 องศาถึง 0 องศา และ 0 องศา ถึง 15 องศา	40
4.29 กราฟแสดงการทำงานของระบบเมื่อค่าเป้าหมาย 0 องศา	40
4.30 กราฟแสดงการทำงานของระบบเมื่อค่าเป้าหมาย -15 องศา	41
4.31 ลักษณะของอุปกรณ์เมื่อ set point เท่ากับ -15 องศา	41
4.32 จอแสดงผลขณะกำหนดค่าเป้าหมาย	41
4.33 ระบบควบคุมโดยใช้ตัวควบคุมพีไอดี	42
4.34 หน้าจอแสดงผลค่าพารามิเตอร์ที่ใช้ควบคุมระบบ	42

สารบัญตาราง

ตารางที่	หน้า
1. ตารางสรุปผลค่าความคลาดเคลื่อนที่เกิดจากโมดูล GY-521	33
2. ประสิทธิภาพการทำงานของมอเตอร์และ Electronic speed controller	33



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญญาประดิษฐ์

ในปัจจุบัน ระบบควบคุมได้เข้ามามีบทบาทในการใช้ชีวิตของมนุษย์เป็นอย่างมาก และยังเป็นระบบการทำงานที่มีแนวโน้มที่จะพัฒนาและเติบโตเพิ่มมากขึ้นเรื่อยๆ มีหน้าที่หลักเพื่ออำนวยความสะดวกสบายในการดำเนินชีวิตแก่มนุษย์ ไม่ว่าจะเป็นระบบตั้งเวลาในเครื่องไมโครเวฟ การใช้รีโมทคอนโทรลสำหรับเปิด-ปิดรถยนต์ หรือการควบคุมอุณหภูมิในเตาอบ เป็นต้น และหากพิจารณาถึงประโยชน์ของเครื่องจักรที่ใช้การทำงานแบบระบบอัตโนมัติจะเห็นได้ว่าเครื่องจักรอัตโนมัติสามารถควบคุมและช่วยลดความเสียหายจากการทำงานโดยแรงงานมนุษย์ มีการทำงานที่สะดวก ง่ายตาย และสามารถผลิตสินค้าที่ได้มาตรฐาน รวมถึงระบบควบคุมอีกหลายประเภท ที่นำมาใช้งานเช่น ระบบควบคุมแบบฟัซซี (Fuzzy) แบบพีไอดี (PID) หรือ แบบโครงข่ายประสาทเทียม (Neural network) จากที่กล่าวมาทำให้ผู้จัดทำมีความสนใจในระบบควบคุมโดยใช้ตัวควบคุมแบบพีไอดี ที่สามารถศึกษาและทำความเข้าใจกับกระบวนการทำงานของระบบควบคุมด้วยตัวควบคุมแบบ PID ได้ โดยเหตุผลที่เลือกใช้คานเนื่องมาจาก คานถูกใช้ในการศึกษาระบบควบคุมมานาน เป็นระบบไม่เชิงเส้น (nonlinear) ไม่มีเสถียรภาพวงเปิด (open-loop unstable) และไม่ซับซ้อน ทำให้ง่ายต่อการทำความเข้าใจและนำไปสู่การศึกษาระบบอื่นๆ ที่ซับซ้อนมากขึ้น สำหรับปัญญาประดิษฐ์ฉบับนี้ผู้จัดทำได้สร้างระบบคานสมดุลโดยใช้ใบพัดเป็นตัวสร้างแรงยกที่ปลายของคานทั้ง 2 ข้าง และใช้ตัวควบคุมแบบ PID (proportional-integral-derivative controller) ในการควบคุมระบบ ซึ่งระบบที่สร้างขึ้นมานี้เป็นระบบที่เข้าใจง่ายเหมาะแก่การนำไปศึกษาการทำงานของระบบควบคุมแบบ PID นอกจากนี้ยังเหมาะแก่การนำไปพัฒนาต่อยอดไดรอนอีกด้วย โดยระบบคานสมดุลที่สร้างขึ้นสามารถป้อนค่าเป้าหมายหรือค่ามุมเอียงของคานได้จากการปรับค่าจากแผงควบคุม นอกจากนี้ยังสามารถปรับแต่งค่าพารามิเตอร์ของตัวควบคุมแบบ PID ได้จากแผงควบคุมอีกด้วย

1.2 วัตถุประสงค์ของปัญญาประดิษฐ์

วิทยานิพนธ์ฉบับนี้มีวัตถุประสงค์เพื่อ ออกแบบ สร้าง และทดสอบระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี เพื่อศึกษากระบวนการทำงานของระบบควบคุมพีไอดี และสามารถควบคุมการทำงานได้ง่าย และสามารถนำไปปรับใช้กับระบบการทำงานแบบต่างๆ ได้

1.3 ขอบเขตของโครงการ

1. ศึกษารูปแบบการทำงานของระบบคานสมดุลที่ใช้ในการควบคุมตำแหน่งในแนวดิ่ง
2. ออกแบบและสร้างโครงสร้างของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี
3. ทดสอบการทำงานของชุดอุปกรณ์ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาค้นคว้าทฤษฎีและเอกสารที่เกี่ยวข้อง
2. ศึกษาหลักการการทำงานของไมโครคอนโทรลเลอร์และเซนเซอร์
3. ออกแบบโครงสร้างระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี
4. สั่งซื้ออุปกรณ์ที่ใช้ประกอบโครงสร้าง
5. ทดสอบอุปกรณ์ที่ใช้ในการสร้างชุดควบคุม
6. ทดสอบเซนเซอร์และชุดควบคุมไมโครคอนโทรลเลอร์กับจอแสดงผล
7. เขียนโปรแกรมสั่งการและแสดงผล
8. ทดสอบและแก้ไขการทำงานของระบบ
9. สรุปและจัดทำเอกสารรายงานการวิจัย

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับทักษะในด้านการวางแผน การตัดสินใจในระหว่างการทำโครงการ
2. ได้รับทักษะในการเขียนโปรแกรมและการออกแบบ
3. ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี สามารถทำงานได้ตามความคาดหวัง

1.6 รายละเอียดของปฏิญานินพนธ์

เนื้อหาที่จะกล่าวในปฏิญานินพนธ์ฉบับนี้ประกอบด้วย 5 บท และ 6 ภาคผนวกซึ่งมีรายละเอียดดังต่อไปนี้

บทที่ 1 บทนำ เป็นการกล่าวถึงที่มาของปฏิญานินพนธ์ วัตถุประสงค์ของการทำปฏิญานินพนธ์ ขอบเขตของโครงการ ขั้นตอนการดำเนินงาน ประโยชน์ที่คาดว่าจะได้รับและรายละเอียดของปฏิญานินพนธ์

บทที่ 2 ทฤษฎีที่เกี่ยวข้องเป็นการกล่าวถึง ทฤษฎี หลักการ อุปกรณ์ และความรู้ที่เกี่ยวข้องในการออกแบบ ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี เป็นการเพิ่มเติมความรู้ ทฤษฎีและความเข้าใจในอุปกรณ์ต่างๆ ก่อนเริ่มทำการโครงการ

บทที่ 3 ขั้นตอนการดำเนินงาน เป็นการอธิบายขั้นตอนการทำงานโดยละเอียดทั้งในด้านฮาร์ดแวร์และด้านซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 ผลการดำเนินงาน เป็นการแสดงผลการทำงานของอุปกรณ์ต่างๆและการทำงาน
ของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี รวมถึงอธิบายถึงวิธีการปรับจูนพารามิเตอร์ตัวควบคุม
บทที่ 5 ผลสรุปและข้อเสนอแนะ เป็นบทสรุปภาพรวมของชิ้นงานรวมถึงสิ่งที่จะพัฒนา
ต่อไปในอนาคต

ภาคผนวก ก โมดูล GY-521

ภาคผนวก ข Electronic speed controller

ภาคผนวก ค มอเตอร์ไร้แปรงถ่าน

ภาคผนวก ง จอแสดงผล LCD

ภาคผนวก จ Arduino UNO

ภาคผนวก ฉ โปรแกรมการทำงานของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

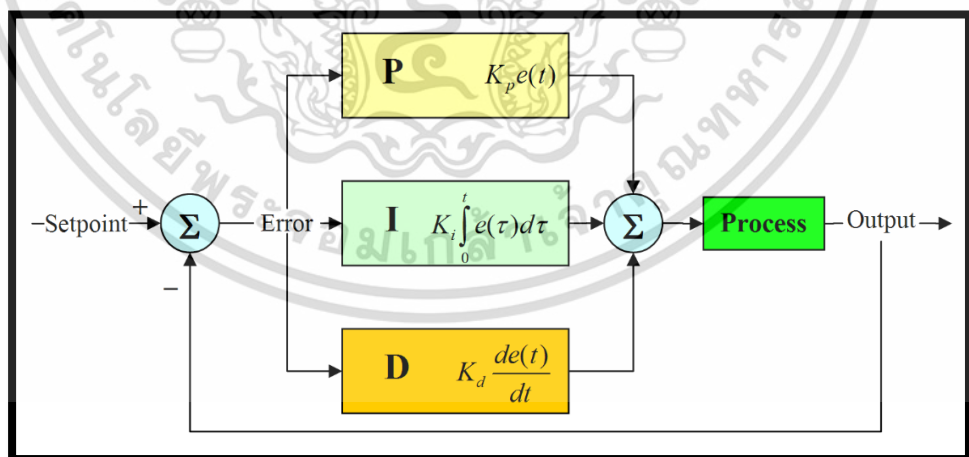
ทฤษฎีที่เกี่ยวข้อง

ในส่วนของทฤษฎีที่เกี่ยวข้อง ผู้จัดทำได้แบ่งออกเป็นสองส่วน คือ ทฤษฎีที่ใช้ในการทำงานของระบบ และ ฮาร์ดแวร์ซอฟต์แวร์ที่ใช้ในระบบ ในส่วนแรกจะประกอบไปด้วย หลักการควบคุมแบบ PID, แร็งยก, หลักการทำงานของ Complementary filter ในส่วนที่สองคือซอฟต์แวร์และฮาร์ดแวร์ที่ใช้ในระบบประกอบไปด้วย โมดูล GY-521, มอเตอร์ไร้แปรงถ่าน, โมดูลแสดงผล LCD, เพาเวอร์ซัพพลาย, บอร์ด Arduino UNO และโปรแกรม Arduino IDE

ทฤษฎีที่ใช้ในการทำงานของระบบ

2.1 หลักการควบคุมแบบ PID

ระบบควบคุมแบบสัดส่วน-ปริพันธ์-อนุพันธ์ (proportional – integral – derivative control system) เป็นระบบควบคุมแบบป้อนกลับที่ใช้กันอย่างกว้างขวาง ซึ่งค่าที่นำไปใช้ในการคำนวณเป็นค่าความผิดพลาดที่หามาจากความแตกต่างของตัวแปรในกระบวนการ (process value) และค่าที่ต้องการ (ค่าเป้าหมาย value) ตัวควบคุมจะพยายามลดค่าผิดพลาดให้เหลือน้อยที่สุดด้วยการปรับค่าสัญญาณขาเข้าของกระบวนการ ค่าตัวแปรของ PID ที่ใช้จะปรับเปลี่ยนตามธรรมชาติของระบบ



รูปที่ 2.1 แผนภาพบล็อกไดอะแกรมของการควบคุมแบบพีไอดี

(ที่มา : <https://www.primusthai.com/primus/Knowledge/info?ID=142>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีคำนวณของ PID ขึ้นอยู่กับสามตัวแปรคือค่าสัดส่วน ค่าปริพันธ์ และ ค่าอนุพันธ์ ค่าสัดส่วนกำหนดจากผลของความผิดพลาดในปัจจุบัน ค่าปริพันธ์กำหนดจากผลบนพื้นฐานของผลรวมความผิดพลาดที่ซึ่งพียงผ่านพ้นไปและค่าอนุพันธ์กำหนดจากผลบนพื้นฐานของอัตราการเปลี่ยนแปลงของความผิดพลาด น้ำหนักที่เกิดจากการรวมกันของทั้งสามตัวแปรนี้จะใช้ในการปรับกระบวนการ โดยการปรับค่าคงที่ใน ตัวควบคุม PID สามารถปรับรูปแบบการควบคุมให้เหมาะกับที่กระบวนการต้องการได้ การตอบสนองของตัวควบคุมจะอยู่ในรูปของการไหวตัวของตัวควบคุมจนถึงค่าความผิดพลาด ค่าโอเวอร์ชูต (overshoot) และ ค่าแกว่งของระบบ (oscillation) การควบคุมแบบ PID ไม่รับประกันได้ว่าจะจะเป็นระบบควบคุมที่เหมาะสมที่สุดหรือสามารถทำให้กระบวนการมีความเสถียรแน่นอน การประยุกต์ใช้งานบางครั้งอาจใช้เพียงหนึ่งถึงสองรูปแบบ ขึ้นอยู่กับกระบวนการเป็นสำคัญ PID บางครั้งจะถูกเรียกว่าการควบคุมแบบ PI, PD, P หรือ I ขึ้นอยู่กับว่าใช้รูปแบบใดบ้าง โดยในปริยญาณิพนธ์ฉบับนี้จะนำทฤษฎีในส่วนของการคำนวณมาใช้เพื่อประยุกต์ในการเขียนโปรแกรมเพื่อหาค่า พารามิเตอร์ควบคุมของระบบแบ่งเป็น ตัวควบคุมสัดส่วน หลักการคือสัญญาณจากตัวควบคุมที่ส่งไปปรับกระบวนการมีค่าเป็นสัดส่วนกับความคลาดเคลื่อนคือ

$$u(t) = K_p e(t) \quad (1)$$

เมื่อ K_p คือ ค่าอัตราขยาย
 $e(t)$ คือ ค่าความผิดพลาด
 t คือ เวลา

ตัวควบคุมแบบอินทิกรัล สามารถอธิบายได้ในสมการ

$$u(t) = K_p (e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau) \quad (2)$$

เมื่อ K_p คือ อัตราขยาย

T_i คือ integral time

การควบคุมแบบอินทิกรัล มีลักษณะเช่นเดียวกับการควบคุมสัดส่วนตรงผลกระทบของการเพิ่มอัตราขยายของตัวควบคุม หากอัตราขยายมีค่ามากเกินไปจะทำให้ผลตอบของระบบมีการแกว่งโดยทั่วไป Integral time ($T_i = 1/K_i \text{ sec}$ โดยที่ $K_i = \text{repeats/sec}$) เป็นตัวแสดงว่า อัตราการตอบสนองของกระบวนการต่อสัญญาณการควบคุม ค่า T_i ที่น้อยกว่าจะทำให้ตัวควบคุมมีการตอบสนองที่เร็วว่ในระยะเริ่มต้น โดยที่ความคลาดเคลื่อนยังเป็นค่าบวกอยู่ ดังนั้นกว่าความคลาดเคลื่อน จะเป็นศูนย์ (ซึ่งทำให้เทอม $\int_0^t e(\tau) d\tau$ หยุดทำงาน) เทอมไบแอสก็จะมีค่าสูงกว่าที่ ต้องการ ดังนั้นผลตอบสนองจึงเกิดส่วนพุ่งเกิน (overshoot) สูงกว่าค่ากำหนดเป็นผลให้ตัวอินทิกรัลทำหน้าที่ปรับให้ความคลาดเคลื่อนมีค่าลดลง การใช้ตัวอินทิกรัลในการควบคุม ควรระวังในเรื่องของ

ความคลาดเคลื่อนขนาดใหญ่ (เช่น เกิดการเปลี่ยนแปลงค่ากำหนดขนาดใหญ่) เพราะจะทำให้เกิดเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหา integral windup ถึงแม้ว่า Ti มีค่าถูกต้องในสภาวะการทำงานธรรมดา แต่สัญญาณควบคุมอาจถึงจุดอิ่มตัวขณะผลตอบเกิดส่วนพุ่งเกิน ตัวควบคุมแบบสัดส่วนและแบบอินทิกรัล ต่างก็มีข้อจำกัดอยู่ที่ความคลาดเคลื่อนขนาดใหญ่ ซึ่งเป็นปัญหาต่อการควบคุมกระบวนการ แต่ความคลาดเคลื่อนขนาดใหญ่นี้ สามารถรู้ได้ล่วงหน้าโดยพิจารณาจากแนวโน้มของความคลาดเคลื่อน หรืออัตราการเปลี่ยนแปลงของสัญญาณนั่นเอง ตัวอนุพันธ์มีหลักการทำงาน คือ ตัวควบคุมตอบสนองต่ออัตราการเปลี่ยนแปลงของความคลาดเคลื่อน ถึงแม้ว่าความคลาดเคลื่อนมียังค่าเล็กน้อย สัญญาณออกของตัวอนุพันธ์ไม่ได้สัมพันธ์กับขนาดของความคลาดเคลื่อน แต่ขึ้นอยู่กับอัตราการเปลี่ยนแปลงของความคลาดเคลื่อน ถ้าความคลาดเคลื่อนมีค่าคงที่ ตัวอนุพันธ์จะให้สัญญาณออกเป็นศูนย์ คุณลักษณะ ข้อนี้ มีผลดีคือ ตัวควบคุมจะมีผลตอบสนองที่เกิดก่อนที่ความคลาดเคลื่อนจะเพิ่มมากขึ้น และทำให้ระบบมีผลตอบสนองที่ เร็วขึ้น ตัวควบคุมแบบอนุพันธ์สามารถเขียนได้ดังนี้

$$u(t) = K_p (e(t) + T_d \frac{de(t)}{dt}) \quad (3)$$

เมื่อ T_d คือ derivative time

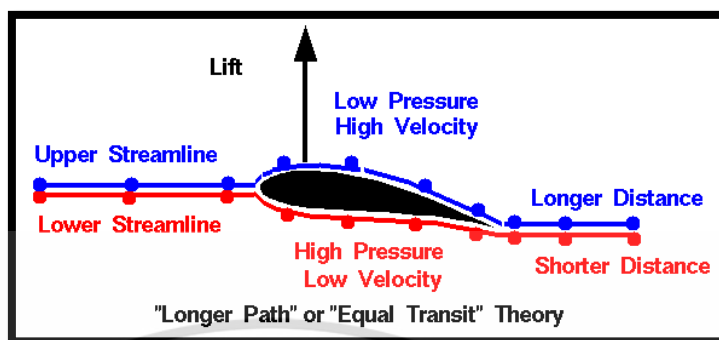
การเพิ่ม T_d จะทำให้ผลตอบสนองของ ตัวอนุพันธ์มีค่ามากขึ้น เนื่องจากตัวอนุพันธ์มีความไวต่อการเปลี่ยนแปลงมาก ดังนั้นจึงนิยมใช้กับค่าที่วัดได้เท่านั้น แต่ไม่ใช้กับค่ากำหนด เพราะการเปลี่ยนค่ากำหนดมักจะเป็นแบบขั้น (step) ทำให้ผลตอบสนองของตัวอนุพันธ์เป็นพัลส์ และทำให้เกิดการกระแทก (bump) ของอุปกรณ์ในกระบวนการ สำหรับค่ากำหนดใช้เฉพาะกับตัวควบคุมสัดส่วนและอินทิกรัล ตัวอนุพันธ์คือตัวควบคุมที่ก่อให้เกิดผลตรงข้ามกับตัวอินทิกรัล ดังนั้นจึงใช้ในการปรับปรุงกระบวนการที่มีการล่าช้าทางเวลา (time lag) มากๆ ทำให้ผลตอบสนองรวดเร็วขึ้น และช่วงเวลากว้างที่สั้นลง ข้อเสียของตัวอนุพันธ์ คือ มีความไวต่อสัญญาณรบกวนเป็นอย่างมาก เพราะมีผลตอบสนองโดยตรงต่ออัตราการเปลี่ยนแปลงของสัญญาณที่วัดได้ ดังนั้นแม้สัญญาณรบกวนจะมีขนาดเล็ก แต่ก็อาจก่อให้เกิดการเปลี่ยนแปลงต่อสัญญาณออกของตัวควบคุม จึงเป็นไปได้ที่จะใช้ตัวอนุพันธ์ในการควบคุมผลของสัญญาณรบกวน ยิ่งไปกว่านั้นระบบใดที่มีสัญญาณรบกวนมาก จะไม่สามารถใช้ตัวอนุพันธ์ ในวงการอุตสาหกรรมส่วนใหญ่นิยมใช้เพียงตัวควบคุม PI เท่านั้น

2.2 แรงยก

แรงยก คือแรงทางอากาศพลศาสตร์ (aerodynamic force) ชนิดหนึ่ง ที่ทำให้เครื่องบินสามารถลอยบนอากาศได้ แรงยกของเครื่องบินเกิดจากผลของมุมปะทะของปีกที่ติดตั้งกับลำตัวเครื่องบิน และผลของความแตกต่างของความดันที่กระทำที่พื้นผิวของปีกเครื่องบินด้านบนและด้านล่าง ในขณะที่บิน พื้นผิวของปีกเครื่องบินด้านบนจะมีความดันกระทำต่ำกว่าพื้นผิวของปีกเครื่องบินด้านล่าง ความดันที่กระทำกระจายตัวอยู่บนพื้นผิวดังกล่าวจะสร้างแรงกระทำลงบนพื้นผิวนั้นๆ เมื่อพื้นผิวด้านล่างมีความดันกระทำสูงกว่า จึงมีแรงกระทำเกิดขึ้นสูงกว่าที่พื้นผิวด้านบน และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

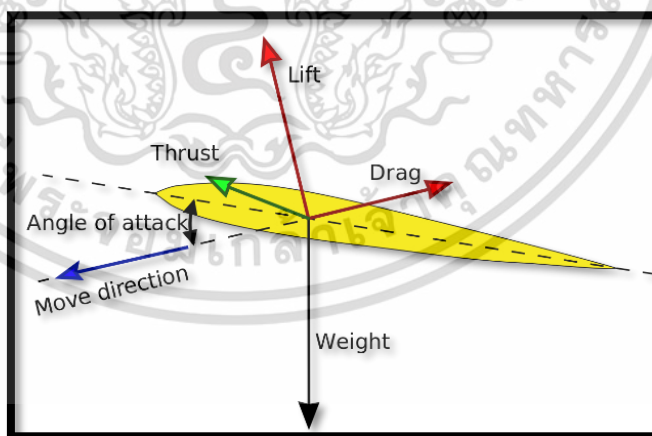
แรงยกคือผลต่างที่เหลื่อมอยู่ของแรงทั้งสอง ซึ่งสามารถใช้หลักการเรื่องของไหลของเบอนูลลี (Bernoulli) และกฎของนิวตัน (Newton law) มาอธิบายกลไกการเกิดแรงยกได้



รูปที่ 2.2 แรงกระทำที่เกิดขึ้นในภาคตัดขวางของปีก 1

(ที่มา : <http://www.rmutphysics.com/charud/virtualexperiment/virtual2>)

สาเหตุที่พื้นผิวด้านล่างของปีกและพื้นผิวด้านบนของปีกมีความดันกระทำไม่เท่ากันนั้น เป็นเพราะความเร็วของอากาศที่ไหลผ่านพื้นผิวทั้งสองฝั่งนั้นมีค่าไม่เท่ากัน ซึ่งการออกแบบรูปร่าง รูปทรงของปีก รวมทั้งมุมปะทะของการติดตั้งปีกเข้ากับลำตัวเครื่องบิน เป็นปัจจัยสำคัญในการทำให้เกิดปรากฏการณ์นี้ การที่เครื่องบินจะลอยได้นั้น แรงยกที่มีทิศทางยกลำตัวเครื่องบินขึ้นนั้นต้องมีขนาดมากกว่าแรงกระทำทั้งหมดที่มีทิศทางตรงกันข้าม หรือทิศทางลงสู่พื้น ซึ่งแรงกระทำดังกล่าวได้แก่น้ำหนักของเครื่องบิน แรงกระทำทางอากาศพลศาสตร์ และแรงอื่นๆ



รูปที่ 2.3 แรงกระทำที่เกิดขึ้นในภาคตัดขวางของปีก 2

(ที่มา : <http://www.rmutphysics.com/charud/virtualexperiment/virtual2>)

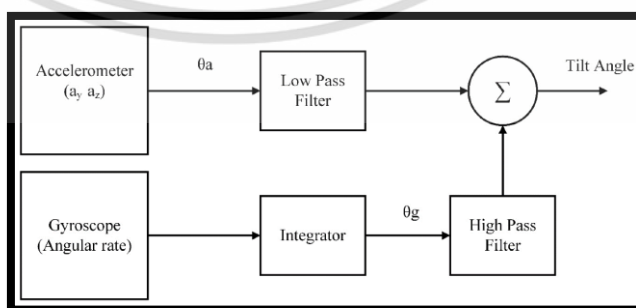
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 หลักการทำงานของ Complementary filter

Complementary filter เป็นอัลกอริทึมที่มีลักษณะคล้ายกับ Kalman filter ซึ่งใช้ในการรวมค่าที่ได้จาก Accelerometer (deg) และ Gyroscope (deg/s) ให้ได้มุมที่มีความถูกต้อง Complementary filter เป็นตัวกรองแบบดิจิทัลที่มีทฤษฎีในการทำงานมากพอสมควร ซึ่งในเบื้องต้นมีส่วนประกอบดังนี้ ค่าความเร็วเชิงมุมที่ได้จาก Gyroscope (deg/s) ค่าความเร่งเชิงมุมที่ได้จาก Accelerometer (deg) คาบของการประมวลผล มุมที่ได้จากตัววัดความเร่ง (red) และตัวที่ใช้ในการหาตำแหน่ง สมมติทราบความเร็วรถ (km/h) เราสามารถหาตำแหน่งได้จากการนำเวลามาคูณกับความเร็ว เช่นเดียวกันนี้เมื่อทราบความเร็วเชิงมุมของ Gyroscope สามารถหามุมได้โดยนำความเร็วเชิงมุมมาคูณกับเวลาซึ่งในที่นี้จะใช้ timer ของไมโครคอนโทรลเลอร์ในการจับเวลาและสามารถเขียนเป็นโค้ดได้ดังนี้

1. position += Gyro_angle[0]*dt เมื่อ dt คือ elapedTime
2. low-pass filter เป็นตัวที่ใช้ในการลดสัดส่วนของตัวแปรให้อยู่ในรูป short term เขียนเป็นโค้ดได้ดังนี้ $Total_angle[0] = (0.98)*Total_angle[0] + (0.02)* Acceleration_angle[0]$
3. high-pass filter จะมีลักษณะการทำงานที่ตรงข้ามกับ low-pass filter
4. sample period คืออัตราในการสุ่มค่ามาประมวลผลเช่นประมวลผลทุกๆ 0.01 วินาที จะได้ sample rate เป็น 100 Hz
5. time constant เป็นตัวแปรที่มีความสัมพันธ์กับระยะเวลาของสัญญาณ ซึ่งจะเป็นตัวกำหนดการประมวลผลของสัญญาณมีลักษณะเป็นดังนี้ $y = (a)*(y) + (1-a)*(x)$ ระยะเวลาในการทำงานสามารถได้จาก $t = (Total_angle[0]*dt)/(1-Total_angle[0])$ และเมื่อนำมาเขียนเป็นสมการ Complementary filter จะเขียนได้เป็น

$$\text{ค่ามุมที่ได้} = (0.98)*(Total_angle + Gyro_angle*elapedTime) + (0.02)*(Acc_angle) \quad (4)$$



รูปที่ 2.4 หลักการทำงานของ Complementary filter

(ที่มา : <https://robotics.stackexchange.com/questions/1717>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทฤษฎีของฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในระบบ

2.4 โมดูล GY-521

โมดูลวัดความเอียง (GY-521 MPU6050 3axis gyroscope accelerometer sensor module) สามารถวัดความเอียงได้แบบ 3 ทิศทาง ความละเอียดในการวัด 16 บิต ใช้การเชื่อมต่อกับไมโครคอนโทรลเลอร์ผ่าน I2C โมดูลนี้ได้รวมเอาเซนเซอร์ Gyroscope และ Accelerometer เข้าไว้ด้วยกัน

2.4.1 Accelerometer

ใช้สำหรับวัดความเร่ง หรือความเอียง เมื่อเราจับเซนเซอร์วางขนานกับพื้นโลก ค่าที่ได้จาก Accelerometer ในแกน z จะประมาณ 10 เนื่องจากถูกแรงโน้มถ่วงของโลกดึงไว้ แต่เมื่อเราเอียงเซนเซอร์ไปในทิศทางต่าง ๆ ค่าในแต่ละแกนจะเปลี่ยนไป เช่น เอียงเซนเซอร์ไปทางซ้ายที่ 90 องศาและหยุดนิ่ง ค่าที่ได้แกน z จะเป็น 0 แต่แกน x หรือ y จะเป็น 10 เนื่องจากแรงโน้มถ่วงดึงให้ค่าในแกน x หรือ y ลงด้านล่าง

2.4.2 Gyroscope

ใช้วัดความเร่งในการเคลื่อนที่ ความสามารถของ Accelerometer จะให้ค่าที่คงที่อยู่เสมอ แตกต่างจาก Gyroscope ที่จะให้ค่าเมื่อมีการเคลื่อนที่เท่านั้น หน่วยวัดเป็น องศาเรเดียนต่อวินาที (rad/s) การใช้งานวัดความเอียง การหมุนต่าง ๆ นิยมใช้งานทั้ง Accelerometer และ Gyroscope คู่กัน เนื่องจาก Accelerometer ไม่สามารถวัดการหมุนในทิศทางที่เซนเซอร์วางขนานกับพื้นโลกได้ (เมื่อหมุนเซนเซอร์ที่วางขนานกับพื้นโลก ค่า x y และ z จะไม่เปลี่ยนแปลง) จึงจำเป็นต้องใช้ Gyroscope มาใช้วัดร่วมด้วย เซนเซอร์นี้สามารถนำไปประยุกต์ใช้ร่วมกับไมโครคอนโทรลเลอร์เพื่อควบคุมความสมดุล หรือนำไปใช้กับบาลานซ์โรบอต (balanced robot) เพื่อให้หุ่นยนต์สามารถตั้งตรงได้อยู่เสมอ นอกจากนี้ยังนำไปวัดความลาดเอียงเพื่อประยุกต์ใช้อื่น ๆ เช่น นำไปติดกับไม้เท้าเพื่อตรวจจับการล้มของผู้สูงอายุ

2.5 มอเตอร์ไร้แปรงถ่าน

มอเตอร์ไร้แปรงถ่าน (brushless DC motor , BLDC) มีอีกชื่อหนึ่งว่ามอเตอร์เปลี่ยนทิศทางด้วยกระแสไฟฟ้า เนื่องจากไม่มีแปรงถ่านที่โรเตอร์และสับเปลี่ยนทิศทางด้วยไฟฟ้าที่ตำแหน่งเฉพาะบนโรเตอร์ BLDC เป็นมอเตอร์ซิงโครนัสแม่เหล็กถาวรที่มีรูปคลื่น EMF เป็นเอกลักษณ์ จึงทำงานคล้ายกับมอเตอร์ DC มีแปรงถ่าน BLDC จะไม่ทำงานจากแหล่งจ่ายไฟ DC โดยตรง แต่หลักในการทำงานนั้นจะเหมือนมอเตอร์ DC BLDC มีโรเตอร์พร้อมแม่เหล็กถาวรและสเตเตอร์พันขดลวด มอเตอร์ BLDC จึงเป็นมอเตอร์ DC ที่กลับด้านในออกด้านนอก ไม่มีแปรงถ่านและตัวสับเปลี่ยนทิศทางและเชื่อมต่อขดลวดกับอิเล็กทรอนิกส์ควบคุม อิเล็กทรอนิกส์ควบคุมนี้จะทดแทนฟังก์ชันของตัวสับเปลี่ยนทิศทางและให้พลังงานกับขดลวดที่เหมาะสม ขดลวดจะได้รับพลังงานในรูปแบบซึ่งเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมุนรอบสเตเตอร์ ขดลวดสเตเตอร์ที่ได้รับพลังงานจะส่งไปยังแม่เหล็กของมอเตอร์และสลักเมื่อโรเตอร์มีแนวตรงกับสเตเตอร์ BLDC เหมาะสำหรับการใช้งานที่ต้องการความน่าเชื่อถือสูง ประสิทธิภาพสูง และมีกำลังต่อปริมาณสูง โดยทั่วไปแล้ว BLDC ถือเป็นมอเตอร์ประสิทธิภาพสูงที่สามารถให้แรงบิดปริมาณมากในหลายช่วงกำลังได้

2.6 โมดูลแสดงผล LCD

คำว่า LCD ย่อมาจากคำว่า Liquid Crystal Display ซึ่งเป็นจอที่ทำมาจากผลึกคริสตัลเหลว หลักการคือด้านหลังจอจะมีไฟส่องสว่าง หรือที่เรียกว่า backlight อยู่ เมื่อมีการปล่อยกระแสไฟฟ้าเข้าไปกระตุ้นที่ผลึก ก็จะทำให้ผลึกโปร่งแสง ทำให้แสงที่มาจากไฟ backlight แสดงขึ้นมาบนหน้าจอ ส่วนอื่นที่โดนผลึกปิดกั้นไว้ จะมีสีที่แตกต่างกันตามสีของผลึกคริสตัล เช่น สีเขียว หรือ สีฟ้า ทำให้เมื่อมองไปที่จอก็จะพบกับตัวหนังสือสีขาว แล้วพบกับพื้นหลังสีต่างๆ กัน จอ LCD จะแบ่งเป็น 2 แบบใหญ่ๆ ตามลักษณะการแสดงผลดังนี้

2.6.1 Graphic LCD

เป็นจอที่สามารถกำหนดได้ว่าจะให้แต่ละจุดบนหน้าจอขึ้นแสง หรือปล่อยแสงออกไป ทำให้จอสามารถสร้างรูปขึ้นมาบนหน้าจอได้ การระบุขนาดจะระบุในลักษณะของจำนวนจุด (pixels) ในแต่ละแนว เช่น 128x64 หมายถึงจอที่มีจำนวนจุดตามแนวนอน 128 จุด และมีจุดตามแนวตั้ง 64 จุด

2.6.2 Character LCD

เป็นจอที่แสดงผลเป็นตัวอักษรตามช่องแบบตายตัว เช่น จอ LCD ขนาด 16x2 หมายถึงใน 1 แถว มีตัวอักษรใส่ได้ 16 ตัว และมีทั้งหมด 2 บรรทัดให้ใช้งาน ส่วน 20x4 จะหมายถึงใน 1 แถว มีตัวอักษรใส่ได้ 20 ตัว และมีทั้งหมด 4 บรรทัด Character LCD มีการเชื่อมต่อจะมีด้วยกัน 2 แบบคือ

2.6.2.1 การเชื่อมต่อแบบขนาน

เป็นการเชื่อมต่อจอ LCD เข้ากับบอร์ด Arduino โดยตรง โดยจะแบ่งเป็นการเชื่อมต่อแบบ 4 บิต และการเชื่อมต่อแบบ 8 บิต ใน Arduino จะนิยมเชื่อมต่อแบบ 4 บิต เนื่องจากใช้สายในการเชื่อมต่อน้อยกว่า

2.6.2.2 การเชื่อมต่อแบบอนุกรม

เป็นการเชื่อมต่อกับจอ LCD ผ่านโมดูลแปลงรูปแบบการเชื่อมต่อกับจอ LCD จากแบบขนาน มาเป็นการเชื่อมต่อแบบอื่นที่ใช้สายน้อยกว่า เช่น การใช้โมดูล I2C Serial Interface จะเป็นการนำโมดูลเชื่อมเข้ากับตัวจอ LCD แล้วใช้บอร์ด Arduino เชื่อมต่อกับบอร์ดโมดูลผ่านโปรโตคอล I2C ทำให้ใช้สายเพียง 4 เส้น ก็ทำให้หน้าจอแสดงผลข้อความต่างๆออกมาได้

2.7 เพาเวอร์ซัพพลาย

เพาเวอร์ซัพพลายที่เป็นสวิตซิงเพาเวอร์ซัพพลาย (switching power supply) ซึ่งเป็นแหล่งจ่ายไฟตรงคงค่าแรงดันแบบหนึ่ง และสามารถเปลี่ยนแรงดันไฟจากไฟสลับโวลต์สูง ให้เป็นแรงดันไฟตรงค่าต่ำ เพื่อใช้ในงานอิเล็กทรอนิกส์ได้เช่นเดียวกันแหล่งจ่ายไฟเชิงเส้น (linear power supply) ถึงแม้เพาเวอร์ซัพพลายทั้งสองแบบจะต้องมีการใช้หม้อแปลงในการลดทอนแรงดันสูงให้เป็นแรงดันต่ำเช่นเดียวกัน แต่สวิตซิงเพาเวอร์ซัพพลายจะต้องการใช้หม้อแปลงที่มีขนาดเล็ก และน้ำหนักเบา เมื่อเทียบกับแหล่งจ่ายไฟเชิงเส้น อีกทั้งสวิตซิงเพาเวอร์ซัพพลายยังมีประสิทธิภาพสูงกว่าอีกด้วย ในปัจจุบันสวิตซิงเพาเวอร์ซัพพลายได้เข้ามามีบทบาทกับชีวิตเราอย่างมาก เครื่องใช้อิเล็กทรอนิกส์ขนาดเล็กซึ่งต้องการแหล่งจ่ายไฟที่มีกำลังสูงแต่มีขนาดเล็ก เช่น เครื่องคอมพิวเตอร์ เครื่องโทรสาร และ โทรศัพท์ จำเป็นจะต้องใช้สวิตซิงเพาเวอร์ซัพพลาย แนวโน้มการนำสวิตซิงเพาเวอร์ซัพพลายมาใช้ในเครื่องใช้อิเล็กทรอนิกส์ทุกประเภทจึงเป็นไปได้สูง การศึกษาหลักการการทำงานและการออกแบบสวิตซิงเพาเวอร์ซัพพลายจึงเป็นสิ่งจำเป็นที่ไม่อาจหลีกเลี่ยงได้สำหรับผู้ที่เกี่ยวข้องกับงานอิเล็กทรอนิกส์ทุกประเภท

2.8 บอร์ด Arduino UNO

คำว่า Uno เป็นภาษาอิตาลี ซึ่งแปลว่าหนึ่ง เป็นบอร์ด Arduino รุ่นแรกๆ ที่ออกมา มีขนาดประมาณ 68.6x53.4mm เป็นบอร์ดมาตรฐานที่นิยมใช้งานมากที่สุด เนื่องจากเป็นขนาดที่เหมาะสมสำหรับการเริ่มต้นเรียนรู้ Arduino และมี Shields ให้เลือกใช้งานได้มากกว่าบอร์ด Arduino รุ่นอื่นๆ ที่ออกแบบมาเฉพาะมากกว่า โดยบอร์ด Arduino Uno ได้มีการพัฒนาเรื่อยมา ตั้งแต่ R2 R3 และรุ่นย่อยที่เปลี่ยนชิปไอซีเป็นแบบ SMD

2.9 โปรแกรม Arduino IDE

Arduino IDE คือโปรแกรมสำหรับใช้เขียนโปรแกรม, คอมไพล์ และอัปโหลดโปรแกรมลงบอร์ด Arduino หรือบอร์ดตัวอื่นๆ ที่คล้ายกัน เช่น Generic ESP8266 modules, NodeMCU หรือ WeMos D1 เป็นต้น แนวคิดการใช้งานโปรแกรม Arduino IDE เรียงลำดับขั้นตอนดังนี้

1. เขียนโปรแกรมด้วยภาษา C/C++ สำหรับ Arduino
2. คอมไพล์หรือแปลโปรแกรมภาษา C/C++ ให้เป็นภาษาสำหรับไมโครคอนโทรลเลอร์ และบันทึกเป็น Intel Hex File
3. อัปโหลด Intel Hex File ลงบนไมโครคอนโทรลเลอร์ซึ่งอยู่บนบอร์ด Arduino ผ่านสาย USB หรือผ่าน Programmer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินงาน

3.1 ขั้นตอนการดำเนินงาน

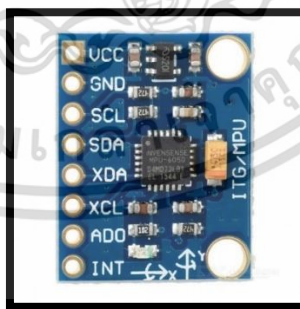
ขั้นตอนการดำเนินงานได้วางแผนไว้ดังนี้

1. ศึกษาค้นคว้าทฤษฎีและเอกสารที่เกี่ยวข้อง
2. ศึกษาหลักการการทำงานของไมโครคอนโทรลเลอร์และเซนเซอร์ที่จะนำมาใช้
3. ออกแบบและวางแผนการสร้างโครงสร้างและติดตั้งอุปกรณ์
4. สั่งซื้ออุปกรณ์ที่ใช้ประกอบโครงสร้าง
5. ทดสอบอุปกรณ์ที่ใช้ในการสร้างชุดควบคุม
6. ทดสอบเซนเซอร์และชุดควบคุมไมโครคอนโทรลเลอร์กับจอแสดงผล
7. เขียนโปรแกรมควบคุมการทำงานของระบบ
8. สร้างชุดอุปกรณ์ตามแบบที่วางแผนไว้
9. ทดสอบและแก้ไขการทำงานของระบบ
10. สรุปและจัดทำเอกสารรายงานการวิจัย

3.2 อุปกรณ์ที่เกี่ยวข้อง

3.2.1 โมดูล GY-521

ทำหน้าที่รับค่ามุมที่เปลี่ยนแปลงของระบบคานเทียบกับระนาบที่กำหนดไว้โดยจะส่งค่าไปประมวลผลที่อุปกรณ์ไมโครคอนโทรลเลอร์ เพื่อปรับระนาบของระบบคานให้เท่ากับค่าที่กำหนดให้ ดังรูปที่ 3.1



รูปที่ 3.1 โมดูล GY-521

(ที่มา : [https://www.arduinoall.com/product/34/โมดูล GY-521](https://www.arduinoall.com/product/34/โมดูล%20GY-521))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 Electronic speed controller

ทำหน้าที่ในการควบคุมมอเตอร์ไร้แปรงถ่านผ่านคำสั่งจากอุปกรณ์ไมโครคอนโทรลเลอร์
อุปกรณ์ Electronic speed controller แสดงดังรูปที่ 3.2



รูปที่ 3.2 อุปกรณ์ Electronic speed controller
(ที่มา : <http://www.readytosky.com>)

3.2.3 มอเตอร์ไร้แปรงถ่าน

มีหน้าที่ในการปรับมุมของระนาบคาน ซึ่งควบคุมการทำงานด้วย Electronic speed controller ดังรูปที่ 3.3



รูปที่ 3.3 มอเตอร์ไร้แปรงถ่าน
(ที่มา : <http://www.readytosky.com>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 เพาเวอร์ซัพพลาย

ทำหน้าที่ให้พลังงานแก่ระบบ โดยใช้หม้อแปลงอินพุตขนาด 220 โวลต์ เอาต์พุตขนาด 12 โวลต์ 50 แอมแปร์ ดังรูปที่ 3.4

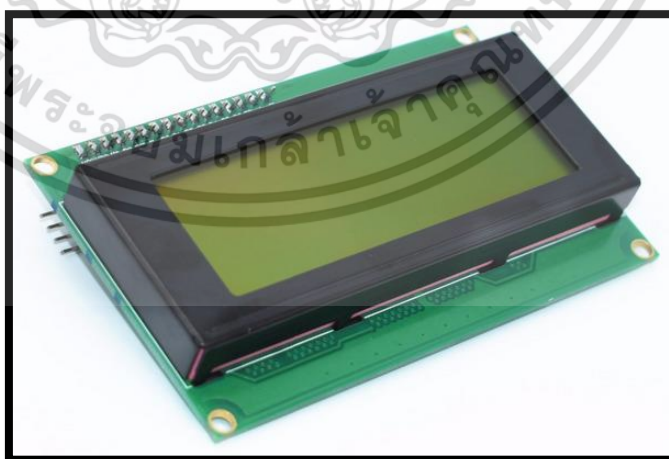


รูปที่ 3.4 เพาเวอร์ซัพพลาย

(ที่มา : <https://www.sunsky-online.com>)

3.2.5 จอแสดงผล LCD

เป็นโมดูลแสดงผลแบบ 20 ตัวอักษร 4 แถวมีทำหน้าที่แสดงค่าเป้าหมาย และค่าพารามิเตอร์พีไอดี ที่กำหนดให้กับระบบ ดังรูปที่ 3.5



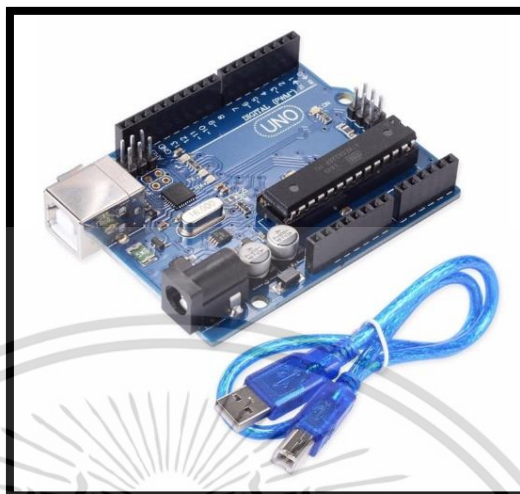
รูปที่ 3.5 อุปกรณ์แสดงผล LCD

(ที่มา : www.arduinoall.com/product/lcddisplay)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6 บอร์ด Arduino UNO

ทำหน้าที่ประมวลผลและควบคุมการทำงานของระบบ ดังรูปที่ 3.6



รูปที่ 3.6 อุปกรณ์ Arduino UNO

(ที่มา : www.arduinoall.com/product/arduinouno)

3.3 ซอฟต์แวร์ที่เกี่ยวข้อง

3.3.1 โปรแกรม Arduino IDE

ใช้สำหรับเขียนโปรแกรมการทำงานต่างๆ ให้กับบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO

ดังรูปที่ 3.7



รูปที่ 3.7 โปรแกรม Arduino IDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การออกแบบและการวางแผนการทำงาน

3.4.1 การออกแบบและวางแผนทางด้านฮาร์ดแวร์

1. ศึกษาและเลือกเซนเซอร์ที่ต้องการ
2. ศึกษาและเลือกไมโครคอนโทรลเลอร์ที่ต้องการใช้งาน
3. ทดสอบอุปกรณ์เซนเซอร์ จอแสดงผล และมอเตอร์
4. ออกแบบระบบคานสมดุล
5. ติดตั้งอุปกรณ์ ตามรูปแบบที่ออกแบบไว้

3.4.2 การออกแบบและวางแผนทางด้านซอฟต์แวร์

1. ศึกษาโปรแกรมที่สามารถเขียนคำสั่งให้กับอุปกรณ์ไมโครคอนโทรลเลอร์ที่จะนำมาใช้
2. ศึกษาวิธีการเขียนโปรแกรม
3. ฝึกใช้โปรแกรมด้วยการเขียนคำสั่งเพื่อทดสอบอุปกรณ์ เซนเซอร์ และมอเตอร์
4. เขียนโปรแกรม Arduino เพื่อรับค่าจากเซนเซอร์ เพื่อใช้ในการควบคุมการทำงานของมอเตอร์และส่งค่าไปที่จอแสดงผลตามรูปแบบที่กำหนด
5. ทดสอบโปรแกรมและระบบการทำงานของอุปกรณ์

3.5 วิธีการดำเนินงาน

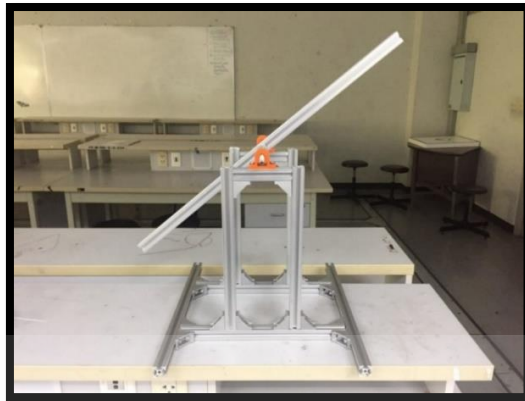
3.5.1 ออกแบบและสร้างโครงสร้างของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี

ประกอบโครงสร้างของระบบคานสมดุลตามที่ได้ออกแบบไว้โดยอุปกรณ์ที่ใช้ทำโครงสร้างคือ อลูมิเนียมโปรไฟล์ ขั้นตอนในการประกอบโครงสร้างมีดังนี้

1. ออกแบบลักษณะและขนาดของโครงสร้างที่ต้องการ
2. ตัดอลูมิเนียมโปรไฟล์ตามขนาดที่ได้ออกแบบไว้
3. ประกอบโครงสร้างของระบบคานสมดุล



รูปที่ 3.8 โครงสร้างของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี (1)
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในงานเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 โครงสร้างของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี (2)

3.5.2 การทดสอบเซนเซอร์

ทดสอบการทำงานของเซนเซอร์โดยการเขียนโค้ดเพื่อรับค่าเอาต์พุตของเซนเซอร์และนำมาแสดงผลที่ serial monitor ในโปรแกรม Arduino IDE โดยผลการทดสอบแสดงให้เห็นว่าอุปกรณ์สามารถทำงานได้ตามปกติ

3.5.2.1 โปรแกรมที่ใช้ในการทดสอบโมดูล GY-521

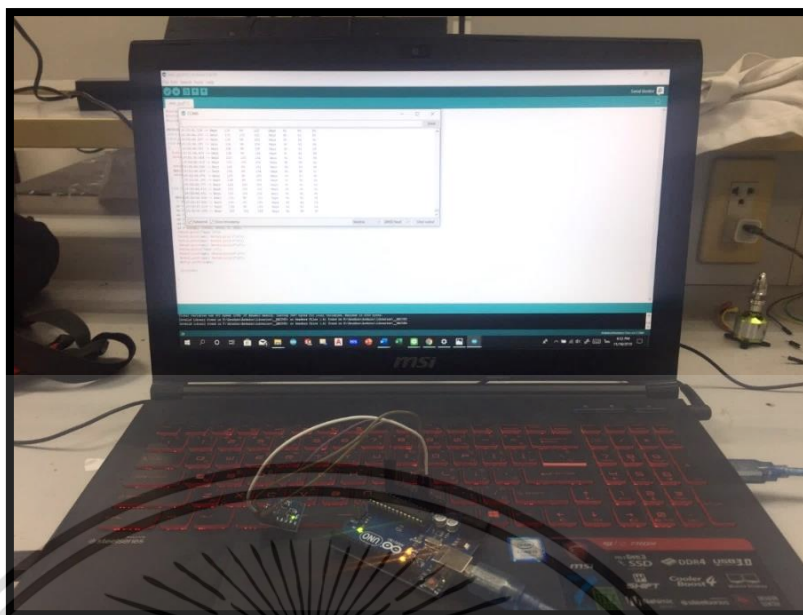
```

// ทดสอบโมดูล GY-521
#include "Wire.h"
#include "MPU6050.h"
MPU6050 MPU(0x68);
int16_t ax, ay, az;
int16_t gx, gy, gz;
void setup() {
  Wire.begin();
  Serial.begin(9600);
  MPU.begin();
  MPU.getAxisInnov(0x0000); // "Connection Failed";
}
void loop() {
  MPU.getMOTLoc(xax, yax, zaz, gax, gay, gaz);
  // show position from 0-180 deg
  ax = map(xax, -18000, 18000, 0, 180);
  ay = map(yax, -18000, 18000, 0, 180);
  az = map(zax, -18000, 18000, 0, 180);
  gx = map(gax, -18000, 18000, 0, 360);
  gy = map(gay, -18000, 18000, 0, 360);
  gz = map(gaz, -18000, 18000, 0, 360);
  Serial.print("Ax:"); Serial.print(ax);
  Serial.print("Ay:"); Serial.print(ay);
  Serial.print("Az:"); Serial.print(az);
  Serial.print("Gx:"); Serial.print(gx);
  Serial.print("Gy:"); Serial.print(gy);
  Serial.print("Gz:"); Serial.print(gz);
  Serial.println();
}

```

รูปที่ 3.10 โปรแกรมที่ใช้ในการทดสอบ โมดูล GY-521

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 การทดสอบโมดูล GY-521

3.5.3 การทดสอบมอเตอร์และ Electronic speed controller

เนื่องจากมอเตอร์และ Electronic speed controller ที่ใช้ในระบบเป็นอุปกรณ์ที่ผ่านการใช้งานมาแล้ว ดังนั้นจึงได้ทดสอบเพื่อสังเกตการทำงานและประสิทธิภาพของอุปกรณ์ ทดสอบโดยการเขียนคำสั่งในการควบคุมมอเตอร์ด้วยตัวต้านทานปรับค่าได้และวัดความเร็วรอบของมอเตอร์ด้วยอุปกรณ์วัดความเร็วรอบ โดยกำหนดคาบเวลาที่ใช้ในการทดสอบของมอเตอร์และ Electronic speed controller เป็น 1200 μ s

3.5.3.1 โปรแกรมที่ใช้ในการทดสอบการทำงานของชุดมอเตอร์

```

Control_Brushless_Motor_Arduino_1.6.7
File Edit Sketch Tools Help
Control_Brushless_Motor
#include <Servo.h> //Using servo library to control ESC
Servo esc; //Creating a servo object with name as esc

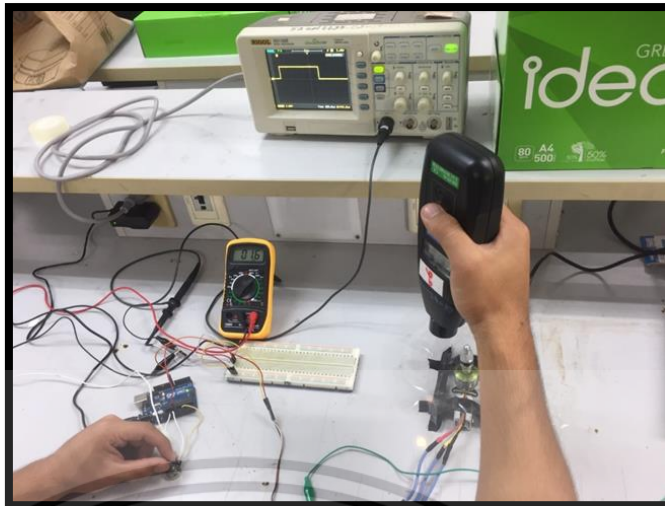
void setup()
{
  esc.attach(8); //Specify the esc signal pin, here as D8
  esc.writeMicroseconds(1000); //initialize the signal to 1000
  Serial.begin(9600);
}

void loop()
{
  int val; //Creating a variable val
  val = analogRead(A0); //Read input from analog pin A0 and store in val
  val = map(val, 0, 1023, 1000, 2000); //mapping the val to minimum and maximum input
  esc.writeMicroseconds(val); //using val as the signal to esc
}

```

รูปที่ 3.12 โปรแกรมที่ใช้ในการทดสอบการทำงานของมอเตอร์และ Electronic speed controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 ทดสอบการทำงานและประสิทธิภาพของมอเตอร์

3.5.4 เขียนโปรแกรมที่ใช้ในการทำงานของระบบ

ขั้นตอนนี้เป็นขั้นตอนในการเขียนโปรแกรมเพื่อรับข้อมูลจากเซนเซอร์มาประมวลผลเพื่อนำไปใช้ในการควบคุมการทำงานของมอเตอร์

```

Bvwm_balancing_system_using_PID_controller_Code | Arduino 1.8.10
File Edit Sketch Tools Help

Bvwm_balancing_system_using_PID_controller_Code

#include <Wire.h>
#include <Servo.h>
#include <LiquidCrystal_I2C.h> //กำหนดค่าจะเป็นค่าที่ใช้ Library

LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE); //ใช้กับ address ของ LCD
Servo right_prop; //ใช้กับ ESC.
Servo left_prop;

//กำหนดค่าแปรผันในการคำนวณ
int16_t Acc_rawX, Acc_rawY, Acc_rawZ, Gyr_rawX, Gyr_rawY, Gyr_rawZ; //กำหนดค่าแปรผันของ Gyro และ Acc
float Acceleration_angle[2];
float Gyro_angle[2];
float Total_angle[2];
//float testgyro;

float elapsedTime, time, timePrev;
float rad_to_deg = 180/3.141592654;

float PID, pwmLeft, pwmRight, error, previous_error;

double chrottle=1300; //กำหนดค่าในการทำงานของมอเตอร์
  
```

รูปที่ 3.14 หน้าต่างโปรแกรม Arduino IDE (1)

จากรูปที่ 3.14 จะประกอบไปด้วยสามส่วนตามลำดับ ในส่วนแรกจะเป็นการประกาศใช้ไลบรารีที่จำเป็นต่อการทำงานของระบบ ส่วนที่สองจะเป็นการกำหนดการทำงานของอุปกรณ์ในที่นี่ประกอบไปด้วย มอเตอร์และจอแสดงผล ส่วนที่สามจะเป็นการกำหนดตัวแปรในการรับข้อมูลของเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Bwam_balancing_system_using_PID_controller_Code | Arduino 1.8.10
File Edit Sketch Tools Help

Bwam_balancing_system_using_PID_controller_Code
//////////////////////////////////////////////////////////////////เรียกใช้อุปกรณ์และเซตค่าเริ่มต้นในการทำงาน//////////////////////////////////////////////////////////////////
void setup() {
time = millis();
lcd.begin(20,4); //คำสั่งในภาษารียกใช้งาน LCD ประเภท 20*4
lcd.setBacklight(HIGH);
Wire.begin(); //คำสั่งเริ่มต้นการทำงานและใช้งาน ฟังก์ชัน wire
Wire.beginTransmission(0x68); //เรียกใช้ GX 521 ตาม Address จาก Data Sheet
Wire.write(0x6B);
Wire.write(0);
Wire.endTransmission(true);
Serial.begin(250000);
right_prop.attach(3); //กำหนดพอร์ทที่ใช้ในการส่งสัญญาณ PWM
left_prop.attach(5);

left_prop.writeMicroseconds(1000); //ค่าเริ่มต้นในการทำงานของ ESC.
right_prop.writeMicroseconds(1000);

delay(5000); // ถ่วงเวลาส่วนนี้ในภาษาริ่ทำงาน
}

```

รูปที่ 3.15 หน้าต่างโปรแกรม Arduino IDE (2)

จากรูปที่ 3.15 คือโปรแกรมส่วนที่เรียกใช้อุปกรณ์และเซตค่าเริ่มต้นในการทำงานของอุปกรณ์ในที่นี้ประกอบไปด้วย จอแสดงผล เซนเซอร์ และ Electronic speed controller สาเหตุที่จำเป็นในการเซตค่าเริ่มต้นของอุปกรณ์ ยกตัวอย่างของ Electronic speed controller หากได้รับค่าเริ่มต้นที่ไม่ใช่ค่าต่ำสุดอุปกรณ์จะไม่สามารถทำงานได้ และเนื่องจากไลบรารีของเซอร์โว คาบต่ำสุดที่สามารถทำได้คือ 1000 μ s

```

Bwam_balancing_system_using_PID_controller_Code | Arduino 1.8.10
File Edit Sketch Tools Help

Bwam_balancing_system_using_PID_controller_Code

void loop() {

float pid_p=0; //กำหนดค่าเริ่มต้นของ PID
float pid_i=0;
float pid_d=0;

float xdeg = analogRead(A0); // กำหนดค่าป้อนให้กับ Analog
float xKp = analogRead(A1);
float xKi = analogRead(A2);
float xKd = analogRead(A3);
float deg,ii,pp,dd; //กำหนดค่าป้อนค่าให้ map
deg = map(xdeg,0,1023,-4000,4000);
ii = map(xKi,0,1023,0,10000);
pp = map(xKp,0,1023,0,10000);
dd = map(xKd,0,1023,0,10000);

float pi,ii,di,deg1; //กำหนดค่าแมปของ (หน่วย) map มัดนิยม
deg1 = deg/100;
pi = pp/1000;
ii = ii/1000;
di = dd/1000;

double kp= pi; // กำหนดค่าแปรที่ซับซ้อนไว้คำนวณ
double ki= ii;
double kd= di;

float desired_angle = deg1; // กำหนดค่าเซตพอนต์ที่ต้องการในโปรแกรมนี้จากค่าที่ -40 - 40 องศา
}

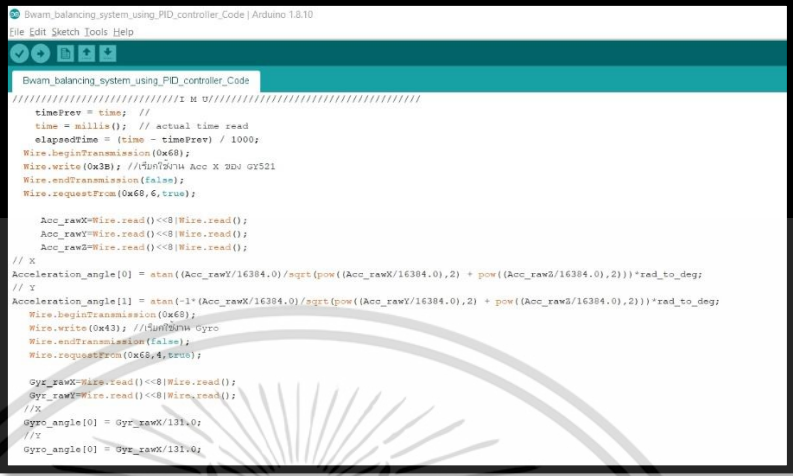
```

รูปที่ 3.16 หน้าต่างโปรแกรม Arduino IDE (3)

จากรูปที่ 3.16 จะประกอบไปด้วยสี่ส่วน โดยส่วนที่หนึ่ง เป็นการกำหนดค่าเริ่มต้นของพีไอดี ส่วนที่สองคือการกำหนดตัวแปรที่ใช้รับค่าแอนะล็อกเพื่อใช้ในการควบคุมค่าคงที่พีไอดี ส่วนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามคือการกำหนดตัวแปรในการเก็บค่าคงที่ ที่ได้จากการทำให้เป็นจำนวนทศนิยมด้วยฟังก์ชัน map ส่วนที่สี่คือส่วนที่กำหนดตัวแปรสำหรับรับค่าที่จะนำไปใช้คำนวณในระบบและค่าเป้าหมาย



```

Swam_balancing_system_using_PID_controller_Code | Arduino 1.8.10
File Edit Sketch Tools Help

Swam_balancing_system_using_PID_controller_Code
////////////////////////////////////////////////////////////////////////////////////////////////////////////////
timePrev = time; //
time = millis(); // actual time read
elapsedTime = (time - timePrev) / 1000;
Wire.beginTransmission(0x68);
Wire.write(0x20); //รับข้อมูล Acc X 2D0 0x211
Wire.endTransmission(false);
Wire.requestFrom(0x68, 6, true);

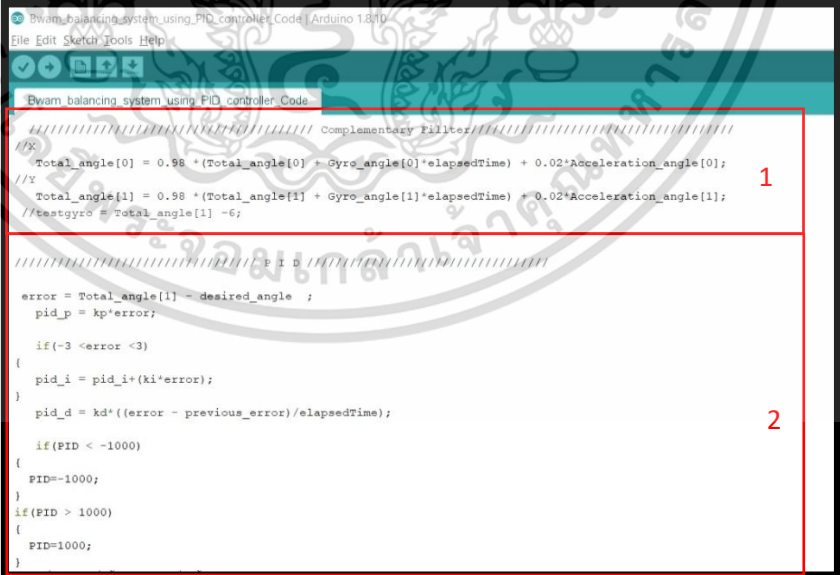
Acc_rawX=Wire.read()<<8|Wire.read();
Acc_rawY=Wire.read()<<8|Wire.read();
Acc_rawZ=Wire.read()<<8|Wire.read();

// X
Acceleration_angle[0] = atan((Acc_rawY/16384.0)/sqrt(pow((Acc_rawX/16384.0),2) + pow((Acc_rawZ/16384.0),2)))*rad_to_deg;
// Y
Acceleration_angle[1] = atan(-1*(Acc_rawX/16384.0)/sqrt(pow((Acc_rawY/16384.0),2) + pow((Acc_rawZ/16384.0),2)))*rad_to_deg;
Wire.beginTransmission(0x6B);
Wire.write(0x43); //รับข้อมูล Gyro
Wire.endTransmission(false);
Wire.requestFrom(0x6B, 4, true);

Gyr_rawX=Wire.read()<<8|Wire.read();
Gyr_rawY=Wire.read()<<8|Wire.read();
//X
Gyro_angle[0] = Gyr_rawX/131.0;
//Y
Gyro_angle[1] = Gyr_rawY/131.0;
  
```

รูปที่ 3.17 หน้าต่างโปรแกรม Arduino IDE (4)

จากรูปที่ 3.17 จะประกอบไปด้วยส่วนเดียวคือ IMU ในส่วนนี้เป็นคำสั่งเพื่อให้ตัวเซนเซอร์ ส่งค่าที่วัดได้กลับมาคำนวณซึ่งในที่นี้ เป็นการเรียกใช้แอดเดรสของอุปกรณ์คือแอดเดรสของ Gyroscope และ Accelerometer และนำค่าดิบที่ได้มาคำนวณให้เป็นค่ามุมเพื่อนำไปคำนวณในระบบต่อ



```

Swam_balancing_system_using_PID_controller_Code | Arduino 1.8.10
File Edit Sketch Tools Help

Swam_balancing_system_using_PID_controller_Code
//////////////////////////////////////////////////////////////////////////////////////////////////////////////// Complementary Filter////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//X
Total_angle[0] = 0.98 * (Total_angle[0] + Gyro_angle[0]*elapsedTime) + 0.02*Acceleration_angle[0];
//Y
Total_angle[1] = 0.98 * (Total_angle[1] + Gyro_angle[1]*elapsedTime) + 0.02*Acceleration_angle[1];
//testgyro = Total_angle[1] -6;

//////////////////////////////////////////////////////////////////////////////////////////////////////////////// P I D////////////////////////////////////////////////////////////////////////////////////////////////////////////////

error = Total_angle[1] - desired_angle ;
pid_p = kp*error;

if(-3 <error <3)
{
  pid_i = pid_i+(ki*error);
}
pid_d = kd*((error - previous_error)/elapsedTime);

if(PID < -1000)
{
  PID=-1000;
}
if(PID > 1000)
{
  PID=1000;
}
  
```

รูปที่ 3.18 หน้าต่างโปรแกรม Arduino IDE (5)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.18 ประกอบไปด้วยสองส่วนโดยในส่วนแรกคือโค้ดในการทำงานของ complementary filter เพื่อลดสัญญาณรบกวนของเซนเซอร์ ส่วนที่สองคือ การกำหนดการทำงานของ การประมวลผลการคำนวณระบบควบคุมพีไอดี

```

Swam_balancing_system_using_PID_controller_Code
File Edit Sketch Tools Help

Swam_balancing_system_using_PID_controller_Code
}
//นำค่า PID ที่ออกมาบวกเข้าไปกับการขึ้นคานสัญญาณของ zsc.
pwmLeft = throttle + PID;
pwmRight = throttle - PID;

if (pwmRight < 1000) //กำหนดค่าต่ำสุดและสูงสุดของคานสัญญาณเนื่องจากคานสัญญาณในไมโครจะขยับได้เพียง 1000-2000 ไมโครเซค
{
  pwmRight = 1000;
}
if (pwmRight > 2000)
{
  pwmRight = 2000;
}
//Left
if (pwmLeft < 1000)
{
  pwmLeft = 1000;
}
if (pwmLeft > 2000)
{
  pwmLeft = 2000;
}

left_prop.writeMicroseconds (pwmLeft); //นำค่าที่หามาเขียนค่าลงสัญญาณที่โดยการควบคุม
right_prop.writeMicroseconds (pwmRight);
previous_error = error;

```

รูปที่ 3.19 หน้าต่างโปรแกรม Arduino IDE (6)

จากรูปที่ 3.19 ประกอบไปด้วยสองส่วนโดยในส่วนแรก คือการกำหนดขอบเขตของสัญญาณที่ใช้ควบคุมมอเตอร์ไม่ให้เกินจากช่วง 1000-2000 μ s และส่วนที่สองคือคำสั่งให้มอเตอร์ทำงานตามสัญญาณที่ได้รับในที่นี้คือ pwmLeft และ pwmRight และตัวแปรเก็บค่าความคลาดเคลื่อนเพื่อใช้ในการคำนวณ

```

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////ค่าของ lcd//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
lcd.setCursor(1,0); //รับค่าเพื่อแสดงผลในส่วนของเดกรี
Serial.print(deg1);
lcd.print("Degree :"),
lcd.print(deg1);

lcd.setCursor(6,1); //รับค่าเพื่อแสดงผลในส่วนของ pi
Serial.print(pi);
lcd.print("Pi :"),
lcd.print(pi);

lcd.setCursor(6,2); //รับค่าเพื่อแสดงผลในส่วนของ xi
Serial.print(xi);
lcd.print("xi :"),
lcd.print(xi);

lcd.setCursor(6,3); //รับค่าเพื่อแสดงผลในส่วนของ dd
Serial.print(dd);
lcd.print("dd :"),
lcd.print(di);

} //จบการทำงาน

```

รูปที่ 3.20 หน้าต่างโปรแกรม Arduino IDE (7)

จากรูปที่ 3.20 ประกอบด้วยส่วนของคำสั่งที่ใช้ในการควบคุมจอแสดงผล ประเภท 20*4 ให้แสดงค่าคงที่พีไอดี และค่าเป้าหมายที่ต้องการกำหนดให้กับระบบ

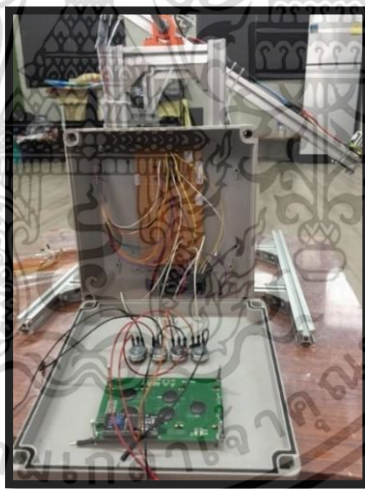
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.5 ประกอบกล่องควบคุมสำหรับระบบ

กล่องควบคุมจะประกอบไปด้วย สองส่วนหลัก โดยส่วนแรกจะเป็นส่วนของจอแสดงผล และส่วนรับค่าพารามิเตอร์ของพีไอดี และส่วนที่สองคือภายในกล่องจะประกอบไปด้วย แผ่น PCB และบอร์ด Arduino UNO



รูปที่ 3.21 ลักษณะภายนอกของกล่องควบคุม

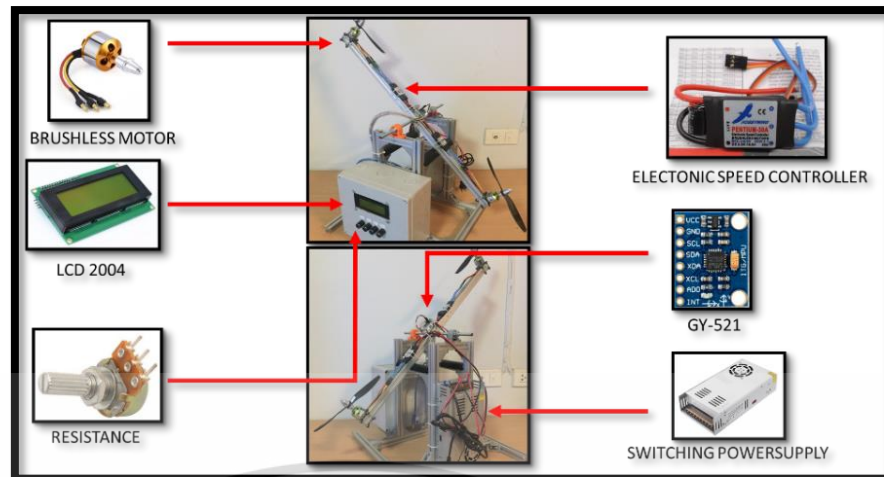


รูปที่ 3.22 ลักษณะภายในของกล่องควบคุม

3.5.6 การติดตั้งอุปกรณ์ที่ใช้ในการทำงาน

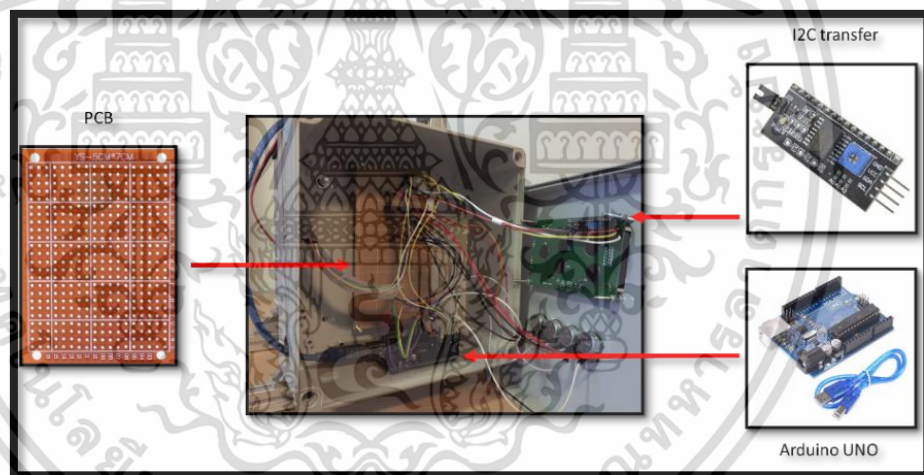
การติดตั้งอุปกรณ์จะแบ่งเป็นสองส่วนคือส่วนของโครงสร้างและส่วนของกล่องควบคุมการทำงานโดยในส่วนของโครงสร้างจะประกอบด้วย มอเตอร์ Electronic speed controller โมดูล GY-521 และเพาเวอร์ซัพพลาย ในส่วนของกล่องควบคุมการทำงานของระบบจะประกอบด้วย บอร์ดไมโครคอนโทรลเลอร์ จอแสดงผล ตัวต้านทานสำหรับปรับค่าพารามิเตอร์ แผ่นวงจร และตัวแปลง I2C สำหรับจอแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.23 อุปกรณ์ที่ใช้ในระบบการทำงาน (1)

จากรูปที่ 3.23 แสดงให้เห็นถึงอุปกรณ์และตำแหน่งของอุปกรณ์ในส่วน of โครงสร้างเมื่อติดตั้งบนชุดระบบคานสมดุลโดยใช้ตัวควบคุมแบบพีไอดี



รูปที่ 3.24 อุปกรณ์ที่ใช้ในระบบการทำงาน (2)

จากรูปที่ 3.24 แสดงให้เห็นถึงอุปกรณ์และตำแหน่งของอุปกรณ์ในส่วน of กล่องควบคุมระบบการทำงาน of ระบบเมื่อติดตั้งบนชุดระบบคานสมดุลโดยใช้ตัวควบคุมแบบพีไอดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

ในส่วนของบทนี้จะกล่าวถึงผลลัพธ์ในการทดสอบอุปกรณ์ที่ใช้ในระบบ และผลลัพธ์ในการทดสอบการทำงานของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดีโดยรายละเอียดของอุปกรณ์ที่ทดสอบประกอบด้วย โมดูล GY-521 และ ชุดมอเตอร์และ Electronic speed controller ในส่วนของการทดสอบการทำงานของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี จะเป็นการสังเกตการทำงานเมื่อใส่ค่าพารามิเตอร์ควบคุมให้กับระบบจากกราฟผลตอบสนอง

4.1 ผลการทดสอบอุปกรณ์

การทดสอบอุปกรณ์ที่ใช้ในระบบการทำงานของระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดีแบ่งเป็น โมดูล GY-521 และ ชุดมอเตอร์และ Electronic speed controller

4.1.1 ผลการทดสอบการทำงานของ โมดูล GY-521

ทดสอบโดยการเขียนโค้ดเพื่อดึงข้อมูลจากอุปกรณ์มาแสดงใน serial monitor ของโปรแกรม Arduino IDE และการเปรียบเทียบค่ามุมของอุปกรณ์ที่วัดจาก ไมโครแทรกเตอร์ชนิดครึ่งวงกลมกับค่ามุมที่ได้จากเซนเซอร์

ก. ทดสอบโดยการเขียนโค้ดเพื่อดึงข้อมูลจากอุปกรณ์มาแสดงใน serial monitor ของโปรแกรม Arduino IDE

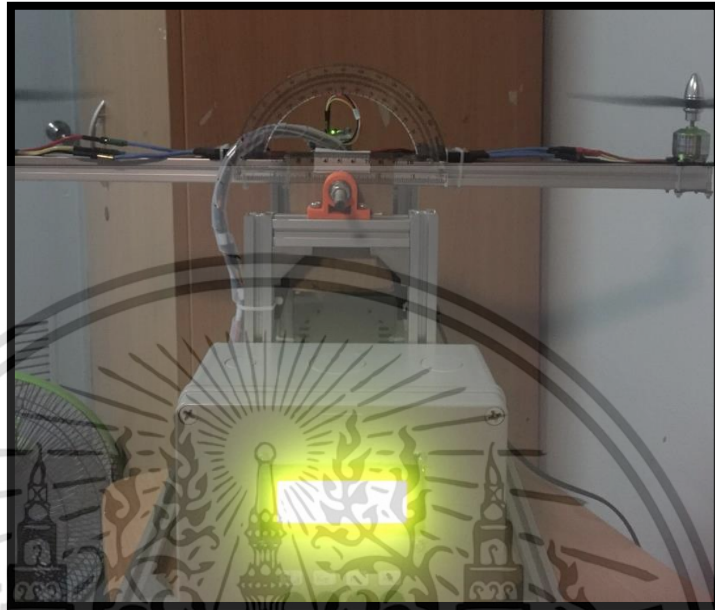
```
COM6
12:26:58.882 -> gyromax : -2.52 gyroZ : -7.75 gyroX : -1.39
12:26:58.888 -> angleX : 11.91 angleZ : 17.77
12:26:58.892 -> gyroXmax : 282.20 gyroYmax : -83.25 gyroZmax : 31.44
12:26:58.897 -> angleX : 11.76 angleZ : 7.54 angleY : 71.64
12:26:58.900 ->
-----
12:27:04.041 -> temp : 31.24
12:27:04.044 -> gyroZ : 0.44 gyroX : 0.19 gyroY : 0.46
12:27:04.114 -> gyroX : 0.32 gyroY : 0.86 gyroZ : 0.32
12:27:04.187 -> angleXmax : 13.62 angleZmax : 7.83
12:27:04.204 -> gyroXmax : 281.62 gyroZmax : -88.40 gyroYmax : -33.85
12:27:04.280 -> angleX : 11.59 angleZ : 7.45 angleY : -21.95
12:27:04.274 ->
-----
12:27:04.1369 -> temp : 31.24
12:27:04.140 -> gyroZ : 0.32 gyroX : 0.13 gyroY : 0.37
12:27:04.145 -> gyroX : 8.70 gyroY : 10.15 gyroZ : 4.72
12:27:04.146 -> angleXmax : 7.73 angleZmax : 4.72
12:27:04.1519 -> gyroXmax : 281.62 gyroZmax : -88.40 gyroYmax : -33.72
12:27:04.1564 -> angleX : 10.47 angleZ : 6.93 angleY : -22.72
12:27:04.157 ->
-----
12:27:04.452 -> temp : 31.24
12:27:04.456 -> gyroZ : 0.08 gyroX : 0.80 gyroY : 1.65
12:27:04.529 -> gyroX : 88.09 gyroY : 353.01 gyroZ : 13.44
12:27:04.534 -> angleXmax : 85.26 angleZmax : -1.84
12:27:04.539 -> gyroXmax : 307.60 gyroZmax : -301.87 gyroYmax : -61.21
12:27:04.543 -> angleX : 31.53 angleZ : 18.99 angleY : -41.21
12:27:04.543 ->
-----
12:27:04.917 -> temp : 31.19
12:27:04.922 -> gyroZ : -0.16 gyroX : 0.19 gyroY : 0.64
12:27:04.932 -> gyroX : 24.48 gyroY : -37.44 gyroZ : 13.48
12:27:04.934 -> angleXmax : 13.21 angleZmax : 16.93
12:27:04.939 -> gyroXmax : 322.96 gyroZmax : -61.31 gyroYmax : -35.65
12:27:04.941 -> angleX : 20.25 angleY : -1.45 angleZ : -50.35
12:27:04.941 ->
-----
12:27:05.209 -> temp : 31.24
12:27:05.214 -> gyroZ : 0.12 gyroX : 0.47 gyroY : 0.74
12:27:05.233 -> gyroX : 122.38 gyroY : 285.61 gyroZ : -110.45
12:27:05.236 -> angleXmax : 37.75 angleZmax : 4.82
12:27:05.241 -> gyroXmax : 304.44 gyroZmax : 169.44 gyroYmax : -73.49
12:27:05.241 -> angleX : 18.07 angleY : 13.88 angleZ : -73.49
12:27:05.244 ->
-----
12:27:05.549 -> temp : 31.24
12:27:05.554 -> gyroZ : 0.06 gyroX : 0.10 gyroY : 0.36
12:27:05.624 -> gyroX : 0.52 gyroY : 0.71 gyroZ : 0.36
12:27:05.629 -> angleXmax : 0.00 angleZmax : 0.00
12:27:05.634 -> gyroXmax : 281.20 gyroZmax : 62.80 gyroYmax : -74.20
12:27:05.634 -> angleX : 0.21 angleY : 0.55 angleZ : -74.20
12:27:05.634 ->
-----
12:27:05.944 -> temp : 31.24
12:27:05.949 -> gyroZ : 0.06 gyroX : 0.10 gyroY : 0.36
12:27:05.954 -> gyroX : 0.52 gyroY : 0.71 gyroZ : 0.36
12:27:05.959 -> angleXmax : 0.00 angleZmax : 0.00
12:27:05.964 -> gyroXmax : 281.20 gyroZmax : 62.80 gyroYmax : -74.20
12:27:05.964 -> angleX : 0.21 angleY : 0.55 angleZ : -74.20
12:27:05.964 ->
-----
```

รูปที่ 4.1 ข้อมูลจาก โมดูล GY-521

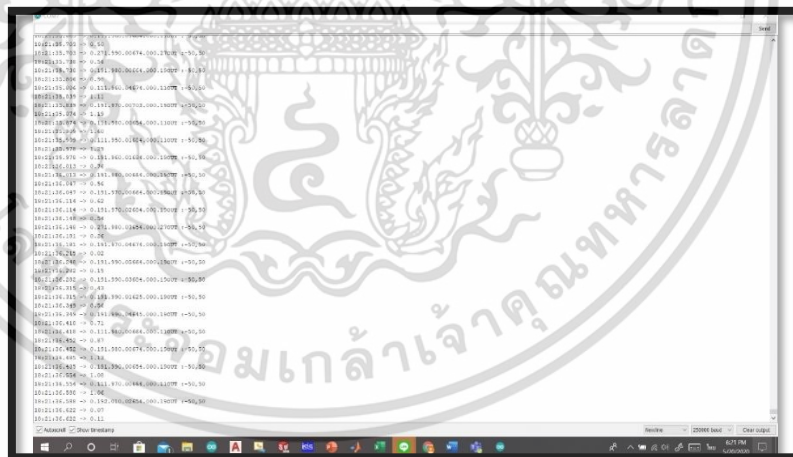
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข. การเปรียบเทียบค่ามุมของอุปกรณ์ที่วัดจาก ไมโครแทรกเตอร์ชนิดครึ่งวงกลมกับค่ามุมที่ได้จากเซนเซอร์

มุมของคานขณะทำงานที่ 0 องศา



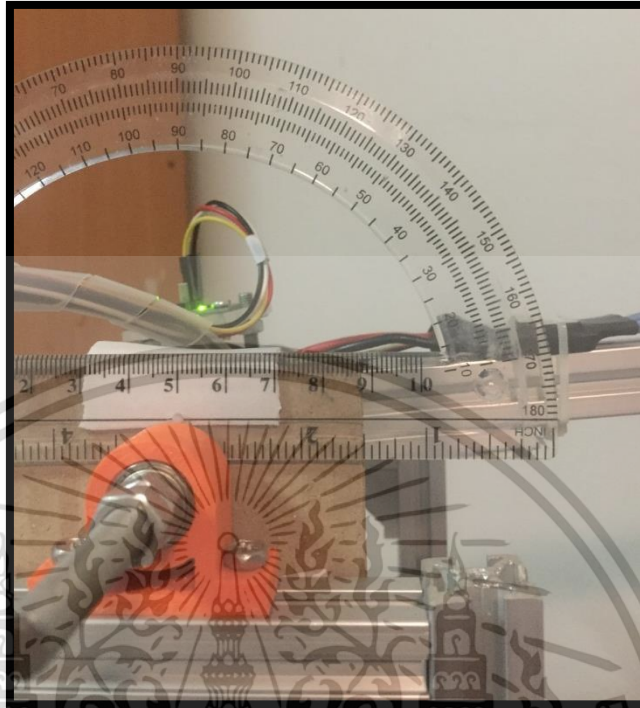
รูปที่ 4.2 มุมของคานขณะ 0 องศา



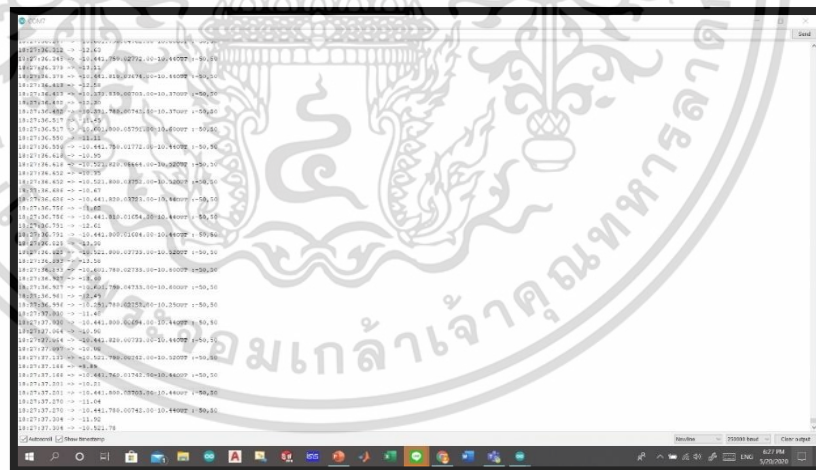
รูปที่ 4.3 มุมที่ได้จากเซนเซอร์ขณะ 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมของคานขณะทำงานที่ -10 องศา



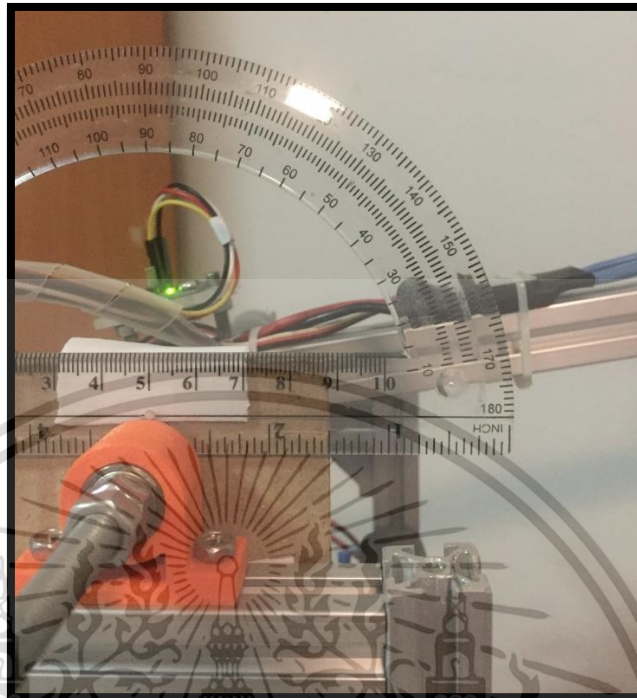
รูปที่ 4.4 มุมของคานขณะ -10 องศา



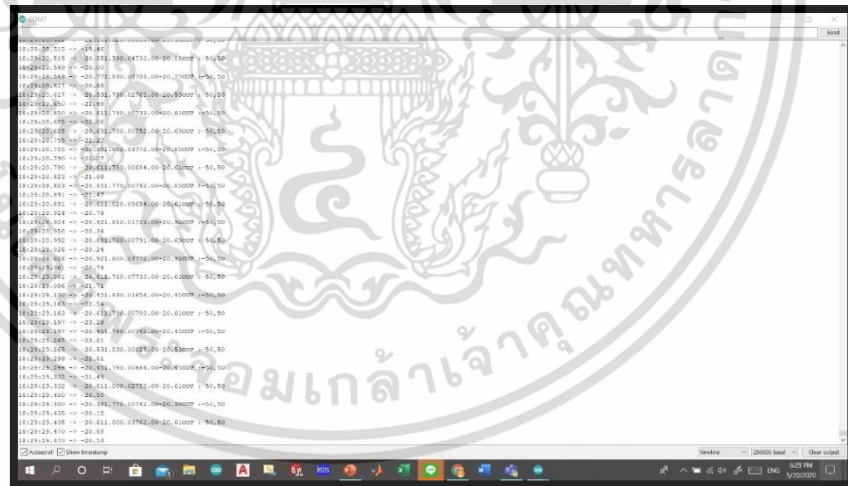
รูปที่ 4.5 มุมที่ได้จากเซนเซอร์ขณะ -10 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมของคานขณะทำงานที่ -20 องศา



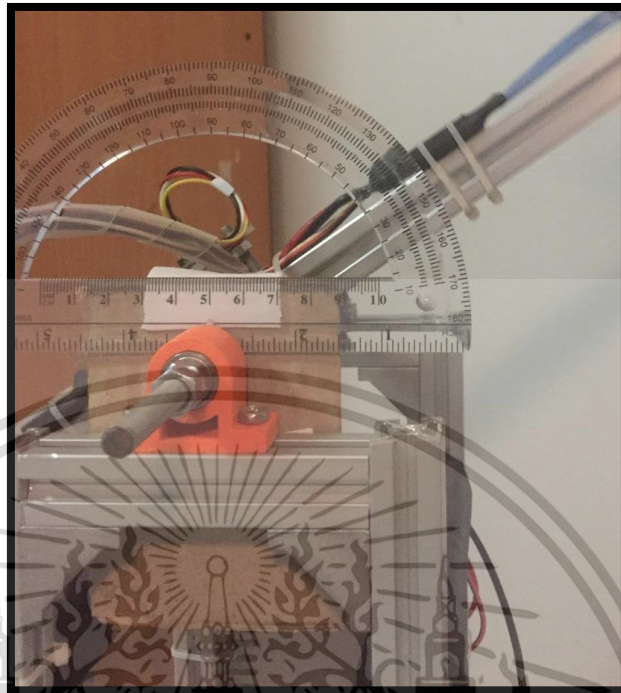
รูปที่ 4.6 มุมของคานขณะ -20 องศา



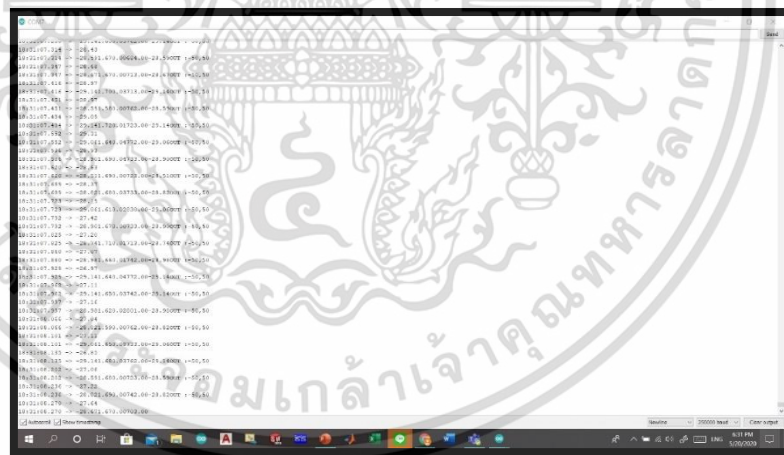
รูปที่ 4.7 มุมที่ได้จากเซนเซอร์ขณะ -20 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมของคานขณะทำงานที่ -30 องศา



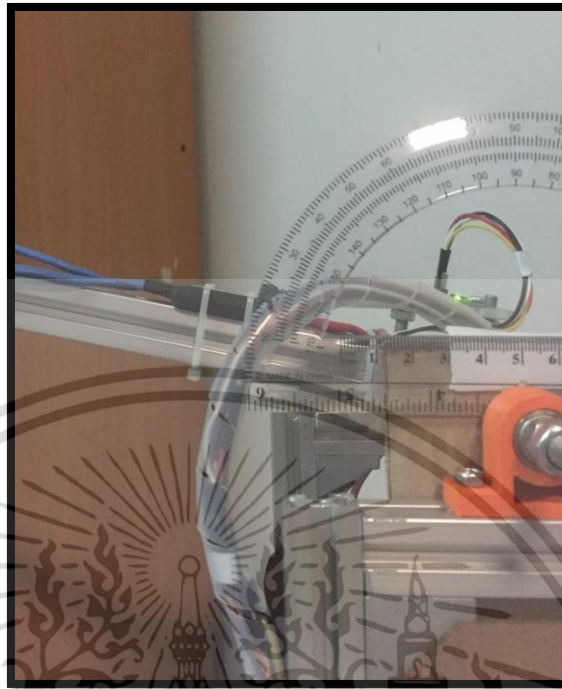
รูปที่ 4.8 มุมของคานขณะ -30 องศา



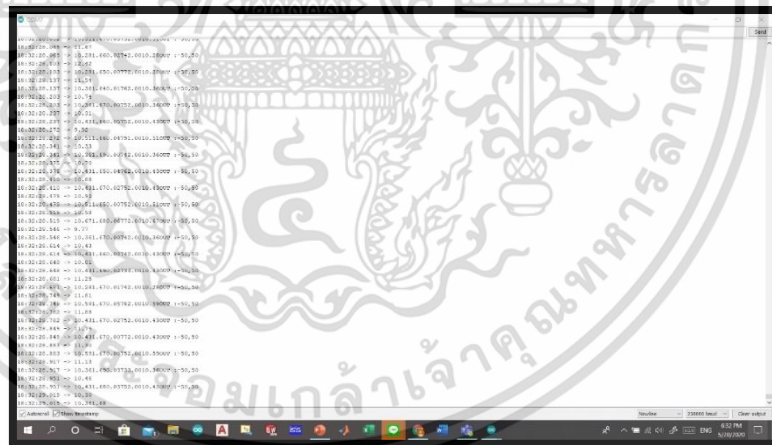
รูปที่ 4.9 มุมที่ได้จากเซนเซอร์ขณะ -30 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมของคานขณะทำงานที่ 10 องศา



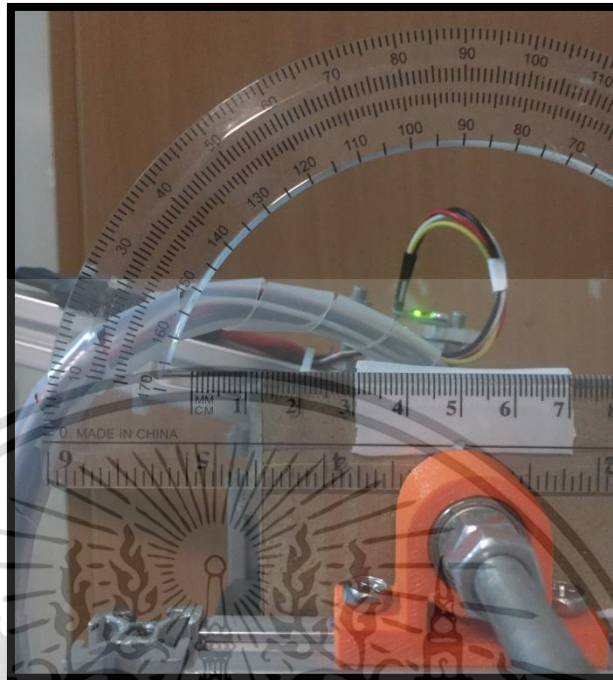
รูปที่ 4.10 มุมของคานขณะ 10 องศา



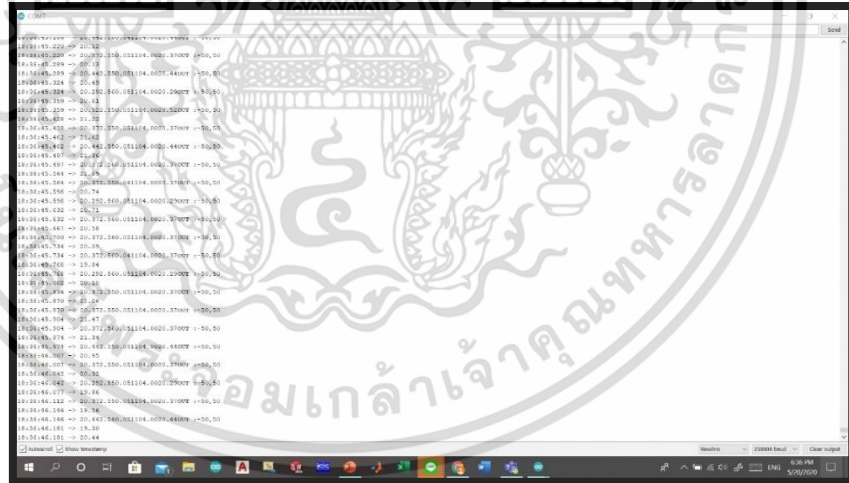
รูปที่ 4.11 มุมที่ได้จากเซนเซอร์ขณะ 10 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมของคานขณะทำงานที่ 20 องศา



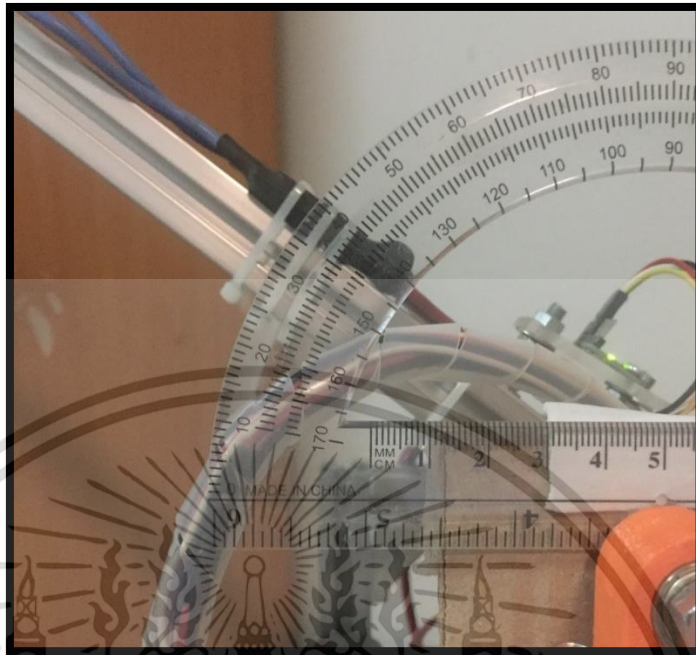
รูปที่ 4.12 มุมของคานขณะ 20 องศา



รูปที่ 4.13 มุมที่ได้จากเซนเซอร์ขณะ 20 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมของคานขณะทำงานที่ 30 องศา



รูปที่ 4.14 มุมของคานขณะ 30 องศา

```

การคำนวณค่าคงที่สปริงของคาน
101 37:23.920 -> 30.452,660,081104,0030,450000 +50,50
102 37:23.960 -> 30.50
103 37:23.995 -> 30.5112,300,041270,0030,440000 +50,50
104 37:24.030 -> 31.05
105 37:24.065 -> 30.5118,550,051025,0030,430000 +50,50
106 37:24.100 -> 31.61
107 37:24.135 -> 30.5125,800,061104,0030,380000 +50,50
108 37:24.170 -> 31.80
109 37:24.205 -> 30.5132,050,071104,0030,360000 +50,50
110 37:24.240 -> 31.84
111 37:24.275 -> 30.5138,300,081104,0030,360000 +50,50
112 37:24.310 -> 31.80
113 37:24.345 -> 30.452,660,081104,0030,450000 +50,50
114 37:24.380 -> 31.87
115 37:24.415 -> 30.5145,550,091104,0030,360000 +50,50
116 37:24.450 -> 31.84
117 37:24.485 -> 30.5151,800,101104,0030,360000 +50,50
118 37:24.520 -> 31.80
119 37:24.555 -> 30.452,660,081104,0030,450000 +50,50
120 37:24.590 -> 31.80
121 37:24.625 -> 30.5158,050,111104,0030,360000 +50,50
122 37:24.660 -> 31.80
123 37:24.695 -> 30.452,660,081104,0030,450000 +50,50
124 37:24.730 -> 31.84
125 37:24.765 -> 30.5164,300,121104,0030,360000 +50,50
126 37:24.800 -> 31.80
127 37:24.835 -> 30.452,660,081104,0030,450000 +50,50
128 37:24.870 -> 31.84
129 37:24.905 -> 30.5170,550,131104,0030,360000 +50,50
130 37:24.940 -> 31.80
131 37:24.975 -> 30.452,660,081104,0030,450000 +50,50
132 37:25.010 -> 31.84
133 37:25.045 -> 30.5176,800,141104,0030,360000 +50,50
134 37:25.080 -> 31.80
135 37:25.115 -> 30.452,660,081104,0030,450000 +50,50
136 37:25.150 -> 31.84
137 37:25.185 -> 30.5183,050,151104,0030,360000 +50,50
138 37:25.220 -> 31.80
139 37:25.255 -> 30.452,660,081104,0030,450000 +50,50
140 37:25.290 -> 31.84
141 37:25.325 -> 30.5189,300,161104,0030,360000 +50,50
142 37:25.360 -> 31.80
143 37:25.395 -> 30.452,660,081104,0030,450000 +50,50
144 37:25.430 -> 31.84
145 37:25.465 -> 30.5195,550,171104,0030,360000 +50,50
146 37:25.500 -> 31.80
147 37:25.535 -> 30.452,660,081104,0030,450000 +50,50
148 37:25.570 -> 31.84
149 37:25.605 -> 30.5201,800,181104,0030,360000 +50,50
150 37:25.640 -> 31.80
151 37:25.675 -> 30.452,660,081104,0030,450000 +50,50
152 37:25.710 -> 31.84
153 37:25.745 -> 30.5208,050,191104,0030,360000 +50,50
154 37:25.780 -> 31.80
155 37:25.815 -> 30.452,660,081104,0030,450000 +50,50
156 37:25.850 -> 31.84
157 37:25.885 -> 30.5214,300,201104,0030,360000 +50,50
158 37:25.920 -> 31.80
159 37:25.955 -> 30.452,660,081104,0030,450000 +50,50
160 37:25.990 -> 31.84
161 37:26.025 -> 30.5220,550,211104,0030,360000 +50,50
162 37:26.060 -> 31.80
163 37:26.095 -> 30.452,660,081104,0030,450000 +50,50
164 37:26.130 -> 31.84
165 37:26.165 -> 30.5226,800,221104,0030,360000 +50,50
166 37:26.200 -> 31.80
167 37:26.235 -> 30.452,660,081104,0030,450000 +50,50
168 37:26.270 -> 31.84
169 37:26.305 -> 30.5233,050,231104,0030,360000 +50,50
170 37:26.340 -> 31.80
171 37:26.375 -> 30.452,660,081104,0030,450000 +50,50
172 37:26.410 -> 31.84
173 37:26.445 -> 30.5239,300,241104,0030,360000 +50,50
174 37:26.480 -> 31.80
175 37:26.515 -> 30.452,660,081104,0030,450000 +50,50
176 37:26.550 -> 31.84
177 37:26.585 -> 30.5245,550,251104,0030,360000 +50,50
178 37:26.620 -> 31.80
179 37:26.655 -> 30.452,660,081104,0030,450000 +50,50
180 37:26.690 -> 31.84
181 37:26.725 -> 30.5251,800,261104,0030,360000 +50,50
182 37:26.760 -> 31.80
183 37:26.795 -> 30.452,660,081104,0030,450000 +50,50
184 37:26.830 -> 31.84
185 37:26.865 -> 30.5258,050,271104,0030,360000 +50,50
186 37:26.900 -> 31.80
187 37:26.935 -> 30.452,660,081104,0030,450000 +50,50
188 37:26.970 -> 31.84
189 37:27.005 -> 30.5264,300,281104,0030,360000 +50,50
190 37:27.040 -> 31.80
191 37:27.075 -> 30.452,660,081104,0030,450000 +50,50
192 37:27.110 -> 31.84
193 37:27.145 -> 30.5270,550,291104,0030,360000 +50,50
194 37:27.180 -> 31.80
195 37:27.215 -> 30.452,660,081104,0030,450000 +50,50
196 37:27.250 -> 31.84
197 37:27.285 -> 30.5276,800,301104,0030,360000 +50,50
198 37:27.320 -> 31.80
199 37:27.355 -> 30.452,660,081104,0030,450000 +50,50
200 37:27.390 -> 31.84
201 37:27.425 -> 30.5283,050,311104,0030,360000 +50,50
202 37:27.460 -> 31.80
203 37:27.495 -> 30.452,660,081104,0030,450000 +50,50
204 37:27.530 -> 31.84
205 37:27.565 -> 30.5289,300,321104,0030,360000 +50,50
206 37:27.600 -> 31.80
207 37:27.635 -> 30.452,660,081104,0030,450000 +50,50
208 37:27.670 -> 31.84
209 37:27.705 -> 30.5295,550,331104,0030,360000 +50,50
210 37:27.740 -> 31.80
211 37:27.775 -> 30.452,660,081104,0030,450000 +50,50
212 37:27.810 -> 31.84
213 37:27.845 -> 30.5301,800,341104,0030,360000 +50,50
214 37:27.880 -> 31.80
215 37:27.915 -> 30.452,660,081104,0030,450000 +50,50
216 37:27.950 -> 31.84
217 37:27.985 -> 30.5308,050,351104,0030,360000 +50,50
218 37:28.020 -> 31.80
219 37:28.055 -> 30.452,660,081104,0030,450000 +50,50
220 37:28.090 -> 31.84
221 37:28.125 -> 30.5314,300,361104,0030,360000 +50,50
222 37:28.160 -> 31.80
223 37:28.195 -> 30.452,660,081104,0030,450000 +50,50
224 37:28.230 -> 31.84
225 37:28.265 -> 30.5320,550,371104,0030,360000 +50,50
226 37:28.300 -> 31.80
227 37:28.335 -> 30.452,660,081104,0030,450000 +50,50
228 37:28.370 -> 31.84
229 37:28.405 -> 30.5326,800,381104,0030,360000 +50,50
230 37:28.440 -> 31.80
231 37:28.475 -> 30.452,660,081104,0030,450000 +50,50
232 37:28.510 -> 31.84
233 37:28.545 -> 30.5333,050,391104,0030,360000 +50,50
234 37:28.580 -> 31.80
235 37:28.615 -> 30.452,660,081104,0030,450000 +50,50
236 37:28.650 -> 31.84
237 37:28.685 -> 30.5339,300,401104,0030,360000 +50,50
238 37:28.720 -> 31.80
239 37:28.755 -> 30.452,660,081104,0030,450000 +50,50
240 37:28.790 -> 31.84
241 37:28.825 -> 30.5345,550,411104,0030,360000 +50,50
242 37:28.860 -> 31.80
243 37:28.895 -> 30.452,660,081104,0030,450000 +50,50
244 37:28.930 -> 31.84
245 37:28.965 -> 30.5351,800,421104,0030,360000 +50,50
246 37:29.000 -> 31.80
247 37:29.035 -> 30.452,660,081104,0030,450000 +50,50
248 37:29.070 -> 31.84
249 37:29.105 -> 30.5358,050,431104,0030,360000 +50,50
250 37:29.140 -> 31.80
251 37:29.175 -> 30.452,660,081104,0030,450000 +50,50
252 37:29.210 -> 31.84
253 37:29.245 -> 30.5364,300,441104,0030,360000 +50,50
254 37:29.280 -> 31.80
255 37:29.315 -> 30.452,660,081104,0030,450000 +50,50
256 37:29.350 -> 31.84
257 37:29.385 -> 30.5370,550,451104,0030,360000 +50,50
258 37:29.420 -> 31.80
259 37:29.455 -> 30.452,660,081104,0030,450000 +50,50
260 37:29.490 -> 31.84
261 37:29.525 -> 30.5376,800,461104,0030,360000 +50,50
262 37:29.560 -> 31.80
263 37:29.595 -> 30.452,660,081104,0030,450000 +50,50
264 37:29.630 -> 31.84
265 37:29.665 -> 30.5383,050,471104,0030,360000 +50,50
266 37:29.700 -> 31.80
267 37:29.735 -> 30.452,660,081104,0030,450000 +50,50
268 37:29.770 -> 31.84
269 37:29.805 -> 30.5389,300,481104,0030,360000 +50,50
270 37:29.840 -> 31.80
271 37:29.875 -> 30.452,660,081104,0030,450000 +50,50
272 37:29.910 -> 31.84
273 37:29.945 -> 30.5395,550,491104,0030,360000 +50,50
274 37:29.980 -> 31.80
275 37:30.015 -> 30.452,660,081104,0030,450000 +50,50
276 37:30.050 -> 31.84
277 37:30.085 -> 30.5401,800,501104,0030,360000 +50,50
278 37:30.120 -> 31.80
279 37:30.155 -> 30.452,660,081104,0030,450000 +50,50
280 37:30.190 -> 31.84
281 37:30.225 -> 30.5408,050,511104,0030,360000 +50,50
282 37:30.260 -> 31.80
283 37:30.295 -> 30.452,660,081104,0030,450000 +50,50
284 37:30.330 -> 31.84
285 37:30.365 -> 30.5414,300,521104,0030,360000 +50,50
286 37:30.400 -> 31.80
287 37:30.435 -> 30.452,660,081104,0030,450000 +50,50
288 37:30.470 -> 31.84
289 37:30.505 -> 30.5420,550,531104,0030,360000 +50,50
290 37:30.540 -> 31.80
291 37:30.575 -> 30.452,660,081104,0030,450000 +50,50
292 37:30.610 -> 31.84
293 37:30.645 -> 30.5426,800,541104,0030,360000 +50,50
294 37:30.680 -> 31.80
295 37:30.715 -> 30.452,660,081104,0030,450000 +50,50
296 37:30.750 -> 31.84
297 37:30.785 -> 30.5433,050,551104,0030,360000 +50,50
298 37:30.820 -> 31.80
299 37:30.855 -> 30.452,660,081104,0030,450000 +50,50
300 37:30.890 -> 31.84
301 37:30.925 -> 30.5439,300,561104,0030,360000 +50,50
302 37:30.960 -> 31.80
303 37:30.995 -> 30.452,660,081104,0030,450000 +50,50
304 37:31.030 -> 31.84
305 37:31.065 -> 30.5445,550,571104,0030,360000 +50,50
306 37:31.100 -> 31.80
307 37:31.135 -> 30.452,660,081104,0030,450000 +50,50
308 37:31.170 -> 31.84
309 37:31.205 -> 30.5451,800,581104,0030,360000 +50,50
310 37:31.240 -> 31.80
311 37:31.275 -> 30.452,660,081104,0030,450000 +50,50
312 37:31.310 -> 31.84
313 37:31.345 -> 30.5458,050,591104,0030,360000 +50,50
314 37:31.380 -> 31.80
315 37:31.415 -> 30.452,660,081104,0030,450000 +50,50
316 37:31.450 -> 31.84
317 37:31.485 -> 30.5464,300,601104,0030,360000 +50,50
318 37:31.520 -> 31.80
319 37:31.555 -> 30.452,660,081104,0030,450000 +50,50
320 37:31.590 -> 31.84
321 37:31.625 -> 30.5470,550,611104,0030,360000 +50,50
322 37:31.660 -> 31.80
323 37:31.695 -> 30.452,660,081104,0030,450000 +50,50
324 37:31.730 -> 31.84
325 37:31.765 -> 30.5476,800,621104,0030,360000 +50,50
326 37:31.800 -> 31.80
327 37:31.835 -> 30.452,660,081104,0030,450000 +50,50
328 37:31.870 -> 31.84
329 37:31.905 -> 30.5483,050,631104,0030,360000 +50,50
330 37:31.940 -> 31.80
331 37:31.975 -> 30.452,660,081104,0030,450000 +50,50
332 37:32.010 -> 31.84
333 37:32.045 -> 30.5489,300,641104,0030,360000 +50,50
334 37:32.080 -> 31.80
335 37:32.115 -> 30.452,660,081104,0030,450000 +50,50
336 37:32.150 -> 31.84
337 37:32.185 -> 30.5495,550,651104,0030,360000 +50,50
338 37:32.220 -> 31.80
339 37:32.255 -> 30.452,660,081104,0030,450000 +50,50
340 37:32.290 -> 31.84
341 37:32.325 -> 30.5501,800,661104,0030,360000 +50,50
342 37:32.360 -> 31.80
343 37:32.395 -> 30.452,660,081104,0030,450000 +50,50
344 37:32.430 -> 31.84
345 37:32.465 -> 30.5508,050,671104,0030,360000 +50,50
346 37:32.500 -> 31.80
347 37:32.535 -> 30.452,660,081104,0030,450000 +50,50
348 37:32.570 -> 31.84
349 37:32.605 -> 30.5514,300,681104,0030,360000 +50,50
350 37:32.640 -> 31.80
351 37:32.675 -> 30.452,660,081104,0030,450000 +50,50
352 37:32.710 -> 31.84
353 37:32.745 -> 30.5520,550,691104,0030,360000 +50,50
354 37:32.780 -> 31.80
355 37:32.815 -> 30.452,660,081104,0030,450000 +50,50
356 37:32.850 -> 31.84
357 37:32.885 -> 30.5526,800,701104,0030,360000 +50,50
358 37:32.920 -> 31.80
359 37:32.955 -> 30.452,660,081104,0030,450000 +50,50
360 37:32.990 -> 31.84
361 37:33.025 -> 30.5533,050,711104,0030,360000 +50,50
362 37:33.060 -> 31.80
363 37:33.095 -> 30.452,660,081104,0030,450000 +50,50
364 37:33.130 -> 31.84
365 37:33.165 -> 30.5539,300,721104,0030,360000 +50,50
366 37:33.200 -> 31.80
367 37:33.235 -> 30.452,660,081104,0030,450000 +50,50
368 37:33.270 -> 31.84
369 37:33.305 -> 30.5545,550,731104,0030,360000 +50,50
370 37:33.340 -> 31.80
371 37:33.375 -> 30.452,660,081104,0030,450000 +50,50
372 37:33.410 -> 31.84
373 37:33.445 -> 30.5551,800,741104,0030,360000 +50,50
374 37:33.480 -> 31.80
375 37:33.515 -> 30.452,660,081104,0030,450000 +50,50
376 37:33.550 -> 31.84
377 37:33.585 -> 30.5558,050,751104,0030,360000 +50,50
378 37:33.620 -> 31.80
379 37:33.655 -> 30.452,660,081104,0030,450000 +50,50
380 37:33.690 -> 31.84
381 37:33.725 -> 30.5564,300,761104,0030,360000 +50,50
382 37:33.760 -> 31.80
383 37:33.795 -> 30.452,660,081104,0030,450000 +50,50
384 37:33.830 -> 31.84
385 37:33.865 -> 30.5570,550,771104,0030,360000 +50,50
386 37:33.900 -> 31.80
387 37:33.935 -> 30.452,660,081104,0030,450000 +50,50
388 37:33.970 -> 31.84
389 37:34.005 -> 30.5576,800,781104,0030,360000 +50,50
390 37:34.040 -> 31.80
391 37:34.075 -> 30.452,660,081104,0030,450000 +50,50
392 37:34.110 -> 31.84
393 37:34.145 -> 30.5583,050,791104,0030,360000 +50,50
394 37:34.180 -> 31.80
395 37:34.215 -> 30.452,660,081104,0030,450000 +50,50
396 37:34.250 -> 31.84
397 37:34.285 -> 30.5589,300,801104,0030,360000 +50,50
398 37:34.320 -> 31.80
399 37:34.355 -> 30.452,660,081104,0030,450000 +50,50
400 37:34.390 -> 31.84
401 37:34.425 -> 30.5595,550,811104,0030,360000 +50,50
402 37:34.460 -> 31.80
403 37:34.495 -> 30.452,660,081104,0030,450000 +50,50
404 37:34.530 -> 31.84
405 37:34.565 -> 30.5601,800,821104,0030,360000 +50,50
406 37:34.600 -> 31.80
407 37:34.635 -> 30.452,660,081104,0030,450000 +50,50
408 37:34.670 -> 31.84
```

จากผลการทดลองในส่วนของการเปรียบเทียบมุมของอุปกรณ์และมุมที่ได้เซนเซอร์สามารถสรุปได้ตามตารางดังนี้

ตารางที่ 1 ตารางสรุปผลค่าความคลาดเคลื่อนที่เกิดจากโมดูล GY-521

ขนาดของมุมที่เปรียบเทียบ	ขนาดมุมของอุปกรณ์	ค่าเฉลี่ยของมุมที่ได้จากเซนเซอร์	ค่าความคลาดเคลื่อน (%)
-10	-10	-11.667	16.67
-20	-20	-20.886	4.43
-30	-30	-29.141	2.86
0	0	0.061	0.06
10	10	10.674	6.74
20	20	20.673	3.36
30	30	30.685	2.28

จากตารางแสดงผลดังกล่าวแสดงให้เห็นค่าความคลาดเคลื่อนหรือความผิดพลาดในการคำนวณค่ามุมของอุปกรณ์ ซึ่งจากตารางแสดงให้เห็นความคลาดเคลื่อน ตั้งแต่ 0.06–16.67 เปอร์เซ็นต์ ซึ่งถือว่าเป็นความคลาดเคลื่อนที่สูงมากแต่หากเทียบเป็น ค่ามุมของระนาบจะเห็นว่าอุปกรณ์มีความคลาดเคลื่อนประมาณ 1-2 องศา ซึ่งเป็นค่าที่ไม่สามารถแยกได้ด้วยตาเปล่าหรือมีความแตกต่างกันน้อยมาก

4.1.2 ผลการทดสอบการทำงานของชุดมอเตอร์ และ Electronic speed controller

ผลของการทดสอบอุปกรณ์มอเตอร์ และ Electronic speed controller โดยอุปกรณ์ที่ทดสอบทั้งหมดคือมอเตอร์ 4 ตัวและ Electronic speed controller 4 ตัวโดยผลการทดสอบพบว่ามอเตอร์หมายเลข 3 และ 4 มีประสิทธิภาพในการทำงานดีกว่า 1 และ 2 และ Electronic speed controller หมายเลข 2 และ 4 มีประสิทธิภาพในการทำงานดีกว่า 1 และ 3 เป็นเหตุผลในการเลือกใช้มอเตอร์หมายเลข 2 และ 4 และใช้ Electronic speed controller หมายเลข 1 และ 3

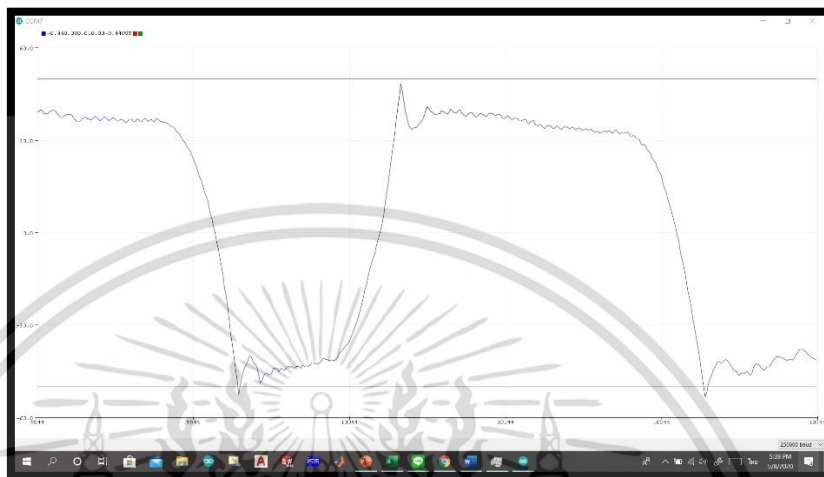
ตารางที่ 2 ประสิทธิภาพในการทำงานของมอเตอร์และ Electronic speed controller

ESC. 1	RPM. (1200us)	ESC. 2	RPM. (1200us)	ESC. 3	RPM. (1200us)	ESC. 4	RPM. (1200us)
M1	2821	M1	3437	M1	3343	M1	3386
M2	2850	M2	3437	M2	3352	M2	3375
M3	2984	M3	3558	M3	3515	M3	3572
M4	2924	M4	3521	M4	3425	M4	3492

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลการทดสอบการทำงานของระบบ

ในส่วนของการทำสอบการทำงานของระบบจะทดสอบด้วยการปรับค่าพารามิเตอร์ที่ใช้ในการควบคุมระบบการทำงานแบ่งเป็น ค่าเป้าหมาย K_p K_i K_d โดยการทดลองของระบบจะเริ่มจากการเปิดเครื่องและสังเกตการทำงานของอุปกรณ์เมื่อตั้งค่าเป้าหมายที่ 0 องศา

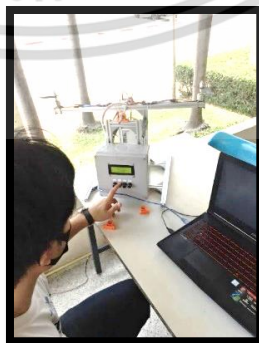


รูปที่ 4.16 กราฟแสดงการทำงานของระบบเมื่อเปิดอุปกรณ์

จากรูปที่ 4.16 แสดงให้เห็นถึงการทำงานของระบบเมื่อเปิดเครื่องหรือเมื่อยังไม่ได้ใส่ค่าคงที่ในการควบคุมระบบ จากกราฟจะเห็นว่า การทำงานของระบบไม่สามารถเข้าสู่ค่าเป้าหมายได้ และการทำงานของระบบยังไม่มีเสถียรภาพ

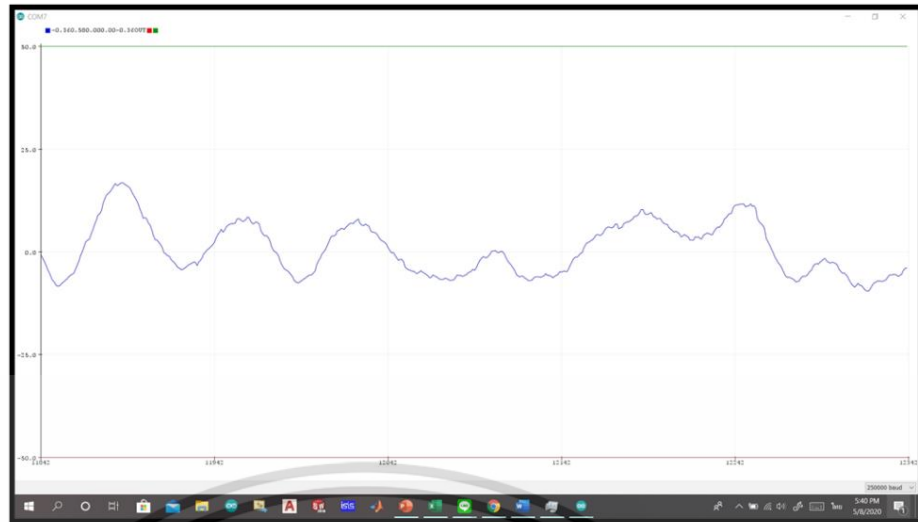
4.2.1 ผลการทดสอบการทำงานของระบบด้วยการปรับค่า K_p

ในการทดสอบการใส่ค่า K_p ให้กับระบบผู้จัดทำได้กำหนดให้ค่าเป้าหมายเท่ากับ 0 องศา เพื่อให้สามารถสังเกตการทำงานของระบบได้ง่ายโดยการใส่ค่า K_p จะเพิ่มจนวนรอบของอุปกรณ์สวิงอยู่ระหว่างค่าเป้าหมาย



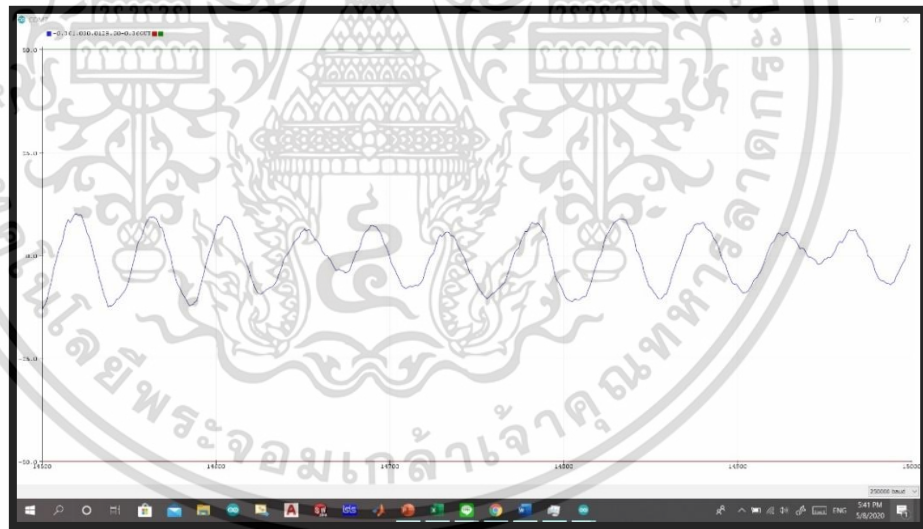
รูปที่ 4.17 วิธีการในการปรับค่า K_p ให้กับระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 กราฟแสดงการทำงานของระบบเมื่อปรับค่า $K_p = 0.5$

จากรูปที่ 4.18 แสดงให้เห็นถึงการทำงานของระบบเมื่อใส่ค่า $K_p = 0.5$ ให้กับระบบ จากภาพจะสังเกตเห็นว่าระบบเริ่มพยายามเข้าสู่ค่าเป้าหมายโดยการสวิตช์ระหว่างค่าเป้าหมาย แต่การสวิตช์ยังไม่คงที่แสดงให้เห็นว่า เสถียรภาพของระบบต่ำ



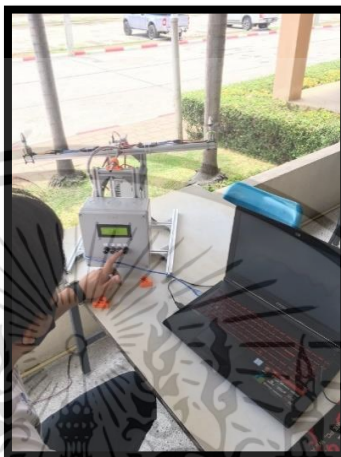
รูปที่ 4.19 กราฟแสดงการทำงานของระบบเมื่อปรับค่า $K_p = 1.0$

จากรูปที่ 4.19 แสดงให้เห็นว่าระบบมีเสถียรภาพที่ดีขึ้นหลังจากปรับค่า K_p เป็น 1.0 สังเกตได้จากการเปลี่ยนแปลงของกราฟซึ่งมีลักษณะในการทำงานคงที่มากขึ้นแม้ว่าจะสวิง

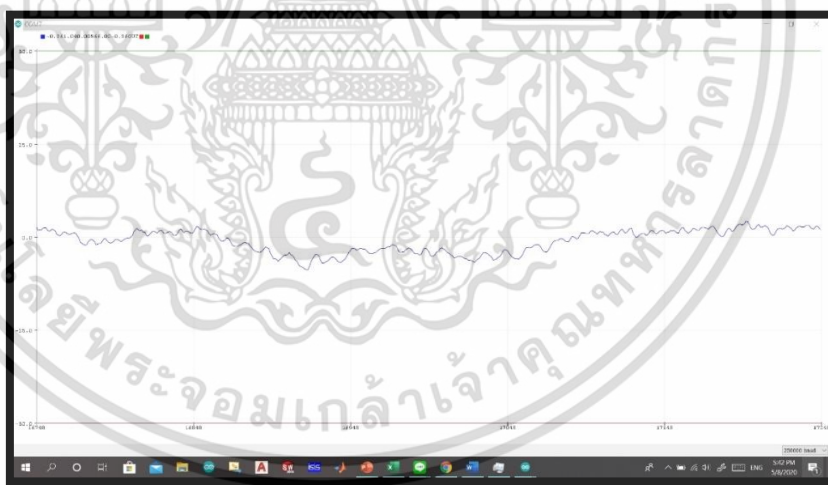
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 ผลการทดสอบการทำงานของระบบด้วยการปรับค่า Kd

เมื่อระนาบของคานเริ่มจะสวิงระหว่างค่าเป้าหมายแสดงให้เห็นว่าระบบเริ่มจะพยายามเข้าสู่ค่าเป้าหมาย เมื่อใส่ค่า Kd เข้าไปในระบบจะทำให้ระบบเข้าสู่ค่าเป้าหมายได้เร็วยิ่งขึ้นหรือช่วยลดการสวิงของระบบได้เนื่องจาก Kd จะชะลออัตราการเปลี่ยนแปลงของสัญญาณขาออกของระบบด้วยเหตุนี้ส่งผลให้ระบบเข้าสู่สมดุล



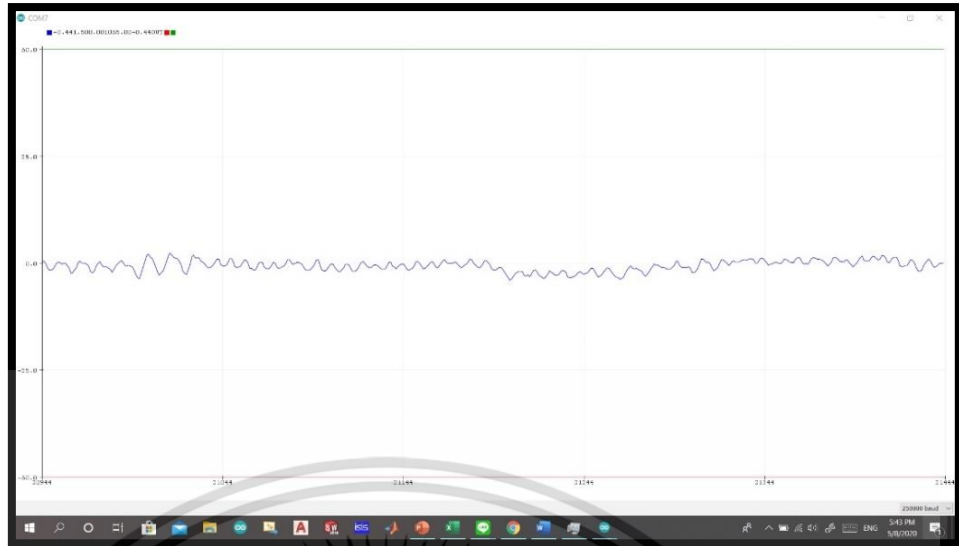
รูปที่ 4.20 วิธีการในการปรับค่า Kd ให้กับระบบ



รูปที่ 4.21 กราฟแสดงการทำงานของระบบเมื่อปรับค่า $K_d = 0.5$

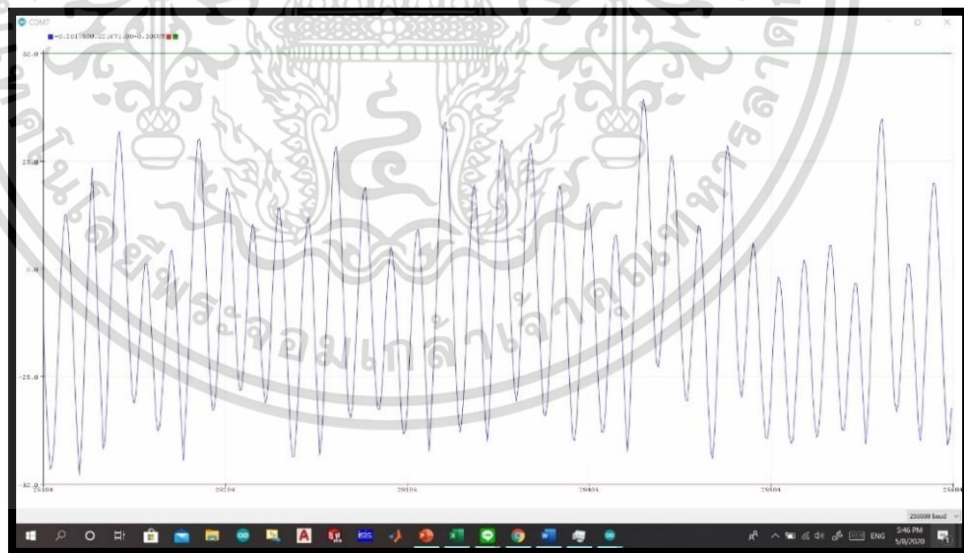
จากรูปที่ 4.21 แสดงให้เห็นการทำงานของระบบเมื่อเพิ่มค่า Kd ให้กับอุปกรณ์จากภาพจะสังเกตเห็นได้ว่าการทำงานของอุปกรณ์จะมีการสวิงที่ลดลงเนื่องจากค่า Kd จะชะลออัตราการเปลี่ยนแปลงของสัญญาณขาออกด้วยเหตุนี้ช่วยให้อุปกรณ์จะเข้าสู่ค่าเป้าหมาย แต่การใส่ค่า Kd ให้กับอุปกรณ์มากเกินไปจะส่งผลให้การทำงานของอุปกรณ์มีการตอบสนองไวมากต่อความผิดพลาดและสามารถทำให้การทำงานไม่เสถียรหรืออาจจะทำให้เกิดค่าพุ่งเกินได้ดังรูปที่ 4.23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 กราฟแสดงลักษณะการทำงานของระบบเมื่อปรับค่า $K_d = 1.0$

จากรูปที่ 4.22 จะสามารถสังเกตเห็นได้ว่าลักษณะการทำงานของระบบจะสวีน้อยกว่าการใส่ค่าคงที่ K_p เพียงอย่างเดียวในระบบ แต่ในการเพิ่มค่าคงที่ K_d ให้กับระบบ ถ้าใส่มากเกินไปจะส่งผลทำให้การทำงานของระบบเกิดโอเวอร์ชูต และจะไม่เข้าสู่ค่าเป้าหมาย และ ไม่สามารถทำให้ระบบเสถียรได้ดังรูปที่ 4.23

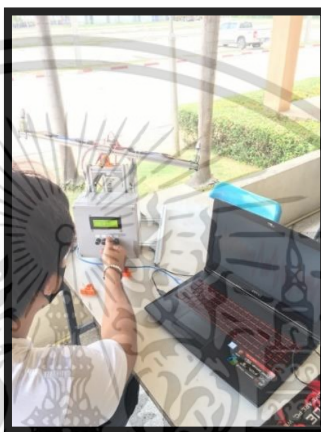


รูปที่ 4.23 กราฟแสดงการทำงานของระบบเมื่อเกิดค่าพุ่งเกิน

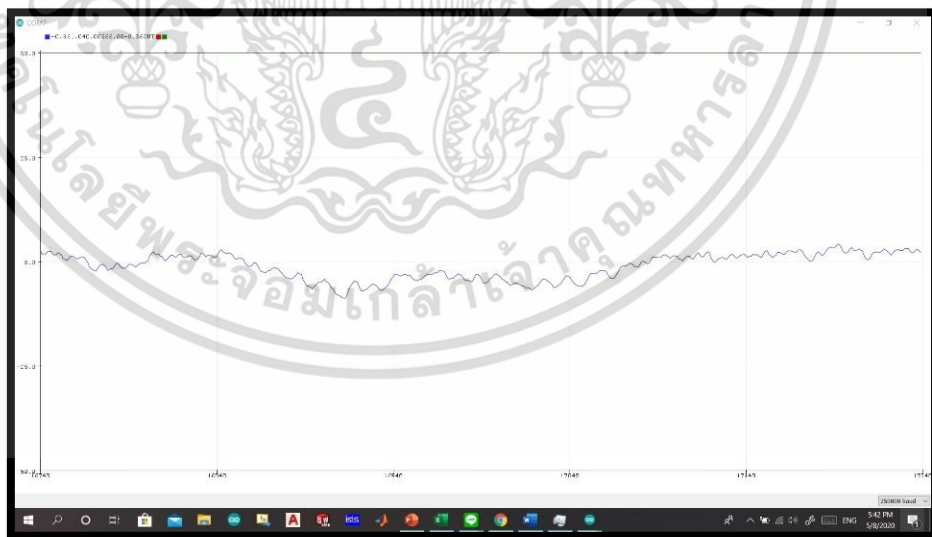
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 ผลการทดสอบการทำงานของระบบด้วยการปรับค่า Ki

จากการทำงานของระบบเมื่อใส่ค่า K_p และ K_d ในการควบคุมระบบการทำงานแล้วพบปัญหาคือระบบของอุปกรณ์ไม่สามารถเข้าสู่ค่าเป้าหมายได้แต่การทำงานของระบบมีความเสถียร ปัญหานี้เกิดจาก ออฟเซต ของอุปกรณ์ ดังนั้นผู้จัดทำจึงใส่ค่าคงที่ K_i เข้าไปในระบบเพื่อแก้ไขในส่วนของออฟเซตแต่ในส่วนของการเพิ่มค่า K_i ให้กับอุปกรณ์ จะเร่งการเข้าสู่ค่าเป้าหมายและขจัดความผิดพลาดที่เกิดจากการใช้ K_p แต่อย่างไรก็ตาม ค่า K_i จะตอบสนองต่อค่าความผิดพลาดสะสมจึงสามารถทำให้การทำงานเกิดโอเวอร์ชูตได้

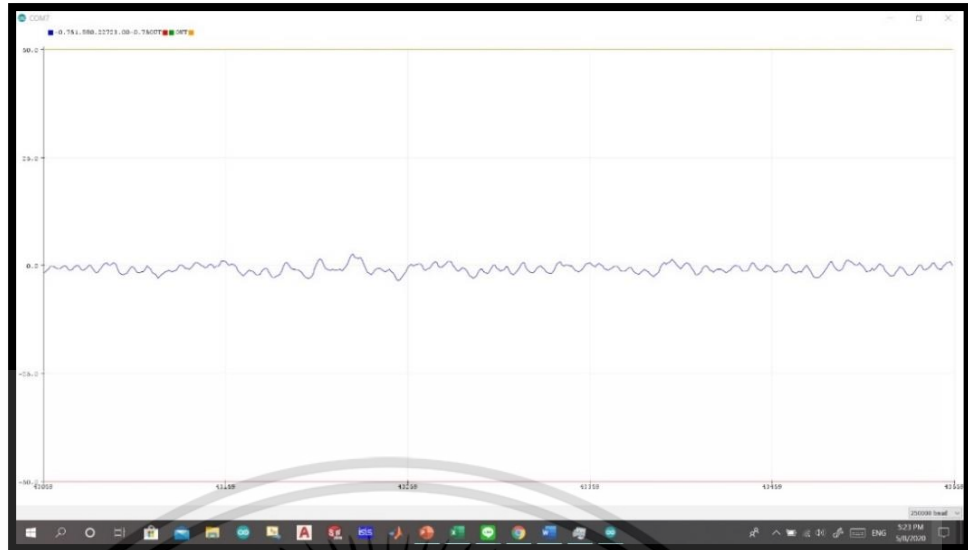


รูปที่ 4.24 วิธีการในการเพิ่มค่า K_i ให้กับระบบ



รูปที่ 4.25 กราฟแสดงระบบการทำงานของอุปกรณ์ขณะเกิดออฟเซต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

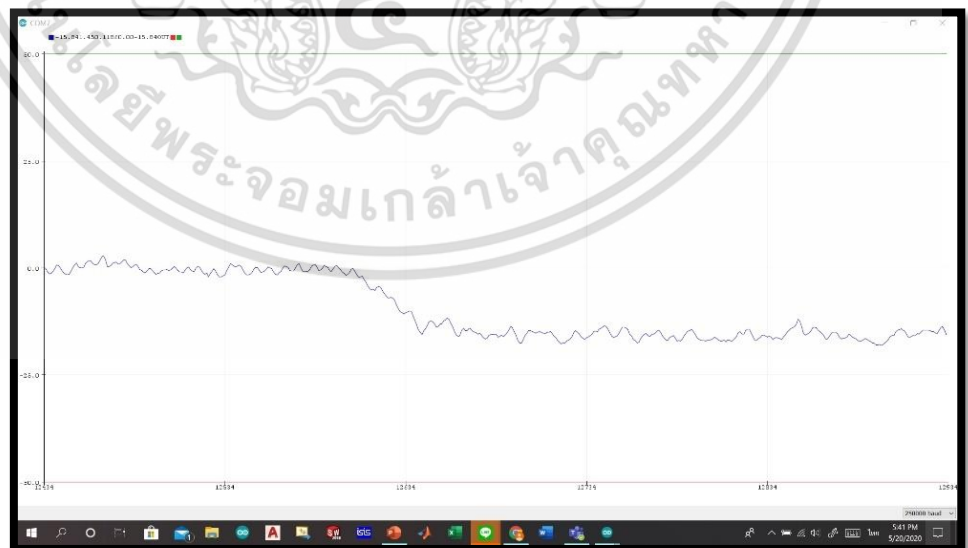


รูปที่ 4.26 กราฟแสดงการทำงานของระบบเมื่อเพิ่มค่า $K_i = 0.3$

4.2.4 ผลการทดสอบการทำงานด้วยการปรับค่าเป้าหมาย

ในการทดสอบระบบด้วยการเพิ่มค่าเป้าหมายจะเริ่มจากการกำหนดค่าเป้าหมาย ให้เท่ากับ 0 องศา เปลี่ยนเป็น -15 องศา และเปลี่ยนจาก -15 องศา ถึง 0 องศา และ 0 องศา ถึง 15 องศา แบบ ขึ้นบันไดเพื่อให้สามารถสังเกตการทำงานของระบบจากกราฟได้

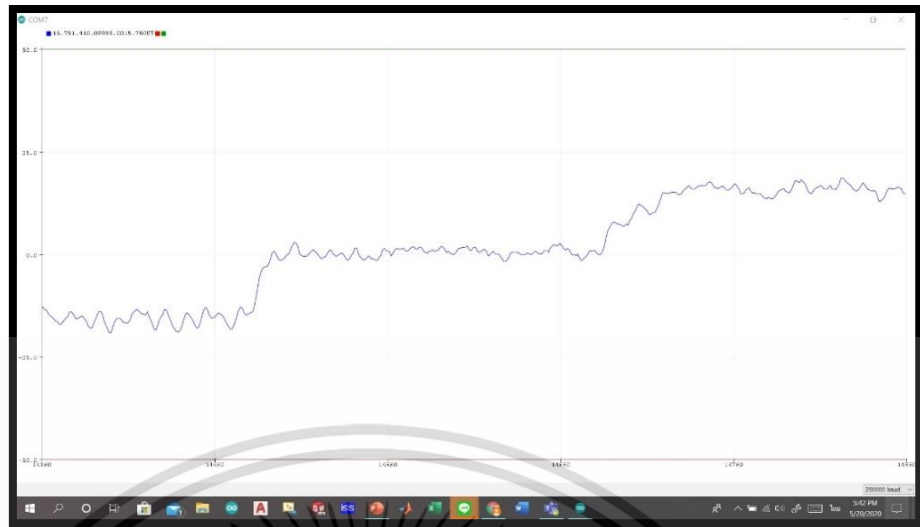
การปรับค่าเป้าหมายของอุปกรณ์จาก 0 องศาถึง -15 องศาและ -15 องศา ถึง 15 องศา จากกราฟการทำงานของอุปกรณ์ในขณะที่เปลี่ยนค่าเป้าหมาย หรือมุ่มอ้างอิงจะเห็นว่าการทำงานของระบบยังสามารถทำงานได้ตามปกติ



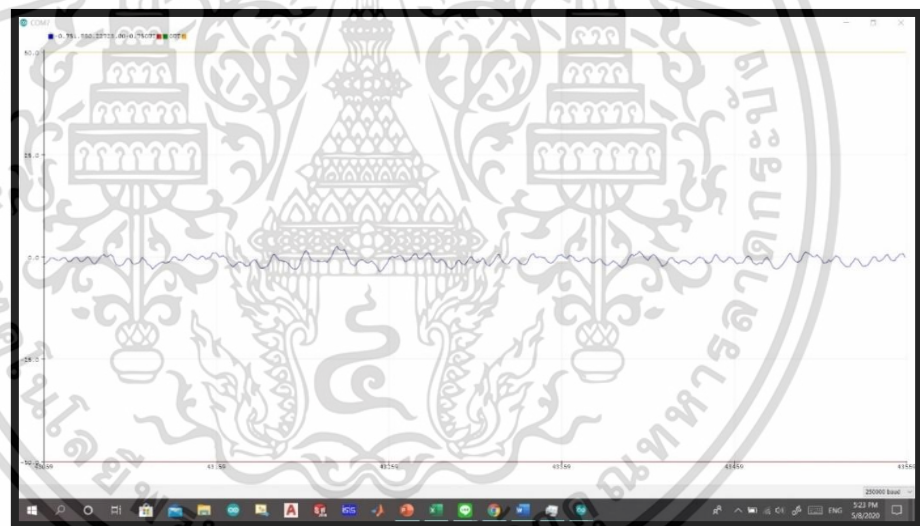
รูปที่ 4.27 กราฟแสดงการทำงานของระบบเมื่อค่าเป้าหมาย

เปลี่ยนจาก 0 องศา ถึง -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

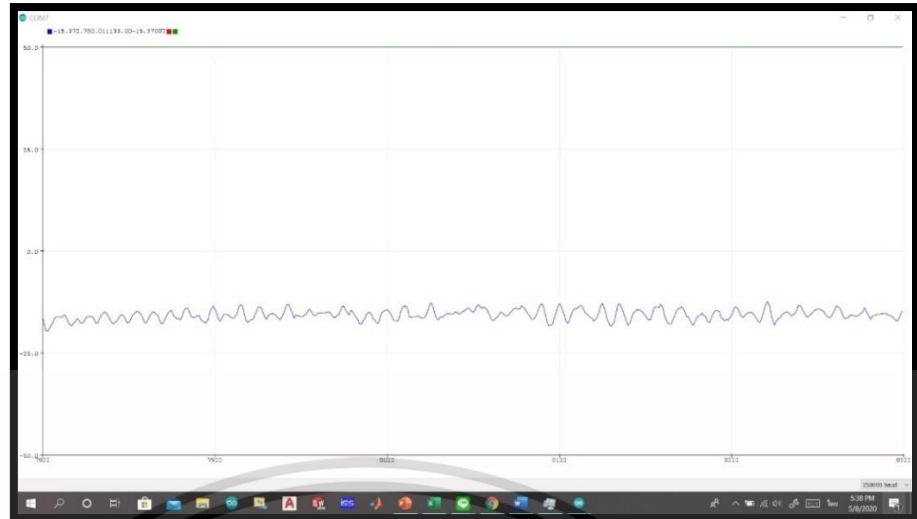


รูปที่ 4.28 กราฟแสดงการทำงานของระบบเมื่อค่าเป้าหมาย
เปลี่ยนจาก -15 องศาถึง 0 องศา และ 0 องศา ถึง 15 องศา

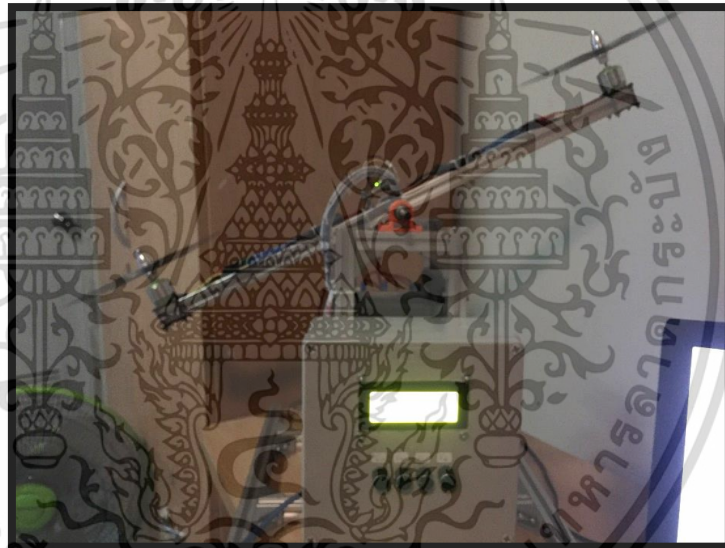


รูปที่ 4.29 กราฟแสดงการทำงานของระบบเมื่อค่าเป้าหมายเป็น 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.30 กราฟแสดงการทำงานของระบบเมื่อค่าเป้าหมายเป็น -15 องศา



รูปที่ 4.31 ลักษณะของระบบเมื่อค่าเป้าหมายเป็น -15 องศา



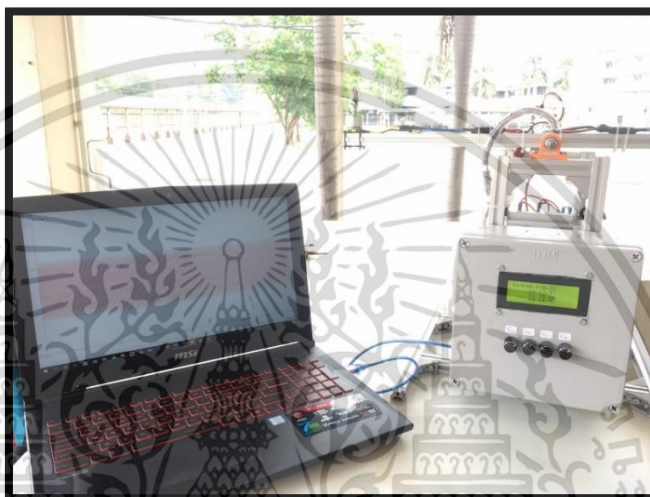
รูปที่ 4.32 จอแสดงผลขณะกำหนดค่าเป้าหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.32 ตัวชิ้นงานได้ถูกกำหนดค่าเป้าหมายอยู่ที่ -15 องศา จึงทำให้ระบบการทำงานมีมุมอ้างอิงเป็น -15 องศา หรือ ต้องรักษาให้ระนาบของคานอยู่ที่ -15 องศา

4.3 ผลการประดิษฐ์ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี

ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดีจะประกอบไปด้วยโมดูลเซนเซอร์ (โมดูล GY-521) กล้องควบคุม และชุดมอเตอร์สองชุด โดยกล้องควบคุมจะประกอบไปด้วยจอแสดงผล LCD ตัวต้านทานปรับค่าได้เพื่อใช้กำหนดค่าของ พีไอดี แก่ระบบ และบอร์ด Arduino



รูปที่ 4.33 ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี

ในการทำงานจำเป็นต้องกำหนดค่าเป้าหมาย K_p K_i และ K_d ให้กับระบบซึ่งแสดงให้เห็นผ่านทางจอแสดงผลโดยสามารถกำหนดค่าด้วยตัวต้านทานปรับค่าได้ที่ติดอยู่ที่กล้องควบคุม จำนวนสี่อันซึ่งค่าคงที่แต่ละชนิดมีการทำงานที่แตกต่างกันและส่งผลกระทบต่อการทำงานของระบบโดยตรง



รูปที่ 4.34 หน้าจอแสดงผลค่าพารามิเตอร์ที่ใช้ควบคุมระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการดำเนินงานตามขั้นตอนข้างต้นโดยเริ่มจากตั้งแต่ขั้นตอนของการศึกษาทฤษฎีและหลักการ พร้อมทั้งตั้งขอบเขตของโครงการ จึงแบ่งการสรุปการดำเนินงานเป็น 2 ส่วนดังนี้

1. ในส่วนโครงสร้างและอุปกรณ์ การสร้างระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี ในส่วนนี้ได้ทำการศึกษาหลักการการทำงานของเซนเซอร์ที่ใช้และอุปกรณ์ไมโครคอนโทรลเลอร์ที่ครอบคลุมการทำงานของระบบต่างๆ ที่ได้กล่าวมาข้างต้นรวมถึงการออกแบบให้อุปกรณ์สามารถควบคุมการทำงานได้ง่าย

2. การเขียนโปรแกรม เป็นส่วนที่เขียนโค้ดเพื่อรับค่าจากเซนเซอร์เพื่อรับข้อมูล และนำไปประมวลผลด้วยไมโครคอนโทรลเลอร์ และการเขียนโปรแกรมเพื่อควบคุมการทำงานของอุปกรณ์ โดยปรับค่าตัวต้านทานจากภายนอก และแสดงผลผ่านหน้าจออุปกรณ์ LCD

หลังจากที่ได้ทำการศึกษาข้อมูลต่างๆ และสร้างชุดระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี จึงนำอุปกรณ์มาทดสอบระบบการทำงาน โดยพบว่าอุปกรณ์สามารถควบคุมการทำงานได้โดยปรับค่าตัวต้านทานซึ่งควบคุม ค่าเป้าหมาย K_p K_i และ K_d โดยแสดงผลผ่านจอ LCD ได้ทั้งนี้พบว่าการทำงานของระบบเป็นไปโดยราบรื่นตามความคาดหมายแต่ยังมีข้อบกพร่องคือการทำงานของไมโครคอนโทรลเลอร์ขณะเปิดใช้งานอุปกรณ์ไม่สามารถประมวลผลทำให้จำเป็นต้องทำการรีเซ็ตไมโครคอนโทรลเลอร์ทุกครั้งเวลาเปิดใช้งานอุปกรณ์ และ ความคลาดเคลื่อนของอุปกรณ์ที่เกิดจากตำแหน่งที่ติดตั้งของเซนเซอร์ ทั้งนี้ระบบคานสมดุลโดยใช้ตัวควบคุมพีไอดี ยังเป็นเพียงต้นแบบในอนาคตสามารถแก้ไขหรือต่อยอดให้อุปกรณ์มีระบบการทำงานที่ดีและการควบคุมที่สะดวกมากยิ่งขึ้น

5.2 ปัญหาและอุปสรรคในการดำเนินงาน

1. เนื่องจากในการทำโครงการเรื่องนี้เป็นการทำความเข้าใจกับระบบควบคุมแบบพีไอดี ซึ่งผู้จัดทำมีความรู้ในส่วนนี้ค่อนข้างน้อยจึงทำให้ใช้เวลาในการทบทวนค่อนข้างมาก

2. การศึกษาการเขียนโปรแกรมและการเขียนโปรแกรมที่ใช้ในการทำงานของระบบ มีความล่าช้าเป็นอย่างมากเนื่องจากผู้ทำขาดความชำนาญในการเขียนโปรแกรม Arduino และ ภาษา C

3. ขาดความชำนาญในการประกอบกล่องควบคุมระบบส่งผลให้เกิดข้อผิดพลาดหลายจุดที่ทำให้กล่องขาดความสวยงาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางแก้ไขปัญหา

1. ศึกษาและสอบถามแนวทางเพิ่มเติมจากสื่อการสอนต่างๆ รวมถึงอาจารย์ที่ปรึกษาและอาจารย์ท่านอื่นๆ
2. ศึกษาโค้ดที่ใช้ในการทำงานของระบบผ่านสื่อการสอนหลายวิธีเช่น Youtube Google หรือ Arduino.com รวมถึงการสอบถามอาจารย์ที่ปรึกษา และเพื่อนๆ ในสาขาวิชา
3. เตรียมอุปกรณ์สำรองเพื่อแก้ไขหากเกิดความผิดพลาด

5.4 ข้อเสนอแนะ

เนื่องจากการทำงานของระบบต้องอาศัยการประมวลผลของโมดูล Arduino UNO ซึ่งเมื่อเปิดเครื่องอุปกรณ์อาจจะไม่สามารถทำงานได้เนื่องจากการทำงาน ที่ผิดพลาดจึงทำให้ต้องกดปุ่มรีเซ็ตที่อยู่ด้านหลังของกล่องควบคุม และในการทำงานของระบบสามารถต่อยอดเพื่อให้อุปกรณ์สามารถทำงานได้อย่างมีประสิทธิภาพ โดยการเขียนโค้ดให้การทำงานของระบบมีความเสถียรมากยิ่งขึ้น และอุปกรณ์บ้างชิ้นสร้างขึ้นด้วยเครื่องพิมพ์ 3 มิติ ซึ่งอายุในการใช้งานและความคงทนของชิ้นงานที่สร้างขึ้นไม่สูงมากและอาจทำให้อุปกรณ์เสียหายได้ ดังนั้นอาจมีการหาอุปกรณ์ที่มีอายุการใช้งานและความคงทนสูงมาใช้งานเพื่อแก้ปัญหาส่วนนี้

เอกสารอ้างอิง

- [1] “หลักการทํางานของ โมดูล GY-521” เข้าถึงได้จาก : <http://xtcomp.co.za/image/catalog/โมดูล GY-521. pdf>
- [2] “หลักการทํางานของ Electronic speed controller” เข้าถึงได้จาก : <https://manualzz.com/doc/4193601/user-manual-of-ae-20a%E3%80%81-25a%E3%80%81-30a-มอเตอร์ไร้แปรงถ่าน-speed-controller>
- [3] “หลักการทํางานของ Brushless Motor” เข้าถึงได้จาก : <https://arduino-shop.eu/docs/produkty/0/118/1496222410.pdf>
- [4] “หลักการทํางานของ LCD” เข้าถึงได้จาก : <https://www.arduinoall.com/product/468/%E0%B9%82%E0%B8%A1%E0%B8%94%E0%B8%B9%E0%B8%A5%E0%B8%88%E0%B8%AD-lcd-2004-lcd-blue-screen-20x4-lcd-with-backlight-%E0%B9%84%E0%B8%A1%E0%B9%88%E0%B8%A1%E0%B8%B5-i2c>
- [5] “ข้อมูลของระบบควบคุมพีไอดี” เข้าถึงได้จาก : <https://th.wikipedia.org/wiki/%E0%B8%A3%E0%B8%B0%E0%B8%9A%E0%B8%9A%E0%B8%84%E0%B8%A7%E0%B8%9A%E0%B8%84%E0%B8%B8%E0%B8%A1%E0%B8%9E%E0%B8%B5%E0%B9%84%E0%B8%AD%E0%B8%94%E0%B8%B5>
- [6] “ข้อมูลของแรงยก ” เข้าถึงได้จาก : <https://th.wikipedia.org/wiki/%E0%B9%81%E0%B8%A3%E0%B8%87%E0%B8%A2%E0%B8%81%E0%B8%97%E0%B8%B2%E0%B8%87%E0%B8%AD%E0%B8%B2%E0%B8%81%E0%B8%B2%E0%B8%A8%E0%B8%9E%E0%B8%A5%E0%B8%A8%E0%B8%B2%E0%B8%AA%E0%B8%95%E0%B8%A3%E0%B9%8C>
- [7] “ข้อมูล Complementary Filter” เข้าถึงได้จาก : https://industrialold.uru.ac.th/dbproject /pdf/chapter 2_1424226881.pdf



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

โมดูล GY-521



SYNACORP TRADING & SERVICES

No.9, 1st Floor, Lrg 1/SS2, Bandar Tasek Mutiara, 14120 Simpang Ampat, S.Prai (S), Penang.
Tel : +604.504.1617 Hunting Line : 012.4033.474 Fax : +604.502.1726
(Website) <http://www.synacorp.my> (Email) sales@synacorp.com.my

MPU 6050 GY-521 3 Axis Gyro Accelerometer Sensor Module Arduino



The MPU-6050 sensor module contains an accelerometer and a gyro in a single chip. It is very accurate, as it contains 16-bits analog to digital conversion hardware for each channel. Therefore it captures the x, y, and z channel at the same time. The sensor uses the I2C-bus to interface with the Arduino.

Application:

- Motion-enabled game and application framework
- Location based services, points of interest
- Handset and portable gaming
- Motion-based game controllers
- Wearable sensors for health, fitness and sports
- Toys

Features:

- Use the chip: MPU-6050.
- Power supply: 3-5v (internal low dropout regulator).
- Communication modes: standard IIC communications protocol.
- Chip built-in 16bit AD converter, 16-bit data output.
- Immersion Gold PCB machine welding process to ensure quality.
- Tri-Axis angular rate sensor (gyro) with a sensitivity up to 131 LSBs/dps and a full-scale range of ± 250 , ± 500 , ± 1000 , and ± 2000 dps
- Tri-Axis accelerometer with a programmable full scale range of $\pm 2g$, $\pm 4g$, $\pm 8g$ and $\pm 16g$

For purchase & enquiries, please contact sales@synacorp.com.my or call 04-5041617

รูปที่ ก.1 คุณสมบัติของ GY - 521

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

Electronic speed controller

User Manual of AE-20A、 25A、 30A Electronic Speed Controller (ESC)

Thanks for purchasing our Electronic Speed Controller (ESC). High power system for RC model can be very dangerous, so we strongly suggest you read this manual carefully. In that we have no control over the correct use, installation, application, or maintenance of our products, no liability shall be assumed nor accepted for any damages, losses or costs resulting from the use of the product. Any claims arising from the operating, failure or malfunctioning etc. will be denied. We assume no liability for personal injury, property damage or consequential damages resulting from our product or our workmanship. As far as is legally permitted, the obligation to compensation is limited to the invoice amount of the affected product.

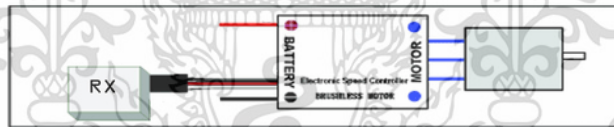
Features :

- High performance microprocessor brings out the best compatibility with all kinds of motors and the higher driving efficiency, start smoothly, suitable for most brushless motor on market.
- Imported low impedance capacitors, providing good anti-jamming capability, greatly reduce the possibility the loss of control.
- Supported motor speed (Maximum): 19000rpm(2 poles), 6300rpm (6 poles), 3400rpm(14poles)
- Throttle range can be configured to be compatible with all transmitters currently available on market Smooth, linear, quick and precise throttle response.
- With Super-Soft and Governor Mode for helicopter, compatible with Fixed-wing aircraft and Helicopter.
- With protected over-temperature function, ensure the safe of using ESC and providing good anti-jamming capability.
- With the design of cooling fins, decrease using temperature.
- The size is small , proper and easy for installation.

Direction for use:

- The range of input voltage: 5.5V-16.8V(Lipo 2-4cells/Nimh 6-12cells). Don't surpass voltage to use ,otherwise it will break the product.
- The power should be connected in correct way .Red is anode and black is cathode.
- Setting for high throttle response speed, current will be increased in expedite instantaneous.
- With continuous heavy current working, it needs the heat output of ESC goes better.

Wiring Diagram : Attention to the anode and cathode of the power and connection the power for testing after checked, the wrong connection will break the controller.



For normal start process direction:

Start transmitters and move the throttle stick to the bottom position	Connection the ESC to the battery, a special tone 123.....(ding-ding-ding), that means the connection is ok.	Start throttle ,the motor starting	If haven't move the throttle, the ESC will emit the tone begin to report the function of setted menu	Default: 1.Ding (No brake) 2.Ding-Ding (Medium Timing) 3.Ding-Ding(Medium cut-off threshold) 4.Ding (Fixed-Wing) 5.Ding (Normal throttle response)
---	--	------------------------------------	--	---

After checking the connection is ok, start transmitters and move the throttle stick to the bottom position, then the ESC connection the power.

After connection the ESC to the motor, a special tone "1 2 3" — " (ding - ding - ding) " , that means the self-testing

รูปที่ ข.1 คุณสมบัติ Electronic speed controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค
มอเตอร์ไร้แปรงถ่าน

ECLIPSERA
MODULES

1496222410
VÝROBNÍ ČÍSLO


BLDC motor outrunnerový 1400 kV A2212/10T

1. POPIS

Univerzální BLDC motor s rotujícím pláštěm (tzv. outrunner). Svě uplatnění najde především v modelářství, při výrobě hraček a v domácí automatizaci.

Základní charakteristika motoru:

- vhodné pro modely o hmotnosti 300 až 800 g
- napěťová konstanta Kv: 1400 ot./V
- lehká konstrukce - hmotnost 47 g
- účinnost 78 %



2. SPECIFIKACE

Typ	A2212/10T	Počet Li-poi článků	2 až 3
Napájecí napětí	7 až 12 V	Počet NiCd/NiMH	6 až 10
Max. špičkový proud (60 s)	16 A	Průměr hřídele	3,17 mm
Přikón	180 W	Pro modely o hmotnosti cca	300 až 800 g
El. účinnost	78 %	Hmotnost motoru	47 g
napěťová konstanta Kv	1400 ot./V	Rozměry (mm)	27,5 x 30

Pozn. 1: Údaj udává, kolik sítiově zapojených článků je možné pro napájení motoru (resp. řídicí elektroniky) použít.

3. ZAPOJENÍ

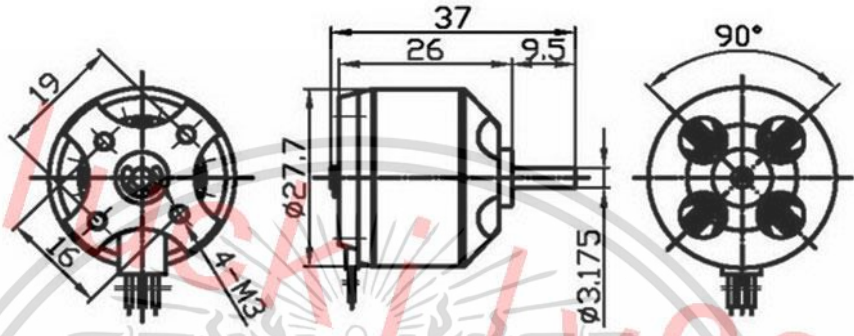
Upozornění: Motor nelze provozovat samostatně bez řídicí elektroniky - driveru.
Zapojení vodičů závisí na daném typu řídicí elektroniky.

ECLIPSERA s.r.o. Distributor pro ČR.

Copyright © 2017 ECLIPSERA s.r.o.
Verze 1.0

รูปที่ ค.1 คุณสมบัติ มอเตอร์ไร้แปรงถ่าน (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Order No.	Model	Motor size	Shaft size	Weight	KV (rpm/V)	Max Power	Battery	Prop	Ri(M Ω)	ESC
221206	D2826-6	$\Phi 28 \times 26 \text{mm}$	$\Phi 3.175 \times 37 \text{mm}$	50g	2200	342W	2-3Li-Po	7x3/7x4	0.033	40A
221210	D2826-10				1400	205W		9x4.7/7x4	0.078	
221213	D2826-13				1000	150W		11x7/10x5	0.127	
221215	D2826-15				930	130W		12x6/10x7	0.139	

รูปที่ ค.2 คุณสมบัติ มอเตอร์ไร้แปรงถ่าน (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

จอแสดงผล LCD

ABSOLUTE MAXIMUM RATINGS

Item	Symbol	Min.	Max.	Unit
Supply Voltage (Logic)	V _{dd} - V _{ss}	-0.3	7.0	V
Supply Voltage (LCD)	V _{dd} - V ₀	-0.3	13.0	V
Input Voltage	V _i	-0.3	V _{dd} + 0.3	V
Operating Temp.	T _{opr}	0	50	°C
Storage Temp.	T _{stg}	-10	60	°C

MECHANICAL DATA

Item	Max.	Unit
Module Size (W X H X T)	98.0 X 60.0 X 13.5	mm
Viewing Area (W X H)	76.0 X 26.0	mm
Character Size (W X H)	2.95 X 4.75	mm
Dot Size (W X H)	0.55 X 0.55	mm
View Angle	6 or 12 O'Clock	

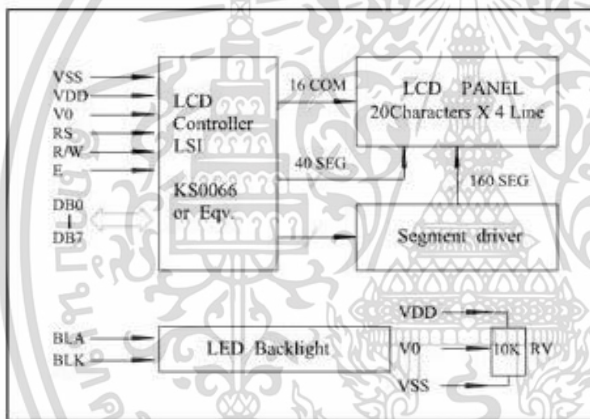
ELECTRICAL CHARACTERISTICS (V_{dd}=5V±0.25V)

Item	Symbol	Test Condition	Min.	Typ.	Max.	Unit
Input High Voltage	V _{ih}	--	2.0	--	V _{dd}	V
Input Low Voltage	V _{il}	--	-0.3	--	0.8	V
Output High Voltage	V _{oh}	I _{oh} =-0.2mA	2.4	--	V _{dd}	V
Output Low Voltage	V _{ol}	I _{ol} =-1.6mA	0	--	0.4	V
Supply Current	I _{dd}	V _{dd} =5.0V	--	2.0	3.0	mA
LCD Driving Voltage	V _{dd} - V ₀	T _a =25 °C	--	4.7	--	V

PIN CONNECTIONS

Pin	Symbol	Level	Function
1	VSS	0V	Power GND
2	VDD	+5V	Power supply for logic
3	V0	--	Operation voltage for LCD
4	RS	H/L	H:Data L:Instruction code
5	R/W	H/L	H:Read L:Write
6	E	H/L	Enable signal
7	DB0	H/L	Data bus line
8	DB1	H/L	
9	DB2	H/L	
10	DB3	H/L	
11	DB4	H/L	
12	DB5	H/L	
13	DB6	H/L	
14	DB7	H/L	Power supply for LED backlight
15	BLA	+5V	
16	BLK	0V	

BLOCK DIAGRAM



LCD Type

LCD Color	Backlight
Yellow Green	LED/Yellow Green/4.2V (-250mA)
Blue	LED/White /3.2V (-150mA)

รูปที่ ง.1 คุณสมบัติ จอแสดงผล LCD (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ
Arduino UNO

Technical Specification

EAGLE files: [arduino_duomilabovr_uno_design.zip](#) Schematic: [arduino_uno_schematic.pdf](#)

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by boot loader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

the board

รูปที่ จ.1 คุณสมบัติ Arduino UNO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ฉ

โปรแกรมการทำงานของระบบคานสมดุล

```
#include <Wire.h>
#include <Servo.h>
#include <LiquidCrystal_I2C.h> //ในส่วนแรกจะเป็นการเรียกใช้ Library
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE); //เรียกใช้ address ของ LCD
Servo right_prop;//เรียกใช้ ESC.
Servo left_prop;

////////////////////////////////////กำหนดตัวแปรที่ใช้ในการคำนวณ////////////////////////////////////

int16_t Acc_rawX, Acc_rawY, Acc_rawZ,Gyr_rawX, Gyr_rawY, Gyr_rawZ; //กำหนดตัวแปรรับค่าของ Gyro และ Acc

float Acceleration_angle[2];
float Gyro_angle[2];
float Total_angle[2];
float elapsedTime, time, timePrev;
float rad_to_deg = 180/3.141592654;
float PID, pwmLeft, pwmRight, error, previous_error;
double throttle=1300; //ค่าเริ่มต้นในการทำงานของมอเตอร์

////////////////////////////////////เรียกใช้อุปกรณ์และเซตค่าเริ่มต้นในการทำงาน////////////////////////////////////

void setup() {
time = millis();
lcd.begin(20,4); //คำสั่งในการเรียกใช้งาน LCD ประเภท 20*4
lcd.setBacklight(HIGH);
Wire.begin(); //คำสั่งเริ่มต้นการทำงานและใช้งาน ฟังก์ชัน wire
Wire.beginTransmission(0x68); //เรียกใช้ GY 521 ตาม Address จาก Data Sheet
Wire.write(0x6B)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Wire.write(0);
Wire.endTransmission(true);
Serial.begin(250000);
right_prop.attach(3); //กำหนดพอร์ทที่ใช้ในการส่งสัญญาณ PWM
left_prop.attach(5);
left_prop.writeMicroseconds(1000); //ค่าเริ่มต้นในการทำงานของ ESC.
right_prop.writeMicroseconds(1000);

```

```

delay(5000);
}
void loop() {

float pid_p=0; //กำหนดค่าเริ่มต้นของ PID
float pid_i=0;
float pid_d=0;
float xdeg = analogRead(A0); // กำหนดตัวแปรที่จะรับค่า Analog
float xKp = analogRead(A1);
float xKi = analogRead(A2);
float xKd = analogRead(A3);
float deg,ii,pp,dd; //กำหนดตัวรับค่าที่ใช้ map
deg = map(xdeg,0,1023,-4000,4000);
ii = map(xKi,0,1023,0,10000);
pp = map(xKp,0,1023,0,10000);
dd = map(xKd,0,1023,0,10000);
float p1,i1,d1,deg1; //กำหนดตัวแปรเพื่อในค่าที่ map มีทศนิยม
deg1 = deg/100;
p1 = pp/1000;
i1 = ii/1000;
d1 = dd/1000;

```

```

double kp= p1; // กำหนดตัวแปรที่รับค่าไปใช้คำนวณ

```

```

double ki= i1;

```

```

double kd= d1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float desired_angle = deg1; // กำหนดค่า set point ที่ต้องการในโปรแกรมนี้จำกัดที่ -40 - 40 องศา
```

```
//////////////////////////////////I M U//////////////////////////////////
```

```
timePrev = time;
time = millis(); // actual time read
elapsedTime = (time - timePrev) / 1000;
Wire.beginTransmission(0x68);
Wire.write(0x3B); //เรียกใช้งาน Acc X ของ GY521
Wire.endTransmission(false);
Wire.requestFrom(0x68,6,true);

Acc_rawX=Wire.read()<<8|Wire.read();
Acc_rawY=Wire.read()<<8|Wire.read();
Acc_rawZ=Wire.read()<<8|Wire.read();
// X
Acceleration_angle[0] = atan((Acc_rawY/16384.0)/sqrt(pow((Acc_rawX/16384.0),2) +
pow((Acc_rawZ/16384.0),2)))*rad_to_deg;
// Y
Acceleration_angle[1] = atan(-1*(Acc_rawX/16384.0)/sqrt(pow((Acc_rawY/16384.0),2) +
pow((Acc_rawZ/16384.0),2)))*rad_to_deg;
Wire.beginTransmission(0x68);
Wire.write(0x43); //เรียกใช้งาน Gyro
Wire.endTransmission(false);
Wire.requestFrom(0x68,4,true);

Gyr_rawX=Wire.read()<<8|Wire.read();
Gyr_rawY=Wire.read()<<8|Wire.read();
//X
Gyro_angle[0] = Gyr_rawX/131.0;
//Y
```

```
Gyro_angle[0] = Gyr_rawX/131.0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

////////////////////////////////// Complementary Filter//////////////////////////////////

//X
Total_angle[0] = 0.98 *(Total_angle[0] + Gyro_angle[0]*elapsedTime) +
0.02*Acceleration_angle[0];
//Y
Total_angle[1] = 0.98 *(Total_angle[1] + Gyro_angle[1]*elapsedTime) +
0.02*Acceleration_angle[1];
//testgyro = Total_angle[1] -6;

////////////////////////////////// P I D ////////////////////////////////////

error = Total_angle[1] - desired_angle ;
pid_p = kp*error;
if(-3 <error <3)
{
pid_i = pid_i+(ki*error);
}
pid_d = kd*((error - previous_error)/elapsedTime);
if(PID < -1000)
{
PID=-1000;
}
if(PID > 1000)
{
PID=1000;
}
//นำค่า PID ที่ได้มาคำนวณเพื่อใช้ในการปรับคาสัญญาณของ ESC.
pwmLeft = throttle + PID;
pwmRight = throttle - PID;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(pwmRight < 1000)//กำหนดค่าต่ำสุดและสูงสุดของคาบสัญญาณเนื่องจากคาบสัญญาณใน
    ไบรารี ของเซอร์โวสามารถสร้างได้เพียง 1000-2000 ไมโครเซค
{
pwmRight= 1000;
}
if(pwmRight > 2000)
{
pwmRight=2000;
}
//Left
if(pwmLeft < 1000)
{
pwmLeft= 1000;
}
if(pwmLeft > 2000)
{
pwmLeft=2000;
}
left_prop.writeMicroseconds(pwmLeft); //คำสั่งให้มอเตอร์ทำงานตามขนาดสัญญาณที่ได้
                                       จากการควบคุม
right_prop.writeMicroseconds(pwmRight);
previous_error = error;
////////////////////////////////////ส่วนของ LCD////////////////////////////////////

lcd.setCursor(1,0); //รับค่าเพื่อแสดงผลในส่วนของเซตพ้อย
Serial.print(deg1);
lcd.print("Degree :"),
lcd.print(deg1);

lcd.setCursor(6,1); //รับค่าเพื่อแสดงผลในส่วนของ Kp
Serial.print(p1);
lcd.print("Kp :"),
lcd.print(p1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lcd.setCursor(6,2);    //รับค่าเพื่อแสดงผลในส่วนของ Ki
Serial.print(i1);
lcd.print("Ki :"),
lcd.print(i1);

lcd.setCursor(6,3);    //รับค่าเพื่อแสดงผลในส่วนของ Kd
Serial.print(dd);
lcd.print("Kd :"),
lcd.print(d1);

} //จบการทำงาน

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้