การออกแบบตัวควบคุมการเคลื่อนไหวใต้น้ำสำหรับหุ่นยนต์ดำน้ำ

ROBOT SUBMARINE CONTROLLER

นายอภิชิต นาคำ รหัส 54360957

นายอภิรัฐ นาคำ รหัส 54360964

นายกิตตินันทน์ กันแก้ว รหัส 54363095

30 ก.ย. 2558

16914556

ปร.

0252

2557

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมเครื่องกล ภาควิชาวิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2557

ปีการศึกษา 2557

ใบรับรองโครงงาน

หัวข้อโครงงาน การออกแบบตัวควบคุมการเคลื่อนไหวใต้น้ำสำหรับหุ่นยนต์ดำน้ำ

ผู้ดำเนินโครงงาน นายอภิชิต นาคำ 54360957
นายอภิรัฐ นาคำ 54360964
นายกิตตินันทน์ กันแก้ว 54363095

ที่ปรึกษาโครงงาน นายชูพงศ์ ช่วยเพ็ญ

ภาควิชา วิศวกรรมเครื่องกล

ปีการศึกษา 2557

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานวิศวกรรมเครื่องกลฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมเครื่องกล

คณะกรรมการสอบโครงงานวิศวกรรมเครื่องกล

.....ที่ปรึกษา
(นายชูพงศ์ ช่วยเพ็ญ)

.....กรรมการ
(ผศ.ดร.อนันต์ชัย อยู่แก้ว)

.....กรรมการ
(อ.สุรเจษฎ์ สุขไชยพร)

ชื่อหัวข้อโครงงาน การออกแบบตัวควบคุมการเคลื่อนไหวใต้น้ำสำหรับหุ่นยนต์ดำน้ำ

ผู้จัดทำโครงงาน นายอภิชิต นาคำ 54360957
นายอภิรัฐ นาคำ 54360964
นายกิตตินันทน์ กันแก้ว 54363095

ที่ปรึกษาโครงงาน นายชูพงศ์ ช่วยเพ็ญ

สาขาวิชา วิศวกรรมเครื่องกล

ภาควิชา วิศวกรรมเครื่องกล

ปีการศึกษา 2557

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ เป็นการออกแบบตัวควบคุมการเคลื่อนไหวใต้น้ำสำหรับหุ่นยนต์ดำน้ำ (Robot submarine controller) เนื่องจากมหาวิทยาลัยต่างๆทั้งในและต่างประเทศได้มีการประดิษฐ์ เพื่อการแข่งขันขึ้น โดยการออกแบบนั้นต้องสอดคล้องต่อกติกาที่คณะกรรมการได้กำหนดไว้ ไม่ว่าจะเป็นในเรื่องของ น้ำหนัก ขนาด และการทำงานแบบอัตโนมัติทั้งหมด ซึ่งการจัดแข่งขันในที่นี้ก็เพื่อนำหุ่นยนต์ดำน้ำที่ชนะการแข่งขัน ไปใช้สำรวจในสภาพแวดล้อมจริง อาจจะเป็นการสำรวจรอยรั่วของท่อส่งน้ำมันในทะเล คนหาศพผู้ประสบภัยจากเหตุการณ์การต่างๆในทะเล เป็นต้น ผู้จัดทำโครงงานจึงออกแบบระบบตัวควบคุมการเคลื่อนไหวใต้น้ำสำหรับหุ่นยนต์ดำน้ำ ให้มีประสิทธิภาพสูงสุด สามารถนำไปใช้งานได้ในสภาพแวดล้อมจริง และตอบสนองต่อความต้องการดังกล่าวข้างต้น

ในการดำเนินโครงงาน จะทำการศึกษาและรวบรวมทฤษฎี เพื่อสร้างแบบจำลองทางคณิตศาสตร์ที่มีความสัมพันธ์ของค่าสัญญาณการควบคุมกับการเคลื่อนที่ของหุ่นยนต์ดำน้ำ และออกแบบระบบควบคุมการเคลื่อนที่ของหุ่นยนต์ให้สามารถเคลื่อนที่ได้อย่างอัตโนมัติแล้วทำการออกแบบตัวควบคุมโดยใช้บอร์ด Arduino Mega 2560 ร่วมกับ Analog Joystick และศึกษาเครื่องมือในโปรแกรม MATLAB ที่ช่วยให้สามารถเข้าใจถึงการทำงานของระบบได้ดียิ่งขึ้น จากนั้นจัดหาวัสดุและมอเตอร์ที่สามารถใช้งานร่วมกันได้เพื่อขับเคลื่อนหุ่นยนต์ดำน้ำ และทำการติดตั้งเข้ากับหุ่นยนต์ดำน้ำต้นแบบ เพื่อนำไปทดสอบกับสภาพแวดล้อมจริง

ในการทดสอบกับสภาพแวดล้อมจริงแบบจำลองทางคณิตศาสตร์ที่สร้างขึ้นมาสามารถอธิบายการเคลื่อนที่ของหุ่นยนต์ดำน้ำได้โดยการคิดจากกรอบอ้างอิงแกน x y z ทำให้มีความสัมพันธ์การเคลื่อนที่ X(t) ของหุ่นยนต์ดำน้ำ เป็นไปได้ตามที่ต้องการ

Project title	Robot Submarine Controller
Author	Mr. Apichit Nakum ID 54360957
	Mr. Apirat Nakum ID 54360964
	Mr. Kittinan Kankaew ID 54363095
Project advisor	Mr. Choopong Chuaypen
Major	Mechanical Engineering
Department	Mechanical Engineering
Academic year	2014

Abstract

This thesis describes an underwater robot designed for submarine controller. The reason why I chose this subject, is because many universities make these robots and compete with each other. I have designed a robot conform to the rules of the committee, considering the weight, the dimensions and all the automations. The winner of this competition will be used in real environments. An example of its use, would be the exploration of an oil pipeline leak in the sea, or to search for victims after a maritime accident. The project is designed to prepare the motion controller system for underwater robotic submarine for maximum efficiency. It can be used in a real environment and respond to the demand.

In this project, I studied and compiled various theories to create a mathematical model of the control signal value to control the movement of the robot submarine, and I designed a system that can control the robot's movements automatically. Then I designed a controller, using Arduino Mega 2560 (2017) boards, together with an analog joystick, and I also studied tools from the MATLAB program that can help me understand the system better. Then, I searched for the supplies and the motor that are used together to propel the diving robot and installed it on the prototype to test it in the actual environment.

In the real environment test, the mathematical model that was built could describe the robot submarine's motion successfully by using the reference axis x y z into a relationship of the motion X (t) of the robot submarine.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีนั้น ต้องขอขอบคุณอาจารย์ชูพงศ์ ช่วยเพ็ญ อาจารย์ที่ปรึกษาปริญญานิพนธ์ ซึ่งท่านได้ให้คำปรึกษา แนะนำข้อผิดพลาดต่างๆ และช่วยแก้ปัญหาข้อบกพร่องของการดำเนินโครงงานด้วยดีตลอดมา จนทำให้ปริญญานิพนธ์นี้มีความสมบูรณ์และถูกต้อง

ขอขอบคุณอาจารย์ และบุคลากรของภาควิชาวิศวกรรมเครื่องกลทุกท่าน ที่คอยให้ความช่วยเหลือในการใช้อาคารปฏิบัติการภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร รวมถึงอุปกรณ์เครื่องมือต่างๆ ที่ทางภาควิชาวิศวกรรมเครื่องกลได้จัดเตรียมให้ยืมใช้งานจนงานที่ทำสำเร็จลุล่วงเป็นอย่างดีตลอดมา

สุดท้ายนี้ ผู้จัดทำโครงงานขอกราบขอบพระคุณบิดา มารดา และครอบครัว ที่สนับสนุนในด้านการเงิน และให้กำลังใจแก่ผู้จัดทำโครงงานเสมอจนสำเร็จการศึกษา รวมถึงเพื่อนๆ ที่คอยแนะนำช่วยเหลือแก่ผู้จัดทำโครงงานด้วยดีตลอดมา

ผู้จัดทำโครงงาน

นายอภิชิต นาคำ

นายอภิรัฐ นาคำ

นายกิตตินันทน์ กันแก้ว

เมษายน 2558

สารบัญ

หน้า

ใบรับรองปริญญานิพนธ์ ..... ก
บทคัดย่อ ..... ข
บทคัดย่อภาษาอังกฤษ (Abstract) ..... ค
กิตติกรรมประกาศ ..... ง
สารบัญ ..... จ
สารบัญตาราง ..... ฉ
สารบัญรูป ..... ช

บทที่ 1 บทนำ 1
1.1 ความเป็นมาและความสำคัญของโครงงาน 1
1.2 วัตถุประสงค์ของโครงงาน 2
1.3 ขอบเขตของโครงงาน 2
1.4 ขั้นตอนการดำเนินการโครงงาน 3
1.5 ระยะเวลาในการดำเนินโครงงาน 3
1.6 สถานที่ในการดำเนินโครงงาน 4
1.7 ประโยชน์ที่คาดว่าจะได้รับ 4
1.8 รายละเอียดงบประมาณตลอดโครงงาน 4

บทที่ 2 หลักการและทฤษฎีเบื้องต้น 5
2.1 หลักการการออกแบบตัวควบคุมการเคลื่อนที่ของหุ่นยนต์ดำน้ำ 5
2.1.1 แรงลอยตัว 5
2.1.2 ความหนืด 5
2.1.3 แรงขับ 6
2.1.4 ความเร็วรอบของมอเตอร์ 6
2.1.5 แรงดันไฟฟ้า 6

สารบัญ (ต่อ)

หน้า

2.1.6 แนวความคิดของแบบจำลองทางคณิตศาสตร์ 7
2.2 วรรณกรรมปริทรรศน์ 8
2.3 หลักการของอุปกรณ์ที่เกี่ยวข้อง 10
2.3.1 ไมโครคอนโทรลเลอร์ (Microcontroller) 10
ก. บอร์ด STM34F4 Discovery 10
ข. บอร์ดอาดูอิโน่ (Arduino) รุ่น Mega 2560 11
ค. วงจรบอร์ดขับมอเตอร์ (Motor Driver) 13
2.3.2 มอเตอร์ไฟฟ้ากระแสตรง (DC Motor) 13
ก. Motor Driver 14
1. มอเตอร์กระแสตรงที่ใช้ในการทดลอง 14
2. มอเตอร์ที่ใช้เป็นตัวขับให้หุ่นยนต์ดำน้ำ 16
2.3.3 อุปกรณ์ป้อนสัญญาณ 18
ก. Analog Joystick 18
ข. IMU 19
1. เซนเซอร์วัดความเร่ง (Accelerometer) 19
2. ไจโรเซนเซอร์ (Gyro Sensor) 20
2.4 หลักการการควบคุมมอเตอร์ด้วยชุดควบคุมสำเร็จรูป 22
2.5 หลักการการควบคุมความเร็วรอบและทิศทางการหมุนของมอเตอร์ 22

บทที่ 3 วิธีดำเนินโครงงาน 23
3.1 ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง 24
3.2 สมการการเคลื่อนที่ของหุ่นยนต์ดำน้ำ 25
3.3 สมการการเคลื่อนที่ของมอเตอร์ 27
3.4 การควบคุมความเร็วรอบของมอเตอร์ 28
3.5 การสร้างแบบจำลองทางคณิตศาสตร์ด้วยชุดควบคุมสำเร็จรูป 30

สารบัญ (ต่อ)

หน้า

3.6 IMU....32
3.7 Manual Operate....33
3.8 Auto Operate....34

บทที่ 4 บทสรุปและข้อเสนอแนะ....37

เอกสารอ้างอิง....39

ภาคผนวก ก ขั้นตอนการสร้างแบบจำลองทางคณิตศาสตร์ด้วยรูปภาพ หรือ GUI....41
ภาคผนวก ข โค้ดสำหรับการควบคุมการเคลื่อนที่ของหุ่นยนต์ดำน้ำ....51

ประวัติผู้จัดทำโครงงาน....58

สารบัญตาราง

ตารางที่ หน้า

1.1 ขั้นตอนการดำเนินโครงงาน.....3

สารบัญรูป

รูปที่ หน้า

2.1 แนวแกนการเคลื่อนที่แบบ 3 มิติ....8

2.2 แนวการเคลื่อนที่แบบ 2 มิติ....8

2.3 Board STM34F4 Discovery....11

2.4 Arduino Mega 2560....12

2.5 Board Motor Driver....13

2.6 มอเตอร์ไฟฟ้ากระแสตรง....14

2.7 FAULHABER MOTOREN 2342L012C R....14

2.8 ทิศทางการไหลที่เป็นผลรวมของแนวทั้งสอง....16

2.9 มอเตอร์ขับหุ่นยนต์ดำน้ำ....17

2.10 Analog Joystick....18

2.11 แกน XYZ บนเครื่องที่มีขนาดมากกว่า 7 นิ้ว....20

2.12 แกน XYZ บนเครื่องที่มีขนาดน้อยกว่า 7 นิ้ว....20

2.13 ลักษณะการเอียงในทิศทางต่างๆ....21

2.14 ความพิเศษของการเคลื่อนที่ในแนวแกน z....21

3.1 ผังงานแสดงขั้นตอนการดำเนินโครงการ....23

3.2 แบบหุ่นยนต์ดำน้ำ....24

3.3 รูปภาพแสดงตัวแปรที่เกี่ยวข้องสำหรับมอเตอร์กระแสตรง....28

3.4 ชุดคำสั่งสำหรับควบคุมหุ่นยนต์ดำน้ำ....30

3.5 ความเร็วรอบของมอเตอร์ที่ใช้ในการทดสอบ....31

3.6 ตัวอย่างส่วนการทำงานของไจโรสโคป (Gyroscope)....32

3.7 หน้าต่างแสดงผลมุมเอียงในแนวแกนทั้ง 3 แนวแกน....33

3.8 ตัวอย่างส่วนการทำงานของการควบคุมด้วยตนเอง (Manual Operate)....34

3.9 ตัวอย่างส่วนการทำงานของการดำเนินงานแบบอัตโนมัติ (Auto Operate)....36

ก.1 Encoder....43

ก.2 Target setup....43

สารบัญรูป (ต่อ)

รูปที่ หน้า

ก.3 UART Setup....44
ก.4 UART Tx....44
ก.5 Digital Output 1,2....45
ก.6 Basic PWM....45
ก.7 Pulse Generator....46
ก.8 Counter Free-Running....46
ก.9 Subsystem....47
ก.10 Encoder Read....48
ก.11 Volatile Data Storage (Out of date) 1,2....48
ก.12 Volatile Data Storage Read 1,2....49
ก.13 Data Type Conversion....49
ก.14 Pulse Generator....50

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของโครงงาน

ต้องยอมรับว่าในปัจจุบัน รวมไปถึงอนาคต นวัตกรรมและเทคโนโลยีเป็นปัจจัยสำคัญอย่างมากในการพัฒนาประเทศรวมทั้งสังคมของโลกด้วย การสำรวจหาแหล่งพลังงานน้ำมันหรือก๊าซธรรมชาติในทุกวันนี้ จะต้องไปสำรวจในที่ที่ยากลำบาก และต้องสำรวจทรัพยากรใต้น้ำลึก และในอนาคตปริมาณเชื้อเพลิงที่สำรวจได้ภายในประเทศกำลังลดลง ประกอบกับการนำเข้าเชื้อเพลิงอาจมีราคาสูงตาม ดังนั้น การพัฒนาเทคโนโลยีด้านหุ่นยนต์ดำน้ำ (Autonomous under vecer : AUV) จึงเป็นปัจจัยสำคัญที่จะช่วยพัฒนาศักยภาพการสำรวจทรัพยากรในประเทศได้เป็นอย่างดี รวมทั้งมีความปลอดภัยในการทำงาน แต่เนื่องจากว่าหุ่นยนต์ดำน้ำยังเป็นเทคโนโลยีที่ใหม่และยังไม่มีการพัฒนามากเท่าไรนัก จึงมีความคิดที่จะพัฒนาเทคโนโลยีของหุ่นยนต์ดำน้ำขึ้นในประเทศไทย โดยตัวหุ่นยนต์ดำน้ำจะทำงานด้วยตัวเองแบบอัตโนมัติ เนื่องจากการทำงานของหุ่นยนต์ดำน้ำ เป็นการทำงานใต้น้ำ ซึ่งมีข้อจำกัดหลายๆอย่าง เช่น ความดัน การมองภาพใต้น้ำ และ การสื่อสารผ่านตัวกลางที่ไม่ใช่อากาศ จึงมีความจำเป็นมากที่หุ่นยนต์ดำน้ำควรมีการตัดสินใจได้ด้วยตัวเอง

องค์ความรู้ที่จะประดิษฐ์หุ่นยนต์ดำน้ำอัตโนมัติมีส่วนที่สำคัญอยู่หลายส่วน แต่เราจะเน้นไปที่ระบบ ควบคุมแบบอัตโนมัติ ทั้งในส่วนของหุ่นยนต์ดำน้ำและในส่วนของมอเตอร์ สำหรับควบคุมและขับเคลื่อนหุ่นยนต์ดำน้ำ โดยสิ่งที่เราจะต้องทำคือ หาความสัมพันธ์ของปริมาณทางกายภาพ (Physics properties) ที่มีผลต่อการเคลื่อนที่ เพื่อนำไปสร้างแบบจำลองทางคณิตศาสตร์ที่มีความสัมพันธ์ของค่าสัญญาณการควบคุมกับการเคลื่อนที่ของหุ่นยนต์ดำน้ำ และออกแบบระบบควบคุมการเคลื่อนที่ของหุ่นยนต์ให้สามารถเคลื่อนที่ได้อย่างอัตโนมัติ โดยใช้คอมพิวเตอร์ช่วยในการออกแบบ (computer simulation)

1.2 วัตถุประสงค์ของการศึกษา

1.2.1 เพื่อศึกษาความสัมพันธ์ของปริมาณทางกายภาพ (Physics properties) ที่มีผลต่อการเคลื่อนที่ของหุ่นยนต์ดำน้ำ

1.2.2 เพื่อสร้างแบบจำลองทางคณิตศาสตร์ (mathematical model) ของหุ่นยนต์ดำน้ำ เพื่อใช้ในการอธิบายความสัมพันธ์ของค่าสัญญาณควบคุมกับการเคลื่อนที่ของหุ่นยนต์ดำน้ำ

1.2.3 เพื่อศึกษาและออกแบบระบบควบคุมการเคลื่อนที่ของหุ่นยนต์ดำน้ำให้สามารถเคลื่อนที่ได้อย่างอัตโนมัติ ตามค่าเป้าหมายที่กำหนดให้ โดยใช้คอมพิวเตอร์ช่วยในการออกแบบ (computer simulation)

1.2.4 เพื่อศึกษาและประยุกต์ใช้ระบบควบคุมที่ออกแบบกับต้นแบบหรือส่วนต้นแบบเพื่อยืนยันผลสำเร็จของการใช้งานตัวควบคุมที่ออกแบบนี้

1.3 ขอบเขตของโครงงาน

1.3.1 ทำการสังเคราะห์แบบจำลองทางคณิตศาสตร์ที่ใช้ในการอธิบายความสัมพันธ์ของตำแหน่งของหุ่นยนต์ดำน้ำกับค่าสัญญาณความคุมทางไฟฟ้าจากตัวควบคุม (ค่าแรงดันไฟฟ้าที่ใช้ในการควบคุมความเร็วของมอเตอร์สร้างแรงขับ)

1.3.2 ออกแบบตัวควบคุมตำแหน่งสำหรับหุ่นยนต์ดำน้ำที่ทำงานด้วยมอเตอร์ไฟฟ้าเพื่อสร้างแรงขับ

1.3.3 ทำการทดสอบสมรรถนะการทำงานของตัวควบคุมดังกล่าวด้วยการจำลองทางคอมพิวเตอร์ ภายใต้เงื่อนไขแบบต่างๆ

1.3.4 ทำการประยุกต์ตัวควบคุมที่ออกแบบกับตัวต้นแบบหรือส่วนต้นแบบของหุ่นยนต์ดำน้ำ

1.4 ขั้นตอนการดำเนินการโครงงาน

1.4.1 ศึกษาความสัมพันธ์ของปริมาณทางกายภาพที่เกิดขึ้นและมีผลต่อการเคลื่อนที่ใต้น้ำของหุ่นยนต์ดำน้ำ และสร้างแบบจำลองทางคณิตศาสตร์เพื่อใช้ในการออกแบบตัวควบคุม

1.4.2 สร้างแบบจำลองทางคณิตศาสตร์ด้วยโปรแกรมคอมพิวเตอร์สำเร็จรูป และทำการศึกษาพฤติกรรมของหุ่นยนต์ดำน้ำต่อค่าการควบคุมที่กำหนดให้

1.4.3 ออกแบบและทดสอบสมรรถนะของตัวควบคุมการเคลื่อนที่สำหรับหุ่นยนต์ดำน้ำ โดยการจำลองด้วยคอมพิวเตอร์

1.4.4 สร้างต้นแบบหรือส่วนของตัวต้นแบบเพื่อใช้ในการประยุกต์ใช้ตัวควบคุมที่ออกแบบ

1.4.5 สรุปและประเมินผลโครงงาน

1.5 ระยะเวลาในการดำเนินโครงงาน

ตาราง 1.1 ตารางแสดงขั้นตอนการดำเนินโครงงาน

กิจกรรม	2557						2558				
	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.
1.ศึกษาทฤษฎีรวบรวมข้อมูลรายละเอียดต่างๆ ของงานวิจัยที่เกี่ยวข้อง											
2. ทำรายงานปริทัศน์											
3. แก้โครงร่างโครงงานและสอบหัวข้อโครงร่าง											
4. สร้างแบบจำลองทางคณิตศาสตร์ด้วยโปรแกรมคอมพิวเตอร์สำเร็จรูป											
5.ออกแบบและทดสอบสมรรถนะของตัวควบคุมการเคลื่อนที่ โดยการจำลองด้วยคอมพิวเตอร์											
6. สร้างต้นแบบหรือส่วนของตัวต้นแบบเพื่อใช้ในการประยุกต์ใช้ตัวควบคุมที่ออกแบบ											
7. สรุปและประเมินผลโครงการ											

1.6 สถานที่ในการดำเนินโครงงาน

ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

1.7 ประโยชน์ที่คาดว่าจะได้รับ

1.7.1 มีความรู้ความเข้าใจถึงการทำงานของหุ่นยนต์ดำน้ำ ตลอดจนเข้าใจความสัมพันธ์ของปริมาณทางกายภาพที่เกิดขึ้นและมีผลต่อการเคลื่อนที่ใต้น้ำ

1.7.2 มีความรู้ความเข้าใจในกระบวนการออกแบบระบบควบคุมการเคลื่อนที่ของหุ่นยนต์ดำน้ำ

1.7.3 สามารถใช้โปรแกรมคอมพิวเตอร์สำเร็จรูปเพื่อใช้ในการศึกษาและสร้างแบบจำลองทางคณิตศาสตร์ ตลอดจนใช้ในการออกแบบและทดสอบความสามารถของตัวควบคุมที่ออกแบบ

1.7.4 สามารถประยุกต์ใช้งานตัวควบคุมที่ออกแบบกับตัวต้นแบบหรือส่วนต้นแบบได้จริง

1.8 รายละเอียดงบประมาณตลอดโครงงาน

1.8.1 ค่ากระดาษ	500	บาท
1.8.2 ค่าจัดทำรูปเล่ม	1000	บาท
1.8.3 ค่าวัสดุอุปกรณ์อื่นๆ	1500	บาท

บทที่ 2

หลักการและทฤษฎีเบื้องต้น

2.1 หลักการการออกแบบตัวควบคุมการเคลื่อนที่ของหุ่นยนต์ดำน้ำ

หลักการของการออกแบบตัวควบคุมการเคลื่อนที่ของหุ่นยนต์ดำน้ำ มีด้วยกันสองส่วน คือ พลศาสตร์การเคลื่อนที่ของหุ่นยนต์ดำน้ำ และการออกแบบตัวควบคุมแบบป้อนกลับ ซึ่งพลศาสตร์การเคลื่อนที่ของหุ่นยนต์ดำน้ำจะอยู่บนพื้นฐานของการเคลื่อนที่ตามกฎการเคลื่อนที่ของ นิวตันนั้น คือ

$$\Sigma F = ma \tag{2.1}$$

ซึ่งจะใช้ในการอธิบายถึงความสัมพันธ์ของแรงที่เกิดขึ้นและเกี่ยวข้องกับการเคลื่อนที่ของหุ่นยนต์ดำน้ำ กับผลของแรงซึ่งก็คือการเคลื่อนที่หุ่นยนต์ดำน้ำ ซึ่งแรงที่เกิดขึ้นและมีผลต่อการเคลื่อนที่ของหุ่นยนต์ดำน้ำได้แก่

2.1.1 แรงลอยตัว โดยอาศัยหลักการของอาคีมีดิส ซึ่งมีใจความสำคัญว่า วัตถุที่จมในของเหลวหมดทั้งก้อนหรือจมแต่เพียงบางส่วนจะถูกแรงลอยตัวกระทำ และแรงลอยตัวจะมีค่าเท่ากับน้ำหนักของของเหลวที่วัตถุนั้นแทนที่

$$F_b = \rho_L g v \tag{2.2}$$

2.1.2 แรงต้านจากความหนืด เป็นแรงต้านทานการไหลในตัวของไหลที่ขึ้นอยู่กับของไหลนั้นๆ ซึ่งเมื่อวัตถุจะเคลื่อนที่ ต้องคำนึงถึงแรงตัวนี้ด้วย

$$F_{vis} = K_{vis} V_{vis} \tag{2.3}$$

โดยที่ K_{vis} คือ เป็นค่าคงที่ (ขึ้นอยู่กับรูปทรงของวัตถุ)

V_{vis} คือ ความเร็ว

2.1.3 แรงขับ ซึ่งเกิดจากใบพัดที่ขับด้วยมอเตอร์ ซึ่งมีความสัมพันธ์ระหว่าง แรงดันไฟฟ้ากับความเร็วรอบของมอเตอร์ เป็นต้นกำลังของการขับเคลื่อนของวัตถุ ที่จะทำให้สามารถเคลื่อนที่ไปได้

$$T = k_T I_a \tag{2.4}$$

$$= \frac{k_T (V_L - E_c)}{R_a}$$

$$T = \frac{k_T (V_L - k_c \omega)}{R_a}$$

2.1.4 ความเร็วรอบมอเตอร์ ซึ่งต้องเลือกให้สัมพันธ์กับงานที่ออกแบบไว้

$$\omega = \frac{k_T V_L - TR_a}{k_T k_c} \tag{2.5}$$

โดยที่ k_c เป็นค่าคงที่ของแรงดันไฟฟ้าต้านกลับ (V/rad/s)

k_T เป็นค่าคงที่ของแรงบิด (N.m/A)

มอเตอร์ไฟฟ้ากระแสตรง ซึ่งถือว่าเป็นแรงที่ใช้ในการขับเคลื่อนให้ได้ตามจุดหมายที่ต้องการ โดยแรงขับมอเตอร์นี้ ได้จากมอเตอร์ไฟฟ้าทำหน้าที่เปลี่ยนแปลงความเร็วรอบของมอเตอร์ออกมาเป็นแรงขับของน้ำที่ได้ออกมา ปัจจัยของแรงขับขึ้นอยู่กับเส้นผ่านศูนย์กลางด้วย แต่ความเร็วของมอเตอร์ไฟฟ้านี้ขึ้นอยู่กับปริมาณทางไฟฟ้า หรือ กระแสที่ใช้ในการควบคุม แต่ในที่นี้เราใช้การควบคุมขนาดของแรงดันทางไฟฟ้า เพื่อควบคุมความเร็วรอบของมอเตอร์ไฟฟ้า เพื่อจุดหมายสุดท้าย นั้นคือ แรงขับที่เราต้องการ ที่ใช้ในการเคลื่อนที่ซึ่งสมการไฟฟ้าที่ได้ก็คือ

2.1.5 แรงดันไฟฟ้า เป็นแรงที่จะต้องใส่ให้กับมอเตอร์ให้ได้ความเร็วรอบเพื่อให้วัตถุเคลื่อนที่ไปอยู่ตามตำแหน่งที่ต้องการ

$$E_a = IR \tag{2.6}$$

จากหลักการข้างต้นที่ได้กล่าวมา เราจะเห็นได้ว่าเราสามารถหาค่าความสัมพันธ์ของแรงดันไฟฟ้าที่จะได้จากระบบควบคุมการเคลื่อนที่เพื่อให้หุ่นยนต์ดำน้ำเคลื่อนที่ได้ตามตำแหน่ง ทิศทาง ความเร็ว และความเร่ง ที่ต้องการได้ ซึ่งจะอยู่ในรูปของสมการแบบจำลองทางคณิตศาสตร์ เพื่อให้ง่ายต่อการออกแบบระบบควบคุม

2.1.6 แนวความคิดของแบบจำลองทางคณิตศาสตร์ การควบคุมการทำงานของหุ่นยนต์สำรวจใต้น้ำ เป็นการสั่งการให้สร้างแรงขับ เพื่อขับดันหุ่นยนต์สำรวจให้น้ำให้เคลื่อนที่ไปตามทิศทาง ตำแหน่ง และความเร็ว ที่ต้องการ โดยให้การสร้างสมการการควบคุมให้อยู่ในรูป

$$\ddot{x} = A\dot{x} + Bu \tag{2.7}$$

โดยที่ u คือ แรงขับเคลื่อน

$\dot{x}$ คือ สภาวะปัจจุบันของหุ่นยนต์สำรวจใต้น้ำ

เนื่องจากแรงขับที่ได้เกิดจากการควบคุมความเร็วรอบของตัวส่งกำลังขับ (Thruster) ซึ่งทำงานด้วยมอเตอร์กระแสตรง (DC Motor) ดังนั้นตัวควบคุมจริงที่ใช้ในการสั่งการก็คือ แรงดันไฟฟ้า (Voltage) ที่จ่ายให้กับมอเตอร์ เพื่อควบคุมความเร็วรอบ เป็นผลให้ได้แรงขับที่ต้องการ ดังนั้น จึงมีความสัมพันธ์ทางคณิตศาสตร์ดังนี้

1. ความเร็วรอบของมอเตอร์กระแสตรงต่อแรงดันไฟฟ้า

$$P(s) = \frac{\dot{\theta}(s)}{E(s)} = \frac{K}{(Ls+R)(Js+b)+K^2(s)} \tag{2.8}$$

2. ตัวส่งกำลังขับต่อความเร็วรอบของมอเตอร์กระแสตรง

$$\frac{T(s)}{\dot{\theta}(s)} = Js^2 + bs \tag{2.9}$$

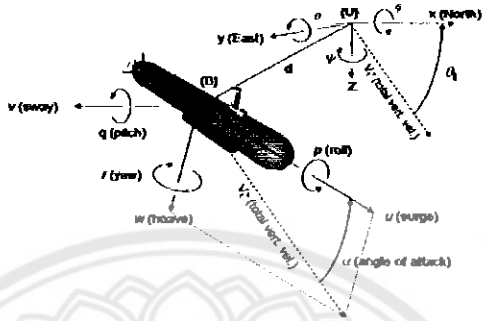
3. ตำแหน่งการเคลื่อนที่ของหุ่นยนต์ดำน้ำต่อตัวส่งกำลังขับ

$$\frac{X(t)}{T(s)} = \frac{\sqrt{\left(\frac{2K_t E_a(s)}{(ms+K)R}\right)^2 + \left(\frac{4K_t E_a(s) + (\rho_L v_g + m)g}{(ms+K)R}\right)^2}}{(Js+b)\dot{\theta}(s)} \tag{2.10}$$

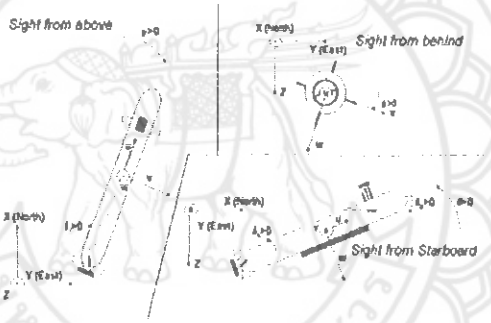
ซึ่งความสัมพันธ์ดังกล่าว สามารถควบคุมตำแหน่งการเคลื่อนที่ของหุ่นยนต์ดำน้ำได้ด้วยแรงดันไฟฟ้าที่จ่ายให้กับมอเตอร์กระแสตรงทำให้มีแรงขับที่ต้องการได้

2.2 วรรณกรรมปริทรรศน์

ผลงานของ Lionel Lapierre ได้ทำการศึกษาระบบการเคลื่อนที่เรือดำน้ำซึ่งเป็นการเคลื่อนที่แบบ 6 แนวแกน เพื่อควบคุมเรือดำน้ำ โดยมีหลักการคร่าวๆ ดังนี้



รูปที่ 2.1 แนวแกนการเคลื่อนที่แบบ 3 มิติ



รูปที่ 2.2 แนวการเคลื่อนที่แบบ 2 มิติ

ที่มา : http://www.sciencedirect.com/science/article/pii/S002980180800228X

รูปที่ 2.1 และ รูปที่ 2.2 แสดงให้เห็นถึงระบบพิกัดที่ใช้ในการวิเคราะห์การเคลื่อนที่ของเรือดำน้ำในระบบพิกัด 3 มิติและภาพฉายใน 2 มิติตามลำดับ ในการควบคุมการเคลื่อนที่ของเรือดำน้ำจะแบ่งการควบคุมออกเป็น 2 มุมมองคือ จากผู้สังเกต และ จากเรือดำน้ำ ซึ่งสามารถตั้งระบบพิกัดของทั้งสองมุมมองให้สอดคล้องกันดังในรูปที่ 2.1 และ 2.2 โดยมีความสัมพันธ์ทางจลน์ศาสตร์ดังนี้

กรอบ {U} คือระบบพิกัดจากมุมมองผู้สังเกต และกรอบ {B} คือระบบพิกัดจากมุมมองหุ่นยนต์ดำน้ำ จุด $d=[x,y,z]$ เป็นตำแหน่งของจุดกำเนิดของกรอบ {B} กับที่อ้างอิงกับกรอบ {U} และสามารถอธิบายความหมายของตัวแปรต่างๆ ได้ดังนี้

[u,v,w] คือค่าความเร็วเชิงเส้นในทิศทางพุ่งตรงไปข้างหน้าหรือพุ่งชน (surge) ในทิศทางด้านข้างหรือทิศทางการแกว่งตัว (sway) และในทิศทางแนวดิ่ง (heave) ตามลำดับ โดยอ้างอิงกับระบบพิกัด {B}

[ϕ,θ,ψ] คือค่าเชิงมุมได้แก่ มุมหมุน (roll) มุมยก (pitch) และมุมเหวียง (yaw) ตามลำดับ โดยวัดมุมการหมุนรอบแกนในระบบพิกัด {U}

[p,q,r] คือค่าความเร็วเชิงมุมในที่หมุนรอบแนวแกน [u,v,w] โดยความสัมพันธ์ทางจนล์ศาสตร์ระหว่าง [p,q,r] และ [$\dot{\phi},\dot{\theta},\dot{\psi}$] เป็นไปตามสมการที่ (1)

V_t^v กำหนดผลรวมความเร็วในระนาบการดำน้ำ $v_t^v = \sqrt{u^2 + w^2}$

α คือมุมปะทะ หาได้จาก $\alpha = -\arctan(w/u)$

θ_t คือ มุมของผลรวมเวกเตอร์ความเร็วที่ระนาบพื้นผิวตามแนวนอน $\theta_t = \theta + \alpha$

δ_t^{ω} และ δ_t^{ω} มุมหักเหของหางเสือทั้งด้านบนและด้านล่างของหางเสือ เป็นการควบคุมแบบอิสระ ซึ่งเป็นการชดเชยการเกิดการหมุนจากตัวส่งแรง(thruster)

δ_b และ δ_s คือมุมหักเห ที่หัวเรือและท้ายเรือ บนผิว ตามลำดับ ส่วนผิวคู่ ด้านข้างเรือและกราบขวาเรือ แสดงการความคุมพื้นผิวสำหรับมุมหักเห

ซึ่ง สมการสำเร็จจลน์ศาสตร์ของหุ่นยนต์ดำน้ำเมื่อเทียบจากมุมมองของหุ่นยนต์ดำน้ำไปยังมุมมองผู้สังเกตการณ์คือ

$$\dot{x} = u\cos\psi\cos\theta - v\sin\psi\cos\phi + v\cos\psi\sin\theta\sin\phi + w\sin\psi\sin\phi + w\cos\psi\sin\theta\cos\phi \quad (2.11)$$

$$\dot{y} = u\sin\psi\cos\theta + v\cos\psi\cos\phi + v\sin\psi\sin\theta\sin\phi - w\cos\psi\sin\phi + w\sin\psi\sin\theta\cos\phi \quad (2.12)$$

$$\dot{z} = -u\sin\theta + v\cos\theta\sin\phi + w\cos\theta\cos\phi \quad (2.13)$$

$$\dot{\phi} = p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \quad (2.14)$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \quad (2.15)$$

$$\dot{\psi} = q\sin\phi/\cos\theta + r\cos\phi/\cos\theta \quad (2.16)$$

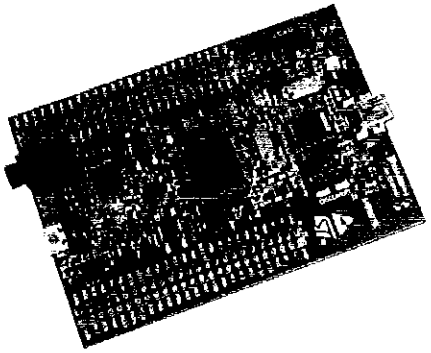
สมการที่ (1) จะแสดงให้เห็นถึงความสัมพันธ์ของตำแหน่งและการเคลื่อนที่ของเรือดำน้ำเมื่อมองจากมุมมองผู้สังเกตการณ์ (ระบบพิกัด {U}) กับความเร็วในแต่ละทิศทางที่เกิดขึ้นจากการขับเคลื่อนของระบบขับเคลื่อนของเรือดำน้ำซึ่งจะให้ค่าความเร็วในแต่และทิศทางเทียบกับระบบพิกัด {B} หรือจากมุมมองของเรือดำน้ำเอง ซึ่งเมื่อเราทราบหรือกำหนดค่าความเร็วที่เกิดขึ้นที่เรือดำน้ำก็จะสามารถใช้สมการที่ (1) ในการพิจารณาค่าความเร็วที่เกิดขึ้นจากมุมมองของผู้สังเกตการณ์ได้ หรืออาจจะใช้ในทางกลับกัน ทั้งนี้ค่าความเร็วที่เกิดขึ้นหรือในทางกลับกันก็คือค่าความเร็วที่ต้องการในการเคลื่อนที่ที่เรือดำน้ำต้องใช้ จะเกิดขึ้นจากระบบขับเคลื่อนของเรือดำน้ำ ซึ่งสามารถนำไปใช้ในการพัฒนาระบบควบคุมการเคลื่อนที่แบบอัตโนมัติได้

2.3 หลักการของอุปกรณ์ที่เกี่ยวข้อง

2.3.1 ไมโครคอนโทรลเลอร์ (Microcontroller)

ก. STM34F4 Discovery กลุ่ม STM34F4 ถูกพัฒนาโดยบริษัท เอมเมจิน จำกัด ทำให้สามารถนำเครื่องมือของคณิตศาสตร์ที่มีอยู่ในโปรแกรม MATLAB ซึ่งก็คือ Blockset Waijung มาใช้งานได้อย่างสะดวก Blockset Waijung เป็นโมดุลเสริมที่ใช้ติดตั้งใน Simulink ซึ่งใช้สำหรับเขียนโค้ดการทำงานเพื่อใช้งานสำหรับไมโครคอนโทรลเลอร์ โดยใช้ GUI ที่สามารถทำความเข้าใจและพัฒนาได้ง่าย

เนื่องจากกำลังไฟฟ้าของไมโครคอนโทรลเลอร์มีจำกัดจึงไม่สามารถขับมอเตอร์กระแสตรงได้โดยตรง จึงจำเป็นต้องมีวงจรขยายเพื่อใช้ในการควบคุมมอเตอร์ด้วย วงจรที่ว่าก็คือ DC Motor Drive โดยรับค่าระดับแรงดันเฉลี่ย หรือสัญญาณ PWM จากไมโครคอนโทรลเลอร์และสั่งงานมอเตอร์กระแสตรง ต่อไป



รูปที่ 2.3 Board STM34F4 Discovery

ที่มา : http://songklod2524.blogspot.com/2014/10/dc-motor-speed-control-by-stm32f4.html

ข. บอร์ดอาดูอิโน่ (Arduino) รุ่น Mega 2560 เป็นบอร์ดที่ออกแบบมาสำหรับงานที่ต้องใช้ I/O มาก เช่น งานที่ต้องการรับสัญญาณจาก Sensor หรือควบคุมมอเตอร์ Servo หลายๆ ตัว ซึ่งเป็นบอร์ดที่เหมาะสมกับการทำงานในการควบคุมมอเตอร์สำหรับขับเคลื่อนหุ่นยนต์ดำน้ำ สามารถตอบสนองต่อความต้องการใช้งานของการทำงานได้เป็นอย่างดี และข้อดีของบอร์ดอาดูอิโน่คือ มีตัวอย่างโค๊ดให้ศึกษาเพื่อเป็นแนวทางการทำงานได้มากมาย โดยบอร์ด Arduino Mega 2560 เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ทำงานบนพื้นฐานของ ATmega2560 โดยบอร์ด Arduino Mega 2560 นี้มีทุกสิ่งที่ไมโครคอนโทรลเลอร์จำเป็นต้องใช้อย่างการต่อไฟเลี้ยงสามารถเชื่อมรับพลังงานโดยการเชื่อมต่อ micro USB connector หรือ จาก power supply จากภายนอกได้ โดยแหล่งพลังงานจะถูกเลือกโดยอัตโนมัติ แหล่งจ่ายจากภายนอกสามารถมาได้จาก AC-to-DC adapter หรือจากแบตเตอรี่ โดยต่อเข้ากับ 2.1 mm center-positive plug ไปยังช่องเสียบแหล่งจ่าย และการต่อเข้ากับแบตเตอรี่สามารถทำได้โดยการต่อเข้ากับ GND และ Vin pin header ของ power conecter บอร์ดสามารถทำงานได้ในช่วงแรงดัน 6 ถึง 20 volts ถ้า แหล่งจ่ายมีค่าต่ำกว่า 7 V อาจส่งผลให้ 5 V pin มีแรงดันที่ต่ำกว่า 5V และ บอร์ดอาจจะไม่เสถียร แต่ถ้าหากแรงดันมีค่าสูงกว่า 12 V อาจส่งผลให้บอร์ด Overheat และอาจทำให้บอร์ดเสียหายได้ ดังนั้นช่วงแรงดันที่เหมาะสมกับบอร์ดคือ 7 V ถึง 12 V โดยที่ VIN เป็น input voltage ของบอร์ด อาดูอิโน่ (Arduino) โดยใช้แหล่งจ่ายจากภายนอกดังนี้

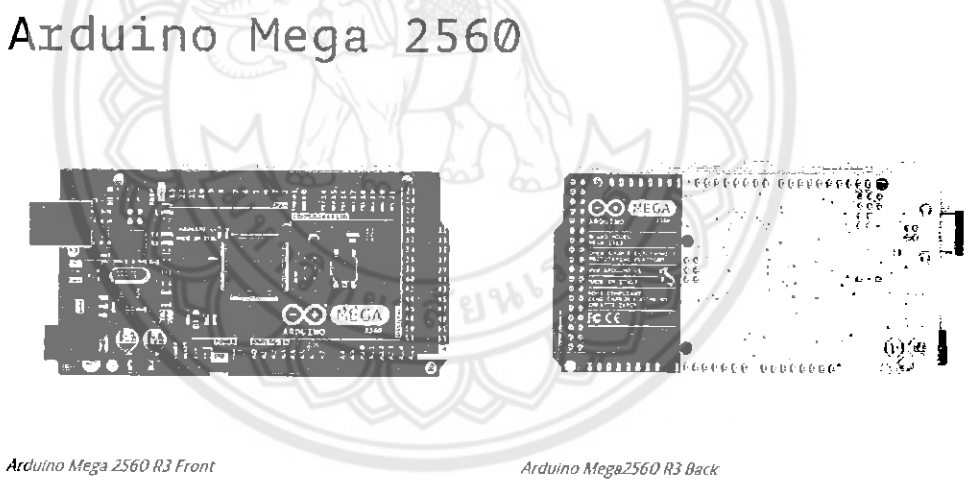
- 5V เป็น output pin ที่ควบคุม 5V จากบอร์ด

- 3V3 เป็น 3.3 volt supply ที่สร้างขึ้นจาก regulator บนบอร์ด ให้กระแสได้สูงสุด 50 mA

- GND เป็น ground pin

- IOREF เป็น pin ที่ให้ voltage reference กับไมโครคอนโทรลเลอร์ เพื่อเลือกค่าแรงดันให้กับ shield ที่มาเชื่อมต่อกับบอร์ด

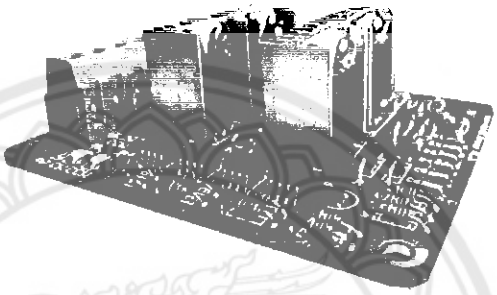
ความจำของ Arduino Mega 2560 มีหน่วยความจำ 256 KB (8 KB ใช้สำหรับ bootloader) นอกจากนี้ยังมีอีก 8 KB สำหรับ SRAM และ 4 KB สำหรับ EEPROM และในแต่ละ digital pins ทั้ง 54 pins บนบอร์ด Arduino Uno สามารถเป็นได้ทั้ง input และ output โดยจะทำงานที่แรงดัน 5 V และให้กระแสสูงสุด 40 mA



รูปที่ 2.4 Arduino Mega 2560

ที่มา : http://arduino.cc/en/Main/arduinoBoardMega2560

ค. วงจรบอร์ดขับมอเตอร์ (Motor Driver) ในการควบคุมตำแหน่งหรือความเร็วของมอเตอร์กระแสตรงเราสามารถทำได้โดยการควบคุมระดับแรงดัน (Voltage) ที่จ่ายให้แก่มอเตอร์กระแสตรง เทคนิคหนึ่งของการควบคุมระดับแรงดันไฟฟ้าก็คือการทำงานของ PWM (Pulse Width Modulation) เป็นการปรับช่วงและขนาดความถี่ของสัญญาณไฟฟ้า ซึ่งจะมีผลต่อระดับแรงดันเฉลี่ยที่จ่ายให้ ซึ่งจะกล่าวโดยละเอียดต่อไป

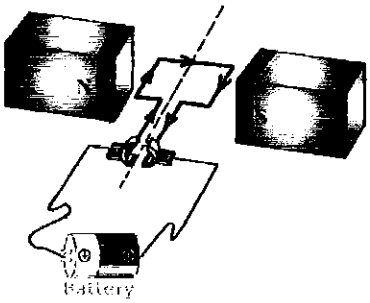


รูปที่ 2.5 Board Motor Driver

ที่มา : http://www.mechanicsources.com/index.php/en/component

/content/category/10-products

2.3.2 มอเตอร์ไฟฟ้ากระแสตรง (DC Motor) เมื่อมีกระแสไหลผ่านเข้าไปในมอเตอร์กระแสจะแบ่งออกไป 2 ทาง คือ ส่วนที่หนึ่งจะผ่านเข้าไปที่ขดลวดสนามแม่เหล็ก (Field coil) ทำให้เกิดสนามแม่เหล็กขึ้นและอีกส่วนหนึ่งจะผ่านแปลงถ่านคาร์บอนและผ่านคอมมิวเตเตอร์ เข้าไปในขดลวดอาร์เมเจอร์ทำให้เกิดสนามแม่เหล็กขึ้นเช่นกัน ซึ่งทั้งสองสนามจะเกิดขึ้นขณะเดียวกัน ตามคุณสมบัติของเส้นแรงแม่เหล็กแล้วจะไม่มีการตัดกัน จะมีแต่การหักล้างและการเสริมกัน ซึ่งทำให้เกิดแรงบิดในอาร์เมเจอร์ ทำให้อาร์เมเจอร์หมุนซึ่งในการหมุนนั้นจะเป็นไปตามกฎมือซ้ายของเฟลมมิ่ง (Fleming's lefthand rule

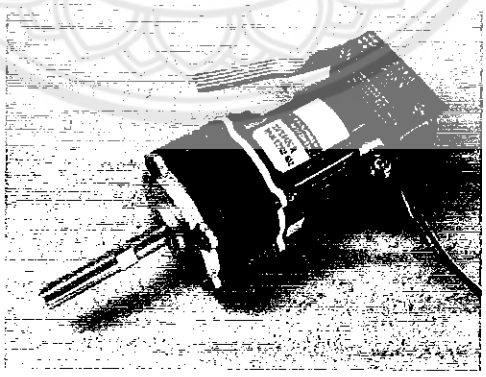


รูปที่ 2.6 มอเตอร์ไฟฟ้ากระแสตรง

ที่มา : http://www.atom.rmutphysics.com/charud/oldnews/0/286/15/14/DC-MOTOR/motorDC_simulation_files/frame.htm

ก.Motor Driver

1. มอเตอร์กระแสตรงที่ใช้ในการทดลอง เพื่อหาสมการ Regression ในการทำให้มอเตอร์มีความเสถียร ภายใน 1 รอบ มีจำนวน Pulse เกิดขึ้นทั้งหมด 120 Pulse/0.001 sec ภายในมอเตอร์ LEGO นั้นมีระบบทดเกียร์ภายใน อัตราส่วนการทดเกียร์ทั้งหมดมีค่า 64:1 หมายความว่ามอเตอร์หมุนภายใน 64 รอบจะมีค่าเท่ากับ ส่วนหมุนจริงภายนอก 1 รอบ จึงมีค่าพลัส (pulse) เท่ากับ $12 \times 64 = 768$ Pulse/rev



รูปที่ 2.7 FAULHABER MOTOREN 2342L012C R

ที่มา : http://www.ebay.com/itm/FAULHABER-DC-12V-Motor-2342L012CR-Gear-64-1-Encoder-12CPR-USED-free-ship-/15100521615

มอเตอร์กระแสตรง มีคุณสมบัติสามารถหมุนปรับทิศทางของมอเตอร์ได้ ซึ่งการเปลี่ยนทิศการหมุนของมอเตอร์นั้นทำได้โดยการสลับค่าคงที่ในโปรแกรม Simulink ที่ได้เขียนไว้ เพื่อป้อนคำสั่งให้ได้ทิศทางตามที่เราต้องการ คุณสมบัติของมอเตอร์ DC อีกประการนั้นคือ สามารถทำการส่งสัญญาณ Encoder ไปยังบอร์ด STM32F4 ที่ Encoder Capture ซึ่งจะได้ค่ามุมการเคลื่อนที่และตำแหน่งออกมา

การอ่านค่าความเร็วมอเตอร์หาค่าได้จากการนำค่าตำแหน่งที่ได้ ทำการหาค่าอนุพันธ์ (Differential) ขอค่าระยะทางที่อ่านค่าได้เทียบกับเวลา

$$\omega_l = \frac{dP}{dt} \tag{2.17}$$

เมื่อ ω_l = ค่าความเร็ว (pulse/sec)

P = ค่า pulse ที่อ่านได้จาก Encoder (pulse)

อ่านค่าเป็นรอบของมอเตอร์ สามารถหาค่าได้จาก

$$R = \frac{P}{720} \tag{2.18}$$

เมื่อ R = ค่ารอบมอเตอร์ (rev)

ดังนั้นเมื่อต้องการอ่านค่าความเร็วของมอเตอร์

$$N = \frac{dR}{dt} \tag{2.19}$$

เมื่อ N = ค่าความเร็วรอบมอเตอร์ (rev/sec)

และสามารถหาค่าความเร็วรอบของมอเตอร์ในหน่วยอื่นได้เป็น

$$n = N \times 60 \tag{2.20}$$

เมื่อ n = ค่าความเร็วรอบมอเตอร์ (rev/min)

หรือ แสดงในรูปอนุพันธ์ได้เป็น

$$n = \frac{60}{720} \times \frac{dP}{dt} \tag{2.21}$$

และสามารถหาค่าความเร็วรอบของมอเตอร์ในหน่วยอื่นได้เป็น

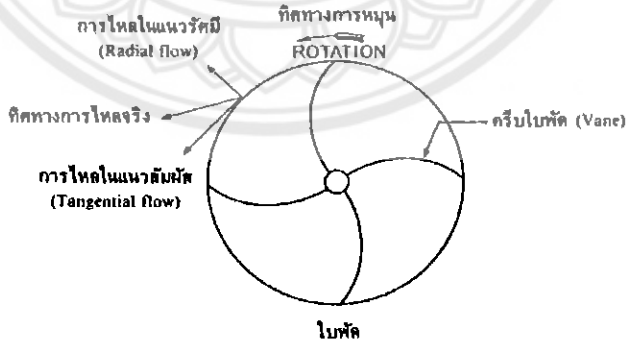
$$\omega = n \times 2\pi \qquad (2.22)$$

หรือ แสดงในรูปอนุพันธ์ได้เป็น

$$\omega = \frac{2\pi}{720} \times \frac{dP}{dt} \qquad (2.23)$$

เมื่อ ω = ค่าความเร็ว (rad/sec)

2. มอเตอร์ที่ใช้เป็นตัวขับให้หุ่นยนต์ดำน้ำ เป็นมอเตอร์ที่ใช้หลักการเดียวกับปั๊มน้ำแบบจุ่ม ซึ่งเป็นหนึ่งในชนิดของปั๊มน้ำแบบหอยโข่ง ปั๊มแบบนี้ทำงานโดยอาศัยการหมุนของใบพัดหรืออิมเพลเลอร์(Impeller) ที่ได้รับการถ่ายเทกำลังจากเครื่องยนต์ต้นกำลังหรือมอเตอร์ไฟฟ้า เมื่อใบพัดหมุนพลังงานจากเครื่องยนต์ก็จะถูกถ่ายเทโดยการผลักดันของครีบใบพัด (vane) ต่อของเหลวที่อยู่รอบๆทำให้เกิดการไหลในแนวสัมผัสกับเส้นรอบวง (Tangential flow) เมื่อมีการไหลในลักษณะดังกล่าวก็จะเกิดแรงเหวี่ยงหนีศูนย์กลาง (Centrifugal force) และเป็นผลให้มีการไหลจากจุดศูนย์กลางของใบพัดออกไปสู่แนวเส้นรอบวงทุกทิศทาง (Radial flow) ดังนั้นของเหลวที่ถูกใบพัดผลักดันออกมาก็จะมีทิศทางการไหลที่เป็นผลรวมของแนวทั้งสอง ดังรูปที่

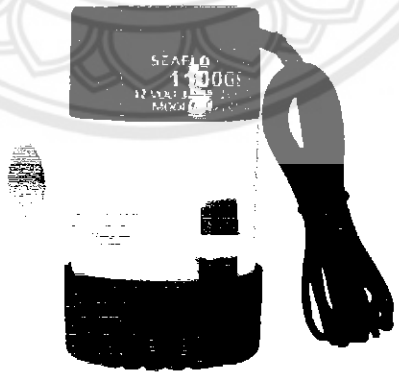


รูปที่ 2.8 ทิศทางการไหลที่เป็นผลรวมของแนวทั้งสอง

ที่มา : http://202.129.59.73/tn/september54/Pump1.htm

โดยหลักฃลศาสตร์เมื่อของเหลวถูกหมุนให้เกิดแรงหนีศูนย์กลางความกดดันของของเหลวจะมีค่ามากขึ้นเมื่ออยู่ห่างจากศูนย์กลางของใบพัดมากขึ้น เมื่อความเร็วของใบพัดซึ่งหมุนอยู่ในภาชนะปิดมากพอ ความกดดันที่จุดศูนย์กลางก็จะต่ำกว่าความกดดันของบรรยากาศ ดังนั้นปั๊มแบบอาศัยแรงเหวี่ยงหนีจุดศูนย์กลางที่แท้จริงจึงมีทางให้ของเหลวไหลเข้าหรือทางดูด (Suction Opening) อยู่ที่ศูนย์กลางใบพัด

ของเหลวที่ถูกดูดเข้าทางศูนย์กลาง เมื่อถูกผลักดันออกไปด้วยแรงผลักดันของครีบใบพัดและแรงเหวี่ยงหนีจุดศูนย์กลาง ก็จะไหลออกมาตลอดแนวเส้นรอบวง ดังนั้นใบพัดจึงจำเป็นต้องอยู่ในเรือนปั๊ม (Casing) เพื่อทำหน้าที่รวบรวมและผันของเหลวเหล่านี้ไปสู่ทางจ่าย (Discharge Opening) เพื่อต่อเข้ากับท่อส่งหรือระบบใช้งานต่อไป ดังนั้นจะมีจุดหนึ่งซึ่งผนังภายในของเรือนปั๊มเข้ามาชิดกับขอบของใบพัดมากจุดดังกล่าวนี้เรียกว่า ลิ้นของเรือนปั๊ม (Tonque of the easing) จากลิ้นของเรือนปั๊มไปตามทิศทางการหมุนของใบพัด จะมีของเหลวไหลออกมามากขึ้นตามความยาวของเส้นรอบวงของใบพัดที่เพิ่มขึ้น ดังนั้นช่องว่างซึ่งเป็นทางเดินของของเหลวระหว่างผนังของเรือนปั๊มกับใบพัดก็จะต้องเพิ่มขนาดขึ้นด้วย โดยหลักการแล้วอัตราการเพิ่มพื้นที่หน้าตัดจะคงที่เพื่อให้ความเร็วของการไหลสม่ำเสมอซึ่งจะเป็นผลให้มีการสูญเสียพลังงานน้อยลงนั่นเอง อย่างไรก็ตาม ความเร็วของการไหลจะลดลงเนื่องจากพลังงานบางส่วนถูกเปลี่ยนมาเป็นพลังงานศักย์ (Potential Energy) ในรูปของความดัน (Pressure head) แทน

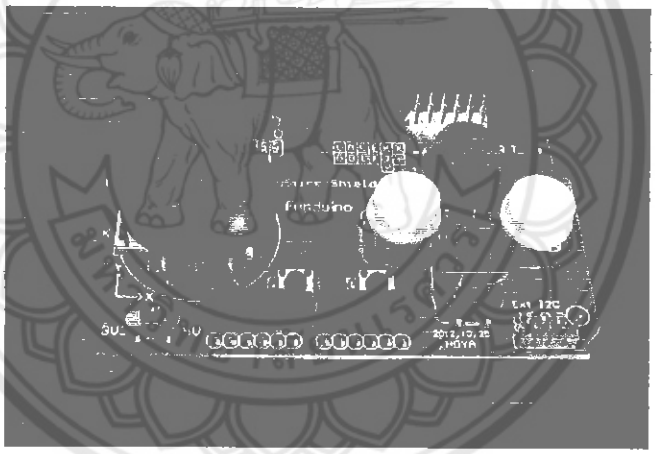


รูปที่ 2.9 มอเตอร์ขับหุ่นยนต์ดำน้ำ

ที่มา : http://s1372.photobucket.com/user/thaiwatersystem/media/SEAFLO-1100_zpsb62c8c17.jpg.html

2.3.3 อุปกรณ์ป้อนสัญญาณ

ก. Analog Joystick เป็นอุปกรณ์ที่ทำมาเพื่อประยุกต์ใช้งานร่วมกับ Arduino เพื่อสร้างสัญญาณ PWM (Pulse Width Modulation) ความถี่ 50 Hz ซึ่งสามารถใช้ควบคุมการทำงานของอุปกรณ์อื่นได้ เช่น R/C Servo หรืออุปการณ์ ESC (Electronic Speed Controller) สำหรับมอเตอร์แบบ Brushless เช่น ในกรณีที่ใช้สำหรับเครื่องบินบังคับหรือหุ่นยนต์ การทำงานของ Analog Joystick อุปกรณ์ X/Y Joystick (หรือ Thumb Joystick) มีคันโยกแบบสองแกน โดยมีตัวต้านทานปรับค่าได้สองตัวสำหรับแต่ละแกน ถ้านำไปต่อกับแรงดันไฟเลี้ยง +3V และ GND ก็จะใช้งานเป็นวงจรแบ่งแรงดัน (Voltage Divider) และค่าของสัญญาณหรือระดับแรงดัน จะเปลี่ยนตามตำแหน่งของคันโยก ถ้านำไปต่อกับขาแบบอนาล็อก-อินพุต (Analog Input) ของไมโครคอนโทรลเลอร์ เช่น Arduino ก็สามารถระบุตำแหน่งหรือมุมของคันโยกในแต่ละแกนได้



รูปที่ 2.10 Analog Joystick

ที่มา : http://www.arduinoall.com/product/77/joystick-shield-expansion-board-for-arduino

ข. IMU (Inertial Measurement Unit) เป็นอุปกรณ์วัดความเอียงของวัตถุที่ประกอบไปด้วยไจโรเซนเซอร์ (Gyro Sensor) และเซนเซอร์วัดความเร่ง (Accelerometer) ไว้ในตัวเดียวกัน ซึ่งเมื่อทำงานร่วมกันแล้ว สามารถทำในเกิดความละเอียดในการอ่านค่าและมีประสิทธิภาพในการทำงานมากกว่าตัว ไจโรเซนเซอร์ หรือเซนเซอร์วัดความเร่ง แบบตัวเดียวแน่นอน ตัวอย่างเช่น

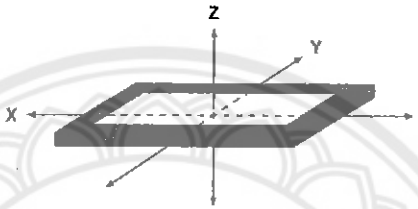
- ขับรถความเร็วคงที่ไปข้างหน้า : เซนเซอร์ทั้งสองตัวอ่านได้ 0
- ขับรถตรงไปข้างหน้าโดยมีอัตราเร่ง : เซนเซอร์วัดความเร่ง อ่านค่าได้ค่าคงที่ ไจโรเซนเซอร์อ่านได้ 0
- ขับรถเข้าโค้ง(รัศมีวงกลม)ด้วยความเร็วคงที่ : เซนเซอร์วัดความเร่ง อ่านค่าได้ค่าไม่คงที่ ไจโรเซนเซอร์ อ่านได้ค่าคงที่
- ขับรถเข้าโค้ง(รัศมีวงกลม)โดยมีความเร่ง : เซนเซอร์วัดความเร่ง อ่านค่าได้ไม่คงที่ ไจโรเซนเซอร์ อ่านค่าได้ไม่คงที่
- ขับรถหมุนรอบตัวเอง : เซนเซอร์วัดความเร่ง อ่านค่าได้ 0 ไจโรเซนเซอร์ อ่านได้ค่าคงที่

โดยมีหลักการทำงานของเซนเซอร์แต่ละตัวดังนี้

1. เซนเซอร์วัดความเร่ง (Accelerometer) ใช้สำหรับเป็นตัวชี้ว่าวัตถุหรือหุ่นยนต์ดำน้ำอยู่ในสถานะนิ่งเฉย (Static) หรือเคลื่อนไหวทันทีทันใด หรือหยุดทันทีทันใด (Dynamic) ทำให้เซนเซอร์วัดความเร่ง เป็นเซนเซอร์ สำหรับบอกสถานะการเอียงได้เป็นอย่างดี (Tilt Sensor) ประโยชน์ที่ได้จากเซนเซอร์วัดความเร่ง ที่นอกเหนือจากความสะดวกในการใช้งานแล้ว สิ่งใกล้ตัวที่สุดในสมัยนี้ก็คือเกมส์ในมือถือ คือการที่เกมส์หลายๆประเภท ให้ความสมจริงในการควบคุมที่ดีขึ้น ไม่ว่าจะเกมส์ขับรถที่สามารถเอียงซ้าย ขวา เสมือนเป็นพวงมาลัยในการควบคุม เอียงออกจากตัว เอียงเข้าหาตัว เสมือนการเร่งหรือเบรก การบังคับหุ่นยนต์ดำน้ำก็เช่นเดียวกับการที่เราบังคับเกมส์ เซนเซอร์วัดความเร่ง ก็จะเป็นตัวควบคุมทิศทางการเคลื่อนที่ของหุ่นยนต์ดำน้ำ แต่เนื่อง เซนเซอร์วัดความเร่ง ควบคุมได้แค่สองแนวแกน ซึ่งชีวิตจริง เราคงไม่สามารถบังคับให้หุ่นยนต์ดำน้ำเคลื่อนที่ให้ตรงแนว เอียงตรงแนว หรือเคลื่อนไหวแบบตรงไปตรงมาได้ตลอดเวลา ทำให้ในบางการเคลื่อนไหวหรือการเปลี่ยนมุมของหุ่นยนต์ดำน้ำ อาจตอบสนองไม่ได้ตั่งใจ จึงได้มีการต่อยอดเซนเซอร์วัดความเร่งให้ทำงานได้อย่างมีประสิทธิภาพมากขึ้น โดยการพัฒนาเซนเซอร์วัดความเร่ง ให้ครอบคลุมทั้ง 3 แนวแกน ซึ่งเรียกว่าไจโรสโคป (Gyroscope)

2. ไจโรเซนเซอร์ (Gyro Sensor) เป็นเครื่องมือสำหรับการวัดหรือการรักษาการปรับทิศทาง โดยที่จะเป็นความเร่งเชิงมุมของแกน XYZ สำหรับแกน XYZ บนอุปกรณ์ใดๆ ส่วนมากจะมีตำแหน่งดังนี้

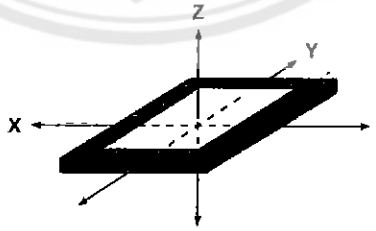
ตำแหน่งสำหรับแกน X กับ Y จะขึ้นอยู่กับตัวอุปกรณ์ ในรูปที่ 2.11 นี้ คือแกน XYZ บนเครื่องที่มีขนาดมากกว่า 7 นิ้ว โดยมีแกน X ตามแนวกว้างของจอ และแกน Y ตามแนวสูง



รูปที่ 2.11 แกน XYZ บนเครื่องที่มีขนาดมากกว่า 7 นิ้ว

ที่มา : http://www.akexorcist.com/2013/03/android-code-gyroscope.html

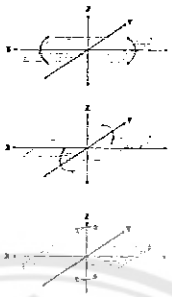
ตำแหน่งสำหรับแกน X กับ Y จะขึ้นอยู่กับตัวอุปกรณ์ในรูปที่ 2.12 นี้ คือแกน XYZ บนเครื่องที่มีขนาดน้อยกว่า 7 นิ้ว โดยจะเห็นว่าแนวแกน X กับ Y ไม่เหมือนกับเครื่องมือที่มีขนาดมากกว่า 7 นิ้ว ดังนั้นเวลาใช้ไจโรสโคปก็ให้คำนึงถึงเรื่องนี้ด้วย



รูปที่ 2.12 แกน XYZ บนเครื่องที่มีขนาดน้อยกว่า 7 นิ้ว

ที่มา : http://www.akexorcist.com/2013/03/android-code-gyroscope.html

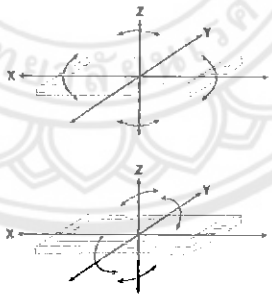
การวัดความเร่งในการเอียงก็คือการเอียงในแต่ละทิศ ที่จะมีลักษณะการเอียงในทิศทางต่างๆ ดังนี้



รูปที่ 2.13 ลักษณะการเอียงในทิศทางต่างๆ

ที่มา : http://www.akexorcist.com/2013/03/android-code-gyroscope.html

ให้สังเกตว่า แกน X และ Y จะมีแค่ขึ้นลงเท่านั้น แต่แกน Z จะพิเศษกว่าคือมีทั้งสองแกนที่เคลื่อนที่ ดังนั้นเวลาที่เอียงไปทางแกน X แกน Z ก็จะเอียงด้วย และเมื่อเอียงไปทางแกน Y แกน Z ก็จะเอียงตาม



รูปที่ 2.14 ความพิเศษของการเคลื่อนที่ในแนวแกน z

ที่มา : http://accelerometer-gyroscope-sensor-app.blogspot.com/

ดังนั้น ไจโรสโคปจะวัดความเร่งเชิงมุมในแต่ละแกน ก็คือ เวลาที่เครื่องอยู่นิ่งๆไม่มีการขยับค่าแต่ละแกนก็จะเป็น 0

2.4 หลักการการควบคุมมอเตอร์ด้วยชุดควบคุมสำเร็จรูป

การควบคุมมอเตอร์โดยโปรแกรม Simulink เป็นซอฟต์แวร์ที่ทำงานอยู่บน Matlab ใช้ในการสร้างแบบจำลองทางคณิตศาสตร์ด้วยรูปภาพ เครื่องมือสร้างแบบจำลอง การเลียนแบบ และเครื่องมือสำหรับวิเคราะห์ข้อมูล เพื่อแสดงถึงการออกแบบแนวความคิดของระบบด้วย นั่นคือ Simulink ติดต่อกับผู้ใช้ผ่านทางรูปภาพ หรือ GUI (Graphic User Interface) ในการสร้างไดอะแกรมของแบบจำลอง นอกจากนั้น Simulink ยังประกอบด้วยไลบรารีบล็อกพื้นฐานและขั้นสูงเฉพาะสาขาวิชา ทั้งระบบเชิงเส้น (Linear System) ระบบไม่เชิงเส้น (Nonlinear System) ระบบเวลาต่อเนื่อง (Continuous-time) ระบบที่เวลาแบบแซมปิ้ง (Sample time) ระบบไฮบริด (Hybrid) Simulink เปลี่ยนแปลงค่าพารามิเตอร์ได้ขณะที่เรากำลังเลียนแบบระบบอยู่ ทำให้เราเห็นการเปลี่ยนแปลงที่เกิดขึ้นได้ทันที สุดท้าย Simulink สามารถเชื่อมต่อข้อมูลกับ Matlab ได้โดยตรง ทำให้เราสามารถเชื่อมต่อกับซอฟต์แวร์และฮาร์ดแวร์ภายนอกได้

2.5 หลักการการควบคุมความเร็วรอบและทิศทางการหมุนของมอเตอร์

ทำได้โดยการปรับระดับแรงดันโดยใช้หลักการเปลี่ยนค่าความต้านทานของวงจร ซึ่งทำให้แรงดันตกคร่อมเปลี่ยนไปด้วย แต่จะเห็นว่ามีวิธีนี้เป็นการสร้างความสูญเสียเนื่องจากกำลังไฟฟ้าตกคร่อมตัวต้าน (พลังงานไฟฟ้าเปลี่ยนเป็นพลัง งานความร้อน)

การควบคุมตำแหน่งของมอเตอร์ก็คือ การควบคุมความเร็วนั่นเอง ยกตัวอย่างเช่น หากเราต้องการให้มอเตอร์ หมุน10 องศา ในตอน 0-10 องศา เราต้องควบคุมให้มอเตอร์มีความเร็ว (ค่าหนึ่ง) และ เมื่อถึงตำแหน่ง 10 องศาแล้วเรา จึงสั่งให้มอเตอร์มีความเร็วเป็น 0 ในขณะเดียวกัน หากเรากลับทิศทางการหมุนของมอเตอร์ นั่นก็คือเราสั่งให้มอเตอร์ หมุนด้วยความเร็วที่เป็นลบนั่นเอง

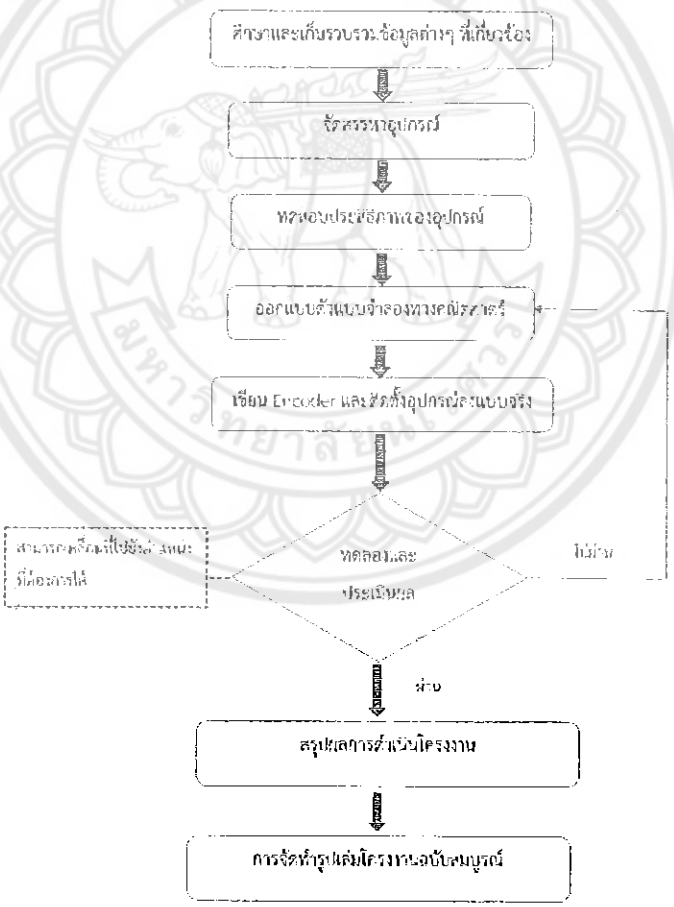
ในการออกแบบระบบควบคุมตำแหน่ง เราจะต้องสร้างระบบควบคุมความเร็วก่อน และอาศัยสัญญาณผลต่างของตำแหน่ง ที่ต้องการเคลื่อนที่ไป และสัญญาณที่ได้จากอุปกรณ์ป้อนกลับ (เรียกสัญญาณนี้ว่าสัญญาณคลาดเคลื่อน) สัญญาณที่ได้นี้จะถูกป้อนเป็นสัญญาณขาเข้าของตัวควบคุม หลังจากนั้นตัวควบคุมจะประมวลผลเพื่อให้สัญญาณขับ เพื่อให้มอเตอร์หมุนไปในตำแหน่งที่ต้องการ

จะเห็นว่าหากตำแหน่งที่ต้องการเคลื่อนที่ไปมีค่าเท่ากับตำแหน่งที่อ่านได้จากการอ่านตำแหน่งป้อนกลับ สัญญาณคลาด เคลื่อนจะมีค่าเป็นศูนย์ นั่นคือจะไม่มีสัญญาณขับออกมาจากตัวควบคุม ทำให้มอเตอร์ไม่มีการหมุนหรือมอเตอร์มีความ เร็วเป็นศูนย์นั่นเอง

บทที่ 3

วิธีการดำเนินโครงงาน

วิธีการดำเนินโครงงาน เป็นการวางแผนการทำงาน เพื่อออกแบบตัวควบคุมการเคลื่อนไหวใต้น้ำสำหรับหุ่นยนต์ดำน้ำ ตั้งแต่ขั้นตอนการศึกษาและเก็บรวบรวมข้อมูลต่างๆ ที่เกี่ยวข้อง จนถึงการจัดทำรูปเล่มโครงงานฉบับสมบูรณ์ สามารถแสดงขั้นตอนการดำเนินโครงงานในรูปของผังงานแสดงขั้นตอนการดำเนินโครงงาน แสดงดังรูปที่ 3.1



รูปที่ 3.1 ผังงานแสดงขั้นตอนการดำเนินโครงการ

3.1 ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ทำการศึกษางานวิจัยที่เกี่ยวข้องกับการออกแบบตัวควบคุมการเคลื่อนไหวใต้น้ำสำหรับหุ่นยนต์ดำน้ำ ซึ่งในที่นี้ได้ศึกษางานวิจัยของ Lionel Lapierre ซึ่งเป็นงานวิจัยที่เกี่ยวกับการออกแบบตัวควบคุม การเคลื่อนที่ทั้งหมดของหุ่นยนต์ดำน้ำ งานวิจัยนี้ได้แบ่งการมองออกเป็น 2 กรอบอ้างอิง คือ ผู้สังเกตหรือผู้บังคับและการมองโดยตัวของพาหนะและได้แสดงสมการการเคลื่อนที่ไว้ โดยครอบคลุมทั้งในส่วนของแกนการเคลื่อนที่ของมอเตอร์ 3 แกน การเคลื่อนที่ของตัวหุ่นยนต์ดำน้ำ 6 แกน แล้วนำค่าตัวแปรหรือปัจจัยต่างๆ ที่ส่งผลต่อหุ่นยนต์ดำน้ำ มาคำนวณและหาความสัมพันธ์เพื่อให้หุ่นยนต์ดำน้ำเคลื่อนที่ไปได้ตามที่ได้กำหนดไว้

จากการออกแบบหุ่นจำลองของกลุ่มการออกแบบทางกลของหุ่นยนต์ดำน้ำเพื่อการสำรวจและปฏิบัติงานใต้น้ำ มีลักษณะคล้ายคลึงกับ หุ่นจำลองของงานวิจัยที่ได้ศึกษา จึงได้หยิบทฤษฎีและสมการมาประยุกต์ใช้ เพื่อหาสมการการเคลื่อนที่ของหุ่นยนต์ดำน้ำ

รูปที่ 3.2 แบบหุ่นยนต์ดำน้ำ

3.2 สมการการเคลื่อนที่ของหุ่นยนต์ดำน้ำ

สำหรับแบบจำลองหุ่นยนต์ดำน้ำมีสมการการเคลื่อนที่ที่อาศัยตัวแรงขับจากมอเตอร์นั่นคือ ทอร์ก หรือแรงในรูปของการหมุน แรงขับหรือทอร์ก (ในที่นี้จะพูดถึงทอร์กของมอเตอร์) นั้นขึ้นอยู่กับความเร็วรอบในการหมุนยิ่งมีความเร็วรอบในการหมุนมากก็จะทำให้ทอร์กที่ได้มีค่ามากตามไปด้วย สำหรับหุ่นยนต์ดำน้ำนี้จะใช้สมการที่ควบคุมความเร็วรอบของมอเตอร์เพื่อให้ได้ทอร์กตามที่ต้องการออกมา รวมไปถึงการใช้ทอร์กเพื่อควบคุมตำแหน่งการเคลื่อนที่ของตัวหุ่นยนต์ดำน้ำถ้าเปรียบเทียบ $\tau = I\alpha$ ของการหมุน ก็คือ แรง $F = ma$ ในการเคลื่อนที่แบบเส้นตรงนั่นเอง

จาก $\Sigma F = ma$

ΣF_x; $$2T - F_{vis} = m\ddot{x}$$

$$2\frac{K_t E_a}{R} - K_{vis} V_{vis} = m\dot{V}$$

$$2\frac{K_t E_a}{R} - K_{vis}\dot{X}_x = m\ddot{X}_x$$

$$2\frac{K_t E_a}{R} - K_{vis}\dot{X}_x = m\ddot{X}_x$$

$$2\frac{K_t E_a}{R} - K_{vis}\dot{X}_x = m\ddot{X}_x$$

$$m\ddot{X}_x + K_{vis}\dot{X}_x - \frac{2K_t E_a}{R} = 0 \qquad (3.1)$$

แปลงลาปาสจะได้

$$ms^2 X(s) + KsX(s) - \frac{2K_t E_a(s)}{R} = 0$$

$$(ms^2 + Ks)X(s) = \frac{2K_t E_a(s)}{R}$$

$$X_x(s) = \frac{1}{s}\frac{2K_t E_a(s)}{(ms + K)R}$$

$$X_x(s) = \frac{2K_t E_a(s)}{(ms + K)R} \qquad (3.2)$$

$\Sigma Fy; \quad F = 0$ (กรณีไม่มีแรงจากมอเตอร์)

$\Sigma F_y;$ $4T - F_{vis} - F_b + mg = ma$ (กรณีมีแรงจากมอเตอร์)

$$4\frac{K_t E_a}{R} - K_{vis} V_{vis} - F_b + mg = ma$$

$$4\frac{K_t E_a}{R} - K_{vis} V_{vis} - \rho_L v_g g + mg = m\dot{V}$$

$$4\frac{K_t E_a}{R} - K_{vis} V_{vis} - (\rho_L v_g + m)g = m\dot{V}$$

$$4\frac{K_t E_a}{R} - K_{vis}\dot{X}_y - (\rho_L v_g + m)g = m\ddot{X}_y$$

$$m\ddot{X}_y + K_{vis}\dot{X}_y + (\rho_L v_g + m)g - 4\frac{K_t E_a}{R} = 0 \qquad (3.3)$$

แปลงลาปาสจะได้

$$ms^2X(s) + KsX(s) + (\rho_L v_g + m)g - 4\frac{K_t E_a(s)}{R} = 0$$

$$X_y(s) = \frac{1}{s}\frac{4K_t E_a(s) + (\rho_L v_g + m)g}{(ms + K)R}$$

$$X_y(s) = \frac{4K_t E_a(s) + (\rho_L v_g + m)g}{(ms + K)R} \qquad (3.4)$$

ดังนั้น

$$X(t) = \sqrt{(X_x)^2 + (X_y)^2} \qquad (3.5)$$

เมื่อ $X_x = \dfrac{2K_t E_a(s)}{(ms + K)R}$

$$X_y = \frac{4K_t E_a(s) + (\rho_L v_g + m)g}{(ms + K)R}$$

3.3 สมการการเคลื่อนที่ของมอเตอร์

ดีซีมอเตอร์ (DC Motor) ที่ใช้ร่วมกับดีซีแอมปลิไฟล์ (DC Amplifie) ทั้งในระบบการบังคับตำแหน่งและการบังคับความเร็วมักจะได้รับการประยุกต์ใช้เป็นส่วนประกอบสร้างกำลังงานในระบบการนำร่องและระบบบังคับต่างๆ และเนื่องจากวิทยาการเกี่ยวกับสารแม่เหล็กและการขยายด้วยโซลิคสเตททำให้ดีซีมอเตอร์แบบแม่เหล็กถาวรได้รับความนิยมใช้เป็นส่วนประกอบการขับเคลื่อนในระบบการบังคับแบบปิดลูฟต่างๆ มากขึ้น การออกแบบและการชดเชยระบบดังกล่าวได้อย่างเหมาะสมจะต้องใช้โมเดลทางคณิตศาสตร์เป็นส่วนประกอบทั้งหมดในระบบ ซึ่งสามารถแสดงได้ดังนี้

สมการแรงขับของมอเตอร์

$$T(t) = J\frac{d^2\theta(t)}{dt^2} + b\frac{d\theta(t)}{dt}$$

$$T(t) = J\ddot{\theta} + b\dot{\theta}$$

$$T(s) = Js^2[\dot{\theta}(s)] + bs[\dot{\theta}(s)]$$

$$T(s) = (Js^2 + bs)\dot{\theta}(s) \qquad (3.6)$$

สมการแรงดันไฟฟ้า

$$E_a(t) = L\frac{di(t)}{dt} + Ri(t) + k_b\frac{d\theta(t)}{dt}$$

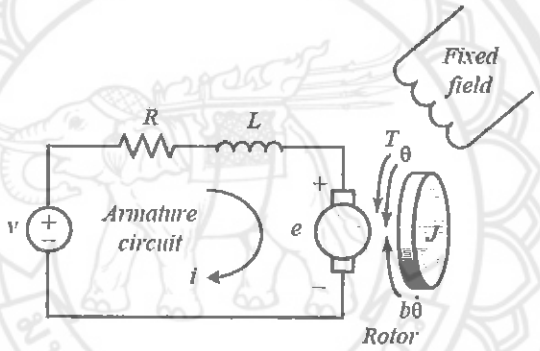
$$L\frac{di(t)}{dt} + Ri(t) = E_a(t) - k_b\frac{d\theta(t)}{dt}$$

$$LsI(s) + RI(s) = E_a(s) - Ks\dot{\theta}(s)$$

$$(Ls + R)I(s) = E_a(s) - Ks\dot{\theta}(s) \qquad (3.7)$$

3.4 การควบคุมความเร็วรอบของมอเตอร์

การควบคุมความเร็วของมอเตอร์กระแสตรงนั้น จะทำได้โดยการปรับแรงดันที่จ่ายให้มอเตอร์ และจะถูกจำกัดด้วยขนาดแรงดัน โดยสามารถปรับได้ไม่เกินค่าแรงดันสูงสุดที่จ่ายให้มอเตอร์ ส่วนการควบคุมแรงบิด ทำได้โดยการควบคุมกระแสที่ผ่านขดลวดอาเมเจอร์และกระแสที่จ่ายให้กับขดลวดสเตเตอร์ หรือใช้ตัวต้านทานปรับค่าแบบขดลวด (Wire Wound Resistor) แต่ก็จะเกิดกำลังไฟฟ้าสูญเสียที่ตัวต้านทาน ดังนั้นจึงมักนิยมใช้การควบคุม ด้วยวงจรพัลซ์วิธมอดูเลเตอร์ (Pulse Width Modulator) ซึ่งจะใช้วิธีจ่ายไฟให้แก่มอเตอร์เป็นช่วง ๆ โดยการควบคุมแรงดัน คือการปรับช่วงกว้างของพัลซ์ที่จ่ายให้ ซึ่งการใช้วงจรพัลซ์วิธมอดูเลเตอร์ จะทำให้ลดกำลังสูญเสียได้มาก



รูปที่ 3.3 รูปภาพแสดงตัวแปรที่เกี่ยวข้องสำหรับมอเตอร์กระแสตรง

ที่มา : http://ctms.engin.umich.edu

/CTMS/index.php?example=MotorSpeed&section=SystemModeling

กำหนดตัวแปรต่างๆ ที่มีผลต่อการเคลื่อนที่ของมอเตอร์

J คือ แรงเฉื่อยของ rotor

Kt คือ ค่าคงที่ของทอร์กในมอเตอร์

b คือ ค่าคงที่ของแรงเสียดทานเนื่องจากความหนืด

L คือ ขดลวดเหนี่ยวนำ

Ke คือ ค่าคงที่ของพลังงานจลน์

R คือ ค่าความต้านทานไฟฟ้า

หา mathematical model ได้จาก Newton's 2nd low และ Kirchhoff's voltage law

$$T = K_t i_a \tag{3.8}$$

$$E_a = K_b \dot{\theta} \tag{3.9}$$

จะเห็นได้ว่า $K_t = K_b = K$

จากสมการที่ (3.8) จะได้ $J\ddot{\theta} + b\dot{\theta} = Ki$ (3.10)

จากสมการที่ (3.9) จะได้ $L\frac{di}{dt} + Ri = E_a - K\dot{\theta}$ (3.11)

หา transfer function

จากสมการที่ (3.10) จะได้ $(Js+b)\dot{\theta}(s) = KI(s)$ (3.12)

จากสมการที่ (3.11) จะได้ $(Ls+R)I(s) = E_a(s) - Ks\dot{\theta}(s)$ (3.13)

นำ (3.13) / (3.12) จะได้ $\frac{E_a(s) - Ks\dot{\theta}(s)}{s(Js+b)\dot{\theta}(s)} = \frac{(Ls+R)I(s)}{KI(s)}$

$$KE_a(s) - K^2 s\dot{\theta}(s) = s(Ls+R)(Js+b)\dot{\theta}(s)$$

$$\frac{KE_a(s) - K^2 s\dot{\theta}(s)}{\dot{\theta}(s)} = s(Ls+R)(Js+b)$$

$$\frac{E_a}{\dot{\theta}_{(s)}} - Ks = \frac{s(Ls+R)(Js+b)}{K}$$

$$\frac{E_a}{\dot{\theta}_{(s)}} = \frac{s(Ls+R)(Js+b) + K^2 s}{K}$$

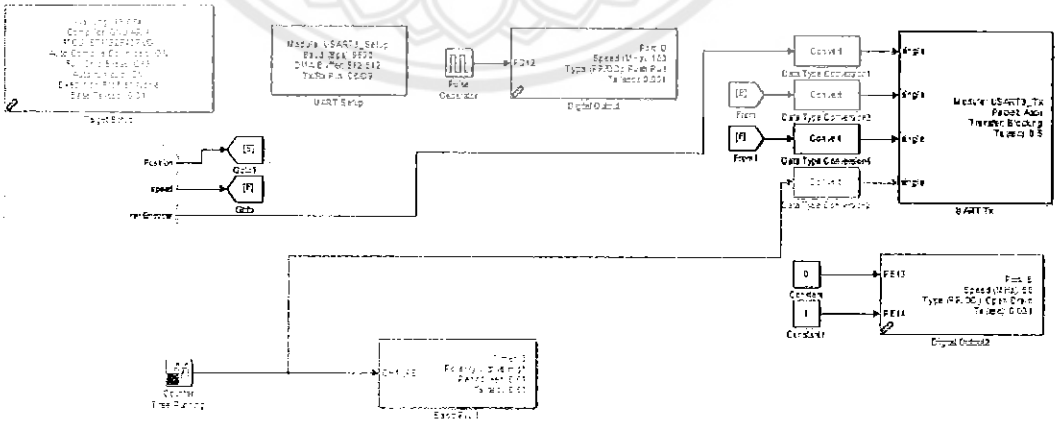
$$\frac{\dot{\theta}(s)}{E_a(s)} = \frac{K}{s(Ls+R)(Js+b) + K^2 s}$$

$$\frac{\dot{\theta}(s)}{E_a(s)} = \frac{1}{s}\frac{K}{(Ls+R)(Js+b) + K^2}$$

$$P(s) = \frac{\dot{\theta}(s)}{E_a(s)} = \frac{K}{(Ls+R)(Js+b) + K^2} \tag{3.14}$$

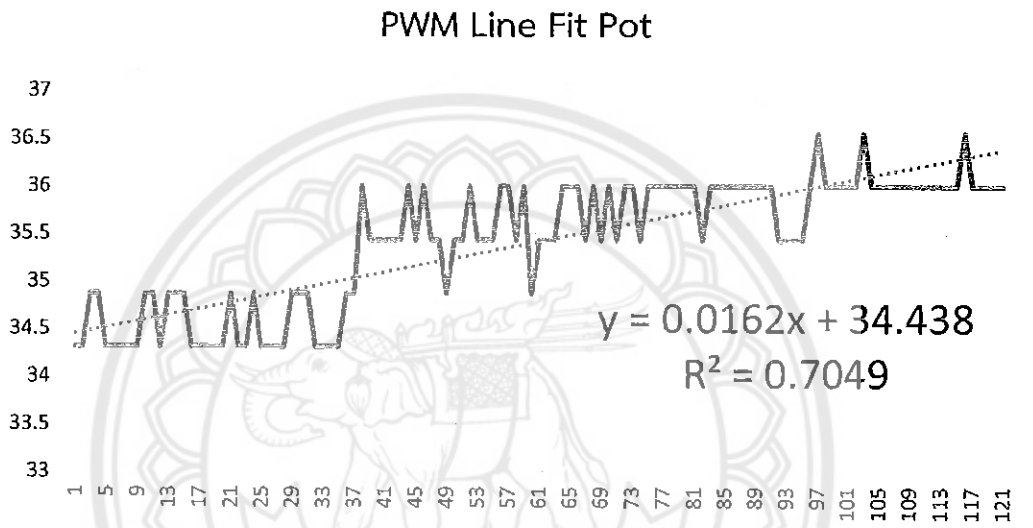
3.5 การสร้างแบบจำลองทางคณิตศาสตร์ด้วยชุดควบคุมสำเร็จรูป

การสร้างแบบจำลองคณิตศาสตร์ด้วยชุดควบคุมสำเร็จรูป ใช้โปรแกรมที่มีชื่อว่า Simulink เพื่อสั่งการหุ่นยนต์ดำน้ำ ทำได้โดยการใช้เครื่องมือต่างๆที่อยู่ภายในโปรแกรมมากมาย ซึ่งก็มีเครื่องมือหนึ่งที่ชื่อว่า waijung โดยที่จะต้องทำการติดตั้งโปรแกรมเสริมเพิ่มเข้าไปเพื่อที่โปรแกรมจะสามารถใช้งานร่วมกับบอร์ด STM34F4 Discovery ได้ การเขียนคำสั่งจะต้องมี input เพื่อสั่งการ และoutput เพื่อดูการตอบสนองของคำสั่ง โดยสามารถกำหนดเครื่องมือต่างๆได้ ไม่ว่าจะเป็น ระยะทาง ความเร็ว ความเร่ง เป็นต้น ซึ่งคำสั่งต่างๆต้องสอดคล้องกับการเคลื่อนที่ของหุ่นยนต์ดำน้ำ แต่เครื่องมือที่มีอยู่ในโประแกรมนั้นเป็นเพียงบล็อกที่วางสมการพื้นฐานไว้เพื่อให้สามารถนำมาใช้ง่าย จึงได้ทำการทดลองเพื่อหาสมการสมดุลของการเคลื่อนที่ โดยการทำการจ่ายเปอร์เซ็นต์ค่า ความต่างศักย์เข้าไปเป็นลำดับคือ 25% 50% 75% 100% แบบสุ่มคงที่ จากนั้นจะใช้โปรแกรมด๊อกไลท์ (Docklight) อ่านค่าการตอบสนองที่ได้เขียนไว้ เพื่อนำค่าที่อ่านได้ไปวิเคราะห์ แล้วหาสมการเส้นตรงออกมา เพราะการสร้างแบบแบบจำลองทางคณิตศาสตร์ด้วยรูปภาพ หรือ GUI (Graphic User Interface) ในโปรแกรม Simulink นั้น จำเป็นที่จะต้องมีสมการเพื่อให้ระบบสมบูรณ์มากยิ่งขึ้น ซึ่งจากการทดลองนั้นสามารถนำสมการทางคณิตศาสตร์ จากการใส่ input ทั้งสองแบบคือ constant และ counter มาปรับใช้เพื่อเขียนสมการคำสั่งในโปรแกรม Simulink ได้ เนื่องจากว่า input ทั้งสองแบบนั้น แสดงผลออกมาในแนวโน้มที่ไปในทิศทางเดียวกัน ความชันที่ได้ก็มีค่าเท่าๆกัน ซึ่งทั้งสองแบบนี้เป็นการสุ่มเลือกมาเพื่อการทดลอง สามารถใส่ input แบบอื่นได้ เพื่อเป็นการต่อยอดการทดลองที่ดีขึ้นต่อไป



รูปที่ 3.4 ชุดคำสั่งสำหรับควบคุมหุ่นยนต์ดำน้ำ

การป้อนสัญญาณ constant input แบบสุ่มค่าคงที่ เมื่อทำการทดลองและเก็บค่าข้อมูลที่ได้ ก็จะนำข้อมูลเหล่านั้น มาทำการหาสมการ Regression โดยการเปรียบเทียบเปอร์เซ็นต์การจ่ายค่า PWM เพื่อให้ได้ค่าความเร็วรอบออกมา หลังจากนั้นหาสมการเชิงเส้นออกมา เพื่ออธิบายการกระจายตัวของข้อมูลและจะสามารถตัดสินใจได้ว่า ข้อมูลที่ได้สามารถใช้งานได้จริงหรือไม่



รูปที่ 3.5 ความเร็วรอบของมอเตอร์ที่ใช้ในการทดสอบ

จากกราฟจะเห็นได้ว่าความเร็วรอบเป็นสัดส่วนโดยตรงต่อค่าการจ่ายเปอร์เซ็นต์ PWM แสดงให้เห็นโดยสมการ $y = 0.0162x + 34.438$ นั่นคือเมื่อทำการจ่ายเปอร์เซ็นต์ PWM เพิ่มขึ้น ก็จะทำให้มีค่าความเร็วรอบเพิ่มขึ้น เป็นความสัมพันธ์แบบแปรผันตรง โดยปกติแล้วค่า R^2 ยิ่งมีค่าใกล้ 1 มากๆจะแสดงให้เห็นว่าข้อมูลเป็นข้อมูลที่ดี แต่จากค่า $R^2 = 0.7049$ บนรูปที่ 3.5 แสดงให้เห็นถึงค่าข้อมูลที่แปรผันไป เนื่องจากว่าเราไม่ได้จ่ายค่า PWM คงที่แต่เราทำการจ่าย PWM แบบสุ่มค่า R^2 ที่ได้ออกมาถือว่ายอมรับได้ โดยการที่การป้อนค่า input แบบ counter ก็ให้ค่าออกมาคล้ายคลึงกัน

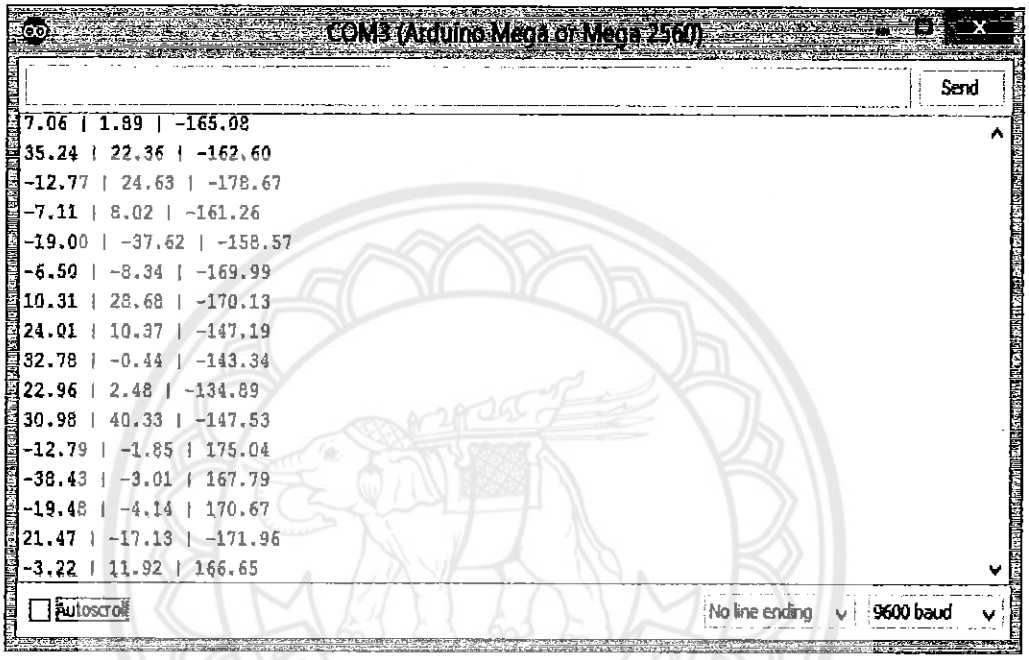
3.6 IMU

การเขียนโค้ดเพื่ออ่านค่า IMU ทำได้โดยการศึกษาโค้ดจากแหล่งอ้างอิงจากเว็บไซต์ที่มีชื่อว่า bildr.block เพื่อนำมาปรับใช้ให้เข้ากับการใช้งานของหุ่นยนต์ดำน้ำ โดนการนำไฟล์ที่มีชื่อว่า FreeSixIMU ไปไว้ในโฟลเดอร์ libraries ใน arduino เพื่อให้ arduino ดึงไฟล์ออกมาใช้งานได้ ทำให้สามารถอ่านค่ามุมเอียงของหุ่นยนต์ดำน้ำทั้ง 3 แนวแกนได้ แล้วนำค่ามุมเอียงที่อ่านค่าได้มาเป็นข้อมูลพื้นฐานในการสร้างโค้ดหลักให้หุ่นยนต์ดำน้ำสามารถเคลื่อนที่ได้ตามที่ต้องการ

```
#include <FreeSixIMU.h>
#include <FIMU_ADXL345.h>
#include <FIMU_ITG3200.h>
#include <Wire.h>

float angles[3]; // yaw pitch roll
// Set the FreeSixIMU object
FreeSixIMU sixDOF = FreeSixIMU();

void setup(){
  Wire.begin();
  delay(5);
  sixDOF.init(); //begin the IMU
  delay(5);
  Serial.begin(9600);
{

void loop(){
  sixDOF.getEuler(angles);
  Serial.print(angles[0]);
  Serial.print(" | ");
  Serial.print(angles[1]);
  Serial.print(" | ");
  Serial.println(angles[2]);
  delay(100);
}
```

รูปที่ 3.6 ตัวอย่างส่วนการทำงานของไจโรสโคป (Gyroscope)

เมื่อทำการต่อวงจรไฟฟ้าแต่ละบอร์ดเข้าด้วยกันแล้ว จะสามารถอ่านค่าของมุมเอียงในแต่ละแนวแกนของหุ่นยนต์ดำน้ำได้จากจอแสดงผลดังนี้ โดยค่ามุมเอียงเรียงลำดับจาก แกน X แกน Y แกน Z ตามลำดับ



รูปที่ 3.7 หน้าต่างแสดงผลของมุมเอียงในแนวแกนทั้ง 3 แนวแกน

3.7 Manual Operate

เป็นการใช้ Analog Joystick ควบคุมการเคลื่อนที่ของหุ่นยนต์ดำน้ำโดยผู้ควบคุมทำการควบคุมด้วยตนเอง ซึ่งก้านควบคุมเป็นตัวกำหนดทิศทางการเคลื่อนที่ โดยการสั่งการทำงานของ PWM ที่ได้ให้เหมาะสมกับทิศทางการเคลื่อนที่ โดยเมื่อโยกก้านควบคุมไปด้านหน้าจะทำให้ PWM ของมอเตอร์ทั้งสองมีค่าเท่ากัน แต่เมื่อโยกก้านควบคุมไปในทิศทางด้านซ้ายหรือด้านขวา จะทำให้ PWM ของมอเตอร์ที่อยู่ทิศทางตรงกันข้ามกับที่เลี้ยวมีค่า PWM มากกว่ามอเตอร์อีกตัว ทำให้เกิดการเปลี่ยนทิศทางการเคลื่อนที่ได้ตามต้องการ โดยมีลิงค์วีดีโอสำหรับการทดลองควบคุมหุ่นยนต์ดำน้ำในภาคผนวก ข สามารถเข้าไปดูเพื่อเป็นแนวทางการปฏิบัติได้

```
// analog read form joystick
    aX = analogRead(PIN_ANALOG_X);
    aY = analogRead(PIN_ANALOG_Y);
    // manual operate
    if ((aY-aYoffset) < 0 ) {
       PWM0 = 0;
    }
    else {
       PWM0 = (220*(aY-aYoffset)/yrang);
    }
    if ((aX-aXoffset) < (-xspan) ){
      TurnLeft();
    }
    else if ((aX-aXoffset) > xspan){
      TurnRight();
    }
    else if (((aX-aXoffset) >= (-xspan))&&(((aX-aXoffset) <= xspan))){
      Heading();
    }
    else {
      idelPosition();
    }
  }
```

รูปที่ 3.8 ตัวอย่างส่วนการทำงานของการควบคุมด้วยตนเอง (Manual Operate)

3.8 Auto Operate

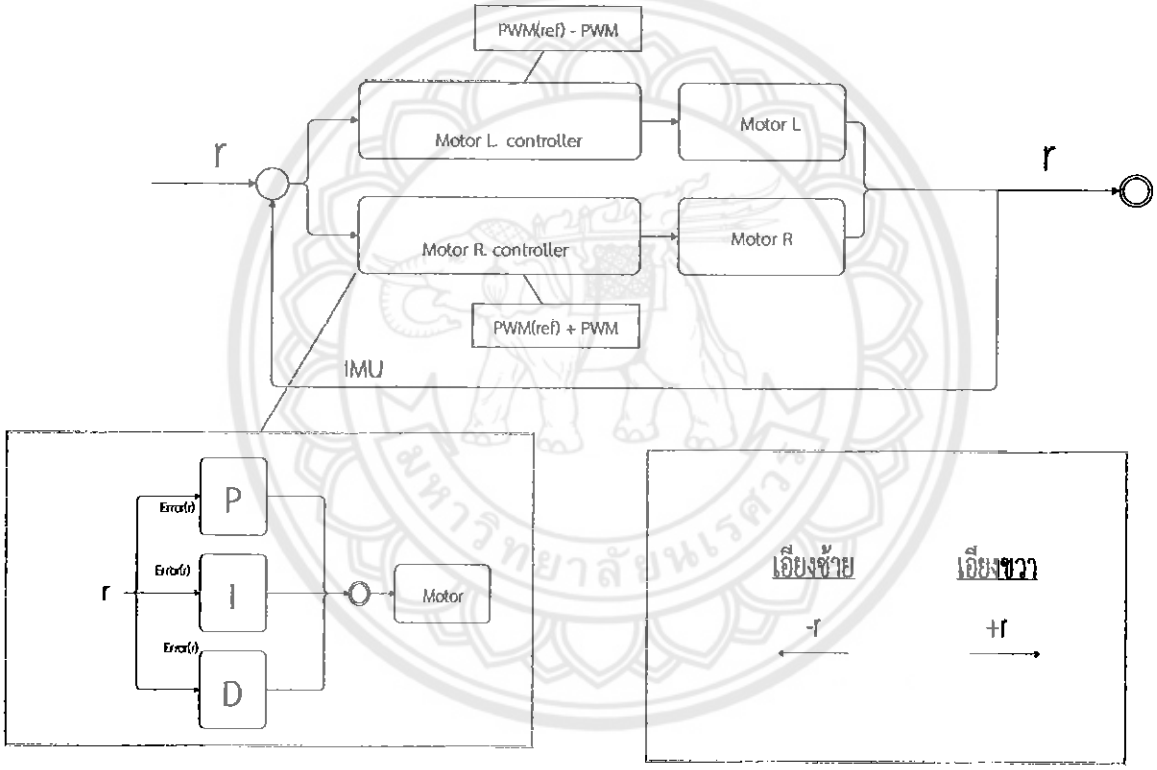
ในส่วนของการควบคุมแบบอัตโนมัติ เป็นการสั่งการแบบที่ไม่ต้องมีการบังคับ joystick ซึ่งสามารถทำได้โดยการเขียนโปรแกรมป้อนเส้นทางที่ต้องการ เพื่อให้หุ่นยนต์ดำน้ำเคลื่อนที่ได้แบบอัตโนมัติ ซึ่งโค้ดที่ทำการสร้างขึ้นมาสามารถเปลี่ยนแปลงได้ เมื่อต้องการเส้นทางที่แตกต่างไปจากเดิม สิ่งสำคัญที่ทำให้หุ่นยนต์ดำน้ำเคลื่อนที่ได้อย่างสมดุลคือการอ่านค่ามุมเอียงมาจาก IMU แล้วทำการอ่านค่า Error ของมอเตอร์ทั้งสองตัว เพื่อนำค่า Error ที่ได้ไปปรับค่า PWM ของมอเตอร์ทั้งสอง โดยจะแสดงได้จากสมการดังนี้

$$\mathbf{Error = angles(0) - offsetYaw} \tag{3.15}$$

$$\mathbf{PWM1 = (K * Error) + 220} \tag{3.16}$$

$$\mathbf{PWM2 = (K * (-Error)) + 220} \tag{3.17}$$

สามารถอธิบายกระบวนการทำงานแบบ Close loop ได้โดยแผนภาพดังต่อไปนี้



จากแผนภาพสามารถอธิบายการทำงานของการรับค่าสัญญาณจาก IMU จากนั้นระบบจะทำการป้อนกลับค่าสัญญาณและคิดค่า error ของมุมที่อ่านค่าออกมาได้ก่อนจะส่งไปยังส่วนของระบบการควบคุมเพื่อทำการปรับค่า PWM แล้วส่งไปยังมอเตอร์หรือตัวส่งกำลัง

การที่จะเปลี่ยนเส้นทางของหุ่นยนต์ดำน้ำ สามารถตั้งค่าว่าจะให้หุ่นยนต์ดำน้ำเป็นไปในทิศทางใด โดยการเรียงลำดับการเคลื่อนที่ของหุ่นยนต์ดำน้ำ ซึ่งสามารถอธิบายด้วยตัวอย่างโค้ดดังนี้

```
// automatic operate
    sixDOF.getEuler(angles);
    if (initial == 0){
      offsetYaw = angles[2];
      initial = 1;
    }
    Heading();
    Serial.print("Heading");
    Serial.println();
    delay(1000);
    TurnRight();
    Serial.print("Right");
    Serial.println();
    delay(100);
    TurnLeft();
    Serial.print("Left");
    Serial.println();
    delay(100);
    Heading();
    Serial.print("Heading");
    Serial.println();
    delay(1000);
    TurnLeft();
    Serial.print("Left");
    Serial.println();
    delay(100);
```

รูปที่ 3.9 ตัวอย่างส่วนการทำงานของการดำเนินงานแบบอัตโนมัติ (Auto Operate)

บทที่ 4

บทสรุปและข้อเสนอแนะ

4.1 สรุปผลการดำเนินโครงงาน

จากผลการดำเนินโครงงานจะได้แบบจำลองทางคณิตศาสตร์ออกมาเพื่อควบคุมหุ่นยนต์ดำน้ำ โดยแบบจำลองทางคณิตศาสตร์ที่ได้คือ ความเร็วรอบของมอเตอร์กระแสตรงต่อแรงดันไฟฟ้า ตัวส่งกำลังขับต่อความเร็วรอบของมอเตอร์กระแสตรง ตำแหน่งการเคลื่อนที่ของหุ่นยนต์ดำน้ำต่อตัวส่งกำลังขับ ซึ่งความสัมพันธ์ดังกล่าว สามารถควบคุมตำแหน่งการเคลื่อนที่ของหุ่นยนต์ดำน้ำได้ด้วยแรงดันไฟฟ้าที่จ่ายให้กับมอเตอร์กระแสตรงซึ่งก็คือ input ที่จ่ายเข้าไป และทำให้ได้แรงขับที่ต้องการออกมาได้ซึ่งก็คือ output ที่ตอบสนองออกมานั่นเอง เมื่อนำสมการที่ได้ออกมาไปใช้กับระบบควบคุมแบบ PID จะสามารถทำให้หุ่นยนต์ดำน้ำทำงานแบบอัตโนมัติได้ดีและมีความเสถียรมากยิ่งขึ้น ซึ่งการออกแบบแบบจำลองทางคณิตศาสตร์นี้ ทำให้หุ่นยนต์ดำน้ำเคลื่อนที่ได้อย่างมีประสิทธิภาพ เป็นไปตามวัตถุประสงค์ที่ได้คาดหวังไว้

4.2 ปัญหา

4.2.1 เมื่อนำหุ่นยนต์ดำน้ำไปทดสอบกับสภาพแวดล้อมจริงพบว่า มอเตอร์ทั้งสองที่นำมาใช้งานมีกำลังขับไม่เท่ากัน ทำให้การควบคุมทิศทางการเคลื่อนที่ของหุ่นยนต์ดำน้ำทั้งแบบควบคุมโดยผู้ใช้เองและแบบอัตโนมัติ ไม่สามารถเป็นไปได้ตามที่ได้กำหนดค่าเอาไว้

4.2.2 บอร์ดอาดูอิโน่ (Arduino) รุ่น Mega 2560 ที่นำมาใช้งาน ไม่สามารถทำงานได้เมื่ออ่านค่ามุมเอียงจาก IMU ไปพร้อมกับการจ่าย PWM ให้กับมอเตอร์ ซึ่งทำการแก้ปัญหาโดยการเขียนโค้ดให้หุ่นยนต์ดำน้ำเคลื่อนที่ไปแบบอัตโนมัติ โดยไม่ได้ทำการรับค่ามุมเอียงจาก IMU สมมุติว่าการเคลื่อนของหุ่นยนต์ดำน้ำมีมุมเอียงที่เท่ากัน ถึงจะมีมุมเอียงที่ไม่เท่ากันก็จะไม่เท่ากันอย่างคงที่ หุ่นยนต์ดำน้ำก็จะรักษามุมเอียงนี้ไปตลอดการเคลื่อนที่ โดยสามารถศึกษาความรู้เพิ่มเติมได้ที่วีดีโอในเว็บไซต์ยูทูป (youtube) ซึ่งสามารถดูได้ที่ภาคผนวก ข.

4.3 ข้อเสนอแนะ

4.3.1 การสร้างหุ่นยนต์ดำน้ำต้นแบบ ควรสร้างกล่องสำหรับใส่บอร์ดต่างๆ ให้เล็กกะทัดรัด เพื่อความสะดวกในการเคลื่อนย้ายเพราะน้ำหนักที่มีผลต่อการเคลื่อนที่

4.3.2 สายไฟที่ใช้กับหุ่นยนต์ดำน้ำต้นแบบ ควรใช้สายไฟเดี่ยวขนาด 0.5 มิลลิเมตรจะไม่ทำให้ทิศทางการเคลื่อนที่ของหุ่นยนต์ดำน้ำเปลี่ยนไป และช่องที่จะนำสายไฟหรืออุปกรณ์อื่นๆออกมาจากตัวหุ่นยนต์ดำน้ำ ควรที่จะเป็นเป็นปลั๊กเสียบกันน้ำเพราะจะทำให้ไม่มีปัญหาเรื่องน้ำเข้าไปทำให้บอร์ดต่างๆเสียหาย แต่เนื่องจากว่าราคาของปลั๊กเสียบกันน้ำมีราคาค่อนข้างสูง จึงควรพิจารณาเรื่องนี้ด้วย

4.3.3 มอเตอร์ที่ใช้ขับหุ่นยนต์ดำน้ำ ควรมีกำลังที่เท่ากันทั้งสองตัวหรือใกล้เคียงกันมากที่สุด เพื่อการเขียนโปรแกรมและกำหนดค่าให้กับหุ่นยนต์ดำน้ำนั้น จะได้เกิดค่าความผิดพลาดน้อยที่สุด

4.3.4 เนื่องจากการทดสอบของโครงงานนี้ใช้แบบ Step function ซึ่งทำให้ค่าความถดถอยมีค่าไม่เหมาะสม ฟังก์ชั่นในการทดสอบ ควรเป็นฟังก์ชั่นที่มีความต่อเนื่อง เช่น สมการเส้นตรง สมการเอ็กซ์โพเนนเชียล สมการลอการึทึม

เอกสารอ้างอิง

[1] Lionel Lapierre.(November 25 ,2008,).Robust diving control of an AUV.Retrieved October 10,2013, (Online) October 30 , 2015,from http://www.sciencedirect.com/science/article/pii/S002980180800228X

[2] สงกรานต์ แก้วสองศรี .(22 พฤษภาคม 2556). Simulink คือ อะไร. (ออนไลน์) สืบค้นเมื่อ 12 มกราคม 2558,จาก http://whatsimulink.blogspot.com/2013/05/simulink.html

[3] มงคล พรหมเทศ. งานไฟฟ้าทั่วไป. กรุงเทพฯ : เอมพันธ์, 2548.

ไวพจน์ ศรีธัญ. เครื่องกลไฟฟ้ากระแสตรง. กรุงเทพฯ : วังอักษร, 2548.

"มอเตอร์ไฟฟ้ากระแสตรง,"(ออนไลน์) สืบค้นเมื่อ 20 มกราคม 2558 จาก http://edu.e-tech.ac.th/mdec/learning/e-web/sara01.htm, 2009.

[4]"DC Series Motor,"(ออนไลน์) สืบค้นเมื่อ 20 มกราคม 2558 from http://www.tpub.com/neets/book5/32NE0440.GIF, 2009

[5] Mr. Premier (March 13, 2009,).การควบคุม Servo motor. Retrieved January 24, 2013, (Online) Febuary 25 2015 fromhttp://www.premier-ac.co.th/index.php?lay=boardshow&ac=webboard_show&No=1195843

[6] Prof. Bill Messner (January 1, 2012,).Control Tutorials. Retrieved January 24, 2013, (Online) Febuary 25 2015 from http://ctms.engin.umich.edu/CTMS/index.php?example=MotorSpeed&section=SystemModeling

[7] บุญสืบ โพธิ์ศรี ; และคนอื่น ๆ. (2550).ชนิดของตัวต้านทาน.(ออนไลน์) สืบค้นเมื่อ 1 มีนาคม 2558, จาก http://www.atom.rmutphysics.com/charud/oldnews/0/285/22/resistor1/resistor2.html

[8] วิบูลย์ บุญยธโรกุล.(2550). ลักษณะและการทำงานของปั๊ม. (ออนไลน์) สืบค้นเมื่อ 3 มีนาคม 2558 , จาก http://202.129.59.73/tn/september54/Pump1.htm

[9] ห้องปฏิบัติการระบบสมองกลฝังตัว สาขาวิศวกรรมคอมพิวเตอร์ (CprE) ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ (KMUTNB). (17 ตุลาคม 2013).การสร้างสัญญาณ PWM สำหรับ R/C Servo และ ESC. (ออนไลน์) สืบค้นเมื่อ 15 มีนาคม 2558 จาก http://cpre.kmutnb.ac.th/esl/learning/index.php?article=analog-joystick-pwm

[10] Siriwimon Sunthon. (สิงหาคม 2557). ศึกษาข้อมูลของบอร์ด Arduino Mega2560. (ออนไลน์) สืบค้นเมื่อ 20 เมษายน 2558, จาก http://mbeddedweekly.blogspot.com/2014/08/arduino-mega2560.html

[11] อ.มนตรี สุขเลื่อง.(2550). การจำลองมอเตอร์ไฟฟ้ากระแสตรง DC Motor Simulation. (ออนไลน์) สืบค้นเมื่อ 20 เมษายน 2558, จาก http://www.atom.rmutphysics.com/charud/oldnews/0/286/15/14/DCMOTOR/motorDC_simulation_files/frame.htm

[12] ฟิสิกส์ราชมงคล.(2550). การควบคุมมอเตอร์. (ออนไลน์) สืบค้นเมื่อ 25 เมษายน 2558, จาก http://www.rmutphysics.com/charud/scibook/electric4/topweek9.htm

ภาคผนวก ก

ขั้นตอนการสร้างแบบจำลองทางคณิตศาสตร์ด้วยรูปภาพ

หรือ GUI (Graphic User Interface)

1.ขั้นตอนการเขียน Encoder

1. ต่อวงจรตามรูปที่ 3.1 Encoder

2. ตรวจสอบและระวังการต่อระบบไฟฟ้าเลี้ยง +5Vdc +12Vdc และ GND ให้ถูกต้อง

3. สร้าง Model และตั้งค่าในเมนู Simulation → Configuration Parameters

3.1 Start time: 0

3.2 Stop time: inf

3.3 Type: Fix-Step

3.4 Solver: Discrete

4. วาง block ที่จำเป็นดังนี้

Blockset ที่ใช้งานรูปที่ 3.1 Encoder

Target setup ซึ่งอยู่ใน waijung blockset →STM32F4 Target →Device configuration

UART Setup ซึ่งอยู่ใน waijung blockset →STM32F4 Target →On-Chip peripherals → UART

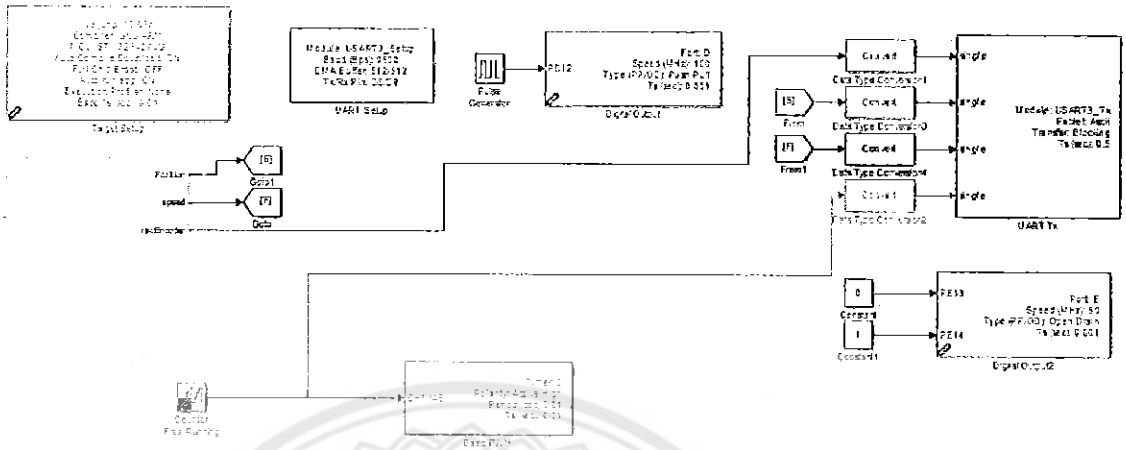
UART Tx ซึ่งอยู่ใน waijung blockset →STM32F4 Target →On-Chip peripherals → UART

Digital Output ซึ่งอยู่ใน waijung blockset →STM32F4 Target →On-Chip peripherals →IO

Basic PWM ซึ่งอยู่ใน waijung blockset →STM32F4 Target →On-Chip peripherals →TIM

Pulse Generator ซึ่งอยู่ใน Simulink →Source

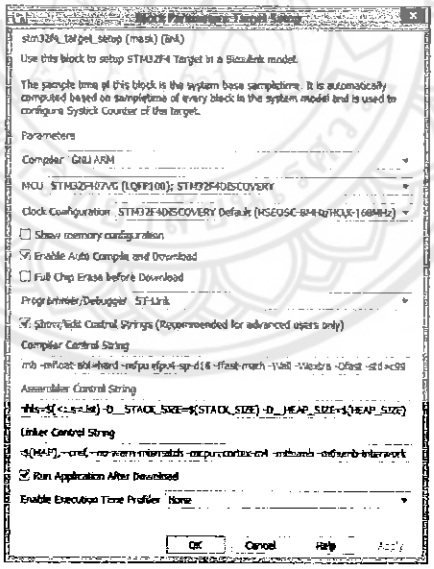
Counter Free-Running ซึ่งอยู่ใน Simulink → Source



รูปที่ ก.1 Encoder

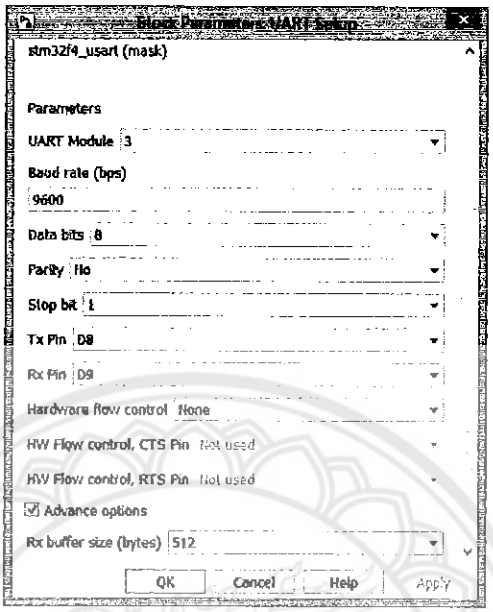
1.1 การตั้งค่าในโปรแกรม Encoder เป็นดังนี้

Target setup



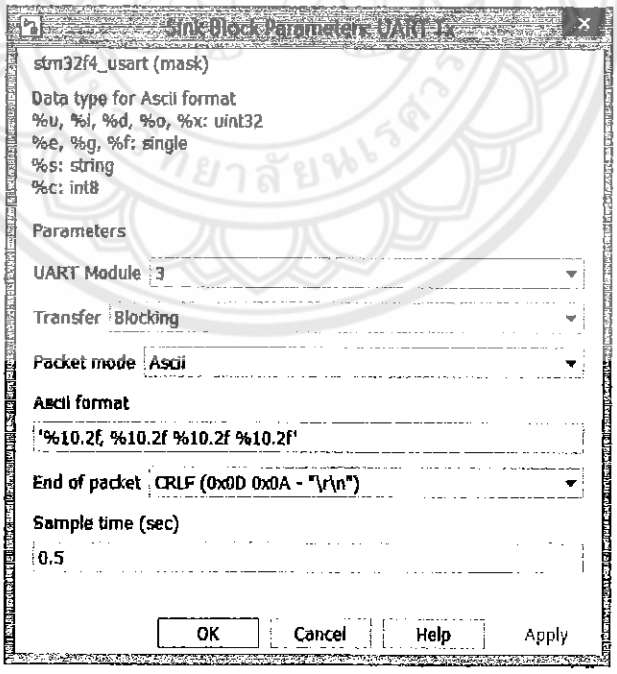
รูปที่ ก.2 Target setup

UART Setup



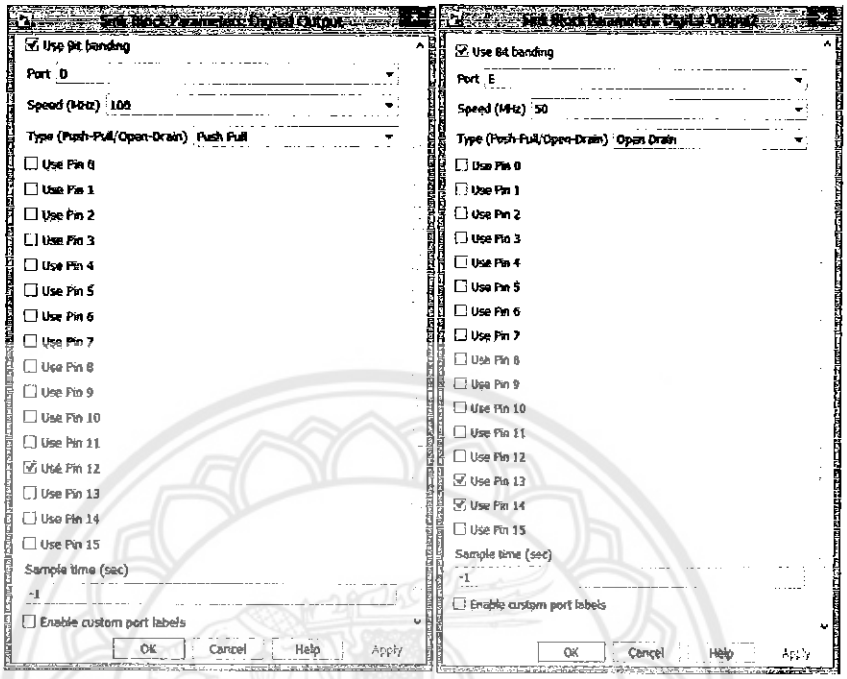
รูปที่ ก.3 UART Setup

UART Tx



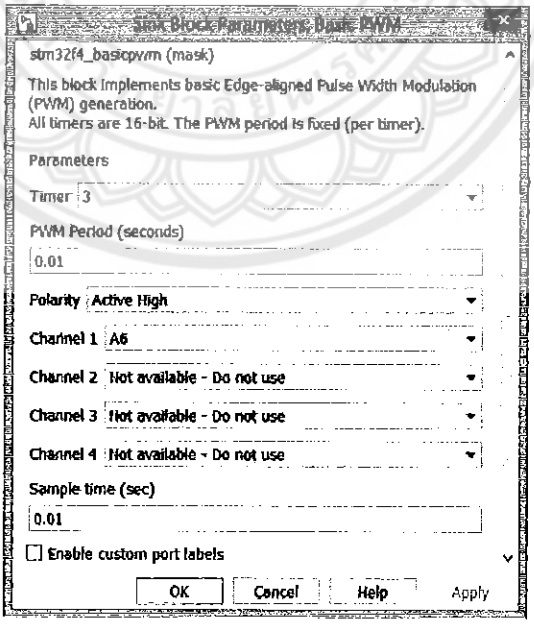
รูปที่ ก.4 UART Tx

Digital Output 1,2



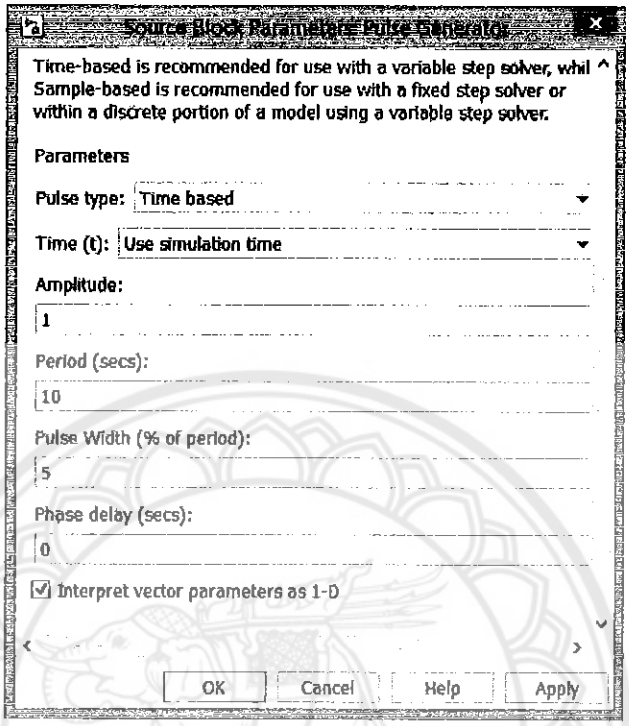
รูปที่ ก.5 Digital Output 1,2

Basic PWM



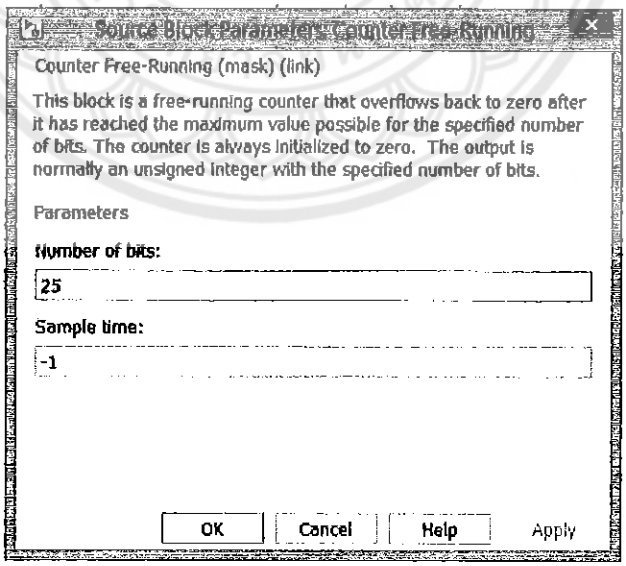
รูปที่ ก.6 Basic PWM

Pulse Generator



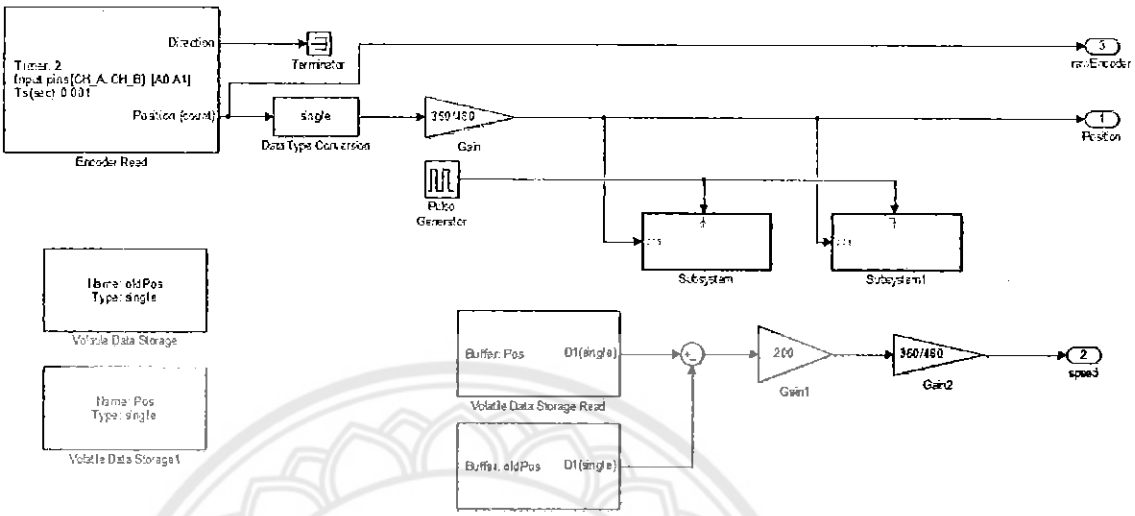
รูปที่ ก.7 Pulse Generator

Counter Free-Running



รูปที่ ก.8 Counter Free-Running

2 ขั้นตอนการเขียนภายในของ subsystem



รูปที่ ก.9 Subsystem

Blockset ที่ใช้งานรูปที่ 3.9 Subsystem

Encoder Read: ซึ่งอยู่ใน waijung blockset→STM32F4 Target→On-Chip peripherals→TIM

Volatile Data Storage: ซึ่งอยู่ใน waijung blockset→ Hardware Modules →Character LCD→Simple aMG CLCD Display Setup→LCD Setup

Data Type Conversion: ซึ่งอยู่ใน Simulink→Commonly Used Blocks

Terminator: ซึ่งอยู่ใน Simulink→Commonly Used Blocks

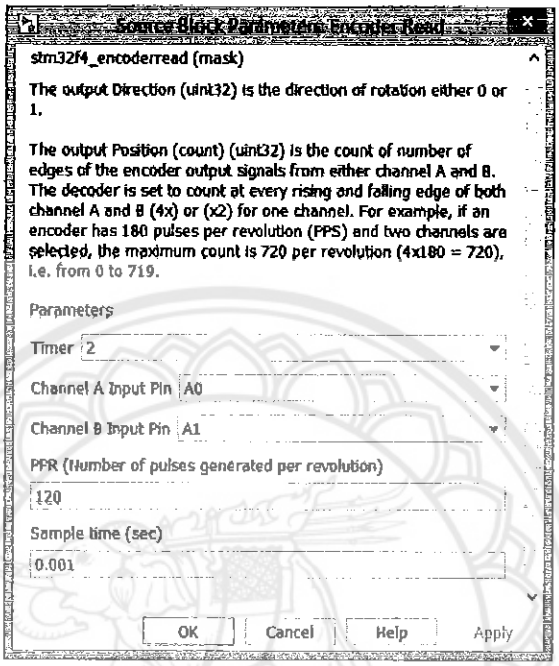
Gain: ซึ่งอยู่ใน Simulink→Commonly Used Blocks

Trigger: ซึ่งอยู่ใน Simulink→ Commonly Used Blocks

Pulse Generator: ซึ่งอยู่ใน Simulink→Source

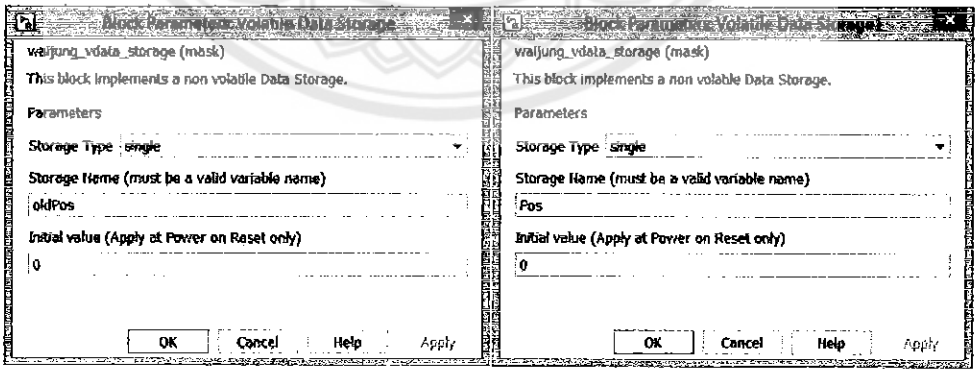
2.1 การตั้งค่าในโปรแกรมเป็นดังนี้

Encoder Read



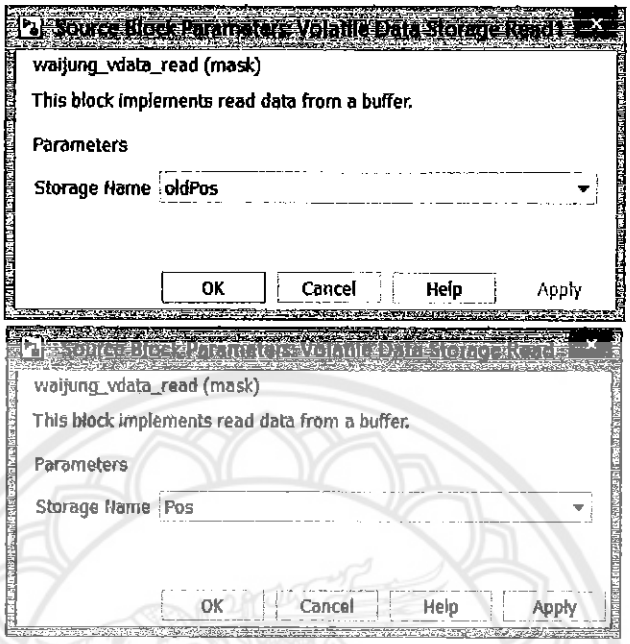
รูปที่ ก.10 Encoder Read

Volatile Data Storage (Out of date) 1,2



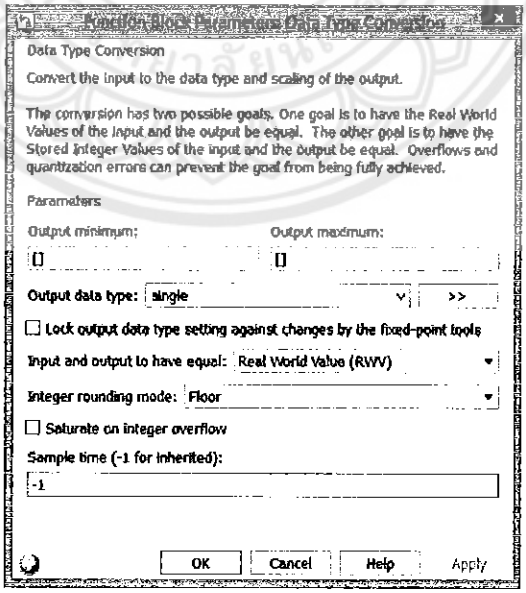
รูปที่ ก.11 Volatile Data Storage (Out of date) 1,2

Volatile Data Storage Read 1,2



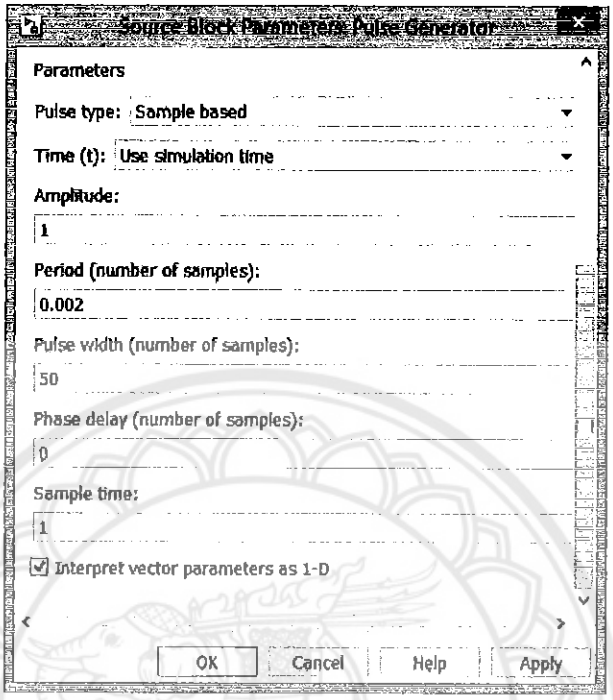
รูปที่ ก.12 Volatile Data Storage Read 1,2

Data Type Conversion



รูปที่ ก.13 Data Type Conversion

Pulse Generator



รูปที่ ก.14 Pulse Generator

ภาคผนวก ข

โค้ดสำหรับการควบคุมการเคลื่อนที่ของหุ่นยนต์ดำน้ำ

```
#include <FreeSixIMU.h>
#include <FIMU_ADXL345.h>
#include <FIMU_ITG3200.h>
#include <Wire.h>


int M1CW  = 7;
int M1CCW = 9;
int M1PWM = 10;

int M2CW  = 11;
int M2CCW = 12;
int M2PWM = 13;

unsigned char PWM0 = 0;
unsigned char PWM1 = 0;
unsigned char PWM2 = 0;
int incomingByte = 0;
const byte PIN_ANALOG_X = A0;
const byte PIN_ANALOG_Y = A1;
int aX, aY;
int xrang, yrang, aXoffset, aYoffset, xspan;
float angles[3]; // yaw pitch roll
// Set the FreeSixIMU object
FreeSixIMU sixDOF = FreeSixIMU();

const int buttonD3 = 3;  // Manual
const int buttonD4 = 4;  // Stop
const int buttonD5 = 5;  // Auto
int buttonD3State = HIGH;
int lastButtonD3State = LOW;
int buttonD4State = HIGH;
int lastButtonD4State = LOW;
int buttonD5State = HIGH;
int lastButtonD5State = LOW;
long lastDebounceD3Time = 0;
long debounceD3Delay = 50;
long lastDebounceD4Time = 0;
long debounceD4Delay = 50;
long lastDebounceD5Time = 0;
long debounceD5Delay = 50;

int OperateMode;
```

```
void TurnLeft()
{
    PWM1 = PWM0*0.5;
    PWM2 = PWM0;
    digitalWrite(M1CW,HIGH);
    digitalWrite(M1CCW,LOW);
    digitalWrite(M2CW,HIGH);
    digitalWrite(M2CCW,LOW);
    analogWrite(M1PWM, PWM1);
    analogWrite(M2PWM, PWM2);
}

void TurnRight()
{
    PWM1 = PWM0;
    PWM2 = PWM0*0.5;
    digitalWrite(M1CW,HIGH);
    digitalWrite(M1CCW,LOW);
    digitalWrite(M2CW,HIGH);
    digitalWrite(M2CCW,LOW);
    analogWrite(M1PWM, PWM1);
    analogWrite(M2PWM, PWM2);
}

void Heading()
{
    PWM1 = PWM0;
    PWM2 = PWM0;
    digitalWrite(M1CW,HIGH);
    digitalWrite(M1CCW,LOW);
    digitalWrite(M2CW,HIGH);
    digitalWrite(M2CCW,LOW);
    analogWrite(M1PWM, PWM1);
    analogWrite(M2PWM, PWM2);
}

void idelPosition()
{
    PWM1 = 0;
    PWM2 = 0;
    digitalWrite(M1CW,HIGH);
    digitalWrite(M1CCW,HIGH);
    digitalWrite(M2CW,HIGH);
    digitalWrite(M2CCW,HIGH);
    analogWrite(M1PWM, PWM1);
    analogWrite(M2PWM, PWM2);
}
```

```
void displayOutput()
{
  // display result
  Serial.print("x:");
  Serial.print(aX);
  Serial.print(" ");

  Serial.print("y:");
  Serial.print(aY);
  Serial.print(" ");

  Serial.print("PWM:");
  Serial.print(PWM0);

  Serial.print(" ");
  Serial.print(angles[0]);
  Serial.print(" | ");
  Serial.print(angles[1]);
  Serial.print(" | ");
  Serial.println(angles[2]);

  Serial.println();
}

void D3check() // Manual
{
   int readingD3 = digitalRead(buttonD3);
  if (readingD3 != lastButtonD3State) {
    lastDebounceD3Time = millis();
  }
  if ((millis() - lastDebounceD3Time) > debounceD3Delay) {
    if (readingD3 != buttonD3State) {
      buttonD3State = readingD3;
      if (buttonD3State == HIGH) {
        Serial.print("D3");
        Serial.println();
        OperateMode = 2;
      }
    }
  }
  lastButtonD3State = readingD3;
}
```

```
void D4check() // Stop
{
    int readingD4 = digitalRead(buttonD4);
  if (readingD4 != lastButtonD4State) {
    lastDebounceD4Time = millis();
  }
  if ((millis() - lastDebounceD4Time) > debounceD4Delay) {
    if (readingD4 != buttonD4State) {
      buttonD4State = readingD4;
      if (buttonD4State == HIGH) {
        Serial.print("D4");
        Serial.println();
        OperateMode = 0;
      }
    }
  }
  lastButtonD4State = readingD4;
}
void D5check() // Auto
{
  int readingD5 = digitalRead(buttonD5);
  if (readingD5 != lastButtonD5State) {
    lastDebounceD5Time = millis();
  }
  if ((millis() - lastDebounceD5Time) > debounceD5Delay) {
    if (readingD5 != buttonD5State) {
      buttonD5State = readingD5;
      if (buttonD5State == HIGH) {
        Serial.print("D5");
        Serial.println();
        OperateMode = 1;
      }
    }
  }
  lastButtonD5State = readingD5;
}
```

```
void setup()
{
  pinMode(M1CW,OUTPUT);
  pinMode(M1CCW,OUTPUT);

  pinMode(M2CW,OUTPUT);
  pinMode(M2CCW,OUTPUT);

  pinMode(3,INPUT);
  pinMode(4,INPUT);
  pinMode(5,INPUT);

  Wire.begin();
  delay(5);
  sixDOF.init(); //begin the IMU
  delay(5);
  Serial.begin(9600);

  aXoffset = analogRead(PIN_ANALOG_X);
  aYoffset = analogRead(PIN_ANALOG_Y)+10;
  xrang = 1023-aXoffset;
  yrang = 1023-aYoffset;
  xspan = 10;
}
void loop()
{
  D3check();
  D4check();
  D5check();

  if (OperateMode == 2) {
    // analog read form joystick
    aX = analogRead(PIN_ANALOG_X);
    aY = analogRead(PIN_ANALOG_Y);

    // manual operate
    if ((aY-aYoffset) < 0 ) {
       PWM0 = 0;
    }
    else {
       PWM0 = (220*(aY-aYoffset)/yrang);
    }
    if ((aX-aXoffset) < (-xspan) ){
      TurnLeft();
    }
    else if ((aX-aXoffset) > xspan){
      TurnRight();
    }
    else if (((aX-aXoffset) >= (-xspan))&&(((aX-aXoffset) <= xspan))){
```

```
    Heading();
  }
  else {
    idelPosition();
  }
}
else if (OperateMode == 1){3

      // automatic operate
      PWM0 = 220;
      sixDOF.getEuler(angles);
      Heading();
      Serial.print("Heading");
      Serial.println();
      delay(5000);
      TurnLeft();
      Serial.print("Left");
      Serial.println();
      delay(1000);
      Heading();
      Serial.print("Heading");
      Serial.println();
      delay(5000);
      TurnRight();
      Serial.print("Right");
      Serial.println();
      delay(1000);
      Heading();
      Serial.print("Heading");
      Serial.println();
      delay(5000);
      OperateMode = 0;
      }
      else if (OperateMode == 0) {
        idelPosition();
      }
```

ลิงค์วีดีโอสำหรับการทดลองควบคุมหุ่นยนต์ดำน้ำ (Manual Operate)

- https://www.youtube.com/watch?v=OqAxi-_5PKo

ลิงค์วีดีโอสำหรับการทดลองควบคุมหุ่นยนต์ดำน้ำ (Auto Operate)

- https://www.youtube.com/watch?v=eEQ3garvg7o

ประวัติผู้จัดทำโครงงาน

ชื่อ-นามสกุล นายอภิชิต นาคำ
วันเดือนปีเกิด วันอังคาร ที่ 19 มกราคม 2536
ที่อยู่ บ้านเลขที่ 285 หมู่ 4 ตำบลทับคล้อ อำเภอทับคล้อ จังหวัดพิจิตร 66150
ประวัติการศึกษา
- พ.ศ.2553 จบชั้นมัธยมศึกษาตอนปลายที่โรงเรียนตะพาหินจังหวัดพิจิตร
- ปัจจุบันกำลังศึกษาระดับปริญญาตรี ชั้นปีที่ 4 สาขาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ชื่อ-นามสกุล นายอภิรัฐ นาคำ
วันเดือนปีเกิด วันอังคาร ที่ 19 มกราคม 2536
ที่อยู่ บ้านเลขที่ 285 หมู่ 4 ตำบลทับคล้อ อำเภอทับคล้อ จังหวัดพิจิตร 66150
ประวัติการศึกษา
- พ.ศ.2553 จบชั้นมัธยมศึกษาตอนปลายที่โรงเรียนตะพาหินจังหวัดพิจิตร
- ปัจจุบันกำลังศึกษาระดับปริญญาตรี ชั้นปีที่ 4 สาขาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ชื่อ-นามสกุล นายกิตตินันทน์ กันแก้ว
วันเดือนปีเกิด วันศุกร์ ที่ 2 ตุลาคม 2535
ที่อยู่ บ้านเลขที่ 129 หมู่ 3 ตำบลผางาม อำเภอเวียงชัย จังหวัดเชียงราย 57210
ประวัติการศึกษา
- พ.ศ.2553 จบชั้นมัธยมศึกษาตอนปลายที่โรงเรียนเมืองเชียงราย
- ปัจจุบันกำลังศึกษาระดับปริญญาตรี ชั้นปีที่ 4 สาขาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร