

การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือ
โดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติ
Model Development for Boat Control
by Automatic Coordinates and Directions Detection

ธนกร อ่อนละมูล

580510598

การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่
ปีการศึกษา 2561

การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือ
โดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติ
Model Development for Boat Control
by Automatic Coordinates and Directions Detection

ธนกร อ่อนละมูล
580510598

การค้นคว้าอิสระนี้ได้รับการพิจารณาอนุมัติให้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่
ปีการศึกษา 2561

คณะกรรมการสอบการค้นคว้าอิสระ

..... ประธานกรรมการ
(อาจารย์ ดร.ศุภกิจ อาวิพันธุ์)

..... กรรมการ
(อาจารย์ ดร.วิจักขณ์ ศรีสังจะเลิศวาจา)

วันที่ 21 เดือน พฤษภาคม พ.ศ. 2562

กิตติกรรมประกาศ

ในการพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัตินี้ ผู้พัฒนา ขอขอบพระคุณผู้ที่คอยให้คำแนะนำและเป็นที่ปรึกษาตลอดการดำเนินการการค้นคว้าอิสระ ตั้งแต่การเริ่มดำเนินงานจนสำเร็จลุล่วงไปด้วยดี

ขอกราบขอบพระคุณ อาจารย์ ดร.ศุภกิจ อาวิพันธุ์ อาจารย์ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่ ซึ่งเป็นที่ปรึกษาการค้นคว้าอิสระ ผู้คอยให้คำปรึกษาเกี่ยวกับการออกแบบเครือข่ายที่ใช้ในระบบและให้ข้อเสนอแนะในการค้นคว้าอิสระนี้

ขอกราบขอบพระคุณ อาจารย์ ดร.วิจักขณ์ ศรีสัจจะเลิศวาจา อาจารย์ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่ ที่กรุณารับเป็นกรรมการสอบการค้นคว้าอิสระนี้ รวมทั้งยังคอยให้คำแนะนำเกี่ยวกับการทำเอกสารเล่มนี้

ขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.ทินกร กันยานี อาจารย์ภาควิชาเคมี คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่ ซึ่งเป็นผู้สนับสนุนให้เกิดโครงการนี้ขึ้น นอกจากนี้ยังให้แนวคิดเกี่ยวกับการออกแบบโครงสร้างของระบบในโครงการนี้

ขอขอบพระคุณ นายคำรณ อรุณเรือ ผู้ช่วยวิจัยห้องปฏิบัติการวิจัยการจำลองขนาดใหญ่ ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ ซึ่งเป็นผู้ที่คอยให้ความรู้และคำแนะนำเกี่ยวกับการประยุกต์ใช้งานอุปกรณ์สื่อสารระหว่างสรรพสิ่ง รวมทั้งให้ข้อเสนอแนะในการแก้ปัญหาในการจัดทำโครงการ

ขอขอบคุณ นายพีรพัชร อรุณเรือ นักศึกษาจากภาควิชาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าคุณทหารลาดกระบัง ซึ่งเป็นผู้ที่คอยให้คำปรึกษาและคำแนะนำเกี่ยวกับการใช้งานอุปกรณ์อิเล็กทรอนิกส์อย่างถูกต้อง

สุดท้ายนี้ผู้จัดทำจึงขอขอบพระคุณทุกท่านที่เป็นส่วนหนึ่งในการทำให้เกิด การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติมา ณ โอกาสนี้

นายธนกร อ่อนละมูล

580510598

หัวข้อการค้นคว้าอิสระ การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติ

ชื่อเจ้าของโครงการ นายธนกร อ่อนละมุล **รหัสประจำตัว** 580510598

วิทยาศาสตร์บัณฑิต สาขาวิทยาการคอมพิวเตอร์

อาจารย์ที่ปรึกษา อาจารย์ ดร.ศุภกิจ อวิปันธุ์

บทคัดย่อ

การทดลองวิเคราะห์คุณภาพของน้ำในแหล่งน้ำธรรมชาติที่มีบริเวณกว้างและมีความลึกมาก มีความจำเป็นที่ต้องใช้เรือเพื่อเดินทางไปยังจุดที่ต้องการเก็บตัวอย่างน้ำ ทำให้ต้องใช้เวลาในการเตรียมความพร้อมและมีความเสี่ยงที่จะเกิดข้อผิดพลาดต่าง ๆ ได้ จากปัญหาดังกล่าว โครงการนี้จึงมีวัตถุประสงค์ที่จะสร้างเรือที่สามารถเคลื่อนที่ตามพิกัดที่ระบุ และสามารถนำอุปกรณ์เซนเซอร์ (Sensor) มาติดตั้ง และส่งข้อมูลกลับมายังสถานีฐานได้แบบทันที

เรือขับเคลื่อนกึ่งอัตโนมัติถูกพัฒนาขึ้นด้วยอุปกรณ์ที่อยู่ภายใต้แนวคิดอินเทอร์เน็ตของสรรพสิ่งหรือไอโอที (IoT: Internet of Things) และใช้การรับส่งข้อมูลด้วยเทคโนโลยีเอ็นบีไอโอที (NB-IoT: Narrow Band Internet of Things) เรือจะประกอบด้วย 2 ฟังก์ชัน ได้แก่ ฟังก์ชันขับเคลื่อนอัตโนมัติมีหน้าที่สำหรับควบคุมการเคลื่อนที่ของเรือให้ไปยังพิกัดที่ระบุไว้แบบกึ่งอัตโนมัติและฟังก์ชันกู้เรืออัตโนมัติมีหน้าที่สำหรับควบคุมเรือกลับมาที่จุดเริ่มต้นในกรณีฉุกเฉิน แอปพลิเคชันจะประกอบด้วย 2 ส่วน ได้แก่ ส่วนสำหรับควบคุมเรือและส่วนสำหรับติดตามทำงานของเรือ อุปกรณ์อิเล็กทรอนิกส์ที่ใช้พัฒนาเรือ ได้แก่ Arduino Mega 2560 R3, Devio NB-Shield I, โมดูลจีพีเอส, โมดูลเข็มทิศดิจิทัล, โวลต์เตจเซนเซอร์และโมดูลขับเคลื่อนมอเตอร์ ซอฟต์แวร์และภาษาโปรแกรมที่ใช้ ได้แก่ Arduino IDE และ C++ สำหรับพัฒนาระบบภายในเรือ, Android Studio และ Java สำหรับพัฒนาแอปพลิเคชัน, Python และ PHP สำหรับพัฒนาเซิร์ฟเวอร์และจัดเก็บข้อมูลด้วย MySQL

ผลการทดสอบเรือและแอปพลิเคชัน มีดังนี้ แอปพลิเคชันสามารถส่งและรับข้อมูลระหว่างเรือโดยใช้เซิร์ฟเวอร์เป็นตัวกลางได้ เรือสามารถเคลื่อนที่ไปยังพิกัดที่ระบุและหยุดอยู่ที่จุดดังกล่าวตามช่วงเวลาที่กำหนดไว้เพื่อส่งข้อมูลกลับไปยังเซิร์ฟเวอร์ และสุดท้ายเรือจึงจะกลับมายังจุดเริ่มต้น พิกัดที่เรือหยุดมีความคลาดเคลื่อนไปจากพิกัดที่ระบุ 5 เมตรโดยประมาณ ซึ่งเป็นความคลาดเคลื่อนที่เกิดจากอุปกรณ์รับสัญญาณจีพีเอส

Independent Study Title	Model Development for Boat Control by Automatic Coordinates and Directions Detection	
Author	Mr. Tanakorn Onlamoon	Student ID 580510598
Bachelor of Science	Computer Science	
Supervisor	Dr. Suphakit Awiphan	

Abstract

Experimental analysis of water quality in natural sources of water that vast and deep is necessary to use a boat for travelling to the location where the sample is needed. This increases the time to prepare and poses more risk of errors. Due to the problems mentioned above, this project therefore purposes to develop a boat that can move according to the specified coordinated and additional sensor equipment can be equipped with the boat and send the data back to the base station in a real-time manner.

The automatic boat was developed with devices based on the concept of IoT (Internet of Things) and transmit data with NB-IoT (Narrow Band Internet of Things) technology. Firstly, the boat will consist of 2 functions: the automatic driving function, which is responsible for controlling the motion identification of the boat to the specified coordinates in a automatic manner and the automatic salvage function is responsible for controlling the boat back to the starting point in an emergency. Secondly, the application will consist of 2 features: the boat controlling feature and the boat's operational status tracking feature. Thirdly, electronic devices used to develop the boat include Arduino Mega 2560 R3, Devio NB-Shield I, GPS module, digital compass module, voltage sensor and motor driver module. Finally, software and programming language used include Arduino IDE and C++ for developing internal systems, Android Studio and Java for application development, Python and PHP for server development and store data with MySQL database.

Experimental results for the boat and the application are as follows. Firstly, the application can send and receive data between the boat via the server. Secondly, the boat can move to the specified coordinates and stop at that point for a specified period to send the data back to the server. Thirdly, the boat will return to the starting point. In conclusion, the coordinates of the ship stop are approximately 5 meters from the specified coordinates which is the error caused by the GPS receiver device

สารบัญ

หัวข้อ	หน้า
กิตติกรรมประกาศ	ก
บทคัดย่อ	ข
ABSTRACT	ค
สารบัญ	จ
สารบัญตาราง	ช
สารบัญภาพ	ณ
บทที่ 1 บทนำ	1
1.1 หลักการและเหตุผล	1
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ประโยชน์ที่จะได้รับจากการศึกษาเชิงประยุกต์และเชิงทฤษฎี	2
1.4 ขอบเขตของโครงการ/วิธีการวิจัย	2
1.5 แผนการดำเนินงานและระยะเวลาดำเนินงาน	4
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	6
2.1 อินเทอร์เน็ตของสรรพสิ่ง	6
2.2 เอ็นพีไอโอที	7
2.3 เครือข่ายคลาวด์	9
2.4 ระบบกำหนดตำแหน่งบนโลก	10
2.5 ระยะทางยูคลิด	10
2.6 ภาคทฤษฎี	11
2.7 อาวุธโน้มน้าว	12
2.8 บอร์ดเสริมเอไอเอสเอ็นพีไอโอที	13
2.9 โมดูลขับเคลื่อนมอเตอร์ L298N	14

สารบัญ (ต่อ)

หัวข้อ	หน้า
2.10 โมดูลจีพีเอส Ublox NEO-M8N	17
2.11 โมดูลเชื่อมต่อจีทีเอสสามแกน DA5883 GY-273	18
2.12 โวลเตจเซนเซอร์	18
บทที่ 3 การวิเคราะห์และออกแบบระบบ	19
3.1 ปัญหาและความต้องการของระบบ	19
3.2 ลักษณะการทำงานของระบบ	20
3.3 การออกแบบสถาปัตยกรรมของเรือ	21
3.4 แผนผังวงจรอิเล็กทรอนิกส์	22
3.5 แผนภาพกิจกรรม	24
3.6 แผนภาพลำดับ	28
3.7 แผนภาพสถานะ	37
บทที่ 4 การออกแบบส่วนต่อประสานกับผู้ใช้	41
4.1 การออกแบบส่วนต่อประสาน	41
4.2 การทำงานของส่วนต่อประสาน	42
บทที่ 5 การพัฒนาและทดสอบระบบ	46
5.1 การพัฒนาระบบ	46
5.2 การทดสอบระบบ	51
บทที่ 6 บทสรุป	55
6.1 สรุปผลการดำเนินการ	55
6.2 ปัญหาและข้อจำกัด	56
6.3 ข้อเสนอแนะ	56
เอกสารอ้างอิง	57

สารบัญ (ต่อ)

หัวข้อ	หน้า
ภาคผนวก ก	60
ภาคผนวก ข	65

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 1.1 ขั้นตอนการดำเนินงานและระยะเวลาการดำเนินงาน	5
ตารางที่ 2.1 ภาคทิศที่วัดจากทิศเหนือวนตามเข็มนาฬิกา	11
ตารางที่ 2.2 รายละเอียดของบอร์ดพัฒนาอาดุยโนเมกะ 2560	12
ตารางที่ 2.3 รายละเอียดของบอร์ดเสริมเอ็นพีไอโอที	14
ตารางที่ 2.4 ตัวอย่างคำสั่งสำหรับใช้งานบอร์ดเสริมเอไอเอสเอ็นพีไอโอที	14
ตารางที่ 2.5 รายละเอียดของโมดูลขับเคลื่อนมอเตอร์ L298N	15
ตารางที่ 2.6 การควบคุมทิศทางการหมุนของโมดูลขับเคลื่อนมอเตอร์ L298N	16
ตารางที่ 2.7 รายละเอียดของโมดูลจีพีเอส Ublox NEO-M8N	17
ตารางที่ 2.8 โมดูลเชื่อมต่อจีพีเอสสามแกน DA5883 GY-273	18
ตารางที่ 3.1 สัญลักษณ์และความหมายของแผนผังวงจรอิเล็กทรอนิกส์	22
ตารางที่ 3.2 สัญลักษณ์และความหมายของแผนภาพกิจกรรม	24
ตารางที่ 3.3 สัญลักษณ์และความหมายของแผนภาพลำดับ	28
ตารางที่ 3.4 สัญลักษณ์และความหมายของแผนภาพสถานะ	37
ตารางที่ 3.5 แสดงตารางสถานะของเรือ	40
ตารางที่ 5.1 การตรวจสอบความพร้อมทำงานของเรือ	52
ตารางที่ 5.2 การควบคุมมอเตอร์	53
ตารางที่ 5.3 การส่งข้อมูลระหว่างเรือและแอปพลิเคชันผ่านทางเซิร์ฟเวอร์	53
ตารางที่ 5.4 การทำงานของระบบขับเคลื่อนอัตโนมัติ	54

สารบัญภาพ

ภาพ	หน้า
ภาพที่ 2.1 ลักษณะของระบบไอโอที	6
ภาพที่ 2.2 ลักษณะของระบบเอ็นบีไอโอที	8
ภาพที่ 2.3 ประโยชน์ของเอ็นบีไอโอที	8
ภาพที่ 2.4 ลักษณะของเครือข่ายคลาวด์	9
ภาพที่ 2.5 การระบุตำแหน่งของจีพีเอส	10
ภาพที่ 2.6 ส่วนบนของทรงกลมบนท้องฟ้า	11
ภาพที่ 2.7 Arduino Mega 2560 Rev3	12
ภาพที่ 2.8 โมดูลขับเคลื่อนมอเตอร์ L298N	15
ภาพที่ 2.9 กราฟแสดงแรงดันไฟฟ้าเฉลี่ยจากตัวตรวจจับ	15
ภาพที่ 2.10 การทำงานของวงจรเอชบริดจ์เพื่อควบคุมทิศทางการหมุนของมอเตอร์กระแสตรง	16
ภาพที่ 2.11 โมดูลจีพีเอส Ublox NEO-M8N และเสาอากาศ	17
ภาพที่ 2.12 โมดูลเชื่อมต่อ GY-273 HMC5883L	18
ภาพที่ 2.13 โวลต์จิกเซนเซอร์	18
ภาพที่ 3.1 ลักษณะการทำงานของระบบ	20
ภาพที่ 3.2 อุปกรณ์ภายในเรือขับเคลื่อนอัตโนมัติ	21
ภาพที่ 3.3 แผนผังวงจรการเชื่อมต่อของอุปกรณ์อิเล็กทรอนิกส์ภายในเรือ	23
ภาพที่ 3.4 แผนภาพกิจกรรมของแอปพลิเคชัน	25
ภาพที่ 3.5 แผนภาพกิจกรรมของแอปพลิเคชันส่วนติดตามสถานะของเรือ	25
ภาพที่ 3.6 แผนภาพกิจกรรมของแอปพลิเคชันส่วนควบคุมเรือ	26
ภาพที่ 3.7 แผนภาพกิจกรรมของเรือ	27
ภาพที่ 3.8 แผนภาพลำดับการทำงานของเรือ	30
ภาพที่ 3.9 แผนภาพลำดับการติดตั้งอุปกรณ์	31

สารบัญภาพ (ต่อ)

ภาพ	หน้า
ภาพที่ 3.10 แผนภาพลำดับการตรวจสอบข้อผิดพลาดของอุปกรณ์ภายในเรือ	32
ภาพที่ 3.11 แผนภาพลำดับการควบคุมเรือไปยังจุดหมาย	33
ภาพที่ 3.12 แผนภาพลำดับการควบคุมเรือกลับมายังจุดเริ่มต้น	34
ภาพที่ 3.13 แผนภาพลำดับการติดตามสถานะของเรือผ่านแอปพลิเคชัน	35
ภาพที่ 3.14 แผนภาพลำดับการควบคุมเรือผ่านแอปพลิเคชัน	36
ภาพที่ 3.15 แผนภาพสถานะของเรือ	38
ภาพที่ 3.16 แผนภาพสถานะของแอปพลิเคชัน	39
ภาพที่ 4.1 การทำงานของส่วนต่อประสาน	41
ภาพที่ 4.2 หน้าจอหลัก	42
ภาพที่ 4.3 หน้าจอควบคุมเรือ	43
ภาพที่ 4.4 หน้าจอแสดงสถานะของเรือ	44
ภาพที่ 4.5 หน้าจอสำหรับเลือกตำแหน่งบนแผนที่กูเกิลแมพ	45
ภาพที่ 5.1 เรือขับเคลื่อนอัตโนมัติ	46
ภาพที่ 5.2 การหาผลต่างของทิศที่เรือหันและทิศจากเรือไปยังเป้าหมาย	47
ภาพที่ 5.3 ช่วงของการตัดสินใจเลือกทิศทาง	48
ภาพที่ 5.4 การตรวจจับพิกัดของเรือและระยะห่างไปยังเป้าหมาย	49
ภาพที่ 5.5 แหล่งน้ำธรรมชาติ ณ ศูนย์ธรรมชาติวิทยาดอยสุเทพ	51
ภาพที่ ก.1 ไฟล์ .apk ของแอปพลิเคชัน	60
ภาพที่ ก.2 สอบถามการติดตั้งแอปพลิเคชัน	61
ภาพที่ ก.3 กำลังติดตั้งแอปพลิเคชัน	62
ภาพที่ ก.4 ติดตั้งแอปพลิเคชันสำเร็จ	63
ภาพที่ ก.5 การทำงานของยูทิลิตี้เซิร์ฟเวอร์	64

สารบัญภาพ (ต่อ)

ภาพ	หน้า
ภาพที่ ก.6 ไฟล์เอพีไอ	64
ภาพที่ ข.1 หน้าจอหลัก	65
ภาพที่ ข.2 หน้าจอควบคุมเรือ	66
ภาพที่ ข.3 หน้าจอแสดงสถานะของเรือ	67
ภาพที่ ข.4 หน้าจอสำหรับเลือกตำแหน่งบนแผนที่กูเกิลแมพ	68

บทที่ 1

บทนำ

การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติ เป็นการคว่ำอิสระสำหรับหลักสูตรปริญญาตรีวิทยาศาสตร์บัณฑิต สาขาวิทยาการคอมพิวเตอร์ โดยการศึกษาค้นคว้าและพัฒนาแอนดรอยด์แอปพลิเคชัน (Android Application) เพื่อสั่งงานให้เรือเคลื่อนที่ไปยังพิกัดปลายทางตามค่าที่ระบุไว้ และมีการประยุกต์ใช้การติดต่อสื่อสารภายใต้แนวคิดเอ็นบีไอโอที เพื่อส่งข้อมูลระหว่างเรือขับเคลื่อนอัตโนมัติกับสถานีฐานให้สามารถควบคุมการเคลื่อนที่และติดตามสถานะของเรือได้ผ่านแอปพลิเคชันได้ ในส่วนของบทนำผู้จัดทำได้นำเสนอ หลักการและเหตุผล วัตถุประสงค์ของโครงการ ประโยชน์ที่คาดว่าจะได้รับ ขอบเขตของโครงการ แผนการดำเนินงานและระยะเวลาการดำเนินงาน

1.1 หลักการและเหตุผล

ในยุคปัจจุบันที่ความต้องการสินค้าอุตสาหกรรมเพิ่มมากขึ้น ส่งผลให้มีโรงงานที่ผลิตสินค้าประเภทนี้มากขึ้นตามไปด้วย ซึ่งในขั้นตอนการผลิตก็นำมาสู่ของเสียต่าง ๆ ที่ถูกปล่อยออกมาจากโรงงานอุตสาหกรรม เช่น น้ำเสีย ไอเสีย เป็นต้น โดยสิ่งเหล่านี้สามารถทำลายทั้งป่าไม้ แม่น้ำหรือแหล่งน้ำจนอาจทำให้ระบบนิเวศของแหล่งธรรมชาติเสียสมดุล ด้วยเหตุนี้การทดลองวัดคุณภาพของแหล่งธรรมชาติจึงถือเป็นสิ่งสำคัญที่จะช่วยบ่งชี้ว่าแหล่งธรรมชาตินั้นยังมีสภาพสมบูรณ์อยู่หรือไม่

การทดลองวัดคุณภาพของแหล่งธรรมชาติต่าง ๆ มีขั้นตอนที่แตกต่างกัน ในกรณีของการวัดคุณภาพแหล่งน้ำ ขั้นตอนการเก็บตัวอย่างน้ำเพื่อนำกลับมาทดลองก็ถือเป็นหนึ่งในขั้นตอนที่สำคัญ เนื่องจากเป็นปัจจัยหลักที่สามารถใช้บ่งชี้ได้ว่าแหล่งน้ำมีความสมบูรณ์หรือมีสภาพปนเปื้อนก็คือน้ำจากแหล่งน้ำนั้น ๆ ถ้าหากแหล่งน้ำนั้นมีความกว้างและความลึกของน้ำมาก ก็จำเป็นที่จะต้องใช้เรือเพื่อไปยังยังจุดที่ต้องการเก็บตัวอย่างน้ำ ทำให้มักจะเสียเวลาและเสี่ยงต่อการเกิดอุบัติเหตุได้ อีกทั้งการเก็บตัวอย่างน้ำจะทำได้ค่อนข้างยากหากน้ำในแหล่งน้ำมีความปนเปื้อนของสารเคมีสูง ซึ่งอาจทำให้เกิดอันตรายหากสัมผัสกับมนุษย์โดยตรง

จากปัญหาที่กล่าวมาข้างต้น ผู้จัดทำจึงได้ศึกษาและคิดพัฒนาเรือที่มีระบบขับเคลื่อนให้ไปถึงตำแหน่งที่ต้องการจะเก็บตัวอย่างน้ำแบบอัตโนมัติด้วยเทคโนโลยีการสื่อสารเอ็นบีไอโอที เพื่อเป็นอุปกรณ์ช่วยสำหรับการเก็บตัวอย่างน้ำในแหล่งน้ำ ทั้งแหล่งน้ำที่มีลักษณะเป็นบึงกว้าง ลึก หรือแหล่งน้ำที่มีการปนเปื้อนสูง ซึ่งเรือขับเคลื่อนอัตโนมัตินี้จะสามารถช่วยประหยัดเวลา ลดแรงงานคนและช่วยลดอุบัติเหตุที่อาจเกิดขึ้นจากการเดินทางโดยเรือได้

1.2 วัตถุประสงค์ของโครงการ

- 1) เพื่อพัฒนาเรือขับเคลื่อนอัตโนมัติโดยใช้เทคโนโลยีการสื่อสารเ็นบีไอโอที
- 2) เพื่อพัฒนาแอนดรอยด์แอปพลิเคชันสำหรับควบคุมและติดตามสถานะของเรือ
- 3) เพื่อพัฒนาเซิร์ฟเวอร์สำหรับสร้างช่องทางสื่อสารระหว่างเรือและแอปพลิเคชัน

1.3 ประโยชน์ที่จะได้รับจากการศึกษาเชิงประยุกต์และเชิงทฤษฎี

สามารถนำเรือขับเคลื่อนอัตโนมัติไปใช้ในการเก็บตัวอย่างน้ำ โดยการควบคุมหุ่นยนต์ระยะไกลหรือนำไปประยุกต์ใช้กับงานทางน้ำด้านอื่น ๆ ได้

1.4 ขอบเขตของโครงการ/วิธีการวิจัย

1.4.1 ขอบเขตทางสถาปัตยกรรม

เครื่องมือที่ใช้ประกอบการพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติ มีดังนี้

- 1) ฮาร์ดแวร์ (Hardware) ที่ใช้ในการพัฒนาระบบ ประกอบด้วย
 - 1.1) เครื่องคอมพิวเตอร์ชนิดพกพา มีหน่วยประมวลผล Intel Core i7 2.6 GHz ฮาร์ดดิสก์ (Hard Disk) ขนาดความจุ 1 เทระไบต์ (Terabyte) และหน่วยความจำหลักขนาด 16 กิกะไบต์ (Gigabyte)
 - 1.2) บอร์ดพัฒนา Arduino Mega 2560 R3
 - 1.3) บอร์ดเสริม Devio NB-Shield I
 - 1.4) โมดูลขับเคลื่อนมอเตอร์ L298N
 - 1.5) โมดูลจีพีเอส Ublox NEO-M8N
 - 1.6) โมดูลเข็มทิศดิจิทัลสามแกน DA5883 GY-273
 - 1.7) โวลต์จเซนเซอร์
 - 1.8) มอเตอร์กระแสตรงจำนวน 2 ตัว
 - 1.9) ใบพัด เพลาลเหล็กและท่อเพลาลเหล็กของเรือ FT012 จำนวนอย่างละ 2 ชิ้น
 - 1.10) แบตเตอรี่ 12 โวลต์ (Volt) 3 เซลล์ (Cells) 1500 มิลลิแอมป์ (Milliamps)
 - 1.11) โฟมสำหรับเป็นวัสดุสร้างเรือ
 - 1.12) กล่องพลาสติกสำหรับบรรจุอุปกรณ์อิเล็กทรอนิกส์
- 2) ซอฟต์แวร์ (Software) ที่ใช้ในการพัฒนาระบบ ประกอบด้วย
 - 2.1) ระบบปฏิบัติการไมโครซอฟท์วินโดวส์เอ็นเอดูเคชัน (Microsoft Window 10 Education)
 - 2.2) ระบบปฏิบัติการเซนต์โอเอสเซเว่น (CentOS 7)

- 2.3) โปรแกรมอาดุยโนไอดีอี (Arduino IDE) ใช้ในการพัฒนาระบบของเรือด้วยภาษาซีพลัสพลัส (C++ Programming Language)
- 2.4) โปรแกรมแอนดรอยด์สตูดิโอ (Android Studio) ใช้ในพัฒนาแอปพลิเคชันสำหรับควบคุมและติดตามการทำงานของเรือด้วยภาษาจาวา (Java Programming Language)
- 2.5) ภาษาไพธอน (Python Programming Language) และภาษาพีเอชพี (PHP Programming Language) สำหรับพัฒนาเซิร์ฟเวอร์ (Server)
- 3) ระบบที่ได้ทำการพัฒนาขึ้นนี้ทำงานผ่านเครือข่ายคลาวด์ (Cloud Network)

1.4.2 ขอบเขตของระบบงาน

การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติจะแบ่งระบบออกเป็นทั้งหมด 3 ส่วน คือ ยานพาหนะในรูปแบบของเรือ แอปพลิเคชันสำหรับควบคุมและติดตามสถานะของเรือ และเซิร์ฟเวอร์สำหรับสร้างช่องทางการสื่อสารระหว่างเรือและแอปพลิเคชัน โดยจะแบ่งฟังก์ชันออกได้ ดังนี้

1) ยานพาหนะสำหรับติดตั้งอุปกรณ์อิเล็กทรอนิกส์

ยานพาหนะจะถูกออกแบบเป็นลักษณะของเรือ ซึ่งจะมีฟังก์ชันการทำงานทั้งหมด 2 ฟังก์ชัน คือ ค้นหาเส้นทางและนำเรือไปยังจุดหมายและนำเรือกลับมาเมื่อการทำงานของเรือขัดข้อง

1.1) ระบบขับเคลื่อนอัตโนมัติ

เป็นการตรวจหาพิกัดและทิศทาง เพื่อนำเรือไปยังจุดหมายด้วยการควบคุมมอเตอร์สองข้างร่วมกับโมดูลต่าง ๆ ภายในเรือ ลำดับการทำงานของฟังก์ชันนี้สามารถอธิบายได้ดังนี้

- กำหนดให้ตำแหน่งปัจจุบันของเรือเป็นจุดเริ่มต้น
- ผู้ใช้กำหนดจุดหมายบนแผนที่ในแอปพลิเคชัน
- ระบบอัตโนมัติควบคุมเรือไปยังจุดหมายและกลับมาที่จุดเริ่มต้น

1.2) นำเรือกลับมาเมื่อการทำงานของเรือขัดข้อง

ฟังก์ชันนี้จะสามารถค้นหาเส้นทางกลับด้วยตนเอง โดยจะควบคุมเรือกลับมาที่จุดเริ่มต้นได้ เพื่อลดภาระหน้าที่ของผู้ใช้ที่จะต้องเดินทางไปเก็บเรือด้วยตนเอง โดยมีกรณีดังนี้

- พลังงานแบตเตอรี่ของเรือต่ำ
- เรือออกห่างจากสถานีฐานมากเกินไป
- การเชื่อมต่อกับเรือขัดข้อง

2) แอปพลิเคชันสำหรับควบคุมและติดตามสถานะของเรือ

แอปพลิเคชันสำหรับควบคุมและติดตามสถานะของเรือจะแบ่งออกเป็น 2 ส่วนคือ ส่วนควบคุมเรือและส่วนติดตามสถานะของเรือ โดยอธิบายได้ดังนี้

2.1) ส่วนสำหรับควบคุมเรือ

ส่วนนี้จะใช้สำหรับควบคุมเรือโดยการเลือกตำแหน่งเป้าหมายและสั่งให้เรือเดินทางไปยังตำแหน่งดังกล่าว หรือสั่งให้เรือกลับไปที่จุดเริ่มต้นขณะที่กำลังเดินทาง

2.2) ส่วนสำหรับติดตามสถานะของเรือ

สามารถติดตามสถานะต่าง ๆ ของเรือซึ่งดึงข้อมูลมาจากเซิร์ฟเวอร์ได้ ซึ่งข้อมูลที่จะนำมาแสดงบนแอปพลิเคชัน ได้แก่

- สถานะการทำงานปัจจุบันของเรือ
- ตำแหน่งปัจจุบันของเรือในรูปแบบละติจูดและลองจิจูด
- ระยะทางระหว่างเรือและเป้าหมายหน่วยเมตร
- ความเร็วการเคลื่อนที่หน่วยเมตรต่อวินาที (Meter Per Second)
- พลังงานที่เหลืออยู่ในหน่วยเปอร์เซ็นต์
- ความแรงของสัญญาณจากสถานีฐานหรืออาร์เอสเอสไอ (RSSI : Receive Strength Signal Indicator) ในหน่วยดีบีเอ็ม (dBm: Decibels-Milliwatt)

3) เซิร์ฟเวอร์สำหรับสร้างช่องทางสื่อสารระหว่างเรือและแอปพลิเคชัน

เนื่องจากเครือข่ายอินเทอร์เน็ตที่ไม่สามารถสร้างการเชื่อมต่อไปยังแอนดรอยด์แอปพลิเคชันโดยตรงได้ ผู้พัฒนาจึงได้พัฒนาเซิร์ฟเวอร์สำหรับเป็นตัวกลางสำหรับสื่อสารระบบทั้งสอง

1.4.3 ขอบเขตของข้อมูล

ข้อมูลที่เกี่ยวข้องกับการพัฒนามีดังนี้

- 1) ข้อมูลพิกัดละติจูดและลองจิจูดจากโมดูลจีพีเอส
- 2) ข้อมูลทิศทางของเรือจากโมดูลเข็มทิศ
- 3) ข้อมูลสถานะของเรือ
- 4) ข้อมูลแผนที่อิเล็กทรอนิกส์จากกูเกิลแมพ (Google Maps)

1.5 แผนการดำเนินงานและระยะเวลาดำเนินงาน

การศึกษานี้เริ่มดำเนินงานตั้งแต่เดือน สิงหาคม พ.ศ. 2561 สิ้นสุดเดือน พฤษภาคม พ.ศ. 2562 แสดงรายละเอียดการดำเนินงานดังตารางที่ 1.1 โดยมีขั้นตอนการดำเนินงานดังนี้

- 1) ศึกษาขอบเขตและปัญหา
- 2) ศึกษาทฤษฎีที่เกี่ยวข้อง
- 3) ออกแบบระบบ
- 4) พัฒนาระบบตามที่ได้ออกแบบไว้ พร้อมทั้งทำการทดสอบระบบไปพร้อม ๆ กัน
- 5) ทดสอบการใช้งานระบบ
- 6) จัดทำเอกสารประกอบระบบ พร้อมทั้งคู่มือการใช้งานระบบ

ซึ่งขั้นตอนการดำเนินงานทั้ง 6 ขั้นตอนนี้ ตามแผนการดำเนินงานจะถูกแบ่งเป็นช่วง ดังตารางที่ 1.1

ตารางที่ 1.1 ขั้นตอนการดำเนินงานและระยะเวลาการดำเนินงาน

ระยะเวลา ขั้นตอนการดำเนินงาน	พ.ศ.2561					พ.ศ.2562				
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.
1. ศึกษาขอบเขตและปัญหา										
2. ศึกษาทฤษฎีที่เกี่ยวข้อง										
3. ออกแบบระบบ										
4. พัฒนาระบบ										
5. ทดสอบและแก้ไขระบบงาน										
6. สรุปผลและจัดทำเอกสาร										

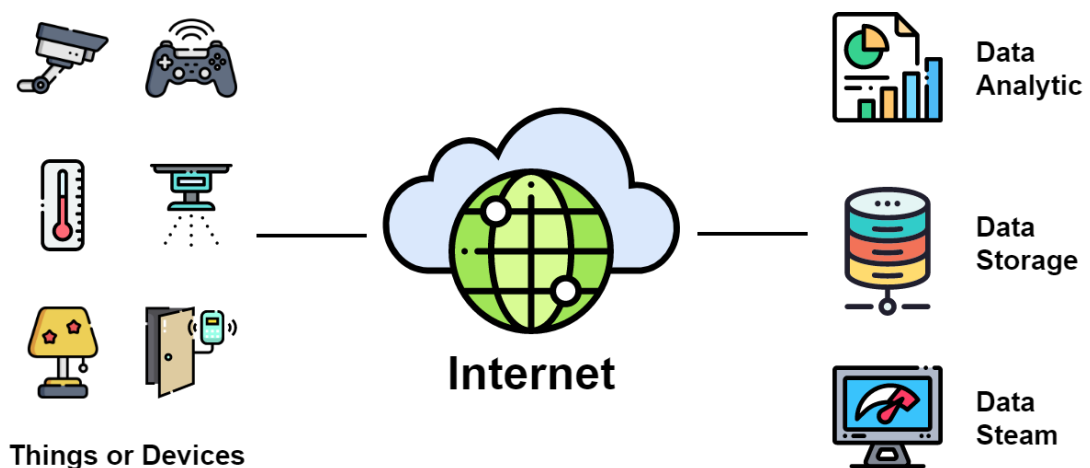
บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

จากการศึกษาขอบเขตงานและวัตถุประสงค์ในการพัฒนาแบบจำลองสำหรับระบบควบคุม เรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติ ได้มีการศึกษาองค์ความรู้จากหลักการและทฤษฎีซอฟต์แวร์ที่เหมาะสมต่อการพัฒนาระบบ รวมถึงการใช้งานอุปกรณ์อิเล็กทรอนิกส์ที่เกี่ยวข้อง เพื่อเป็นแนวความคิดในการออกแบบและพัฒนาโปรแกรม ซึ่งมีรายละเอียดดังต่อไปนี้

2.1 อินเทอร์เน็ตของสรรพสิ่ง

อินเทอร์เน็ตของสรรพสิ่งหรือไอโอที [1] คือการที่อุปกรณ์หรือเครื่องมือต่าง ๆ สามารถสื่อสารและเชื่อมต่อกันผ่านทางโพรโทคอล (Protocol) การสื่อสารได้ เช่น สัญญาณวิทยุ เอ็นบีไอโอที ไวไฟ (Wi-Fi) และบลูทูธ (Bluetooth) เป็นต้น ซึ่งมนุษย์ยังสามารถควบคุมอุปกรณ์เหล่านี้ผ่านทางเครือข่ายดังกล่าวได้ อุปกรณ์ไอโอทีที่เป็นลักษณะของเซนเซอร์ยังสามารถสร้างข้อมูลในปริมาณที่มหาศาลได้ ทำให้มีการนำข้อมูลเหล่านี้ไปทำการวิเคราะห์เพื่อนำไปใช้ประโยชน์ในด้านต่าง ๆ ยกตัวอย่างเช่น ข้อมูลอุณหภูมิและความชื้นจากเครื่องปรับอากาศขนาดใหญ่ที่ติดตั้งภายในห้างสรรพสินค้า ถูกนำไปทำวิเคราะห์ข้อมูล (Data Analytic) เพื่อหาแนวโน้มที่เครื่องจะชำรุดและนำไปวางแผนซ่อมบำรุงได้ล่วงหน้า ซึ่งจะช่วยลดค่าใช้จ่ายในการบำรุงรักษาลงไปได้ นอกจากนี้ยังมีการนำข้อมูลเหล่านี้ไปแสดงผลแบบทันที (Data Stream) เพื่อทำให้มนุษย์สามารถติดตามการทำงานของอุปกรณ์หรือเครื่องมือจากระยะไกลได้



ภาพที่ 2.1 ลักษณะของระบบไอโอที

คุณลักษณะของอุปกรณ์และเครือข่ายไอโอที คือ อุปกรณ์ไอโอทีต้องทำการเชื่อมต่อแบบไร้สายและต้องใช้เวลาทำงานต่ำ เพื่อให้สามารถใช้งานได้ในระยะเวลานาน สามารถใช้แบตเตอรี่เป็นแหล่ง

พลังงานอย่างเดียวก็สามารถใช้ได้เป็นระยะเวลานานหลายเดือน ซึ่งเหมาะกับลักษณะงานที่ต้องทำการวางแผนเซอร์หรืออุปกรณ์จำนวนมาก

ส่วนประกอบของอุปกรณ์ไอโอทีด้วยส่วนมากจะประกอบด้วยหน่วยประมวลผล หน่วยความจำ เซนเซอร์ วงจรรับส่งคลื่นวิทยุและแหล่งจ่ายพลังงาน หน่วยประมวลผลและหน่วยความจำจะทำหน้าที่รับข้อมูลที่ได้รับจากเซนเซอร์ อุปกรณ์ไอโอทีที่จะมีจำนวนมากในเครือข่ายหนึ่ง ๆ ราคาของอุปกรณ์ไอโอทีจึงต้องมีราคาถูกและเพื่อให้ระบบโดยรวมมีราคาไม่สูงมากนักการออกแบบเครือข่ายจึงต้องหลีกเลี่ยงการเลือกใช้อุปกรณ์ที่มีความซับซ้อนและมีราคาแพง โดยเลือกที่จะกำหนดให้การใช้การประมวลผลและหน่วยความจำน้อย นอกจากนี้ยังต้องพิจารณาถึง ค่าใช้จ่ายในการติดตั้งและดูแลระบบ โดยในบางลักษณะงานที่มีความต้องการเสถียรภาพของการส่งสัญญาณ อาจต้องทำการออกแบบในลักษณะที่มีความยืดหยุ่น เมื่ออุปกรณ์ถูกป้อนพลังงานเพื่อเริ่มเข้าร่วมเครือข่ายจากการเปิดเครื่องครั้งแรกและสามารถปรับเส้นทางการส่งข้อมูลจากปัญหาการเสียหายของอุปกรณ์บางส่วนในระบบที่เป็นอุปกรณ์ ส่งต่อข้อมูลภายในเครือข่าย

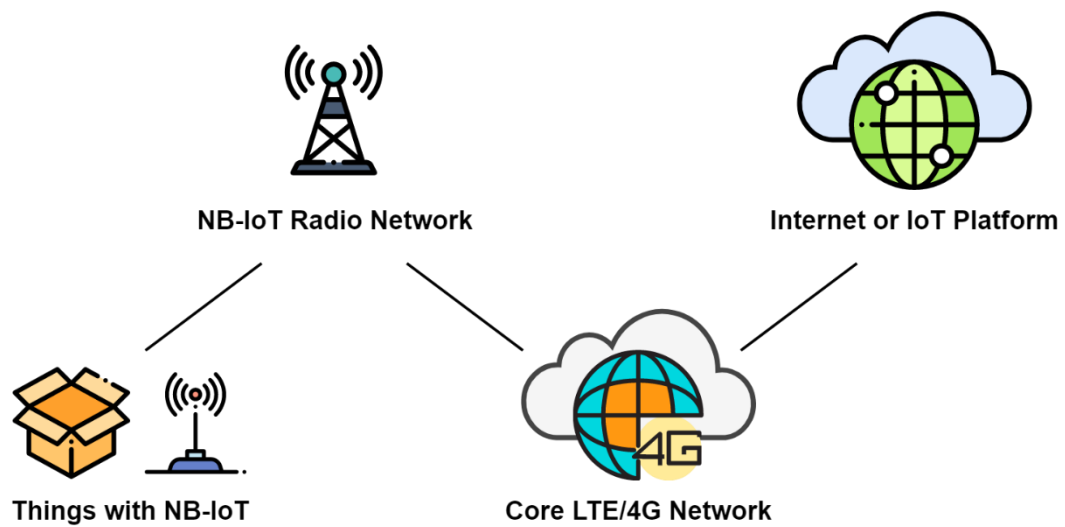
2.2 เอ็นบีไอโอที

เอ็นบีไอโอ [2] เป็นระบบเครือข่ายที่ถูกออกแบบโดย 3GPP ซึ่งเป็นผู้ดูแลมาตรฐานทางด้านการสื่อสารบนโครงข่ายโทรศัพท์เคลื่อนที่ มีลักษณะเป็นระบบเครือข่ายที่ใช้พลังงานต่ำ มีความเร็วในการสื่อสารและความถี่ในการส่งข้อมูลต่ำ อุปกรณ์เอ็นบีไอโอทีได้ทำงานบนคลื่นความถี่เดียวกันกับเครือข่ายของสัญญาณโทรศัพท์เคลื่อนที่

เครือข่ายของสัญญาณโทรศัพท์เคลื่อนที่ที่มีทั้งหมดสามย่านความถี่ ได้แก่

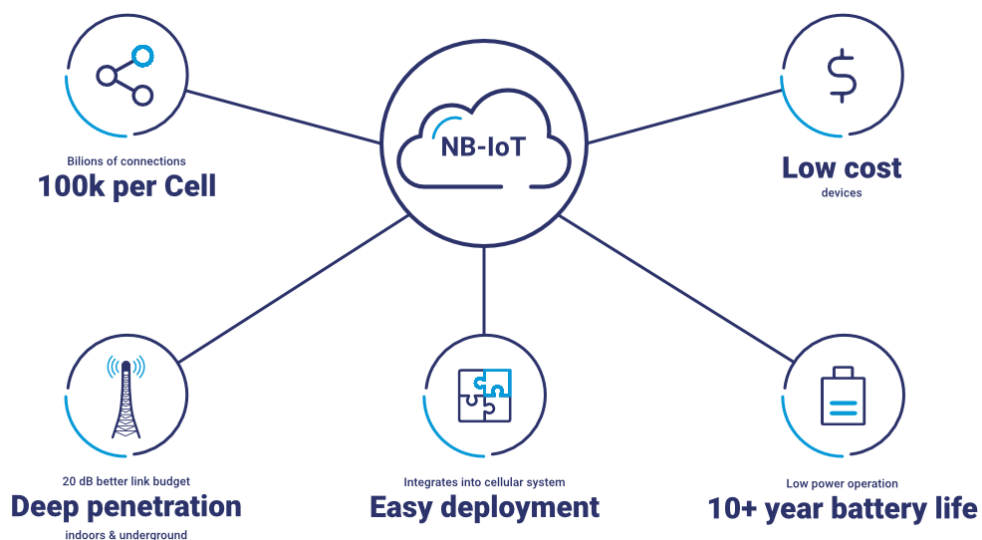
- จีเอสเอ็ม (GSM: Global System for Mobile Communications)
- 3 จี (3G: 3rd Generation Mobile Telecommunication)
- แอลทีอี (LTE: Long Term Evolution)

คลื่นทั้งสามอยู่ในย่านความถี่ที่หน่วยงานรัฐเป็นเจ้าของ (Licensed Band) ที่จะดำเนินการโดยผู้ให้บริการด้านเครือข่ายโทรคมนาคมที่ได้รับอนุญาตในการใช้คลื่นความถี่นั้น ๆ เอ็นบีไอโอทีใช้แถบความถี่อย่างน้อย 180 กิโลเฮิร์ตซ์ (kHz: Kilohertz) ซึ่งสามารถใช้อยู่บนคลื่นความถี่หนึ่งช่องของจีเอสเอ็ม ใช้อยู่บนแถบความถี่คู่ของ แอลทีอีหรือใช้อยู่บนคลื่นความถี่เดียวกันกับแอลทีอี อุปกรณ์เอ็นบีไอโอทีจะสามารถส่งข้อมูลผ่านโครงข่ายโทรศัพท์เคลื่อนที่ไปยังแอปพลิเคชันเซิร์ฟเวอร์ได้โดยตรง ดังนั้นผู้ให้บริการโครงข่ายโทรศัพท์เคลื่อนที่ที่เป็นผู้ดำเนินการสื่อสารข้อมูลกับอุปกรณ์เอ็นบีไอโอทีจึงไม่จำเป็นต้องมีอุปกรณ์อินเทอร์เน็ตเกตเวย์



ภาพที่ 2.2 ลักษณะของระบบเอ็นบีไอโอที

ประโยชน์ของเอ็นบีไอโอที คือ เป็นเทคโนโลยีที่ใช้พลังงานต่ำ สามารถเชื่อมต่ออุปกรณ์ได้เป็นจำนวนมาก มีพื้นที่ครอบคลุมเป็นวงกว้าง สัญญาณสามารถทะลุผ่านสิ่งกีดขวางได้ดี ต้นทุนอุปกรณ์ต่ำ ซึ่งสามารถแสดงได้ดังภาพที่ 2.3



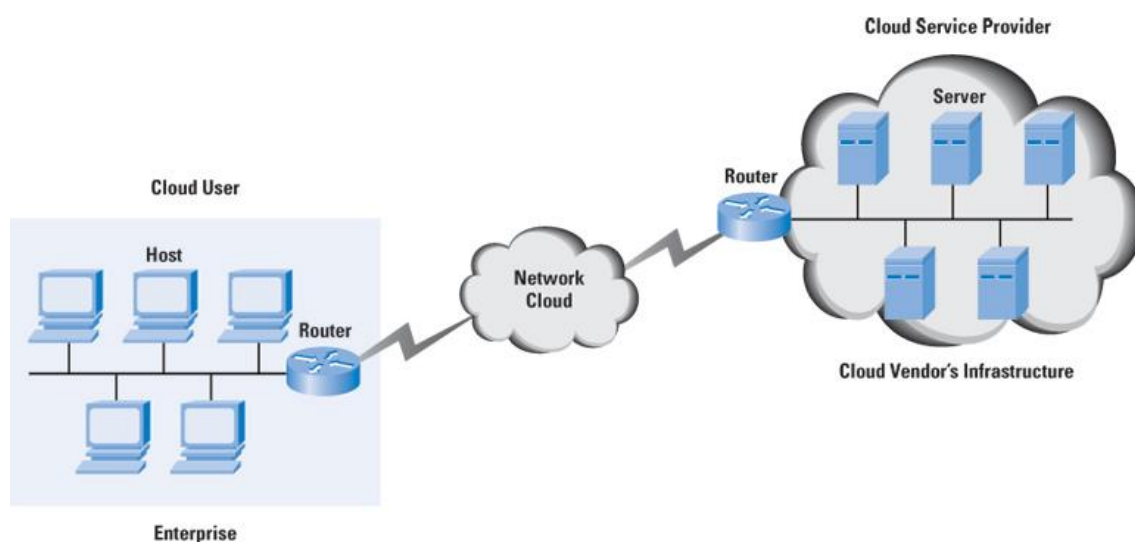
ภาพที่ 2.3 ประโยชน์ของเอ็นบีไอโอที

ที่มา <https://www.grandmetric.com/2018/02/16/narrowband-internet-of-things-nb-iot-an-overview/>

2.3 เครือข่ายคลาวด์

เครือข่ายคลาวด์ [3] เป็นการทำงานของเซิร์ฟเวอร์ขนาดใหญ่หรือเป็นการทำงานด้วยเซิร์ฟเวอร์หลาย ๆ เครื่อง ทำให้มีความสามารถในการประมวลผลแบบมหาศาล คลาวด์เซิร์ฟเวอร์จึงสามารถสร้างบริการ (Service) ขึ้นมาเพื่อทำงานได้หลากหลาย

ความแตกต่างระหว่างคลาวด์เซิร์ฟเวอร์และเครื่องเซิร์ฟเวอร์ทั่วไป คือ ภายในเครื่องเซิร์ฟเวอร์เมื่อมีการสร้างระบบปฏิบัติการ (Operating System) ขึ้นมา ระบบปฏิบัติการจะทำงานอยู่ภายในเครื่องเซิร์ฟเวอร์นั้น ๆ เมื่อเครื่องเซิร์ฟเวอร์มีปัญหา ระบบปฏิบัติการก็ไม่สามารถทำงานต่อได้ แต่สำหรับคลาวด์นั้น จะประกอบด้วยเครื่องเซิร์ฟเวอร์หลาย ๆ เครื่องช่วยกันทำงาน เช่น ถ้าต้องการสร้างบริการที่เป็นคอมพิวเตอร์เสมือน (Virtual Machine) แล้ว ระบบคลาวด์ก็จะทำการจำลองคอมพิวเตอร์เสมือนนั้น ๆ ขึ้นมา พร้อมกับจัดสรรทรัพยากรให้กับคอมพิวเตอร์เสมือนและด้วยระบบคลาวด์ที่ประกอบด้วยเครื่องเซิร์ฟเวอร์หลาย ๆ เครื่อง จึงทำให้เมื่อมีเซิร์ฟเวอร์ตัวใดตัวหนึ่งมีปัญหา ก็จะไม่มีผลต่อการทำงานของคอมพิวเตอร์เสมือนหรือบริการอื่น ๆ ที่กำลังทำงานอยู่ในขณะนั้น เพราะคลาวด์ จะมีการจัดสรรทรัพยากรเข้ามาทำงานแทนที่ได้ตลอดเวลา



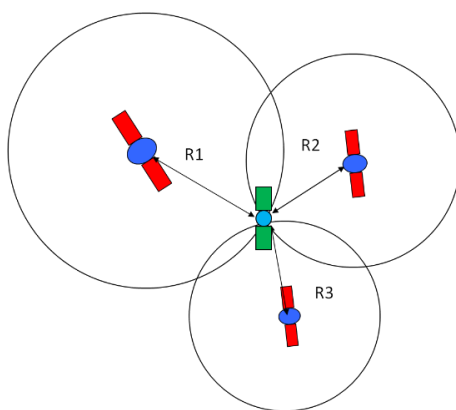
ภาพที่ 2.4 ลักษณะของเครือข่ายคลาวด์

ที่มา <https://www.cisco.com/c/en/us/about/press/internet-protocol-journal/back-issues/table-contents-45/123-cloud1.html>

2.4 ระบบกำหนดตำแหน่งบนโลก

ระบบกำหนดตำแหน่งบนโลกหรือจีพีเอส (GPS: Global Positioning System) [4] เป็นระบบที่ใช้ระบุตำแหน่งบนโลก สามารถระบุตำแหน่งพิกัด 3 แกนคือ (X, Y, Z) และระบุความเร็วของการเคลื่อนที่ ณ ตำแหน่งที่รับสัญญาณจากดาวเทียมนาฟสตาร์ (Navstar) ที่โคจรรอบโลก

เทคนิคการหาตำแหน่งของจีพีเอส คือ หากทราบตำแหน่งของดาวเทียมและรู้ระยะทางจากดาวเทียมถึงจุดรับสัญญาณ จะสามารถหาตำแหน่งของจุดรับสัญญาณได้ ซึ่งใช้ระยะทางแบบยูคลิดในการคำนวณ โดยใช้ดาวเทียม 3 ตัวในการหาตำแหน่งที่รัศมีตัดกันดังภาพที่ 2.5



ภาพที่ 2.5 การระบุตำแหน่งของจีพีเอส

ที่มา https://www.nasa.gov/directorates/heo/scan/communications/policy/GPS_History.html

2.5 ระยะทางยูคลิด

ระยะทางยูคลิด (Euclidean Distance) [5] คือระยะทางระหว่างจุดสองจุดในแนวเส้นตรง ระยะทางแบบยูคลิดระหว่างจุดสองจุด \mathbf{p} และ \mathbf{q} คือความยาวของส่วนของเส้นตรง $\overline{\mathbf{pq}}$ ถ้า $\mathbf{p} = (p_1, p_2, \dots, p_n)$ และ $\mathbf{q} = (q_1, q_2, \dots, q_n)$ เป็นจุดสองจุดบนปริภูมิยูคลิด n มิติ จะคำนวณระยะทางได้จากสมการที่ 2.1

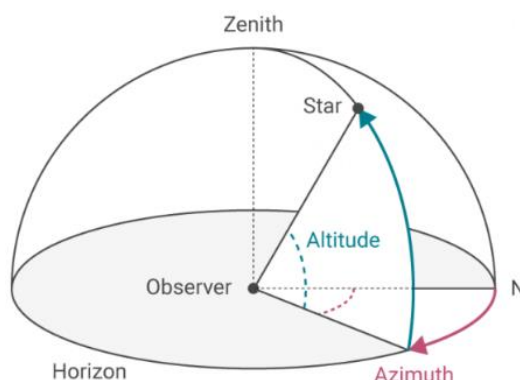
$$d(\mathbf{p}, \mathbf{q}) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \dots + (p_n - q_n)^2} \quad 2.1$$

ในระนาบสองมิติแบบยูคลิด ถ้า $\mathbf{p} = (p_1, p_2)$ และ $\mathbf{q} = (q_1, q_2)$ แล้ว ระยะทางระหว่าง \mathbf{p} และ \mathbf{q} สามารถคำนวณได้ดังสมการที่ 2.2

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2} \quad 2.2$$

2.6 ภาคติศ

ภาคติศหรือแอซิมัท (Azimuth) [6] เป็นการวัดมุมในพิกัดทรงกลม โดยลากเส้นจากจุดหนึ่งไปยังจุดที่สนใจและฉายลงบนระนาบที่อ้างอิง จะทำให้เกิดมุมระหว่างเส้นได้จากการฉายและเส้นที่อ้างอิงบนระนาบ ดังแสดงในภาพที่ 2.6



ภาพที่ 2.6 ส่วนบนของทรงกลมบนท้องฟ้า

ที่มา <https://www.timeanddate.com/astronomy/horizontal-coordinate-system.html>

ภาคติศที่บอกทิศทางให้กับเข็มทิศ จะใช้เส้นขนานหรือเส้นละติจูดเป็นระนาบอ้างอิง ซึ่งวัดจากแนวทิศเหนือวนตามเข็มนาฬิกา มีค่าตั้งแต่ 0-360 องศา สามารถแสดงได้ดังตารางที่ 2.1

ตารางที่ 2.1 ภาคติศที่วัดจากทิศเหนือวนตามเข็มนาฬิกา

ทิศเทียบกับองศา			
ทิศเหนือ	0°, 360°	ทิศใต้	180°
ทิศตะวันออกเฉียงเหนือ	45°	ทิศตะวันตกเฉียงใต้	225°
ทิศตะวันออก	90°	ทิศตะวันตก	270°
ทิศตะวันตกเฉียงใต้	135°	ทิศตะวันตกเฉียงเหนือ	315°

การระบุภาคติศบนแผนที่ (Cartographical Azimuth) จะสามารถระบุได้เมื่อทราบพิกัดของจุดสองจุดในแนวราบ (Cartographical Coordinates) ซึ่งสามารถคำนวณได้ดังสมการที่ 2.3

$$\alpha = \frac{180}{\pi} \text{atan2}(X_2 - X_1, Y_2 - Y_1) \quad 2.3$$

จากสมการที่ 2.3 ค่าที่ได้จากการคำนวณจะเป็นมุมในพิกัดวงกลม หากคำนวณออกมาแล้วเป็นค่าลบสามารถบวกด้วย 360 เพื่อแปลงเป็นค่าบวก

2.7 อาดูยโนเมกะ

อาดูยโนเมกะ (Arduino Mega) [7] เป็นบอร์ดไมโครคอนโทรลเลอร์ที่สามารถโปรแกรมการทำงานได้ นำวงจรอิเล็กทรอนิกส์หรือเซ็นเซอร์โมดูลจากภายนอกเชื่อมต่อเข้ามาที่ช่องอินพุตและเอาต์พุตของบอร์ดได้ นอกจากนี้ยังสามารถติดตั้งบอร์ดเสริม (Shield) ประเภทต่าง ๆ เช่น บอร์ดเสริมสำหรับเชื่อมต่อบลูทูธ บอร์ดเสริมสำหรับเชื่อมต่อไวไฟ หรือบอร์ดเสริมสำหรับขยายช่องจีพีไอโอ (GPIO: General-purpose input/output) เป็นต้น นอกจากนี้ทางผู้พัฒนาอาดูยโนยังได้เปิดเผยข้อมูลทั้งด้านฮาร์ดแวร์และซอฟต์แวร์ของตัวบอร์ดอาดูยโนรุ่นต่าง ๆ



ภาพที่ 2.7 Arduino Mega 2560 Rev3 [7]

บอร์ดอาดูยโนเมกะมีช่องสำหรับเชื่อมต่อกับอุปกรณ์ภายนอก มีดังนี้

- 1) ช่องจีพีไอโอ จำนวน 60 ช่อง
- 2) ช่องฮาร์ดแวร์ซีเรียล (Hardware Serial Ports) จำนวน 4 ช่อง
- 3) พอร์ตยูเอสบี (USB: Universal Serial Bus)
- 4) พอร์ตไอซีเอสพี (ICSP: In Circuit Serial Programming)

ตารางที่ 2.2 รายละเอียดของบอร์ดพัฒนาอาดูยโนเมกะ 2560

คุณสมบัติ	รายละเอียด
ไมโครคอนโทรลเลอร์	ATmega2560
แรงดันไฟฟ้าที่ทำงานได้	5 โวลต์
แรงดันไฟฟ้าอินพุตที่เหมาะสม	7 ~ 12 โวลต์
แรงดันไฟฟ้าอินพุตสูงสุด	6 ~ 20 โวลต์
ช่องดิจิทัลอินพุตและเอาต์พุต	54 ช่อง (มี 15 ช่องที่สามารถใช้สัญญาณ PWM ได้)
ช่องแอนะล็อกอินพุต	16 ช่อง

ตารางที่ 2.2 รายละเอียดของบอร์ดพัฒนาอาดุยโนเมกะ 2560 (ต่อ)

คุณสมบัติ	รายละเอียด
หน่วยความจำแฟลช (Flash Memory)	16 กิโลไบต์ (Kilobyte)
เอสแรม (SRAM)	8 กิโลไบต์
อีอีพรอม (EEPROM)	4 กิโลไบต์
ความเร็วสัญญาณนาฬิกา	16 เมกะเฮิร์ตซ์ (Megahertz)

2.8 บอร์ดเสริมเอไอเอสเอ็นบีไอโอที

บอร์ดเสริมเอไอเอสเอ็นบีไอโอที (AIS NB-IoT Shield) หรือ เดวิโอเอ็นบีชีลด์วัน (Devio NB-Shield I) [8] เป็นบอร์ดสื่อสารสำหรับเครือข่ายเอ็นบีไอโอทีของเอไอเอส ใช้สำหรับเชื่อมต่อกับเครือข่ายโทรศัพท์เคลื่อนที่เพื่อเข้าถึงเครือข่ายอินเทอร์เน็ต บอร์ดเสริมนี้สามารถติดตั้งลงบนบอร์ดอาดุยโนเมกะ โดยจะการใช้การสื่อสารแบบยูเออาร์ที (UART: Universal Asynchronous Receiver-Transmitter) เพื่อรับหรือส่งข้อมูล



ภาพที่ 2.8 Devio NB-Shield I

ที่มา <http://www.bbestit.com/ลองเล่นบอร์ด-devio-nb-shield-i-ทางเริ่ม>

ส่วนประกอบหลักของบอร์ดเสริมเอไอเอสเอ็นบีไอโอที มีดังนี้

- 1) ชิป Quectel BC95 ซึ่งเป็นโมดูลเอ็นบีไอโอที
- 2) อีซิม (eSIM) สำหรับเข้าถึงเครือข่ายโทรศัพท์เคลื่อนที่
- 3) เสาอากาศสำหรับรับสัญญาณ

ตารางที่ 2.3 รายละเอียดของบอร์ดเสริมเอ็นบีไอโอที

คุณสมบัติ	รายละเอียด
โมดูลการสื่อสาร	Quectel BC95
เทคโนโลยีเครือข่าย	LTE Cat. NB1 หรือ เอ็นบีไอโอที
ความถี่	แบนด์ 8 หรือ 900 เมกะเฮิร์ตซ์
การส่งข้อมูลดาวน์โหลด (Downlink)	24 กิโลบิตต่อวินาที (Kbps : Kilobit Per Second)
การส่งข้อมูลอัปลิงค์ (Uplink)	15.625 กิโลบิตต่อวินาที
โพรโตคอลสแตค (Protocol Stack)	- ยูดีพี (UDP: User Datagram Protocol) - โคเอพี (CoAP: Constraint Application Protocol)

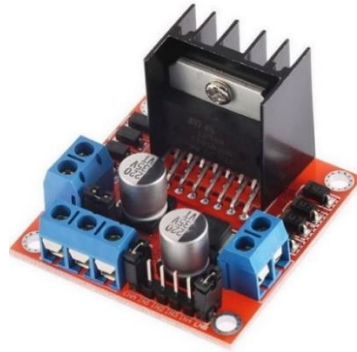
บอร์ดเสริมเอไอเอสเอ็นบีไอโอที จะมีคำสั่งสำหรับใช้งานบอร์ดหรือคำสั่งเอที (AT Command) ซึ่งอ้างอิงจาก Quectel BC95 [9] มีตัวอย่างคำสั่งที่สามารถใช้งานได้ดังตารางที่ 2.4

ตารางที่ 2.4 ตัวอย่างคำสั่งสำหรับใช้งานบอร์ดเสริมเอไอเอสเอ็นบีไอโอที

คำสั่ง	ผลลัพธ์
AT+NPING	ปิง (Ping) ไปยังไอพีที่ระบุ
AT+NSORC	สร้างซ็อกเก็ต
AT+NSOST	รับและส่งแพ็กเก็ตยูดีพี ขนาดไม่เกิน 512 ไบต์ (Btye)
AT+NSORF	รับแพ็กเก็ตยูดีพี ขนาดไม่เกิน 512 ไบต์
AT+CGMR	อ่านเวอร์ชันของเฟิร์มแวร์
AT+CGSN	อ่านหมายเลขประจำโมดูล (IMEI)
AT+CIMI	อ่านหมายเลขประจำซิม (IMSI)
AT+CGPADDR	อ่านไอพีที่ได้รับจากเครือข่าย
AT+CSQ	อ่านความแรงสัญญาณ

2.9 โมดูลขับเคลื่อนมอเตอร์ L298N

โมดูลขับเคลื่อนมอเตอร์ L298N (Motor Drive Module L298N) [10] เป็นโมดูลที่สามารถควบคุมความเร็วการหมุนและทิศทางการหมุนของมอเตอร์กระแสตรงได้ โมดูลนี้สามารถควบคุมการทำงานของมอเตอร์กระแสตรงได้สองตัวแบบแยกอิสระจากกัน



ภาพที่ 2.8 โมดูลขับมอเตอร์ L298N

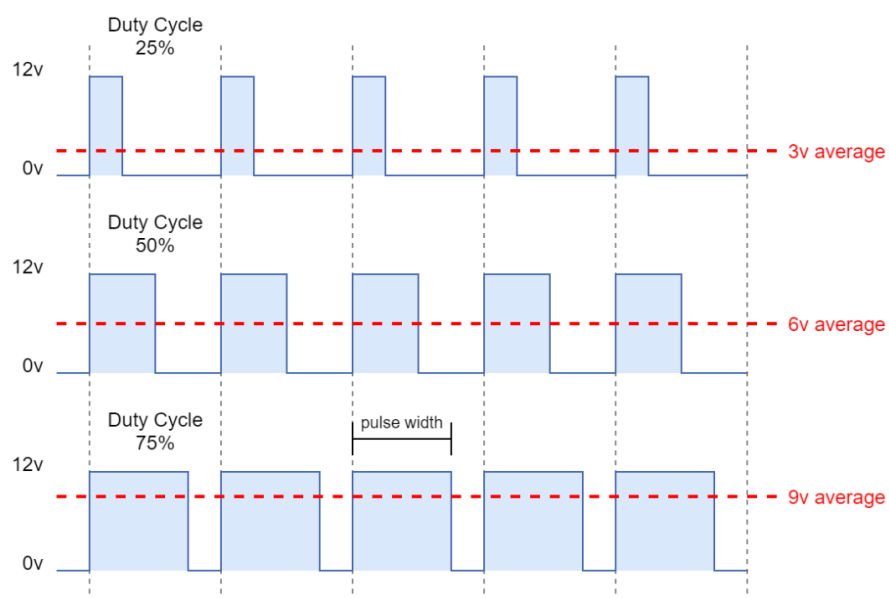
ที่มา <https://makeradvisor.com/tools/l298n-motor-driver>

ตารางที่ 2.5 รายละเอียดของโมดูลขับมอเตอร์ L298N

คุณสมบัติ	รายละเอียด
ชิปขับเคลื่อนเอชบริดจ์คู่	L298N
แรงดันไฟฟ้าที่ทำงานได้	5 ~ 35 โวลต์
กระแสขับเคลื่อนสูงสุด	2 แอมป์

1) การปรับความเร็วการหมุนของมอเตอร์กระแสตรงด้วยเทคนิคพีดับเบิลยูเอ็ม

ความเร็วของมอเตอร์กระแสตรงสามารถควบคุมได้โดยการเปลี่ยนแปลงแรงดันไฟฟ้าอินพุต โดยใช้เทคนิคพีดับเบิลยูเอ็ม (PWM: Pulse Width Modulation) ซึ่งเป็นเทคนิคที่ปรับค่าเฉลี่ยของแรงดันไฟฟ้าอินพุตโดยส่งชุดสัญญาณพัลส์เปิดกับปิด แรงดันไฟฟ้าเฉลี่ยจะแปรผันตามความกว้างของสัญญาณ (Pulse Width) หรือเรียกว่าดีวตี้ไซเคิล (Duty Cycle) ดังภาพที่ 2.9

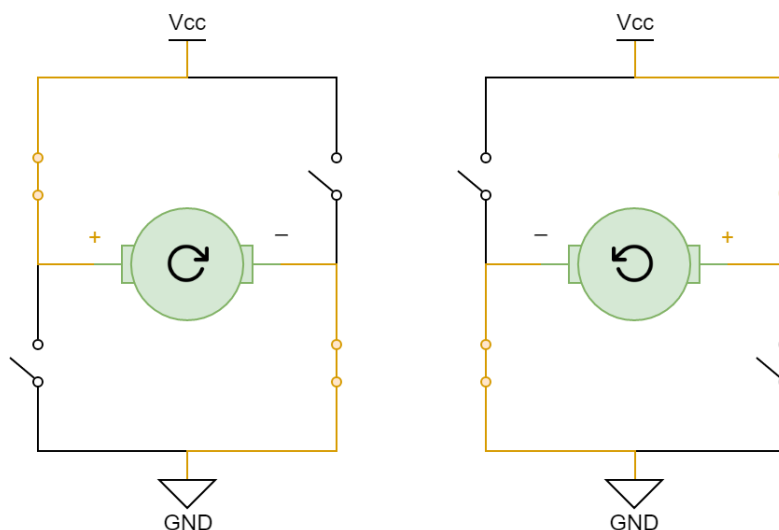


ภาพที่ 2.9 กราฟแสดงแรงดันไฟฟ้าเฉลี่ยจากดีวตี้ไซเคิล

การควบคุมความเร็วการหมุนของมอเตอร์กระแสตรงด้วยโมดูลขับมอเตอร์ L298N สามารถปรับค่าได้ระหว่าง 0 ถึง 255 ซึ่งหมายถึงอัตราส่วนของปริมาณกระแสไฟที่ต้องการจ่ายเข้าไปยังมอเตอร์แต่ละตัว

2) การควบคุมทิศทางการหมุนของมอเตอร์ด้วยวงจรเอชบริดจ์

ทิศทางการหมุนของมอเตอร์กระแสตรงสามารถควบคุมได้โดยการเปลี่ยนขั้วของแรงดันไฟฟ้า โดยใช้วงจรเอชบริดจ์ (H-Bridge Circuit) ซึ่งประกอบด้วยสวิตช์สี่ตัวโดยมีมอเตอร์อยู่ตรงกลางและเชื่อมต่อวงจรเป็นรูปตัวเอช ดังภาพที่ 2.10



ภาพที่ 2.10 การทำงานของวงจรเอชบริดจ์เพื่อควบคุมทิศทางการหมุนของมอเตอร์กระแสตรง

การควบคุมทิศทางการหมุนของโมดูลขับมอเตอร์ L298N สำหรับมอเตอร์ตัวแรกจะใช้ช่อง Input 1 กับ Input 2 ในการควบคุม และใช้ช่อง Input 3 และ Input 4 ในการควบคุมมอเตอร์ตัวที่สอง สามารถแสดงทิศทางการหมุนและหยุดหมุนได้ดังตารางที่ 2.6

ตารางที่ 2.6 การควบคุมทิศทางการหมุนของโมดูลขับมอเตอร์ L298N

Input 1 & 3	Input 2 & 4	Spinning Direction
LOW (0)	LOW (0)	Stop
LOW (0)	HIGH (1)	Backward
HIGH (1)	LOW (0)	Forward
HIGH (1)	HIGH (1)	Stop

2.10 โมดูลจีพีเอส Ublox NEO-M8N

โมดูลจีพีเอส Ublox NEO-M8N (GPS Module Ublox NEO-M8N) [11] เป็นโมดูลที่สามารถติดตามระบบนำทางจีพีเอสและจีเอ็นเอ็มอีเอส (GNSS: Global Navigation Satellite System) ได้โดยใช้เสาอากาศ (Antenna) เป็นตัวรับสัญญาณจากอวกาศ ข้อมูลที่ถูกส่งออกมาจากโมดูลจะอยู่ในลักษณะโพรโตคอลเอ็นเอ็มอีเอ (NMEA: National Marine Electronics Association) การอ่านข้อมูลจากโมดูลมายังบอร์ดพัฒนาจะต้องใช้การสื่อสารแบบยูเออาร์ที

โมดูลจีพีเอส Ublox NEO-M8N มีไลบรารีที่รองรับการทำงานของโมดูลนี้ คือ ไลบรารีไทนี่จีพีเอสพลัส (TinyGPSPlus Library) มีความสามารถในการแปลงข้อมูลจากโมดูลที่อยู่ในรูปแบบเอ็นเอ็มอีเอ เป็นค่าต่าง ๆ เช่น ละติจูดและลองจิจูดของตำแหน่งที่รับสัญญาณหรือเวลาสากลเชิงพิกัด (Coordinated Universal Time) เป็นต้น



ภาพที่ 2.11 โมดูลจีพีเอส Ublox NEO-M8N และเสาอากาศ

ที่มา <https://www.arduinoall.com/product/1113>

ตารางที่ 2.7 รายละเอียดของโมดูลจีพีเอส Ublox NEO-M8N

คุณสมบัติ	รายละเอียด
ชิปจีพีเอส	Ublox NEO-M8N
แรงดันไฟฟ้าที่ทำงานได้	3.3 ~ 5 โวลต์
อุณหภูมิที่ทำงานได้	-40 ~ 85 องศาเซลเซียส (°C)
ความแม่นยำของการระบุตำแหน่ง	±2.5 เมตร
โพรโตคอลการสื่อสาร	<ul style="list-style-type: none">- เอ็นเอ็มอีเอ- อาร์ทีซีเอ็ม (RTCM: Radio Technical Commission for Maritime Services)- ยูบีเอ็กซ์ (UBX: Ublox)

2.11 โมดูลเข็มทิศดิจิทัลสามแกน DA5883 GY-273

โมดูลเข็มทิศดิจิทัลสามแกน DA5883 GY-273 (3-Axis Digital Compass Module DA5883 GY-273) เป็นโมดูลเข็มทิศอิเล็กทรอนิกส์ ประกอบด้วยเซนเซอร์ HMC5883L [12] ที่ใช้สำหรับตรวจจับสนามแม่เหล็กโลกได้ ข้อมูลที่ได้จากโมดูลนี้จะอยู่ในรูปแบบพิกัดสามมิติได้แก่ X, Y และ Z ซึ่งสามารถนำไปคำนวณหามุมแอสิมัทได้



ภาพที่ 2.12 โมดูลเข็มทิศ GY-273 HMC5883L

ที่มา <https://www.amazon.in/GY-273-HMC5883L-Compass-Magnetometer-Arduino/dp/B00NU93CGC>

ตารางที่ 2.8 โมดูลเข็มทิศดิจิทัลสามแกน DA5883 GY-273

คุณสมบัติ	รายละเอียด
ชิปเข็มทิศ	HMC5883L
รุ่นของโมดูล	GY-273
แรงดันไฟฟ้าที่ทำงานได้	3.3 ~ 5 โวลต์
ช่วงการวัด	$\pm 1.3 \sim 8$ เกาส์เซียน (Gaussian)
โปรโตคอลการสื่อสาร	ไอส์แควร์ซี (I ² C: Inter-Integrated Circuit)

2.12 โวลเตจเซนเซอร์

โวลเตจเซนเซอร์ [13] เป็นเซนเซอร์ที่ใช้สำหรับวัดแรงดันไฟฟ้ากระแสตรง สามารถนำโมดูลนี้ตัวร่วมเข้ากับวงจรเพื่อวัดแรงดันไฟฟ้าได้



ภาพที่ 2.13 โวลเตจเซนเซอร์

ที่มา <https://www.arduinoall.com/product/598>

บทที่ 3

การวิเคราะห์และออกแบบระบบ

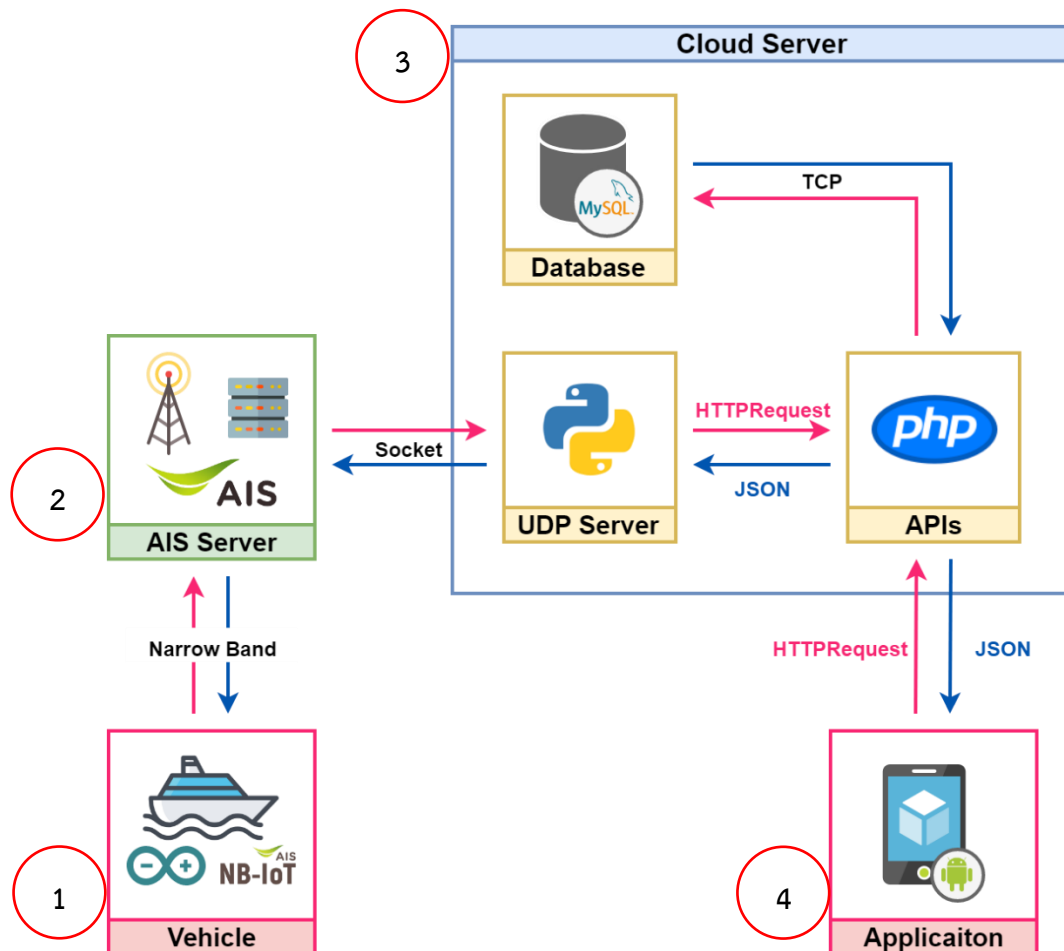
การวิเคราะห์และออกแบบระบบเป็นขั้นตอนที่มีความสำคัญอย่างมากไม่ว่าจะเป็นการพัฒนา
ระบบใหม่ หรือพัฒนาเพื่อปรับปรุงระบบงานเดิม เนื่องจากเป็นขั้นตอนที่ทำให้เห็นถึงโครงสร้างการ
ทำงานของระบบและการติดต่อประสานกับบุคคลที่เกี่ยวข้องกับระบบ การวิเคราะห์ระบบเริ่มจาก
การศึกษาจากระบบเดิม หรือศึกษาความต้องการที่จะทำระบบใหม่ ทำให้ทราบถึงปัญหาที่เกิดขึ้นและ
สามารถจัดการแก้ไขกับปัญหานั้นได้การออกแบบ เป็นการนำเอาความรู้หรือความต้องการมาเป็น
แบบแผนในการสร้าง เพื่อให้ระบบสามารถทำงานได้จริง ขั้นตอนการวิเคราะห์และออกแบบระบบจะ
ช่วยให้ลดความซับซ้อนของการทำงาน จัดการกับการทำงานที่ซ้ำซ้อนให้เกิดความถูกต้องแม่นยำ

3.1 ปัญหาและความต้องการของระบบ

จากระบบเดิมที่ผู้ใช้จะต้องใช้เรือเพื่อไปยังจุดที่ต้องการเก็บตัวอย่างน้ำ ทำให้มักจะ
เสียเวลาและเสี่ยงต่อการเกิดอุบัติเหตุได้ อีกทั้งการเก็บตัวอย่างน้ำจะทำได้ค่อนข้างยากหากน้ำใน
แหล่งน้ำมีความปนเปื้อนสูง จากการวิเคราะห์ปัญหาทำให้ได้ความต้องการของระบบที่สามารถแก้ไข
ปัญหาที่กล่าวมาข้างต้น จึงได้พัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและ
ทิศทางแบบอัตโนมัติ ซึ่งเป็นยานพาหนะที่สามารถติดตั้งอุปกรณ์สำหรับการเก็บตัวอย่างน้ำหรือ
วิเคราะห์คุณภาพน้ำและเดินทางไปยังจุดที่ต้องการด้วยระบบอัตโนมัติได้ ซึ่งมีหน้าที่หลักดังนี้

- 1) เรือต้องสามารถไปยังจุดหมายและกลับมายังจุดเริ่มต้นได้
- 2) สามารถควบคุมเรือและติดตามสถานะการทำงานของเรือผ่านแอปพลิเคชันได้
- 3) มีระบบกู้เรือที่สามารถตรวจพบปัญหาการทำงานเรือได้ เช่น พลังงานแบตเตอรี่ต่ำ หรือ
สัญญาณอินเทอร์เน็ตขาดข้อง เป็นต้น เพื่อนำพาเรือกลับมาที่จุดเริ่มต้นได้ทันที
- 4) ไม่ครอบคลุมถึงการพัฒนาเซนเซอร์สำหรับตรวจจับสิ่งกีดขวางในระหว่างที่เรือกำลัง
เดินทางและอุปกรณ์สำหรับเก็บตัวอย่างน้ำ
- 5) สภาพแวดล้อมที่เรือจะถูกใช้งานมีลักษณะเป็นบ่อน้ำหรืออ่างเก็บน้ำ

3.2 ลักษณะการทำงานของระบบ



ภาพที่ 3.1 ลักษณะการทำงานของระบบ

จากภาพที่ 3.1 สามารถอธิบายลักษณะการทำงานของระบบได้ดังนี้

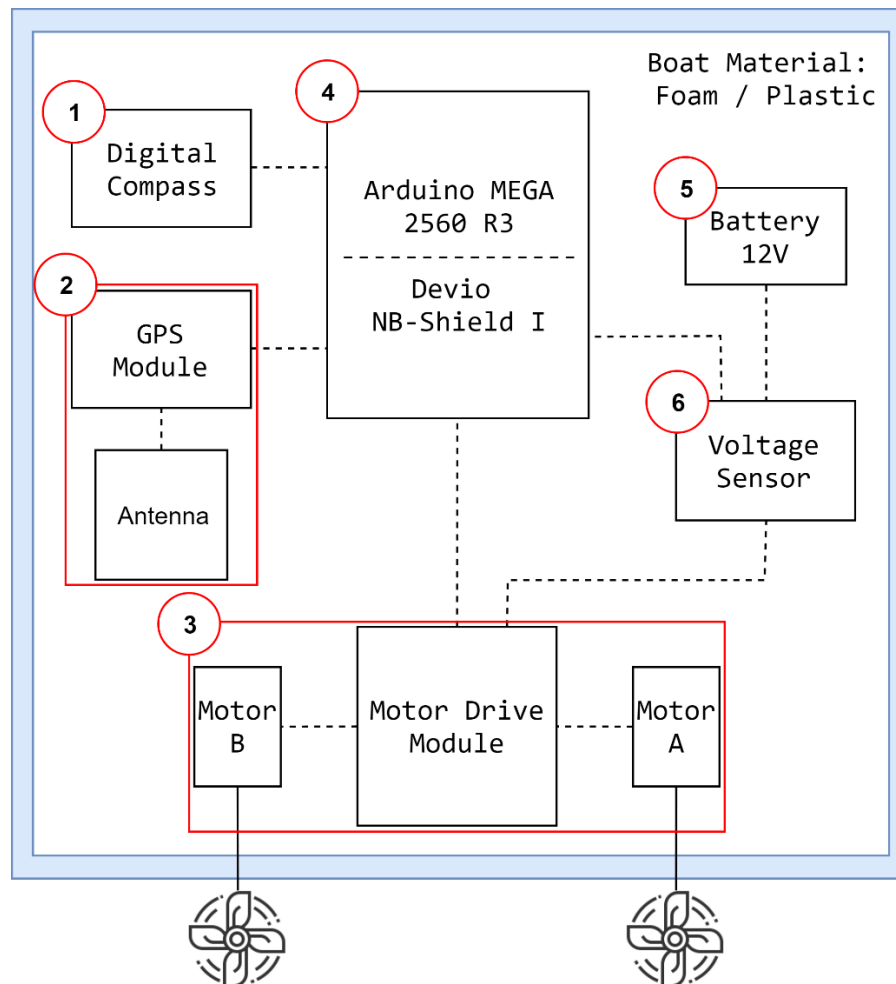
ส่วนที่ 1 คือ ยานพาหนะที่มีลักษณะเป็นเรือ มีอุปกรณ์สำหรับควบคุมการทำงานต่าง ๆ ภายในเรือ ซึ่งจะอธิบายรายละเอียดการใช้งานในหัวข้อที่ 3.3

ส่วนที่ 2 คือ เครือข่ายของเอไอเอสที่รับสัญญาณเอ็นบีไอโอที และรับส่งข้อมูลไปยังเซิร์ฟเวอร์ในส่วนที่ 3

ส่วนที่ 3 คือ คลาวด์เซิร์ฟเวอร์ เป็นช่องทางการติดต่อสื่อสารระหว่างเรือและแอปพลิเคชัน

ส่วนที่ 4 คือ แอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ ใช้สำหรับควบคุมและติดตามสถานะการทำงานของเรือ

3.3 การออกแบบสถาปัตยกรรมของเรือ



ภาพที่ 3.2 อุปกรณ์ภายในเรือขับเคลื่อนอัตโนมัติ

จากภาพที่ 3.2 อุปกรณ์ที่ใช้ภายในเรือมีดังนี้

ส่วนที่ 1 โมดูลจีพีเอส ใช้สำหรับระบุตำแหน่งของเรือบนแผนที่

ส่วนที่ 2 เซ็นเซอร์ดิจิตอลและเสาอากาศ ใช้สำหรับค้นหาทิศทางที่เรือหันอยู่

ส่วนที่ 3 โมดูลขับเคลื่อนมอเตอร์ ใช้สำหรับควบคุมการทำงานของมอเตอร์ทั้งสองลูก

ส่วนที่ 4 บอร์ดอาduinoเมกะ ใช้ควบคุมการทำงานของอุปกรณ์อิเล็กทรอนิกส์ทั้งหมดบนเรือและ
บอร์ดเสริมเอไอเอสเอ็นพีไอโอที ใช้สำหรับส่งข้อมูลเข้าสู่เครือข่ายอินเทอร์เน็ต

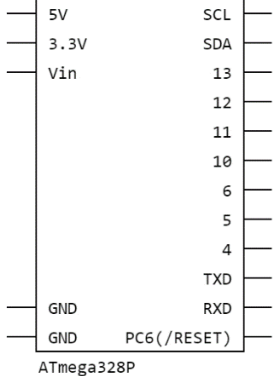


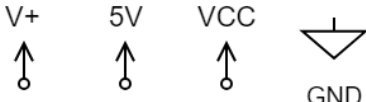


ส่วนที่ 5 แหล่งพลังงาน ใช้สำหรับจ่ายพลังงานให้กับเรือ

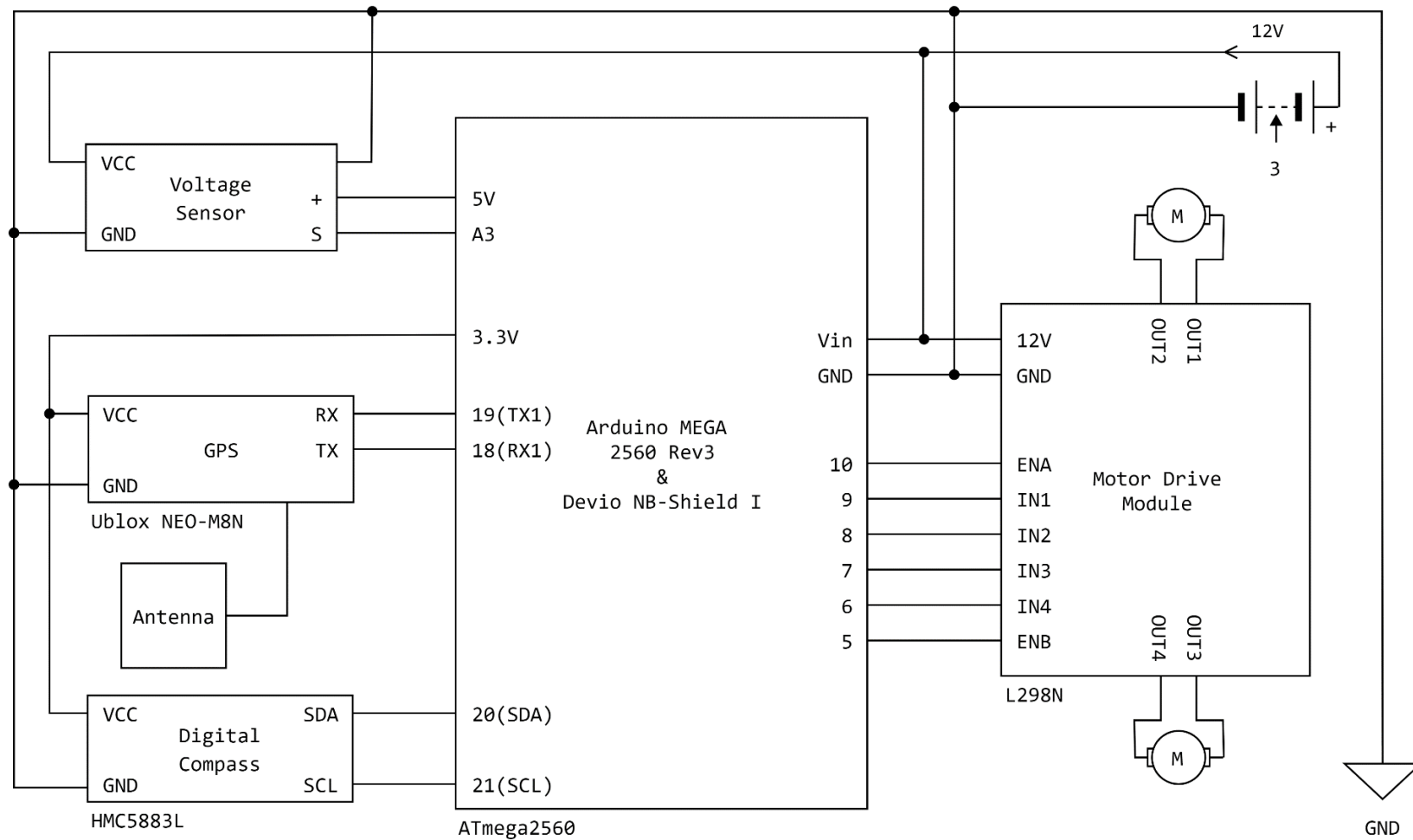
ส่วนที่ 6 โวลเตจเซนเซอร์ ใช้วัดแรงดันไฟฟ้าของแหล่งพลังงาน

3.4 แผนผังวงจรอิเล็กทรอนิกส์

แผนผังวงจรอิเล็กทรอนิกส์ (Schematic) คือแผนผังที่ใช้สำหรับแสดงการเชื่อมต่อของอุปกรณ์อิเล็กทรอนิกส์ภายในวงจร แผนผังวงจรอิเล็กทรอนิกส์ของเรือขับเคลื่อนอัตโนมัติสามารถแสดงในภาพที่ 3.3 ซึ่งสัญลักษณ์ที่ใช้แสดงแผนผังวงจรอิเล็กทรอนิกส์ดังตารางที่ 3.1 ดังนี้

ตารางที่ 3.1 สัญลักษณ์และความหมายของแผนผังวงจรอิเล็กทรอนิกส์

ชื่อสัญลักษณ์	คำอธิบาย	ตัวอย่างสัญลักษณ์
Integrated Circuits / Microcontrollers	วงจรประกอบด้วยขา (pins) ที่ถูกกำกับด้วยชื่อหรือตัวเลข	
Wires	สายไฟสำหรับนำไฟฟ้า	
Junctions and Nodes	การแบ่งทางแยกให้กับสายไฟ	
Voltage Nodes	โหนดที่เกี่ยวข้องกับแรงดันไฟฟ้า	
Batteries	เซลล์แบตเตอรี่	
Motor	มอเตอร์	



ภาพที่ 3.3 แผนผังวงจรการเชื่อมต่อของอุปกรณ์อิเล็กทรอนิกส์ภายในเรือ

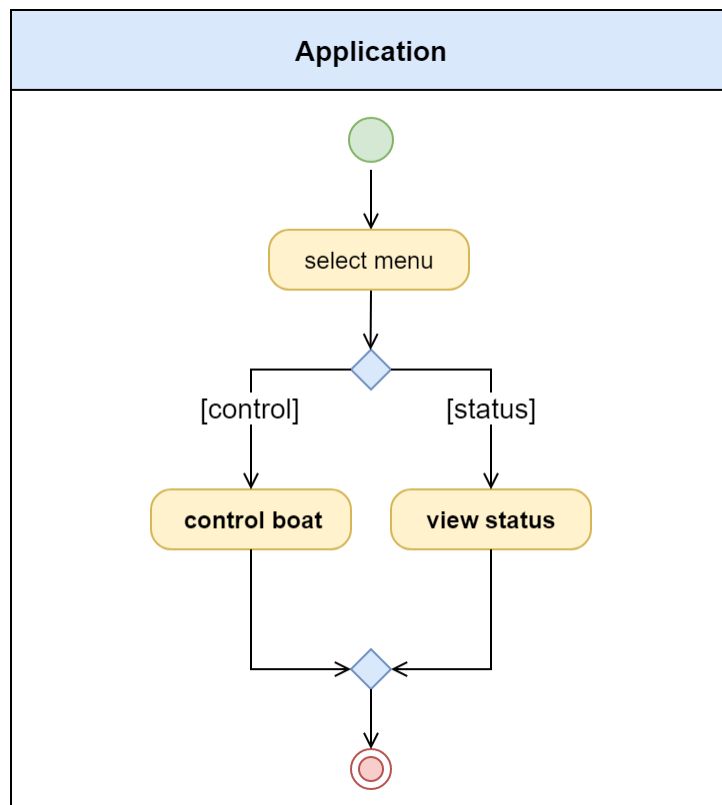
3.5 แผนภาพกิจกรรม

แผนภาพกิจกรรม (Activity Diagram) เป็นแผนภาพที่ใช้อธิบายขั้นตอนการทำงานของระบบในลักษณะกระแสการไหลของงาน (Work Flow) แผนภาพกิจกรรมของเรือขับเคลื่อนอัตโนมัติสามารถแสดงในภาพที่ 3.4 - ภาพที่ 3.7 และสัญลักษณ์ที่ใช้แสดงแผนภาพกิจกรรมดังตารางที่ 3.2 ดังนี้

ตารางที่ 3.2 สัญลักษณ์และความหมายของแผนภาพกิจกรรม

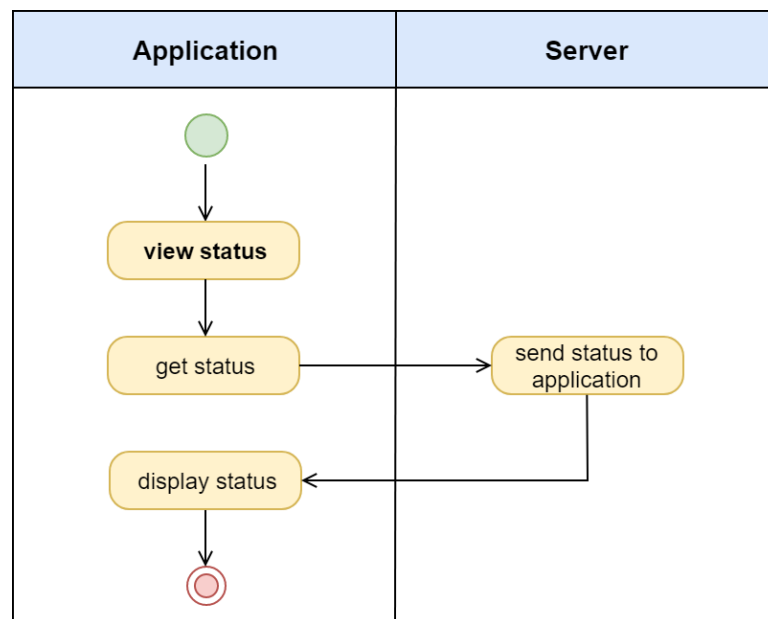
ชื่อสัญลักษณ์	คำอธิบาย	ตัวอย่างสัญลักษณ์
Initial State	จุดเริ่มต้นของระบบ	
Final State	จุดสิ้นสุดของระบบ	
Activity	งานหรือกิจกรรมที่เกิดขึ้น	
Control Flow	ทิศทางการไหลของงาน	
Swimlanes	การจัดกลุ่มของกิจกรรมที่มีความเกี่ยวข้องกันไว้ด้วยกัน	
Fork / Merge	แยกเงื่อนไขหรือรวมการไหลของงาน	
Time Event	กิจกรรมที่ต้องใช้เวลาหรือรอให้เหตุการณ์เกิดขึ้นก่อน	

3.5.1 แผนภาพกิจกรรมของแอปพลิเคชัน



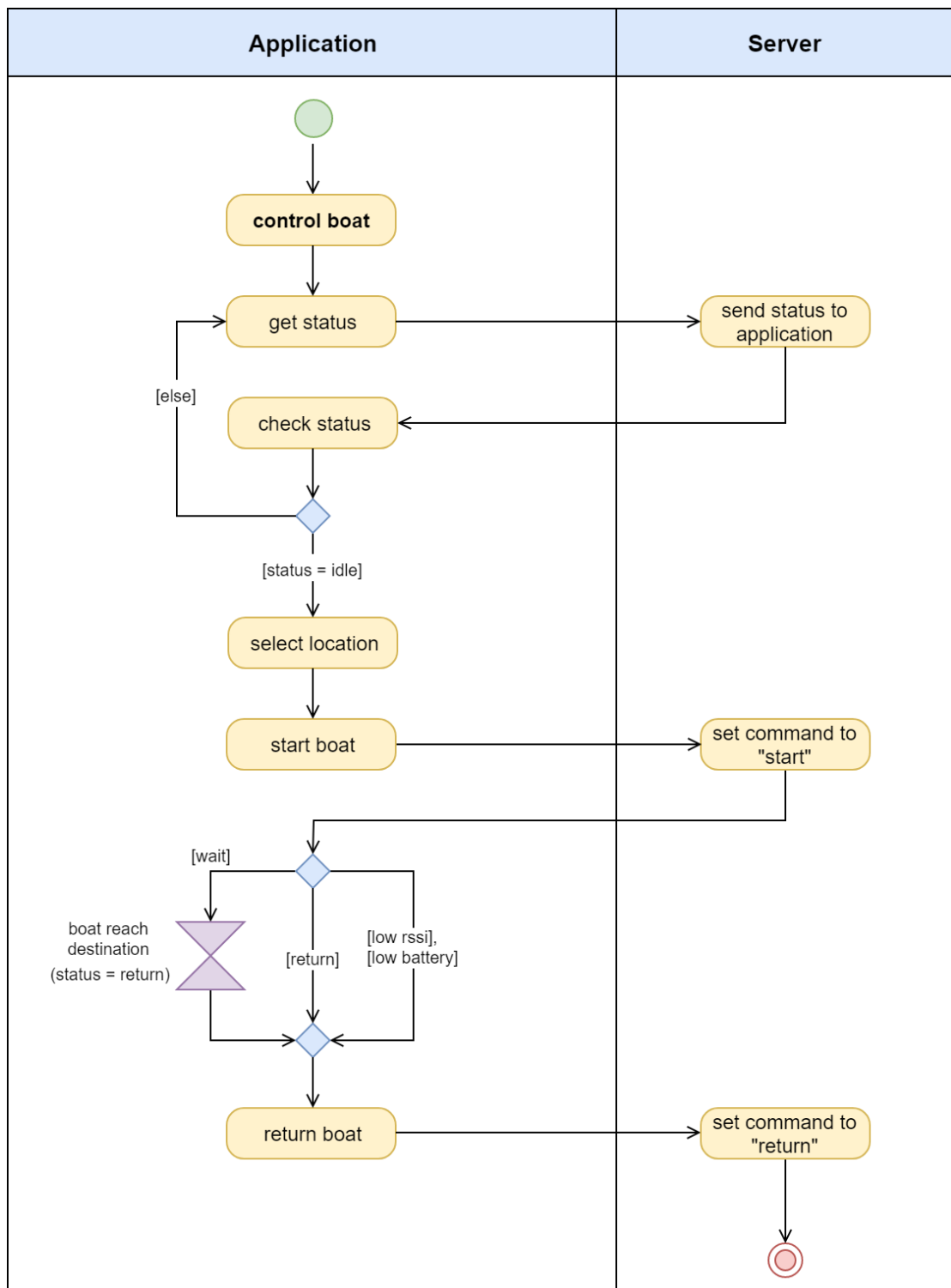
ภาพที่ 3.4 แผนภาพกิจกรรมของแอปพลิเคชัน

3.5.2 แผนภาพกิจกรรมของแอปพลิเคชันส่วนติดตามสถานะของเรือ



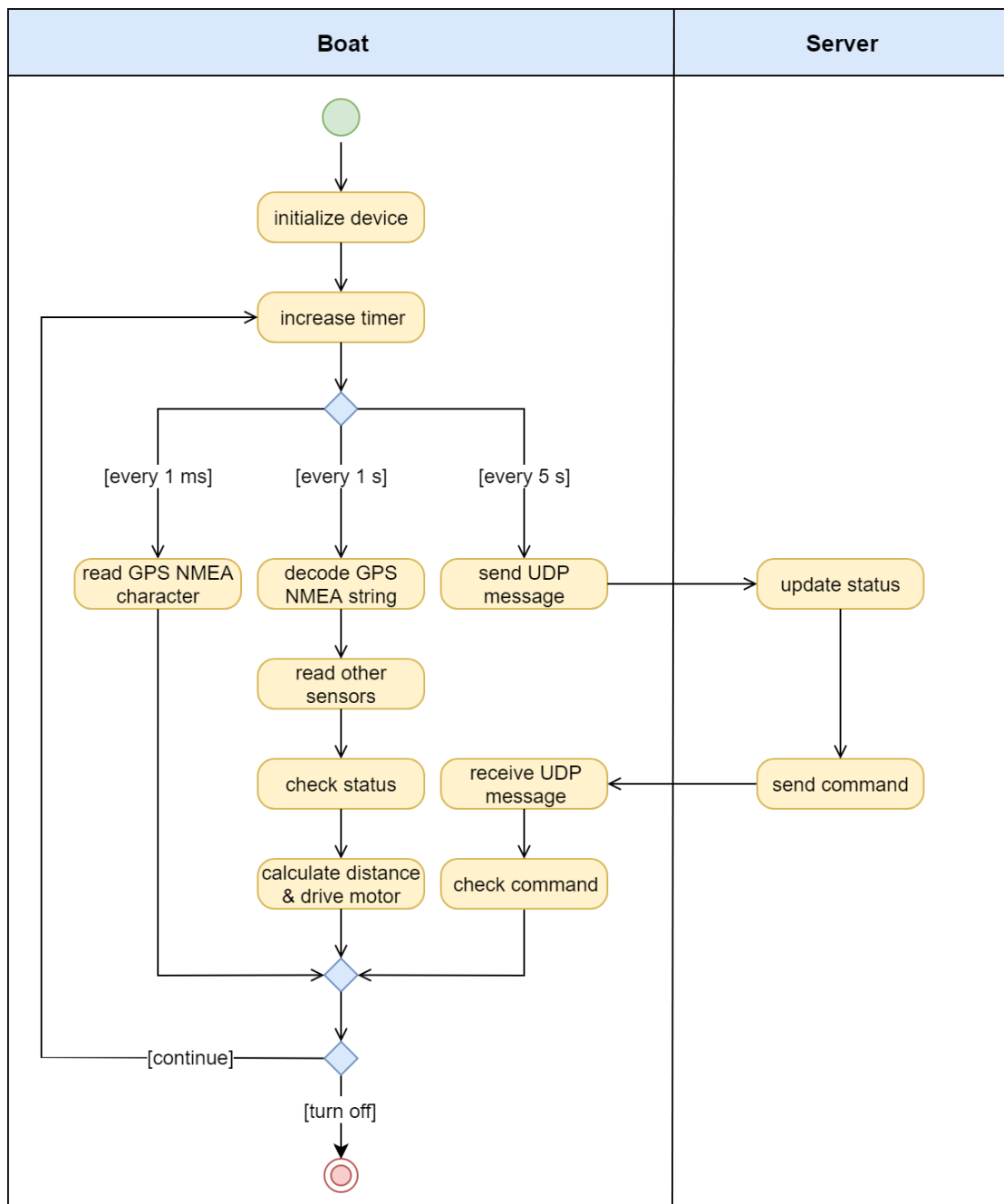
ภาพที่ 3.5 แผนภาพกิจกรรมของแอปพลิเคชันส่วนติดตามสถานะของเรือ

3.5.3 แผนภาพกิจกรรมของแอปพลิเคชันส่วนควบคุมเรือ



ภาพที่ 3.6 แผนภาพกิจกรรมของแอปพลิเคชันส่วนควบคุมเรือ

3.5.4 แผนภาพกิจกรรมของเรือ



ภาพที่ 3.7 แผนภาพกิจกรรมของเรือ


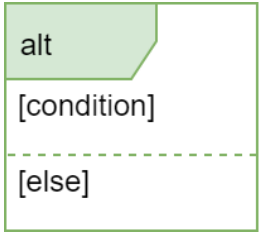
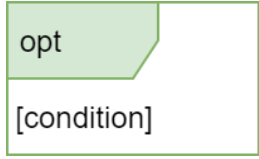

3.6 แผนภาพลำดับ

แผนภาพลำดับ (Sequence Diagram) คือแผนภาพที่ประกอบด้วยคลาส (Class) หรือวัตถุ (Object) ซึ่งใช้สำหรับแสดงลำดับของการกระทำที่เกิดขึ้นภายในระบบที่มีการติดต่อกันระหว่างคลาสหรือวัตถุ การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติ มีการออกแบบแผนภาพลำดับดังภาพที่ 3.8 - ภาพที่ 3.12 และแอปพลิเคชันสำหรับควบคุมและติดตามสถานะของเรือมีการออกแบบแผนภาพลำดับดังภาพที่ 3.13 และภาพที่ 3.14 ซึ่งสัญลักษณ์ที่ใช้แสดงในแผนภาพลำดับสามารถอธิบายได้ดังตารางที่ 3.3 ดังนี้

ตารางที่ 3.3 สัญลักษณ์และความหมายของแผนภาพลำดับ

ชื่อสัญลักษณ์	คำอธิบาย	ตัวอย่างสัญลักษณ์
Actor	ผู้ที่เกี่ยวข้องกับระบบ	 Actor
Lifeline	เส้นแสดงชีวิตของคลาสหรือวัตถุ	
Object	วัตถุที่ทำหน้าที่ตอบสนองต่อ Actor	 Object
Message	การส่งข้อความจากวัตถุหนึ่งไปยังอีกวัตถุหนึ่ง	
Self Call	การประมวลผลและคืนค่าที่ได้ภายในวัตถุเดียวกัน	 self call
Sequence Diagram Name	ชื่อของแผนภาพลำดับ	 SD : Name

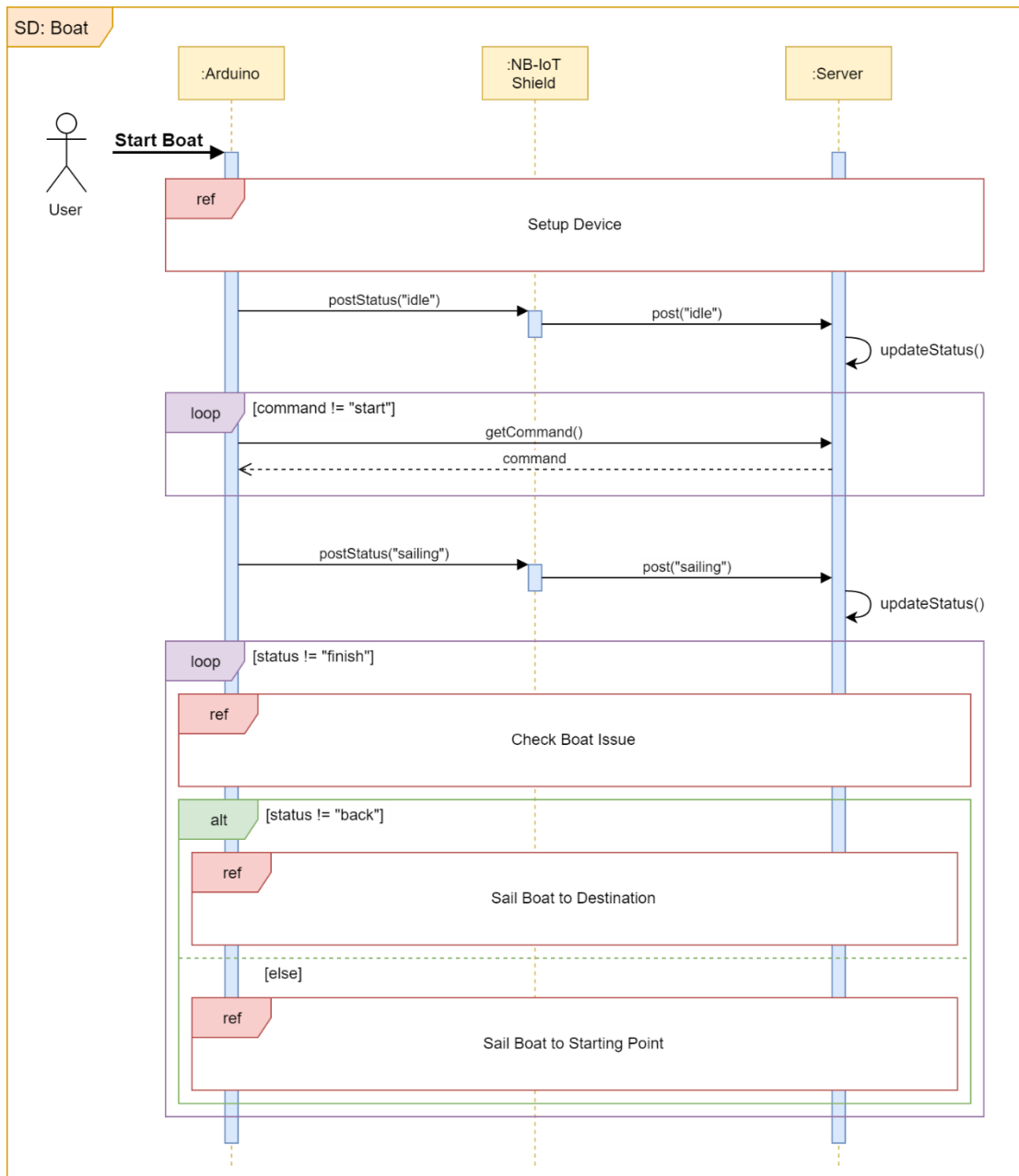
ตารางที่ 3.3 สัญลักษณ์และความหมายของแผนภาพลำดับ (ต่อ)

ชื่อสัญลักษณ์	คำอธิบาย	ตัวอย่างสัญลักษณ์
Loop	แสดงกรอบส่วนที่มีการทำงานซ้ำ	
Alternate	แสดงกรอบส่วนที่มีทางเลือกมากกว่าหนึ่งทางเลือก	
Option	แสดงกรอบส่วนที่มีทางเลือกเพื่อบอกว่าถูกใช้งานเมื่อใด	
Reference	แสดงการอ้างอิงถึงแผนภาพลำดับที่แยกออกมา	

3.6.1 แผนภาพลำดับการทำงานของเรือ

แผนภาพลำดับการทำงานของเรือขับเคลื่อนอัตโนมัติ สามารถอธิบายลำดับการทำงานตามภาพที่ 3.8 ได้ดังนี้

- 1) เริ่มการทำงานของเรือ
- 2) เริ่มติดตั้งโปรแกรมสำหรับใช้งานอุปกรณ์หรือโมดูลอื่น ๆ บนเรือ ซึ่งแบ่งเป็นแผนภาพลำดับการติดตั้งอุปกรณ์ ดังแสดงในภาพที่ 3.9
- 3) ส่งสถานะ ว่าง ให้กับเซิร์ฟเวอร์เพื่อบอกว่า เรือพร้อมรับคำสั่งแล้ว
- 4) ตรวจสอบคำสั่งจากเซิร์ฟเวอร์ เพื่อหยุดรอรับคำสั่งเดินเรือและตำแหน่งจุดหมาย
- 5) เมื่อได้รับคำสั่งให้เดินเรือจากเซิร์ฟเวอร์ ระบบจะส่งสถานะ กำลังเดินเรือ ไปยังเซิร์ฟเวอร์ และเริ่มทำการเดินเรือไปยังจุดหมาย
- 6) ตรวจสอบปัญหาการทำงานของอุปกรณ์ในเรือ ดังแสดงในภาพที่ 3.10
- 7) เมื่อไม่พบปัญหาจะเริ่มควบคุมเรือไปและกลับ ดังแสดงในภาพที่ 3.11 และภาพที่ 3.12



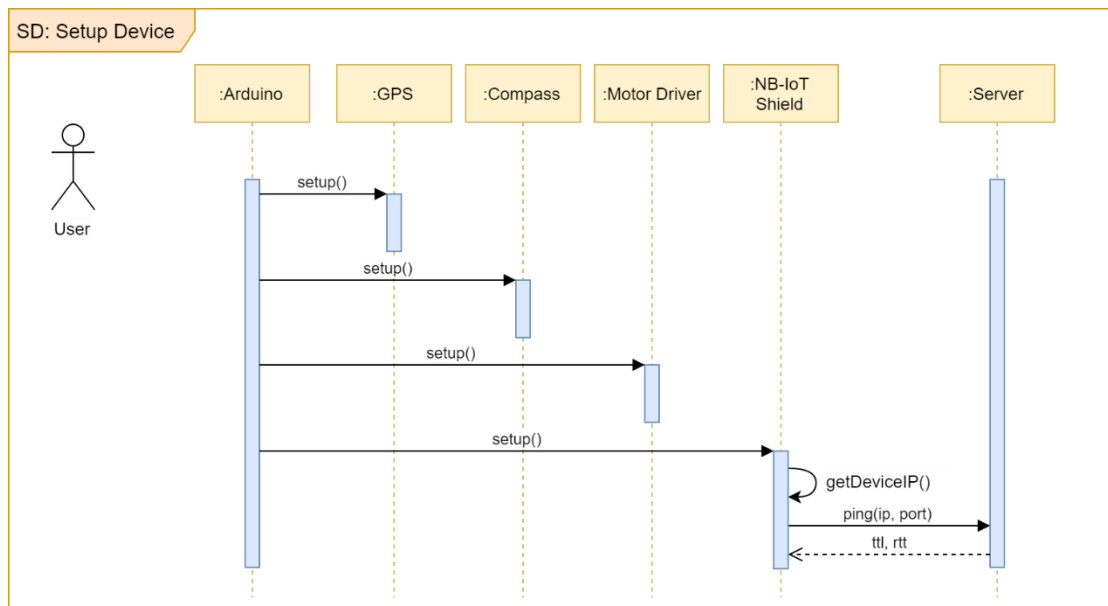
ภาพที่ 3.8 แผนภาพลำดับการทำงานของเรือ

3.6.2 แผนภาพลำดับการติดตั้งอุปกรณ์

แผนภาพลำดับการติดตั้งอุปกรณ์ เพื่อตั้งค่าเริ่มต้นให้กับอุปกรณ์ สามารถอธิบายลำดับการทำงานตามภาพที่ 3.9 ได้ดังนี้

- 1) ติดตั้งโมดูลจีพีเอส เพื่อให้สามารถรับข้อมูลจากโมดูลได้ เช่น ละติจูดและลองจิจูด
- 2) ติดตั้งโมดูลเข็มทิศ เพื่อให้สามารถรับข้อมูลจากโมดูลได้ เช่น ภาคทิศ
- 3) ติดตั้งโมดูลขับเคลื่อนมอเตอร์ เพื่อตั้งค่าเริ่มต้นสำหรับควบคุมมอเตอร์

- 4) เชื่อมต่อระบบเข้ากับสถานีฐานผ่านเอ็นบีไอโอทีและรับไอพีแอดเดรส (IP Address) ของอุปกรณ์ จากนั้นทำการปิงกับเครือข่ายคลาวด์เพื่อตรวจสอบความแรงของสัญญาณ

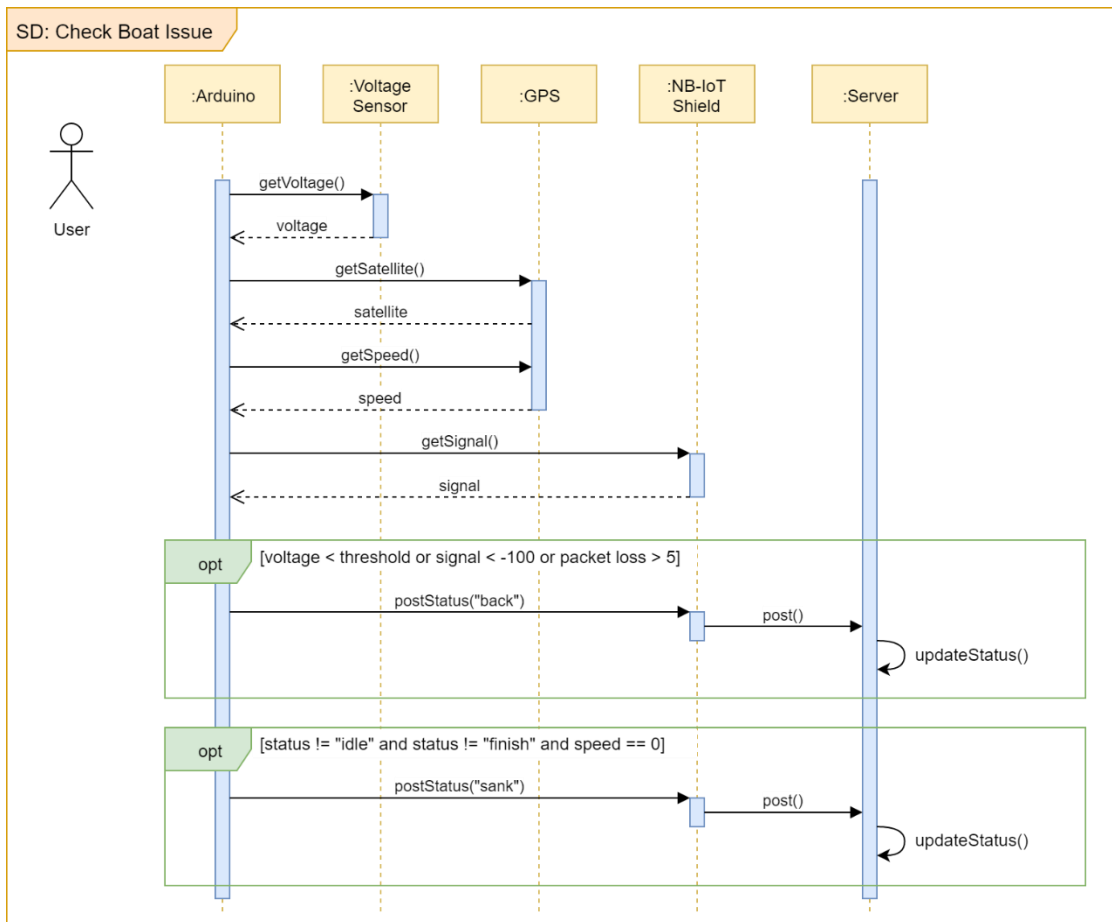


ภาพที่ 3.9 แผนภาพลำดับการติดตั้งอุปกรณ์

3.6.3 แผนภาพลำดับการตรวจสอบข้อผิดพลาดของอุปกรณ์ภายในเรือ

แผนภาพลำดับการตรวจสอบข้อผิดพลาดของอุปกรณ์ภายในเรือ สามารถอธิบายลำดับการทำงานตามภาพที่ 3.10 ได้ดังนี้

- 1) อ่านค่าโวลเตจของแบตเตอรี่จากโวลเตจเซนเซอร์
- 2) อ่านค่าจำนวนดาวเทียมที่ตรวจพบและความเร็วการเคลื่อนที่จากโมดูลจีพีเอส
- 3) อ่านค่าความแรงของสัญญาณจากบอร์ดเสริมเอ็นบีไอโอที
- 4) หากพบข้อผิดพลาดให้ส่งสถานะ กลับไปที่เซิร์ฟเวอร์
- 5) หากพบว่าเรือไม่เคลื่อนที่แต่ยังอยู่ในช่วงการเดินทางเรือ ให้ส่งสถานะ เรือล่ม ไปที่เซิร์ฟเวอร์

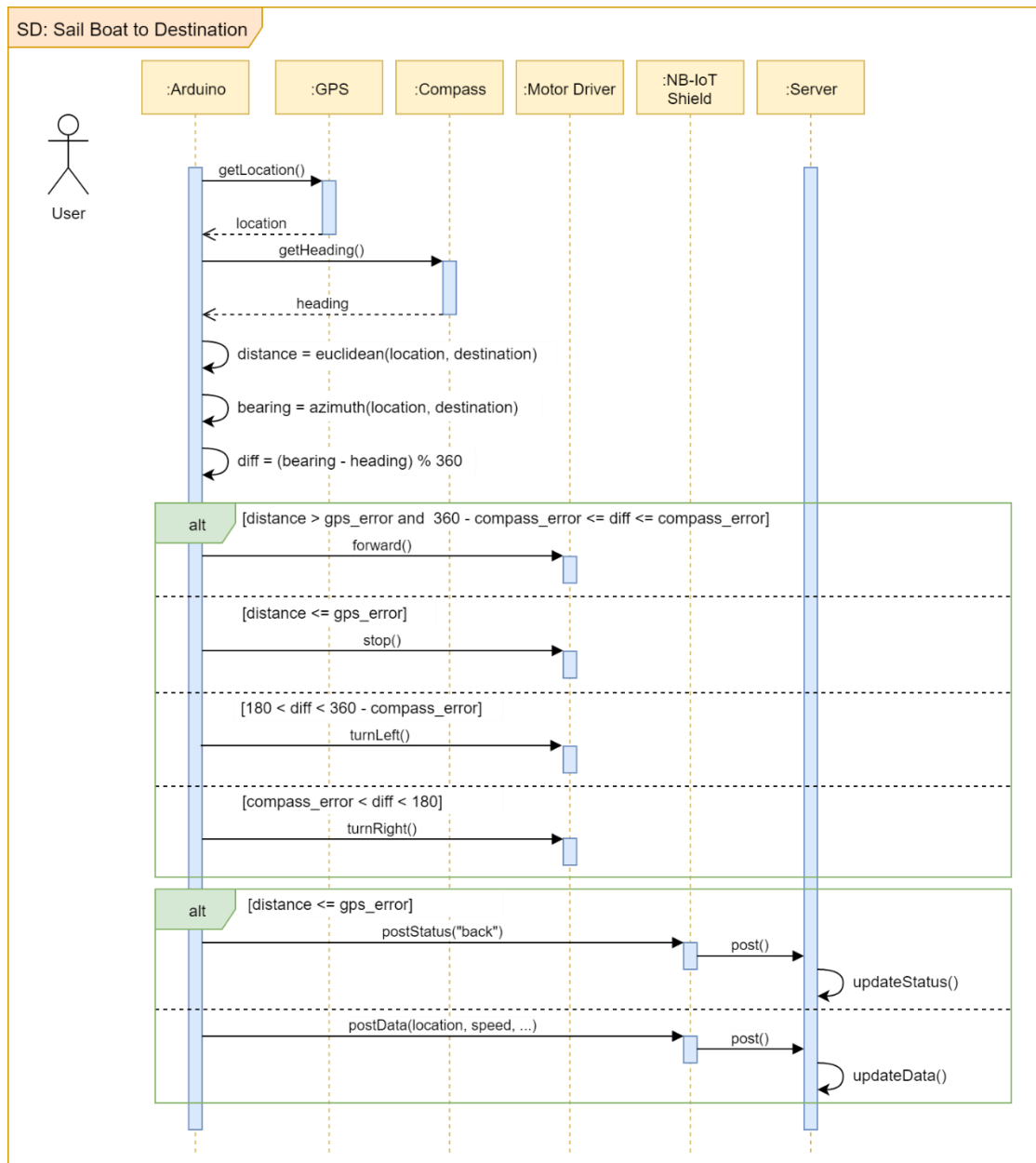


ภาพที่ 3.10 แผนภาพลำดับการตรวจสอบข้อผิดพลาดของอุปกรณ์ภายในเรือ

3.6.4 แผนภาพลำดับการควบคุมเรือไปยังจุดหมาย

แผนภาพลำดับการควบคุมเรือไปยังจุดหมาย สามารถอธิบายลำดับการทำงานตามภาพที่ 3.11 ได้ดังนี้

- 1) อ่านค่าตำแหน่งปัจจุบันของเรือจากโมดูลจีพีเอสในรูปแบบละติจูดและลองจิจูด
- 2) อ่านค่าทิศทางที่เรือกำลังหันจากโมดูลเข็มทิศ
- 3) คำนวณระยะทางและทิศทางระหว่างตำแหน่งปัจจุบันของเรือและตำแหน่งจุดหมาย
- 4) ควบคุมการทำงานของมอเตอร์ผ่านโมดูลขับเคลื่อนมอเตอร์ตามค่าที่คำนวณได้
- 5) เมื่อถึงจุดหมายให้ส่งสถานะ กลับไปที่เซิร์ฟเวอร์ หากยังไม่ถึงจุดหมาย ให้ส่งข้อมูลต่าง ๆ ของเรือไปแทน



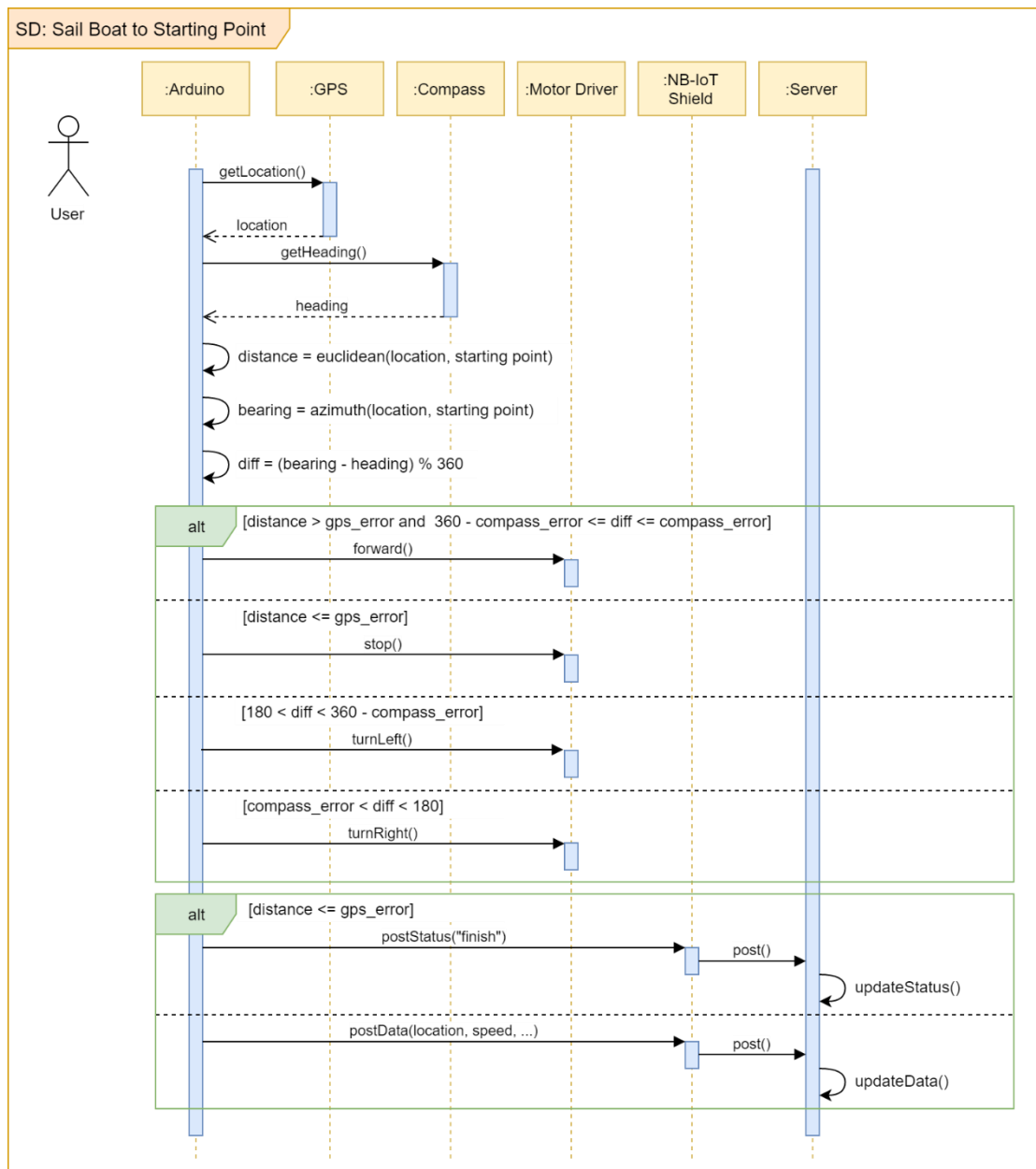
ภาพที่ 3.11 แผนภาพลำดับการควบคุมเรือไปยังจุดหมาย

3.6.5 แผนภาพลำดับการควบคุมเรือกลับมายังจุดเริ่มต้น

แผนภาพลำดับการควบคุมเรือกลับมายังจุดเริ่มต้น สามารถอธิบายลำดับการทำงานตามภาพที่ 3.12 ได้ดังนี้

- 1) อ่านค่าตำแหน่งปัจจุบันของเรือจากโมดูลจีพีเอสในรูปแบบละติจูดและลองจิจูด
- 2) อ่านค่าทิศทางที่เรือกำลังหันจากโมดูลเข็มทิศ
- 3) คำนวณระยะทางและทิศทางระหว่างตำแหน่งปัจจุบันของเรือและตำแหน่งจุดเริ่มต้น
- 4) ควบคุมการทำงานของมอเตอร์ผ่านโมดูลขับมอเตอร์ตามค่าที่คำนวณได้

- 5) เมื่อถึงจุดหมายให้ส่งสถานะ สิ้นสุด (Finish) ไปที่เซิร์ฟเวอร์ หากยังไม่ถึงจุดเริ่มต้นให้ส่งข้อมูลต่าง ๆ ของเรือไปแทน



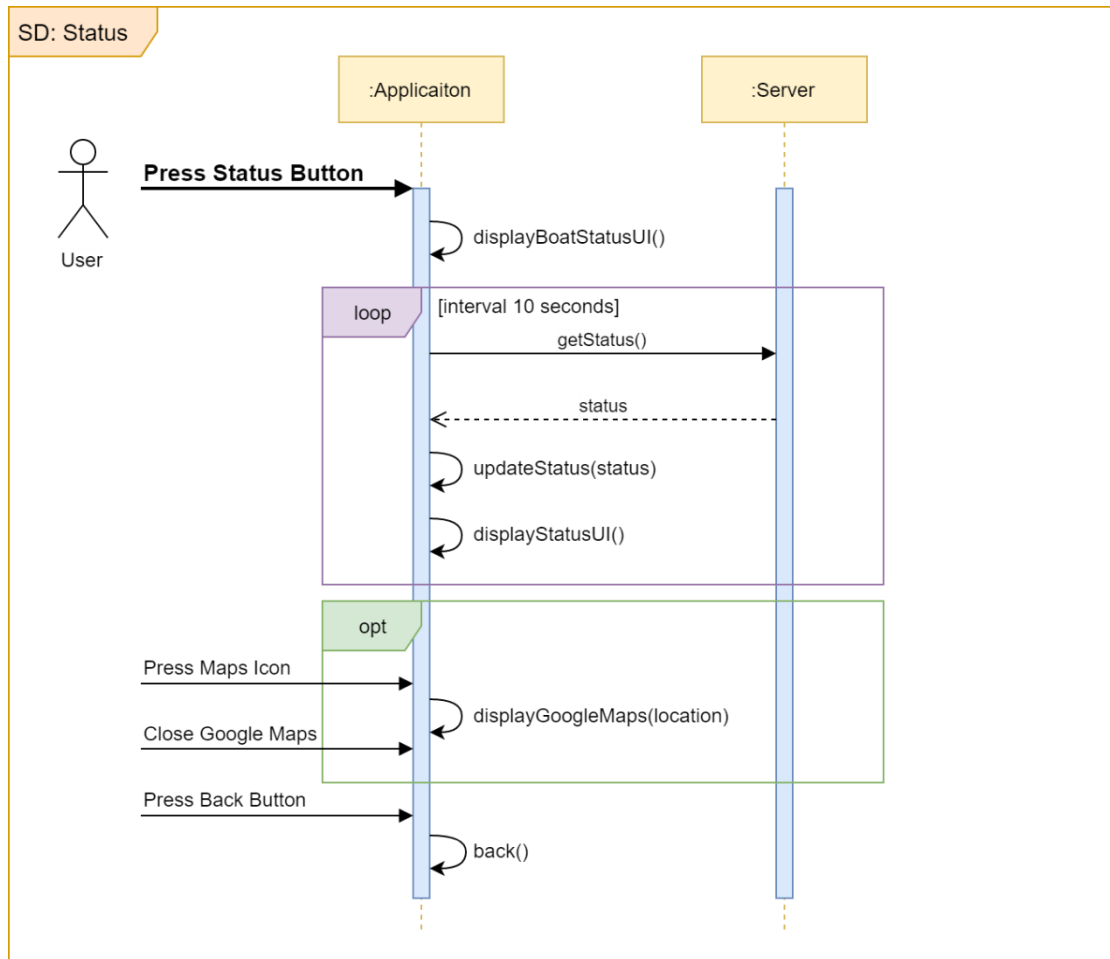
ภาพที่ 3.12 แผนภาพลำดับการควบคุมเรือกลับมายังจุดเริ่มต้น

3.6.6 แผนภาพลำดับการติดตามสถานะของเรือผ่านแอปพลิเคชัน

แผนภาพลำดับการติดตามสถานะของเรือผ่านแอปพลิเคชัน สามารถอธิบายลำดับการทำงานตามภาพที่ 3.13 ได้ดังนี้

- 1) เข้าสู่หน้าจอแสดงสถานะของเรือ ได้แก่ ตำแหน่งปัจจุบันของเรือ, ความเร็วการเคลื่อนที่ของเรือและพลังงานในแบตเตอรี่
- 2) ดึงสถานะและข้อมูลของเรือจากเซิร์ฟเวอร์มาแสดงผลบนแอปพลิเคชัน

- 3) สามารถแสดงตำแหน่งปัจจุบันของเรือผ่านกูเกิลแมพได้
- 4) กลับไปยังหน้าจอหลัก



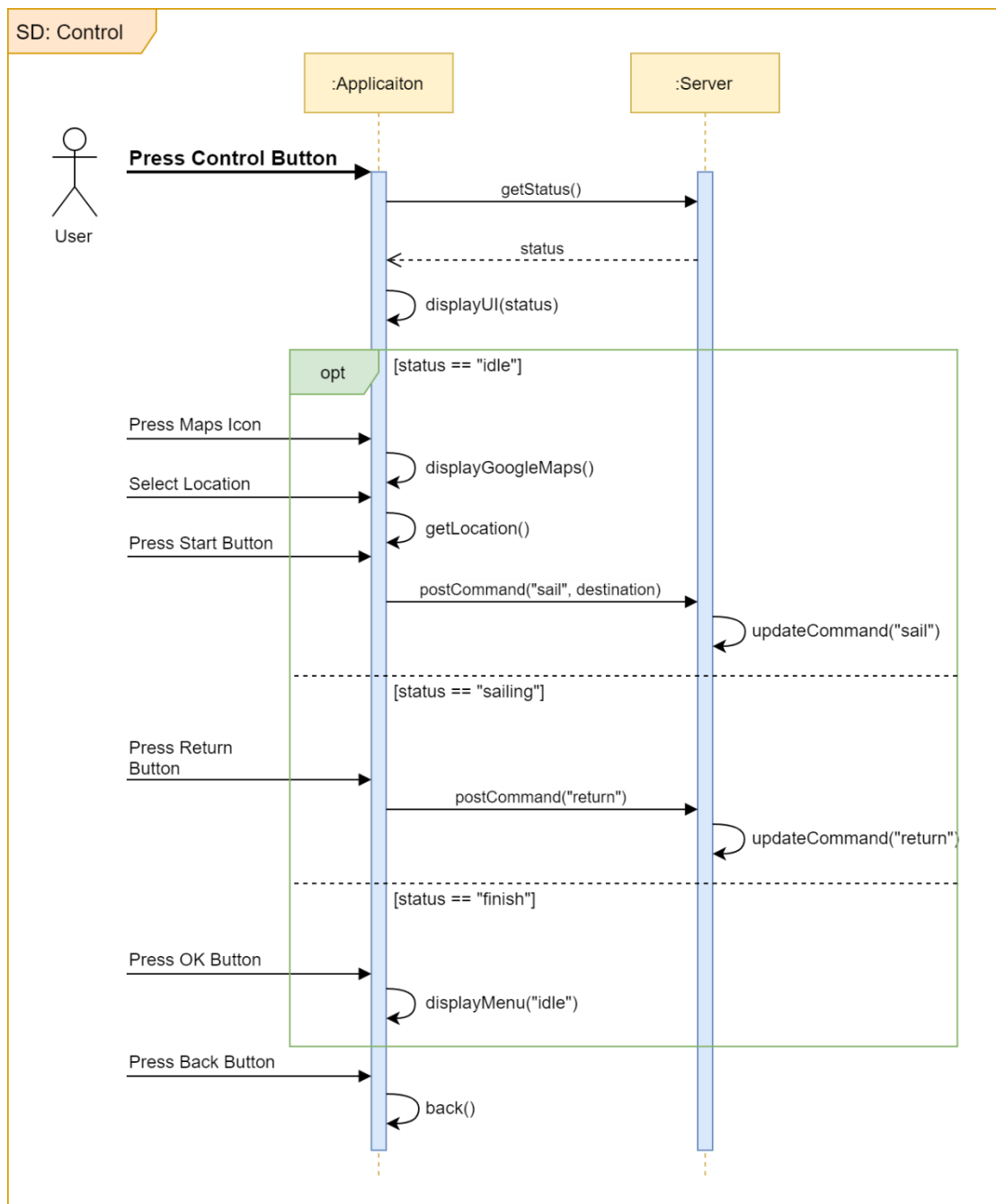
ภาพที่ 3.13 แผนภาพลำดับการติดตามสถานะของเรือผ่านแอปพลิเคชัน

3.6.7 แผนภาพลำดับการควบคุมเรือผ่านแอปพลิเคชัน

แผนภาพลำดับการควบคุมเรือผ่านแอปพลิเคชัน สามารถอธิบายลำดับการทำงานตามภาพที่ 3.14 ได้ดังนี้

- 1) เข้าสู่หน้าจอควบคุมเรือจากหน้าจอหลัก
- 2) ดึงสถานะของเรือจากเซิร์ฟเวอร์
- 3) แสดงเมนูตามค่าสถานะดังนี้
 - 3.1) ถ้าเรืออยู่ในสถานะ วาง ผู้ใช้จะสามารถเลือกตำแหน่งจุดหมายเพื่อใช้งานระบบขับเคลื่อนอัตโนมัติได้
 - 3.2) ถ้าเรืออยู่ในสถานะ กำลังเดินเรือ ผู้ใช้จะสามารถกดสั่งให้เรือกลับมาที่จุดเริ่มต้นได้
 - 3.3) ถ้าเรืออยู่ในสถานะ สิ้นสุด ผู้ใช้จะสามารถกดปุ่มยืนยันเพื่อรับทราบว่าการทำงานของเรือสิ้นสุดแล้ว

4) กลับไปยังหน้าจอหลัก




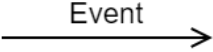


ภาพที่ 3.14 แผนภาพลำดับการควบคุมเรือผ่านแอปพลิเคชัน

3.7 แผนภาพสถานะ

แผนภาพสถานะ (State Diagram) เป็นแผนภาพที่ใช้แสดงถึงสถานะของวัตถุในช่วงเวลาหนึ่ง และเหตุการณ์ต่าง ๆ ที่สามารถทำให้สถานะของวัตถุมีการเปลี่ยนแปลงไปจากเดิม ซึ่งสถานะของเรือขับเคลื่อนอัตโนมัติและแอปพลิเคชันสำหรับติดตามการทำงานของเรือจะเก็บไว้บนเซิร์ฟเวอร์ โดยสามารถแสดงแผนภาพสถานะได้ดังภาพที่ 3.15 และภาพที่ 3.16 ซึ่งสัญลักษณ์ที่ใช้แสดงในแผนภาพสถานะสามารถอธิบายได้ดังตารางที่ 3.4 ดังนี้

ตารางที่ 3.4 สัญลักษณ์และความหมายของแผนภาพสถานะ

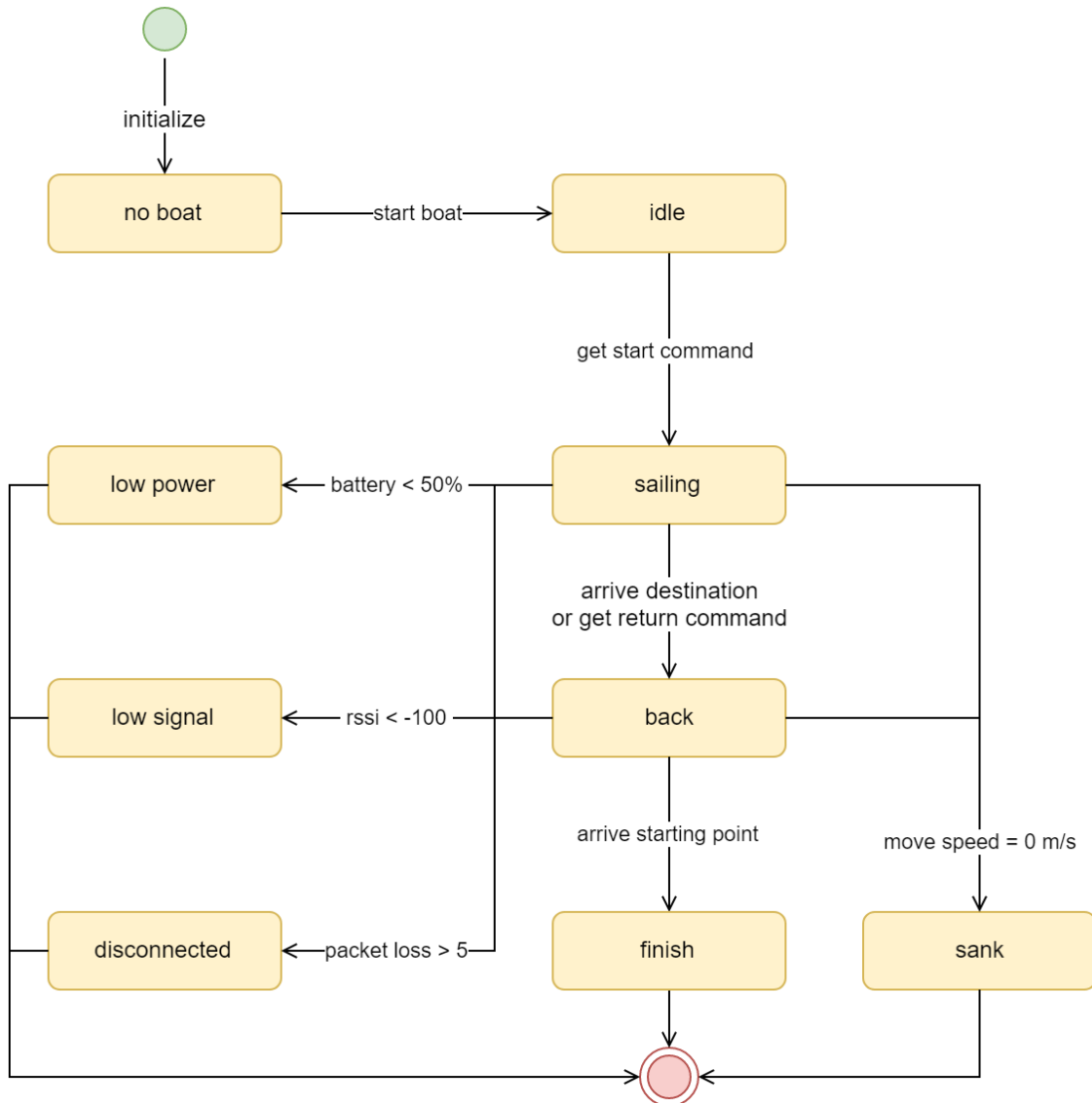
ชื่อสัญลักษณ์	คำอธิบาย	ตัวอย่างสัญลักษณ์
Initial State	จุดเริ่มต้นของคลาสหรือวัตถุ	
Final State	จุดสิ้นสุดของสถานะ	
State	สถานะของวัตถุในช่วงเวลาหนึ่ง	
Event	เหตุการณ์ที่ส่งผลให้วัตถุเปลี่ยนสถานะ	

3.7.1 แผนภาพสถานะของเรือ

แผนภาพสถานะของเรือ สามารถอธิบายลำดับการเปลี่ยนแปลงของสถานะ ตามภาพที่ 3.12 ได้ดังนี้

- 1) เริ่มต้นสถานะจะถูกตั้งเป็น ไม่พบเรือ
- 2) สถานะจะถูกเปลี่ยนเป็น วาง เมื่อเรือถูกสตาร์ท
- 3) สถานะจะถูกเปลี่ยนเป็น กำลังเดินเรือ เมื่อเรือได้รับคำสั่งให้เริ่มเดินเรือจากแอปพลิเคชันผ่านทางเซิร์ฟเวอร์
- 4) สถานะจะถูกเปลี่ยนเป็น กลับ เมื่อได้รับคำสั่งให้กลับหรือเมื่อเรือไปถึงจุดหมาย
- 5) สถานะจะถูกเปลี่ยนเป็น พลังงานต่ำ เมื่อตรวจพบว่าพลังงานแบตเตอรี่ที่เหลืออยู่ไม่เพียงพอที่จะเดินทางต่อ
- 6) สถานะจะถูกเปลี่ยนเป็น สัญญาณต่ำ เมื่อตรวจพบว่าเรือออกห่างจากระยะสัญญาณมากเกินไป

- 7) สถานะจะถูกเปลี่ยนเป็น ขาดการเชื่อมต่อ เมื่อการส่งข้อมูลสูญหายมากเกินไป
- 8) สถานะจะถูกเปลี่ยนเป็น สิ้นสุด เมื่อเรือกลับมาถึงจุดเริ่มต้น
- 9) สถานะจะถูกเปลี่ยนเป็น เรือล่ม เมื่อเรือไม่มีการเคลื่อนที่เป็นระยะเวลาหนึ่ง

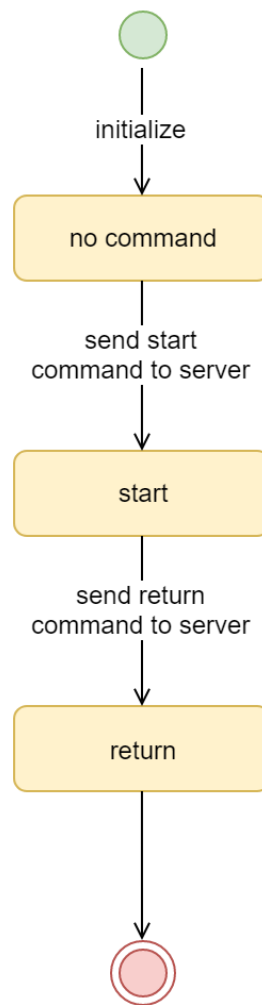


ภาพที่ 3.15 แผนภาพสถานะของเรือ

3.7.2 แผนภาพสถานะของแอปพลิเคชัน

แผนภาพสถานะของแอปพลิเคชัน สามารถอธิบายลำดับการเปลี่ยนแปลงของสถานะ ตามภาพที่ 3.13 ได้ดังนี้

- 1) เริ่มต้นสถานะจะถูกตั้งเป็น ไม่มีคำสั่ง
- 2) สถานะจะถูกเปลี่ยนเป็น เดินเรือ เมื่อผู้ใช้สั่งให้เดินเรือ
- 3) สถานะจะถูกเปลี่ยนเป็น กลับมา เมื่อผู้ใช้สั่งให้เรือกลับมาที่จุดเริ่มต้น



ภาพที่ 3.16 แผนภาพสถานะของแอปพลิเคชัน

3.8 การออกแบบฐานข้อมูล

ระบบนี้จะใช้ฐานข้อมูลเพื่อเป็นตัวกลางสำหรับใช้ติดต่อระหว่าง โดยจะใช้มายเอสคิวแอล (MySQL) เป็นตัวช่วยในการสร้างและจัดการฐานข้อมูล รายละเอียดต่อไปนี้จะกล่าวถึงการออกแบบฐานข้อมูลของระบบ

ตารางที่ 3.5 แสดงตารางสถานะของเรือ

ชื่อตาราง		boat			
คำอธิบาย		ตารางเก็บข้อมูลสถานะของเรือ			
คีย์หลัก		boat_id			
ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิดข้อมูล	ขนาด (ไบต์)	ตัวอย่างข้อมูล
1	boat_id	รหัสของเรือ	char	9	b1
2	status	สถานะ	varchar	25	idle
3	curr_loc	ตำแหน่งปัจจุบันของเรือ	varchar	25	18.800677_98.955631
4	dest_loc	ตำแหน่งเป้าหมาย	varchar	25	18.800677_98.955631
5	home_loc	ตำแหน่งเริ่มต้น	varchar	25	18.800677_98.955631
6	battery	พลังงานที่เหลืออยู่	smallint	2	100
7	rss_i	ความแรงของสัญญาณ	smallint	2	-80
8	distance	ระยะทางที่เหลืออยู่	float	4	166.5
9	speed	ความเร็วการเคลื่อนที่	float	4	3.5
10	cmd	คำสั่งจากแอปพลิเคชัน	varchar	25	start

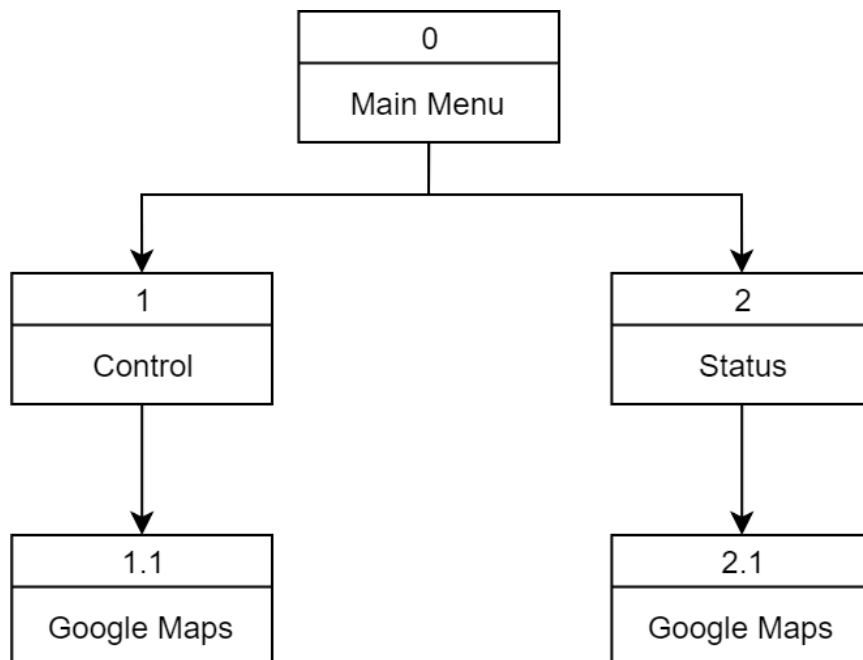
บทที่ 4

การออกแบบส่วนต่อประสานกับผู้ใช้

การออกแบบส่วนต่อประสานกับผู้ใช้ เป็นส่วนที่มีความสำคัญอย่างยิ่งในการจัดทำระบบ เนื่องจากเป็นส่วนที่ต้องใช้ในการติดต่อสื่อสารกับผู้ใช้งาน จึงจำเป็นต้องมีการออกแบบให้มีลักษณะการใช้งานง่าย ซึ่งในบทนี้จะแสดงรายละเอียดของการออกแบบส่วนต่อประสานกับผู้ใช้ดังนี้

4.1 การออกแบบส่วนต่อประสาน

การออกแบบส่วนต่อประสานสามารถแบ่งออกตามหน้าที่การดำเนินการต่าง ๆ ได้ดังนี้

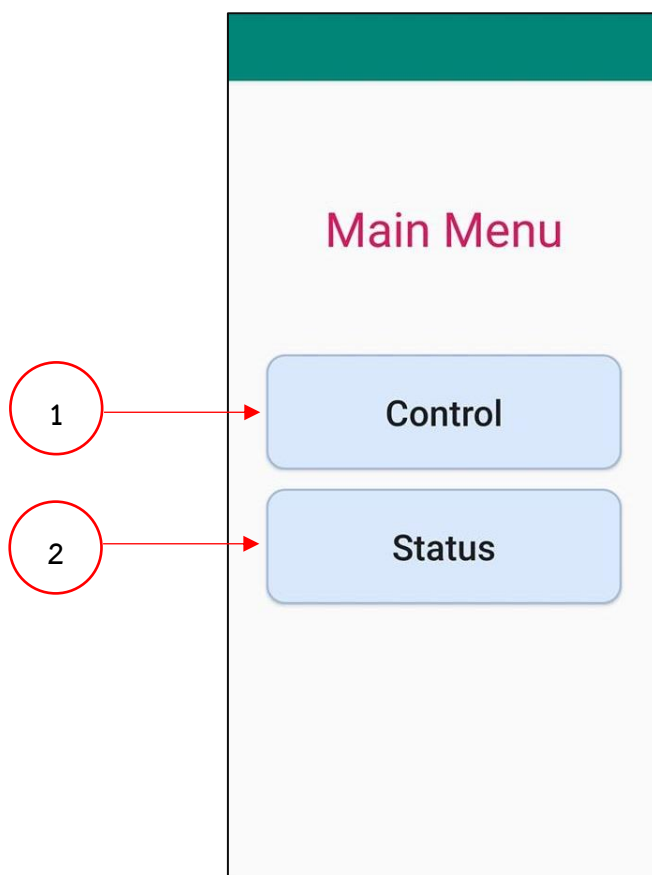


ภาพที่ 4.1 การทำงานของส่วนต่อประสาน

4.2 การทำงานของส่วนต่อประสาน

4.2.1 หน้าจอหลัก

หน้าจอหลัก (Main Menu) เป็นหน้าจอแรกที่แสดงเมื่อเข้าสู่แอปพลิเคชัน สามารถแสดงตัวอย่างหน้าจอได้ดังภาพที่ 4.2



ภาพที่ 4.2 หน้าจอหลัก

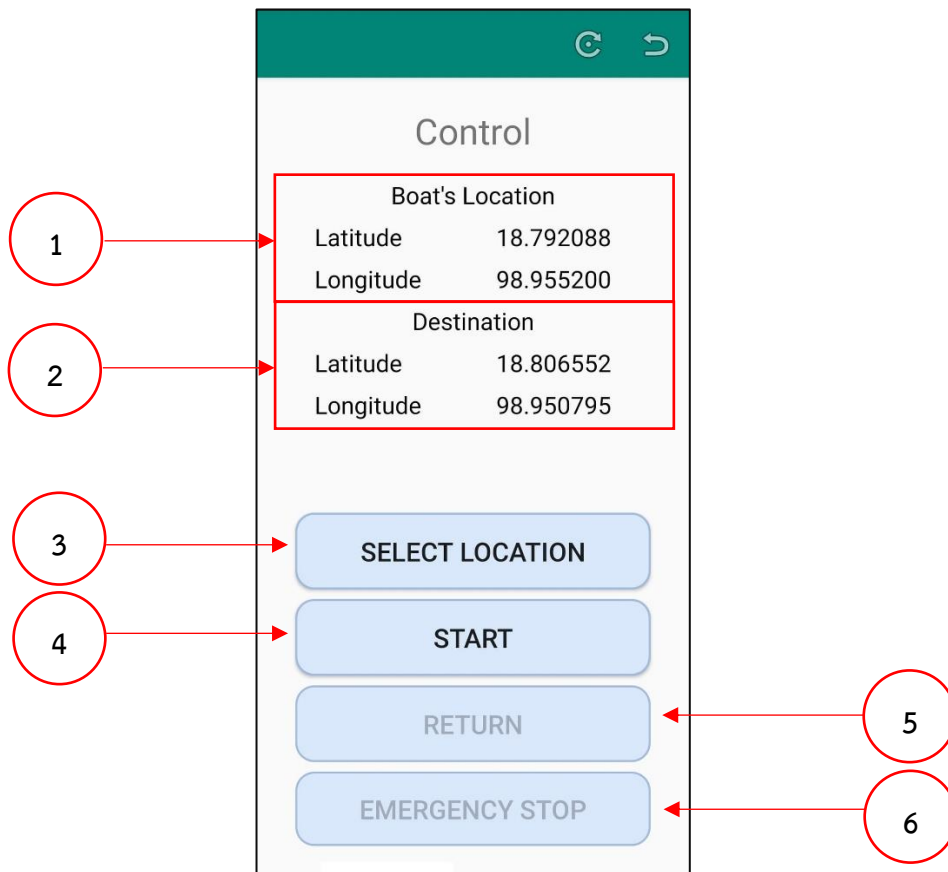
จากภาพที่ 4.2 สามารถอธิบายรายละเอียดของจอภาพได้ดังนี้

ส่วนที่ 1 ปุ่มกดเพื่อไปยังหน้าจอควบคุมเรือ ดังแสดงในภาพที่ 4.3

ส่วนที่ 2 ปุ่มกดเพื่อไปยังหน้าจอแสดงสถานะของเรือ ดังแสดงในภาพที่ 4.4

4.2.2 หน้าจอกควบคุมเรือ

หน้าจอกควบคุมเรือ (Control) เป็นหน้าจอสำหรับควบคุมเรือซึ่งสามารถเลือกตำแหน่งจุดหมายให้กับเรือได้ สามารถแสดงตัวอย่างหน้าจอได้ดังภาพที่ 4.3



ภาพที่ 4.3 หน้าจอกควบคุมเรือ

จากภาพที่ 4.3 สามารถอธิบายรายละเอียดของจอภาพได้ดังนี้

ส่วนที่ 1 ตำแหน่งปัจจุบันของเรือที่แสดงในรูปแบบพิกัดละติจูดและลองจิจูด

ส่วนที่ 2 ตำแหน่งของเป้าหมายที่แสดงในรูปแบบพิกัดละติจูดและลองจิจูด สามารถเลือกได้จากส่วนที่ 3

ส่วนที่ 3 ปุ่มกดเพื่อไปยังหน้าจอเลือกตำแหน่งบนกูเกิลแมพ

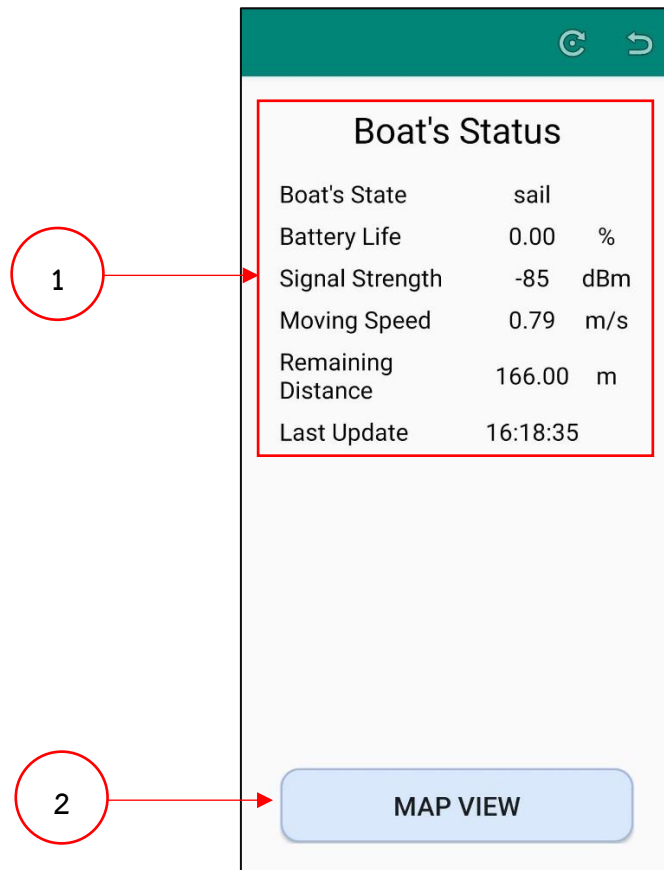
ส่วนที่ 4 ปุ่มกดเพื่อสั่งให้เรือเริ่มทำงาน หากกดปุ่มนี้จะทำการส่งคำสั่งเริ่มการทำงานและพิกัดเป้าหมายไปยังเรือผ่านทางเซิร์ฟเวอร์

ส่วนที่ 5 ปุ่มกดเพื่อสั่งให้เรือกลับมาที่จุดเริ่มต้น จะเปิดให้ใช้งานได้หลังจากกดปุ่มในส่วนที่ 4

ส่วนที่ 6 ปุ่มกดเพื่อสั่งให้เรือหยุด จะเปิดให้ใช้งานได้หลังจากกดปุ่มในส่วนที่ 4

4.2.3 หน้าจอติดตามสถานะ

หน้าจอติดตามสถานะ (Status) เป็นหน้าจอที่แสดงสถานะต่าง ๆ ของเรือ เช่น สถานะปัจจุบันของเรือ พลังงานที่เหลืออยู่ในแบตเตอรี่ ความแรงของสัญญาณเอ็นบีไอโอที ความเร็ว การเคลื่อนที่ ระยะทางที่เหลืออยู่ สามารถแสดงตัวอย่างหน้าจอได้ดังภาพที่ 4.4



ภาพที่ 4.4 หน้าจอแสดงสถานะของเรือ

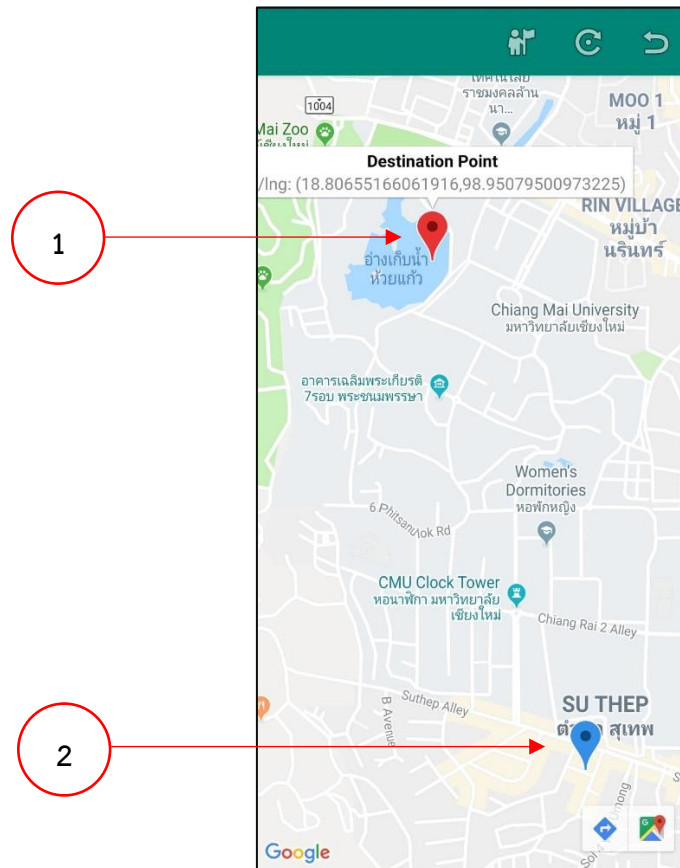
จากภาพที่ 4.4 สามารถอธิบายรายละเอียดของจอภาพได้ดังนี้

ส่วนที่ 1 สถานะต่าง ๆ ของเรือ

ส่วนที่ 2 ปุ่มกดเพื่อไปยังหน้าจอแสดงตำแหน่งของเรือและเป้าหมายบนกูเกิลแมพ

4.2.4 หน้าจอเกิลแมพ

หน้าจอเกิลแมพ ใช้สำหรับแสดงตำแหน่งปัจจุบันของเรือ หรือเลือกตำแหน่งจุดหมายให้กับเรือ สามารถแสดงตัวอย่างหน้าจอได้ดังภาพที่ 4.5



ภาพที่ 4.5 หน้าจอสำหรับเลือกตำแหน่งบนแผนที่กูเกิลแมพ

จากภาพที่ 4.5 สามารถอธิบายรายละเอียดของจอภาพได้ดังนี้

ส่วนที่ 1 ตำแหน่งที่ถูกเลือกบนแผนที่

ส่วนที่ 2 ตำแหน่งปัจจุบันของเรือ

บทที่ 5

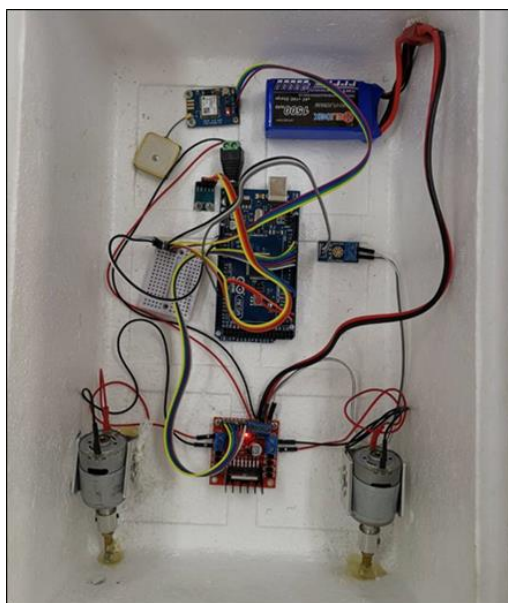
การพัฒนาและทดสอบระบบ

5.1 การพัฒนาระบบ

การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติจะแบ่งย่อยระบบออกเป็น 3 ส่วน ได้แก่ ยานพาหนะสำหรับติดตั้งอุปกรณ์ไอโอที แอปพลิเคชันสำหรับควบคุมและติดตามสถานะการทำงานของยานพาหนะ และเซิร์ฟเวอร์สำหรับสร้างช่องทางสื่อสารระหว่างยานพาหนะและแอปพลิเคชัน

5.1.1 ยานพาหนะสำหรับติดตั้งอุปกรณ์ไอโอที

ยานพาหนะถูกพัฒนาให้อยู่ในรูปแบบของเรือ โดยจะมีระบบขับเคลื่อนอัตโนมัติที่ควบคุมการทำงานด้วยอุปกรณ์ไอโอที ดังที่กล่าวไว้ในบทที่ 3 และใช้โปรแกรมอาดุยโนไอทีสำหรับพัฒนาระบบด้วยภาษาซีพลัสพลัส อุปกรณ์หลักที่ใช้สำหรับควบคุมระบบทั้งหมดภายในเรือคือบอร์ดอาดุยโนเมกะ



ภาพที่ 5.1 เรือขับเคลื่อนอัตโนมัติ

ระบบขับเคลื่อนอัตโนมัตินี้จะถูกแบ่งออกเป็น 4 ส่วน ดังต่อไปนี้

- 1) การส่งข้อมูลระหว่างเรือและเซิร์ฟเวอร์

ในส่วนนี้จะรับส่งข้อมูลแบบยูทีพีด้วยบอร์ดเสริมเอไอเอสเอ็นพีไอไอที ข้อมูลที่ถูกส่งไปยังเซิร์ฟเวอร์ ซึ่งเป็นค่าสถานะของเรือ ได้แก่

- สถานะการทำงานปัจจุบันของเรือ

- ตำแหน่งปัจจุบันของเรือในรูปแบบละติจูดและลองจิจูด
- ระยะทางระหว่างเรือและเป้าหมายหน่วยเมตร
- ความเร็วการเคลื่อนที่หน่วยเมตรต่อวินาที
- พลังงานที่เหลืออยู่ในหน่วยเปอร์เซ็นต์
- ค่าความแรงของสัญญาณหน่วยดีบีเอ็ม

ข้อมูลบางส่วนที่ส่งไปยังเซิร์ฟเวอร์มีรูปแบบตัวอย่างดังนี้

"{status:idle,loc:18.800677_98.955631,battery:98,rtt:87,etc.}"

เนื่องจากข้อมูลรูปแบบแรกมีขนาดของแพ็กเก็ตใหญ่เกินความจำเป็นเพราะมีการใส่คำนิยาม (Label) ลงไป ทำให้มีพื้นที่สำหรับส่งข้อมูลจากเซนเซอร์อื่น ๆ น้อยลง จึงมีการออกแบบเป็นรูปแบบใหม่ซึ่งมีลักษณะดังนี้

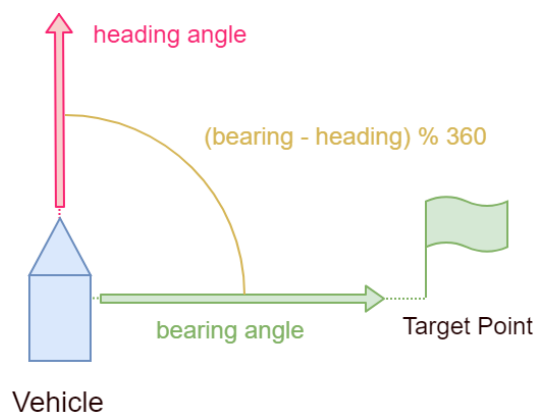
"{idle,18.800677_98.955631,98,-87,etc.}"

2) การตรวจจับทิศทางของเรือ

ในส่วนนี้จะใช้สำหรับตรวจจับทิศทางของเรือว่าหันไปทางทิศใดด้วยโมดูลเข็มทิศ นำค่าทิศที่ได้จากโมดูลตัวนี้ไปคำนวณหาทิศทางระหว่างเรือและเป้าหมาย เพื่อตัดสินใจให้โมดูลขับเคลื่อนควบคุมมอเตอร์ไปยังทิศทางไหน ซึ่งได้แก่

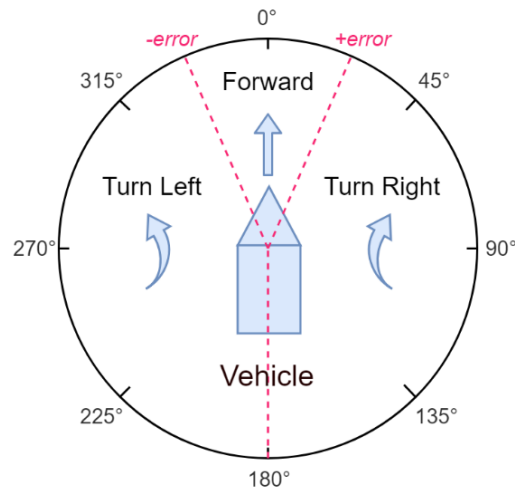
- ตรงไป (Forward) ด้วยการหมุนมอเตอร์ทั้งสองข้างด้วยความเร็วเท่ากัน
- เลี้ยวซ้าย (Turn Left) ด้วยการหมุนมอเตอร์ข้างซ้ายช้ากว่ามอเตอร์ข้างขวา
- เลี้ยวขวา (Turn Right) ด้วยการหมุนมอเตอร์ข้างซ้ายเร็วกว่ามอเตอร์ข้างขวา

การคำนวณหาผลต่างของทิศที่เรือกำลังหันหน้า (Heading) และทิศจากเรือไปยังเป้าหมาย (Bearing) จะต้องมอดุโล (Modulo) ด้วย 360 เพื่อให้อยู่ในช่วงของภาคทิศ ดังภาพที่ 5.2



ภาพที่ 5.2 การหาผลต่างของทิศที่เรือหันและทิศจากเรือไปยังเป้าหมาย

นำค่าที่คำนวณได้ไปเปรียบเทียบกับภาพที่ 5.3 ซึ่งทิศทางที่ได้จากโมดูลเข็มทิศจะความคลาดเคลื่อนประมาณ 2 องศา ทำให้ช่วงที่ตัดสินใจว่าควบคุมตรงไปด้านหน้าต้องเพิ่มขึ้นตามค่าความคลาดเคลื่อน



ภาพที่ 5.3 ช่วงของการตัดสินใจเลือกทิศทาง

ตัวอย่างรหัสเทียม (Pseudo Code) ของฟังก์ชันที่ใช้สำหรับคำนวณทิศทาง มีดังนี้

```
function driveMotor(heading, bearing):
    # heading = direction from compass
    # bearing = azimuth(start, target)

    diff = (bearing - heading) % 360

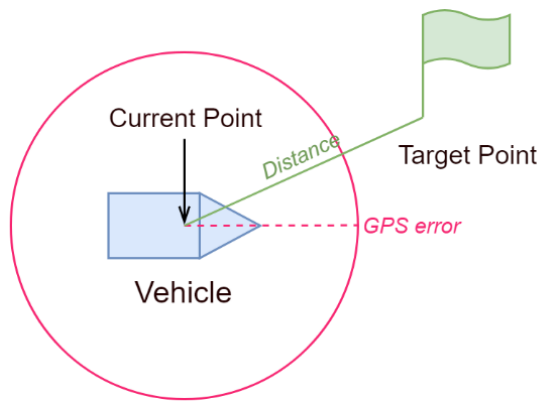
    bot = 360 - compass_error
    mid = 180
    top = 0 + compass_error

    if (bot <= diff <= 360) or (0 <= diff <= top) then
        motor.forward()
    else if (top < diff <= mid) then
        motor.turnRight()
    else if (mid < diff < bot) then
        motor.turnLeft()
    end if
```

3) การตรวจจับตำแหน่งของเรือ

ในส่วนนี้จะใช้สำหรับตรวจจับพิกัดของเรือด้วยโมดูลจีพีเอส ค่าละติจูดและลองจิจูดที่ได้จากโมดูลจีพีเอส จะถูกนำไปคำนวณหาระยะห่างระหว่างตำแหน่งปัจจุบันของเรือและเป้าหมาย เพื่อตัดสินใจให้โมดูลขับเคลื่อนมอเตอร์ควบคุมมอเตอร์ไปด้านหน้าหรือไม่

ตำแหน่งพิกัดจากโมดูลจีพีเอสจะมีความคลาดเคลื่อน 2.5 เมตรโดยประมาณ ทำให้ต้องสร้างรัศมีกว้าง 5 เมตรเพื่อใช้ตรวจจับตำแหน่งเพิ่มขึ้นมา ดังภาพที่ 5.4



ภาพที่ 5.4 การตรวจจับพิกัดของเรือและระยะทางไปยังเป้าหมาย

ตัวอย่างรหัสเทียมของฟังก์ชันที่ใช้สำหรับคำนวณระยะทาง มีดังนี้

```
function distanceBetween(start, target):
    d = euclidean(start, target)

    if (d < gps_err) then
        driveMotor(heading, bearing)
    end if
```

4) การทำงานของระบบกู้เรือ

ส่วนนี้จะทำงานเมื่อแบตเตอรี่ต่ำกว่าที่กำหนดไว้ ค่าความแรงของสัญญาณต่ำหรือเกิดการสูญหายของแพ็กเก็ต (Packet Loss) มากเกินไป ซึ่งปัญหาทั้งหมดที่กล่าวมานี้ ยังไม่ส่งผลกระทบต่อระบบขับเคลื่อนของเรือมากนัก ทำให้ยังสามารถนำเรือกลับมายังจุดเริ่มต้นได้ แต่ในกรณีความเร็วของเรือต่ำกว่าปกติอาจจะมีสาเหตุมาจากการปัญหาทำงานในส่วนของมอเตอร์ได้ เช่น มีวัชพืชเข้ามาติดขัดที่บริเวณใบพัดของเรือ ส่งผลให้จำเป็นต้องหยุดการทำงานของใบพัดเนื่องจากอาจจะเกิดความเสียหายได้

ความถี่การทำงานภายในเรือจะถูกแบ่งออกเป็น 3 ช่วง เนื่องจากแต่ละโมดูลมีโปรโตคอลหรือรูปแบบการส่งข้อมูลที่แตกต่างกัน ได้แก่

- 1) ทุก ๆ 1 มิลลิวินาที จะเป็นการอ่านอักขระจากโมดูลจีพีเอส ซึ่งมีรูปแบบของข้อมูลเป็น เอ็นเอ็มอีโอ โดยหากใช้ความถี่ช่วงนี้ จะทำให้สามารถอ่านข้อมูลแต่ละรอบได้หมดภายใน 1 วินาทีได้
- 2) ทุก ๆ 1 วินาที จะเป็นการนำข้อมูลจากโมดูลจีพีเอสไปถอดรหัส (Decode) เพื่อนำข้อมูลมาใช้ และอ่านข้อมูลจากโมดูลอื่น ๆ ที่เหลือ เช่น โมดูลเข็มทิศ และโวลต์เตจ เซนเซอร์
- 3) ทุก ๆ 5 วินาที จะเป็นการส่งและรับข้อมูลระหว่างเซิร์ฟเวอร์ ซึ่งแต่ละครั้งจะใช้เวลาประมาณ 2 ถึง 5 วินาที จึงจะส่งผ่านแพ็กเก็ตทั้งหมดได้

5.1.2 แอปพลิเคชันสำหรับควบคุมและติดตามสถานะการทำงานของเรือ

แอปพลิเคชันที่ถูกพัฒนาขึ้นสำหรับใช้งานบนระบบปฏิบัติการแอนดรอยด์ ด้วยโปรแกรมแอนดรอยด์สตูดิโอและภาษาจาวา และใช้ Google Maps API สำหรับแสดงแผนที่ของกูเกิลแมพ

แอปพลิเคชันจะใช้วิธีการเรียกใช้เอพีไอบนเซิร์ฟเวอร์ผ่าน HTTP Request Methods ซึ่งจะได้รับข้อมูลจากเซิร์ฟเวอร์ส่วนนี้ในรูปแบบเจสัน สาเหตุที่ใช้วิธีนี้ในการดึงข้อมูลคือแอนดรอยด์แอปพลิเคชันไม่สามารถสร้างการเชื่อมต่อกับฐานข้อมูลบนเซิร์ฟเวอร์โดยตรงได้

5.1.3 เซิร์ฟเวอร์สำหรับเป็นช่องทางสื่อสารระหว่างเรือและแอปพลิเคชัน

เซิร์ฟเวอร์ที่ใช้ในโครงการนี้จะติดตั้งอยู่บนระบบปฏิบัติการเซนต์โอเอสเซว่น ซึ่งจะแยกส่วนที่ติดต่อกับเรือและแอปพลิเคชันออกจากกันแต่ยังคงใช้ฐานข้อมูลร่วมกัน

1) ส่วนที่ใช้สำหรับติดต่อกับเรือ

เซิร์ฟเวอร์ในส่วนนี้มีลักษณะเป็นยูติลิตี้เซิร์ฟเวอร์และใช้ซ็อกเก็ตสำหรับรับส่งข้อมูลระหว่างเรือ ข้อมูลที่ได้รับจากเรือจะถูกเก็บในฐานข้อมูล ภาษาที่ใช้พัฒนาส่วนนี้คือภาษาไพธอน

เมื่อเซิร์ฟเวอร์ได้รับข้อมูลจากเรือ เซิร์ฟเวอร์จะเรียกใช้งานเอพีไอเพื่อในข้อมูลดังกล่าวไปอัปเดตในฐานข้อมูล หลังจากนั้นจึงส่งข้อมูลกลับไปยังเรือโดยข้อมูลที่ส่งให้เรือได้แก่

- คำสั่งจากแอปพลิเคชัน คือ เดินทางไปยังจุดหมายและกลับมายังจุดเริ่มต้น
- ตำแหน่งเป้าหมายหากคำสั่งที่ส่งเป็นคำสั่งให้เดินทางไปยังจุดหมาย

ข้อมูลที่ส่งไปยังเรือมีรูปแบบเหมือนกับข้อที่ 5.1.1 ข้อ 1) ดังนี้

```
"{start, 18.800677_98.955631}"
```

หากไม่มีการส่งตำแหน่งของเป้าหมายไปข้อมูลจะเหลือแค่เพียงคำสั่งเท่านั้น เช่น

```
"{return}"
```

2) ส่วนที่ใช้สำหรับติดต่อกับแอปพลิเคชัน

เซิร์ฟเวอร์ในส่วนนี้จะเป็เว็บไซต์ (Web Service) โดยจะมี API ที่พัฒนาด้วยภาษาพีเอชพี ใช้สำหรับดึงข้อมูลจากฐานข้อมูลและส่งให้กับแอปพลิเคชันที่เรียก ซึ่งจะส่งออกไปในรูปแบบของเจสัน ยกตัวอย่างเช่น

```

{"result": [{
    "status" : "idle",
    "cmd" : "start",
    "curr" : "18.800677_ 98.955631",
    "dest" : "18.782011_ 99.802512",
    "home" : "18.800677_ 98.955631",
    "time" : "2019-04-04 15:37:50"
}]
}

```

5.2 การทดสอบระบบ

หลังจากผ่านขั้นตอนการพัฒนาระบบ ผู้พัฒนาได้ทำการทดสอบระบบซึ่งเป็นหนึ่งในขั้นตอนที่จำเป็นและสำคัญในการดำเนินงาน ซึ่งในการทดสอบระบบผู้พัฒนาได้กำหนดสภาพแวดล้อมและปัจจัยที่มีผลต่อการทดลอง และแบ่งการทดสอบออกเป็นส่วน ๆ เพื่อทดสอบกรณีต่าง ๆ ที่เกิดขึ้นได้อย่างครบถ้วนและแสดงผลในรูปแบบของตาราง

5.2.1 สภาพแวดล้อมที่ทำการทดลอง

สถานที่สำหรับทำการทดลองคือแหล่งน้ำ ณ ศูนย์ธรรมชาติวิทยาดอยสุเทพ ตำบลสุเทพ อำเภอเมือง จังหวัดเชียงใหม่ แหล่งน้ำนี้มีลักษณะเป็นแหล่งน้ำธรรมชาติ มีสิ่งกีดขวางค่อนข้างน้อย ดังภาพที่ 5.5 ทำให้สภาพแวดล้อมนี้เหมาะสมต่อการทดสอบเรือขับเคลื่อนอัตโนมัติ



ภาพที่ 5.5 แหล่งน้ำธรรมชาติ ณ ศูนย์ธรรมชาติวิทยาดอยสุเทพ

5.2.2 ปัจจัยที่มีผลต่อการทดลอง

ปัจจัยที่มีผลต่อการทดลองคือ สิ่งที่ส่งผลให้การทดลองมีความคลาดเคลื่อนไปจากที่ตั้งเป้าไว้ ในการทดลองนี้จะมีปัจจัยจากอุปกรณ์ที่ใช้พัฒนาและสภาพแวดล้อมที่ทำการทดลองดังนี้

ปัจจัยที่มีผลต่อการทดลองที่เกิดจากอุปกรณ์ ได้แก่

- น้ำหนักรวมของโฟมและอุปกรณ์ภายในเรือ
- พลังงานแบตเตอรี่
- ความเร็วรอบการหมุนของมอเตอร์
- ความไม่เสถียรของเครือข่ายอินเทอร์เน็ต

ปัจจัยที่มีผลต่อการทดลองที่เกิดจากสภาพแวดล้อม ได้แก่

- แรงแลมและคลื่นน้ำ ส่งผลให้มีการเปลี่ยนแปลงของตำแหน่งเรือ
- วัชพืช มีโอกาสเข้ามาติดที่ใบพัดเรือส่งผลให้การทำงานของใบพัดขัดข้องได้
- สิ่งกีดขวางบนผิวน้ำและใต้น้ำ

5.2.3 กรณีทดสอบ

ทดสอบกรณีต่าง ๆ ที่เกิดขึ้นกับทุกส่วนของระบบ ได้แก่ เรือ แอปพลิเคชันและเซิร์ฟเวอร์ โดยจะแสดงผลการทดสอบในรูปแบบของตารางซึ่งแสดงดังตารางที่ 5.1 - ตารางที่ 5.4 ดังนี้

ตารางที่ 5.1 การตรวจสอบความพร้อมทำงานของเรือ

ชื่อกรณีทดสอบ		การตรวจสอบความพร้อมทำงานของเรือ		
คำอธิบาย		ทดสอบการรับส่งข้อมูลของแต่ละโมดูลภายในเรือ		
ข้อที่	กรณีทดสอบ	ผลลัพธ์ที่คาดหวัง	ผลการทดสอบ	
			สำเร็จ	ล้มเหลว
1	ไม่สามารถเชื่อมต่อเครือข่ายอินเทอร์เน็ตได้	แสดงผลว่าสามารถไม่สามารถเชื่อมต่อเครือข่ายได้	✓	
2	ไม่มีการเชื่อมต่อของโมดูลจีพีเอส	แสดงผลว่าไม่สามารถรับข้อมูลจากโมดูลจีพีเอสได้	✓	
3	ไม่มีการเชื่อมต่อโมดูลเข็มทิศ	แสดงผลว่าสามารถรับข้อมูลจากโมดูลเข็มทิศได้	✓	
4	ไม่มีการเชื่อมต่อของโมดูลขับมอเตอร์	แสดงผลว่าไม่สามารถสั่งงานโมดูลขับมอเตอร์ได้		✓
5	ไม่มีการเชื่อมต่อของโวลเตจเซนเซอร์	แสดงผลว่าไม่มีการส่งข้อมูลจากโวลเตจเซนเซอร์ได้		✓

ตารางที่ 5.2 การควบคุมมอเตอร์

ชื่อกรณีทดสอบ		การควบคุมมอเตอร์		
คำอธิบาย		ทดสอบควบคุมการหมุนของมอเตอร์ทั้งสองด้าน		
ข้อที่	กรณีทดสอบ	ผลลัพธ์ที่คาดหวัง	ผลการทดสอบ	
			สำเร็จ	ล้มเหลว
1	หมุนมอเตอร์ทั้งสองด้วย ความเร็วเท่ากันทิศตามเข็มนาฬิกา	เรือเคลื่อนที่ตรงไปด้านหน้า	✓	
2	หมุนมอเตอร์ด้านซ้ายช้ากว่า มอเตอร์ด้านขวา	เรือเลี้ยวไปทางด้านซ้าย	✓	
3	หมุนมอเตอร์ด้านซ้ายเร็วกว่า มอเตอร์ด้านขวา	เรือเลี้ยวไปทางด้านขวา	✓	
4	หมุนมอเตอร์ทั้งสองด้วย ความเร็วเท่ากันทิศทวนเข็มนาฬิกา	เรือเคลื่อนที่ถอยไปด้านหลัง	✓	
5	หยุดหมุนมอเตอร์ทั้งสองข้าง	เรือมีความเร็วการเคลื่อนที่ลดลง	✓	

ตารางที่ 5.3 การส่งข้อมูลระหว่างเรือและแอปพลิเคชันผ่านทางเซิร์ฟเวอร์

ชื่อกรณีทดสอบ		การส่งข้อมูลระหว่างเรือและแอปพลิเคชันผ่านทางเซิร์ฟเวอร์		
คำอธิบาย		ทดสอบการส่งข้อมูลระหว่างเรือและแอปพลิเคชันผ่านทางเซิร์ฟเวอร์ว่าเกิดข้อผิดพลาดใด ๆ หรือไม่		
ข้อที่	กรณีทดสอบ	ผลลัพธ์ที่คาดหวัง	ผลการทดสอบ	
			สำเร็จ	ล้มเหลว
1	ส่งแพ็กเก็ตจากเรือไปยังเซิร์ฟเวอร์	เซิร์ฟเวอร์ได้รับแพ็กเก็ตจากเรือและข้อมูลมีความสมบูรณ์	✓	
2	ส่งแพ็กเก็ตจากเซิร์ฟเวอร์ไปยังเรือ	เรือได้รับแพ็กเก็ตจากเซิร์ฟเวอร์และข้อมูลมีความสมบูรณ์	✓	
3	เรียกใช้งานเอฟไอบนเซิร์ฟเวอร์ผ่านแอปพลิเคชัน	สามารถดึงข้อมูลในฐานข้อมูลและส่งกลับไปให้แอปพลิเคชันในรูปแบบของเจสัน ได้	✓	

ตารางที่ 5.4 การทำงานของระบบขับเคลื่อนอัตโนมัติ

ชื่อกรณีทดสอบ		การทำงานของระบบขับเคลื่อนอัตโนมัติ		
คำอธิบาย		ทดสอบการทำงานของระบบขับเคลื่อนอัตโนมัติและการทำงานของระบบกู้เรืออัตโนมัติเมื่อเกิดกรณีฉุกเฉิน		
ข้อที่	กรณีทดสอบ	ผลลัพธ์ที่คาดหวัง	ผลการทดสอบ	
			สำเร็จ	ล้มเหลว
1	เรือได้รับคำสั่งให้เดินทางไปยังจุดหมาย	เรือต้องเคลื่อนที่ไปยังจุดหมาย	✓	
2	เรือถึงจุดหมาย	เรือต้องเปลี่ยนจุดหมายไปที่จุดเริ่มต้นและเดินทางกลับ	✓	
3	แบตเตอรี่ต่ำกว่ากำหนด		✓	
4	ความสัญญาณต่ำกว่ากำหนด		✓	
5	แฟ็กเก็ตสูญหายเกินกำหนด		✓	
6	เรือได้รับคำสั่งให้กลับมาในขณะที่กำลังเดินทาง		✓	
7	เรือกลับมาถึงจุดเริ่มต้น	มอเตอร์ทั้งสองในเรือจะต้องหยุดการทำงาน	✓	

บทที่ 6

บทสรุป

ในบทนี้จะกล่าวถึงการสรุปผลการดำเนินงานในโดยภาพรวม จากการศึกษาและพัฒนาโปรแกรมสำหรับตรวจจับผิดและทิศทางแบบอัตโนมัติ โดยจะประกอบด้วย สรุปผลการดำเนินการ ปัญหาและข้อจำกัดของระบบ และข้อเสนอแนะซึ่งจะเป็นประโยชน์ต่อการนำไปพัฒนาต่อยอระบบที่คล้ายคลึงได้ในอนาคต

6.1 สรุปผลการดำเนินการ

การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาผิดและทิศทางแบบอัตโนมัติ ได้รับข้อกำหนด (Requirement) จาก ผู้ช่วยศาสตราจารย์ ดร.ทินกร กันยานี อาจารย์ภาควิชาเคมี คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่ หลังจากได้รับข้อกำหนดผู้พัฒนาจึงทำการวิเคราะห์ระบบจนได้ข้อสรุปว่า ควรแบ่งย่อยระบบออกเป็น 3 ส่วน ได้แก่ ยานพาหนะสำหรับติดตั้งอุปกรณ์ไอโอที แอปพลิเคชันสำหรับควบคุมและติดตามสถานะการทำงานของยานพาหนะ และเซิร์ฟเวอร์สำหรับสร้างช่องทางสื่อสารระหว่างยานพาหนะและแอปพลิเคชัน

หลังจากการออกแบบระบบ ผู้พัฒนาได้ศึกษาค้นคว้าเทคโนโลยีและอุปกรณ์ที่เหมาะสมสำหรับการใช้เป็นเครื่องมือในการพัฒนาจนได้ข้อสรุปว่า อุปกรณ์ไอโอทีที่เหมาะสมต่อการพัฒนายานพาหนะ คือ

- การที่จะทราบพิกัดบนแผนที่โลกได้จำเป็นต้องใช้โมดูลจีพีเอสสำหรับรับสัญญาณ
- โมดูลจีพีเอสสามารถตรวจหาทิศทางได้แต่จะต้องมีการเคลื่อนที่เกิดขึ้นก่อน ทำให้เกิดความล่าช้า จึงมีการใช้โมดูลเข็มทิศเข้ามาตรวจหาทิศทางแทน
- บอร์ดอาดุยโนเมกะซึ่งมีหน่วยความจำที่สูง ทำให้สามารถรองรับการทำงานของบอร์ดเสริมเอไอเอสเอ็นบีไอโอทีและโมดูลจีพีเอสพร้อมกันได้

สำหรับเทคโนโลยีที่เหมาะสมต่อการพัฒนาระบบ ได้แก่

- โปรแกรมอาดุยโนไอดีอี เป็นโปรแกรมที่สามารถอัปโหลดไฟล์ภาษาซีพลัสพลัสลงบนไมโครคอนโทรลเลอร์ได้
- โปรแกรมแอนดรอยด์สตูดิโอ เป็นโปรแกรมที่สามารถพัฒนาแอปพลิเคชันสำหรับระบบปฏิบัติการแอนดรอยด์ได้ ด้วยภาษาจาวา
- ภาษาไพธอน เป็นภาษาที่มีวากยสัมพันธ์ที่เข้าใจง่าย และยังมีไลบรารีรองรับการทำบริการต่าง ๆ ทำให้สามารถนำมาพัฒนาเซิร์ฟเวอร์ได้ง่าย

หลังจากกำหนดเทคโนโลยีและอุปกรณ์สำหรับพัฒนาระบบ ผู้พัฒนาได้ศึกษาและทดลองใช้งานอุปกรณ์อิเล็กทรอนิกส์ร่วมกับออกแบบกลไกการทำงานของเพลามอเตอร์สำหรับเรือ ก่อนที่จะลงมือพัฒนาเรือต้นแบบด้วยกล่องโฟมพร้อมกับระบบขับเคลื่อนอัตโนมัติ แอนดรอยด์แอปพลิเคชัน และเซิร์ฟเวอร์ตามลำดับ

ผลจากการทดสอบระบบทั้งหมด พบว่า

- โมดูลเซ็นเซอร์และโมดูลจีพีเอสมีความคลาดเคลื่อนเล็กน้อย ซึ่งไม่มีผลต่อการทำงานของระบบขับเคลื่อนมากนัก
- ระบบขับเคลื่อนอัตโนมัติของเรือทำงานได้อย่างถูกต้อง
- แอปพลิเคชันสามารถควบคุมและติดตามสถานะของเรือได้อย่างถูกต้อง
- เซิร์ฟเวอร์รับและส่งข้อมูลให้กับเรือและแอปพลิเคชันได้อย่างถูกต้อง

6.2 ปัญหาและข้อจำกัด

ปัญหาหลักของระบบคือการรับและส่งข้อมูลด้วยเครือข่ายอินเทอร์เน็ตที่ พบว่ายังมีความไม่เสถียรของเครือข่าย เนื่องจากบางครั้งการเชื่อมต่อถูกตัดขาดแม้ว่าจะอยู่ในพื้นที่ที่มีสัญญาณ

6.3 ข้อเสนอแนะ

การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัตินี้สามารถนำไปพัฒนาต่อยอดได้หลายด้าน ผู้พัฒนาจึงมีข้อเสนอแนะเกี่ยวกับการพัฒนาระบบให้มีประสิทธิภาพมากขึ้น ดังนี้

- 1) พัฒนาระบบตรวจจับสิ่งกีดขวางเพิ่ม ด้วยการประยุกต์ใช้การประมวลผลภาพ (Image Processing) ร่วมกับคอมพิวเตอร์วิทัศน์ (Computer Vision) หรือ การประยุกต์ใช้โซนาร์เซนเซอร์ (Sonar Sensor) ร่วมกับเซอร์โวมอเตอร์ (Servo Motor) เพื่อพัฒนาเป็นอุปกรณ์ที่มีลักษณะเหมือนเรดาร์ (Radar) สามารถนำไปใช้งานสำหรับตรวจจับวัตถุได้
- 2) พัฒนาเซิร์ฟเวอร์และแอปพลิเคชันที่สามารถจัดการบริหารข้อมูลจากเรือหลาย ๆ ลำพร้อมกันได้ ซึ่งจะเป็นการขยายขอบเขตของระบบให้สามารถนำไปประยุกต์ใช้กับงานที่หลากหลายได้มากขึ้น
- 3) ฟังก์ชันกู้เรืออัตโนมัติในส่วนของการตรวจสอบปริมาณแบตเตอรี่สามารถปรับปรุงเป็น การคำนวณปริมาณพลังงานแบตเตอรี่กับระยะทาง เพื่อหาว่าปริมาณพลังงานเพียงพอที่จะเดินทาง

เอกสารอ้างอิง

- [1] ปรีชา กอเจริญ, "เทคโนโลยีการสื่อสารไร้สายสำหรับอินเทอร์เน็ตของสรรพสิ่ง," *NBTC Journal 2*, pp. 268-287, 2017.
- [2] Wang, Y-P. Eric, et al., "A Primer on 3GPP Narrowband Internet of Things (NB-IoT)," *IEEE Communications Magazine vol. 55 no. 3*, pp. 117-123, 2017.
- [3] Azman, Mohammad, et al., "Cloud of Things: Integrating Internet of Things and Cloud Computing and the Issues Involved," *International Bhurban Conference on Applied Science & Technology (IBCAST)*, pp. 414-419, 2014.
- [4] Norton J.H., "Navstar Global Positioning System," *The International Hydrographic Review*, p. 59(1), 2015.
- [5] Witchayangkoon Boonsap, Dumrongchai Puttipol, "Computational Linearized Least Squares for 2D Positioning Using Mathcad Programming," *KMUTT Research & Development Journal 32*, p. 31, 2009.
- [6] Great Britain: Ministry of Defence, in *Admiralty Manual of Navigation: General Navigation, Coastal Navigation and Pilotage v. 1*, 1987, p. 97.
- [7] Arduino.cc, "Arduino Mega 2560 Rev3," 2018. [Online]. Available: <https://store.arduino.cc/usa/arduino-mega-2560-rev3>. [Accessed 9 December 2018].
- [8] Artiya, "ตัวอย่างวิธีรับส่งข้อมูลผ่าน AIS NB-IoT Shield กับ UDP Server," 17 May 2018. [Online]. Available: <https://medium.com/@artiya4u/ais-nb-iot-udp-connect-f86f803b61a>. [Accessed 10 December 2018].
- [9] Quectel, Quectel BC95 Compact NB-IoT Module with Ultra-low Power Consumption, 2017.
- [10] Handson Technology, "L298N Dual H-Bridge Motor Driver," 2017. [Online]. Available: <http://www.handsontec.com/dataspecs/L298N%20Motor%20Driver.pdf>. [Accessed 11 December 2018]

- [11] Ublox, "NEO-M8 u-blox M8 concurrent GNSS modules Data Sheet," 2015. [Online]. Available: https://www.u-blox.com/sites/default/files/NEO-M8_DataSheet_%28UBX-13003366%29.pdf. [Accessed 10 December 2018].
- [12] Honeywell, "3-Axis Digital Compass IC HMC5883L," 2013. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/HMC5883L_3-Axis_Digital_Compass_IC.pdf. [Accessed 11 December 2018].
- [13] Henry's Bench, "Arduino 25V Voltage Sensor Module User Manual," [Online]. Available: <http://henrysbench.capnfatz.com/henrys-bench/arduino-voltage-measurements/arduino-25v-voltage-sensor-module-user-manual>. [Accessed 11 December 2018].

ภาคผนวก

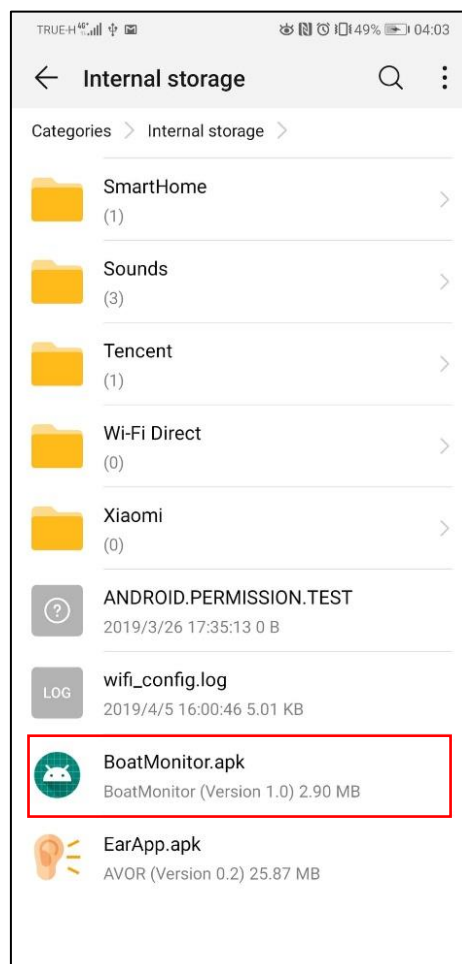
ภาคผนวก ก

คู่มือการติดตั้งระบบ

การติดตั้งซอฟต์แวร์เพื่อเตรียมความพร้อมสำหรับการใช้งานจะมีส่วนที่สำคัญ 2 ส่วน คือ แอปพลิเคชันสำหรับควบคุมและติดตามสถานะของเรือและเซิร์ฟเวอร์สำหรับรับส่งข้อมูลระหว่างเรือและแอปพลิเคชัน

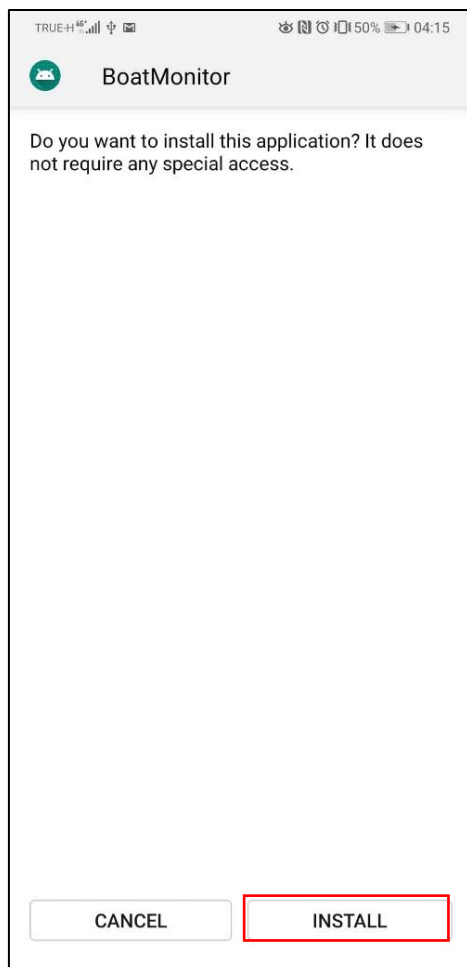
ก.1 การติดตั้งแอปพลิเคชัน

ในขั้นตอนแรก ผู้ใช้จะต้องมีไฟล์ของแอปพลิเคชันที่อยู่ในลักษณะของไฟล์ .apk ซึ่งเป็นไฟล์ที่ใช้สำหรับติดตั้งแอปพลิเคชันลงบนระบบปฏิบัติการแอนดรอยด์ ดังภาพที่ ก.1



ภาพที่ ก.1 ไฟล์ .apk ของแอปพลิเคชัน

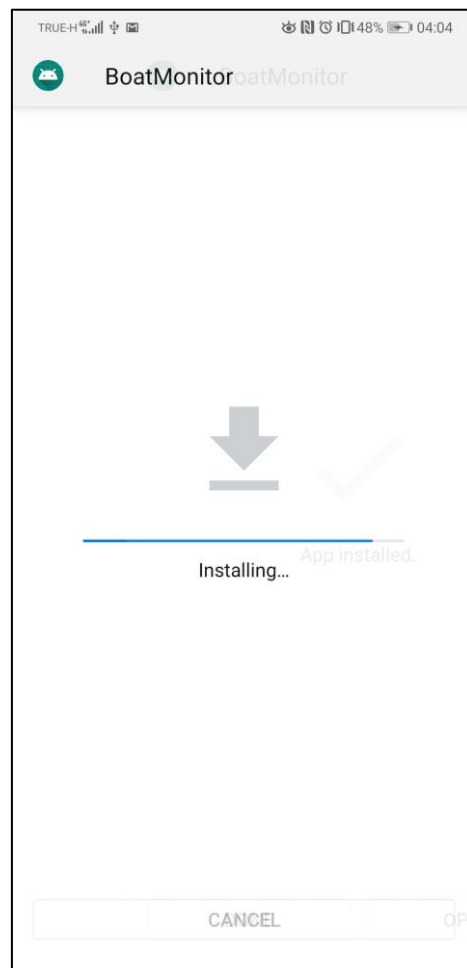
ขั้นตอนต่อไป ให้ทำการเปิดไฟล์ .apk โดยจะมีการสอบถามว่าจะติดตั้งแอปพลิเคชันดังกล่าวหรือไม่ ดังภาพที่ ก.2 สำหรับแอปพลิเคชันนี้จะไม่มีการร้องขอสิทธิการเข้าถึงข้อมูลใด ๆ เพิ่มเติม



ภาพที่ ก.2 สอบถามการติดตั้งแอปพลิเคชัน

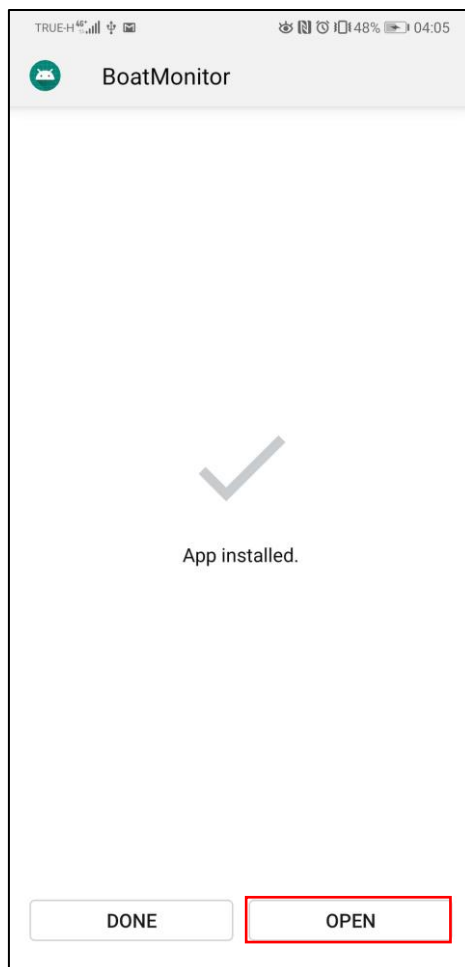
จากภาพที่ ก.2 ให้กดปุ่มติดตั้ง (Install) เพื่อดำเนินการติดตั้งแอปพลิเคชัน

หลังจากที่กดปุ่มติดตั้งแอปพลิเคชัน ให้ทำการรอนจนกว่าแอปพลิเคชันจะติดตั้งเสร็จ



ภาพที่ ก.3 กำลังติดตั้งแอปพลิเคชัน

ขั้นตอนสุดท้าย หลังจากติดตั้งแอปพลิเคชันเสร็จ จะแสดงหน้าจอว่าแอปพลิเคชันถูกติดตั้งแล้ว ให้ทำการกดปุ่มเปิด (Open) เพื่อเปิดใช้งานแอปพลิเคชัน ดังภาพที่ ก.4



ภาพที่ ก.4 ติดตั้งแอปพลิเคชันสำเร็จ

ก.2 การติดตั้งเซิร์ฟเวอร์

การติดตั้งเซิร์ฟเวอร์จะแบ่งไฟล์ออกเป็น 3 ส่วนคือ ไฟล์ยูติลิตีเซิร์ฟเวอร์ ไฟล์เอพีไอ และไฟล์ฐานข้อมูล โดยมีวิธีการติดตั้งดังนี้

1) ยูติลิตีเซิร์ฟเวอร์

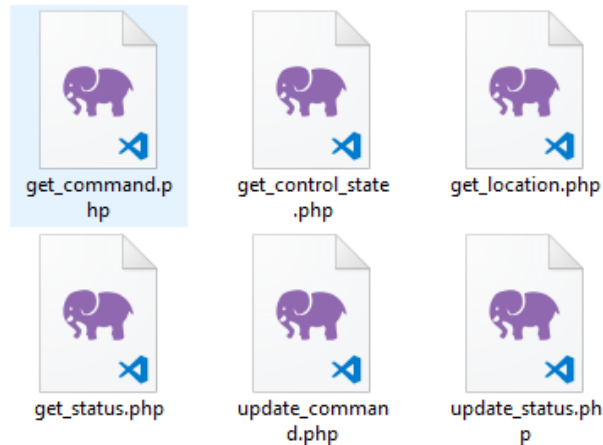
จะต้องมีไฟล์ของเซิร์ฟเวอร์นามสกุล .py หรือไฟล์ไพธอนและเปิดการทำงานดังภาพที่ ก.5

```
$ python server.py
3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit (Intel)]
Socket Start...
Server Start...
Incoming-->{idle,18.800677_98.955631,98,-87,etc.}
Sending-->{no command}
```

ภาพที่ ก.5 การทำงานของยูติลิตีเซิร์ฟเวอร์

2) เอพีไอ

นำไฟล์เอพีไอ ซึ่งอยู่ในลักษณะของไฟล์นามสกุล .php ไปวางไว้บน HTTP Server สำหรับเซิร์ฟเวอร์โอเอสเซว่นจะอยู่ที่ไดเรกทอรี /var/www/html/



ภาพที่ ก.6 ไฟล์เอพีไอ

3) ฐานข้อมูล

นำเข้าไฟล์ฐานข้อมูลนามสกุล .sql ไปยัง ฐานข้อมูลที่ชื่อ autopilot_boat โดยสามารถเปิดใช้งานเป็น localhost ได้ แต่ยูติลิตีเซิร์ฟเวอร์ และเอพีไอต้องอยู่บนเซิร์ฟเวอร์เครื่องเดียวกัน

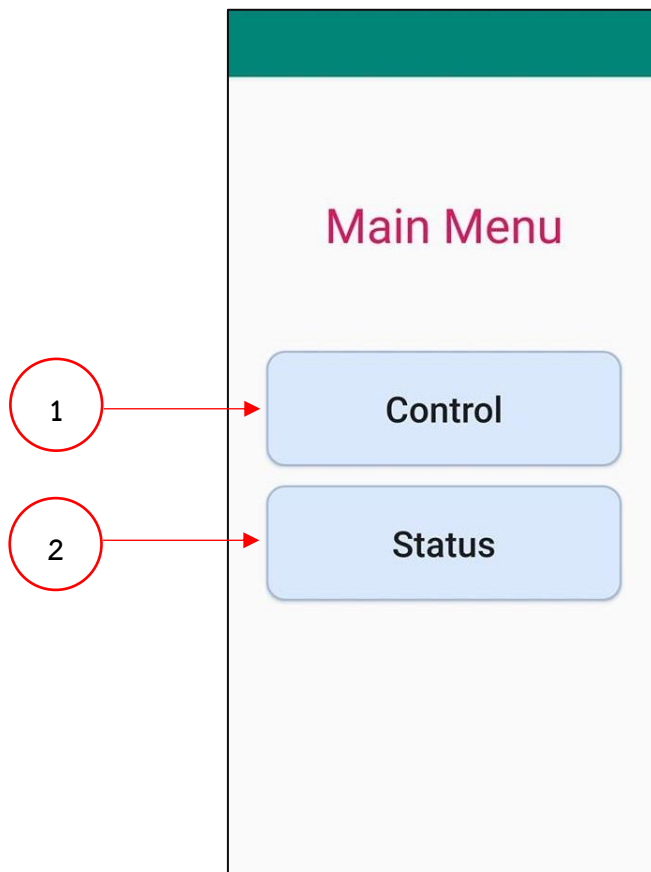
ภาคผนวก ข

คู่มือการใช้งานระบบ

การพัฒนาแบบจำลองสำหรับระบบควบคุมเรือโดยการตรวจหาพิกัดและทิศทางแบบอัตโนมัติ ผู้ใช้จะต้องควบคุมการทำงานและติดตามสถานะของเรือผ่านแอนดรอยด์แอปพลิเคชัน โดยมีรายละเอียดการใช้งานดังนี้

ข.1 หน้าจอหลักของแอปพลิเคชัน

หน้าจอหลัก เป็นหน้าจอแรกที่แสดงเมื่อเข้าสู่แอปพลิเคชัน สามารถแสดงตัวอย่างหน้าจอได้ดังภาพที่ ข.1



ภาพที่ ข.1 หน้าจอหลัก

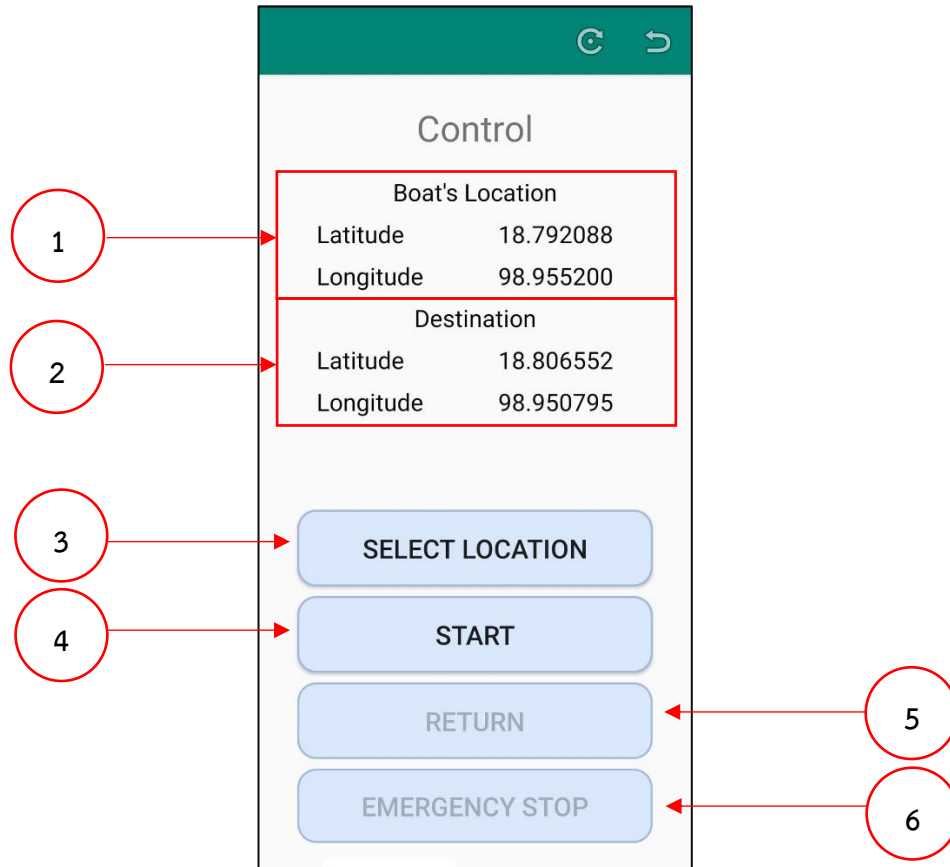
จากภาพที่ ข.1 สามารถอธิบายรายละเอียดของจอภาพได้ดังนี้

ส่วนที่ 1 ปุ่มกดเพื่อไปยังหน้าจอควบคุมเรือ

ส่วนที่ 2 ปุ่มกดเพื่อไปยังหน้าจอแสดงสถานะของเรือ

ข.2 หน้าจอบคุมเรือ

หน้าจอบคุมเรือ เป็นหน้าจอสำหรับควบคุมเรือซึ่งสามารถเลือกตำแหน่งจุดหมายให้กับเรือได้ สามารถแสดงตัวอย่างหน้าจอได้ดังภาพที่ ข.2



ภาพที่ ข.2 หน้าจอบคุมเรือ

จากภาพที่ ข.2 สามารถอธิบายรายละเอียดของจอภาพได้ดังนี้

ส่วนที่ 1 ตำแหน่งปัจจุบันของเรือที่แสดงในรูปแบบพิกัดละติจูดและลองจิจูด

ส่วนที่ 2 ตำแหน่งของเป้าหมายที่แสดงในรูปแบบพิกัดละติจูดและลองจิจูด สามารถเลือกได้จากส่วนที่ 3

ส่วนที่ 3 ปุ่มกดเพื่อไปยังหน้าจอเลือกตำแหน่งบนกูเกิลแมพ

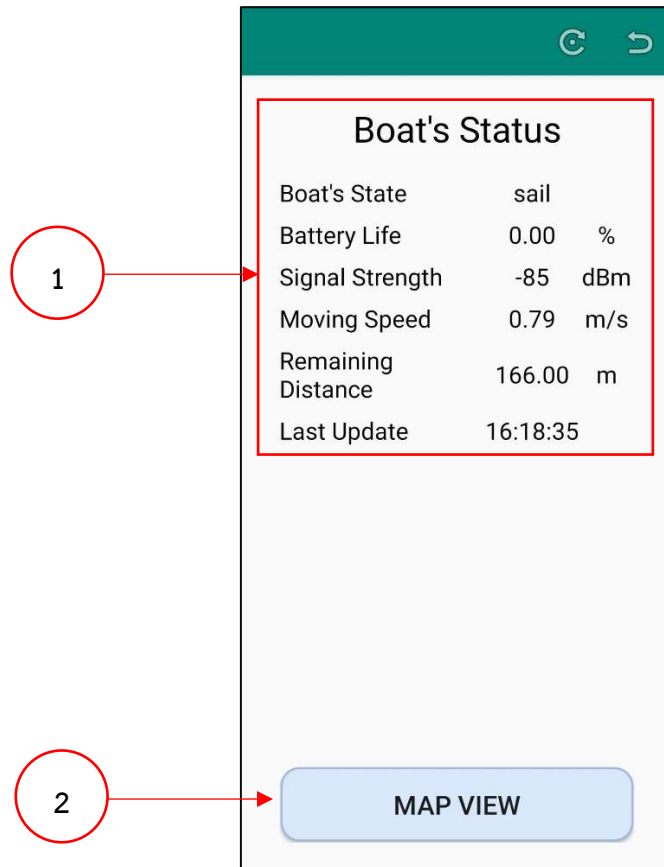
ส่วนที่ 4 ปุ่มกดเพื่อสั่งให้เรือเริ่มทำงาน หากกดปุ่มนี้จะทำการส่งคำสั่งเริ่มการทำงานและพิกัดเป้าหมายไปยังเรือผ่านทางเซิร์ฟเวอร์

ส่วนที่ 5 ปุ่มกดเพื่อสั่งให้เรือกลับมาที่จุดเริ่มต้น จะเปิดให้ใช้งานได้หลังจากกดปุ่มในส่วนที่ 4

ส่วนที่ 6 ปุ่มกดเพื่อสั่งให้เรือหยุด จะเปิดให้ใช้งานได้หลังจากกดปุ่มในส่วนที่ 4

ข.3 หน้าจอติดตามสถานะการทำงานของเรือ

หน้าจอติดตามสถานะเป็นหน้าจอที่แสดงสถานะต่าง ๆ ของเรือ เช่น สถานะปัจจุบันของเรือ พลังงานที่เหลืออยู่ในแบตเตอรี่ ความแรงของสัญญาณเอ็นบีไอโอที ความเร็วการเคลื่อนที่ ระยะทางที่เหลืออยู่ สามารถแสดงตัวอย่างหน้าจอได้ดังภาพที่ ข.3



ภาพที่ ข.3 หน้าจอแสดงสถานะของเรือ

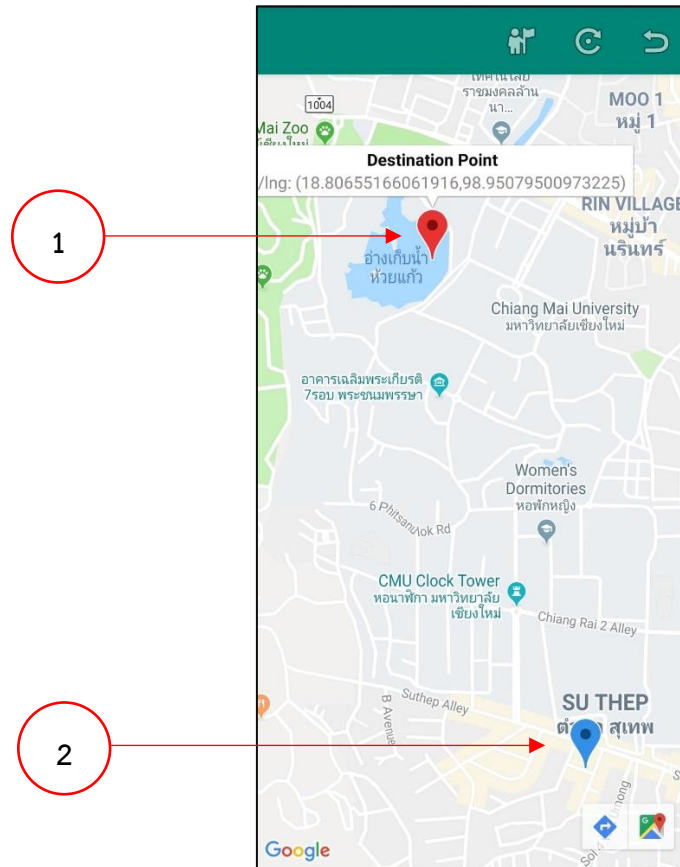
จากภาพที่ ข.3 สามารถอธิบายรายละเอียดของจอภาพได้ดังนี้

ส่วนที่ 1 สถานะต่าง ๆ ของเรือ

ส่วนที่ 2 ปุ่มกดเพื่อไปยังหน้าจอแสดงตำแหน่งของเรือและเป้าหมายบนกูเกิลแมพ

ข.4 หน้าจอเกิลแมพ

หน้าจอเกิลแมพ ใช้สำหรับแสดงตำแหน่งปัจจุบันของเรือ หรือเลือกตำแหน่งจุดหมายให้กับเรือ สามารถแสดงตัวอย่างหน้าจอได้ดังภาพที่ ข.4



ภาพที่ ข.4 หน้าจอสำหรับเลือกตำแหน่งบนแผนที่กูเกิลแมพ

จากภาพที่ ข.4 สามารถอธิบายรายละเอียดของจอภาพได้ดังนี้

ส่วนที่ 1 ตำแหน่งที่ถูกเลือกบนแผนที่

ส่วนที่ 2 ตำแหน่งปัจจุบันของเรือ