ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

หุ่นยนต์ตรวจจับและจัดการวัตถุ

OBJECT DETECTION AND MANIPULATION ROBOT

H004761

โดย

พชร ธนบุญญาสกุล

สุเมธ ลี้

อาจารย์ที่ปรึกษา

ดร.โอฬาร วงศ์วิรัตน์

อาจารย์ที่ปรึกษาร่วม

อาจารย์ อนันตพัฒน์ อนันตชัย

เลขหมู่.....
เลขทะเบียน 04761
วัน,เดือน,ปี - 7 ต.ค. 2551

.b.....
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

สาขาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์ตรวจจับและจัดการวัตถุ

OBJECT DETECTION AND MANIPULATION ROBOT

พชร ธนบุญญาสกุล

สุเมธ ลี้

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

สาขาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OBJECT DETECTION AND MANIPULATION ROBOT

PODCHARA THANABOONYASAKUL

SUMATE LEE

A PROJECT SUMMITTED IN PARTIAL FULLFILMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2/2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2008

FACULTY OF INFORMATION TECHNOLOGY

KING MONKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบรับรองปริญญานิพนธ์ ประจำปีการศึกษา 2550

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง หุ่นยนต์ตรวจจับ และจัดการวัตถุ

OBJECT DETECTION AND MANIPULATION ROBOT

ผู้จัดทำ

1. นาย พชร	**ธนบุญญาสกุล**	**รหัสประจำตัว 47070041**
2. นาย สุเมธ	**ลี้**	**รหัสประจำตัว 47070054**

.....**อาจารย์ที่ปรึกษา**

(ดร. โอฬาร วงศ์วิรัตน์)

.....**อาจารย์ที่ปรึกษาร่วม**

(อาจารย์ อนันตพัฒน์ อนันตชัย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงาน	หุ่นยนต์ตรวจจับและจัดการวัตถุ		
นักศึกษา	นาย พชร	ธนบุญญาสกุล	รหัสนักศึกษา 47070041
	นาย สุเมธ	ลี้	รหัสนักศึกษา 47070054
รหัสประจำตัว	47070041		
	47070054		
ปริญญา	วิทยาศาสตรบัณฑิต		
สาขาวิชา	เทคโนโลยีสารสนเทศ		
ปีการศึกษา	2550		
อาจารย์ผู้ควบคุมโครงงาน	ดร. โอพาร วงศ์วิรัตน์		
อาจารย์ผู้ควบคุมโครงงานร่วม	อ.อนันตพัฒน์ อนันตชัย		

บทคัดย่อ

ปัจจุบันเทคโนโลยีระบบฝังตัว (Embedded System) ได้รับความนิยมและมีบทบาทสำคัญในชีวิตประจำวันมากขึ้น แต่การพัฒนางานด้านระบบฝังตัวส่วนใหญ่ยังต้องพึ่งพาเทคโนโลยีจากต่างประเทศเป็นหลัก ซึ่งแนวทางในการพัฒนางานด้านระบบฝังตัวอย่างมีประสิทธิภาพในประเทศไทยนั้นยังไม่ได้รับการส่งเสริมเท่าที่ควร ดังนั้นหากการพัฒนางานด้านระบบฝังตัวอย่างเป็นระบบได้รับการนำมาประยุกต์ใช้ ก็จะทำให้เป็นจุดเริ่มต้นในการสร้างแอพพลิเคชั่น (Application) ที่มีประสิทธิภาพได้

โครงงานนี้จึงเป็นการประยุกต์แนวทางการพัฒนาแอปพลิเคชั่นอย่างเป็นระบบให้กับระบบฝังตัว รวมทั้งเป็นการประยุกต์ใช้เครื่องมือ (Tools) สมัยใหม่ เช่น ยูเอ็มแอล (UML: Unified Model Language) และแนวคิดการออกแบบเชิงวัตถุ (Object Oriented Programming) เป็นต้น ในการพัฒนาซอฟท์แวร์ด้วยโปรแกรมภาษาจาวา (Java) และใช้อุปกรณ์ชุดทดลองเลโก้ (Lego) รุ่นเอ็นเอ็กซ์ที (NXT) เป็นฮาร์ดแวร์สำหรับการพัฒนาแอพพลิเคชั่นหุ่นยนต์ตรวจจับและจัดการวัตถุขึ้น โดยมีหลักการทำงานคือ หุ่นยนต์สามารถตรวจจับลูกบอลสีโดยใช้อุปกรณ์เซ็นเซอร์แสง (Light Sensor) เป็นตัวแยกแยะ และใช้อุปกรณ์เซ็นเซอร์คลื่นอัลตราโซนิค (Ultrasonic Sensor) เป็นตัววัดระยะทางจากลูกบอลเป้าหมาย เพื่อที่หุ่นยนต์จะสามารถเคลื่อนที่ไปจัดการกับลูกบอลเป้าหมายตามเงื่อนไขและข้อกำหนดได้อย่างถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title	Object Detection and Manipulation Robot
Student	Mr. Podchara Thanabooyasakul Student ID. 47070041
	Mr. Sumate Lee Student ID. 47070054
Degree	Bachelor of Science
Programme	Information Technology
Academic Year	2007
Advisor	Dr. Olarn Wongwirat
Co-Advisor	Mr.Anuntapat Anuntachai

ABSTRACT

Presently, embedded system technology is being favor and plays a major role in our daily life. Nevertheless, the development of embedded system still acquires technology from a foreign country mainly. In Thailand, the efficient approach in embedded system development has not been encouraged adequately. Therefore, if the systematic development process can be applied into the embedded system properly, it will be a starting point to create application efficiently.

This project is the application of system development approach into the embedded system and the deployment of modern tools, such as Unified Model Language (UML) and Object Oriented Programming (OOP), to develop application software of embedded system by using Java programming language. The project also employs Lego NXT as the hardware for development of object detection and manipulation robot. Basically, the robot can detect and classify the color of ball via a light sensor. Furthermore, an ultrasonic sensor is used to detect the distance of the target ball. As the result, the robot can move to manipulate the target ball correctly according to the assign condition.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์สำเร็จลุล่วงได้อย่างดี ด้วยความกรุณาจากอาจารย์ที่ปรึกษา อ.ดร. โอฬาร วงศ์วิรัตน์ และ อาจารย์ที่ปรึกษาร่วม อ. อนันตพัฒน์ อนันตชัย คอยให้ความช่วยเหลือ แนะแนวทางต่างๆ และช่วยแก้ไขปัญหาต่างๆที่เกิดขึ้น รวมถึงแนะแนวคิดต่างๆที่ได้ถ่ายทอดมา ทำให้เกิดแนวทางในการทำปริญญานิพนธ์โครงงานนี้ ตลอดทุกช่วงเวลา ทางคณะผู้จัดทำ ขอกราบขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณ คณาจารย์คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารฯลาดกระบัง ทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้ต่าง รวมทั้งได้แนะแนวคิดต่างๆ ในการทำปริญญานิพนธ์ให้สำเร็จลุล่วงไปด้วยดี

ขอขอบคุณ เจ้าหน้าที่ฝ่ายทะเบียนทุกๆท่านที่ได้แนะนำและอำอวยความสะดวกในการดำเนินงานต่างๆที่เกี่ยวข้องกับการทำปริญญานิพนธ์ ทั้งทางด้านโครงงานและด้านเอกสารต่างๆ

ขอขอบคุณ บิดา มารดา ที่คอยสนับสนุนและเป็นกำลังใจให้ตลอดในการทำปริญญานิพนธ์โครงงานนี้

ขอขอบคุณและขอบใจ เพื่อนๆ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารฯลาดกระบัง รุ่นที่ 2 ที่คอยให้กำลังใจและความช่วยเหลือตลอดมา

พชร ธนบุญญาสกุล

สุเมธ ลี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VII

สารบัญรูปภาพ.....IX

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและความสำคัญของปัญหา.....1

1.2 วัตถุประสงค์ของการศึกษา.....1

1.3 ทฤษฎีหรือแนวคิดที่ใช้ในโครงงาน.....2

1.4 ขอบเขตของโครงงาน.....2

1.5 ขั้นตอนของการศึกษา.....3

1.6 ประโยชน์ที่คาดว่าจะได้รับ.....3

บทที่ 2 ทฤษฎีบท.....4

2.1 ระบบเทคโนโลยีสมองกลฝังตัว.....4

2.2 องค์ประกอบฮาร์ดแวร์.....4

2.2.1 คุณลักษณะเฉพาะของเทคโนโลยีสมองกลฝังตัว.....4

2.2.2 สถาปัตยกรรมของระบบสมองกลฝังตัว.....5

2.2.2.1 เอ็มพียู.....6

2.2.2.2 บัส.....6

2.2.2.3 หน่วยความจำ.....7

2.2.2.4 อินพุตเอาต์พุตพื้นฐาน.....7

2.2.3 เทคโนโลยีที่เกี่ยวข้องกับหน่วยประมวลผล.....8

2.2.3.1 เทคโนโลยีอินเทอร์รัพต์.....8

2.2.3.2 เทคโนโลยีดีเอ็มเอ.....9

2.2.3.3 เทคโนโลยีหน่วยความจำแคช.....9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.3 องค์ประกอบซอฟต์แวร์.....10
2.3.1 ระบบปฏิบัติการ.....10
2.3.1.1 เคอร์เนล.....11
2.3.1.2 ดีไวซ์ไดรเวอร์.....11
2.3.1.3 มิดเดิลแวร์.....11
2.3.2 แอพพลิเคชั่นโปรแกรม.....11
2.4 เอ็นเอ็กซ์ทีฮาร์ดแวร์.....12
2.4.1 สมองกลฝังตัวเอ็นเอ็กซ์ที.....12
2.4.1.1 ส่วนประกอบภายนอก.....12
2.4.1.2 ส่วนประกอบภายใน.....14
2.4.1.3 สถาปัตยกรรมภายในของสมองกลฝังตัวเอ็นเอ็กซ์ที.....14
2.4.1.4 เอาต์พุตพอร์ต.....15
2.4.1.5 อินพุตพอร์ต.....16
2.4.2 อุปกรณ์เสริม.....17
2.4.2.1 อุปกรณ์เซ็นเซอร์ด้วยแสง.....17
2.4.2.2 อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่.....18
2.4.2.3 กลไกมอเตอร์.....19
2.5 เอ็นเอ็กซ์ทีซอฟต์แวร์.....20
2.5.1 เอ็นเอ็กซ์เจ.....20
2.5.2 อีคลิพซ์.....20
2.5.3 สิ่งแวดล้อมการพัฒนาแบบครอสส์.....20
2.5.4 โครงสร้างระบบปฏิบัติการของเอ็นเอ็กซ์ที.....21
2.5.4.1 ฟังก์ชันการทำงานบนระบบปฏิบัติการของเอ็นเอ็กซ์ที.....21
1. ฟังก์ชันแฟ้มข้อมูล.....22
2. ฟังก์ชันบลูทูธ.....23
3. ฟังก์ชันระบบ.....25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.6 แนวทางในการพัฒนาระบบ..... 25
2.6.1 วงจรการพัฒนาระบบ 25
2.6.2 แบบจำลองการพัฒนาระบบ.....27
2.6.2.1 แบบจำลองเพิ่มขึ้น 27
2.6.3 ยูเอ็มแอล 27
2.6.3.1 แผนภาพยูสเคส..... 28
2.6.3.2 แผนภาพคลาส.....28
2.6.3.3 แผนภาพวัตถุ.....29
2.6.3.4 แผนภาพลำดับเหตุการณ์.....30
2.6.3.5 แผนภาพปฏิสัมพันธ์..... 30
2.6.3.6 แผนภาพสถานะ.....31
2.6.3.7 แผนภาพกิจกรรม.....31

บทที่ 3 การพัฒนาระบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ.....35
3.1 ความต้องการของระบบ.....35
3.1.1 หน้าที่ของระบบ.....35
3.1.2 ข้อกำหนดของการทดสอบ..... 36
3.2 การออกแบบระบบ.....36
3.2.1 แผนภาพยูสเคส 37
3.2.2 ยูสเคสเดสคริพชั่น..... 37
3.2.3 แผนภาพกิจกรรม..... 42
3.2.4 แผนภาพคลาส.....46
3.2.5 แผนภาพลำดับ.....48
3.2.6 แผนภาพปฏิสัมพันธ์.....54
3.2.7 แผนภาพสถานะ..... 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.3 การออกแบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ.....57

3.2.1 การออกแบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ.....57

3.2.2 การประกอบหุ่นยนต์ตรวจจับ และจัดการวัตถุ.....59

บทที่ 4 การทดลอง และผลการทดลอง.....65

4.1 การทดลองระบบตรวจสอบเส้นขอบสนาม.....65

4.2 การทดลองระบบการค้นหา และเข้าถึงลูกบอลในสนาม.....68

4.3 การทดลองระบบการค้นหา และจัดการวัตถุ.....71

บทที่ 5 สรุปโครงงาน และข้อเสนอแนะ.....79

5.1 สรุปโครงงาน.....79

5.2 ปัญหา และอุปสรรค.....79

5.2.1 ปัญหาทางด้านฮาร์ดแวร์.....79

5.2.2 ปัญหาทางด้านซอฟต์แวร์.....80

5.2.3 ปัญหาทางด้านการพัฒนาโปรแกรม.....80

5.3 ข้อเสนอแนะ.....80

บรรณานุกรม.....81

ภาคผนวก ก. 82

ภาคผนวก ข. 89

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

3.1 Use Case Description – Push Object.....37

3.2 Use Case Description – Separate Object.....38

3.3 Use Case Description – Search Object.....39

3.4 Use Case Description – Movement.....40

3.5 Use Case Description – Line Detection.....41

3.6 การเชื่อมโยงระหว่างความต้องกับการออกแบบหุ่นยนต์..... 58

4.1 การทดลองระบบตรวจสอบเส้นขอบสนาม..... 66

4.2 การทดลองระบบการค้นหา และเข้าถึงลูกบอลในสนาม.....68

4.3 การทดลองระบบการค้นหา และจัดการวัตถุ..... 71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

2.1 โครงสร้างโดยสังเขปของระบบสมองกลฝังตัว.....5

2.2 สถาปัตยกรรมพื้นฐานของระบบสมองกลฝังตัว.....5

2.3 การทำงานพื้นฐานของเอ็มพียู.....6

2.4 โครงสร้างพื้นฐานหน่วยความจำ.....7

2.5 การแปลงข้อมูลดิจิตอลเป็นอนาล็อก และสัญญาณอนาล็อกเป็นดิจิตอล.....8

2.6 โฟลว์ของการประมลผลการร้องขออินเทอร์รัพต์.....9

2.7 การทำงานของหน่วยความจำแคช.....10

2.8 อุปกรณ์เอ็นเอ็กซ์ที.....12

2.9 หมายเลข 1 คือปุ่มสำหรับเปิดโหมดออนหรือเข้าหรือการดำเนินงาน.....13

2.10 หมายเลข 2 คือปุ่มสำหรับสำหรับเลื่อนซ้าย และขวาบนรายการเลือกของเอ็นเอ็กซ์.....13

2.11 หมายเลข 3 คือปุ่มสำหรับเคลียร์หรือย้อนกลับไป.....13

2.12 สถาปัตยกรรมภายในของสมองกลฝังตัวเอ็นเอ็กซ์ที.....14

2.13 เอาต์พุตพอร์ต.....15

2.14 อินพุตพอร์ต.....16

2.15 อุปกรณ์เซ็นเซอร์ด้วยแสง.....17

2.16 ลักษณะการมองเห็นของอุปกรณ์เซ็นเซอร์ด้วยแสง.....17

2.17 ทดสอบความสามารถของอุปกรณ์เซ็นเซอร์ด้วย.....17

2.18 อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่.....18

2.19 ทดสอบความสามารถของอุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่.....18

2.20 มอเตอร์สำหรับควบคุมกลไกต่างๆ.....19

2.21 ทดสอบความสามารถของกลไกมอเตอร์.....19

2.22 รายละเอียดรายการเลือกต่างๆ บนหน่วยแสดงผลของเอ็นเอ็กซ์ที.....21

2.23 แสดงรายละเอียดข้อมูลของฟังก์ชันแฟ้มข้อมูล.....22

2.24 รายการเลือกการกระทำการกับโปรแกรม และลบแฟ้มข้อมูล.....22

2.25 ฟังก์ชันการทำงานบลูทูธ.....23

2.26 ฟังก์ชันการทำงานบลูทูธเมื่อทำการเซตกำลังเป็นออน.....23

2.27 รายชื่ออุปกรณ์ที่มาติดต่อด้วย.....24

2.28 รายละเอียดของรายชื่ออุปกรณ์ที่มาติดต่อด้วย.....24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

2.29 ฟังก์ชันสำหรับจัดรูปแบบหรือลบข้อมูลในหน่วยความจำเอ็นเอ็กซ์ทีทั้งหมด.....25

2.30 แสดงขั้นตอนของเอสดีแอลซี..... 26

2.31 แสดงรูปแบบของการทำงานวนหลายๆ รอบ..... 27

2.32 แสดงตัวอย่างของแผนภาพยูสเคส.....28

2.33 ตัวอย่างของแผนภาพคลาส..... 29

2.34 ตัวอย่างของแผนภาพวัตถุ.....29

2.35 แสดงตัวอย่างแผนภาพลำดับ.....30

2.36 ตัวอย่างของแผนภาพปฏิสัมพันธ์..... 30

2.37 แสดงตัวอย่างแผนภาพสถานะ..... 31

2.38 แสดงแผนภาพกิจกรรมแบบมีทางเลือกตัดสินใจ.....32

2.39 แสดงแผนภาพกิจกรรมแบบมีการทำงานพร้อมกัน.....33

2.40 การแบ่งส่วนด้วยสวิมเลนส์.....34

3.1 แผนภาพยูสเคสของระบบติดตาม และจัดการวัตถุ.....36

3.2 แผนภาพแสดงกิจกรรมการค้นหาวัตถุ..... 42

3.3 แผนภาพแสดงกิจกรรมการตรวจสอบเส้นขอบสนาม..... 43

3.4 แผนภาพแสดงกิจกรรมการแยกแยะสีของวัตถุ..... 44

3.5 แผนภาพแสดงกิจกรรมการผลักวัตถุ.....45

3.6 แผนภาพแสดงกิจกรรมการเคลื่อนที่ของหุ่นยนต์.....46

3.7 แผนภาพคลาสของระบบติดตาม และจัดการวัตถุ..... 47

3.8 แผนภาพลำดับการแยกแยะสี.....48

3.9 แผนภาพลำดับการตรวจสอบเส้นขอบสนาม..... 49

3.10 แผนภาพลำดับการเคลื่อนที่.....50

3.11 แผนภาพลำดับการค้นหาวัตถุ.....51

3.12 แผนภาพลำดับการผลักลูกบอล.....53

3.13 แผนภาพปฏิสัมพันธ์การแยกแยะสีของวัตถุ.....54

3.14 แผนภาพปฏิสัมพันธ์การตรวจสอบเส้นขอบสนาม.....55

3.15 แผนภาพปฏิสัมพันธ์การค้นหา.....55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.16 แผนภาพปฏิสัมพันธ์การผลัก..... 55

3.17 แผนภาพปฏิสัมพันธ์การเคลื่อนที่.....56

3.18 แผนภาพสถานะการควบคุมการเคลื่อนที่หรือเดินไปข้างหน้า..... 56

3.19 หุ่นยนต์ตรวจจับ และจัดการวัตถุ..... 57

3.20 ลักษณะการวางฐานมอเตอร์..... 59

3.21 ลักษณะการวางอุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่.....60

3.22 ลักษณะการต่อฟันเฟืองมอเตอร์กับแขนผลัก..... 61

3.23 ลักษณะการวางตัวตรวจสอบ และแยกแยะวัตถุ..... 62

3.24 ลักษณะการวางอุปกรณ์ตรวจสอบเส้นขอบสนาม.....63

3.25 ลักษณะการวางอุปกรณ์คอนโทรลเลอร์..... 64

4.1 การทดลองหุ่นยนต์ตรวจสอบเส้นขอบสนาม..... 65

4.2 หุ่นยนต์ตรวจพบเส้นขอบสนาม.....65

4.3 หุ่นยนต์เคลื่อนที่หลบไปด้านข้าง..... 66

4.4 หุ่นยนต์ตรวจพบเส้นขอบสนามครั้งที่2..... 66

4.5 หุ่นยนต์เคลื่อนที่หลบไปด้านข้าง..... 67

4.6 การทดลองหุ่นยนต์เพื่อค้นหา และเข้าถึงลูกบอลในสนาม.....68

4.7 หุ่นยนต์ทำการเริ่มค้นหาตำแหน่งของวัตถุ..... 68

4.8 หุ่นยนต์ทำการเก็บค่าระยะทางของวัตถุที่..... 69

4.9 หุ่นยนต์ตรวจพบตำแหน่งของวัตถุเจอ..... 69

4.10 หุ่นยนต์เคลื่อนที่เข้าหาวัตถุ..... 70

4.11 การทดลองหุ่นยนต์ค้นหา และจัดการวัตถุ..... 71

4.12 หุ่นยนต์ทำการเริ่มค้นหาตำแหน่งของวัตถุ..... 71

4.13 หุ่นยนต์ทำการเก็บค่าระยะทางของที่เจอ.....72

4.14 หุ่นยนต์ตรวจพบตำแหน่งของวัตถุที่อยู่ใกล้ที่สุด.....72

4.15 หุ่นยนต์แยกแยะสีของลูกบอล.....73

4.16 หุ่นยนต์หลบหลีกวัตถุที่เป็นเป้าหมาย..... 73

4.17 หุ่นยนต์ตรวจพบเส้นขอบสนาม.....74

4.18 หุ่นยนต์ทำการค้นหาวัตถุ..... 74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.19 หุ่นยนต์ตรวจสอบแยกแยะสีวัตถุ..... 75

4.20 หุ่นยนต์ทำการค้นหาวัตถุ..... 75

4.21 หุ่นยนต์ตรวจพบเส้นขอบสนาม.....76

4.22 หุ่นยนต์ทำการค้นหาวัตถุ..... 76

4.23 หุ่นยนต์ตรวจสอบแยกแยะสีวัตถุ..... 77

4.24 หุ่นยนต์ทำการผลักวัตถุ.....77

ก.1 การติดตั้งวิถี..... 82

ก.2 การเซตค่าวิถี..... 83

ก.3 ใส่ค่าวิถีลงในระบบ..... 83

ก.4 วิธีการลบ Firmware.....84

ก.5 ขั้นตอนการสร้างแฟ้มข้อมูลโครงงาน..... 85

ก.6 การเพิ่มชุดคำสั่งที่ไว้ใช้ในการพัฒนาหุ่นยนต์ให้กับโปรแกรม Eclipse.....86

ก.7 แสดงผลการตั้งค่าให้กับปุ่ม Compile.....87

ก.8 การตั้งค่าให้กับปุ่มบรรจุ..... 88

ก.9 ขั้นตอนการคอมไพล์ และบรรจุข้อมูลลงในหุ่นยนต์..... 88

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันเทคโนโลยีของระบบสมองกลฝังตัว (Embedded System) มีบทบาทสำคัญในชีวิตประจำวันมากขึ้นโดยเฉพาะอุปกรณ์ไฟฟ้าต่างๆ ที่มีระบบสมองกลฝังตัวเข้ามาใช้มากขึ้น แต่เทคโนโลยีสมองกลฝังตัวส่วนใหญ่ต้องพึ่งพิงจากต่างประเทศ ซึ่งในประเทศไทยในปัจจุบันยังไม่มีการพัฒนาระบบสมองกลฝังตัวเป็นมาตรฐานที่แน่ชัด ที่จะส่งผลให้การพัฒนาระบบสมองกลฝังตัวเป็นไปอย่างมีประสิทธิภาพ เพื่อตอบสนองต่อความต้องการในการพัฒนาระบบอย่างมีทิศทาง อีกทั้งยังไม่มีการใช้แนวทางการพัฒนาระบบเข้ามาช่วยในการพัฒนา ที่จะเป็นพื้นฐานในการประยุกต์ใช้งานต่อไปในอนาคต

ดังนั้นหากมีแนวทางในการพัฒนาระบบสมองกลฝังตัวอย่างมีระบบจะทำให้เป็นจุดเริ่มต้นในการสร้างแอพพลิเคชั่นที่มีประสิทธิภาพได้ โดยจะใช้เครื่องมือที่ใช้ในการพัฒนาอย่างยูเอ็มแอล (UML: Unified Modeling Language) กับแนวคิดการออกแบบเชิงวัตถุ (Object Oriented Programming) เป็นเครื่องมือที่ใช้ในการพัฒนาระบบ เพื่อให้เข้าใจถึง ความต้องการ โครงสร้างองค์ประกอบ และสิ่งที่เกี่ยวข้องกับระบบ รวมไปถึงขั้นตอนการทำงาน และความรู้เกี่ยวกับระบบทุกขั้นตอน มาใช้พัฒนาระบบให้กับชุดจำลองเลโก้ (LEGO) รุ่นเอ็นเอ็กซ์ที (NXT) ในการพัฒนาเป็นหุ่นยนต์ตรวจจับและจัดการวัตถุ

1.2 วัตถุประสงค์ของการศึกษา

โครงงานนี้มุ่งหวังเพื่อให้ความรู้ในการพัฒนาระบบสมองกลฝังตัวอย่างเป็นกระบวนการ สามารถนำไปประยุกต์ใช้ในการพัฒนาระบบสมองกลฝังตัวให้กับอุปกรณ์ต่างๆ รวมไปถึงส่งเสริมให้เกิดการพัฒนาระบบสมองกลฝังตัวขั้นพื้นฐานดังนี้

1. เพื่อศึกษาและประยุกต์ใช้หลักการทำงาน ทั้งทางด้านฮาร์ดแวร์ (Hardware) และซอฟต์แวร์ (Software) ของระบบสมองกลฝังตัว
2. เพื่อศึกษาแนวทางในพัฒนาระบบสมองกลฝังตัวโดยใช้ยูเอ็มแอล
3. เพื่อศึกษาและใช้งานเครื่องมือในการพัฒนาระบบสมองกลฝังตัวสำหรับชุดทดลองเลโก้รุ่นเอ็นเอ็กซ์ที
4. เพื่อพัฒนาหุ่นยนต์ตรวจจับและจัดการวัตถุ โดยใช้ชุดทดลองเลโก้รุ่นเอ็นเอ็กซ์ที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ทฤษฎีหรือแนวคิดที่ใช้ในโครงงาน

ระบบสมองกลฝังตัวเป็นระบบที่มีการประมวลผลภายในตัวเอง ซึ่งมีการทำงานเฉพาะทาง ส่งผลทำให้อุปกรณ์ที่ระบบสมองกลฝังตัวเข้าไปทำงาน มีการทำงานตามกระบวนการที่ได้โปรแกรมไว้ โดยระบบสมองกลฝังตัวมีหลักการทำงานเปรียบเสมือนคอมพิวเตอร์ขนาดเล็ก ซึ่งจะมีระบบประฏิบัติการ (Operation System) คอยตอบสนองการทำงานแบบเรียลไทม์ และเรียกใช้คำสั่งที่ใช้ในการบริการแอพพลิเคชัน (Application) และมีดีไวซ์ไดรเวอร์ (Device Driver) ที่ทำหน้าที่ในการดูแลควบคุมส่วนของอุปกรณ์อินพุต และเอาท์พุต อันสุดท้ายคือ มิดเดิลแวร์ (Middleware) หรือโมดูล (Module) ที่ทำหน้าที่สนับสนุนฟังก์ชัน (Function) ทั่วไปของระบบปฏิบัติการ โดยโครงงานนี้ได้นำชุดทดลองเอ็นเอ็กซ์ทีที่เป็นสมองกลฝังตัวมาพัฒนาประกอบกับแนวทางในการพัฒนาระบบเชิงวัตถุที่ใช้
ยูเอ็มแอลมาเป็นเครื่องมือในพัฒนา ซึ่งยูเอ็มแอลก็คือภาษารูปภาพที่เข้ามาช่วยในการบรรยายภาพการทำงานของระบบ เพื่อให้เกิดความเข้าใจ และก่อให้เกิดแบบจำลองที่สามารถนำมาช่วยในการพัฒนาระบบสมองกลฝังตัวที่เข้าใจง่าย

1.4 ขอบเขตของโครงงาน

โครงงานนี้จัดทำขึ้นมาเพื่อศึกษาชุดทดลองเลโก้รุ่นเอ็นเอ็กซ์ที และแนวทางการพัฒนาแอพพลิเคชั่น ดังต่อไปนี้

1. ศึกษาวิเคราะห์ และออกแบบแอพพลิเคชั่นระบบสมองกลฝังตัวหุ่นยนต์ตรวจจับ และจัดการวัตถุโดยใช้ยูเอ็มแอล
2. ศึกษาวิเคราะห์ และออกแบบฮาร์ดแวร์ระบบสมองกลฝังตัวโดยใช้ชุดทดลองเลโก้รุ่นเอ็นเอ็กซ์ทีสำหรับหุ่นยนต์ตรวจจับ และจัดการวัตถุ
3. พัฒนาซอฟต์แวร์ตามยูเอ็มแอลที่ได้ออกแบบไว้ให้สามารถใช้งานในหุ่นยนต์ตรวจจับและจัดการวัตถุได้
4. ทำการทดลอง และวิเคราะห์ผลลัพธ์ที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ขั้นตอนของการศึกษา

ขั้นตอนที่ 1 ศึกษาทฤษฎีที่เกี่ยวข้อง ประกอบด้วยสถาปัตยกรรมของระบบสมองกลฝังตัว ทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์

ขั้นตอนที่ 2 ศึกษาเทคโนโลยีฮาร์ดแวร์ของชุดทดลองเลโก้รุ่นเอ็นเอ็กซ์ที และซอฟต์แวร์ที่ใช้ในการพัฒนา

ขั้นตอนที่ 3 ศึกษาวิเคราะห์ความต้องการของระบบ

ขั้นตอนที่ 4 ออกแบบแอพพลิเคชั่นระบบสมองกลฝังตัว "หุ่นยนต์ตรวจจับและจัดการวัตถุ" โดย
ใช้ยูเอ็มแอล เป็นเครื่องมือในออกแบบ

ขั้นตอนที่ 5 ออกแบบฮาร์ดแวร์หุ่นยนต์ตรวจจับเส้นทาง และอุปกรณ์ในการทดลองตามความต้องการของระบบ

ขั้นตอนที่ 6 พัฒนาโปรแกรมสำหรับควบคุมการทำงานของหุ่นยนต์ตรวจจับและจัดการวัตถุ

ขั้นตอนที่ 7 ทดลองหุ่นยนต์ตรวจจับและจัดการวัตถุในขอบเขตความต้องการที่กำหนดและวิเคราะห์ผลการทดลอง

ขั้นตอนที่ 8 สรุปผลการทดลอง ปัญหาอุปสรรค และข้อเสนอแนะในการพัฒนาหุ่นยนต์ตรวจจับและจัดการวัตถุ

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1. เรียนรู้และเข้าใจเทคโนโลยีระบบสมองกลฝังตัว และการประยุกต์ใช้งาน
2. เรียนรู้และเข้าใจแนวทางการพัฒนาแอพพลิเคชั่นระบบสมองกลฝังตัวอย่างมีระบบ
3. สามารถประยุกต์การออกแบบหุ่นยนต์ให้มีความยืดหยุ่นต่อการใช้งาน และการพัฒนาระบบภายใต้อุปกรณ์ที่มีอยู่อย่างจำกัดได้
4. สามารถ ออกแบบและพัฒนาแอพพลิเคชั่นสำหรับหุ่นยนต์ตรวจจับและจัดการวัตถุตามหลักยูเอ็มแอลได้
5. สามารถ พัฒนาแอพพลิเคชั่นสำหรับหุ่นยนต์ตรวจจับและจัดการวัตถุตามที่ได้ออกแบบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีบท

2.1 ระบบเทคโนโลยีสมองกลฝังตัว (Embedded Technology System)

ระบบสมองกลฝังตัวเสมือนระบบคอมพิวเตอร์ขนาดเล็กที่ออกแบบมาเพื่อควบคุมการทำงานเฉพาะทางหรืออุปกรณ์เฉพาะด้านที่มีฟังก์ชัน (Function) การใช้งานที่ชัดเจน และสามารถตอบสนองการทำงานแบบเรียลไทม์ (Real Time) คือระบบฟังก์ชันการทำงานที่สร้างขึ้นต้องสามารถประมวลผล และตอบสนองคำร้องหรืออินพุตจากผู้ใช้ได้ทันทีหรือภายในระยะเวลาที่กำหนดเอาไว้

โดยการตอบสนองแบบเรียลไทม์มี 2 แบบ คือ ฮาร์ดเรียลไทม์ (Hard-Real Time) คือ ระบบสมองกลฝังตัวจะต้องตอบสนองกับการทำงานได้ทันที ส่วนซอฟต์เรียลไทม์ (Soft-Real Time) คือ ระบบสมองกลฝังตัวที่มีความยืดหยุ่นในเรื่องของเวลาที่ตอบสนองสามารถตอบสนองกับการทำงานได้ภายในระยะเวลาที่กำหนด หรือไม่ล่าช้าเกินไป

2.2 องค์ประกอบฮาร์ดแวร์ (Hardware Component)

2.2.1 คุณลักษณะเฉพาะของเทคโนโลยีสมองกลฝังตัว (Characteristics of Embedded Technology)

ลักษณะเฉพาะของเทคโนโลยีสมองกลฝังตัว เป็นเทคโนโลยีที่มีความหลากหลายมากทั้งด้านการใช้งาน ทั้งขนาดรูปร่าง เพื่อให้ง่ายต่อการพัฒนาจำกัดขอบเขตไม่ให้เกิดความหลากหลายมากเกินไป รวมถึงสามารถใช้งานร่วมกันในสาขาที่แตกต่างกันได้ จึงได้นำเอาฟังก์ชันคอมพิวเตอร์มาสร้างเป็นพื้นฐานร่วมกัน ส่งผลให้ซอฟต์แวร์เป็นสิ่งสำคัญในการพัฒนาระบบสมองกลฝังตัว

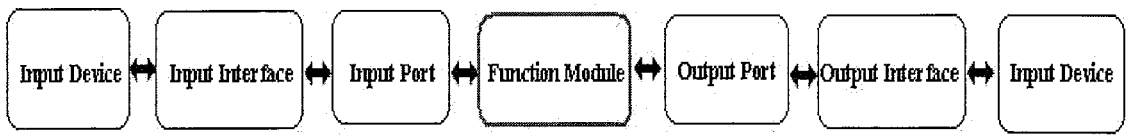
ประโยชน์ที่ได้จากการแปลงฟังก์ชันเชิงวงจรไมโครชิบ (Microchip) และฟังก์ชันการทำงานสำหรับประมวลผลตอบสนองต่อเหตุการณ์เพื่อควบคุมไมโครชิบ

1. ขนาดสัดส่วนของตัวฮาร์ดแวร์มีขนาดเล็กลง กระบวนการออกแบบและพัฒนามีความยืดหยุ่นมากขึ้น เนื่องจากองค์ประกอบเชิงฮาร์ดแวร์ถูกพัฒนาไปอยู่บนชิบ (SOC: System-On-Chip)

2. ทำให้เราสามารถสร้างฟังก์ชันงานที่ยากหรือซับซ้อนลงในอุปกรณ์ที่มีขนาดเล็กได้

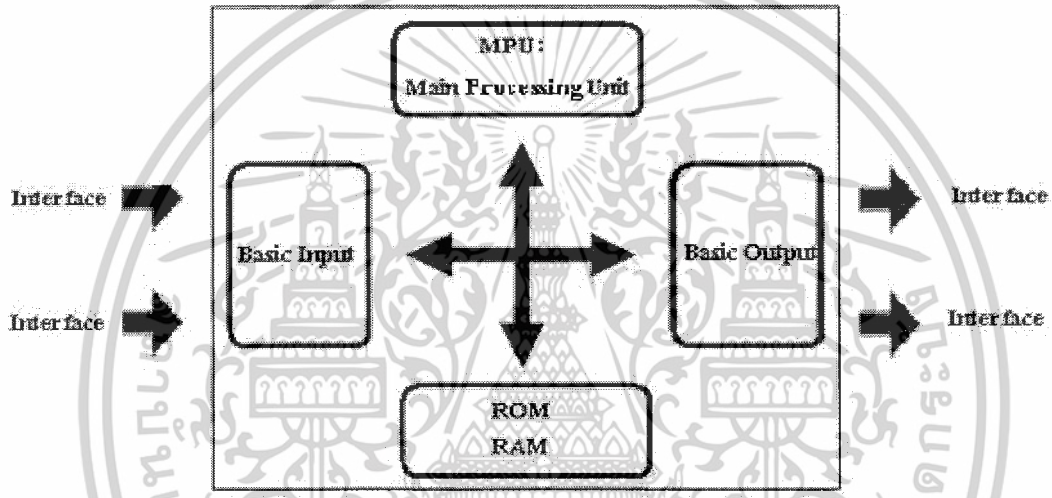
ภาพรวมส่วนประกอบฮาร์ดแวร์ของระบบสมองกลฝังตัว กล่าวถึงอุปกรณ์ที่เป็นองค์ประกอบในระบบสมองกลฝังตัว ซึ่งประกอบได้ด้วย อุปกรณ์อินพุตเอาต์พุต อินเทอร์เฟสอินพุตเอาต์พุต พอร์ตอินพุตเอาต์พุต ดังรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 โครงสร้างโดยสังเขปของระบบสมองกลฝังตัว (Japan System House Association. 2549 : 89)

2.2.2 สถาปัตยกรรมของระบบสมองกลฝังตัว (Embedded System Architecture)



รูปที่ 2.2 สถาปัตยกรรมพื้นฐานของระบบสมองกลฝังตัว (Japan System House Association. 2549 : 89)

ลักษณะโครงสร้างพื้นฐานของระบบสมองกลฝังตัวดังรูปที่ 2.2 ประกอบด้วย

1. เอ็มพียู (MPU: Micro Processing Unit) เป็นหน่วยประมวลผลการทำงานต่างๆ ของระบบสมองกลฝังตัว

2. รอม (ROM: Read Only Memory) หน่วยความจำสำหรับเก็บข้อมูลที่ไม่จำเป็นต้องมีการเปลี่ยนแปลง

3. แรม (RAM: Random Access Memory) หน่วยความจำสำหรับเก็บข้อมูลที่มีการเปลี่ยนแปลงบ่อย การเก็บค่าหรือเขียนทับค่าเก่า

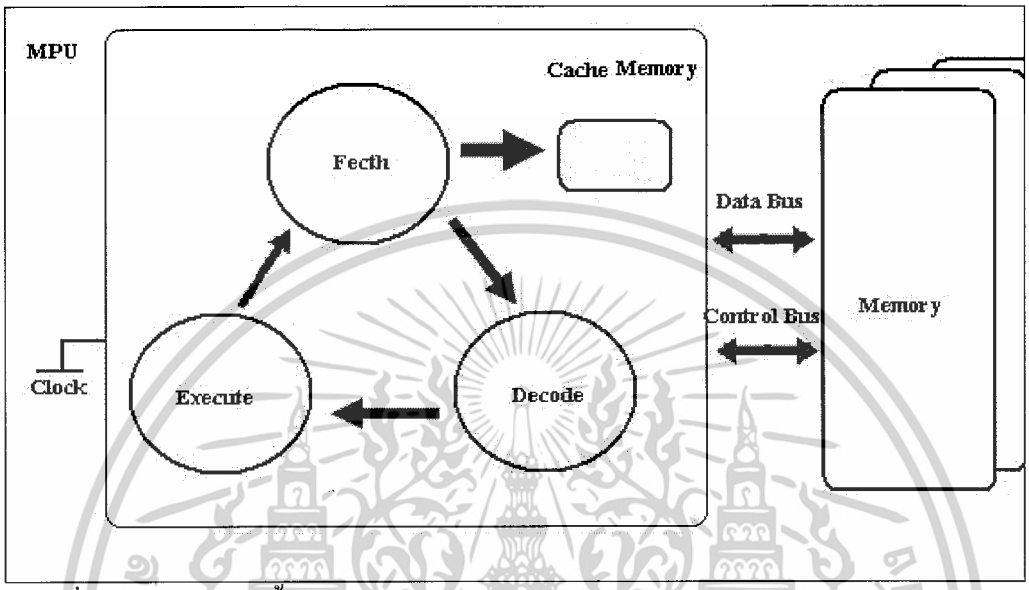
4. อินพุตพอร์ต (Input Ports) เป็นส่วนที่รับข้อมูลหรือสัญญาณต่างๆ ของระบบสมองกลฝังตัว โดยส่งข้อมูลเป็นดิจิตอลไปยังเอ็มพียู เพื่อนำไปประมวลผลต่อ

5. เอาต์พุตพอร์ต (Output Ports) เป็นตัวส่งสัญญาณที่ได้จากการคำนวณของเอ็มพียู ไปยังอุปกรณ์เอาต์พุตต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของอุปกรณ์เซนเซอร์จำเป็นที่จะต้องมีอินเทอร์เฟส ที่ทำหน้าที่แปลงสัญญาณจากตัวเซนเซอร์เป็นสัญญาณที่ส่วนอินพุตของไมโครคอมพิวเตอร์รับได้

2.2.2.1 เอ็มพียู (MPU: Micro Processing Unit)



รูปที่ 2.3 การทำงานพื้นฐานของเอ็มพียู (Japan System House Association. 2549 : 92)

ลักษณะการทำงานพื้นฐานของเอ็มพียูดังรูปที่ 2.3 ซึ่งแบ่งขั้นตอนการทำงานหลักๆ คือ

1. เฟตซ์ (Fetch) เป็นการดึงคำสั่งจากหน่วยความจำแคช
2. ดีโค้ด (Decode) เป็นการถอดรหัสคำสั่งที่ดึงมาจากขั้นตอนเฟตซ์
3. เอ็ซคิวต์ (Execute) เป็นการนำคำสั่งที่ถอดรหัสแล้วมาประมวลผล

2.2.2.2 บัส (Bus)

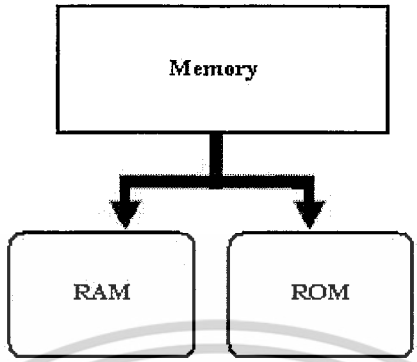
เป็นช่องทางในการติดต่อสื่อสาร แลกเปลี่ยนข้อมูลของระบบสมองกลฝังตัว แบ่งตามอุปกรณ์ที่ใช้งานได้ 2 ประเภทหลักๆ คือ บัสภายใน (Internal Bus) และบัสภายนอก (External Bus)

1. บัสภายในเป็นช่องทางสำหรับส่งผ่านข้อมูลร่วมกันภายในไมโครคอนโทรเลอร์ ระหว่าง เอ็มพียู หน่วยความจำ และอินพุตเอาต์พุตพื้นฐาน แบ่งออกเป็น 3 ประเภท ได้แก่ แอดเดรสบัส (Address Bus) เป็นตัวกำหนดขนาดของพื้นที่แอดเดรสที่บัสนั้นสามารถอ้างอิงถึงได้ บัสข้อมูล (Data Bus) เป็นตัวบอกขนาดของข้อมูลที่ใช้ส่งผ่านบัส และบัสควบคุม (Control Bus) เป็นตัวควบการส่งข้อมูล จังหวะการรับส่งข้อมูล รวมถึงการเริ่มและสิ้นสุดการรับส่งข้อมูลผ่านช่องทางบัส

2. บัสภายนอกเป็นช่องทางสำหรับส่งผ่านข้อมูลระหว่างตัวไมโครคอนโทรเลอร์กับอุปกรณ์ภายนอกต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.3 หน่วยความจำ (Memory)



รูปที่ 2.4 โครงสร้างพื้นฐานหน่วยความจำ

หน่วยความจำเป็นที่สำหรับจัดเก็บโปรแกรม และข้อมูลที่จำเป็นสำหรับไมโครคอนโทรลเลอร์ ดังรูปที่ 2.4 โดยลักษณะของหน่วยความจำแบ่งออกเป็น 2 ประเภท ดังต่อไปนี้

1. แรมเป็นหน่วยความจำชั่วคราว ต้องจ่ายไฟในการจับเก็บข้อมูลที่บันทึกไว้ มีความเร็วในการเข้าถึงข้อมูลสูง โดยมีแอดเดรสเป็นตัวระบุตำแหน่งของข้อมูลนั้น ๆ

2. รอมเป็นหน่วยความจำถาวร ไม่ต้องจ่ายไฟในการจัดเก็บข้อมูลที่บันทึกไว้ โดยมีแอดเดรสเป็นตัวระบุตำแหน่งของข้อมูลนั้นๆ

2.2.2.4 อินพุตเอาต์พุตพื้นฐาน (Basic Input/Output)

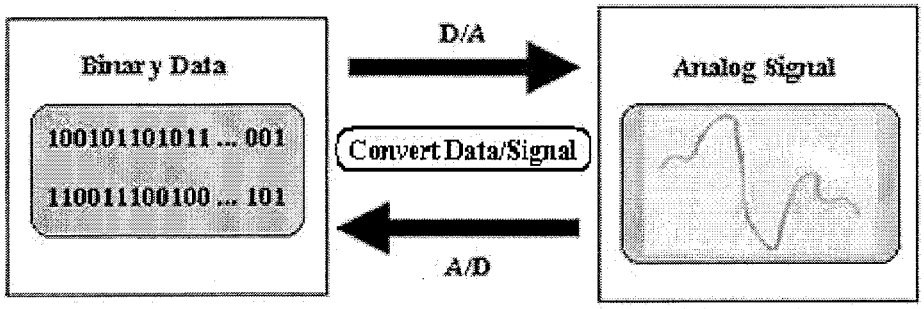
เป็นส่วนที่ติดต่อกับอุปกรณ์ภายนอกใช้รับส่งข้อมูล ซึ่งข้อมูลที่ใช้ในการรับส่งนั้นมีทั้งสัญญาณดิจิตอล (Digital Signal) และสัญญาณอนาล็อก (Analog Signal) ดังต่อไปนี้

1. ระดับสัญญาณไฟฟ้าในวงจรดิจิตอลมี 2 ระดับ คือ ระดับ H มีแรงดันไฟฟ้ามาจากแหล่งจ่ายไฟ และระดับ L มีแรงดันไฟฟ้าเป็น 0 โวลต์ (Volt) แต่ในโปรแกรมจะแทนค่าสัญญาณดิจิตอลด้วยเลขฐานสองโดย H มีค่าเป็นเลข “1” และ L มีค่าเป็นเลข “0”

2. ระดับสัญญาณไฟฟ้าวงจรอนาล็อกเป็นสัญญาณแบบต่อเนื่องหรืออาจมีการเปลี่ยนแปลงไปมาตลอดเวลา

กระบวนการแปลงข้อมูลดิจิตอล และสัญญาณอนาล็อกเพื่อใช้งานหรือติดต่อกับอุปกรณ์ภายนอกสามารถแปลงข้อมูลดิจิตอลเป็นอนาล็อก และสัญญาณอนาล็อกเป็นดิจิตอลดังรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 การแปลงข้อมูลดิจิตอลเป็นอนาล็อก และสัญญาณอนาล็อกเป็นดิจิตอล (Japan System House Association. 2549 : 89)

การแปลง D/A เป็นการแปลงสัญญาณดิจิตอลเป็นอนาล็อก

การแปลง A/D เป็นการแปลงสัญญาณอนาล็อกเป็นดิจิตอล

2.2.3 เทคโนโลยีที่เกี่ยวข้องกับหน่วยประมวลผล

2.2.3.1 เทคโนโลยีอินเทอร์รัพต์ (Interrupt Technology)

การทำงานของระบบสมองกลฝังตัวเป็นแบบเรียลไทม์ สามารถตอบสนองต่ออินพุตที่รับเข้ามา และประมวลผลได้ภายในเวลาที่กำหนด แต่หน่วยประมวลผลเอ็มพียูสามารถประมวลผลหรือทำงานได้ทีละอย่าง เมื่อมีงานหรือเหตุการณ์อื่น ๆ เข้ามาหน่วยประมวลผลทำงานจากโปรแกรมเดิมไปอีกทำงานโปรแกรมหนึ่งแทน เหตุการณ์เหล่านี้เป็นปัจจัยของการเกิดอินเทอร์รัพต์ ดังนั้นเทคโนโลยีอินเทอร์รัพต์มีไว้เพื่อตอบสนองเหตุการณ์หลาย ๆ เหตุการณ์ และจะต้องกลับไปทำงานโปรแกรมเดิมให้เสร็จสมบูรณ์ภายในเวลาที่กำหนด

เมื่อมีความต้องการอินเทอร์รัพต์ (Interrupt Request) ทำให้มีการสลับการทำงานไปยังโปรแกรมมวลผลอินเทอร์รัพต์ (Interrupt Processing Program) โดยการสลับนี้มี 2 วิธี ดังต่อไปนี้

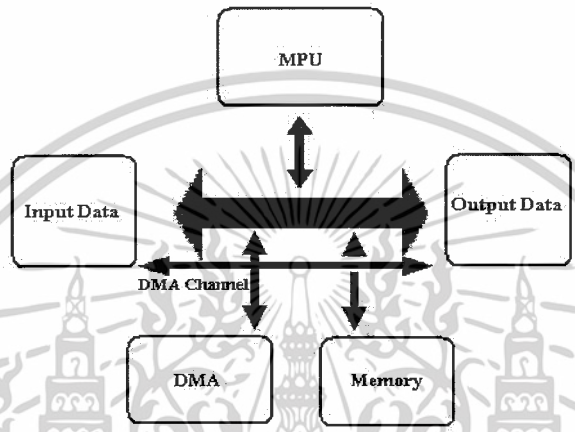
1. เป็นการเข้าไปยังแอดเดรสในหน่วยความจำสำหรับจัดการกับโปรเซส (Process) นั้นโดยตรง

2. ใช้ตารางเวกเตอร์ (Vector Table) ทำหน้าที่เก็บตำแหน่งการทำงานของโปรแกรมต่าง ๆ ไว้ ในตาราง เมื่อเกิดอินเตอร์รัพต์ขึ้นก็สามารถอ้างอิงตำแหน่งของการทำงานจากตารางเข้าไปจัดการกับโปรเซสนั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.2 เทคโนโลยีดีเอ็มเอ (DMA Technology)

ดีเอ็มเอเป็นการเคลื่อนย้ายข้อมูลระหว่างหน่วยความจำและอุปกรณ์อินพุตเอาท์พุต หรือหน่วยความจำด้วยกันเอง โดยมีฮาร์ดแวร์เป็นตัวควบคุมการเคลื่อนย้ายข้อมูลจากหน่วยความจำโดยตรงไม่ต้องอาศัยการทำงานของเอ็มพียูผ่านช่องทางหรืออุปกรณ์ช่องสัญญาณการส่งข้อมูล (DMA Channel) ดังรูปที่ 2.6



รูปที่ 2.6 โฟลว์ของการประมวลผลการร้องขออินเทอร์รัพต์

การเคลื่อนย้ายข้อมูลแบบดีเอ็มเอมีอยู่ 3 โหมด คือ

1. การเคลื่อนย้ายข้อมูลเป็นไบต์ (Byte Transfer Mode) เป็นการขอสิทธิ์ในการครอบครองบัส หลังจากที่เคลื่อนย้ายข้อมูลขนาด 1 ไบต์ แล้วก็จะคืนสิทธิ์ในการใช้บัสให้แก่เอ็มพียูทำงานต่อไป

2. การเคลื่อนย้ายข้อมูลเป็นบล็อก (Block Transfer Mode) เป็นการจองบัสจนกว่าจะเคลื่อนย้ายข้อมูลครบตามขนาดที่กำหนดไว้ หลังจากนั้นก็จะคืนสิทธิ์ในการใช้บัสให้แก่เอ็มพียูทำงานต่อไป

3. การเคลื่อนย้ายข้อมูลตามความต้องการ (Burst Transfer Mode) เป็นการเคลื่อนย้ายข้อมูลอย่างต่อเนื่อง โดยมีตัวควบคุมคอยกำหนดสถานการณ์ครอบครองสิทธิ์ในการใช้บัส

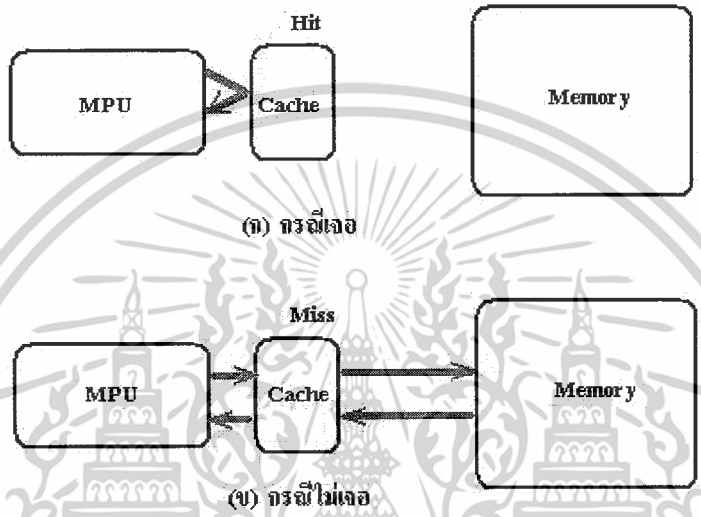
2.2.3.3 เทคโนโลยีหน่วยความจำแคช (Cache Memory Technology)

เป็นหน่วยความจำที่มีขนาดเล็กแต่มีความเร็วสูงติดตั้งอยู่ระหว่างเอ็มพียู และหน่วยความจำหลัก โดยมีข้อมูลที่ก๊อบปี้ (Copy) มาจากหน่วยความจำหลักมาไว้ที่หน่วยความจำแคช เพื่อให้เอ็มพียูเป็นตัวอ่านข้อมูลจากหน่วยความจำแคช แยกเป็น 2 กรณี ดังรูปที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เอ็มพียูอ่านข้อมูลในหน่วยความจำแคชแล้วพบข้อมูลที่ต้องการ (Hit) จะทำการประมวลผลข้อมูลนั้นเลย

2. เอ็มพียูไม่พบข้อมูลที่ต้องการในหน่วยความจำแคช (Miss Hit) หน่วยความจำแคชจะทำการก๊อบปี้ข้อมูลที่ต้องการจากหน่วยความจำหลักมาแทนที่ข้อมูลเก่า แล้วให้เอ็มพียูใช้งานจากหน่วยความจำแคชใหม่อีกครั้ง



รูปที่ 2.7 การทำงานของหน่วยความจำแคช (Japan System House Association. 2549 : 89)

2.3 องค์ประกอบซอฟต์แวร์ (Software Component)

เป็นชุดคำสั่งหรือโปรแกรมเพื่อให้อุปกรณ์คอนโทรลเลอร์เกิดการทำงานให้เป็นไปตามความต้องการของผู้ใช้งาน โดยซอฟต์แวร์ทำหน้าที่จัดการ และควบคุมการประมวลผลต่างๆ ของระบบสมองกลฝังตัว ซอฟต์แวร์ของระบบสมองกลฝังตัวแบ่งออกเป็น 2 ประเภท คือ ระบบปฏิบัติการ (Operating System) และแอพพลิเคชั่นโปรแกรม (Application Program) (Japan System House Association. 2549 : 18)

2.3.1 ระบบปฏิบัติการ

ใช้ควบคุมการทำงานทั้งหมดของระบบสมองกลฝังตัว คอยจัดการทรัพยากรเพื่อรองรับการทำงานของแอพพลิเคชั่นต่างๆ ซึ่งประกอบด้วย เคอร์เนล (Kernel) ดีไวซ์ไดรเวอร์ (Device Driver) และมิดเดิลแวร์ (Middleware)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1.1 เคอร์เนล

เป็นหน่วยสนับสนุนขั้นพื้นฐานที่มีฟังก์ชันพื้นฐานทำหน้าที่บริการแอปพลิเคชั่นต่างๆ ที่การดำเนินงานอยู่บนเคอร์เนลนั้น โดยเคอร์เนลมีหน้าที่จัดการควบคุมเอ็มพียูเพื่อให้ง่ายต่อการสร้างระบบเรียลไทม์ โดยเคอร์เนลมีหน้าที่ในการสร้างฟังก์ชันพื้นฐานของระบบปฏิบัติการ ซึ่งประกอบด้วย

1. ฟังก์ชันการจัดการตารางเวลางาน (Task Scheduling Function) เป็นการใช้ตารางเวลาในการจัดเก็บค่าความสำคัญ (Priority) ของแต่ละงาน โดยงานที่มีค่าความสำคัญสูงกว่าค่าพรีเอมพ์ชั่น (Preemption) หรือค่าความสำคัญสูงที่สุดก็จะได้ทำงานก่อน ซึ่งกระบวนการการเกิดอินเทอร์รัพต์ประกอบด้วย การเริ่มงาน การหยุดงาน การอินพุตเอาต์พุต การหยุดโปรเซส

2. ฟังก์ชันการควบคุมอินเทอร์รัพต์ (Interruption Control Function) เป็นการจัดการ และควบคุมอินเทอร์รัพต์ที่เกิดขึ้น โดยการบันทึกค่าของตัวจัดการอินเทอร์รัพต์ และการตั้งค่าเวกเตอร์ (Vector)

3. ซิสเต็มคอล (System Call Function) เป็นฟังก์ชันที่ทำให้แทสค์ (Task) สามารถทำงานพร้อม ๆ กันได้ และสามารถตอบสนองภายในเวลาที่กำหนด โดยมีการเริ่มงาน (Task Start) หยุดงาน (Task Stop) การลบงาน (Task Cancellation) การติดต่อสื่อสารระหว่างงาน (Inter-Task Communication) การเข้าจังหวะ (Synchronization) และการป้องกันการเกิดภาวะตาย (Dead Lock) (Mutual Exclusive Control) เป็นตัวที่จัดการอินเทอร์รัพต์

2.3.1.2 ดีไวซ์ไดรเวอร์ (Device Driver)

ดีไวซ์ไดรเวอร์เป็นฟังก์ชันสำคัญที่ช่วยสนับสนุนการทำงานของระบบสมองกลฝังตัว และเป็นโมดูล (Module) ที่ทำให้เกิดฟังก์ชันในการเชื่อมต่อ และควบคุมอุปกรณ์ไอ/โอ (I/O Device)

2.3.1.3 มิดเดิลแวร์ (Middleware)

มิดเดิลแวร์เป็นโมดูลที่ทำหน้าที่สนับสนุนฟังก์ชันการทำงานทั่วไปของระบบปฏิบัติการ เช่น ระบบแฟ้มข้อมูล (File System) ในระบบทั่วไปดีไวซ์ไดรเวอร์ และ มิดเดอแวร์ จะรวมเข้ากับเคอร์เนลประกอบกันเป็นระบบปฏิบัติการส่วนประกอบทั้งหมด นอกจากเป็นอุปกรณ์ชนิดพิเศษซึ่งสามารถต้องติดตั้ง ดีไวซ์ไดรเวอร์หรือมิดเดอแวร์ได้เองภายหลัง

2.3.2 แอพพลิเคชั่นโปรแกรม

โปรแกมที่ถูกพัฒนาขึ้นเพื่อทำหน้าที่เฉพาะอย่าง หรือประมวลผลแก้ไขปัญหาโดยเฉพาะภายใต้การทำงานของระบบปฏิบัติการที่จัดเตรียมที่กล่าวมาข้างต้น

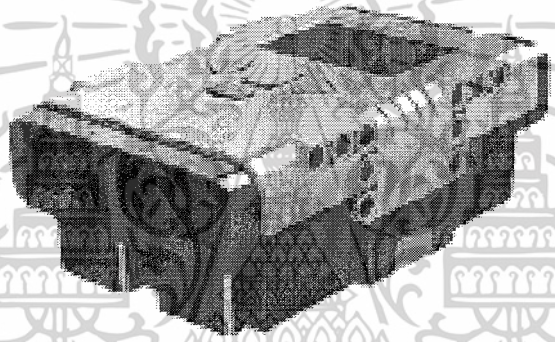
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 เอ็นเอ็กซ์ทีฮาร์ดแวร์ (NXT Hardware)

เอ็นเอ็กซ์ทีเปรียบเสมือนตัวคอนโทรลเลอร์ เป็นศูนย์กลางในการประมวลผล ควบคุมการทำงานต่างๆ ของหุ่นยนต์ รวมทั้งใช้ในการติดต่อสื่อสารระหว่างอุปกรณ์ต่างๆ พัฒนาโดยบริษัทเลโก้ (The Lego Group)

2.4.1 สมองกลฝังตัวเอ็นเอ็กซ์ที (NXT)

เอ็นเอ็กซ์ทีเป็นอุปกรณ์ฮาร์ดแวร์เปรียบเสมือนสมองของหุ่นยนต์ (Robot) ค่อยสั่งการ และควบคุมการทำงานต่างๆ ของหุ่นยนต์ (Lego Group. 2007 : http://mindstorms.lego.com /Overview/ The_NXT.aspx) ดังรูปที่ 2.8



รูปที่ 2.8 อุปกรณ์เอ็นเอ็กซ์ที (Lego Group. 2007 : http://mindstorms.lego.com /Overview /The_NXT.aspx)

2.4.1.1 ส่วนประกอบภายนอก

1. มอเตอร์พอร์ต (Motor ports) ในเอ็นเอ็กซ์ทีมีเอาต์พุต 3 ตัว สำหรับควบคุมมอเตอร์ ประกอบด้วยพอร์ต เอ (A), บี (B) และซี (C) (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx)

2. เซ็นเซอร์พอร์ต (Sensor Ports) ในเอ็นเอ็กซ์ทีมีอินพุต 4 พอร์ต สำหรับรับผลจากเซ็นเซอร์ ประกอบด้วย พอร์ต 1, 2, 3 และ 4 (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx)

3. ยูเอสบีพอร์ต (USB port) เป็นช่องทางการติดต่อกับยูเอสบีจากภายนอก สำหรับบรรจุลง (Down Load) โปรแกรมจากเครื่องคอมพิวเตอร์ไปยังตัวเอ็นเอ็กซ์ทีนี้ หรือบรรจุขึ้น (Up Load) ข้อมูลจากตัวหุ่นยนต์ไปยังเครื่องคอมพิวเตอร์ ซึ่งมีความเร็วสูงสุดในการรับส่งข้อมูลอยู่ที่

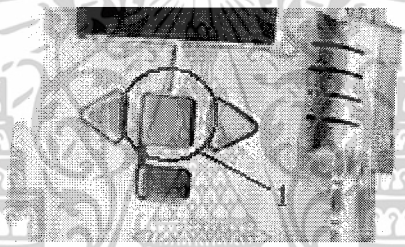
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12 เม็กกะบิตต่อวินาที (12 Mbit/s) และยังสามารถใช้การติดต่อในรูปแบบบลูทูธ (Bluetooth) เพื่อส่งข้อมูล และรับข้อมูลได้ (Lego Group. 2007 : http://mindstorms.lego.com /Overview /The_NXT.aspx)

4. ลำโพง (Loudspeaker) ใช้สำหรับแสดงเสียงเมื่อมีการดำเนินงาน (Run) โปรแกรมเกิดขึ้น หรือเกิดข้อผิดพลาดเกิดขึ้น (Exception) (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx)

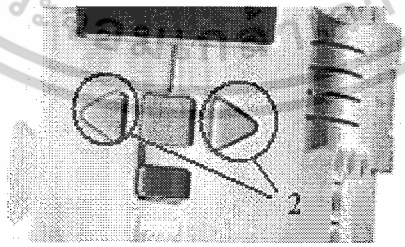
5. ปุ่มบนตัวเอ็นเอ็กซ์ (NXT Buttons) ใช้สำหรับควบคุมการใช้งานบนรายการเลือก (Menu) ของตัวเอ็นเอ็กซ์ ประกอบด้วย

ปุ่มกดสีส้ม (Orange Button) ใช้สำหรับ เปิดโหมดออน (On) หรือเข้า (Enter) หรือการดำเนินงาน (Run) (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx) ดังรูปที่ 2.9



รูปที่ 2.9 หมายเลข 1 คือปุ่มสำหรับเปิดโหมดออนหรือเข้าหรือการดำเนินงาน

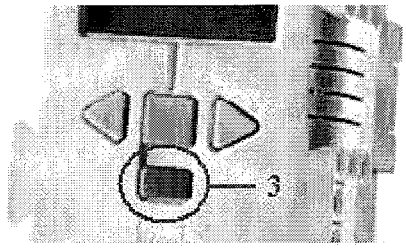
ปุ่มลูกศรสีเทาสว่าง (Light Grey Arrows) ใช้สำหรับเลื่อนซ้าย และขวาบนรายการเลือกของเอ็นเอ็กซ์ (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx) ดังรูปที่ 2.10



รูปที่ 2.10 หมายเลข 2 คือปุ่มสำหรับสำหรับเลื่อนซ้าย และขวาบนรายการเลือกของเอ็นเอ็กซ์

ปุ่มกดสีเทาเข้ม (Dark Grey Button) ใช้สำหรับลบล้าง (Clear) หรือย้อนกลับไป (Go back) (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx) ดังรูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 หมายเลข 3 คือปุ่มสำหรับเคลียร์หรือย้อนกลับไป

6. หน่วยแสดงผลของตัวเอ็นเอ็กซ์ (NXT Display) เป็นแบบแอลซีดี (LCD) มีขนาด 100 x 64 พิเซล (Pixel) ใช้สำหรับแสดงข้อมูลที่จำเป็นต่อการทำงาน เช่น โหมดทางเลือกการทำงาน (Options) รูปภาพไอคอน (Icons) (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx)

7. แหล่งพลังงาน (Power source) ใช้ถ่านแบตเตอรี่เอเอ 6 ก้อน (6 AA batteries) (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx)

2.4.1.2 ส่วนประกอบภายใน

1. หน่วยประมวลผลหลัก (Main Processor) รุ่นเออาร์เอ็มเซเว่น (Atmel® 32-bit ARM7® processor, AT91SAM7S256), 256 KB FLASH, 64 KB RAM, 48 MHz (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx)

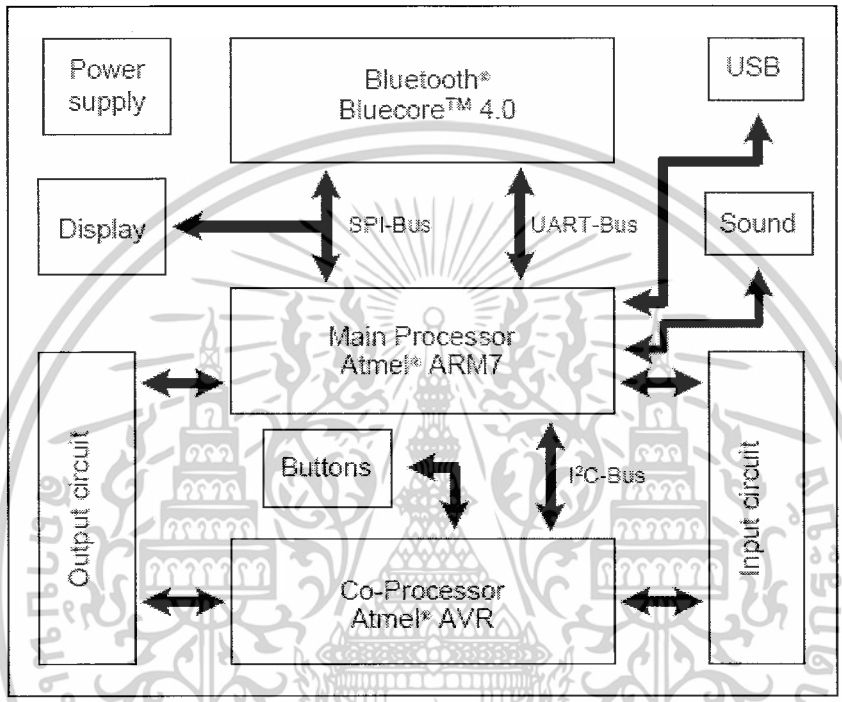
2. หน่วยประมวลผลร่วม (Co-processor) รุ่นเอวีอาร์ (Atmel® 8-bit AVR processor, ATmega48), 4 KB FLASH, 512 Byte RAM (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx)

3. การติดต่อสื่อสารไร้สายด้วยบลูทูธเวอร์ชั่น 2.0 (Bluetooth Class II V2.0 complaint) (Lego Group. 2007 : http://mindstorms.lego.com /Overview/The_NXT.aspx)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.3 สถาปัตยกรรมภายในของสมองกลฝังตัวเอ็นเอ็กซ์ที

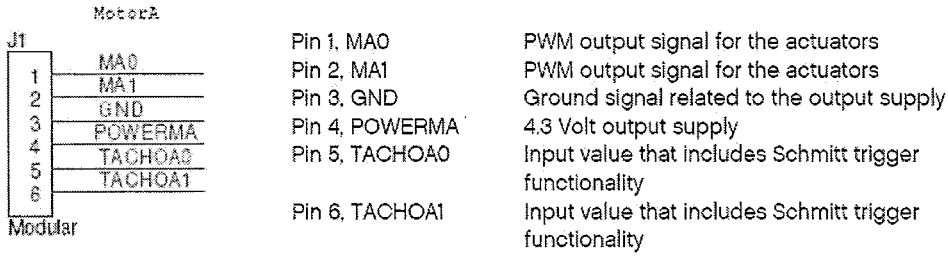
สถาปัตยกรรมภายในของสมองกลฝังตัวเอ็นเอ็กซ์ที ประกอบด้วย หน่วยประมวลผลหลักรุ่นเออาร์เอ็ม หน่วยประมวลผลร่วมรุ่นเอวีอาร์ บลูทูธ ยูเอสบีพอร์ต อินพุต 4 พอร์ต เอาต์พุต 3 พอร์ต หน่วยแสดงผล ลำโพง ส่วนติดต่อกับปุ่มควบคุม และแหล่งพลังงาน ดังรูปที่ 2.12



รูปที่ 2.12 สถาปัตยกรรมภายในของสมองกลฝังตัวเอ็นเอ็กซ์ที

2.4.1.4 เอาต์พุตพอร์ต (Output Ports)

ลักษณะสถาปัตยกรรมภายในของเอาต์พุตพอร์ตที่ใช้เชื่อมต่ออุปกรณ์ เพื่อส่งสัญญาณข้อมูลไปยังอุปกรณ์ภายนอก ดังรูปที่ 2.13



รูปที่ 2.13 เอาต์พุตพอร์ต (Lego Group. 2007 : http://nxtasy.org/wp-content/uploads/2006/08/digital%20vs%20analog.pdf)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

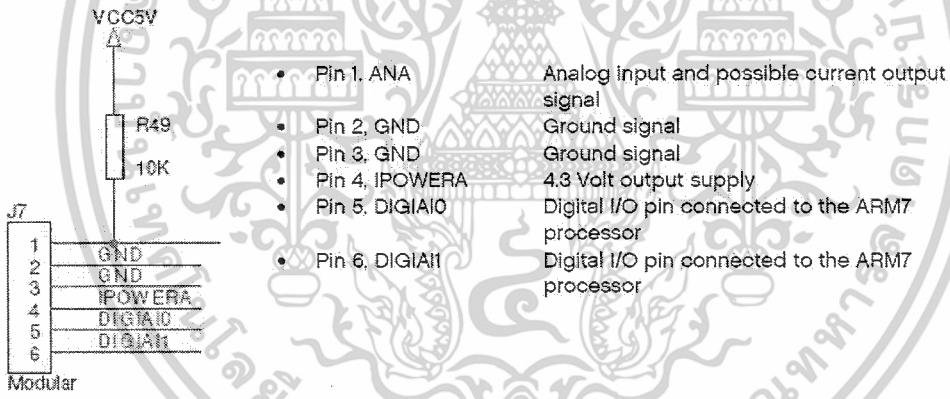
รายละเอียดแต่ละพิน (Pin) บนสถาปัตยกรรมภายในของเอาต์พุตพอร์ต ประกอบด้วย

1. พินที่ 1 ช่องทางส่งสัญญาณเอาต์พุต
2. พินที่ 2 ช่องทางส่งสัญญาณเอาต์พุต
3. พินที่ 3 สายกราวนด์ (Ground)
4. พินที่ 4 ช่องทางส่งศักย์ไฟฟ้า
5. พินที่ 5 ช่องทางสำหรับรับค่าอินพุตโดยใช้ฟังก์ชันชมิตต์ทริกเกอร์ (Schmitt trigger)
6. พินที่ 6 ช่องทางสำหรับรับค่าอินพุตโดยใช้ฟังก์ชันชมิตต์ทริกเกอร์

ฟังก์ชันชมิตต์ทริกเกอร์เป็นส่วนของอินพุต์ที่เพิ่มเข้ามาระหว่างพอร์ต และอินพุตพินบนหน่วยประมวลผลเออาร์เอ็มเซเว่น

2.41.5 อินพุตพอร์ต (Input Ports)

ลักษณะสถาปัตยกรรมภายในของอินพุตพอร์ตที่ใช้เชื่อมต่ออุปกรณ์ เพื่อรับสัญญาณข้อมูลจากอุปกรณ์ภายนอกเข้ามาประมวลผล ดังรูปที่ 2.14



รูปที่ 2.14 อินพุตพอร์ต (Lego Group. 2007 : http://nxtasy.org/wp-content/uploads/2006/08/digital%20vs%20analog.pdf)

รายละเอียดแต่ละพินบนสถาปัตยกรรมภายในของอินพุตพอร์ต

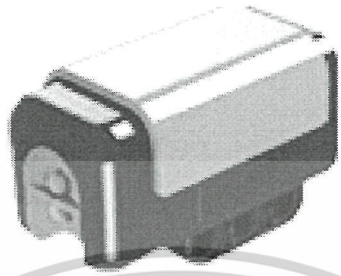
1. พินที่ 1 ช่องทางส่งสัญญาณอนาล็อก
2. พินที่ 2 กราวนด์
3. พินที่ 3 กราวนด์
4. พินที่ 4 ช่องทางส่งศักย์ไฟฟ้า
5. พินที่ 5 ช่องทางไอ/โอ (I/O) ที่ใช้เชื่อมต่อกับหน่วยประมวลผลเออาร์เอ็มเซเว่น
6. พินที่ 6 ช่องทางไอโอที่ใช้เชื่อมต่อกับหน่วยประมวลผลเออาร์เอ็มเซเว่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

2.5.2 อุปกรณ์เสริม

2.5.2.1 อุปกรณ์เซ็นเซอร์ด้วยแสง (Light Sensor)

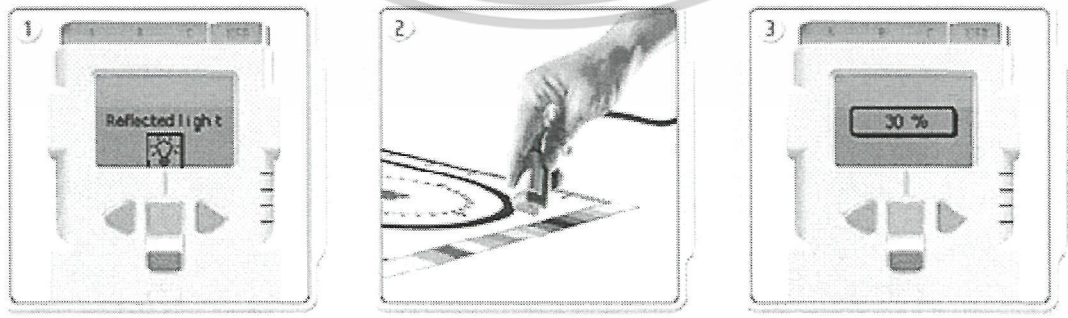


รูปที่ 2.15 อุปกรณ์เซ็นเซอร์ด้วยแสง (Lego Group. 2007 : http://mindstorms.lego.com/Overview /Light_Sensor.aspx)

จากรูปที่ 2.15 เป็นอุปกรณ์สำหรับจำแนกความแตกต่างของสีจากความเข้มของสีบนพื้นผิววัตถุกับแสงภายในห้อง [4] ซึ่งลักษณะของการอ่านค่าจากอุปกรณ์นี้จะอ่านได้จากค่าความเข้มของสี ดังรูปที่ 2.16 แสดงลักษณะของสีที่มนุษย์เรามองเห็นกับสีที่อุปกรณ์นี้มองเห็น

รูปที่ 2.16 ลักษณะการมองเห็นของอุปกรณ์เซ็นเซอร์ด้วยแสง (Lego Group. 2007 : http://mindstorms.lego.com/Overview/Light_Sensor.aspx)

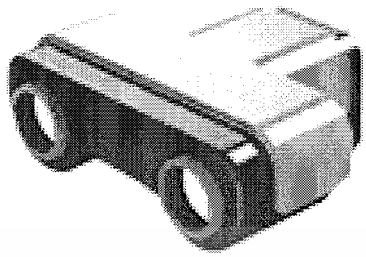
โดยให้ผลรับออกมาเป็นเปอร์เซ็นต์ ตามตัวอย่างการในรูปที่ 2.17



รูปที่ 2.17 ทดสอบความสามารถของอุปกรณ์เซ็นเซอร์ด้วยแสง (Lego Group. 2007 : http://mindstorms.lego.com/Overview/Light_Sensor.aspx)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
04761

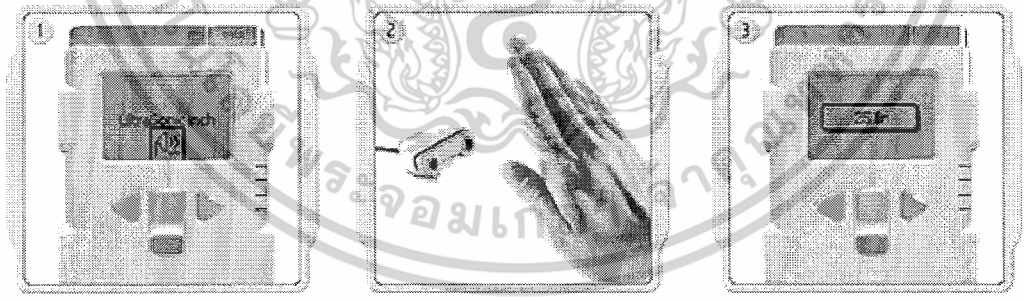
2.5.2.2 อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ (Ultrasonic Sensor)



รูปที่ 2.18 อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ (Lego Group. 2007 : http://mindstorms.lego.com/ Overview/ Ultrasonic_Sensor.aspx)

จากรูปที่ 2.18 เป็นอุปกรณ์ที่ช่วยในการมองเห็นของหุ่นยนต์ โดยใช้คลื่นความถี่ที่สูงกว่ามนุษย์จะได้ยินยิงไปยังเป้าหมายเพื่อรอการสะท้อนกลับของสัญญาณที่ปล่อยออกไป แล้วนำค่าที่ได้มาคำนวณหาระยะทางอีกทีหนึ่ง ช่วยในการวัดระยะห่างและการตรวจสอบการเคลื่อนไหวของหุ่นยนต์ (Lego Group. 2007 : http://mindstorms.lego.com/Overview/ Ultrasonic_Sensor.aspx)

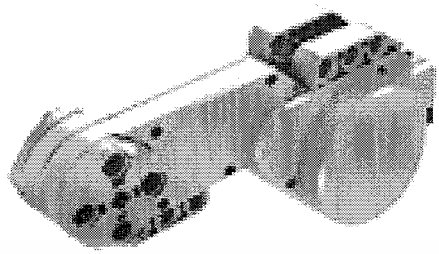
อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ให้ผลลัพธ์อยู่ในรูปหน่วยเซนติเมตร (Centimeters) โดยค่าจะอยู่ระหว่าง 0 ถึง 255 เซนติเมตร ความแม่นยำอยู่ที่ +/- 3 เซนติเมตร (Lego Group. 2007 : http://mindstorms.lego.com/Overview/ Ultrasonic_Sensor.aspx) ตัวอย่างการทดสอบดังรูปที่ 2.19



รูปที่ 2.19 ทดสอบความสามารถของอุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ (Lego Group. 2007 : http://mindstorms.lego.com/Overview/ Ultrasonic_Sensor.aspx)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2.3 กลไกมอเตอร์ (Servo Motors)



รูปที่ 2.20 มอเตอร์สำหรับควบคุมกลไกต่างๆ (Lego Group. 2007 : http://mindstorms.lego.com/ Overview/ Interactive_Servo_Motors.aspx)

จากรูปที่ 2.20 เป็นอุปกรณ์ที่ทำให้หุ่นยนต์มีความสามารถในการเดิน ประกอบด้วยมอเตอร์ 3 ตัว โดยมีมอเตอร์ 2 ตัว ทำงานพร้อมกัน (Synchronize) ทำให้หุ่นยนต์สามารถเดินเป็นเส้นตรงได้ มอเตอร์แต่ละตัวมีเซ็นเซอร์สำหรับการหมุน(Rotation) ของแกนมอเตอร์เพื่อให้หุ่นยนต์เคลื่อนที่ได้อย่างแม่นยำไม่เกิน +/- 1 องศา (Degree) โดยการหมุนของแกนมอเตอร์ 1 รอบ มีค่าเท่ากับ 360 องศา (Lego Group. 2007 : http://mindstorms.lego.com/Overview/Interactive_Servo_Motors.aspx)

โดยมอเตอร์แต่ละตัวสามารถปรับค่าความเร็วในการหมุนมอเตอร์ได้ ขึ้นอยู่กับการใส่ค่าศักย์ไฟฟ้าตามโปรแกรมที่พัฒนาขึ้นมาได้ (Lego Group. 2007 : http://mindstorms.lego.com /Overview/Interactive_Servo_Motors.aspx) ดังรูปที่ 2.21

รูปที่ 2.21 ทดสอบความสามารถของกลไกมอเตอร์ (Lego Group. 2007 : http://mindstorms.lego.com/Overview/Interactive_Servo_Motors.aspx)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 เอ็นเอ็กซ์ทีซอฟต์แวร์ (NXT Software)

ในการพัฒนาระบบจำเป็นต้องมีเครื่องมือในการพัฒนาซึ่งซอฟต์แวร์ที่ใช้ในการพัฒนาประกอบด้วยชุดคำสั่งที่ใช้ควบคุมหุ่นยนต์โดยเฉพาะ โดยเครื่องมือที่ใช้ในการพัฒนามีดังนี้

2.5.1 เอ็นเอ็กซ์เจ (NXJ)

เอ็นเอ็กซ์เจเป็นชุดคำสั่งในการพัฒนาชุดทดลองเอ็นเอ็กซ์ที ซึ่งมีพื้นฐานจากภาษาจาวา (Java Language) รองรับการพัฒนาโปรแกรมเชิงวัตถุ (Object Oriented Programming) จัดความลำดับความสำคัญของงาน (Preemptive Threads) การใช้งานอาร์เรย์หลายมิติ (Multi-Dimensional Arrays) การแจ้งข้อผิดพลาด (Exceptions) และมีเอพีไอหรือส่วนต่อประสานโปรแกรม (API: Application Programming Interface) สำหรับใช้ในการพัฒนาโปรแกรม สามารถทำงานบนทุกระบบปฏิบัติการ พัฒนาโดยเลจอส (LeJOS) (LeJOS. 2006 : http://lejos.sourceforge.net)

2.5.2 อีคลิพซ์ (Eclipse)

อีคลิพซ์เป็นไอดีอีหรือสิ่งแวดล้อมสำหรับการพัฒนาแบบเบ็ดเสร็จ (IDE: Integrated development environment) ที่ใช้ในการพัฒนาภาษาจาวาโดยเฉพาะ ซึ่งในการพัฒนาชุดทดลองเอ็นเอ็กซ์ทีจะต้องมีการนำเข้า (Import) ของชุดคำสั่งเอ็นเอ็กซ์เจเพื่อพัฒนาออกแบบโปรแกรม

2.5.3 สิ่งแวดล้อมการพัฒนาแบบครอสส์ (Cross development Environment)

สิ่งแวดล้อมการพัฒนาแบบครอสส์เป็นการพัฒนาโปรแกรมที่ใช้งานกับสมองกลฝังตัว โดยมีการเขียนโปรแกรม (Code) คอมไพล์ (Compile) แก้จุดบกพร่อง (Debug) โปรแกรมบนเครื่องคอมพิวเตอร์ (Computer) ก่อน ซึ่งจะได้แฟ้มข้อมูล (File) นามสกุลคลาส (Class) แล้วจึงเชื่อมโยง (Link) ส่งข้อมูลแฟ้มข้อมูลไปยังสมองกลฝังตัว โดยที่ฝั่งของสมองกลฝังจะเป็นตัวดำเนินงาน (Run) แฟ้มข้อมูลที่รับเข้ามาเพื่อประมวลผลทำตามคำสั่งอีกทีหนึ่ง

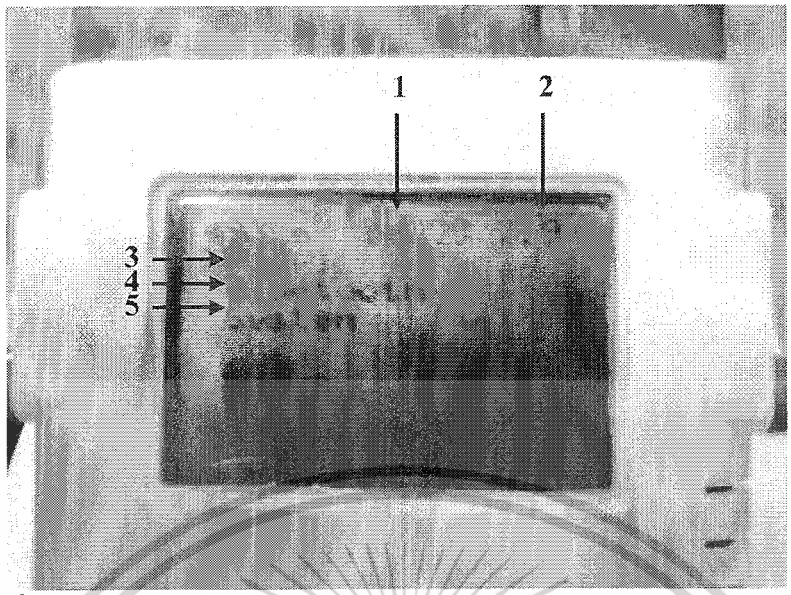
2.5.4 โครงสร้างระบบปฏิบัติการของเอ็นเอ็กซ์ที

กล่าวถึงลักษณะฟังก์ชันการทำงานต่างๆ บนระบบปฏิบัติการของเอ็นเอ็กซ์ที

2.5.4.1 ฟังก์ชันการทำงานบนระบบปฏิบัติการของเอ็นเอ็กซ์ที

รายละเอียดรายการเลือกต่างๆ บนหน่วยแสดงผลเอ็นเอ็กซ์ที ประกอบด้วย ชื่อระบบปฏิบัติการที่ใช้ (หมายเลข 1) ระดับพลังงานของแบตเตอรี่ (หมายเลข 2) ฟังก์ชันแฟ้มข้อมูล (หมายเลข 3) ฟังก์ชันบลูทูธ (หมายเลข 4) และฟังก์ชันระบบ (หมายเลข 5) ดังรูปที่ 2.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 รายละเอียดรายการเลือกต่างๆ บนหน่วยแสดงผลของเอ็นเอ็กซ์ที

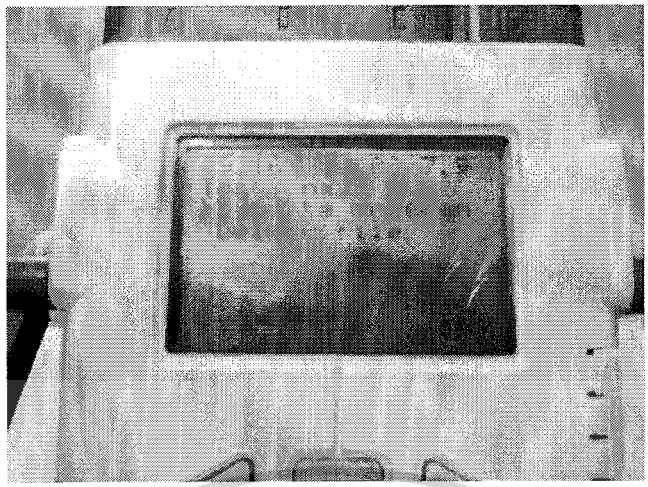
1. ฟังก์ชันแฟ้มข้อมูล

ฟังก์ชันแฟ้มข้อมูลเป็นส่วนที่ใช้แสดงแฟ้มข้อมูลที่เก็บไว้ในหน่วยความจำเอ็นเอ็กซ์ที ดังรูปที่ 2.23

รูปที่ 2.23 แสดงรายละเอียดข้อมูลของฟังก์ชันแฟ้มข้อมูล

เมื่อเลือกแฟ้มข้อมูลที่จะดำเนินงานแล้ว จะมีรายการเลือกอีก 2 รายการ คือ กระทำการกับโปรแกรม (Execute Program) และลบแฟ้มข้อมูล (Delete File) ดังรูปที่ 2.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 รายการเลือกการกระทำการกับโปรแกรม และลบแฟ้มข้อมูล

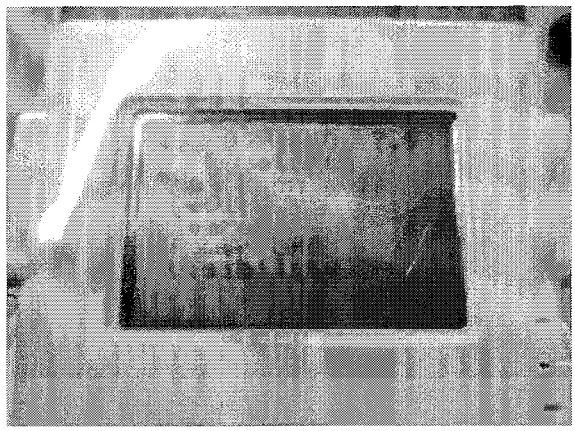
2. ฟังก์ชันบลูทูธ

ฟังก์ชันบลูทูธเป็นฟังก์ชันในการติดต่อสื่อสารไร้สายระหว่างตัวเอ็นเอ็กซ์ทีกับอุปกรณ์คอมพิวเตอร์หรือตัวเอ็นเอ็กซ์ทีด้วยกันเอง ดังรูปที่ 2.25

รูปที่ 2.25 ฟังก์ชันการทำงานบลูทูธ

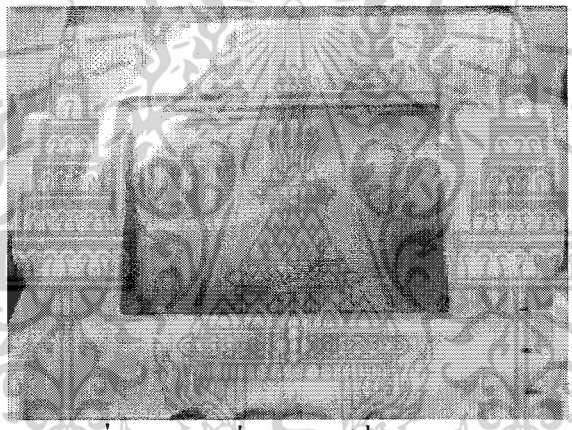
เมื่อทำการเซตกำลังเป็นออน (Power On) ฟังก์ชันบลูทธจะแสดงสถานะการทำงานของบลูทูธ รายชื่ออุปกรณ์ที่มาติดต่อด้วย (Devices) ค้นหาอุปกรณ์ไร้สาย (Search) การเซตกำลังเป็นปิด (Power Off) และการซ่อนตัวของอุปกรณ์ (Invisible) ดังรูปที่ 2.26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.26 ฟังก์ชันการทำงานบลูทูธเมื่อทำการเซตกำลังเป็นออน

เมื่อเข้ารายการเลือกรายชื่ออุปกรณ์ที่มาติดต่อด้วยดังรูปที่ 2.27



รูปที่ 2.27 รายชื่ออุปกรณ์ที่มาติดต่อด้วย

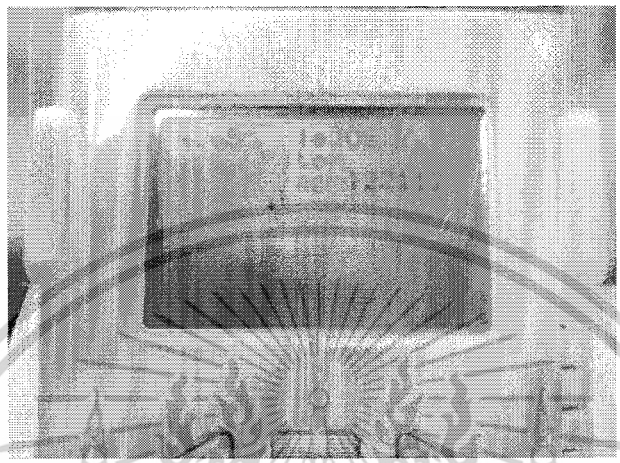
เมื่อเลือกชื่ออุปกรณ์ที่มาติดต่อด้วยจะแสดงรายละเอียดของอุปกรณ์นั้นๆ ดังรูปที่ 2.28

รูปที่ 2.28 รายละเอียดของรายชื่ออุปกรณ์ที่มาติดต่อด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ฟังก์ชันระบบ

ฟังก์ชันระบบเป็นฟังก์ชันสำหรับจัดรูปแบบ (Format) หรือลบข้อมูลในหน่วยความจำเอ็นเอ็กซ์ทีทั้งหมด ดังรูปที่ 2.29



รูปที่ 2.29 ฟังก์ชันสำหรับจัดรูปแบบหรือลบข้อมูลในหน่วยความจำเอ็นเอ็กซ์ทีทั้งหมด

2.6 แนวทางในการพัฒนาระบบ

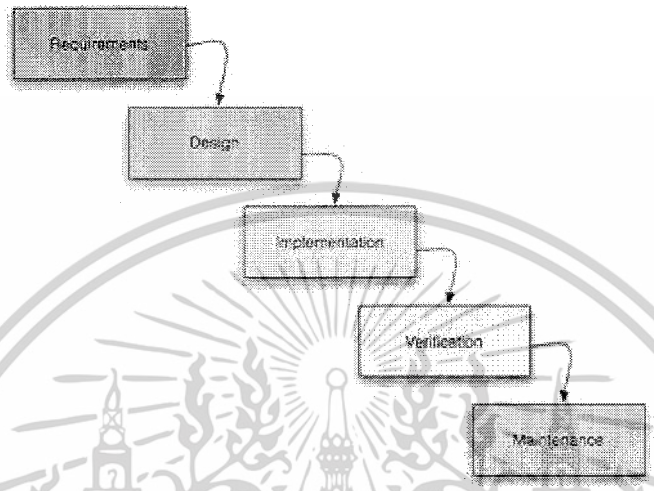
ปัจจุบันปัญหาของการพัฒนาซอฟท์แวร์ มีหลายอย่าง เช่น งานไม่เสร็จตามเวลาที่กำหนด ทำให้สูญเสียค่าใช้จ่ายเกินค่าล่าช้า งานที่ได้ไม่มีความน่าเชื่อถือ เกิดข้อผิดพลาดขึ้น และไม่สามารถสร้างความพึงพอใจให้กับลูกค้าได้ ซึ่งสามารถแก้ไขได้โดยใช้กระบวนการในการพัฒนาตามหลักการในการพัฒนาต่างๆ เพื่อให้ได้แนวทางในการพัฒนาที่มีประสิทธิภาพ

ระบบสมองกลฝังตัวนั้นเป็นระบบที่ซับซ้อน มีความหลากหลาย ในด้านฟังก์ชันการทำงาน และเป็นระบบที่ฮาร์ดแวร์ และซอฟต์แวร์ต้องทำงานร่วมกัน ถ้าไม่มีการวางแผนการที่ดี การพัฒนาที่มีประสิทธิภาพก็จะไม่เกิดขึ้น ทำให้พบปัญหาต่างๆ ตามมา และยากต่อการแก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1 วงจรการพัฒนาระบบ (SDLC: System Development Life Cycle)

วงจรการพัฒนาระบบเป็นกระบวนการที่ใช้ในการพัฒนาระบบสารสนเทศ โดยจะแสดงถึงกิจกรรมต่างๆ เป็นขั้นตอน ซึ่งจะแบ่งเป็นขั้นต่างๆได้ดังรูปที่ 2.30



รูปที่ 2.30 แสดงขั้นตอนของเอสดีแอลซี (Storm Shadow Software. 2004 : http://stormshadowsoftware.co.uk/350px-Waterfall_model.png)

ลำดับขั้นตอนการพัฒนามีดังต่อไปนี้

1. การกำหนดและเลือกระบบที่จะพัฒนา (Identification & Selection) คือการวิเคราะห์ถึงความเป็นไปได้ จัดลำดับความสำคัญ และเลือกระบบที่จะพัฒนา

2. การวางแผน (Planning) คือการวางแผนการพัฒนา ด้านทรัพยากรต่างๆ ระยะเวลา ขอบเขตของงานที่พัฒนา ให้มีความสอดคล้องและมีความเป็นไปได้

3. การวิเคราะห์ระบบ (Analysis) คือการรวบรวมข้อมูลจากระบบงานเดิม มาเขียนเป็นโครงสร้างการปฏิบัติงานโดยใช้เครื่องมือต่างๆ เพื่อให้ง่ายต่อการทำความเข้าใจในขั้นตอนต่อๆ ไป

4. การออกแบบ (Design) คือการออกแบบรูปแบบของระบบ เช่น หน้าจอจียูไอ (GUI) หรือรายงานต่างๆ ที่ได้จากระบบที่สร้างขึ้น

5. การสร้างระบบ (Implementation) คือการเขียนโปรแกรมตามขั้นตอนที่ได้ออกแบบไว้ การนำโปรแกรมมาทดสอบหาข้อผิดพลาด เมื่อผ่านการทดสอบแล้วจึงจะนำไปติดตั้งและใช้งานจริง

6. การดูแลและบำรุงรักษาระบบ (Maintenance) เป็นขั้นตอนสุดท้ายในการพัฒนาระบบ ซึ่งการดูแลและบำรุงรักษาระบบ จะมีการปรับปรุงแก้ไขข้อบกพร่องที่เกิดขึ้นของโปรแกรม หลังจากที่ได้มีการใช้งานจริง ให้ปรแกรมสามารถทำงานมีประสิทธิภาพดียิ่งขึ้น

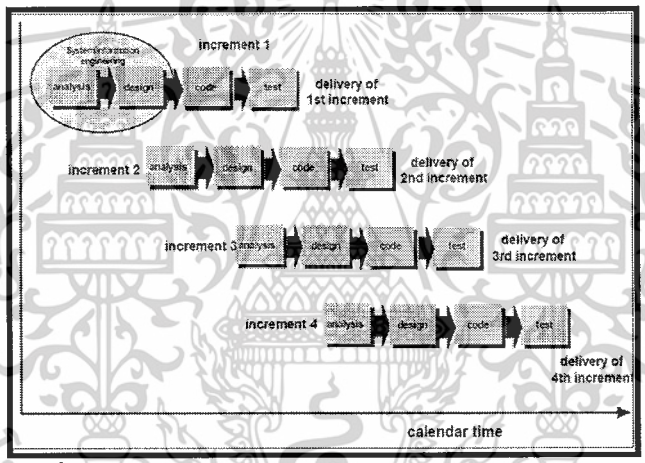
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2 แบบจำลองการพัฒนาระบบ

แบบจำลองการพัฒนาระบบเป็นแนวทางในการพัฒนาระบบ โดยแบ่งกิจกรรมออกเป็นกระบวนการต่างๆ ประกอบด้วย การเก็บความต้องการของระบบ (Requirement) การวิเคราะห์ความต้องการ การออกแบบพัฒนาระบบ และการทดสอบระบบ ซึ่งการใช้แบบจำลองจะช่วยในการวางแผน และประเมินระยะเวลาในการพัฒนาโครงงานได้

2.6.2.1 แบบจำลองเพิ่มขึ้น (Incremental Model)

แบบจำลองเพิ่มขึ้นเป็นรูปแบบการแบ่งงานเป็นงานย่อยๆ โดยทำการพัฒนาให้สำเร็จในแต่ละรอบซึ่งสามารถตรวจสอบ และติดตามความก้าวหน้าของโครงงานเป็นส่วนๆ ได้ ทำให้เกิดความเชื่อถือมากขึ้น ดังรูปที่ 2.31



รูปที่ 2.31 แสดงรูปแบบของการทำงานวนหลายๆ รอบ

2.6.3 ยูเอ็มแอล (UML: Unified Modeling Language)

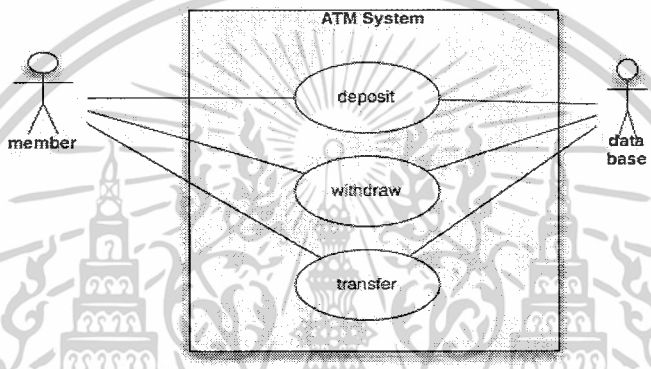
ยูเอ็มแอลเป็นรูปภาพ หรือสัญลักษณ์ (Graphical Language) ใช้เพื่อถ่ายทอดความคิดของผู้พัฒนาที่มีต่อระบบมาเป็นแผนภาพ ซึ่งประกอบไปด้วยรูปภาพหรือสัญลักษณ์ตามกฎในการสร้างแผนภาพนั้น (กิตติ ภักดีวัฒนะกุล และพนิดา พานิชกุล. 2548 : 80)

ยูเอ็มแอลเป็นภาษาสำหรับสร้างแบบจำลองของระบบที่ได้รับการพัฒนาขึ้นมาเพื่อใช้ในการวิเคราะห์ และออกแบบระบบเชิงวัตถุโดยเฉพาะ โดยมีแผนภาพ (Diagram) เป็นสิ่งที่ใช้แสดงความสัมพันธ์ของสิ่งต่างๆ ในระบบ ซึ่งช่วยในการพัฒนาให้เป็นไปอย่างมีทิศทาง ทำให้ง่ายต่อความเข้าใจ และการนำเสนอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.3.1 แผนภาพยูสเคส (Use Case Diagram)

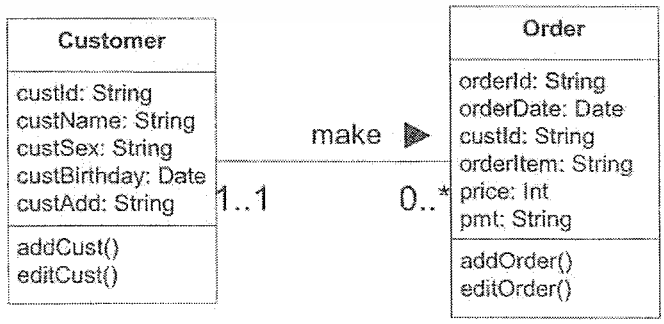
แผนภาพยูสเคสเป็นแผนภาพที่ใช้แสดงความสัมพันธ์ระหว่างผู้ที่มาติดต่อกับระบบที่พัฒนา โดยการพัฒนาแผนภาพยูสเคสจะต้องทราบถึงความต้องการของระบบทั้งหมดก่อน โดยศึกษาว่าความต้องการของระบบมีงานอะไรที่ต้องทำ และงานที่ทำนั้นต้องติดต่อกับระบบหรือผู้ใช้อย่างไร ซึ่งต้องมีการกำหนดขอบเขตที่ชัดเจน และแน่นอน ดังรูป 2.32 เป็น ตัวอย่างของแผนภาพยูสเคสที่เป็นระบบเอทีเอ็ม (ATM) แสดงให้เห็นถึง ระบบการบริการของธนาคารที่ให้บริการกับสมาชิกของธนาคาร



รูปที่ 2.32 แสดงตัวอย่างของแผนภาพยูสเคส

2.6.3.2 แผนภาพคลาส (Class Diagram)

แผนภาพคลาสเป็นแผนภาพที่ใช้ในการแสดงกลุ่มของคลาส โครงสร้างของคลาส อินเตอร์เฟส (Interface) และแสดงความสัมพันธ์ (Relationship) ระหว่างคลาส โดยคลาสจะแบ่งเป็นสามส่วน ส่วนแรกคือชื่อ ส่วนที่สองคือคุณลักษณะของคลาส (Attribute) มีลักษณะเป็นอย่างไร ส่วนที่สามคือพฤติกรรม (Behavior) ของคลาส มีความสามารถทำอะไรได้บ้าง และส่วนที่ใช้ในการแสดงถึงความสัมพันธ์ระหว่างคลาส ดังรูปที่ 2.33

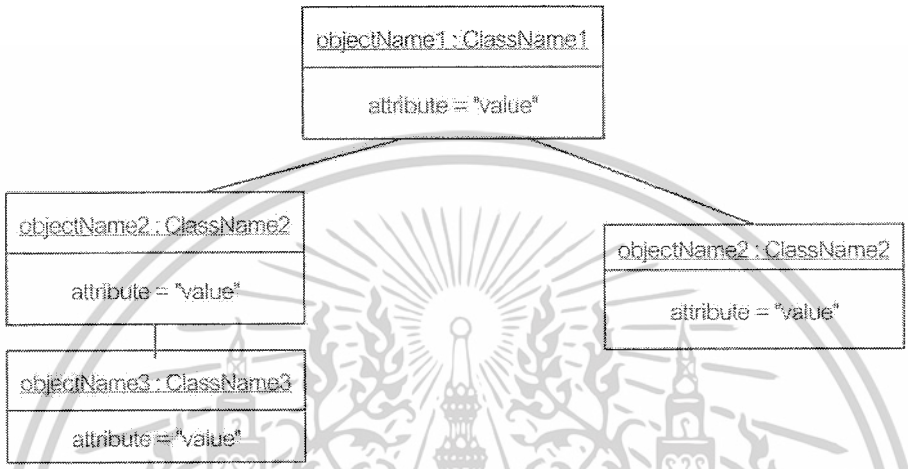


รูปที่ 2.33 ตัวอย่างของแผนภาพคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.3.3 แผนภาพวัตถุ (Object Diagram)

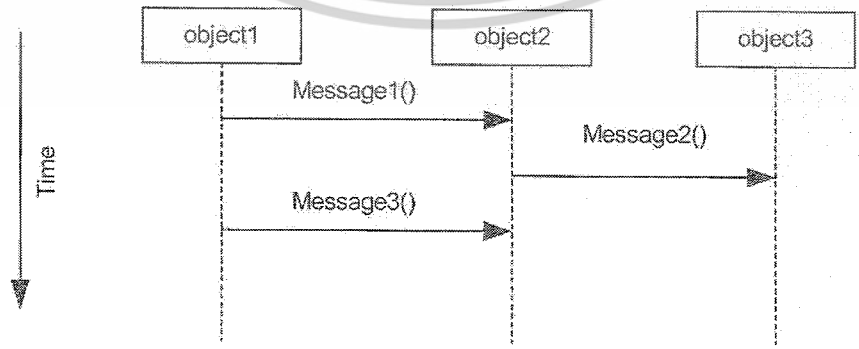
แผนภาพวัตถุเป็นแผนภาพที่แสดงถึงความสัมพันธ์ระหว่างการทำงานของอ็อบเจกต์ที่เกิดจากคลาสต่างๆ ซึ่งจะสนใจเฉพาะช่วงเวลาใดเวลาหนึ่งว่าออปเจ็กต์ใดบ้างที่ทำงานอยู่ และมีความสัมพันธ์กันอย่างไร ดังรูปที่ 2.34



รูปที่ 2.34 ตัวอย่างของแผนภาพวัตถุ

2.6.3.4 แผนภาพลำดับเหตุการณ์ (Sequence Diagram)

แผนภาพลำดับเหตุการณ์เป็นแผนภาพที่แสดงให้เห็นถึงการปฏิสัมพันธ์ (Interaction) ระหว่างอ็อบเจกต์ โดยเน้นที่การส่งสาร (Message) ระหว่างอ็อบเจกต์ตามลำดับของเวลา (Time Sequence) ที่เกิดเหตุการณ์ขึ้นจากก่อนไปหลัง โดยจะมีสัญลักษณ์แสดงให้เห็นลำดับของการส่งสารตามเวลาส่งอย่างชัดเจน แสดงลักษณะของแผนภาพลำดับเหตุการณ์ (กิตติ ภักดีวัฒนะกุล และพนิดา พานิชกุล. 2548 : 80) ดังรูปที่ 2.35

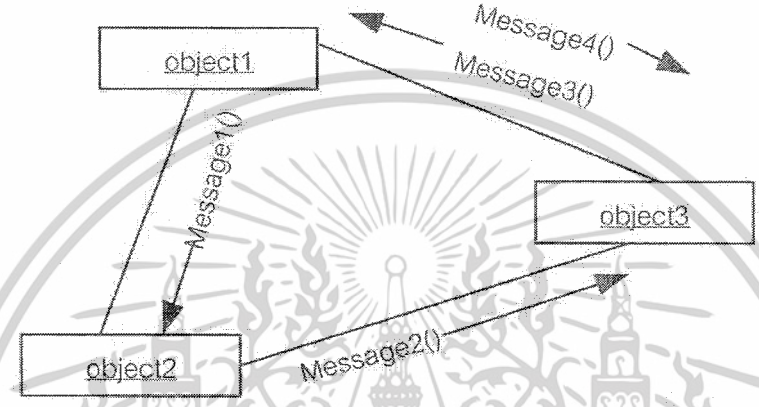


รูปที่ 2.35 แสดงตัวอย่างแผนภาพลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.3.5 แผนภาพปฏิสัมพันธ์ (Collaboration Diagram)

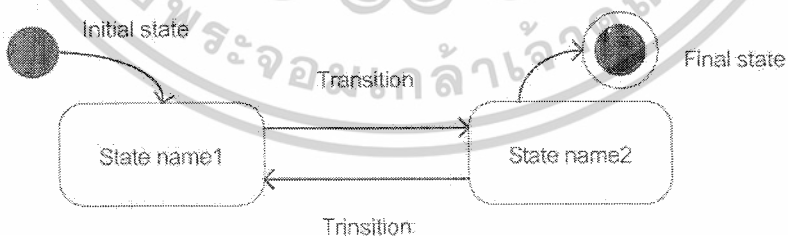
แผนภาพปฏิสัมพันธ์เป็นแผนภาพที่แสดงให้เห็นถึงการปฏิสัมพันธ์ระหว่างอ็อบเจ็กต์กับแผนภาพลำดับเหตุการณ์ ในส่วนของมุมมองแผนภาพปฏิสัมพันธ์จะเน้นความสัมพันธ์เชิงโครงสร้างมากกว่าแผนภาพลำดับเหตุการณ์ ที่มองลำดับการทำงานของออปเจ็กต์เป็นหลัก โดยทั้งสองแผนภาพคือสิ่งเดียวกัน แต่มีมุมมองที่ต่างกัน ดังรูปที่ 2.36



รูปที่ 2.36 ตัวอย่างของแผนภาพปฏิสัมพันธ์ (กิตติ ภักดีวัฒนะกุล และพนิดา พานิชกุล. 2548 : 269)

2.6.3.6 แผนภาพสถานะ (State Chart Diagram)

แผนภาพสถานะเป็นแผนภาพที่แสดงถึงสถานะของออปเจ็กต์นั้นๆ ในเวลาใดเวลาหนึ่ง โดยจะเน้นมุมมองเชิงโครงสร้างของระบบ เมื่อถูกเปลี่ยนสถานะแล้วจะทำงานอะไรต่อไป ซึ่งการเปลี่ยนสถานะนั้นจำเป็นต้องมีสารเป็นตัวกระตุ้นให้เปลี่ยนสถานะ ดังรูปที่ 2.37



รูปที่ 2.37 แสดงตัวอย่างแผนภาพสถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.3.7 แผนภาพกิจกรรม (Activity Diagram)

แผนภาพกิจกรรมเป็นแผนภาพที่แสดงถึงกิจกรรมที่เกิดขึ้นในออปเจ็กต์นั้น มีลักษณะคล้ายกับโพลวชาร์ต (Flow Chart) (กิตติ ภักดีวัฒนะกุล และพนิดา พานิชกุล. 2548 : 272)

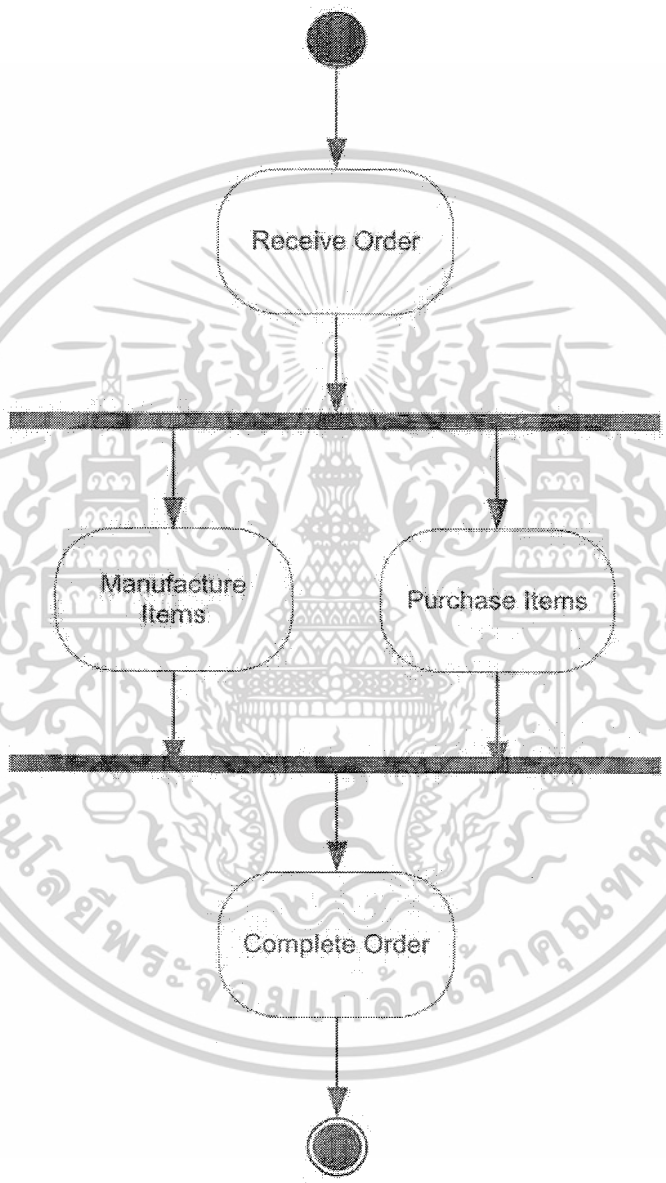
มีสามแบบคือแบบแรกมีทางเลือกในการตัดสินใจ แบบที่สองโพลวชาร์ต และแบบที่สามทำงานพร้อมกัน ซึ่งแบบที่มีการแสดงถึงความสัมพันธ์ระหว่างออปเจ็กต์คือแบบสวิมเลนส์ (Swimlanes) แบบแรกเป็นแบบมีทางเลือกในการตัดสินใจดังรูปที่ 2.38

รูปที่ 2.38 แสดงแผนภาพกิจกรรมแบบมีทางเลือกตัดสินใจ (กิตติ ภักดีวัฒนะกุล และพนิดา พานิชกุล. 2548 : 272)

โดยมีทางเลือกให้ไปทางได้ทางหนึ่งด้วยเงื่อนไขที่สร้างขึ้นมา ก่อนที่จะกลับมาทำงานหลักอีกครั้งหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

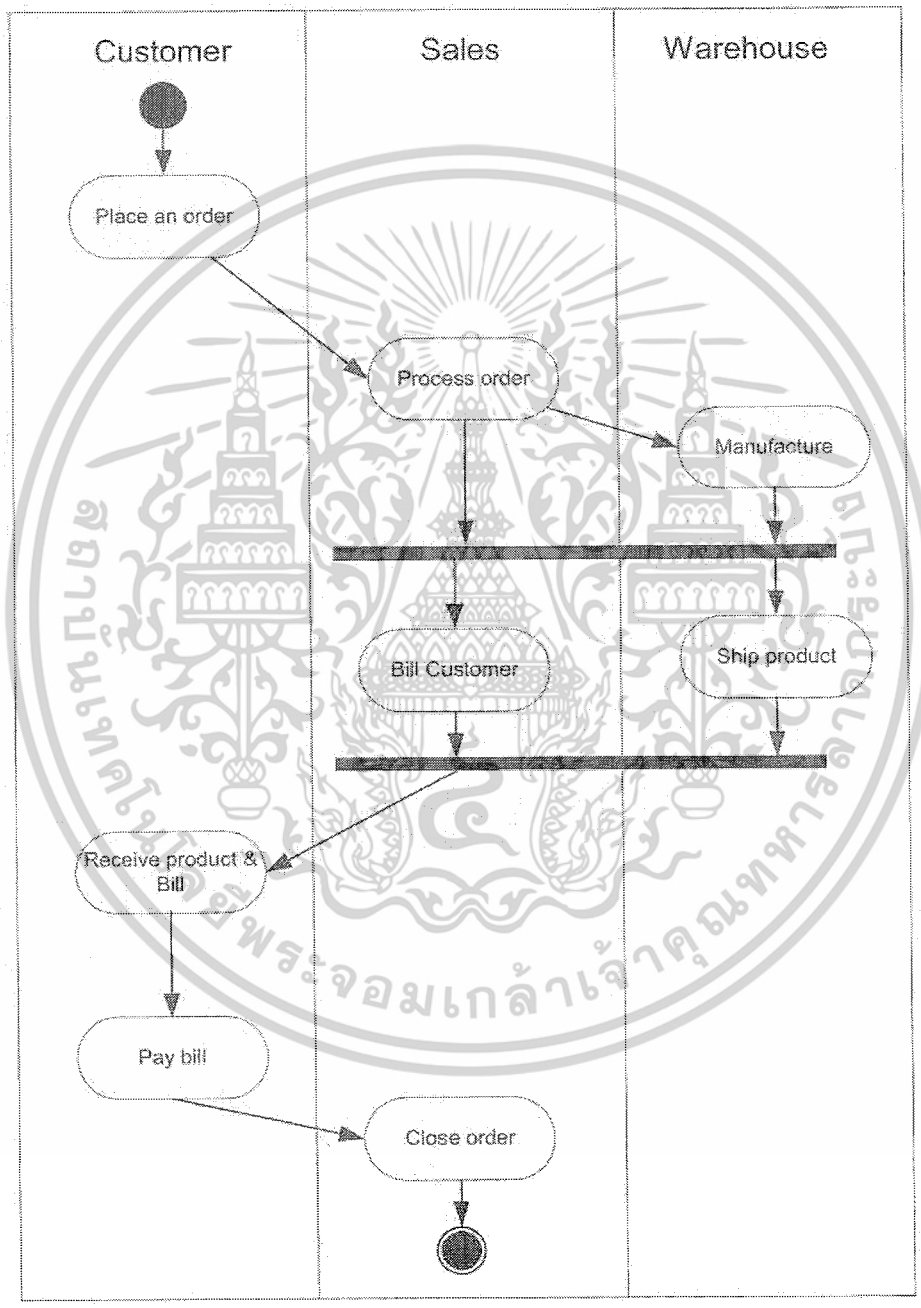
แบบที่สองคือการทำงานพร้อมกัน ซึ่งสำหรับลักษณะการทำงานพร้อมกันจะเรียกว่า ทรานซิชั่นฟอร์ค (Transition Fork) เมื่อถึงจุดเปลี่ยนแยก โดยจะมีเส้นตรงลากขวางแนวนอนก่อนเข้าสู่กิจกรรมที่ต้องกระทำพร้อมกัน เมื่อเสร็จสิ้นก็มีเส้นตรงลากขวางแนวนอนอีกเส้น เพื่อเป็นจุดรวมทิศทางของกิจกรรมถือว่าจบสิ้นกระบวนการแบบที่สอง ดังรูปที่ 2.39



รูปที่ 2.39 แสดงแผนภาพกิจกรรมแบบมีการทำงานพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบที่สามคือแบบแบ่งส่วนด้วยสวิมเลนเป็นการแบ่งส่วนการทำงานออกเป็นหลายๆ ส่วน ที่เกี่ยวข้องกันใน 1 กิจกรรม โดยจะแบ่งเป็นแนวตั้งคล้ายกับสวิมเลนส์ ซึ่งแสดงถึงการติดต่อกัน ระหว่างออปเจ็กต์ได้ด้วย ดังรูปที่ 2.40



รูปที่ 2.40 การแบ่งส่วนด้วยสวิมเลนส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การพัฒนาระบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ

ในบทนี้จะอธิบายถึงการประยุกต์กระบวนการพัฒนาระบบเพื่อใช้ในการพัฒนาหุ่นยนต์ตรวจจับ และจัดการวัตถุ โดยดำเนินการตามทฤษฎีที่ใช้ในการออกแบบ และพัฒนาระบบ โดยเริ่มจากการศึกษาปัญหา และความต้องการของระบบ เพื่อนำมาวิเคราะห์ความต้องการในการออกแบบทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์ (Software) และการพัฒนาโปรแกรมที่ใช้จริง โดยมีรายละเอียดในการพัฒนาดังนี้

3.1 ความต้องการของระบบ

เป็นสิ่งสำคัญในการในการออกแบบ และพัฒนาในกระบวนการขั้นต่อไป ดังนั้นจะต้องมีการกำหนดความต้องที่ชัดเจน ตามลักษณะหน้าที่การทำงานของระบบ

3.1.1 หน้าที่ของระบบ

1. หุ่นยนต์สามารถเคลื่อนที่เข้าไปหาวัตถุที่อยู่ในสนามได้
2. หุ่นยนต์สามารถตรวจสอบ และแยกแยะสีของลูกบอลได้ โดยในสนามทดสอบจะมีลูกบอลสีอื่นๆ ประกอบอยู่ด้วย ดังนั้นจะต้องผลักลูกบอลให้ถูกสีตามที่กำหนดได้
3. หุ่นยนต์สามารถผลักลูกบอลได้
4. หุ่นยนต์มีขอบเขตในการติดตาม

3.1.2 ข้อกำหนดของการทดสอบ

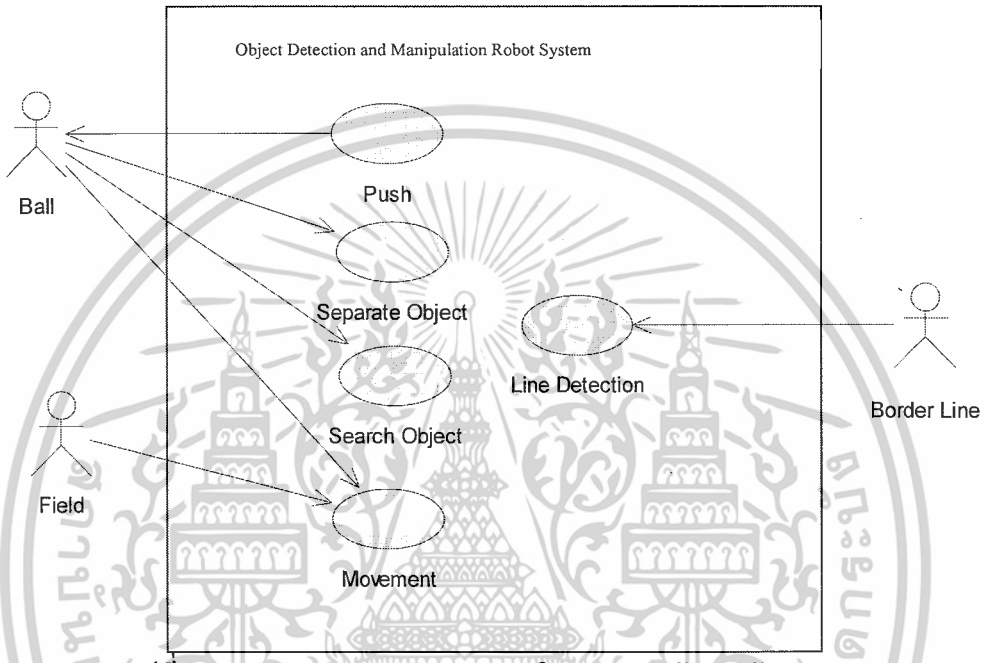
1. มีขอบเขตของสนาม
2. มีลูกบอลสีอื่นที่ไม่ใช่สีเดียวกับลูกบอลที่ผลักออกไป
3. สนามมีสีขาว และเส้นขอบสนามมีสีดำ
4. ลูกบอลหรือวัตถุมีสีน้ำเงิน และ สีแดง อย่างละ 1ลูก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบระบบ

3.2.1 แผนภาพยูสเคส (Use Case Diagram)

การออกแบบแผนภาพยูสเคสจะแสดงถึงความสัมพันธ์ (Relationship) และขอบเขตงานของระบบ (Boundary) ดังรูปที่ 3.1



รูปที่ 3.1 แผนภาพยูสเคสของระบบติดตาม และจัดการวัตถุ

จากรูปที่ 3.1 แผนภาพยูสเคสของระบบประกอบด้วย ผู้กระทำ (Actor) ทั้งหมด 3 ตัว คือ ลูกบอล สีของพื้นสนาม และเส้นขอบสนาม ส่วนหน้าที่ของระบบประกอบไปด้วย การเคลื่อนที่ที่มีความสัมพันธ์กับสีของพื้นสนาม และลูกบอล การผลักบอล การแยกแยะวัตถุ การตรวจจับเส้นขอบสนาม และการตรวจจับวัตถุ ทั้งหมดเป็นหน้าที่ของระบบที่จะต้องทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 ยูสเคสเดสคริพชั่น (Use Case Description)

แสดงถึงความสัมพันธ์ และรายละเอียดเกี่ยวกับแผนภาพยูสเคสของระบบ

ตารางที่ 3.1 Use Case Description – Push Object

Use Case Name: Push	ID:01	Importance Level: High
Primary Actor: Ball	Use Case Type:	
Stakeholders and Interests: -		
Brief Description: ทำหน้าที่ในการผลักลูกบอล		
Trigger: เมื่อระบบตรวจพบเจอลูกบอล Type:		
Relationships: Association: Ball Include: Extend: Generalization		
Normal Flow of Events: รับค่าสีผิวของลูกบอล วิเคราะห์ค่าที่ได้ 3. มอเตอร์ทำการผลักลูกบอล		
Sub Flows: S-2: 1. อ่านค่าสีผิวของลูกบอลแล้วเป็นเป้าหมาย ส่งเมสเสกให้มอเตอร์ทำผลักเบา 2. อ่านค่าสีผิวของลูกบอลแล้วไม่ใช่เป้าหมาย ส่งเมสเสกให้มอเตอร์ทำผลักแรง		
Alternative /Exception Flows:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 Use Case Description – Separate Object

Use Case Name: Separate Object	ID:02	Importance Level: High
Primary Actor: Ball	Use Case Type:	
Stakeholders and Interests: -		
Brief Description: เป็นส่วนที่ใช้ในการแยกแยะสีของลูกบอล		
Trigger: เมื่ออยู่ใกล้วัตถุ Type:		
Relationships: Association: Ball Include: Extend: Generalization		
Normal Flow of Events: รับค่าสีของลูกบอลจากเซ็นเซอร์แสง ตรวจสอบสี		
Sub Flows: S-2: ไม่วัตถุให้ทำการค้นหาต่อไป		
Alternative /Exception Flows:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 Use Case Description – Search Object

Use Case Name: Search Object	ID:03	Importance Level: High
Primary Actor: Ball	Use Case Type:	
Stakeholders and Interests: -		
Brief Description: เป็นส่วนที่ใช้ในการค้นหาทิศทางลูกบอลที่ใกล้ที่สุด		
Trigger: เมื่อระบบเริ่มทำงาน, หลังจากผลักลูกบอล, ตรวจพบเส้นขอบสนาม Type:		
Relationships: Association: Ball Include: Extend: Generalization		
Normal Flow of Events: รับค่าระยะทางของวัตถุรอบด้าน หาค่าที่น้อยที่สุด เคลื่อนที่ไปยังทิศทางของค่าที่น้อยที่สุด		
Sub Flows: S-1:		
Alternative /Exception Flows:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 Use Case Description – Movement

Use Case Name: Movement	ID:04	Importance Level: High
Primary Actor: Ball, Sound	Use Case Type:	
Stakeholders and Interests: -		
Brief Description: เป็นส่วนที่ใช้ในการเคลื่อนที่ของหุ่นยนต์		
Trigger: เมื่อตรวจพบเจอวัตถุ, ได้รับค่าสัญญาณเสียง Type:		
Relationships: Association: Ball, Sound Include: Extend: Generalization		
Normal Flow of Events: รับค่าระยะของวัตถุที่ใกล้ที่สุด เคลื่อนที่เข้าไปหาวัตถุ		
Sub Flows: S-1: 1. หากค่าที่ได้รับพบว่าตรวจไม่พบวัตถุใดเลยให้หยุดการทำงาน S-3: 3. ตวรจสอบค่าระยะของวัตถุอยู่ในขอบเขตที่กำหนดไว้ ให้หยุดการเคลื่อนไหว		
Alternative /Exception Flows:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 Use Case Description – Line Detection

Use Case Name: Line Detection	ID:05	Importance Level: High
Primary Actor: Border Line	Use Case Type:	
Stakeholders and Interests: -		
Brief Description: เป็นส่วนที่ใช้ในการควบคุมไม่ให้หุ่นยนต์ออกนอกสนาม		
Trigger: เมื่อได้รับค่าของเส้นขอบสนาม Type:		
Relationships: Association: Border Line Include: Extend: Generalization		
Normal Flow of Events: รับค่าตรวจสอบเส้นขอบสนาม ตรวจสอบค่าเส้นขอบสนาม		
Sub Flows: S-2: 1. ค่าไม่ตรงกับค่าเส้นขอบสนาม ให้ทำงานต่อไป 2. ค่าตรงกับเส้นขอบสนาม ให้หาวัตถุอื่นต่อไป		
Alternative /Exception Flows:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

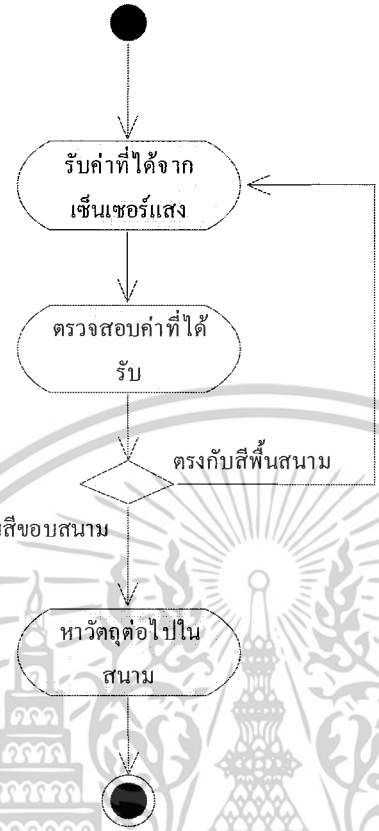
3.2.3 แผนภาพกิจกรรม (Activity Diagram)

เป็นแผนภาพแสดงกิจกรรมของระบบ โดยจะแสดงกระบวนการขั้นตอนของการทำงานของระบบแต่ละกิจกรรม ดังรูปต่อไปนี้

รูปที่ 3.2 แผนภาพแสดงกิจกรรมการค้นหาวัตถุ

จากรูปที่ 3.2 แสดงถึงกิจกรรมที่ทำในการค้นหาวัตถุ ซึ่งมีวิธีการคือ เก็บค่าระยะห่างระหว่างหุ่นยนต์กับวัตถุรอบตัวหุ่นยนต์ และนำค่าที่ได้มาหาค่าที่น้อยที่สุด จากนั้นจึงเก็บค่าอีกครั้งเพื่อเปรียบเทียบกับค่าที่น้อยที่สุดเพื่อหาทิศทางของวัตถุที่อยู่ใกล้

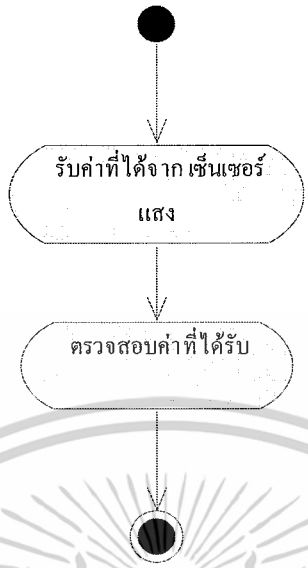
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แผนภาพแสดงกิจกรรมการตรวจสอบเส้นขอบสนาม

จากรูปที่ 3.3 แสดงกิจกรรมที่ทำในการตรวจสอบเส้นขอบสนาม โดยรับค่าแสงที่ได้จากเซ็นเซอร์แสงมาตรวจสอบว่า ค่าที่ได้รับมานั้นเป็นค่าสีของพื้นสนามหรือสีของเส้นขอบสนาม ซึ่งจะต้องทำวนซ้ำไปเรื่อยๆ จนกว่าจะจบการทำงานของระบบ

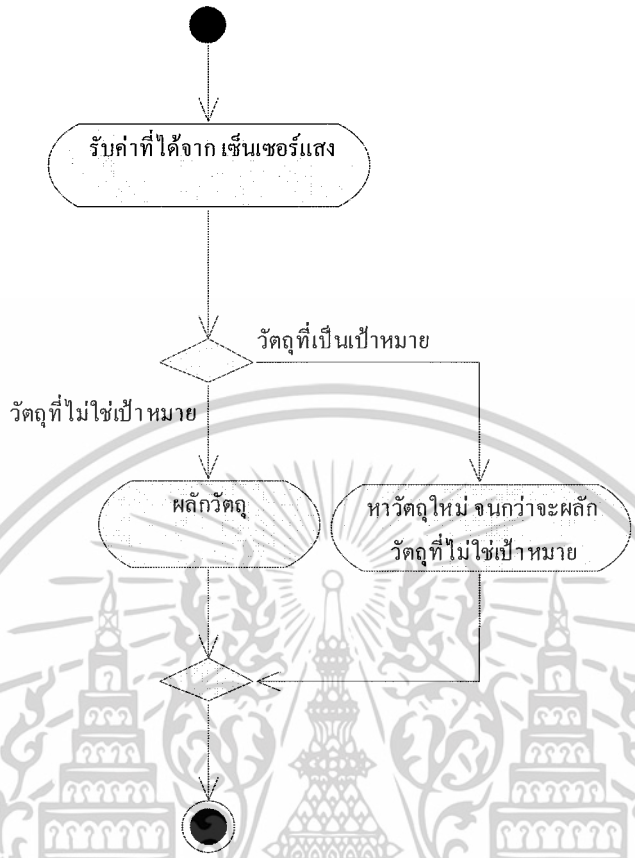
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 แผนภาพแสดงกิจกรรมการแยกแยะสีของวัตถุ

จากรูปที่ 3.4 แสดงของกิจกรรมที่ทำในการแยกแยะสีของวัตถุ โดยกระบวนการนี้จะมีหน้าที่ในการแยกแยะสีของวัตถุเท่านั้น ซึ่งผลหรือค่าที่ได้การแยกแยะสีจะถูกระบบนำมาตรวจสอบอีกที

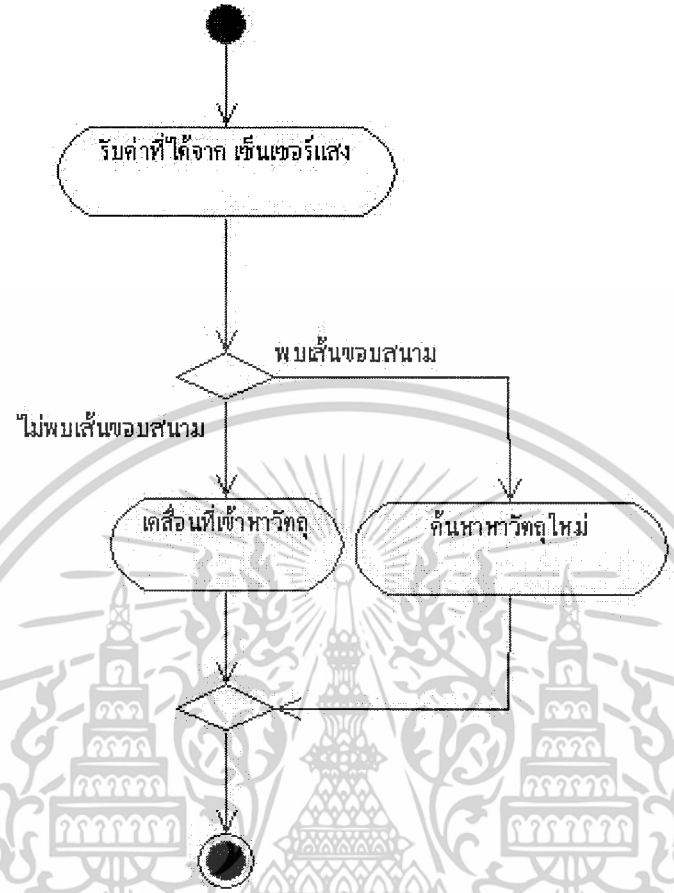
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 แผนภาพแสดงกิจกรรมการผลักวัตถุ

จากรูปที่ 3.5 แสดงกิจกรรมที่ทำในการผลักวัตถุ ซึ่งระบบจะต้องทำการแยกแยะสีของวัตถุก่อนเพื่อจะได้จัดการกับวัตถุที่อยู่ตรงหน้าได้อย่างถูกต้อง โดยใช้เซ็นเซอร์แสงในการตรวจสอบสีของวัตถุเพื่อมาแยกแยะวัตถุที่เป็นเป้าหมาย และวัตถุที่ไม่ใช่เป้าหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



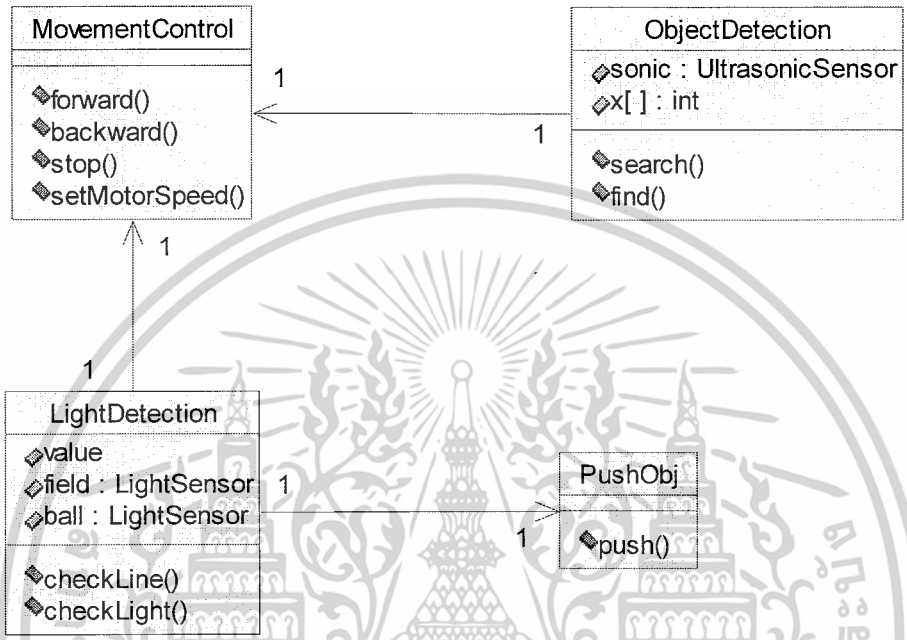
รูปที่ 3.6 แผนภาพแสดงกิจกรรมการเคลื่อนที่ของหุ่นยนต์

จากรูปที่ 3.6 แสดงกิจกรรมที่ทำในการเคลื่อนที่องหุ่นยนต์ ซึ่งระหว่างที่หุ่นยนต์เคลื่อนที่นั้นจะต้องตรวจสอบเส้นขอบสนามอยู่เสมอ เพื่อไม่ให้หุ่นยนต์วิ่งออกนอกสนามได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 แผนภาพคลาส (Class Diagram)

แสดงโครงสร้างของระบบในมุมมองเชิงวัตถุ ประกอบด้วยอินเตอร์เฟส (Interface) ความสัมพันธ์ระหว่างคลาส (Relationship) ใช้แสดงลักษณะ และรายละเอียดของแต่ละคลาส



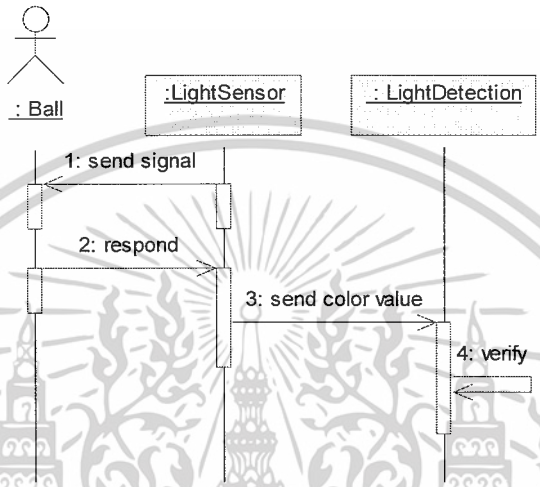
รูปที่ 3.7 แผนภาพคลาสของระบบติดตาม และจัดการวัตถุ

แผนภาพคลาสในรูปที่ 3.7 แสดงถึงโครงสร้างของระบบ ประกอบด้วย 4 คลาสหลักๆ ซึ่งแต่ละคลาส มีความสัมพันธ์แบบหนึ่งต่อหนึ่ง เท่านั้นเนื่องจากข้อจำกัดทางด้านซอฟท์แวร์ ทำให้ไม่สามารถสร้างอ็อบเจกต์ของคลาสแต่ละคลาสได้มากกว่า 1ตัว ไม่ได้กล่าวถึงในส่วนที่เป็นขั้นตอนวิธี (Algorithm) ของระบบ ซึ่งส่วนนั้นจะอยู่ในคลาสหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 แผนภาพลำดับ (Sequence Diagram)

เป็นแผนภาพที่แสดงปฏิสัมพันธ์ (Interaction) ระหว่างอ็อบเจกต์ โดยมีการส่งสาร (Message) ระหว่างอ็อบเจกต์ตามลำดับของเวลาที่เกิดเหตุการณ์ใช้แสดงถึงขั้นตอการทำงาน และลำดับของการสื่อสสารระหว่างอ็อบเจกต์ตอบโต้กัน



รูปที่ 3.8 แผนภาพลำดับการแยกแยะสี

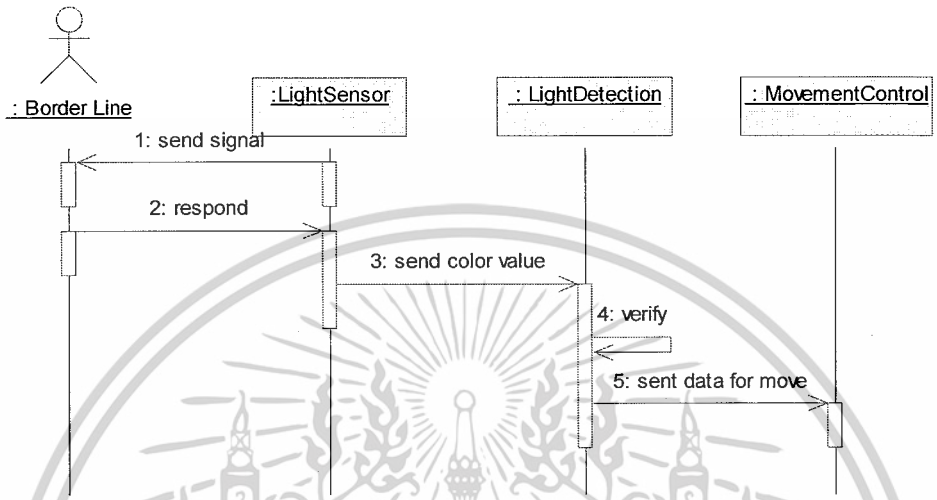
จากรูป 3.8 เป็นขั้นตอนของการแยกแยะวัตถุ ซึ่งจะต้องใช้อุปกรณ์เซนเซอร์ด้วยแสงเพื่อรับค่าสีของวัตถุมา และตัดสินว่าเป็นวัตถุที่เราต้องการหรือไม่ ซึ่งหลังจากตัดสินใจแล้วค่อยนำผลที่ได้ไปเข้ากระบวนการทำงานอื่นต่อ อาทิเช่น ผลักลูกบอล หรือหาวัตถุอื่นต่อไปซึ่งยังไม่สามารถระบุแน่นอนได้

ตัวอย่างโปรแกรมที่ 1

```
public boolean ObjCheck(){
        x = y.readValue();
        if (x>44){
                ball = true; }
        else{
                ball = false; }
        return ball;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่างที่ 1 เมื่อระบบเรียกใช้ฟังก์ชัน ObjCheck() ฟังก์ชันนี้จะ รับค่าสีของวัตถุจากเซ็นเซอร์แสงแล้วนำมาประมวลผลซึ่งผลของสีรับเข้ามาจะมีผลต่อค่าตัวแปร ball โดยตรง ซึ่งระบบจะส่งค่า ball กลับไปที่โปรแกรมหลัก เพื่อทำงานต่อไป



รูปที่ 3.9 แผนภาพลำดับการตรวจสอบเส้นขอบสนาม

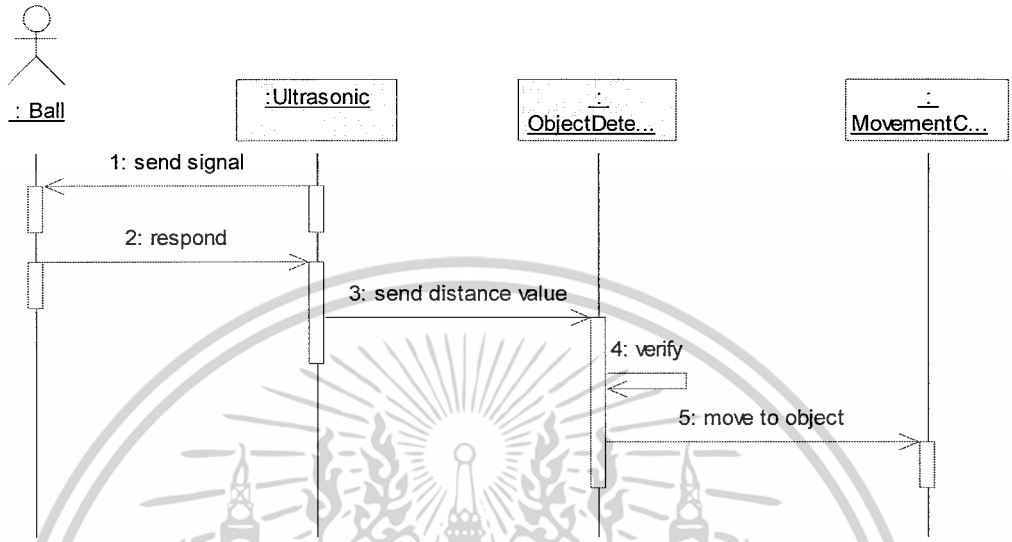
จากรูปที่ 3.9 เป็นขั้นตอนการตรวจจับเส้นขอบสนาม ซึ่งจะมีผลต่อการเคลื่อนไหวของหุ่นยนต์ โดยจะใช้อุปกรณ์เซนเซอร์ด้วยแสง ตรวจจับสีของพื้นสนาม ก่อนที่จะนำมาประมวลผลว่าใช่หรือไม่ใช่ขอบสนาม

ตัวอย่างโปรแกรมที่ 2

```
while(l1.lineCheck()){
        m.forward1();
         }
        m.backward(900);
        try {
                Thread.sleep(300);
                m.TurnLeft(200);
                l1.line=true;
        } catch (InterruptedException e) { }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่างโปรแกรมที่ 2 เป็นการตรวจสอบค่าของสนาม เมื่อค่าที่ได้นั้นเป็นค่าของสีพื้นสนามหุ่นยนต์จะเดินหน้าต่อไป



รูปที่ 3.10 แผนภาพลำดับการเคลื่อนที่

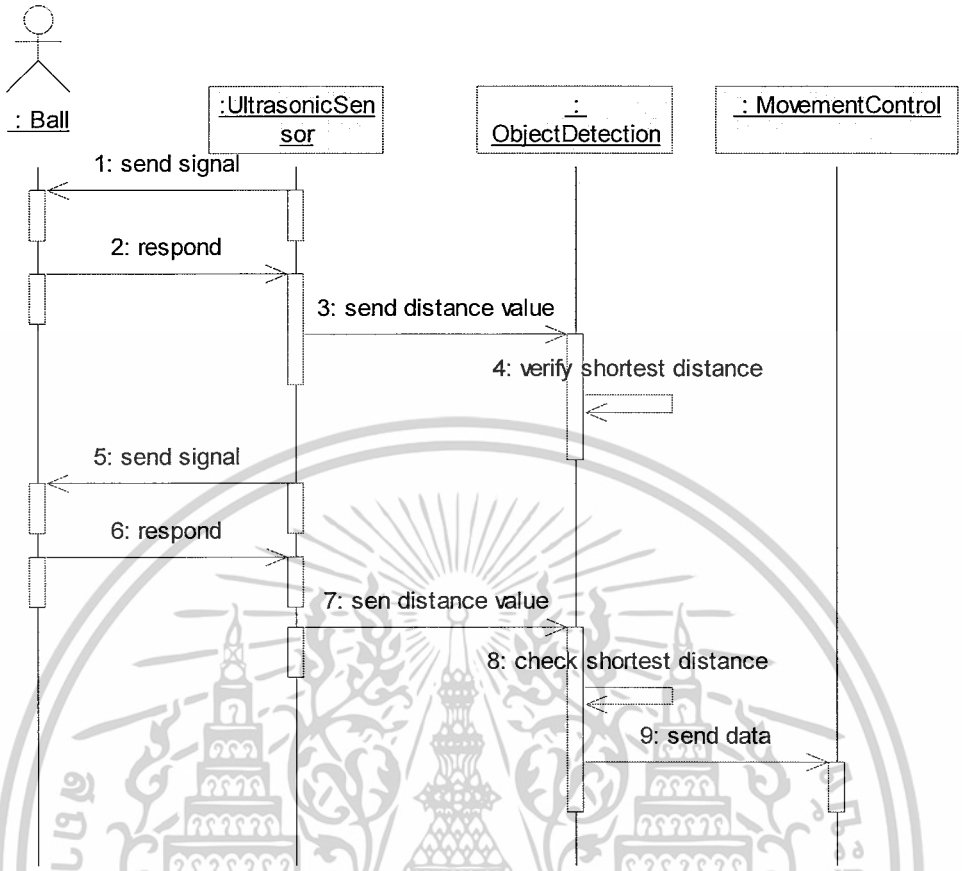
จากรูปที่ 3.10 เป็นขั้นตอนการเคลื่อนที่เข้าหาวัตถุ ซึ่งจะมีการปรับความเร็วในการเคลื่อนที่ให้เหมาะสมกับระยะทาง โดยใช้อุปกรณ์เซนเซอร์ด้วยความถี่ ตรวจจับระยะห่างที่เหลือ และนำค่าที่ได้มาปรับความเร็วให้มอเตอร์ที่ใช้ในการขับเคลื่อน

ตัวอย่างโปรแกรมที่ 3

```
c= dc.getDistance();
if(c>=dc.x[0]){
Motor.A.setSpeed(200);
Motor.C.setSpeed(200);
 Motor.C.forward();
}
```

จากตัวอย่างโปรแกรมที่ 3 ตัวแปร C จะได้รับค่าที่ได้จากฟังก์ชัน getDistance ในคลาส DetectObject ซึ่งค่าที่ได้นั้นจะนำมาเปรียบเทียบกับค่าที่จับวัตถุ หากค่าสูงกว่า จะเดินเข้าหาวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 แผนภาพลำดับการค้นหาวัตถุ

จากรูปที่ 3.11 เป็นลำดับความสัมพันธ์ ของการค้นหาวัตถุ ซึ่งจะต้องหาระยะของวัตถุทั้งหมดรอบตัว และนำค่าที่ได้มาหาค่าน้อยที่สุดเพื่อให้รู้ว่าระยะห่างของวัตถุที่อยู่ใกล้ที่สุดเป็นเท่าใด จากนั้นหุ่นยนต์จะหมุนเพื่อวัดระยะทางเพื่อเปรียบเทียบกับค่าที่น้อยที่สุดที่ได้เพื่อรู้ว่าวัตถุที่อยู่ใกล้ที่สุดนั้นอยู่ในทิศทางใดของหุ่นยนต์ เมื่อจบกระบวนการนี้หุ่นยนต์จะต้องหันไปหาวัตถุที่อยู่ใกล้ที่สุด

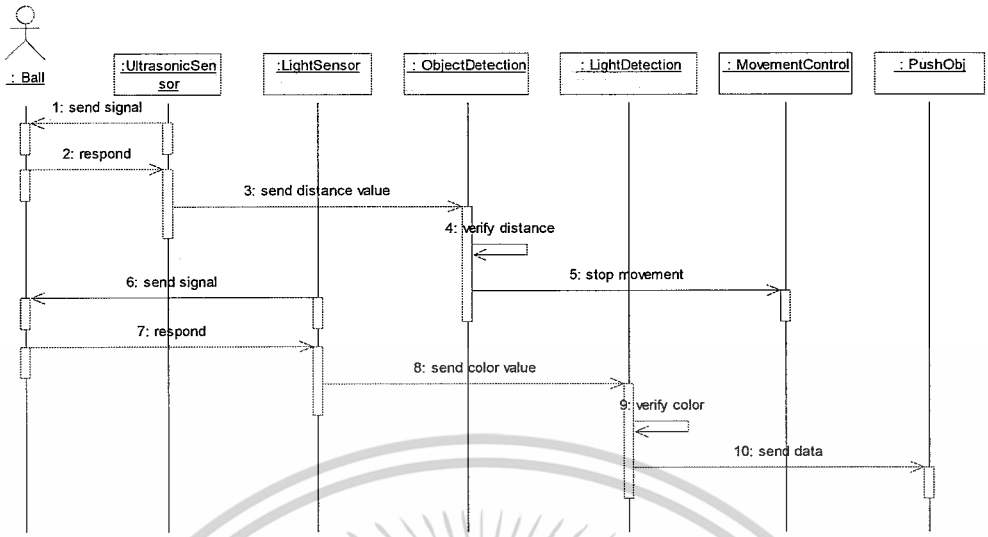
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างโปรแกรมที่ 4

```
while(r<50){
        x[r]=sonic.getDistance();
        Motor.A.rotate(-10);
        r++;
}
for (int i=0;i<50;i++) {
        for (int j=50-1;j>i;j--) {
                 if (x[j-1] > x[j]) {
                tmp = x[j];
                x[j] = x[j-1];
                x[j-1] = tmp;
                    }
         }
 }
for(int r=0;r<50;r++){
        compare=sonic.getDistance();
        Motor.A.rotate(10);
        if(compare-x[0]<=3){
                Motor.A.rotate(40);
                break;
        }
```

จากตัวอย่างโปรแกรมที่ 4 หุ่นยนต์จะหมุนเพื่อเก็บค่าระยะห่างระหว่างหุ่นยนต์กับวัตถุ และนำมาเรียงค่าเพื่อหาค่าที่ต่ำสุดและกระบวนการสุดท้ายคือหาทิศทางที่วัตถุอยู่ใกล้ที่สุด จึงจบการทำงานของการค้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 แผนภาพลำดับการผลักลูกบอล

จากรูปที่ 3.12 ลำดับการทำงานของงานในการผลักบอลซึ่งก่อนที่จะผลักลูกบอลได้นั้นมีสิ่งที่ต้องพิจารณาอยู่ 2 สิ่ง คือ ระยะทางใกล้พอหรือไม่หากตรวจสอบจากระยะทางที่ได้รับมาพบว่าวัตถุอยู่ในระยะแล้วหุ่นยนต์จะต้องหยุดเคลื่อนที่ และทำการแยกแยะสีของวัตถุ ซึ่งหุ่นยนต์จะจัดการกับวัตถุที่ไม่ใช่เป้าหมายโดยการผลักบอลลูกนั้นออกไป

ตัวอย่างโปรแกรมที่ 5

```
c= dc.getDistance();
if(c>=dc.x[0]){
        Motor.A.setSpeed(200);
        Motor.C.setSpeed(200);
         Motor.C.forward();
        }else if(c<10){
        Motor.A.setSpeed(100);
        Motor.C.setSpeed(100);
        Thread.sleep(650);
        Motor.A.stop();
        Motor.C.stop();
                if(l1.ObjectCheck()){
                        Motor.A.backward();
```

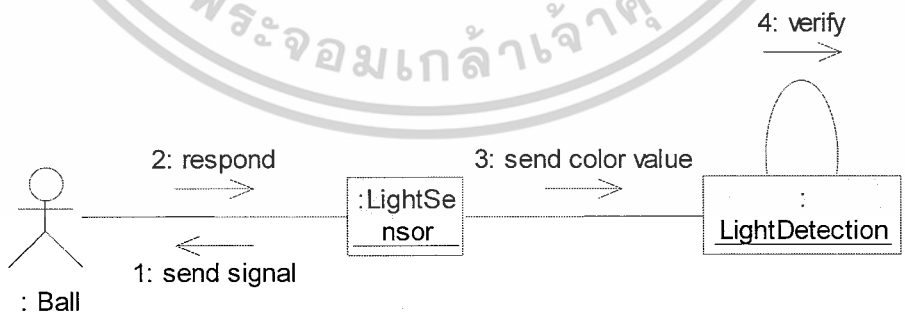
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                Motor.C.backward();
                Thread.sleep(1500);
                m.TurnLeft(60);
                Thread.sleep(1000);
                dc.find();
        }else{
                p.push();
                break;
                }
        }
}
```

จากตัวอย่างโปรแกรมที่ 5 หุ่นยนต์จะวิ่งเข้าหาวัตถุ เมื่อเข้าถึงระยะกำหนดแล้ว จะถึงขั้นตอนของการแยกแยะวัตถุ ซึ่งจะมีการรับค่าจากเซ็นเซอร์แสงเพื่อนำมาตรวจสอบก่อนที่จะจัดการกับวัตถุที่อยู่ตรงหน้าได้อย่างถูกต้อง

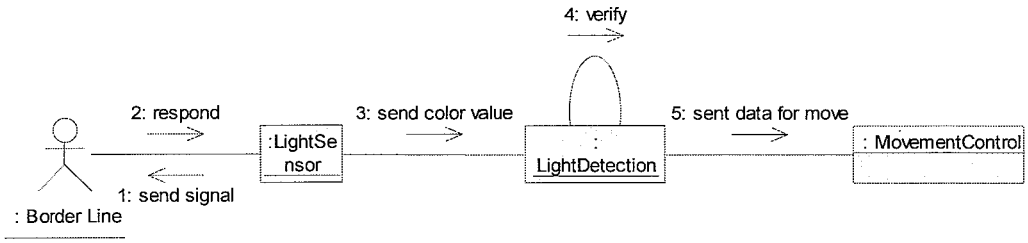
3.2.6 แผนภาพปฏิสัมพันธ์ (Collaboration Diagram)

แผนภาพปฏิสัมพันธ์เป็นแผนภาพแสดงถึงความสัมพันธ์ระหว่างอ็อบเจกต์ตามลักษณะการทำงานแต่ละงานในระบบซึ่งต่างจากแผนภาพลำดับการทำงานที่แสดงถึงลำดับของการทำงาน ซึ่งแผนภาพทั้ง 2 คือสิ่งเดียวกัน จากรูปที่ 3.13 ถึง 3.17 เป็นแผนภาพปฏิสัมพันธ์ของระบบโดยแบ่งตามแต่ละหน้าที่ของระบบ ดังต่อไปนี้

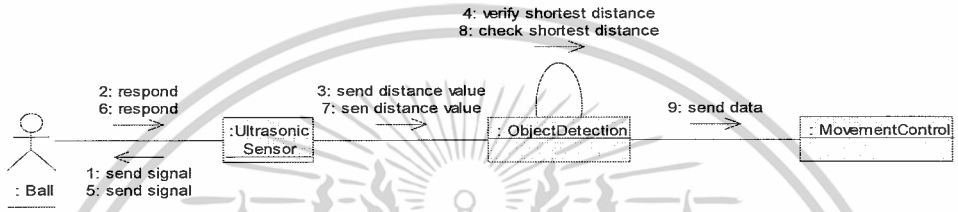


รูปที่ 3.13 แผนภาพปฏิสัมพันธ์การแยกแยะสีของวัตถุ

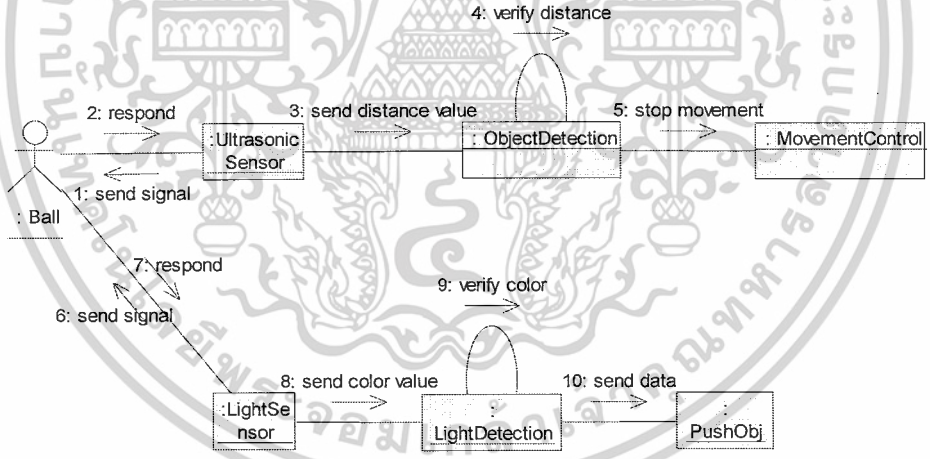
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



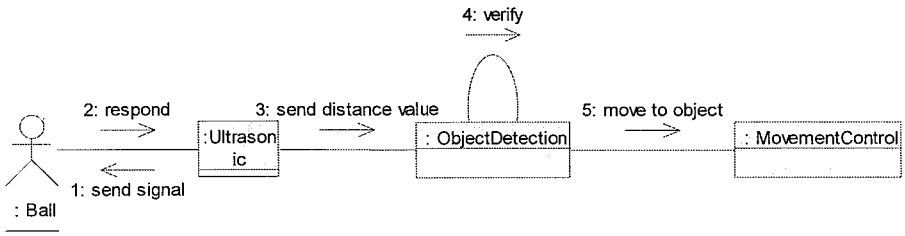
รูปที่ 3.14 แผนภาพปฏิสัมพันธ์การตรวจสอบเส้นขอบสนาม



รูปที่ 3.15 แผนภาพปฏิสัมพันธ์การค้นหา



รูปที่ 3.16 แผนภาพปฏิสัมพันธ์การผลัก

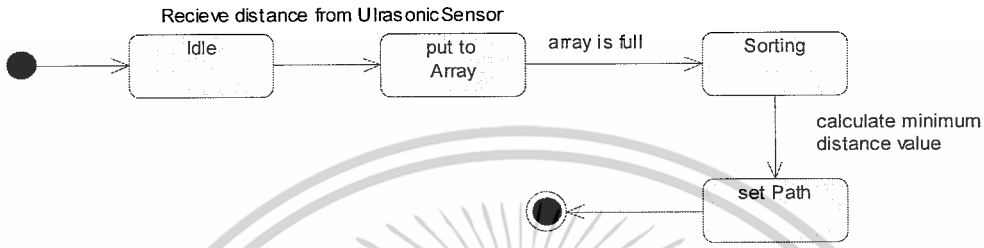


รูปที่ 3.17 แผนภาพปฏิสัมพันธ์การเคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.7 แผนภาพสถานะ (State Chart Diagram)

เป็นแผนภาพแสดงให้เห็นพฤติกรรมของอ็อบเจกต์ โดยประกอบด้วยสถานะต่างๆ ของอ็อบเจกต์ และเหตุการณ์ต่างๆ ที่ทำให้สถานะของอ็อบเจกต์นั้นเปลี่ยน และการกระทำที่เกิดขึ้นเมื่อสถานะของระบบเปลี่ยนไป โดยแผนภาพสถานะของระบบที่ซับซ้อนและน่าสนใจมีดังนี้



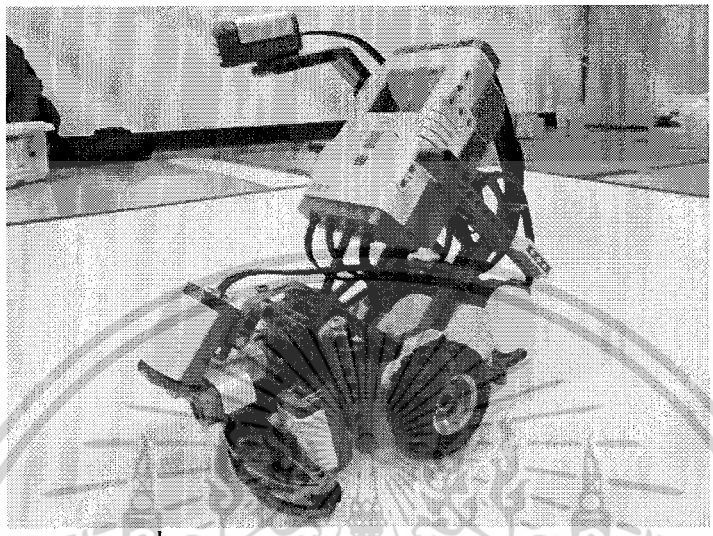
รูปที่ 3.18 แผนภาพสถานะการควบคุมการเคลื่อนที่หรือเดินไปข้างหน้า

จากรูปที่ 3.18 แผนภาพการเปลี่ยนสถานะของการค้นหาวัตถุ ซึ่งจะต้องมีการรับค่าจนกว่าอาร์เรย์จะเต็ม ซึ่งเมื่อจบงานของการรับค่าแล้ว จะเปลี่ยนสถานะมาทำงานในส่วนของการจัดเรียงข้อมูลในอาร์เรย์ เพื่อหาค่าที่น้อยสุดเมื่อได้ค่าที่น้อยแล้ว จะถึงขั้นตอนของการหาทิศทางของวัตถุที่อยู่ใกล้ที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ

3.2.1 การออกแบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ



รูปที่ 3.19 หุ่นยนต์ตรวจจับ และจัดการวัตถุ

จากรูปที่ 3.19 แสดงลักษณะหุ่นยนต์โดยรวมที่ได้ออกแบบไว้ โดยการออกแบบหุ่นยนต์จะต้องคำนึงถึงความต้องการของระบบเป็นหลัก เพื่อให้หุ่นยนต์มีความยืดหยุ่น และมีประสิทธิภาพในการทำงานมากที่สุด การออกแบบจะแบ่งตามความต้องการของระบบ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

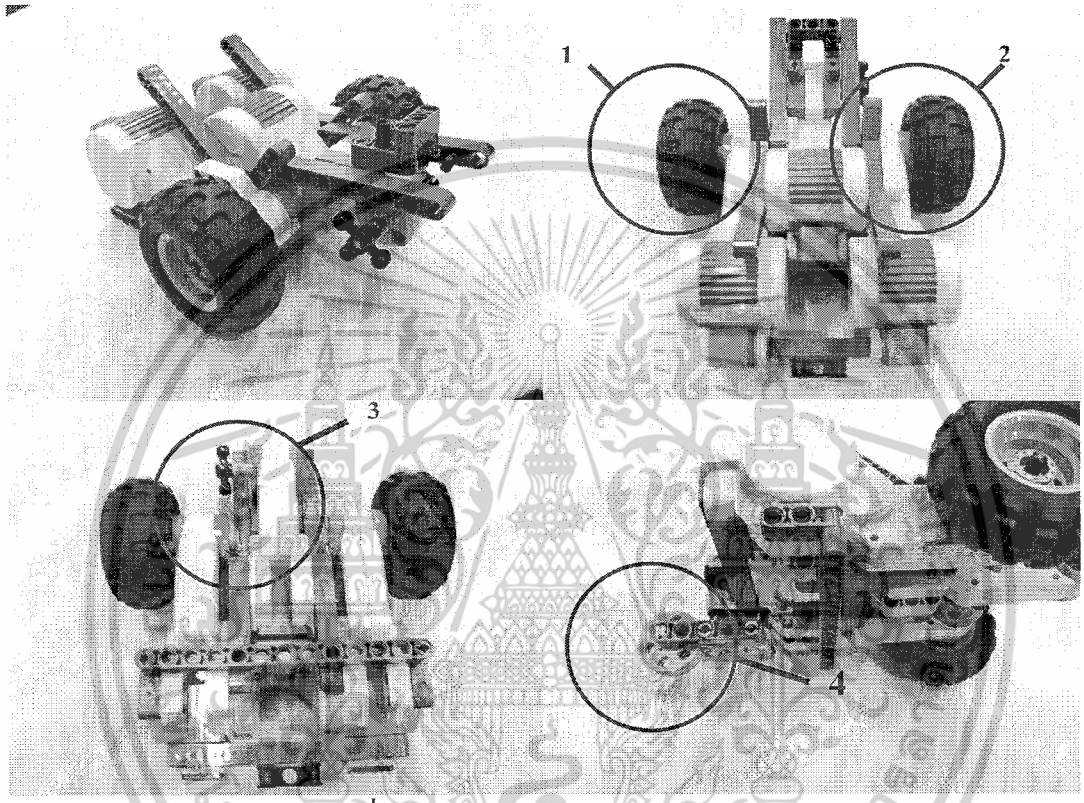
ตารางที่ 3.6 การเชื่อมโยงระหว่างความต้องกับการออกแบบหุ่นยนต์

ความต้องการของระบบ	การออกแบบหุ่นยนต์
1. หุ่นยนต์ต้องมีขอบเขตในการติดตาม	1. ใช้อุปกรณ์เซ็นเซอร์ด้วยแสงตัวที่ 1 เป็นตัวตรวจสอบสีบนสนาม และเส้นขอบเขตของสนาม โดยติดตั้งอุปกรณ์นี้ให้มีลักษณะขนานกับพื้น
2. หุ่นยนต์สามารถตรวจจับวัตถุในสนามได้	2. ใช้อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่เป็นตัวตรวจจับวัตถุ โดยติดตั้งอุปกรณ์นี้ให้อยู่ในระดับเดียวกันกับวัตถุ
3. หุ่นยนต์สามารถแยกแยะสีของวัตถุได้	3. ใช้อุปกรณ์เซ็นเซอร์ด้วยแสงตัวที่ 2 เป็นตัวตรวจสอบสีของวัตถุ โดยติดตั้งอุปกรณ์นี้ให้อยู่ใกล้กับวัตถุมากที่สุด
4. หุ่นยนต์สามารถจัดการผลักวัตถุได้	4. ใช้อุปกรณ์มอเตอร์ในการผลักวัตถุ โดยติดตั้งอุปกรณ์นี้อยู่ด้านหน้า และอยู่ระดับเดียวกันกับวัตถุ รวมถึงออกแบบสร้างกลไกฟันเฟือง และแขนที่ใช้ในการผลักลูกบอลจากชุดตัวต่อเลโก้
5. หุ่นยนต์สามารถเคลื่อนที่ และติดตามวัตถุได้	5. ใช้อุปกรณ์มอเตอร์ 2 ตัว ในการเคลื่อนที่เข้าไปหาวัตถุ รวมถึงใช้เป็นฐานรองรับการติดตั้งอุปกรณ์อื่นๆ ของระบบ โดยติดตั้งเป็นคู่ขนานกันระหว่างมอเตอร์ทั้ง 2 ตัว
6. หุ่นยนต์สามารถสั่งการด้วยเสียงได้	6. ใช้อุปกรณ์เซ็นเซอร์ด้วยเสียง ในการวิเคราะห์สัญญาณเสียงที่รับเข้ามา โดยติดตั้งอยู่ส่วนบนของหุ่นยนต์ เพื่อหลีกเลี่ยงผลกระทบโดยตรงจากเสียงของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การประกอบหุ่นยนต์ตรวจจับ และจัดการวัตถุ

1. เริ่มจากการสร้างโครงสร้างส่วนที่ใช้ในการขับเคลื่อน ซึ่งจะต้องรับน้ำหนักของตัวคอนโทรลเลอร์ และอุปกรณ์เซ็นเซอร์ต่างๆ ตลอดจนชิ้นส่วนต่างๆ จึงต้องมีสมดุล และความแข็งแกร่งทนทานมากที่สุด ดังรูปที่ 3.20

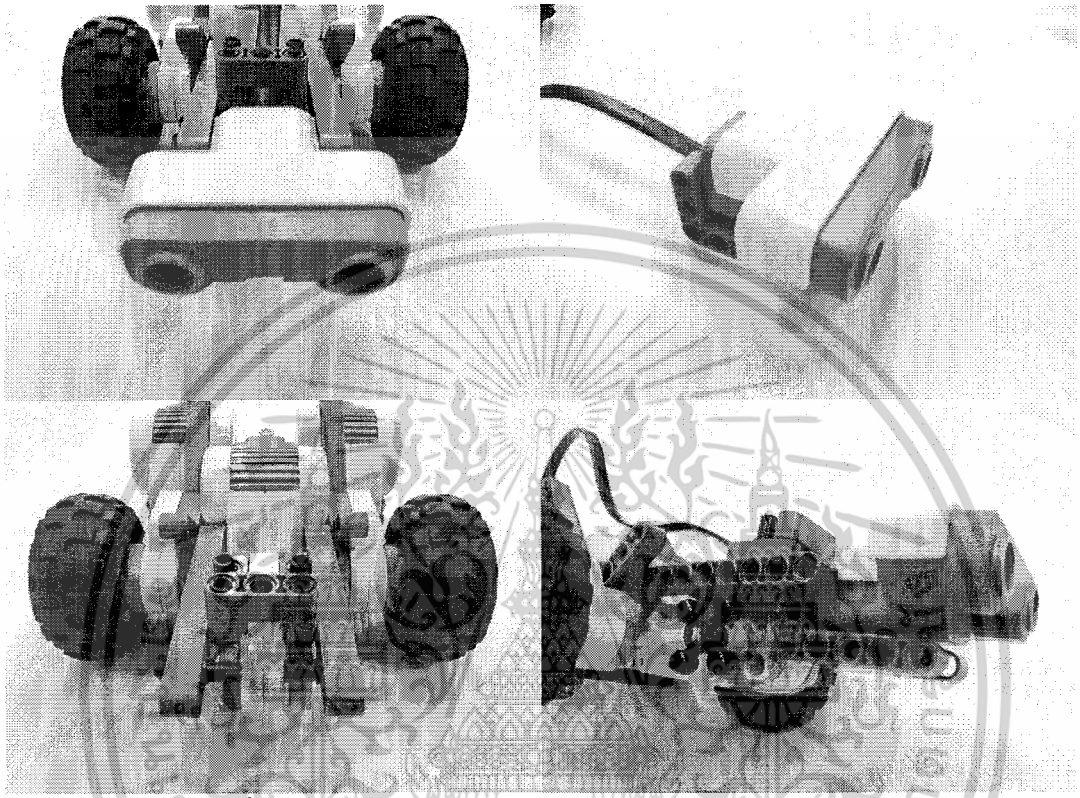


รูปที่ 3.20 ลักษณะการวางฐานมอเตอร์

ลักษณะรูปแบบมอเตอร์ที่นำมาต่อเป็นฐานให้แก่หุ่นยนต์โดยใช้ในการขับเคลื่อน 2 ตัว คือ มอเตอร์หมายเลข 1 และ 2 ส่วนมอเตอร์หมายเลข 3 ใช้สำหรับเป็นกลไกในการผลักลูกบอล และกลไกหมายเลข 4 ใช้สำหรับโฟลว (Flow) ตามการเคลื่อนที่ของหุ่นยนต์ดังรูปที่ 3.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

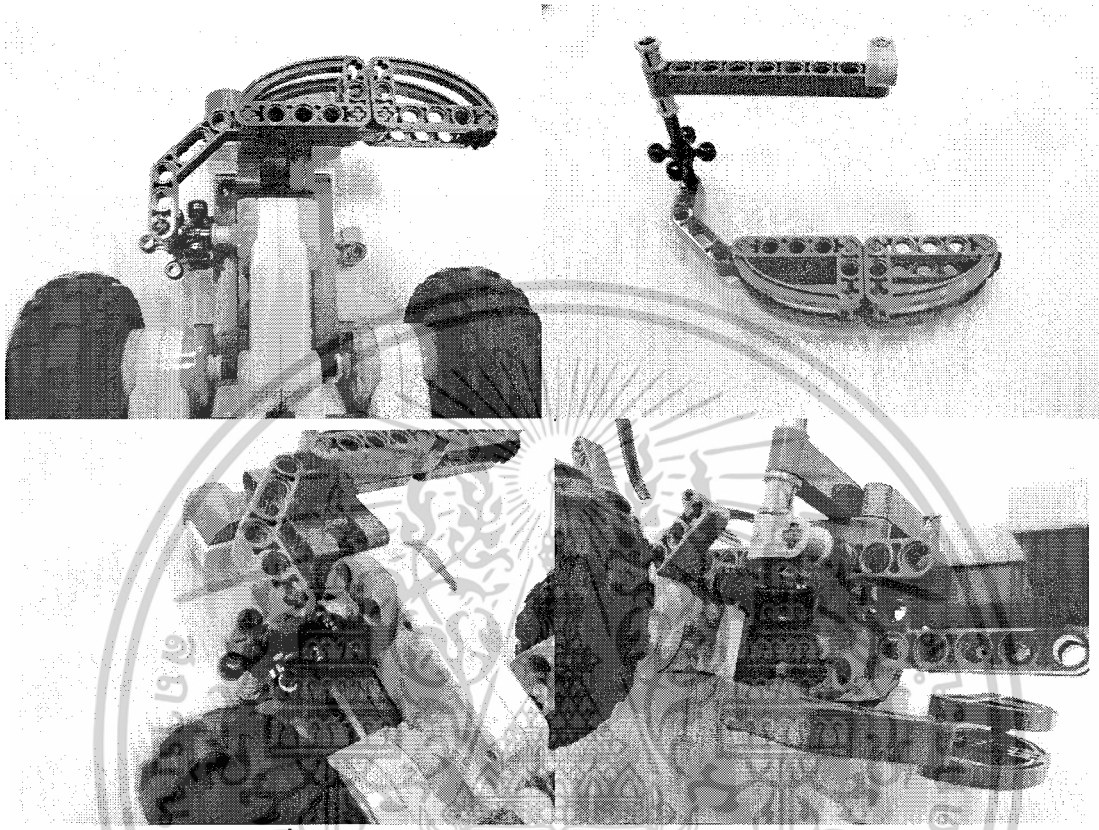
2. อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ที่ใช้ในการตรวจจับ และวัดระยะวัตถุ โดยลักษณะการทำงานใช้หลักการสะท้อนความถี่เป็นแนวเส้นตรงระหว่างตัวเซ็นเซอร์กับวัตถุ จึงต้องวางให้อยู่ด้านหน้าของหุ่นยนต์ และอยู่ระดับเดียวกันกับตัววัตถุ ดังรูปที่3.21



รูปที่ 3.21 ลักษณะการวางอุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

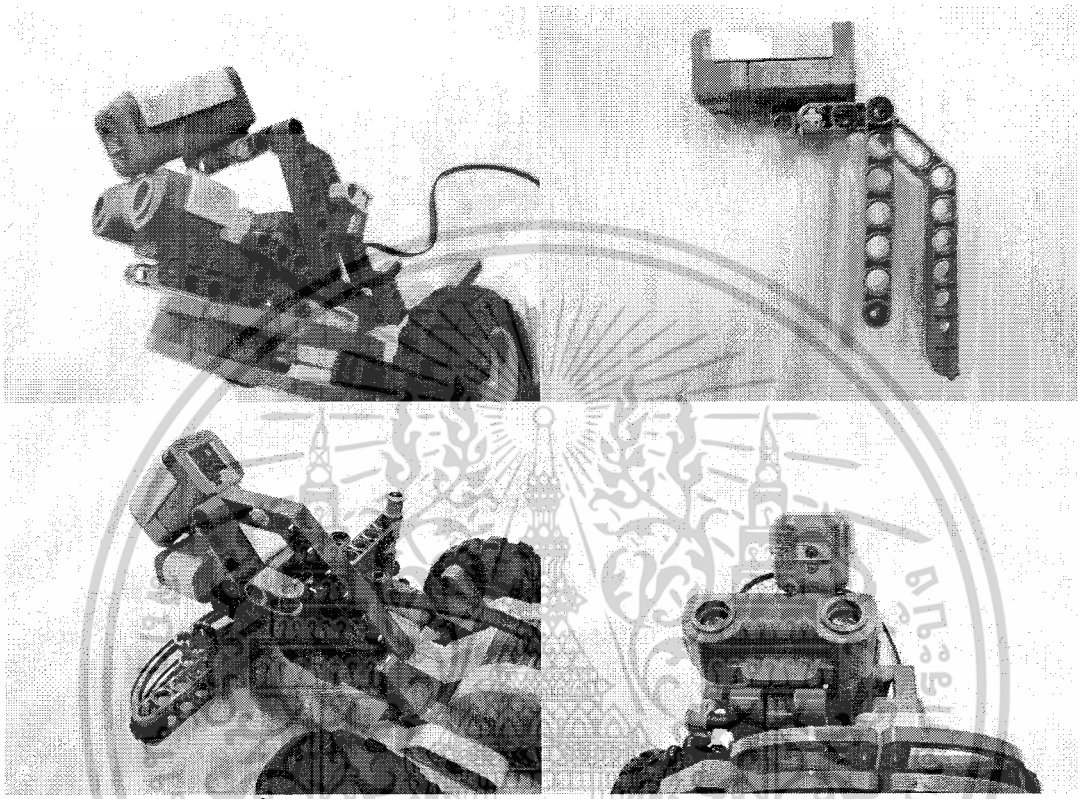
3. กลไกการผลักลูกบอลโดยใช้ชุดตัวต่อ ฟันเฟือง และมอเตอร์ โดยมีฟันเฟืองต่อกับมอเตอร์ และฟันเฟืองที่ต่อกับแขนที่ใช้ในการผลักวัตถุ มีลักษณะดังรูปที่ 3.22



รูปที่ 3.22 ลักษณะการต่อฟันเฟืองมอเตอร์กับแขนผลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

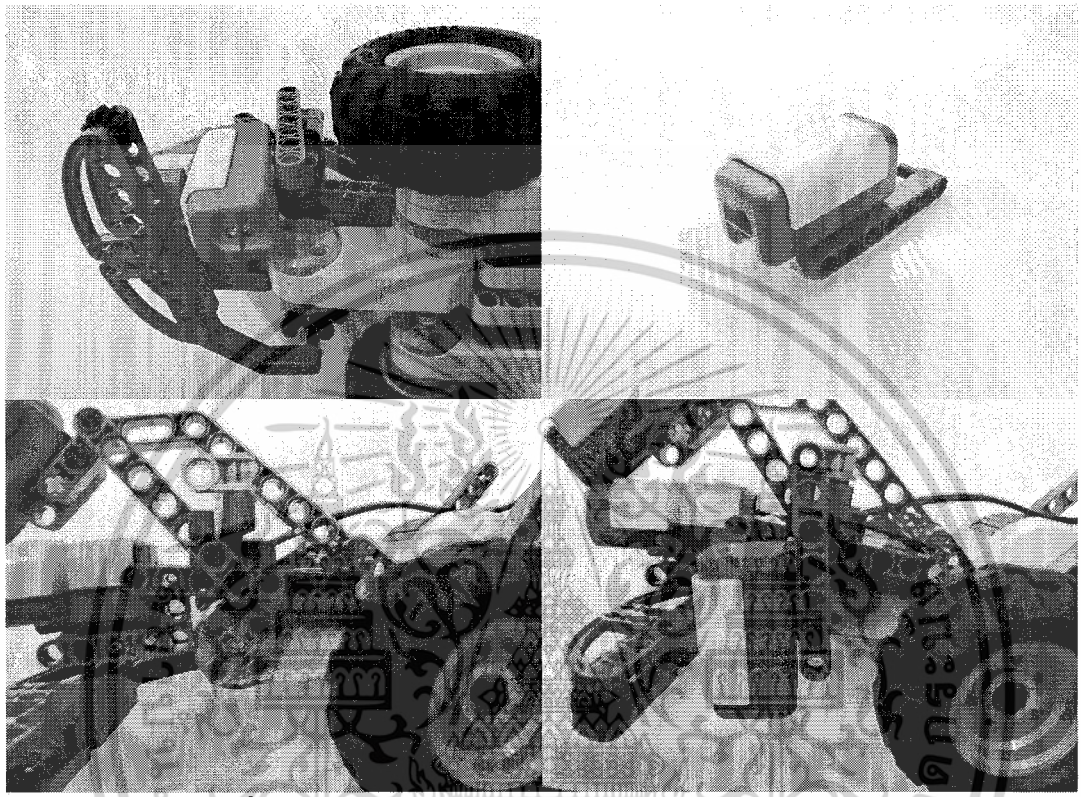
4. อุปกรณ์เซ็นเซอร์ด้วยแสงที่ใช้ในการตรวจสอบ และแยกแยะวัตถุหรือลูกบอล เนื่องจากตัวเซ็นเซอร์นีมีข้อจำกัดของระยะทางในการตรวจสอบที่ใกล้ โดยจะวางอยู่ด้านหน้าของหุ่นยนต์ให้ใกล้กับรัศมีของตัววัตถุมากที่สุด เพื่อให้ง่ายต่อการตรวจสอบ และแยกแยะวัตถุ ดังรูปที่ 3.23



รูปที่ 3.23 ลักษณะการวางตัวตรวจสอบ และแยกแยะวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

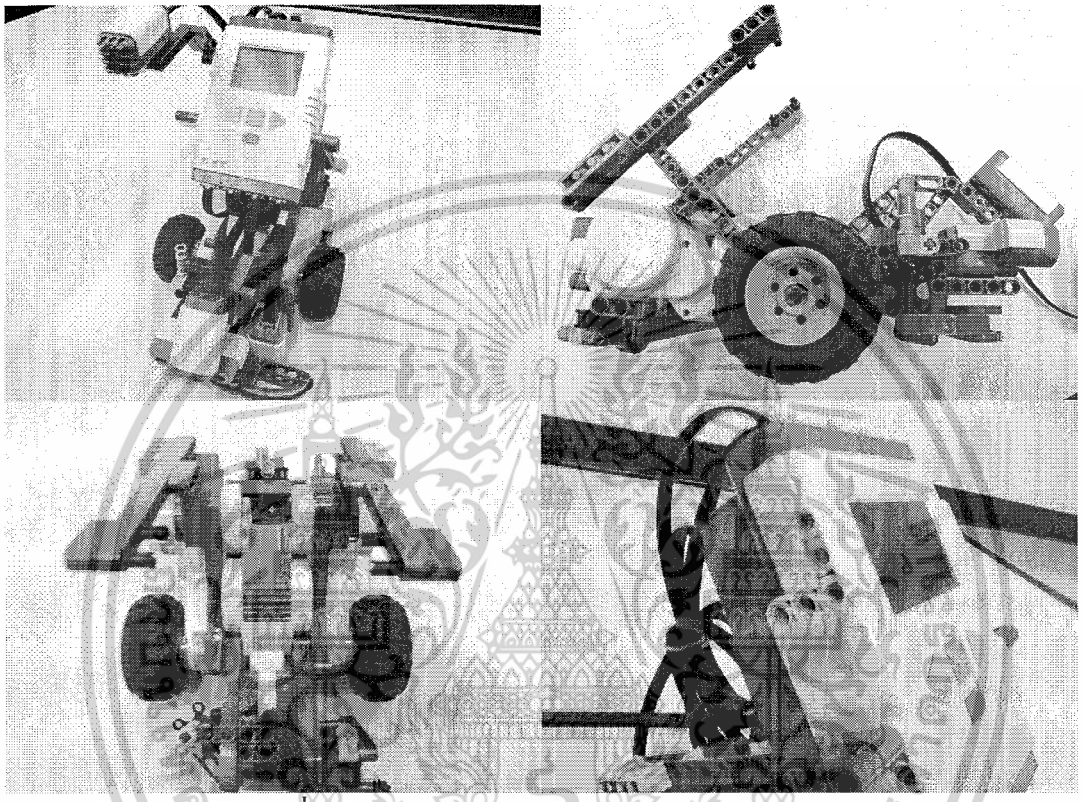
5. อุปกรณ์เซ็นเซอร์ด้วยแสงที่ใช้ในการตรวจสอบพื้นสนาม และเส้นขอบสนาม จะวางในลักษณะหัวเซ็นเซอร์ขนานกับพื้น อยู่ด้านหน้าตัวมอเตอร์ เพื่อให้ง่ายต่อการตรวจสอบสนาม และเส้นขอบสนามเมื่อเคลื่อนที่ไปข้างหน้าดังรูปที่ 3.24



รูปที่ 3.24 ลักษณะการวางอุปกรณ์ตรวจสอบเส้นขอบสนาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ลักษณะการวางตัวคอนโทรลเลอร์ ใช้สำหรับเป็นศูนย์กลางในการประมวลผลการทำงานของระบบ และเป็นตัวเชื่อมต่อระหว่างอุปกรณ์เซ็นเซอร์ต่างๆ เข้าด้วยกัน จะวางอยู่บนฐานมอเตอร์เคลื่อนที่ เพื่อให้ง่ายต่อการใช้งานอินเตอร์เฟส (Interface) ของตัวคอนโทรลเลอร์ ดังรูปที่ 3.25



รูปที่ 3.25 ลักษณะการวางอุปกรณ์คอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลอง และผลการทดลอง

การทดลองระบบหุ่นยนต์ตรวจจับ และติดตามวัตถุ โดยแบ่งการทดลองออกเป็น 3 การทดลองด้วยกัน ดังต่อไปนี้

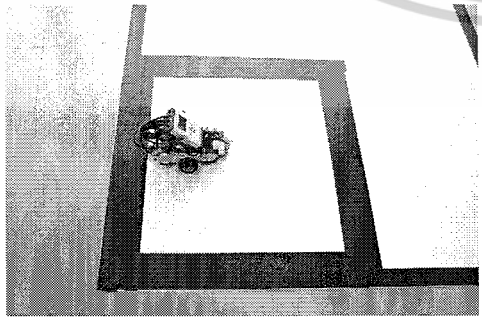
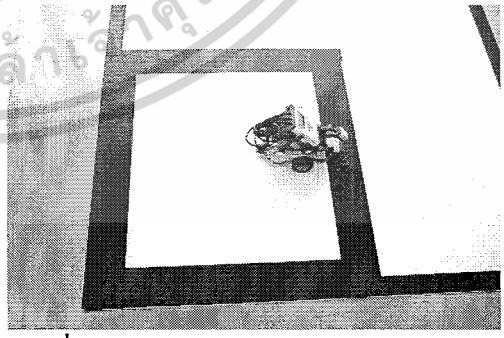
4.1 การทดลองระบบตรวจสอบเส้นขอบสนาม

วัตถุประสงค์ เพื่อทำการดูผลการตอบสนองของหุ่นยนต์ เมื่อหุ่นยนต์อยู่บนสนาม และเคลื่อนที่ไปเจอเส้นขอบสนาม

สมมติฐาน ถ้าใช้อุปกรณ์เซ็นเซอร์ด้วยแสง (Light Sensor) ช่วยในตรวจจับเส้นขอบสนาม และพื้นสนาม แล้วหุ่นยนต์ตรวจจับพบเส้นขอบสนามจะต้องทำการหยุด และเคลื่อนที่กลับมาหรือเคลื่อนที่ไปทิศทางอื่นเพื่อไม่ให้หุ่นยนต์ออกนอกสนาม

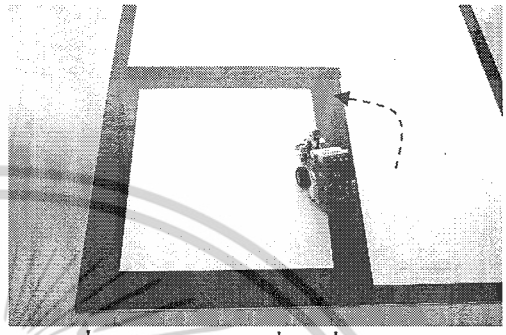
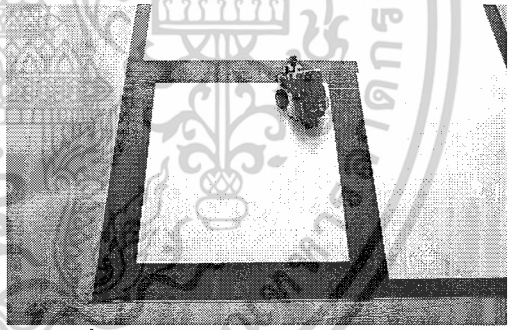
วิธีการทดลอง

ตารางที่ 4.1 การทดลองระบบตรวจสอบเส้นขอบสนาม

วิธีการทดลอง	ผลการทดลอง
1. นำหุ่นยนต์มาไว้ภายในสนาม และปล่อยให้หุ่นยนต์เคลื่อนในสนามอย่างอิสระ โดยกำหนดเส้นขอบเขตของสนามเป็นสีดำ และภายในสนามมีพื้นที่เป็นสีขาว ดังรูปที่ 4.1  **รูปที่ 4.1** การทดลองหุ่นยนต์ตรวจสอบเส้นขอบสนาม	1. หุ่นยนต์เคลื่อนที่ไปข้างหน้า จนตรวจพบเส้นขอบสนามดังรูปที่ 4.2  **รูปที่ 4.2** หุ่นยนต์ตรวจพบเส้นขอบสนาม

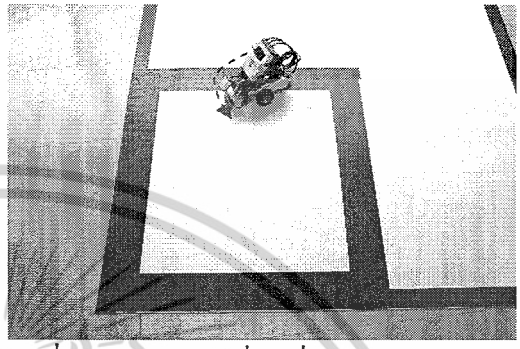
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 การทดลองระบบตรวจสอบเส้นขอบสนาม (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	2. หุ่นยนต์เคลื่อนหลบไปด้านซ้ายเมื่อตรวจพบเส้นขอบสนามดังรูปที่ 4.3  **รูปที่ 4.3** หุ่นยนต์เคลื่อนที่หลบไปด้านข้าง 3. หุ่นยนต์เคลื่อนที่ไปข้างหน้า และตรวจพบเส้นขอบสนามดังรูปที่ 4.4  **รูปที่ 4.4** หุ่นยนต์ตรวจพบเส้นขอบสนามครั้งที่2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 การทดลองระบบตรวจสอบเส้นขอบสนาม (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	4. หุ่นยนต์เคลื่อนหลบไปด้านซ้ายเมื่อตรวจพบเส้นขอบสนาม ดังรูปที่ 4.5  **รูปที่ 4.5** หุ่นยนต์เคลื่อนที่หลบไปด้านข้าง

วิเคราะห์ หุ่นยนต์สามารถตรวจจับ และหาทิศทางของวัตถุได้ แต่มีความคลาดเคลื่อนขึ้นบางกรณี เนื่องจากวัตถุที่ใช้ในการทดลองมีลักษณะเป็นทรงกลมทำให้สามารถสะท้อนคลื่นกลับมาได้หลายทิศทางจนทำให้เป็นสาเหตุที่ทำให้อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ทำงานคลาดเคลื่อน

สรุป เนื่องจากตำแหน่งการวางอุปกรณ์เซ็นเซอร์ด้วยแสงที่ถูกออกแบบมาให้มีระยะใกล้กับพื้นสนามมากๆ นั้น ทำให้ความแม่นยำในการวิเคราะห์สีสูง การทดลองจึงสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

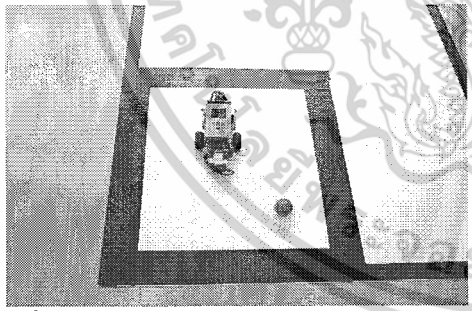
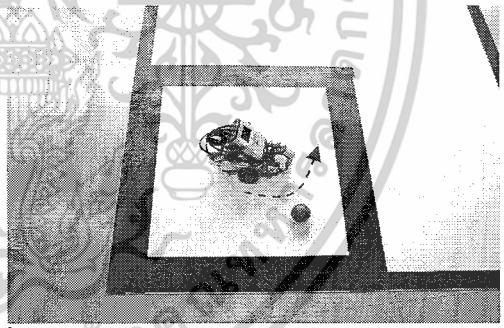
4.2 การทดลองระบบการค้นหา และเข้าถึงลูกบอลในสนาม

วัตถุประสงค์ เพื่อทำการดูผลการตอบสนองของหุ่นยนต์ เมื่อต้องทำการค้นหา และเข้าถึงลูกบอลในระยะที่เหมาะสม ภายใต้สภาพแวดล้อมที่มีลูกบอลที่เป็นเป้าหมาย 1 ลูก รวมถึงสนามที่มีขอบเขตจำกัดไว้

สมมติฐาน ถ้าใช้ความสัมพันธ์ของอุปกรณ์เซ็นเซอร์ต่างๆ ประกอบด้วยอุปกรณ์เซ็นเซอร์ด้วยแสง 2 ชุด และอุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ให้ทำงานร่วมกัน แล้วจะทำให้หุ่นยนต์ค้นหา และเคลื่อนที่เข้าถึงลูกบอลในสนามได้

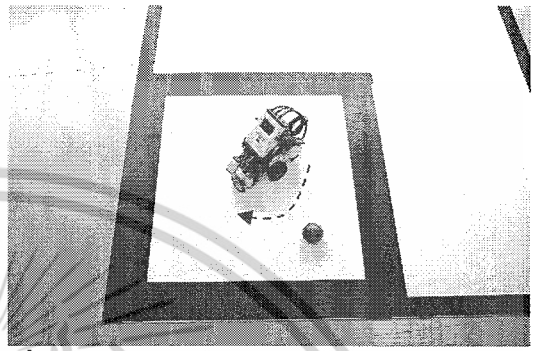
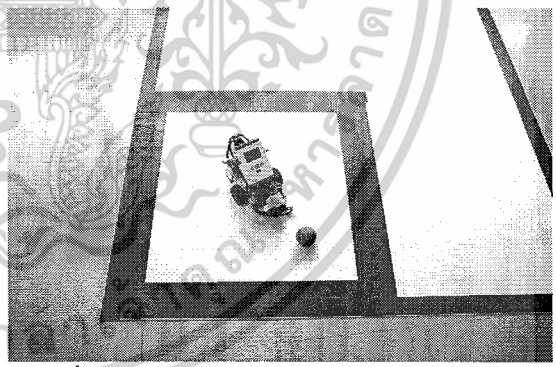
วิธีการทดลอง

ตารางที่ 4.2 การทดลองระบบการค้นหา และเข้าถึงลูกบอลในสนาม

วิธีการทดลอง	ผลการทดลอง
1. นำหุ่นยนต์มาไว้ภายในสนาม และนำลูกบอลสีแดงมาไว้ภายในสนามด้วยระยะทางห่างจากหุ่นยนต์ไม่เกิน 30 เซนติเมตร แล้วดูการตอบสนองของหุ่นยนต์ดังรูปที่ 4.6  **รูปที่ 4.6** การทดลองหุ่นยนต์เพื่อค้นหา และเข้าถึงลูกบอลในสนาม	1. หุ่นยนต์หมุนตัวไปทางซ้าย เพื่อเริ่มทำการค้นหาดังรูปที่ 4.7  **รูปที่ 4.7** หุ่นยนต์ทำการเริ่มค้นหาตำแหน่งของวัตถุ

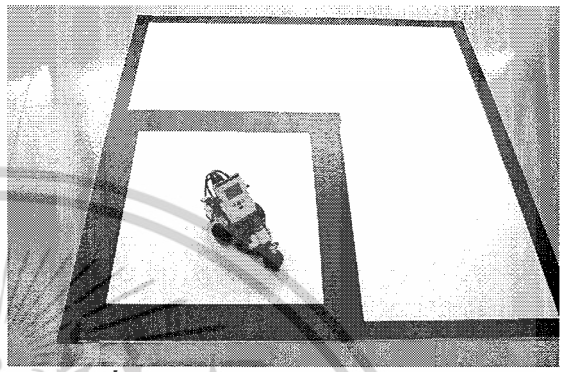
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 การทดลองระบบการค้นหา และเข้าถึงลูกบอลในสนาม (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	2. หุ่นยนต์หันกลับมาทางขวาเพื่อเก็บค่าระยะทางของวัตถุที่เจอ ดังรูปที่ 4.8  **รูปที่ 4.8** หุ่นยนต์ทำการเก็บค่าระยะทางของวัตถุที่เจอ 3. หุ่นยนต์หันหน้าไปหาวัตถุที่ตรวจพบเจอในสนาม ดังรูปที่ 4.9  **รูปที่ 4.9** หุ่นยนต์ตรวจพบตำแหน่งของวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 การทดลองระบบการค้นหา และเข้าถึงลูกบอลในสนาม (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	4. หุ่นยนต์เคลื่อนที่เข้าไปหาวัตถุ และหยุดในห่างที่เหมาะสม ดังรูปที่ 4.10  **รูปที่ 4.10** หุ่นยนต์เคลื่อนที่เข้าหาวัตถุ

วิเคราะห์ เกิดอุปสรรคในการทดลองเนื่องจากลูกบอลมีลักษณะเป็นทรงกลม ทำให้อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่จับระยะทางไม่ได้หรืออ่านค่าคลาดเคลื่อน ส่งผลให้หุ่นยนต์อาจวิ่งเลยลูกบอลไป และทำการค้นหาใหม่เรื่อยๆ จนสามารถเข้าถึงลูกบอลที่อยู่ในสนาม และจัดการผลักลูกบอลที่ไม่ใช่เป้าหมายออกนอกสนามได้ ซึ่งการทดลองบางกรณีกระบวนการค้นหาของหุ่นยนต์อาจเคลื่อนที่เข้าไปชนกับลูกบอลที่อยู่ในสนามได้บางครั้ง

สรุป เนื่องจากข้อจำกัดของฮาร์ดแวร์ที่อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ต้องใช้ช่วงเวลาในการรับค่าห่างกัน 0.2 วินาที รวมถึงการข้อจำกัดของซอฟต์แวร์ซึ่งหากทำการเรียกใช้งานจากคลาสต่างๆ ในระบบเยอะเกินทำให้การตอบสนองแบบเรียลไทม์ช้าลง จึงจะทำให้ค่าที่ได้มาเกิดความคลาดเคลื่อน ทำให้ขั้นตอนการค้นหาต้องเกิดข้อผิดพลาด และต้องระยะเวลาการค้นหานาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

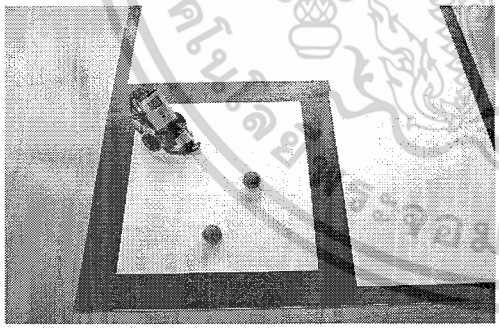
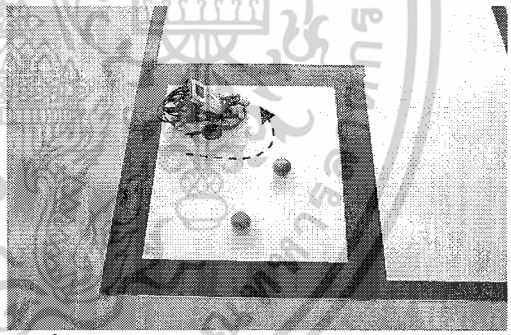
4.3 การทดลองระบบการค้นหา และจัดการวัตถุ

วัตถุประสงค์ เพื่อทดลองดูการตอบสนองของหุ่นยนต์ เมื่อหุ่นยนต์เริ่มทำการค้นหา เข้าถึง และจัดการกับลูกบอล ภายใต้สภาพแวดล้อมที่มีลูกบอลที่เป็นเป้าหมาย และไม่ใช่เป้าหมาย รวมถึงสนามที่มีขอบเขตจำกัดไว้

สมมติฐาน ถ้าใช้ความสัมพันธ์ของอุปกรณ์เซ็นเซอร์ต่างๆ ประกอบด้วยอุปกรณ์เซ็นเซอร์ด้วยแสง 2 ชุด และอุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ให้ทำงานร่วมกัน แล้วจะทำให้หุ่นยนต์ค้นหา และจัดการกับลูกบอลทั้งสองอยู่ในสนามได้

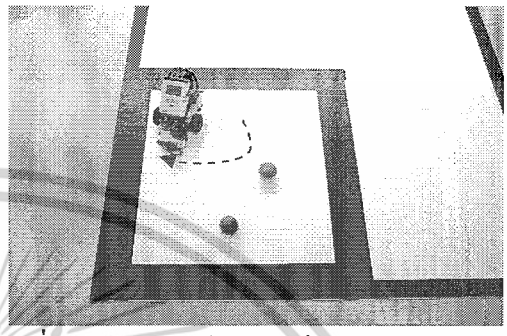
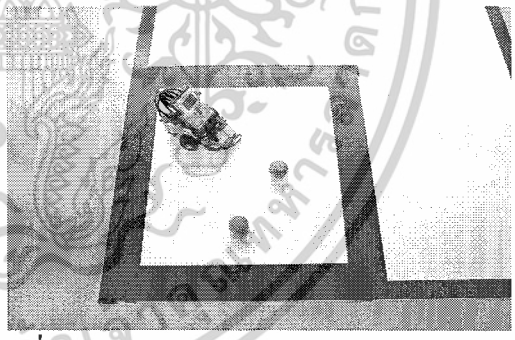
วิธีการทดลอง

ตารางที่ 4. 3 การทดลองระบบการค้นหา และจัดการวัตถุ

วิธีการทดลอง	**ผลการทดลอง**
1. นำหุ่นยนต์มาไว้ภายในสนาม นำลูกบอลมาไว้ภายในสนามด้วยระยะทางต่างกันรอบจุดเริ่มต้นการทำงานของหุ่นยนต์ โดยกำหนดให้ลูกบอลสีน้ำเงินคือลูกที่เป็นเป้าหมาย และลูกบอลสีน้ำเงินไม่ใช่เป้าหมาย ดังรูปที่ 4.11  **รูปที่ 4.11** การทดลองหุ่นยนต์ค้นหา และจัดการวัตถุ	1. หุ่นยนต์หมุนตัวไปทางซ้าย เพื่อเริ่มทำการค้นหา ดังรูปที่ 4.12  **รูปที่ 4.12** หุ่นยนต์ทำการเริ่มค้นหาตำแหน่งของวัตถุ

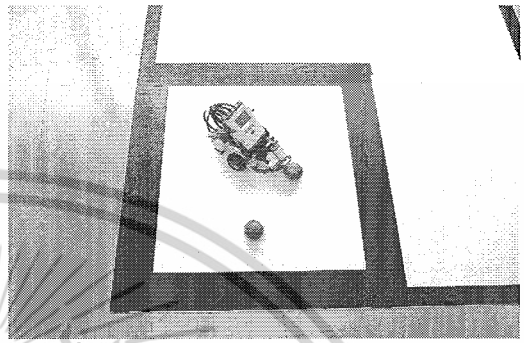
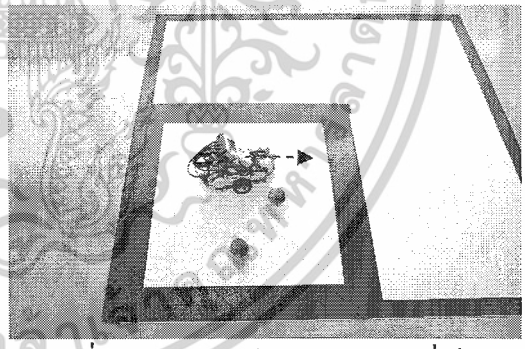
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4. 3 การทดลองระบบการค้นหา และจัดการวัตถุ (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	2. หุ่นยนต์หันกลับมาทางขวาเพื่อเก็บค่าระยะทางของวัตถุที่เจอ ดังรูปที่ 4.13  **รูปที่ 4.13** หุ่นยนต์ทำการเก็บค่าระยะทางของวัตถุที่เจอ 3. หุ่นยนต์หันหน้าไปหาวัตถุที่อยู่ใกล้ที่สุด ดังรูปที่ 4.14  **รูปที่ 4.14** หุ่นยนต์ตรวจพบตำแหน่งของวัตถุที่อยู่ใกล้ที่สุด

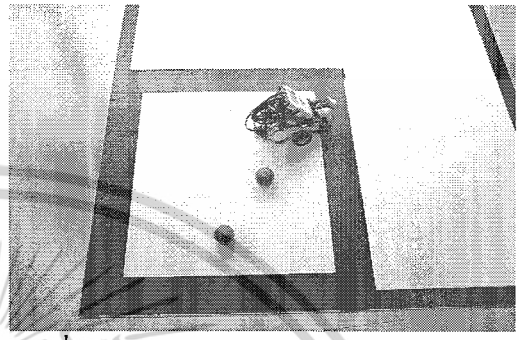
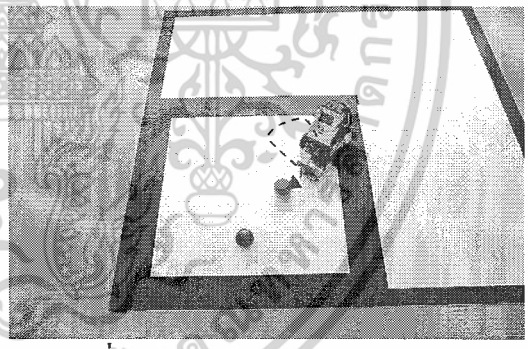
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4. 3 การทดลองระบบการค้นหา และจัดการวัตถุ (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	4. หุ่นยนต์เคลื่อนที่เข้าไปหาวัตถุ และทำการตรวจสอบแยกแยะสีของลูกบอล ดังรูปที่ 4.15  **รูปที่ 4.15** หุ่นยนต์แยกแยะสีของลูกบอล 5. หุ่นยนต์ทำการเคลื่อนที่หลบ โดยถอยหลังกลับ และเลี้ยวไปทางซ้าย เพื่อค้นหาลูกบอลที่ไม่ใช่เป้าหมายต่อไป ดังรูปที่ 4.16  **รูปที่ 4.16** หุ่นยนต์หลบหลีกวัตถุที่เป็นเป้าหมาย

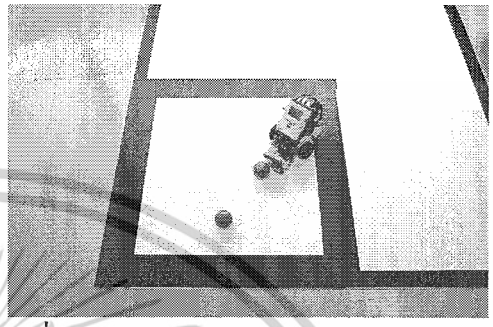
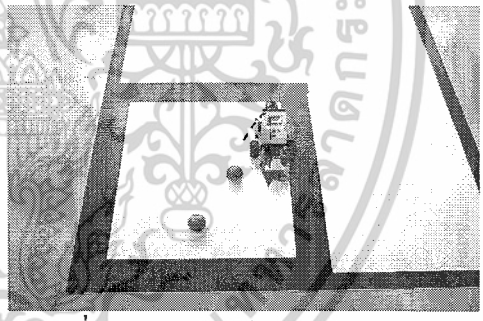
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4. 3 การทดลองระบบการค้นหา และจัดการวัตถุ (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	6. หุ่นยนต์เคลื่อนที่ไปเจอเส้นขอบสนาม ดังรูปที่ 4.17  **รูปที่ 4.17** หุ่นยนต์ตรวจพบเส้นขอบสนาม 7. หุ่นยนต์ทำการค้นหาวัตถุที่อีกครั้งหลังตรวจพบเส้นขอบสนาม ดังรูปที่ 4.18  **รูปที่ 4.18** หุ่นยนต์ทำการค้นหาวัตถุ

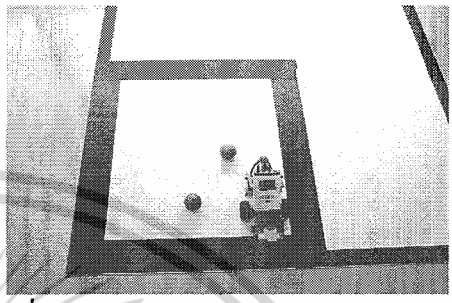
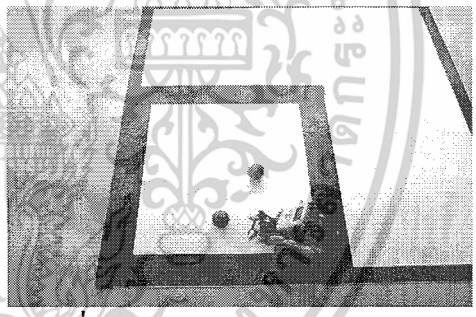
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4. 3 การทดลองระบบการค้นหา และจัดการวัตถุ (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	8. หุ่นยนต์เคลื่อนที่เข้าไปหา และแยกแยะสีของลูกบอล ดังรูปที่ 4.19  **รูปที่ 4.19** หุ่นยนต์ตรวจสอบแยกแยะสีวัตถุ 7. หุ่นยนต์ตรวจสอบว่าเป็นลูกบอลเป้าหมาย ก็ทำการเคลื่อนที่หลบ และค้นหาต่อ ดังรูปที่ 4.20  **รูปที่ 4.20** หุ่นยนต์ทำการค้นหาวัตถุ

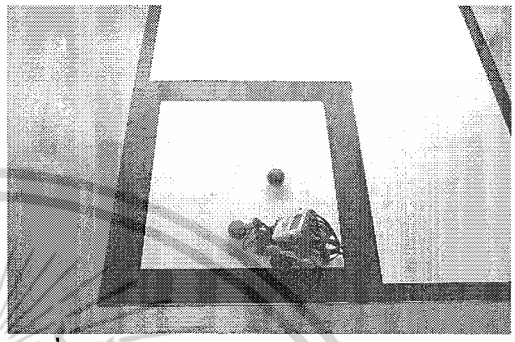
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4. 3 การทดลองระบบการค้นหา และจัดการวัตถุ (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	8. หุ่นยนต์เคลื่อนที่ไปเจอเส้นขอบสนาม ดังรูปที่ 4.21  **รูปที่ 4.21** หุ่นยนต์ตรวจพบเส้นขอบสนาม 9. หุ่นยนต์ทำการค้นหาวัตถุที่อีกครั้งหลังตรวจพบเส้นขอบสนาม ดังรูปที่ 4.22  **รูปที่ 4.22** หุ่นยนต์ทำการค้นหาวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4. 3 การทดลองระบบการค้นหา และจัดการวัตถุ (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	10. หุ่นยนต์เคลื่อนที่เข้าไปหา และแยกแยะสีของลูกบอล ดังรูปที่ 4.23  **รูปที่ 4.23** หุ่นยนต์ตรวจสอบแยกแยะสีวัตถุ 11. หุ่นยนต์ตรวจสอบแยกแยะสีของลูกบอล พบว่าไม่ใช่เป้าหมาย จึงทำการผลักลูกบอล ดังรูปที่ 4.24 **รูปที่ 4.24** หุ่นยนต์ทำการผลักวัตถุ

วิเคราะห์ เกิดอุปสรรคในการทดลองเนื่องจากลูกบอลมีลักษณะเป็นทรงกลม ทำให้อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่จับระยะทางไม่ได้หรืออ่านค่าคลาดเคลื่อน ส่งผลให้หุ่นยนต์อาจวิ่งเลยลูกบอลไป และทำการค้นหาใหม่เรื่อยๆ จนสามารถเข้าถึงลูกบอลที่อยู่ในสนาม และจัดการผลักลูกบอลที่ไม่ใช่เป้าหมายออกนอกสนามได้ ซึ่งการทดลองบางกรณีกระบวนการค้นหาของหุ่นยนต์อาจเคลื่อนที่เข้าไปชนกับลูกบอลที่อยู่ในสนามได้บางครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุป เนื่องจากข้อจำกัดของฮาร์ดแวร์ที่อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ต้องใช้ช่วงเวลาในการรับค่าห่างกัน 0.2 วินาที รวมถึงการข้อจำกัดของซอฟต์แวร์ซึ่งหากทำการเรียกใช้งานจากคลาสต่างๆ ในระบบเยอะเกินทำให้การตอบสนองแบบเรียลไทม์ช้าลง จึงจะทำให้ค่าที่ได้มาเกิดความคลาดเคลื่อน ทำให้ขั้นตอนการค้นหาต้องเกิดข้อผิดพลาด และต้องระยะเวลาการค้นหานาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปโครงงาน และข้อเสนอแนะ

5.1 สรุปโครงงาน

การพัฒนาระบบสมองกลฝังตัวด้วยชุดควบคุมเอ็นเอ็กซ์ที โดยใช้หลักการพัฒนาออกแบบเชิงวัตถุที่มีการอธิบายการทำงาน ลงลึกถึงขั้นตอนของการทำงานโดยใช้รูปแบบของแผนภาพที่เข้าใจง่ายพัฒนาเป็น หุ่นยนต์ตรวจจับและจัดการวัตถุ โดยหุ่นยนต์จะต้องตรวจจับวัตถุ แยกแยะวัตถุและจัดการกับวัตถุได้อย่างถูกต้อง

โครงงานนี้พัฒนาหุ่นยนต์ตรวจจับและจัดการวัตถุขึ้นมาให้สามารถตรวจจับและเคลื่อนที่เข้าหาวัตถุเพื่อแยกแยะและจัดการกับเป้าหมายได้อย่างถูกต้องโดยอาศัยการทำงานของอุปกรณ์เซ็นเซอร์ด้วยแสงและอุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่(Ultrasonic) ซึ่งในการทดลองยังมีความแน่นอนอยู่บ้างเนื่องจากสภาพแวดล้อมและรูปทรงของวัตถุที่ตรวจจับ

โครงงานนี้บรรลุวัตถุวัตถุประสงค์ในการพัฒนาระบบตามแนวทางพัฒนาเพื่อเป็นต้นแบบของการพัฒนาอย่างเป็นระบบ โดยใช้ยูเอ็มแอล (UML: Unified Modeling Language) ซึ่งจะทำให้การพัฒนาระบบสมองกลฝังตัวมีประสิทธิภาพมากขึ้น ซึ่งหุ่นยนต์ตรวจจับและจัดการวัคถุ ยังสามารถพัฒนาต่อไปได้อีกในแง่ของการแข่งขันหรือการใช้งาน

5.2 ปัญหา และอุปสรรค

5.2.1 ปัญหาทางด้านฮาร์ดแวร์

อุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่มีข้อจำกัดในการวัดระยะ และการตรวจจับวัตถุ ซึ่งจะไม่สามารถวัดระยะทางของวัตถุที่อยู่ในระยะที่ใกล้กว่า 7 เซนติเมตรได้ ส่งผลทำให้อุปกรณ์เซ็นเซอร์นี้ไม่สามารถตรวจจับหรือมองเห็นวัตถุนั้นได้ และลูกบอลที่มีตำแหน่งซ้อนทับกันอุปกรณ์เซ็นเซอร์จะไม่สามารถตรวจสอบได้ว่ามีลูกบอลซ้อนทับกันอยู่

อุปกรณ์เซ็นเซอร์ด้วยแสงมีข้อจำกัดในเรื่องของระยะทางที่สามารถจะแยกแยะสี โดยวัตถุหรือพื้นสนามจะต้องอยู่ห่างจากอุปกรณ์เซ็นเซอร์ไม่เกิน 3 เซนติเมตร ซึ่งเป็นระยะที่ใกล้มาก ยากต่อการควบคุมหุ่นยนต์ให้เข้าใกล้วัตถุขนาดนั้น เพราะอุปกรณ์เซ็นเซอร์ด้วยคลื่นความถี่ก็มีข้อจำกัดเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2 ปัญหาทางด้านซอฟต์แวร์

เนื่องจากซอฟต์แวร์เอ็นเอ็กซ์เจเป็นรุ่นทดลอง (Beta Version) และยังไม่เป็นที่นิยมใช้กัน ซึ่งคำสั่งสำหรับการพัฒนา และคู่มือการใช้งานยังไม่ถูกพัฒนาให้มีความสมบูรณ์ ทำให้เกิดปัญหาการทำงานของคำสั่งบางอย่างยังไม่สามารถทำงานได้จริง เช่น เมื่อทำการใช้คำสั่งให้มอเตอร์เคลื่อนที่วิ่งไปข้างหน้าพร้อมกันด้วยใช้ชุดคำสั่งเดียว ทำการคอมไพล์ และรันซอฟต์แวร์ มอเตอร์จะหมุนเพียงข้างเดียว ถึงแม้ว่าภาษานี้จะสนับสนุนการพัฒนาระบบเชิงวัตถุก็ตาม แต่การควบคุม อ็อปเจ็กต์นั้นต้องประหยัดพื้นที่หน่วยความจำด้วย เพื่อให้สามารถทดสอบโปรแกรมได้อย่างสมบูรณ์ และยังมีข้อจำกัดอย่างอื่นเช่น ช่องทางเข้า (Input Ports) ที่ 4 ไม่รองรับการใช้เซ็นเซอร์ด้วยคลื่นความถี่ ซึ่งช่องทางเข้าที่สามารถใช้งานเซ็นเซอร์ด้วยคลื่นความถี่มากที่สุดคือช่องทางเข้าที่ 3 ระบบไม่รองรับการใช้อาร์เรย์หลายมิติ ระบบไม่มีการจัดสรรหน่วยความจำที่ดีเมื่อดำเนินงาน (Run) เดียวกันหลายๆ ครั้งจะทำให้ผลการทดลองผิดเพี้ยนไป

5.2.3 ปัญหาทางด้านการพัฒนาโปรแกรม

เนื่องจากชุดคำสั่งที่ใช้ยังเป็นเป็นรุ่นทดลอง (Beta Version) ซึ่งมีพื้นฐานมาจากภาษา จาวา ที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ ซึ่งเมื่อมีการสร้างอ็อบเจกต์ ที่มีการเชื่อมต่อกับช่องทางเข้าเดียวกันมากกว่า 1ครั้งจะทำให้โปรแกรมไม่สามารถดำเนินงานได้ และปัญหาอีกอย่างคือ การจัดการค่าที่ได้รับจากเซ็นเซอร์ต่างๆ เพื่อนำมาตีความต้องมีการเขียนโปรแกรมครอบคุมจัดการให้ดี

5.3 ข้อเสนอแนะ

การพัฒนาโปรแกรมเพื่อควบคุมอุปกรณ์ฮาร์ดแวร์ต่างๆ จำเป็นที่จะต้องทำการทดสอบระบบตลอดเวลาเพื่อหาข้อผิดพลาด และแก้ไขในส่วนต่างๆ ที่เกิดขึ้นให้ได้ โดยจะต้องปรับสมรรถภาพของกระบวนการการควบให้มีประสิทธิภาพมากขึ้น เพื่อรองรับกับสภาพแวดล้อมที่ได้กำหนดเอาไว้ได้หลากหลายรูปแบบ

วัตถุที่ใช้ในการทดลอง เป็นทรงกลมทำให้การสะท้อนคลื่นกลับมาหาเซ็นเซอร์ด้วยคลื่นความถี่ มีความไม่แม่นยำ ซึ่งหากจะให้ได้ผลดีควรจะใช้วัตถุที่เป็นเหลี่ยม เช่น ลูกบาศก์ เป็นต้น จะทำให้เซ็นเซอร์ด้วยคลื่นความถี่ทำงานอย่างถูกต้องแม่นยำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

กิตติ ภักดีวัฒนะกุล และพนิดา พานิชกุล. 2548. **คัมภีร์การพัฒนาระบบเชิงวัตถุด้วย UML และ Java**. 1. กรุงเทพฯ : เคทีพี.

Japan System House Association. 2549. **เทคโนโลยีสมองกลฝังตัว**. แปลจาก Embedded Technology for Development of Embedded System. แปลโดย ธนารัตน์ ธีระมั่นคง. 1. กรุงเทพฯ : สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น).

Lego Group. **The NXT**. 2007. [Online]. Available : http://mindstorms.lego.com/Overview/The_NXT.aspx

Lego Group. **The NXT**. 2007. [Online]. Available : http://nxtasy.org/wp-content/uploads/2006/08/digital%20vs%20analog.pdf

Lego Group. **The NXT**. 2007. [Online]. Available : http://mindstorms.lego.com/Overview/Light_Sensor.aspx

Lego Group. **The NXT**. 2007. [Online]. Available : http://mindstorms.lego.com/Overview/Ultrasonic_Sensor.aspx

Lego Group. **The NXT**. 2007. [Online]. Available : http://mindstorms.lego.com/Overview/Interactive_Servo_Motors.aspx

LeJOS. **NXJ**. 2006. [Online]. Available : http://lejos.sourceforge.net

Storm Shadow Software. **SDLC**. 2004. [Online]. Available : http://stormshadowsoftware.co.uk/350px-Waterfall_model.png

Word Press and Tarski. 2008. [Online]. Available : http://www.bartneck.de/2008/03/04/java-lego-nxt-eclipse-tutorial/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.
การติดตั้ง และเซตค่าซอท์ฟแวร์เพื่อติดต่อกับเอ็นเอ็กซ์ที (NXT)

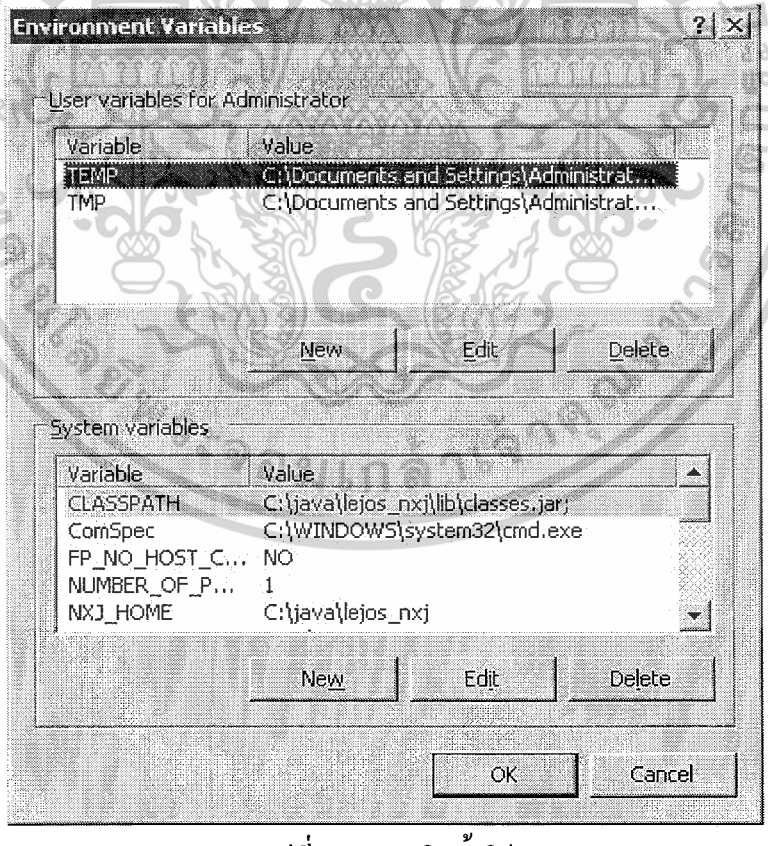
วิธีการติดตั้งโปรแกรมสำหรับการพัฒนาชุดทดลองเอ็นเอ็กซ์ที

1. วิธีการติดตั้ง (Install)

ในเครื่องคอมพิวเตอร์(Computer) ที่ใช้ในการพัฒนา จำเป็นที่จะต้องมีชุดพัฒนาโปรแกรมภาษาจาวา (Java Software Development Kit) ก่อน แล้วเริ่มทำการติดตั้งตามขั้นตอนต่อไปนี้

1. บรรจุลง (Download) ชุดคำสั่งที่ใช้ในการพัฒนาหุ่นยนต์จาก www.lejos.org
2. แกะแฟ้มข้อมูล (File) ที่ได้ไว้ใน C:\
3. ทำการเซตค่าที่วิถี (Path) ให้แก่คอมพิวเตอร์ โดยเข้าไปที่

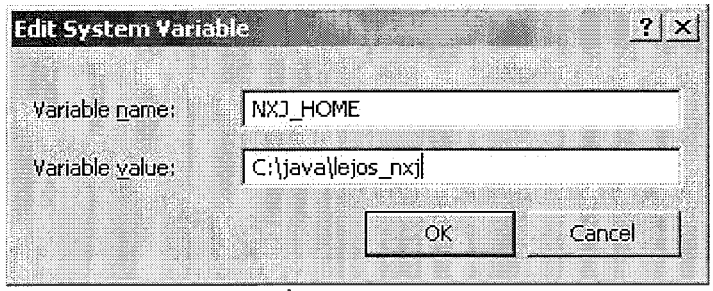
Control Panel > System > Advance >Environment Variables ดังรูปที่ ก.1



รูปที่ ก.1 การติดตั้งวิถี

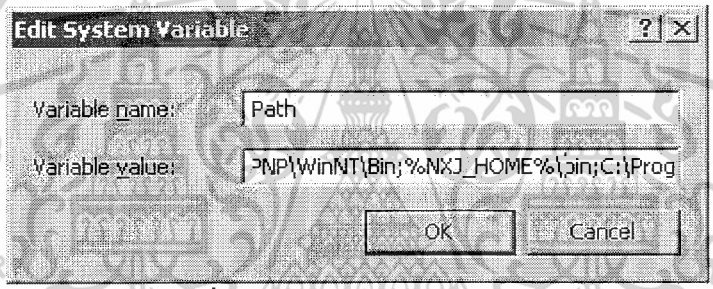
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. โดยเซตค่าในส่วนของ System Variables ให้กดที่ปุ่ม “New” และใส่ข้อมูลดังรูปที่ ก.2



รูปที่ ก.2 การเพิ่มวิถี

5. ตั้งค่าในหมวดของวิถี ดังรูป ก.3 แล้วใส่ค่าดังนี้ “;%NXJ_HOME%\bin” ต่อท้ายจากเดิมที่มีอยู่แล้ว

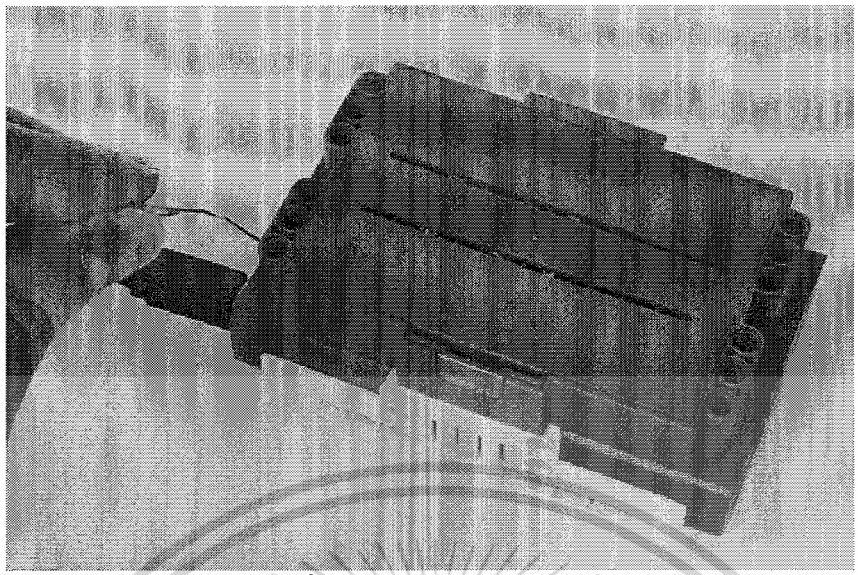


รูปที่ ก.3 การใส่ค่าวิถีลงในระบบ

เมื่อตั้งค่าเสร็จแล้วให้กดปุ่ม “OK” เพื่อออกจากการตั้งค่า จากนั้นทำการบรรจุไดรเวอร์ (Driver) สำหรับเอ็นเอ็กซ์ทีที่ http://libusb-win32.sourceforge.net/#downloads ซึ่งจะมี 2 แฟ้มข้อมูล นำมาติดตั้งที่เครื่อง

เมื่อติดตั้งเสร็จแล้ว ทำการติดตั้ง NXJ Firmware ลงในตัวหุ่นยนต์ โดยจะต้องทำการลบเฟิร์มแวร์ ที่มีอยู่ทิ้งก่อน โดยกดปุ่มที่อยู่ด้านหลังตัวควบคุมหุ่นยนต์ค้างเอาไว้ประมาณ 20 วินาที ดังรูปที่ ก.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.4 วิธีการลบเฟิร์มแวร์

จากนั้นเชื่อมโยง (Link) กับเครื่องคอมพิวเตอร์ที่ติดตั้งตัวโปรแกรมแล้ว ด้วยช่องทางยูเอสบี (USB Port) แล้วไปที่ Start > run >cmd แล้วพิมคำสั่ง nxjflash เพื่อทำการติดตั้งเฟิร์มแวร์ลงบนตัวเอ็นเอ็กซ์ที เมื่อติดตั้งเสร็จแล้วหากพบว่าไม่เฟิร์มแวร์หรือมีอาการค้างอยู่ ให้ถอดแบตเตอรี่ (Battery) ออกแล้วใส่ใหม่อีกครั้ง

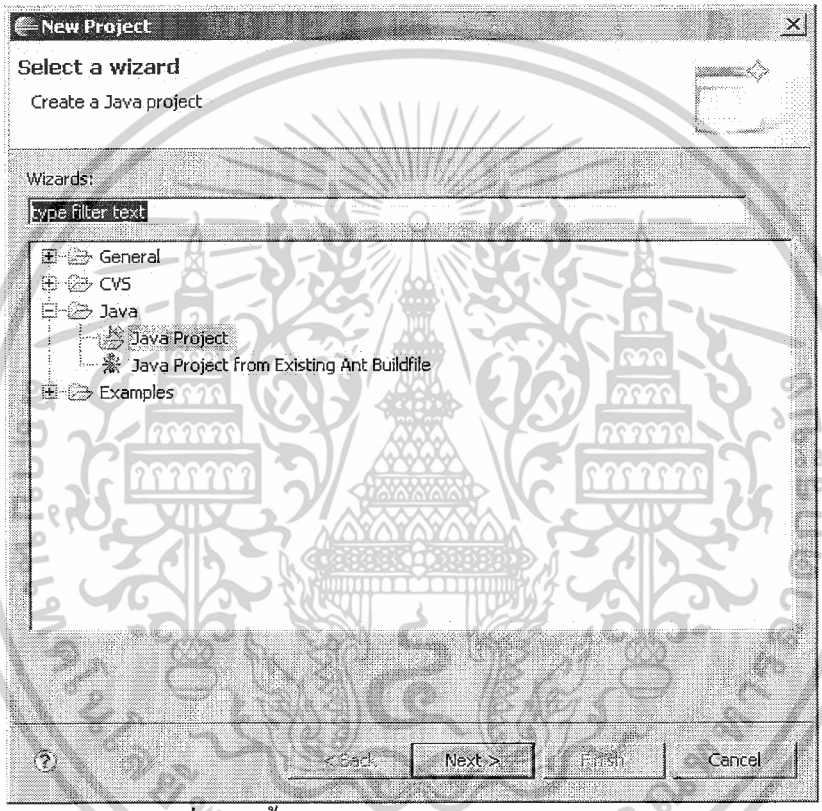
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การตั้งค่าซอฟท์แวร์ เพื่อพัฒนาโปรแกรม

1. บรรจุลงโปรแกรมอีคลิพส์ (Eclipse) ที่ www.eclipse.org

2. ซึ่งโปรแกรมอีคลิพส์ไม่มีการติดตั้งโดยตรงกับเครื่องคอมพิวเตอร์ ถอน (Extract) แฟ้มข้อมูลออกมาก็สามารถเริ่มต้นใช้โปรแกรมได้ทันที

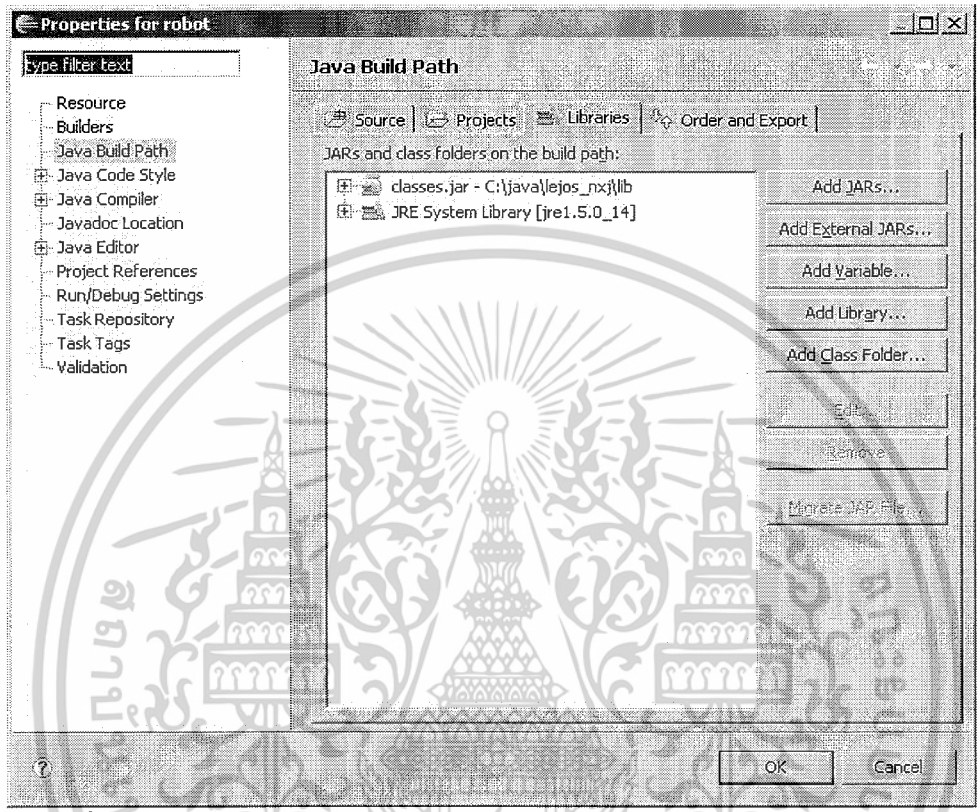
3. เปิดโปรแกรมอีคลิพส์แล้วไปที่ File > New > Project เพื่อสร้างโครงงาน (Project) ใหม่ ซึ่งเป็นโครงงานจาวา (Java Project) ดังรูปที่ ก.5



รูปที่ ก.5 ขั้นตอนการสร้างแฟ้มข้อมูลโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อสร้างโครงงานเสร็จแล้ว จะต้องมีการเพิ่มชุดคำสั่งที่ไว้ใช้ในการพัฒนาหุ่นยนต์ให้กับโปรแกรมอีคลิพส์โดยการไปคลิกขวาที่โครงงานที่เราสร้าง > build path >Configure Build Path… จะได้ผลดังรูปที่ ก.6



รูปที่ ก.6 การเพิ่มชุดคำสั่งที่ไว้ใช้ในการพัฒนาหุ่นยนต์ให้กับโปรแกรมอีคลิพส์

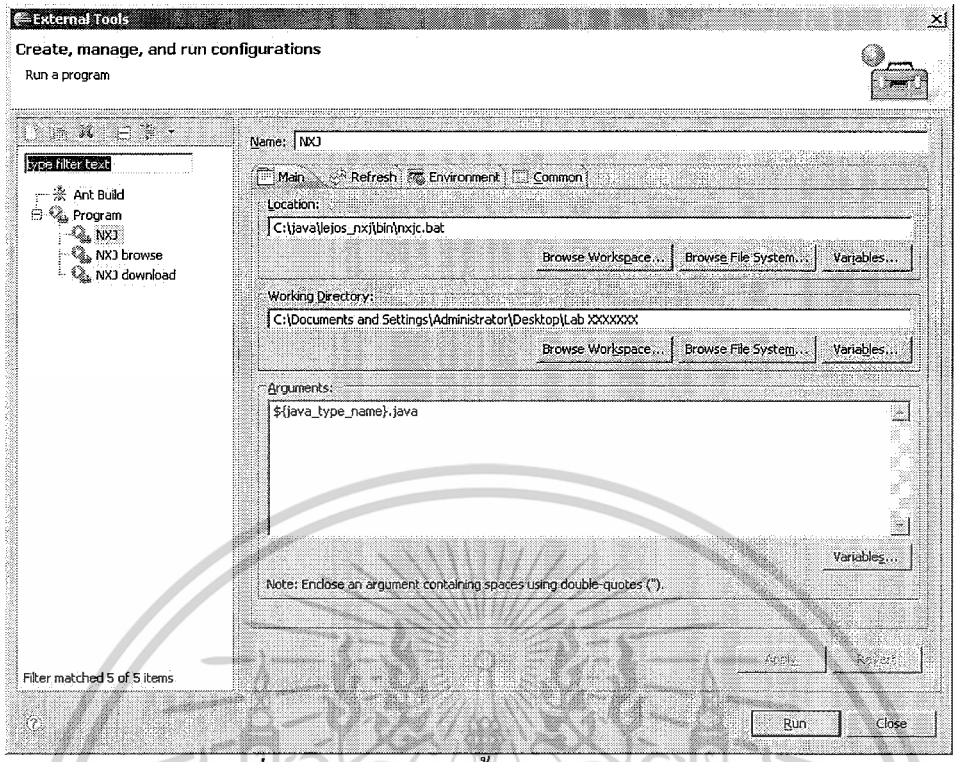
ทำการเพิ่มแฟ้มข้อมูลเสริมที่ Add External JARs และเลือกแฟ้มข้อมูลที่ C:\java\lejos_nxj\lib\classes.jar

เมื่อทำการเพิ่มแฟ้มข้อมูลเสร็จแล้ว ก็มาถึงส่วนของการคอมไพล์ (Compile) ซึ่งสามารถตั้งค่าได้ที่ดำเนินงาน (Run) ที่ External

Tools >Open External Tools Dialog…

โดยตั้งค่าดังรูปที่ ก.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.7 แสดงผลการตั้งค่าให้กับปุ่ม Compile

สำหรับปุ่มคำสั่ง Compile

Working Directory: นั้น เป็นที่อยู่ของโฟล์เดอร์ที่เป็นที่อยู่ของโครงงาน

Location: C:\java\lejos_nxj\bin\nxjc.bat

Argument: ${java_type_name}.java

และเพิ่มปุ่มคำสั่งอีก 1ปุ่มคือ คำสั่งสำหรับการติดตั้งโปรแกรมที่คอมไพล์เสร็จแล้วลงในตัวหุ่นยนต์ โดยใช้วิธีเดิมและตั้งค่าดังนี้

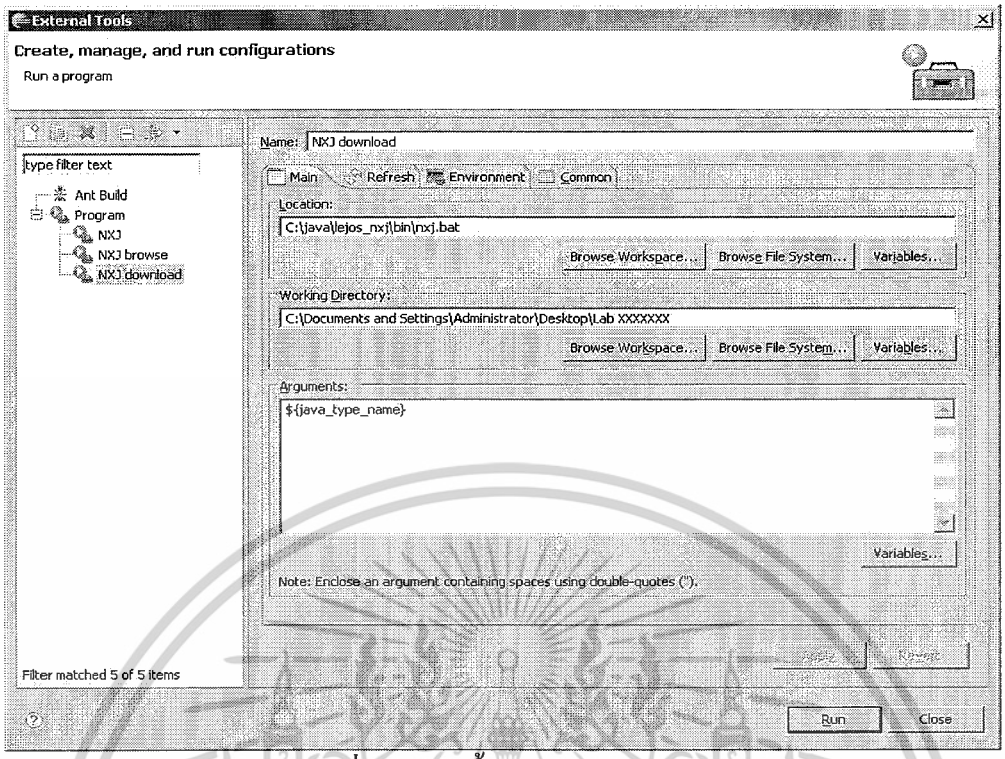
สำหรับการบรรจุลงโปรแกรมลงหุ่นยนต์

Working Directory: นั้น เป็นที่อยู่ของโฟล์เดอร์ที่เป็นที่อยู่ของโครงงาน

Location: C:\java\lejos_nxj\bin\nxj.bat

Argument: ${java_type_name}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

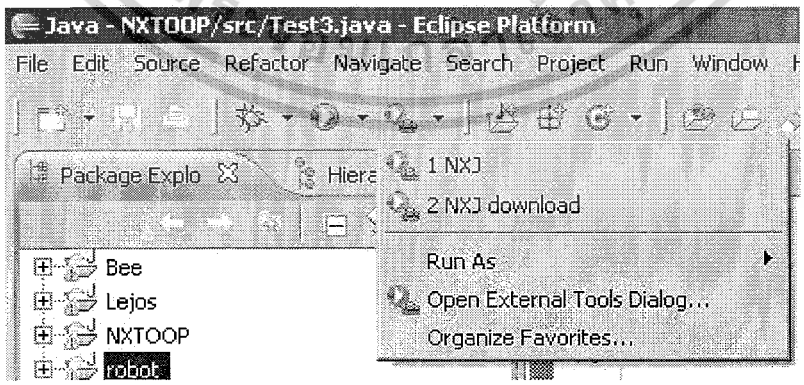


รูปที่ ก.8 การตั้งค่าให้กับปุ่มบรรจุ

เป็นการเสร็จสิ้นการติดตั้ง

3. วิธีใช้งาน

ทำการเขียนโปรแกรมจากนั้นกดที่เครื่องหมายปุ่มเขียวที่มีลูกศรสีขาวมีกล่องสีแดงอยู่ด้านล่าง เลือกตัวคอมไพล์เพื่อสร้างแฟ้มข้อมูลนามสกุลคลาส จากนั้นจึงติดตั้งโปรแกรมลงหุ่นยนต์ด้วยปุ่มคำสั่ง ดังรูปที่ ก.9



รูปที่ ก.9 ขั้นตอนการคอมไพล์ และบรรจุข้อมูลลงในหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.
ตัวอย่างโปรแกรมที่พัฒนา

1. ตัวอย่างโปรแกรมคลาสควบคุมการเคลื่อนที่ (Movement Control Class) ของระบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ

```
Class  MovementControl
import lejos.nxt.Button;
import lejos.nxt.LCD;
import lejos.nxt.Motor;

public class MovementControl {
LightControl l1 = null;
DetectObject2 dec=null;

public void forward1 (){
        Motor.A.forward();
        Motor.C.forward();

}
        public void backward (int x){
         Motor.A.setSpeed(x);
         Motor.C.setSpeed(x);
         Motor.A.backward();
         Motor.C.backward();
 }
 public void Stop (){
         Motor.A.stop();
         Motor.C.stop();
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
public void TurnLeft (int degree){
        Motor.A.rotate(degree);
}
public void TurnRight (int degree){
        Motor.A.rotate(-degree);
}
public void setMotorSpeed(int x){
        Motor.A.setSpeed(x);
        Motor.C.setSpeed(x);
}
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ตัวอย่างโปรแกรมคลาสแยกแยะสีของวัตถุ และพื้นของสนาม (Light Control Class) ของระบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ

```
Class LightControl
import lejos.nxt.LightSensor;
import lejos.nxt.Motor;
import lejos.nxt.SensorPort;

public class LightControl {
boolean line = true;
boolean ball =false;
LightSensor Ball = new LightSensor(SensorPort.S2);
LightSensor field = new LightSensor(SensorPort.S1);
int x ,y,z=30;
        public boolean lineCheck(){
                x = field.readValue();
                if (x>45){
                        line = true;
                }
                else{
                        line = false;
                }
                return line;
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        public int ObjectCheck() throws InterruptedException{
                for(int f=0;f<10;f++){
                        y = Ball.readValue();
                        if(y>z){
                                z = y;
                        }
                }
                return z;
        }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ตัวอย่างโปรแกรมคลาสจัดการวัตถุโดยการผลัก (Push Object Class) ของระบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ

Class PushObj

```
import lejos.nxt.Motor;

public class PushObj {
        public void push() throws InterruptedException{
                Motor.B.setSpeed(900);
                 Motor.B.forward();
                        Thread.sleep(200);
                Motor.B.setSpeed(900);
                 Motor.B.backward();
                                Thread.sleep(270);
                                Motor.B.stop();
        }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ตัวอย่างโปรแกรมคลาสค้นหาตำแหน่ง และทิศทางของวัตถุ (Detect Object Class) ของระบบหุ่นยนต์ตรวจจับ และจัดการวัตถุ

```
Class DetectObject
import lejos.nxt.Button;
import lejos.nxt.LCD;
import lejos.nxt.LightSensor;
import lejos.nxt.Motor;
import lejos.nxt.SensorPort;
import lejos.nxt.UltrasonicSensor;

public class DetectObject2 {
        UltrasonicSensor sonic = new UltrasonicSensor(SensorPort.S3);
        int x[] = new int[50],bb;
        int count,r,tmp,compare;
        public  void search() {
                Motor.A.rotate(300);
                r=0;
                while(r<50){
                        x[r]=sonic.getDistance();
                        Motor.A.rotate(-10);
                        r++;
                }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        for (int i=0;i<50;i++) {
            for (int j=50-1;j>i;j--) {
             if (x[j-1] > x[j]) {
        tmp = x[j];
        x[j] = x[j-1];
        x[j-1] = tmp;
              }
            }
         }
        LCD.clear();
        LCD.drawString("min ="+x[0],1,5);
        LCD.refresh();
        try {
                Thread.sleep(1000);
        } catch (InterruptedException e) {
        }
        for(int r=0;r<50;r++){
                compare=sonic.getDistance();
                Motor.A.rotate(10);
                if(compare-x[0]<=3)
                {
                        Motor.A.rotate(40);
                        break;
                        }
                try {
                        Thread.sleep(200);
                } catch (InterruptedException e) {
                }
                }
}
public  int getDistance() {
        int c=sonic.getDistance();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                return c;

        }

public  void find (){
                for(int r=0;r<60;r++){
                        compare=sonic.getDistance();
                        Motor.C.stop();
                        Motor.A.rotate(10);
                        if(compare<=70)
                        {       //Motor.A.rotate(45);
                                break;
                                }
                        try {
                                Thread.sleep(200);
                        } catch (InterruptedException e) {
                        }
                }
        }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. โปรแกรมคลาสที่ใช้ในการทดลองทั้ง 3 การทดลอง

ประกอบด้วย คลาสตรวจสอบเส้นขอบสนาม คลาสการค้นหา และคลาสการเข้าถึงลูกบอลในสนาม และคลาสการค้นหา และจัดการวัตถุ ดังนี้คำสั่ง (Code) ต่อไปนี้

4.1 คลาสการค้นหา

```
Class Test1
import lejos.nxt.Button;
import lejos.nxt.Motor;

public class Test1 {
        public static void main(String[] args) {
                LightControl l1 = new LightControl();
                MovementControl m = new MovementControl();
                m.setMotorSpeed(900);

                while(!Button.ESCAPE.isPressed()){

                        while(l1.lineCheck()){
                                        m.forward1();
                                }
                        m.backward(900);
                        try {
                                Thread.sleep(300);
                                m.TurnLeft(200);
                                l1.line=true;
                        } catch (InterruptedException e) {
                        }
                }
        }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 คลาสการเข้าถึงลูกบอลในสนาม

```
Class Test2
import lejos.nxt.Button;
import lejos.nxt.LCD;
import lejos.nxt.Motor;

public class Test2 {
        public static void main(String[] args) throws InterruptedException {
                DetectObject2 dc =new DetectObject2();
                MovementControl m = new MovementControl();
                LightControl l1 =new LightControl();
                TrackLine tr = new TrackLine();
                tr.setLight(l1);
                dc.search();
                int c=30 ;
                while(!Button.ESCAPE.isPressed()){
                        if(l1.lineCheck()==true){
                                c= dc.getDistance();
                                        if(c>=dc.x[0]){
                                                LCD.clear();
                                        LCD.drawInt(dc.sonic.getDistance(), 6, 2);
                                        LCD.refresh();

                                            Motor.A.setSpeed(200);
                                            Motor.C.setSpeed(200);
                                        if(c==255){
                                            LCD.drawInt(111, 6, 2);
                                            Motor.C.stop();
                                            LCD.drawInt(222, 6, 2);
                                            Thread.sleep(200);
                                            LCD.drawInt(333, 6, 2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                Motor.A.stop();
                LCD.drawInt(444, 6, 2);
                Thread.sleep(200);

            }
            LCD.drawInt(555, 6, 2);
            Motor.A.forward();
            LCD.drawInt(666, 6, 2);
            Motor.C.forward();
        }else if(c<10){
                        Motor.A.setSpeed(100);
                        Motor.C.setSpeed(100);
                        Thread.sleep(650);
                        Motor.A.stop();
                        Motor.C.stop();
                    if(l1.ObjectCheck()>42){
                        LCD.clear();
                        LCD.drawString("ok",1,5);
LCD.drawString("color"+l1.Ball.readValue(),1,6);
                        LCD.refresh();
                        l1.ball=false;
                        Thread.sleep(200);
                        try {
                                Thread.sleep(2000);
                                LCD.clear();
                        } catch (InterruptedException e) {
                        }
                        System.gc();
                        break;
                    }else{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                    Motor.A.backward();
                                    Motor.C.backward();
                                    Thread.sleep(1500);
                                    m.TurnLeft(60);
                                    Thread.sleep(1000);
                                    dc.find();
                                }
                            }
                        }else{
                            tr.track();
                            dc.find();
                            c= dc.getDistance();
                            m.setMotorSpeed(300);
                            m.forward1();
                            l1.line=true;
                        }
                    }
                }
            }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 คลาสการค้นหา และจัดการวัตถุ

Class Test3

```
import lejos.nxt.Button;
import lejos.nxt.LCD;
import lejos.nxt.Motor;

public class Test3 {
        public static void main(String[] args) throws InterruptedException {
                DetectObject2 dc =new DetectObject2();
                MovementControl m = new MovementControl();
                LightControl l1 =new LightControl();
                dc.search();
                PushObj p = new PushObj();
                int c,d=30 ;
                while(!Button.ESCAPE.isPressed()){
                        if(l1.lineCheck()==true){
                                c= dc.getDistance();
                                        if(c>=dc.x[0]){
                                                        LCD.clear();
                                                LCD.drawInt(dc.sonic.getDistance(), 6, 2);
                                                LCD.refresh();
                                                    Motor.A.setSpeed(200);
                                                    Motor.C.setSpeed(200);
                                                if(c==255){
                                                    LCD.drawInt(111, 6, 2);
                                                    Motor.C.stop();
                                                    LCD.drawInt(222, 6, 2);
                                                    Thread.sleep(200);
                                                    LCD.drawInt(333, 6, 2);
                                                    Motor.A.stop();
                                                    LCD.drawInt(444, 6, 2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                Thread.sleep(200);
            }
            LCD.drawInt(555, 6, 2);
            Motor.A.forward();
            LCD.drawInt(666, 6, 2);
            Motor.C.forward();
        }else if(c<10){
        //////////////
                        Motor.A.setSpeed(100);
                        Motor.C.setSpeed(100);
                        Thread.sleep(650);
                        Motor.A.stop();
                        Motor.C.stop();
                        ////////////////ball///////
                        if(l1.ObjectCheck()>44){
                        Motor.A.backward();
                        Motor.C.backward();
                        Thread.sleep(1500);
                        m.TurnLeft(60);
                        Thread.sleep(1000);
                        dc.find();
                    }else{
                                p.push();
                                break;
                    }
    ////////////////////
            }
    }else{
            Motor.C.rotate(-50);
            dc.find();
            c= dc.getDistance();
            m.setMotorSpeed(300);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ-นามสกุล	นาย สุเมธ ลี้
วัน เดือน ปีเกิด	3 มกราคม 2529
สถานที่การศึกษาปัจจุบัน	2550 คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อ-นามสกุล	นาย พชร ธนบุญญาสกุล
วัน เดือน ปีเกิด	9 สิงหาคม 2527
สถานที่การศึกษาปัจจุบัน	2550 คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้