

ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

ระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสบภัย

RESCUE ROBOT CONTROL SYSTEMS



H005943

โดย



รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2551

b.1217645X  
i.....

รพ.  
๖๗๑๒  
๒๐๕๑

เลขหมู่.....  
เลขทะเบียน..... 05943  
วัน,เดือน,ปี. ๒๖.๓.๒๕๕๑

สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# RESCUE ROBOT CONTROL SYSTEMS



**A SYSTEM DEVELOPMENT PROJECT  
OF THE REQUIREMENT FOR THE DEGREE OF  
MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY  
FACULTY OF INFORMATION TECNOLOGY  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่ **2/ 2008** เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**COPYRIGHT 2009**

**FACULTY OF INFORMATION TECHNOLOGY**

**KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	ระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสบกัย
นักศึกษา	นายสิทธิศักดิ์ แผละออง
รหัสนักศึกษา	50066430
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2551
อาจารย์ที่ปรึกษา	ดร.โอฬาร วงศ์วิรัตน์

### บทคัดย่อ

ความก้าวหน้าทางเทคโนโลยีในปัจจุบันทำให้เกิดนวัตกรรมใหม่ในการประยุกต์ระบบฝังตัวเข้ากับหุ่นยนต์ช่วยเหลือผู้ประสบกัยที่สามารถช่วยลดความเสี่ยงกับอันตรายที่อาจเกิดขึ้นกับผู้ช่วยเหลือ

โครงการพัฒนาระบบงานนี้จึงเป็นการประยุกต์แนวทางการพัฒนาระบบเชิงวัตถุเข้ากับระบบฝังตัวเพื่อจัดสร้างหุ่นยนต์ช่วยเหลือผู้ประสบกัย ซึ่งประกอบด้วย ระบบควบคุมการเคลื่อนที่ ระบบควบคุมการหมุนของกล้องวีดีโอ ระบบตรวจสอบระดับสัญญาณบลูทูธ (Bluetooth) และระดับพลังงานของหุ่นยนต์ รวมทั้งการออกแบบโครงสร้างทางฮาร์ดแวร์ของหุ่นยนต์ โดยใช้ชุดทดลองระบบฝังตัวเลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที (LEGO MINDSTORMS NXT) ที่ติดตั้งเฟิร์มแวร์ (Firmware) เลจอส (LeJOS) และใช้จาวาไอดีอี (Java IDE) เป็นเครื่องมือในการพัฒนาซอฟต์แวร์ ระบบควบคุมหุ่นยนต์ดังกล่าว โครงการพัฒนาระบบงานนี้สามารถใช้เป็นแนวทางในการพัฒนาหุ่นยนต์ช่วยเหลือผู้ประสบกัยในสถานะการณ์จริงต่อไป

<b>Title</b>	Rescue Robot Control Systems
<b>Student</b>	Mr. Sitthisak Paelaong
<b>Student ID.</b>	50066430
<b>Degree</b>	Master of Science
<b>Programme</b>	Information Technology
<b>Major</b>	Information Science
<b>Academic Year</b>	2008
<b>Advisor</b>	Dr. Olarn Wongwirat

## ABSTRACT

Current advancement of technology leads the innovation of embedded systems that are applicable to rescue robot arena. The rescue robot can reduce human risk that may occur to a rescue team.

This development project is the application of an object-oriented systems development approach into the embedded systems. The aim of this project is to implement the rescue robot that composes of a motion control system, camera rotation control system and bluetooth signal strength and robot power monitoring system. The project includes the hardware design of robot structure by using the Lego Mindstroms NXT installing LeJOS firmware. The Java IDE is also employed to implement the software of robot control systems. This development project can be deployed as the approach to develop the rescue robot in practice in the future.

# กิตติกรรมประกาศ

รายงานโครงการพัฒนาระบบฉบับนี้สำเร็จ ด้วยคำแนะนำ และคำปรึกษาที่ดีจากอาจารย์ที่ปรึกษา คร.โอพาร วงศ์วิรัตน์ ตลอดจนปรับปรุงแก้ไขจนรายงาน โครงการเล่มนี้สำเร็จสมบูรณ์อย่างดี ข้าพเจ้ารู้สึกทราบบ้างในความอนุเคราะห์จากท่านอาจารย์ และขอขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณ Lawrie Griffiths และ Gloomyandy จากทีมพัฒนาโครงการเลจอส (LeJOS) สำหรับคำแนะนำต่างๆเกี่ยวกับเลจอส เอนเอ็กซ์เจ (LeJOS NXJ)

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในคณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่อง ทำให้ข้าพเจ้าสามารถทำรายงาน โครงการฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึง ได้มาจากรายงาน โครงการฉบับนี้ ข้าพเจ้าขอบแต่ผู้มีพระคุณทุกท่าน

สิทธิศักดิ์ แพทะอง

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VIII
สารบัญรูป.....	IX
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์.....	1
1.3 ขอบเขตของโครงการ.....	2
1.4 ขั้นตอนของการพัฒนาโครงการ.....	2
บทที่ 2 การพัฒนาระบบโดยใช้ชุดทดลอง ระบบฝังตัวแล็ปท็อป ไมโครคอนโทรลเลอร์ อาร์ดียูเอที.....	3
2.1 ระบบฝังตัว.....	3
2.1.1 โครงสร้างของระบบฝังตัว.....	3
2.1.2 ส่วนประกอบทางด้านฮาร์ดแวร์.....	4
2.1.2.1 เอ็มพียู.....	5
2.1.2.2 หน่วยความจำ.....	5
2.1.2.3 บัส.....	7
2.1.2.4 อุปกรณ์อินพุตและเอาต์พุตพื้นฐาน.....	7
2.1.3 ส่วนประกอบทางด้านซอฟต์แวร์.....	7
2.1.3.1 ระบบปฏิบัติการ.....	8
2.1.3.2 โปรแกรมประยุกต์.....	10
2.2 การพัฒนาระบบด้วยวงจรการพัฒนาระบบ.....	10
2.2.1 กำหนดปัญหาของระบบเดิม (Problem Definition).....	10
2.2.2 การวิเคราะห์ระบบ (Analysis).....	11
2.2.3 การออกแบบระบบ (Design).....	11
2.2.4 การพัฒนาระบบ (Development).....	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา IV และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

	หน้า
2.2.5 การทดสอบระบบ (Testing).....	11
2.2.6 การนำไปใช้งานจริง (Implement) .....	11
2.2.7 การบำรุงรักษาและพัฒนาระบบต่อ (Maintenance).....	12
2.3 เลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที.....	12
2.3.1 เอนเอ็กซ์ทีคอนโทรลเลอร์.....	12
2.3.2 เอนเอ็กซ์ทีมอเตอร์ (NXT Motor).....	14
2.4 เลจอส และเลจอส เอนเอ็กซ์เจ.....	16
2.4.1 JSR-82 .....	17
2.4.2 บลูโคฟ .....	18
2.4.3 ข้อดีของเลจอส เอนเอ็กซ์เจ .....	19
บทที่ 3 การวิเคราะห์และออกแบบระบบ.....	20
3.1 ความต้องการของระบบ.....	20
3.1.1 ความต้องการที่เป็นหน้าที่หลัก (Functional Requirement).....	20
3.1.2 ความต้องการที่ไม่ใช่หน้าที่หลัก (Non-Functional Requirement).....	20
3.2 การวิเคราะห์และออกแบบโครงสร้างหุ่นยนต์.....	20
3.3 การวิเคราะห์และออกแบบระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสาภัย.....	23
3.3.1 ภาพรวมของระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสาภัย.....	23
3.3.2 การวิเคราะห์ระบบ.....	27
3.3.3 ยูสเคสไดอะแกรม (Use Case diagram).....	28
3.3.4 คำอธิบายยูสเคส (Use Case Description).....	29
3.3.5 คลาสไดอะแกรม (Class Diagram).....	33
3.3.6 ไดอะแกรมส่วนสร้างและยกเลิกการเชื่อมต่อ.....	36
3.3.7 ไดอะแกรมส่วนการส่งและรับคำสั่งระหว่างฝั่งผู้ควบคุมและฝั่งหุ่นยนต์.....	39
3.3.8 ไดอะแกรมส่วนการตรวจสอบค่าพลังงานของแบตเตอรี่และ ระดับสัญญาณของการสื่อสารไร้สายแบบบลูทูธ.....	41
3.3.9 ไดอะแกรมส่วนการควบคุมการเคลื่อนที่ของหุ่นยนต์.....	42
3.3.10 ไดอะแกรมส่วนการควบคุมการหมุนกลิ้งวีดีโอ.....	45
3.3.11 ไดอะแกรมส่วนการทำซิงโครไนซ์การทำงานของระบบ.....	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา V และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

	หน้า
3.3.12 ไดอะแกรมส่วนการควบคุมผ่านจอยสติค.....	48
3.4 การออกแบบสภาพแวดล้อมที่ใช้ในการทดลอง.....	49
บทที่ 4 การทดลองและผลการทดลอง.....	51
4.1 การทดลองเกี่ยวกับการเชื่อมต่อระหว่างเครื่องควบคุม และหุ่นยนต์ผ่านการสื่อสาร ไร้สายแบบบลูทูธ.....	51
4.1.1 การทดลองสร้างและยกเลิกการเชื่อมต่อ.....	51
4.1.2 การทดลองส่งข้อมูลระดับสัญญาณบลูทูธ และค่าแรงดันไฟฟ้าของแบตเตอรี่ระหว่างเครื่อง.....	54
4.1.3 การทดลองกลไกการทำงานเมื่อการเชื่อมต่อถูกตัดขาด.....	55
4.2 การทดลองควบคุมการเคลื่อนที่ของหุ่นยนต์.....	57
4.3 การทดลองควบคุมการหมุนกลิ้งวีดีโอ.....	61
4.4 การทดลองการเคลื่อนที่ของหุ่นยนต์ในสภาพแวดล้อมจำลอง.....	64
4.5 การทดลองเพื่อศึกษาระยะเวลาการทำงานอย่างต่อเนื่อง ของหุ่นยนต์ช่วยเหลือผู้ประสบภัย.....	67
4.6 การทดลองเพื่อศึกษาพฤติกรรมการตัดขาด ของสัญญาณบลูทูธเมื่อหุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุม.....	70
4.7 การทดลองกลไกการป้องกันหุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุม.....	73
บทที่ 5 สรุปผลและข้อเสนอแนะ.....	74
5.1 สรุปผลโครงการ.....	74
5.2 ปัญหาและอุปสรรค.....	74
5.3 ข้อเสนอแนะ.....	75
บรรณานุกรม.....	76
ภาคผนวก.....	77
ภาคผนวก ก.ขั้นตอนการติดตั้ง LeJOS NXJ บน Windows XP.....	78
ภาคผนวก ข.ขั้นตอนการติดตั้งเฟิร์มแวร์ LeJOS NXJ บน LEGO MINDSTORMS NXT.....	84
ภาคผนวก ค.ขั้นตอนการปรับแต่ง Netbeans เพื่อใช้งานร่วมกับ LeJOS NXJ.....	89

## สารบัญ (ต่อ)

	หน้า
ภาคผนวก จ.คู่มือการใช้งานระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสพภัย.....	105
ประวัติผู้เขียน.....	108



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา **VII** ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง

ตารางที่	หน้า
3.1 แสดงการออกแบบส่วนการขับเคลื่อนหุ่นยนต์.....	21
3.2 แสดงการออกแบบส่วนการหมุนกล้อวงวีดีโอ.....	22
3.3 แสดงรูปแบบคำสั่งที่ส่งระหว่างเครื่องควบคุมและหุ่นยนต์.....	25
3.4 การวิเคราะห์ความต้องการและฟังก์ชันการทำงานของระบบ.....	27
3.5 คำอธิบายยูสเคสการสร้างการเชื่อมต่อ.....	29
3.6 คำอธิบายยูสเคสการยกเลิกการเชื่อมต่อ.....	29
3.7 คำอธิบายยูสเคสการส่งและรับคำสั่ง.....	30
3.8 คำอธิบายยูสเคสควบคุมการเคลื่อนที่ของหุ่นยนต์.....	31
3.9 คำอธิบายยูสเคสควบคุมการหมุนกล้อวงวีดีโอ.....	31
3.10 คำอธิบายยูสเคสตรวจสอบพลังงานและสัญญาณบลูทูธ.....	32
3.11 คำอธิบายยูสเคสควบคุมผ่านจอยสติค.....	32
4.1 แสดงการทดลองและผลการทดลองเกี่ยวกับการเชื่อมต่อ ระหว่างเครื่องควบคุมและหุ่นยนต์ผ่านการสื่อสารไร้สายแบบบลูทูธ.....	51
4.2 แสดงการทดลองและผลการทดลองส่งข้อมูลระดับสัญญาณบลูทูธ และค่าแรงดันไฟฟ้าของแบตเตอรี่ระหว่างเครื่องควบคุมและหุ่นยนต์.....	54
4.3 แสดงการทดลองและผลการทดลองกลไกการทำงานเมื่อการเชื่อมต่อถูกตัดขาด.....	56
4.4 แสดงการทดลองและผลการทดลองควบคุมการเคลื่อนที่ของหุ่นยนต์.....	58
4.5 แสดงการทดลองและผลการทดลองควบคุมการหมุนกล้อวงวีดีโอ.....	62
4.6 แสดงการทดลองและผลการทดลองการเคลื่อนที่ของหุ่นยนต์ในสภาพแวดล้อมจำลอง.....	65
4.7 แสดงรูปแบบการแสดงผลกับระดับแรงดันไฟฟ้า.....	69
4.8 แสดงการทดลองเพื่อศึกษาพฤติกรรมการตัดขาด ของสัญญาณบลูทูธเมื่อหุ่นยนต์เคลื่อนที่ออกจากระยะควบคุม.....	70

# สารบัญรูป

รูปที่	หน้า
2.1 โครงสร้างโดยสังเขปของระบบฝังตัว.....	4
2.2 สถาปัตยกรรมพื้นฐานทางด้านฮาร์ดแวร์ของระบบฝังตัว .....	4
2.3 การทำงานของเอ็มพียู.....	5
2.4 การแบ่งประเภทหน่วยความจำ .....	6
2.5 ส่วนประกอบของซอฟต์แวร์.....	8
2.6 ฟังก์ชันการทำงานหลักของเคอร์เนล .....	9
2.7 โครงสร้างของดีไวซ์ไดรเวอร์ .....	10
2.8 เอนเอ็ทซ์ทีคอน โทรลเลอร์ .....	12
2.9 คอนเนคเตอร์แบบ RJ12 .....	13
2.10 แบตเตอรี่ชนิดลิเทียม ไอออน โพลีเมอร์ .....	13
2.11 ความสัมพันธ์ของส่วนการทำงานต่างๆของเอนเอ็ทซ์ทีคอน โทรลเลอร์ .....	14
2.12 เอนเอ็ทซ์ทีมอเตอร์.....	14
2.13 แสดงส่วนประกอบภายในของเอนเอ็ทซ์ทีมอเตอร์ .....	15
2.14 แสดงจุดศูนย์กลางมวลของเอนเอ็ทซ์ทีมอเตอร์ .....	15
2.15 กราฟแสดงความสัมพันธ์ระหว่างความเร็วใน การหมุนของเอนเอ็ทซ์ทีมอเตอร์และค่าแรงดันไฟฟ้า.....	16
2.16 Lego MINDSTORMS Robotics Invention System (RIS).....	16
2.17 สเตคไดอะแกรม (Stack diagram) การทำงานของบลูโคพ .....	18
3.1 ส่วนการขับเคลื่อนทั้ง 4 ล้อ.....	21
3.2 ส่วนควบคุมการหมุนของล้อในแนวตั้ง (บน-ล่าง).....	22
3.3 ส่วนควบคุมการหมุนของล้อในแนวระนาบ (ซ้าย-ขวา).....	22
3.4 การควบคุมการหมุนของล้อทั้งแนวระนาบและแนวตั้ง.....	23
3.5 ภาพรวมของการควบคุมหุ่นยนต์ช่วยเหลือผู้ประสาบภัย.....	23
3.6 แสดงยูสเคสไดอะแกรมของระบบควบคุมการเคลื่อนที่.....	28
3.7 แสดงคลาสไดอะแกรมของระบบควบคุมหุ่นยนต์ ช่วยเหลือผู้ประสาบภัยฝังเครื่องควบคุม.....	33
3.8 แสดงคลาสไดอะแกรมของระบบควบคุมหุ่นยนต์ ช่วยเหลือผู้ประสาบภัยฝังหุ่นยนต์.....	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา IX จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.9 แสดงแอกทิวทัศน์ไดอะแกรมของการสร้างการเชื่อมต่อไร้สายแบบบลูทูธ.....	36
3.10 แสดงแอกทิวทัศน์ไดอะแกรมของการยกเลิกการเชื่อมต่อไร้สายแบบบลูทูธ.....	37
3.11 แสดงซีเควณไดอะแกรมของการสร้างและยกเลิกการเชื่อมต่อไร้สายแบบบลูทูธ.....	37
3.12 แสดงแอกทิวทัศน์ไดอะแกรมของการส่งและรับคำสั่งระหว่าง ฝั่งผู้ควบคุม (Operator) และฝั่งหุ่นยนต์ (Robot).....	39
3.13 แสดงซีเควณไดอะแกรมของยูสเคสการส่งและรับคำสั่ง.....	40
3.14 แสดงแอกทิวทัศน์ไดอะแกรมของการตรวจสอบค่าพลังงานของแบตเตอรี่ และระดับสัญญาณของการสื่อสารไร้สายแบบบลูทูธ.....	41
3.15 แสดงซีเควณไดอะแกรมของการตรวจสอบค่าพลังงานของแบตเตอรี่ และตรวจสอบระดับสัญญาณของการสื่อสารไร้สายแบบบลูทูธ.....	41
3.16 แสดงแอกทิวทัศน์ไดอะแกรมการควบคุมการเคลื่อนที่ ของเอนเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 1.....	42
3.17 แสดงแอกทิวทัศน์ไดอะแกรมการควบคุมการเคลื่อนที่ ของเอนเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 2.....	43
3.18 แสดงซีเควณไดอะแกรมของการขับเคลื่อนหุ่นยนต์.....	43
3.19 แสดงลักษณะการหมุนของหุ่นยนต์.....	44
3.20 แสดงแอกทิวทัศน์ไดอะแกรมการหมุนกลิ้งวีดีโอ ของเอนเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 1.....	45
3.21 แสดงแอกทิวทัศน์ไดอะแกรมการหมุนกลิ้งวีดีโอ ของเอนเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 2.....	46
3.22 แสดงซีเควณไดอะแกรมของการหมุนกลิ้งวีดีโอ.....	46
3.23 แสดงซีเควณไดอะแกรมของการทำซิงโครไนซ์การทำงานฝั่งเครื่องควบคุม.....	47
3.24 แสดงซีเควณไดอะแกรมของการทำซิงโครไนซ์การทำงานฝั่งหุ่นยนต์.....	48
3.25 แสดงแอกทิวทัศน์ไดอะแกรมการควบคุมผ่านจอยสติ๊ก.....	48
3.26 แสดงขนาดส่วนต่างๆของสนามทดสอบ.....	49
3.27 แสดงลักษณะของสนามทดสอบที่มีการจำลองสภาพแวดล้อม.....	50
4.1 แสดงสถานะการเชื่อมต่อเสร็จสมบูรณ์ฝั่งเครื่องควบคุม.....	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา X และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.2 แสดงสถานะการเชื่อมต่อเสร็จสมบูรณ์ฝั่งหุ่นยนต์.....	52
4.3 แสดงสถานะการยกเลิกการเชื่อมต่อเสร็จสมบูรณ์ฝั่งเครื่องควบคุม.....	52
4.4 แสดงสถานะการยกเลิกการเชื่อมต่อเสร็จสมบูรณ์ฝั่งหุ่นยนต์.....	52
4.5 แสดงสถานะรอการเชื่อมต่อฝั่งหุ่นยนต์.....	53
4.6 แสดงส่วนการสร้างและยกเลิกการเชื่อมต่อของโปรแกรมฝั่งเครื่องควบคุม.....	53
4.7 แสดงค่าระดับสัญญาณบลูทูธและค่าระดับแรงดันไฟฟ้า ของแบตเตอรี่ที่ฝั่งเครื่องควบคุม.....	54
4.8 แสดงส่วนแสดงระดับสัญญาณบลูทูธ ระดับแรงดันไฟฟ้าของแบตเตอรี่ และส่วนกำหนดช่วงเวลากการส่งข้อมูลของ โปรแกรมฝั่งเครื่องควบคุม.....	55
4.9 แสดงการปิดการรับส่งสัญญาณบลูทูธฝั่งเครื่องควบคุม.....	56
4.10 แสดงสถานะการเชื่อมต่อตัดขาด.....	56
4.11 แสดงสถานะรอการเชื่อมต่อฝั่งหุ่นยนต์.....	56
4.12 แสดงการปิดการทำงานของหุ่นยนต์.....	57
4.13 แสดงสถานะการเชื่อมต่อตัดขาด.....	57
4.14 แสดงผลการตอบกลับการเดินหน้าและถอยหลังมายังฝั่งเครื่องควบคุม.....	58
4.15 แสดงผลการตอบกลับการหมุนซ้ายและขวามายังฝั่งเครื่องควบคุม.....	58
4.16 แสดงผลการตอบกลับการปรับหมุนซ้ายและขวามายังฝั่งเครื่องควบคุม.....	59
4.17 แสดงผลการตอบกลับผลหยุดการเคลื่อนที่มายังฝั่งเครื่องควบคุม.....	59
4.18 แสดงผลการตอบกลับการปรับความเร็วการเคลื่อนที่มายังฝั่งเครื่องควบคุม.....	60
4.19 แสดงส่วนการควบคุมการเคลื่อนที่ของ โปรแกรมฝั่งเครื่องควบคุม.....	60
4.20 แสดงส่วนการควบคุมการหมุนกล้องวีดีโอของ โปรแกรมฝั่งเครื่องควบคุม.....	63
4.21 แสดงความสัมพันธ์ระหว่างการหมุนกล้องกับ ส่วนแสดงทิศทางการหันของกล้องในแนวตั้ง.....	63
4.22 แสดงความสัมพันธ์ระหว่างการหมุนของกล้องกับ ส่วนแสดงทิศทางการหันของกล้องในแนวระนาบ.....	64
4.23 แสดงลักษณะของสนามทดสอบที่จำลองสภาพแวดล้อม และทิศทางการเคลื่อนที่ของหุ่นยนต์.....	65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา **XI** และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.24 แสดงลักษณะของเส้นทางลาดชัน.....	66
4.25 แสดงลักษณะของเส้นทางขรุขระและทางต่างระดับ.....	66
4.26 แสดงผลการขึ้นลงทางลาดชันของหุ่นยนต์.....	66
4.27 แสดงผลการเคลื่อนที่ผ่านเส้นทางขรุขระและทางต่างระดับ.....	67
4.28 แสดงการลักษณะการทดสอบระยะเวลาการทำงานอย่างต่อเนื่องของหุ่นยนต์.....	68
4.29 กราฟแสดงข้อมูลค่าแรงดันไฟฟ้าที่เอนเอ็ชท์ที่คอนโทรลเลอร์ตัวที่ 1 และ 2 ส่งกลับมายังเครื่องควบคุมในทุกๆนาที่.....	69
4.30 แสดงการวัดและบันทึกระยะทางที่หุ่นยนต์เคลื่อนที่.....	71
4.31 กราฟแสดงความสัมพันธ์ระหว่างค่าระดับสัญญาณบลูทูธ และระยะทางที่ระยะทางที่หุ่นยนต์เคลื่อนที่ห่างออกไปจากเครื่องควบคุม.....	72

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ภัยพิบัติที่เกิดจากธรรมชาติ เช่น อัคคีภัย อุทกภัย ภัยจากแผ่นดินไหว เป็นต้น หรือภัยจากแหล่งอื่น ๆ รวมทั้งภัยที่เกิดจากน้ำมือของมนุษย์โลก ย่อมสร้างความเสียหายแก่ทรัพย์สินในทุก ๆ ด้าน ที่ยิ่งร้ายไปกว่านั้น คือความเสียหายทางด้านชีวิตที่เรียกกลับคืนมาไม่ได้ แต่เมื่อภัยเหล่านี้เกิดขึ้นแล้ว การกู้ภัยคือสิ่งที่สำคัญต่อเหตุการณ์นั้นๆเป็นอย่างมาก ปัจจุบันการกู้ภัยเป็นไปในรูปแบบของมูลนิธิแต่ละหน่วยงานที่มีอาสาสมัครเข้าช่วยเหลือชีวิตจากเหตุภัยพิบัติต่าง ๆ ถึงอย่างไรก็ตามในการกู้ภัยเช่นนี้ก็เสี่ยงกับอันตรายที่ไม่คาดคิดซึ่งอาจเกิดขึ้นกับอาสาสมัครได้ตลอดเวลา ดังนั้นจึงมีการคิดค้น และมีการพัฒนาหุ่นยนต์ให้มีประสิทธิภาพใกล้เคียงกับการทำงานของมนุษย์โดยนำความรู้ด้านวิทยาศาสตร์และเทคโนโลยีมาประยุกต์ใช้ ซึ่งโครงสร้างหุ่นยนต์ที่สร้างขึ้นจะต้องมีความสามารถในการช่วยเหลือค้นหาผู้รอดชีวิตหรือผู้เสียชีวิต ที่ตกค้างอยู่ในซากปรักหักพังต่าง ๆ ซึ่งเป็นการปฏิบัติงานที่ท้าทายและมีประโยชน์ต่อสังคม แต่เนื่องจากปัจจุบันการพัฒนาาระบบฝังตัวของหุ่นยนต์ยังขาดการศึกษาและพัฒนาอย่างเป็นระบบ รวมทั้งไม่เป็นไปตามมาตรฐาน ทำให้การพัฒนางานในด้านนี้ล่าช้า ขาดประสิทธิภาพ ขาดต่อการเปลี่ยนแปลงแก้ไข และสิ้นเปลืองงบประมาณในการพัฒนา

ดังนั้นโครงการนี้จึงเป็นการประยุกต์แนวทางการพัฒนาระบบเข้ากับการพัฒนาระบบฝังตัวของหุ่นยนต์ เพื่อให้การพัฒนาเป็นไปตามมาตรฐาน มีประสิทธิภาพ และได้ผลลัพธ์เป็นไปตามความต้องการของระบบที่วางไว้

### 1.2 วัตถุประสงค์

1. เพื่อประยุกต์การออกแบบและพัฒนาระบบเชิงวัตถุเข้ากับระบบฝังตัวของหุ่นยนต์ช่วยเหลือผู้ประสบภัย
2. เพื่อออกแบบและพัฒนาโครงสร้างของหุ่นยนต์ช่วยเหลือผู้ประสบภัยโดยใช้ชุดทดลองระบบฝังตัวเลโก้ มายสตรอม รูนเอนเอ็กซ์ที (LEGO MINDSTORMS NXT)
3. เพื่อออกแบบและพัฒนาระบบควบคุมการเคลื่อนที่ ระบบควบคุมการหมุนของกล้องวิดีโอ ระบบตรวจสอบระดับสัญญาณบลูทูธ (Bluetooth) และพลังงานของหุ่นยนต์ช่วยเหลือผู้ประสบภัยโดยใช้ชุดทดลองระบบฝังตัวเลโก้ มายสตรอม รูนเอนเอ็กซ์ที ที่ติดตั้งเฟิร์มแวร์ (Firmware) ที่มีชื่อว่าเล จอส (LeJOS)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เพื่อเป็นแนวทางในการพัฒนาหุ่นยนต์ช่วยเหลือผู้ประสบภัยในสถานการณ์จริงต่อไป

### 1.3 ขอบเขตของโครงการ

พัฒนาระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสบภัยให้สามารถควบคุมการเคลื่อนที่ของหุ่นยนต์และการหมุนของกล้องวิดีโอได้อย่างถูกต้อง และในขณะทำการควบคุมสามารถทราบได้ถึงระดับสัญญาณบลูทูธและพลังงานของหุ่นยนต์ที่เหลืออยู่ พร้อมทั้งพัฒนารูปแบบคำสั่ง (Command pattern) ที่รับส่งระหว่างเครื่องควบคุมและหุ่นยนต์ ให้สามารถปฏิบัติงานได้จริงในสภาพแวดล้อมจำลองที่สร้างขึ้น

### 1.4 ขั้นตอนของการพัฒนาโครงการ

1. ศึกษาระบบฝังตัว(Embedded Systems) และแนวทางในการประยุกต์ใช้งาน
2. ศึกษากระบวนการพัฒนาระบบเชิงวัตถุ
3. ศึกษาส่วนประกอบต่างๆและหลักการทำงานของชุดทดลองระบบฝังตัวเลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที (LEGO MINDSTORMS NXT)
5. ศึกษาการเขียนโปรแกรมผ่านเฟิร์มแวร์ (Firmware) เลจอส (LeJOS) เพื่อควบคุมการทำงานของหุ่นยนต์
6. ศึกษาและวิเคราะห์ความต้องการของระบบหุ่นยนต์ช่วยเหลือผู้ประสบภัย
7. วิเคราะห์และออกแบบแอปพลิเคชันทั้งฝั่งหุ่นยนต์และเครื่องควบคุม
8. ทดสอบการทำงานของแอปพลิเคชันที่สร้างขึ้น

## บทที่ 2

# การพัฒนาระบบโดยใช้ชุดทดลอง ระบบฝังตัวเลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที

ในบทนี้จะอธิบายถึงทฤษฎีและเทคโนโลยีที่ใช้ในการพัฒนาระบบโดยใช้ชุดทดลองระบบฝังตัวเลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที

### 2.1 ระบบฝังตัว

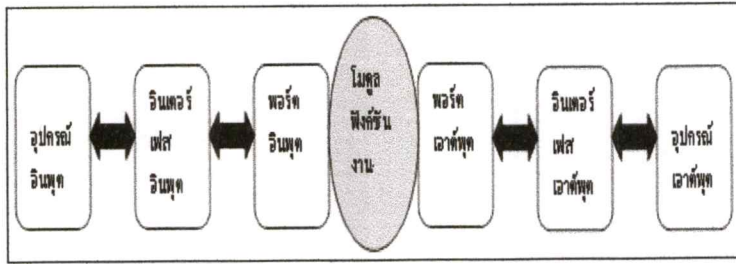
เป็นระบบอิเล็กทรอนิกส์ที่ใช้สำหรับงานควบคุมรวมถึงการแสดงผลการทำงานต่าง ๆ โดยที่ระบบเหล่านี้ถูกใช้เป็นส่วนหนึ่งของระบบและอุปกรณ์ควบคุม เครื่องมือ เครื่องจักรต่าง ๆ การใช้คำว่า “ระบบสมองกลฝังตัว” เนื่องจากระบบเหล่านี้เป็นส่วนหนึ่งของระบบใหญ่ ในหลายกรณีที่ใช้ทั่วไปอาจไม่ทราบว่าอุปกรณ์ควบคุม เครื่องมือ เครื่องจักรรวมถึงระบบใดที่ใช้งานเป็นประจำเหล่านั้นเป็นระบบสมองกลฝังตัว ในบางครั้งแม้แต่ผู้ที่มีความรู้ทางด้านเทคนิคก็ไม่สามารถระบุได้แน่ชัดว่าสิ่งใดมีระบบสมองกลฝังตัวอยู่ จนกว่าจะมีการทำงานและตรวจสอบกับระบบและอุปกรณ์ควบคุมนั้นระบบหนึ่งเลยที่เดียว ระบบสมองกลฝังตัว นี้แม้ไม่ใช่เครื่องคอมพิวเตอร์ แต่ก็มีการประมวลผลหรือโปรแกรมที่ประกอบด้วยชิป (Chip) ที่มีวงจรซับซ้อน โดยจะมีหลักการการทำงาน คือ มีสัญญาณข้อมูลเข้า (Input) จากอุปกรณ์ เซนเซอร์ (Sensor) เข้าสู่ระบบ และมีสัญญาณผลลัพธ์ (Output) ของระบบไปควบคุมบังคับสวิทช์เครื่องควบคุมต่าง ๆ เช่น สวิทช์ เครื่องจักร หรือ วาล์วควบคุมทิศทางรถไฟของท่อทางต่าง ๆ

#### 2.1.1 โครงสร้างของระบบฝังตัว

โดยทั่วไประบบฝังตัวจะประกอบขึ้นจากทรัพยากรระบบ 2 ส่วนคือฮาร์ดแวร์ (Hardware) และซอฟต์แวร์ (Software) ได้แก่ หน่วยประมวลผล (Microcontroller) หน่วยความจำ (Memory) อุปกรณ์อินพุต/เอาต์พุต (Input/Output Devices) โปรแกรม ข้อมูล เป็นต้น นอกจากนี้ อาจมีทรัพยากรด้านเน็ตเวิร์ก (Network) ซึ่งเชื่อมโยงอุปกรณ์ต่าง ๆ เข้าด้วยกัน

การทำงานของระบบฝังตัวส่วนใหญ่จะมีอุปกรณ์อินพุต/เอาต์พุต (Input/Output Devices) ทำหน้าที่ติดต่อกับสิ่งแวดล้อมภายนอก ข้อมูลจะถูกนำเข้ามาและส่งออกผ่านทางอินเตอร์เฟซเข้าสู่พอร์ตอินพุต/เอาต์พุต โดยมีโมดูลฟังก์ชันงานทำหน้าที่ในการประมวลผลการทำงานต่าง ๆ โดยที่อุปกรณ์ที่เป็นอินพุต/เอาต์พุตและ โมดูลฟังก์ชันงานนี้จะถูกควบคุมด้วยส่วนที่เป็นซอฟต์แวร์ของระบบฝังตัว ดังรูปที่ 2.1

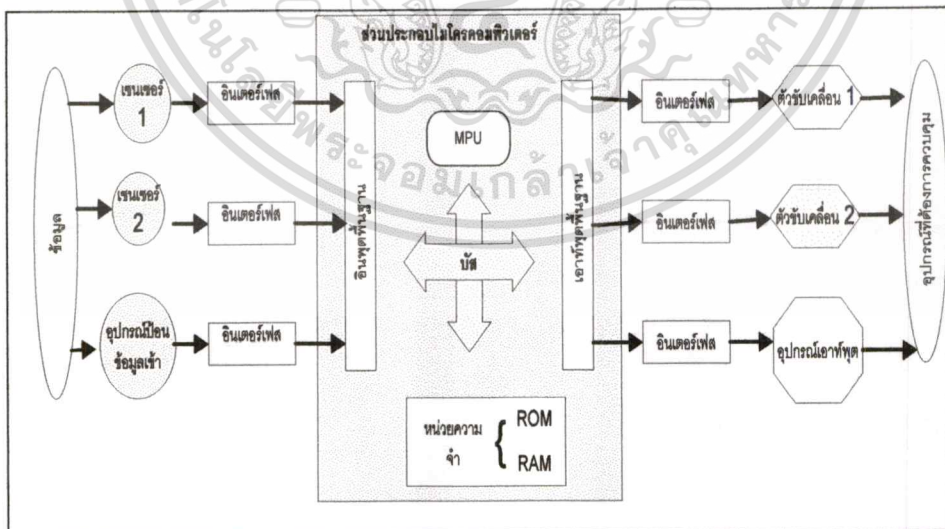
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 2.1** โครงสร้างโดยสังเขปของระบบฝังตัว (Japan System House Association. 2549. แปลและเรียบเรียงโดย ดร.ชนารัตน์ ชีระมันคง : 89)

### 2.1.2 ส่วนประกอบทางด้านฮาร์ดแวร์

ส่วนประกอบทางด้านฮาร์ดแวร์ของระบบฝังตัวจะแตกต่างกันไปตามการประยุกต์ใช้งานแต่จะประกอบไปด้วยโครงสร้างพื้นฐานเช่นเดียวกัน คือ ส่วนไมโครคอมพิวเตอร์ (Microcomputer) ซึ่งภายในประกอบไปด้วย หน่วยประมวลผลหรือเอ็มพียู (MPU: micro-processing unit) หน่วยความจำ (Memory) พอร์ตอินพุต/เอาต์พุตพื้นฐาน (basic I/O) ซึ่งจะติดต่อกันผ่านทางบัส (BUS) และมีการเชื่อมต่อไมโครคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่านทางอินเตอร์เฟซ (Interface) ซึ่งทำหน้าที่ในการแปลงสัญญาณให้อยู่ในรูปแบบที่เหมาะสมต่อการใช้งาน และการทำงานของเอ็มพียู เช่น การปรับจังหวะเวลา การจัดการเหตุการณ์ที่เข้ามาจากหลายช่องสัญญาณ เป็นต้น ดังรูปที่ 2.2



**รูปที่ 2.2** สถาปัตยกรรมพื้นฐานทางด้านฮาร์ดแวร์ของระบบฝังตัว (Japan System House Association. 2549. แปลและเรียบเรียงโดย ดร.ชนารัตน์ ชีระมันคง : 90)

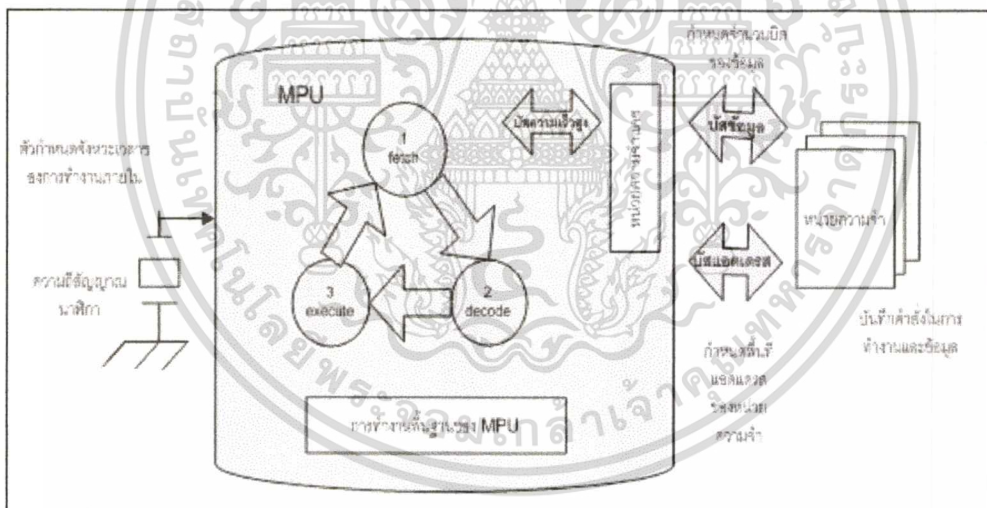
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.2.1 เอ็มพียู

เอ็มพียูเป็นหัวใจหลักในการทำงานของไมโครคอมพิวเตอร์ ทำหน้าที่ในการประมวลผลคำสั่งและควบคุมการทำงานต่าง ๆ โดยปกติแล้วเอ็มพียูจะมีการทำงานโดยเริ่มจากการอ่านคำสั่ง (Fetch) แล้วทำการตีความคำสั่ง (Decode) จากนั้นจึงทำตามคำสั่ง (Execute) นอกจากนี้ยังอาจมีการทำงานแบบไปป์ไลน์ (Pipeline) คือ การประมวลผลหลาย ๆ คำสั่ง ไปพร้อมกันดังรูปที่ 2.3 โดยเอ็มพียูสามารถแบ่งโครงสร้างภายในได้เป็น 2 ประเภทหลัก ๆ คือ CISC และ RISC

1. CISC (Complex Instruction Set Computer) เป็นเอ็มพียูที่ออกแบบมาให้มีคำสั่งในการทำงานที่ซับซ้อนจำนวนมาก เพื่อให้สามารถตอบสนองกับความต้องการโปรแกรมฟังก์ชันต่าง ๆ มีความซับซ้อนสูง ความถี่ของสัญญาณนาฬิกามีหลายลูก และสามารถเข้าถึงหน่วยความจำได้โดยตรงไม่ต้องผ่านรีจิสเตอร์

2. RISC (Reduced Instruction Set Computing) ถูกออกแบบมาให้มีคำสั่งจำนวนน้อย โครงสร้างไม่ซับซ้อน ใช้คำสั่งง่ายๆ เพื่อให้ประมวลผลได้อย่างรวดเร็ว สามารถใช้กับสัญญาณนาฬิกาที่มีความถี่สูงได้ การทำงานมักถูกออกแบบมาให้ใช้รีจิสเตอร์ภายใน



รูปที่ 2.3 การทำงานของเอ็มพียู (Japan System House Association. 2549. แพลตและเรียบเรียง โดย ดร.ชนารัตน์ ชีระมันคง : 93)

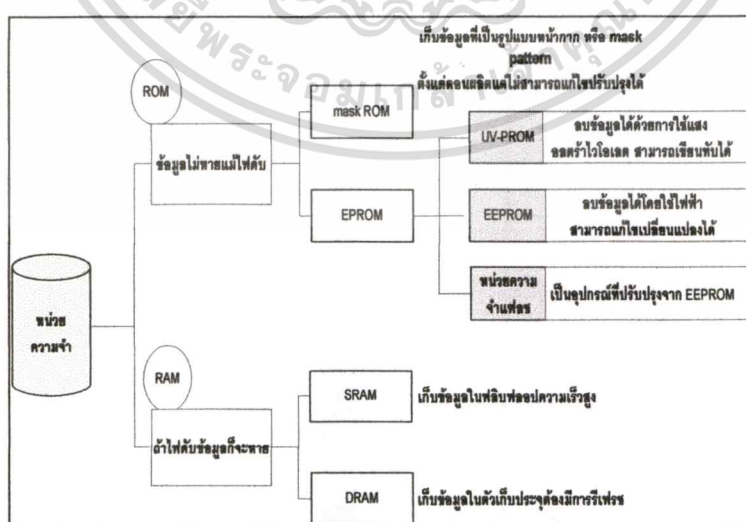
### 2.1.2.2 หน่วยความจำ

หน่วยความจำเป็นส่วนสำหรับเก็บคำสั่งและข้อมูลต่าง ๆ โดยจะเก็บข้อมูลในรูปตัวเลขฐานสอง ซึ่งก็คือสัญญาณทางไฟฟ้านั่นเอง สำหรับการเก็บข้อมูลจะเก็บรวมกันเป็นกลุ่มมิตข้อมูล เช่น 8 บิต รวมกันเป็น 1 ไบต์ และมีการกำหนดตำแหน่งที่อยู่ข้อมูล (Address) เพื่อใช้อย่าง

ถึงในการเขียนหรืออ่านข้อมูลนั้น หน่วยความจำแบ่งได้เป็น 2 ประเภท คือ หน่วยความจำรอม (ROM: Read Only Memory) และแรม (RAM: Random Access Memory) ดังรูปที่ 2.4

1. รอม เป็นหน่วยความจำที่มีคุณสมบัติพิเศษ คือ ข้อมูลจะไม่หายไปไหนแม้ไม่มีไฟเลี้ยง (non-volatile) ดังนั้นรอมจึงเป็นที่เก็บโปรแกรมเริ่มต้นที่จะถูกเอ็มพียูอ่านเข้าไปเพื่อตีความคำสั่ง ขณะเริ่มต้นการทำงานหลังจากที่มีการจ่ายไฟเข้าระบบ ซึ่งโปรแกรมนี้อาจถูกเรียกทำงานก่อนเพื่อ โหลดโปรแกรม โดยทั่วไปรอมแบ่งได้เป็น 2 ประเภทหลัก ๆ คือ mask ROM ซึ่งเป็นรอมที่ถูกเขียนค่าไว้เริ่มต้นตั้งแต่ขั้นตอนการผลิต และ EPROM(Erasable Programmable Read-Only Memory) เป็นรอมที่จะเขียนข้อมูลเข้าภายหลังหรือสามารถลบได้ นอกจากนี้ EPROM ยังสามารถแบ่งได้เป็นประเภทต่าง ๆ ได้แก่ UV-PROM เป็นรอมที่สามารถลบข้อมูลได้ด้วยการฉายแสงอัลตราไวโอเลตหรือยูวี (UV) และ EEPROM เป็นรอมที่สามารถลบหรือเขียนข้อมูลเข้าไปใหม่ด้วยไฟฟ้า

2. แรมเป็นหน่วยความจำที่สามารถเขียนหรืออ่านได้ แต่ข้อมูลจะหายเมื่อไม่มีการจ่ายไฟ (Volatile) เอ็มพียูจะทำงานโดยเก็บหรือเขียนทับค่าของตัวแปรที่ใช้งานหรือเป็นบัฟเฟอร์สำหรับตัวแปรหรือข้อมูลต่างๆ ซึ่งขนาดจะต้องไม่เล็กไปกว่าขนาดที่จำเป็นสำหรับการใช้งาน โดยทั่วไป แรมแบ่งได้เป็น 2 ประเภท คือ SRAM (Static Random Access Memory) และ DRAM (Dynamic Random Access Memory) โดยที่ SRAM เป็นแรมที่ประกอบเป็นวงจรฟลิปฟล็อปทำงานด้วยความเร็วสูง นิยมนำมาเป็นหน่วยความจำแบบแคช(Cache memory) ส่วน DRAM เป็นแรมที่มีการเก็บข้อมูลที่ต้องมีการรีเฟรช(Refresh) เพื่อให้ข้อมูลยังคงอยู่ สามารถสร้างให้เป็นหน่วยความจำที่มีขนาดใหญ่ได้ แต่ความเร็วในการเข้าถึงจะช้ากว่าแบบ SRAM



รูปที่ 2.4 การแบ่งประเภทหน่วยความจำ (Japan System House Association. 2549. แปลและ

เรียบเรียงโดย ดร.ธนรัตน์ วีระมันคง : 94)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.2.3 บัส

บัส(Bus) คือ เส้นทางการติดต่อสื่อสารระหว่างข้อมูล สามารถแบ่งตามตำแหน่งการใช้งานได้เป็น 2 ประเภทหลัก ๆ ได้แก่ บัสภายใน (Internal Bus) เป็นกลุ่มของสัญญาณไมโครคอนโทรลเลอร์ที่ใช้ร่วมกันสำหรับส่งผ่านข้อมูลระหว่างเอ็มพียู หน่วยความจำ อินพุต/เอาต์พุต และบัสภายนอก (External Bus) เป็นกลุ่มของสายสัญญาณที่ใช้ส่งข้อมูลระหว่างไมโครคอมพิวเตอร์และอุปกรณ์ภายนอกต่าง ๆ บัสแบ่งได้เป็น 3 ประเภทหลัก ๆ ได้แก่

1. แอดเดรสบัส (Address bus) ใช้สำหรับระบุตำแหน่งในการรับส่งข้อมูล
2. คาด้าบัส (Data bus) ใช้สำหรับแลกเปลี่ยนข้อมูล
3. คอลโทรลบัส (Control bus) เป็นตัวกำหนดและควบคุม แอดเดรสบัสและคาด้าบัส

### 2.1.2.4 อุปกรณ์อินพุตและเอาต์พุตพื้นฐาน

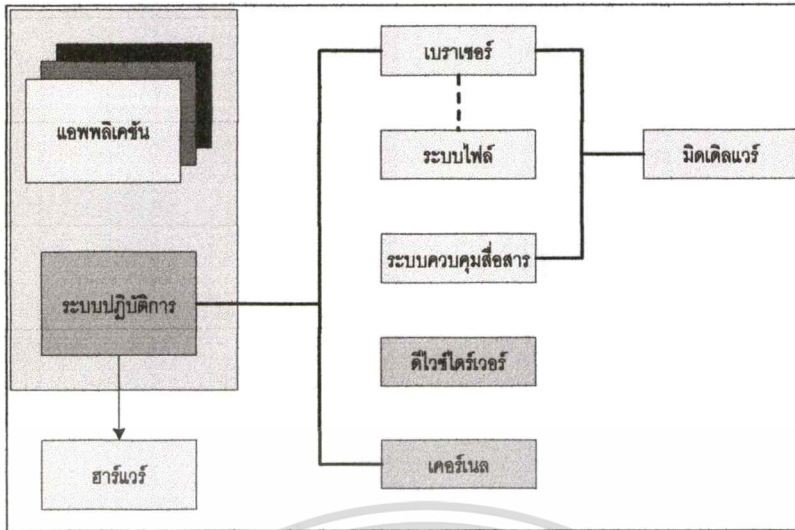
ในระบบฝังตัวนั้นเอ็มพียูจะมีการรับส่งข้อมูลจากภายนอกเข้ามาประมวลผลก่อนที่จะส่งออกไปภายนอกอีกครั้ง ดังนั้นจะต้องมีการแลกเปลี่ยนข้อมูลระหว่างเอ็มพียูกับอุปกรณ์ภายนอกผ่านทางอินเทอร์เฟซ(I/O interface) ซึ่งอุปกรณ์ภายนอกประกอบด้วยอุปกรณ์อินพุตและอุปกรณ์เอาต์พุต

อุปกรณ์อินพุตพื้นฐาน เป็นส่วนรับข้อมูลและสัญญาณต่างๆ ของไมโครคอมพิวเตอร์ โดยจะส่งข้อมูลในรูปแบบของข้อมูลดิจิทัลให้กับเอ็มพียูเพื่อนำไปประมวลผล โดยทั่วไปเรียกว่า อินพุตพอร์ต โดยผ่านอุปกรณ์หรือวงจรไฟฟ้าเพื่อทำหน้าที่แปลงสัญญาณให้อยู่ในรูปแบบที่เหมาะสมเรียกว่า วงจรอินเทอร์เฟซ (Interface circuit) อุปกรณ์อินพุตเช่น เซนเซอร์

อุปกรณ์เอาต์พุตพื้นฐานเป็นช่องทางออกของสัญญาณที่เกิดจากการประมวลผลของเอ็มพียู โดยจะมีการแปลงข้อมูลแบบดิจิทัลให้เป็นสัญญาณทางกายภาพที่เหมาะสมเรียกว่าเอาต์พุตพอร์ต ซึ่งเป็นส่วนสำคัญในการเชื่อมการทำงานของเอ็มพียูกับอุปกรณ์ภายนอก อุปกรณ์เอาต์พุตได้แก่ ตัวขับเคลื่อน (Actuator) เช่นมอเตอร์ จอแสดงผล เป็นต้น

### 2.1.3 ส่วนประกอบของซอฟต์แวร์

ส่วนประกอบของซอฟต์แวร์โดยทั่วไปแบ่งได้เป็น 2 ประเภทใหญ่ ๆ คือ ระบบปฏิบัติการ (Operating system) และ โปรแกรมประยุกต์ (application program) ดังรูปที่ 2.5



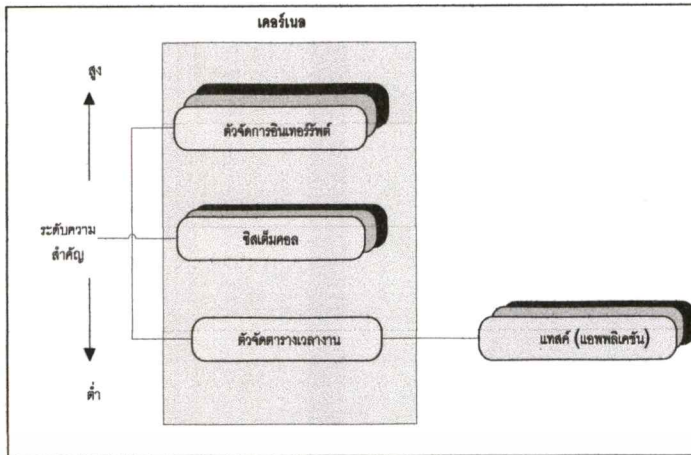
รูปที่ 2.5 ส่วนประกอบของซอฟต์แวร์ (Japan System House Association. 2549.

แปลและเรียบเรียงโดย คร.ชนารัตน์ ชีระมันคง : 95)

### 2.1.3.1 ระบบปฏิบัติการ

ระบบปฏิบัติการเป็นส่วนหนึ่งของซอฟต์แวร์ระบบที่ทำหน้าที่เป็นตัวกลางระหว่างฮาร์ดแวร์และโปรแกรมประยุกต์ทั่วไป บางครั้งอาจเป็นเฟิร์มแวร์ก็ได้ มีหน้าที่หลักๆ คือการจัดสรรทรัพยากรในระบบเพื่อให้บริการซอฟต์แวร์ประยุกต์ ควบคุมการรับส่งและจัดเก็บข้อมูล จัดสรรพื้นที่หน่วยความจำ รวมทั้งทำหน้าที่จัดสรรเวลาในการประมวลผล ระบบปฏิบัติการประกอบด้วยเคอร์เนล (Kernel) มิดเดิลแวร์ (Middleware) และ ดีไวซ์ไดรเวอร์ (Devices Driver)

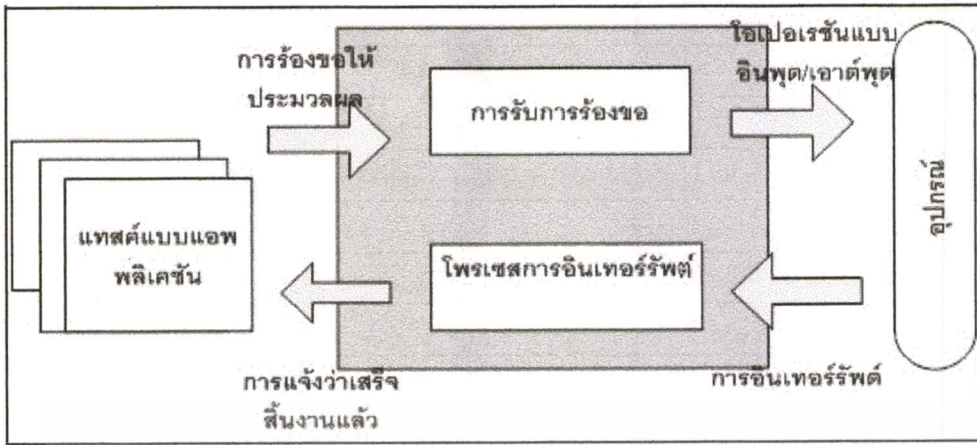
เคอร์เนล เป็น โมดูลกลางที่ทำหน้าที่ในการให้กำเนิดฟังก์ชันพื้นฐานของระบบปฏิบัติการ ให้กำเนิดฟังก์ชันของการจัดการงานแบบจับเคลื่อนด้วยเหตุการณ์ เรียกว่าเรียลไทม์เคอร์เนล (Real-time kernel) หรือเรียลไทม์โอเอส (Real-time Operating System) ซึ่งเป็นส่วนสนับสนุนฟังก์ชันพื้นฐาน ที่ทำให้เกิดการประมวลผลแบบเรียลไทม์ การทำงานหลักของเคอร์เนลจะประกอบไปด้วย ฟังก์ชันการจัดการงานทำหน้าที่ในการจัดการเหตุการณ์ที่เกิดขึ้นพร้อมกันหลายๆ เหตุการณ์ ฟังก์ชันควบคุมการอินเทอร์รัพต์ (Interrupt) ทำหน้าที่จัดการและดูแลอินเทอร์รัพต์ที่เกิดขึ้นตามแต่ละประเภท และฟังก์ชันซิสเต็มคอลล (System call) ทำหน้าที่จัดการการเรียกใช้บริการจากระบบ ดังรูปที่ 2.6



**รูปที่ 2.6** ฟังก์ชันการทำงานหลักของเครือข่าย (Japan System House Association, 2549).  
แปลและเรียบเรียง โดย ดร.ธนารัตน์ ชีระมันคง : 96)

มิดเดิลแวร์ เป็น โมดูลที่สนับสนุนฟังก์ชันทั่วไปของระบบ ทำหน้าที่เป็นตัวกลาง เป็นตัวประสานบริการ(Service) ต่างๆ ให้กับแอปพลิเคชัน โดยผ่านทางเครือข่ายที่ควบคุมการทำงานของระบบปฏิบัติการ ทำให้ใช้งานแอปพลิเคชันต่างๆ ได้สะดวกขึ้น ซึ่งอาจประกอบด้วยไลบรารี (Library) เช่นระบบไฟล์ ฟังก์ชันควบคุมการสื่อสาร เป็นต้น

ดีไวซ์ไครเวอร์ เป็นส่วนควบคุมอุปกรณ์อินพุต/เอาต์พุต เพื่อให้เกิดฟังก์ชันในการรับส่งข้อมูลกับโปรแกรมประยุกต์ บางครั้งเรียกว่าระบบควบคุมอินพุตเอาต์พุต (Input & Output Control System - IOCS) ระบบปฏิบัติการโดยทั่วไปดีไวซ์ไครเวอร์จะเป็นส่วนหนึ่งของระบบปฏิบัติการเลย แต่ถ้าเป็นซอฟต์แวร์ระบบฝังตัวนั้นจะแตกต่างกันไปตามผลิตภัณฑ์และการใช้งานอุปกรณ์ ดังนั้นจึงไม่สามารถใช้ฟังก์ชันของดีไวซ์ไครเวอร์แบบเฉพาะเจาะจงลงไปในระบบปฏิบัติการได้ ส่วนใหญ่จะมีเพียงดีไวซ์ไครเวอร์พื้นฐานหลักๆ เช่น ตัวจับเวลา (timer) อินเตอร์เฟซชนิดซีเรียล (Serial I/F) เป็นต้น สำหรับดีไวซ์ไครเวอร์อื่นๆ นั้นอาจจะต้องการจากซอฟต์แวร์อื่นที่มีอยู่ หรืออาจต้องพัฒนาขึ้นมาเอง โดยดีไวซ์ไครเวอร์ มีโครงสร้างดังรูปที่ 2.7



รูปที่ 2.7 โครงสร้างของดีไวซ์ไครเวอร์ (Japan System House Association. 2549.  
แปลและเรียบเรียงโดย คร.ชนารัตน์ ชีระมันคง : 97)

### 2.1.3.2 โปรแกรมประยุกต์

โปรแกรมประยุกต์เป็นซอฟต์แวร์ที่ใช้กับงานด้านต่าง ๆ ตามความต้องการของผู้ใช้ที่สามารถนำมาใช้ประโยชน์ได้โดยตรง ทำให้การทำงานได้สะดวกขึ้น หรืออาจเป็นซอฟต์แวร์ใช้งานเฉพาะด้านซึ่งผู้ใช้เป็นผู้พัฒนาขึ้นเองเพื่อให้เหมาะสมกับสภาพการทำงานของตนเอง สามารถแบ่งซอฟต์แวร์ประยุกต์ออกเป็นสองกลุ่มคือ ซอฟต์แวร์สำเร็จรูป คือ โปรแกรมที่ถูกสร้างหรือเขียนขึ้นมาเสร็จเรียบร้อยแล้ว พร้อมทั้งจะนำไปใช้งานต่างๆ ได้ทันที เช่น Microsoft Office เป็นต้น และซอฟต์แวร์ที่ใช้งานเฉพาะด้าน เช่น ซอฟต์แวร์ประมวลผลคำ ซอฟต์แวร์ระบบบัญชี เป็นต้น

## 2.2 การพัฒนาระบบด้วยวงจรการพัฒนาระบบ

การพัฒนาระบบงานด้วยวงจรการพัฒนาระบบ (System Development Life Cycle : SDLC) มีขั้นตอนในการดำเนินงานเป็นลำดับ ทำให้การพัฒนาเป็นไปตามลำดับขั้นตอน โดยวงจรการพัฒนาระบบงานมีขั้นตอนการทำงานเรียงตามลำดับ 7 ขั้นตอน ได้แก่

### 2.2.1 กำหนดปัญหาของระบบเดิม (Problem Definition)

ขั้นตอนนี้เป็นการกำหนดขอบเขตของปัญหา สาเหตุของปัญหา ตลอดจนกลยุทธ์ในการแก้ไขปัญหา นักวิเคราะห์ระบบจะต้องศึกษาระบบงานเดิม (Current System) โดยหาเป้าหมายที่ชัดเจนของงานต่าง ๆ ประกอบกับนำคอมพิวเตอร์เข้าไปใช้ในส่วนต่างๆ ของระบบจากการสุ่มตัวอย่าง การสอบถามหาข้อมูล การสัมภาษณ์ การออกแบบสอบถาม การสังเกตพฤติกรรมของผู้ใช้ และสิ่งแวดล้อมเพื่อสืบค้น เก็บรวบรวมข้อมูลที่เป็นความต้องการของระบบจากผู้ใช้ เพื่อกำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัตถุประสงค์ที่สามารถวัดผลได้ ตลอดจนกำหนดขอบเขตของการพัฒนาระบบ ทำเอกสารขอ อนุมัติทำโครงการ งบประมาณ

### 2.2.2 การวิเคราะห์ระบบ (Analysis)

การวิเคราะห์ระบบจะรวบรวมข้อมูลต่าง ๆ ที่ได้จากขั้นตอนที่ 1 มาเขียนเป็นไดอะแกรม การไหลของข้อมูล (Data Flow Diagram) พจนานุกรมข้อมูล (Data Dictionary) และ โครงสร้างการ ตัดสินใจ (Structured decision) มาช่วยในการวิเคราะห์ เพื่อแก้ไขปัญหาให้ถูกต้อง และนักวิเคราะห์ ระบบต้องมีการทำงานร่วมกับผู้ใช้ระบบเพื่อได้ความต้องการจาก ผู้ใช้โดยแท้จริง นำผลการ วิเคราะห์ไปจัดทำเอกสารและให้มีการลงนาม

### 2.2.3 การออกแบบระบบ (Design)

หลังจากการวิเคราะห์ระบบแล้ว ขั้นตอนนี้จะต้องทำการวางโครงสร้างของระบบงาน ทั้ง ในรูปลักษณะทั่วไปและเฉพาะ เพื่อแก้ไขปัญหาที่เกิดขึ้น โดยการแจกแจงรายละเอียดที่แน่ชัดของ แต่ละงาน ซึ่งขั้นตอนนี้จะ ได้ Purpose System เพื่อทำการออกแบบ Output, Input, E-R model และ Database เพื่อให้ได้ระบบงานที่สมบูรณ์ เพื่อส่งขั้นตอนนี้ไปยังโปรแกรมเมอร์ในการเขียนชุดคำสั่ง ต่อไป

### 2.2.4 การพัฒนาระบบ (Development)

ขั้นตอนนี้จะเป็นการทำงานร่วมกันระหว่างโปรแกรมเมอร์ และนักวิเคราะห์ระบบเพื่อ พัฒนาซอฟต์แวร์ ซึ่งจะต้องนำส่วนที่ได้จากการวิเคราะห์ในขั้นตอนที่ 2 และการออกแบบในส่วนที่ 3 มาใช้ โดยโปรแกรมเมอร์จะเป็นผู้เขียน โปรแกรม ตรวจสอบข้อผิดพลาด กำหนดความปลอดภัย ของระบบและทดสอบโปรแกรมรวมถึงทำเอกสาร โปรแกรมสำหรับผู้ใช้ ระบบอีกด้วย

### 2.2.5 การทดสอบระบบ (Testing)

ก่อนที่จะนำระบบที่สร้างขึ้น ไปใช้จริงนั้นจะต้องมีการทดสอบระบบก่อน ซึ่งบางครั้งผู้ ทดสอบอาจเป็นตัวโปรแกรมเมอร์เองหรือในบางกรณีอาจให้ นักวิเคราะห์ระบบ และผู้ใช้ระบบ ทดสอบ ซึ่งการทดสอบมี 3 วิธี คือ Module Test, Component Test และ Final Test

### 2.2.6 การนำไปใช้งานจริง (Implement)

หลังจากทดสอบเสร็จสิ้น จึงนำระบบมาติดตั้งให้แก่ผู้ใช้ระบบ ได้ทดลองใช้จริง และผู้ใช้ ต้องผ่านการทดสอบ ซึ่งถือว่าเป็นขั้นตอนสุดท้ายของนักวิเคราะห์ระบบ ที่ต้องรับผิดชอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.7 การบำรุงรักษาและพัฒนาระบบต่อ (Maintenance)

หลังจากนำระบบใหม่มาติดตั้งให้กับผู้ใช้ระบบ ผู้ใช้ระบบยังไม่คุ้นเคยกับการทำงานของระบบใหม่ ดังนั้นจึงต้องมีการอบรม ให้คำแนะนำอย่างต่อเนื่อง คอยดูแลบำรุงรักษาฐานข้อมูล และช่วยเหลือผู้ใช้ระบบในการปฏิบัติงาน

## 2.3 เลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที

เลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที(LEGO MINDSTORMS NXT)เป็นชุดทดลองระบบฝังตัว ซึ่งประกอบด้วยส่วนประกอบสำคัญซึ่งได้แก่

### 2.3.1 เอนเอ็กซ์ทีคอนโทรลเลอร์



รูปที่ 2.8 เอนเอ็กซ์ทีคอนโทรลเลอร์ (<http://www.active-robots.com/products/lego/nxt-accessories.shtml>)

เอนเอ็กซ์ทีคอนโทรลเลอร์ (NXT Controller) ดังรูปที่ 2.1 ทำหน้าที่ควบคุมส่วนการทำงานต่างๆซึ่งมีส่วนการทำงานย่อยๆดังนี้

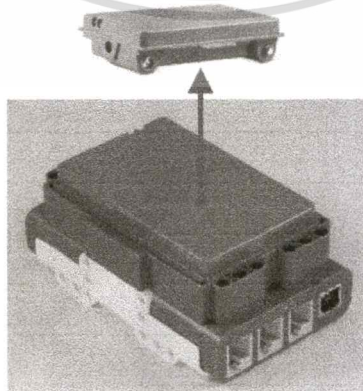
- ใช้ไมโครคอนโทรลเลอร์(Microcontroller) 32 บิต ตระกูล ARM7 เป็นหน่วยประมวลผลหลัก(Main processor)
- ใช้ไมโครคอนโทรลเลอร์ 8 บิต ตระกูล AVR เป็นหน่วยประมวลผลเสริม(Co-processor)
- ใช้บลูทูธชิพ (Bluetooth chip) รุ่น BlueCore 4 (BC4) Class 1
- รองรับ USB(Universal Serial Bus) 2.0 ความเร็วการสื่อสารผ่านพอร์ต(Port)สูงสุด 12 เมกะบิตต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีพอร์ตขาเข้า(Input port)จำนวน 4 พอร์ตสำหรับต่อกับเซนเซอร์ และพอร์ตขาออก(Output port)จำนวน 3 พอร์ตสำหรับต่อกับมอเตอร์
- เอนเอ็็กซ์ทีคอนโทรลเลอร์ทำหน้าที่ควบคุมส่วนการทำงานต่างๆผ่านทางสายคู่บิดเกลียว (Twisted Pair) ชนิดคอนเนคเตอร์แบบอาร์เจ12(RJ12) ดังรูปที่ 2.9
- มีปุ่มกดจำนวน 4 ปุ่มเพื่อรับคำสั่งจากผู้ใช้งาน
- จอแสดงผลขนาด 26 x 40.6 มิลลิเมตร ความละเอียด 100 x 64 พิกเซล
- ลำโพงรองรับเสียงที่มีอัตราการแซมเปิลที่อยู่ในช่วง 2 ถึง 16 กิโลเฮิรต์
- ใช้แหล่งพลังงานจากถ่านอัลคาไลน์(Alkaline) ขนาด AA จำนวน 6 ก้อน หรือ แบตเตอรี่ชนิดลิเทียมไอออนโพลิเมอร์ (Lithium ion polymer) 1400 มิลลิแอมป์ แบบชาร์จซ้ำได้ดังรูปที่ 2.10
- จากส่วนประกอบทั้งหมดของเอนเอ็็กซ์ทีคอนโทรลเลอร์สามารถแสดงความสัมพันธ์ของส่วนการทำงานต่างๆได้ดังรูปที่ 2.11

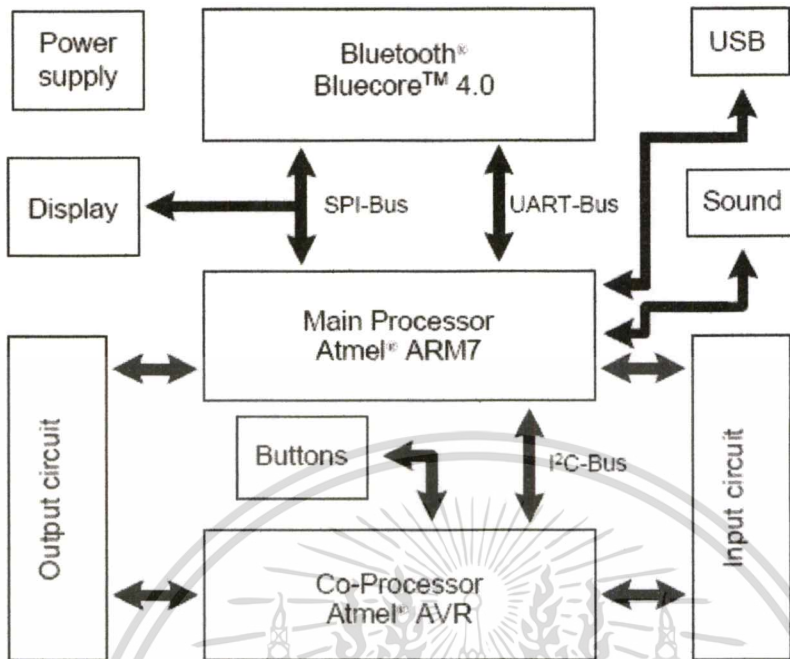


รูปที่ 2.9 คอนเนคเตอร์แบบ RJ12 (Phil Wade, 2006. <http://www.kidtechnic.com/15minuteswithNXT.tml>)



รูปที่ 2.10 แบตเตอรี่ชนิดลิเทียมไอออนโพลิเมอร์ (<http://www.nxtprograms.com/help/parts/9797.html>)

เอกสารนี้เป็นเอกสารที่เผยแพร่เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



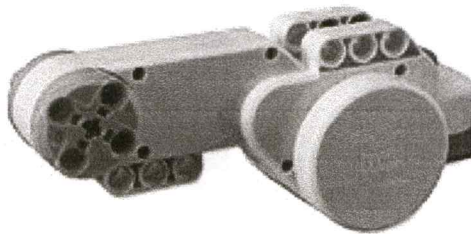
รูปที่ 2.11 ความสัมพันธ์ของส่วนการทำงานต่างๆของเอนเอ็กซ์ทีคอนโทรลเลอร์

([http://mindstormsxt.blogspot.com/2006\\_08\\_01\\_archive.html](http://mindstormsxt.blogspot.com/2006_08_01_archive.html))

### 2.3.2 เอนเอ็กซ์ทีมอเตอร์ (NXT Motor)

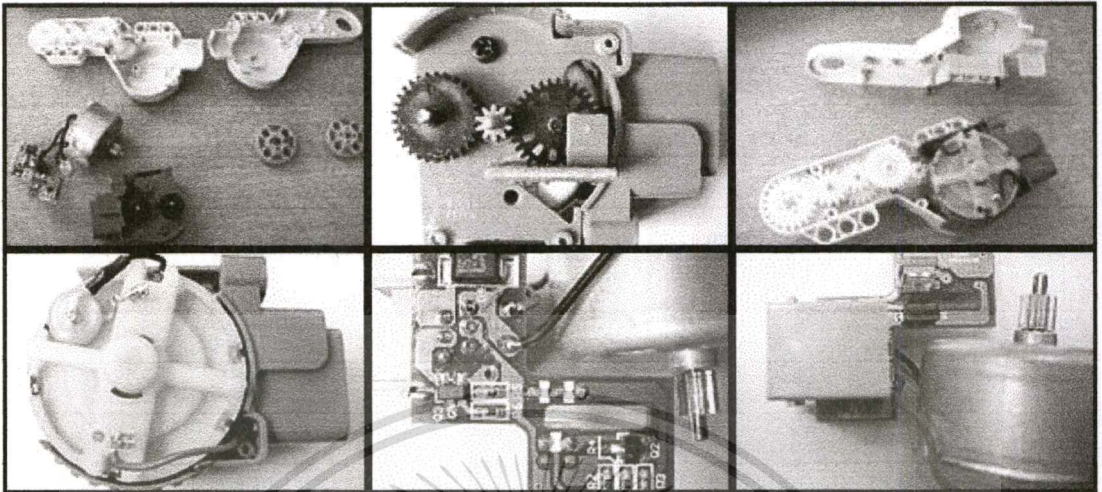
ชุดทดลองระบบฝังตัวเลโก้ มาสเตอร์ม รุ่นเอนเอ็กซ์ที 1 ชุด สามารถเชื่อมต่อกับเอนเอ็กซ์ทีมอเตอร์ดังรูปที่ 2.12 ได้มากที่สุดจำนวน 3 พอร์ต โดยเชื่อมต่อกับพอร์ตขาออก เอนเอ็กซ์ทีมอเตอร์ มีส่วนประกอบภายในดังรูปที่ 2.13 และมีจุดศูนย์กลางมวลดังรูปที่ 2.14

ความเร็วในการหมุนของเอนเอ็กซ์ทีมอเตอร์มีความสัมพันธ์กับค่าแรงดันไฟฟ้าที่เอนเอ็กซ์ทีคอนโทรลเลอร์ทำการจ่ายให้ดังรูปที่ 2.8

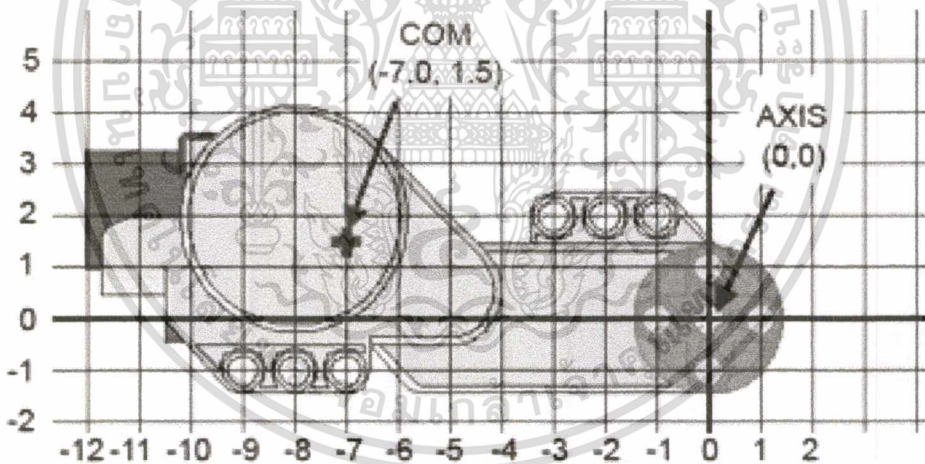


รูปที่ 2.12 เอนเอ็กซ์ทีมอเตอร์ (<http://www.g9toengineering.com>)

เอกสารนี้เป็นเอกสารที่สงวนไว้/[summerschool/Legomindstorms.htm](http://summerschool/Legomindstorms.htm)) ม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

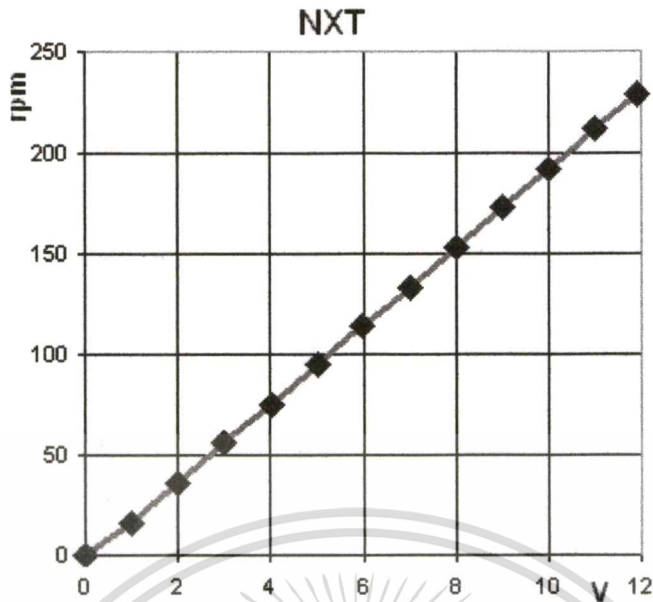


รูปที่ 2.13 แสดงส่วนประกอบภายในของเอนเอ็กซ์ทีมอเตอร์ (Daniele Benedettelli, 2008. <http://www.philohome.com/nxtmotor/nxtmotor.htm>)



รูปที่ 2.14 แสดงจุดศูนย์กลางมวลของเอนเอ็กซ์ทีมอเตอร์ (Kevin Knuth, 2008. <http://www.philohome.com/nxtmotor/nxtmotor.htm>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 2.15** กราฟแสดงความสัมพันธ์ระหว่างความเร็วในการหมุนของเอนเอ็กซ์ทีมอเตอร์และค่าแรงดันไฟฟ้า (<http://www.philohome.com/motors/motorcomp.htm>)

## 2.4 เลจอส และเลจอส เอนเอ็กซ์เจ

เลจอส(LeJOS) คือโครงการ โอเพนซอร์ส (Open source) ซึ่งพัฒนาจากโครงการ ทีนวีเอ็ม (TinyVM) เดิมซึ่งพัฒนาเกี่ยวกับจาวาวิซวลแมชชีน (Java Virtual Machine) สำหรับชุดทดลองระบบฝังตัวเลโก้ มาสเตอร์ม รูนอาร์ซีเอ็กซ์ (LEGO MINDSTORMS RCX) ซึ่งรู้จักกันดีและมีชื่อเสียงมากในชื่อ RIS (Robotics Invention System) ดังรูปที่ 2.16 ที่พัฒนาโดย Jose Solorzano



**รูปที่ 2.16** Lego MINDSTORMS Robotics Invention System (RIS) (Pavel Krcal.

2005. <http://www.it.uu.se/edu/course/homepage/realtid/h05/ass2>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลจอส เอนเอ็กซ์เจ (LeJOS NXJ) คือ Java programming environment สำหรับชุดทดลองระบบฝังตัวเลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที ซึ่งทำให้สามารถควบคุมการทำงานผ่านภาษาจาวาได้ โดยเลจอส เอนเอ็กซ์เจ ประกอบด้วย

1. เฟิร์มแวร์ (Firmware) ซึ่งมีการรวมเอาจาวาเวอร์ชวล (Virtual) แมชชีนที่ทำหน้าที่แทนเฟิร์มแวร์พื้นฐานเดิมของเลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที
2. ชุดคำสั่งที่จัดเตรียม Application Programming Interface (API) สำหรับ เลจอส เอนเอ็กซ์เจ
3. PC API หรือ API ฝั่งเครื่องควบคุม สำหรับการเขียนโปรแกรมติดต่อไปยัง เลจอส เอนเอ็กซ์เจฝั่งหุ่นยนต์โดยใช้ Java streams ผ่านทางช่องทางการสื่อสารไร้สายแบบบลูทูธ (Bluetooth) ซึ่งมีการนำเอาบลูโคฟ (Bluecove) ที่ทำการรวมเอา JSR-82 (Java Specification Requests-82) มาใช้งานร่วมด้วย

#### 2.4.1 JSR-82

JSR-82 เป็น Official Java Bluetooth API ซึ่งถูกกำหนดมาตรฐานโดย JSR-82 Expert Group JSR-82 จะประกอบไปด้วยแพ็คเกจ (package) จำนวน 2 แพ็คเกจ คือ

1. Javax.bluetooth (ประกอบไปด้วย 13 คลาสสำหรับการสื่อสารไร้สายด้วยบลูทูธ โปรโตคอล)
2. Javax.obex (ประกอบไปด้วย 8 คลาสสำหรับใช้ในการส่งผ่านอ็อบเจกต์ระหว่างอุปกรณ์)

โปรโตคอล OBEX ถูกใช้สำหรับส่งอ็อบเจกต์ไปมาระหว่างอุปกรณ์มาเป็นเวลานานแล้ว ด้วยเทคโนโลยีอินฟราเรด ซึ่งบลูทูธได้ทำการดัดแปลงโปรโตคอลนี้มาใช้ในการส่งอ็อบเจกต์เช่นกัน และเนื่องจาก JSR-82 เป็น Bluetooth API อย่างเป็นทางการของภาษาจาวา ดังนั้นผู้ผลิตทุกรายที่จะนำเอาไปใช้ จะต้องทำการรวมเลเยอร์ (layer) และ โปรไฟล์ (Profile) มาตรฐานตามที่กำหนด รวมไว้ใน SDK (Software development kit) ของตนด้วย

Bluetooth SDK ที่ใช้ JSR-82 จะต้องมี Bluetooth Stack layer มาตรฐานดังต่อไปนี้

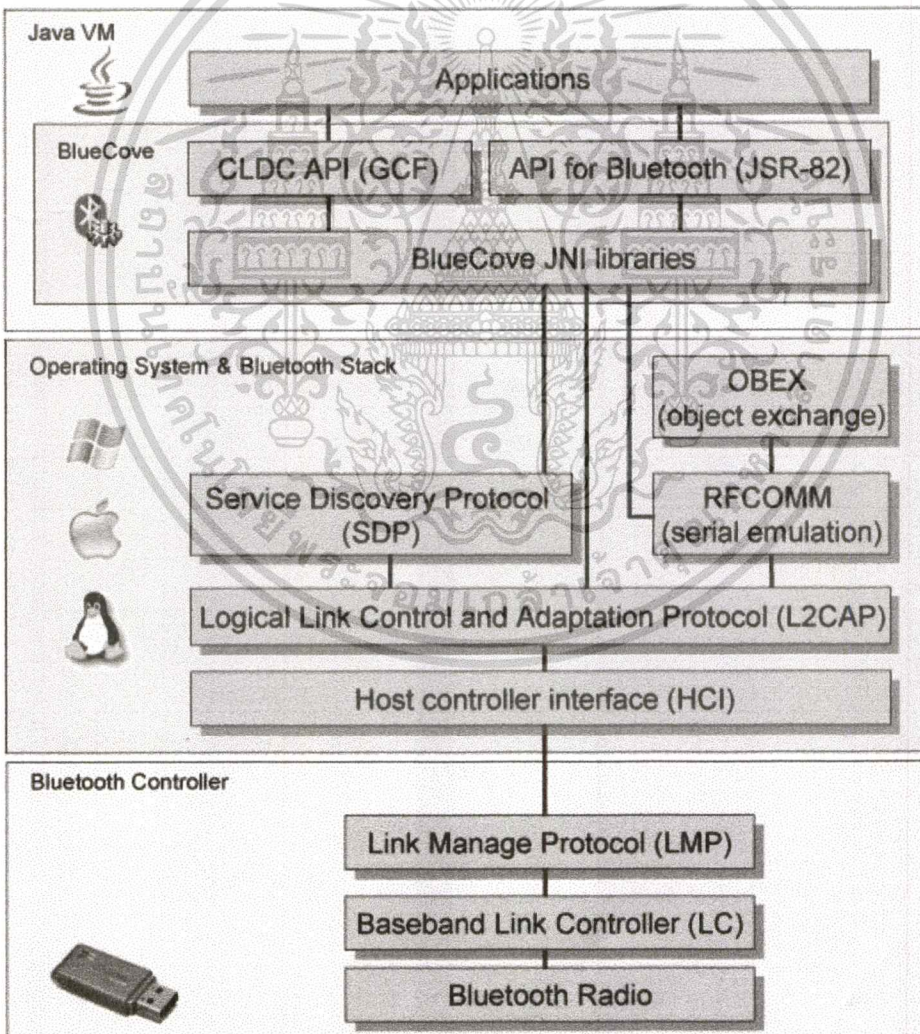
1. Host Controller Interface (HCI)
2. Logical Link Control and Adaptation Protocol (L2CAP)
3. Service Discovery Protocol (SDP)
4. RFCOMM

และจะต้องมีโปรไฟล์ดังต่อไปนี้

1. Generic Access Profile
2. Service Discovery Application Profile
3. Serial Port Profile
4. Generic Object Exchange Profile

## 2.4.2 บลูโคฟ

บลูโคฟ(Bluecove) คือจาวาไลบรารี (Java library) ที่นำเอา JSR-82 มาใช้งานบน J2SE (Java Standard Edition) ดังนั้นจึงทำให้จาวาแอปพลิเคชันที่ทำงานอยู่บนจาวาวิซวลแมชชีนที่นำเอาบลูโคฟมาใช้งานสามารถติดต่อกับอุปกรณ์อื่นๆที่มีบลูทูธได้ดังรูปที่ 2.17



รูปที่ 2.17 สแตคโคอะแกรม (Stack diagram) การทำงานของบลูโคฟ

(BlueCove Team. 2009. <http://snapshot.bluecove.org/>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.4.3 ข้อดีของเลออส เอนเอ็กซ์เจ

1. รองรับการเขียนโปรแกรมเชิงวัตถุ(Object-Oriented Programming)
2. สามารถทำงานร่วมกับ IDE(Integrated Development Environment) อย่างเช่น Eclipse หรือ Netbeans
4. การทำงานรวดเร็วกว่าเอนเอ็กซ์จี(NXT-G) ที่ใช้ในเฟรมเวิร์กพื้นฐานเดิมของเลโก้ มาสตรอม รุ่นเอนเอ็กซ์จี
3. รองรับการทำงานกับบลูทูธและระหว่าง NXT กับ NXT หรืออุปกรณ์อื่นๆที่รองรับ SPP(Serial Port Profile)
4. รองรับการทำงานแบบมัลติเทรด(Multithread)
5. รองรับการควบคุมมอเตอร์ที่มีความแม่นยำสูง



## บทที่ 3

### การวิเคราะห์และออกแบบระบบ

ในบทนี้จะกล่าวถึงขั้นตอนการวิเคราะห์และออกแบบทั้งส่วน โครงสร้างของหุ่นยนต์และระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสพภัย โดยเริ่มจากการเขียนรายละเอียดความต้องการของระบบแล้วทำการวิเคราะห์และออกแบบกระบวนการทำงานส่วนต่างๆของระบบ

#### 3.1 ความต้องการของระบบ

##### 3.1.1 ความต้องการที่เป็นหน้าที่หลัก (Functional Requirement)

- ระบบสามารถควบคุมทิศทางและความเร็วในการเคลื่อนที่ของหุ่นยนต์ได้
- หุ่นยนต์สามารถเคลื่อนที่ได้ในสภาพพื้นผิวที่ขรุขระและลาดชัน
- ระบบต้องสามารถควบคุมการหมุนของกล้องได้
- ระบบสามารถควบคุมหุ่นยนต์ได้จากระยะไกลผ่านการสื่อสารไร้สายแบบบลูทูธ (Bluetooth)
- ระบบสามารถตรวจสอบค่าพลังงานของแบตเตอรี่ได้
- ระบบสามารถตรวจสอบระดับสัญญาณของการสื่อสารไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์ได้
- ระบบสามารถบันทึกข้อมูลสถานะการทำงานต่างๆของระบบได้
- ระบบสามารถแสดงผลการทำงานด้วยเสียงเพื่อให้ผู้ควบคุมสามารถทราบถึงสถานะการทำงานของหุ่นยนต์

##### 3.1.2 ความต้องการที่ไม่ใช่หน้าที่หลัก (Non-Functional Requirement)

- ระบบควบคุมการเคลื่อนที่ของหุ่นยนต์สามารถใช้งานได้ง่าย
- ระบบสามารถทำงานร่วมกับระบบอื่นของหุ่นยนต์ได้

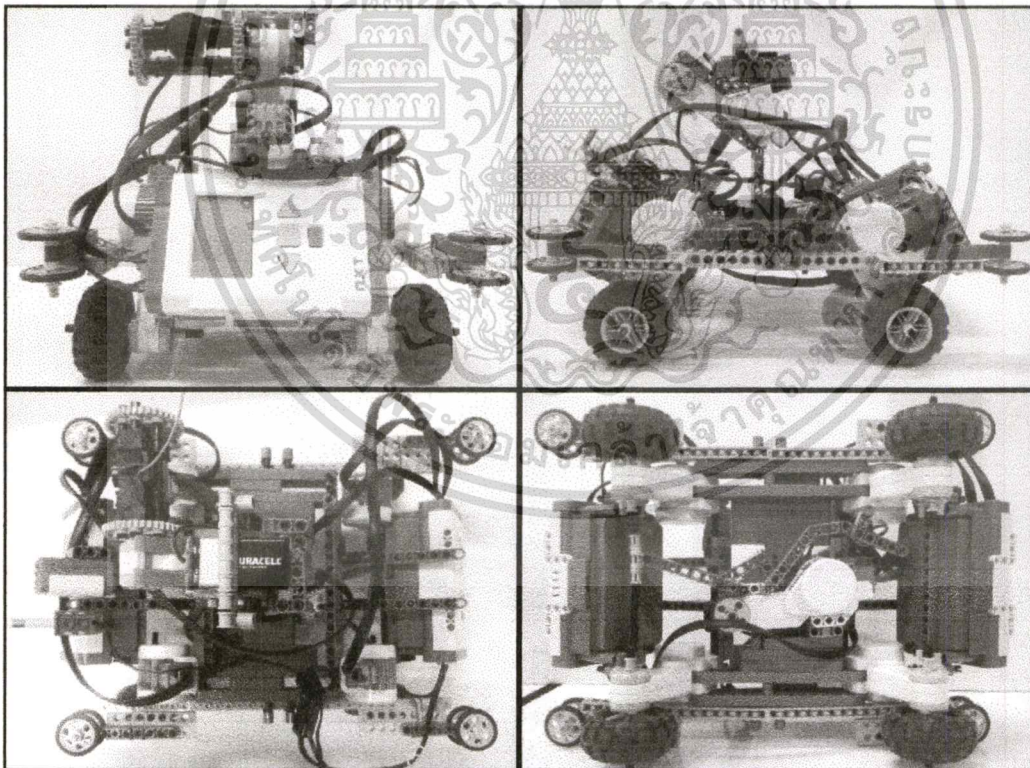
#### 3.2 การวิเคราะห์และออกแบบโครงสร้างหุ่นยนต์

เนื่องจากการทำงานของหุ่นยนต์ช่วยเหลือผู้ประสพภัยมีลักษณะการทำงานที่ต้องเคลื่อนที่เข้าไปยังพื้นที่ประสพภัยเพื่อค้นหาผู้รอดชีวิต ดังนั้นการควบคุมให้หุ่นยนต์สามารถเคลื่อนที่ได้อย่างมั่นคงและสามารถหมุนกล้องวีดีโอเพื่อให้เห็นสภาพแวดล้อมรอบข้าง ได้จึงเป็นสิ่งสำคัญ จากลักษณะการทำงานของหุ่นยนต์ดังกล่าวสามารถนำมาออกแบบ โครงสร้างหุ่นยนต์ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 แสดงการออกแบบส่วนการขับเคลื่อนหุ่นยนต์

ความต้องการของระบบ	การออกแบบหุ่นยนต์
<p>ระบบสามารถควบคุมทิศทางและ ความเร็วในการเคลื่อนที่ของหุ่นยนต์ได้</p> <p>หุ่นยนต์สามารถเคลื่อนที่ได้ในสภาพ พื้นผิวที่ขรุขระและลาดชัน</p>	<p>หุ่นยนต์ช่วยเหลือผู้ประสพภัยออกแบบให้ มีการขับเคลื่อนแบบ 4 ล้อ ทำให้มีแรง ขับเคลื่อนเพียงพอที่เคลื่อนที่ได้บนพื้นผิวที่ลาด ชัน โดยล้อคู่หน้าและล้อคู่หลังควบคุมโดยเอน เอ็กซ์ทีคอนโทรลเลอร์คนละตัว ส่วนตัวของ หุ่นยนต์มีการยกให้สูงขึ้นจากระดับพื้นเพื่อให้ สามารถเคลื่อนที่ได้ดีในสภาพพื้นผิวที่ขรุขระ</p> <p>ผังรูปที่ 3.1</p>

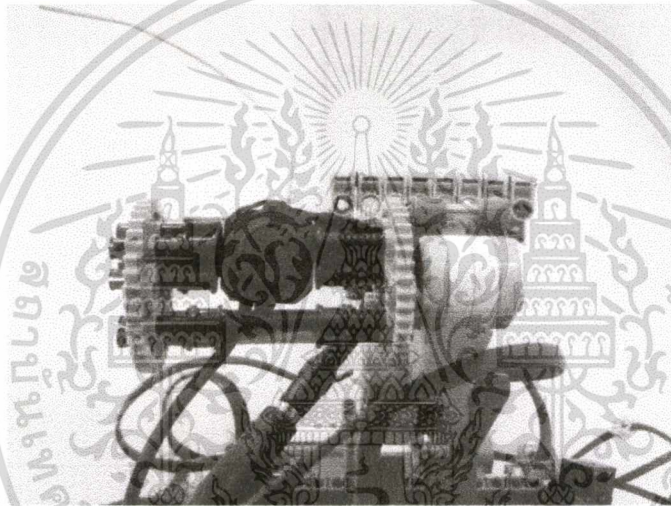


รูปที่ 3.1 ส่วนการขับเคลื่อนทั้ง 4 ล้อ

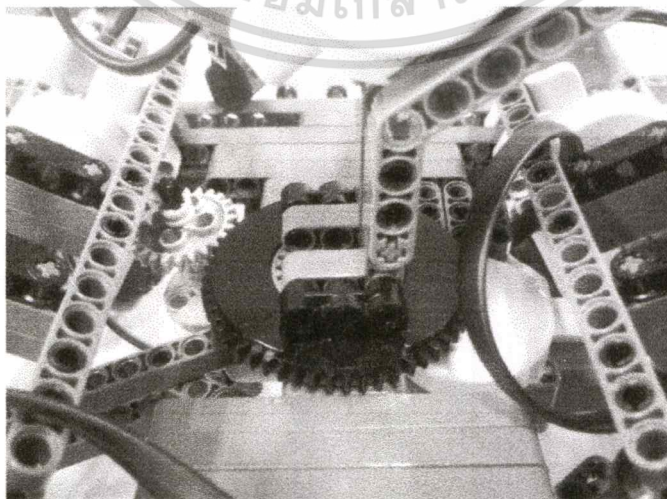
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 3.2 แสดงการออกแบบส่วนการหมุนกึ่งวีดีโอ

ความต้องการของระบบ	การออกแบบหุ่นยนต์
ระบบต้องสามารถควบคุมการหมุนของกึ่งได้อ	บริเวณส่วนกลางของหุ่นยนต์มีการติดตั้งแท่นหมุนกึ่งวีดีโอ ซึ่งสามารถหมุนได้ทั้งในแนวตั้งดังรูปที่ 3.2 และแนวระนาบดังรูปที่ 3.3 ทำให้สามารถเห็นสภาพแวดล้อมได้โดยรอบดังรูปที่ 3.4

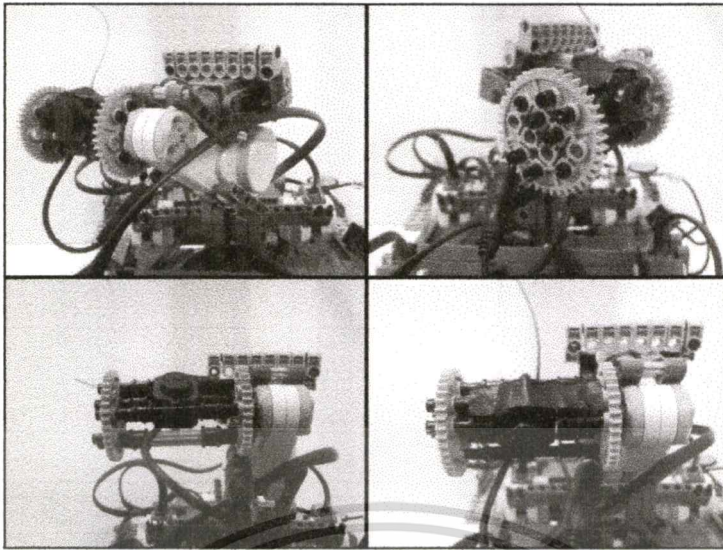


รูปที่ 3.2 ส่วนควบคุมการหมุนของกึ่งได้อในแนวตั้ง (บน-ล่าง)



รูปที่ 3.3 ส่วนควบคุมการหมุนของกึ่งได้อในแนวระนาบ (ซ้าย-ขวา)

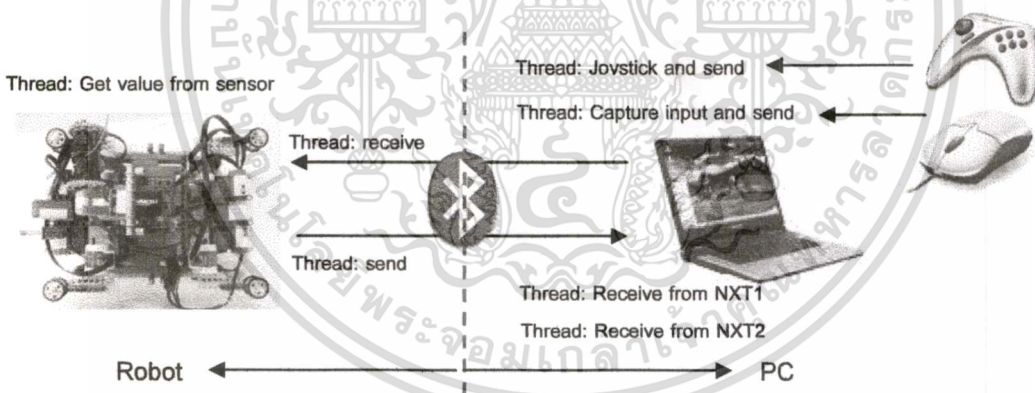
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 การควบคุมการหมุนของกลิ้งทั้งแนวระนาบและแนวตั้ง

### 3.3 การวิเคราะห์และออกแบบระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสพภัย

#### 3.3.1 ภาพรวมของระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสพภัย



รูปที่ 3.5 ภาพรวมของการควบคุมหุ่นยนต์ช่วยเหลือผู้ประสพภัย

การควบคุมหุ่นยนต์ช่วยเหลือผู้ประสพภัยสามารถทำการควบคุมได้เมื่อเครื่องควบคุมได้ทำการเชื่อมต่อสัญญาณบลูทูธกับหุ่นยนต์ ต่อจากนั้นจึงจะสามารถทำการควบคุมผ่าน GUI (Graphical User Interface) ที่เครื่องควบคุมหรืออาจจะควบคุมผ่านจอยสติ๊กที่เชื่อมต่อกับเครื่องควบคุม โดยเมื่อทำการป้อนคำสั่งให้กับระบบ ระบบจะทำการประมวลผลและทำการส่งรหัสคำสั่ง (Command code) ไปยังฝั่งหุ่นยนต์ โดยรหัสคำสั่งจะมีรูปแบบดังตารางที่ 3.3 และเมื่อหุ่นยนต์ได้รับรหัสคำสั่ง จะทำการตีความรหัสคำสั่งดังกล่าวว่าเป็นคำสั่งใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และทำงานตามคำสั่งนั้น เมื่อทำงานเสร็จเรียบร้อย ระบบที่ฝั่งหุ่นยนต์จะทำการตอบผลลัพธ์ของการทำงานว่าคำสั่งดังกล่าวได้ทำงานเสร็จสิ้นแล้ว ไปยังฝั่งเครื่องควบคุม ดังรูปที่ 3.5

เนื่องจากการทำงานทั้งฝั่งเครื่องควบคุมและฝั่งหุ่นยนต์มีการทำงานหลายๆอย่างพร้อมกัน จึงจำเป็นต้องมีการสร้างเธรด (Thread) เพื่อให้สามารถทำงานหลายๆอย่างได้ในเวลาเดียวกัน โดยฝั่งเครื่องควบคุมประกอบด้วยเธรดการทำงานหลักจำนวน 4 เธรดดังนี้

1. เธรดสำหรับการรับคำสั่งหรือผลลัพธ์จากเอนเอ็ชท์คอนโทรลเลอร์ตัวที่ 1 (receive from NXT1 thread)
2. เธรดสำหรับการรับคำสั่งหรือผลลัพธ์จากเอนเอ็ชท์คอนโทรลเลอร์ตัวที่ 2 (receive from NXT2 thread)
3. เธรดสำหรับการส่งคำสั่งจากการกดปุ่มคำสั่ง (Capture input and send thread)
4. เธรดสำหรับการส่งคำสั่งจากการกดจอยสติ๊ก (Joystick and send thread)

ส่วนฝั่งหุ่นยนต์ เอนเอ็ชท์คอนโทรลเลอร์แต่ละตัวประกอบด้วยเธรดการทำงานหลักจำนวน 3 เธรดดังนี้

1. เธรดสำหรับการรับคำสั่งจากเครื่องควบคุม (Receive thread)
2. เธรดสำหรับการส่งคำสั่งหรือผลลัพธ์ไปยังเครื่องควบคุม (Send thread)
3. เธรดสำหรับการดึงข้อมูลจากเซนเซอร์ (Get value from sensor thread)

ตารางที่ 3.3 แสดงรูปแบบคำสั่งที่ส่งระหว่างเครื่องควบคุมและหุ่นยนต์

ส่งจากเครื่องควบคุมไปยังหุ่นยนต์		ส่งจากหุ่นยนต์ไปยังเครื่องควบคุม	
รหัส	ความหมาย	รหัส	ความหมาย
1000	ตัดการเชื่อมต่อ		
1001	ตรวจสอบระดับสัญญาณบลูทูธ	1001xxxx	ระดับสัญญาณ = xxxx
1002	ตรวจสอบระดับแรงดันไฟฟ้าของแบตเตอรี่	1002xxxx	ระดับแรงดัน = xxxx
1003	เดินหน้า	10031111	กระบวนการเสร็จสมบูรณ์
1004	ถอยหลัง	10041111	กระบวนการเสร็จสมบูรณ์
1005	หมุนซ้าย	10051111	กระบวนการเสร็จสมบูรณ์
1006	หมุนขวา	10061111	กระบวนการเสร็จสมบูรณ์
1007	หยุดเคลื่อนที่	10071111	กระบวนการเสร็จสมบูรณ์
1008	ปรับการหมุนซ้าย	10081111	กระบวนการเสร็จสมบูรณ์
1009	ปรับการหมุนขวา	10091111	กระบวนการเสร็จสมบูรณ์
1010	เล่นเสียงเพลง	10101111	กระบวนการเสร็จสมบูรณ์
1011	สลับ ไปยัง unsafe mode	10111111	กระบวนการเสร็จสมบูรณ์
1012	สลับ ไปยัง Safe mode	10121111	กระบวนการเสร็จสมบูรณ์
		10131111	แจ้งเตือนระดับสัญญาณ
1014	เตือนเอนเอ็กซ์ที่คอนโทรลเลอร์ที่ยังไม่หยุดการเคลื่อนที่		
1015	หมุนกลิ้งในแนวตั้ง(-)	1015xxxx	ทิศทางองศาปัจจุบัน = xxxx
1016	หมุนกลิ้งในแนวตั้ง(+)	1016xxxx	ทิศทางองศาปัจจุบัน = xxxx
1017	หมุนกลิ้งในแนวระนาบ(-)	1017xxxx	ทิศทางองศาปัจจุบัน = xxxx
1018	หมุนกลิ้งในแนวระนาบ(+)	1018xxxx	ทิศทางองศาปัจจุบัน = xxxx
1019		10191111	กระบวนการเสร็จสมบูรณ์
1020	หมุนกลิ้งในแนวตั้งกลับไปยังจุดอ้างอิง	10201111	กระบวนการเสร็จสมบูรณ์
1021	หมุนกลิ้งในแนวระนาบกลับไปยังจุดอ้างอิง	10211111	กระบวนการเสร็จสมบูรณ์
1022	หยุดการเคลื่อนที่ก่อนจะหมุนกลิ้ง	10221111	กระบวนการเสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3(ต่อ)

ส่งจากเครื่องควบคุมไปยังหุ่นยนต์		ส่งจากหุ่นยนต์ไปยังเครื่องควบคุม	
รหัส	ความหมาย	รหัส	ความหมาย
2000	ตรวจสอบความดังของเสียง	2000xxxx	ระดับเสียง = xxxx
2001	ตรวจสอบอุณหภูมิ	2001xxxx	ระดับอุณหภูมิ = xxxx
5555	กำหนดช่วงการส่งข้อมูล	55551111	กระบวนการเสร็จสมบูรณ์
7001	ปรับความเร็วเป็นระดับช้า	70011111	กระบวนการเสร็จสมบูรณ์
7002	ปรับความเร็วเป็นระดับปานกลาง	70021111	กระบวนการเสร็จสมบูรณ์
7003	ปรับความเร็วเป็นระดับเร็ว	70031111	กระบวนการเสร็จสมบูรณ์
6666	คำสั่งล้มเหลว	66661111	กระบวนการล้มเหลว
8888	กำหนดให้เอนเอ็กซ์ทีคอนโทรลเลอร์เป็นตัวที่ 1	88881111	กระบวนการเสร็จสมบูรณ์
9999	กำหนดให้เอนเอ็กซ์ทีคอนโทรลเลอร์เป็นตัวที่ 2	99991111	กระบวนการเสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

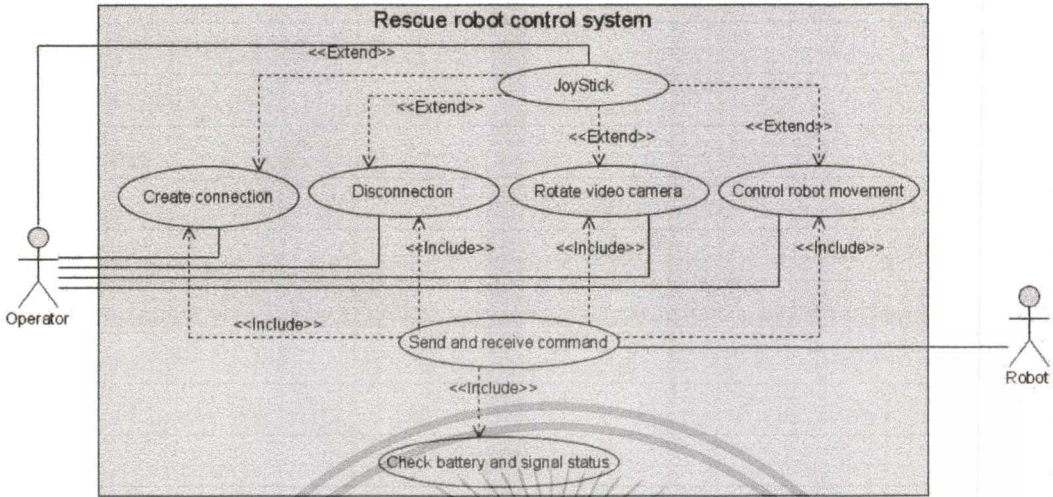
### 3.3.2 การวิเคราะห์ระบบ

ตารางที่ 3.4 การวิเคราะห์ความต้องการ และฟังก์ชันการทำงานของระบบ

ความต้องการของระบบ	ยูสเคสที่รองรับการทำงาน	แอกเตอร์
1. ระบบสามารถควบคุมทิศทางและความเร็วในการเคลื่อนที่ของหุ่นยนต์ได้	Control robot movement	Operator
2. ระบบต้องสามารถควบคุมการหมุนของกล้องได้	Rotate video camera	Operator
3. ระบบสามารถควบคุมหุ่นยนต์ได้จากระยะไกลผ่านการสื่อสารไร้สายแบบบลูทูธ	Create connection Disconnection Send and receive command	Operator Robot
4. ระบบสามารถตรวจสอบค่าแรงดันไฟฟ้าของแบตเตอรี่ได้	Check battery and signal status	
5. ระบบสามารถตรวจสอบระดับสัญญาณของการสื่อสารไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์ได้	Check battery and signal status	
6. ระบบควบคุมการเคลื่อนที่ของหุ่นยนต์สามารถใช้งานได้ง่าย	JoyStick	Operator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.3 ยูสเคสไดอะแกรม (Use Case diagram)



รูปที่ 3.6 แสดงยูสเคสไดอะแกรมของระบบควบคุมการเคลื่อนที่

ระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสบภัย (Rescue robot control system) ดังรูปที่ 3.6 ประกอบด้วยยูสเคสดังนี้

ยูสเคสการสร้างการเชื่อมต่อ (Create connection) ทำหน้าที่ในการสร้างการเชื่อมต่อไร้สายแบบลูทระหว่างเครื่องควบคุมและหุ่นยนต์ พร้อมกับกำหนดหน้าที่การทำงานให้กับเอนเอ็กซ์ทีคอนโทรลเลอร์ทั้งสองตัว

ยูสเคสการยกเลิกการเชื่อมต่อ (Disconnection) ทำหน้าที่ในการยกเลิกการเชื่อมต่อไร้สายแบบลูทระหว่างเครื่องควบคุมและหุ่นยนต์

ยูสเคสการส่งและรับคำสั่ง (Send and receive command) ทำหน้าที่ในส่งและรับคำสั่งระหว่างฝั่งผู้ควบคุม (Operator) และฝั่งหุ่นยนต์ (Robot) พร้อมทั้งตรวจสอบรหัสคำสั่งว่ารหัสคำสั่งดังกล่าวสั่งให้ระบบทำงานอะไรก็จะส่งงานไปยังยูสเคสการทำงานอื่นที่เกี่ยวข้องกับคำสั่งนั้นๆ

ยูสเคสการควบคุมการเคลื่อนที่ของหุ่นยนต์ (Control robot movement) ทำหน้าที่ในการขับเคลื่อนหุ่นยนต์ซึ่งอาศัยการควบคุมการหมุนของเอนเอ็กซ์ทีมอเตอร์

ยูสเคสการหมุนกล้องวิดีโอ (Rotate video camera) ทำหน้าที่ในการหมุนกล้องวิดีโอซึ่งอาศัยการควบคุมการหมุนของเอนเอ็กซ์ทีมอเตอร์ที่ต่ออยู่กับแกนของกล้องวิดีโอ ทำให้สามารถมองเห็นสภาพแวดล้อมได้โดยรอบ

ยูสเคสการตรวจสอบพลังงานและสัญญาณ (Check battery and signal status) ทำหน้าที่ในการตรวจสอบระดับพลังงานของหุ่นยนต์ และตรวจสอบระดับสัญญาณของการสื่อสารไร้สายแบบลูทระหว่างเครื่องควบคุมและหุ่นยนต์

ยูสเคสควบคุมผ่านจอยสติค (JoyStick) ทำหน้าที่ในการควบคุมการทำงานของหุ่นยนต์ผ่าน

จอยสติค เอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.4 คำอธิบายยูสเคส(Use Case Description)

ตารางที่ 3.5 คำอธิบายยูสเคสการสร้างการเชื่อมต่อ

Use Case Name: สร้างการเชื่อมต่อ	ID: 1	Importance Level: High
Primary Actor: ผู้ควบคุม	Use Case Type: Function	
Brief Description: สร้างการเชื่อมต่อไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์		
Pre-Condition: หุ่นยนต์ได้ทำการรัน โปรแกรมเรียบร้อยแล้ว		
Post-Condition: การเชื่อมต่อเสร็จสมบูรณ์		
Trigger: ผู้ควบคุมกดปุ่มสร้างการเชื่อมต่อ		
Relationships: Association: ผู้ควบคุม		
Normal Flow of Events: <ol style="list-style-type: none"> <li>1. ฟังก์ชันเครื่องควบคุมทำการเชื่อมต่อไปยังฟังก์ชันหุ่นยนต์</li> <li>2. ฟังก์ชันหุ่นยนต์ทำการตอบรับการเชื่อมต่อ</li> <li>3. ฟังก์ชันเครื่องควบคุมแสดงสถานะการเชื่อมต่อเสร็จสมบูรณ์พร้อมแสดงผลต่อด้วยเสียง</li> </ol>		
Exceptional Conditions: หากตรวจพบมีการสร้างการเชื่อมต่ออยู่แล้วจะทำการแจ้งเตือนผู้ควบคุม		

ตารางที่ 3.6 คำอธิบายยูสเคสการยกเลิกการเชื่อมต่อ

Use Case Name: ยกเลิกการเชื่อมต่อ	ID: 2	Importance Level: High
Primary Actor: ผู้ควบคุม	Use Case Type: Function	
Brief Description: ยกเลิกการเชื่อมต่อไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์		
Pre-Condition: การเชื่อมต่อไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์ได้ถูกสร้างไว้แล้ว		
Post-Condition: การเชื่อมต่อถูกยกเลิก		
Trigger: ผู้ควบคุมกดปุ่มยกเลิกการเชื่อมต่อ		
Relationships: Association: ผู้ควบคุม		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 3.6 (ต่อ)

<p>Normal Flow of Events:</p> <ol style="list-style-type: none"> <li>1. ฟังก์ชันควบคุมทำการส่งคำสั่งยกเลิกการเชื่อมต่อ ไปยังฟังก์ชัน</li> <li>2. ฟังก์ชันทำการยกเลิกการเชื่อมต่อ</li> <li>3. ฟังก์ชันควบคุมทำการตรวจจบการเชื่อมต่อ เมื่อการเชื่อมต่อถูกยกเลิก ทำการแสดงสถานะยกเลิกการเชื่อมต่อพร้อมแสดงผลด้วยเสียง</li> </ol>
<p>Exceptional Conditions: หากตรวจพบว่าการสร้างการเชื่อมต่อไม่ได้ถูกสร้างไว้ก่อนจะทำการแจ้งเตือนผู้ควบคุม</p>

### ตารางที่ 3.7 คำอธิบายยูสเคสการส่งและรับคำสั่ง

Use Case Name: ส่งและรับคำสั่ง	ID: 3	Importance Level: High
Primary Actor: หุ่นยนต์	Use Case Type: Function	
Brief Description: ส่งและรับคำสั่งระหว่างฟังก์ชันควบคุมและฟังก์ชัน		
Pre-Condition: การเชื่อมต่อไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์ได้ถูกสร้างไว้แล้ว		
Post-Condition: คำสั่งถูกส่งจากต้นทาง ไปยังปลายทาง		
Trigger: ต้นทางต้องการส่งข้อมูล ไปปลายทาง		
Relationships:		
Association: หุ่นยนต์		
<p>Normal Flow of Events:</p> <p><b>กรณีต้นทางคือเครื่องควบคุม</b></p> <ol style="list-style-type: none"> <li>1. ฟังก์ชันควบคุมทำการรับคำสั่งจากผู้ควบคุมแล้วทำการแปลงข้อมูลให้อยู่ในรูปแบบที่กำหนด</li> <li>2. ฟังก์ชันควบคุมทำการส่งข้อมูลไปยังฟังก์ชัน</li> <li>3. ฟังก์ชันทำการรับข้อมูลแล้วทำการแปลงข้อมูลให้อยู่ในรูปแบบคำสั่งที่กำหนดไว้</li> </ol> <p><b>กรณีต้นทางคือหุ่นยนต์</b></p> <ol style="list-style-type: none"> <li>1. ฟังก์ชันทำการทำการแปลงข้อมูลให้อยู่ในรูปแบบที่กำหนดไว้แล้วส่งข้อมูลไปยังฟังก์ชันควบคุม</li> <li>2. ฟังก์ชันทำการรับข้อมูลแล้วทำการแปลงข้อมูลให้อยู่ในรูปแบบคำสั่งที่กำหนด</li> </ol>		
Exceptional Conditions: หากระว่างการส่งข้อมูลเกิดข้อผิดพลาด การเชื่อมต่อจะถูกยกเลิก		

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 3.8 คำอธิบายยูสเคสควบคุมการเคลื่อนที่ของหุ่นยนต์

Use Case Name: ควบคุมการเคลื่อนที่ของหุ่นยนต์	ID: 4	Importance Level: High
Primary Actor: ผู้ควบคุม	Use Case Type: Function	
Brief Description: ขับเคลื่อนหุ่นยนต์ซึ่งอาศัยการควบคุมการหมุนของเอนเอ็กซ์ทีมอเตอร์		
Pre-Condition: การเชื่อมต่อไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์ได้ถูกสร้างไว้แล้ว		
Post-Condition: หุ่นยนต์เคลื่อนที่ตรงตามคำสั่ง		
Trigger: ผู้ควบคุมกดปุ่มควบคุมการเคลื่อนที่		
Relationships: Association: ผู้ควบคุม		
Normal Flow of Events: <ol style="list-style-type: none"> <li>1. ฟังก์ชันควบคุมทำการรับคำสั่งจากผู้ควบคุม</li> <li>2. ฟังก์ชันควบคุมทำการส่งข้อมูลไปยังฟังก์ชันหุ่นยนต์</li> <li>3. ฟังก์ชันหุ่นยนต์ทำการรับข้อมูลแล้วทำการเคลื่อนที่ตามคำสั่งดังกล่าว</li> </ol>		
Exceptional Conditions:		

### ตารางที่ 3.9 คำอธิบายยูสเคสควบคุมการหมุนกล้องวิดีโอ

Use Case Name: ควบคุมการหมุนกล้องวิดีโอ	ID: 5	Importance Level: High
Primary Actor: ผู้ควบคุม	Use Case Type: Function	
Brief Description: หมุนกล้องวิดีโอ		
Pre-Condition: การเชื่อมต่อไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์ได้ถูกสร้างไว้แล้ว		
Post-Condition: หมุนกล้องวิดีโอตรงตามทิศทาง		
Trigger: ผู้ควบคุมกดปุ่มควบคุมการหมุนกล้องวิดีโอ		
Relationships: Association: ผู้ควบคุม		
Normal Flow of Events: <ol style="list-style-type: none"> <li>1. ฟังก์ชันควบคุมทำการรับคำสั่งจากผู้ควบคุม</li> <li>2. ฟังก์ชันควบคุมทำการส่งข้อมูลไปยังฟังก์ชันหุ่นยนต์</li> <li>3. ฟังก์ชันหุ่นยนต์ทำการรับข้อมูลแล้วทำการเคลื่อนที่ตามคำสั่งดังกล่าว</li> </ol>		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 3.9 (ต่อ)

Exceptional Conditions: กล้องวีดีโอสามารถหมุนในแนวระนาบได้ในช่วง -180 ถึง 180 องศา และ หมุนในแนวตั้งได้ในช่วง 0 ถึง 180 องศา หากผู้ควบคุมทำการหมุน นอกเหนือจากองศาดังกล่าว ระบบจะไม่ประมวลผลคำสั่ง

### ตารางที่ 3.10 คำอธิบายยูสเคสตรวจสอบพลังงานและสัญญาณบลูทูธ

Use Case Name: ตรวจสอบพลังงานและสัญญาณบลูทูธ	ID:6	Importance Level: High
Primary Actor:	Use Case Type: Function	
Brief Description: ตรวจสอบระดับพลังงานของหุ่นยนต์ และตรวจสอบระดับสัญญาณของการสื่อสาร ไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์		
Pre-Condition: การเชื่อมต่อ ไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์ได้ถูกสร้างไว้แล้ว		
Post-Condition: ระบบแสดงระดับพลังงานและสัญญาณบลูทูธ		
Trigger: การเชื่อมต่อถูกสร้างขึ้น		
Relationships: Association: ผู้ควบคุม		
Normal Flow of Events: <ol style="list-style-type: none"> <li>1. ในทุกๆ 0.5 วินาที ฟังก์ชันจะทำการตรวจสอบระดับพลังงานและระดับสัญญาณบลูทูธ</li> <li>2. ฟังก์ชันทำการส่งข้อมูลระดับพลังงานและระดับสัญญาณบลูทูธ ไปยังฟังก์ชันเครื่องควบคุม</li> <li>3. ฟังก์ชันเครื่องควบคุมทำการแสดงผลระดับพลังงานและระดับสัญญาณบลูทูธ</li> </ol>		
Exceptional Conditions: หากระดับสัญญาณที่ฟังก์ชันมีค่าต่ำกว่า 203 ระบบจะเริ่มกลไกป้องกัน หุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุม		

### ตารางที่ 3.11 คำอธิบายยูสเคสควบคุมผ่านจอยสติ๊ก

Use Case Name: ควบคุมผ่านจอยสติ๊ก	ID:7	Importance Level: High
Primary Actor: ผู้ควบคุม	Use Case Type: Function	
Brief Description: ควบคุมการทำงานของหุ่นยนต์ผ่านจอยสติ๊ก		
Pre-Condition: การเชื่อมต่อ ไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์ได้ถูกสร้างไว้แล้ว		
Post-Condition: หุ่นยนต์ทำงานตรงตามคำสั่งของจอยสติ๊ก		

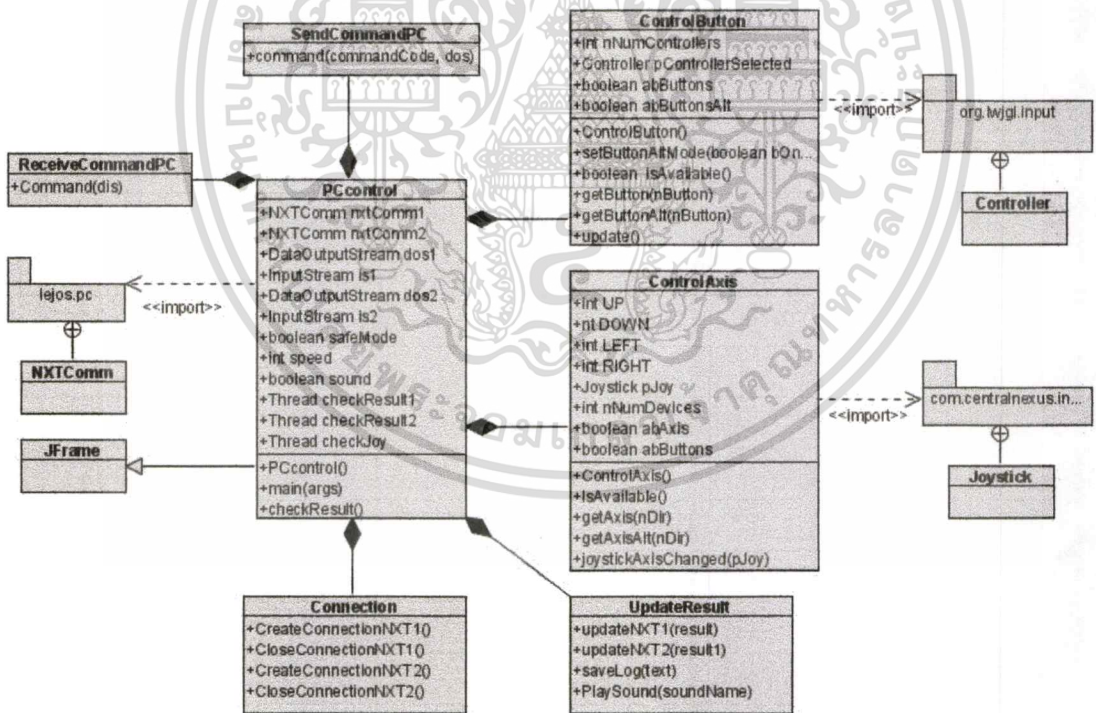
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 3.11 (ต่อ)

Trigger: ผู้ควบคุมกดปุ่มควบคุมการทำงานของจอยสติ๊ก
Relationships: Association: ผู้ควบคุม
Normal Flow of Events: <ol style="list-style-type: none"> <li>1. ฟังก์ชันควบคุมทำการรับคำสั่งจากผู้ควบคุมผ่านจอยสติ๊ก</li> <li>2. ฟังก์ชันควบคุมทำการส่งข้อมูลไปยังฟังก์ชันอื่น</li> <li>3. ฟังก์ชันอื่นทำการรับข้อมูลแล้วทำการเคลื่อนที่ตามคำสั่งดังกล่าว</li> </ol>
Exceptional Conditions:

### 3.3.5 คลาสไดอะแกรม (Class Diagram)

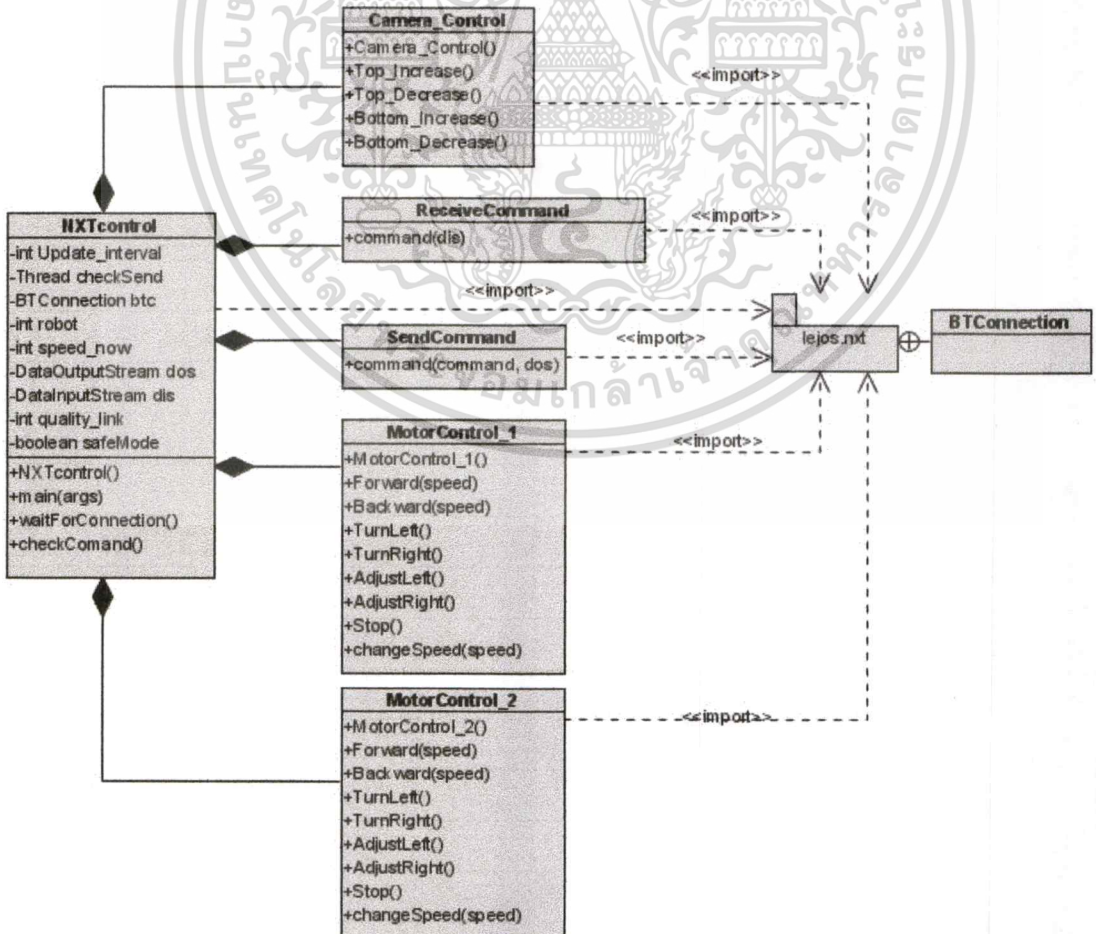


รูปที่ 3.7 แสดงคลาสไดอะแกรมของระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสภภัยฟังก์ชันควบคุม

จากรูปที่ 3.7 คือ คลาสในส่วนของเครื่องควบคุมที่สามารถแบ่งออกเป็นคลาสหลักได้ทั้งหมด 7 คลาสดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คลาสควบคุมเครื่องควบคุม (PC control) เป็นคลาสหลักที่ใช้ในการควบคุมการทำงานทั้งหมดของระบบฝั่งเครื่องควบคุม
- คลาสการเชื่อมต่อ (Connection) เป็นคลาสที่ใช้ทำการเชื่อมต่อ ยกเลิก การสื่อสารไร้สายแบบบลูทูธระหว่างเครื่องควบคุมและหุ่นยนต์
- คลาสส่งคำสั่ง (Send command PC) เป็นคลาสที่ใช้ส่งรหัสคำสั่งไปยังฝั่งหุ่นยนต์ในรูปแบบของรหัสคำสั่ง 4 หลัก
- คลาสรับคำสั่ง (Receive command PC) เป็นคลาสที่ใช้ในการรับผลลัพธ์ของคำสั่งที่ส่งมาจากฝั่งหุ่นยนต์ในรูปแบบของรหัสคำสั่ง 4 หลักตามด้วยผลลัพธ์
- คลาสแสดงผลให้เป็นปัจจุบัน (Update result) เป็นคลาสที่ใช้ในการวิเคราะห์ผลลัพธ์ที่ได้จากฝั่งหุ่นยนต์และแสดงผล
- คลาสควบคุมปุ่ม (Control button) และคลาสควบคุมแกน(Control axis) เป็นคลาสที่ใช้ในการควบคุมการทำงานของจอยสติค



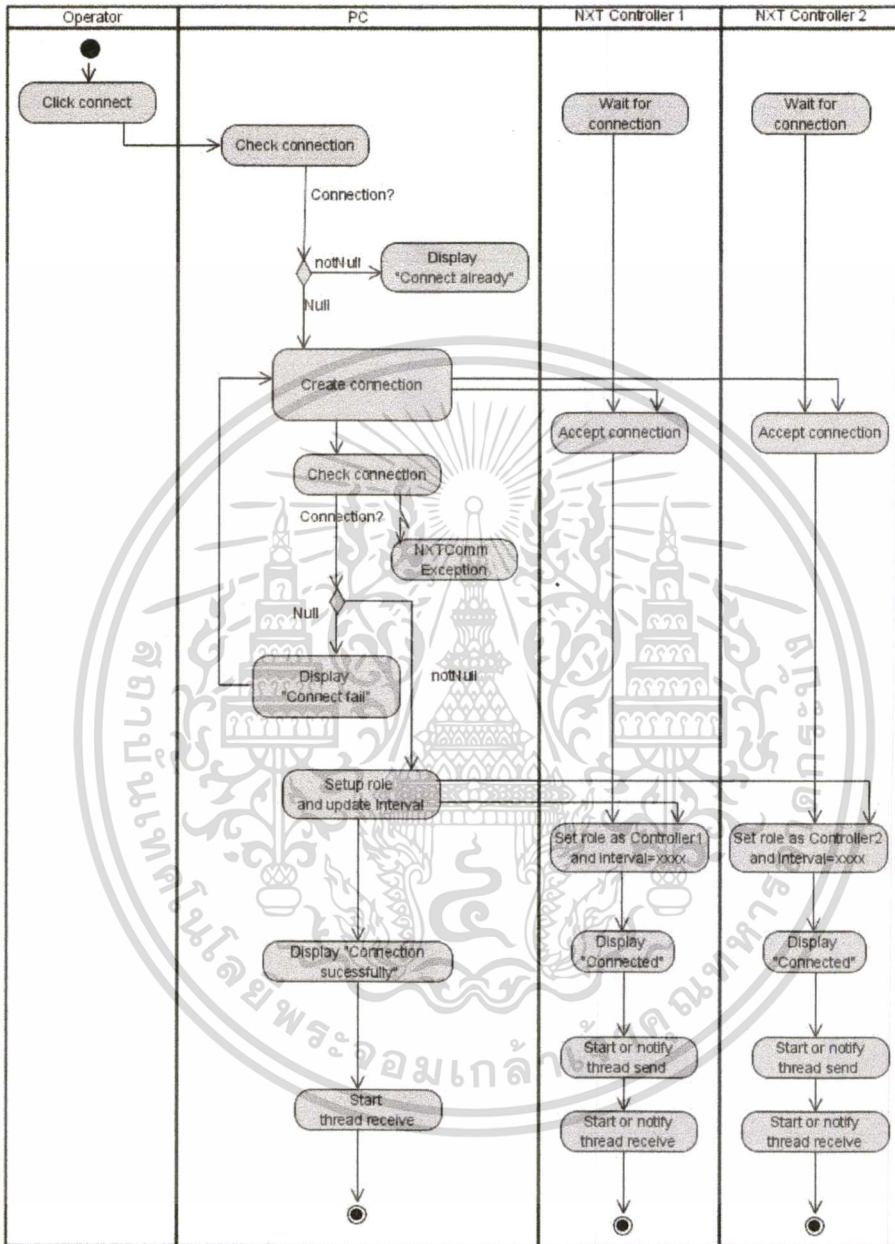
รูปที่ 3.8 แสดงคลาสโค้ดระบบของระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสาภัยฝั่งหุ่นยนต์ด้านการค้า  
 ไม่ว่าการณ์ใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.8 คือ คลาสในส่วนของหุ่นยนต์ที่สามารถแบ่งออกเป็นคลาสหลักได้ทั้งหมด 6 คลาสดังนี้

- คลาสควบคุมหุ่นยนต์ (NXT control) เป็นคลาสหลักที่ใช้ในการควบคุมการทำงานทั้งหมดของระบบฝั่งหุ่นยนต์รวมถึงการรอการเชื่อมต่อจากฝั่งเครื่องควบคุม
- คลาสรับคำสั่ง (Receive command) เป็นคลาสที่ใช้ในการรับรหัสคำสั่งจากฝั่งเครื่องควบคุมในรูปแบบของรหัสคำสั่ง 4 หลัก
- คลาสส่งคำสั่ง (Send command ) เป็นคลาสที่ใช้ส่งผลลัพธ์ของคำสั่งที่ได้รับจากฝั่งเครื่องควบคุมรูปแบบของรหัสคำสั่ง 4 หลักตามด้วยผลลัพธ์
- คลาสควบคุมมอเตอร์ 1 (Motor control 1) และคลาสควบคุมมอเตอร์ 2 (Motor control 2) เป็นคลาสที่ใช้ในการควบคุมการทำงานมอเตอร์คู่หน้า และคู่หลัง
- คลาสควบคุมกล้องวิดีโอ (Camera control) เป็นคลาสหลักที่ใช้ในการควบคุมการหมุนของกล้องวิดีโอโดยควบคุมการหมุนของมอเตอร์ 2 ตัวคือมอเตอร์ด้านบนที่หมุนในแนวตั้ง และมอเตอร์ด้านล่างที่หมุนในแนวระนาบ

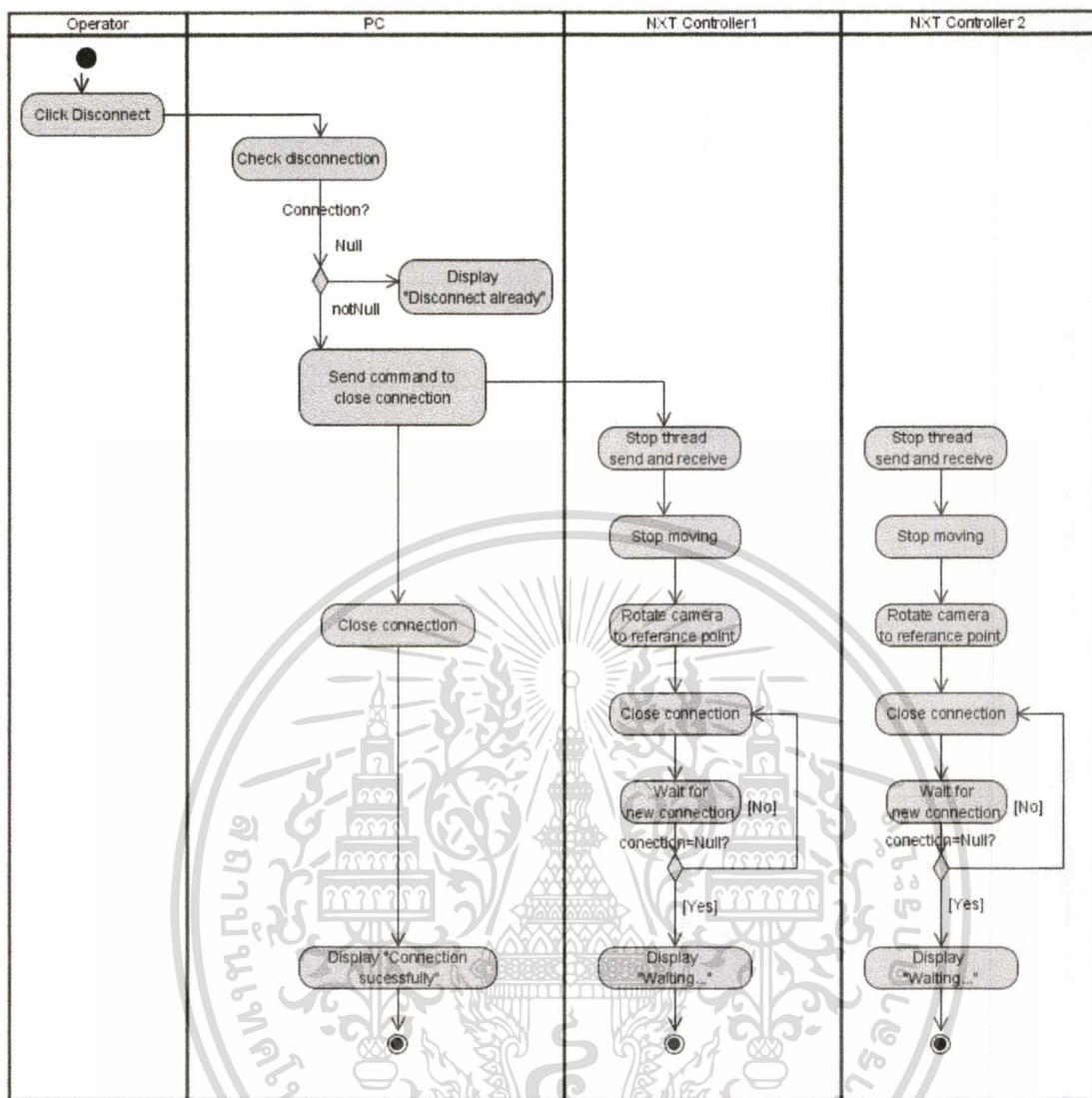


### 3.3.6 ไลอะแกรมส่วนสร้างและยกเลิกการเชื่อมต่อ

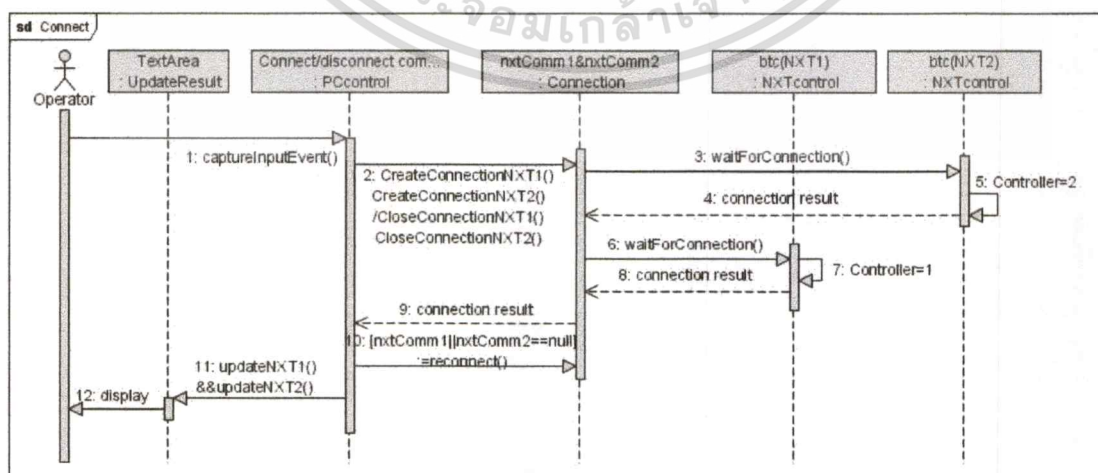


รูปที่ 3.9 แสดงแอกทิวิตี้ไลอะแกรมของการสร้างการเชื่อมต่อไร้สายแบบบลูทูธ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 แสดงเอกทวิทัศน์ไดอะแกรมของการยกเลิกการเชื่อมต่อไร้สายแบบบลูทูธ



รูปที่ 3.11 แสดงซีควเอนไดอะแกรมของการสร้างและยกเลิกการเชื่อมต่อไร้สายแบบบลูทูธ

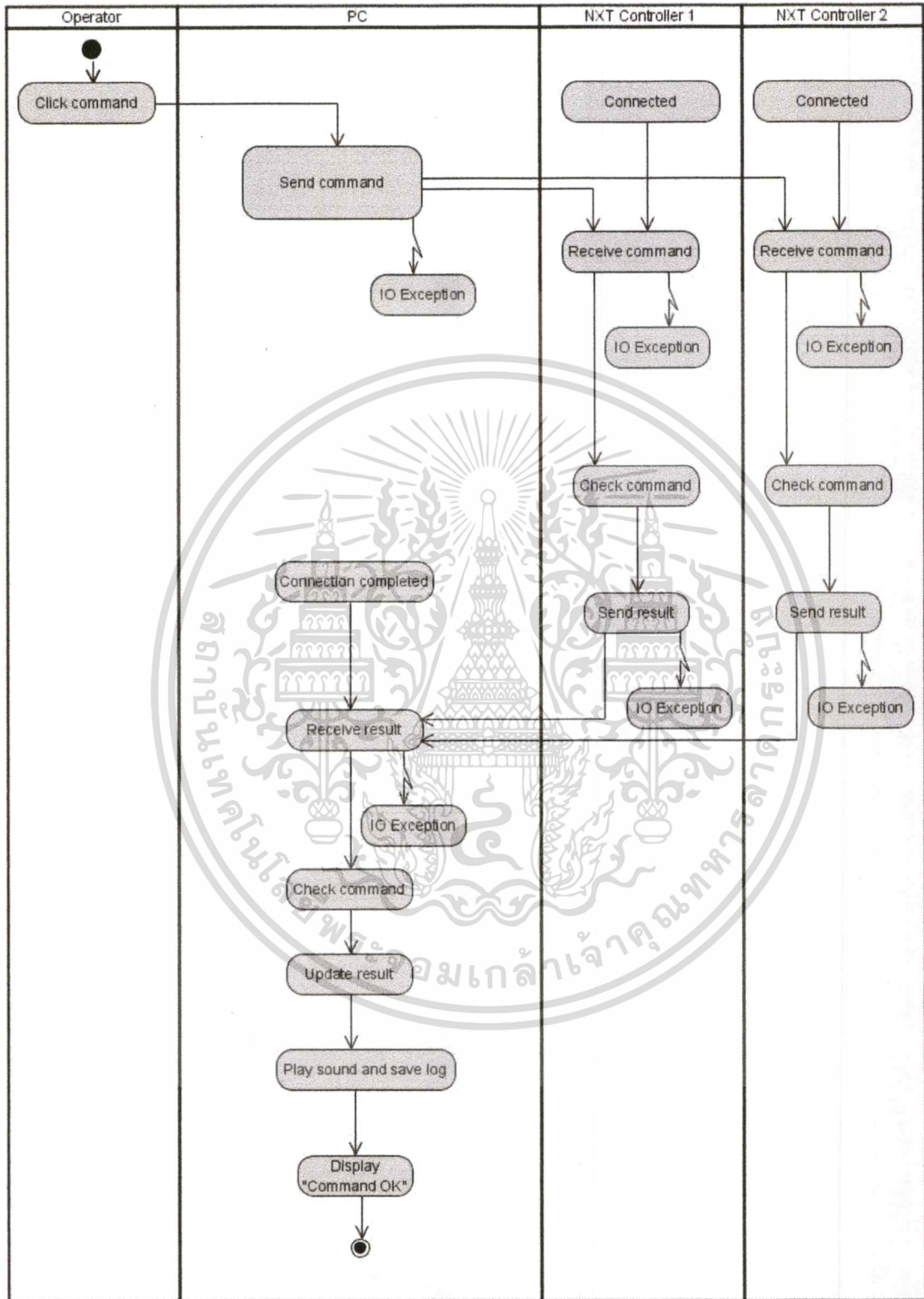
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.9 และ 3.11 เมื่อผู้ควบคุมหุ่นยนต์ทำการส่งการเชื่อมต่อไปยังเครื่องควบคุม เครื่องควบคุมจะทำการส่งคำสั่งในการขอเปิดการเชื่อมต่อไปยังหุ่นยนต์ ในขณะที่เอนเอ็กซ์ทีคอนโทรลเลอร์ทั้งสองเองก็อยู่ในสถานะรอการเชื่อมต่อ เมื่อได้รับการเชื่อมต่อจากฝั่งเครื่องควบคุมก็จะทำการสร้างช่องทางการเชื่อมต่อ หากการเชื่อมต่อเกิดปัญหาก็จะทำการเชื่อมต่อซ้ำ

เมื่อทำการเชื่อมต่อได้แล้วฝั่งเครื่องควบคุมจะทำการส่งคำสั่งในการกำหนดหน้าที่และช่วงเวลา(ช่วงเวลาในการส่งค่าพลังงานและระดับสัญญาณ) ให้กับเอนเอ็กซ์ทีคอนโทรลเลอร์แต่ละตัว จากนั้นระบบฝั่งหุ่นยนต์จะทำการเริ่มการทำงานของเทรคส่งและรับข้อมูล พร้อมกับฝั่งเครื่องควบคุมจะทำการเริ่มการทำงานของเทรครับข้อมูล

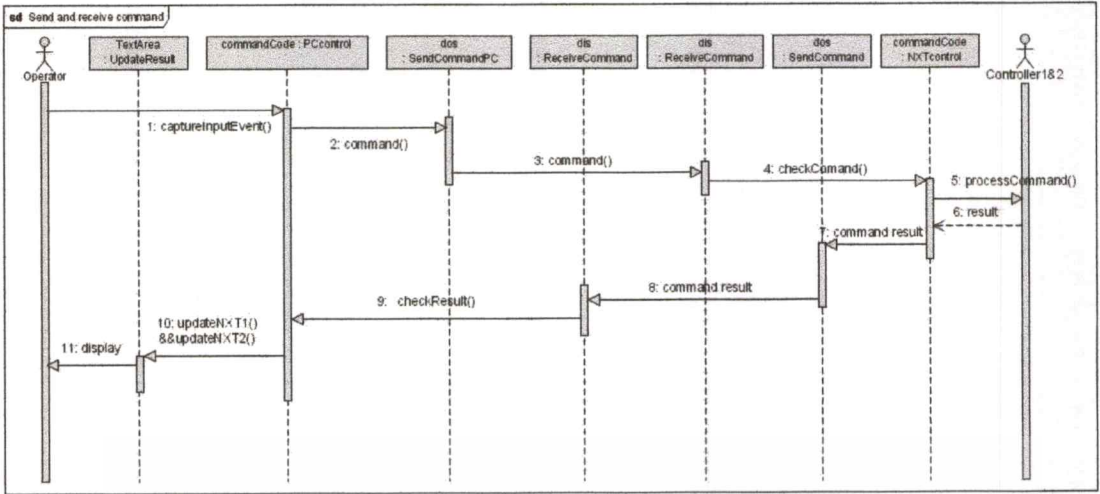
จากรูปที่ 3.10 และ 3.11 เมื่อผู้ควบคุมหุ่นยนต์ทำการสั่งยกเลิกการเชื่อมต่อไปยังเครื่องควบคุม เครื่องควบคุมจะทำการส่งคำสั่งในการขอปิดการเชื่อมต่อไปยังฝั่งหุ่นยนต์ เมื่อฝั่งหุ่นยนต์ได้รับคำสั่งขอปิดการเชื่อมต่อก็จะทำการหยุดเทรคส่งและรับข้อมูล ต่อจากนั้นจะทำการหยุดการเคลื่อนที่ของหุ่นยนต์และปรับกล้องวีดีโอให้หมุนกลับมาอยู่ยังตำแหน่งอ้างอิง ถึงจะทำการปิดการเชื่อมต่อและเข้าสู่สถานะรอการเชื่อมต่อ พร้อมกับที่ฝั่งเครื่องควบคุมก็จะทำการปิดการเชื่อมต่อเช่นกัน แต่หากพบว่าฝั่งหุ่นยนต์ไม่สามารถปิดการเชื่อมต่อได้ซึ่งอาจเกิดได้จากปัญหาจากบลูทูธชิพทำงานผิดพลาด ระบบจะทำการปิดการเชื่อมต่อซ้ำจนกว่าจะปิดการเชื่อมต่อได้

### 3.3.7 ไลอองแกรมส่วนการส่งและรับคำสั่งระหว่างฝั่งผู้ควบคุมและฝั่งหุ่นยนต์



รูปที่ 3.12 แสดงเอกทวิตรีไลอองแกรมของการส่งและรับคำสั่งระหว่างฝั่งผู้ควบคุม (Operator) และฝั่งหุ่นยนต์ (Robot)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



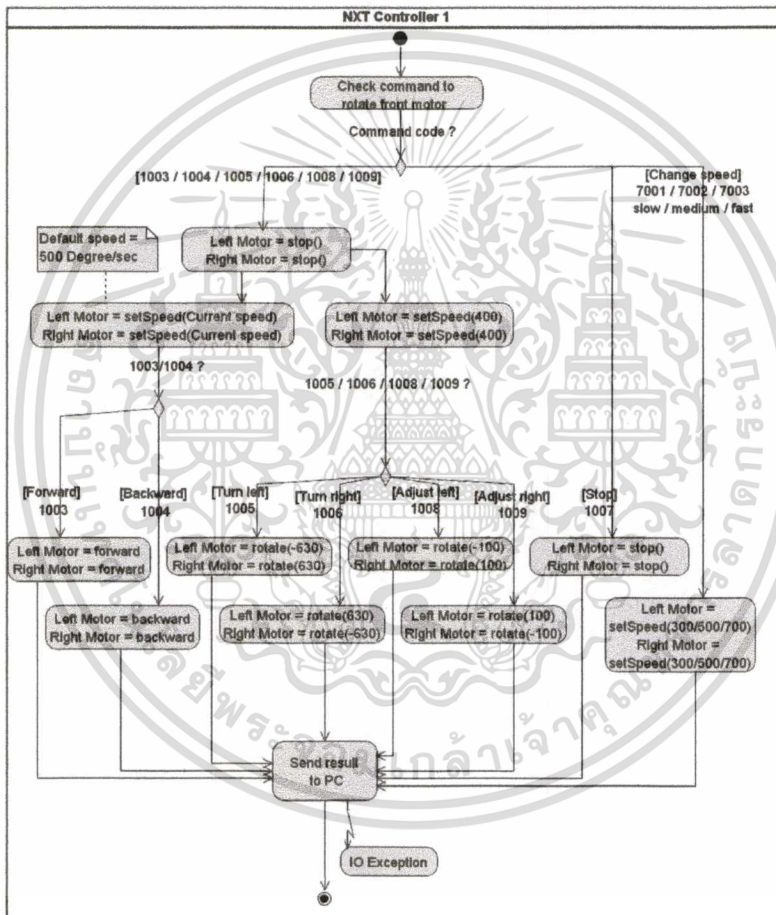
รูปที่ 3.13 แสดงซีควเอน โคอะแกรมของยูสเคสการส่งและรับคำสั่ง

จากรูปที่ 3.12 และ 3.13 เมื่อผู้ควบคุมหุ่นยนต์ทำการส่งคำสั่งใดๆ ไปยังเครื่องควบคุม เครื่องควบคุมจะทำการส่งรหัสคำสั่งในรูปแบบรหัส 4 หลัก ดังตารางที่ 3.1 ไปยังฝั่งหุ่นยนต์ เมื่อฝั่งหุ่นยนต์ได้รับคำสั่งจะทำการตรวจสอบรหัสคำสั่งว่ารหัสคำสั่งดังกล่าวส่งให้ระบบทำงานอะไรก็จะส่งงานไปยังยูสเคสการทำงานอื่นที่เกี่ยวข้องกับคำสั่งนั้นๆ และเมื่อจบการทำงานก็จะส่งผลการทำงานกลับไปฝั่งเครื่องควบคุมในรูปแบบรหัส 4 หลักตามด้วยผลลัพธ์ ต่อจากนั้น โปรแกรมฝั่งเครื่องควบคุมจะทำการแสดงผลและบันทึกข้อมูลการทำงานไว้



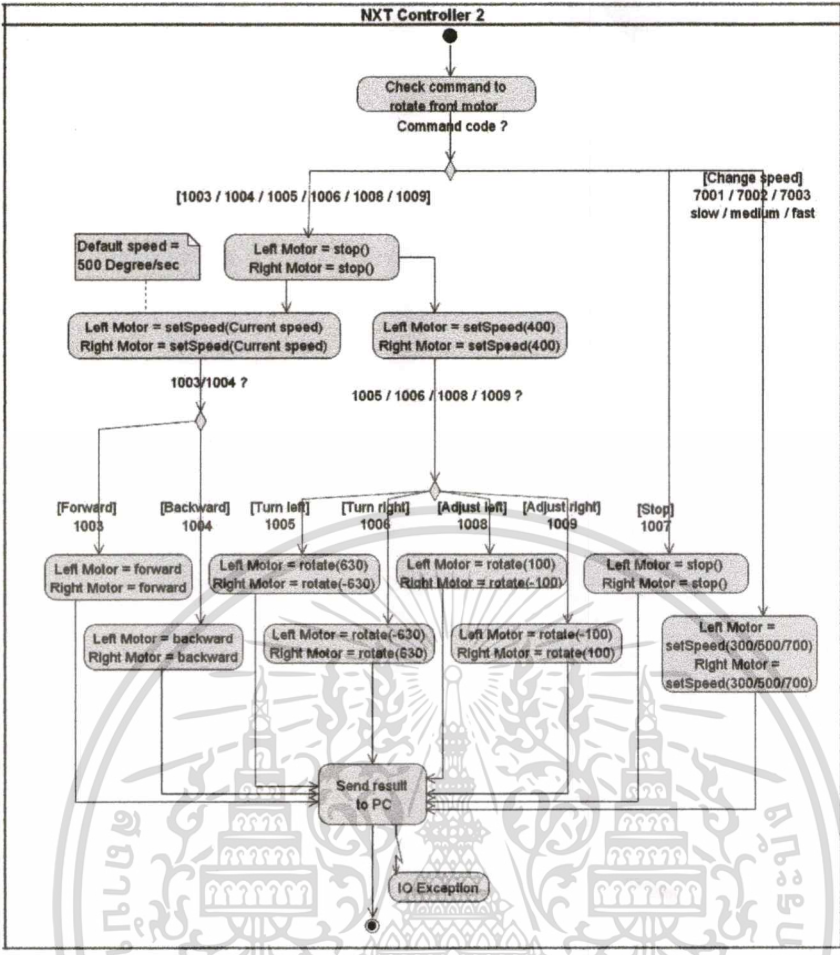
อยู่ในเซฟโหมด (Safe mode) คือจะมีการตรวจสอบระดับสัญญาณบลูทูธ หากค่าระดับสัญญาณที่ ฟังหุ่นยนต์มีค่าน้อยกว่า 203 ระบบจะทำการหยุดการเคลื่อนที่ของเอนเอ็กซ์ทีคอนโทรลเลอร์ตัวนั้น และทำการส่งคำสั่งเตือน (Risky bits) ไปยังเครื่องควบคุม เพื่อให้เครื่องควบคุมแจ้งเตือนไปยัง เอนเอ็กซ์ทีคอนโทรลเลอร์อีกตัวเพื่อสั่งให้หยุดการเคลื่อนที่ แล้วต่อจากนั้นจะทำการปิดการทำงานของ เซฟโหมดโดยอัตโนมัติ เพื่อให้ผู้ควบคุมสามารถทำการควบคุมให้หุ่นยนต์เคลื่อนที่ออกจาก บริเวณนั้น

### 3.3.9 ไคอะแกรมส่วนการควบคุมการเคลื่อนที่ของหุ่นยนต์

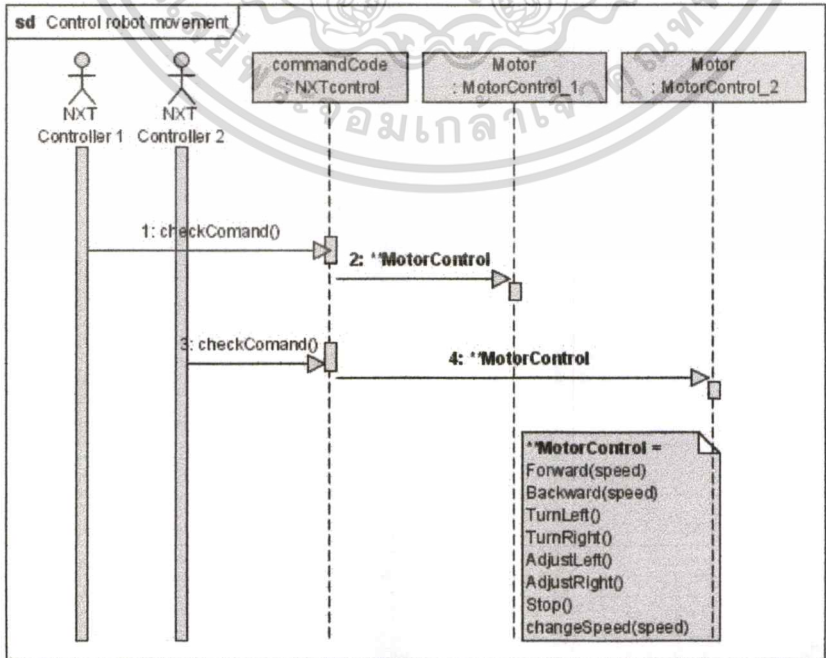


รูปที่ 3.16 แสดงแอกทิวิตไคอะแกรมการควบคุมการเคลื่อนที่ของเอนเอ็กซ์ทีคอนโทรลเลอร์

ตัวที่ 1



รูปที่ 3.17 แสดงเอกวิทัศน์โคอะแกรมการควบคุมการเคลื่อนที่ของเอนเอ็กซ์ทีคอนโทรลเลอร์ ตัวที่ 2



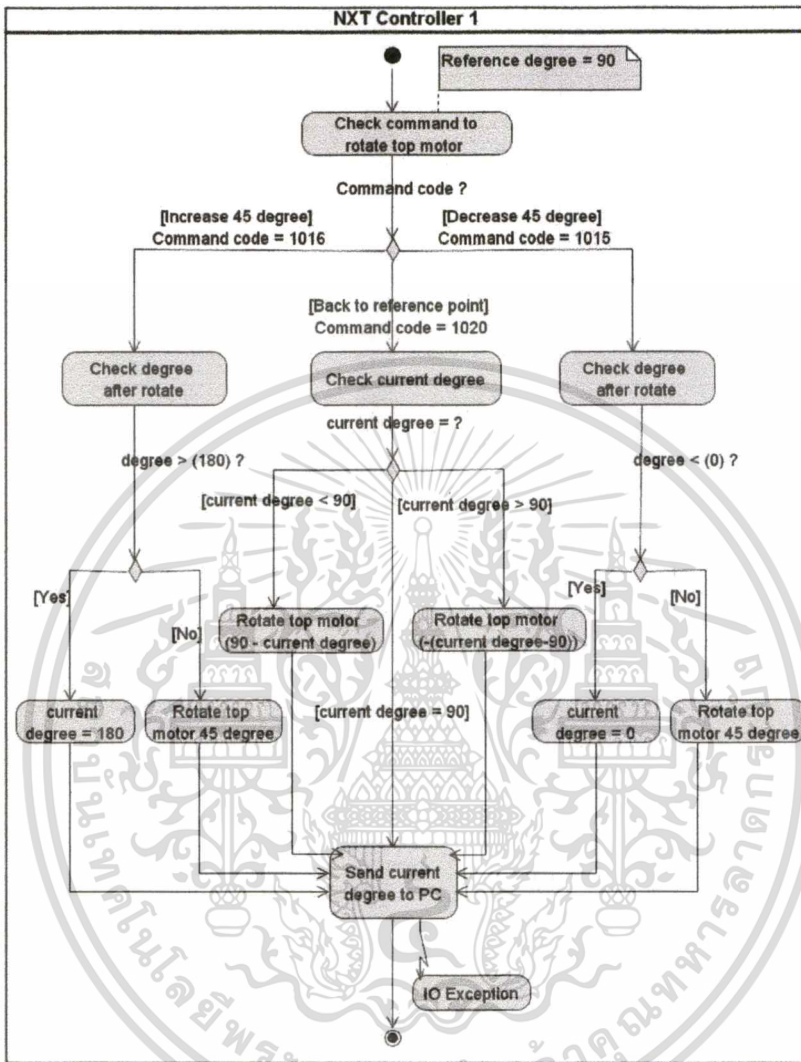
รูปที่ 3.18 แสดงซีควเอนโคอะแกรมของการขับเคลื่อนหุ่นยนต์ เอกสารนี้เป็นเอกสารที่ถึงวันหมดอายุโดยอัตโนมัติเมื่อคุณหยุดเห็นไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.16 ถึง 3.18 เมื่อระบบฝังหุ่นยนต์ได้รับคำสั่งจากฝังเครื่องควบคุมเรียบร้อยแล้ว ก็จะทำการตรวจสอบว่าคำสั่งดังกล่าวเป็นคำสั่งใด โดยเอนเอ็ทซ์ที่คอนโทรลเลอร์ตัวที่ 1 จะทำการควบคุมมอเตอร์คู่หน้าส่วนเอนเอ็ทซ์ที่คอนโทรลเลอร์ตัวที่ 2 จะทำการควบคุมมอเตอร์คู่หลัง โดยทิศทางของมอเตอร์คู่หน้าและคู่หลังจะหมุนในทิศตรงกันข้ามกัน และเมื่อทำการเคลื่อนที่เสร็จเรียบร้อยแล้วจะมีการตอบกลับผลการทำงาน ไปยังฝังเครื่องควบคุม

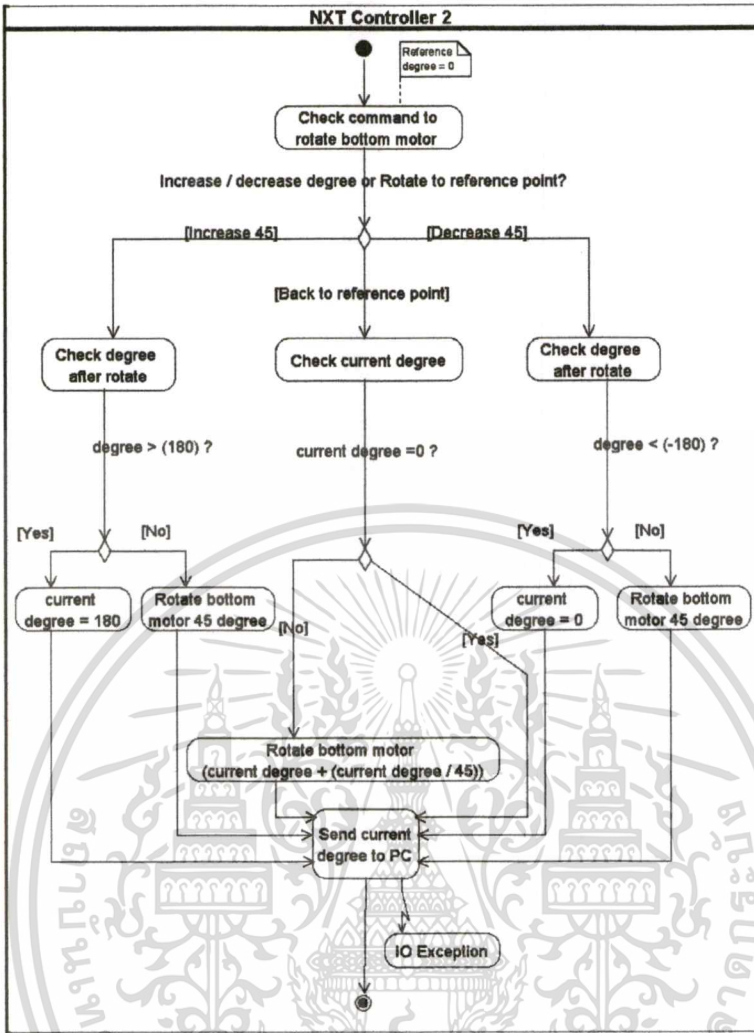
การหมุนซ้ายและขวาของหุ่นยนต์เป็นการหมุนโดยยึดเอาบริเวณจุดศูนย์กลางของหุ่นยนต์ เป็นจุดหมุนดังรูปที่ 3.19 จึงเป็นการหมุนที่ใช้พื้นที่น้อยที่สุด



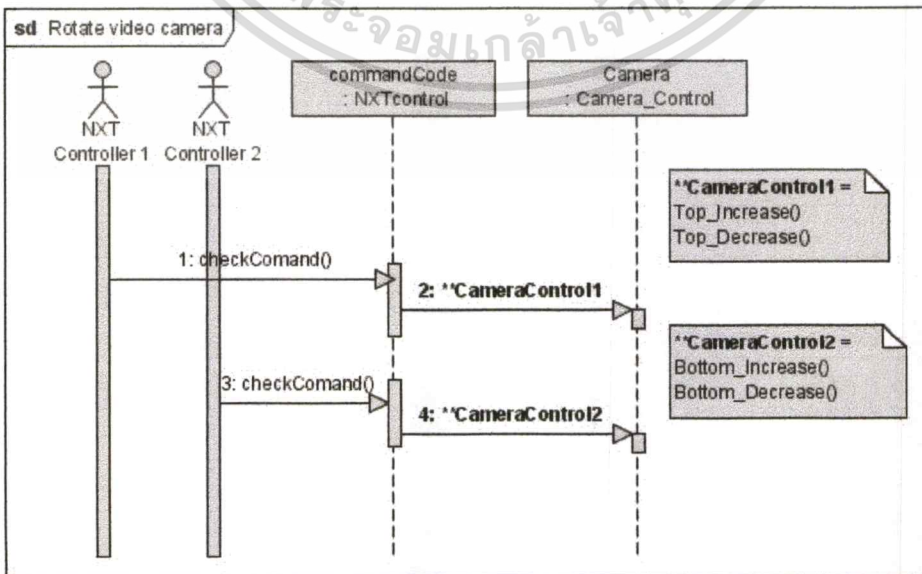
### 3.3.10 โปรแกรมส่วนการควบคุมการหมุนกลองวิดีโอ



รูปที่ 3.20 แสดงแอกทิวิตี้โปรแกรมการหมุนกลองวิดีโอของเอ็นเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 1



รูปที่ 3.21 แสดงแอกทิวิตีไดอะแกรมการหมุนกล้องวิดีโอของเอ็นเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 2

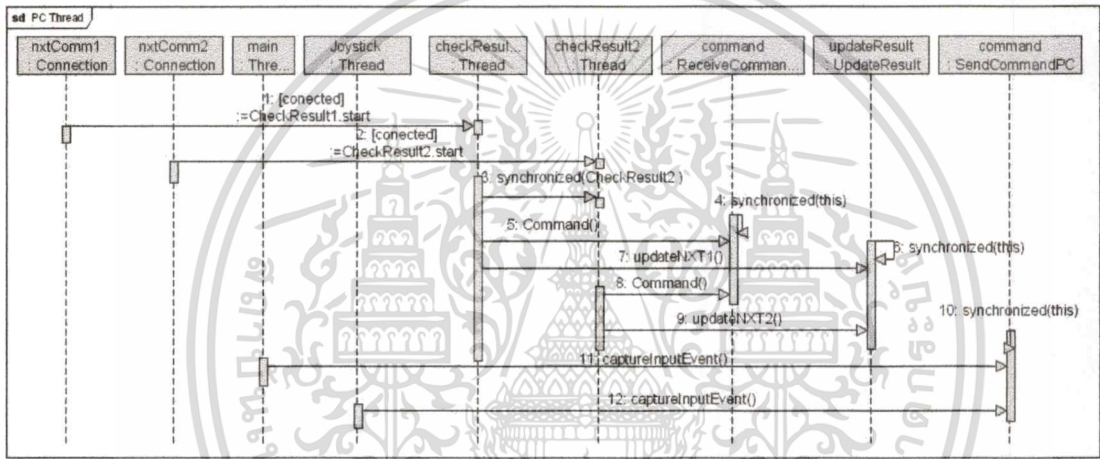


รูปที่ 3.22 แสดงซีควีนไดอะแกรมของการหมุนกล้องวิดีโอ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น มิได้อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

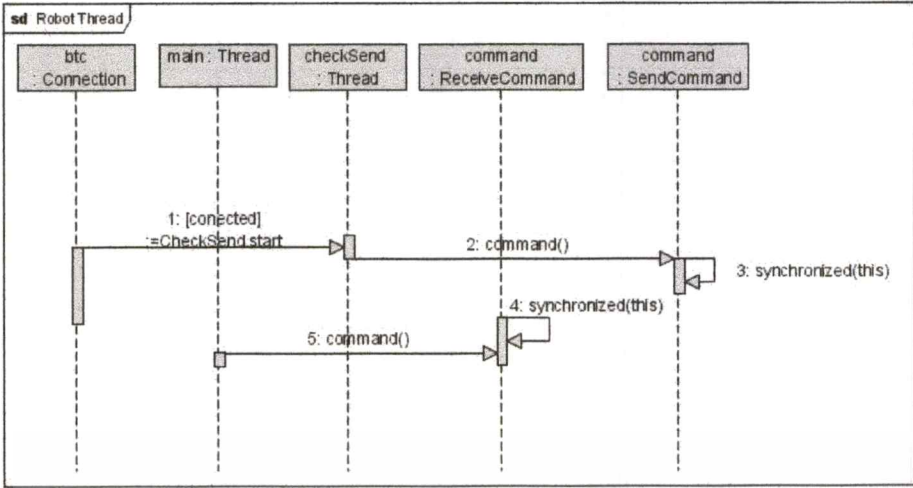
จากรูปที่ 3.20 ถึง 3.22 เมื่อระบบฝั่งหุ่นยนต์ได้รับคำสั่งจากฝั่งเครื่องควบคุมเรียบร้อยแล้วก็จะทำการตรวจสอบว่าคำสั่งดังกล่าวเป็นคำสั่งใด หากเป็นคำสั่งที่ต้องการหมุนกลิ้งในแนวระนาบ คำสั่งจะถูกส่งไปยังเอนเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 2 เพื่อหมุนมอเตอร์ตัวล่าง แต่หากเป็นคำสั่งที่ต้องการหมุนกลิ้งในแนวคิง คำสั่งจะถูกส่งไปยังเอนเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 1 เพื่อหมุนมอเตอร์ตัวบน และเมื่อทำการหมุนกลิ้งเสร็จเรียบร้อยแล้วจะทำการส่งค่าองศาเพื่อแสดงทิศทางการหันของกลิ้งไปยังเครื่องควบคุม

### 3.3.11 โค้ดแกรมส่วนการทำซิงโครไนซ์การทำงานของระบบ



รูปที่ 3.23 แสดงซีควเอนโค้ดแกรมของการทำซิงโครไนซ์การทำงานของฝั่งเครื่องควบคุม

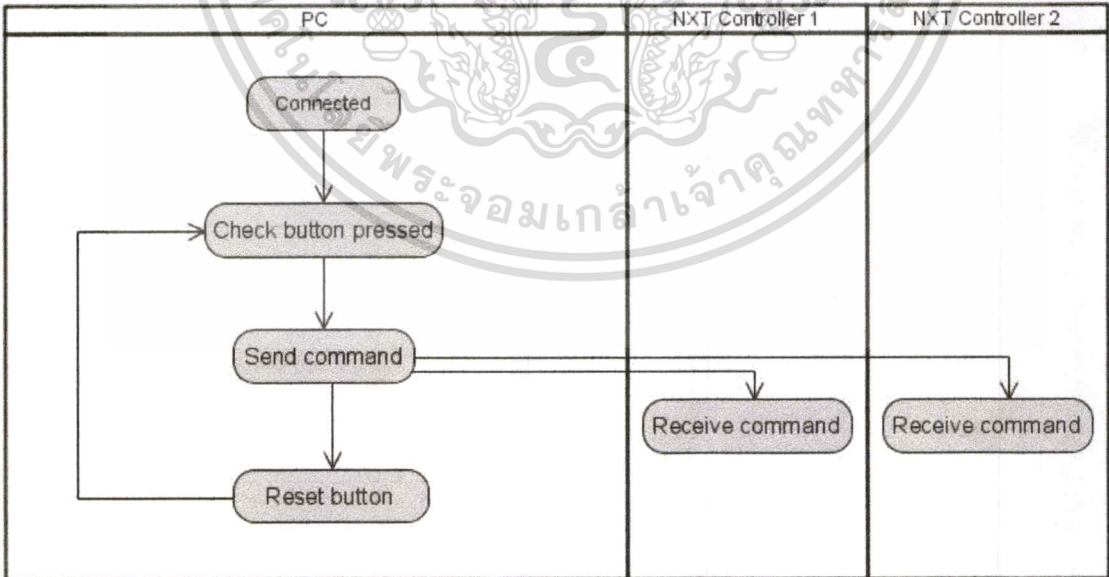
จากรูปที่ 3.23 เมื่อการเชื่อมต่อเสร็จสมบูรณ์จะทำการเริ่มเทรคการตรวจสอบผลลัพธ์ (CheckResult) ซึ่งรับข้อมูลจากส่วนการรับข้อมูลจากฝั่งหุ่นยนต์ (ReceiveCommand) หลังจากได้รับข้อมูลและทำการตรวจสอบผลลัพธ์แล้วก็จะทำการปรับปรุงผลลัพธ์ (UpdateNXT) และทำการแสดงผล ในขณะเดียวกันเมื่อมีการควบคุมการทำงานของหุ่นยนต์เทรคการส่งคำสั่งควบคุม (SendCommandPC) ก็จะเริ่มการทำงาน ซึ่งในเทรคการทำงานฝั่งเครื่องควบคุมทั้งหมดจะต้องมีการทำซิงโครไนซ์เพื่อให้การใช้ทรัพยากรในระบบเป็นไปอย่างถูกต้อง



รูปที่ 3.24 แสดงซีควเอน ไคอะแกรมของการทำซิงโครไนซ์การทำงานฝั่งหุ่นยนต์

จากรูปที่ 3.24 เมื่อการเชื่อมต่อเสร็จสมบูรณ์จะทำการเริ่มเทรคการส่งผลลัพธ์ (CheckSend) ส่วนเทรคการทำงานหลัก (Main) จะทำการเริ่มการรับคำสั่ง (ReceiveCommand) ซึ่งในเทรคการทำงานฝั่งหุ่นยนต์ทั้งหมดจะต้องมีการทำซิงโครไนซ์เพื่อให้การใช้ทรัพยากรในระบบเป็นไปอย่างถูกต้อง

### 3.3.12 ไคอะแกรมส่วนการควบคุมผ่านจอยสติ๊ก

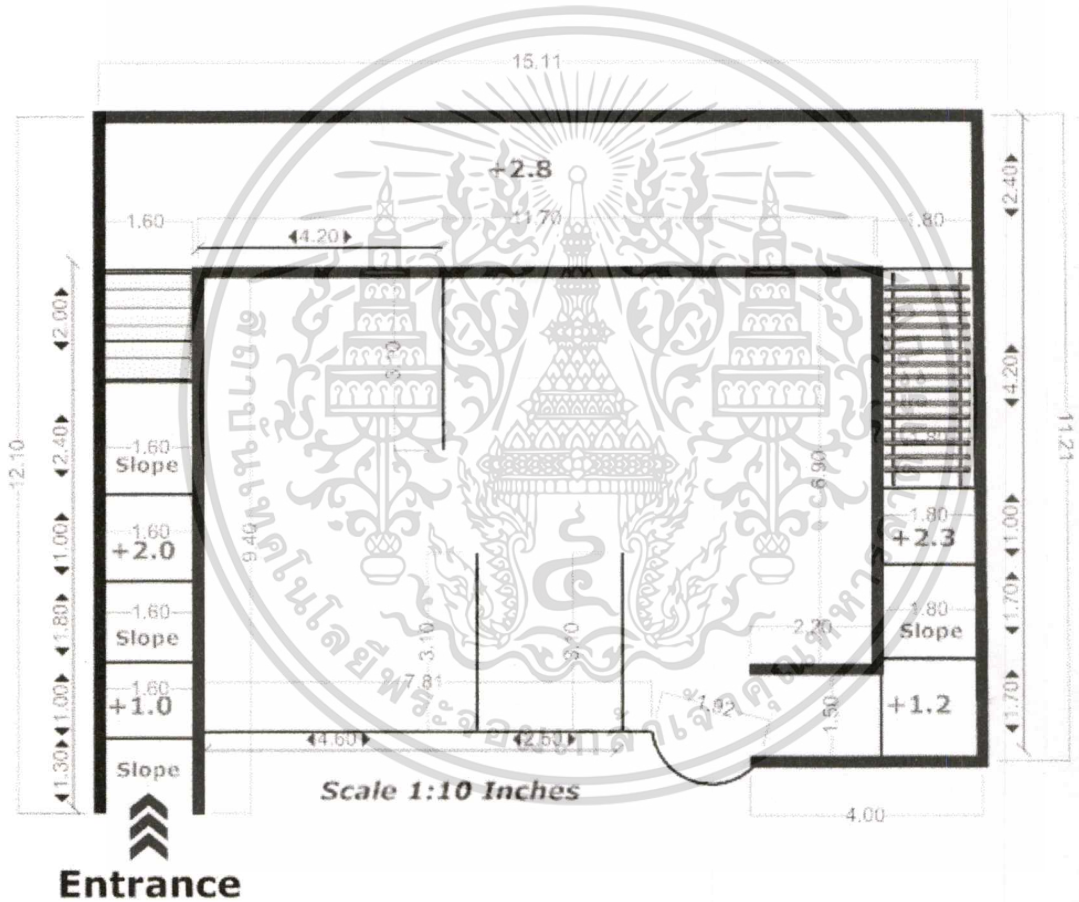


รูปที่ 3.25 แสดงแอคทีวิตี ไคอะแกรมการควบคุมผ่านจอยสติ๊ก

จากรูปที่ 3.25 เมื่อการเชื่อมต่อเสร็จสมบูรณ์ระบบจะทำการตรวจสอบการคัปของจอยสติก หากมีการคัปจะทำการตรวจสอบว่าปุ่มที่กดคือคำสั่งใด แล้วทำการส่งคำสั่งนั้นไปยังฝั่งหุ่นยนต์ หลังจากนั้นจะทำการคืนสถานะของปุ่มดังกล่าวเพื่อรอการกดครั้งต่อไป

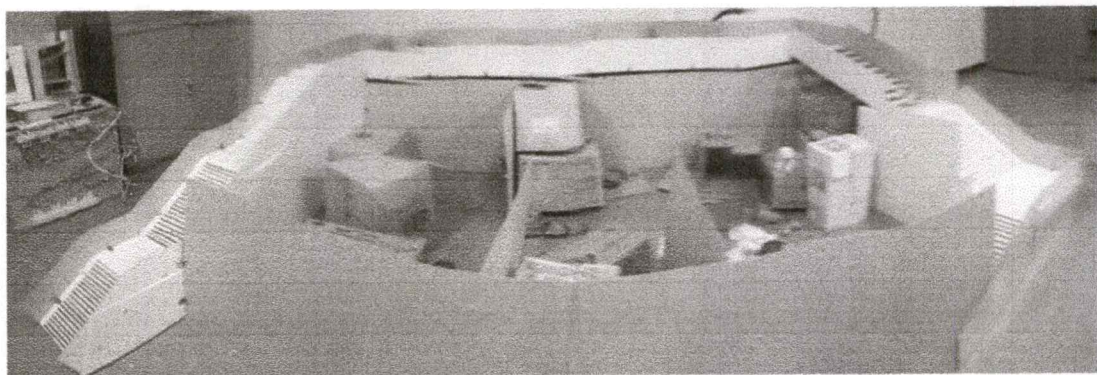
### 3.4 การออกแบบสภาพแวดล้อมที่ใช้ในการทดลอง

เพื่อทดสอบว่าระบบที่สร้างสามารถตอบสนองต่อความต้องการของระบบได้จริงจึงต้องมีการออกแบบสภาพแวดล้อมในสนามทดลองให้ประกอบด้วยส่วนประกอบต่างๆ ได้แก่ เส้นทางลาดชัน เส้นทางขรุขระต่างระดับ และสิ่งกีดขวางต่างๆ ดังรูป 3.26 และ 3.27



รูปที่ 3.26 แสดงขนาดส่วนต่างๆของสนามทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.27 แสดงลักษณะของสนามทดสอบที่มีการจำลองสภาพแวดล้อม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การทดลองและผลการทดลอง

จากการวิเคราะห์และออกแบบในบทที่ 3 ทั้งในส่วนโครงสร้างของหุ่นยนต์และระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสาบภัย ในบทนี้จะเป็นการทดลองการทำงานของหุ่นยนต์ในส่วนต่างๆตามที่ได้ออกแบบไว้ ซึ่งแยกย่อยการทดลองและสรุปผลการทดลองแต่ละการทดลองได้ดังนี้

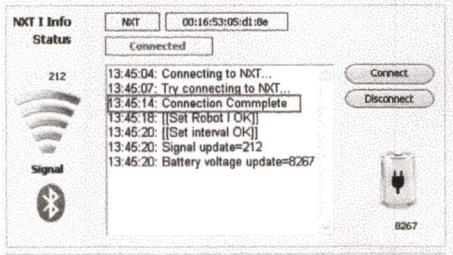
#### 4.1 การทดลองเกี่ยวกับการเชื่อมต่อระหว่างเครื่องควบคุมและหุ่นยนต์ผ่านการสื่อสารไร้สายแบบบลูทูธ

##### 4.1.1 การทดลองสร้างและยกเลิกการเชื่อมต่อ

**วัตถุประสงค์** เพื่อทดสอบการสร้างและยกเลิกการเชื่อมต่อระหว่างเครื่องควบคุมและหุ่นยนต์

**สมมุติฐาน** เมื่อหุ่นยนต์อยู่ในสถานะรอการเชื่อมต่อ เครื่องควบคุมสามารถสร้างการเชื่อมต่อไปยังหุ่นยนต์ได้ และเมื่อเครื่องควบคุมทำการยกเลิกการเชื่อมต่อ หุ่นยนต์จะต้องกลับมาอยู่ในสถานะรอการเชื่อมต่ออีกครั้ง

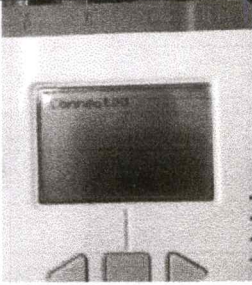
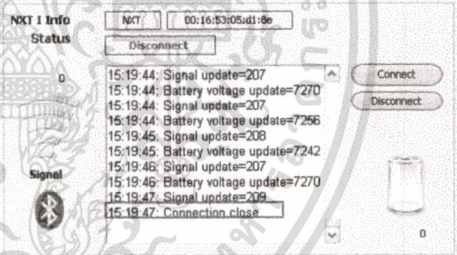

**ตารางที่ 4.1** แสดงการทดลองและผลการทดลองเกี่ยวกับการเชื่อมต่อระหว่างเครื่องควบคุมและหุ่นยนต์ผ่านการสื่อสารไร้สายแบบบลูทูธ

วิธีการทดลอง	ผลการทดลอง
1. ทำการเปิดโปรแกรมฝั่งหุ่นยนต์ และสร้างการเชื่อมต่อจากเครื่องควบคุมไปยังหุ่นยนต์ผ่านปุ่มควบคุมของโปรแกรมดังรูปที่ 4.6 หรือจอยสติ๊กสังเกตลักษณะการทำงานของหุ่นยนต์ และ โปรแกรมฝั่งเครื่องควบคุม	โปรแกรมฝั่งเครื่องควบคุมสามารถทำการเชื่อมต่อไปยังหุ่นยนต์ได้สำเร็จดังรูปที่ 4.1 และฝั่งหุ่นยนต์จะทำการแสดงผลการเชื่อมต่อดังรูปที่ 4.2  <p><b>รูปที่ 4.1</b> แสดงสถานะการเชื่อมต่อเสร็จสมบูรณ์ฝั่งเครื่องควบคุม</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ

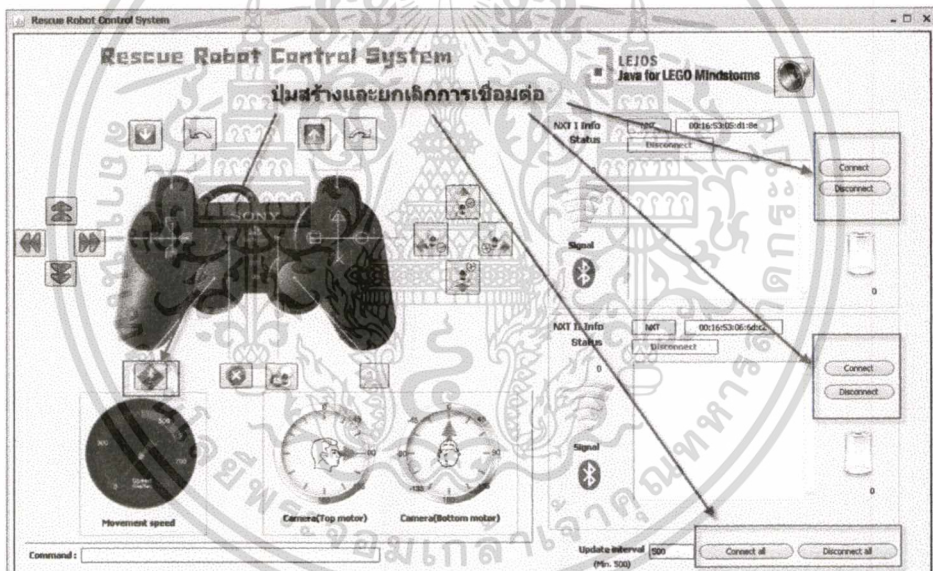
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตารางที่ 4.1 (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	 <p>รูปที่ 4.2 แสดงสถานะการเชื่อมต่อเสร็จสมบูรณ์ของฟังก์ชันอัตโนมัติ</p>
<p>2.ยกเลิกการเชื่อมต่อผ่านปุ่มควบคุมของโปรแกรมดังรูปที่ 4.6 หรือจอยสติ๊ก สังเกตลักษณะการทำงานของหุ่นยนต์ และ โปรแกรมฟังก์ชันควบคุม</p>	<p>เมื่อทำการยกเลิกการเชื่อมต่อ โปรแกรมฟังก์ชันควบคุมและหุ่นยนต์จะทำการปิดการเชื่อมต่อ ดังรูปที่ 4.3 และ 4.4 ตามลำดับ หลังจากนั้นหุ่นยนต์จะกลับมาอยู่ในสถานะรอการเชื่อมต่ออีกครั้งโดยทันที ดังรูปที่ 4.5</p>  <p>รูปที่ 4.3 แสดงสถานะการยกเลิกการเชื่อมต่อเสร็จสมบูรณ์ของฟังก์ชันควบคุม</p>
	 <p>รูปที่ 4.4 แสดงสถานะการยกเลิกการเชื่อมต่อเสร็จสมบูรณ์ของฟังก์ชันอัตโนมัติ</p>

ตารางที่ 4.1 (ต่อ)

วิธีการทดลอง	ผลการทดลอง
	 <p>รูปที่ 4.5 แสดงสถานะรอการเชื่อมต่อฝั่งหุ่นยนต์</p>



รูปที่ 4.6 แสดงส่วนการสร้างและยกเลิกการเชื่อมต่อของโปรแกรมฝั่งเครื่องควบคุม

### วิเคราะห์ผลการทดลอง

เครื่องควบคุมจะสามารถสร้างการเชื่อมต่อไปยังหุ่นยนต์ได้ก็ต่อเมื่อฝั่งหุ่นยนต์ได้ทำการเปิดโปรแกรมแล้วเท่านั้น สำหรับการยกเลิกการเชื่อมต่อนั้นเมื่อฝั่งเครื่องควบคุมทำการปิดการเชื่อมต่อแล้ว ฝั่งหุ่นยนต์จะกลับมาอยู่ในสถานะรอการเชื่อมต่ออีกครั้งโดยทันที แต่จากการทดลองหลายครั้งพบว่าอาจจะเกิดเหตุการณ์ที่บลูทูธชิพไม่สามารถปิดการเชื่อมต่อที่ฝั่งหุ่นยนต์ได้โดยทันที แต่ระบบสามารถแก้ไขปัญหานี้ได้โดยอาจกินเวลามากขึ้นแต่ใช้เวลาไม่เกิน 1 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

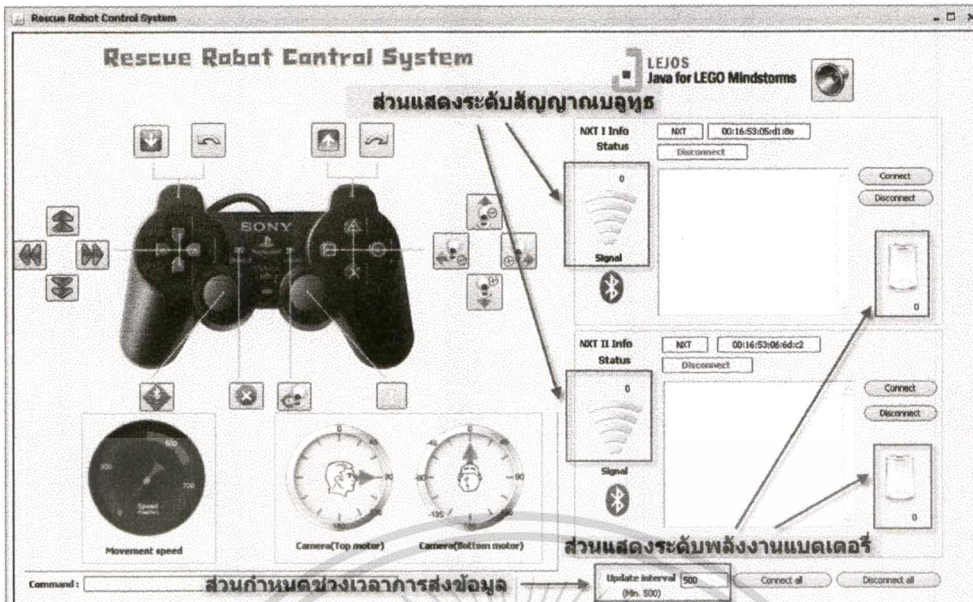
#### 4.1.2 การทดลองส่งข้อมูลระดับสัญญาณบลูทูธและค่าแรงดันไฟฟ้าของแบตเตอรี่ระหว่างเครื่องควบคุมและหุ่นยนต์

**วัตถุประสงค์** เพื่อทดสอบการส่งข้อมูลค่าระดับสัญญาณบลูทูธและค่าระดับแรงดันไฟฟ้าของแบตเตอรี่มายังฝั่งเครื่องควบคุมเมื่อการเชื่อมต่อเสร็จสมบูรณ์แล้ว

**สมมุติฐาน** เมื่อการเชื่อมต่อระหว่างเครื่องควบคุมและหุ่นยนต์เสร็จสมบูรณ์แล้ว ฝั่งหุ่นยนต์ทำการส่งค่าระดับสัญญาณบลูทูธและค่าระดับแรงดันไฟฟ้าของแบตเตอรี่มายังฝั่งเครื่องควบคุมตามระยะเวลาที่กำหนดไว้

**ตารางที่ 4.2** แสดงการทดลองและผลการทดลองส่งข้อมูลระดับสัญญาณบลูทูธและค่าแรงดันไฟฟ้าของแบตเตอรี่ระหว่างเครื่องควบคุมและหุ่นยนต์

วิธีการทดลอง	ผลการทดลอง
<p>ทำการเปิดโปรแกรมฝั่งหุ่นยนต์และสร้างการเชื่อมต่อจากเครื่องควบคุมไปยังหุ่นยนต์โดยการกำหนดช่วงเวลาการส่งข้อมูลจากฝั่งหุ่นยนต์เท่ากับ 500 มิลลิวินาที ดังรูปที่ 4.8 สังเกตลักษณะการทำงานของหุ่นยนต์ และ โปรแกรมฝั่งเครื่องควบคุม</p>	<p>เมื่อการเชื่อมต่อระหว่างเครื่องควบคุมและหุ่นยนต์เสร็จสมบูรณ์ ฝั่งหุ่นยนต์ทำการส่งค่าระดับสัญญาณบลูทูธและค่าระดับแรงดันไฟฟ้าของแบตเตอรี่มายังฝั่งเครื่องควบคุมดังรูปที่ 4.7</p> <div data-bbox="696 1223 1176 1500" style="text-align: center;"> </div> <p>รูปที่ 4.7 แสดงค่าระดับสัญญาณบลูทูธและค่าระดับแรงดันไฟฟ้าของแบตเตอรี่ที่ฝั่งเครื่องควบคุม</p>



รูปที่ 4.8 แสดงส่วนแสดงระดับสัญญาณบลูทูธ ระดับแรงดันไฟฟ้าของแบตเตอรี่ และส่วนกำหนดช่วงเวลาการส่งข้อมูลของโปรแกรมฝั่งเครื่องควบคุม

#### วิเคราะห์ผลการทดลอง

เมื่อการเชื่อมต่อระหว่างเครื่องควบคุมและหุ่นยนต์เสร็จสมบูรณ์ ฝั่งหุ่นยนต์ทำการส่งค่าระดับสัญญาณบลูทูธและค่าระดับแรงดันไฟฟ้าของแบตเตอรี่มายังฝั่งเครื่องควบคุม โดยระยะเวลาของช่วงการส่งแต่ละครั้งพบว่าอาจใช้เวลามากกว่า 500 มิลลิวินาที เนื่องจากดีเลย์ของการส่งข้อมูล


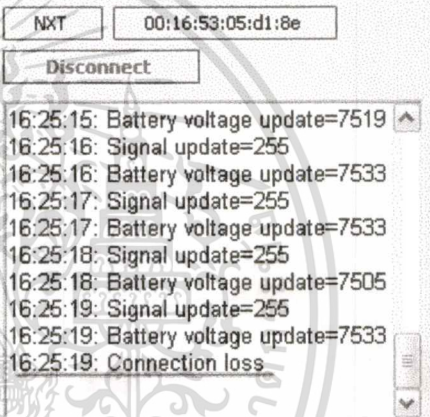

#### 4.1.3 การทดลองกลไกการทำงานเมื่อการเชื่อมต่อถูกตัดขาด

**วัตถุประสงค์** เพื่อทดสอบการทำงานเมื่อการเชื่อมต่อถูกตัดขาด

**สมมุติฐาน**

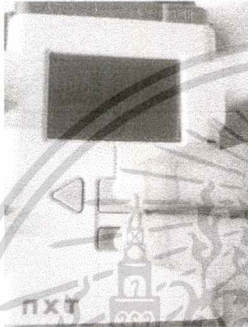
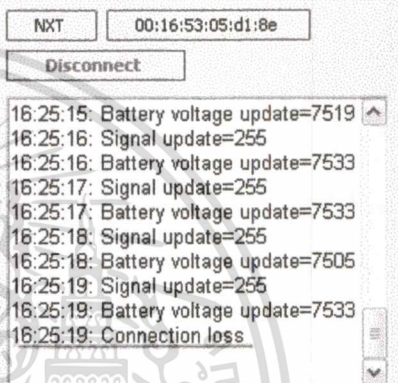
1. หากการเชื่อมต่อถูกตัดขาดจากข้อผิดพลาดที่ฝั่งเครื่องควบคุม เช่น เครื่องควบคุมปิดการทำงานอย่างกะทันหัน หุ่นยนต์จะต้องทำการตัดการเชื่อมต่อและกลับไปอยู่ในสถานะรอการเชื่อมต่อ
2. หากการเชื่อมต่อถูกตัดขาดจากข้อผิดพลาดที่ฝั่งหุ่นยนต์ เช่น แบตเตอรี่ของหุ่นยนต์หมด เครื่องควบคุมจะต้องทำการตัดการเชื่อมต่อและแจ้งเตือนผู้ควบคุม

**ตารางที่ 4.3** แสดงการทดลองและผลการทดลองกลไกการทำงานเมื่อการเชื่อมต่อถูกตัดขาด

วิธีการทดลอง	ผลการทดลอง
<p>1.ทำการเปิด โปรแกรมฟังหุ่นยนต์ และสร้างการเชื่อมต่อจากเครื่องควบคุมไปยังหุ่นยนต์</p> <p>2.จำลองสถานะการณ์การเชื่อมต่อถูกตัดขาดจากข้อผิดพลาดที่ฟังเครื่องควบคุม โดยการปิดการรับส่งสัญญาณบลูทูธดังรูปที่ 4.9 สังเกตลักษณะการทำงานของหุ่นยนต์ และ โปรแกรมฟังเครื่องควบคุม</p>  <p>รูปที่ 4.9 แสดงการปิดการรับส่งสัญญาณบลูทูธฟังเครื่องควบคุม</p>	<p>โปรแกรมฟังเครื่องควบคุมตรวจพบว่าการเชื่อมต่อถูกตัดขาดดังรูปที่ 4.10 ส่วนฟังหุ่นยนต์ใช้เวลา 4 ถึง5 วินาทีในการตรวจพบว่าการเชื่อมต่อถูกตัดขาดแล้วจึงทำการตัดการเชื่อมต่อและกลับไปอยู่ในสถานะรอการเชื่อมต่อดังรูปที่ 4.11</p>  <p>รูปที่ 4.10 แสดงสถานะการเชื่อมต่อตัดขาด</p>  <p>รูปที่ 4.11 แสดงสถานะรอการเชื่อมต่อฟังหุ่นยนต์</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตารางที่ 4.3 (ต่อ)

วิธีการทดลอง	ผลการทดลอง
<p>3. ทำซ้ำข้อ 1 และจำลองสถานะการณ์การเชื่อมต่อถูกตัดขาดจากข้อผิดพลาดที่ฝั่งหุ่นยนต์ โดยการปิดการทำงานของหุ่นยนต์ ดังรูปที่ 4.12 สังเกตลักษณะการทำงานของหุ่นยนต์ และ โปรแกรมฝั่งเครื่องควบคุม</p>  <p>รูปที่ 4.12 แสดงการปิดการทำงานของหุ่นยนต์</p>	<p>เมื่อทำการปิดการทำงานของหุ่นยนต์แล้ว ฝั่งเครื่องควบคุมใช้เวลา 15 ถึง 20 วินาทีในการตรวจพบว่าการเชื่อมต่อถูกตัดขาดแล้วจึงทำการตัดการเชื่อมต่อดังรูปที่ 4.13</p>  <p>รูปที่ 4.13 แสดงสถานะการเชื่อมต่อตัดขาด</p>

## วิเคราะห์ผลการทดลอง



เมื่อการเชื่อมต่อถูกตัดขาดระบบทั้งสองฝั่งจะตรวจพบข้อผิดพลาดและทำการปิดการเชื่อมต่อเดิมแล้วกลับสู่สถานะก่อนการสร้างการเชื่อมต่อ แต่จากการทดลองพบว่าระยะเวลาที่ใช้ในการตรวจพบว่าการเชื่อมต่อถูกตัดขาดของระบบฝั่งเครื่องควบคุมมีมากกว่าฝั่งหุ่นยนต์ถึงประมาณ 4 เท่า

## 4.2 การทดลองควบคุมการเคลื่อนที่ของหุ่นยนต์

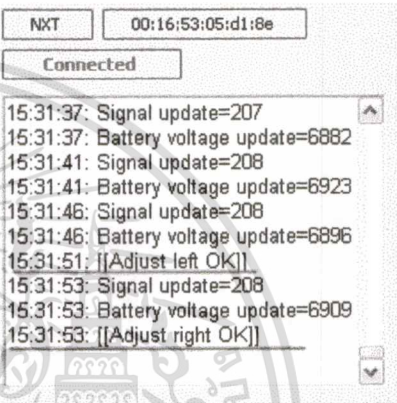
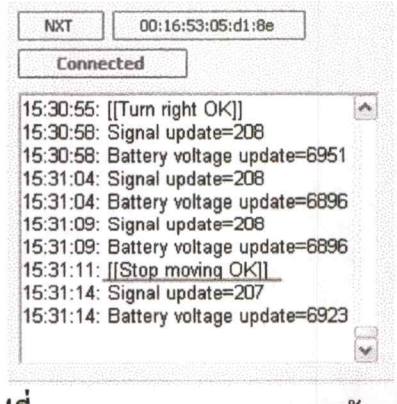
**วัตถุประสงค์** เพื่อทดสอบความถูกต้องระหว่างคำสั่งและการเคลื่อนที่ของหุ่นยนต์ รวมถึงการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุม

**สมมุติฐาน** เมื่อมีการส่งคำสั่งการเคลื่อนที่ไปยังหุ่นยนต์ผ่านปุ่มควบคุมของโปรแกรมหรือจอยสติ๊ก หุ่นยนต์สามารถเคลื่อนที่ได้ตรงตามคำสั่งที่ส่งการ และมีการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุมและมีการแสดงผลด้วยเสียง

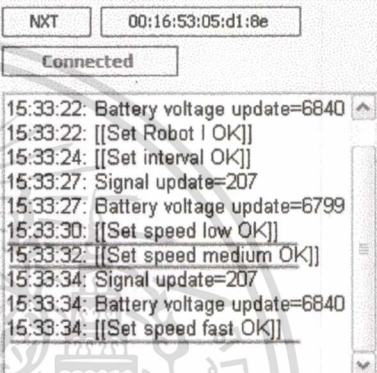
ตารางที่ 4.4 แสดงการทดลองและผลการทดลองควบคุมการเคลื่อนที่ของหุ่นยนต์

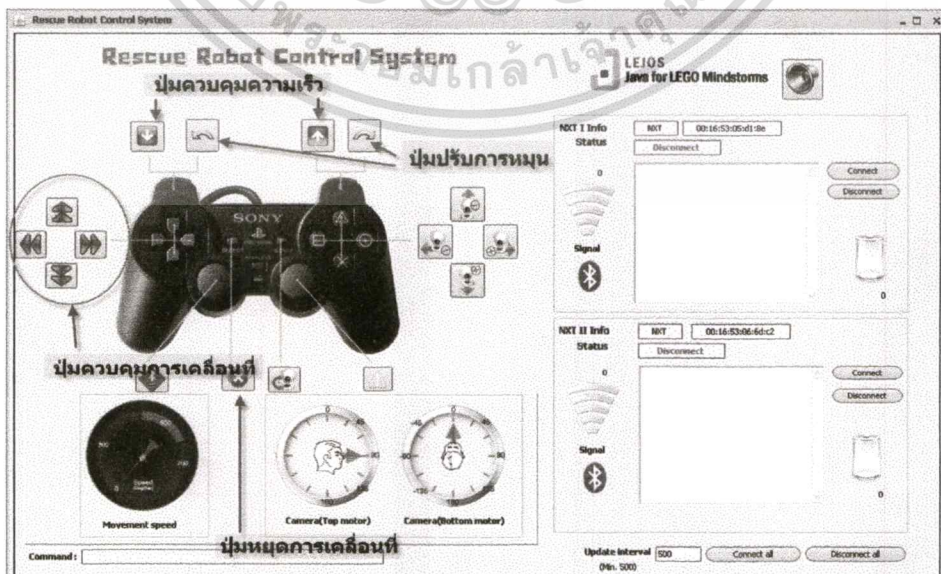
วิธีการทดลอง	ผลการทดลอง
<p>1.ทำการเปิดโปรแกรมฝั่งหุ่นยนต์ และสร้างการเชื่อมต่อจากเครื่องควบคุมไปยังหุ่นยนต์</p> <p>2.ทดลองควบคุมการเดินหน้าและถอยหลัง ทดลองสั่งคำสั่งเดินหน้าและถอยหลังจากปุ่มดังรูปที่4.19 พร้อมกับสังเกตลักษณะการทำงานของหุ่นยนต์</p>	<p>หุ่นยนต์สามารถเคลื่อนที่ไปด้านหน้าและด้านหลังได้ถูกต้องตามคำสั่งที่ได้รับจากเครื่องควบคุม และมีการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุมดังรูปที่ 4.14</p>  <p>รูปที่ 4.14 แสดงผลการตอบกลับการเดินหน้าและถอยหลังมายังฝั่งเครื่องควบคุม</p>
<p>3.ทดลองควบคุมการหมุนซ้ายและขวา ทดลองสั่งคำสั่งหมุนซ้ายและขวาจากปุ่มดังรูปที่ 4.19 พร้อมกับสังเกตลักษณะการทำงานของหุ่นยนต์</p>	<p>หุ่นยนต์สามารถหมุนซ้ายและขวาได้ถูกต้องตามคำสั่งที่ได้รับจากเครื่องควบคุม โดยองศาที่หมุนไปประมาณ 90 องศา และมีการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุมดังรูปที่ 4.15</p>  <p>รูปที่ 4.15 แสดงผลการตอบกลับการหมุนซ้ายและขวามายังฝั่งเครื่องควบคุม</p>

## ตารางที่ 4.4 (ต่อ)

วิธีการทดลอง	ผลการทดลอง
<p>4.ทดลองควบคุมปรับการหมุนซ้ายและขวา ทดลองสั่งคำสั่งปรับการหมุนซ้าย และขวาจากปุ่มดังรูปที่ 4.19 พร้อมกับ สังเกตลักษณะการทำงานของหุ่นยนต์</p>	<p>หุ่นยนต์สามารถปรับการหมุนซ้าย และขวาได้ถูกตามคำสั่งที่ได้รับจาก เครื่องควบคุม และมีการตอบกลับผล การทำงานมายังฝั่งเครื่องควบคุมดังรูป ที่ 4.16</p>  <p>รูปที่ 4.16 แสดงผลการตอบกลับการ ปรับหมุนซ้ายและขวามายังฝั่งเครื่อง ควบคุม</p>
<p>5.ทดลองควบคุมการหยุดการเคลื่อนที่ ทดลองสั่งคำสั่งหยุดการเคลื่อนที่จาก ปุ่มดังรูปที่ 4.19 ในขณะที่หุ่นยนต์เดินหน้า หรือถอยหลังพร้อมกับสังเกตลักษณะการ ทำงานของหุ่นยนต์</p>	<p>หุ่นยนต์สามารถหยุดการเคลื่อนที่ได้ ถูกตามคำสั่งที่ได้รับจากเครื่องควบคุม และมีการตอบกลับผลการทำงานมายัง ฝั่งเครื่องควบคุมรูปที่ 4.17</p>  <p>รูปที่ 4.17 แสดงผลการตอบกลับการ หยุดการเคลื่อนที่มายังฝั่งเครื่องควบคุม</p>

ตารางที่ 4.4 (ต่อ)

วิธีการทดลอง	ผลการทดลอง
<p>6.ทดลองควบคุมการปรับความเร็วการเคลื่อนที่</p> <p>ทดลองตั้งคำสั่งปรับความเร็วจากปุ่มดังรูปที่ 4.19 ในขณะที่หุ่นยนต์เดินหน้าหรือถอยหลังให้ครบทั้ง 3 ระดับคือ ช้า ปานกลาง และ เร็ว พร้อมกับสังเกตลักษณะการทำงานของหุ่นยนต์</p>	<p>หุ่นยนต์สามารถควบคุมการปรับความเร็วการเคลื่อนที่ได้ถูกต้องตามคำสั่งที่ได้รับจากเครื่องควบคุม และมีการตอบกลับผลการทำงานมาซึ่งฝั่งเครื่องควบคุมดังรูปที่ 4.18</p>  <p>รูปที่ 4.18 แสดงผลการตอบกลับการปรับความเร็วการเคลื่อนที่มายังฝั่งเครื่องควบคุม</p>



รูปที่ 4.19แสดงส่วนการควบคุมการเคลื่อนที่ของโปรแกรมฝั่งเครื่องควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## วิเคราะห์ผลการทดลอง

หุ่นยนต์สามารถเคลื่อนที่ได้ตรงตามคำสั่งที่สั่งการ และมีการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุม แต่พบว่าในบางครั้งการเคลื่อนที่ของหุ่นยนต์มีความคลาดเคลื่อนเกิดขึ้นได้จากผลกระทบจากปัจจัยต่างๆดังนี้

1. ปัจจัยด้านสภาพพื้นผิวที่แตกต่างกันทำให้การยึดติดระหว่างล้อและพื้นผิวต่างกันไป ทำให้การหมุนซ้ายหรือขวาไม่เท่ากับ 90 องศา

2. ปัจจัยด้านการเริ่มทำงานที่ไม่พร้อมกัน เนื่องจากการทำงานของหุ่นยนต์อาศัยการควบคุมมอเตอร์คู่หน้าและหลังผ่านเอนเอ็ทซ์ที่คอนโทรลเลอร์คนละตัว ซึ่งไม่สามารถทำการซิงโครไนซ์ข้อมูลซึ่งกันและกันได้ ทำให้เมื่อมีคำสั่งไปยังเอนเอ็ทซ์ที่คอนโทรลเลอร์แต่ละตัว อาจทำให้การควบคุมการทำงานมอเตอร์คู่หน้าและหลังเริ่มการทำงานได้ไม่พร้อมกัน จึงอาจทำให้เกิดความคลาดเคลื่อนในกรณีควบคุมการเดินหน้าถอยหลัง และการเลี้ยวซ้าย ขวา

แนวทางแก้ไขสำหรับปัญหา คือการปรับการหมุนซ้ายและขวา ซึ่งผู้ควบคุมสามารถปรับทิศทางของหุ่นยนต์ให้เป็นไปตามที่ต้องการได้ โดยการปรับหนึ่งครั้งจะเท่ากับการหมุนล้อ 100 องศา หรือประมาณ 0.27 รอบ

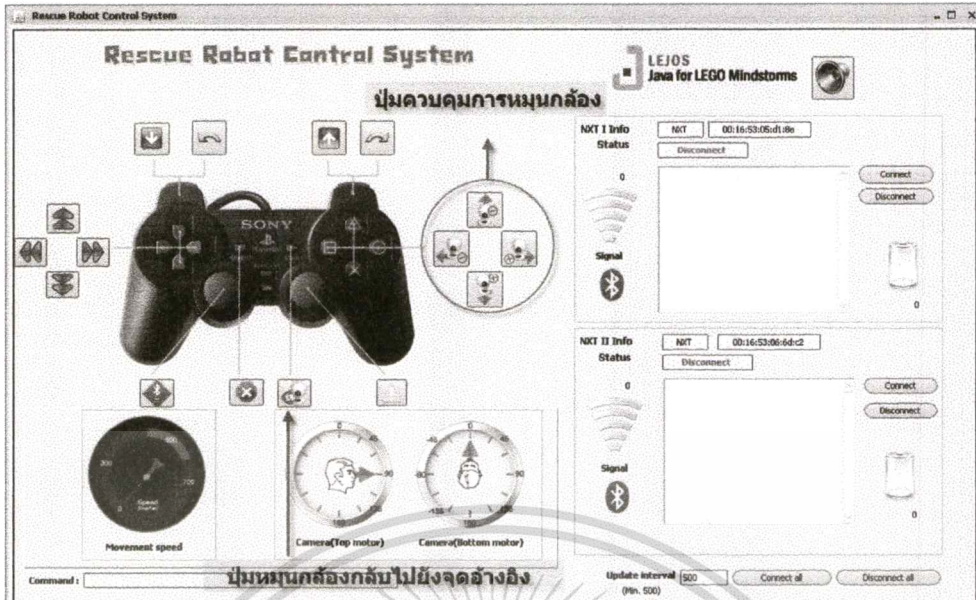
## 4.3 การทดลองควบคุมการหมุนกล้องวิดีโอ

**วัตถุประสงค์** เพื่อทดสอบความถูกต้องระหว่างคำสั่งและการหมุนของกล้องวิดีโอ รวมถึงการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุม

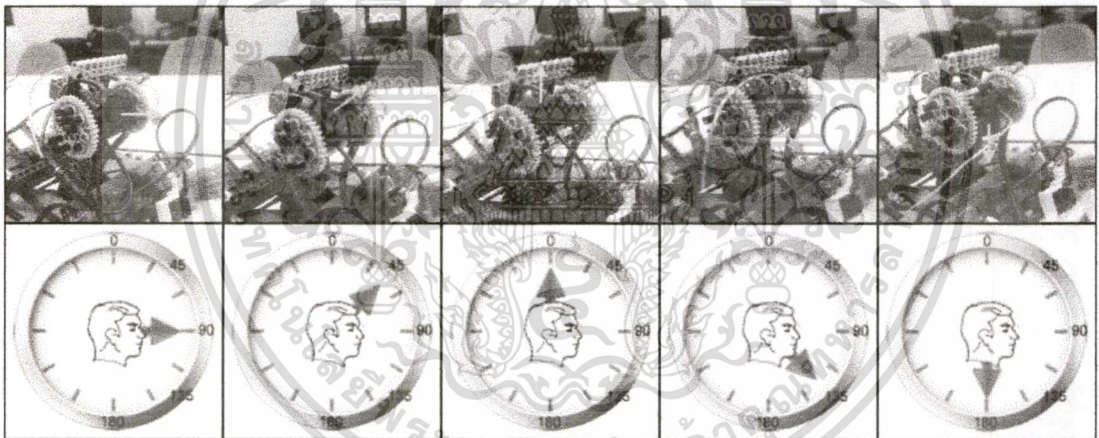
**สมมุติฐาน** เมื่อมีการสั่งคำสั่งการเคลื่อนที่ไปยังหุ่นยนต์ผ่านปุ่มควบคุมของโปรแกรมหรือจอยสติ๊ก หุ่นยนต์สามารถหมุนกล้องได้ตรงตามคำสั่งที่สั่งการ และมีการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุมและมีการแสดงผลด้วยรูปภาพแสดงทิศทางการหันของกล้อง

ตารางที่ 4.5 แสดงการทดลองและผลการทดลองควบคุมการหมุนกลิ้งวีดีโอ

วิธีการทดลอง	ผลการทดลอง
<p>1. ทำการเปิดโปรแกรมฝั่งหุ่นยนต์ และสร้างการเชื่อมต่อจากเครื่องควบคุมไปยังหุ่นยนต์</p> <p>2. ทดลองควบคุมการหมุนกลิ้งวีดีโอในแนวตั้ง จากปุ่มดังรูปที่ 4.20</p>	<p>หุ่นยนต์สามารถหมุนกลิ้งวีดีโอในแนวตั้งได้ถูกตามคำสั่งที่ได้รับจากเครื่องควบคุม และมีการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุมดังรูป โดยการหมุนกลิ้งวีดีโอในแนวตั้งจะสามารถหมุนได้ครั้งละ 45 องศา และมีมุมการหมุนจำกัดอยู่ที่ 0 ถึง 180 องศา ดังรูปที่ 4.21</p>
<p>3. ทดลองควบคุมการหมุนกลิ้งวีดีโอในแนวระนาบ จากปุ่มดังรูปที่ 4.20</p>	<p>หุ่นยนต์สามารถหมุนกลิ้งวีดีโอในแนวระนาบได้ถูกตามคำสั่งที่ได้รับจากเครื่องควบคุม และมีการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุมดังรูป โดยการหมุนกลิ้งวีดีโอในแนวตั้งจะสามารถหมุนได้ครั้งละ 45 องศา และมีมุมการหมุนจำกัดอยู่ที่ -180 ถึง 180 องศา ดังรูปที่ 4.22</p>
<p>4. ทดลองควบคุมการหมุนกลิ้งวีดีโอกลับไปยังจุดอ้างอิง จากปุ่มดังรูปที่ 4.20</p>	<p>หุ่นยนต์สามารถหมุนกลิ้งวีดีโอกลับมายังจุดอ้างอิงได้</p>

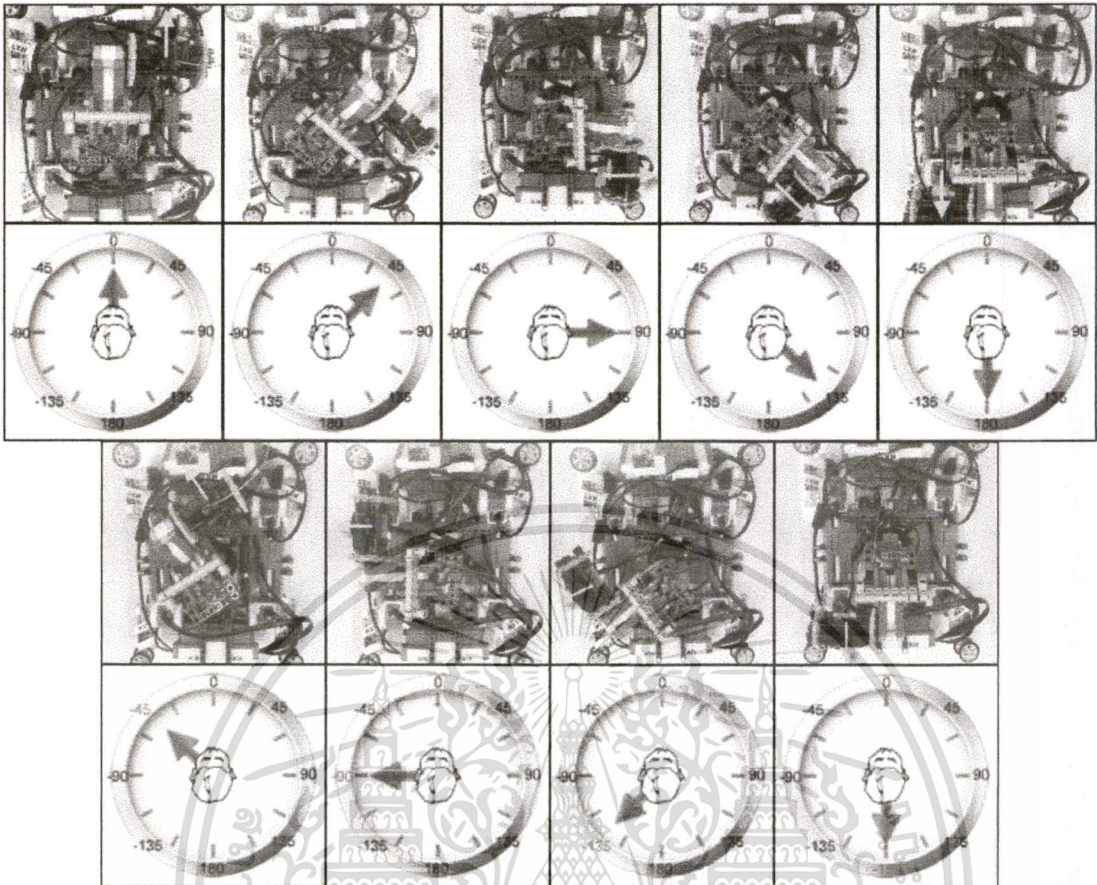


รูปที่ 4.20 แสดงส่วนการควบคุมการหมุนกลิ้งวิดีโอของโปรแกรมฝั่งเครื่องควบคุม



รูปที่ 4.21 แสดงความสัมพันธ์ระหว่างการหมุนกลิ้งกับส่วนแสดงทิศทางก้นของกล้องในแนวดิ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 แสดงความสัมพันธ์ระหว่างการหมุนของกลองกับส่วนแสดงทิศทางการหันของกลองในแนวระนาบ

#### วิเคราะห์ผลการทดลอง

หุ่นยนต์สามารถหมุนกลองวิดีโอได้ทั้งในแนวตั้งและแนวระนาบได้ถูกต้องตามคำสั่งที่ได้รับจากเครื่องควบคุม และสามารถควบคุมการหมุนกลองวิดีโอกลับให้กลับยังจุดอ้างอิงได้ โดยการควบคุมทุกครั้งจะมีการตอบกลับผลการทำงานมายังฝั่งเครื่องควบคุม และแสดงผลด้วยรูปภาพแสดงทิศทางการหันของกลอง

#### 4.4 การทดลองการเคลื่อนที่ของหุ่นยนต์ในสภาพแวดล้อมจำลอง

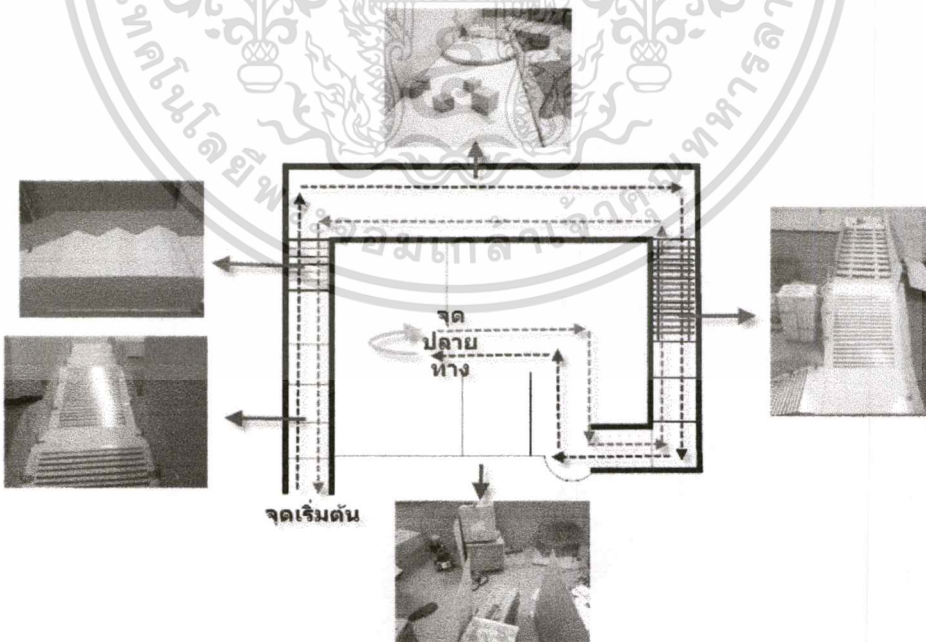
**วัตถุประสงค์** เพื่อทดสอบการเคลื่อนที่ของหุ่นยนต์ในสภาพพื้นผิวที่แตกต่างกัน ได้แก่ ทางลาดชัน ทางขรุขระและทางต่างระดับ และหาระดับความเร็วในการเคลื่อนที่ที่เหมาะสมต่อการขึ้นลงทางลาดชัน

**สมมุติฐาน** หุ่นยนต์สามารถเคลื่อนที่ผ่านทางลาดชัน ทางขรุขระและทางต่างระดับได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

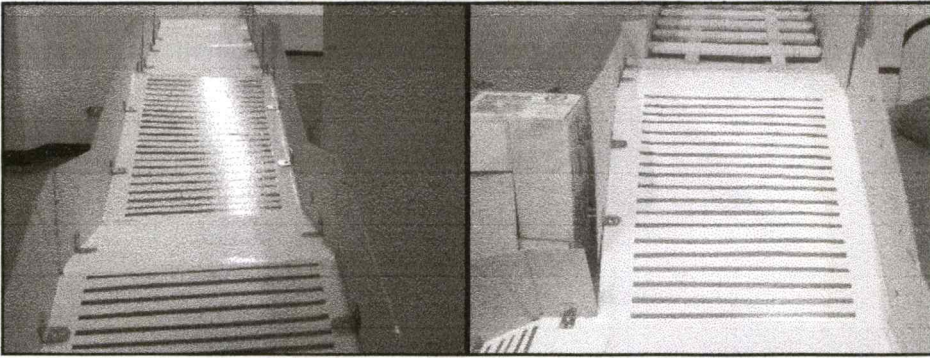
**ตารางที่ 4.6** แสดงการทดลองและผลการทดลองการเคลื่อนที่ของหุ่นยนต์ในสภาพแวดล้อมจำลอง

วิธีการทดลอง	ผลการทดลอง
<p>1.ทำการเปิด โปรแกรมผังหุ่นยนต์ และสร้าง การเชื่อมต่อจากเครื่องควบคุมไปยัง หุ่นยนต์</p> <p>2.ทำการควบคุมหุ่นยนต์ตามเส้นทางจากรูป ที่ 4.23 โดยเริ่มเคลื่อนที่ตามเส้นประจาก จุดเริ่มต้น ไปยังจุดปลายทางจากนั้นจึง เคลื่อนที่กลับมายังจุดเริ่มต้น โดยระหว่าง การเคลื่อนที่จะต้องพบกับเส้นทางลาดชันดัง รูปที่ 4.24 ทางขรุขระและทางต่างระดับดัง รูปที่ 4.25</p> <p>หมายเหตุ : ขณะที่เคลื่อนที่ขึ้นลงทางลาด ชัน ทดลองปรับความเร็วการเคลื่อนที่ให้ สูงขึ้นและต่ำลง</p>	<p>หุ่นยนต์สามารถเคลื่อนที่ไปตาม เส้นทางที่ระบุไว้และสามารถกลับมา ยังจุดเริ่มต้นได้ โดยสามารถเคลื่อนที่ ผ่านทางลาดชัน ทางขรุขระและทาง ต่างระดับได้ดังรูป 4.26 และ 4.27 ตามลำดับ</p> <p>จากการทดลองปรับความเร็ว ขณะที่เคลื่อนที่ขึ้นลงทางลาดชันพบว่า ความเร็วระดับสูงเหมาะสมในการขึ้น ทางชันมากที่สุด ในขณะที่การ เคลื่อนที่ลงทางลาด ความเร็วระดับต่ำ มีความเหมาะสมมากที่สุด</p>

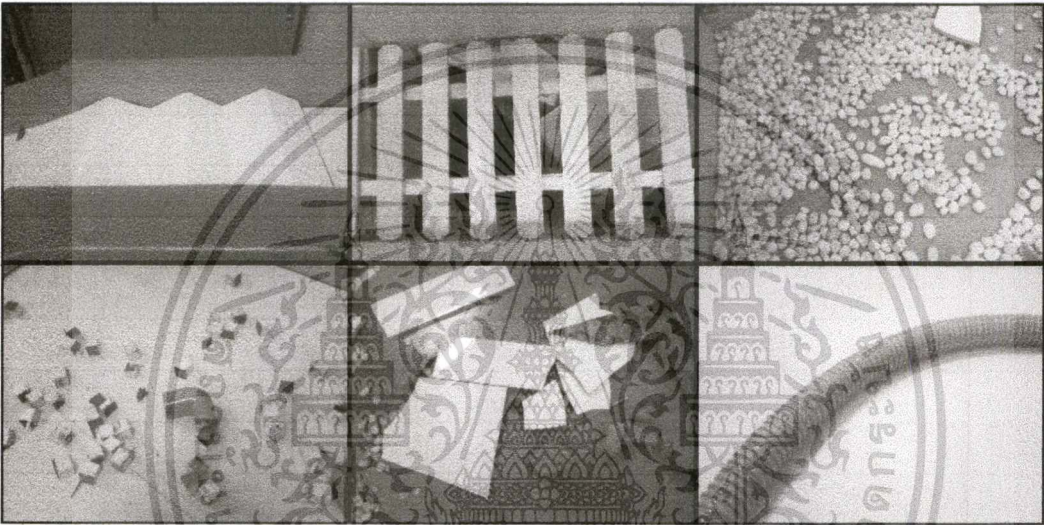


**รูปที่ 4.23** แสดงลักษณะของสนามทดสอบที่จำลองสภาพแวดล้อม และทิศทาง การเคลื่อนที่ของหุ่นยนต์

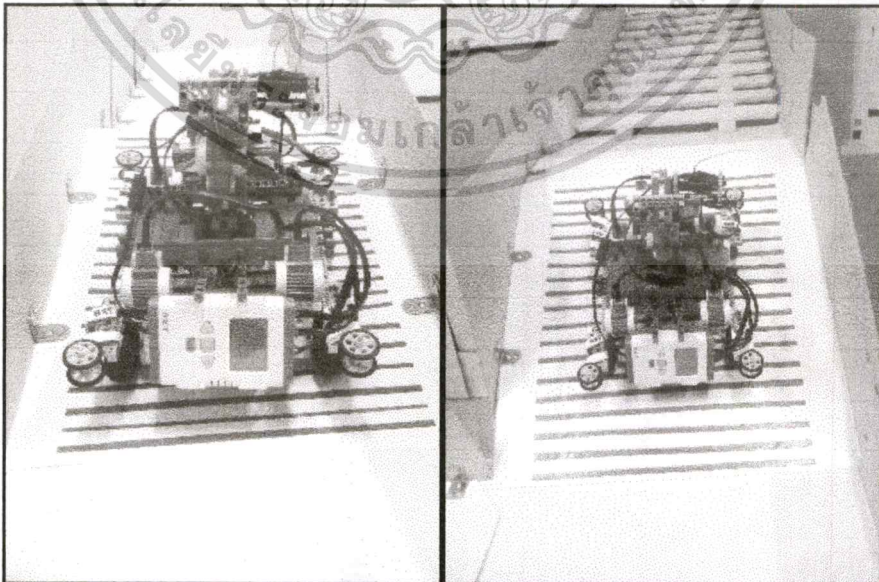
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.24 แสดงลักษณะของเส้นทางลาดชัน

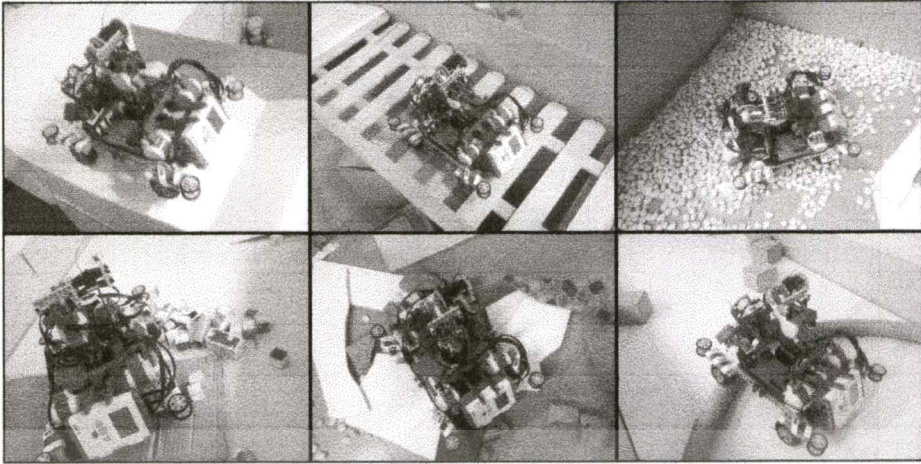


รูปที่ 4.25 แสดงลักษณะของเส้นทางขรุขระและทางต่างระดับ



รูปที่ 4.26 แสดงผลการขึ้นลงทางลาดชันของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.27 แสดงผลการเคลื่อนที่ผ่านเส้นทางขรุขระและทางต่างระดับ

### วิเคราะห์ผลการทดลอง

หุ่นยนต์สามารถเคลื่อนที่ไปตามเส้นทางที่ระบุไว้และสามารถกลับมายังจุดเริ่มต้นได้ โดยสามารถเคลื่อนที่ผ่านทางลาดชัน ทางขรุขระและทางต่างระดับได้ โดยพบว่าในการเคลื่อนที่ขึ้นทางชัน การใช้ความเร็วการเคลื่อนที่ระดับสูงจะมีความเหมาะสมมากที่สุด เนื่องจากมีแรงขับเคลื่อนมากที่สุด ส่วนการเคลื่อนที่ลงทางลาด การใช้ความเร็วการเคลื่อนที่ระดับต่ำมีความเหมาะสมที่สุดเนื่องจากการเคลื่อนที่ลงด้วยความเร็วต่ำช่วยรักษาสมดุลในการเคลื่อนที่ของหุ่นยนต์ไม่ให้ไหลไปกับทางลาด

จากการทดลองพบว่าความสามารถในการยึดเกาะพื้นผิวของล้อหุ่นยนต์จะลดลงเมื่อต้องเคลื่อนที่ผ่านสภาพพื้นผิวที่มีฝุ่นหรือทราย ส่งผลให้ความสามารถในการเคลื่อนที่ขึ้นทางชันลดต่ำลง

## 4.5 การทดลองเพื่อศึกษาระยะเวลาการทำงานอย่างต่อเนื่องของหุ่นยนต์ช่วยเหลือผู้ประสพภัย

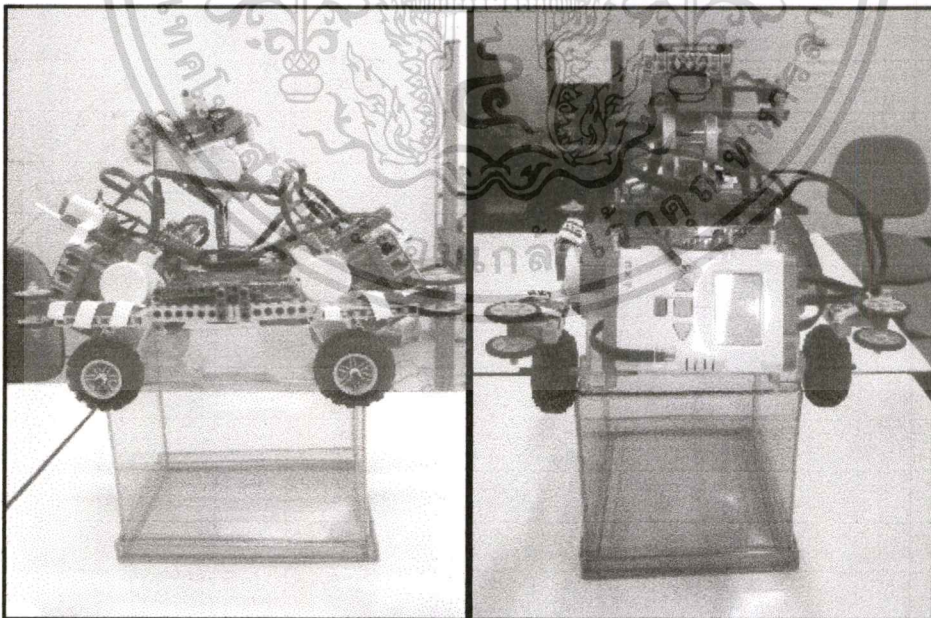
**วัตถุประสงค์** เพื่อศึกษาระยะเวลาการทำงานอย่างต่อเนื่องของหุ่นยนต์ช่วยเหลือผู้ประสพภัยโดยอาศัยการวิเคราะห์ข้อมูลค่าแรงดันไฟฟ้าจากส่วนบันทึกข้อมูลของโปรแกรมฝั่งเครื่องควบคุม

- สมมุติฐาน**
1. เอนเอ็กซ์ทีมอเตอร์สามารถทำงานต่อเนื่องได้นานกว่า 30 นาที
  2. หุ่นยนต์ช่วยเหลือผู้ประสพภัยสามารถทำงานอย่างต่อเนื่องได้มากกว่า 1 ชั่วโมง

หมายเหตุ : เนื่องจากชุดทดลองระบบฝังตัวเลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที่ไม่รองรับการตรวจสอบค่าพลังงาน โดยตรง ดังนั้นจึงต้องทำการศึกษาพฤติกรรมการลดลงของแรงดันไฟฟ้าของแบตเตอรี่เพื่อให้เป็นตัวชี้วัดระดับพลังงานที่เหลืออยู่

### วิธีการทดลอง

1. ทำการชาร์จแบตเตอรี่ของหุ่นยนต์จนเต็ม โดยสังเกตได้จากเอนเอ็กซ์ที่คอนโทรลเลอร์ดังรูป
2. หุ่นยนต์วางบนแท่นสำหรับทดสอบดังรูปที่ 4.28
3. ทำการเปิดโปรแกรมฝังหุ่นยนต์ และสร้างการเชื่อมต่อจากเครื่องควบคุมไปยังหุ่นยนต์
4. สั่งคำสั่งเดินหน้าด้วยความเร็วระดับกลาง และรอจนกว่าจะครบ 50 นาทีแล้วจึงทำการยกเลิกการเชื่อมต่อ จากนั้นนำข้อมูลค่าแรงดันไฟฟ้าของแบตเตอรี่ของโปรแกรมฝังเครื่องควบคุมมาวิเคราะห์
5. ปิดการทำงานของหุ่นยนต์เป็นเวลา 20 นาที
6. ทำซ้ำข้อ 3 ถึง 5 แต่ข้อ 4 เปลี่ยนระยะเวลาเป็น 20 นาที ทำซ้ำไปจนกว่าแบตเตอรี่ของหุ่นยนต์จะหมด

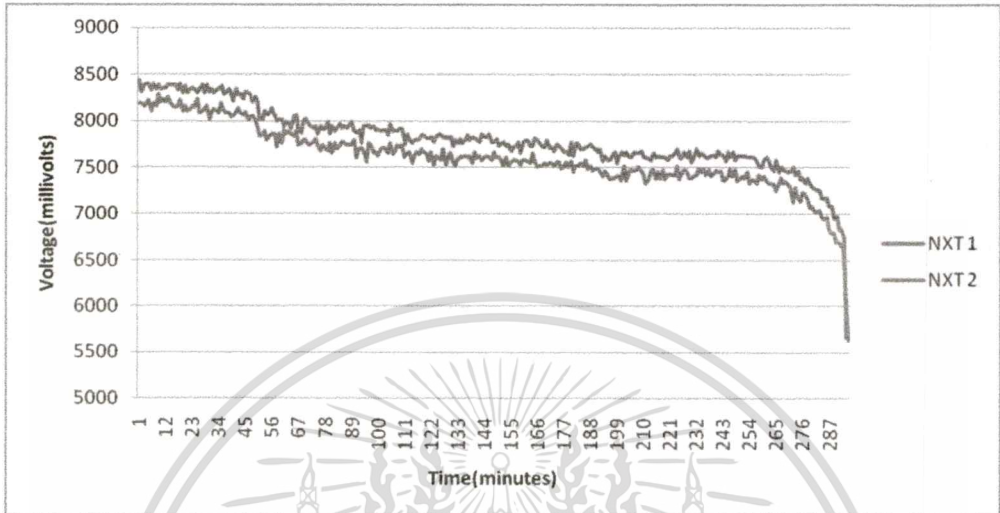


รูปที่ 4.28 แสดงลักษณะการทดสอบระยะเวลาการทำงานอย่างต่อเนื่องของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ผลการทดลอง

จากการทดลองเมื่อนำเอาข้อมูลค่าแรงดันไฟฟ้าที่เอนเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 1 และ 2 ส่งมายังเครื่องควบคุมมาเขียนเป็นกราฟได้ดังรูปที่ 4.29



รูปที่ 4.29 กราฟแสดงข้อมูลค่าแรงดันไฟฟ้าที่เอนเอ็กซ์ทีคอนโทรลเลอร์ตัวที่ 1 และ 2 ส่งกลับมายังเครื่องควบคุมในทุกๆ นาที

### วิเคราะห์ผลการทดลอง

จากผลการทดลองที่ได้จากการทดลองในขั้นตอนที่ 4 ทำให้ทราบว่าเอนเอ็กซ์ทีมอเตอร์สามารถทำงานต่อเนื่องได้นานกว่า 30 นาที และจากขั้นตอนถัดไปจนแบตเตอรี่หมดใช้เวลาถึงประมาณ 280 นาที ทำให้ทราบว่าหุ่นยนต์ช่วยเหลือผู้ประสพภัยสามารถทำงานอย่างต่อเนื่องได้มากกว่า 1 ชั่วโมง

จากกราฟดังรูปที่ 4.29 ทำให้ทราบถึงพฤติกรรมการลดลงของแรงดันไฟฟ้าของแบตเตอรี่ก่อนที่พลังงานจะหมดซึ่งสามารถนำไปเป็นตัวชี้วัดระดับพลังงานที่เหลืออยู่ของแบตเตอรี่ได้ดังตารางที่ 4.7

ตารางที่ 4.7 แสดงรูปแบบการแสดงผลกับระดับแรงดันไฟฟ้า

รูปแบบการแสดงผล				
ระดับแรงดันไฟฟ้า	มาก (มากกว่า 8000 มิลลิโวลต์)	ปานกลาง (อยู่ในช่วง 7300 ถึง 8000 มิลลิโวลต์)	น้อย (น้อยกว่า 7300 มิลลิโวลต์)	ว่างเปล่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.6 การทดลองเพื่อศึกษาพฤติกรรมการตัดขาดของสัญญาณบลูทูธเมื่อหุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุม

**วัตถุประสงค์** ศึกษาพฤติกรรมการตัดขาดของสัญญาณเมื่อหุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุม

**สมมุติฐาน** เมื่อหุ่นยนต์เคลื่อนที่ห่างออกไป ระดับสัญญาณบลูทูธที่วัดได้ที่หุ่นยนต์ควรจะมีแนวโน้มที่ต่ำลง

ตารางที่ 4.8 แสดงการทดลองเพื่อศึกษาพฤติกรรมการตัดขาดของสัญญาณบลูทูธเมื่อหุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุม

วิธีการทดลอง	ผลการทดลอง
1. ทำการเชื่อมต่อระหว่างเครื่องควบคุมและหุ่นยนต์ จากนั้นทำการตั้งงานให้หุ่นยนต์เคลื่อนที่ไปข้างหน้าด้วยความเร็วการหมุนของมอเตอร์ 700 องศาต่อวินาที ระหว่างที่หุ่นยนต์เคลื่อนที่ไปข้างหน้าจะมีการส่งระดับสัญญาณและจำนวนองศาการหมุนของล้อที่ได้มีการหมุนไปในทุกๆ 500 มิลลิวินาที ค่าดังกล่าวจะถูกทำการบันทึกเพื่อใช้ในการวิเคราะห์ต่อไป	1. หุ่นยนต์มีการส่งระดับสัญญาณสัญญาณและจำนวนองศาการหมุนของล้อกลับมายังเครื่องควบคุม
2. เมื่อหุ่นยนต์เคลื่อนที่ไปข้างหน้าจนถึงระยะที่การเชื่อมต่อระหว่างเครื่องควบคุมและหุ่นยนต์ถูกตัดขาด หุ่นยนต์จะหยุดเคลื่อนที่ ส่วนเครื่องควบคุมจะทำการแจ้งเตือนการตัดขาดของการเชื่อมต่อ ทำการวัดและบันทึกระยะทางที่หุ่นยนต์เริ่มเคลื่อนที่จนหยุดการเคลื่อนที่ดังรูปที่ 4.30 นำค่าที่ได้จากการวัดจริงมาเปรียบเทียบกับค่าได้จากการบันทึกไว้ที่เครื่องควบคุม	2. หุ่นยนต์หยุดการเคลื่อนที่ระยะทาง 36.5 เมตร จากการวัดจริง

## ตารางที่ 4.8 (ต่อ)

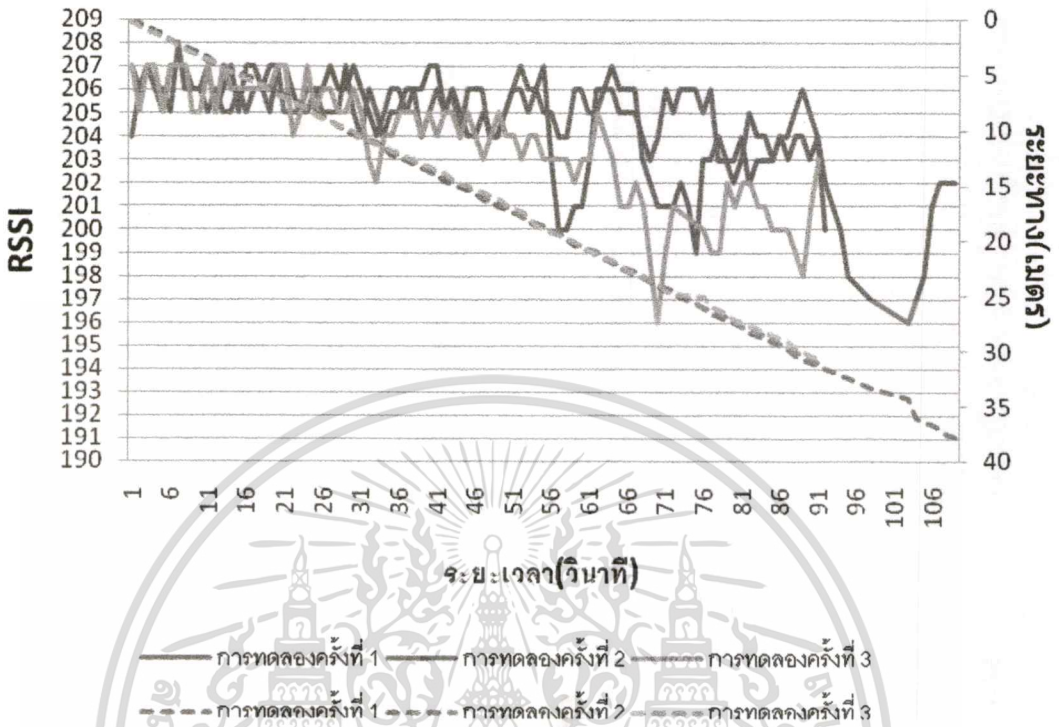
วิธีการทดลอง	ผลการทดลอง
3. ทำการทดลองขั้นตอนที่ 1 และ 2 ซ้ำอีก 2 ครั้ง	3.ผลที่ได้คล้ายเคียงกับการทดลองครั้งแรก โดยการทดลองครั้งที่ 2 ระยะทางที่วัดได้จริงคือ 33.5 เมตร และการทดลองครั้งที่ 2 ระยะทางที่วัดได้จริงคือ 32.6 เมตร



รูปที่ 4.30 แสดงการวัดและบันทึกระยะทางที่หุ่นยนต์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนำเอาค่าระดับสัญญาณมาเขียนเป็นกราฟเทียบกับระยะเวลาทุกๆ 1 วินาทีดังรูปที่ 4.31



รูปที่ 4.31 กราฟแสดงความสัมพันธ์ระหว่างค่าระดับสัญญาณบลูทูธและระยะทางที่ หุ่นยนต์เคลื่อนที่ห่างออกไปจากเครื่องควบคุม

#### วิเคราะห์ผลการทดลอง

จากผลการทดลองที่ได้พบว่าหลังจากหุ่นยนต์เคลื่อนที่ห่างออกไปประมาณ 15 เมตรค่าระดับสัญญาณมีการแกว่งค่อนข้างมาก และพบว่าที่จุดที่มีการตัดขาดของสัญญาณนั้น ค่าระดับสัญญาณ ไม่มีความแน่นอน

การตรวจจับการตัดขาดของสัญญาณเมื่อหุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุม พบว่าเมื่อหุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุม บลูทูธชิพใช้ระยะเวลานานถึงประมาณ 20 วินาทีในการตรวจจับข้อผิดพลาดจากการตัดขาดของสัญญาณ และแอนเ็็กซ์ที่คอนโทรลเลอร์แต่ละตัวยังใช้เวลาในส่วนนี้ไม่เท่ากันอีกด้วย

สรุปได้ว่าเมื่อหุ่นยนต์เคลื่อนที่ห่างออกไป ระดับสัญญาณบลูทูธที่วัดได้ที่หุ่นยนต์จะมีแนวโน้มที่ต่ำลง แต่ไม่สามารถระบุได้แน่ชัดว่าที่ระดับสัญญาณเท่าใดสัญญาณจึงจะตัดขาด

#### 4.7 การทดลองกลไกการป้องกันหุ่นยนต์เคลื่อนที่ออกนอกระยะเวลาควบคุม

**วัตถุประสงค์** เพื่อทดลองกลไกการป้องกันหุ่นยนต์เคลื่อนที่ออกนอกระยะเวลาควบคุม

**สมมุติฐาน** เมื่อหุ่นยนต์ทำการตรวจสอบระดับสัญญาณบลูทูธแล้วพบว่ามีค่าน้อยกว่าระดับสัญญาณที่เหมาะสม ซึ่งในที่นี้กำหนดไว้ที่ระดับ 203 หุ่นยนต์จะทำการหยุดการเคลื่อนที่และทำการแจ้งเตือนไปยังฝั่งเครื่องควบคุม เพื่อป้องกันไม่ให้ผู้ควบคุมทำการควบคุมหุ่นยนต์จนออกนอกระยะเวลาควบคุม

ตารางที่ 4.9 แสดงการทดลองกลไกการป้องกันหุ่นยนต์เคลื่อนที่ออกนอกระยะเวลาควบคุม

วิธีการทดลอง	ผลการทดลอง
ทำการเชื่อมต่อระหว่างเครื่องควบคุมและหุ่นยนต์ จากนั้นทำการสั่งงานให้หุ่นยนต์เคลื่อนที่ไปข้างหน้า สังเกตระบบเมื่อระดับสัญญาณลดลงถึงระดับ 203	กลไกป้องกันหุ่นยนต์เคลื่อนที่ออกนอกระยะเวลาควบคุมทำงานเมื่อระดับสัญญาณลดลงถึงระดับ 203

#### วิเคราะห์ผลการทดลอง

จากผลการทดลองพบว่าเมื่อหุ่นยนต์เคลื่อนที่ห่างออกไปจากเครื่องควบคุมจนวัดระดับสัญญาณได้ที่ระดับ 203 กลไกการป้องกันจะเริ่มทำงาน โดยที่เอนเอ็กซ์ทีคอนโทรลเลอร์ที่ตรวจพบระดับสัญญาณก่อนจะหยุดการเคลื่อนที่ของมอเตอร์แล้วส่งรหัสเตือน(Risky bits) ไปยังเครื่องควบคุม จากนั้นเครื่องควบคุมจะทำการแจ้งเตือนไปยังเอนเอ็กซ์ทีคอนโทรลเลอร์อีกตัว เพื่อสั่งให้หยุดการเคลื่อนที่ ในขณะเดียวกันทางฝั่งเครื่องควบคุมจะปิดการทำงานของเซฟโหมคโดยอัตโนมัติเพื่อให้ผู้ควบคุมสามารถควบคุมหุ่นยนต์ออกจากบริเวณนั้นได้

## บทที่ 5

# สรุปผลและข้อเสนอแนะ

### 5.1 สรุปผลโครงการ

จากการศึกษาวิธีการออกแบบและพัฒนาระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสบภัย เป็นการศึกษาและพัฒนาแอปพลิเคชันระบบฝังตัว โดยประยุกต์ใช้ระบบเชิงวัตถุในการพัฒนาระบบฝังตัว มีการวิเคราะห์และออกแบบในส่วนระบบควบคุมการเคลื่อนที่ ระบบควบคุมการหมุนของกล้องวีดีโอ ระบบตรวจสอบระดับสัญญาณบลูทูธและพลังงานของหุ่นยนต์ รวมทั้งออกแบบและพัฒนาโครงสร้างของหุ่นยนต์ช่วยเหลือผู้ประสบภัย เพื่อให้สามารถปฏิบัติงานได้จริงในสภาพแวดล้อมจำลองที่สร้างขึ้น จากการทดสอบระบบและสรุปผลที่ได้แสดงให้เห็นว่าระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสบภัยทำหน้าที่ในการรับคำสั่งจากผู้ควบคุมที่ฝังเครื่องควบคุมแล้วทำการส่งคำสั่งไปยังฝังหุ่นยนต์ในรูปแบบคำสั่งสร้างขึ้น เมื่อฝังหุ่นยนต์ได้รับคำสั่งจะทำการตีความและปฏิบัติงานตามคำสั่งที่ได้รับ จากนั้นจะทำการตอบกลับผลการทำงานไปยังฝังเครื่องควบคุมเพื่อให้ผู้ควบคุมทราบถึงสถานะการทำงานปัจจุบัน นอกจากนี้ในขณะที่ทำการควบคุมผู้ควบคุมสามารถทราบได้ถึงระดับสัญญาณบลูทูธและพลังงานของหุ่นยนต์ที่เหลืออยู่เนื่องจากหุ่นยนต์จะทำการส่งข้อมูลดังกล่าวกลับมายังฝังเครื่องควบคุมตลอดเวลา จากการทำงานในทุกส่วนทำให้เห็นได้ว่าระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสบภัยสามารถใช้งานได้จริงในสภาพแวดล้อมที่จำลองขึ้น

จากการศึกษาและทดลองทำโครงการนี้ทำให้สามารถเรียนรู้แนวทางในการพัฒนาระบบและประยุกต์ระบบเชิงวัตถุมาใช้กับงานระบบฝังตัวได้ สามารถนำเอาการวิเคราะห์และออกแบบพัฒนาระบบมาช่วยในการพัฒนางานด้านระบบฝังตัวให้มีรูปแบบและขั้นตอนในการพัฒนาอย่างเป็นระบบ ช่วยให้การพัฒนาตรงกับความต้องการที่วางไว้ ส่งผลให้งานด้านระบบฝังตัวมีความสมบูรณ์มากยิ่งขึ้น

### 5.2 ปัญหาและอุปสรรค

1. ปัญหาการสูญหายของข้อมูลในบางช่วงของการส่งค่าระดับสัญญาณบลูทูธกลับมายังฝังเครื่องควบคุมเนื่องจากเกิดข้อผิดพลาดจากการทำงานของบลูทูธชิพ จากการศึกษาการทำงานของบลูทูธชิพพบว่าการเรียกใช้เมธอด `GetSignalStrength()` บลูทูธชิพจะทำการสลับการทำงานจากโหมดการส่งข้อมูลไปยังโหมดควบคุม ซึ่งจากการสลับโหมดการทำงานขณะที่มีการส่งข้อมูลนี้อาจทำให้เกิดไอโอเอ็กเซพชัน (IO exception) ที่ฝังหุ่นยนต์ได้ ดังนั้นจึงต้องมี

การตรวจจับเอ็กซ์เรย์ และกำหนดช่วงเวลาในการส่งข้อมูลไว้ให้น้อยที่สุดที่ 500 มิลลิวินาที เพื่อลดผลกระทบที่เกิดจากการสูญหายของข้อมูล

2. ปัญหาด้านการเริ่มทำงานที่ไม่พร้อมกันระหว่างมอเตอร์คู่หน้าและคู่หลัง เนื่องจากการทำงานของหุ่นยนต์อาศัยการควบคุมมอเตอร์จากเอนเอ็กซ์ทีคอนโทรลเลอร์คนละตัว ซึ่งไม่สามารถทำการซิงโครไนซ์ข้อมูลซึ่งกันและกันได้ ส่งผลให้เกิดความคลาดเคลื่อนต่อองศาการหมุนของหุ่นยนต์

3. ปัญหาการศึกษาข้อมูลการทำงานเชิงลึกของฮาร์ดแวร์ภายในชุดทดลองระบบฝังตัวเลโก้ มายสตรอม รุ่นเอนเอ็กซ์ที เนื่องจากทางผู้ผลิตไม่เปิดเผยและให้ข้อมูลเกี่ยวกับการทำงานในเชิงลึก จึงทำให้การพัฒนาระบบค่อนข้างยาก

4. ปัญหาเกี่ยวกับเลจอส เอนเอ็กซ์เจทีไม่รองรับฟังก์ชันการตรวจสอบระดับพลังงานที่เหลืออยู่ของแบตเตอรี่ จากการศึกษาข้อมูลพบว่าระบบตรวจสอบระดับพลังงานภายในของเอนเอ็กซ์ทีคอนโทรลเลอร์ใช้แรงดันไฟฟ้าเป็นตัวชี้วัดระดับพลังงานที่เหลืออยู่

5. จากการทดลองพบว่าเมื่อหุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุม บลูทูธซีพีใช้เวลานานถึงประมาณ 20 วินาทีในการตรวจจับข้อผิดพลาดจากการตัดขาดของสัญญาณและเอ็กซ์ทีคอนโทรลเลอร์แต่ละตัวยังใช้เวลาในส่วนนี้ไม่เท่ากันอีกด้วย

### 5.3 ข้อเสนอแนะ

1. จากการศึกษาพฤติกรรมการตัดขาดของสัญญาณเมื่อหุ่นยนต์เคลื่อนที่ออกนอกระยะควบคุมพบว่าไม่สามารถสามารถระบุได้แน่ชัดว่าที่ระดับสัญญาณเท่าใดสัญญาณจึงจะตัดขาด อาจทำการแก้ไขโดยการทดลองด้วยจำนวนครั้งที่มากขึ้นเพื่อให้สามารถระบุได้แน่ชัดขึ้น

2. จากลักษณะล้อของหุ่นยนต์อาจมีข้อจำกัดในการเคลื่อนในสภาพแวดล้อมที่มีความต่างระดับหรือขรุขระมาก ดังนั้นหากสามารถเปลี่ยนไปใช้ล้อแบบตีนตะขาบ(Caterpillar wheel) จะสามารถช่วยให้ข้อจำกัดส่วนนี้ลดน้อยลง

3. จากลักษณะของระบบฝังตัวที่มีความเกี่ยวข้องกันระหว่างซอฟต์แวร์และฮาร์ดแวร์ ทำให้หากมีการปรับเปลี่ยน โครงสร้างทางฮาร์ดแวร์ในอนาคตอาจส่งผลต่อซอฟต์แวร์ที่ทำงานอยู่ได้

4. จากการทดลองเพื่อศึกษาระยะเวลาการทำงานอย่างต่อเนื่องของหุ่นยนต์ช่วยเหลือผู้ประสาบภัยเป็นลักษณะการทดลองมอเตอร์แบบไม่มีโหลด(Motor No-Load Test) ซึ่งเวลาที่ได้จากทดลองอาจแตกต่างจากเวลาที่ได้จากการเคลื่อนที่จริงเนื่องจากในการเคลื่อนที่จริงมอเตอร์ต้องรับน้ำหนักจากส่วนประกอบต่างๆของหุ่นยนต์ ทำให้มอเตอร์ต้องใช้พลังงานที่มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

วรเศรษฐ สุวรรณิก. 2551. **Java GUI using NetBeans**. กรุงเทพฯ:วรรณิก.

Andrew Davison. 2006. **Building a Game Pad Controller with JInput**. [Online]

Available: <http://fivedots.coe.psu.ac.th/~ad/jg2/ch11/index.html>

John, Satzinger W. et al. 2006. **Systems Analysis & Design in a Changing World**.

Canada: Thomson Course Technology.

Juan Antonio Brena Mora. 2008. **Develop leJOS programs step by step**. [Online]

Available: [http://www.juanantonio.info/jab\\_cms.php?id=146](http://www.juanantonio.info/jab_cms.php?id=146)

Juan Antonio Brena Mora. 2008. **Multithreading with Java leJOS**. [Online]

Available: [http://www.juanantonio.info/jab\\_cms.php?id=146](http://www.juanantonio.info/jab_cms.php?id=146)

Paul Andrews. et al. 2006. **The leJOS NXJ Tutorial**. [Online]

Available: <http://lejos.sourceforge.net/nxt/nxj/tutorial/index.htm>





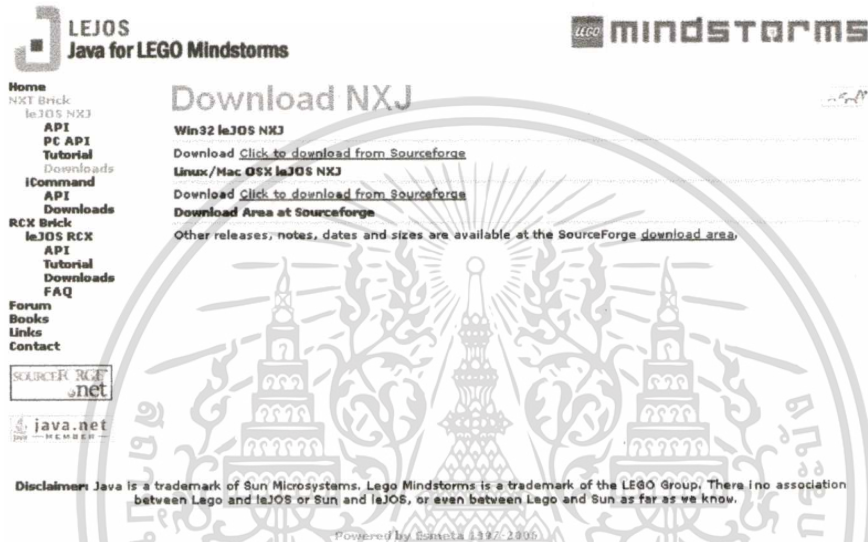
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

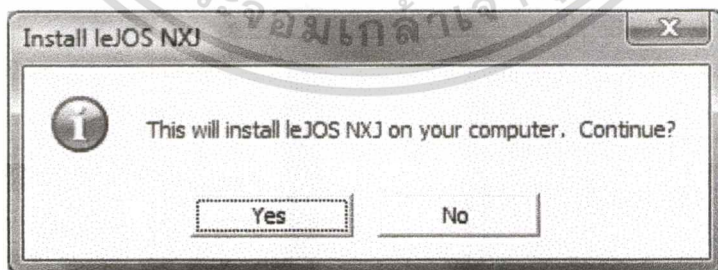
## ขั้นตอนการติดตั้ง LeJOS NXJ บน Windows XP

1. เข้าไปยังเว็บไซต์ <http://lejos.sourceforge.net/nxjdownloads.php> เพื่อทำการดาวน์โหลดโปรแกรมมาติดตั้งโดยไฟล์สำหรับติดตั้งจะมีชื่อว่า leJOS\_NXJ\_version-Setup.exe ดังรูปที่ ก.1



รูปที่ ก.1 แสดงหน้าจอดาวน์โหลดโปรแกรม leJOS NXJ

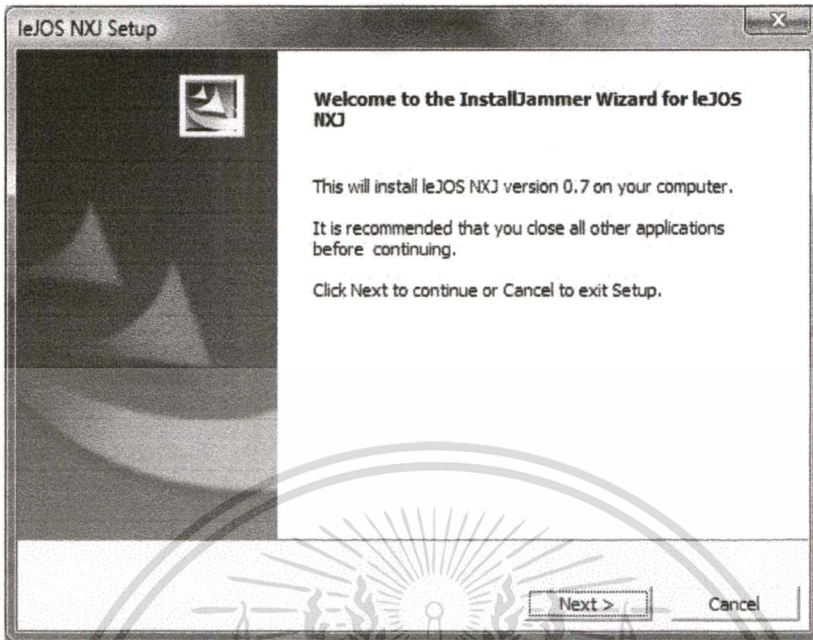
2. เมื่อทำการดาวน์โหลดเสร็จเรียบร้อย ให้การรันไฟล์ติดตั้ง แล้วคลิกปุ่ม Yes ดังรูปที่ ก.2



รูปที่ ก.2 แสดงไดออล็อกเมื่อดาวน์โหลดโปรแกรมเสร็จเรียบร้อยแล้ว

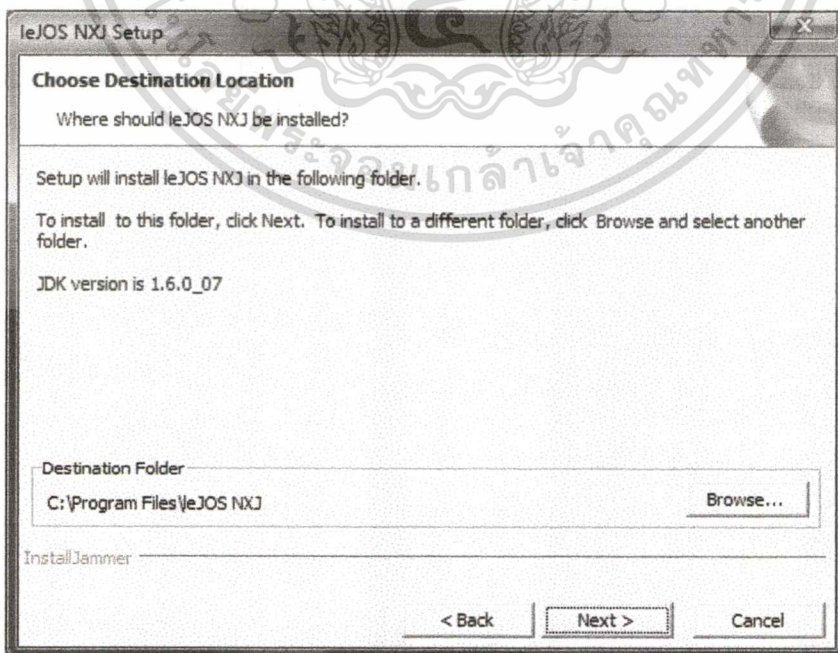
3. จากนั้นทำการคลิกปุ่ม Next เพื่อให้โปรแกรมทำการติดตั้งโปรแกรม LeJOS NXJ รวมทั้งปรับแต่งสภาพแวดล้อมการใช้งาน ดังรูปที่ ก.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.3 แสดงการติดตั้งโปรแกรม LeJOS NXJ

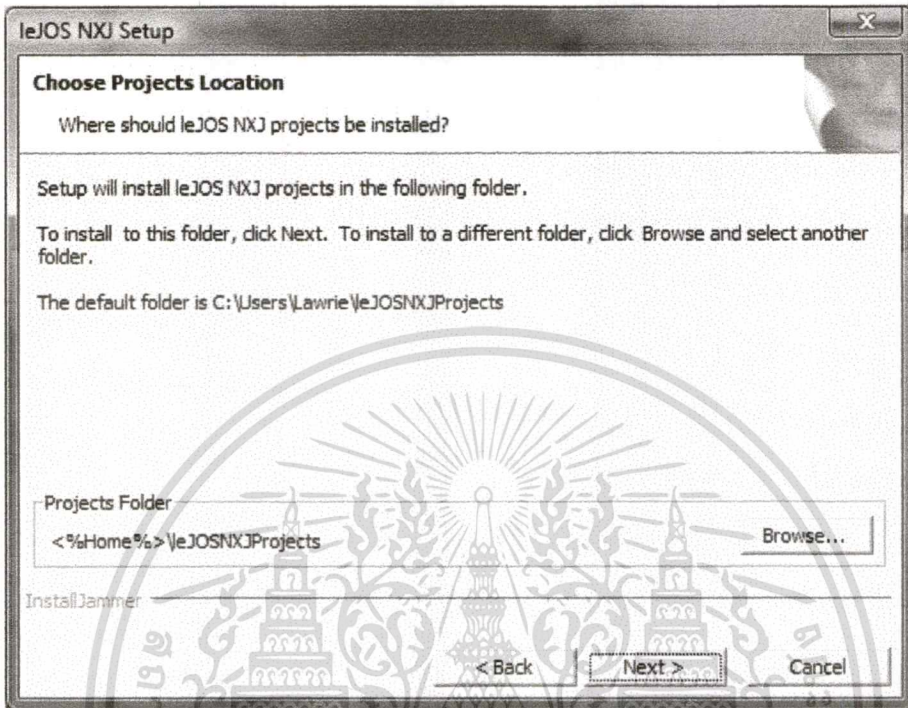
4. จากนั้นจะเป็นการกำหนดไดเรกทอรีหลัก(Root direstory) ที่จะทำการติดตั้งโปรแกรม แล้วคลิกปุ่ม Next ดังรูปที่ ก.4



รูปที่ ก.4 แสดงการกำหนดไดเรกทอรีหลักที่จะทำการติดตั้งโปรแกรม

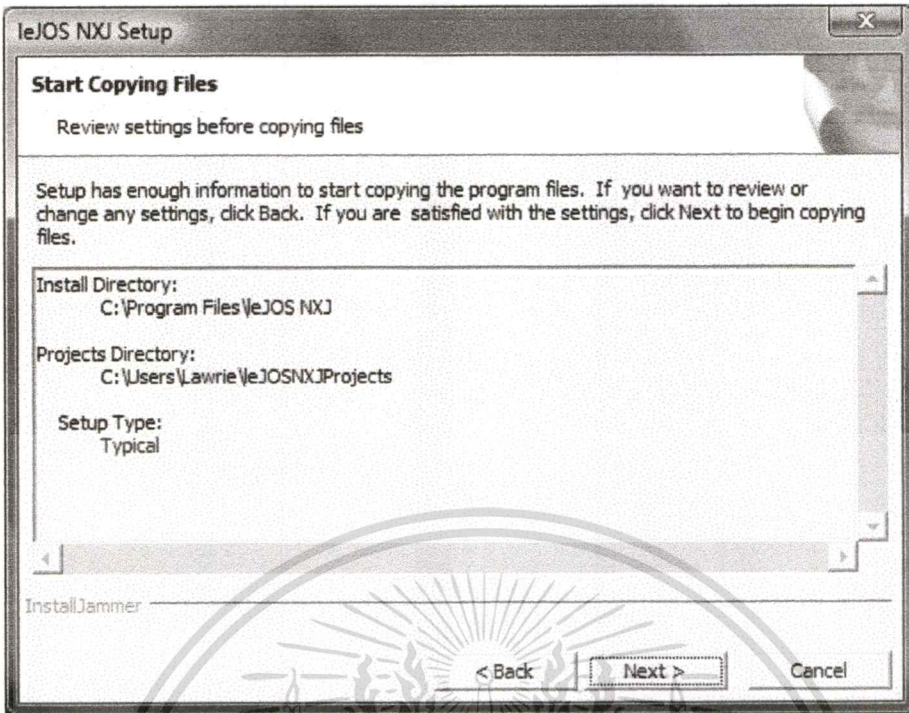
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. จากนั้นจะเป็นการกำหนดไดเรกทอรีที่จะทำการติดตั้งโปรเจกต์ตัวอย่างสำหรับเป็นแนวทางในการพัฒนาระบบ ดังรูปที่ ก.5



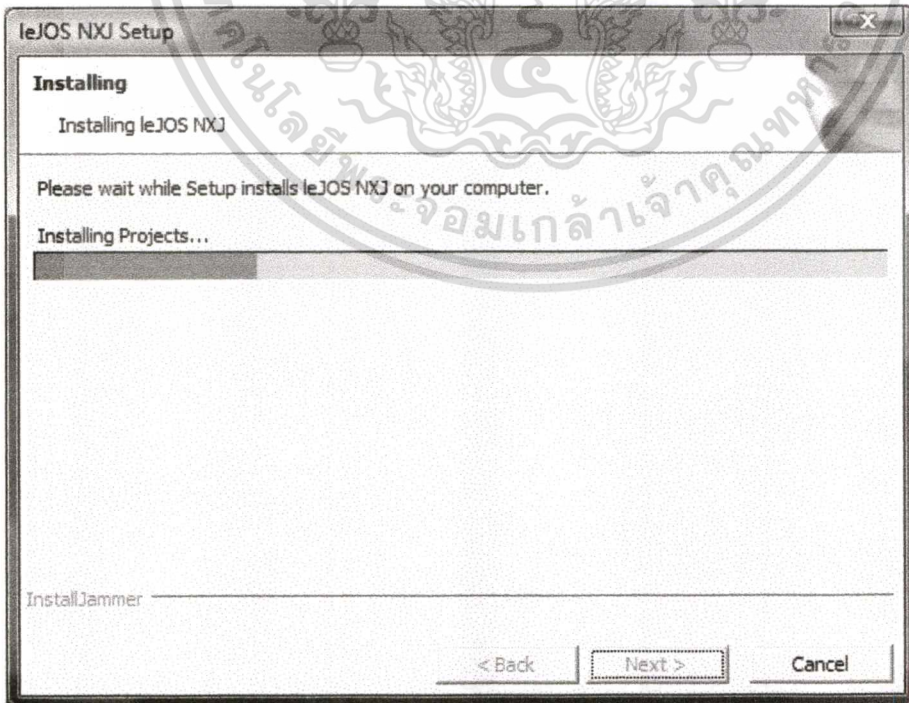
รูปที่ ก.5 แสดงการกำหนดไดเรกทอรีหลักที่จะทำการติดตั้งโปรแกรม

6. จากนั้นโปรแกรมติดตั้งจะทำการแสดงรายละเอียดการติดตั้งอีกครั้ง หากรายละเอียดถูกต้องให้คลิกปุ่ม Next ดังรูปที่ ก.6



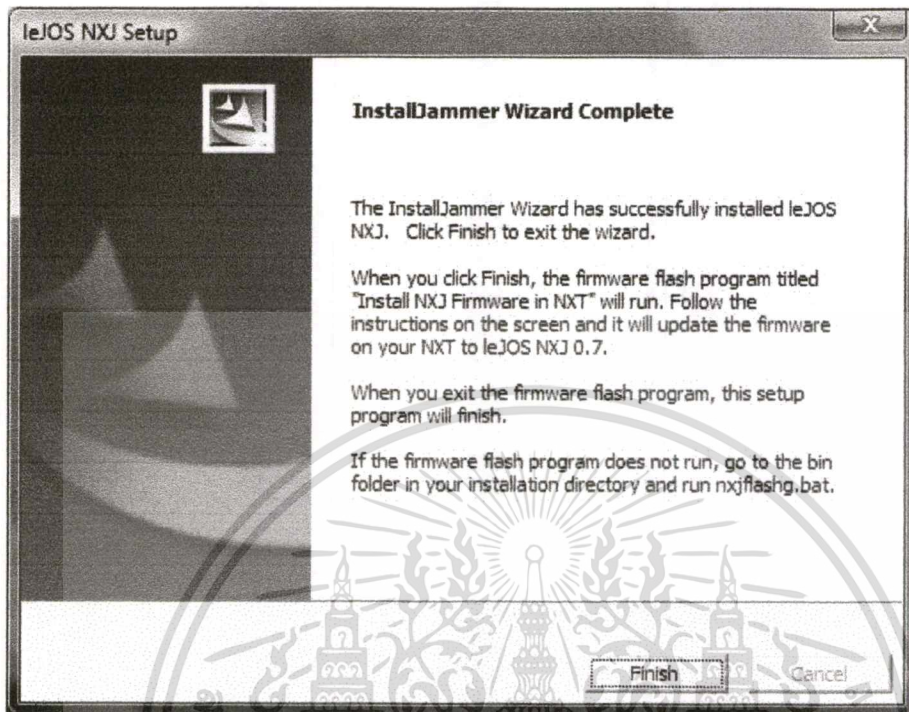
รูปที่ ก.6 แสดงรายละเอียดการติดตั้ง

7. โปรแกรมติดตั้งจะทำการติดตั้ง LeJOS NXJ ดังรูปที่ ก.7



เอกสารนี้เป็นเอกสารที่สงวนรูปที่ ก.7 แสดงหน้าจอกำลั้งติดตั้ง ติดตั้ง LeJOS NXJ หน้าไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. เมื่อการติดตั้งเสร็จสมบูรณ์โปรแกรมติดตั้งจะแสดงดังรูปที่ ก.8



รูปที่ ก.8 แสดงหน้าจอการติดตั้งเสร็จสมบูรณ์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ขั้นตอนการติดตั้งเฟิร์มแวร์ LeJOS NXJ

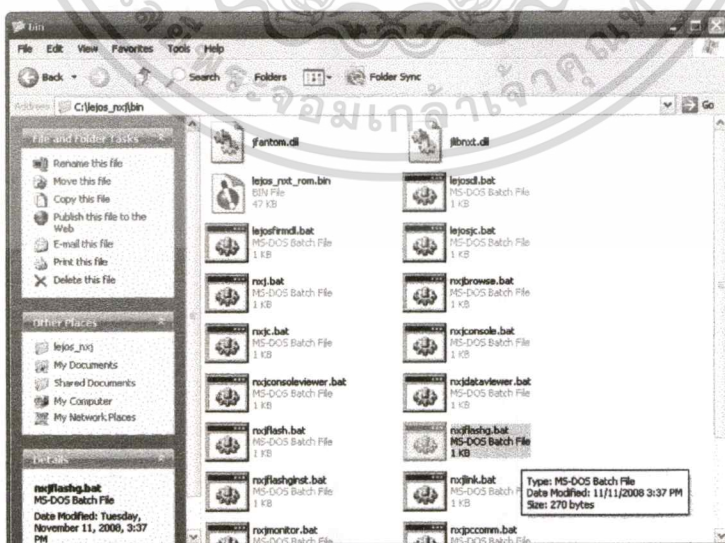
## บน LEGO MINDSTORMS NXT

1. หากเฟิร์มแวร์ที่ติดตั้งอยู่กับ LEGO MINDSTORMS NXT เป็นเฟิร์มแวร์ดั้งเดิมของเลโก้ จะต้องทำการรีเซตการทำงานก่อนทำการติดตั้งเฟิร์มแวร์ของ LeJOS NXJ โดยการกดปุ่มรีเซตดังรูปที่ ค้างไว้ 5 วินาที ดังรูปที่ ข.1



รูปที่ ข.1 แสดงการรีเซตการทำงานก่อนทำการติดตั้งเฟิร์มแวร์ของ LeJOS NXJ

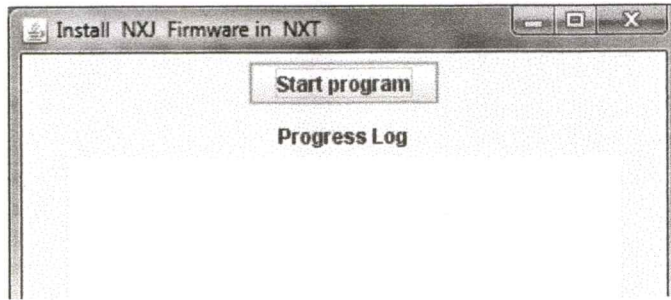
2. หลังจากนั้นทำการเข้าไปยังไดเรกทอรี `..\lejos_nxj\bin` แล้วทำการรันไฟล์ที่ชื่อ `nxjflashg.bat` เพื่อเปิดโปรแกรมสำหรับติดตั้งเฟิร์มแวร์ของ LeJOS NXJ ดังรูปที่ ข.2



รูปที่ ข.2 แสดงไดเรกทอรีของไฟล์ ชื่อ `nxjflashg.bat`

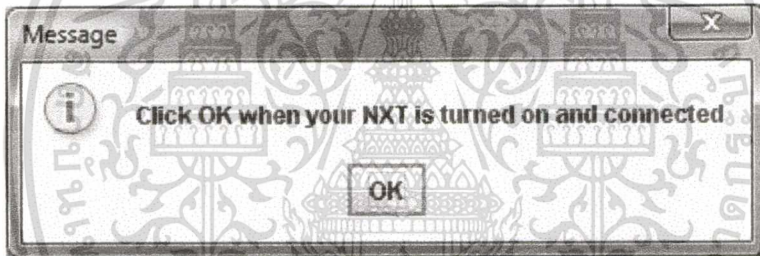
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการคลิก Start program เพื่อติดตั้งเฟิร์มแวร์ของ LeJOS NXJ ดังรูปที่ ข.3



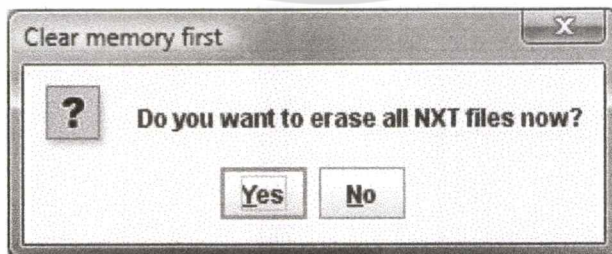
รูปที่ ข.3 แสดงหน้าจอการติดตั้งเฟิร์มแวร์ของ LeJOS NXJ

4. จากนั้นจะมีข้อความบอกรให้ทำการเชื่อมต่อ USB ระหว่างเอนเอ็กซ์ทีคอนโทรลเลอร์และเครื่องคอมพิวเตอร์ จากนั้นคลิก OK ดังรูปที่ ข.4

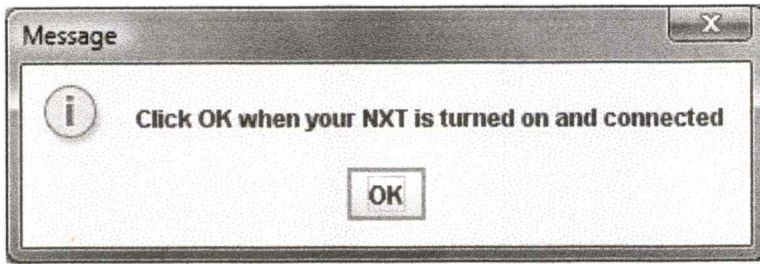


รูปที่ ข.4 แสดงกล่องข้อความบอกรให้ทำการเชื่อมต่อ USB ระหว่างเอนเอ็กซ์ทีคอนโทรลเลอร์และเครื่องคอมพิวเตอร์

5. จากนั้นจะมีข้อความเพื่อยืนยัน ดังรูปที่ ข.5 และรูปที่ ข.6 ให้ทำการคลิก Yes และ OK ตามลำดับ

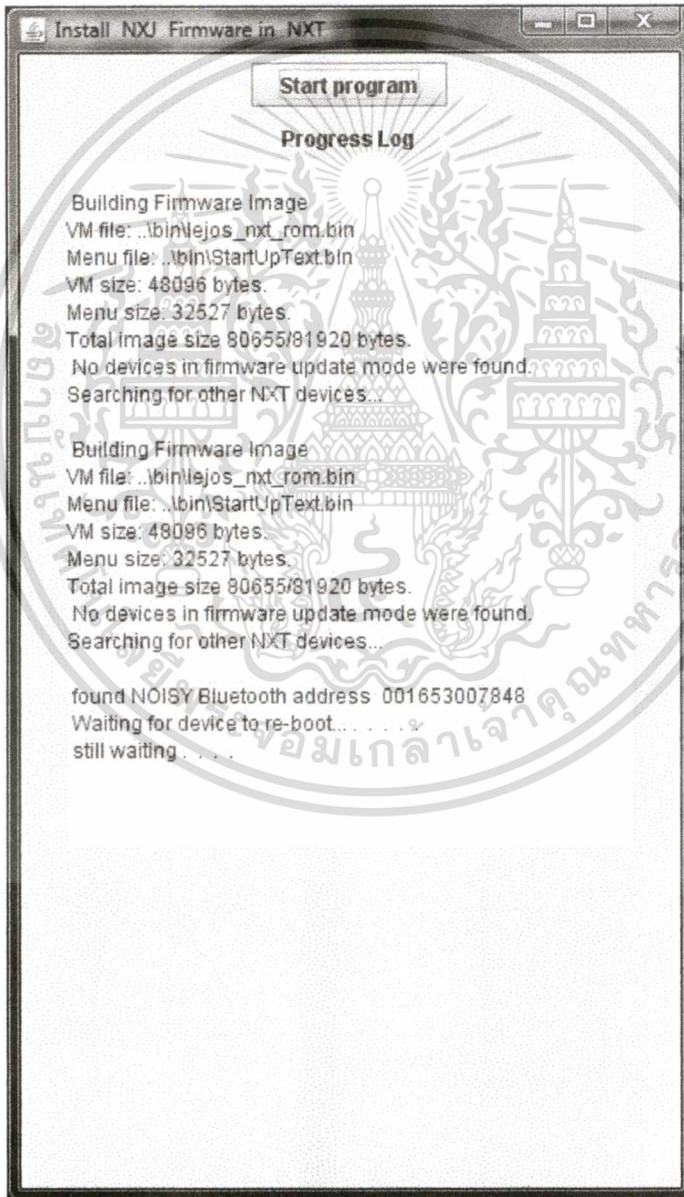


รูปที่ ข.5 แสดงข้อความยืนยัน



รูปที่ ข.6 แสดงข้อความเตือนให้เปิดและเชื่อมต่อ NXT

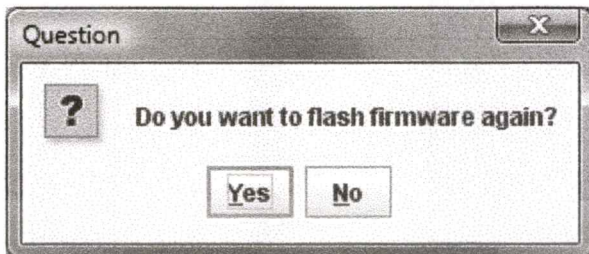
6. โปรแกรมจะเริ่มทำการติดตั้งเฟิร์มแวร์ของ LeJOS NXJ และแสดงกระบวนการ ดังรูปที่ ข.7



รูปที่ ข.7 แสดงการติดตั้งเฟิร์มแวร์ของ LeJOS NXJ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เมื่อทำการติดตั้งเฟิร์มแวร์ของ LeJOS NXJ เสร็จเรียบร้อยแล้ว จะมีข้อความถามว่าต้องการติดตั้งเฟิร์มแวร์ของ LeJOS NXJ ใหม่อีกครั้งหรือไม่ ให้ตอบ No หากไม่ต้องการหรือตอบ Yes หากต้องการติดตั้งเฟิร์มแวร์ของ LeJOS NXJ ใหม่ให้กับเอนเอ็กซ์ทีคอนโทรลเลอร์ตัวอื่น ดังรูปที่ ข.8



รูปที่ ข.8 แสดงการติดตั้งเฟิร์มแวร์ของ LeJOS NXJ เสร็จเรียบร้อยแล้ว



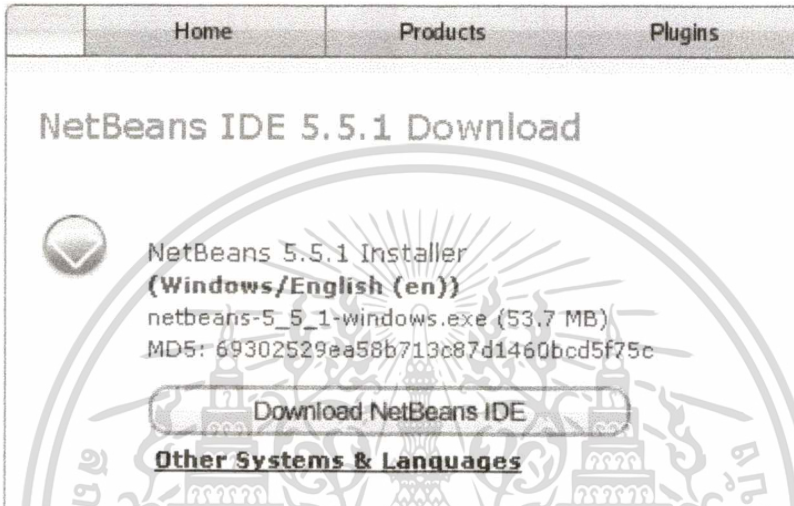


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ขั้นตอนการปรับแต่ง Netbeans เพื่อใช้งานร่วมกับ LeJOS NXJ

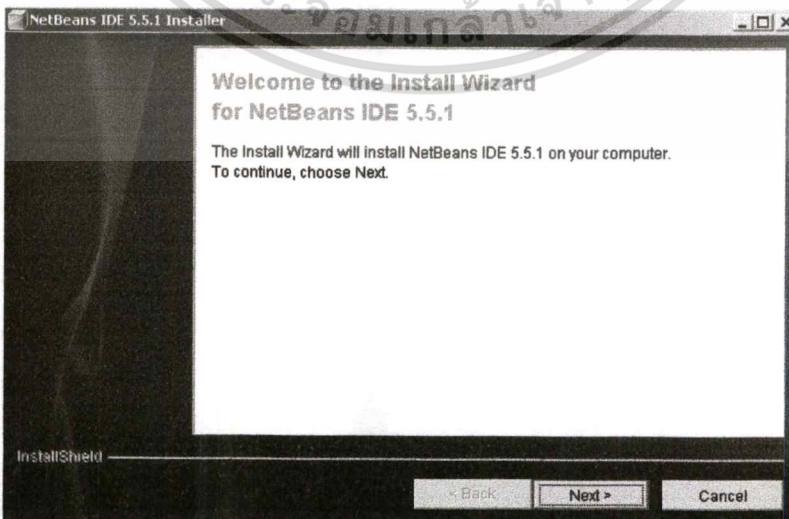
### 1. การติดตั้ง Netbeans

1.1 ดาวน์โหลดโปรแกรมได้จาก <http://www.netbeans.info/downloads/index.php> ดังรูปที่ ค.1 ให้ click ที่ปุ่ม Download NetBeans IDE เพื่อเลือก mirror site ในการ download



รูปที่ ค.1 แสดงหน้าจอในการดาวน์โหลดโปรแกรม netbeans

1.2 เมื่อ download เสร็จแล้ว จะได้ไฟล์ netbeans-5\_5\_1-windows.exe มา จากนั้นติดตั้งโปรแกรม NetBeans 5.5.1 โดย double click ที่ไฟล์ netbeans-5\_5\_1-windows.exe โปรแกรม Installer แสดงข้อความต้อนรับ ดังรูปที่ ค.2 click ที่ Next เพื่อไปหน้าต่อไป

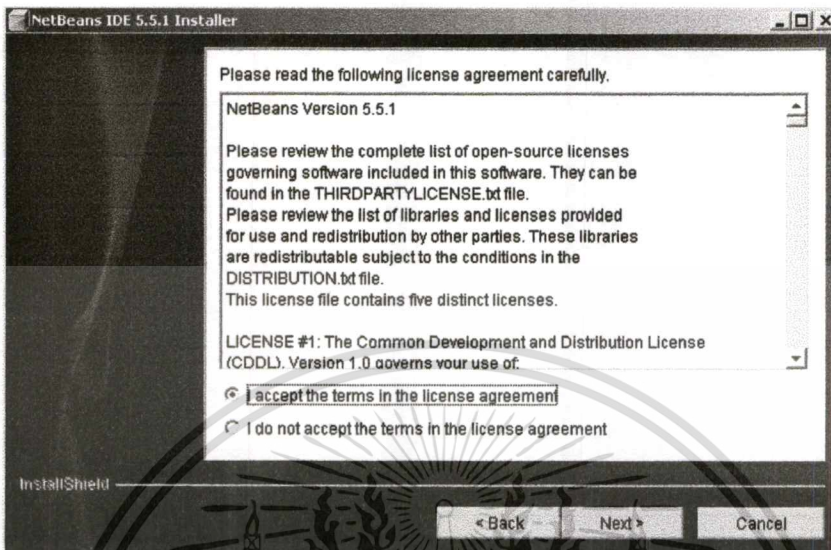


รูปที่ ค.2 แสดงข้อความต้อนรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

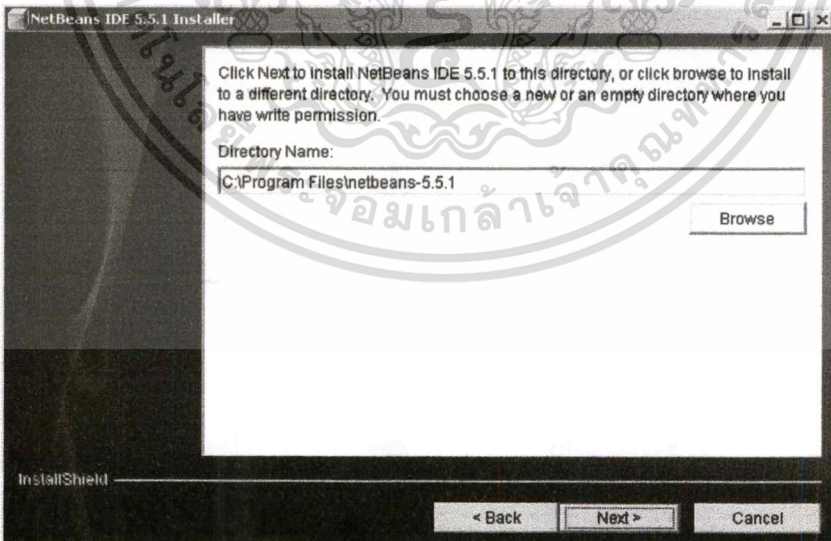
1.3 เลือก I accept the terms in the license agreement แล้ว click Next เพื่อไปหน้าต่อไป ดัง

รูปที่ ค.3



รูปที่ ค.3 แสดงหน้าจอแสดงเงื่อนไขของโปรแกรม

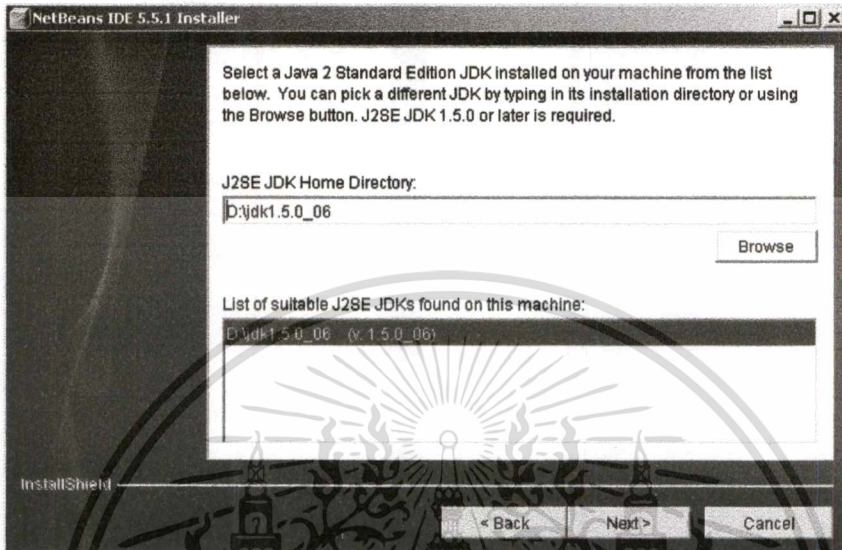
1.4 โปรแกรม Installer จะให้เราระบุ path ที่เราต้องการติดตั้ง NetBeans ให้เราระบุ path ที่เราต้องการ ดังรูปที่ ค.4 จากนั้น click ที่ Next เพื่อไปหน้าต่อไป



รูปที่ ค.4 แสดงหน้าจอให้ผู้ระบุ path ที่ต้องการติดตั้งโปรแกรม

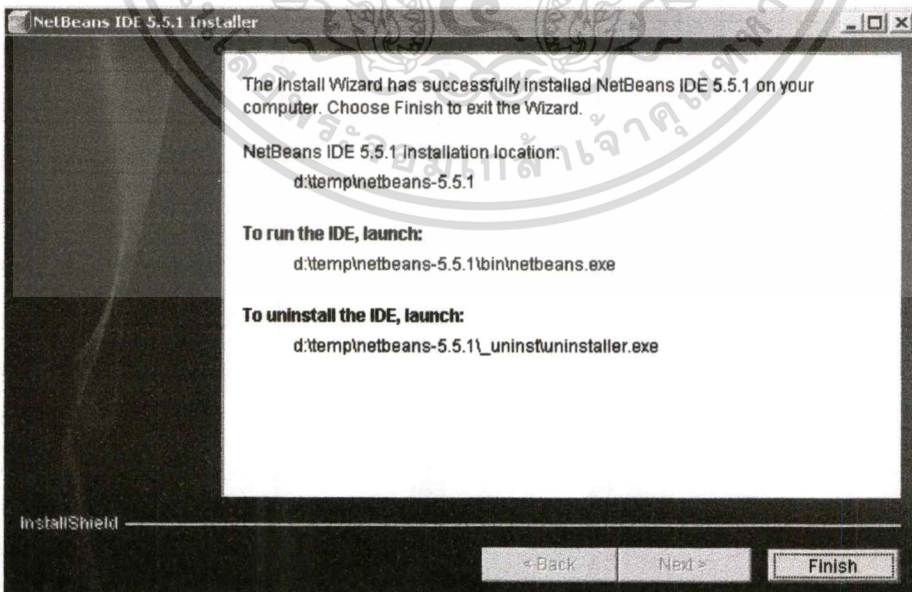
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 โปรแกรม Installer จะมองหา JDK 5 หรือ 6 ที่มีอยู่ในเครื่องและแสดง path ของ JDK ที่พบ ในรูปโปรแกรม Installer พบ JDK version 1.5.0\_06 ในกรณีที่เราลง JDK ไว้หลาย version เราสามารถเลือกใช้ version ที่เราต้องการได้ โดยเลือก Browse click ที่ Next เพื่อไปหน้าต่อไป



รูปที่ ค.5 แสดง path ของ JDK ที่พบในเครื่องคอมพิวเตอร์

1.6 โปรแกรม Installer เริ่มติดตั้งโปรแกรม NetBeans ลงใน path ที่ระบุ เมื่อติดตั้งโปรแกรมเสร็จเรียบร้อยแล้ว click ที่ Finish เพื่อจบโปรแกรม ดังรูปที่ ค.6

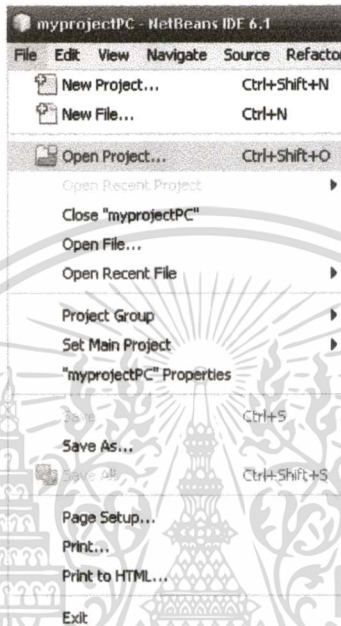


รูปที่ ค.6 แสดงหน้าจอเมื่อการติดตั้งโปรแกรมเสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

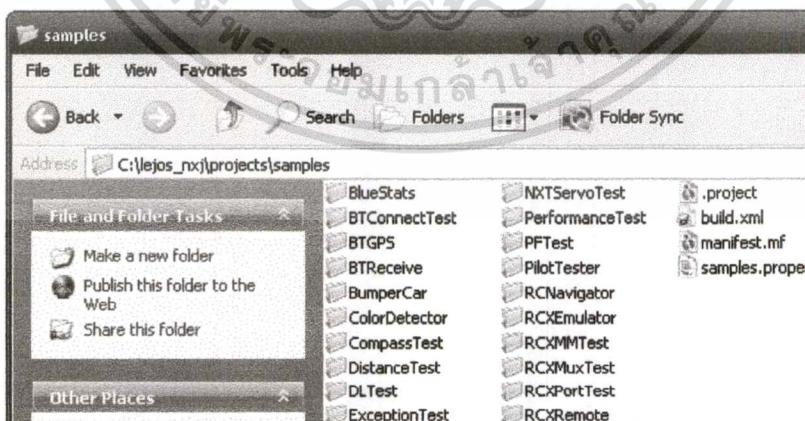
## 2. การสร้างโปรเจกเพื่อพัฒนาระบบฝังหุ่นยนต์

2.1 ทำการเปิด leJOS NXJ samples project โดยเลือกที่เมนู File >Open project ดังรูปที่ ค.7 แล้วทำการ Browse ไปยังตำแหน่งใดเรีกทอที่ติดตั้งตัวอย่างโปรเจกไว้ โดยโปรเจกตัวอย่างที่พัฒนาในฝังหุ่นยนต์จะอยู่ในโฟลเดอร์ samples



รูปที่ ค.7 แสดงการเปิด leJOS NXJ samples project

2.2 เมื่อทำการเปิดโปรเจกตัวอย่างจะพบว่ามีโปรเจกตัวอย่าง ดังรูปที่ ค.8

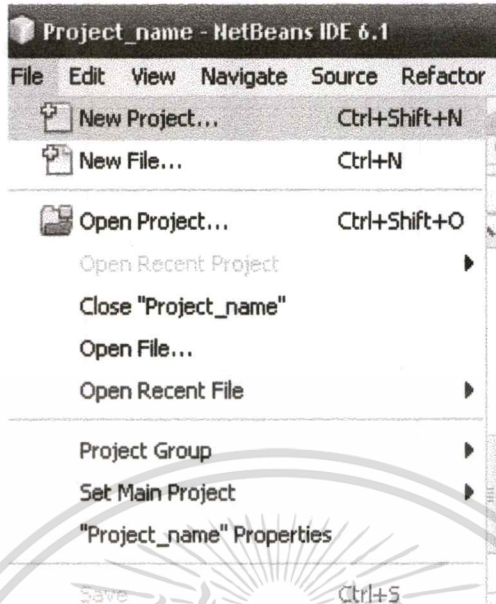


รูปที่ ค.8 แสดงการ Browse ไปยังตำแหน่งใดเรีกทอที่ติดตั้งตัวอย่างโปรเจก

2.3 จากนั้นทำการสร้างโปรเจกใหม่โดยการไปที่ file > New Project >Java>Java

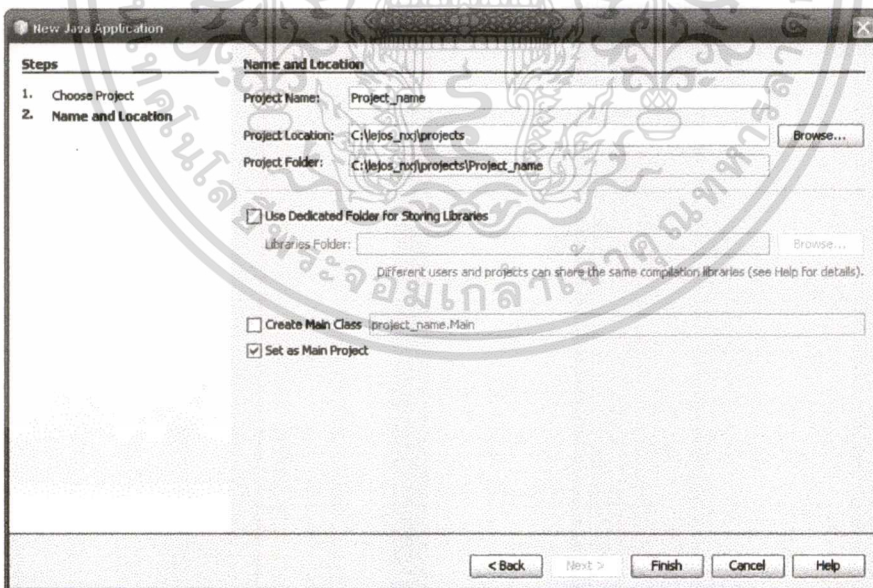
Applicaton แล้วคลิกปุ่ม Next ดังรูปที่ ค.9

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ค.9 แสดงการสร้างโปรเจกใหม่

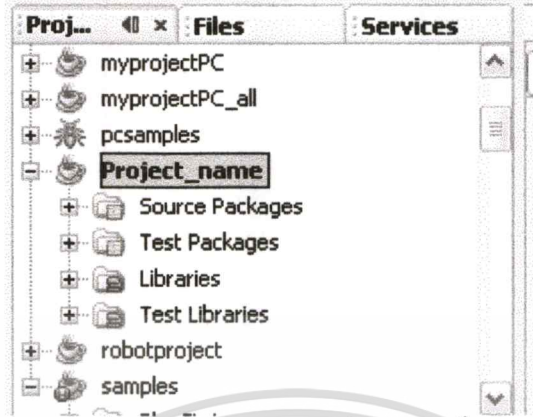
2.4 จากนั้นให้ทำการตั้งชื่อโปรเจก ในที่นี้ใช้ชื่อว่า “Project\_name” กำหนดไคเร็กทอรีที่เก็บโปรเจก และกำหนดตัวเลือกต่างๆดังรูป แล้วคลิกปุ่ม Finish ดังรูปที่ ค.10



รูปที่ ค.10 แสดงการการตั้งชื่อโปรเจก

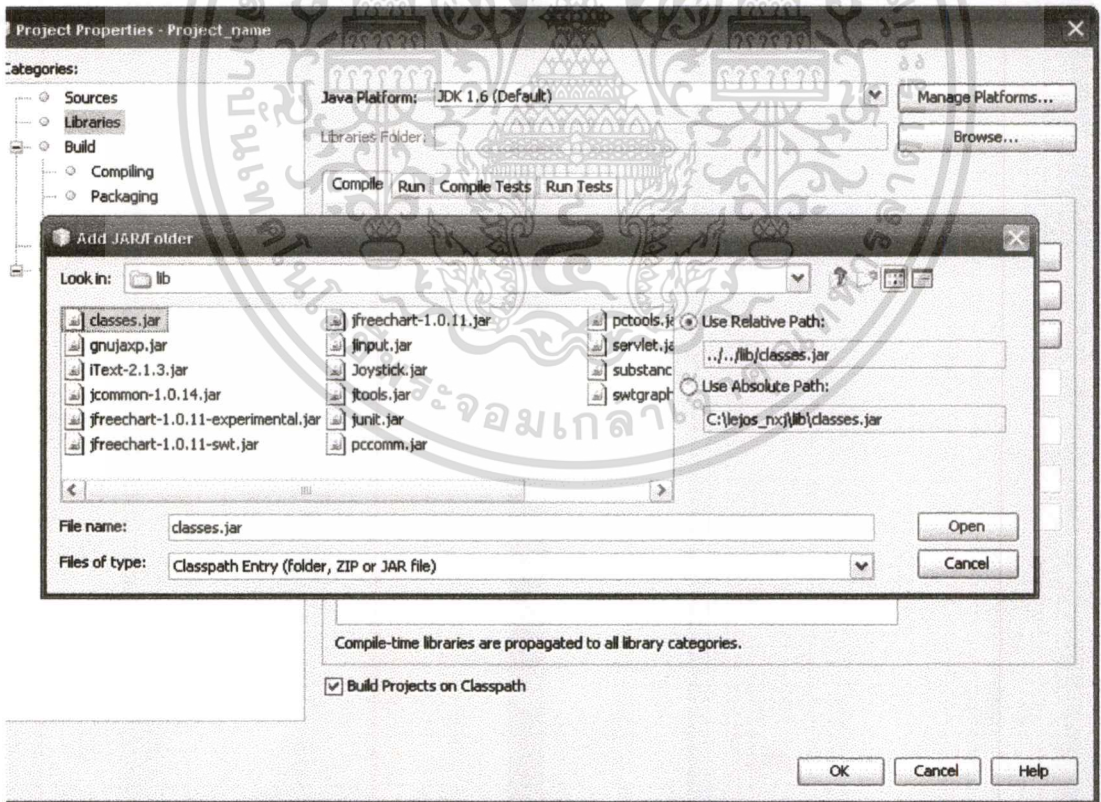
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5 จากนั้นจะปรากฏโปรเจกต์ที่สร้างไว้ในแท็บ Project ดังรูปที่ ค.11



รูปที่ ค.11 แสดงโปรเจกต์ที่ถูกสร้างขึ้น

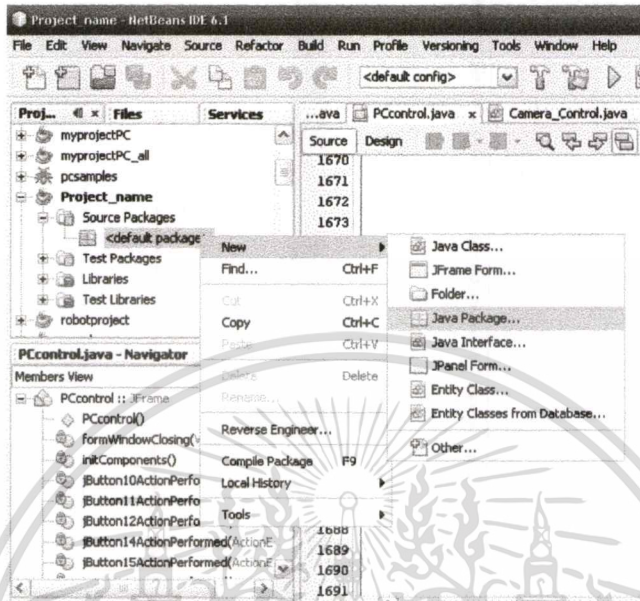
2.6 จากนั้นให้ทำการคลิกขวาที่โปรเจกต์ที่สร้างไว้ เลือก Properties > Libraries > Add jar/folder แล้วทำการเพิ่ม classes.jar ซึ่งอยู่ใน ../lejos\_nxj/lib ดังรูปที่ ค.12



รูปที่ ค.12 แสดงการเพิ่มคลาสลงในโปรเจกต์

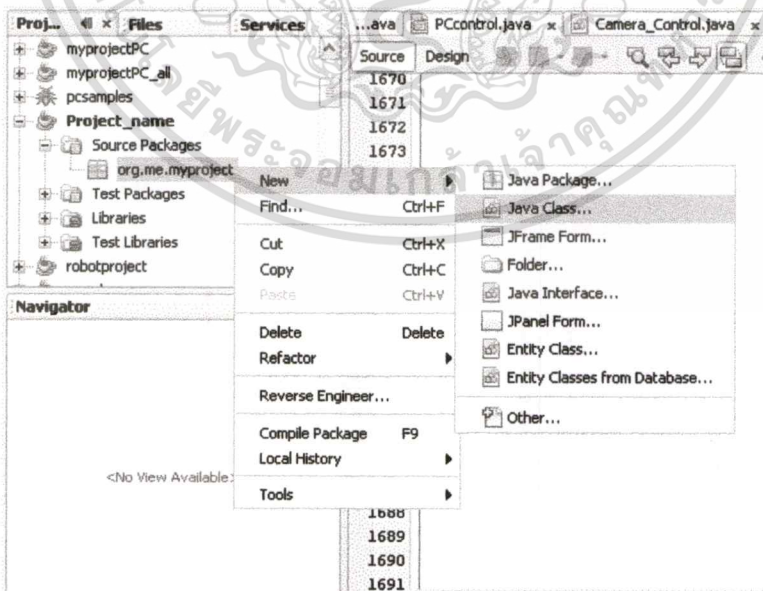
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 ทำการสร้าง Java Package โดยคลิกขวาที่โฟลเดอร์ Source Packages >New> Java Package และทำการใส่ชื่อของ Java Package ในที่นี้ใช้ชื่อว่า “org.me.myproject” ดังรูปที่ ค.13



รูปที่ ค.13 แสดงการสร้างจาวาแพคเกจ

2.8 จากนั้นทำการสร้าง Java main class โดยการคลิกขวาที่ Package org.me.myproject >New>Java class และทำการใส่ชื่อของ Java class ในที่นี้ใช้ชื่อว่า “HelloWorld” ดังรูปที่ ค.14



รูปที่ ค.14 แสดงการสร้างคลาสในโปรเจก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 จากนั้นทำการแก้ไขโค้ดภายในไฟล์ HelloWorld ให้เป็นดังรูปที่ 15 เพื่อทดสอบการทำงานของโปรแกรม

```
package org.me.myproject;

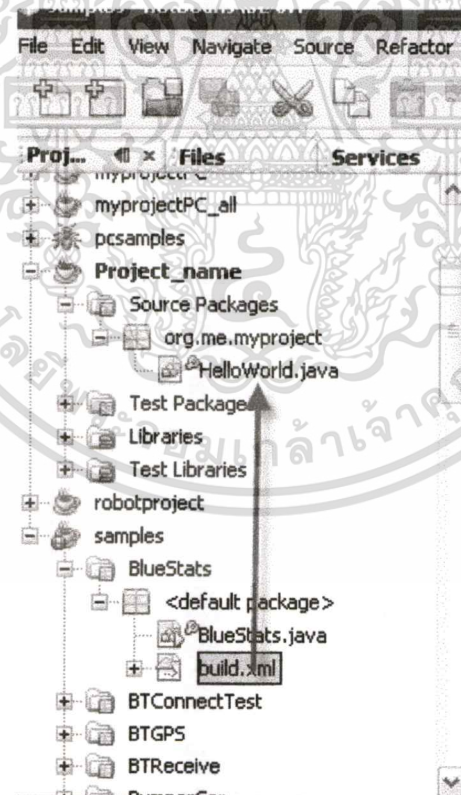
import lejos.nxt.*;

public class HelloWorld {

    public static void main(String[] args) {
        System.out.println("Hello World");
        Button.waitForPress();
    }
}
```

รูปที่ ค.15 แสดงการแก้ไขโค้ดภายในไฟล์ HelloWorld

2.10 จากนั้นทำการ copy ไฟล์ชื่อ build.xml จากตัวอย่างโปรเจกต์ที่ได้เปิดไว้ในข้อ 1.1 และ 1.2 ไปไว้ยังโฟลเดอร์ Source Packages ดังรูปที่ ค.16



รูปที่ ค.16 แสดงการ copy ไฟล์ชื่อ build.xml จากตัวอย่างโปรเจกต์ที่ได้เปิดไว้ ไปยังโฟลเดอร์ Source Packages

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.11 จากนั้นเปิดไฟล์ build.xml ขึ้นมาแล้วทำการแก้ไขดังนี้

- เปลี่ยนข้อความ `<basename property="program" file="."/>` เป็น `<property name="program" value="HelloWorld">`

- ทำการเพิ่ม property:

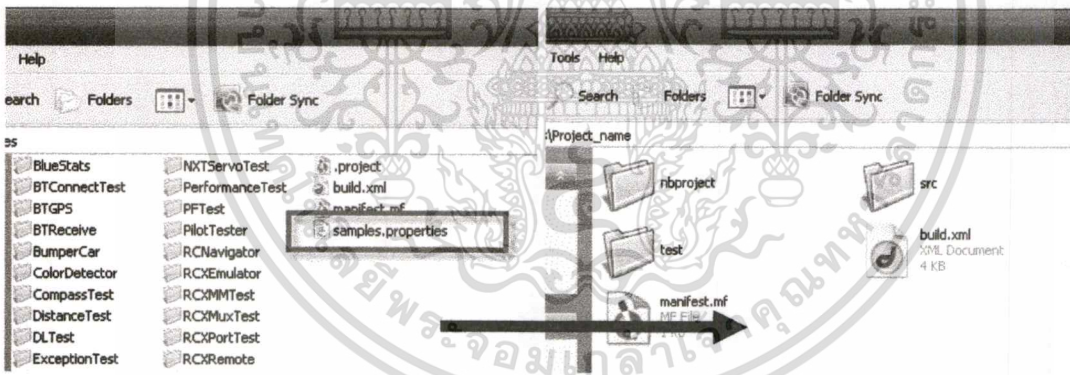
```
<property name="class" value="org.me.myproject.HelloWorld">
```

- จากนั้นในส่วน link target, เปลี่ยนบรรทัดที่มีข้อความ

```
<arg value="{program}"/> เป็น <arg value="{class}"/>
```

## 2.12 สุดท้ายทำการ copy ไฟล์ที่มีชื่อว่า sample.properties จากโฟลเดอร์

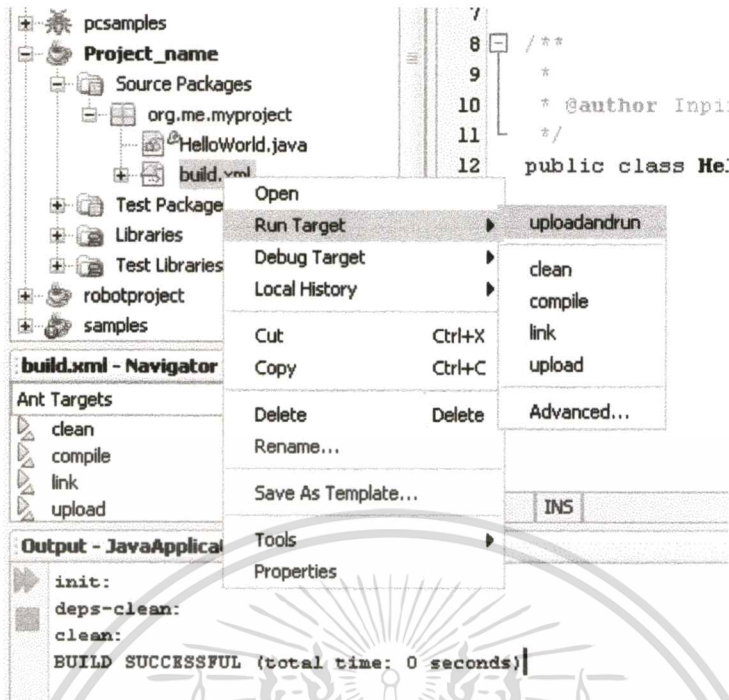
...\\ejos\_nxj\projects\samples ไปไว้ยังโฟลเดอร์โปรเจกที่กำลังพัฒนา ในที่นี้คือโฟลเดอร์ชื่อ Project\_name ดังรูปที่ ก.17



รูปที่ ก.17 แสดงการ copy ไฟล์ที่มีชื่อว่า sample.properties จากโฟลเดอร์ ...\\ejos\_nxj\

projects\samples ไปไว้ยังโฟลเดอร์โปรเจกที่กำลังพัฒนา

2.13 เมื่อต้องการอัปโหลดไฟล์ไปยังหุ่นยนต์ให้ทำการเชื่อมต่อระหว่างหุ่นยนต์และเครื่องคอมพิวเตอร์ด้วยสาย USB หรือ บลูทูธ จากนั้นให้ทำการเปิดหุ่นยนต์ แล้วทำการคลิกขวาที่ `build.xml`>`Run Target`>`uploadandrun` จากนั้นหาก `upload` สำเร็จจะมีข้อความขึ้นที่แท็บ Output ว่า “BUILD SUCCESSFUL” ดังรูปที่ ก.18



รูปที่ ค.18 แสดงต้องการอัปเดตไฟล์ไปยังหุ่นยนต์

### 3. การสร้างโปรเจกต์เพื่อพัฒนาระบบฝังเครื่องควบคุม(PC)

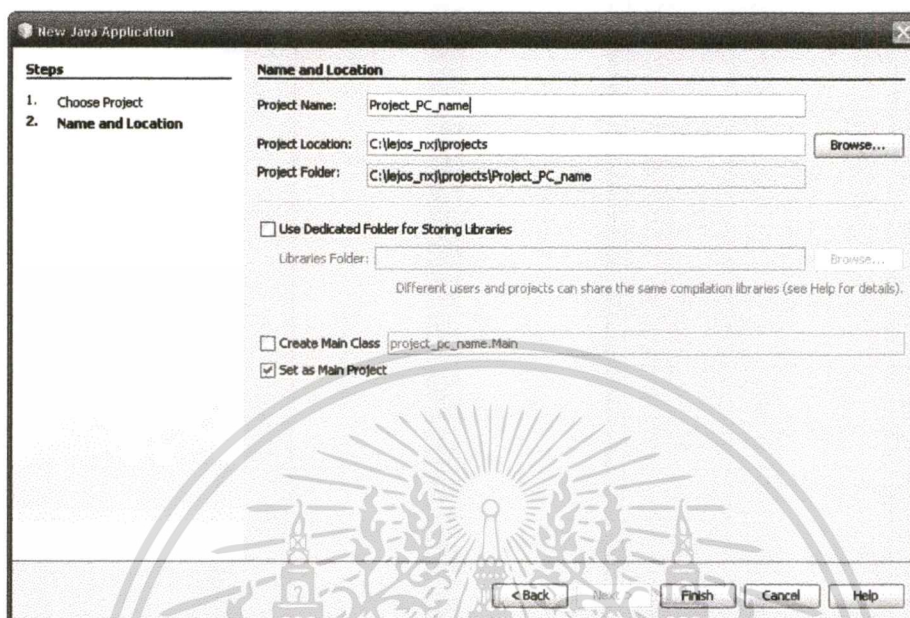
3.1 ทำการสร้างโปรเจกต์ใหม่โดยการไปที่ file > New Project > Java > Java Application แล้วคลิกปุ่ม Next ดังรูปที่ ค.19



รูปที่ ค.19 แสดงการสร้างโปรเจกต์ใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 จากนั้นให้ทำการใส่ชื่อโปรเจก ในที่นี้ใช้ชื่อว่า “Project\_PC\_name” กำหนดไดเรกทอรีที่เก็บโปรเจก และกำหนดตัวเลือกต่างๆดังรูป แล้วคลิกปุ่ม Finish ดังรูปที่ ค.20



รูปที่ ค.20 แสดงการตั้งชื่อโปรเจกและไดเรกทอรีที่เก็บโปรเจก

3.3 จากนั้นให้ทำการคลิกขวาที่โปรเจกที่สร้างไว้ เลือก Properties > Libraries > Add jar/folder แล้วทำการเพิ่มไลบรารีดังนี้

- pccomm.jar และ pctools.jar ซึ่งอยู่ใน ../jejos\_nxj/lib
- bluecove.jar ซึ่งอยู่ใน ../jejos\_nxj\3rdparty\lib

เมื่อทำการเพิ่มไลบรารีทั้งหมดนี้แล้ว ภายในโปรเจกที่สร้างขึ้นนั้นสามารถสร้าง java class ได้ตามปกติ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ขั้นตอนการติดตั้งและปรับแต่งเพื่อใช้งานจอยสติ๊ก

ในการติดต่อกับจอยสติ๊กได้อาศัย The Lightweight Java Game Library (LWJGL) และ Joy stick for java ในการทำให้ระบบที่พัฒนาขึ้นสามารถควบคุมการทำงานของหุ่นยนต์ผ่านจอยสติ๊กได้ ซึ่งทดลองใช้กับจอยสติ๊กรุ่น PU301 ซึ่งผลิตโดยบริษัท DILONG ดังรูปที่ ซึ่งสามารถดาวน์โหลด driver ได้ที่ <http://www.zidoupload.com/f-ai95hClr>



รูปที่ ง.1 แสดงจอยสติ๊กรุ่น PU301 ซึ่งผลิตโดยบริษัท DILONG

### 1. ขั้นตอนการติดตั้ง The Lightweight Java Game Library

1.1 ทำการดาวน์โหลด The Lightweight Java Game Library จาก

[http://sourceforge.net/project/showfiles.php?group\\_id=58488&package\\_id=54362&release\\_id=642951](http://sourceforge.net/project/showfiles.php?group_id=58488&package_id=54362&release_id=642951) โดย ไฟล์มีชื่อว่า lwjgl-2.0.1.zip ดังรูปที่ ง.2

SOURCEFORGE.NET [Log in](#) [Create account](#) [Com](#)

Lightweight Java Game Library [Summary](#) [Download](#) [More](#)



**You have selected the LWJGL 2.0.1 release.**

Please choose the file that best matches your architecture or operating system from the list of files contained in this release.

Options

Package	Release	Filename	Size	Architecture
<u>Official Releases</u>				
<b>Latest</b>	<b>LWJGL 2.0.1</b>	(2008-11-25 21:47)		
		<a href="#">lwjgl-2.0.1.zip</a>	2424888	Platform-Independent
		<a href="#">lwjgl_applet-2.0.1.zip</a>	3225960	Platform-Independent
		<a href="#">lwjgl-docs-2.0.1.zip</a>	1508963	Platform-Independent
		<a href="#">lwjgl-source-2.0.1.zip</a>	1437397	Platform-Independent
<b>Totals:</b>	<b>1</b>	<b>4</b>	<b>8597208</b>	

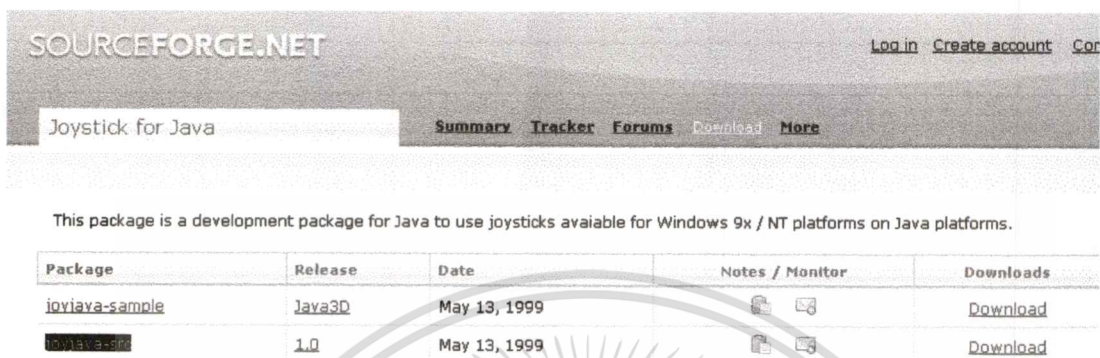
## รูปที่ ง.2 แสดงการดาวน์โหลด The Lightweight Java Game Library





- 1.2 เมื่อดาวน์โหลดสำเร็จให้ทำการแตกไฟล์ .zip ดังกล่าว
- 1.3 Netbeans ไปรเจกต์ที่ต้องการทำงานร่วมกับจอยสติคค์ต้องทำการเพิ่มไลบรารีทุกตัวที่อยู่ใน ...lwjgl-2.0.1\lwjgl-2.0.1\jar
- 1.4 นำไฟล์ที่อยู่ใน ...lwjgl-2.0.1\lwjgl-2.0.1\native\win32 ไปใส่ไว้ใน C:\Program Files\Java\jdk1.6.0\_10\bin
- 1.5 ทำการรีสตาร์ทคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2. ขั้นตอนการติดตั้ง Joy stick for java

2.1 ทำการดาวน์โหลด Joy stick for java จาก [http://sourceforge.net/project/showfiles.php?group\\_id=55526](http://sourceforge.net/project/showfiles.php?group_id=55526) โดยไฟล์มีชื่อว่า joyjava-src > joystick.zip ดังรูปที่ ๓.3



Package	Release	Date	Notes / Monitor	Downloads
<a href="#">joyjava-sample</a>	Java3D	May 13, 1999	 	<a href="#">Download</a>
<a href="#">joyjava-src</a>	1.0	May 13, 1999	 	<a href="#">Download</a>

### รูปที่ ๓.3 แสดงการดาวน์โหลด Joy stick for java

- 2.2 เมื่อดาวน์โหลดสำเร็จให้ทำการแตกไฟล์ .zip ดังกล่าว
- 2.3 Netbeans โปรเจกต์ที่ต้องการทำงานร่วมกับจอยสติคต้องการเพิ่มไลบรารีชื่อ Joystick.jar ที่อยู่ใน ...joystick-0-7\joystick
- 2.4 นำไฟล์ที่อยู่ใน joystick-0-7\joystick\lib ไปใส่ไว้ใน C:\Program Files\Java\jdk1.6.0\_10\jre\bin
- 2.5 ทำการรีสตาร์ทคอมพิวเตอร์

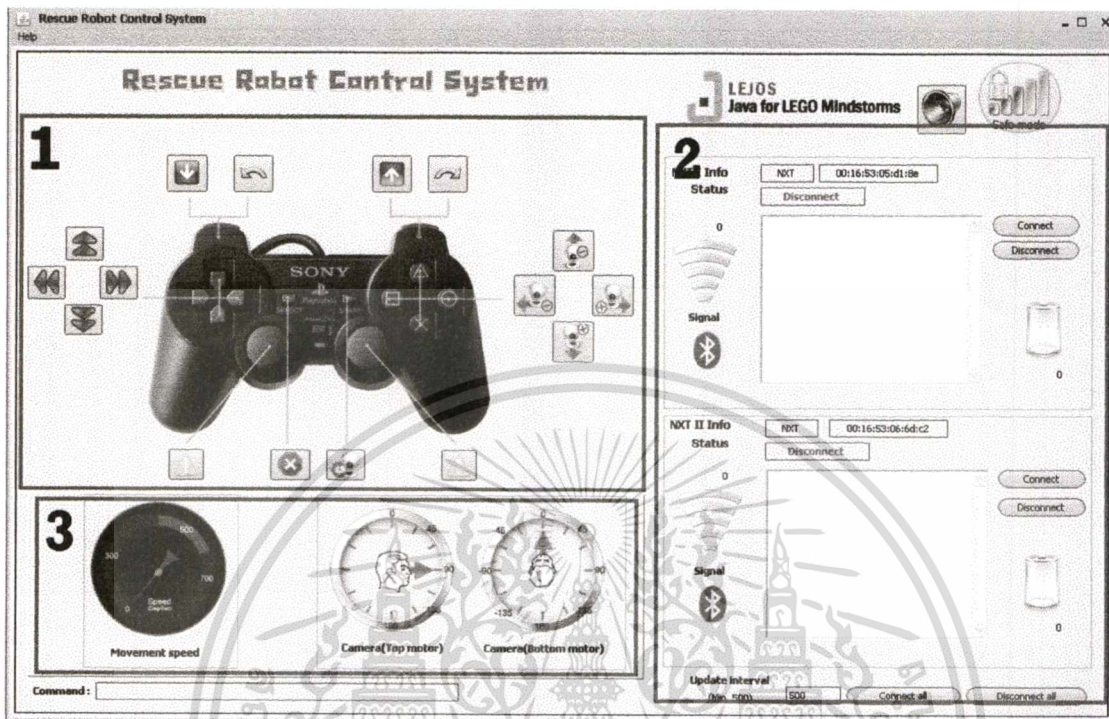


**ภาคผนวก จ**

**คู่มือการใช้งานระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสาบภัย**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# คู่มือการใช้งานระบบควบคุมหุ่นยนต์ช่วยเหลือผู้ประสพภัย

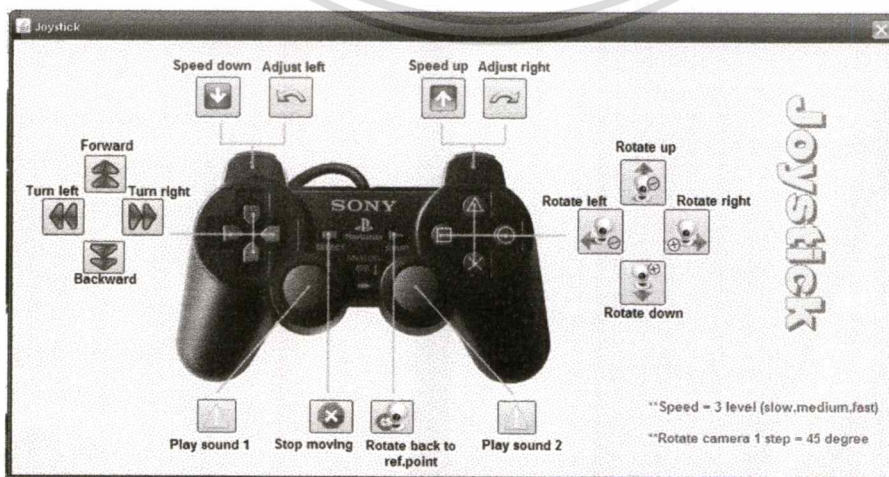


รูปที่ จ.1 แสดงส่วนการทำงานต่างๆของระบบ

จากรูปที่ จ.1

ส่วนที่ 1 คือ ส่วนควบคุมการเคลื่อนที่

ผู้ควบคุมสามารถควบคุมการทำงานต่างๆของหุ่นยนต์ได้ โดยสามารถดูคำอธิบายปุ่มต่างๆดังรูปที่ จ.2 ได้จากเมนู Help > Joystick



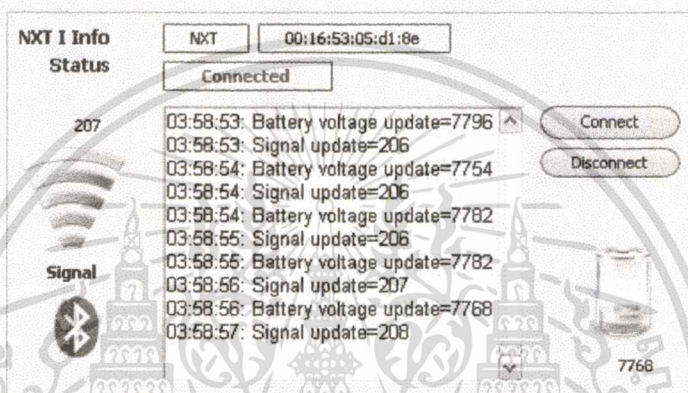
รูปที่ จ.2 แสดงส่วนอธิบายการใช้งานปุ่มต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนที่ 2 คือ ส่วนสร้างการเชื่อมต่อและแสดงระดับสัญญาณ/ระดับแบตเตอรี่

ผู้ควบคุมสามารถการเชื่อมต่อหุ่นยนต์โดยคลิกที่ปุ่ม Connect all หรือหากต้องการยกเลิกการเชื่อมต่อให้คลิกที่ปุ่ม Disconnect all นอกจากนี้ผู้ควบคุมยังสามารถเลือกทำการเชื่อมต่อ/ยกเลิกการเชื่อมต่อ เฉพาะเอนเอ็กซ์ทีคอนโทรลเลอร์ตัวใดตัวหนึ่งก็สามารถทำได้

เมื่อทำการเชื่อมต่อเสร็จสมบูรณ์ระบบจะแสดงระดับสัญญาณ/ระดับแบตเตอรี่ดังรูปที่ จ.3



รูปที่ จ.3 แสดงระดับสัญญาณ/ระดับแบตเตอรี่

ส่วนที่ 3 คือ ส่วนแสดงทิศทางการหมุนของกล้องและระดับความเร็วในการเคลื่อนที่ของหุ่นยนต์

## ประวัติผู้เขียน

ชื่อผู้เขียน นายสิทธิศักดิ์ แผละออง

วันเดือนปีเกิด 1 มีนาคม 2528

สถานที่เกิด ภูเก็ต

### ประวัติการศึกษา

จบการศึกษาระดับมัธยมศึกษาตอนปลาย โรงเรียนพิชัยรัตนาคาร ปีการศึกษา 2546

จบการศึกษาระดับปริญญาตรี วิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สาขา

วิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยสงขลานครินทร์ ปีการศึกษา 2550



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้