รหัสโครงการ SUT7-711-44-12-37

รายงานการวิจัย

การออกแบบสร้างหุ่นยนต์บังคับด้วยคอมพิวเตอร์ เพื่อใช้ในการพัฒนางานทางด้านปัญญาประดิษฐ์

Design of Computer-based Control Robot for Artificial Intelligence Applications

ได้รับทุนอุดหนุนการวิจัยจาก
มหาวิทยาลัยเทคโนโลยีสุรนารี

ผลงานวิจัยเป็นความรับผิดชอบของหัวหน้าโครงการวิจัยแต่เพียงผู้เดียว

รหัสโครงการ SUT7-711-44-12-37

รายงานการวิจัย

การออกแบบสร้างหุ่นยนต์บังคับด้วยคอมพิวเตอร์ เพื่อใช้ในการพัฒนางานทางด้านปัญญาประดิษฐ์

Design of Computer-based Control Robot for Artificial Intelligence Applications

คณะผู้วิจัย

หัวหน้าโครงการ

ผศ.ดร.อาทิตย์ ศรีแก้ว

สาขาวิชาวิศวกรรมไฟฟ้า

สำนักวิชาวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีสุรนารี

ได้รับทุนอุดหนุนการวิจัยจากมหาวิทยาลัยเทคโนโลยีสุรนารี ปีงบประมาณพ.ศ. 2544

ผลงานวิจัยเป็นความรับผิดชอบของหัวหน้าโครงการวิจัยแต่เพียงผู้เดียว

มีนาคม 2547

กิตติกรรมประกาศ

โดยมิได้กล่าวชื่อไว้ ณ ที่นี้ ขอขอบพระคุณทุกๆ ท่านจากใจจริงที่ให้ความช่วยเหลือทั้งโดยตรงและโดยอ้อม ทำให้งานวิจัยนี้สำเร็จลุล่วงไปได้ด้วยดี การวิจัยครั้งนี้ได้รับทุนอุดหนุนการวิจัยจากมหาวิทยาลัยเทคโนโลยีสุรนารี ปีงบประมาณ 2545

บทคัดย่อ

ในการออกแบบสร้างหุ่นยนต์จำเป็นต้องใช้เทคโนโลยีหลากหลายประกอบเข้าด้วยกัน เทคโนโลยีต่างๆ ที่จำเป็นต้องใช้นี้ขึ้นอยู่กับประเภทของหุ่นยนต์ที่ต้องการและวัตถุประสงค์ในการนำเอาหุ่นยนต์ไปใช้งาน งานวิจัยนี้นำเสนอการออกแบบและพัฒนาสร้างหุ่นยนต์เคลื่อนที่ได้ในที่ราบ โดยเน้นที่คุณลักษณะสำคัญในการนำเอาไปประยุกต์ใช้งานทางด้านปัญญาประดิษฐ์ได้ ไม่ว่าจะเป็นการควบคุมแบบไร้สายจากคอมพิวเตอร์ส่วนบุคคลและระบบการมองเห็นของหุ่นยนต์ เทคโนโลยีที่เลือกใช้เน้นไปที่เทคโนโลยีที่มีต้นทุนไม่สูงและสามารถหาได้ภายในประเทศ นอกไปจากนั้นแล้วในงานวิจัยนี้ยังได้ทำการพัฒนาระบบตัวอย่างในการประยุกต์ใช้หุ่นยนต์เคลื่อนที่ในการติดตามการเคลื่อนที่ของวัตถุ โดยอาศัยการคัดแยกสีของวัตถุเป็นเป้าหมาย เทคโนโลยีการออกแบบและพัฒนาที่ได้จากงานวิจัยนี้สามารถนำเอาไปใช้เป็นพื้นฐานในการออกแบบหุ่นยนต์ประเภทอื่นๆ ได้อย่างมีประสิทธิภาพ

Abstract

There are various kinds of technology that are essential for designing a robot. All depends on types of a robot and objectives of using that robot. This research presents design and development of a mobile robot which has important features that can be used in artificial intelligent fields of study. These include the ability of wireless control from computer workstation and the capability of robot vision system. All technology used in this work is considerably affordable to keep the budget at low cost. Moreover, this research has also developed an application of using the mobile robot as a color-based object tracking robot to test the performance of the system. All skill and technology achieved from this research can be used fundamentally in developing other kinds of robots.

สารบัญ

หน้า

กิตติกรรมประกาศ ..... ก
บทคัดย่อภาษาไทย ..... ข
บทคัดย่อภาษาอังกฤษ ..... ค
สารบัญ ..... ง
สารบัญตาราง ..... ฉ
สารบัญภาพ ..... ช
บทที่ 1 บทนำ
วัตถุประสงค์ของการวิจัย 2
ขอบเขตของการวิจัย 2
วิธีดำเนินการวิจัย 2
ประโยชน์ที่ได้รับจากการวิจัย 3
บทที่ 2 หุ่นยนต์เคลื่อนที่ได้
บทนำ 4
หุ่นยนต์ Shadow Liberator 4
หุ่นยนต์ Helpmate 6
หุ่นยนต์ ATVR 7
หุ่นยนต์ Xavier 8
หุ่นยนต์ Nomad 10
หุ่นยนต์ MDOF 11
หุ่นยนต์ Rhino 12
หุ่นยนต์ Nomad 200 13
สรุป 14
บทที่ 3 การออกแบบและสร้างหุ่นยนต์ GUNDAM
บทนำ 15
GUNDAM I 16
GUNDAM II 38
GUNDAM III 47
สรุป 48

สารบัญ (ต่อ)

หน้า

บทที่ 4 ผลการทดลองประยุกต์ใช้งานหุ่นยนต์เคลื่อนที่ GUNDAM

บทนำ 50

การประยุกต์ใช้การประมวลผลภาพในการควบคุมหุ่นยนต์ 52

สรุป.....58

บทที่ 4 บทสรุป

สรุปผลการวิจัย 61

ข้อเสนอแนะ 63

ภาคผนวก

ภาคผนวก ก 64

ประวัติผู้วิจัย 71

สารบัญตาราง

ตารางที่ **หน้า**

3.1 ลักษณะต่างๆ ทางโครงสร้างของหุ่นยนต์ GUNDAM 24

3.2 หน้าที่ของขา (pin) จากพอร์ตขนานที่ใช้ควบคุมภาคส่ง 32

3.3 หน้าที่ของขาเอาต์พุตที่ออกจาก IC 8255 33

3.4 ฟังก์ชันการทำงานของอินพุต/เอาต์พุตต่างๆ ของไมโครคอนโทรลเลอร์..... 34

3.5 ลักษณะข้อมูลที่ใช้ควบคุมอุปกรณ์ส่ง FM 35

3.6 หน้าที่การทำงานของแต่ละช่อง..... 36

3.7 พารามิเตอร์ของแมโคร CTL_CODE..... 41

สารบัญภาพ

รูปที่ **หน้า**

2.1 หุ่นยนต์ Shadow Liberator (แหล่งที่มา: http://www.shadow.org.uk)..... 5
2.2 หุ่นยนต์ Helpmate (แหล่งที่มา: http://www.pyxis.com)..... 6
2.3 หุ่นยนต์ ATVR-Jr (แหล่งที่มา: http://rescue.isr.ist.utl.pt/index.php)..... 8
2.4 หุ่นยนต์ Xavier (แหล่งที่มา: http://pecan.srv.cs.cmu.edu/~Xavier).....9
2.5 หุ่นยนต์ Amelia (แหล่งที่มา: http://www-2.cs.cmu.edu/~Xavier/amelia.html) 9
2.6 ทีมหุ่นยนต์ในโครงการ Minnow (แหล่งที่มา: http://www-2.cs.cmu.edu/~coral/minnow) 10
2.7 หุ่นยนต์ Nomad ในการสำรวจพื้นที่ห่างไกลในแอนตาร์คติค (แหล่งที่มา: http://www.frc.ri.cmu.edu/projects/meteorobot2000) 11
2.8 หุ่นยนต์ MDOF (แหล่งที่มา: http://www.engin.umich.edu/research/mrl/00MoRob.html) 12
2.9 หุ่นยนต์ไกด์นำทาง Rhino (แหล่งที่มา: http://www.informatik.uni-bonn.de/~rhino) 13
2.10 หุ่นยนต์ไกด์นำทาง Minerva (แหล่งที่มา: http://www-2.cs.cmu.edu/~minerva) 13
2.11 หุ่นยนต์ Nomad 200 (แหล่งที่มา: http://underdog.stanford.edu/robots.html) 14
3.1 ระบบปรับศูนย์ล้อ 4 ล้อ 18
3.2 การปรับศูนย์ล้อ 19
3.3 ส่วนของโครงสร้างชุดหน้าและชุดหลัง 20
3.4 การประกอบโครงสร้างชุดหน้าและชุดหลังเข้าด้วยกัน 20
3.5 ระบบการปรับศูนย์โครงสร้าง 21
3.6 ลักษณะของหุ่นยนต์ GUNDAM เมื่อมองจากด้านข้าง 22
3.7 ลักษณะของหุ่นยนต์ GUNDAM เมื่อมองจากด้านบน 23
3.8 ลักษณะของหุ่นยนต์ GUNDAM เมื่อมองจากด้านหน้า 23
3.9 ส่วนประกอบต่างๆ ของหุ่นยนต์ GUNDAM 25
3.10 ส่วนประกอบต่างๆ ของหุ่นยนต์ GUNDAM มองจากด้านข้าง..... 25
3.11 วงจรควบคุมของหุ่นยนต์ GUNDAM 27
3.12 ลักษณะการทำงานของอุปกรณ์บังคับเครื่องบินเล็ก 29
3.13 ลักษณะสัญญาณ PWM ที่ออกจากภาครับของอุปกรณ์บังคับเครื่องบินเล็ก 30
3.14 การดัดแปลงภาคส่งอุปกรณ์บังคับเครื่องบินเล็ก 31
3.15 การดัดแปลงภาคส่งอุปกรณ์บังคับเครื่องบินเล็ก (ปรับปรุง) 32

สารบัญภาพ (ต่อ)

รูปที่ **หน้า**

3.16 แผนผังการดัดแปลงภาครับวงจรบังคับเครื่องบินเล็ก 33
3.17 การทำงานของโปรแกรม GDEYEs..... 42
3.18 โปรแกรม GDEyes 46
3.19 วงจรควบคุมและวงจรส่งข้อมูล 47
3.20 วงจรขับมอเตอร์ด้วยการปรับขนาดความกว้างของสัญญาณพัลส์ (PWM) 48
4.1 คุณลักษณะสำคัญของหุ่นยนต์ GUNDAM I (ส่วนที่ 1)..... 50
4.2 คุณลักษณะสำคัญของหุ่นยนต์ GUNDAM I (ส่วนที่ 2)..... 50
4.3 การติดตามวัตถุด้วยสี..... 53
4.4 การเลือกพื้นที่เฉพาะส่วนของวัตถุที่ต้องการโดยใช้ Lasso Tool..... 54
4.5 ตัวอย่างฮิสโตแกรมสีแดง สีเขียวและสีน้ำเงินของลูกบอล 54
4.6 การจัดเก็บภาพในหน่วยความจำและการคำนวณหาตำแหน่งพิกเซล 55
4.7 ผลการทำคัดแยกสี (color segmentation) 56
4.8 การบังคับเลี้ยวซ้าย-ขวา 57
4.9 การบังคับเดินหน้า-ถอยหลัง 58
4.10 ภาพถ่ายหุ่นยนต์ GUNDAM III 59
4.11 ภาพถ่ายหุ่นยนต์ GUNDAM III (ต่อ) 60

บทที่ 1
บทนำ

เทคโนโลยีหุ่นยนต์ได้รับการพัฒนาจากอดีตมาอย่างต่อเนื่อง หลายๆ องค์กรหรือหน่วยงานทั้งภาคเอกชนและรัฐบาลได้ให้ความสนใจและความสำคัญในการพัฒนาเทคโนโลยีหุ่นยนต์เพิ่มมากขึ้นเรื่อยๆ ในอดีตนั้น เทคโนโลยีหุ่นยนต์ยังคงจำกัดอยู่ในระดับห้องปฏิบัติการ การใช้งานหุ่นยนต์เองยังจำกัดอยู่ในโรงงานอุตสาหกรรม อย่างไรก็ดี ในปัจจุบันได้มีการพัฒนาเทคโนโลยีด้านต่างๆ ที่มีส่วนช่วยอำนวยความสะดวกของมนุษย์เพิ่มมากขึ้น เทคโนโลยีที่เกี่ยวข้องกับหุ่นยนต์เองได้เริ่มเข้ามามีบทบาทใกล้ชิดกับมนุษย์ หรือสามารถกล่าวได้ว่าเทคโนโลยีหุ่นยนต์ได้เข้ามามีส่วนในวิตประจำวันของคนเรามากขึ้นทุกที

โดยปกติแล้วหุ่นยนต์สามารถแบ่งได้คร่าวๆ เป็น 2 ประเภท คือหุ่นยนต์ที่ไม่มีการเคลื่อนที่และหุ่นยนต์ที่เคลื่อนที่ได้ หุ่นยนต์ที่ไม่มีการเคลื่อนที่คือหุ่นยนต์ที่มีฐานยึดติดกับที่ใดที่หนึ่ง จึงไม่มีการเคลื่อนที่ของทั้งตัวหุ่นยนต์ในขณะที่มีการปฏิบัติงาน ตัวอย่างเช่นหุ่นยนต์แขนกลพูม่า (puma robot) ที่ใช้ในงานอุตสาหกรรมทั่วๆ ไปเป็นต้น ในขณะที่หุ่นยนต์ที่เคลื่อนที่ได้ (หรือ mobile robot) จะสามารถเคลื่อนที่ได้อย่างอิสระ การออกแบบสร้างหรือการนำไปประยุกต์ใช้งานของหุ่นยนต์ทั้งสองชนิดจึงค่อนข้างมีความแตกต่างกันพอสมควร

ในส่วนของเทคโนโลยีการควบคุมหุ่นยนต์นั้นมีการค้นคว้าพัฒนากันอย่างมากมาย ไม่ว่าจะเป็นการควบคุมในระดับต่ำ (low level control) เช่นการควบคุมตำแหน่งหรือความเร็วของมอเตอร์ภายในโครงสร้างของหุ่นยนต์ ฯลฯ หรือการควบคุมในระดับสูง (high level control) เช่นการควบคุมทิศทางในระบบนำวิถีของหุ่นยนต์เคลื่อนที่ได้ (mobile robot navigation) ฯลฯ เทคโนโลยีการควบคุมหุ่นยนต์นี้มีความแตกต่างกันออกไปตามโครงสร้างของหุ่นยนต์ ชนิดของตัวตรวจจับที่ใช้และวัตถุประสงค์ในการประยุกต์ใช้งานหุ่นยนต์

งานวิจัยนี้มุ่งเน้นไปที่การศึกษาเทคโนโลยีหุ่นยนต์เคลื่อนที่ได้ เพื่อนำมาประยุกต์ออกแบบสร้างหุ่นยนต์เคลื่อนที่ได้ โดยสามารถนำเอาหุ่นยนต์ไปประยุกต์ใช้กับงานทางด้านปัญญาประดิษฐ์ได้ ดังนั้นคุณลักษณะเบื้องต้นของหุ่นยนต์ที่ออกแบบสร้างนี้ จะต้องมีความสามารถในการควบคุมด้วยคอมพิวเตอร์ส่วนบุคคลแบบไร้สาย เพื่อความคล่องตัวในการทำงานในโหมดการเคลื่อนที่ และมีเพียงการใช้ตัวตรวจจับภาพจากกล้องวิดีโอแบบไร้สายสำหรับการควบคุมการทำงานของหุ่นยนต์

เนื่องมาจากการมองเห็นของมนุษย์ถือเป็นการตรวจจับตามธรรมชาติอย่างหนึ่ง ที่ซึ่งมีการนำเอาไปประยุกต์ใช้งานด้านการมองเห็นของคอมพิวเตอร์ (computer vision) หรือการมองเห็นของเครื่องจักร (machine vision) อย่างแพร่หลาย และมีประสิทธิภาพ การควบคุมหุ่นยนต์ด้วยสัญญาณภาพจึงมีความเหมาะสมในงานทางด้านปัญญาประดิษฐ์ และเป็นหัวข้อวิจัยที่กำลังเป็นที่สนใจเพิ่มมาก

ขึ้นในปัจจุบัน ส่วนการควบคุมด้วยคอมพิวเตอร์นั้น เป็นสิ่งจำเป็นสำหรับการประมวลผลสัญญาณภาพ เนื่องมาจากการประมวลผลสัญญาณภาพจำเป็นจะต้องมีการคำนวณข้อมูลในปริมาณที่มาก การประมวลผลด้วยไมโครคอนโทรเลอร์จึงไม่เพียงพอต่อการใช้งานทั่วๆ ไป นอกไปจากนั้นแล้วการเชื่อมต่อ (interface) เพื่ออ่านข้อมูลสัญญาณภาพจากกล้องวิดีโอเข้าสู่หน่วยความจำของไมโครคอนโทรลเลอร์โดยตรงนั้น ค่อนข้างจะมีข้อจำกัดหลายๆ อย่าง ไม่ว่าจะเป็นเรื่องของความเร็วหรือปริมาณหน่วยความจำที่ไม่เพียงพอต่อการจัดการข้อมูลภาพซึ่งมีปริมาณค่อนข้างมาก หุ่นยนต์ควรที่จะมีความสามารถในการส่งสัญญาณภาพไปทำการประมวลผลที่หน่วยประมวลผลซึ่งมีประสิทธิภาพเพียงพอ

หุ่นยนต์เคลื่อนที่ที่มีใช้ในระดับห้องปฏิบัติการทั่วๆ ไป ส่วนใหญ่แล้วจะเป็นผลิตภัณฑ์เชิงการค้าจากบริษัทผู้ผลิตและพัฒนาทางด้านหุ่นยนต์ ซึ่งหุ่นยนต์เคลื่อนที่ของแต่ละบริษัทมีคุณสมบัติแตกต่างกันไป อย่างไรก็ดี ผลิตภัณฑ์ทางด้านหุ่นยนต์ยังอยู่ในระดับการวิจัย ที่ซึ่งเทคโนโลยีต่างๆ ยังคงมีการพัฒนาอย่างต่อเนื่อง ปริมาณการนำเอาผลิตภัณฑ์หุ่นยนต์มาใช้ในชีวิตประจำวันยังไม่มากเพียงพอ ด้วยเหตุนี้จึงทำให้ราคาต้นทุนการผลิตยังค่อนข้างสูง โดยเฉพาะในประเทศไทยที่ซึ่งการพัฒนาเทคโนโลยีเพื่อผลิตหุ่นยนต์ยังอยู่ในระดับค่อนข้างต่ำ ห้องปฏิบัติการทางด้านหุ่นยนต์หลายๆ แห่งยังคงต้องนำเข้าชิ้นส่วนหรือตัวหุ่นยนต์จากภายนอกประเทศด้วยค่าใช้จ่ายที่สูง

1. วัตถุประสงค์

งานวิจัยนี้ได้ทำการศึกษาเทคโนโลยีด้านหุ่นยนต์ เพื่อนำมาออกแบบพัฒนาหุ่นยนต์เคลื่อนที่โดยมุ่งเน้นให้หุ่นยนต์มีคุณลักษณะสำคัญที่เพียงพอต่อการนำมาประยุกต์ใช้งานทางด้านปัญญาประดิษฐ์ เช่น มีความสามารถในการรับภาพจากกล้องวิดีโอที่ติดบนตัวหุ่นยนต์เป็นต้น นอกจากนั้นแล้วการออกแบบหุ่นยนต์ที่ได้ควรจะใช้ชิ้นส่วนที่สามารถหาได้ภายในประเทศ โดยไม่จำเป็นต้องสั่งเป็นพิเศษจากต่างประเทศ เพื่อให้ค่าใช้จ่ายในการสร้างมีราคาไม่สูง

2. ขอบเขตของการวิจัย

ขอบเขตของการวิจัยคือการพัฒนาออกแบบสร้างหุ่นยนต์ ซึ่งสามารถสรุปคุณสมบัติที่สำคัญของหุ่นยนต์ที่จะทำการสร้างได้ดังนี้

- สามารถเคลื่อนที่ได้ในพื้นราบ โดยใช้การขับเคลื่อนแบบล้อ
- สามารถควบคุมหุ่นยนต์แบบไร้สายได้จากคอมพิวเตอร์ส่วนบุคคล
- สามารถส่งภาพจากกล้องวิดีโอที่ติดตั้งอยู่บนตัวหุ่นยนต์ไปยังคอมพิวเตอร์ เพื่อทำการประมวลผลเชิงปัญญาประดิษฐ์ได้

3. วิธีดำเนินการวิจัย

การดำเนินการวิจัยสามารถสรุปโดยย่อได้เป็นลำดับดังนี้

- ศึกษาเทคโนโลยีฮาร์ดแวร์ที่เกี่ยวข้องกับการออกแบบสร้างหุ่นยนต์เคลื่อนที่ โดยเฉพาะเทคโนโลยีชนิดไร้สาย
- ทำการออกแบบและสร้างหุ่นยนต์ตามคุณสมบัติที่กำหนดไว้
- ทดสอบการทำงานของหุ่นยนต์ว่าสามารถทำงานได้ตามที่ออกแบบ
- ทดลองพัฒนาระบบปัญญาประดิษฐ์ทางด้านการมองเห็นของคอมพิวเตอร์ เพื่อทดสอบประสิทธิภาพของระบบในแง่การประยุกต์ใช้งาน

4. ประโยชน์ที่คาดว่าจะได้รับ

- ได้ศึกษาเทคโนโลยีในการสร้างหุ่นยนต์เคลื่อนที่ ซึ่งสามารถนำมาประยุกต์ใช้งานด้านหุ่นยนต์แบบอื่นๆ หรือในงานด้านอื่นๆ ที่เกี่ยวข้อง เช่นระบบควบคุม ฯลฯ
- ได้ลดการนำเข้าหุ่นยนต์จากต่างประเทศเพื่อใช้ในงานวิจัย
- ได้มีโอกาสในการพัฒนาหุ่นยนต์เพื่อสามารถนำไปใช้ในระดับที่นอกเหนือไปจากงานวิจัย เช่นนำไปใช้ในงานอุตสาหกรรม หรืองานประชาสัมพันธ์ เป็นต้น
- ได้หุ่นยนต์ที่มีคุณสมบัติเหมาะสมสำหรับการพัฒนางานทางด้านปัญญาประดิษฐ์ ซึ่งถือเป็นส่วนสำคัญที่สุดในการพัฒนางานวิจัยต่อไป

รายงานวิจัยฉบับนี้แบ่งเนื้อหาสำคัญออกเป็นทั้งหมด 5 บท โดยในบทที่ 2 จะทำการกล่าวถึงตัวอย่างของหุ่นยนต์เคลื่อนที่ที่มีใช้ในห้องปฏิบัติการต่างๆ ในบทที่ 3 จะกล่าวถึงการออกแบบสร้างหุ่นยนต์เคลื่อนที่ในงานวิจัยนี้ การสร้างหุ่นยนต์ดังกล่าวได้มีช่วงการพัฒนาอยู่สามช่วงใหญ่ๆ เนื่องจากในขณะที่ทำการออกแบบสร้างนั้น เทคโนโลยีต่างๆ ทั้งทางซอฟท์แวร์และฮาร์ดแวร์มีการเปลี่ยนแปลง การสร้างหุ่นยนต์จึงต้องมีการเปลี่ยนแปลงตามความเหมาะสม ในบทที่ 4 จะได้กล่าวถึงผลการทดสอบการใช้งานหุ่นยนต์ที่ได้ โดยได้ทำการพัฒนาตัวอย่างการประยุกต์ใช้งานหุ่นยนต์เคลื่อนที่ในการติดตามวัตถุ (object tracking) โดยใช้ระบบการมองเห็นของหุ่นยนต์เป็นหลัก บทสรุปงานวิจัยมีรายละเอียดอยู่ในบทที่ 5

บทที่ 2
หุ่นยนต์เคลื่อนที่ได้

1. บทนำ

หุ่นยนต์เคลื่อนที่ได้ที่มีใช้ในห้องปฏิบัติการนั้น มีทั้งที่ถูกสร้างและพัฒนาขึ้นเองภายในห้องปฏิบัติการและจากผลิตภัณฑ์เชิงการค้า ข้อได้เปรียบของหุ่นยนต์ที่สร้างและพัฒนาเองภายในห้องปฏิบัติการจะมีคุณสมบัติที่ตรงกับความต้องการในการพัฒนางานวิจัย หุ่นยนต์ประเภทนี้จะมีโครงสร้างและการควบคุมเฉพาะอย่าง อย่างไรก็ดีการพัฒนาหุ่นยนต์เองจำเป็นต้องใช้เวลาค่อนข้างมากกว่าจะได้มาซึ่งหุ่นยนต์ที่สามารถนำมาประยุกต์ใช้ในระดับงานวิจัยได้ ในขณะเดียวกัน หุ่นยนต์เชิงการค้าที่มีบริษัททางด้านหุ่นยนต์เป็นผู้ผลิตและพัฒนาอย่างต่อเนื่อง สามารถตอบสนองการใช้งานในระดับงานวิจัยได้ในระดับหนึ่ง เนื่องจากการนำเอาหุ่นยนต์ไปใช้งานนั้นมีได้หลากหลาย หุ่นยนต์เชิงการค้าจึงมีคุณสมบัติทั่วๆ ไป มากกว่าจะเป็นคุณสมบัติเฉพาะอย่าง ผู้ใช้เองจึงอาจจะต้องทำการปรับแต่งดัดแปลงให้มีคุณสมบัติและความสามารถตามที่ต้องการ ในกรณีนี้ผู้ใช้ไม่จำเป็นต้องเสียเวลาในการพัฒนาตัวหุ่นยนต์ หรือแม้แต่ซอฟท์แวร์ในการควบคุม แต่ผู้ใช้สามารถเน้นไปที่การพัฒนาการประยุกต์ใช้งาน อัลกอริทึมในระดับสูงหรือการประยุกต์ใช้งานทางด้านปัญญาประดิษฐ์แทน โดยทั้งหมดแลกกับราคาของตัวหุ่นยนต์ที่ค่อนข้างจะสูง

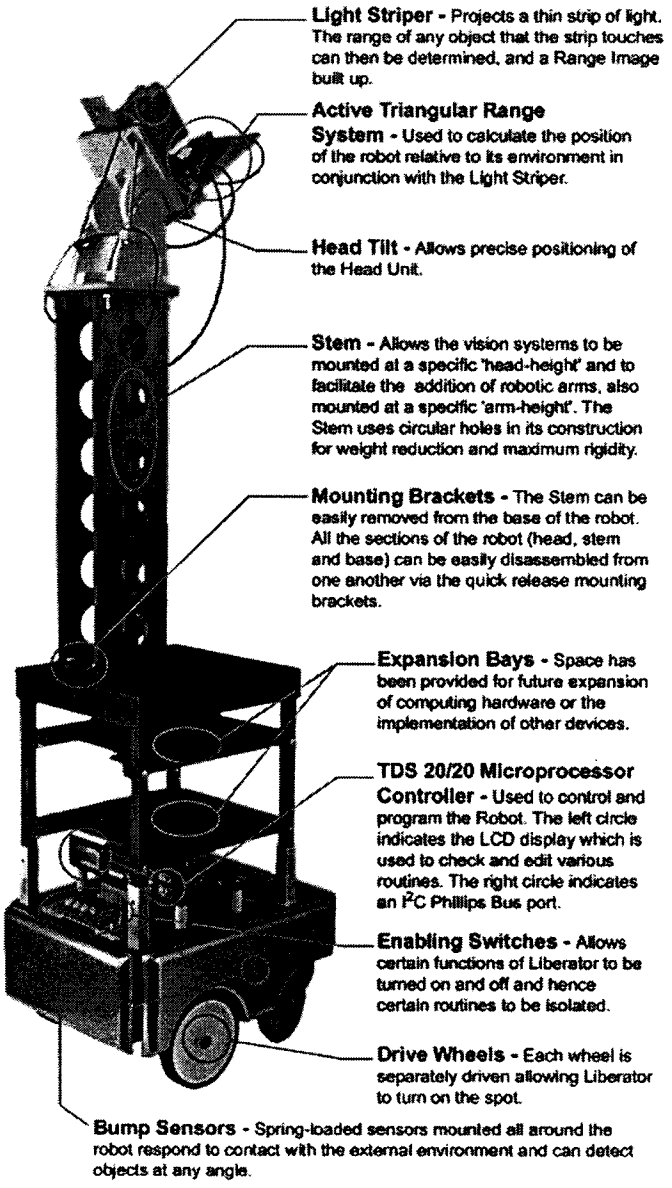
ในบทนี้จะกล่าวถึงรายละเอียดโดยสังเขปของตัวอย่างหุ่นยนต์เคลื่อนที่แบบต่างๆ ที่มีใช้ในห้องปฏิบัติการ ในปัจจุบันได้มีงานวิจัยสร้างและพัฒนาหุ่นยนต์ประเภท "มนุษยนต์" หรือ "humanoid robot" ที่ซึ่งการเคลื่อนที่ของหุ่นยนต์จะเป็นลักษณะของการเดินสองขาแบบมนุษย์ (walker) อย่างไรก็ตามงานวิจัยนี้จะเน้นไปที่การพัฒนาสร้างหุ่นยนต์เคลื่อนที่ที่มีระบบการเคลื่อนที่แบบล้อ การศึกษาภูมิหลังของหุ่นยนต์แบบต่างจึงจำกัดอยู่ที่หุ่นยนต์เคลื่อนที่แบบล้อเท่านั้น

2. หุ่นยนต์ Shadow Liberator

หุ่นยนต์ Shadow Liberator (ดูรูปที่ 2.1 ประกอบ) เป็นหุ่นยนต์เคลื่อนที่ต้นแบบ พัฒนาโดยบริษัท Shadow Robot Company Ltd. ในประเทศอังกฤษ หุ่นดังกล่าวถูกพัฒนาขึ้นโดยมีจุดประสงค์หลักที่จะทำการวิจัยระบบการมองเห็นของหุ่นยนต์ก่อนที่จะนำไปใช้กับหุ่นยนต์เดินได้ของบริษัทเองที่เรียกว่า Shadow Biped หุ่นยนต์ Shadow Liberator มีขนาดใกล้เคียงกับมนุษย์ โดยมีความสูงประมาณ 155 เซนติเมตร ติดตั้งกล้องในส่วนบนสุด ลำตัวมีช่องสำหรับการติดตั้งชิ้นส่วนเพิ่มเติม เช่น แขนกล ฯลฯ สัญญาณภาพจากกล้องจะถูกส่งไปยังคอมพิวเตอร์ส่วนบุคคล ที่ซึ่งมีการ์ดจับสัญญาณภาพเข้าสู่คอมพิวเตอร์เพื่อทำการประมวลผลและส่งคำสั่งควบคุมกลับมายังตัวหุ่นยนต์อีกที

ระบบขับเคลื่อนของ Shadow Liberator ประกอบไปด้วยการขับเคลื่อนหลักแบบ 2 ล้อ โดยมีอีก 2 ล้อทำการประคองการเคลื่อนที่ (คล้ายกับรถเข็นคนพิการ) มอเตอร์ควบคุมแบบปรับความกว้าง

พัลส์ (pulse width modulation) แต่ละมอเตอร์มีตัวเข้ารหัส (encoder) เพื่อทำการส่งข้อมูลของการหมุนของมอเตอร์ไปยังตัวควบคุม ตัวควบคุมใช้ คอมพิวเตอร์ Triangle Digital 2020 16 บิต ANSI Forth โดยส่งข้อมูลสื่อสารกับคอมพิวเตอร์หลักผ่านทางบัส I2C รอบๆ ฐานของหุ่นยนต์มีการติดตั้งตัวตรวจจับในกรณีที่หุ่นยนต์เคลื่อนที่ไปแตะสิ่งใด

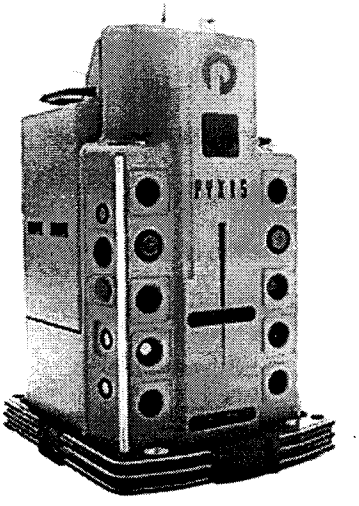


รูปที่ 2.1 หุ่นยนต์ Shadow Liberator (แหล่งที่มา: http://www.shadow.org.uk)

ปัจจุบันหุ่นยนต์ Shadow Liberator ไม่ได้รับการพัฒนาต่อแต่อย่างใด เนื่องจากทางบริษัทได้มุ่งเน้นไปทำการพัฒนาหุ่นยนต์เดินได้ Shadow Biped แทน

3. หุ่นยนต์ Helpmate

หุ่นยนต์ Helpmate (ดูรูปที่ 2.2 ประกอบ) จากบริษัท Pyxis Corporation เป็นหุ่นยนต์เดินแบบไม่ใช้เส้นนำทาง (trackless) จุดประสงค์หลักในการออกแบบสร้างหุ่นยนต์ Helpmate เพื่อใช้ในบริการภายในโรงพยาบาล บริษัท Pyxis Corporation เป็นบริษัทที่ทำธุรกิจทางด้านสุขภาพอนามัยโดยเฉพาะระบบการจัดจ่ายยาเวชภัณฑ์แบบอัตโนมัติ วิศวกรของบริษัทจึงได้ออกแบบหุ่นยนต์เพื่อทำงานภายในอาคารแบบอัตโนมัติ



รูปที่ 2.2 หุ่นยนต์ Helpmate (แหล่งที่มา: http://www.pyxis.com)

หุ่นยนต์ Helpmate มีขนาดความกว้าง 87.5 เซนติเมตร ความยาว 80 เซนติเมตรและความสูง 140 เซนติเมตร น้ำหนักตัว 270 กิโลกรัม สามารถบรรทุกน้ำหนัก 90 กิโลกรัม ซึ่งมีขนาดความกว้าง 60 เซนติเมตร ความสูง 65 เซนติเมตรและความลึก 40 เซนติเมตร (ประมาณ 5.5 ฟุต^3) หุ่นยนต์ Helpmate มีการติดตั้งตัวตรวจจับแบบอุลตราโซนิค 18 ตัวสำหรับตรวจจับวัตถุด้านหน้าและด้านข้างของหุ่นยนต์ รอบๆ ฐานของหุ่นยนต์ยังมีตัวตรวจจับอีก 6 ตัวสำหรับตรวจจับสิ่งกีดขวางภายในรัศมีประมาณ 22.5 เซนติเมตร ด้านหน้าของหุ่นยนต์ยังมีการติดตั้งกล้องสำหรับมองหาวัตถุในระยะ 180 เซนติเมตร ข้อมูลภาพที่ได้จะถูกนำไปคำนวณหาเส้นทางเดินของหุ่นยนต์ (ข้อมูลจาก http://www.pyxis.com)

หุ่นยนต์ Helpmate สามารถเดินผ่านช่องทางแคบขนาดประมาณ 1 เมตรได้อย่างอัตโนมัติ ระบบขับเคลื่อนสามารถขึ้นลงในที่ที่มีความชันประมาณ 10% ความเร็วเฉลี่ยของหุ่นยนต์อยู่ที่ประมาณ 60 เซนติเมตร/วินาที ตัวหุ่นยนต์ยังมีความสามารถในการเชื่อมต่อกับการ์ดอินเตอร์เน็ตแบบไร้สายได้อีกด้วย

หุ่นยนต์ Helpmate ถูกออกแบบสร้างเพื่อใช้งานในโรงพยาบาล หุ่นยนต์จึงได้รับการออกแบบให้มีความปลอดภัยในการใช้งานมากที่สุด ไม่ว่าจะเป็นการหลบหลีกสิ่งกีดขวาง หรือการหยุดทำงานอย่างทันทีทันใดเมื่อมีเหตุผิดปกติเกิดขึ้น มีการนำเอาหุ่นยนต์ Helpmate ไปประยุกต์ใช้ในงานวิจัย ยกตัวอย่างเช่นห้องปฏิบัติการหุ่นยนต์ Intelligent Robotics Laboratory จาก Vanderbilt University นำเอาหุ่นยนต์ Helpmate ไปทำการดัดแปลงทั้งทางด้านฮาร์ดแวร์และซอฟท์แวร์เพื่อใช้ในการพัฒนาระบบทางด้านการนำวิถี ดังรายละเอียดใน

http://eecs.vanderbilt.edu/cis/irl/helpmate.shtml

4. หุ่นยนต์ ATVR

หุ่นยนต์ ATVR เป็นหุ่นยนต์เคลื่อนที่ภายนอกอาคารหรือที่เรียกว่าเป็น outdoor (สามารถใช้ภายในอาคารได้เหมือนกัน) หุ่นยนต์ ATVR หรือ All Terrain Vehicle Robot ได้ถูกออกแบบมาจากรถสะเทินน้ำสะเทินบก เพียงแต่มีขนาดที่เล็กลงและมีระบบควบคุมด้วยคอมพิวเตอร์ ฝ่าย Real World Interface ของบริษัท i-Robots ในประเทศสหรัฐอเมริกาเป็นผู้เริ่มพัฒนาหุ่นยนต์ดังกล่าว (ปัจจุบันเลิกพัฒนาแล้ว) หุ่นยนต์ตระกูล ATVR มีอยู่หลายแบบได้แก่ ATVR-Micro ATVR-Jr (Junior) และ ATVR แสดงหุ่นยนต์ ATVR-Jr รูปที่ 2.3 แสดงหุ่นยนต์ ATVR-Jr ซึ่งมีคุณสมบัติเด่นๆ ดังนี้

ตัวตรวจจับอุลตราโซนิค 17 ตัว (ด้านหน้า 6 ตัว ด้านข้าง 10 ตัวและด้านหลัง 2 ตัว) ติดตั้งคอมพิวเตอร์ Pentium III ทั้งแบบ CPU เดี่ยวและ CPU คู่ ติดต่อสื่อสารแบบไร้สายด้วย RS-232 หรืออินเตอร์เนตการ์ดขนาด 3 Mbps ใช้ดีซีมอเตอร์ขนาด 24 โวลต์ในการขับเคลื่อน ความเร็วในการเคลื่อนที่ประมาณ 1 เมตร/วินาที รับน้ำหนักได้ 25 กิโลกรัม ขนาดความสูง 55 เซนติเมตร ความยาว 62.2 เซนติเมตร ความกว้าง 62.2 เซนติเมตร น้ำหนักตัวหุ่นยนต์ 50 กิโลกรัม

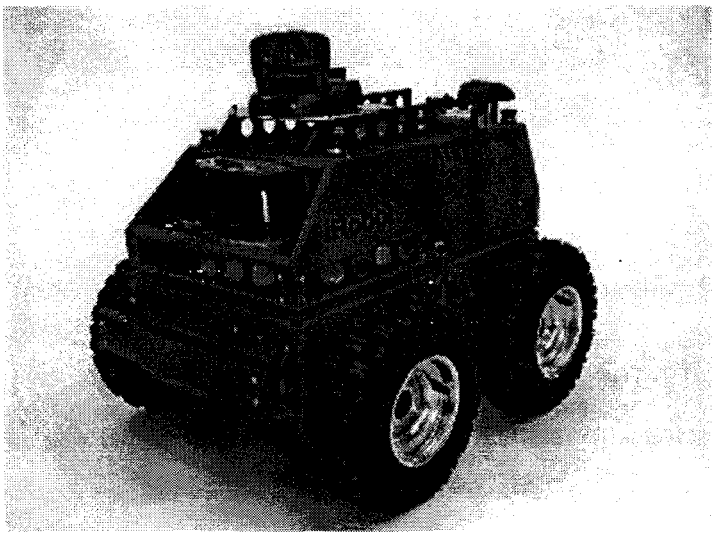
ถึงแม้ว่าในปัจจุบันบริษัทผู้ผลิตหุ่นยนต์ ATVR จะเลิกทำการพัฒนาแล้ว แต่ยังมีผู้นำเอา AVTR ไปประยุกต์ใช้ในงานวิจัยอยู่บ้าง เช่น Institute for Systems and Robotics ในประเทศโปรตุเกสที่ทำการพัฒนาโครงการหุ่นยนต์ช่วยเหลือที่มีระบบการนำวิถีแบบร่วมมือ (cooperative navigation for rescue robots) มีการนำเอา หุ่นยนต์ ATVR-Jr มาทำการดัดแปลง โดยทำการเพิ่มระบบ GPS (global positioning system) และติดตั้งกล้องวิดีโอสำหรับระบบการมองเห็น รายละเอียดงานวิจัยนี้สามารถหาเพิ่มเติมได้จาก

http://rescue.isr.ist.utl.pt/index.php

อีกห้องปฏิบัติการที่มีการนำเอาหุ่นยนต์ ATVR-Jr ไปประยุกต์ใช้ในงานวิจัยได้แก่ Intelligent Robotics Laboratory ของ Vanderbilt University ในประเทศสหรัฐอเมริกา ดังรายละเอียดใน

http://eecs.vanderbilt.edu/cis/irl/index.shtml

ห้องปฏิบัติการนี้ได้ทำการพัฒนาซอฟท์แวร์สำหรับหุ่นยนต์เคลื่อนที่โดยอัตโนมัติ (Mobile Autonomous Robot Software) โดยมีหุ่นยนต์ ATVR-Jr เป็นหุ่นยนต์หลักที่ใช้ในการพัฒนา

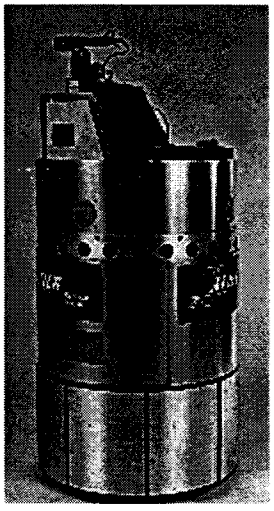


รูปที่ 2.3 หุ่นยนต์ ATVR-Jr (แหล่งที่มา: http://rescue.isr.ist.utl.pt/index.php)

5. หุ่นยนต์ Xavier

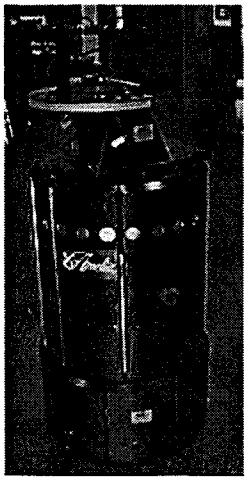
หุ่นยนต์ Xavier (ดูรูปที่ 2.4 ประกอบ) ได้รับการออกแบบสร้างและพัฒนาโดยกลุ่มนักศึกษาจากห้องปฏิบัติการ Learning Robot Lab ของ Carnegie Mellon University เพื่อใช้งานเป็นหุ่นยนต์ต้นแบบในงานวิจัย โครงสร้างหลักของหุ่นยนต์ Xavier ได้มาจากหุ่นยนต์ RWI B24 (โดย Real World Interface Inc.) มีขนาดเส้นผ่าศูนย์กลาง 60 เซนติเมตร ขับเคลื่อน 4 ล้อ ติดตั้งตัวตรวจจับทั้งแบบอุลตราโซนิคและแบบเลเซอร์ รวมไปถึงกล้องวิดีโอสำหรับระบบการมองเห็นของหุ่นยนต์

บนตัวหุ่นยนต์ติดตั้งบอร์ดประมวลผลแบบ 66 MHz Intel 486 และคอมพิวเตอร์ชนิด notebook 486 เชื่อมต่อด้วยอินเตอร์เนตแบบไร้สาย การควบคุมมีทั้งแบบกราฟิคผ่านทาง notebook บนตัวหุ่นยนต์ การควบคุมแบบทางไกลด้วยซอฟท์แวร์พิเศษ (เรียกว่า zephyr) และการควบคุมด้วยเสียง รายละเอียดเพิ่มเติมของหุ่นยนต์และงานวิจัยที่เกี่ยวข้องมีอยู่ใน http://pecan.srv.cs.cmu.edu/~Xavier/



รูปที่ 2.4 หุ่นยนต์ Xavier (แหล่งที่มา: http://pecan.srv.cs.cmu.edu/~Xavier)

นอกไปจากหุ่นยนต์ Xavier แล้ว หุ่นยนต์ตัวใหม่ที่มีประสิทธิภาพเหนือกว่า Xavier ชื่อว่า Amelia ได้รับการออกแบบสร้างและพัฒนาขึ้นมา (ดูรูปที่ 2.5 ประกอบ) หุ่นยนต์ Amelia ยังคงใช้โครงสร้างของหุ่นยนต์ Xavier เป็นต้นแบบในการพัฒนา สิ่งที่เหนือกว่าในหุ่นยนต์ Amelia คือ ความเร็ว (80 เซนติเมตร/วินาที) และการปรับปรุงระบบการควบคุมที่แม่นยำกว่า Amelia มีตัวตรวจจับอยู่ 3 ชนิด ได้แก่ตัวตรวจจับอุลตราโซนิคและอินฟาเรด รวมไปถึงกล้องวิดีโอสำหรับระบบการมองเห็นของหุ่นยนต์

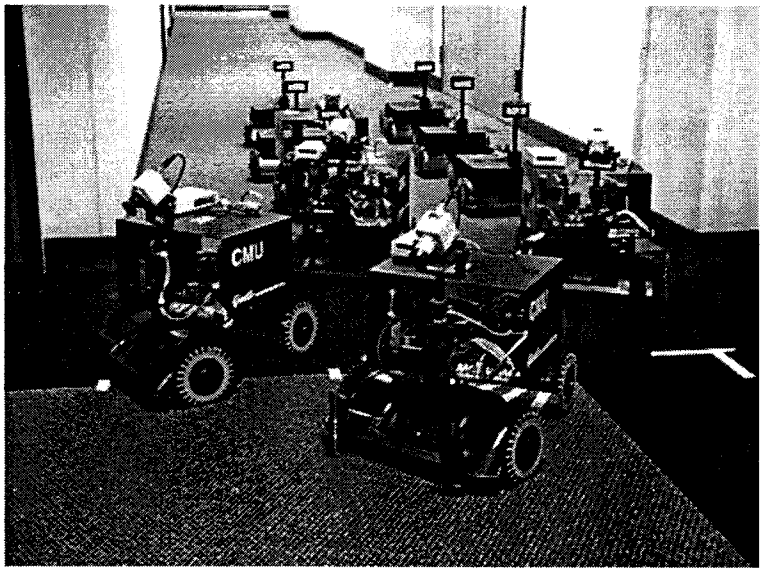


รูปที่ 2.5 หุ่นยนต์ Amelia (แหล่งที่มา: http://www-2.cs.cmu.edu/~Xavier/amelia.html)

นอกเหนือไปจากหุ่นยนต์ทั้งสองข้างต้นแล้ว ที่ Carnegie Mellon University ยังได้ทำการพัฒนาหุ่นยนต์เคลื่อนที่แบบต่างๆ โดยมีการวิจัยพัฒนาหุ่นยนต์ที่นำไปประยุกต์ใช้ทางด้านงานอดิเรกหรือการแข่งขันหุ่นยนต์ ยกตัวอย่างหุ่นยนต์ที่น่าสนใจเช่น

- CMUCam robot - ชุดคิทสำหรับประยุกต์ใช้งานด้านการมองเห็นของหุ่นยนต์

- Palm Pilot robot – ชุดคิทสำหรับสร้างหุ่นยนต์ที่ควบคุมด้วย Palm Pilot หรือ Gameboy หรือแม้แต่เครื่องคิดเลข (อ้างอิงของยี่ห้อ Texas Instrument)
- Lego rcx comm – ชุดสร้างหุ่นยนต์ด้วยตัวต่อ Lego ที่ซึ่งสามารถสื่อสารกันได้
- Robot emotion – การพัฒนาหุ่นยนต์ที่มีอารมณ์และความรู้สึก
- Minnow project – การพัฒนาหุ่นยนต์หลายๆ ตัวในการทำงานร่วมกันเป็นทีม มีการนำเอาหุ่นยนต์ไปร่วมแข่งขันในการแข่งขันหุ่นยนต์เตะฟุตบอลนานาชาติหรือ RoboCup (ดูรูปที่ 2.6 ประกอบ)



รูปที่ 2.6 ทีมหุ่นยนต์ในโครงการ Minnow (แหล่งที่มา: http://www-2.cs.cmu.edu/~coral/minnow)

6. หุ่นยนต์ Nomad

หุ่นยนต์ Nomad (ดูรูปที่ 2.7) เป็นโครงการวิจัยร่วมกันระหว่าง Carnegie Mellon University กับองค์การอวกาศ NASA โดยมีจุดหมายที่จะพัฒนาสร้างหุ่นยนต์เคลื่อนที่แบบอัตโนมัติสำหรับทำการสำรวจและเก็บตัวอย่างหินในพื้นที่ห่างไกลของแอนตาร์คติค (Antarctic) หุ่นยนต์ Nomad สามารถเคลื่อนที่ในพื้นผิวที่ไม่สามารถคาดเดาได้ ทำการสำรวจค้นหาและแยกแยะหินได้อย่างอัตโนมัติ โครงการนี้เป็นส่วนหนึ่งในการเริ่มต้นโครงการสำรวจดาวอังคารในเวลาต่อมา นอกไปจากนั้นแล้วผู้วิจัยยังเล็งเห็นที่จะพัฒนาระบบสำหรับการสำรวจภูมิประเทศ การค้นหาสิ่งมีชีวิตและการสำรวจขั้วโลกแบบอัตโนมัติ

หุ่นยนต์ Nomad ถูกออกแบบให้มีโครงสร้างที่มีประสิทธิภาพในการเคลื่อนที่ เนื่องมาจากข้อจำกัดของสภาพแวดล้อมในพื้นที่เป้าหมาย (แอนตาร์คติค) ล้อทั้งสี่ขับเคลื่อนโดยอิสระ ระบบกันสะเทือนแบบพิเศษ ระบบการควบคุมบังคับแบบอิเล็กทรอนิกส์ ด้วยโครงสร้างพิเศษต่างๆ เหล่านี้ทำให้หุ่นยนต์ Nomad สามารถเคลื่อนที่ไปได้ทุกสภาพพื้นผิว

ระบบคอมพิวเตอร์ควบคุมประกอบไปด้วยคอมพิวเตอร์สำหรับระบบควบคุมแบบเวลาจริง และคอมพิวเตอร์สำหรับการนำวิถีแบบอัตโนมัติ ระบบควบคุมแบบเวลาจริงทำหน้าที่ในการควบคุมหุ่นยนต์ให้เคลื่อนที่ไปตามข้อมูลจากระบบนำวิถีแบบอัตโนมัติ การควบคุมใช้ระบบ PID โดยการป้อนกลับของความเร็วถูกนำมาใช้ในการควบคุมมอเตอร์และการป้อนกลับของตำแหน่งถูกนำมาใช้ในการควบคุมระบบการเลี้ยวของหุ่นยนต์ ระบบเวลาจริงนี้ยังทำการประมวลผลข้อมูลจากตัวตรวจจับหลากหลายชนิด ไม่ว่าจะเป็นตัวถอดรหัส (encoder) ค่าตำแหน่งเทียบกับโลก (DGPS) หรือแม้แต่กระแสและแรงดันของแต่ละมอเตอร์เป็นต้น ระบบนำวิถีแบบอัตโนมัติทำการประมวลผลข้อมูลจากตัวตรวจจับที่เกี่ยวข้อง เช่นตัวตรวจจับแบบเลเซอร์สำหรับวัดระยะทาง ตัวตรวจจับสัญญาณภาพจากกล้องวิดีโอ ฯลฯ รายละเอียดต่างของโครงการนี้มีอยู่ใน

http://www.frc.ri.cmu.edu/projects/meteorobot2000

รูปที่ 2.7 หุ่นยนต์ Nomad ในการสำรวจพื้นที่ห่างไกลในแอนตาร์คติค (แหล่งที่มา: http://www.frc.ri.cmu.edu/projects/meteorobot2000)

7. หุ่นยนต์ MDOF

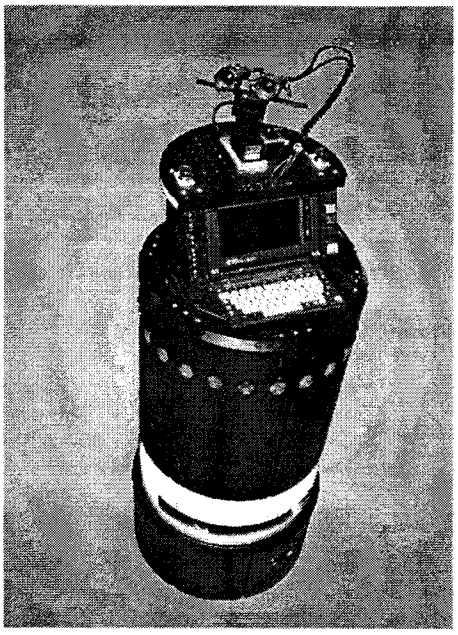
หุ่นยนต์ MDOF (ย่อมาจาก Multi-Degree-of-Freedom) สร้างและพัฒนาโดยห้องปฏิบัติการ Mobile Robotics Lab ของ University of Michigan จุดเด่นของหุ่นยนต์ MDOF คือความสามารถในการเคลื่อนที่และหมุนตัวไปยังทิศทางใดๆ ได้ในขณะเดียวกัน (ดูรูปที่ 2.8 ประกอบ) หุ่นยนต์ MDOF ยังถูกออกแบบให้มีโครงสร้างพิเศษ ทำให้หุ่นยนต์สามารถเคลื่อนที่หรือควบคุมจากระยะไกลได้ถึงแม้ว่ามอเตอร์ วงจรขยายกำลังหรือองค์ประกอบสำคัญส่วนอื่นๆ จะเสียหายก็ตาม หุ่นยนต์ MDOF

ถูกใช้ในงานวิจัยด้านต่างๆ เช่นการหลบหลีกสิ่งกีดขวาง ระบบนำวิถีอัตโนมัติ การระบุตำแหน่งของหุ่นยนต์ รวมไปถึงการพัฒนาหุ่นยนต์เพื่อใช้ในงานช่วยเหลือคนพิการอีกด้วย

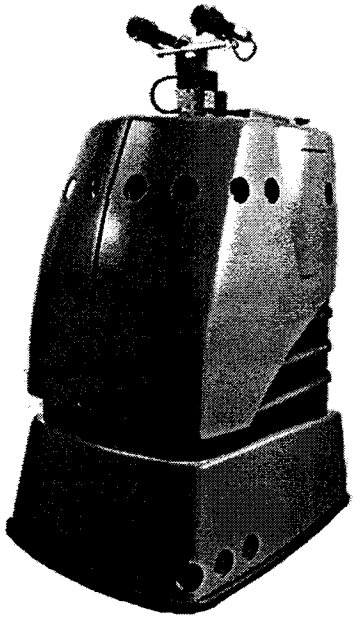
รูปที่ 2.8 หุ่นยนต์ MDOF (แหล่งที่มา: http://www.engin.umich.edu/research/mrl/00MoRob.html)

8. หุ่นยนต์ Rhino

หุ่นยนต์ Rhino (ดูรูปที่ 2.9 ประกอบ) เป็นหุ่นยนต์ไกด์นำทางภายในพิพิธภัณฑ์ พัฒนาร่วมกันระหว่าง Institut für Informatik III ของ Rhenische Friedrich-Wilhelms-Universität Bonn ประเทศเยอรมนี และ Carnegie Mellon University ประเทศสหรัฐอเมริกา หุ่นยนต์ Rhino เป็นหุ่นยนต์เคลื่อนที่ได้ภายในที่ราบในอาคาร สามารถนำทางและทำงานต่างๆ ได้อย่างอัตโนมัติ ส่วนฮาร์ดแวร์ของหุ่นยนต์ Rhino สร้างโดย Real World Interface Inc. ส่วนหลักๆ ประกอบไปด้วยคอมพิวเตอร์ควบคุม 3 เครื่อง เชื่อมต่อภายนอกด้วยการ์ดอินเตอร์เนตไร้สาย ติดตั้งตัวตรวจจับชนิดอัลตราโซนิค 24 ตัว ชนิดอินฟาเรด 56 ตัว ชนิดสัมผัส 24 ตัว เลเซอร์วัดระยะทาง 2 ตัวและกล้องวิดีโอแบบสเตอริโอ ปัจจุบันหุ่นยนต์ Rhino ทำงานอยู่ที่พิพิธภัณฑ์สถาน Deutsches Museum Bonn ในประเทศเยอรมนี นอกจากนั้นแล้ว ยังได้มีการพัฒนาหุ่นยนต์ Minerva ซึ่งเป็นการพัฒนาต่อจากหุ่นยนต์ Rhino โดยหุ่นยนต์ Minerva ทำงานอยู่ที่ Smithsonian National Museum of American History ในประเทศสหรัฐอเมริกา หุ่นยนต์ Minerva แสดงในรูปที่ 2.10



รูปที่ 2.9 หุ่นยนต์ไกด์นำทาง Rhino (แหล่งที่มา: http://www.informatik.uni-bonn.de/~rhino)

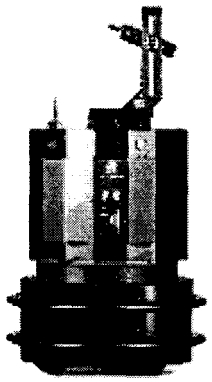


รูปที่ 2.10 หุ่นยนต์ไกด์นำทาง Minerva (แหล่งที่มา: http://www-2.cs.cmu.edu/~minerva)

9. หุ่นยนต์ Nomad 200

ห้องปฏิบัติการหุ่นยนต์ Stanford CS Robotics Lab ของ Stanford University ได้ทำการพัฒนาระบบหุ่นยนต์เคลื่อนที่สำหรับใช้ในการสังเกตการณ์ หุ่นยนต์ Nomad 200 (ดูรูปที่ 2.11) ที่ใช้ในงานวิจัยนี้เป็นผลิตภัณฑ์ที่ออกแบบและสร้างโดยบริษัท Nomadic Technologies ในมลรัฐแคลิฟอร์เนีย หุ่นยนต์ Nomad 200 มีขนาดความสูงประมาณ 120 เซนติเมตร เส้นผ่าศูนย์กลางขนาด 60 เซนติเมตร ตัวหุ่นยนต์ถูกติดตั้งตัวตรวจจับแบบชนิดอุลตราโซนิค 16 ชุด พร้อมทั้งระบบเลเซอร์

วัดระยะทาง นอกไปจากนั้นยังมีตัวตรวจจับชนิดอินฟาเรดอีก 16 ชุด อย่างไรก็ตาม ระบบการทำงานหลักของหุ่นยนต์ได้มาจากการประมวลผลสัญญาณภาพจากกล้องวิดีโอ 2 ตัว ที่ซึ่งทำหน้าที่แตกต่างกัน กล่าวคือกล้องวิดีโอตัวแรกทำหน้าที่ในการติดตามการเคลื่อนที่ของวัตถุ ในขณะที่กล้องวิดีโออีกตัวทำหน้าที่ตรวจจับและจดจำสภาวะแวดล้อมรอบๆ เส้นทางการเดินทาง บนตัวหุ่นยนต์มีคอมพิวเตอร์ชนิด Pentium 166 MHz สำหรับทำหน้าที่ควบคุมและประมวลผลต่างๆ หุ่นยนต์ติดต่อสื่อสารกับคอมพิวเตอร์หลักผ่านทางอินเตอร์เนตการ์ดชนิดไร้สาย



รูปที่ 2.11 หุ่นยนต์ Nomad 200 (แหล่งที่มา: http://underdog.stanford.edu/robots.html)

10. สรุป

ในปัจจุบันได้มีหน่วยงาน องค์กรและห้องปฏิบัติการต่างๆ เป็นจำนวนมากที่ได้พัฒนาเทคโนโลยีทางด้านหุ่นยนต์เคลื่อนที่ ซึ่งมีทั้งหุ่นยนต์ที่ออกแบบและสร้างขึ้นภายในห้องปฏิบัติการเอง และทั้งที่นำเอาหุ่นยนต์ผลิตภัณฑ์เชิงการค้ามาดัดแปลงประยุกต์ให้เข้ากับงานวิจัย การออกแบบและสร้างหุ่นยนต์เองนั้นมีข้อดีที่จะได้หุ่นยนต์ที่มีคุณสมบัติตรงตามจุดประสงค์ในการใช้งาน ในขณะที่การใช้ผลิตภัณฑ์หุ่นยนต์เชิงการค้ามาประยุกต์ใช้งานนั้นสามารถลดเวลาในการพัฒนาสร้างหุ่นยนต์เอง แต่ต้นทุนที่ใช้อาจจะค่อนข้างสูงได้ โดยเฉพาะในประเทศไทยที่ซึ่งไม่มีบริษัทเอกชนใดๆ ที่พัฒนาเทคโนโลยีหุ่นยนต์ในระดับงานวิจัยออกมาจำหน่ายแต่อย่างใด

ตัวอย่างหุ่นยนต์เคลื่อนที่ที่นำเสนอข้างต้นเป็นเพียงส่วนหนึ่งเท่านั้น ยังมีหุ่นยนต์ตามห้องปฏิบัติการและบริษัทผู้ผลิตอีกมากมายที่ยังไม่ได้กล่าวถึง อย่างไรก็ตามหุ่นยนต์เหล่านี้มีคุณสมบัติที่เป็นที่ต้องการร่วมกันของผู้ต้องการประยุกต์ใช้หุ่นยนต์ทั่วๆ ไปดังนี้

- มีการเชื่อมต่อเพื่อการติดต่อและควบคุมแบบไร้สาย
- มีการติดตั้งตัวตรวจจับแบบต่างๆ
- มีหน่วยประมวลผลระดับสูงบนตัวหุ่นยนต์

คุณสมบัติข้างต้นเป็นเพียงคุณสมบัติโดยรวม รายละเอียดของคุณสมบัติต่างๆ เหล่านี้จะได้ถูกวิเคราะห์และกล่าวถึงในบทต่อไป

บทที่ 3
การออกแบบและสร้างหุ่นยนต์เคลื่อนที่ GUNDAM

1. บทนำ

ในบทนี้จะได้กล่าวถึงรายละเอียดในการออกแบบและสร้างหุ่นยนต์เคลื่อนที่ในงานวิจัยนี้ รายละเอียดการออกแบบจะคำนึงถึงคุณสมบัติโดยรวมของหุ่นยนต์เคลื่อนที่ดังต่อไปนี้

ชื่อหุ่นยนต์ (Codename) : GUNDAM

- มีการเชื่อมต่อเพื่อการติดต่อและควบคุมแบบไร้สาย
- มีการติดตั้งตัวตรวจจับด้วยภาพจากกล้องวิดีโอ
- มีหน่วยประมวลผลระดับสูง

เพื่อความสะดวกในการอธิบายรายละเอียด ชื่อของหุ่นยนต์เคลื่อนที่ที่พัฒนาในงานวิจัยนี้จะเรียกว่า GUNDAM ในงานวิจัยสร้างหุ่นยนต์เคลื่อนที่นี้ ได้เริ่มทำการออกแบบสร้างในปีพ.ศ. 2543 ซึ่งเทคโนโลยีต่างๆ ในการพัฒนางานทางด้านหุ่นยนต์ยังถือว่าไม่พร้อมและแพร่หลายมากนัก เทคโนโลยีหลายๆ อย่างเองยังมีราคาที่ค่อนข้างสูง การออกแบบหุ่นยนต์ GUNDAM ในงานวิจัยนี้สามารถแบ่งการพัฒนาออกได้เป็น 3 ช่วงดังรายละเอียดต่อไปนี้

- GUNDAM I (พ.ศ. 2543) – ถือเป็นช่วงแรกในการพัฒนาฮาร์ดแวร์ของ GUNDAM โดยเป้าหมายเน้นไปที่การควบคุมแบบไร้สายจากเครื่องคอมพิวเตอร์ส่วนบุคคล พร้อมทั้งสามารถส่งภาพจากกล้องวิดีโอที่ติดตั้งบนตัว GUNDAM เข้าสู่คอมพิวเตอร์เพื่อทำการประมวลผลในระดับสูงได้
- GUNDAM II (พ.ศ. 2544) – หลังจากโครงสร้างฮาร์ดแวร์หลักๆ ของ GUNDAM ได้รับการสร้างแล้ว การทดสอบการประมวลผลภาพบนตัว GUNDAM จึงได้รับการพัฒนาในช่วงเวลาต่อมา ในช่วงนี้การพัฒนาส่วนใหญ่จะเป็นทางด้านซอฟท์แวร์ โดยเน้นไปที่ซอฟท์แวร์การมองเห็นของหุ่นยนต์
- GUNDAM III (พ.ศ. 2545 – ปัจจุบัน) – ในช่วงเวลานี้ เทคโนโลยีต่างๆ ที่เกี่ยวข้องกับการพัฒนาหุ่นยนต์ได้มีการปรับปรุงไปอย่างมากมาย GUNDAM III จึงเป็นช่วงการปรับปรุงทดลองใช้เทคโนโลยีใหม่ๆ ทางด้านฮาร์ดแวร์ ได้แก่ตัวส่งข้อมูลแบบไร้สายและบอร์ดไมโครคอนโทรลเลอร์สำหรับการควบคุม

สำหรับหุ่นยนต์ที่ทำการพัฒนาในงานวิจัยนี้มีขอบเขตข้อจำกัดว่าเป็นหุ่นยนต์เคลื่อนที่ในที่ราบ โดยใช้ล้อในการขับเคลื่อนเป็นหลัก มีการควบคุมแบบไร้สายและสามารถส่งภาพแบบไร้สายจากกล้องวิดีโอบนตัวหุ่นยนต์ไปประมวลผลที่คอมพิวเตอร์ได้ รายละเอียดการพัฒนาหุ่นยนต์ในช่วงต่างๆ มีดังต่อไปนี้

2. GUNDAM I

การออกแบบและพัฒนาสร้างหุ่นยนต์ GUNDAM ในช่วงแรกนี้ประกอบไปด้วยการพัฒนาใน 2 ส่วนใหญ่ๆ ดังนี้

- การออกแบบโครงสร้างทางกายภาพ
- การออกแบบส่วนประกอบในการทำงานต่างๆ

2.1 การออกแบบโครงสร้างของหุ่นยนต์ GUNDAM I

ในหัวข้อนี้จะกล่าวถึงรายละเอียดในการออกแบบโครงสร้าง ปัจจัยสำคัญที่ต้องพิจารณาในการออกแบบโครงสร้างหุ่นยนต์ ส่วนประกอบทางโครงสร้างต่างๆ ของหุ่นยนต์ ดังรายละเอียดต่อไปนี้

2.1.1 ปัจจัยสำคัญที่ต้องพิจารณาในการออกแบบโครงสร้างหุ่นยนต์

ในหัวข้อนี้จะกล่าวถึงปัจจัยสำคัญทางโครงสร้างทั่วๆ ไปที่ต้องพิจารณา ปัจจัยเหล่านี้เป็นปัญหาพื้นฐานที่จะต้องคำนึงถึงเสมอในการออกแบบทางโครงสร้างของหุ่นยนต์ โดยมีปัจจัยที่น่าสนใจหลักๆ อยู่ 4 ข้อดังต่อไปนี้

- ความเที่ยงตรงในการขึ้นรูปชิ้นส่วน

สำหรับปัญหานี้เป็นปัญหาที่สำคัญมาก เนื่องจากในการประกอบชิ้นส่วนต่างๆ เข้าด้วยกันเป็นตัวหุ่นยนต์นั้น ชิ้นส่วนทุกๆ ชิ้นจะต้องได้ขนาดถูกต้องตามแบบที่ได้ทำการออกแบบไว้ จึงจะทำให้ส่วนประกอบต่างๆ ทำงานสัมพันธ์กันอย่างมีประสิทธิภาพ หากว่ามีความผิดพลาดเกิดขึ้นในชิ้นส่วนต่างๆ ที่นำมาประกอบกัน อาจจะมีผลให้การประกอบชิ้นส่วนต่างๆ ทำได้ยากหรืออาจจะทำไม่ได้เลยก็ได้

- ความบิดเบี้ยวทางโครงสร้าง

เกิดจากการยึดรั้งชิ้นส่วนต่างๆ เข้าด้วยกัน ทำให้โครงสร้างเกิดบิดเบี้ยวผิดรูปไป ซึ่งจะมีผลโดยตรงในการทำให้การควบคุมหุ่นยนต์มีความยุ่งยากและขาดความเที่ยงตรง ปัญหานี้จะต้องแก้ไขตั้งแต่ในขั้นตอนของการออกแบบทางโครงสร้าง เพื่อไม่ให้มีการยึดรั้งชิ้นส่วนต่างๆ ของหุ่นยนต์ในลักษณะที่จะมีผลทำให้โครงสร้างบิดเบี้ยวจนมีปัญหายุ่งยากตามมาในภายหลัง

- การจัดสรรพื้นที่ในการติดตั้งอุปกรณ์

ในส่วนของการจัดสรรพื้นที่ในการติดตั้งอุปกรณ์นั้น ในกรณีที่ตัวหุ่นยนต์มีขนาดที่เล็กลงมากๆ การจัดสรรพื้นที่ในการติดตั้งอุปกรณ์จะมีความยุ่งยากมาก ดังนั้นในการออกแบบตัวหุ่นยนต์ควรจะต้องพิจารณาออกแบบหุ่นยนต์ให้สามารถที่จะติดตั้งอุปกรณ์ทุกชิ้นลงบนตัวหุ่นได้ ทั้งนี้จะต้องคำนึงถึงผลของสัญญาณรบกวนระหว่างอุปกรณ์ที่ติดตั้งลงบนตัวหุ่นยนต์ด้วย

- สัญญาณรบกวนระหว่างอุปกรณ์

สำหรับปัญหาของสัญญาณรบกวนระหว่างอุปกรณ์นั้น จะมีผลมากกับอุปกรณ์ที่ทำงานที่ความถี่สูงๆ ซึ่งอาจจะมีผลทำให้อุปกรณ์ดังกล่าวทำงานได้ไม่ดีเท่าที่ควร ในทางปฏิบัติแล้วจะยอมให้มีการรบกวนของสัญญาณรบกวนดังกล่าวเกิดขึ้นได้ในระดับหนึ่งที่ยอมรับได้

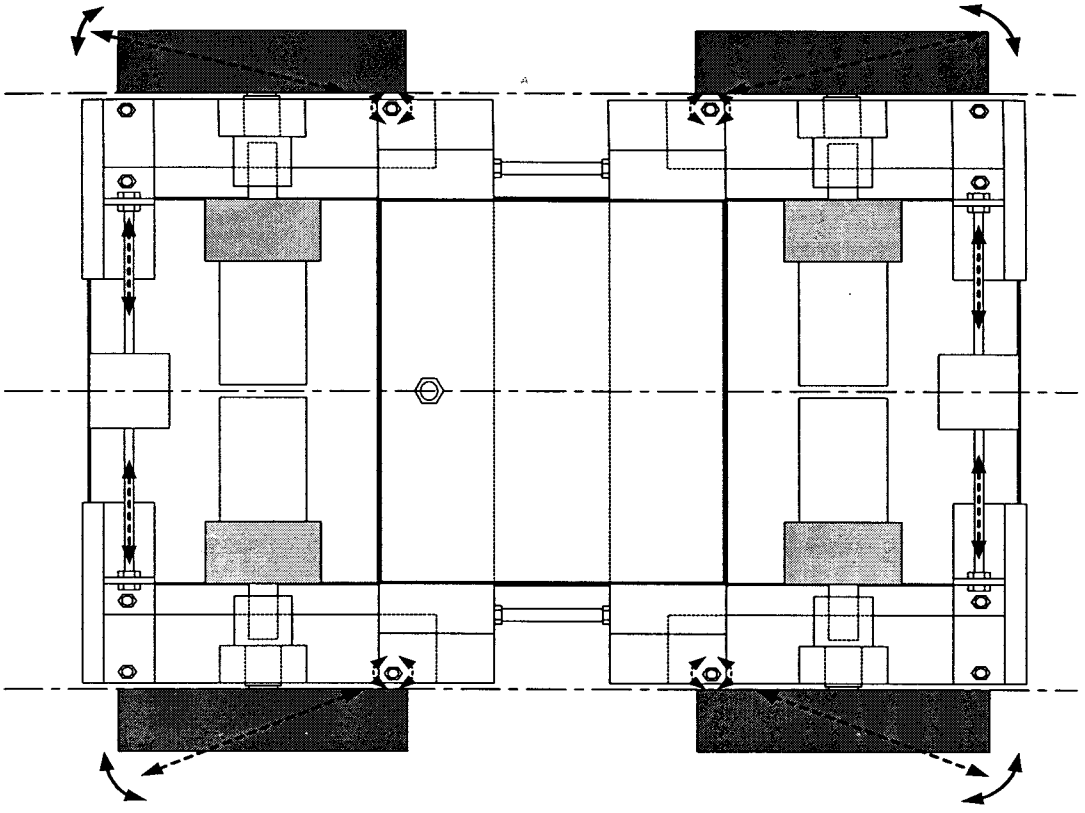
2.1.2 การออกแบบระบบการปรับศูนย์ทางโครงสร้างของหุ่นยนต์

ในการออกแบบทางโครงสร้างของหุ่นยนต์ GUNDAM นั้น ผู้ออกแบบได้ตระหนักและให้ความสำคัญกับความคลาดเคลื่อนทางโครงสร้างต่างๆ ที่จะเกิดขึ้น เนื่องจากความคลาดเคลื่อนต่างๆ ดังกล่าว จะมีผลโดยตรงต่อความเที่ยงตรงของโปรแกรมควบคุมหุ่นยนต์ ซึ่งจะมีผลให้โปรแกรมควบคุมหุ่นยนต์มีความซับซ้อนและยุ่งยากเกินไป จนในที่สุดกลายเป็นความยุ่งยากโดยไม่จำเป็น และเพื่อลดผลจากความคลาดเคลื่อนต่างๆ ที่ไม่ต้องการดังกล่าว จึงจำเป็นจะต้องออกแบบโครงสร้างของหุ่นให้มีความยืดหยุ่นสูง สามารถที่จะชดเชยผลของความคลาดเคลื่อนต่างๆ ทางโครงสร้างที่ไม่ต้องการให้มีผลต่อประสิทธิภาพทางโครงสร้างน้อยที่สุด โดยที่ยังคงความแข็งแรงทางโครงสร้างให้มากที่สุด

จากเหตุผลที่ต้องการให้โครงสร้างของหุ่นยนต์สามารถที่จะถูกควบคุมได้โดยง่าย อีกทั้งยังต้องการให้โครงสร้างของหุ่นยนต์มีความยืดหยุ่น สามารถที่จะทนทานต่อความคลาดเคลื่อนและความบิดเบี้ยวต่างๆ ทางโครงสร้างได้ ซึ่งจากความต้องการทางโครงสร้างดังกล่าว ทำให้ต้องออกแบบโครงสร้างให้สามารถปรับศูนย์ทางโครงสร้างได้ ซึ่งวิธีนี้จะทำให้โครงสร้างของหุ่นยนต์มีความยืดหยุ่น สามารถที่จะทนทานต่อความคลาดเคลื่อนและความบิดเบี้ยวต่างๆ ทางโครงสร้างได้ ช่วยให้การเคลื่อนที่ในแนวเส้นตรงของหุ่นยนต์เป็นไปอย่างมีประสิทธิภาพ ทำให้โครงสร้างหุ่นยนต์สามารถที่จะถูกควบคุมโดยโปรแกรมควบคุมได้โดยง่ายและมีประสิทธิภาพ ช่วยลดความยุ่งยากซับซ้อนของโปรแกรมควบคุมหุ่นยนต์ ซึ่งมีผลให้หุ่นยนต์สามารถถูกควบคุมได้ง่าย ไม่ว่าจะถูกควบคุมโดยมนุษย์หรือโดยโปรแกรมคอมพิวเตอร์ก็ตาม ซึ่งในรายละเอียดของระบบการปรับศูนย์ทางโครงสร้างจะมีดังต่อไปนี้

- ระบบปรับศูนย์ล้อ 4 ล้อแยกอิสระ

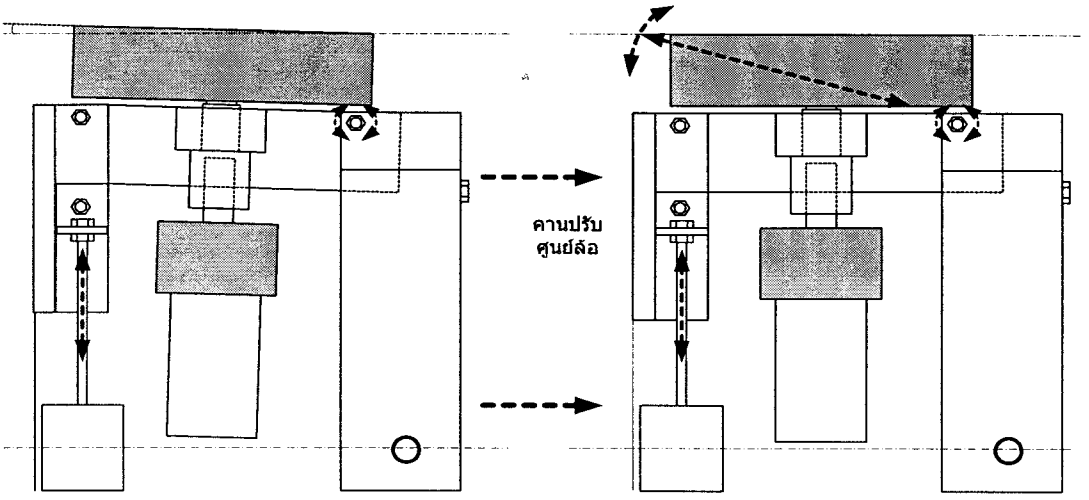
ระบบนี้ออกแบบสำหรับการแก้ปัญหาความคลาดเคลื่อนของการขึ้นรูปชิ้นส่วนในระบบขับเคลื่อน หากว่าความคลาดเคลื่อนดังกล่าวมีไม่มากนัก ก็จะสามารถปรับศูนย์ล้อแต่ละล้อเพื่อชดเชยกับความคลาดเคลื่อนที่เกิดขึ้นในแต่ละล้อของหุ่นยนต์ได้



รูปที่ 3.1 ระบบปรับศูนย์ล้อ 4 ล้อ

จากรูปที่ 3.1 จะเห็นได้ว่าเราสามารถแยกปรับตั้งศูนย์ล้อของหุ่นยนต์ทั้ง 4 ล้อทีละล้ออิสระจากกัน ในการปรับตั้งศูนย์ล้อจะต้องปรับตั้งให้ล้อทั้ง 4 ล้ออยู่ในแนวที่เหมาะสมทุกล้อ กล่าวคือแนวขอบในของล้อทุกล้อจะต้องอยู่ในแนวขนานกับแนวศูนย์กลางของโครงหุ่นยนต์ เพื่อให้แนวแรงของการขับเคลื่อนจากล้อทุกล้ออยู่ในแนวเดียวกัน ทำให้การการขับเคลื่อนของหุ่นยนต์ในแนวเส้นตรงสามารถทำได้อย่างมีประสิทธิภาพมากขึ้น ซึ่งจะช่วยลดความยุ่งยาก ในการเขียนโปรแกรมควบคุมการเคลื่อนที่ของหุ่นยนต์ได้อย่างมาก

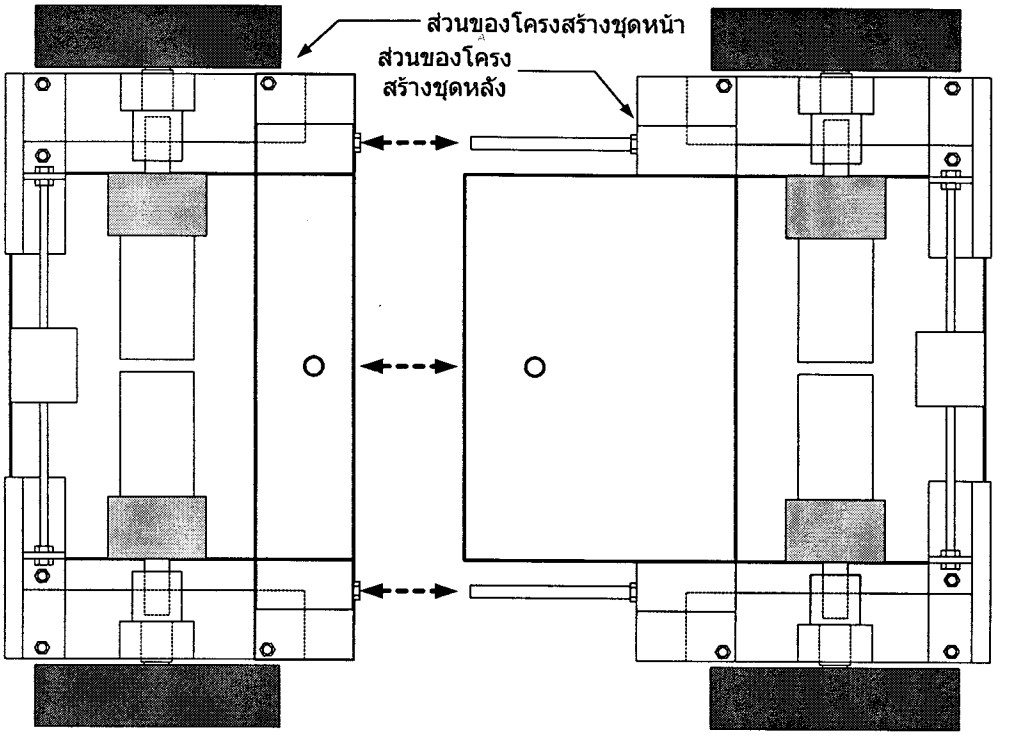
จากรูปที่ 3.2 ซึ่งจะเห็นว่าในระบบการปรับศูนย์ล้อนั้น จะประกอบด้วยคานปรับศูนย์ล้อ ซึ่งจะเป็นส่วนที่ทำหน้าที่ในการปรับเลื่อนศูนย์ล้อและทำหน้าที่รักษาความเที่ยงตรงของศูนย์ล้อที่ผ่านการปรับศูนย์แล้ว



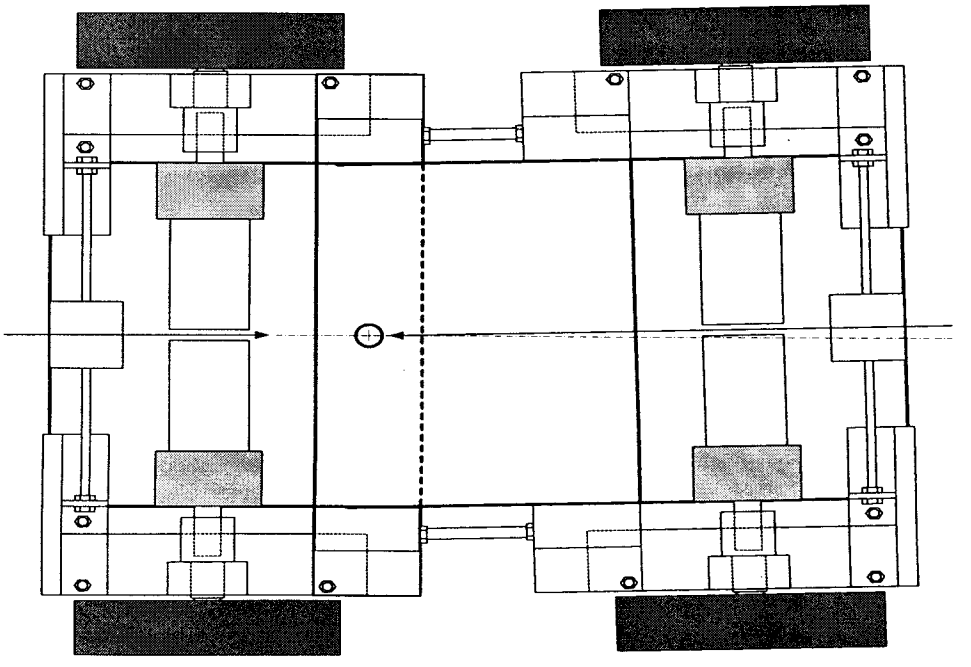
รูปที่ 3.2 การปรับศูนย์ล้อ

- ระบบปรับศูนย์ของโครงสร้าง

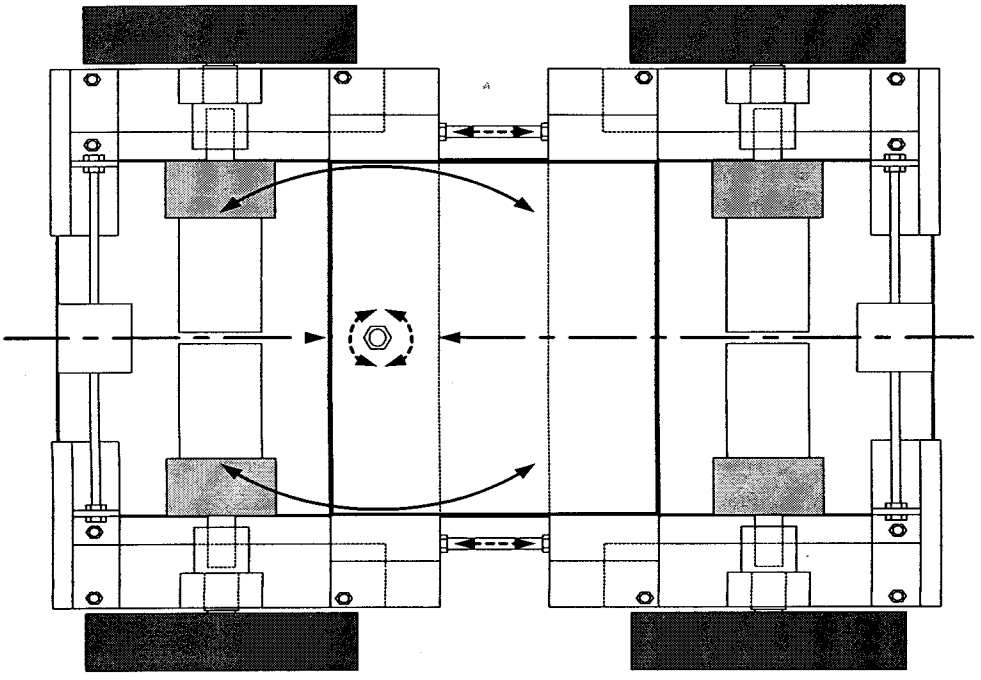
ระบบนี้ออกแบบสำหรับแก้ไขปัญหาการบิดเบี้ยวของโครงสร้าง ซึ่งเป็นระบบเสริมจากระบบการปรับศูนย์ล้อ 4 ล้อ ในการปรับศูนย์ล้อของหุ่นยนต์นั้นจะช่วยให้หุ่นยนต์สามารถเคลื่อนที่ในแนวเส้นตรงได้ระดับหนึ่งเท่านั้น เนื่องจากยังมีความบิดเบี้ยวของโครงสร้างอยู่ ดังนั้นจึงมีความจำเป็นที่จะต้องมีระบบเสริมขึ้นมา เพื่อแก้ไขปัญหาของการบิดเบี้ยวของโครงสร้างที่เกิดขึ้น ในระบบการปรับศูนย์ของโครงสร้างนั้น ระบบจะแยกส่วนของโครงสร้างออกเป็น 2 ส่วน คือ ส่วนของโครงสร้างชุดหน้าและส่วนของโครงสร้างชุดหลัง ดังแสดงในรูปที่ 3.3 ซึ่งจะเห็นได้ว่า ในการปรับศูนย์ล้อนั้นจะต้องปรับศูนย์ล้อหน้าให้สัมพันธ์กับศูนย์ของโครงสร้างชุดหน้า ส่วนศูนย์ล้อหลังนั้นจะต้องปรับให้สัมพันธ์กับศูนย์ของโครงสร้างชุดหลังเช่นกัน ดังนั้น เมื่อนำโครงสร้างชุดหน้าและชุดหลังมาประกอบเข้าด้วยกันแล้วดังรูปที่ 3.4 จะเห็นว่าระบบศูนย์ของชุดโครงสร้างทั้ง 2 ชุดจะยังไม่ตรงกัน ซึ่งจะต้องมีการปรับศูนย์ของโครงสร้างทั้ง 2 ชุดให้อยู่ในระดับเดียวกันและตรงกันดังรูปที่ 3.5



รูปที่ 3.3 ส่วนของโครงสร้างชุดหน้าและชุดหลัง



รูปที่ 3.4 การประกอบโครงสร้างชุดหน้าและชุดหลังเข้าด้วยกัน



รูปที่ 3.5 ระบบการปรับศูนย์โครงสร้าง

2.1.3 การจัดสรรพื้นที่ในการติดตั้งอุปกรณ์

ในส่วนของการจัดสรรพื้นที่ในการติดตั้งอุปกรณ์นั้น จะเป็นการออกแบบในการจัดหาตำแหน่งที่เหมาะสมที่สุดบนตัวหุ่นยนต์ให้กับอุปกรณ์แต่ละชิ้นที่จะต้องติดตั้งบนตัวหุ่นยนต์ ซึ่งในที่นี้จะใช้หลักในการพิจารณาออกแบบเบื้องต้นอยู่ 3 ประการ ดังต่อไปนี้

- ขนาดรูปร่างของหุ่นยนต์

ในเรื่องของขนาดรูปร่างของหุ่นยนต์นั้น จะเป็นข้อจำกัดหลักที่ทำให้ต้องมีการออกแบบจัดสรรพื้นที่ในการติดตั้งอุปกรณ์ต่างๆ ซึ่งหุ่นยนต์ที่มีขนาดเล็ก แต่มีอุปกรณ์ที่จะต้องติดตั้งลงบนตัวหุ่นยนต์มาก ก็จะยิ่งทำให้การออกแบบติดตั้งอุปกรณ์ต่างๆ เหล่านั้นมีความยุ่งยากขึ้น ทั้งนี้ยังต้องคำนึงถึงรูปร่างของหุ่นยนต์ ที่จะได้หลังจากติดตั้งอุปกรณ์ต่างๆ ลงบนตัวหุ่นยนต์แล้วว่าจะยังคงรูปร่างของหุ่นยนต์ที่ได้ทำการออกแบบไว้ในตอนแรกหรือไม่

- ขนาดของอุปกรณ์ที่จะนำมาติดตั้ง

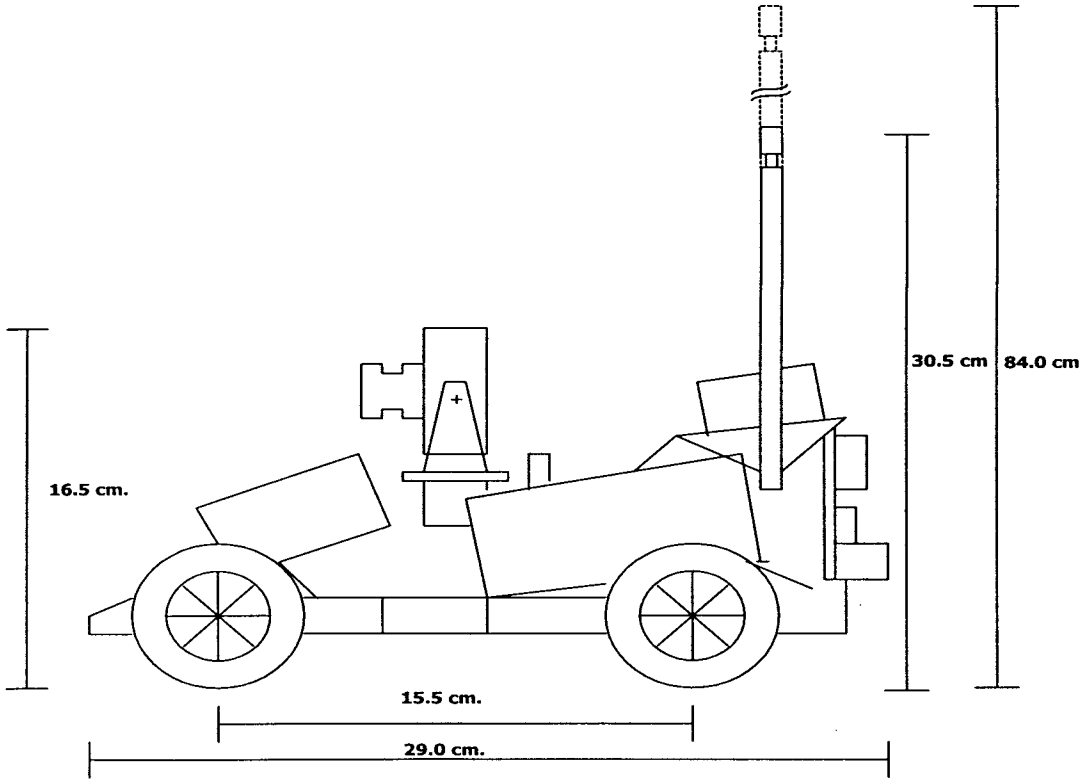
ในส่วนของขนาดของอุปกรณ์ที่จะนำมาติดตั้งนั้น จะต้องมีขนาดที่เหมาะสมกับตัวหุ่นยนต์นั้นคือ หากออกแบบให้หุ่นยนต์มีขนาดเล็ก อุปกรณ์ที่จะนำมาติดตั้งนั้นก็จะต้องมีขนาดเล็กด้วยเช่นกัน โดยในการออกแบบจัดสรรพื้นที่ในการติดตั้งอุปกรณ์นั้น มีความจำเป็นที่จะต้องทราบขนาดของอุปกรณ์ต่างๆ ที่จะนำมาติดตั้งทุกอุปกรณ์ก่อน เพื่อที่จะทำให้การออกแบบจัดสรรพื้นที่ในการติดตั้งอุปกรณ์สามารถทำได้อย่างถูกต้อง

- สัญญาณรบกวนระหว่างอุปกรณ์ที่จะนำมาติดตั้ง

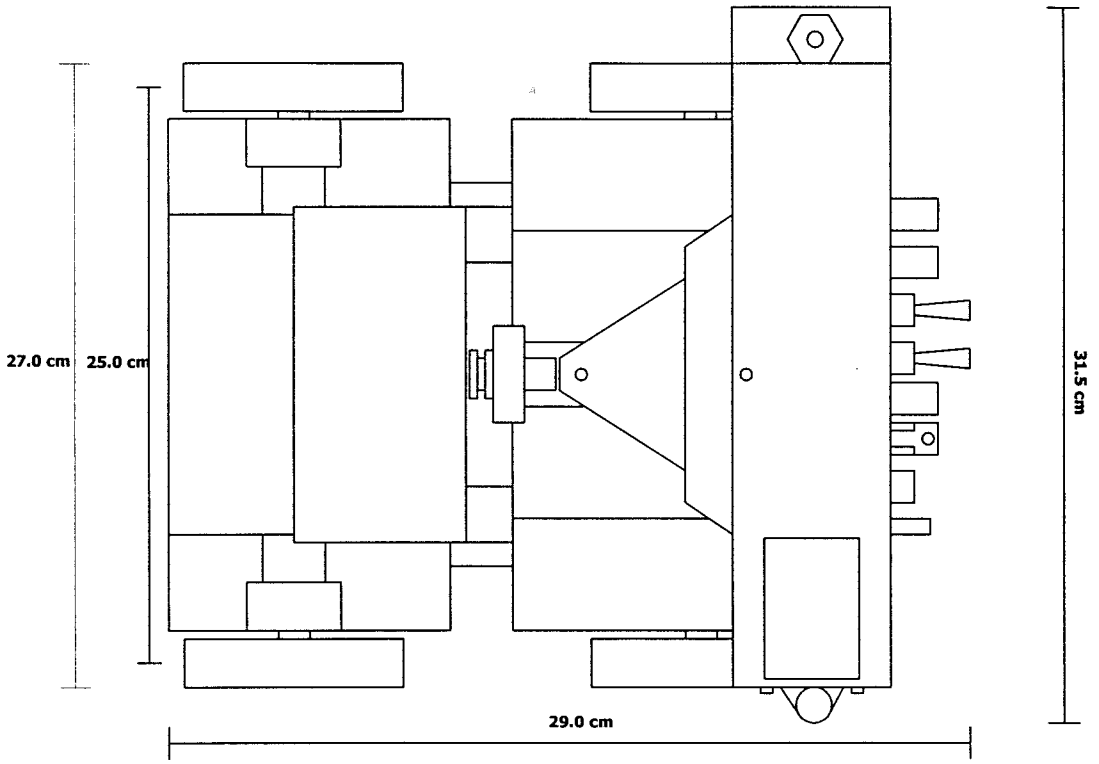
ในเรื่องของสัญญาณรบกวนระหว่างอุปกรณ์ที่จะนำมาติดตั้งนั้น จะนำมาพิจารณาเพื่อให้การออกแบบติดตั้งอุปกรณ์นั้นๆ ไม่เอื้อให้เกิดปัญหาสัญญาณรบกวนระหว่างอุปกรณ์ ทั้งนี้เนื่องมาจากตำแหน่งในการติดตั้งอุปกรณ์นั้น จะมีผลช่วยลดหรือบรรเทาปัญหาสัญญาณรบกวนระหว่างอุปกรณ์ หากอุปกรณ์แต่ละชิ้นถูกติดตั้งในตำแหน่งที่เหมาะสม และจะไม่มีผลช่วยลดปัญหานี้เลย หากว่าอุปกรณ์ถูกติดตั้งในตำแหน่งที่ไม่เหมาะสม

2.1.4 โครงสร้างโดยรวมของหุ่นยนต์ GUNDAM

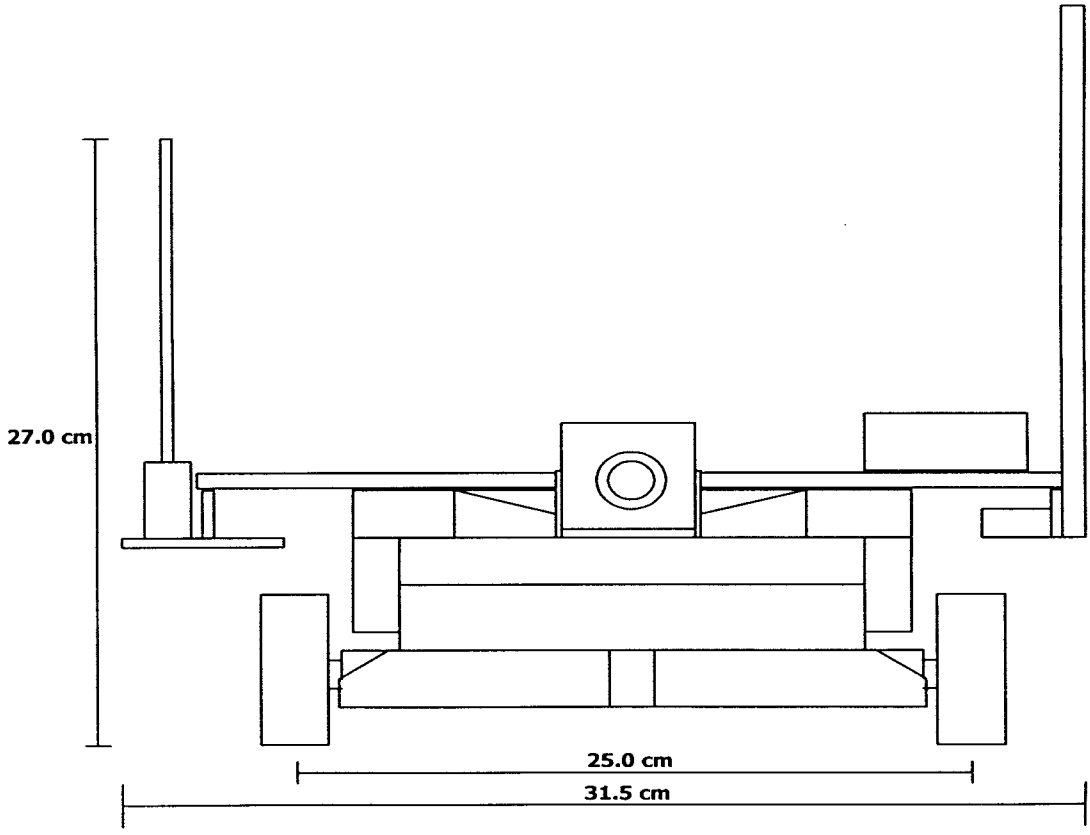
จากรายละเอียดต่างๆ ที่ได้พิจารณามาแล้วในข้างต้น จะเห็นได้ว่าในขั้นตอนของการออกแบบทางโครงสร้างของหุ่นยนต์นั้น จะต้องมีความละเอียดรอบคอบทางด้านโครงสร้างเป็นอย่างมาก อีกทั้งยังต้องตระหนักถึงปัญหาต่างๆ ที่อาจจะเกิดขึ้น และออกแบบให้โครงสร้างสามารถที่จะแก้ไขหรือป้องกันปัญหาต่างๆ ดังกล่าวได้ อย่างไรก็ตาม ในหัวข้อนี้จะกล่าวถึงลักษณะของหุ่นยนต์ GUNDAM จากรูปที่ 3.6รูปที่ 3.7 และรูปที่ 3.8 จะเห็นลักษณะคร่าวๆ ของหุ่นยนต์ ซึ่งรายละเอียดต่างๆ มีแสดงอยู่ในตารางที่ 3.1



รูปที่ 3.6 ลักษณะของหุ่นยนต์ GUNDAM เมื่อมองจากด้านข้าง



รูปที่ 3.7 ลักษณะของหุ่นยนต์ GUNDAM เมื่อมองจากด้านบน



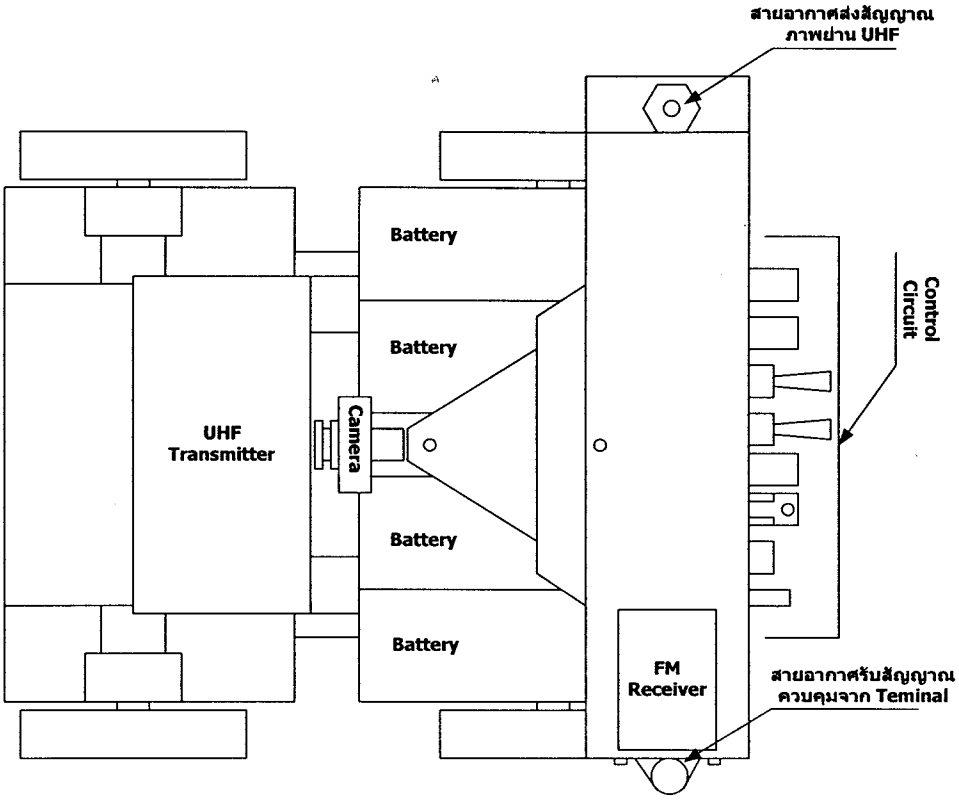
รูปที่ 3.8 ลักษณะของหุ่นยนต์ GUNDAM เมื่อมองจากด้านหน้า

ลักษณะของหุ่นยนต์ GUNDAM มองจากทางด้านข้าง	เซนติเมตร
ด้านข้างยาว	29.0
ระยะห่างระหว่างฐานล้อหน้า - หลัง	15.5
ความสูงรวมเมื่อสายอากาศยืดสุด	84.0
ความสูงรวมเมื่อสายอากาศหดสุด	30.5
ความสูงของตัวหุ่นยนต์ เมื่อไม่รวมความสูงของสายอากาศ	16.5
ลักษณะของหุ่นยนต์ GUNDAM มองจากทางด้านบน	เซนติเมตร
ด้านหน้ากว้าง	31.5
ระยะฐานล้อ	25.0
ลักษณะของหุ่นยนต์ GUNDAM มองจากทางด้านหน้า	เซนติเมตร
ความสูงสายอากาศส่งสัญญาณภาพ	27.0
ความสูงสายอากาศรับสัญญาณควบคุมจากคอมพิวเตอร์หดสุด / ยืดสุด	30.5 / 84.0

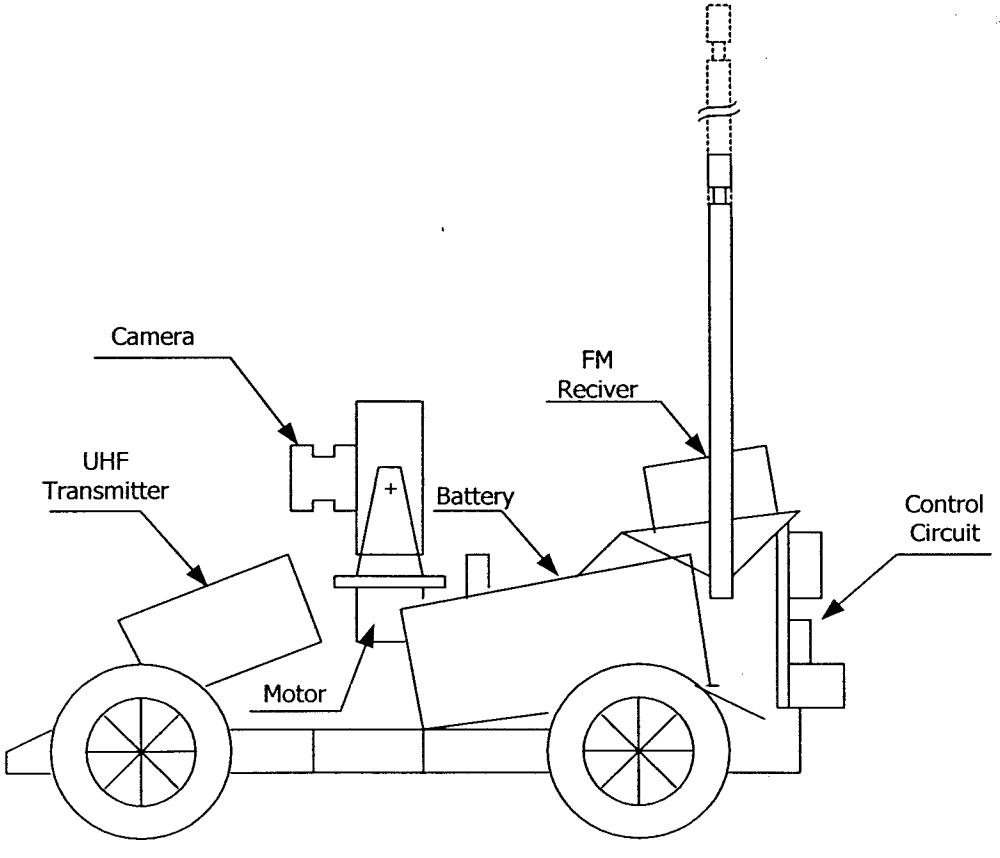
ตารางที่ 3.1 ลักษณะต่างๆ ทางโครงสร้างของหุ่นยนต์ GUNDAM

2.2 การออกแบบส่วนประกอบในการทำงานต่างๆ

จากรูปที่ 3.9 และรูปที่ 3.10 จะเห็นว่าหุ่นยนต์ประกอบด้วยส่วนประกอบต่างๆ ซึ่งจะกล่าวถึงเฉพาะส่วนประกอบที่สำคัญๆ ดังต่อไปนี้



รูปที่ 3.9 ส่วนประกอบต่างๆ ของหุ่นยนต์ GUNDAM



รูปที่ 3.10 ส่วนประกอบต่างๆ ของหุ่นยนต์ GUNDAM มองจากด้านข้าง

2.2.1 อุปกรณ์ส่งสัญญาณภาพย่าน UHF (UHF Transmitter)

ใช้สำหรับส่งสัญญาณภาพที่รับมาจากกล้องถ่ายภาพ เพื่อส่งไปให้คอมพิวเตอร์เทอร์มินัล (terminal) ซึ่งเป็นภาคควบคุมหุ่นยนต์ให้สามารถมองเห็นเส้นทางข้างหน้าและรอบๆ ตัวหุ่นยนต์ได้ โดยจากรูปที่ 3.9 และรูปที่ 3.10 จะเห็นได้ว่าอุปกรณ์ดังกล่าวถูกติดตั้งไว้เหนือชุดมอเตอร์ขับเคลื่อนของล้อหน้าทางด้านหน้าหุ่นยนต์ โดยที่ระหว่างชุดมอเตอร์ขับเคลื่อนของล้อหน้ากับอุปกรณ์ดังกล่าว จะมีฝาครอบอลูมิเนียมปิดครอบชุดมอเตอร์อยู่ เพื่อป้องกันและลดสัญญาณรบกวนจากการทำงานของชุดมอเตอร์ขับเคลื่อน ซึ่งอาจจะมีผลไปรบกวนการทำงานของอุปกรณ์ส่งสัญญาณภาพ ทำให้ภาพที่คอมพิวเตอร์เทอร์มินัลมีคุณภาพที่ลดลงได้

2.2.2 กล้องถ่ายภาพ (Video Camera)

เป็นอุปกรณ์ที่ทำหน้าที่ถ่ายภาพและนำสัญญาณภาพที่ได้ป้อนให้กับอุปกรณ์ส่งสัญญาณภาพ เพื่อส่งสัญญาณภาพไปให้กับคอมพิวเตอร์เทอร์มินัลต่อไป จากรูปที่ 3.9 และรูปที่ 3.10 จะเห็นว่ากล้องถ่ายภาพจะถูกติดตั้งไว้ในตำแหน่งตรงกลางของหุ่นยนต์ ซึ่งการออกแบบติดตั้งที่ตำแหน่งดังกล่าวเพื่อให้กรณีที่หุ่นยนต์หมุนตัว จะได้ว่าความเร็วที่กล้องหมุนเท่ากับความเร็วที่หุ่นยนต์หมุน ทำให้คอมพิวเตอร์เห็นความเร็วของภาพที่หมุนไปเท่ากับความเร็วจริงของหุ่นยนต์ที่หมุนอยู่ ซึ่งทำให้ภาพที่เห็นมีความสมจริงมากกว่าที่จะติดตั้งกล้องไว้ที่ตำแหน่งอื่นๆ อย่างไรก็ตาม ตัวกล้องจะถูกออกแบบให้ติดตั้งไว้บนมอเตอร์ทำให้สามารถที่จะถ่ายภาพทางด้านข้างของหุ่นยนต์ทั้งสองข้างได้โดยที่หุ่นยนต์ไม่ต้องหมุนตัวเลย

2.2.3 แบตเตอรี่ (Battery)

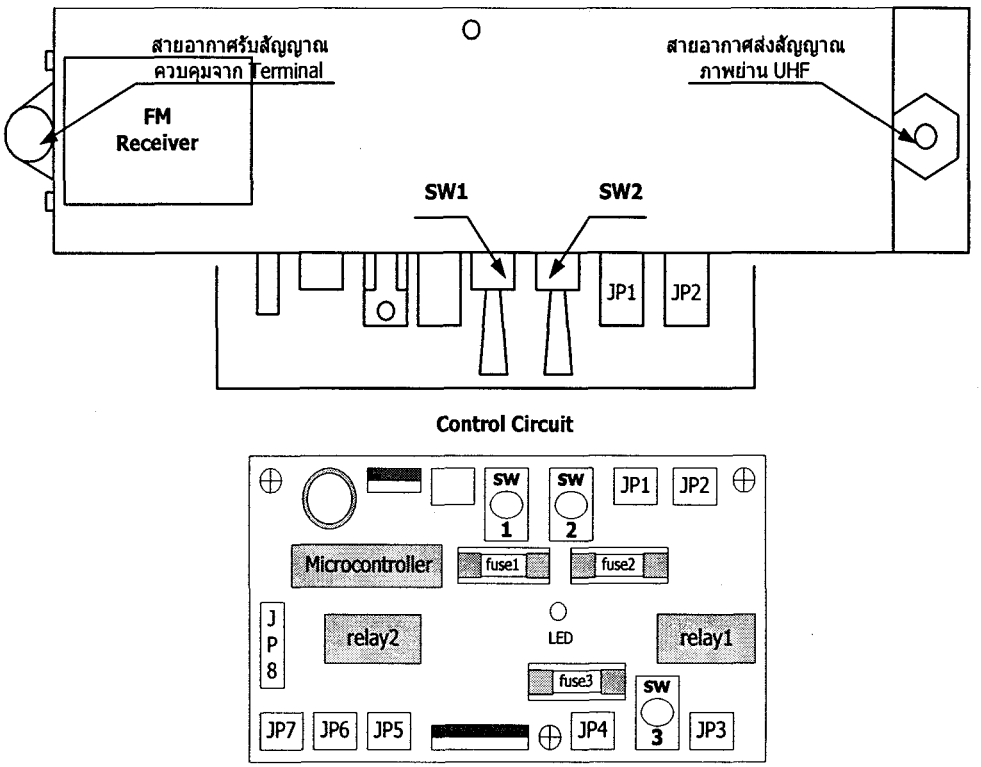
เป็นแหล่งพลังงานของหุ่นยนต์ โดยใช้แบตเตอรี่ทั้งหมด 4 ก้อน ซึ่งถูกใช้เป็นแหล่งพลังงานเพื่อจ่ายให้กับชุดมอเตอร์ในระบบขับเคลื่อน 2 ก้อน ส่วนอีก 2 ก้อนที่เหลือถูกใช้เป็นแหล่งพลังงานสำหรับจ่ายให้กับกล้องถ่ายรูปและอุปกรณ์ส่งสัญญาณภาพ 1 ก้อนและก้อนสุดท้ายสำหรับวงจรควบคุมหุ่นยนต์และอุปกรณ์รับสัญญาณควบคุมจากคอมพิวเตอร์เทอร์มินัล

2.2.4 อุปกรณ์รับสัญญาณควบคุมจากคอมพิวเตอร์เทอร์มินัล (FM Receiver)

เป็นอุปกรณ์ที่ทำหน้าที่ในการถอดรหัสสัญญาณ FM ซึ่งเป็นรหัสคำสั่งต่างๆ ที่ถูกส่งมาจากคอมพิวเตอร์เทอร์มินัล รหัสคำสั่งที่ได้จะถูกนำไปป้อนให้วงจรควบคุมหุ่นยนต์ เพื่อควบคุมให้หุ่นยนต์ทำตามคำสั่งของ เทอร์มินัลต่อไป จากรูปที่ 3.9 และรูปที่ 3.10 จะเห็นว่าอุปกรณ์ดังกล่าวถูกติดตั้งไว้ทางด้านท้ายของหุ่นยนต์ ซึ่งจะเป็นตำแหน่งที่ไกลจากตำแหน่งในการติดตั้งอุปกรณ์ส่งสัญญาณภาพ ทั้งนี้ก็เพื่อเป็นการป้องกันหรือลดการเกิดสัญญาณรบกวนระหว่างอุปกรณ์ตามหลักการการจัดสรรพื้นที่ในการติดตั้งอุปกรณ์

2.2.5 วงจรควบคุมหุ่นยนต์

จากรูปที่ 3.10 จะเห็นว่าวงจรควบคุมจะถูกติดตั้งในตำแหน่งทางด้านท้ายของหุ่นยนต์ โดยวงจรควบคุมนี้จะมีทำหน้าที่หลักๆ ดังต่อไปนี้



รูปที่ 3.11 วงจรควบคุมของหุ่นยนต์ GUNDAM

- ตรวจสอบความถูกต้องของรหัสคำสั่งต่างๆ ที่ได้จากการถอดรหัสสัญญาณ FM ของอุปกรณ์รับสัญญาณควบคุมจากเทอร์มินัล เนื่องจากในระหว่างที่สัญญาณควบคุมจากเทอร์มินัล ถูกส่งผ่านทางอากาศมาจนถึงตัวหุ่นยนต์นั้น สัญญาณดังกล่าวอาจถูกรบกวนจากสัญญาณรบกวนต่างๆ จนทำให้ข้อมูลหรือรหัสคำสั่งที่อยู่ในสัญญาณดังกล่าวได้รับความเสียหาย ดังนั้นรหัสคำสั่งที่ได้จากการถอดรหัสสัญญาณ FM ดังกล่าวจึงเป็นรหัสคำสั่งที่ไม่ถูกต้อง และหากหุ่นยนต์ยังมีการปฏิบัติตามรหัสคำสั่งที่ผิดพลาดดังกล่าวแล้ว อาจมีผลทำให้หุ่นยนต์ทำงานผิดพลาดจนเกิดความเสียหายต่อตัวหุ่นยนต์ได้ ดังนั้นเพื่อเป็นการป้องกันความเสียหายของตัวหุ่นยนต์จากกรณีดังกล่าว วงจรควบคุมของตัวหุ่นยนต์จึงจำเป็นที่จะต้องมีความสามารถในการตรรวจสอบความถูกต้องของรหัสคำสั่งต่างๆ ที่ได้รับมา และหากตรวจพบความผิดพลาดเกิดขึ้นกับรหัสคำสั่งใดแล้ว วงจรควบคุมก็จะไม่สั่งให้หุ่นยนต์ปฏิบัติตามรหัสคำสั่งนั้นๆ

- นำรหัสคำสั่งที่ถูกต้องไปควบคุมวงจรขับมอเตอร์ เพื่อให้หุ่นยนต์ปฏิบัติตามรหัสคำสั่งที่ได้รับมาต่อไป

จากรูปที่ 3.11 แสดงให้เห็นถึงสวิตซ์ควบคุมวงจร ฟิวส์และจัมพ์เปอร์ (jumper) ต่างๆ ซึ่งมีรายละเอียดดังต่อไปนี้

SW1 : เป็นสวิตซ์จ่ายไฟเลี้ยงให้กับวงจรอิเล็คทรอนิคของวงจรควบคุมรวมทั้งวงจรของ FM Receiver ด้วย ซึ่งเมื่อสวิตซ์อยู่ที่ตำแหน่ง ON จะทำให้ LED สว่างและ LED จะดับเมื่อสวิตซ์อยู่ที่ตำแหน่ง OFF

SW2 : เป็นสวิตซ์จ่ายไฟสำหรับขับมอเตอร์ของระบบขับเคลื่อนของหุ่นยนต์

SW3 : เป็นสวิตซ์จ่ายไฟสำหรับกล้องถ่ายรูปและภาคส่งสัญญาณภาพของหุ่นยนต์

Fuse1 : เป็นฟิวส์ที่ทำหน้าที่ในการตัดวงอิเล็คทรอนิคออกจากแหล่งจ่ายไฟฟ้าที่จ่ายไฟฟ้ามาจากสวิตซ์ SW1 ซึ่งเป็นการป้องอุปกรณ์ต่างๆ จากความที่อาจจะเกิดขึ้น ในกรณีเกิดไฟฟ้าลัดวงจรในระบบของวงจรอิเล็คทรอนิคต่างๆ ที่เชื่อมต่อกับแหล่งจ่ายไฟฟ้าโดยสวิตซ์ SW1

Fuse2 : ทำหน้าที่ในการตัดวงจรมอเตอร์ขับเคลื่อนออกจากแหล่งจ่ายไฟฟ้าที่เชื่อมต่อกันโดยสวิตซ์ SW2 ทั้งนี้เพื่อเป็นการป้องกันความเสียหายที่อาจจะเกิดขึ้น ในกรณีที่เกิดไฟฟ้าลัดวงจรขึ้นในวงจรของระบบขับเคลื่อน

Fuse3 : ทำหน้าที่ในการตัดวงจรอุปกรณ์ส่งสัญญาณภาพและวงจรของกล้องถ่ายภาพออกจากแหล่งจ่ายไฟฟ้า เพื่อเป็นการป้องกันความเสียหายที่อาจจะเกิดขึ้นกับอุปกรณ์ดังกล่าว ในกรณีเกิดไฟฟ้าลัดวงจรขึ้นในระบบของอุปกรณ์ดังกล่าว

JP1 : สำหรับแหล่งจ่ายไฟฟ้า(12 V) จ่ายไฟฟ้าให้กับวงจรของสวิตซ์ SW1

JP2 : สำหรับแหล่งจ่ายไฟฟ้า(24 V) จ่ายไฟฟ้าให้กับวงจรของสวิตซ์ SW2

JP3 : สำหรับจ่ายไฟฟ้าจากวงจรควบคุมไปยังอุปกรณ์ส่งสัญญาณภาพและกล้องถ่ายภาพ

JP4 : สำหรับจ่ายไฟฟ้าจากวงจรควบคุมไปยังชุดมอเตอร์ขับเคลื่อนทางด้านขวา

JP5 : สำหรับแหล่งจ่ายไฟฟ้า(12 V) จ่ายไฟฟ้าให้กับวงจรของสวิตซ์ SW3

JP6 : สำหรับจ่ายไฟฟ้าจากวงจรควบคุมไปยังชุดสเต็ปมอเตอร์ที่ทำหน้าที่ในการปรับทิศทางของกล้องถ่ายภาพ

JP7 : สำหรับจ่ายไฟฟ้าจากวงจรควบคุมไปยังชุดมอเตอร์ขับเคลื่อนทางด้านซ้าย

JP8 : สำหรับรับรหัสคำสั่งต่างๆ ที่ได้จากอุปกรณ์ถอดรหัสสัญญาณ FM และสำหรับจ่ายไฟฟ้าจากวงจรควบคุมไปยังอุปกรณ์ดังกล่าวด้วย

2.3 วงจรควบคุม

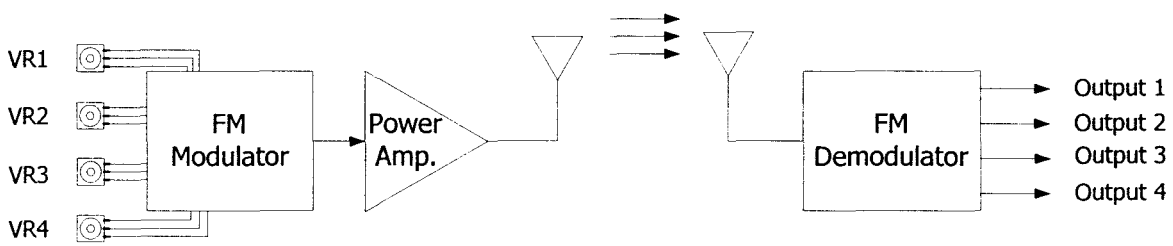
ในหัวข้อนี้จะกล่าวถึงวงจรที่ใช้ทั้งหมดในตัวหุ่นยนต์ GUNDAM I ซึ่งจะประกอบไปด้วย วงจรรับส่งภาพผ่านคลื่นความถี่ย่านความถี่ UHF ซึ่งถูกใช้ส่งภาพจากตัวหุ่นยนต์ไปยังเทอร์มินัล วงจรรับส่งในย่านความถี่ FM ซึ่งถูกใช้ในการส่งข้อมูลจากเทอร์มินัลไปควบคุมตัวหุ่นยนต์และสุดท้ายคือวงจรขับมอเตอร์

2.3.1 วงจรรับส่งสัญญาณภาพในย่านความถี่ UHF

ในวงจรรับส่งภาพผ่านคลื่นความถี่ย่าน UHF นั้นค่อนข้างมีความซับซ้อน หากว่าจะประดิษฐ์ขึ้นมาใช้งานเองจะทำให้เสียเวลาในการศึกษาวิจัยมาก ดังนั้นเพื่อให้มีความสะดวกรวดเร็วในการพัฒนาหุ่นยนต์ GUNDAM นี้จึงได้เลือกที่จะจัดซื้อวงจรรับส่งสัญญาณภาพที่ใช้ความถี่ย่าน UHF ที่มีจำหน่ายอยู่ทั่วๆ ไปมาใช้งาน โดยวงจรในภาคส่งจะใช้ตัวส่ง UHF และวงจรภาครับจะใช้ทีวีจูนเนอร์ (TV Tuner) ที่ใช้กับคอมพิวเตอร์เทอร์มินัลทั่วๆ ไปได้

2.3.2 วงจรรับส่งข้อมูลในระบบ FM

เช่นเดียวกับวงจรรับส่งสัญญาณภาพ วงจรรับส่งข้อมูลในระบบส่งแบบ FM นั้น หากต้องการให้วงจรดังกล่าวมีความสามารถในการรับส่งข้อมูลได้ในระยะไกลมากๆ โดยที่ยังคงมีประสิทธิภาพในการรับส่งข้อมูลสูงนั้น จะมีความซับซ้อนยุ่งยากมาก ซึ่งหากจะประดิษฐ์ขึ้นมาใช้งานเองก็จะทำให้เสียเวลาศึกษาวิจัยในการประดิษฐ์มากเช่นกัน ดังนั้นเพื่อความสะดวกรวดเร็วของการสร้างหุ่นยนต์ GUNDAM นี้ เราจึงเลือกที่จะจัดซื้อวงจรรับส่งข้อมูลในระบบ FM มาใช้งาน โดยวงจรที่ซื้อมาใช้งานนั้นจะเป็นวงจรที่ใช้ในการบังคับเครื่องบินเล็ก ซึ่งอุปกรณ์ดังกล่าวจะมีลักษณะการทำงานดังรูปที่ 3.12



รูปที่ 3.12 ลักษณะการทำงานของอุปกรณ์บังคับเครื่องบินเล็ก

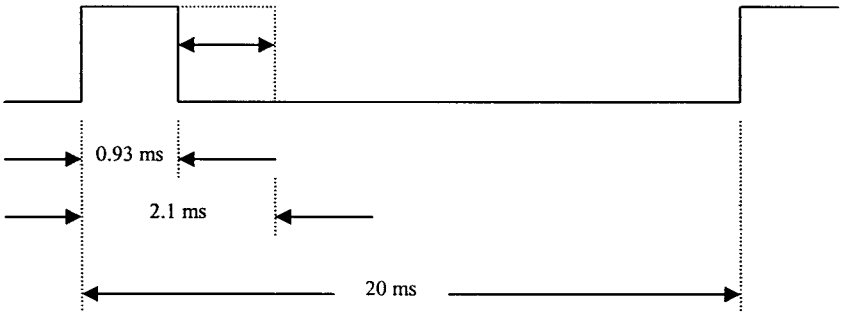
หลักการทำงานของอุปกรณ์บังคับเครื่องบินเล็กนั้น จะแบ่งเป็น 2 ส่วนด้วยกันได้แก่ ภาคส่งและภาครับ ดังรายละเอียดต่อไปนี้

- ภาคส่ง

จะมีความต้านทานปรับค่าได้อยู่ 4 ตัว ซึ่งจะถูกใช้ในการบังคับทิศทางของเครื่องบิน ความต้านทานปรับค่าได้ทั้ง 4 ตัวจะถูกใช้ทั้ง 3 ขา โดยขาบนจะถูกป้อนด้วยแรงดันไฟฟ้า +5V, ขากลางจะเป็น เอาต์พุต ซึ่งจะคอยส่งข้อมูลไปให้วงจร FM มอดูเลเตอร์และขาล่างสุดจะเป็นกราวด์ของระบบ ซึ่งลักษณะของวงจรดังกล่าวจะมีหน้าที่ในการปรับค่าแรงดันไฟฟ้าเอาต์พุตให้มีค่าอยู่ในช่วง 0V – 5V ซึ่งแรงดันไฟฟ้า เอาต์พุตที่ได้จะถูกนำไปป้อนให้กับวงจร FM มอดูเลเตอร์ต่อไป

- ภาครับ

จะมีเอาต์พุตอยู่ 4 เส้น ซึ่งสัญญาณที่ได้จะเป็นสัญญาณ PWM (Pulse Width Modulate) สัญญาณ 1 คาบมีลักษณะดังรูปที่ 3.13 กล่าวคือ หากภาคส่งป้อนเอาต์พุต 5V ให้กับวงจร FM มอดดูเลเตอร์จะทำให้ที่ภาครับให้สัญญาณเอาต์พุตที่มีช่วงของค่าลอจิก 1 เท่ากับ 2.1 ms หากภาคส่งป้อนเอาต์พุต 0V ให้กับวงจร FM มอดดูเลเตอร์ จะทำให้ที่ภาครับให้สัญญาณเอาต์พุตที่มีช่วงของค่าลอจิก 1 เท่ากับ 0.93 ms



รูปที่ 3.13 ลักษณะสัญญาณ PWM ที่ออกจากภาครับของอุปกรณ์บังคับเครื่องบินเล็ก

ในการนำเอาอุปกรณ์บังคับเครื่องบินเล็กมาประยุกต์ใช้ในหุ่นยนต์ GUNDAM จะมีลักษณะที่ต้องการของวงจรรับส่งสัญญาณ FM ดังต่อไปนี้

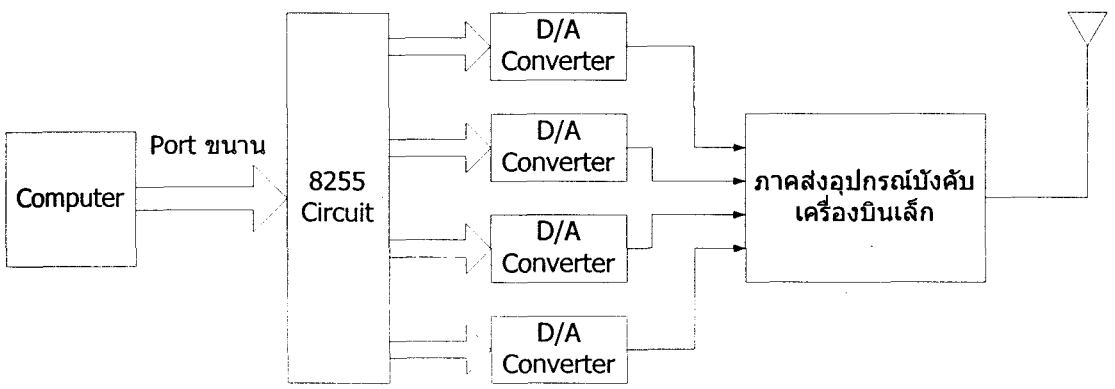
- วงจรของภาคส่ง FM จะต้องมีความสามารถในการรับข้อมูลที่เป็นข้อมูลดิจิตอลจากคอมพิวเตอร์ส่วนบุคคลได้
- สัญญาณที่ออกจากภาครับจะต้องสามารถนำไปใช้ควบคุมมอเตอร์ชนิด DC จำนวน 3 ตัว โดยที่เป็น DC มอเตอร์ธรรมดาแบบปรับความเร็วได้ 2 ตัวและเป็นเซอร์โวมอเตอร์อีก 1 ตัว

2.3.3 การดัดแปลงอุปกรณ์บังคับเครื่องบินเล็กให้เป็นเครื่องรับส่งข้อมูลที่ต้องการ เมื่อเปรียบเทียบหลักการทำงานของอุปกรณ์บังคับเครื่องบินเล็กกับลักษณะวงจรรับส่งในย่านความถี่ FM ที่ต้องการในการออกแบบหุ่นยนต์ GUNDAM นี้ จะเห็นได้ว่าสิ่งที่จะต้องเพิ่มเติมเข้ามาในอุปกรณ์บังคับเครื่องบินเล็กมีอยู่ 2 ส่วนด้วยกัน คือ อุปกรณ์เชื่อมต่อระหว่างคอมพิวเตอร์กับอุปกรณ์

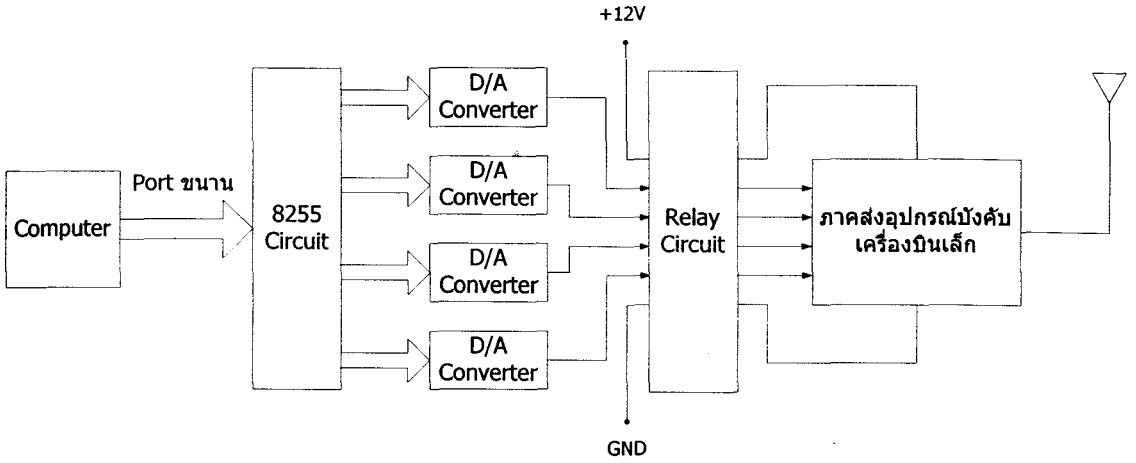
บังคับเครื่องบินเล็กและอุปกรณ์ถอดรหัสสัญญาณ PWM จากภาครับอุปกรณ์บังคับเครื่องบินเล็กและวงจรขับมอเตอร์

- อุปกรณ์เชื่อมต่อระหว่างคอมพิวเตอร์กับอุปกรณ์บังคับเครื่องบินเล็ก

จากหลักการทำงานของภาคส่งของอุปกรณ์บังคับเครื่องบินเล็ก จะเห็นได้ว่าอินพุตที่ต้องป้อนให้กับอุปกรณ์นั้น จะเป็นสัญญาณแอนะลอก (0V – 5V) แต่คอมพิวเตอร์ไม่สามารถจ่ายสัญญาณแอนะลอกได้ ดังนั้นจึงต้องใช้อุปกรณ์แปลงสัญญาณดิจิตอลเป็นแอนาลอกหรือ DAC (Digital to Analog Converter) ในการแปลงสัญญาณดิจิตอลให้เป็นสัญญาณแอนะลอก โดยตัวแปลง D/A ที่เลือกใช้นั้นจะใช้วงจรชนิด R-To-R Ladder และเนื่องจากสัญญาณอินพุตที่ป้อนให้ภาคส่งอุปกรณ์บังคับเครื่องบินเล็กควรมีความละเอียดสูง ดังนั้นจึงได้ออกแบบให้อุปกรณ์ดังกล่าวมีความละเอียดที่ 8 บิตและหากพิจารณาถึงอินพุตที่จะต้องป้อนให้กับภาคส่งของอุปกรณ์บังคับเครื่องบินเล็กแล้ว พบว่าจะต้องใช้อินพุตทั้งหมดจำนวน 4 อินพุต ซึ่งทำให้ต้องใช้ตัวแปลง D/A ทั้งหมดจำนวน 4 ตัวด้วยเช่นกัน ดังนั้นจึงต้องใช้อินพุตที่เป็นสัญญาณดิจิตอล ป้อนให้กับตัวแปลง D/A จำนวน 4 ตัว รวมทั้งหมดเป็น 32 บิต แต่พอร์ตขนานของคอมพิวเตอร์มีพอร์ตเอาต์พุตเพียง 12 พอร์ต ซึ่งทำให้ต้องมีการขยายพอร์ตของคอมพิวเตอร์ เพื่อให้เพียงพอกับความต้องการที่จะใช้งาน โดยในที่นี้จะใช้ IC 8255 (รายละเอียดดูได้จากชีทข้อมูลของ IC 8255 ทั่วไป) จำนวน 2 ตัวดังรูปที่ 3.14 และเมื่อทำการประกอบวงจรแล้วจะพบว่า มีบางครั้งที่อุปกรณ์ดังกล่าวจะไม่สามารถใช้งานได้ ซึ่งผู้วิจัยเองก็ยังไม่ถึงทราบสาเหตุที่แท้จริง แต่จากการทดสอบพบว่า หากทำการจ่ายไฟให้กับภาคส่งของอุปกรณ์บังคับเครื่องบินเล็กก่อน จากนั้นจึงทำการป้อนอินพุตที่เป็นสัญญาณแอนะลอกทั้ง 4 ที่ได้จากตัวแปลง D/A ให้กับอุปกรณ์ดังกล่าวแล้วจะทำให้โอกาสที่อุปกรณ์ดังกล่าวจะใช้งานไม่ได้ลดลง ดังนั้นจึงต้องเพิ่มวงจรสวิตซ์รีเลย์ (Relay Switch) เข้ามาควบคุมอินพุตทั้ง 4 และไฟเลี้ยงวงจร จะทำให้ได้ระบบใหม่ดังรูปที่ 3.15 โดยทั้งนี้ได้ทำการแสดงอินพุตทั้งหมดที่ใช้ควบคุมภาคส่งไว้ในตารางที่ 3.2 และตารางที่ 3.3



รูปที่ 3.14 การดัดแปลงภาคส่งอุปกรณ์บังคับเครื่องบินเล็ก



รูปที่ 3.15 การดัดแปลงภาคส่งอุปกรณ์บังคับเครื่องบินเล็ก (ปรับปรุง)

Parallel Port Pin No.	Active	ลักษณะการใช้งาน
1 (Controls Bit 0)	Low	Pin A0 ใช้ควบคุม Address ของ IC 8255
2 (Data Bit 0)	High	Data Bus ที่รับส่งกับ IC 8255
3 (Data Bit 1)	High	
4 (Data Bit 2)	High	
5 (Data Bit 3)	High	
6 (Data Bit 4)	High	
7 (Data Bit 5)	High	
8 (Data Bit 6)	High	
9 (Data Bit 7)	High	
10 (Status Bit 6)	High	
11 (Status Bit 7)	Low	
12 (Status Bit 5)	High	-
13 (Status Bit 4)	High	-
14 (Control Bit 1)	Low	Pin A1 ใช้ควบคุม Address ของ IC 8255
15 (Status Bit 3)	High	-
16 (Control Bit 2)	High	Pin WR ของ IC 8255 ตัวที่ 2
17 (Control Bit 3)	Low	Pin WR ของ IC 8255 ตัวที่ 1
18-25 (GND)	-	GND

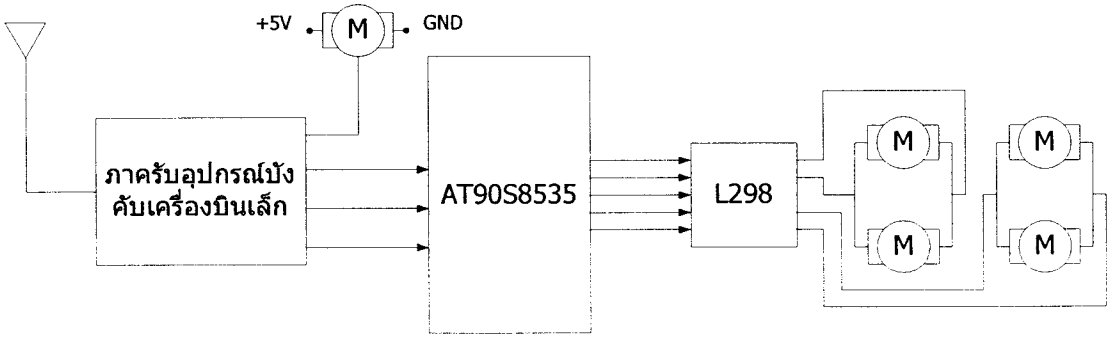
ตารางที่ 3.2 หน้าที่ของขา (pin) จากพอร์ตขนานที่ใช้ควบคุมภาคส่ง

IC8255 ตัวที่ 1	PORTA (A1A0 = 00)	D/A (Channel 1) PORTA0 = LSB, PORTA7 = HSB
	PORTB (A1A0 = 01)	D/A (Channel 2) PORTB0 = LSB, PORTB7 = HSB
	PORTC (A1A0 = 10)	PORTC0 มีหน้าที่เปิดปิดไฟเลี้ยงที่เข้าตัวส่งอุปกรณ์บังคับเครื่องบินเล็ก PORTC1 มีหน้าที่เปิดปิดสายข้อมูล 4 สายจาก D/A เข้าตัวส่งอุปกรณ์บังคับเครื่องบินเล็ก
IC8255 ตัวที่ 2	PORTA (A1A0 = 00)	D/A (Channel 3) PORTA0 = LSB, PORTA7 = HSB
	PORTB (A1A0 = 01)	D/A (Channel 4) PORTB0 = LSB, PORTB7 = HSB
	PORTC (A1A0 = 10)	-

ตารางที่ 3.3 หน้าที่ของขาเอาต์พุตที่ออกจาก IC 8255

- อุปกรณ์ถอดรหัสสัญญาณ PWM จากภาครับของอุปกรณ์บังคับเครื่องบินเล็กและวงจรขับมอเตอร์

ในที่นี้ได้ใช้ไมโครคอนโทรลเลอร์ตระกูล AVR เบอร์ 90S2313 เป็นอุปกรณ์ถอดรหัสสัญญาณ PWM จากภาครับของอุปกรณ์บังคับเครื่องบินเล็ก โดยจะรับสัญญาณจากภาครับอุปกรณ์บังคับเครื่องบินเล็กมา 3 ช่อง (อีก 1 ช่องจะใช้ควบคุมเซอร์โวมอเตอร์โดยตรงเนื่องจากตัวรับนี้ถูกออกแบบมาใช้ควบคุมเซอร์โวมอเตอร์) และเมื่อไมโครคอนโทรลเลอร์ได้ทำการถอดรหัสสัญญาณ PWM แล้วข้อมูลที่ได้จะถูกนำไปทำการตรวจสอบความถูกต้องก่อน จากนั้นจึงจะนำข้อมูลที่ได้ดังกล่าวไปทำการแปลเป็นรหัสคำสั่งเพื่อควบคุมหุ่นยนต์ต่อไป ทั้งนี้หุ่นยนต์จะถูกควบคุมโดยไมโครคอนโทรลเลอร์ ซึ่งจะทำการส่งสัญญาณไปควบคุม DC มอเตอร์ ผ่านทางวงจรขับมอเตอร์ โดยที่วงจรขับมอเตอร์นั้นจะใช้ IC L298 เป็นตัวขับมอเตอร์ (ดูรายละเอียดได้จากชีทข้อมูลของ IC L298 ทั่วไป) ระบบการทำงานของภาครับทั้งหมดแสดงไว้ในรูปที่ 3.16 ลักษณะการถอดรหัสสัญญาณ



รปที่ 3.16 แผนผังการดัดแปลงภาครับวงจรบังคับเครื่องบินเล็ก

PWM นั้นจะกล่าวในหัวข้อการควบคุมอินพุต/เอาต์พุตต่างๆ ของไมโครคอนโทรลเลอร์ ซึ่งได้แสดงไว้ในตารางที่ 3.4

- การเข้ารหัสและถอดรหัส

จะทำโดยแบ่งช่วงเวลาการเปลี่ยนแปลงของสัญญาณ PWM ออกเป็นช่วงย่อยๆ (จะทำการเข้ารหัสและถอดรหัสเฉพาะช่อง 1 3 และ 4 เท่านั้น เพราะช่องที่ 2 จะนำไปควบคุมทิศทางกล้องซึ่งใช้เซอร์โวมอเตอร์ โดยอุปกรณ์บังคับเครื่องบินเล็กนั้นได้ออกแบบมาให้ใช้กับมอเตอร์ตัวนี้) โดยแต่ละช่องจะมีความสามารถส่งข้อมูลไปควบคุมความกว้างของสัญญาณ PWM ได้สูงสุด 8 บิตต่อช่อง (เป็น D/A ขนาด 8 บิต) ซึ่งจะทำให้สัญญาณ PWM ที่สามารถส่งได้มีความระเอียดทางเวลาสูงสุด 4.57 us แต่จากการทดลองพบว่าการส่งข้อมูลที่ 8 บิตต่อช่องจะมีโอกาสผิดพลาดค่อนข้างสูง ดั งนั้นเพื่อเพิ่มประสิทธิภาพของระบบจึงลดจำนวนบิตลงเหลือ 3 บิตต่อช่อง ดังนั้นการส่งข้อมูลควบคุมล้อทั้ง 4 ล้อจะส่งได้สูงสุด 9 บิตในเวลาเดียวกัน ซึ่งลักษณะข้อมูลในการส่งจะแสดงไว้ในตารางที่ 3.5ตารางที่ 3.5

PortB (Output)	B.7	ต่อเข้าขา Input1 ของ L298 มีหน้าที่ควบคุมทิศทางการหมุนของ Motor Channel 1
	B.6	ต่อเข้าขา Input2 ของ L298 มีหน้าที่ควบคุมทิศทางการหมุนของ Motor Channel 1
	B.5	ต่อเข้าขา Input3 ของ L298 มีหน้าที่ควบคุมทิศทางการหมุนของ Motor Channel 2
	B.4	ต่อเข้าขา Input4 ของ L298 มีหน้าที่ควบคุมทิศทางการหมุนของ Motor Channel 2
	B.3	ต่อเข้าขา Enable ของทั้ง 2 Channel มีหน้าที่ส่งสัญญาณ PWM เพื่อควบคุมความเร็ว
	B.2	-
	B.1	-
	B.0	-
PortD (Input)	D.7	-
	D.6	-
	D.5	-
	D.4	รับสัญญาณจาก Channel 4
	D.3	รับสัญญาณจาก Channel 3
	D.2	รับสัญญาณจาก Channel 1
	D.1	-
	D.0	-

ตารางที่ 3.4 ฟังก์ชันการทำงานของอินพุต/เอาต์พุตต่างๆ ของไมโครคอนโทรลเลอร์

บิต 2	บิต 1	บิต 0	ข้อมูลที่ป้อนให้ D/A (0-255)	ช่วงเวลาของสัญญาณ PWM (ms)	ช่วงเวลาในการตัดสินใจในการถอดรหัส (ms)
0	0	0	0	0.93	0 - 1.02
0	0	1	37	1.09	1.02 - 1.15
0	1	0	68	2.38	1.15 - 1.3
0	1	1	99	1.37	1.3 - 1.44
1	0	0	130	1.58	1.44 - 1.58
1	0	1	162	1.66	1.58 - 1.73
1	1	0	193	1.8	1.73 - 1.87
1	1	1	255	2.1	1.87 - 2.5

ตารางที่ 3.5 ลักษณะข้อมูลที่ใช้ควบคุมอุปกรณ์ส่ง FM

2.3.4 ระบบควบคุม

ในหัวข้อนี้จะกล่าวถึงลักษณะและวิธีการควบคุมอุปกรณ์ทั้งหมดซึ่งจะถูกแบ่งออกเป็น 2 หัวข้อคือ ลักษณะของระบบโดยรวม การเชื่อมต่อระหว่างคอมพิวเตอร์กับอุปกรณ์ส่ง FM

- ลักษณะของระบบโดยรวม

ระบบจะมีลักษณะดังนี้คือคอมพิวเตอร์จะส่งข้อมูลไปยัง อุปกรณ์ส่ง FM ซึ่ง อุปกรณ์ส่ง FM นี้จะประกอบไปด้วย 4 ช่องและอุปกรณ์รับ FM ในตัวหุ่นยนต์จะรับข้อมูลและส่งข้อมูลช่อง 1 3 และ 4 ไปให้ไมโครคอนโทรลเลอร์เพื่อไปถอดรหัสและส่งข้อมูลไปควบคุมมอเตอร์ล้อทั้ง 4 ล้อต่อไป ส่วนช่อง 2 นั้นจะนำไปควบคุมมอเตอร์เพื่อปรับทิศทางกล้อง (เซอร์โวมอเตอร์) โดยตรง ลักษณะข้อมูลที่ใช้ในแต่ละช่องได้แสดงไว้ดังตารางที่ 3.5 และหน้าที่ของแต่ละช่องมีแสดงไว้ในตารางที่ 3.6

ช่อง	บิต 2	บิต 1	บิต 0	หน้าที่
1	-			Code Error โดยจะใช้ Code 001 ส่งไปทุกครั้งเมื่อมีการส่งข้อมูล
2	-			ปรับทิศทางกล้อง โดยจะใช้ทั้ง 8 บิต (มี 256 ระดับ)
3	0	0	0	Speed 1 ต่ำสุด
	0	0	1	Speed 2
	0	1	0	Speed 3
	0	1	1	Speed 4
	1	0	0	Speed 5
	1	0	1	Speed 6
	1	1	0	Speed 7
	1	1	1	Speed 8 สูงสุด
4	0	0	0	คำสั่งให้หุ่นยนต์หยุด
	0	0	1	คำสั่งให้หุ่นยนต์เดินหน้า
	0	1	0	คำสั่งให้หุ่นยนต์ถอยหลัง
	0	1	1	คำสั่งให้หุ่นยนต์เลี้ยวขวา
	1	0	0	คำสั่งให้หุ่นยนต์เลี้ยวซ้าย
	1	0	1	-
	1	1	0	-
	1	1	1	-

ตารางที่ 3.6 หน้าที่การทำงานของแต่ละช่อง

- การเชื่อมต่อระหว่างคอมพิวเตอร์กับอุปกรณ์ส่งสัญญาณ FM

จากรูปที่ 3.15 การดัดแปลงภาคส่งอุปกรณ์บังคับเครื่องบินเล็ก จะเห็นได้ว่าการเชื่อมต่อระหว่างคอมพิวเตอร์กับอุปกรณ์ส่ง FM จะต้อง เชื่อมต่อผ่าน IC 8255 โดยลักษณะขาอินพุต/เอาต์พุตต่างๆ ที่ได้กล่าวไปแล้วในข้างต้น วิธีการควบคุมอุปกรณ์ส่ง FM จะมีขั้นตอนดังนี้

1. ตั้งต้นการทำงานพอร์ตของ IC 8255 ทั้ง 2 ตัว
2. ปิด SW. ที่จ่ายไฟและข้อมูลอินพุตที่เข้าสู่ภาคส่งอุปกรณ์บังคับเครื่องบินเล็ก
3. เปิด SW. ที่จ่ายไฟที่เข้าสู่ภาคส่งอุปกรณ์บังคับเครื่องบินเล็ก
4. เปิด SW. ข้อมูลอินพุตที่เข้าสู่ภาคส่งอุปกรณ์บังคับเครื่องบินเล็ก
5. เริ่มส่งข้อมูลได้

- ตัวอย่างต้นรหัสโปรแกรม

การสั่งให้หุ่นยนต์เดินหน้าด้วยความเร็วสูงสุดและปรับกล้องให้หันตรง โดยใช้ภาษา C

```
_outp(0x378,128);    //Initial Port ของ IC8255 ทั้ง 2 ตัว
_outp(0x378+2,8);    //0011
_outp(0x378+2,4);    //1111
_outp(0x378+2,8);    //0011
_outp(0x378+2,4);    //1111
_outp(0x378,0);      //ปิด SW. ที่จ่ายไฟและสาย Data ทั้ง4
_outp(0x378+2,9);
_outp(0x378+2,1);
_outp(0x378+2,9);
_outp(0x378+2,1);
_outp(0x378,1);      //เปิด SW. ที่จ่ายไฟ
_outp(0x378+2,9);
_outp(0x378+2,1);
_outp(0x378+2,9);
_outp(0x378+2,1);
_outp(0x378,3);      //เปิด SW. สาย Data ทั้ง4
_outp(0x378+2,9);
_outp(0x378+2,1);
_outp(0x378+2,9);
_outp(0x378+2,1);
_outp(0x378,34);     //Channel 1 Code error
_outp(0x378+2,11);   //0000    1011 1011
_outp(0x378+2,3);    //1000    0011
_outp(0x378+2,11);   //0000    1011
_outp(0x378+2,3);    //1000    0011
_outp(0x378,128);    //Channel 2 Camera
_outp(0x378+2,10);   //0001    1010 1011
```

```
_outp(0x378+2,2);              //1001    0010
_outp(0x378+2,10);             //0001    1010
_outp(0x378+2,2);              //1001    0010

_outp(0x378,255);              //Channel 3 Speed
_outp(0x378+2,11);             //0000    1011 1011
_outp(0x378+2,15);             //0100    1111
_outp(0x378+2,11);             //0000    1011
_outp(0x378+2,15);             //0100    1111

_outp(0x378,37);               //Channel 4 Movement
_outp(0x378+2,10);             //0001    1010 1011
_outp(0x378+2,14);             //0101    1110
_outp(0x378+2,10);             //0001    1010
_outp(0x378+2,14);             //0101    1110
```

หลังจากการออกแบบและสร้างหุ่นยนต์ GUNDAM I เรียบร้อยแล้ว หุ่นยนต์มีความสามารถพื้นฐานในการเคลื่อนที่ตามคำสั่งจากคอมพิวเตอร์และสามารถส่งภาพจากกล้องวิดีโอในย่าน UHF แบบไร้สายได้ ในการพัฒนาช่วงต่อไปเป็นการเน้นการประมวลผลภาพเพื่อควบคุมการทำงานของหุ่นยนต์

3. GUNDAM II

การพัฒนาหุ่นยนต์ GUNDAM II มุ่งเน้นไปที่การพัฒนาซอฟท์แวร์ โดยเฉพาะโปรแกรมในการประมวลผลภาพ เพื่อสามารถนำมาใช้ในการควบคุมหุ่นยนต์ได้ เป้าหมายหลักคือสามารถควบคุมหุ่นยนต์ด้วยข้อมูลจากการประมวลผลภาพได้ การพัฒนาในส่วนนี้สามารถแบ่งได้ออกเป็น 2 ส่วนใหญ่ๆ ดังนี้

- การพัฒนาโปรแกรมไดร์เวอร์เพื่อควบคุมการเคลื่อนที่ของหุ่นยนต์
- การพัฒนาโปรแกรมประมวลผลภาพสำหรับใช้เป็นคำสั่งในการเคลื่อนที่ของหุ่นยนต์

เนื่องมาจากการประมวลผลภาพต้องมีการใช้ทรัพยากรของคอมพิวเตอร์ค่อนข้างสูง ในการพัฒนาหุ่นยนต์ GUNDAM II จึงเลือกที่จะทำการพัฒนาโปรแกรมบนระบบปฏิบัติการ Windows 2000 เป็นหลัก เนื่องมาจากความเสถียรภาพของระบบปฏิบัติการและความเร็วในการทำงานที่เหนือกว่า

Windows 98 โปรแกรมการควบคุมหุ่นยนต์ GUNDAM II จึงต้องทำการพัฒนาในรูปแบบใหม่ดังรายละเอียดต่อไปนี้

3.1 การพัฒนาโปรแกรมไดร์เวอร์

โปรแกรมไดร์เวอร์เป็นโปรแกรมสำหรับเป็นตัวกลางเชื่อมต่อระหว่างระบบปฏิบัติการของคอมพิวเตอร์และฮาร์ดแวร์ ซึ่งในที่นี้คือวงจรควบคุมหุ่นยนต์ เพื่อให้หุ่นยนต์สามารถรับคำสั่งจากคอมพิวเตอร์เทอร์มินัลภายใต้ระบบปฏิบัติการ Windows 2000 ได้ ในการเขียนโปรแกรมควบคุมหุ่นยนต์ GUNDAM บนระบบปฏิบัติการ Windows 2000 นั้นจะมีความยุ่งยากกว่าบน Windows 98 เนื่องจากไม่สามารถใช้ฟังก์ชัน _inp และ _outp ซึ่งเป็นฟังก์ชันที่ใช้ในการติดต่อควบคุมพอร์ตขนานบน Windows 2000 ได้ดังนั้นจึงแก้ปัญหาโดยการสร้างไดร์เวอร์เพื่อเป็นตัวกลางในการเชื่อมต่อระหว่างวงจรควบคุมหุ่นยนต์ และโปรแกรมควบคุมบนคอมพิวเตอร์ เครื่องมือที่ใช้ในการพัฒนาโปรแกรมได้แก่

- Microsoft Visual C++ 6.0
- Microsoft Windows 2000 DDK

3.1.1 โครงสร้างของไดรเวอร์

- DriverEntry เป็นจุดเริ่มการทำงานของไดรเวอร์ ฟังก์ชันนี้จะถูกเรียกเพียงครั้งแรกเมื่อไดรเวอร์ถูกโหลด มีโครงสร้างดังนี้

```
NTSTATUS DriverEntry(
                IN PDRIVER_OBJECT  pDriverObject,
           IN PUNICODE_STRING pRegistryPath)
{
     NTSTATUS status;
     ...
     ...
     ...
     ...
     return status;
}
```

ฟังก์ชันนี้จะเป็นจุดที่กำหนดชื่อของฟังก์ชันหลักที่จะถูกเรียกใช้ โดยกำหนดชื่อฟังก์ชันหลักที่เราสร้างขึ้นให้กับ MajorFunction Array ดังนี้

```
pDriverObject->MajorFunction[IRP_MJ_CREATE] =
DispatchCreate;
pDriverObject->MajorFunction[IRP_MJ_CLOSE]  =
DispatchClose;
pDriverObject->MajorFunction[IRP_MJ_DEVICE_CONTROL] =
GDDeviceControl;
pDriverObject->DriverUnload = GDUnload;
```

โดยที่ DispatchCreate DispatchClose GDDeviceControl และ GDUnload เป็นชื่อฟังก์ชันที่เราสร้างขึ้น ใน MajorFunction Array ยังมีส่วนที่สามารถกำหนดได้อีกขึ้นอยู่กับอุปกรณ์ที่ไดร์เวอร์ไปควบคุม ในกรณีนี้จะกำหนดเฉพาะที่ใช้งาน คือ

IRP_MJ_CREATE จะถูกเรียกเมื่อเรียกใช้ฟังก์ชัน CreateFile เพื่อติดต่อกับอุปกรณ์
IRP_MJ_CLOSE จะถูกเรียกเมื่อเรียกใช้ฟังก์ชัน CloseHandle เพื่อหยุดการติดต่อ
IRP_MJ_DEVICE_CONTROL จะถูกเรียกเมื่อเรียกใช้ฟังก์ชัน DeviceIoControl เพื่อส่งคำสั่งไปควบคุมอุปกรณ์

ส่วนฟังก์ชันอื่นที่ต้องกำหนดคือ DriverUnload เป็นฟังก์ชันที่ทำงานเมื่อไดรเวอร์ถูกลบออกจากหน่วยความจำ

- GDCreateDevice เป็นฟังก์ชันที่ถูกเรียกใช้ใน DriverEntry โดยเรียกใช้หลังจากที่กำหนดชื่อฟังก์ชันหลักแล้ว ฟังก์ชันนี้จะทำการสร้างอุปกรณ์ออบเจ็คท์ (device object) และทำหน้าที่กำหนดชื่อภายใน (internal name) (เพื่อให้ Windows 2000 รู้จัก ซึ่งจะถูกใช้ภายในเท่านั้น) และกำหนดชื่อ Symbolic Link (ใช้สำหรับแอพพลิเคชัน) ซึ่งจะถูกใช้เป็นพารามิเตอร์ในการเรียกฟังก์ชัน CreateFile
- GDUnload เป็นฟังก์ชันที่หยุดการทำงานและลบไดรเวอร์ออกจากระบบ ถูกเรียกใช้โดย I/O Manager ของ Windows 2000
- DispatchCreate เป็นฟังก์ชันที่รองรับการเรียกใช้จากฟังก์ชัน CreateFile เท่านั้น
- DispatchClose ทำหน้าที่คืนบัพเฟอร์ให้กับระบบ
- GDDeviceControl เป็นฟังก์ชันที่รับคำสั่งจากแอพพลิเคชัน ซึ่งคำสั่งที่รับมาจะอยู่ในรูปแบบคำสั่งเฉพาะของอุปกรณ์ที่เราสร้างขึ้น โดยการสร้างคำสั่งจะใช้มาโคร CTL_CODE จาก ใน Windows 2000 DDK มีรูปแบบการใช้งานดังนี้

```
CTL_CODE( DeviceType, ControlCode, TransferType,
RequireAccess )
```

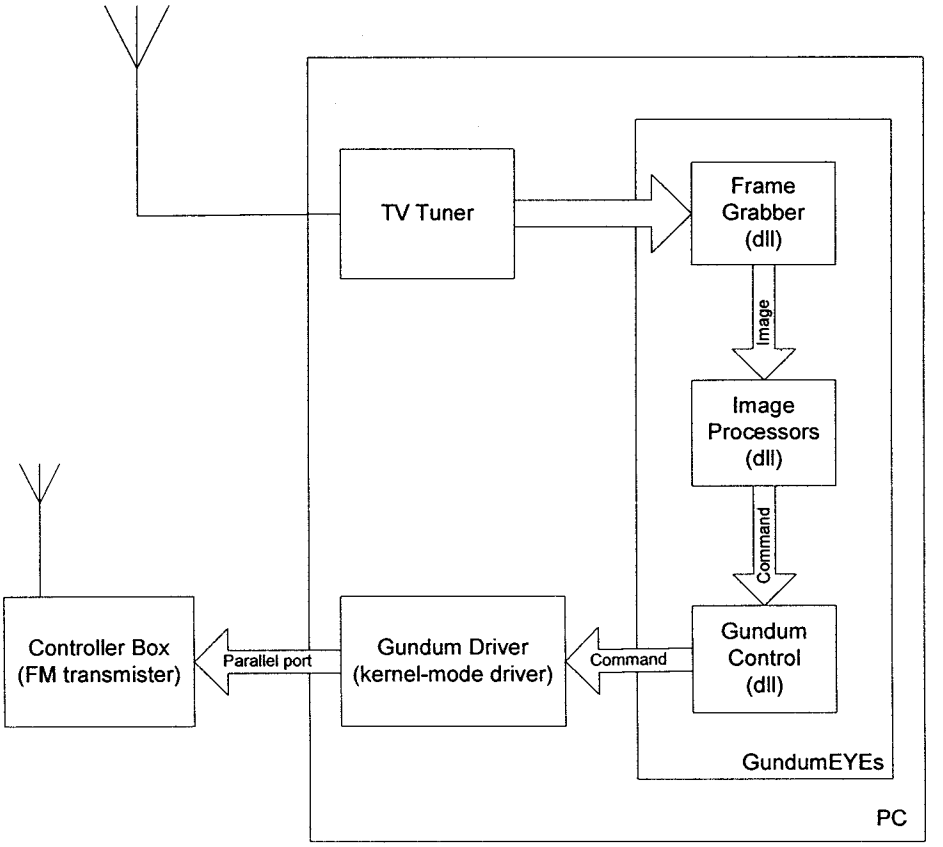
รายละเอียดตัวอย่างการสร้างโปรแกรมไดร์เวอร์พร้อมการติดตั้งมีอยู่ในภาคผนวก ก.

พารามิเตอร์	คำอธิบาย
DeviceType	FILE_DEVICE_XXX เป็นค่าเดียวกันกับที่ใช้ในฟังก์ชัน IoCreateDevice (เป็นฟังก์ชันที่ใช้ใน GDCreateDevice) • 0x0000 ถึง 0x7FFF เป็นค่าเฉพาะของไมโครซอฟท์ • 0x8000 ถึง 0xFFFF เป็นค่าที่เราสามารถกำหนดขึ้นใช้ได้
ControlCode	Driver-defined IOCTL code • 0x000 ถึง 0x7FF เป็นค่าเฉพาะของไมโครซอฟท์ • 0x800 ถึง 0xFFF เป็นค่าที่สามารถใช้ได้
TransferType	Buffer passing mechanism • METHOD_BUFFERED • METHOD_IN_DIRECT • METHOD_OUT_DIRECT • METHOD_NEITHER
RequiredAccess	Requestor access requirement • FILE_ANY_ACCESS • FILE_READ_DATA • FILE_WRITE_DATA • FILE_READ_DATA | FILE_WRITE_DATA

ตารางที่ 3.7 พารามิเตอร์ของมาโคร CTL_CODE

3.2 การพัฒนาโปรแกรมควบคุม

โปรแกรมที่พัฒนาขึ้นเพื่อทำการควบคุมหุ่นยนต์ GUNDAM (ในระดับสูง) มีชื่อรหัสโปรแกรมคือ GDEYEs ซึ่งประกอบด้วย 3 ส่วนหลักๆ คือ (1) ส่วนรับภาพหรือ frame grabber (2) ส่วนประมวลผลภาพหรือ image processor และ (3) ส่วนควบคุมหุ่นยนต์หรือ GUNDAM control โดยแต่ละส่วนจะพัฒนาในรูปแบบ Dynamic-Link Library (DLL) เพื่อให้ง่ายต่อการพัฒนาและใช้งาน



รูปที่ 3.17 การทำงานของโปรแกรม GDEYEs

โปรแกรม GDEYEs ถูกออกแบบในกรอบแนวคิดดังนี้

(1) สามารถเลือกเปลี่ยนวิธีการประมวลผลแบบต่างๆ ได้ในขณะรันโปรแกรม

(2) เนื่องจากใช้การสร้างโปรแกรมแบบ multiple documents ดังนั้น แต่ละ document จะต้องมีส่วนข้อมูลของตัวเอง ซึ่งเก็บค่าพารามิเตอร์ต่างๆ ที่จำเป็นสำหรับวิธีการประมวลผลในแต่ละแบบ การเลือกเขียนโปรแกรมในลักษณะนี้จะทำให้สามารถเลือกใช้ Image Processor ตัวเดียวกัน แต่พารามิเตอร์ต่างกันในแต่ละ document ได้

(3) สามารถกำหนดเวลาในการรับภาพจากการ์ดทีวีจูนเนอร์ (TV tuner card) ได้ เพื่อป้องกันไม่ให้โปรแกรมทำงานช้าเกินไป ซึ่งจะทำให้การประมวลผลเพื่อสั่งงานหุ่นยนต์ผิดพลาด

(4) สามารถตั้งค่าพารามิเตอร์ของวิธีการประมวลผลแบบต่างๆ ได้

รายละเอียดของแต่ละส่วนของโปรแกรม GDEYEs มีดังต่อไปนี้

3.2.1 ส่วนรับภาพ (Video Capture)

เป็นส่วนที่เชื่อมต่อกับการ์ดทีวีจูนเนอร์เพื่อรับภาพเข้ามาประมวลผลที่คอมพิวเตอร์หลัก ประกอบด้วยฟังก์ชันดังนี้

- ฟังก์ชันเปิดการติดต่อกับการ์ดทีวีจูนเนอร์มีโครงสร้างดังนี้

```
BOOL OpenCapDevice(int at_x, int at_y, CWnd *pParentWnd)
{
...
...
}
```

ฟังก์ชัน OpenCapDevice จะทำการสร้างวินโดว์ขึ้นมาเพื่อให้สามารถรับ windows message ได้ โดยพารามิเตอร์ที่ผ่านให้กับฟังก์ชันนี้ประกอบด้วยตำแหน่งเริ่มต้นของวินโดว์และ parent window

- ฟังก์ชันปิดการติดต่อกับการ์ดทีวีจูนเนอร์

ฟังก์ชันนี้จะปิดการติดต่อและทำลายวินโดว์ที่ได้สร้างด้วยฟังก์ชัน OpenCapDevice

```
BOOL CloseCapDevice()
{
   ...
   ...
}
```

- ฟังก์ชันรับภาพ

```
LPBITMAPINFO GetOneFrame()
{
   ...
   ...
}
```

ฟังก์ชันนี้จะคืนค่าพอยเตอร์ของตัวโครงสร้างชนิด BITMAPINFO ซึ่งประกอบด้วยรายละเอียดต่างๆ ของภาพแบบบิตแมพ แต่ที่จำเป็นในการประมวลผล

สำหรับโปรแกรมนี้คือ ตัวแปรที่เก็บภาพ ขนาดภาพและโหมดสี การเข้าถึงค่าเหล่านี้สามารถทำได้โดยการใช้มาโคร ดังต่อไปนี้

```
#define    IMAGEWIDTH(lpd)              \
(LPBITMAPINFOHEADER)lpd)->biWidth

#define    IMAGEHEIGHT(lpd)             \
((LPBITMAPINFOHEADER)lpd)->biHeight

#define    IMAGEBITS(lpd)               \
((LPBITMAPINFOHEADER)lpd)->biBitCount

#define    IMAGEDATA(lpd)               \
(((LPBYTE)lpd)+(((LPBITMAPINFOHEADER)lpd)->biSize))
```

โดยที่ IMAGEWIDTH และ IMAGEHEIGHT ใช้สำหรับหาค่าขนาดของภาพ

IMAGEBITS ใช้สำหรับหาโหมดสีของภาพ เช่น 24 บิต

IMAGEDATA ใช้สำหรับเข้าถึงตัวแปรที่เก็บภาพ

- ฟังก์ชันกำหนดรูปแบบของภาพ ได้แก่ ขนาดภาพ และ โหมดสี

```
BOOL VideoFormatDlg()
{
   ...
   ...
}
```

- ฟังก์ชันเลือก Capture Device

```
BOOL VideoSourceDlg()
{
   ...
   ...
}
```

- ฟังก์ชันตั้งค่ารูปแบบของภาพ

ก่อนจะเริ่มทำการประมวลผลภาพจะต้องกำหนดรูปแบบของภาพคือ ความกว้าง ความยาว และโหมดสีของภาพเพื่อให้เหมาะสมกับการประมวลผลภาพที่ใช้ มีโครงสร้างดังนี้

```
BOOL SetVideoFormat(int cx, int cy, DWORD dwResolution)
```

```
{
    ...
    ...
}
```

- ฟังก์ชันหาขนาดของหน่วยความจำที่ BITMAPINFO structure ใช้

```
int GetBITMAPINFOsize(LPBITMAPINFO lpBmpInfo)
{
    ...
    ...
}
```

3.2.2 ส่วนประมวลผลภาพ (Image Processor)

จากแนวคิดของโปรแกรมที่ได้กล่าวมาแล้วข้างต้น เราจะต้องสร้างให้แต่ละ document มีข้อมูลเป็นของตัวเอง โดยทั่วไปแล้วเมื่อทำการโหลด DLL ขึ้นมาจะมีการจองพื้นที่หน่วยความจำสำหรับตัวแบบ global แต่เราไม่สามารถใช้วิธีประกาศตัวแปรแบบ global ได้เนื่องจากในการโหลด DLL นั้นจะทำการจองพื้นที่หน่วยความจำเพียงครั้งแรกที่ทำการโหลดเท่านั้น ทำให้แต่ละ document ใช้ข้อมูลร่วมกัน ซึ่งไม่ตรงกับแนวคิดของโปรแกรมที่ได้กล่าวมาแล้วข้างต้น ดังนั้นเราจึงต้องสร้างฟังก์ชันสำหรับจองหน่วยความจำขึ้นมาให้แต่ละ document เรียกใช้

สำหรับการเลือกเปลี่ยนวิธีการประมวลผลในขณะทำงานนั้น Image Processor แต่ละตัวจะต้องมีฟังก์ชันหลักที่เหมือนกันเพื่อให้สามารถเรียกใช้ได้ในรูปแบบเดียวกัน การสร้างส่วนนี้สามารถทำได้ง่ายโดยใช้วิซาร์ด เริ่มต้นจากคอมไพล์โปรแกรม ImgPrsWizard ก่อนซึ่งอยู่ในไดเรกทอรี Tool ของ CD-Rom จะได้ Application Wizard เพิ่มขึ้นมา เมื่อทำการสร้างโปรเจ็คใหม่ให้เลือกชนิดโปรเจ็คเป็น Image Processor AppWizard หลังจากสร้างโปรเจ็คด้วย App Wizard ดังกล่าวแล้วจะพบฟังก์ชันต่างๆ ดังนี้

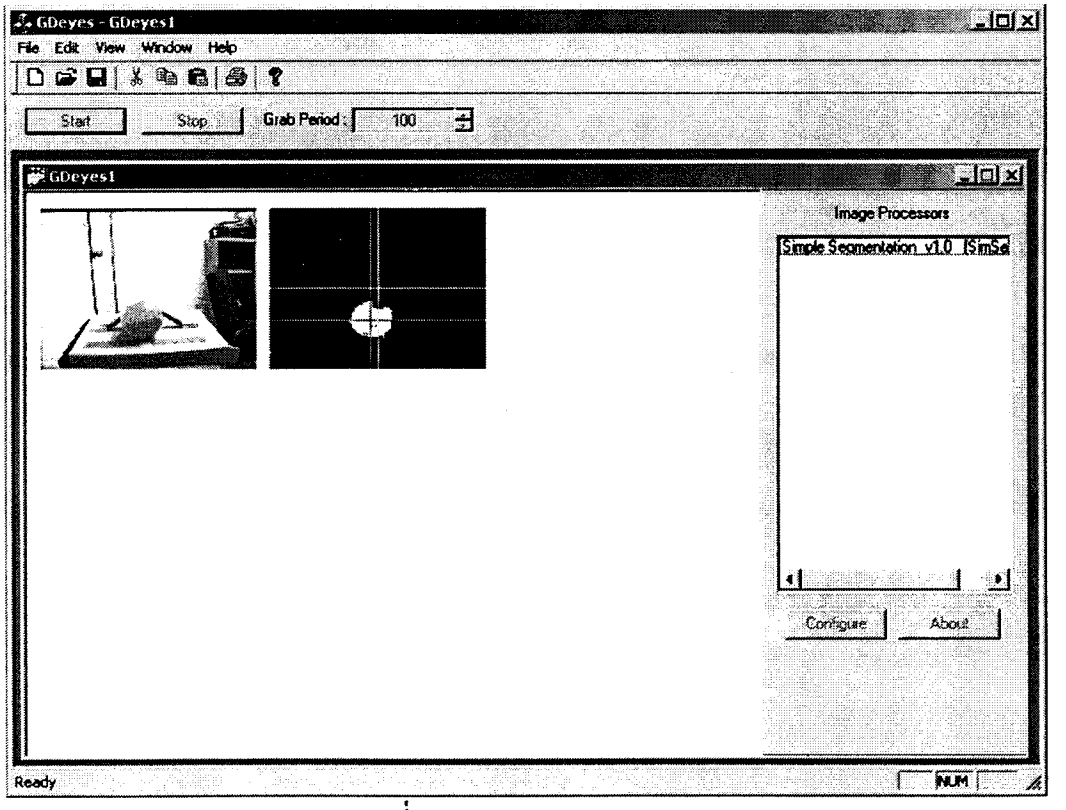
- ImgPrsProcessImage สำหรับประมวลผลภาพและสร้างคำสั่งเพื่อควบคุมหุ่นยนต์
- ImgPrsAllocConfigData สำหรับจองหน่วยความจำที่เก็บค่าพารามิเตอร์ต่างๆ การประกาศตัวแปรดังกล่าว จะทำในตัวโครงสร้าง SImgPrsData ที่อยู่ในไฟล์ ImgPrsData.h
- ImgPrsFreeConfigData สำหรับคืนหน่วยความจำที่เก็บตัวโครงสร้าง SImgPrsData ให้กับระบบเมื่อปิด document
- ImgPrsAbout สำหรับเพิ่มคำอธิบายของวิธีประมวลผล

- `ImgPrsConfig` ฟังก์ชันนี้จะโหลด dialog box สำหรับการกำหนดค่าขึ้นมาเท่านั้น ไม่ควรแก้ไขฟังก์ชันนี้ สำหรับการเพิ่มเติมส่วนกำหนดค่าต่างๆ จะทำในคลาส `CConfigDlg`

3.2.3 ส่วนควบคุมหุ่นยนต์ (Gundum Control)

ส่วนนี้เป็นส่วนที่ทำหน้าที่ควบคุมการทำงานของหุ่นยนต์ อาทิเช่น เดินหน้า ถอยหลัง เลี้ยวซ้าย-ขวา เป็นต้น การสร้างส่วนนี้จะทำในรูปแบบของไดรเวอร์โดยทำงานแบบ Kernel-mode ซึ่งวิธีการสร้างได้อธิบายไว้ในหัวข้อเรื่องการพัฒนาโปรแกรมไดรเวอร์

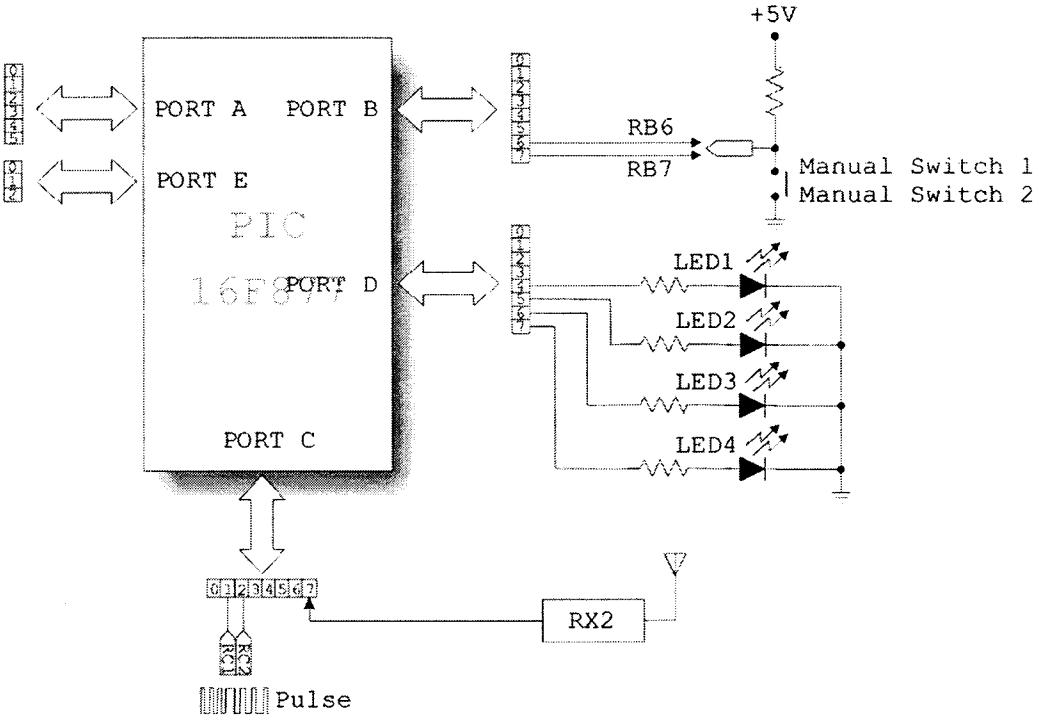
การเชื่อมต่อส่วนต่างๆ ที่กล่าวมาเข้ากับโปรแกรมหลักจะแตกต่างกันคือ ส่วนรับภาพและส่วนควบคุมหุ่นยนต์จะเป็นแบบ Implicit linking แต่ส่วนประมวลผลภาพจะเป็นแบบ explicit linking เพื่อให้สามารถเลือกโหลดเฉพาะวิธีการประมวลผลที่ต้องการขณะรันโปรแกรมได้ และทำให้สามารถเพิ่มวิธีการประมวลผลแบบอื่นๆ ที่ต้องการโดยไม่ต้องทำการแก้ไขส่วนโปรแกรมหลัก ซึ่งจะช่วยลดความซับซ้อนในการพัฒนาโปรแกรมโดยรวม รูปที่ 3.18 แสดงตัวอย่างหน้าจอของโปรแกรม GDEyes



รูปที่ 3.18 โปรแกรม GDEyes

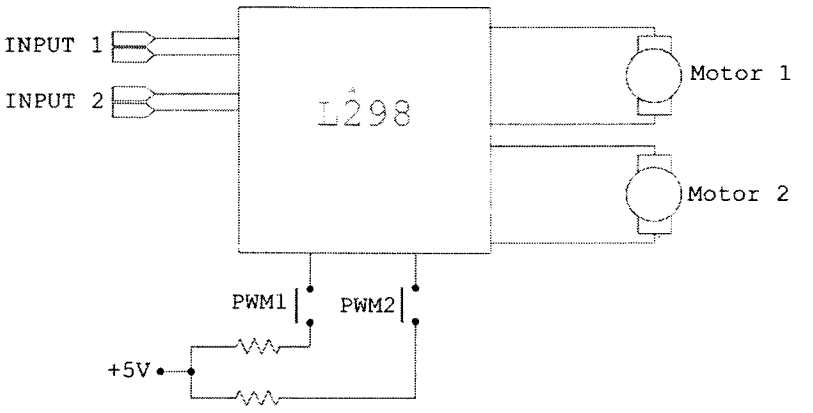
4. GUNDAM III

การพัฒนาหุ่นยนต์ GUNDAM ในช่วงที่ 3 เป็นการพัฒนาในส่วนของวงจรส่ง เนื่องจากเทคโนโลยีในการส่งข้อมูลแบบไร้สายได้มีการเปลี่ยนแปลงก้าวหน้า ทำให้ราคาต้นทุนในการซื้อวงจรรับส่งแบบไร้สายถูกลงและสามารถหาซื้อได้ภายในประเทศ ดังนั้นการปรับปรุงหุ่นยนต์ GUNDAM ในช่วงสุดท้ายนี้จึงได้เน้นไปที่การออกแบบวงจรควบคุมและส่งข้อมูลแบบใหม่ เนื่องจากวงจรควบคุมแบบเก่าใช้การส่งข้อมูลผ่าน D/A และ A/D ซึ่งใช้เชื่อมต่อกับวงจรภาครับ/ส่งของเครื่องบินเล็ก การเชื่อมต่อแบบเก่าจึงไม่เหมาะสมกับวงจรรับส่งแบบใหม่ ซึ่งใช้มาตรฐานในการส่งข้อมูลแบบอนุกรมหรือ RS-232 วงจรควบคุมจึงต้องถูกออกแบบสร้างใหม่เพื่อให้รองรับการใช้งานร่วมกับการส่งข้อมูลแบบ RS-232 รายละเอียดของวงจรควบคุมและวงจรส่งข้อมูลใหม่มีดังต่อไปนี้



รูปที่ 3.19 วงจรควบคุมและวงจรส่งข้อมูล

วงจรในรูปที่ 3.19 แสดงวงจรควบคุมด้วยไมโครคอนโทรลเลอร์ตระกูล PIC เบอร์ 16F877 ซึ่งเป็นชิพที่นิยมใช้ในการประยุกต์ใช้ไมโครคอนโทรลเลอร์ทั่วๆ ไป นอกเหนือไปจากส่วนส่งข้อมูลด้วยมาตรฐาน RS-232 ผ่านทางขา 7 ของ PORT C แล้ว วงจรดังกล่าวยังได้รับการออกแบบให้สามารถส่งสัญญาณพัลส์ เพื่อใช้ในการป้อนให้กับไอซีขับกระแสสำหรับขับเคลื่อนมอเตอร์โดยสามารถปรับความเร็วของมอเตอร์ได้ ดังแสดงในรูปที่ 3.20



รูปที่ 3.20 วงจรขับมอเตอร์ด้วยการปรับขนาดความกว้างของสัญญาณพัลส์ (PWM)

วงจรส่งข้อมูลแบบไร้สายใช้ชิพของ RadioMatrix เบอร์ TX2-433-160 สำหรับตัวส่งและ RX2-433-160 สำหรับตัวรับ วงจรดังกล่าวสามารถส่งข้อมูลด้วยมาตรฐาน RS-232 ได้ที่ความเร็วประมาณ 160 kbps ในระยะทางประมาณ 300 เมตร (ในที่โล่ง) การส่งข้อมูลใช้คลื่นความถี่ย่าน UHF (433 MHz) และการเข้ารหัสแบบ FM เนื่องจากการนำเอาตัวส่งข้อมูลแบบไร้สายมาใช้แทนตัวส่งสัญญาณที่ดัดแปลงมาจากเครื่องบังคับเครื่องบินเล็ก ทำให้ไม่ต้องใช้การเทคนิค D/A และ A/D ในการส่งข้อมูล มีผลให้การส่งข้อมูลเพื่อทำการควบคุมหุ่นยนต์ GUNDAM จากคอมพิวเตอร์มีประสิทธิภาพเพิ่มขึ้น

นอกไปจากการปรับปรุงวงจรควบคุมและวงจรส่งข้อมูลข้างต้นแล้ว ใน GUNDAM III ยังมีการเปลี่ยนแปลงในส่วนของวงจรรับส่งภาพ ซึ่งวงจรใหม่ใช้ความถี่ 2.4 GHz มีขนาดวงจรที่เล็กลงและประสิทธิภาพในการรับส่งภาพที่ดีขึ้น

5. สรุป

หุ่นยนต์เคลื่อนที่ GUNDAM ได้มีการออกแบบและพัฒนาเป็น 3 ช่วงใหญ่ๆ โดยในช่วงแรกเป็นการออกแบบเชิงกลของตัวหุ่นยนต์ GUNDAM I ให้มีความสามารถในการเคลื่อนที่ในที่ราบเรียบได้อย่างเหมาะสม โดยมีการออกแบบให้มีความสามารถในการปรับโครงสร้างให้มีความเที่ยงตรง การออกแบบให้มีการขับเคลื่อนแบบ 4 ล้อ เพื่อความสะดวกในการบังคับเลี้ยว รวมไปถึงการส่งข้อมูลแบบไร้สายที่นำเอาระบบบังคับเครื่องบินเล็กมาประยุกต์ใช้ ที่ซึ่งเทคโนโลยีในการออกแบบวงจรส่งข้อมูลแบบไร้สายในขณะที่ทำการพัฒนาในช่วงแรกนี้ยังไม่ถือว่าเหมาะสมที่สุด

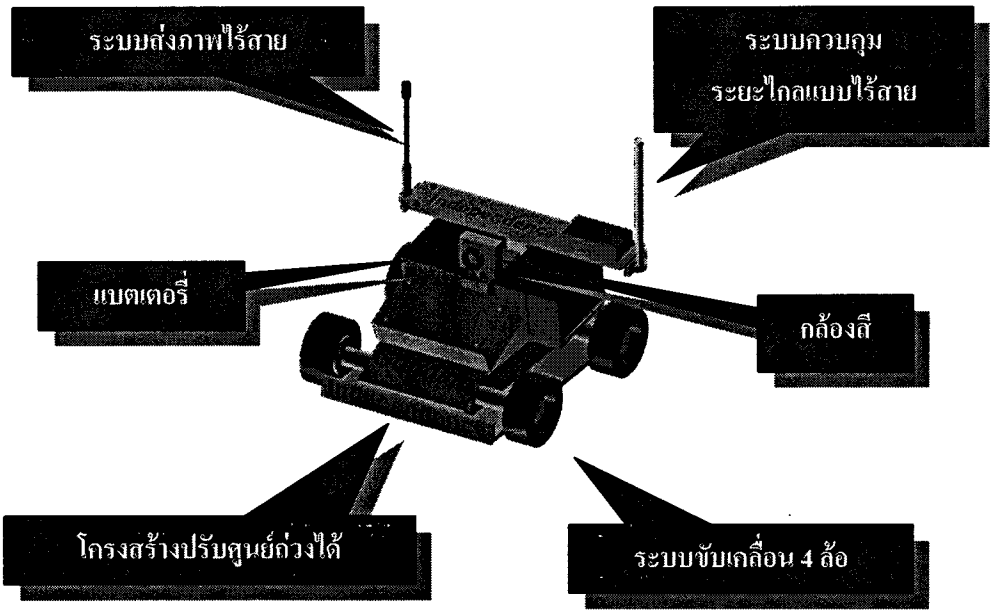
ในช่วงที่สองเป็นการปรับปรุงเชิงโปรแกรม โดยเน้นการพัฒนาโปรแกรมไดรเวอร์สำหรับเป็นส่วนเชื่อมต่อระหว่างการควบคุมจากคอมพิวเตอร์กับตัวหุ่นยนต์ GUNDAM II ภายใต้ระบบปฏิบัติการ WINDOWS 2000 นอกไปจากนั้นแล้วยังได้มีการพัฒนาโปรแกรมประยุกต์ในการทดสอบการประมวลผลสัญญาณภาพจากตัวหุ่นยนต์ โดยรายละเอียดในส่วนนี้มีอยู่ในบทต่อไป

ช่วงที่สามเป็นการปรับปรุงวงจรควบคุมและวงจรส่งข้อมูล โดยมีการเลือกใช้วงจรที่ทันสมัย สามารถส่งข้อมูลแบบไร้สายด้วยมาตรฐาน RS-232 เพื่อเพิ่มประสิทธิภาพในการเชื่อมต่อระหว่างคอมพิวเตอร์กับตัวหุ่นยนต์ นอกไปจากนั้นแล้วยังมีการปรับปรุงวงจรขับเคลื่อนมอเตอร์ โดยมีการใช้เทคนิคปรับความกว้างของสัญญาณพัลส์ในการปรับความเร็วของมอเตอร์ ในส่วนของวงจรรับส่งภาพก็มีการเลือกใช้วงจรใหม่ขนาด 2.4 GHz ซึ่งมีขนาดวงจรเล็กลงและมีประสิทธิภาพในการรับส่งภาพที่ดีขึ้น เทคโนโลยีวงจรต่างๆ ที่เลือกใช้ใน GUNDAM III เป็นเทคโนโลยีที่ได้มีการพัฒนาใช้ในช่วงเวลาภายหลัง (หลังปีพ.ศ. 2545) ทำให้มีตัวเลือกการใช้องค์ประกอบต่างๆ ของหุ่นยนต์ได้อย่างมีประสิทธิภาพเพิ่มขึ้นได้

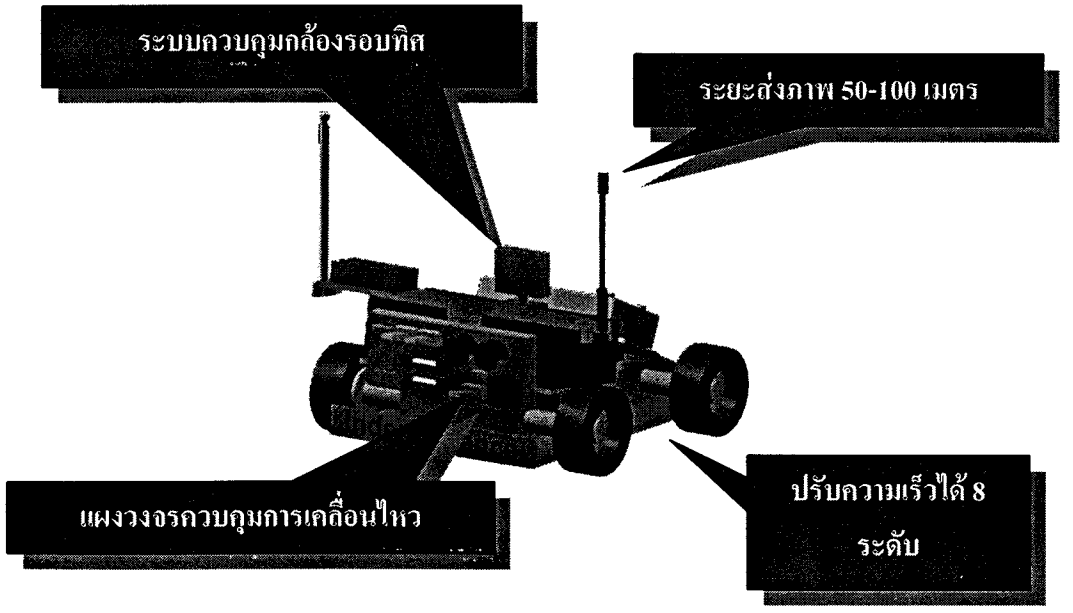
บทที่ 4
ผลการทดลองประยุกต์ใช้งานหุ่นยนต์เคลื่อนที่ GUNDAM

1. บทนำ

หุ่นยนต์ GUNDAM ที่สร้างเสร็จสมบูรณ์มีคุณลักษณะและโครงสร้างที่สำคัญดังแสดงในรูปที่ 4. 1รูปที่ 4. 2 ซึ่งเป็นคุณสมบัติของหุ่นยนต์ GUNDAM I



รูปที่ 4. 1 คุณลักษณะสำคัญของหุ่นยนต์ GUNDAM I (ส่วนที่ 1)



รูปที่ 4. 2 คุณลักษณะสำคัญของหุ่นยนต์ GUNDAM I (ส่วนที่ 2)

หุ่นยนต์ GUNDAM I สามารถควบคุมได้จากคอมพิวเตอร์แบบไร้สายในระยะทางประมาณ 100 – 200 เมตรในที่โล่ง ถ้าสภาวะแวดล้อมเปลี่ยนไป เช่นอยู่ภายในอาคาร ที่ซึ่งมีผนังกั้นระหว่างตัวหุ่นยนต์ GUNDAM และคอมพิวเตอร์ควบคุม ระยะทางในการควบคุมแบบไร้สายจะลดลงตามสภาพ แบตเตอรี่ของหุ่นยนต์ GUNDAM มีทั้งหมด 4 ก้อน โดยแบ่งแบตเตอรี่ 2 ก้อนสำหรับมอเตอร์ขับเคลื่อน แบตเตอรี่อีก 2 ก้อนใช้สำหรับวงจรควบคุมและส่งข้อมูลแบบไร้สาย และวงจรส่งภาพวีดีโอแบบไร้สายอย่างละก้อน การส่งภาพวิดีโอสามารถส่งได้ในระยะทางประมาณ 20 เมตร ขึ้นอยู่กับสภาพแวดล้อม มอเตอร์ขับเคลื่อนแบบ 4 ล้อ พร้อมโครงสร้างที่สามารถปรับศูนย์ถ่วงได้ ความเร็วในการเคลื่อนที่ของหุ่นยนต์สามารถปรับได้ทั้งหมด 8 ระดับ โดยปรับระดับได้จากโปรแกรมควบคุมจากคอมพิวเตอร์ ข้อมูลจำเพาะของหุ่นยนต์ GUNDAM I สามารถสรุปได้ดังต่อไปนี้

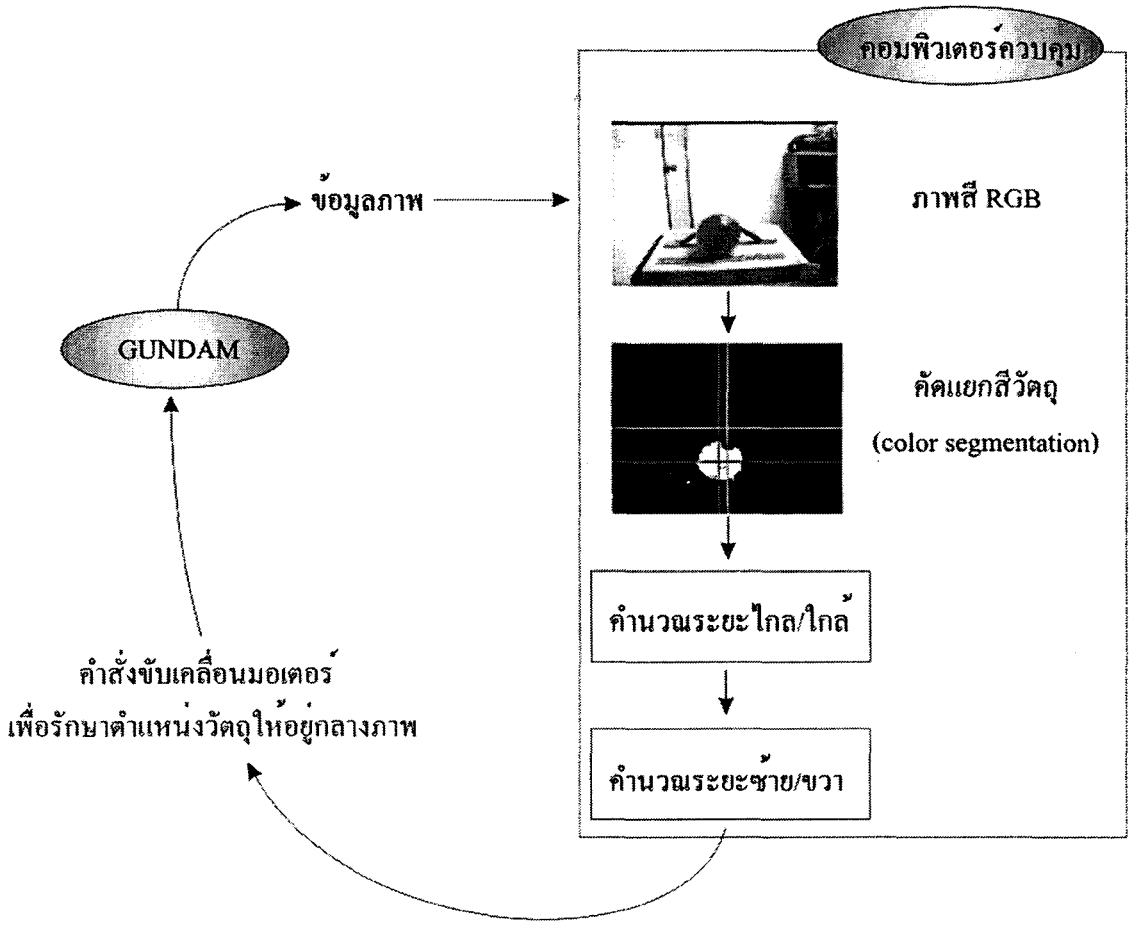
- ด้านหน้ากว้าง 29.0 ซม. ด้านข้างยาว 31.5 ซม.
- น้ำหนักตัว 6 กก.
- ระบบขับเคลื่อนแบบ 4 ล้อ
- ปรับความเร็วในการเคลื่อนที่ได้ 8 ระดับด้วยเทคนิค PWM(Pulse Wide Modulation)
- ปรับศูนย์ล้อ 4 ล้อได้อิสระ เพื่อให้การติดตั้งล้อมีความเที่ยงตรงสูง
- โครงสร้างปรับศูนย์ได้ เพื่อความเที่ยงตรงในการขับเคลื่อน
- ตัวหุ่นควบคุมด้วยไมโครคอนโทรลเลอร์ ตระกูล AVR เบอร์ AT90S2313-10PC เพื่อให้มีความยืดหยุ่นในการทำงานและเพื่อรองรับการพัฒนาปรับปรุงในอนาคต
- ระบบขับเคลื่อนควบคุมจากคอมพิวเตอร์ด้วยระบบไร้สาย
- ความถี่ที่ในการติดต่อกับคอมพิวเตอร์ควบคุมใช้อยู่ในย่าน 35 MHz.
- ระยะรัศมีที่สามารถควบคุมได้มีค่ามากกว่า 100 ม. ในที่โล่ง
- ระยะรัศมีที่สามารถรับสัญญาณภาพได้มีค่ามากกว่า 20 ม.ในที่โล่ง
- สัญญาณภาพที่ใช้อยู่ในย่าน UHF
- ใช้ระบบภาพสี เพื่อการประมวลผลข้อมูลที่ดีกว่า
- มีระบบควบคุมด้วยมือ (manual) จากคอมพิวเตอร์และระบบควบคุมแบบอัตโนมัติ ด้วยเทคนิคการประมวลผลภาพเชิงปัญญาประดิษฐ์
- มีระบบควบคุมแบบพิเศษผ่านอินเตอร์เนต (เป็นการทดลองประยุกต์เพื่อพัฒนาการควบคุมหุ่นยนต์ระยะทางไกลผ่านอินเตอร์เนต รายละเอียดไม่ได้นำเสนอไว้ในงานวิจัยฉบับนี้)

หุ่นยนต์ GUNDAM II เพิ่มการพัฒนาโปรแกรมควบคุมจากคอมพิวเตอร์ พร้อมกับตัวอย่างการประยุกต์ใช้การประมวลผลสัญญาณภาพในการควบคุม ซึ่งรายละเอียดผลการทดลองมีอยู่ในตอนต่อไปนี้

หุ่นยนต์ GUNDAM III ปรับปรุงวงจรการควบคุมและส่งข้อมูลแบบไร้สายใหม่ โดยสามารถส่งข้อมูลได้ในระยะทางประมาณ 300 เมตร ด้วยมาตรฐาน RS-232 นอกไปจากนั้นแล้ว การปรับระดับความเร็วยังสามารถทำได้หลายระดับ โดยทำการปรับความกว้างของสัญญาณพัลส์ที่ใช้ในการขับมอเตอร์ ที่ซึ่งการปรับความกว้างของสัญญาณพัลส์ดังกล่าวสามารถตั้งค่าในโปรแกรมให้มีจำนวนระดับตามที่ต้องการได้ ระบบส่งภาพวีดีโอไร้สายของ GUNDAM III ยังได้รับการปรับปรุง โดยนำเอาตัวรับส่งภาพไร้สายด้วยความถี่ 2.4GHz มาใช้แทนวงจรส่งภาพแบบเก่า วงจรรับส่งภาพใหม่นี้มีขนาดวงจรที่เล็กลงและมีประสิทธิภาพในการรับส่งภาพวีดีโอที่ดีกว่า ในรูปที่ 4.19 และรูปที่ 4.20 แสดงภาพถ่ายมุมต่างๆ ของหุ่นยนต์ GUNDAM III ที่สร้างเสร็จเรียบร้อยแล้ว

2. การประยุกต์ใช้การประมวลผลภาพในการควบคุมหุ่นยนต์

เพื่อการทดสอบประสิทธิภาพในการใช้การประมวลผลภาพในการควบคุมหุ่นยนต์ ในงานวิจัยนี้จึงได้ทดลองออกแบบให้หุ่นยนต์สามารถเคลื่อนที่ติดตามวัตถุได้อย่างอัตโนมัติ (object tracking) การติดตามวัตถุจะใช้ข้อมูลสีของวัตถุเป็นหลัก ขั้นตอนการติดตามวัตถุโดยรวมมีดังรูปที่ 4.3 กล้องวิดีโอจาก GUNDAM จะทำการส่งภาพกลับมายังคอมพิวเตอร์ควบคุม ซึ่งสำหรับ GUNDAM I จะทำการจับภาพเข้าสู่คอมพิวเตอร์ผ่านทางการ์ดทีวีจูนเนอร์ (TV tuner card) ในขณะที่ GUNDAM III จะทำการจับภาพเข้าสู่คอมพิวเตอร์โดยการรับภาพจากตัวรับสัญญาณวีดีโอ ผ่านการ์ดจับสัญญาณภาพเข้าสู่คอมพิวเตอร์ (video capture card) โปรแกรมประมวลผลภาพจะทำการคัดแยกสีของวัตถุโดยสีดังกล่าวจะได้รับการกำหนดไว้ก่อน เมื่อทำการคัดแยกสีของวัตถุที่ได้แล้ว จะทำการคำนวณหาคำสั่งขับเคลื่อนมอเตอร์เพื่อทำการขับเคลื่อนหุ่นยนต์ให้ทำการติดตามวัตถุ โดยที่หุ่นยนต์จะทำการเคลื่อนที่เพื่อให้วัตถุที่ถูกติดตามอยู่ภายในจุดศูนย์กลางของภาพเสมอ คำสั่งขับเคลื่อนที่ได้จะถูกส่งกลับไปยังหุ่นยนต์เพื่อทำการขับเคลื่อนมอเตอร์ต่อไป รายละเอียดของการประมวลผลภาพเพื่อทำการควบคุมหุ่นยนต์มีดังต่อไปนี้ (วิธีการประมวลผลภาพที่ใช้เป็นเทคนิคพื้นฐานทั่วๆ ไป โดยมีจุดประสงค์เพื่อทดสอบประสิทธิภาพการทำงานของหุ่นยนต์ GUNDAM เท่านั้น)

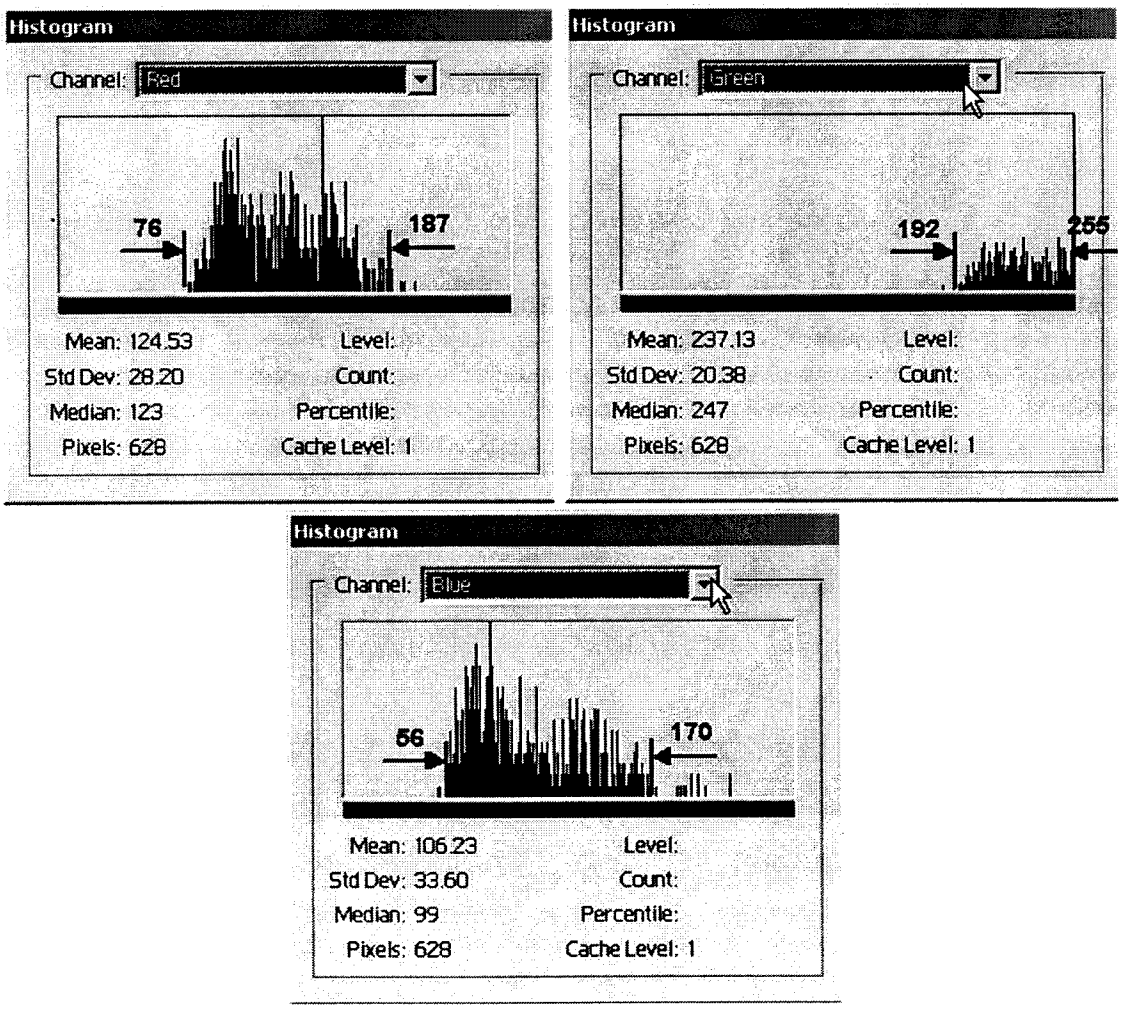


รูปที่ 4. 3 การติดตามวัตถุด้วยสี

2.1 ทำการสร้างแบบจำลองสีของวัตถุ โดยทำการบันทึกภาพที่รับมาจากการ์ดจับสัญญาณภาพเข้าสู่คอมพิวเตอร์ แล้วนำไปเปิดด้วยโปรแกรม Photoshop เพื่อหาฮิสโตแกรม โดยการเลือกเฉพาะส่วนของภาพที่เป็นวัตถุด้วย Lasso Tool ดังรูปที่ 4.4 แล้วคลิ๊กที่เมนู Image->Histogram เพื่ออ่านค่าช่วงสีของแต่ละแม่สี (แดง เขียวและน้ำเงิน) ของวัตถุที่ต้องการดังแสดงรูปที่ 4.5 ตัวอย่างค่าแม่สีที่อ่านได้มีดังนี้ (ช่วงของค่าสีคือ 0 – 255 โดยตัวอย่างของวัตถุที่ใช้ในที่นี้คือลูกบอลสีเขียว)

สีแดง	76-187
สีเขียว	192-255
สีน้ำเงิน	56-170

รูปที่ 4.4 การเลือกพื้นที่เฉพาะส่วนของวัตถุที่ต้องการโดยใช้ Lasso Tool

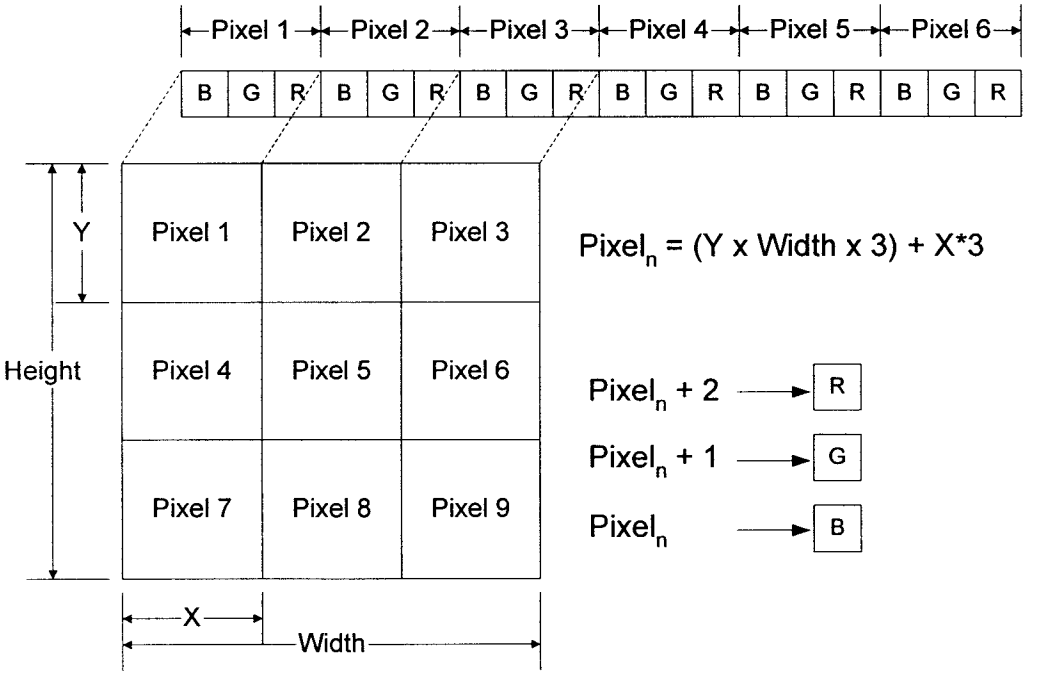


รูปที่ 4.5 ตัวอย่างฮิสโตแกรมสีแดง สีเขียวและสีน้ำเงินของลูกบอล

2.2 ทำการตรวจสอบพิกเซลทั้งหมดในภาพว่ามีพิกเซลใดบ้างที่อยู่ในช่วงของแบบจำลองสีที่กำหนดไว้ โดยทำการเปรียบเทียบทั้งสามแม่สีของแบบจำลองสี ถ้าพิกเซลใดอยู่ในแบบจำลองสีของวัตถุให้เปลี่ยนเป็นสีขาว ถ้าพิกเซลใดไม่อยู่ให้ทำการเปลี่ยนเป็นสีดำ ขั้นตอนนี้เป็นการคัดแยกสีนั่นเอง (color segmentation) ในการโปรแกรมนั้นเราจะต้องเข้าใจถึงวิธีการจัดเก็บภาพในหน่วยความจำเสียก่อน ซึ่งการเก็บภาพจะใช้วิธีเก็บเป็นแบบอาร์เรย์ของพิกเซล แต่ละพิกเซลจะประกอบด้วย 3 สีเรียงต่อกันโดยเรียงกลับกันเป็น GBR (ไม่ใช่ RGB) ดังนั้นจะต้องใช้อาร์เรย์ขนาด $3N$ โดยที่

N = จำนวนพิกเซลทั้งหมดของภาพ = ความกว้างของภาพ (พิกเซล) x ความยาวของภาพ (พิกเซล)

ปกติระบบสีที่ใช้ในการประมวลผลด้วยคอมพิวเตอร์ทั่วๆ ไปจะใช้ระดับสีขนาด 24 บิต (16.7 ล้านระดับสี) ดังนั้น 1 พิกเซลจะใช้หน่วยความจำขนาด 24 บิต โดยเก็บสีแดง สีเขียวและสีน้ำเงินอย่างละ 8 บิต ทำให้ขนาดหน่วยความจำที่ต้องใช้เท่ากับ $3N$ ไบต์ การเข้าถึงพิกเซลใดๆในภาพจะต้องคำนวณหาจุดเริ่มต้นของพิกเซลนั้นก่อน ดังแสดงในรูปที่ 4.6



รูปที่ 4.6 การจัดเก็บภาพในหน่วยความจำและการคำนวณหาตำแหน่งพิกเซล

ในการเปรียบเทียบสีจะทำทุกพิกเซลและทุกแม่สี โดยการเข้าถึงข้อมูลแต่ละสีของพิกเซลจะใช้การอ้างแบบพอยน์เตอร์ เนื่องจากตัวแปรที่เก็บภาพเป็นตัวแปรชนิดพอยน์เตอร์ การเข้าถึงข้อมูลสีดังกล่าวสามารถเขียนได้ 2 วิธี คือ

ImageData[Pixel$_n$+2] สำหรับสีแดง

ImageData[Pixel$_n$+1] สำหรับสีเขียว

ImageData[Pixel$_n$] สำหรับสีน้ำเงิน

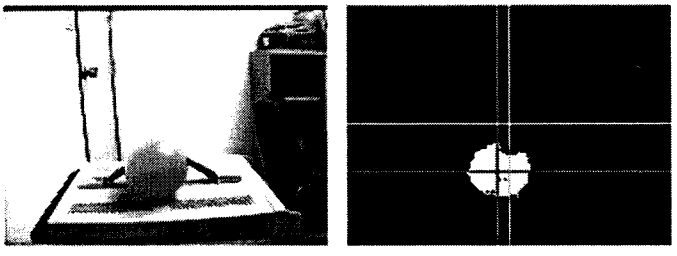
หรือ

*(ImageData+Pixel$_n$+2) สำหรับสีแดง

*(ImageData+Pixel$_n$+1) สำหรับสีเขียว

*(ImageData+Pixel$_n$) สำหรับสีน้ำเงิน

เนื่องจากการประมวลผลจะต้องดำเนินการกับทุกพิกเซล ขนาดภาพที่ใหญ่จะทำให้ใช้เวลาในการประมวลผลมาก ดังนั้นในที่นี้จะใช้ภาพที่มีขนาดเล็กเพื่อให้สามารถประมวลผลได้อย่างรวดเร็ว โดยขนาดที่เลือกใช้ในงานนี้คือ 160 x 120 พิกเซล ผลการคัดแยกสีมีแสดงในรูปที่ 4.7



รูปที่ 4.7 ผลการทำคัดแยกสี (color segmentation)

2.3 ทำการคำนวณระยะห่างของวัตถุจากจุดศูนย์กลางของภาพ จากภาพที่ผ่านการคัดแยกสีแล้ว ตำแหน่งของวัตถุเทียบกับภาพในหน่วยของพิกเซลสามารถคำนวณได้ดังนี้ (วิธีนี้เป็นวิธีการหาจุดศูนย์ถ่วงของวัตถุหรือ center of gravity)

- นับจำนวนพิกเซลทั้งหมดที่มีค่าเป็นสีขาว เก็บค่าไว้ในตัวแปร `WhitePixels`
- หาผลรวมของพิกัดในแนวแกน X และ ผลรวมของพิกัดในแนวแกน Y ของพิกเซลสีขาวทั้งหมด เก็บค่าไว้ในตัวแปร `SumX` และ `SumY` ตามลำดับ
- หาตำแหน่งของวัตถุโดยการนำผลรวมของพิกัดในแนวแกน X และ Y หารด้วยจำนวนพิกเซลสีขาวดังนี้

```
CenterX = SumX/WhitePixels
CenterY = SumY/WhitePixels
```

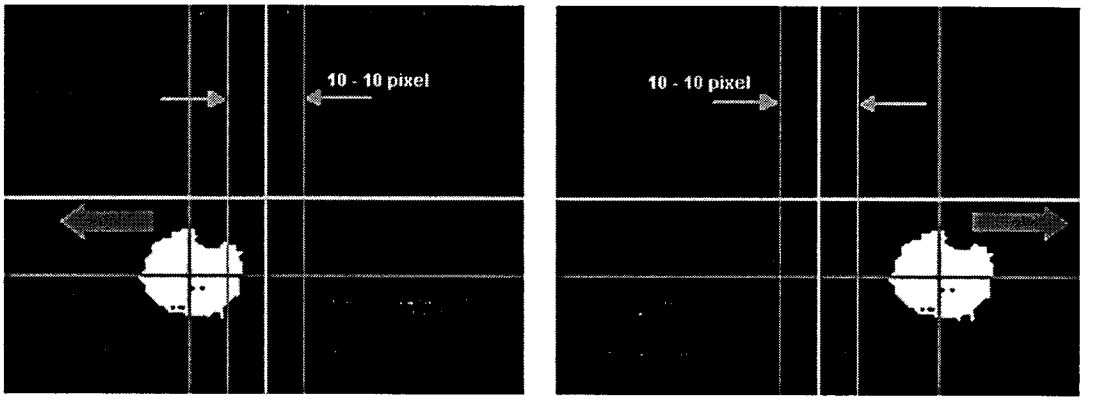
- คำนวณหารัศมีโดยใช้สูตรการหาพื้นที่ของวงกลม

```
Radius = sqrt(WhitePixels/PI)
```

2.4 ทำการคำนวณหาคำสั่งขับเคลื่อนมอเตอร์เพื่อควบคุมให้วัตถุอยู่ภายในจุดศูนย์กลางของภาพเสมอ (ตัวหุ่นยนต์จะเกิดการเคลื่อนที่ ภาพที่รับจากกล้องจะเปลี่ยนไปตามการเคลื่อนที่) การขับเคลื่อนมอเตอร์จะขึ้นอยู่กับตำแหน่งปัจจุบันของวัตถุในภาพ กล่าวคือถ้าวัตถุอยู่ทางฝั่งซ้ายของภาพ (เทียบกับจุดศูนย์กลางของภาพ) หุ่นยนต์จะทำการเคลื่อนที่ไปทางซ้าย ถ้าวัตถุอยู่ทางฝั่งขวาของภาพ หุ่นยนต์จะทำการเคลื่อนที่ไปทางขวา การบังคับเลี้ยวซ้าย-ขวาจะใช้แกนตั้ง (แกน Y) สำหรับการเปรียบเทียบกับตำแหน่งของวัตถุ ดังแสดงในรูปที่ 4.8

- แกนตั้งห่างกันเกิน 10 พิกเซลและวัตถุอยู่ด้านซ้าย (รูปที่ 4.8 ก) ให้เลี้ยวขวา
- แกนตั้งห่างกันเกิน 10 พิกเซลและวัตถุอยู่ด้านขวา (รูปที่ 4.8 ข) ให้เลี้ยวซ้าย

การเลี้ยวจะทำจนกระทั่งแกนตั้งห่างกันน้อยกว่า 10 พิกเซล จากนั้นจึงทำการบังคับเดินหน้า-ถอยหลัง

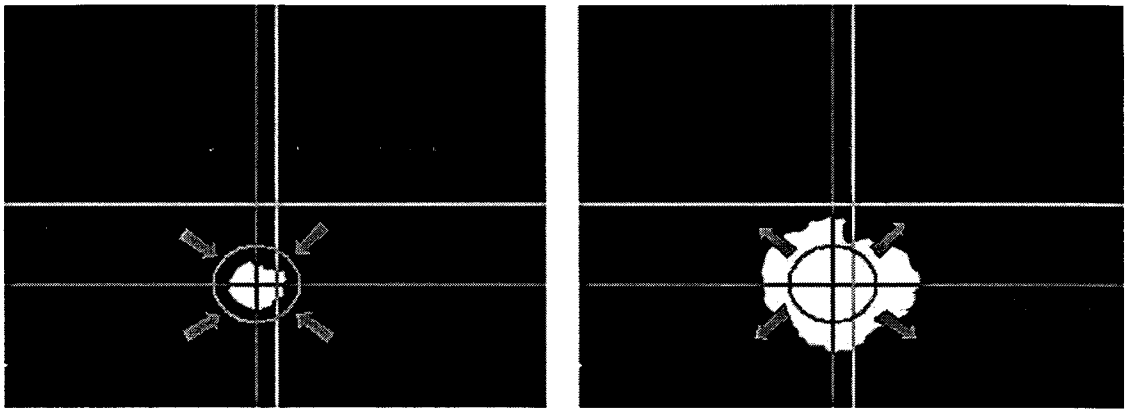


(ก) ให้เลี้ยวซ้าย (ข) ให้เลี้ยวขวา

รูปที่ 4.8 การบังคับเลี้ยวซ้าย-ขวา

หลังจากขับเคลื่อนหุ่นยนต์เพื่อให้วัตถุอยู่ในย่านศูนย์กลางของภาพแล้ว ในกรณีที่วัตถุเคลื่อนที่ใกล้-ไกลจากตัวหุ่นยนต์ หุ่นยนต์จะต้องวิ่งตามการเคลื่อนที่ของวัตถุ การขับเคลื่อนหุ่นยนต์ในที่นี้จะเป็นการบังคับให้หุ่นยนต์เดินหน้า-ถอยหลัง การบังคับเดินหน้า-ถอยหลังใช้ขนาดรัศมีที่คำนวณได้มาเปรียบเทียบกับรัศมีในแบบจำลองที่สร้างขึ้น กล่าวคือเมื่อขนาดของวัตถุในภาพมีขนาดเล็กลง แสดงว่าวัตถุเคลื่อนที่ออกห่างจากตัวหุ่นยนต์ ดังนั้นหุ่นยนต์จะเคลื่อนที่เดินหน้าเพื่อติดตามวัตถุ ในทางตรงกันข้าม เมื่อขนาดของวัตถุในภาพมีขนาดใหญ่ขึ้น แสดงว่าวัตถุเคลื่อนที่เข้าใกล้ตัวหุ่นยนต์ หุ่นยนต์จะเคลื่อนที่ถอยหลังเพื่อคงระดับความห่างระหว่างตัวหุ่นยนต์กับวัตถุไว้ระยะหนึ่ง คำสั่งในการขับเคลื่อนเพื่อเดินหน้า-ถอยหลังจะใช้รัศมีของวัตถุที่คำนวณไว้เรียบร้อยแล้วในการพิจารณาแทนขนาดของวัตถุ ดังแสดงในรูปที่ 4.9 ตามเงื่อนไขดังต่อไปนี้

- ขนาดภาพวัตถุ < ขนาดแบบจำลอง (รูปที่ 4.9-ก) ให้เดินหน้า
- ขนาดภาพวัตถุ > ขนาดแบบจำลอง (รูปที่ 4.9-ข) ให้ถอยหลัง



(ก) ให้เดินหน้า (ข) ให้ถอยหลัง

รูปที่ 4.9 การบังคับเดินหน้า-ถอยหลัง

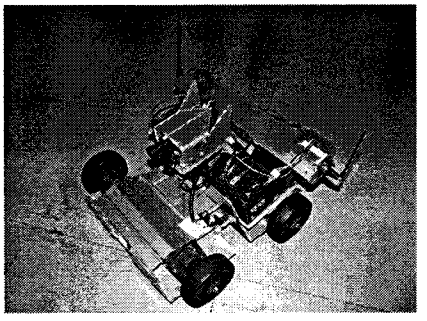
ผลการทดสอบการติดตามวัตถุด้วยสีให้ผลเป็นที่พอใจ โดยหุ่นยนต์สามารถรับรู้และคัดแยกสีของวัตถุได้ถูกต้อง ทำให้สามารถเคลื่อนที่ติดตามวัตถุได้อย่างถูกต้อง

3. สรุป

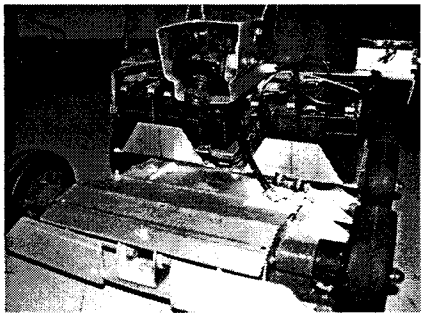
หุ่นยนต์ GUNDAM สามารถทำงานได้ตามวัตถุประสงค์ที่ออกแบบไว้ โดยใช้การควบคุมที่เน้นวิธีการทางปัญญาประดิษฐ์ นั่นคือใช้การประมวลผลสัญญาณภาพในการหาตำแหน่งของวัตถุที่ต้องการติดตามและทำการแปลงเป็นคำสั่งขับเคลื่อนหุ่นยนต์ ให้ติดตามวัตถุได้อย่างถูกต้อง โดยปกติแล้วหุ่นยนต์เคลื่อนที่จะใช้การประมวลผลสัญญาณภาพจากกล้องบนตัวหุ่นยนต์ เป็นข้อมูลหลักในการตัดสินใจในการเคลื่อนที่ของหุ่นยนต์ (active vision) ถึงแม้ว่าจะมีการติดตั้งตัวตรวจจับประเภทอื่นๆ เพื่อช่วยการประมวลผลตัดสินใจในการเคลื่อนที่เช่นตัวตรวจจับระยะทางอัลตราโซนิคหรืออินฟาเรด เป็นต้น การประมวลผลยังคงใช้การประมวลผลข้อมูลภาพจากกล้องเป็นหลัก

ในทางปฏิบัติโดยทั่วไป การประมวลผลสัญญาณภาพเป็นขบวนการที่ต้องใช้เวลาในการคำนวณมาก นอกไปจากนั้นโดยส่วนใหญ่แล้ว มาตรฐานการเชื่อมต่อระหว่างอุปกรณ์จับภาพจะเป็นการเชื่อมต่อกับคอมพิวเตอร์ส่วนบุคคล หน่วยประมวลผลที่ใช้จึงจำเป็นจะต้องมีศักยภาพเพียงพอที่จะทำการคำนวณข้อมูลภาพที่มีปริมาณมากและสามารถเชื่อมต่อกับอุปกรณ์จับภาพ ดังนั้นแล้วในงานวิจัยนี้ การประมวลผลสัญญาณภาพจึงได้ถูกออกแบบให้ทำการส่งข้อมูลภาพไร้สาย จากตัวหุ่นยนต์ไปยังหน่วยประมวลผลซึ่งเป็นคอมพิวเตอร์ส่วนบุคคล เมื่อทำการประมวลผลแล้วคอมพิวเตอร์ควบคุมจะทำการส่งคำสั่งขับเคลื่อนกลับไปที่หุ่นยนต์ เพื่อทำการเคลื่อนที่ตาม

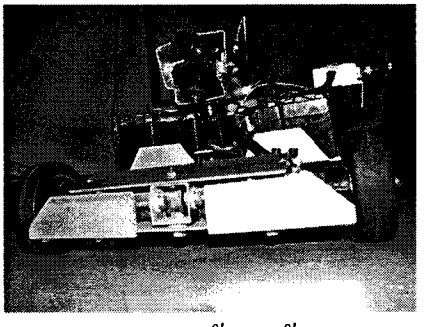
วัตถุประสงค์ต่อไป หุ่นยนต์ GUNDAM สามารถปฏิบัติงานหลักๆ ของการประยุกต์ใช้การประมวลผลสัญญาณภาพในการควบคุมการทำงานของหุ่นยนต์ได้ ประสิทธิภาพในเชิงความเร็วของผลตอบสนองจะขึ้นอยู่กับคุณภาพของฮาร์ดแวร์ ดังนั้นแล้วหุ่นยนต์ GUNDAM สามารถทำงานได้ในระดับหนึ่ง สำหรับการประมวลผลข้อมูลที่มีปริมาณมากและซับซ้อน หรือการตอบสนองต่อสภาวะแวดล้อมที่เปลี่ยนแปลงอย่างยุ่งยากหรือรวดเร็ว อาจจะมีผลต่อประสิทธิภาพที่ด้อยลงบ้าง อย่างไรก็ดี หุ่นยนต์ GUNDAM ได้แสดงการทำงานได้ตรงตามเป้าหมายที่ออกแบบไว้ จึงเพียงพอต่อการนำเอาไปประยุกต์ใช้งานทางด้านปัญญาประดิษฐ์อื่นๆ ต่อไป



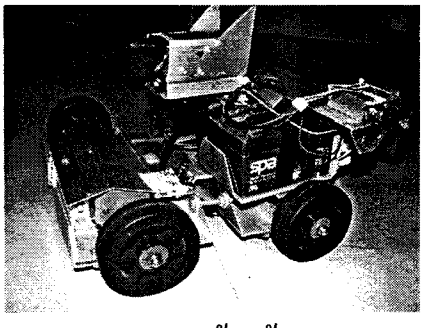
(ก) GUNDAM III



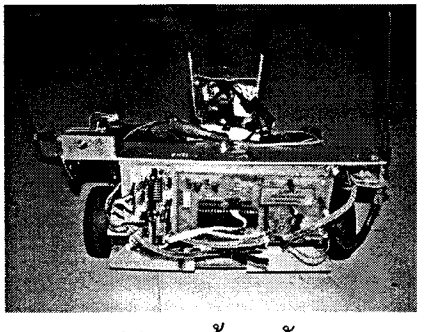
(ข) ภาพถ่ายใกล้



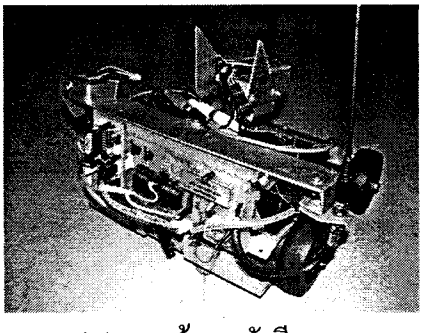
(ค) ภาพด้านหน้า



(ง) ภาพด้านข้าง

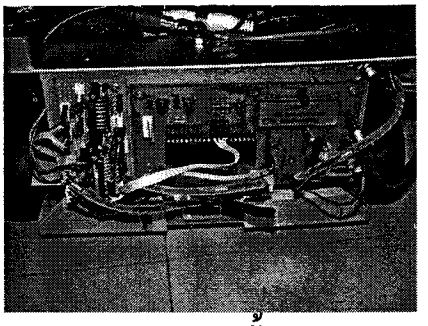


(จ) ภาพด้านหลัง

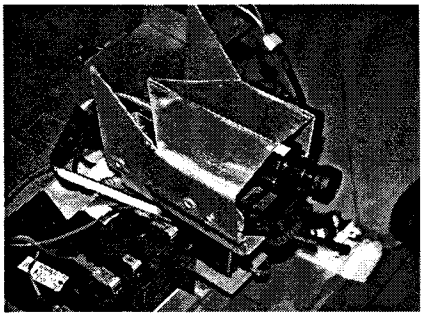


(ฉ) ภาพด้านหลังอีกมุม

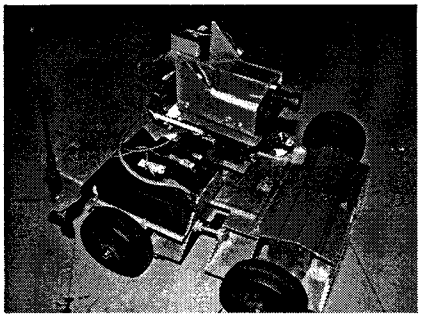
รูปที่ 4.10 ภาพถ่ายหุ่นยนต์ GUNDAM III



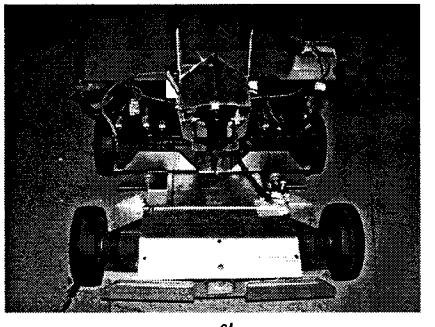
(ช) แผงวงจรทั้งหมด



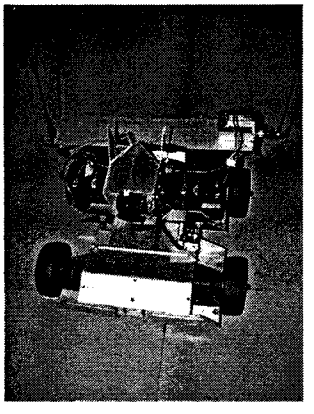
(ซ) กล้องวิดีโอขนาดเล็ก



(ญ) อีกมุมของ GUNDAM



(ฎ) ภาพด้านบน



(ฏ) หุ่นยนต์ GUNDAM

รูปที่ 4.11 ภาพถ่ายหุ่นยนต์ GUNDAM III (ต่อ)

บทที่ 5
บทสรุป

1. สรุปผลการวิจัย

ในงานวิจัยนี้ได้ทำการศึกษาออกแบบและพัฒนาหุ่นยนต์เคลื่อนที่ในที่เรียบ หุ่นยนต์ที่ได้มีคุณลักษณะสำคัญที่จำเป็นต่อการนำไปประยุกต์ใช้งานด้านต่างๆ โดยเฉพาะงานทางด้านปัญญาประดิษฐ์ด้วยงบประมาณที่ไม่สูงมาก เราสามารถพัฒนาออกแบบหุ่นยนต์ที่มีคุณลักษณะเพียงพอต่อการพัฒนางานทางด้านปัญญาประดิษฐ์ โดยมีคุณลักษณะสำคัญดังต่อไปนี้

- มีระบบประมวลผลสัญญาณภาพจากกล้องวิดีโอบนตัวหุ่นยนต์ ที่สามารถส่งข้อมูลภาพแบบไร้สายไปประมวลผลบนคอมพิวเตอร์สมรรถนะสูงได้
- มีระบบควบคุมหุ่นยนต์จากคอมพิวเตอร์ส่วนบุคคลได้ เพื่อความสะดวกในการพัฒนาโปรแกรมหรืออัลกอริทึมการประมวลผลในระดับสูงแบบต่างๆ
- มีโครงสร้างของหุ่นยนต์เคลื่อนที่ได้

ระบบการประมวลผลสัญญาณภาพเป็นสิ่งจำเป็นในการพัฒนางานทางด้านปัญญาประดิษฐ์ เนื่องมาจากมนุษย์เราใช้การมองเห็นเป็นหลักในการทำงานด้านต่างๆ ดังนั้นการมีระบบการประมวลผลสัญญาณภาพจึงถือเป็นสิ่งสำคัญสำหรับการออกแบบและพัฒนาหุ่นยนต์เคลื่อนที่ ที่ซึ่งสามารถประยุกต์ใช้งานทางด้านปัญญาประดิษฐ์ได้ อย่างไรก็ตาม ระบบการประมวลผลสัญญาณภาพเป็นระบบที่ต้องใช้พลังการคำนวณของคอมพิวเตอร์ค่อนข้างสูง รวมไปถึงการเชื่อมต่อกับอุปกรณ์จับสัญญาณภาพที่ค่อนข้างซับซ้อน ลำพังระบบไมโครคอนโทรลเลอร์เองจึงไม่สามารถที่จะรองรับในส่วนนี้ได้ การออกแบบสร้างหุ่นยนต์เคลื่อนที่ที่มีหน่วยประมวลผลที่มีสมรรถนะสูงจำเป็นต้องใช้งบประมาณที่สูง เช่นการเลือกใช้บอร์ดควบคุมชนิด single board PC หรือ Embedded PC (เช่น PC/104 หรือ EBX) ในบางกรณีมีการเลือกใช้บอร์ดควบคุมที่ดัดแปลงมาจากคอมพิวเตอร์ส่วนบุคคลที่ซึ่งถึงแม้ว่าจะมีสมรรถนะที่สูง แต่ก็มีข้อเสียในเรื่องขนาดของตัวหุ่นยนต์ที่จะต้องใหญ่ รวมไปถึงการใช้ไฟที่สูง ทำให้ขนาดของแบตเตอรี่ต้องมีขนาดใหญ่ขึ้นไปด้วย ในการพัฒนาหุ่นยนต์เคลื่อนที่ได้ซึ่งเน้นไปที่งานทางปัญญาประดิษฐ์ ประสิทธิภาพโดยรวมของระบบจะมาจากระบบการมองเห็นของหุ่นยนต์ ขนาดของหุ่นยนต์ที่ใหญ่จึงค่อนข้างจะเกินความจำเป็น (ยกเว้นกรณีที่ต้องบรรทุกหนัก เดินทางไกลหรือเดินทางในสภาพแวดล้อมที่จำเป็นต้องมีขนาดของหุ่นยนต์ที่ใหญ่) ในงานวิจัยนี้จึงได้เลือกทำการส่งภาพข้อมูลจากกล้องวิดีโอที่ติดอยู่บนตัวหุ่นยนต์ไปยังคอมพิวเตอร์ควบคุม ซึ่งทำการหน้าที่ในการประมวลผลภาพ เราจึงสามารถเลือกใช้คอมพิวเตอร์ส่วนบุคคลที่มีสมรรถนะสูง ในราคาที่ไม่แพงได้อย่างสะดวก อุปกรณ์จับภาพเข้าสู่คอมพิวเตอร์จะใช้การ์ดจับภาพซึ่งใช้ในงานกราฟิกทั่วๆ ไป (ไม่ใช่ frame grabber) ทำให้ได้ราคางบประมาณที่ค่อนข้างต่ำ โดยทำการเน้นไปที่การพัฒนา

โปรแกรมเป็นหลัก เพื่อทำให้ประสิทธิภาพโดยรวมของระบบการมองเห็นเพียงพอต่อการประยุกต์ใช้งานทางปัญญาประดิษฐ์ การพัฒนาโปรแกรมบนคอมพิวเตอร์ส่วนบุคคลจะทำในระดับภาษาสูง เช่น การใช้ Visual C++ ของบริษัทไมโครซอฟท์ หรือ Borland C++ ของบริษัทบอร์แลนด์เป็นต้น ทำให้การพัฒนาอัลกอริทึมในระดับสูงเป็นไปอย่างมีประสิทธิภาพ

ในส่วนของฮาร์ดแวร์ การออกแบบวงจรส่วนต่างๆ ของหุ่นยนต์ในงานวิจัยนี้ได้มีการพัฒนาปรับปรุงตามเทคโนโลยีปัจจุบัน โดยเน้นไปที่อุปกรณ์ที่สามารถหาได้ภายในประเทศ ในส่วนของไมโครคอนโทรลเลอร์สำหรับวงจรควบคุมการขับเคลื่อนของมอเตอร์ เราสามารถเลือกใช้วงจรทั่วๆ ไป โดยทำการทดลองใช้ชิพไมโครคอนโทรลเลอร์แบบต่างๆ (เน้นที่ตระกูล AVR ใน GUNDAM I และตระกูล PIC ใน GUNDAM III) ชิพไมโครคอนโทรลเลอร์ตระกูลต่างๆ มีทั้งข้อดีและข้อเสียแตกต่างกันไป อย่างไรก็ดี ส่วนที่เป็นปัญหาที่สุดในงานวิจัยนี้ก็คือการออกแบบให้หุ่นยนต์สามารถควบคุมแบบไร้สายจากคอมพิวเตอร์ ในขณะที่เริ่มต้นออกแบบ GUNDAM I นั้น วงจรส่งข้อมูลแบบไร้สายยังมีความยุ่งยากซับซ้อน ในงานวิจัยนี้เลือกการนำเอาวงจรบังคับวิทยุสำหรับเครื่องบินเล็กมาทำการประยุกต์ใช้ โดยทำการเพิ่ม A/D และ D/A เข้าไปเพื่อให้สามารถเชื่อมต่อกับระบบคอมพิวเตอร์และไมโครคอนโทรลเลอร์ได้ (วงจรบังคับเครื่องบินเล็กที่ใช้เป็นแบบแอนาลอกซึ่งมีราคาไม่สูงนักเมื่อเทียบกับระบบดิจิตอล) การประยุกต์ใช้ดังกล่าวมีข้อจำกัดในหลายๆ ด้าน โดยเฉพาะทางด้านจำนวนช่องการส่งข้อมูลที่จำกัด ทำให้ต้องทำการปรับปรุงวงจรส่งข้อมูลแบบไร้สายใหม่ โดยในช่วงเวลาต่อมาของการพัฒนา GUNDAM III ได้มีบริษัทผู้ผลิตชิพวงจรรับส่งข้อมูลแบบไร้สาย ด้วยมาตรฐาน RS-232 ทำให้ค่อนข้างสะดวกในการใช้งาน รวมไปถึงประสิทธิภาพในการส่งข้อมูลที่ดีขึ้นด้วย

โครงสร้างของหุ่นยนต์เคลื่อนที่นั้นมีได้หลากหลาย ขึ้นอยู่กับจุดประสงค์ในการใช้งานแบบต่างๆ ในงานวิจัยนี้ได้เลือกที่จะออกแบบให้หุ่นยนต์มีขนาดเล็กที่สุดเท่าที่จะทำได้ โดยมีโครงสร้างเชิงกลที่ปรับศูนย์การเคลื่อนที่ได้ ทำให้การเคลื่อนที่ของหุ่นยนต์มีความผิดเพี้ยนที่น้อยมาก การขับเคลื่อนล้อด้วยมอเตอร์อิสระ 4 ล้อ ทำให้การเคลื่อนที่บังคับเลี้ยวซ้าย-ขวาเป็นไปอย่างมีประสิทธิภาพ เพียงพอต่อการประยุกต์ใช้งานทางด้านปัญญาประดิษฐ์ในระดับหนึ่ง นอกไปจากนั้นแล้ว การปรับระดับความเร็วของการขับเคลื่อนได้หลายระดับ ทำให้มีความยืดหยุ่นในการประยุกต์ใช้งานได้อย่างมีประสิทธิภาพขึ้นอีกด้วย

จากข้อสรุปทั้งหมดข้างต้น ทำให้หุ่นยนต์เคลื่อนที่จากงานวิจัยนี้มีคุณลักษณะที่ตรงตามจุดประสงค์ของการออกแบบ การทดสอบการทำงานของหุ่นยนต์เป็นไปอย่างถูกต้อง การทดลองประยุกต์ใช้ระบบการมองเห็นของหุ่นยนต์เป็นไปตามวัตถุประสงค์

2. ข้อเสนอแนะ

ในการออกแบบและพัฒนางานทางด้านหุ่นยนต์ จำเป็นจะต้องใช้เทคโนโลยีหลายๆ อย่างผสมผสานกัน เพื่อให้ได้หุ่นยนต์ที่มีประสิทธิภาพและทำหน้าที่ได้ตามต้องการ เมื่อพิจารณาองค์ประกอบย่อยๆ ของหุ่นยนต์แล้ว สมควรที่จะพิจารณาเทคโนโลยีปัจจุบัน ซึ่งนอกจากจะมีประสิทธิภาพสูงขึ้นตามกาลเวลาแล้ว ยังมีราคาต้นทุนที่ถูกลงอีกด้วย ปัจจุบันอุปกรณ์ที่มีจำหน่ายภายในประเทศนั้นเพียงพอต่อการพัฒนาสร้างหุ่นยนต์ได้อย่างมีประสิทธิภาพ ดังนั้นเราจึงควรจะให้ความสำคัญในการพัฒนาสร้างหุ่นยนต์ขึ้นเอง เพื่อส่งเสริมให้เทคโนโลยีการพัฒนาหุ่นยนต์แพร่หลายภายในประเทศ สามารถลดการนำเข้าของการสั่งซื้อหุ่นยนต์และส่วนประกอบจากต่างประเทศได้ และทำให้เพิ่มทักษะในการออกแบบหุ่นยนต์เพื่อใช้งานแบบต่างๆ ได้อย่างหลากหลาย

ในส่วนของระบบการมองเห็นของคอมพิวเตอร์หรือการมองเห็นของหุ่นยนต์ ซึ่งภายในประเทศเองถือว่ามีระดับการศึกษาค้นคว้าและพัฒนาที่ค่อนข้างน้อย โดยส่วนใหญ่จะมีในระดับห้องปฏิบัติการในระดับบัณฑิตศึกษาขึ้นไป ระบบการมองเห็นของคอมพิวเตอร์นับวันจะเพิ่มความสำคัญขึ้นในหลายๆ ด้าน เนื่องมาจากเป็นระบบหลักที่มนุษย์เราใช้ในการดำเนินชีวิต สามารถใช้ในการแก้ปัญหาด้านต่างๆ ได้เป็นอย่างดี เช่นระบบหุ่นยนต์ชาญฉลาดในโรงงาน ที่ซึ่งสามารถหยิบชิ้นส่วนบนสายพานได้โดยไม่จำเป็นจะต้องจำกัดให้ชิ้นส่วนมีตำแหน่งการวางที่คงที่ หรือแม้แต่ระบบการโต้ตอบระหว่างหุ่นยนต์กับผู้ใช้ ซึ่งมีการใช้ระบบการมองเห็นของคอมพิวเตอร์เป็นหลัก ดังนั้นการพัฒนางานทางด้านการมองเห็นของคอมพิวเตอร์ หรือการมองเห็นของหุ่นยนต์จึงสมควรที่จะได้รับการส่งเสริมให้แพร่หลายมากยิ่งขึ้น

ภาคผนวก ก

ตัวอย่างต้นรหัสการเขียนและติดตั้งโปรแกรมไดร์เวอร์

1. การกำหนดฟังก์ชัน CTL_CODE

```
#define IOCTL_POWER_ON  CTL_CODE( FILE_DEVICE_UNKNOWN,      \
                                  0x801,                    \
                                  METHOD_BUFFERED,          \
                                  FILE_READ_DATA|FILE_WRITE_DATA)
```

หลังจากที่ฟังก์ชันรับรหัสคำสั่งและพารามิเตอร์อื่นๆ แล้วจะดำเนินงานตามคำสั่งแล้วส่งคำสั่งที่ Controller Box ผ่านทางพอร์ตขนาน โดยการส่งคำสั่งในแต่ละครั้งจะประกอบด้วย 4 ส่วน คือ

- Error Code (สำหรับตรวจสอบความผิดพลาด)
- Camera Angle (สำหรับควบคุมการหมุนของกล้อง)
- Speed (สำหรับควบคุมความเร็ว)
- Movement (สำหรับควบคุมการเคลื่อนที่)

แต่คำสั่งที่รับจากแอพพลิเคชันจะเป็นเพียงส่วนหนึ่งใน 4 ส่วนนี้เพราะการรับคำสั่งจะรับได้ทีละส่วนเท่านั้น ดังนั้นจึงต้องสร้างตัวแปรที่เก็บคำสั่งเก่าไว้และเปลี่ยนเฉพาะส่วนที่เปลี่ยนแปลง ดังนี้

```
struct _COMMAND {
        UCHAR ErrorCode; // Channel 1
        UCHAR CamAngle;  // Channel 2
        UCHAR Speed;     // Channel 3
        UCHAR Movement;  // Channel 4
    } GundumCmd;
```

หลังจากนั้นจึงส่งคำสั่งที่ทำการแก้ไขแล้วออกไปที่พอร์ตขนาน โดยส่งทั้ง 4 ส่วนต่อเนื่องกันไปจึงรวมเป็นฟังก์ชันเดียวกัน ดังนี้

```
VOID SendCommand(IN PDEVICE_EXTENSION pDevExt)
{
    WriteData( pDevExt, pDevExt->GundumCmd.ErrorCode );
    WriteControl( pDevExt, 11);
    WriteControl( pDevExt, 3);
```

```
        WriteData( pDevExt, pDevExt->GundumCmd.CamAngle );
        WriteControl( pDevExt, 10);
        WriteControl( pDevExt, 2);

        WriteData( pDevExt, pDevExt->GundumCmd.Speed );
        WriteControl( pDevExt, 10);
        WriteControl( pDevExt, 15);

        WriteData( pDevExt, pDevExt->GundumCmd.Movement );
        WriteControl( pDevExt, 10);
        WriteControl( pDevExt, 14);
}
```

สำหรับการเปิด-เปิด จะต้องส่งคำสั่งดังนี้

(1) การเปิด

```
// Initial port 8255
WriteData( pDevExt, 128 );
WriteControl( pDevExt, 8 );
WriteControl( pDevExt, 4 );

// power off
WriteData( pDevExt, 0 );
WriteControl( pDevExt, 9 );
WriteControl( pDevExt, 1 );

// power on
WriteData( pDevExt, 1 );
WriteControl( pDevExt, 9 );
WriteControl( pDevExt, 1 );

// turn data line on
WriteData( pDevExt, 3 );
WriteControl( pDevExt, 9 );
WriteControl( pDevExt, 1 );

pDevExt->GundumCmd.ErrorCode   =   (UCHAR)ERROR_CODE;
pDevExt->GundumCmd.CamAngle    =   (UCHAR)CAM_ANGLE_CENTER;
pDevExt->GundumCmd.Speed       =   (UCHAR)SPEED_1;
pDevExt->GundumCmd.Movement    =   (UCHAR)FORWARD;
SendCommand( pDevExt );
```

(2) การปิด

```
// power off
WriteData( pDevExt, 0 );
WriteControl( pDevExt, 9 );
WriteControl( pDevExt, 1 );
```

2. การคอมไพล์ไดรเวอร์

1) ทำการติดตั้ง Microsoft Windows 2000 DDK

2) สร้างไฟล์ที่ชื่อว่า SOURCES ในซอร์สไดเร็กทอรีและพิมพ์ข้อความดังนี้

```
TARGETNAME=Gundum
TARGETPATH=obj
TARGETTYPE=DRIVER
MSC_WARNING_LEVEL=-W3-WX

SOURCES= \
     Gundum.c \
     DevCtrl.c \
     gundumver.rc

USE_MAPSYM=1
```

หมายเหตุ การพิมพ์จะต้องพิมพ์ติดกันไปในแต่ละบรรทัดโดยไม่เว้นวรรค

3) สร้างไฟล์ชื่อว่า MAKEFILE ในซอร์สไดเร็กทอรีและพิมพ์ข้อความดังนี้

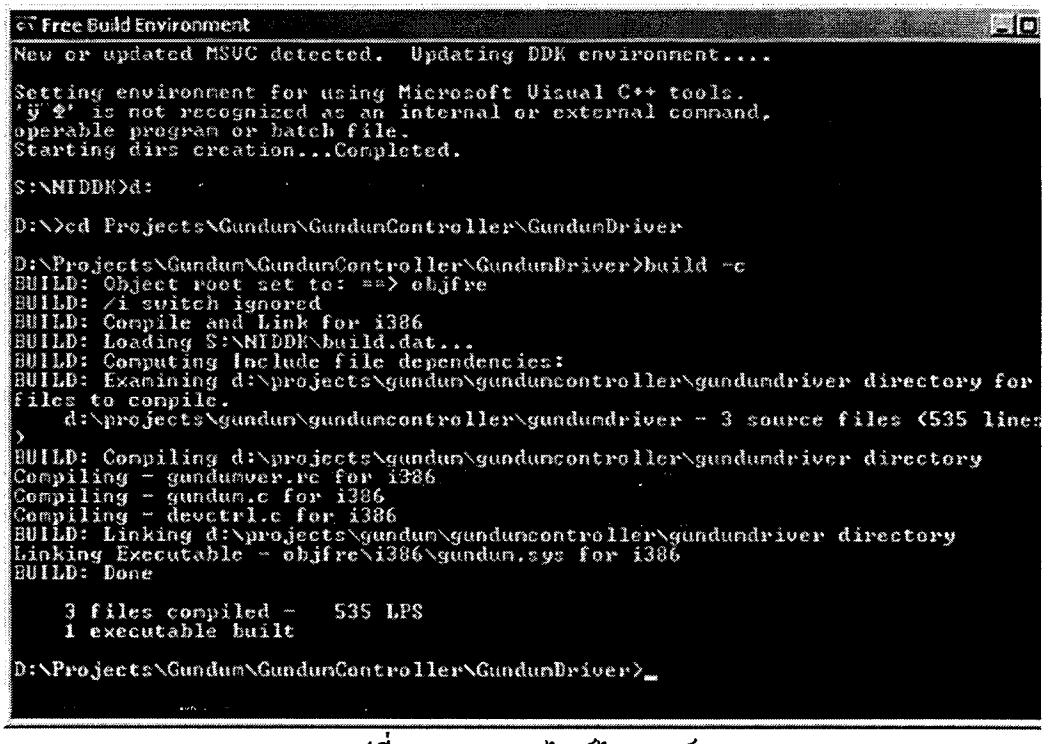
```
!INCLUDE $(NTMAKEENV)\MAKEFILE.DEF
```

4) เรียกใช้โปรแกรมคอมไพล์จาก Start Menu -> Programs -> Development Kits -> Windows 2000 DDK - > Free Build Environment

5) เปลี่ยนไดเร็กทอรีไปที่ซอร์สไดเร็กทอรี (ใช้คำสั่ง CD ที่คอมมานด์พร็อม)

6) สั่ง BUILD (พิมพ์ BUILD)

7) หลังจากสั่ง BUILD แล้วไดรเวอร์ที่คอมไพล์แล้วจะอยู่ที่ไดเร็กทอรี objfre\i386 ซึ่งจะมีนามสกุลเป็น .sys



รูปที่ ค-1 การคอมไพล์ไดรเวอร์

3. การสร้างฟังก์ชันติดต่อกับไดรเวอร์

ในการติดต่อกับไดรเวอร์จะใช้ฟังก์ชัน `CreateFile` เปิดการเชื่อมต่อเหมือนกับการเปิดพอร์ตโดยทั่วไป และใช้ฟังก์ชัน `CloseHandle` สำหรับการปิดการเชื่อมต่อ ดังนั้นเพื่อให้ง่ายต่อการใช้งานจึงสร้างฟังก์ชันสำหรับการเปิด-ปิด, ควบคุมการเคลื่อนไหวของ GUNDAM ไว้ด้วยกัน ดังนี้

```
class CGundum
{
public:
   BOOL IsPowerON();
   BOOL SetSpeed(BYTE byteSpeed);
   BOOL TurnCameraCenter();
   BOOL TurnCameraRight();
   BOOL TurnCameraLeft();
   BOOL PowerOFF();
   BOOL PowerON();
   BOOL Stop();
   BOOL Backward();
   BOOL Forward();
   BOOL TurnRight();
   BOOL TurnLeft();
   CGundum();
```

```
    virtual ~CGundum();
protected:
    HANDLE m_hDevice;
    BOOL WriteData(DWORD dwIoControlCode, BYTE byData);
BOOL ReadData();
};
```

การส่งคำสั่งควบคุมไปที่ไดรเวอร์จะใช้ฟังก์ชัน DeviceIoControl โดยสร้างเป็นฟังก์ชันเฉพาะสำหรับการส่งคำสั่ง คือ WriteData ในคลาส CGundum

4. การติดตั้งไดรเวอร์

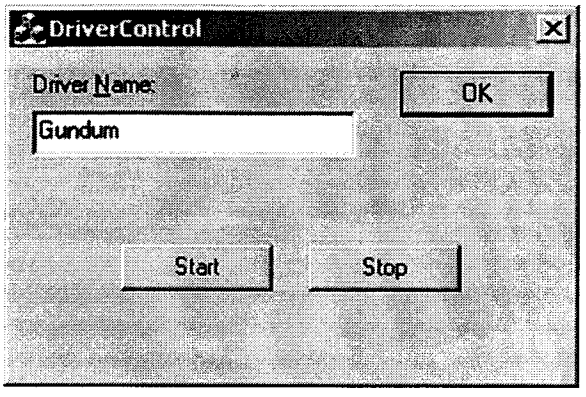
1) ทำการสร้างไฟล์รีจิสทรี (registry) ซึ่งมีนามสกุลเป็น .reg และพิมพ์โค้ดดังนี้

```
Windows Registry Editor Version 500

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Gundum]
"Type"=dword:1
"ErrorControl"=dword:1
"Start"=dword:3
"DisplayName"= "Gundum driver"
```

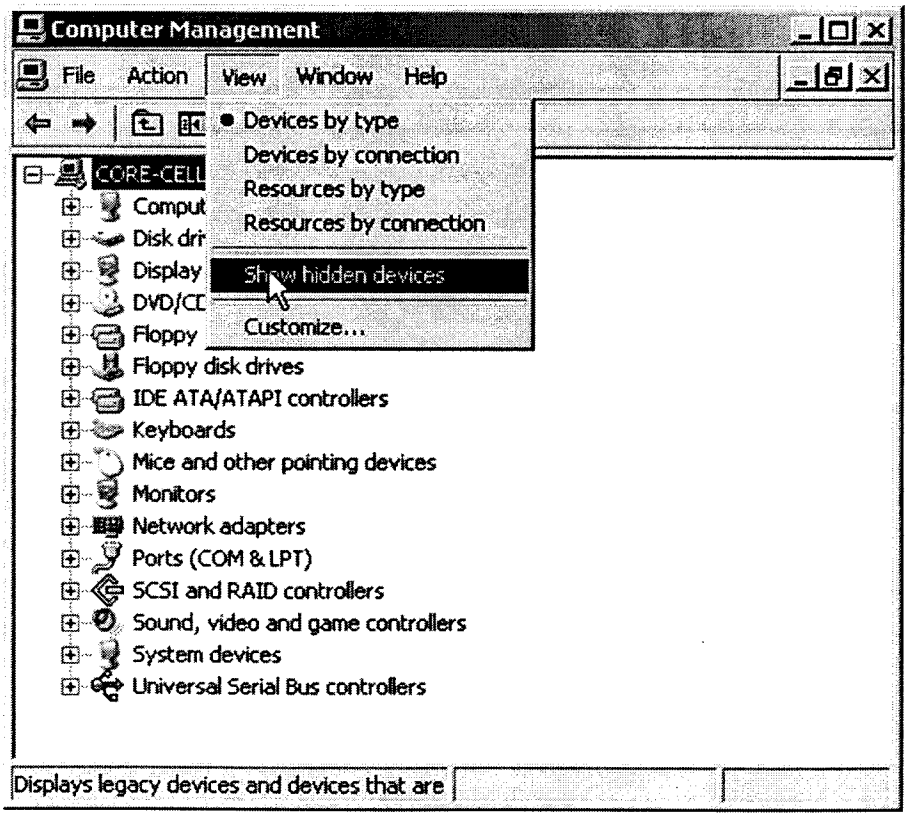
2) หลังจากสร้างไฟล์ตามขั้นตอนที่ 1 แล้วให้ดับเบิ้ลคลิกที่ไฟล์ เพื่อรวมข้อความข้างต้นเข้าไปในรีจิสทรี แล้วทำการบูตวินโดว์ใหม่

3) เรียกใช้โปรแกรม DriverControl.exe จาก CD-Romในโฟลเดอร์ Tool และเติมในช่อง Driver Name ว่า Gundum (เป็นชื่อ Symbolic Link) ดังรูปที่ 4-1 จากนั้นคลิกที่ปุ่ม START ไดรเวอร์จะถูกโหลด

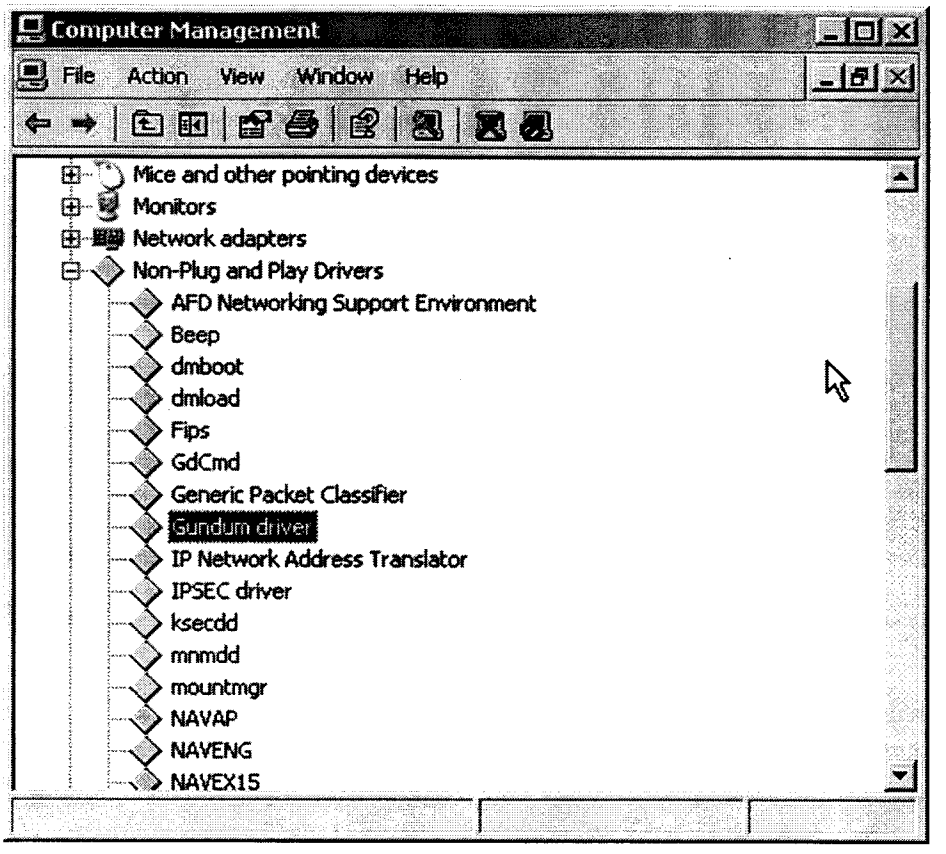


รูปที่ ค-2 การโหลดไดรเวอร์

4) เปิด Device Manager และเลือก View -> Show hidden devices ดังรูปที่ ค-3 หลังจากเลือกแล้วจะพบ GUNDAM Driver ใน Non-Plug and Play Drivers ดังรูปที่ ค-4



รูปที่ ค-3 การกำหนดให้แสดงไดรเวอร์



รูปที่ ค-4 แสดง GUNDAM driver

ประวัติผู้วิจัย

1) ชื่อ (ภาษาไทย) อาทิตย์ ศรีแก้ว

(ภาษาอังกฤษ) Arthit Srikaew

2) ตำแหน่งปัจจุบัน ผู้ช่วยศาสตราจารย์

3) หน่วยงานที่อยู่

สาขาวิศวกรรมไฟฟ้า สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี
111 ถ.มหาวิทยาลัย ต.สุรนารี
อ.เมือง จ.นครราชสีมา 30000
โทรศัพท์ (044) 22-4405
โทรสาร (044) 22-4220

4) ประวัติการศึกษา

ปีที่จบ	ระดับการศึกษา	อักษรย่อปริญญาและชื่อเต็ม	สาขาวิชา	ชื่อสถาบัน	ประเทศ
2538	ปริญญาตรี	วศ.บ./วิศวกรรมศาสตรบัณฑิต (เกียรตินิยมอันดับ 2)	วิศวกรรมอิเลคทรอนิคส์	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง	ไทย
2540	ปริญญาโท	M.S. / Master of Science	Electrical Engineering	Vanderbilt University	USA
2544	ปริญญาเอก	Ph.D. / Doctor of Philosophy	Electrical Engineering	Vanderbilt University	USA

5) สาขาวิชาการที่มีความชำนาญพิเศษ

Computer Vision, Robot Vision, Image Processing, Artificial Intelligence, Intelligent System