สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หุ่นยนต์ปฏิบัติการเสี่ยงภัยแทนมนุษย์

HUMAN UNREACHABLE OPERATION ROBOT

T055126

นายสุวิเรศ แสงประทีปทอง
นายอนุพล พินิจนรชัย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลขหมู่ .....
เลขทะเบียน 55126
วัน,เดือน,ปี - 8 มิ.ย. 2548

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

หุ่นยนต์ปฏิบัติการเสี่ยงภัยแทนมนุษย์

HUMAN UNREACHABLE OPERATION ROBOT

โดย

นายสุวิเรศ แสงประทีปทอง

นายอนุพล พินิจนรชัย

อาจารย์ที่ปรึกษา

อ.เจริญ วงษ์ชุ่มเย็น

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2546

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง หุ่นยนต์ปฏิบัติการเสี่ยงภัยแทนมนุษย์

HUMAN UNREACHABLE OPERATION ROBOT

คณะผู้จัดทำ นายสุวิเรส แสงประทีปทอง รหัส 43010506

นายอนุพล พินิจนรชัย รหัส 43010516

..... อาจารย์ที่ปรึกษา

(อ.เจริญ วงษ์ชุ่มเย็น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์ปฏิบัติการเสี่ยงภัยแทนมนุษย์

นายสุวิเรศ แสงประทีปทอง 43010506
นายอนุพล พินิจนรชัย 43010516
อ.เจริญ วงษ์ชุ่มเย็น อาจารย์ที่ปรึกษา
ปีการศึกษา 2546

บทคัดย่อ

หุ่นยนต์ปฏิบัติการเสี่ยงภัยแทนมนุษย์ (Human Unreachable Operation Robot : HUOR) ได้รับแนวคิดมาจากข้อเท็จจริงที่ว่า ในการทำงานต่าง ๆ ของมนุษย์นั้นมีข้อจำกัดหลากหลายประการที่ไม่อำนวยให้มนุษย์สามารถทำงานได้อย่างมีประสิทธิภาพตามที่ต้องการ โดยเฉพาะข้อจำกัดทางด้านสรีระร่างกายที่ทนต่อสภาวะแวดล้อมได้ในระดับหนึ่งเท่านั้น จากเหตุนี้เองที่เป็นแรงกระตุ้นในการคิดค้นและพัฒนาหุ่นยนต์ต้นแบบที่สามารถจำลองการทำงานบางอย่างของมนุษย์ได้ ซึ่งหุ่นยนต์ที่พัฒนาขึ้นนี้ได้รับการออกแบบให้สามารถเคลื่อนที่ได้ 2 ระบบ มีระบบแขนกล มีกล้องไร้สายส่งสัญญาณภาพกลับมายังผู้ควบคุม และใช้ชุดพัฒนาคอม 86 เป็นส่วนประมวลผลกลาง โดยพัฒนาโปรแกรมควบคุมการทำงานส่วนต่าง ๆ ในแบบกึ่งอัตโนมัติ (Semi-automatic Machine) ด้วยภาษาซี

จากปัจจัยพื้นฐานทางโครงสร้างและวงจรควบคุมการทำงานที่ได้รับการออกแบบขึ้นมานี้ ทำให้หุ่นยนต์ปฏิบัติการเสี่ยงภัยแทนมนุษย์มีความยืดหยุ่นในการทำงานสูง นั่นหมายความว่าผู้พัฒนารุ่นหลังสามารถนำแนวคิดนี้ไปใช้ในการพัฒนาโปรแกรมควบคุมการทำงานให้มีความซับซ้อนมากขึ้น รวมถึงการพัฒนาปัญญาประดิษฐ์หรือ AI (Artificial Intelligent) ให้หุ่นยนต์สามารถทำงานได้อย่างมีประสิทธิภาพตามที่เราต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HUMAN UNREACHABLE OPERATION ROBOT

Mr.Suwiress Sangpratheepthong

Mr.Anupon Pinijnorrachai

Mr.Charoean Vongchumyen Advisor

Academic Year 2003

ABSTRACT

Human Unreachable Operation Robot: HUOR have an idea from limit of human ability to work. Especially, limit of human body that can protect himself from just some environment. Because this reason that we decide to develop the robot that can be done those works. well, this robot is designed for 2 mobility system .there are robot arm, wireless camera sending VDO signal to the controller and using COM 86 development toolkit as CPU. Using C language to control Semi-automatic Machine.

According to infrastructure and controlling circuit of it with high flexibility. It means that next generation developer can bring this idea to develop controlling program with more complex including developing with AI to make them efficiency perform works as we want.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

"โปรเจคฮาร์ดแวร์" หลาย ๆ คนเคยเตือนว่า อย่าทำเลย เหนื่อยนะ ลำบากแย่ และอะไรต่ออิอะไรมากมาย รวมทั้งอาจารย์ที่ปรึกษาของผมเองก็เคยเตือนอยู่เช่นกัน "ทำโปรเจคฮาร์ดแวร์ต้องขยันนะ จะมาเร่งทำตอนท้ายไม่ได้หรอก..." ใช่ครับ ผมเห็นด้วยจริง ๆ เพราะเวลาที่ผ่านมา 1 ปีเต็ม ๆ เราสองคนที่มีความฝันว่าอยากจะทำหุ่นยนต์ที่สุดยอดสักตัวให้เป็นที่เชยชมต่อสาธารณชน ได้ใช้ความรู้ที่ร่ำเรียนมา 3 ปีกว่า ๆ ในรั้วลาดกระบังนี้ให้มากที่สุด จนสุดท้ายวันนี้ได้เกิดหุ่นยนต์ตัวหนึ่งที่มีนามว่า HUOR ขึ้น

แต่...กว่าจะมาถึงวันนี้นั้น ได้มีผู้ร่วมเดินเคียงข้างกันมา ช่วยเหลือพวกเรามากบ้าง น้อยบ้าง ตามแต่ศรัทธา อันจะขอกล่าวสาธยาย ณ บัดนี้

อาจารย์เจริญ วงษ์ชุ่มเย็น อาจารย์ที่ปรึกษาของเรา หากไม่ได้อาจารย์คอยชี้แนะและคอยช่วยเหลือให้คำปรึกษาอยู่เบื้องหลังล่ะก็... เราคงไม่มีวันนี้

อาจารย์นเรศ มาลัย อาจารย์ที่ปรึกษาอีกท่านหนึ่งที่คอยเป็นห่วง และคอยเตือน และกระตุ้นพวกเราเสมอมา

พ่อ (ป๋าป๊า), แม่ (มามี๊า) และทุกคนที่บ้าน สำหรับความเป็นห่วง กำลังใจ และกำลังทรัพย์ (ซึ่งใช้จ่ายไปมากมายเหลือเกิน)

พี่ดิง สำหรับสว่านแท่น เครื่องตัด ที่ให้ยืมมาใช้งานนานมาก นานจริง ๆ ครับ

จู กุ้ง น้องฮ้ง ที่มาช่วยพิมพ์รายงาน ถึงจะเพียงน้อยนิด แต่ก็แบ่งเบาไปได้มากทีเดียว

น้อง ๆ SIC ที่เป็นกำลังใจและคอยให้การช่วยเหลืออย่างดับ ๆ

เพื่อน ๆ ที่คอยหาข้าวหาน้ำมาประทังชีวี เวลาที่งานยุ่ง ๆ

ห้อง ESL เป็นที่สิงสถิต...ตลอดมา

ภาควิชา คณะ และสถาบัน ที่หล่อหลอมทั้งความรู้และประสบการณ์มาตลอดระยะเวลาที่อยู่ที่นี่

และสุดท้าย....

....คู่โปรเจค คุณนั่นแหละครับ สุดยอดมาก !

อนุพล พินิจนรชัย

สุวิเรศ แสงประทีปทอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปภาพ	IX
สารบัญตาราง	XIII
บทที่ 1 บทนำ	1
1.1 หลักการและเหตุผล	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของการพัฒนา	2
1.4 ขั้นตอนการพัฒนา	3
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	4
2.1 ชุดพัฒนาคอม 86	4
2.1.1 จุดเด่นของชุดพัฒนาคอม 86	4
2.1.2 สถาปัตยกรรมทางด้านฮาร์ดแวร์	6
2.1.2.1 ไมโครคอนโทรลเลอร์ Am186CC	6
2.1.2.2 หน่วยความจำ (Memory)	8
2.1.2.3 พอร์ตอนุกรม	10
2.1.2.4 ระบบรีเซต	11
2.1.2.5 ยูอาร์ต (UART)	11
2.1.2.6 เพาเวอร์ซัพพลาย (Power supply)	11
2.1.2.7 แอลอีดีอินดิเคเตอร์ (LED Indicator)	11
2.1.3 สถาปัตยกรรมทางด้านซอฟแวร์	12
2.2 ไมโครคอนโทรลเลอร์ MCS-51	12
2.2.1 ความแตกต่างของไมโครคอนโทรลเลอร์กับไมโครโปรเซสเซอร์	12
2.2.2 จะเลือกใช้ไมโครโปรเซสเซอร์หรือไมโครคอนโทรลเลอร์เมื่อใด	14
2.2.3 โครงสร้างและสถาปัตยกรรมของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	14
2.2.4 คุณสมบัติทางเทคนิคของไมโครคอนโทรลเลอร์ตระกูล MCS-51	15
2.2.5 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51	16
2.2.6 โครงสร้างและการทำงานของพอร์ต	18
2.2.7 การใช้งานเป็นพอร์ตอินพุต	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้าที่
2.2.8 การใช้งานเป็นพอร์ตเอาท์พุต	20
2.2.9 การอ่านค่าลอจิกจากพอร์ต	20
2.2.10 จังหวะการทำงานของไมโครคอนโทรลเลอร์ MCS-51	21
2.2.11 ไทเมอร์โหมดรีจิสเตอร์ (TMOD)	21
2.2.12 ไทเมอร์คอนโทรลรีจิสเตอร์ (TCON)	22
2.2.13 ไทเมอร์โหมดและโอเวอร์โฟลว์แฟลก	23
2.2.13.1 13-บิต ไมเมอร์โหมด (Mode 0)	23
2.2.13.2 16-บิต ไมเมอร์โหมด (Mode 1)	23
2.2.13.3 8-บิต ออโต้-รีโหมดโมหด (Mode 2)	24
2.2.13.4 สปริต ไทเมอร์โหมด (Mode 3)	24
2.2.14 คล็อกกิงซอร์ส (Clocking Source)	25
2.2.14.1 การใช้เป็นตัวจับเวลา (Timer)	25
2.2.14.2 การใช้เป็นตัวนับ (Counter)	26
2.2.14.3 การเริ่ม หยุด และการควบคุมไทเมอร์	26
2.3 การใช้งานพอร์ตสื่อสารอนุกรม	28
2.3.1 มาตรฐานพอร์ตอนุกรมแบบ RS-232	28
2.3.2 คอนเนกเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ	28
2.3.3 ยูอาร์ท (UART)	31
2.4 การรับ-ส่งข้อมูลดิจิตอลผ่านคลื่นวิทยุ	32
2.4.1 การส่งแบบทิศทางเดียว (Simplex)	32
2.4.2 การส่งแบบสองทิศทางแต่ต่างเวลากัน (Half-Duplex)	33
2.4.3 การส่งแบบสองทิศทางที่เวลาเดียวกัน (Full-Duplex)	33
2.5 การทำงานของเซอร์โวมอเตอร์	33
2.5.1 หลักการทำงานของเซอร์โวมอเตอร์	33
2.5.2 การทำงานภายในของเซอร์โวมอเตอร์	35
บทที่ 3 การออกแบบโครงสร้างหุ่นยนต์	36
3.1 ระบบการเคลื่อนที่ (Robot Movement System)	36
3.1.1 การเคลื่อนที่ในโหมดล้อทุกทิศทาง (Omni Wheel Movement Mode)	36
3.1.2 การเคลื่อนที่ในโหมดการก้าวขา (4-Legs Movement Mode)	37
3.2 ระบบแขนกล (Robot Arm System)	38
3.3 ระบบการมองเห็น (Robot Eye System)	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่ 4 การออกแบบวงจรควบคุม	หน้าที่
4.1 พีไอโอ พอร์ตอินพุท / เอาท์พุทของชุดพัฒนาคอม 86	40
4.2 ไอซาบนชุดพัฒนาคอม 86	41
4.3 การเลือกใช้งานสัญญาณอินพุทและเอาท์พุท	42
4.4 บัฟเฟอร์และแลตช์ (Buffer and Latch)	43
4.5 วงจรควบคุมการทำงาน	45
4.5.1 วงจรบอร์ดพีไอโอ	45
4.5.2 วงจรบอร์ดไอซา	46
4.5.3 วงจรรับ-ส่งคลื่นวิทยุ	47
บทที่ 5 การออกแบบส่วนโปรแกรมการทำงาน	48
5.1 โปรแกรมการทำงานบนชุดพัฒนาคอม 86	48
5.1.1 กระบวนการทำงานหลัก	48
5.1.2 กระบวนการตรวจสอบการทำงานของเซนเซอร์	48
5.1.3 กระบวนการตรวจหาระยะทาง	49
5.1.4 กระบวนการควบคุมตา	49
5.1.5 กระบวนการควบคุมมือ	49
5.1.6 กระบวนการควบคุมแขน	49
5.1.7 กระบวนการควบคุมการเคลื่อนที่	49
5.1.8 กระบวนการเปลี่ยนโมดการเคลื่อนที่	49
5.1.9 กระบวนการติดต่อสื่อสาร	49
5.2 โปรโตคอลการรับ-ส่งข้อมูล	50
5.2.1 รูปแบบกลุ่มข้อมูล	50
5.2.2 คำสั่งควบคุมจากคอมพิวเตอร์	51
5.2.3 สถานะการทำงานจากชุดพัฒนาคอม 86	53
5.3 กระบวนการควบคุมจังหวะการทำงานของขา	55
บทที่ 6 การออกแบบส่วนติดต่อผู้ใช้	58
6.1 ส่วนการติดต่อกับกล้องวงจรปิดไร้สาย	59
6.1.1 ฟังก์ชันเริ่มต้นการติดต่อกับกล้องวงจรปิด	60
6.1.2 ฟังก์ชันการบันทึกภาพนิ่ง	60
6.1.3 ฟังก์ชันการบันทึกภาพวีดีโอ	60
6.1.4 ฟังก์ชันหยุดการบันทึกภาพวีดีโอ	60
6.1.5 ฟังก์ชันยกเลิกการติดต่อกับกล้องวงจรปิด	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้าที่
6.2 ส่วนควบคุมการทำงานของหุ่นยนต์	60
6.2.1 ฟังก์ชันการเคลื่อนที่	61
6.2.2 ฟังก์ชันการเปลี่ยนโหมดการเคลื่อนที่	61
6.2.3 ฟังก์ชันการเปลี่ยนโหมดการสั่งการ	61
6.2.4 ฟังก์ชันการปรับเปลี่ยนมุมกล้อง	61
6.2.5 ฟังก์ชันการเปิด-ปิดไฟ	61
6.2.6 ฟังก์ชันการเคลื่อนไหวแขน	61
6.2.7 ฟังก์ชันการเคลื่อนไหวมือ	61
6.3 ส่วนตรวจสอบการทำงานของหุ่นยนต์	62
6.3.1 ฟังก์ชันตรวจสอบการทำงานของเซนเซอร์	62
6.3.2 ฟังก์ชันตรวจสอบทิศทางการเคลื่อนที่	62
6.4 ส่วนแสดงหน้าต่างการใช้งานของผู้ใช้	62
6.5 ส่วนติดต่อสื่อสาร	63
6.5.1 ฟังก์ชันตรวจสอบเริ่มการเชื่อมต่อ	63
6.5.2 ฟังก์ชันยกเลิกการเชื่อมต่อ	63
6.5.3 ฟังก์ชันการส่งข้อมูล	63
6.5.4 ฟังก์ชันการรับข้อมูล	63
บทที่ 7 ผลการทดสอบ	64
7.1 การทดสอบการทำงานเฉพาะส่วนโครงสร้าง	65
7.1.1 ผลทดสอบการทำงานของกล้องวงจรปิดไร้สาย	65
7.1.2 ผลทดสอบการทำงานส่วนระบบการมองเห็น	66
7.1.3 ผลทดสอบการทำงานส่วนระบบแขนกล	66
7.1.4 ผลทดสอบการทำงานส่วนขา	68
7.2 การทดสอบการทำงานวงจรควบคุม	68
7.2.1 ผลทดสอบการทำงานของวงจรขับมอเตอร์	68
7.2.2 ผลทดสอบการทำงานของบอร์ดไอซา	69
7.2.3 ผลทดสอบการทำงานของบอร์ดพีไอโอ	70
7.2.4 ผลทดสอบการทำงานของวงจรรับ-ส่งสัญญาณวิทยุ	70
บทที่ 8 บทวิจารณ์และสรุป	72
8.1 สรุปความสามารถของหุ่นยนต์ปฏิบัติการณ์เสี่ยงภัยแทนมนุษย์	72
8.2 อุปสรรค์และปัญหาที่พบ	73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้าที่
8.3 แนวทางการพัฒนาต่อ	73
ภาคผนวก ก. รายละเอียดแบบโครงสร้างหุ่น	74
ภาคผนวก ข. รายละเอียดวงจรการทำงาน	81
ภาคผนวก ค. รายละเอียดการทำงานของโปรแกรม	86
ภาคผนวก ง. วิธีการติดตั้งและใช้งาน	96
ภาคผนวก จ. คู่มือการใช้งานโปรแกรม	107
บรรณานุกรม	114

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

		หน้าที่
รูปที่ 1-1	แสดงภาพรวมการทำงานของหุ่นยนต์	3
รูปที่ 2-1	แสดงรูปแบบการพัฒนาโปรแกรมโดยใช้บอร์ดคอม 86	5
รูปที่ 2-2	แสดงการเชื่อมต่อของบอร์ดขยายกับบอร์ดคอม 86	5
รูปที่ 2-3	ส่วนประกอบต่าง ๆ ของบอร์ดคอม 86	6
รูปที่ 2-4	องค์ประกอบของไมโครคอนโทรลเลอร์AM186CC	7
รูปที่ 2-5	แสดงการแบ่งการใช้งานหน่วยความจำหลัก	9
รูปที่ 2-6	แสดงการจัดวางหน่วยความจำในส่วนของอินพุต/เอาท์พุต	9
รูปที่ 2-7	คอนเนคเตอร์ของพอร์ตอนุกรม	10
รูปที่ 2-8	โครงสร้างพื้นฐานของไมโครโปรเซสเซอร์	13
รูปที่ 2-9	โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51	15
รูปที่ 2-10	แสดงการจัดวางตำแหน่งขาของไมโครคอนโทรลเลอร์ MCS-51	16
รูปที่ 2-11	แสดงวงจรภายในของแต่ละพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	19
รูปที่ 2-12	แสดงการทำงานของไทเมอร์ในโหมดต่าง ๆ	25
รูปที่ 2-13	แสดงความถี่ของสัญญาณนาฬิกาที่จะเข้าไทเมอร์	26
รูปที่ 2-14	แสดงการใช้บิตควบคุม TR	27
รูปที่ 2-15	แสดงระบบทั้งหมดของไทเมอร์ 1	27
รูปที่ 2-16	แสดงการจัดขาของคอนเนคเตอร์พอร์ตอนุกรมตามมาตรฐาน RS-232	29
รูปที่ 2-17	แสดงการต่อกับอุปกรณ์ภายนอกในลักษณะต่าง ๆ	30
รูปที่ 2-18	แสดงองค์ประกอบระบบสื่อสารพื้นฐาน	32
รูปที่ 2-19	แสดงการส่งข้อมูลแบบ Simplex	32
รูปที่ 2-20	แสดงการส่งข้อมูลแบบ Half – duplex	33
รูปที่ 2-21	แสดงการส่งข้อมูลแบบ Full – duplex	33
รูปที่ 2-22	แสดงการตอบสนองของเซอร์โวมอเตอร์เมื่อจ่ายพัลส์ขนาด 1.4 ms	34
รูปที่ 2-23	แสดงการตอบสนองของเซอร์โวมอเตอร์เมื่อจ่ายพัลส์ขนาด 1 ms	34
รูปที่ 2-24	แสดงการตอบสนองของเซอร์โวมอเตอร์เมื่อจ่ายพัลส์ขนาด 2 ms	34
รูปที่ 2-25	แสดงภาคการทำงานของเซอร์โวมอเตอร์	35
รูปที่ 3-1	แสดงการเคลื่อนที่ในโหมดล้อทุกทิศทาง	36
รูปที่ 3-2	ล้อแบบ Omni	36
รูปที่ 3-3	แสดงเคลื่อนที่ในโหมดการก้าวขา	37
รูปที่ 3-4	แสดงลักษณะของขาในโหมดต่าง ๆ	37
รูปที่ 3-5	แสดงระบบแขนกล	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

หน้าที่

รูปที่ 3-6 แสดงลักษณะของมือสว่าน 38
รูปที่ 3-7 แสดงลักษณะของมือจับ 2 ง่าม 38
รูปที่ 3-8 แสดงระบบการมองเห็น 39
รูปที่ 3-9 แสดงมุมมองต่าง ๆ ของกล้อง 39
รูปที่ 4-1 แสดงคอนเนคเตอร์ของ PIO บนชุดพัฒนาคอม 86 40
รูปที่ 4-2 แสดงคอนเนคเตอร์ของ ISA บนชุดพัฒนาคอม 86 41
รูปที่ 4-3 แสดงบล็อกไดอะแกรมของวงจรบอร์ดพีไอโอ 45
รูปที่ 4-4 แสดงบล็อกไดอะแกรมของวงจรบอร์ดไอซา 46
รูปที่ 4-5 แสดงฟังก์ชั่นการทำงานของ TX-2 / RX-2 47
รูปที่ 4-6 แสดงการรับ-ส่งข้อมูล ผ่านวงจร RF 47
รูปที่ 5-1 แสดงกระบวนการการทำงานหลักของหุ่นยนต์ 48
รูปที่ 5-2 แสดงรูปแบบสถานการณ์ก้าวเดินในโหมดการเดินด้วยขา 56
รูปที่ 6-1 แสดงหน้าหลักของโปรแกรมส่วนติดต่อผู้ใช้ 58
รูปที่ 6-2 แสดง Use case การทำงานของส่วนติดต่อผู้ใช้ 59
รูปที่ 6-3 แสดงเมนูการเรียกใช้ฟังก์ชันส่วนแสดงภาพ 59
รูปที่ 6-4 แสดงปุ่มควบคุมการทำงานของหุ่นยนต์ 60
รูปที่ 6-5 แสดงการแสดงผลสถานะของหุ่นยนต์ 62
รูปที่ 6-6 แสดงหน้าต่างการติดต่อสื่อสาร 63
รูปที่ 7-1 แสดงภาพถ่ายโครงสร้างและส่วนประกอบของ HUOR 64
รูปที่ 7-2 แสดงภาพถ่ายระบบการมองเห็น 66
รูปที่ 7-3 แสดงภาพถ่ายระบบแขนกล 67
รูปที่ 7-4 แสดงภาพถ่ายลักษณะของมือทั้ง 2 ข้าง 67
รูปที่ 7-5 แสดงภาพถ่ายลักษณะของขา 68
รูปที่ 7-6 แสดงภาพถ่ายวงจรขับมอเตอร์ 69
รูปที่ 7-7 แสดงภาพถ่ายบอร์ดไอซาที่ทำการประกอบกับคอม86 69
รูปที่ 7-8 แสดงภาพถ่ายบอร์ดพีไอโอที่ทำการประกอบกับคอม86 70
รูปที่ 7-9 แสดงภาพถ่ายชุดวงจรรับ-ส่งข้อมูลผ่านคลื่นวิทยุ 71
รูปที่ ก-1 แสดงแบบโครงสร้างหุ่นยนต์ 3 มิติ 74
รูปที่ ก-2 แสดงแบบลำตัวหุ่นยนต์ 3 มิติ 75
รูปที่ ก-3 แสดงแบบโครงร่างลำตัวหุ่นยนต์ 75
รูปที่ ก-4 แสดงแบ้บยึดกล้องวงจรปิด 3 มิติ 76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

หน้าที่

รูปที่ ก-5 แสดงแบบโครงร่างยึดกล้องวงจรปิด 76
รูปที่ ก-6 แสดงแบบแขนกล 3 มิติ 77
รูปที่ ก-7 แสดงแบบโครงร่างแขนกล 77
รูปที่ ก-8 แสดงแบบมือจับและมือสว่าน 3 มิติ 78
รูปที่ ก-9 แสดงแบบโครงร่างมือจับและมือสว่าน 78
รูปที่ ก-10 แสดงแบบโครงสร้างขา 3 มิติ 79
รูปที่ ก-11 แสดงแบบโครงร่างโครงสร้างขา 79
รูปที่ ก-12 แสดงแบบล้อ Omni 3 มิติ 80
รูปที่ ก-13 แสดงแบบโครงร่างล้อ Omni 80
รูปที่ ข-1 แสดงขาไอซี L298N 81
รูปที่ ข-2 แสดงลายวงจร Top Layer ของวงจรขับมอเตอร์ 82
รูปที่ ข-3 แสดงลายวงจร Bottom Layer ของวงจรขับมอเตอร์ 82
รูปที่ ข-4 แสดงขาการใช้งาน AT89C2051 83
รูปที่ ข-5 แสดงลายวงจรแปลงข้อมูลระหว่างแบบขนานและแบบอนุกรม 83
รูปที่ ข-6 แสดงลายวงจรอินฟราเรดเซนเซอร์แบบเดี่ยว 84
รูปที่ ข-7 แสดงลายวงจรอินฟราเรดเซนเซอร์ตรวจสอบเส้นทางการเดิน 84
รูปที่ ข-8 แสดงลายวงจร Top Layer ของวงจรบอร์ดไอซีและบอร์ดพีไอโอ 85
รูปที่ ข-9 แสดงลายวงจร Bottom Layer ของวงจรบอร์ดไอซีและบอร์ดพีไอโอ 85
รูปที่ ค-1 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการทำงานหลัก 86
รูปที่ ค-2 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการส่งข้อมูล 87
รูปที่ ค-3 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการรับข้อมูล 88
รูปที่ ค-4 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการการเคลื่อนที่ 89
รูปที่ ค-5 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการเปลี่ยนโหมดการเคลื่อนที่ 90
รูปที่ ค-6 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการควบคุมตา 91
รูปที่ ค-7 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการควบคุมแขน 92
รูปที่ ค-8 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการควบคุมมือ 93
รูปที่ ค-9 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการตรวจสอบการทำงานของเซนเซอร์ 94
รูปที่ ค-10 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการตรวจหาระยะทาง 95
รูปที่ ง-1 แสดงการเชื่อมต่อสาย serial console เข้ากับบอร์ดคอม86 96
รูปที่ ง-2 แสดงการตั้งค่าต่างๆ ของโปรแกรมเทอร์มินัล 97
รูปที่ ง-3 แสดงการเชื่อมต่ออแดปเตอร์กับบอร์ดคอม 86 97

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

		หน้าที่
รูปที่ ง-4	แสดงตัวอย่างการแสดงผลของบอร์ดคอม 86 เมื่อบอร์ดเริ่มทำงาน	98
รูปที่ ง-5	แสดงตัวอย่างการส่ง file image ไปโปรแกรมลงบนบอร์ดคอม 86	99
รูปที่ ง-6	แสดงตัวอย่างการส่ง file จาก PC	101
รูปที่ ง-7	แสดงตัวอย่างการรับ file จาก com86	102
รูปที่ ง-8	แสดงการสร้างโปรเจคใหม่ใน Borland C++	103
รูปที่ ง-9	แสดงการตั้งค่าให้โปรเจคใน Borland C++	103
รูปที่ ง-10	แสดงวิธีการคอมไพล์ใน Borland C++	104
รูปที่ ง-11	แสดงวิธีการสร้างไฟล์ Execute ใน Borland C++	104
รูปที่ ง-12	แสดงวิธีการเพิ่ม Component ใน Visual Basic 6.0	105
รูปที่ ง-13	แสดงการเลือกใช้งาน Component VideoOCX ใน Visual Basic 6.0	105
รูปที่ ง-14	แสดงเครื่องมือ VideoOCX ใน Visual Basic 6.0	106
รูปที่ ง-15	แสดงเฮดเดอร์ไฟล์ของโปรแกรม Keil 8051	106
รูปที่ จ-1	แสดงองค์ประกอบรวมของโปรแกรมควบคุม	107
รูปที่ จ-2	แสดงรายละเอียดของเมนูการใช้งานฟังก์ชันของกล้องวงจรปิด	108
รูปที่ จ-3	แสดงหน้าต่าง Capture Image ที่ปรากฏขึ้น	108
รูปที่ จ-4	แสดงภาพตัวอย่างที่เกิดจากการบันทึกภาพโดยโปรแกรม	109
รูปที่ จ-5	แสดงหน้าต่างทำการบันทึกไฟล์วีดีโอ	109
รูปที่ จ-6	แสดงรายละเอียดส่วนแสดงสถานการณ์ทำงานของหุ่นยนต์	110
รูปที่ จ-7	แสดงสถานะทิศทางการเคลื่อนที่	110
รูปที่ จ-8	แสดงการเปลี่ยนแปลงพอร์ตสื่อสาร	111
รูปที่ จ-9	แสดงการเปลี่ยนมุมกล้องด้วยเมาส์	112
รูปที่ จ-10	แสดงการปรับเปลี่ยนมุมมองของผู้ใช้	113
รูปที่ จ-11	แสดงหน้าจอช่วยเหลือการใช้งานคีย์บอร์ด	113

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้าที่
ตารางที่ 2-1 คุณสมบัติของไมโครคอนโทรลเลอร์	13
ตารางที่ 2-2 หน้าที่พิเศษของพอร์ต 1 ในตัวไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	18
ตารางที่ 2-3 รีจิสเตอร์ TMOD (Timer Mode)	22
ตารางที่ 2-4 การใช้ Timer โหมดต่างๆ	22
ตารางที่ 2-5 แสดงความหมายแต่ละบิตของรีจิสเตอร์ TCON (Timer Control)	23
ตารางที่ 4-1 แสดงอุปกรณ์ที่ต้องทำการควบคุม	42
ตารางที่ 4-2 แสดงจำนวนบิตที่ใช้ในแต่ละพอร์ต	43
ตารางที่ 4-3 แสดงการใช้งานไมโครคอนโทรลเลอร์ MCS-51	44
ตารางที่ 5-1 แสดงรูปแบบกลุ่มคำสั่งของข้อมูล	50
ตารางที่ 5-2 แสดงรูปแบบคำสั่งอินาเบิลการทำงานส่วนต่าง ๆ	51
ตารางที่ 5-3 แสดงรูปแบบข้อมูลสั่งการระบบการเคลื่อนที่และตา	51
ตารางที่ 5-4 แสดงรูปแบบข้อมูลสั่งการระบบแขนและมือข้างซ้าย	52
ตารางที่ 5-5 แสดงรูปแบบข้อมูลสั่งการระบบแขนและมือข้างขวา	52
ตารางที่ 5-6 แสดงรูปแบบข้อมูลสถานะของระบบการเคลื่อนที่และทิศทาง	53
ตารางที่ 5-7 แสดงรูปแบบข้อมูลสถานะของอินฟราเรดเซนเซอร์และอุลตร้าโซนิคเซนเซอร์	54
ตารางที่ 5-8 แสดงรูปแบบข้อมูลทิศทาง (Data Direction)	55
ตารางที่ 7-1 แสดงรายการอุปกรณ์ที่ใช้	65
ตารางที่ จ-1 แสดงปุ่มคีย์บอร์ดที่ใช้ควบคุมหุ่นยนต์	112

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 หลักการและเหตุผล

หุ่นยนต์ (Robot) คือเครื่องจักรกลชนิดหนึ่ง ที่มีลักษณะการทำงานแบบอัตโนมัติ (Automatics Machine) หรือกึ่งอัตโนมัติ (Semi automatics Machine) และสามารถโปรแกรมให้ทำงานอย่างใดอย่างหนึ่ง หรือหลายอย่างได้ ในปัจจุบันนั้น หุ่นยนต์ได้เข้ามามีบทบาทในวิถีชีวิตของมนุษย์มากขึ้น โดยเฉพาะงานทางด้านอุตสาหกรรมการผลิตต่าง ๆ อาทิเช่น อุตสาหกรรมประกอบชิ้นส่วนรถยนต์ อุตสาหกรรมพ่นสี เป็นต้น และในทางการแพทย์ก็ได้มีการใช้หุ่นยนต์ช่วยในการรักษาคนไข้ เป็นพยาบาลคอยดูแลคนไข้ รวมถึงการผ่าตัดอีกด้วย ซึ่งได้แสดงให้เห็นถึงว่าเทคโนโลยีการผลิตหุ่นยนต์ในปัจจุบันนั้นมีความก้าวหน้าไปมาก ในการที่จะพัฒนาและสร้างสรรค์หุ่นยนต์ที่สามารถนำมาแบ่งเบาภาระการทำงานของมนุษย์ลงได้ แต่ทั้งนี้ก็ยังมีหุ่นยนต์อีกประเภทหนึ่ง นั่นก็คือหุ่นยนต์นักสำรวจและปฏิบัติการแทนมนุษย์ หรือ "หุ่นยนต์ภาคสนาม (Field Robot)" ที่ได้รับความสนใจและเป็นสิ่งท้าทายนักคิดนักวิจัยหลากหลายสาขาทั่วโลก ในการที่จะพัฒนาหุ่นยนต์ประเภทนี้ให้ตอบสนองต่อความต้องการของมนุษย์ให้มากที่สุด เนื่องจากว่าในการทำงานภาคสนามนั้นมีความสลับซับซ้อนและมีปัญหาที่ไม่คาดคิดเกิดขึ้นมากมาย ทำให้ในบางครั้งนั้นการพัฒนาโปรแกรมก็มีข้อจำกัดที่มากขึ้นด้วย

ในการทำงานต่าง ๆ ของมนุษย์นั้น มนุษย์ต้องพบเจอกับสภาพแวดล้อมที่แตกต่างกัน บางสภาวะแวดล้อมก็มีความเหมาะสมในการปฏิบัติงานของมนุษย์ แต่ก็มีบางสภาวะที่ไม่เอื้อให้การปฏิบัติงานของมนุษย์ดำเนินไปได้ เนื่องจากว่ามนุษย์มีขอบเขตและปัจจัยหลาย ๆ สิ่ง โดยเฉพาะในเรื่องของสรีระร่างกายที่ทำให้มนุษย์ถูกจำกัดไว้ด้วยสภาพแวดล้อมที่เหมาะสมเท่านั้น ด้วยเหตุนี้เราจึงต้องเร่งคิดค้นและพัฒนาหุ่นยนต์ที่สามารถเข้าไปปฏิบัติงานแทนมนุษย์ในสภาพแวดล้อมต่าง ๆ ได้ เพื่อให้การทำงานของมนุษย์เกิดประสิทธิภาพสูงสุด เนื่องจากว่าหุ่นยนต์ที่มนุษย์ประดิษฐ์ขึ้นนั้น เราสามารถเลือกวัตถุดิบ (Material) ที่จะนำมาเป็นองค์ประกอบของชิ้นส่วนต่าง ๆ ได้ จึงทำให้หุ่นยนต์ของเรามีความแข็งแกร่งและทนต่อสภาพแวดล้อมอันเลวร้ายได้ และ ปัญญาประดิษฐ์ หรือ AI (Artificial Intelligent) ก็จะมีบทบาทสำคัญในการพัฒนาหุ่นยนต์ให้สามารถปฏิบัติงานได้ดังที่มนุษย์ต้องการ สุดท้ายแล้ว มนุษย์เราก็สามารถปฏิบัติงานได้ในทุกสภาวะแวดล้อม ซึ่งจะส่งผลต่อความก้าวหน้าในการกู้ภัยช่วยเหลือเพื่อนมนุษย์ รวมถึงการสำรวจและการพัฒนางานวิจัย ต่าง ๆ อีกด้วย

1.2 วัตถุประสงค์

1. เพื่อสร้างหุ่นยนต์ต้นแบบที่สามารถทำงานบางอย่างคล้ายคลึงกับมนุษย์ได้
2. เพื่อพัฒนาหุ่นยนต์ที่มนุษย์สามารถควบคุมจากระยะไกลได้ในระดับหนึ่ง
3. เพื่อพัฒนาโปรแกรมเพื่อควบคุมหุ่นยนต์ให้สามารถทำงานในแบบกึ่งอัตโนมัติได้
4. จุดประกายและเป็นตัวอย่างแนวความคิดให้กับนักเรียน นักศึกษาไทยในการออกแบบและพัฒนาหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของการพัฒนา

ในการพัฒนาหุ่นยนต์ปฏิบัติการเสี่ยงภัยแทนมนุษย์ (**H**uman **U**nreachable **O**peration **R**obot) หรือ HUOR จะครอบคลุมตั้งแต่การออกแบบและสร้างตัวหุ่นยนต์จนถึงการเขียนโปรแกรมควบคุมการทำงาน โดยมีชุดพัฒนาคอม 86 เป็นส่วนประมวลผลกลาง ซึ่งใช้ภาษาซีหรือซีพลัสพลัสในการพัฒนาโปรแกรม นอกจากนี้ผู้ควบคุมและหุ่นยนต์จะต้องสามารถรับ-ส่งข้อมูลไร้สายผ่านคลื่นวิทยุได้ ด้วยโปรโตคอลที่เหมาะสม และสุดท้ายคือการพัฒนาส่วนติดต่อผู้ใช้ (User Interface) ที่น่าสนใจและง่ายต่อการใช้งาน

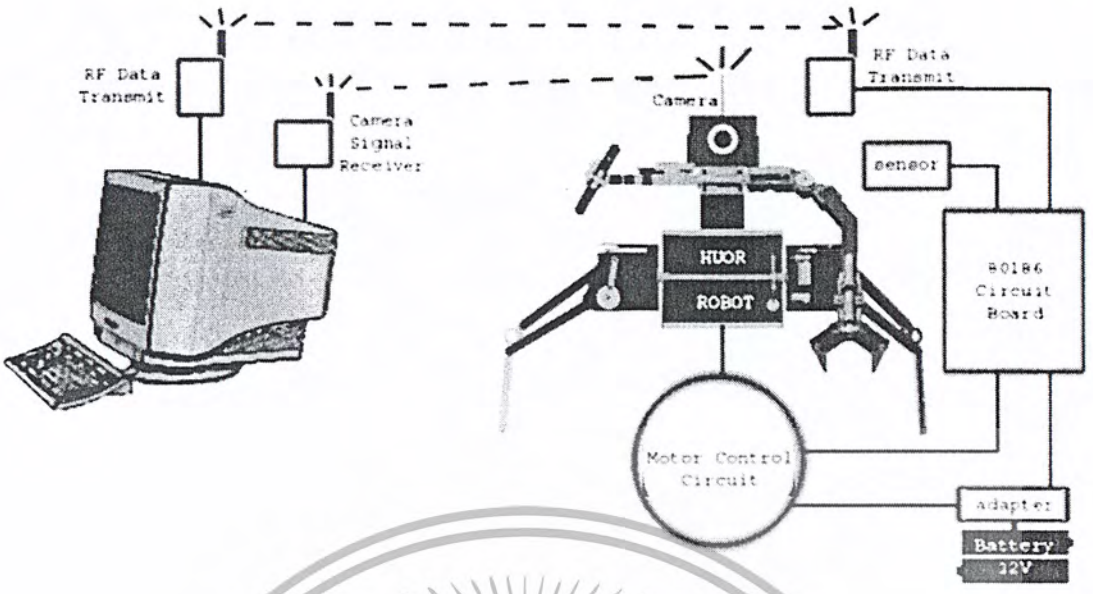
ผู้พัฒนาได้ทำการแบ่งขอบเขตในการพัฒนาออกเป็น 3 ส่วนที่สำคัญ ดังนี้

1. ลักษณะเชิงโครงสร้างของหุ่นยนต์
   - สามารถเคลื่อนที่ได้ใน 2 ลักษณะคือ การเคลื่อนที่ด้วยล้อ และการเคลื่อนที่ด้วยขา 4 ขา
   - มีแขน 2 ข้างที่สามารถเคลื่อนที่ได้ 2 แกน และมีมือที่สามารถทำงานในลักษณะต่าง ๆ กันได้
   - มีกล้องไร้สายเปรียบเสมือนตาเพื่อส่งสัญญาณภาพกลับมายังผู้ควบคุม สามารถหมุนปรับมุมกล้องได้

2. โปรแกรมการทำงานของหุ่นยนต์
   - สามารถทำงานตามที่ผู้ควบคุมสั่งการได้ในทุกฟังก์ชัน
   - มีการทำงานในแบบกึ่งอัตโนมัติ อาทิเช่น ในการเคลื่อนที่ด้วยระบบขา ผู้ควบคุมทำการส่งเพียงแค่ทิศทางเท่านั้น ส่วนจังหวะการก้าวขาของหุ่นยนต์ จะเป็นไปโดยอัตโนมัติ
   - มีการตรวจสอบสิ่งแวดล้อมด้วยเซนเซอร์ (Sensor) เพื่อช่วยเหลือการทำงานในส่วนต่าง ๆ ได้ เช่น มีเซนเซอร์ติดไว้ที่มือ เพื่อตรวจสอบการหยิบจับสิ่งของ เป็นต้น
   - สามารถติดต่อกับผู้ควบคุมเพื่อส่งสถานะการทำงานปัจจุบันและแจ้งความผิดปกติที่เกิดขึ้นได้
   - มีโปรโตคอลที่เหมาะสมในการติดต่อระหว่างผู้ควบคุมและหุ่นยนต์

3. โปรแกรมส่วนติดต่อกับผู้ใช้
   - มีปุ่มควบคุมการทำงานของหุ่นยนต์ทั้งหมด โดยออกแบบให้ง่ายต่อการควบคุม
   - มีการรับสัญญาณภาพจากกล้องมาแสดงผล และสามารถบันทึกภาพนิ่งไว้ได้ (Capture Image)
   - มีการแสดงสถานะการทำงานของหุ่นยนต์ในปัจจุบันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1-1 แสดงภาพรวมการทำงานของหุ่นยนต์

1.4 ขั้นตอนการพัฒนา

1. ศึกษาและวิเคราะห์ความเป็นไปได้ในภาพรวม
2. ออกแบบโครงสร้างของหุ่นยนต์และเลือกหาวัสดุที่เหมาะสม
3. ศึกษาและออกแบบวงจรการทำงานส่วนต่าง ๆ
4. ศึกษาและทดสอบการทำงานของชุดพัฒนาคอม 86
5. ศึกษาและทดลองการรับ-ส่งข้อมูลไร้สายด้วยคลื่นวิทยุ
6. ศึกษาการนำสัญญาณภาพจากกล้องไร้สายมาใช้งาน
7. ศึกษาและออกแบบโครงสร้างการเขียนโปรแกรมของระบบ
8. ประกอบโครงสร้างหุ่นยนต์และวงจรการทำงานต่าง ๆ
9. พัฒนาโปรแกรมการทำงานบนตัวหุ่นยนต์
10. พัฒนาโปรโตคอลการรับ-ส่งข้อมูลไร้สาย
11. ทดสอบการทำงานของหุ่นยนต์ร่วมกับโปรโตคอลรับ-ส่งข้อมูล
12. พัฒนาโปรแกรมส่วนติดต่อผู้ใช้
13. ทดสอบการทำงานและแก้ไขข้อผิดพลาด
14. ประเมินผลการทำงานและจัดทำเอกสารรายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 ชุดพัฒนาคอม 86

ชุดพัฒนาคอม 86 (COM86) เป็นชุดเครื่องมือสำหรับใช้ในการพัฒนาอุปกรณ์ทางด้านระบบฝังตัวที่อยู่บนพื้นฐานการพัฒนาโดยการใช้แพลตฟอร์มที่มีไมโครโพรเซสเซอร์ตระกูล x86 ซึ่งเป็นที่คุ้นเคยกับผู้ใช้งานในปัจจุบันเป็นส่วนประกอบหลักในการทำงาน

2.1.1 จุดเด่นของชุดพัฒนาคอม 86

- ใช้ไมโครคอนโทรลเลอร์ที่มีความสามารถสูง

ภายในชุดพัฒนาคอม 86 มีบอร์ดคอม 86 ซึ่งได้เลือกใช้ไมโครคอนโทรเลอร์ Am186CC ของบริษัท AMD ซึ่งเป็นไมโครคอนโทรเลอร์ขนาด 16 บิต ทำงานอยู่ที่ความเร็ว 24 MHz ทำให้มีความสามารถในการประมวลผลต่างๆเพิ่มมากขึ้นเมื่อเทียบกับการพัฒนาโดยใช้ไมโครคอนโทรเลอร์ขนาด 8 บิต ที่นิยมใช้กันอยู่ในปัจจุบัน นอกจากนี้ภายในตัวไมโครคอนโทรเลอร์เองได้รวมเอาอุปกรณ์พื้นฐานต่าง ๆ ที่จำเป็นในการใช้งาน เช่น DRAM controller, Interrupt controller, ยูอาร์ท เข้าไว้ภายในตัวไมโครคอนโทรเลอร์ซึ่งทำให้ง่ายต่อการออกแบบอุปกรณ์เพื่อทำเป็นสินค้าภายหลังการพัฒนาตัวอย่าง เช่น การนำบอร์ดคอม 86 ไปพัฒนาเป็นอุปกรณ์อย่างหนึ่งที่ต้องการควบคุมการทำงานผ่านทาง USB พอร์ต จะสามารถทำได้โดยง่าย เนื่องจากไม่ต้องการออกแบบวงจรควบคุมการทำงานในส่วน USB นี้เพิ่มอีก เพียงแต่ออกแบบวงจรการทำงานส่วนอื่นที่ต้องการเพิ่มเติม ทำให้ทุ่นเวลาในการพัฒนาเป็นต้น นอกจากนี้ ภายในไมโครคอนโทรเลอร์ยังเพิ่มส่วนโมดูลทางด้านการสื่อสารต่างๆเข้าไปในตัว ทำให้เป็นไมโครคอนโทรเลอร์ที่มีความสารถเด่นทางด้านการสื่อสาร เหมาะกับการพัฒนาอุปกรณ์ที่ต้องใช้คุณสมบัติการสื่อสารต่างๆซึ่งจะสามารถทำได้ง่าย เมื่อเทียบกับการใช้ไมโครคอนโทรเลอร์ขนาดเล็กแบบเดิมซึ่งมีข้อจำกัดต่าง ๆ อยู่มากมาย

- มีเครื่องมือในการพัฒนาโปรแกรมมาก

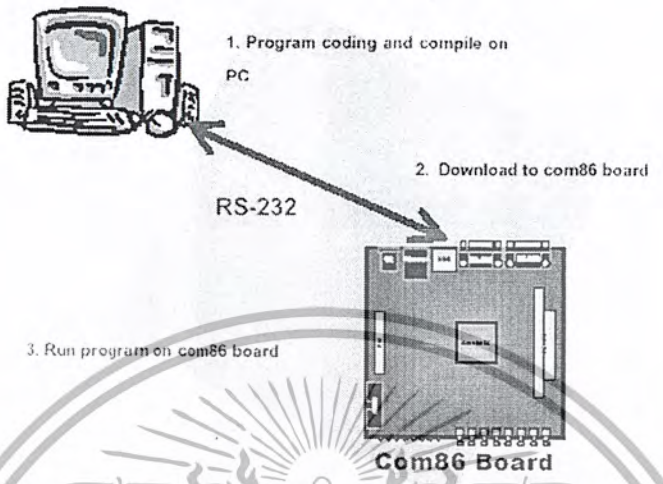
เนื่องจากการพัฒนาโปรแกรม โดยใช้ชุดพัฒนาคอม 86 นั้น เป็นเสมือนกับการพัฒนาโปรแกรมบนของคอมพิวเตอร์ส่วนบุคคลทั่วไป ดังนั้นจึงสามารถนำโปรแกรมเครื่องมือต่าง ๆ ที่ใช้พัฒนาโปรแกรมสำหรับคอมพิวเตอร์ส่วนบุคคลมาใช้งานได้มากมาย และโปรแกรมเครื่องมือเหล่านี้ก็เป็นที่คุ้นเคยของนักพัฒนาจึงทำให้สามารถพัฒนาอุปกรณ์ได้อย่างรวดเร็ว ลดต้นทุนในการพัฒนาต่างๆได้เป็นจำนวนมาก นอกจากนี้ภายในชุดพัฒนายังมีชุดซอฟต์แวร์เครื่องมือต่าง ๆ ที่จำเป็นสำหรับช่วยการพัฒนาโปรแกรมด้วยอีกทางหนึ่ง

- ง่ายในการเรียนรู้และพัฒนา

บอร์ดคอม 86 ภายในชุดพัฒนาคอม 86 นี้ มีสถาปัตยกรรมของไมโครคอนโทรเลอร์แบบ x86 ซึ่งเป็นสถาปัตยกรรมเดียวกับคอมพิวเตอร์ส่วนบุคคลทั่วไป โดยมีชุดคำสั่งที่เข้ากันได้กับชุดคำสั่งของ 80286

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

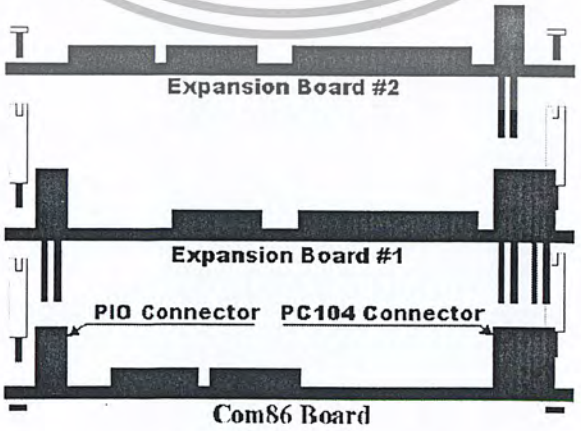
(Real mode) ทำให้ง่ายต่อการเรียนรู้ในการพัฒนาโปรแกรมต่าง ๆ และยังสามารถพัฒนา และทดสอบซอฟแวร์ต่าง ๆได้บนคอมพิวเตอร์ ก่อนที่จะทดสอบบอร์ดคอม 86 ทำให้สามารถพัฒนาอุปกรณ์ทางด้านระบบฝังตัวต่าง ๆ ได้อย่างรวดเร็ว ซึ่งรูปแบบการพัฒนามีลักษณะเป็นไปดังรูป



รูปที่ 2-1 แสดงรูปแบบการพัฒนาโปรแกรมโดยใช้บอร์ดคอม 86

- มีความสามารถในการเพิ่มขยายได้

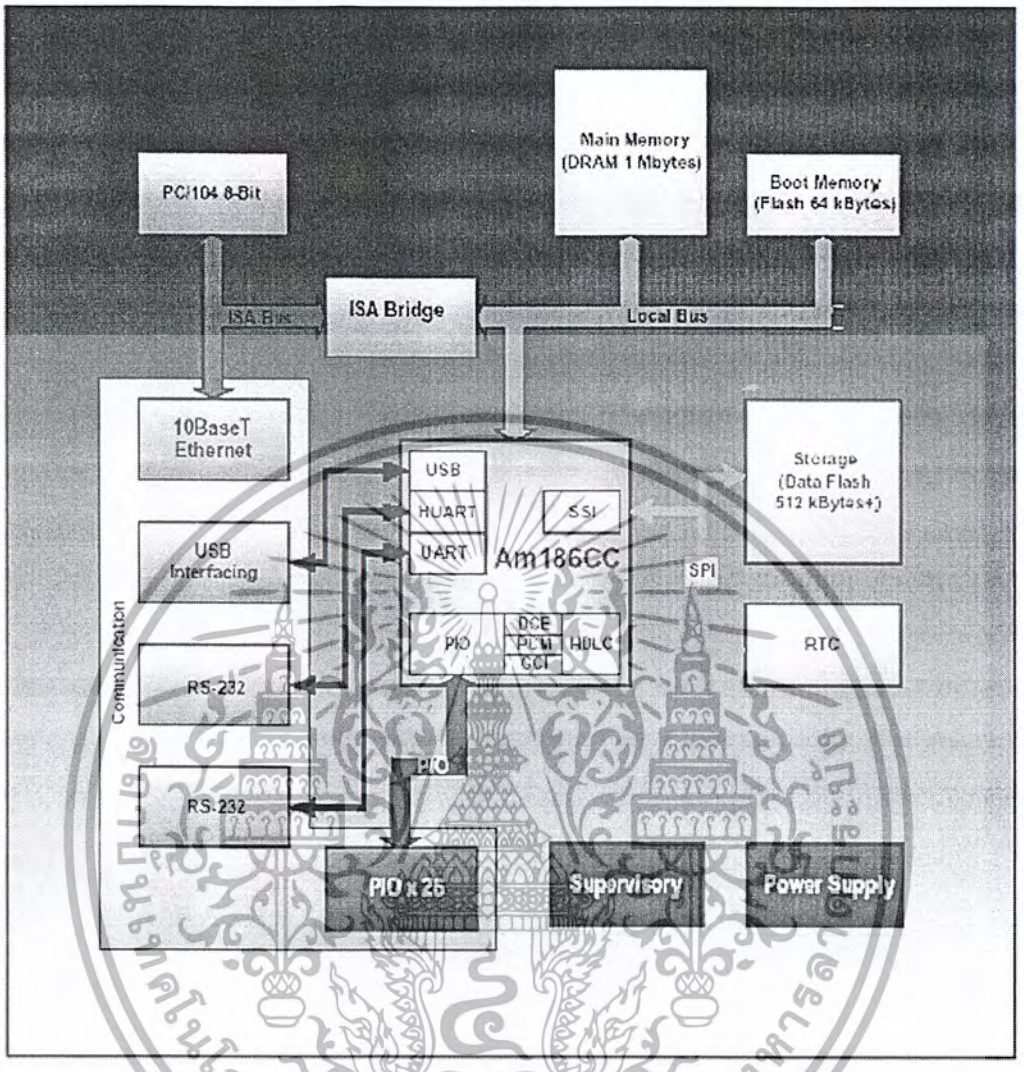
ในส่วนของการเพิ่มขยายของฮาร์ดแวร์สามารถทำได้โดยผ่านทางคอนเนคเตอร์ ของบอร์ด ในรูปแบบของมาตรฐาน PC/104 ซึ่งจะมีการเชื่อมต่อแบบ BUS interface แบบ ISA BUS ขนาด 8 บิต คล้ายกับ ISA BUS ที่มีอยู่ในคอมพิวเตอร์ส่วนบุคคลทั่วไป ซึ่งผู้พัฒนาสามรถออกแบบฮาร์ดแวร์ให้เป็นลักษณะของบอร์ดขยาย และ นำมาเชื่อมต่อกับบอร์ด คอม 86 และเพียงเขียนซอฟแวร์ไดรเวอร์สำหรับควบคุมการทำงานของส่วนที่เพิ่มเข้ามาใหม่ซึ่ง จะทำงานร่วมกับระบบปฏิบัติการในการรองรับการทำงานของแอพพลิเคชั่นต่างๆ ที่เพิ่มเข้ามา นอกจากนี้ตัวบอร์ด คอม 86 ยังสามรถเชื่อมต่อกับอุปกรณ์ภายนอกในลักษณะของ PIO (Programmable Input/Output) เหมือนกับการใช้งานไมโครคอนโทรเลอร์รุ่นอื่น ๆ ได้ด้วยเช่นกัน



รูปที่ 2-2 แสดงการเชื่อมต่อของบอร์ดขยายกับบอร์ดคอม 86

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 สถาปัตยกรรมทางด้านฮาร์ดแวร์



รูปที่ 2-3 ส่วนประกอบต่าง ๆ ของบอร์ดคอม 86

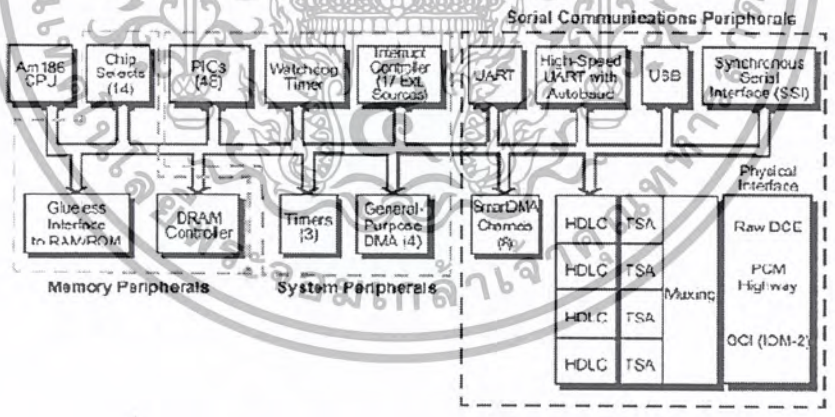
2.1.2.1 ไมโครคอนโทรลเลอร์ Am186CC

ไมโครคอนโทรลเลอร์ AM186CC ถือว่าเป็นหัวใจหลักที่ควบคุมการทำงานส่วนต่าง ๆ ของบอร์ด คอม 86 ซึ่งไมโครคอนโทรลเลอร์ตัวนี้เป็นไมโครคอนโทรลเลอร์ขนาด 16 บิต ในตระกูล อี86 ของบริษัทเอ เอ็ม ดี ชื่อว่า Am186CC ซึ่งเป็นคอมมิวนิเคชันไมโครคอนโทรลเลอร์ (Communication Microcontroller) ที่ใช้ แกนหลัก (Core) ของ 80186 โดยได้เพิ่มความสามารถทางด้านการสื่อสาร และส่วนประกอบอื่น ๆ โดยมี ส่วนสำคัญที่ได้เพิ่มเข้าไปดังนี้

- ส่วนเฮชดีแอลซี (HDLC) หรือ ไฮเลเวลดาต้าลิ้งค์คอนโทรล (High–level Data Link Control) สำหรับเป็นส่วนจัดการเรื่องการติดต่อสื่อสารแบบ เอชดีแอลซี ที่สามารถเชื่อมต่อกับ ดีซีอี, พีซีเอ็ม ไฮเวย์ และ จีซีไอ ผ่านทางพีไอโอโดยมีทีซ่า (TSAs) หรือ ไทม์สลอตแอสซายเนอร์ (Time Slot Assigners) สำหรับการทำงานในแบบมัลติเพลกซ์ (Multiplex)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ยูเอสบีเพอร์ริเฟอรอลคอนโทรล (USB Peripheral Controller) สำหรับการทำเป็นอุปกรณ์ยูเอสบี
- เอชยูอาร์ต (HUART) หรือ ไฮสปีดยูอาร์ต (High-speed UART) สำหรับใช้งานในการเชื่อมต่อ แบบอนุกรมความเร็วสูง โดยสามารถสื่อสารได้ในอัตราเร็วสูงสุด 460 กิโลบิตต่อวินาที (Kbit /s) สามารถตรวจสอบอัตราเร็วในการเชื่อมต่อได้อัตโนมัติ (Automatic Baud Rate Detection) และมีฮาร์ดแวร์แฮนเชคกิ้ง(Handshaking)
- ยูอาร์ต (UART) หรือ ยูนิเวอร์ซัลอะซิงโคนัสรีซีพเวอร์/ทรานส์มิตเตอร์ (Universal Asynchronous Receiver/Transmitter) สำหรับการเชื่อมต่อแบบอนุกรมโดยสามารถสื่อสารได้ในอัตราเร็วสูงสุด 115.2 กิโลบิตต่อวินาที และมีฮาร์ดแวร์แฮนเชคกิ้ง
- เอสเอสไอ (SSI) หรือ ซิงโคนัสซีเรียลอินเทอร์เฟส (Synchronous Serial Interface) สำหรับการเชื่อมต่อ สื่อสารแบบอนุกรมความเร็วสูง 25 เมกกะบิต่วินาที (Mb/s) ซึ่งสามารถสื่อสารแบบ
- โปรแกรมเมเบิลไทม์เมอร์ (Programmable Timer) ขนาด 16 บิต 3 ตัว
- ฮาร์ดแวร์วอชด็อกไทม์ (Hardware Watchdog Time)
- ดีแรมคอนโทรลเลอร์ (DRAM Controller)
- อินเทอร์รัพคอนโทรลเลอร์ (Interrupt Controller) 36 มาสก์เอเบิลอินเทอร์รัพ
- โปรแกรมเมเบิลอินพุทเอาท์พุตซิกแนล (Programmable I/O Signal) ซึ่งมีจำนวน 48 แชนแนล



รูปที่ 2-4 องค์ประกอบของไมโครคอนโทรลเลอร์AM186CC

ไมโครคอนโทรลเลอร์ Am186CC นี้ เป็นไมโครคอนโทรลเลอร์ที่ที่มีสถาปัตยกรรมแบบ x86 มีชุด คำสั่งที่เข้ากันได้กับชุดคำสั่งของ 80286 (Real Mode) สนับสนุนการอ้างอิงหน่วยความจำหลักได้ถึง 1 เมกกะไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2.2 หน่วยความจำ (Memory)

ภายในบอร์ดคอม86 ประกอบด้วยหน่วยความจำหลายส่วนประกอบเข้าด้วยกันซึ่งมีรายละเอียดต่าง ๆ ดังนี้

1) แรม (RAM)

หน่วยความจำส่วนนี้เป็นส่วนที่ใช้ในการทำงานของโปรแกรมต่างๆ ของบอร์ดคอม86 ซึ่งมีขนาด 1 เมกกะไบต์ อันเนื่องมาจากจากสถาปัตยกรรมของไมโครคอนโทรลเลอร์หน่วยความจำในส่วนนี้จะใช้ เป็นพื้นที่ในการใช้งานสำหรับโปแกรมต่างๆทั้งระบบปฏิบัติการและโปรแกรมของนักพัฒนา โดยหน่วย ความจำนี้เป็นหน่วยความจำชนิดอีดีโอ (EDO) ขนาด 1 เมกกะไบต์โดยมีการควบคุมการทำงานจากส่วนของดีแรมคอนโทรลเลอร์ภายในไมโครคอนโทรลเลอร์

2) แฟลชไบออส

หน่วยความจำส่วนนี้เป็นหน่วยความจำแบบแฟลชขนาด 64 กิโลไบต์สำหรับใช้ในการเก็บโปรแกรม ไบออสสำหรับควบคุมการทำงานพื้นฐานของระบบ หน่วยความจำในส่วนนี้จะถูกเรียกใช้งานเป็นอันดับแรกเมื่อระบบเริ่มทำงาน

3) ซีเรียลดาต้าแฟลช (Serial DataFlash)

หน่วยความจำแบบแฟลชตัวนี้จะทำงานทำหน้าที่เป็นสโตร์เรจ (Storage) สำหรับใช้ในการเก็บข้อมูลต่างๆ ของระบบที่ไม่ต้องการให้เกิดการสูญหายในขณะที่บอร์ดคอม86 ขาดกระแสไฟฟ้าหล่อเลี้ยง เช่น ข้อมูลจากการตรวจวัดต่างๆ, โปรแกรมแอพพลิเคชั่นต่างๆ รวมไปถึงระบบปฏิบัติการด้วย ซึ่งบนบอร์ด คอม86 นี้ได้ติดตั้ง ดาต้าแฟลช (IC U12) เบอร์ เอที45ดีบี041ขนาด 512 กิโลไบต์สำหรับใช้งานในส่วนนี้ และได้ออกแบบให้สามารถขยายขนาดได้ตามความต้องการ ซึ่งในการใช้งานนักพัฒนาจะมองหน่วยความจำในส่วนนี้เปรียบเสมือนดิสก์ไดรฟ์อันหนึ่ง (ไดรฟ์ A) ของเครื่องคอมพิวเตอร์

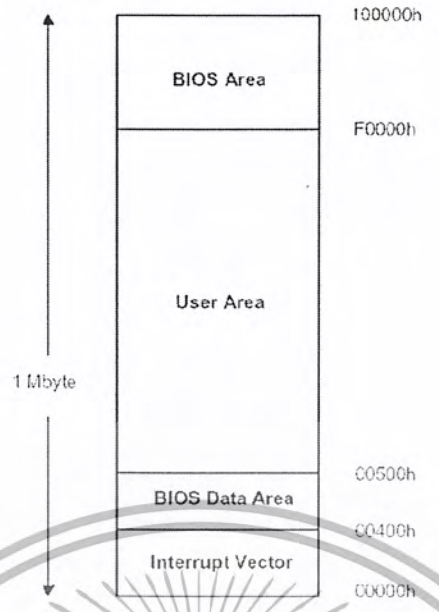
4) แผนผังหน่วยความจำ

จากหน่วยจำทั้งหมดที่กล่าวมาในข้างต้นสามารถแสดงเป็นภาพการจัดวางของหน่วยความจำทั้งหมดได้ดังนี้คือ

- หน่วยความจำหลัก

หน่วยความจำในส่วนนี้มีขนาด 1 เมกกะไบต์ สำหรับเป็นพื้นที่ใช้งานของโปรแกรมต่าง ๆ ซึ่งมีการจัดแบ่ง ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

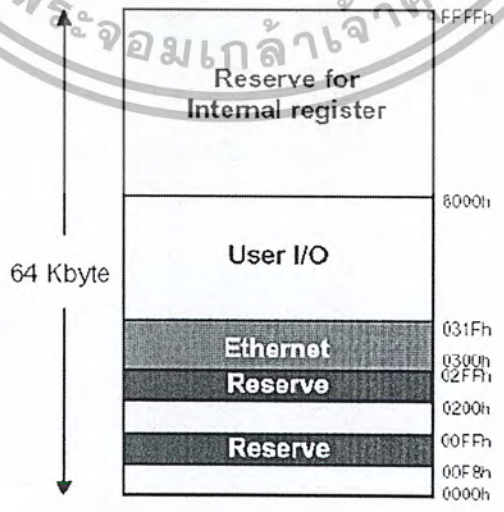


รูปที่ 2-5 แสดงการแบ่งการใช้งานหน่วยความจำหลัก

จากรูปหน่วยความจำช่วงแรกคือจากตำแหน่ง 00000h จนถึงตำแหน่ง 003FFh เป็นส่วนของอินเทอร์รัพ เวกเตอร์เทเบิล (Interrupt Vector Table) สำหรับรองรับการทำงานของการอินเทอร์รัพ (Interrupt) ที่เกิดขึ้น ส่วนต่อ มาคือช่วงตำแหน่ง 00400h ถึง 004FFh หน่วยความจำ ส่วนนี้ใช้สำหรับเก็บข้อมูลของไบออส (BIOS) ส่วนถัดมาคือช่วงตำแหน่ง 00500h ถึง F0000h ส่วนนี้จะเป็นพื้นที่สำหรับใช้งาน โดยโปรแกรมทั่วไปทั้งระบบปฏิบัติการและโปรแกรมของนักพัฒนาที่สร้างขึ้นมา ส่วนที่เหลือในช่วงตำแหน่ง F0000h ถึง FFFFFh ส่วนนี้จะเป็นพื้นที่ที่ถูกสงวนไว้สำหรับไบออสของบอร์ดคอม86

- **หน่วยความจำที่เป็นส่วนอินพุท/เอาท์พุท (I/O)**

ส่วนนี้คือพื้นที่สำหรับใช้ในการติดต่อกับอุปกรณ์อินพุท/เอาท์พุทต่างๆ และใช้ในการติดต่อกับรีจิสเตอร์ภายในของไมโครคอนโทรลเลอร์ ซึ่งมีการจัดวางดังรูป



รูปที่ 2-6 แสดงการจัดวางหน่วยความจำในส่วนของอินพุท/เอาท์พุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพจะแบ่งตำแหน่งของหน่วยความจำได้ออกเป็น 2 ส่วนหลักคือส่วนบนตั้งแต่ตำแหน่งที่ 8000h ถึงตำแหน่ง FFFFh ส่วนนี้จะเป็นพื้นที่สำหรับการใช้งานโดยตัวบอร์ดคอม86 และเป็นส่วนที่ใช้ในการติดต่อรีจิสเตอร์ต่างๆของไมโครคอนโทรลเลอร์ซึ่งบริเวณนี้จะถูกสงวนไว้ไม่สามารถนำมาใช้งานได้ ส่วนที่สองคือส่วนของหน่วยความจำส่วนล่างคือ ตั้งแต่ตำแหน่ง 0000h ถึงตำแหน่ง7FFFh บริเวณนี้จะเป็นพื้นที่สำหรับให้นักพัฒนานำส่วนที่ว่างไปใช้งานโดยเว้นส่วนที่สงวนไว้ใช้งานสำหรับตัวไมโครคอนโทรลเลอร์คือ ตำแหน่ง 00F8h ถึงตำแหน่ง 00FFh และส่วนที่สงวนไว้สำหรับการใช้งานของวงจรอีเทอร์เน็ทคือในตำแหน่ง 0300h จนถึง 031Fh บริเวณที่เหลือนอกจากที่กล่าวมานักพัฒนาสามารถนำไปใช้งานได้ตามต้องการ

2.1.2.3 พอร์ตอนุกรม

บนบอร์ดคอม86 มีพอร์ตการสื่อสารแบบอนุกรมตามมาตรฐาน อาร์เอส-232 สำหรับการนำไปใช้งานจำนวน 2 พอร์ตด้วยกัน ซึ่งอยู่ในรูปแบบของคอนเนคเตอร์แบบ ดีบี9 บนบอร์ด (J8, J9) ซึ่งพอร์ตอนุกรม ทั้ง 2 นี้เชื่อมต่อมาจากวงจรยูอาร์ต ที่อยู่ภายในไมโครคอนโทรลเลอร์ AM186CC และทำการปรับแรงดันจาก 5 โวลท์ให้เข้ากับมาตรฐาน อาร์เอส232 โดยชิพ อาร์เอส232 ไดร์เวอร์ (U9, U10) ซึ่งพอร์ตอนุกรมทั้งสองพอร์ตนี้ พอร์ต คอม1 จะถูกนำไปใช้งานสำหรับเป็นซีเรียลคอนโซล (Serial Console) สำหรับใช้ในการแสดงผลควบคุมการทำงาน ต่างๆ ของบอร์ดในขณะพัฒนาโปรแกรม ส่วนพอร์ต คอม2 นั้นสามารถนำไปใช้งานได้โดยทั่วไป ซึ่งการเชื่อมต่อสัญญาณต่างๆ ของพอร์ตอนุกรมทั้งสองนี้เป็นไปดังภาพ

'NC คือขาสัญญาณที่ไม่มีการเชื่อมต่อ

รูปที่ 2-7 คอนเนคเตอร์ของพอร์ตอนุกรม

สำหรับในการพัฒนานั้น ภายในชุดพัฒนาคอม86 จะมีสายเชื่อมต่อระหว่างพอร์ตคอม1 ที่ทำหน้าที่เป็น คอนโซล (Console) กับคอมพิวเตอร์ซึ่งจะเป็นสายชนิดนูโมเด็ม (null modem) สำหรับใช้ในการแสดงผลและรับ-ส่ง ข้อมูลต่างๆ ในระหว่างการพัฒนาโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2.4 ระบบรีเซต

ภายในบอร์ดคอม86 จะมีวงจรสำหรับสร้างสัญญาณรีเซตให้กับอุปกรณ์ต่างๆ บนบอร์ดโดยใช้ชิพเอลเอ็ม3722 (U15) เป็นตัวควบคุมการทำงานโดยวงจรจะสร้างสัญญาณรีเซตเมื่อมีการกดสวิทช์รีเซต (SW1) บนบอร์ดหรือในกรณีที่แรงดันไฟฟ้าบนบอร์ดในส่วนของแหล่งจ่ายแรงดัน 3.3 โวลต์มีแรงดันต่ำกว่า 3.08 โวลต์

2.1.2.5 ยูอาร์ต UART (Universal Asynchronous Receiver - Transmitter)

เป็นส่วนประกอบที่ใช้สำหรับการสื่อสารระหว่าง อุปกรณ์อินพุตและเอาท์พุต ที่มีการ ส่งข้อมูลเป็นแบบ อนุกรมซึ่ง ยูอาร์ตจะควบคุมอัตราการส่งข้อมูลที่ทั้งสองฝั่งของการรับส่ง เพื่อไม่ให้เกิดปัญหาเมื่ออัตราเร็วในการรับส่งข้อมูลทั้ง 2 ด้านไม่เท่ากัน

2.1.2.6 พาวเวอร์ซัพพลาย (Power supply)

ระบบจ่ายไฟของบอร์ดคอม86 แบ่งได้ออกเป็น 2 ส่วนหลักด้วยกันคือ ในส่วนแรกจะเป็นระบบจ่ายไฟแรงดัน 5 โวลท์ (Volt) ซึ่งจะมีไอซีแอลดีโอ LDO (U14) ทำหน้าที่จ่ายกระแสไฟฟ้าสำหรับเลี้ยงอุปกรณ์บอร์ด ที่ต้องการแรงดันไฟฟ้าในการทำงานที่ 5 โวลท์ เช่น ดีแรม, แฟลชไบออส, เรียลไทม์คล็อก เป็นต้น และสำหรับอุปกรณ์ที่ต้องการแรงดันไฟฟ้า 3.3 โวลท์ ในการทำงานนั้นจะมีไอซีแอลดีโอ LDO (U16) ทำหน้าที่จ่ายกระแสไฟฟ้าให้ซึ่งอุปกรณ์เหล่านี้ได้แก่ตัวไมโครคอนโทรลเลอร์, แฟลชดิสก์เป็นต้น ซึ่งอินพุตของภาคพาวเวอร์ซัพพลายของบอร์ดคอม86 นั้นจะรับเป็นไฟฟ้ากระแสตรงที่แรงดันประมาณ 8 โวลท์ และทำการลดระดับแรงดันลงเป็น 5 โวลท์และ 3.3 โวลท์ตามลำดับ โดยภายในชุดพัฒนาจะมีอแดปเตอร์ให้ 1 ตัวสำหรับเป็นส่วนจ่ายกระแสไฟฟ้าให้กับวงจรทั้งหมด

โดยทั่วไปส่วนวงจรจ่ายกระแสไฟฟ้าแรงดัน 5 โวลท์และ 3.3 โวลท์จะสามารถจ่ายกระแสไฟให้ไฟฟ้ากับวงจรที่ต่อเพิ่มเติมได้อีก 300 มิลลิแอมป์ (mA) หากมีการต่อวงจรส่วนขยายที่ต้องการใช้งานกระแสไฟฟ้ามากกว่านี้ แนะนำให้ควรมีการต่อวงจรพาวเวอร์ซัพพลายจากภายนอกเพิ่มเติมเพื่อป้องการเสียหายของ ส่วนวงจรภาคจ่ายไฟของบอร์ดคอม86

2.1.2.7 แอลอีดีอินดิเคเตอร์ (LED Indicator)

บนบอร์ดคอม86 จะมีแอลอีดี (LED) สำหรับแสดงสถานะต่างๆดังนี้คือ ดวงแรกเป็นแอลอีดีของพาวเวอร์ซัพพลาย (D4) มีสีแดงโดยจะติดสว่าง เมื่อบอร์ดคอม86 ได้รับการจ่ายกระแสไฟฟ้าจากภายนอกและวงจรพาวเวอร์ซัพพลาย 5 โวลต์ และ 3 โวลต์ สามารถทำงานได้อย่างถูกต้อง ดวงที่ 2 คือ LED สีเขียวของส่วนวงจรอีเทอร์เน็ท(D2) ซึ่งดวงนี้จะติดสว่างขึ้นเพื่อแสดงว่าการมีการเชื่อมต่อสายอีเทอร์เน็ทและพร้อมที่จะรับส่ง ข้อมูล ดวงที่ 3 คือ Lแอลอีดีดวงสีเขียวของส่วนวงจรอีเทอร์เน็ทที่เหลืออีกดวงหนึ่ง (D1) ซึ่งดวงนี้จะติดกระพริบเมื่อมีการรับส่ง ข้อมูลผ่านทางอีเทอร์เน็ทพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 สถาปัตยกรรมทางด้านซอฟต์แวร์

1) BIOS เป็นโปรแกรมที่คอยควบคุมการทำงานของฮาร์ดแวร์ต่างๆบนบอร์ดคอม86 เป็นส่วนที่จัดการการสื่อสารระหว่างฮาร์ดแวร์กับแอพพลิเคชันซอพท์แวร์ และระบบปฏิบัติการ

2) Driver เป็นโปรแกรมขนาดเล็กที่ทำหน้าที่ติดต่อสื่อสารกับฮาร์ดแวร์ อื่นๆที่ไบออสไม่ได้รองรับการทำงานไว้ เช่น ฮาร์ดแวร์ที่สร้างขึ้นมาเพิ่มเติมเพื่อเชื่อมต่อเข้ากับบอร์ดคอม86

3) Operating System ระบบปฏิบัติการเป็นโปรแกรมที่ควบคุมการทำงานของโปรแกรมต่าง ๆ ในระบบ ทั้งในส่วนของการจัดการหน่วยความจำ เป็นส่วนสื่อสารระหว่างโปรแกรมแอพพลิเคชันของผู้ใช้และ ส่วนของฮาร์ดแวร์ระบบต่าง ๆ

4) Packet Driver เป็นโปรแกรมจัดการเกี่ยวกับแพ็กเก็ตของการส่งข้อมูล สำหรับแอพพลิเคชันทางด้านการสื่อสาร

5) TCP/IP เป็นโปรแกรมจัดการทางด้านโพรโตคอลทีซีพี/ไอพีสำหรับแอพลิเคชันทางด้านการสื่อสาร

6) Network Application แอพพลิเคชั่นทางด้านการสื่อสาร เป็นโปรแกรมแอพพลิเคชันทางด้านการสื่อสาร ต่างๆ เช่น โปรแกรมเว็บเซิร์ฟเวอร์ในระบบฝังตัว (Embedded web server) เป็นต้น

7) Application เป็นโปรแกรมแอพลิเคชันอื่นๆ ที่ไม่ได้ใช้ความสามารถทางด้านการสื่อสาร เช่น โปรแกรมควบคุมการทำงานของหุ่นยนต์ หรือเครื่องจักร เป็นต้น

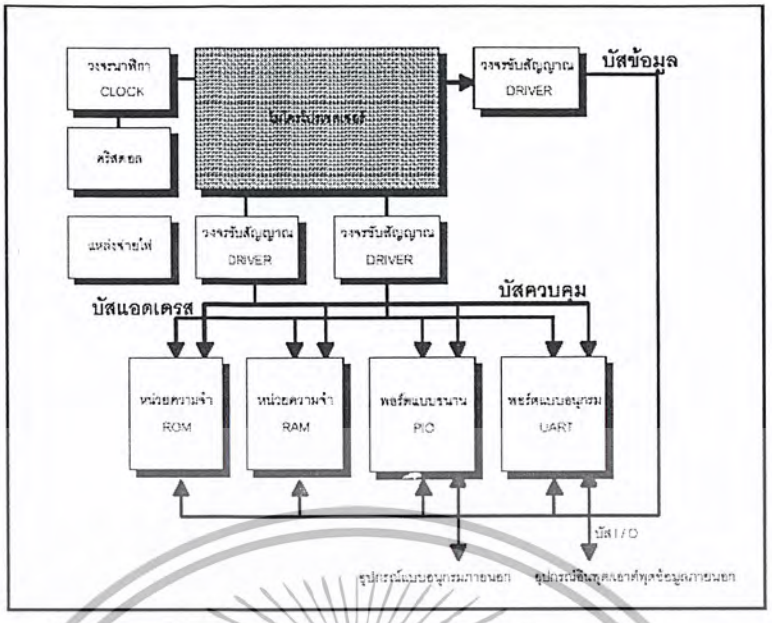
2.2 ไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์มาจากคำ 2 คำรวมกันคือ "ไมโคร" (Micro) ซึ่งหมายถึง ไมโครโปรเซสเซอร์ (Microprocessor) ซึ่งเป็นอุปกรณ์ประมวลผลข้อมูลขนาดเล็ก ซึ่งภายในประกอบด้วย หน่วยประมวลผลกลางหรือ ซีพียู (CPU : Central Processing Unit) หน่วยคำนวณทางคณิตศาสตร์และลอจิก (ALU : Arithmetic Logic Unit) วงจรเชื่อมต่อหน่วยความจำ และวงจรเชื่อมต่อสัญญาณนาฬิกา อีกคำหนึ่งคือคำว่า " คอนโทรลเลอร์" (Controller) หมายถึงอุปกรณ์ควบคุม โดยที่สามารถเขียนโปรแกรมเพื่อกำหนดรูปแบบการควบคุมได้อย่างอิสระ

2.2.1 ความแตกต่างของไมโครคอนโทรลเลอร์กับไมโครโปรเซสเซอร์

โครงสร้างของไมโครโปรเซสเซอร์ซึ่งประกอบด้วยหน่วยประมวลผลกลาง หน่วยคำนวณทางคณิตศาสตร์และลอจิก บัสข้อมูลและแอดเดรสสำหรับติดต่อกับหน่วยความจำภายนอก และวงจรกำเนิดสัญญาณนาฬิกา นั่นหมายความว่า การใช้งานไมโครโปรเซสเซอร์จะต้องเชื่อมต่อกับอุปกรณ์อินพุตและเอาท์พุท ต้องอาศัยอุปกรณ์ที่เรียกว่า ไอซีพอร์ตขยาย (Port Expander) ทำให้การสร้างระบบควบคุมด้วยไมโครโปรเซสเซอร์จึงมีความต้องการอุปกรณ์จำนวนมาก ส่งผลให้ขนาดของระบบใหญ่พอสมควร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-8 โครงสร้างพื้นฐานของไมโครโปรเซสเซอร์

จะเห็นได้อย่างชัดเจนว่า ภายในไมโครคอนโทรลเลอร์มีอุปกรณ์พื้นฐานเหมือนกับไมโครโปรเซสเซอร์หากแต่จะบรรจุหน่วยความจำโปรแกรม หน่วยความจำข้อมูลและพอร์ตอินพุทเอาท์พุทไว้ภายในพร้อมสรรพ ผู้ใช้งานจึงเพียงแต่เขียนโปรแกรมควบคุมลงบนตัวไมโครคอนโทรลเลอร์ แล้วต่ออุปกรณ์ที่ใช้ในการสร้างวงจรกำเนิดสัญญาณนาฬิกา อาทิ คริสตอล ตัวเก็บประจุ เป็นต้น สุดท้ายเชื่อมต่ออุปกรณ์อินพุทเอาท์พุทที่ต้องการเข้ากับพอร์ตของไมโครคอนโทรลเลอร์เพียงเท่านี้ก็สามารถใช้ไมโครคอนโทรลเลอร์ได้แล้ว ส่งผลให้ขนาดและราคาของระบบลดลงอย่างมาก

คุณสมบัติ	ไมโครโปรเซสเซอร์	ไมโครคอนโทรลเลอร์
ขนาดของหน่วยประมวลผลกลาง	ไม่น้อยกว่า 8 บิต	ส่วนใหญ่จะมีขนาด 8 บิต
หน่วยคำนวณทางคณิตศาสตร์และลอจิก	มีอยู่ภายใน	มีอยู่ภายใน
วงจรกำเนิดสัญญาณนาฬิกา	มีอยู่ภายใน	มีอยู่ภายใน
การเชื่อมต่อกับหน่วยความจำโปรแกรม	เชื่อมต่อภายนอกเท่านั้น	ใช้ได้ทั้งภายในและภายนอก
การเชื่อมต่อกับหน่วยความจำข้อมูล	เชื่อมต่อภายนอกเท่านั้น	ใช้ได้ทั้งภายในและภายนอก
การเชื่อมต่อกับพอร์ตอินพุทเอาท์พุท	เชื่อมต่อภายนอกเท่านั้น	มีอยู่ภายในสามารถขยายได้
ไทเมอร์ เคาน์เตอร์	ไม่มีในซีพียูขนาดเล็ก	มีอย่างน้อย 1 ตัวขนาด 8-16 บิต
วอตช์ด็อกไทเมอร์	ไม่มีในซีพียูขนาดเล็ก	มีอย่างน้อย 1 ตัว
จำนวนขาต่อใช้งาน	ไม่น้อยกว่า 40 ขา	มีตั้งแต่ 8 ขึ้นไป

ตารางที่ 2-1 คุณสมบัติของไมโครค้อนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตาม ไมโครคอนโทรลเลอร์ก็สามารถเชื่อมต่อกับหน่วยความจำภายนอกได้เช่นกัน โดยพิจารณาให้หน่วยความจำภายนอกนั้นเป็นอุปกรณ์อินพุทเอาท์พุทตัวหนึ่ง แล้วใช้ขาพอร์ตที่มีอยู่ทำการติดต่อ ในตารางที่ 2-1 เป็นตารางสรุปความแตกต่างที่สำคัญระหว่างไมโครโปรเซสเซอร์และไมโครคอนโทรลเลอร์

2.2.2 จะเลือกใช้ไมโครโปรเซสเซอร์หรือไมโครคอนโทรลเลอร์เมื่อใด

ถ้าเป็นการสร้างระบบควบคุมขนาด 8 บิต มีความต้องการเชื่อมต่อกับอุปกรณ์ภายนอกไม่มากนัก (น้อยกว่า 10 แบบ) ควรเลือกใช้ไมโครคอนโทรลเลอร์ และถ้าหากต้องมีการประมวลผลข้อมูลลงในหน่วยความจำข้อมูลมากเป็นกิโลไบต์ ควรออกแบบให้ไมโครคอนโทรลเลอร์ในระบบควบคุมนี้เชื่อมต่อกับหน่วยความจำภายนอก

ถ้าหากต้องใช้งานกับข้อมูลมากกว่า 8 บิตตลอดเวลาและต้องการความเร็วในการทำงานสูง ๆ สามารถติดต่อกับหน่วยความจำได้เป็นจำนวนมาก ๆ ติดต่อกับอุปกรณ์อินพุทเอาท์พุทได้จำนวนมากมายในเวลาเดียวกัน ควรเลือกใช้ไมโครโปรเซสเซอร์ ดังจะเห็นได้จากในเครื่องคอมพิวเตอร์ที่ใช้ไมโครโปรเซสเซอร์เป็นหัวใจหลักในการทำงาน ทำให้สามารถเชื่อมต่อกับหน่วยความจำได้มากเป็นหน่วยกิกกะบิต ขนาดของข้อมูลสูงถึง 64 บิต ความเร็วสูงเป็นหลายร้อยเมกกะเฮิรตซ์เป็นต้น

ดังนั้นจึงพอสรุปได้ว่า ไมโครคอนโทรลเลอร์เหมาะสมสำหรับการสร้างระบบควบคุมในขณะที่ไมโคร โปรเซสเซอร์เหมาะสมสำหรับการสร้างระบบประมวลผลข้อมูลความเร็วสูงและระบบควบคุมที่มีขนาดใหญ่มาก ๆ

2.2.3 โครงสร้างและสถาปัตยกรรมของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีหน่วยความจำภายในเป็นแบบแฟลช (Flash Memory) ของ Atmel Corporation มีเบอร์ขึ้นต้นด้วย AT89 เหตุผลที่ใช้ไมโครคอนโทรลเลอร์แบบนี้มาใช้งานมีด้วยกันหลายประการดังนี้

1) หน่วยความจำโปรแกรมภายในตัวไมโครคอนโทรลเลอร์เป็นแบบแฟลช ทำให้สามารถลบและเขียนใหม่ได้นับพันครั้ง จึงสามารถใช้งานในรูปแบบของไมโครคอนโทรลเลอร์ชิพเดี่ยวไม่ต้องใช้หน่วยความจำภายนอก ส่งผลให้สามารถใช้งานพอร์ตอินพุทเอาท์พุทของไมโครคอนโทรลเลอร์ได้อย่างเต็มประสิทธิภาพ

2) ต้นทุนและเวลาในการพัฒนาระบบไมโครคอนโทรลเลอร์ลดลงอย่างมากเนื่องจากไม่ต้องใช้เครื่องมือพัฒนาจำพวกอีมูเลเตอร์และเครื่องโปรแกรมอีพรอม

3) บริษัทผู้ผลิตได้ทำการผลิตไมโครคอนโทรลเลอร์ตระกูลนี้ออกมาหลายเบอร์ และมีความสามารถต่างกันไป ทำให้มีทางเลือกในการใช้งานสูง

4) ด้วยการใช้หน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ ทำให้สามารถป้องกันการคัดลอกข้อมูลของหน่วยความจำเป็นอย่างดี

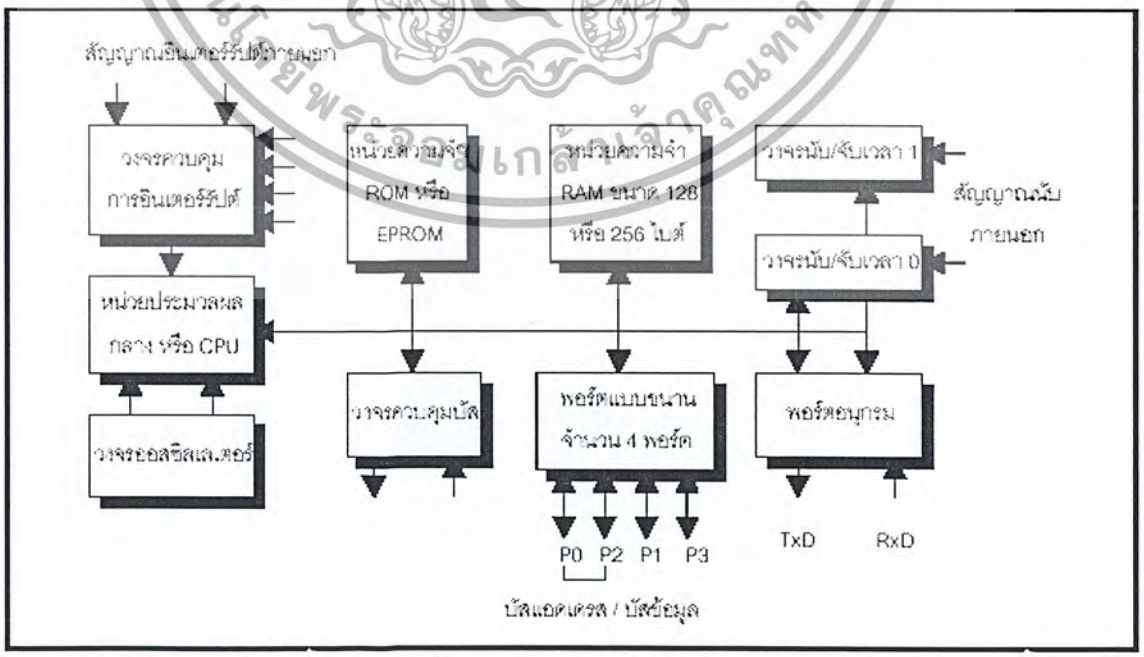
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) ในบางเบอร์ของไมโครคอนโทรลเลอร์ที่ผลิตโดย Atmel สามารถโปรแกรมข้อมูลในหน่วยความจำได้โดยไม่ต้องถอดตัวไมโครคอนโทรลเลอร์ออกมาทำการโปรแกรมใหม่หรือเรียกว่าการโปรแกรมในวงจร หรือในระบบ (In-system Programming) โดยใช้ลักษณะการติดต่อแบบ SPI (Serial Peripheral Interface) ทำให้การพัฒนาหรือซ่อมบำรุงตลอดจนการปรับปรุง หรืออัปเกรดข้อมูลในหน่วยความจำทำได้อย่างสะดวกภายใต้งบ ประมาณที่ไม่สูงมากนัก

6) ชุดคำสั่งและสถาปัตยกรรมพื้นฐานเหมือนกับไมโครคอนโทรลเลอร์ MCS-51 ของผู้ผลิตอื่น ไม่ว่าจะเป็น อินเทล , ซีเมนส์ หรือ ดัลลัส

2.2.4 คุณสมบัติทางเทคนิคของไมโครคอนโทรลเลอร์ตระกูล MCS-51

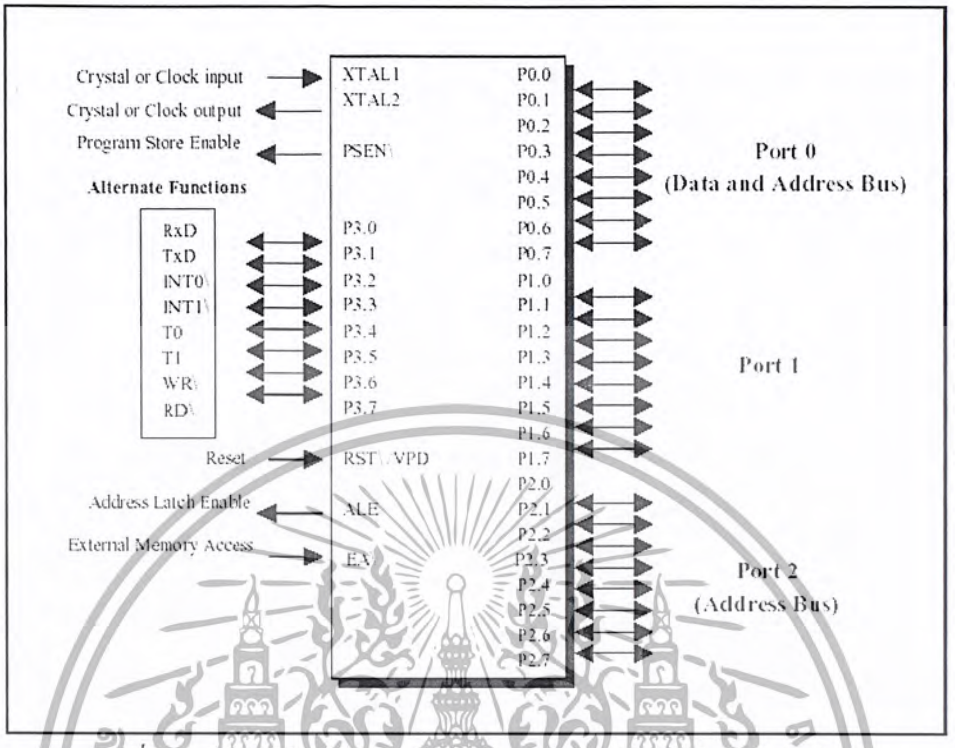
- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลช สามารถลบและเขียนใหม่ได้พันครั้ง
- หน่วยความจำข้อมูลพื้นฐานเป็นแบบแรมในบางเบอร์จะมีหน่วยความจำแบบอีอีพรอมเพิ่มเติม
- ขาพอร์ตแบบ สองทิศทาง สามารถใช้งานได้ทั้งอินพุทและเอาท์พุท
- มีวงจรสื่อสารอนุกรมแบบฟูลดูเพลกซ์
- ไทเมอร์/เคาน์เตอร์ขนาด 16 บิตอย่างน้อย 2 ตัว
- สามารถรองรับแหล่งกำเนิดอินเตอร์รัพต์ได้ 6 ประเภท
- สามารถขยายหน่วยความจำภายนอกเพิ่มเติมได้สูงสุด 64 กิโลไบต์
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ภายในชิพ
- มีวงจรสื่อสารข้อมูลแบบ SPI สำหรับในอนุกรม AT89Sxx
- มีวอตช์ด็อกไทเมอร์ในตัวสำหรับ AT89Sxx



รูปที่ 2-9 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.5 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51



รูปที่ 2-10 แสดงการจัดวางตำแหน่งขาของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 ทุกเบอร์จะมีสถาปัตยกรรมและขาใช้งานพื้นฐานเหมือนกันดังแสดงในรูป 2-8 และรูปที่ 2-9 โดยมีรายละเอียดข้างต้นดังนี้

- ขา Vcc ใช้สำหรับต่อไฟเลี้ยง +5
- ขา GND เป็นขากราวด์สำหรับต่อกราวด์ของระบบ
- ขาพอร์ต (P0.0-P0.7) มี 8 ขาแต่ละขากำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุทสำหรับใช้งานทั่วไป ถ้าต้องการกำหนดให้ขาพอร์ต 0 ขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูลเป็น 1 ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสภาวะปล่อยลอยจึงมีอินพุทอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุทได้ นอกจากนี้ขาพอร์ตนี้ยังถูกใช้งานติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำภายนอก (A0-A7) และขาข้อมูล (D0-D7) โดยการใช้กระบวนการมัลติเพลกซ์เข้าช่วยเพื่อสลับการทำงานให้เป็นได้ทั้งขาติดต่อแอดเดรสและขาข้อมูล

ขาพอร์ต 1 (P1.0-P1.7) มี 8 ขาแต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าต้องการกำหนดให้ขาพอร์ต 1 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล 1 ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย นอกจากนั้นในอนุกรม AT89Sxx จะใช้ขา P1.0 เป็นขาอินพุทสำหรับนับค่าของไทเมอร์ 2 และขา P1.1 เป็นขาอินพุททริกเกอร์ของไทเมอร์2 ในขณะที่ขา P 1.4 ถึง P1.7 เป็นขาสำหรับต่อเชื่อม SPI เพื่อทำการโปรแกรมข้อมูลในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ขาพอร์ต 2 (P2.0-P2.7) มี 8 ขาแต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าต้องการกำหนดให้ขาพอร์ต 2 ขาใดขาหนึ่งให้เป็นได้ทั้งอินพุตสามารถทำได้โดยการเขียนข้อมูล 1 ไปยังบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอยจึงมีอินพุตอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนี้ขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความจำภายนอก (A8-A15)

ขาพอร์ต 3 (P3.0-P3.7) มี 8 ขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาท์พุทสำหรับใช้งานทั่วไป ถ้าต้องการให้ขาพอร์ต 3 ขาใดขาหนึ่งเป็นอินพุต สามารถทำได้โดยการเขียนข้อมูล 1 ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขานั้นมีสถานะปล่อยลอยจึงมีอินพุตอิมพีแดนซ์สูง สามารถใช้งานเป็นขาอินพุตได้ นอกจากนั้นขาพอร์ต 3 ยังทำหน้าที่พิเศษ ดังมีรายละเอียดข้างต้น ต่อไปนี้

- P3.0 ใช้เป็นขาอินพุตสำหรับรับข้อมูลจากการสื่อสารข้อมูลแบบอนุกรมหรือขา RxD
- P3.1 ใช้เป็นขาอินพุตสำหรับรับข้อมูลจากการสื่อสารแบบอนุกรมหรือขา TxD
- P3.2 ใช้เป็นขาอินพุตสำหรับรับสัญญาณอินเตอร์รัปต์จากภายนอกช่องที่ 0 หรือขา INTO
- P3.3 ใช้เป็นขาอินพุตสำหรับรับสัญญาณอินเตอร์รัปต์จากภายนอกช่องที่ 1 หรือขา INTI
- P3.4 ใช้เป็นขาอินพุตสำหรับรับสัญญาณไทเมอร์จากภายนอกช่องที่ 0 หรือขา T0
- P3.5 ใช้เป็นขาอินพุตสำหรับรับสัญญาณไทเมอร์จากภายนอกช่องที่ 1 หรือขา T1
- P3.6 ใช้เป็นขาสัญญาณ WR ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก
- P3.7 ใช้เป็นขาสัญญาณ RD ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก

ขารีเซต ใช้ในการรีเซตการทำงานของไมโครคอนโทรลเลอร์ โดยการป้อนสัญญาณเพื่อรีเซตสถานะที่ขานี้ต้องอยู่ในระดับรีเซตอย่างน้อย 2 แมชชีนไซเคิล โดยที่วงจรกำเนิดสัญญาณนาฬิกายังคงทำงานต่อเนื่องไปอย่างเป็นปกติ

ขา ALE/PROG (Address Latch Enable/Program pulse input) เป็นขาที่ใช้ในการควบคุมการแลตช์ของขาพอร์ต 0 เมื่อมีการใช้งานหน่วยความจำภายนอก นอกจากนั้นขานี้ยังใช้เป็นขาสำหรับรับพัลซ์ของการโปรแกรมสำหรับโปรแกรมข้อมูลลงในไมโครคอนโทรลเลอร์ MCS-51 ในรุ่นที่มีหน่วยความจำโปรแกรมเป็นอีอีพรอม

ขา PSEN (Program Store Enable) ขานี้ใช้ในการส่งสัญญาณเพื่อร้องขอติดต่อกับหน่วยโปรแกรมภายนอกตัวไมโครคอนโทรลเลอร์ จะส่งสัญญาณออกมาที่ขานี้ 2 ครั้ง ในแต่ละแมชชีนไซเคิล แต่ถ้าหากติดต่อกับหน่วยความจำภายนอก ขานี้จะไม่มีการส่งสัญญาณใดๆออกมา

ขา EA/Vpp (External Access enable /Programing voltage input) ใช้สำหรับเลือกการติดต่อหน่วยความจำโปรแกรมจากภายนอกหรือภายในตัวไมโครคอนโทรลเลอร์หรือหากขานี้เป็น " 0 " เป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำโปรแกรมภายนอก แต่ถ้าหากขานี้เป็น " 1 " เป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำโปรแกรมภายในไมโครคอนโทรลเลอร์ นอกจากนี้ที่ขานี้ยังเป็นขาอินพุตสำหรับรับแรงดันไฟสูงสำหรับการโปรแกรมหน่วยความจำภายในไมโครคอนโทรลเลอร์ สำหรับในไมโครคอนโทรลเลอร์ MCS-51 แฟลชต้องการแรงดันสำหรับโปรแกรม +12 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
55126

ขา XTAL1 และ XTAL2 เป็นขาสำหรับต่อคริสตอลเพื่อสร้างสัญญาณนาฬิกาในการกำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์

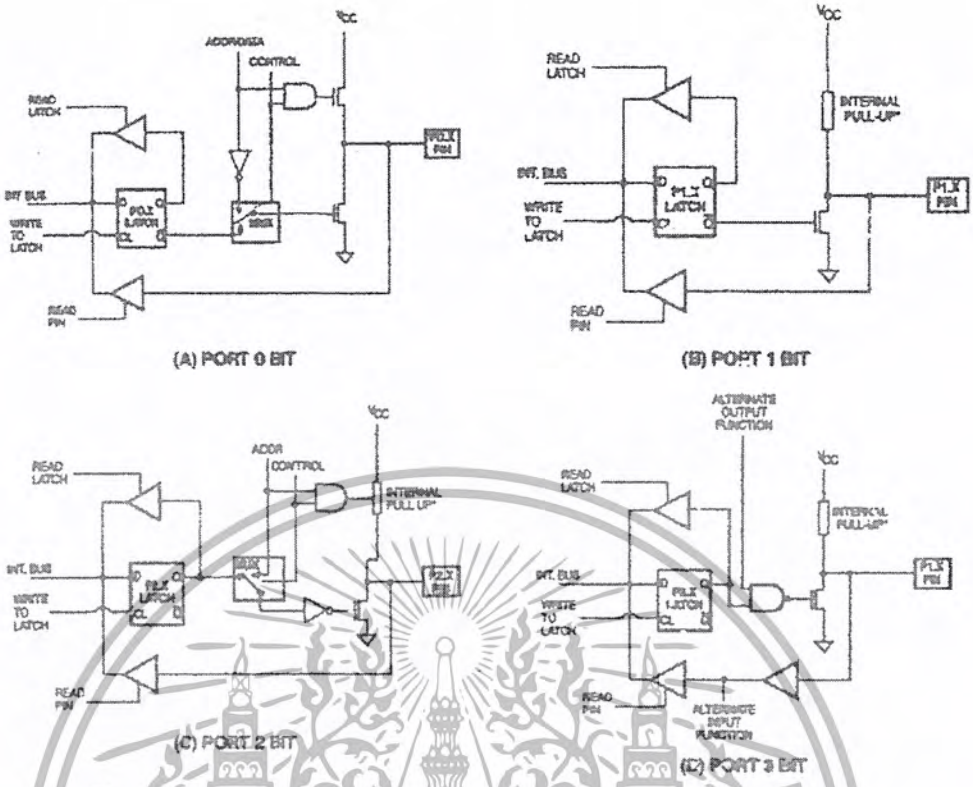
2.2.6 โครงสร้างและการทำงานของพอร์ต

ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ต คือ พอร์ต 0 ถึง พอร์ต 3 แต่ละพอร์ตมีขนาด 8 บิต เป็นพอร์ตแบบ 2 ทิศทาง กล่าวคือ สามารถเป็นได้ทั้งอินพุตสำหรับรับข้อมูลเข้า และเอาท์พุทสำหรับส่งสัญญาณข้อมูลออก ทุกพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีวงจรแลตช์และวงจรขับตลอดจนบัฟเฟอร์อินพุตที่พอร์ต 0 และพอร์ต 2 จะใช้งานเป็นพอร์ตอินพุตและเอาท์พุทสำหรับงานทั่วไปและใช้ติดต่อกับหน่วยความจำภายนอก สำหรับพอร์ต 3 ทั้งพอร์ตและพอร์ต 1 บางขา นอกจากจะใช้เป็นขาอินพุตและเอาท์พุทตามปกติแล้ว ยังสามารถใช้งานในหน้าที่พิเศษได้อีก ขึ้นอยู่กับว่าเป็นไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ใด ดังในตารางที่ 2-2

ขา	เบอร์ของไมโครคอนโทรลเลอร์	หน้าที่พิเศษ
P 1.0	AT89C52/AT89Sxx	ขา T2 เป็นขาอินพุตนับค่าของไทเมอร์/เคาน์เตอร์ 2 และเป็นขาเอาท์พุทของการกำเนิดสัญญาณนาฬิกาโดยไทเมอร์2 (clock out)
P 1.1	AT89C52/AT89Sxx	ขา T2EX เป็นขาอินพุตทริกเกอร์สำหรับการแคปเจอร์/รีโหลดและควบคุมทิศทางของสัญญาณ
P 1.4	AT89Sxx	ขา SS(Slave Select) เป็นขาเลือกการติดต่อในกรณีที่ไมโครคอนโทรลเลอร์เป็นอุปกรณ์สเลฟ ในระบบการติดต่อแบบ SPI
ขา	เบอร์ของไมโครคอนโทรลเลอร์	หน้าที่พิเศษ
P 1.5	AT89Sxx	ขา MOSI (Master data output, Slave data input) ใช้ในการติดต่อกับพอร์ต SPI
P 1.6	AT89Sxx	ขา MISO (Master data input, Slave data output)
P1.7	AT89Sxx	ขา SCK(Master clock output) เป็นขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต SPI

ตารางที่ 2-2 หน้าที่พิเศษของพอร์ต 1 ในตัวไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-11 แสดงวงจรภายในของแต่ละพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ในรูปด้านบนแสดงวงจรภายในของแต่ละพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดยภาพ A เป็นวงจรพอร์ต 0 วงจรแลตช์ของแต่ละบิตในแต่ละพอร์ตก็คือ วงจรดีฟลิปฟลอบนั่นเอง การอ่านค่าสถานะของพอร์ตและสถานะของวงจรแลตช์สามารถกระทำได้อย่างอิสระต่อกันด้วยสัญญาณที่แยกออกจากกันนั่นคือสัญญาณอ่านข้อมูลจากขาพอร์ต และสัญญาณอ่านข้อมูลจากวงจรแลตช์ส่วนการเขียนข้อมูลมายังพอร์ตต้องส่งสัญญาณมายังขา CLK ของดีฟลิปฟลอบในขณะที่ข้อมูลจะส่งผ่านมาทางขาบัสข้อมูลภายในเข้าสู่ขา D ของดีฟลิปฟลอบ

ที่พอร์ตนี้มีวงจรมัลติเพลกซ์สำหรับกำหนดลักษณะการทำงานของพอร์ตว่า ต้องการใช้งานที่เป็นขาพอร์ตอินพุตเอาท์พุทปกติ หรือใช้ในการติดต่อกับหน่วยความจำภายนอกไมโครคอนโทรลเลอร์

เนื่องจากที่ขาพอร์ต 0 ไม่มีวงจรพูลอัปภายในหากมีการนำพอร์ต 0 ไปใช้งานเป็นพอร์ตขาอินพุตจะต้องต่อตัวต้านทานพูลอัปภายนอกเข้าที่ขาพอร์ต 0 ทุกขาด้วย

ในภาพ B เป็นวงจรของพอร์ต 1 ซึ่งมีลักษณะโดยทั่วไปคล้ายกับพอร์ต 0 หากแต่ไม่มีวงจรมัลติเพลกซ์เนื่องจากพอร์ตนี้จะไม่ใช้ติดต่อกับหน่วยความจำภายนอก แต่จะมีวงจรพูลอัปภายในที่แต่ละบิตของพอร์ตนนี้แทน

ในภาพ C เป็นวงจรภายในของพอร์ต 2 จะคล้ายกับพอร์ต 0 มาก ต่างกันเพียงมีวงจรพูลอัปเพิ่มเติมเข้ามา ส่วนในภาพ D เป็นวงจรภายในของพอร์ต 3 จะเห็นได้คล้ายกับพอร์ต 1 มีการเพิ่มเติมวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัฟเฟอร์ และวงจรอินพุตเอาท์พุทเมื่อทำงานในฟังก์ชั่นพิเศษเข้ามา เนื่องจากพอร์ต 3 สามารถนำไปใช้งานในหน้าที่พิเศษได้ทุกสาขา

2.2.7 การใช้งานเป็นพอร์ตอินพุต

เนื่องจากพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเป็นได้ทั้งอินพุตและเอาท์พุท ดังนั้นจึงมีความจำเป็นอย่างยิ่งต้องทำความเข้าใจถึงการกำหนดลักษณะการทำงานให้แก่พอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ในการกำหนดให้เป็นพอร์ตอินพุต ต้องเริ่มต้นด้วยการเขียนข้อมูล " 1 " มาที่แต่ละบิตของพอร์ตที่ต้องการใช้งานเป็นอินพุต เพื่อหยุดการทำงานของเฟตที่ใช้ในการขับสัญญาณเอาท์พุทของบิตนั้นๆ ทำให้ขาสัญญาณของพอร์ตเชื่อมต่อเข้ากับวงจรพูลอัปภายในโดยตรง ส่งผลให้ขาพอร์ตนั้นมีลอจิกเป็น " 1 " สามารถรับสัญญาณลอจิก "0 " จากอุปกรณ์ภายนอกจะถูกส่งเข้ามาแล้วเก็บไว้ในวงจรบัฟเฟอร์ภายในพอร์ต แล้วรอให้ซีพียูมาอ่านค่าเข้าไปเมื่อเป็นเช่นนนี้ อุปกรณ์ภายนอกที่เชื่อมต่อกับพอร์ตอินพุตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชควรกำหนดให้ทำงานในสภาวะลอจิก " 0 " จะดีและสะดวกที่สุด (ซึ่งในปัจจุบันอุปกรณ์อินพุตที่เชื่อมต่อไมโคร คอนโทรลเลอร์แทนทั้งหมดทำงานที่ลอจิก 0 แล้ว)

2.2.8 การใช้งานเป็นพอร์ตเอาท์พุต

โดยปกติแล้ว ขาพอร์ตจะกำหนดให้มีลักษณะเป็นเอาท์พุทอยู่แล้วดังนั้นจึงสามารถส่งข้อมูลออกไปได้อย่างง่ายดายและตรงไปตรงมากล่าวคือ เมื่อต้องการส่งข้อมูล " 0 " ออกไปทางเอาท์พุทก็ให้เขียนข้อมูล " 0 " ไปวงจรแลตช์ ซึ่งจะส่งต่อไปขับเฟต ทำให้เฟตทำงานที่ขาพอร์ตที่กำหนดให้ทำงานก็จะเกิดลอจิก " 0 " ขึ้น ในทางตรงข้ามถ้าหากต้องการส่งข้อมูล " 1 " ออกไป ก็ให้เขียนข้อมูล " 1 " ไปยังวงจรแลตช์วงจรขับก็จะหยุดทำงาน ทำให้ขาพอร์ตที่ต่อกับวงจรพูลอัปภายในเกิดเป็นลอจิก " 1 " ที่ขาพอร์ตนั้น ซึ่งจะคล้ายกับการกำหนดให้เป็นขาอินพุตมาก เพียงแต่แตกต่างกันที่กระบวนการในการเคลื่อนย้ายข้อมูล โดยถ้าเป็นอินพุตจะมีสัญญาณอ่านข้อมูลที่บัฟเฟอร์แต่ถ้าเป็นเอาท์พุทจะไม่มีการอ่านข้อมูลที่บัฟเฟอร์แต่อย่างใด เว้นแต่ในกรณีที่ต้องการตรวจสอบข้อมูลที่ส่งออกมาทางเอาท์พุท

เมื่อใช้งานพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเป็นพอร์ตเอาท์พุท แต่ละขา (หรือแต่ละบิต) ของแต่ละพอร์ตมีความสามารถในการจ่ายกระแสหรือที่เรียกว่า กระแสซอร์ส (Source Current) ได้สูงสุด 10 mA และทุกขารวมกันในแต่ละพอร์ต (ทั้ง 8 บิต) สูงสุด 26 mA สำหรับพอร์ต 0 และพอร์ต 2 ได้สูงสุด 15mA สำหรับพอร์ต 1-3 ในกรณีที่ใช้งานทุกพอร์ตจะสามารถจ่ายกระแสได้รวมกันสูงสุด 71 mA ดังนั้นในการใช้งานเป็นพอร์ตเอาท์พุทเพื่อไม่ให้เกิดปัญหาเกี่ยวกับความสามารถในการจ่ายกระแสจึงควรต่อบัฟเฟอร์ทางเอาท์พุทเพื่อช่วยในการขับกระแสอีกทางหนึ่ง

2.2.9 การอ่านค่าลอจิกจากพอร์ต

ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชสามารถอ่านค่าลอจิกจากพอร์ตได้ 2 ลักษณะ คือ อ่านค่าจากขาพอร์ตโดยตรง และอ่านจากวงจรแลตช์ของแต่ละพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่พอร์ตต่อกับขาเบสทรานซิสเตอร์ ชนิด NPN และขาอิมิตเตอร์ของทรานซิสเตอร์ตัวนั้นต่อลงกราวด์ หากมีการส่งข้อมูล " 1 " ไปยังทรานซิสเตอร์ จะทำให้ทรานซิสเตอร์ทำงานสถานะลอจิกที่ขาพอร์ตเป็น " 0 " เนื่องจากเมื่อทรานซิสเตอร์ทำงาน จะเสมือนว่าพอร์ตนั้นถูกต่อลงกราวด์ทำให้หากอ่านลอจิกที่ขาพอร์ตจะได้ผลตรงข้ามกับที่ส่งออกมา แต่ถ้าหากทำงานอ่านค่าลอจิกที่วงจรแลตช์จะได้ค่าที่ตรงกับค่าที่ต้องการส่งจริง ดังนั้นในการอ่านค่าลอจิกพอร์ตจึงต้องเลือกวิธีการให้เหมาะสมกับอุปกรณ์ที่มาต่อด้วย

2.2.10 จังหวะการทำงานของไมโครคอนโทรลเลอร์ MCS-51

ในการใช้งานไมโครคอนโทรลเลอร์ MCS-51 จะต้องทำความเข้าใจถึงจังหวะการทำงานของซีพียูและลำดับขั้นตอนการประมวลผลคำสั่ง ในการประมวลผลคำสั่งของซีพียูจะมีขั้นตอนหลักๆ 2 ขั้นตอนคือ กระบวน การเฟตช์ (Fetch) เป็นการเรียกคำสั่งออกมาจากหน่วยความจำโปรแกรมแล้วทำการแปลงรหัสคำสั่งนั้นเป็นภาษาเครื่องเพื่อเตรียมการประมวลผล ขั้นตอนต่อมาคือ กระบวนการเอ็กซิคิวต์

กระบวนการเอ็กซิคิวต์ (Execute) เป็นการกระทำตามคำสั่งที่กำหนดหรือตามที่เฟตช์ขึ้นมา โดยกระบวนการก่อนหน้านี้ เมื่อทำการเอ็กซิคิวต์คำสั่งเรียบร้อยแล้ว ก็จะไปเริ่มกระบวนการเฟตช์คำสั่งใหม่ต่อไป

เมื่อเริ่มจ่ายไฟให้แก่ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช จะเกิดการรีเซตในลักษณะที่เรียกว่า เพาเวอร์ออนรีเซต (power-on reset) ทำให้ซีพียูไปเริ่มต้นทำงานที่แอดเดรส 0000h ของหน่วยความจำโปรแกรมจังหวะการทำงานของซีพียูจะเป็นไปตามรูปแบบโดยได้รับการกำหนดมาจากรอบการทำงานของไมโครคอนโทรล เลอร์ MCS-51 โดยใน 1 รอบการทำงานหรือแมชชีนไซเคิลจะแบ่งย่อยออกเป็น 6 สเตต (state) กำหนดให้ชื่อเป็น S1-S6 ในแต่ละสเตตจะมีค่าเวลาเท่ากับ 2 คาบเวลาของสัญญาณนาฬิกา ถ้าสัญญาณนาฬิกามีความถี่ 12 MHz จะมีคาบเวลาเท่ากับ 1 ms คาบเวลาทั้งสองภายในหนึ่งสเตตจะเรียกว่า เฟส 1 (Phase 1) และเฟส 2 (Phase 2)

การเอ็กคิวช์คำสั่งที่ใช้เวลา 1 ไซเคิลเริ่มต้นที่สเตต 1 เป็นการอ่านออปโค้ต อันเป็นการแลตช์ค่าของออปโค้ตส่งไปให้รีจิสเตอร์คำสั่ง (Instruction Register : IR) การเฟตช์ครั้งที่สองจะเกิดขึ้นที่สเตตที่ 4 ภายในแมชชีนไซเคิลเดียวกัน ในกรณีที่เป็นคำสั่งไบต์เดียว การเฟตช์ครั้งที่สองภายในแมชชีนไซเคิลจะถูกตัดทิ้งไป ในคำสั่งที่มีใช้เวลา 1 ไซเคิลจะสิ้นสุดการทำงานลงในสเตตที่ 6 ของแมชชีนไซเคิลเดียวกัน

ความเร็วในการทำงานภายในของไมโครคอนโทรลเลอร์เท่ากับ

ความถี่ของสัญญาณนาฬิกา (ค่าของคริสตอลที่ติดอยู่กับขา XTAL 1 และ XTAL 2)/12

เวลา 1 แมชชีนไซเคิล = 1/ ความเร็วในการทำงานภายในของไมโครคอนโทรลเลอร์

2.2.11 ไทเมอร์โหมดรีจิสเตอร์ (TMOD)

ตัวรีจิสเตอร์ TMOD มีรีจิสเตอร์ควบคุมไทเมอร์ (Timer) จะแบ่งออกเป็น 2 กลุ่ม กลุ่มละ 4 บิต โดย 4 บิตบนจะเป็นตัวควบคุมไทเมอร์ 1ส่วน 4 บิตล่างจะเป็นการควบคุมไทเมอร์ 0 ความหมายของแต่ละบิต ดูในตารางที่ 2.3 ซึ่งตัวรีจิสเตอร์นี้เป็นตัวเลือกการทำงานว่าจะให้ตัว ไทเมอร์/เคาเตอร์ (Counter)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำงานในโหมดใด และเป็นไทเมอร์หรือเคาทเตอร์ รีจิสเตอร์ทีคอน (TCON) ไม่สามารถจะโปรแกรมเข้าไปในระดับบิตได้ (Not Bit-Addressable) ซึ่งการใช้งานมักจะโปรแกรมเข้าไปครั้งเดียวในตำแหน่งเริ่มต้นของโปรแกรม

บิต	ชื่อ	Timer	ความหมาย
7	GATE	1	Gate bit ถ้าบิตนี้เซตวงจรจะทำงานเมื่อ INT1 เป็น High
6	C/T	1	เป็นบิตเลือก Counter/Timer 1 = ใช้เป็น Counter 0 = ใช้เป็น Timer
5	M1	1	Mode bit 1 (ดูตาราง 5-3)
4	M0	1	Mode bit 0 (ดูตาราง5-3)
3	GATE	0	บิต Gate ของ Timer 0
2	C/T	0	บิตเลือก Counter/Timer ของ Timer 0
1	M1	0	Timer 0 M 1 bit
0	M0	0	Timer 0 M 0 bit

ตารางที่ 2-3 รีจิสเตอร์ TMOD (Timer Mode)

M1	M0	Mode	ความหมาย
0	0	0	ใช้เป็น Timer แบบ 13-bit (8048 Mode)
0	1	1	ใช้เป็น Timer แบบ 16-bit
1	0	2	ใช้เป็น Timer แบบ 8-bit Auto-reload Mode
1	1	3	Split Timer Mode : แยก Timer 0 ออกเป็น Timer 8 บิตสองตัวคือ TL0 และ TH0 โดยไม่ใช้ Timer 1

ตารางที่ 2-4 การใช้ Timer โหมดต่างๆ

2.2.12 ไทเมอร์คอนโทรลรีจิสเตอร์ (TCON)

รีจิสเตอร์ทีคอนเป็นรีจิสเตอร์ที่บอกสถานะและควบคุมบิตไทเมอร์ 0 และไทเมอร์ 1 ซึ่งดูได้จากตารางที่ 2-5 รีจิสเตอร์นี้สามารถเข้าถึงข้อมูลระดับบิตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต	ชื่อ	ตำแหน่งบิต	ความหมาย
TCON.7	TF1	8FH	บิตแฟลกแสดงการโอเวอร์โฟลว์ของ Timer1 จะ Set โดย Hardware และ Clear โดย Software
TCON.6	TR1	8EH	บิตควบคุมการปิด-เปิด Timer 1 Set และ Clear โดย Software
TCON.5	TF0	8DH	แฟลกแสดงการ โอเวอร์โฟลว์ของ Timer 0
TCON.4	TR0	8CH	บิตควบคุมการปิด-เปิด Timer 0
TCON.3	IE1	8BH	บิตแฟลกแสดงการอินเทอร์รัพจาก INT1 จะ Set โดย Hardware และสามารถ Clear ได้ด้วย Software
TCON.2	IT1	8AH	บิตเลือกชนิดของสัญญาณอินเทอร์รัพจากอินเทอร์รัพภายนอก INT1 จะสามารถ Set และ Clear ได้ด้วย Software
TCON.1	IE0	89H	บิตแฟลกแสดงการอินเทอร์รัพจาก INT0
TCON.0	IT0	88H	บิตเลือกชนิดของสัญญาณอินเทอร์รัพจากอินเทอร์รัพภายนอก INT0

ตารางที่ 2-5 แสดงความหมายแต่ละบิตของรีจิสเตอร์ TCON (Timer Control)

2.2.13 ไทเมอร์โหมดและโอเวอร์โฟล์แฟลก

เมื่อใช้ไทเมอร์ 0 และไทเมอร์ 1 จะต้องใช้รีจิสเตอร์คู่ TLx และ THx โดยค่า x จะเป็นตัวบอกว่าเป็นไทเมอร์ 0 หรือไทเมอร์ 1 การใช้ไทเมอร์สามารถใช้งานได้หลายโหมด ดังแสดงในรูปที่ 2-11 ซึ่งเราสามารถเซตค่าโหมดการทำงานได้ โดยการโปรแกรมในรีจิสเตอร์ TMOD

2.2.13.1 13-บิต ไทเมอร์โหมด (Mode 0)

การทำงานในโหมด 0 นี้จะเป็นการใช้ไทเมอร์แบบ 13 บิต ดังแสดงในรูปที่ 2.12 (ก) ซึ่งจะใช้ 5 บิตล่างของ TLxโดยไม่สนใจ 3 บิตที่เหลือ และ 8 บิตของ THx การทำงานในโหมดนี้ เมื่อบิตของ TLx นับไปจนเป็น "1" ทุกบิตจะส่ง Clock 1 ลูกให้ THx นับต่อและเมื่อนับเป็น "1" ทุกบิต และเปลี่ยนกลับเป็น "0" จะเกิดโอเวอร์โฟล์แฟลกเกิดขึ้น

2.2.13.2 16-บิต ไทเมอร์โหมด (Mode 1)

การทำงานในโหมดนี้จะเหมือนกับการทำงานในโหมด 0 แต่เป็นไทเมอร์แบบ 16 บิต ซึ่งการนับจะเริ่มตั้งแต่ 0000H, 0001H, 0002H, ไปเรื่อยๆและจะเกิด Overflow ขึ้น เมื่อมีการเปลี่ยนจาก FFFFH เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0000H ดังรูปที่ 2.12 (ข) ซึ่งเป็นการเซตโอเวอร์โฟล์แฟลก และค่านี้จะเกิดขึ้นในบิต TFx ของรีจิสเตอร์ TCON ซึ่งสามารถอ่านและเขียนด้วยโปรแกรม

การใช้ตัวไทเมอร์นี้ค่าของบิตสูงสุด (MSB) คือค่าบิต 7 ของ THx ส่วนบิตต่ำสุด(LSB) คือ บิต 0 ของ TLx บิต LSB จะเป็น Toggles เมื่อมีสัญญาณอินพุตเข้ามา ถูกหารด้วย 2 ดังนั้นจะพบว่าบิต MSB จะ Toggles ด้วยค่าความถี่ของสัญญาณอินพุตหารด้วย 65,536 (2^{16}) และค่า Timer รีจิสเตอร์นี้ (TLx/THx) สามารถอ่านและเขียนได้ด้วยการ โปรแกรม ดังนั้นสามารถนำไปประยุกต์ใช้งานได้ตามต้องการ

2.2.13.3 8 บิต ออโต้-รีโหลดโหมด (Mode 2)

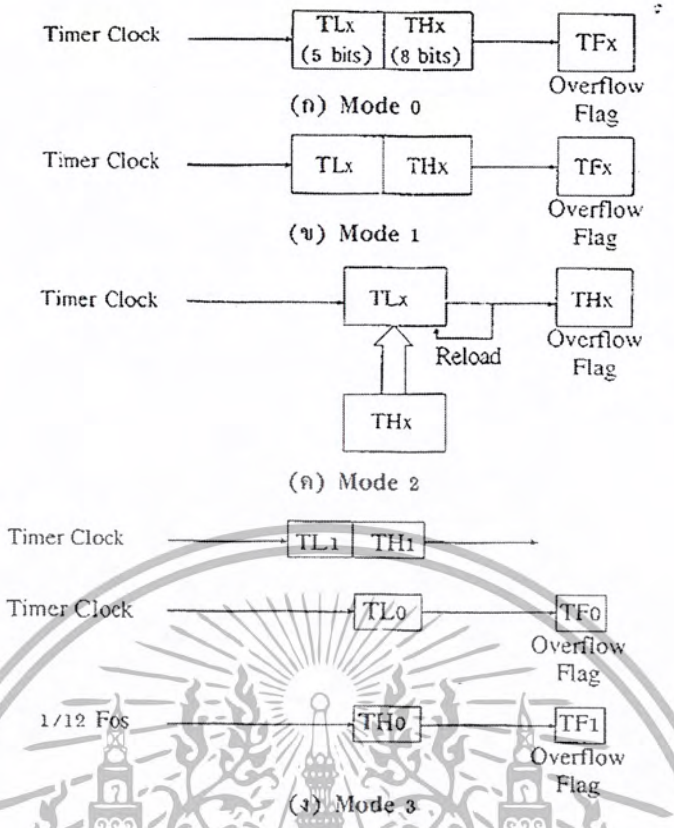
การทำงานในโหมด 2 เรียกอีกอย่างหนึ่งว่า 8-bit Auto-reload Mode โดยใช้ Timer ไบต์ต่ำ (TLx) เป็น Timer แบบ 8 บิต เมื่อไบต์ต่ำเกิดโอเวอร์โฟล์ หรือเกิดการเปลี่ยนแปลงจาก FFH เป็น 00H จะมีการโหลดค่าที่เก็บไว้ในไบต์สูง (THx) ไปเก็บไว้ในไบต์ต่ำ (TLx) ซึ่งจะเป็นค่าเริ่มต้นของการนับครั้งต่อไป นิยมใช้สร้างเป็นฐานเวลาที่สามารถโปรแกรมได้ การทำงานในโหมดนี้แสดงดังรูปที่ 2.12 (ค)

2.2.13.4 สปริตไทเมอร์โหมด (Mode 3)

การทำงานในโหมด 3 นี้ Timer 1 จะไม่ทำงาน ตัว Timer 0 จะแยกเป็น 2 ตัว ตัวละ 8 บิต คือ TL0 และ TH0 เกิดโอเวอร์โฟล์ จะมีการเซตบิต TF0 และ TF1 ดังแสดงในรูปที่ 2.12(ง)

การทำงานในโหมด 3 นี้ ไทเมอร์ 1 จะไม่ถูกใช้งานแต่เราสามารถสวิตช์ให้ ไทเมอร์ 1 ไปทำงานในโหมดอื่นได้ แต่การทำงานของไทเมอร์ 1 จะไม่มีการอินเทอร์รัพเกิดขึ้น เพราะบิต TF1 ถูกใช้ในการนับของ TH0 ในการทำงานของโหมด 3 ไปแล้ว เราอาจมองว่าถ้าให้ไทเมอร์ ทำงานในโหมด 3 ทำให้เรามีไทเมอร์เพิ่มขึ้น คือ TH0 และ TL0 ในไทเมอร์ 0 โหมด 3 และโปรแกรมให้ไทเมอร์ 1 ไปทำงานในโหมดอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-12 แสดงการทำงานของไทเมอร์ในโหมดต่าง ๆ

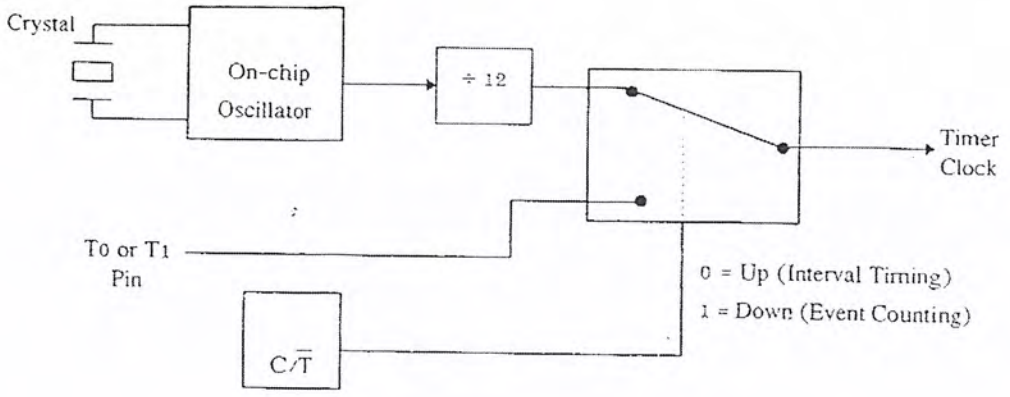
2.2.14 คล็อคกิงซอร์ส (Clocking Source)

ในรูปที่ 2-12 ไม่ได้แสดงว่าไทเมอร์คล็อค (Timer Clock) นำมาจากที่ใดซึ่งการใช้ไทเมอร์ นี้สามารถใช้ได้ 2 หน้าที่ คือเป็นตัวจับเวลา (Timer) และเป็นตัวนับ (Counter) ซึ่งสามารถโปรแกรมได้โดยการเซตหรือรีเซตบิต C/T ในรีจิสเตอร์ TMOD

2.2.14.1 การใช้เป็นตัวจับเวลา (Timer)

ถ้าบิต C/T ใน TMOD เป็นลอจิก "0" จะเป็นการเลือกให้ไทเมอร์นำคล็อคมาจากวงจรออสซิเลเตอร์ (Oscillator) ในชิพ ซึ่งสัญญาณนาฬิกาจะเข้ามาทุกๆ แมชชินไซเคิล (Machine Cycle) หรืออาจกล่าวได้ว่าค่าใน THx และ TLx จะมีค่าเพิ่มขึ้นด้วยอัตราการนับแต่ละครั้งใช้เวลาเท่ากับ1/12 ของความถี่ของสัญญาณนาฬิกาที่ใช้บนชิพ ดังแสดงในรูปที่ 2.13 ถ้า MCS – 51 ใช้สัญญาณนาฬิกา 12 MHz การนับจะมีความถี่เท่ากับ 1 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-13 แสดงความถี่ของสัญญาณนาฬิกาที่จะเข้าไทเมอร์

2.2.14.2 การใช้เป็นตัวนับ (Counter)

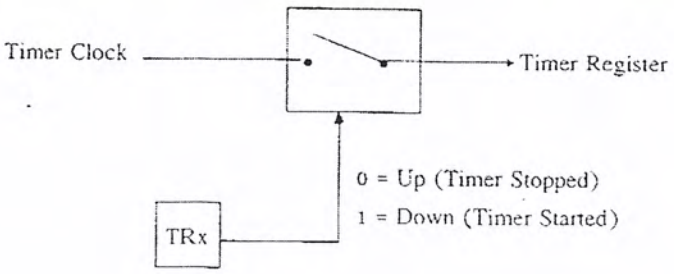
ถ้าบิต C/T เป็น "1" ตัวไทเมอร์จะนำคล็อค มาจากภายนอกโดยใช้ขา P3.4 หรือ T0 เป็นขาอินพุตคล็อค ให้กับไทเมอร์ 0 และใช้ขา P3.5 หรือ T1 เป็นอินพุตคล็อคให้กับไทเมอร์ 1 ดังรูปที่ 2-12 หรืออาจมองว่า ถ้าจะให้นับอะไรสัญญาณที่จะนับให้ต่อกับขา T0 และ T1 ในการใช้เป็นเคาทเตอร์ สัญญาณที่เข้ามามีการเปลี่ยนแปลงจาก "1" เป็น "0" จะทำให้วงจรนับ TLx มีค่าเพิ่มขึ้น 1 ภายใน MCS – 51 นี้ จะตรวจสอบขาอินพุต T0 และ T1 ในช่วงเวลาเฟส 2 ของ State 5 (S5P2) ถ้าพบว่ามีค่าเป็น "1" ต่อมาในอีกหนึ่งแมชชีนไซเคิลที่เฟส 2 ของสเตท 5 (S5P2) ลอจิกอินพุตเปลี่ยนเป็น "0" จะทำให้ค่าในไทเมอร์เพิ่มขึ้น 1 ดังนั้น จะเห็นได้ว่าการนับ 1 ครั้ง จะต้องใช้เวลา 2 แมชชีนไซเคิล ดังนั้นความถี่สูงสุดที่จะให้ไทเมอร์ทำงานเป็นเคาเตอร์ นับได้ จะมีค่ามากที่สุด 500 kHz ถ้า MCS – 51 ทำงานที่ความถี่สัญญาณนาฬิกา 12 MHz

2.2.14.3 การเริ่ม หยุด และการควบคุมไทเมอร์

ในรูปที่ 2.12 จะแสดงลักษณะของไทเมอร์รีจิสตอร์ซึ่งจะเห็นว่าประกอบด้วย TLx และ THx และเมื่อเกิด Overflow จะเกิดเอาต์พุตที่บิต TFx สำหรับสัญญาณนาฬิกาที่จะเข้าไปในไทเมอร์ จะมาจาก 2 ส่วนดังแสดงในรูปที่ 2.,13 ต่อไปจะกล่าวถึงว่าเราจะควบคุมให้เริ่ม, หยุดตัวไทเมอร์ ได้อย่างไร

วิธีเริ่มและหยุดตัวไทเมอร์ สามารถควบคุมได้ที่บิต TRx ในรีจิสเตอร์ TCON โดยปกติแล้ว TRx จะเคลียร์หลังจากที่ระบบถูกรีเซต ซึ่งจะเป็นการให้ไทเมอร์ไม่นับและTRx นี้จะเซตได้จากชุดคำสั่ง หรือการโปรแกรม พิจารณารูปที่ 2.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-14 แสดงการใช้บิตควบคุม TR

ตัวบิต TRx จะเป็นส่วนที่สามารถเข้าถึงข้อมูลในระดับบิตได้ (Bit Addressable) ในรีจิสเตอร์ TCON ถ้าจะให้ไทเมอร์ 0 เริ่มทำงานจะเขียนคำสั่งได้ดังนี้

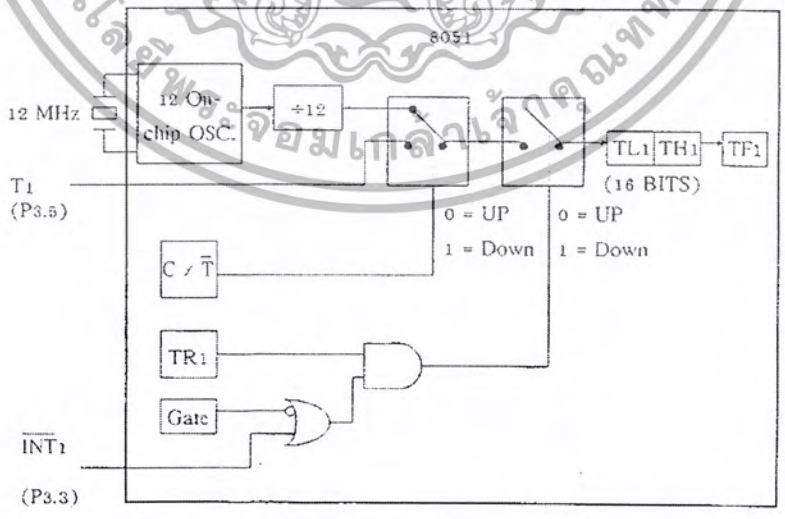
SETB TR0

ถ้าจะหยุดทำงานเขียนคำสั่งได้ดังนี้

CLRB TR0

ในการเขียนโปรแกรมภาษาแอสแซมบลี สามารถใช้สัญลักษณ์ TR0 ในคำสั่ง SETB TR0 เลยได้ เพราะตัวแอสแซมบลีจะตีความ TR0 เป็น Bit Address ตำแหน่ง 8CH

วิธีควบคุมไทเมอร์ สามารถควบคุมได้ที่บิต GATE ใน TMOD และขาอินเทอร์รัพจากภายนอก INTx ถ้า INT0 เป็นลอจิก "0" และโปรแกรมให้ Timer 0 ทำงานในโหมด 2 เมื่อ TL0 / TH0 = 0000H, GATE = 1 และ TR0 = 1 เมื่อ INT0 ขึ้นเป็นลอจิก "1" ตัวไทเมอร์ จะ "Gate On" และจะให้สัญญาณนาฬิกาความถี่ 1 MHz เมื่อ INT0 ลงเป็น "0" ตัว Timer "Gate Off" สัญญาณที่ได้จะมีความกว้างของสัญญาณนาฬิกา 1 ไมโครวินาที ส่งเข้าไปใน TL0 / TH0



รูปที่ 2-15 แสดงระบบทั้งหมดของไทเมอร์ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2-15 จะเป็นระบบที่สมบูรณ์ของไทเมอร์ 1 เมื่อทำงานในโหมด 1 ซึ่งเป็น ไทเมอร์ 16 บิต โดยใช้รีจิสเตอร์ TL1 / TH1 และ Overflow Flag TF1 ในรูปจะเห็นถึงการควบคุมแหล่งกำเนิดคล็อค การเริ่มทำงานและการหยุดทำงาน

2.3 การใช้งานพอร์ตสื่อสารอนุกรม

2.3.1 มาตรฐานพอร์ตอนุกรมแบบ RS-232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อสารผ่านสายโทรศัพท์ไปยังคอมพิว เตอร์อีกชุดซึ่งอยู่ห่างไกลกัน โดยคณะกรรมการที่เรียกว่าสมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3 ถึง -12V แสดงว่ามีข้อมูล (Mark) และ +3 ถึง +12 V แสดงว่าเป็นช่องว่าง (Space)

มาตรฐาน RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) กับวงจรข้อมูลปลายทาง (Data Circuit Terminating : DCE) ไว้ว่า อุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรลเลอร์หรือไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้งสองจะกระทำผ่านมาตรฐาน RS-232

ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งที่เห็นได้ชัดคือ คอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ที่โมเด็มจะเป็นแบบ DCE

สำหรับการใช้งานบนคอมพิวเตอร์ พอร์ตอนุกรม RS-232 มักถูกใช้เชื่อมต่อกับโมเด็มหรือเมาส์ โดยสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุดถึง 20 เมตร

2.3.2 คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้คอนเน็กเตอร์แบบ DB-25 ตัวผู้หรือ DB-9 ตัวผู้ ซึ่งคอนเน็กเตอร์แบบ DB-25 จะมีขาต่อใช้งานเพียง 9 เส้นเช่นเดียวกับคอนเน็กเตอร์แบบ DB-9 เนื่องจากขาอื่นๆ ที่เคยใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนัก จึงถูกยกเลิกไป โดยแสดงรูปร่างและตำแหน่งขาในรูปที่ 2.16

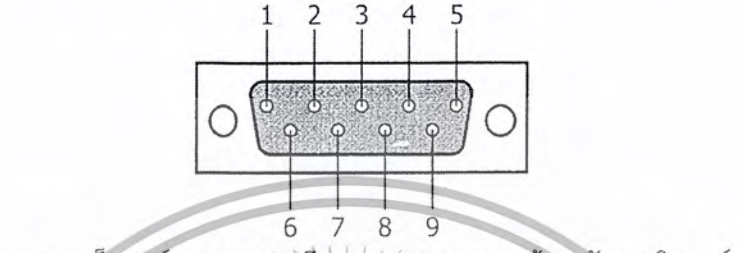
สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังในรูปที่ 2-17 ลูกศรในรูปแสดงถึงทิศทางของข้อมูล ในรูปที่ 2-16 (ก) เป็นการเชื่อมต่อแบบ Null modem หรือการเชื่อมต่อโดยตรงโดยไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบหรือแฮนด์เช็กเต็มรูปแบบ ส่วนในรูปที่ 2-16 (ข) เป็นการเชื่อมต่อแบบ Null modem ในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นสำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

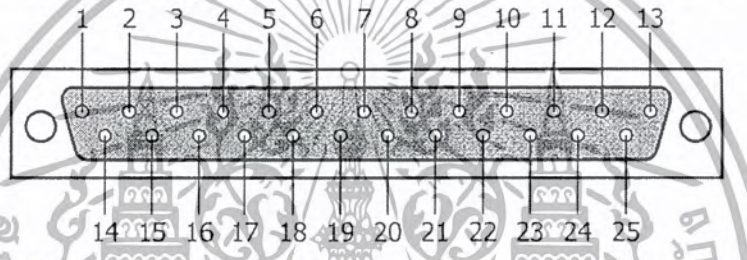
รับข้อมูล และเส้นสุดท้ายเป็นกราวด์ สำหรับรายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232 มีดังนี้

- **Data Carrier Detect : DCD** หรืออาจเรียกว่า Carrier Detect :CD ขานี้จะแอกตีฟเมื่อมีการส่งสัญญาณพาห์จากอุปกรณ์สื่อสารข้อมูลเช่น โมเด็ม สำหรับการใช้งานปกติ ขานี้จะไม่ได้ถูกใช้งานมากนัก

- **Receive Data : RD** หรือ **RxD** ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์ โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์ บัฟเฟอร์



(ก) คอนเน็กเตอร์อนุกรม 9 ขาหรือแบบ DB-9 (มองจากด้านหลังคอมพิวเตอร์)

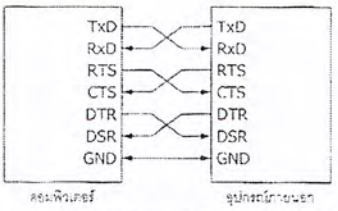


(ข) คอนเน็กเตอร์อนุกรม 25 ขาหรือแบบ DB-25 (มองจากด้านหลังคอมพิวเตอร์)

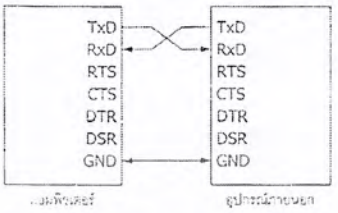
คอนเน็กเตอร์ DB-9	คอนเน็กเตอร์ DB-25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
1	8	Data Carrier Detect :DCD	อินพุต
2	3	Received Data : RxD	อินพุต
3	2	Transmitted Data : TxD	เอาต์พุต
4	20	Data Terminal Ready :DTR	เอาต์พุต
5	7	Signal Ground : GND	-
6	6	Data Set Ready : DSR	อินพุต
7	4	Request To Send : RTS	เอาต์พุต
8	5	Clear To Send : CTS	อินพุต
9	22	Ring Indicator : RI	อินพุต

รูปที่ 2-16 แสดงการจัดขาของคอนเน็กเตอร์พอร์ตอนุกรมตามมาตรฐาน RS-232 ทั้งแบบ DB-9 และ DB-25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ แบบ Null modem



(ข) การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ แบบ RS-232 โดยใช้สายสัญญาณเพียง 3 เส้น

รูปที่ 2-17 แสดงการต่อกับอุปกรณ์ภายนอกในลักษณะต่าง ๆ

- **Transmitted Data : TD** หรือ **TxD** ใช้ส่งข้อมูลออกจากคอมพิวเตอร์ โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป

- **Data Terminal Ready : DTR** เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่า ต้องการติดต่อด้วย โดยขา DTR นี้ต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง และขา DTR ของอุปกรณ์ปลายทางต้องเชื่อมต่อกับขา DSR ของคอมพิวเตอร์ ถ้าใช้การเชื่อมต่อแบบ Null Modem ซึ่งใช้สายในการเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกัน และต้องต่อกับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาห์

- **Signal Ground : GND** กราวด์ระบบ

- **Data Set Ready : DSR** ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DSR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขา DTR

- **Request To Send : RTS** เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบ Null modem 3 สาย จะต้องเชื่อมต่อขา RTS และ CTS ของตัวมันเองเข้าด้วยกัน เพื่อจะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา

- **Clear To Send : CTS** ขานี้จะคอยรับสัญญาณจากขา RTS เมื่อรับสัญญาณได้ ข้อมูลที่ขา TxD จะถูกส่งออกไป ดังนั้นขานี้จึงถูกใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมที่จะรับข้อมูลหรือไม่

- **Ring Indicator : RI** ใช้แสดงสถานะสัญญาณเรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็มและโปรแกรมมีการตรวจสอบสัญญาณนี้เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 ยูอาร์ต (UART)

ยูอาร์ตมาจากคำว่า Universal Asynchronous Receiver Transmitter ซึ่งหมายถึงอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสนั่นเอง สำหรับการสื่อสารอนุกรมบนคอมพิวเตอร์แล้วยูอาร์ต ถือว่าเป็นหัวใจสำคัญของการสื่อสารอนุกรม

หน้าที่หลักของยูอาร์ต คือทำหน้าที่แปลงข้อมูลที่อยู่ในรูปแบบขนานจากคอมพิวเตอร์ให้อยู่ในรูปแบบอนุกรมแบบอะซิงโครนัส แล้วส่งออกไป และทำหน้าที่แปลงสัญญาณอนุกรมแบบอะซิงโครนัสที่ป้อนเข้ามายัง ยูอาร์ตให้เป็นแบบขนานก่อนที่จะส่งเข้าสู่คอมพิวเตอร์ ซึ่งนอกจาก ยูอาร์ตจะส่งข้อมูลไปยังคอมพิวเตอร์แล้ว ยังแจ้งข้อมูลอื่นๆให้คอมพิวเตอร์รับทราบด้วย เช่น อัตราเร็วในการรับส่งข้อมูล (บอดเรต) , รูปแบบการส่งข้อมูล, ความผิดพลาดที่เกิดขึ้นระหว่างการถ่านทอดข้อมูล (ผิดพลาดจากพาริตี้, เฟรมข้อมูล,โอเวอร์รัน) เป็นต้น

ภายในยูอาร์ตจะมีส่วนของวงจรสร้างบอดเรตแบบโปรแกรมได้ (programmable buadrate generator) โดยการกำหนดค่าตัวหารให้กับสัญญาณนาฬิกาของ ยูอาร์ต โดยตัวหารนี้มีขนาด 16 บิต ดังนั้นจึงสามารถกำหนดตัวหารอยู่ในช่วง 1-65,535 ยูอาร์ตสามารถรับส่งข้อมูลได้ทั้งแบบฮาล์ฟดูเพล็กซ์ (half duplex) และฟูลดูเพล็กซ์ (full duplex) โดยการส่งแบบฮาล์ฟดูเพล็กซ์เป็นการส่งแบบทิศทางเดียว ส่วนการส่งแบบฟูลดูเพล็กซ์นั้นสามารถรับและส่งข้อมูลได้ในคราวเดียวกัน

ชนิดของ ยูอาร์ต

ในเครื่องคอมพิวเตอร์ทั่วไปมี ยูอาร์ต ที่ใช้งานกันอยู่ 2 เบอร์คือ 8250 ซึ่งเป็น ยูอาร์ต มาตรฐานที่มีใช้กันมายาวนาน ยูอาร์ต เบอร์นี้จะมีบัฟเฟอร์สำหรับรับและส่งข้อมูลตำแหน่งเดียวกัน ทำให้การรับและส่งข้อมูลถูกจำกัดความเร็วอยู่ที่ 57.6 กิโลบิตต่อวินาทีเท่านั้น แต่ ยูอาร์ต เบอร์นี้ก็ถือว่าเป็นต้นแบบของยูอาร์ตที่ใช้ในคอมพิวเตอร์ โดยคอมพิวเตอร์ทุกๆรุ่นจะต้องสนับสนุนการทำงานตามรูปแบบของ ยูอาร์ต เบอร์นี้

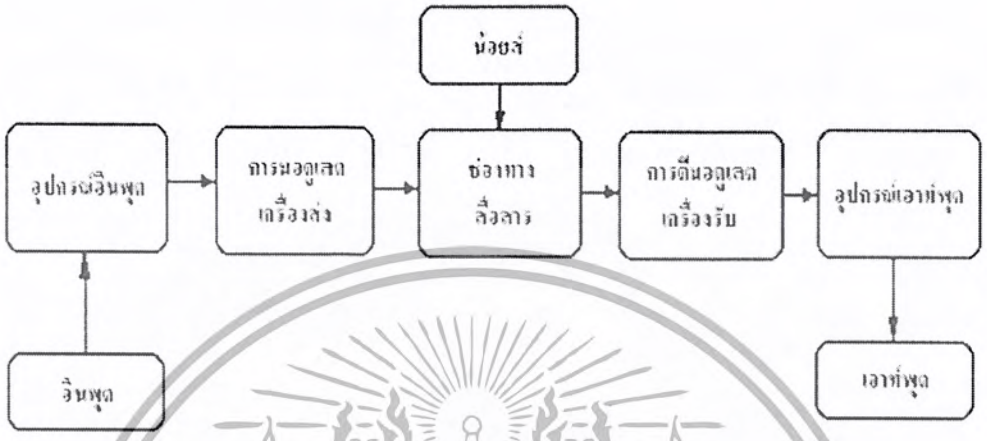
ยูอาร์ต อีกเบอร์หนึ่งคือ 16450 มีความสามารถรับส่งข้อมูลได้ที่ความเร็ว 115,200 บิตต่อวินาที และเพิ่มรีจิสเตอร์สำหรับพักข้อมูลสำหรับ ยูอาร์ต นอกจากนั้นยังเพิ่มส่วนของชิฟต์รีจิสเตอร์แบบ FIFO (First In First Out) ขนาด 16 ไบต์เข้าไป ทำให้สามารถสนับสนุนความเร็วในการรับส่งข้อมูลที่ 256 กิโลบิตต่อวินาทีได้ โดยคอมพิวเตอร์ในปัจจุบันใช้ ยูอาร์ต เบอร์นี้หรือใหม่กว่า เช่น เบอร์ TL16C750 ซึ่งมีรีจิสเตอร์แบบ FIFO ขนาด 64 ไบต์ ทำงานได้ที่ระดับแรงดัน +5V และ +3V มีโหมดประหยัดพลังงาน สามารถรับส่งข้อมูลได้ที่ระดับความเร็ว 1 เมกะบิตต่อวินาที เมื่อใช้สัญญาณนาฬิกา 16 MHz

อย่างไรก็ตาม ความเร็วในการส่งข้อมูลที่มากมายของ ยูอาร์ต เบอร์ใหม่ๆ ก็ไม่ได้ช่วยให้การรับส่งข้อมูลของคอมพิวเตอร์เร็วขึ้น เนื่องจากว่าคอมพิวเตอร์ยังใช้ความถี่ของสัญญาณนาฬิกาในการแปลงข้อมูลเพียง 1.8432 MHz เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การรับ-ส่งข้อมูลดิจิตอลผ่านคลื่นวิทยุ

ในระบบสื่อสารไม่ว่าจะเป็นระบบใดก็ตาม โดยพื้นฐานจะประกอบไปด้วยอุปกรณ์อินพุท (Input device), เครื่องส่ง, ช่องทางสื่อสาร (Communication Channel) ซึ่งมักจะมีนอยส์ (Noise) มารบกวน, เครื่องรับ และอุปกรณ์เอาต์พุต (Output device) แสดงดังรูปที่ 2-18



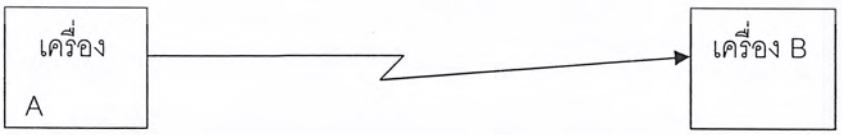
รูปที่ 2-18 แสดงองค์ประกอบระบบสื่อสารพื้นฐาน

ในการสื่อสารแบบแบบดิจิตอลจะใช้ข้อมูลเป็นรหัส "0" หรือ "1" ได้แก่ เลขฐานสอง และเลขฐานสิบหก เป็นต้น โดยการเปลี่ยนสัญญาณอนาล็อก (Analog) เป็นดิจิตอล (Digital) นั้น จะใช้วิธีการสุ่มตัวอย่าง (Sampling) จากสัญญาณอนาล็อกโดยวิธีการทางคณิตศาสตร์ ซึ่งจะได้เป็น เลขฐานสองออกมา

การส่งสัญญาณ

2.4.1 การส่งแบบทิศทางเดียว (Simplex)

หมายถึง การส่งสัญญาณจากด้านส่งไปยังด้านรับเพียงด้านเดียว โดยที่ไม่สามารถโต้ตอบระหว่างกันได้ เช่น การกระจายเสียงของสัญญาณวิทยุ หรือ โทรทัศน์ เป็นต้น

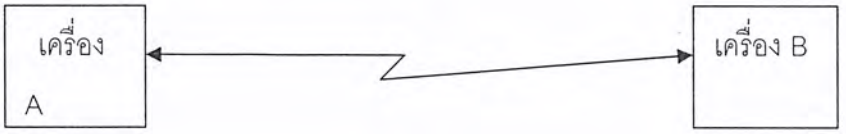


รูปที่ 2-19 แสดงการส่งข้อมูลแบบ Simplex

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 การส่งแบบสองทิศทางแต่ต่างเวลากัน (Half – duplex)

หมายถึง ทั้งทางด้านส่งและด้านรับสามารถโต้ตอบระหว่างกันได้ โดยมีข้อกำหนดว่าต้องมีด้านหนึ่งเป็นด้านรับเสมอ (ผลัดกันส่ง) เช่น วิทยุสมัครเล่น



รูปที่ 2-20 แสดงการส่งข้อมูลแบบ Half – duplex

2.4.3 การส่งแบบสองทิศทางที่เวลาเดียวกัน **(Full – duplex)**

หมายถึง การที่ด้านรับและด้านส่งสามารถส่งสัญญาณได้พร้อมกันในเวลาเดียวกัน โดยที่ไม่ต้องผลัดกันส่งเหมือนแบบ Half –duplex

รูปที่ 2-21 แสดงการส่งข้อมูลแบบ Full - duplex

2.5 การทำงานของเซอร์โวมอเตอร์

2.5.1 หลักการทำงานของเซอร์โวมอเตอร์

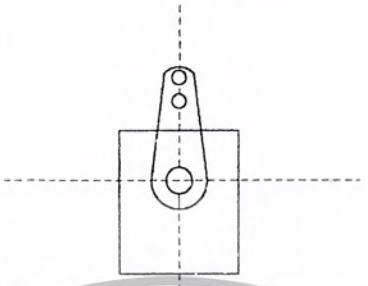
เซอร์โวมอเตอร์ ประกอบด้วย มอเตอร์ความเร็วสูง ภายในมีเฟืองทดรอบ ให้หมุนช้าลง เพื่อจะได้มีกำลังแรงบิดสูงขึ้น นอกจากนี้ยังมีวงจรควบคุมมอเตอร์ ซึ่งวงจรนี้จะนำค่าแรงดันเฉลี่ยของพัลส์รูปสี่เหลี่ยม เข้าไปเปรียบเทียบกับค่าแรงดันค่าหนึ่งที่มีอยู่ในวงจร ถ้าค่าต่างกัน วงจรควบคุมจะสั่งให้มอเตอร์หมุนไปตามทิศทาง ซึ่งขึ้นอยู่กับขนาดความกว้างของพัลส์ โดยที่แกนเฟืองทดรอบจะถูกพ่วงไปขับแกนของ VR จะถูกปรับค่า ทำให้ค่าแรงดันเปรียบเทียบของวงจรควบคุมมอเตอร์เปลี่ยนไปด้วย จนกระทั่งค่าเฉลี่ยของพัลส์ในวงจรควบคุมมอเตอร์ เท่ากับค่าเฉลี่ยของพัลส์ที่เข้ามา จึงทำให้มอเตอร์หยุดหมุนได้

เซอร์โวมอเตอร์ จะมีสายไฟ 3 เส้น คือ สายไฟเลี้ยง สายกราวด์ และสายสัญญาณพัลส์ควบคุมซึ่งลักษณะของสัญญาณ พัลส์ที่ใช้ควบคุมตำแหน่งของเซอร์โวมอเตอร์ จะเป็นการส่งพัลส์ที่มีความกว้างต่างกัน เพื่อควบคุมให้เซอร์โวมอเตอร์ หมุนไปยังตำแหน่งที่ต้องการ โดยที่มีความกว้างของพัลส์จะเป็นตัวกำหนดขนาด และทิศทางของการหมุนแกนเซอร์โวมอเตอร์สำหรับคาบเวลาหรือระยะห่างระหว่างพัลส์แต่ละลูก จะเป็นตัวกำหนดแรงบิดของมอเตอร์

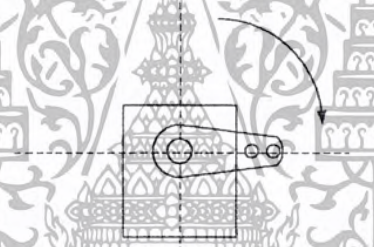
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้ากำหนดให้ในสภาวะปกติ เมื่อป้อนพัลส์สี่เหลี่ยม ที่มีความกว้างขนาด 1.5 ms ให้กับเซอร์โวมอเตอร์ แขนของเซอร์โวมอเตอร์จะอยู่ตำแหน่งกลาง

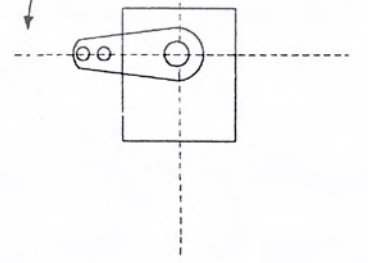
เมื่อป้อนพัลส์สี่เหลี่ยม ที่มีความกว้างขนาด 2.0 ms ให้กับเซอร์โวมอเตอร์ แขนของเซอร์โวมอเตอร์จะหมุนทวนเข็มนาฬิกา ตามที่แสดงไว้ในรูป ดังต่อไปนี้



รูปที่ 2-22 แสดงการตอบสนองของเซอร์โวมอเตอร์เมื่อจ่ายพัลส์ขนาด 1.4 ms



รูปที่ 2-23 แสดงการตอบสนองของเซอร์โวมอเตอร์เมื่อจ่ายพัลส์ขนาด 1 ms



รูปที่ 2-24 แสดงการตอบสนองของเซอร์โวมอเตอร์เมื่อจ่ายพัลส์ขนาด 2 ms

รูปที่ 2-22 ถึง 2-24 แสดงการตอบสนองของเซอร์โวมอเตอร์ต่อสัญญาณพัลส์ในความถี่ที่ต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

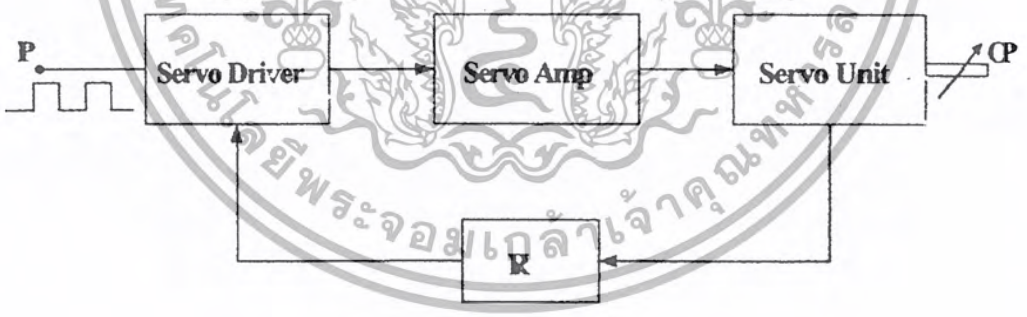
ดังนั้นถ้าจ่ายสัญญาณพัลส์ที่มีความกว้างมากหรือน้อยกว่าความกว้างของพัลส์ 1.5 ms ก็จะทำให้เซอร์โวมอเตอร์หมุนต่างทิศกัน ทั้งตามเข็มและทวนเข็มนาฬิกา โดยตำแหน่งของแขนหมุนเซอร์โวมอเตอร์ จะเบี่ยงเบนออกจากจุดกึ่งกลาง เป็นสัดส่วนกับความกว้างของพัลส์ที่จ่ายให้

2.5.2 การทำงานภายในของเซอร์โวมอเตอร์

ในเซอร์โวมอเตอร์หนึ่งตัวจะประกอบไปด้วย 3 ภาค การทำงานแต่ละภาคมีหน้าที่และการทำงานดังนี้

- ภาคขับเซอร์โว ประกอบด้วย วงจรสร้างสัญญาณพัลส์ และวงจรเปรียบเทียบสัญญาณพัลส์ที่สร้างขึ้น กับสัญญาณพัลส์ I / P ที่รับเข้ามา
- ภาคขยายเซอร์โว ประกอบด้วย วงจร RC Network ที่ช่วยหน่วงสัญญาณให้เซอร์โวสามารถทำงานได้ตลอดช่วงคาบเวลา จนกระทั่งมีสัญญาณลูกต่อไปมา รวมถึงวงจรกลับขั้วแรงดันไฟฟ้าควบคุมทิศทางการหมุนของมอเตอร์
- หน่วยเซอร์โว ประกอบด้วย มอเตอร์ความเร็วสูง เฟืองทดรอบ แกนหมุน อุปกรณ์ต่างๆ และ VR (ตัวต้านทานปรับค่าได้) ทำหน้าที่ป้อนกลับตำแหน่ง (Position Feedback)

ซึ่งในขณะที่มอเตอร์หมุน VR จะถูกปรับค่า Feedback กลับมาปรับและเปรียบเทียบค่าความกว้างของพัลส์ที่ภาคขับเซอร์โว เมื่อขนาดความกว้างของพัลส์ มีค่าเฉลี่ยของค่าแรงดันเท่ากันมอเตอร์จะหยุดหมุนทันที ซึ่งรูปที่ 2-25 ได้แสดงภาคการทำงานของเซอร์โวมอเตอร์ตามที่ได้กล่าวไว้ในข้างต้น และได้แสดงไว้ดังรูปต่อไปนี้



รูปที่ 2-25 แสดงภาคการทำงานของเซอร์โวมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบโครงสร้างหุ่นยนต์

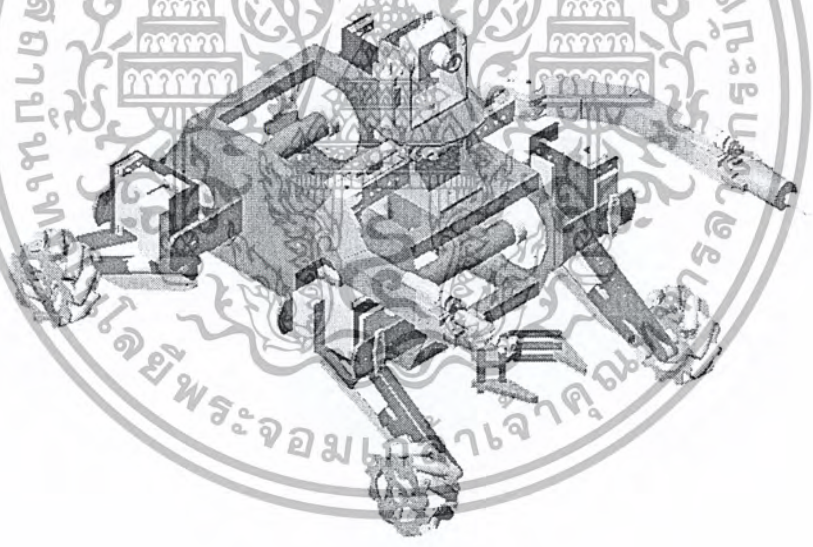
ในบทนี้จะทำการอธิบายลักษณะทางโครงสร้างของ HUOR ตั้งแต่ระบบการเคลื่อนที่ (Robot Movement System) ระบบแขนกล (Robot Arm System) และระบบการมองเห็น (Robot Eye System) ซึ่งในการออกแบบแต่ละส่วนนั้น เราได้ทำการออกแบบจากความต้องการใช้งานเป็นพื้นฐาน ดังนี้

3.1 ระบบการเคลื่อนที่ (Robot Movement System)

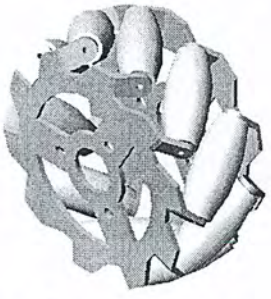
เพื่อให้เกิดความคล่องตัว และสามารถเคลื่อนที่ได้ในทุกสภาพพื้นผิว การเคลื่อนที่ของ HUOR จึงได้รับการออกแบบให้ทำงานได้ 2 โหมด ดังนี้

3.1.1 การเคลื่อนที่ในโหมดล้อทุกทิศทาง (Omni Wheel Movement Mode)

เนื่องจากต้องการให้ HUOR สามารถเคลื่อนที่ได้ทั้ง 8 ทิศทาง และมีความคล่องตัวและสามารถหลบหลีกสิ่งกีดขวางได้ดี จึงเลือกใช้ล้อแบบทุกทิศทาง (Omni Direction Wheel) ดังรูป



รูปที่ 3-1 แสดงการเคลื่อนที่ในโหมดล้อทุกทิศทาง



รูปที่ 3-2 ล้อแบบ Omni

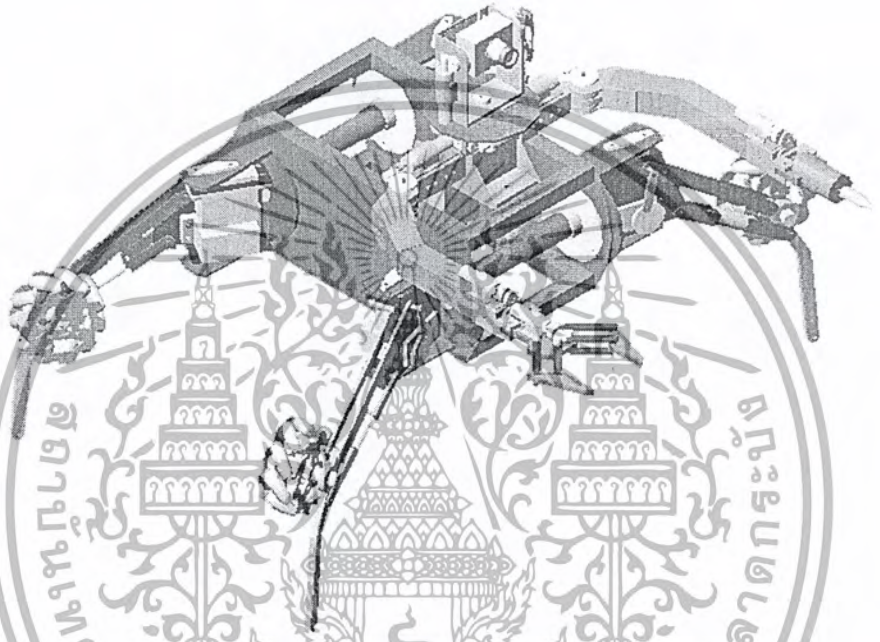
จากรูป เห็นได้ว่าในการเคลื่อนที่ด้วยล้อแบบทุกทิศทางนั้น ล้อทั้ง 4 จะเป็นอิสระต่อกัน เนื่องจากส่วนที่สัมผัสกับพื้นผิวนั้นเป็นลูกกลิ้งที่เอียงทำมุม 45 องศา ทำให้เราสามารถควบคุมการเคลื่อนที่ได้ในทุกทิศทาง ด้วยมอเตอร์ที่ติดอยู่กับล้อทั้ง 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 การเคลื่อนที่ในโหมดการก้าวขา (4-Legs Movement Mode)

การเคลื่อนที่ในสภาพผิวที่ไม่สามารถใช้ล้อได้ HUOR จะทำการเปลี่ยนระบบการเคลื่อนที่ (Change Movement Mode) ไปสู่ระบบการก้าวขา 4 ขา และทำการตรวจสอบรูปแบบตำแหน่งของขาอัตโนมัติก่อนการก้าวเดินทุกครั้ง โดยที่เรา control เพียงแค่ทิศทางเท่านั้น

ในการทำงานของขาแต่ละท่อนนั้น เราใช้ Servo Motor ในการขับเคลื่อน เนื่องจากการก้าวขาแต่ละครั้งนั้นต้องการความแม่นยำสูง ซึ่ง Servo Motor มีความสามารถในการกำหนดการทำงานเป็นองศาได้ แต่มีข้อเสียคือ ราคาแพง



รูปที่ 3-3 แสดงเคลื่อนที่ในโหมดการก้าวขา

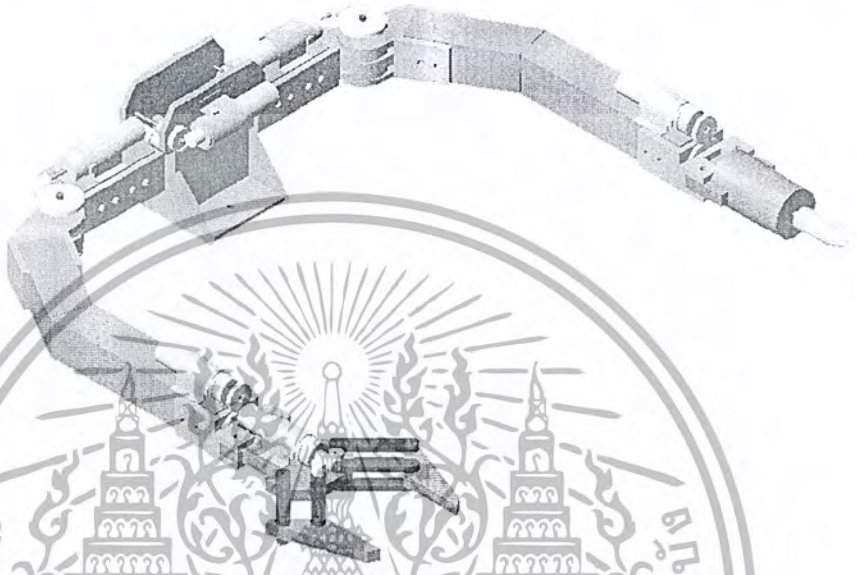
Walker mode **Change mode** **Running mode**

รูปที่ 3-4 แสดงลักษณะของขาในโหมดต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

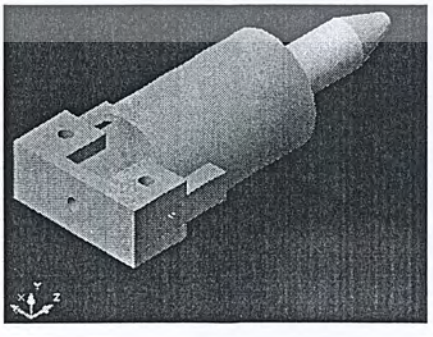
3.2 ระบบแขนกล (Robot Arm System)

ในการออกแบบส่วนนี้นับว่าเป็นส่วนที่ต้องการความละเอียดอ่อนเป็นอย่างมาก เนื่องจากว่าในการที่ HUOR จะสามารถทำงานแทนมนุษย์ได้นั้น แขนกลของ HUOR จะต้องมีความสามารถที่คล้ายคลึงกับการทำงานของแขนมนุษย์ให้มากที่สุดเท่าที่จะเป็นไปได้ แขนทั้ง 2 ของ HUOR ประกอบไปด้วยข้อต่อ (Joint) ที่เป็นจุดหมุนให้แขนสามารถเคลื่อนที่ไปได้ทั้งในแนวระนาบ (แกน X) และ แนวดิ่ง (แกน Y) ดังรูป

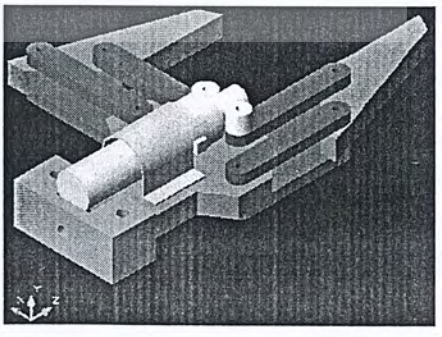


รูปที่ 3-5 แสดงระบบแขนกล

เพื่อให้เกิดความหลากหลายในการทำงานเราได้ออกแบบให้มือทั้ง 2 ข้างของ HUOR นั้นมีลักษณะการใช้งานที่แตกต่างกันคือ มือข้างขวานั้นมีลักษณะเป็นมือจับ 2 ง่ามสำหรับคีบจับสิ่งของต่าง ๆ ในการทำงาน โดยมีแผ่นยางกันลื่นที่ปลายทั้งสอง สำหรับมือข้างซ้ายจะมีลักษณะเป็นสว่าน ซึ่งในส่วนนี้เราสามารถเปลี่ยนแปลงให้เกิดความเหมาะสมในการทำงานได้ อาทิเช่น หากต้องการเจาะรูก็สามารถเปลี่ยนไปใช้ดอกสว่านขนาดที่ต้องการ ได้สามารถเปลี่ยนเป็นใบควง เพื่อใช้ในการขันสกรูได้ นอกจากนี้เรายังสามารถเปลี่ยนเป็นใบมีดจิ๋วเพื่อใช้ตัดสิ่งของได้อีกด้วย แสดงดังรูปด้านล่าง



รูปที่ 3-6 แสดงลักษณะของมือสว่าน

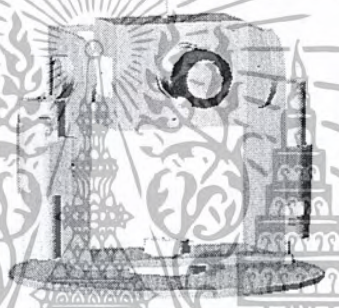


รูปที่ 3-7 แสดงลักษณะของมือจับ 2 ง่าม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

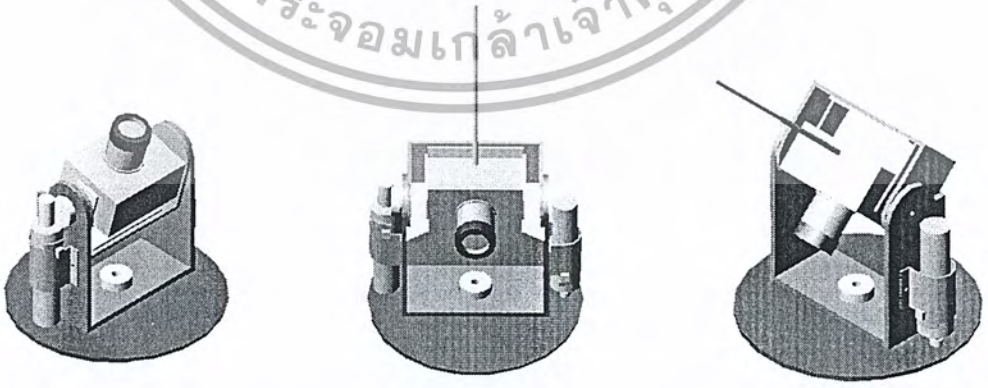
3.3 ระบบการมองเห็น (Robot Eye System)

ในส่วนนี้เราได้นำกล้องวงจรปิดไร้สายเข้ามาช่วยในการมองเห็น เปรียบเสมือนดวงตาของมนุษย์ เนื่องจากว่าในการควบคุมจากระยะไกลนั้น สิ่งที่มีส่วนสำคัญมากที่สุดนั่นก็คือมุมมองและการมองเห็น ซึ่งจะช่วยผู้ควบคุมได้เป็นอย่างมากในการตัดสินใจที่จะสั่งการให้ HUOR ปฏิบัติการอย่างใดอย่างหนึ่ง โดยออกแบบให้สามารถเคลื่อนที่ได้ทั้งในแนวระนาบและแนวดิ่ง เพื่อให้ผู้ควบคุมปรับเปลี่ยนมุมมองให้เหมาะสมในสภาวะนั้น ๆ ได้ นอกจากนี้แล้ว เรายังสามารถบันทึกสัญญาณภาพจาก HUOR เก็บเป็นไฟล์รูปภาพในคอมพิวเตอร์ได้อีกด้วย ซึ่งสิ่งนี้มีความจำเป็นอย่างยิ่งในการปฏิบัติงานจริง



รูปที่ 3-8 แสดงระบบการมองเห็น

รูปด้านล่างได้แสดงให้เห็นว่ามุมมองของ HUOR นั้นมีความละเอียดและคล้ายคลึงกับมนุษย์เป็นอย่างมาก ทั้งนี้เพื่อช่วยให้การควบคุมจากระยะไกลของมนุษย์เป็นไปได้ง่ายขึ้นในทางปฏิบัติ



รูปที่ 3-9 แสดงมุมมองต่าง ๆ ของกล้อง

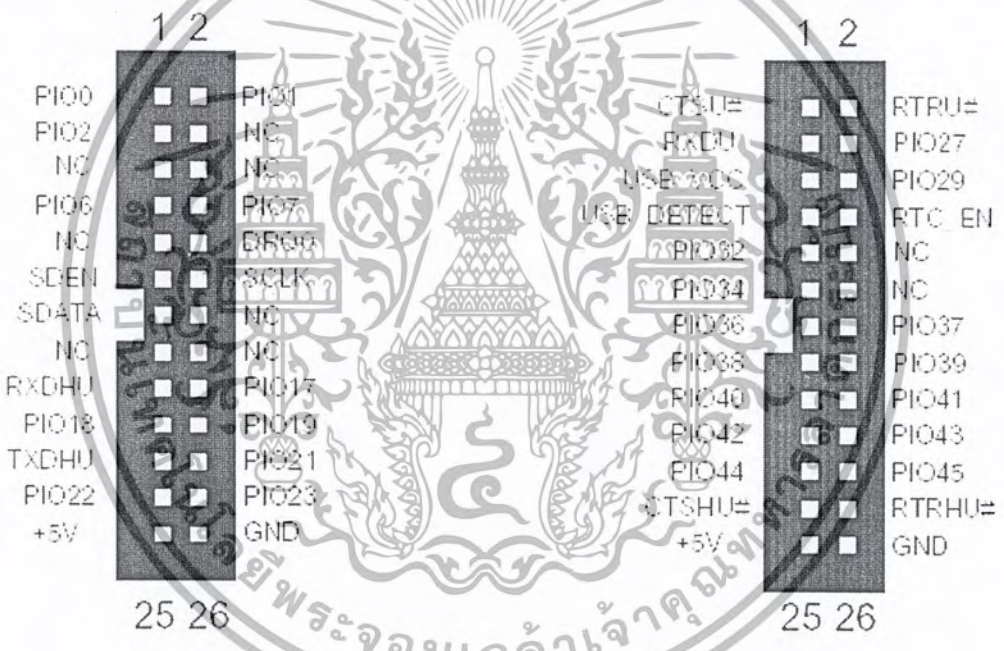
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การออกแบบวงจรควบคุม

4.1 พีไอโอ พอร์ตอินพุท / เอาท์พุทของชุดพัฒนาคอม 86

จากการศึกษาการใช้งานชุดพัฒนาคอม 86 นั้นพบว่ามีพอร์ตอินพุท/เอาท์พุท พีไอโอ(PIO) ซึ่งมีลักษณะการใช้งานเหมือนกับไมโครคอนโทรลเลอร์โดยทั่วไป สามารถเขียนโปรแกรมควบคุมการใช้งานได้เป็นอิสระจากกัน โดยบนชุดพัฒนาคอม 86 นี้มีพีไอโอพอร์ต ให้นักพัฒนาได้ใช้งานทั้งสิ้นจำนวน 25 แชนแนลจากทั้งหมด 48 แชนแนลที่มีในไมโครคอนโทรลเลอร์ เนื่องจากการที่สัญญาณพีไอโอ บางสัญญาณถูกนำไปใช้งานในลักษณะของการมัลติเพล็กกับสัญญาณอื่นของชุดพัฒนาคอม 86 ซึ่งพีไอโอที่ใช้งานได้บนชุดพัฒนาคอม 86 และการเชื่อมต่อมายังคอนเนคเตอร์ของพีไอโอ ต่าง ๆ เป็นไปดังรูป



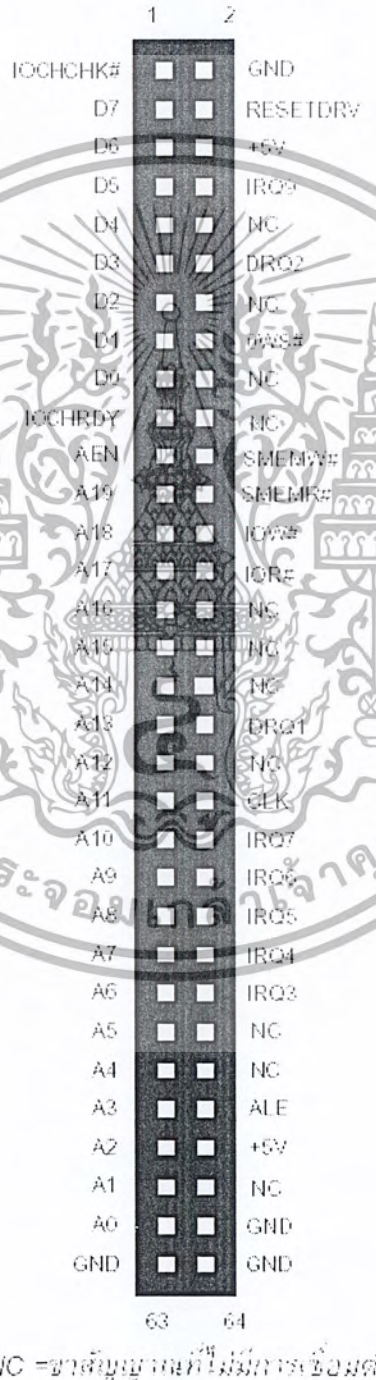
รูปที่ 4-1 แสดงคอนเนคเตอร์ของ PIO บนชุดพัฒนาคอม 86

จากรูปในส่วนที่เป็นชื่อของสัญญาณอื่นที่มิใช่พีไอโอหมายเลขต่าง ๆ นั้นคือพีไอโอที่ถูกใช้งานในรูปแบบของสัญญาณอื่นที่มัลติเพล็กซ์อยู่ และสามารถนำมาใช้งานเป็นพีไอโอได้ในกรณีที่ไม่ต้องการใช้งาน เช่น ขา CTSHU# ถ้าไม่ต้องการใช้งานส่วนของ โฟลว์คอนโทรลของพอร์ตอนุกรมคอม 1 (COM1) ก็สามารถนำมาใช้งานเป็นพีไอโอ 46 ได้ ส่วนขาเอ็นซี (NC) คือขาพีไอโอ ที่ถูกนำไปใช้ในลักษณะของสัญญาณอื่นที่มัลติเพล็กซ์อยู่และไม่สามารถนำมาใช้งานได้ ซึ่งไม่มีการนำมาเชื่อมต่อให้ไว้ที่คอนเนคเตอร์ส่วนขาที่เหลือจะเป็นพีไอโอ ที่สามารถใช้งานได้ตามปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ไอซาบนชุดพัฒนาคอม 86

ไอซา (ISA) มีรูปแบบของการเชื่อมต่อกับอุปกรณ์แบบขนาน โดยได้รับการออกแบบให้มีลักษณะการเชื่อมต่อใช้งานคล้ายกับการเชื่อมต่อใช้งานไอซาบัส (ISA Bus) ของเครื่องคอมพิวเตอร์โดยทั่วไป จะแตกต่างกันเพียงรูปแบบการจัดวางอุปกรณ์ต่าง ๆ ที่พยายามจัดวางในรูปลักษณะของมาตรฐานพีซี 104 (PC/104) ซึ่งการออกแบบวงจรต่าง ๆ ที่นำมาเชื่อมต่อนั้นสามารถออกแบบในลักษณะเดียวกันกับการออกแบบการ์ดไอซา (ISA Card) ของเครื่องคอมพิวเตอร์ จะมีคอนเนคเตอร์สำหรับใช้ในการเชื่อมต่อกับการ์ดขยายภายนอกที่จะนำมาเชื่อมต่อ เป็นไปดังรูป



(NC =ขาสัญญาณที่ไม่มีการเชื่อมต่อ)

รูปที่ 4-2 แสดงคอนเนคเตอร์ของ ISA บนชุดพัฒนาคอม 86

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การเลือกใช้งานสัญญาณอินพุทและเอาท์พุท

HUOR นั้นมีฟังก์ชั่นการทำงานหลายอย่าง อาทิเช่น การเคลื่อนไหวส่วนต่าง ๆ ของร่างกาย, การอ่านค่าเซนเซอร์, การรับส่งข้อมูลระหว่างคอมพิวเตอร์และชุดพัฒนาคอม 86 ซึ่งการทำงานต่าง ๆ มีความเป็นอิสระต่อกัน จากการออกแบบโครงสร้าง HUOR การควบคุมการเคลื่อนไหวก็คือการควบคุมมอเตอร์และเซอร์โวนั่นเอง ส่วนอินฟราเรดเซนเซอร์ที่ติดไว้ตามจุดต่างๆ ทั้ง 12 จุดนั้นแต่ละจุดใช้เพียง 1 บิตในการรับข้อมูลเข้ามาในคอม 86 ในการหาระยะทางนั้นเราได้นำโมดูลอัลตราโซนิค (Ultrasonic Module) มาใช้

มอเตอร์กระแสตรง-เซอร์โวมอเตอร์				
ส่วนประกอบ	จำนวน	จำนวนมอเตอร์ที่ใช้	จำนวนเซอร์โวที่ใช้	จำนวนบิตที่ใช้
ตา	1	2	0	4
แขน	2	2	0	8
มือ	2	2	0	8
ขา	4	0	2	8
ล้อ	4	1	0	8
ลำตัว	1	0	1	1
อินฟราเรดเซนเซอร์				
ส่วนประกอบ	**จำนวน**	**จำนวนเซนเซอร์ที่ใช้**		**จำนวนบิตที่ใช้**
แขน	2	1		2
ขา	4	1		4
ลำตัว (ด้านหน้า)	1	4		4
ลำตัว (ด้านล่าง)	1	4		4
อัลตราโซนิคเซนเซอร์				
ส่วนประกอบ	จำนวน	เซนเซอร์ตัวส่ง	เซนเซอร์ตัวรับ	จำนวนบิตที่ใช้
ตา	1	1	1	2
หลอดไฟ				
ส่วนประกอบ		**จำนวน**		**จำนวนบิตที่ใช้**
ตา		1		1
รวมจำนวนบิตที่ใช้ในการควบคุมทั้งหมด			**54**	

ตารางที่ 4-1 แสดงอุปกรณ์ที่ต้องทำการควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นต้องมีการออกแบบวงจรการทำงานต่าง ๆ ขึ้นมาติดต่อกับพอร์ตของชุดพัฒนาคอม 86 ซึ่งในชุดพัฒนาคอม 86 มี พอร์ตอินพุท/เอาท์พุท ให้เลือกใช้อยู่ 3 พอร์ตคือ PIO, ISA, GCI และจากการทดลองเขียนโปรแกรมติดต่อพอร์ตต่างๆ ดูพบว่าพอร์ตที่เหมาะสมที่จะนำมาใช้กับ HUOR คือพอร์ต PIO และ พอร์ต ISA เนื่องจาก พอร์ต PIO เราสามารถควบคุมได้ถึงระดับบิตทำให้ง่ายในการควบคุมโดยการอ้างบิตที่ต้องการติดต่อโดยตรงและมีจำนวนบิตให้เลือกใช้งานได้ถึง 25 บิต พอร์ต PIO ใช้ 3 แอดเดรสในการติดต่อคือ PIO1 PIO2 และPIO3 แต่ในแต่ละแอดเดรสไม่สามารถควบคุมได้ทุกบิตคือ PIO1 ได้ 6 บิต PIO2 ได้ 10 บิต PIO3 ได้ 12 บิต ส่วนพอร์ต ISA นั้นเราสามารถเพิ่มช่องทางข้อมูลในการติดต่อได้มากถึง 65536 ช่องทางโดยการเพิ่มส่วน แอดเดรสดีโค้ดเดอร์ (Address Decoder) ข้อเสียของ ISA คือมีการทำงานที่ช้า และต้องเพิ่มวงจรส่วนแอดเดรสดีโค้ดเดอร์ หากต้องการติดต่อข้อมูลจำนวนมากหรือหลายแอดเดรส

4.4 บัฟเฟอร์และแลตช์ (Buffer and Latch)

ในการกำหนดตำแหน่งพอร์ตที่ใช้ในการควบคุมหุ่นนั้น สิ่งที่ต้องคำนึงถึงเป็นลำดับแรกคือจำนวนบิตที่ต้องการควบคุมหรือติดต่อกับส่วนต่าง ๆ สิ่งต่อมาคือ ข้อมูลนั้นเป็นข้อมูลเข้า (Input) หรือข้อมูลออก (Output) แล้วนำมาจัดกลุ่มรวมเข้าไว้ด้วยกัน

เพื่อให้ประหยัดคำสั่งที่ใช้ควบคุมรวมทั้งง่ายในการกำหนดค่าที่จะควบคุม ควรจัดข้อมูลส่วนที่มักจะมีการควบคุมพร้อม ๆ กันไว้ด้วยกัน และควรมีจำนวนบิตรวมครบ 1 ไบต์ (8 บิต) เช่น มือซ้ายและแขนซ้าย (4 บิต + 4 บิต)

จากพอร์ตและจำนวนบิตบนชุดพัฒนาคอม 86 จึงได้ทำการกำหนดการรับส่งข้อมูลต่าง ๆ กับพอร์ตบนชุดพัฒนาคอม 86 ตามตารางที่ 4-2 ดังนี้

Control(No.)(Mtr.)	Bits	In	Out	Latch	Buffer	PIO1(6)	PIO2(10)	PIO3(12)	ISA
Hand(2)(2)	8		/	/					/
Arm(2)(2)	8		/	/					/
Light(1)	1		/	/					/
Mode(1)	1		/	/					/
Movement	5		/	/					/
Eye(1)(2)	4		/		/	/			
Rang Finder(1)	8	/			/		/		
Infrared Sensor(10)	12	/			/			/	
Bit Total	47	20	39	29	23	4	8	12	23

ตารางที่ 4-2 แสดงจำนวนบิตที่ใช้ในแต่ละพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Control(bits)(No.)	C51_1	C51_2	C51_3	C51_4	C51_5	In/Out
Mode(1)	/	/	/	/	/	I
Direction Data(5)	/	/	/	/		I
Wheel_1(2)	/					O
Wheel_2(2)		/				O
Wheel_3(2)			/			O
Wheel_4(2)				/		O
Leg_1(1)(2)	/					O
Leg_2(1)(2)		/				O
Leg_3(1)(2)			/			O
Leg_4(1)(2)				/		O
Range Finder						
Range(8)					/	O
US Sender(1)					/	O
US Receiver(1)					/	I
Mode Changing						
Change Mode(1)					/	O
Total Bit	I=6,O=4	I=6,O=4	I=6,O=4	I=6,O=4	I=2,O=10	

ตารางที่ 4-3 แสดงการใช้งานไมโครคอนโทรลเลอร์ MCS-51

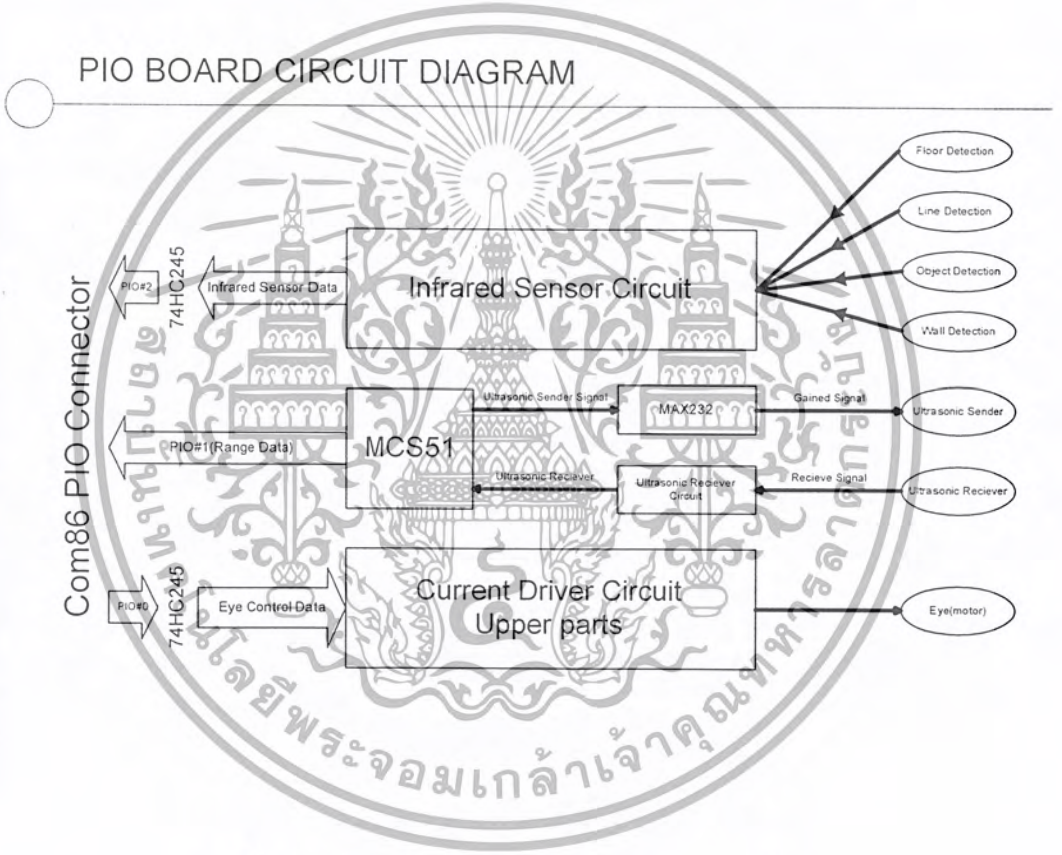
จากตารางที่ 4-2 ในส่วนของการเคลื่อนที่ (Movement) และส่วนหาระยะทาง (Range Finder) ได้นำ ไมโครคอนโทรลเลอร์ MCS51(89C2051) เข้ามาช่วยในการควบคุมการเคลื่อนไหวของขาและการควบคุมโมดูลอัลตราโซนิค รวมทั้งคำนวณระยะทาง เพื่อเป็นการลดภาระการทำงานของคอม 86 ลงซึ่งคอม 86 จะทำงานเพียงส่งทิศทางการเคลื่อนที่และรับระยะทางที่ตรวจวัดได้มาจากไมโครคอนโทรลเลอร์ MCS51 เท่านั้น ดังตารางที่ 4-3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 วงจรควบคุมการทำงาน

4.5.1 วงจรบอร์ดพีไอโอ

จากหัวข้อที่แล้ว เราทราบแล้วว่าในแต่ละส่วนการทำงานนั้นต้องใช้อุปกรณ์ใด จำนวนเท่าใด และจะต้องใช้บิตในการควบคุมการทำงานอย่างไร ซึ่งในหัวข้อนี้จะอธิบายถึงวงจรที่ได้รับการออกแบบในส่วนของพีไอโอพอร์ต ซึ่งคอนเนคเตอร์พีไอโอของชุดพัฒนาคอม 86 นั้นมีจำนวน 2 คอนเนคเตอร์ แต่การอ้างถึงในการเขียนโปรแกรมนั้นจะอ้างถึงรีจิสเตอร์ ซึ่งมีอยู่ 3 ตัว คือ PIODATA0, PIODATA1 และ PIODATA2 ซึ่งจะทำการควบคุมหมายเลขพีไอโอ PIO0-15, PIO16-31, PIO32-47 ตามลำดับ ดังนั้นในการออกแบบวงจรในส่วนนี้จะต้องคำนึงถึงความสะดวกในการพัฒนาโปรแกรมต่อไปด้วย รูปด้านล่างแสดงบล็อก ไดอะแกรม (Block Diagrams) ของวงจรบอร์ดพีไอโอ โดยอ้างอิงกับตารางที่ 4-2

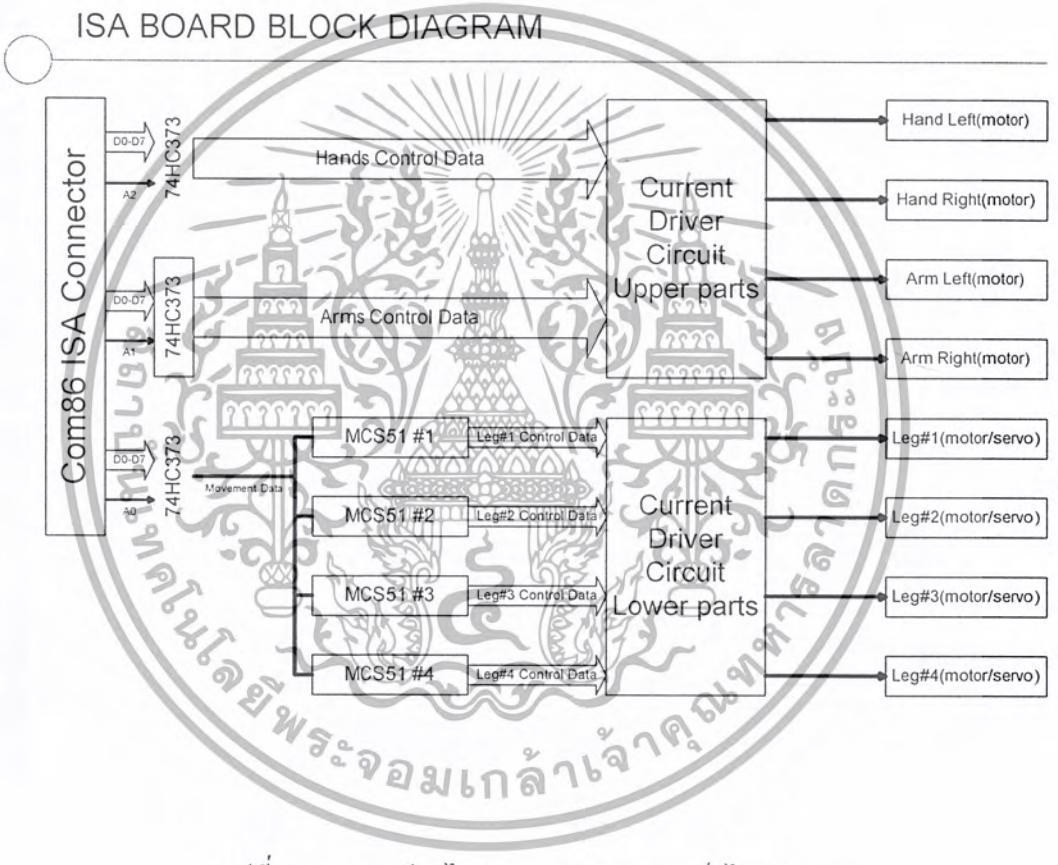


รูปที่ 4-3 แสดงบล็อกไดอะแกรมของวงจรบอร์ดพีไอโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 วงจรบอร์ดไอซา

ในการใช้งานติดต่อกับไอซาพอร์ตนั้น เราสามารถอ้างแอดเดรสได้โดยตรง ซึ่งจากตารางที่ 4-2 ทำให้เราทราบถึงจำนวนบิตที่จำเป็นต้องใช้งานสำหรับไอซาพอร์ต เราจึงนำข้อมูลที่ได้มาทำการออกแบบวงจรควบคุมการทำงาน โดยใช้แอดเดรสในการอีนาเบิล(enable) การทำงานของแลตช์ข้อมูล โดยในที่นี้เราใช้แลตช์ในการควบคุมการทำงานจำนวน 3 ตัวคือ ควบคุมการทำงานของระบบการเคลื่อนที่, ควบคุมการเคลื่อนไหวของมือและแขนทั้งสอง ซึ่งแลตช์แต่ละตัวจะอีนาเบิลให้กับข้อมูลการทำงานกับแอดเดรสที่ต่างกันคือแอดเดรส 0x100B, 0x100D และ 0x100E โดยจะสามารถทำงานพร้อมกันได้หากได้รับแอดเดรส 0x1008 รูปด้านล่างแสดงบล็อกไดอะแกรมของวงจรบอร์ดไอซา



รูปที่ 4-4 แสดงบล็อกไดอะแกรมของวงจรบอร์ดไอซา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

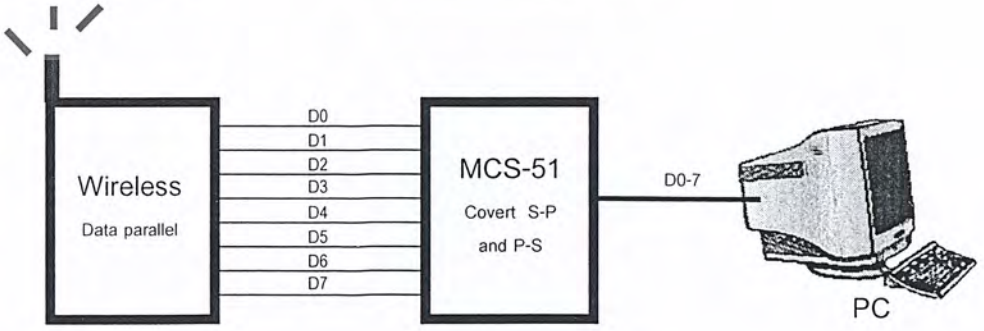
4.5.3 วงจรรับ-ส่งคลื่นวิทยุ

สิ่งหนึ่งที่สำคัญของ HUOR คือการติดต่อสื่อสารในการรับ-ส่งข้อมูลแบบไร้สาย ซึ่งในส่วนนี้ต้องการการทำงานที่เป็นเวลาจริง (Real Time) มากที่สุดเท่าที่จะเป็นไปได้ แต่เรามีข้อจำกัดในเรื่องของเงินทุน เนื่องจากชุดวงจรไร้สาย (wireless) ที่มีประสิทธิภาพสูงทั้งความน่าเชื่อถือของข้อมูลและระยะส่งที่ไกลนั้น มีราคาแพง เราจึงต้องทำการสรรหาชุดรับ-ส่งแบบไร้สาย ที่มีราคาถูกและมีประสิทธิภาพที่ยอมรับได้มาใช้งาน

ชุดวงจรรับ-ส่งข้อมูลจากรถกระป๋อง ซึ่ง มี TX-2 / RX-2 เป็น IC ที่เป็นตัวรับส่งข้อมูลระหว่างกัน สามารถรับส่งได้ 5 ฟังก์ชั่น (Five functions remote controls) มีจุดเด่นคือราคาถูกและมีประสิทธิภาพที่ยอมรับได้

รูปที่ 4-5 แสดงฟังก์ชั่นการทำงานของ TX-2 / RX-2

ในการใช้งานของ HUOR นั้น วงจรรถกระป๋องต้องมีการใช้งาน 2 ส่วนคือ ภาครับและภาคส่ง ซึ่งเราได้นำวงจร 2 ชุดที่มีความถี่ต่างกัน (Difference Frequency) มาใช้ร่วมกัน และในการทำงานของวงจรรถกระป๋องนั้นมีการทำงานเป็นบิต (แบบขนาน) แต่เราต้องการติดต่อกับคอมพิวเตอร์ (PC) และชุดพัฒนาคอม 86 ผ่านทางพอร์ตสื่อสารอนุกรม (Serial port) ทำให้ต้องมีการแปลงข้อมูลจาก Serial to Parallel และ Parallel to Serial ซึ่งในที่นี้เราได้เลือกใช้ไมโครคอนโทรลเลอร์ MCS-51 ซึ่งมี ยูอาร์ต ภายในอยู่แล้วมาใช้งาน ดังรูป

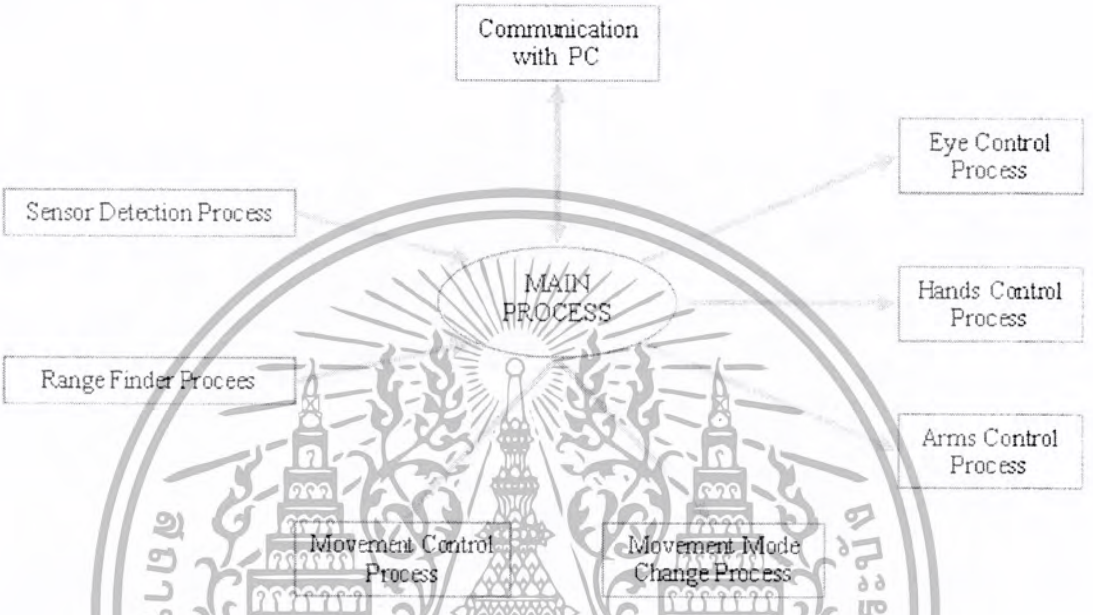


รูปที่ 4-6 แสดงการรับ-ส่งข้อมูล ผ่านวงจร RF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การออกแบบส่วนโปรแกรมการทำงาน

5.1 โปรแกรมการทำงานบนชุดพัฒนาคอม 86



รูปที่ 5-1 แสดงกระบวนการการทำงานหลักของหุ่นยนต์

5.1.1 กระบวนการทำงานหลัก : Main Process(Com86)

เป็นศูนย์กลางการทำงานของแต่ละกระบวนการทำงานในชุดพัฒนาคอม 86 โดยจะนำค่าหรือข้อมูลต่าง ๆ ในแต่ละกระบวนการทำงาน เช่น กระบวนการตรวจสอบการทำงานของเซนเซอร์ (Sensor Detection Process), กระบวนการตรวจหาระยะทาง (Range Finder Process) หรือค่าที่รับจากคอมพิวเตอร์มาประมวลผลและทำการควบคุมกระบวนการการเคลื่อนที่ (Movement Control Process), กระบวนการการเปลี่ยนโหมดการเคลื่อนที่ (Movement Mode Change Process) หรือกระบวนการที่ใช้ในการเคลื่อนไหวส่วนต่าง ๆ ของหุ่นยนต์ ดังนั้นกระบวนการนี้จึงเปรียบเสมือนศูนย์กลางความคิดของหุ่นยนต์

5.1.2 กระบวนการตรวจสอบการทำงานของเซนเซอร์ : Sensor Detection Process

เป็นกระบวนการที่จะทำหน้าที่ตรวจสอบค่าสถานะของอินฟราเรดเซนเซอร์ (Infrared Sensor) ที่ติดตั้งไว้บนตัวหุ่นยนต์ ณ ตำแหน่งต่าง ๆ ซึ่งจะทำการตรวจจับ (Detection) กรณีต่างๆ ได้แก่ การตรวจจับความเหมาะสมของพื้นในการก้าวเดิน (Floor Detection), การตรวจจับเส้น (Line Detection) ในกรณีที่เคลื่อนที่ตามเส้น, การตรวจจับสิ่งกีดขวาง (Wall Detection), การตรวจจับวัตถุที่ปลายมือ (Object Detection) และส่งข้อมูลให้กระบวนการทำงานหลักประมวลผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.3 กระบวนการตรวจหาระยะทาง : Range Finder Process

ทำหน้าที่ตรวจวัดระยะทาง โดยทำงานร่วมกับชุดอัลตราโซนิคเซนเซอร์ (Ultrasonic Sensor Module) ซึ่งเป็นอุปกรณ์ที่มีหลักการทำงานคือ อัลตราโซนิคเซนเซอร์จะทำการส่งคลื่นเสียงออกไป เมื่อเสียงกระทบวัตถุ ก็จะสะท้อนกลับมาและเมื่อนำค่าเวลาที่จับได้มาคำนวณเทียบกับความเร็วเสียง ก็จะได้ระยะทางออกมา ส่งต่อให้กระบวนการทำงานหลักประมวลผล

5.1.4 กระบวนการควบคุมตา : Eye Control Process

มีหน้าที่ในการรับคำสั่งจากกระบวนการทำงานหลัก เพื่อทำการควบคุมทิศทางการมองของกล้อง ที่ทำหน้าที่เปรียบเสมือนตาของหุ่นยนต์ โดยจะทำการควบคุมมอเตอร์ 2 ตัว คือ มอเตอร์ควบคุมทิศทางการมองเห็นในแนวนอน และมอเตอร์ควบคุมทิศทางการมองเห็นในแนวตั้ง

5.1.5 กระบวนการควบคุมมือ : Hands Control Process

มีหน้าที่ในการรับคำสั่งจากกระบวนการทำงานหลัก เพื่อทำการควบคุมการเคลื่อนไหวของมือทั้งสองข้างเช่น การหยิบจับ-ปล่อย ,การหมุนข้อมือ

5.1.6 กระบวนการควบคุมแขน : Arms Control Process

มีหน้าที่ในการรับคำสั่งจากกระบวนการทำงานหลัก เพื่อทำการควบคุมการเคลื่อนไหวของแขนทั้งสองข้าง เช่น การยกแขนขึ้น-ลง, การกวาดแขนซ้าย-ขวาในแนวระดับ

5.1.7 กระบวนการควบคุมการเคลื่อนที่ : Movement Control Process

ควบคุมการเคลื่อนที่ของหุ่นยนต์ โดยสัมพันธ์กับ รูปแบบการเคลื่อนที่ คือ

Walking mode – การเคลื่อนที่ในลักษณะการเดินด้วยขา 4 ขา โดยกระบวนการทำงานจะทำการเปรียบเทียบตำแหน่งขาปัจจุบัน (current state) กับตำแหน่งขาที่ต้องการ (next state) แล้วจะทำการคำนวณความเร็ว ทิศทางและองศาการวางของขาแต่ละข้าง และทำการควบคุมขาแต่ละข้างต่อไป

Running mode – การเคลื่อนที่ในลักษณะของล้อออมนิ (omni wheel) ซึ่งมีความคล่องตัวและความสามารถในการเคลื่อนที่ได้ถึง 8 ทิศทาง โดยกระบวนการทำงานจะรับคำสั่งทิศทางจากกระบวนการทำงานหลัก และทำการประมวลผลเพื่อควบคุมมอเตอร์ล้อ โดยมอเตอร์ล้อแต่ละตัวเป็นอิสระต่อกัน

5.1.8 กระบวนการเปลี่ยนโหมดการเคลื่อนที่ : Movement Mode Change Process

เมื่อได้รับคำสั่งให้เปลี่ยนโหมดการเคลื่อนที่ กระบวนการทำงานจะควบคุมการทำงานของระบบกลไกที่ใช้ในการเปลี่ยนโหมด รวมทั้งปรับตั้งตำแหน่งขาของหุ่นให้อยู่ในลักษณะตามการเคลื่อนที่ที่ต้องการ

5.1.9 กระบวนการติดต่อสื่อสาร : Communication with PC

เป็นกระบวนการที่ควบคุมการติดต่อสื่อสารกับคอมพิวเตอร์ โดยผ่านอุปกรณ์ติดต่อไร้สายที่เชื่อมต่อกับพอร์ตอนุกรมของคอม86 ซึ่งข้อมูลที่จะส่งต้องทำการเข้ารหัสให้มีขนาด 8 บิตก่อนเพื่อประโยชน์ในด้านประสิทธิภาพการสื่อสาร ในส่วนของการรับข้อมูลก็ต้องนำข้อมูล 8 บิตนั้นมาถอดรหัสแล้วทำการส่งข้อมูลให้กระบวนการทำงานหลักการประมวลผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 โปรโตคอลการรับ-ส่งข้อมูล

จากการศึกษาพบว่าวงจรรับ-ส่งข้อมูลผ่านคลื่นวิทยุที่มีอยู่นั้นมีการใช้งานได้เพียง 2 ช่องสัญญาณ (Channel) นั่นหมายความว่าเราสามารถทำการรับ-ส่งบิตข้อมูลที่ต้องการให้ต่างกันได้เพียง 1 คู่ โดย TX-2 และ RX-2 จะมีฟังก์ชันการทำงานภายในทำการสร้างบิตข้อมูลที่ตรงกันข้าม (not bit) โดยอัตโนมัติ และมี 1 ช่องสัญญาณที่มีบิตพิเศษ 1 บิตเพื่อเพิ่มการใช้งานที่หลากหลายมากขึ้น เราจึงได้บิตข้อมูลในการใช้งานจริง 5 ฟังก์ชัน แต่ทั้งนี้ บิตข้อมูลทั้ง 5 นั้นไม่เป็นอิสระต่อกัน ซึ่งในจุดนี้เป็นข้อจำกัดหนึ่งในการทำงาน เราจึงต้องทำการออกแบบโปรโตคอลในการรับส่งข้อมูลให้มีความถูกต้องตอบสนองการใช้งานได้

วิธีการเลือกส่งเฉพาะบิตข้อมูลที่เป็น '1' (High bit) โดยทำการส่งข้อมูลในรูปแบบของตำแหน่งบิตข้อมูลที่เป็น '1' ออกไป แล้วฝ่ายรับจะนำข้อมูลที่ได้ไปทำการเปรียบเทียบกับตารางเพื่อหาว่าตำแหน่งใดที่เป็น '1' ซึ่งก็จะทำให้ทราบว่าข้อมูลตำแหน่งก่อนหน้าเป็น '0' ทำอย่างนี้ไปเรื่อย ๆ จนเมื่อส่งข้อมูลครบ 1 byte ก็ทำการส่งคำสั่งควบคุม (control word) "stop bit" ปิดท้าย เพื่อบอกให้ฝ่ายรับทราบว่าส่งข้อมูลครบ 1 ไบต์แล้ว ให้รอรับข้อมูลไบต์ ต่อไปได้เลย

ตารางด้านล่างแสดงความหมายของคำสั่งต่าง ๆ ซึ่งแบ่งได้เป็น 2 กลุ่ม คือ กลุ่มของการส่งข้อมูล (Data) และ กลุ่มของคำสั่งควบคุม (Control)

5.2.1 รูปแบบกลุ่มข้อมูล

ความหมาย	channel 1			channel 2	
	TB	FW	BW	L	R
ไม่มีข้อมูล	0	0	0	0	0
ข้อมูล D7	0	1	0	1	0
ข้อมูล D6	0	1	0	0	1
ข้อมูล D5	0	0	1	1	0
ข้อมูล D4	0	0	1	0	1
ข้อมูล D3	0	1	0	0	0
ข้อมูล D2	0	0	1	0	0
ข้อมูล D1	0	0	0	1	0
ข้อมูล D0	0	0	0	0	1

ตาราง 5-1 แสดงรูปแบบกลุ่มคำสั่งของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2 คำสั่งควบคุมจากคอมพิวเตอร์

เป็นคำสั่งที่คอมพิวเตอร์ส่งให้ คอม86 ประกอบไปด้วย เฮดเดอร์ไบต์ (Header byte) ซึ่งจะเป็นไบต์ (byte) ที่กำหนดว่าส่วนใดบ้างที่จะทำงานหรือหยุดทำงาน รวมทั้งบอกว่าจะมีข้อมูลตามมาอีกกี่ไบต์ (Data byte) ซึ่งข้อมูลที่ตามมาจะเป็นตัวกำหนดทิศทางการเคลื่อนไหวของส่วนต่าง ๆ ตามลำดับ

ไบต์ข้อมูล (Data byte) เป็นตัวกำหนดทิศทางการเคลื่อนไหวของส่วนต่าง ๆ ตามลำดับ คือ หากมีการกำหนดให้ สามารถเคลื่อนที่, มือและแขนซ้ายขยับได้ จะได้ว่าเฮดเดอร์ไบต์จะมีค่า 11110000B และจะมีข้อมูลตามมาอีก 2 ไบต์คือ ข้อมูลกำหนดการเคลื่อนไหวของ การเคลื่อนที่ 1 ไบต์, มือและแขนซ้าย 1 ไบต์ ตามลำดับ

Header '11' : Enable and Disable part

D7	D6	D5	D4	D3	D2	D1	D0
1	1	**Move/Eye**	**Left**	**Right**	**Mv_mode**	**Opt_mode**	**Light**
		0 – dis	0 – dis	0 – dis	0 – legs	0 – man	0 – off
		1 – en	1 - en	1 – en	1 – wheel	1 – auto	1 - on

ตาราง 5-2 แสดงรูปแบบคำสั่งอีนาเบิลการทำงานส่วนต่าง ๆ

Data => Movement and Eyes

D7	D6	D5	D4	D3	D2	D1	D0
xx		**Movement**			**Eyes**		
00 – move		000 – Forward			000 – Upper		
01 – eye		001 – Backward			001 – Lower		
10 – both		010 – Turn left			010 – Slice left		
11 – not use		011 – Turn right			011 – Slice right		
		100 – Forward / right			100 – Upper / right		
		101 – Forward / left			101 – Upper / left		
		110 – Backward / right			110 – Lower / right		
		111 – Backward / left			111 – Lower / left		

ตาราง 5-3 แสดงรูปแบบข้อมูลสั่งการระบบการเคลื่อนที่และตา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data => Left arms and Left hands

D7	D6	D5	D4	D3	D2	D1	D0
xx		Left hands			Left arms		
00 – arms		000 – Grip			000 – Upper		
01 – hands		001 – Release			001 – Lower		
10 – both		010 – Spin left			010 – Spin left		
11 – not use		011 – Spin right			011 – Spin right		
		100 – Grip / right			100 – Upper / right		
		101 – Grip / left			101 – Upper / left		
		110 – Release / right			110 – Lower / right		
		111 – Release / left			111 – Lower / left		

ตาราง 5-4 แสดงรูปแบบข้อมูลสั่งการระบบแขนและมือข้างซ้าย

Data => Right arms and Right hands

D7	D6	D5	D4	D3	D2	D1	D0
xx		Right hands			Right arms		
00 – arms		000 – Upper			000 – Upper		
01 – hands		001 – Lower			001 – Lower		
10 – both		010 – Spin left			010 – Spin left		
11 – not use		011 – Spin right			011 – Spin right		
		100 – Upper / right			100 – Upper / right		
		101 – Upper / left			101 – Upper / left		
		110 – Lower / right			110 – Lower / right		
		111 – Lower / left			111 – Lower / left		

ตาราง 5-5 แสดงรูปแบบข้อมูลสั่งการระบบแขนและมือข้างขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.3 สถานะการทำงานจากชุดพัฒนาคอม 86

เป็นข้อมูลที่ คอม86 ส่งให้ พีซีเพื่อแจ้งสถานะการทำงาน ได้แก่ ทิศทางการเคลื่อนที่(Movement and Direction), โหมดการเคลื่อนที่ (Movement mode), ชนิดของเส้นที่ตรวจจับได้ (Line Detect Type), วัตถุที่มือทั้งสองข้าง (Object Detection), สถานะของพื้นผิวที่หุ่นกำลังเคลื่อนที่ไป (Floor Detection), ระยะทางที่ห่างจากวัตถุใดๆ (Range) เพื่อนำมาแสดงผลบนส่วนติดต่อผู้ใช้ หรือสามารถนำมาประมวลผล ซึ่งกำหนดให้ 2 บิตแรกเป็นตัวกำหนดว่าไบต์นั้นเป็นข้อมูลของอะไร คือ

'00' : Movement and Direction

'01' : Line detect type, Hand sensor, Floor Detection

'10' : Movement mode, 4-bit high length finder

'11' : Hands sensor, 4-bit low length finder

ในการส่งข้อมูล คอม86 จะทำการส่งเฉพาะข้อมูลไบต์ที่มีการเปลี่ยนแปลงเท่านั้น เพื่อเพิ่มประสิทธิภาพในการรับ-ส่งข้อมูล ดังนั้นข้อมูลที่คอมพิวเตอร์ได้รับจึงไม่จำเป็นต้องมีลำดับเหมือนการส่งข้อมูลจาก คอมพิวเตอร์ให้แก่คอม86

Header '00' : Movement and direction

D7	D6	D5	D4	D3	D2	D1	D0
0	0	x	x	Move	Direction		
				0 – stop	000 – Forward		
				1 – move	001 – Backward		
					010 – Turn left		
					011 – Turn right		
					100 – Forward / right		
					101 – Forward / left		
					110 – Backward / right		
					111 – Backward / left		

ตาราง 5-6 แสดงรูปแบบข้อมูลสถานะของระบบการเคลื่อนที่และทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Header '01' : Line detect type , Wall Detection , Floor Detection

D7	D6	D5	D4	D3	D2	D1	D0
0	1	Line type		L-sensor	R-sensor	Floor Detection	
		00 – type 1		0 – null	0 – null	00 – ok go on	
		01 – type 2		1 – fine	1 – fine	01 – no (turn right)	
		10 – type 3				10 – no (turn left)	
		11 – type 4				11 – no (stop)	

Header '10' : 4-bit high length finder , movement mode

D7	D6	D5	D4	D3	D2	D1	D0
1	0	Movement mode		4-bit high range finder			
		00 – upper walk		0000 - 1111			
		01 – lower walk					
		10 – wheels					
		11 – null					

Header '11' : 4-bit low length finder , object detection

D7	D6	D5	D4	D3	D2	D1	D0
1	1	Hands sensor		4-bit low range finder			
		00 – null		0000 - 1111			
		01 – right hand					
		10 – left hand					
		11 – both					

ตาราง 5-7 แสดงรูปแบบข้อมูลสถานะของอินฟราเรดเซนเซอร์และอัลตร้าโซนิคเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 กระบวนการควบคุมจังหวะการทำงานของขา

เป็นกระบวนการที่ทำงานบนไมโครคอนโทรลเลอร์ 89c2051 (MCS-51) โดยรับข้อมูลการเคลื่อนที่ (Movement Data) จาก พอร์ตไอซาของชุดพัฒนาคอม86 แล้วนำมาประมวลผลเพื่อทำการควบคุมการเคลื่อนไหวของขาและล้อแต่ละข้าง

ข้อมูลที่ได้รับจะมีทั้งหมด 6 บิต ประกอบไปด้วย ระดับความเร็ว (Speed) 1 บิต, โหมดการเคลื่อนที่ (Movement Mode) 1 บิต, ข้อมูลทิศทาง (Direction Data) 4 บิต

Direction Data(5 bit) C51(1-4) เห็นข้อมูลเหมือนกัน(Data 4 bit , Speed 1 bit)

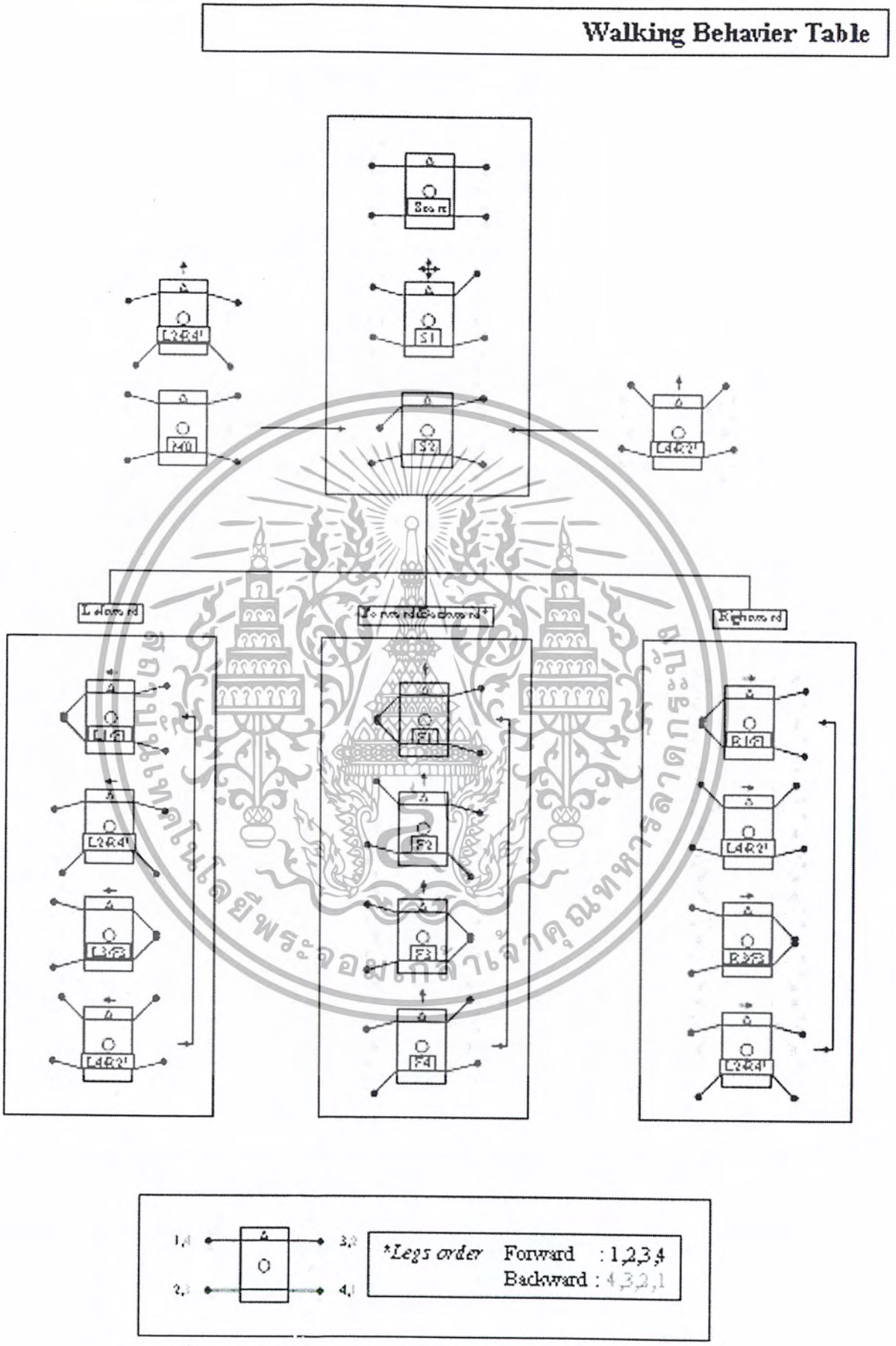
Data	Direction	Leg Mode	Omni Mode	Wheel Mode
Initial State		[4,1,3,2]	[0,0,0,0]	[0,0,0,0]
0100	up	up	Up	up
0101	down	down	Down	down
0110	left	rotate left	Left	rotate left
0111	right	rotate right	Right	rotate right
1000	up+left	rotate left + up	135'	rotate left + up
1001	down+right	rotate left + down	315'	rotate left + down
1010	down+left	rotate right + down	225'	rotate right + down
1011	up+right	rotate right + up	45'	rotate right + up
1100	rotate left	-	rotate left	-
1101	rotate right	-	rotate right	-
1111	leg lift	leg lift up for wheel mode	leg lift down for leg mode	leg lift down for leg mode
0000	stop	Stop	Stop	stop

Movement Mode : 0 (running mode), 1 (walking mode)

Speed : 0 (low speed), 1(high speed)

ตารางที่ 5-8 แสดงรูปแบบข้อมูลทิศทาง (Data Direction)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-2แสดงรูปแบบสถานการณ์ก้าวเดินในโหมดการเดินด้วยขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะขา (Leg State) คือ ข้อมูลที่บอกตำแหน่งของขาทั้ง 4 ข้างตัวอย่างเช่น LS[4,1,3,2] เป็นตำแหน่งเริ่มต้นของขาจะได้ว่า ขาที่ 1 อยู่ที่ตำแหน่งที่ 4 , ขาที่ 2 อยู่ที่ตำแหน่งที่ 1 , ขาที่ 3 อยู่ที่ตำแหน่งที่ 3 และขาที่ 4 อยู่ที่ตำแหน่งที่ 2 โดยตำแหน่งของขาจะมีการเรียงเหมือนกับขาของ IC ทั่วไป

จากตารางที่ 5-2 และ รูปที่ 5-2 เมื่อ MCS-51 ทั้ง 4 ตัว รับข้อมูลการเคลื่อนที่จากคอม 86 จะทำการตรวจสอบตำแหน่งขาในสเตทปัจจุบัน(Current State)กับตำแหน่งขาในสเตทที่จะก้าวไป(Next State) โดยการเปลี่ยนสเตทนั้นจะสัมพันธ์กับทิศทางที่ HUOR จะเคลื่อนไป ยกตัวอย่างเช่น ในรูปที่ 5-1 หากสเตทขาเริ่มต้นคือ LS[4,1,3,2] แล้วได้รับคำสั่งให้เคลื่อนที่ไปข้างหน้า สเตทถัดไปคือ LS[1,2,4,3] หมายความว่า ขาที่ 2 จะก้าวไปที่ตำแหน่งที่ 2 ,ขาที่ 3 จะก้าวไปที่ตำแหน่งที่ 4 ขาที่ 4 จะก้าวไปที่ตำแหน่งที่ 3 ส่วนขา 1 นั้นจะยกและก้าวไปในตำแหน่งที่ 1

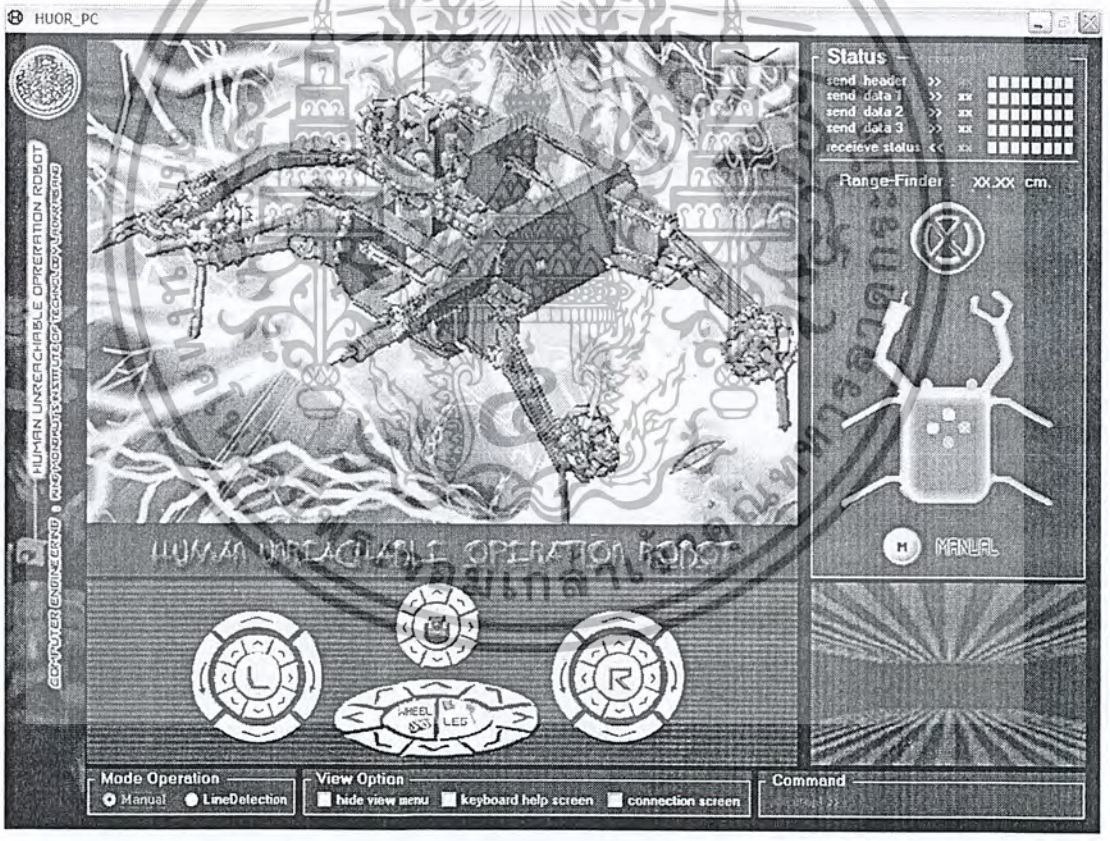
แต่ละการก้าวขาและยกขาไปในตำแหน่งต่างๆนั้น MCS51 ที่ควบคุมขาแต่ละข้างจะจ่ายสัญญาณพัลส์ให้แก่เซอร์โวที่ควบคุมการยกหรือก้าวขา ซึ่งช่วงพัลส์บวกที่ต่างกันก็จะส่งผลให้ตำแหน่งของขาต่างกันด้วย ส่วนในโหมด Running Mode นั้น MCS51 ควบคุมเพียงทิศทางการหมุนของล้อเท่านั้น

ในการทำงานนั้น MCS51 แต่ละตัวก็จะควบคุมขาเพียงข้างเดียว โดยจะมีการสื่อสารกันระหว่าง MCS51 เพื่อให้จังหวะการก้าวเดินเป็นไปอย่างถูกต้องและส่งผลให้ HUOR มีการทรงตัวที่ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
การออกแบบส่วนติดต่อผู้ใช้

การออกแบบโปรแกรมส่วนติดต่อผู้ใช้ (Graphic User Interface) สำหรับ HUOR นั้น เลือกใช้ Microsoft Visual Basic 6.0 ในการพัฒนาโปรแกรม เนื่องจากเป็นภาษาที่ง่ายต่อการเข้าใจและมีคอมโพเนนท์ (Component) ที่ตรงกับความต้องการแนวคิดในการออกแบบจะเน้นตรงที่หน้าตาโปรแกรมนั้นต้องมีความน่าสนใจ และมีการวางปุ่มควบคุมต่าง ๆ ในตำแหน่งที่เหมาะสม และที่สำคัญที่สุดคือจะต้องออกแบบโดยคำนึงถึงความง่ายต่อการเข้าใจและการใช้งาน (User Friendly) เนื่องจากว่าการควบคุม HUOR นั้นค่อนข้างซับซ้อนและมีคำสั่งในการควบคุมมากมาย จึงเป็นจุดหนึ่งที่ต้องให้ความสำคัญในการจัดวางองค์ประกอบระหว่างการเขียนโปรแกรม รูปด้านล่างแสดงถึงองค์ประกอบของหน้าหลักของโปรแกรมส่วนติดต่อผู้ใช้



รูปที่ 6-1 แสดงหน้าหลักของโปรแกรมส่วนติดต่อผู้ใช้

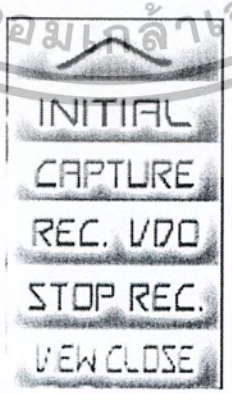
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

องค์ประกอบของโปรแกรมติดต่อผู้ใช้มี 5 ส่วนดังนี้

รูปที่ 6-2 แสดง Use case การทำงานของส่วนติดต่อผู้ใช้

6.1 ส่วนการติดต่อกับกล้องวงจรปิดไร้สาย

เป็นส่วนที่แสดงภาพเคลื่อนไหว โดยรับสัญญาณมาจากกล้องวงจรปิดไร้สาย ซึ่งในส่วนนี้จะมีฟังก์ชันการทำงานดังต่อไปนี้



รูปที่ 6-3 แสดงเมนูการเรียกใช้ฟังก์ชันส่วนแสดงภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.1 ฟังก์ชันเริ่มต้นการติดต่อกับกล้องวงจรปิด : initial_view()

เป็นฟังก์ชันที่ทำการกำหนดค่าเริ่มต้นให้กับการแสดงภาพเคลื่อนไหว การติดต่อกับอุปกรณ์รับสัญญาณภาพ การกำหนดขนาดของการแสดงผลภาพเคลื่อนไหว

6.1.2 ฟังก์ชันการบันทึกภาพนิ่ง : capture_view()

เป็นฟังก์ชันที่ช่วยให้ผู้ควบคุมหุ่นยนต์ทำการบันทึกภาพนิ่งจากภาพเคลื่อนไหวที่แสดงผลอยู่ โดยสามารถเลือกแฟ้มและตั้งชื่อไฟล์ที่ต้องการบันทึกลงฮาร์ดดิสก์ได้ และอนุญาตให้บันทึกเป็นไฟล์รูปภาพหลายประเภท เช่น JPEG,BMP และ GIF เป็นต้น

6.1.3 ฟังก์ชันการบันทึกภาพวีดีโอ : record_view()

เป็นฟังก์ชันที่ช่วยให้ผู้ควบคุมสามารถทำการบันทึกภาพเคลื่อนไหวที่แสดงผลอยู่ในรูปของไฟล์วีดีโอ โดยไฟล์ที่บันทึกได้จะมีนามสกุลเป็น .avi

6.1.4 ฟังก์ชันหยุดการบันทึกภาพวีดีโอ : stop_record_view()

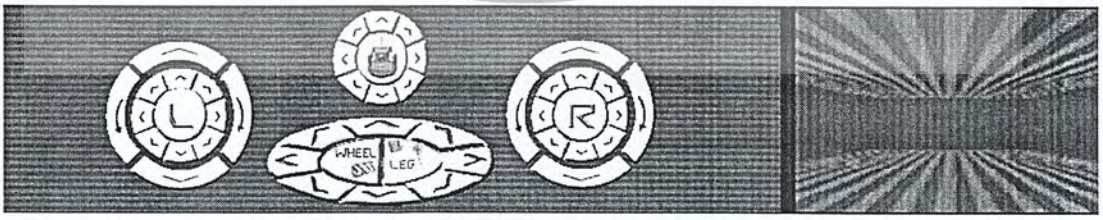
เป็นฟังก์ชันที่สั่งให้โปรแกรมหยุดการบันทึกไฟล์วีดีโอ

6.1.5 ฟังก์ชันยกเลิกการติดต่อกับกล้องวงจรปิด : close_view()

เป็นฟังก์ชันที่ทำการปิดการเชื่อมต่อกับอุปกรณ์รับสัญญาณภาพ ทำให้ภาพเคลื่อนไหวที่แสดงอยู่ทางหน้าจอโปรแกรมถูกปิดลงด้วย

6.2 ส่วนควบคุมการทำงานของหุ่นยนต์

เป็นส่วนที่มีบทบาทและมีความสำคัญที่สุด เนื่องจากว่าเป็นส่วนที่ทำหน้าที่สั่งการควบคุมการทำงานทั้งหมดของหุ่นยนต์ โดยใช้ปุ่มในการติดต่อกับผู้ใช้ ซึ่งต้องออกแบบอย่างรอบคอบและต้องครอบคลุมการใช้งานทั้งหมดด้วย โดยมีฟังก์ชันการทำงานดังนี้



รูปที่ 6-4 แสดงปุ่มควบคุมการทำงานของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2.1 ฟังก์ชันการเคลื่อนที่ :

เป็นฟังก์ชันที่ทำหน้าที่ในการควบคุมการเคลื่อนที่ของหุ่นยนต์ โดยทำการรับทิศทางการเคลื่อนที่จากการกดปุ่มของผู้ใช้ ว่าต้องการให้ไปทิศทางใด ซึ่งมีอยู่ 8 กรณี คือ move_forward(), move_backward(), move_left(), move_right(), move_forward_left(), move_forward_right(), move_backward_left(), move_backward_right() แล้วจึงส่งข้อมูลนั้นไปยังส่วนติดต่อสื่อสาร

6.2.2 ฟังก์ชันการเปลี่ยนโหมดการเคลื่อนที่ : change_movement_mode()

เป็นฟังก์ชันที่ทำหน้าที่ในการเปลี่ยนโหมดการเคลื่อนที่ระหว่างการเคลื่อนที่ในโหมดล้อทุกทิศทางและการเคลื่อนที่ในโหมดการเดินด้วยขา

6.2.3 ฟังก์ชันการเปลี่ยนโหมดการสั่งการ : change_operation_mode()

เป็นฟังก์ชันที่ใช้เปลี่ยนรูปแบบการสั่งการของผู้ใช้ให้กับหุ่นยนต์ โดยสามารถสั่งให้หุ่นยนต์ทำงานในแบบธรรมดา (Manual) หรือการควบคุมตามปกติ และแบบอัตโนมัติ (Autometic) เช่น Line Detection Mode เป็นการสั่งการให้หุ่นยนต์ทำการตรวจสอบและเดินตามเส้นเองโดยอัตโนมัติ

6.2.4 ฟังก์ชันการปรับเปลี่ยนมุมกล้อง :

เป็นฟังก์ชันที่ทำหน้าที่ในการปรับเปลี่ยนมุมกล้องของผู้ใช้ให้อยู่ในทิศทางที่ต้องการ ซึ่งจะทำการรับทิศทางการปรับเปลี่ยนมาจากการกดปุ่มของผู้ใช้ ซึ่งมีอยู่ 8 กรณีคือ cam_up(), cam_down(), cam_left(), cam_right(), cam_up_left(), cam_up_right(), cam_down_left(), cam_down_right() แล้วจึงส่งข้อมูลนั้นไปยังส่วนติดต่อสื่อสาร

6.2.5 ฟังก์ชันการเปิด/ปิดไฟ : cam_light()

เป็นฟังก์ชันที่ช่วยให้ผู้ใช้สามารถทำการเปิดหรือปิดไฟด้านหน้ากล้องของตัวหุ่นยนต์ได้

6.2.6 ฟังก์ชันการเคลื่อนไหวแขน :

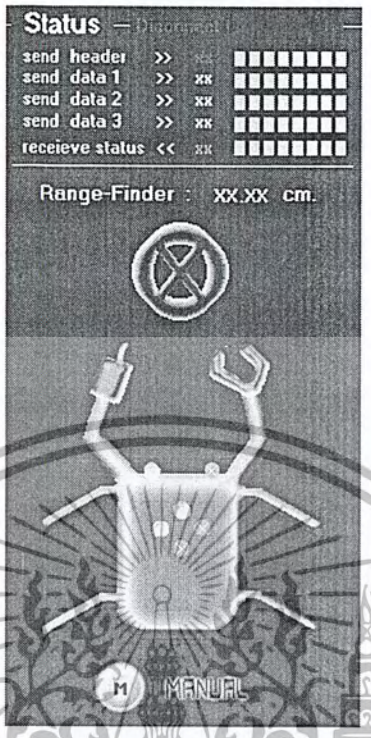
เป็นฟังก์ชันที่ทำหน้าที่ในการเคลื่อนไหวแขนของหุ่นยนต์ โดยจะทำการรับทิศทางการเคลื่อนไหวมาจากการกดปุ่มของผู้ใช้ ซึ่งมีอยู่ 8 กรณีคือ arm_up(), arm_down(), arm_left(), arm_right(), arm_up_left(), arm_up_right(), arm_down_left(), arm_down_right() แล้วจึงส่งข้อมูลนั้นไปยังส่วนติดต่อสื่อสาร

6.2.7 ฟังก์ชันการเคลื่อนไหวมือ :

เป็นฟังก์ชันที่ทำหน้าที่ในการเคลื่อนไหวมือของหุ่นยนต์ โดยจะทำการรับทิศทางการเคลื่อนไหวมาจากการกดปุ่มของผู้ใช้ ซึ่งมีอยู่ 4 กรณีคือ hand_grab(), arm_release(), arm_rotate_left(), arm_rotate_right() แล้วจึงส่งข้อมูลนั้นไปยังส่วนติดต่อสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 ส่วนตรวจสอบการทำงานของหุ่นยนต์



รูปที่ 6-5 แสดงการแสดงผลสถานะของหุ่นยนต์

6.3.1 ฟังก์ชันตรวจสอบการทำงานของเซนเซอร์: sensor_detect()

เป็นฟังก์ชันที่ใช้ตรวจสอบการทำงานของเซนเซอร์ว่ามีการเปลี่ยนแปลงหรือไม่ อย่างไร โดยทำการรอรับข้อมูลจากส่วนติดต่อสื่อสาร เช่น หากมีการรับข้อมูลว่าเซนเซอร์ที่มือมีการเปลี่ยนแปลงแสดงว่ามีบางสิ่งเกิดขึ้นที่มือ อาจจะสามารถหยิบจับสิ่งของได้แล้ว หรืออาจจำของที่หยิบอยู่นั้นตกพื้น เป็นต้น

6.3.2 ฟังก์ชันตรวจสอบทิศทางการเคลื่อนที่ : move_detect()

เป็นฟังก์ชันที่ทำการรับข้อมูลจากส่วนติดต่อสื่อสารว่าขณะนี้หุ่นยนต์มีการเคลื่อนที่จริงไปในทิศทางใด ซึ่งอาจเหมือนหรือแตกต่างจากคำสั่งที่ได้รับ เช่น หากหุ่นยนต์ไม่สามารถเดินหน้าต่อไปได้เนื่องจากพบสิ่งกีดขวางอยู่ ก็จะทำการหลบหลีกหรือหยุดแล้วส่งสถานะกลับมาให้ผู้ควบคุมทราบ เป็นต้น

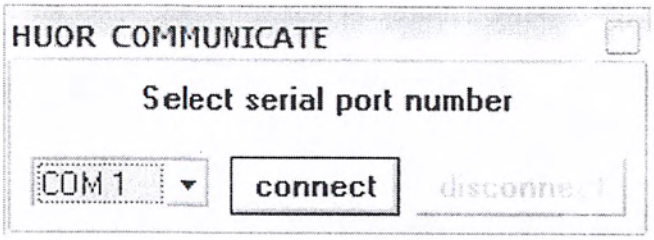
6.4 ส่วนแสดงหน้าต่างการใช้งานของผู้ใช้

ในส่วนนี้เป็นส่วนที่ทำหน้าที่จัดการการแสดงผลทางหน้าจอของโปรแกรม ว่าผู้ใช้ต้องการให้มีหน้าต่างใดขึ้นมาบ้าง และไม่ให้หน้าต่างใดขึ้นมาบ้าง เช่นหน้าต่างแสดงความช่วยเหลือในการใช้คีย์บอร์ด หน้าต่างการติดต่อสื่อสาร เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.5 ส่วนติดต่อสื่อสาร

ในส่วนนี้จะเป็นส่วนที่คอยดูแลการติดต่อสื่อสารรวมถึงการรับส่งข้อมูล ซึ่งมีฟังก์ชันการทำงานดังนี้



รูปที่ 6-6 แสดงหน้าต่างการติดต่อสื่อสาร

6.5.1 ฟังก์ชันเริ่มการเชื่อมต่อ : connect()

เป็นฟังก์ชันที่ทำหน้าที่เริ่มการติดต่อสื่อสารกับพอร์ตอนุกรม ทำการตั้งค่าต่าง ๆ เลือกหมายเลขพอร์ตที่ต้องการใช้งาน เป็นต้น

6.5.2 ฟังก์ชันยกเลิกการเชื่อมต่อ : disconnect()

เป็นฟังก์ชันที่ทำหน้าที่ยกเลิกการติดต่อสื่อสารกับพอร์ตอนุกรม

6.5.3 ฟังก์ชันการส่งข้อมูล : send_data()

เป็นฟังก์ชันที่ทำหน้าที่ควบคุมการส่งข้อมูลให้กับหุ่นยนต์ โดยรับข้อมูลมาจากส่วนการควบคุมการทำงาน

6.5.4 ฟังก์ชันการรับข้อมูล : receieve_data()

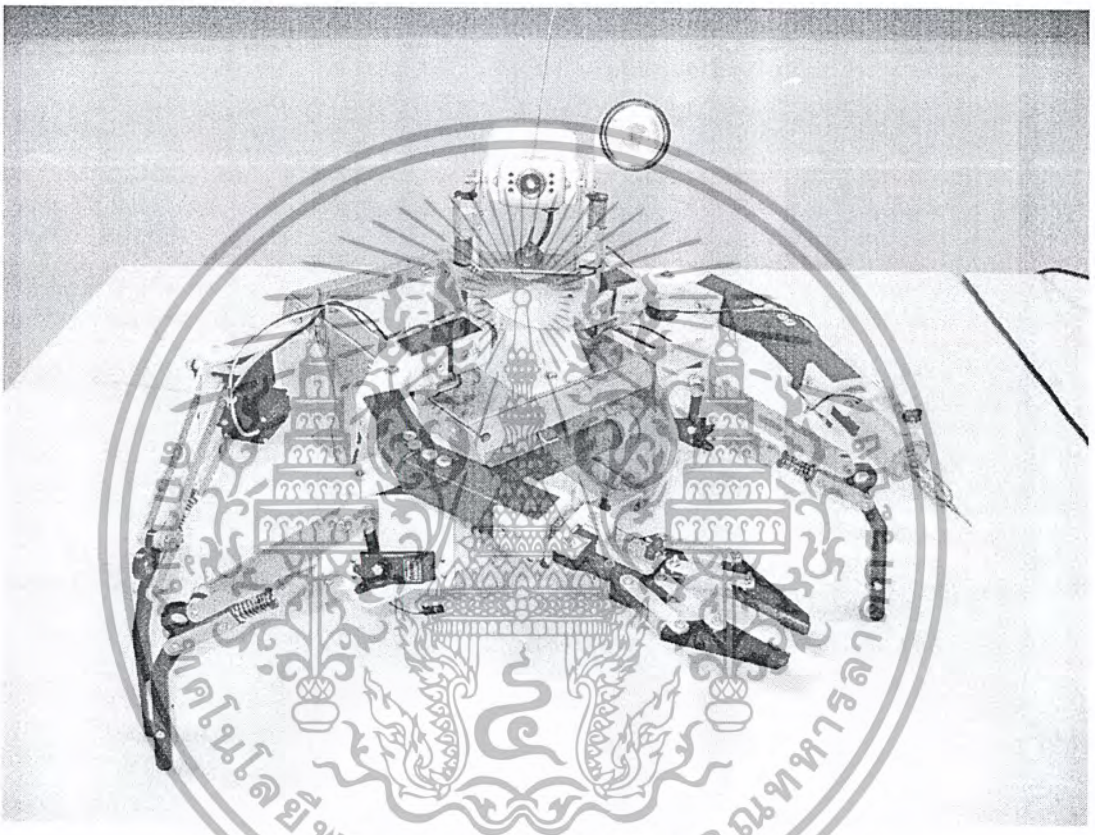
เป็นฟังก์ชันที่ทำหน้าที่รอรับข้อมูลจากหุ่นยนต์ แล้วนำข้อมูลที่ได้ส่งให้กับส่วนตรวจสอบการทำงานของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

ผลการทดสอบ

จากการดำเนินงานที่ผ่านมา ตั้งแต่การวางแผน การศึกษาหาข้อมูล การออกแบบ จน กระทั่งการจัดสร้างเป็นชิ้นงานตามที่ต้องการ ทำให้เราได้ HUOR ที่มีความพร้อมในการที่จะทำการทดสอบว่าสามารถทำงานได้ตามที่คาดคิดไว้หรือไม่ โดยในบทนี้จะกล่าวถึงการทดสอบการทำงานของ HUOR



รูปที่ 7-1 แสดงภาพถ่ายโครงสร้างและส่วนประกอบของ HUOR

ส่วนประกอบโครงสร้างของ HUOR โดยส่วนใหญ่ได้เลือกใช้อลูมีเนียมที่ทำการตัดแต่งขึ้นรูปตามแบบที่ออกแบบไว้ เหตุผลที่เลือกใช้อลูมีเนียมเพราะอลูมีเนียมมีความแข็งแรงและมีน้ำหนักเบา แต่ก็มีบางส่วนที่เลือกใช้พลาสติกแทนเพราะเป็นส่วนที่มีความละเอียด ซึ่งหากใช้อลูมีเนียมจะเป็นการเพิ่มความยากลำบากในการทำงาน นอกจากนี้ยังมีการใช้อุปกรณ์ต่าง ๆ ดังตารางด้านล่างเป็นส่วนประกอบอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับ	อุปกรณ์	จำนวน	ส่วนประกอบ
1	กล้องวงจรปิดไร้สาย	1 ตัว	ระบบการมองเห็น
2	หลอดไฟฉาย	1 ชุด	ระบบการมองเห็น
3	มอเตอร์กระแสตรง 3-9V	14 ตัว	ระบบการมองเห็น (2), ระบบแขนกล (8), ระบบการเคลื่อนที่ (4)
4	เซอร์โวมอเตอร์มาตรฐาน (3 กก.)	9 ตัว	ระบบการเคลื่อนที่
5	อินฟราเรดเซนเซอร์ ตัวรับและส่ง	12 คู่	ระบบแขนกล (2), ระบบการเคลื่อนที่ (4), บนตัวหุ่นยนต์ (6)

ตารางที่ 7-1 แสดงรายการอุปกรณ์ที่ใช้

ในการทดสอบการทำงานของส่วนประกอบต่าง ๆ นั้นได้แบ่งออกเป็น 2 ส่วนคือ การทดสอบการทำงานเฉพาะส่วนโครงสร้าง และ การทดสอบการทำงานจากวงจรควบคุม ซึ่งมีรายละเอียดดังนี้

7.1. การทดสอบการทำงานเฉพาะส่วนโครงสร้าง

ในการทดสอบการทำงานของโครงสร้างของ HUOR นั้นทำการทดสอบโดยการจ่ายกระแสไฟอุปกรณ์โดยตรงเพื่อดูผลการทำงานของส่วนต่าง ๆ ว่าเป็นไปตามที่ต้องการหรือไม่ หากพบว่ามีการทำงานไม่ถูกต้องก็จะต้องทำการปรับปรุงแก้ไขให้มีการทำงานให้ถูกต้องยิ่งขึ้น ซึ่งปัญหาส่วนใหญ่ที่พบนั้นเกิดจากการออกแบบระบบเครื่องกล (Mechanic) และการเลือกใช้วัสดุให้เกิดความเหมาะสมซึ่งไม่สามารถคาดการณ์ผลลัพธ์ได้ล่วงหน้าหากแต่ได้คาดคะเนไว้เท่านั้น จึงต้องมีการแก้ไขและออกแบบโครงสร้างใหม่อยู่เสมอ โดยผลลัพธ์ที่ได้นั้นอาจจะไม่ใช่การทำงานที่ให้มีความแม่นยำ 100 เปอร์เซ็นต์ แต่สุดท้ายแล้วก็ได้ใช้ความพยายามในการปรับแต่งให้เกิดผลลัพธ์ในระดับที่ยอมรับได้ สำหรับรายละเอียดจากการทดสอบมีดังต่อนี้

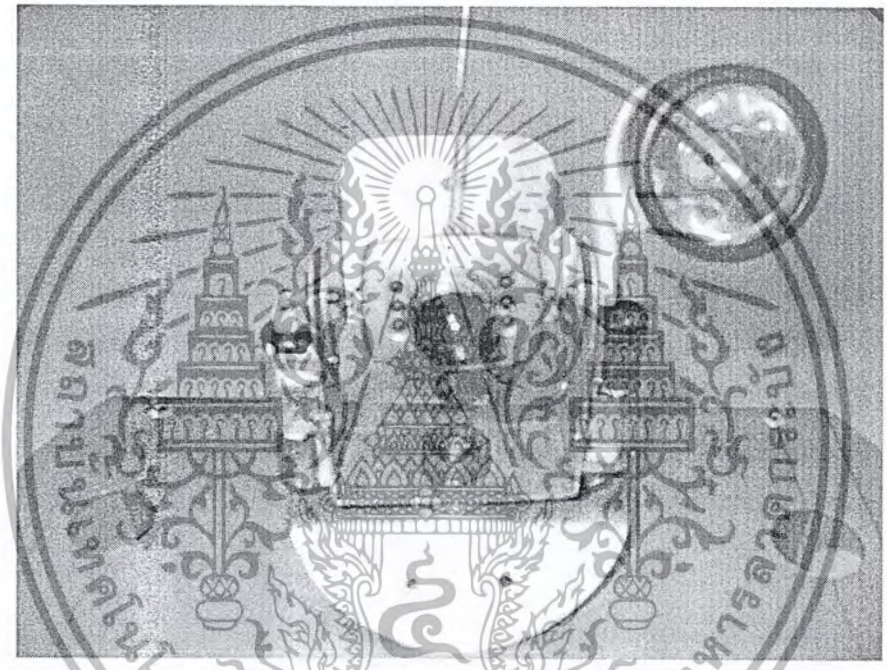
7.1.1 ทดสอบการทำงานของกล้องวงจรปิดไร้สาย

ทำการทดสอบโดยการจ่ายกระแสไฟ 8 โวลต์จากหม้อแปลง (Adapter) ที่ได้มากับตัวกล้องพบว่าการทำงานเป็นไปตามปกติ สามารถใช้การ์ดวีดีโอแคปเจอร์ (Video Capture Card) ยี่ห้อ V-Gear ร่วมกับโปรแกรม Cyberlink PowerVCR II เพื่อรับสัญญาณภาพแสดงออกมาทางหน้าจอโปรแกรม หลังจากนั้นนำกล้องวงจรปิดออกห่างจากตัวรับสัญญาณจนกว่าสัญญาณจะขาดหายไป เมื่อวัดระยะทางแบบกระจัดจะได้ไกลสุดประมาณ 30 เมตร กล้องวงจรปิดนี้ใช้คลื่นไมโครเวฟในการรับ-ส่งสัญญาณจึงสามารถทะลุทะลวงสิ่งกีดขวางได้บ้าง สรุปได้ว่ากล้องวงจรปิดไร้สายสามารถทำงานได้ดีมากตามที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.2 ทดสอบการทำงานส่วนระบบการมองเห็น

ทำการทดสอบโดยการจ่ายกระแสไฟ 5 โวลต์ ให้กับมอเตอร์ที่ติดอยู่ทั้ง 2 แกน คือ แนวระนาบ และแนวดิ่ง พบว่าในครั้งแรกการทำงานไม่เป็นไปตามที่ต้องการเนื่องจากการมอเตอร์ที่ใช้นั้นมีรอบสูง ทำให้การเคลื่อนไหวมีความเร็วและรุนแรงเกินไป ซึ่งยากต่อการควบคุมให้อยู่ในตำแหน่งที่ต้องการและอาจทำให้ผู้ใช้เกิดอาการเวียนหัวขณะควบคุมหุ่นยนต์ได้ จึงทำการเปลี่ยนมอเตอร์ให้เป็นรอบต่ำ เมื่อทำการทดสอบ ปัญหานี้จึงหมดไป อีกปัญหาหนึ่งที่พบคือการเคลื่อนไหวในแนวระนาบ (แกน X) นั้นต้องใช้มอเตอร์ในการหมุนสายพาน ซึ่งมีขนาดใหญ่เกินไป ทำให้บางครั้งมอเตอร์ทำการหมุนเฉพาะสายพานแต่ฐานที่ติดกับกล้องไม่หมุนตาม ทำการแก้ไขโดยการหาซื้อสายพานที่มีขนาดพอเหมาะ เมื่อทดสอบอีกครั้งได้ผลการทำงานตรงตามความตั้งใจ

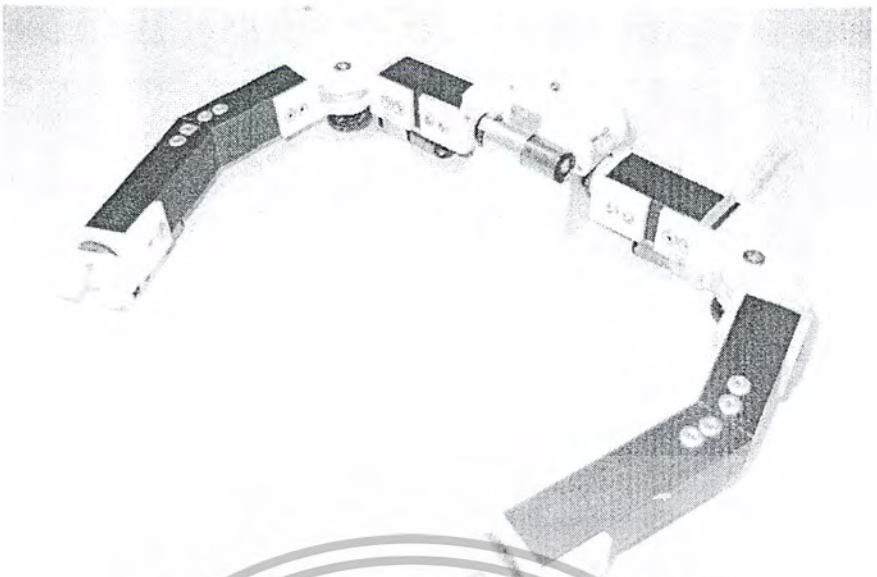


รูปที่ 7-2 แสดงภาพถ่ายระบบการมองเห็น

7.1.3 ทดสอบการทำงานส่วนระบบแขนกล

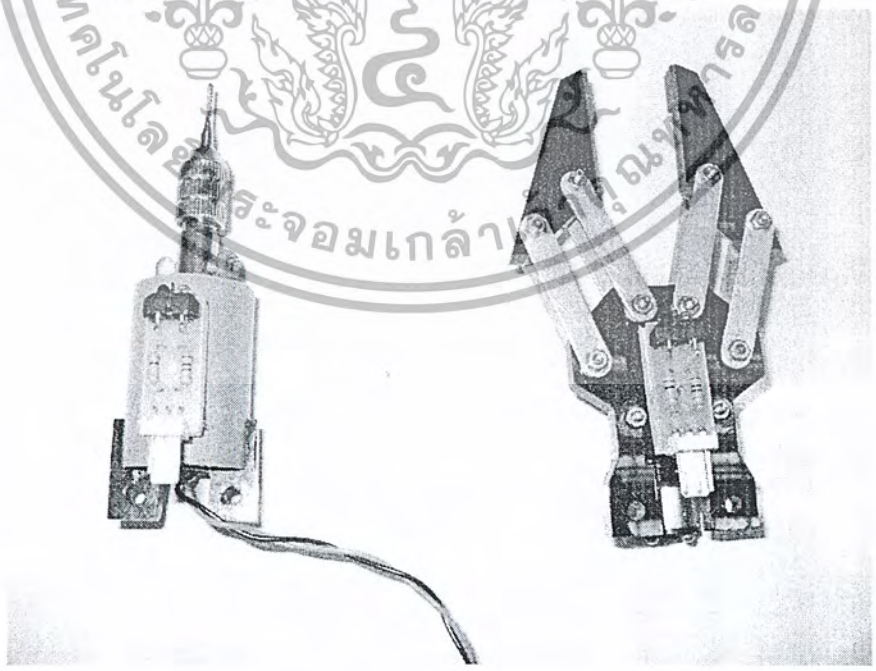
ทำการทดสอบโดยการจ่ายไฟ 9 โวลต์ให้กับมอเตอร์แต่ละตัว พบว่ามีปัญหาจากการทำงานของแขนทั้งสองข้างเกิดขึ้น ซึ่งเป็นปัญหาจากการออกแบบ พบว่าส่วนหัวไหล่และข้อศอกไม่มีความมั่นคงในการเคลื่อนไหว เนื่องจากโครงสร้างไม่แข็งแรงพอ และไม่สามารถที่จะรับน้ำหนักแขนได้ เนื่องจากการทดกำลังจากมอเตอร์ยังทำได้ไม่ดี ซึ่งทำการแก้ไขโดยการแก้ไขแบบจุดหมุนที่ใช้ในการเคลื่อนไหว ทั้งในส่วนหัวไหล่และข้อศอกและออกแบบระบบการทดกำลังของหัวไหล่ใหม่ เมื่อนำมาทดสอบการทำงานอีกครั้งก็พบว่าผลที่ได้เป็นที่พอใจมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7-3 แสดงภาพถ่ายระบบแขนกล

สำหรับการทดสอบการทำงานของมือคีบนั้นก็มีปัญหาเกิดขึ้นเช่นกัน เนื่องจากส่วนประกอบของมือคีบ มีความละเอียดอ่อนและต้องการความแน่นอนในการทำงาน แต่หลังจากที่มีการปรับแต่งจุดเชื่อมต่อ จุดหมุน และการยึดเฟืองกับมอเตอร์แล้ว พบว่าการทำงานเป็นไปตามที่ต้องการ ซึ่งการเคลื่อนไหวหรือการหยิบจับวัตถุนั้น มือที่สร้างขึ้นยังมีกำลังน้อยจึงไม่สามารถจับยกวัตถุที่มีน้ำหนักมากได้ และการทำงานบางครั้งอาจจะมีความไม่แน่นอนเกิดขึ้นบ้าง แต่ในภาพรวมก็ยังสามารถทำงานได้เป็นปกติ

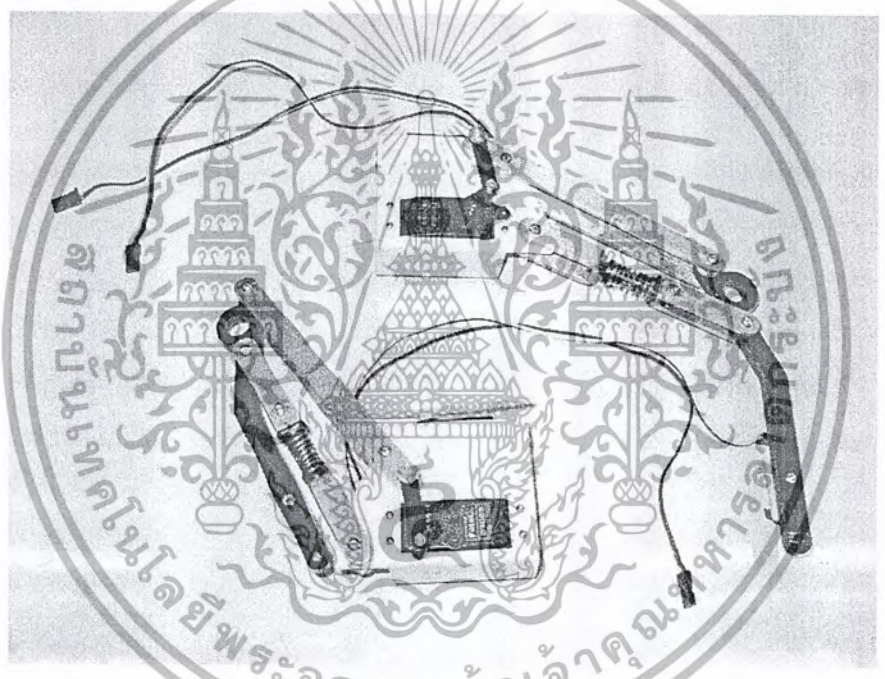


รูปที่ 7-4 แสดงภาพถ่ายลักษณะของมือทั้ง 2 ข้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.4 ทดสอบการทำงานส่วนขา

ในการทดสอบการทำงานของขาครั้งแรกนั้นพบว่า HUOR ไม่สามารถยืนได้ตามที่ต้องการ เนื่องจากว่าได้ใช้มอเตอร์กระแสตรงธรรมดาที่สามารถรับแรงได้มากสุดที่ 1 กิโลกรัม ซึ่งไม่เพียงพอต่อการรับน้ำหนักทั้งหมดของตัวหุ่นยนต์ จึงต้องทำการปรับเปลี่ยนแบบใหม่ทั้งหมดเพื่อรองรับกับเซอร์โวมอเตอร์ที่สามารถรับแรงได้ 3 กิโลกรัม เมื่อทำการทดสอบพบว่าขาที่สร้างใหม่สามารถรับน้ำหนักได้ดี สำหรับการทดสอบนั้นเนื่องจากว่าขานั้นแต่ละข้างใช้เซอร์โวมอเตอร์ในการทำงาน ทั้งการเคลื่อนที่ในการยกและกวาดขา ซึ่งต้องการสัญญาณพัลส์ เพื่อกำหนดตำแหน่งในการหมุนให้กับเซอร์โวมอเตอร์ ดังนั้นเราจึงต้องทำการเขียนโปรแกรมให้กับไมโครคอนโทรลเลอร์ MCS-51 เพื่อให้กำเนิดสัญญาณพัลส์ที่จะใช้ในการทดสอบ จากนั้นจึงทำการจ่ายไฟ 5 โวลต์ให้กับเซอร์โวมอเตอร์ พบว่าขาแต่ละข้างสามารถเคลื่อนไหวได้ตามที่ต้องการและได้ทำการแก้ไขจุดหมุนให้มีความแข็งแรงมากขึ้น เพื่อป้องกันการผิดพลาดที่อาจเกิดขึ้นเมื่อนำไปใช้งานจริง



รูปที่ 7-5 แสดงภาพถ่ายลักษณะของขา

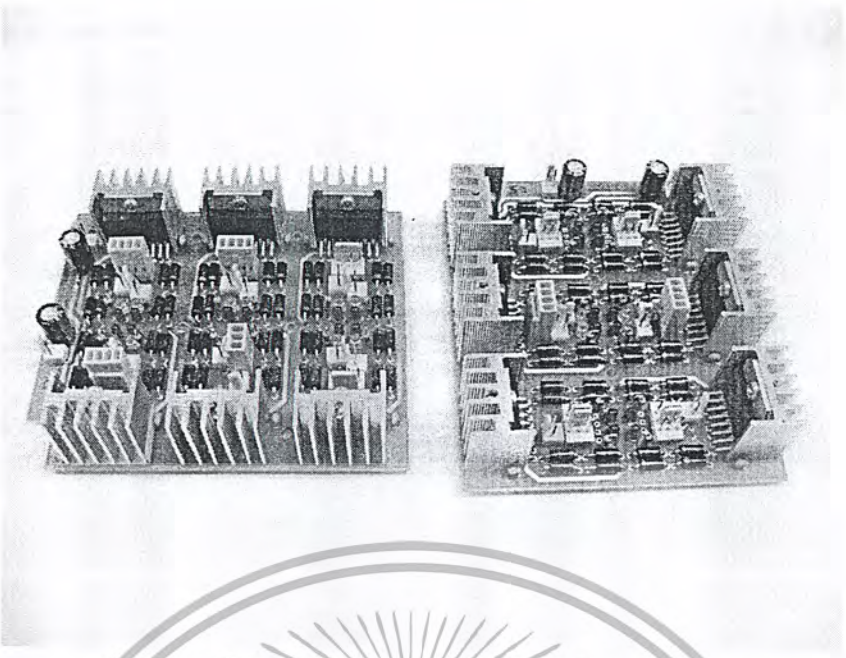
7.2 การทดสอบการทำงานวงจรควบคุม

การทดสอบการทำงานของวงจรควบคุมนั้นมีวิธีการที่แตกต่างกันตามลักษณะการทำงานของแต่ละวงจร ดังนี้

7.2.1 ผลทดสอบการทำงานของวงจรขับมอเตอร์

ทำการทดสอบโดยการจ่ายไฟ 5 โวลต์และ 9 โวลต์ให้กับวงจร แล้วนำมอเตอร์มาต่อกับคอนเนคเตอร์เอาท์พุท จากนั้นนำสายสัญญาณไฟ 5 โวลต์มาใช้เป็นอินพุทเปรียบเสมือนลอจิก 1 ต่อเข้ากับคอนเนคเตอร์อินพุท สังเกตการหมุนของมอเตอร์ แล้วทำเช่นเดียวกันนี้กับคอนเนคเตอร์ทุกชุด พบว่ามอเตอร์สามารถทำงานได้ตามที่ต้องการกับวงจรขับมอเตอร์ทั้ง 2 บอร์ด

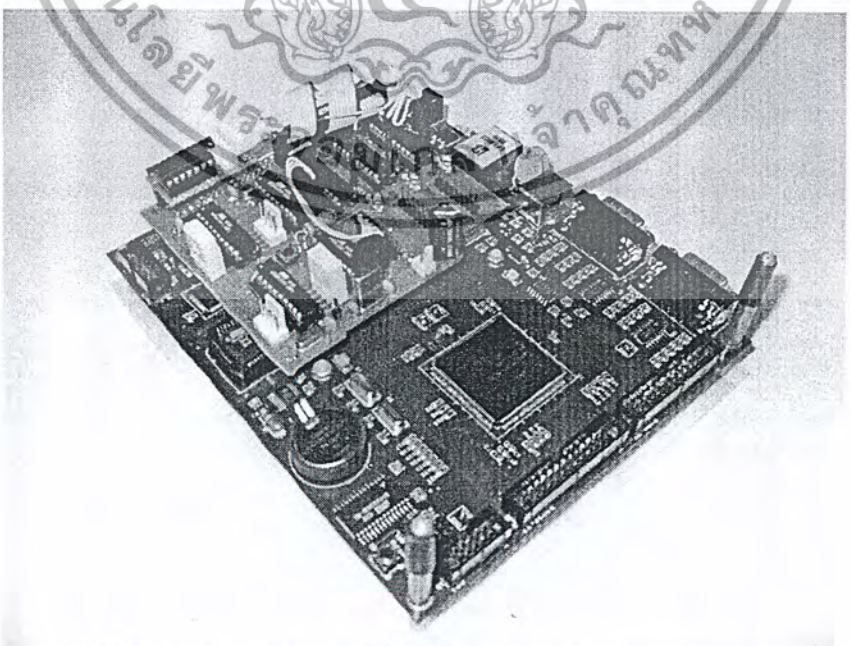
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7-6 แสดงภาพถ่ายวงจรขับมอเตอร์

7.2.2 ผลทดสอบการทำงานของบอร์ดไอซา

ทำการทดสอบโดยการนำบอร์ดไอซา ไปต่อกับชุดพัฒนาคอม 86 แล้วใช้โปรแกรมดีบัก (Debug) ทดสอบการส่งข้อมูลในแอดเดรสที่ต่างกัน ซึ่งทางผู้พัฒนาได้ทำบอร์ดทดสอบพิเศษขึ้นมาเพื่อใช้ดูผลข้อมูลที่ได้ทางแอลอีดี (LED) ซึ่งจากการทดสอบนั้นบอร์ดไอซาสามารถทำงานได้ตามที่ต้องการ วงจรสามารถแยกแอดเดรสให้แลตช์ข้อมูลแต่ละตัวทำงานหรือไม่ทำงานได้ และข้อมูลที่ได้ในแต่ละพอร์ตเป็นข้อมูลที่ถูกต้อง หลังจากนั้นได้ทำการทดสอบส่งข้อมูลไปเป็นอินพุทให้กับวงจรขับมอเตอร์พบว่ามอเตอร์สามารถทำงานได้ตามปกติ

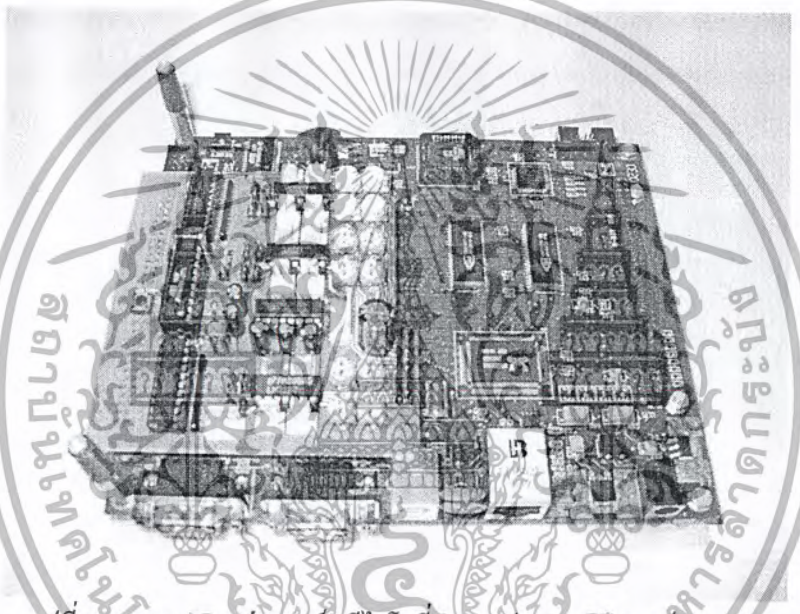


รูปที่ 7-7 แสดงภาพถ่ายบอร์ดไอซาที่ทำการประกอบกับบอร์ดคอม86

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.3 ผลทดสอบการทำงานของบอร์ดพีไอโอ

ทำการทดสอบโดยการนำบอร์ดพีไอโอต่อเข้ากับชุดพัฒนาคอม 86 แล้วทำการเขียนโปรแกรมทดสอบรับค่าอินพุทจากเซนเซอร์ที่ต่ออยู่กับคอนเนคเตอร์บนบอร์ด พบว่าผลที่ได้นั้นเกิดความผิดพลาดในทุกจุด แต่เมื่อนำบอร์ดพีไอโอมาทำการทดสอบแยกจากชุดพัฒนาคอม 86 พบว่าวงจรสามารถทำงานได้ตามปกติโดยดูการแสดงผลจากหลอดแอลอีดีที่อยู่บนบอร์ด จึงตั้งข้อสันนิษฐานว่ากระแสนั้นหายไปจากการแสดงผลของหลอดแอลอีดี เมื่อใช้มัลติมิเตอร์ทำการวัดความต่างศักดิ์ของขาเอาท์พุทเทียบกับกราวด์ของวงจรแล้วได้ค่าเพียง 2.5 โวลต์ ซึ่งควรจะเป็น 5 โวลต์ ดังนั้นจึงทำการถอดหลอดแอลอีดีแสดงผลออกทั้งหมดแล้วทำการทดสอบร่วมกับชุดพัฒนาคอม 86 อีกครั้งพบว่าสามารถทำงานได้ตามปกติ จึงทำการดัดแปลงบอร์ดโดยทำคอนเนคเตอร์สำหรับหลอดแอลอีดีแสดงผลไว้ เพื่อใช้ในการปรับระยะการตรวจสอบของเซนเซอร์แต่ละตัว ดังรูป



รูปที่ 7-8 แสดงภาพถ่ายบอร์ดพีไอโอที่ทำการประกอบกับบอร์ดคอม86

7.2.4 ผลทดสอบการทำงานของวงจรรับ-ส่งสัญญาณวิทยุ

ทำการทดสอบโดยการรับ-ส่งข้อมูลผ่านโปรแกรมไฮเปอร์เทอร์มินัล (Hyper Terminal) ทางพอร์ตอนุกรม ซึ่งมีไมโครคอนโทรลเลอร์ MCS-51 เป็นตัวกลางระหว่างวงจรรับ-ส่งสัญญาณวิทยุ เพื่อทำการควบคุมอัตราการส่งข้อมูล และแปลงข้อมูลจากอนุกรมไปสู่แบบขนานสำหรับฝ่ายส่งและข้อมูลแบบขนานไปสู่แบบอนุกรมสำหรับฝ่ายรับ ในการทดลองวงจรรับ-ส่งข้อมูลได้จัดทำเป็น 3 ขั้นตอน ดังนี้

ขั้นที่ 1 ทำการวนส่งข้อมูลแบบทิศทางเดียว (Simplex) ครั้งละ 1 บิตไปเรื่อย ๆ โดยแสดงผลออกทางหน้าจอโปรแกรมไฮเปอร์เทอร์มินัล (Hyper Terminal) เพื่อดูความถูกต้อง และหาค่าหน่วงเวลา (Delay) ที่น้อยที่สุดที่ไม่ทำให้ข้อมูลเกิดความผิดพลาด

ผลที่ได้ จากการทดลองทำให้ได้ค่าหน่วงเวลา ที่น้อยที่สุดที่ไม่ทำให้ข้อมูลผิดพลาด แต่ความเร็วในการรับ-ส่งข้อมูลที่ได้ยังไม่เป็นที่น่าพอใจ

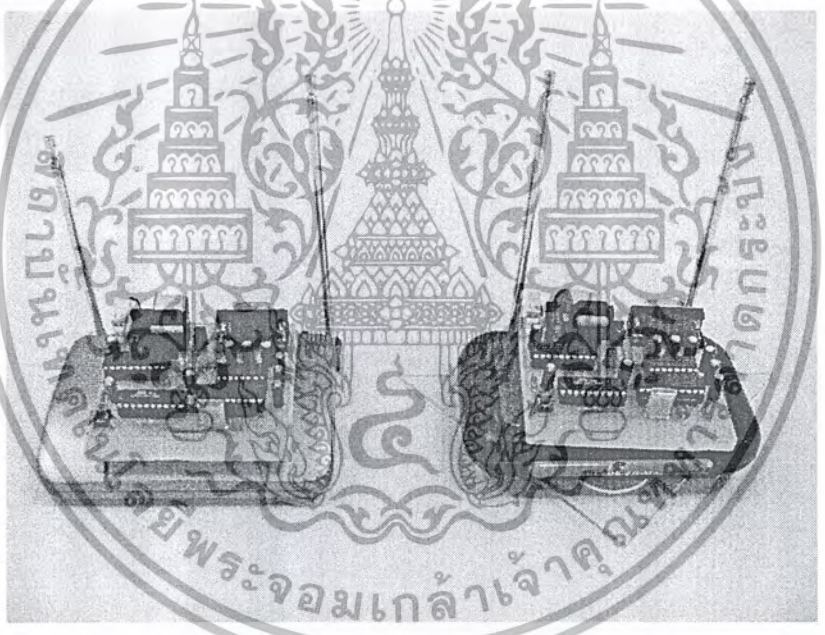
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่ 2 ทำการส่งข้อมูลแบบทิศทางเดียว (Simplex) โดยให้ตัวส่ง รับอินพุทจาก user ทางโปรแกรม Hyper Terminal แล้วทำการส่งข้อมูลครั้งละ 1 byte ตามการทำงานของโปรโตคอล แล้วให้ตัวรับแสดงผลตัวอักษรที่ได้รับทางหน้าจอโปรแกรมไฮเปอร์เทอร์มินัล

ผลที่ได้ สามารถรับ-ส่งข้อมูลที่ถูกต้องได้เป็นที่พอใจ โดยข้อมูลมีความถูกต้องมากกว่า 90% (ใช้ค่าหน่วงเวลา จากการทดลองขั้นที่ 1)

ขั้นที่ 3 ทำการทดสอบการทำงานแบบสองทิศทางแต่ต่างเวลากัน (Half – duplex) เต็มรูปแบบ โดยให้มีการสถาปนาการเชื่อมต่อระหว่างกันก่อน และระหว่างการส่งข้อมูลให้มีการ acknowledge ถึงกัน รวมถึงสามารถทำการตรวจสอบได้ว่าข้อมูลที่ได้รับนั้นผิดพลาดหรือไม่

ผลที่ได้ ไม่สามารถทำงานได้ตามที่ต้องการ เนื่องจากความสามารถที่จำกัดของวงจรรับ-ส่งข้อมูล และโปรโตคอลมีความซับซ้อนมากเกินไป



รูปที่ 7-9 แสดงภาพถ่ายชุดวงจรรับ-ส่งข้อมูลผ่านคลื่นวิทยุ

จากผลการทดลองเราสามารถสรุปได้ว่าชุดวงจรรับ-ส่งข้อมูลผ่านคลื่นวิทยุที่ทำการทดสอบนี้ไม่เหมาะสมต่อการใช้งานที่ซับซ้อนของโปรโตคอลที่คิดขึ้น เนื่องจากมีอัตราเร็วในการส่งข้อมูลที่ต่ำเกินไป และข้อมูลยังขาดความน่าเชื่อถืออยู่มาก ซึ่งเราสามารถทำการแก้ไขได้โดยการหาวงจรรับ-ส่งข้อมูลชุดใหม่ที่มีประสิทธิภาพมากกว่า ซึ่งจากการศึกษาเพิ่มเติมพบว่ามีวงจรที่เหมาะสมต่อการใช้งาน แต่มีราคาที่ค่อนข้างสูง ดังนั้นหุ่นยนต์ HUOR ต้นแบบนี้จะใช้การติดต่อรับ-ส่งข้อมูลผ่านสายสื่อสารอนุกรม ซึ่งสามารถพัฒนาต่อให้เป็นหุ่นยนต์ไร้สายเต็มตัวโดยการเปลี่ยนตัวกลางการนำส่งข้อมูลจากสายสื่อสารเป็นตัวกลางไร้สายอื่นแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8
บทวิจารณ์และสรุป

การทำงานต่าง ๆ ของมนุษย์นั้นมีบางสภาวะที่ไม่เอื้อให้การปฏิบัติงานของมนุษย์ดำเนินไปได้ โดยเฉพาะในเรื่องของสรีระร่างกาย การพัฒนาหุ่นยนต์ภาคสนามในประเทศไทยนั้นยังมีน้อย และส่วนใหญ่ไม่ได้รับการพัฒนาส่งเสริมเพิ่มเติมขีดความสามารถให้สูงขึ้นอย่างต่อเนื่อง รวมทั้งหุ่นยนต์ที่พัฒนาขึ้นนั้นก็ทำงานได้เฉพาะอย่าง ทำให้ความรู้และและความเชี่ยวชาญในเรื่องของหุ่นยนต์ภาคสนามยังจำกัดอยู่เฉพาะบางกลุ่มเท่านั้น

จากการศึกษาค้นคว้าตลอดจนทำการสร้างหุ่นยนต์ต้นแบบ HUOR ขึ้นมาในการทำงานด้านโครงสร้างการเคลื่อนไหวของส่วนแขนและมือนั้นได้รับผลเป็นที่น่าพอใจ ปัญหาที่พบส่วนใหญ่เป็นผลมาจากส่วนประกอบทางกล ส่วนการทำงานของขาเรื่องของการทรงตัวยังไม่ดีนัก เนื่องจากการกลไกการเคลื่อนไหวที่ออกแบบนั้นเป็นกลไกแบบง่าย ทำให้การทรงตัวทำได้ยาก ส่วนสุดท้ายคือเรื่องการสื่อสารแบบไร้สายผ่านสัญญาณวิทยุ เนื่องจากผู้วิจัยได้ทำการเลือกใช้วงจรรถกระป๋องซึ่งมีแบนด์วิดท์ต่ำมาก ทำให้ส่งข้อมูลได้ช้า อีกทั้งความถี่และกำลังส่งน้อย ส่งผลให้ข้อมูลที่ได้รับนั้นมีอัตราความผิดพลาดสูง หุ่นยนต์ HUOR ต้นแบบนี้จึงยังไม่สามารถทำงานแบบไร้สายได้ แต่ทั้งนี้ก็มีแนวทางในการพัฒนาต่อได้หลากหลายวิธีการ

ในการที่จะพัฒนาและสร้างสรรค์หุ่นยนต์ ที่สามารถนำมาแบ่งเบาภาระการทำงานของมนุษย์ลงได้นั้น HUOR ถือได้ว่าเป็นหุ่นยนต์ต้นแบบที่สามารถนำมาเป็นพื้นฐานการทดลองเขียนโปรแกรม หรือพัฒนาปัญญาประดิษฐ์ได้เนื่องจากตัว HUOR นั้นมีความยืดหยุ่นมากทั้งในด้านการเคลื่อนที่ หรือการทำงานของแขนกล รวมทั้งแนวคิดการควบคุมหุ่นยนต์ระยะไกลด้วยอุปกรณ์ไร้สาย ก็ยิ่งเพิ่มขีดความสามารถขึ้นไปอีก สิ่งเหล่านี้เป็นการตอกย้ำว่า HUOR นั้นเป็นจุดเริ่มต้นของการพัฒนาหุ่นยนต์ภาคสนามในประเทศไทยหากได้รับการส่งเสริมที่ดีและต่อเนื่อง

8.1 สรุปความสามารถของหุ่นยนต์ปฏิบัติการเสี่ยงภัยแทนมนุษย์

1. สามารถเคลื่อนที่ได้ด้วยขา 4 ขา
2. สามารถทำการเคลื่อนที่ในแบบกึ่งอัตโนมัติ
3. มีแขนและมือทั้งซ้าย-ขวา และสามารถเคลื่อนที่ได้ทั้ง 2 แกน
4. ผู้ควบคุมปรับเปลี่ยนมุมมองได้ ทั้ง 2 แกน
5. ติดต่อสื่อสารกับผู้ควบคุมได้ โดยผ่านสายสื่อสารอนุกรม
6. ผู้ใช้งานสามารถบันทึกไฟล์รูปภาพและไฟล์วีดีโอได้
7. มีโปรแกรมส่วนติดต่อผู้ใช้ (GUI) ช่วยให้ผู้ใช้งานสะดวกขึ้นสำหรับการควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.2 อุปสรรคและปัญหาที่พบ

1. การออกแบบโครงสร้างในส่วนของกลไกการทำงานต่าง ๆ เช่น ชุดเฟือง ข้อต่อต่างๆ ยังไม่มีความแข็งแรงและแม่นยำเพียงพอที่จะสามารถนำไปใช้ในสถานการณ์จริงได้
2. ระบบการเปลี่ยนโหมดการเคลื่อนที่ ไม่สามารถทำงานได้เนื่องจากระบบคานที่ออกแบบไม่สามารถรับน้ำหนักทั้งหมดของตัวหุ่นยนต์ได้ จึงทำให้ HUOR สามารถเคลื่อนที่ได้เพียงโหมดการเดินด้วยขาเท่านั้น
3. ชุดพัฒนาคอม 86 ยังไม่มีความเสถียรมากพอในการทดสอบโปรแกรมที่พัฒนาขึ้น บางครั้งจึงเกิดข้อผิดพลาดขึ้น แต่มาจากปัญหาทางฮาร์ดแวร์ไม่ใช่ซอร์ฟแวร์
4. การรับ-ส่งข้อมูลจำนวนมาก ๆ ผ่านชุดรับส่งไร้สายไม่สามารถทำได้ เนื่องจากข้อจำกัดทางด้านฮาร์ดแวร์ที่เลือกใช้
5. วงจรการทำงานต่าง ๆ เมื่อนำมาเชื่อมต่อเข้าด้วยกันทุกส่วนแล้ว เกิดการรบกวนกันของสัญญาณทำให้การทำงานมีความผิดพลาด

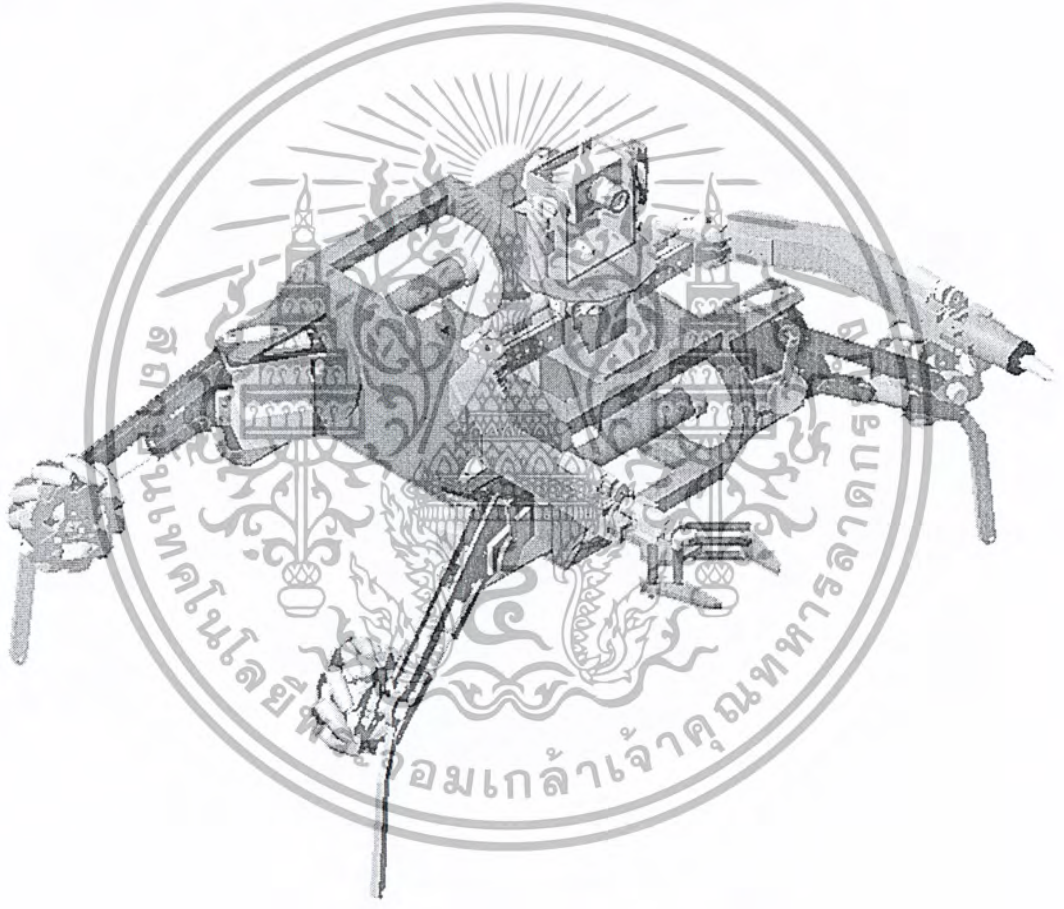
8.3 แนวทางการพัฒนาต่อ

1. ทำการพัฒนาออกแบบในส่วนของระบบการเปลี่ยนโหมดการเคลื่อนที่ให้รับน้ำหนักด้วยชุดเฟืองแทนการใช้ระบบคาน
2. ทำการออกแบบวงจรรับ-ส่งข้อมูลผ่านสัญญาณวิทยุที่มีแบนด์วิดท์สูง และมีความน่าเชื่อถือเพื่อให้หุ่นยนต์สามารถทำงานในแบบไร้สายได้
3. การพัฒนาโปรแกรมการทำ Image Processing ของหุ่นรักษาความปลอดภัย
4. ทำการพัฒนาโปรแกรมทางด้านปัญญาประดิษฐ์(Artificial Intelligent)สำหรับหุ่นยนต์เดิน 4 ขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.
รายละเอียดแบบโครงสร้างหุ่นยนต์

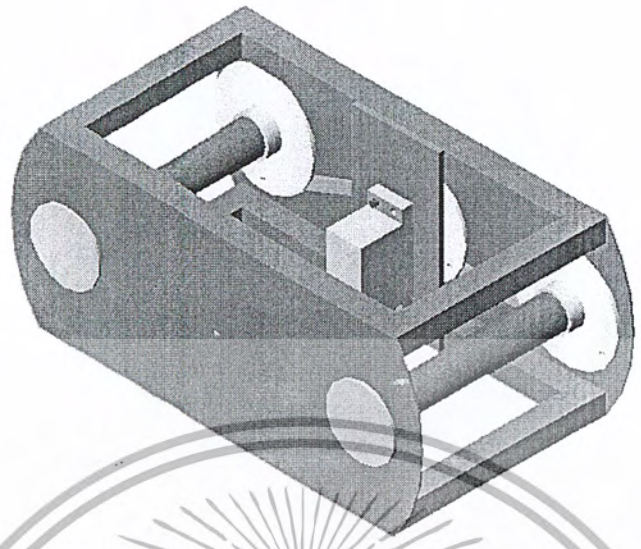
ในการออกแบบ ได้ใช้โปรแกรม AutoCAD 2002 เป็นเครื่องมือในการออกแบบ เนื่องจากเป็นโปรแกรมที่ใช้งานสะดวก มีเครื่องมือเพียงพอกับความต้องการและสามารถศึกษาค้นคว้าหาข้อมูลเพิ่มเติมได้ง่าย ในบทนี้จะทำการแสดงภาพรายละเอียดของส่วนประกอบต่าง ๆ โดยในแต่ละส่วนประกอบจะแสดงภาพทั้งแบบ 3 มิติ (3D) และภาพแบบโครงร่าง (Layout) เพื่อให้ผู้ที่ต้องการศึกษาโครงสร้างของหุ่นยนต์สามารถนำแนวคิกในการออกแบบไปประยุกต์ใช้ให้มีความเหมาะสมต่อไป



รูปที่ ก-1 แสดงแบบโครงสร้างหุ่นยนต์ 3 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. แบบลำตัวหุ่นยนต์

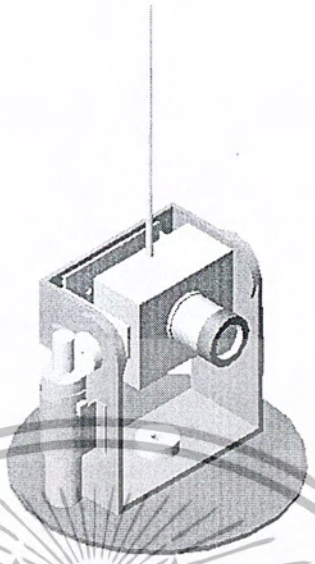


รูปที่ ก-2 แสดงแบบลำตัวหุ่นยนต์ 3 มิติ

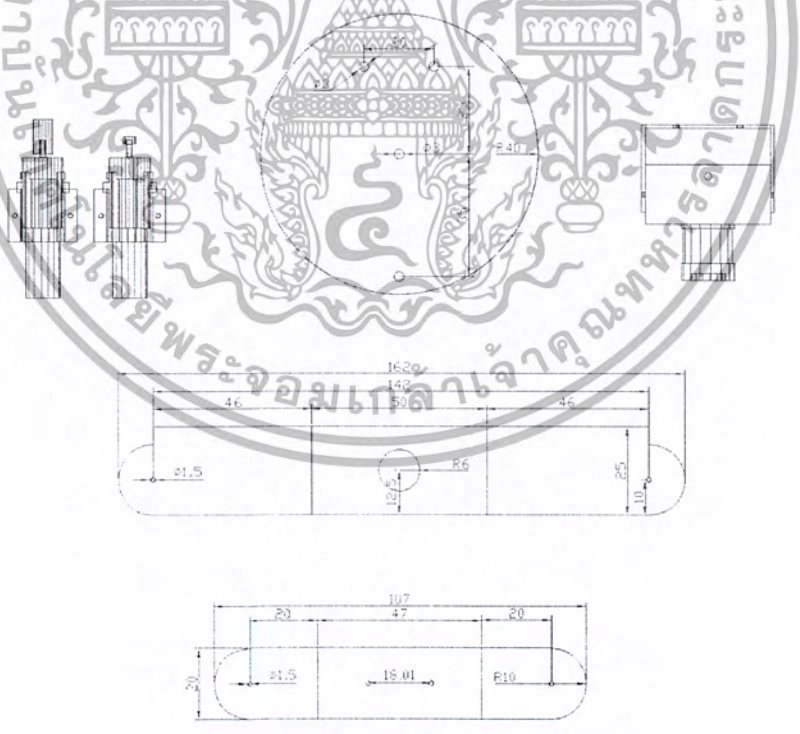
รูปที่ ก-3 แสดงแบบโครงร่างลำตัวหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. แบบยึดกล้องวงจรปิด



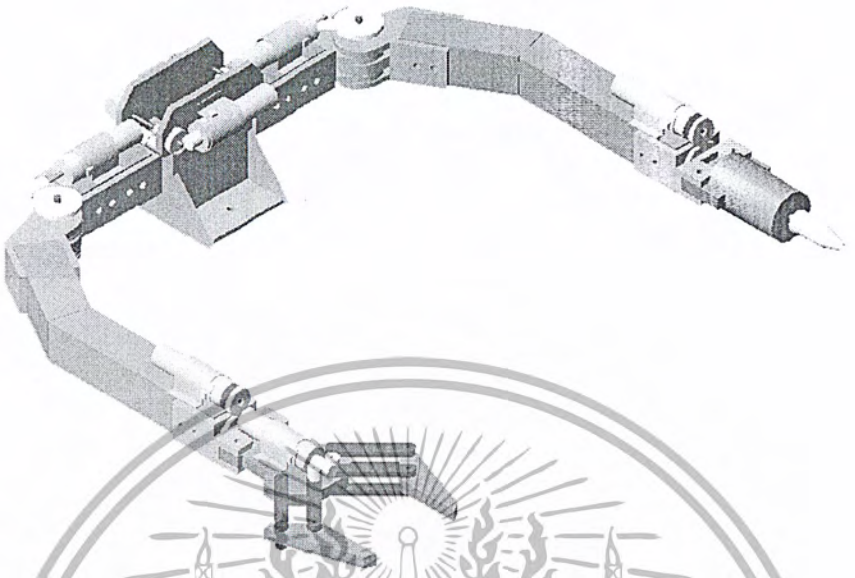
รูปที่ ก-4 แสดงแบบยึดกล้องวงจรปิด 3 มิติ



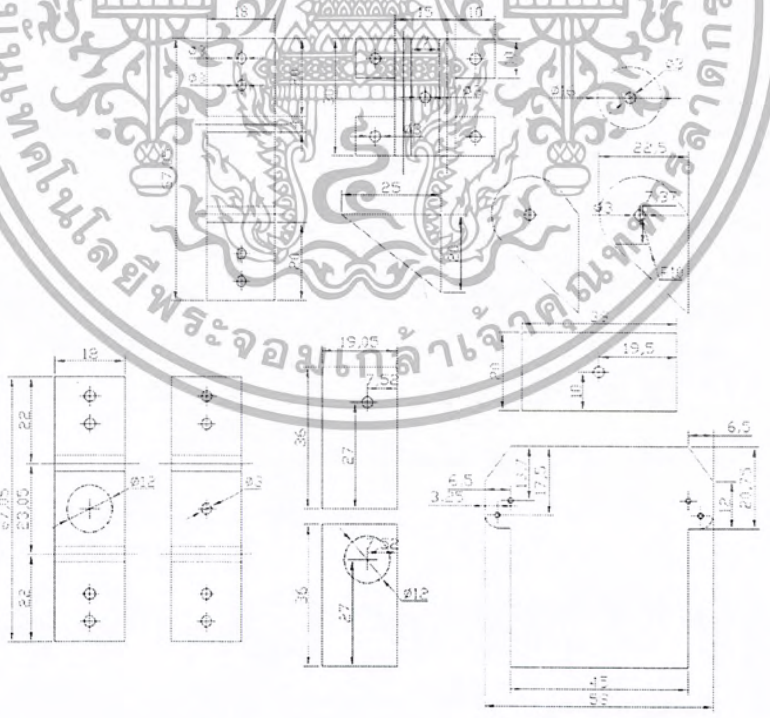
รูปที่ ก-5 แสดงแบบโครงร่างยึดกล้องวงจรปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แบบแขนกล



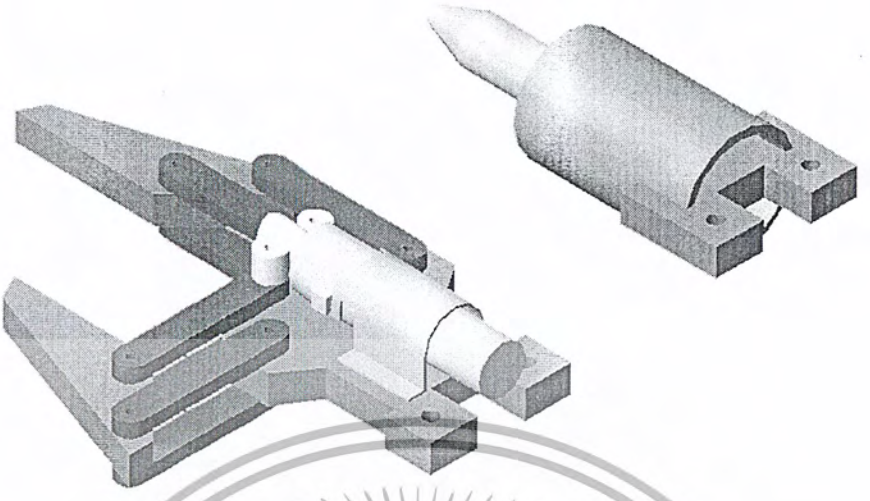
รูปที่ ก-6 แสดงแบบแขนกล 3 มิติ



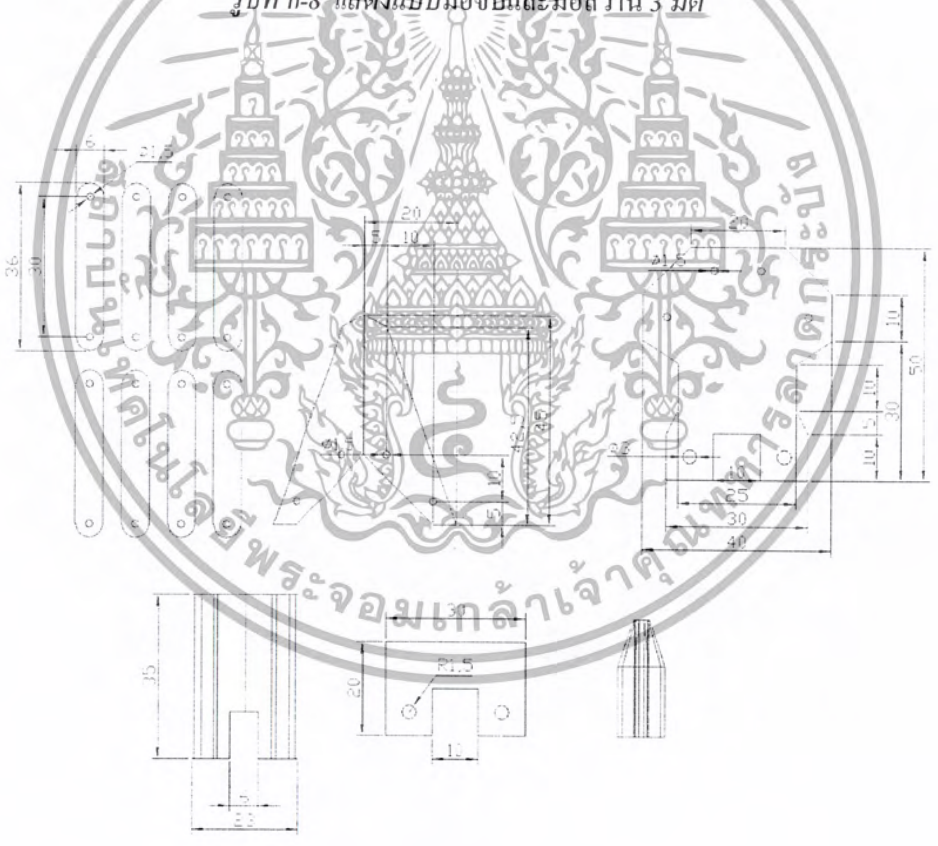
รูปที่ ก-7 แสดงแบบโครงร่างแขนกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. แบบมือจับและมือสว่าน



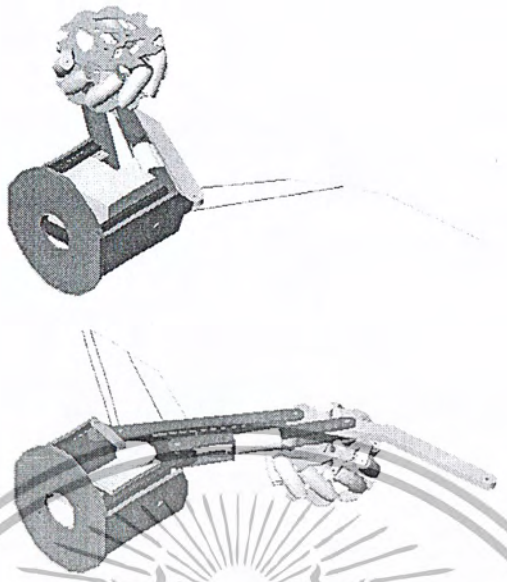
รูปที่ ก-8 แสดงแบบมือจับและมือสว่าน 3 มิติ



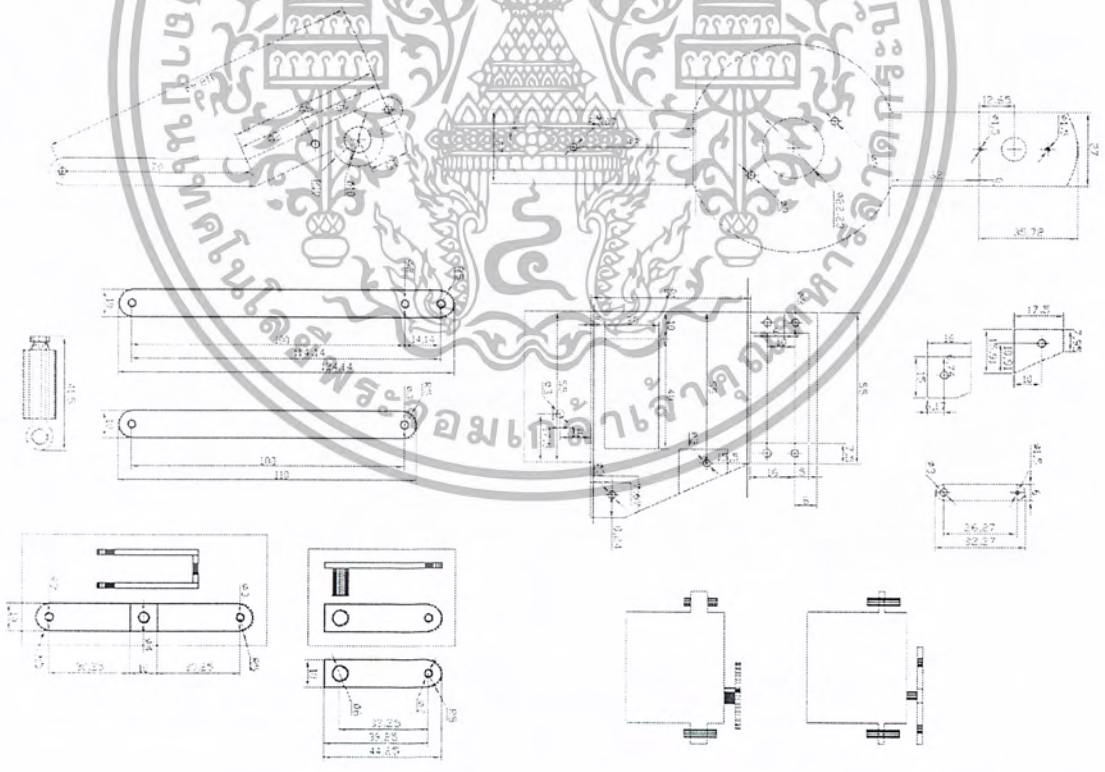
รูปที่ ก-9 แสดงแบบโครงร่างมือจับและมือสว่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. แบบโครงสร้างขา



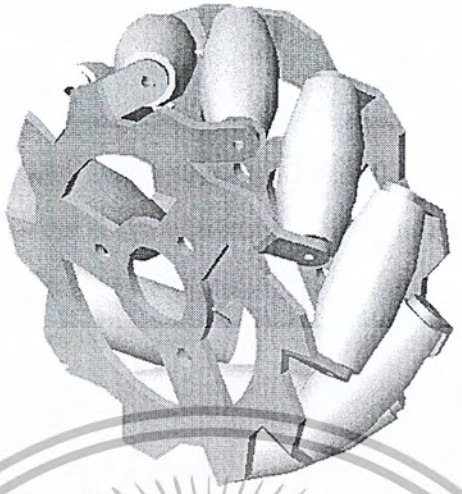
รูปที่ ก-10 แสดงแบบโครงสร้างขา 3 มิติ



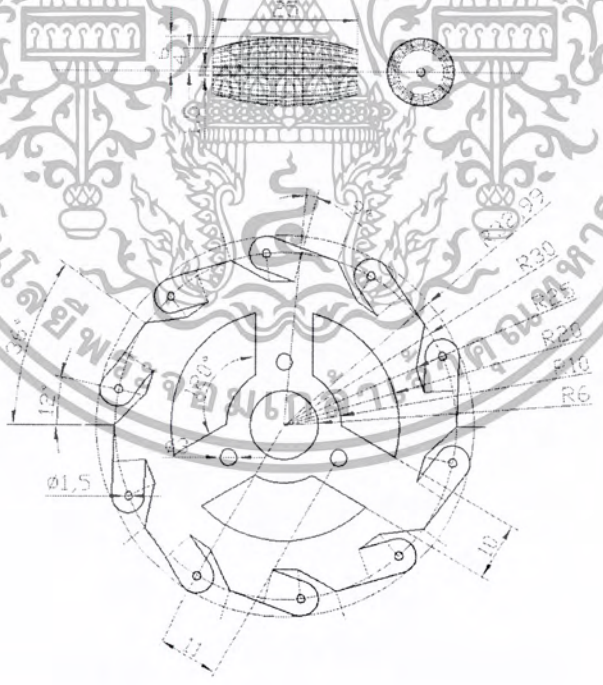
รูปที่ ก-11 แสดงแบบโครงร่างโครงสร้างขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. แบบล้อ Omni



รูปที่ ก-12 แสดงแบบล้อ Omni 3 มิติ



รูปที่ ก-13 แสดงแบบโครงร่างล้อ Omni

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

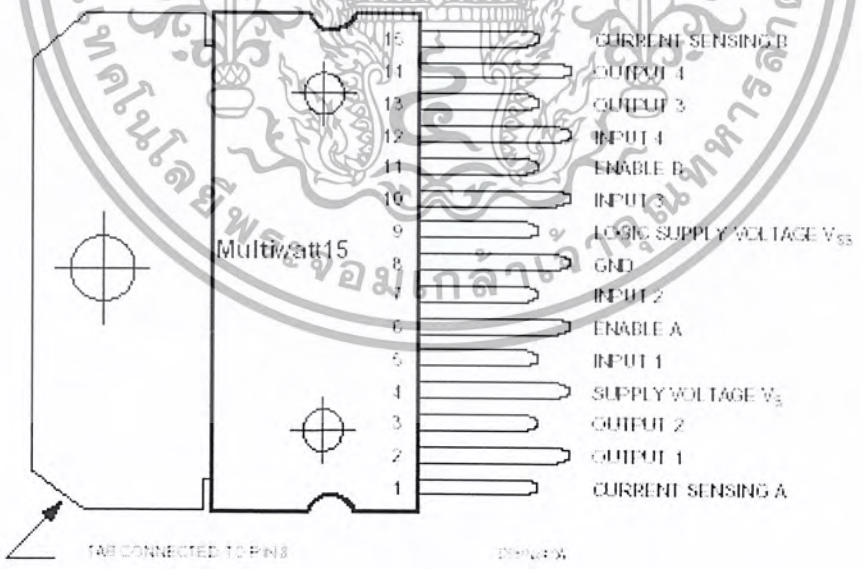
รายละเอียดวงจรการทำงาน

ในบทนี้จะทำการแสดงรายละเอียดวงจรควบคุมการทำงานต่าง ๆ ของหุ่นยนต์ โดยทำการแสดงภาพลายวงจรที่ใช้ในการกัดแผ่นวงจรจริง เพื่อเป็นแนวทางให้กับผู้ที่สนใจได้นำไปประยุกต์ใช้ให้เข้ากับความต้องการได้ โดยในที่นี้ทางผู้พัฒนาได้เลือกใช้โปรแกรม โพรเทล 99 เอสอี (Protel 99SE) เป็นเครื่องมือในการออกแบบและสร้างวงจรการทำงานต่าง ๆ เนื่องจากเป็นโปรแกรมที่ได้รับความนิยมและใช้งานง่าย อีกทั้งร้านที่รับทำแผ่นวงจรพีซีบี (PCB) ก็เลือกใช้โปรแกรมนี้ ทำให้ได้รับความสะดวกในการพัฒนาเป็นอย่างมาก

1. วงจรขับมอเตอร์

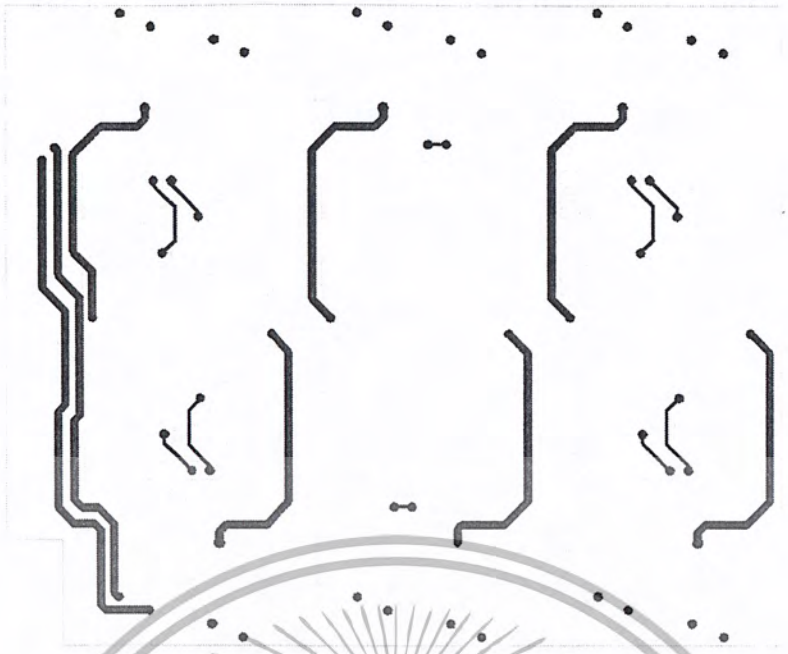
ในการใช้งานมอเตอร์นั้นเราจำเป็นต้องมีการขยายกระแส เนื่องจากหากนำสัญญาณที่ได้จากวงจรมาใช้งานเลยนั้นจะได้กระแสต่ำไม่มีกำลังมากพอในการทำงาน จึงต้องมี ไอซี(IC) พิเศษที่ทำหน้าที่ขยายกระแสให้กับสัญญาณลอจิก ก่อนที่จะนำไปใช้ขับเคลื่อนมอเตอร์จริง

สำหรับวงจรขับมอเตอร์ (Drive Motor) ของ HUOR นั้น เราเลือกใช้ ไอซี L298N ที่สามารถขยายแรงดันได้สูงถึง 46 Volt ขยายกระแสตรง (DC) ได้ถึง 4 แอมแปร์ สามารถป้องกันอันตรายจากคลื่นรบกวนได้สูง และยังมีการป้องกันหากมีอุณหภูมิสูงเกินไปอีกด้วย

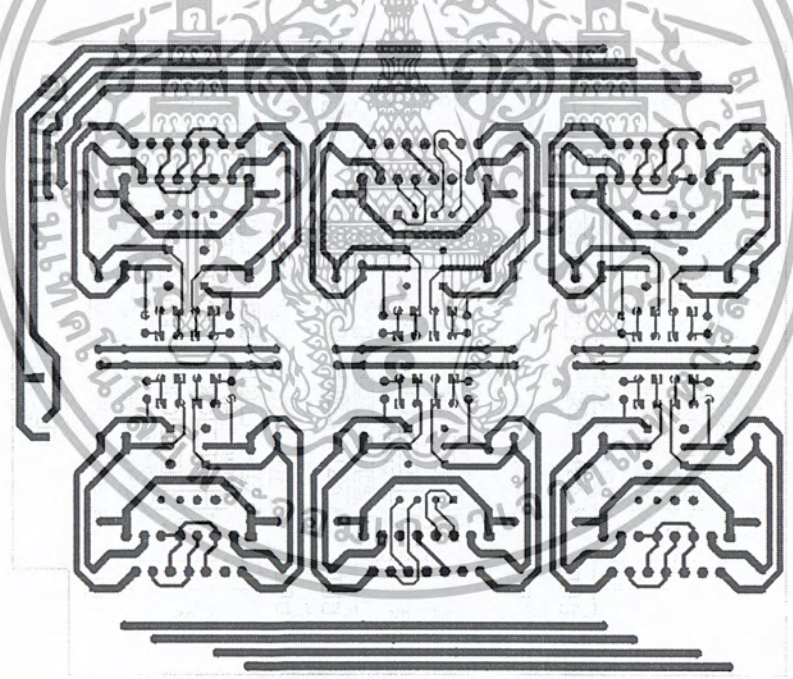


รูปที่ ข-1 แสดงขาไอซี L298N

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข-2 แสดงลายวงจร Top Layer ของวงจรขับมอเตอร์

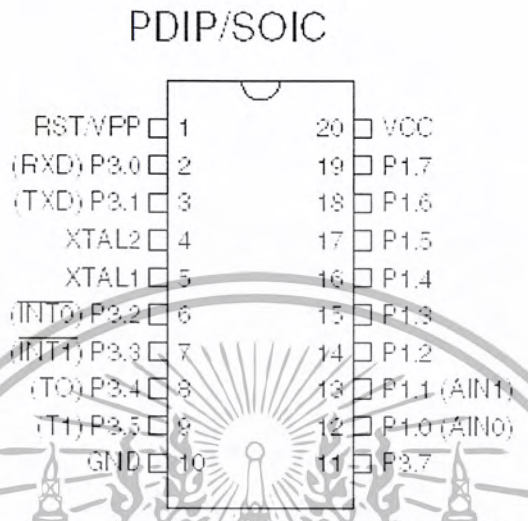


รูปที่ ข-3 แสดงลายวงจร Bottom Layer ของวงจรขับมอเตอร์

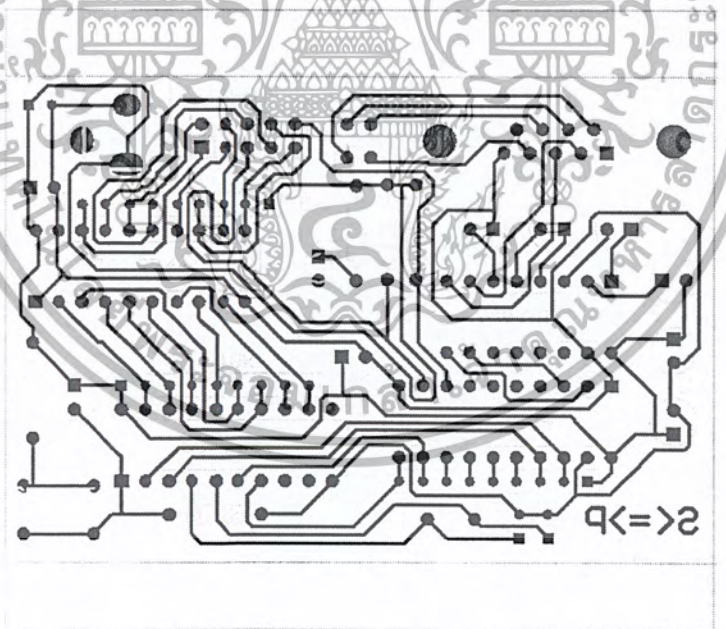
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. วงจรแปลงข้อมูลระหว่างแบบอนุกรมและแบบขนาน

ได้ทำการเลือกใช้ไมโครคอนโทรลเลอร์ MCS-51 ผู้พัฒนาได้ทำการเลือกใช้เบอร์ AT89C2051 ของ Atmel ซึ่งมี 20 ขา มาใช้งาน เนื่องจากมีขนาดเล็ก ช่วยให้ประหยัดเนื้อที่ในการออกแบบวงจร และมีความสามารถเพียงพอต่อการใช้งาน ดังรูป



รูปที่ ข-4 แสดงขาการใช้งาน AT89C2051



รูปที่ ข-5 แสดงลายวงจรแปลงข้อมูลระหว่างแบบอนุกรมและแบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. วงจรอินฟราเรดเซนเซอร์

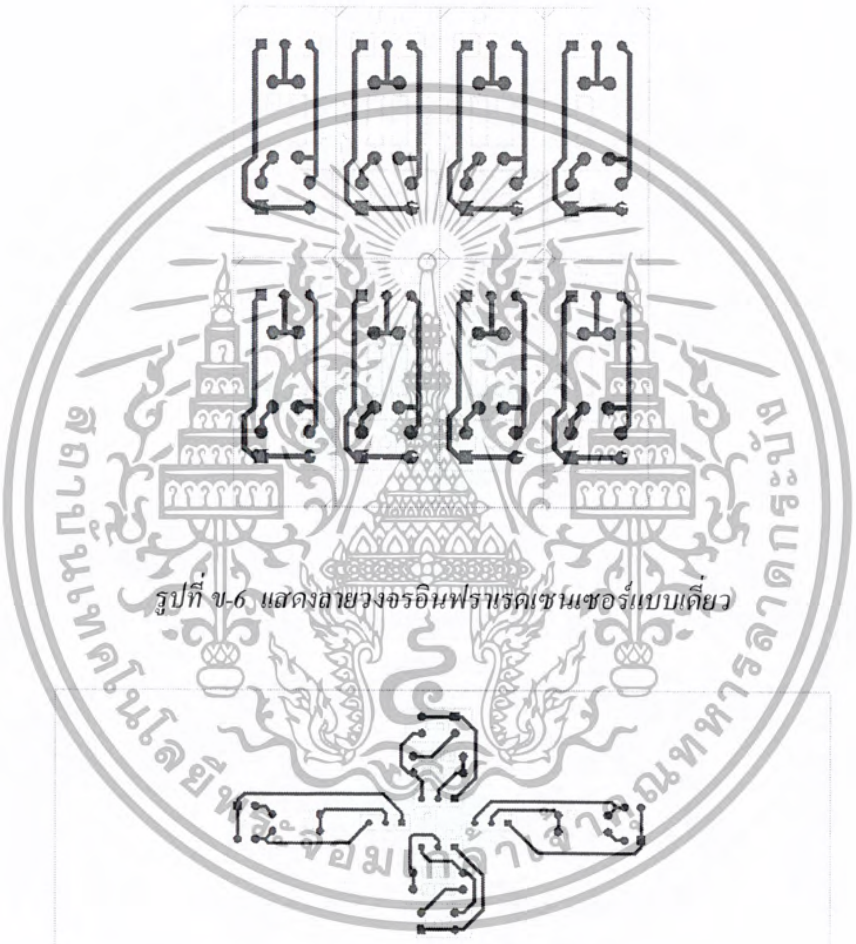
HUOR นั้นใช้อินฟราเรดเซนเซอร์ใน 4 ลักษณะด้วยกัน คือ

1) เพื่อตรวจสอบสิ่งกีดขวางหรือวัตถุที่เข้าใกล้โดยทำการติดไว้ด้านหน้าจำนวน 2 ตัว

2) เพื่อเป็นการตรวจสอบพื้นผิวด้านใต้ว่ามีเส้นทางการเดินอย่างไร ในกรณีสั่งการแบบ Line Detection Operation Mode จำนวน 4 ตัว

3) เพื่อเป็นการตรวจสอบการสัมผัสวัตถุของมือ จำนวน 2 ตัว

4) เพื่อเป็นการตรวจสอบพื้นผิวทางเดินของขาแต่ละข้าง จำนวน 4 ตัว



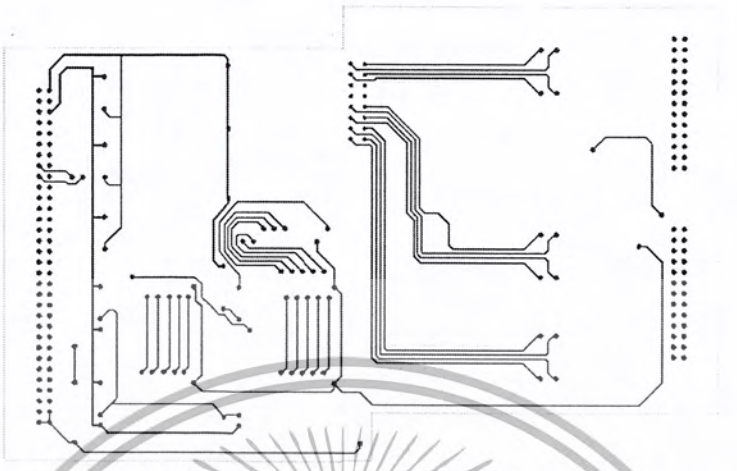
รูปที่ ข-6 แสดงลายวงจรอินฟราเรดเซนเซอร์แบบเดี่ยว

รูปที่ ข-7 แสดงลายวงจรอินฟราเรดเซนเซอร์ตรวจสอบเส้นทางการเดิน

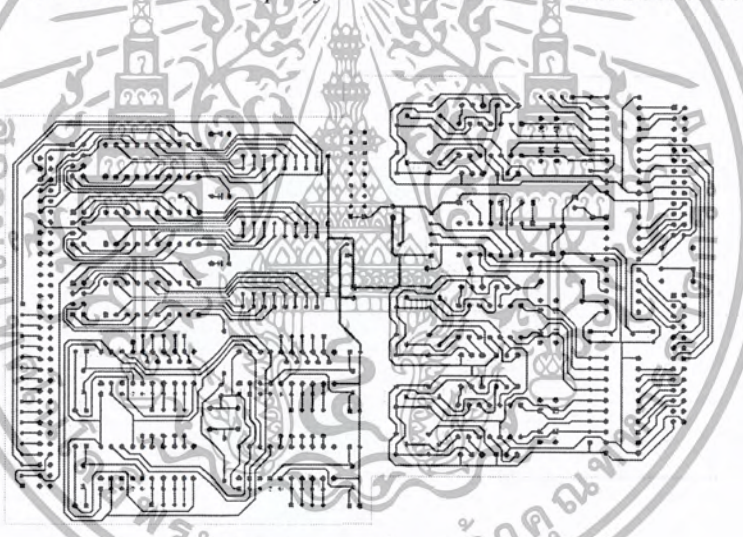
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. วงจรบอร์ดไอซาและบอร์ดพีไอโอ

ทำการแสดงลายวงจรของบอร์ดไอซาและบอร์ดพีไอโอ ดังนี้



รูปที่ ข-8 แสดงลายวงจร Top Layer ของวงจรบอร์ดไอซาและบอร์ดพีไอโอ



รูปที่ ข-9 แสดงลายวงจร Bottom Layer ของวงจรบอร์ดไอซาและบอร์ดพีไอโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

รายละเอียดการทำงานของโปรแกรม

1. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการทำงานหลักบนชุดพัฒนาคอม 86

รูปที่ ค-1 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการทำงานหลัก

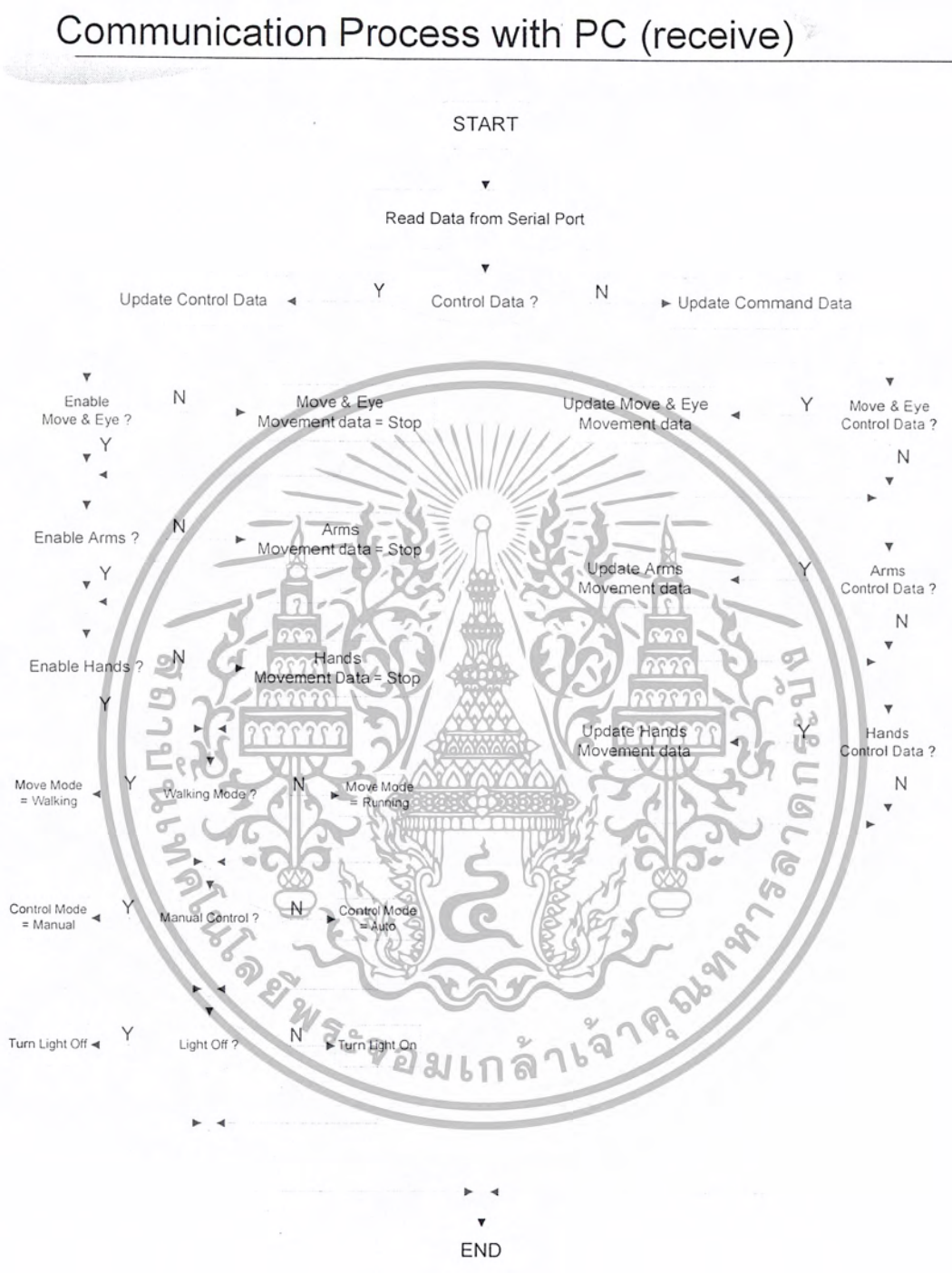
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการส่งข้อมูล

รูปที่ ค-2 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

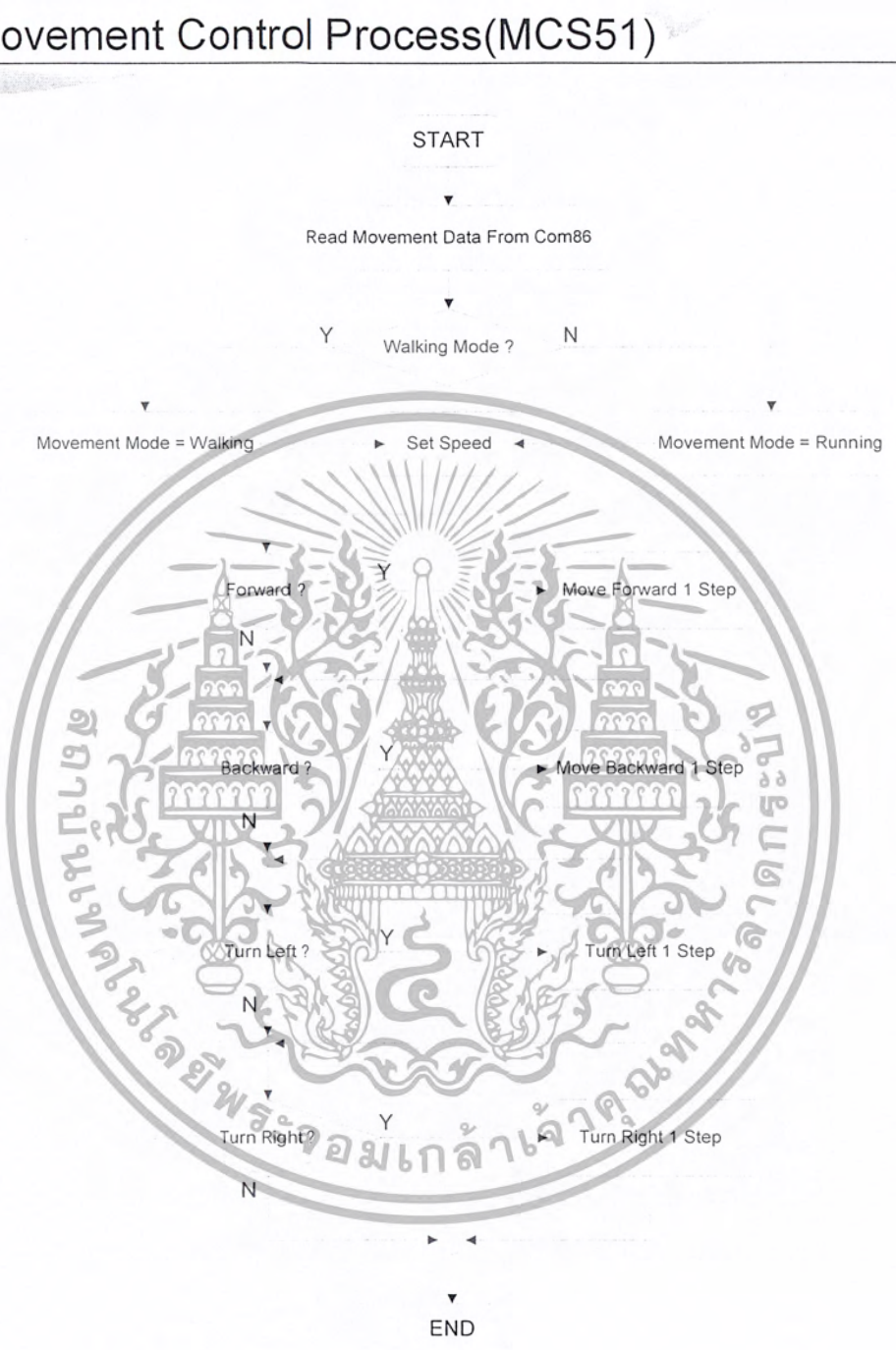
3. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการรับข้อมูล



รูปที่ ค-3 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการการเคลื่อนที่บนไมโครคอนโทรลเลอร์ MCS-51



รูปที่ ค-4 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการการเคลื่อนที่

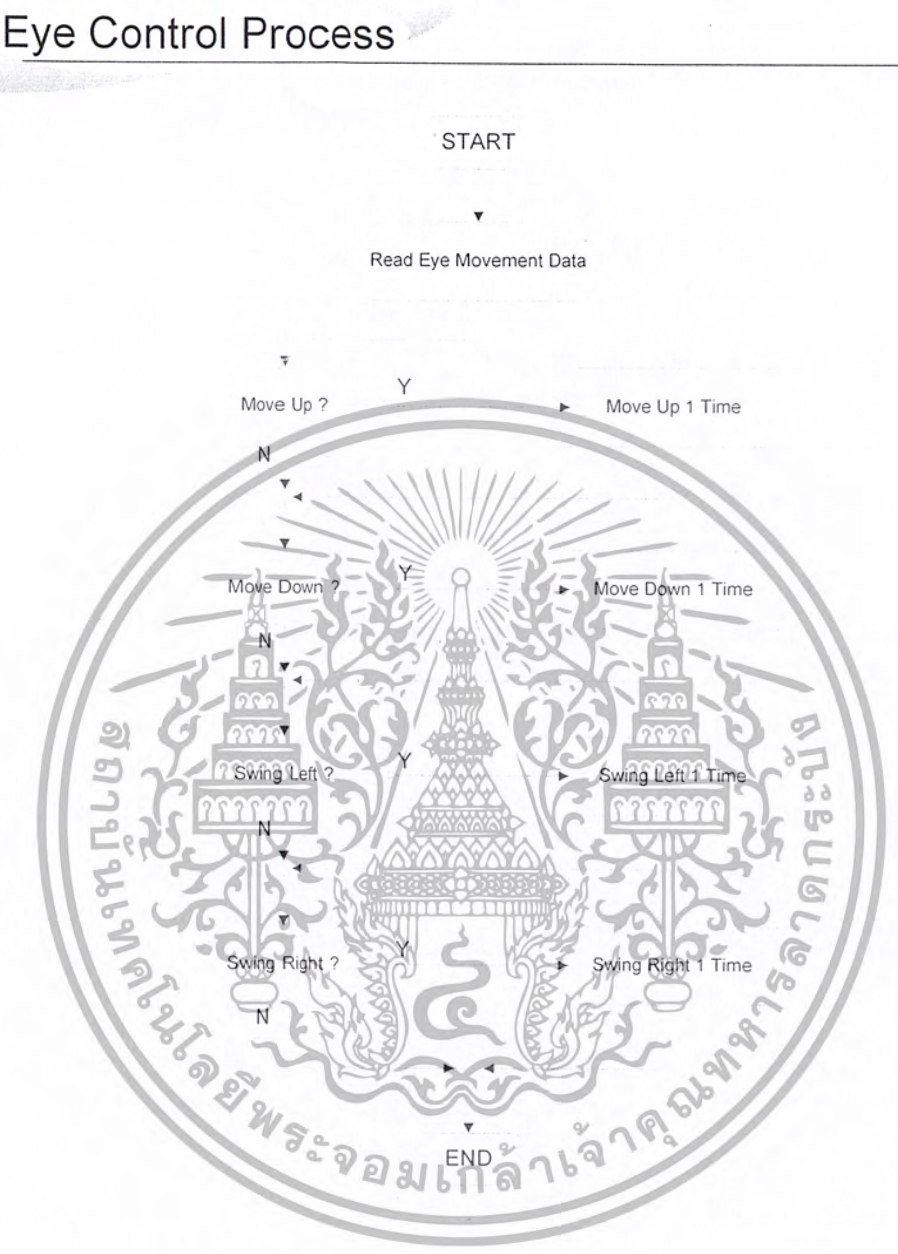
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการเปลี่ยนโหมดการเคลื่อนที่ บนไมโครคอนโทรลเลอร์ MCS-51

รูปที่ ค-5 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการเปลี่ยนโหมดการเคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

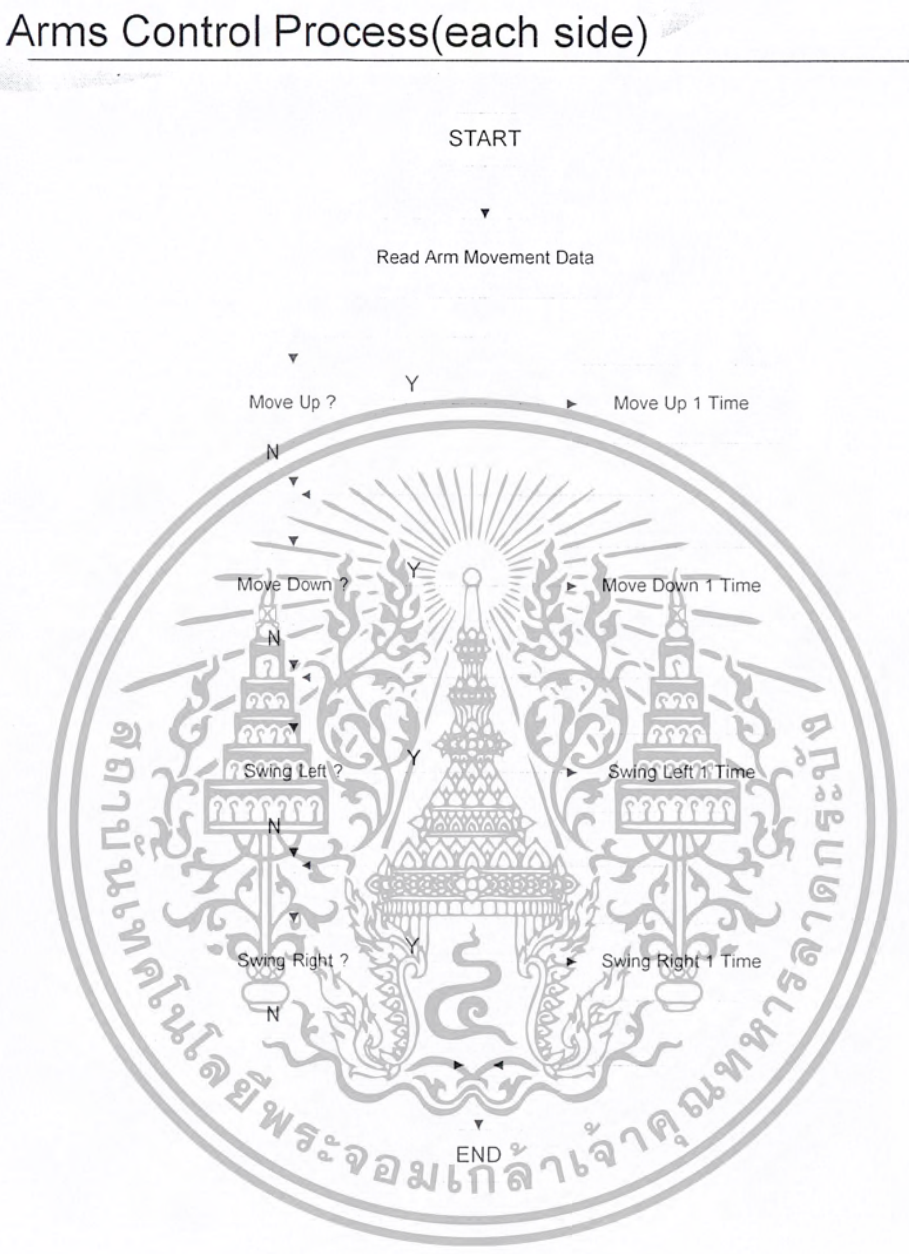
6. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการควบคุมตา



รูปที่ ค-6 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการควบคุมตา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

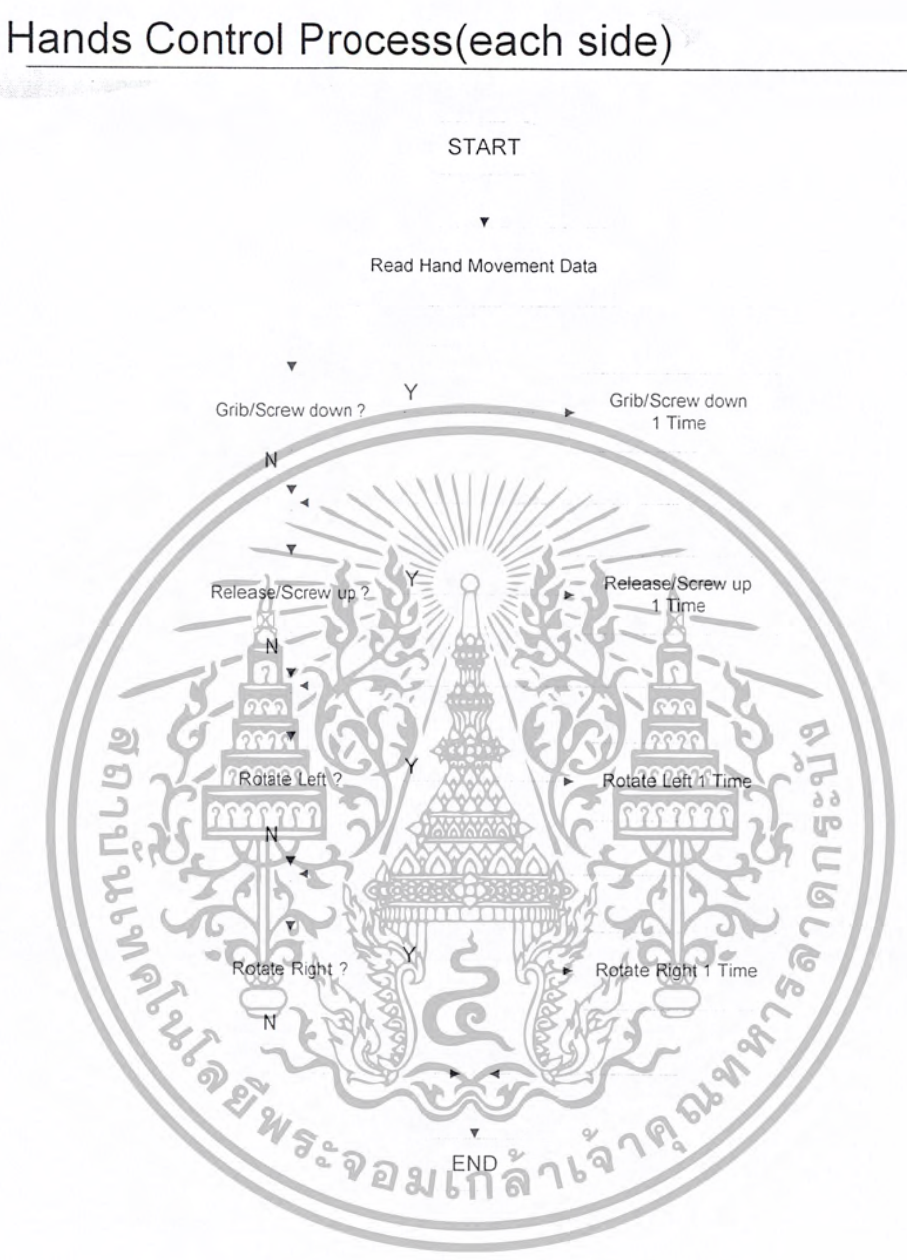
7. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการควบคุมแขน



รูปที่ ค-7 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการควบคุมแขน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

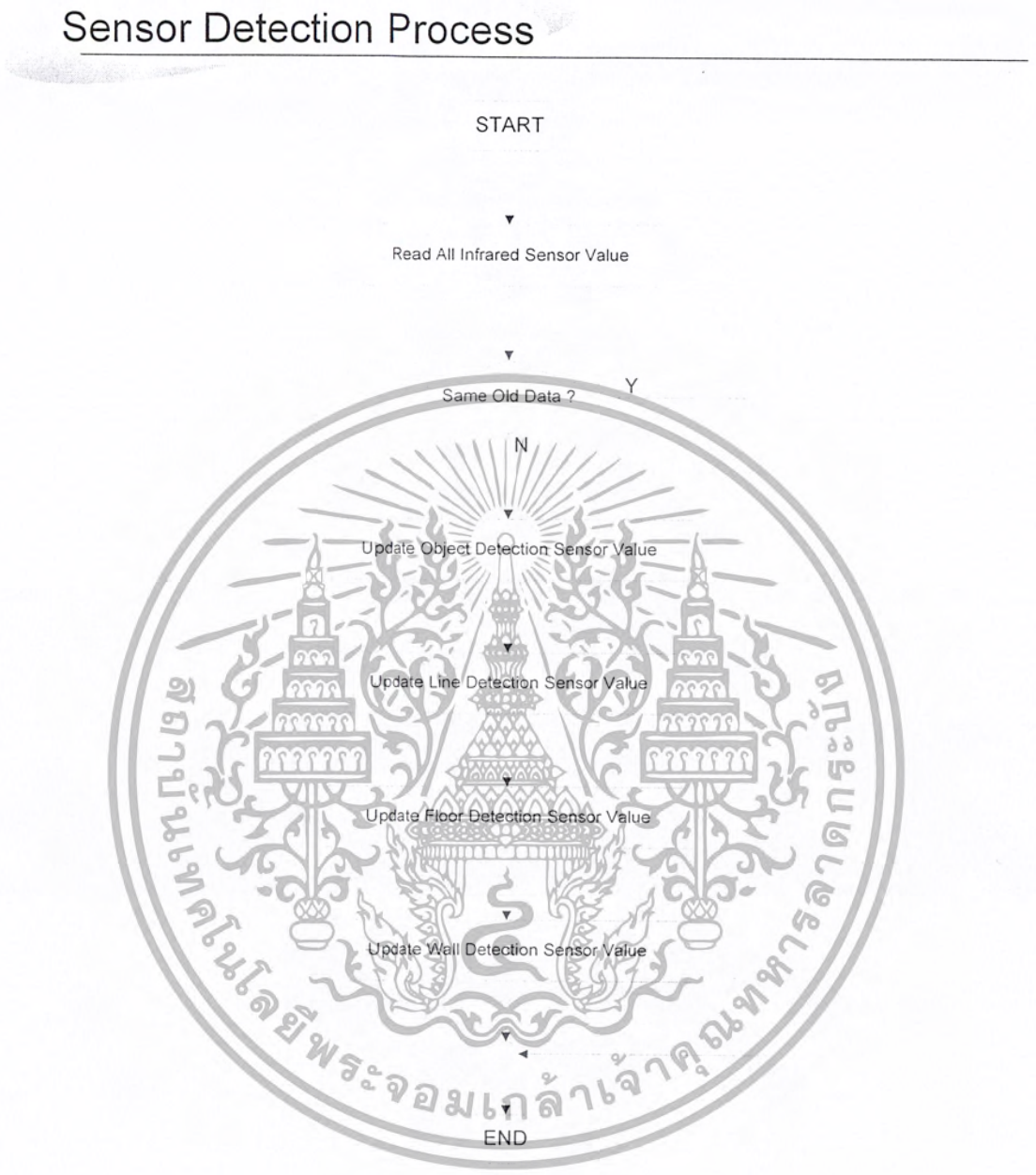
8. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการควบคุมมือ



รูปที่ ค-8 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการควบคุมมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการตรวจสอบการทำงานของเซนเซอร์



รูปที่ ค-9 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการตรวจสอบการทำงานของเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. โฟลว์ชาร์ตแสดงการทำงานของกระบวนการตรวจหาระยะทาง

รูปที่ ค-10 แสดงโฟลว์ชาร์ตการทำงานของกระบวนการตรวจหาระยะทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.
วิธีการติดตั้งและการใช้งาน

1. ชุดพัฒนาคอม 86

ความต้องการของระบบเบื้องต้น

ในการใช้งานชุดพัฒนาคอม 86 นั้นจะมีความต้องการเบื้องต้นทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์ ดังนี้

- เครื่องคอมพิวเตอร์สำหรับใช้คอมไพล์และตรวจสอบการทำงานของโปรแกรม 1 เครื่อง โดย จะต้องมีพอร์ตอนุกรมสำหรับใช้งานได้อย่างน้อย 1 พอร์ต และมีการ์ดอีเธอร์เน็ตอย่างน้อย 1 การ์ด (หากต้องการพัฒนาโปแกรมที่ใช้คุณสมบัติทางด้าน Network)
- โปรแกรมเทอร์มินัล เช่น Hyper Terminal เป็นต้น
- สายสื่อสารอนุกรม serial
- คอมไพล์เลอร์ที่สามารถคอมไพล์โปรแกรมสำหรับไมโครโพรเซสเซอร์ตระกูล 80186 แบบ Real mode เช่น turbo C, Turbo assembly หรืออื่นๆ ตามที่ผู้พัฒนาต้องการใช้งาน

1.1 การติดตั้งใช้งาน

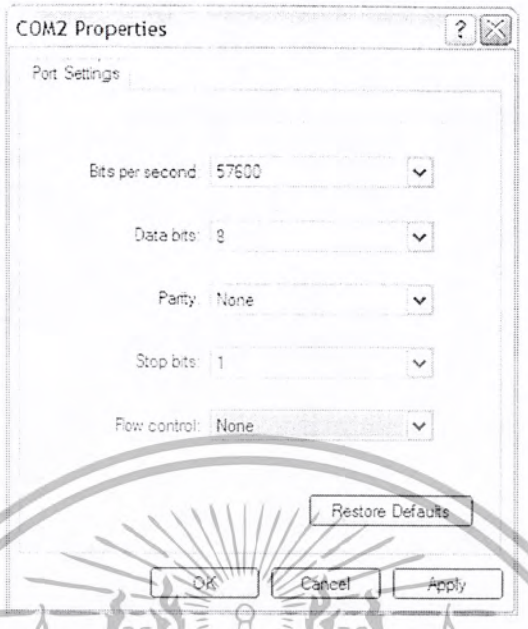
ในการติดตั้งใช้ชุดพัฒนาคอม86 ให้ปฏิบัติตามขั้นตอนต่างๆดังนี้

1. เชื่อมต่อสายของ Serial cable ที่ให้มากับพอร์ตอนุกรมของคอมพิวเตอร์ที่ต้องการใช้งาน และเชื่อมต่อสายอีกด้านเข้ากับบอร์ดคอม86

รูปที่ ง-1 แสดงการเชื่อมต่อสาย serial console เข้ากับบอร์ดคอม86

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เปิดโปรแกรม Hyper Terminal ที่ใช้งานขึ้นมาและทำการกำหนดค่าต่างๆ ดังนี้

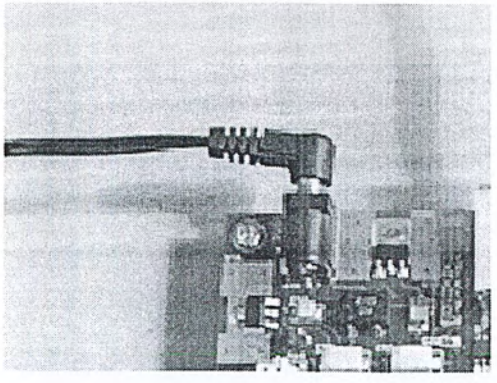


รูปที่ ง-2 แสดงการตั้งค่าต่างๆ ของโปรแกรมเทอร์มินัล

- เลือกพอร์ตการเชื่อมต่อของคอมพิวเตอร์ ตามที่ได้ทำการเชื่อมต่อในขั้นตอนที่ 2
- กำหนดอัตราการรับส่งข้อมูลอยู่ที่ 57600 บิตต่อวินาที
- ข้อมูลเป็นแบบ 8 บิต
- ไม่มีพาริตี้
- Stop บิตเท่ากับ1
- ยกเลิกการทำโฟลว์คอนโทรลทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์

และสั่งเริ่มการเชื่อมต่อ

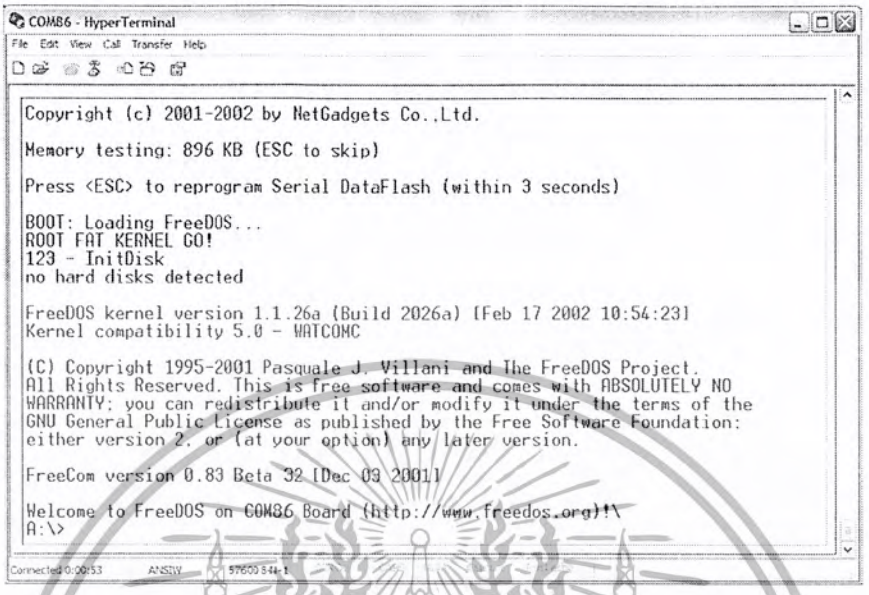
3. ต่อปลั๊กของอแดปเตอร์เข้ากับบอร์ดคอม86 และต่ออแดปเตอร์เข้ากับแหล่งจ่ายไฟของอาคาร



รูปที่ ง-3 แสดงการเชื่อมต่ออแดปเตอร์กับบอร์ดคอม 86

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. กดรีเซตสวิตช์จะเห็นโปรแกรมบนบอร์ดเริ่มทำงานโดยการแสดงผลผ่านโปรแกรมเทอร์มินัลที่ใช้งาน



รูปที่ ง-4 แสดงตัวอย่างการแสดงผลของบอร์ดคอม 86 เมื่อบอร์ดเริ่มทำงาน

1.2 การ download image file

ในการใช้งานชุดพัฒนาคอม86นั้น นักพัฒนาสามารถใช้งานระบบปฏิบัติการอื่นๆ นอกจาก FreeDOS ได้เช่นกันโดยใน BIOS ของบอร์ดคอม86นั้นได้บริการให้นักพัฒนาสามารถดาวน์โหลดโปรแกรมระบบปฏิบัติการที่ตนต้องการใช้งานได้เองโดยผ่านฟังก์ชันการทำงานของ BIOS ซึ่งสามารถทำได้โดยเมื่อปรากฏข้อความ

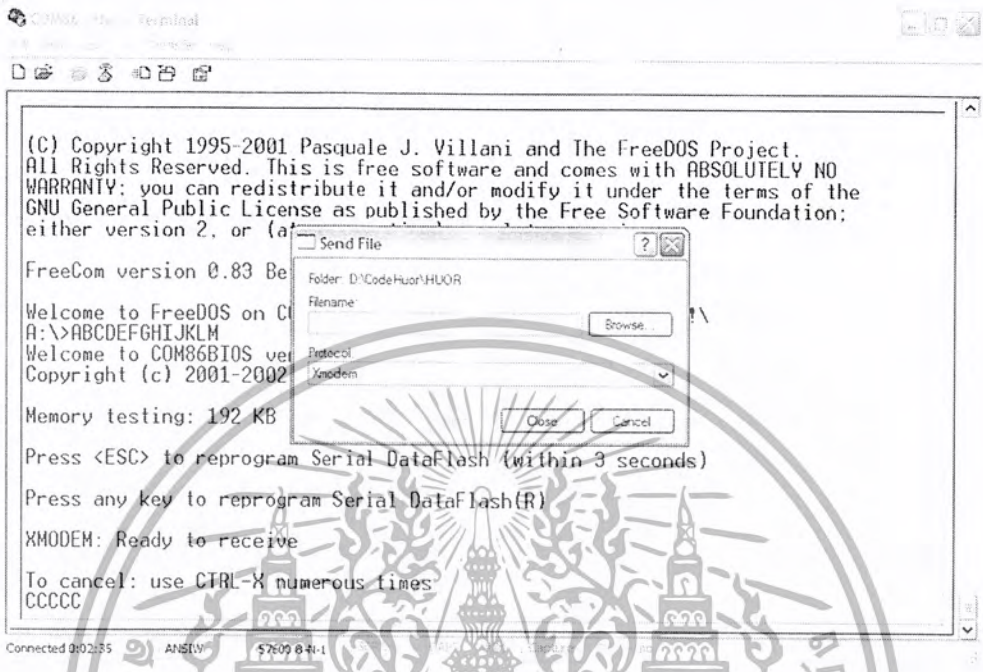
```
Press <ESC> to reprogram Serial DataFlash (within 3 seconds)
```

แสดงขึ้นมาให้กดปุ่ม ESC เพื่อเริ่มการโปรแกรม Serial DataFlash โดยโปรแกรมที่จะโปรแกรมลงไปนั้นจะอยู่ในรูปของ Image file โดยซึ่งภายในชุดพัฒนาจะมี โปรแกรมสำหรับสร้าง image file นี้ รวมทั้งตัว image file ของ FreeDOS ให้ไปด้วย ซึ่งเมื่อกดปุ่ม ESC จะขึ้นข้อความดังนี้

```
Press any key to reprogram Serial DataFlash
XMODEM: Ready to receive
To cancel: use CTRL-X numerous times
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัว C จะขึ้นมาแสดงถึงการพร้อมรับการ download Image file ของบอร์ดคอม86 ซึ่งหากต้องการยกเลิกสามารถทำได้โดยการกดปุ่ม Ctrl –X หลายๆครั้งติดต่อกัน โดยในการส่ง Image file นี้จะใช้ protocol X-Modem ในการส่ง image file ไปยังบอร์ดคอม86 ซึ่งตัวอย่างการส่ง file เป็นไปดังรูป



รูปที่ ง-5 แสดงตัวอย่างการส่ง file image ไปโปรแกรมลงบนบอร์ดคอม 86

เมื่อ Image file ถูก download และติดตั้งเสร็จจะขึ้นข้อความ

```
Flash image programmed complete
```

จากนั้นระบบจะเริ่มทำการ load ระบบปฏิบัติการที่โปรแกรมขึ้นมาและเริ่มการทำงานในส่วนต่อไป ตาม Boot sequence ในการพัฒนาโปรแกรมนั้นนักพัฒนาสามารถ Download Image ได้ตลอดเวลาตามต้องการ เช่น เมื่อเกิดการทำงานผิดพลาดหรือต้องการติดตั้งค่า configuration ในเบื้องต้นของระบบภายในชุดพัฒนานี้จะมีตัว image file ที่มีโปรแกรมสำหรับใช้งานต่าง ๆ ในเบื้องต้นอยู่ใน CD ROM ของชุดพัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 การ Transfer file ระหว่าง PC/COM86

ในการพัฒนาแอพลิเคชั่นต่างๆบนบอร์ดคอม86 นั้น นักพัฒนาย่อมเกิดความต้องการในการรับส่ง file ระหว่างบอร์ดคอม86 กับภายนอกขึ้นมาอย่างแน่นอน ซึ่งในชุดพัฒนาคอม86 ได้มีโปรแกรมยูทิลิตี้ตัวหนึ่งในการรองรับการทำงานในส่วนนี้เป็นโปรแกรมชื่อว่า transfer.exe เป็นโปรแกรมที่ช่วยในการรับ-ส่ง file ต่างๆ ผ่านทางพอร์ตอนุกรมคล้ายกับการโปรแกรม Image file ของบอร์ดคอม86 ซึ่งจะใช้โปรโตคอล X-modem ในการรับส่งไฟล์เช่นเดียวกันที่ความเร็ว 57600 bps ซึ่งรูปแบบการใช้งานของโปรแกรมมีดังนี้

Transfer [/S] [/R] [/H] [/?] filename

/S คือเลือกใช้การส่ง file ไปยังPC

/R คือเลือกใช้การรับ file จาก PC (ค่านี้เป็นค่า default ของโปรแกรม)

/? หรือ /H เป็นการเรียกดูรูปแบบการใช้งานของโปรแกรม

filename คือชื่อของไฟล์ที่ต้องการจะรับหรือส่ง

ตัวอย่างการรับ file จาก PC

การรับ file จาก PC สามารถทำได้บน DOS prompt ของบอร์ดคอม86 เรียกใช้โปรแกรมดังนี้

```
A:\>transfer /R  test.txt
```

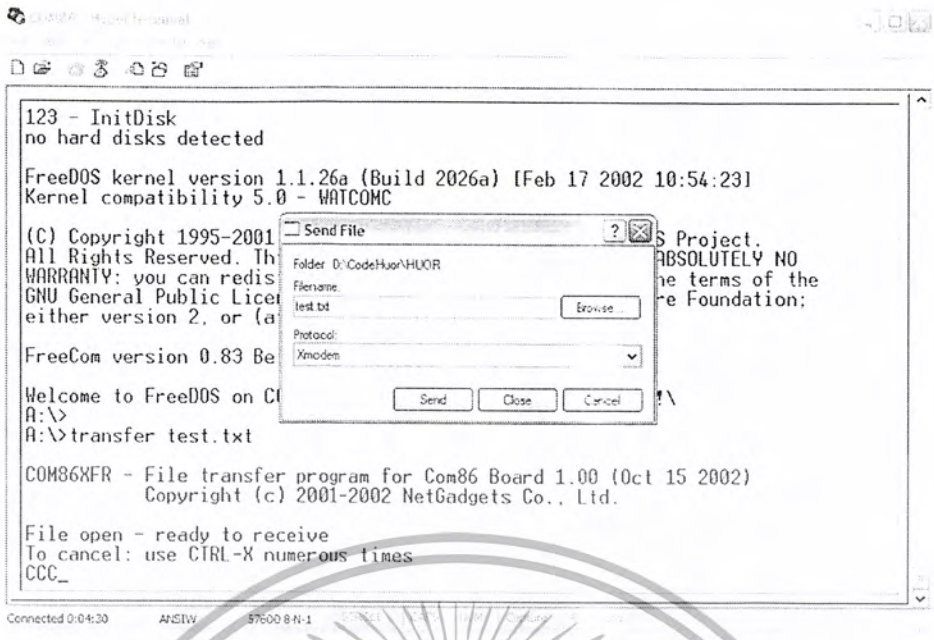
เมื่อกด ปุ่ม Enter โปรแกรมแสดงข้อความต่างๆ ดังนี้

```
TRANSFER - File transfer program for Com86 Board 1.00 (Dec 30 2001)
Copyright (c) 2001 NetGadgets Co., Ltd.

File open - ready to receive
To cancel: use CTRL-X numerous times
```

หลังจากนั้นทำการส่ง file โดยการใช้โปรแกรม terminal ของทางฝั่ง PC โดยเลือกโพรโตคอลในการส่ง file เป็น X-modem ดังตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-6 แสดงตัวอย่างการส่ง file จาก PC

เมื่อทางฝั่งบอร์ดคอม86 รับ file เสร็จสิ้นก็จะกลับไปยัง DOS prompt เพื่อรอรับคำสั่งในการทำงานต่อไป

หมายเหตุ

เนื่องจากข้อจำกัดของโพรโตคอล X-modem ที่ใช้จะทำให้ขนาดของไฟล์ต่างๆ ที่รับ-ส่งนั้นถูกปัดให้มีขนาดลงตัวที่ 128 ไบต์

ตัวอย่างการส่ง fileไปยังเครื่อง PC

การส่งไฟล์ไปจากบอร์ดคอม86 ยังเครื่อง PC จะกระทำเช่นเดียวกับการรับไฟล์จากเครื่อง PC โดยการเรียกใช้งานโปรแกรม Transfer บน DOS Prompt ดังตัวอย่าง

```
A:\>transfer /s   test.txt
```

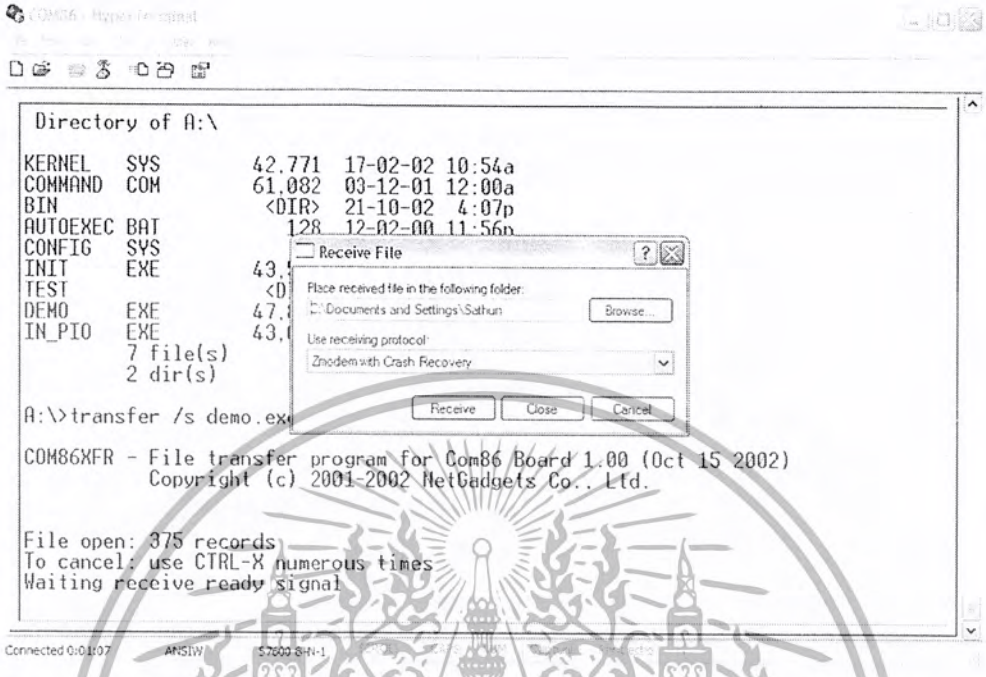
เมื่อกดคีย์ enter โปรแกรมจะแสดงข้อความดังนี้

```
A:\>transfer /s test.txt

TRANSFER - File transfer program for Com86 Board 1.00 (Dec 30 2001)
           Copyright (c) 2001 NetGadgets Co., Ltd.
File open: 2 records
To cancel: use CTRL-X numerous times
Waiting receive ready signal
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ไปที่โปรแกรมเทอร์มินัลที่ใช้งานอยู่และเลือกทำการรับ file จากเทอร์มินัลที่ใช้งานอยู่โดยในการรับนี้ให้เลือกใช้โพรโตคอล X-modem ในการรับส่งข้อมูลเช่นเดียวกันดังตัวอย่าง



รูปที่ ง-7 แสดงตัวอย่างการรับ file จาก com86

ซึ่งหลังจากส่งไฟล์เสร็จก็จะกลับมายัง DOS prompt เพื่อรอรับคำสั่งในการทำงานต่อไป

1.4 การรันโปรแกรมบนบอร์ดคอม86

ในการรันโปรแกรมบนบอร์ดคอม86 นั้นจะกระทำเหมือนกับการรันโปรแกรมต่าง ๆ บน DOS ของเครื่อง PC ปกติโดยเพียง download file ที่ต้องการจะนำมารันเข้ามายังบอร์ดคอม 86 ก่อน จากนั้นไปยังไดเรคทอรีที่เก็บไฟล์โปรแกรมที่ต้องการเรียกใช้งานเอาไว้ และทำการเรียกโปรแกรมตามปกติเหมือนกันกับบนคอมพิวเตอร์ทั่วไป ในกรณีที่ต้องการให้โปรแกรมเริ่มทำงานทุกครั้งโดยอัตโนมัติเมื่อเริ่มระบบสามารถทำได้โดยการสั่งเรียกใช้งานโปรแกรมนั้นๆ ภายใน file Autoexec.bat ซึ่งเมื่อระบบเริ่มทำงานโปรแกรมนั้นๆ จะถูกเรียกใช้งานได้โดยอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Borland C++ version 5.0

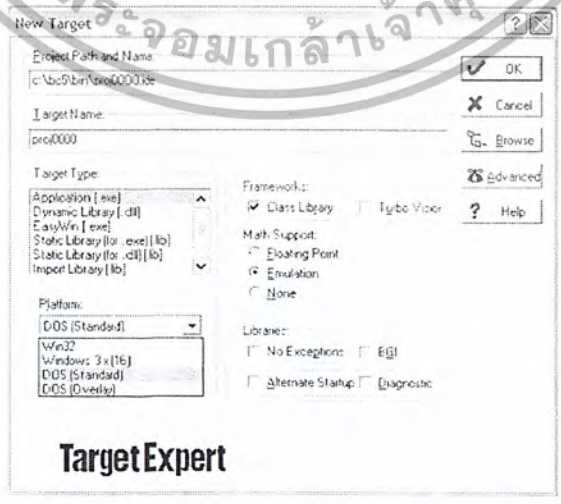
เนื่องจากชุดพัฒนาคอม 86 นั้นใช้ระบบปฏิบัติการดอสแบบ 16 บิต นั่นหมายความว่าโปรแกรมที่สามารถนำมาใช้งานได้นั้นจะต้องเป็นโปรแกรมแบบ 16 บิตเช่นกัน ดังนั้นเราจึงต้องหาคอมไพเลอร์ที่สามารถทำการคอมไพล์โปรแกรมที่เราพัฒนาขึ้นในแบบ 16 บิตได้ ซึ่ง Borland C++ version 5.0 นี้สามารถทำงานได้ตามที่เราต้องการ โดยมีขั้นตอนการใช้งาน ดังนี้

2.1 ขั้นตอนการสร้างโปรเจคแบบดอส 16 บิต

ขั้นตอนที่ 1 ทำการเปิดโปรแกรม Borland C++ ขึ้นมา สร้างโปรเจคใหม่ ไปที่ File -> New -> Project ดังรูป

รูปที่ ง-8 แสดงการสร้างโปรเจคใหม่ใน Borland C++

ขั้นตอนที่ 2 ทำการตั้งชื่อ Project name, Target name, เลือก Target Type ซึ่งค่าเริ่มต้นคือ Application [.exe] และทำการเลือก Platform เป็น DOS (Standard) แล้วกดปุ่มตกลง (OK) ดังรูป

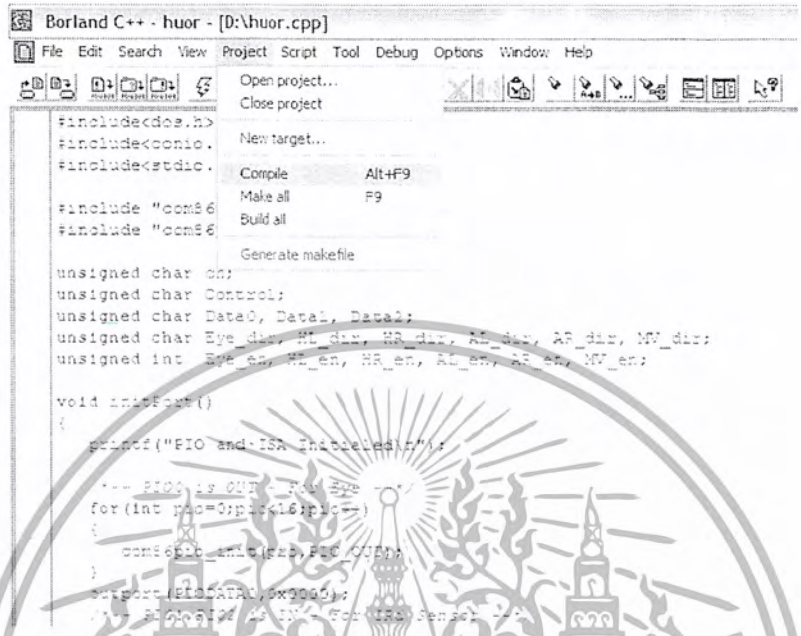


รูปที่ ง-9 แสดงการตั้งค่าให้โปรเจคใน Borland C++

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

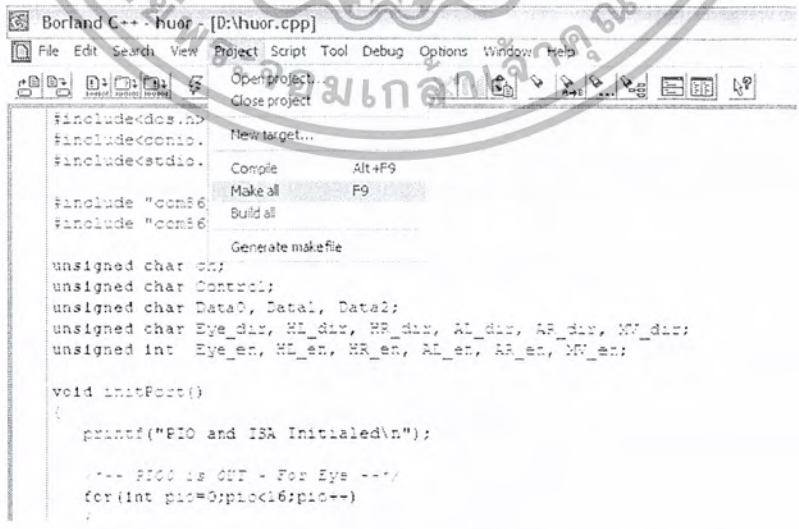
2.2 ขั้นตอนการคอมไพล์โปรเจคแบบดอส 16 บิต

ขั้นตอนที่ 1 หลังจากทำการเขียนโค้ดโปรแกรมแล้ว ให้ไปที่ Project -> Compile หรือกด Alt+F9 ที่คีย์บอร์ดดังรูป



รูปที่ ง-10 แสดงวิธีการคอมไพล์ใน Borland C++

ขั้นตอนที่ 2 หากโปรแกรมที่เราเขียนมีข้อผิดพลาด จะมีหน้าต่างแสดงข้อความผิดพลาดนั้น ๆ ให้ทำการแก้ไขจนกว่าจะถูกต้องสมบูรณ์ จากนั้นทำการสร้างไฟล์ execute (.exe) ของโปรแกรม โดยให้ไปที่ Project -> Make all หรือกด F9 ที่คีย์บอร์ด ดังรูป



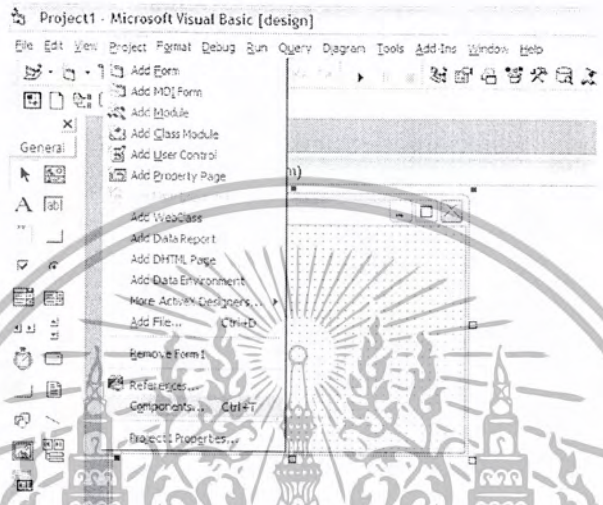
รูปที่ ง-11 แสดงวิธีการสร้างไฟล์ Execute ใน Borland C++

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การใช้งาน VideoOCX ร่วมกับ Microsoft Visual Basic 6.0

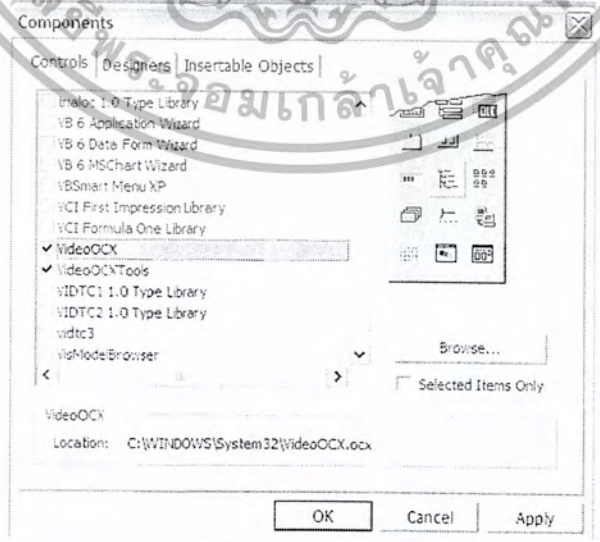
VideoOCX เป็น component ที่ใช้ในการติดต่อกับ Video capture card ซึ่งต้องทำการติดตั้งก่อนจึงจะสามารถใช้งานได้ ซึ่งสามารถทำการติดตั้งได้เหมือนกับโปรแกรมทั่วไป เมื่อทำการติดตั้งเรียบร้อยแล้วให้ทำการเรียกโปรแกรม Microsoft Visual Basic 6.0 ขึ้นมาแล้วทำการเพิ่ม component ตามขั้นตอนดังนี้

ขั้นตอนที่ 1 ไปที่ Project -> Component หรือกด Ctr+T ที่คีย์บอร์ด ดังรูป



รูปที่ ง-12 แสดงวิธีการเพิ่ม Component ใน Visual Basic 6.0

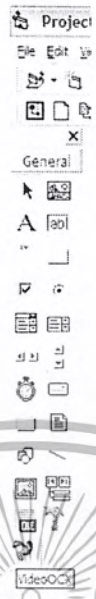
ขั้นตอนที่ 2 ให้ทำเครื่องหมายหน้า component ชื่อ "VideoOCX" และ "VideoOCX Tools" ดังรูป



รูปที่ ง-13 แสดงการเลือกใช้งาน Component VideoOCX ใน Visual Basic 6.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

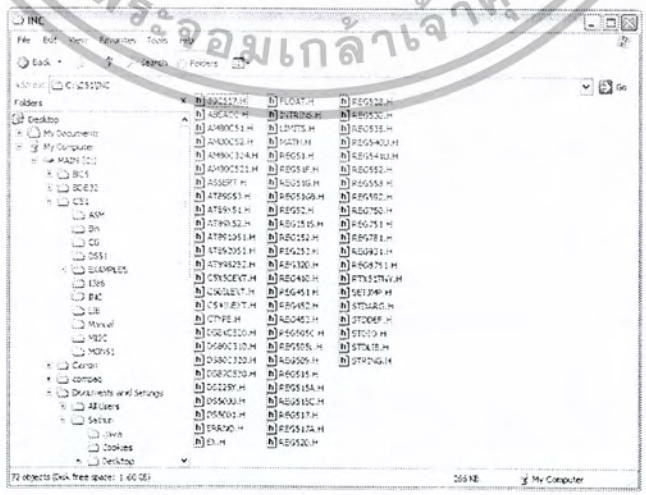
ที่แถบเครื่องมือจะปรากฏไอคอนเพื่อเรียกใช้งาน VideoOCX ขึ้นดังรูป



รูปที่ ง-14 แสดงเครื่องมือ VideoOCX ใน Visual Basic 6.0

4. Keil 8051 C Compiler version 5.50

ทำการติดตั้งและใช้งานเหมือนกับโปรแกรมทั่วไป ซึ่งโปรแกรม Keil หรือ C51 นี้เป็นเครื่องมือที่ช่วยให้เราสามารถเขียนโค้ดโปรแกรมบนไมโครคอนโทรลเลอร์ MCS-51 ได้สะดวกขึ้นด้วยภาษา C โดยทำการ include เฮดเดอร์ไฟล์ (.h) ที่ตรงกับยี่ห้อและเบอร์ของคอนโทรลเลอร์ที่เราเลือกใช้ โดยดูได้จากโฟล์เดอร์ที่ชื่อ " INC " ในโฟล์เดอร์หลักของโปรแกรม จากนั้นให้ทำการคอมไพล์และสร้าง hex file เพื่อนำไปโปรแกรม (Flash Program) ให้กับไมโครคอนโทรลเลอร์ เพื่อการใช้งานต่อไป

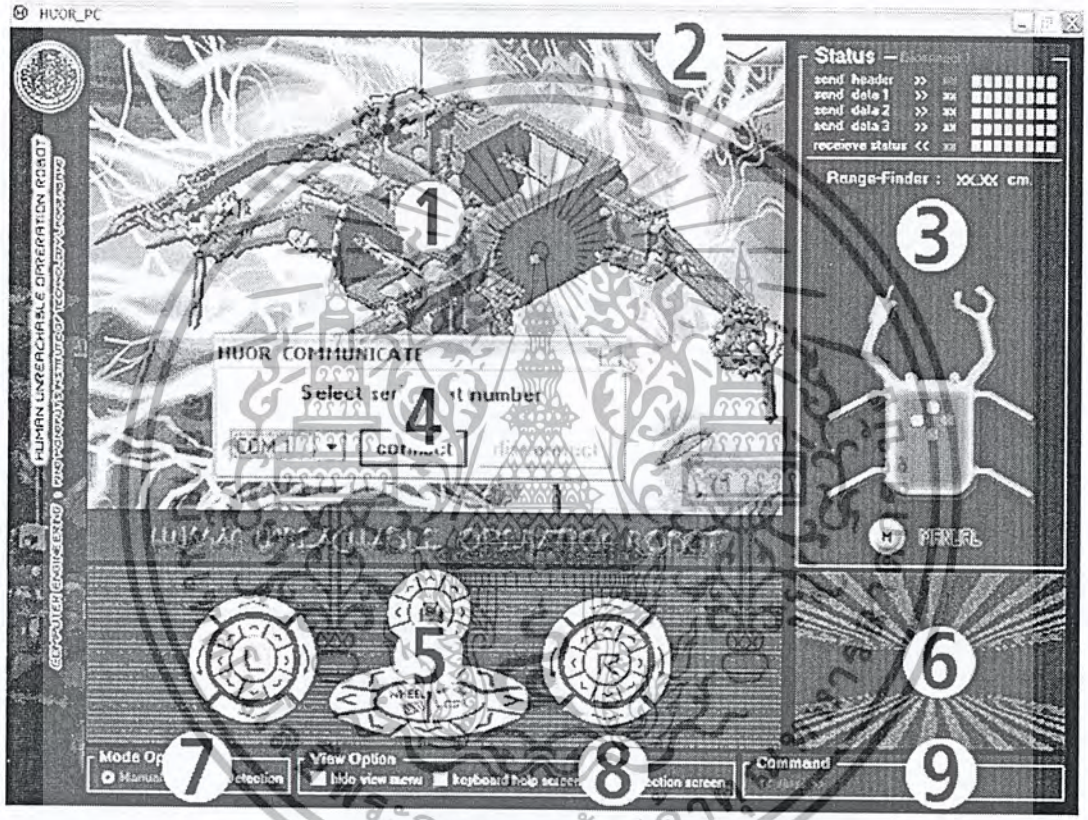


รูปที่ ง-15 แสดงเฮดเดอร์ไฟล์ของโปรแกรม Keil 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ.
คู่มือการใช้งานโปรแกรม

ในการใช้งานโปรแกรมควบคุมการทำงานของหุ่นยนต์ HUOR นั้นสามารถเรียกใช้โปรแกรมได้เลย ไม่ต้องทำการติดตั้งแต่อย่างใด จากรูปด้านล่าง ทำการแบ่งองค์ประกอบของโปรแกรมออกเป็น 9 ส่วน ดังนี้



รูปที่ จ-1 แสดงองค์ประกอบรวมของโปรแกรมควบคุม

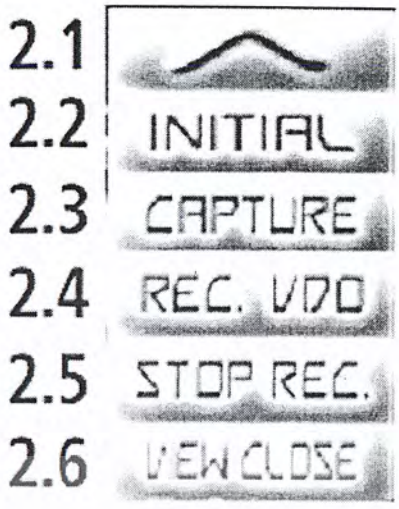
1. ส่วนแสดงสัญญาณภาพจากกล้องวงจรปิด

เป็นส่วนที่ใช้ในการแสดงสัญญาณภาพจากกล้องวงจรปิดโดยในตอนเริ่มโปรแกรมจะแสดงภาพสกรีนของโปรแกรมจนกว่าจะมีการเชื่อมต่อการสื่อสารกับพอร์ตอนุกรม (Connect) หรือจนกว่าจะเริ่มการเชื่อมต่อกับกล้องวงจรปิดไร้สายโดยตรง (Initial view)

2. ส่วนแสดงเมนูการใช้งานฟังก์ชันของกล้องวงจรปิด

เป็นปุ่มที่ใช้แสดงเมนูการใช้งานฟังก์ชันของกล้องวงจรปิด โดยเมื่อเราใช้เมาส์ (Mouse) กดปุ่มแล้วโปรแกรมจะแสดงเมนู ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

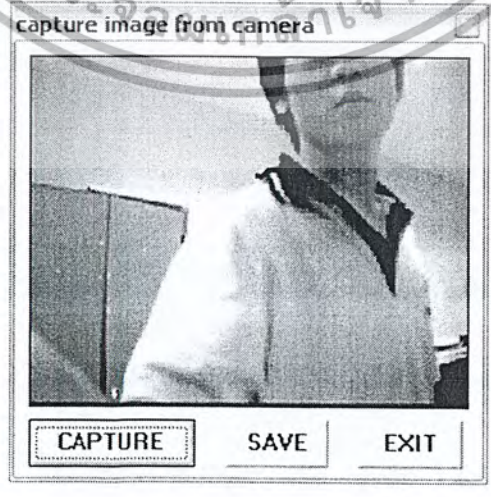


รูปที่ จ-2 แสดงรายละเอียดของเมนูการใช้งานฟังก์ชันของกล้องวงจรปิด

2.1 **Show/Hide Menu Button** เป็นเมนูที่ใช้ในการแสดงหรือซ่อนเมนูการใช้งานฟังก์ชันของกล้องวงจรปิด

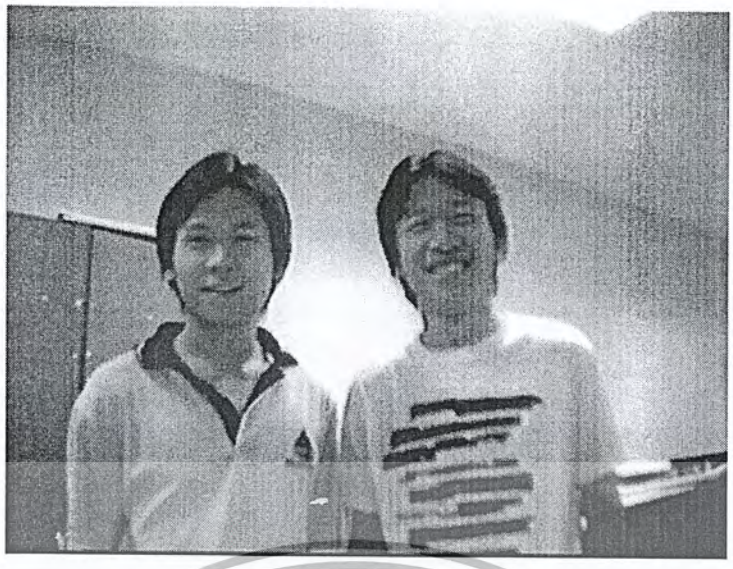
2.2 **Initial Button** เป็นเมนูที่ใช้ในการเริ่มการเชื่อมต่อกับกล้องวงจรปิดโดยตรง โดยไม่สนใจว่าโปรแกรมทำการเชื่อมต่อการสื่อสารผ่านพอร์ตสื่อสารอนุกรมได้หรือไม่

2.3 **Capture Image Button** เป็นเมนูที่ใช้ในการเรียกฟังก์ชันการบันทึกภาพนิ่ง ณ เวลานั้น เก็บเป็นไฟล์รูปภาพเก็บลงในเครื่องคอมพิวเตอร์ โดยสามารถบันทึกทั้ง JPEG (.jpg), Bitmap (.bmp) และ GIF (.gif) โดยค่าตั้งต้นของโปรแกรมคือ JPEG รูปด้านล่างเป็นหน้าต่างที่ปรากฏขึ้นเมื่อเราเรียกใช้ฟังก์ชัน Capture Image เราสามารถทำการเลือกภาพที่ต้องการได้โดยกดปุ่ม "CAPTURE" หากได้ภาพที่ต้องการแล้วให้ทำการกดปุ่ม "SAVE" เพื่อทำการบันทึกรูป จากนั้นให้ทำการตั้งชื่อไฟล์ที่ต้องการ (ไม่จำเป็นต้องใส่นามสกุล เว้นแต่ต้องการบันทึกเป็นไฟล์ชนิดอื่นที่ไม่ใช่ JPEG)



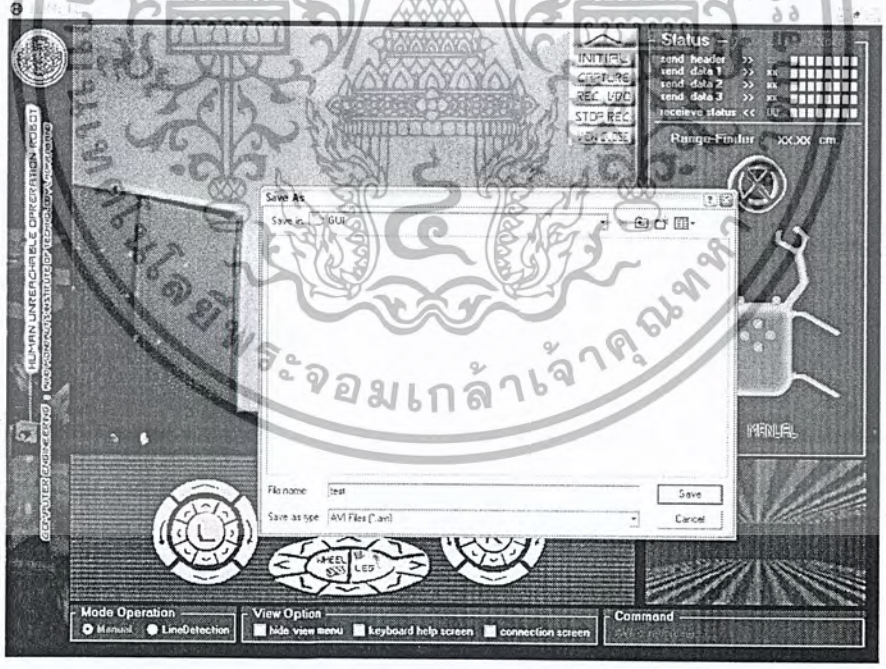
รูปที่ จ-3 แสดงหน้าต่าง Capture Image ที่ปรากฏขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ จ-4 แสดงภาพตัวอย่างที่เกิดจากการบันทึกภาพโดยโปรแกรม

2.4 **Record Video Button** เป็นเมนูที่ใช้ในการเรียกฟังก์ชันการบันทึกภาพวีดีโอจากกล้องวงจรปิดลงในเครื่องคอมพิวเตอร์ ซึ่งจะมีหน้าต่างปรากฏขึ้นมาเพื่อให้เราตั้งชื่อไฟล์วีดีโอที่ทำการบันทึก ดังรูป โดยวีดีโอที่ทำการบันทึกไว้นั้นจะมีนามสกุลเป็น .avi ขนาด 320 x 240 พิกเซล



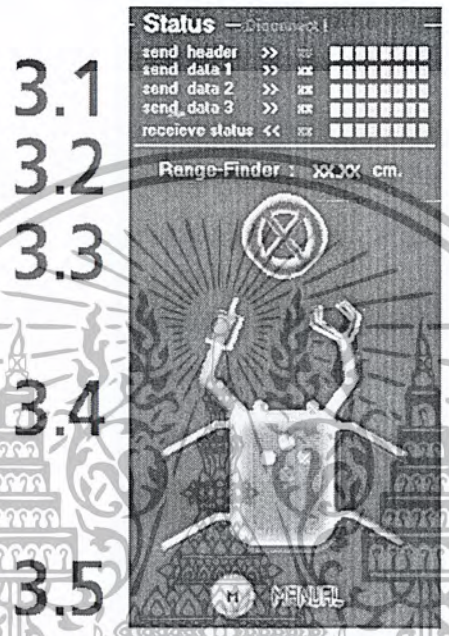
รูปที่ จ-5 แสดงหน้าต่างทำการบันทึกไฟล์วีดีโอ

2.5 **Stop Video record Button** เป็นเมนูที่ใช้ในการยกเลิกการบันทึกวีดีโอ เมื่อทำการกดปุ่มนี้แล้วโปรแกรมจะทำการหยุดบันทึกวีดีโอและทำการบันทึกลงเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 View Close Button เป็นเมนูที่ทำงานตรงกันข้ามกับ Initial Button ใช้ในการยกเลิกการแสดงสัญญาณภาพจากกล้องวงจรปิด โดยไม่ส่งผลกระทบต่อการติดต่อสื่อสารข้อมูลกับหุ่นยนต์ โดยเมื่อผู้ใช้ทำการกดปุ่มแล้วภาพวีดีโอที่หน้าจอจะเปลี่ยนกลับไปเป็นภาพสกรีนของโปรแกรม

3. ส่วนแสดงสถานะการทำงานของหุ่นยนต์ โดยมีรายละเอียด ดังนี้



รูปที่ จ-6 แสดงรายละเอียดส่วนแสดงสถานการณ์ทำงานของหุ่นยนต์

3.1 แสดงสถานะการรับ-ส่ง ข้อมูลในระดับบิต เพื่อให้ผู้ใช้สามารถตรวจสอบการรับ-ส่งข้อมูลระหว่างคอมพิวเตอร์และหุ่นยนต์ HUOR ว่ามีความถูกต้องหรือไม่ โดยจะแสดงบิตข้อมูลตั้งแต่ D7-D0

3.2 แสดงระยะทางที่หาได้จากการคำนวณ ว่าขณะนี้หุ่นยนต์ HUOR อยู่ห่างจากกำแพงเป็นระยะทางเท่าใด โดยแสดงเป็นหน่วยเซนติเมตร (cm.)

3.3 แสดงทิศทางการเคลื่อนที่ ว่าในขณะนี้หุ่นยนต์ HUOR กำลังเคลื่อนที่ไปในทิศทางใด โดยสามารถแสดงการเคลื่อนที่ได้ 9 กรณีคือทิศทางการเคลื่อนที่ 8 ทิศทางและการหยุด ดังรูป

รูปที่ จ-7 แสดงสถานะทิศทางการเคลื่อนที่

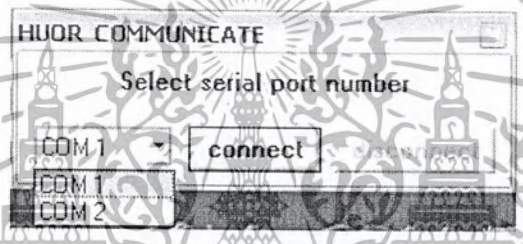
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 แสดงสถานะของอินฟราเรดเซนเซอร์ตามส่วนต่าง ๆ โดยหากอินฟราเรดเซนเซอร์มีการตรวจพบสิ่งกีดขวาง ก็จะมีการส่งสัญญาณเพื่อบอกให้เราทราบ โดยโปรแกรมจะทำการรับข้อมูลมาทำการแสดงผลว่าเซนเซอร์ส่วนใดที่เกิดการเปลี่ยนแปลง

3.5 แสดงโหมดการสั่งการ โดยโปรแกรมจะแสดงโหมดการสั่งการของผู้ใช้ ซึ่งมีอยู่ 2 โหมด คือ โหมดการสั่งการปกติ (Manual Operation Mode) และโหมดการตรวจสอบเส้นทางการเดินด้วยเซนเซอร์ (Line Detection Operation Mode)

4. ส่วนติดต่อสื่อสารผ่านพอร์ตสื่อสารอนุกรม

ผู้ใช้งานสามารถเลือกใช้พอร์ตสื่อสารอนุกรมได้ทั้ง COM1 และ COM2 โดยโปรแกรมจะยอมให้เกิดการเชื่อต่อได้ต่อเมื่อพอร์ตสื่อสารที่ผู้ใช้เลือกนั้นไม่ได้ถูกใช้งานอยู่ และต้องทำการยกเลิกการเชื่อมต่อ (Disconnect) ก่อนทุกครั้งที่จะเปลี่ยนแปลงพอร์ตสื่อสาร ดังรูปด้านล่างแสดงตัวอย่างการเปลี่ยนพอร์ตสื่อสาร



รูปที่ จ-8 แสดงการเปลี่ยนแปลงพอร์ตสื่อสาร

5. ส่วนควบคุมการทำงานของหุ่นยนต์

ผู้ใช้งานสามารถควบคุมการทำงานของหุ่นยนต์ได้ทั้งทางคีย์บอร์ดและเมาส์ และสามารถใช้งานได้พร้อมกัน โดยมีรายละเอียดการใช้งานทางคีย์บอร์ด ดังนี้

ปุ่มคีย์บอร์ด	การทำงานของหุ่นยนต์
Y	เดินหน้า
H	ถอยหลัง
G	เลี้ยวซ้าย
J	เลี้ยวขวา
Q	หมุนข้อมือซ้ายไปทางซ้าย
W	หมุนมือสว่านไปทางซ้าย
E	หมุนมือสว่านไปทางขวา
R	หมุนข้อมือซ้ายไปทางขวา
A	เคลื่อนแขนซ้ายไปทางซ้าย
S	เคลื่อนแขนซ้ายลง

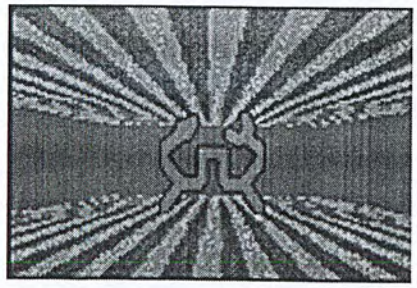
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปุ่มคีย์บอร์ด	การทำงานของหุ่นยนต์
D	เคลื่อนแขนซ้ายลง
F	เคลื่อนแขนซ้ายไปทางขวา
I	หมุนข้อมือขวาไปทางซ้าย
O	ปล่อยมือ
P	หนีบวัตถุ
[	หมุนข้อมือขวาไปทางขวา
K	เคลื่อนแขนขวาไปทางซ้าย
L	เคลื่อนแขนขวาลง
;	เคลื่อนแขนขวาลง
'	เคลื่อนแขนขวาไปทางขวา
num 8	เลื่อนมุมกล้องขึ้น
num 2	เลื่อนมุมกล้องลง
num 4	เลื่อนมุมกล้องไปทางซ้าย
num 6	เลื่อนมุมกล้องไปทางขวา
T	เปิด-ปิดไฟ
U	เปลี่ยนโหมดการเคลื่อนที่

ตารางที่ จ-1 แสดงปุ่มคีย์บอร์ดที่ใช้ควบคุมหุ่นยนต์

6. ส่วนควบคุมการเปลี่ยนมุมกล้องด้วยเมาส์

ในส่วนนี้จะเป็นการเปลี่ยนมุมกล้องด้วยเมาส์ ที่จะช่วยให้ผู้ใช้สามารถควบคุมการทำงานของหุ่นยนต์ด้วยคีย์บอร์ดและเมาส์ไปพร้อมกันได้สะดวกและมีประสิทธิภาพยิ่งขึ้น โดยให้ผู้ใช้ทำการลากเมาส์ไปในบริเวณที่เป็นพื้นที่ที่จัดไว้ โดยสังเกตได้จากเคอร์เซอร์ (Cursor) ของเมาส์จะเปลี่ยนไปดังรูป จากนั้นให้ผู้ใช้ทำการกดเมาส์ซ้าย 1 ครั้ง โปรแกรมก็จะเข้าสู่ฟังก์ชันการเปลี่ยนมุมกล้องด้วยเมาส์ทันที หากต้องการเข้าสู่การทำงานปกติ ให้ผู้ใช้ทำการกดเมาส์ขวา 1 ครั้งจะเป็นการยกเลิกการใช้เมาส์ในการเปลี่ยนมุมกล้องเข้าสู่การทำงานแบบปกติ



รูปที่ จ-9 แสดงการเปลี่ยนมุมกล้องด้วยเมาส์

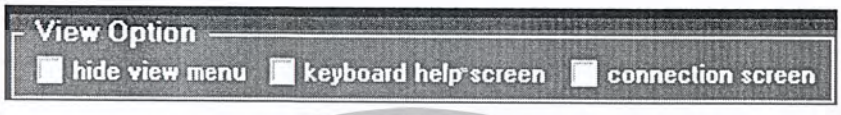
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ส่วนการเปลี่ยนโหมดการสั่งการ

ในส่วนนี้จะเป็นการเปลี่ยนแปลงโหมดการสั่งการของผู้ใช้ให้หุ่นยนต์ได้รู้ถึงความต้องการ ซึ่งมีอยู่ 2 โหมดคือ โหมดการสั่งการปกติ (Manual Operation Mode) และโหมดการตรวจสอบเส้นทางการเดินด้วยเซนเซอร์ (Line Detection Operation Mode)

8. ส่วนแสดงมุมมองของผู้ใช้

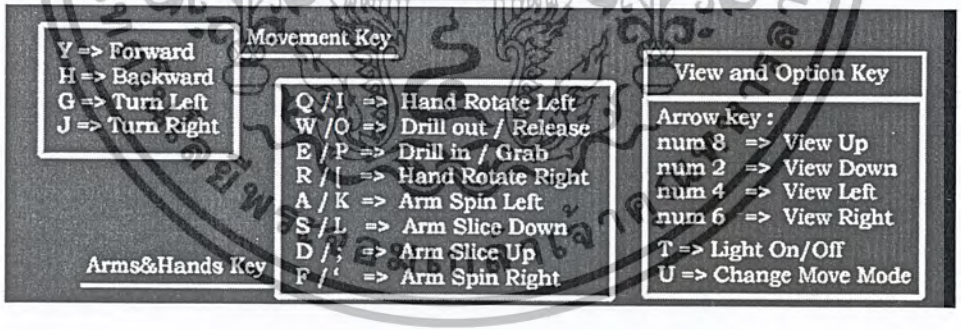
ซึ่งมีอยู่ 3 ส่วนที่อนุญาตให้ผู้ใช้สามารถแสดงหรือซ่อนหน้าต่างการใช้งานได้คือ



รูปที่ จ-10 แสดงการปรับเปลี่ยนมุมมองของผู้ใช้

8.1 การแสดง/ซ่อนเมนูการใช้งานของกล้องวงจรปิด โดยหากผู้ใช้งานไม่ต้องการให้โปรแกรมแสดงในส่วนของเมนูการใช้งานของกล้องวงจรปิด ซึ่งโดยปกติจะอยู่บนมุมขวาด้านบนของภาพวีดีโอแสดงอยู่ที่หน้าจอ

8.2 การแสดง/ซ่อนหน้าจอช่วยเหลือการใช้งานคีย์บอร์ด ซึ่งในการใช้งานโปรแกรมเพื่อควบคุมหุ่นยนต์นั้น ในบางครั้งผู้ใช้งานอาจสับสนหรือเกิดความสังเลในการใช้งานปุ่มคีย์บอร์ด ซึ่งผู้ใช้งานสามารถทำการแสดงหน้าจอช่วยเหลือการใช้งานคีย์บอร์ดได้ ดังรูป



รูปที่ จ-11 แสดงหน้าจอช่วยเหลือการใช้งานคีย์บอร์ด

8.3 การแสดง/ซ่อนหน้าจอการติดต่อสื่อสาร เนื่องจากว่าหน้าจอการติดต่อสื่อสารนั้นจะปรากฏกลางหน้าจอและไม่สามารถเคลื่อนย้ายได้ ผู้ใช้งานสามารถซ่อนหน้าจอนี้ได้ โดยในทุกครั้งที่เปิดโปรแกรมขึ้นมา จะปรากฏหน้าจอการติดต่อสื่อสารขึ้นมาเสมอ

9. ส่วนแสดงคำสั่งของผู้ควบคุม

เป็นส่วนแสดงคำสั่งของผู้ใช้งานว่ามีการสั่งการให้หุ่นยนต์ทำอะไร โดยจะแสดงทุกครั้งที่มีการสั่งการอย่างใดอย่างหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

หนังสืออ้างอิง

[1] นุกูล กระจาย, "การเขียนโปรแกรมในดอสและวินโดวส์ ด้วยบอร์แลนด์ C++ 5.0", ซีเอ็ดยูเคชั่น, 2544.

[2] ธีรวัฒน์ ประกอบผล, "การประยุกต์ใช้งานไมโครคอนโทรลเลอร์", สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), 2544.

[3] ธีรวัฒน์ ประกอบผล, "การพัฒนาไมโครคอนโทรลเลอร์ด้วยภาษาซี", สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), 2545.

[4] วรพจน์ กรแก้ววัฒนกุล และชัยวัฒน์ ลิ้มพรจิตรวิไล, "เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช ฉบับ AT89C5X ของ Atmel", อินโนเวตีฟ เอ็กเพอริเมนต์, -

[5] -, "Visual Basic 6.0 Win32 API เทคนิคและการประยุกต์", อัปเปอร์ เมเนจเมนต์, -

[6] อภิเนตร อูนากูล, "กระบวนการและวิธีการพัฒนาเว็บแอพพลิเคชันโดยใช้ UML", แผนกตำรา คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2546.

[7] อภิชาติ ถู่พลับ, "เริ่มต้นเขียนโปรแกรมติดต่อและควบคุมฮาร์ดแวร์ด้วย Visual Basic", อินโฟเพรส, 2546.

[8] อรรถพล บุญยะโภคา วรพจน์ กรแก้ววัฒนกุล และชัยวัฒน์ ลิ้มพรจิตรวิไล, "เรียนรู้และปฏิบัติการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่านพอร์ตอนุกรม", อินโนเวตีฟ เอ็กเพอริเมนต์, -

[9] สัญญา นามี, "เริ่มต้นอย่างมืออาชีพด้วย AutoCAD 2002", อินโฟเพรส, 2545.

เว็บไซต์อ้างอิง

[1] http://www.acroname.com

[2] http://www.lynxmotion.com

[3] http://www.thaizone99.com

[4] http://www.naijiw.com

[5] http://www.ee.uwa.edu.au

[6] http://www.thaiio.com

[7] http://www.vbcode.com

[8] http://www.tesa.or.th

[9] http://www.hobby.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้