สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การผลิตผสมผสานด้วยคอมพิวเตอร์ : ส่วนหุ่นยนต์เสมือนมนุษย์

COMPUTER INTERGRATED MANUFACTURING: HUMANOID ROBOT SECTION

T119419

นาย ณฐพล จึงพัฒนปรีชา

นาย ธนเดช จันทราพานิชกุล

นาย ธนวรรธน์ ทิวาสัมฤทธิ์

นาย ธนวัฒน์ ดำรงกิจชัยพร

เลขหมู่.....
เลขทะเบียน 119419
วัน,เดือน,ปี - 7 S.A. 2554

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

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมแมคคาทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปีการศึกษา 2553

COMPUTER INTEGRATED MANUFACTURING :

HUMANOID ROBOT SECTION

Mr.Natapol Jungpattanapreecha

Mr.Tanadech Chantrapanichkul

Mr.Thanawat Thiwasamrit

Mr.Thanawat Damrongkitchaiporn

THIS THESIS IS SUBMITTED IN PARTAIL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN MECHATRONICS ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2010

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การผลิตผสมผสานด้วยคอมพิวเตอร์ : ส่วนหุ่นยนต์เสมือนมนุษย์

COMPUTER INTERGRATED MANUFACTURING: HUMANOID ROBOT SECTION

ผู้จัดทำ	นาย ณฐพล	จึงพัฒนปรีชา	50010416
	นาย ธนเดช	จันทราพานิชกุล	50010601
	นาย ธนวรรธน์	ทิวาสัมฤทธิ์	50010622
	นาย ธนวัฒน์	ดำรงกิจชัยพร	50010623

.....อาจารย์ที่ปรึกษา

(รองศาสตราจารย์ ดร.วันชัย ริ้วรุจา)

.....อาจารย์ที่ปรึกษา

(ดร.วรรณดี เพชรมณีล้ำค่า)

.....อาจารย์ที่ปรึกษา

(อาจารย์ เทพจิตร์ เชยโภคา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของผลงานก่อนการนำไปใช้

การผลิตผสมผสานด้วยคอมพิวเตอร์ : ส่วนหุ่นยนต์เสมือนมนุษย์

โดย

นาย ณฐพล จึงพัฒนปรีชา 50010416

นาย ธนเดช จันทราพานิชกุล 50010601

นาย ธนวรรธน์ ทิวาสัมฤทธิ์ 50010622

นาย ธนวัฒน์ ดำรงกิจชัยพร 50010623

อาจารย์ที่ปรึกษา

รองศาสตราจารย์ ดร. วันชัย ริ้วรุจา

ดร. วรรณดี เพชรมณีล้ำค่า

อาจารย์ เทพจิตร์ เชยโภคา

ปีการศึกษา 2553

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ นำเสนอหุ่นยนต์เสมือนมนุษย์ โดยโครงสร้างของหุ่นยนต์ประกอบแผ่นอะคริลิค เซอร์โวมอเตอร์ และวงจรอิเล็กทรอนิกส์สำเร็จรูป จุดมุ่งหมายเพื่อนำสิ่งที่ได้เรียนรู้ในสาขาวิชาแมคคาทรอนิกส์มานำมาประยุกต์ใช้ในการทำปริญญานิพนธ์ ไม่ว่าจะเป็นการใช้เครื่องคอมพิวเตอร์ควบคุมเชิงตัวเลข การใช้โปรแกรมภาษาซี เป็นต้น

ขั้นตอนการดำเนินการ เริ่มจากการศึกษาและออกแบบตัวหุ่นยนต์เสมือนมนุษย์โดยใช้โปรแกรมโซลิดเวิร์ค และเครื่องคอมพิวเตอร์ควบคุมเชิงตัวเลข ซึ่งแต่ละข้อต่อของตัวหุ่นยนต์เสมือนมนุษย์จะถูกขับเคลื่อนด้วยเซอร์โวมอเตอร์ แล้วจึงศึกษาและออกแบบวงจรอิเล็กทรอนิกส์ของหุ่นยนต์ได้แก่ วงจรควบคุมการทำงานของตัวหุ่นยนต์ และวงจรสวิตซ์ควบคุม จากการศึกษาพบว่าหุ่นยนต์เสมือนมนุษย์สามารถทำได้ตามท่าทางที่กำหนดไว้ไม่ว่าจะเป็นท่าทางการเดิน ท่าทางการเดินเลี้ยวซ้ายเลี้ยวขวา ท่าทางการไหว้

ไม่อนุญาตให้ทำซ้ำ ดัดแปลง เผยแพร่ หรือนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTER INTEGRATED MANUFACTURING

: HUMANOID ROBOT SECTION

By

Mr.Natapol Jungpattanapreecha

Mr.Tanadech Juntrapanichkul

Mr.Thanawat Thiwasamrit

Mr.Thanawat Damrongkitchaiporn

Advisors

Assoc.Prof. Dr. Vanchai Riewruja

Dr. Wandee Petchmaneelumka

Mr. Thepjit Cheypoca

Academic Year 2010

Abstract

This thesis presents humanoid robot. The structure of robot comprises few pieces of acrylic, servo motor and electronic circuits. The purpose of this thesis is to apply knowledge in field of mechanics engineering for humanoid robot such as computer numerical control machine and C programming language.

Procedure of this thesis begins at study and design humanoid robot with Solid works program and computer numerical control (CNC). Each joint of robot is driven by servo motors. Then, electronics circuits are studied and designed for robot such as switch and control circuits. The results shows humanoid robot can walk, turn left, turn right and gesture of respect.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์ฉบับนี้ สามารถสำเร็จลุล่วงไปได้ด้วยดี เพราะได้รับความช่วยเหลือเป็นอย่างดีจาก รศ.ดร.วันชัย ริ้วรุจา , ดร.วรรณดี เพชรมณีล้ำค่า , อ.เทพจิตร เชยโภคา ที่กรุณาให้คำปรึกษาแนะนำที่ดีมาโดยตลอดตั้งแต่ต้น รวมทั้งเอื้อเฟื้ออุปกรณ์ที่จำเป็น และความช่วยเหลืออื่น ๆที่เป็นประโยชน์ต่อโครงงาน ผู้จัดทำรู้สึกซาบซึ้งและขอกราบขอบพระคุณอย่างสูง

ขอบคุณเพื่อน ๆ ทุกคน ที่รวมแรง ร่วมใจกันทำโครงงานชิ้นนี้ ให้ความช่วยเหลือในการแบ่งปันข้อมูล การเสียสละ และ การให้อภัยซึ่งกัน รวมถึงการช่วยเหลือของบุคคลภายนอกที่เข้ามาช่วยในส่วนของชิ้นงาน หรือ การให้ข้อมูลความรู้ที่เป็นประโยชน์ในการทำโครงงาน

สุดท้ายนี้ผู้จัดทำขอกราบขอบพระคุณบิดา มารดา และครอบครัว ที่คอยเป็นกำลังใจที่ดีตลอดมา รวมถึงสนับสนุนในเรื่องของงบประมาณที่ขาดเหลือ ตลอดจนเป็นแรงบันดาลใจที่ดีที่สุดที่ทำให้โครงงานนี้สำเร็จสมบูรณ์ลงได้

ผู้จัดทำ

นาย ณฐพล จึงพัฒนปรีชา

นาย ธนเดช จันทราพานิชกุล

นาย ธนวรรธน์ ทิวาสัมฤทธิ์

นาย ธนวัฒน์ ดำรงกิจชัยพร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญภาพ VII

สารบัญตาราง IX

บทที่ 1 บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา 1

1.3 ขอบเขตของการศึกษา 2

1.4 รายละเอียดของปริญญานิพนธ์ 2

บทที่ 2 การผลิตผสมผสานด้วยคอมพิวเตอร์

2.1 ภาพรวมของระบบการผลิตผสมผสานด้วยคอมพิวเตอร์ 3

2.2 ปัจจัยที่ท้าทาย 4

2.3 ส่วนประกอบภายในระบบจำลองกระบวนการการผลิต 4

2.3.1 ระบบลำเลียง 4

2.3.1.1 รถลำเลียงสินค้า 4

2.3.1.2 สถานีรับ-ส่งสินค้า 5

2.3.2 แขนกล 5

2.3.2.1 ส่วนควบคุม 5

2.3.2.2 ส่วนประมวลผล 6

2.3.2.3 ส่วนเซอร์โวมอเตอร์ 6

2.3.2.4 ส่วนที่ประกอบเป็นแขนกล 6

2.3.3 ส่วนแสดงผล 6

2.4 หลักการทำงานของระบบ 7

2.4.1 โหมดควบคุมด้วยมือ 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.4.2 โหมดอัตโนมัติ 8

2.5 การติดต่อสื่อสารกันของระบบ 8

บทที่ 3 หุ่นยนต์สองขา (Humanoid)

3.1 ขั้นตอนการทำงาน 9

3.2 เครื่องมือและอุปกรณ์ 10

3.2.1 ฮาร์ดแวร์ 10

3.2.2 ซอฟต์แวร์ 10

3.3 การออกแบบโครงสร้างหุ่นยนต์ 10

3.3.1 ตัวอย่างชิ้นส่วนหุ่นยนต์จากโปรแกรมโซลิดเวิร์ค 12

3.4 มอเตอร์กระแสตรง 15

3.4.1 การควบคุมทิศทางการหมุนของมอเตอร์กระแสตรง 15

3.4.2 การควบคุมความเร็วของมอเตอร์กระแสตรง 15

3.5 เซอร์โวมอเตอร์ 16

3.5.1 ความหมายของเซอร์โวมอเตอร์ 16

3.5.2 ส่วนประกอบภายใน 16

3.5.3 หลักการทำงานของเซอร์โวมอเตอร์ 16

3.6 ส่วนประกอบของไมโครคอนโทรลเลอร์ 19

3.6.1 หน่วยประมวลผล 19

3.6.2. หน่วยความจำ 19

3.6.3. พอร์ทสัญญาณเข้าและสัญญาณออก 19

3.6.4. คุณสมบัติอื่นๆ 19

3.6.5. เอวีอาร์ ไมโครคอนโทรลเลอร์ 19

3.7 หลักการของพัลส์ วิด มอดดูเรชัน (Pulse Width Modulation, PWM) 19

3.8 วงจรอิเล็กทรอนิกส์สำเร็จรูป 20

3.9 เซอร์โวมอเตอร์ที่ใช้ในโครงงาน 22

3.10 วงจรของหุ่นยนต์เสมือนมนุษย์ 25

3.10.1 วงจรบอร์ดสวิตซ์ควบคุม (Keypad Joystick) 26

3.11 หลักการทำงาน 28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 4 การทดลองและผลการทดลอง

4.1 การออกแบบท่าต่างๆให้หุ่นยนต์ 29

4.2 ท่าและการเคลื่อนไหวของหุ่นยนต์เสมือนมนุษย์ 30

4.2.1 ท่ายืนตรง 30

4.2.2 ท่ายืนย่อเข่า 30

4.2.3 ท่ายืนโค้งคำนับ 30

4.2.4 ท่ายืนสวัสดี 31

4.2.5 ท่าเดินไปข้างหน้าระยะการก้าวสั้น 31

4.2.6 ท่าเดินไปข้างหน้าระยะการก้าวยาว 32

4.2.7 ท่าเดินถอยหลังระยะการก้าวยาว 32

4.2.8 ท่าหันตัวไปทางซ้าย 33

4.2.9 ท่าหันตัวไปทางขวา 33

บทที่ 5 บทวิจารณ์และสรุป

5.1 สรุปผลการทดลอง 34

5.2 ปัญหาที่และอุปสรรค 34

5.3 แนวทางการแก้ไข 35

5.4 ข้อเสนอแนะและแนวทางในการค้นคว้าพัฒนา 35

ภาคผนวก ก โปรแกรมการออกแบบและผลิตชิ้นส่วนหุ่นยนต์ 37

ภาคผนวก ข เอกสารคู่มืออุปกรณ์อิเล็กทรอนิกส์ 45

ภาคผนวก ค โปรแกรมควบคุมหุ่นเสมือนมนุษย์ 51

ภาคผนวก ง โปสเตอร์หุ่นยนต์สองขา 70

เอกสารอ้างอิง 71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่ หน้า

2.1 รถลำเลียงสินค้า 5

2.2 สถานีรับ-ส่งสินค้า 5

2.3 หุ่นยนต์แขนกล 6

2.4 ส่วนแสดงผล 7

2.5 การทำงานของระบบการผลิตผสมผสานด้วยคอมพิวเตอร์ 7

3.1 แผนภาพแสดงขั้นตอนการทำงานของหุ่นยนต์ 9

3.2 ส่วนประกอบของหุ่นยนต์สองขา 11

3.3 โครงสร้างหุ่นยนต์สองขา 11

3.4 ชิ้นส่วนลำตัว 1 12

3.5 ชิ้นส่วนลำตัว 2 12

3.6 ชิ้นตัวลำตัวด้านล่าง 13

3.7 ชิ้นส่วนลำตัวด้านบน 13

3.8 ชิ้นส่วนแขน 13

3.9 ชิ้นส่วนหัว 14

3.10 ชิ้นส่วนข้อต่อของน่อง 14

3.11 ชิ้นส่วนของเซอร์โวมอเตอร์ 17

3.12 โครงสร้างของเซอร์โวมอเตอร์ 16

3.13 สัญญาพัลส์กับเซอร์โวมอเตอร์ 18

3.14 การมอดูเลตความกว้างพัลส์ (Pulse Width Modulation, PWM) 20

3.15 ตัวอย่างการต่อ บอร์ดอินพุท/เอาท์พุท ของ อีทีที ด้วยขั้ว 10 ขา 20

3.16 ตัวอย่างการต่อ บอร์ดอินพุท/เอาท์พุท ของ อีทีที ด้วยขั้ว 10 ขา 21

3.17 เซอร์โวมอเตอร์ รุ่น MG945 22

3.18 เซอร์โวมอเตอร์รุ่น S3003 23

3.19 Digital servo motor EK 2-0508 24

3.20 วงจรของหุ่นยนต์สองขา 25

3.21 วงจรบอร์ดสวิตซ์ควบคุม (Keypad Joystick) 26

เอกสารนี้มีไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ(ต่อ)

รูปที่ **หน้า**

3.22 บอร์ดสวิตซ์ควบคุม (Keypad Joystick) 26

3.23 การเชื่อมต่อเซอร์โวมอเตอร์กับขาของเอวีอาร์ เมก้า 1280 27

3.24 การเชื่อมต่อเซอร์โวมอเตอร์กับขาของเอวีอาร์ เมก้า 1280 27

3.25 วงจรบอร์ดสวิตซ์ควบคุม ต่อกับ วงจรเอวีอาร์ 28

4.1 ตำแหน่งเซอร์โวมอเตอร์ แต่ละตัวของหุ่นยนต์เสมือนมนุษย์ 29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

4.1 มุมองศาของเซอร์โวมอเตอร์แต่ละตัวในท่ายืนตรง 30

4.2 มุมองศาของเซอร์โวมอเตอร์แต่ละตัวในท่ายืนย่อเข่า 30

4.3 ของเซอร์โวมอเตอร์แต่ละตัวในท่ายืนโค้งคำนับ 30

4.4 มุมองศาของเซอร์โวมอเตอร์แต่ละตัวในท่ายืนสวัสดี 31

4.5 มุมองศาของเซอร์โวมอเตอร์แต่ละตัวในท่าเดินไปข้างหน้าระยะการก้าวสั้น 31

4.6 มุมองศาของเซอร์โวมอเตอร์แต่ละตัวในท่าเดินไปข้างหน้าระยะการก้าวยาว 32

4.7 มุมองศาของเซอร์โวมอเตอร์แต่ละตัวในท่าเดินถอยหลังระยะการก้าวยาว 32

4.8 มุมองศาของเซอร์โวมอเตอร์แต่ละตัวในท่าหันตัวไปทางซ้าย 33

4.9 มุมองศาของเซอร์โวมอเตอร์แต่ละตัวในท่าหันตัวไปทางขวา 33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การศึกษาในสาขาวิศวกรรมแมคคาทรอนิกส์ เป็นการศึกษาและประยุกต์ใช้ทฤษฎีต่างๆ ในการออกแบบและควบคุมให้ระบบมีประสิทธิภาพมากขึ้น ซึ่งปัจจุบันวิศวกรรมแมคคาทรอนิกส์ได้เข้ามามีบทบาททางด้านการพัฒนาระบบอุตสาหกรรมมากขึ้น เนื่องจากมีการขยายตัวทางอุตสาหกรรมที่มีแนวโน้มเพิ่มขึ้นเรื่อยๆ เพื่อให้การผลิตมีประสิทธิภาพยิ่งขึ้น อุตสาหกรรมการผลิตแบบอัตโนมัติจึงเข้ามามีบทบาทในวงการอุตสาหกรรมผนวกเข้ากับเทคโนโลยีหุ่นยนต์ จึงได้มีการคิดค้นและพัฒนาหุ่นยนต์ในรูปแบบต่างๆในระบบอุตสาหกรรมมากยิ่งขึ้น เนื่องจากมีความต้องการใช้หุ่นยนต์ที่สามารถนำมาใช้งานได้จริง มีความสามารถในการขนส่งสินค้า มีการประมวลผลได้อย่างรวดเร็ว และมีความใกล้เคียงกับสิ่งมีชีวิต ด้วยเหตุนี้ได้มีการคิดค้นและจำลองระบบอุตสาหกรรมขึ้น โดยประกอบไปด้วยรถที่ใช้ในระบบขนส่ง และหุ่นยนต์ โดยหุ่นยนต์ส่วนใหญ่เป็นแขนกลชนิดยึดอยู่กับที่ (fixed robot) และหุ่นยนต์แบบเคลื่อนที่ด้วยขา (walking robot) ซึ่งมีหุ่นยนต์ 2 ขา (humanoid robot) และหุ่นยนต์ 6 ขา (hexapods robot)

หุ่นยนต์ที่เคลื่อนที่ด้วยขาสองขาหรือหุ่นยนต์เสมือนมนุษย์ มีความสามารถในการเคลื่อนที่ได้ดีกว่าหุ่นยนต์ที่เคลื่อนที่ด้วยขาแบบอื่น แต่เนื่องจากหุ่นยนต์สองขามีส่วนประกอบของโครงสร้างหลายชิ้นส่วนจึงทำให้การควบคุมให้หุ่นยนต์มีเสถียรภาพทั้งขณะที่อยู่กับที่และก้าวเดินเป็นเรื่องที่ค่อนข้างยากและซับซ้อน ดังนั้นในปริญญานิพนธ์ฉบับนี้จึงได้มีการทำวิจัยการควบคุมดังกล่าว

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1.2.1 เพื่อศึกษาการทำงานของระบบผสมผสานด้วยคอมพิวเตอร์ (Computer Integrated Manufacturing,CIM)

1.2.2 ทำการศึกษาและประดิษฐ์หุ่นยนต์เสมือนมนุษย์ ซึ่งประกอบด้วยอุปกรณ์ที่ ประกอบเป็นโครงสร้างทางกล อุปกรณ์ทางอิเล็กทรอนิกส์ที่เป็นระบบควบคุม รวมถึงส่วนของการควบคุมและประมวลผล

1.2.3 ทำการออกแบบชิ้นงานที่ใช้ต้นทุนในการผลิตต่ำ และเลือกใช้วัสดุที่มีความเหมาะสมกับการทำชิ้นงาน เพื่อการจำลองระบบจากงานอุตสาหกรรม

1.2.4 สามารถนำไปพัฒนาต่อยอดในเชิงพาณิชย์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของการศึกษา

1.3.1 หุ่นยนต์สองขาสามารถเคลื่อนไหวได้

1.3.2 หุ่นยนต์สองขาสามารถแสดงท่าทางได้ตามที่กำหนดได้

1.3.3 สามารถเป็นแนวทางในการประดิษฐ์หุ่นยนต์ที่ใช้ในเชิงอุตสาหกรรมได้

1.4 รายละเอียดของปริญญานิพนธ์

เนื้อหาที่จะกล่าวในปริญญานิพนธ์ฉบับนี้ประกอบด้วย

บทที่ 1 บทนำ กล่าวนำถึงความเป็นมาและความสำคัญของปัญหา ความมุ่งหมายและวัตถุประสงค์ของการศึกษา ขอบเขตการศึกษา และรายละเอียดของปริญญานิพนธ์ของแต่ละบท

บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง กล่าวถึง การผลิตผสมผสานด้วยคอมพิวเตอร์ รวมถึงการจำลองกระบวนการการผลิต ทั้งแบบควบคุมด้วยมือ และควบคุมอัตโนมัติ

บทที่ 3 หลักการออกแบบและนำเสนอโครงสร้างของระบบ รวมถึงแนวคิดในการเขียนโปรแกรมควบคุมหุ่นยนต์

บทที่ 4 การทดลอง เป็นส่วนของการทดสอบองค์ประกอบต่างๆในระบบ ตลอดจนการทดลองโปรแกรมที่ใช้ในการควบคุม

บทที่ 5 บทวิจารณ์และสรุป จะสรุปผลการดำเนินงาน ปัญหาที่เกิดขึ้น และข้อเสนอแนะในการปรับปรุงโครงงานนี้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
การผลิตผสมผสานด้วยคอมพิวเตอร์

การผลิตผสมผสานด้วยคอมพิวเตอร์ (Computer Integrated Manufacturing ,CIM) เป็นระบบการผลิตที่ใช้คอมพิวเตอร์เข้ามาควบคุมกระบวนการผลิตทั้งหมด การผสมผสานของระบบทำให้มีการแลกเปลี่ยนข้อมูลระหว่างหน่วยงานแต่ละหน่วยได้ ทำให้แต่ละหน่วยรับรู้ความก้าวหน้าซึ่งกันและกัน ข้อดี คือ ระบบการผลิตจะมีความรวดเร็วและมีข้อผิดพลาดน้อย แม้ว่าข้อดีหลักของ CIM คือ ความสามารถในการสร้างกระบวนการผลิตอัตโนมัติ โดยทั่วไปแล้วระบบ CIM จะเป็นกระบวนการควบคุมแบบปิด (Closed-loop Control Processes) บนพื้นฐานของข้อมูล ณ ปัจจุบันที่ได้รับจากตัวตรวจรู้ (Sensor)

2.1 ระบบการผลิตผสมผสานด้วยคอมพิวเตอร์

Computer Integrated Manufacturing (CIM) เป็นทั้งกระบวนการผลิตและชื่อของระบบอัตโนมัติที่ควบคุมด้วยคอมพิวเตอร์ โดยมีหน้าที่สนับสนุนการทำงานและเป็นระบบการจัดการของระบบการผลิตที่ประกอบ ด้วยฝ่ายต่างๆ เช่น ฝ่ายงานวิศวกรรม ฝ่ายงานการผลิต ฝ่ายการตลาด และฝ่ายการสนับสนุนอื่นๆ ขอบข่ายหน้าที่การทำงานของการผลิตผสมผสานด้วยคอมพิวเตอร์ มีหลากหลายอย่าง เช่น ออกแบบ วิเคราะห์ วางแผน จัดซื้อ จัดการบัญชีต้นทุน ควบคุมคงคลัง และการกระจายผลิตภัณฑ์ เหล่านี้จะถูกเชื่อมโยงโดยคอมพิวเตอร์ไปยังอุปกรณ์ หรือหน่วยต่างๆ ภายในองค์กรการผลิตผสมผสานด้วยคอมพิวเตอร์ จะทำให้สามารถควบคุมกระบวนการได้โดยตรงและสามารถแสดงการทำงานปัจจุบันของ ทุกกระบวนการทำงาน

ข้อแตกต่าง 3 ประการ ที่ทำให้การผลิตผสมผสานด้วยคอมพิวเตอร์แตกต่างจาก ระบบการผลิตแบบอื่นๆ คือ

1. สื่อที่ใช้ในการเก็บข้อมูล การคืนสภาพกระบวนการ การควบคุม และการนำเสนอ
2. กลไกการตรวจจับและการดัดแปลงกระบวนการต่างๆ
3. อัลกอรึทึมของการประมวลผลข้อมูลที่ได้จากการตรวจจับและการดัดแปลงส่วนประกอบต่างๆ

การผลิตผสมผสานด้วยคอมพิวเตอร์ จะประกอบด้วยคอมพิวเตอร์ที่มีการแลกเปลี่ยนข้อมูลระหว่างกัน 2 เครื่องขึ้นไป เช่น ตัวควบคุมหุ่นยนต์อุตสาหกรรม (Industrial Robot) กับเครื่องคอมพิวเตอร์ควบคุมเชิงตัวเลข (Computer Numerical Control ,CNC) สิ่งที่ต้องคำนึงถึงในการติดตั้งระบบ การผลิตผสมผสานด้วยคอมพิวเตอร์ คือ ปริมาณการผลิต ประสบการณ์ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

องค์การและบุคลากร ระดับการผสมผสานของส่วนการผลิตและส่วนต่างๆ การผลิตผสมผสานด้วยคอมพิวเตอร์มีประโยชน์มากที่สุดในองค์การที่มีระดับการใช้ข้อมูลสารสนเทศภายในองค์กรสูง

2.2 ปัจจัยที่ท้าทาย

สิ่งที่ท้าทายในการพัฒนาระบบการผลิตผสมผสานด้วยคอมพิวเตอร์ ให้มีประสิทธิภาพ มีอยู่ 3 สิ่งหลักๆ ด้วยกัน คือ

2.2.1 การผสมผสานของส่วนประกอบต่างๆ ที่มาจากหลายๆซัพพลายเออร์ ; เมื่อมีเครื่องจักรที่แตกต่างกัน เช่น การผลิตผสมผสานด้วยคอมพิวเตอร์ สายพานลำเลียง และหุ่นยนต์ อุปกรณ์เชื่อมต่อย่อมแตกต่างกันไป กรณีของระบบขนถ่ายวัสดุอัตโนมัติ (Automated Guided Vehicle System, AGV) ก็มีระยะเวลาของแบตเตอรี่ที่แตกต่างกัน ซึ่งทำให้เกิดปัญหาในการจัดการระบบ

2.2.2 การผสมผสานของข้อมูล ในการระบบอัตโนมัติระดับสูงๆ การผสมผสานระหว่างข้อมูลที่ใช้ในเครื่องจักรและจากแรงงานที่ควบคุมเครื่อง จักร ต้องมีความเหมาะสมถูกต้อง เพื่อไม่ให้เกิดความผิดพลาดของสัญญาณที่ใช้ในการควบคุมเครื่องจักร

2.2.3 การควบคุมกระบวนการ คอมพิวเตอร์ถูกใช้ในการช่วยมนุษย์ในการดำเนินงาน ส่วนใหญ่จะต้องใช้วิศวกรรมคอมพิวเตอร์ เพื่อออกแบบระบบและโปรแกรมให้เข้ากับระบบการผลิตผสมผสานด้วยคอมพิวเตอร์ ที่ได้จัดตั้ง เป็นสิ่งที่ท้าทายมากเช่นกัน

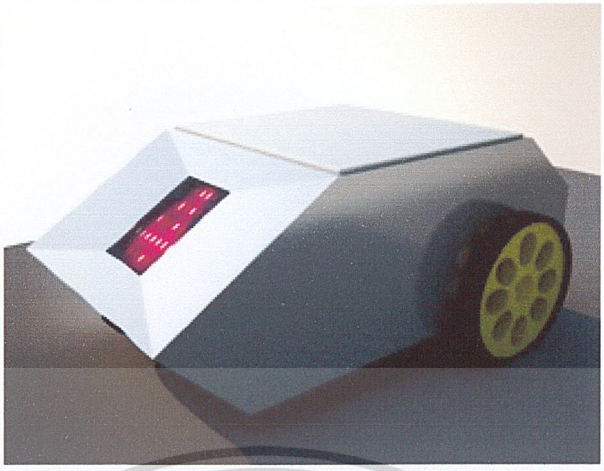
2.3 ส่วนประกอบภายในระบบจำลองกระบวนการการผลิต

การจำลองระบบโรงงานนี้ประกอบไปด้วย รถลำเลียงสินค้าที่ทำตามคำสั่งที่ส่งผ่านมาทางมอนิเตอร์ โดยรถลำเลียงสินค้าจะวิ่งบนรางไฟฟ้า ซึ่งทำหน้าที่ลำเลียงสินค้าจากหน่วยการทำงานหนึ่งไปยังอีกหน่วยการทำงานหนึ่ง และมีแขนกลทำการหยิบจับในการขนส่งสินค้าเข้า-ออกระหว่างรถลำเลียงกับหน่วยการทำงาน ซึ่งจะมีหุ่นยนต์แมลง 6 ขา มารับสินค้าจากแขนกลไปยังหุ่นยนต์ 2 ขา เพื่อไปเก็บในคลังสินค้าต่อไป

2.3.1 ระบบลำเลียง ประกอบไปด้วยรถลำเลียงสินค้าและสถานีรับ - ส่งสินค้า

2.3.1.1 รถลำเลียงสินค้า ทำหน้าที่รับส่งสินค้าจากสถานีหนึ่งไปยังสถานีหนึ่ง ซึ่งใช้หลักการของการส่งข้อมูลผ่านทางสัญญาณอินฟราเรด โดยรถจะเคลื่อนที่ไปตามรางไฟฟ้า และเมื่อตัวรถนั้นเคลื่อนที่มาจอดที่สถานี จากนั้นวงจรรับ-ส่งสัญญาณจะเริ่มทำงาน โดยตัวรถจะส่งสัญญาณไปยังสถานี ส่วนที่ทำหน้าที่รับสัญญาณที่ส่งไปก็คือ สถานี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 รถลำเลียงสินค้า

2.3.1.2 สถานีรับ-ส่งสินค้า ทำหน้าที่รับข้อมูลทางตัวควบคุมหลัก และตรวจเงื่อนไขของรถรับ-ส่งสินค้า ซึ่งใช้หลักการของการส่งข้อมูลผ่านทางสัญญาณอินฟราเรดเช่นเดียวกัน โดยจะเริ่มทำงานเมื่อรถเคลื่อนที่เข้ามาจอดที่สถานี และส่วนสถานีได้รับสัญญาณอินฟราเรด ก็จะทำส่งมายังตัวควบคุมหลัก ให้ทำการประมวลผลและตัดสินใจ

รูปที่ 2.2 สถานีรับ-ส่งสินค้า

2.3.2 แขนกล ทำหน้าที่เคลื่อนย้ายสินค้าระหว่างรถลำเลียงกับสถานีรับ-ส่งสินค้า ซึ่งแขนกลมีส่วนประกอบ 4 ส่วนหลักได้แก่

2.3.2.1 ส่วนควบคุม แขนกลจะถูกควบคุมการทำงานทั้งหมดจากบอร์ดสวิตซ์ควบคุม(Keypad Joystick)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2.2 ส่วนประมวลผล ทำหน้าที่ในการประมวลผลคำสั่งที่ได้จากส่วนควบคุมจากบอร์ดแสตมป์ 168 (STAMP 168) แล้วทำการประมวลผลค่าที่ได้มา จากนั้นทำการส่งข้อมูลที่ได้ไปยังเซอร์โวมอเตอร์ให้ทำงาน

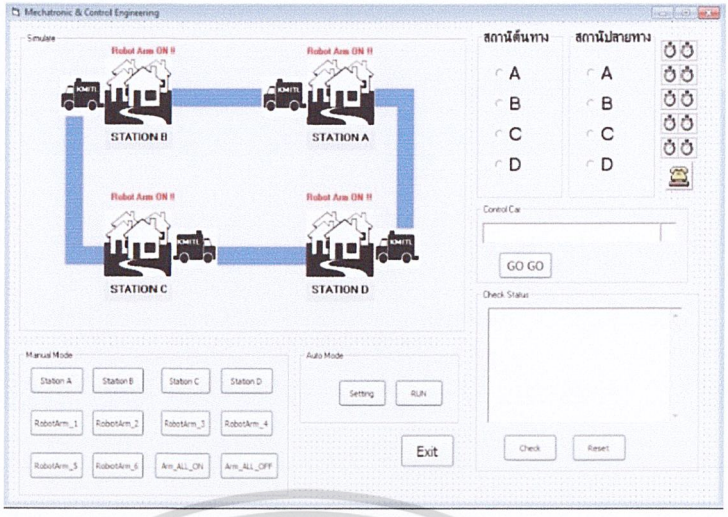
2.3.2.3 ส่วนเซอร์โวมอเตอร์ เป็นส่วนที่ขับเคลื่อนแขนกลให้เคลื่อนไหวไปยังตำแหน่งที่ต้องการ ซึ่งเซอร์โวมอเตอร์จะถูกติดตั้งไว้ในส่วนของข้อต่อต่างๆของแขนกลจำนวน 6 จุด

2.3.2.4 ส่วนที่ประกอบเป็นแขนกล จะแบ่งออกเป็น 3 ส่วน คือ คลิ๊ปเปอร์ แขน และฐาน ซึ่งชิ้นส่วนทุกชิ้นทำมาจากแผ่นอะคริลิคและได้ทำการออกแบบจากโปรแกรมโซลิดเวิร์คเวอร์ชั่น 2010 แล้วนำไปแปลงเป็นค่าจีโค๊ด ออกมาแล้วนำค่าจีโค๊ดที่ได้นำไปกัดชิ้นงานผ่านเครื่องซีเอ็นซีและนำชิ้นงานที่ได้นำไปประกอบเข้ากับเซอร์โวมอเตอร์

รูปที่ 2.3 หุ่นยนต์แขนกล

2.3.3 ส่วนแสดงผล ทำหน้าที่แสดงผลการทำงานของระบบ เพื่อให้เห็นภาพการทำงานของแขนกลและการเคลื่อนที่ของรถขนส่งสินค้าในแต่ละสถานี โดยการทำงานมีทั้งระบบอัตโนมัติและระบบควบคุมด้วยมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



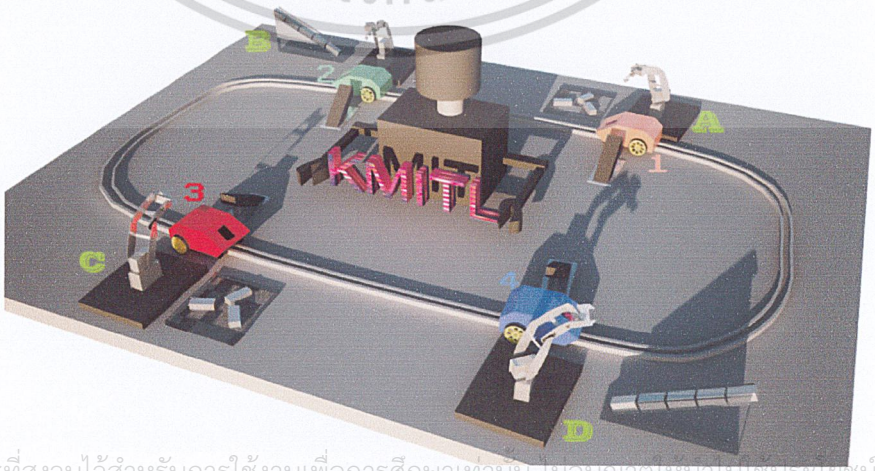
รูปที่ 2.4 ส่วนแสดงผล

2.4 หลักการทำงานของระบบ

จะมีด้วยกัน 2 โหมด คือ

2.4.1 โหมดควบคุมด้วยมือ

ในการทำงานของโหมดควบคุมด้วยมือ เราจะเป็นผู้สั่งการทำงานของระบบ ว่าเราต้องการให้รถวิ่งออกจากสถานีใด แล้วไปจอดที่สถานีใด โดยรถที่เหลืออีก สามคัน จะวิ่งไปจอดถัดไปอีกหนึ่งสถานี อย่างเช่น ถ้าเราต้องการให้รถคันที่หนึ่ง วิ่งไปยังสถานีที่สี่ ดังนั้น รถคันที่สองจะวิ่งไปจอดที่สถานีที่หนึ่ง รถคันที่สามจะวิ่งไปจอดที่สถานีที่สอง และรถคันที่สี่จะวิ่งไปจอดสถานีที่สาม จะเป็นอย่างนี้ไปเรื่อยๆ ไม่ว่าเราจะทำการเลือกรถคันที่เท่าไร สถานีต้นทางหรือปลายทางใด รถคันที่เหลือจะวิ่งไปจอดตามสถานีตามที่กล่าวดังข้างต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

รูปที่ 2.5 การทำงานของระบบการผลิตผสมผสานด้วยคอมพิวเตอร์

ถ้าเรามองจากรูปภาพข้างบน แล้วกำหนดให้รถคันที่หนึ่ง สอง สาม และ สี่ เป็น 1 , 2 , 3 และ 4 ส่วนสถานีที่หนึ่ง สอง สาม และ สี่ เป็น A , B , C และ D โดยเลือกให้รถคันที่ 1 วิ่งไปยังสถานีที่ D จะได้ 1D – 2A – 3B – 4C

2.4.2 โหมดอัตโนมัติ

เมื่อเราทำการเขียนโปรแกรมและทำการสั่งให้ระบบทำงาน ระบบจะทำงานตามที่เรากำหนด ซึ่งจะคล้ายกับManual Mode แต่จะต่างกันตรงที่ ถ้าให้รถคันที่ 1 วิ่งไปยังสถานีที่ B ดังนั้น รถคันที่ 2 จะวิ่งไปยังสถานีที่ C รถคันที่ 3 จะวิ่งไปยังสถานีที่ D และรถคันที่ 4 จะวิ่งไปยังสถานีที่ A และจะวิ่งอย่างนี้ไปเรื่อยๆ ตลอดการทำงาน (1B – 2C – 3D – 4A)

2.5 การติดต่อสื่อสารกันของระบบ

ในการติดต่อกันระหว่างรถกับสถานีใช้อินฟราเรดในการติดต่อสื่อสาร ตัวส่ง คือ หลอดอินฟราเรด และ ตัวรับ คือ โมดูลรับสำเร็จรูป(3ขา) จะส่งด้วยความถี่ 40 กิโลเฮิรตซ์ โดยประมาณ ประโยชน์เพื่อเป็นความถี่หลักในการตรวจรับว่าเป็นสัญญาณตัวจริงไม่ใช่สัญญาณรบกวน ตัวรับแบบโมดูล (3ขา) โมดูลจะรับสัญญาณที่กระพริบด้วยความถี่ประมาณ 40 กิโลเฮิรตซ์ ถ้าตรงก็จะให้เอาท์พุทที่ขาเอาท์พุทเป็น 0 หลักการของมันก็มีแค่ส่งแสงอินฟราเรดไปยังวัตถุที่ต้องการตรวจจับ ถ้าพบวัตถุนั้นก็จะสะท้อนแสงกลับมายังตัวรับ โดยจะติดตั้ง ตัวรับและส่งสัญญาณนี้ไว้ที่ รถ กับ สถานี อย่างละ1ชุด

ส่วนการติดต่อกันระหว่างตัวควบคุมหลัก กับ สถานี นั้นเราใช้สายแพในการรับส่งข้อมูล โดยส่วนตัวควบคุมหลัก จะทำหน้าที่เป็นตัวส่งข้อมูลไปยังสถานี เพื่อสั่งการว่าให้รถที่มาจอดเคลื่อนที่ไปสถานีใดต่อไป

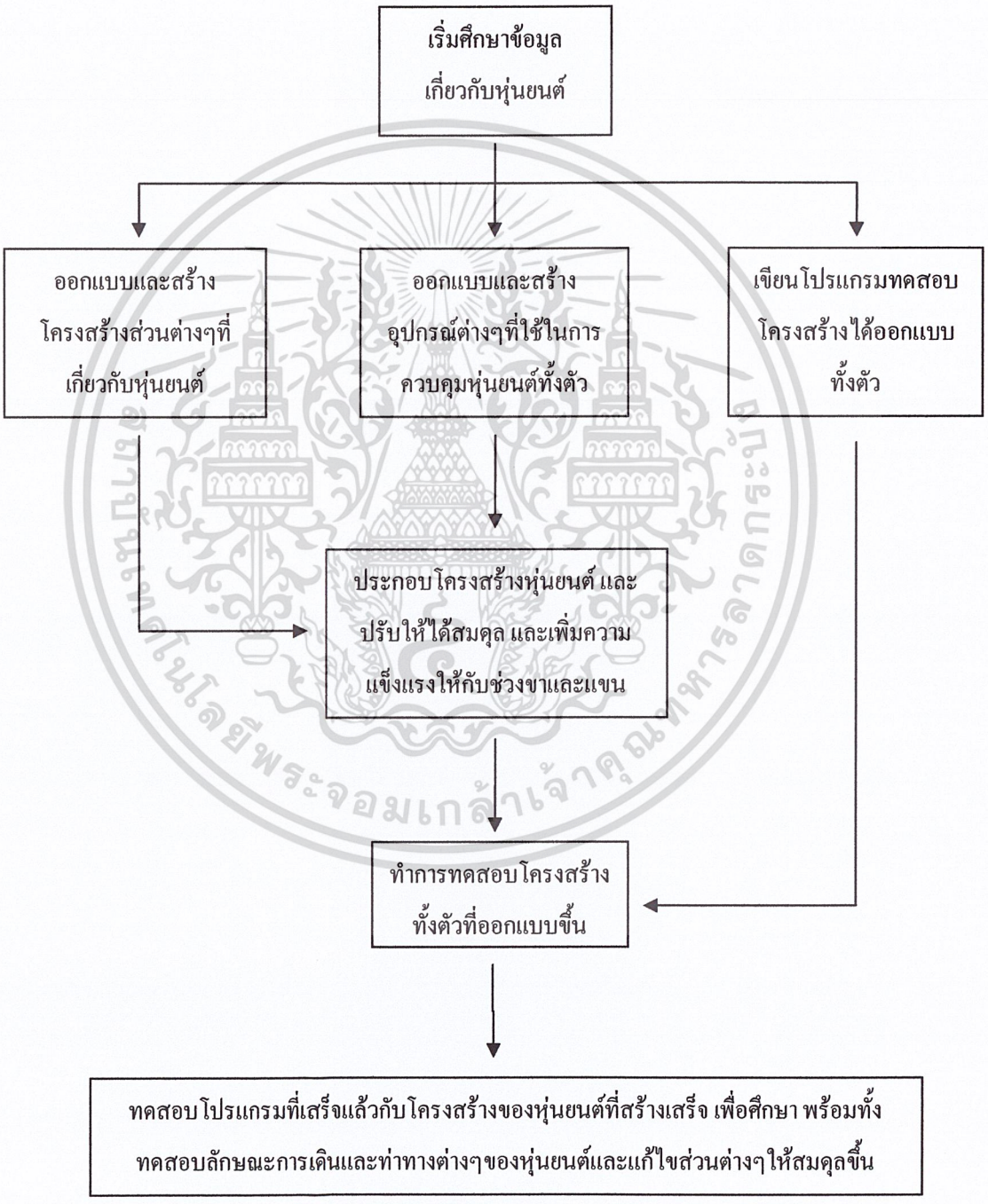
และการติดต่อกันระหว่างตัวควบคุมหลัก กับ คอมพิวเตอร์จะใช้สายยูเอสบี ในการเชื่อมต่อเพื่อที่จะส่งข้อมูลจากคอมพิวเตอร์ไปยังตัวควบคุมหลัก ซึ่งตัวควบคุมหลัก จะได้สั่งงานไปยังสถานี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

หุ่นยนต์เสมือนมนุษย์

3.1 ขั้นตอนการทำงาน



รูปที่ 3.1 แผนภาพแสดงขั้นตอนการทำงานของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 เครื่องมือและอุปกรณ์

3.2.1 ฮาร์ดแวร์

- อีซี่ เมก้า 1280 (Easy Mega1280) จำนวน 1 แผง
- เซอร์โว เอ็มจี945 (Servo MG945) จำนวน 14 ตัว
- เซอร์โว เอส3003 (Servo S3003) จำนวน 4 ตัว
- ดิจิตอล เซอร์โว อีเค2-0508 (Digital servo EK2-0508) จำนวน 5 ตัว
- อะคริลิค 2 มิลลิเมตร จำนวน 5 แผ่น
- อะคริลิค 3 มิลลิเมตร จำนวน 5 แผ่น

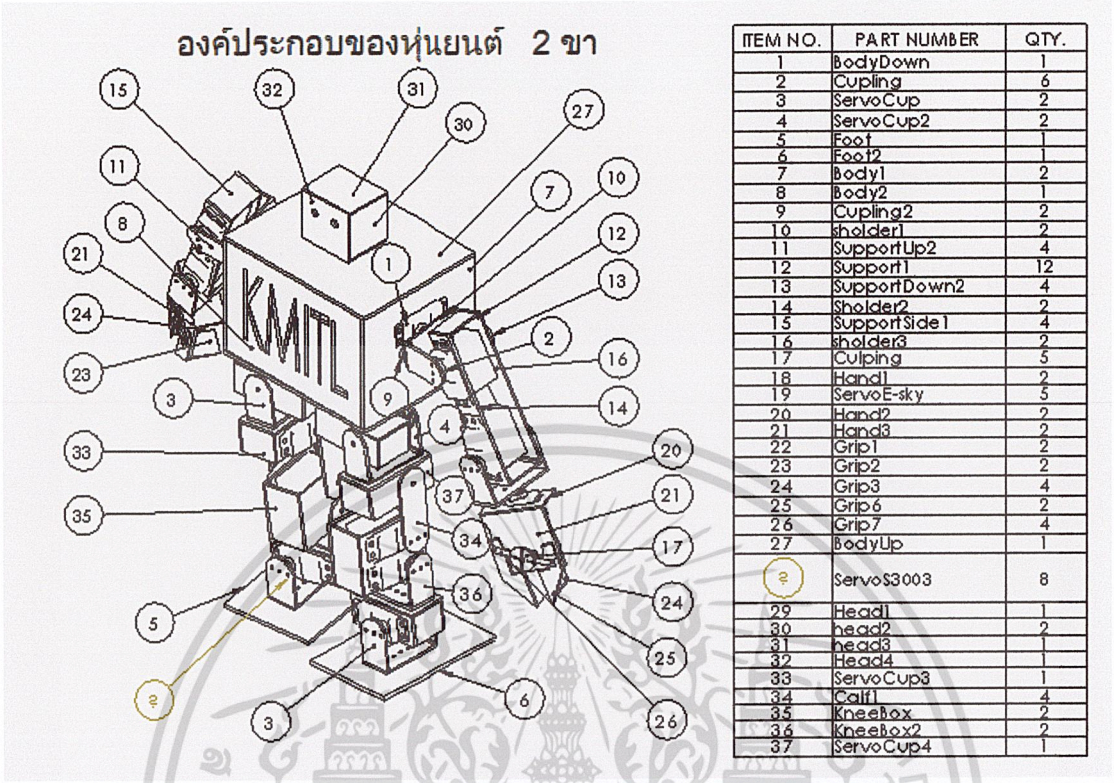
3.2.2 ซอฟต์แวร์

- โปรแกรมออดิโน เวอร์ชั่น 2008 (Program Arduino version 2008)
- โปรแกรมโซลิดเวิร์ค เวอร์ชั่น 2008 (Program SolidWorks version 2008)
- โปรแกรมโซลิดแคม เวอร์ชั่น 2008 (Program SolidCam version 2008)
- โปรแกรม Mach3

3.3 การออกแบบโครงสร้างหุ่นยนต์

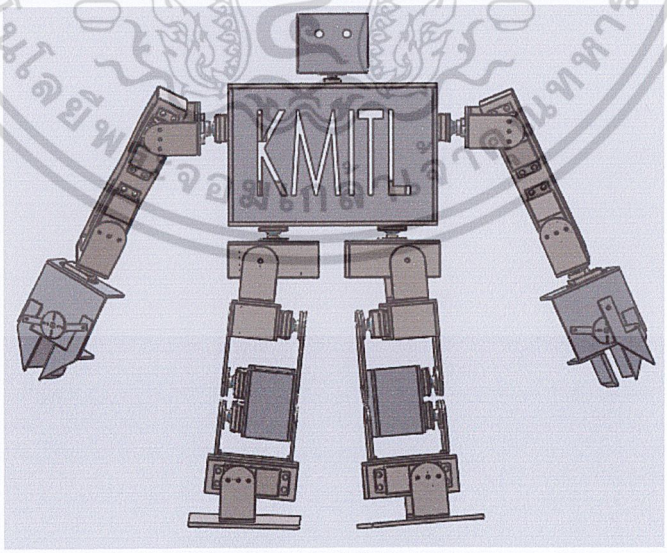
วัสดุที่ใช้คือ อะคริลิคเนื่องจากอะคริลิคเป็นวัสดุที่มีความเหนียว (toughness) ความโปร่งใส (transparent) สามารถขึ้นรูปได้ง่าย โดยมีคุณสมบัติที่น่าสนใจแล้วได้เลือกนำมาใช้ทำ คือ ทนต่อจุดเดือด จุดหลอมเหลวสูง สามารถทนต่อแรงกระแทก (impact strength) สูง แต่อาจเกิดรอยขูดขีดได้ง่ายและไม่ทนทานต่อตัวทำละลายหลายชนิด เนื่องจากมีเนื้อพลาสติกอ่อน รวมทั้งยังมีความทนทานต่อสภาพแวดล้อมดีอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ITEM NO.	PART NUMBER	QTY.
1	BodyDown	1
2	Cupling	6
3	ServoCup	2
4	ServoCup2	2
5	Foot	1
6	Foot2	1
7	Body1	2
8	Body2	1
9	Cupling2	2
10	sholder1	2
11	SupportUp2	4
12	Support1	12
13	SupportDown2	4
14	Sholder2	2
15	SupportSide1	4
16	sholder3	2
17	Culping	5
18	Hand1	2
19	ServoE-sky	5
20	Hand2	2
21	Hand3	2
22	Grip1	2
23	Grip2	2
24	Grip3	4
25	Grip6	2
26	Grip7	4
27	BodyUp	1
๒	ServoS3003	8
29	Head1	1
30	head2	2
31	head3	1
32	Head4	1
33	ServoCup3	1
34	Calf1	4
35	Kneebox	2
36	Kneebox2	2
37	ServoCup4	1

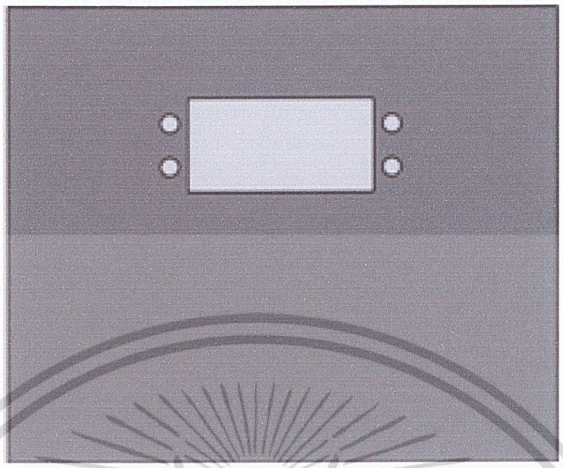
รูป 3.2 ส่วนประกอบของหุ่นยนต์สองขา



รูป 3.3 โครงสร้างหุ่นยนต์สองขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

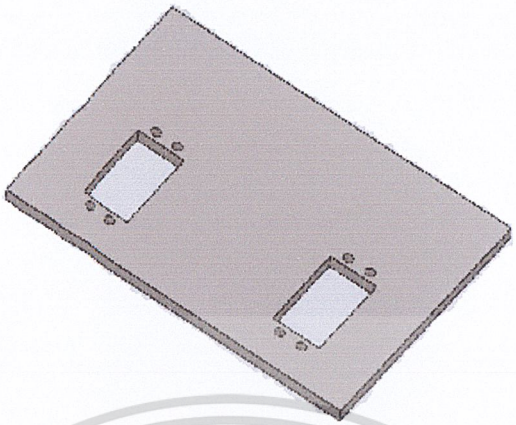
3.3.1 ตัวอย่างชิ้นส่วนหุ่นยนต์จากโปรแกรมโซลิดเวิร์ค



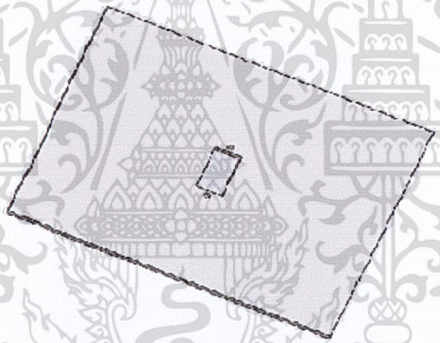
รูป 3.4 ชิ้นส่วนลำตัว 1

รูป 3.5 ชิ้นส่วนลำตัว 2

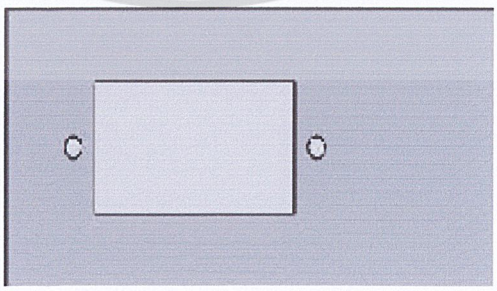
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.6 ชิ้นตัวลำตัวด้านล่าง

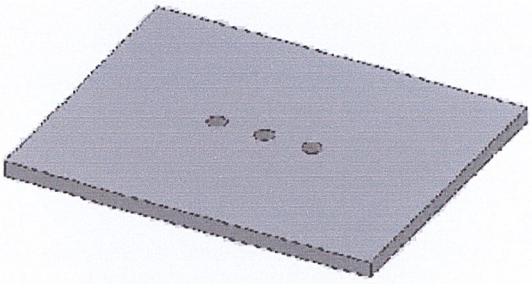


รูป 3.7 ชิ้นส่วนลำตัวด้านบน



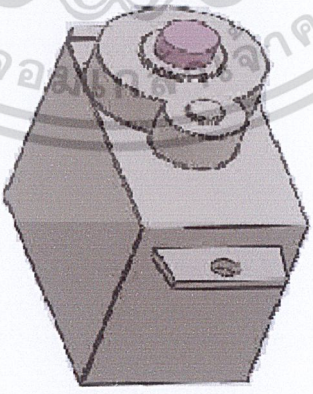
รูป 3.8 ชิ้นส่วนแขน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.9 ชิ้นส่วนหัว

รูป 3.10 ชิ้นส่วนข้อต่อของน่อง



รูป 3.11 ชิ้นส่วนของเซอร์โวมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 มอเตอร์กระแสตรง

ในหัวข้อนี้จะกล่าวถึงมอเตอร์กระแสตรง (DC motor) ในแง่ของทฤษฎีการทำงาน นอกจากนี้ก็จะกล่าวถึงพื้นฐานควบคุมการทำงานของมอเตอร์กระแสตรงทั้งการควบคุมทิศทางการหมุนและการควบคุมความเร็วในการหมุนมอเตอร์กระแสตรงเป็นเครื่องกลไฟฟ้าที่เปลี่ยนพลังงานไฟฟ้าเป็นพลังงานทางกล ผ่านทางแกนหมุนหรือเพลามอเตอร์ สามารถที่จะหมุนได้เนื่องจากจะต้องมีสนามแม่เหล็ก จาก 2 แหล่ง กระทำต่อกันโดยที่สนามแม่เหล็กทั้ง 2 แหล่งอาจจะเป็นแบบที่ได้จากการผ่านกระแสไฟฟ้าผ่านขดลวดสเตเตอร์ (Stator Winding)และลวดอาร์เมเจอร์ (Armature) แต่มอเตอร์กระแสตรงที่นิยมใช้จะเป็นแบบที่มีแม่เหล็กถาวร (Permanent Magnet) เป็นตัวสร้างสนามแม่เหล็กแทนขดลวดสเตเตอร์ (Stator) และใช้การผ่านกระแสไฟฟ้าเข้าไปที่ขดลวดอาร์เมเจอร์ เนื่องจากจะลดความสูญเสียจากการที่ไม่มี Field winding นั้นคือประสิทธิภาพที่ดีขึ้น นอกจากนี้ยังมีขนาดเล็ก ราคาถูก สนามแม่เหล็กที่เกิดจากแม่เหล็กถาวร (Permanent Magnet) และเกิดจากการผ่านไฟฟ้ากระแสตรงเข้าไปในขดลวดอาร์เมเจอร์ จะทำให้เกิดแรงบิด (Torque) เกิดขึ้นที่โรเตอร์ ซึ่งจะทำให้เกิดการหมุนได้นั่นเอง

3.4.1 การควบคุมทิศทางการหมุนของมอเตอร์กระแสตรง

สามารถทำได้โดยการควบคุมทิศทางการไหลกระแสไฟฟ้าที่ไหลผ่านขดลวดอาเมเจอร์ ซึ่งสามารถทำได้โดยการกลับขั้วไฟฟ้าของแหล่งจ่าย โดยที่การควบคุมการไหลของกระแสไฟฟ้านั้นส่วนใหญ่แล้วจะใช้วิธีการต่อวงจรอิเล็กทรอนิกส์ที่เรียกว่า เอช-บริจด์ (H-Bridge) เข้ากับมอเตอร์

3.4.2 การควบคุมความเร็วของมอเตอร์กระแสตรง

เนื่องจากส่วนขดลวดสเตเตอร์ ถูกแทนด้วยแม่เหล็กถาวร การควบคุมความเร็วของมอเตอร์จึงทำได้โดยการเปลี่ยนค่าแรงดันอาร์เมเจอร์ นั่นคือ เราสามารถที่จะควบคุมความเร็วของมอเตอร์ได้แต่จะไม่เกินความเร็วพื้นฐาน (ที่จำกัดด้วยสนามแม่เหล็กจากแม่เหล็กถาวรและข้อจำกัดเรื่องการทนกระแสได้ของขดลวดอาร์เมเจอร์ และข้อกำหนดของแหล่งจ่ายไฟฟ้า)

ความเร็วของมอเตอร์กระแสตรงนี้ จะขึ้นกับความต่างศักย์ของแหล่งจ่ายไฟฟ้าที่มอเตอร์ต่ออยู่ โดยที่ความเร็วในการหมุนของมอเตอร์กระแสตรงจะแปรผันโดยตรงกับค่าความต่างศักย์ นั่นคือ หากเราสามารถที่จะเปลี่ยนแปลงค่าความต่างศักย์ไฟฟ้าที่จ่ายให้มอเตอร์

สำหรับอีกวิธีหนึ่งที่นิยมให้มากคือ การเปลี่ยนระดับความต่างศักย์ไฟฟ้าเพื่อที่จะควบคุมความเร็วของมอเตอร์กระแสตรง จะใช้วิธีการจ่ายกระแสไฟฟ้าที่ได้จากแหล่งจ่ายไฟฟ้าที่ได้จากแหล่งที่เป็นคลื่นรูปสี่เหลี่ยมที่สามารถเปลี่ยนแปลงช่วงเวลาในการเปิดและปิดแหล่งจ่ายได้ ซึ่งเรียกว่า การมอดูเลตความกว้างพัลส์ (Pulse Width Modulation) ซึ่งก็คือการที่ให้แหล่งจ่ายไฟฟ้าเปิดและปิดสลับกันไปโดยหากแหล่งจ่ายเปิดมากกว่าปิด ก็จะทำให้ค่าเฉลี่ยของความต่างศักย์ที่มาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั่นคือ มอเตอร์จะหมุนเร็วแต่หากมีการปิดมากกว่าเปิด ก็จะทำให้ค่าเฉลี่ยของความต่างศักย์ที่น้อยกว่าและมอเตอร์ก็จะหมุนช้าลงนั่นเอง

3.5 เซอร์โวมอเตอร์

ในการทำโครงงานครั้งนี้ ตัวมอเตอร์ที่ใช้ขับเคลื่อนข้อต่อต่างๆ นั้นที่จะต้องมีทอร์คที่มากพอ และสามารถความคุมองศาของมอเตอร์ได้โดยง่ายเนื่องจากมีจำนวนข้อต่อเป็นจำนวนมาก จึงนำ 5 เซอร์โวมอเตอร์เข้ามาใช้งานในเหตุผลดังกล่าว

3.5.1 ความหมายของเซอร์โวมอเตอร์

เซอร์โวมอเตอร์คือมอเตอร์ที่นำมาใช้ในระบบการควบคุมทางความเร็วและตำแหน่งร่วมกันโดยระบบเซอร์โวมอเตอร์ที่ดีจะต้องตอบสนองต่อความเร็วและการเข้าถึงตำแหน่งหรือการเคลื่อนที่ไปที่ระยะเป้าหมายอย่างเหมาะสม ระบบเซอร์โวมอเตอร์จะมีการป้อนกลับของความเร็วและตำแหน่งกลับมาที่ส่วนควบคุมหรือการขับ แล้วแต่การออกแบบ การป้อนกลับความเร็วอาจใช้เทอร์โมมิเตอร์ (Tachometer) และการป้อนกลับตำแหน่งจะใช้โพเทนชิโอมิเตอร์(Potentiometer), เอ็นโคเดอร์ (Encoder) หรือ ตัวกำเนิดสัญญาณพัลส์ (Pulse Generator) หรือบางครั้งจะใช้ รีโซเวอร์ (Resover) ก็ได้ ที่มีหลักการเหมือน เอซี เจนเนอเรเตอร์ เพราะสัญญาณออกเป็นไซน์

3.5.2 ส่วนประกอบภายในรีโหมด คอนโทรล (remote control)

เซอร์โวมอเตอร์เป็นอุปกรณ์สำหรับขับเคลื่อนหุ่นยนต์หรืองานที่จำเป็นต้องกำหนดการหมุนเป็นช่วงๆ หรือตามองศาที่ต้องการ โดยกำหนดการหมุนในลักษณะครึ่งวงกลมโดยจะรับสัญญาณพัลส์ เข้ามาเป็นตัวบอกให้มอเตอร์ปรับตำแหน่ง ภายในตัวเซอร์โวมอเตอร์ประกอบด้วยแผงควบคุมซึ่งทำหน้าที่รับสัญญาณจากไมโครคอนโทรลเลอร์มาควบคุมการหมุนหรือเคลื่อนที่ไปยังตำแหน่งที่ต้องการ ชุดฟันเฟืองที่ติดตั้งไว้ภายในประกอบด้วยเฟืองพลาสติกซึ่งทำหน้าที่เพิ่มกำลังหรือแรงบิดให้กับตัวเซอร์โวมอเตอร์ การรับสัญญาณพัลส์จากไมโครคอนโทรลเลอร์จะรับสัญญาณเพียงหนึ่ง อินพุท/เอาท์พุท เท่านั้นจึงประหยัดขา อินพุท/เอาท์พุท ได้มากกว่าการใช้สเตปปิ้งมอเตอร์ (Stepping motor) ติดตั้งง่ายเพราะมีลักษณะเป็นสี่เหลี่ยมและมีขนาด 1.6 x 0.8 x 1.4 นิ้ว น้ำหนัก 49 กรัม ใช้ไฟได้ตั้งแต่ 4.8 - 6 โวลต์ ไฟฟ้ากระตรงกินกระแส 7.2 มิลลิแอมแปร์ เมื่อไม่มีโหลด

3.5.3 หลักการทำงานของเซอร์โวมอเตอร์

เซอร์โวมอเตอร์เป็นมอเตอร์ไฟฟ้ากระแสตรงดีซี มอเตอร์ประกอบด้วยชุดเกียร์ และส่วนควบคุมต่างๆ ไว้ในโมดูลเดียวกัน มีสายต่อใช้งานเพียง 3 เส้นเท่านั้น คือ VCC, GND และสายสัญญาณควบคุม (Control Line) ซึ่งสามารถควบคุมให้เซอร์โวมอเตอร์หมุนซ้าย หรือ ขวาได้ +90 องศา - 90 องศา (180 องศา) โดยสัญญาณที่ใช้ควบคุมนี้จะเป็นสัญญาณ การมอดูเลตความกว้าง

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

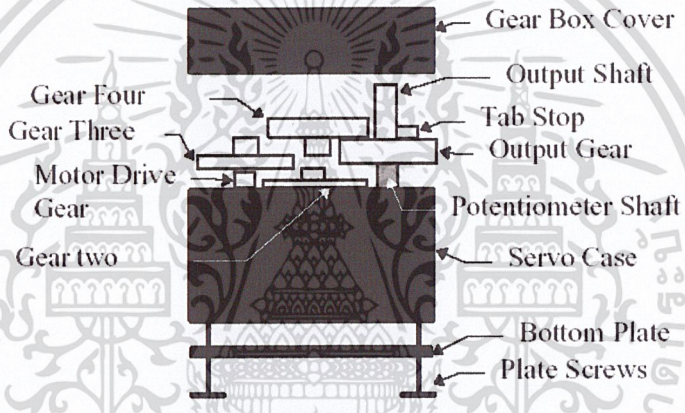
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

พัลส์ แบบ TTL Level ระดับแรงดันที่จ่ายให้มอเตอร์นี้จะอยู่ในช่วงประมาณ 4 ถึง 6 โวลต์ โดยสามารถสั่งงานในการหมุนให้หมุนไปได้ตามองศาต่างๆ ที่ต้องการ ได้ด้วยตัวของเซอร์โวมอเตอร์เอง ไม่ต้องมีส่วนควบคุม หรือเซนเซอร์ใดๆ กลับมาตรวจสอบอีกทำให้ง่าย และสะดวกในการนำไปประยุกต์ใช้งานต่างๆ

1. การควบคุมการทำงานของเซอร์โวมอเตอร์ทำได้โดยการป้อนสัญญาณความกว้างของพัลส์ให้กับตัวเซอร์โวมอเตอร์ซึ่งจะได้ทิศทางการหมุนและตำแหน่งของการหมุน

2. สามารถใช้งานกับไฟดีซี (DC) ได้ 4 - 6 โวลต์, หมุนได้ 180 องศา และสามารถปรับแต่งตัวเซอร์โวมอเตอร์ให้สามารถหมุนได้รอบตัวได้

3. ขั้วต่อจะเป็นแบบมาตรฐาน : ขั้ว JR TYPE

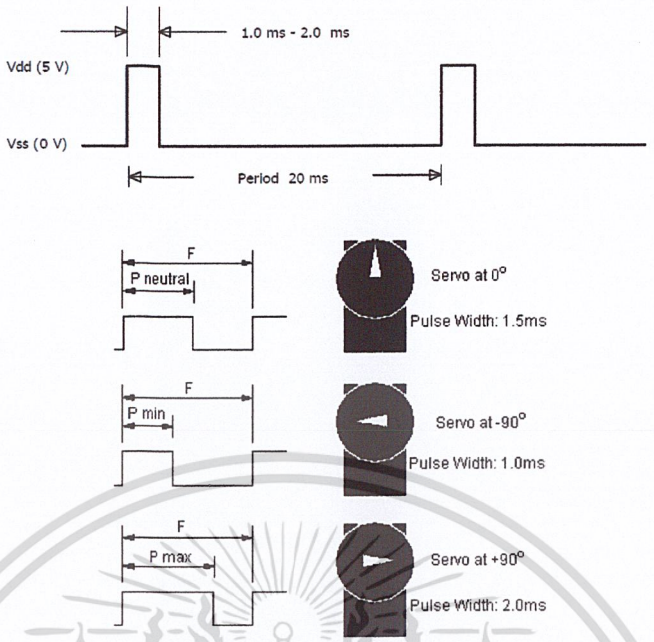


รูป 3.12 โครงสร้างของเซอร์โวมอเตอร์

การควบคุมการทำงานทำได้โดย การป้อนสัญญาณความกว้างพัลส์ ให้กับมอเตอร์ซึ่งตำแหน่งและทิศทางการหมุนของมอเตอร์นี้จะขึ้นอยู่กับขนาดของความกว้างของพัลส์นั้นๆ โดยทั่วไปแล้วความกว้างของสัญญาณพัลส์จะมีจุดให้อ้างอิง 3 จุด ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

119419



รูป 3.13 สัญญาณพัลส์กับเซอร์โวมอเตอร์

1. สัญญาณความกว้างพัลส์ขนาด 1.5 มิลลิวินาที จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม 0 องศา หรือ จุดกึ่งกลางของมอเตอร์

2. สัญญาณความกว้างพัลส์ขนาด 1 มิลลิวินาที จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม –90 หรือในทิศทางทวนเข็มนาฬิกา

3. สัญญาณความกว้างพัลส์ขนาด 2 มิลลิวินาที จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม +90 องศา หรือในทิศทางตามเข็มนาฬิกา

ส่วนการที่จะควบคุมให้มอเตอร์หมุนเป็นมุมอื่นๆ นั้นก็สามารถทำได้โดยการป้อนสัญญาณพัลส์เป็นระดับความกว้างต่างๆ โดยอ้างอิงจากจุด ทั้ง 3 จุดที่กล่าวมานี้ ซึ่งสัญญาณพัลส์นี้จะต้องจ่ายให้มอเตอร์ทุกๆ 20 มิลลิวินาที เพื่อรักษาสภาพตำแหน่งของมอเตอร์ไว้ก็คือ จะอาศัยการเปรียบเทียบช่วงเวลาของความกว้างพัลส์ที่จ่ายให้กับมอเตอร์ทางขาสัญญาณควบคุมกับค่าเวลาของวงจรอาร์ซี (RC) ภายในบอร์ดควบคุมในตัวของมอเตอร์ ซึ่งค่าเวลาของวงจรอาร์ซี (RC) นี้จะมีการเปลี่ยนแปลงตามการหมุนของมอเตอร์ เนื่องจากตัวต้านทานปรับค่าจะถูกยึดติดอยู่กับแกนหมุนของมอเตอร์ ซึ่งการหมุนของมอเตอร์จะทำให้ค่าความต้านทานของตัวต้านทานปรับค่าวีอาร์(VR) เปลี่ยนแปลงไป เป็นผลทำให้ค่าเวลาของวงจรอาร์ซี (RC) เปลี่ยนแปลงตามไปด้วย โดยในขณะที่เราป้อนสัญญาณความกว้างพัลส์ให้กับมอเตอร์ทางขาสัญญาณควบคุม สัญญาณนี้จะถูกนำไปเปรียบเทียบกับค่าเวลาของวงจรอาร์ซี (RC) หากค่าทั้ง 2 ไม่เท่ากันมอเตอร์ก็จะหมุนทำให้ค่าเวลาของวงจรอาร์ซี (RC) เปลี่ยนแปลงจนกระทั่งค่าเวลาความกว้างพัลส์ของ วงจรอาร์ซี (RC) เปลี่ยนแปลงจนเท่ากับสัญญาณพัลส์ทางขาควบคุม (Control line) มอเตอร์จึงจะหยุดหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

3.6 ส่วนประกอบของไมโครคอนโทลเลอร์

3.6.1 หน่วยประมวลผล (Central Processor Unit) เป็นส่วนที่ตัดสินใจเกี่ยวกับการทำงานต่างๆ ซึ่งจะทำงานตามโปรแกรมที่เราเขียนและอัดเข้าไปในตัวไมโครคอนโทรลเลอร์

3.6.2. หน่วยความจำ(Memory) เป็นตัวที่จะเก็บข้อมูลต่างๆที่ต้องใช้ในไมโครคอนโทรลเลอร์ ไม่ว่าจะเป็นหน่วยความจำ โปรแกรมหรือหน่วยความจำข้อมูล โดยหน่วยความจำที่ใช้ได้แก่ รอม(Rom), อีพรอม(EPROM), อีอีพรอม(EEPROM), แรม(RAM) รวมทั้งหน่วยความจำแฟลช (FLASH)

3.6.3. พอร์ทสัญญาณเข้าและสัญญาณออก (Input & Output) เป็นส่วนที่จะใช้ในการติดต่อกับอุปกรณ์ภายนอก

3.6.4. คุณสมบัติอื่นๆ ไมโครคอนโทรลเลอร์สมัยใหม่ จะมีฟังก์ชั่นพิเศษเพิ่มเติม เช่น ไทม์ (Time) / เคาท์เตอร์(Counter) /อะนาล็อก ทู ดิจิตอล (Analog to Digital Converter) / อะนาล็อก คอมพาเรเตอร์ (Analog Comparator) / ยูอาร์ท (UART) / ยูสาร์ท (USART)

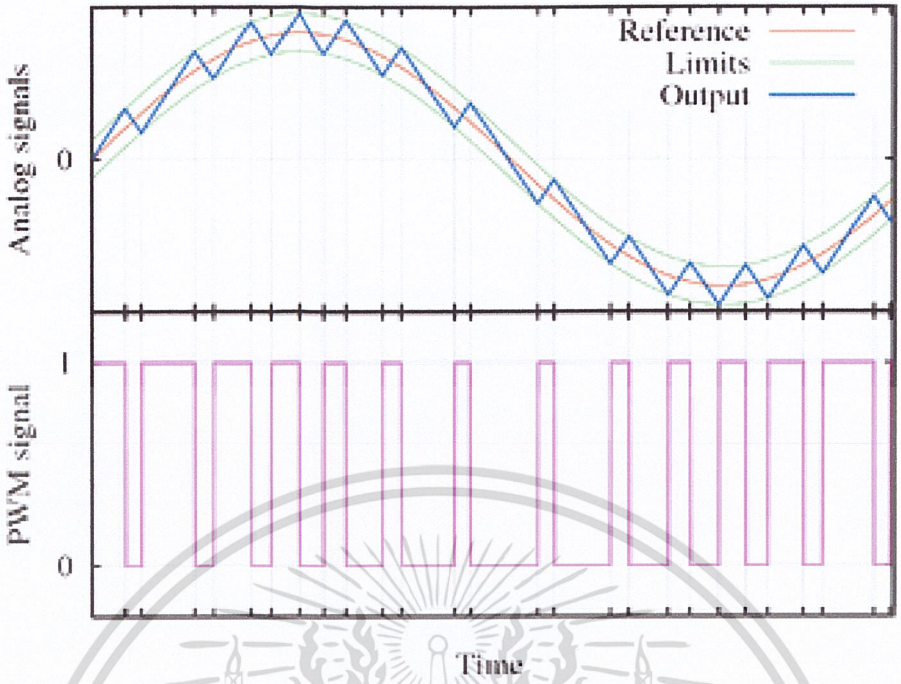
3.6.5. เอวีอาร์ ไมโครคอนโทรลเลอร์ (AVR Microcontroller)

3.7 หลักการของการมอดูเลตความกว้างพัลส์ (Pulse Width Modulation (PWM))

การมอดูเลตความกว้างพัลส์ เป็นเหมือนเครื่องมือชนิดหนึ่งที่สามารถนำมาประยุกต์ใช้งานในการนำเอาสัญญาณที่ใช้ไปควบคุมอุปกรณ์เพื่อให้ได้เอาท์พุตที่ต้องการและยังสามารถใช้งานทางด้านอิเล็กทรอนิกส์กำลัง งานด้านสื่อสารและงานทางด้านเครื่องมือวัด เป็นต้น

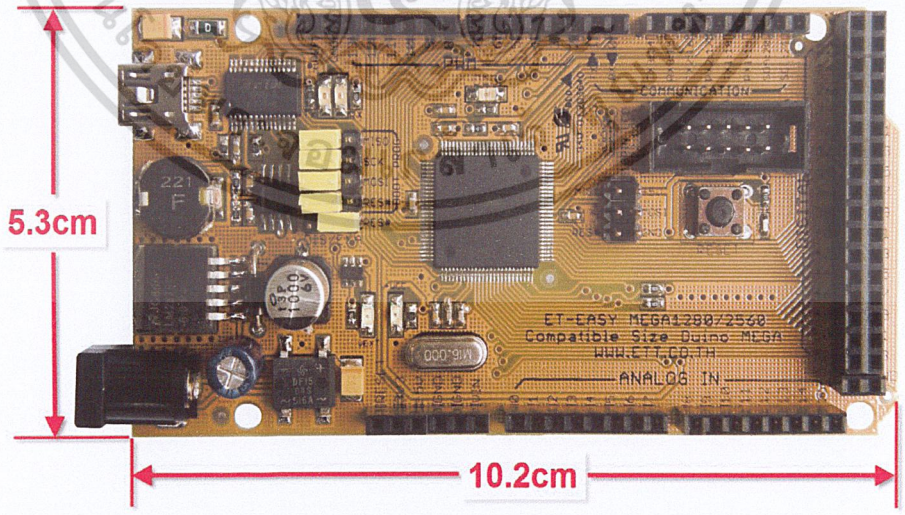
การมอดูเลตความกว้างพัลส์ ก็คือสัญญาณแบบพัลส์ ที่สามารถปรับ ช่วง (Duration) หรือตำแหน่ง (Position) ของสัญญาณพัลส์ได้โดยเราสามารถนำเอาสัญญาณข้างต้นมาใช้งานควบคุมการทำงานของมอเตอร์เพื่อให้สามารถปรับความเร็วของมอเตอร์ ให้ช้าหรือเร็วได้ โดยการเพิ่มรูปคลื่นทางซีกบวกให้มากกว่าด้านลบมอเตอร์จะหมุนเร็วขึ้น ถ้าให้ด้านลบมากกว่าด้านบวกมอเตอร์จะหมุนช้าลง ซึ่งในโครงงานนี้ต้องการความเร็วของมอเตอร์ที่ช้าเพราะจะสามารถควบคุมการเคลื่อนที่ของหุ่นยนต์ได้ง่ายและมีค่าความผิดพลาดที่น้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



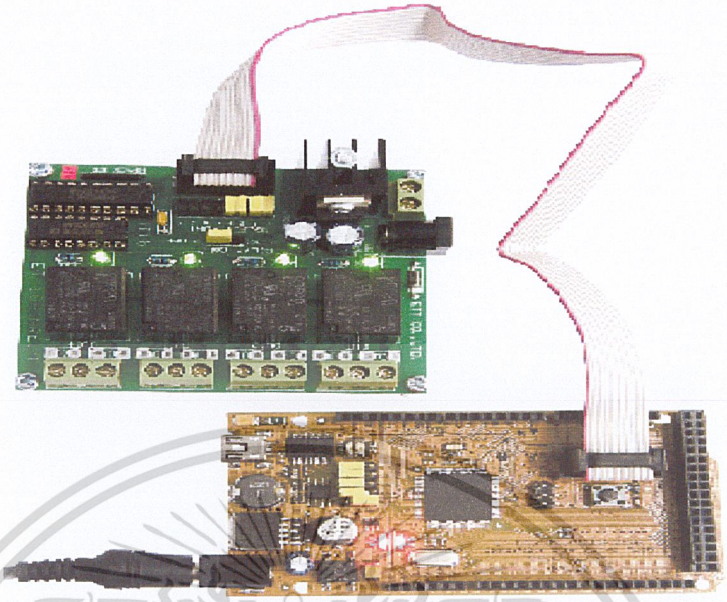
รูป 3.14 การมอดูเลตความกว้างพัลส์ (Pulse Width Modulation, PWM)

3.8 วงจรอิเล็กทรอนิกส์สำเร็จรูป



รูป 3.15 ตัวอย่างการติดกันบอร์ดอินพุท/เอาท์พุท ของอีทีที ด้วยขั้ว 10 ขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



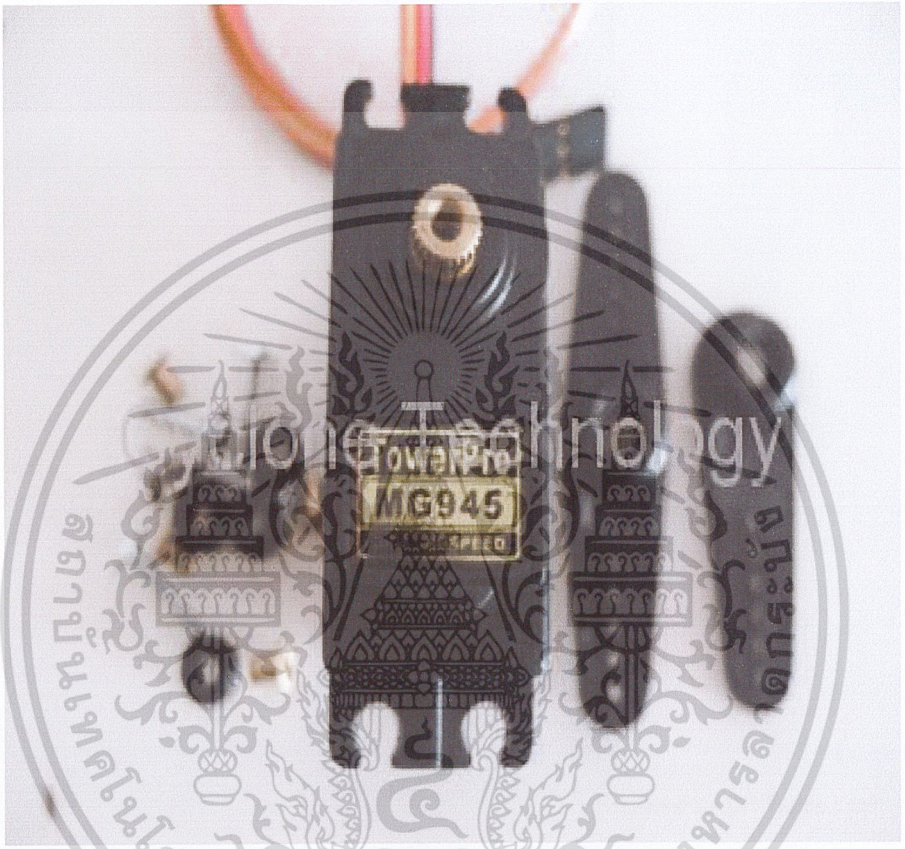
รูป 3.16 ตัวอย่างการติดกันบอร์ดอินพุท/เอาท์พุท ของอีทีที ด้วยขั้ว 10 ขา

มีการปรับปรุงโปรแกรมให้ใช้กับชิพเอวีอาร์(AVR) รุ่นใหญ่ขึ้น เพื่อให้มีจำนวนอินพุท/เอาท์พุท ทั้งดิจิตอล (Digital), อะนาล็อก (Analog), พีดับเบิ้ลยูเอ็ม (PWM), ยูอาร์ท (UART) และ ขนาดหน่วยความจำเพิ่มมากขึ้นกว่าเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9 เซอร์โวมอเตอร์ที่ใช้ในโครงงาน

เซอร์โวมอเตอร์ที่ใช้ในโครงงานนี้คือ MG945 S3003 และ Digital servo E-sky EK2-0508 โดยเซอร์โวมอเตอร์ แต่ละตัวมีรายละเอียดดังนี้



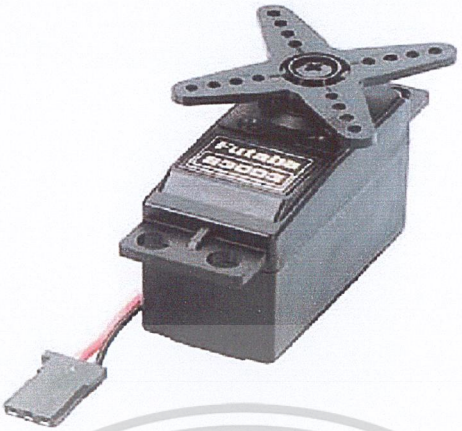
รูป 3.17 เซอร์โวมอเตอร์ รุ่น MG945

-เซอร์โวมอเตอร์ เวอร์ชั่น MG945

Detailed Specifications

Operation speed:	0.25Sec/60degrees
Stall Torque:	12KG/CM
Voltage Range:	4.8v – 7.2v
Dimension:	40.7mm x 19.7mm x 42.9mm
Weight:	55g

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.18 เซอร์โวมอเตอร์ เวอร์ชั่น S3003

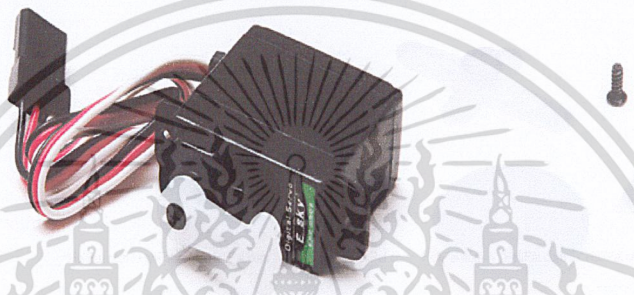
-เซอร์โวมอเตอร์ เวอร์ชั่น S3003

Detailed Specifications

Control System:	+Pulse Width Control 1520usec Neutral
Required Pulse:	3-5 Volt Peak to Peak Square Wave
Operating Voltage:	4.8 – 6.0 Volts
Operating Temperature Range:	-20 to +60 Degree C
Operating Speed (4.8V):	0.23 sec/60 degrees at no load
Operating Speed (6.0V):	0.19 sec/60 degrees at no load
Stall l Torque (4.8V):	44 oz/in. (3.2 kg.cm)
Stall Torque (6.0V):	56.8 oz/in. (4.1 kg.cm)
Operating Angle:	45 Deg. One side pulse traveling 400 usec360
Modifiable:	Yes
Current Drain (4.8V):	7.2mA/idle
Current Drain (6.0V):	8mA/idle
Direction:	Counter Clockwise Pulse Traveling 1520-1900 usec

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Motor Type:	3 Pole Ferrite
Potentiometer Drive:	Indirect Drive
Bearing Type:	Plastic Bearing
Gear Type:	All Nylon Gears
Connector Wire Length:	12”
Dimensions:	1.6” x 0.8” x 1.4” (41 x 20 x 36mm)



รูป 3.19 Digital servo motor EK 2-0508

-เซอร์โวมอเตอร์ เวอร์ชั่น EK 2-0508

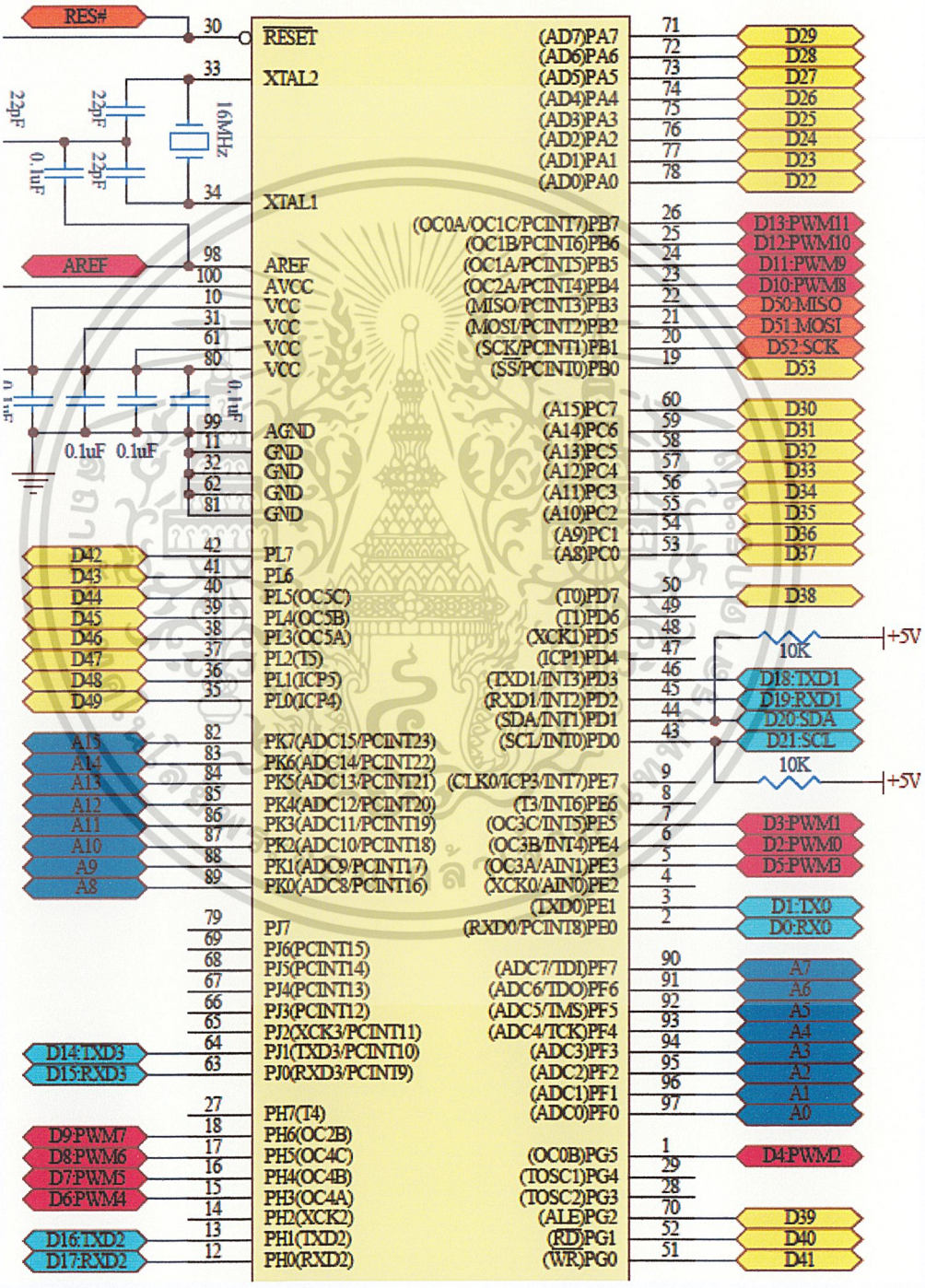
Detailed Specifications

Standard Voltage:	DC5V +/- 1V
PWM Input Range:	Pulse Cycle 20+-2ms, Positive Pulse 1~2ms
Speed:	60 degrees / 0.1s
Torque:	> 1KG/cm (Vcc=5V)
Max Corner:	> 150 degrees
Operation Temperature:	-20 degrees ~ 70 degrees
Weight:	7.5g
Dimensions:	22.8 * 11.5 * 20.8mm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.10 วงจรของหุ่นยนต์เสมือนมนุษย์

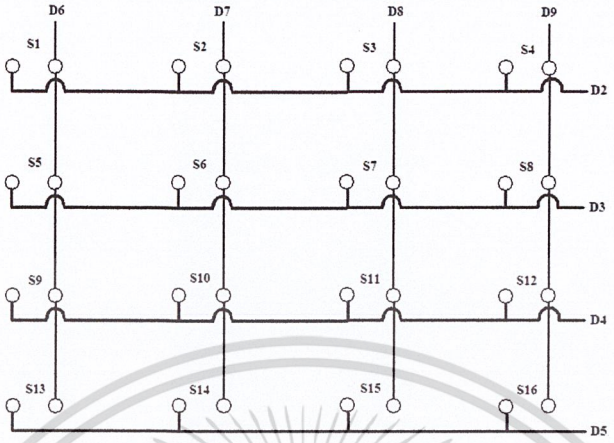
ในการควบคุมหุ่นยนต์สองขาได้ใช้ ไมโครคอลโทรลเลอร์ เอวีอาร์ เมก้า 1280 (Microcontroller AVR MEGA 1280) ในการควบคุมการทำงาน



รูป 3.20 วงจรของหุ่นยนต์เสมือนมนุษย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.10.1 วงจรบอร์ดสวิตซ์ควบคุม (Keypad Joystick)

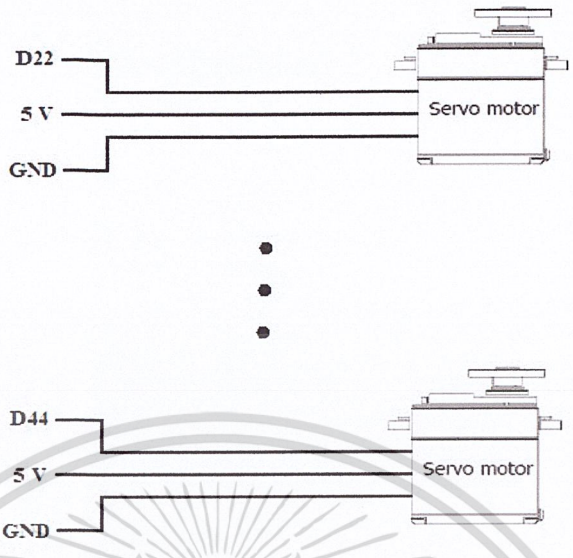


รูปที่ 3.21 วงจรบอร์ดสวิตซ์ควบคุม (Keypad Joystick)

ในการควบคุมหุ่นยนต์สองขาในเบื้องต้น ในการจัดท่าหุ่นยนต์ ได้ทำการประยุกต์ใช้วงจรควบคุมการจัดท่าของหุ่นยนต์แขนกลมาใช้ในการจัดท่าการเดินของหุ่นยนต์ 2 ขา โดยใช้ดิจิตอลพิน ดี2-ดี9 (Digital Pin D2-D9) ซึ่งแต่ละขา จะส่งสัญญาณโลจิก1 (Logic 1) ออกมา เช่น สวิตช์เอส1 (S1) จะมี ขา ดี2-ดี6 (Pin D2 กับ D6) เชื่อมกันอยู่ เมื่อกดปุ่มเอส1 (S1) ลอจิก (Logic) จากดี2 - ดี9 (D2 - D9) จะเชื่อมถึงกัน ทำให้ไมโครคอนโทรลเลอร์ รู้ว่ามีการกดปุ่มเอส1 (S1) ตามที่เขียนโปรแกรมไว้

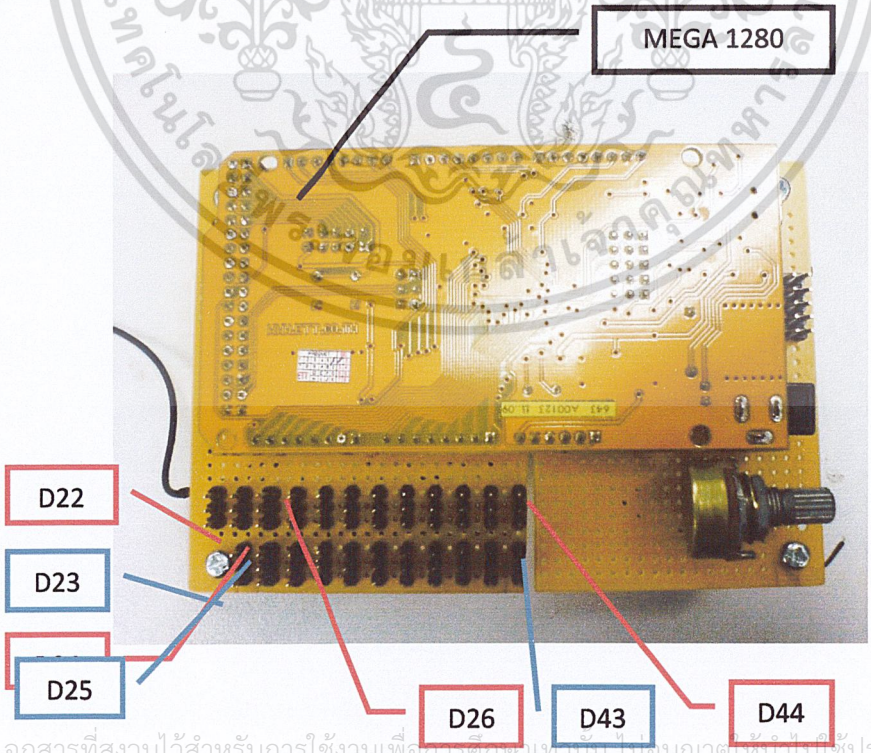
รูป 3.22 บอร์ดสวิตซ์ควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



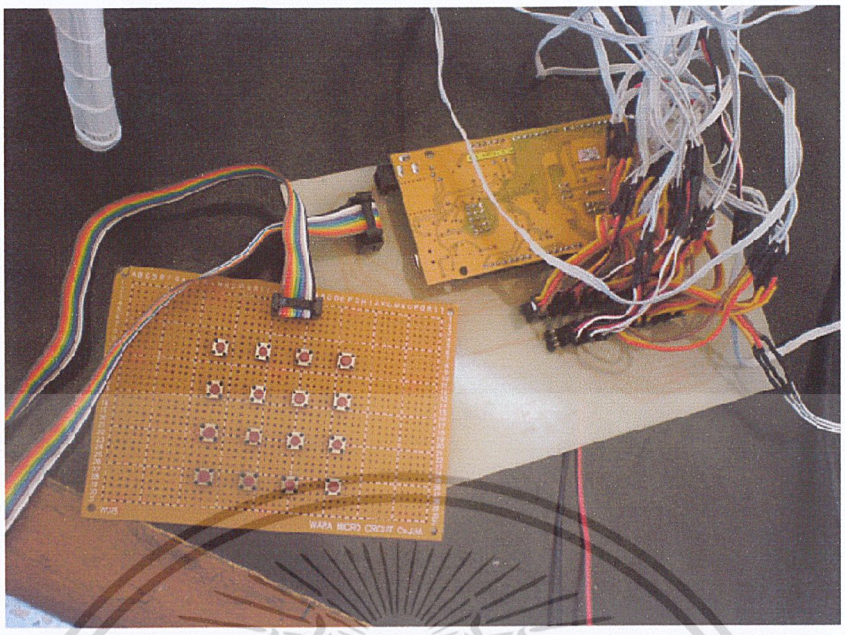
รูปที่ 3.23 การเชื่อมต่อเซอร์โวมอเตอร์ กับขา ของเอวีอาร์ เมก้า 1280

หุ่นยนต์สองขา ประกอบด้วยเซอร์มอเตอร์ จำนวน 23 ตัว ควบคุมเซอร์โวมอเตอร์ โดยเชื่อมต่อสายสัญญาณกับขา ดี22-ดี44 (Pin D22 – D44) ที่กำหนดให้ปล่อยค่าดิวตี้ ไซเคิล พัลส์ (Duty Cycle Pulse) ตามที่เขียนโปรแกรมไว้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

รูป 3.24 การเชื่อมต่อเซอร์โวมอเตอร์กับขาของเอวีอาร์ เมก้า 1280



รูป 3.25 บอร์ดสวิตซ์ควบคุม ต่อกับ วงจรเอวีอาร์

3.11 หลักการทำงาน

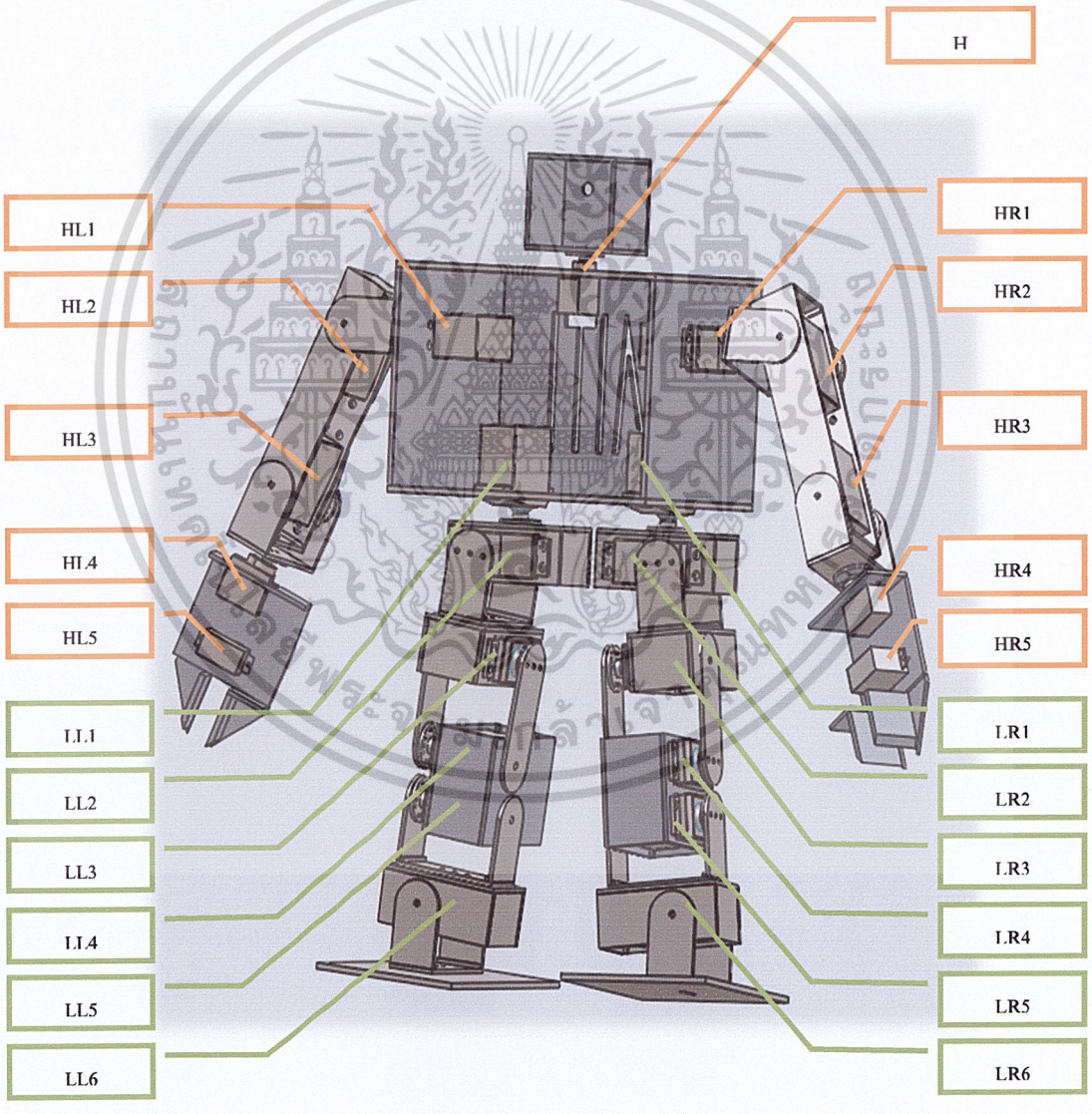
ในการควบคุมหุ่นยนต์สองขาได้ใช้เอวีอาร์ เมก้า 1280 (AVR MEGA 1280) ในการควบคุมการทำงาน และได้ทำการประยุกต์ใช้วงจรควบคุมการจัดท่าของหุ่นยนต์แขนกลมาใช้ในการจัดท่าการเดินของหุ่นยนต์สองขา โดยใช้ดิจิตอล พิน ดี2-ดี9 (Digital Pin D2-D9) ซึ่งแต่ละขา จะส่งสัญญาณลอจิก1 (Logic 1) ออกมา เช่น สวิตช์เอส1 (S1) จะมี ขาดี2 - ดี6 (Pin D2 - D6) เชื่อมกันอยู่ เมื่อกดปุ่มเอส1 (S1) ลอจิก(Logic) จาก ดี2 - ดี9 (D2 - D9) จะเชื่อมถึงกันทำให้ไมโครคอนโทรลเลอร์รู้ว่ามีการกดปุ่มเอส1 (S1) ตามที่เขียนโปรแกรมไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 การออกแบบท่าต่างๆให้หุ่นยนต์

การออกแบบท่าทางของหุ่น ทำได้โดยเขียนโปรแกรมสำหรับหาค่ามุมของเซอร์โวมอเตอร์แต่ละตัว แล้วแสดงค่ามุมนั้น บนหน้าจอคอมพิวเตอร์ แล้วจดบันทึกค่ามุมทุกตัวในท่านั้นๆ กำหนดตำแหน่งเซอร์โวมอเตอร์ ตามรูปภาพนี้



รูปที่ 4.1 ตำแหน่งเซอร์โวมอเตอร์ แต่ละตัวของหุ่นยนต์เสมือนมนุษย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ท่าและการเคลื่อนไหวของหุ่นเสมือนมนุษย์

4.2.1 ท่ายืนตรง

ตารางที่ 4.1 มุมองศาของเซอร์โวมอเตอร์ แต่ละตัวในท่ายืนตรง

STEP/SV	LL1	LL2	LL3	LL4	LL5	LL6	LR1	LR2	LR3	LR4	LR5	LR6
ยืน	92	110	90	102	58	54	106	122	56	84	96	96

4.2.2 ท่ายืนย่อเข่า

ตารางที่ 4.2 มุมองศาของเซอร์โวมอเตอร์ แต่ละตัวในท่ายืนย่อเข่า

STEP/SV	LL1	LL2	LL3	LL4	LL5	LL6	LR1	LR2	LR3	LR4	LR5	LR6
ย่อเข่า	92	114	94	86	42	60	106	124	50	106	116	98

4.2.3 ท่ายืนโค้งคำนับ

ตารางที่ 4.3 มุมองศาของเซอร์โวมอเตอร์ แต่ละตัวในท่ายืนโค้งคำนับ

STEP/SV	LL1	LL2	LL3	LL4	LL5	LL6	LR1	LR2	LR3	LR4	LR5	LR6
คำนับ	92	110	72	132	60	60	110	122	72	60	92	98

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 ท่ายืนสวัสดี

ตารางที่ 4.4 มุมองศาของเซอร์โวมอเตอร์ แต่ละตัวในท่ายืนสวัสดี

STEP1/SV	HL1	HL2	HL3	HL4	HL5		HR1	HR2	HR3	HR4	HR5	
พนมมือ	40	180	0	82	90		150	0	160	142	90	
STEP2/SV	LL1	LL2	LL3	LL4	LL5	LL6	LR1	LR2	LR3	LR4	LR5	LR6
คำนับ	92	110	72	132	60	60	110	122	72	60	92	98
STEP3/SV	LL1	LL2	LL3	LL4	LL5	LL6	LR1	LR2	LR3	LR4	LR5	LR6
ยืน	92	110	90	102	58	54	106	122	56	84	96	96
STEP4/SV	HL1	HL2	HL3	HL4	HL5		HR1	HR2	HR3	HR4	HR5	
เหยียดแขนตรง	138	156	46	82	90		60	14	92	134	90	

4.2.5 ท่าเดินไปข้างหน้าระยะการก้าวสั้น

ตารางที่ 4.5 มุมองศาของเซอร์โวมอเตอร์ แต่ละตัวในท่าเดินไปข้างหน้าระยะการก้าวสั้น

STEP/SV	LL1	LL2	LL3	LL4	LL5	LL6	LR1	LR2	LR3	LR4	LR5	LR6
เดิน 1	92	110	82	114	58	60	106	122	44	94	94	100
เดิน 2	92	110	102	90	54	60	106	122	44	94	94	100
เดิน 3	92	110	100	88	56	60	106	122	68	72	92	100
เดิน 4	92	110	100	88	56	60	106	122	54	86	90	100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.6 ท่าเดินไปข้างหน้าระยะการก้าวยาว

ตารางที่ 4.6 มุมองศาของเซอร์โวมอเตอร์ แต่ละตัวในท่าเดินไปข้างหน้าระยะการก้าวยาว

STEP/SV	LL1	LL2	LL3	LL4	LL5	LL6	LR1	LR2	LR3	LR4	LR5	LR6
เดิน 1	92	92	90	102	60	28	106	104	42	84	90	72
เดิน 2	96	94	90	102	60	38	118	118	130	14	82	72
เดิน 3	92	128	98	110	55	80	106	146	130	14	95	118
เดิน 4	92	128	18	160	61	80	106	156	130	14	112	136
เดิน 5	92	110	90	102	58	54	106	122	56	84	96	96

4.2.7 ท่าเดินถอยหลังระยะการก้าวยาว

ตารางที่ 4.7 มุมองศาของเซอร์โวมอเตอร์ แต่ละตัวในท่าเดินถอยหลังระยะการก้าวยาว

STEP/SV	LL1	LL2	LL3	LL4	LL5	LL6	LR1	LR2	LR3	LR4	LR5	LR6
เดิน 1	92	128	18	160	61	80	106	156	130	14	112	136
เดิน 2	92	128	98	110	55	80	106	146	130	14	95	118
เดิน 3	96	94	90	102	60	38	118	118	130	14	82	72
เดิน 4	92	128	18	160	61	80	106	156	130	14	112	136
เดิน 5	92	92	90	102	60	28	106	104	42	84	90	72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.8 ท่าหันตัวไปทางซ้าย

ตารางที่ 4.8 มุมองศาของเซอร์โวมอเตอร์ แต่ละตัวในท่าหันตัวไปทางซ้าย

STEP/SV	LL1	LL2	LL3	LL4	LL5	LL6	LR1	LR2	LR3	LR4	LR5	LR6
หันซ้าย 1	88	110	50	132	62	78	106	146	56	84	106	116
หันซ้าย 2	134	110	68	116	62	78	106	146	56	84	106	116
หันซ้าย 3	92	90	106	84	58	40	110	124	92	56	96	105
หันซ้าย 4	92	110	90	102	58	54	106	122	56	84	96	96

4.2.9 ท่าหันตัวไปทางขวา

ตารางที่ 4.9 มุมองศาของเซอร์โวมอเตอร์ แต่ละตัวในท่าหันตัวไปทางขวา

STEP/SV	L1	L2	L3	L4	L5	L6	R1	R2	R3	R4	R5	R6
หันขวา 1	92	90	90	102	58	28	106	118	56	76	96	80
หันขวา 2	92	122	100	102	58	76	72	142	56	94	92	114
หันขวา 3	92	122	100	102	58	76	106	142	42	108	92	114
หันขวา 4	92	110	90	102	58	54	106	122	56	84	96	96

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

5.1 สรุปผลการทดลอง

จากการที่ได้ศึกษาระบบ CIM และได้ประดิษฐ์หุ่นยนต์สองขาเพื่อการขยายตัวทางอุตสาหกรรมขึ้นมานั้น เป็นการสร้างหุ่นยนต์ที่สามารถเลียนแบบการเคลื่อนไหวของมนุษย์ได้ โดยสามารถควบคุมให้เคลื่อนไหวในช่วงขาและช่วงแขน คือ เดินได้แบบมนุษย์และสามารถหนีบวัตถุขึ้นและวางวัตถุลงได้ โดยในแต่ละข้อต่อนั้น จะใช้ Servo motor ในการเคลื่อนไหวข้อต่อ จากนั้นทำการเขียนโปรแกรมเพื่อควบคุม Servo motor ให้หมุนไปในตำแหน่งต่างๆ เพื่อสร้างท่าทางของหุ่นยนต์ออกมา ในการควบคุมหุ่นยนต์ นอกจากต้องเขียนโปรแกรมแล้ว ยังมีอุปกรณ์ที่สำคัญคือแหล่งจ่ายไฟซึ่งได้ออกแบบให้สามารถรองรับความต้องการพลังงานไฟฟ้าของ Servo motor ได้

ซึ่งหุ่นยนต์ 2 ขาที่ทำได้นั้น สามารถเดินได้ และทำท่าทางได้ตามที่กำหนด เลี้ยวซ้าย เลี้ยวขวา และทำท่าทางการไหว้ได้ เนื่องจากหุ่นยนต์มีน้ำหนักค่อนข้างเยอะ ช่วงขาของหุ่นยนต์ยังไม่แข็งแรงพอ จึงทำให้ยากต่อการเดินและการก้าวขา เวลาเดินหุ่นยนต์อาจเสียหลักและล้มลงได้ จึงจะต้องมีการพัฒนาโปรแกรมและแก้ไขโครงสร้างตัวหุ่นยนต์ให้สมบูรณ์และมีเสถียรภาพต่อไป

5.2 ปัญหาและอุปสรรค

5.2.1 เนื่องจากต้องใช้เวลาในการเรียนรู้การทำงานของตัวหุ่นยนต์ จึงทำให้เกิดความล่าช้าในการทำงาน

5.2.2 ชิ้นส่วนที่ใช้ประกอบตัวหุ่นยนต์ มีขนาดไม่พอดีกัน ทำให้หุ่นยนต์ที่ประกอบออกมาแล้วมีรูปร่างไม่สวยงามเท่าที่ควร

5.2.3 หุ่นยนต์นั้นจะต้องมีการเซ็ทค่าต่างๆ ในระบบเป็นเวลานานพอสมควรก่อนที่หุ่นยนต์จะทำงาน และเมื่อหุ่นยนต์ทำงานแล้วเกิดการล้มลงแล้ว จะต้องทำการเซ็ทค่าใหม่ ก่อนที่หุ่นยนต์จะทำงานต่อไป

5.2.4 การทำงานของหุ่นยนต์ไม่ต่อเนื่องกัน

5.2.5 ขาดความรู้ความสามารถที่เพียงพอในการประดิษฐ์หุ่นยนต์

5.2.6 หุ่นยนต์ที่ประดิษฐ์มาแล้วนั้น ระบบยังขาดเสถียรภาพอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางการแก้ไขปัญหา

5.3.1 จะต้องออกแบบและเลือกใช้วัสดุในการทำตัวหุ่นยนต์ให้เหมาะสม

5.3.2 มีความสนใจและใฝ่รู้ที่จะศึกษาค้นคว้าข้อมูลในการทำงาน

5.3.3 ควรวัดขนาดชิ้นงานให้เป็นมาตรฐานเดียวกัน และใช้ความละเอียด เพื่องานที่สมบูรณ์ และสวยงาม

5.3.4 ปรับปรุงระบบให้มีเสถียรภาพมากขึ้น

5.3.5 ศึกษาข้อมูลของหุ่นยนต์กับผู้รู้ ค้นคว้าจากอินเตอร์เน็ตหรือหนังสือเพิ่มเติม

5.4 ข้อเสนอแนะและแนวทางในการค้นคว้าพัฒนา

จากการที่ทำการศึกษาจะต้องมีการค้นคว้าและศึกษาการทำงานของหุ่นยนต์สองขาให้มากยิ่งขึ้น เพื่อการออกแบบและเขียนโปรแกรมที่สมบูรณ์มากขึ้น ควรมีหลักการคำนวณเพื่อหาจุดที่สมดุลในการเคลื่อนที่ของหุ่นยนต์สองขา รวมทั้งพัฒนาและนำผลการทดลองที่มีมาทำให้หุ่นยนต์สองขาสามารถใช้งานได้จริงในเชิงอุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
โปรแกรมการออกแบบและผลิตชิ้นส่วนหุ่นยนต์

โปรแกรมโซลิดเวิร์ค ซึ่งเขียนเพื่อใช้ออกแบบหุ่นยนต์สองขาและทำการแปลงเป็นจีโค้ด (G-code) โดยใช้โปรแกรมซิลิดแคม (Solidcam) เพื่อผลิตชิ้นส่วนของหุ่นยนต์ ซึ่งมีหลักการเขียนโปรแกรมโค้ดและตัวอย่างของโค้ด ดังต่อไปนี้

G CODE LIST

Command	**Description**
G00	เดินเป็นเส้นตรง ควบคุมความเร็วตามRAPID
G01	เดินเป็นเส้นตรง ควบคุมความเร็วตามFEED OVERIDE
G02	เดินเป็นเส้นโค้ง ทิศทางตามเข็มนาฬิกา CW
G03	เดินเป็นเส้นโค้ง ทิศทางทวนเข็มนาฬิกา CCW
G04	หยุดทำงานชั่วขณะDwell, Exact stop
G09	หยุดตรงตำแหน่ง Exact stop
G10	ตั้งค่าข้อมูล Programmable data input
G17	การทำงานพร้อมกันของแกนXpYp plane selection
G18	การทำงานพร้อมกันของแกนZpXp plane selection
G19	การทำงานพร้อมกันของแกนYpZp plane selection
G20	การทำงานในระบบนิ้ว Input in inch
G21	การทำงานในระบบเมตริก Input in mm
G22	เปิดการทำงานฟังก์ชั่น Stored stroke check function on
G23	ปิดการทำงานฟังชั่น Stored stroke check function off
G25	ปิดการทำงานฟังก์ชั่น Spindle speed fluctuation detection off
G26	เปิดการทำงานฟังก์ชั่น Spindle speed fluctuation detection on
G27	ตรวจสอบการเลื่อนกลับตำแหน่งอ้างอิง Reference position
G28	เลื่อนกลับตำแหน่งอ้างอิง Automatic return to reference position
G29	เลื่อนออกจากตำแหน่งอ้างอิง Automatic return from reference position
G30	เลื่อน กลับตำแหน่งอ้างอิงที่ 2nd, 3rd and 4th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command	Description
G40	ยกเลิกคำสั่ง G41 G42
G41	ชดเชยค่ารัศมีของมีด ตัดทางด้านซ้าย(มีดตัดอยู่ด้านซ้ายของทิศทางการเดิน)
G42	ชดเชยค่ารัศมีของมีด ตัดทางด้านขวา(มีดตัดอยู่ด้านขวาของทิศทางการเดิน)
G43	ค่าความยาวของมีดตัด เป็นบวก
G44	ค่าความยาวของมีดตัด เป็นบวก
G45	เพิ่มค่ารัศมีของมีด ตัดTool offset increase
G46	ลดค่ารัศมีของมีด ตัดTool offset decrease
G47	เพิ่มค่ารัศมีของมีด ตัด2เท่า Tool offset double increase
G48	ลดค่ารัศมีของมีด ตัด2เท่า Tool offset double decrease
G49	ยกเลิกค่าความยาวของ มีดตัดTool length
G50	Scaling cancel
G51	Scaling
G52	การตั้งระบบ Local coordinate system setting
G53	เลือกMachine coordinate system selection
G54	ค่า X0 Y0 Z0 ของงานตัวที่1
G55	ค่า X0 Y0 Z0 ของงานตัวที่2
G56	ค่า X0 Y0 Z0 ของงานตัวที่3
G57	ค่า X0 Y0 Z0 ของงานตัวที่4
G58	ค่า X0 Y0 Z0 ของงานตัวที่5
G59	ค่า X0 Y0 Z0 ของงานตัวที่6
G61	โหมดหยุดตรง ตำแหน่งExact stop mode
G30	เลื่อน กลับตำแหน่งอ้างอิงที่ 2nd, 3rd and 4th
G40	ยกเลิกคำสั่ง G41 G42
G41	ชดเชยค่ารัศมีของมีด ตัดทางด้านซ้าย(มีดตัดอยู่ด้านซ้ายของทิศทางการเดิน)
G42	ชดเชยค่ารัศมีของมีด ตัดทางด้านขวา(มีดตัดอยู่ด้านขวาของทิศทางการเดิน)
G43	ค่าความยาวของมีดตัด เป็นบวก
G44	ค่าความยาวของมีดตัด เป็นบวก
G45	เพิ่มค่ารัศมีของมีด ตัดTool offset increase
G46	ลดค่ารัศมีของมีด ตัดTool offset decrease
G47	เพิ่มค่ารัศมีของมีด ตัด2เท่า Tool offset double increase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command	Description
G48	ลดค่ารัศมีของมีด ตัด2เท่า Tool offset double decrease
G49	ยกเลิกค่าความยาวของ มีดตัดTool length
G50	Scaling cancel
G51	Scaling
G52	การตั้งระบบ Local coordinate system setting
G53	เลือกMachine coordinate system selection
G54	ค่า X0 Y0 Z0 ของงานตัวที่ 1
G55	ค่า X0 Y0 Z0 ของงานตัวที่ 2
G56	ค่า X0 Y0 Z0 ของงานตัวที่ 3
G57	ค่า X0 Y0 Z0 ของงานตัวที่ 4
G58	ค่า X0 Y0 Z0 ของงานตัวที่ 5
G59	ค่า X0 Y0 Z0 ของงานตัวที่ 6
G61	โหมดหยุดตรง ตำแหน่งExact stop mode
G62	โหมดเดินมุมAutomatic corner override
G65	เรียกมาโคร โปรแกรม Macro call
G67	ยกเลิกคำสั่งมาโคร โปรแกรมMacro modal call cancel
G73	การเจาะแบบ Peck drilling cycle
G74	การทำเกลียวCounter tapping cycle
G76	การคว้านละเอียดFine boring cycle
G80	ยกเลิกคำสั่งCanned cycle cancel
G81	การเจาะDrilling cycle, spot boring cycle
G82	การเจาะDrilling cycle or counter boring cycle
G83	การเจาะPeck drilling cycle
G84	การทำเกลียวTapping cycle
G85	การใช้ดอกรีมเมอร์ Boring cycle
G86	การคว้านBoring cycle
G87	การคว้านBack boring cycle
G88	การคว้านBoring cycle
G89	การคว้านBoring cycle
G90	คำสั่งโดยคิดจากจุด อ้างอิงAbsolute command

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command	Description
G91	คำสั่งโดยคิดจากจุดที่ อยู่ครั้งสุดท้าย Increment command
G92	เปลี่ยนระบบ coordinate system or clamp at maximum spindle speed
G94	ฟีดต่อนาที Feed per minute
G95	ฟีดต่อรอบ Feed per rotation
G98	การเลื่อนกลับสู่ ตำแหน่ง Z+ ตัวสุดท้าย
G99	การเลื่อนกลับสู่ ตำแหน่ง R

M CODE LIST

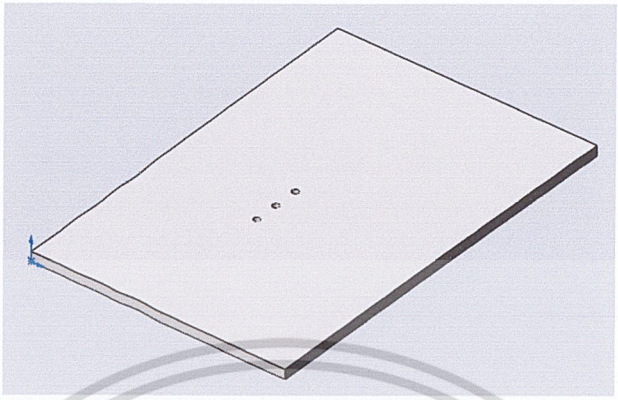
Command	Description
M00	Program Stop หยุดการทำงานของโปรแกรม
M01	Optional Program Stop หยุดการทำงานของเครื่องจักร
M02	End-of-Program จบการทำงานโปรแกรม
M30	End-of-Program จบการทำงานโปรแกรม
M03	Spindle Start Forward หัวหมุนตามเข็มนาฬิกา (CW)
M04	Spindle Start Reverse หัวหมุนทวนเข็มนาฬิกา (CCW)
M05	Spindle Stop หัวหยุดหมุน
M06	Run Tool Change Cycle , เปลี่ยนทูลM6 Txx
M07	Thru Spindle
M08	Coolant ON- Flood Coolant เปิดน้ำหล่อเย็น
M09	Coolant OFF ปิดน้ำหล่อเย็น
M11	Set Tool Carousel position to 1(ATC Tool Changer ONLY)
M19	cycle positioning(Not for spindle orientation) หัวอยู่ในตำแหน่งเปลี่ยนTool
M20	Does nothing Standard.
M22	Chip Conveyor Toggle ON/OF option
M29	Rigid Tap M29 Sxxxx in block prior to G84
M41	Low Gear Select
M42	High Gear Select
M48	100%Spindle Speed Override Forced

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command	Description
M49	Released
M60	AAxis Brake ON
M61	AAxis Brake OFF
M62	B Axis Brake ON
M63	B Axis Brake OFF
M64	M64/65 Output ON
M65	M64/65 Output OFF
M66	M66/67 Output ON, Chip Conveyor ON
M67	M66/67 Output OFF, Chip Conveyor ON
M68	M68/69 Output ON, Wash Down ON
M69	M68/69 Output OFF, Wash Down OFF
M78	M78/79 Output ON, Cooled spindle ON(2016 &3016L)
M79	M78/79 Output OFF, Cooled spindle OFF(2016 &3016L)
M80	MIDACO Pallet Changer 2 When the machine has MIDACO pallet option
M85	AAxis Rotary Table Enable
M86	AAxis Rotary Table Disable
M87	B Axis Rotary Table Enable
M88	B Axis Rotary Table Disable
M98	Sub-Program Call เรียก sub โปรแกรม
M99	Sub Program End จบ sub โปรแกรม ย้อนกลับ Program หลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างโค้ดที่ใช้ในการผลิตชิ้นงาน



รูปที่ ก.1 ชิ้นงานที่ใช้ในการเขียนโค้ด

```
%
O1 (FOOT)
N5 G0 G40 G49 G80 G21 (Initialisation)
N10 G0 G53 Z0 (Retour aux origines machine)
N15 G0 G53 X0 Y0
N20 (Outil n 1 - Diametre 2.0 D1 H1)
N25 T1 M6 D1 H1
N30 S1000 M3
N35 G0 Z5
N40 (F-contour-T1)
N45 G0 G54 X-1. Y20.
N50 G43 H1 Z5.
N55 G0 Z2.
N60 G1 Z-0.875 F33
N65 G1 Y80. F100
N70 G2 X0. Y81. I1. J0.
N75 G1 X120.
N80 G2 X121. Y80. I0. J-1.
N85 G1 Y0.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

N90 G2 X120. Y-1. I-1. J0.

N95 G1 X0.

N100 G2 X-1. Y0. I0. J1.

N105 G1 Y20.

N110 G1 Z-1.75 F33

N115 G1 Y80. F100

N120 G2 X0. Y81. I1. J0.

N125 G1 X120.

N130 G2 X121. Y80. I0. J-1.

N135 G1 Y0.

N140 G2 X120. Y-1. I-1. J0.

N145 G1 X0.

N150 G2 X-1. Y0. I0. J1.

N155 G1 Y20.

N160 G1 Z-2.625 F33

N165 G1 Y80. F100

N170 G2 X0. Y81. I1. J0.

N175 G1 X120.

N180 G2 X121. Y80. I0. J-1.

N185 G1 Y0.

N190 G2 X120. Y-1. I-1. J0.

N195 G1 X0.

N200 G2 X-1. Y0. I0. J1.

N205 G1 Y20.

N210 G1 Z-3.5 F33

N215 G1 Y80. F100

N220 G2 X0. Y81. I1. J0.

N225 G1 X120.

N230 G2 X121. Y80. I0. J-1.

N235 G1 Y0.

N240 G2 X120. Y-1. I-1. J0.

N245 G1 X0.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

N250 G2 X-1. Y0. I0. J1.

N255 G1 Y20.

N260 G1 Y80.

N265 G2 X0. Y81. I1. J0.

N270 G1 X120.

N275 G2 X121. Y80. I0. J-1.

N280 G1 Y0.

N285 G2 X120. Y-1. I-1. J0.

N290 G1 X0.

N295 G2 X-1. Y0. I0. J1.

N300 G1 Y20.

N305 G0 Z5.

N310 G0 G53 Z5

N315 G0 G53 X0 Y0 M5

M30

%

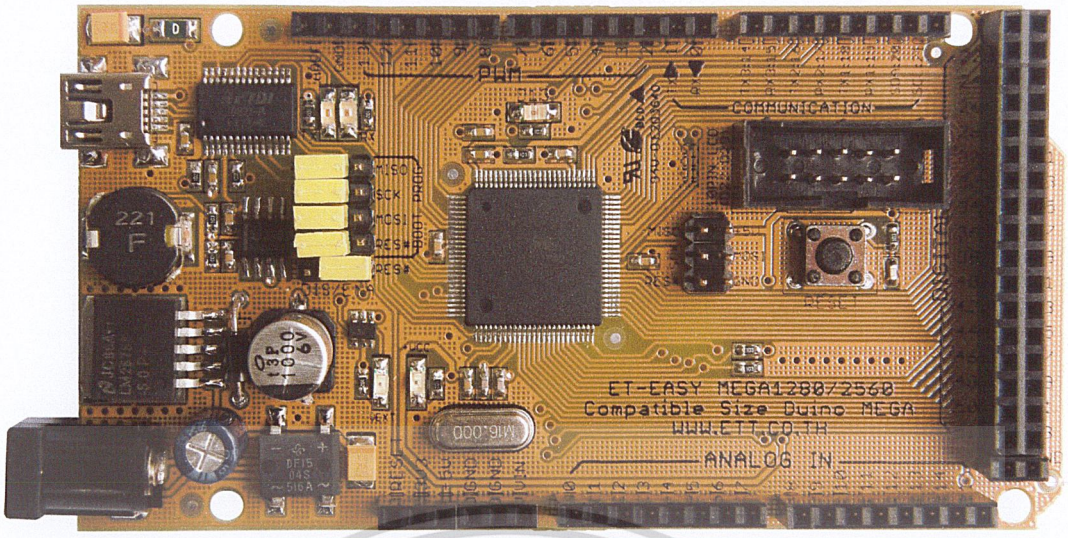
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
เอกสารคู่มืออุปกรณ์อิเล็กทรอนิกส์

เอกสารคู่มือการใช้งานของ Easy Mega1280

จากการที่ Arduino ที่เป็นโครงการพัฒนาระบบ MCU ของ AVR แบบ Open Source ได้รับการแนะนำเผยแพร่ออกมาสู่สาธารณะ ซึ่งได้รับความนิยมกันอย่างแพร่หลายจากผู้คนทั่วโลกภายในระยะเวลา อันรวดเร็ว ทางด้านของ Software ก็มีการพัฒนากันอย่างต่อเนื่อง ซึ่งในขณะนี้ (สิงหาคม 2552) โปรแกรมของ Arduino ได้รับการปรับปรุงเป็น Version "arduino-0016" แล้ว โดยทางด้าน Hardware เองก็ได้มีการพัฒนาปรับปรุงอย่างต่อเนื่องควบคู่กันไปด้วยเช่นเดียวกัน ซึ่งจากเดิมที่มีการพัฒนาโปรแกรมให้รองรับกับการใช้งานได้กับชิพ MCU รุ่นเล็ก 28 ขา อย่าง ATMEGA8,ATMEGA88/ATMEGA168/ATMEGA328 สำหรับเป็นจุดเริ่มต้นให้ผู้สนใจได้ใช้เป็นเครื่องมือและอุปกรณ์ในการศึกษาทดลองเรียนรู้ ตลอดจนถึงนำไปประยุกต์ใช้งานกันนั้น มาถึงวันนี้ ขนาดของทรัพยากรต่างๆ รวมทั้งขนาดของหน่วยความจำ สำหรับเขียนโปรแกรม ที่มีอยู่ในชิพ AVR รุ่นเล็กที่มีอยู่เริ่มไม่เพียงพอกับการประยุกต์ใช้งานในงานบางประเภทแล้ว ทาง Arduino เอง จึงได้ทำการพัฒนาให้ Arduino สามารถรองรับการใช้งานขนาดใหญ่ขึ้นอีก โดยปรับปรุงโปรแกรมให้ใช้กับชิพ AVR รุ่นใหญ่ขึ้น เพื่อให้มีจำนวน I/O ทั้ง Digital, Analog, PWM, UART และ ขนาดหน่วยความจำเพิ่มมากขึ้นกว่าเดิม แต่ก็ยังคงใช้แนวทางในการพัฒนาโปรแกรม แบบเดียวกันกับรุ่นเล็กทุกประการ โดยได้เลือกใช้ชิพ AVR เบอร์ ATMEGA1280 และออกแบบพัฒนา Hardware บอร์ดขึ้นมารองรับ โดยใช้รหัสชื่อรุ่นว่า "Arduino Mega" ออกวางจำหน่าย และได้มีการเผยแพร่รายละเอียดทาง Hardware ต่างๆให้ผู้สนใจนำไปพัฒนาต่อยอดกันได้เอง แต่เนื่องจากชิพดังกล่าวมีโครงสร้างตัวถังเป็นแบบ SMD จึงทำให้เป็นอุปสรรคสำหรับผู้ใช้งานจำนวนไม่น้อยในการที่จะผลิตหรือสร้างบอร์ดขึ้นใช้งานเอง ทาง อีทีที จึงได้นำ ATMEGA1280 มาพัฒนาเป็นบอร์ดโดยให้มีโครงสร้างการทำงานเช่นเดียวกันกับ Arduino Mega ขึ้นมา โดยใช้ชื่อว่า **"ET-EASY MEGA1280"** โดยได้ออกแบบให้มีการจัดสรร Pin I/O ต่างๆ รวมทั้งขนาดให้ตรงตามมาตรฐานของบอร์ด **"Arduino Mega"** เพียงแต่ได้มีการปรับปรุงข้อจำกัด บางอย่างให้ดีขึ้นกว่า Arduino Mega รุ่นมาตรฐาน เพื่อเพิ่มความสะดวกกับผู้ใช้งานมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.1 ET EASY MEGA1280 (Duino Mega)

คุณสมบัติของบอร์ด

- ใช้ ATMEGA1280 เป็น MCU ประจำบอร์ด Run ความถี่ 16MHz จาก Crystal Oscillator
- 128KByte Flash (สงวนไว้ 4KByte สำหรับ Boot loader) / 8KByte SRAM / 4KByte EEPROM
- รองรับการพัฒนาโปรแกรมด้วยภาษา C++ ของ Arduino ตามแบบ Arduino Mega ได้ 100%
- ใช้ USB Bridge ของ FTDI เบอร์ FT232RL พร้อม Over Current Protection สำหรับติดต่อ สื่อสารและ Download Code จากคอมพิวเตอร์ให้บอร์ด พร้อม Jumper สำหรับปรับใช้งานบอร์ดเป็นการProgram Boot loader ให้กับ MCU จากพอร์ต USB ในบอร์ดได้เอง โดยไม่ต้องใช้เครื่องโปรแกรม AVRISP จากภายนอก
- 54 Pin Digital I/O โดยมี 14Pin สามารถโปรแกรมหน้าที่เป็น PWM ได้
- 16 Pin Analog Input (ADC ขนาด 10 บิต 16 ช่อง)
- 4 UART(Hardware Serial Port) แบบ TTL Logic
- ขนาดของ PCB บอร์ด และ ตำแหน่ง Pin Connector ต่างๆ ตรงกันกับ Arduino Mega ทั้งหมด ทำให้สามารถนำไปติดตั้งใช้งานร่วมกับบอร์ด Shield แบบต่างๆที่มีการผลิตขึ้นมาใช้งานร่วมกันกับบอร์ด Arduino Mega ได้ทั้งหมด โดยบอร์ดมีขนาด PCB Size 5.3cm x 10.2cm
- มีขั้ว Header 10Pin IDE ของ 8 บิต Digital I/O(D22...D29) สำหรับเชื่อมต่อกับ LCD หรือบอร์ดI/O แบบต่างๆ ของ อีทีที เพิ่มความสะดวกในการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รองรับการใช้งานกับ External Supply ทั้งแบบ AC และ DC ขนาด 7-12V โดยเลือกใช้ Regulate แบบ Switching ขนาด 1A (LM2575-5V) ลดปัญหาเรื่องความร้อนเมื่อมีการใช้กระแสสูงๆ สามารถใช้แหล่งจ่ายจากพอร์ต USB ได้ในกรณีใช้กระแสไม่เกิน 500mA โดยมีวงจรเลือกแหล่งจ่ายอัตโนมัติ โดยจะตัดการใช้ไฟเลี้ยงจาก USB โดยอัตโนมัติ เมื่อมีการต่อแหล่งจ่ายจากภายนอกให้บอร์ด

- **หมายเลข 1** คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงจากภายนอก สามารถใช้ได้กับแหล่งจ่ายทั้งแบบ AC และ DC พร้อมวงจร Bridge Rectifier และ Regulate แบบ Switching ช่วยลดความร้อนของ IC Regulate เมื่อมีการดึงกระแสมากๆ ได้เป็นอย่างดี สามารถใช้กับแรงดัน Input 7-12V

- **หมายเลข 2** เป็นขั้วต่อ USB สำหรับติดต่อสื่อสารกับคอมพิวเตอร์ PC โดยใช้ FT232RL เป็น USB Bridge ในการเชื่อมต่อระหว่างคอมพิวเตอร์ PC และ MCU ในบอร์ด และยังสามารถใช้ไฟจาก พอร์ต USB เป็นแหล่งจ่ายให้กับบอร์ดได้ด้วย โดยจะมี Poly Fuse ขนาด 500mA สำหรับ ป้องกันการดึงกระแสเกินจากพอร์ต USB ด้วย และที่พิเศษคือมีวงจรสำหรับตรวจสอบแหล่งจ่าย เพื่อสลับการใช้งานแหล่งจ่ายจาก USB ไปเป็น External Supply ได้เอง โดยอัตโนมัติ โดยเมื่อไม่ได้ต่อ External Supply บอร์ดจะใช้ไฟจากพอร์ต USB เป็นแหล่งจ่ายในการทำงาน แต่เมื่อ มีการต่อ External Supply วงจรจะสลับไปใช้แหล่งจ่ายจาก External Supply เองโดยอัตโนมัติ

  - o **LED +VCC** ใช้แสดงสถานะเมื่อมีการจ่ายไฟให้กับบอร์ด
  - o **LED VEXT** ใช้แสดงสถานะเมื่อมีการจ่ายไฟจาก External Supply

- **หมายเลข 3** เป็น LED VEXT ใช้แสดงสถานะเมื่อมีการจ่ายไฟเลี้ยงจาก External Supply

- **หมายเลข 4** เป็น LED +VCC ใช้แสดงสถานะของแหล่งจ่ายไฟเลี้ยง (+VCC) ของบอร์ด โดยเมื่อบอร์ดใช้แหล่งจ่ายจาก External Supply จะแสดงสถานะโดยการให้ LED VEXT และ LED +VCC ติดสว่างพร้อมกันทั้งคู่ แต่ถ้าบอร์ดใช้แหล่งจ่ายจากพอร์ต USB จะแสดงสถานะโดยการให้ LED +VCC ติดสว่างเพียงดวงเดียว

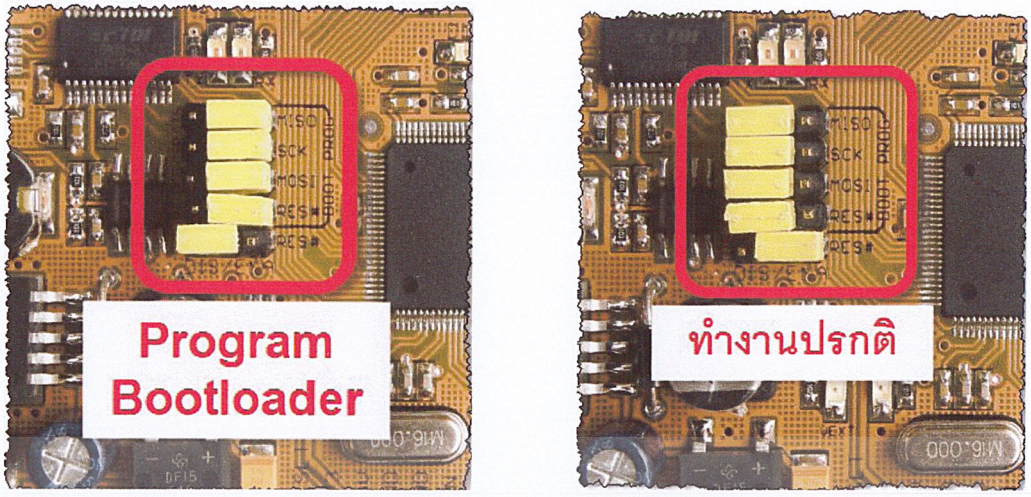
- **หมายเลข 5** เป็น LED แสดงสถานะของ RX และ TX ใช้สำหรับแสดงการรับส่งข้อมูลระหว่างบอร์ด ET-EASY MEGA1280 กับคอมพิวเตอร์ PC ผ่านทางพอร์ต USB

- **หมายเลข 6** เป็น LED D13 ใช้สำหรับทดแสดงการทำงานของ Bootloader และ ใช้ทดสอบการทำงานของบอร์ดจากการควบคุมของ Pin Digital-13 ทำงานด้วย Logic “1” และ หยุดทำงานด้วย Logic “0”

- **หมายเลข 7** เป็นสวิตช์ Reset ใช้สำหรับสั่ง Reset การทำงานของบอร์ด

- **หมายเลข 8** เป็นชุด Jumper สำหรับเลือก การ Program Bootloader ผ่าน USB Port และการใช้งานตามปรกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.2 แสดงการใช้งานของ bootloader

- **หมายเลข 9** เป็นขั้วต่อ AVRISP ใช้สำหรับ Download Code ให้กับ MCU โดยขั้วต่อ AVRISP นี้ จะสามารถใช้งานได้กับเครื่องโปรแกรมทุกรุ่นที่รองรับการใช้งานกับ ATMEGA1280 และใช้ขั้วต่อตรงตามมาตรฐาน AVRISP
- **หมายเลข 10** เป็นขั้วต่อสัญญาณจาก D[22..29] สำหรับเชื่อมต่อกับบอร์ด I/O ของ อีทีที รวมทั้ง จอแสดงผล LCD โดยใช้ร่วมกับ 10PIN LCD หรือ ET-CONV SPI TO LCD โดยในกรณีที่ต้องการนำขั้วต่อ 10PIN ไปเชื่อมต่อเพื่อใช้ควบคุมการแสดงผลของ LCD นั้น เพื่อความสะดวกในการใช้งาน ขอแนะนำให้จัดหาชุด ET-10PIN CLCD (ET-CONV 10 TO LCD) มาเป็นตัวกลางในการเชื่อมต่อระหว่าง ขั้ว IDE 10PIN ของบอร์ด ET-EASY MEGA1280 ดังตัวอย่างโดยเมื่อนำ LCD มาติดตั้งเข้ากับชุด ET-10PIN CLCD เรียบร้อยแล้วจะทำให้สามารถเชื่อมต่อสายสัญญาณระหว่างขั้ว ET-10PIN CLCD กับ 10PIN ได้โดยง่าย โดยใช้การเชื่อมต่อผ่านทางสายแพร 10PIN ได้ทันที

คุณสมบัติของสัญญาณต่างๆของบอร์ด ET-EASY MEGA1280

- **RESET#** เป็นสัญญาณ Input Logic Reset ของ MCU เมื่อเป็น Logic Low จะทำให้ MCU อยู่ในสภาวะรีเซ็ต เมื่อเป็น Logic High จะทำให้ MCU อยู่ในสภาวะทำงานตามปรกติ โดยสัญญาณ RESET# นี้จะถูกควบคุมจาก 2 แหล่ง คือ จาก สวิตช์ RESET ภายในบอร์ด และจากสัญญาณ DTR ของ FT232RL ถ้ามีการเลือก Enable Jumper ของ Auto Reset จาก DTR ไว้
- **+3V3** เป็นแหล่งจ่ายไฟขนาด +3.3V ที่ได้จากวงจร Regulate ภายในของ FT232RL สามารถจ่ายกระแสได้สูงสุด 50mA ซึ่งเมื่อต้องการนำแหล่งจ่าย +3.3V นี้ไปใช้งานเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แหล่งจ่ายให้กับอุปกรณ์ต่างๆ ต้องระมัดระวังเรื่องการดึงกระแสของโหลดด้วย ถ้าโหลดมีการดึงกระแสมากกว่า 50mA อาจทำให้ FT232RL เกิดความเสียหายได้

- **+VIN** เป็นไฟ DC ที่รับมาจาก Jack VIN(External Supply) แต่ผ่านการ Rectifier และ Filter เป็น DC แล้ว มีขนาดแรงดันเฉลี่ยตามขนาดแรงดันที่ป้อนให้กับบอร์ดทาง Jack VIN
- **+5V** เป็นจุดต่อแหล่งจ่ายไฟของบอร์ดออกไปใช้งาน ซึ่งมาจากแหล่งกำเนิด 2 แหล่ง คือ จากพอร์ต USB และจาก External Supply ซึ่งถ้าต่อแหล่งจ่ายให้บอร์ดจาก External Supply ผ่านทาง Jack VIN แหล่งจ่าย +5V นี้จะมาจาก Switching Regulate (LM2575-5V) สามารถจ่ายกระแสได้สูงสุด ถึง 1A แต่ถ้าใช้แหล่งจ่ายจากพอร์ต USB แหล่งจ่าย +5V นี้จะมาจากพอร์ต USB โดยตรง โดยจะมี ฟิวส์ แบบ Poly ขนาด 500mA ต่อป้องกันการดึงกระแสเกินเพื่อป้องกันความเสียหายของพอร์ต USB โดยจะจ่ายกระแสได้สูงสุดไม่เกิน 500mA ขึ้นอยู่กับคุณสมบัติในการจ่ายกระแสของพอร์ต USB และการ Configure ค่าให้กับ FT232RL ด้วย
- **A0-A15** เป็นขาสัญญาณ Analog Input แบบ ADC มีขนาดความละเอียด 10บิต มี 16 Pin สามารถรับแรงดัน Analog Input ได้ 0-5VDC
- **D0-D53** เป็นขาสัญญาณ Digital Input/Output แบบ TTL มีทั้งหมด 54 Pin สามารถใช้ทำหน้าที่เป็น Input หรือ Output ตามการกำหนดจากโปรแกรม โดยมีบาง Pin สามารถกำหนดหน้าที่ใช้งานเป็นฟังก์ชั่นพิเศษต่างๆเพิ่มเติมได้อีก
    - o **D0-D1** ถูกสงวนไว้ใช้ทำหน้าที่เป็นพอร์ตสื่อสารอนุกรม RS232 (UART0) โดยได้ทำการเชื่อมต่อกับ USB Bridge ของ FT232RL เพื่อใช้ Upload Code ให้กับบอร์ด และยังสามารถใช้ทดลองติดต่อสื่อสารรับส่งข้อมูลระหว่างบอร์ดกับคอมพิวเตอร์ PC ได้
    - o **D2-D13** สามารถ โปรแกรมหน้าที่เป็น PWM ขนาด 8 บิต มี 14 Pin ได้
    - o **D14** สามารถ โปรแกรมหน้าที่เป็น TX3 สำหรับ ส่งข้อมูลของ UART3 ได้
    - o **D15** สามารถ โปรแกรมหน้าที่เป็น RX3 สำหรับ รับข้อมูลให้กับ UART3 ได้
    - o **D16** สามารถ โปรแกรมหน้าที่เป็น TX2 สำหรับ ส่งข้อมูลของ UART2 ได้
    - o **D17** สามารถ โปรแกรมหน้าที่เป็น RX2 สำหรับ รับข้อมูลให้กับ UART2 ได้
    - o **D18** สามารถ โปรแกรมหน้าที่เป็น TX1 สำหรับ ส่งข้อมูลของ UART1 ได้ด้วย
    - o **D19** สามารถ โปรแกรมหน้าที่เป็น RX1 สำหรับ รับข้อมูลให้กับ UART1 ได้
    - o **D20** สามารถ โปรแกรมหน้าที่เป็น SDA ของ I2C Bus สำหรับใช้สื่อสารกับ I2C
    - o **D21** สามารถ โปรแกรมหน้าที่เป็น SCL ของ I2C Bus สำหรับใช้สื่อสารกับ I2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **AREF** เป็นสัญญาณ Analog Reference จากภายนอกที่ต้องการป้อนให้กับ MCU ซึ่งตามปรกติแล้ว ATMEGA1280 สามารถโปรแกรมให้เลือกใช้แรงดันอ้างอิงจากภายในได้อยู่แล้วโดยสามารถ เลือกเป็น 1.1V หรือ 2.56V หรือ AVCC(+5V) โดยไม่จำเป็นต้องป้อนแรงดันอ้างอิงจากภายนอกให้กับบอร์ดอีก แต่ถ้าต้องการแรงดันอ้างอิงที่มีความแตกต่างจากที่กล่าวมาแล้วก็สามารถป้อนเป็น แรงดันอ้างอิงจากภายนอกผ่านทางขา AREF นี้เข้าไปเองได้ระหว่าง 0-5V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

โปรแกรมควบคุมหุ่นยนต์เสมือนมนุษย์

```
//INCLUDE LIBRARY
#include <Servo.h>
#include <Keypad.h>
#include <EEPROM.h>
//-------------------------------------------------------------------------------------------
//KEYPAD SETUP
const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
//define the symbols on the buttons of the keypads
char hexaKeys[ROWS][COLS] =
{
  {'0','1','2','3'},
  {'4','5','6','7'},
  {'8','9','a','b'},
  {'c','d','e','f'}
};
byte rowPins[ROWS] = {7, 6, 5, 9}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {8, 2, 3, 4}; //connect to the column pinouts of the keypad

//initialize an instance of class NewKeypad
Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);
//-------------------------------------------------------------------------------------------
//SERVO SETUP
//LEFT LEG
Servo servo1;
Servo servo2;
Servo servo3;
Servo servo4;
Servo servo5;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Servo servo6;

//RIGHT LEG
Servo servo7;
Servo servo8;
Servo servo9;
Servo servo10;
Servo servo11;
Servo servo12;

//LEFT ARM
Servo servo13;
Servo servo14;
Servo servo15;
Servo servo16;
Servo servo17;

//RIGHT ARM
Servo servo18;
Servo servo19;
Servo servo20;
Servo servo21;
Servo servo22;

//HEAD
Servo servo23;
//-------------------------------------------------------------------------------------------
//VARIABLE
//LEFT LEG
int pos1 = 92;
int pos2 = 110;
int pos3 = 90;
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int pos4 = 102;
int pos5 = 58;
int pos6 = 54;

//RIGHT LEG
int pos7 = 106;
int pos8 = 122;
int pos9 = 56;
int posa = 84;
int posb = 96;
int posc = 96;

//LEFT ARM
int posd = 138;
int pose = 156;
int posf = 46;
int posg = 82;
int posh = 90;

//RIGHT ARM
int posi = 60;
int posj = 14;
int posk = 92;
int posl = 134;
int posm = 90;

//HEAD
int posn = 63;
//--------------------------------------------------------------------------------
//PIN SETUP
void setup()
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  servo1.attach(23);
  servo2.attach(25);
  servo3.attach(27);
  servo4.attach(29);
  servo5.attach(31);
  servo6.attach(33);

  servo7.attach(22);
  servo8.attach(24);
  servo9.attach(26);
  servo10.attach(28);
  servo11.attach(30);
  servo12.attach(32);

  servo13.attach(35);
  servo14.attach(37);
  servo15.attach(39);
  servo16.attach(41);
  servo17.attach(43);

  servo18.attach(34);
  servo19.attach(36);
  servo20.attach(38);
  servo21.attach(40);
  servo22.attach(42);

  servo23.attach(44);
}
//--------------------------------------------------------------------------------
//PROGRAM
void loop()
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char customKey = customKeypad.getKey();

servo1.write(pos1);
servo2.write(pos2);
servo3.write(pos3);
servo4.write(pos4);
servo5.write(pos5);
servo6.write(pos6);

servo7.write(pos7);
servo8.write(pos8);
servo9.write(pos9);
servo10.write(posa);
servo11.write(posb);
servo12.write(posc);

servo13.write(posd);
servo14.write(pose);
servo15.write(posf);
servo16.write(posg);
servo17.write(posh);

servo18.write(posi);
servo19.write(posj);
servo20.write(posk);
servo21.write(posl);
servo22.write(posm);

servo23.write(posn);
//-------------------------------------------------------------------------
//BOW
    if (customKey == '4')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 float z = 50;
  float new1 = 92;
  float new2 = 110;
  float new3 = 72;
  float new4 = 132;
  float new5 = 60;
  float new6 = 60;
  float new7 = 110;
  float new8 = 122;
  float new9 = 72;
  float newa = 60;
  float newb = 92;
  float newc = 98;

  float old1 = pos1;
  float old2 = pos2;
  float old3 = pos3;
  float old4 = pos4;
  float old5 = pos5;
  float old6 = pos6;
  float old7 = pos7;
  float old8 = pos8;
  float old9 = pos9;
  float olda = posa;
  float oldb = posb;
  float oldc = posc;

  float p1 = (new1-old1)/z;
  float p2 = (new2-old2)/z;
  float p3 = (new3-old3)/z;
  float p4 = (new4-old4)/z;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float p5 = (new5-old5)/z;
float p6 = (new6-old6)/z;
float p7 = (new7-old7)/z;
float p8 = (new8-old8)/z;
float p9 = (new9-old9)/z;
float pa = (newa-olda)/z;
float pb = (newb-oldb)/z;
float pc = (newc-oldc)/z;

 for(int d=0;d<=z;d++)
 {
  float k1 = old1+(p1*d);
  float k2 = old2+(p2*d);
  float k3 = old3+(p3*d);
  float k4 = old4+(p4*d);
  float k5 = old5+(p5*d);
  float k6 = old6+(p6*d);
  float k7 = old7+(p7*d);
  float k8 = old8+(p8*d);
  float k9 = old9+(p9*d);
  float ka = olda+(pa*d);
  float kb = oldb+(pb*d);
  float kc = oldc+(pc*d);

  servo1.write(k1);
  servo2.write(k2);
  servo3.write(k3);
  servo4.write(k4);
  servo5.write(k5);
  servo6.write(k6);
  servo7.write(k7);
  servo8.write(k8);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            servo9.write(k9);
            servo10.write(ka);
            servo11.write(kb);
            servo12.write(kc);

            delay(10);

            pos1 = new1;
            pos2 = new2;
            pos3 = new3;
            pos4 = new4;
            pos5 = new5;
            pos6 = new6;
            pos7 = new7;
            pos8 = new8;
            pos9 = new9;
            posa = newa;
            posb = newb;
            posc = newc;
          }
      }
//-------------------------------------------------------------------------------------------
//FORWARD SHORT
      if (customKey == '6')
      {
       float z = 4;
       for (int bb=0; bb <= 1; bb++)  //1
       {
        float new1 = 92;
        float new2 = 110;
        float new3 = 82;
        float new4 = 114;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float new5 = 58;
float new6 = 60;
float new7 = 106;
float new8 = 122;
float new9 = 44;
float newa = 94;
float newb = 94;
float newc = 100;

float old1 = pos1;
float old2 = pos2;
float old3 = pos3;
float old4 = pos4;
float old5 = pos5;
float old6 = pos6;
float old7 = pos7;
float old8 = pos8;
float old9 = pos9;
float olda = posa;
float oldb = posb;
float oldc = posc;

float p1 = (new1-old1)/z;
float p2 = (new2-old2)/z;
float p3 = (new3-old3)/z;
float p4 = (new4-old4)/z;
float p5 = (new5-old5)/z;
float p6 = (new6-old6)/z;
float p7 = (new7-old7)/z;
float p8 = (new8-old8)/z;
float p9 = (new9-old9)/z;
float pa = (newa-olda)/z;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float pb = (newb-oldb)/z;
float pc = (newc-oldc)/z;


 for(int d=0;d<=z;d++)
 {
  float k1 = old1+(p1*d);
  float k2 = old2+(p2*d);
  float k3 = old3+(p3*d);
  float k4 = old4+(p4*d);
  float k5 = old5+(p5*d);
  float k6 = old6+(p6*d);
  float k7 = old7+(p7*d);
  float k8 = old8+(p8*d);
  float k9 = old9+(p9*d);
  float ka = olda+(pa*d);
  float kb = oldb+(pb*d);
  float kc = oldc+(pc*d);

  servo1.write(k1);
  servo2.write(k2);
  servo3.write(k3);
  servo4.write(k4);
  servo5.write(k5);
  servo6.write(k6);
  servo7.write(k7);
  servo8.write(k8);
  servo9.write(k9);
  servo10.write(ka);
  servo11.write(kb);
  servo12.write(kc);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  delay(10);
```

```
        pos1 = new1;
        pos2 = new2;
        pos3 = new3;
        pos4 = new4;
        pos5 = new5;
        pos6 = new6;
        pos7 = new7;
        pos8 = new8;
        pos9 = new9;
        posa = newa;
        posb = newb;
        posc = newc;
      }
  }
  for (int bb=0; bb <= 1; bb++)  //2
  {
    float new1 = 92;
    float new2 = 110;
    float new3 = 107;
    float new4 = 90;
    float new5 = 54;
    float new6 = 60;
    float new7 = 106;
    float new8 = 122;
    float new9 = 40;
    float newa = 94;
    float newb = 94;
    float newc = 100;


    float old1 = pos1;
    float old2 = pos2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float old3 = pos3;
float old4 = pos4;
float old5 = pos5;
float old6 = pos6;
float old7 = pos7;
float old8 = pos8;
float old9 = pos9;
float olda = posa;
float oldb = posb;
float oldc = posc;

float p1 = (new1-old1)/z;
float p2 = (new2-old2)/z;
float p3 = (new3-old3)/z;
float p4 = (new4-old4)/z;
float p5 = (new5-old5)/z;
float p6 = (new6-old6)/z;
float p7 = (new7-old7)/z;
float p8 = (new8-old8)/z;
float p9 = (new9-old9)/z;
float pa = (newa-olda)/z;
float pb = (newb-oldb)/z;
float pc = (newc-oldc)/z;

 for(int d=0;d<=z;d++)
 {
  float k1 = old1+(p1*d);
  float k2 = old2+(p2*d);
  float k3 = old3+(p3*d);
  float k4 = old4+(p4*d);
  float k5 = old5+(p5*d);
  float k6 = old6+(p6*d);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float k7 = old7+(p7*d);
float k8 = old8+(p8*d);
float k9 = old9+(p9*d);
float ka = olda+(pa*d);
float kb = oldb+(pb*d);
float kc = oldc+(pc*d);

servo1.write(k1);
servo2.write(k2);
servo3.write(k3);
servo4.write(k4);
servo5.write(k5);
servo6.write(k6);
servo7.write(k7);
servo8.write(k8);
servo9.write(k9);
servo10.write(ka);
servo11.write(kb);
servo12.write(kc);

delay(10);

pos1 = new1;
pos2 = new2;
pos3 = new3;
pos4 = new4;
pos5 = new5;
pos6 = new6;
pos7 = new7;
pos8 = new8;
pos9 = new9;
posa = newa;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      posb = newb;
      posc = newc;
    }
  }
  for (int bb=0; bb <= 1; bb++)  //3
  {
   float new1 = 92;
   float new2 = 110;
   float new3 = 105;
   float new4 = 88;
   float new5 = 56;
   float new6 = 60;
   float new7 = 106;
   float new8 = 122;
   float new9 = 68;
   float newa = 72;
   float newb = 92;
   float newc = 100;

   float old1 = pos1;
   float old2 = pos2;
   float old3 = pos3;
   float old4 = pos4;
   float old5 = pos5;
   float old6 = pos6;
   float old7 = pos7;
   float old8 = pos8;
   float old9 = pos9;
   float olda = posa;
   float oldb = posb;
   float oldc = posc;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float p1 = (new1-old1)/z;
float p2 = (new2-old2)/z;
float p3 = (new3-old3)/z;
float p4 = (new4-old4)/z;
float p5 = (new5-old5)/z;
float p6 = (new6-old6)/z;
float p7 = (new7-old7)/z;
float p8 = (new8-old8)/z;
float p9 = (new9-old9)/z;
float pa = (newa-olda)/z;
float pb = (newb-oldb)/z;
float pc = (newc-oldc)/z;

 for(int d=0;d<=z;d++)
 {
  float k1 = old1+(p1*d);
  float k2 = old2+(p2*d);
  float k3 = old3+(p3*d);
  float k4 = old4+(p4*d);
  float k5 = old5+(p5*d);
  float k6 = old6+(p6*d);
  float k7 = old7+(p7*d);
  float k8 = old8+(p8*d);
  float k9 = old9+(p9*d);
  float ka = olda+(pa*d);
  float kb = oldb+(pb*d);
  float kc = oldc+(pc*d);

  servo1.write(k1);
  servo2.write(k2);
  servo3.write(k3);
  servo4.write(k4);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    servo5.write(k5);
    servo6.write(k6);
    servo7.write(k7);
    servo8.write(k8);
    servo9.write(k9);
    servo10.write(ka);
    servo11.write(kb);
    servo12.write(kc);

    delay(10);

    pos1 = new1;
    pos2 = new2;
    pos3 = new3;
    pos4 = new4;
    pos5 = new5;
    pos6 = new6;
    pos7 = new7;
    pos8 = new8;
    pos9 = new9;
    posa = newa;
    posb = newb;
    posc = newc;
   }
}
for (int bb=0; bb <= 1; bb++)  //4
{
  float new1 = 92;
  float new2 = 110;
  float new3 = 105;
  float new4 = 88;
  float new5 = 56;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float new6 = 60;
float new7 = 106;
float new8 = 122;
float new9 = 50;
float newa = 86;
float newb = 90;
float newc = 100;

float old1 = pos1;
float old2 = pos2;
float old3 = pos3;
float old4 = pos4;
float old5 = pos5;
float old6 = pos6;
float old7 = pos7;
float old8 = pos8;
float old9 = pos9;
float olda = posa;
float oldb = posb;
float oldc = posc;

float p1 = (new1-old1)/z;
float p2 = (new2-old2)/z;
float p3 = (new3-old3)/z;
float p4 = (new4-old4)/z;
float p5 = (new5-old5)/z;
float p6 = (new6-old6)/z;
float p7 = (new7-old7)/z;
float p8 = (new8-old8)/z;
float p9 = (new9-old9)/z;
float pa = (newa-olda)/z;
float pb = (newb-oldb)/z;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float pc = (newc-oldc)/z;


 for(int d=0;d<=z;d++)
 {
  float k1 = old1+(p1*d);
  float k2 = old2+(p2*d);
  float k3 = old3+(p3*d);
  float k4 = old4+(p4*d);
  float k5 = old5+(p5*d);
  float k6 = old6+(p6*d);
  float k7 = old7+(p7*d);
  float k8 = old8+(p8*d);
  float k9 = old9+(p9*d);
  float ka = olda+(pa*d);
  float kb = oldb+(pb*d);
  float kc = oldc+(pc*d);

  servo1.write(k1);
  servo2.write(k2);
  servo3.write(k3);
  servo4.write(k4);
  servo5.write(k5);
  servo6.write(k6);
  servo7.write(k7);
  servo8.write(k8);
  servo9.write(k9);
  servo10.write(ka);
  servo11.write(kb);
  servo12.write(kc);
  delay(10);
  pos1 = new1;
  pos2 = new2;
```

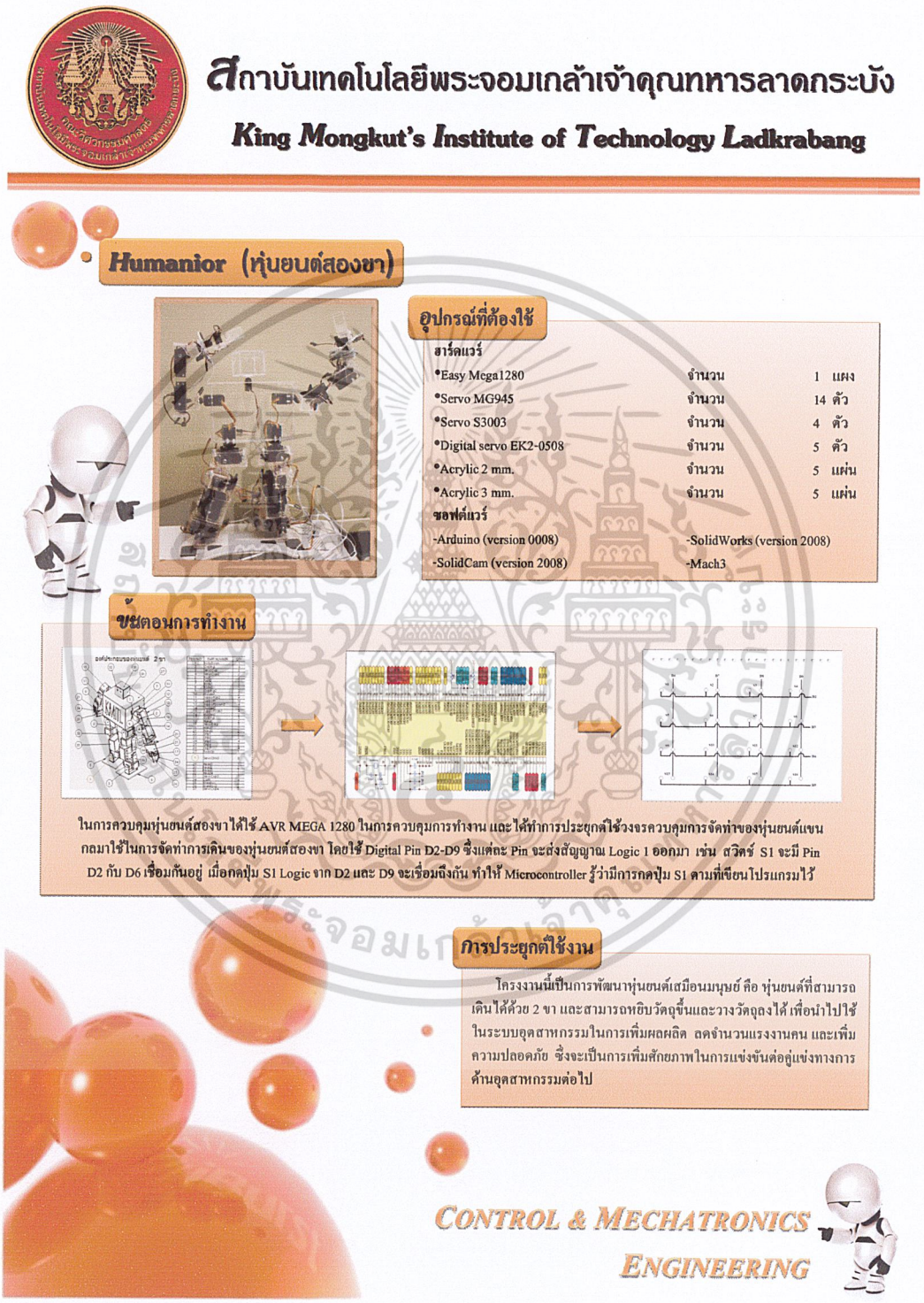
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                pos3 = new3;
                pos4 = new4;
                pos5 = new5;
                pos6 = new6;
                pos7 = new7;
                pos8 = new8;
                pos9 = new9;
                posa = newa;
                posb = newb;
                posc = newc;
            }
        }
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

โปสเตอร์หุ่นยนต์สองขา



รูปที่ ค1 โปสเตอร์หุ่นยนต์สองขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] Mitchell,F.H. system : an introduction to computer-integrated manufacturing/F.H. Mitchell. Englewood Cliffs, NJ: Prentice-Hall International , C 1991

[2] ETT CO.,LTD. "คู่มือการใช้งานบอร์ดไมโครคอนโทรลเลอร์รุ่น ET-EASY MEGA1280 (Duino Mega)." [Online]. Available : www.ett.co.th. 2009.

[3] Schönung, A.; Stemmler, H. "**Pulse-width modulation**." [Online]. Available : http://en.wikipedia.org/wiki/Pulse-width_modulation . 2009

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้