สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

# การผลิตผสมผสานด้วยคอมพิวเตอร์ : ส่วนแขนกล

# COMPUTER INTEGRATED MANUFACTURING : ROBOT ARM SECTION

นางสาวกุลรัตน์ ศิลาวรรณ์
นางสาวธนิสรา บุญยะชัย
นางสาวธัญญาภรณ์ คุณรัตนไตร

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

เลขหมู่.....
เลขทะเบียน 119401
วัน,เดือน,ปี - 7 S.A. 2554

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมแมคคาทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# COMPUTER INTEGRATED MANUFACTURING :

# ROBOT ARM SECTION


**Ms.Kulrat Silawan**

**Ms.Thanissara Boonyachai**

**Ms.Tanyaporn Kulratanatrai**


**THIS THESIS IS SUBMITTED IN PARTIAL 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 : ส่วนแขนกล

Computer Integrated Manufacturing : Robot Arm Section


ผู้จัดทำ นางสาวกุลรัตน์ ศิลาวรรณ์ 50010138

นางสาวธนิสรา บุญยะชัย 50010658

นางสาวธัญญาภรณ์ คุณรัตนไตร 50010676


.....อาจารย์ที่ปรึกษา

(รองศาสตราจารย์ ดร.วันชัย ริ้วรุจา)

.....อาจารย์ที่ปรึกษา

(ดร.วรรณดี เพชรมณีล้ำค่า)

.....อาจารย์ที่ปรึกษา

(อาจารย์ เทพจิตร์ เชยโภคา)


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# การผลิตผสมผสานด้วยคอมพิวเตอร์ : ส่วนแขนกล

โดย
นางสาวกุลรัตน์ ศิลาวรรณ์ 50010138
นางสาวธนิสรา บุญยะชัย 50010658
นางสาวธัญญาภรณ์ คุณรัตนไตร 50010676

อาจารย์ที่ปรึกษา
รองศาสตราจารย์ ดร.วันชัย ริ้วรุจา
ดร.วรรณดี เพชรมณีล้ำค่า
อาจารย์ เทพจิตร์ เชยโภคา

ปีการศึกษา 2553

## บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอเกี่ยวกับแขนกล ซึ่งเป็นส่วนหนึ่งของระบบการผลิตผสมผสานด้วยคอมพิวเตอร์ โดยแขนกลในระบบนี้มีหน้าหลักอยู่ 2 หน้าที่คือ หนึ่งแขนกลจะจับวัตถุจากด้านข้างนำมาใส่บนรถ และสองแขนกลจะทำหน้าที่นำวัตถุออกจากรถมาวางลงที่ถาดด้านข้าง แขนกลจะเป็นการจำลองจากการเคลื่อนที่ของแขนมนุษย์ โดยใช้เซอร์โวมอเตอร์เป็นจุดหมุนที่ข้อต่างๆ

ขั้นตอนการออกแบบและการดำเนินการ เริ่มต้นจากการออกแบบแขนกลโดยโปรแกรมการออกแบบโครงสร้างซึ่งสามารถวาดแบบออกมาในรูปแบบสามมิติ ทำให้สามารถเห็นภาพได้ชัดเจนยิ่งขึ้น อีกทั้งช่วยทดสอบได้ว่าแขนกลที่ออกแบบนั้นสามารถเคลื่อนที่ได้ตามที่ต้องการหรือไม่ โครงสร้างของแขนกลทำมาจากแผ่นอะคริลิคความหนา 2 มิลลิเมตรซึ่งจะทำให้แขนกลมีน้ำหนักที่พอดีกับขนาดของส่วนฐานของแขนกล และจะมีเซอร์โวมอเตอร์ที่แต่ละข้อต่อของแขนกลทุกจุด การควบคุมเซอร์โวมอเตอร์ทำได้โดยการกำหนดพัลส์ของเซอร์โวมอเตอร์โดยการเขียนโปรแกรมควบคุมผ่านบอร์ด Stamp168 ในบอร์ดควบคุมนี้สามารถตั้งค่าให้เซอร์โวมอเตอร์หมุนในทิศทางและตำแหน่งที่ต้องการได้ และบันทึกค่าที่ต้องการนั้นเพื่อให้แขนกลสามารถทำงานโดยอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# COMPUTER INTEGRATED MANUFACTURING : ROBOT ARM SECTION

By

Ms.Kulrat Silawan

Ms.Thanissara Boonyachai

Ms.Tanyaporn Kulratanatrai

Advisors

Assoc.Prof. Dr. Vanchai Riewruja

Dr. Wandee Petchmaneelumka

Mr. Thepjit Cheypoca

Academic Year 2010

## ABSTRACT

This thesis presents about robot arm which is a part of Computer Integrated Manufacturing. There are 2 main functions of this robot arm. One function, it will take a thing from tray beside robot to the machine. Another one, it will take a thing from the machine to the tray beside robot. It is a model of human's movement by using servo motor for rotating.

Procedure of operation starts from design the structure of robot arm by using SolidWorks program. It can aid to view clearly and test the movement of robot. Robot arm's structure is made of acrylic (2 mm) that completely weigh robot. Servomotor control can do by generating the pulse from programming through Stamp 168 board. This control board can set the angle or position of servo motor and record the required value for automatic mode.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# กิตติกรรมประกาศ

ในการจัดทำปริญญานิพนธ์ฉบับนี้ สามารถสำเร็จลุล่วงได้ตรงตามเป้าหมายเนื่องมาจาก การได้รับคำแนะนำและการติดตามการดำเนินโครงงานของอาจารย์ที่ปรึกษาโครงงานทั้งสามท่าน ดังนี้ รศ.ดร. วันชัย ริ้วรุจา, ดร.วรรณดี เพชรมณีล้ำค่า และอ.เทพจิตร์ เชยโภคา ซึ่งอาจารย์ทั้งสามได้ชี้แนะแนวทางการดำเนินการต่างๆ แนวคิด การวางแผนงาน อีกทั้งช่วยจัดหาอุปกรณ์และแนะนำการใช้งาน ทำให้สะดวกในการทำโครงงานนี้เป็นอย่างมาก ทางคณะผู้จัดทำจึงขอบขอบพระคุณเป็นอย่างสูง

ขอบคุณเพื่อนๆ ทั้งที่เป็นกลุ่มโครงงานเดียวกันและเพื่อนคนอื่นๆ ที่ให้ความช่วยเหลือทำให้โครงงานนี้สามารถสำเร็จลุล่วงได้ตามเป้าหมาย

ผู้จัดทำ

นางสาวกุลรัตน์ ศิลาวรรณ์

นางสาวธนิสรา บุญยะชัย

นางสาวธัญญาภรณ์ คุณรัตนไตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ


| | หน้า |
|---|---|
| **บทคัดย่อ** | I |
| **บทคัดย่อภาษาอังกฤษ** | II |
| **กิตติกรรมประกาศ** | III |
| **สารบัญ** | IV |
| **สารบัญภาพ** | VI |
| **สารบัญตาราง** | VIII |
| **บทที่ 1 บทนำ** | 1 |
| 1.1 กล่าวนำ | 1 |
| 1.2 วัตถุประสงค์ในการทำปริญญานิพนธ์ | 2 |
| 1.3 ขอบเขตและขั้นตอนการศึกษาการจัดทำโครงงาน | 2 |
| 1.4 รายละเอียดของปริญญานิพนธ์ | 2 |
| **บทที่ 2 Computer Integrated Manufacturing (CIM)** | 3 |
| 2.1 ส่วนประกอบภายในระบบจำลองกระบวณการการผลิต | 4 |
| 2.1.1 ระบบลำเลียง | 4 |
| 2.1.1.1 รถลำเลียงสินค้า | 4 |
| 2.1.1.2 สถานีรับ-ส่งสินค้า | 4 |
| 2.1.2 แขนกล | 5 |
| 2.1.2.1 ส่วนควบคุม | 5 |
| 2.1.2.2 ส่วนประมวลผล | 5 |
| 2.1.2.3 ส่วนเซอร์โวมอเตอร์ | 5 |
| 2.1.2.4 ส่วนที่ประกอบเป็นแขนกล | 5 |
| 2.1.3 ส่วนแสดงผล | 6 |
| 2.2 หลักการทำงานของระบบ | 7 |
| 2.2.1โหมดควบคุมด้วยมือ | 7 |
| 2.2.2 โหมดควบคุมอัตโนมัติ | 7 |
| 2.3 การติดต่อสื่อสารกันของระบบ | 8 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)


| | หน้า |
|---|---|
| **บทที่ 3 แขนกล (Robot Arm)** | 9 |
| 3.1 หลักการทำงานของแขนกล | 10 |
| 3.2 โครงสร้างทางกายภาพของแขนกล | 12 |
| 3.2.1 การออกแบบโครงสร้างโดยใช้โปรแกรม SolidWorks | 12 |
| 3.2.1.1 ส่วนฐานของแขนกล | 13 |
| 3.2.1.2 ฐานหมุนของแขนกล | 13 |
| 3.2.1.3 แขนส่วนที่ (1) | 15 |
| 3.2.1.4 แขนส่วนที่ (2) | 15 |
| 3.2.1.5แขนส่วนที่ (3) | 16 |
| 3.2.1.6 ตัวหนีบจับ | 17 |
| 3.2.1.7 โครงสร้างโดยรวมของแขนกล | 20 |
| 3.2.2 อุปกรณ์ที่ใช้ต่อแขนกล1แขน | 21 |
| 3.2.3 การประกอบโครงสร้างของแขนกล | 25 |
| 3.3 การควบคุมทิศทางการหมุนของเซอร์โวมอเตอร์ | 26 |
| 3.3.1 เซอร์โวมอเตอร์ | 26 |
| 3.3.2 หลักการทำงานของเซอร์โวมอเตอร์ | 27 |
| 3.3.3 การทำงานของเซอร์โวมอเตอร์แต่ละส่วน | 28 |
| 3.4 วงจรอิเลกทรอนิกส์ควบคุมการบังคับแขนกล | 32 |
| 3.5 บอร์ดควบคุมการทำงาน | 34 |
| 3.6 ส่วนโปรแกรมการควบคุมการทำงานของแขนกล | 41 |
| 3.6.1การใช้งานบอร์ด Arduino ในการควบคุมการทำงาน | 41 |
| 3.6.2 ส่วนประกอบของไมโครคอนโทรเลอร์ | 43 |
| 3.6.3ไมโครคอนโทรลเลอร์ตระกูล AVR | 43 |
| **บทที่ 4 สรุปผลการทดลอง** | 45 |
| **บทที่ 5 บทวิจารณ์และสรุป** | 47 |
| **ภาคผนวก ก โปรแกรมควบคุมแขนกล** | 50 |
| **ภาคผนวก ข โปสเตอร์แขนกล** | 71 |
| **เอกสารอ้างอิง** | 72 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญภาพ


| รูปที่ | หน้า |
|---|---|
| 1.1 โครงสร้างของแขนกล โดยโปรแกรม SolidWorks | 1 |
| 2.1 รูปภาพแสดงรถลำเลียงสินค้า | 4 |
| 2.2 รูปภาพแสดงสถานีรับ-ส่งสินค้า | 4 |
| 2.3 รูปภาพแสดงแขนกล | 5 |
| 2.4 รูปภาพแสดงส่วนแสดงผล | 6 |
| 2.5 รูปภาพแสดงระบบรวม | 6 |
| 3.1 ส่วนประกอบของแขนกล | 10 |
| 3.2 ไดอะแกรมควบคุมการทำงานของแขนกล | 11 |
| 3.3 แผนผังไดอะแกรมแสดงภาพรวมการทำงานของแขนกล | 12 |
| 3.4 ส่วนประกอบฐานของแขนกล | 13 |
| 3.5 ส่วนประกอบของฐานหมุน | 14 |
| 3.6 ส่วนประกอบของแขนส่วนที่ (1) | 15 |
| 3.7 ส่วนประกอบของแขนส่วนที่ (2) | 15 |
| 3.8 ส่วนประกอบของแขนส่วนที่ (3) | 16 |
| 3.9 ส่วนประกอบของตัวหนีบจับ (1) | 17 |
| 3.10 ส่วนประกอบของตัวหนีบจับ (2) | 18 |
| 3.11 ส่วนประกอบของตัวหนีบจับ (3) | 19 |
| 3.12 ส่วนประกอบของตัวหนีบจับ (4) | 20 |
| 3.13 โครงสร้างโดยรวมของแขนกล | 21 |
| 3.14 Servoรุ่น MG 945 | 22 |
| 3.15 Servoรุ่น S3003 | 23 |
| 3.16 สัดส่วนของ Servoรุ่น S3003 | 23 |
| 3.17 Digital servo motor EK 2-0508 | 25 |
| 3.18 โครงสร้างแขนกลที่สมบูรณ์ | 26 |
| 3.19 โครงสร้างของเซอร์โวมอเตอร์ | 27 |
| 3.20 จุดอ้างอิงความกว้างของสัญญาณพัลส์ | 27 |
| 3.21 เซอร์โวมอเตอร์ส่วนคลิ๊ปเปอร์และส่วนควบคุมการหมุนของมือจับ | 29 |
| 3.22 เซอร์โวมอเตอร์ส่วนควบคุมการยกมือจับขึ้น-ลง | 29 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญภาพ (ต่อ)


| รูปที่ | หน้า |
|---|---|
| 3.23 เซอร์โวมอเตอร์ส่วนควบคุมแขนให้ยกสิ่งของขึ้น-ลง | 30 |
| 3.24 เซอร์โวมอเตอร์ส่วนควบคุมแขนท่อนล่าง | 30 |
| 3.25 เซอร์โวมอเตอร์ส่วนฐาน | 31 |
| 3.26 ตำแหน่งเซอร์โวมอเตอร์ในแต่ละข้อต่อ | 31 |
| 3.27 สวิตช์แบบเมตริกซ์ | 32 |
| 3.28 วงจรสวิตซ์ควบคุม (Keypad Joystick) | 33 |
| 3.29 บอร์ดควบคุมการเคลื่อนไหวของหุ่นยนต์ | 33 |
| 3.30 โครงสร้างของบอร์ด ET-EASY168 STAMP | 34 |
| 3.31 ลักษณะของบอร์ด ET-EASY168 STAMP | 36 |
| 3.32 ขาสัญญาณ | 37 |
| 3.33 หน้าจอโปรแกรม arduino | 42 |
| 3.34 การเขียนโปรแกรมโดยใช้ภาษาซี | 42 |
| 3.35 การ Upload โปรแกรมลงบอร์ด Arduino โดยกดปุ่ม Upload to I/O Board | 43 |
| 3.36 ตัวอย่างการต่อ บอร์ด I/O ของ อีทีที ด้วยขั้ว 10Pin | 44 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง


| ตารางที่ | หน้า |
|---|---|
| 3.1 การจัดสรรขาสัญญาณของบอร์ด ET-EASY168 STAMP | 37 |
| 3.2 หน้าที่ของขาสัญญาณ | 38 |
| 4.1 ตารางการทดลองของ Servo Motor เรื่องความกว้างของพัลส์ | 45 |





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1
# บทนำ

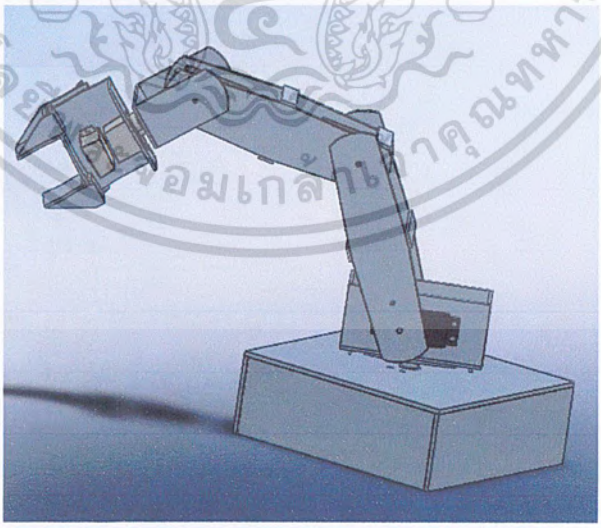
## 1.1 กล่าวนำ

ในยุคปัจจุบันนี้เป็นยุคที่เทคโนโลยีมีการพัฒนาอย่างรวดเร็ว โดยเฉพาะในระบบอุตสาหกรรมขนาดใหญ่ที่จะพยายามหาเทคโนโลยีในการผลิตในขั้นตอนต่างๆ เพื่อเป็นการลดเวลาการทำงาน ลดต้นทุนในระยะยาว ซึ่งสามารถทำได้โดยการวางระบบให้เหมาะสมและระบบนั้นต้องเป็นระบบที่เกิดความผิดพลาดน้อยจึงเกิดการพัฒนาโดยการใช้เครื่องจักรกลแทนแรงงานมนุษย์ เนื่องจากเครื่องจักรกลสามารถทำงานได้ในระยะเวลาที่นาน เกิดความผิดพลาดน้อย อีกทั้งช่วยประหยัดต้นทุนในระยะยาวได้ด้วย จึงเกิดการพัฒนาระบบการผลิตที่เรียกว่า การผลิตผสมผสานด้วยคอมพิวเตอร์ (Computer Integrated Manufacturing หรือ CIM) ในระบบการผลิตนี้จะนำคอมพิวเตอร์มาช่วยควบคุมและตรวจสอบในแต่ละขั้นตอนของการผลิต

ในระบบการผลิตผสมผสานด้วยคอมพิวเตอร์ที่จำลองขึ้นมานี้ จะประกอบด้วยส่วนต่างๆดังนี้

1. ส่วนควบคุมและส่วนแสดงผล
2. ส่วนของสายการลำเลียง

ในส่วนของสายการลำเลียงจะประกอบด้วย แขนกล และรถรางอยู่ในส่วนนี้ ซึ่งในปริญญานิพนธ์ฉบับนี้จะกล่าวถึงแขนกล โครงสร้าง หลักการออกแบบ การควบคุม และหลักการทำงานของแขนกล



**รูปที่ 1.1** โครงสร้างของแขนกล โดยโปรแกรม SolidWorks

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.2 วัตถุประสงค์ในการทำปริญญานิพนธ์

1. เพื่อศึกษาเกี่ยวกับระบบผสมผสานด้วยคอมพิวเตอร์
2. เพื่อศึกษาด้านการออกแบบโครงสร้างแขนกลให้ทำงานได้อย่างมีประสิทธิภาพ
3. เพื่อศึกษาการเขียนโปรแกรมควบคุมแขนกล โดยเขียนโปรแกรมควบคุมที่เซอร์โวมอเตอร์แต่ละตัวตามจุดข้อต่อของแขนกล
4. สามารถนำไปพัฒนาต่อยอดในเชิงพาณิชย์ได้

## 1.3 ขอบเขตและขั้นตอนการศึกษาการจัดทำโครงงาน

การออกแบบโครงสร้างของแขนกล ต้องศึกษาการเคลื่อที่ของแขน จุดหมุนต่างๆ ที่จะนำมาออกแบบให้แขนกลสามารถเคลื่อนที่ได้อย่างอิสระ โดยเบื้องต้นต้องออกแบบโดยใช้โปรแกรมทางคอมพิวเตอร์เพื่อให้สามารถดูได้ว่าที่ออกแบบไว้นั้น แขนกลจะสามารถทำงานได้หรือไม่ ถัดมาเมื่อออกแบบโดยอาศัยโปรแกรมแล้ว จะทำการกัดชิ้นงานออกมาประกอบเป็นแขนกลขึ้นมา โดยจะมีการปรับปรุงแก้ไขลักษณะทางโครงสร้างเพื่อให้แข็งแรงใช้งานได้จริง จากนั้นจึงศึกษาเกี่ยวกับโปรแกรมและบอร์ดที่จะใช้ควบคุมการเคลื่อนไหวของแขนกล เมื่อทดลองใส่โปรแกรมให้แขนกลทำงานก็จะสังเกตการเคลื่อนไหวของแขนกลในแต่ละส่วนว่าสามารถทำงานได้สมบูรณ์หรือไม่ เกิดการขัดข้องในส่วนไหน แล้วจึงปรับแก้ในส่วนนั้น ถ้ามีการขัดข้องทางโครงสร้างก็จะปรับแก้ทำให้โครงสร้างแข็งแรงขึ้นในส่วนนั้น หรือถ้าหากแขนกลเคลื่อนที่ได้ไม่สมบูรณ์โดยที่เหตุเกิดมาจากการตั้งค่าทางโปรแกรมทำให้เซอร์โวมอเตอร์ทำงานไม่ราบเรียบ จึงต้องมีการปรับแก้ทางด้านโปรแกรมต่อไป จนกว่าแขนกลจะสามารถทำงานได้จริงและสมบูรณ์

## 1.4 รายละเอียดของปริญญานิพนธ์

เนื้อหาที่จะกล่าวในปริญญานิพนธ์ฉบับนี้ประกอบด้วย

บทที่ 1 บทนำ กล่าวนำถึงวัตถุประสงค์ หลักการ ขั้นตอนการศึกษา และการจัดทำโครงงาน รวมทั้งรายละเอียดของปริญญานิพนธ์ในแต่ละบท

บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง กล่าวถึงหลักการและทฤษฎีที่เกี่ยวข้องในปริญญานิพนธ์นี้

บทที่ 3 หลักการออกแบบ การประกอบโครงสร้างของระบบ รวมถึงแนวคิดในการออกแบบ

บทที่ 4 การทดลอง เป็นส่วนการทดสอบองค์ประกอบต่างๆ ในระบบ

บทที่ 5 บทวิจารณ์และสรุป จะสรุปผลการดำเนินงาน ปัญหาที่เกิดขึ้น และแนวทางการปรับปรุงพัฒนาโครงงานนี้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 2

# Computer Integrated Manufacturing (CIM)

# การผลิตผสมผสานด้วยคอมพิวเตอร์

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

CIM เป็นทั้งกระบวนการผลิตและชื่อของระบบอัตโนมัติที่ควบคุมด้วยคอมพิวเตอร์ โดยมีหน้าที่สนับสนุนการทำงานและเป็นระบบการจัดการของระบบการผลิตที่ประกอบด้วยฝ่ายต่างๆ เช่น ฝ่ายงานวิศวกรรม ฝ่ายงานการผลิต ฝ่ายการตลาด และฝ่ายการสนับสนุนอื่นๆ ขอบข่ายหน้าที่การทำงานของ CIM มีหลากหลายอย่าง เช่น ออกแบบ วิเคราะห์ วางแผน จัดซื้อ จัดการบัญชีต้นทุน ควบคุมคงคลัง และการกระจายผลิตภัณฑ์ เหล่านี้จะถูกเชื่อมโยงโดยคอมพิวเตอร์ไปยังอุปกรณ์หรือหน่วยต่างๆ ภายในองค์กร CIM จะทำให้สามารถควบคุมกระบวนการได้โดยตรงและสามารถแสดงการทำงานปัจจุบันของทุกกระบวนการทำงาน ข้อแตกต่าง 3 ประการที่ทำให้ CIM มีความแตกต่างจาก ระบบการผลิตรูปแบบอื่นๆ นั่นคือ

1. สื่อที่ใช้ในการเก็บข้อมูล การคืนสภาพกระบวนการ การควบคุม และการนำเสนอ
2. กลไกการตรวจจับและการดัดแปลงกระบวนการต่างๆ
3. อัลกอริทึมของการประมวลผลข้อมูล ที่ได้จากการตรวจจับและการดัดแปลงส่วนประกอบต่างๆ

CIM จะประกอบด้วยคอมพิวเตอร์ที่มีการแลกเปลี่ยนข้อมูลระหว่างกัน 2 เครื่องขึ้นไป เช่น ตัวควบคุมหุ่นยนต์อุตสาหกรรม (Industrial Robot) กับ ตัวคุมเครื่อง CNC สิ่งที่ต้องคำนึงถึงในการติดตั้งระบบ CIM คือ ปริมาณการผลิต ประสบการณ์ขององค์การและบุคลากร ระดับการผสมผสานของส่วนการผลิตและส่วนต่างๆ CIM มีประโยชน์มากที่สุดในองค์การที่มีระดับการใช้ข้อมูลสารสนเทศภายในองค์กรสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.1 ส่วนประกอบภายในระบบจำลองกระบวนการการผลิต

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

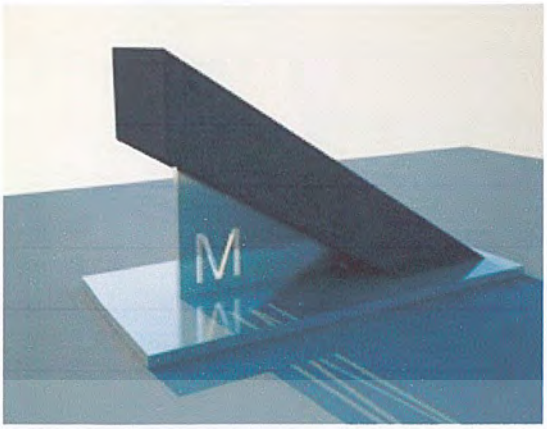
**2.1.1 ระบบลำเลียง** ประกอบไปด้วยรถลำเลียงสินค้าและสถานีรับ-ส่งสินค้า

2.1.1.1 **รถลำเลียงสินค้า** ทำหน้าที่รับส่งสินค้าจากสถานีหนึ่งไปยังสถานีหนึ่ง ซึ่งใช้หลักการของการส่งข้อมูลผ่านทางสัญญาณ infrared โดยรถจะเคลื่อนที่ไปตามรางไฟฟ้า และเมื่อตัวรถนั้นเคลื่อนที่มาจอดที่สถานี จากนั้นวงจรรับ-ส่งสัญญาณจะเริ่มทำงาน โดยตัวรถจะส่งสัญญาณไปยังสถานี ส่วนที่ทำหน้าที่รับสัญญาณที่ส่งไปก็คือ สถานี

**รูปที่ 2.1** รถลำเลียงสินค้า

2.1.1.2 **สถานีรับ-ส่งสินค้า** ทำหน้าที่รับข้อมูลทางตัวควบคุมหลัก (main controller) และตรวจเงื่อนไขของรถรับ-ส่งสินค้า ซึ่งใช้หลักการของการส่งข้อมูลผ่านทางสัญญาณอินฟาเรด (infrared) เช่นเดียวกัน โดยจะเริ่มทำงานเมื่อรถเคลื่อนที่เข้ามาจอดที่สถานี และส่วนสถานีได้รับสัญญาณอินฟาเรด ก็จะทำส่งมายังตัวควบคุมหลัก ให้ทำการประมวลผลและตัดสินใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 2.2** สถานีรับ-ส่งสินค้า

**2.1.2 แขนกล** ทำหน้าที่เคลื่อนย้ายสินค้าระหว่างรถลำเลียงกับสถานีรับ-ส่งสินค้า ซึ่งแขนกลมีส่วนประกอบ 4 ส่วนหลักได้แก่

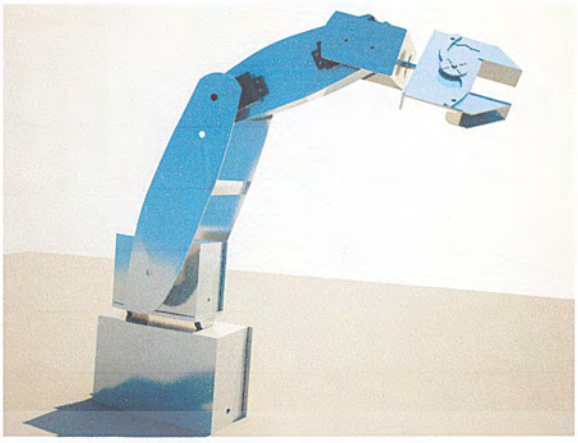
2.1.2.1 ส่วนควบคุม แขนกลจะถูกควบคุมการทำงานทั้งหมดจากบอร์ดสวิตซ์ควบคุม (Keypad Joystick) ซึ่งเป็นเมทริกซ์สวิตท์ 4x4 มีทั้งหมด 16 สวิตท์ เซอร์โวหนึ่งตัวจะใช้สวิตท์ควบคุมทิศทางการหมุนจำนวน 2 ตัว โดยควบคุมให้เซอร์โวมอเตอร์หมุนได้ในทิศทางตามเข็มนาฬิกาและทวนเข็มนาฬิกา ดังนั้นจึงใช้สวิตท์ควบคุมเซอร์โวทั้งหมด 12 ตัว และจะมีสวิตท์ที่ใช้ทำงานอีก 2 ตัว ด้วยกัน คือ หนึ่งสวิตท์ที่ทำหน้าที่เป็นคำสั่งให้จำค่าการหมุนของเซอร์โวมอเตอร์ที่สั่งไว้ สองคือสวิตท์ที่ทำหน้าที่สั่งให้แขนกลทำงานโดยอัตโนมัติคือคำสั่ง RUN

2.1.2.2 ส่วนประมวลผล ทำหน้าที่ในการประมวลผลคำสั่งที่ได้จากส่วนควบคุมจากบอร์ดแสตมป์ 168(STAMP 168) แล้วทำการประมวลผลค่าที่ได้มา จากนั้นทำการส่งข้อมูลที่ได้ไปยังเซอร์โวมอเตอร์ของแขนกลให้ทำงานสอดคล้องตามที่ระบบได้กำหนดมา

2.1.2.3 ส่วนเซอร์โวมอเตอร์ เป็นส่วนที่ขับเคลื่อนแขนกลให้เคลื่อนไหวไปยังตำแหน่งที่ต้องการ ซึ่งเซอร์โวมอเตอร์จะถูกติดตั้งไว้ในส่วนของข้อต่อต่างๆของแขนกลจำนวน 6 จุด ตั้งแต่ฐานของแขนกล ส่วนแขน และคลิ๊ปเปอร์ ซึ่งจะทำให้แขนกลสามารถเคลื่อนที่ได้ในองศาที่อิสระ

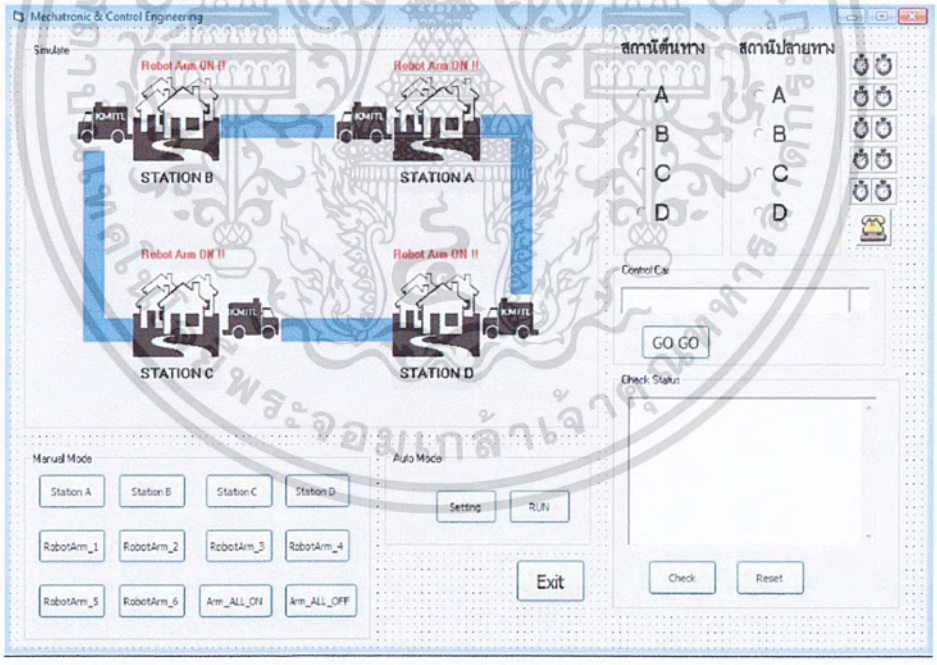
2.1.2.4 ส่วนที่ประกอบเป็นแขนกล จะแบ่งออกเป็น 3 ส่วนด้วยกัน คือ ฐาน แขน และคลิ๊ปเปอร์ ซึ่งชิ้นส่วนทุกชิ้นทำมาจากแผ่นอะคริลิคความหนา 2 มิลลิเมตร และได้ทำการออกแบบจากโปรแกรม SolidWorks Version2010 จากนั้นจึงนำไปแปลงค่าให้กัดชิ้นงานเป็นค่าคำสั่ง G-Code ออกมา แล้วนำโค้ดโปรแกรมที่เป็นคำสั่ง G-Code ที่ได้นำไปกัดชิ้นงานผ่านเครื่องซีเอ็นซี โดยใช้โปรแกรม Mach3 และนำชิ้นงานที่ได้นำไปประกอบเข้าด้วยกันเริ่มตั้งแต่ฐานของแขนกล ส่วนแขน ไปจนถึงส่วนของคลิปเปอร์ ถัดมาจึงประกอบเข้ากับเซอร์โวมอเตอร์ในแต่ละจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ดัดแปลง เผยแพร่ หรือแสวงหาประโยชน์ทางการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 2.3** รูปภาพแสดงแขนกล

**2.1.3 ส่วนแสดงผล** ทำหน้าที่แสดงผลการทำงานของระบบ เพื่อให้เห็นภาพการทำงานของแขนกลและการเคลื่อนที่ของรถขนส่งสินค้าในแต่ละสถานี โดยการทำงานมีทั้งระบบควบคุมอัตโนมัติและระบบควบคุมด้วยมือ




**รูปที่ 2.4** ส่วนแสดงผล

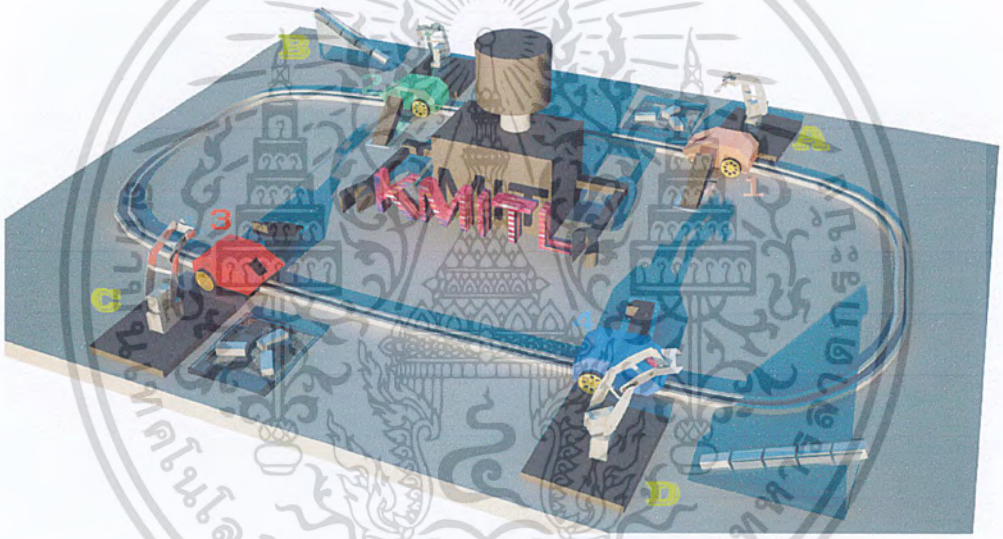
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2 หลักการทำงานของระบบ

จะมีด้วยกัน 2 โหมด คือ

### 2.2.1 โหมดควบคุมด้วยมือ (Manual mode)

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




**รูปที่ 2.5** ระบบการทำงานรวมของโครงงาน

ถ้าเรามองจากรูปที่ 2.5 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

### 2.2.2 โหมดควบคุมอัตโนมัติ (Automatic mode)

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 การติดต่อสื่อสารกันของระบบ

ในการติดต่อกันระหว่างรถกับสถานีเราใช้หลักการของวงจรอินฟาเรด ตัวส่ง คือ หลอดอินฟาเรด และ ตัวรับ คือ โมดูลรับสำเร็จรูป(3ขา) จะส่งด้วยความถี่ 40KHz โดยประมาณ ประโยชน์เพื่อเป็นความถี่หลักในการตรวจรับว่าเป็นสัญญาณตัวจริงไม่ใช่สัญญาณรบกวน ตัวรับแบบโมดูล (3ขา) โมดูลจะรับสัญญาณที่กระพริบด้วยความถี่ประมาณ 40KHz ถ้าตรงก็จะให้เอาท์พุทที่ขาเอาท์พุทเป็น 0 หลักการของมันก็มีแค่ส่งแสงอินฟราเรดไปยังวัตถุที่ต้องการตรวจจับ ถ้าตรวจพบวัตถุนั้นก็จะสะท้อนแสงกลับมายังตัวรับ โดยจะติดตั้ง ตัวรับและส่ง สัญญาณนี้ไว้ที่ รถ กับ สถานีอย่างละ1ชุด

ส่วนการติดต่อกันระหว่างตัวควบคุมหลัก กับ สถานี นั้นเราจะใช้สายแพในการรับ-ส่งข้อมูล โดยในส่วนของตัวควบคุมหลัก จะทำหน้าที่เป็นตัวส่งข้อมูลไปยังสถานี เพื่อสั่งการว่าให้รถที่มาจอดนั้นเคลื่อนที่ไปสถานีใดต่อไป

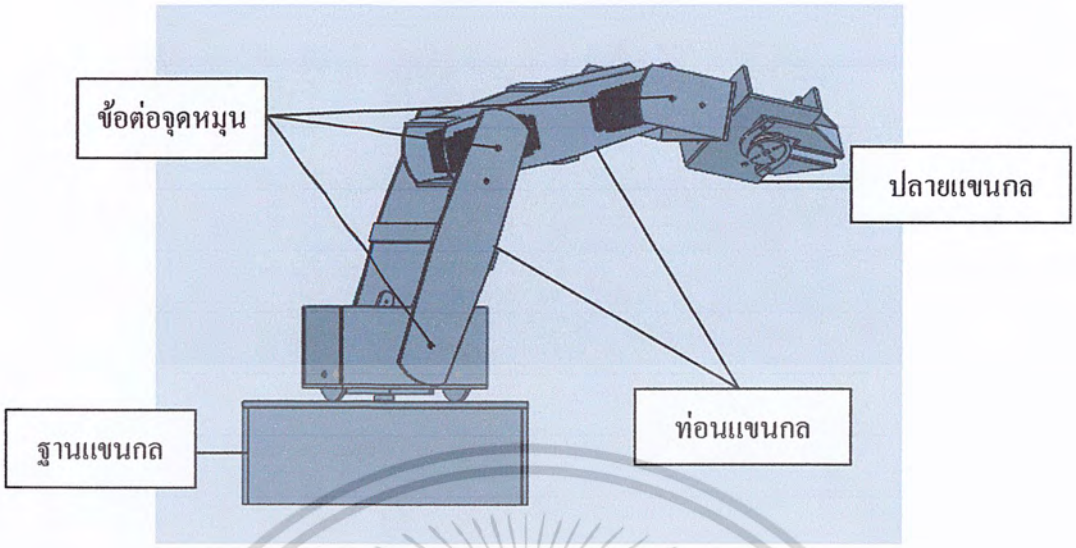
ในส่วนการติดต่อกันระหว่างตัวควบคุมหลัก กับ คอมพิวเตอร์จะใช้สาย USB ในการเชื่อมต่อ เพื่อที่จะส่งข้อมูลจากคอมพิวเตอร์ไปยังตัวควบคุมหลัก ซึ่ง ตัวควบคุมหลัก จะได้สั่งงานไปยังสถานี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3
# แขนกล

แขนกล( Robot Arm )เป็นหุ่นยนต์ชนิดหนึ่งที่นำมาใช้งานในวงการอุตสาหกรรมการผลิต ได้ถูกนำมาใช้แทนแรงงานมนุษย์ในงานที่ต้องทำอย่างต่อเนื่องตลอด 24 ชั่วโมง, งานที่ต้องทำซ้ำ ๆ กันตลอดเวลา,งานที่เป็นอันตราย, งานที่หนักและยากเกินที่มนุษย์จะทำไหว ปกติมนุษย์ก็สามารถทำงานได้ทุกอย่างแต่ข้อจำกัดของมนุษย์นั้นไม่สามารถทำงานได้อย่างต่อเนื่องยาวนานจะเกิดความเหน็ดเหนื่อยเมื่อยล้าจึงต้องมีการพักผ่อน เมื่อคนทำงานในที่อันตรายเช่นงานที่เกี่ยวกับสารเคมีที่มีพิษ ถ้าป้องกันไม่ดีก็จะมีผลต่อสุขภาพได้ เมื่อเป็นข้อจำกัดอย่างนี้หุ่นยนต์ก็จะเข้ามามีบทบาทในการทำงานดังกล่าว และข้อดีของการที่มีหุ่นยนต์ทำงานแทนคนนั้นนอกจากที่กล่าวไว้ข้างต้นแล้ว ประสิทธิภาพการทำงานก็จะดีขึ้น, มีความแน่นอน แม่นยำ, สามารถทำงานผลิตได้โดยไม่ต้องพัก, จำนวนชิ้นงานที่ทำก็มากขึ้น, ทำงานได้โดยไม่มีวันหยุด ส่วนข้อเสียก็มี เช่นมีราคาสูง ต้องมีผู้ชำนาญการในการควบคุมหุ่นยนต์ ไม่เหมาะในโรงงานที่กำลังผลิตน้อย แขนกลอุตสาหกรรมที่เราสามารถพบเห็นได้โดยทั่วไป เช่น ในโรงงานผลิตและประกอบรถยนต์, งานเชื่อมอุตสาหกรรม, งานประกอบเครื่องจักร, งานในโรงงานผลิตเหล็ก และงานเกี่ยวกับคลังสินค้าขนาดใหญ่ และอื่นๆ อีกมากมาย ซึ่งล้วนแต่เป็นลักษณะอุตสาหกรรมขนาดใหญ่ แขนกลอุตสาหกรรมนั้นมีส่วนประกอบอยู่หลายส่วนได้แก่ ฐาน (Base) ของหุ่นยนต์, ท่อนชิ้นส่วนที่เป็นแขนกล, ข้อต่อจุดหมุน (Joints) ตามชิ้นส่วนที่ต่อกัน, ปลายของแขนกลที่ใช้ทำงานยกตัวอย่างเช่นมือคีบจับ, หัวเชื่อม, อุปกรณ์ประกอบชิ้นส่วน, ปืนพ่นสี, หัวเจาะ ฯลฯ คอมพิวเตอร์ที่มาควบคุมแขนกลนั้นจะทำหน้าที่ควบคุมในส่วนที่เป็นมอเตอร์แบบสเต็บ (Step motors: เป็นมอเตอร์ที่มาจากมอเตอร์โดยทั่วไป กล่าวคือมอเตอร์แบบสเต็บนั้นมีความสามารถหมุน และหยุดได้ตามความต้องการ ตามระยะที่ได้ตั้งโปรแกรมไว้ และสามารถทำซ้ำ ๆ กันได้ในการเคลื่อนที่ ส่วนมอเตอร์โดยทั่วไปเมื่อป้อนพลังงานก็จะหมุนตลอด และเวลาหยุดจะหมุนฟรีไปหลายรอบซึ่งเป็นผลมาจากแรงเฉื่อย) มอเตอร์แบบสเต็ปจึงทำให้หุ่นยนต์ได้เคลื่อนไหวได้ตามโปรแกรมที่ได้ตั้งไว้ นอกจากมอเตอร์แบบสเต็บแล้ว แขนกลที่มีขนาดใหญ่ที่นำมาใช้ในงานหนักอาจจะใช้มอเตอร์ไฮดรอลิกส์ หรือมอเตอร์ลมนิวแมติกส์ แทนก็ได้ แขนกลจะมีระบบเซนเซอร์ไว้คอยตรวจจับการทำงานเพื่อให้หุ่นยนต์นั้นได้มีการเคลื่อนที่ได้อย่างถูกต้อง เกิดความแน่นอนในการเคลื่อนที่ของแขนกลมากยิ่งขึ้น ซึ่งทำให้สามารถกำหนดการเคลื่อนไหวของแขนกลได้เป็นอย่างดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่3.1** ส่วนประกอบของแขนกล

แขนกลอุตสาหกรรมโดยทั่วไปจะมีข้อต่อ 6 ข้อต่อ โดยคล้ายกับแขนของมนุษย์ที่เริ่มนับจากหัวไหล่ ข้อศอก และมือ ในหุ่นยนต์จะมีฐานหุ่นคล้ายบ่าเพื่อรองรับโครงสร้างที่มีกรเคลื่อนที่ เราเรียกข้อต่อจุดหมุนว่าเป็นองศาอิสระ (Degrees Of Freedom: DOF) หมายถึงมันสามารถที่จะเคลื่อนไหวได้อย่างอิสระภายใต้ระยะจุดหมุนที่หมุนได้ ถ้าเปรียบเทียบกับแขนมนุษย์ที่สามารถยกแขนให้เคลื่อนที่จากตำแหน่งไปสู่ตำแหน่งหนึ่ง แขนกลก็เหมือนกันแขนกลก็สามารถทำการเคลื่อนที่ได้จากจุดหนึ่งไปสู่จุดหนึ่งในระยะขอบเขตรัศมีการเคลื่อนที่ ในการรับน้ำหนักของแขนกลก็จะมีเซ็นเซอร์วัดความดันบอกสถานะน้ำหนักที่รับได้ว่าเกินกำลังลังของหุ่นหรือไม่เมื่อน้ำหนักที่ทำงานเกินเครื่องก็จะเตือน และแขนกลก็จะไม่ทำงาน

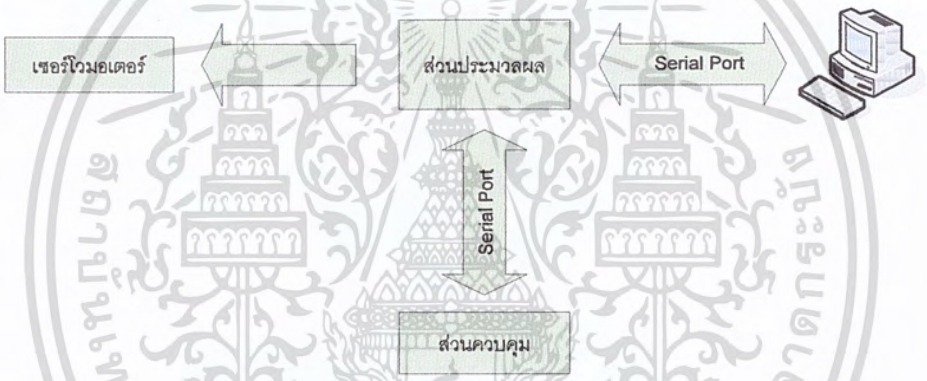
## 3.1 หลักการทำงานของแขนกล

เนื่องจากการทำงานของแขนกลเข้ามามีบทบาทในการทำงานอย่างแพร่หลายของภาคอุตสาหกรรมในปัจจุบันมากขึ้น โดยนำมาใช้แทนแรงงานคนในงานที่มีข้อจำกัด ทางกลุ่มจึงต้องการประดิษฐ์แขนกลที่มีความละเอียดแม่นยำในการหยิบจับ และการควบคุมทางอิเล็กทรอนิกส์ที่มีความเสถียรมากขึ้น โดยในเบื้องต้นต้องทำความเข้าใจถึงหลักการทำงานในภาพรวมของแขนกลก่อนว่ามีหลักการทำงานอย่างไร หลังจากนั้นแล้วจึงพิจารณาส่วนย่อยต่างๆอันประกอบด้วย ส่วนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นโครงสร้างของแขนกล วงจรอิเล็กทรอนิกส์ ระบบควบคุม และส่วนของโปรแกรมที่ใช้ประมวลผลและควบคุมระบบ

แนวคิดการออกแบบการทำงานของแขนกลเพื่อให้แขนกลสามารถเคลื่อนที่ไปมาและหยิบจับวัตถุได้ตามที่ต้องการโดยนำคอมพิวเตอร์ส่วนบุคคลเข้ามาใช้ในการควบคุมการเคลื่อนที่ของแขนกลซึ่งระบบการทำงานของแขนกลในการจับวัตถุ เริ่มจากการส่งข้อมูลเข้าคอมพิวเตอร์ แล้วทำการประมวลผล แล้วส่งข้อมูลที่ได้จากการประมวลผลไปเข้าไมโครโปรเซสเซอร์ทำหน้าที่ประมวลผลอีกครั้งก่อนส่งข้อมูลไปควบคุมการทำงานของแขนกล จึงสามารถควบคุมการทำงานของแขนกลได้ตามต้องการหลักการควบคุมการทำงานของแขนกลนั้น สามารถเขียนเป็นไดอะแกรมได้ดังรูปที่ 3.2




**รูปที่ 3.2** ไดอะแกรมควบคุมการทำงานของแขนกล

จากรูปที่3.2 จะเห็นว่าภาพรวมการทำงานของแขนกลประกอบด้วย 3 ส่วนหลักดังนี้

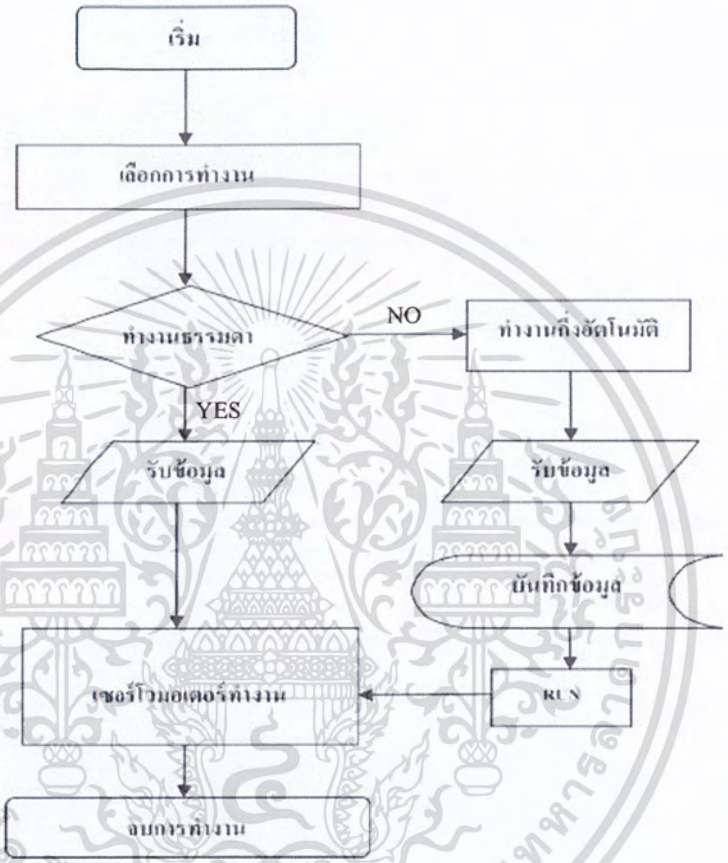
- ส่วนควบคุม
- ส่วนเซอร์โวมอเตอร์
- ส่วนประมวลผล

การออกแบบการควบคุมการทำงานของแขนกลออกแบบให้มีหลักการทำงานอยู่ 2 โหมด คือ

1.โหมดควบคุมด้วยมือ โดยใช้บอร์ดควบคุมการเคลื่อนไหวของแขนกล โดยสามารถสั่งโดยการกดสวิตท์คำสั่ง RUN ในบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.โหมดควบคุมอัตโนมัติ โดยทำการควบคุมการทำงานผ่านโปรแกรมArduinoแล้วเชื่อมต่อไปสู่ไมโครโปรเซสเซอร์ เพื่อไปควบคุมการทำงานของแขนกลต่อไป ซึ่งหลักการในภาพรวมการทำงานของแขนกล สามารถแสดงแผนผังไดอะแกรมดังรูปที่ 3.3




รูปที่ 3.3 แผนผังไดอะแกรมแสดงภาพรวมการทำงานของแขนกล

## 3.2 โครงสร้างทางกายภาพของแขนกล

### 3.2.1 การออกแบบโครงสร้างโดยใช้โปรแกรม SolidWorks

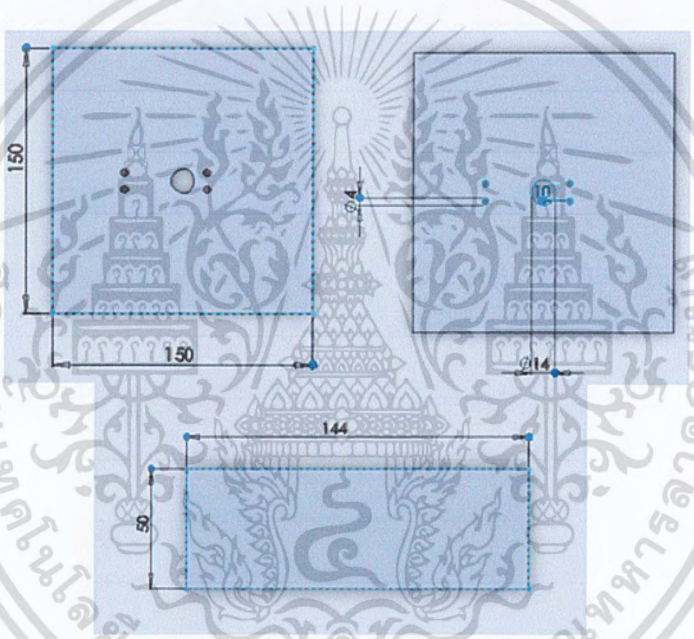
โปรแกรม Solid Works เป็นโปรแกรมการออกแบบและการเขียนแบบที่จำเป็นต่อการสร้างชิ้นส่วนของแขนกล โดยจุดเริ่มต้นของการเขียนแบบและออกแบบ คือ การสร้างชิ้นงานเบื้องต้นที่เรียกว่า Part ขึ้นมาก่อน ซึ่งPart จะเป็นส่วนประกอบพื้นฐานที่จะถูกนำมาประกอบรวมกันกลายเป็นชิ้นงานที่มีขนาดใหญ่และมีความซับซ้อนขึ้น ซึ่งจะถูกนำไปใช้ในการสร้างเป็นแขนกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราเขียนแบบเสร็จแล้วก็จะได้ G-code เพื่อเป็นคำสั่งให้เครื่องจักร CNC กัดชิ้นงานตามรูปแบบที่เราออกแบบไว้ โดยเมื่อกัดชิ้นงานออกเป็นส่วนย่อยๆหลายๆชิ้นแล้วก็นำมาประกอบเป็นแขนกล โดยใช้โปรแกรม SolidWorks ก่อน เพื่อป้องกันความผิดพลาดในการประกอบชิ้นงาน ซึ่งการประกอบชิ้นงานจะสำเร็จได้ต้องอาศัยการออกแบบชิ้นส่วนที่มีความสัมพันธ์กัน เพื่อให้สามารถประกอบเข้ากันได้ เมื่อได้ภาพ 3 มิติจากการประกอบแล้วจึงนำแต่ละชิ้นส่วนมาประกอบจริง

ส่วนประกอบต่างๆของแขนกลที่ต้องออกแบบโดยใช้โปรแกรม SolidWorks มีดังนี้

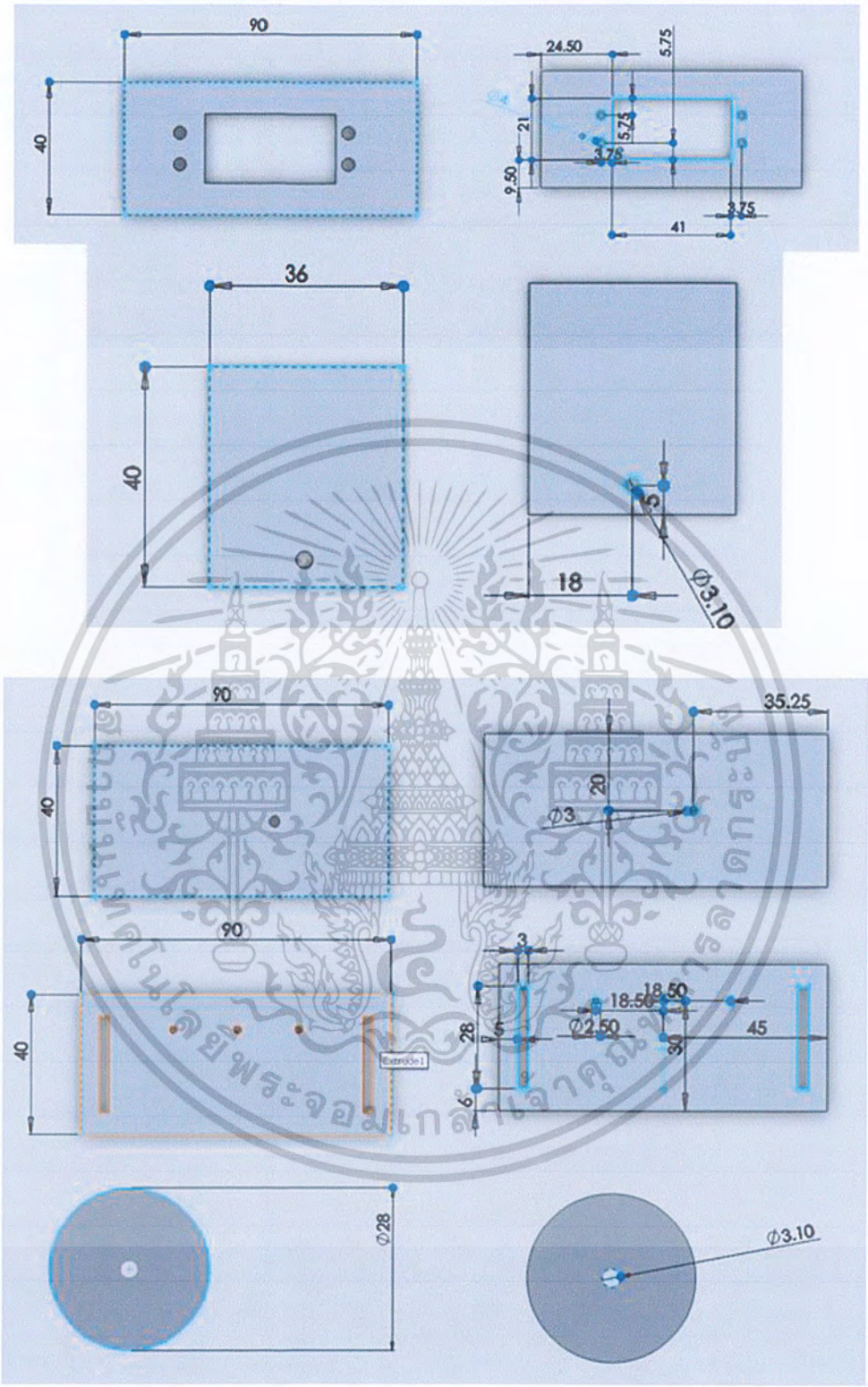
**3.2.1.1 ส่วนฐานของแขนกล** ส่วนนี้ทำหน้าที่เป็นตัวยึดให้แขนกลเกิดความมั่นคง สร้างสมดุลในการถ่วงน้ำหนักซึ่งจะช่วยให้แขนนั้นสามารถจับของที่อยู่ไกลแล้วตัวแขนกลไม่ล้มลง




**รูปที่ 3.4** ส่วนประกอบฐานของแขนกล

**3.2.1.2 ฐานหมุนของแขนกล** ทำหน้าที่ เป็นฐานเพื่อให้แขนกลนั้นหมุนไปหยิบของได้

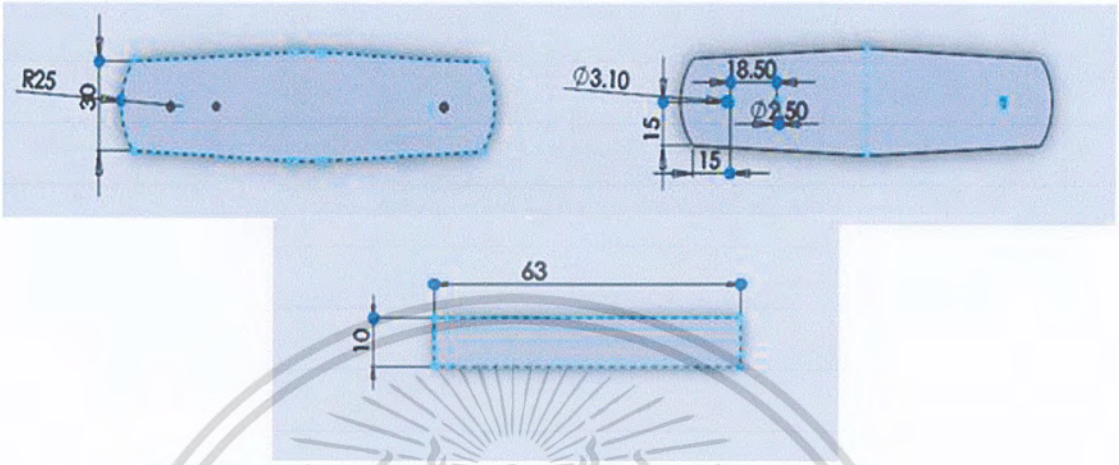
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 3.5** ส่วนประกอบของฐานหมุน

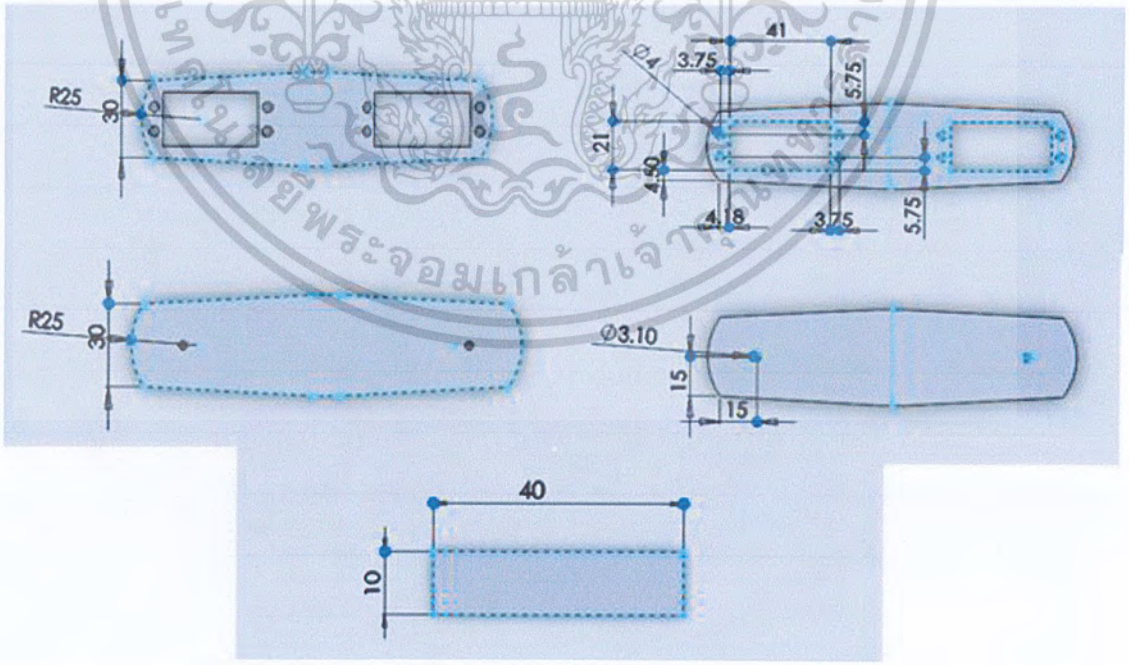
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**3.2.1.3 แขนส่วนที่ (1)** ทำหน้าที่คล้ายกับท่อนแขนที่ต่อจากหัวไหล่ของมนุษย์ ทำการยืดระยะในการหยิบของ




**รูปที่ 3.6** ส่วนประกอบของแขนส่วนที่ (1)

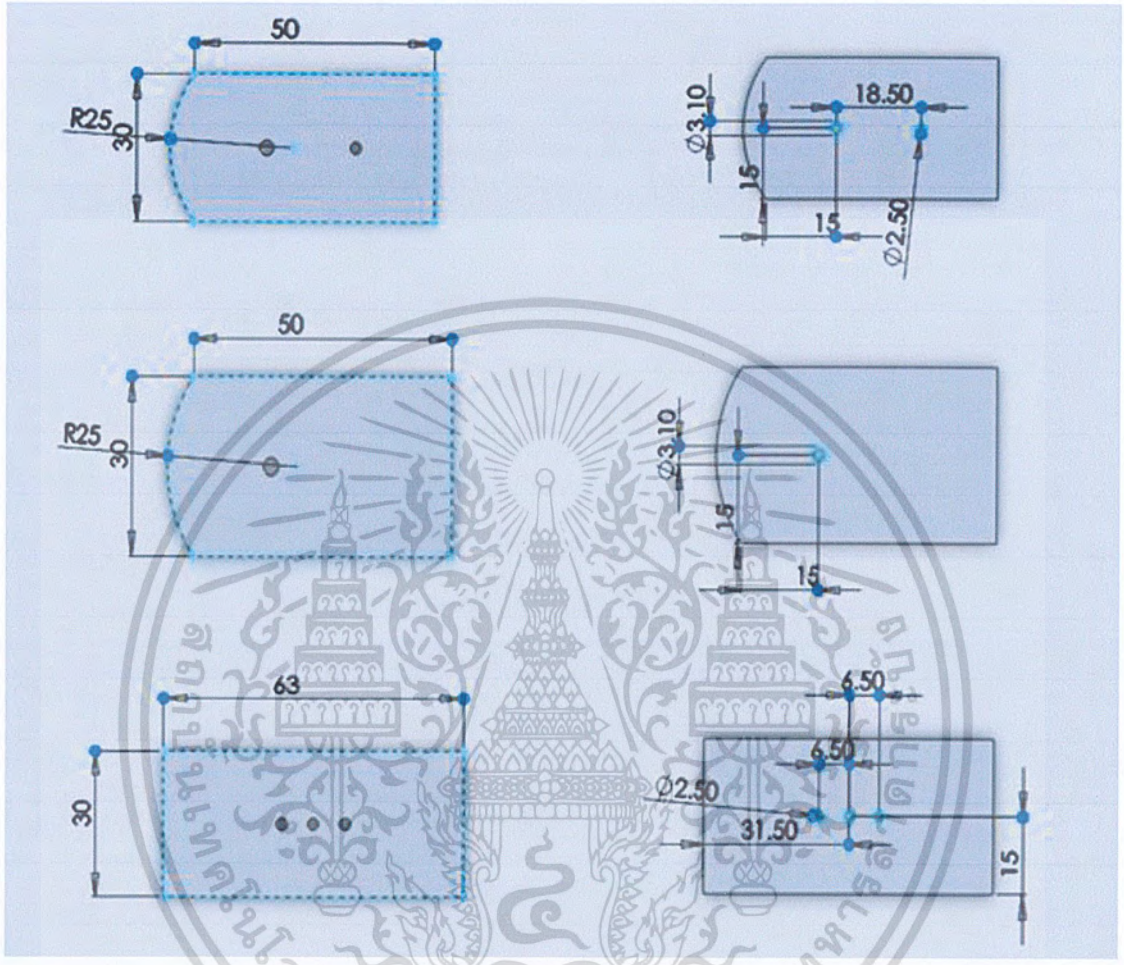
**3.2.1.4 แขนส่วนที่ (2)** ทำหน้าที่คล้ายกับท่อนแขนที่ต่อจากข้อศอกของมนุษย์ ทำการยืดระยะในการหยิบของ




**รูปที่ 3.7** ส่วนประกอบของแขนส่วนที่ (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**3.2.1.5 แขนส่วนที่ (3)** ทำหน้าที่คล้ายกับข้อมือที่ต่อจากแขนของมนุษย์ ทำการโยกขึ้นลงในการหยิบของ

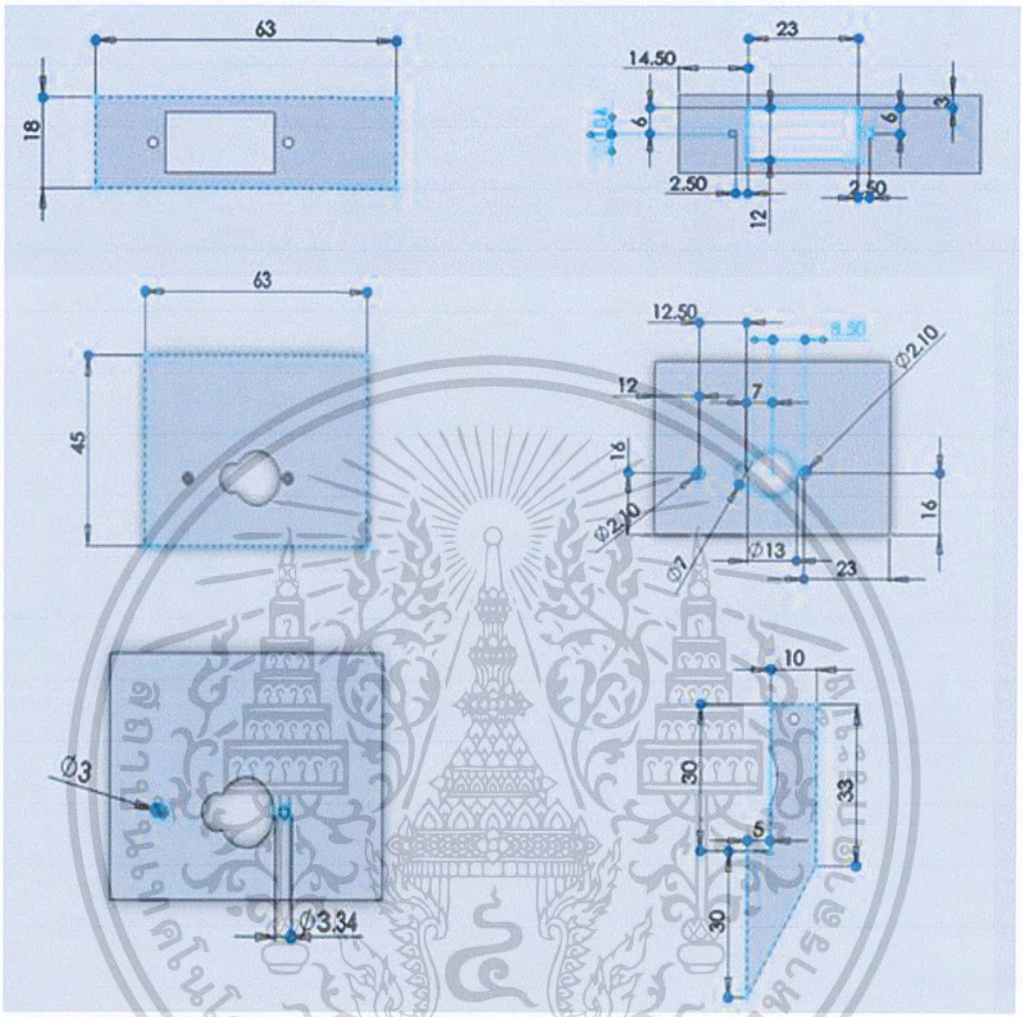



**รูปที่ 3.8** ส่วนประกอบของแขนส่วนที่ (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

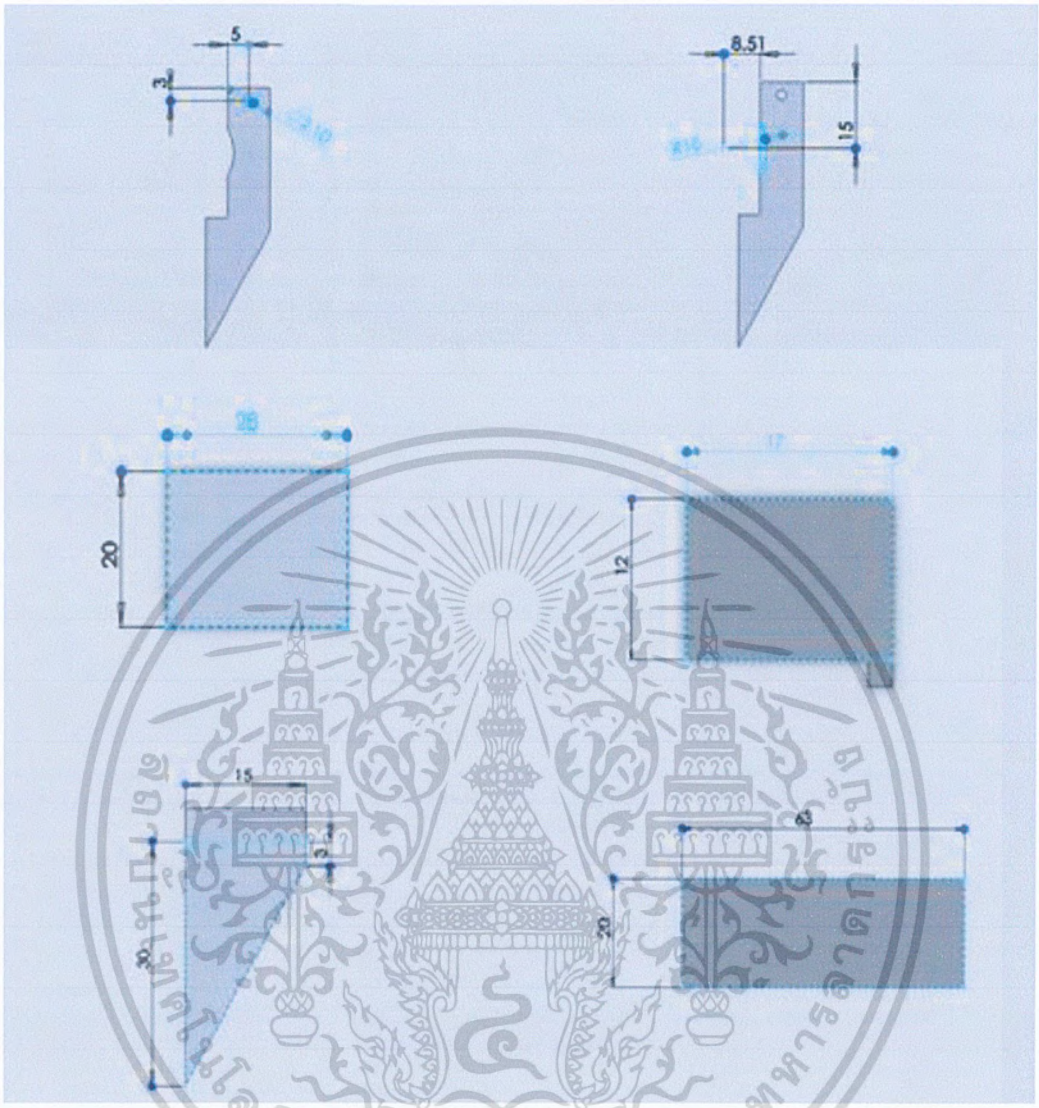
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง


**3.2.1.6 ตัวหนีบจับ** ทำหน้าที่คล้ายส่วนมือของมนุษย์



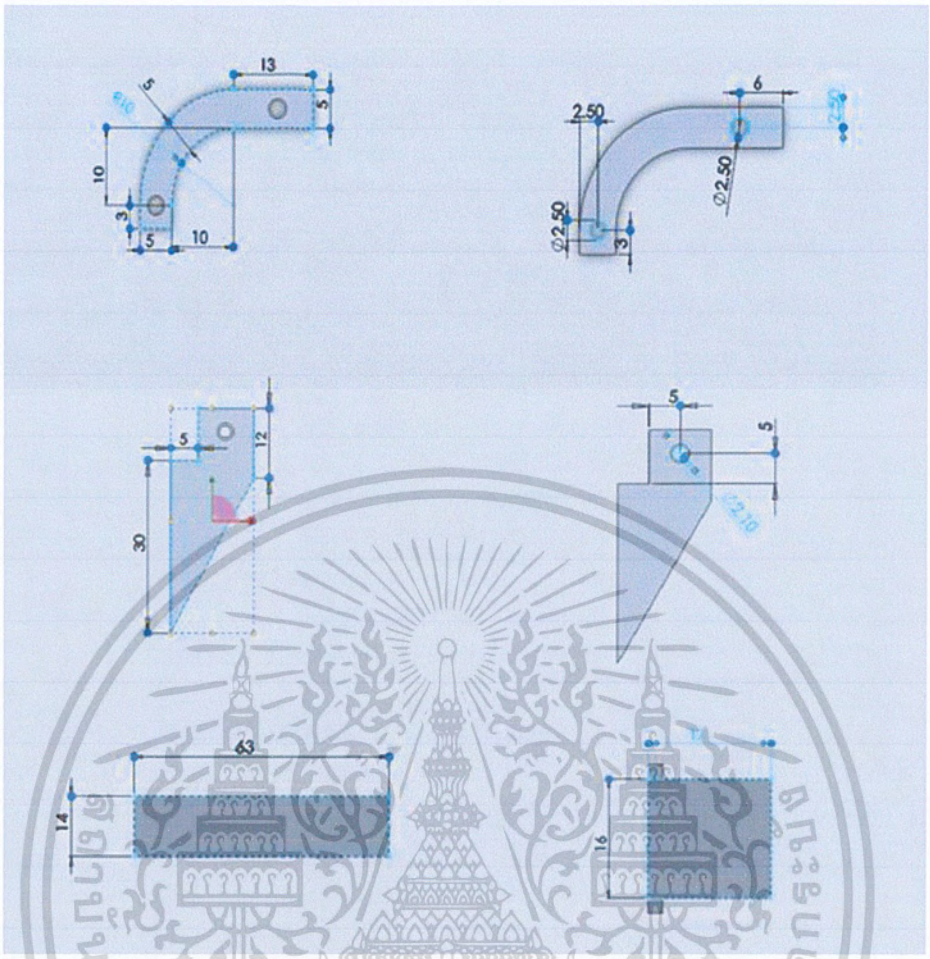

**รูปที่ 3.9** ส่วนประกอบของตัวหนีบจับ (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
119401



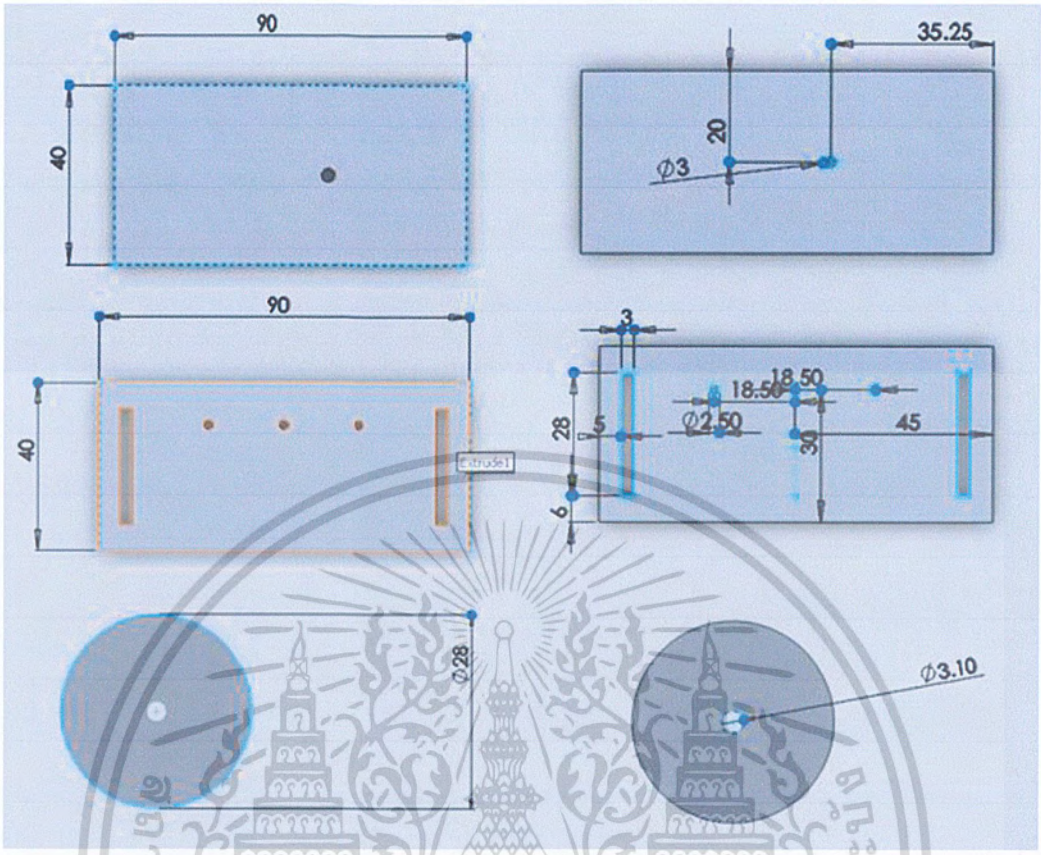

**รูปที่ 3.10** ส่วนประกอบของตัวหนีบจับ (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 3.11** ส่วนประกอบของตัวหนีบจับ (3)

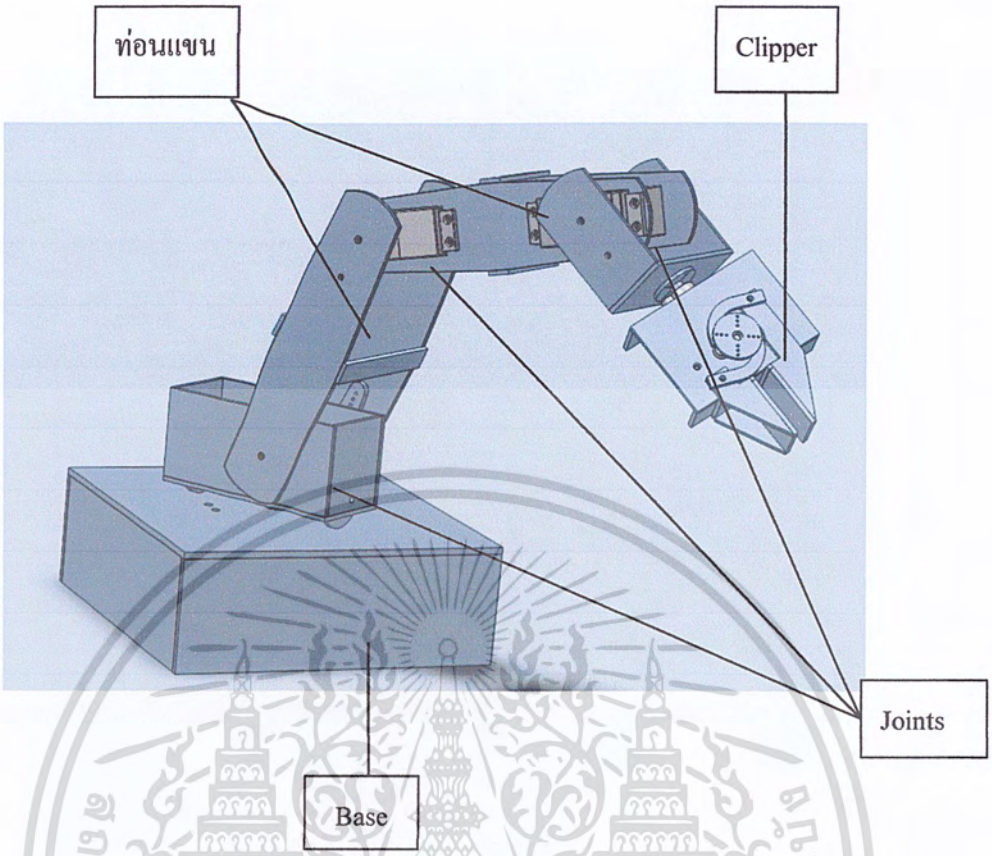
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่3.12 ส่วนประกอบของตัวหนีบจับ (4)

**3.2.1.7 โครงสร้างโดยรวมของแขนกล**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



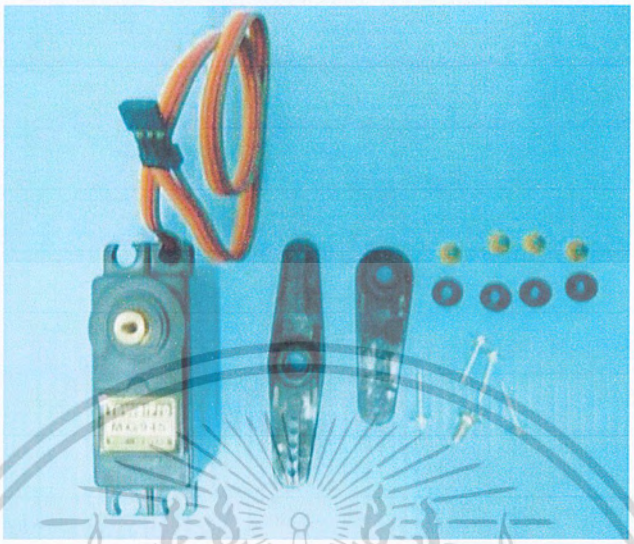

**รูปที่ 3.13** โครงสร้างโดยรวมของแขนกล

**3.2.2 อุปกรณ์ที่ใช้ต่อแขนกล1แขน**

1. Servoรุ่น MG945 4 ตัว
2. Servoรุ่น S3003 2 ตัว
3. Digital servo E-sky EK2-0508 1 ตัว
4. อะคริลิค หนา2 mm. 30*30 2 แผ่น
5. น้ำยาติดอะคริลิค 1 ขวด
6. อุปกรณ์ติดยึด เช่น น๊อต สกรู หกเหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Servoรุ่น MG945**



**รูปที่ 3.14** Servoรุ่น MG 945

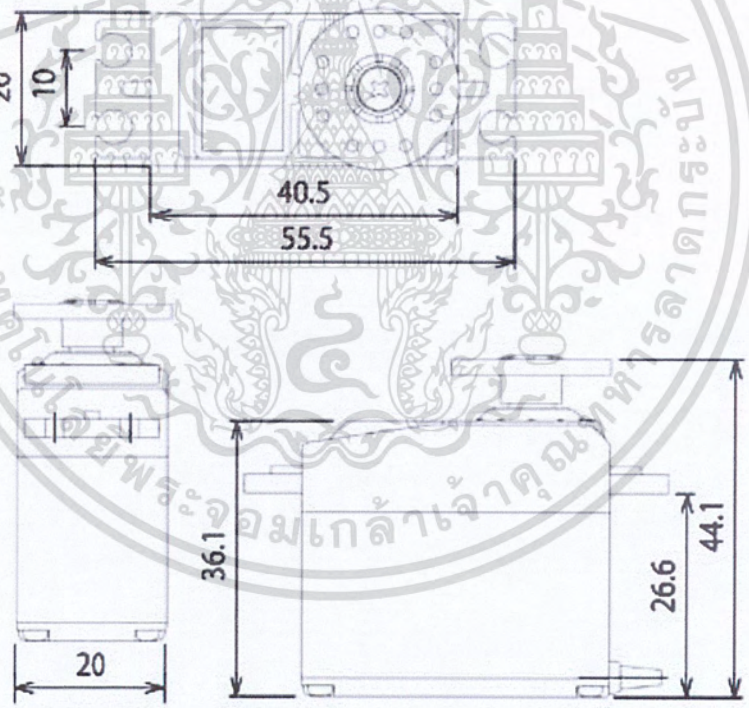
**รายละเอียดคุณสมบัติ (Detailed Specifications)**

- Dimension:40.7*19.7*42.9mm
- Stall torque:12kg·cm(4.8v)
- Operating speed:0.25sec/60degree(4.8v)
- Operating voltage:4.8-7.2V
- Temperature range:0C_ 55C
- Dead band width : 10us
- weight:64.0g
- quantity:1
- MG945 metal gear servo 12kg stall torque
- MG945 metal gear standard size servo
- Compatible with all kind of RC receiver

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Servoรุ่น S3003**

**รูปที่ 3.15** Servoรุ่น S3003




**รูปที่3.16** สัดส่วนของ Servoรุ่น S3003

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**รายละเอียดคุณสมบัติ (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): 0.23sec/60 degrees at no load
- Operating Speed (6.0V): 0.19sec/60 degrees at no load
- Stall Torque (4.8V): 44 oz/in. (3.2kg.cm)
- Stall Torque (6.0V): 56.8 oz/in. (4.1kg.cm)
- Operating Angle: 45 Deg. one side pulse traveling 400usec
- 360 Modifiable: Yes
- Direction: Counter Clockwise/Pulse Traveling 1520-1900usec
- Current Drain (4.8V): 7.2mA/idle
- Current Drain (6.0V): 8mA/idle
- 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)
- Weight: 1.3oz. (37.2g)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Digital servo motor EK 2-0508**

**รูปที่ 3.17** Digital servo motor EK 2-0508

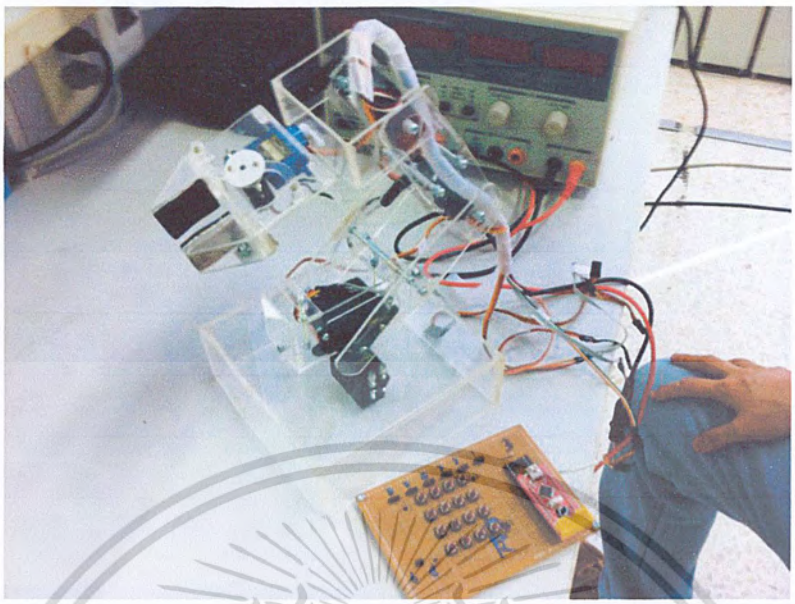
**รายละเอียดคุณสมบัติ (Detailed Specifications)**

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

### 3.2.3 การประกอบโครงสร้างของแขนกล

โครงสร้างของแขนกลทำมาจากแผ่นอะคริลิคหนา 2 มิลลิเมตร ทำให้มีน้ำหนักที่พอดีกับการใช้เซอร์โวมอเตอร์ในการเคลื่อนที่ของแขนกลแต่ละส่วน โดยมีเซอร์โวมอเตอร์อยู่ที่แต่ละข้อของแขนกล ทำให้สามารถเคลื่อนที่ได้เปรียบเสมือนแขนของมนุษย์ เซอร์โวมอเตอร์แต่ละตัวจะถูกควบคุมโดยการส่งสัญญาณจากบอร์ดควบคุม STAMP 168 ที่ถูกบันทึกโปรแกรมลงไว้ในบอร์ดแล้วตั้งค่าการหมุนของเซอร์โวมอเตอร์ด้วยการทดลองให้หมุนทีละตัวแล้วบันทึกค่านั้นทีละครั้งจนครบกระบวนการทำงานการประกอบแขนกลที่สมบูรณ์ แสดงดังรูปที่ 3.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 โครงสร้างแขนกลที่สมบูรณ์

## 3.3 การควบคุมทิศทางการหมุนของเซอร์โวมอเตอร์

### 3.3.1 เซอร์โวมอเตอร์

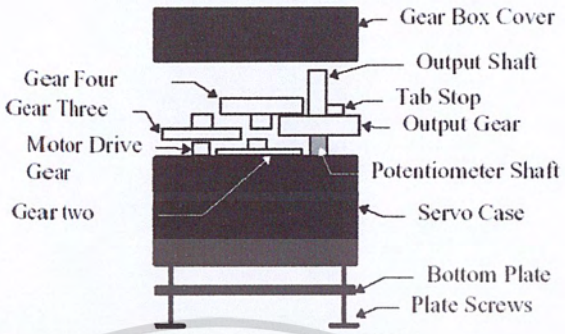
เซอร์โวมอเตอร์เป็นมอเตอร์ไฟฟ้ากระแสตรง (DC motor) ประกอบด้วยชุดเกียร์ และส่วนควบคุมต่างๆ ไว้ในโมดูลเดียวกัน มีสายต่อใช้งานเพียง 3 เส้นเท่านั้น คือ VCC, GND และสายสัญญาณควบคุม(Control Line) ซึ่งสามารถควบคุมให้ตัวเซอร์โวมอเตอร์หมุนซ้าย หรือ ขวาได้ +90 องศา - 90 องศา (180 องศา) โดยสัญญาณที่ใช้ควบคุมนี้จะเป็นสัญญาณการมอดูเลตความกว้างพัลส์ (Pulse Width Modulation ,PWM) แบบ TTL Level ระดับแรงดันที่จ่ายให้มอเตอร์นี้จะอยู่ในช่วงประมาณ 4 ถึง 6 V โดยสามารถสั่งงานในการหมุนให้หมุนไปได้ตามองศาต่างๆ ที่ต้องการได้ด้วยตัวของเซอร์โวมอเตอร์เอง ไม่ต้องมีส่วนควบคุมใดๆ กลับมาตรวจสอบอีกทำให้ง่าย และสะดวกในการในการนำไปประยุกต์ใช้งานต่างๆ

- การควบคุมการทำงานของเซอร์โวมอเตอร์ทำได้โดยการป้อนสัญญาณความกว้างของพัลส์ให้กับตัวเซอร์โวมอเตอร์ซึ่งจะได้ทิศทางการหมุนและตำแหน่งของการหมุน

- สามารถใช้งานกับไฟ DC ได้ 4 - 6 V, หมุนได้ 180 องศา และสามารถปรับแต่งตัวเซอร์โวมอเตอร์ให้สามารถหมุนได้รอบตัวได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

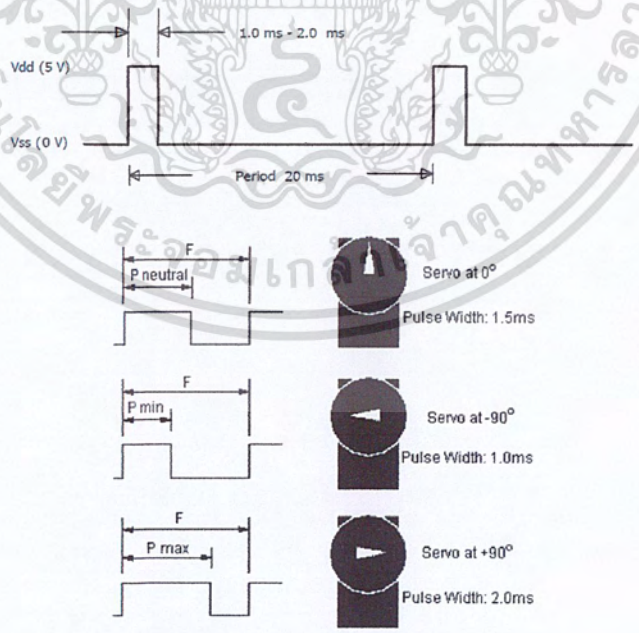
- ขั้วต่อจะเป็นแบบมาตรฐาน : ขั้ว JR TYPE




**รูปที่ 3.19** โครงสร้างของเซอร์โวมอเตอร์

### 3.3.2 หลักการทำงานของเซอร์โวมอเตอร์

การควบคุมการทำงานทำได้โดย การป้อนสัญญาณความกว้างพัลส์ ให้กับมอเตอร์ซึ่งตำแหน่งและทิศทางการหมุนของมอเตอร์นี้จะขึ้นอยู่กับขนาดของความกว้างของพัลส์นั้นๆ โดยทั่วไปแล้วความกว้างของสัญญาณพัลส์จะมีจุดให้อ้างอิง 3 จุด ดังรูป




**รูปที่ 3.20** จุดอ้างอิงความกว้างของสัญญาณพัลส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สัญญาณความกว้างพัลส์ขนาด 1.5 ms จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม 0 องศา หรือ จุดกึ่งกลางของมอเตอร์
- สัญญาณความกว้างพัลส์ขนาด 1 ms จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม - 90 องศา หรือในทิศทางทวนเข็มนาฬิกา
- สัญญาณความกว้างพัลส์ขนาด 2 ms จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม + 90 องศา หรือในทิศทางตามเข็มนาฬิกา

ส่วนการที่จะควบคุมให้มอเตอร์หมุนเป็นมุมอื่นๆ นั้นก็สามารถทำได้โดยการป้อนสัญญาณพัลส์เป็นระดับความกว้างต่างๆ โดยอ้างอิงจากจุด ทั้ง 3 จุดที่กล่าวมานี้ ซึ่งสัญญาณพัลส์นี้จะต้องจ่ายให้มอเตอร์ทุกๆ 20 ms (Period) เพื่อรักษาสภาพตำแหน่งของมอเตอร์ไว้ก็คือ จะอาศัยการเปรียบเทียบช่วงเวลาของความกว้างพัลส์ที่จ่ายให้กับมอเตอร์ทางขาสัญญาณควบคุมกับค่าเวลาของวงจร RC ภายในบอร์ดควบคุมในตัวของมอเตอร์ ซึ่งค่าเวลาของวงจร RC นี้จะมีการเปลี่ยนแปลงตามการหมุนของมอเตอร์ เนื่องจากตัวต้านทานปรับค่าจะถูกยึดติดอยู่กับแกนหมุนของมอเตอร์ ซึ่งการหมุนของมอเตอร์จะทำให้ค่าความต้านทานของตัวต้านทานปรับค่า (VR) เปลี่ยนแปลงไป เป็นผลทำให้ค่าเวลาของวงจร RC เปลี่ยนแปลงตามไปด้วย โดยในขณะที่เราป้อนสัญญาณความกว้างพัลส์ให้กับมอเตอร์ทางขาสัญญาณควบคุม สัญญาณนี้จะถูกนำไปเปรียบเทียบกับค่าเวลาของวงจร RC หากค่าทั้ง 2 ไม่เท่ากันมอเตอร์ก็จะหมุนทำให้ค่าเวลาของวงจร RC เปลี่ยนแปลงจนกระทั่งค่าเวลาความกว้างพัลส์ของ วงจร RC เปลี่ยนแปลงจนเท่ากับสัญญาณพัลส์ทางขาควบคุม มอเตอร์จึงจะหยุดหมุน

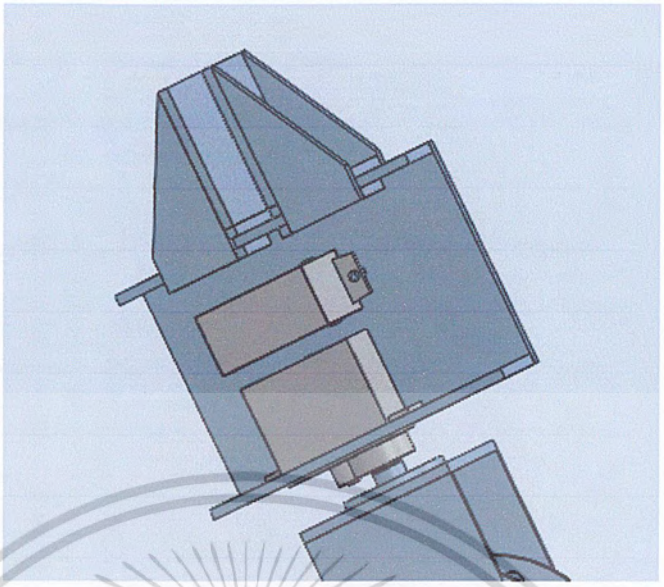
### 3.3.3 การทำงานของเซอร์โวมอเตอร์แต่ละส่วน

การทำงานของเซอร์โวมอเตอร์ทั้ง 6 ตัวจะแบ่งหน้าที่การทำงานดังนี้

- **ส่วนมือจับ** จะมีเซอร์โวมอเตอร์ ทั้งหมด 3 ตัวด้วยกันคือ

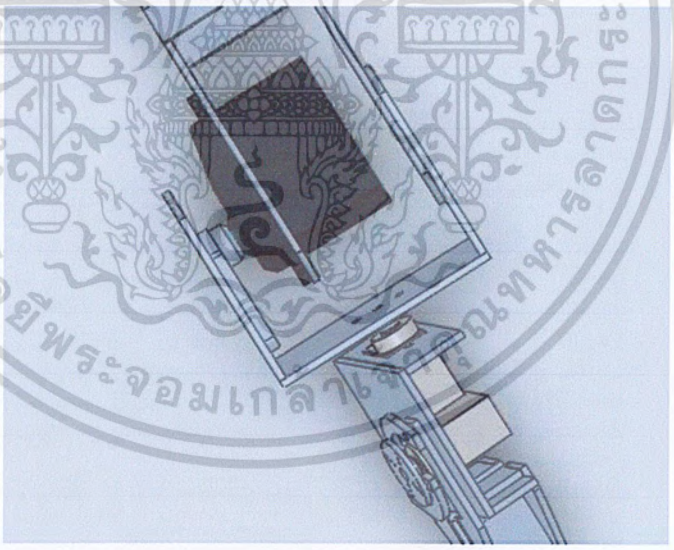
- ตัวที่ 1 จะใช้เซอร์โวมอเตอร์รุ่น Esky EK2-0508 จะควบคุมส่วนคลิ๊ปเปอร์หรือมือจับทำหน้าที่ขับเคลื่อนมือจับให้จับวัตถุ
- ตัวที่ 2 จะใช้เซอร์โวมอเตอร์รุ่น SG-90 จะควบคุมการหมุนซ้าย-ขวาของมือจับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 3.21** เซอร์โวมอเตอร์ส่วนคลิ๊ปเปอร์และส่วนควบคุมการหมุนของมือจับ

- ตัวที่ 3 จะใช้เซอร์โวมอเตอร์รุ่น S3003 จะควบคุมการยกมือจับขึ้น-ลง

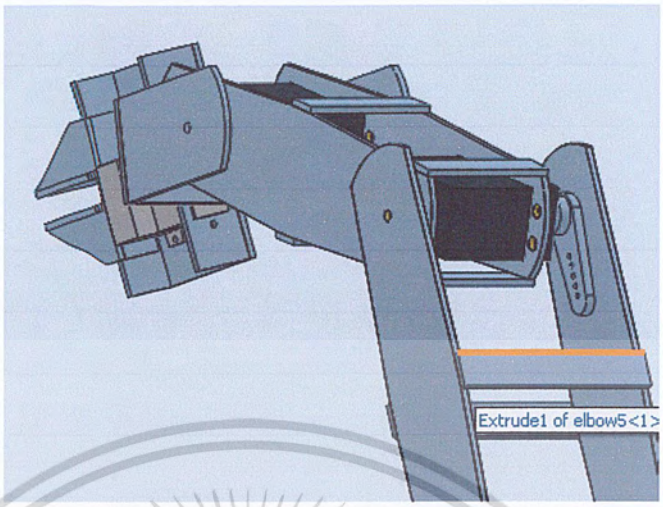


**รูปที่ 3.22** เซอร์โวมอเตอร์ส่วนควบคุมการยกมือจับขึ้น-ลง

- **ส่วนแขน** จะแบ่งเป็น 2 ส่วนคือ แขนท่อนบนและแขนท่อนล่าง ซึ่งจะมีเซอร์โวมอเตอร์รุ่น S3003 ทั้งหมด 2 ตัวด้วยกันคือ

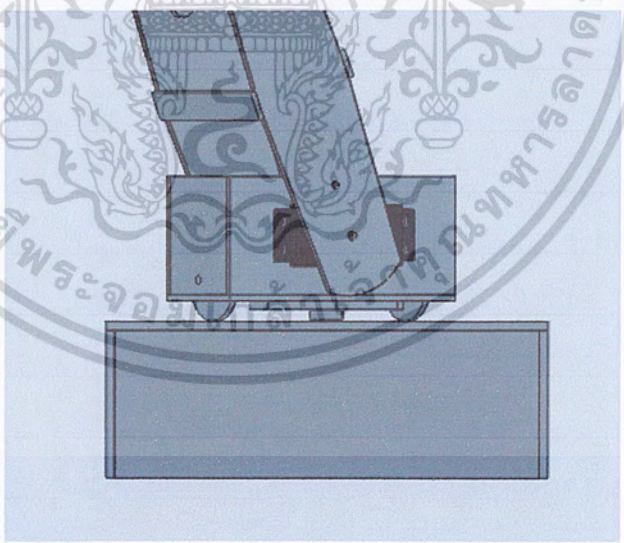
- ตัวที่ 1 จะควบคุมแขนให้ยกสิ่งของขึ้น-ลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 3.23** เซอร์โวมอเตอร์ส่วนควบคุมแขนให้ยกสิ่งของขึ้น-ลง

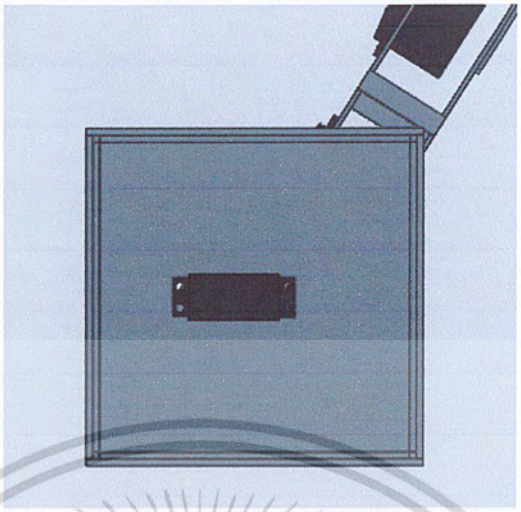
- ตัวที่ 2 จะควบคุมแขนท่อนล่าง เพื่อที่จะควบคุมทั้งแขนและมือจับให้สามารถที่จะไปหยิบจับวัตถุที่มีระยะห่างออกไป ซึ่งจะเป็นส่วนที่รับน้ำหนักมากที่สุดของแขน



**รูปที่ 3.24** เซอร์โวมอเตอร์ส่วนควบคุมแขนท่อนล่าง

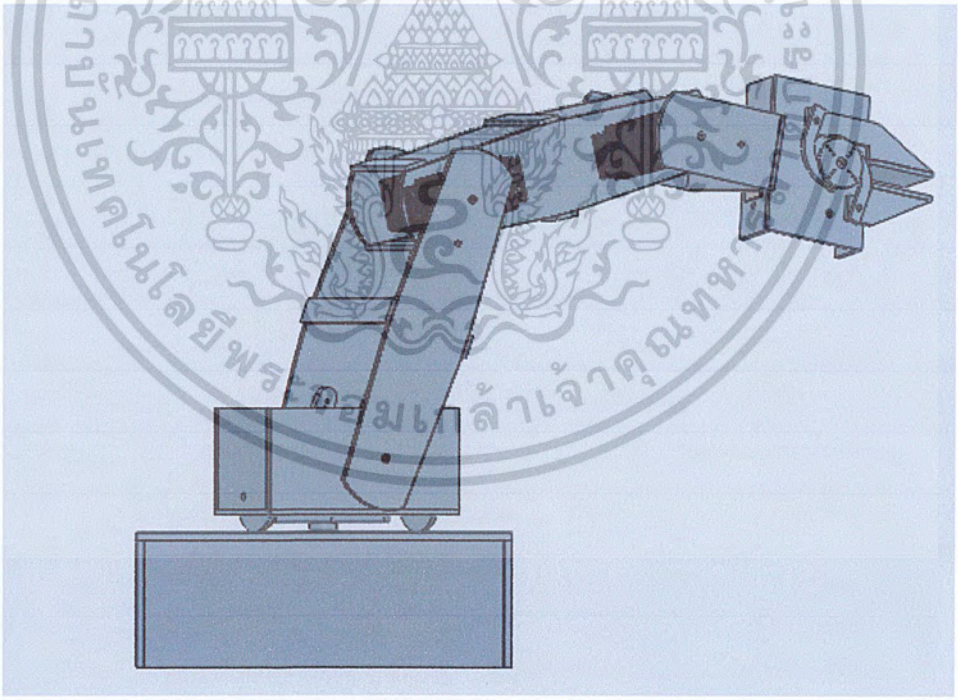
- **ส่วนฐาน** จะมีเซอร์โวมอเตอร์รุ่น S3003 เพียง 1 ตัวด้วยกัน จะทำหน้าที่หมุนแขนกลทั้งระบบไปทางซ้าย-ขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 3.25** เซอร์โวมอเตอร์ส่วนฐาน

การแสดงตำแหน่งของเซอร์โวมอเตอร์ในแต่ละข้อต่อ แสดงดังรูปที่ 3.26

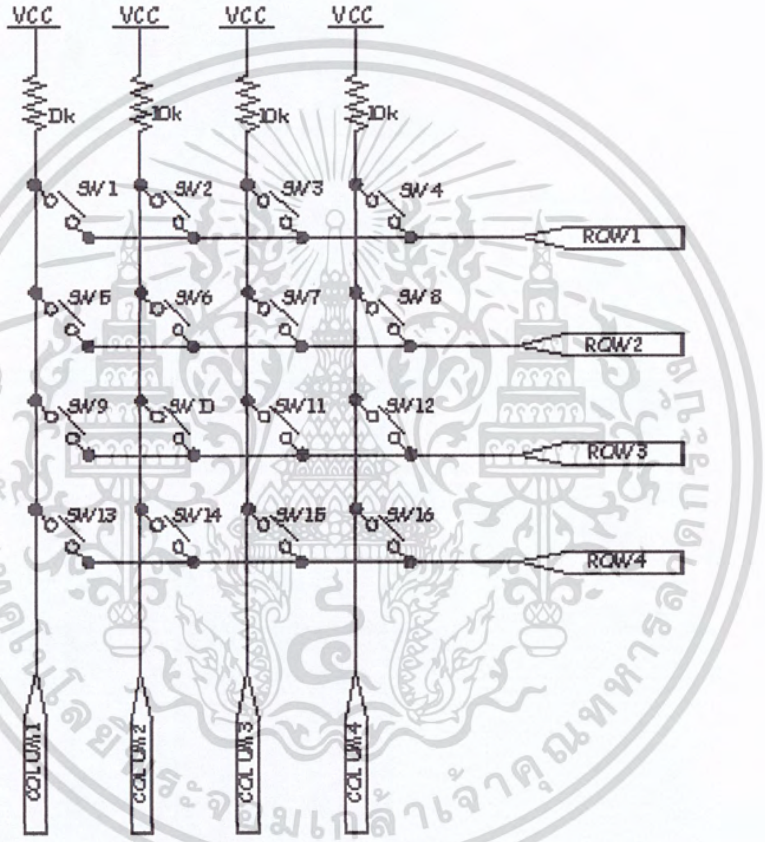


**รูปที่ 3.26** ตำแหน่งเซอร์โวมอเตอร์ในแต่ละข้อต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

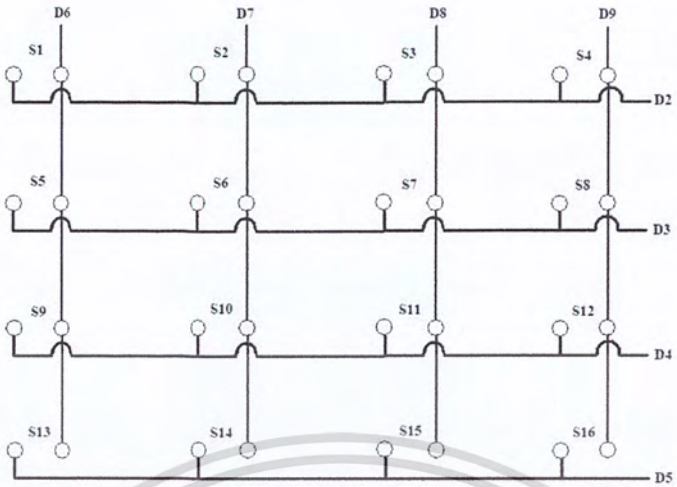
### 3.4 วงจรอิเล็กทรอนิกส์ควบคุมการบังคับแขนกล

วงจรของสวิตช์เมตริกขนาด 4 x 4 แสดงดังรูปที่ 3.27 สวิตช์แต่ละตัวจะเชื่อมต่อกันแบบแถวและแบบหลัก ในรูปแบบของเมตริกซ์ ในการตรวจสอบการกดสวิตช์แบบเมตริกซ์นั้น ทำได้โดยการป้อนค่าตรวจสอบค่าหนึ่งไปยังด้านแถว และตรวจสอบการเปลี่ยนแปลงที่เกิดขึ้นทางด้านหลัก อาจจะกล่าวได้ว่าตำแหน่งของการกดสวิตช์ต่างๆ ได้จากการเขียนโปรแกรมเพื่อวนลูปส่งค่าตรวจสอบไปทางด้านแถวและตรวจสอบค่าที่รับเข้ามาทางด้านหลัก

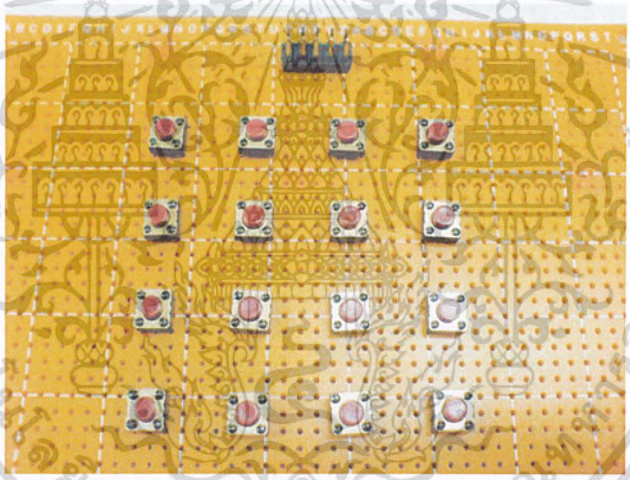



**รูปที่ 3.27** สวิตช์แบบเมตริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 3.28** วงจรสวิตซ์ควบคุม (Keypad Joystick)



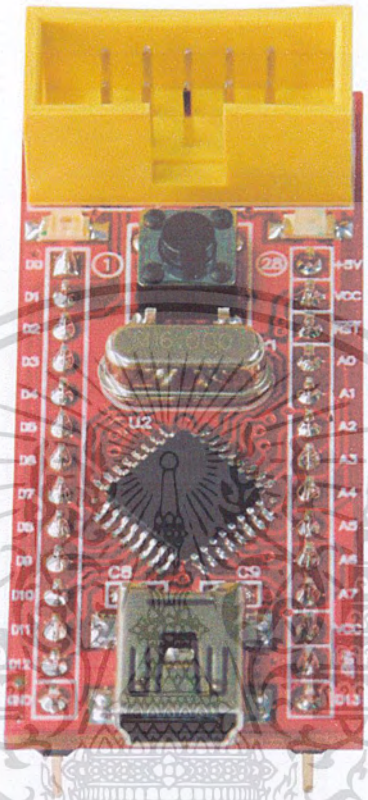
**รูปที่ 3.29** บอร์ดควบคุมการเคลื่อนไหวของหุ่นยนต์

ประยุกต์ใช้วงจรควบคุมในการกำหนดการเคลื่อนที่ของแขนกล โดยใช้ Digital Pin D2-D9 ซึ่งแต่ละ Pin จะส่งสัญญาณ Logic 1 ออกมา เช่น สวิตช์ S1 จะมี Pin D2 กับ D6 เชื่อมกันอยู่ เมื่อกดปุ่ม S1 Logic จาก D2 และ D9 จะเชื่อมถึงกัน ทำให้ไมโครคอนโทรลเลอร์ (Microcontroller) รู้ว่ามีการกดปุ่ม S1 ตามที่เขียนโปรแกรมไว้ โดยเซอร์โวมอเตอร์ 1 ตัวจะถูกควบคุมด้วยสวิตช์ 2 ตัว เพื่อควบคุมทิศทางในการหมุนตามเข็มหรือทวนเข็มนาฬิกา ตามที่ต้องการให้เคลื่อนที่ในทิศทางนั้น เมื่อได้ตำแหน่งที่ต้องการแล้วให้กดสวิตช์ที่เป็นการบันทึกค่า จะสั่งให้จำค่านั้นไว้ ซึ่งในบอร์ดควบคุมนี้สามารถบันทึกค่าได้ทั้งหมด 8 ค่า เมื่อทำการบันทึกค่าทั้งหมดแล้ว ทำการกดสวิตช์ RUN จะเป็นการสั่งให้แขนกลทำงานได้อัตโนมัติตามที่สั่งค่าไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.5 บอร์ดควบคุมการทำงาน

- **บอร์ด Stamp168**



**รูปที่ 3.30** โครงสร้างของบอร์ด ET-EASY168 STAMP

ET-EASY168 STAMP เป็นบอร์ดไมโครคอนโทรเลอร์ในตระกูล AVR8 ขนาดเล็กจิ๋ว โดยมีขนาดของบอร์ดเพียง 2cm x 5cm เท่านั้น ซึ่งขนาดบอร์ด ประมาณเท่ากับตัวถังของไอซี 28 DIP 300 โดยเลือกใช้ไมโครคอนโทรเลอร์ตระกูล AVR8 เบอร์ ATmega168 ของ ATMEL เป็น MCU ประจำบอร์ด โดยเลือกใช้ MCU ที่มีรูปร่างตัวถังแบบ 32 TQFP พร้อมวงจรรอบนอกที่จำเป็นอย่าง Oscillator และ Reset รวมไว้ด้วยภายในบอร์ด นอกจากนี้แล้วภายในตัวบอร์ดยังได้รวมเอาไอซี USB Bridge ของ FTDI เบอร์ FT232R เพื่อใช้ติดต่อสื่อสารแบบอนุกรมด้วย RS232 กัคอมพิวเตอร์ PC ผ่านทางพอร์ต USB ได้โดยตรง

ทำให้บอร์ด ET-EASY168 STAMP เป็นบอร์ดทดลองขนาดเล็กที่เพียบพร้อมไปด้วยวงจรพื้นฐานที่จำเป็นต่อการใช้งานไมโครคอนโทรลเลอร์ตระกูล AVR8 อย่างแท้จริง เพียงแต่เสียบสาย

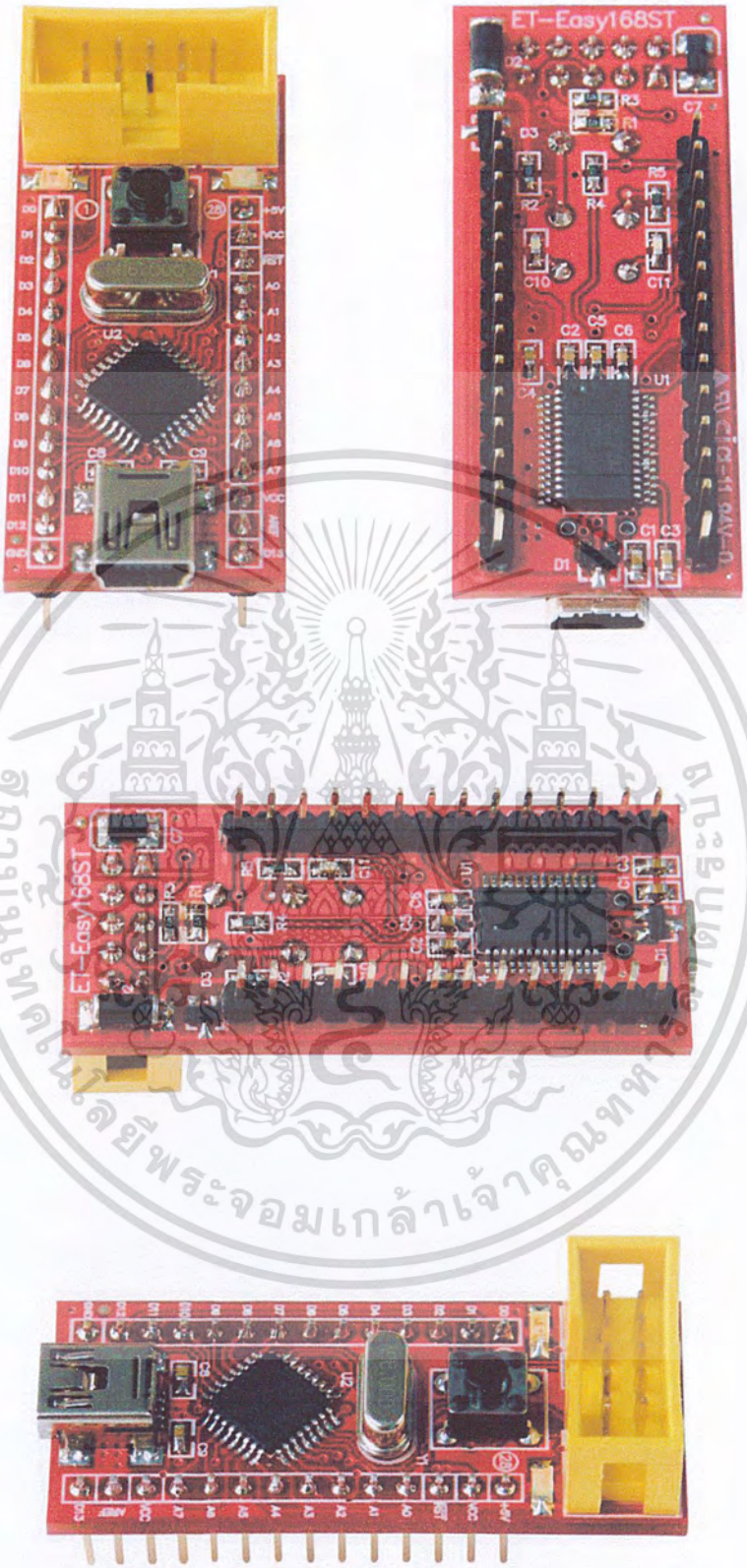
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

USB จากพอร์ตUSB ของเครื่องคอมพิวเตอร์ PC เข้ากับขั้ว USB ของบอร์ด ET-EASY168 STAMP ก็สามารถทำการเขียนโปรแกรม และ Download Code ให้กับ MCU เพื่อทำการทดลองได้ทันที

**คุณสมบัติของบอร์ด**

- เลือกใช้ MCU ตระกูล AVR8 เบอร์ ATmega168 ของ ATMEL Run ความถี่ 16.00 MHz
  - มีหน่วยความจำ Flash สำหรับเขียนโปรแกรม 16KByte ถ้าใช้การพัฒนาโปรแกรมผ่านระบบ AVRISP หรือ 14Kbyte เมื่อใช้การพัฒนาโปรแกรมผ่านระบบ Boot Loader RS232
  - มี SRAM ใช้งานขนาด 1KByte และ EEPROM ใช้งานขนาด 512 Byte
  - มี GPIO ใช้งานจำนวน 22 บิต
- Digital GPIO จำนวน 14 บิต
- Analog Input (ADC) ขนาดความละเอียด 10บิต จำนวน 8 ช่อง
- ใช้งานกับแรงดันไฟตรงขนาด +5VDC โดยใช้ได้ทั้งกับแหล่งจ่าย +5VDC/500mA จากพอร์ต USBและจากแหล่งจ่าย +5VDC จากภายนอกได้ด้วย พร้อม LED Power แสดงสถานะของแหล่งจ่าย
- มีวงจร External Reset แบบ RC Reset และ Switch Reset พร้อมภายในบอร์ด
- ขั้วต่อใช้งานวางตัวบน Pin Header ระยะห่าง 2.54mm(100mil) ขนาด 28 Pin (ด้านละ 14Pin)ระยะห่าง 600mil(1.5cm) ง่ายต่อการนำไปต่อประยุกต์ใช้งาน และ ขยายวงจร I/O สามารถใช้กับ Project Board และ PCB เอนกประสงค์ได้โดยง่าย
- มีขั้วต่อ USB สำหรับเชื่อมต่อสื่อสารกับคอมพิวเตอร์ PC ผ่าน USB Bridge ของ FTDI ในรูปแบบของการสื่อสารอนุกรม RS232 สำหรับใช้งานสื่อสารและ Download Code ให้กับ MCU ในบอร์ด
- มีขั้ว AVRISP แบบ IDE 10PIN สำหรับใช้ Download โปรแกรมให้กับ MCU ภายในบอร์ดในกรณีไม่ต้องการใช้การพัฒนาโปรแกรมผ่านทาง Boot Loader
- มี LED แสดงสถานะ โดยต่อกับ PB5 ของ AVR (Digital-13 ของ Arduino Project) สำหรับใช้เป็นอุปกรณ์ทดลองการทำงานอย่างง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

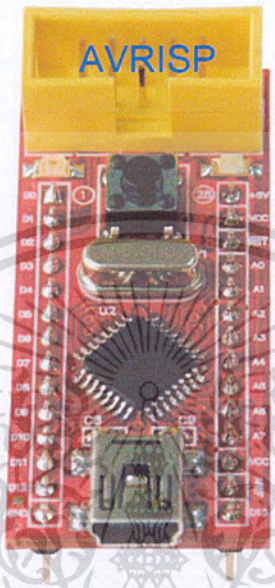


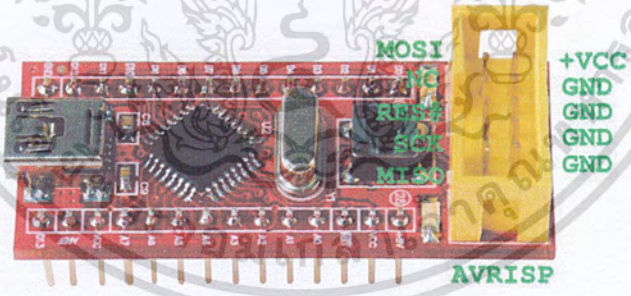

**รูปที่ 3.31** ลักษณะของบอร์ด ET-EASY168 STAMP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.1** การจัดสรรขาสัญญาณของบอร์ด ET-EASY168 STAMP

| AVR | Arduino | Pin | ET-EASY168 STAMP | Pin | Arduino | AVR |
|---|---|---|---|---|---|---|
| PD0 | Digital-0 | 1 | | 28 | +5V(+Vin) | +5V(+Vin) |
| PD1 | Digital-1 | 2 | | 27 | +VCC(+5V) | +VCC(+5V) |
| PD2 | Digital-2 | 3 | | 26 | RESET# | RESET(PC6) |
| PD3 | Digital-3 | 4 | | 25 | Analog-0 | PC0/ADC0 |
| PD4 | Digital-4 | 5 | | 24 | Analog-1 | PC1/ADC1 |
| PD5 | Digital-5 | 6 | | 23 | Analog-2 | PC2/ADC2 |
| PD6 | Digital-6 | 7 | | 22 | Analog-3 | PC3/ADC3 |
| PD7 | Digital-7 | 8 | | 21 | Analog-4 | PC4/ADC4 |
| PB0 | Digital-8 | 9 | | 20 | Analog-5 | PC5/ADC5 |
| PB1 | Digital-9 | 10 | | 19 | Analog-6 | ADC6 |
| PB2 | Digital-10 | 11 | | 18 | Analog-7 | ADC7 |
| PB3 | Digital-11 | 12 | | 17 | +VCC(+5V) | +VCC(+5V) |
| PB4 | Digital-12 | 13 | | 16 | +AREF | +AREF |
| GND | GND | 14 | | 15 | Digital-13 | PB5 |




**รูปที่ 3.32** ขาสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.2** หน้าที่ของขาสัญญาณ

| AVR | Arduino | Pin | AVRISP | Pin | Arduino | AVR |
|---|---|---|---|---|---|---|
| PB3 | Digital-11 | MOSI | | +VCC | +VCC | +VCC |
| - | - | NC | | GND | GND | GND |
| RES# | RES# | RES# | | GND | GND | GND |
| PB5 | Digital-13 | SCK | | GND | GND | GND |
| PB4 | Digital-12 | MISO | | GND | GND | GND |

**หน้าที่ของขาสัญญาณในการใช้งานแบบ "Arduino Project"**

- **+5V(+Vin)** เป็นขาสำหรับใช้เป็นจุดรับแรงดันขนาด +5VDC จากภายนอกเพื่อใช้เป็นแหล่งจ่ายไฟเลี้ยงให้กับบอร์ด
- **+VCC(+5V)** เป็นขาแหล่งจ่ายไฟจุดเดียวกันกับที่ป้อนให้กับ +VCC ของ MCU ซึ่งจุดนี้จะรับแรงดันมาจาก 2 แหล่ง ด้วยกันคือ ขารับแรงดัน **+5V(+Vin)** จากขา 28 ของบอร์ด และ จากขา+**VUSB(+5V)** จากขั้ว USB ของบอร์ด โดยมีไดโอด ป้องกันการย้อนกลับของแรงดันไว้แล้ว
- **+AREF** เป็นขาสำหรับรับสัญญาณแรงดันอ้างอิง (Analog Reference) ให้กับวงจร Analog Input ในกรณีต้องการใช้แรงดันอ้างอิงจากภายนอก
- **RESET#** เป็นขาสัญญาณ RESET ของ CPU ทำงานที่ Logic "0"
- **Digital[0..13]** เป็นขา I/O แบบดิจิตอล สามารถใช้งานเชื่อมต่อกับสัญญาณ Logic TTL (5V) ต่างๆ
- **Analog[0..7]** เป็นขาอินพุต แบบแอนะล็อค สามารถรับอินพุต แบบแอนะล็อค 0..+5V

**หน้าที่ของขาสัญญาณในการใช้งานแบบ "AVR Micro Controller"**

- **+5V(+Vin)** เป็นขาสำหรับใช้เป็นจุดรับแรงดันขนาด +5VDC จากภายนอกเพื่อใช้เป็นแหล่งจ่ายไฟเลี้ยงให้กับบอร์ด
- **+VCC(+5V)** เป็นขาแหล่งจ่ายไฟจุดเดียวกันกับที่ป้อนให้กับ +VCC ของ MCU ซึ่งจุดนี้จะรับแรงดันมาจาก 2 แหล่ง ด้วยกันคือ ขารับแรงดัน **+5V(+Vin)** จากขา 28 ของบอร์ด และ จากขา+**VUSB(+5V)** จากขั้ว USB ของบอร์ด โดยมีไดโอด ป้องกันการย้อนกลับของแรงดันไว้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **+AREF** เป็นขาสำหรับรับสัญญาณแรงดันอ้างอิง (Analog Reference) ให้กับวงจร แอนะล็อคอินพุต ในกรณีต้องการใช้แรงดันอ้างอิงจากภายนอก
- **RESET#** เป็นขาสัญญาณ RESET ของ CPU ทำงานที่ Logic "0"
- **PB[0..5]** เป็นขา I/O แบบดิจิตอล สามารถใช้งานเชื่อมต่อกับสัญญาณ Logic TTL (5V) ต่างๆ
- **PD[0..7]** เป็นขา I/O แบบดิจิตอล สามารถใช้งานเชื่อมต่อกับสัญญาณ Logic TTL (5V) ต่างๆ
- **PC[0..5]** เป็นขา I/O ซึ่งสามารถกำหนดให้เป็นได้ทั้ง ดิจิตอลI และ แอนะล็อคอินพุต
- **ADC6,ADC7** เป็นขาอินพุต แบบแอนะล็อค สามารถรับอินพุต แบบแอนะล็อค 0..+5V

**การพัฒนาโปรแกรมของบอร์ด ET-EASY168 STAMP**

ในการพัฒนาโปรแกรมของบอร์ด ET-EASY168 STAMP นั้น ผู้ใช้สามารถเลือกใช้วิธีการพัฒนาโปรแกรมได้ 2 รูปแบบด้วยกัน คือ

- **AVR Micro Controller** เป็นการพัฒนาโปรแกรมตามรูปแบบของ AVR Micro Controller ปรกติซึ่งสามารถเลือกใช้โปรแกรมภาษาใดๆก็ได้ที่รองรับการใช้งานร่วมกับ AVR เบอร์ ATmega168 ซึ่งผู้ใช้สามารถเลือกใช้โปรแกรมในการพัฒนาได้ตามความถนัด เช่น ภาษาเบสิก BASCOM-AVRหรือ ภาษาซี เช่น Code Vision และ WinAVR เป็นต้น
- **Arduino Project** เป็นการพัฒนาโปรแกรมโดยใช้โปรแกรมและชุดคำสั่งในการเขียนโปรแกรมด้วยภาษาซี (C++) ของ "Arduino Project" ซึ่งเป็นโครงการพัฒนาไมโครคอนโทรลเลอร์ AVR แบบOpen Source ซึ่งกำลังได้รับความนิยมอย่างแพร่หลาย เนื่องจากเป็นโครงการที่เปิดเผยทั้งSource Code ในการพัฒนาให้ทั้งหมดและยังมีตัวอย่างโครงงานพร้อมตัวอย่างโปรแกรมการทดลองต่างๆแจกจ่ายให้ผู้สนใจนำมาใช้ศึกษา เรียนรู้และทดลอง ได้ฟรี โดยไม่เสียค่าใช้จ่ายใดๆ ผู้สนใจสามารถเข้าไปค้นหารายละเอียดต่างๆของ Arduino Project นี้ได้ http://www.arduino.cc/

**การพัฒนาโปรแกรมของ ET-EASY168 STAMP แบบ Arduino Project**

ตามปรกติแล้วบอร์ด ET-EASY168 STAMP จะทำการ ติดตั้งโปรแกรม Bootloader ไว้ให้เรียบร้อยแล้ว โดยใช้ Bootloader ชื่อ "BOOT_EASY168_AUTO_16MHZ.HEX" ซึ่งเป็น Bootloader ที่ทางทีมงาน อีทีที ได้นำต้นฉบับจาก Arduino มาปรับปรุงเงื่อนไขการทำงานใหม่ เพื่อให้การทำงานสอดคล้องกับระบบฮาร์ดแวร์ของบอร์ด ET-EASY168 STAMP ได้ดียิ่งขึ้น โดยโปรแกรม Bootloader นี้จะใช้สำหรับติดต่อสื่อสารเพื่อสั่ง Upload Code จากคอมพิวเตอร์ PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้กับ MCU ในบอร์ดทำงาน โดยไม่ต้องใช้เครื่องโปรแกรมภายนอกให้ยุ่งยาก ซึ่ง คุณสมบัติของ Bootloader ที่ทาง อีทีที ปรับปรุงเพิ่มเติมขึ้น มีคุณสมบัติการทำงานเป็นดังนี้

- สื่อสารกับโปรแกรมภายนอกด้วย Protocol แบบ STK500 (STK500V1)
- ใช้ความเร็ว Baudrate 19200 โดยใช้ความถี่ XTAL 16 MHz
- โปรแกรม Bootloader มีขนาด 2KByte ทำงานที่ตำแหน่ง 0x3800-0x3FFF
- ใช้ LED ที่ต่อกับขา Digital-13(PB5) เป็นตัวแสดงสถานะในขณะที่ Bootloader ทำงาน
- โปรแกรมใน Bootloader จะทำงานโดยอัตโนมัติทุกครั้งหลังการรีเซ็ต โดย MCU จะเริ่มต้นทำงานใน Bootloader นี้ก่อนเสมอ ซึ่งถ้าไม่มีการติดต่อสื่อสารจากโปรแกรม Arduino ภายในเวลาประมาณ 3 วินาที MCU ก็จะออกจากการทำงานใน Bootloader เพื่อไปเริ่มต้นทำงานตามคำสั่งที่เป็นของผู้ใช้ โดยอัตโนมัติ โดยในขณะที่ Bootloader ทำงาน ตอนเริ่มต้นจะเห็น LED ที่ต่อไว้กับขาสัญญาณ Digital-13(PB5) กระพริบ 3 ครั้ง แล้วติดค้าง เพื่อรอการติดต่อสื่อสารจากโปรแกรมสำหรับสั่งให้ทำการ Upload Code ให้กับ MCU ซึ่งถ้าไม่มีการติดต่อสื่อสารจากโปรแกรมของ Arduino ภายในระยะเวลาประมาณ 3 วินาที โปรแกรมก็จะกระโดดไปทำงานในตำแหน่งเริ่มต้นที่เป็นส่วนของโปรแกรมที่ผู้ใช้เขียนขึ้นทันที

นอกจากนี้แล้วผู้ใช้ยังสามารถเปลี่ยนไปใช้โปรแกรมของ Bootloader รุ่นมาตรฐาน ตามรูปแบบโปรแกรมของArduinoได้อีกด้วยโดยสามารถเลือกใช้โปแกรมของBootloader ที่มีชื่อ “ATmegaBOOT_168_diecimila.hex”โดยไฟล์ดังกล่าวจะถูกบรรจุไว้ใน Drive และโฟลเดอร์เดียวกันกับที่ผู้ใช้ซึ่งได้ทำการติดตั้งโปรแกรมของArduinoไว้แล้วคือ บรรจุที่ Dirve “C:\arduino0012\hardware\bootloaders\atmega168\ATmegaBOOT_168_diecimila.hex”

แต่อย่างไรก็ตามในการที่จะสามารถทำการติดตั้งโปรแกรม Bootloader Code ให้กับ MCU ได้นั้นผู้ใช้จำเป็นต้องมีเครื่องมือสำหรับทำหน้าที่ Program Code ให้กับ MCU อยู่ด้วย โดยใช้เครื่องโปรแกรมที่มีขั้วต่อตามมาตรฐานของ “AVRISP” ของ ATMEL แบบ IDE 10 PIN ได้ทันที โดยวิธีการให้ศึกษาเพิ่มเติมจากหัวข้อ “การโปรแกรม Bootloader ให้กับบอร์ด ET-EASY168 STAMP” ในตอนท้าย

**การพัฒนาโปรแกรมของ ET-EASY168 STAMP แบบ AVR Microcontroller**

ในกรณีที่ผู้ใช้ต้องการที่จะพัฒนาโปรแกรมให้กับบอร์ด ET-EASY168 STAMP แบบปรกติ ในรูปแบบของไมโครคอนโทรลเลอร์ AVR ก็สามารถทำได้ โดยสามารถเลือกภาษาในการเขียนโปรแกรม ได้เองซึ่งในกรณีนี้ผู้ใช้สามารถเลือกใช้ภาษาใดๆก็ได้ที่รองรับการใช้งานกับ MCU ตระกูล AVR เบอร์ATmega168 โดย ผู้ใช้งานสามารถบริหารจัดการระบบทรัพยากรต่างๆที่อยู่ภายในตัว ATmega168 ได้เองทั้งหมด โดยสามารถทำได้ 2 แนวทางคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

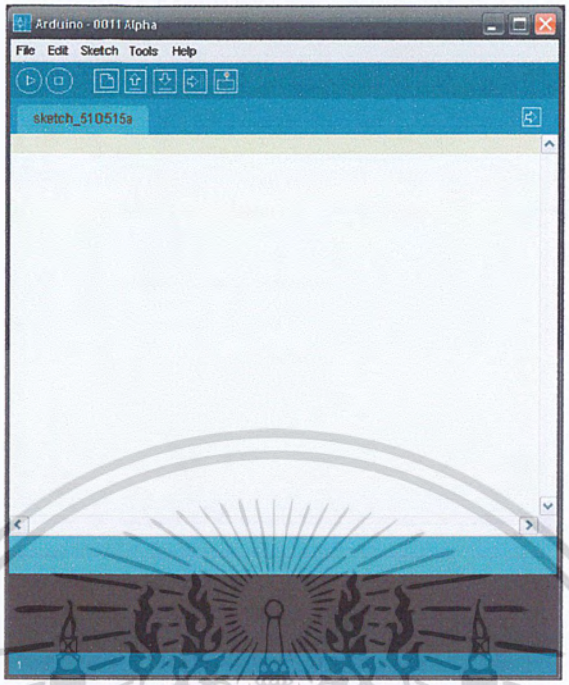
- **การพัฒนาโปรแกรมโดยใช้เครื่องโปรแกรมภายนอก** ในกรณีนี้จะมีข้อดีคือ ไม่สูญเสียทรัพยากรใดๆเลย ผู้ใช้สามารถใช้งานและกำหนดคุณสมบัติการทำงานของทรัพยากรต่างๆที่มีอยู่ใน MCU ได้เอง ตามต้องการ แต่มีข้อเสียคือ ต้องมีเครื่องโปรแกรมภายนอก สำหรับทำหน้าที่โปรแกรม Code ให้กับ MCU โดยสามารถใช้ได้กับเครื่องโปรแกรมทุกรุ่นที่ รองรับการใช้งานกับ MCU เบอร์ ATmega168 และมีขั้วต่อตามมาตรฐาน AVRISP ของ ATMEL
- **การพัฒนาโปรแกรมโดยใช้งานร่วมกับ Bootloader** ในกรณีนี้จะมีข้อดี คือ เมื่อทำการเขียนโปรแกรมเสร็จแล้วสามารถสั่ง Program Code ให้กับ MCU ผ่านทางโปรแกรม Bootloader ที่ติดตั้งไว้แล้วได้ทันที โดยไม่ต้องใช้เครื่องโปรแกรมภายนอก แต่มีข้อจำกัด คือ ต้องเสียพื้นที่หน่วยความจำ Flash สำหรับเก็บ Code โปรแกรมไป 2KByte (0x3800 -0x3FFF) เพื่อใช้ติดตั้งโปรแกรม Bootloader ซึ่งตามปรกติแล้วบอร์ด ET-EASY168 STAMP จะทำการ ติดตั้งโปรแกรม Bootloader ไปให้เรียบร้อยแล้ว โดยในกรณีนี้ผู้ใช้จะสามารถเขียนโปรแกรมใช้งานได้ทั้งหมดจำนวน 14 Kbyte จากที่มีอยู่ทั้งหมด 16Kbyte โดยต้องเขียนโปรแกรมให้มีตำแหน่งการทำงานของ Code อยู่ระหว่างตำแหน่งแอดเดรส 0x0000 ถึง 0x37FF (0x1C00 - 0x1FFF K Word) ส่วน SRAM,EEPROM และทรัพยากรอื่นๆที่มีอยู่ใน MCU ผู้ใช้ยังสามารถใช้งานได้ครบตามจำนวนที่มีอยู่ในตัว MCU ทุกประการ

## 3.6 ส่วนโปรแกรมการควบคุมการทำงานของแขนกล

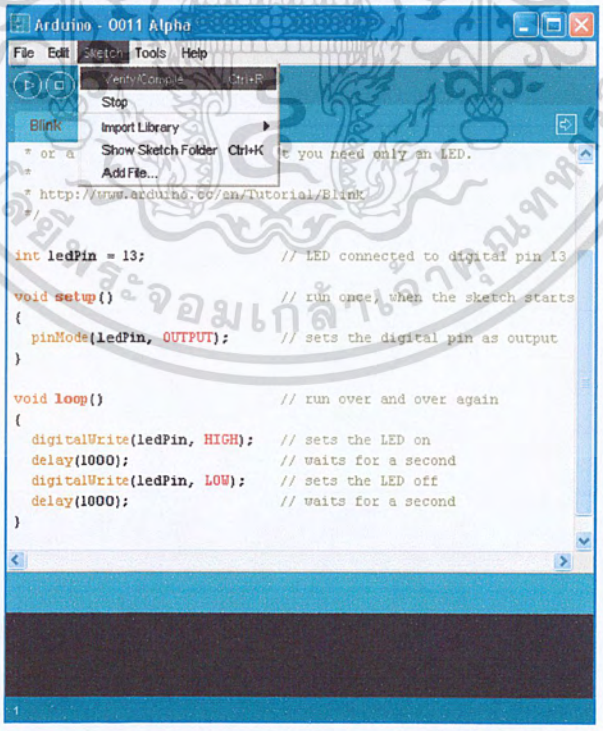
### 3.6.1การใช้งานบอร์ด Arduino ในการควบคุมการทำงาน

1. เตรียมบอร์ด Arduino และสาย USB
2. Download โปรแกรมพัฒนา Arduino
3. ติดตั้ง USB drivers
4. เชื่อมต่อ คอมพิวเตอร์กับ Arduino
5. เรียกโปรแกรมพัฒนา Arduino โปรแกรมจะถูกนำมาใช้งาน
6. เขียนโปรแกรมโดยใช้ภาษาซี
7. Upload โปรแกรมลงบอร์ด Arduino โดยกดปุ่ม Upload to I/O Board

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

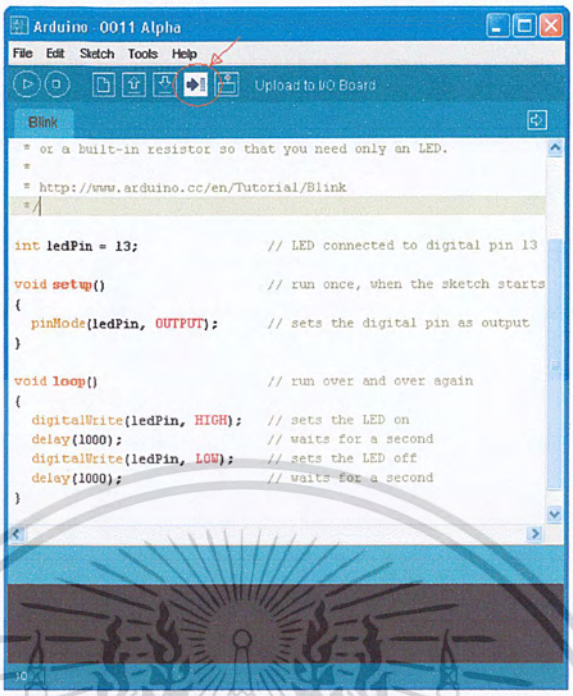



**รูปที่ 3.33** หน้าจอโปรแกรม arduino




**รูปที่ 3.34** การเขียนโปรแกรมโดยใช้ภาษาซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 3.35** การ Upload โปรแกรมลงบอร์ด Arduino โดยกดปุ่ม Upload to I/O Board

### 3.6.2 ส่วนประกอบของไมโครคอนโทรเลอร์

**1.หน่วยประมวลผล (Central Processor Unit)** เป็นส่วนที่ตัดสินใจเกี่ยวกับการทำงานต่างๆ ซึ่งจะทำงานตามโปรแกรมที่เราเขียนและอัดเข้าไปในตัวไมโครคอนโทรลเลอร์

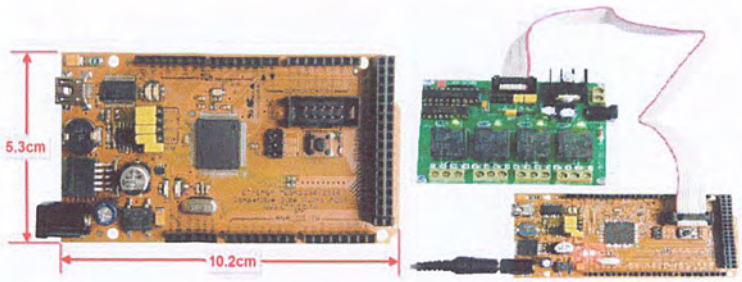
**2.หน่วยความจำ (Memory)** เป็นตัวที่จะเก็บข้อมูลต่างๆ ที่ต้องใช้ในไมโครคอนโทรลเลอร์ ไม่ว่าจะเป็นหน่วยความจำ โปรแกรมหรือหน่วยความจำข้อมูล โดยหน่วยความจำที่ใช้ได้แก่ Rom, EPROM, EEPROM,RAM รวมทั้งหน่วยความจำ FLASH

**3.พอร์ทสัญญาณเข้าและสัญญาณออก** เป็นส่วนที่จะใช้ในการติดต่อกับอุปกรณ์ภายนอก

**4.คุณสมบัติอื่นๆ** ไมโครคอนโทรลเลอร์สมัยใหม่ จะมีฟังก์ชั่นพิเศษเพิ่มเติม เช่น Time, Counter ,Analog to Digital Converter , Analog Comparator ,UART และUSART

### 3.6.3 ไมโครคอนโทรลเลอร์ตระกูล AVR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 3.36** ตัวอย่างการต่อบอร์ด I/O ของ อีทีที ด้วยขั้ว 10Pin

มีการปรับปรุงโปรแกรมให้ใช้กับชิพ AVR รุ่นใหญ่ขึ้น เพื่อให้มีจำนวน I/O ทั้ง Digital, Analog, PWM, UART และ ขนาดหน่วยความจำเพิ่มมากขึ้นกว่าเดิม

**คุณสมบัติของบอร์ด**

- ใช้ ATMEGA1280 เป็น MCU ประจำบอร์ด Run ความถี่ 16MHz จาก Crystal Oscillator
- 128KByte Flash (สงวนไว้ 4KByte สำหรับ Bootloader) / 8KByte SRAM / 4KByteEEPROM
- รองรับการพัฒนาโปรแกรมด้วยภาษา C++ ของArduino ตามแบบArduino Mega ได้100%
- ใช้ USB Bridge ของ FTDI เบอร์ FT232RL พร้อม Over Current Protection สำหรับติดต่อสื่อสารและ Download Code จากคอมพิวเตอร์ให้บอร์ด พร้อม Jumper สำหรับปรับใช้งานบอร์ดเป็นการProgram Bootloader ให้กับ MCU จากพอร์ต USB ในบอร์ดได้เอง โดยไม่ต้องใช้เครื่องโปรแกรมAVRISP 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 หรือ บอร์ด I/O แบบต่างๆ ของ อีทีที เพิ่มความสะดวกในการใช้งาน
- รองรับการใช้งานกับ External Supply ทั้งแบบ AC และ DC ขนาด 7-12V โดยเลือกใช้ Regulate แบบ Switching ขนาด 1A (LM2575-5V) ลดปัญหาเรื่องความร้อนเมื่อมีการใช้กระแสสูงๆ สามารถใช้แหล่งจ่ายจากพอร์ต USB ได้ในกรณีใช้กระแสไม่เกิน 500mA โดยมีวงจรเลือกแหล่งจ่ายอัตโนมัติโดยจะตัดการใช้ไฟเลี้ยงจาก USB โดยอัตโนมัติ เมื่อมีการต่อแหล่งจ่ายจากภายนอกให้บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 4

# ผลการทดลอง

ในบทนี้ จะกล่าวถึงผลการทดลอง โครงสร้างของแขนกล โปรแกรมของแขนกล และการทำงานของ เซอร์โวมอเตอร์โดยมีรายละเอียดของการทดลองดังนี้

## 4.1 การทดลอง Servo Motor ของแขนกล

จากการทดลองเมื่อทำการจ่ายไฟให้กับวงจร ตัวแขนกลจะทำการเซ็ตความพัลส์ให้อยู่ในช่วงกึ่งกลางของค่าสูงสุดและต่ำสุดของพัลส์บวก ที่เซอร์โวมอเตอร์ทำให้เซอร์โวมอเตอร์สามารถทำงานได้ เมื่อทำการคอลโทรล โดยกดปุ่มที่ปุ่มควบคุม ค่าพัลส์ก็จะเพิ่มค่าหรือลดค่าตามที่กำหนดไว้ในวงจรสั่งงานผ่านพอร์ต

**ตารางที่ 4.1** ตารางการทดลองของเซอร์โวมอเตอร์ เรื่องความกว้างของพัลส์ในแต่ละตำปหน่งของแขนกล

| ตำแหน่ง | สถานะ | พัลส์บวก (ms) | พัลส์ลบ (ms) | ความกว้างของพัลส์ (ms) |
|---|---|---|---|---|
| S1 ฐาน | Start | 2 | 25 | 27 |
| | Min | 1 | 24 | 27 |
| | Max | 3 | 22 | 27 |
| S2 ท่อนแขน | Start | 1 | 26 | 27 |
| | Min | 0.5 | 25.5 | 27 |
| | Max | 2 | 24 | 27 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| ตำแหน่ง | สถานะ | พัลส์บวก (ms) | พัลส์ลบ (ms) | ความกว้างของพัลส์ (ms) |
|---|---|---|---|---|
| S3 ท่อนแขน | Start | 2 | 25 | 27 |
| | Min | 1 | 24 | 27 |
| | Max | 3 | 22 | 27 |
| S4 ท่อนแขน | Start | 1.5 | 25.5 | 27 |
| | Min | 1 | 24 | 27 |
| | Max | 2.5 | 24.5 | 27 |
| S5 ท่อนแขน | Start | 2 | 25 | 27 |
| | Min | 1 | 26 | 27 |
| | Max | 3 | 24 | 27 |
| S2 Clipper | Start | 1 | 26 | 27 |
| | Min | 1 | 26 | 27 |
| | Max | 2 | 25 | 27 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5
# บทวิจารณ์และสรุป

## 5.1 สรุปผลการทดลอง

โครงงานนี้เป็นการพัฒนาแขนกล คือ คือหุ่นยนต์ที่สามารถหยิบจับวัตถุขึ้นและวางวัตถุลงได้ เพื่อนำไปใช้ในระบบอุตสาหกรรมในการเพิ่มผลผลิต ลดจำนวนแรงงานคน และเพิ่มความปลอดภัย ซึ่งจะเป็นการเพิ่มศักยภาพในการแข่งขันต่อคู่แข่งทางการด้านอุตสาหกรรมต่อไป ซึ่งจากโครงงานนี้ได้ประดิษฐ์แขนกลขึ้นมานั้น สามารถหยิบจับและวางวัตถุลงบนรถขนส่งได้ แต่ยังไม่แข็งแรงและแม่นยำพอที่จะสามารถนำไปใช้ในส่วนอุตสาหกรรมได้จริง จะต้องมีการแก้ไขปรับปรุง และพัฒนาต่อไปในอนาคต

## 5.2 ปัญหาที่พบและแนวทางแก้ไข

### 5.2.1 การทดลองโครงสร้างของแขนกล

จากการทดลองที่ผ่านมาตั้งแต่การออกแบบโครงสร้างของแขนกล และการประกอบโครงสร้างในช่วงแรกเราได้ทดลองออกแบบ และทำแขนกลออกมาก่อน 1 ตัว ผลปรากฏว่า แขนกลไม่สามารถขยับได้ตามที่เราต้องการ และการบังคับทิศทางไม่ดีเท่าที่ควร เพราะขนาดไม่เหมาะสมทำให้ไม่สามารถจับวัตถุได้ และขยับได้ตามต้องการ เมื่อได้แก้ไขขนาด และโครงสร้างเป็นที่เหมาะสมแล้ว แขนกลก็สามารถขยับได้สมบูรณ์มากขึ้น แต่ระหว่างการทดลองแขนกลมีการชำรุดจากการใช้งานบ่อยครั้ง ทำให้ต้องปรับปรุงแก้ไขโครงสร้าง และประกอบใหม่ให้แข็งแรงมากขึ้น เมื่อได้ปรับปรุงแก้ไขโครงสร้างแล้ว ทำให้โครงสร้างแข็งแรงจนเป็นที่น่าพอใจ และเหมาะสมจนสามาถนำไปใช้งานได้อย่างมีประสิทธิภาพ

### 5.2.2 การทดลองโปรแกรมการควบคุมของแขนกล

ในตอนแรกได้เขียนโปรแกรม ตั้งค่า และบันทึกค่าในตำแหน่งที่ต้องการแล้ว แต่แขนกลไม่สามารถเคลื่อนไหวได้ตามต้องการ ไม่มีความแม่นยำเท่าที่ควร จึงมีการปรับแก้โปรแกรม และเขียนโปรแกรมขึ้นมาใหม่ พร้อมกับต้องตั้งค่าใหม่ด้วย เพื่อให้สามารถทำงานได้มีประสิทธิภาพ แต่เนื่องจากมีการปรับเปลี่ยนโครงสร้าง และแก้ไข ทำให้ต้องมีการตั้งค่า และบันทึกค่าใหม่ แต่เมื่อแก้ไขแล้วแขนกลก็สามารถทำงานได้อย่างมีประสิทธิภาพมากพอกับการใช้งาน เมื่อต่อเข้ากับระบบแล้วสามารถทำงานได้เป็นอย่างดีตามที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.3 ข้อเสนอแนะและแนวทางในการค้นคว้าพัฒนา

จากการที่เราใช้แผ่นอคิลิคเป็นส่วนประกอบของแขนกลซึ่งเป็นวัสดุที่ไม่ค่อยคงคน ในการสร้างส่วนประกอบของแขนกลควรนำเอาวัสดุที่มีความแข็งแรงสูงทนต่อแรงรับที่ถูกกระทำได้เช่น อลูมิเนียม เหล็ก เป็นต้นและเนื่องด้วยเซอร์โวมอเตอร์ที่ใช้ในปัจจุบันมีให้เลือกใช้หลายขนาดและมีราคาแตกต่างกันออกไป เราควรศึกษาเพิ่มเติมว่าควรจะใช้เซอร์โวมอเตอร์ชนิดและขนาดไหนที่จะเหมาะสมกับงบประมานที่เรามีและสามารถทำงานได้เหมาะสมกับความต้องการที่เราจะให้หุ่นยนต์ทำหน้าที่อะไรและนอกจากนั้นควรศึกษาวิธีการทำงาน ท่าทางการเคลื่อนไหวของแขนกล เพื่อเป็นการพัฒนาท่าทางการจับวัตถุของแขนกลในท่าทางที่ง่ายและหลากหลาย เหมาะกับการใช้จริงในโรงงานอุตสาหกรรมต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก ก

# โปรแกรมควบคุมแขนกล

บอร์ด Stamp168 มีไมโครคอนโทรเลอร์(Microcontroller Unit หรือ MCU) ตระกูล AVR8 เบอร์ ATmega168 ของ ATMEL เป็น MCU ประจำบอร์ด มีcodeการเขียนโปรแกรมมีดังนี้

```
#include <Servo.h>
#include <Keypad.h>
#include <EEPROM.h>

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] = {3, 2, 10, 9}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {7, 6, 5, 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 servo1;
Servo servo2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Servo servo3;
Servo servo4;
Servo servo5;
Servo servo6;

//variable
int pos1 = 90;
int pos2 = 45;
int pos3 = 117;
int pos4 = 75;
int pos5 = 73;
int pos6 = 90;

int addr =0;
int addr2 =0;

byte pos11 = 0;
byte pos22 = 0;
byte pos33 = 0;
byte pos44 = 0;
byte pos55 = 0;
byte pos66 = 0;

int val = 0;

void setup()
{
  servo1.attach(8);  // attaches the servo on pin 8 to the servo object
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  servo2.attach(0);  // attaches the servo on pin 0 to the servo object
  servo3.attach(1);  // attaches the servo on pin 1 to the servo object
  servo4.attach(11); // attaches the servo on pin 11 to the servo object
  servo5.attach(12); // attaches the servo on pin 12 to the servo object
  servo6.attach(13); // attaches the servo on pin 13 to the servo object
}

void loop()
{
  char customKey = customKeypad.getKey();
  val = analogRead(1);

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

//ปุ่มจัดท่า
  //1+
     if (customKey == '0')
     {
      pos1 = pos1 + 2;
      if (pos1 > 180)
        {
          pos1 = 180;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
   delay (100);
}
//////
//1-
 if (customKey == '1')
{
   pos1 = pos1 - 2;
   if (pos1 < 0)
     {
       pos1 = 0;
     }
   delay (100);
}
//////
//2+
 if (customKey == '2')
{
  pos2 = pos2 + 2;
  if (pos2 > 180)
    {
      pos2 = 180;
    }
  delay (100);
}
//////
//2-
  if (customKey == '3')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  pos2 = pos2 - 2;
  if (pos2 < 0)
    {
      pos2 = 0;
    }
  delay (100);
 }
//////
//3+
 if (customKey == '4')
{
 pos3 = pos3 + 2;
 if (pos3 > 180)
   {
     pos3 = 180;
   }
 delay (100);
}
//////
//3-
 if (customKey == '5')
{
  pos3 = pos3 - 2;
  if (pos3 < 0)
    {
      pos3 = 0;
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  delay (100);
 }
//////
//4+
 if (customKey == '6')
{
  pos4 = pos4 + 2;
  if (pos4 > 180)
    {
      pos4 = 180;
    }
  delay (100);
}
//////
//4-
 if (customKey == '7')
{
   pos4 = pos4 - 2;
   if (pos4 < 0)
     {
       pos4 = 0;
     }
   delay (100);
 }
//////
//5+
  if (customKey == '8')
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 pos5 = pos5 + 2;
 if (pos5 > 180)
   {
    pos5 = 180;
   }
 delay (100);
}
//////
//5-
 if (customKey == '9')
{
  pos5 = pos5 - 2;
  if (pos5 < 0)
    {
     pos5 = 0;
    }
  delay (100);
 }
//////
//6+
 if (customKey == 'a')
{
 pos6 = pos6 + 2;
 if (pos6 > 180)
   {
    pos6 = 180;
   }
 delay (100);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    //////
    //6-
     if (customKey == 'b')
    {
      pos6 = pos6 - 2;
      if (pos6 < 0)
        {
          pos6 = 0;
        }
      delay (100);
    }

//-------------------------------------------------------------------//

// เก็บค่า //
if (customKey == 'c')
 {
  pos11 = pos1;
  EEPROM.write(addr, pos11);
  addr = addr + 1;

  pos22 = pos2;
  EEPROM.write(addr, pos22);
  addr = addr + 1;

  pos33 = pos3;
  EEPROM.write(addr, pos33);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  addr = addr + 1;


  pos44 = pos4;
  EEPROM.write(addr, pos44);
  addr = addr + 1;


  pos55 = pos5;
  EEPROM.write(addr, pos55);
  addr = addr + 1;


  pos66 = pos6;
  EEPROM.write(addr, pos66);
  addr = addr + 1;


  if (addr == 512) // addr maximun size : 512 byte //
    {
      addr = 0;
    }


 delay(500);  // dalay เพื่อป้องกันการกดปุ่มซ้ำ อาจทำให้จำค่าผิดได้ //
}


//-------------------------------------------------------------------//


// อ่านค่า //
 if (customKey == 'e')
 {
   pos11 = EEPROM.read(addr-6);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    pos1 = pos11;


    pos22 = EEPROM.read(addr-5);
    pos2 = pos22;


    pos33 = EEPROM.read(addr-4);
    pos3 = pos33;


    pos44 = EEPROM.read(addr-3);
    pos4 = pos44;


    pos55 = EEPROM.read(addr-2);
    pos5 = pos55;


    pos66 = EEPROM.read(addr-1);
    pos6 = pos66;


    if (addr == 512)
     {
       addr = 0;
     }


    delay(500);
  }
//--------------------------------------------------------------------------//


//cycle
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (customKey == 'f')
{
 addr2 = 0;
 for (int y=1;y<=8;y++)
 {
    pos11 = EEPROM.read(addr2);
    addr2 = addr2 + 1;

    pos22 = EEPROM.read(addr2);
    addr2 = addr2 + 1;

    pos33 = EEPROM.read(addr2);
    addr2 = addr2 + 1;

    pos44 = EEPROM.read(addr2);
    addr2 = addr2 + 1;

    pos55 = EEPROM.read(addr2);
    addr2 = addr2 + 1;

    pos66 = EEPROM.read(addr2);
    addr2 = addr2 + 1;

    int old1 = pos1;
    int old2 = pos2;
    int old3 = pos3;
    int old4 = pos4;
    int old5 = pos5;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int old6 = pos6;


int new1 = pos11;
int new2 = pos22;
int new3 = pos33;
int new4 = pos44;
int new5 = pos55;
int new6 = pos66;


if (old1 > new1)
{
 for(old1 ; old1 > new1 ; old1 -= 1)
 {
    servo1.write(old1);
    delay(15);
 }
}
else (old1 < new1);
{
 for(old1 ; old1 < new1 ; old1 += 1)
 {
    servo1.write(old1);
    delay(15);
 }
}
    pos1 = new1;
    if (old2 > new2)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 for(old2 ; old2 > new2 ; old2 -= 1)
 {
   servo2.write(old2);
   delay(15);
 }
}
else (old2 < new2);
{
 for(old2 ; old2 < new2 ; old2 += 1)
 {
   servo2.write(old2);
   delay(15);
 }
}
   pos2 = new2;

if (old3 > new3)
{
 for(old3 ; old3 > new3 ; old3 -= 1)
 {
   servo3.write(old3);
   delay(15);
 }
}
else (old3 < new3);
{
 for(old3 ; old3 < new3 ; old3 += 1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
     servo3.write(old3);
     delay(15);
  }
}
     pos3 = new3;

if (old4 > new4)
{
  for(old4 ; old4 > new4 ; old4 -= 1)
  {
     servo4.write(old4);
     delay(15);
  }
}
else (old4 < new4);
{
  for(old4 ; old4 < new4 ; old4 += 1)
  {
     servo4.write(old4);
     delay(15);
  }
}
     pos4 = new4;

if (old5 > new5)
{
     for(old5 ; old5 > new5 ; old5 -= 1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
     servo5.write(old5);
     delay(15);
  }
 }
 else (old5 < new5);
 {
  for(old5 ; old5 < new5 ; old5 += 1)
  {
     servo5.write(old5);
     delay(15);
  }
 }
     pos5 = new5;

 if (old6 > new6)
 {
  for(old6 ; old6 > new6 ; old6 -= 1)
  {
     servo6.write(old6);
     delay(15);
  }
 }
 else (old6 < new6);
 {
  for(old6 ; old6 < new6 ; old6 += 1)
  {
     servo6.write(old6);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            delay(15);
          }
        }
            pos6 = new6;

    }
     if (addr2 > 47)
       {
         addr2 = 0;
       }
  }

  //-------------------------------------------------------------------------------
  //AUTO

  if (val >= 900)
 {
  addr2 = 0;
  for (int y=1;y<=8;y++)
  {
     pos11 = EEPROM.read(addr2);
     addr2 = addr2 + 1;

     pos22 = EEPROM.read(addr2);
     addr2 = addr2 + 1;

     pos33 = EEPROM.read(addr2);
     addr2 = addr2 + 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
pos44 = EEPROM.read(addr2);
addr2 = addr2 + 1;

pos55 = EEPROM.read(addr2);
addr2 = addr2 + 1;

pos66 = EEPROM.read(addr2);
addr2 = addr2 + 1;

int old1 = pos1;
int old2 = pos2;
int old3 = pos3;
int old4 = pos4;
int old5 = pos5;
int old6 = pos6;

int new1 = pos11;
int new2 = pos22;
int new3 = pos33;
int new4 = pos44;
int new5 = pos55;
int new6 = pos66;

if (old1 > new1)
{
  for(old1 ; old1 > new1 ; old1 -= 1)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    servo1.write(old1);
    delay(15);
  }
}
else (old1 < new1);
{
  for(old1 ; old1 < new1 ; old1 += 1)
  {
    servo1.write(old1);
    delay(15);
  }
}
    pos1 = new1;

if (old2 > new2)
{
  for(old2 ; old2 > new2 ; old2 -= 1)
  {
    servo2.write(old2);
    delay(15);
  }
}
else (old2 < new2);
{
  for(old2 ; old2 < new2 ; old2 += 1)
  {
    servo2.write(old2);
    delay(15);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
}
    pos2 = new2;

if (old3 > new3)
{
  for(old3 ; old3 > new3 ; old3 -= 1)
  {
    servo3.write(old3);
    delay(15);
  }
}
else (old3 < new3);
{
  for(old3 ; old3 < new3 ; old3 += 1)
  {
    servo3.write(old3);
    delay(15);
  }
}
    pos3 = new3;

if (old4 > new4)
{
  for(old4 ; old4 > new4 ; old4 -= 1)
  {
    servo4.write(old4);
    delay(15);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
}
else (old4 < new4);
{
  for(old4 ; old4 < new4 ; old4 += 1)
  {
     servo4.write(old4);
     delay(15);
  }
}
     pos4 = new4;

if (old5 > new5)
{
  for(old5 ; old5 > new5 ; old5 -= 1)
  {
     servo5.write(old5);
     delay(15);
  }
}
else (old5 < new5);
{
  for(old5 ; old5 < new5 ; old5 += 1)
  {
     servo5.write(old5);
     delay(15);
  }
}
```

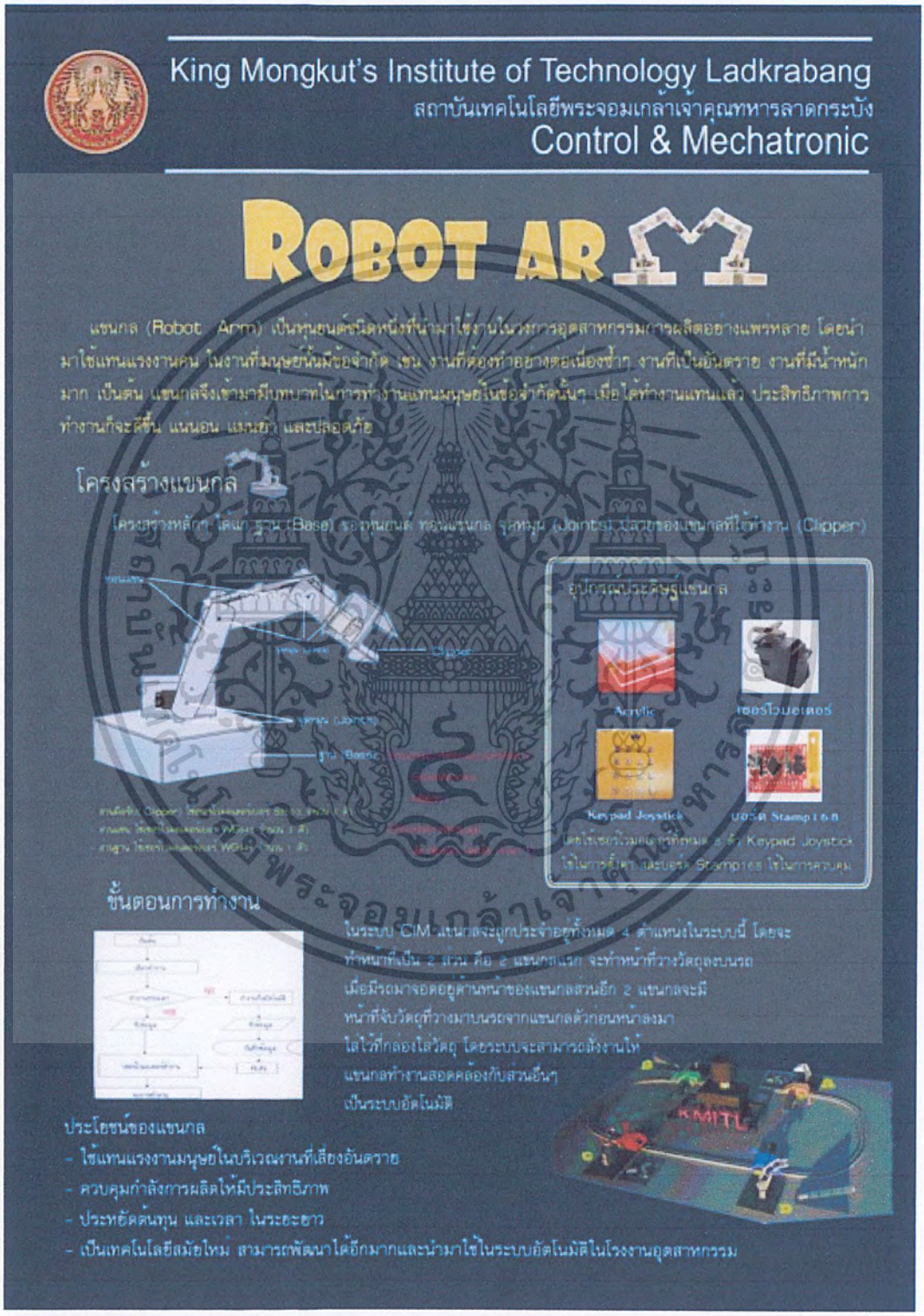
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        pos5 = new5;


    if (old6 > new6)
    {
     for(old6 ; old6 > new6 ; old6 -= 1)
     {
        servo6.write(old6);
        delay(15);
     }
    }
    else (old6 < new6);
    {
     for(old6 ; old6 < new6 ; old6 += 1)
     {
        servo6.write(old6);
        delay(15);
     }
    }
        pos6 = new6;


  }
   if (addr2 > 47)
    {
      addr2 = 0;
    }
 }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก ข

# โปสเตอร์แขนกล




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**รูปที่ ข.1** โปสเตอร์แขนกล

# เอกสารอ้างอิง

[1] **Mitchell, F.H.** CIM systems : an introduction to computer-integrated manufacturing / **F.H. Mitchell. Englewood Cliffs, NJ. Prentice-Hall International. c1991.**

[2] **เดชฤทธิ์ มณีธรรม. คัมภีร์หุ่นยนต์. กรุงเทพมหานคร : วี.ซี.พี. ซัคเซสกรุ๊ป, หจก. 2549.**

[3] **ธนพงศ์ เสนแสง, นพพร เชื่อมสุวรรณ. "แขนกล." ปริญญานิพนธ์วิศวกรรมศาสตร บัณฑิต สาขาวิศวกรรมระบบควบคุม คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2548.**

[4] **อีทีที. เรียนรู้เข้าใจใช้งานไมโครคอนโทรลเลอร์ตระกูล AVR ด้วย Arduino. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร. อีทีที จำกัด. 2553.**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้