สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เก้าอี้นวดไฟฟ้าที่สามารถปรับลักษณะการทำงานได้

PROGRAMMABLE MASSAGE CHAIR

โดย

นายกวิณ บุณยสมิต

นายเตวิช พันธุศักดิ์

อาจารย์ที่ปรึกษา

ผศ.ดร.อุนนัต พิณโสภณ

เลขหมู่.....
เลขทะเบียน.....62392
วัน,เดือน,ปี.....17 ส.ค. 2549

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมบัณฑิต

ภาควิชาวิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2548

ภาควิชาวิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เก้าอี้นวดไฟฟ้าที่สามารถปรับลักษณะการทำงานได้

PROGRAMMABLE MASSAGE CHAIR

ผู้จัดทำ

1. นายกวิณ บุณยสมิต รหัสประจำตัว 45010029
2. นายเตวิช พันธุศักดิ์ รหัสประจำตัว 45010283

อาจารย์ที่ปรึกษา

(ผศ.ดร.อุนนัต พิณโสภณ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เก้าอี้นวดไฟฟ้าที่สามารถปรับลักษณะการทำงานได้

นายกวิณ บุณยสมิต รหัสประจำตัว 45010029
นายเตวิช พันธุศักดิ์ รหัสประจำตัว 45010283
ผศ.ดร.อุนนัต พิณโสภณ อาจารย์ที่ปรึกษา

บทคัดย่อ

โครงงานนี้จัดทำขึ้นเพื่อออกแบบ และสร้างเก้าอี้นวดไฟฟ้าที่สามารถปรับลักษณะการทำงานได้ โดยการเคลื่อนที่ของลูกนวดของเก้าอี้นวดไฟฟ้าสามารถเคลื่อนที่ได้ในสองแนวแกนพิกัด ระบบควบคุม และโปรแกรมควบคุมการทำงานของเก้าอี้นวดไฟฟ้าจะมีลักษณะโครงสร้างแบบสถาปัตยกรรมเปิด เก้าอี้นวดไฟฟ้าที่ออกแบบ และสร้างในโครงงานนี้จึงมีขีดความสามารถที่จะปรับลักษณะการทำงานให้มีความเหมาะสมกับสรีระของผู้ใช้ และยังสามารถปรับปรุง เปลี่ยน หรือเพิ่มโปรแกรมลักษณะการทำงานได้ตลอดอายุการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROGRAMMABLE MASSAGE CHAIR

Mr. Kavin Punyasmita Student ID 45010029
Mr. Tewit Pantusak Student ID 45010283
Asst.Prof.Dr. Unnat Pinsopon Adviser

ABSTRACT

The purpose of this project is to design and fabricate a programmable massage chair. The massage balls that attached to the arms can move in two-axis direction. The control system and program are designed via the open control architecture. Thus this massage chair can be adjusted for a suitable performance according to the user's bodies. Further more, it can be improved, changed or added the new programs for the life time of the chair.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ได้สำเร็จลุล่วงไปด้วยดี เนื่องจากได้รับความช่วยเหลือและความเมตตาจาก ผศ.ดร. อุนนัต พิณโสภณ อาจารย์ที่ปรึกษาโครงงาน ที่ได้ให้คำแนะนำ ข้อคิดเห็นต่างๆ คณะผู้จัดทำขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบคุณอาจารย์มณฑา เทียมเมือง และพี่ธวัชชัย ศรีกาหลง ที่ให้คำปรึกษา แนะนำวิธีการในการสร้างส่วนประกอบทางกลต่างๆของเก้าอี้นวด

ขอขอบคุณพี่อุกฤษฎ์ หมื่นจิตร ที่ช่วยเป็นที่ปรึกษาในด้านการเขียนโปรแกรมควบคุม และการสร้างแผงวงจรควบคุมเก้าอี้นวด

สุดท้ายนี้ทางคณะผู้จัดทำขอกราบขอบพระคุณ บิดา มารดา อันเป็นที่เคารพรัก ที่ได้ให้กำลังใจและให้ความช่วยเหลือทางการเงินเป็นอย่างดีตลอดมาจนปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี

คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้คณะผู้จัดทำขอมอบแด่ผู้มีพระคุณทุกท่าน

ควิณ บุณยสมิต
เตวิช พันธุศักดิ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้าที่

บทคัดย่อภาษาไทย		I
บทคัดย่อภาษาอังกฤษ		II
กิตติกรรมประกาศ		III
สารบัญ		IV
สารบัญตาราง		V
สารบัญรูปภาพ		VI
บทที่ 1	บทนำ	
	1.1 ความสำคัญและที่มา	1
	1.2 วัตถุประสงค์	1
	1.3 ขอบเขตของโครงงาน	1
	1.4 ขั้นตอนการดำเนินโครงงาน	1
บทที่ 2	ทฤษฎีและหลักการ	
	2.1 อุปกรณ์ที่ใช้ในเก้าอี้นวดไฟฟ้า	2
	2.2 ไมโครคอนโทรลเลอร์ MCS-51	10
	2.3 โปรแกรมภาษาซี	16
บทที่ 3	การออกแบบโครงสร้างทางกลของเก้าอี้นวดไฟฟ้า	
	3.1 ส่วนประกอบของเก้าอี้นวดไฟฟ้าที่เคลื่อนที่ในแนวนอน	29
	3.2 ส่วนประกอบของเก้าอี้นวดไฟฟ้าที่เคลื่อนที่ในแนวตั้ง	34
บทที่ 4	การออกแบบระบบควบคุมและโปรแกรมควบคุมการทำงาน	
	4.1 การออกแบบวงจรไฟฟ้า	35
	4.2 การออกแบบโปรแกรมการสั่งงาน	39
	4.3 การใช้งานเก้าอี้นวดไฟฟ้า	41
บทที่ 5	รูปแบบและผลการทดลอง	
	5.1 รูปแบบการทดลอง	46
	5.2 ผลการทดลอง	46
บทที่ 6	สรุปผลการดำเนินงาน	
	6.1 สรุปผลการดำเนินงาน	48
	6.2 ปัญหาที่พบขณะดำเนินงานและแนวทางแก้ไข	48
บรรณานุกรม		49
ภาคผนวก ก	ซอร์สโค้ดภาษาซี	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้าที่

ตารางที่ 2-1 แสดงถึงคุณสมบัติของสายพานชนิดต่างๆ 2
ตารางที่ 2-2 แสดงถึงข้อมูลรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทม์เมอร์/เคาน์เตอร์ 0 ทำงานเป็นไทม์เมอร์ 22
ตารางที่ 2-3 แสดงถึงข้อมูลรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทม์เมอร์/เคาน์เตอร์ 0 ทำงานเป็นเคาน์เตอร์ 22
ตารางที่ 2-4 แสดงถึงข้อมูลรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทม์เมอร์/เคาน์เตอร์ 1 ทำงานเป็นไทม์เมอร์ 23
ตารางที่ 2-5 แสดงถึงข้อมูลรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทม์เมอร์/เคาน์เตอร์ 1 ทำงานเป็นเคาน์เตอร์ 23
ตารางที่ 2-6 แสดงถึงความสัมพันธ์ของรีจิสเตอร์ IP และ IPH 26
ตารางที่ 3-1 แสดงถึงค่าตัวประกอบความล้า 32
ตารางที่ 4-1 แสดงถึงน้ำหนักเฉลี่ยของผู้ใหญ่ไทยปกติในช่วงความสูงต่างๆ 40
ตารางที่ 4-2 แสดงถึงการหาค่าเฉลี่ยความกว้างของแผ่นหลังที่ส่วนสูงและน้ำหนักต่างๆ 40
ตารางที่ 5-1 แสดงถึงระยะเวลาเฉลี่ยในการเคลื่อนที่ของชุดนวดที่ภาระต่างๆ 46
ตารางที่ 5-2 แสดงถึงระยะการเคลื่อนที่ของลูกนวดในแบบบังคับอัตโนมัติ 47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้าที่

รูปที่ 2-1 แสดงถึงการติดตั้งสายพานแบบเปิด 3
รูปที่ 2-2 แสดงถึงสายพานที่มีด้านหย่อนและตึง 4
รูปที่ 2-3 แสดงถึงส่วนประกอบภายในของแบริ่งแบบกลิ้งหรือตลับลูกปืน 5
รูปที่ 2-4 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลมร่องลึก 5
รูปที่ 2-5 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลมร่องลึกมีรอยบากเติมเม็ดกลม 6
รูปที่ 2-6 แสดงถึงลักษณะภายในของแมกนีโตแบริ่ง 6
รูปที่ 2-7 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลมสัมผัสเชิงมุม 6
รูปที่ 2-8 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลมปรับแนวแกนได้เอง 7
รูปที่ 2-9 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลมกันรุน 7
รูปที่ 2-10 แสดงถึงทิศทางของแรงกระทำบนเกลียวของสกรู 8
รูปที่ 2-11 แสดงถึงมอเตอร์กระแสตรงที่ใช้ในโครงงานนี้ 9
รูปที่ 2-12 แสดงถึงมอเตอร์กระแสสลับที่ใช้ในโครงงานนี้ 9
รูปที่ 2-13 แสดงถึงเพลาพร้อมล้อสายพาน 10
รูปที่ 2-14 แสดงถึงโครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 อนุกรม P89V51 12
รูปที่ 2-15 แสดงถึงการจัดขามาตรฐานของไมโครคอนโทรลเลอร์ MCS-51 เบอร์ P89V51RD2 14
รูปที่ 2-16 แสดงถึงไดอะแกรมการทำงานในโหมด 0 ของไทม์เมอร์/เคาน์เตอร์ 0 หรือ 1 20
รูปที่ 2-17 แสดงถึงไดอะแกรมการทำงานในโหมด 1 ของไทม์เมอร์/เคาน์เตอร์ 0 หรือ 1 20
รูปที่ 2-18 แสดงถึงไดอะแกรมการทำงานในโหมด 2 ของไทม์เมอร์/เคาน์เตอร์ 0 หรือ 1 21
รูปที่ 2-19 แสดงถึงไดอะแกรมการทำงานในโหมด 3 ของไทม์เมอร์/เคาน์เตอร์ 0 22
รูปที่ 3-1 แสดงถึงโครงสร้างของเก้าอี้นวดที่ออกแบบ 29
รูปที่ 3-2 แสดงถึงชุดนวดที่เคลื่อนที่ในแนวนอน 29
รูปที่ 3-3 แสดงถึงแรงและโมเมนต์บิดในแนวดิ่งและแนวระดับของเพลา 30
รูปที่ 3-4 แสดงถึงฐาดรองรับชุดนวดที่เคลื่อนที่ในแนวตั้ง 34
รูปที่ 3-5 แสดงถึงเก้าอี้นวดไฟฟ้าที่ประกอบเสร็จ 34
รูปที่ 4-1 แสดงถึงวงจรไมโครคอนโทรลเลอร์ MCS-51 35
รูปที่ 4-2 แสดงถึงวงจรแหล่งจ่ายไฟ 36
รูปที่ 4-3 แสดงถึงวงจรขับมอเตอร์กระแสตรงแบบทครอบ 37
รูปที่ 4-4 แสดงถึงวงจรขับมอเตอร์กระแสตรงและมอเตอร์กระแสสลับ 38
รูปที่ 4-5 แสดงถึงการควบคุมการทำงานของเก้าอี้นวดไฟฟ้า 39
รูปที่ 4-6 แสดงถึงรีโมตควบคุมการทำงานของเก้าอี้นวด 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

หน้าที่

รูปที่ 4-7 แสดงถึงภาพรวมของการสั่งงานผ่านรีโมต 43
รูปที่ 4-8 แสดงถึงแผงผังลำดับการทำงานของโปรแกรมการทำงานแบบบังคับด้วยตนเอง 44
รูปที่ 4-9 แสดงถึงแผงผังลำดับการทำงานของโปรแกรมการทำงานแบบอัตโนมัติ 45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

การใช้ชีวิตของผู้คนที่อาศัยตามเมืองใหญ่ในปัจจุบันมักจะใช้เวลาส่วนใหญ่นั่งทำงาน และยังต้องใช้เวลาอีกไม่น้อยในการนั่งรถโดยสาร รวมทั้งการทำงานและการออกกำลังกายที่หนักเกินกำลัง เป็นสาเหตุสำคัญที่ก่อให้เกิดอาการปวดเมื่อยบนแผ่นหลัง เก้าอี้นวดไฟฟ้าซึ่งเป็นอุปกรณ์ช่วยบรรเทาอาการปวดเมื่อยแผ่นหลังในเบื้องต้นจึงมีการขยายตัวในตลาดเครื่องใช้ไฟฟ้ามากขึ้น มีจำหน่ายอยู่ด้วยกันจากหลากหลายผู้ผลิต สนนราคาตั้งแต่ระดับราคาใกล้แสนบาทจนถึงระดับหลายแสนบาท การใช้งานเก้าอี้นวดไฟฟ้ายังไม่เป็นที่แพร่หลายเนื่องจากราคาที่ยังสูงพอสมควร เก้าอี้นวดไฟฟ้าที่มีจำหน่ายในตลาดส่วนใหญ่จะมีโปรแกรมลักษณะการทำงานที่ตายตัว เปลี่ยนแปลงไม่ได้

ผู้ทำโครงงานจึงได้คิดออกแบบ และสร้างเก้าอี้นวดไฟฟ้าที่สามารถปรับลักษณะการทำงานได้ โดยเก้าอี้นวดสามารถนวดได้ในสองแนวแกนพิกัด คือ แนวตั้งและแนวนอน ในส่วนของระบบควบคุมการทำงานนั้นจะใช้ไมโครคอนโทรลเลอร์ในการควบคุม ซึ่งข้อดีที่จะทำให้เก้าอี้นวดไฟฟ้าในโครงงานนี้แตกต่างจากเก้าอี้นวดทั่วไปตามท้องตลาด คือ ผู้ใช้งานสามารถออกแบบลักษณะการนวดให้เหมาะสมกับสรีระตัวเองได้ ลักษณะการนวดไม่มีขีดจำกัดขึ้นอยู่กับการออกแบบโปรแกรม และโปรแกรมการทำงานของเก้าอี้นวดไฟฟ้ายังสามารถทำการดัดแปลงได้ตลอดอายุการใช้งาน

1.2 วัตถุประสงค์

1. ออกแบบและสร้างเก้าอี้นวดไฟฟ้าที่สามารถปรับลักษณะการทำงานได้
2. ออกแบบระบบควบคุมและโปรแกรมควบคุมการทำงานของเก้าอี้นวดไฟฟ้า โดยมีลักษณะโครงสร้างที่เป็นแบบสถาปัตยกรรมเปิด

1.3 ขอบเขตของโครงงาน

1. เก้าอี้นวดไฟฟ้าสามารถปรับลักษณะการทำงานได้ตามความเหมาะสมของสรีระผู้ใช้งาน
2. โปรแกรมควบคุมสามารถปรับปรุง หรือปรับเปลี่ยนได้ตลอดอายุการใช้งาน

1.4 ขั้นตอนการดำเนินโครงงาน

1. ศึกษาการทำงานของเก้าอี้นวดไฟฟ้าที่มีขายอยู่ทั่วไปตามท้องตลาด
2. ออกแบบและสร้างเก้าอี้นวดไฟฟ้าที่สามารถปรับลักษณะการทำงานได้
3. ออกแบบระบบควบคุมของเก้าอี้นวดไฟฟ้า
4. ติดตั้งและทำการทดลอง
5. สรุปผลการทดลองและข้อผิดพลาด
6. นำเสนอผลงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

ทฤษฎีและหลักการที่ใช้ในโครงงานนี้แบ่งเป็น 2 ส่วนหลักๆ คือ ทฤษฎีที่เกี่ยวข้องกับการออกแบบทางด้านฮาร์ดแวร์ และทฤษฎีที่เกี่ยวข้องกับการออกแบบทางด้านซอฟท์แวร์ โดยในส่วนของทางด้านฮาร์ดแวร์นั้นจะอ้างถึงการออกแบบอุปกรณ์ต่างๆที่ใช้ประกอบเป็นเก้าอี้นวด ส่วนในด้านซอฟท์แวร์ก็จะกล่าวถึงรายละเอียดเกี่ยวกับไมโครคอนโทรลเลอร์ และภาษาที่ใช้ในการเขียนโปรแกรม

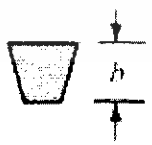
2.1 อุปกรณ์ที่ใช้ในเก้าอี้นวดไฟฟ้า

ส่วนประกอบของเก้าอี้นวดไฟฟ้าสามารถแบ่งออกเป็น 2 ส่วนหลัก คือ ส่วนโครงสร้างทางกลของเก้าอี้นวดไฟฟ้า และส่วนระบบควบคุมการทำงาน ซึ่งในส่วนประกอบของโครงสร้างทางกลนั้นสามารถแยกรายละเอียดปลีกย่อยที่สำคัญได้ดังนี้

2.1.1 สายพาน

สายพานที่ใช้ในการส่งกำลังสามารถแบ่งได้เป็น 4 ประเภทคือ

1. สายพานแบน (Flat Belt) มีภาคตัดขวางเป็นรูปสี่เหลี่ยมผืนผ้า
2. สายพานวี (V Belt) มีภาคตัดขวางเป็นรูปสี่เหลียมคางหมู
3. สายพานไทม์มิ่ง (Timing Belt) มีภาคตัดขวางเป็นรูปสี่เหลี่ยมคางหมูหลายรูปวางขนานกันและยึดปิดด้านบนร่วมกัน
4. สายพานกลม หรือสายพานเชือก (Round Belt) มีภาคตัดขวางเป็นรูปกลม

Belt Type	Figure	Joint	Size Range	Centre Distance
Flat		Yes	t = 0.03 to 0.20 in 0.75 to 5 mm	No upper limit
Round		Yes	d = 1/8 to 3/4 in	No upper limit
V		None	b = 0.31 to 0.91 in 8 to 19 mm	Limited
Timing		None	p = 2 mm and up	Limited

ตารางที่ 2-1 แสดงถึงคุณสมบัติของสายพานชนิดต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งในการออกแบบและสร้างระบบการทำงานของชุดนวดนั้น ใช้สายพานแบบไทม์มิ่งเพื่อให้เกิดความไวต่อการตอบสนองต่อแรงขับเคลื่อนที่ส่งมาจากมอเตอร์กระแสตรง และเป็นการป้องกันการลื่นไถล นอกจากนี้ การส่งกำลังโดยใช้สายพานยังมีข้อดี คือ

1. สามารถดูดซับการกระแทก และการสั่นสะเทือนได้ดี
2. ติดตั้งง่ายไม่ต้องการการหล่อลื่น
3. เสียงค่อนข้างเบา

สมการหาความยาวของสายพาน

$$\theta_d = \pi - 2\sin^{-1}\frac{D-d}{2c} \tag{2.1}$$

$$\theta_D = \pi + 2\sin^{-1}\frac{D-d}{2c} \tag{2.2}$$

$$L = \sqrt{4c^2 - (D-d)^2} + \frac{1}{2}(D\theta_D + d\theta_d) \tag{2.3}$$

โดยที่ θ_d คือ มุมของหน้าสัมผัสที่มูเล่ย์ตัวใหญ่

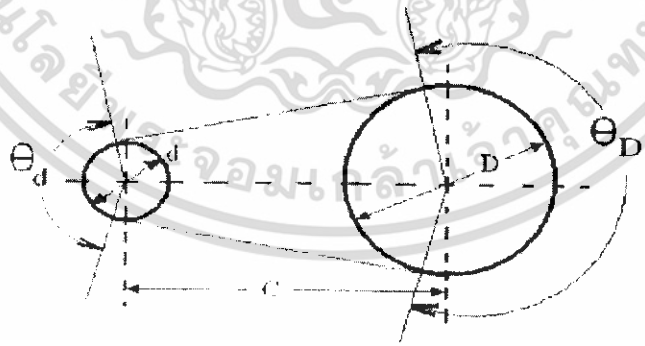
θ_D คือ มุมของหน้าสัมผัสที่มูเล่ย์ตัวเล็ก

C คือ ระยะห่างระหว่างมูเล่ย์ทั้งสอง

D คือ เส้นผ่านศูนย์กลางของมูเล่ย์ตัวใหญ่

d คือ เส้นผ่านศูนย์กลางของมูเล่ย์ตัวเล็ก

และ L คือ ความยาวของสายพาน



รูปที่ 2-1 แสดงถึงการติดตั้งสายพานแบบเปิดสำหรับขับเพลาที่ขนานกันให้หมุนไปในทิศทางเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการหาการส่งกำลังผ่านสายพาน

$$\frac{F_1}{F_2} = e^{f\theta} \tag{2.4}$$

$$P = (F_1 - F_2)\,V \tag{2.5}$$

โดยที่ F_1 คือ แรงดึงสายพานด้านตึง

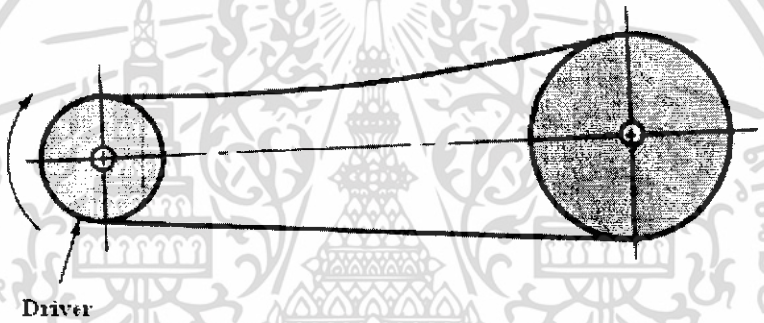
F_2 คือ แรงดึงสายพานด้านหย่อน

f คือ สัมประสิทธิ์ของแรงเสียดทาน

θ คือ มุมหน้าสัมผัส

P คือ กำลังที่ส่งผ่านสายพาน

และ V คือ ความเร็วสายพาน

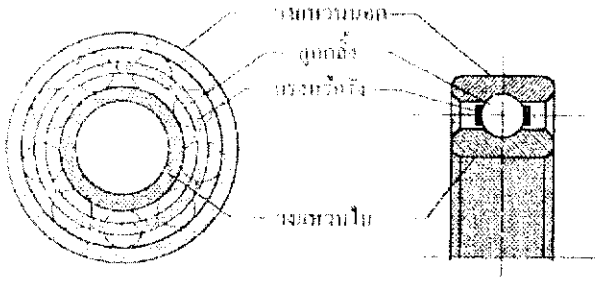


รูปที่ 2-2 แสดงถึงสายพานที่มีด้านหย่อนและตึง

2.1.2 แบริ่ง

แบริ่งแบบลูกกลิ้งหรือตลับลูกปืน เป็นชิ้นส่วนที่ใช้รองรับเพลาและส่งถ่ายภาระจากเพลาผ่านลูกกลิ้ง (Rolling Element) ซึ่งอยู่ระหว่างวงแหวนในและวงแหวนนอก แบริ่งแบบกลิ้งประกอบด้วยวงแหวนในและวงแหวนนอก ซึ่งวงแหวนในใช้สวมเข้ากับเพลา และวงแหวนนอกยึดอยู่ในตัวเรือนของแบริ่ง ภายในตัวแบริ่งจะมีลูกกลิ้งแบบเม็ดกลม หรือเม็ดทรงกระบอกอยู่ระหว่างวงแหวนในและวงแหวนนอกโดยมีกรงหรือรัง (Cage) ยึดดั่นแยกลูกกลิ้งให้มีระยะห่างคงที่ และเมื่อวงแหวนใดวงแหวนหนึ่งหมุนลูกกลิ้งจะกลิ้งอยู่ในรางของวงแหวน ซึ่งทำให้ความเสียดทานระหว่างรางและลูกกลิ้งลดลงมาก แต่เนื่องจากพื้นที่ผิวสัมผัสระหว่างรางและลูกกลิ้งมีน้อย ประกอบกับภาระต่อหนึ่งหน่วยพื้นที่มีค่าสูง ลูกกลิ้งและวงแหวนจึงต้องทำจากเหล็กกล้าที่มีความแข็งและความต้านแรงสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-3 แสดงถึงส่วนประกอบภายในของแบริ่งแบบกลิ้งหรือตลับลูกปืน

คุณสมบัติทั่วๆไปของแบริ่งแบบกลิ้ง ได้แก่

1. ความเสียดทานขณะเริ่มต้นหมุนและเมื่อหมุนแล้วเกือบเท่ากัน ยกเว้นที่ความเร็วรอบสูง
2. ต้องการการหล่อลื่นและบำรุงรักษาน้อย
3. ใช้เนื้อที่ตามแนวแกนน้อย
4. มีอายุการใช้งานจำกัดเนื่องจากรางหรือลูกกลิ้งมักจะเกิดการสึกหรอหรือความล้าที่ผิว
5. การถอดเปลี่ยนใหม่ทำได้ง่าย
6. แบริ่งแบบกลิ้งบางประเภทสามารถรับภาระได้ทั้งในแนวรัศมีและในแนวแกน

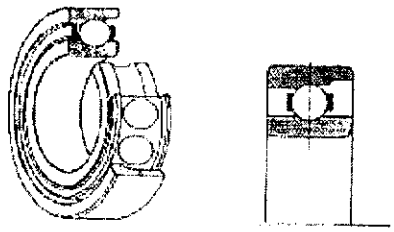
2.1.2.1 ประเภทของแบริ่งแบบเม็ดกลม

1. แบริ่งแบบเม็ดกลมร่องลึก (Deep Groove Ball Bearing) เป็นแบริ่งแบบเม็ดกลมร่องลึกแถวเดียวมีรางเป็นร่องลึกสำหรับให้เม็ดกลมกลิ้ง โดยทั่วไปใช้สำหรับรับภาระในแนวรัศมี แต่สามารถรับภาระในแนวแกนได้ถึงร้อยละ 70 ของภาระในแนวรัศมี เป็นแบริ่งที่ใช้งานกว้างขวางมากที่สุด

รูปที่ 2-4 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลมร่องลึก

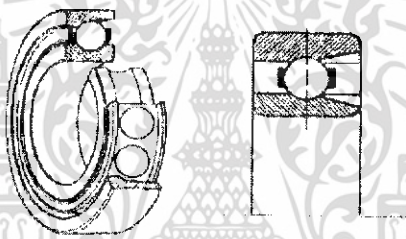
2. แบริ่งแบบเม็ดกลมร่องลึกมีรอยบากเติมเม็ดกลม (Filling Notch) จะมีรอยบากบนด้านหนึ่งของวงแหวนสำหรับเติมเม็ดกลม ซึ่งทำให้เพิ่มความสามารถในการรับภาระในแนวรัศมี แต่ความสามารถในการรับภาระในแนวแกนจะลดลง เนื่องจากเม็ดกลมชนรอยบาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



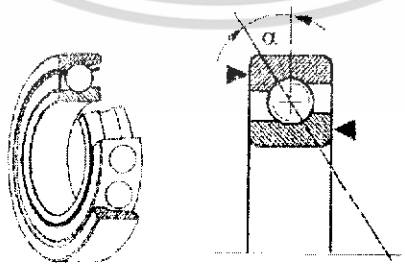
รูปที่ 2-5 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลมร่องลึกมีรอยบากเติมเม็ดกลม

3. แมกนีโตแบริ่ง (Magneto Bearing) ร่องที่วงแหวนในของแบริ่งแบบนี้จะตื้นกว่าแบริ่งแบบเม็ดกลมร่องลึก ด้านหนึ่งของวงแหวนนอกจะมีบ่า ร่องอีกด้านไม่มีบ่า วงแหวนนอกสามารถแยกส่วนออกมาได้ ซึ่งนับเป็นข้อดีต่อการประกอบ แมกนีโตแบริ่งเป็นแบริ่งขนาดเล็กมีขนาดเส้นผ่านศูนย์กลางเพลาตั้งแต่ 4 ถึง 30 มิลลิเมตร



รูปที่ 2-6 แสดงถึงลักษณะภายในของแมกนีโตแบริ่ง

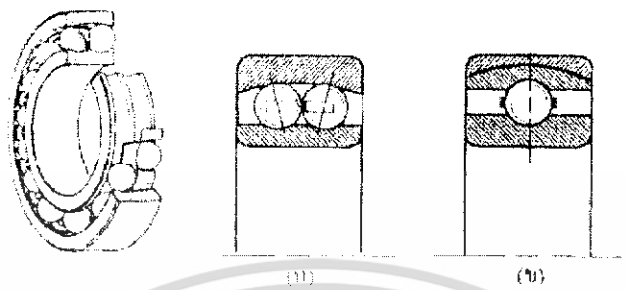
4. แบริ่งแบบเม็ดกลมสัมผัสเชิงมุม (Angular Contact Ball Bearing) แบริ่งแบบนี้แยกส่วนไม่ได้ แต่สามารถรับภาระในแนวแกนได้สูง โดยความสามารถในการรับภาระจะขึ้นอยู่กับมุมสัมผัส (Contact Angle) α โดยมุมสัมผัส α ที่โตกว่าจะสามารถรับภาระสูงกว่า ซึ่งในงานที่มีความเร็วรอบสูงมักจะใช้แบริ่งแบบเม็ดกลมที่มีมุมสัมผัสน้อยๆ



รูปที่ 2-7 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลมสัมผัสเชิงมุม

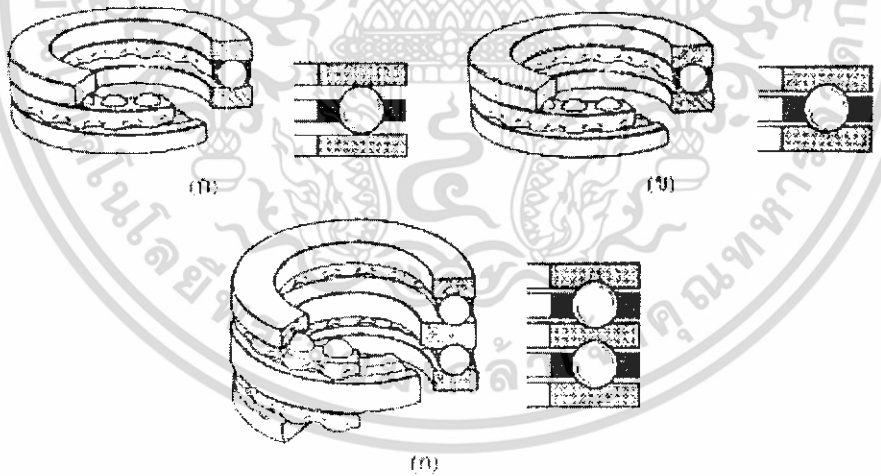
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. แบริ่งแบบเม็ดกลมปรับแนวแกนได้เอง (Self-Aligning Ball Bearing) เป็นแบริ่งแบบแยกส่วนไม่ได้ และมีความสามารถในการรับภาระต่ำกว่าแบบร่องลึก เนื่องจากรัศมีของวงแหวนโตขึ้นทำให้เกิดความเค้นสัมผัสสูง มีทั้งแบบปรับแนวแกนได้เองภายใน แบบปรับแนวแกนได้เองภายนอก



รูปที่ 2-8 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลม แบบปรับแนวแกนได้เองจากภายใน (ก) และแบบปรับแนวแกนได้เองจากภายนอก (ข)

6. แบริ่งแบบเม็ดกลมกันรุน (Thrust Ball Bearing) มีอยู่ 3 ประเภทได้แก่ แบบรางราบ, แบบรางร่องเดี่ยว และแบบรางร่องคู่ แบริ่งแบบนี้สามารถแยกส่วนได้หากต้องการได้แนวแกนที่เที่ยงตรงของเพลา และในการใช้งานนั้นแบริ่งชนิดนี้จะต้องการภาระต่ำสุดที่ความเร็วรอบสูง

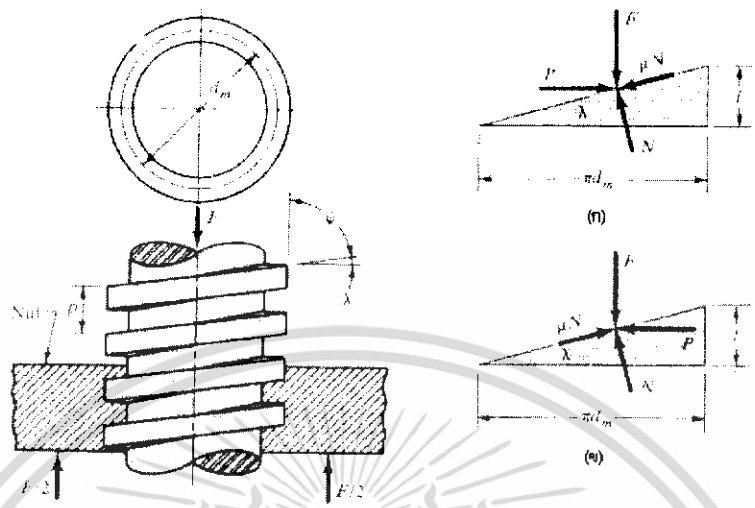


รูปที่ 2-9 แสดงถึงลักษณะภายในของแบริ่งแบบเม็ดกลมกันรุน แบบรางราบ (ก) แบบรางร่องเดี่ยว (ข) และแบบรางร่องคู่ (ค)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 สกรูส่งกำลัง

การยกน้ำหนักขึ้นหรือลงของสกรูส่งกำลังสามารถคำนวณได้โดย



รูปที่ 2-10 แสดงถึงทิศทางของแรงกระทำบนเกลียวของสกรูเมื่อ ยกภาระขึ้น (ก) และ ยกภาระลง (ข)

$$T_R = \frac{Fd_m}{2}\left(\frac{l + \pi\mu d_m}{\pi d_m - \mu l}\right) \tag{2.6}$$

$$T_L = \frac{Fd_m}{2}\left(\frac{\pi\mu d_m - l}{\pi d_m + \mu l}\right) \tag{2.7}$$

โดยที่ T_R คือ แรงที่จำเป็นต้องใช้ในการยกภาระขึ้น, นิวตัน-เมตร

T_L คือ แรงที่จำเป็นต้องใช้ในการยกภาระลง, นิวตัน-เมตร

F คือ ขนาดของภาระ, นิวตัน

d_m คือ เส้นผ่านศูนย์กลางเสมือน, มิลลิเมตร

μ คือ สัมประสิทธิ์ของแรงเสียดทาน

2.1.4 มอเตอร์กระแสตรง

มอเตอร์กระแสตรงจะมีการทำงานเพียง 3 แบบเท่านั้น คือ หมุนตามเข็มนาฬิกา หมุนทวนเข็มนาฬิกา และหยุดนิ่ง ซึ่งอัตราความเร็วในการหมุนและแรงบิดจะขึ้นอยู่กับกระแสและแรงดันที่จ่ายให้กับตัวมอเตอร์ ถ้าหากแรงบิดของมอเตอร์ไม่เพียงพอต่อการทำงานเราก็สามารถที่จะทำการทดรอบของการหมุนลงก็จะได้แรงบิดเพิ่มมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการหากำลังและแรงบิดของมอเตอร์

$$P = T_T\omega = IV \tag{2.8}$$

$$\omega = \frac{2\pi}{60} \times N \tag{2.9}$$

โดยที่ P คือ กำลังขับของมอเตอร์, วัตต์

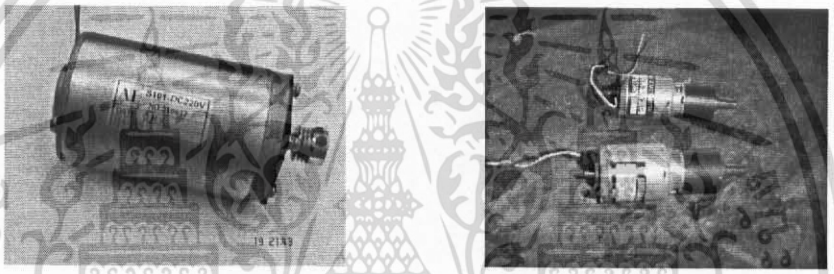
I คือ กระแสไฟฟ้า, แอมแปร์

V คือ แรงดันไฟฟ้า, โวลต์

T_T คือ แรงบิดหรือทอร์ค, นิวตัน-เมตร

ω คือ ความเร็วเชิงมุม, rad/s

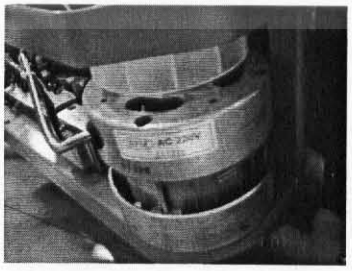
N คือ ความเร็วรอบ, rpm



รูปที่ 2-11 แสดงถึงมอเตอร์กระแสตรงที่ใช้ในโครงงานนี้

2.1.5 มอเตอร์กระแสสลับ

มอเตอร์กระแสสลับ (Alternating Current Motor) หมายถึง มอเตอร์ที่ต้องใช้กับไฟฟ้ากระแสสลับให้ไหลไปในขดลวดอาร์เมเจอร์ มอเตอร์กระแสสลับมีลักษณะและส่วนประกอบที่สำคัญเหมือนไดนาโมกระแสสลับ ต่างกันแต่วิธีใช้เท่านั้น โดยที่กระแสไฟฟ้ามีทิศทางการไหลสลับกลับไปกลับมาอยู่ตลอดเวลา การทำงานของมอเตอร์กระแสสลับที่ใช้ในโครงงานนี้มีการทำงาน 3 แบบ เช่นเดียวกับมอเตอร์กระแสตรง คือ หมุนตามเข็มนาฬิกา หมุนทวนเข็มนาฬิกา และหยุดนิ่ง



รูปที่ 2-12 แสดงถึงมอเตอร์กระแสสลับที่ใช้ในโครงงานนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6 เพลา

ในการออกแบบเพลานั้นมีสิ่งที่ต้องคำนึงถึงหลายสิ่งด้วยกัน เช่น โมเมนต์ดัด โมเมนต์บิด ซึ่งจะทำให้เกิดความเค้นที่จุดต่างๆของเพลา นอกจากนั้นความแข็งเกร็งก็เป็นสิ่งสำคัญที่ควรคำนึงถึง โดยเฉพาะความแข็งเกร็งทางด้านระยะโก่ง เพราะจะต้องใช้ระยะโก่งของเพลาที่อยู่ภายใต้แรงภายนอกเป็นตัวสำคัญในการกำหนดระยะเบียด (Clearance) ระหว่างล้อสายพาน และต้องเลือกชนิดของแบริ่งที่รองรับเพลาให้เหมาะสม โดยเลือกแบริ่งที่อนุญาตให้มีการเยื้องแนวสำหรับการใช้งานได้พอเหมาะกับระยะโก่งของเพลาที่จะเกิดขึ้น ซึ่งอาจจะเป็นแบริ่งแบบธรรมดาหรือแบริ่งแบบปรับแนวแกนได้เอง (Self-Aligning Bearing)

สมการหาขนาดของเพลา

$$d^3 = \frac{16}{\pi\tau}\left((C_t T)^2 + (C_m M)^2\right)^{1/2} \tag{2.8}$$

โดยที่ d คือ ขนาดเส้นผ่านศูนย์กลางของเพลา, มิลลิเมตร

τ คือ ความเค้นเฉือน, นิวตันต่อตารางมิลลิเมตร

C_t คือ ตัวประกอบความล้าเนื่องจากการดัด

C_m คือ ตัวประกอบความล้าเนื่องจากการบิด

T คือโมเมนต์บิด, นิวตัน-มิลลิเมตร

M คือ โมเมนต์ดัด, นิวตัน-มิลลิเมตร

รูปที่ 2-13 แสดงถึงเพลาพร้อมล้อสายพาน

2.2 ไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ (Microcontroller) เป็นชื่อของอุปกรณ์อิเล็กทรอนิคส์แบบหนึ่งที่รวมเอาหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์และลอจิก วงจรรับสัญญาณอินพุต วงจรขับสัญญาณเอาท์พุท หน่วยความจำ วงจรกำเนิดสัญญาณนาฬิกาไว้ด้วยกัน ทำให้สามารถนำไปใช้งานแทนวงจรอิเล็กทรอนิคส์ที่ซับซ้อนได้เป็นอย่างดี ช่วยลดจำนวนอุปกรณ์และขนาดของระบบ ในขณะที่มีขีด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสามารถสูงขึ้น ภายใต้งบประมาณที่เหมาะสม ดังนั้น ไมโครคอนโทรลเลอร์จึงเป็นอุปกรณ์ที่ใช้ในการควบคุม โดยที่สามารถเขียนโปรแกรมเพื่อกำหนดรูปแบบการควบคุมได้อย่างอิสระ

2.2.1 สาเหตุที่เลือกใช้ไมโครคอนโทรลเลอร์เบอร์ P89V51RD2

ในโครงงานนี้จะอ้างถึงไมโครคอนโทรลเลอร์ MCS-51 ซึ่งมีหน่วยความจำภายในเป็นแบบแฟลช (Flash Memory) ของ Philips เบอร์ P89V51RD2 เหตุผลที่ใช้ไมโครคอนโทรลเลอร์แบบนี้มีหลายประการด้วยกัน คือ

1. หน่วยความจำโปรแกรมภายในตัวไมโครคอนโทรลเลอร์เป็นแบบแฟลช ทำให้สามารถลบและเขียนใหม่ได้นับพันครั้ง จึงสามารถใช้งานในรูปแบบของไมโครคอนโทรเลอร์ชิปเดี่ยวได้โดยไม่ต้องใช้หน่วยความจำภายนอก ส่งผลให้สามารถใช้งานพอร์ตอินพุต-เอาท์พุตของไมโครคอนโทรลเลอร์ได้อย่างเต็มประสิทธิภาพ

2. การใช้ต้นทุนและเวลาในการพัฒนาระบบไมโครคอนโทรลเลอร์ลดลงอย่างมากเนื่องจากไม่ต้องใช้เครื่องมือพัฒนาจำพวกอีมูเลเตอร์ และเครื่องโปรแกรมอีพรอม

3. ด้วยการใช้หน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ ทำให้สามารถป้องกันการคัดลอกข้อมูลของหน่วยความจำโปรแกรมได้เป็นอย่างดี

4. มีข้อได้เปรียบ คือ สามารถทำการโปรแกรมข้อมูลในหน่วยความจำโปรแกรมได้โดยไม่ต้องถอดตัวไมโครคอนโทรเลอร์ออกมาทำการโปรแกรมใหม่หรือเรียกว่า "การโปรแกรมในวงจร" หรือ "การโปรแกรมในระบบ (In-System Programming)" ทำให้การพัฒนาหรือการซ่อมบำรุง ตลอดจนการปรับปรุงข้อมูลในหน่วยความจำโปรแกรมทำได้อย่างสะดวก ภายใต้งบประมาณที่ไม่สูงมากนัก

5. ชุดคำสั่งและสถาปัตยกรรมพื้นฐานเหมือนกับไมโครคอนโทรลเลอร์ MCS-51 ของผู้ผลิตอื่น

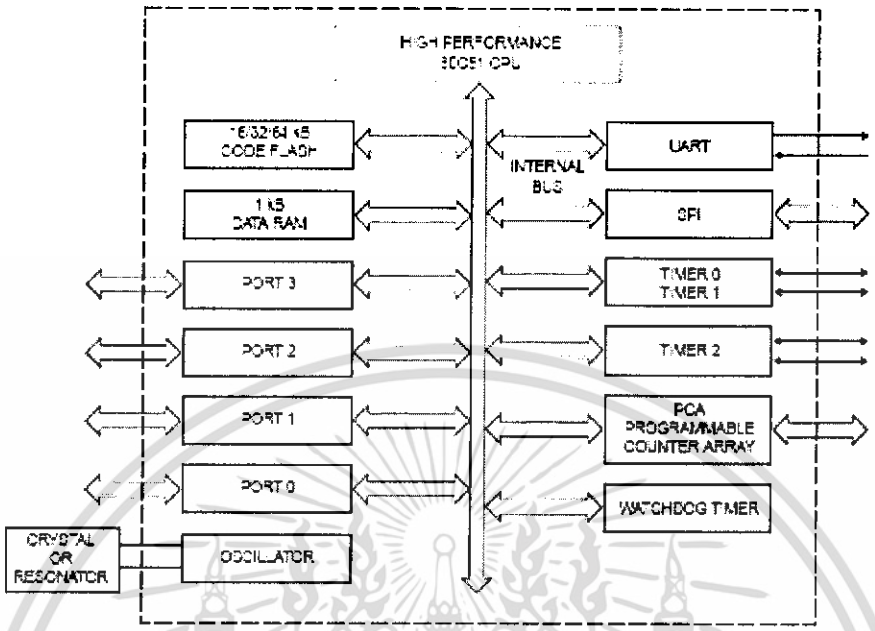
2.2.2 คุณสมบัติของไมโครคอนโทรลเลอร์ MCS-51 ตระกูล P89V51

คุณสมบัติของไมโครคอนโทรลเลอร์ MCS-51 ตระกูล P89V51 ที่สำคัญมีดังนี้

- ใช้การประมวลผลจากส่วนกลาง 80C51 ซีพียู
- ต้องการแหล่งจ่ายไฟ 5 โวลต์ ที่มีความถี่ตั้งแต่ 0 เมกกะเฮิรตซ์ ถึง 40 เมกกะเฮิรตซ์
- สามารถใช้หน่วยความจำ 16,32,64 กิโลไบต์สำหรับทำการโปรแกรมได้แบบการโปรแกรมในวงจรหรือการโปรแกรมในระบบ (In-Application Programming or In-System Programming)
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ในชิป
- มีวงจรสื่อสารอนุกรมแบบ SPI (Serial Peripheral Interface)
- มีพอร์ตที่สามารถรับหรือส่งข้อมูลได้ทั้ง 2 ทิศทาง จำนวน 4 พอร์ต พอร์ตละ 8 บิต และมีขาพอร์ต 1 จำนวน 3 ขา ซึ่งสามารถต่อกับกระแสไฟสูงถึง 16 มิลลิแอมแปร์
- มีรีจิสเตอร์สำหรับใช้งานเป็นไทม์เมอร์ หรือ เคาน์เตอร์ เพื่อนับจำนวนสัญญาณนาฬิกาภายในชิป หรือนับการเปลี่ยนสถานะของสัญญาณภายนอก 16 บิต จำนวน 3 ตัว เพื่อใช้สำหรับนับจำนวนพัลส์ วัดความกว้างของพัลส์ หรือใช้วัดช่วงเวลา
- มีวอทช์ด็อกไทม์เมอร์ (Watchdog Timer) ในตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- จัดลำดับความสำคัญของสัญญาณอินเตอร์รัปต์จากแหล่งต่างๆ 8 แหล่งได้ 4 ระดับ



รูปที่ 2-14 แสดงถึงโครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 อนุกรม P89V51

ในโครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ P89V51RD2 จะพบว่ามีข้อน่าสังเกตอยู่หลายส่วน เช่น วงจรเชื่อมต่ออนุกรมแบบ SPI ซึ่งในไมโครคอนโทรลเลอร์เบอร์นี้ใช้การเขียนข้อมูลลงในหน่วยความจำโปรแกรมโดยไม่ต้องถอดตัวชิปออกจากระบบ หรือที่เรียกว่า การโปรแกรมในวงจร และยังมีไทม์เมอร์/เคาน์เตอร์ขนาด 16 บิตจำนวน 3 ตัว นอกจากนี้ยังมีวงจรวอทช์ด็อกไทม์เมอร์ที่ใช้ในการตรวจสอบการทำงานผิดพลาดของซีพียูอีกด้วย

2.2.3 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ P89V51RD2 จะมีสถาปัตยกรรมและขาใช้งานพื้นฐาน ดังแสดงในรูปที่ 2-15 โดยมีรายละเอียดขั้นต้น ดังนี้

1. ขาพอร์ต 0 (P0.0-P0.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาท์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุตสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วยเพื่อให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (Float) ซึ่งมีอินพุตอิมพีแดนซ์สูง จึงสามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนั้นขาพอร์ตนี้ยังถูกใช้ในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำภายนอก (A0-A7) และขาข้อมูล (D0-D7) โดยใช้กระบวนการมัลติเพล็กซ์เข้าช่วย เพื่อสลับการทำงานให้เป็นได้ทั้งขาติดต่อแอดเดรสและขาข้อมูล

2. ขาพอร์ต 1 (P1.0-P1.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาท์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการให้ขาพอร์ตใดเป็นอินพุต สามารถทำได้โดยการเขียนข้อมูล "1" ไปยัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ละบิตของพอร์ตที่ต้องการติดต่อด้วยเพื่อให้ขาพอร์ตนั้นมีสถานะปล่อยลอย นอกจากนั้นขาที่ P1.5 P1.6 และ P1.7 ยังสามารถต่อกับกระแสไฟสูงถึง 16 มิลลิแอมแปร์ ขาพอร์ต 1 สามารถติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำภายนอกและขาข้อมูลเช่นเดียวกัน

3. ขาพอร์ต 2 (P2.0-P2.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาท์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุตสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย นอกจากนั้นขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความจำภายนอก (A8-A15)

4. ขาพอร์ต 3 (P3.0-P3.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาท์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุตสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (Float) จึงมีอินพุตอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนั้นขาพอร์ต 3 ยังเป็นขาที่มีหน้าที่การใช้งานพิเศษ ดังมีรายละเอียดขั้นต้นต่อไปนี้

- ขา P3.0 ใช้เป็นขาอินพุตสำหรับรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา RXD
- ขา P3.1 ใช้เป็นขาอินพุตสำหรับส่งข้อมูลจากการสื่อสารแบบอนุกรม หรือขา TXD
- ขา P3.2 ใช้เป็นขาอินพุตรับสัญญาณอินเตอร์รัปต์จากภายนอกช่อง 0 หรือขา $\overline{\text{INT0}}$
- ขา P3.3 ใช้เป็นขาอินพุตรับสัญญาณอินเตอร์รัปต์จากภายนอกช่อง 1 หรือขา $\overline{\text{INT1}}$
- ขา P3.4 ใช้เป็นขาอินพุตรับสัญญาณไทม์เมอร์จากภายนอกช่อง 0 หรือขา T0
- ขา P3.5 ใช้เป็นขาอินพุตรับสัญญาณอินเตอร์รัปต์จากภายนอกช่อง 1 หรือขา T1
- ขา P3.6 ใช้เป็นขาสัญญาณ $\overline{\text{WR}}$ ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก
- ขา P3.7 ใช้เป็นขาสัญญาณ $\overline{\text{RD}}$ ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก

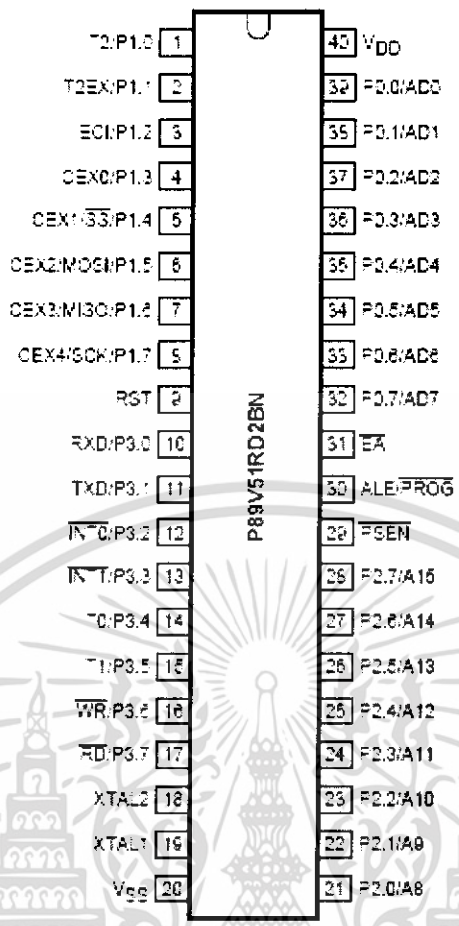
ซึ่งการใช้งานขาพอร์ต 3 ในหน้าที่พิเศษดังกล่าวนี้ จะต้องโหลดค่า "1" ไปยังแต่ละบิตที่ต้องการใช้ก่อนทุกครั้ง

5. ขา $\overline{PSEN}$ (Program Store Enable) ขานี้ใช้ในการส่งสัญญาณเพื่อร้องขอติดต่อกับหน่วยความจำโปรแกรมภายนอก เมื่อไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก ตัวไมโครคอนโทรลเลอร์จะส่งสัญญาณออกมาที่ขานี้ 2 ครั้งในแต่ละแมชชีนไซเคิล แต่ถ้าหากติดต่อกับหน่วยความจำข้อมูลภายนอก ขานี้จะไม่มีการส่งสัญญาณใดๆออกมา

6. ขารีเซต (RST) ใช้ในการรีเซตการทำงานของไมโครคอนโทรลเลอร์ โดยในการป้อนสัญญาณเพื่อรีเซตสถานะที่ขานี้ต้องอยู่ในระดับอย่างน้อย 2 แมชชีนไซเคิล โดยที่วงจรกำเนิดสัญญาณนาฬิกายังคงทำงานต่อเนื่องไปอย่างปกติ

7. ขา $\overline{EA}$ (External Access Enable) ใช้สำหรับเลือกการติดต่อหน่วยความจำจากโปรแกรมภายนอกหรือภายใน ขานี้จะต้องเชื่อมต่อกับขา V_{SS} เพื่อให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำภายนอก ถ้าขานี้เชื่อมต่อกับ V_{DD} จะเป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำภายในนอกจากนี้ขานี้ยังใช้เป็นขาอินพุตสำหรับรับแรงดันไฟสูง ซึ่งสามารถทนแรงดันได้ถึง 12 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-15 แสดงถึงการจัดขามาตรฐานของไมโครคอนโทรลเลอร์ MCS-51 เบอร์ P89V51RD2

8. ขา $ALE / \overline{PROG}$ (Address Latch Enable/Program Pulse Input) เป็นขาที่ใช้ในการควบคุมการแลตช์ของขาพอร์ต 0 เมื่อมีการใช้งานหน่วยความจำภายนอก นอกจากนี้ยังใช้เป็นขาสำหรับรับพัลส์ของการโปรแกรมในไมโครคอนโทรลเลอร์ที่มีหน่วยความจำแบบแฟลช

9. ขา XTAL1 และขา XTAL2 เป็นขาสำหรับต่อคริสตัลเพื่อสร้างสัญญาณนาฬิกาในการกำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์

10. ขา V_{DD} ใช้ต่อกับแหล่งจ่ายไฟ

11. ขา V_{SS} เป็นขากราวด์

2.2.4 โครงสร้างและการทำงานของพอร์ตในไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ต คือ พอร์ต 0 จนถึงพอร์ต 3 แต่ละพอร์ตมีขนาด 8 บิต เป็นพอร์ตแบบ 2 ทิศทาง กล่าวคือ สามารถเป็นได้ทั้งอินพุตสำหรับปรับสัญญาณข้อมูลเข้า และเอาท์พุตสำหรับส่งสัญญาณข้อมูลออก ทุกพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีวงแลตช์ และวงจรขับตลอดจนบัฟเฟอร์อินพุต

ที่พอร์ต 0 และพอร์ต 2 จะใช้งานเป็นพอร์ตอินพุต และเอาท์พุตสำหรับการใช้งานทั่วไปและใช้ในการติดต่อกับหน่วยความจำภายนอก สำหรับพอร์ต 3 ทั้งพอร์ตและพอร์ต 1 บางขา นอกจากจะใช้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาพอร์ตอินพุต-เอาท์พุต ตามปกติแล้ว ยังสามารถใช้งานในหน้าที่พิเศษได้อีก ขึ้นอยู่กับว่าเป็นไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ใด

วงจรภายในของแต่ละพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช เป็นวงจรของพอร์ต 0 และวงจรแลตช์ของแต่ละบิตในแต่ละพอร์ตก็คือวงจรดีฟลิปฟลอปนั่นเอง การอ่านค่าสถานะของพอร์ตและสถานะของวงจรแลตช์สามารถกระทำได้อย่างอิสระด้วยสัญญาณที่แยกจากกัน นั่นคือ สัญญาณอ่านข้อมูลจากขาพอร์ต และสัญญาณอ่านข้อมูลจากวงจรแลตช์ ส่วนการเขียนข้อมูลมายังพอร์ตต้องส่งสัญญาณมายังขา CLK ของดีฟลิปฟลอปในขณะที่ข้อมูลจะผ่านมายังขาบัสข้อมูลภายในเข้าสู่ขา D ของดีฟลิปฟลอป ที่พอร์ตนี้มีวงจรมัลติเพล็กซ์สำหรับกำหนดลักษณะการทำงานของพอร์ต ว่าต้องการใช้งานเป็นขาพอร์ตอินพุต-เอาท์พุต ปกติหรือใช้ในการติดต่อกับหน่วยความจำภายนอกไมโครคอนโทรลเลอร์ เนื่องจากที่ขาพอร์ต 0 ไม่มีวงจรพูลอัปภายใน หากมีการนำพอร์ต 0 ไปใช้งานเป็นพอร์ตอินพุตจะต้องต่อตัวต้านทานพูลอัปภายนอกเข้าที่ขาพอร์ต 0 ทุกขาด้วย

พอร์ต 1 ซึ่งมีลักษณะโดยทั่วไปคล้ายกับพอร์ต 0 หากแต่ไม่มีวงจรมัลติเพล็กซ์ เนื่องจากพอร์ตนี้จะไม่ใช้ในการติดต่อกับหน่วยความจำภายนอก แต่จะมีวงจรพูลอัปภายในที่แต่ละบิตของพอร์ตนี้แทน

วงจรภายในของพอร์ต 2 จะคล้ายกับพอร์ต 0 มาก ต่างกันเพียงมีวงจรพูลอัปเพิ่มเติมเข้ามา ส่วนวงจรภายในของพอร์ต 3 พบว่ามีลักษณะคล้ายกับพอร์ต 1 แต่มีการเพิ่มเติมวงจรบัฟเฟอร์ และวงจรอินพุต-เอาท์พุต เพื่อทำงานในฟังก์ชั่นพิเศษเข้ามา เนื่องจากพอร์ต 3 สามารถนำไปใช้งานหน้าที่พิเศษได้ทุกขา

2.2.4.1 การใช้งานเป็นพอร์ตอินพุต

เนื่องจากพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชสามารถเป็นได้ทั้งอินพุตและเอาท์พุต ดังนั้นจึงมีความจำเป็นอย่างยิ่งต้องทำความเข้าใจถึงการกำหนดลักษณะการทำงานให้แก่พอร์ตของไมโครคอนโทรเลอร์ MCS-51 แบบแฟลช

ในการกำหนดให้เป็นพอร์ตอินพุตต้องเริ่มต้นด้วยการเขียนข้อมูล “1” มาที่แต่ละบิตของพอร์ตที่ต้องการใช้งานเป็นอินพุต เพื่อหยุดการทำงานของเฟสที่ใช้ในการขับสัญญาณเอาท์พุตของบิตนั้นๆ ทำห้าสัญญาณของพอร์ตเชื่อมต่อเข้ากับวงจรพูลอัปภายในโดยตรง ส่งผลให้ขาพอร์ตนั้นมีลอจิกเป็น “1” สามารถรับสัญญาณลอจิก “0” จากอุปกรณ์ภายนอกได้ง่าย สัญญาณข้อมูลจากอุปกรณ์ภายนอกจะถูกส่งเข้ามาแล้วเก็บไว้ในวงจรบัฟเฟอร์ภายในพอร์ต แล้วรอให้ซีพียูมาอ่านค่าเข้าไป เมื่อเป็นเช่นนี้ อุปกรณ์ภายนอกที่เชื่อมกับพอร์ตอินพุตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช ควรกำหนดให้ทำงานในสภาวะลอจิก “0” จะดีและสะดวกที่สุด ซึ่งในปัจจุบันอุปกรณ์อินพุตที่เชื่อมต่อไมโครคอนโทรลเลอร์แบบทั้งหมดทำงานที่ลอจิก “0” แล้ว

2.2.4.2 การใช้งานเป็นพอร์ตเอาท์พุต

โดยปกติแล้ว ขาพอร์ตจะกำหนดให้มีลักษณะเป็นเอาท์พุตอยู่แล้ว ดังนั้นจึงสามารถส่งข้อมูลออกไปได้อย่างง่ายดายและตรงไปตรงมา กล่าวคือ เมื่อต้องการส่งข้อมูล “0” ออกไปทางเอาท์พุตก็ให้เขียนข้อมูล “0” ไปยังวงจรแลตช์ ซึ่งก็จะส่งต่อไปยังเฟส ทำให้เฟสทำงาน ที่ขาพอร์ตที่กำหนดให้ทำงานก็จะเกิดลอจิก “0” ขึ้นในทางตรงข้ามหากต้องการส่งข้อมูล “1” ออกไปก็ให้เขียนข้อมูล “1” ไปยังวงจรแลตช์ วงจรขับก็จะหยุดทำงาน ทำให้ที่ขาพอร์ตเชื่อมต่อกับวงจรพูลอัปภายในเกิดเป็นลอจิก “1” ที่ขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ตนั้น ซึ่งจะคล้ายกับการกำหนดให้เป็นขาอินพุตมาก เพียงแต่แตกต่างกันที่กระบวนการในการเคลื่อนย้ายข้อมูล โดยถ้าเป็นอินพุตจะมีสัญญาณมาอ่านข้อมูลที่บัฟเฟอร์แต่ถ้าเป็นเอาท์พุตจะไม่มีการอ่านข้อมูลที่บัฟเฟอร์แต่อย่างใด เว้นแต่ในกรณีที่ต้องการตรวจสอบข้อมูลที่ส่งออกมาทางเอาท์พุต

เมื่อใช้งานเป็นพอร์ตเอาท์พุตแต่ละขา หรือแต่ละบิต ของแต่ละพอร์ตมีความสามารถในการจ่ายกระแส หรือที่เรียกว่า กระแสซอร์ส (Source Current) ได้สูงสุด 10 mA และทุกขารวมกันในแต่ละพอร์ต (ทั้ง 8 บิต) สูงสุด26 mA สำหรับพอร์ต 0 และ 15 mA สำหรับพอร์ต 1-3 ในกรณีที่ใช้งานทุกพอร์ตเอาท์พุตจะสามารถจ่ายกระแสได้รวมกันสูงสุด 71 mA ดังนั้นในการใช้งานเป็นพอร์ตเอาท์พุตเพื่อไม่ให้เกิดปัญหาเกี่ยวกับความสามารถ ในการจ่ายกระแสจึงควรต่อวงจรบัฟเฟอร์ทางเอาท์พุตเพื่อช่วยในการขับกระแสอีกทางหนึ่ง

2.3 โปรแกรมภาษาซี

ในโครงงานนี้ผู้จัดทำได้เลือกใช้ภาษาซีในการเขียนโปรแกรมควบคุมเนื่องจาก

1. เนื่องจากภาษาซีเป็นภาษาที่มีการกำหนดมาตรฐานสำหรับเครื่องคอมพิวเตอร์ทุกรุ่นและระบบปฏิบัติการทุกชนิด ทำให้โครงสร้างทางภาษา ฟังก์ชันและไลบรารี (Library) ต่างๆ สามารถนำไปใช้งานระหว่างเครื่องแต่ละรุ่นและระบบปฏิบัติการแต่ละชนิดได้

2. ปัจจุบันนี้มีการพัฒนาตัวแปรภาษาซีขึ้นมาสำหรับใช้กับเครื่องทุกรุ่น และระบบปฏิบัติการทุกชนิด ดังนั้นไม่ว่าเราจะใช้เครื่องรุ่นใด และใช้ระบบปฏิบัติการชนิดใดก็ตาม ก็สามารถเขียนโปรแกรมภาษาซีได้

3. โปรแกรมที่เขียนด้วยภาษาซีมีขนาดเล็กและทำงานได้เร็ว

4. ภาษาซีมีโครงสร้างทางภาษาที่ดี และเครื่องหมายสำหรับดำเนินการ ไม่ว่าจะเป็นการคำนวณทาง คณิตศาสตร์ ตรรกศาสตร์ หรือการเปรียบเทียบ มีประสิทธิภาพการทำงานสูง

5. สามารถเขียนคำสั่งภาษาซีเพื่อควบคุมการทำงานอุปกรณ์ฮาร์ดแวร์บางส่วนได้

6. มีฟังก์ชั่นสำเร็จรูปสำหรับทำงานประเภทต่างๆให้เลือกใช้มากมาย ซึ่งช่วยประหยัดเวลาในการเขียนคำสั่ง นอกจากนี้ถ้าฟังก์ชั่นที่ภาษาซีเตรียมไว้ให้ ใช้งานได้ไม่ตรงตามต้องการทั้งหมด เราสามารถเขียนคำสั่งเพิ่มเติมลงไปได้

2.3.1 การใช้งานไทม์เมอร์ของไมโครคอนโทรลเลอร์ MCS-51 ด้วยโปรแกรมภาษาซี

การใช้งานไทม์เมอร์/เคาน์เตอร์ (Timer/Counter) เป็นส่วนประกอบที่สำคัญของไมโครคอนโทรลเลอร์ เนื่องจากในการทำงานของไมโครคอนโทรลเลอร์จะต้องมีการเก็บและตรวจสอบค่าของเวลา และจำนวนของสัญญาณนาฬิกาอยู่ตลอดเวลา ทั้งนี้เพื่อประโยชน์ในการสร้างฐานเวลา สร้างสัญญาณพัลส์เปรียบเทียบค่าเวลา หรือเปรียบเทียบค่าของการนับ รวมไปถึงการกำหนดอัตราการถ่ายทอดข้อมูลในการสื่อสารข้อมูลของพอร์ตอนุกรมด้วย ในไมโครคอนโทรลเลอร์ MCS-51 เบอร์ P89V51RD2 ของ Philips มีวงจร ไทม์เมอร์/เคาน์เตอร์ ขนาด 16 บิต 3 ตัวประกอบด้วย ไทม์เมอร์ 0 (Timer0) ไทม์เมอร์ 1 (Timer1) และไทม์เมอร์ 2 (Timer2) เรียกสั้นๆว่า T0 T1 และ T2โดยรีจิสเตอร์ไทม์เมอร์/เคาน์เตอร์ทั้งสามตัวสามารถกำหนดให้ทำงานเป็นตัวตั้งเวลาหรือไทม์เมอร์ และตัวนับหรือเคาน์เตอร์ได้อย่างอิสระต่อกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.3.1.1 การทำงานเป็นไทม์เมอร์

เมื่อกำหนดให้ทำงานเป็นตัวตั้งเวลา หรือไทม์เมอร์ ค่าของรีจิสเตอร์จะเพิ่มขึ้น ในทุกๆ แมชชีนไซเคิล ดังนั้นเมื่อทำงานเป็นไทม์เมอร์ รีจิสเตอร์จะทำการนับค่าของแมชชีนไซเคิลนั่นเองและเนื่องจากแมชชีนไซเคิลประกอบด้วยคาบเวลาของวงจรกำเนิดสัญญาณนาฬิกา 6 คาบเวลา ดังนั้นอัตราในการนับของรีจิสเตอร์จึงเท่ากับ 1/6 ของความถี่สัญญาณนาฬิกาในกรณีเลือกโหมด 6 สัญญาณนาฬิกา/ไซเคิล (สำหรับ P89V51RD2 เลือกได้ทั้ง 12 และ 6 สัญญาณนาฬิกา/ไซเคิล)

2.3.1.2 การทำงานเป็นเคาน์เตอร์

เมื่อทำงานเป็นตัวนับหรือเคาน์เตอร์ ค่าของรีจิสเตอร์จะเพิ่มขึ้นก็ต่อเมื่อมีการเปลี่ยนแปลงของระดับลอจิกจาก "1" เป็น "0" เกิดขึ้นที่ขาอินพุตทางฮาร์ดแวร์ของวงจรไทม์เมอร์/เคาน์เตอร์ ซึ่งก็คือ ขา T0 (P3.4), ขา T (P3.5) และ ขา T2 (P1.0) โดยจะมีการสุ่มรับสัญญาณจากขาอินพุตในทุกๆคาบเวลาที่ 2 ของสเตทที่ 5 (S5P2) ในแต่ละแมชชีนไซเคิล

เมื่อสัญญาณอินพุตเปลี่ยนแปลงจาก "1" เป็น "0" เป็นเวลาหนึ่งไซเคิล ในไซเคิลต่อมาค่าของการนับจะเพิ่มขึ้นหนึ่งค่า และจะไปปรากฏในรีจิสเตอร์ภายในคาบเวลาที่ 1 ของสเตทที่ 3 (S3P1) ของแมชชีนไซเคิลต่อไปหลังจากที่ตรวจจับพบการเปลี่ยนแปลงที่ขาไทม์เมอร์อินพุตแล้ว เมื่อเป็นเช่นนี้ในกระบวนการตรวจจับการเปลี่ยนแปลงของสัญญาณอินพุตที่ขาไทม์เมอร์จะต้องใช้ 2 แมชชีนไซเคิล อัตราการนับของเคาน์เตอร์จึงเท่ากับ 1/2 ของความถี่สัญญาณนาฬิกาใน P89V51RD2 ดังนั้น ความถี่สูงสุดของสัญญาณอินพุตที่ไทม์เมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 สามารถตรวจจับได้จึงเท่ากับความถี่ของสัญญาณนาฬิกาหารด้วย 12 ยกตัวอย่างในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ P89V51RD2 ซึ่งใช้ความถี่สัญญาณนาฬิกาได้สูงสุด 20MHz จะสามารถตรวจจับความถี่ของสัญญาณได้สูงสุดประมาณ 1.667MHz

2.3.2 รีจิสเตอร์ที่เกี่ยวข้อง

รายละเอียดของรีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของไทม์เมอร์/เคาน์เตอร์ ในไมโครคอนโทรลเลอร์ MCS-51 มีดังนี้

2.3.2.1 รีจิสเตอร์ไทม์เมอร์

มีด้วยกัน 4 ตัวคือ TL0 มีแอดเดรสอยู่ที่ 0x80A TH0 มีแอดเดรสอยู่ที่ 0x80C TL1 มีแอดเดรสอยู่ที่ 0x80B และ TH1 มีแอดเดรสอยู่ที่ 0x80D รีจิสเตอร์ทั้ง 4 ตัวจะอยู่ในพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษหรือ SFR รีจิสเตอร์แต่ละตัวมีขนาด 8 บิต แต่ในการใช้งานโดยทั่วไปมักใช้ร่วมกันโดยจัดเป็นคู่คือ TL0 กับ TH0 รวมกันเป็นรีจิสเตอร์ Timer 0 ขนาด 16 บิตและ TL1 กับ TH1 รวมกันเป็นรีจิสเตอร์ Timer 1 ขนาด 16 บิตโดยใน TL0 และ TL1 เก็บข้อมูล 8 บิตล่าง ส่วน TH0 และ TH1 เก็บข้อมูล 8 บิตบนรีจิสเตอร์ไทม์เมอร์ทั้ง2 คู่เมื่อนำมาใช้ร่วมกันจะสามารถเก็บค่าของการนับได้สูงสุด 65,536 หรือ 0xFFFF เมื่อนับถึงค่านี้แล้วจะวนไปเริ่มนับ 0x0000 ใหม่ และเมื่อเกิดการนับรอบใหม่ บิต TF0 หรือ TF1 ในรีจิสเตอร์ TCON ที่ใช้ควบคุมการทำงานของไทม์เมอร์จะเกิดการเซตเพื่อแจ้งให้ทราบว่า นับเกินค่าสูงสุดแล้ว การเซตบิต TF0 หรือ TF1 ขึ้นอยู่กับว่าเลือกใช้งานรีจิสเตอร์ไทม์เมอร์ตัวใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
62392

2.3.2.2 รีจิสเตอร์ควบคุมการทำงานของไทม์เมอร์/เคาน์เตอร์ หรือ TCON (Timer/Counter Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 0x88 ในพื้นที่ของรีจิสเตอร์ SFR สามารถเข้าถึงได้ในระดับบิต มีรายละเอียดการทำงานดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TF1 (Timer 1 Overflow Flag): เซตด้วยกระบวนการทางฮาร์ดแวร์ เมื่อค่าของรีจิสเตอร์ไทม์เมอร์ 1 เกิดการนับเกินหรือเกิดโอเวอร์โฟลว การเคลียร์บิตนี้ทำได้ด้วยกระบวนการทางฮาร์ดแวร์เช่นกันโดยบิตนี้จะเคลียร์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น

TR1 (Timer 1 Run Control Bit): ใช้ในการเปิดการทำงานของไทม์เมอร์ 1 (เอ็นเอเบิลหรือดิสเอเบิล) ทำการเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ถ้าต้องการให้ไทม์เมอร์ 1 ทำงานต้องเซตบิตนี้ให้เป็น "1"

TF0 (Timer 0 Overflow Flag): เซตด้วยกระบวนการทางฮาร์ดแวร์ เมื่อค่าของรีจิสเตอร์ไทม์เมอร์ 0 เกิดการนับเกินหรือเกิดโอเวอร์โฟลว การเคลียร์บิตนี้ทำได้ด้วยกระบวนการทางฮาร์ดแวร์เช่นกันโดยบิตนี้จะเคลียร์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น

TR0 (Timer 0 Run Control Bit): ใช้ในการเปิดการทำงานของไทม์เมอร์ 0 (เอ็นเอเบิลหรือดิสเอเบิล) ทำการเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ถ้าต้องการให้ไทม์เมอร์ 0 ทำงานต้องเซตบิตนี้ให้เป็น "1"

IE1 (External Interrupt 1 Edge Flag): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อสามารถตรวจจับขอบขาของสัญญาณอินเตอร์รัปต์จากภายนอกที่ขาอินพุตอินเตอร์รัปต์ 1 (INT 1) ได้ และจะทำการเคลียร์เมื่อมีการบริการอินเตอร์รัปต์เกิดขึ้น

IT1 (Interrupt 1 Type Control Bit): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ โดยใช้ในการเลือกลักษณะของสัญญาณ อินเตอร์รัปต์จากภายนอกที่ต้องการให้ทำการตอบสนองสำหรับขาอินพุตอินเตอร์รัปต์ 1 (INT1) การเซตและเคลียร์ทำด้วยกระบวนการซอฟต์แวร์

"0" เลือกขอบขาลงของสัญญาณ (Falling Edge)

"1" เลือกระดับลอจิกต่ำ (Low Level Triggered)

IE0 (External Interrupt 0 Edge Flag): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อสามารถตรวจจับขอบขาของสัญญาณอินเตอร์รัปต์จากภายนอกที่ขาอินพุตอินเตอร์รัปต์ 0 (INT 0) ได้ และจะทำการเคลียร์เมื่อมีการบริการอินเตอร์รัปต์เกิดขึ้น

IT0 (Interrupt 0 Type Control Bit): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ โดยใช้ในการเลือกลักษณะของสัญญาณ อินเตอร์รัปต์จากภายนอกที่ต้องการให้ทำการตอบสนองสำหรับขาอินพุตอินเตอร์รัปต์ 0 (INT0) การเซตและเคลียร์ทำด้วยกระบวนการซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"0" เลือกขอบขาลงของสัญญาณ (Falling Edge)

"1" เลือกระดับลอจิกต่ำ (Low Level Triggered)

2.3.2.3 รีจิสเตอร์เลือกโหมดการทำงานของไทม์เมอร์/เคาน์เตอร์ หรือ TMOD (Timer/Counter Mode Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 0x89 ในพื้นที่ของรีจิสเตอร์ SFR ไม่สามารถเข้าถึงได้ในระดับบิต แบ่งการทำงานเป็น 2 ส่วนคือ 4 บิตล่างเลือกใช้โหมดการทำงานของไทม์เมอร์ 0 และ 4 บิตบนใช้เลือกโหมดการทำงานของไทม์เมอร์ 1

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
GATE	C/T	M1	M0	GATE	C/T	M1	M0
ไทม์เมอร์ 1				ไทม์เมอร์ 0			

GATE: ใช้เลือกลักษณะการควบคุมการทำงานของไทม์เมอร์/เคาน์เตอร์

"0" ไทม์เมอร์/เคาน์เตอร์จะทำงานเมื่อบิต TRx ในรีจิสเตอร์ TCON เป็น "1"

"1" ไทม์เมอร์/เคาน์เตอร์จะทำงานเมื่อบิต TRx ในรีจิสเตอร์ TCON เป็น "1" และสถานะลอจิกที่ขาอินพุตอินเตอร์รัปต์ INT0 และ INT1 เป็น "1"

C/T (Timer or Counter Selector): ใช้เลือกลักษณะการทำงานของไทม์เมอร์/เคาน์เตอร์

"0" เลือกทำงานเป็นไทม์เมอร์ โดยใช้สัญญาณอินพุตจากสัญญาณนาฬิกาภายใน

"1" เลือกให้ทำงานเป็นเคาน์เตอร์ โดยรับสัญญาณอินพุตทางขา T0 หรือ T1

M1, M0 (Mode Selector Bit): ใช้เลือกโหมดการทำงานของไทม์เมอร์/เคาน์เตอร์

"00" เลือกให้ทำงานในโหมดไทม์เมอร์/เคาน์เตอร์ 13 บิต

"01" เลือกให้ทำงานในโหมดไทม์เมอร์/เคาน์เตอร์ 16 บิต

"10" เลือกให้ทำงานในโหมดไทม์เมอร์/เคาน์เตอร์ขนาด 8 บิตแบบตั้งค่าอัตโนมัติ

"11" สำหรับไทม์เมอร์ 0 เลือกให้ทำงานในโหมดไทม์เมอร์/เคาน์เตอร์แยกส่วน โดยแยกออกเป็นไทม์เมอร์/เคาน์เตอร์ 8 บิต 2 ตัว รีจิสเตอร์ TL0 จะได้รับการควบคุมการเปิดปิดจากบิต TR0 ในรีจิสเตอร์ TCON และในรีจิสเตอร์ TH0 ซึ่งเป็นไทม์เมอร์/เคาน์เตอร์ 8 บิต อีกตัวหนึ่งจะได้รับการควบคุมจากบิต TR1 ในรีจิสเตอร์ TCON ในกรณีของไทม์เมอร์ 1 เป็นการสั่งให้ ไทม์เมอร์/เคาน์เตอร์ 1 หยุดทำงาน (ดิสเอเบิล)

2.3.3 โหมดการทำงานของไทม์เมอร์/เคาน์เตอร์ 0 และ 1

ไทม์เมอร์ 0 และไทม์เมอร์ 1 สามารถเลือกโหมดการทำงานได้ 4 โหมดคือ

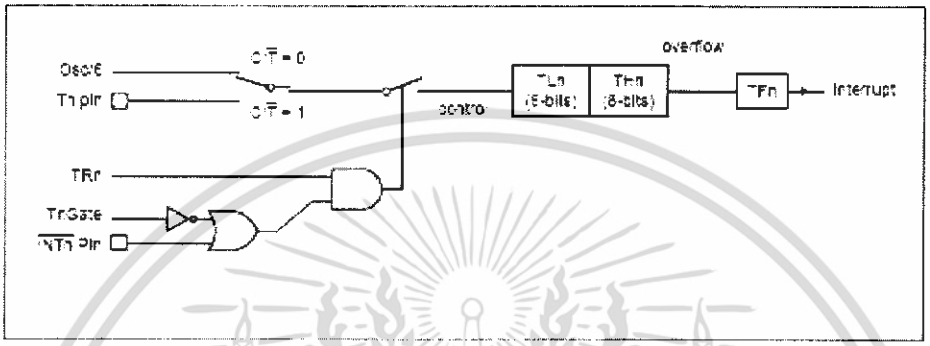
2.3.3.1 การทำงานในโหมด 0: ไทม์เมอร์เคาน์เตอร์ 13 บิต

มีไดอะแกรมการทำงานแสดงในรูปที่ 2-16 ในที่นี้จะใช้ไทม์เมอร์/เคาน์เตอร์ 1 ในการอธิบาย โหมดนี้จะเป็นการกำหนดให้ใช้งานรีจิสเตอร์ TL1 เพียง 5 บิต และ TH1 ครบ 8 บิต โดย TL0 จะทำหน้าที่คล้ายกับเป็นปรีสเกลเลอร์หาร 32 สัญญาณอินพุตสำหรับการนับจะเลือกสัญญาณนาฬิกาภายในหรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายนอกผ่านทางขา T1 ขึ้นอยู่กับการควบคุมของบิต C/T และ GATE ในรีจิสเตอร์ TMOD บิต TR1 ในรีจิสเตอร์ TCON และสถานะของลอจิกที่ขาอินพุต INT1 เมื่อ TL1 นับครบ 32 คือจาก 0-31 ก็จะส่งสัญญาณไปยัง TH1 เพื่อทำการเพิ่มค่า ดังนั้นในโหมดนี้ค่าของการนับจะมีขนาด 13 บิต เมื่อทำการนับครบรอบ ก็จะทำการเซตบิต TF1 ในรีจิสเตอร์ TCON

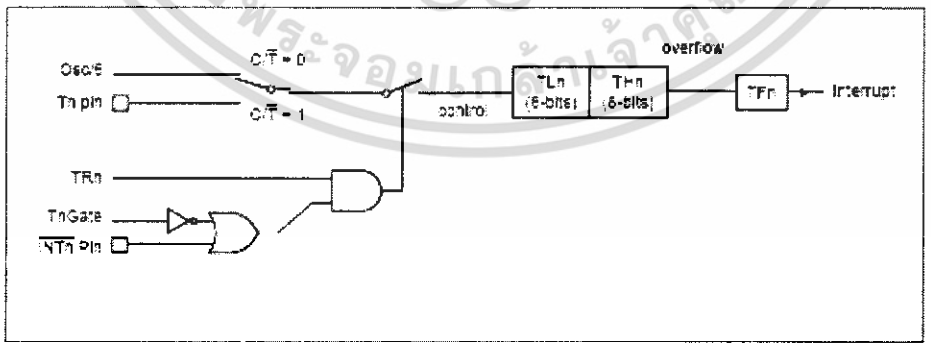
ส่วนการทำงานในโหมดนี้ของไทม์เมอร์/เคาน์เตอร์ 0 มีลักษณะเหมือนกันทุกประการ เพียงแต่เปลี่ยนรีจิสเตอร์และขาสัญญาณที่เกี่ยวข้องให้เป็นของไทม์เมอร์/เคาน์เตอร์ 0



รูปที่ 2-16 แสดงถึงไดอะแกรมการทำงานในโหมด 0 ของไทม์เมอร์/เคาน์เตอร์ 0 หรือ 1

2.3.3.2 การทำงานในโหมด 1: ไทม์เมอร์/เคาน์เตอร์ 16 บิต

มีไดอะแกรมการทำงานแสดงในรูปที่ 2-17 ในที่นี้จะใช้ไทม์เมอร์ 1 ในการอธิบาย การทำงานในโหมดนี้จะคล้ายกับโหมด 0 แต่จะใช้งานรีจิสเตอร์ TL1 และ TH1 ครบ 8 บิต ดังนั้นในโหมดนี้ค่าของการนับจะมีขนาด 16 บิต คือ 0x0000-0xFFFF เมื่อทำการนับครบรอบค่าของการนับจะเปลี่ยนจาก 0xFFFF เป็น 0x0000 ก็จะเซตบิต TF1 ในรีจิสเตอร์ TCON ส่วนการทำงานในโหมดนี้ของไทม์เมอร์ 0 เหมือนกันทุกประการ เพียงแต่เปลี่ยนรีจิสเตอร์และขาสัญญาณที่เกี่ยวข้องให้เป็นของไทม์เมอร์ 0

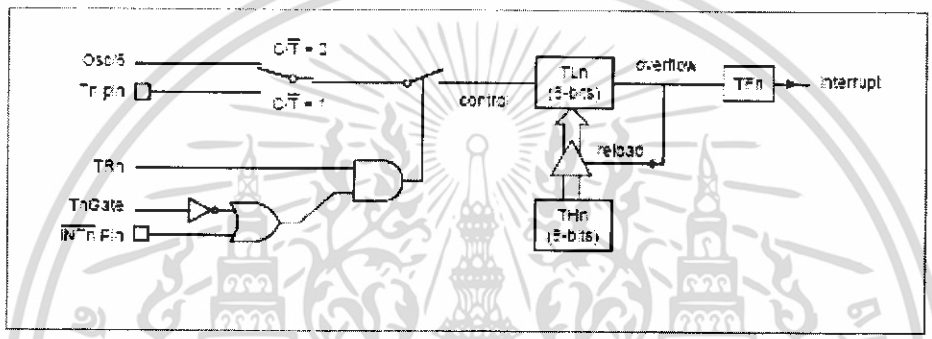


รูปที่ 2-17 แสดงถึงไดอะแกรมการทำงานในโหมด 1 ของไทม์เมอร์/เคาน์เตอร์ 0 หรือ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3.3 การทำงานในโหมด 2: ไทม์เมอร์/เคาน์เตอร์ 8 บิต แบบตั้งค่าอัตโนมัติ

มีไดอะแกรมการทำงานแสดงในรูปที่ 2-18 ในที่นี้จะใช้ไทม์เมอร์ 1 ในการอธิบาย การทำงานในโหมด นี้จะแยกรีจิสเตอร์ไทม์เมอร์ออกเป็น 2 ตัว ตัวละ 8 บิต โดยรีจิสเตอร์ TL1 ทำหน้าที่เป็นตัวนับค่าส่วน TH1 ใช้ในการเก็บค่าเริ่มต้นของการนับ เมื่อเริ่มต้นการทำงาน ค่าของรีจิสเตอร์ TH1 จะถูกส่งไปยังรีจิสเตอร์ TL1ทำให้เมื่อเริ่มต้นการทำงานค่าของรีจิสเตอร์ TL1 และ TH1 จะเหมือนกัน เมื่อ TL1 นับถึง 0xFF และจะเริ่มต้นการนับรอบใหม่ จะเซตบิต TF1 พร้อมๆกับรับค่าการนับเริ่มต้นจาก TH1 ใหม่โดยอัตโนมัติ หรือเรียกกระบวนการนี้ว่า รีโหลด (Reload) แม้ว่าจะมีการส่งค่าเริ่มต้นไปยัง TL1 แล้วก็ตาม ค่าของข้อมูลในรีจิสเตอร์ TH1 ก็ยังคงเป็นค่าเดิม ไม่มีการเปลี่ยนแปลงจนกว่าจะมีการกำหนดค่าใหม่ด้วยกระบวนการทางซอฟท์แวร์



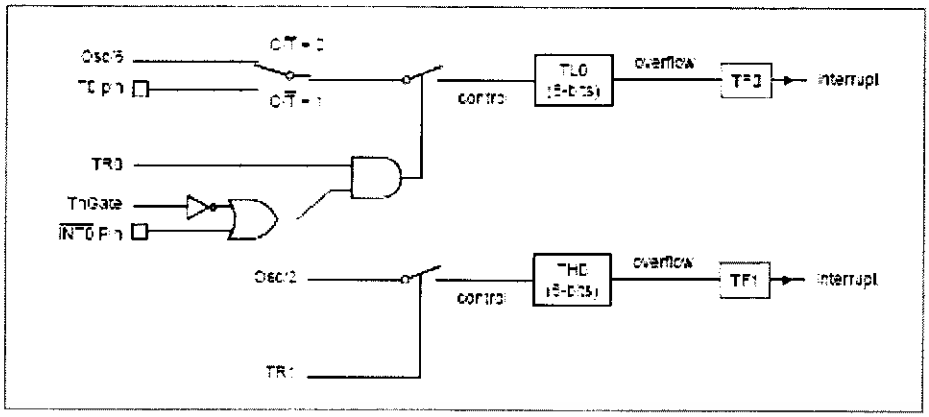
รูปที่ 2-18 แสดงถึงไดอะแกรมการทำงานในโหมด 2 ของไทม์เมอร์/เคาน์เตอร์ 0 หรือ 1

2.3.3.4 การทำงานในโหมด 3: ไทม์เมอร์/เคาน์เตอร์แยกส่วน หรือไทม์เมอร์/เคาน์เตอร์ 8 บิต

ในโหมดนี้เป็นโหมดเดียวที่การทำงานของไทม์เมอร์ 0 และ ไทม์เมอร์ 1 ไม่เหมือนกัน จะอธิบายในส่วนของไทม์เมอร์ 1 ก่อน เมื่อเข้าสู่โหมดนี้ จะเป็นการสั่งให้ไทม์เมอร์/เคาน์เตอร์หยุดนับค่าของการนับก่อนหน้านี้จะถูกเก็บไว้ในรีจิสเตอร์ไทม์เมอร์ 1 มีการทำงานเหมือนกับการดิสเอเบิลไทม์เมอร์ 1 ด้วยการเคลียร์บิต TR 1 ในรีจิสเตอร์ TCON

ส่วนการทำงานของไทม์เมอร์ 0 ในโหมดนี้ มีไดอะแกรมแสดงการทำงานในรูปที่ 2-19 การทำงานในโหมดนี้จะแยกรีจิสเตอร์ไทม์เมอร์ 0 ออกเป็น 2 ตัว ตัวละ 8 บิต คือรีจิสเตอร์ TL0 และ TH1 โดยแยกการทำงานออกจากกัน รีจิสเตอร์ TL0 สามารถเลือกการทำงานได้เหมือนกับไทม์เมอร์/เคาน์เตอร์ตามปกติ ส่วนรีจิสเตอร์ TH0 สามารถทำงานได้ในโหมดไทม์เมอร์ได้เพียงอย่างเดียว กล่าวคือ สามารถรับสัญญาณอินพุตจากสัญญาณนาฬิกาภายในเพียงทางเดียวเท่านั้น แต่การแจ้งการนับเกินยังคงเหมือนเดิม หากแต่ TL0 แจ้งผ่านบิต TF0 ในขณะที่ TH0 จะแจ้งผ่านทางบิต TF1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-19 แสดงถึงไดอะแกรมการทำงานในโหมด 3 ของไทม์เมอร์/เคาน์เตอร์ 0

2.3.4 ข้อมูลที่ใช้เลือกโหมดการทำงานของไทม์เมอร์/เคาน์เตอร์ 0 และ 1

เนื่องจากมีตัวแปรอยู่หลายตัวที่ใช้ในการควบคุมและเลือกโหมดการทำงานของไทม์เมอร์ 0 และไทม์เมอร์ 1 เพื่อให้เกิดความสะดวกในการใช้งานจึงได้สรุปข้อมูลที่ใช้กำหนดรูปแบบการทำงานของไทม์เมอร์ 0 และ ไทม์เมอร์ 1 ไว้ในตารางที่ 2-2 ถึง 2-5

โหมด	ฟังก์ชั่นของไทม์เมอร์	TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทม์เมอร์/เคาน์เตอร์ 13บิต	0x00	0x08
1	ไทม์เมอร์/เคาน์เตอร์ 16 บิต	0x01	0x09
2	8 บิตตั้งค่าอัตโนมัติ	0x02	0x0A
3	ไทม์เมอร์/เคาน์เตอร์แยกส่วน	0x03	0x0B

ตารางที่ 2-2 แสดงถึงข้อมูลรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทม์เมอร์/เคาน์เตอร์ 0 ทำงานเป็นไทม์เมอร์

โหมด	ฟังก์ชั่นของไทม์เมอร์	TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทม์เมอร์/เคาน์เตอร์ 13บิต	0x04	0x0C
1	ไทม์เมอร์/เคาน์เตอร์ 16 บิต	0x05	0x0D
2	8 บิตตั้งค่าอัตโนมัติ	0x06	0x0E
3	ไทม์เมอร์/เคาน์เตอร์แยกส่วน	0x07	0x0F

ตารางที่ 2-3 แสดงถึงข้อมูลรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทม์เมอร์/เคาน์เตอร์ 0 ทำงานเป็นเคาน์เตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมด	ฟังก์ชั่นของไทม์เมอร์	TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทม์เมอร์/เคาน์เตอร์ 13บิต	0x00	0x80
1	ไทม์เมอร์/เคาน์เตอร์ 16 บิต	0x10	0x90
2	8 บิตตั้งค่าอัตโนมัติ	0x20	0xA0
3	หยุดการทำงาน	0x30	0xB0

ตารางที่ 2-4 แสดงถึงข้อมูลรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทม์เมอร์/เคาน์เตอร์ 1 ทำงานเป็นไทม์เมอร์

โหมด	ฟังก์ชั่นของไทม์เมอร์	TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทม์เมอร์/เคาน์เตอร์ 13บิต	0x40	0xC0
1	ไทม์เมอร์/เคาน์เตอร์ 16 บิต	0x50	0xD0
2	8 บิตตั้งค่าอัตโนมัติ	0x60	0xE0
3	-	-	-

ตารางที่ 2-5 แสดงถึงข้อมูลรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทม์เมอร์/เคาน์เตอร์ 1 ทำงานเป็นเคาน์เตอร์

2.3.5 ตัวอย่างโปรแกรมภาษาซีในการกำหนดค่ารีจิสเตอร์เพื่อใช้งานไทม์เมอร์

```
TMOD = 0x21;    // เป็นการตั้งค่าไทม์เมอร์ 0 ให้ทำงานใน โหมด 1
                // เป็นการตั้งค่าไทม์เมอร์ 1 ให้ทำงานใน โหมด 2
TH1 = 0xA6;     // ตั้งค่าของการรีโหลดอัตโนมัติจาก TH1= => TL1 เท่ากับ 0xA6
TL1 = 0x83;     // ตั้งค่าเริ่มต้นในการนับของไทม์เมอร์ 1 โหมด 2 ที่ 0x83
                // เมื่อนับจนเกิน 0xFF จะเกิดโอเวอร์โฟลว
                // ทำให้มีการโหลดค่า 0xA6 ลงในรีจิสเตอร์ TL1 ใหม่
TH0 = 0xC2;     // ตั้งค่าการนับ 8 บิตบนของไทม์เมอร์ 0 โหมด 1
TL0 = 0xF0;     // ตั้งค่าการนับ 8 บิตล่างของไทม์เมอร์ 0 โหมด 1
                // รวมแล้วได้การตั้งค่าการนับเป็นเลข 16 บิต ⟶ 0xC2F0
TF1 = 0;        // เคลียร์บิตโอเวอร์โฟลวของไทม์เมอร์ 1
TF0 = 0;        // เคลียร์บิตโอเวอร์โฟลวของไทม์เมอร์ 0
TR1 = 1;        // เปิดไทม์เมอร์ 1 ให้เริ่มต้นการนับ
TR0 = 1;        // เปิดไทม์เมอร์ 0 ให้เริ่มต้นการนับ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.6 การเขียนโปรแกรมภาษาซีเพื่อตอบสนองการอินเตอร์รัปต์ของไมโครคอนโทรลเลอร์

การอินเตอร์รัปต์ (Interrupt) เป็นกระบวนการขัดจังหวะการทำงานของไมโครคอนโทรลเลอร์ MCS-51 มาตรฐานสามารถตอบสนองการอินเตอร์รัปต์ที่เกิดขึ้นได้จาก 5 แหล่งกำเนิดประกอบด้วย อินเตอร์รัปต์จากภายนอกผ่านทางขา $\overline{INT0}$ และ $\overline{INT1}$ อินเตอร์รัปต์จากไทม์เมอร์ 0 และ 1 และการอินเตอร์รัปต์จากพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ ในขณะที่ไมโครคอนโทรลเลอร์ P89V51RD2 สามารถกำหนดลำดับความสำคัญของการตอบสนองอินเตอร์รัปต์ได้ถึง 4 ระดับจากแหล่งกำเนิดสัญญาณอินเตอร์รัปต์ 7 แหล่ง โดยแหล่งกำเนิดสัญญาณอินเตอร์รัปต์ที่เพิ่มเติมเข้ามาคือ สัญญาณอินเตอร์รัปต์จากไทม์เมอร์ 2 และจากโมดูลวงจรนับแบบโปรแกรมได้หรือ PCA

มีรีจิสเตอร์ 3 ตัวที่เกี่ยวข้องคือ รีจิสเตอร์ IE IP และ IPH แสดงรายละเอียดด้านล่าง การกำหนดลำดับความสำคัญของการตอบสนองอินเตอร์รัปต์จะใช้ข้อมูลในแต่ละบิตของรีจิสเตอร์ IP และ IPH ดังแสดงความสัมพันธ์ด้านล่าง ยกตัวอย่าง ต้องการให้การอินเตอร์รัปต์จากไทม์เมอร์ 2 มีลำดับความสำคัญสูงสุดหรือระดับ 3 ต้องกำหนดให้บิต PT2 หรือบิต 5 ของรีจิสเตอร์ IP และบิต PT2H หรือบิต 5 ของรีจิสเตอร์ IPH เป็น "1" ทั้งคู่ จึงตอบสนองการอินเตอร์รัปต์ได้จาก 6 แหล่ง โดยเพิ่มการรับสัญญาณอินเตอร์รัปต์จากไทม์เมอร์/เคาน์เตอร์ 2 อีกหนึ่งแหล่งกำเนิด

2.3.6.1 การจัดการอินเตอร์รัปต์ในไมโครคอนโทรลเลอร์ MCS-51

เมื่อมีการอินเตอร์รัปต์ภายในไมโครคอนโทรลเลอร์ MCS-51 เกิดขึ้นและเอ็นเอเบิลการตอบสนองการอินเตอร์รัปต์ไว้ กระบวนการหลังจากนั้น ซีพียูจะกระโดดไปยังแอดเดรสในหน่วยความจำที่กำหนดไว้ เรียกแอดเดรสนี้ว่า แอดเดรสอินเตอร์รัปต์เวกเตอร์ (Interrupt Vector Address) ดังนั้นจะต้องเขียนโปรแกรมย่อยการบริการอินเตอร์รัปต์ไว้ที่แอดเดรสอินเตอร์รัปต์เวกเตอร์นี้ โดยค่าของแอดเดรสอินเตอร์รัปต์เวกเตอร์จะแตกต่างกันไปในการอินเตอร์รัปต์แบบต่างๆ ดังมีรายละเอียดต่อไปนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต2	บิต 1	บิต 0
EA	EC	ET2	ES	ET1	EX1	ET0	EX0

บิต	ชื่อบิต	รายละเอียด
IE.7	EA	เอ็นเอเบิลหรือดิสเอเบิลการอินเตอร์รัปต์หลัก
IE.6	EC	เอ็นเอเบิลการอินเตอร์รัปต์หลักจากโมดูล PCA
IE.5	ET2	เอ็นเอเบิลการอินเตอร์รัปต์หลักจากไทม์เมอร์ 2
IE.4	ES	เอ็นเอเบิลการอินเตอร์รัปต์หลักจากพอร์ตอนุกรม
IE.3	ET1	เอ็นเอเบิลการอินเตอร์รัปต์หลักจากไทม์เมอร์ 1
IE.2	EX1	เอ็นเอเบิลการอินเตอร์รัปต์หลักจากสัญญาณภายนอกผ่านขา $\overline{INT1}$
IE.1	ET0	เอ็นเอเบิลการอินเตอร์รัปต์หลักจากไทม์เมอร์ 0
IE.0	EX0	เอ็นเอเบิลการอินเตอร์รัปต์หลักจากสัญญาณภายนอกผ่านขา $\overline{INT0}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต2	บิต 1	บิต 0
-	PPC	PT2	PS	PT1	PX1	PT0	PX0

บิต	ชื่อบิต	รายละเอียด
IE.7	-	-
IE.6	PPC	บิตกำหนดลำดับความสำคัญของการอินเตอร์รัปต์จากโมดูล PCA
IE.5	PT2	บิตกำหนดลำดับความสำคัญของการอินเตอร์รัปต์จากไทม์เมอร์ 2
IE.4	PS	บิตกำหนดลำดับความสำคัญของการอินเตอร์รัปต์จากพอร์ตอนุกรม
IE.3	PT1	บิตกำหนดลำดับความสำคัญของการอินเตอร์รัปต์จากไทม์เมอร์ 1
IE.2	PX1	บิตกำหนดลำดับความสำคัญของการอินเตอร์รัปต์จากสัญญาณภายนอกผ่านขา $\overline{INT1}$
IE.1	PT0	บิตกำหนดลำดับความสำคัญของการอินเตอร์รัปต์จากไทม์เมอร์ 0
IE.0	PX0	บิตกำหนดลำดับความสำคัญของการอินเตอร์รัปต์จากสัญญาณภายนอกผ่านขา $\overline{INT0}$

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต2	บิต 1	บิต 0
-	PPCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H

บิต	ชื่อบิต	รายละเอียด
IPH.7	-	-
IPH.6	PPCH	-
IPH.5	PT2H	-
IPH.4	PSH	-
IPH.2	PX1H	บิตกำหนดลำดับความสำคัญสูงสุดให้แก่การอินเตอร์รัปต์ จากสัญญาณภายนอกผ่านขา $\overline{INT1}$
IPH.1	PT0H	-
IPH.0	PX0H	บิตกำหนดลำดับความสำคัญสูงสุดให้แก่การอินเตอร์รัปต์ จากสัญญาณภายนอกผ่านขา $\overline{INT0}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิตกำหนดลำดับความสำคัญ		ระดับของลำดับความสำคัญ
IPH.x	IP.x	
0	0	ระดับ 0 (ต่ำสุด)
0	1	ระดับ 1
1	0	ระดับ 2
1	1	ระดับ 3 (สูงสุด)

ตารางที่ 2-6 แสดงถึงความสัมพันธ์ของรีจิสเตอร์ IP และ IPH ในการกำหนดระดับของลำดับความสำคัญในการตอบสนองการอินเตอร์รัปต์

การอินเตอร์รัปต์จากภายนอกที่ขา $\overline{INT0}$ มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0x0003
การอินเตอร์รัปต์จากไทม์เมอร์ 0 มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0x000B
การอินเตอร์รัปต์จากภายนอกที่ขา $\overline{INT1}$ มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0x0013
การอินเตอร์รัปต์จากไทม์เมอร์ 1 มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0x001B
การอินเตอร์รัปต์จากพอร์ตอนุกรม มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0x0023
การอินเตอร์รัปต์จากไทม์เมอร์ 2 มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0x002B
การอินเตอร์รัปต์จากโมดูล PCA มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0x0033

2.3.6.2 การเขียนโปรแกรมย่อยบริการอินเตอร์รัปต์

มีหลักการโดยทั่วๆ ไป ดังนี้

1. ต้องเริ่มต้นด้วยแอดเดรสอินเตอร์รัปต์เวกเตอร์เสมอ เพื่อให้การตรวจสอบการทำงานทำได้ง่ายและแยกส่วนของโปรแกรมหลักหรือโปรแกรมย่อยอื่นๆ อย่างชัดเจนด้วยคำสั่ง ORG xxxx (ค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์)

2. เมื่อเข้าสู่โปรแกรมย่อย ควรเก็บค่าของรีจิสเตอร์หรือแฟลกที่ใช้แสดงสถานะต่างๆ ซึ่งต้องมีการใช้งานในโปรแกรมย่อยบริการอินเตอร์รัปต์นี้ไว้ในสแต็กเสียก่อน เพื่อป้องกันความผิดพลาดที่อาจเกิดขึ้นต่อการทำงานของทั้งโปรแกรมบริการอินเตอร์รัปต์นี้และโปรแกรมหลัก

3. เมื่อเขียนโปรแกรมบริการอินเตอร์รัปต์เรียบร้อยแล้ว ให้ทำการคืนค่าของรีจิสเตอร์ที่นำมาใช้ในโปรแกรมบริการอินเตอร์รัปต์ ยกเว้นรีจิสเตอร์ที่ต้องการนำผลการกระทำในบริการอินเตอร์รัปต์นี้ไปใช้งาน ซึ่งในทางปฏิบัติจริง ไม่พบมากนัก และไม่แนะนำให้เขียนโปรแกรมในลักษณะนี้

2.3.6.3 การตอบสนองการอินเตอร์รัปต์จากสัญญาณภายนอก

เป็นการตรวจสอบสัญญาณที่ส่งเข้ามายังมา $\overline{INT0}$ และ $\overline{INT1}$ หากตรงตามเงื่อนไขจะทำให้เกิดการอินเตอร์รัปต์ขึ้น โดยการเอนเอเบิลอินเตอร์รัปต์แบบนี้กระทำได้โดยการกำหนดค่าในรีจิสเตอร์ IE ที่บิต EX0 สำหรับการอินเตอร์รัปต์ที่ขา $\overline{INT0}$ และบิต EX1 สำหรับการอินเตอร์รัปต์ที่ขา $\overline{INT1}$ และเลือกเงื่อนไขการตรจสอบสัญญาณในรีจิสเตอร์ TCON ที่บิต IE0 สำหรับสัญญาณอินเตอร์รัปต์ ที่ขา $INT0$ และบิต IE1 สำหรับสัญญาณอินเตอร์รัปต์ที่ขา $\overline{INT1}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เงื่อนไขการตรวจสอบสัญญาณอินเตอร์รัปต์ที่ขา $\overline{INT0}$ และ $\overline{INT1}$ มีด้วยกัน 2 ลักษณะคือ

1. ตรวจสอบระดับลอจิก ถ้าบิต IEx ในรีจิสเตอร์ TCON เป็น "0" จะเกิดการอินเตอร์รัปต์จากภายนอกที่ขา $\overline{INT0}$ หรือ $\overline{INT1}$ ได้ก็ต่อเมื่อตรวจพบลอจิก "0" เมื่อเกิดการอินเตอร์รัปต์แล้ว ต้องทำให้ขานี้กลับสู่ระดับลอจิก "1" ก่อนที่การบริการอินเตอร์รัปต์เสร็จสิ้น เพื่อป้องกันการอินเตอร์รัปต์ซ้อน

2. ตรวจสอบขอบขาของสัญญาณ ถ้าบิต IEx ในรีจิสเตอร์ TCON เป็น "1" จะเกิดการอินเตอร์รัปต์จากภายนอกที่ขา $\overline{INT0}$ หรือ $\overline{INT1}$ ได้ก็ต่อเมื่อตรวจพบการเปลี่ยนแปลงของสัญญาณภายนอกที่ขา $\overline{INT0}$ หรือ $\overline{INT1}$ จาก "1" เป็น "0" หรือตรวจพบขอบขาลงของสัญญาณที่ป้อนมายังขา $\overline{INT0}$ หรือ $\overline{INT1}$ และต้องรักษาสถานะลอจิก "0" นี้เป็นเวลาอย่างน้อย 1 แมชชีนไซเคิล จึงถือว่าเกิดอินเตอร์รัปต์

เมื่อเกิดการอินเตอร์รัปต์ขึ้นซีพียูภายในไมโครคอนโทรลเลอร์จะกระโดดไปยังแอดเดรส 0x0003 สำหรับการอินเตอร์รัปต์ที่ขา $\overline{INT0}$ และ 0x0003 สำหรับการอินเตอร์รัปต์ที่ขา $\overline{INT1}$ ในรูปที่ เป็นไดอะแกรมแสดงการทำงานของการอินเตอร์รัปต์ในลักษณะนี้ โดยรูปที่เป็นการอินเตอร์รัปต์ที่ขา $\overline{INT0}$ ส่วนในรูปที่ เป็นการอินเตอร์รัปต์ที่ขา $\overline{INT1}$

เมื่อต้องการให้มีการตอบสนองการอินเตอร์รัปต์จากสัญญาณภายนอก จะต้องเซตบิต EA (Enable Interrupt All) เพื่อเปิดการตอบสนองอินเตอร์รัปต์ทั้งหมดของระบบเสียก่อน โดยใช้คำสั่ง จากนั้นเลือกให้มีตอบสนองต่ออินเตอร์รัปต์ใดๆ ในระบบได้ดังนี้

กำหนด	ความหมาย	ลำดับอินเตอร์รัปต์
EX0 = 1;	ให้มีการตอบสนองอินเตอร์รัปต์จากภายนอกที่ขา $\overline{INT0}$	Interrupt o
ET0 = 1;	ให้มีการตอบสนองอินเตอร์รัปต์จากไทม์เมอร์ 0	Interrupt 1
EX1 = 1;	ให้มีการตอบสนองอินเตอร์รัปต์จากภายนอกที่ขา $\overline{INT1}$	Interrupt 2
ET1 = 1;	ให้มีการตอบสนองอินเตอร์รัปต์จากไทม์เมอร์ 1	Interrupt 3
ES = 1;	ให้มีการตอบสนองอินเตอร์รัปต์จากพอร์ตอนุกรม	Interrupt 4
ET2 = 1;	ให้มีการตอบสนองอินเตอร์รัปต์จากไทม์เมอร์ 2	Interrupt 5

เมื่อกำหนดชนิดของการอินเตอร์รัปต์ที่ต้องการตอบสนองแล้ว (จะกำหนดกี่ชนิดก็ได้) จะต้องตอบสนองการเขียนฟังก์ชั่นเพื่อตอบสนองการอินเตอร์รัปต์ขึ้นมากำกับในแต่ละตัวด้วยว่า จะให้มีการทำคำสั่งอะไรบ้างเมื่อเกิดการอินเตอร์รัปต์ชนิดนั้น ๆ ขึ้น โดยต้องกำหนดลำดับของการตอบสนองอินเตอร์รัปต์ให้ถูกต้องด้วย มีรูปแบบของการประเมินฟังก์ชันดังนี้

```
void ชื่อฟังก์ชั่น (void) interrupt x
{
        คำสั่งที่ 1;
        คำสั่งที่ 2;
}
x = 0, 1, 2, 3, 4, 5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.6.4 การตอบสนองอินเตอร์รัปต์จากไทม์เมอร์ 0 และ 1

แหล่งกำเหนิดอินเตอร์รัปต์จากไทม์เมอร์ 0 และ 1 จัดเป็นแหล่งกำเนิดอินเตอร์รัปต์ภายในแบบหนึ่งโดยการใช้การเกิดโอเวอร์โฟลวจากการนับค่าในไทม์เมอร์/เคาน์เตอร์ ภายในไมโครคอนโทรลเลอร์ MCS 51 เมื่อไทม์เมอร์ 0 เกิดการโอเวอร์โฟลวก็จะทำให้การเซตบิต TFO ในรีจิสเตอร์ TCON และถ้าไทม์เมอร์ 1 เกิดการโอเวอร์โฟลว บิต FT1 ในรีจิสเตอร์ TCON จะได้รับการเซตเช่นเดียวกัน

ค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์ของการอินเตอรัปต์ แบบนี้อยู่ที่ 0x000B สำหรับไทม์เมอร์ 0 และ 0x001B สำหรับไทม์เมอร์ 1

อย่างไรก็ตามอินเตอร์รัปต์แบบนี้จะเกิดขึ้นหรือมีการตอบสนองก็ต่อเมื่อมีการเอ็นเอเบิลการอินเตอร์-รัปต์ โดยการเซตบิต EA ETO และ ET1 ในรีจิสเตอร์ IE

หากในโปรแกรมมีการเอ็นเอเบิลการอินเตอร์รัปต์จากไทม์เมอร์ 0 จะต้องเขียนฟังก์ชั่นสำหรับตอบสนองอินเตอร์รัปต์เป็น

```
void sevice_timer 0 (void)  interrupt 1
{
        ..............
        ...........
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

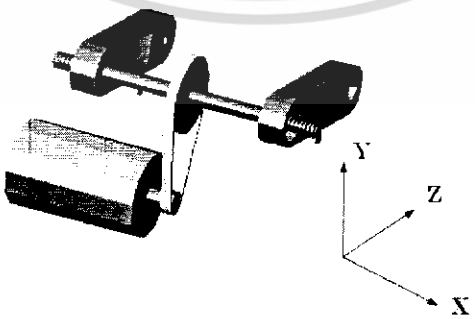
บทที่ 3
การออกแบบโครงสร้างทางกลของเก้าอี้นวดไฟฟ้า

ในการออกแบบเก้าอี้นวดนั้น โดยหลักการแล้วต้องการให้ชุดนวดสามารถเคลื่อนที่ได้ในสองแนวแกนพิกัดโดยกำหนดให้การเคลื่อนที่ในแกนนอนเปรียบเสมือนแนวแกน X และการเคลื่อนที่ในแนวตั้งเปรียบเสมือนแนวแกน Y

รูปที่ 3-1 แสดงถึงโครงสร้างของเก้าอี้นวดที่ออกแบบ

3.1 ส่วนประกอบของเก้าอี้นวดไฟฟ้าที่เคลื่อนที่ในแนวนอน

ในแนวแกน X นั้น จะใช้เพลาที่มีล้อสายพานและเกลียวซึ่งอยู่ที่ปลายทั้งสองข้างของเพลาเป็นอุปกรณ์หลัก โดยเกลียวทั้งสองข้างนั้นได้ถูกออกแบบให้เป็นเกลียวซ้าย และเกลียวขวาในแต่ละข้างของเพลา แขนนวดพร้อมกับลูกนวดจะถูกนำมาสวมที่เกลียวทั้งสองข้างนี้ มอเตอร์กระแสตรงจะถูกใช้ในการเคลื่อนที่ของลูกนวดของเก้าอี้นวดไฟฟ้า โดยจะติดตั้งสายพานระหว่างแกนของมอเตอร์และล้อสายพานของเพลา เมื่อมอเตอร์ดีซีหมุนในทิศทางทวนเข็มนาฬิกาจะทำให้แขนนวดเคลื่อนที่เข้าหากันในแนวแกน X และเมื่อมอเตอร์หมุนตามเข็มนาฬิกาก็จะทำให้แขนนวดเคลื่อนที่ออกจากกันในแนวแกน X



รูปที่ 3-2 แสดงถึงชุดนวดที่เคลื่อนที่ในแนวนอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากการออกแบบขนาดของเพลาของเก้าอี้นวด ต้องคำนึงถึงแรงกระทำที่เกิดขึ้นในชิ้นส่วนต่างๆ ซึ่งอาจทำให้ชิ้นส่วนแตกหักหรือเสียหายได้ ดังนั้นทฤษฎีที่เกี่ยวข้องกับการออกแบบเครื่องจักรจึงถูกนำมาประยุกต์ใช้ในโครงงานนี้ ในที่นี้เราจะใช้ทฤษฎีเรื่องโมเมนต์ดัด (Bending) เข้ามาช่วยในการออกแบบ โดยมีเงื่อนไขที่เกี่ยวข้องกับการออกแบบ ดังนี้

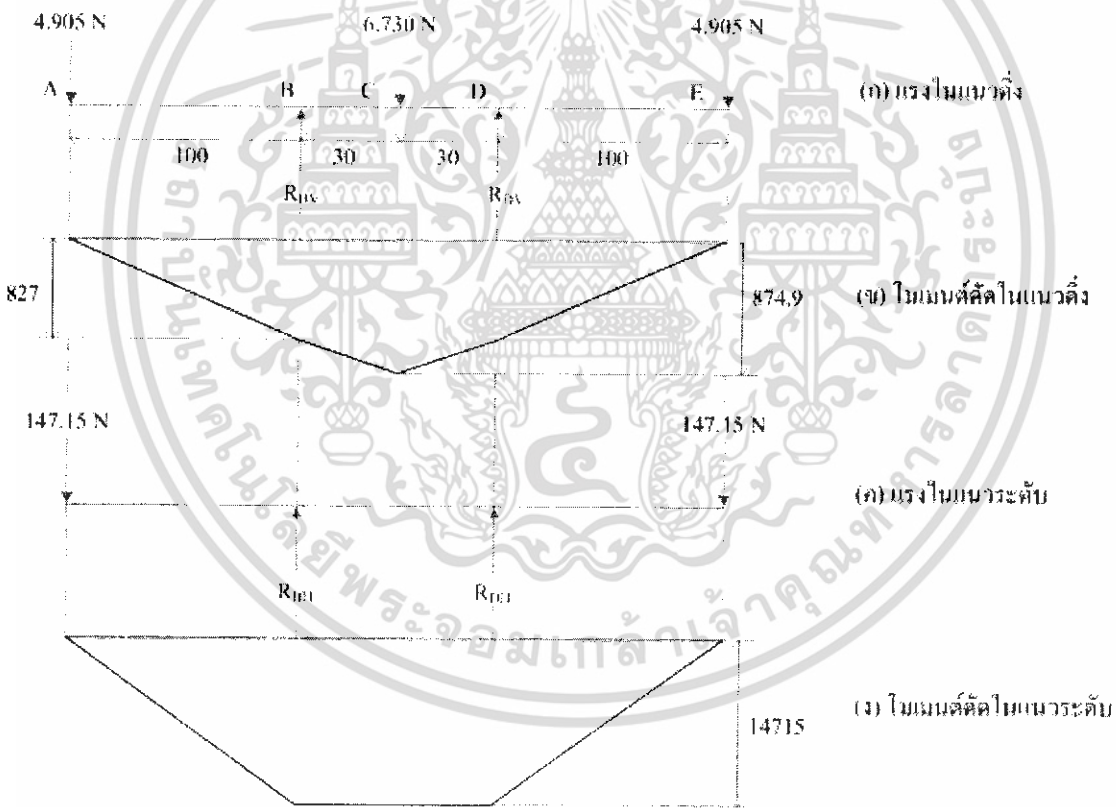
1. น้ำหนักของล้อสายพาน คือ 0.686 กิโลกรัม
2. น้ำหนักของแขนนวด คือ 0.5 กิโลกรัม
3. แขนนวดรับภาระสูงสุดข้างละ 15 กิโลกรัม
4. เส้นผ่านศูนย์กลางของล้อสายพาน คือ 70 มิลลิเมตร

จากรูปที่ 3-3 กำหนดให้

จุด A และ E คือ จุดที่รองรับน้ำหนักของแขนนวด

จุด B และ C คือ แบริ่ง

และ จุด D คือ ล้อสายพาน



รูปที่ 3-3 แสดงถึงแรงและโมเมนต์บิดในแนวดิ่งและแนวระดับของเพลา

ขั้นตอนแรก หาแรงปฏิกิริยาในแนวดิ่ง ดังนี้

แรงกระทำที่จุด A และ E = (0.5) (9.81) = 4.905 N

แรงกระทำที่จุด C = (0.686) (9.81) = 6.730 N

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รวมโมเมนต์ที่จุด B

$$(4.905)(100) + (R_{DV})(60) - (6.730)(30) - (4.905)(160) = 0$$

$$R_{DV} = 8.27 \text{ N}$$

$$R_{BV} = 4.905 + 6.730 + 4.905 - 8.27 = 8.27 \text{ N}$$

ดังนั้น โมเมนต์ดัดในแนวดิ่งที่จุด B และ D คือ

$$M_{BV} = 100\ R_{BV} = (100)(8.27) = 827 \text{ Nmm}$$

$$M_{DV} = 100\ R_{DV} = (100)(8.27) = 827 \text{ Nmm}$$

โมเมนต์ดัดในแนวดิ่งที่จุด C คือ

$$M_{CV} = (130)(6.730) = 874.9 \text{ Nmm}$$

ซึ่งแสดงไว้ดังรูปที่ 3-3 (ก) และ (ข)

จากนั้นจึงหาแรงปฏิกิริยาในแนวระดับเป็นขั้นต่อไป

แรงกระทำที่จุด A และ E = (15) (9.81) = 147.15 N

รวมโมเมนต์ที่จุด B

$$(147.15)(100) + (R_{DH})(60) - (147.15)(160) = 0$$

$$R_{DH} = 147.15 \text{ N}$$

$$R_{BH} = 147.15 \text{ N}$$

ดังนั้น โมเมนต์ดัดในแนวดิ่งที่จุด B และ D คือ

$$M_{BH} = 100\ R_{BH} = (100)(147.15) = 14715 \text{ Nmm}$$

$$M_{DH} = 100\ R_{DH} = (100)(147.15) = 14715 \text{ Nmm}$$

ดังแสดงไว้ในรูปที่ 3-3 (ค) และ (ง)

จากการพิจารณารูปที่ 3-3 (ข) และ (ง) จะเห็นว่าโมเมนต์ดัดสูงสุดจะเกิดขึ้นที่ตำแหน่ง B หรือ D จึงต้องรวมโมเมนต์ด้วยวิธีเวกเตอร์ที่สองตำแหน่งดังกล่าว

$$M_B = M_D = \sqrt{M_{DV}^2 + M_{DH}^2} = (827^2 + 14715^2)^{1/2} = 14738.22 \text{ Nmm}$$

โมเมนต์บิดเนื่องจากการส่งกำลังจากสายพาน คือ

$$T = FR$$

$$T = (30)(9.81)(70/2)$$

$$T = 10300.5 \text{ Nmm}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิดของแรง	C_m	C_t
เพลาอยู่นิ่ง :		
แรงสม่ำเสมอหรือเพิ่มขึ้นช้าๆ	1.0	1.0
แรงกระตุก	1.5 – 2.0	1.5 – 2.0
เพลาหมุน :		
แรงสม่ำเสมอหรือเพิ่มขึ้นช้าๆ	1.5	1.0
แรงกระตุกอย่างเบา	1.5 – 2.0	1.0 – 1.5
แรงกระตุกอย่างแรง	2.0 – 3.0	1.5 – 3.0

ตารางที่ 3-1 แสดงถึงค่าตัวประกอบความล้า

จากตารางที่ 3-1 เลือกเพลาหมุนแบบแรงสม่ำเสมอหรือเพิ่มขึ้นช้าๆ ดังนั้น

$C_m = 1.5$ และ $C_t = 1.0$

จากโค้ดของ ASME ระบุเอาไว้ว่าเพลาซึ่งมีใช้อยู่ในงานธรรมดาทั่วไป ควรจะมีค่าความเค้นเฉือนใช้งานดังนี้

$\tau_d = 55\ N/mm^2$ สำหรับเพลาที่ไม่มีร่องลิ่ม

$\tau_d = 41\ N/mm^2$ สำหรับเพลาที่มีร่องลิ่ม

ในที่นี้เลือกเพลาแบบไม่มีร่องลิ่ม ดังนั้น $\tau_d = 55\ N/mm^2$

จาก $$d^3 = \frac{16}{\pi\tau}\left((C_t T)^2 + (C_m M)^2\right)^{1/2}$$

$$d^3 = \frac{16}{55\pi}\left((1.0\times 10300.5)^2 + (1.5\times 14738.22)^2\right)^{1/2}$$

$d = 13.12$ mm

เพื่อความปลอดภัยในการออกแบบ เลือกใช้เพลาขนาดเส้นผ่านศูนย์กลาง 14 มิลลิเมตร

และในการออกแบบเพื่อหาขนาดของมอเตอร์ดีซี ต้องการออกแบบให้มอเตอร์มีกำลังเพียงพอต่อการขับชุดนวด ซึ่งมีเงื่อนไขและสมการที่เกี่ยวข้องดังนี้

1. ภาระสูงสุดที่ชุดนวดสามารถรับได้ คือ 50 กิโลกรัม ซึ่งกระทำตามแนวแกน Z
2. เส้นผ่านศูนย์กลางของเพลา คือ 0.014 เมตร
3. สัมประสิทธิ์ของแรงเสียดทาน คือ 0.08
4. ระยะพิตช์ คือ 4 มิลลิเมตร
5. เส้นผ่านศูนย์กลางของล้อสายพาน คือ 70 มิลลิเมตร
6. เส้นผ่านศูนย์กลางของแกนมอเตอร์ คือ 20 มิลลิเมตร

ในการคำนวณหาแรงบิด (Torque) สามารถนำสมการที่ (2.6) และ (2.7) มาประยุกต์ใช้ได้ โดยเป็นการเลื่อนภาระไปทางขวาและซ้ายตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$d_m = d - p/2 = 14 - 2 = 12$ mm $= 0.012$ m

$l = np = (1)(4) = 4$ mm $= 0.004$ m

$F = (50)(9.81) = 490.5$ N

จาก $$T_R = \frac{Fd_m}{2}\left(\frac{l + \pi\mu d_m}{\pi d_m - \mu l}\right)$$

$$T_R = \frac{490.5 \times 0.012}{2}\left(\frac{0.004 + \pi \times 0.08 \times 0.012}{\pi \times 0.012 - 0.08 \times 0.004}\right)$$

$T_R = 0.55$ Nm

จาก $$T_L = \frac{Fd_m}{2}\left(\frac{\pi\mu d_m - l}{\pi d_m + \mu l}\right)$$

$$T_L = \frac{490.5 \times 0.012}{2}\left(\frac{\pi \times 0.08 \times 0.012 - 0.004}{\pi \times 0.012 + 0.08 \times 0.004}\right)$$

$T_L = -0.076$ Nm

ค่าที่ได้ต้องนำมารวมกับแรงบิดที่เกิดจาก Trust Collar ซึ่งมีเส้นผ่านศูนย์กลาง (d_c) 0.025 เมตร

จาก $$T_c = \frac{F\mu_c d_c}{2}$$

$$T_c = \frac{490.5 \times 0.08 \times 0.025}{2}$$

$T_c = 0.491$ Nm

ดังนั้น $T_R = 0.55 + 0.491 = 1.041$ Nm

และ $T_L = -0.076 + 0.491 = 0.415$ Nm

ต้องการให้ล้อสายพานหมุนด้วยความเร็วรอบ 20 รอบต่อนาที สามารถหาความเร็วรอบของมอเตอร์ได้ ดังนี้

จาก $$\frac{d_1}{d_2} = \frac{N_2}{N_1}$$

$N_2 = (0.07/0.02)(20)$

$N_2 = 70$ รอบต่อนาที

จาก $P = T\omega = T(2\pi N/60)$

เลือกใช้ค่าแรงบิดมากที่สุด คือ 1.041 Nm เพื่อหาขนาดของมอเตอร์

$P = (1.041)(2\pi \times 70/60)$

$P = 7.631$ Watts

เพื่อความปลอดภัยในการออกแบบ จึงเลือกใช้มอเตอร์ขนาด 175 โวลต์ ซึ่งกินกระแสสูงสุด 210 มิลลิแอมแปร์

จาก $P = IV$

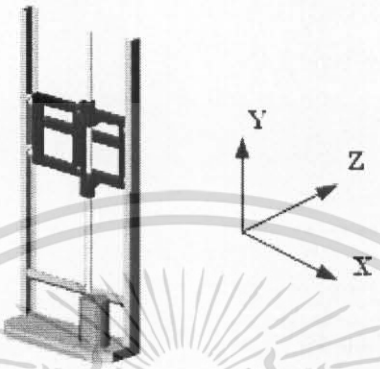
$P = (0.210)(175)$

$P = 36.750$ Watts

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

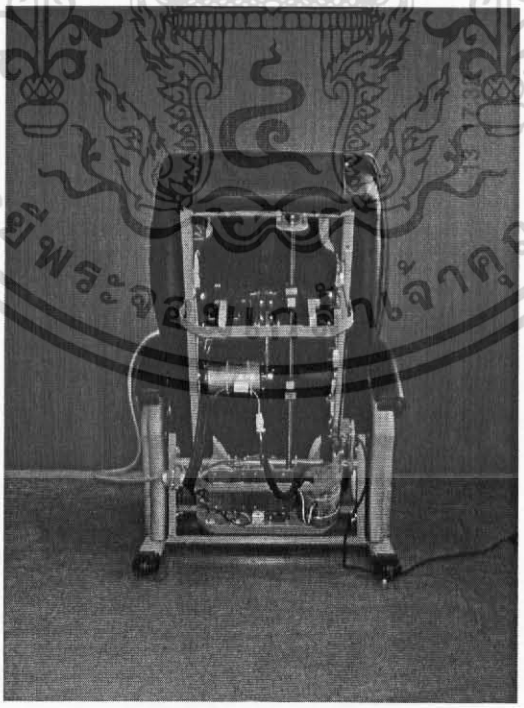
3.2 ส่วนประกอบของเก้าอี้นวดไฟฟ้าที่เคลื่อนที่ในแนวตั้ง

ในแนวแกน Y นั้น จะมีฉากที่ใช้รองรับชุดนวดซึ่งเคลื่อนที่ในแนวนอนดังกล่าวสวมอยู่กับสกรูส่งกำลังในแนวตั้ง มอเตอร์กระแสสลับจะถูกใช้ในควบคุมการหมุนของสกรูส่งกำลังโดยผ่านสายพานเช่นเดียวกัน เมื่อมอเตอร์เอซีหมุนตามเข็มนาฬิกาจะทำให้ฉากรองรับชุดนวดเคลื่อนที่ขึ้นในแนวตั้ง ในทางกลับกันถ้ามอเตอร์หมุนทวนเข็มนาฬิกาก็จะทำให้ฉากรองรับชุดนวดเคลื่อนที่ลง



รูปที่ 3-4 แสดงถึงฉากรองรับชุดนวดที่เคลื่อนที่ในแนวตั้ง

ลูกนวดซึ่งอยู่ที่ปลายของแขนนวดนั้นจะถูกติดตั้งร่วมกับมอเตอร์ดีซีแบบทดรอบ หรือเกียร์มอเตอร์ เพื่อที่จะให้แรงบิดสูงเพียงพอ โดยลักษณะของการติดตั้งจะเป็นแบบเยื้องศูนย์ (Eccentric)



รูปที่ 3-5 แสดงถึงเก้าอี้นวดไฟฟ้าที่ประกอบเสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบระบบควบคุมและโปรแกรมควบคุมการทำงาน

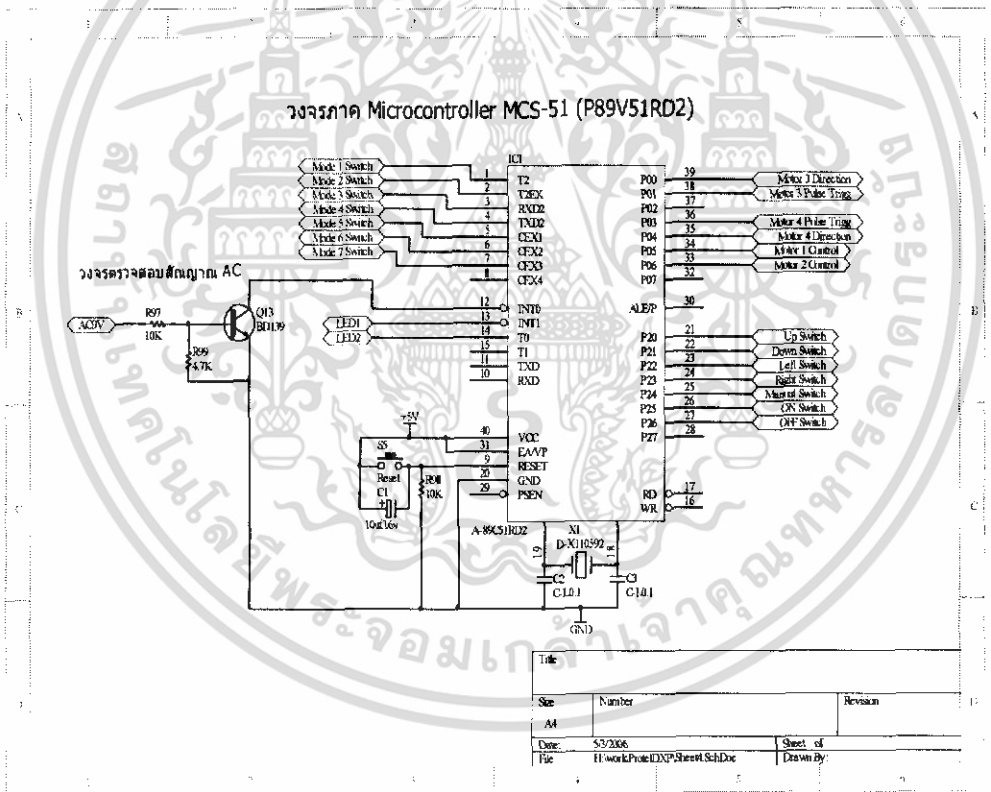
4.1 การออกแบบวงจรไฟฟ้า

ในการออกแบบวงจรไฟฟ้าจะใช้ Protel 99 SE ในการออกแบบวงจร การออกแบบแผ่นปริ้นท์จะแยกออกเป็น 3 ส่วน ดังนี้

1. แผงวงจรไมโครคอนโทรลเลอร์ MCS-51 ซึ่งใช้ไมโครคอนโทรลเลอร์เบอร์ P89V51RD2

2. แผงวงจรขับมอเตอร์กระแสตรงแบบทดรอบ 2 ตัว ที่ใช้ในการขับลูกนวด และแผงวงจรแหล่งจ่ายไฟ ซึ่งอยู่บนแผงวงจรเดียวกัน

3. แผงวงจรขับมอเตอร์กระแสตรงและมอเตอร์กระแสสลับ ที่ใช้ในการเคลื่อนที่ 2 แนวแกน ซึ่งอยู่บนแผงวงจรเดียวกัน



รูปที่ 4-1 แสดงถึงวงจรไมโครคอนโทรลเลอร์ MCS-51

วงจรนี้จะแบ่งออกเป็น 2 ส่วน คือภาคไมโครคอนโทรลเลอร์ กับ ภาคตรวจสอบสัญญาณ AC

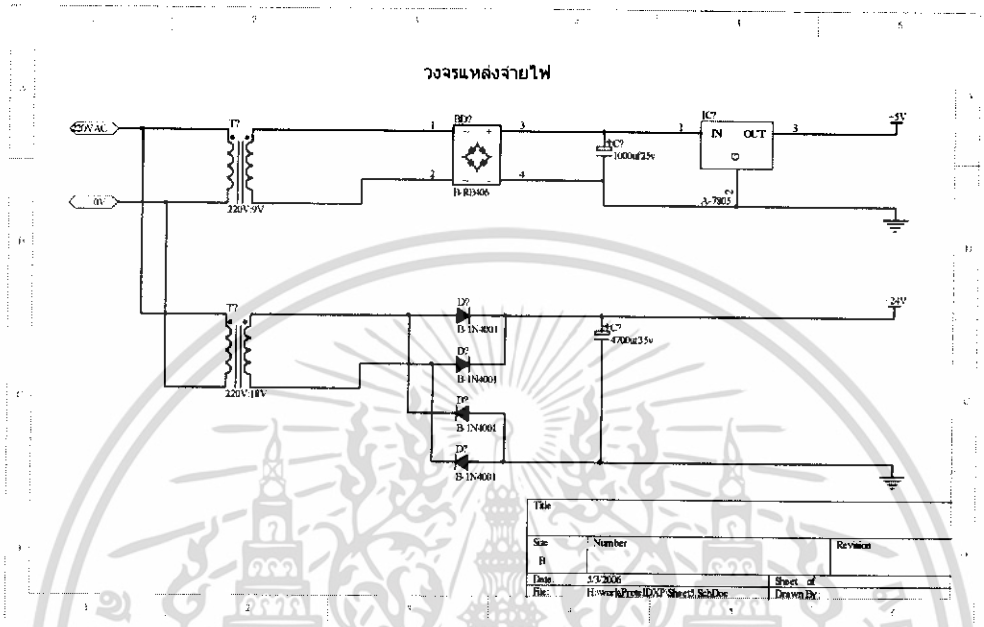
1. วงจรไมโครคอนโทรลเลอร์ คือ ส่วนที่จะสั่งการทำงานควบคุมอุปกรณ์ทุกส่วน ซึ่งจะมีอุปกรณ์ ดังนี้

- ตัวประมวลผล คือ P89V51RD2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- วงจรกำเนิดความถี่ ได้แก่ X-TAL 11.0592 MHz, C เซรามิก 22 pf 50 V
- วงจร Reset ประกอบไปด้วย S5 Push Button Switch, C 10 µf 16 V, R 10 kΩ

2. วงจรตรวจสอบสัญญาณ AC จะทำหน้าที่เช็คคลื่นของไฟ AC แล้วส่งไปที่คอนโทรลเลอร์เพื่อควบคุมความเร็วของมอเตอร์ DC 175 V ซึ่งประกอบไปด้วย Transistor BD139, R 10 kΩ, R 4.7 kΩ



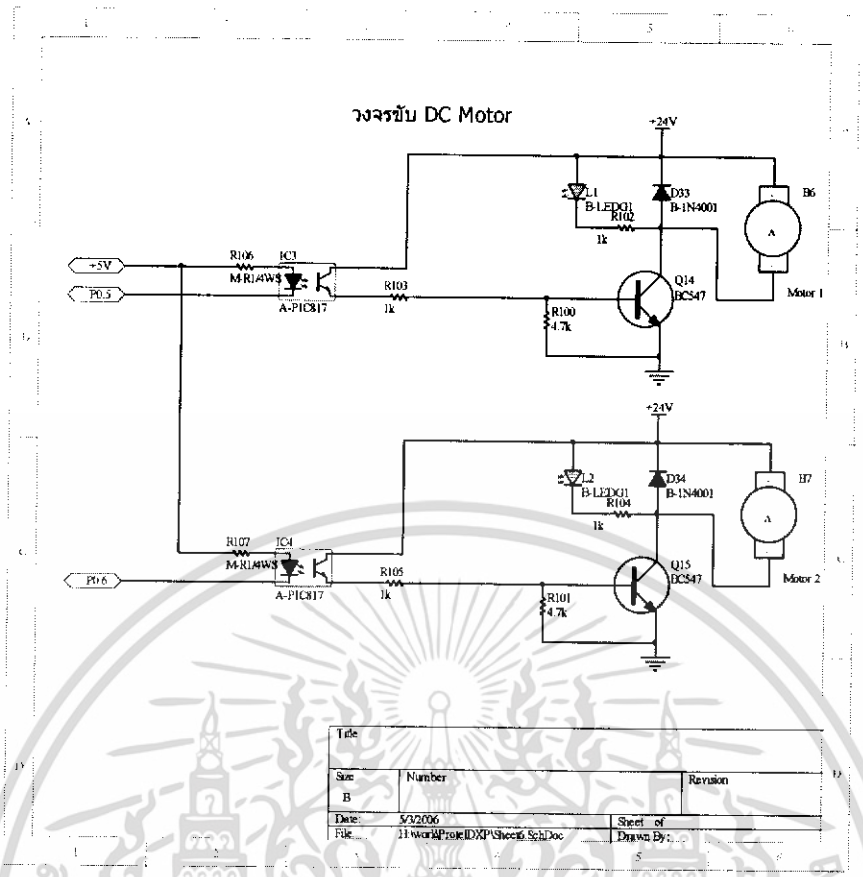
รูปที่ 4-2 แสดงถึงวงจรแหล่งจ่ายไฟ

ในส่วนนี้จะเป็นวงจรจ่ายไฟแบ่งออกเป็น 2 ชุดคือชุดจ่ายไฟ 5 โวลต์ กับ 24 โวลต์

1. ชุดจ่ายไฟ 5 โวลต์ ประกอบไปด้วย หม้อแปลง 9 VAC ทำหน้าที่แปลงไฟจาก 220 VAC ให้เป็น 9 VAC จากนั้นก็จะเข้าบริดจ์ไดโอดเพื่อเปลี่ยนจากไฟ AC เป็น DC และจะถูก Filter ด้วย C 1000 µf 25 V หลังจาก Filter เสร็จก็จะได้ไฟ DC ประมาณ 12.7 V ซึ่งได้จากสูตร 9×1.414 จากนั้นก็จะผ่านตัว Regulator 7805 ที่จะทำหน้าที่แปลงไฟให้เหลือ 5 V เพื่อไปจ่ายให้วงจรไมโครคอนโทรลเลอร์

2. ชุดจ่ายไฟ 24 โวลต์ ประกอบไปด้วย หม้อแปลง 24 VAC ทำหน้าที่แปลงไฟจาก 220 VAC ให้เป็น 24 VAC จากนั้นก็จะเข้าบริดจ์ไดโอดเพื่อเปลี่ยนจากไฟ AC เป็น DC และจะถูก Filter ด้วย C 4700 µf 25 V หลังจาก Filter เสร็จก็จะได้ไฟ DC ประมาณ 25.4 V ซึ่งได้จากสูตร 18×1.414 และก็จะไปจ่ายให้ชุดขับมอเตอร์ DC 24 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



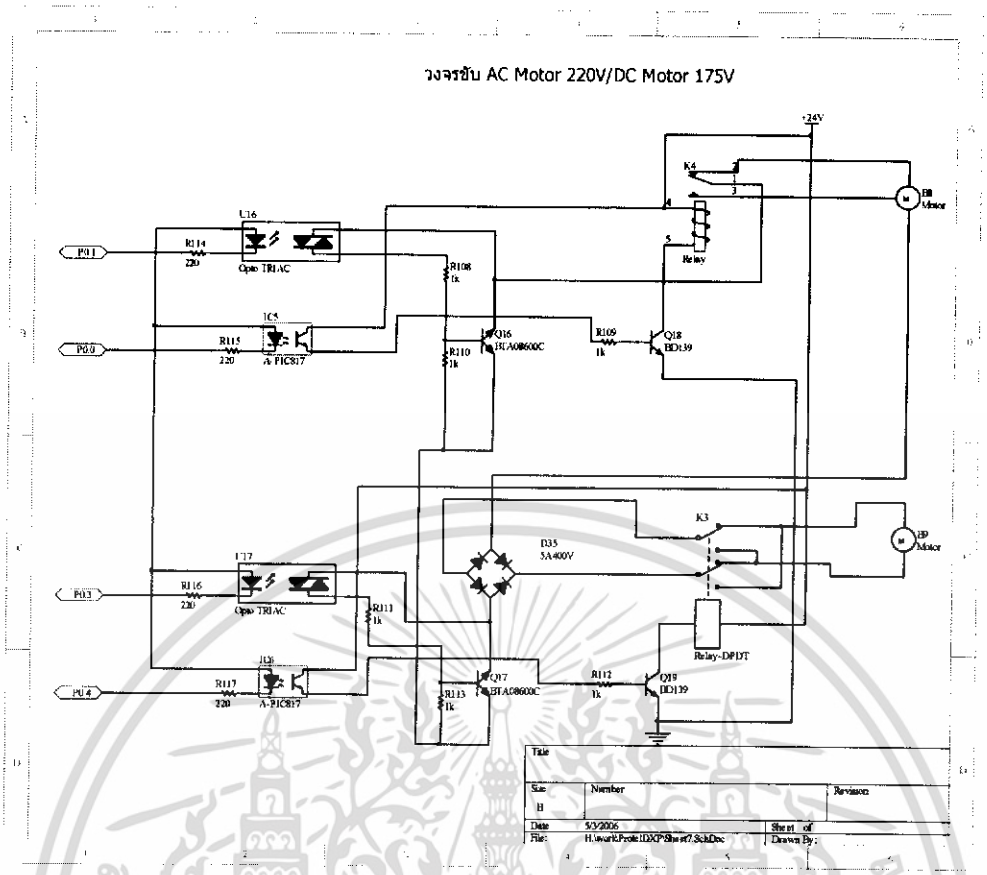
รูปที่ 4-3 แสดงถึงวงจรขับมอเตอร์กระแสตรงแบบทครอบ

ในส่วนนี้จะเป็นวงจรขับมอเตอร์ขนาด 24 VDC 2 ชุดที่เหมือนกัน ซึ่งจะมีส่วนประกอบดังนี้

1. Opto Isolator ประกอบไปด้วย PC817 กับ R 470 โอห์มในส่วนนี้จะทำหน้าที่แยกกราวด์ระหว่างชุดไมโครคอนโทรลเลอร์กับชุดขับมอเตอร์

2. Power Transistor ประกอบไปด้วย BC547 และมี R 1 kΩ กับ R 4.7 kΩ เป็นตัวจัดกระแสไบอัสให้กับ Transistor และยังมีวงจรกัน Back EMF จากมอเตอร์โดยใช้ Diode 1N4001 ส่วน LED สีเขียวกับ R 1 kΩ เป็นตัวแสดงผลการทำงานของชุดขับมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-4 แสดงถึงวงจรขับมอเตอร์กระแสตรงและมอเตอร์กระแสสลับ

ในส่วนนี้จะเป็นวงจรขับมอเตอร์ AC 220 V กับ มอเตอร์ DC 175 V

1. มอเตอร์ AC 220 V จะแบ่งเป็น 2 ส่วน คือ ส่วนเปิด-ปิด กับส่วนกลับทางหมุน

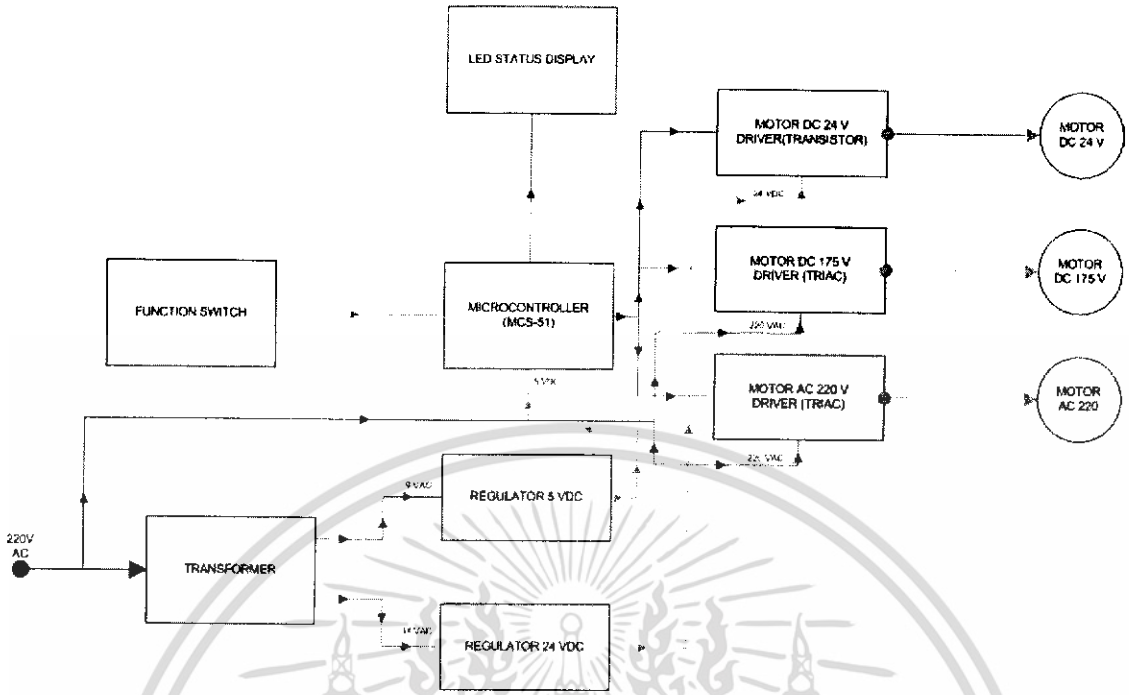
ส่วนประกอบของการเปิด-ปิด จะมี MOC3023 กับ R 220 โอห์ม เป็นวงจร Isolator และมี BTA08600C ซึ่งเป็น TRIAC จะทำหน้าที่เปิด-ปิดไฟให้กับมอเตอร์โดยจะมี R 1 kΩ เป็นตัวจัดทริค

ส่วนวงจรกลับทางจะใช้ Relay เป็นตัวสลับขดลวดของมอเตอร์ให้หมุนกลับทางได้ โดยจะมี Transistor BD139 เป็นตัวขับ Relay และมี PC817 เป็นวงจร Isolator ระหว่างวงจรไมโครคอนโทรลเลอร์กับวงจรขับ Relay

2. ในส่วนของวงจรขับ มอเตอร์ DC 175 V จะมีส่วนประกอบเหมือนกันกับ มอเตอร์ AC 220 V แต่จะเพิ่มตัว Diode Bridge 5 A 400 V กับ C 100 μf 400 V มาแปลงไฟจาก AC เป็น DC ก่อนเข้า Relay เพื่อกลับทางหมุน แล้วเข้าจึงมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BLOCK DIAGRAM



รูปที่ 4-5 แสดงถึงการควบคุมการทำงานของเก้าอี้นวดไฟฟ้า

ผู้ใช้งานเก้าอี้นวดไฟฟ้าสามารถสั่งการผ่านทางรีโมต ซึ่งอินเตอร์เฟสอยู่กับไมโครคอนโทรลเลอร์ MCS-51 วงจรไมโครคอนโทรลเลอร์จะสั่งการผ่านวงจรไดร์ฟมอเตอร์ทั้ง 3 วงจร มอเตอร์กระแสสลับจะเป็นการควบคุมการเคลื่อนที่ตามแนวตั้ง ส่วนมอเตอร์กระแสตรงจะเป็นการควบคุมการเคลื่อนที่ตามแนวนอน และยังมีมอเตอร์กระแสตรงอีกวงจรเพื่อควบคุมการหมุนของลูกนวด

4.2 การออกแบบโปรแกรมการสั่งงาน

เนื่องจากวัตถุประสงค์หลักของโครงงานนี้ คือ การออกแบบเก้าอี้นวดที่สามารถปรับลักษณะการทำงานได้ ลักษณะการทำงานนั้นจะแปรเปลี่ยนไปตามลักษณะรูปร่างของผู้ใช้งาน ซึ่งมีตัวแปรต้นที่สนใจ คือ น้ำหนักและส่วนสูงของผู้ใช้งาน ส่วนตัวแปรตาม คือ ความกว้างของแผ่นหลังของผู้ใช้ แน่นอนว่าสมการที่ใช้หาความสัมพันธ์ของน้ำหนัก ส่วนสูง และความกว้างของแผ่นหลังนั้น ไม่มีสมการหรือกฎเกณฑ์ที่ตายตัว ดังนั้นจึงใช้วิธีการดังนี้

ในขั้นตอนแรกของการคำนวณจะอ้างอิงจากตารางที่ 4-1 ซึ่งแสดงถึงน้ำหนักเฉลี่ยในช่วงความสูงต่างๆ โดยการรวบรวมข้อมูลความกว้างของแผ่นหลังตามช่วงส่วนสูงและน้ำหนักในแต่ละแถว โดยรวบรวมให้ได้ช่วงละ 20 ตัวอย่าง จากนั้นจึงหาค่าเฉลี่ยของความกว้างของแผ่นหลังในแต่ละช่วง เมื่อได้ค่าที่ต้องการแล้วจึงนำไปออกแบบโปรแกรมของเก้าอี้นวด โดยกำหนดค่าคงที่ต่างๆกัน ทำให้ช่วงระยะการนวดของแผ่นหลังในแนวนอนมีค่าไม่เท่ากันขึ้นอยู่กับสรีระของผู้ใช้งาน

สำหรับซอร์สโค้ดของโปรแกรมดูได้จากภาคผนวกท้ายเล่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสูง (เซนติเมตร)	คนไทย	น้ำหนัก(กิโลกรัม)			
		องค์การอนามัยโลก	โครงร่างคนอเมริกัน(พศ.2523)		
			เล็ก	กลาง	ใหญ่
ชาย					
150-154	45-51	54-57	-	-	-
155-159	49-55	57-60	56-60	57-63	59-68
160-164	52-58	60-63	58-62	59-66	62-71
165-169	55-63	63-66	59-64	61-68	64-74
170-174	58-66	67-69	61-67	62-71	66-78
175-179	59-67	70-73	63-70	66-74	68-81
หญิง					
140-144	39-45	45-47	-	-	-
145-149	44-50	47-50	44-50	48-55	52-60
150-154	46-52	50-52	46-53	49-58	53-63
155-159	49-57	53-55	47-55	51-60	56-66
160-164	52-60	56-59	49-58	54-63	58-70
165-169	53-65	59-62	52-61	57-66	61-73

ตารางที่ 4-1 แสดงถึงน้ำหนักเฉลี่ยของผู้ใหญ่ไทยปกติในช่วงความสูงต่างๆ เปรียบเทียบกับมาตรฐานขององค์การอนามัยโลกและของคนอเมริกัน

ความสูง(เซนติเมตร)	น้ำหนัก(กิโลกรัม)	ความกว้างแผ่นหลัง(เซนติเมตร)	ค่าเฉลี่ย ($\bar{x}$)
ชาย			
150-154	45-51	-	-
155-159	49-55	-	-
160-164	52-58	26,29,29,29,29,29.5,29.5,29.5,30,30, 30,31,31,31,31.5,31.5,31.5,33,34,34	28.750
165-169	55-63	28,28,28,28,30,30.5,31,31,31,31, 32.5,33,33,33,33,33,33,34,34,34.5	31.475
170-174	58-66	29.5,30,30,30,31,32,32,32,32.5,32.5, 32.5,33,33,33,33,33,33,34,35,35,	32.300
175-179	59-67	28,29,31,31,31,31,31.5,31.5,32,32 32.5,33,33,33.5,33.5,34,34,34,34,35	32.225
หญิง			
140-144	39-45	-	-
145-149	44-50	-	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

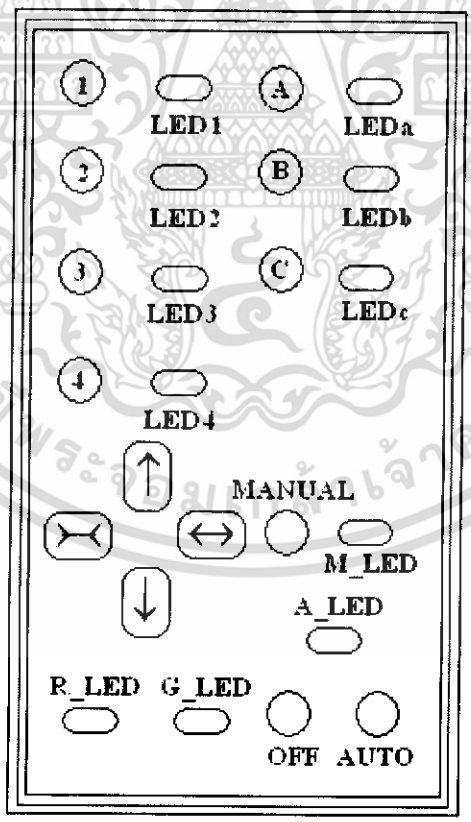
150-154	46-52	24,25,25,25.5,25.5,25.5,25.5,26,26,26.5, 26.5,26.5,27,27,27,27,27,27,27,28	26.225
155-159	49-57	25,25,26,26,26,26,26.5,26.5,27,27, 28,28,28,28,28,28,28,28.5,28.5,28.5	27.125
160-164	52-60	25.5,27,27,27,27,27,27,27,27.5,27.5, 27.5,28,28,28,29,29,29,29,29,29	27.750
165-169	53-65	-	-

ตารางที่ 4-2 แสดงถึงการหาค่าเฉลี่ยความกว้างของแผ่นหลังที่ส่วนสูงและน้ำหนักต่างๆ

4.3 การใช้งานเก้าอี้นวดไฟฟ้า

ในการควบคุมการทำงานของเก้าอี้นวดไฟฟ้านั้น ผู้จัดทำได้ออกแบบไว้ให้ควบคุมโดยรีโมตสวิตซ์แบบมีสาย เพื่อความสะดวกของผู้ใช้งาน

ลักษณะของรีโมตควบคุมดังแสดงในรูปที่ 4-6 โดยอธิบายการทำงานของสวิตซ์แต่ละปุ่มดังนี้



รูปที่ 4-6 แสดงถึงรีโมตควบคุมการทำงานของเก้าอี้นวด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ปุ่ม AUTO กดเมื่อต้องการเข้าสู่โหมดการทำงานอัตโนมัติ เมื่อกดปุ่มนี้แล้ว A_LED จะติดค้างอยู่ตลอดและสามารถเลือกกดปุ่ม 1-4 หรือ A-C ได้ ขึ้นอยู่กับว่าผู้ใช้งานต้องการลักษณะการนวดแบบใด

2. ปุ่ม 1 เมื่อกดปุ่มนี้แล้ว ลูกนวดจะนวดในแนวแกนนอน โดยที่ชุดนวดจอยู่ที่ตำแหน่งบนสุด เปรียบเสมือนการนวดไหล่นั่นเอง หากต้องการจะเปลี่ยนการทำงานอัตโนมัติแบบอื่นก็สามารถกดปุ่มอื่นได้ทันที

3. ปุ่ม 2 เมื่อกดปุ่มนี้แล้ว ลูกนวดจะนวดในแนวแกนนอน โดยที่ชุดนวดจอยู่ที่ตำแหน่งล่างสุด เปรียบเสมือนการนวดเอวนั่นเอง หากต้องการจะเปลี่ยนการทำงานอัตโนมัติแบบอื่นก็สามารถกดปุ่มอื่นได้ทันที

4. ปุ่ม 3 เมื่อกดปุ่มนี้แล้ว ลูกนวดจะนวดในแนวแกนตั้ง โดยที่ลูกนวดจะอยู่ที่ตำแหน่งนอกสุด หากต้องการจะเปลี่ยนการทำงานอัตโนมัติแบบอื่นก็สามารถกดปุ่มอื่นได้ทันที

5. ปุ่ม 4 เมื่อกดปุ่มนี้แล้ว ลูกนวดจะนวดในแนวแกนตั้ง โดยที่ลูกนวดจะอยู่ที่ตำแหน่งในสุด หากต้องการจะเปลี่ยนการทำงานอัตโนมัติแบบอื่นก็สามารถกดปุ่มอื่นได้ทันที

6. ปุ่ม A เมื่อกดปุ่มนี้แล้ว ชุดนวดจะเคลื่อนที่ขึ้นลงในแนวแกนตั้ง พร้อมกับลูกนวดก็จะเคลื่อนที่เข้า-ออก ในแนวแกนนอนเป็นระยะ 5 เซนติเมตรไปพร้อมๆกัน เหมาะสำหรับผู้ใช้งานที่มีความกว้างของแผ่นหลังมาก หากต้องการจะเปลี่ยนการทำงานอัตโนมัติแบบอื่นก็สามารถกดปุ่มอื่นได้ทันที

7. ปุ่ม B เมื่อกดปุ่มนี้แล้ว ชุดนวดจะเคลื่อนที่ขึ้นลงในแนวแกนตั้ง พร้อมกับลูกนวดก็จะเคลื่อนที่เข้า-ออก ในแนวแกนนอนเป็นระยะ 4 เซนติเมตรไปพร้อมๆกัน เหมาะสำหรับผู้ใช้งานที่มีความกว้างของแผ่นหลังปานกลาง หากต้องการจะเปลี่ยนการทำงานอัตโนมัติแบบอื่นก็สามารถกดปุ่มอื่นได้ทันที

8. ปุ่ม C เมื่อกดปุ่มนี้แล้ว ชุดนวดจะเคลื่อนที่ขึ้นลงในแนวแกนตั้ง พร้อมกับลูกนวดก็จะเคลื่อนที่เข้า-ออก ในแนวแกนนอนเป็นระยะ 3 เซนติเมตรไปพร้อมๆกัน เหมาะสำหรับผู้ใช้งานที่มีความกว้างของแผ่นหลังน้อย หากต้องการจะเปลี่ยนการทำงานอัตโนมัติแบบอื่นก็สามารถกดปุ่มอื่นได้ทันที

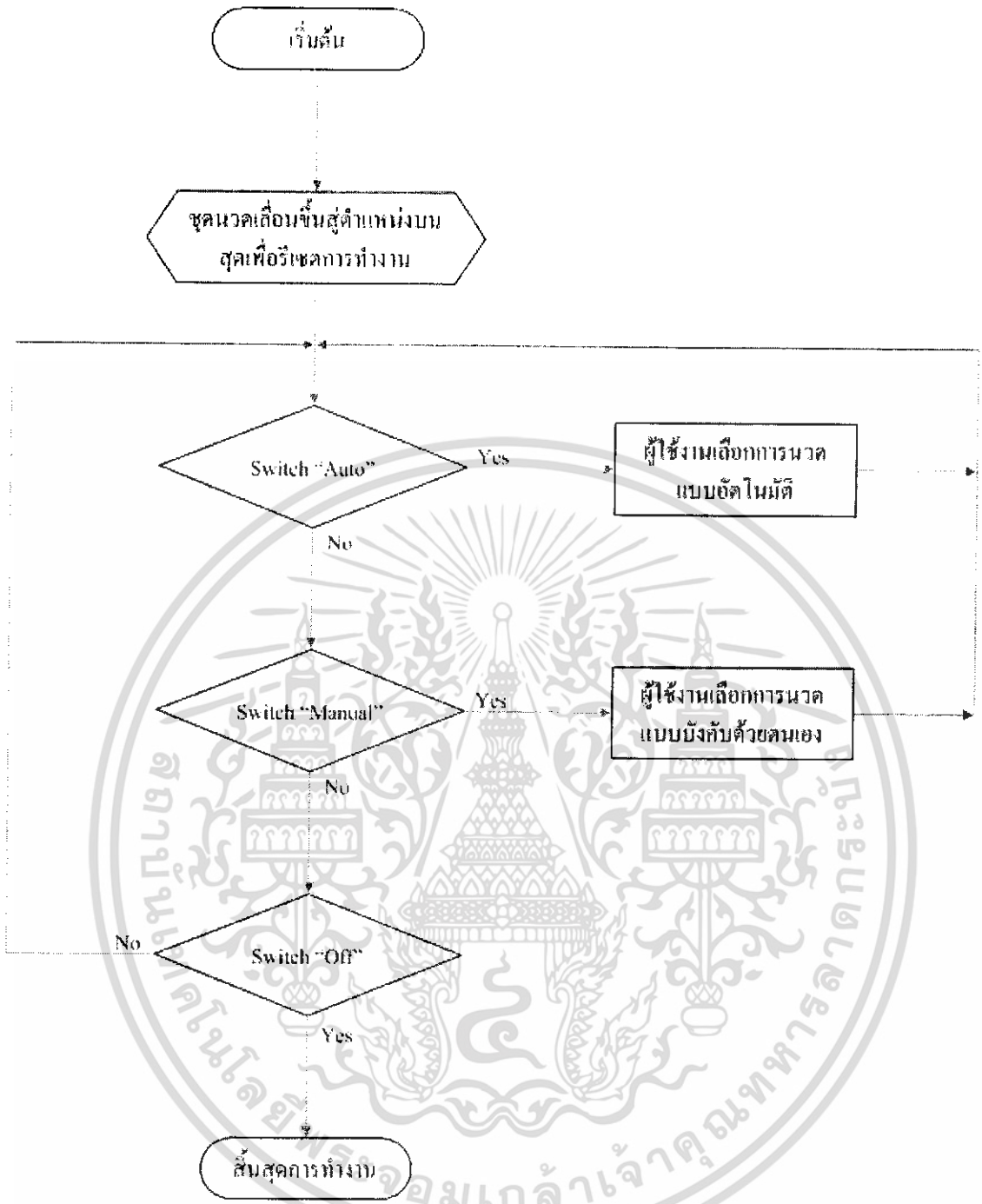
ปุ่ม 1-4 และปุ่ม A-C จะมี LED กำกับไว้ในแต่ละปุ่มเพื่อแสดงสถานะของการทำงาน

9. ปุ่ม MANUAL กดเมื่อต้องการเข้าสู่โหมดการทำงานแบบบังคับด้วยตัวเอง เมื่อกดปุ่มนี้แล้ว M_LED จะติดค้างอยู่ตลอดและสามารถเลือกกดปุ่มควบคุมทิศทางได้ ขึ้นอยู่กับว่าผู้ใช้งานต้องการลักษณะการนวดแบบใด หากต้องการจะเปลี่ยนการทำงานเป็นแบบอัตโนมัติก็สามารถกดปุ่ม AUTO ได้ทันที

10. ปุ่มควบคุมทิศทาง 4 ปุ่ม (ขึ้น-ลง-เข้า-ออก) ปุ่มเหล่านี้สามารถกดได้เมื่อผู้ใช้งานต้องกดปุ่ม MANUAL ก่อนเท่านั้น ในขณะที่กดปุ่มเหล่านี้ หากชุดนวดสามารถเลื่อนไปในทิศทางที่กดได้ G_LED ก็จะติดและถ้าชุดนวดไม่สามารถเลื่อนต่อไปได้อีก R_LED ก็จะติดแทนเพื่อให้ผู้ที่ใช้งานทราบ

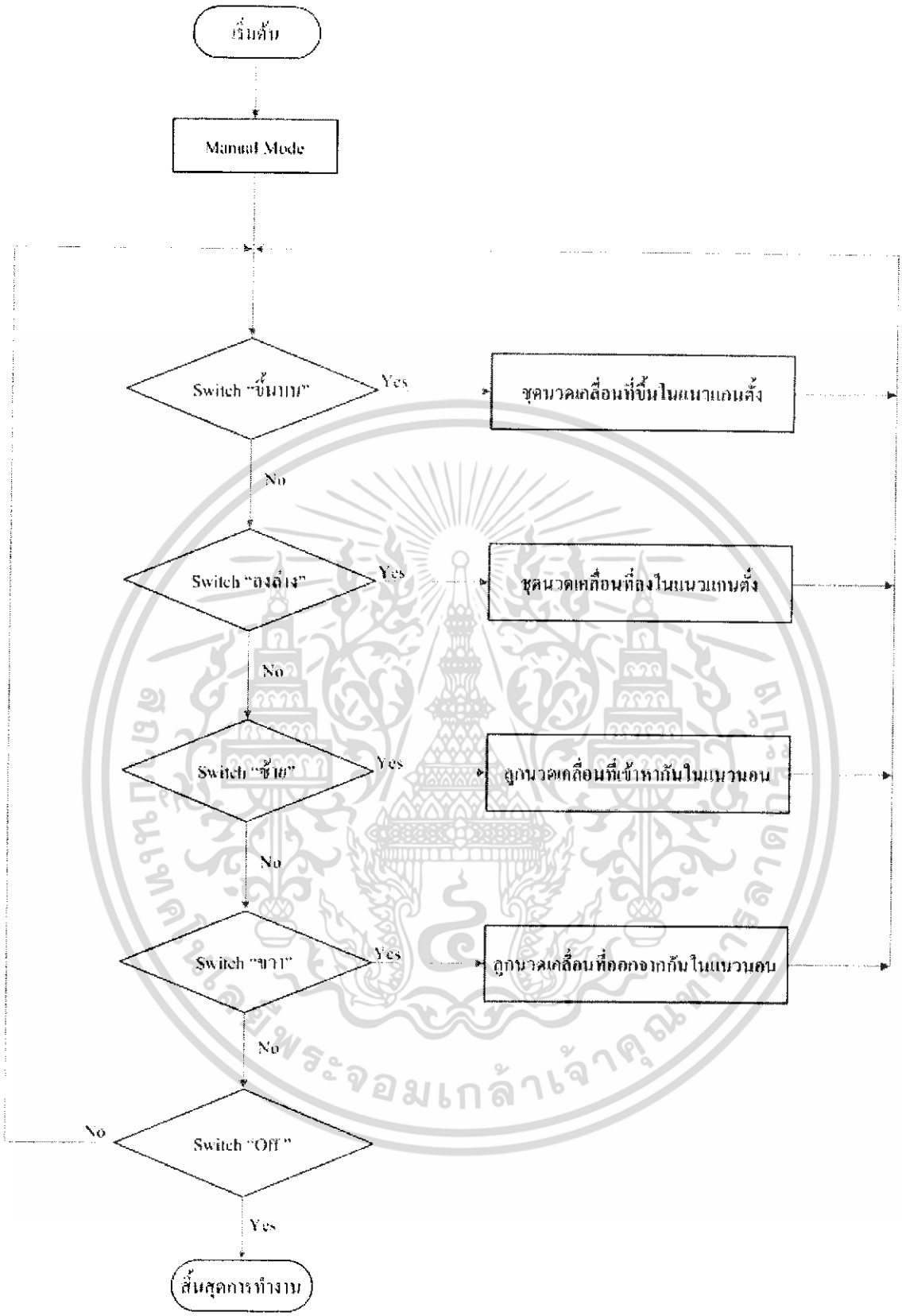
11. ปุ่ม OFF กดเมื่อต้องการหยุดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



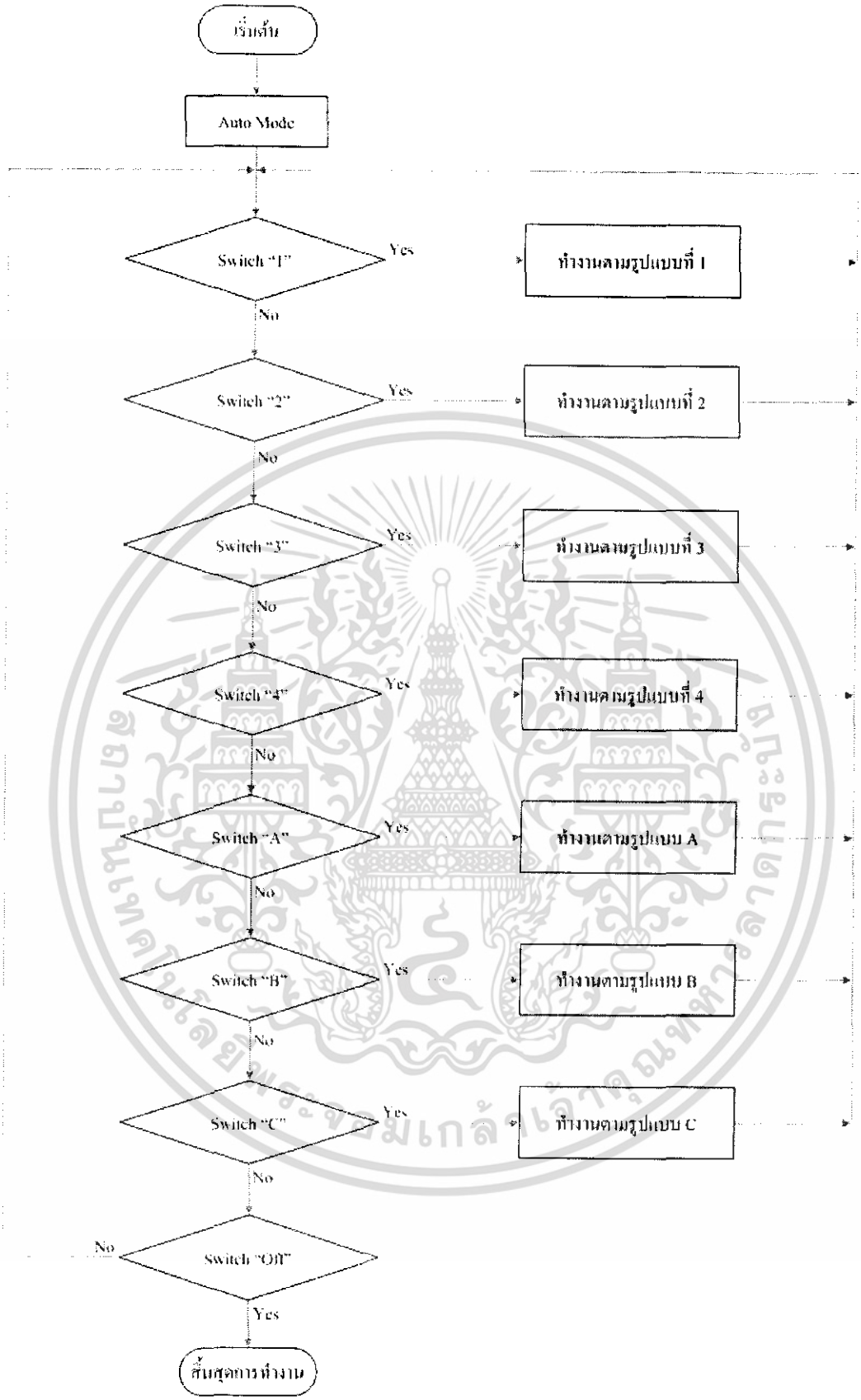
รูปที่ 4-7 แสดงถึงภาพรวมของการสั่งงานผ่านรีโมต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-8 แสดงถึงแผนผังลำดับการทำงานของโปรแกรมการทำงานแบบบังคับด้วยตนเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-9 แสดงถึงแผงผังลำดับการทำงานของโปรแกรมการทำงานแบบอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
รูปแบบและผลการทดลอง

5.1 รูปแบบการทดลอง

การทดลองของโครงงานนี้จะแบ่งเป็น 2 แบบ โดยยึดตามลักษณะการทำงานของเก้าอี้นวดเป็นหลัก ดังนี้

1. สำหรับการทำงานแบบบังคับด้วยตนเองหรือ Manual Mode จะเป็นการทดสอบระหว่างการทำงานของเก้าอี้นวดไฟฟ้าในสภาวะไร้น้ำหนัก และเมื่อมีน้ำหนักมากดทับ โดยทดลองช่วงค่าน้ำหนักต่างกันหลายค่า

2. สำหรับการทำงานแบบอัตโนมัติ เฉพาะปุ่ม A ถึงปุ่ม C จะเป็นการทดสอบระหว่างการทำงานของเก้าอี้นวดไฟฟ้าในสภาวะไร้น้ำหนัก และเมื่อมีน้ำหนักมากดทับเช่นเดียวกัน

5.2 ผลการทดลอง

1. สำหรับผลการทดลองของการทำงานแบบบังคับด้วยตนเอง ชุดนวดสามารถเคลื่อนได้ทั้ง 2 แนวแกนการเคลื่อนที่ โดยในสภาวะไร้ภาระจะเคลื่อนที่จากตำแหน่งบนสุดจนถึงตำแหน่งล่างสุดใช้เวลา 11.2 วินาที และจากตำแหน่งนอกสุดจนถึงตำแหน่งในสุดใช้เวลา 6.5 วินาที ระยะเวลาที่ใช้ในการเคลื่อนที่ของชุดนวดจะแปรผันตามภาระที่ได้รับ หรือกล่าวได้ว่ายิ่งน้ำหนักของผู้ใช้งานมากขึ้นเท่าใดก็จะใช้เวลาในการเคลื่อนที่มากขึ้นเท่านั้น โดยน้ำหนักที่เพิ่มขึ้นในช่วงแรกอาจจะมีผลกระทบเพียงเล็กน้อย และจะเพิ่มมากขึ้นเรื่อยๆตามตารางที่ 5-1

ภาระหรือน้ำหนักของผู้ใช้งาน (กิโลกรัม)	ระยะเวลาเฉลี่ย (วินาที)	
	บน-ล่าง	นอก-ใน
0	11.2	6.5
46-50	12.3	6.7
51-55	12.8	7.1
56-60	13.6	7.6
61-65	14.9	8.6
66-70	16.0	9.2
71-85	17.5	10.5
86-90	21.7	12.8

ตารางที่ 5-1 แสดงถึงระยะเวลาเฉลี่ยในการเคลื่อนที่ของชุดนวดที่ภาระต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ผลการทดลองสำหรับการทำงานแบบอัตโนมัติ ในสภาวะไร้น้ำหนักนั้นลูกนวดมีการเคลื่อนที่ตรงตามที่ได้ออกแบบโปรแกรมไว้ คือ 5 เซนติเมตร 4 เซนติเมตร และ 3 เซนติเมตร และเมื่อมีภาระมากระทำต่อลูกนวดจะทำให้ระยะการเคลื่อนที่ลดน้อยลงกว่าที่ได้ออกแบบไว้ เนื่องจากในการออกแบบนั้นได้ใช้ค่าคงที่ของเวลาไปกำหนดให้มอเตอร์ดีซีหมุน เมื่อมีภาระเพิ่มเข้ามาจึงทำให้จำนวนรอบที่มอเตอร์หมุนลดลง ในขณะที่เวลายังคงมีค่าเท่าเดิม

ข้อควรสังเกต สำหรับการเลือกการทำงานแบบ A จะเห็นว่าภาระที่เพิ่มขึ้นมาไม่ได้ส่งผลให้ระยะทางการเคลื่อนที่น้อยลง เนื่องจากเป็นระยะทางมากที่สุดที่ลูกนวดสามารถเคลื่อนที่ได้ ในการเขียนโปรแกรมจึงไม่มีความจำเป็นที่จะต้องกำหนดค่าคงที่ เพียงแต่สั่งให้ลูกนวดวิ่งไปเรื่อยๆจนกระทั่งชนกับลิมิตสวิตซ์ แล้วจึงวิ่งกลับมา

ภาระหรือน้ำหนักของผู้ใช้งาน(กิโลกรัม)	ระยะการเคลื่อนที่ (เซนติเมตร)		
	โปรแกรม A	โปรแกรม B	โปรแกรม C
0	5.0	4.0	3.0
46-50	5.0	3.9	3.0
51-55	5.0	3.9	2.9
56-60	5.0	3.8	2.7
61-65	5.0	3.5	2.6
66-70	5.0	3.2	2.5
71-85	5.0	3.0	2.2
86-90	5.0	2.4	1.9

ตารางที่ 5-2 แสดงถึงระยะการเคลื่อนที่ของลูกนวดในแบบบังคับอัตโนมัติ

ดังนั้นจึงเป็นการเหมาะสมตามที่ได้ออกแบบโปรแกรมไว้ กรณีที่ผู้ใช้งานมีน้ำหนักมากควรเลือกโปรแกรมการทำงานแบบ A และโปรแกรมแบบ B และ C ลดหลั่นลงมาตามน้ำหนักของผู้ใช้ ซึ่งสอดคล้องกับความกว้างของแผ่นหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
สรุปผลการดำเนินงาน

6.1 สรุปผลการดำเนินงาน

โครงงานนี้สามารถสร้างและออกแบบเก้าอี้นวดที่สามารถปรับลักษณะการทำงานได้ ซึ่งมีโปรแกรมการทำงานหลายรูปแบบ อีกทั้งยังมีโปรแกรมที่สามารถเปลี่ยนแปลงตามสรีระของผู้ใช้งานได้ โดยการทำงานเป็นไปตามเป้าหมายที่วางไว้

โปรแกรมเหล่านี้สามารถปรับปรุง หรือปรับเปลี่ยนได้ตลอดอายุการใช้งาน โดยผู้ที่ต้องการปรับปรุงสามารถเขียนซอร์สโค้ดสั่งงานใหม่และโปรแกรมลงในไมโครคอนโทรลเลอร์ได้โดยตรง โดยผ่านทางซีเรียพอร์ต

6.2 ปัญหาที่พบขณะดำเนินงานและแนวทางแก้ไข

1. ระยะทางการเคลื่อนที่ของลูกนวดในแนวนอนค่อนข้างสั้น เนื่องจากเกลียวทั้งสองข้างของเพลามีความยาวเพียง 50 มิลลิเมตรในแต่ละข้าง ทำให้ต้องระมัดระวังในการควบคุมเป็นอย่างมาก เพื่อมิให้เกิดความเสียหายต่อชุดนวด แนวทางการแก้ไขคือควรเพิ่มระยะการเคลื่อนที่ให้มากขึ้น ซึ่งอาจจะต้องออกแบบโครงสร้างของเก้าอี้นวดไฟฟ้าใหม่ทั้งหมด

2. เกียร์มอเตอร์อาจจะมีแรงบิดไม่เพียงพอ ถ้าผู้ใช้งานมีน้ำหนักมากเกินไป แนวทางการแก้ไขคือการหาซื้อมอเตอร์ใหม่ที่มีแรงบิดมากกว่าเดิม แต่ทั้งนี้ต้องคำนึงถึงขนาดของมอเตอร์ด้วยเช่นกัน ถ้าไม่สามารถหามอเตอร์ที่มีขนาดเหมาะสมได้ อาจจะต้องใช้การทดรอบโดยต้องออกแบบชุดนวดนวดใหม่เช่นเดียวกันกับข้อแรก

3. เนื่องจากวงจรควบคุมต้องควบคุมอุปกรณ์จำนวนมากในเครื่อง จึงมีแผงวงจรหลายแผงและมีสายไฟเป็นจำนวนมากโยงจากส่วนต่างๆของเครื่องเข้าสู่แผงควบคุม และนอกจากนี้ยังมีการเชื่อมโยงสายสัญญาณต่างๆระหว่างแผงอีกด้วย ทำให้เกิดการสับสนวุ่นวายเป็นอย่างมาก ดังนั้นการออกแบบแผงวงจรจึงควรรวมแผงวงจรต่างๆเข้ามาไว้ในแผงเดียวกัน และจัดให้มีจุดรวมสัญญาณและจุดส่งสัญญาณต่างๆเป็นกลุ่มๆอย่างมีระเบียบ

4. แผงวงจรบางส่วนถูกออกแบบตามทฤษฎีโดยที่ไม่ได้ทดลองก่อน ทำให้วงจรบางส่วนไม่สามารถทำงานได้ตามที่ออกแบบไว้ เนื่องจากปัจจัยอื่นๆที่ไม่ได้คาดคิดมาก่อน ดังนั้นแผงวงจรจึงควรได้รับการออกแบบหรือคำแนะนำจากผู้มีประสบการณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ธีรบูลย์ หล่อวิเชียรรุ่ง, นคร ภักดีชาติ, ชัยวัฒน์ ลิ้มพรจิตรวิไล, "ปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51 ด้วยโปรแกรมภาษา C", Innovative Experiment Co.,Ltd.

[2] ศ.ดร. วริทธิ์ อึ้งภากรณ์, รศ. ชาญ ถนัดงาน, "การออกแบบเครื่องจักรกล เล่ม 1-2", ซีเอ็ดยูเคชั่น, 2546

[3] Joseph E. Shigley, Charles R. Mischke, Richard G. Budynas, "Mechanical Engineering Design", Fifth Edition, McGrawhill.

[4] www.thaiio.com

[5] www.semiconductors.philips.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
ซอร์สโค้ดภาษาซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <REG51F.H>

sbit Up_Sw=P2^7;
sbit Do_Sw=P1^5;
sbit L_Sw=P1^7;
sbit R_Sw=P1^3;
sbit On_Sw=P2^4;
sbit Off_Sw=P2^5;

sbit G_LED=P2^3;
sbit R_LED=P2^1;
sbit M_Mode=P3^3;

sbit Mode1_Sw=P1^4;
sbit Mode2_Sw=P2^6;
sbit Mode3_Sw=P1^2;
sbit Mode4_Sw=P1^6;
sbit Mode5_Sw=P2^2;
sbit Mode6_Sw=P2^0;
sbit Mode7_Sw=P3^4;

sbit M1=P0^7;
sbit M2=P0^6;
sbit M3=P0^1;
sbit M3D=P0^0;
sbit M4=P0^3;
sbit M4D=P0^4;

sbit SWM1=P1^0;
sbit SWM2=P1^1;
sbit SWM3=P1^2;
sbit SWM4=P1^3;
sbit SWM5=P1^4;
sbit SWM6=P1^5;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sbit SWM7=P1^6;

sbit LIM_L=P3^5;
sbit LIM_R=P3^6;
sbit LIM_U=P1^0;
sbit LIM_D=P1^1;

bit M3ON;
bit M4ON;

char Mode;
int Trigger1,Trigger2,TrigCounter;
unsigned int AutoStartX,AutoStartY,Position;

#define Pos4cm 26500
#define Pos3cm 22000
void dusec(unsigned char time)
{
        while(time--);
}
void dmsec(unsigned int time)
{
        while(time--)
        {
                dusec(95);
        }
}
void InitTMR0(void)
{
        TMOD=0x22; ///   Timer 0 In 16 bit timer Mode
        TF0=0;
        TH0=0xB0;
        TL0=0xB0;
        IE=0x83;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        TR0=1;
        // INT0  P3.2
        IT0=1;
        EX0=1;

}

//// Detect AC Signal
void INT0_ISR(void) interrupt 0
{
        TrigCounter=0;
}
////////////////// Timer 0
void TMR0_INT(void) interrupt 1 // interrupt timer 1
{
        if(TrigCounter<1000)
        {
                if((TrigCounter==168)&&M3ON)  M3=0; else M3=1;
                TrigCounter++;
        }
}
void InitMode1(void)
{
        M1=0;
        M2=0;
        M3ON=1;
        M4=0;// // ON
        M4D=1;// Up
        Mode=1;
        AutoStartX=0;
        AutoStartY=0;
}
void InitMode2(void)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        M1=0;
        M2=0;
        M3ON=1;
        M4=0;// // ON
        M4D=0;// Down
        Mode=2;
        AutoStartX=0;
        AutoStartY=0;
}
void InitMode3(void)
{
        M1=0;
        M2=0;
        M3ON=1;
        M3D=0;
        M4=0;// // ON
        M4D=1;// Up
        Mode=3;
        AutoStartX=0;
        AutoStartY=0;
}
void InitMode4(void)
{
        M1=0;
        M2=0;
        M3ON=1;
        M3D=1;
        M4=0;// // ON
        M4D=0;// Down
        Mode=4;
        AutoStartX=0;
        AutoStartY=0;
}
void InitMode5(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
        M1=0;
        M2=0;
        M3ON=1;
        M4=0;// // ON
        M4D=1;// Up
        Mode=5;
        AutoStartX=0;
        AutoStartY=0;
}
void InitMode6(void)
{
        M1=0;
        M2=0;
        M3ON=1;
        M3D=1;
        M4=0;// // ON
        M4D=0;// Down
        Mode=6;
        Position=0;
        AutoStartX=0;
        AutoStartY=0;
}
void InitMode7(void)
{
        M1=0;
        M2=0;
        M3ON=1;
        M3D=1;
        M4=0;// // ON
        M4D=0;// Down
        Mode=7;
        Position=0;
        AutoStartX=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        AutoStartY=0;
}
void RunMode1(void)
{
        if(AutoStartX>0)
        {
                AutoStartX--;
                if(AutoStartX==1)
                        M3ON=1;
        }
        if((M4D==1) && (LIM_U==0))
        {       M4=1;   }

        if((M3D==1) && (LIM_L==0))
        {       M3ON=0;M3D=0; AutoStartX=4000;          }
        if((M3D==0) && (LIM_R==0))
        {       M3ON=0;M3D=1; AutoStartX=4000; }

}
void RunMode2(void)
{
        if(AutoStartX>0)
        {
                AutoStartX--;
                if(AutoStartX==1)
                        M3ON=1;
        }
        if((M4D==0) && (LIM_D==0))
        {       M4=1;   }

        if((M3D==1) && (LIM_L==0))
        {       M3ON=0;M3D=0; AutoStartX=4000;          }
        if((M3D==0) && (LIM_R==0))
        {       M3ON=0;M3D=1; AutoStartX=4000; }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
void RunMode3(void)
{
        if(AutoStartY>0)
        {
                AutoStartY--;
                if(AutoStartY==1)
                        M4=0;
        }
        if((M3D==0) && (LIM_R==0))
        {       M3ON=0;         }// M3 OFF

        if((M4D==1) && (LIM_U==0))
        {       M4=1;M4D=0; AutoStartY=4000;  }
        if((M4D==0) && (LIM_D==0))
        {       M4=1;M4D=1; AutoStartY=4000; }

}
void RunMode4(void)
{
        if(AutoStartY>0)
        {
                AutoStartY--;
                if(AutoStartY==1)
                        M4=0;
        }
        if((M3D==1) && (LIM_L==0))
        {       M3ON=0;         }// M3 OFF

        if((M4D==1) && (LIM_U==0))
        {       M4=1;M4D=0; AutoStartY=4000;  }
        if((M4D==0) && (LIM_D==0))
        {       M4=1;M4D=1; AutoStartY=4000; }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
void RunMode5(void)
{
        if(AutoStartY>0)// Y- Axis Hold Off
        {
                AutoStartY--;
                if(AutoStartY==1)
                        M4=0;
        }
        if(AutoStartX>0)// X- Axis Hold Off
        {
                AutoStartX--;
                if(AutoStartX==1)
                        M3ON=1;
        }

        if((M3D==1) && (LIM_L==0))
        {       M3ON=0;M3D=0; AutoStartX=4000;          }
        if((M3D==0) && (LIM_R==0))
        {       M3ON=0;M3D=1; AutoStartX=4000; }


        if((M4D==1) && (LIM_U==0))
        {       M4=1;M4D=0; AutoStartY=4000;  }
        if((M4D==0) && (LIM_D==0))
        {       M4=1;M4D=1; AutoStartY=4000; }
}
void RunMode6(void)
{
        if(AutoStartY>0)// Y- Axis Hold Off
        {
                AutoStartY--;
                if(AutoStartY==1)
                        M4=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
        if(AutoStartX>0)// X- Axis Hold Off
        {
                AutoStartX--;
                if(AutoStartX==1)
                        M3ON=1;
        }
        if(M3D==0)// Count Pos 4 cm
        {
                Position++;
        }

        if((M3D==1) && (LIM_L==0))
        {       M3ON=0;M3D=0; AutoStartX=2000;          Position=0;}
        if((M3D==0) && ((LIM_R==0)||(Position>Pos4cm)))
        {       M3ON=0;M3D=1; AutoStartX=2000; }

        if((M4D==1) && (LIM_U==0))
        {       M4=1;M4D=0; AutoStartY=4000;  }
        if((M4D==0) && (LIM_D==0))
        {       M4=1;M4D=1; AutoStartY=4000; }

}
void RunMode7(void)
{
        if(AutoStartY>0)// Y- Axis Hold Off
        {
                AutoStartY--;
                if(AutoStartY==1)
                        M4=0;
        }
        if(AutoStartX>0)// X- Axis Hold Off
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                AutoStartX--;
                if(AutoStartX==1)
                        M3ON=1;
        }
        if(M3D==0)// Count Pos 4 cm
        {
                Position++;
        }

        if((M3D==1) && (LIM_L==0))
        {       M3ON=0;M3D=0; AutoStartX=5000;      Position=0;}
        if((M3D==0) && ((LIM_R==0)||(Position>Pos3cm)))
        {       M3ON=0;M3D=1; AutoStartX=5000; }

        if((M4D==1) && (LIM_U==0))
        {       M4=1;M4D=0; AutoStartY=10000; }
        if((M4D==0) && (LIM_D==0))
        {       M4=1;M4D=1; AutoStartY=10000; }

}
void ClearLED(void)
{
        Mode1_Sw=1;
        Mode2_Sw=1;
        Mode3_Sw=1;
        Mode4_Sw=1;
        Mode5_Sw=1;
        Mode6_Sw=1;
        Mode7_Sw=1;
        M_Mode=1;
        On_Sw=1;
        M3ON=0;
        M4=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        dmsec(350);
}
void InitPosition(void)
{
        M4=0;
        M4D=1;
        while(1)
        {
                if((M4D==1) && (LIM_U==0))
                {       M4=1; break;    }
        }
}
void CheckSwitch(void)
{
        if((Mode1_Sw==0)&&(Mode!=1)&& (On_Sw==0)) {
        ClearLED();InitMode1();Mode1_Sw=0;On_Sw=0;    }
        if((Mode2_Sw==0)&&(Mode!=2)&& (On_Sw==0)) {
        ClearLED();InitMode2();Mode2_Sw=0;On_Sw=0;    }
        if((Mode3_Sw==0)&&(Mode!=3)&& (On_Sw==0)) {
        ClearLED();InitMode3();Mode3_Sw=0;On_Sw=0;    }
        if((Mode4_Sw==0)&&(Mode!=4)&& (On_Sw==0)) {
        ClearLED();InitMode4();Mode4_Sw=0;On_Sw=0;    }
        if((Mode5_Sw==0)&&(Mode!=5)&& (On_Sw==0)) {
        ClearLED();InitMode5();Mode5_Sw=0;On_Sw=0;    }
        if((Mode6_Sw==0)&&(Mode!=6)&& (On_Sw==0)) {
        ClearLED();InitMode6();Mode6_Sw=0;On_Sw=0;    }
        if((Mode7_Sw==0)&&(Mode!=7)&& (On_Sw==0)) {
        ClearLED();InitMode7();Mode7_Sw=0;On_Sw=0;    }
        if((M_Mode==0)&&(Mode!=8)) {  ClearLED();M1=1;M2=1;M_Mode=0; Mode=8;
On_Sw=1;        }// Manual Mode
        if((On_Sw==0)&&((Mode>7)||(Mode<1))) { M_Mode=1;On_Sw=0;Mode=0;
        G_LED=1;R_LED=1;}               // Nor
        if((Off_Sw==1)) {       ClearLED();G_LED=1;R_LED=1;M1=1;M2=1;Mode=0;      }
                // Ex
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
void RUN(void)
{
        while(1)
        {
                CheckSwitch();
                //  Manual Control X Y
                if(Mode==8)
                {
                        if((LIM_L==1) && (L_Sw==1))
                        {
                                G_LED=0;
                                R_LED=1;
                                M3ON=1;
                                M3D=1;
                                M1=0;
                                M2=0;
                        }
                        else if((LIM_R==1) && (R_Sw==1))
                        {
                                G_LED=0;
                                R_LED=1;
                                M3ON=1;
                                M3D=0;
                                M1=0;
                                M2=0;
                        }
                        else
                        {
                                M3ON=0;
                        }

                        if((LIM_U==1) && (Up_Sw==1))
                        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                G_LED=0;
                R_LED=1;
                M4=0;   // M4 ON
                M4D=1;
                M1=0;
                M2=0;
        }
        else if((LIM_D==1) && (Do_Sw==1))
        {
                G_LED=0;
                R_LED=1;
                M4=0;  // M4 ON
                M4D=0;
                M1=0;
                M2=0;
        }
        else
        {
                M4=1;
        }
//  LED Display
  if((LIM_L==0) && (L_Sw==1)){R_LED=0;G_LED=1;}
        else if((LIM_R==0) && (R_Sw==1)){R_LED=0;G_LED=1;}
        else if((LIM_U==0) && (Up_Sw==1)){R_LED=0;G_LED=1;}
        else if((LIM_D==0) && (Do_Sw==1)){R_LED=0;G_LED=1;}
        else if((M3ON==0)&&(M4==1)){R_LED=1;G_LED=1;M1=1;M2=1;}


}
switch(Mode)
{
        case 1:
                RunMode1();
        break;
        case 2:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    RunMode2();
                break;
                case 3:
                    RunMode3();
                break;
                case 4:
                    RunMode4();
                break;
                case 5:
                    RunMode5();
                break;
                case 6:
                    RunMode6();
                break;
                case 7:
                    RunMode7();
                break;
            }
        }
}
void main (void)
{
        unsigned char i,j;
        InitTMR0();
        i=0;
        j=0;
        dmsec(500);
        ClearLED();
        InitPosition();
        dmsec(1000);
        while(1)
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            RUN();
        }
    }
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้