

อุปกรณ์รับข้อมูลแบบถุงมือ

GLOVE BASED INPUT



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมู่.....
เลขทะเบียน...46164
วัน, เดือน, ปี 20 ธ.ค. 2546

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

h11286581

ปริญญานิพนธ์ปีการศึกษา 2544

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง อุปกรณ์รับข้อมูลแบบถุงมือ

GLOVE BASED INPUT

ผู้จัดทำ

- | | | |
|-----------------|-----------------|-----------------------|
| 1. นาย ชัยรัตน์ | ภักดีพานิชพงศ์ | รหัสประจำตัว 41014098 |
| 2. นาย วุฒ | ตั้งสิทธิ์ภักดี | รหัสประจำตัว 41014373 |



อุปกรณ์รับข้อมูลแบบถุงมือ

นาย ชัยรัตน์ ภักดีพานิชพงศ์ 41014098

นาย วุฒินันท์ ตั้งสิทธิ์ภักดี 41014373

รศ. สมศักดิ์ มิตะดา อาจารย์ที่ปรึกษา

อาจารย์ นเรศ มาลัย อาจารย์ที่ปรึกษา

ปีการศึกษา 2544

บทคัดย่อ

ในปัจจุบันนี้ การทำงานของอุปกรณ์ต่างๆ มักจะถูกควบคุมด้วยคอมพิวเตอร์เป็นส่วนใหญ่ ซึ่งวิธีการที่เราป้อนข้อมูลต่างๆ ลงคอมพิวเตอร์เพื่อให้คอมพิวเตอร์ทำกิจกรรมหนึ่งๆ นั้น บางกิจกรรมมีการรับค่าจากผู้ใช้อย่างไรก็ตาม แต่อุปกรณ์ที่ผู้ใช้สามารถป้อนให้คอมพิวเตอร์รับรู้ได้นั้นมีเพียงไม่กี่ชนิด ซึ่งอุปกรณ์ที่มีอยู่นั้นเป็นอุปกรณ์พื้นฐานที่รับข้อมูลจากผู้ใช้โดยรวมเสียมากกว่า โดยงานเฉพาะด้านนั้น อาจจำเป็นต้องมีอุปกรณ์รับข้อมูลเสริมขึ้นมา เพื่อให้ผู้ใช้สามารถใช้งานและควบคุมการทำกิจกรรมนั้นๆ ได้ดีและสะดวกยิ่งขึ้นกว่าเดิม

ในท้องตลาดเมืองไทยทุกวันนี้หากเราจะหาอุปกรณ์ที่มีไว้ใช้งานเฉพาะด้านจะหาซื้อได้ยาก นอกจากนั้นบางชนิดอาจจะต้องสั่งซื้อจากต่างประเทศ ซึ่งมีราคาแพง

โครงการนี้จึงต้องการสร้างอุปกรณ์รับข้อมูลแบบถุงมือ ที่รับค่าการเคลื่อนไหวของมือ ซึ่งประกอบไปด้วยส่วนหลักๆ คือข้อมือและนิ้วมือ และจะทำการสร้างวงจรที่จะทำการแปลงค่าที่ได้จากถุงมือนี้เพื่อที่จะส่งผลการเคลื่อนไหวของมือเราไปเข้าสู่คอมพิวเตอร์ และทำการแสดงผลที่ได้รับเป็นกราฟฟิกแบบ 3 มิติ

นอกจากนั้น โครงการนี้ยังสนับสนุนในการสร้างอุปกรณ์ที่สามารถสร้างเองได้ง่าย ไม่ซับซ้อน และมีราคาที่ย่อมเยากว่าอุปกรณ์ที่ซื้อมาจากต่างประเทศ โดยมีคุณภาพในระดับพอสมควรระดับหนึ่ง

Glove based Input

Chairat Bhakdibanijabongse

Varin Tangsitpakdee

Assoc. Prof. Somsak Mitatha

Advisor

Narath Marlai

Advisor

Academic Year 2001

ABSTRACT

Recently, we use computer to control most of accessories. Some activities need more methods to get the inputs but we have only a few type of input devices to communicate with computer to get the data that we want. Because of normal devices that we have only be a mass production but some application or activities need some much powerful input device for compatible and convenient.

In Thai market place, It's so hard to find some devices to use for a special job,. And some devices we must find from aboard market ,that so expensive.

This project wants to make glove based input device, to detect hand movement that compose of wrist and fingers, and make adapter circuit to change analog signals to digital signal and shows the results in 3D graphics on monitor.

Moreover, this project still support an easily way to make, not so complexity and cheaper than buying from abroad in properly quality.

สารบัญ

หน้าที่

บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
สารบัญ	III
สารบัญตาราง	VII
สารบัญภาพ	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ขอบเขตของโครงการ	2
1.4 ขั้นตอนการทำงาน	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ	4
บทที่ 2 ธรรมชาติของมือ	5
3.1 การเคลื่อนไหวของข้อมือ	5
3.1.1 ฟเลกชัน (Flexion) / เอ็กซเทนชัน (Extension)	5
3.1.2 อัลนา ดีฟิเอนชัน / เรเดียล ดีฟิเอนชัน	6
3.2 การวัดการเคลื่อนไหวของนิ้วมือและข้อนิ้วมือ	6
3.2.1 การเคลื่อนไหวของข้อ คาร์โปเมตาแคปัล อันที่ 1	7
3.2.2 การเคลื่อนไหวของ ข้อต่อ เมตาคาร์โปฟาแลงกิว อันที่ 1	9
3.2.3 การเคลื่อนไหวของ ข้อต่อ อินเตอร์ฟาแลงกิว ของนิ้วหัวแม่มือ	9
3.2.4 การเคลื่อนไหวของ ข้อต่อ เมตาคาร์โปฟาแลงกิว	9
3.2.5 การเคลื่อนไหวของ ข้อต่อ พร็อกซิมอล อินเตอร์ฟาแลงกิว (เวนนิ้วหัวแม่มือ)	10
3.2.6 การเคลื่อนไหวของ ข้อต่อดิสทอล อินเตอร์ฟาแลงกิว (เวนนิ้วหัวแม่มือ)	10

สารบัญ

หน้าที่

บทที่ 3 ไคเร็กซ์เอ็กซ์	11
3.1 รู้จักกับไคเร็กซ์เอ็กซ์	11
3.2 รู้จักกับคอม (COM)	11
3.3 ส่วนประกอบของไคเร็กซ์เอ็กซ์	12
3.3.1 ไคเร็กซ์ดรอ	12
3.3.2 ไคเร็กซ์สามดี	12
3.3.3 ไคเร็กซ์ชวาร์	12
3.3.4 ไคเร็กซ์เพลย์	12
3.3.5 ไคเร็กซ์อินพุต	12
3.3.6 ไคเร็กซ์มิวสิก	12
3.3.7 ไคเร็กซ์เซตอัป	12
3.3.8 ออโต้เพลย์	12
3.4 สิ่งที่ต้องกำหนดในไมโครซอฟท์วิชวลเบสิก	12
3.5 ผู้โลกสามมิติ	13
3.6 แสดงผลลัพธ์ของไคเร็กซ์เอ็กซ์ 6	15
บทที่ 4 การออกแบบในส่วนของฮาร์แวร์	17
4.1 การออกแบบวงจรโดยรวม	17
4.1.1 เซนเซอร์	17
4.1.2 วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล	17
4.1.3 ไมโครคอนโทรลเลอร์	18
4.1.4.4 วงจรแปลงสัญญาณ	18
4.1.5 คอมพิวเตอร์	18
4.1.6 การเขียนโปรแกรมกราฟฟิอินเตอร์เฟส	18
4.2 การออกแบบในส่วนของฮาร์แวร์	19
4.2.1 การออกแบบฮาร์แวร์ในส่วนของคุณมือ	19
4.2.1.1 แนวคิดในการออกแบบคุณมือ	19
4.2.1.2 การออกแบบตัวคุณมือ	20
4.2.2 การออกแบบฮาร์แวร์ในส่วนของคุณวงจร	23
4.2.2.1 แนวคิดของการออกแบบวงจร	23

สารบัญ

หน้าที่

4.2.2.2 อุปกรณ์หลักที่ใช้ในวงจร	24
4.2.2.3 เหตุผลในการเลือกใช้อุปกรณ์ต่างๆ	24
4.2.2.4 รายละเอียดการต่อวงจร	25
4.2.2.4.1 การต่อ ADC0817 เข้ากับเซนเซอร์(ความต้านทานปรับค่าได้)	25
4.2.2.4.2 การใช้งาน ADC0817	26
4.2.2.4.3 การใช้งาน MAX232	26
4.2.2.4.4 วงจรโดยรวมและการทำงานของวงจรโดยรวม	27
บทที่ 5 การออกแบบในส่วนของซอฟต์แวร์	
5.1 โปรแกรมควบคุม 89c51	31
5.1.1 โครงสร้างการทำงานหลักๆ ของโปรแกรมที่จะทำการบันทึกลง ไอซี 89C51	31
5.1.2 ซอสโค้ดของโปรแกรมตัวส่งข้อมูลบนไอซี 89C51	32
5.2 โปรแกรมรับค่าจาก 89C51 และแสดงผลเป็นเท็กซ์โหมดบนคอมพิวเตอร์	34
5.2.1 การทดสอบการรับค่าจาก 89c51 บนคอมพิวเตอร์	34
5.2.2 โปรแกรมการรับข้อมูลบนคอมพิวเตอร์	34
5.2.2.1 อินเตอร์เฟซบนวีลเบลิก	31
5.2.2.2 ซอสโค้ดของโปรแกรมบนวีลเบลิก	35
5.3 ส่วนการแสดงผลเป็นกราฟฟิกโหมดบนคอมพิวเตอร์	37
5.3.1 ไฟล์มาตรฐานอุตสาหกรรมเมซ (.3ds)	39
5.3.2 การทำงานของโปรแกรม	39
5.3.3 ซอสโค้ดของโปรแกรมแสดงผลโดเร็กซ์สามดี	41
บทที่ 6 ผลการทดลอง	53
6.1 การทดลองวิธีที่ 1	53
6.1.1 ผลการทดลองวิธีที่ 1	53
6.1.2 สรุปผลการทดลองวิธีที่ 1	54
6.2 การทดลองวิธีที่ 2	54
6.2.1 ผลการทดลองวิธีที่ 2	56
6.2.2 สรุปผลการทดลองวิธีที่ 2	58

สารบัญ

หน้าที่

บทที่ 7 บทวิจารณ์และสรุป	59
7.1 สรุปผลการทำงาน	59
7.2 ปัญหาและอุปสรรคในการทำงาน	59
7.2.1 ส่วนงานทางด้านฮาร์ดแวร์	59
7.2.1.1 ส่วนของวงจร	59
7.2.1.2 ส่วนของตัวถูงมือ	59
7.2.2 ส่วนงานทางด้านซอฟต์แวร์	59
7.3 แนวทางการพัฒนาต่อไป	60
ภาคผนวก ก.	61
ภาคผนวก ข.	65
บรรณานุกรม	73



สารบัญตาราง

หน้าที่

ตารางที่ 3.1	คลาสหลักที่จะใช้ในการทำโปรแกรม	15
ตารางที่ 6.1	แสดงค่าที่เซนเซอร์เปลี่ยนแปลงในแต่ละข้อ และองศาที่เคลื่อนไหวจริงจากค่าเฉลี่ยทำการทดลอง 100 ครั้ง	57



สารบัญภาพ

	หน้าที่
รูปที่ 1.1 ขอบเขตของโครงการ	2
รูปที่ 2.1 การเคลื่อนไหวของข้อมือ	6
รูปที่ 2.2 การเคลื่อนไหวของข้อ คาร์โปเมตาแคแพล อันที่ 1 : แอ็บดักชัน	7
รูปที่ 2.3 การเคลื่อนไหวของข้อ คาร์โปเมตาแคแพล อันที่ 1 : แพลเมอร์ฟเลกชัน	8
รูปที่ 2.4 การเคลื่อนไหวของข้อ คาร์โปเมตาแคแพล อันที่ 1 : เซอร์คัมดักชัน	8
รูปที่ 2.5 ออปโพสิชัน ของนิ้วหัวแม่มือ	9
รูปที่ 2.6 การเคลื่อนไหวของข้อนิ้วมือ	10
รูปที่ 3.1 ตัวอย่างชินกราฟ	14
รูปที่ 4.1 บล็อกไดอะแกรมโดยรวม	17
รูปที่ 4.2 บล็อกไดอะแกรมที่นำไปออกแบบ	18
รูปที่ 4.3 ตำแหน่งการวางเซนเซอร์บนมือ	19
รูปที่ 4.4 การออกแบบในส่วนการเคลื่อนไหวในแนวฟเลกชัน / เอ็กซเทนชัน	20
รูปที่ 4.5 การออกแบบในส่วนการเคลื่อนไหวในแนวแอ็บดักชัน / แอดดักชัน	21
รูปที่ 4.6 การเคลื่อนไหวของนิ้วโป้งในแนวโรเตชัน	22
รูปที่ 4.7 รูปแบบถุงมือที่ออกแบบ	22
รูปที่ 4.8 Ratiometric Conversion System	25
รูปที่ 4.9 การใช้งาน ADC0817	26
รูปที่ 4.10 การต่อ MAX232 ในการใช้งานจริง	27
รูปที่ 4.11 สคีมติกของวงจรโดยรวม	29
รูปที่ 4.12 วงจรแหล่งจ่ายไฟ	30
รูปที่ 5.1 อินเตอร์เฟสหน้าจอโปรแกรมการรับค่าจากไอซี 89C51	34
รูปที่ 5.2 ชินกราฟของโปรแกรมในมุมมองแบบทรีวิว	38

สารบัญภาพ

	หน้าที่
รูปที่ 5.3 ภาพไฟล์มาตรฐานอุตสาหกรรมเมซที่เราใช้ในโรงงานนี้	39
รูปที่ 5.4 ภาพของฟอร์มของโปรแกรมบนวิซวลเบสิก	40
รูปที่ 5.5 ภาพการใช้วิซวลเบสิกโหลดวัตถุสามมิติจากไฟล์เมซไคเรกซ์เอ็กซ์	40
รูปที่ 5.6 ภาพผลลัพธ์ของการผนวกอุปกรณ์อินพุตแบบดึงมือกับไคเรกซ์เอ็กซ์	41
รูปที่ 6.1 แสดงค่าองศาที่วัดได้ในแต่ละข้อนิ้ว	53
รูปที่ 6.2 การทดลองขั้นที่ 1	54
รูปที่ 6.3 การทดลองขั้นที่ 2	55
รูปที่ 6.4 การทดลองขั้นที่ 4	55
รูปที่ 6.5 การทดลองขั้นที่ 5	56
รูปที่ 6.6 การทดลองขั้นที่ 6	56
รูปที่ ก.1 ภาพมือด้านบน	61
รูปที่ ก.2 ข้อมือและมือ	62
รูปที่ ก.3 ภาพมือด้านบน	63
รูปที่ ก.4 ข้อมือและมือ	64
รูปที่ ข.1 สคิเมติกของแผ่นปรินต์แผ่นที่ 1	65
รูปที่ ข.2 Bottom Layer	66
รูปที่ ข.3 Top Layer	66
รูปที่ ข.4 Bottom Silkscreen Overlay	66
รูปที่ ข.5 Top Silkscreen Overlay	67
รูปที่ ข.6 Top Solder Mask Print	67
รูปที่ ข.7 Bottom Solder Mask Print	67
รูปที่ ข.8 Top Pad Master	68
รูปที่ ข.9 Bottom Pad Master	68
รูปที่ ข.10 สคิเมติกของแผ่นปรินต์แผ่นที่ 2	69
รูปที่ ข.11 Bottom Layer	70
รูปที่ ข.12 Top Layer	70
รูปที่ ข.13 Top Silkscreen Overlay	70

สารบัญภาพ

หน้าที่

รูปที่ ข.14 Top Solder Mask Print	71
รูปที่ ข.15 Bottom Solder Mask Print	71
รูปที่ ข.16 Top Pad Master	71
รูปที่ ข.17 Bottom Pad Master	72



บทที่ 1

บทนำ

1.1 ความเป็นมาของโครงการ

เนื่องจากอุปกรณ์ในการควบคุมการใช้งานต่างๆ ซึ่งในปัจจุบัน มีไม่กี่ชนิดเท่านั้นที่แพร่หลาย เช่น เมาส์ คีย์บอร์ด จอยสติ๊ก ฯลฯ ซึ่งการควบคุมเหล่านี้เป็นอุปกรณ์พื้นฐานสำหรับการควบคุมอุปกรณ์หรือคำสั่งต่างๆ ผ่านคอมพิวเตอร์ ซึ่งคอมพิวเตอร์ก็ได้สนับสนุนการใช้งานอุปกรณ์เหล่านี้ได้คืออยู่แล้ว แต่ว่าการควบคุมอุปกรณ์เหล่านั้นใช้มือในการควบคุมอุปกรณ์ต่าง ๆ เพื่อให้อุปกรณ์เหล่านั้นติดต่อกับคอมพิวเตอร์หนึ่ง ซึ่งในกิจกรรมบางกิจกรรมโดยเฉพาะอย่างยิ่งการติดต่อกับกราฟิก 3 มิติ เช่น การใช้โปรแกรม 3D Studio MAX การวาดรูปภาพ รวมไปถึงการควบคุมอุปกรณ์ภายนอกผ่านทางพอร์ตของคอมพิวเตอร์เช่น แขนกลนั้น หากเราใช้เมาส์หรือคีย์บอร์ดในการควบคุมอุปกรณ์เหล่านี้ เราจะเสียเวลาและต้องใช้ความพยายามอย่างสูงในการควบคุมอุปกรณ์เหล่านั้นให้ทำงานดังที่เราต้องการได้

ซึ่งเหตุผลที่มีการติดต่อแบบสามมิติคือ

1. งานบางอย่างโดยธรรมชาติแล้วเป็นสามมิติ ดังนั้นแก้ปัญหาด้วยการติดต่อแบบสองมิติจึงมีความยุ่งยากและซับซ้อนมากกว่าการติดต่อแบบสามมิติมาก
2. มนุษย์เป็นสามมิติดังนั้นจึงได้มีการพัฒนาการใช้มือและความรู้ออกมาในรูปแบบสามมิติทั้งสิ้น
3. เป็นช่องทางในการสื่อสารพิเศษ การนำเสนอข้อมูลแบบสามมิติทำให้มีความเข้าใจได้มากขึ้นและรวดเร็วขึ้นด้วย
4. เป็นการรวมการทำงานเข้ากับสิ่งแวดล้อมได้ โดยที่เราไม่ต้องนั่งทำงานอยู่ที่โต๊ะโดยใช้คีย์บอร์ดอีกต่อไป

เราจึงต้องการสร้างอุปกรณ์สำหรับใช้ควบคุมการทำงานต่างๆ โดยอาศัยการเคลื่อนไหวของมือคนเราในการควบคุมกิจกรรมต่างๆ ที่เราต้องการ เช่น การทำภาพเคลื่อนไหว 3 มิติของมนุษย์ ซึ่งจะนำมาใช้ในการทำภาพยนตร์หรือเกมได้ ซึ่งเป็นการลดค่าใช้จ่ายจากการซื้อผลิตภัณฑ์ชนิดนี้จากต่างประเทศ ซึ่งผลที่ได้จากการทำเองนี้ อาจจะไม่ได้ผลการวัดค่าที่ละเอียดมากเท่าของต่างประเทศ แต่เมื่อนำมาเทียบกับราคาที่ใช้ในการสร้างกับผลลัพธ์ที่ได้ถือว่าคุ้มที่จะทำขึ้นมา ซึ่งอุปกรณ์ที่เราใช้ในการสร้างนั้นหาซื้อได้ง่ายและราคาถูก อีกทั้งโครงสร้างของวงจรไม่ซับซ้อนมากนัก เพื่อสามารถประดิษฐ์ และสามารถพัฒนาต่อไปได้

1.2 วัตถุประสงค์ของโครงการ

- สามารถติดต่อกับคอมพิวเตอร์และแสดงผลในคอมพิวเตอร์ในรูปแบบกราฟฟิก 3 มิติได้
- มีความทนทานพอสมควร
- สามารถขนย้ายได้ไม่ยากนัก
- ใช้งานได้ง่าย
- ใช้อุปกรณ์ที่หาได้ในท้องถิ่น
- มีราคาไม่แพงมากนัก
- สามารถทำงานได้จริง
- สามารถพัฒนาต่อได้ เพื่อนำไปประยุกต์ใช้งานด้านต่างๆ ได้

1.3 ขอบเขตของโครงการ

1. สร้างอุปกรณ์ในการตรวจจับการเคลื่อนไหวของมือ ซึ่งประกอบด้วยส่วนนิ้วมือ และส่วนข้อมือ โดยใช้เซนเซอร์ (ความต้านทานปรับค่าได้) ในการตรวจจับการเคลื่อนไหวของมือ
2. ออกแบบและสร้างวงจรที่ใช้แปลงสัญญาณอนาล็อกจากเซนเซอร์ (ที่ใช้ตรวจจับการเคลื่อนไหวของมือ) แล้วนำไปแปลงเป็นสัญญาณดิจิทัล แล้วนำผลข้อมูลที่ได้ไปเข้าสู่ไมโครคอนโทรลเลอร์เพื่อทำการจัดการรับข้อมูลที่ละจุดจากเซนเซอร์และนำมาจัดรูปแบบเพื่อที่จะนำไปส่งเข้าสู่คอมพิวเตอร์ผ่านพอร์ตอนุกรม
3. สร้างโปรแกรมที่รับค่าจากวงจรที่เราสร้างขึ้นมานำผลที่ได้รับจากวงจรไปแสดงผลลัพท์ออกมาในรูปแบบของกราฟฟิก 3 มิติ เพื่อแสดงผลการเคลื่อนไหวของมือ (นิ้วมือและข้อมือ)



รูปที่ 1.1 ขอบเขตของโครงการ

1.4 ขั้นตอนการทำงาน

แบ่งการทำงานของโครงงานนี้ออกเป็น 3 ขั้นตอนหลักๆ ดังนี้

ขั้นตอนที่ 1 ออกแบบและสร้างวงจรฮาร์ดแวร์

เราจะต้องคิดหาวิธีในการเลือกใช้เซนเซอร์ที่เหมาะสม (ความต้านทานปรับค่าได้) และออกแบบวงจรในการเปลี่ยนค่าที่ได้รับจากเซนเซอร์นั้นๆ เปลี่ยนไปเป็นสัญญาณดิจิทัลแล้วนำผลลัพธ์ที่ได้ส่งเข้าสู่คอมพิวเตอร์อีกทีหนึ่ง ซึ่ง ณ จุดนี้เราแบ่งการทำงานออกเป็นส่วนย่อยๆ ได้ดังนี้

- ศึกษาการเคลื่อนไหวของมือมนุษย์ รวมไปถึงข้อต่อต่างๆ เพื่อที่เราจะได้รับทราบถึงแนวการเคลื่อนไหวของมือ ข้อนิ้ว ข้อมือต่างๆ ว่าสามารถยืดหด หรือเคลื่อนไหวได้ที่แกน รวมถึงขอบเขตองศาการเคลื่อนไหวในแต่ละแกนอีกด้วย
- ศึกษาเซนเซอร์ชนิดต่างๆ ที่สามารถจะนำมาใช้ในการประยุกต์ใช้เพื่อติดเข้ากับมือได้ ทำให้เราสามารถวาดภาพการประกอบเซนเซอร์ชนิดนั้นๆ เป็นถুমมือได้ นอกจากนั้นเรายังต้องคำนึงถึงประสิทธิภาพ ราคา แหล่งที่จะซื้อ การประกอบเข้ากับมืออีกด้วย เพื่อให้ได้เซนเซอร์ที่เหมาะสมมือมากที่สุด
- ศึกษาวิธีการแปลงค่าสัญญาณอนาล็อกจากเซนเซอร์เพื่อแปลงไปเป็นสัญญาณดิจิทัล เพื่อที่จะได้ส่งค่าไปประมวลผลต่างๆ ได้
- ศึกษาวิธีการส่งค่าสัญญาณดิจิทัลผ่านพอร์ตอนุกรม เพราะว่าค่าที่ใช้กันในวงจรต่างๆ ไม่ว่าจะเป็นไมโครโปรเซสเซอร์หรือไมโครคอนโทรลเลอร์นั้น ต่างก็ใช้ชิตเต็มบัสในหลัก 2^n บิต เมื่อ $n > 2$ (8, 16, 32, 64 บิต เป็นต้น) แต่พอร์ตอนุกรมนั้น อนุญาตให้ส่งข้อมูลได้ที่ละบิตเท่านั้น เราจึงต้องศึกษาวิธีการส่งและโปรโตคอลที่ใช้ส่งและรับข้อมูลด้วย
- ออกแบบวงจรรวมตั้งแต่ส่วนที่รับข้อมูลเข้ามาจากเซนเซอร์ผ่านวงจรแปลงสัญญาณจากอนาล็อกเป็นสัญญาณดิจิทัล วงจรจัดรูปแบบข้อมูลที่ได้รับ วงจรแปลงข้อมูลจากข้อมูล 8 บิตเป็นข้อมูลบิตเดียวเรียงกันไปเพื่อส่งเข้าคอมพิวเตอร์ผ่านพอร์ตอนุกรม
- ทดสอบต่อวงจรแต่ละส่วนเพื่อทดสอบเป็นโมดูลย่อยๆ ทีละส่วน
- ทดสอบต่อวงจร โมดูลรวมทั้งหมด และเขียน โปรแกรมทดสอบรับข้อมูลอย่างง่าย ว่าได้ค่าตามที่ต้องการหรือไม่

ขั้นตอนที่ 2 ออกแบบและสร้างตัวตรวจจับการเคลื่อนไหวของมือ

เมื่อเราได้เซนเซอร์ที่จะใช้กับวงจรที่ทดสอบแล้วใช้งานได้ ทีนี้เราก็มาออกแบบรูปแบบของอุปกรณ์ที่จะใช้ตรวจจับการเคลื่อนไหวของมือว่าจะวางตำแหน่งไหน ยังไง เพื่อที่เราจะได้สร้างอุปกรณ์ชิ้นนี้ให้เป็นรูปร่างและใช้ส่งข้อมูลได้ โดยมีรายละเอียดดังนี้

- ศึกษาและเลือกวัสดุต่างๆ ที่จะนำเซนเซอร์มาประกอบเข้าเป็นรูปร่าง
- ทดสอบต่อวัสดุทีละส่วนย่อยๆ เช่น ทีละข้อนิ้ว ทีละนิ้วมือ

- ทดสอบต่อวัสดุโดยรวม ประกอบวัสดุรวมทั้งหมดเข้าด้วยกัน และต่อเข้ากับวงจรที่เราได้สร้างไว้แล้ว

ขั้นตอนที่ 3 เขียนโปรแกรมแสดงผลค่าที่รับมาจากวงจรในรูปแบบกราฟฟิค 3 มิติ

เมื่อเราได้วงจรรวมกับอุปกรณ์ตรวจจับการเคลื่อนไหวของมือแล้ว ทีนี้ก็เหลือส่วนแสดงผลซึ่งเราจะแสดงผลในรูปแบบกราฟฟิค 3 มิติ เพื่อให้เราดูผลลัพธ์ที่ได้รับจากอุปกรณ์มาแสดงผลในรูปแบบที่เราเข้าใจได้ง่าย เพื่อทดสอบและพิสูจน์ว่าผลข้อมูลที่ได้รับนั้น ตรงกับที่ได้รับจากตัวอุปกรณ์นั้นจริงๆ โดยมีรายละเอียดดังนี้

- ศึกษาเอ็นจินต์ต่างๆ ว่าตัวไหนเหมาะสมมาใช้งาน
- ศึกษาวิธีการเขียนและใช้เอ็นจินต์นั้น
- สร้างโมเดล 3 มิติเพื่อใช้ในการแสดงผล
- เขียนโปรแกรมทดสอบเอ็นจินต์เพื่อควบคุมการเคลื่อนไหวของโมเดล
- เขียนโปรแกรมเพื่อนำมาทดสอบวงจรที่เราได้สร้างไว้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

สามารถสร้างอุปกรณ์ตรวจจับการเคลื่อนไหวตั้งแต่แบบง่ายๆ จนพัฒนาได้เป็นอุปกรณ์ตรวจจับการเคลื่อนไหวที่สามารถใช้งานได้สมบูรณ์และถูกต้องซึ่งจะสามารถนำมาประยุกต์ใช้งานได้หลายด้านต่างๆ ซึ่งส่วนใหญ่เป็นการใช้ใน VE (Visual Environment) ในด้านต่าง ๆ ดังนี้

- ด้านพาณิชย์ : ถ้าเราสามารถลดราคาค่าต้นทุนและสร้างไคร์เวอร์ให้ใช้งานได้ และหาซอฟต์แวร์ที่เหมาะสมมาใช้งานให้กับผู้ใช้ได้เยอะๆ ก็สามารถนำมาประยุกต์ในการค้าขายได้
- ด้านสื่อภาษา : การแปลงการเคลื่อนไหวของมือเป็นคำพูด ซึ่งเหมาะสำหรับคนใบ้จะใช้ในการสื่อสารกับบุคคลธรรมดาที่ไม่รู้ภาษามือได้
- ด้านวิทยาศาสตร์ : การใช้กับงานที่เสี่ยงต่อความปลอดภัยของมนุษย์ เช่น กู้ระเบิด โดยต่อเข้ากับแขนกล หรือการทดลองวิจัยสารเคมีอันตรายเป็นต้น
- ด้านบันเทิง : ใช้การเคลื่อนไหวของมือและนิ้วมือในการสร้างเสียงดนตรีเช่น เป่าขลุ่ย หรือการโยกย้ายเป็นจังหวะเสียงเพลงในทำนองต่างๆ
- ด้านการแพทย์ : ใช้ในการผ่าตัดและวินิจฉัยโรคผ่านเครือข่าย (telemedicine), การผ่าตัดโดยใช้หุ่นยนต์
- ด้านการศึกษา : ใช้เป็นสื่อการสอนในการเรียนรู้ในสภาวะเสมือนจริง

บทที่ 2

ธรรมชาติของมือ

มือเป็นอวัยวะส่วนหนึ่งของร่างกายที่มีการเคลื่อนไหวอยู่ตลอดเวลา โดยมนุษย์เราใช้มือในการช่วยทำกิจกรรมต่างๆ ในชีวิตประจำวัน ไม่ว่าจะเป็นการกิน การหยิบจับสิ่งของ การนับ การคำนวณ รวมถึงการสื่อสารต่างๆ ด้วย ดังนั้นหากเราต้องการจะสร้างอุปกรณ์ที่ใช้ตรวจจับการเคลื่อนไหวโดยใช้มือเป็นตัวควบคุมนั้น เราต้องเข้าใจถึงสรีระของมือ เพื่อที่จะได้ออกแบบและสร้างอุปกรณ์ที่เหมาะสมกับมือได้

สเตอแมน และ เซทเซอร์(1993) (Sturman and Zeltzer(1993)) ได้ให้ความหมายของ การรับข้อมูลจากทั้งมือ (whole-hand inputs) ว่า“เป็นการใช้ความสามารถของมืออย่างเต็มที่และโดยตรงสำหรับการควบคุมงานที่มีคอมพิวเตอร์เป็นสื่อ...[การจัดหาการเชื่อมต่อที่] ทำให้ประโยชน์จาก ความเป็นธรรมชาติ, ความชำนาญ, และ การดัดแปลงได้ง่าย ของมือนมนุษย์ ได้อย่างเต็มที่” (“ the full and direct use of the hand’s capabilities for the control of computer- mediated tasks... [providing an interface that] makes maximal use of the naturalness, dexterity, and adaptability of the human hand.”)

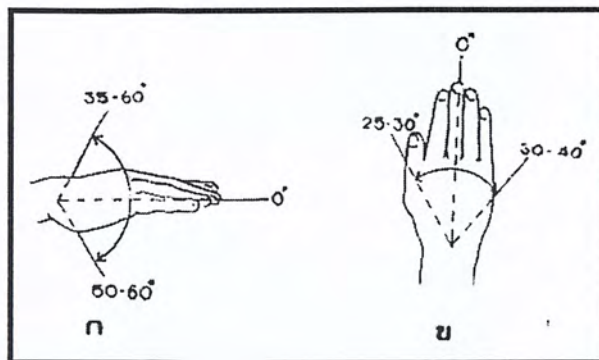
หมายเหตุ : ศัพท์ภาษาอังกฤษทั้งหมดในบทนี้ ส่วนใหญ่เป็นศัพท์ทางการแพทย์ สามารถดูรูปอธิบายได้ที่ภาคผนวก ก.

การเคลื่อนไหวของอวัยวะส่วนต่างๆ

2.1 การเคลื่อนไหวของข้อมือ

ท่าเหยียดข้อมือตรงเป็นท่า 0 องศาของข้อมือ คือแกนตามยาวของกระดูกเมตาแคปัล (Metacarpal) กับของกระดูก อัลนา ขนานกัน การเคลื่อนไหวของข้อมือมี 2 แบบ ดังนี้

2.1.1 ฟเลกชัน (Flexion) / เอ็กซ์เทนชัน (Extension) (รูปที่ 2.1 ก) คือเคลื่อนไหวในแนวแซกทิทัล (Sagittal) การเคลื่อนไหวไปด้านหลังมือเป็น เอ็กซ์เทนชัน การเคลื่อนไหวไปทางฝ่ามือเป็น ฟเลกชัน มุมการเคลื่อนไหวใช้มุมระหว่างแกนทางด้านข้างของกระดูก



รูปที่ 2.1 การเคลื่อนไหวของข้อมือ

(ก) การเคลื่อนไหว ฟเลกชัน / เอ็กซ์เทนชัน

(ข) การเคลื่อนไหว อัลนา ดีฟิเอชัน (Ulnar Deviation) / เรเดียล ดีฟิเอชัน (Radial Deviation)

เมตาคารปัล ทั้ง 5 อัน (ซึ่งเป็นแกนร่วม) กับแกนตามยาวของกระดูก อัลนา

ค่าปรกติ ฟเลกชัน / เอ็กซ์เทนชัน S $60^{\circ} / 0^{\circ} / 5^{\circ}$

2.1.2 อัลนา ดีฟิเอชัน / เรเดียล ดีฟิเอชัน (รูปที่ 2.1 ข) เป็นการเคลื่อนไหวในระนาบฟรอนทัล (Frontal Plane) ทำ 0 องศา คล้ายกับข้อ (2.1.1) แต่ใช้แกนตามยาวของแขนท่อนปลายแทนที่จะใช้แกนตามยาวของกระดูก อัลนา อันเดียว และมุมการเคลื่อนไหวใช้มุมระหว่างแกนตามยาวของกระดูก เมตาคารปัล อันที่ 3 กับแกนตามยาวของแขนท่อนปลาย

ค่าปรกติ อัลนา ดีฟิเอชัน / เรเดียล ดีฟิเอชัน F $30^{\circ} / 0^{\circ} / 20^{\circ}$

2.2 การวัดการเคลื่อนไหวของนิ้วมือและข้อมือนิ้วมือ

ท่า 0 องศาของนิ้วคือท่าที่นิ้วมือเหยียดตรง ควรบันทึกข้อมือนิ้วมือน้อยกว่าการใช้หมายเลข อย่างไรก็ตามถ้าใช้หมายเลข นิ้วหัวแม่มือควรเป็นนิ้วที่ 1 นิ้วชี้เป็นนิ้วที่ 2 และต่อไปตามลำดับ และใช้อักษรย่อสำหรับข้อต่อต่างๆ ดังนี้

IP	=	ข้อต่อ อินเตอร์ฟาแลงกีว (Interphalangeal Joint)
DTP	=	ข้อต่อ ดิสทอล อินเตอร์ฟาแลงกีว (Distal Interphalangeal Joint)
PIP	=	ข้อต่อ พร็อกซิมอล อินเตอร์ฟาแลงกีว (Proximal Interphalangeal Joint)
MP	=	ข้อต่อ เมตาคาร์โปฟาแลงกีว (Metacarpophalangeal Joint)
CM	=	ข้อต่อ คาร์โปเมตาคารปัล (Carpometacarpal Joint)

ข้อมือนิ้วมือโดยมากจะเคลื่อนไหวได้คล้ายๆ กัน คือ ฟเลกชัน / เอ็กซ์เทนชัน และเป็นการเคลื่อนไหวรวมของ MP และ IP การวัด ฟเลกชัน / เอ็กซ์เทนชัน ของแต่ละข้อมือนิ้วมือใช้ โกนิโอมิเตอร์ (Goniometer) ขนาดเล็กๆ พอดีกับนิ้วมือ โดยบันทึกเป็นการเคลื่อนไหวของข้อต่างๆ

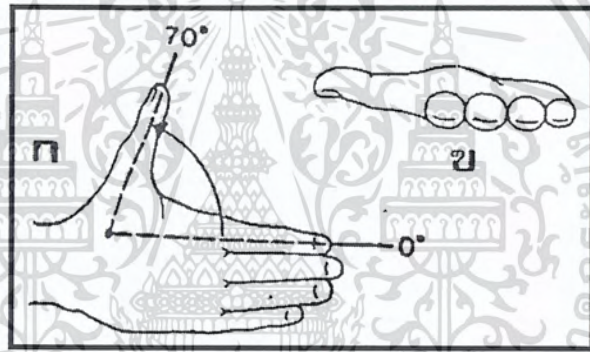
ข้อ MP มักจะมี โอเวอร์เอ็กซ์เทนชัน เล็กน้อย แต่ PIP และ DIP มักไม่ค่อยพบเช่นนั้น

แอ็บดักชัน (Abduction) / แอ็ดดักชัน (Adduction) ของนิ้ว การวัดอาจวัดระยะห่างระหว่างปลายนิ้วทั้งสองแทนที่จะบันทึกการเคลื่อนไหวเป็นองศาก็ได้

สำหรับนิ้วหัวแม่มือการเคลื่อนไหวที่สำคัญอยู่ที่ข้อ CM และเมื่อรวมข้อ MP และ IP ด้วยแล้ว จะทำให้การเคลื่อนไหวของนิ้วหัวแม่มือเป็นไปได้หลายแบบ ยุ่งยากขึ้นเล็กน้อย ได้แก่

2.2.1 การเคลื่อนไหวของข้อ คาร์โปเมตาแคปัล แอนที 1 มีดังนี้

ก. แอ็บดักชัน / แอ็ดดักชัน (รูปที่ 2.2) เป็นการเคลื่อนไหวในระนาบพรีแนล ทำ 0 องศา คือทำนิ้วหัวแม่มือเหยียดตรงและชิดกับนิ้วชี้ การเคลื่อนไหวของนิ้วหัวแม่มือจากท่านี้อือ ถ้ากางนิ้วหัวแม่มือออกจากนิ้วชี้ก็เป็น แอ็บดักชัน ส่วนการหุบนิ้วหัวแม่มือเข้า (แอ็ดดักชัน) จะมีค่าเป็น 0° มุมการเคลื่อนไหวคือมุมระหว่างแกนตามยาวของนิ้วชี้กับแกนตามยาวของนิ้วหัวแม่มือที่เปลี่ยนตำแหน่งไป



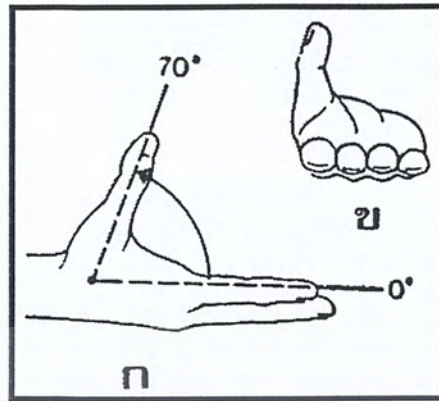
รูปที่ 2.2 การเคลื่อนไหวของข้อ คาร์โปเมตาแคปัล แอนที 1 : แอ็บดักชัน

(ก) มองทางด้านหลังมือ

(ข) มองทางด้านปลายนิ้วมือ

ค่าปรกติ แอ็บดักชัน / แอ็ดดักชัน F $70^{\circ} / 0^{\circ} / 0^{\circ}$

ข. แพลเมอร์ฟเลกชัน (Palmer Flexion) / โดแซลเอ็กซ์เทนชัน (Dorsal Extension) (รูปที่ 2.3) ซึ่งอาจเรียกในภาษาฝรั่งเศสว่า แอนติพัลชัน (Antepulsion) / รีโทรพัลชัน (Retropulsion) การเคลื่อนไหวของนิ้วหัวแม่มือมาทางด้านฝ่ามือในระนาบ แซกิททัล เป็น แพลเมอร์ฟเลกชัน จากท่า 0 องศา (ทำนิ้วหัวแม่มือเหยียดชิดกับนิ้วชี้) ส่วนการเคลื่อนไหวของนิ้วหัวแม่มือไปทางด้านหลังมือในระนาบนี้ (โดแซลเอ็กซ์เทนชัน) จะทำไม่ได้ เนื่องจากติดกระดูก เมตาแคปัล แอนที 2



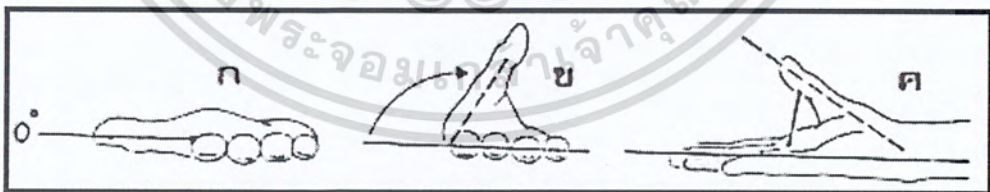
รูปที่ 2.3 การเคลื่อนไหวของข้อ คาร์โปเมตาแคปัล อันที่ 1 : แพลเมอร์ฟเลกชัน

(ก) มองทางด้านข้างของนิ้วมือ

(ข) มองทางด้านปลายนิ้วมือ

ค่าปรกติ แพลเมอร์ฟเลกชัน / โคแอสเตอเคชัน S $70^\circ / 0^\circ / 0^\circ$

ก. เซอร์คัมดักชัน (Circumduction) / รีโทรโพซิชัน (Retroposition) (รูปที่ 2.4) คือการเคลื่อนไหวที่ข้อ คาร์โปเมตาแคปัล อันที่ 1 หมุนนิ้วหัวแม่มือเข้าหานิ้วก้อย เป็นการเคลื่อนไหวผสมระหว่าง เซอร์คัมดักชัน กับ โรเตชัน (Rotation) (การเคลื่อนไหวต้องการนิ้วหัวแม่มือกับนิ้วชี้ให้ห่างกันเต็มที่) การเคลื่อนไหวของนิ้วหัวแม่มือไปทางหลังมือในทิศทางตรงกันข้าม เรียกว่า รีโทรโพซิชัน ทำ 0 องศา คือ ทำ เอ็บดักชัน เต็มที่ของนิ้วหัวแม่มือ หมุนการเคลื่อนไหวใช้มุมที่แกนตามยาวของนิ้วหัวแม่มือเคลื่อนไปจาก ระนาบแพลเมอร์



รูปที่ 2.4 การเคลื่อนไหวของข้อ คาร์โปเมตาแคปัล อันที่ 1 : เซอร์คัมดักชัน

(ก) ทำ 0 องศา

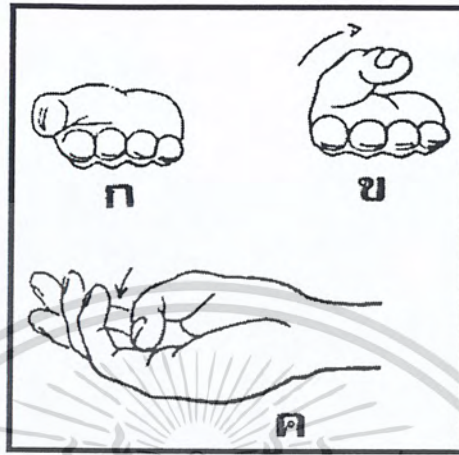
(ข) ทำ เซอร์คัมดักชัน (มองทางด้านปลายนิ้วมือ)

(ค) ทำ เซอร์คัมดักชัน (มองทางด้านข้างนิ้วก้อย)

ค่าปรกติ เซอร์คัมดักชัน / รีโทรโพซิชัน CR $90^\circ / 0^\circ / 20^\circ$

ส่วน ออปโพซิชัน (Opposition) นั้น เป็นท่าของหัวแม่มือที่เคลื่อนไหวโดยงอข้อนิ้วที่หัวแม่มือทุกข้อ แล้ววัดการเคลื่อนไหวโดยใช้ระยะห่างจากรอยพับข้อปลายนิ้วหัวแม่มือกับรอยฝ่ามืออันปลายสุด

ตามขวางบนฝ่ามือ ตรงกับตำแหน่งของข้อ เมตาคาร์โปฟาแลงกีว (Metacarpophalangeal) ของนิ้วกลาง (รูปที่ 2.5)



รูปที่ 2.5 ออปปิซิชัน ของนิ้วหัวแม่มือ

- (ก) มือปรกติในท่าก่อนเคลื่อนไหว (มองทางด้านปลายนิ้วมือ)
- (ข) มีโรเตชัน และ เซอร์คัมดักชัน
- (ค) งอข้อนิ้วหัวแม่มือทุกข้อ

2.2.2 การเคลื่อนไหวของ ข้อต่อ เมตาคาร์โปฟาแลงกีว อันที่ 1 มีการเคลื่อนไหวระนาบเดียว คือ แซกิทเทล (S) ทำ ฟเลกชัน / เอ็กซเทนชัน ทำ 0 องศา คือทำให้นิ้วหัวแม่มือตรง มุมการเคลื่อนไหวคือ มุมระหว่างแกนยาวของ พร็อกซิมอล เฟแลงซ์ (Proximal Phalanx) กับแกนตามยาวกระดูก เมตาคาเพล อันที่ 1

ค่าปรกติ ฟเลกชัน / เอ็กซเทนชัน S $50^{\circ} / 0^{\circ} / 15^{\circ}$

2.2.3 การเคลื่อนไหวของ ข้อต่อ อินเตอร์ฟาแลงกีว ของนิ้วหัวแม่มือ มีการเคลื่อนไหวระนาบเดียว คือ แซกิทเทล (S) ทำ ฟเลกชัน / เอ็กซเทนชัน ทำ 0 องศา คือทำให้นิ้วหัวแม่มือเหยียดตรง มุมการเคลื่อนไหวคือมุมระหว่างแกนตามยาวของ ดิสทอล เฟแลงซ์ (Distal Phalanx) กับของ พร็อกซิมอล เฟแลงซ์

ค่าปรกติ ฟเลกชัน / เอ็กซเทนชัน S $85^{\circ} / 0^{\circ} / 15^{\circ}$

2.2.4 การเคลื่อนไหวของ ข้อต่อ เมตาคาร์โปฟาแลงกีว อันที่ 2 - 5 (รูปที่ 2.6 ก) การเคลื่อนไหวมีระนาบเดียวคือ แซกิทเทล (S) ทำ ฟเลกชัน / เอ็กซเทนชัน ทำ 0 องศา คือทำให้นิ้วมือเหยียดตรง ฝ่ามือแบราบ มุมการเคลื่อนไหวคือมุมระหว่างแกนตามยาวของ พร็อกซิมอล เฟแลงซ์ กับแกนตามยาวของกระดูก เมตาคาเพล ของนิ้วนั้นๆ

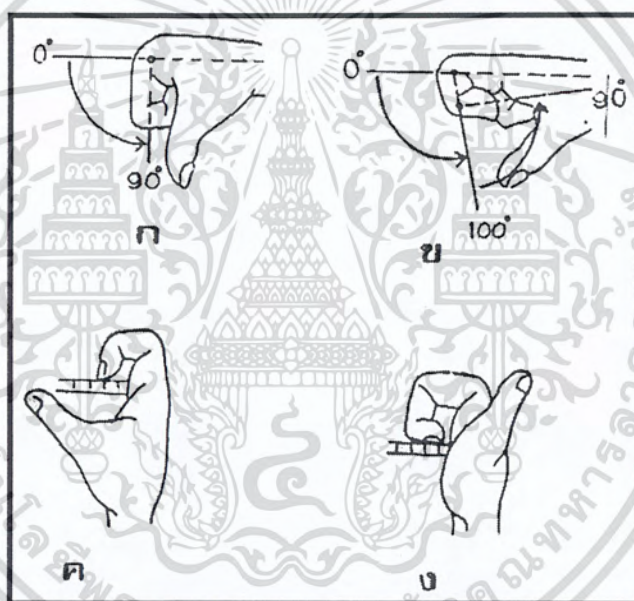
ค่าปรกติ ฟเลกชัน / เอ็กซเทนชัน S $90^{\circ} / 0^{\circ} / 30^{\circ}$

2.2.5 การเคลื่อนไหวของ ข้อต่อ พร็อกซิมอล อินเตอร์ฟาแลงกัว (นิ้วหัวแม่มือ) (รูปที่ 2.6 ข) มีการเคลื่อนไหวระนาบเดียวคือ แซกิทเทิล (S) ทำ ฟเลกชัน / เอ็กซเทนชัน ทำ 0 องศา คือทำนิ้วเหยียดตรง การเคลื่อนไหวบันทึกโดยวัดมุมระหว่างแกนตามยาวของ พร็อกซิมอล เฟแลงซ์ กับของ มิดเดิล เฟแลงซ์ (Middle Phalanx) ของนิ้วนั้นๆ

ค่าปรกติ ฟเลกชัน / เอ็กซเทนชัน S $100^{\circ} / 0^{\circ} / 0^{\circ}$

2.2.6 การเคลื่อนไหวของ ข้อต่อดิสทอล อินเตอร์ฟาแลงกัว (นิ้วหัวแม่มือ) (รูปที่ 2.6 ข) มีการเคลื่อนไหวระนาบเดียว คือ แซกิทเทิล (S) ทำ ฟเลกชัน / เอ็กซเทนชัน ทำ 0 องศา คือทำนิ้วเหยียดตรง การเคลื่อนไหวบันทึกโดยวัดมุมระหว่างแกนตามยาวของ มิดเดิล เฟแลงซ์ กับของ ดิสทอล เฟแลงซ์ ของนิ้วนั้นๆ

ค่าปรกติ ฟเลกชัน / เอ็กซเทนชัน S $80^{\circ} / 0^{\circ} / 0^{\circ}$



รูปที่ 2.6 การเคลื่อนไหวของข้อนิ้วมือ

- (ก) ฟเลกชัน ที่ ข้อต่อ เมตาคาร์โปฟาแลงกัว
- (ข) ฟเลกชัน ที่ พร็อกซิมอล และ ข้อต่อดิสทอล อินเตอร์ฟาแลงกัว
- (ค) และ (ง) การตรวจ ฟเลกชัน ของข้อนิ้วมือ โดยการวัดระยะห่างระหว่างปลายนิ้วมือถึงรอยขวางกลางฝ่ามือ (ตรงกับ ข้อต่อ เมตาฟาแลงกัว) หรือถึงรอยพับที่โคนนิ้วทางด้านฝ่ามือ แทนการวัดมุม

บทที่ 3

ไคเร็กซ์เอ็กซ์

3.1 รู้จักกับไคเร็กซ์เอ็กซ์

ไมโครซอฟท์ไคเร็กซ์เอ็กซ์เอสดีเค (ชุดพัฒนาซอฟต์แวร์) นั้นเป็นเครื่องมือที่ใช้ในการพัฒนาซอฟต์แวร์ทางมัลติมีเดียของทางบริษัทไมโครซอฟท์นั่นเอง ซึ่งรวมถึงสามารถพัฒนาโปรแกรมคอมพิวเตอร์ได้อีกด้วย โดยเริ่มแรกนั้นเราจะเห็นเฉพาะเซคเตอร์ไฟล์และไลบรารีไฟล์ของไคเร็กซ์ครอเท่านั้น หลังจากนั้นต่อมา งานด้าน 3 มิติเริ่มเข้ามามีบทบาทมากขึ้น ทางบริษัทไมโครซอฟท์จึงได้ทำการติดต่อกับทางบริษัทจากอังกฤษที่ชื่อว่า เรียลลิตี้ แล็บ เพื่อขอซื้อ เรนเดอร์มอร์ฟิกส์ มาจัดการในงานกราฟิก 3 มิตินั่นเอง ทำให้ทางไมโครซอฟท์มีความคิดที่จะเพิ่มเติมในส่วนอื่นๆ อีกเพื่อความสมบูรณ์ในตัว เอสดีเค จึงได้ออก ไมโครซอฟท์ไคเร็กซ์เอ็กซ์เอสดีเค เวอร์ชัน 2 ออกมาเพื่อทำการแนะนำตลาด ซึ่งก็ได้รับการตอบรับจากโปรแกรมเมอร์ทั้งหลายเป็นอย่างดี ทำให้เกิดการพัฒนาเวอร์ชันใหม่ขึ้นเรื่อยๆ อันได้แก่ ไมโครซอฟท์ไคเร็กซ์เอ็กซ์เอสดีเค เวอร์ชัน 3, เวอร์ชัน 5, เวอร์ชัน 6, เวอร์ชัน 7 และเวอร์ชัน 8 (ล่าสุดในขณะนี้เขียน) ทำให้มีอินเทอร์เน็ตเฟสใหม่ๆ ขึ้นมา รวมถึงการเพิ่มเมรูดบางอย่างเข้าไปอีก เพื่อตอบสนองการใช้งานให้มากขึ้นที่ผู้ใช้นั่นเอง

3.2 รู้จักกับคอม (COM)

คอม (คอมโพเนนต์ ออปเจกต์ โมเดล) เป็นส่วนประกอบที่สำคัญอย่างยิ่งของ ไมโครซอฟท์ไคเร็กซ์เอ็กซ์เอสดีเค เพราะไม่ว่าจะอยู่ส่วนไหนของโปรแกรมที่ใช้งานของ ไมโครซอฟท์ไคเร็กซ์เอ็กซ์เอสดีเคเอพีไอ จะต้องมีส่วนนี้เป็นส่วนประกอบตลอดเวลา

เนื่องจากการที่ ไคเร็กซ์เอ็กซ์เอสดีเค นั้นประกอบไปด้วยหลายๆ ส่วนโดยที่แต่ละส่วนนั้นจำเป็นอย่างยิ่งที่จะต้องมีความสัมพันธ์กันตลอด ดังนั้นจึงจำเป็นที่จะต้องมี คอม ไว้ใช้ในการสร้างความสัมพันธ์ในแต่ละส่วนในเวลาที่ต้องใช้งานร่วมกัน

คอม มีอยู่ด้วยกัน 2 ส่วนคือ คอมอินเตอร์เฟส และ คอมออปเจกต์ โดยโครงสร้างของ คอมอินเตอร์เฟส นั้นมีลักษณะโครงสร้างต่างๆ เหมือนกับการสร้างคลาสของ ซีพลัสพลัส เป็นอย่างมาก โดยมีการรวมตัวแปรและฟังก์ชันไว้ในตัวของมัน ส่วน คอมออปเจกต์ ก็จะเป็นตัวคอยชี้สั่งการว่า ในอินเตอร์เฟสนั้นจะให้ทำหน้าที่อะไร

ยกตัวอย่างเช่น : `ddraw -> Initialize (ddraw, NULL);`

อธิบายได้ว่า : `ddraw` เป็น คอมออปเจกต์ โดยบอกให้ `IdirectDraw4` อินเตอร์เฟส ทำการ Initialize

3.3 ส่วนประกอบของไดเรกซ์เอ็กซ์

เนื่องด้วยเราได้ใช้ชุดของไดเรกซ์เอ็กซ์เวอร์ชัน 6 ดังนั้นจึงขออธิบายโครงสร้างส่วนประกอบของไดเรกซ์เอ็กซ์เวอร์ชัน 6 ว่ามีส่วนประกอบสำคัญๆ ดังนี้

3.3.1 ไดเรกซ์ดรอ เป็นส่วนประกอบหลักอย่างแรกของ ไดเรกซ์เอ็กซ์เอสดีเค เลขที่เดียวเนื่องจากมีหน้าที่จัดการกับการแสดงผลทางจอภาพ และให้ความเร็วในการประมวลผลสูงกว่าการใช้คอสและการใช้จีดีไอ (กราฟิก ดีไวซ์ อินเตอร์เฟส) ของวินโดวส์ เนื่องจากการส่งข้อมูลที่ทำได้โดยตรงนั่นเอง อีกทั้งยังประกอบไปด้วยฟังก์ชันต่างๆ ที่คอยสนับสนุน การโปรแกรมแบบ 2 มิติ ซึ่งส่วนแล้วแต่มีประสิทธิภาพแทบทั้งสิ้น

3.3.2 ไดเรกซ์สามดี มีประสิทธิภาพสูงมาก ในการใช้ในการสร้างโปรแกรม 3 มิติบนวินโดวส์ เนื่องจากการได้เปรียบจากการสามารถใช้งานร่วมกับการ์ดเร่งความเร็ว 3 มิติได้อย่างมีประสิทธิภาพเพราะมีฟังก์ชันคอยสนับสนุนอยู่นั่นเอง มีส่วนประกอบอยู่ด้วยกัน 2 ส่วนคือ ไดเรกซ์สามดี รีเทนนิ่ง โหมดเอฟไอ และ ไดเรกซ์สามดี อิมมิเดียต โหมดเอฟไอ

3.3.3 ไดเรกซ์ซาวน์ มีฟังก์ชันที่ช่วยให้โปรแกรมเมอร์ทำงานได้ง่ายขึ้น ในการใส่เสียงเพลงประกอบและเสียงเอฟเฟกต์ต่างๆ เข้าไปในโปรแกรม โดยหากว่าเราได้ทำความเข้าใจกับหลักสถาปัตยกรรมของไดเรกซ์ซาวน์แล้ว เราจะสามารถทำงานได้อย่างมีประสิทธิภาพ

3.3.4 ไดเรกซ์เพลย์ ทำให้การสร้างเกมระบบผู้เล่นหลายๆ คน ทำได้ง่ายยิ่งขึ้น และสะดวกในการใช้งานร่วมกับโมเด็ม และระบบเน็ตเวิร์กต่างๆ

3.3.5 ไดเรกซ์อินพุต ทำให้การสร้างโปรแกรมเพื่อรองรับระบบการสั่งงานจากผู้เล่นไม่ว่าจะเป็นจอยสติค เมาส์ หรือคีย์บอร์ดทำได้ง่ายและมีประสิทธิภาพเป็นอย่างยิ่ง อีกทั้งยังมีฟังก์ชันที่สนับสนุนการเพิ่มเติมอุปกรณ์การเล่นที่จะมีขึ้นมาใหม่อีกในอนาคต

3.3.6 ไดเรกซ์มิวสิก มีลักษณะที่คล้ายๆ กับไดเรกซ์ซาวน์นั่นเอง แต่จะมีฟังก์ชันที่คอยสนับสนุนการใช้งานไฟล์มิดิต่างๆ

3.3.7 ไดเรกซ์เซตอัป ช่วยในการติดตั้งส่วนประกอบที่จำเป็นต้องใช้ของไดเรกซ์เอ็กซ์ให้กับผู้ใช้ อย่างอัตโนมัติ ทำให้สะดวกอย่างยิ่ง

3.3.8 ออโต้เพลย์ เป็นส่วนประกอบที่สำคัญของระบบวินโดวส์ ทำให้โปรแกรมสามารถที่จะทำการติดตั้งลงในฮาร์ดดิสก์ได้อย่างอัตโนมัติ โดยโปรแกรมติดตั้งจะเริ่มทำงานเมื่อผู้ใช้นำแผ่นโปรแกรมใส่เข้าไปในไดรฟ์ซีดีรอมนั่นเอง

3.4 สิ่งที่ต้องกำหนดในไมโครซอฟท์วิซวลเบสิก

เนื่องจาก ไมโครซอฟท์ไดเรกซ์เอ็กซ์เอสดีเค นั้นยังไม่ได้ทำการสนับสนุนการเขียนแบบ วิน32 (Win32) ใน ไมโครซอฟท์วิซวลเบสิก ดังนั้นการที่จะใช้งานก็จำเป็นอย่างยิ่งที่จะต้องเขียนการประกาศฟังก์ชันขึ้นมาเอง ซึ่งก็จะเป็นการสิ้นเปลืองเวลาโดยใช่เหตุ เนื่องจากว่าปัจจุบันนี้ระบบอินเทอร์เน็ตได้เข้า

มามีบทบาทเป็นอย่างยิ่งในการติดต่อสื่อสาร ทำให้เราได้รับการช่วยเหลือจากผู้ที่มีความรู้มากกว่า ได้ทำการเขียน ไลต์ไบลารี (Type Library) ขึ้นมาให้ ซึ่งก็คือไฟล์ Directx6.tlb

วิธีการกำหนดให้ใช้ไคลเร็กซ์เอ็กซ์ได้ในวิซวลเบสิกมีดังนี้

3.4.1 เลือกเมนู Projects และทำการเลือกคำสั่ง Reference

3.4.2 เลือกคำสั่ง Browse แล้วทำการเลือกไฟล์ Directx6.tlb

3.4.3 ทำการติ๊กถูกที่บรรทัด DirectX 6 Type Library v 0.8 ณ Patrice Scribe 1998

3.5 ูโลกลสามมิติ

อย่างไรก็ดี หากเรามองลึกเข้าไปในไคลเร็กซ์เอ็กซ์ จะเห็นว่ามันประกอบด้วยอินเตอร์เฟสและเมทรีดอยู่มากมายจนก่อให้เกิดความงุนงงและสับสน แต่ถ้าหากเราต้องการใช้ไคลเร็กซ์เอ็กซ์เพื่อสร้างอินเตอร์เฟสสามมิติต่างๆ ก็ไม่จำเป็นจะต้องทราบรายละเอียดทั้งหมดของไคลเร็กซ์เอ็กซ์

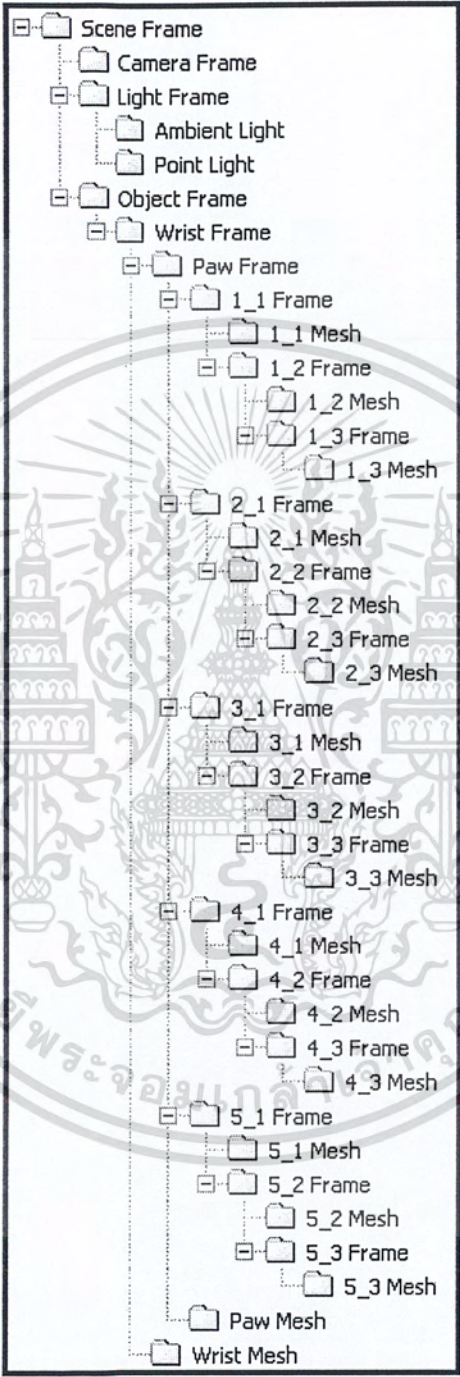
ตัวอย่างเช่น เมื่อต้องการสร้างแอปพลิเคชันสามมิติ สิ่งที่เราต้องใช้คืออินเตอร์เฟส ไคลเร็กซ์สามมิติ โดยมีสองทางเลือกคือ การเขียนโปรแกรมแบบอิมมิตีวโมด หรือ รีเทนนิ่งโมด ซึ่งอิมมิตีวโมดจะใช้งานยากและทำงานใกล้ชิดกับฮาร์ดแวร์ ในขณะที่รีเทนนิ่งโมดใช้งานได้ง่ายกว่าและใกล้เคียงกับวิซวลเบสิก ดังนั้นเราควรเลือกใช้รีเทนนิ่งโมดเป็นอันดับแรก อย่างไรก็ตามการใช้งานง่ายของรีเทนนิ่งโมดทำให้มันมีสมรรถภาพต่ำกว่าอิมมิตีวโมด

สิ่งแรกที่เราควรทราบคือ ในการเขียนแอปพลิเคชันที่ต้องการ เราต้องใช้ออปเจกต์ ไคลเร็กซ์เอ็กซ์ 6 ตัวใดบ้าง ตัวอย่างเช่น โปรแกรมเราใช้ไคลเร็กซ์สามมิติ เราก็ควรจะรู้ว่าไคลเร็กซ์สามมิติมีคลาสอะไรบ้าง (ดูตารางที่ 3.1)

ต่อไปคือ การสร้างซีนกราฟ ซึ่งใช้เพื่อเก็บข้อมูลต่างๆ ที่สร้าง โดยรีเทนนิ่งโมด จะใช้ออปเจกต์เฟรมเพื่อช่วยให้เราสามารถสร้างและจัดการซีนกราฟได้ง่ายขึ้น ซีนกราฟประกอบด้วยลำดับชั้นเฟรมซึ่งแต่ละเฟรมประกอบด้วยออปเจกต์รีเทนนิ่งโมดอย่างน้อยหนึ่งตัว ซึ่งตัวอย่างซีนกราฟแสดงดังรูปที่ 3.1

เฟรมระดับบนสุดเรียกว่าเฟรมซีน ซึ่งปกติแล้วประกอบด้วยเฮอร์คาเมร่า, เฟรมไลท์ และเฟรมออปเจกต์ โดยเฟรมคาเมร่าทำหน้าที่เกี่ยวกับแนวคิดภายในซีนและเฟรมไลท์ทำหน้าที่เก็บออปเจกต์ที่ไม่ได้ปรากฏให้เห็นแต่ใช้เพื่อทำให้ออปเจกต์อื่นแสดงออกมา ท้ายสุดเฟรมออปเจกต์ประกอบด้วยเฟรมลูกที่ทำหน้าที่เก็บรูปทรงสามมิติที่สร้างขึ้นมาจากโค้ดโดยตรงหรือโหลดจากไฟล์ที่มีอยู่แล้ว

รีเทนนิ่งโมดของไคลเร็กซ์เอ็กซ์ 6 นั้นสนับสนุนแสงห้าแบบคือ ไคลเร็กซ์นอล, แอมเบียน, พ็อตน์, พ็อตน์-พาราแรล และ สปอตไลท์ เราใช้ออปเจกต์ไลท์ส่องแสงให้กับรูปทรงต่างๆ เพื่อสร้างซีน จากนั้นไคลเร็กซ์เอ็กซ์ 6 จะปรับแต่งสีของรูปทรงแต่ละอันตามตำแหน่งของแสงที่ส่องใส่ออปเจกต์นั้นโดยก่อนที่จะนำแสงไปส่องใส่อุปทรงในซีน เราต้องนำแสงไปเชื่อมต่อกับเฟรมเพื่อจัดตำแหน่งแสง จากนั้นเราสามารถปรับตำแหน่งแสงโดยผ่านทางเฟรมที่แสงเชื่อมต่ออยู่



รูปที่ 3.1 ตัวอย่างชีนกราฟ

คลาส	คำอธิบาย
Direct3DRM3	ใช้เพื่อสร้างออปเจกต์รีเทนนิ่งโหมคที่ต้องการ, โหลดรูปทรงสามมิติ, ใช้พื้นผิวดูและจัดการตัวแปรระดับสูงบางตัว
DirectDraw6	ใช้เพื่อพูดคุยกับกับฮาร์ดแวร์สองมิติ สำหรับโปรแกรมเราเนี่ย เราใช้มันเพื่อแทนคลาส Direct3DRMDevice3 เท่านั้น
Direct3DRMFrame3	เราสร้างคลาสนี้จากคลาส Direct3DRM3 เพื่อเก็บรูปทรงสามมิติ, แสงไว้ในเฟรมก่อนที่จะแสดงออกมา เรายังใช้สี, เปลี่ยนรูปแบบและความเร็วของออปเจกต์ผ่านทางเฟรมนี้เช่นกัน
Direct3DRMDevice3	เราสร้างคลาสนี้จากคลาส Direct3DRM3 เพื่อควบคุมสถานที่ที่กำหนดขึ้นผ่านทาง แชนเดิลวินโดว์
Direct3DRMViewport	เราสร้างคลาสนี้จากคลาส Direct3DRM3 เพื่อจัดการการกำหนดขึ้น เรายังใช้คลาสนี้เพื่อหาว่าผู้ใช้คลิกรูปทรงสามมิติอันใด

ตารางที่ 3.1 คลาสหลักที่จะใช้ในการทำโปรแกรม

3.6 แสดงผลลัพธ์ของไดเรกซ์เอ็กซ์ 6

ที่ผ่านมาเราได้ทราบถึงวิธีการเริ่มต้นสร้างออปเจกต์หลักของไดเรกซ์เอ็กซ์ 6 และโครงการเพื่อเก็บส่วนประกอบต่างๆ ของวัตถุสามมิติ ขั้นตอนต่อไปคือการจัดเตรียมวินโดว์เพื่อให้ไดเรกซ์เอ็กซ์ 6 ใช้เพื่อแสดงขึ้นที่เราสร้างขึ้น โดยกระบวนการใช้ไดเรกซ์เอ็กซ์ 6 กับวินโดว์มีความซับซ้อนเล็กน้อย เราต้องสร้างออปเจกต์คลิปเปอร์ด้วยแชนเดิลวินโดว์ของฟอร์มหรือฟิเจอร์บ็อกที่ใช้สำหรับแสดงผล จากนั้นใช้ออปเจกต์คลิปเปอร์นี้เพื่อสร้างดีไวซ์ซึ่งเป็นออปเจกต์อันทรงพลังที่ใช้ควบคุมขึ้นแล้วใช้ออปเจกต์ดีไวซ์เพื่อสร้างออปเจกต์วิวพอร์ต โดยระบุความกว้าง, ความสูง และออปเจกต์เฟรม (ซึ่งหมายความว่าเราต้องสร้างวิวพอร์ตใหม่ทุกครั้งที่มีการปรับขนาดฟอร์ม) ออปเจกต์เฟรมที่ระบุให้ครีเอตวิวพอร์ตนี้ใช้เพื่อแสดงมุมมองของภาพสามมิติซึ่งปกติแล้วคือเฟรมคามเรานั่นเอง ดังนั้นการเปลี่ยนตำแหน่งด้วยเมทอด Set Position หรือเปลี่ยนรูปแบบเฟรมคามเรานี้คือการเปลี่ยนมุมมองขึ้น

ตอนนี้เราได้อุปกรณ์ประกอบพื้นฐานอันได้แก่การส่องแสงขึ้นกราฟและการจัดขนาดออปเจกต์วิวพอร์ตและออปเจกต์ดีไวซ์เพื่อสร้างขึ้นในฟอร์มวิซวลเบสิกแล้ว ขั้นตอนที่เหลือคือการสร้างรูปทรงสามมิติขึ้นมาตามต้องการ โดยรีเทนนิ่งโหมคมีออปเจกต์ เมฆ ที่ประกอบด้วยออปเจกต์เฟส (กำหนดพื้นผิว), เวกทิส (กำหนดจุดตัด) และเวกเตอร์ (กำหนดทิศทาง) เพื่อใช้กำหนดรูปทรงสามมิติขึ้นมา

นอกจากการสร้างรูปทรงด้วยการกำหนดลักษณะต่างๆ ผ่านทางออปเจกต์เมชขึ้นมา เรายังสามารถจัดเตรียมไฟล์ขึ้นมาก่อนแล้วโหลดไฟล์เข้าไปในออปเจกต์ MeshBuilder ซึ่งใช้เพื่อสร้างเมชสามมิติขึ้นมา เราจะพบไฟล์เมชจำนวนมากบนอินเทอร์เน็ตรวมถึงไฟล์ตัวอย่างของวิชวลเบสิกที่มาพร้อมกับไดเรกซ์เอ็กซ์เอสดีเค นอกจากนี้ เอสดีเคยังมีโปรแกรมยูทิลิตี้ Convs3ds.exe เพื่อแปลงไฟล์มาตรฐานอุตสาหกรรมเมช (.3ds) เป็นไฟล์เมชของไดเรกซ์เอ็กซ์ (.x)

การโหลดไฟล์เมชที่ทำไว้ล่วงหน้าจะช่วยลดความยุ่งยากในการทำงานกับออปเจกต์เมชโดยตรง เพราะเราเพียงใช้ออปเจกต์ Direct3DRMMeshBuilder3 เพื่อโหลดและจัดการเมชเท่านั้นก็เป็นอันเสร็จสิ้น

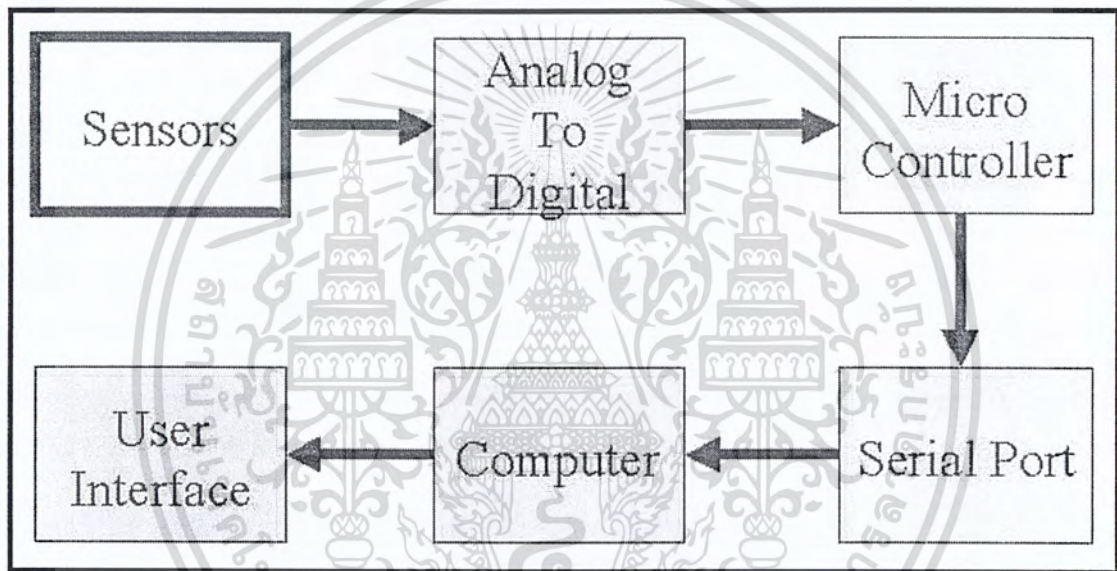


บทที่ 4

การออกแบบในส่วนของฮาร์ดแวร์

4.1 การออกแบบวงจรโดยรวม

เมื่อเราคิดจะสร้างอุปกรณ์ขึ้นมาชิ้นหนึ่งนั้น เราจะต้องมาคิดถึงภาพโดยรวมของอุปกรณ์ทั้งหมดว่าจะทำงานเป็นส่วนๆ อย่างไรบ้างซึ่งโครงงานนี้จะสร้างอุปกรณ์ที่มีการทำงานเป็นส่วนๆ โดยรวมดังรูปที่ 4.1



รูปที่ 4.1 บล็อกไดอะแกรมโดยรวม

จากรูปที่ 4.1 จะเห็นได้ว่าโครงงานนี้แบ่งออกได้เป็น 6 ส่วนใหญ่ๆ คือ

4.1.1 เซนเซอร์ : เมื่อเราคิดจะตรวจจับการเคลื่อนไหวของข้อมือแล้ว เราจะต้องมาคิดว่าเราจะใช้อะไรในการตรวจจับการเคลื่อนไหวนั้น เราจึงต้องศึกษาเซนเซอร์ชนิดต่างๆ เพื่อที่เราจะได้เลือกชนิดของเซนเซอร์ที่เหมาะสมกับโครงงานนี้ รวมไปถึงการออกแบบตัวอุปกรณ์ที่ทำให้เซนเซอร์เหล่านี้สามารถรับข้อมูลการเปลี่ยนแปลงของข้อมือและนิ้วมือได้อย่างเหมาะสม

4.1.2 วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล : สัญญาณที่เราจะได้รับจากเซนเซอร์นั้นไม่ใช่สัญญาณที่เราจะใช้สื่อสารติดต่อกับไอซีต่างๆ ได้ เพราะว่าสัญญาณต่างๆ ที่ใช้ในไอซีนั้นจะเป็นสัญญาณดิจิทัล (0 กับ 1) คือจะไม่มีค่าแรงดันสัญญาณนอกไปจาก 0 โวลต์ กับ 5 โวลต์ ดังนั้นเราจึงต้องหาวิธีแปลงค่าสัญญาณเพื่อที่จะเราได้้นำค่าที่ได้ส่งเข้าคอมพิวเตอร์ได้

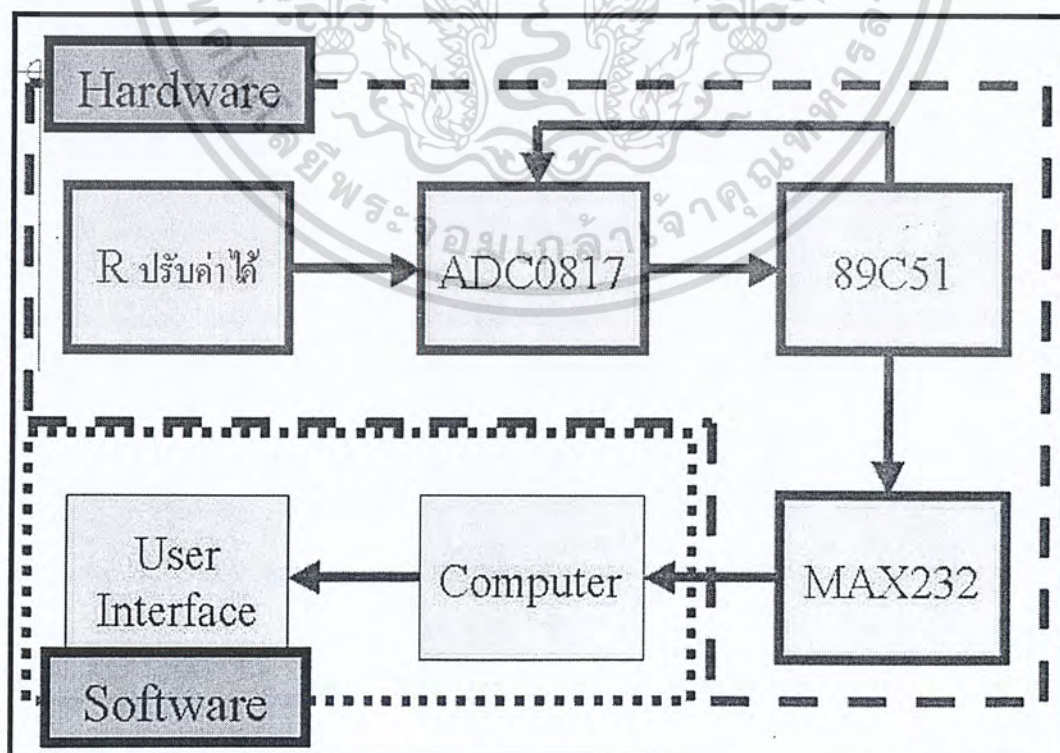
4.1.3 ไมโครคอนโทรลเลอร์ : เป็นหน่วยประมวลผลกลางของวงจรที่เราจะสร้างขึ้น เพื่อทำหน้าที่ในการรับค่าจากเซนเซอร์จากมือของเราที่ถูกแปลงเป็นสัญญาณดิจิตอลเรียบร้อยแล้ว แล้วทำการจัดเรียงข้อมูลให้เรียบร้อย เพราะว่าเราจะต้องทำการส่งค่าที่ได้เข้าสู่คอมพิวเตอร์ซึ่งจะต้องส่งข้อมูลที่ได้ผ่านพอร์ตอนุกรม ซึ่งเป็นการส่งข้อมูลแบบเรียงบิตทีละบิต เราจึงต้องทำการเรียงข้อมูลให้เรียบร้อยก่อน แล้วจึงค่อยทำการส่งอีกทีหนึ่ง

4.1.4 วงจรแปลงสัญญาณ : เพราะจากวงจรที่เราสร้างเพื่อจะนำข้อมูลส่งผ่านสายผ่านพอร์ตอนุกรมเข้าคอมพิวเตอร์นั้น วงจรจะต้องทำการส่งข้อมูลออกไปสู่อนุกรม ดังนั้นจึงต้องมีการแปลงค่าหลายๆ อย่าง เพื่อที่จะทำการส่งข้อมูลไปสู่ที่อื่นได้

4.1.5 คอมพิวเตอร์ : ทำการรับข้อมูลจากวงจรที่เราส่งผ่านสายเข้ามา เพราะว่าถ้าวงจรเราส่งข้อมูลเข้ามาแต่ไม่มีคนรับ (ไม่ได้ทำส่วนรับข้อมูลไว้) ก็เหมือนมีคนโยนของให้แต่ไม่มีคนรับ ของก็เสียเปล่า ดังนั้นเราจึงต้องเขียนโปรแกรมรับข้อมูลส่วนนี้ไว้ด้วย

4.1.6 การเขียนโปรแกรมกราฟฟิโกอินเตอร์เฟส : เพื่อที่จะนำผลลัพธ์จากวงจรที่ได้รับมาแสดงผลเป็นกราฟฟิก 3 มิติ เพื่อให้เราเข้าใจได้ง่ายว่าเรารับข้อมูลจากเซนเซอร์ได้ถูกต้องอย่างที่เราร้องการหรือไม่

จากบล็อกไดอะแกรมโดยรวมในรูปที่ 4.1 เราสามารถแบ่งออกได้เป็น 2 ส่วนใหญ่ ๆ คือ ส่วนของฮาร์ดแวร์ และส่วนของซอฟต์แวร์เพื่อนำไปใช้ในการออกแบบดังรูปที่ 4.2 ซึ่งรายละเอียดในส่วนของฮาร์ดแวร์จะอธิบายในหัวข้อต่อไป ส่วนของซอฟต์แวร์นั้นจะอธิบายในบทต่อไป



รูปที่ 4.2 บล็อกไดอะแกรมที่นำไปออกแบบ

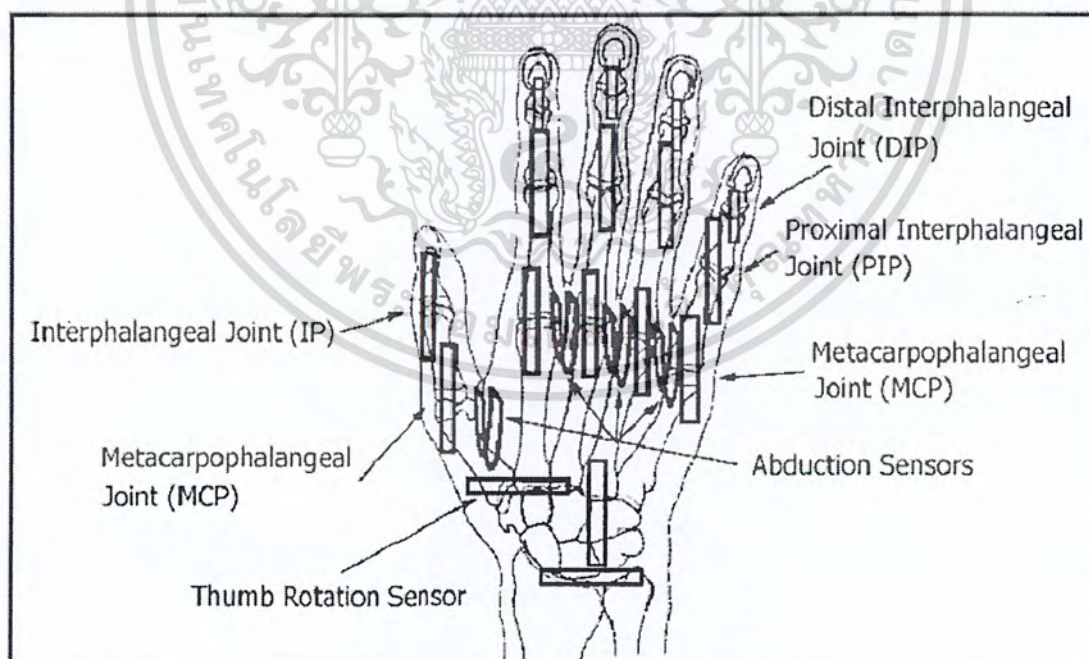
4.2 การออกแบบในส่วนของฮาร์ดแวร์

ในส่วนของฮาร์ดแวร์นั้นถ้าแบ่งตามอุปกรณ์ที่จะต้องทำจะสามารถแบ่งเป็น 2 ส่วนใหญ่ ๆ คือ ส่วนของถุงมือที่ใช้เซนเซอร์เป็น R ปรับค่าได้ และส่วนที่สองคือส่วนของวงจรซึ่งทำหน้าที่แปลงสัญญาณจากอนาล็อกเป็นดิจิทัลและเรียบเรียงส่งข้อมูลไปยังคอมพิวเตอร์ผ่านพอร์ตอนุกรม

4.2.1 การออกแบบฮาร์ดแวร์ในส่วนของถุงมือ

4.2.1.1 แนวคิดในการออกแบบถุงมือ

จากการศึกษาการเคลื่อนไหวของนิ้วมือและข้อมือในบทที่ 2 และเลือกใช้เซนเซอร์เป็นความต้านทานปรับค่าได้ทำให้เราทราบว่าเราต้องใช้ความต้านทานปรับค่าได้ทั้งหมด จำนวน 22 ตัว ตามข้อนิ้วต่างและแต่ละข้อ โดยที่แต่ละนิ้วนั้นจะใช้ 4 ตัว 3 ตัวจะใช้วัดการเคลื่อนไหวในแนว ฟเลกชัน / เอ็กซเทนชัน ข้อของนิ้วซึ่งแต่ละนิ้วมี 3 ข้อและความต้านทานอีก 1 ตัวสำหรับวัดการเคลื่อนไหวของนิ้วในแนว แอ็บดักชัน / แอ็ดดักชัน ซึ่งจะใช้ความต้านทานปรับค่าได้ทั้งหมด 20 ตัว และอีกสองตัวใช้สำหรับข้อมือโดยตัวหนึ่งใช้วัดการเคลื่อนไหวในแนว ฟเลกชัน / เอ็กซเทนชัน และวัดการเคลื่อนไหวในแนว แอ็บดักชัน / แอ็ดดักชัน อีกตัวหนึ่ง รวมเป็น 22 ตัว ซึ่งวางตามตำแหน่งบนมือดังรูปที่ 4.3

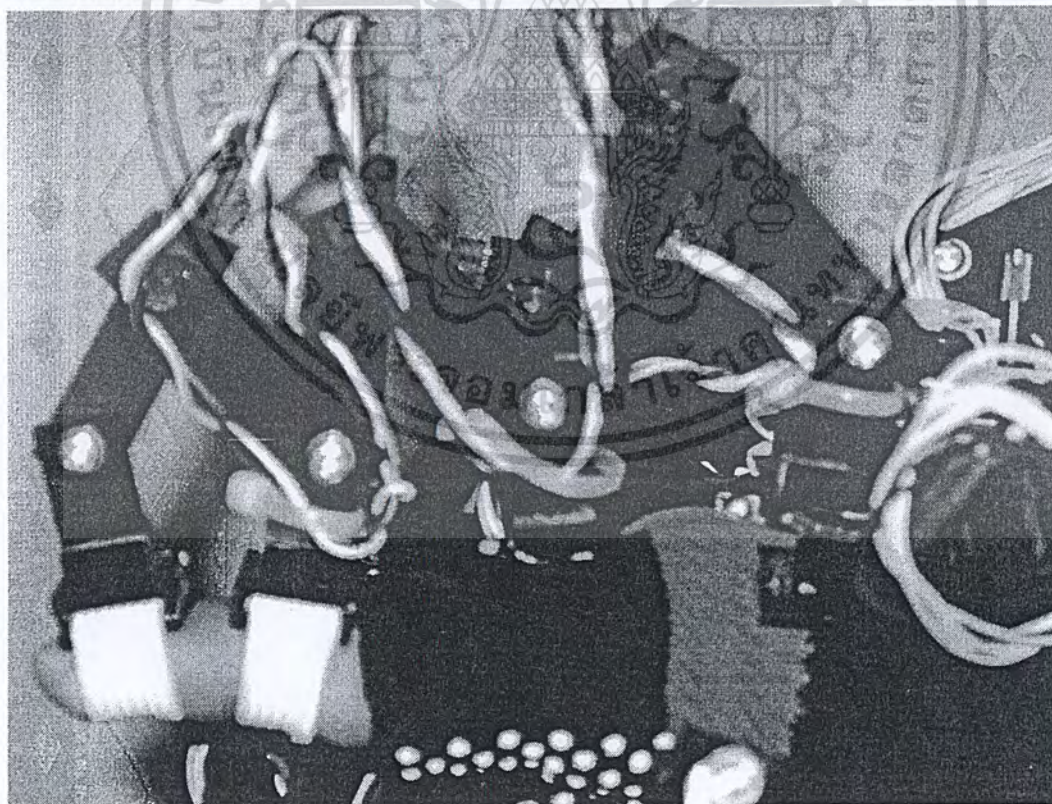


รูปที่ 4.3 ตำแหน่งการวางเซนเซอร์บนมือ

4.2.1.2 การออกแบบตัวถุงมือ

สำหรับการออกแบบตัวถุงมือให้จับการเคลื่อนไหวของข้อมือและนิ้วมือนั้นในส่วนของนิ้วมือและข้อมือนั้นจะมีการเคลื่อนไหวใน 2 แนวหลัก ๆ คือ แนว ฟเลกชัน / เอ็กซเทนชัน และแนว แอ็บดักชัน / แอดดักชัน ส่วนของนิ้วโป้งนั้นจะมีการเคลื่อนไหวที่ซับซ้อนเพิ่มขึ้นอีกก็คือการโรเตชันของนิ้วมือนั้นการออกแบบตัวถุงมือจึงแบ่งออกเป็น 3 ส่วนดังนี้

การเคลื่อนไหวในแนวฟเลกชัน / เอ็กซเทนชัน นั้นจะมีในข้อ DIP, PIP, MCP ของนิ้วชี้, กลาง, นาง และก้อย ส่วนนิ้วโป้งนั้นจะมีอยู่ในข้อ IP และ MCP และในส่วนของข้อมือ สำหรับการออกแบบตัวถุงมือเพิ่มในสามารถตรวจจับการเคลื่อนไหวในแนวนั้นนั้นจะมีลักษณะเป็นคาน 2 ชั้นที่ปลายด้านหนึ่งของทั้งสองบรรจบกันโดยขึ้นติดอยู่กับตัวฐานเซนเซอร์ และอีกชั้นหนึ่งติดอยู่กับแกนหมุนของเซนเซอร์ ซึ่งมีลักษณะคล้ายกับเข็มฉีดยาและเข็มยาวของนาฬิกาที่มีจุดหมุนร่วมกันคือตัวเซนเซอร์ ดังนั้นเมื่อคานมีการหมุนก็จะทำให้เซนเซอร์เปลี่ยนแปลงค่าไปด้วย ส่วนปลายด้านที่เหลือของคานทั้งสองจะติดอยู่บนฐานที่สามารถหมุนจากกันได้โดยอิสระ ซึ่งเราก็นำฐานนั้นติดบนข้อนิ้วที่อยู่ติดกันทำให้เซนเซอร์นั้นอยู่ระหว่างข้อนิ้วและวัดการเคลื่อนไหวของข้อค่อนิ้วนั้นได้โดยเมื่อให้ข้อค่อนิ้วเป็นจุดหมุนก็จะทำให้คานที่ติดอยู่บนฐานซึ่งอยู่บนข้อนิ้วนั้นเคลื่อนไหวไปด้วย ส่งผลให้คานมีการบิดและหดตัวตามการหมุนของข้อนิ้วซึ่งเมื่อคานมีการเคลื่อนไหวก็จะส่งผลให้เซนเซอร์เปลี่ยนแปลงค่าไปด้วยดังรูปที่ 4.4



รูปที่ 4.4 การออกแบบในส่วนการเคลื่อนไหวในแนวฟเลกชัน / เอ็กซเทนชัน

การเคลื่อนไหวกว้างในแนวแอ็บดักชัน / แอ็ดดักชัน นั้นจะมีในข้อ MCP ของนิ้วทุกนิ้วและข้อมือซึ่งการออกแบบก็มีลักษณะคล้ายกับส่วนของการตรวจจับการเคลื่อนไหวกว้างในแนวฟเลกชัน / เอ็กเทนชันแต่มีลักษณะสั้นกว่าเพราะการเคลื่อนไหวกว้างนี้มีองศาไม่มากนักโดยปลายคานาด้านหนึ่งติดอยู่กับปลายของคานาของเซนเซอร์ที่วัดในแนวฟเลกชัน / เอ็กเทนชัน ของข้อ MCP เองและปลายคานาอีกด้านหนึ่งติดอยู่กับตัวถ่วงมือดังรูปที่ 4.5

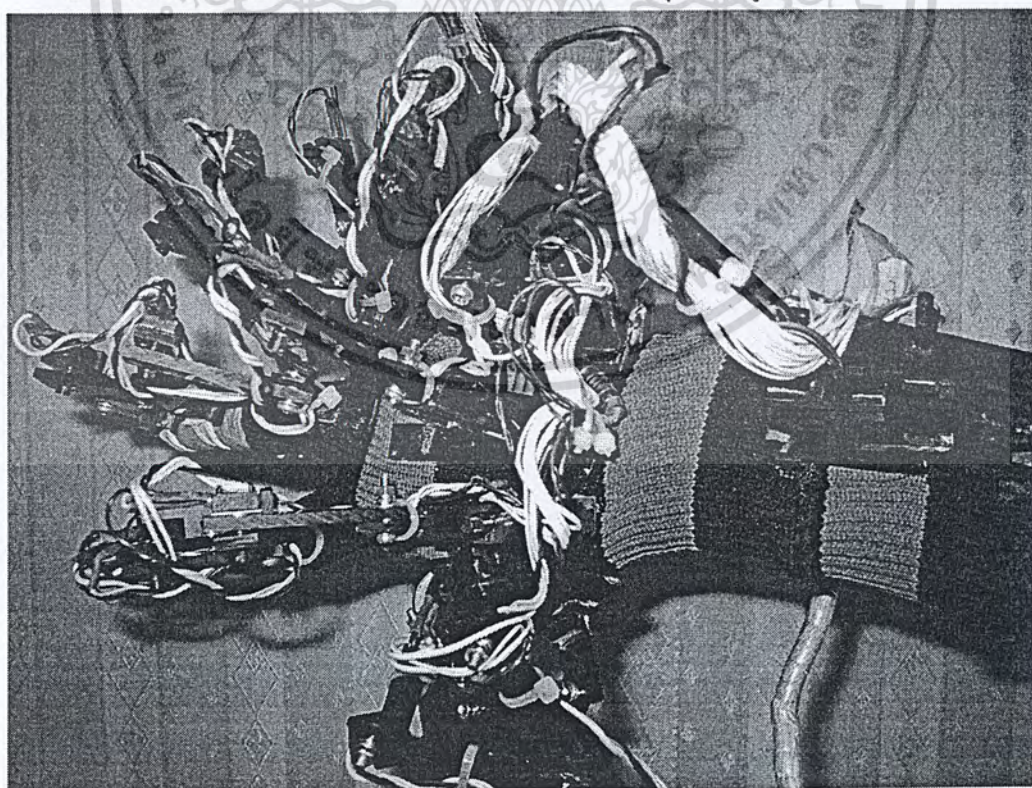


รูปที่ 4.5 การออกแบบในส่วนการเคลื่อนไหวกว้างในแนวแอ็บดักชัน / แอ็ดดักชัน

การเคลื่อนไหวกว้างของนิ้วโป้งในแนวโรเตชัน สำหรับการออกแบบเซนเซอร์เพื่อตรวจจับในส่วนนี้นั้นจะคล้ายกับส่วนที่กล่าวมาข้างต้นโดยคานาด้านหนึ่งทำเป็นฐานซึ่งจะติดอยู่กับถ่วงมือส่วนคานาอีกด้านจะติดกับปลายคานาที่วัดการเคลื่อนไหวกว้างในแนวแอ็บดักชัน / แอ็ดดักชัน ของข้อ MCP ของนิ้วโป้งดังแสดงในรูปที่ 4.6



รูปที่ 4.6 การเคลื่อนไหวของนิวโ่งในแนวโรเตชัน
ซึ่งเมื่อนำการออกแบบในแต่ละส่วนมาประกอบกันจะได้ดังมื่อดังรูปที่ 4.7



รูปที่ 4.7 รูปแบบถุงมือที่ออกแบบ

4.2.2 การออกแบบฮาร์ดแวร์ในส่วนของวงจร

4.2.2.1 แนวคิดของการออกแบบวงจร

ในโครงการนี้ เราได้ศึกษาเซนเซอร์มาหลายชนิด ซึ่งหลักๆ มีดังนี้

- ความต้านทานปรับค่าได้
- เบนเซนเซอร์ (Bend Sensors)
- ฮอลล์เอฟเฟกต์เซนเซอร์ (Hall Effect Sensors)
- โคดดิชเซนเซอร์ (Code Disc Sensors)
- สเตรนเกจ (Strain Gauge)
- ไฟเบอร์ออปติก (Fiber Optic)

ซึ่งในโครงการนี้ เราเลือกใช้ความต้านทานปรับค่าได้เป็นเซนเซอร์ในวงจร เมื่อได้เซนเซอร์แล้ว ส่วนที่เราจะต้องพิจารณาถัดไปคือวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล ซึ่งได้เลือกใช้ไอซี ADC0817 ในโครงการนี้ เนื่องจากเราเลือกความต้านทานปรับค่าได้เป็นเซนเซอร์ของเรา ดังนั้นเราเพียงแค่อ่านค่าความต้านทานที่เปลี่ยนแปลงไป มาเปลี่ยนเป็นค่าแรงดันที่เปลี่ยนไปแทน โดยการจ่ายแรงดันไฟฟ้าผ่านความต้านทานปรับค่าได้ ดังสมการ $V = IR$ ซึ่งจะเห็นจากสมการ จะได้ว่าค่าแรงดันไฟฟ้า (V) นั้นแปรผันตามกับค่าความต้านทาน (R) $V \propto R$ เราจึงอาศัยค่าแรงดันที่เปลี่ยนไปนั้นผ่าน ADC ธรรมชาติ ซึ่งจะทำการเปลี่ยนค่าแรงดันไฟฟ้าที่เปลี่ยนแปลงไป ไปเป็นสัญญาณดิจิทัลได้

จากนั้น เราก็มาหาวิธีการในการจัดรูปแบบการทำงานของวงจรทั้งหมด ซึ่งหัวใจแห่งการทำงานทั้งหมดของวงจรมันก็คือ ไมโครคอนโทรลเลอร์ ซึ่งเป็นไอซีอเนกประสงค์ เราสามารถป้อนโปรแกรมสั่งให้ไอซีตัวนี้ทำงานต่างๆ ตามที่เราต้องการได้ ในโครงการของเราต้องการไอซีนี้ในการควบคุมการรับข้อมูลจาก ADC0817 ที่จะทำการรับข้อมูลที่ส่งค่าจากเซนเซอร์ตัวไหนในแต่ละเวลาหนึ่งๆ เพื่อที่จะได้นำข้อมูลที่ได้รับเข้ามาทำการจัดเรียงให้เหมาะสม เพื่อที่จะได้นำข้อมูลส่งต่อออกไปสู่คอมพิวเตอร์ผ่านพอร์ตอนุกรม ซึ่งไอซีตัวนี้เป็นตัวควบคุมหลักของวงจรนี้เลยทีเดียว

วงจรแปลงสัญญาณ เป็นวงจรที่ใช้แปลงค่าแรงดันไฟฟ้าที่ได้รับจากไมโครคอนโทรลเลอร์ กับคอมพิวเตอร์ เพราะว่าไมโครคอนโทรลเลอร์มีแรงดันไฟฟ้าที่ใช้ในวงจรไม่เท่ากับที่พอร์ตอนุกรมใช้ส่งสื่อสารข้อมูลต่างๆ เนื่องจากสาเหตุต่างๆ เช่นระยะทางของการส่งข้อมูล ในที่นี้เราเลือกใช้ MAX232 ในการเป็นไอซีที่ใช้แปลงสัญญาณ

4.2.2.2 อุปกรณ์หลักที่ใช้ในวงจร

1. ตัวต้านทานปรับค่าได้ (10 K Ω) เป็นจำนวน	22	ตัว
2. ADC0817	2	ตัว
3. 89C51	1	ตัว
4. MAX232	1	ตัว
5. หัวพอร์ตอนุกรมตัวผู้และตัวเมีย	1	ชุด
6. หัวพอร์ต DB37 ตัวผู้และตัวเมีย	2	ชุด
7. NE555	1	ตัว
8. คริสตัล 11.0592 MHz	1	ตัว
9. 74LS04	1	ตัว
10. 74LS08	1	ตัว
11. MC7805	1	ตัว
12. Diode Bridge	1	ตัว

4.2.2.3 เหตุผลในการเลือกใช้อุปกรณ์ต่างๆ

ความต้านทานปรับค่าได้ : เนื่องจากราคาของเซนเซอร์ต่างๆ ที่จะใช้นั้น มีปัญหาต่างๆ กัน ไม่ว่าจะเป็นขนาดที่ใหญ่เกินไป การประกอบตัวเซนเซอร์ยากทำให้ไม่ทนทาน ราคาแพง ความซับซ้อนของวงจรขยายสัญญาณ หาซื้อในเมืองไทยไม่ได้ เป็นต้น ทำให้เราเลือกความต้านทานปรับค่าได้เพราะเมื่อเปรียบเทียบคุณภาพกับราคาแล้ว ถือว่าคุ้มกว่าซื้อของต่างประเทศเพราะมีความละเอียดเพียงพอในระดับหนึ่ง รวมถึงยังหาซื้อได้ง่ายและราคาถูก อีกทั้งง่ายต่อการบำรุงรักษาแม้จะเกิดปัญหาของความไม่แน่นอน ความต้านทานปรับค่าได้เมื่อใช้ไปนาน ๆ แต่ก็แทบจะไม่ผลต่อวงจรเพราะค่าเปลี่ยนไปไม่เยอะ

ADC0817 : เนื่องจาก ADC0817 นี้เป็นไอซีที่รวมเอา Multiplexer 16 Channels กับ ADC ขนาด 8 บิต เข้าไว้ด้วยกัน ทำให้เราสามารถลดความยุ่งยากของตัววงจรเราได้ และทำให้มีความทนทานมากกว่าด้วยเพราะวงจรง่าย ไม่ใหญ่ ไม่มีสายไฟโยงระเะระกะ โอกาสผิดพลาดมีน้อยกว่า

89C51 : เนื่องจากเป็นไมโครคอนโทรลเลอร์ที่มีจำนวนพอร์ตเหมาะสมกับวงจรของเรามากที่สุด รวมถึงมีความแพร่หลายในการใช้งานและหาได้ง่ายอีกด้วย และยังมีสนับสนุน UART ในตัวอีกด้วย ทำให้เราไม่ต้องกังวลเรื่องโปรโตคอลของพอร์ตอนุกรมในเลขอร์ล่างๆ อีกด้วย

MAX232 : เป็นอุปกรณ์ที่ใช้ติดต่อระหว่างไมโครคอนโทรลเลอร์กับพอร์ตอนุกรม ซึ่งไอซีตัวนี้สนับสนุนมาตรฐาน RS232 ซึ่งเป็นตัวกำหนดระดับแรงดันไฟฟ้า และสัญญาณของขาต่างๆ ของพอร์ตอนุกรมนี้ ทำให้เหมาะสมกับงานนี้

4.2.2.4 รายละเอียดการต่อวงจร

4.2.2.4.1 การต่อ ADC0817 เข้ากับเซนเซอร์(ความต้านทานปรับค่าได้)

ADC017 ถูกออกแบบเป็นคาค้าแควิซชันซิสเต็ม (Data Acquisition System) ที่สมบูรณ์สำหรับการแปลงสัญญาณตามอัตราส่วน(Ratiometric Conversion) ซึ่งในการแปลงสัญญาณตามอัตราส่วนการเปลี่ยนแปลงทางกายภาพจะถูกวัดและแสดงค่าโดยคิดเป็นเปอร์เซ็นต์ของค่าที่มากที่สุด (Full scale) ซึ่งไม่จำเป็นต้องเกี่ยวข้องกับค่าสัมบูรณ์มาตรฐาน (absolute standard) สำหรับค่าแรงดันไฟฟ้าของอินพุตของ ADC0817 แสดงโดยสมการดังนี้

$$\frac{V_{IN}}{V_{fs} - V_Z} = \frac{D_X}{D_{MAX} - D_{MIN}}$$

V_{IN} = Input voltage into the ADC0816

V_{fs} = Full-scale voltage

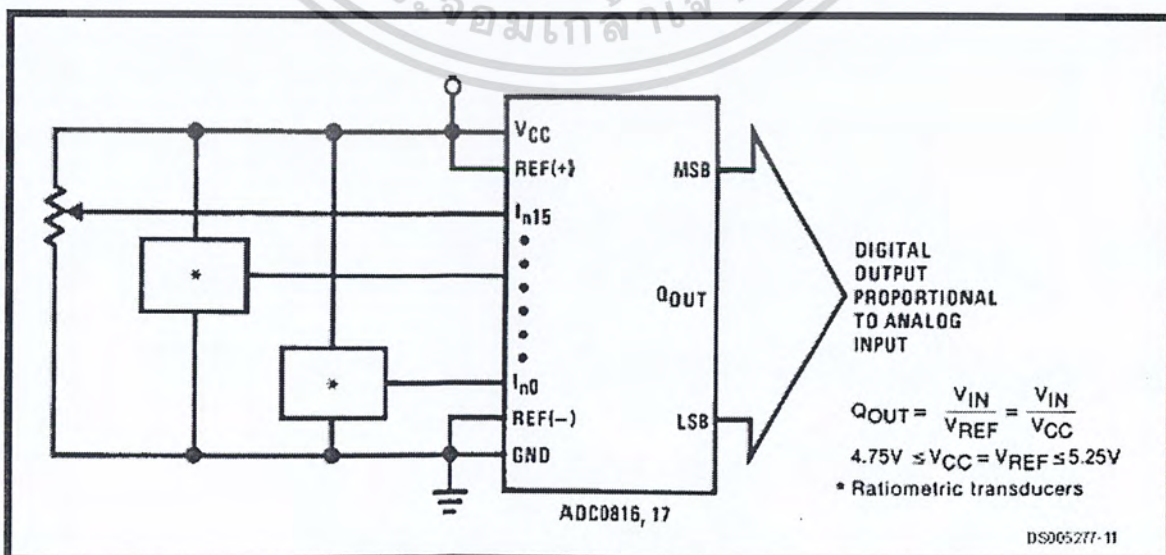
V_Z = Zero voltage

D_X = Data point being measured

D_{MAX} = Maximum data limit

D_{MIN} = Minimum data limit

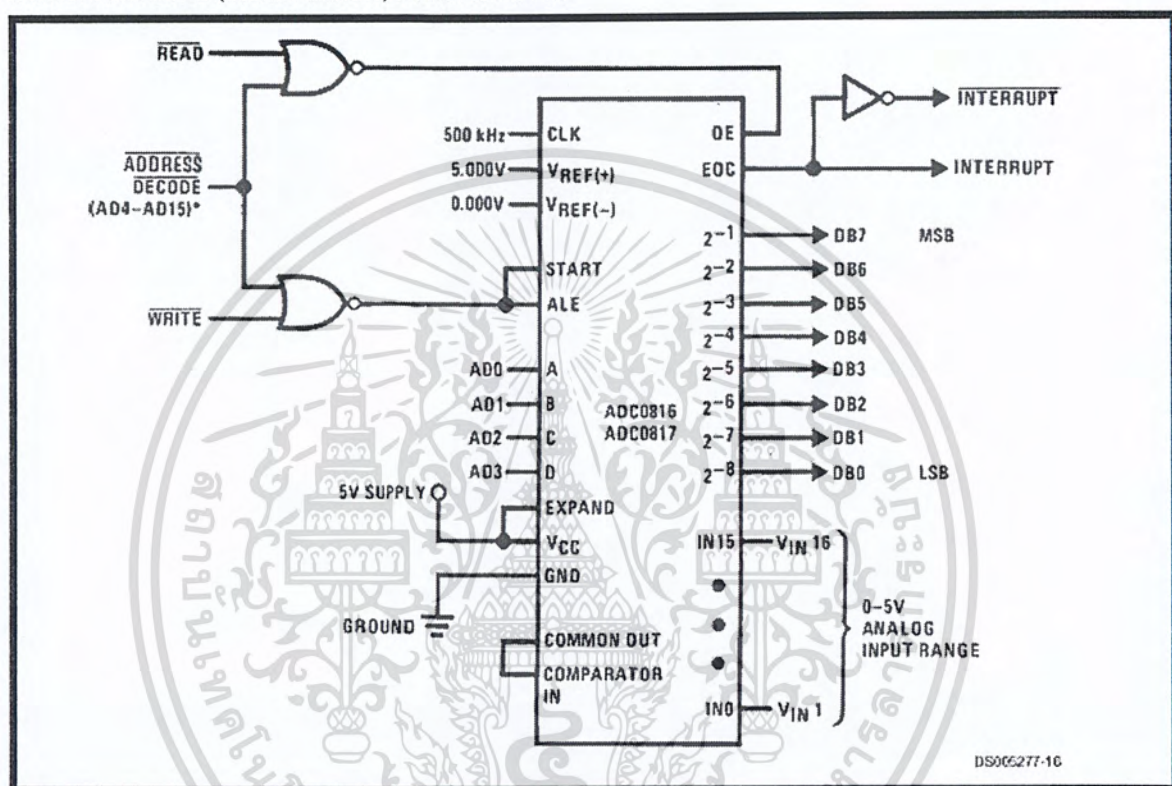
ซึ่งการต่อกับ R ปรับค่าได้นั้นเมื่อแกนหมุนของความต้านทานปรับค่าได้เปลี่ยนแปลงตำแหน่งไป ADC0817 จะทำการวัดแรงดันไฟฟ้าที่ได้เป็นสัดส่วนกับแรงดันไฟฟ้าที่ตกตลอดตัวมันทั้งหมดซึ่ง ADC0817 มีข้อดีคือ ระยะเวลาที่อินพุตเท่ากับระยะเวลาแรงดันที่จ่ายให้กับตัว ADC0817 ดังนั้นจึงสามารถต่อแรงดันที่จ่ายให้ตัว ADC0817 คล่อมความต้านทานปรับค่าได้และต่อขาที่ปรับค่าได้เป็นอินพุตให้กับ ADC0817 ได้เลยดังรูปที่ 4.8



รูปที่ 4.8 Ratiometric Conversion System

4.2.2.4.2 การใช้งาน ADC0817

สำหรับการนำ ADC0817 เพื่อนำไปใช้งานจะต้องคล้ายดังรูปที่ 4.9 ซึ่งขา COMPARATOR IN จะต่อเข้ากับ COMPARATOR OUT ซึ่งเป็นการนำค่าสัญญาณ analog ที่ได้จาก multiplexer เข้ามาเป็นอินพุตเพื่อแปลงเป็นสัญญาณดิจิทัลออกเลย แต่ส่วนที่ต่อเข้ากับ START, ALE, OE จะเปลี่ยนแปลงไปตามการดีโคดแอดเดรส (decode address) เพื่อนำไปใช้งาน

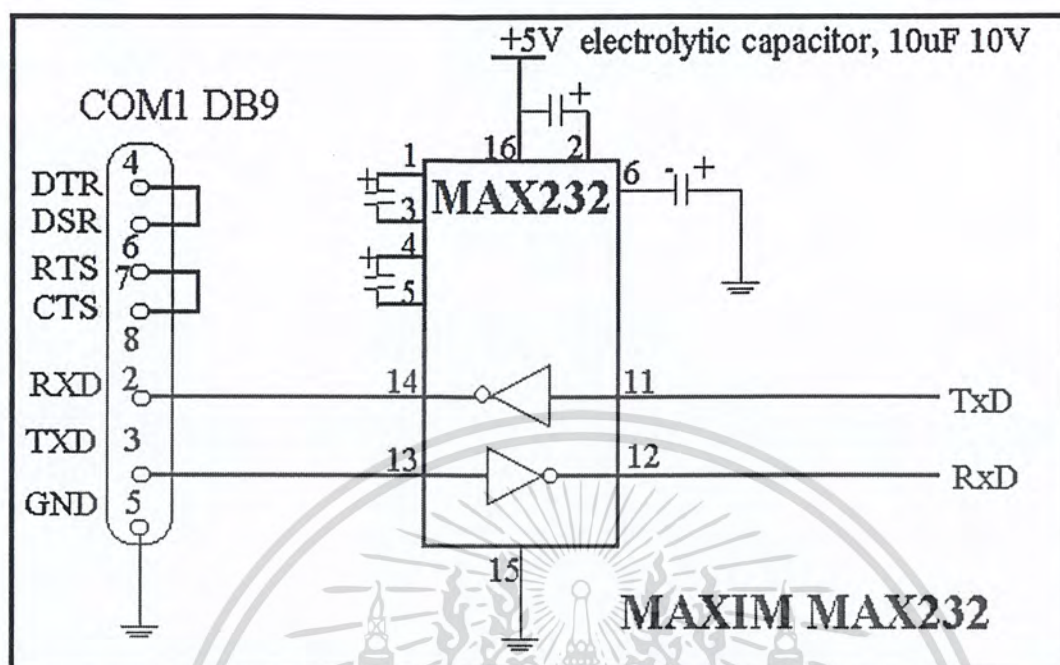


รูปที่ 4.9 การใช้งาน ADC0817

4.2.2.4.3 การใช้งาน MAX232

ซึ่งจากรูป 8.2 เราเพียงแค่อัปเดตค่าตัวเก็บประจุ 1 μF ให้กับไอซีตัวนี้ และต่อไฟเลี้ยงกับกราวด์ให้ ไอซีก็สามารถใช้งานได้เลยดังรูปที่ 4.10

จากรูปที่ 4.10 เราเพียงแค่นำขา TXD ของไอซี MAX 232 นี้ไปต่อเข้ากับขา RXD ของ 89C51 และนำขา RXD ของไอซี MAX232 ไปต่อเข้ากับขา TXD ของ 89C51 จากนั้นเราก็สามารถเขียนโปรแกรม 89C51 เพื่อทดสอบการใช้งานการรับส่งข้อมูลผ่านพอร์ตอนุกรมนี้ได้เลย



รูปที่ 4.10 การต่อ MAX232 ในการทำงานจริง

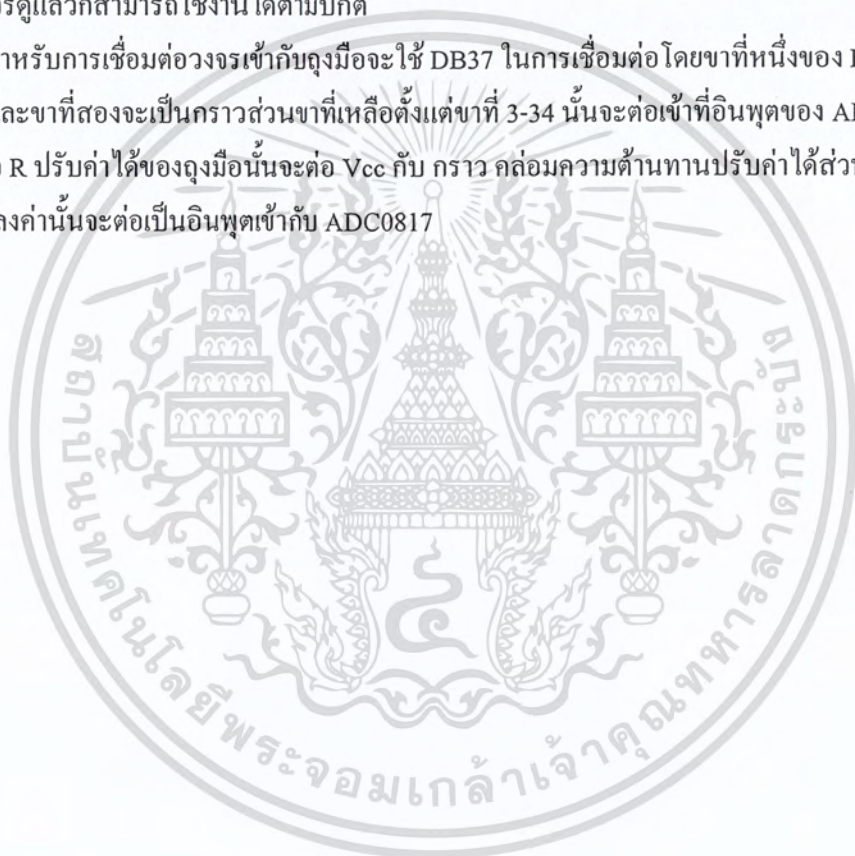
4.2.2.4.4 วงจรโดยรวมและการทำงานของวงจรโดยรวม

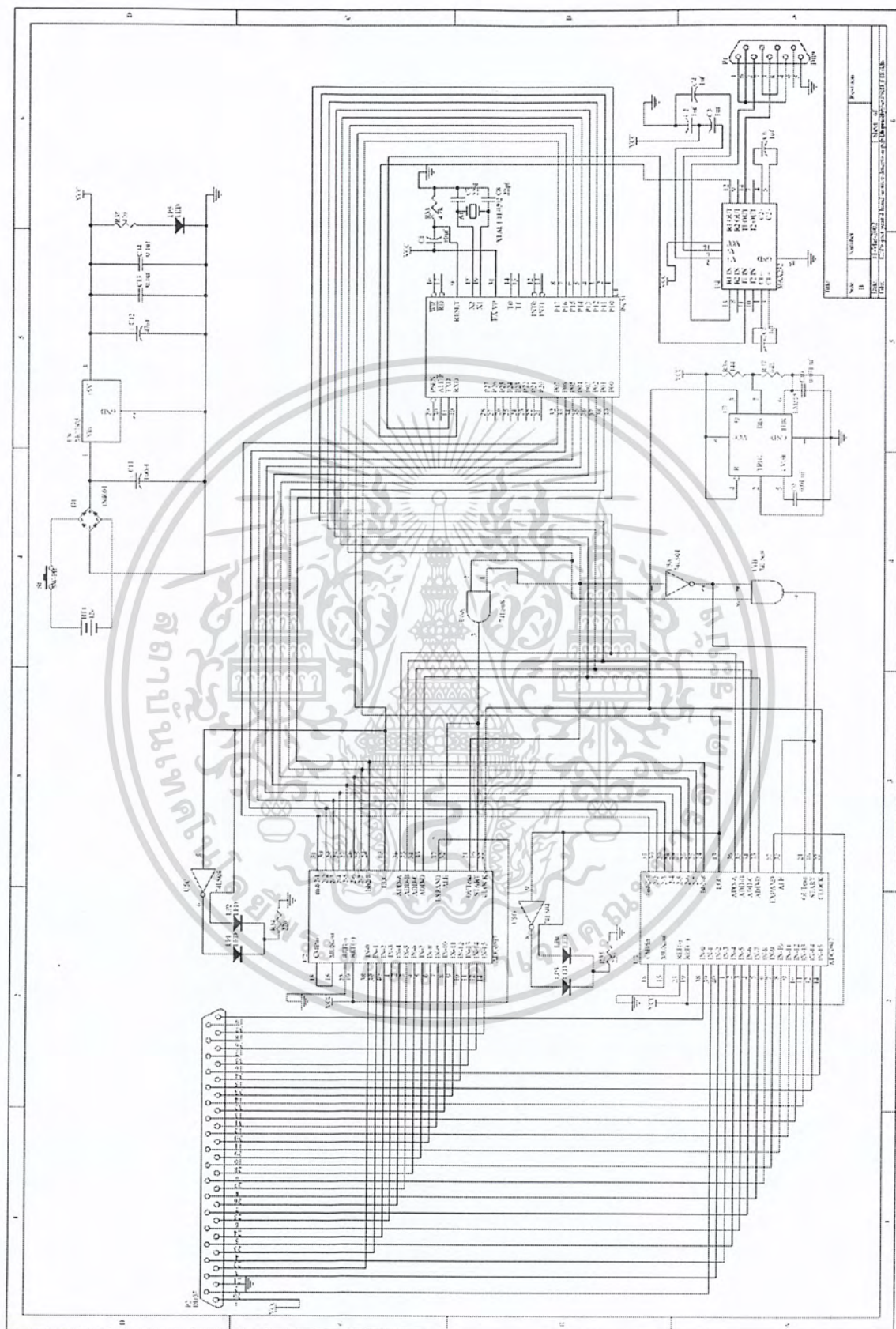
สเกมิติก (Schematic) ของวงจรแสดงดังรูปที่ 4.11 โดยเราจะใช้ ADC0817 จำนวน 2 ตัวต่อเข้ากับ 89c51 เนื่องจาก ADC0817 แต่ละตัวจะรับอินพุตได้แค่ 16 ตัวเท่านั้นแต่เราต้องใช้ถึง 22 ตัว จากนั้น 89c51 จะทำการควบคุมการอ่านและส่งข้อมูลโดยการอ่านข้อมูลนั้นจะสแกนวนลูปอ่านข้อมูลแต่ละแอดเดรสไปเรื่อย ๆ โดยใช้พอร์ต 0 (ขาที่ 32-39) รับข้อมูลจำนวน 8 บิตมา ส่วนบิตที่ 0-5 ของพอร์ตที่ 1 เป็นตัวคิโคตแอดเดรสซึ่ง 4 บิตแรก (บิตที่ 0-3) จะเป็นตัวส่งเข้าที่ขา ADD-A ถึง ADD-D ของ ADC0817 ทั้งสองตัว เพื่อเลือกว่าจะอ่านแอดเดรสไหนส่วนบิตที่ 4 และ 5 นั้นจะเป็นตัวเลือกว่าจะใช้ ADC0817 ตัวไหนโดยบิตที่ 4 (ขาที่ 5) จะต่อเข้ากับขา OUTena ของ ADC0817 โดยตัวแรกจะต่อเข้าขา OUTena ส่วนตัวที่สองต่อผ่าน นอตเกต (NOT gate) ก่อนแล้วจึงต่อเข้าขา OUTena ซึ่งจะทำหน้าที่ควบคุมว่าให้เอาต์พุตของ ADC0817 ตัวไหนแสดงผลออกมาส่วน ส่วนบิตที่ 5 (ขาที่ 6) นั้นจะใช้กับร่วมกับบิตที่ 4 โดยทั้งสองขาใช้เป็นอินพุตให้กับ แอนเกต (AND gate) และเอาต์พุตที่ได้ก็จะนำไปเป็นอินพุตของขา ALE เพื่อสั่งให้มันทำงานโดยบิตที่ 4 นั้นใน ADC0817 ตัวแรกจะใช้เป็นอินพุตให้กับแอนเกตได้เลยส่วนที่สองต้องผ่าน นอตเกตก่อนซึ่งก็หมายความว่าเมื่อบิตที่ 4 เป็น 1 จะเป็นตัวควบคุมให้ ADC0817 สามารถเอาต์พุตออกมาได้ในขณะที่ ADC0817 จะไม่สามารถส่งเอาต์พุตออกมาได้เพราะขา OUTena เป็น 0 เนื่องจากผ่าน นอตเกตทำให้เป็น 0 และเมื่อ บิตที่ 5 เป็น 1 ด้วยก็จะทำให้เอาต์พุตของ แอนเกตซึ่งต่อกับขา ALE เป็น 1 ทำให้ ADC0817 ตัวแรกทำงาน ในทางกลับกันเมื่อบิตที่ 4 เป็น 0 จะทำให้เอาต์พุตของ ADC0817 ตัวที่สอง

ออกมาและเมื่อบิตที่ 5 เป็น 1 ด้วยก็จะทำให้ ADC0817 ตัวที่สองทำงาน ส่วนบิตที่ 6 และ 7 นั้น จะทำหน้าที่เป็นตัวรับอินพุตจากขา EOC ของ ADC0817 แต่ละตัวซึ่งขานี้จะส่งสัญญาณ 1 ออกมาเมื่อทำงานเสร็จแล้ว เมื่อได้อินพุตจาก ADC0817 แล้ว 89C51 ก็จะทำข้อมูลที่ได้อัดไปยังคอมพิวเตอร์ผ่านพอร์ตอนุกรมโดยใช้ขา TXD เป็นตัวส่งและ RXD เป็นตัวรับ โดยผ่าน MAX232 ซึ่งได้อธิบายแล้วในบทที่ 8

นอกจากนี้ ADC0817 นั้นจะต้องใช้สัญญาณนาฬิกา ในการทำงานดังนั้นเราจึงใช้ 555 เป็นตัวกำเนิดสัญญาณนาฬิกาซึ่งตามปกติ ADC0817 จะใช้สัญญาณนาฬิกา ได้ตั้งแต่ 10 – 1280 kHz ซึ่งโดยทั่วไปจะใช้ 640 kHz แต่ 555 ตัวที่เราต้องการเพื่อให้ได้สัญญาณนาฬิกา 640 kHz ไม่มีในท้องตลาด (LM555) จึงใช้ 555 ที่มีอยู่ในท้องตลาดทั่วไปซึ่งสร้างสัญญาณนาฬิกาได้สูงสุดประมาณ 250 kHz ซึ่งได้ลองต่อวงจรดูแล้วก็สามารถใช้งานได้ตามปกติ

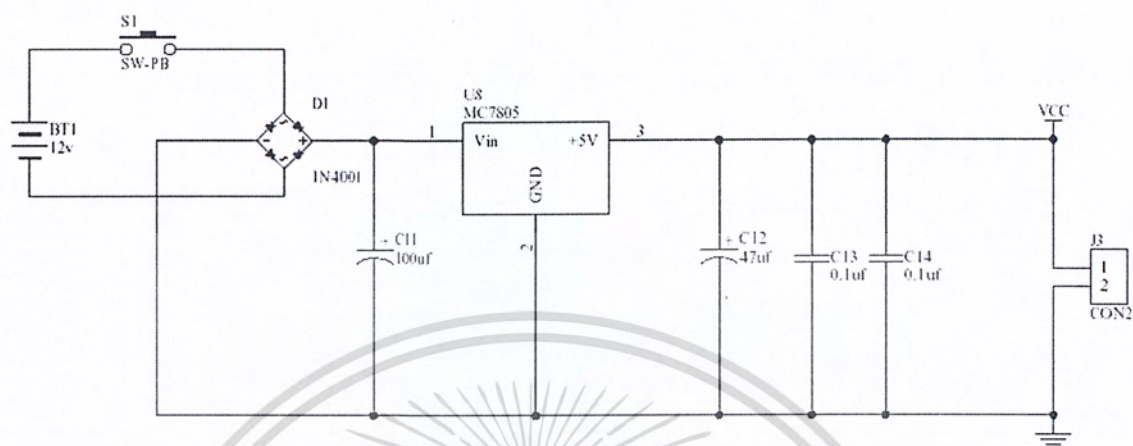
สำหรับการเชื่อมต่อวงจรเข้ากับถุมือจะใช้ DB37 ในการเชื่อมต่อโดยขาที่หนึ่งของ DB37 จะเป็น Vcc และขาที่สองจะเป็นกราวส่วนขาที่เหลือตั้งแต่ขาที่ 3-34 นั้นจะต่อเข้าที่อินพุตของ ADC0817 โดยการต่อ R ปรับค่าได้ของถุมือนั้นจะต่อ Vcc กับ กราว คล่อมความต้านทานปรับค่าได้ส่วนขาที่เปลี่ยนแปลงค่านั้นจะต่อเป็นอินพุตเข้ากับ ADC0817





รูปที่ 4.11 สถิติของวงจรโดยรวม

สำหรับส่วนของแหล่งจ่ายไฟนั้นจะใช้ ไดโอดบริจ (diode bridge) และ MC7805 กับตัวเก็บประจุอีกจำนวนหนึ่งในการแปลงแรงดันไฟฟ้าจาก 9 โวลต์ เป็น 5 โวลต์ ดังรูปที่ 4.12



รูปที่ 4.12 วงจรแหล่งจ่ายไฟ

สำหรับการรายละเอียดของลายแผ่นปริ้นนั้นดูได้จากภาคผนวก ข.

บทที่ 5

การออกแบบในส่วนของซอฟต์แวร์

หลังจากที่เราได้ออกแบบวงจรทุกอย่างเป็นที่เรียบร้อยแล้ว จากนั้นเราก็จะเข้าสู่กระบวนการทางฝั่งซอฟต์แวร์ ซึ่งประกอบด้วย 3 ส่วนคือ ส่วนของโปรแกรมควบคุม 89c51 ส่วนของโปรแกรมรับค่าจาก 89C51 และแสดงผลเป็นเทกซ์โฮมดบนคอมพิวเตอร์ และส่วนการแสดงผลเป็นกราฟฟิกโฮมดบนคอมพิวเตอร์

5.1 โปรแกรมควบคุม 89c51

โปรแกรมควบคุม 89c51 นั้นเป็นโปรแกรมที่ทำการส่งข้อมูลที่ได้รับจากตัวเซนเซอร์ ซึ่งไอซี 89C51 จะเป็นตัวจัดการกระบวนการการเลือกเซนเซอร์ที่จะส่ง ไปให้ผู้ใช้งานฝั่งคอมพิวเตอร์ได้ใช้งานได้

โดยการป้อนข้อมูลบันทึกลงไปไอซี 89C51 นั้น เราจะต้องทำการเขียนโปรแกรมโดยสามารถใช้โปรแกรมภาษาต่างๆ เช่น แอสเซมบลี ภาษาซี เป็นต้น โดยเราต้องมีการเลือกใช้คอมไพเลอร์ที่สนับสนุนกับการแปลงข้อมูลเพื่อจะได้ทำการบันทึกลงไอซีได้อีกด้วย

5.1.1 โครงสร้างการทำงานหลักๆ ของโปรแกรมที่จะทำการบันทึกลงไอซี 89C51

- ส่วนของการเลือกตัวเซนเซอร์ที่จะทำการรับเข้าตัว 89C51 ในแต่ละเวลาขณะหนึ่งๆ
- ส่วนของการส่งติดตั้งค่าสำหรับการติดต่อการส่งข้อมูลผ่านทางพอร์ตอนุกรม
- ส่วนของการส่งค่าที่ได้รับจากตัวเซนเซอร์แล้วทำการส่งบอกคอมพิวเตอร์ได้ว่าเป็นค่าจากเซนเซอร์ตัวไหน และมีค่าจากเซนเซอร์นั้นเป็นเท่าไร

จากนั้นเราก็จะทำการเลือกใช้คอมไพเลอร์ที่เหมาะสมและสะดวก ซึ่งในที่นี้โปรแกรมของเราค่อนข้างมีความเรียบง่าย เพราะว่ามีโครงสร้างของโปรแกรมที่ไม่ยุ่งยากซับซ้อนดังจะเห็นได้จากโครงสร้างตามหัวข้อ 5.1.1 นี้ ดังนั้นเราจึงเลือกใช้ภาษาแอสเซมบลีในการเขียน โดยคอมไพเลอร์ที่เราใช้นั้นมีชื่อว่า SXA51 ซึ่งเมื่อเราเขียนโปรแกรมในภาษาแอสเซมบลี (ซึ่งจะมีนามสกุลของไฟล์ว่า .A51) แล้ว SXA51 จะทำการแปลงไฟล์ .A51 ที่ได้ไปเป็นนามสกุล .HEX ถ้าไม่พบข้อผิดพลาดในการเขียนโปรแกรม .A51 นี้ ซึ่งหลังจากนั้น เราก็จะนำไฟล์ .HEX นี้ไปใช้ในการบันทึกข้อมูลลงไอซี 89C51 ด้วยเครื่องบันทึกอีกทีหนึ่ง

5.1.2 ขอสกัดของโปรแกรมตัวส่งข้อมูลบนไอซี 89C51

ORG 0000H

jmp Main

; ฟังก์ชัน delay เป็นฟังก์ชันสำหรับถ่วงเวลา เพื่อให้เวลาแก่ไอซี ADC0817

; ซึ่งต้องอาศัยเวลาในการทำงาน

delay: push 0

push 1

mov r0,#00h

loop: mov r1,#0F0h

djnz r1,\$

djnz r0,loop

pop 1

pop 0

ret

; ฟังก์ชัน Main เป็นส่วนของการติดตั้งค่าให้แก่ 89C51 เพื่อทำการบอกว่า

; จะติดต่อพอร์ตอนุกรมด้วยอัตราบอดเรต 9600 โดยมีสตீปบิต 1 บิต

Main: mov scon,#42h

mov a,pcon

clr acc.7

mov pcon,a

mov tmod,#20h

mov th1,#0FDh

setb tr1

mov a,#15h

; ฟังก์ชัน con เป็นส่วนของการวนลูปเพื่อทำการเลือกตัวเซนเซอร์

; และทำการส่งผลลัพธ์ที่ได้เข้าสู่พอร์ตอนุกรม

con: jnb ti,\$

```

clr    ti
mov    sbuf,#00h
mov    p1,a
mov    r0,a
djnz   r0,con2
mov    p1,#0h
setb   p1.5
clr    p1.5
call   delay
mov    p2,p0
jnb    ti,$
clr    ti
mov    sbuf,p0
jnb    ti,$
clr    ti
mov    sbuf,#00h
mov    p1,#15h
mov    r0,#15h
mov    a,#15h
con2:  setb   p1.5
clr    p1.5
call   delay
mov    p2,p0
jnb    ti,$
clr    ti
mov    sbuf,p0

jmp    con
end

```

5.2 โปรแกรมรับค่าจาก 89C51 และแสดงผลเป็นเทกซ์โฮมดบนคอมพิวเตอร์

5.2.1 การทดสอบการรับค่าจาก 89c51 บนคอมพิวเตอร์

เมื่อเราทำการเขียนโปรแกรมและบันทึกลงไอซี 89C51 เป็นที่เรียบร้อยแล้ว และหลังจากที่เราได้นำไอซี 89C51 ไปใส่ในวงจรของเราและทำการต่อสายอนุกรมเชื่อมเข้ากับคอมพิวเตอร์เป็นที่เรียบร้อยแล้ว เราก็จะทำการทดสอบว่าคอมพิวเตอร์ได้รับค่าตามที่เราส่งหรือไม่ โดยในที่นี้ เราใช้โปรแกรมบนคอมพิวเตอร์ที่ชื่อว่า Pcpplus ซึ่งเป็นโปรแกรมที่ใช้ทดสอบว่ามีค่าวิ่งเข้าจากคอมพิวเตอร์ใดๆ หรือไม่ โดยหลังจากเราได้มีการกำหนดค่าให้แก่โปรแกรม Pcpplus นี้ว่าจะให้ดูค่าจากพอร์ตไหนด้วยอัตราบอร์ตเรตเท่าไรแล้ว ถ้ามีค่าวิ่งเข้ามา โปรแกรมจะแสดงค่าทุกค่าที่วิ่งเข้ามาให้เราได้เห็น ซึ่งเราสามารถดูค่าบนโปรแกรมนี้ได้ ว่ามีค่าตามที่เรต้องการหรือไม่

5.2.2 โปรแกรมการรับข้อมูลบนคอมพิวเตอร์

หลังจากที่เราได้ทดสอบว่ามีค่าวิ่งเข้าคอมพิวเตอร์ถูกต้องตามที่เรต้องการแล้ว เราก็มารับเขียนส่วนของโปรแกรมที่จะทำการรับข้อมูลขึ้นมาเอง โดยเราสามารถเลือกใช้ภาษาได้หลายภาษาบนคอมพิวเตอร์เพื่อที่จะทำการรับค่าผ่านทางพอร์ตอนุกรม ในที่นี้ เราได้เลือกใช้ภาษาวิซวลเบสิกในการเขียนโปรแกรมตรงส่วนนี้เพราะความง่ายในการเขียนโปรแกรมและความสะดวกในการทำอินเตอร์เฟสอีกด้วย

5.2.2.1 อินเตอร์เฟสบนวิซวลเบสิก

Glove base Input							
01:		11:		21:			
02:		12:		22:			
03:		13:					
04:		14:					
05:		15:					
06:		16:					
07:		17:					
08:		18:					
09:		19:					
10:		20:					

รูปที่ 5.1 อินเตอร์เฟสหน้าจอโปรแกรมการรับค่าจากไอซี 89C51

ในการทำอินเตอร์เฟสหน้าจอนี้ ต้องมีการเรียกใช้คอมโพเนนต์เพิ่มขึ้นมาอีกหนึ่งตัว คือคอมโพเนนต์พอร์ตอนุกรม ซึ่งวิธีเพิ่มคอมโพเนนต์นี้เข้ามาในโปรเจกต์เรานั้น ทำดังนี้

- กดเมาส์ขวามือแถบทูลบ็อกซ์ทางด้านซ้ายมือ จากนั้นเลือก Components...
- ในแท็บคอนโทรล ให้เลื่อนลงมาข้างล่างเรื่อยๆ จนพบคำว่า Microsoft Comm Control 6.0 จากนั้นให้คลิกเลือก
- กดปุ่ม OK

จากนั้นในแถบทูลบ็อกซ์ทางด้านซ้ายมือจะมีคอมโพเนนต์ตัวใหม่ขึ้นมาแล้วให้เราลากมาแปะบนฟอร์มของเรา

5.2.2.2 ข้อสั้ดของโปรแกรมบนวิซวลเบสิก

Option Explicit

Private check As Integer

Private mode As Boolean

Private buffer(0 To 3) As Variant

Private num(0 To 3) As Integer

‘ ฟังก์ชัน Form_Load() เป็นการติดตั้งค่าเริ่มต้นให้กับฟอร์ตออนุกรมว่า

‘ ให้มีค่าบอดเรต 9600 และมีสต่อบิต 1 บิต

Private Sub Form_Load()

MSComm1.Settings = "9600,n,8,1"

MSComm1.CommPort = 1

MSComm1.InputLen = 1

MSComm1.PortOpen = True

MSComm1.RThreshold = 1

check = 0

mode = True

End Sub

‘ ฟังก์ชัน MSComm1_OnComm() จะทำงานเมื่อมีค่าข้อมูลวิ่งเข้ามาผ่านทางฟอร์ตออนุกรม

‘ ซึ่งฟังก์ชันนี้จะแบ่งเป็น 2 โหมด คือ โหมดการชิงโครนัสค่า กับโหมดการเอาค่าไปใช้จริง

Private Sub MSComm1_OnComm()

On Error GoTo Com_Error

If mode Then

```

If check < 4 Then
    buffer(check) = MSComm1.Input
    num(check) = Asc(buffer(check))
    If check = 3 Then
        If num(2) = num(0) + 1 Then mode = False
    End If
Else
    buffer(0) = MSComm1.Input
    mode = False
End If
Else
    buffer(0) = MSComm1.Input
    If check > 1 Then
        num(0) = Asc(buffer(0))
        check = 0
    Else
        Before(num(0) + 1) = Asc(buffer(0))
        Text1(num(0)).Text = Asc(buffer(0))
    End If
End If
End If
check = check + 1
Exit Sub
Com_Error:
check = 0
mode = True
Exit Sub
End Sub

```

เหตุที่ต้องมีโหมดการชิง โครนส์คำก็เป็นเพราะว่า โพรโตคอลที่เราได้เขียนไว้ในการรับข้อมูลคือ ค่าหมายเลขของเซนเซอร์ (ซึ่งจะมีค่าระหว่าง 0 ถึง 21) กับค่าภายในเซนเซอร์นั้น (ซึ่งจะมีค่าระหว่าง 0 ถึง 255) ดังนั้น ครั้งแรกสุดเมื่อเรารันโปรแกรมบนวิซวลเบสิกนั้น ถ้าเราได้รับค่าตั้งแต่ 22 ถึง 255 เราก็จะทราบว่า นี่คืค่าภายในเซนเซอร์ ค่าต่อไปที่เราจะได้รับคือค่า หมายเลขของเซนเซอร์ตัวถัดไป แต่ว่าถ้าเราได้รับค่าครั้งแรกสุดเป็น 0 ถึง 21 เราไม่สามารถรู้ได้เลยว่า 0 ถึง 21 นี่เป็นค่าหมายเลขของเซนเซอร์ หรือค่า

ภายในเซเชอร์ ซึ่งการทำจึงโครนัสจะอาศัยหลักการที่ว่า ถ้าเราได้ค่า 0 ถึง 21 มา (สมมติว่าเก็บไว้ในตัวแปรชื่อ a) ให้เราลดค่าถัดไปอีก 2 ค่า (สมมติว่าเก็บไว้ในตัวแปรชื่อ b) ว่ามีค่ามากกว่าครั้งแรกสุดอยู่ 1 หรือไม่ ($b = a + 1$) หรือไม่ ถ้าใช่แสดงว่าค่าครั้งแรกสุดที่เราได้รับนั้น เป็นค่าหมายเลขของเซเชอร์นั่นเอง แต่เราต้องทำเผื่อกรณีที่ค่าภายในเซเชอร์มีค่าที่เข้ากับหลักการนี้ ($b = a + 1$) ด้วยการใส่ว่าถ้าค่าผิดพลาด ให้ทำการเข้าสู่โหมดจึงโครนัสใหม่

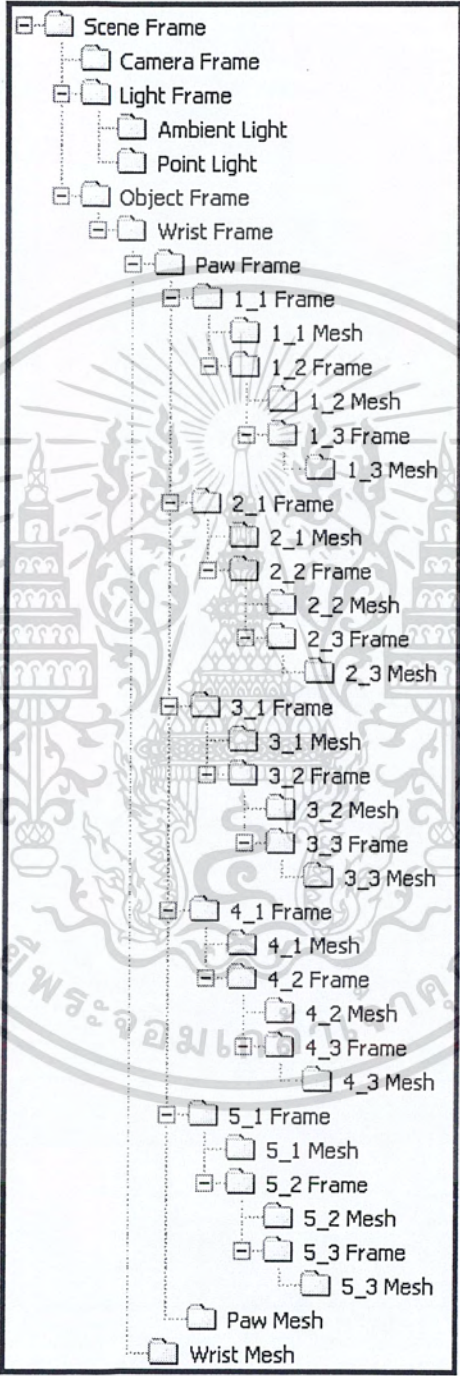
5.3 ส่วนการแสดงผลเป็นกราฟฟิคโหมดบนคอมพิวเตอร์

เมื่อเราได้ค่าที่ได้จากการติดต่อกับตัวสูงมือที่เราสร้างขึ้นมาแล้วนั้น การแสดงค่าเป็นตัวเลข อาจจะทำให้ผู้ใช้เข้าใจต่อผลลัพธ์ที่แสดงออกมาได้ยาก เราจึงต้องการจะนำเสนอการแสดงผลแบบ 3 มิติ

ในปัจจุบันนี้ เรามีทุลที่ช่วยสร้าง (สามมิติสตูดิโอแมกซ์, โลท์เวฟ, มายา, ทรูสเปซ เป็นต้น) และเอ็นจินต์ที่ใช้ในการช่วยแสดงผลแบบ 3 มิติ (ชีดเธิกซ์, มอร์ฟิต เป็นต้น) อยู่มากมาย แต่ในที่นี้ เราจะใช้ชุดฟังก์ชันที่วินโดว์มีให้อยู่แล้วในการแสดงผล 3 มิติ นั่นคือ ไคเรกซ์เธิกซ์

โดยโปรแกรมของเรามีขึ้นกราฟดังรูป 5.2

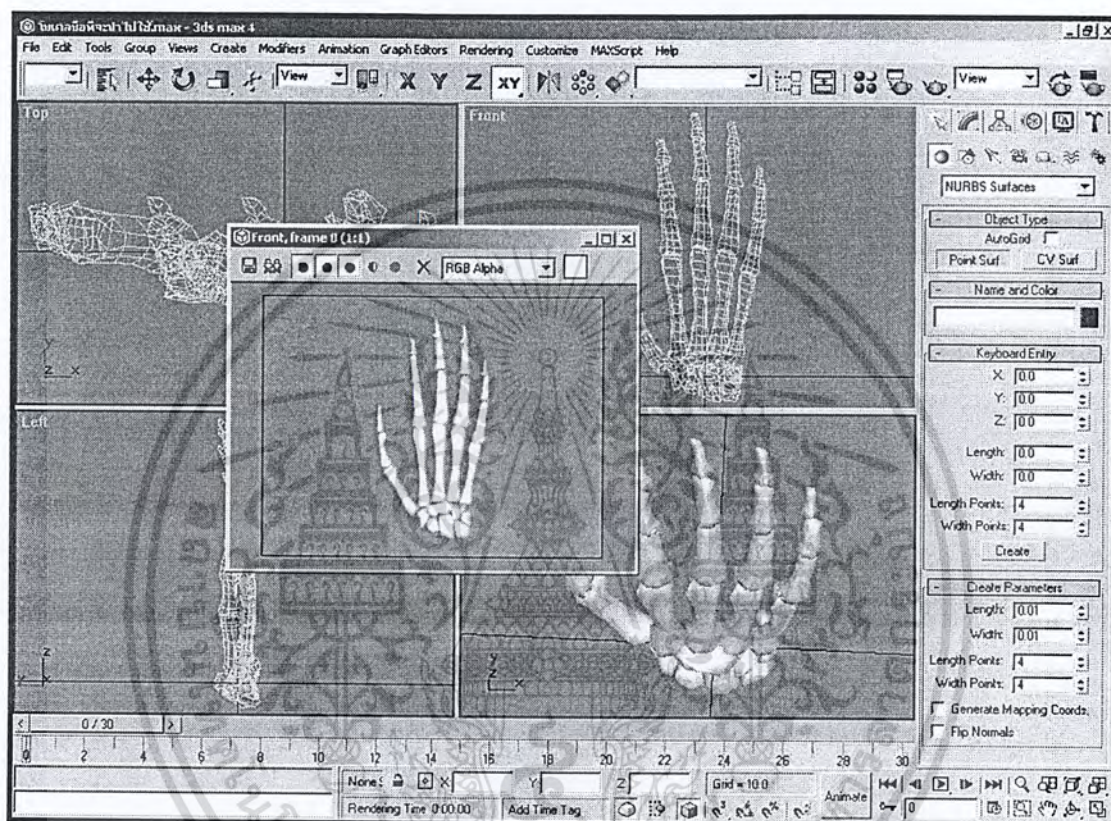




รูปที่ 5.2 ซีนกราฟของโปรแกรมในมุมมองแบบทรีวิว

5.3.1 ไฟล์มาตรฐานอุตสาหกรรมเมฆ (.3ds)

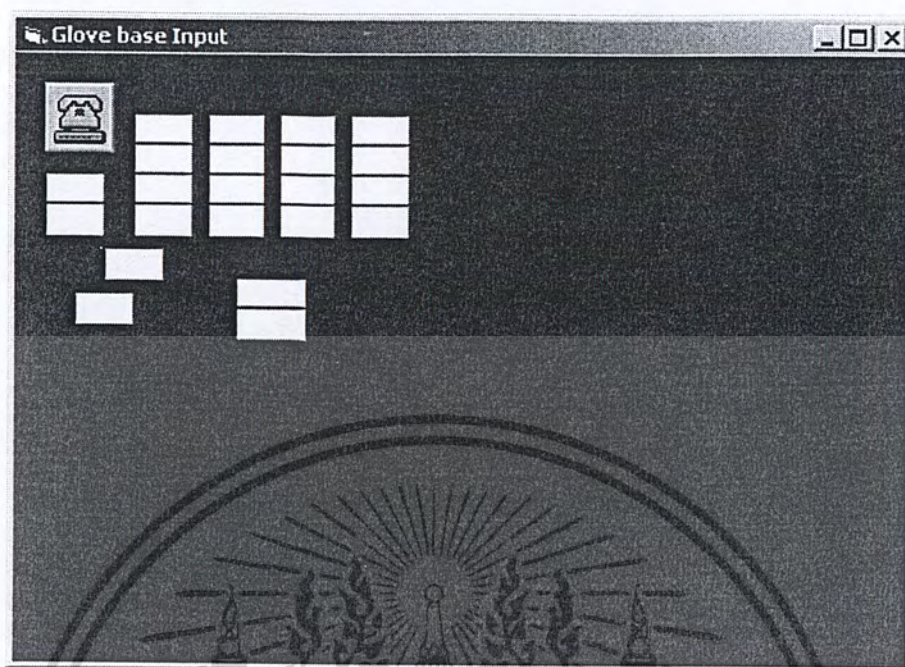
ในที่นี้เราจะไม่กล่าวถึงวิธีการสร้างไฟล์มาตรฐานอุตสาหกรรมเมฆนี้ เพราะเราสามารถหาดาวน์โหลดได้ทั่วไปในอินเทอร์เน็ตและจะได้แสดงภาพตัวอย่างของมือที่เราใช้งานในโปรแกรมวิชวลเบสิก ซึ่งมีหน้าตาดังรูปที่ 5.3



รูปที่ 5.3 ภาพไฟล์มาตรฐานอุตสาหกรรมเมฆที่เราใช้ในโครงการนี้

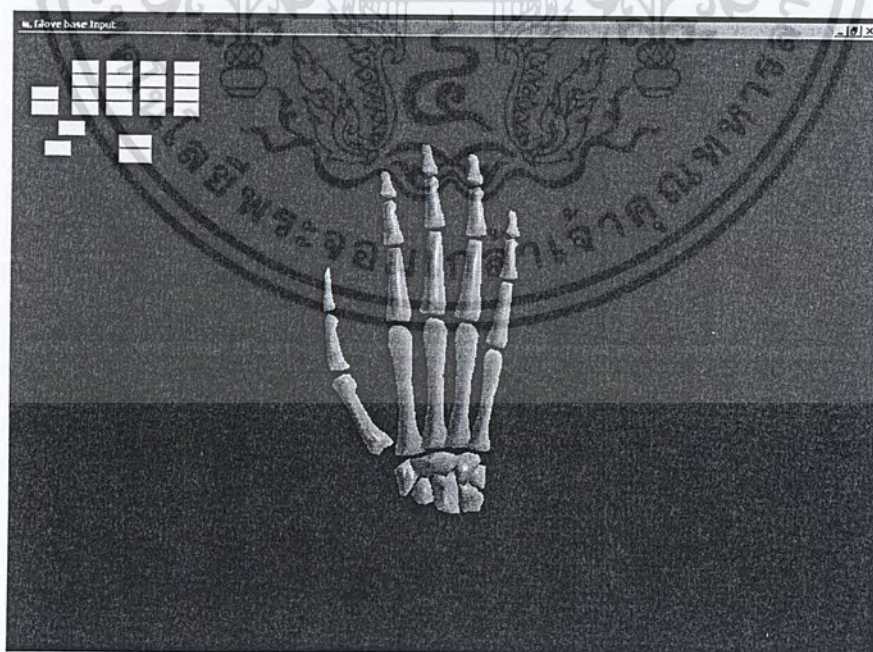
5.3.2 การทำงานของโปรแกรม

เมื่อรันโปรแกรมแล้วเมื่อผู้ใช้ขยับมือแล้วฟอร์มจะทำให้มือที่เราโหลดไว้ในโปรแกรมเคลื่อนไหวตาม เราสามารถใช้วิชวลเบสิกทำงานตามที่กล่าวมานี้ได้เพราะวิชวลเบสิกสามารถดักจับอีเวนต์ของฟอร์มคอนทROLS ได้อยู่แล้ว ซึ่งหน้าจอของฟอร์มบนวิชวลเบสิกที่เราสร้างขึ้นมานี้เราไม่ต้องลากคอนโทรลใดเข้ามาใช้เลย นอกจากส่วนของกริดติดกับฟอร์มคอนทROLS เท่านั้นดังรูปที่ 5.4



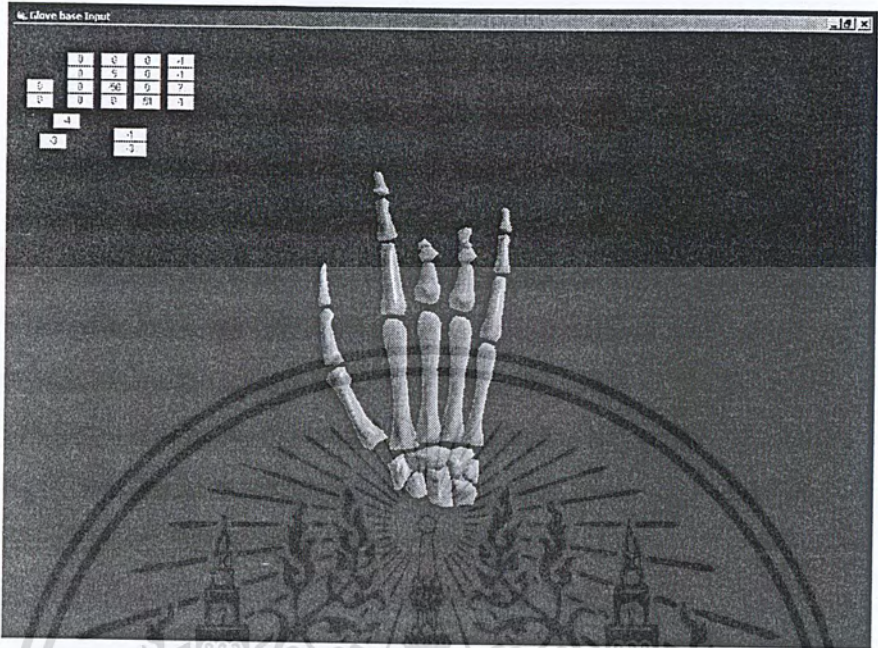
รูปที่ 5.4 ภาพของฟอร์มของโปรแกรมบนวิซวลเบสิก

และในภาพที่ 5.5 นี้จะแสดงผลของการโหลดวัตถุสามมิติจากไฟล์เมช ไคเร็กซ์เอ็กซ์โดยใช้วิซวลเบสิกบนฟอร์มนี้



รูปที่ 5.5 ภาพการใช้วิซวลเบสิกโหลดวัตถุสามมิติจากไฟล์เมช ไคเร็กซ์เอ็กซ์

และรูปที่ 5.6 แสดงผลลัพธ์ของการทำงานของโปรแกรมเมื่อนำตัวถูงมือที่เราได้สร้างขึ้นมาบังคับมือที่เราโหลดขึ้นมาบนฟอร์มนี้



รูปที่ 5.6 ภาพผลลัพธ์ของการผนวกอุปกรณ์อินพุตแบบถูงมือกับไคเรกซ์เอ็กซ์

5.3.3 ข้อโค้ดของโปรแกรมแสดงผลไคเรกซ์สามดี

Option Explicit
Option Base 1

Const Sin1 = 0.0174524!
Const Cos1 = 0.9998476!

Private check As Integer
Private mode As Boolean
Private buffer(0 To 3) As Variant
Private num(0 To 3) As Integer

Dim pD3drm As IDirect3DRM3
Dim Scene As IDirect3DRMFrame3
Dim Camera As IDirect3DRMFrame3
Dim pClipper As IDirectDrawClipper

```

Dim Dev                As IDirect3DRMDevice3
Dim pViewport          As IDirect3DRMViewPort
Dim l1                 As IDirect3DRMLight
Dim l2                 As IDirect3DRMLight
Dim Lights             As IDirect3DRMFrame3
Dim Hand_Builder(17)  As IDirect3DRMMeshBuilder3
Dim hand_mesh(17)     As IDirect3DRMMesh
Dim Hand(17)           As IDirect3DRMFrame3
Dim Filename(17)      As String
Dim Save_Before        As Boolean
Dim Before(22)         As Integer
Dim Map_File_Index(22) As Integer
Dim Delay              As Integer

```

‘ ฟังก์ชัน Display(Index, Value) เป็นฟังก์ชันที่ทำการเคลื่อนวัตถุสามมิติตามค่าที่ได้รับจากพอร์ตอนุกรม

```
Private Sub Display(Index As Integer, Value As Integer)
```

```
Dim Temp As Integer
```

```
Dim i As Integer
```

```
Dim j As Integer
```

```
Temp = Before(Index) - Value
```

```
Text1(Index - 1).Text = Temp
```

```
j = Map_File_Index(Index)
```

‘ ทำการ map ค่าการเคลื่อนไหวที่กำหนดใน Form_Load ซึ่งจะมี 22

‘ ค่าคือ 17 ข้อ แต่ข้อแรกของแต่ละนิ้วเคลื่อนไหวได้สองทิศทางเลข

‘ แยกเพิ่มได้อีก 5 ค่า

```
Select Case Index
```

```
Case 22
```

```
Delay = Delay + 1
```

```
If Delay = 2 Then
```

```
UpdateScreen
```

```
Delay = 0
```

```
End If
```

Case 1, 4, 5, 6, 11, 13, 14, 20, 21

If Temp $\neq$ 0 Then ‘ ถ้าไม่มีการหมุนเลยคือมีค่าศูนย์แปลว่าเราไม่ต้องหมุนวัตถุเลย

If Temp > 0 Then

For i = 1 To Temp

Hand(j).SetOrientation Hand(j), 0, Sin1, Cos1, 0, 1, 0

Next i

Else

For i = 1 To Abs(Temp)

Hand(j).SetOrientation Hand(j), 0, -Sin1, Cos1, 0, 1, 0

Next i

End If

End If

Before(Index) = Value

Case 2, 7, 9, 10, 16, 17, 18

If Temp $\neq$ 0 Then

If Temp > 0 Then

For i = 1 To Temp

Hand(j).SetOrientation Hand(j), 0, -Sin1, Cos1, 0, 1, 0

Next i

Else

For i = 1 To Abs(Temp)

Hand(j).SetOrientation Hand(j), 0, Sin1, Cos1, 0, 1, 0

Next i

End If

End If

Before(Index) = Value

Case 3, 12

If Temp $\neq$ 0 Then

If Temp > 0 Then

For i = 1 To Temp

Hand(j).SetOrientation Hand(j), 1, 0, 0, 0, 1, 0

Hand(j).SetOrientation Hand(j), 0, Sin1, Cos1, 0, 1, 0

Hand(j).SetOrientation Hand(j), -1, 0, 0, 0, 1, 0

Next i

```

Else
    For i = 1 To Abs(Temp)
        Hand(j).SetOrientation Hand(j), 1, 0, 0, 0, 1, 0
        Hand(j).SetOrientation Hand(j), 0, -Sin1, Cos1, 0, 1, 0
        Hand(j).SetOrientation Hand(j), -1, 0, 0, 0, 1, 0
    Next i
End If
End If
Before(Index) = Value
Case 8, 15, 19
If Temp <> 0 Then
    If Temp > 0 Then
        For i = 1 To Temp
            Hand(j).SetOrientation Hand(j), 1, 0, 0, 0, 1, 0
            Hand(j).SetOrientation Hand(j), 0, -Sin1, Cos1, 0, 1, 0
            Hand(j).SetOrientation Hand(j), -1, 0, 0, 0, 1, 0
        Next i
    Else
        For i = 1 To Abs(Temp)
            Hand(j).SetOrientation Hand(j), 1, 0, 0, 0, 1, 0
            Hand(j).SetOrientation Hand(j), 0, Sin1, Cos1, 0, 1, 0
            Hand(j).SetOrientation Hand(j), -1, 0, 0, 0, 1, 0
        Next i
    End If
End If
End Select
Before(Index) = Value
End Sub

```

‘ ฟังก์ชัน Form_KeyDown(KeyCode, Shift) ไว้ทำการเคลื่อนย้ายกล่องตามคีย์บอร์ดที่กด

```
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
Select Case KeyCode
```

‘ กด Enter เป็นการเปลี่ยนโหมดจากคาลิเบสโหมดเป็นโหมดเรนเดอร์

Case 13

Save_Before = False

‘ปุ่ม 2, 4, 6, 8 บนที่ NumPad ไว้เปลี่ยนตำแหน่งที่ตั้งกล้อง

Case 104

Camera.SetOrientation Camera, 0, -Sin1, Cos1, 0, Cos1, 0

Case 98

Camera.SetOrientation Camera, 0, Sin1, Cos1, 0, Cos1, 0

Case 102

Camera.SetOrientation Camera, -Sin1, 0, Cos1, 0, 1, 0

Case 100

Camera.SetOrientation Camera, Sin1, 0, Cos1, 0, 1, 0

Case 101 ‘ปุ่ม 5 บนที่ NumPad ไว้รีเซ็ตตำแหน่งที่ตั้งกล้อง

Camera.SetOrientation Scene, 0, 0, 1, 0, 1, 0

End Select

Call UpdateScreen

End Sub

‘ ฟังก์ชัน Form_Load() ทำหน้าที่กำหนดค่าต่างๆ ก่อนเข้าสู่การวนloopทำงาน

Private Sub Form_Load()

Dim a As Integer

‘ Filename(1 to 17) เป็นชื่อส่วนประกอบต่างๆ บนมือ

Filename(1) = "5_3" ‘ ชื่อปลายนิ้วก้อย

Filename(2) = "4_3" ‘ ชื่อปลายนิ้วนาง

Filename(3) = "3_3" ‘ ชื่อปลายนิ้วกลาง

Filename(4) = "2_3" ‘ ชื่อปลายนิ้วชี้

Filename(5) = "1_3" ‘ ชื่อปลายนิ้วโป้ง

Filename(6) = "5_2" ‘ ชื่อกลางนิ้วก้อย

Filename(7) = "4_2" ‘ ชื่อกลางนิ้วนาง

Filename(8) = "3_2" ‘ ชื่อกลางนิ้วกลาง

Filename(9) = "2_2" ‘ ชื่อกลางนิ้วชี้

Filename(10) = "1_2" ‘ ชื่อกลางนิ้วโป้ง

Filename(11) = "5_1" ‘ ชื่อแรกนิ้วก้อย

Filename(12) = "4_1" ‘ ชื่อแรกนิ้วนาง

Filename(13) = "3_1" ‘ข้อแรกนิ้วกลาง

Filename(14) = "2_1" ‘ข้อแรกนิ้วชี้

Filename(15) = "1_1" ‘ข้อแรกนิ้วโป้ง

Filename(16) = "paw" ‘อุ้งมือ

Filename(17) = "wrist" ‘ข้อมือ

‘ Map_File_Index(1 to 22) เป็นการกำหนดการเคลื่อนไหวของข้อต่างๆ

‘ ซึ่งจะถูกใช้ในฟังก์ชัน Display

Map_File_Index(1) = 5 ‘ การเคลื่อนไหวเบอร์ 1 เป็นของข้อปลายนิ้วโป้ง (ซึ่งแทนในการ

‘ เคลื่อนไหวจะถูกกำหนดใน display function อีกที)

Map_File_Index(2) = 10 ‘ การเคลื่อนไหวเบอร์ 2 เป็นของข้อกลางนิ้วโป้ง

Map_File_Index(3) = 15 ‘ การเคลื่อนไหวเบอร์ 3 เป็นของข้อแรกนิ้วโป้ง

Map_File_Index(4) = 15 ‘ การเคลื่อนไหวเบอร์ 4 เป็นของข้อแรกนิ้วโป้ง (ที่ข้อแรกของทุกนิ้ว

‘ เคลื่อนไหวได้สองแกน เลยต้องกำหนดสองค่า)

Map_File_Index(5) = 4 ‘ การเคลื่อนไหวเบอร์ 5 เป็นของข้อปลายนิ้วชี้

Map_File_Index(6) = 9 ‘ การเคลื่อนไหวเบอร์ 6 เป็นของข้อกลางนิ้วชี้

Map_File_Index(7) = 14 ‘ การเคลื่อนไหวเบอร์ 7 เป็นของข้อแรกนิ้วชี้

Map_File_Index(8) = 14 ‘ การเคลื่อนไหวเบอร์ 8 เป็นของข้อแรกนิ้วชี้

Map_File_Index(9) = 3 ‘ การเคลื่อนไหวเบอร์ 9 เป็นของข้อปลายนิ้วกลาง

Map_File_Index(10) = 8 ‘ การเคลื่อนไหวเบอร์ 10 เป็นของข้อกลางนิ้วกลาง

Map_File_Index(11) = 13 ‘ การเคลื่อนไหวเบอร์ 11 เป็นของข้อแรกนิ้วกลาง

Map_File_Index(12) = 13 ‘ การเคลื่อนไหวเบอร์ 12 เป็นของข้อแรกนิ้วกลาง

Map_File_Index(13) = 2 ‘ การเคลื่อนไหวเบอร์ 13 เป็นของข้อปลายนิ้วนาง

Map_File_Index(14) = 7 ‘ การเคลื่อนไหวเบอร์ 14 เป็นของข้อกลางนิ้วนาง

Map_File_Index(15) = 12 ‘ การเคลื่อนไหวเบอร์ 15 เป็นของข้อแรกนิ้วนาง

Map_File_Index(16) = 12 ‘ การเคลื่อนไหวเบอร์ 16 เป็นของข้อแรกนิ้วนาง

Map_File_Index(17) = 1 ‘ การเคลื่อนไหวเบอร์ 17 เป็นของข้อปลายนิ้วก้อย

Map_File_Index(18) = 6 ‘ การเคลื่อนไหวเบอร์ 18 เป็นของข้อกลางนิ้วก้อย

Map_File_Index(19) = 11 ‘ การเคลื่อนไหวเบอร์ 19 เป็นของข้อแรกนิ้วก้อย

Map_File_Index(20) = 11 ‘ การเคลื่อนไหวเบอร์ 20 เป็นของข้อแรกนิ้วก้อย

Map_File_Index(21) = 16 ‘ การเคลื่อนไหวเบอร์ 21 เป็นของอุ้งมือ

Map_File_Index(22) = 17 ‘ การเคลื่อนไหวเบอร์ 22 เป็นของข้อมือ

‘ ส่วนต่อไปเป็นส่วนกำหนดค่าไครซ์สามติปกติ ซึ่งจะมีการสร้างชินกราฟตามหัวข้อ (11.5)

```

Direct3DRMCreate pD3drm
pD3drm.CreateFrame Nothing, Scene
pD3drm.CreateFrame Scene, Camera
pD3drm.CreateFrame Scene, Lights
Camera.SetPosition Scene, 0, 0, 0
Lights.SetPosition Scene, 10, 10, -10
pD3drm.CreateLightRGB D3DRMLIGHT_POINT, 1, 1, 1, 11
Lights.AddLight l1
pD3drm.CreateLightRGB D3DRMLIGHT_AMBIENT, 0.5, 0.5, 0.5, 12
Scene.AddLight l2
Call Create3DWorld
Camera.SetPosition Scene, 0, 0, -25
pD3drm.CreateFrame Scene, Hand(17)
Hand(17).AddVisual hand_mesh(17)
pD3drm.CreateFrame Hand(17), Hand(16)
Hand(16).AddVisual hand_mesh(16)
For a = 11 To 15
    pD3drm.CreateFrame Hand(16), Hand(a)
    Hand(a).AddVisual hand_mesh(a)
Next a
For a = 6 To 10
    pD3drm.CreateFrame Hand(a + 5), Hand(a)
    Hand(a).AddVisual hand_mesh(a)
Next a
For a = 1 To 5
    pD3drm.CreateFrame Hand(a + 5), Hand(a)
    Hand(a).AddVisual hand_mesh(a)
Next a

```

‘ การกำหนดตำแหน่งของนิ้วบนหน้าจอ

```

Hand(17).SetPosition Scene, 0, -4, 0
Hand(16).SetPosition Hand(17), 1.25, 1.8, 0
Hand(15).SetPosition Hand(16), -3.5, 0, 0
Hand(14).SetPosition Hand(16), -2.5, 3.3, 0

```

```

Hand(13).SetPosition Hand(16), -1.5, 3.6, 0
Hand(12).SetPosition Hand(16), -0.5, 3.4, 0
Hand(11).SetPosition Hand(16), 0.4, 2.5, 0
Hand(10).SetPosition Hand(15), -0.8, 1.75, 0
Hand(9).SetPosition Hand(14), -0.2, 2, 0
Hand(8).SetPosition Hand(13), 0, 2.25, 0
Hand(7).SetPosition Hand(12), 0.2, 2.2, 0
Hand(6).SetPosition Hand(11), 0.25, 1.85, 0
Hand(5).SetPosition Hand(10), -0.25, 1.6, 0
Hand(4).SetPosition Hand(9), -0.15, 1.2, 0
Hand(3).SetPosition Hand(8), -0.15, 1.35, 0
Hand(2).SetPosition Hand(7), -0.05, 1.35, 0
Hand(1).SetPosition Hand(6), 0.1, 1.1, 0

```

Save_Before = True ‘ใช้แบ่งไหมดการทำงานระหว่างคาลิเบสไหมดกับไหมดเรนเดอร์

‘ การกำหนดค่าของพอร์ตอนุกรม

```
MSComm1.Settings = "9600,n,8,1"
```

```
MSComm1.CommPort = 1
```

```
MSComm1.InputLen = 1
```

```
MSComm1.PortOpen = True
```

```
MSComm1.RThreshold = 1
```

```
check = 0
```

```
mode = True
```

```
End Sub
```

‘ ฟังก์ชัน UpdateScreen() เป็นการเรนเดอร์ภาพบนวิวพอร์ต

```
Public Sub UpdateScreen()
```

```
    pViewport.Clear
```

```
    pViewport.Render Scene
```

```
    Dev.Update
```

```
Exit Sub
```

```
End Sub
```

‘ ฟังก์ชัน Create3Dworkd() เป็นการโหลดวัตถุจากไฟล์เมชไคเรกซ์เอ็กซ์ (.x)

Public Sub Create3DWorld()

Dim a As Integer

For a = 1 To 17

pD3drm.CreateMeshBuilder Hand_Builder(a)

With Hand_Builder(a)

.Load App.Path & "\" & Filename(a) & ".x", 0, 0, 0

.Scale 10, 5, 5

.SetColorRGB 0.88, 0.67, 0.53

.CreateMesh hand_mesh(a)

End With

Hand_Builder(a).CreateMesh hand_mesh(a)

Next

End Sub

‘ ฟังก์ชัน Form_Paint() ถ้าฟอร์มมีการเปลี่ยนแปลงให้ทำการเรนเดอร์ภาพใหม่

Private Sub Form_Paint()

Call UpdateScreen

End Sub

‘ ฟังก์ชัน Form_Resize() ถ้ามีการย่อ, ขยายฟอร์ม ต้องมีการเปลี่ยนขนาดวิวพอร์ตใหม่

Private Sub Form_Resize()

Dim FormWidth As Long

Dim formHeight As Long

FormWidth = ScaleX(ScaleWidth, ScaleMode, vbPixels)

formHeight = ScaleY(ScaleHeight, ScaleMode, vbPixels)

If Not (pViewport Is Nothing) Then Set pViewport = Nothing

If Not (pClipper Is Nothing) Then Set pClipper = Nothing

DirectDrawCreateClipper 0, pClipper, Nothing

pClipper.SetHWND 0, hWnd

pD3drm.CreateDeviceFromClipper pClipper, ByVal 0&, FormWidth, formHeight, Dev

pD3drm.CreateViewport Dev, Camera, 0, 0, FormWidth, formHeight, pViewport

With Dev

.SetDither 1

.SetQuality D3DRMLIGHT_ON Or D3DRMFILL_SOLID Or D3DRMSHADE_GOURAUD

End With

Call UpdateScreen

End Sub

‘ ฟังก์ชัน Form_Unload(Cancel As Integer) เป็นการคืนค่าลบค่าที่ได้ของไวดอนใช้ในโปรแกรมทั้ง

Private Sub Form_Unload(Cancel As Integer)

Dim a As Integer

Set pD3drm = Nothing

Set Scene = Nothing

Set pClipper = Nothing

Set Camera = Nothing

Set pViewport = Nothing

Set Dev = Nothing

Set l1 = Nothing

Set l2 = Nothing

Set Lights = Nothing

For a = 1 To 17

Set Hand_Builder(a) = Nothing

Set hand_mesh(a) = Nothing

Set Hand(a) = Nothing

Next a

End Sub

· ฟังก์ชัน MSComm1_OnComm() เหมือนที่อธิบายไว้ในบทที่ 10

```
Private Sub MSComm1_OnComm()
```

```
On Error GoTo Com_Error
```

```
    If mode Then
```

```
        If check < 4 Then
```

```
            buffer(check) = MSComm1.Input
```

```
            num(check) = Asc(buffer(check))
```

```
            If check = 3 Then
```

```
                If num(2) = num(0) + 1 Then mode = False
```

```
            End If
```

```
        Else
```

```
            buffer(0) = MSComm1.Input
```

```
            mode = False
```

```
        End If
```

```
    Else
```

```
        buffer(0) = MSComm1.Input
```

```
        If check > 1 Then
```

```
            num(0) = Asc(buffer(0))
```

```
            check = 0
```

```
        Else
```

```
            If Save_Before Then
```

```
                Before(num(0) + 1) = Asc(buffer(0))
```

```
                Text1(num(0)).Text = Asc(buffer(0))
```

```
            Else: Display num(0) + 1, Asc(buffer(0))
```

```
            End If
```

```
        End If
```

```
    End If
```

```
    check = check + 1
```

```
Exit Sub
```

Com_Error:

check = 0

mode = True

Exit Sub

End Sub



บทที่ 6

ผลการทดลอง

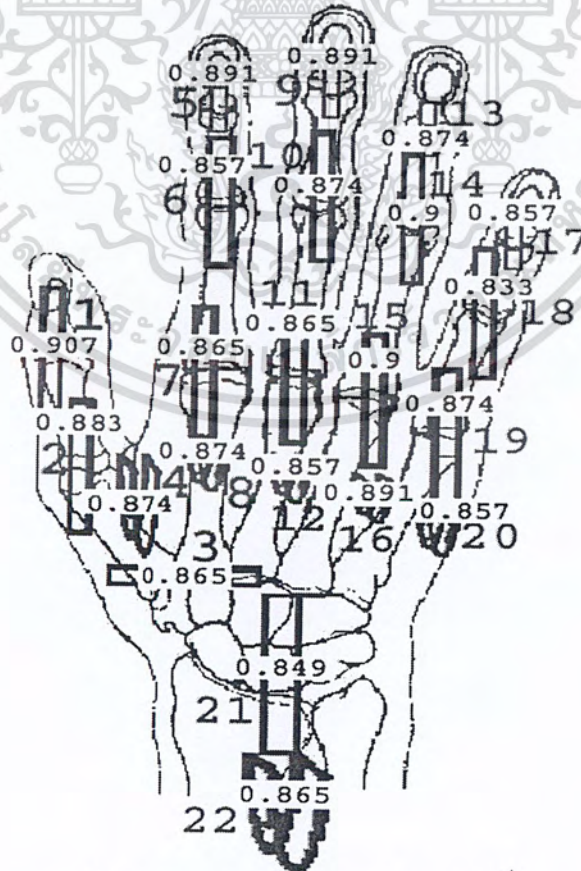
เพื่อให้ได้รู้ถึงค่าความละเอียดขององศาที่วัดได้จากการเลื่อนไหวของนิ้วมือและข้อมือเราจึงทำการทดลองเพื่อทดสอบอุปกรณ์ซึ่งมีท่า 2 วิธีดังนี้

6.1 การทดลองวิธีที่ 1

ทำการทดลองโดยการวัดค่าความละเอียดขององศาที่วัดได้คือการเปลี่ยนแปลงค่าของเซนเซอร์ 1 ค่าโดยครั้งแรกทำการขยับแกนของเซนเซอร์ให้อยู่ในแนว 0° จากนั้นทำการขยับแกนหมุนของเซนเซอร์ในอยู่ในแนว 180° และทำการวัดค่าที่เซนเซอร์ที่เปลี่ยนแปลงในแต่ละข้อจนครบ 22 ข้อโดยการทดลองนี้เราไม่ใส่ถุงมือแต่อย่างใด

6.1.1 ผลการทดลองวิธีที่ 1

ผลการทดลองนี้แสดงดังที่รูปที่ 6.1



รูปที่ 6.1 แสดงค่าองศาที่วัดได้ในแต่ละข้อนิ้ว

6.1.2 สรุปผลการทดลองวิธีที่ 1

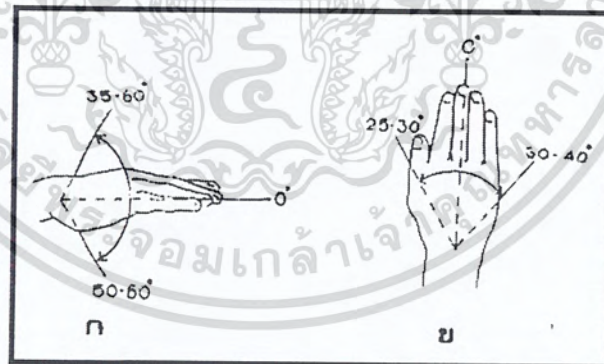
การผลการทดลองจะเห็นได้ว่าความละเอียดของเซนเซอร์ที่ทำมาใช้จริง ๆ นั้นมีความละเอียดอยู่ที่ประมาณ 0.8-0.9 ซึ่งถือว่ามีความละเอียดมากพอสมควรทีเดียว

6.2 การทดลองวิธีที่ 2

ทำการทดลองโดยการใส่ถุงมือและขยับมือในท่ากำแบให้ เกิดฟลักซ์ชัน / เอ็คชเทนชัน แอ็บดักชัน / แอ็ดดักชัน และ อัลนา คีฟิเอชัน / เรเดียล คีฟิเอชัน ให้มากที่สุดหรือพูดอีกอย่างก็คือพยายามให้เซนเซอร์เปลี่ยนแปลงค่าให้ได้มากที่สุดนั่นเอง โดยหาค่าสัมบูรณ์ที่เซนเซอร์แต่ละตัวเปลี่ยนแปลงซึ่งเราได้ทำการเขียนโปรแกรมให้ทำการเก็บค่าสัมบูรณ์ในการทำทดลองแต่ละครั้งโดยอัตโนมัติไว้แล้ว ทำการทดลองอย่างนี้จำนวน 100 ครั้ง จากนั้นนำค่าที่ได้ 100 ค่าของแต่ละเซนเซอร์มาหาค่าเฉลี่ยแล้วนำไปหารกับค่าองศาที่ขยับได้จริงที่ตำแหน่งตรงกับที่เซนเซอร์นั้นวัด ก็จะได้ค่าองศาต่อการเปลี่ยนแปลงค่าเซนเซอร์ 1 ค่า

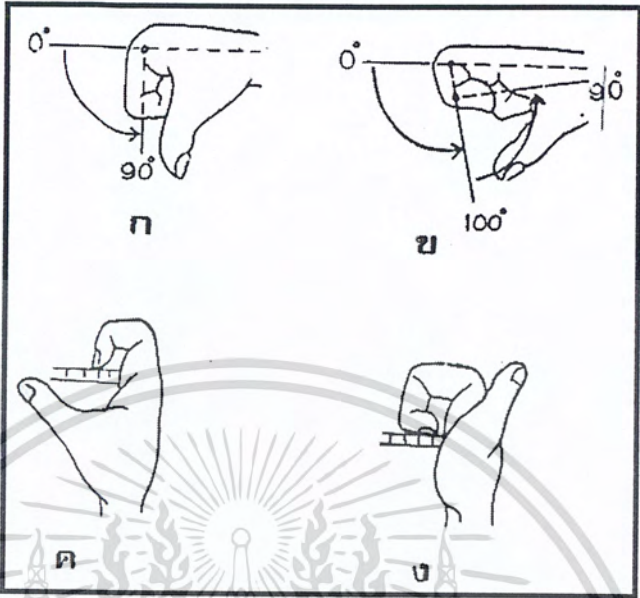
ซึ่งรายละเอียดของวิธีการทำการทดลองมีดังนี้

ขั้นที่ 1 ทำการใส่ถุงมือจากนั้นทำการเคลื่อนไหวในส่วนของข้อมือในแนวฟลักซ์ชัน / เอ็คชเทนชัน ให้ได้มากที่สุดดังรูปที่ 6.2ก จากนั้นทำการเคลื่อนไหวในแนวแอ็บดักชัน / แอ็ดดักชัน ให้ได้มากที่สุดดังรูปที่ 6.2ข



รูปที่ 6.2 การทดลองขั้นที่ 1

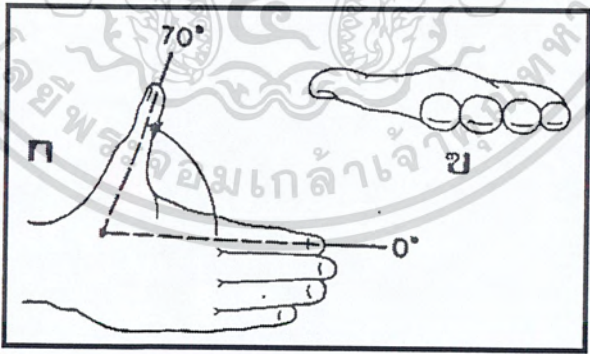
ขั้นที่ 2 ทำการขยับนิ้วทั้ง 4 (ยกเว้นนิ้วโป้ง) ในแนวฟเลกชัน / เอ็กซเทนชันให้ได้มากที่สุดดังรูปที่ 6.3



รูปที่ 6.3 การทดลองขั้นที่ 2

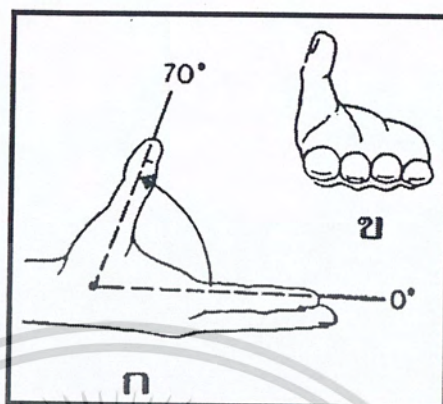
ขั้นที่ 3 ทำการเคลื่อนไหวในแนวแอ็บดักชัน / แอดดักชันของนิ้วทั้ง 4 ให้ได้มากที่สุดโดยตอนแรกให้มืออยู่ในท่าดังรูปที่ 6.2ข จากนั้นทำการกางมือให้ได้มากที่สุด

ขั้นที่ 4 ทำการเคลื่อนไหวนิ้วโป้งโดยขั้นแรกหุบนิ้วโป้งแนบกับมือจากนั้นกางออกในแนวแอ็บดักชันดังรูปที่ 6.4



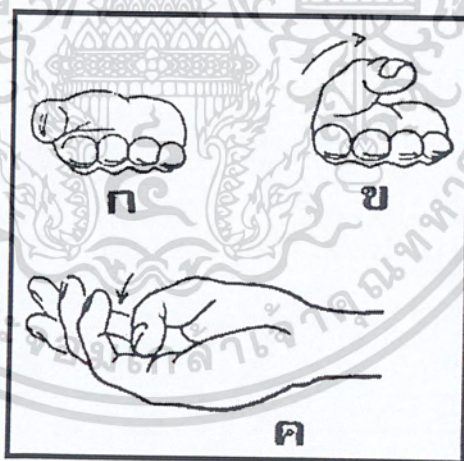
รูปที่ 6.4 การทดลองขั้นที่ 4

ขั้นที่ 5 ทำการเคลื่อนไหวนิ้วโป้งในแนวเพลเมอร์ฟเลกชันโดยกางนิ้วโป้งออกมาทางด้านฝ่ามือ
 ดังรูปที่ 6.5



รูปที่ 6.5 การทดลองขั้นที่ 5

ขั้นที่ 6 ทำการเคลื่อนไหวนิ้วโป้งในแนวโรเตชันโดยหุบนิ้วโป้งไว้ข้างฝ่ามือนี้อย่างรูปที่ 6.6ก
 จากนั้นเคลื่อนไหวนิ้วโป้งให้ไปทางนิ้วก้อยให้ได้มากที่สุดดังรูปที่ 6.6ข



รูปที่ 6.6 การทดลองขั้นที่ 6

6.2.1 ผลการทดลองวิธีที่ 2

สำหรับผลการทดลองของวิธีที่ 2 โดยทำการทดลองจำนวน 100 ครั้งนั้นแสดงดังตารางที่ 6.1

ข้อนิ้ว มือหรือ ข้อมือ	ข้อข้อที่เซนเซอร์ทำ การวัดการเคลื่อนไหว หรือลักษณะการ เคลื่อนไหว	ค่าเฉลี่ยที่ เซนเซอร์ เปลี่ยน แปลง	ค่าที่ เซนเซอร์ เปลี่ยนแปลง มากที่สุด	ค่าที่ เซนเซอร์ เปลี่ยน แปลงน้อย ที่สุด	องศา ข้อมือหรือ ข้อนิ้ว เคลื่อนไหวได้	ค่าความ ละเอียดที่ เซนเซอร์วัด ได้ต่อ 1 ค่า
นิ้วโป้ง	IP	89.68	72	109	100°	1.12°
	MCP	68.53	56	75	65°	0.95°
	Abduction Sensor	55.14	50	60	70°	1.27°
	Rotation Sensor	88.69	78	99	110°	1.24°
นิ้วชี้	DIP	72.71	46	134	80°	1.10°
	PIP	121.92	103	153	100°	0.82°
	MCP	121.75	101	158	120°	0.99°
	Abduction Sensor	33.02	25	43	30°	0.91°
นิ้ว กลาง	DIP	57.89	42	114	80°	1.38°
	PIP	107.28	96	116	100°	0.93°
	MCP	99.71	84	111	120°	1.20°
	Abduction Sensor	25.51	18	34	30°	1.18°
นิ้วนาง	DIP	67.1	43	111	80°	1.19°
	PIP	77.14	62	134	100°	1.30°
	MCP	79.49	65	92	120°	1.51°
	Abduction Sensor	30.61	24	37	30°	0.98°
นิ้วก้อย	DIP	30.49	14	52	80°	2.62°
	PIP	68.90	53	85	100°	1.45°
	MCP	111	93	123	120°	1.08°
	Abduction Sensor	34.91	26	42	30°	0.86°
ข้อมือ	Flexion/Extention	89.90	76	106	110°	1.22°
	Ulnar Deviation/Radial Deviation	37.40	24	76	50°	1.34°

ตารางที่ 6.1 แสดงค่าที่เซนเซอร์เปลี่ยนแปลงในแต่ละข้อและองศาที่เคลื่อนไหวจริงจากค่าเฉลี่ยทำการ

ทดลอง 100 ครั้ง

6.2.2 สรุปผลการทดลองวิธีที่ 2

จากการทดลองจะเห็นว่าค่าองศาที่วัดได้ต่อการเปลี่ยนแปลงค่าของเซนเซอร์หนึ่งค่านั้นจะอยู่ที่ประมาณ $0.82^{\circ} - 1.51^{\circ}$ ซึ่งจะมีเพียงข้อ DIP ของนิวก็้อยเท่านั้นที่ค่ามากถึง 2.62° ซึ่งอาจจะเป็นเพราะว่าข้อปลายของนิวก็้อยนั้นมีแรงไม่มากพอที่จะทำให้เซนเซอร์เปลี่ยนแปลงค่าได้มาก ซึ่งสังเกตว่าข้อปลายของทุกข้อยกเว้นนิวโป่งนั้นก็ประสบปัญหานี้เช่นกัน และการที่ค่ามากที่สุด และน้อยสุดนั้นแตกต่างกันพอสมควรก็เนื่องมาจากความผิดพลาดของผู้ทำการทดลองซึ่งการทำการทดลองจำนวนมากอาจเกิดการเมื่อยล้าทำให้ค่าที่ได้ไม่สม่ำเสมอเท่าที่ควร อีกทั้งขนาดมือของผู้ทำการทดลองก็มีส่วนทำให้การทดลองแตกต่างกันด้วย ซึ่งผลการทดลองโดยรวมเซนเซอร์วัดค่าที่มีความละเอียดพอสมควรทีเดียว



บทที่ 7

บทวิจารณ์และสรุป

7.1 สรุปผลการทำงาน

จากการทดลองเราพบว่าเมื่อขยับถุงมือแล้ว ค่าที่เปลี่ยนแปลงไปในคอมพิวเตอร์นั้นเปลี่ยนแปลงไปด้วยความละเอียดพอสมควรเป็นที่น่าพอใจ แต่ก็ยังมีบางข้อต่อเท่านั้นที่ยังไม่ละเอียดเท่าที่ควร และ ความสอดคล้องของการเคลื่อนไหวในแต่ละข้อยังไม่ดีนัก ซึ่งจะต้องปรับปรุงในส่วนการออกแบบตัวถุงมือให้ดีขึ้นต่อไปอีก แต่ในส่วนของการแสดงผลของมือในโฮมดกราฟฟิก 3 มิตินั้นยังมีข้ออยู่ทำให้บางข้อเช่นข้อ MCP ของบางข้อมือนั้นยังแสดงผลได้ไม่ดีนักคือ บางทีก็แสดงค่าที่ถูกต้องออกมา แต่บางทีก็ไม่ซึ่งเราจะต้องการปรับปรุงในส่วนนี้ต่อไป

7.2 ปัญหาและอุปสรรคในการทำงาน

7.2.1 ส่วนงานทางด้านฮาร์ดแวร์

7.2.1.1 ส่วนของวงจร

ไม่สามารถหาไอซีตัวที่ต้องการใช้งานได้จากการศึกษาการทำงานของไอซีและทำการต่อวงจรเพื่อทำการทดสอบการทำงานของไอซีต่างๆ พบว่าไอซีบางตัวเช่น ไอซี 555 นั้นตัวที่ต้องการใช้คือ LM555 ซึ่งเป็นของบริษัท National นั้นหาไม่ได้ในท้องตลาด ดังนั้นจึงต้องใช้ 555 ซึ่งมีอยู่ทั่วไปซึ่งสร้างสัญญาณนาฬิกาได้ไม่ถึงตามที่ค่าชี้ทแนะนำแต่เมื่อนำมาทดลองใช้แล้วก็ไม่มีปัญหาแต่อย่างไร และ ADC0816 ซึ่งหาไม่ได้เช่นกันจึงได้ใช้ ADC0817 แทนซึ่งเหมือนกันเพียงแต่เปอร์เซ็นต์ความผิดพลาดมากกว่าเท่านั้น

7.2.1.2 ส่วนของตัวถุงมือ

เนื่องจากว่าถุงมือที่เราสร้างนั้นไม่มีตัวอย่างหรือแนวทางการสร้างถุงมือ ข้อมูลที่มีก็คือรูปถ่ายของถุงมือที่เป็นของต่างประเทศซึ่งค้นหามาจากตามเว็บไซต์ต่าง ๆ ทำให้การสร้างถุงมือนี้นี้จึงต้องอาศัยการลองผิดลองถูกและการประยุกต์จากรูปถุงมือที่มีอยู่นำมาสร้างเป็นถุงมือนั้นแบบทำให้เป็นอุปสรรคต่อการสร้างฮาร์ดแวร์พอสมควร

7.2.2 ส่วนงานทางด้านซอฟต์แวร์

จากการศึกษาและการเขียนโปรแกรมการแสดงผลโคเร็กซ์เอ็กซ์ มานั้น ได้พบว่ามีความซับซ้อนต่อการเข้าใจ แต่เมื่อเข้าใจแล้วก็สามารถที่จะเขียนโปรแกรมแสดงผลอย่างง่ายๆ ได้โดยไม่มีความยากลำบากเท่าใดนัก แต่ได้พบปัญหาคือเมื่อเมื่อรันโปรแกรมไปซักพัก โปรแกรมจะเกิดบัคขึ้นในการ transform metric ส่งผลให้วัตถุขนาดใหญ่อันหนึ่ง ซึ่งส่วนนี้เราได้พยายามแก้ทุกวิถีทาง ไม่ว่าจะแก้ไข เขียน

ใหม่ เปลี่ยนแนวการเขียน จนสุดท้ายก็ยังพบปัญหาเดิม อีกทั้งได้ขอความช่วยเหลือจากผู้มีความรู้ความสามารถที่ต่างๆ ก็มีผู้ให้ความช่วยเหลือ ได้ทำการศึกษาโปรแกรมของเราเพื่อตรวจหาสาเหตุของบั๊ก และมีการติดต่อกันตลอดเวลาเป็นเวลากว่า 3 อาทิตย์ แต่ก็ยังปรากฏว่าหาสาเหตุของบั๊กไม่เจอเช่นกัน เราจึงหันมาใช้เอนจินตัวอื่นเลยคือมอร์ฟิต แต่ว่ามอร์ฟิตทำงานช้า และไม่เหมาะกับงานเท่าที่ควร หลังจากเขียนไปได้ซักพักได้เห็นว่า ทำงานช้ามากและมีความสวยงามน้อยกว่า จึงได้ลองเขียนโปรแกรมไคเร็กซ์เอ็กซ์ตัวเก่าใหม่อีกครั้ง และได้ทำการเปลี่ยนวิธีการเขียนโปรแกรม โดยการรีเซ็ตค่าเก่าทุกครั้งที่ได้รับค่าใหม่มา จนในที่สุดก็แก้ปัญหาได้ และได้ทราบถึงปัญหาที่แท้จริง คือเป็นผลจากการทำงานเลขจุดทศนิยมมากเกินไป ทำให้เกิดการผิดพลาดและแองคิป ซึ่งรวมเวลาที่ใช้ทั้งหมดกับปัญหานี้รวมเป็นเวลาถึง 2 เดือนทีเดียว

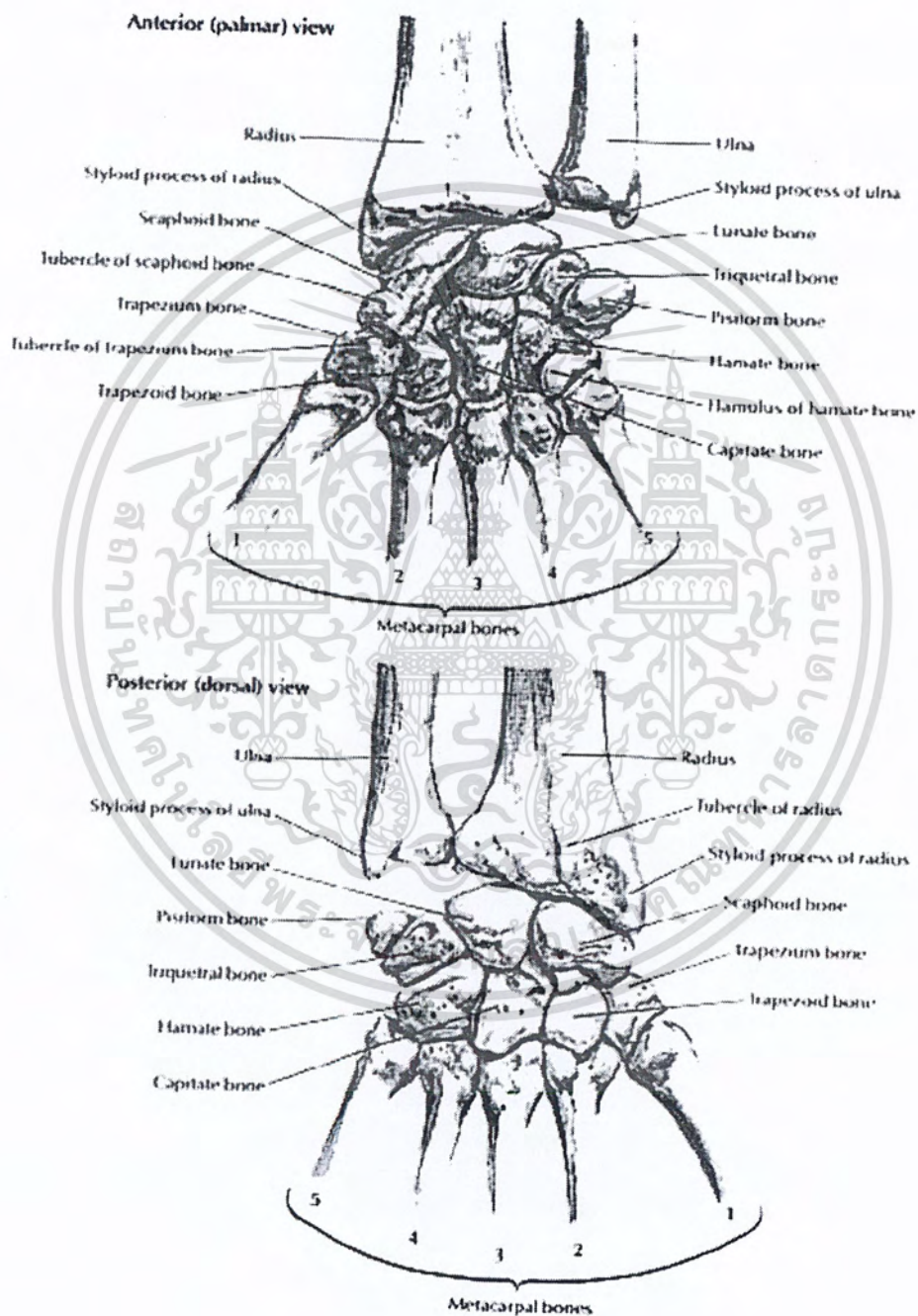
7.3 แนวทางการพัฒนาต่อไป

โครงการนี้เป็นโครงการต้นแบบ จึงมีหลายสิ่งที่ต้องแก้ไข และหลายสิ่งที่จะพัฒนาให้มีขึ้น ซึ่งมีแนวทางการพัฒนาดังนี้

- เพื่อให้การสร้างถุงมือต้นแบบนั้นง่ายขึ้นจึงใช้ไม่เป็นวัสดุในการทำตัวถุงมือซึ่งมีความทนทานพอสมควรแต่ก็ไม่มากนัก อีกทั้งสายไฟที่เชื่อมต่อยังไม่แข็งแรงพอ จึงต้องปรับปรุงให้ถุงมือมีความทนทานมากขึ้น อีกทั้งถุงมือนี้ออกแบบให้เหมาะกับผู้ออกแบบคนเดียวจึงปรับปรุงถุงมือให้เหมาะสมกับผู้ใช้ทุกคน และการรับค่าของถุงมือนั้นยังไม่ละเอียดมากพอ และการเคลื่อนไหวยาง่ายวัดค่าได้ไม่ดีนักอันเนื่องมาจากความสัมพันธ์ในแต่ละข้อยังไม่สอดคล้องกันนัก จึงต้องปรับปรุงการออกแบบตัวถุงมือให้สามารถรับค่าให้ได้ละเอียดมากกว่านี้
- ทำไคร์เวอร์ให้กับอุปกรณ์ชนิดนี้ เพื่อที่จะสามารถใช้งานในคอมพิวเตอร์ได้หลากหลายยิ่งขึ้น
- สร้างเกมหรือโปรแกรมที่แสดงให้เห็นถึงคุณสมบัติของเครื่องมือควบคุมในงานที่เหมาะสม เช่น โปรแกรมแปลภาษามือเป็นตัวหนังสือหรือพูดออกมาได้
- พัฒนาจากจากถุงมือให้มากกว่านี้ เช่นอาจจะเพิ่มทั้งแขนหรือทั้งตัว เพื่อใช้งานในแนวต่างๆ เช่นบังคับแขนกล ทำการทดลองวิจัยสารเคมีอันตรายแทนคนได้

ภาคผนวก ก.

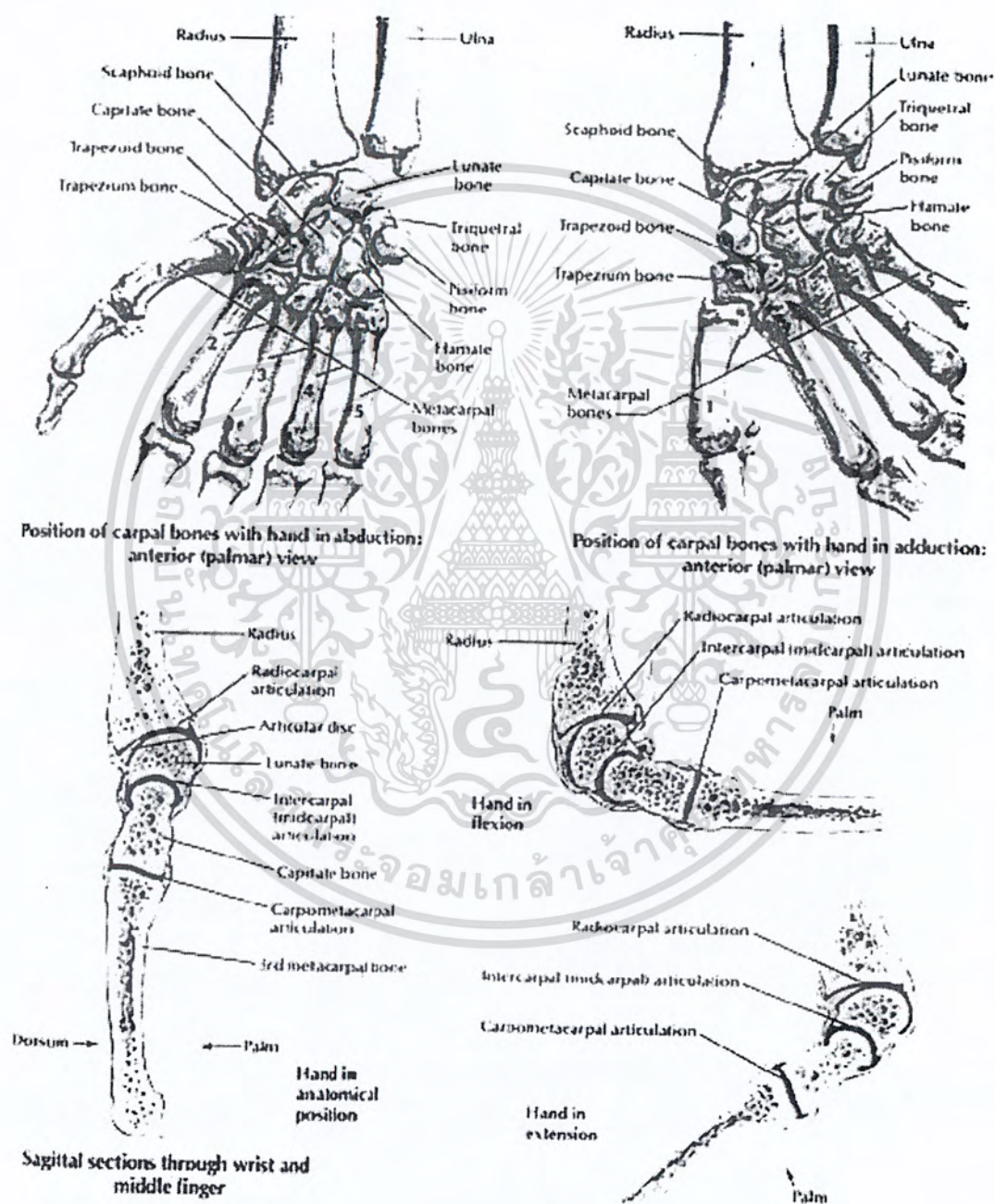
Carpal Bones



UPPER LIMB

รูปที่ ก.1 ภาพมือด้านบน

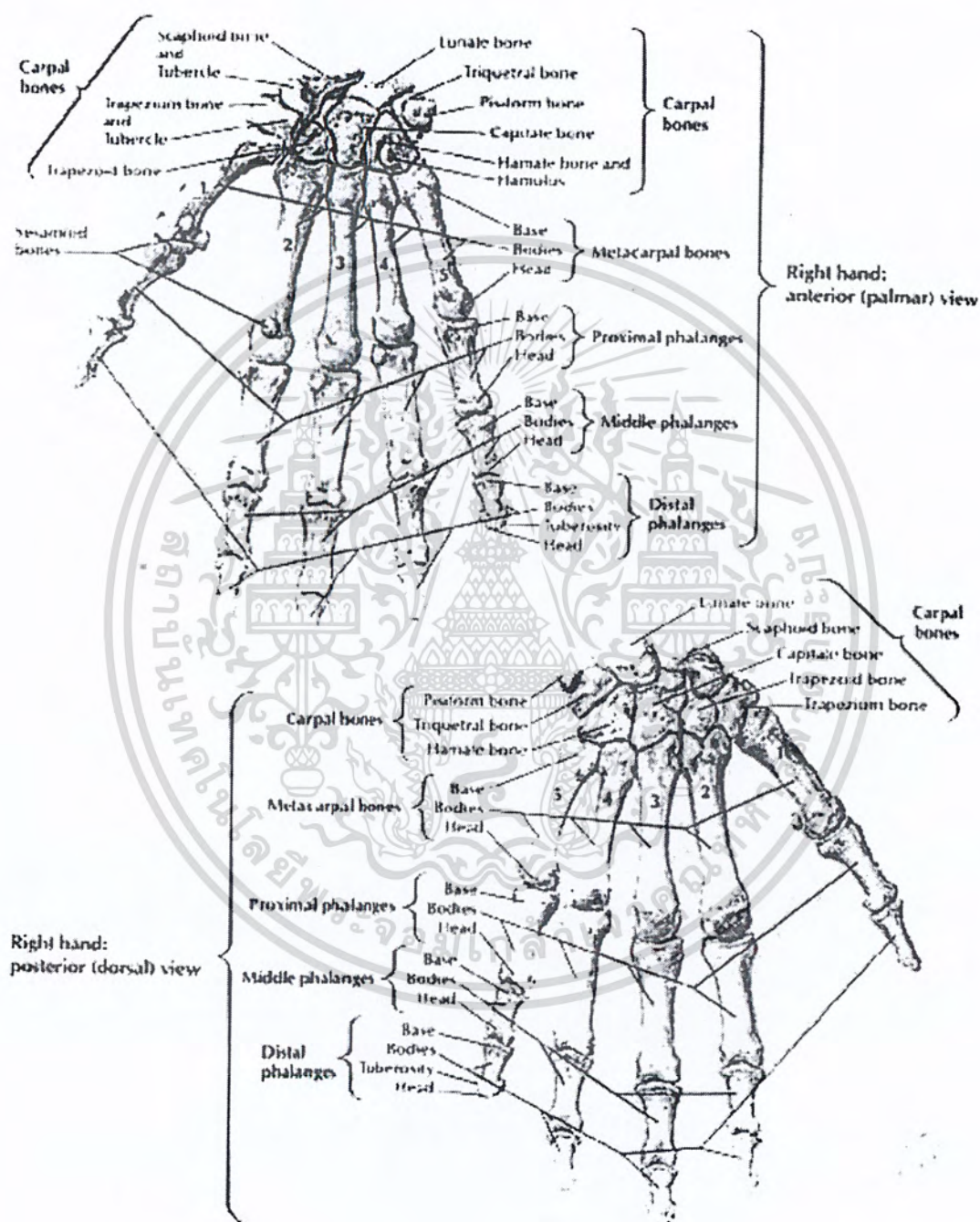
Movements of Wrist



WRIST AND HAND

รูปที่ ก.2 ข้อมือและมือ

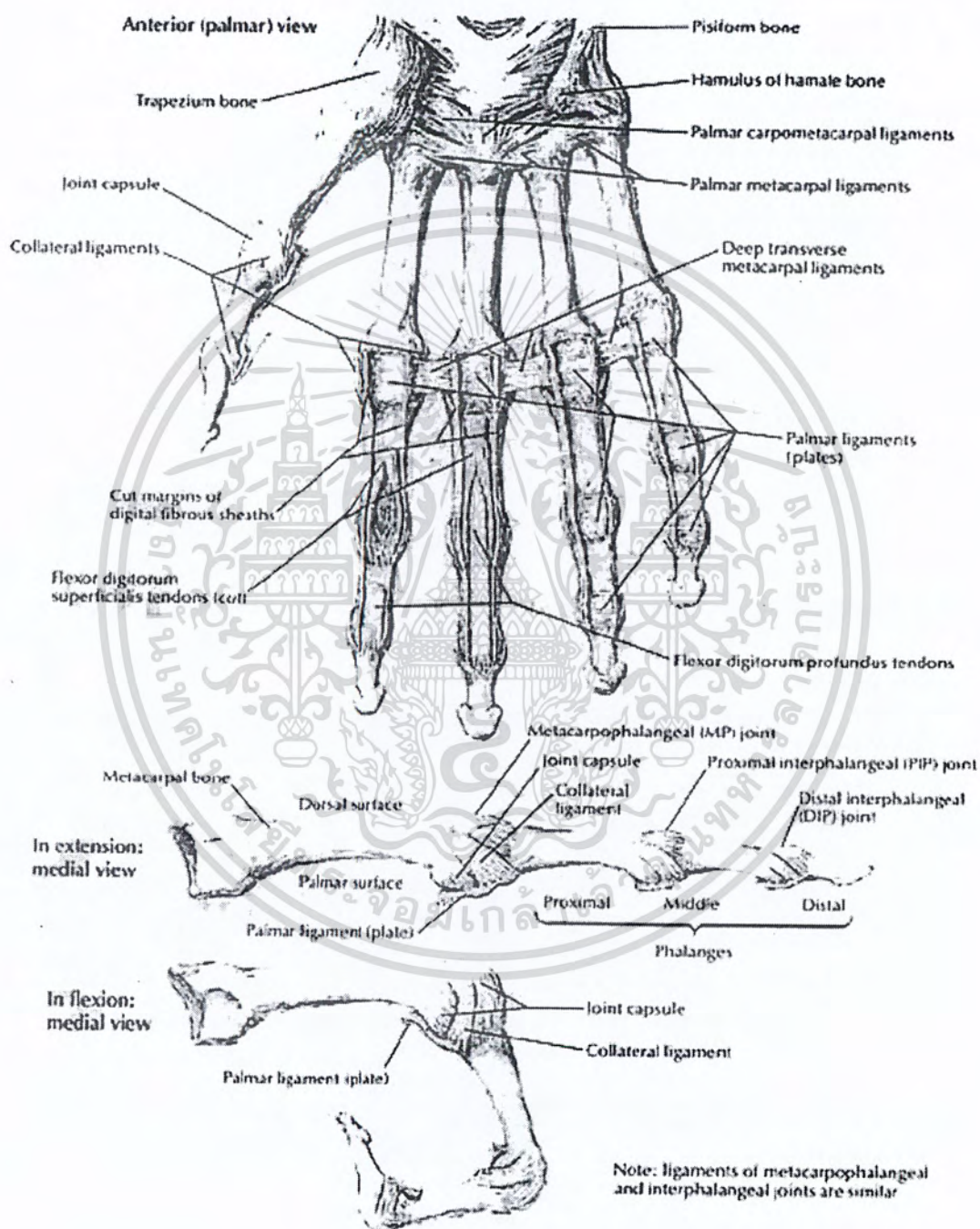
Bones of Wrist and Hand



UPPER LIMB

รูปที่ ก.3 ภาพมือด้านบน

Metacarpophalangeal and Interphalangeal Ligaments



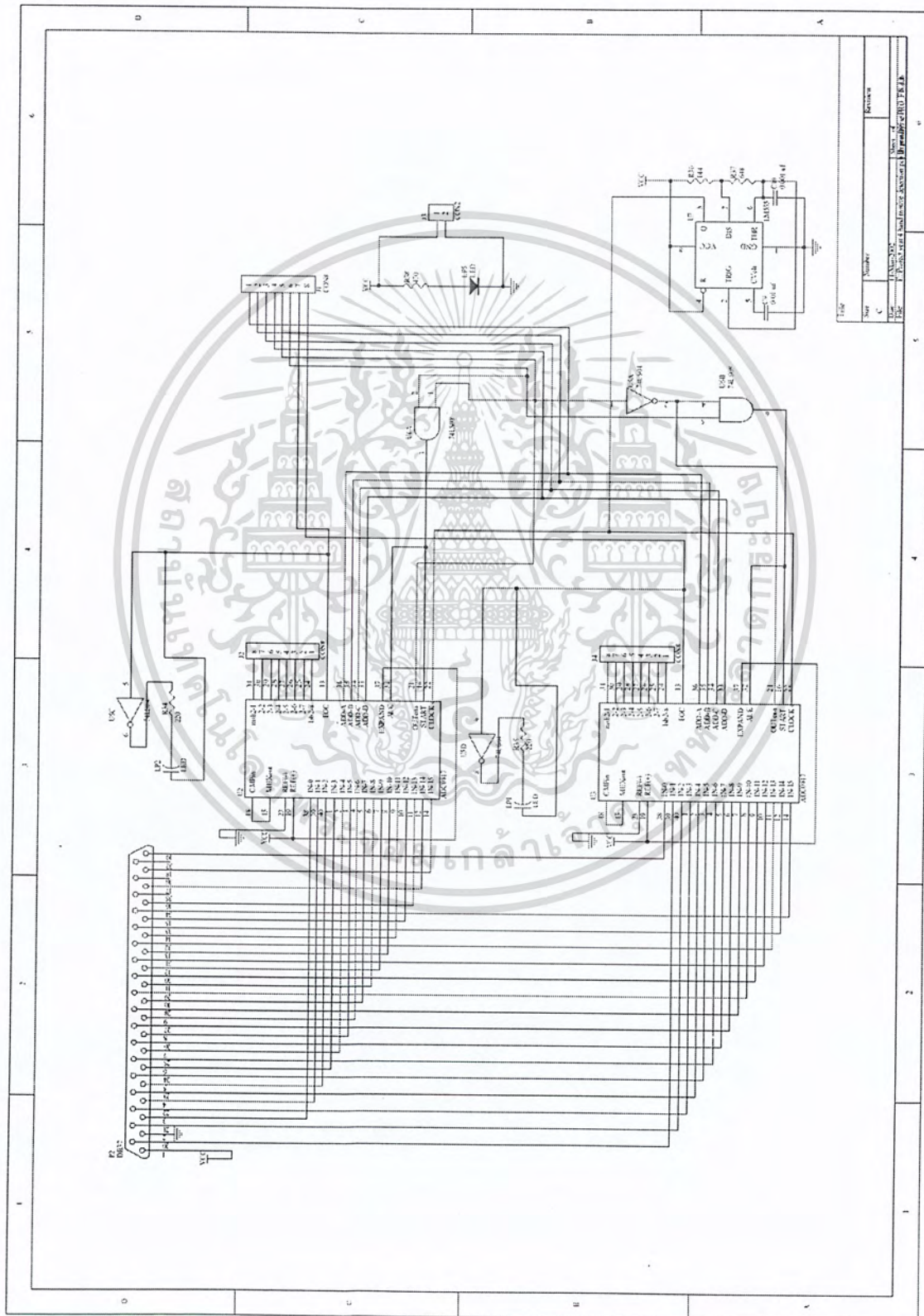
WRIST AND HAND

รูปที่ ๓.๔ ข้อมือและมือ

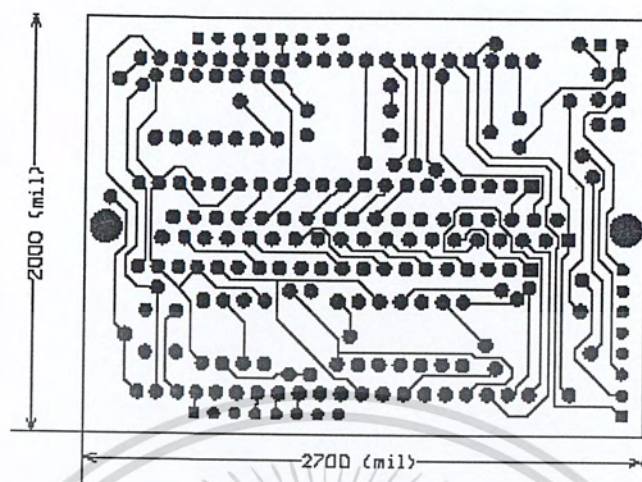
ภาคผนวก ข.

สำหรับการนำสถิติมาทำเป็นแผ่นปรี้นต์นั้นเพื่อความกะทัดรัดจึงแบ่งแผ่นปรี้นต์ 2 แผ่นซ้อน

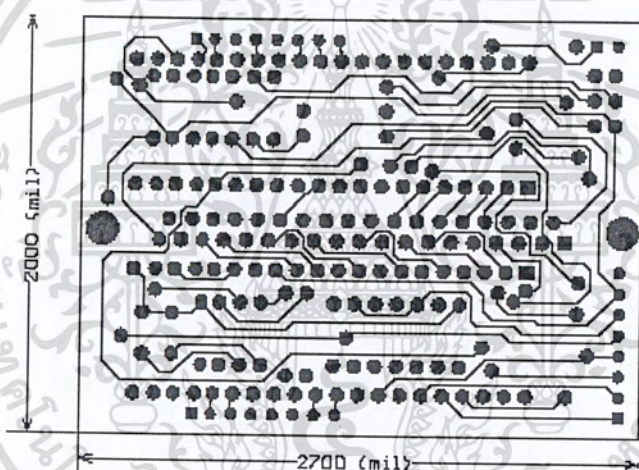
กัน



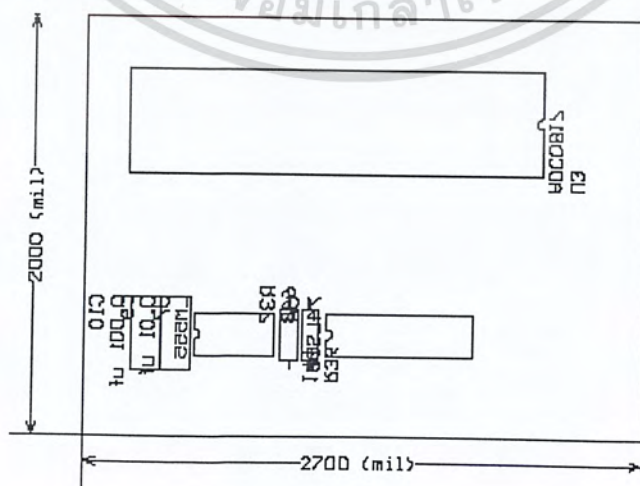
รูปที่ ข.1 สถิติของแผ่นปรี้นต์ที่ 1



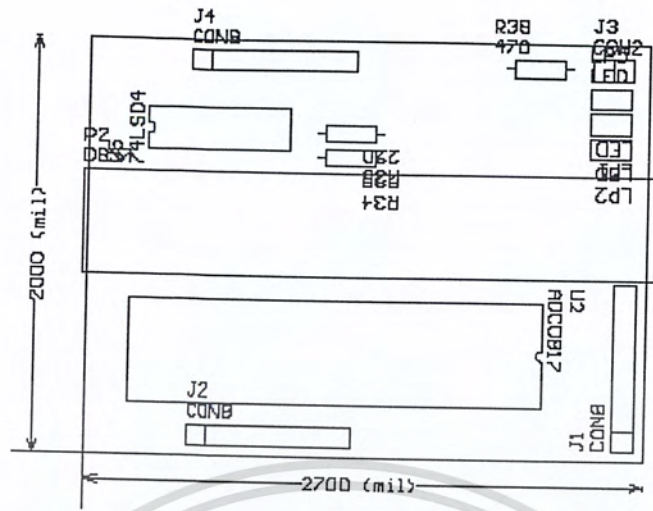
รูปที่ ๒.2 Bottom Layer



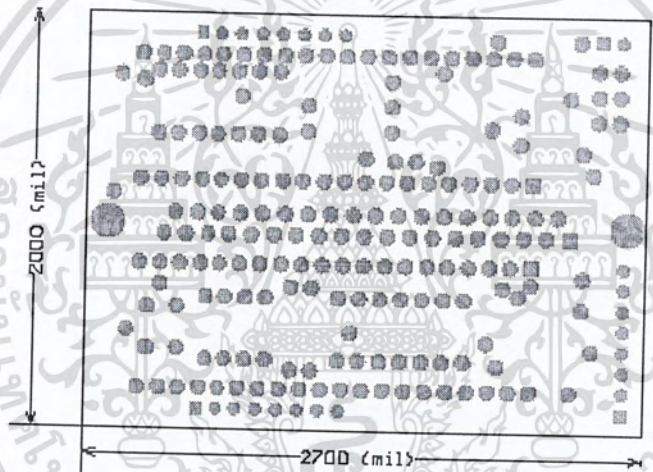
รูปที่ ๒.3 Top Layer



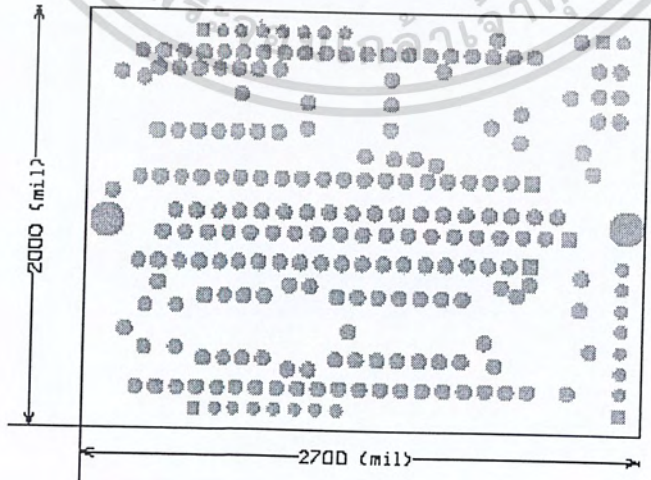
รูปที่ ๒.4 Bottom Silkscreen Overlay



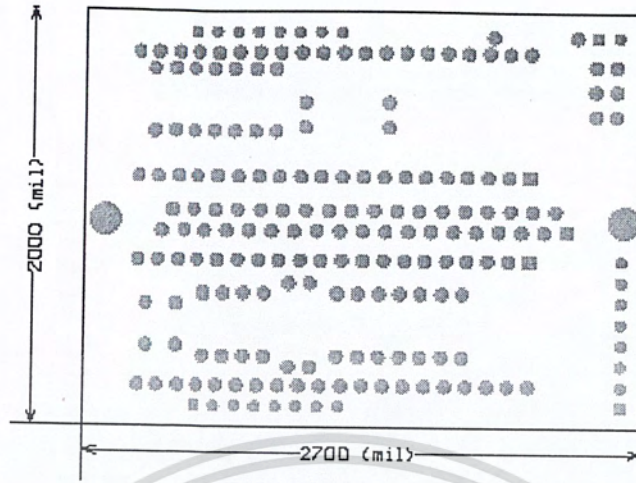
รูปที่ ข.5 Top Silkscreen Overlay



รูปที่ ข.6 Top Solder Mask Print



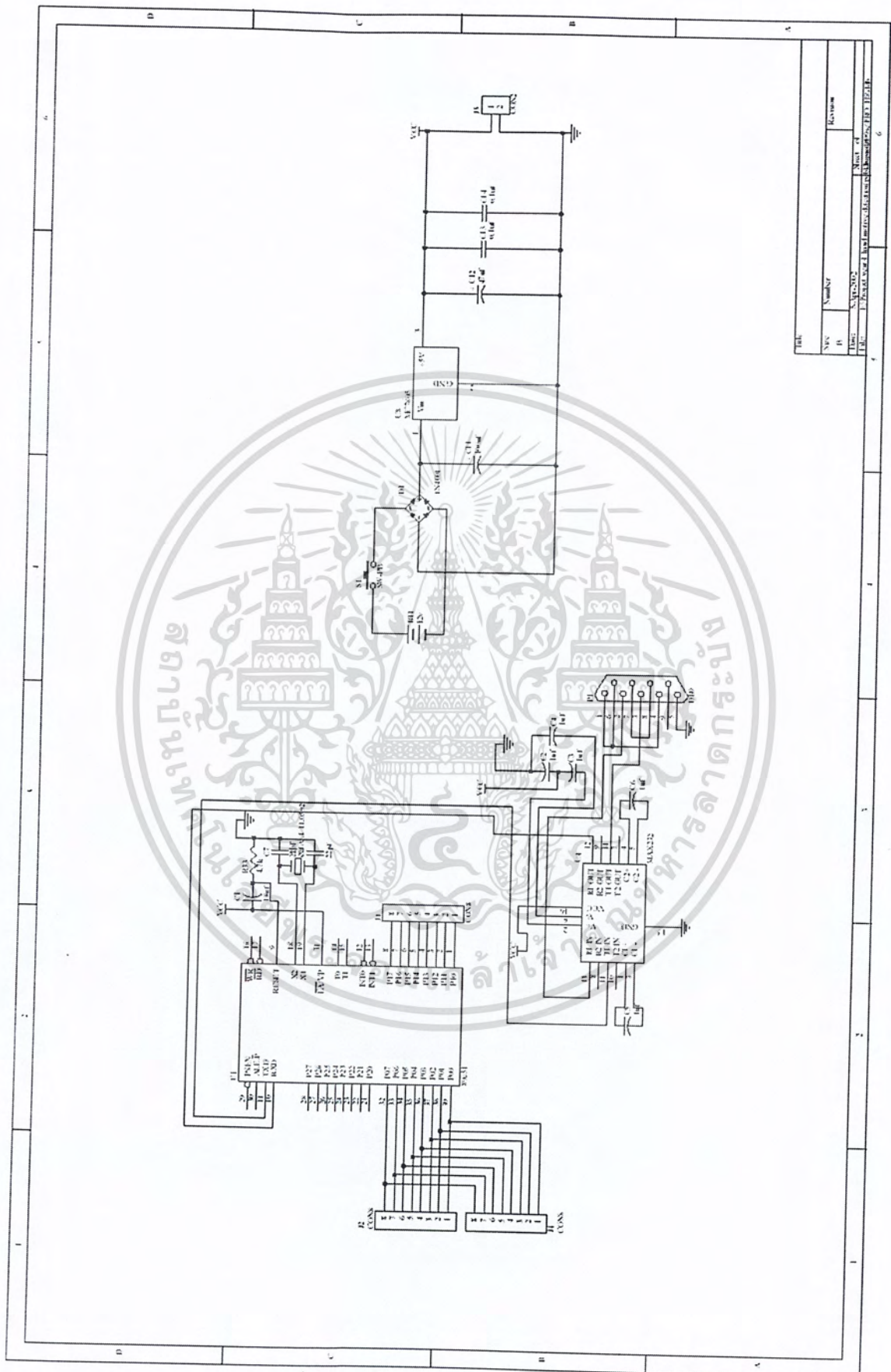
รูปที่ ข.7 Bottom Solder Mask Print



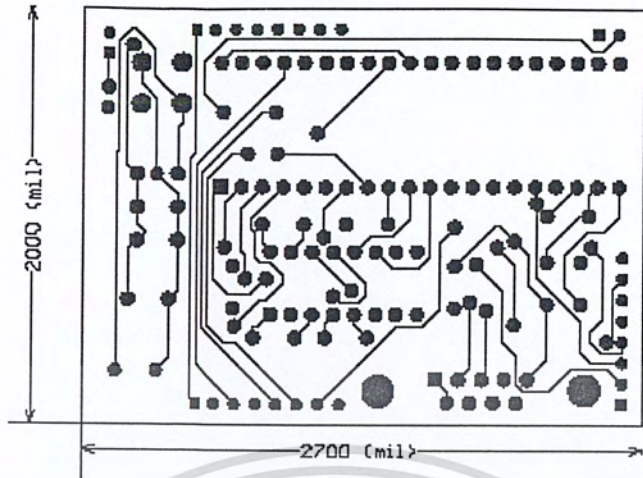
รูปที่ ข.8 Top Pad Master



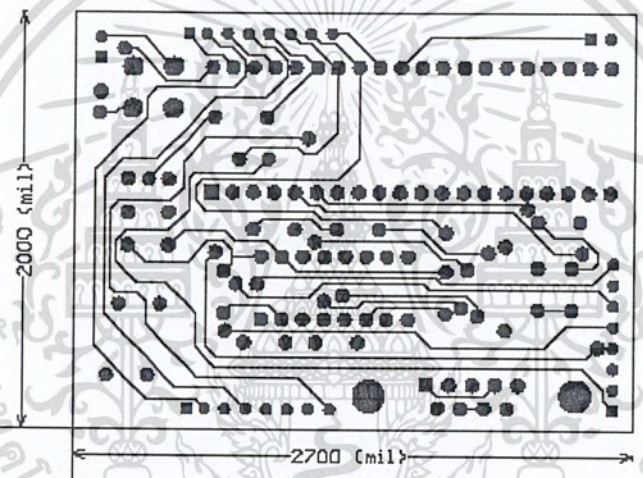
รูปที่ ข.9 Bottom Pad Master



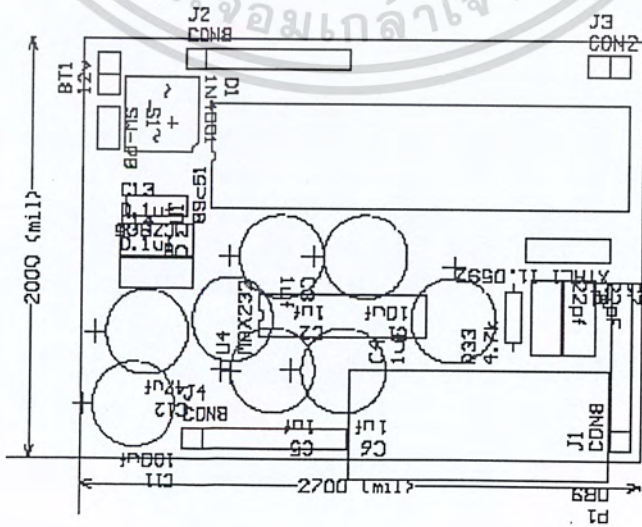
รูปที่ ข.10 สถิติเมตริกของแผ่นปรี้นแผ่นที่ 2



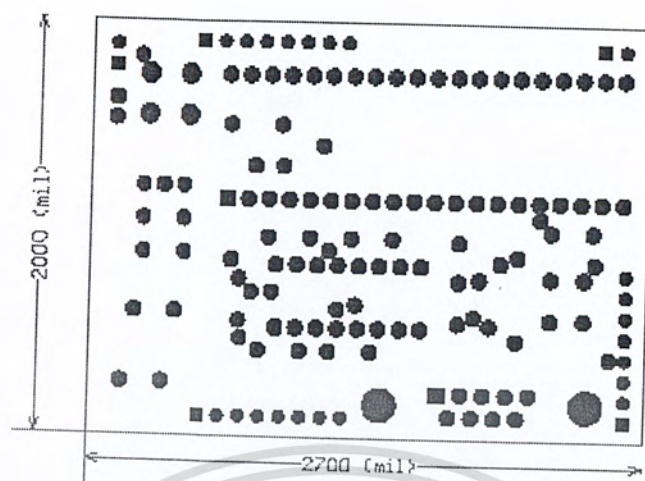
รูปที่ ข.11 Bottom Layer



รูปที่ ข.12 Top Layer



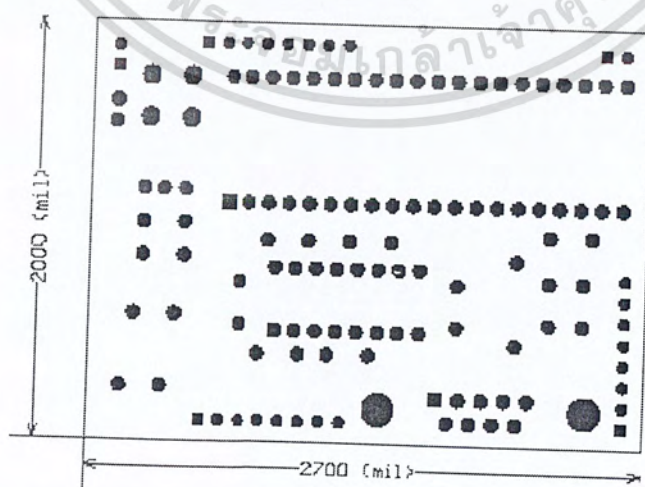
รูปที่ ข.13 Top Silkscreen Overlay



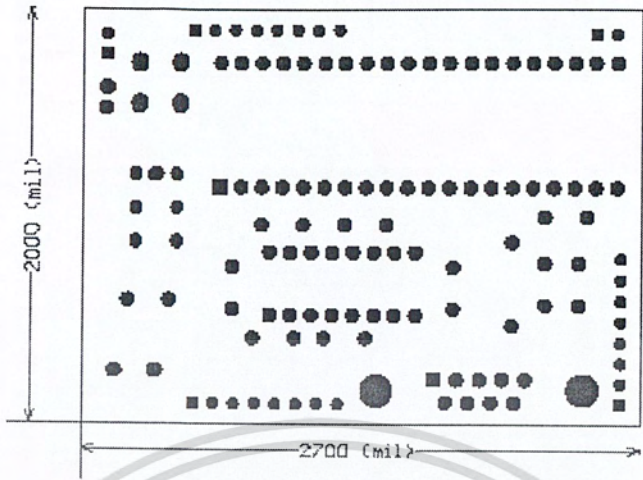
รูปที่ ข.14 Top Solder Mask Print



รูปที่ ข.15 Bottom Solder Mask Print



รูปที่ ข.16 Top Pad Master



รูปที่ ข.17 Bottom Pad Master



บรรณานุกรม

- [1] นิตยสาร @DEV : issue Vol.1 No.5 october 2000
- [2] ชัยวัฒน์ คำรัตน์: 3d game programming with DIRECTX เล่ม 1, มกราคม, 2542
- [3] ชัยวัฒน์ คำรัตน์: 3d game programming with DIRECTX เล่ม 2, เมษายน, 2542
- [3] ชีรวัฒน์ ประกอบผล: การประยุกต์ใช้งานไมโครคอนโทรลเลอร์, 235 หน้า, 2543

