สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เกมช่วยเหลือคนหัดเดิน

Movement Improving Game for Handicaps

โดย

นายชาติ์นนท์ เกตุแก้ว

นายบุญชัย จินดาถาวรกิจ

เลขหมู่ ..... TK 7871 .I58 2548

เลขทะเบียน c2853

วัน,เดือน,ปี ..... 23 ส.ค. 2549

b 11632860

i .....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาตรีวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOVEMENT IMPROVING GAME FOR HANDICAPS

BY

MR. CHANON KETKAEW

MR. BOONCHAI JINDATHAWORNKIJ

A THESIS SUBMITTED IN FULLFILLMENT OF

THE REQUIRMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2005

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เกมช่วยเหลือคนหัดเดิน	
ชื่อนักศึกษา	นายชาติ์นนท์ เกตุแก้ว	รหัสนักศึกษา 45010182
	นายบุญชัย จินดาถาวรกิจ	รหัสนักศึกษา 45010425
อาจารย์ที่ปรึกษา	รศ. ดร. ปิติเขต สู้รักษา	
	ผศ. บุณย์ชนะ ภู่ระหงษ์	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ภาควิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2548	

ปริญญานิพนธ์นี้ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง โดยได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเรียบร้อยแล้ว

.....

(รศ.ดร. ปิติเขต สู้รักษา)

.....

(ผศ.บุณย์ชนะ ภู่ระหงษ์)

อาจารย์ผู้ควบคุมปริญญานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	**เกมช่วยเหลือคนหัดเดิน**	
ชื่อนักศึกษา	นายชาตินนท์ เกตุแก้ว	รหัสนักศึกษา 45010182
	นายบุญชัย จินดาถาวรกิจ	รหัสนักศึกษา 45010425
อาจารย์ที่ปรึกษา	รศ.ดร. ปิติเขต สู้รักษา	
	ผศ.บุณย์ชนะ ภู่ระหงษ์	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ปีการศึกษา	2548	

บทคัดย่อ

ปริญญานิพนธ์นี้กล่าวถึงการพัฒนาเกม สำหรับนำมาใช้กับเครื่องช่วยเดิน เพื่อให้ผู้ทำกายภาพบำบัดโดยการเดินได้เล่นในระหว่างการฝึกเดิน เพื่อไม่ให้รู้สึกเบื่อหน่าย และมีสิ่งกระตุ้นให้รู้สึกอยากทำการกายภาพบำบัด โดยในเกมจะเป็นการจำลองสนามวิ่งมาซ้อนทับกับภาพจริงที่กล้องจับได้ ผู้ป่วยจะใช้เครื่องช่วยเดินสำเร็จรูปแบบ 2 ล้อที่ทำการติดตั้งอุปกรณ์เพิ่มเติม และมีจอมอนิเตอร์ไว้สำหรับเล่นเกม ภาพที่ผู้ป่วยเห็นในจอมอนิเตอร์จะเป็นภาพของสถานที่จริงที่ผู้ป่วยกำลังหัดเดินอยู่ โดยผู้ป่วยจะทำการเดินให้ตรงเส้นทางที่กำหนดจึงจะได้แต้ม แต่หากเดินออกนอกเส้นทางแต้มก็จะลดลงตามระยะจนกว่าจะจบเกมถึงโดยการเข้าเส้นชัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Movement Improving Game for Handicaps
Student	Mr. Chanon Ketkaew ID. 45010182
	Mr. Boonchai Jindathawornkij ID. 45010425
Advisor	Asso. Prof. Dr.Pitikhate Sooraksa
	Asst. Prof. Boonchana Phurahong
Graduate Level	Bachelor Degree of Information Engineering
Department	Information Engineering
Academic Year	2005

ABSTRACT

This thesis is descript computer game for gait trainer apparatus for patient can walk in Physical therapy method and play a computer game for funny and concentrate in walking. The game is a Computer simulation is received a real world image from web camera and mixing with a walking track that generated by a computer. Patient use gait trainer apparatus with 2 wheels and optional equipments such as Monitor Screen and 2 Sensors. In the game, a patient must walks along to track lineally and get a point. In other word, if the walking is out of track, the received points are decreases.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี เพราะได้รับความช่วยเหลือและความร่วมมือจากหลายฝ่ายด้วยกัน โดยมี ผศ.บุณย์ชนะ ภู่ระหงษ์ อาจารย์ที่ปรึกษาซึ่งเป็นผู้ริเริ่มโครงงานนี้ คอยช่วยเหลือให้คำแนะนำที่ดีตลอดมา รวมทั้งเพื่อนๆภาควิศวกรรมสารสนเทศทุกๆคนที่คอยเป็นกำลังใจและแนะนำแนวทางเมื่อพบเจออุปสรรคในการทำโครงงาน ส่งผลให้โครงงานผ่านอุปสรรคมาได้ด้วยดี

ท้ายที่สุดนี้ขอกราบขอบพระคุณบุคคลที่สำคัญที่สุดที่ทำให้มีวันนี้ คือ คุณพ่อ-คุณแม่ที่คอยให้กำลังใจส่งเสียเลี้ยงดูสนับสนุนในทุกๆด้าน รวมถึงครูบาอาจารย์ทุกท่านที่สนับสนุนประสิทประสาทวิชาความรู้ จึงขอกราบขอบพระคุณมา ณ ที่นี้ด้วย

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ฉ
สารบัญตาราง	ซ
บทที่ 1 บทนำ	
1.1 แนวคิดและที่มาของปัญหา	1
1.2 วัตถุประสงค์ของโครงงาน	1
1.3 ขอบเขตของโครงงาน	2
1.4 ผลที่คาดว่าจะได้รับ	2
1.5 ขั้นตอนการทำงาน	2
1.6 อุปกรณ์ที่ต้องใช้	3
บทที่ 2 ทฤษฎีและหลักการ	
2.1 การสื่อสารแบบอนุกรม	5
2.2 ดีซี มอเตอร์ เอนโค้ดเดอร์ (DC Motor Encoder)	9
2.3 การทำงานในโหมด Pulse Width Modulation ของวงจร Programmable Counter Array (PCA) ในไมโครคอนโทรลเลอร์ T89C51AC2	11
2.4 วงจร Analog-to-Digital (ADC) ในไมโครคอนโทรลเลอร์ T89C51AC2	14
บทที่ 3 การออกแบบ	
3.1 โครงสร้างโดยรวม	19
3.2 การออกแบบส่วนฮาร์ดแวร์	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
3.3 การออกแบบส่วนซอฟแวร์	22
3.4 การออกแบบส่วน User Interface	24
บทที่ 4 ผลการทดลอง	
4.1 ผลการทดลองการนับค่า pulse ของเซนเซอร์ซ้ายและขวา จากการหมุนของมอเตอร์ด้วยโปรแกรม Serial Watcher	26
4.2 ขั้นตอนการเล่นเกม	27
4.3 ผลการทดลองจากเกม	30
บทที่ 5 บทสรุปและวิจารณ์	
5.1 สรุปผลการดำเนินงาน	31
5.2 ปัญหาที่เกิดและแนวทางแก้ไข	31
5.3 ข้อจำกัดของโครงงาน	31
5.4 แนวทางการพัฒนาต่อ	32
บรรณานุกรม	33
ภาคผนวก	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

ภาพ	หน้า
รูปที่ 2.1 แสดงโครงสร้างโดยรวม	4
รูปที่ 2.2 ลักษณะของคอนเน็กเตอร์แบบ D-Type	6
รูปที่ 2.3 แสดงการเปรียบเทียบคอนเน็กเตอร์แบบ DB25 และ DB9	7
รูปที่ 2.4 แสดงoptical encoder	10
รูปที่ 2.5 แสดง magnetic encoder	10
รูปที่ 2.6 แสดงลักษณะ PCA Timer	12
รูปที่ 2.7 แสดง PCA ในโหมด PWM	13
รูปที่ 2.8 แสดงวงจร ADC ในไมโครคอนโทรลเลอร์ T89C51AC2	14
รูปที่ 2.9 แสดงลักษณะของ ADCF Register	16
รูปที่ 2.10 แสดงลักษณะของ ADCLK Register	16
รูปที่ 2.11 แสดงลักษณะของ ADCON Register	17
รูปที่ 2.12 แสดงลักษณะของ ADDH Register	17
รูปที่ 2.13 แสดงลักษณะของ ADDL Register	18
รูปที่ 3.1 แสดงผังโครงสร้างโดยรวม	19
รูปที่ 3.2โครงสร้างเครื่องช่วยเดินสำเร็จรูปด้านหน้าและด้านข้าง	20
รูปที่ 3.3โครงสร้างเครื่องช่วยเดินติดตั้งมอเตอร์ และแท่นวางคอมพิวเตอร์โน๊ตบุ๊คด้านหน้าและด้านข้าง	21
รูปที่ 3.4 แสดงผังการออกแบบวงจรรวม	21
รูปที่ 3.5 รูปถ่ายแผงวงจรไมโครคอนโทรลเลอร์	22
รูปที่ 3.6 แสดง Flow Chart การทำงานของไมโครคอนโทรลเลอร์	22
รูปที่ 3.7 แสดง Flow Chart การทำงานของเกม	23
รูปที่ 3.8 หน้าจอ Select Profile	24
รูปที่ 3.9 หน้าจอ Create New Profile	24
รูปที่ 3.10 หน้าจอหลัก Advance Motion The Game	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

ภาพ	หน้า
รูปที่ 4.1 แสดงการนับ pulse จากเซนเซอร์ด้านซ้ายและขวา	26
รูปที่ 4.2 หน้าจอ Select Profile	27
รูปที่ 4.3 หน้าจอ Create New Profile กรอกข้อมูลผู้เล่นใหม่	27
รูปที่ 4.4 แสดงการเลือกสนาม	28
รูปที่ 4.5 ลักษณะสนามจำลองในเกม	28
รูปที่ 4.6 แสดงการเลือกโหมด	29
รูปที่ 4.7 หน้าจอหลักของเกม	29
รูปที่ 4.8 แสดงการเดินตามเส้นทางที่กำหนด	30
รูปที่ 4.9 แสดงการเดินออกนอกเส้นทาง	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ภาพ	หน้า
ตารางที่ 2.1 แสดงรายละเอียดของสายสัญญาณ RS-232	7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 แนวคิดและที่มาของปัญหา

ในปัจจุบันการทำกายภาพบำบัด เป็นสิ่งที่ผู้ป่วยต้องใช้ความพยายามและความอดทนอย่างมาก เนื่องจากต้องฝึกซ้ำๆแบบเดิมเป็นเวลานาน อีกทั้งยังต้องทนอึดอัดกับสภาพร่างกายของตัวเอง จึงมีหลายครั้งที่ผู้ป่วยบางรายเลิกทำกายภาพบำบัด หรือมาทำไม่สม่ำเสมอ จนทำให้อวัยวะส่วนที่ต้องเข้ารับการบำบัดไม่สามารถกลับมาทำงานได้อย่างปกติ โครงงานนี้จึงได้เกิดขึ้นมาด้วยแนวคิดที่ว่า หากเราสามารถทำให้ผู้ป่วยที่เข้ารับการทำกายภาพบำบัด รู้สึกสนุกไปกับการทำกายภาพบำบัดก็จะเป็นการกระตุ้นให้ผู้ป่วยอยากทำกายภาพบำบัด โดยโครงงานนี้จะเป็นการช่วยเหลือผู้ที่ทำกายภาพบำบัดด้วยวิธีการเดิน

โดยที่โครงงานจะประกอบไปด้วยส่วนที่เป็นเครื่องช่วยเดินสำหรับให้ผู้ทำกายภาพบำบัดได้พยุงตัว และในเครื่องช่วยเดินนี้จะมีจอมอนิเตอร์และหน่วยประมวลผลกลาง(CPU) สำหรับให้ผู้ทำกายภาพบำบัดได้เล่นเกม เพื่อไม่ให้รู้สึกเบื่อหน่ายในขณะฝึกเดิน

1.2 วัตถุประสงค์ของโครงงาน

1.2.1เพื่อศึกษาทฤษฎีและวิธีใช้งานอุปกรณ์ฮาร์แวร์(hardware)ซึ่งมีดังนี้

- ดีซี มอเตอร์ ออพติคอล เอนโค้ดเดอร์ (DC Motor Optical Encoder)
- ไมโครคอนโทรลเลอร์ (Microcontroller)
- กล้องวีดีโอ

1.2.2 เพื่อศึกษาทฤษฎีในการรับ-ส่งข้อมูลผ่านพอร์ตอนุกรม (serial port)

1.2.3 เพื่อสามารถพัฒนาซอฟแวร์ (software) ในการควบคุมไมโครคอนโทรลเลอร์ ให้สามารถควบคุมการรับส่งข้อมูลระหว่างอุปกรณ์ตัวอื่นๆได้

1.2.4 เพื่อสามารถพัฒนาซอฟแวร์ ในการประมวลผลข้อมูลที่รับมาผ่านพอร์ตอนุกรม แล้วนำข้อมูลนั้นมาแสดงผลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน

เป็นการศึกษาทฤษฎีและพัฒนาซอฟแวร์ในการรับ-ส่งข้อมูลระหว่างอุปกรณ์ แล้วนำข้อมูลมาประมวลผลเพื่อพัฒนาโปรแกรมประยุกต์ที่มีลักษณะเป็นเกมเพื่อให้ผู้เข้ารับการทำกายภาพบำบัดได้เล่น เพื่อลดความเบื่อหน่ายและเพิ่มความสนุกสนานในการทำกายภาพบำบัดโดยแบ่งเป็นส่วนต่างๆดังนี้

1.3.1 ส่วนของซอฟแวร์ที่ใช้ในการควบคุมไมโครคอนโทรลเลอร์ให้สามารถส่งข้อมูลที่รับมาจากอุปกรณ์ต่างๆผ่านทางพอร์ตอนุกรม

1.3.2 ส่วนของซอฟแวร์ที่ใช้การประมวลผลข้อมูลที่รับมาทางพอร์ตอนุกรม แล้วนำมาแสดงผลในโปรแกรมเกม

1.3.3 ส่วนของการทำเครื่องช่วยเดิน

1.4 ผลที่คาดว่าจะได้รับ

1.4.1 สามารถพัฒนาซอฟแวร์ที่ใช้ในการรับ-ส่งข้อมูลระหว่างอุปกรณ์ต่างๆกับไมโครคอนโทรลเลอร์ได้

1.4.2 เรียนรู้เทคโนโลยีการพัฒนาโปรแกรม โดยใช้โปรแกรมMicrosoft Visual Basic 6 ในการประมวลผลข้อมูลที่รับมาจากพอร์ตอนุกรมแล้วนำมาแสดงผลในโปรแกรมได้

1.4.3 สามารถนำโครงงานไปใช้กับผู้ป่วยที่ทำกายภาพบำบัดโดยการเดิน เพื่อเพิ่มความสนุกสนานในการทำกายภาพบำบัด

1.5 ขั้นตอนการทำงาน

1.5.1 ศึกษาทฤษฎีการทำงานของอุปกรณ์ในข้อ 2.1

1.5.2 ศึกษาทฤษฎีการรับ-ส่งข้อมูลผ่านพอร์ตอนุกรม

1.5.3 ศึกษาโปรแกรม Microsoft Visual Basic 6 และ Keil สำหรับควบคุมไมโครคอนโทรลเลอร์

1.5.4 ออกแบบและทำเครื่องช่วยเดิน

1.5.5 เขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์โดยใช้โปรแกรม Keil

1.5.6 เขียนโปรแกรมเกมโดยใช้โปรแกรมMicrosoft Visual Basic 6.0

1.5.7 ทดสอบและแก้ไขโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 อุปกรณ์ที่ต้องใช้

1.6.1 เครื่องคอมพิวเตอร์สำหรับพัฒนาโปรแกรมจำนวน 2 เครื่อง

1.6.2 ชุดอุปกรณ์ที่ใช้ทำเครื่องช่วยเดิน

1.6.3 ดีซี มอเตอร์ ออพติคอล เอนโค้ดเดอร์

1.6.4 ไฟโต้เซ็นเซอร์แบบก้ามปู

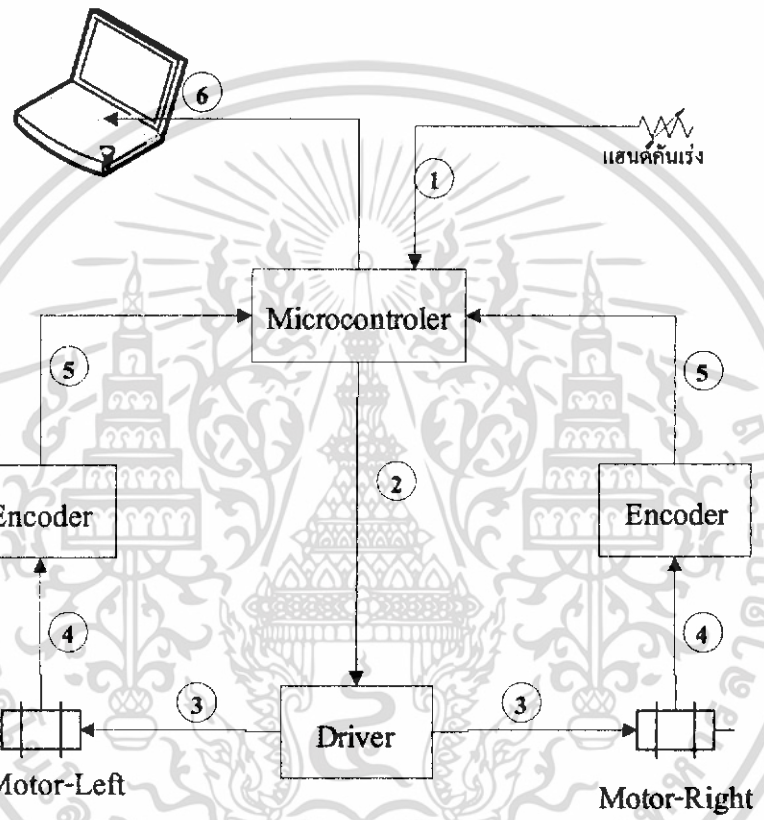
1.6.5 ชุดอุปกรณ์สำหรับไมโครคอนโทรลเลอร์

1.6.6 กล้องวีดีโอ(Webcam)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

ในบทนี้จะกล่าวถึงทฤษฎีเบื้องต้นต่างๆ ที่ใช้เป็นแนวคิดพื้นฐานในการออกแบบโครงงาน โดยขอกล่าวถึงขั้นตอนการทำงานโดยรวมซึ่งเป็นดังนี้



รูปที่ 2.1 แสดงโครงสร้างโดยรวม

ขั้นตอนตามหมายเลขต่างๆ แสดงได้ดังนี้

(1) รับค่าแรงดันไฟฟ้ามาจากแฮนด์คันเร่งเร่งความเร็วเพื่อนำไปเข้าสู่วงจร ADC ในไมโครคอนโทรลเลอร์

(2) ไมโครคอนโทรลเลอร์ส่งค่าที่ได้ออกมาไปยัง Driver เพื่อขับมอเตอร์

(3) Driver ทำการขับมอเตอร์

(4) ค่าการหมุน(จำนวนรอบ)ของมอเตอร์เข้าสู่วงจร Encoder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

** หมายเหตุ ค่าการหมุนของมอเตอร์อาจได้มาจากการหมุนของมอเตอร์โดยเชิงกล (ไม่ผ่าน Driver) ก็ได้

⑤ ค่าที่ได้จากการ Encoder ถูกส่งไปประมวลผลที่ไมโครคอนโทรลเลอร์

⑥ ไมโครคอนโทรเลอร์ส่งค่าที่ได้จากการ Encoder เข้าคอมพิวเตอร์โดยผ่านทางพอร์ตอนุกรม (RS-232) เพื่อประมวลผลต่อไป

ส่วนต่อไปจากนี้จะกล่าวถึงทฤษฎีเบื้องต้นหลักๆ ที่ใช้ในโครงงานที่อยู่ภายในโครงสร้างโดยรวมข้างต้นมีดังนี้

2.1 การสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรมในเครื่องคอมพิวเตอร์นั้นจะมีความเร็วในการสื่อสารน้อยกว่าแบบขนาน เพราะว่าการเคลื่อนย้ายข้อมูลแบบอนุกรมนั้นเป็นการส่งข้อมูลครั้งละ 1 บิต แต่พอร์ตขนานนั้นสามารถส่งข้อมูลได้ครั้งละหลายๆบิตพร้อมกัน แต่การส่งข้อมูลแบบอนุกรมนั้นจะสามารถส่งข้อมูลได้ในระยะทางที่ไกลกว่าแบบขนาน อีกทั้งสายสัญญาณที่ใช้ยังมีน้อยกว่าแบบขนาน โดยการสื่อสารแบบอนุกรมสามารถแบ่งออกเป็น 3 รูปแบบดังนี้

1.Simplex สามารถส่งข้อมูลได้อย่างเดียว เป็นการสื่อสารแบบทางเดียว

2.Half-Duplex สามารถส่งข้อมูลไปยังปลายทางและรับข้อมูลจากปลายทางได้ แต่ไม่สามารถทำการส่งและรับข้อมูลได้ในเวลาเดียวกัน

3.Full-Duplex สามารถส่งและรับข้อมูลได้ในเวลาเดียวกัน

นอกจากนี้แล้วยังสามารถแบ่งประเภทของการสื่อสารแบบอนุกรมตามลักษณะสัญญาณในการส่งได้อีก 2 แบบคือ

- **การสื่อสารแบบซิงโครนัส** (Synchronous) จะใช้สัญญาณนาฬิกาควบคุมการรับส่งสัญญาณ เช่น สายคีย์บอร์ดคอมพิวเตอร์ โดยจะมีสายสัญญาณเส้นหนึ่งเป็นสายสัญญาณนาฬิกา ส่วนอีกเส้นหนึ่งเป็นสายของข้อมูล (และมักจะมีสายกราวน์ด้วย)
- **การสื่อสารแบบอะซิงโครนัส** (Asynchronous) จะใช้สายข้อมูลเพียงตัวเดียว แต่จะแต่จะใช้รูปแบบการส่งข้อมูล หรือ Bit Pattern เป็นตัวกำหนดว่าส่วนไหนเป็นส่วนเริ่มต้นข้อมูล, ส่วนไหนเป็นตัวข้อมูล, ส่วนไหนเป็นส่วนตรวจสอบความถูกต้องของข้อมูล และส่วนไหนเป็นส่วนปิดท้ายของข้อมูล โดยต้องกำหนดให้สัญญาณนาฬิกาเท่ากันทั้งภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และภาครับ ซึ่งจะมีอุปกรณ์พิเศษที่ชื่อว่า UART หรือ Asynchronous Receiver/Transmitter คอยควบคุมการรับและส่งข้อมูล

2.1.1 มาตรฐาน RS-232C

มาตรฐาน RS-232C เป็นมาตรฐานที่ได้รับการออกแบบเพื่อที่จะทำให้อุปกรณ์ต่อพ่วงจากผู้ผลิตต่างกันสามารถทำงานร่วมกันได้ ทั้งที่มาตรฐานหลายชนิดได้รับการออกแบบขึ้นมา แต่มาตรฐาน RS-232C ได้รับความนิยมและใช้กันกว้างขวางมากที่สุด โดยถูกประกาศใช้ในปี 1969 โดยสมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA)

มาตรฐาน RS-232C ได้แบ่งอุปกรณ์ออกเป็น 2 ประเภท คือ

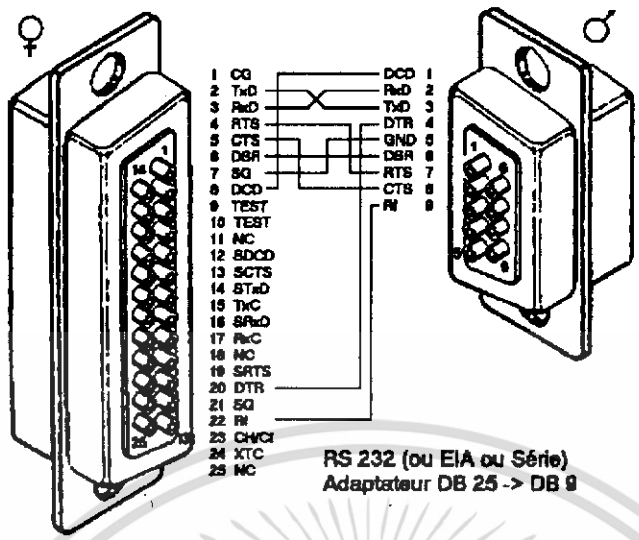
1. อุปกรณ์ DTE (Data Terminal Equipment) เป็นอุปกรณ์สำหรับส่งข้อมูล (Output)
2. อุปกรณ์ DCE (Data Communication Equipment) เป็นอุปกรณ์สำหรับรับข้อมูล (Input)

ตามมาตรฐาน RS-232C แล้วคอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งคอนเน็กเตอร์ที่นิยมใช้กันอยู่จะเป็นชนิด D-Type แบบ 9 ขา และแบบ 25 ขา โดยจะติดตั้งอยู่หลังเครื่องคอมพิวเตอร์ ระดับแรงดันจะมีค่าระหว่าง -3 V ถึง -15 V

รูปที่ 2.2 ลักษณะของคอนเน็กเตอร์แบบ D-Type

หัวต่อแบบ D-Type ที่ใช้ในการสื่อสารแบบอนุกรมของเครื่องคอมพิวเตอร์นั้น จะมีอยู่ 2 ลักษณะคือ แบบ 9 ขา และแบบ 25 ขา หรืออาจจะเรียกว่า DB9 และ DB25 ซึ่งทั้งสองจะมีลักษณะการทำงานของสัญญาณเหมือนกัน แต่การจัดเรียงไม่เหมือนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงการเปรียบเทียบคอนเน็กเตอร์แบบ DB25 และ DB9

D-Type 25 Pin	D-Type 9 Pin	สัญลักษณ์	ชื่อสัญญาณ
Pin 2	Pin 3	TD	Transmit Data
Pin 3	Pin 2	RD	Receive Data
Pin 4	Pin 7	RTS	Request To Send
Pin 5	Pin 8	CTS	Clear To Send
Pin 6	Pin 6	DSR	Data Set Ready
Pin 7	Pin 5	SG	Signal Ground
Pin 8	Pin 1	CD	Carrier Detect
Pin 20	Pin 4	DTR	Data Terminal Ready
Pin 22	Pin 9	RI	Ring Indicator

ตารางที่ 2.1 แสดงรายละเอียดของสายสัญญาณ RS-232

รายละเอียดของสายสัญญาณ

- Transmit Data : TD ใช้สำหรับส่งข้อมูลอนุกรมออกจากคอมพิวเตอร์
- Receive Data : RD ใช้สำหรับรับข้อมูลอนุกรมเข้ามายังคอมพิวเตอร์
- Request To Send : RTS ใช้สำหรับส่งข้อมูลไปยังอุปกรณ์ปลายทาง เพื่อร้องขอให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ปลายทางส่งข้อมูลกลับมา

- Clear To Send : DTS ใช้สำหรับตรวจสอบว่าอุปกรณ์ที่เชื่อต่อด้วยพร้อมที่จะรับข้อมูลหรือไม่ โดยจะคอยรับสัญญาณ RTS เมื่อทุกอย่างพร้อมก็จะทำการส่งข้อมูลออกทางขา TD
- Data Set Ready : DSR ใช้สำหรับตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง จะใช้คู่กับขา DTR
- Signal Ground : SG เป็นกราวน์ของระบบ
- Carrier Detect : CD ขานี้จะ Active เมื่อมีการส่งสัญญาณ Carrier จากโมเด็ม
- Data Terminal Ready : DTR ใช้สำหรับบอกให้อุปกรณ์ปลายทางรับรู้ว่าต้องการติดต่อโดยขา DTR นี้ต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง
- Ring Indicator : RI ขานี้จะ Active เมื่อโมเด็มได้รับสัญญาณเรียกเข้าจากสายโทรศัพท์

2.1.2 องค์ประกอบของการรับส่งข้อมูลแบบอนุกรม

การสื่อสารแบบอนุกรมที่นิยมใช้กับคอมพิวเตอร์นั้น เป็นการสื่อสารแบบอะซิงโครนัส นั่นคือ ต้องใช้สายสัญญาณเส้นเดียวทำหน้าที่ทั้งส่งส่วนที่เป็นข้อมูล และส่วนที่ใช้ควบคุมการส่งข้อมูล ดังนั้นข้อมูลที่อ่านได้แต่ละบิตจากการส่งแบบอนุกรม จึงต้องถูกแยกแยะว่าใช้สำหรับวัตถุประสงค์ใด โดยเราสามารถแบ่งได้เป็น 4 ส่วนคือ

1. Start Bit ขนาด 1 บิต
2. บิตข้อมูล (Data Character) ขนาด 7 หรือ 8 บิต
3. Parity Bit ขนาด1 บิต
4. Stop Bit ขนาด 1 หรือ 2 บิต

แต่ละตัวอักษรที่ถูกส่งออกไปเป็นกลุ่มจะประกอบไปด้วยบิตเริ่มต้น บิตข้อมูล บิตพาริตี้ (จะมีหรือไม่มีก็ได้) และบิตจบ โดยพอสรุปหน้าที่ของแต่ละส่วนได้ดังนี้

- Start Bit หรือบิตเริ่มต้น จะใส่ที่จุดเริ่มต้นเสมอ เพื่อเตือนอุปกรณ์ฝ่ายรับว่าข้อมูลกำลังจะมาถึง
- Data Character หรือบิตข้อมูล การส่งบิตข้อมูลจะส่งเป็นกลุ่มๆ โดยทั่วไปจะส่งเป็น 7 บิต หรือ 8 บิต ซึ่งเพียงพอสำหรับการส่ง Ascii Word

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Parity Bit หรือบิตพาริตี้ ใช้ในการตรวจสอบความถูกต้องของข้อมูลที่ส่ง โดยจะใส่บิตพาริตี้เข้าไป แต่ทั้งตัวรับและตัวส่งจะต้องรู้กันว่าใช้บิตพาริตี้แบบไหนในการส่งข้อมูล ซึ่งหลักการในการกำหนดบิตพาริตี้มีหลายแบบดังนี้

  - พาริตี้คู่ (Even Parity) ค่าของบิตพาริตี้นี้เมื่อรวมกับทุกๆบิตของข้อมูลแล้ว จะต้องมีจำนวนบิตที่เป็นเลข 1 เป็นเลขคู่ ตัวอย่างเช่น ข้อมูล 1000101 มีเลข 1 ทั้งหมด 3 ตัว ดังนั้นบิตพาริตี้นี้จะเป็น 0

  - พาริตี้คี่ (Odd Parity) ค่าของบิตพาริตี้นี้เมื่อรวมกับทุกๆบิตของข้อมูลแล้ว จะต้องมีจำนวนบิตที่เป็นเลข 1 เป็นเลขคู่ ตัวอย่างเช่น ข้อมูล 1000101 มีเลข 1 ทั้งหมด 3 ตัว ดังนั้นบิตพาริตี้นี้จะเป็น 1

  - ไม่มีพาริตี้ (None) ถ้าตั้งบิตพาริตี้เป็น None ทั้งภาครับและภาคส่งจะไม่มีการตรวจสอบบิตพาริตี้

- Stop Bit หรือบิตจบ เป็นบิตที่ส่งมาปิดท้ายข้อมูล

2.1.3 อัตราเร็วในการรับส่งข้อมูลแบบอนุกรม

การที่อุปกรณ์ 2 อย่างจะติดต่อสื่อสารกันได้นั้น จะต้องทำงานด้วยอัตราเร็วที่เท่ากัน ซึ่งอัตราเร็วในการสื่อสารแบบอะซิงโครนัสคือ ค่าบอดเรต (Baud Rate) มีหน่วยเป็นบิตต่อวินาที ซึ่งค่าอัตราเร็วในการสื่อสารแบบอนุกรมสำหรับมาตรฐาน RS-232C นั้นมีใช้ตั้งแต่ 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200 บิตต่อวินาที

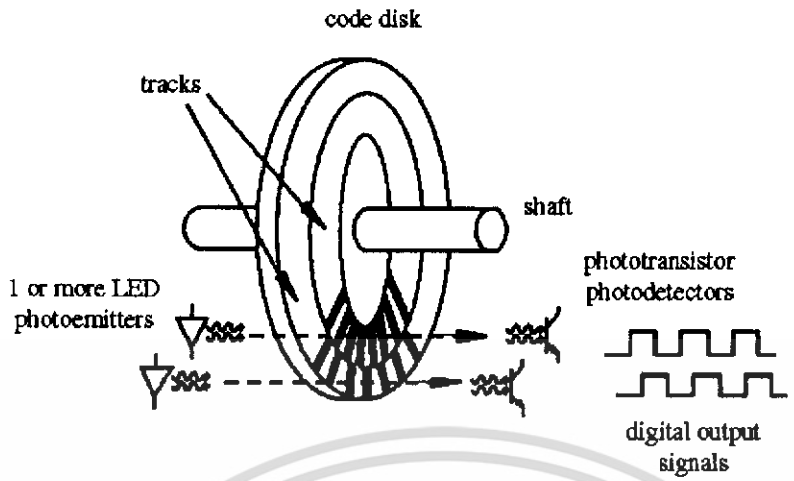
2.2 ดีซี มอเตอร์ เอนโค้ดเดอร์ (DC Motor Encoder)

ตัวเข้ารหัสที่ใช้กับมอเตอร์กระแสตรงนิยมใช้กันมี 2 แบบ คือ ตัวเข้ารหัสแบบใช้แสงและตัวเข้ารหัสแบบแม่เหล็ก

2.2.1 ตัวเข้ารหัสแบบใช้แสง (Opical Encoder)

ตัวเข้ารหัสชนิดนี้เป็นที่นิยมมากที่สุด ประด้วยส่วนต่างๆ คือ แผ่นจานที่หมุนได้ (rotating disk), แหล่งกำเนิดแสง (light source) และอุปกรณ์ตรวจจับแสง (photo detector) แผ่นจานจะติดอยู่กับเพลา (shaft) โดยที่ตัวแผ่นจานจะมีส่วนที่ทึบแสงและส่วนที่ถูกเจาะเป็นรูหรือวัสดุโปร่งแสงสลับกันไป ซึ่งจะเป็นตัวกำหนดรูปแบบของรหัส เมื่อแผ่นจานหมุนก็จะเกิดการรบกวนการตกกระทบของแสงบนอุปกรณ์ตรวจจับแสง ทำให้เกิดสัญญาณดิจิตอลหรือพัลส์ออกมา

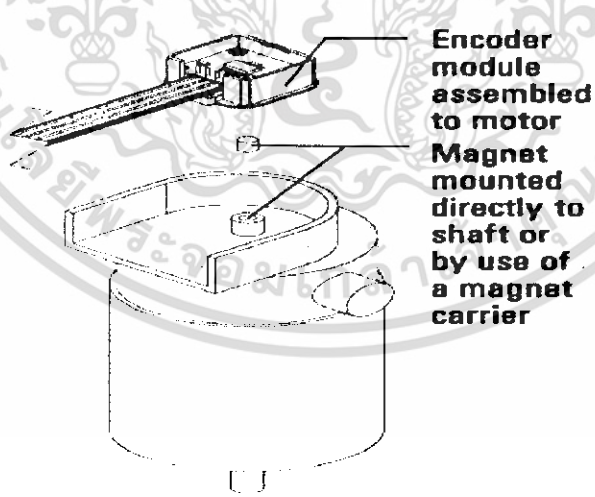
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 แสดงoptical encoder

2.2.2 ตัวเข้ารหัสแบบแม่เหล็ก (Magnetic Encoder)

ตัวเข้ารหัสชนิดนี้ประกอบด้วยเกียร์ซึ่งหมุนได้ ทำด้วยโลหะซึ่งมีธาตุเหล็ก และ magnetic pick-up ซึ่งเป็นแม่เหล็กถาวรและอุปกรณ์ตรวจจับ ส่วนของเกียร์จะติดอยู่กับเพลาซึ่งหมุนได้ และมีฟัน (gear tooth) ที่มีขนาดและระยะห่างเท่ากันอยู่ที่ขอบของเกียร์ซึ่งจะเป็นตัวกำหนดรูปแบบของสัญญาณที่ส่งออกมา เมื่อเกียร์หมุนจะทำให้ฟันหมุนเปลี่ยนไปเรื่อยๆ ซึ่งจะไปรบกวนสนามแม่เหล็กถาวรใน magnetic pick-up การเปลี่ยนแปลงของสนามแม่เหล็กนี้จะถูกตรวจจับโดยอุปกรณ์ตรวจจับแล้วเปลี่ยนให้เป็นสัญญาณส่งออกไป



รูปที่ 2.5 แสดง magnetic encoder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัว magnetic pick-up แบ่งออกเป็น 2 ชนิด ดังนี้

1. Hall effect อุปกรณ์ตรวจจับทำจากสารกึ่งตัวนำ โดยการทำงานขึ้นอยู่กับ hall effect พัลส์ที่ได้เกิดจากฟันของเกียร์แต่ละซี่ที่เคลื่อนที่ผ่านตัวตรวจจับ
2. Variable reluctance ตัวตรวจจับทำจากขดลวดที่พันกันไว้ในสนามแม่เหล็ก เมื่อฟันของเกียร์เคลื่อนที่ผ่านไปก็จะรบกวนสนามแม่เหล็ก ทำให้เกิดการเปลี่ยนแปลงของสนามแม่เหล็กขึ้น การเปลี่ยนแปลงนี้จะทำให้เกิดพัลส์ของแรงดันไฟฟ้าที่ขดลวดของตัวตรวจจับ โดยที่พัลส์ที่ได้จะขึ้นอยู่กับอัตราการเปลี่ยนแปลงของสนามแม่เหล็ก

โครงงานนี้เลือกใช้ตัวเข้ารหัสแบบใช้แสงเนื่องจากเมื่อแผ่นจานหมุน อุปกรณ์ตรวจจับแสงก็สามารถตรวจจับการเปลี่ยนแปลงได้ทันที ในขณะที่ตัวเข้ารหัสแบบแม่เหล็กนั้นแผ่นจานจะต้องหมุนด้วยความเร็วค่าหนึ่ง ซึ่งจะต้องเพียงพอที่จะทำให้เกิดการเปลี่ยนแปลงของสนามแม่เหล็ก อุปกรณ์ตรวจจับจึงจะสามารถตรวจพบ

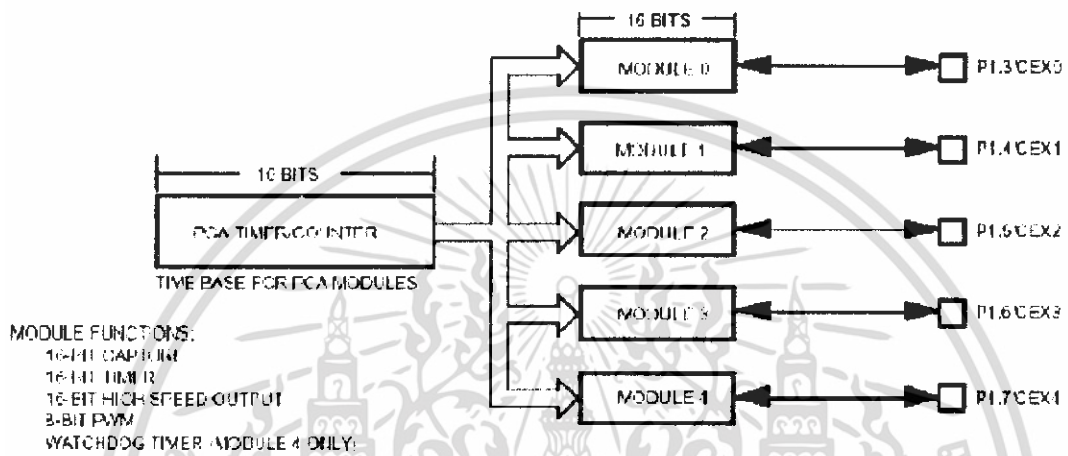
2.3 การทำงานในโหมด Pulse Width Modulation ของวงจร Programmable Counter Array (PCA) ในไมโครคอนโทรลเลอร์ T89C51AC2

ในโครงงานนี้เราเลือกใช้ไมโครคอนโทรลเลอร์ เบอร์ T89C51AC2 เนื่องจากมีคุณสมบัติพิเศษเพิ่มเติมของระบบ Timer นั่นก็คือวงจร Programmable Counter Array (PCA) ซึ่งตามปกติแล้วจะไม่มีอยู่ในโครงสร้างมาตรฐานของ CPU ในตระกูล MCS-51 ทั่วไป ซึ่งระบบการทำงานของ Timer/Counter แบบ PCA นี้จะมีขีดความสามารถที่สูงกว่าระบบการทำงานของ Timer/Counter ปรกติของ MCS-51 ปรกติ ซึ่งจะช่วยให้ลดความยุ่งยากของโปรแกรมในการสั่งงาน CPU และช่วยลดภาระการทำงานของ CPU ได้เป็นอย่างมาก โดยลักษณะ โครงสร้างของวงจร PCA จะประกอบไปด้วย ส่วนของวงจร Timer/Counter หลัก (PCA Timer) และเชื่อมต่อเข้ากับวงจรของ Capture/Compare (PCM Module) จำนวน 5 ชุด โดยเชื่อมต่อกันแบบเรียงลำดับ (Array) ซึ่งการทำงานของ PCA Module แต่ละชุดจะอ้างอิงความถี่จากวงจร PCA Counter ชุดเดียวกันหรือใช้ฐานเวลาร่วมกัน

ส่วนการทำงานของวงจร PCA Module ทั้ง 5 ชุดนั้น แต่ละโมดูลสามารถกำหนดหน้าที่การทำงานได้อย่างอิสระ โดยแต่ละโมดูลเองยังสามารถยังสามารถโปรแกรมการทำงานได้หลายๆหน้าที่ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

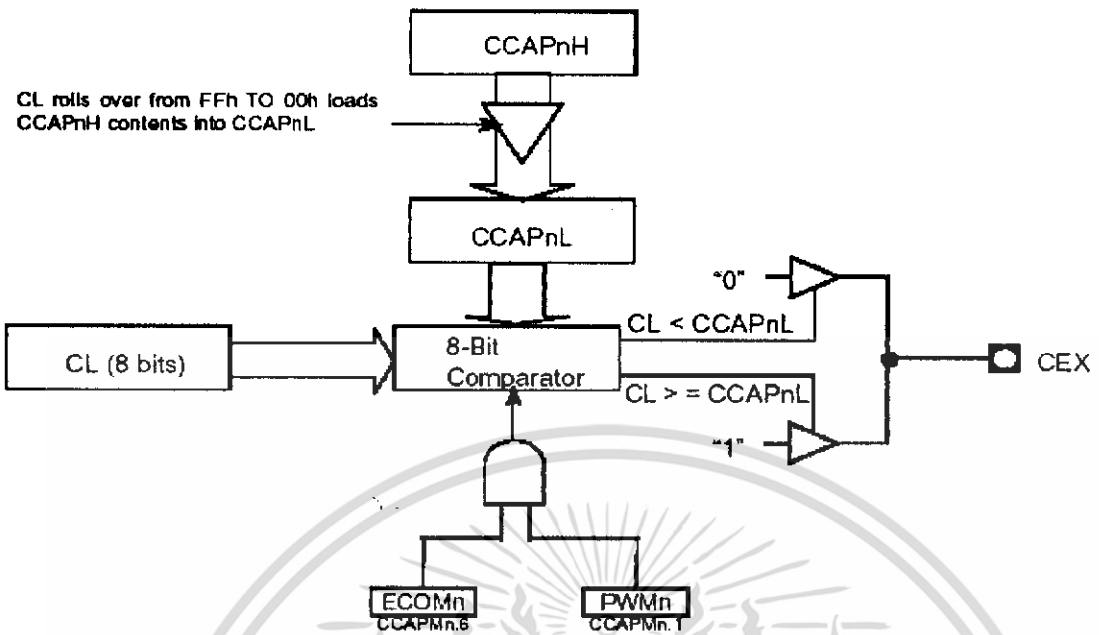
- กำหนดให้การตรวจจับสัญญาณ Input (Capture) โดยสามารถเลือกตรวจจับได้ทั้ง สัญญาณที่เป็นขอบขาขึ้น (Rising Edge) และขอบขาลง (Falling Edge)
- กำหนดให้ทำการสร้างสัญญาณ Output ความเร็วสูง แบบ High Speed Output
- กำหนดให้ทำการสร้างสัญญาณ Pulse Width Modulation หรือ PWM
- กำหนดให้ทำหน้าที่เป็น Timer สำหรับนำสัญญาณนาฬิกา (Software Timer)



รูปที่ 2.6 แสดงลักษณะ PCA Timer

โดยในโครงงานนี้เราเลือกใช้โมดูลที่สร้างสัญญาณ PWM เพื่อใช้ในการเร่งความเร็วของมอเตอร์ โดยในโหมดนี้ โมดูล PCA จะทำหน้าที่สร้างสัญญาณ PWM ออกไปยังขา CEX ของโมดูล PCA โดยการทำงานของโมดูล PCA ในโหมดนี้ จะสามารถควบคุมให้สร้างสัญญาณ PWM ขนาด 8 บิต โดยสามารถกำหนดหรือเปลี่ยนแปลงค่า Duty Cycle ของสัญญาณ PWM ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 แสดง PCA ในโหมด PWM

สำหรับหลักการทำงานของ PCA Timer ในการสร้างสัญญาณ PWM นั้นจะอาศัยผลการเปรียบเทียบค่าการนับแบบ 8 บิตระหว่าง PCA Timer (ซึ่งตามปรกติมี 16 บิต แต่ในโหมดนี้จะใช้การนับแบบ 8 บิต) โดยค่าการนับของ PCA Timer นั้นจะใช้รีจิสเตอร์ CL เป็นตัวเก็บค่าการนับ โดยผลการนับของ CL จะถูกนำไปเปรียบเทียบกับค่าที่กำหนดไว้ในรีจิสเตอร์ CCAPnL โดยถ้าค่าการนับของ PCA Timer ในรีจิสเตอร์ CL มีค่าน้อยกว่าค่าที่กำหนดในรีจิสเตอร์ CCAPnL ก็จะได้ค่าผลลัพธ์ของสัญญาณมีค่าเป็น "0" ที่ Pin Port ของ CPU ที่ทำหน้าที่เป็น Output Pin ของ PCA ช่องนั้นๆ(CEXn = "0") แต่ถ้าค่าการนับของรีจิสเตอร์ CL มีค่าเท่ากับหรือมากกว่าค่าที่กำหนดไว้รีจิสเตอร์ CCAPnL ก็จะได้ผลลัพธ์เป็น "1"ที่ Pin Port ของ CPU ที่ทำหน้าที่เป็น Output Pin ของ PCA ช่องนั้นๆ(CEXn = "1")

โดยจะเห็นว่ารีจิสเตอร์ CL ซึ่งถูกใช้สำหรับการนับนั้น มีขนาดเป็น 8 บิต ดังนั้นค่าผลการนับของรีจิสเตอร์ CL จึงมีค่าการนับวนรอบอยู่ระหว่าง 00H-FFH (0-255) หรือ 256 ค่าพอดี โดยทุกๆครั้งที่ค่าการนับของรีจิสเตอร์ CL เกิดค่าเกิน หรือ Overflow หรือค่าการนับเกินกว่า FFH แล้ว ค่าการนับก็จะวนกลับไปที่ 00H ใหม่ เสมอ ดังนั้นค่า Period หรือค่าคาบสัญญาณ PWM นั้นจะขึ้นอยู่กับผลคูณของ 256 กับค่าความถี่ของสัญญาณนาฬิกาที่ป้อนให้กับวงจร PCA เสมอ ตัวอย่างเช่น สัญญาณนาฬิกาที่ป้อนให้กับ PCA เพื่อทำการนับ มีค่าความถี่ 1 Hz ค่าคาบเวลาทั้งหมดของสัญญาณ PWM ก็จะมีค่าเป็น 256 Hz โดยปริยาย ส่วนค่า Duty Cycle หรือค่าคาบเวลาที่เป็นซีกบวกหรือซีกลบของสัญญาณ PWM นั้นได้จากการกำหนดค่าอ้างอิงการเปรียบเทียบที่กำหนดให้กับรีจิสเตอร์ CCAPnL แทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยค่า Duty Cycle ของสัญญาณ PWM จะกำหนดโดยค่าใน CCAPnL โดยถ้ากำหนดให้ CCAPnL มีค่าน้อย ก็จะได้สัญญาณ PWM ที่มีคาบเวลาด้านบวกมาก แต่ถ้ากำหนดให้มีค่ามาก ก็จะได้สัญญาณ PWM ที่มีคาบเวลาด้านบวกน้อย โดยค่าที่กำหนดลงไปนั้นจะไม่ได้กำหนดให้รีจิสเตอร์ CCAPnL โดยตรง แต่จะกำหนดให้รีจิสเตอร์ CCAPnH แทน โดยจะเห็นได้ว่าในทุกๆครั้งที่ค่าการนับของรีจิสเตอร์ CL นั้นเกิด Overflow หรือเกินกว่า FFH จะมีการส่งสัญญาณไปทำการ Reload ค่าในรีจิสเตอร์ CCAPnH มายังรีจิสเตอร์ CCAPnL ทุกครั้งด้วยเสมอ ตัวอย่างเช่น ถ้ากำหนดค่าให้กับรีจิสเตอร์ CCAPnL (โดยกำหนดผ่าน CCAPnH) ด้วยค่า 10 จะได้ว่า ลักษณะของสัญญาณ PWM ที่ได้จะมีคาบของสัญญาณซีกลบ (“0”) เป็นเวลา 10/256 หน่วยของคาบเวลาทั้งหมด และก็จะมีคาบเวลาในซีกบวก (“1”) เป็นเวลา 246/256 หน่วยของคาบเวลาทั้งหมดเป็นต้น

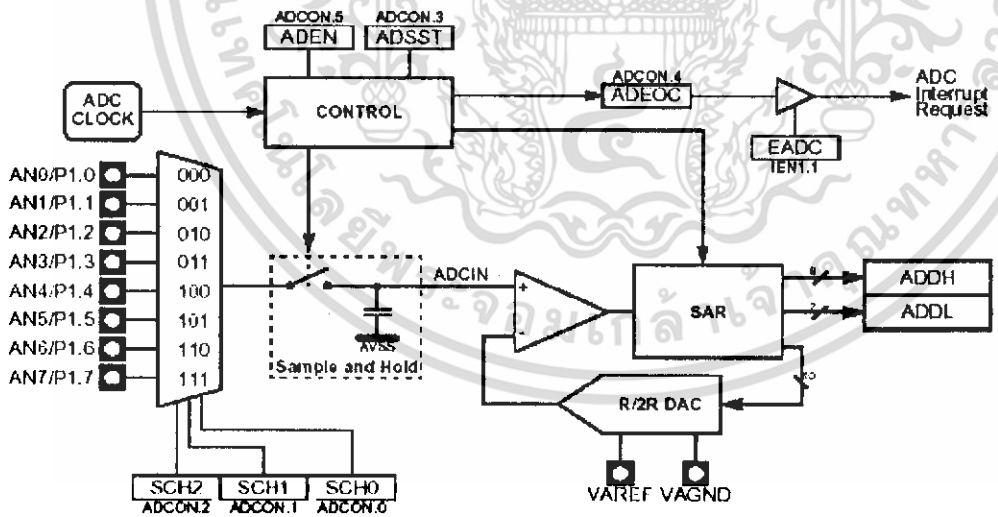
2.4 วงจร Analog-to-Digital (ADC) ในไมโครคอนโทรลเลอร์ T89C51AC2

วงจร Analog-to-Digital (ADC) ก็เป็นคุณสมบัติพิเศษอีกหนึ่งอย่างในไมโครคอนโทรลเลอร์ T89C51AC2 โดยมีหน้าที่แปลงสัญญาณอนาล็อกให้เป็นสัญญาณดิจิตอล

สูตรพื้นฐานของ Analog-to-Digital Converter

Data = (Vin x 255) / 5

โดยลักษณะของวงจร ADC ในไมโครคอนโทรลเลอร์ T89C51AC2 เป็นดังนี้



รูปที่ 2.8 แสดงวงจร ADC ในไมโครคอนโทรลเลอร์ T89C51AC2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปด้านบนจะเป็นการอธิบายถึงการแปลงแรงดันไฟฟ้าไปสู่ข้อมูลแบบ 10 บิต ADC ทั้ง8 channel จะเป็นช่องทางที่เอาไว้รับแรงดันภายนอกเข้ามาได้ 8 ค่า(AN0-AN7) และที่ตัวMultiplexer นั้น เราก็สามารถเลือกการแปลงแบบค่าเดียวจากแปดค่าได้ ค่าที่ได้จะเรียกว่า ADC Input Voltage (ADCIN) และค่า ADCIN นี้ก็จะถูกนำไปแปลงด้วย 10 bit cascaded potentiometric ADC

ลักษณะสำคัญ

- มีช่องInput แบบ Multiplex จำนวน 8 Channel
- 10 bit cascaded potentiometric ADC
- ใช้เวลาในการแปลง 16 ms.
- Zero Error (offset) ± 2 LSB max
- ช่วงค่าแรงดันอ้างอิง (VREF) 2.4 -3.0 Volt
- ADCIN อยู่ในช่วง 0 – 3.0 Volt
- Integral non-linearity typical 1 LSB, max. 2 LSB
- Differential non-linearity typical 0.5 LSB, max. 1 LSB
- Conversion Complete Flag or Conversion Complete Interrupt
- สามารถเลือก ADC Clock ได้

2.4.1 การแปลง Voltage ของวงจร Analog-to-Digital (ADC) ในไมโครคอนโทรลเลอร์ T89C51AC2

ในส่วนของการแปลงพอจะสรุปคร่าวๆได้ดังนี้

เมื่อค่าของ Input Voltage หรือ ADCIN มีค่ามากกว่าหรือเท่ากับ VAREF ตัว ADC ก็จะทำการแปลงสัญญาณที่ได้เป็นค่าเท่ากับ 3FFH (full scale) แต่ถ้าหาก Input Voltage มีค่าน้อยกว่าหรือเท่ากับ VAGND ตัว ADC ก็จะแปลงสัญญาณที่ได้เป็น 000H ส่วน Input Voltage ที่มีค่าอยู่ระหว่าง VAREF กับ VAGND ค่าการแปลงที่ได้ก็จะเป็นกราฟเส้นตรงที่มีความเป็นเชิงเส้น (straight-line linear)

หมายเหตุ : ค่า Input Voltage หรือ ADCIN ไม่ควรจะมีค่าเกินกว่า VAREF (ศึกษาได้ใน datasheet เรื่อง AC-DC)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 รีจิสเตอร์ที่สำคัญของวงจร Analog-to-Digital (ADC) ในไมโครคอนโทรลเลอร์ T89C51AC2

ADCF Register

ADCF (S:F6h)
ADC Configuration

7	6	5	4	3	2	1	0
CH 7	CH 6	CH 5	CH 4	CH 3	CH 2	CH 1	CH 0

Bit Number	Bit Mnemonic	Description
7-0	CH 0:7	**Channel Configuration** Set to use P1.x as ADC input. Clear to use P1.x as standart I/O port.

Reset Value = 0000 0000b

รูปที่ 2.9 แสดงลักษณะของ ADCF Register

ADCLK Register

ADCLK (S:F2h)
ADC Clock Prescaler

7	6	5	4	3	2	1	0
-	-	-	PRS 4	PRS 3	PRS 2	PRS 1	PRS 0

Bit Number	Bit Mnemonic	Description
7-5	-	**Reserved** The value read from these bits are indeterminate. Do not set these bits.
4-0	PRS4:0	**Clock Prescaler** f_{ADC} = fcpu clock/ (4 (or 2 in X2 mode)* (PRS +1))

Reset Value = XXX0 0000b

รูปที่ 2.10 แสดงลักษณะของ ADCLK Register

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADCON Register

ADCON (S:F3h)
ADC Control Register

7	6	5	4	3	2	1	0
-	PSIDLE	ADEN	ADEOC	ADSST	SCH2	SCH1	SCH0

Bit Number	Bit Mnemonic	Description
7	-	
6	PSIDLE	**Pseudo Idle Mode (Best Precision)** Set to put in idle mode during conversion Clear to convert without idle mode.
5	ADEN	**Enable/Standby Mode** Set to enable ADC Clear for Standby mode (power dissipation 1 uW).
4	ADEOC	**End Of Conversion** Set by hardware when ADC result is ready to be read. This flag can generate an interrupt. Must be cleared by software.
3	ADSST	**Start and Status** Set to start an A/D conversion. Cleared by hardware after completion of the conversion
2-0	SCH2:0	**Selection of Channel to Convert** see Table 56

Reset Value = X000 0000b

รูปที่ 2.11 แสดงลักษณะของ ADCON Register

ADDH Register

ADDH (S:F5h Read Only)
ADC Data High Byte Register

7	6	5	4	3	2	1	0
ADAT 9	ADAT 8	ADAT 7	ADAT 6	ADAT 5	ADAT 4	ADAT 3	ADAT 2

Bit Number	Bit Mnemonic	Description
7-0	ADAT9:2	ADC result bits 9-2

Reset Value = 00h

รูปที่ 2.12 แสดงลักษณะของ ADDH Register

62853
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDL Register

ADDL (S:F4h Read Only)
ADC Data Low Byte Register

7	6	5	4	3	2	1	0
-	-	-	-	-	-	ADAT 1	ADAT 0

Bit Number	Bit Mnemonic	Description
7-2	-	**Reserved** The value read from these bits are indeterminate. Do not set these bits.
1-0	ADAT1:0	**ADC result** bits 1-0

Reset Value = 00h

รูปที่ 2.13 แสดงลักษณะของ ADDL Register

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบ

การออกแบบในโครงงานแยกได้เป็น 4 ส่วนหลักๆ คือ

1. ส่วนฮาร์ดแวร์ ประกอบด้วย

   1.1 โครงสร้างของเครื่องช่วยเดิน

   1.2 วงจรไมโครคอนโทรลเลอร์

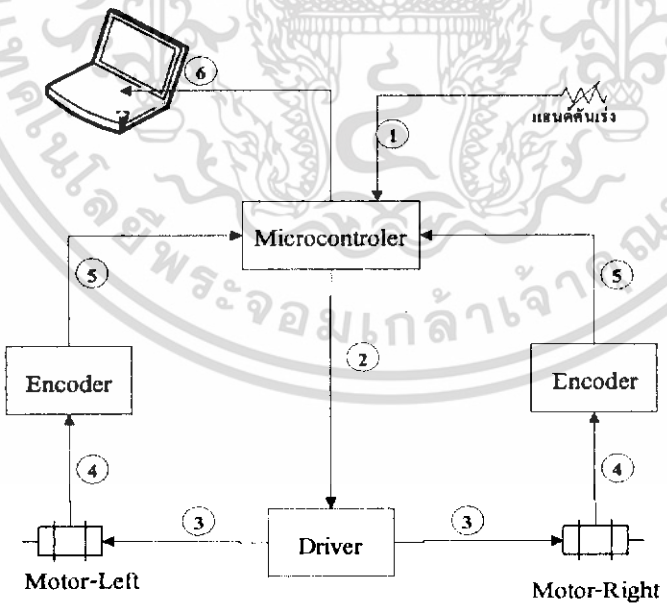
2. ส่วนซอฟแวร์ของวงจรไมโครคอนโทรลเลอร์ ประกอบด้วย โปรแกรมที่ใช้ควบคุมไมโครคอนโทรลเลอร์ ได้แก่ โปรแกรม Analog-to-Digital Converter (ADC) โปรแกรม Pulse Width Modulation (PWM) ที่ใช้ในการเร่งความเร็ว และโปรแกรมตรวจจับการหมุนของมอเตอร์

3. ส่วนซอฟแวร์ในส่วนของเกม

4. ส่วน User Interface

3.1 โครงสร้างโดยรวม

โครงสร้างโดยรวมของโครงงานสามารถแสดงได้ดังนี้



รูปที่ 3.1 แสดงผังโครงสร้างโดยรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.1สามารถแสดงขั้นตอนต่างๆ ดังนี้

① รับค่าแรงดันไฟฟ้ามาจากแฮนด์บิดเร่งความเร็วเพื่อนำไปเข้าสู่วงจร ADC ในไมโครคอนโทรลเลอร์

② ไมโครคอนโทรลเลอร์ส่งค่าที่ได้ออกมาไปยัง Driver เพื่อขับมอเตอร์

③ Driver ทำการขับมอเตอร์

④ ค่าการหมุนของมอเตอร์เข้าสู่วงจร Encoder

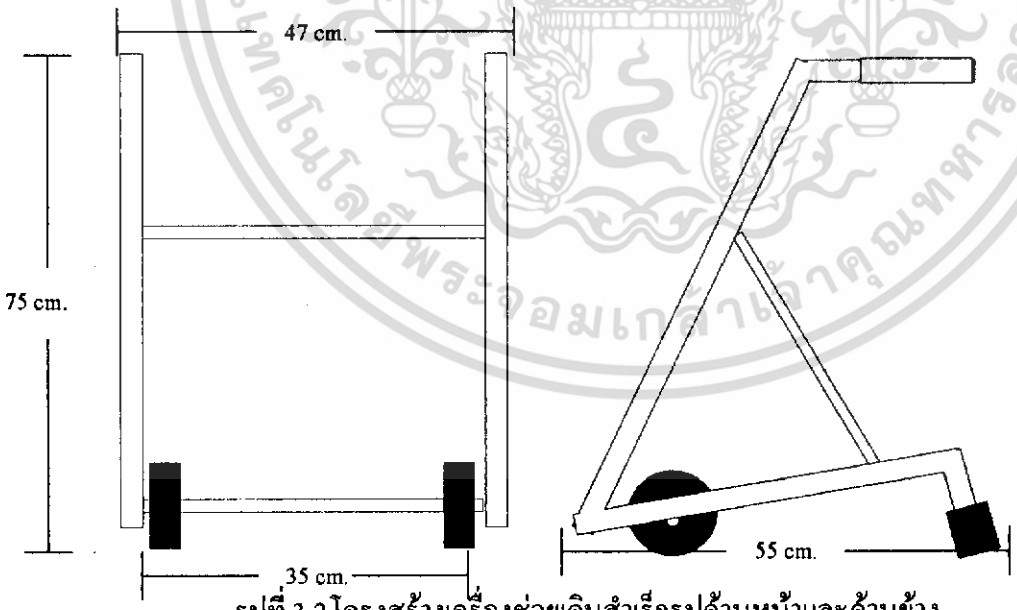
⑤ ค่าที่ได้จากการ Encoder ถูกส่งไปประมวลผลที่ไมโครคอนโทรลเลอร์

⑥ ไมโครคอนโทรเลอร์ส่งค่าที่ได้จากการ Encoder เข้าคอมพิวเตอร์โดยผ่านทางพอร์ตอนุกรม (RS-232) เพื่อประมวลผลต่อไป

3.2 การออกแบบส่วนฮาร์ดแวร์

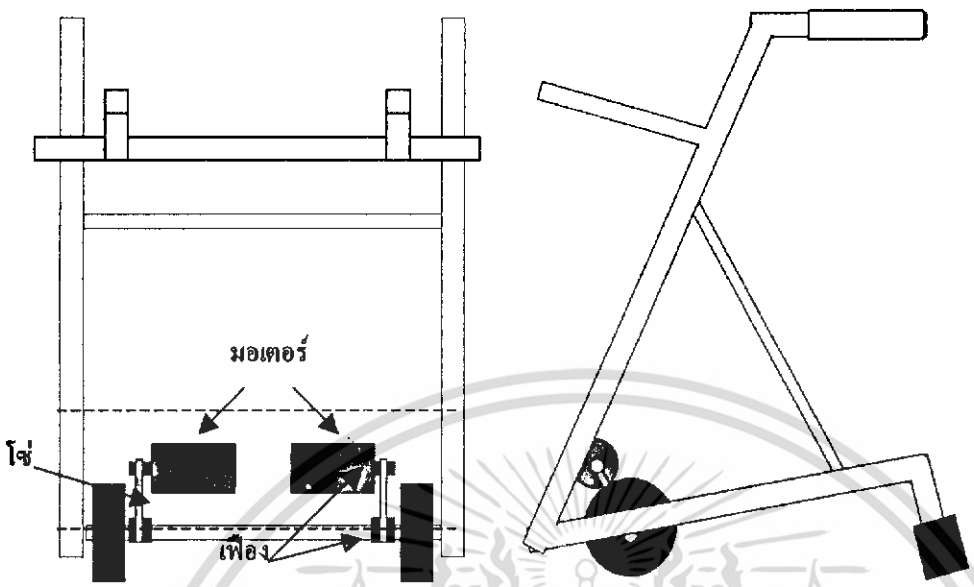
3.2.1 การออกแบบเครื่องช่วยเดิน

ในโครงงานนี้จะนำเครื่องช่วยเดินสำเร็จรูปแบบสองล้อปรับความสูงได้ 5 ระดับ คือ 75, 77.5, 80, 82.5 และ 85 เซนติเมตร มาทำการปรับเปลี่ยนเพิ่มเติม โดยเพิ่มในส่วน เฟืองโซ่ล้อ แท่นวางคอมพิวเตอร์โน้ตบุ๊ค แฮนด์คันเร่ง และมอเตอร์ มีลักษณะโดยรวมแสดงได้ดังนี้



รูปที่ 3.2 โครงสร้างเครื่องช่วยเดินสำเร็จรูปด้านหน้าและด้านข้าง

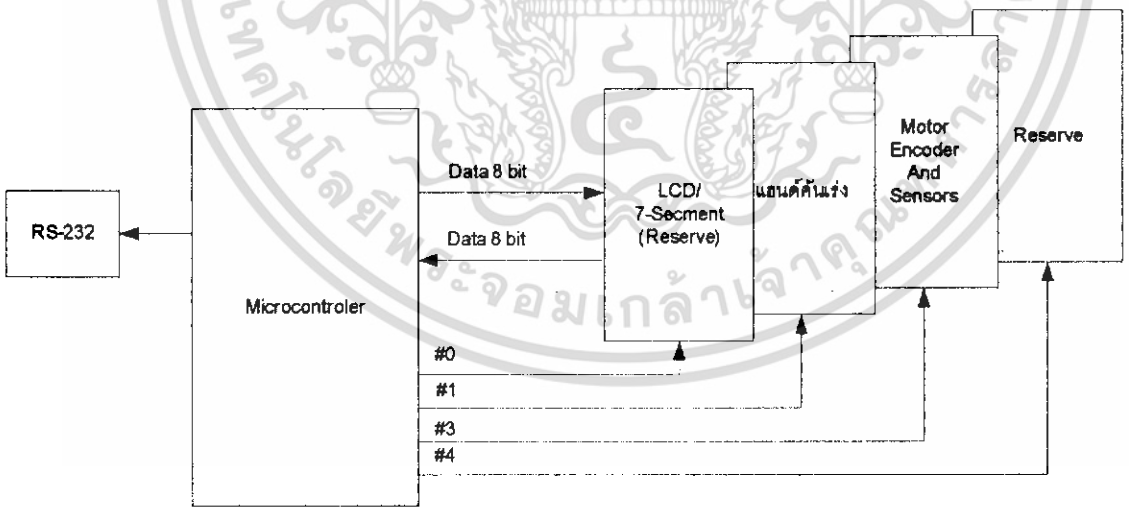
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 โครงสร้างเครื่องช่วยเดินติดตั้งมอเตอร์และแท่นวางคอมพิวเตอร์โน๊ตบุ๊คด้านหน้าและด้านข้าง

3.2.2 การออกแบบวงจรรวม

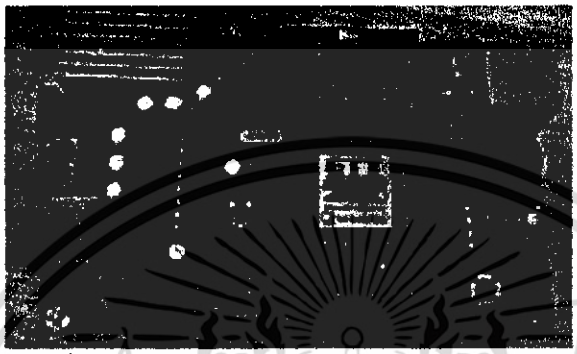
วงจรรวมในส่วนฮาร์ดแวร์ประกอบด้วย วงจรไมโครคอนโทรลเลอร์ที่ต่อกับแฮนด์คันเร่งและมอเตอร์เอนโค้ดเดอร์ 2 ตัว และวงจร RS-232 แสดงผังโดยรวมได้ดังนี้



รูปที่ 3.4 แสดงผังการออกแบบวงจรรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยวงจรรวมจะแยกออกเป็น 2 ส่วนคือ ไมโครคอนโทรลเลอร์ ที่ต่อกับแฮนด์คันเร่งและมอเตอร์ อีกส่วนคือ วงจร RS-232 โดยแฮนด์คันเร่งจะทำหน้าที่เร่งความเร็วการหมุนของมอเตอร์โดยผ่านไมโครคอนโทรลเลอร์ และการหมุนของมอเตอร์ก็จะเป็นอินพุตเข้าไปยังคอมพิวเตอร์โดยผ่านทางพอร์ทอนุกรม (RS-232) เพื่อประมวลผลต่อไป

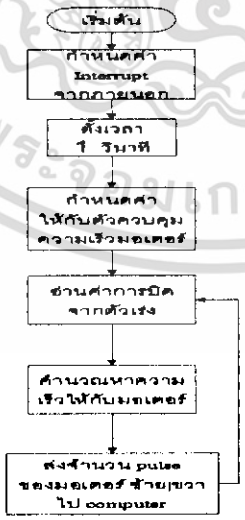


รูปที่ 3.5 รูปถ่ายแผงวงจรไมโครคอนโทรลเลอร์

3.3 การออกแบบส่วนซอฟแวร์

3.3.1 การออกแบบซอฟแวร์ในส่วนวงจรไมโครคอนโทรลเลอร์

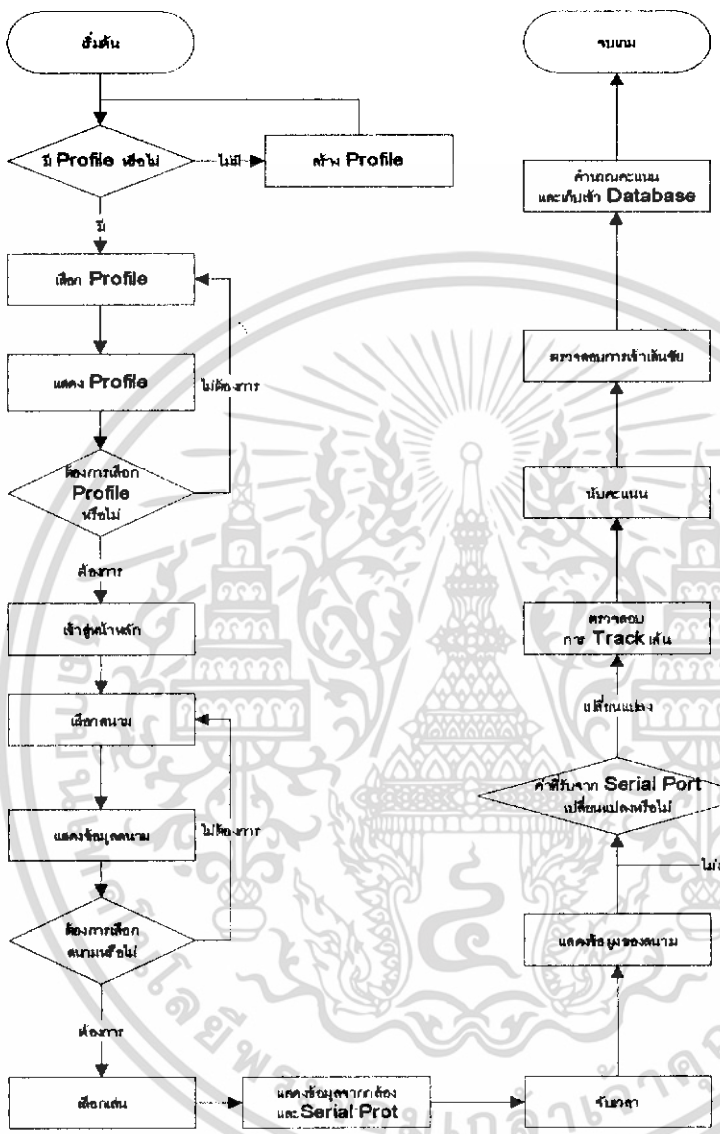
ภายในประกอบด้วยโปรแกรม Analog-to-Digital Converter (ADC) โปรแกรม Pulse Width Modulation (PWM)และโปรแกรมตรวจจับการเคลื่อนที่ของมอเตอร์ซึ่งเซ็นเซอร์จะทำการ Interruptส่งค่าไปยังคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม



รูปที่ 3.6 แสดง Flow Chart การทำงานของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

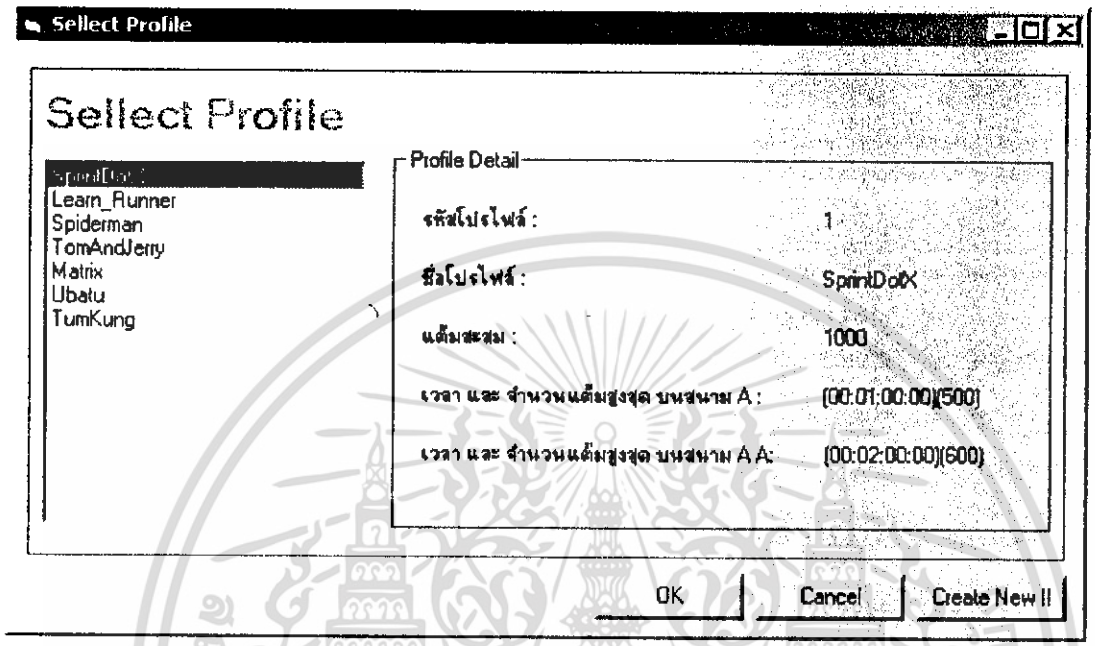
3.3.2 การออกแบบซอฟแวร์ส่วนเกม



รูปที่ 3.7 แสดง Flow Chart การทำงานของเกม

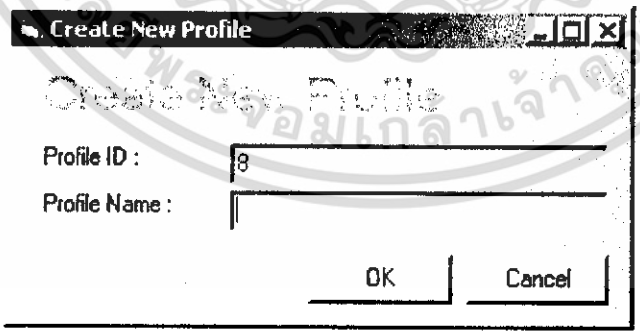
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การออกแบบส่วน User Interface



รูปที่ 3.8 หน้าจอ Select Profile

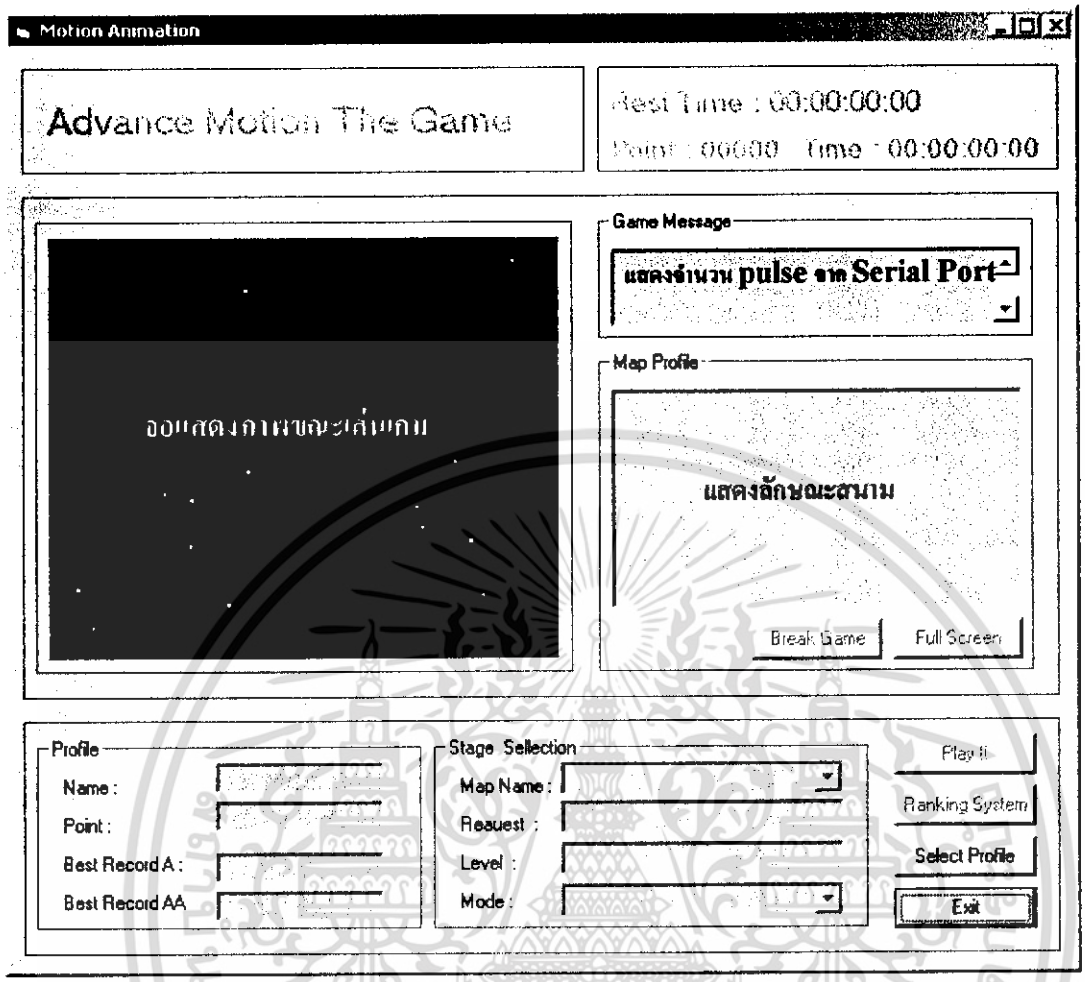
รูปด้านบนเป็นหน้าจอ Select Profile ใช้สำหรับแสดงชื่อผู้เล่น, แต้มสูงสุด, คะแนนสูงสุดและลงทะเบียนผู้เล่นใหม่



รูปที่ 3.9 หน้าจอ Create New Profile

รูปด้านบนเป็นหน้าจอ Create New Profile ใช้สำหรับกรอกข้อมูลผู้เล่นใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 หน้าจอหลัก Advance Motion The Game

รูปด้านบนเป็นหน้าจอหลักที่ใช้เล่นเกมประกอบด้วย

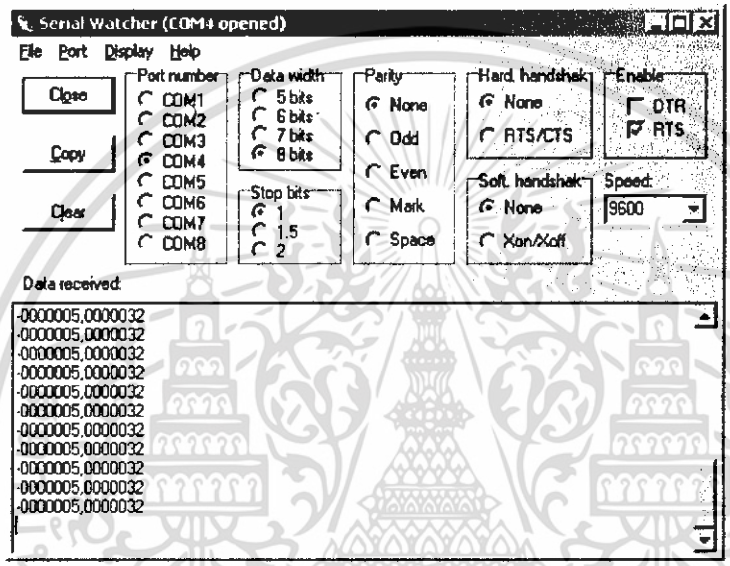
- จอแสดงภาพขณะเล่นเกม
- กรอบ Game Messageแสดงจำนวน pulse ที่รับมาจาก Serial Port ของเซ็นเซอร์ด้านซ้ายและขวา
- กรอบ Map Profile แสดงลักษณะสนาม
- กรอบ Stage Selection ประกอบด้วยเมนูที่ใช้เลือกสนามและโหมดการเล่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 ผลการทดลองการนับค่า pulse ของเซ็นเซอร์ซ้ายและขวาจากการหมุนของมอเตอร์ด้วยโปรแกรม Serial Watcher



รูปที่ 4.1 แสดงการนับ pulse จากเซ็นเซอร์ด้านซ้านและขวา

จากรูปแสดงให้เห็นว่าเซ็นเซอร์ซ้ายจับการหมุนของมอเตอร์ด้านซ้ายซึ่งหมุนไปข้างหลัง (แสดงเป็นค่าลบ)เป็นจำนวน 5 pulse ส่วนเซ็นเซอร์ขวาจับการหมุนของมอร์เตอร์ด้านขวาซึ่งหมุนไปข้างหน้าเป็นจำนวน 32 pulse (แสดงเป็นค่าบวก)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ขั้นตอนการเล่นเกม

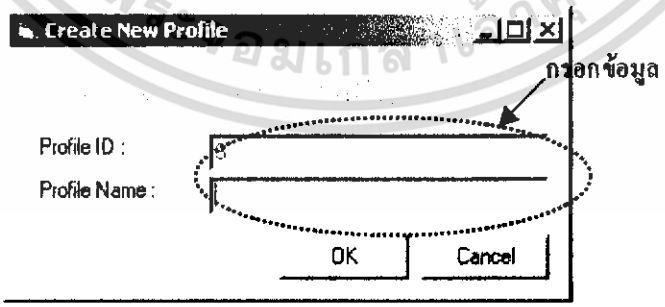
4.2.1 ลงทะเบียน

เข้าสู่หน้าหลักแล้วคลิกที่ Select Profile เพื่อเข้าสู่หน้า Select Profile ซึ่งแสดงจะข้อมูลต่างๆ ประกอบด้วย รหัส, ชื่อ, แต้มสะสม, เวลาและแต้ม สูงสุดในแต่สนามของผู้เล่นแต่ล่ะคน จากนั้นให้คลิกที่ปุ่ม Create New !! เพื่อเข้าสู่หน้า Create New Profile เพื่อกรอกข้อมูลผู้เล่น

รูปที่ 4.2 หน้าจอ Select Profile

4.2.2 กรอกข้อมูลผู้เล่นใหม่

เข้าสู่หน้า Create New Profile แล้วกรอกข้อมูลผู้เล่นใหม่ เมื่อกรอกข้อมูลเสร็จแล้วคลิกปุ่ม OK เพื่อกลับสู่หน้าหลัก

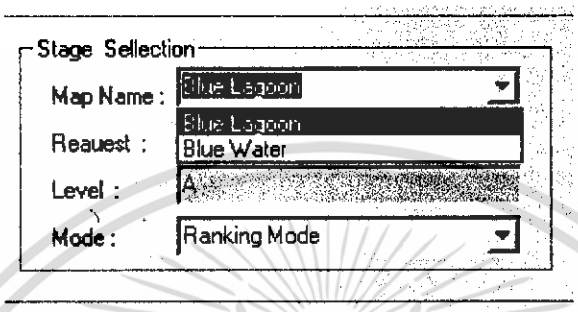


รูปที่ 4.3 หน้าจอ Create New Profile กรอกข้อมูลผู้เล่นใหม่

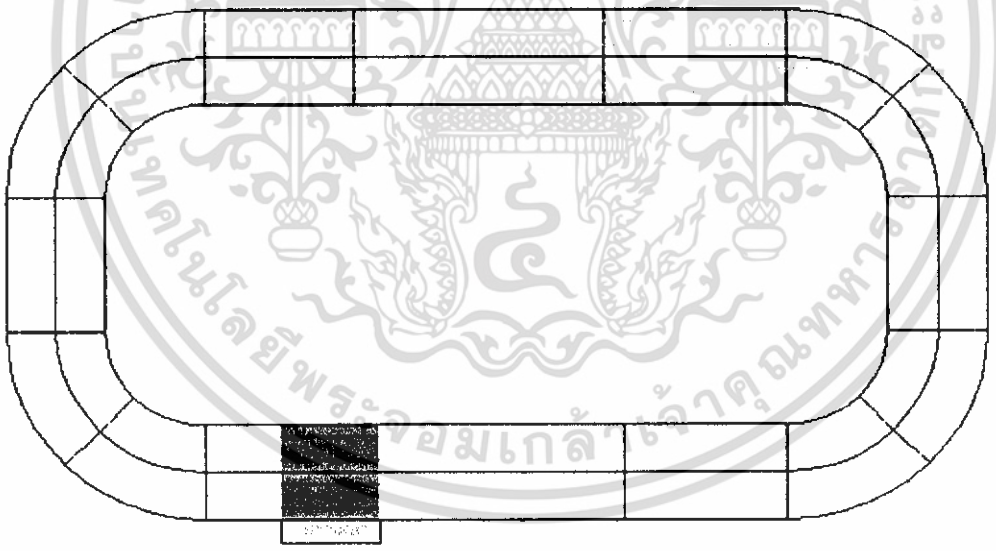
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 เลือกสนาม

เมื่อเข้ากลับเข้าสู่หน้าหลักแล้ว ต่อไปจะเป็นการเลือกสนามในการเล่น โดยเลือกได้ที่เมนู Map Name และในกรอบของ Stage Selection



รูปที่ 4.4 แสดงการเลือกสนาม

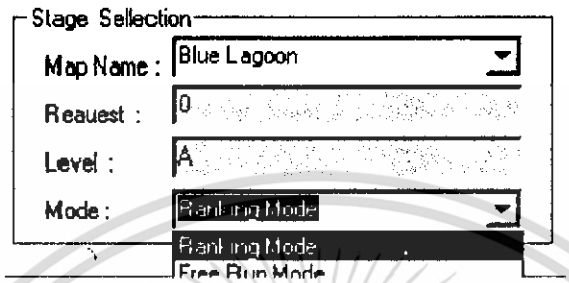


รูปที่ 4.5 ลักษณะสนามจำลองในเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 เลือกโหมดการเล่น

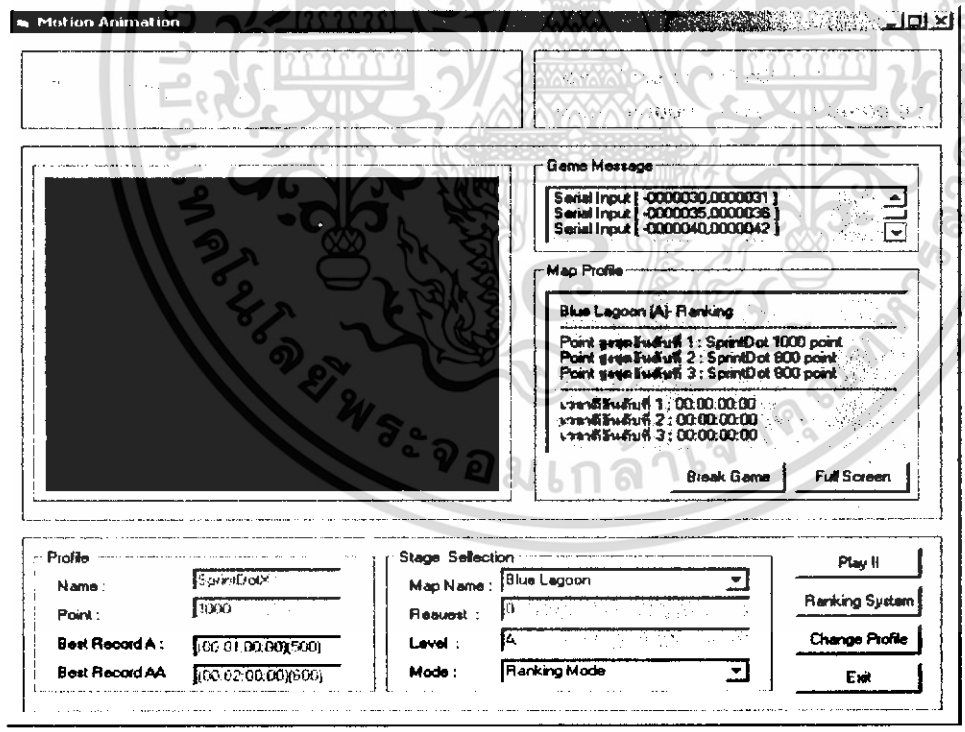
ต่อไปเป็นการเลือกโหมดในการเล่นเกม โดยเลือกที่เมนู Mode แบ่งเป็น 2 โหมด คือ Ranking Mode และ Free Run Mode ในกรอบของ Stage Selection เช่นเดียวกับข้อ4



รูปที่ 4.6 แสดงการเลือกโหมด

4.2.5 เข้าสู่เกม

เข้าสู่หน้าหลักของเกม คลิกที่ Play !! เพื่อเล่นเกม

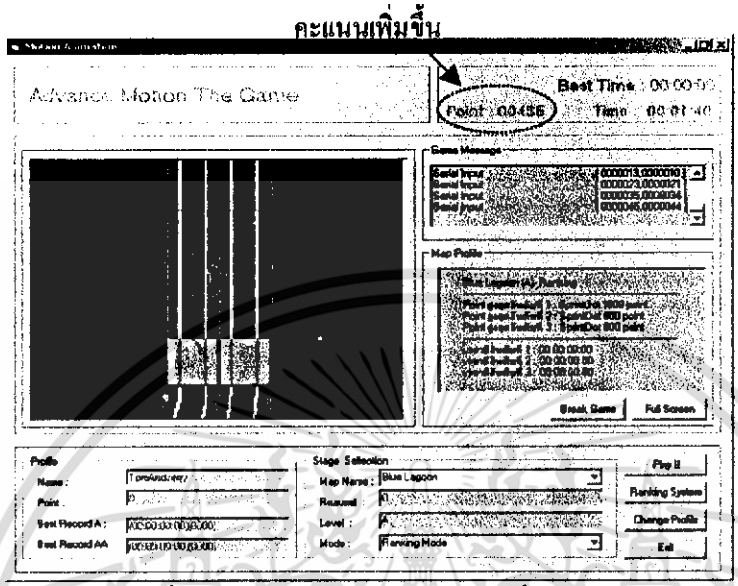


รูปที่ 4.7 หน้าจอหลักของเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

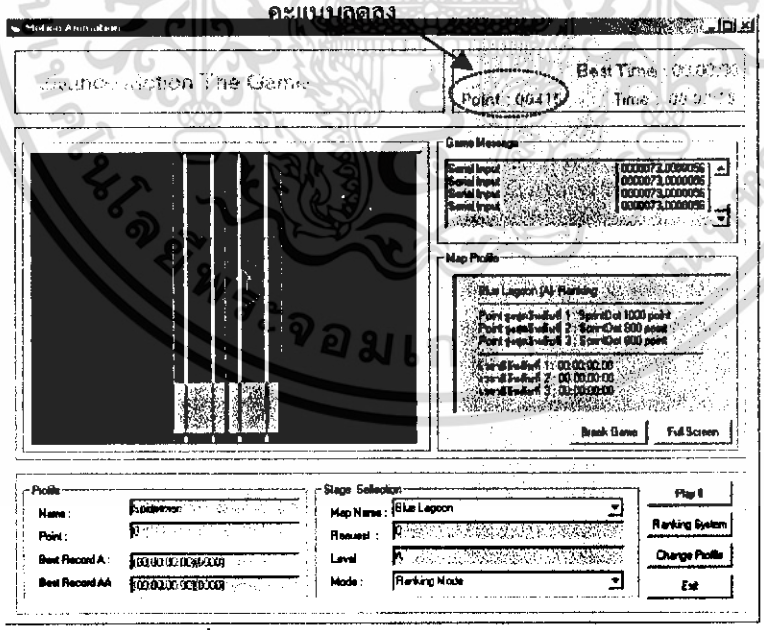
4.3 ผลการทดลองจากเกม

จากรูปด้านล่างแสดงการเดินตามเส้นทางที่กำหนด โดยคะแนนในเกมจะเพิ่มตามลำดับ



รูปที่ 4.8 แสดงการเดินตามเส้นทางที่กำหนด

รูปต่อไปแสดงการเดินออกนอกเส้นทางที่กำหนด จะเห็นว่าคะแนนจะลดลงจากคะแนนเดิม



รูปที่ 4.9 แสดงการเดินออกนอกเส้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุปและวิจารณ์

5.1 สรุปผลการดำเนินงาน

โครงงานนี้สร้างขึ้นมาเพื่อให้ผู้ที่ทำการกายภาพบำบัดได้เล่นเกมขณะที่ใช้เครื่องช่วยเดินเพื่อดึงดูดความสนใจและเพิ่มความสนุกสนานในการทำกายภาพบำบัด โดยการนับค่า pulse จากเซ็นเซอร์ที่ได้จากการหมุนของของมอเตอร์ทั้งสองข้างแล้วส่งเข้ามาประมวลในโปแกรมเกม โดยเป็นสำหรับผู้เล่นหนึ่งคน เกมจะเป็นลักษณะลู่วิ่งและมีทางโค้ง หากวิ่งไปตามเส้นที่กำหนดลู่ก็จะได้คะแนนเพิ่มขึ้นเรื่อยๆ แต่ถ้าหากวิ่งหลุดจากเส้นที่กำหนดคะแนนก็จะลดลงตามลำดับ และจะทำการเก็บสถิติคะแนนและเวลาของผู้เล่นแต่ละคนไว้เปรียบเทียบเพื่อการแข่งขัน

5.2 ปัญหาที่เกิดและแนวทางแก้ไข

ปัญหาที่เกิดและแนวทางการแก้ปัญหาสามารถสรุปได้ดังนี้

- การดำเนินงานล่าช้า เนื่องจากต้องใช้เวลาศึกษาอุปกรณ์ เพื่อนำมาออกแบบ และโปรแกรมต่างๆ
  แนวทางแก้ไข : ควรวางแผนและหาข้อมูลก่อนดำเนินโครงงานอย่างรอบคอบ
- ภาพที่ได้จากกล้องวิดีโอWebcamไม่คมชัดเท่าที่ควร
  แนวทางแก้ไข : ใช้กล้องวิดีโอที่มีความละเอียดสูงกว่ากล้องที่ใช้ในโครงงาน
- วงจรฮาร์แวร์มีสายเชื่อมต่อมากเนื่องจากออกแบบเป็นแบบโมดูลโดยใช้สายจั้มเชื่อมต่อส่งผลให้เสี่ยงต่อการเสียหายได้ง่าย
  แนวทางแก้ไข : ออกแบบวงจรให้รัดกุม และมั่นคงแข็งแรงมากกว่าเดิม
- ตัวเครื่องช่วยเดินหลังจากติดตั้งอุปกรณ์ทั้งหมดแล้วมีน้ำหนักมาก เกิดจากน้ำหนักของมอเตอร์และคอมพิวเตอร์โน๊ตบุ๊ค
  แนวทางแก้ไข : ใช้มอร์เตอร์ให้กำลังขับใกล้เคียงมอเตอร์ตัวเดิมที่มีน้ำหนักน้อยกว่า

5.3 ข้อจำกัดของโครงงาน

จากการทดลองพื้นที่ของสนามจำลองในเกมอาจไม่สัมพันธ์กับพื้นที่ของสถานที่ที่ใช้ในการเล่นจริงๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 แนวทางการพัฒนาต่อ

5.4.1 การพัฒนาต่อในส่วนแอพลิเคชั่น

โดยอาจเปลี่ยนจากรูปแบบเกมไปเป็นแอพพลิเคชั่นรูปแบบอื่นๆ เช่น แอพลิเคชั่นที่ใช้ในการนำทางบน Pocket PC เพื่อลดขนาดและน้ำหนักของเครื่องช่วยเดินสำหรับผู้ชราที่มีอาการหลงลืม

5.4.2 การพัฒนาต่อในส่วนฮาร์ดแวร์

โดยอาจจะทำการเพิ่มอุปกรณ์ต่างๆ เช่น เซ็นเซอร์จับวัตถุ เพื่อทำการแจ้งเตือนเมื่อพบวัตถุอุปสรรคขวางทางในขณะเล่นเกมเป็นการเพิ่มลูกเล่น

5.4.3 ใช้ชุดโปรแกรมภาษาอื่นๆเพื่อเพิ่มประสิทธิภาพในการประมวลผลที่สูงขึ้น เช่น Microsoft Visual C++

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ผศ.ธีรวัฒน์ ประกอบผล,2547, การพัฒนาไมโครคอนโทรลเลอร์ด้วยภาษาซี, สำนักพิมพ์ ส.ส.ท.

[2] นิรุธ อำนวยศิลป์, 2546, คู่มือการเขียนโปรแกรมด้วยภาษาซี, บริษัท ไปรวิชั่น จำกัด

[3] กิตติ ภักดีวัฒนกุล, จำลอง ครูอุตสาหะ, 2546, Visual Basic 6 ฉบับบโปรแกรมเมอร์, บริษัท เคทีพี คอมพ์ แอนด์ คอนซัลท์ จำกัด

[4] อภิชาติ ภู่พลับ, สัจจะ จรัสรุ่งรวีวร, 2546, เริ่มต้นเขียนโปรแกรมติดต่อและควบคุมฮาร์ดแวร์ด้วย Visual Basic, Infopress Developer Book

[5] อภิชาติ ภู่พลับ, สัจจะ จรัสรุ่งรวีวร, 2546, สนุก! กับการประยุกต์ใช้ Visual Basic, Infopress Developer Book

[6] อภิชาติ ภู่พลับ, สัจจะ จรัสรุ่งรวีวร, 2547, รวมคอนโทรลฟรี[สำหรับ VB6], Infopress Developer Book

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมในไมโครคอนโทรลเลอร์

```
#include "89C51AC2.h"
#include "absacc.h"
#include "math.h"
#include "Function.c"
#include "LCD.c"
unsigned char CountR=0,CountL=0,Count;
long  TCountR=0;TCountL=0;
unsigned char Tag1,Tag2;
bit  Flag;
float Sum;
int  RMotor;
/* --- Main Program --- */
unsigned int Read_A2D(unsigned char channel)
{unsigned int Data;

                ADCON = 0x20 + channel;  /*  AtoD  Active Ch  Ready */
                ADCON = ADCON | 0x08; /* Start Conversion  */
                while((ADCON & 0x10) == 0) { }
      Data = ADDH;
                Data = (Data << 2) + ADDL;
      return(Data);
}

main()
{unsigned int Speed;
    EX0 = 1;
        EX1 = 1;
        IT0 = 1;
    IT1 = 1;

        Flag = 0;
    TMOD= 0x01;
    TH0 = 0x10;
    TL0 = 0x00;
    TR0 = 1;
    ET0 = 1;
    EA  = 1;

    Init_LCD();
        Init_Serial(96);
    Wr_String(" < Direction >  ",1);
    CMOD = 0x00;
    CCAPM1 = 0x42;
    CCAP1L = 0x00;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  CCON   = 0x40;
  Delay(20);
  CountR = 0;
  CountL = 0;
while(1)
 {

  Speed = Read_A2D(2)-205;
  Sum   = Speed*255.0;
  Sum   = Sum/1024.0;
  CCAP1H = 255-Sum;

  if(Flag)
  {
  if(Tag1 == 0){TCountR = TCountR + CountR;}
        else {TCountR = TCountR - CountR;

   }
  if(Tag2 == 0){TCountL = TCountL - CountL;}
        else {TCountL = TCountL + CountL;}

                if(TCountL<0){Tx_Byte('-');}
      CountToLCD(abs(TCountL));
      Tx_Byte(BufDisp[6]);
                        Tx_Byte(BufDisp[5]);
      Tx_Byte(BufDisp[4]);
                        Tx_Byte(BufDisp[3]);
                        Tx_Byte(BufDisp[2]);
                        Tx_Byte(BufDisp[1]);
                        Tx_Byte(BufDisp[0]);
                        Tx_Byte(',');

                        if(TCountR<0){Tx_Byte('-');}
      CountToLCD(abs(TCountR));
      Tx_Byte(BufDisp[6]);
                        Tx_Byte(BufDisp[5]);
      Tx_Byte(BufDisp[4]);
                        Tx_Byte(BufDisp[3]);
                        Tx_Byte(BufDisp[2]);
                        Tx_Byte(BufDisp[1]);
                        Tx_Byte(BufDisp[0]);
                        CountR = 0;CountL = 0;

    Flag =0;
  }
  if((P2 & 0x03)== 3)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
     if((P2 & 0x03)== 2){Tag1 = 1;}else
     if((P2 & 0x03)== 1){Tag1 = 0;}
    }
    if((P2 & 0x0C)== 0x0C)
    {if((P2 & 0x0C)== 8){ Tag2 = 1;}else
     if((P2 & 0x0C)== 4){ Tag2 = 0;}
    }
   }
}
void  Ext0_Int(void)  interrupt 0
{

 CountR++;

}
void  Ext1_Int(void)  interrupt 2
{
 CountL++;


}
void  Timer0(void) interrupt 1
{
  TH0 =0x10;
  TL0 = 0x00;
  Count++;
  if(Count == 15)
   { Count = 0;
     Flag =1;
   }

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Option Explicit
'ค่าตัวแปรที่ใช้กับ Database และ ตัวจับเวลา
Dim Conn As New ADODB.Connection
Dim rsMap As New ADODB.Recordset
Dim rsSeek As New ADODB.Recordset
Dim timmerCount As Long
Dim sqlmap As String
Dim stopL As Integer

'vdoCapX
Dim transColor As Long, xx, yy

'Win32GDI
Private Declare Function CreateCompatibleDC Lib "gdi32" (ByVal hdc As Long) As
Long
Private Declare Function DeleteDC Lib "gdi32" (ByVal hdc As Long) As Long
Private Declare Function GetTickCount Lib "kernel32" () As Long
Private Declare Function SelectObject Lib "gdi32" (ByVal hdc As Long, ByVal
hObject As Long) As Long
Private Declare Function TransparentBlt Lib "msimg32" (ByVal hdcDest As Long,
ByVal nXOriginDest As Long, ByVal nYOriginDest As Long, ByVal nWidthDest As
Long, ByVal nHeightDest As Long, ByVal hdcSrc As Long, ByVal nXOriginSrc As
Long, ByVal nYOriginSrc As Long, ByVal nWidthSrc As Long, ByVal nHeightSrc
As Long, ByVal crTransparent As Long) As Long
Private Declare Function GetPixel Lib "gdi32" (ByVal hdc As Long, ByVal x As
Long, ByVal y As Long) As Long

Private gbolRunning As Boolean
Private hdcCamera As Long
Private hdcTrack As Long
Private hdcDetect As Long
Private lbytDrawMethod As Byte
Dim lintSrcPictureX As Integer
Dim lintSrcPictureY As Integer

'ตัวแปรที่ใช้นับค่าเวลา คะแนน ค่าที่ใช้เปรียบเทียบกับ Serial Port
Dim msec, sec, min As Byte

Public Sub create_map(tmp_name As String)
   Dim tmp_str As String
   'ใช้ในการเลือก Map ขึ้นมาแสดง
   sqlmap = "SELECT map.map_id, map.map_name, map.map_request,
map.map_level, "
   sqlmap = sqlmap & "map.map_mode, map.map_picture, map.map_bestpoint1,
map.map_bestpoint2, map.map_bestpoint3, "
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   sqlmap = sqlmap & "map.map_besttime1, map.map_besttime2,
map.map_besttime3 "
   sqlmap = sqlmap & " FROM map"
   sqlmap = sqlmap & "  WHERE (map.map_name='" & tmp_name & "')"

   'แสดงแผนที่และข้อมูล ทั้งหมดลงใน Form
   With rsMap
   If .State = adStateOpen Then .Close
   .ActiveConnection = Conn
   .CursorType = adOpenForwardOnly
   .CursorLocation = adUseClient
   .Open sqlmap

   'กำหนดค่าเริ่มต้นให้ Form
   dbcb_mapname.Enabled = True
   dbcb_mapname.BackColor = &H80000005
   dbcb_mapname.Text = tmp_name
   txt_request.Text = .Fields("map_request").Value
   txt_request.Enabled = False
   txt_request.BackColor = &HC0C0C0
   cb_level.Text = .Fields("map_level").Value
   cb_level.Enabled = False
   cb_level.BackColor = &HC0C0C0
   cb_mode.Enabled = True
   cb_mode.BackColor = &H80000005
   tmp_str = .Fields("map_name").Value
   cb_mode.Text = "Ranking Mode"
   tmp_str = tmp_str & " (" & .Fields("map_level").Value & ")" & "- Ranking" &
vbCrLf
   tmp_str = tmp_str & "----------------------------------------------------------------------
-----" & vbCrLf
   tmp_str = tmp_str & "Point สูงสุดอันดับที่ 1 : " & .Fields("map_bestpoint1").Value &
vbCrLf
   tmp_str = tmp_str & "Point สูงสุดอันดับที่ 2 : " & .Fields("map_bestpoint3").Value &
vbCrLf
   tmp_str = tmp_str & "Point สูงสุดอันดับที่ 3 : " & .Fields("map_bestpoint3").Value &
vbCrLf
   tmp_str = tmp_str & "----------------------------------------------------------------------
-----" & vbCrLf
   tmp_str = tmp_str & "เวลาดีอันดับที่ 1 : " & .Fields("map_besttime1").Value & vbCrLf
   tmp_str = tmp_str & "เวลาดีอันดับที่ 2 : " & .Fields("map_besttime3").Value & vbCrLf
   tmp_str = tmp_str & "เวลาดีอันดับที่ 3 : " & .Fields("map_besttime3").Value & vbCrLf
   lb_besttime.Caption = "Best Time : " & .Fields("map_besttime1").Value
   lb_mapprof.Caption = tmp_str
   lb_mapprof.Visible = True
   End With
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub cmd_break_Click()
      If stopL = 0 Then
         gbolRunning = False
         tmStart = False
         stopL = 1
         cmd_break.Caption = "Resume"
      Else
         gbolRunning = True
         tmStart = True
         stopL = 0
         cmd_break.Caption = "Break Game"
      End If
End Sub

Private Sub cmd_chprofile_Click()
   If cmmStart.PortOpen = True Then
      cmmStart.PortOpen = False
   End If
   Call Form_Load
   frmSelProf.Show
   frmSelProf.Top = 768 / 2 + frmSelProf.Height / 2
   frmSelProf.Left = 3400
End Sub

Private Sub cmd_exit_Click()
   Unload Me
End Sub

Private Sub cmd_full_Click()
AMTG.Hide
FS.Show
End Sub

Private Sub cmd_play_Click()
Frame6.Caption = "Mini Map"
tmStart.Enabled = True
'vdo ocx
vcx.Connected = True
vcx.Preview = True
transColor = 0
xx = 0
yy = 0
Image1.Visible = True
vcx.SetVideoFormat 352, 288
End Sub

Private Sub dbcb_mapname_Click()
   create_map (dbcb_mapname.Text)
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Function cPoint(p As Long) As Single
Dim diff, dmin As Long
p = Abs(p)
diff = Abs(2985 - p)
dmin = ((2985 - 1865) / 100 * (1120 - diff)) / 50
If dmin < 0 Then dmin = 0
cPoint = dmin
End Function

Private Sub Form_Load()
  ' กำหนดค่าพื้นฐานการเชื่อมต่อกับ Serial Port
   Image1.Visible = False
  cmmStart.CommPort = 4
  cmmStart.Settings = "9600, N, 8, 1"
  cmmStart.DTREnable = False

   'กำหนด Connection ในการเชื่อมต่อ Data Base
   With Conn
      If .State = adStateOpen Then .Close
      .ConnectionString = strConn & ";Data Source=C:\AMTG.mdb"
      .ConnectionTimeout = 90
      .Open
   End With

   'ค่าเริ่มต้นของ Form AMTG
   AMTG.Left = 760
   AMTG.Top = 930

   AMTG.cmd_chprofile.Caption = "Select Profile"
   lb_mapprof.Visible = False
   cmd_play.Enabled = False
   cmd_hightscore.Enabled = False
   cmd_break.Enabled = False
   cmd_full.Enabled = False
   tmStart.Enabled = False
   txt_name.Text = ""
   txt_name.Enabled = False
   txt_name.BackColor = &HC0C0C0
   txt_point.Text = ""
   txt_point.Enabled = False
   txt_point.BackColor = &HC0C0C0
   txt_bestA.Text = ""
   txt_bestA.Enabled = False
   txt_bestA.BackColor = &HC0C0C0
   txt_bestAA.Text = ""
   txt_bestAA.Enabled = False
   txt_bestAA.BackColor = &HC0C0C0
   dbcb_mapname.Text = ""
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    dbcb_mapname.Enabled = False
    dbcb_mapname.BackColor = &HC0C0C0
    txt_request.Text = ""
    txt_request.Enabled = False
    txt_request.BackColor = &HC0C0C0
    cb_level.Text = ""
    cb_level.Enabled = False
    cb_level.BackColor = &HC0C0C0
    cb_mode.Text = ""
    cb_mode.Enabled = False
    cb_mode.BackColor = &HC0C0C0
    txt_msg.Text = ""
    txt_msg.Enabled = True
    txt_msg.BackColor = &HC0C0C0
    pic_map.BackColor = &HC0C0C0

    'กำหนดค่าการทำงานให้ GDI
    picBackground.Picture = LoadPicture(App.Path & "\image\bgg.gif")
    gbolRunning = True
    hdcTrack = CreateCompatibleDC(picBackground.hdc)
    SelectObject hdcTrack, LoadPicture(App.Path & "\image\blue_lagoon.gif")
    lintSrcPictureX = 726
    lintSrcPictureY = 550
    lb_point = "Point : 000000"
End Sub

Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer)
    ' คืนค่าหน่วยความจำให้แก่ระบบ
    gbolRunning = False
    DeleteDC hdcCamera
    DeleteDC hdcTrack
    End
End Sub


Private Sub tmStart_Timer()
    'จัดการในรับข้อมูลจาก Serial Port ทุก ๆ 1 วินาที
    On Error Resume Next
    Dim tmpx As String
    Dim tmpTime As String
    Static looper As Integer
    Static llngAnimationCheckTime As Long
    Static DataLeft, DataRight As Long
    Static varySum As Long
    Static dataBufLeft, dataBufRight As Long
    Static mainPoint As Long
    Static Ccouter As Integer

    'ตรวจสอบการ Syn กันของสัญญาณ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If timmerCount >= 3 Then
    Ccouter = Ccouter + 5
    ProgressBar1.Value = Ccouter
    tmpx = cmmStart.Input
    txt_msg.Text = txt_msg.Text & "Serial Input                    [ " & tmpx
& " ]" & vbCrLf
    If Len(tmpx) = 15 Or Len(tmpx) = 16 Or Len(tmpx) = 17 Then
      If InStr(1, tmpx, "-", vbBinaryCompare) = 1 Then
          DataLeft = CLng(Left(tmpx, 8))
          Text4.Text = Left(tmpx, 8)
          If InStr(10, tmpx, "-", vbBinaryCompare) = 1 Then
            DataRight = CLng(Right(tmpx, 8))
            Text5.Text = Right(tmpx, 8)
          Else
            DataRight = CLng(Right(tmpx, 7))
            Text5.Text = Right(tmpx, 7)
          End If
      Else
          DataLeft = CLng(Left(tmpx, 7))
          Text4.Text = Left(tmpx, 7)
          If InStr(9, tmpx, "-", vbBinaryCompare) = 1 Then
            DataRight = CLng(Right(tmpx, 8))
            Text5.Text = Right(tmpx, 8)
          Else
            DataRight = CLng(Right(tmpx, 7))
            Text5.Text = Right(tmpx, 7)
          End If
      End If
    Else
        tmpx = ""
        Exit Sub
    End If

    Text3.Text = varySum
    Text1.Text = DataLeft
    Text2.Text = DataRight
    'คำนวณ ค่าเยื้องซ้าย และ เยื้องขวา
    If dataBufLeft = DataLeft And dataBufRight = DataRight Then
      varySum = 0
    Else
      varySum = (DataLeft - DataRight) * Sin(53) / Sin(37) * 1.5
      dataBufLeft = DataLeft
      dataBufRight = DataRight
      If Image1.Left < 800 And varySum < 0 Then
        Image1.Left = Image1.Left
      ElseIf Image1.Left > 5000 And varySum > 0 Then
        Image1.Left = Image1.Left
      Else
        Image1.Left = Image1.Left + varySum
        mainPoint = mainPoint + cPoint(Image1.Left)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            If Len(CStr(mainPoint)) = 1 Then
               lb_point.Caption = "Points : 00000" & CStr(mainPoint)
            ElseIf Len(CStr(mainPoint)) = 2 Then
               lb_point.Caption = "Points : 0000" & CStr(mainPoint)
            ElseIf Len(CStr(mainPoint)) = 3 Then
               lb_point.Caption = "Points : 000" & CStr(mainPoint)
            ElseIf Len(CStr(mainPoint)) = 4 Then
               lb_point.Caption = "Points : 00" & CStr(mainPoint)
            ElseIf Len(CStr(mainPoint)) = 5 Then
               lb_point.Caption = "Points : 0" & CStr(mainPoint)
            ElseIf Len(CStr(mainPoint)) = 6 Then
               lb_point.Caption = "Points : " & CStr(mainPoint)
            Else
               lb_point.Caption = "Points : 999999"
            End If
         End If
      End If

      msec = msec + 10 'บวกเวลา 1 วินาที 10 รอบ 1 รอบ 0.1s
      If msec = 90 Then
         msec = 0
         sec = sec + 1
      End If
      If sec = 59 Then
         min = min + 1
         sec = 0
      End If

      'แสดงค่าของเวลาออกมาในรูปแบบ นาที:วินาที:มิลลิวินาที
      If min = 0 And sec = 0 Then
         tmpTime = "00" & ":" & "00" & ":" & CStr(msec)
      ElseIf sec < 10 And (min >= 0 And min < 10) Then
         tmpTime = "0" & CStr(min) & ":" & "0" & CStr(sec) & ":" & CStr(msec)
      ElseIf msec = 0 And min <> 0 And sec <> 0 Then
         tmpTime = CStr(min) & ":" & CStr(sec) & ":" & "0" & CStr(msec)
      Else
         tmpTime = CStr(min) & ":" & CStr(sec) & ":" & CStr(msec)
      End If

      'แสดงค่าเวลา
      lb_time.Caption = "Time :  " & tmpTime

   If gbolRunning = True Then
         If GetTickCount() - llngAnimationCheckTime >= 100 Then
         llngAnimationCheckTime = GetTickCount()
         picBackground.Refresh
         TransparentBlt picBackground.hdc, 0, 0, 420, 300, hdcTrack, lintSrcPictureX,
lintSrcPictureY, 420, 300, RGB(255, 255, 255)
         'ส่วนที่ใช้ในการปรับตำแหน่งของ กรอบที่เลื่อนในโปรแกรม
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lintSrcPictureY = lintSrcPictureY - 5
        vcx.SetBitmapOverlay picBackground.Picture.Handle, xx, yy, transColor
        'ใช้ looper

        If looper <> 5 Then
                If lintSrcPictureY = 160 Then
                   looper = looper + 1
                   lintSrcPictureY = 340
                End If
        Else
                If lintSrcPictureY = 0 Then
                gbolRunning = False
                tmStart = False
                End If
        End If
        End If
   End If
   If gbolRunning = False Then
        tmStart.Enabled = False
        Image1.Visible = False
        ProgressBar1.Value = 0
        Ccouter = 0
   End If

ElseIf timmerCount = 0 Then
     txt_msg.Text = txt_msg.Text & "Serial Input
[ SYN]" & vbCrLf
     tmpx = cmmStart.Input
ElseIf timmerCount = 1 Then
     txt_msg.Text = txt_msg.Text & "Serial Input
[ SYN]" & vbCrLf
     tmpx = cmmStart.Input
ElseIf timmerCount = 2 Then
     txt_msg.Text = txt_msg.Text & "Serial Input
[ SYN]" & vbCrLf
     tmpx = cmmStart.Input
End If
     timmerCount = timmerCount + 1
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมส่วน Select Profile

```
Option Explicit
Public Conn As New ADODB.Connection
Dim rsProfile As New ADODB.Recordset
Dim skProfile As New ADODB.Recordset
Public rsMap As New ADODB.Recordset
Dim sqlProfile As String

Private Sub cmd_cancel_Click()
Unload Me
End Sub

Private Sub cmd_new_Click()
frm_new.Show
frm_new.Left = 5475
frm_new.Top = 4350
End Sub

Private Sub cmd_ok_Click()
On Error GoTo err
   Dim sqlmap As String
   AMTG.cmd_chprofile.Caption = "Change Profile"
   AMTG.cmd_play.Enabled = True
   AMTG.cmd_hightscore.Enabled = True
   AMTG.cmd_break.Enabled = True
   AMTG.cmd_full.Enabled = True
   AMTG.txt_msg.Text = "Advance Motion The Game Start                    [OK]"
& vbCrLf
   AMTG.cmmStart.PortOpen = True
   'AMTG.tmStart.Enabled = True
   With skProfile
        AMTG.txt_name.Text = .Fields("profile_name").Value
        AMTG.txt_name.Enabled = False
        AMTG.txt_name.BackColor = &HC0C0C0
        AMTG.txt_point.Text = .Fields("profile_point").Value
        AMTG.txt_point.Enabled = False
        AMTG.txt_point.BackColor = &HC0C0C0
        AMTG.txt_bestA.Text = .Fields("profile_bestreca").Value
        AMTG.txt_bestA.Enabled = False
        AMTG.txt_bestA.BackColor = &HC0C0C0
        AMTG.txt_bestAA.Text = .Fields("profile_bestrecaa").Value
        AMTG.txt_bestAA.Enabled = False
        AMTG.txt_bestAA.BackColor = &HC0C0C0
   End With
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        AMTG.txt_msg.Text = AMTG.txt_msg.Text & "Connection Profile Database47
[OK]" & vbCrLf
        AMTG.txt_msg.Text = AMTG.txt_msg.Text & "Serial Communication
Testing                          [OK]" & vbCrLf
        AMTG.txt_msg.Enabled = True
        AMTG.txt_msg.BackColor = &HC0C0C0
        AMTG.pic_map.BackColor = &HC0C0C0
        Unload frmSelProf
        AMTG.create_map ("Blue Lagoon")
err:
   If err.Number = 3265 Then
      MsgBox "Please sellect profile name from List", vbCritical + vbOKOnly, "Error
From Loading Profile"
   ElseIf err.Number = 8002 Then
      AMTG.txt_msg.Text = AMTG.txt_msg.Text & "Serial Communication Testing
[ERROR]" & vbCrLf
      Unload frmSelProf
   End If
End Sub

Private Sub Command1_Click()
Unload Me
End Sub

Private Sub Command3_Click()

End Sub

Private Sub dbcb_profile_Click()
dbcb_profile_DblClick
End Sub

Private Sub dbcb_profile_DblClick()
Dim tmp_data As String
Dim sqlSeek As String

sqlSeek = "SELECT profile.profile_id, profile.profile_name, profile.profile_point,
profile.profile_bestreca, profile.profile_bestrecaa"
sqlSeek = sqlSeek & " FROM Profile "
sqlSeek = sqlSeek & "WHERE profile.profile_name ='" & dbcb_profile.Text & "'"

With skProfile
   If .State = adStateOpen Then .Close
     .ActiveConnection = Conn
     .CursorType = adOpenForwardOnly
     .CursorLocation = adUseClient
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    .Open sqlSeek

      lbprof_id.Caption = "รหัสโปรไฟล์ :                    "
& .Fields("profile_id").Value
      lbprof_name.Caption = "ชื่อโปรไฟล์ :                    "
& .Fields("profile_name").Value
      lbprof_point.Caption = "แต้มสะสม :                    "
& .Fields("profile_point").Value
      lbprof_bestA.Caption = "เวลา และ จำนวนแต้มสูงสุด บนสนาม A :          "
& .Fields("profile_bestreca").Value
      lbprof_bestAA.Caption = "เวลา และ จำนวนแต้มสูงสุด บนสนาม A A:          "
& .Fields("profile_bestrecaa").Value
End With
End Sub

Private Sub Form_Load()

   lbprof_id.Caption = "รหัสโปรไฟล์ :"
   lbprof_name.Caption = "ชื่อโปรไฟล์ :"
   lbprof_point.Caption = "แต้มสะสม :"
   lbprof_bestA.Caption = "เวลา และ จำนวนแต้มสูงสุด บนสนาม A :"
   lbprof_bestAA.Caption = "เวลา และ จำนวนแต้มสูงสุด บนสนาม A A:"

   sqlProfile = "SELECT * FROM profile ORDER BY profile_id"
   Call refresh_list
End Sub
Public Sub refresh_list()
   With Conn
      If .State = adStateOpen Then .Close
     .ConnectionString = strConn & ";Data Source=C:\AMTG.mdb"
     .ConnectionTimeout = 90
     .Open
   End With
   With rsProfile
      If .State = adStateOpen Then .Close
     .ActiveConnection = Conn
     .CursorType = adOpenForwardOnly
     .CursorLocation = adUseClient
     .Open sqlProfile
   End With
   With dbcb_profile
          Set .RowSource = rsProfile
        .ListField = "profile_name"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        .BoundColumn = "profile_id"
    End With
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมส่วน Create New Profile

```
Option Explicit
Dim Conn As New ADODB.Connection
Dim comAdd As New ADODB.Command
Dim rsCount As New ADODB.Recordset

Private Sub Command1_Click()
Dim sqlAdd As String
   Conn.BeginTrans
   sqlAdd = "INSERT INTO
profile(profile_id,profile_name,profile_point,profile_bestreca,profile_bestrecaa) "
   sqlAdd = sqlAdd & "VALUES('" & txt_id.Text & "','" & txt_name.Text &
"','0','(00:00:00:00)(0000)','(00:00:00:00)(0000)')"
   'Debug.Print sqlAdd
   With comAdd
      .ActiveConnection = Conn
      .CommandType = adCmdText
      .CommandText = sqlAdd
      .Execute
  End With
 Conn.RollbackTrans
  frmSelProf.Conn.Close
  frmSelProf.refresh_list
  Unload Me
End Sub

Private Sub Form_Load()
Dim sqlCount As String
With Conn
      If .State = adStateOpen Then .Close
      .ConnectionString = strConn & ";Data Source=C:\AMTG.mdb"
      .CommandTimeout = 90
      .Open
End With

sqlCount = "SELECT profile.profile_id"
sqlCount = sqlCount & " FROM profile"

With rsCount
      If .State = adStateOpen Then .Close
      .ActiveConnection = Conn
      .CursorType = adOpenForwardOnly
      .CursorLocation = adUseClient
      .Open sqlCount
      txt_id.Text = CInt(.RecordCount) + 1
End With
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้