สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เดินทางไปในโลกเสมือนด้วยจักรยาน

VIRTUAL WORLD NAVIGATION SYSTEM USING A BICYCLE

นายนนท์ คนึงสุขเกษม

นายบุญชู สุเมธาอักษร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เลขหมู่.....

เลขทะเบียน 49977

วัน,เดือน,ปี 16 เม.ย. 2547

.b.....

.i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดินทางไปในโลกเสมือนด้วยจักรยาน

VIRTUAL WORLD NAVIGATION SYSTEM USING A BICYCLE

โดย

นายขนนท์ คนึงสุขเกษม

นายบุญชู สุเมธาอักษร

อาจารย์ที่ปรึกษา

ดร.สมศักดิ์ วลัยรัชต์

อ.สมเกียรติ วังศิริพิทักษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2545

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เดินทางไปในโลกเสมือนด้วยจักรยาน

VIRTUAL WORLD NAVIGATION SYSTEM USING A BICYCLE

ผู้จัดทำ

1. นาย นนท์ คนึงสุขเกษม รหัสประจำตัว 42010157
2. นาย บุญชู สุเมธาอักษร รหัสประจำตัว 42010180

_____ อาจารย์ที่ปรึกษา

(ดร.สมศักดิ์ วลัยรัชต์)

_____ อาจารย์ที่ปรึกษาร่วม

(อ.สมเกียรติ วังศิริพิทักษ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดินทางไปในโลกเสมือนด้วยจักรยาน

นายนนท์ คนึงสุขเกษม 42010157
นายบุญชู สุเมธาอักษร 42010180
ดร.สมศักดิ์ วลัยรัชต์ อาจารย์ที่ปรึกษา
อ.สมเกียรติ วังศิริพิทักษ์ อาจารย์ที่ปรึกษาร่วม
ปีการศึกษา 2545

บทคัดย่อ

ในปัจจุบันเทคโนโลยีการประมวลผลคอมพิวเตอร์ก้าวหน้าไปอย่างรวดเร็ว เราสามารถสร้างโลกของความเสมือนจริง(Virtual Reality World) ซึ่งจำลองลักษณะการทำงานที่ใกล้เคียงกับการทำงานในโลกจริง (Real World) แต่ด้วยข้อจำกัดของอุปกรณ์ติดต่อสื่อสารของระบบคอมพิวเตอร์ ซึ่งโดยทั่วไปแล้วก็คือ คีย์บอร์ดและเมาส์ ยังไม่สามารถช่วยให้เราทำงานในโลกเสมือนได้อย่างเป็นธรรมชาติ ปริญญานิพนธ์ฉบับนี้จึงขอนำเสนอการสร้างโลกเสมือนและพัฒนาอุปกรณ์เพื่อควบคุมการเดินทาง (Navigation system) ในโลกเสมือนได้ในลักษณะเดียวกับโลกจริง โดยอาศัยพาหนะชนิดหนึ่งนั่นก็คือรถจักรยาน

เนื้อหาในปริญญานิพนธ์ฉบับนี้แบ่งเป็น 2 ส่วนคือส่วนของการพัฒนาซอฟท์แวร์ และส่วนของการพัฒนาฮาร์ดแวร์ โดยในส่วนซอฟท์แวร์จะเป็นการสร้างโลกเสมือนในรูปแบบคอมพิวเตอร์กราฟิก 3 มิติที่จำลองโลกเสมือนขึ้น และในส่วนฮาร์ดแวร์เป็นส่วนที่พัฒนาอุปกรณ์ตรวจจับติดตั้งกับรถจักรยานรับค่าสัญญาณต่างๆ นำไปแสดงผลที่สอดคล้องกับในโลกเสมือน นอกจากนั้นยังมีส่วนสร้างการตอบสนองที่รับค่ากลับจากโลกเสมือนอีกด้วย

ระบบนี้พัฒนาขึ้นบนระบบปฏิบัติการวินโดว์ และใช้ชุดคำสั่งเพื่อการพัฒนาโปรแกรม DirectX ภาษาคอมพิวเตอร์ในการพัฒนาโปรแกรมคือ Visual C++ ลักษณะการทำงานระบบนี้อนุญาตผู้ใช้หนึ่งคนถีบรถจักรยาน สามารถบังคับเลี้ยวซ้ายขวา ผู้ใช้จะเห็นภาพโลกเสมือนบนจอภาพคอมพิวเตอร์ซึ่งเป็นภาพคอมพิวเตอร์กราฟิก 3 มิติเคลื่อนที่ตามที่ผู้ใช้บังคับจักรยาน และผู้ใช้สามารถรับรู้ผลตอบสนองกลับจากโลกเสมือนได้ เช่นกรณีรถจักรยานชนวัตถุต่างๆ ในโลกเสมือนจะเกิดการสั่นสะเทือนขึ้นที่ตัวรถจักรยาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VIRTUAL WORLD NAVIGATION SYSTEM USING A BICYCLE

Nont Kanungsukkasem

Boonchoo Sumatauksorn

Dr. Somsak Walairacht Advisor

Somkiat Wangsiripitak Co-Advisor

ABSTRACT

Nowadays, Computation power of the computer has been rapidly increased. The simulated virtual world can be easily created by personal computer. However, conventional input/output devices, such as keyboards or mice, do not allow users to control the virtual world as natural as in the real world. Therefore, in this thesis are propose a navigation system for the simulated virtual world using a bicycle. The user can navigate into the 3-D virtual world as he or she did in the real world.

The content of the thesis can be divided into 2 main parts. First part is describing about the development of the software. The second part explains about the implementation of hardware system including the part of generating user's feedback.

The system is designed based on Windows operating system with DirectX library for graphics display. Software is implemented by using Microsoft Visual C++. One user is allowed to operate the system by riding on a real bicycle. He or she can rides quickly or slowly and turn left or right as in the real world. On the computer display the user can see the view of the virtual world changing according to how the user is riding the bicycle. When the virtual bicycle collides with the virtual objects, the user can sense which represents a kind of feedback occurred from the virtual world.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้ด้วยความกรุณาของ อาจารย์ ดร.สมศักดิ์ วลัยรัชต์ อาจารย์ที่ปรึกษาโครงงาน ซึ่งได้ให้คำปรึกษา ข้อชี้แนะ และความช่วยเหลือในหลายสิ่งหลายอย่าง จนกระทั่งลุล่วงไปได้ด้วยดี คณะผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

ขอกราบขอบพระคุณ คณาจารย์ภาควิชาคอมพิวเตอร์ทุกท่านที่ได้ให้ความรู้ตลอดการศึกษาที่ผ่านมา ขอขอบคุณพี่ๆ และเจ้าหน้าที่ห้องฮาร์ดแวร์ที่ให้ความกรุณาให้ยืมอุปกรณ์ต่างๆ ในการต่อวงจรฮาร์ดแวร์

ขอบคุณและขอบใจ “เท็น”, “ตู่”, “อ้อย” และ “เบิ้ม” ที่ช่วยให้แนวความคิด คำแนะนำและความช่วยเหลือในการออกแบบวงจรฮาร์ดแวร์เพื่อให้ระบบทำงานได้ตามต้องการ รวมถึงเพื่อนๆ ทุกคนที่คอยถามไถ่ด้วยความห่วงใย รวมถึงผู้มีพระคุณทุกท่านที่มิได้เอ่ยนามไว้ ณ ที่นี้

สุดท้ายนี้กราบขอบพระคุณ บิดา มารดา ผู้ให้ทุกสิ่งทุกอย่างกับผู้วิจัยเสมอมา

นนท์ คนึงสุขเกษม

บุญชู สุเมธาอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปภาพ	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	1
1.1 แนะนำเรื่องราวความเสมือนจริง	1
1.2 การประยุกต์ใช้ความเสมือนจริง	1
1.3 วัตถุประสงค์ของโครงงาน	2
1.4 ขอบเขตของโครงงาน	2
1.5 อุปกรณ์และซอฟท์แวร์ที่ใช้	4
1.6 วิธีการดำเนินงาน	4
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	5
2.1 ชุดคำสั่งสำหรับการพัฒนาโปรแกรม : DirectX SDK	5
2.1.1 DirectX คืออะไร	5
2.1.2 DirectX SDK	6
2.1.3 ส่วนประกอบของ DirectX 8.0 SDK	6
2.2 ชุดคำสั่งในการจัดการด้านกราฟิก 3 มิติ: Direct3D SDK	6
2.2.1 ระบบพิกัด 3 มิติ	6
2.2.2 Texture	7
2.3 ทฤษฎีทางด้านฮาร์ดแวร์	8
2.3.1 ความรู้เบื้องต้นเกี่ยวกับพอร์ตขนาน	8
2.3.1.1 ลักษณะทางกายภาพของพอร์ตขนาน	9
2.3.1.2 การนำพอร์ตขนานไปใช้งาน	13
2.3.2 ลักษณะการทำงานของเซ็นเซอร์	14
2.3.3 ลักษณะการทำงานของตัวต้านทานปรับค่าได้	15
บทที่ 3 การพัฒนาระบบ	17
3.1 การพัฒนาในส่วนฮาร์ดแวร์	17
3.1.1 อุปกรณ์ที่ใช้ในการพัฒนาทั้งหมด	18
3.1.2 ส่วนตรวจสอบการบังคับรถจักรยาน	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 ส่วนสร้างปฏิกิริยาตอบสนอง 22
3.1.4 ส่วนติดต่อระหว่างคอมพิวเตอร์กับฮาร์ดแวร์ 23
3.2 การพัฒนาด้านซอฟท์แวร์ 25
3.2.1 วิธีการโหลด .x file เข้ามาในโปรแกรม 25
3.2.2 การรับข้อมูลจากตัวรถจักรยาน 26
3.2.3 การคำนวณทิศทางการเคลื่อนที่ 26
3.2.4 การตรวจสอบการขึ้นลงสะพาน 31
3.2.5 การตรวจสอบการชน 31
3.2.6 การตรวจสอบการเคลื่อนที่ผ่านเกาะกลางถนน 32
บทที่ 4 การทำงานโดยรวมทั้งหมด 34
4.1 ส่วนประกอบภายในโลก 3 มิติ 35
4.2 ลักษณะของโลก 3 มิติ 35
4.3 การเดินทางภายในโลก 3 มิติ 35
บทที่ 5 สรุป 38
5.1 ปัญหาและแนวทางแก้ไข 38
5.1.1 ปัญหาที่เกิดขึ้นในส่วนของฮาร์ดแวร์ 38
5.1.2 ปัญหาที่เกิดขึ้นในส่วนของซอฟท์แวร์ 39
5.2 แนวทางการพัฒนาต่อ 39
บรรณานุกรม 40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้าที่

รูปที่ 1-1 แสดงตัวอย่างอุปกรณ์อินพุต VR 1

รูปที่ 1-2 แสดงตัวอย่างการนำ VR ไปประยุกต์ใช้ 2

รูปที่ 2-1 ตัวอย่างแสดงตำแหน่งของสะพานในโลก 3 มิติ โดยจุดสีแดงมีพิกัดที่ (-133, 8,-93) 7

รูปที่ 2-2 ก ก่อนทำ Texture mapping 8

รูปที่ 2-2 ข หลังทำ Texture mapping 8

รูปที่ 2-3 แสดงลักษณะทางกายภาพของพอร์ตขนาน 10

รูปที่ 2-4 วงจรภายในของพอร์ตคาต้า 11

รูปที่ 2-5 วงจรภายในของพอร์ตคอนโทรล 12

รูปที่ 2-6 วงจรภายในของพอร์ตแสดงสถานะ 13

รูปที่ 2-7 การทำงานของเซ็นเซอร์แสง 14

รูปที่ 2-8 ลักษณะทางกายภาพและการทำงานของเซ็นเซอร์แสง 15

รูปที่ 2-9 ลักษณะทางกายภาพและโครงสร้างภายในของตัวต้านทานปรับค่าได้ 15

รูปที่ 3-1 รูปการติดตั้งอุปกรณ์ที่ได้ออกแบบไว้ 17

รูปที่ 3-2 แสดงรถจักรยานและอุปกรณ์ทั้งหมด 17

รูปที่ 3-3 โครงสร้างส่วนตรวจสอบการบังคับรถจักรยาน 19

รูปที่ 3-4 วงจรตรวจสอบความเร็วของการปั่นรถจักรยาน 19

รูปที 3-5 กระดาษที่ติดอยู่ที่ซี่โครงล้อ และตัวเซ็นเซอร์ 20

รูปที่ 3-6 วงจรเซ็นเซอร์ 20

รูปที 3-7 วงจรตรวจสอบการเลี้ยวของแฮนด์รถจักรยาน 21

รูปที่ 3-8 อุปกรณ์ตรวจสอบการเลี้ยวหลังจากต่อวงจรเสร็จ 21

รูปที 3-9 ตัวต้านทานปรับค่าได้ที่ติดบนโครงรถจักรยาน และมีสายยางพันอยู่กับคอรถจักรยาน 22

รูปที่ 3-10 โครงสร้างส่วนสร้างปฏิกิริยาตอบสนอง 22

รูปที่ 3-11 วงจรควบคุมมอเตอร์ 22

รูปที่ 3-12 อุปกรณ์ควบคุมมอเตอร์ 23

รูปที่ 3-13 มอเตอร์ไฟฟ้า 23

รูปที่ 3-14 วงจรบัฟเฟอร์ 24

รูปที่ 3-15 อุปกรณ์บัฟเฟอร์ 24

รูปที่ 3-16 ตัวอย่างรูปมุมมองบุคคลที่ 1 27

รูปที่ 3-17 แสดงถึงมุมมองเมื่อมีการเลี้ยวขวา และซ้าย 27

รูปที่ 3-18 ก มุมรอบแกน y กรณีของแกน x กับ z 28

รูปที่ 3-18 ข มุมรอบแกน y กรณีของแกน y กับ z 28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3-19 การกำหนดความลาดชันของสะพาน 31
รูปที่ 3-20 วิธีตรวจสอบการชน 31
รูปที่ 3-21 แสดงภาพก่อนชนและขณะชนรั้ว 32
รูปที่ 3-22 การพิจารณาตำแหน่งรถจักรยานเมื่อผ่านเกาะกลางถนน 32
รูปที่ 3-23 แสดงให้เห็นถึงมุมกล้องขณะเคลื่อนที่ผ่านเกาะกลางถนน 33
รูปที่ 4-1 การใช้งาน VR Bicycle 34
รูปที่ 4-2 แสดงให้เห็นถึงโลก 3 มิติทั้งหมดจากมุมมองด้านบน 34
รูปที่ 4-3 แสดงตำแหน่งของส่วนประกอบต่าง ๆ 35
รูปที่ 4-4 จุดวงกลมสีขาวแสดงถึงตำแหน่งเริ่มต้นของรถจักรยาน 35
รูปที่ 4-5 รั้วไม้ 36
รูปที่ 4-6 กำแพง 36
รูปที่ 4-7 เกาะกลางถนน 36
รูปที่ 4-8 เมื่อเจอสะพานสามารถขึ้นได้ และสามารถชนขอบสะพานได้ 37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้าที่

ตารางที่ 1-1 ความสัมพันธ์ระหว่างผลลัพธ์ที่ต้องการกับการเคลื่อนที่ของผู้ใช้ 3
ตารางที่ 2-1 แสดงแอดเดรสของพอร์ตขนาน 13
ตารางที่ 2-2 แสดงความสัมพันธ์ระหว่างตำแหนงขากับ Register bit 14
ตารางที่ 3-1 แสดงการเชื่อมระหว่างบิตของพอร์ตขนาน กับส่วนต่างๆของอุปกรณ์ภายนอก 25

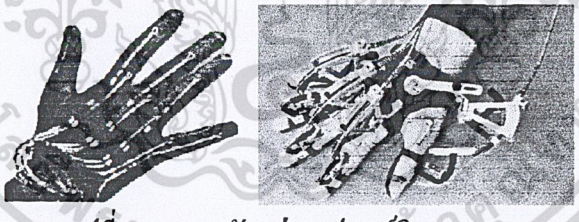
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 แนะนำเรื่องราวความเสมือนจริง

ความเสมือนจริงมาจากคำภาษาอังกฤษว่า Virtual Reality หรือเรียกสั้นๆ ว่า VR โดยที่ VR ถือได้ว่าเป็นศาสตร์แขนงหนึ่งที่รวมเอาความรู้และเทคโนโลยีต่างๆ เข้าไว้ด้วยกัน เช่น การแสดงภาพทั้งในรูปแบบ 2 มิติ และ 3 มิติ การแสดงเสียง และการได้ยิน อุปกรณ์ติดต่อเช่น เมาส์ จอยสติก หรือถุงมือ เป็นต้น "VR คือวิธีการที่มนุษย์จะใช้ในการมองเห็น จัดการ และโต้ตอบกับคอมพิวเตอร์และข้อมูลที่มีความซับซ้อนสูง" ส่วนของการมองเห็นโดยทั่วไปคือ การมองเห็น ภาพกราฟิกที่สร้างด้วยคอมพิวเตอร์ (Computer Graphic) การได้ยินเสียงหรือส่วนรับความสัมผัส ความรู้สึก ที่เป็นผลลัพธ์ออกมาจากโลกเสมือนในคอมพิวเตอร์ ซึ่งอาจจะเป็นโมเดล 3 มิติ ที่จำลองขึ้นตามหลักการทางวิทยาศาสตร์ ผู้ใช้สามารถโต้ตอบกับโลกเสมือนและสามารถจัดการกับวัตถุในโลกได้โดยตรง อาจมีการจำลองการเคลื่อนไหวทางกายภาพ VR มีบทบาทที่สำคัญมากในการที่มนุษย์สามารถโต้ตอบกับโลกเสมือนได้อย่างเป็นธรรมชาติเช่นเดียวกับการโต้ตอบในโลกจริง

ปัจจัยหลักของการทำการโต้ตอบกับโลกเสมือน คือ การติดตามตำแหน่งของวัตถุต่างๆ ในโลกจริง เช่น มือ โดยมีหลายวิธีด้วยกันในการติดตามและควบคุม อุปกรณ์ VR ที่พบเห็นบ่อยที่สุดคือ ถุงมือที่ติดเซนเซอร์ตามจุดต่างๆ เป็นต้น

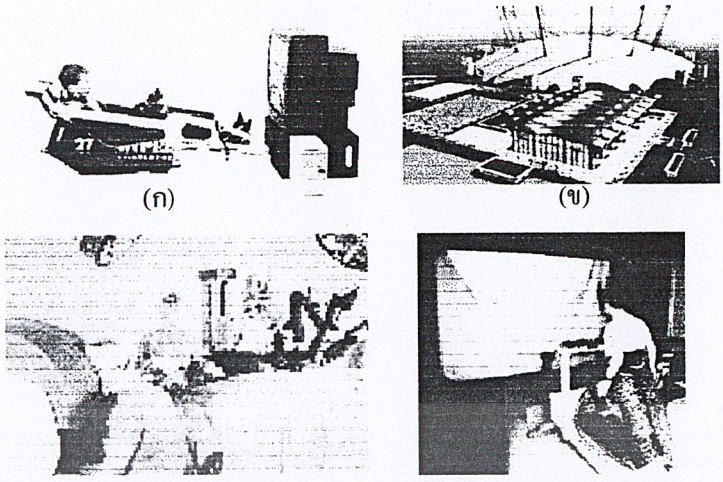


รูปที่ 1-1 แสดงตัวอย่างอุปกรณ์อินพุต VR

1.2 การประยุกต์ใช้ VR

การนำ VR ไปประยุกต์ใช้งานมีด้วยกันหลายสาขา ตั้งแต่การจำลองการทำงานที่สลับซับซ้อน ถึงการก่อสร้างทางสถาปัตยกรรม การทำงานในที่ที่เป็นไปไม่ได้ หรือแม้แต่ในเกม การประยุกต์ใช้จะสร้างโลกเสมือนให้มีความใกล้เคียงกับโลกจริงมากที่สุดเท่าที่จะมากได้ เช่น โมเดลทางสถาปัตยกรรม หรือในบางงานได้เสนอรูปแบบการมองเห็นที่ไม่สามารถมองเห็นได้จากโลกจริง เช่น การจำลองทางวิทยาศาสตร์ หรือ ระบบควบคุมการทำงานในอากาศ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) (ข)

(ค) (ง)

รูปที่ 1-2 แสดงตัวอย่างการนำ VR ไปประยุกต์ใช้

(ก) เกม (ข)การจำลองโมเดลทางสถาปัตยกรรม

(ค) การผ่าตัดทางไกล (ง) การจำลองการเคลื่อนที่บนดาวอังคารของ NASA

โครงงานนี้เป็นตัวอย่างหนึ่งของการนำ VR มาประยุกต์ใช้งาน ซึ่งจะสร้างการแสดงผลกราฟิก 3 มิติ เพื่อการมองเห็น และอาศัยพาหนะชนิดหนึ่ง คือ รถจักรยานเพื่อการเคลื่อนที่ไปในโลกเสมือนและตอบโต้กับผู้ใช้ โดยภาพที่แสดงในโลกเสมือนจะเคลื่อนที่เปลี่ยนแปลงไปตามการถีบรถจักรยาน และหมุนเลี้ยวของแฮนด์จักรยาน และผู้ขับขี่จะได้รับการตอบสนองจากโลกเสมือนมายังรถจักรยานด้วยการสั่นเมื่อมีการชนกับวัตถุต่างๆ ในโลกเสมือน

1.3 วัตถุประสงค์ของโครงงาน

1. เพื่อศึกษาการจำลองการเคลื่อนที่ในโลก 3 มิติ
2. เพื่อสร้างอุปกรณ์ในการติดต่อกับโลก 3 มิติ
3. เพื่อศึกษาและพัฒนาโปรแกรมในส่วนของการแสดงภาพกราฟิก 3 มิติด้วย Direct3D
4. เพื่อศึกษาและพัฒนาการติดต่อผ่านทางพอร์ตขนาน
5. ศึกษาและพัฒนาโปรแกรมและฮาร์ดแวร์ส่วนควบคุมตัวตรวจจับสัญญาณต่าง ๆ

1.4 ขอบเขตของโครงงาน

โครงงานนี้เป็นการสร้างโลกเสมือนจริงที่สามารถติดต่อกับอุปกรณ์ภายนอกได้ผ่านทางพอร์ตขนาน ซึ่งจะนำเสนอโลกเสมือนจริงในรูปของกราฟิก 3 มิติโดยผ่านชุดคำสั่งสำหรับการพัฒนาโปรแกรม DirectX และเพื่อให้ผู้ใช้รู้สึกเช่นเดียวกับการเคลื่อนที่ในโลกนั้นจริงๆ จึงใช้อุปกรณ์ชนิดเดียวกันกับในโลกเสมือน นั่นคือรถจักรยาน โดยผู้ใช้สามารถถีบรถจักรยาน สามารถบังคับเลี้ยวซ้ายเลี้ยวขวา โดยที่การบังคับรถจักรยานจะไปส่งผลให้ภาพที่แสดงขึ้นบนจอภาพนั้นเปลี่ยนไปตามการบังคับนั้นๆ และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนั้นยังมีการแสดงผลตอบสนองมาจากโลกเสมือนกลับมาที่จักรยานโดยการทำให้เกิดความสั่นสะเทือนที่ตัวรถจักรยาน

วิธีการที่ใช้แสดงการเคลื่อนที่ คือการติดอุปกรณ์ตรวจจับเข้ากับรถจักรยานเพื่อตรวจสอบการหมุนของล้อรถจักรยานและการเลี้ยวของแฮนด์จักรยาน แล้วส่งสัญญาณไปประมวลผลโดยโปรแกรมที่พัฒนาขึ้นด้วยภาษา C++ บนระบบปฏิบัติการวินโดว์ การแสดงผลจะปรากฏบนจอภาพคอมพิวเตอร์ และที่ตัวจักรยาน จะเกิดการสั่นสะเทือนโดยใช้มอเตอร์เมื่อเกิดการชนกับวัตถุต่างๆ ในโลกเสมือน

ระบบที่พัฒนาขึ้นสามารถแบ่งออกเป็น 3 ส่วนหลักคือ : ส่วนของโลกเสมือน, ส่วนของการเคลื่อนที่และ ส่วนของการตอบสนอง

1.4.1 ส่วนของโลกเสมือน

โลกเสมือนจะแสดงในรูปแบบของคอมพิวเตอร์กราฟิก 3 มิติ ประกอบไปด้วย ถนน, ตึก, กำแพง, สะพาน โดยโลกเสมือนได้จำลองให้มีลักษณะเป็นเมืองเล็กๆ เมืองหนึ่ง

1.4.2 ส่วนของการเคลื่อนที่

ส่วนนี้จำแนกการเคลื่อนที่ออกได้เป็น

- การเคลื่อนที่ไปข้างหน้า
- การหยุดการเคลื่อนที่
- การเปลี่ยนทิศทางการเคลื่อนที่ (เลี้ยวซ้าย/ขวา)
- การเพิ่มและลดความเร็วของการเคลื่อนที่

โดยได้กำหนดชุดของการเคลื่อนที่ที่ผู้ใช้กระทำในโลกจริง และผลลัพธ์ที่เกิดขึ้นในโลกเสมือน โดยตารางที่ 1 นี้แสดงถึง ความสัมพันธ์ระหว่างผลลัพธ์ที่ต้องการในโลกเสมือนกับสภาวะแวดล้อมจริง และการเคลื่อนที่ที่ต้องทำเพื่อให้ได้มาซึ่งผลลัพธ์นั้น

ผลลัพธ์ที่ต้องการ	การเคลื่อนที่ของผู้ใช้
เคลื่อนที่ไปข้างหน้า	ถีบจักรยาน
หยุดการเคลื่อนที่	หยุดถีบจักรยาน
เปลี่ยนทิศทางการเคลื่อนที่	หมุนคันบังคับ (handlebar)
เพิ่มและลดความเร็วของการเคลื่อนที่	ถีบให้เร็วขึ้นหรือช้าลง

ตารางที่ 1-1 ความสัมพันธ์ระหว่างผลลัพธ์ที่ต้องการกับการเคลื่อนที่ของผู้ใช้

1.4.3 ส่วนของการตอบสนอง

ปฏิกิริยาตอบสนองนั้นอยู่ในรูปของการสั่นสะเทือนที่ตัวจักรยาน คือ เมื่อผู้ใช้เคลื่อนที่ไปชนกับวัตถุหรือสิ่งกีดขวาง จะเกิดการสั่นขึ้นที่รถจักรยาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 อุปกรณ์และซอฟท์แวร์ที่ใช้

- คอมพิวเตอร์ส่วนบุคคลระดับ Pentium 4 ความเร็วไม่น้อยกว่า 1GHz
  - หน่วยความจำไม่น้อยกว่า 256 MB
  - การ์ดจอแสดงผลกราฟิกแบบเร่งความเร็ว 3 มิติ
  - พอร์ตสื่อสารแบบขนาน
- MS Windows 2000
- Microsoft Visual C++ V.6.0
- DirectX SDK
- รถจักรยาน
- อุปกรณ์อิเล็กทรอนิกส์ต่างๆ เช่น
  - ตัวต้านทานปรับค่าได้ (Potentiometer)
  - เซ็นเซอร์แสง (Optical sensor)
  - มอเตอร์
  - Relay

1.6 เนื้อหาของวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้กล่าวถึงเนื้อหาสาระต่างๆ โดยแบ่งออกเป็น 5 บทคือ บทที่ 1 นี้เป็นบทนำในส่วนต่างๆ บทที่ 2 กล่าวถึงทฤษฎีที่เกี่ยวข้องทั้งซอฟท์แวร์และฮาร์ดแวร์ บทที่ 3 จะกล่าวถึงการพัฒนาระบบทั้งหมด บทที่ 4 แสดงผลการทำงานของระบบ และสุดท้าย บทที่ 5 คือ บทสรุป ซึ่งจะกล่าวถึงปัญหาและอุปสรรค การแก้ไข และแนวการพัฒนาโครงงานนี้ต่อไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ชุดคำสั่งสำหรับการพัฒนาโปรแกรม : DirectX SDK

การสร้างเกมสำหรับระบบปฏิบัติการวินโดวส์ในปัจจุบัน ได้มีการพัฒนาชุดคำสั่งขึ้นมา เพื่อใช้อำนวยความสะดวกในการเขียนเกม ซึ่งเป็นชุดคำสั่งที่ทำให้ผู้เขียนโปรแกรมเรียกใช้ความสามารถด้านฮาร์ดแวร์ของระบบได้อย่างเต็มประสิทธิภาพ ไม่ว่าจะเป็นภาพ เสียง หรือการควบคุมเกม ชุดคำสั่งนี้มีชื่อว่า DirectX

ก่อนที่จะมี DirectX เกิดขึ้น นักพัฒนาเกมคอมพิวเตอร์สำหรับดอส หรือวินโดวส์ จะต้องเขียนเกมคอมพิวเตอร์ให้รู้จักกับฮาร์ดแวร์ ซึ่งมีอยู่มากมาย ซึ่งในกรณีที่มีฮาร์ดแวร์ตัวใหม่เกิดขึ้น อาจเกิดปัญหาความไม่สนับสนุนกันระหว่างเกมคอมพิวเตอร์กับฮาร์ดแวร์ตัวใหม่นั้น เมื่อเป็นเช่นนี้ ไมโครซอฟต์บริษัทยักษ์ใหญ่ทางด้านระบบปฏิบัติการคอมพิวเตอร์จึงได้ทำการพัฒนาเทคโนโลยีที่มีความสามารถในการเป็นสื่อกลางการติดต่อระหว่างเกมคอมพิวเตอร์ หรือโปรแกรมมัลติมีเดียต่างๆ กับฮาร์ดแวร์ขึ้นมาโดยใช้ชื่อว่า “DirectX”

2.1.1 DirectX คืออะไร

DirectX ตามความหมายจะหมายถึง ไลบรารีคำสั่ง (Run Time Library) ที่ช่วยทำงานด้านมัลติมีเดียกราฟิกโดยตัว DirectX Foundation จะมีส่วนประกอบที่เรียกว่า HAL (Hardware Abstraction Layer) จะใช้ซอฟต์แวร์ในการตรวจสอบความสามารถของฮาร์ดแวร์ ที่อยู่ในเครื่องคอมพิวเตอร์นั้นอย่างอัตโนมัติ แล้วนำมากำหนดพารามิเตอร์ของแอพพลิเคชัน ให้ตรงตามความเหมาะสมระหว่างเกมคอมพิวเตอร์ หรือโปรแกรมมัลติมีเดีย กับ ไดรเวอร์ของฮาร์ดแวร์ ที่มีส่วนเกี่ยวข้องกับเกมหรือโปรแกรมนั้น ทำให้การประมวลผลโปรแกรมทำงานได้เร็วขึ้น เนื่องจากขั้นตอนต่างๆ จะถูกนำไปประมวลผลโดยตรง ไม่ต้องอาศัยตัวกลางอย่างเช่น GDI (Graphic Device Interface) ทำให้นักพัฒนาเกมคอมพิวเตอร์สามารเขียนโปรแกรมให้สื่อสารกับ DirectX เท่านั้นก็เพียงพอ นอกจากนี้ DirectX Foundation ยังมีส่วนประกอบที่เรียกว่า HEL (Hardware Emulation Layer) ทำให้เราสามารถใช้โปรแกรมมัลติมีเดีย หรือโปรแกรมที่เกี่ยวข้องกับ 3 มิติ บนฮาร์ดแวร์ที่ไม่สนับสนุนใช้งานด้าน 3 มิติ โดยจะทำการจำลองความสามารถบางอย่างที่ฮาร์ดแวร์ตัวนั้นไม่มี ให้สามารถใช้งานได้กับโปแกรมที่ต้องการได้ แม้จะมีข้อเสียอยู่บ้างตรงที่อาจทำให้การทำงานช้าลงบ้างก็ตาม แต่ก็คุ้มค่ากับความสามารถของ DirectX ที่มีอยู่ในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 DirectX SDK

ก่อนอื่นเรามาทำความเข้าใจกับ 2 ส่วนประกอบสำคัญ ดังนี้

1) DirectX Runtime Library
2) DirectX SDK (Software Development Kit)

DirectX Runtime เป็นส่วนประกอบของ DirectX ที่จะทำให้เราสามารถเล่นเกมได้ ซึ่งจะถูกติดตั้งลงในระบบปฏิบัติการวินโดวส์โดยอัตโนมัติ

DirectX SDK เป็นชุดคำสั่งที่ออกแบบมาใช้ร่วมกับตัวแปลภาษาหลายๆ ภาษา เช่น Visual C++ หรือ Visual Basic ซึ่งเราสามารถพัฒนาเกมคอมพิวเตอร์หรือโปรแกรมมัลติมีเดียเองได้โดยการศึกษาวิธีการเขียนและหน้าที่ของคำสั่งต่างๆ

2.1.3 ส่วนประกอบของ DirectX 8.0 SDK

DirectX 8.0 SDK ถูกพัฒนาให้มีความสามารถเพิ่มขึ้นจากเวอร์ชันก่อนๆ โดยสามารถแบ่งออกเป็นส่วนประกอบหลักๆ ได้ดังนี้

1) DirectX Graphics เป็นการรวมกันระหว่าง DirectDraw และ Direct3D โดยมีจุดเด่นตรงที่การประมวลผลกราฟิกแบบขนาน และสามารถจัดการกับกราฟิกให้มีความสมจริงกว่าเวอร์ชันที่ผ่านๆ มา ทำให้ผู้พัฒนาเกมสามารถพัฒนาเกม 3 มิติ ได้ง่ายขึ้น
2) DirectX Audio ประกอบด้วย DirectSound และ DirectMusic ทำหน้าที่เกี่ยวกับเสียงทั้งหมด
3) DirectInput ทำหน้าที่คอยสนับสนุนอุปกรณ์ควบคุม เช่น จอยสติก และอุปกรณ์ควบคุมเกมอื่นๆ ทำให้ผู้พัฒนาเกมสามารถใช้ประโยชน์ได้ง่ายขึ้น
4) DirectPlay ช่วยให้สามารถที่จะเล่นเกมได้ครั้งละหลายๆ คนผ่านระบบเครือข่าย
5) DirectShow ใช้ในการแสดงผลด้านมัลติมีเดีย ประเภทภาพวีดีโอเป็นหลัก

ในโครงงานนี้เราได้ใช้ส่วนประกอบของ DirectX คือ Direct3D ในการสร้างโลกเสมือน 3 มิติ

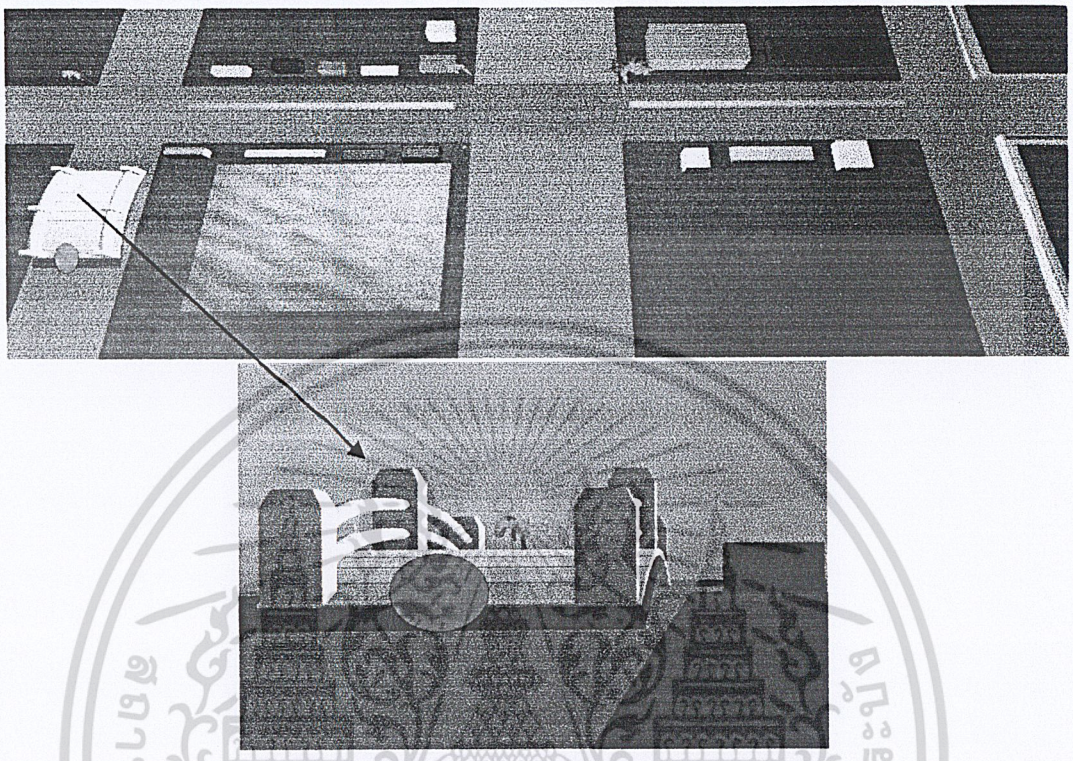
2.2 ชุดคำสั่งในการจัดการด้านกราฟิก 3 มิติ: Direct3D SDK

2.2.1 ระบบพิกัด 3 มิติ

ระบบพิกัด หมายถึงทิศทางของค่าในแนวแกน x, y และ z ของจุดกำเนิด (Origin) ในระบบ 3 มิติ ซึ่งจุดกำเนิดจะมีค่า x, y และ z เป็น (x=0, y=0, z=0) และโดยปกติแล้วทิศทางของค่า x จะเพิ่มขึ้นในทิศทางไปทางด้านขวา และลดลงไปทางด้านซ้ายของจุดกำเนิด ส่วนค่า y จะเพิ่มขึ้นไปทางด้านบน และลดลงไปทางด้านล่าง แต่ค่า z จะมีให้เราเลือกอยู่ 2 แบบ คือแบบแรกจะเพิ่มขึ้นในทิศทางเข้าไปในจอภาพ และแบบที่สองจะเพิ่มขึ้นในทิศทางออกจากจอภาพ โดยแบบแรกเขาเรียกว่า Left-Handed Coordinate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

System ส่วนแบบที่สองจะเรียกว่า Right-Handed Coordinate System ในโปรแกรมของเรานั้น เราใช้แบบ Left-Handed Coordinate System



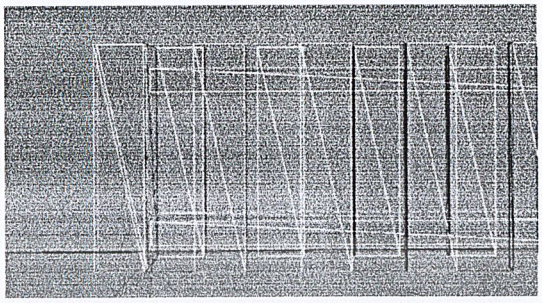
รูปที่ 2-1 ตัวอย่างแสดงตำแหน่งของสะพานในโลก 3 มิติ โดยจุดสีแดงมีพิกัดที่ (-133,8,-93)

2.2.2 Texture

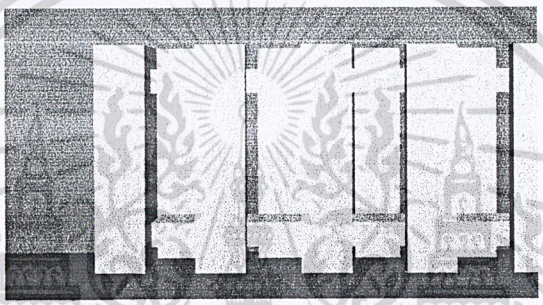
Texture คือ รูปภาพที่จะสามารถสร้าง ความสมจริงให้กับวัตถุ คือเมื่อเราต้องการให้วัตถุที่เราสร้างขึ้นมีความสมจริงเราควรจะสร้างรายละเอียดให้กับพื้นผิวของวัตถุด้วยเช่นกัน เพื่อเพิ่มเติมคุณสมบัติหลักๆ ให้กับวัตถุ เช่น ผนังกำแพงอิฐ เราสามารถที่จะสร้างวัตถุสี่เหลี่ยมและให้สีมันเป็นสีแดงได้ แต่ว่ามันจะดูเหมือนสี่เหลี่ยมสีแดงแทนที่จะเป็นกำแพงอิฐ สิ่งที่เราต้องการที่จะเห็นจริงๆ คืออิฐ เราสามารถที่จะทำแบบนี้ได้โดยการใช้ texture สิ่งที่เราต้องการคือ วัตถุสี่เหลี่ยม และรูปภาพกำแพงอิฐ เราสามารถสร้าง texture ได้ในโปรแกรมวาดรูปทั่วไปแล้วเซฟ เป็น .bmp file เราสามารถที่จะนำ texture ที่ได้ไปแปะติดกับวัตถุที่อยู่ในฉากได้แล้ว ในขั้นตอนนี้เรียกว่า Texture Mapping

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการทำ Texture mapping ดังแสดงในรูปที่ 2-2 โดยรูป 2-2 ก เป็นภาพของรั้วก่อนทำการใส่ texture และรูป 2-2 ข เป็นภาพของรั้วหลังจากใส่ texture ลายไม้ลงไป ทำให้ได้ภาพดูมีความสมจริงมากยิ่งขึ้น



(ก)



(ข)

รูปที่ 2-2 การทำ Texture Mapping

(ก) ก่อนทำ (ข) หลังทำ Texture Mapping

2.3 ทฤษฎีทางด้านฮาร์ดแวร์

2.3.1 ความรู้เบื้องต้นเกี่ยวกับพอร์ตขนาน

การประมวลผลข้อมูลเพื่องานควบคุมนั้น สิ่งแรกที่จะต้องมีคือส่วนของสัญญาณอินพุต ซึ่งอาจจะมาจากตัวตรวจจับต่างๆ ผ่านวงจรภาคหน้าเพื่อเปลี่ยนรูปแบบสัญญาณอินพุตให้เหมาะสมกับการเชื่อมต่อกับคอมพิวเตอร์ เมื่อข้อมูลอินพุตถูกส่งเข้าสู่คอมพิวเตอร์แล้ว คอมพิวเตอร์จะทำการประมวลผลข้อมูลที่ได้มาให้อยู่ในรูปที่เหมาะสมก่อนที่จะส่งออกไปยังภายนอกผ่านอุปกรณ์เอาต์พุต ผ่านจุดเชื่อมต่ออื่นๆ เพื่อควบคุมอุปกรณ์เอาต์พุตต่อไป

การเชื่อมต่อระหว่างคอมพิวเตอร์กับอุปกรณ์ภายนอกทั้งในส่วนของภาคอินพุตและภาคเอาท์พุตนั้น เมื่อเทียบกับการใช้งานการ์ดอินพุตเอาท์พุตที่ต้องติดตั้งอยู่ภายในเครื่องคอมพิวเตอร์แล้วพอร์ตขนานมีข้อได้เปรียบอยู่หลายประการดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในด้านความปลอดภัย การที่ต้องถอดฝาเครื่องคอมพิวเตอร์ออกมาเพื่อเสียบการ์ดเชื่อมต่อลงในสล๊อตของเครื่องคอมพิวเตอร์ อาจจะทำให้เกิดความเสียหายกับส่วนอื่นๆ ของคอมพิวเตอร์ได้ถ้าผู้ใช้งานไม่มีความชำนาญหรือเกิดการต่อวงจรที่ผิดพลาด

ในด้านการเข้ากันได้กับคอมพิวเตอร์ส่วนใหญ่ การเชื่อมต่อโดยใช้การ์ดที่เสียบลงในสล๊อตไม่สามารถใช้กับคอมพิวเตอร์ในปัจจุบันได้ทุกรุ่น ยกตัวอย่างโน้ตบุ๊ก จะไม่มีสล๊อตเสียบแต่จะมีที่เสียบการ์ด PCMCIA แทน ในขณะที่พอร์ตขนานจะทีการติดตั้งอยู่ในคอมพิวเตอร์ทุกเครื่องทั้งนี้เพื่อใช้ในการติดต่อกับเครื่องพิมพ์

ข้อจำกัดด้านพื้นที่ คอมพิวเตอร์บางเครื่องมีการ์ดเชื่อมต่อตัวอื่นๆอยู่แล้ว จนไม่มีสล๊อตเหลือพอสำหรับการเสียบการ์ดเชื่อมต่อเพิ่มเติม

ความสะดวกในการใช้งาน การเชื่อมต่อทางพอร์ตขนานสามารถทำได้ง่ายเพียงต่อสายสำหรับเชื่อมต่อเข้ากับคอนเน็กเตอร์ DB-25 ของพอร์ตขนาน

จำนวนช่องสัญญาณอินพุตเอาต์พุต พอร์ตขนานมีจำนวนพอร์ตอินพุตเอาต์พุตเพียงพอที่จะนำไปใช้งานต่างๆ และยังสามารถขยายให้มีจำนวนพอร์ตเพิ่มขึ้นได้ โดยพอร์ตขนานปกติมีจำนวนขาเอาต์พุต 12 ขา และอินพุต 5 ขา

ความเร็วในการสื่อสารข้อมูลกับพอร์ตขนาน มีความเร็วเท่ากับการติดต่อกับระบบบัสโดยตรง และมีความเร็วมากกว่าการติดต่อผ่านทางพอร์ตอนุกรม

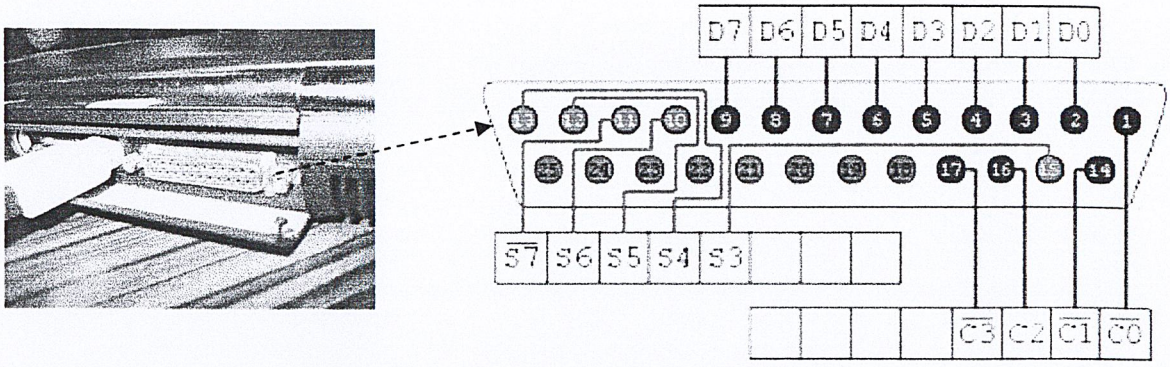
อะไหล่และชิ้นส่วนประกอบ คอนเน็กเตอร์และสายเชื่อมต่อต่างๆ ของการเชื่อมต่อผ่านทางพอร์ตขนาน หาได้ง่ายและราคาไม่แพง หรือจะสร้างขึ้นเองก็สามารถทำได้ง่ายดาย จากคุณสมบัติดังที่กล่าวมาแล้วนั้นทำให้พอร์ตขนานเหมาะอย่างยิ่งที่จะนำมาใช้ในการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกเพื่อควบคุมหรือรับสัญญาณข้อมูล

2.3.1.1 ลักษณะทางกายภาพของพอร์ตขนาน

โดยปกติพอร์ตขนานออกแบบมาให้มีสายสัญญาณอยู่ทั้งหมด 17 เส้น ดังแสดงในรูปที่ 2-3 สายสัญญาณเหล่านั้นจะมีรีจิสเตอร์ 3 ตัวควบคุมการทำงาน ดังนี้

1) พอร์ตเอาต์พุตสำหรับสัญญาณข้อมูล 8 เส้น มีรีจิสเตอร์คาต้าควบคุม
2) พอร์ตอินพุตสำหรับการอ่านค่าสถานะต่างๆจากภายนอกมีอยู่ด้วยกัน 5 เส้น ใช้รีจิสเตอร์สถานะในการควบคุม
3) พอร์ตเอาต์พุตสำหรับส่งสัญญาณควบคุมไปยังอุปกรณ์ภายนอก มีอยู่ด้วยกัน 4 เส้น ใช้รีจิสเตอร์คอนโทรล ในการควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-3 แสดงลักษณะทางกายภาพของพอร์ตขนาน

1) พอร์ตดาต้า (Data Port)

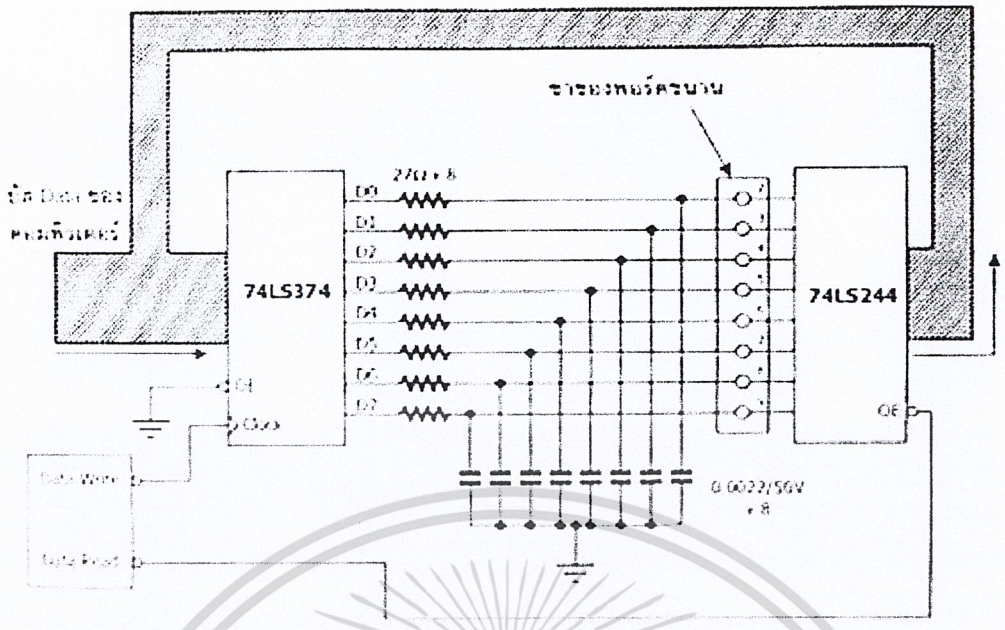
จากรูปที่ 2-4 แสดงให้เห็นว่าพอร์ตดาต้าประกอบไปด้วยบัฟเฟอร์ 1 ตัวและไอซีแลตซ์อีก 1 ตัว เมื่อคอมพิวเตอร์ต้องการส่งข้อมูลไปยังเครื่องพิมพ์ คอมพิวเตอร์จะเขียนข้อมูลไปยังไอซีแลตซ์ 1 ทั้ง 8 บิต เอาต์พุตของไอซีแลตซ์ คือ D0-D7 ซึ่งเอาต์พุตนี้จะไปปรากฏอยู่ที่พอร์ตขนานในตำแหน่งขา 2 ถึงขา 9 และที่ขาเอาต์พุตนี้สัญญาณ Data จะส่งกลับไปเป็นอินพุตของบัฟเฟอร์ 1 ด้วย ทำให้คอมพิวเตอร์สามารถอ่านค่าสถานะปัจจุบันที่เกิดขึ้นกับพอร์ตดาต้าได้

เมื่อคอมพิวเตอร์ส่งข้อมูล ข้อมูลจะถูกส่งมาจากบัสข้อมูลของคอมพิวเตอร์ผ่านไปให้กับไอซี 74LS374 ซึ่งเป็นไอซีแลตซ์ข้อมูล และเมื่อต้องการให้ข้อมูลปรากฏที่เอาต์พุต คอมพิวเตอร์จะส่งสัญญาณ DataWrite ออกไปที่ขา CLK ของ 74LS374 จะถูกกรองด้วยวงจร RC ซึ่งประกอบไปด้วยตัวต้านทานค่า 27 Ω และตัวเก็บประจุ 0.0022μF เพื่อให้ช่วงเวลาที่เปลี่ยนจากลอจิก “0” เป็นลอจิก “1” หรือจากลอจิก “1” เป็นลอจิก “0” เป็นไปอย่างช้าๆ เนื่องจากการเปลี่ยนแรงดันที่รวดเร็วทำให้เกิดสัญญาณรบกวนเหนี่ยวนำข้ามไปยังข้อมูลบิตอื่นๆ ได้ ทำให้ข้อมูลที่ส่งออกไปมีข้อผิดพลาด จากค่าตัวต้านทานและตัวเก็บประจุในวงจรทำให้เกิดการหน่วงเวลาไปประมาณ 60 นาโนวินาที จากวงจรในรูปที่ 2-4 ทำให้เอาต์พุตของพอร์ต Data มีคุณสมบัติดังนี้

- กระแสซิงก์ (Synchronous Current) สูงสุด 24 mA
- กระแสซอร์ส (Source Current) สูงสุด 2.6 mA
- ระดับแรงดันลอจิก “1” เท่ากับ 2.5 V. ~ 5.0 V.
- ระดับแรงดันลอจิก “0” เท่ากับ 0 V. ~ 2.4 V.

สำหรับบัฟเฟอร์สำหรับการอ่านข้อมูลกลับได้แก่เบอร์ 74LS244 ซึ่งเมื่อต้องการอ่านค่าคอมพิวเตอร์จะส่งสัญญาณ DataRead ออกมาเพื่อ enable ไอซี 74LS244 สำหรับพอร์ตขนานแบบมาตรฐาน (Standard Parallel Port: SPP) พอร์ต Data จะต้องใช้เพื่อการส่งค่าออกเอาต์พุตเท่านั้น แต่สำหรับพอร์ตขนานที่มีการสื่อสารสองทิศทาง (Bidirectional Parallel Port) สามรถอ่านค่าจากพอร์ต Data ได้ด้วย แต่ก่อนที่จะอ่านค่าต้องจำไว้เสมอว่าจะต้องป้อนค่าเอาต์พุตให้มีค่าลอจิก “1” ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



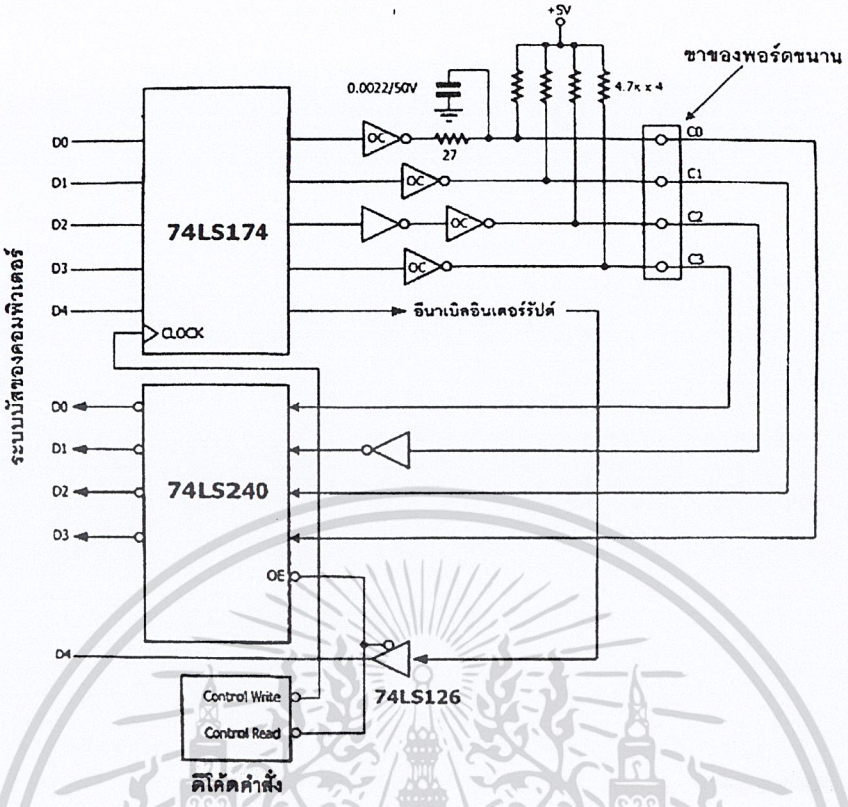
รูปที่ 2-4 วงจรภายในของพอร์ตดาต้า

2) พอร์ตคอนโทรล (Control Port)

พอร์ตคอนโทรลใช้สำหรับคอมพิวเตอร์ควบคุมเครื่องพิมพ์ ซึ่งประกอบไปด้วยบิตเอาต์พุต 4 บิต ที่ต่อไปยังเครื่องพิมพ์ ส่วนบิตเอนเนเบิลอินเตอร์รัพต์ไม่ได้ถูกต่อออกไป รูปที่ 2-5 แสดงบล็อกไดอะแกรมของพอร์ตคอนโทรลเอาต์พุตของพอร์ตคอนโทรลมีอินเวอร์เตอร์แบบคอลเล็กเตอร์เปิดต่อร่วมอยู่ โดยเอาต์พุตเหล่านี้จะถูกพลูอัปไว้ด้วยตัวต้านทานค่า 4.7 kΩ สำหรับบิต C2 จะผ่านอินเวอร์เตอร์ถึงสองตัวทำให้เอาต์พุตของบิต C2 ไม่มีการกลับสถานะลอจิก

สถานะของพอร์ตคอนโทรลสามารถอ่านกลับได้โดยการใช้บัฟเฟอร์เบอร์ 74LS240 ซึ่งเอาต์พุตของ 74LS240 มีอินเวอร์เตอร์อยู่ภายใน ทำให้ค่าที่อ่านได้ตรงกับค่าที่ส่งออกไป การควบคุมการอ่านและการเขียนข้อมูลกับพอร์ตคอนโทรลคอมพิวเตอร์จะส่งข้อมูลมาที่ขา Control Write และ Control Read เนื่องจากเอาต์พุตของพอร์ตคอนโทรลเป็นแบบคอลเล็กเตอร์เปิด ดังนั้นผู้ใช้งานสามารถใช้พอร์ตนี้ในการอ่านค่าสัญญาณอินพุตจากภายนอกได้ โดยก่อนที่จะอ่านค่าจะต้องทำให้ขาพอร์ตที่ต้องการอ่านค่ามีลอจิก “1” เสียก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

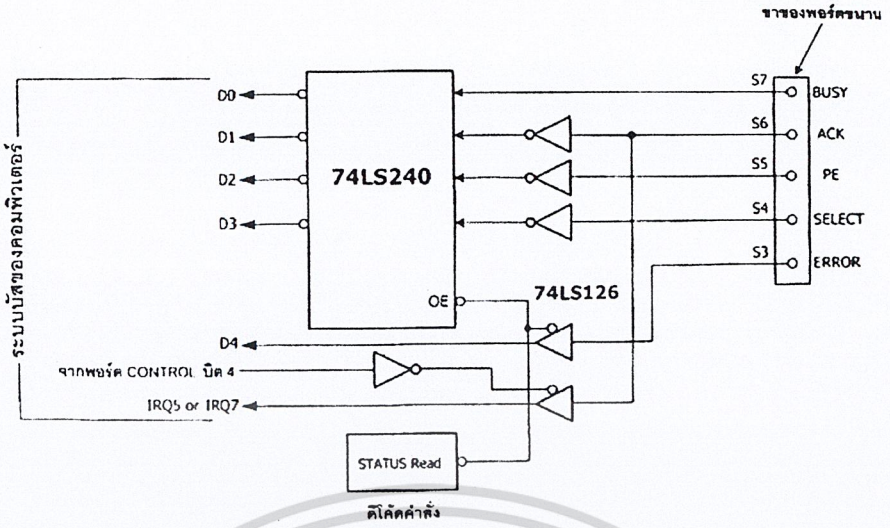


รูปที่ 2-5 วงจรภายในของพอร์ตคอนโทรล

3) พอร์ตแสดงสถานะ (Status Port)

พอร์ตแสดงสถานะ เป็นพอร์ตที่คอมพิวเตอร์ใช้สำหรับการอ่านค่าสถานะจากเครื่องพิมพ์ รูปที่ 2-6 แสดงรายละเอียดภายในของพอร์ตแสดงสถานะ จะสังเกตได้ว่ามีขาสัญญาณอยู่ทั้งหมด 5 สัญญาณด้วยกันและจะเรียกชื่อเป็น S3, S4, S5, S6 และ S7 ซึ่งตัวเลขนั้นหมายถึงตำแหน่งบิตของขาเหล่านี้ภายในตัวรีจิสเตอร์แสดงสถานะ นั่นเอง สำหรับบิต S7 จะมีข้อแตกต่างจากบิตอื่นๆ ที่เมื่อสัญญาณจากภายนอกส่งเข้ามาแล้วจะไม่ผ่านอินเวอร์เตอร์ ในขณะที่ขาอื่นๆ ผ่านอินเวอร์เตอร์ทั้งหมด ดังนั้นเมื่อข้อมูลผ่านจากขาอินพุตไปยัง 74LS240 ซึ่งเอาต์พุตมีการกลับสถานะทำให้บิต S7 เป็นบิตเดียวที่มีการกลับสถานะ นอกจากนี้ในการใช้งานถ้าต้องการให้มีการสร้างสัญญาณอินเตอร์รัพต์จากขอบขาขึ้นของขา S6 สามารถกำหนดค่าได้จากพอร์ตคอนโทรลบิตที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-6 วงจรภายในของพอร์ตแสดงสถานะ

2.3.1.2 การนำพอร์ตขนานไปใช้งาน

สำหรับพอร์ตขนานแบบมาตรฐาน ผู้ใช้งานสามารถนำพอร์ตอินพุต 5 บิต (พอร์ตแสดงสถานะ) พอร์ตเอาต์พุต 4 บิต (พอร์ตคอนโทรล) และพอร์ตเอาต์พุตอีก 8 บิต (พอร์ตคาต้า) ไปใช้งานได้โดยตรง โดยที่ 4 บิตของพอร์ตเอาต์พุต หรือพอร์ตคอนโทรลนั้นสามารถดัดแปลงให้ใช้งานเป็นพอร์ตอินพุตขนาด 4 บิตได้ด้วย ดังนั้นผู้ใช้งานจึงสามารถนำสัญญาณจากพอร์ตขนานที่มีมากถึง 17 เส้นไปใช้งานในการควบคุมโดยใช้ระดับสัญญาณ TTL ซึ่งแต่ละพอร์ตจะมีแอดเดรสสำหรับเรียกใช้งานต่างกันโดยสามารถดูได้จากตารางที่ 2-1 และเมื่อจะนำบิตต่างไปใช้ จะต้องทราบตำแหน่งของบิตนั้นว่าอยู่ที่ตำแหน่งใดของคอนเน็กเตอร์ DB-25 ซึ่งสามารถดูได้จากตารางที่ 2-2

ชื่อพอร์ต	LPT1:		LPT2:		LPT3:	
	ฐานสิบ	ฐานสิบหก	ฐานสิบ	ฐานสิบหก	ฐานสิบ	ฐานสิบหก
DATA	888	378H	956	3BCH	632	278H
STATUS	889	379H	957	3BDH	633	279H
CONTROL	890	37AH	958	3BEH	634	27AH

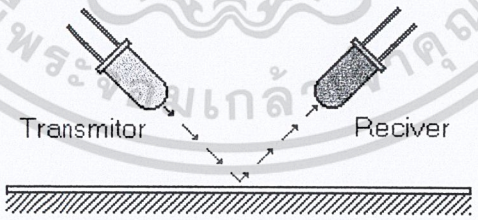
ตารางที่ 2-1 แสดงแอดเดรสของพอร์ตขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Signal Name	Register bit	Signal Pin			Ground Return Pin		
		D-sub (IEEE 1284-A)	Centron-ics (IEEE 1284-B)	IEEE 1284-C	D-sub (IEEE 1284-A)	Centron-ics (IEEE 1284-B)	IEEE 1284-C
Data bit 0	*D0*	2	2	6	19	20	24
Data bit 1	*D1*	3	3	7	19	21	25
Data bit 2	*D2*	4	4	8	20	22	26
Data bit 3	*D3*	5	5	9	20	23	27
Data bit 4	*D4*	6	6	10	21	24	28
Data bit 5	*D5*	7	7	11	21	25	29
Data bit 6	*D6*	8	8	12	22	26	30
Data bit 7	*D7*	9	9	13	22	27	31
nError (nFault)	*S3*	15	32	4	23	29	22
Select	*S4*	13	13	2	24	28	20
PaperEnd	*S5*	12	12	5	24	28	23
nAck	*S6*	10	10	3	24	28	21
Busy	$\overline{S7}$	11	11	1	23	29	19
nStrobe	$\overline{C0}$	1	1	15	18	19	33
nAutoLF	$\overline{C1}$	14	14	17	25	30	35
nInit	*C2*	16	31	14	25	30	32
nSelectIn	$\overline{C3}$	17	36	16	25	30	34
HostLogicHigh				18			18
PeriphLogicHigh				36			36

ตารางที่ 2-2 แสดงความสัมพันธ์ระหว่างตำแหน่งขากับ Register bit

2.3.2 หลักการทำงานของเซ็นเซอร์แสง

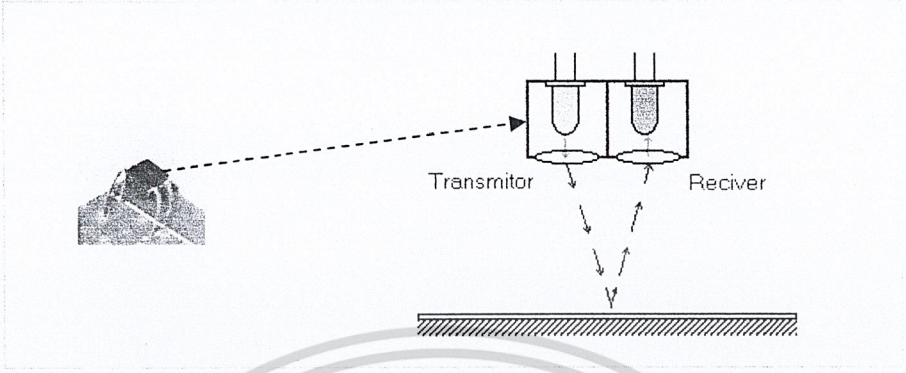


รูปที่ 2-7 การทำงานของเซ็นเซอร์แสง

แสงอินฟราเรด คือแสงที่มีความยาวคลื่นต่ำกว่าแสงสีแดงลงไป ดังนั้นจึงไม่สามารถมองเห็นได้ด้วยสายตา ของมนุษย์ ซึ่งคุณสมบัตินี้เอง จึงทำให้ เซ็นเซอร์ชนิดที่ใช้แสงอินฟราเรด เป็นที่นิยมนำมาใช้กันมาก โดยจะอาศัยหลักการของการสะท้อนของแสง กล่าวคือ ใช้อุปกรณ์ส่งแสง เป็นแหล่งกำเนิด ปล่อยแสงออกไป และเมื่อแสงกระทบกับวัตถุด้านหน้า มันก็จะสะท้อนแสงกลับมา เข้าที่ตัวรับแสง ส่วนอัตราของการสะท้อนกลับนั้น ขึ้นอยู่กับสี และสภาพความมัน ของวัตถุที่สะท้อน เช่น สีดำ จะมีอัตราการ

เอกสารนี้มีการสงวนลิขสิทธิ์ ห้ามนำไปใช้ในการแสวงหาผลประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สะท้อนกลับ น้อยกว่าสีขาว หรือสภาพพื้นผิวที่มีความราบเรียบ เป็นมันวาว จะสามารถสะท้อนแสงได้ดีกว่า พื้นผิวที่มีลักษณะด้าน และขรุขระ เป็นต้น



รูปที่ 2-8 ลักษณะทางกายภาพและการทำงานของเซ็นเซอร์แสง

เซ็นเซอร์ตัวนี้ใช้หลักการของการสะท้อนกลับของแสงอินฟราเรด แต่ตัวเซ็นเซอร์ที่จะนำมาใช้นั้นจะรวมเอาตัวรับแสง และตัวส่งแสง รวมอยู่ในตัวถังเดียวกัน ซึ่งเป็นการสะดวกในการนำมาใช้งาน เพราะเราไม่ต้องมาจัดระยะห่างเอง ไม่ต้องมาจัดมุมสะท้อนเอง เพราะทุกอย่างนั้น ได้ถูกจัดการมาจากโรงงานผลิตเรียบร้อยแล้ว

2.3.3 หลักการทำงานของตัวต้านทานปรับค่าได้ (Potentiometer)

รูปที่ 2-9 ลักษณะทางกายภาพและโครงสร้างภายในของตัวต้านทานปรับค่าได้

โดยพื้นฐานแล้ว ตัวต้านทานปรับค่าได้ประกอบไปด้วย แถบนำไฟฟ้าแบบโค้ง (ส่วนที่เป็นสีดำจากรูปทางขวามือของ รูปที่ 2-9) ที่มีส่วนปลายต่ออยู่คือ A และ C และมีแถบหมุน (wiper) ที่เคลื่อนที่ได้ติดอยู่กับส่วนปลายของ B

ค่าความต้านทานของแถบนำไฟฟ้าทั้งแถบ จะบอกค่าความต้านทานของตัวต้านทานปรับค่าได้ โดยเมื่อหมุนแกน (ส่วนสีน้ำเงิน) ของตัวต้านทานปรับค่าได้ จะทำให้แถบหมุน (ส่วนสีเหลือง) เคลื่อนที่ตามไปด้วย ความยาวของเส้นทางตามแถบที่ได้ระหว่างแถบหมุนและส่วนปลาย ก็คือความต้านทานที่เปลี่ยนค่าไป (ตามเส้นสีแดง) ถ้าหมุนแกนทวนเข็มนาฬิกา ค่าความต้านทานระหว่าง A และ B จะเพิ่มขึ้น เช่นเดียวกันกับค่าระหว่าง B และ C ก็จะลดลง เมื่อหมุนแกนตามเข็มนาฬิกาจนสุดค่าความต้านทาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระหว่าง A และ B จะมีค่าเป็น 0 และเมื่อหมุนทวนเข็มจนสุด ค่าที่ได้จาก A และ B จะได้เท่ากับค่าความต้านทานของตัวต้านทานปรับค่าได้

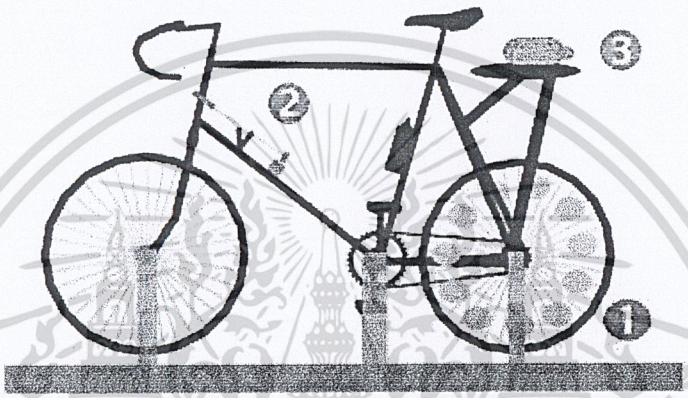
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

บทที่ 3
การพัฒนาระบบ

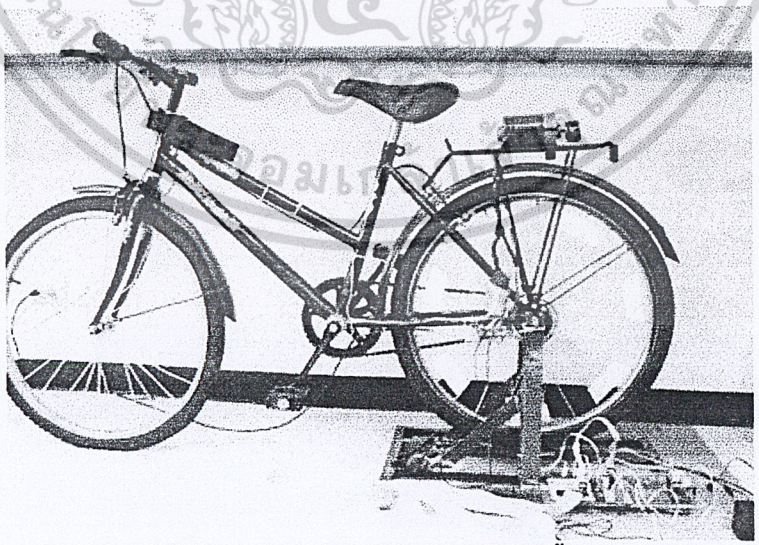
3.1 การพัฒนาส่วนฮาร์ดแวร์

จุดมุ่งหมายหลักในการออกแบบวงจร คือการออกแบบให้วงจรเหล่านี้สามารถส่งค่ากลับมายังคอมพิวเตอร์ และตอบสนองผู้ใช้ได้อย่างถูกต้อง ซึ่งก็จะต้องมีทั้งส่วนอินพุตและเอาต์พุต



รูปที่ 3-1 รูปการติดตั้งอุปกรณ์ที่ได้ออกแบบไว้

จากรูป 3-1 ณ ตำแหน่งหมายเลข 1 และ 2 เป็นส่วนตรวจสอบการบังคับรถจักรยาน และหมายเลข 3 เป็นส่วนสร้างปฏิกิริยาตอบสนอง



รูปที่ 3-2 แสดงรถจักรยานและอุปกรณ์ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

49977

3.1.1 อุปกรณ์ที่ใช้ในการพัฒนาทั้งหมด

1) รถจักรยาน

2) ฐานตั้งรถจักรยาน

3) ตัวต้านทานปรับค่าได้ (Potentiometer)

4) เซ็นเซอร์แสง (Optical Sensor)

5) มอเตอร์

6) สายยางสำหรับพันรอบแฮนด์จักรยาน

7) A/D (Analog to Digital Converter)

8) รีเลย์ (Relay)

9) บัฟเฟอร์

10) ตัวรักษาแรงดัน (Regulator)

11) ตัวต้านทาน

12) ตัวเก็บประจุ

13) สายไฟ

14) ปลั๊กไฟ

15) หม้อแปลงไฟ 220V – 12V

16) สายแพร

17) หัว DB-25

18) หัวต่อ (Connector)

19) บอร์ดไข่ปลา

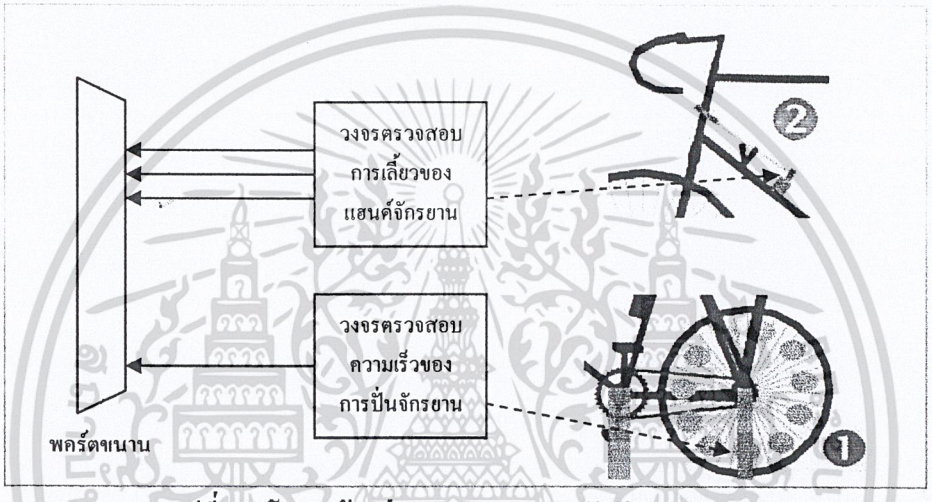
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบวงจรทั้งหมดสามารถแบ่งออกเป็นส่วนหลักๆ ได้ 3 ส่วนด้วยกันคือ

- ส่วนตรวจสอบการบังคับรถจักรยาน
- ส่วนสร้างปฏิกิริยาตอบสนอง
- ส่วนติดต่อระหว่างคอมพิวเตอร์กับฮาร์ดแวร์

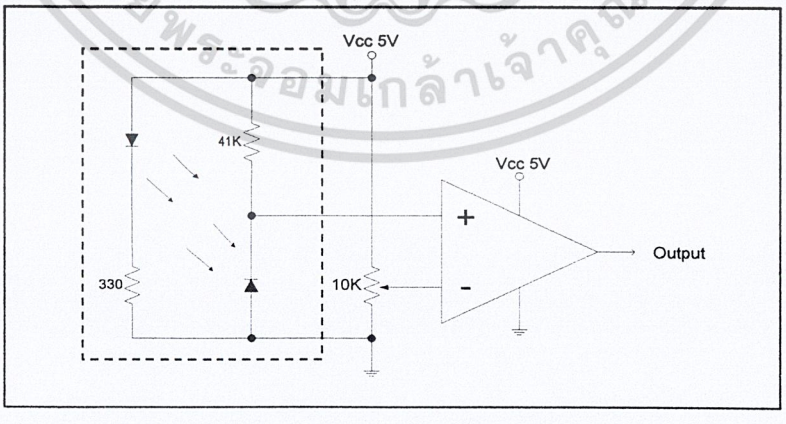
3.1.2 ส่วนตรวจสอบการบังคับรถจักรยาน

ส่วนนี้จะมีการตรวจสอบด้วยกัน 2 จุดคือ ตรวจสอบความเร็วของการปั่นรถจักรยาน และตรวจสอบการเลี้ยวของแฮนด์รถจักรยาน



รูปที่ 3-3 โครงสร้างส่วนตรวจสอบการบังคับรถจักรยาน

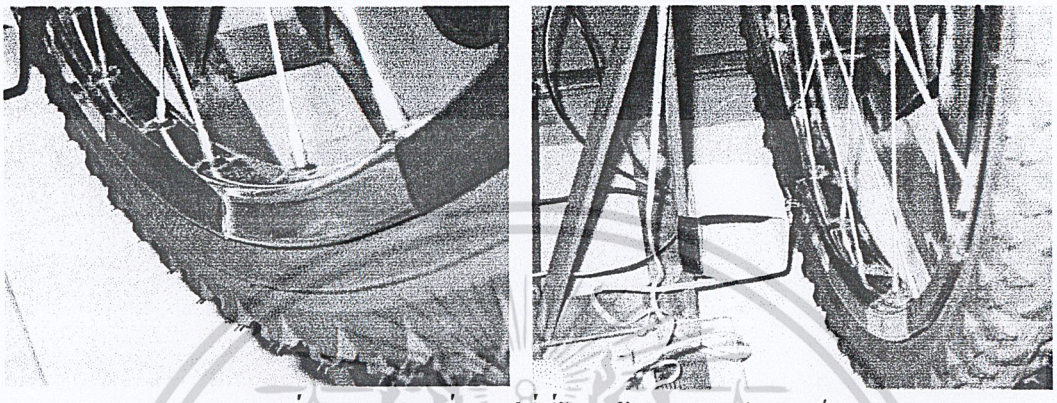
1) การตรวจสอบความเร็วของการปั่นรถจักรยาน จะใช้เซ็นเซอร์แสงเป็นตัวตรวจสอบ โดยได้ออกแบบวงจรเพื่อที่จะใช้ต่อชุดตรวจสอบความเร็วไว้ดังรูปที่ 3-4



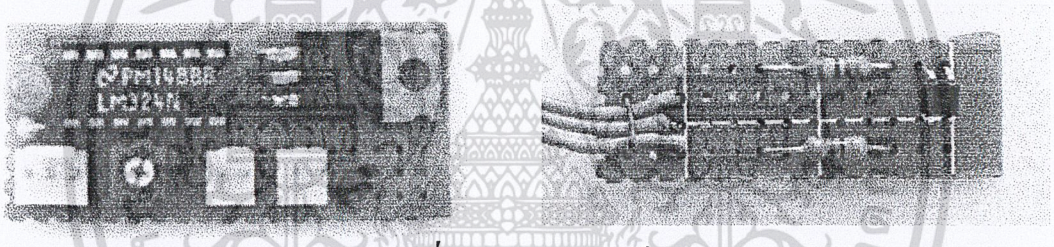
รูปที่ 3-4 วงจรตรวจสอบความเร็วของการปั่นรถจักรยาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการติดวัสดุไม่สะท้อนแสงที่มีสีดำบนยางล้อของรถจักรยาน เพื่อที่เซ็นเซอร์แสงจะได้สามารถตรวจสอบการหมุนของล้อได้ดังรูปที่ 3-5 และ 3-6 โดยติดวัสดุดังกล่าวไว้ทั้งหมด 8 ชิ้นรอบล้อของรถจักรยาน เมื่อล้อหมุน เซ็นเซอร์ที่ตรวจจับอยู่จะส่งค่ากลับมาเป็นความต่างศักย์ไฟฟ้าระหว่าง 0V กับ 5V สลับกันไปเรื่อยๆ โดยความถี่ของค่าความต่างศักย์ไฟฟ้าจะขึ้นกับความเร็วที่เราปั่นรถจักรยาน



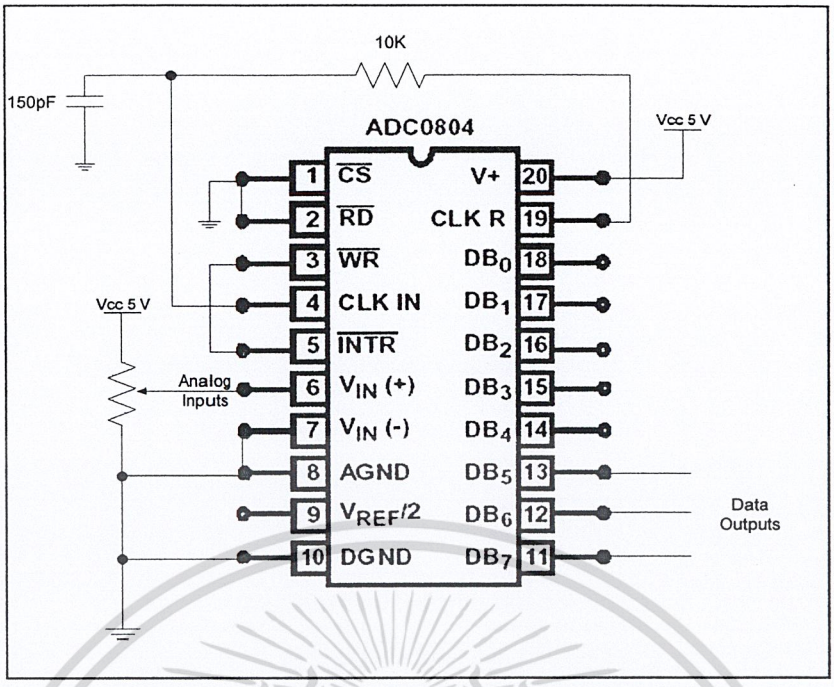
รูปที่ 3-5 กระดาษที่ติดอยู่ที่ซี่โครงล้อ และตัวเซ็นเซอร์



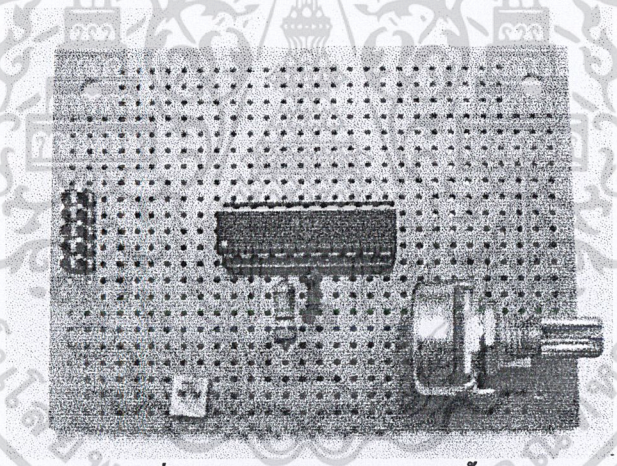
รูปที่ 3-6 วงจรเซ็นเซอร์

2) การตรวจสอบการเลี้ยวของแฮนด์รถจักรยาน เราจะใช้ตัวต้านทานปรับค่าได้เพื่อตรวจสอบระยะการเลี้ยวว่าเลี้ยวไปด้านใด และมุมการเลี้ยวขนาดเท่าไร โดยทำการออกแบบและต่อวงจรดังรูปที่ 3-7 และรูปที่ 3-8 ตามลำดับ ตัวต้านทานปรับค่าได้นั้นจะทำให้ค่าความต่างศักย์เปลี่ยนแปลงแล้วส่งไปให้วงจร A/D แปลงค่าความต้านทานเป็นสัญญาณดิจิตอล 8 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



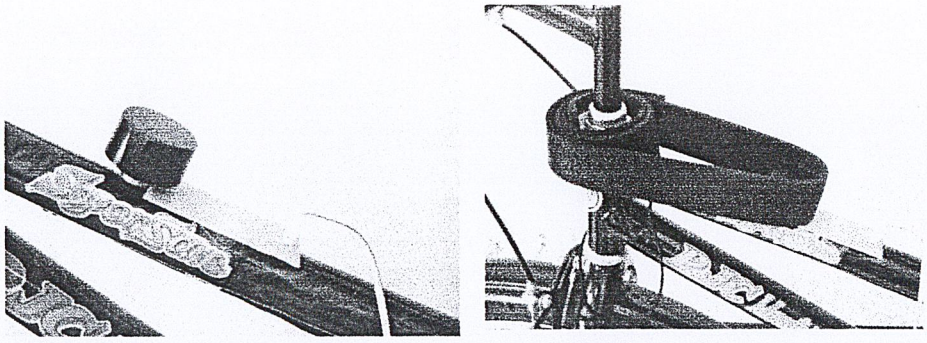
รูปที่ 3-7 วงจรตรวจสอบการเลี้ยวของแฮนด์รถจักรยาน



รูปที่ 3-8 อุปกรณ์ตรวจสอบการเลี้ยว

จากนั้นจึงนำตัวต้านทานปรับค่าได้มาคล้องด้วยสายพานซึ่งคล้องอยู่กับคอรถจักรยานดังรูปที่ 3-9 เพื่อให้เมื่อเลี้ยวซ้ายหรือเลี้ยวขวาแล้ว ตัวต้านทานปรับค่าได้ก็จะหมุนตามแล้วความต้านทานก็จะเปลี่ยน ซึ่งก็จะทำให้ค่าความต่างศักย์ที่ส่งไปให้วงจร A/D เปลี่ยนไปด้วย

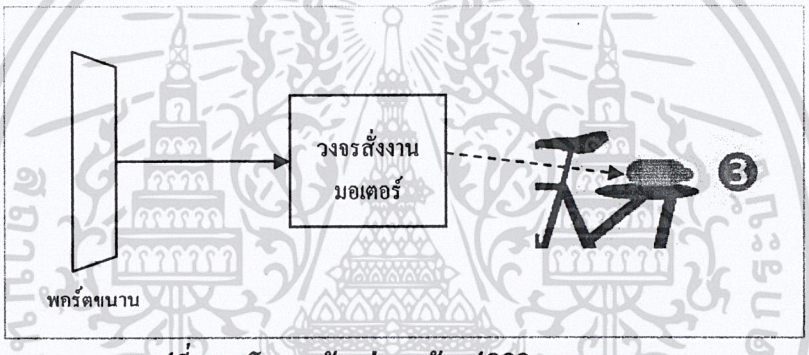
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-9 ตัวต้านทานปรับค่าได้ที่ติดบนโครงรถจักรยาน และมีสายยางพันอยู่กับคอรถจักรยาน

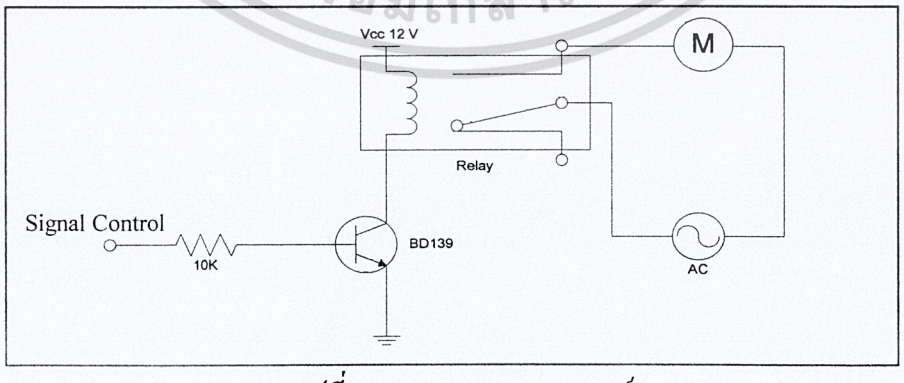
3.1.3 ส่วนสร้างปฏิกิริยาตอบสนอง

ส่วนนี้จะเป็นส่วนที่ตอบสนองต่อผู้ที่ปั่นรถจักรยานด้วยการหมุนของมอเตอร์ที่ติดแป้นลูกเบี้ยว มอเตอร์ดังกล่าวติดตั้งอยู่บนรถจักรยาน โดยจะได้รับคำสั่งให้หมุนที่สั่งมาจากคอมพิวเตอร์ผ่านมาทางพอร์ตขนาน



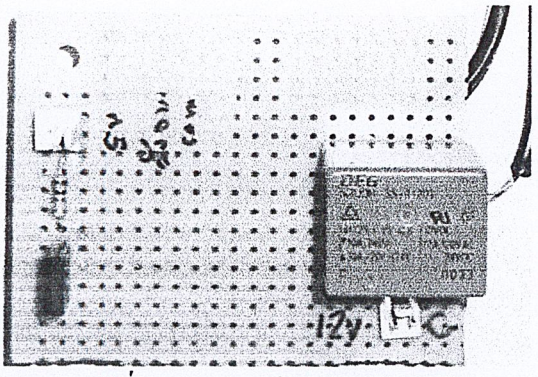
รูปที่ 3-10 โครงสร้างส่วนสร้างปฏิกิริยาตอบสนอง

เมื่อเราสั่งให้มอเตอร์หมุน แป้นลูกเลี้ยวจะก่อให้เกิดการสั่นสะเทือน ซึ่งจะทำให้รถจักรยานสั่นไปได้ วงจรควบคุมมอเตอร์ถูกออกแบบไว้ดังรูปที่ 3-11 โดยมอเตอร์มีรีเลย์เป็นตัวควบคุมการจ่ายกระแสและภาพอุปกรณ์ทั้งหมดแสดงไว้ดังรูปที่ 3-12

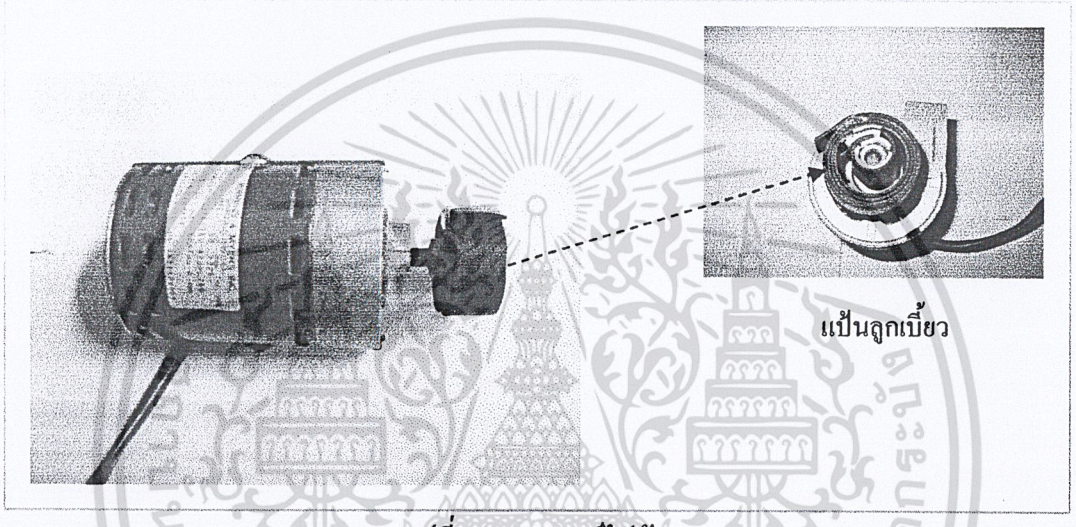


รูปที่ 3-11 วงจรควบคุมมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-12 อุปกรณ์ควบคุมมอเตอร์



รูปที่ 3-13 มอเตอร์ไฟฟ้า

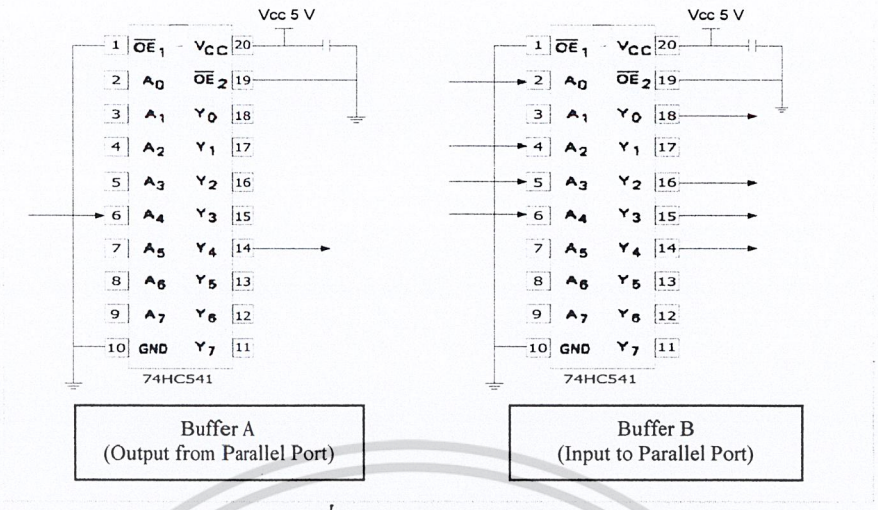
เมื่อเราทำการส่งสัญญาณจากคอมพิวเตอร์ที่มีสถานะเป็น 1 มาจะทำให้มอเตอร์หมุนซึ่งก็จะทำให้รถจักรยานสั่นไปด้วย และเมื่อต้องการที่จะหยุดสั่นก็ส่งค่าสถานะ 0 มาเพื่อสั่งให้มอเตอร์หยุดการทำงาน

การที่เมื่อมอเตอร์หมุนแล้วทำให้รถจักรยานสั่นนั้น เกิดขึ้นจากการติดตั้งแป้นลูกเบี้ยวไว้ที่แกนของมอเตอร์ และเมื่อมอเตอร์หมุนก็จะทำให้แป้นลูกเบี้ยวหมุน ซึ่งจะเหวี่ยงไปมาทำให้เกิดการสั่นได้

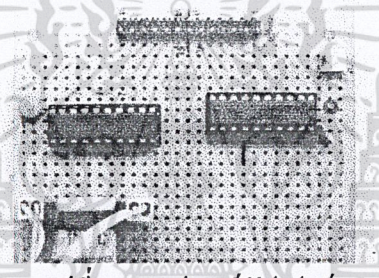
3.1.4 ส่วนติดต่อระหว่างคอมพิวเตอร์กับฮาร์ดแวร์

ส่วนนี้จะกล่าวถึงการติดต่อระหว่างอุปกรณ์ต่างกับพอร์ตขนาน ว่าติดต่อกันผ่านพอร์ตใด และบิตใดบ้าง แต่เนื่องจากการที่จะติดต่อกับพอร์ตขนานโดยตรงนั้นเป็นการเสี่ยงที่อาจจะเกิดอันตรายต่อพอร์ตขนานที่เกิดจากความบกพร่องของวงจรได้ จึงได้เพิ่มส่วนของไอซีบัฟเฟอร์เข้าไปแทรกระหว่างอุปกรณ์ต่างๆกับพอร์ตขนาน ซึ่งได้ออกแบบวงจรไว้ดังรูปที่ 3-14 และเมื่อต่ออุปกรณ์เสร็จจะเป็นดังรูปที่ 3-15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-14 วงจรบัฟเฟอร์



รูปที่ 3-15 อุปกรณ์บัฟเฟอร์

จากรูป 3-14 ทางด้านซ้ายของวงจรนั้นจะเป็นส่วนที่ทำการส่งค่าไปควบคุมมอเตอร์ให้ทำงานจากคอมพิวเตอร์ผ่านทางพอร์ตขนาน ซึ่งก็คือส่งไปยังขาที่ 6 และส่งต่อไปจากขาที่ 14 และทางด้านขวานั้นจะเป็นส่วนรับค่ามาจากอุปกรณ์ต่างๆ เพื่อส่งต่อไปยังพอร์ตขนานแล้วส่งเข้าสู่คอมพิวเตอร์ โดยขาที่ 2 จะรับค่ามาจากส่วนตรวจสอบการปั่นรถจักรยาน และขาที่ 4-6 จะรับค่ามาจากส่วนตรวจสอบการเลี้ยวซ้ายเลี้ยวขวา

โดยการติดต่อกับพอร์ตขนานนั้นในพอร์ตที่ใช้รับข้อมูลมาจากส่วนตรวจสอบการบังคับรถจักรยาน นั้นได้เลือกใช้พอร์ตแสดงสถานะ (379h) เนื่องจากเป็นพอร์ตที่ใช้สำหรับรับข้อมูลอยู่แล้ว โดยไม่ต้องมีการส่งค่าอะไรไปเพื่อรับข้อมูลเลย

และสำหรับส่วนที่ใช้ควบคุมการทำงานของมอเตอร์นั้นได้เลือกใช้พอร์ตดาต้า (378h) เนื่องจากพอร์ตดาต้าเป็นพอร์ตที่มีไว้สำหรับส่งข้อมูลออกไปอยู่แล้ว จึงทำให้สามารถส่งค่าที่ออกไปได้เลยโดยไม่ต้องตั้งค่าหรือต้องเปลี่ยนแปลงค่าใหม่ก่อนส่ง

โดยสามารถระบุบิตที่ใช้ในแต่ละพอร์ตสำหรับการติดต่อระหว่างฮาร์ดแวร์และคอมพิวเตอร์ผ่านทางพอร์ตขนานได้ดังตารางที่ 3-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Signal Name	Register bit	Signal Pin	Buffer Pin	ลักษณะการใช้งาน
Data bit 4	D4	6	6A	ส่งสัญญาณไปควบคุมมอเตอร์
Select	S4	13	18B	รับสัญญาณมาจากส่วนตรวจสอบการปั่นรถจักรยาน
PaperEnd	S5	12	16B	S5, S6 และS7 รับค่ามาจากส่วนตรวจสอบการเลี้ยว
nAck	S6	11	14B	
Busy	S7	10	15B	

(D = Data port, S = Status port, A = BufferA (output from parallel port), B = BufferB (input to parallel port))

ตารางที่ 3-1 แสดงการเชื่อมระหว่างบิตของพอร์ตขนาน กับส่วนต่างๆของอุปกรณ์ภายนอก

3.2 การพัฒนาด้านซอฟท์แวร์

3.2.1 วิธีการโหลด .x file เข้ามาในโปรแกรม

หลังจากที่ได้ออกแบบและสร้างโลกและวัตถุ 3 มิติจากโปรแกรม 3D Studio Max (3DsMax) แล้วนั้น เราจำเป็นต้อง export ออกเป็น .x file ซึ่งสำหรับ 3DsMax นั้นเราต้องใช้โปรแกรมเสริม (Plug-in) ในที่นี้เราใช้ตัวที่มาพร้อมกับ DirectX SDK Extra ขั้นตอนต่อไปคือการโหลดโมเดลเข้ามาในโปรแกรม DirectX ของเรา

ในโปรแกรมเราได้สร้างคลาส CMesh ขึ้นมาสำหรับการโหลดและเรนเดอร์ mesh จาก .x files โดยที่ตัว CMesh Constructor จะเอาไว้โหลด mesh จาก .x file และเก็บไว้ในหน่วยความจำ ตัว Destructor เอาไว้คืนหน่วยความจำที่เราเอาไว้เก็บ mesh และ เมธอด render แสดงถึงวิธีการ เรนเดอร์ ตัว mesh ที่ได้โหลดเอาไว้

CMesh Constructor รับค่า 2 ตัวแปร ตัวแรกเป็นพอยน์เตอร์ชี้ไปที่ ตัว device และตัวที่ 2 เป็นสตริงที่เอาไว้เก็บพาทของ .x file ที่จะเอาไว้โหลด เราใช้ฟังก์ชัน D3DXLoadMeshFromX ในการโหลด Mesh เข้าไว้ในหน่วยความจำ จากนั้นเราได้สร้าง array ขึ้นมา 2 ตัว ตัวแรกเอาไว้เก็บ material อีกตัวเอาไว้เก็บ texture ของโมเดล

ตัวอย่างของการโหลดโมเดล .x file

```
m_pMeshWorld = new CMesh(m_pD3DDevice, "myworld2.x")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การรับข้อมูลจากตัวรถจักรยาน

เนื่องจากเราได้พัฒนาโปรแกรมบนระบบปฏิบัติการ Windows2000 และ Windows XP จึงต้องมีการใช้ไดรเวอร์เพื่อการมีสิทธิ์เรียกใช้พอร์ตขนานได้อย่างเต็มที่ ในที่นี้เราใช้โปรแกรม PrintIO เป็นไดรเวอร์เพื่อการติดต่อ และหลังจากที่ได้เริ่ม ไดรเวอร์ แล้วโปรแกรมของเราก็พร้อมที่จะรับส่งค่าข้อมูลผ่านทางพอร์ตขนาน

สำหรับการอ่านค่าข้อมูล มีด้วยกัน 2 ข้อมูลคือ ข้อมูลการเลี้ยวซ้ายขวา และ ข้อมูลการปั่นรถจักรยาน

เนื่องจากค่าที่อ่านได้จากการใช้คำสั่ง _inp(0x379) จะได้บิตที่ไม่ต้องการมาด้วย คือบิต S2,S1,S0 ดังนั้นในการอ่านค่าแต่ละครั้งจะทำให้ได้ค่าที่ไม่ถูกต้องตามความเป็นจริง เราจึงทำโอเปอร์เรชัน and ด้วยค่า 248 ซึ่งจะได้โค้ดดังนี้

```
temp = (_inp(0x379) & 248 );
```

แต่เนื่องจากว่าบิต S7 นั้นถูกกลับสถานะอยู่เราจึงต้องทำการ XOR กับ 128 อีกครั้งซึ่งจะได้โค้ดดังนี้

```
temp = (_inp(0x379) & 248 ^ 128);
```

แต่เพื่อให้สะดวกเราจึงแยกค่าจากข้อมูลการเลี้ยวซ้ายขวา (S7,S6,S5) เก็บใส่ไว้ในตัวแปร RVary

```
Rvary = (_inp(0x379) & 248 ^ 128) & 224;
```

และค่าจากข้อมูลการปั่นรถจักรยาน (S4) เก็บไว้ในตัวแปร Sensor

```
Sensor = (_inp(0x379) & 248 ^ 128) & 16;
```

3.2.3 การคำนวณทิศทางการเคลื่อนที่

ลักษณะของมุมมองบุคคลที่ 1 หรือมุมมองของผู้เล่นในเกมแนว First-Person Shooting นั้นจะให้ภาพเหมือนกับการมองเห็นจริงของผู้เล่น ซึ่งในโปรแกรมของเรา ได้ใช้มุมมองนี้เป็นมุมมองในการเลี้ยวซ้ายขวา และใช้มุมมองของกล้องอิสระในการเคลื่อนที่ไปมาภายในโลก 3 มิติ ลักษณะของมุมกล้องแบบอิสระ ก็คือ ไม่ว่าเราจะมองไปทางไหน หรือองศาไหน ถ้าเราเคลื่อนที่ไปข้างหน้า มันจะต้องเคลื่อนที่ตรงเข้าหาจุดที่เรามองได้อย่างแม่นยำ หรือถ้าเราถอยหลัง ก็ต้องยังมีสิ่งที่เรามองอยู่ข้างหน้าเสมอ แต่เราจะค่อยๆ ถอยห่างออกจากจุดนั้น ดังนั้น ถ้าเราต้องการขึ้นลงสะพาน เราจะต้องคำนึงถึงแกน y เพิ่มเข้ามาด้วย

1) การเลี้ยวซ้ายขวา

สำหรับการเลี้ยวซ้ายขวานั้น ก็คือการหมุนกล้องซ้ายขวา ถ้าเป็นคำในภาษาอังกฤษจะเรียกการหมุนแบบนี้ว่า Yaw การหมุนกล้องซ้ายขวานี้ ที่จริงแล้วก็คือการหมุนรอบแกน y นั่นเอง โดยตอนเริ่มต้นเราต้องเก็บค่ามุมรอบแกน y ไว้ก่อนเป็นตัวตั้งต้นจะใช้ตัวแปรที่ชื่อว่า CenterAngle จากนั้นเมื่อมีการเลี้ยวซ้ายขวาก็ให้มีการบวกหรือลดค่าจาก CenterAngle โดยการบวกหรือลดจะขึ้นอยู่กับค่า Rvary แล้วเก็บเป็นมุมรอบแกน y เอาไว้จะใช้ตัวแปรที่ชื่อว่า g_fAngleY ปกติแล้ว ค่าของมุมรอบแกน y จะหมุนทวนเข็มนาฬิกา โดยมุม 0.0 เรเดียนหรือ 0.0 องศา จะหันไปทางแกน +z แล้วมุมจะเพิ่มขึ้นไปทางแกน -x, -z, +x

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

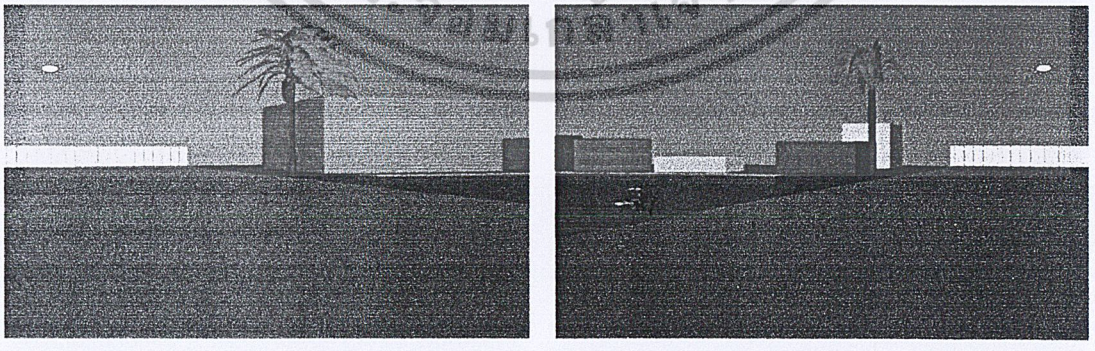
แล้วกลับมาครบรอบที่ +z ตามลำดับ ดังนั้นถ้ามีการบังคับแฮนด์รถจักรยานไปทางซ้าย ก็จะเป็นการเพิ่มค่ามุม ถ้าบังคับไปทางขวาก็จะเป็นการลดค่ามุม เช่น

```
if(Rvary == 32)
{
        g_fAngleY = CenterAngle + 2*ROTATION_SPEED;
}
```

จากนั้นในขั้นตอนการสร้าง View Matrix เราจะใช้ฟังก์ชั่น D3DXMatrixRotationY() เพื่อสร้างเมท-ริกซ์ ผลลัพธ์ที่ได้จากค่าของมุมรอบแกน y

```
D3DXMatrixRotationY(&matRotationY, D3DXToRadian(g_fAngleY));
```

รูปที่ 3-16 ตัวอย่างรูปมุมมองบุคคลที่ 1



รูปที่ 3-17 แสดงถึงมุมมองเมื่อมีการเลี้ยวขวา และซ้าย

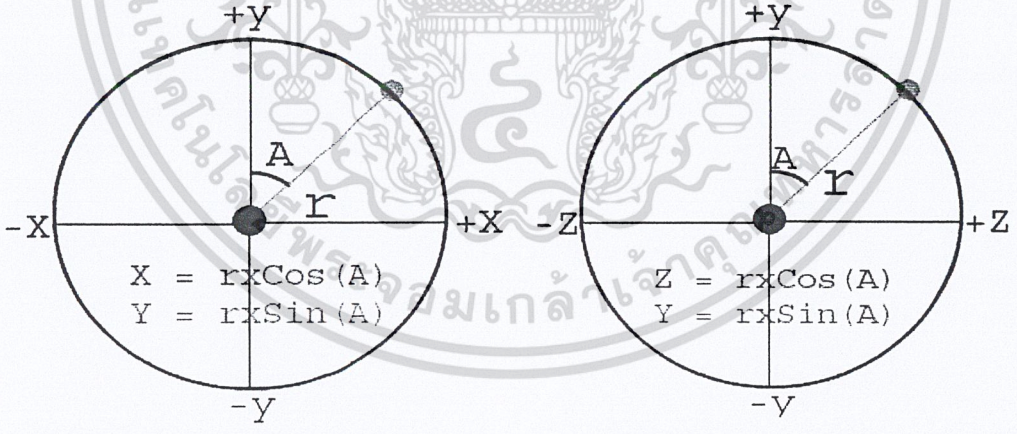
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การเดินหน้าและถอยหลัง

สำหรับการเคลื่อนที่ไปมาภายในโลก 3 มิติ ก็คือการเคลื่อนที่ในแนวระนาบของกล้อง ซึ่งก็คือการเคลื่อนที่อยู่ในแกน x และ z ของ Worldspace (เพราะแกนทั้งสองนี้มันตั้งฉากกันอยู่ในแนวระนาบ) โดยแกน +z จะพุ่งตรงไปข้างหน้าเข้าหาจอ, แกน -z พุ่งไปทางด้านหลัง, แกน -x อยู่ทางซ้าย และแกน +x อยู่ทางด้านขวา เราอาจจะคิดว่า การเคลื่อนที่ไปข้างหน้าและหลังก็คือการเพิ่มและลดค่า z ส่วนการเคลื่อนที่ไปทางซ้ายและขวาก็คือการเพิ่มและลดค่า x ความคิดนี้ถูกต้องในกรณีที่เราหันหน้าเข้าหาแกน +z ของ Worldspace ตลอดเวลาเท่านั้น แต่อย่าลืมว่า เรามีการหมุนตัวด้วย ดังนั้น แกนที่เราหันหน้าเข้าหาอาจจะอยู่ระหว่างแกน x กับ z ของ Worldspace ก็ได้ ซึ่งจะต้องมีการเพิ่มและลดค่าในแกนทั้งสองพร้อมๆ กันตามมุมที่หมุน

และเนื่องจากแกน y ตั้งฉากกับแกน x และ z ดังนั้นเราจะต้องพิจารณาแกน y ถึง 2 ครั้ง ครั้งแรกพิจารณาในกรณีของแกน y กับแกน x ตามรูป 3-18 ก เป็นภาพใน Worldspace เมื่อเรามองอยู่ในแนวแกน z เข้าไป (มองเข้าไปในแนวแกน +z) จะได้ว่า มุม A คือมุมรอบแกน z ดังนั้น ค่าของตำแหน่งกล้องในแกน y จะมีค่าเท่ากับ รัศมีคูณด้วยค่า sin(มุมรอบแกน z) และค่าของตำแหน่งกล้องในแกน x เท่ากับ รัศมีคูณด้วยค่า cos(มุมรอบแกน z)

ทีนี้เราจะพิจารณาในกรณีของแกน y กับแกน z กันบ้าง ตามรูปที่ 3-18 ข แต่เปลี่ยนจากแกน x เป็น z โดยเราจะมองเข้าไปในแนวแกน -x จะได้ว่า มุม A ก็คือมุมรอบแกน x และค่าของตำแหน่งกล้องในแกน y จะยังคงเท่าเดิม (เพราะแกน x และ z มันใช้แกน y ร่วมกัน) ส่วนค่าของตำแหน่งกล้องในแกน z จะเท่ากับ รัศมีคูณด้วย cos(มุมรอบแกน x)



(ก) มุมรอบแกน y กรณีของแกน x กับ z *(ข) มุมรอบแกน y กรณีของแกน y กับ z*

รูปที่ 3-18 การคำนวณค่าตำแหน่งของกล้อง

เนื่องจากแกน x กับแกน z มันอยู่ในแนวระนาบเดียวกัน ดังนั้น มุมรอบแกน x กับ มุมรอบแกน z ก็จะมีค่าเท่ากัน ซึ่งจะใช้อันไหนก็ได้ แต่ในโปรแกรมจะใช้มุมรอบแกน x ซึ่งใช้ชื่อตัวแปรว่า g_fAngleX สำหรับรัศมีนั้นก็คือระยะทางที่กล้องเคลื่อนไป หรือเป็นความเร็วในการเคลื่อนที่ก็ว่าได้ เพราะถ้าระยะห่างมากก็จะเคลื่อนที่ได้เร็ว แทนด้วยค่าคงที่ชื่อ MOVING_SPEED และให้ตัวแปร g_fPosX, g_fPosZ และ g_fPosY แทนตำแหน่งของกล้องในแนวแกน x, z และ y ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชั่น sinf() และ cosf() จะส่งค่ากลับมาเป็นจำนวนจริง และจะต้องใช้มุมแบบเรเดียนเท่านั้น จึงต้องแปลงจากองศาเป็นเรเดียนโดยใช้ฟังก์ชั่น D3DXToRadian() นอกจากนี้ จริงๆ แล้วการเคลื่อนที่ไปข้างหน้าก็คือการบวกค่า x และ z แต่อย่าลืมว่า ตอนที่เราหมุนแกนจากรูปการหาจุดบนเส้นรอบวง ให้มาเป็นแบบ Worldspace นั้น แกน x (หรือแกน y เดิม) มันสลับค่าบวกและลบกันอยู่ ดังนั้นแทนที่จะบวกค่า x (คือแทนที่จะใช้ (+=) ก็ต้องเป็นลบแทน (ใช้ -=))

จะได้สูตรการเคลื่อนกล้องไปข้างหน้าดังนี้

```
- g_fPosX -= sinf(D3DXToRadian(g_fAngleY)) * cosf(D3DXToRadian(g_fAngleX)) *
MOVING_SPEED
- g_fPosY += sinf(D3DXToRadian(g_fAngleX)) * MOVING_SPEED;
- g_fPosZ += cosf(D3DXToRadian(g_fAngleY)) * cosf(D3DXToRadian(g_fAngleX)) *
MOVING_SPEED;
```

จะเห็นว่า เป็นการคูณค่า cos ของมุมรอบแกน x เข้าไปที่ตำแหน่งในแนวแกน x และ z ส่วนค่าของตำแหน่งของกล้องในแนวแกน y จะเป็นค่า sin ของมุมรอบแกน x ซึ่งใช้ค่า sin โดดๆ (เพราะแกน x และ z มันใช้แกน y ร่วมกัน) และไม่ต้องเอาค่า cos มาคูณ (เพราะจากรูป ค่า cos มันเป็นค่าในแนวระนาบของแกน x และ z ไม่ใช่ของแกน y)

สำหรับการเคลื่อนที่ถอยหลัง ก็แค่กลับเครื่องหมายบวกเป็นลบ ลบเป็นบวก ดังนี้

```
- g_fPosX += sinf(D3DXToRadian(g_fAngleY)) * cosf(D3DXToRadian(g_fAngleX)) *_
MOVING_SPEED;
- g_fPosY -= sinf(D3DXToRadian(g_fAngleX)) * MOVING_SPEED;
- g_fPosZ -= cosf(D3DXToRadian(g_fAngleY)) * cosf(D3DXToRadian(g_fAngleX)) *_
MOVING_SPEED;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกสิ่งหนึ่งที่ต้องพิจารณาด้วยก็คือ เมื่อมีการเลี้ยวซ้ายขวาและมีการเคลื่อนที่ ค่ามุมรอบแกน y จะต้องเปลี่ยนไปด้วยพร้อมทั้งค่ามุม CenterAngle ก็จะต้องเปลี่ยนไปด้วยขนาดที่เท่ากันด้วย เพราะเมื่อเราเลี้ยวกลับให้ตรงเหมือนเดิมแล้ว มุมที่เราเห็นก็จะเปลี่ยนไปจากเดิม เช่น

```
if(Rvary == 32)
{
        g_fAngleY = CenterAngle + 2*ROTATION_SPEED;
        CenterAngle = CenterAngle + 2*ROTATION_SPEED;
}
```

เมื่อได้ค่าตำแหน่งของกล้องในแนวแกน x, y และ z แล้ว ดังนั้นเราจึงคำนวณ View Matrix ของกล้องได้ตามลำดับดังนี้

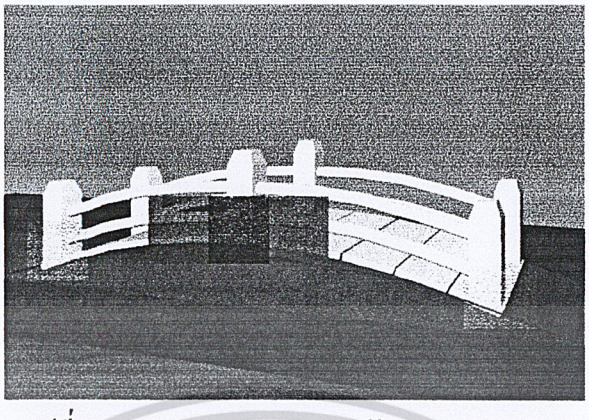
```
vecEyePoint   = D3DXVECTOR3(g_fPosX, g_fPosY, g_fPosZ);
vecLookatPoint = D3DXVECTOR3(g_fPosX, g_fPosY, g_fPosZ + 0.1f);
vecUp         = D3DXVECTOR3(0.0f, 1.0f, 0.0f);
D3DXMatrixLookAtLH(&matView, &vecEyePoint, &vecLookatPoint, &vecUp);
```

จากนั้นต้องนำไปรวมกับ เมทริกซ์ ที่เกิดจากการหมุนรอบแกน x และ y ที่คำนวณได้ก่อนหน้านี้ ก็จะได้มุมมองการเคลื่อนที่ตามที่ต้องการ

```
D3DXMatrixMultiply(&matView, &matView, &matRotation);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 การตรวจสอบการขึ้นลงสะพาน



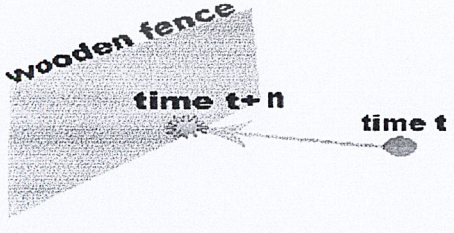
รูปที่ 3-19 การกำหนดความลาดชันของสะพาน

การขึ้นลงสะพานนั้นจะต้องทำการคำนวณหาตำแหน่งของสะพานก่อนว่าอยู่ตรงจุดไหนและขนาดเท่าไหร่ แล้วทำการแบ่งสะพานออกเป็น 7 ส่วนด้วยกันตามแถบสีในรูปที่ 3-19 โดยเราจะแบ่งจากตำแหน่งและขนาดที่ได้มา เพื่อกำหนดให้เมื่อเวลาเราเคลื่อนที่ผ่านแต่ละส่วนให้เงยหน้าขึ้นหรือก้มหน้าลง โดยในส่วนสีส้มนั้นจะเป็นส่วนที่เงยหน้าขึ้นเรื่อยๆ ส่วนสีฟ้านั้นจะก้มหน้าลงเรื่อยๆ ส่วนสีเทาตรงกลางจะให้มองขนานกับพื้น และในส่วนที่ไม่ได้ใส่สีไว้หมายความว่าไม่ให้เปลี่ยนแปลงมุมมอง

3.2.5 การตรวจสอบการชน

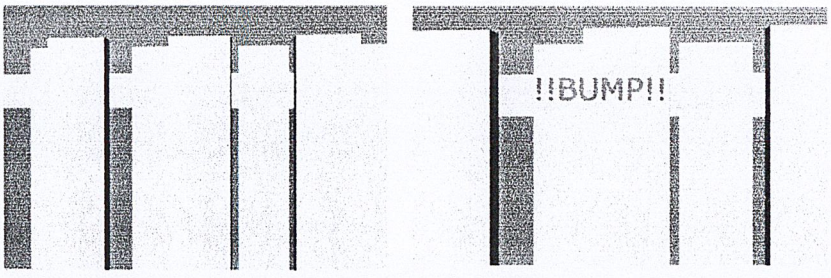
เราจะมีการตรวจสอบการชนตลอดเวลาว่า ตำแหน่งของรถจักรยานปัจจุบันอยู่ในบริเวณของวัตถุหรือบริเวณที่ถูกกำหนดไว้ว่าเข้าไปไม่ได้หรือไม่ เช่น รั้ว กำแพง หรือขอบสะพาน เป็นต้น โดยใช้เมธอด CheckCollision()

ตัวอย่าง จากรูปที่ 3-20 ณ เวลา t รถจักรยานอยู่ที่ตำแหน่งวงกลมสีแดง เมื่อเคลื่อนที่เข้าไปหากำแพง ณ เวลา t+n ตำแหน่งของรถจักรยานจะอยู่ในบริเวณของตัวรั้วไม้ ทำให้เกิดการชนขึ้นที่รั้ว ถ้าเกิดการชนขึ้นจะทำให้ไม่สามารถเดินต่อไปข้างหน้าจนกว่าจะหันไปทางที่ไม่ชนก่อน และจะปรากฏข้อความว่า "!!BUMP!!" ขึ้นมาบนจอภาพ ดังรูปที่ 3-21



รูปที่ 3-20 วิธีตรวจสอบการชน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพก่อนชน ณ เวลา t ภาพขณะชน ณ เวลา t+n

รูปที่ 3-21 แสดงภาพก่อนชนและขณะชนรั้ว

จากนั้นเราจะส่งค่าออกไปสั่งมอเตอร์ที่ติดอยู่กับรถจักรยานให้สั่น โดยจะสั่งงานผ่านพอร์ตดาต้า (378h) ซึ่งจะแบ่งเป็น 2 คำสั่ง คือ

สั่งให้มอเตอร์ทำงาน

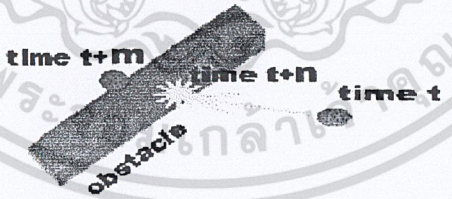
```
_outp(0x378,0xff);
```

สั่งให้มอเตอร์หยุดทำงาน

```
_outp(0x378,0);
```

3.2.6 การตรวจสอบการเคลื่อนที่ผ่านเกาะกลางถนน

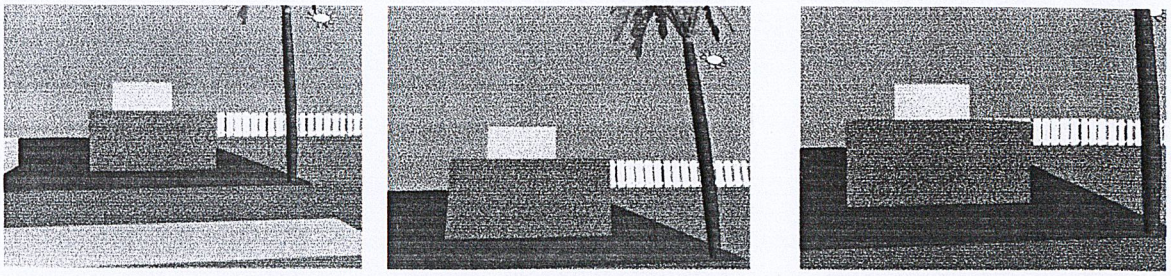
การเคลื่อนที่ผ่านเกาะกลางถนนนั้นเราได้มีการกำหนดตำแน่งของเกาะกลางถนนไว้แล้วจากโลกเสมือนจริงที่เราสร้าง และเมื่อเราเคลื่อนที่ผ่านตำแหน่งที่เป็นเกาะกลางถนน เราก็จะกำหนดให้ตำแหน่งที่มองลอยจากพื้นขึ้นไปอีกระดับหนึ่ง เพื่อให้เกิดความรู้สึกเหมือนกับรถจักรยานกระเด้งขึ้นจากพื้นไปอยู่บนเกาะกลางถนน



รูปที่ 3-22 การพิจารณาตำแหน่งรถจักรยานเมื่อผ่านเกาะกลางถนน

จากรูปที่ 3-22 ณ เวลา t เมื่อรถจักรยานอยู่ที่ตำแหน่งวงกลมสีแดง เมื่อเคลื่อนที่ถึงเกาะกลางถนน ณ เวลา t+n ตำแหน่งของมุมกล้องจะทำการยกขึ้นจนกว่ารถจักรยานจะเคลื่อนที่ผ่านตำแหน่งวงกลมสีแดง ณ เวลา t+m จึงค่อยให้ตำแหน่งของมุมกล้องยกกลับมาที่เดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

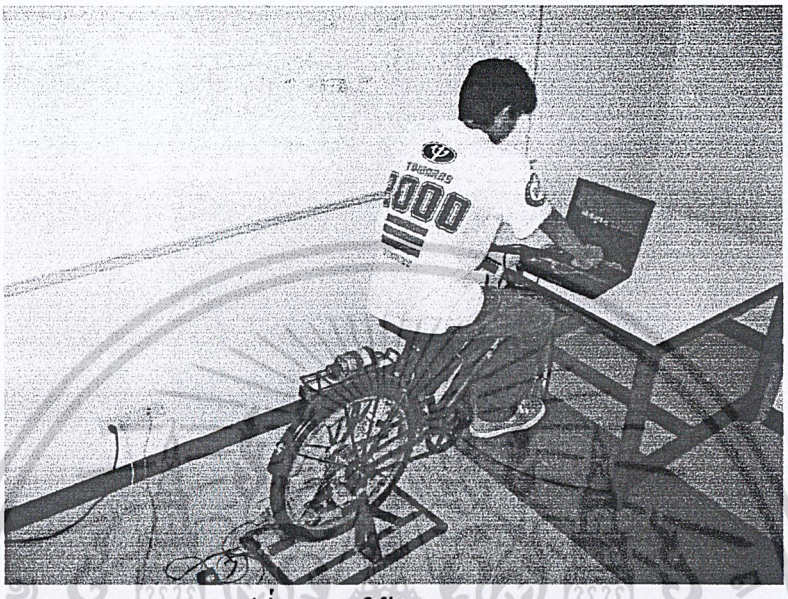


ณ เวลา t ก่อนผ่านเกาะกลาง ณ เวลา t+n ขณะผ่านเกาะกลางถนน ณ เวลา t+m หลังผ่านเกาะกลางถนน

รูปที่ 3-23 แสดงให้เห็นถึงมุมกล้องขณะเคลื่อนที่ผ่านเกาะกลางถนน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทำงานโดยรวมทั้งหมด



รูปที่ 4-1 การใช้งาน VR Bicycle

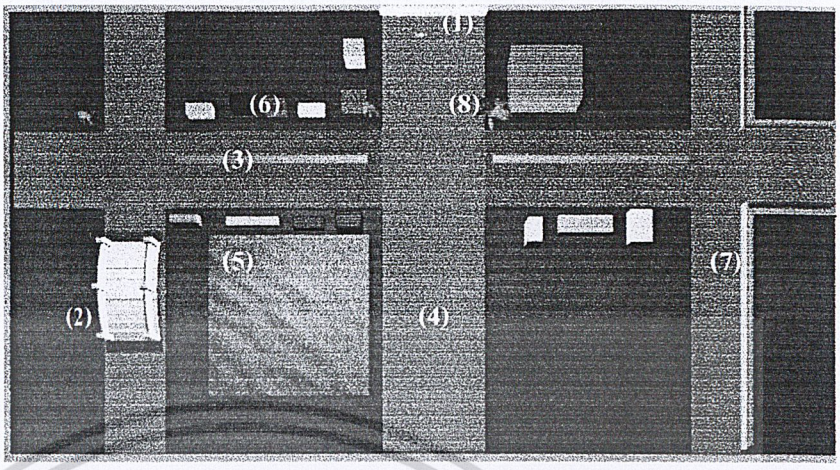
รูปที่ 4-2 แสดงให้เห็นถึงโลก 3 มิติทั้งหมดจากมุมมองด้านบน

รูปที่ 4-2 เป็นรูปที่มองมาจากด้านบน แสดงให้เห็นถึงแผนที่ของโลกทั้งหมด และแสดงถึงตำแหน่งของวัตถุต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 ส่วนประกอบภายในโลกเสมือน 3 มิติ

1) รั้วไม้
2) สะพาน
3) เกาะกลางถนน
4) ถนน
5) สนามหญ้า
6) ตึก
7) กำแพง
8) ต้นไม้



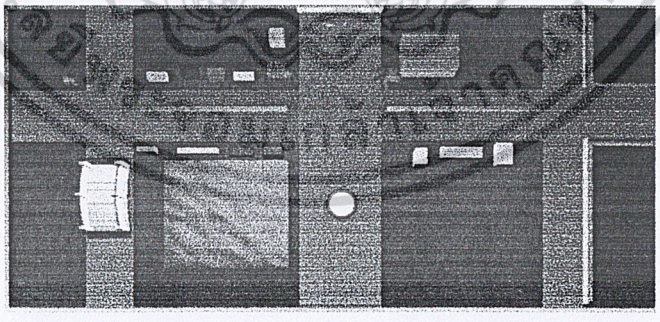
รูปที่ 4-3 แสดงตำแหน่งของส่วนประกอบ

4.2 ลักษณะของโลกเสมือน 3 มิติ

1) แผนที่เป็นสี่เหลี่ยม และส่วนประกอบต่าง ๆ จะอยู่ภายในแผนที่นี้
2) ผู้ใช้ไม่สามารถออกจากแผนที่นี้ได้ คือ ไม่สามารถไปไกลกว่านี้ได้
3) โลกเสมือน 3 มิตินี้ ไม่มีการวน คือเมื่อผู้ใช้ต้องการกลับมา ณ ตำแหน่งเริ่มต้น ผู้ใช้ต้องเดินทางย้อนกลับมาทางเดิม

4.3 การเดินทางภายในโลกเสมือน 3 มิติ

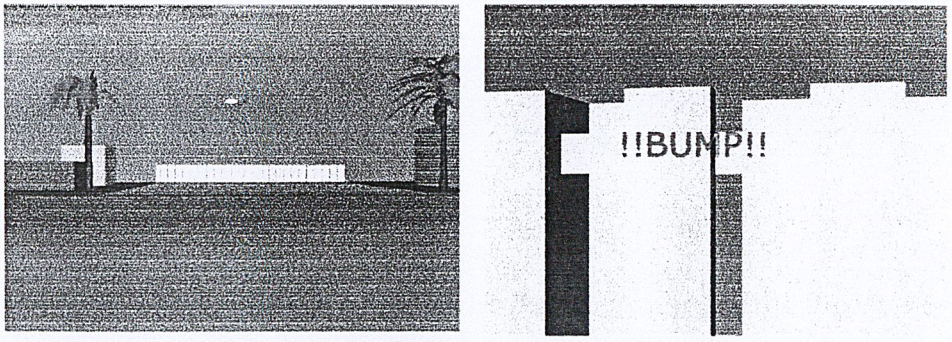
1) เมื่อเริ่มต้น ผู้ใช้จะอยู่ตำแหน่งตรงกลางโลก 3 มิติ ทางถนนด้านล่าง



รูปที่ 4-4 จุดวงกลมสีขาวแสดงถึงตำแหน่งเริ่มต้นของรถจักรยาน

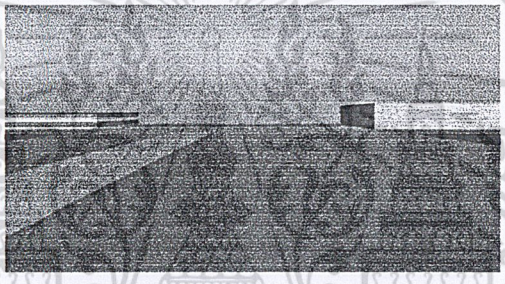
2) เมื่อผู้ใช้เคลื่อนที่ตรงไปจะเจอสี่แยก
   2.1) ถ้าผ่านสี่แยกตรงไปจะเจอรั้วไม้ สามารถชนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



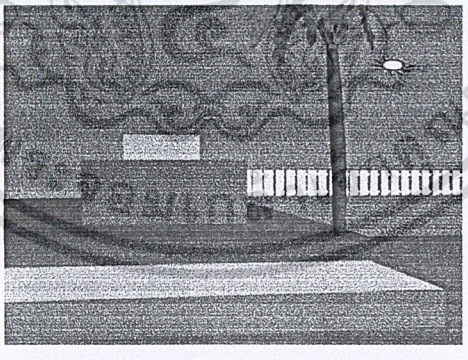
รูปที่ 4-5 รั้วไม้

2.2) ถ้าเลี้ยวขวาและตรงไปจะเจอสี่แยกที่สอง และจะเห็นกำแพงอยู่ทางฝั่งซ้ายและขวาของสี่แยก



รูปที่ 4-6 กำแพง

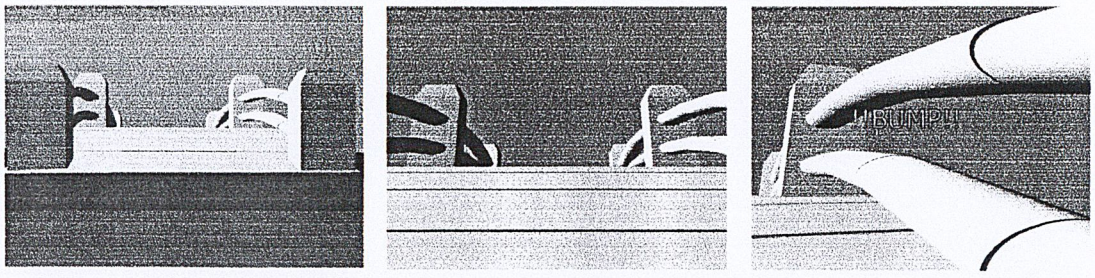
2.3) ถ้าเลี้ยวซ้ายจะเห็นเกาะกลางถนน สามารถเคลื่อนที่ผ่านได้



รูปที่ 4-7 เกาะกลางถนน

- เมื่อเคลื่อนที่ตรงไปเรื่อย ๆ จะเจอสี่แยกที่สาม
  - เมื่อเลี้ยวซ้ายจะเจอสะพาน สามารถเคลื่อนที่ขึ้นลงได้และชนขอบสะพานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-8 เมื่อเจอสะพานสามารถขึ้นได้ และสามารถชนขอบสะพานได้

2.4) เมื่อต้องการกลับไปยังตำแหน่งเริ่มต้น ให้ย้อนกลับไปทางเดิม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุป

โครงงานนี้พัฒนาได้สำเร็จลุล่วงตามเป้าหมาย โดยเราได้ทำการออกแบบและสร้างโลกเสมือน 3 มิติขึ้นมาเป็นโลกเสมือนที่ให้ผู้ใช้สามารถกระทำการโต้ตอบได้ และเราได้ทำการออกแบบและพัฒนาในส่วนของอุปกรณ์อินพุตแล้วเอาไปสร้างโลกเสมือนให้สอดคล้องกับความเป็นจริง ซึ่งก็ได้แก่ ระบบเซ็นเซอร์แสงนำไปติดที่ล้อรถจักรยานเอาไว้ตรวจสอบการเคลื่อนที่ของล้อ และระบบตรวจสอบการเลี้ยวซ้ายขวา นำไปติดไว้ที่ส่วนของแฮนด์รถจักรยาน นอกจากนั้นแล้วเรายังได้ทำการออกแบบและสร้างระบบการสร้างปฏิกิริยาตอบสนองขึ้นมา ซึ่งเราได้ใช้การสั่นของมอเตอร์มาประยุกต์ใช้กับระบบของเรา

สำหรับลักษณะของการทำงานโดยรวมคือ เมื่อผู้ใช้ถีบรถจักรยานตรงไปเรื่อย ๆ ภาพในโลกเสมือนก็จะมีการเคลื่อนที่ เมื่อผู้ใช้เลี้ยวซ้ายขวา ภาพในโลกเสมือนก็จะเลี้ยวตาม และเมื่อผู้ใช้ถีบรถจักรยานชนวัตถุที่ปรากฏอยู่ในโลกเสมือน ระบบก็จะส่งสัญญาณโต้ตอบกลับมาเป็นการสั่นที่ตัวรถจักรยาน

5.1 ปัญหาและแนวทางแก้ไข

5.1.1 ปัญหาที่เกิดในส่วนของฮาร์ดแวร์

1) เมื่อส่งค่า S7-S4 ในพอร์ตแสดงสถานะเป็น 1110 ซึ่งเกิดจากการหมุนตัวต้านทานปรับค่าได้และเซ็นเซอร์ แล้วจะทำให้ค่าที่ออกมาจากพอร์ตคาต้าเปลี่ยนแปลงไปโดยทำให้บางบิตเป็น 1 และบางบิตเป็น pulse จึงต้องค้นหาบิตที่ไม่โดนผลกระทบจึงพบว่าบิตที่ไม่โดนผลกระทบคือบิตที่ 4 จึงได้ใช้บิตที่ 4 นี้แทน
2) ถ้าเสียบสายจากพอร์ตขนานไว้ก่อนที่จะเปิดสวิตซ์ไฟของวงจรทั้งหมด จะทำให้วงจรไม่สามารถรับค่ามาจากส่วนตรวจการเลี้ยวได้ จึงได้ลองผิดลองถูกจนได้เจอวิธีแก้คือให้เปิดสวิตซ์ไฟของวงจรก่อน แล้วจึงค่อยเสียบสายที่ต่อมาจากพอร์ตขนาน
3) ข้อนี้เป็นปัญหาที่เกิดจากอุปกรณ์ไม่มีความถาวรคือในส่วนที่ใช้ในการตรวจสอบการเลี้ยวซ้ายเลี้ยวขวา ที่เป็นตัวต้านทานปรับค่าได้นั้น ถ้ามีคนไปหมุนเล่น ทำให้ไม่ตรงกับตัวคอรถจักรยานที่หันไปทางทิศใดแล้ว จะทำให้ค่าที่ส่งไปเข้าคอมพิวเตอร์ไม่ตรงกับที่บังคับรถจักรยานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.2 ปัญหาที่เกิดขึ้นในส่วนซอฟท์แวร์

1) ปัญหาจากการโหลดโมเดล .x เข้ามาไว้ในโปรแกรม DirectX โมเดลที่เราได้สร้างขึ้นมาจาก 3DsMax เมื่อเรานำมาโหลดเข้าไว้ในโปรแกรม DirectX แล้ว โมเดลที่ได้จะดูกลับหัว เนื่องจากใน 3DsMax กับ DirectX ใช้ระบบพิกัดคนละชนิดกัน ใน 3DsMax ใช้ Right-Handed แต่ใน DirectX ใช้ Left-Handed เราจึงต้องแก้ไขโดยทำการหมุนโมเดลให้ถูกทิศทางเสียใหม่

2) การเพิ่มโมเดลในโลก 3 มิติส่งผลกระทบโดยตรงกับความเร็วของโปรแกรม เมื่อเราต้องการให้โลก 3 มิติของเราดูมีความสมจริงมากยิ่งขึ้น เราจึงทำการใส่วัตถุพร้อม Texture ต่าง ๆ เข้าไป แต่เมื่อยิ่งใส่เข้าไปเยอะเท่าไหร่ก็ยิ่งทำให้โปรแกรมทำงานช้าลงเท่านั้น เราจึงสามารถใส่วัตถุได้เฉพาะเท่าที่จำเป็นเท่านั้น

3) การประมวลผลกราฟิก 3 มิติไม่ทันความต้องการ การเรนเดอร์ภาพในแต่ละเฟรมนั้นต้องใช้เวลาค่อนข้างมากเมื่อเทียบกับการอ่านค่าอินพุตจากพอร์ตขนานแล้ว ทำให้ค่าที่อ่านได้นั้นไม่ใช่ค่าที่ถูกต้อง ทำให้เราพลาดค่าที่ถูกอ่านเข้ามาแล้วไป ซึ่งส่งผลให้โปรแกรมดูจะทำงานได้ไม่ทันความต้องการ เราจึงทำการแก้โดยการเพิ่มแผ่นกระดาษที่ติดกับล้อให้มีความถี่มากขึ้นซึ่งทำให้ค่าที่อ่านได้มีโอกาสเป็นค่าที่ถูกต้องมากยิ่งขึ้น

5.2 แนวทางการพัฒนาต่อ

1) พัฒนาในส่วนของกราฟิก 3 มิติให้มีขนาดของโลก 3 มิติที่ใหญ่ขึ้นพร้อมทั้งมีความสมจริงมากขึ้น หรือสามารถที่จะจำลองสถานที่จริง หรือ อาจจำลองสถานที่ในอดีต เช่น ราชธานีเก่าในสมัยอยุธยา เป็นต้น

2) พัฒนาในส่วนของการเคลื่อนที่ด้วยรถจักรยานให้มีความถูกต้องมากขึ้น โดยการพัฒนาในส่วนของอุปกรณ์อินพุตให้มีความแม่นยำและมีความละเอียดมากขึ้น

3) พัฒนาในส่วนของปฏิกิริยาตอบสนองให้เป็นธรรมชาติมากขึ้น เช่น ระบบทำให้ล้อฝืดเมื่อมีการขึ้นลงสะพาน โดยการติดเบรกที่ล้อเพิ่มเติม เมื่อรถจักรยานขึ้นสะพานก็สั่งให้เบรกที่ล้อทำให้ล้อมีความฝืดซึ่งส่งผลให้เราถีบได้ยากขึ้น เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ชัยวัฒน์ คำรัตน์ : "ก้าวแรกกับ DirectX 8 SDK ด้วย Visual C++" , คอมพิวเตอร์ เอจ เทคโนโลยี, 2544

[2] นิรุธ อำนวยศิลป์ : "เขียนเกมอย่างมืออาชีพด้วย Visual C++ และ DirectX", Info Press, 2545, p64-194

[3] ปิยะบุตร สุทธิดารา : "Discreet 3ds max 4", Info Press, 2544

[4] ยุทธนา ลัลาศวัฒนกุล : "คู่มือการเขียนโปรแกรมและใช้งาน Visual C++ 6.0 ฉบับโปรแกรมเมอร์", Info Press, 2544

[5] กฤษดา ใจเย็น และ ชัยวัฒน์ ลิ้มพรจิตรวิไล : "เรียนรู้และปฏิบัติการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่านพอร์ตขนาน", Innovative Experiment Co., Ltd., p1-40

[6] Roy S. Kalawsky : "The Science of Virtual Reality and Virtual Environments", Addison-Wesley

[7] http://www.cwinapp.com

[8] http://www.andypike.com

[9] http://vads.ahds.ac.uk/guides/vr_guide

[10] http://www.ia.hiof.no/~michaell/home/vr/vrhiof98/

[11] http://francis.dupont.free.fr/coindev/english/direct3d/convert.htm

[12] http://www.3dlinks.com

[13] http://astronomy.swin.edu.au/~pbourke/geomformats/directx/

[14] http://www.mvps.org/vbdx/tutorials/exportx/

[15] http://www.thaiio.com

[16] http://www.lvr.com/parport.htm

[17] http://www.logix4u.cjb.net

[18] http://smile.student.utwente.nl/~jur/portfolio/printio/

[19] http://www.cis.ksu.edu/chert/students/sean/74HC_HCT541_CNV_2.pdf

[20] http://cecsfp.cecs.evansville.edu/pdf/adc0804.pdf

[21] http://www.farnell.com/datasheets/43480001.pdf

[22] http://www.lvr.com/files/pppinout.pdf

[23] http://www.gunpowder.freeserve.co.uk/wheels/potworks.htm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้