สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

กิตติกรรมประกาศ

โครงการวิจัยนี้ได้รับทุนอุดหนุนจาก โครงการสำนักวิจัยการสื่อสารและเทคโนโลยีสารสนเทศ(Research Center for Communications and Information Technology : ReCCIT) สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ประจำปีงบประมาณ พ.ศ. 2551

RCIT
QA
76.9
.V5
ฆ176ก
เลขหมู่.....
เลขทะเบียน.120283
วัน, เดือน, ปี.13 ก.ย. 2555

b.12336464
i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทที่ 1 โปรแกรมสร้างภาพและแอนิเมชั่น Blender

1.1 จุดเด่นของโปรแกรม 1

1.2 ความต้องการของระบบคอมพิวเตอร์โปรแกรม Blender 1

1.3 หลักการทำงานของเบลนเดอร์เกมเอ็นจิ้น 4

บทที่ 2 ถุงมือและพวงมาลัยอิเล็กทรอนิกส์

2.1 อุปกรณ์ถุงมืออิเล็กทรอนิกส์ 7

2.1.1 ตัวชุดอุปกรณ์ถุงมืออิเล็กทรอนิกส์ 7

2.1.2 ตัวชุดอุปกรณ์ถุงมืออิเล็กทรอนิกส์ประกอบด้วย 8

2.1.3 การเชื่อมต่ออุปกรณ์ 8

2.1.4 หลักการทำงานของตัวอุปกรณ์ 9

2.1.5 คำสั่งพื้นฐานในการใช้งาน CGIU 9

2.2 พวงมาลัยอิเล็กทรอนิกส์ 11

2.2.1 การเชื่อมต่ออุปกรณ์ 12

บทที่ 3 การเขียนโปรแกรมติดต่อระหว่างโปรแกรม Blender กับอุปกรณ์รับข้อมูลเข้าประเภทถุงมือและพวงมาลัย

3.1 ถุงมืออิเล็กทรอนิกส์ 13

3.1.1 การติดต่อโดยตรงจาก Interactive Console 13

3.1.2 การใช้งาน Api สำหรับควบคุม CyberGlove ผ่านทางพอร์ทอนุกรม 14

3.2 พวงมาลัยอิเล็กทรอนิกส์ 20

3.2.1 ตัวอย่างการเขียนคำสั่งควบคุม steering wheel ผ่าน

Blender Game Engine ด้วย โมดูล GameLogic 23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

3.2.2 ตัวอย่างงานที่สร้างจากใช้การควบคุมพวงมาลัยอิเล็กทรอนิกส์
ผ่าน Blender Game Engine ด้วย โมดูล GameLogic 28

บทที่ 4 สรุปผลการทดลอง
4.1 ถุงมืออิเล็กทรอนิกส์ 32
4.2 พวงมาลัยอิเล็กทรอนิกส์
เอกสารอ้างอิง 33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
โปรแกรมสร้างภาพและแอนิเมชัน Blender

โปรแกรม Blender เป็นโปรแกรมสร้างงาน 3 มิติที่มีความสามารถในการสร้างโมเดลรูปทางต่างๆ ทั้งยังกำหนดพื้นผิวหรือลาดลายให้กับวัตถุได้ สามารถจัดแสง กำหนดมุมมอง สร้างงานชิ้นให้เป็นอนิเมชั่น พร้อมทั้งใส่งานเอ็ฟเฟ็กต์สร้างความเหมือนจริงและชวนติดตาม จนกระทั่งประมวลผลงานทั้งออกมาเป็นงาน 3 มิติที่สมบูรณ์แบบ นอกจากนั้นยังเป็นโปรแกรมที่รองรับการสร้างเกม โดยมีเครื่องมือที่ช่วยในการทำโมเดลสำหรับเกม มีการทำงานอื่นๆที่เกี่ยวกับการสร้างเกม และเป็นโปรแกรมที่เป็นโอเพ่นซอร์ส เนื่องจากระบบวีอาร์จำเป็นจะต้องมีอุปกรณ์ที่ทำหน้าที่เป็นอุปกรณ์อินพุต ซึ่งอุปกรณ์ที่ทำหน้าที่เป็นอุปกรณ์อินพุทนั้นมีอยู่หลายชนิด โดยอุปกรณ์อินพุทที่จะนำมาทดลองในรายงานฉบับนี้จะเป็นอุปกรณ์ถุงมือและพวงมาลัยอิเล็กทรอนิกส์ซึ่งรายละเอียดจะกล่าวถึงต่อไป

1.1 จุดเด่นของโปรแกรม Blender

โปรแกรม Blender เป็นโปรแกรม Open Source ที่สามารถดาว์นโหลดมาใช้งานได้ฟรี และยังมีความสามารถทัดเทียมกับโปรแกรมสร้างงาน 3 มิติโปรแกรมอื่นๆ โดยจุดเด่นที่น่าสนใจของโปรแกรม Blender มีดังนี้

- เป็นโปรแกรมที่ใช้ทรัพยากรระบบและพื้นที่ในการติดตั้งโปรแกรมน้อย
- มีความสามารถในการสร้างงานได้หลายรูปแบบ เช่น การสร้างการตูนแอนิเมชัน งานดีไซน์ งานสถาปัตยกรรมและตกแต่งภายใน การสร้างสเปเชียลเอ็ฟเฟ็กต์ และการสร้างเกมเป็นต้น
- ทำงานได้หลายแพลตฟอร์ม เช่น Microsoft Windows, Mac OS, Linux
- เป็นโปรแกรมที่มีกลุ่มใช่งานมากกว่า 250,000 คนทั่วโลก รวมทั้งกลุ่มศิลปินที่นิยมใช้ Blender และเปิดเว็บไซต์ให้ความรู้พร้อมเว็ปบอร์ดให้สอบถามปัญหาเกี่ยวกับการใช้งานได้ตลอดเวลา เช่น www.blender.org www.blendernation.com, http://blenderartists.org, http://3d-synthesis.com เป็นต้น

โปรแกรม Blender สามารถใช้เป็นโปรแกรมพื้นฐานสำหรับผู้ที่ต้องการศึกษาการสร้างงาน 3 มิติ รวมทั้งบริษัทขนาดเล็กและขนาดกลาง ก็สามารถนำโปรแกรมไปใช้สร้างชิ้นงานตามวัตถุประสงค์ของบริษัทได้เช่นกัน เนื่องจากไม่ต้องจ่ายงบประมาณในการซื้อโปรแกรมมาใช้นั่นเอง

1.2 ความต้องการของระบบคอมพิวเตอร์โปรแกรม Blender

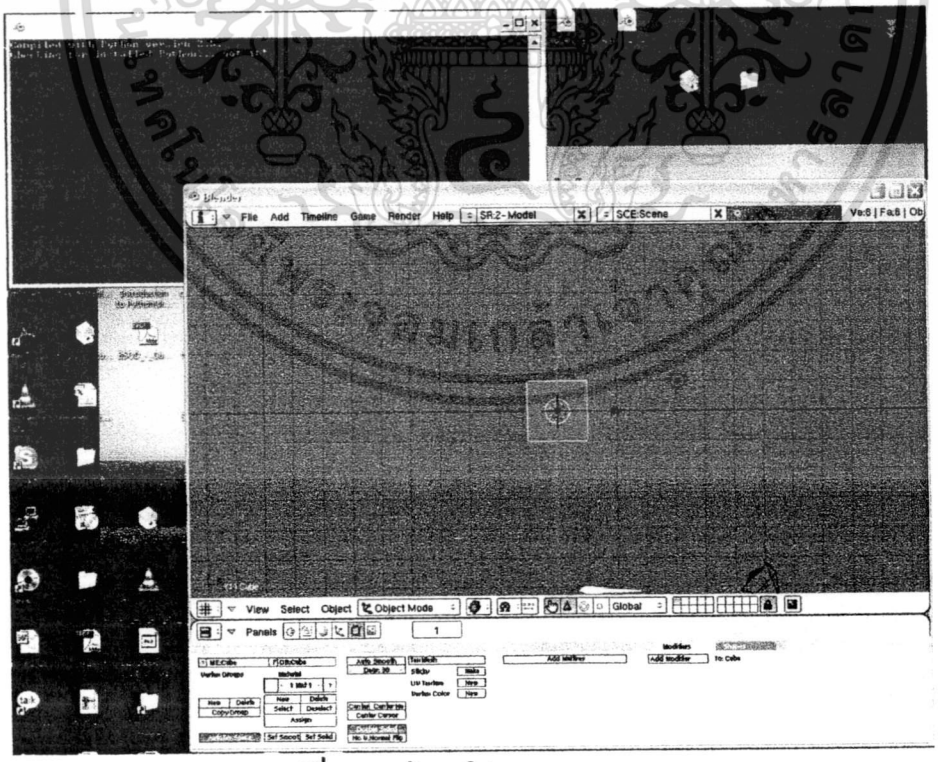
โปรแกรม Blender ถูกพัฒนาให้ทำงานได้ดีขึ้น มีความยืดหยุ่นของหน้าต่างการทำงาน (GUI หรือ Graphic User Interface) สูง และมีขนาดเบา ดังนั้นความต้องการระบบสำหรับคอมพิวเตอร์จึงไม่สูงมากนัก ซึ่งมี่เป็นข้อดีของ Blender

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบคอมพิวเตอร์	รายละเอียด
ความเร็ว CPU	Windows 98, ME, 2000, XP/Mac OS X 10.2,10.4 /Linux 2.26 i386,
	Linux 2.3.2 PPC
แรม	300 MHz หรือสูงกว่า
พื้นที่สำหรับลงโปรแกรม	20 MB
การ์ดแสดงผล	1024x768 16-bit color แรม 8 MB(แนะนำให้รองรับมาตรฐาน OpenGL)
ระบบและอุปกรณ์อื่นๆ	เมาส์แบบ 3 ปุ่ม

ตารางที่ 1.1 แสดงความต้องการของระบบในการใช้โปรแกรม

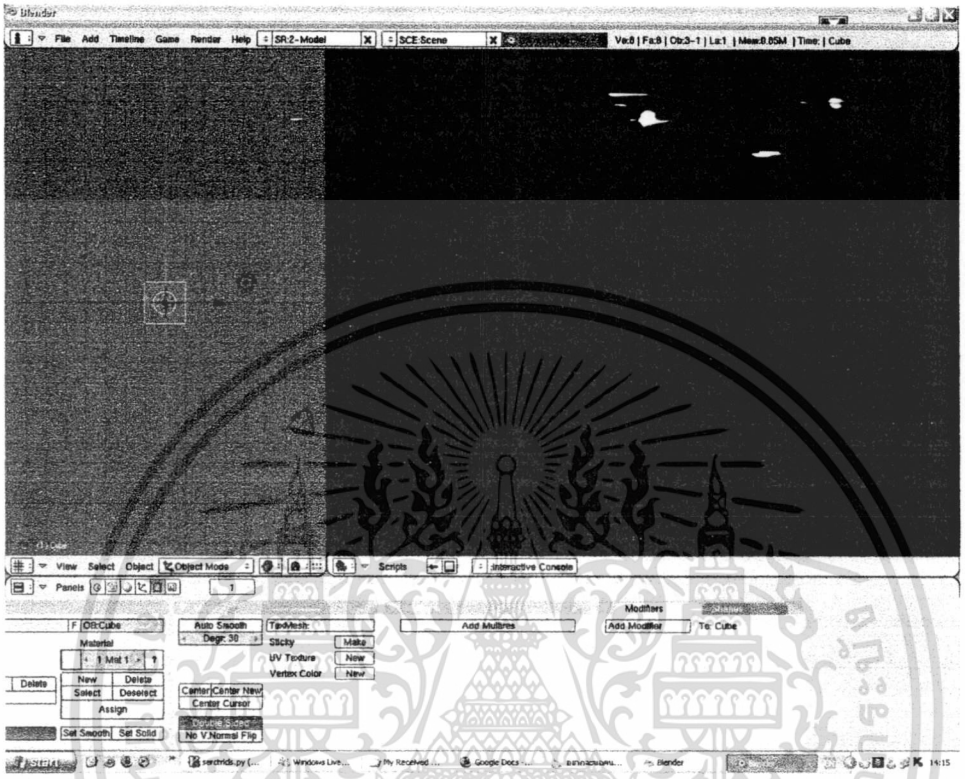
โปรแกรม Blender มีการจัดวางหน้าเป็นลักษณะของ Panel ผู้ใช้งานสามารถเพิ่ม Panel หรือแบ่ง Panel ได้หลายรูปแบบ



รูปที่ 1.1 หน้าจอโปรแกรม Blender

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รวมทั้งมี Panel ที่เราสามารถทำการเขียนภาษา Python เพื่อเพิ่มการควบคุมส่วนประกอบต่างๆเช่น โมเดล ตัวกล้อง แสงเงา การเคลื่อนไหว รวมทั้งการสร้างอนิเมชั่นได้



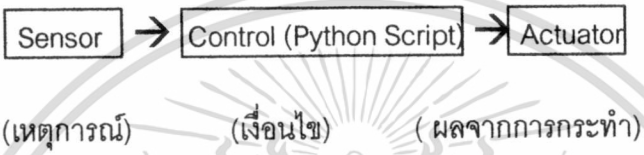
รูปที่ 1.2 แสดงรูปส่วนของ python interactive console script

โดยส่วนของ Python Script นั้นผู้ใช้งานสามารถรวมเอาโมดูลที่ทำงานเฉพาะด้านที่เขียนขึ้นด้วยภาษา Python เช่น โมดูลที่ใช้ในการติดต่อกับพอร์ทอนุกรมที่ชื่อว่า pySerial หรือ โมดูลที่ใช้ทำการติดต่อกับ directX ที่ชื่อว่า DirectPython โดยส่วนตัวของโปรแกรม Blender เองก็จะมีส่วนที่ทำการควบคุมจัดการเกี่ยวกับการสร้างเกมที่ชื่อว่า Blender Game Engine เตรียมไว้ให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

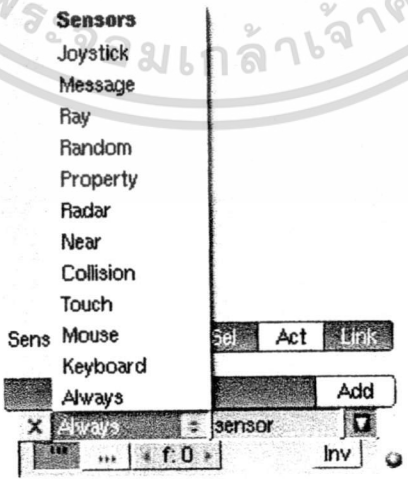
1.3 หลักการทำงานของเบลนเดอร์เกมเอ็นจิ้น (Blender Game Engine)

เบลนเดอร์เกมเอ็นจิ้น ประกอบด้วย สามส่วนที่ทำงานร่วมกัน โดยในการกำหนดการควบคุมในแต่ละครั้ง ทั้งสามส่วนจะทำงานสัมพันธ์กันตามแผนภาพดังนี้



รูปที่ 1.3 แสดงการทำงานของ เบลนเดอร์เกมเอ็นจิ้น

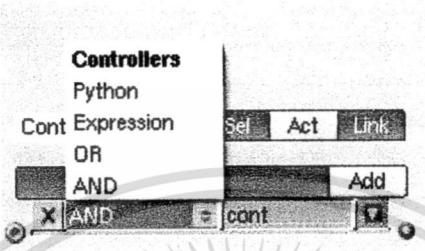
1.3.1 Sensor เกี่ยวกับการรับค่าที่ได้จาก แป้นพิมพ์, เมาส์ หรือ จอยแพด(Joy pad) แล้วกำหนดเป็นเหตุการณ์ต่างๆ เช่น เหตุการณ์นี้เกิดขึ้นตลอดเวลา,เกิดขึ้นเมื่อวัตถุเข้าใกล้สิ่งอื่น,เกิดขึ้นเมื่อใช้เมาส์ , เกิดขึ้นเมื่อกดปุ่ม(กำหนดปุ่มที่ต้องการ),เกิดขึ้นเมื่อวัตถุเข้าใกล้สิ่งอื่นในทิศทางที่กำหนด,เกิดขึ้นเมื่อวัตถุเข้าใกล้สิ่งที่กำหนดโดยไม่มีสิ่งอื่นกีดขวาง รูปที่ 1.3 แสดงประเภทของ Sensor ที่มีทั้งหมด



รูปที่ 1.4 แสดงส่วนของ Sensor

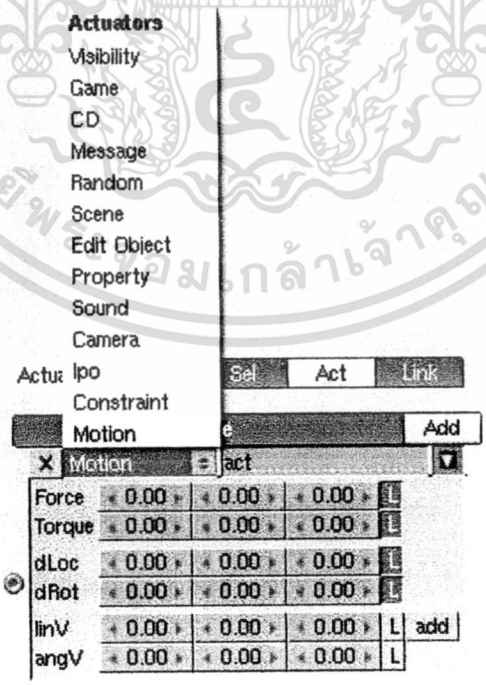
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.2 Controller เป็นเงื่อนไขใช้ควบคุม และ ตัดสินใจ เช่น AND ใช้ควบคุมให้ทำงานเมื่อมีอินพุตเข้ามาทุกตัว ,OR ทำงานเมื่อมี input เข้ามาบางตัว, Expression การกำหนดเงื่อนไข แบบคำสั่งภายใน 1 บรรทัด เป็นต้น



รูปที่ 1.5 แสดงส่วนของ Controller

1.3.3 Actuator จะทำงานเมื่อมีการกระตุ้นมาจาก Controllers เป็นการกำหนดผลที่ต้องการอย่างง่าย อาจแสดงผลออกเป็นรูปแบบต่างๆได้ทั้ง เสียง , ภาพ, และ แรงกระทำกับจอยแพด(Joy pad Force-Feed-Back) ยกตัวอย่างเช่น การหมุน การเคลื่อนที่ การสร้างเสียง เมื่อกดปุ่ม restart game เป็นต้น



รูปที่ 1.6 แสดงส่วนของ Actuator

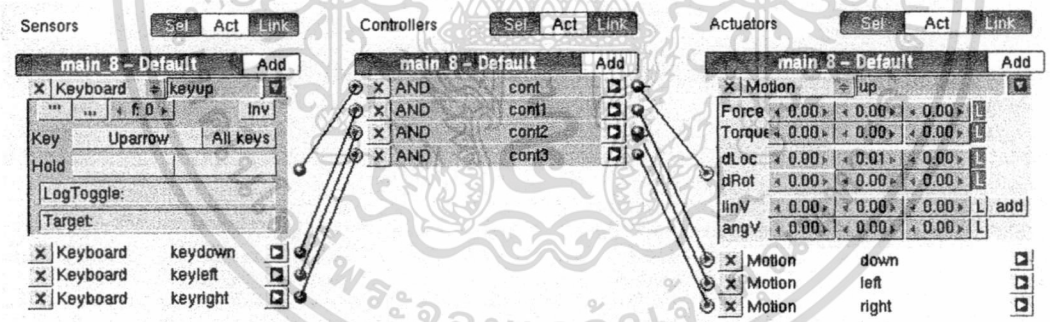
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการกำหนดเหตุการณ์ในเกม

Sensor(เหตุการณ์)	Control(เงื่อนไข)	Actuator(ผลจากการกระทำ)
กดปุ่มลูกศรขึ้น	ปกติ	เดินหน้า
กดปุ่มลูกศรขึ้น	และ	เลี้ยวซ้าย
กดปุ่มลูกศรซ้าย		
เกิดการชน	ปกติ	กระเด้ง

ตารางที่1.2 แสดงตัวอย่างการกำหนดเหตุการณ์ในเกม

ตัวอย่างการกำหนดเหตุการณ์ในเกมด้วยเบลนเดอร์เกมเอ็นจิ้น (Blender Game Engine)



รูปที่ 1.7 แสดงการกำหนดเหตุการณ์ในเกมด้วยเบลนเดอร์เกมเอ็นจิ้น

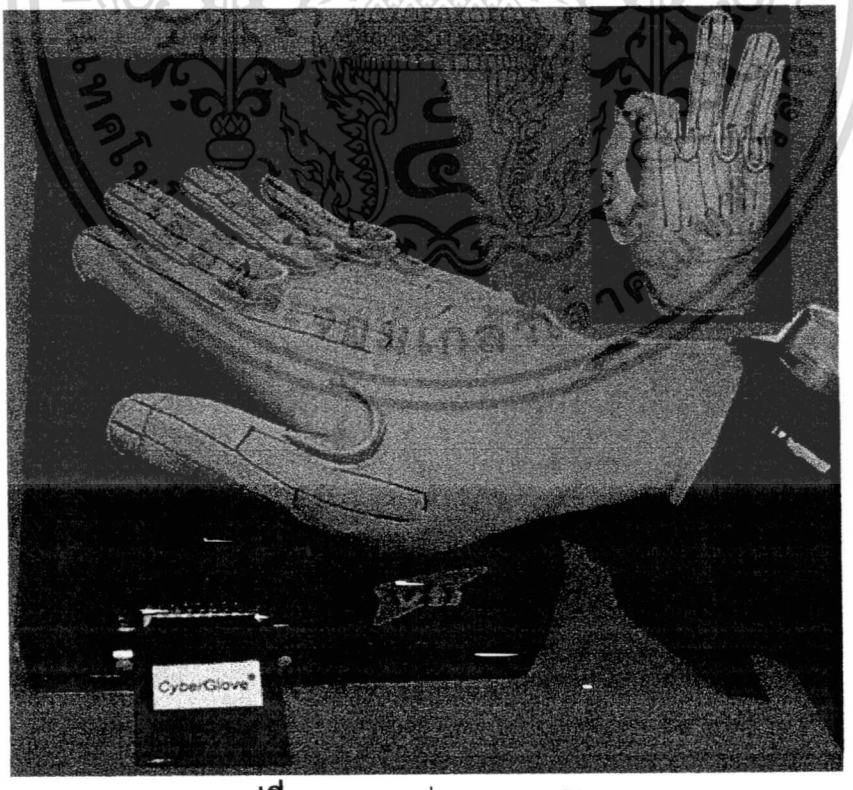
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ถุงมือ และ พวงมาลัยอิเล็กทรอนิกส์

ในปัจจุบันมีอุปกรณ์รับอินพุตหลายประเภทมากมายเช่น คีย์บอร์ด เมาส์ จอยสติ๊ก กล้องวีดีโอ ทำหน้าที่ในการรับข้อมูลในรูปลักษณะต่างๆไว้ในคอมพิวเตอร์ ซึ่งในการทดลองนี้กล่าวถึงการนำอุปกรณ์อินพุตประเภทถุงมืออิเล็กทรอนิกส์ซึ่งรับอินพุตเกี่ยวกับลักษณะท่าทางการเคลื่อนไหวของนิ้วมือผ่าน Sensor และอุปกรณ์พวงมาลัยอิเล็กทรอนิกส์ที่รับอินพุตเกี่ยวกับลักษณะการเคลื่อนไหวของพวงมาลัยปุ่มและคันบังคับต่างๆ นำเพื่อนำข้อมูลเข้ามาใช้ในการประมวลร่วมกับโปรแกรม Blender Game Engine เพื่อการจำลองอย่างมีปฏิสัมพันธ์

2.1 อุปกรณ์ถุงมืออิเล็กทรอนิกส์

เป็นอุปกรณ์ถุงมืออิเล็กทรอนิกส์ที่ประกอบด้วยตัวรับสัมผัส (Sensor) ซึ่งติดตั้งอยู่ตามตำแหน่งต่างๆ บนมือของผู้ใช้เพื่อทำหน้าที่แปลงการเคลื่อนไหวต่างๆของมือเป็นสัญญาณไฟฟ้าซึ่งแสดงอยู่ในรูปตัวเลข องศาต่างๆ โดยนักวิจัยสามารถนำข้อมูลเหล่านั้นมาประยุกต์ใช้งานได้หลายด้าน



รูปที่ 2.1 แสดงรูปของCyberGlove

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 ตัวชุดอุปกรณ์ถุงมืออิเล็กทรอนิกส์ประกอบด้วย

- ส่วนที่ทำหน้าที่รับข้อมูลจะเป็นถุงมือ (CG2202) ที่มีตัวรับสัมผัส 22 ตัวส่งค่าที่แปลงจากการเคลื่อนไหวของนิ้วมือผ่านทางพอร์ทอนุกรม (Serial Port) มาพร้อมกับสายเคเบิ้ลและตัวเชื่อมต่อแบบ D-sub ตัวผู้ 44 พิน (high-density)
- ส่วนอินเตอร์เฟส CyberGlove Interface Unit (CGIU) ซึ่งเป็นส่วนที่ทำการขยายสัญญาณและแปลงสัญญาณให้รูปแบบของข้อมูลดิจิตอล
    - ตัวเชื่อมต่อตัวเมีย 44 พิน (high-density) เพื่อเชื่อมต่อกับถุงมือ
    - LED เพื่อแสดงสถานะเปิด/ปิดอยู่ด้านหน้าของ CGIU
    - ตัวเชื่อมต่อพอร์ทแบบอนุกรม DE-9 ตัวผู้ (RS232C)
    - ตัวเชื่อมต่อพอร์ทแบบ analog/sync DE-9 ตัวเมีย
    - ตัวเชื่อมต่อแหล่งจ่ายไฟแบบ DIN 5 พิน ตัวเมีย
    - สวิตซ์ DIP แบบ 8 ตำแหน่ง
    - สวิตซ์เปิด/ปิด
    - สวิตซ์รีเซ็ท
- สายเชื่อมต่อแหล่งจ่ายไฟแบบ DIN 5 พินตัวผู้
- สายเคเบิ้ลแบบอนุกรมเพื่อเชื่อมต่อชุด CyberGlove กับเครื่องคอมพิวเตอร์

2.1.3 การเชื่อมต่ออุปกรณ์

การเชื่อมต่ออุปกรณ์ถุงมืออิเล็กทรอนิกส์นั้นคือการเชื่อมส่วนที่เป็นถุงมือและเซ็นเซอร์ต่อเข้ากับส่วน CGIU ผ่านทางพอร์ทรับสัญญาณของตัว CGIU และทำการเชื่อมต่อสายสัญญาณจากพอร์ทส่งสัญญาณจากส่วน CGIU เข้าไปที่พอร์ทอนุกรมของเครื่องคอมพิวเตอร์ดังรูป

รูปที่ 2.2 การเชื่อมต่ออุปกรณ์ในชุดอุปกรณ์ถุงมืออิเล็กทรอนิกส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้สายเคเบิ้ลที่มีตัวเชื่อมต่อแบบ 44 พินในการเชื่อมต่อระหว่างถุงมือและเซ็นเซอร์กับส่วนประมวลผล CGIUและใช้สายเคเบิ้ลที่มีตัวเชื่อมต่อแบบอนุกรม CBL-SER110 เพื่อเชื่อมต่อกับพอร์ทอนุกรมของเครื่องคอมพิวเตอร์ ส่วนประมวลผล CGIU จะต้องมีการเชื่อมต่อกับแหล่งจ่ายไฟโดยใช้ตัวเชื่อมต่อแบบ DIN 5 พิน หลังจากติดตั้งและเชื่อมต่อตัวอุปกรณ์ครบแล้วจะมีการปรับแต่งค่าเริ่มต้นที่ส่วน CGIU ที่มีลักษณะเป็นดิปสวิตช์ โดยการปรับแต่งที่ว่านี้เป็นการปรับแต่งค่า baudrate ของตัว CGIU กับพอร์ทอนุกรมของเครื่องคอมพิวเตอร์ สำหรับการทดลองเราจะปรับค่าเริ่มต้นไว้ที่ 38,400

ต่อไปทำการติดตั้งซอฟท์แวร์ VirtualHand ซึ่งมาพร้อมกับตัวชุดอุปกรณ์ เป็นซอฟท์แวร์ไลบรารี่ที่สามารถทำงานได้กับ OpenGL

2.1.4 หลักการทำงานของตัวอุปกรณ์

การทำงานของชุดอุปกรณ์ถุงมืออิเล็กทรอนิกส์เป็นการทำงานโดยที่ภายในตัวถุงมือจะบรรจุเซ็นเซอร์ต่างๆไว้จำนวนหนึ่งเพื่อทำหน้าที่จับการเคลื่อนไหวของนิ้วมือ โดยที่เซ็นเซอร์นั้นจะมีลักษณ์เป็นแผ่นรูปสี่เหลี่ยมผืนผ้ามีความยืดหยุ่นสามารถโค้งงอไปตามลักษณะของนิ้วมือและข้อมือได้ ดังนั้นเมื่อเกิดการเคลื่อนไหวต่างๆ เซ็นเซอร์ตามตำแหน่งที่อยู่บนถึงมือก็จะทำการส่งข้อมูลที่ได้จากการตรวจจับการเคลื่อนไหวของนิ้วมือข้อนิ้วที่ตำแหน่งต่างๆกันไปที่ส่วนประมวลผล CGIU เพื่อทำการแปลงข้อมูลที่อยู่ในแบบอนาลอกไปเป็นข้อมูลแบบดิจิตอลแล้วส่งต่อให้คอมพิวเตอร์เพื่อนำไปประมวลผลในรูปแบบต่างๆ

2.1.5 คำสั่งพื้นฐานในการใช้งาน CGIU

ใช้งานโดยส่วนใหญ่จะเป็นการร้องขอข้อมูลจากตัวเซ็นเซอร์ซึ่งข้อมูลที่ได้จากเซ็นเซอร์นั้นจะมีอยู่ 2 รูปแบบ คือ

- แบบอักขระแอสกี้ (Ascii)
- แบบไบนารี่ (Binary)

ซึ่งการป้อนคำสั่งส่วนใหญ่ถ้าป้อนโดยที่ไม่มีอักขระ ? นำหน้าจะเป็นการป้อนคำสั่งควบคุมแต่ถ้ามีการใส่อักขระ ? นำหน้าคำสั่งนั้นก็จะเป็นคำสั่งที่ใช้เรียกดูข้อมูลหรือเป็นคำสั่งที่ทำการตรวจสอบสถานะของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

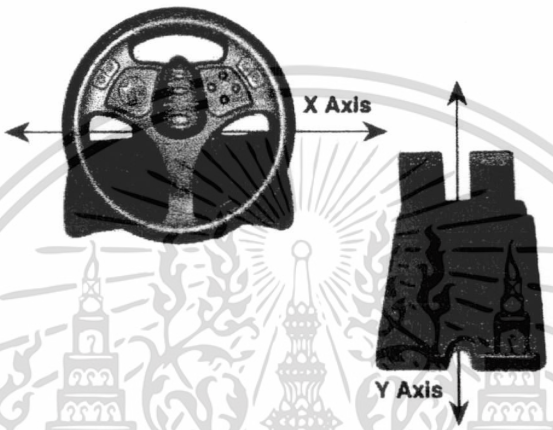
ตัวอย่างของคำสั่งพื้นฐาน

- การร้องขอข้อมูลจากเซ็นเซอร์
  - รูปแบบคำสั่ง 'G' หรือ 'S' เป็นการร้องขอข้อมูลที่อยู่ในรูปแบบไบนารี่
    - 'G' เป็นการร้องขอข้อมูลแบบทีละเร็คคอร์ด
    - 'S' เป็นการร้องขอข้อมูลเป็นลักษณะของสตรีม
  - รูปแบบคำสั่ง 'g' หรือ 's' เป็นการร้องขอข้อมูลที่อยู่ในรูปแบบอักขระแอสกี้
    - 'g' เป็นการร้องขอข้อมูลแบบทีละเร็คคอร์ด
    - 's' เป็นการร้องขอข้อมูลเป็นลักษณะของสตรีม
  - หมายเหตุ การร้องขอข้อมูลในรูปแบบของไบนารี่จะมีความเร็วกว่าแบบอักขระแอสกี้ และลักษณะการส่งคำสั่งควบคุมแบบ 'S' และ 's' จะหยุดการรับข้อมูลเมื่อมีการส่งคำสั่ง '^c'
- การร้องขอข้อมูลของ CGIU และ Glove
  - รูปแบบคำสั่ง '?i' เป็นการร้องขอข้อมูลรายละเอียดต่างๆของ CGIU และ CyberGlove
- การตรวจสอบสถานะและการเชื่อมต่อของ CyberGlove
  - รูปแบบคำสั่ง '?G' เป็นการร้องขอข้อมูลแบบไบนารี่
  - รูปแบบคำสั่ง '?g' เป็นการร้องขอข้อมูลแบบอักขระแอสกี้
- การร้องขอข้อมูลจำนวนของเซ็นเซอร์
  - รูปแบบคำสั่ง '?N' เป็นการร้องขอข้อมูลจำนวนเซ็นเซอร์แบบไบนารี่
  - รูปแบบคำสั่ง '?n' เป็นการร้องขอข้อมูลจำนวนเซ็นเซอร์แบบอักขระแอสกี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

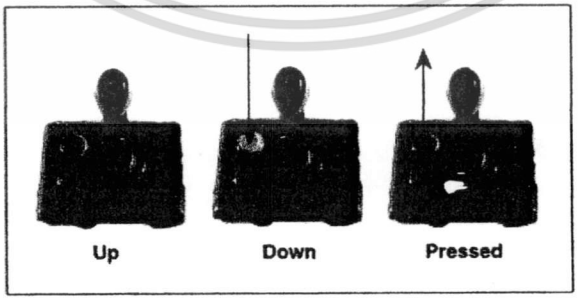
2.2 พวงมาลัยอิเล็กทรอนิกส์

หลักการทำงานของ จอยสติ๊ก โดยลักษณะคล้ายกับการทำงานของเมาส์ โดยจะรับและแสดงการเคลื่อนที่ในทิศทางของ แกน XและY ซึ่งจะรับจากคนละชิ้นส่วนกัน แตกต่างจากการรับค่าของเมาส์ ซึ่งจะรับค่าทิศทางของ X และ Y จากอุปกรณ์ชิ้นเดียวกัน



รูปที่ 2.3 ลักษณะของการรับค่าแสดงทิศทางของอุปกรณ์ประเภทจอยสติ๊กที่เป็นพวงมาลัยควบคุม

จอยสติ๊กจะมีปุ่มคล้ายกับคีย์บอร์ดและเมาส์ โดยจอยสติ๊กส่วนใหญ่นั้นจะมีปุ่มอยู่ระหว่าง 2-10 ปุ่ม ซึ่งจอยสติ๊กนั้นจะแตกต่างจากคีย์บอร์ดและเมาส์ตรงที่มีเหตุการณ์อยู่ 3 เหตุกาณณ์ที่เกิดขึ้นได้คือ UP, DOWN, PRESSED

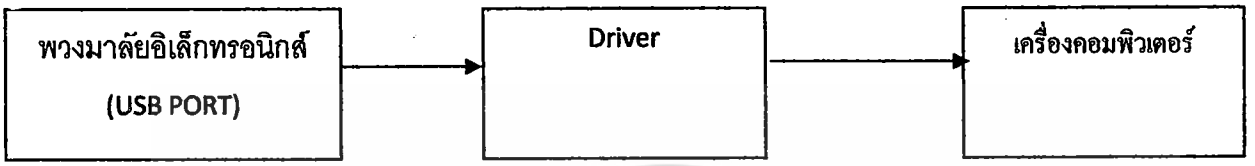


รูปที่ 2.4 ลักษณะเหตุการณ์ของปุ่มบนจอยสติ๊กทั้งสามเหตุการณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 การเชื่อมต่ออุปกรณ์

การเชื่อมต่ออุปกรณ์ พวงมาลัยอิเล็กทรอนิกส์ นั้นโดยจะทำการเชื่อมต่อผ่าน USB port ซึ่งจะต้องทำการลงไดร์เวอร์ของชุดอุปกรณ์ก่อนเครื่องคอมพิวเตอร์จึงจะสามารถมองเห็นได้



รูปที่ 2.5 ลักษณะของการเชื่อมต่อชุดอุปกรณ์เข้ากับเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การเขียนโปรแกรมติดต่อระหว่าง
โปรแกรม Blenderกับอุปกรณ์รับข้อมูลเข้าประเภทถุงมือและพวงมาลัย

การใช้ติดต่อกับอุปกรณ์ต่างๆ จะใช้ภาษาคอมพิวเตอร์ Python ในการติดต่อเนื่องจากมีความง่ายในการเข้าใจ พร้อมทั้งโปรแกรม Blender มีการเตรียมสภาพแวดล้อมในการเขียนภาษาคอมพิวเตอร์ Python ไว้ให้แล้ว ดังนั้น ผู้ใช้งานสามารถเขียนภาษา Python ในรูปแบบของโมดูลและอิมพอร์ทเข้าไปใช้งานในโปรแกรม Blender ได้ทันที

3.1 ถุงมืออิเล็กทรอนิกส์

การติดต่อกับอุปกรณ์ CyberGlove จะใช้งาน Blender ในส่วนของ Script Windows > System > Interactive Python Console โดยได้มีการพัฒนาโมดูลที่ทำการติดต่อกับ CyberGlove ผ่านทางโมดูล pySerial และการควบคุมวัตถุจะใช้โมดูล Blender วิธีในการติดต่อกับ CyberGlove จากโปรแกรม Blender สามารถทำได้ 2 วิธี

1. ติดต่อโดยตรงจาก Interactive Console
2. ใช้งาน Api สำหรับควบคุม CyberGlove ผ่านทางพอร์ทอนุกรม

3.1.1 การติดต่อโดยตรงจาก Interactive Console

ผู้ใช้สามารถทำงานติดต่อผ่าน Interactive Console โดยทำการสร้าง Panel ใหม่ขึ้นมาและเลือกที่ Script Windows > System > Interactive Console

ให้ผู้ใช้งานทำการอิมพอร์ทโมดูล pySerial และ Blender เข้ามาใน Interactive Console

```
>>>import serial
>>>import Blender
```

และทำการสร้างออปเจ็คของโมดูล serial และกำหนดค่าเริ่มต้น 3 ค่าคือ

- port = 'ชื่อช่องทางสื่อสารกับพอร์ทอนุกรม' เครื่องคอมพิวเตอร์ที่ใช้ในการทดลองใช้ COM3
- baudrate = 38400
- timeout = 0

```
>>>ser = serial.Serial(port='COM3',baudrate=38400,timeout=0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการสร้างออปเจ็คของโมดูล serial แล้วให้ทำการเปิดการเชื่อมต่อกับพอร์ทโดยคำสั่ง

```
>>>ser.open()
```

จากนั้นผู้ใช้งานสามารถส่งคำสั่งที่ใช้เพื่อขอข้อมูลหรือสอบถามข้อมูลต่างๆจาก CyberGlove ได้โดยใช้คำสั่ง

```
>>>ser.write('คำสั่งที่ต้องการส่งให้ CyberGlove')
```

หลังจากการใช้งานให้ทำการปิดการเชื่อมต่อและทำลายออปเจ็คของโมดูล serial โดย

```
>>>ser.close()
>>>del ser
```

3.1.2 การใช้งาน Api สำหรับควบคุม CyberGlove ผ่านทางพอร์ทอนุกรม

เปิดการใช้งาน Interactive Console ให้ทำการอิมพอร์ทโมดูล glove ซึ่งเป็น Api ที่เขียนขึ้นเพื่อติดต่อ CyberGloveโดยการใช้คำสั่ง

```
>>>import glove
```

ถ้ามีการฟ้องจาก Interactive Console ว่าไม่เจอโมดูลนี้แสดงว่าไม่ได้มีการเพิ่ม path ของโมดูลเข้าไปให้ Interactive Console รู้จัก เราสามารถเพิ่ม path เข้าไปได้โดยใช้โมดูล sys ซึ่งเป็นโมดูล built-in ของภาษา Python

```
>>>import sys
>>>sys.path.append('path ที่เก็บโมดูล glove')    #ex sys.path.append('c:\\project\\module')
```

หลังจากนั้นให้ทำการสร้างออปเจ็คจากโมดูล glove

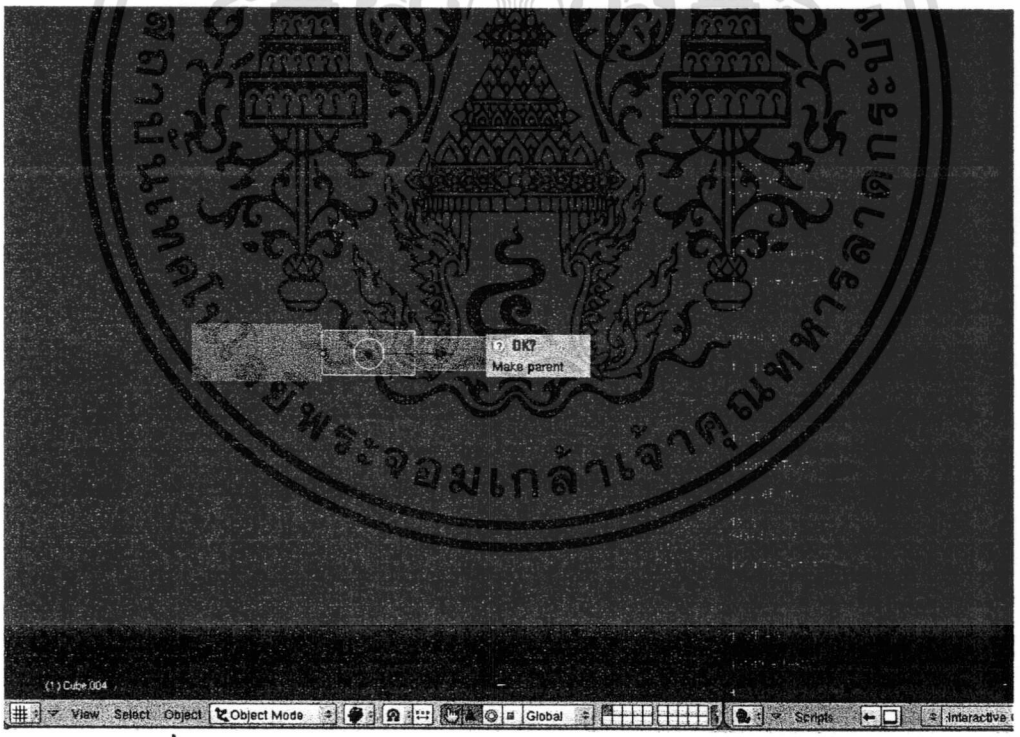
```
>>>ser = glove.Glove()
```

เอกสารนี้ใช้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเป็นการสร้างออปเจ็ค ser โดยมีค่าเริ่มต้นเป็นดังนี้

- port = 'COM3'
- baudrate = 38400
- timeout = 0
- blenderObjName='Cube'
- intime=0.26

โดยที่ค่า blenderObjName เป็นค่าเริ่มต้นของวัตถุแรกของโปรแกรม Blender ชื่อ Cube และ intime เป็นค่าดีเลย์ของโปรแกรมที่ติดต่อกับ CyberGlove เนื่องจากการทดลองนี้มีการใช้วัตถุที่เป็น Mesh จำนวน 3 ชิ้น จึงต้องทำการสร้างวัตถุชนิด Mesh ในลักษณะของ Cube ขึ้นมาอีก 2 ชิ้น ตัวโปรแกรม Blender จะทำการกำหนดชื่อให้เป็น Cube.001, Cube.002 และทำการย้ายจุดหมุนของแต่ละวัตถุไปที่ปลายอีกด้านหนึ่งของตัววัตถุเองหลังจากนั้นให้ทำการกำหนดความสัมพันธ์ระหว่างวัตถุทั้ง 3 เป็นลักษณะ parent ตามลำดับดังรูป



รูปที่ 3.1 แสดงการย้ายจุดหมุนและการกำหนดความสัมพันธ์ของวัตถุ

หลังจากการสร้างออปเจ็คของโมดูล glove แล้วให้ทำการเปิดการติดต่อกับพอร์ทโดยใช้งานคำสั่ง

```
>>>ser.openPort()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ใช้งานสามารถส่งคำสั่งเพื่อที่จะรับข้อมูลเกี่ยวกับ CGIU และ CyberGlove ได้โดยใช้คำสั่ง

```
>>>ser.getGloveInformation()
```

ก่อนเริ่มการรับข้อมูลจาก CyberGlove เราสามารถเปลี่ยนค่าดีเลย์ของโปรแกรมที่ทำการติดต่อกับ CyberGlove ได้โดยใช้คำสั่ง

```
>>>ser.setDelay(ค่าตัวเลขเป็นวินาที)  # ค่าเริ่มต้นเท่ากับ 0.26 วินาที สามารใช้ค่า 0.1 วินาทีได้
```

หลังจากนั้นเราสามารถเรียกดูค่าดีเลย์ในการรับข้อมูลจาก CyberGlove ณ ปัจจุบันได้โดยคำสั่ง

```
>>>ser.getDelay()
```

เนื่องจากมีการรับข้อมูลจาก CyberGlove นั้นสามารถรับข้อมูลได้ 2 รูปแบบคือ

- แบบไบนารี่ (Binary)
- แบบอักขระแอสกี้ (Ascii)

และมีการส่งข้อมูลจาก CyberGlove มาที่เครื่องคอมพิวเตอร์อยู่ 2 แบบคือ

- แบบทีละเร็คคอร์ด
- แบบสตรีม

เพื่อการเริ่มการส่งข้อมูลจาก CyberGlove แบบไบนารี่ทีละเร็คคอร์ดโดยคำสั่ง

```
>>>ser.startGetSenserValue()
```

คำสั่งด้านบนจะมีค่าเทียบเท่ากับการส่งคำสั่งโดยไม่ผ่าน glove Api คือ 'G' โมดูล glove ก็จะมีการส่งข้อมูลการขยับข้อของนิ้วตามตำแหน่งต่างๆ ทำให้เกิดการเคลื่อนของวัตถุที่ชื่อ Cube ในโลกเสมือนในโปรแกรม Blender เมื่อเราต้องการหยุดการรับข้อมูลให้ใช้คำสั่ง

```
>>>ser.stopGetSensorValue()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

คำสั่งด้านบนจะมีค่าเทียบเท่ากับการส่งคำสั่งโดยไม่ผ่าน glove Api คือ '^c' ทั้งนี้ผู้ใช้งานสามารถทำการเปลี่ยนค่าของการทำงานของ loop ในโปรแกรมขณะทำงานโดยใช้คำสั่ง

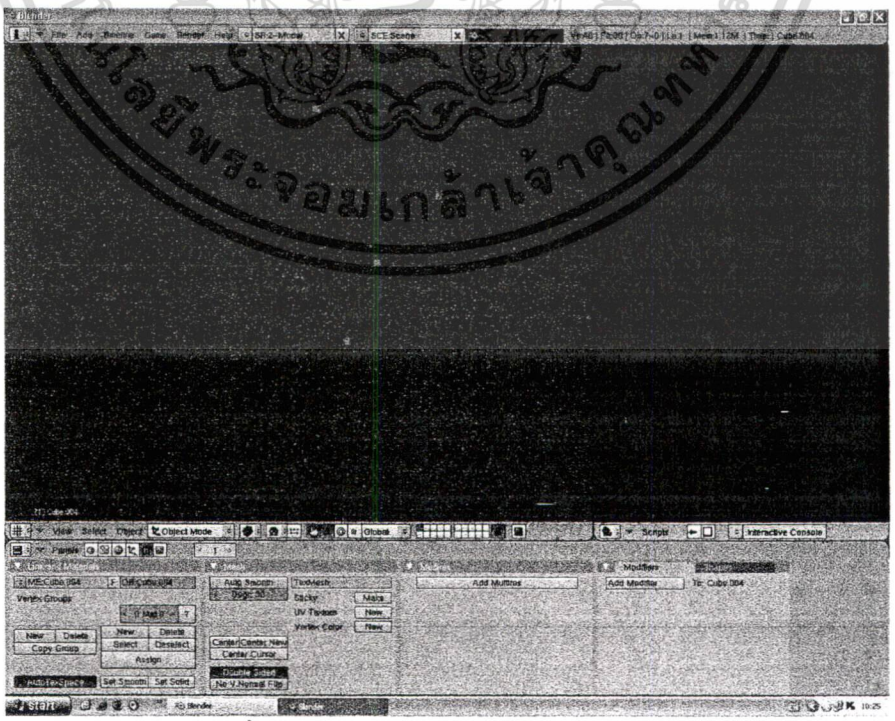
```
>>>ser.setLoopCount(จำนวน loop ที่ต้องการ)
```

จำนวน loop เริ่มต้นจะเป็น 1000 รอบ ถ้าต้องการเรียกดูจำนวน loop ที่ตั้งไว้ในโปรแกรมทำได้โดยการใช้คำสั่ง

```
>>>ser.getLoopCount()
```

สรุปลักษณะการใช้งานชุดคำสั่ง Api glove โดยทั่วไปจะเป็นขั้นตอนดังนี้

1. สร้างออปเจ็ค glove โดยคำสั่ง ser = glove.Glove()
2. เปิดพอร์ททำการเชื่อมต่อรับส่งข้อมูลโดยคำสั่ง ser.openPort()
3. รับข้อมูลมาใช้งานในโปรแกรม Blender โดยคำสั่ง ser.startGetSensorValue()
4. ยกเลิกการรับข้อมูลจากพอร์ทที่ทำการเชื่อมต่อโดยคำสั่ง set.stopGetSensorValue()
5. ปิดพอร์ทการเชื่อมต่อโดยคำสั่ง ser.closePort()
6. ทำลายออปเจ็ค glove โดยคำสั่ง del ser



รูปที่ 3.2 แสดงผลของการใช้งาน glove api

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

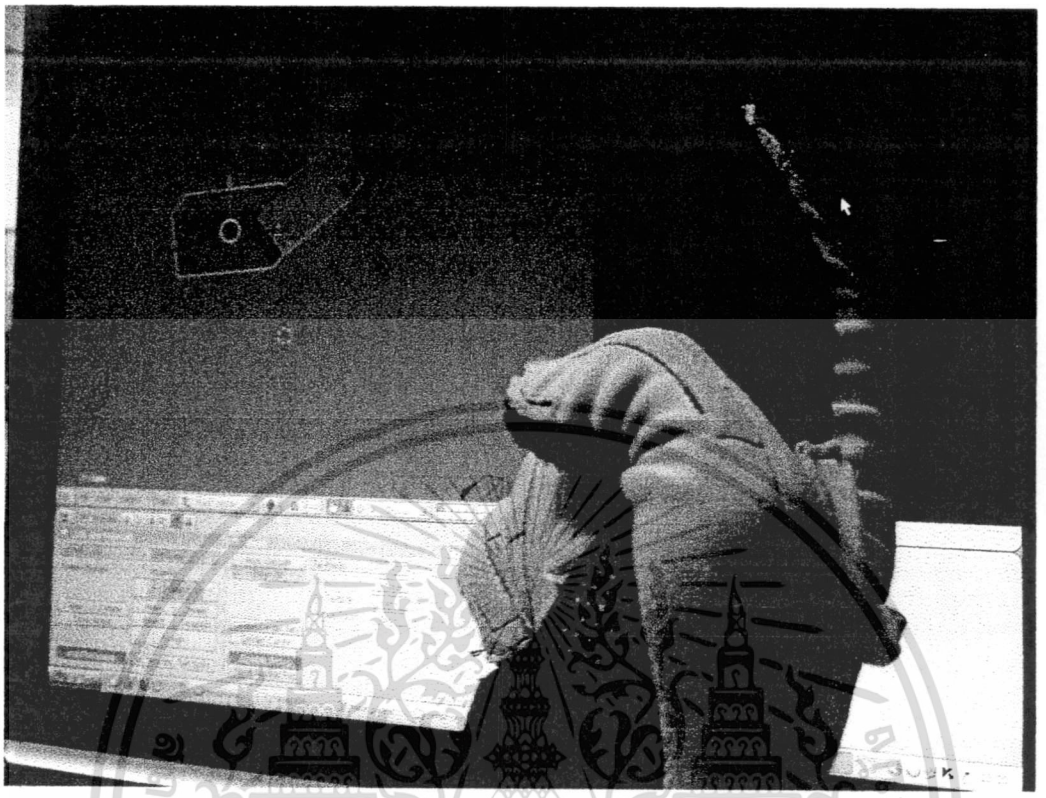
120283

รูปที่ 3.2 ที่แสดงอยู่ด้านบนเป็นรูปที่แสดงผลของการทดลองโดยรับค่าตำแหน่งต่างๆของนิ้วมือและทำการวาดวัตถุ 5 ชิ้น โดยมีการเคลื่อนไหวต่างตามการเคลื่อนไหวของนิ้วมือเพื่อทดลองการรับข้อมูลผ่านจากพอร์ทอนุกรม หลังจากนั้นได้ทำการสร้างวัตถุ 3 ชิ้นเพื่อจำลองลักษณะเป็นข้อนิ้วชี้เพื่อทดลองการเคลื่อนไหวของข้อนิ้ว

รูปที่ 3.3 แสดงลักษณะของนิ้วมือที่ได้รับข้อมูลมาจากถุงมือ

รูปที่ 3.3 เป็นรูปที่แสดงการเคลื่อนไหวของนิ้วมือที่ได้รับข้อมูลจาก CyberGlove แล้วมาจับคู่การเคลื่อนไหวกับวัตถุในโปรแกรม Blender

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 แสดงการทดลองการเคลื่อนไหวของนิ้วชี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 พวงมาลัยอิเล็กทรอนิกส์

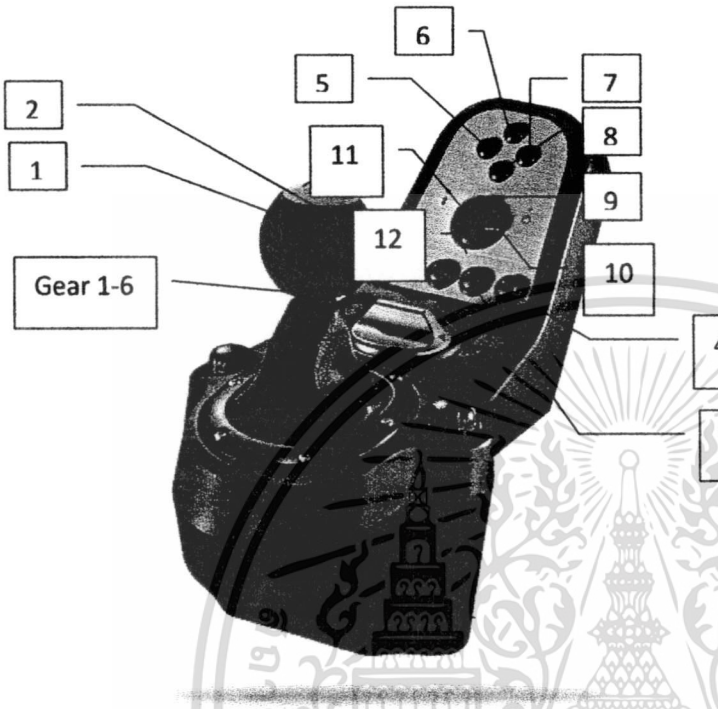
โปรแกรมเบลนเดอร์มีส่วนช่วยรองรับการเชื่อมต่อของอุปกรณ์ประเภทจอยสติ๊กอยู่ในส่วนของเกมเอนจิ้น โดยที่สามารถกำหนดเหตุการณ์ต่างๆในเกมจากส่วนของ Sensor ซึ่งต้องเลือกประเภทของ Sensor ให้เป็น joystick

Sensor ประเภท joystick ยังสามารถเลือกชนิดของ ส่วนที่รับข้อมูลเข้าได้อีกด้วย ซึ่งแบ่งออกเป็น สามชนิด ดังนี้ Hat, Axis และ Button จากการศึกษา อุปกรณ์ จอยสติ๊ก Logitech G25 Racing Wheel เมื่อใช้ร่วมกับ Blender game engine ปุ่มต่างๆ รวมไปถึง เกียร์ Blender game engine จะมองเห็นเป็น sensor joystick ประเภท button ส่วน พวงมลัย, เบรค, คันเร่ง game engine จะมองเห็นเป็น sensor joystick ประเภท Axis และมีส่วนของปุ่มบางปุ่มที่ Blender game engine มองเห็นเป็น Axis คือปุ่ม ลูกศร ขึ้นลงและซ้ายขวา โดยปุ่มนี้จะเป็นการ map ที่เหมือนกับการขยับพวงมลัย เบรคและคันเร่ง

รูปที่ 3.5 Logitech G25 Racing Wheel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

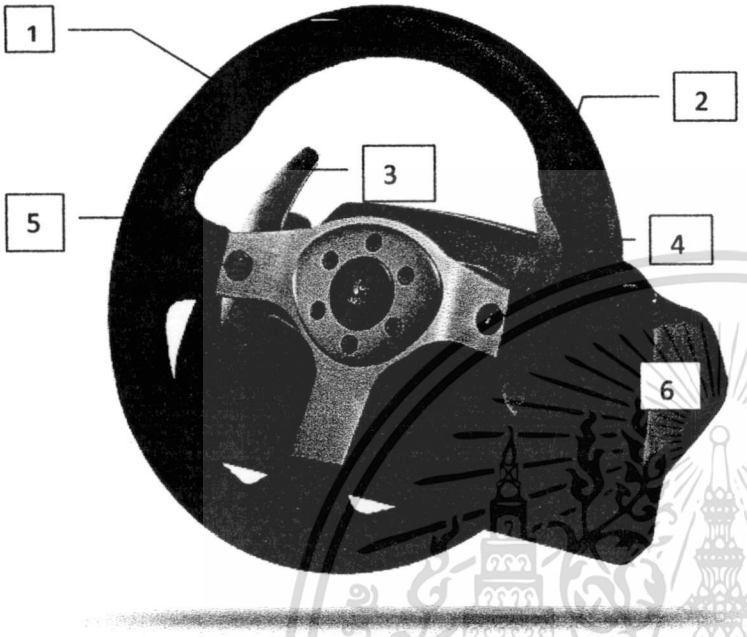
โดยมีการระบุหมายเลขสำหรับการเข้าถึงแต่ละปุ่มของ Steering Wheel ดังรายการต่อไปนี้



รูปที่ 3.6 แสดงตำแหน่งปุ่มต่างๆของ Logitech G25 Racing Wheel

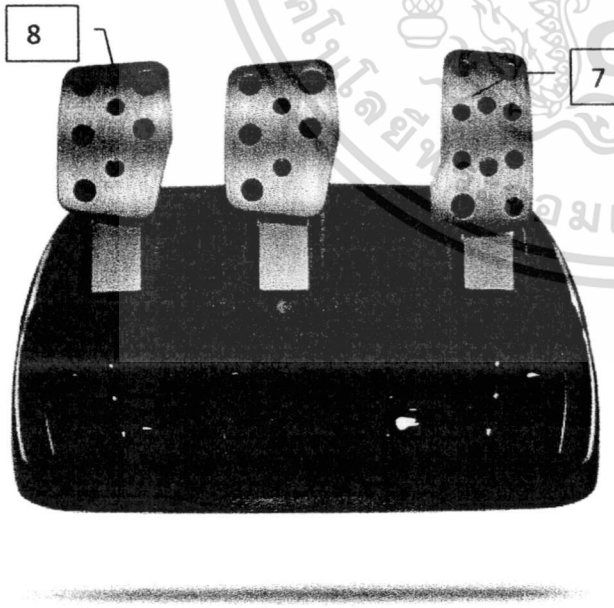
ลำดับที่	จอยสติ๊กประเภท	หมายเลข
1	Button	0
2	Button	1
3	Button	2
4	Button	3
5	Button	16
6	Button	15
7	Button	18
8	Button	17
9	HAT	Direction1
10	HAT	Direction2
11	HAT	Direction8
12	HAT	Direction4
Gear1	Button	8
Gear2	Button	9
Gear3	Button	10
Gear4	Button	11
Gear5	Button	12
Gear6	Button	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ลำดับที่	จอยสติ๊กประเภท	หมายเลข
1	Axis	LeftAxis
2	Axis	RightAxis
3	Button	5
4	Button	4
5	Button	7
6	Button	6
7	Axis	DownAxis
8	Axis	UpAxis

รูปที่3.7 แสดงตำแหน่งปุ่มต่างๆของ Logitech G25 Racing Wheel



รูปที่3.8 แสดงตำแหน่งปุ่มต่างๆของ Logitech G25 Racing Wheel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. 2.1 ตัวอย่างการเขียนคำสั่งควบคุม steering wheel ผ่าน Blender Game Engine ด้วย โมดูล GameLogic

```
import GameLogic
```

ให้ผู้ใช้งานทำการอิมพอร์ทโมดูล GameLogic เข้ามา

```
controller=GameLogic.getCurrentController()
```

เรียกใช้ใช้ฟังก์ชั่น getCurrentController()

```
owner=controller.getOwner()
pos=owner.getPosition()
```

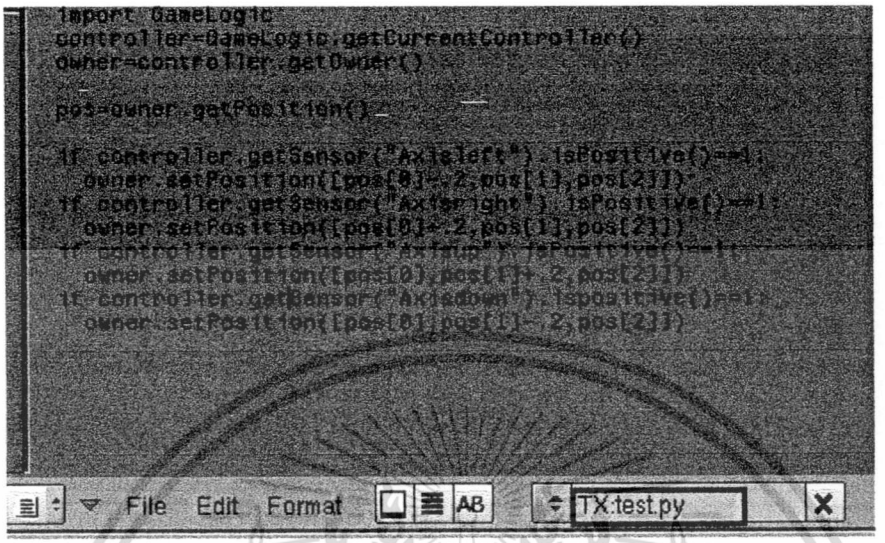
ทำการสร้างออปเจกต์ pos ขึ้นมาเพื่อสำหรับกำหนด ตำแหน่ง

```
if controller.getSensor("Axisleft").isPositive()==1:
        owner.setPosition([pos[0]-.2,pos[1],pos[2]])
if controller.getSensor("Axisright").isPositive()==1:
        owner.setPosition([pos[0]+.2,pos[1],pos[2]])
if controller.getSensor("Axisup").isPositive()==1:
        owner.setPosition([pos[0],pos[1]+.2,pos[2]])
if controller.getSensor("Axisdown").ispositive()==1:
        owner.setPosition([pos[0],pos[1]-2,pos[2]])
```

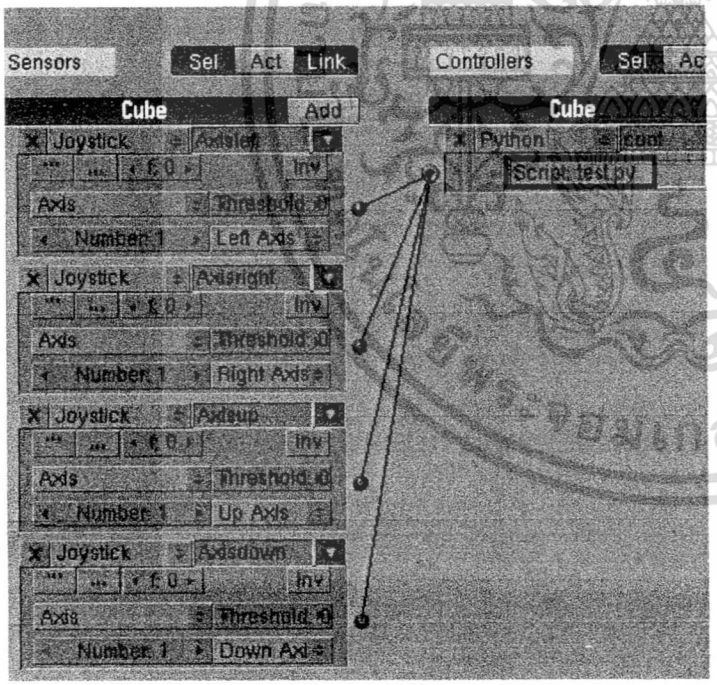
เป็นการกำหนดเงื่อนไขให้ Sensor แต่ละตัวว่าจะ ให้เคลื่อนที่ไปยังทิศทางใด

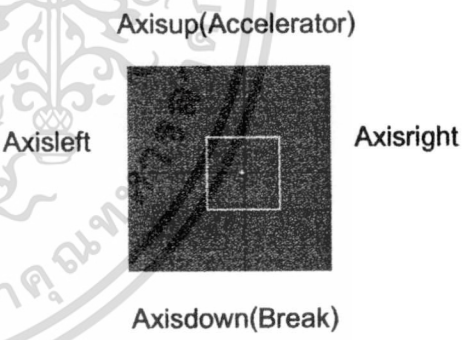
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถเขียนคำสั่งด้วยภาษาไพธอน ด้วยหน้าต่าง Text editor จากตัว Blender Game Engine ได้ดังนี้



รูป 3.9 แสดงการเขียนคำสั่งด้วยภาษาไพธอน ผ่านหน้าต่าง Text editor จากตัว Blender Game Engine





รูป 3.11 แสดงทิศทางการเคลื่อนที่ของวัตถุ

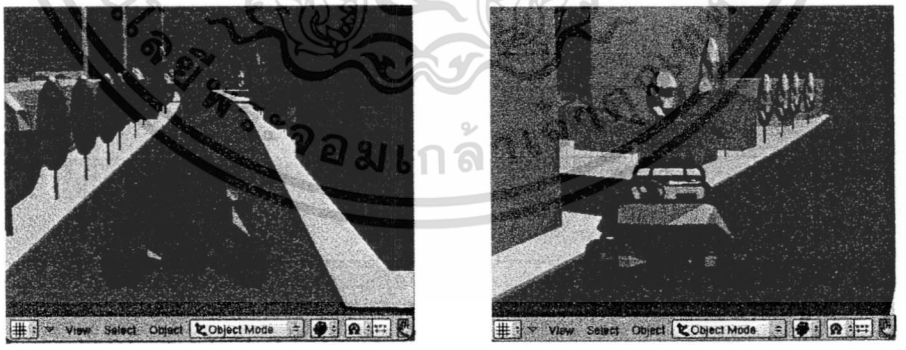
รูป 3.10 แสดงการเรียกใช้ ไพธอนสคริปต์ ในส่วนของ Controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. 2.2 **ตัวอย่างงานที่สร้างจากใช้การควบคุมพวงมาลัยอิเล็กทรอนิกส์ผ่าน** Blender Game Engine **ร่วมกับการเขียนคำสั่งด้วย โมดูล** GameLogic

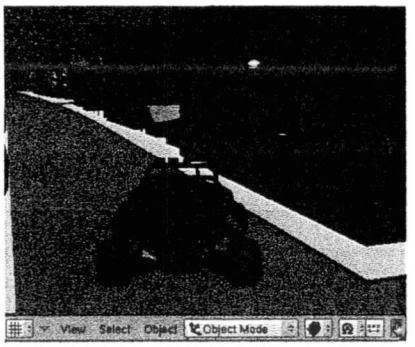
ในตัวอย่างนี้แบ่งการทำงานออกเป็นสองส่วนคือส่วนการเคลื่อนที่ของรถที่ใช้การควบคุมผ่าน UI ของ Blender Game Engine ในที่นี้แบ่งออกเป็น สี่เหตุการณ์คือ

1. เคลื่อนที่ไปข้างหน้า ผ่านการรับข้อมูลของ Sensor "front" โดยรับข้อมูลจาก คันเร่ง ของ steering wheel ที่ถูกกำหนดเป็น joystick ประเภท Axis : Up Axis
2. เคลื่อนที่ไปทางซ้าย ผ่านการรับข้อมูลของ Sensor "left" โดยรับข้อมูลจาก พวงมาลัย ของ steering wheelที่ถูกกำหนดเป็น joystick ประเภท Axis : left Axis
3. เคลื่อนที่ไปทางขวา ผ่านการรับข้อมูลของ Sensor "right" โดยรับข้อมูลจากพวงมาลัย ของ steering wheelที่ถูกกำหนดเป็น joystick ประเภท Axis : right Axis
4. เคลื่อนที่ไปถอยหลัง ผ่านการรับข้อมูลของ Sensor "back" โดยรับข้อมูลจาก เกียร์ถอยหลัง ของ steering wheelที่ถูกกำหนดเป็น joystick ประเภท Button : Number 9



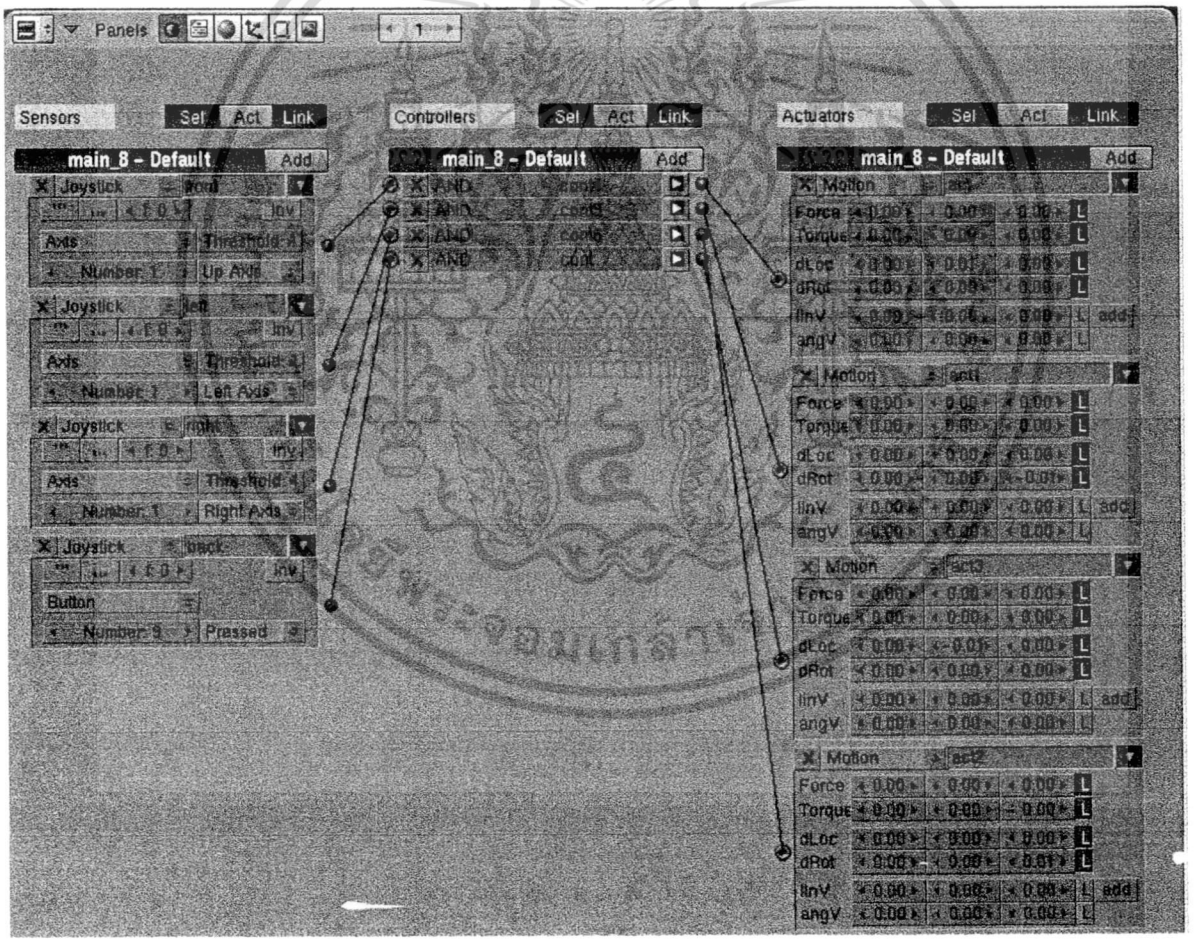
รูปที่ 3.12 แสดงการเคลื่อนที่ไปข้างหน้าของรถ และแสดงการเคลื่อนที่ไปทางซ้ายของรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 แสดงการเคลื่อนที่ไปทางขวาของรถ

โดยมีการกำหนดค่าต่างๆของ Sensor, Controller และ Actuator ตามรูปด้านล่างนี้



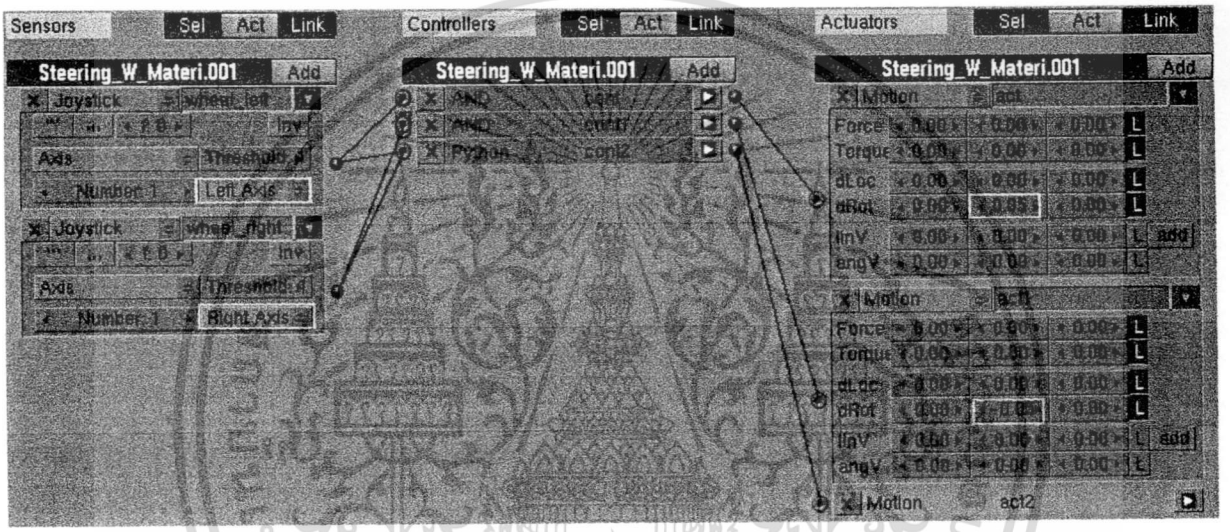
รูปที่ 3.14 แสดงการกำหนดค่าของ Sensor, Controller และ Actuator ใน โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนนี้แสดงการควบคุมการหมุนของพวงมาลัย โดยแบ่งออกเป็น สองเหตุการณ์คือ

1. พวงมาลัยหมุนไปทางซ้าย ผ่านการรับข้อมูลของ Sensor “wheel_left” โดยรับข้อมูลจาก พวงมาลัย ของ steering wheelที่ถูกกำหนดเป็น joystick ประเภท Axis : left Axis

2. พวงมาลัยหมุนไปทางขวา ผ่านการรับข้อมูลของ Sensor “wheel_right” โดยรับข้อมูลจากพวงมาลัย ของ steering wheel ที่ถูกกำหนดเป็น joystick ประเภท Axis : right Axis

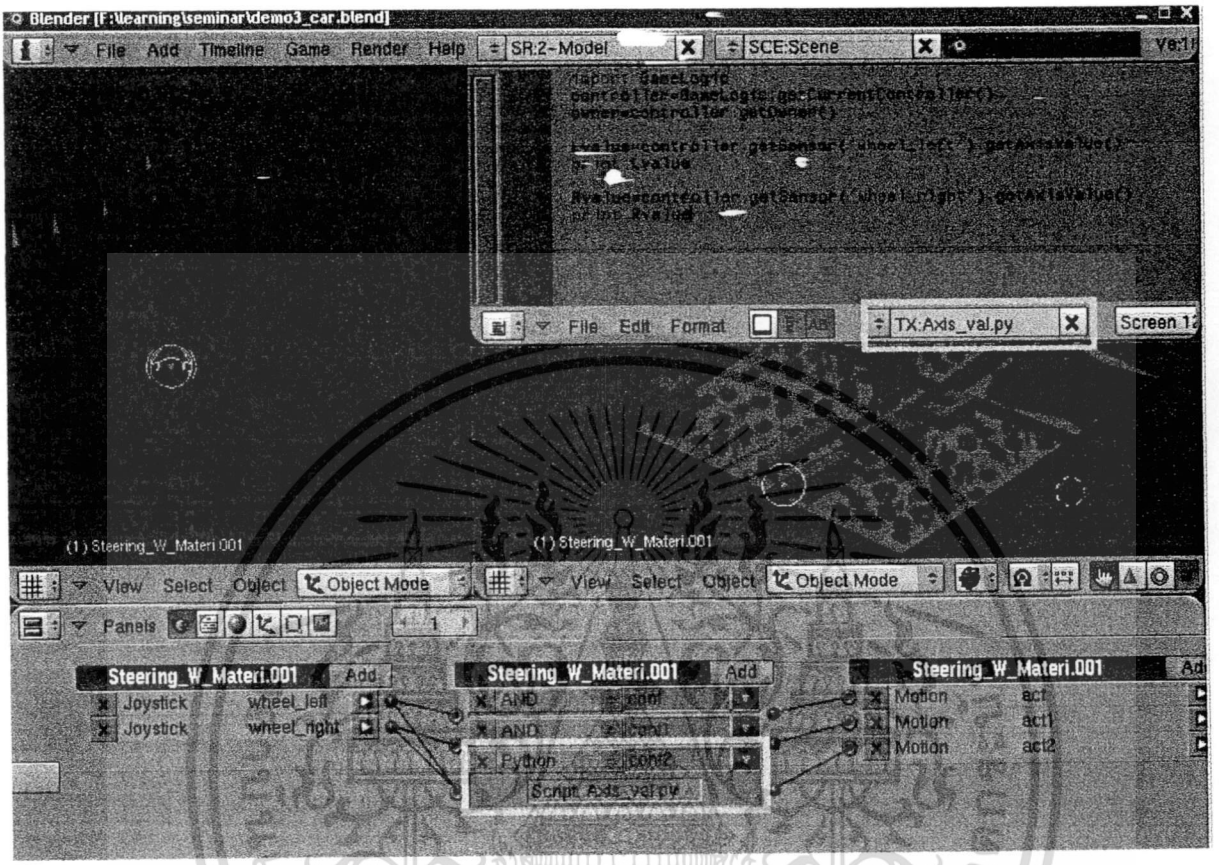


รูปที่ 3.15 แสดงการกำหนดค่าของ Sensor, Controller และ Actuator ใน โปรแกรม

รูปที่ 3.16 แสดงผลการหมุน ซ้ายและ ขวาของพวงมาลัย

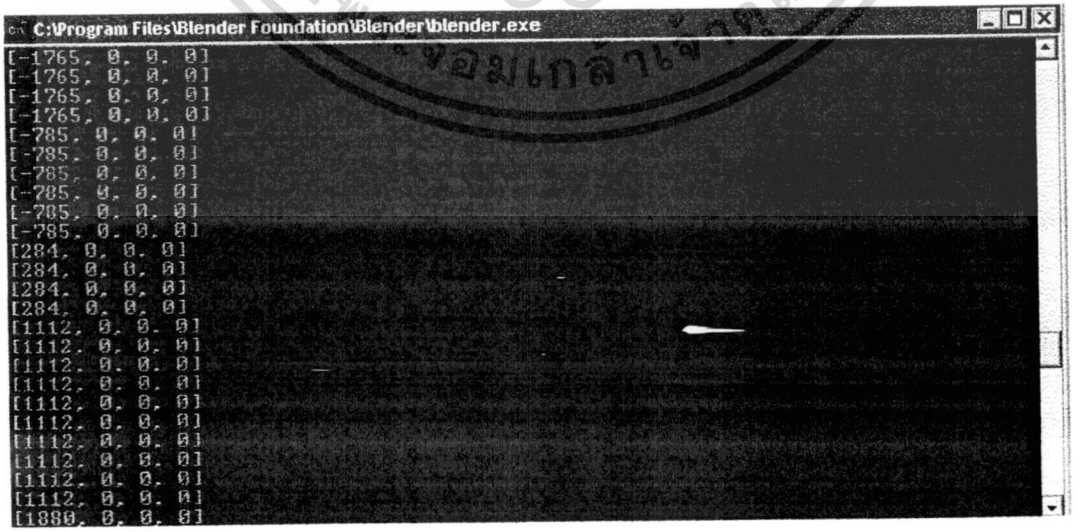
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเราสามารถเขียนคำสั่งให้แสดงค่าที่ได้จากการหมุนพวงมาลัย Steering Wheel ด้วย



รูปที่ 3.17 แสดงการเรียกใช้ไพธอนสคริปต์ในส่วนของ Controller

โดยผลลัพธ์ที่ได้จะแสดงอยู่ในหน้าต่าง Command line ดังรูปต่อไปนี้



รูปที่ 3.18 แสดงผลลัพธ์ที่ได้จากการหมุนพวงมาลัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
สรุปผลการทดลอง

โปรแกรม Blender เป็นโปรแกรมสร้างโมเดลสามมิติและอนิเมชันที่มีประสิทธิภาพ เป็นลักษณะของโปรแกรมที่เป็น opensource ที่เปิดเผย sourcecode ผู้ใช้งานสามารถปรับแต่งได้ตามต้องการ และมีส่วนที่สนับสนุนการเขียน script เพิ่มเติมโดยใช้ภาษา Python ซึ่งเป็นภาษาประเภท dynamic language ที่มีประสิทธิภาพสูง สามารถทำความเข้าใจได้ง่าย มีความยืดหยุ่นสูง ด้วยการร่วมกันใช้งานของโมดูลที่เขียนด้วยภาษา Python หลายตัว เช่น pySerial, DirectPython สามารถทำให้การติดต่อกับอุปกรณ์ถุงมืออิเล็กทรอนิกส์ และ พวงมาลัยอิเล็กทรอนิกส์ เพื่อบังคับวัตถุในโลกเหมือนจริงสามารถทำได้โดยง่าย

4.1 ถุงมืออิเล็กทรอนิกส์

การทดลองนี้ได้ทำการทดลองการเชื่อมต่ออุปกรณ์ถุงมืออิเล็กทรอนิกส์โดยการรับข้อมูลผ่านจากพอร์ทอนุกรมเข้ามาประมวลผลในโปรแกรม Blender โดยตรงโดยไม่ผ่าน Blender Game Engine แต่เนื่องจากตัวภาษา Python เป็นภาษาที่อยู่ในรูปแบบของ Interpreter ที่ทำการแปลคำสั่งเป็นภาษาเครื่องทุกครั้ง การรับข้อมูลแบบเรียลไทม์ที่ส่งมาจากตัวถุงมืออิเล็กทรอนิกส์อาจจะเกิดประเด็นทางด้านความเร็วในการประมวลผลเพื่อควบคุมวัตถุในโปรแกรม Blender ทิศทางในการพัฒนาต่อน่าจะเป็นการปรับแต่งค่าต่างๆของพอร์ทอนุกรมและการเขียนส่วนประมวลผลเพิ่มด้วยภาษา C หรือ C++

4.2 พวงมาลัยอิเล็กทรอนิกส์

การติดต่อกับพวงมาลัยอิเล็กทรอนิกส์ จะเป็นการใช้ UI ของ Blender Game Engine โดยสามารถทำงานกับโปรแกรมBlenderในการควบคุมวัตถุได้ดี

โดยพวงมาลัยอิเล็กทรอนิกส์นั้นมีประเภทของปุ่มต่างๆอยู่ 3 แบบคือ Button, Axis, Hat โดยทำการควบคุมผ่านส่วนของ Sensor ของ Blender Game Engine ผู้ใช้ต้อง เลือกประเภทของ Sensor ให้เป็น Joystick ในการควบคุมพวงมาลัยอิเล็กทรอนิกส์จากส่วนของ Blender Game Engine ยังสามารถใช้ไพธอนสคริปร่วมโดยเรียกใช้ผ่านโมดูล GameLogic เพื่อเพิ่มความสามารถในการปฏิสัมพันธ์ของพวงมาลัยอิเล็กทรอนิกส์ ได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] "Blender User's Documentation". [Online] เข้าถึงได้จาก http://www.blendernation.org,2008

[2] "pySerial User's Documentation". [Online] เข้าถึงได้จาก http://pyserial.wiki.sourceforge.net/pyserial

[3] "pywin32". [Online] เข้าถึงได้จาก https://sourceforge.net/projects/pywin32

[4] "Python Script Documentation". [Online] เข้าถึงได้จาก www.blender.org

[5] "Python Documentation".[Online] เข้าถึงได้จาก www.python.org

[6] "Blender User's Documentation". [Online] เข้าถึงได้จาก:http://en.wikipedia.org/wiki/Game_engine, 2008

[7] "Blender Game engine". [Online] เข้าถึงได้จาก: http://wiki.blender.org/index.php/Game_Engine, 2008

[8] "JoystickSensor" . [Online] เข้าถึงได้จาก:
http://www.tutorialsforblender3d.com/GameModule/ClassSCA_JoystickSensor.html,2008

[10]James Kramer, Mark Yim, Larry Edwards, Allen Boronkay, and Jerry Tian ,VirtualHand Software Library

[11]Reference Manual, Virtual Technologies, 1998

[12]CyberGlove Reference Manual Document Number CG081998-2-1

[13]Peter Walsh, Advanced 3D Game Programming with DirectX 9.0, Peter Walsh, 2003

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้