สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การประยุกต์ใชเทคโนโลยีเซนเซอร์สามสแกนกับการออกกำลังกาย

Application 3D for exercise

ผู้ช่วยศาสตราจารย์บุณย์ชนะ ภู่ระหงษ์

ได้รับทุนสนับสนุนงานวิจัยจากเงินรายได้ ประจำปีงบประมาณ ๒๕๕๔

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

RCH
TR
897.7
บ124ก

เลขหมู่
เลขทะเบียน 131181
วัน,เดือน,ปี 22 พ.ค. 2557

b 12601974
i

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานวิจัย

Application 3D for exercise

ผู้ช่วยศาสตราจารย์บุณย์ชนะ ภู่ระหงษ์

งานวิจัย ประจำปีงบประมาณ ๒๕๕๔

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
บทคัดย่อ		ก
กิตติกรรมประกาศ		ข
สารบัญ		ค
สารบัญรูป		จ
สารบัญตาราง		ช
บทที่ 1 บทนำ		
1.1	ความเป็นมาและความสำคัญของการวิจัย	1
1.2	วัตถุประสงค์ของการวิจัย	1
1.3	ขอบเขตของการวิจัย	2
1.4	ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย	2
1.5	เนื้อหาภายในการวิจัย	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง		
2.1	ทฤษฎีไมโครคอนโทรลเลอร์	3
2.2	ทฤษฎีไมโครคอนโทรลเลอร์ตระกูล AVR รุ่น ATmega 328P	6
2.3	ทฤษฎีการสื่อสารแบบอนุกรม	9
2.4	หลักการพื้นฐานของ USB	11
2.5	ทฤษฎีการสื่อสารแบบ I2C Bus	12
2.6	Accelerometer ADXL345	16
2.7	หลักการพื้นฐานของไดนาโม	18
2.8	วงจรไฟฟ้าอนุกรม	20
2.9	โปรแกรมออกแบบกราฟิก Blender	20
2.10	โปรแกรมภาษา Python	22
บทที่ 3 การออกแบบและการสร้าง		
3.1	หลักการทำงานของระบบโดยรวม	27
3.2	ส่วนของชุดอุปกรณ์	28
3.3	ส่วนของโปรแกรมกราฟิกสามมิติ	35
บทที่ 4 การทดลองและผลการทดลอง		
4.1	การทำงานของอุปกรณ์โดยรวม	40
4.2	การทดสอบการทำงานของบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove	43
4.3	การแสดงกราฟข้อมูลจากเซนเซอร์วัดความเร่ง ADXL345	48
4.4	การทดลองใช้งานโปรแกรมภาษา Python สำหรับการรับค่าข้อมูลจากบอร์ด	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	ไมโครคอนโทรลเลอร์ Arduino เพื่อนำเข้าเก็บไว้ในรูปแบบไฟล์ .text	
4.6	การทดลองนำค่าที่ส่งเข้ามาจากไมโครคอนโทรลเลอร์ Arduino ไปประมวลผลในโปรแกรมกราฟิกสามมิติ	55
บทที่ 5	สรุปและอภิปรายผลการวิจัย	49
5.1	สรุปผลการวิจัย	49
5.2	อภิปรายผลการวิจัย	49
5.3	แนวทางการพัฒนาต่อ	50
บรรณานุกรม		61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	ชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino	4
2.2	สถาปัตยกรรมชั้นสูงแบบ RISC	7
2.3	ไอซี ATmega 328P	8
2.4	การส่งข้อมูลแบบอนุกรม	10
2.5	บิตต่างๆของการรับส่งข้อมูลแบบอนุกรม	10
2.6	ชนิดของหัว Connector แบบ A และแบบ B	11
2.7	ลักษณะการการเชื่อมต่ออุปกรณ์แบบ I^2C BUS	12
2.8	ลักษณะของ Control Byte ของ I2C Bus	13
2.9	เงื่อนไขของสภาวะเริ่มต้น และสิ้นสุดของ I2C Bus	15
2.10	ช่วงเวลารับส่งบิตข้อมูลของ I2C BUS	15
2.11	เซนเซอร์วัดความเร่ง ADXL345	16
2.12	โครงสร้างของ ADXL345	16
2.14	การทำงานพื้นฐานของไดนาโม	18
2.15	ไดนาโมจักรยาน	20
2.15-1	ระบบกราฟิกที่เปรียบเสมือนกล่องดำ (Black Box)	22
2.16	ขั้นตอนการทำงานของโปรแกรม Python	24
3.1	หลักการทำงานโดยรวมของระบบ	27
3.2	วงจรภายในชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove	29
3.3	การออกแบบการเชื่อมต่อข้อมูลแบบ I2C Bus กับบอร์ดทดลอง Arduino	29
3.4	Flow chat หลักการทำงานของ Arduino ในการส่งข้อมูล	31
3.5	Flow chat หลักการอ่านค่าข้อมูลของเซนเซอร์ ADXL345	33
3.6	Flow chat หลักการเขียนค่าข้อมูลของเซนเซอร์ ADXL345	34
3.7	วงจรไดนาโมจักรยานต่อเข้าสู่บอร์ด Arduino	35
3.8	Flow chat ส่วนของการรับข้อมูลโปรแกรมภาษา Python รับค่าข้อมูลจากไมโครคอนโทรลเลอร์บอร์ด Arduino โดยส่งผ่านทาง Serial Port	37
3.9	Flow chat ส่วนของการแยกค่าข้อมูลที่ส่งมาไว้ในไฟล์ .text	38
3.10	Flow chat ส่วนของการออกแบบรูปทรงสามมิติ	39
4.1	ส่วนของอุปกรณ์	40
4.2	ADXL345 พร้อมกล่องสำหรับติดตั้งบนศีรษะ	41
4.3	บอร์ดไมโครคอนโทรลเลอร์ Arduino	41
4.4	ไดนาโมจักรยานและวงจรกรองแรงดัน	42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

4.5	หน้าโปรแกรมเกมสามมิติที่แสดงออกมาทางหน้าจอแสดงผล	42
4.6	หน้าต่างของโปรแกรม Arduino	43
4.7	การเลือกไมโครคอนโทรลเลอร์ของ Arduino	44
4.8	การเลือกพอร์ตอนุกรมที่ใช้ติดต่อไมโครคอนโทรลเลอร์ของ Arduino	44
4.9	หน้าต่างของโปรแกรม Arduino และโปรแกรมที่เขียน	45
4.10	การเลือกเมนูคำสั่งเพื่อทำการคอมไพล์โปรแกรม	45
4.11	แถบแสดงสถานการณ์คอมไพล์	46
4.12	หน้าต่างของโปรแกรมเมื่อทำการอัพโหลดเสร็จสมบูรณ์	46
4.13	หน้าต่างของ Serial Monitor แสดงค่าข้อมูลของเซนเซอร์ ADXL345และไดนาโมจักรยาน	47
4.14	การวางเซนเซอร์ระนาบ Z+	48
4.15	ค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ Z+	48
4.16	การวางเซนเซอร์ระนาบ Z-	49
4.17	ค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ Z-	49
4.18	การวางเซนเซอร์ระนาบ X+	49
4.19	ค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ X+	50
4.20	การวางเซนเซอร์ระนาบ X-	50
4.21	ค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ X-	51
4.22	การวางเซนเซอร์ระนาบ Y+	51
4.23	ค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ Y+	51
4.24	การวางเซนเซอร์ระนาบ Y-	52
4.25	ค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ Y-	52
4.26	เลือกวัตถุในโปรแกรม Blender3D	53
4.27	ย้ายจุดพิกัดเพื่อสร้างโมเดลรูปบ้าน	54
4.28	ตกแต่งโมเดลรูปบ้าน	55
4.29	ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อเอียงศีรษะในระนาบ X-	56
4.30	ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อเอียงศีรษะในระนาบ X+	56
4.31	ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อเอียงศีรษะในระนาบ Y-	57
4.32	ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อเอียงศีรษะในระนาบ Y+	57
4.33	ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อไดนาโมทำงาน	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่

2.1	ตารางแสดงฟังก์ชั่นการทำงานของ ATmega 328P	8
2.2	ตารางรายละเอียดฟังก์ชั่นขาต่างๆของ ADXL345	17
2.3	แสดงความต้องการพื้นฐานของระบบคอมพิวเตอร์ที่ต้องมีเมื่อใช้โปรแกรม Blender	21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทคัดย่อ

การวิจัยนี้ ทำการออกแบบและสร้างโปรแกรมเกมกราฟิกสามมิติที่ตอบสนองตามการเคลื่อนไหวของศีรษะและการปั่นจักรยานของผู้เล่น โดยใช้เทคโนโลยีกราฟิกสามมิติด้วยโปรแกรม Blender3D กับเทคโนโลยีไมโครคอนโทรลเลอร์ตระกูล AVR ผ่านบอร์ดไมโครคอนโทรลเลอร์ Arduino มาประยุกต์ใช้เข้าด้วยกันเพื่อความเพลิดเพลินในการออกกำลังกายและทำให้การออกกำลังกายมีความสนุกสนานมากขึ้น โดยผู้เล่นสามารถเล่นเกมด้วยการปั่นจักรยานให้ภาพกราฟิกในเกมเคลื่อนที่ไปข้างหน้าควบคู่กับการบังคับทิศทาง ด้วยการเคลื่อนไหวศีรษะของผู้เล่น โดยความเร็วในการเคลื่อนที่ของเกมขึ้นอยู่กับความเร็วในการปั่นจักรยานของผู้เล่น

คำสำคัญ: เอวีอาร์, เซนเซอร์วัดความเร่ง

ABSTRACT

This research and build Graphics 3D's game respond to head moving and cycling of player by applying Graphics 3D Technology with Blender and AVR Microcontroller using Arduino Board together. For the purpose of creating fun and enjoyable with Exercise. When player cycling the Graphics game will move forward and player can control direction by moving head. The moving's velocity depend on cycling character of player

Keyword: AVR, Accelerometer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

งานวิจัยครั้งนี้ จะสำเร็จลุล่วงไปไม่ได้หากมิได้รับความอนุเคราะห์จากคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่สนับสนุนงบประมาณในการดำเนินการวิจัย ผู้ให้การช่วยเหลือในการทำวิจัย ได้แก่นางสาวฐิมาภรณ์ สุวาส นายณรงค์ฤทธิ์ ศรีสุกใส และผู้ที่มิได้กล่าวนาม ณ ที่นี้

ผู้ช่วยศาสตราจารย์บุณย์ชนะ ภู่ระหงษ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของการวิจัย

ในปัจจุบันโลกของเทคโนโลยีมีความก้าวหน้าและเติบโตเร็วไปในทุกๆ ด้าน ทั้งทางด้านการสื่อสาร คมนาคม เป็นต้น เทคโนโลยีที่ทันสมัยมากมายต่างเข้ามามีบทบาทในชีวิตของมนุษย์ ดังนั้นมนุษย์ในสังคมปัจจุบันจึงต้องปรับตัวพัฒนาให้เท่าทันเทคโนโลยีอยู่ตลอดเวลา โดยเฉพาะชีวิตในสังคมเมือง เนื่องด้วยสังคมเมืองมีการใช้ชีวิตที่บีบรัด มีข้อจำกัดทางด้านเวลาและสถานที่ ทำให้การดำเนินชีวิตไม่สามารถมีรูปแบบอิสระเช่นยุคสมัยก่อน การดำเนินชีวิตต่างๆ ที่สามารถทำได้สะดวกกลับลำบากมากยิ่งขึ้น อาทิ การออกกำลังกาย คนในสังคมเมืองอันแออัดย่อมต้องมีสถานที่จำกัดในการออกกำลังกาย จึงทำให้ต้องไปหาสถานที่ออกกำลังกายนอกบ้านเพื่อเปิดโลกทัศน์และไม่ให้เกิดความอึดอัดและความจำเจ โดยสถานที่ดังกล่าวอาจจะต้องใช้เวลาในการหาสูงอีกทั้งยังหาได้ยากมีความน่าเบื่อ ทางด้านสถานที่ อากาศ และเวลา ดังนั้นการออกกำลังกายในสังคมเมืองจึงไม่สามารถทำได้โดยสะดวก การออกกำลังกายในที่ร่มจึงเป็นตัวเลือกอันดับต้นๆ ของคนในสังคมเมืองในยุคปัจจุบัน โดยคาดว่าในอนาคตการออกกำลังกายในรูปแบบนี้จะได้รับความนิยมสูงขึ้น แต่เนื่องจากพื้นที่ที่มีจำกัดการออกกำลังกายในรูปแบบดังกล่าวอาจจะสร้างความเบื่อหน่ายได้ง่าย

ดังนั้นการประยุกต์ใช้แอพพลิเคชั่นในรูปแบบกราฟิกสามมิติเพื่อนำเข้ามาผสมผสานกับการออกกำลังกาย อาจช่วยให้การออกกำลังกายมีสีสัน เป็นที่น่าสนใจมากขึ้น นอกจากจะสามารถเสริมสุขภาพกาย ยังสามารถเสริมสุขภาพจิตได้ และอาจจะทำให้บุคคลที่ไม่ชอบออกกำลังกายหันมาปรับเปลี่ยนทัศนคติในด้านการดูแลสุขภาพตนเองมากขึ้น

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 ศึกษาการทำงานของเซนเซอร์วัดความเร่งเบอร์ ADXL345

1.2.2 ศึกษาการทำงานของไมโครคอนโทรลเลอร์ในตระกูล AVR โดยใช้Arduino

1.2.3 ศึกษาหลักการติดต่อข้อมูลระหว่างฮาร์ดแวร์กับซอฟแวร์ผ่านพอร์ตอนุกรม

1.2.4 ศึกษาหลักการรับ-ส่งข้อมูลระหว่างบอร์ด Arduino กับเซนเซอร์วัดความเร่งโดยใช้ I2C BUS

1.2.5 ศึกษาหลักการรับส่งค่าของไดนาโมจักรยาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2.6 ศึกษาโปรแกรมควบคุมกราฟิกสามมิติ ด้วยโปรแกรมภาษา Python

1.3 ขอบเขตของการวิจัย

1.3.1 ออกแบบแอพพลิเคชั่นระบบกราฟิกสามมิติ คล้ายคลึงเกมให้ผู้เล่นสามารถควบคุมทิศทางได้

1.3.2 ผู้เล่น(หรือผู้ใช้งานเครื่องออกกำลังกาย)สามารถกำหนดทิศทางในเกมได้โดยการเอียงศีรษะไปทางทิศนั้นๆ

1.4 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย

1.4.1 เข้าใจหลักการทั่วไปของเซนเซอร์วัดความเร่ง

1.4.2 เข้าใจการเขียนไมโครคอนโทลเลอร์ตลอดจนสามารถประยุกต์ใช้งานได้

1.4.3 มีความรู้ความเข้าใจโครงสร้างการส่งข้อมูลแบบ I2C BUS และสามารถเขียนโปรแกรมติดต่อระหว่างฮาร์ดแวร์ได้

1.4.4 ความเข้าใจหลักการทั่วไปของโปรแกรม Blender3D ตลอดจนสามารถสร้างในรูปแบบของเกมสามมิติได้

1.4.5 ความเข้าหลักการทั่วไปของโปรแกรมภาษา Python และสามารถเขียนควบคุมเกมสามมิติได้

1.5 เนื้อหาภายในการวิจัย

การวิจัยฉบับนี้ประกอบไปด้วยเนื้อหาในบทต่างๆดังนี้

บทที่ 1 กล่าวถึงความเป็นมาและความสำคัญของการวิจัย วัตถุประสงค์ของการวิจัย ขอบเขตของการวิจัย ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย อุปกรณ์ที่ต้องใช้ และขั้นตอนการดำเนินงาน

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่เกี่ยวข้องกับการวิจัยฉบับนี้ได้นำมารวบรวมไว้ เพื่อเป็นพื้นฐานในการศึกษาและการทำการทดลองตามการวิจัย

บทที่ 3 กล่าวถึงการออกแบบโครงสร้างด้านฮาร์ดแวร์และซอฟต์แวร์ เพื่อนำไปสร้างให้พร้อมใช้งานได้จริง

บทที่ 4 กล่าวถึงการทดลองและผลการทดลอง ซึ่งจะอธิบายถึงรายละเอียดการทำงานต่างๆ

บทที่ 5 กล่าวถึงการสรุปผลการดำเนินการ ซึ่งเป็นการนำผลการทดลองมาสรุปและชี้แจงถึงปัญหา อุปสรรคในระหว่างการดำเนินการ พร้อมชี้แนะแนวทางในการศึกษาและการพัฒนาต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

การวิจัยนี้ มีวัตถุประสงค์คือ การประยุกต์ใช้เซ็นเซอร์สามแกนกับกราฟิกสามมิติ เพื่อพัฒนาการออกกำลังกายให้มีรูปแบบที่น่าสนใจและแตกต่างไปจากเดิม โดยอาศัยการทำงานจากสองส่วนหลักๆ คือ ส่วนของอุปกรณ์และส่วนของกราฟิกสามมิติ อุปกรณ์ที่ใช้เป็นชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino ที่มี MCU เป็น ATmega328 ควบคุมร่วมกับเซนเซอร์วัดความเร่งทิศทางแกน X, Y, Z ที่เรียกว่า Accelerometer ADXL345 ใช้ลักษณะการส่งข้อมูลในรูปแบบอนุกรม I2C Bus โดยการทำงานของชุดอุปกรณ์ดังกล่าวจะเกิดขึ้นเมื่อมีการส่งค่าในรูปแบบอนาล็อกจากไดนาโมจักรยาน ในส่วนของกราฟิกสามมิติมีการออกแบบกราฟิกโมเดลด้วยโปรแกรม Blender3D และควบคุมการทำงานระหว่างอุปกรณ์กับกราฟิกสามมิติด้วยโปรแกรมภาษา Python ดังนั้นเพื่อให้อุปกรณ์และโปรแกรมต่างๆสามารถติดต่อกันได้ตามขอบเขตที่วางไว้ จึงต้องศึกษาข้อมูลและทฤษฎีที่เป็นพื้นฐานสำคัญ โดยแบ่งออกเป็นกลุ่มต่างๆ ตามลักษณะการทำงานดังนี้

2.1 ทฤษฎีไมโครคอนโทรลเลอร์

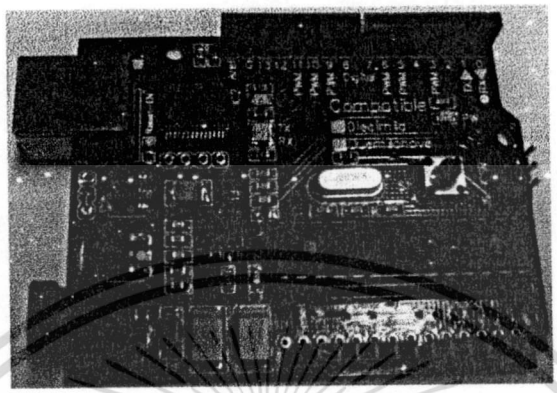
ไมโครคอนโทรลเลอร์ (Microcontroller) เป็นอุปกรณ์อิเล็กทรอนิกส์แบบหนึ่งที่รวมเอาหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์และลอจิก วงจรรับสัญญาณอินพุต วงจรส่งสัญญาณเอาท์พุต รวมถึงหน่วยความจำวงจรกำเนิดสัญญาณนาฬิกาไว้ด้วยกัน ทำให้สามารถนำไปใช้งานแทนวงจรอิเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี โดยไมโครคอนโทรลเลอร์มาจากคำสองคำรวมกันคือ “ไมโคร” ซึ่งหมายถึงไมโครโปรเซสเซอร์ เป็นอุปกรณ์ประมวลผลข้อมูลขนาดเล็กภายในประกอบด้วย หน่วยประมวลผลกลางหรือ CPU ประกอบด้วยหน่วยคำนวณทางคณิตศาสตร์และลอจิก วงจรเชื่อมต่อหน่วยความจำ วงจรกำเนิดสัญญาณนาฬิกา อีกคำหนึ่งคือคำว่า “คอนโทรลเลอร์” หมายถึง อุปกรณ์ควบคุม ดังนั้นไมโครคอนโทรลเลอร์จึงเป็นอุปกรณ์ที่ใช้ในการควบคุม โดยที่สามารถเขียนโปรแกรมเพื่อกำหนดรูปแบบการควบคุมได้อย่างเป็นอิสระ

2.1.1 ชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino รุ่น Duemilanove Compatible

เป็นไมโครคอนโทลเลอร์ตระกูล AVR โดยใช้ไมโครคอนโทลเลอร์เบอร์ ATmega 328 เป็น MCU ประจำบอร์ด โดย MCU รุ่นนี้ มีขา pin ทั้งหมด 28 ขาและมีจุดเด่นคือเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์ขนาดเล็ก แต่เพียบพร้อมไปด้วยทรัพยากรพื้นฐานต่างๆ อย่างครบถ้วน จึงมีความเหมาะสมเป็นอย่างยิ่งในการใช้งานทั่วไป สำหรับภายในมีทั้งระบบฮาร์ดแวร์ของ SPI, UART , I2C, Watchdog, Timer/Counter,PWM เป็นต้นและสามารถใช้ในการพัฒนาโปรแกรมด้วย Arduino ได้ทันที



รูปที่ 2.1 ชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino

2.1.2 คุณสมบัติของบอร์ดไมโครคอนโทรลเลอร์

ชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino มี ATmega328 เป็น MCU ตระกูล AVR ประจำบอร์ด โดยเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบ Crystal Oscillator ที่มีค่า 16 MHz เพื่อให้สามารถใช้งานพอร์ทสื่อสารอนุกรมได้อย่างลงตัว

- บอร์ดสามารถเปลี่ยนการติดตั้ง MCU เป็นแบบ 28 pin หรือเบอร์อื่นในอนุกรมเดียวกันได้โดยไม่ต้องมีการดัดแปลงใดๆ เช่น ATmega88 เป็นต้น
- มีหน่วยความจำแบบ Flash 32 Kbyte โดยแบ่งเป็น Boot loader 2 Kbyte
- มี EPROM 1Kbyte/SRAM 2Kbyte
- มีพอร์ตเป็น 14 digital input/output pins ซึ่งมี 6 pin สามารถสร้างเป็น PWM outputs
- มีพอร์ต 6 Analog input/output pins
- ไฟกระแสตรง (DC) ขา I/O pin มีค่า 40 mA
- ไฟกระแสตรง (DC) ขา 3.3V pin มีค่า 50 mA
- MCU ประจำบอร์ดที่ได้รับการติดตั้ง Boots loader สามารถอัพโหลดโค้ดให้บอร์ดผ่านทางพอร์ตสื่อสารแบบอนุกรมได้ทันที
- มีขั้วต่อ USB Interface สื่อสารข้อมูลแบบอนุกรมเข้า คอมพิวเตอร์ได้ทันที
- มี LED สำหรับแสดงสถานะไฟเลี้ยง
- มี LED แสดงสถานการณ์รับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ใช้ไฟเลี้ยง ประจำบอร์ด 7-12 Volt33

2.1.3 ทฤษฎีภาษา Arduino

"Arduino" เป็นภาษาอิตาลีซึ่งเป็นชื่อของโครงการพัฒนาไมโครคอนโทรลเลอร์ตระกูล AVR แบบ Open source ที่ได้รับการปรับปรุงมาจากโครงการ Open source ของAVR

Arduino มีจุดเด่นในเรื่องของความง่ายในการเรียนรู้และใช้งาน เนื่องจากมีการออกคำสั่งต่างๆขึ้นมาสนับสนุนการใช้งาน ด้วยรูปแบบที่ง่ายไม่ซับซ้อน ซึ่งแม้ว่า Arduino จะมีรูปแบบการใช้งานคล้ายๆกับไมโครคอนโทรลเลอร์อย่าง Basic Stap ของ Parallax แต่ก็มีจุดเด่นกว่ารายอื่นคือ

- ราคาไม่แพง เนื่องจากมี Source Code และวงจร แจกให้ฟรี สามารถต่อวงจรขึ้นมาใช้เองได้ รวมถึงมีการเปิดเผยวงจร Source code ทั้งหมดทำให้สามารถนำไปพัฒนาต่อยอดได้ดี ทั้งด้านฮาร์ดแวร์และซอฟแวร์
- โปรแกรมที่ใช้พัฒนาของ Arduino สามารถรองรับการทำงานทั้ง Window, Linux และ OSx
- รูปแบบคำสั่งง่ายต่อการใช้งาน แต่สามารถนำไปใช้งานจริงๆ กับส่วนที่มีความซับซ้อนมากๆได้ และยังสามารถสร้างคำสั่งรวมถึง Library ใหม่ๆ ขึ้นมาใช้งานเองได้ เมื่อมีความชำนาญมากขึ้น

2.1.4 เปรียบเทียบภาษาซีกับ Arduino

สำหรับการเขียนโปรแกรมของ Arduio นั้นใช้ภาษา C++ ซึ่งเป็นรูปแบบของภาษาซีประยุกต์รูปแบบหนึ่งที่มีโครงสร้างการทำงานของตัวภาษาโปรแกรมโดยรวมคล้ายกับ ภาษาซีมาตรฐานทั่วไป เพียงแต่ได้มีการปรับปรุงเพื่อลดความยุ่งยากในการใช้งานลดลง และให้ผู้ใช้สามารถใช้งานเขียนโปรแกรมได้ง่าย สะดวกมากกว่าการเขียนภาษาซีแบบมาตรฐาน แต่ในความเป็นจริงนั้นโปรแกรมดังกล่าวไม่ใช่ C-compiler โดยตรง เนื่องจาก Arduino จะมีลักษณะการทำงานเช่นเดียวกับ Text Editor ของภาษา C++ ตัวหนึ่งโดยจะทำงานร่วมกับ Utility บางส่วนที่ Arduino สร้างขึ้นมารองรับโดย Arduino จะใช้รูปแบบการทำงานของ Editor เป็นฉากหน้า ในการติดต่อสื่อสารกับผู้ใช้เท่านั้น ส่วนเบื้องหลังจริงๆแล้ว Arduino จะไปเรียกใช้ตัวแปลภาษาซี และ Utility อื่นที่ใช้เป็นเครื่องมือพัฒนาโปรแกรมของไมโครคอนโทรลเลอร์ตระกูล AVR อีกทีหนึ่ง Arduino จะเลือกใช้คอมไพลเลอร์ของ "GNU AVR-GCC Toolchain" ร่วมกับ Library function ของ "avr-libc" ส่วน Utility ที่ใช้ในการอัพโหลดโค้ดให้กับ AVR จะใช้ในส่วนของ "AVRDude"

2.1.5 โครงสร้างการเขียนโปรแกรมภาษาซีของ Arduino

ภาษาซีของ Arduino จะจัดแบ่งรูปแบบโครงสร้างของการเขียนโปรแกรมเป็นส่วนย่อยๆหลายๆส่วน โดยเรียกแต่ละส่วนว่า "ฟังก์ชั่น" และเมื่อนำฟังก์ชันมารวมเข้าด้วยกัน ก็จะเรียกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"โปรแกรม" โดยโครงสร้างการเขียนโปรแกรมของ Arduino ทุกๆโปรแกรมจะประกอบไปด้วยฟังก์ชั่นจำนวนเท่าใดก็ได้ แต่อย่างน้อยสุดต้องมี 2 ฟังก์ชั่นคือ setup() และ loop()

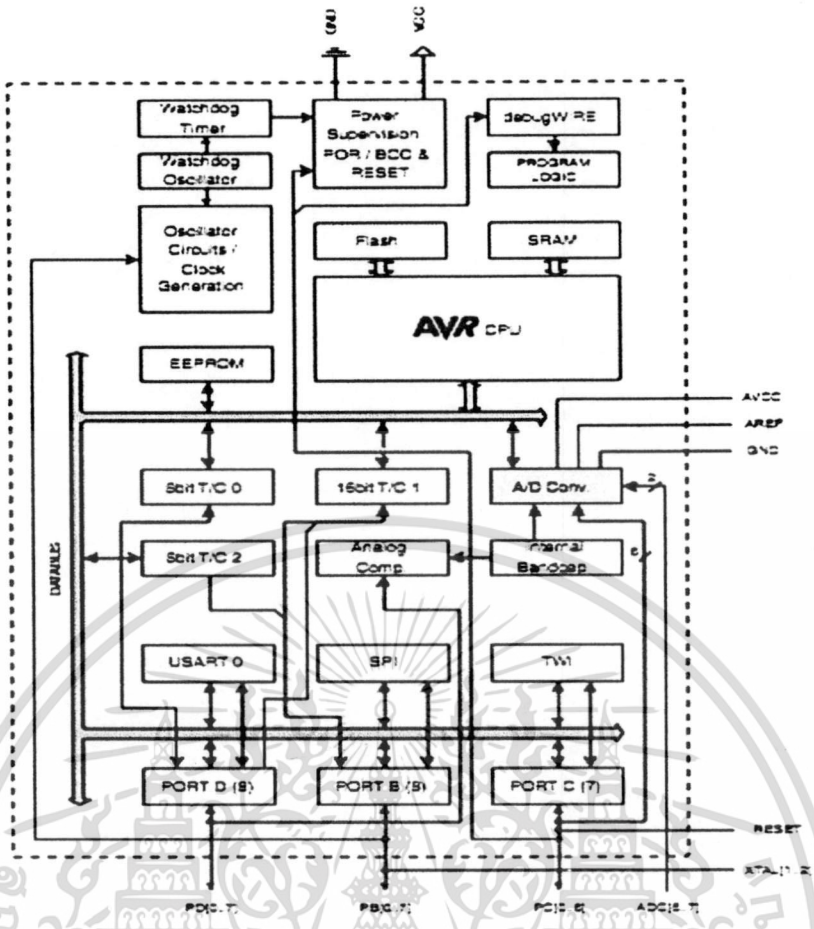
- **Setup()** : เป็นฟังก์ชั่นบังคับที่ต้องกำหนดให้มีทุกๆโปรแกรม ถึงแม้ว่าบางโปรแกรมจะไม่ต้องการใช้งานก็ยังจำเป็นต้องประกาศไว้เสมอ เพียงแต่ไม่ต้องเขียนคำสั่งใดๆไว้หลังวงเล็บปีกกา {} ที่ใช้เป็นตัวกำหนดขอบเขตของฟังก์ชั่น โดยฟังก์ชันนี้จะใช้สำหรับบรรจุคำสั่งที่ต้องการให้โปรแกรมทำงานเพียงรอบเดียว ตอนเริ่มต้นทำงานของโปรแกรมครั้งแรกเท่านั้น ซึ่งได้แก่คำสั่งเกี่ยวกับการ Setup ค่าการทำงานต่างๆ เช่น การกำหนดหน้าที่ของการใช้งานของ PinMode และค่า Baudrate สำหรับการใช้งานสื่อสารพอร์ตอนุกรม เป็นต้น
- **Loop()** : เป็นส่วนฟังก์ชั่นบังคับที่ต้องกำหนดให้มีในทุกโปรแกรม เช่นเดียวกันกับฟังก์ชั่น Setup() โดยฟังก์ชั่น Loop() นี้จะใช้ในการบรรจุคำสั่งที่ต้องการให้โปรแกรมทำงานเป็นวนรอบซ้ำๆ กันไปไม่รู้จบ ซึ่งเปรียบเทียบได้กับ ฟังก์ชั่น main () ใน ANSCI-C นั่นเอง

2.2 ทฤษฎีไมโครคอนโทรลเลอร์ตระกูล AVR รุ่น ATmega 328P

2.2.1 สถาปัตยกรรมชั้นสูงแบบ RISC(Reduce Instruction Set Computer)

- RISC คือตัวทำให้การประมวลผลมีความเร็ว 1 คำสั่ง/1 clock หรือ CPU สามารถประมวลคำสั่งได้ 1 MIPS/MHz
- ชุดคำสั่ง 131 คำสั่งต่อ 1 รอบนาฬิกา
- รีจิสเตอร์ขนาด 8 บิต 32 ตัว
- ความเร็วในการประมวลผลมากกว่า 20 ล้านคำสั่งต่อวินาที (MIPS) ที่ 20 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 สถาปัตยกรรมชั้นสูงแบบ RISC [1]

โดยมีคุณสมบัติต่างๆดังต่อไปนี้

หน่วยความจำ

- แบบ 32 Kbyte สามารถเขียนลบโปรแกรมได้ 10,000 ครั้ง
- แบบ EPROM 1Kbyte สามารถเขียนและลบโปรแกรมได้ 100,000 ครั้ง
- แบบ SRAM 2 Kbyte

ไฟเลี้ยง

- ระหว่าง 1.8 ถึง 2.5 v

ความถี่สัญญาณนาฬิกา

- ระหว่าง 0 ถึง 4 MHz
- มีการรองรับอุปกรณ์ต่อพ่วง
- อุปกรณ์สื่อสารข้อมูลแบบอนุกรมแบบ I2C และ USART

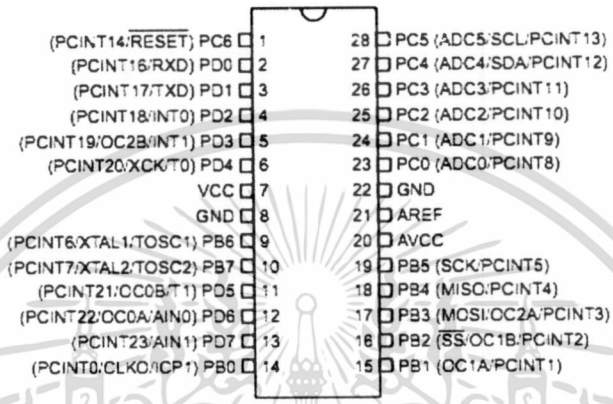
อื่นๆ

- มีระบบ Reset แบบอัตโนมัติเมื่อจ่ายกระแสไฟฟ้าเข้าไมโครคอนโทรลเลอร์
- มีฟังก์ชั่นตรวจสอบแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีระบบการขัดจังหวะทั้งภายในและภายนอก
- มีระบบตรวจจับความผิดพลาดของ CPU
- มีโหมดอนุรักษ์พลังงาน 5 mode ได้แก่ Idle, ADC Noise Reduction, Power-Save, Power-Down, Standby

2.2.2 ส่วนประกอบต่างๆของไอซี ATmega 328P



รูปที่ 2.3 ไอซี ATmega 328P [1]

ตารางที่ 2.1 ตารางแสดงฟังก์ชั่นการทำงานของ ATmega 328P

ชื่อ	รายละเอียด	ขา
GND	ขากราวด์ต่อสายดิน	8,22
VCC	ไฟเลี้ยง 1.8 ถึง 5.5v	7
Port B(PB 7: 0) XTAL1/XTAL2/TOSC1/TOSC2	- เป็นพอร์ต 2 ทิศทาง ขนาด 8 บิต โดยสามารถ กำหนดให้ขาของแต่ละพอร์ตสามารถตั้งค่าให้ Pull up Resistor ได้ (ภายในเป็นอิสระแยกจากกัน เพื่อดึงแรงดันของลอจิก 1 ให้เท่ากับ 5 v) - สามารถใช้งานพิเศษตามความต้องการของ ATmega 328 โดยขึ้นอยู่กับการตั้งค่าสัญญาณ นาฬิกาที่ขา PB6 ที่ใช้เป็นแรงดัน Oscillator และ ขาอินพุตของวงจรสัญญาณ Clock Oscillator	9,10,14-19
PC6/RESET	ขา Reset	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อ	รายละเอียด	ขา
Port D (PD7:0)	- เป็นพอร์ตสองทิศทางขนาด 8 บิต โดยสามารถกำหนดให้ขาของแต่ละพอร์ตสามารถตั้งค่าให้ Pull up Resistor ได้ - สามารถใช้งานพิเศษตามความต้องการของ ATmega 328	1-6,11-13
AVCC	ใช้จ่ายไฟให้กับวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอล มักจะต่อเข้ากับขา VCC	20
AREF	แรงดันอ้างอิงที่ใช้งานในส่วนของวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอลมักต่อกับ VCC	21
ADC7:6 (TQFP and QFN/MLF Package Only)	ขากำลังงานใช้แปลงสัญญาณอนาล็อกเป็นดิจิตอล	23-28
Port C (PC5:0)	- เป็นพอร์ตสองทิศทางขนาด 8 บิต โดยสามารถกำหนดให้ขาของแต่ละพอร์ตสามารถตั้งค่าให้ Pull up Resistor ได้ - สามารถใช้งานพิเศษตามความต้องการของ ATmega 328	23-28

2.3 ทฤษฎีการสื่อสารแบบอนุกรม

ทฤษฎีการสื่อสารแบบอนุกรม เป็นการรับส่งข้อมูลทีละบิตจนครบ 1 ไบต์ ถ้าต้องการส่งข้อมูล 1 ไบต์คือ D0 - D7 อาจจะส่ง D0 ออกไปก่อน แล้วตามด้วย D1 ไปเรื่อยๆจนถึง D7 โดยการส่งข้อมูลแบบอนุกรมมีความแตกต่างกับแบบขนานทั้งข้อดีและข้อเสียดังนี้ การส่งข้อมูลแบบขนานสามารถรับส่งข้อมูลได้อย่างรวดเร็ว เพราะการส่งข้อมูลสามารถทำได้ทีเดียวครบ 1 ไบต์ แต่ถ้าต้องการส่งระยะทางไกลๆจะต้องสิ้นเปลืองสายสัญญาณมาก สำหรับการส่งข้อมูลแบบอนุกรม จะแตกต่างคือ ถ้าส่งระยะทางไกลๆจะประหยัดสายสัญญาณ จากการใช้สายสัญญาณอย่างน้อยสองเส้นคือ สายสัญญาณกับสายกราวด์ แต่การรับส่งข้อมูลจะใช้เวลานานกว่าเนื่องจากเป็นการรับส่งค่าทีละบิต

2.3.1 การสื่อสารข้อมูลแบบ Asynchronous

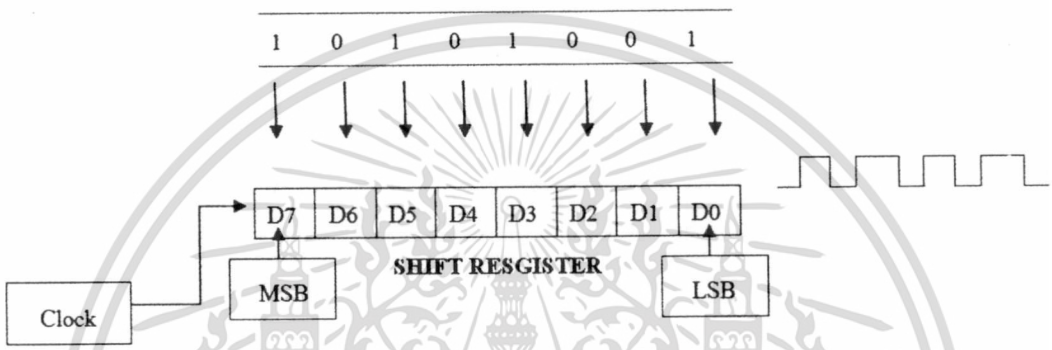
เป็นการรับและส่งข้อมูลโดยไม่จำเป็นต้องมีสัญญาณนาฬิการ่วมด้วยแต่จะใช้การกำหนดค่าอัตราเร็วในการรับส่งข้อมูลให้มีค่าเท่ากัน ซึ่งเรียกอัตรานี้ว่า Baud rate (Bit/second)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบ Asynchronous ประกอบด้วย 4 ส่วน คือ

1. บิตเริ่มต้น (Start bit) มีขนาด 1 บิต
2. บิตข้อมูลแบบอนุกรม มีขนาด 8 บิต
3. บิตตรวจสอบพาริตี้ (parity bit) มีขนาด 1 บิตหรือไม่มีก็ได้
4. บิตหยุด (Stop bit) มีขนาด 1 บิต

เมื่อไมโครคอมพิวเตอร์ต้องการรับส่งข้อมูลแบบอนุกรม ตัวไมโครคอมพิวเตอร์จะส่งข้อมูลออกไปแบบขนานก่อน จากนั้นจะมีอุปกรณ์ที่มาต่อพอร์ต เพื่อแปลงข้อมูลแบบขนานให้เป็นอนุกรมทีหนึ่ง ตัวแปลงข้อมูลนี้อาจพิจารณาว่าเป็น Shift Register



รูปที่ 2.4 การส่งข้อมูลแบบอนุกรม [2]

สำหรับตัวรับข้อมูลแบบอนุกรมนั้นเมื่อตัวรับข้อมูลทำงานจะเป็นการรับเข้ามาใน shift Register แล้วส่งข้อมูลให้ไมโครคอมพิวเตอร์แบบขนานอีกที ระบบคอมพิวเตอร์ในปัจจุบันมีตัวแปลง Pararell to Serial และ Serial to Parallel อยู่ในชิพไอซีเรียกว่า Universal Asynchronous Receiver Transmitter (UART) การส่งข้อมูลแบบอนุกรมนั้นจะต้องมีการเพิ่มเติมข้อมูลบางอย่างเข้าไปเพื่อทำให้ข้อมูลมีความถูกต้องมากขึ้น

Start	D0	D1	D2	D3	D4	D5	D6	D7	Parity	Stop

รูปที่ 2.5 บิตต่างๆของการรับส่งข้อมูลแบบอนุกรม [2]

ถ้ามีการส่งข้อมูลแบบ 8 บิตจะต้องมีการส่งบิตแรกออกไปก่อน เรียกว่า บิตเริ่มต้น (Start Bit) ถ้ามีการส่งข้อมูลหลายๆไบต์ออกมา บิตนี้จะเป็นตัวบอกว่ามีข้อมูลใหม่มาแล้ว โดยทั่วไปบิตเริ่มต้นมักมีระดับลอจิกเป็น “0” ต่อจากนั้นจะเป็น ข้อมูลบิต D0 - D7 จากนั้นตามด้วยบิตตรวจสอบและบิตหยุด เพื่อบอกการสิ้นสุดของบิตข้อมูลซึ่งบิตหยุดอาจมากกว่า 1 บิตก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสื่อสารแบบอนุกรมนี้ การกำหนดอัตราเร็วในการรับส่งข้อมูลจะบอกเป็นบิตต่อวินาที bps ที่เรียกว่าอัตราบอดเรต (BAUD RATE) โดยค่ามาตรฐานที่ใช้กรณีหลายค่า เช่น ค่า9600 หรือ ค่า 115200 เป็นต้น

2.4 หลักการพื้นฐานของ USB

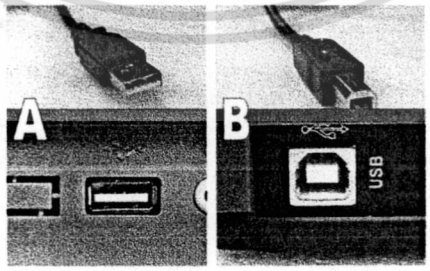
USB หรือ Universal Serial Bus ได้ถูกกำหนดขึ้นมาโดยบริษัทยักษ์ใหญ่ผู้นำทางด้านอุปกรณ์ไฟฟ้า อิเล็คทรอนิคส์และคอมพิวเตอร์ ซึ่งประกอบไปด้วย COMPAQ, IBM, DEC, Intel, Microsoft, NEC และ Northern Telecom ช่วยกันวางมาตรฐานให้อยู่ในรูปแบบเดียวกัน โดยในยุคเริ่มแรกนั้นมาตรฐานของ USB ที่ ออกสู่สาธารณะชนเป็นครั้งแรกเมื่อวันที่ 11 พฤศจิกายน ปี ค.ศ. 1994 คือ Revision 0.7 และได้ปรับปรุงแก้ไขเรื่อยมา จนกระทั่งเมื่อ วันที่ 15 มกราคม ค.ศ. 1996 ได้ออกมาเป็น Revision 1.0 (USB1.0) ได้เป็นผลสำเร็จและยังได้ปรับปรุง พัฒนาแก้ไขปัญหาต่างๆ เรื่อยมาจนเมื่อวันที่ 23 กันยายน ค.ศ. 1998 ได้เป็น Revision 1.1 (USB1.1) เมื่อความเร็วที่ได้ ยังไม่เพียงพอกับความต้องการ

ดังนั้นทางกลุ่มผู้พัฒนา หรือ USB-IF (USB Implementers Forum, Inc.) ได้ร่างมาตรฐาน USB รุ่นใหม่ และได้ข้อสรุป เป็นมาตรฐานที่แน่นอน คือ USB 2.0 ในเดือนเมษายน ปี ค.ศ. 2000 สำหรับความเร็วในการรับ-ส่งข้อมูลนั้น USB1.1 จะมีความเร็วอยู่ที่ 12Mbps ส่วน USB 2.0 นั้น รองรับระดับการรับส่งข้อมูลได้ถึง 3 ระดับ คือ

- ความเร็ว 1.5 Mbps (Low Speed) สำหรับการเชื่อมต่อกับอุปกรณ์ที่ไม่จำเป็นต้องส่งข้อมูลคราวละมากๆ
- ความเร็ว 12 Mbps (Full Speed) สำหรับการเชื่อมต่อกับ USB 1.1
- ความเร็ว 480 Mbps (Hi-Speed) สำหรับการเชื่อมต่อกับ USB 2.0 ด้วยกัน

2.4.1 ชนิดของหัว Connector

Connector มีอยู่ 2 แบบ คือ แบบ A และ แบบ B และ Socket ดังรูป



รูปที่ 2.6 ชนิดของหัว Connector แบบ A และแบบ B [3]

ลักษณะของการทำงานของหัวต่อทั้งสองแบบมีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แบบ A : เป็นการส่งข้อมูลจากอุปกรณ์ไปยังเครื่อง Computer เพื่อการประมวล เรียกว่า UpStream
- แบบ B : เป็นการส่งข้อมูลเข้าหาอุปกรณ์ เรียกว่า DownStream

2.4.2 ลักษณะการเชื่อมต่อ

การเชื่อมต่อใช้งานนั้นสามารถ เชื่อมต่อร่วมกันได้ทั้งที่เป็น USB1.1 และ USB 2.0 แต่จะได้ความเร็วที่ต่างกัน

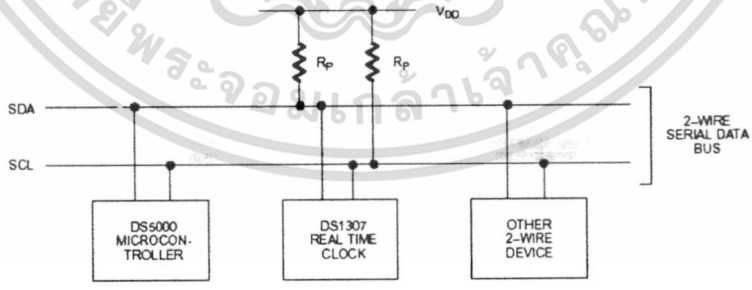
ถ้าหากต่ออุปกรณ์มาตรฐาน USB1.1 บนระบบบัสที่เป็น USB2.0 จะได้ความเร็ว = 12Mbps

ถ้าหากต่ออุปกรณ์มาตรฐาน USB2.0 บนระบบบัสที่เป็น USB1.1 จะได้ความเร็ว = 12Mbps

ถ้าหากต่ออุปกรณ์มาตรฐาน USB2.0 บนระบบบัสที่เป็น USB2.0 จะได้ความเร็ว = 480Mbps

2.5 ทฤษฎีการสื่อสารแบบ I2C Bus

Inter Integrate Circuit Bus (IIC Bus) เรียกสั้นๆว่า "I2C Bus" ซึ่งเป็นชื่อของการติดต่อสื่อสารแบบอนุกรมแบบหนึ่ง ถูกคิดค้นและพัฒนาโดย PHILIPS SEMICONDUCTOR เมื่อหลายปีก่อน แต่เพิ่งมาได้รับความแพร่หลายเมื่อไม่นานมานี้ ซึ่งปัจจุบันนี้ได้มีหลายบริษัทให้ความสนใจในการติดต่อสื่อสารแบบนี้เป็นอย่างมาก เนื่องจากรูปแบบการเชื่อมต่ออุปกรณ์ด้วยระบบบัสแบบนี้จะมีข้อดีคือใช้สัญญาณในการเชื่อมต่อเพียง 2 เส้น (SCL และ SDA) แต่สามารถเชื่อมต่ออุปกรณ์จำนวนหลายตัวในบัสเดียวได้ ซึ่งปัจจุบันถือว่าเป็นยุคสมัยของไมโครคอนโทรลเลอร์ขนาดเล็ก เนื่องจากระบบการทำงานของวงจรต่างๆมุ่งเน้นออกแบบให้มีขนาดเล็กกะทัดรัดและสามารถใช้งานได้หลายหลาย ดังนั้นพวก chip support ต่างๆจึงได้ออกแบบมาเพื่อตอบสนองการใช้งานแบบ I2C Bus มากยิ่งขึ้น



รูปที่ 2.7 ลักษณะการการเชื่อมต่ออุปกรณ์แบบ I²C BUS [4]

2.5.1 การเชื่อมต่อบัสแบบ I2C Bus

การเชื่อมต่อนี้จะใช้สัญญาณทั้งหมด 2 เส้นคือ SCL และ SDA โดยการติดต่อระหว่างอุปกรณ์จะเป็นแบบ 2 ทิศทาง โดยสัญญาณทั้งสองเส้นจะต้องต่อกับตัวต้านทาน Pull up ไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้สถานะของบัสในขณะที่ไม่ถูกใช้งานเป็นลอจิก “1” ทั้งคู่โดยอุปกรณ์ต่างๆที่ถูกออกแบบมาเพื่อเชื่อมต่อกับบัสแบบนี้ จะต้องสร้างวงจรภาคเอาพุตท์ให้เป็นแบบ Open Drain หรือ Open collector เสมอเพื่อให้อุปกรณ์สามารถติดต่อกันได้มากกว่า 1 อุปกรณ์

2.5.2 หลักการของ I2C

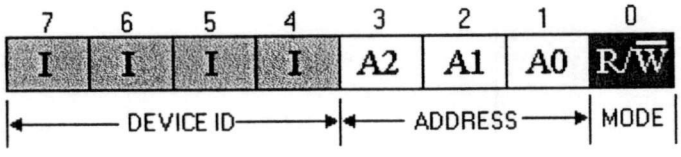
บัส I2C ประกอบด้วยสายสัญญาณ 2 เส้นดังที่ได้กล่าวมาแล้วคือ SDA และ SCL อุปกรณ์ที่ต่อพ่วงบนบัสสามารถมีได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูปแบบของการติดต่อบนบัส หรือเรียกว่า โปรโตคอล (protocol) เพื่อให้ผู้ใช้งานทราบว่า ขณะนี้อุปกรณ์ใดติดต่อกันอยู่ และอุปกรณ์ตัวใดเป็นตัวรับหรือตัวส่ง ต่อไปนี้เป็นการอธิบายลักษณะ หน้าที่ และนิยามของอุปกรณ์ที่ต่ออยู่บนบัส I2C เพื่อเป็นข้อตกลงพื้นฐานก่อนที่จะอธิบายการทำงานของบัส I2C ต่อไป

อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูลเรียกว่า ตัวส่ง (transmitter) อุปกรณ์ที่เป็นผู้รับข้อมูลเรียกว่า ตัวรับ (receiver) ในอุปกรณ์บนบัส I2C สามารถเป็นได้ทั้งตัวรับและตัวส่งบางอุปกรณ์ทำหน้าที่เป็นตัวรับเพียงอย่างเดียวจะไม่มีอุปกรณ์ใดบนบัส I2C เรียกว่า มาสเตอร์ (master) อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส I2C เรียกว่า สเลฟ (slave) โดยข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัส I2C คือ

1.) การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น

2.) ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูงสายข้อมูลต้องรักษาข้อมูลไว้ อย่าให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้น สัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาควบคุมแทน

การรับส่งข้อมูลของ I2C บัส จะเริ่มต้นด้วยการที่ตัวแม่สร้างสภาวะเริ่มต้น (Start Condition) เพื่อขอใช้บัสจากนั้นจึงเริ่มการส่งรหัสควบคุม (Control Byte) เพื่อใช้ระบุตำแหน่งของแอดเดรสตัวลูกที่ต้องการใช้ติดต่อด้วยในระบบบัส โดยค่าตำแหน่งแอดแดรสนี้อุปกรณ์แต่ละตัวจะมีรหัสแอดเดรสที่แตกต่างกันออกไป ไม่มีการซ้ำกันในระบบบัสเดียวกัน โดยรหัส Control Byte จะมีขนาด 8 บิตซึ่ง 7 บิตแรก เริ่มจาก MSB จะเป็นค่าตำแหน่งแอดเดรสของตัวลูก ส่วนบิตที่ 8(LSB) จะเป็นบิตสุดท้ายของไบท์ใช้สำหรับระบุทิศทางของการรับส่ง (R/W) โดยถ้าบิต LSB มีค่าเป็น “0” จะหมายถึงตัวแม่เขียนข้อมูลไปหาตัวลูก โดยข้อมูลจะรับส่งกันครั้งละ 1 ไบต์ และปิดท้ายข้อมูลของแต่ละไบต์ด้วยบิตแสดงการตอบรับ (Acknowledge Bit)โดยลักษณะโครงสร้าง Control Byte ของอุปกรณ์ แบบ I2C มีดังนี้



รูปที่ 2.8 ลักษณะของ Control Byte ของ I2C Bus [4]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.8 จะเห็นได้ว่ารหัส Control Byte ของอุปกรณ์ I2C จะมีขนาด 8 บิต โดยที่บิตที่ 7 ถึงบิตที่ 4 จะเป็นรหัสประจำตัวของอุปกรณ์แต่ละตัวที่ถูกกำหนดไว้ตายตัวจากโรงงาน ซึ่งผู้ใช้งานต้องศึกษาข้อมูลจากคู่มือ Data sheet ของอุปกรณ์นั้นเองว่าอุปกรณ์ที่จะนำมาใช้งานมีรหัสประจำตัวเป็นเท่าใด ส่วนบิตที่ 3 ถึงบิตที่ 1 จะมีไว้สำหรับการเลือกต่ออุปกรณ์ที่อยู่ในบัส โดยค่าของทั้ง 3 บิตนี้ จะต้องเป็นค่าที่ ตรงกับการกำหนดสถานะทางลอจิก ให้กับขาสัญญาณจอง A2,A1,A0 ของอุปกรณ์ด้วย ตัวอย่างเช่น อุปกรณ์ที่มีรหัสประจำตัวเป็น “0111” อาจถูกออกแบบให้สามารถต่อร่วมกันภายในบัสเดียวกันจำนวน 8 ตัว โดยการกำหนดสภาวะของลอจิกให้กับขาสัญญาณ A2,A1,A0 ให้มีความแตกต่างกันจากวงจรที่ต่ออยู่

แต่อย่างไรก็ตาม อุปกรณ์ I2C บางตัว อาจถูกออกแบบให้ต่อได้ในระบบบัสเดียวกันเพียงตัวเดียวโดยไม่มีขาสัญญาณในการเลือกตำแหน่งของอุปกรณ์อยู่ด้วย ค่าของ Control Byte ในตำแหน่งบิตที่ 3 ถึงบิตที่ 1 ก็อาจถูกกำหนดไว้ตายตัวเป็น 000 เสมอก็ได้ ส่วนค่า Control Byte ของบิต 0 นั้นจะถูกกำหนดไว้เป็นทิศทางของข้อมูล โดยถ้าบิต 0 มีค่าเป็น 1หมายถึงตัวแม่ต้องการอ่านค่าข้อมูลจากอุปกรณ์ ถ้ามีค่าเป็น 0 หมายถึงตัวแม่ต้องการเขียนค่า

สำหรับการรับส่งข้อมูลนั้นไม่มีการกำหนดตายตัวว่าจะต้องส่งกันครั้งละกี่ไบต์ ขึ้นอยู่กับข้อตกลงระหว่างอุปกรณ์แต่ละชนิดจะกำหนดขึ้น โดยในการรับส่งแต่ละครั้ง ตัวแม่จะเป็นตัวควบคุมการรับส่งเองทั้งหมด ซึ่งในกรณีที่ตัวแม่ต้องการติดต่อกับอุปกรณ์หลายๆตัวนั้น หลังจากตัวแม่สร้างสภาวะเริ่มต้นเสร็จแล้ว และทำการติดต่อข้อมูลกับอุปกรณ์เรียบร้อยแล้ว ถ้าตัวแม่ต้องการติดต่อข้อมูลกับอุปกรณ์อื่นไม่จำเป็นต้องสร้างสภาวะสิ้นสุด แต่ตัวแม่สามารถสร้างสภาวะเริ่มต้นขึ้นมาใหม่พร้อมกับส่ง Control Byte ใหม่ไปติดต่อกับอุปกรณ์ที่ต้องการได้ทันที เมื่อทำการติดต่อรับส่งข้อมูลเรียบร้อยแล้วจึงสั่งหยุด Stop Condition เพื่อเป็นการเลิกใช้บัสที่มีอยู่และปล่อยให้เป็นบัสว่าง

\

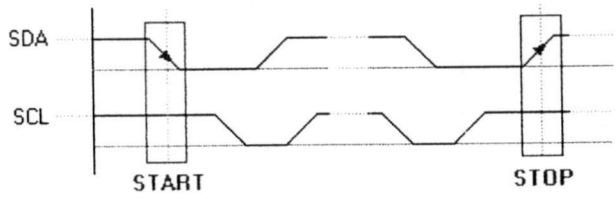
สภาวะที่เกิดขึ้นบนบัส I2C มีด้วยกัน 5 สภาวะ ดังนี้

1.) บัสว่าง (Bus not busy) สภาวะนี้เกิดขึ้นเมื่อสถานะลอจิกบนสาย SDA และ SCL เป็นลอจิกสูงทั้งคู่ นั้นหมายความว่า การถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้

2.) เริ่มต้นการถ่ายทอดข้อมูล (start data transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะเริ่มต้น (START)

3.) หยุดการถ่ายทอดข้อมูล (stop data transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลง
ระดับลอจิกจากต่ำไปสูง ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะหยุด (STOP)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 เงื่อนไขของสภาวะเริ่มต้น และสิ้นสุดของ I2C Bus [4]

4.) ข้อมูลดำรงอยู่บนบัส (data valid) สภาวะนี้เกิดขึ้นถัดจากสภาวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA คือข้อมูลที่ทำการถ่ายทอด เมื่อสาย SCL เป็นลอจิกสูง สถานะที่สาย SDA ต้องคงที่ เพื่อให้อุปกรณ์รับรู้ข้อมูลในจังหวะนั้นว่าเป็น “0” หรือ “1” ข้อมูลอาจเกิดการเปลี่ยนแปลงได้ในขณะที่สาย SCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ขา SDA ต้องคงที่ตลอดช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะลอจิกในขณะที่สาย SCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ทำการควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุดหรือสภาวะเริ่มต้นก็ได้ ทำให้ข้อมูลที่ทำการถ่ายทอดเกิดความผิดพลาดขึ้น

รูปที่ 2.10 ช่วงเวลารับส่งบิตข้อมูลของ I2C BUS [4]

5.) รับข้อมูล (acknowledge) เกิดขึ้นหลังจากที่การถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิตเรียกว่า บิตรับรู้ (acknowledge bit) มีสถานะเป็นลอจิกสูง หลังจากส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษ ซึ่งสัมพันธ์กับสัญญาณนาฬิกา เพื่อตอบสนองบิตรับรู้ที่ส่งมาจาก ตัวส่ง ทางด้านตัวรับจะส่งบิตรับรู้ที่มีสถานะลอจิกต่ำลงบนบัส อุปกรณ์สเลฟที่ถูกอ้างถึงในการติดต่อหรือกำลังติดต่ออยู่ในขณะนั้นก็จะกำเนิดบิตรับรู้เพื่อตอบสนอง ให้ทราบว่าได้รับข้อมูลในแต่ละไบต์เรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

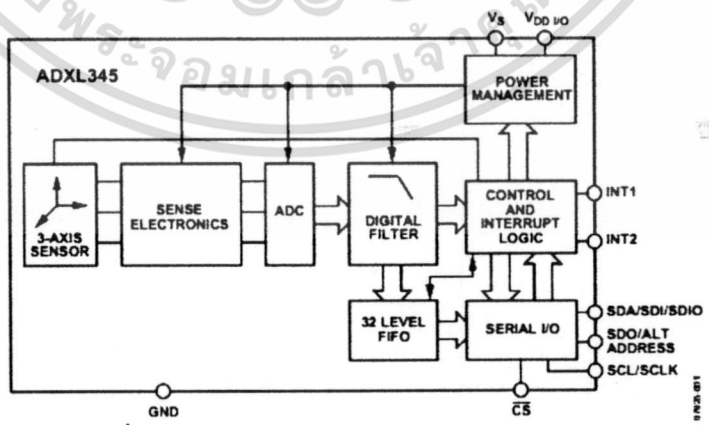
2.6 Accelerometer ADXL345

รูปที่ 2.11 เซนเซอร์วัดความเร่ง ADXL345 [5]

ADXL345 เป็นเซนเซอร์วัดความเร่งสามแกน มีขนาดเล็กบางและใช้กำลังน้อย เซนเซอร์นี้ให้ค่าความละเอียดสูงถึง 16g ซึ่งข้อมูลสามารถส่งได้ถึง 16 บิต สามารถส่งข้อมูลได้ทั้งการส่งข้อมูลแบบ SPI หรือ I2C อีกทั้ง ADXL345 สามารถนำมาประยุกต์ใช้กับ Application ในรูปแบบMotion ได้ดีเพราะสามารถตอบสนองการเคลื่อนไหวได้อย่างละเอียดสูงถึง 3.9 mg/LSB นั่นหมายความว่าสามารถเปลี่ยนแปลงค่ามุมได้ครั้งละต่ำกว่า 1 องศา

2.6.1 โครงสร้างภายในของ ADXL345

ภายในโครงสร้างของเซนเซอร์ ADXL345 มี ADC เพื่อแปลงจากสัญญาณอนาล็อกเป็นสัญญาณดิจิตอล มีตัวกรองสัญญาณดิจิตอล รวมถึงการส่งข้อมูลสองแบบคือ แบบอินเทอร์รัพท์และแบบ 32 LEVEL FIFO อีกทั้งยังมีการส่งข้อมูลแบบอนุกรมได้ทั้งรูปแบบ SPI BUS และ I2C BUS นอกจากนี้ยังมีตัวควบคุมการเกิดการอินเทอร์รัพท์จากภายนอก และยังมีความสามารถด้านจัดการทรัพยากรที่ใช้ภายในอีกด้วย

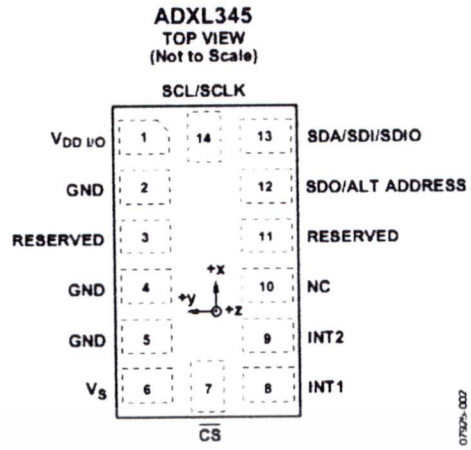


รูปที่ 2.12 โครงสร้างของ ADXL345 [5]

2.6.2 ฟังก์ชั่นการทำงานของขาต่างๆ ของ ADXL345

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 2.13 โครงสร้างของ ADXL345 [5]

ตารางที่ 2.2 ตารางรายละเอียดฟังก์ชั่นขาต่างๆของ ADXL345 [5]

ชื่อ	รายละเอียด	ขา
VDD I/O	ไฟเลี้ยง -0.3 -3.9 volt	1
GND	ขา Ground ต่อสายดิน	2
RESERVED	ขาสำรองไฟ จะต่อเข้า Vs หรือเปิดว่างไว้	3
GND	ขา Ground ต่อสายดิน	4
GND	ขา Ground ต่อสายดิน	5
Vs	ไฟเลี้ยง -0.3 -3.9 volt	6
CS	Chip select	7
INT1	Interrupt 1 Output.	8
INT2	Interrupt 2 Output.	9
NC	ไม่มีการติดต่อ	10
RESERVED	ขาสำรองไฟ จะต่อเข้า Vs หรือเปิดว่างไว้	11
SDO/ALT ADDRESS	Serial Data Output (SPI 4-Wire)/Address สำหรับ I2C	12
SDA/SDI/SDIO	Serial Data(I2C)/Serial Data Input(SPI 4-Wire)/Serial Data Input และ Output (SPI)	13
SCL/SCLK	Serial Data Output (SPI 4-	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
131181

	Wire)/Alternate I2C Address Select (I2C)	

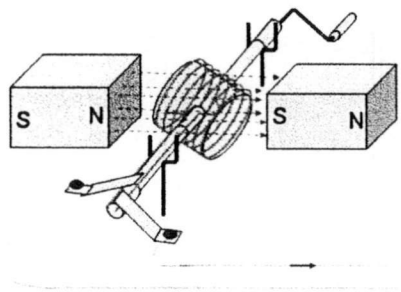
2.6.3 FEATURES

- Ultralow power มีค่าต่ำ 23 μA ในโหมดปกติและ 0.1 μA ในโหมด standby ที่ VS=2.5 V
- Power consumption ควบคุมอัตโนมัติด้วย bandwidth
- ผู้ใช้สามารถเลือกใช้ความละเอียดได้ถึง 16 บิต
- ความละเอียดหลักที่ใช้งาน 10 bit
- Supply voltage range มี 2.0 V ถึง 3.6 V
- I/O voltage range มี 1.7 V ถึง VS
- ส่งข้อมูลผ่าน serial port ใช้ได้ทั้ง SPI (3- และ 4-wire) และ I2C digital interfaces
- Wide temperature อยู่ในช่วง –40°C ถึง +85°C
- 10,000 *g* shock survival
- ขนาดเล็ก 3 mm × 5 mm × 1 mm LGA package
- Bandwidth selectable via serial command
- Flexible interrupt modes mappable to either interrupt pin
- Measurement ranges selectable via serial command
- Wide temperature อยู่ในช่วง –40°C ถึง +85°C

2.7 หลักการพื้นฐานของไดนาโม

เครื่องกำเนิดไฟฟ้าหรือไดนาโม (Generator or dynamo) สามารถเปลี่ยนพลังงานกลเป็นพลังงานไฟฟ้า โดยอาศัยการเหนี่ยวนำในขดลวดของเครื่องกำเนิดไฟฟ้าทำให้เกิดกระแสไฟฟ้าเกิดขึ้น ซึ่งสามารถจัดกระแสไฟได้สองแบบคือ กระแสตรง และกระแสสลับ พลังงานไฟฟ้าจากเครื่องกำเนิดไฟฟ้าเป็นไปตามกฎการอนุรักษ์พลังงาน โดยทั่วไปเครื่องกำเนิดไฟฟ้าจะประกอบด้วยขดลวดเหนี่ยวนำเคลื่อนที่สัมพัทธ์กับแท่งแม่เหล็กทำให้เกิดไฟฟ้า ตัวอย่างการใช้งาน เช่น กรณีไฟฟ้าพลังน้ำ จะมีน้ำตกลงมาหมุนใบพัดของเครื่องกำเนิดไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 การทำงานพื้นฐานของไดนาโม [6]

โดยสามารถคำนวณแรงเคลื่อนไฟฟ้าจากเครื่องกำเนิดไฟฟ้าได้ดังนี้
สมมติ ขดลวดมี N รอบ และแต่ละรอบมีพื้นที่ A ให้ขดลวดหมุนด้วยอัตราเร็วเชิงมุม รอบแกนที่ตั้งฉากกับสนามแม่เหล็ก B แรงเคลื่อนไฟฟ้าคือ

$$E = E_o \sin(w.t) \qquad (2.1)$$

จากสมการที่ (2.1) เมื่อ E_o คือแรงเคลื่อนไฟฟ้าสูงสุด
โดย

$$E_o = NBAw \qquad (2.2)$$

2.7.1 หลักการเหนี่ยวนำให้เกิดกระแสไฟฟ้าของไดนาโม

ไดนาโมเป็นอุปกรณ์ที่ทำหน้าที่เปลี่ยนพลังงานกลให้เป็นพลังงานไฟฟ้า มีส่วนประกอบสำคัญ ได้แก่ ขดลวดที่พันอยู่รอบแกน เรียกว่า อาเมเจอร์ (armature) แม่เหล็ก 2 แท่งหันขั้วต่างกันเข้าหากัน เพื่อให้เกิดสนามแม่เหล็กโดยจะมีเส้นแรงแม่เหล็กพุ่งจากขั้วเหนือไปยังขั้วใต้ และบริเวณขั้วจะมีความเข้มของสนามแม่เหล็กมากกว่าบริเวณอื่นๆ

หลักการเหนี่ยวนำให้เกิดกระแสไฟฟ้าจากไดนาโมอาจทำได้ดังนี้ การหมุนขดลวดตัดสนามแม่เหล็กจะทำให้สนามแม่เหล็กเปลี่ยนแปลงจึงเกิดกระแสไฟฟ้าขึ้น ไมเคิล ฟาราเดย์ (Michael Faraday) นักวิทยาศาสตร์ชาวอังกฤษ (พ.ศ.2334-2410) เป็นผู้ค้นพบหลักการที่ว่า “กระแสไฟฟ้าเหนี่ยวนำเกิดจากการเปลี่ยนแปลงสนามแม่เหล็กที่ผ่านขดลวด” หากต้องการสร้างไดนาโมให้สามารถผลิตกระแสไฟฟ้าได้มากขึ้น สามารถทำได้โดยเพิ่มจำนวนรอบของขดลวด และหมุนขดลวดให้เร็วขึ้น

ไดนาโมสามารถแบ่งออกเป็น 2 ชนิดคือ

1. ไดนาโมไฟฟ้ากระแสสลับ AC Dynamo
   ประกอบด้วยแท่งแม่เหล็ก 2 แท่ง ขดลวดและแหวนลื่นโดย แหวนลื่น 2 วงสัมผัสกับแปรงตัวนำไฟฟ้าซึ่งจะรับกระแสไฟฟ้าจากขดลวดออกสู่วงจรภายนอก
2. ไดนาโมไฟฟ้ากระแสตรง DC Dynamo
   ประกอบด้วยแท่งแม่เหล็ก 2 แท่ง ขดลวดและแหวนแยกโดย แหวนแยกแต่ละอัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัมผัสกับแปรงตัวนำไฟฟ้าซึ่งจะรับกระแสไฟฟ้าจากขดลวดออกสู่วงจรภายนอก โดย ไดนาโมกระแสตรง (Direct current dynamo) หมายถึง ไดนาโมที่ผลิตไฟกระแสตรง (DC) ส่วนประกอบเหมือนกับไดนาโมกระแสสลับทุกอย่าง ต่างกันที่วงแหวนเท่านั้น

รูปที่ 2.15 ไดนาโมจักรยาน [7]

2.8 วงจรไฟฟ้าอนุกรม

สมการแบ่งแรงดัน (Voltage Divider Equation) หมายถึง วงจรอนุกรมที่ใช้ตัวต้านทานอย่างน้อย 2 ตัวเป็นตัวแบ่งแรงดันไฟฟ้า

$$V_{Rn} = V_T \frac{R_n}{R_T} \quad (2.3)$$

โดย V_{Rn} คือ แรงดันตกคร่อมตัวต้านทานใดๆ
V_T คือ แรงดันของแหล่งจ่ายไฟฟ้า
R_n คือ ตัวต้านทานใดๆ ในวงจร
R_T คือ ค่าความต้านทานรวมของวงจร

2.9 โปรแกรมออกแบบกราฟิก Blender

โปรแกรม Blender เป็นโปรแกรมที่นำเอาความสามารถต่างๆในการสร้างสรรค์ไฟล์รูปแบบสามมิติมารวมกัน เช่น การขึ้นรูปโมเดล , เรนเดอร์ , แอนิเมชั่น ฯลฯ โดย Blender สามารถทำงานได้ในระบบปฏิบัติการที่หลากหลาย และมีขนาดไฟล์ติดตั้งที่ไม่มากนักเมื่อเทียบกับโปรแกรมกราฟิกสามมิติตัวอื่นๆ เนื่องจาก Blender มีรูปแบบซอฟต์แวร์แบบ Open Source เพราะตัวดั้งเดิมถูกสร้างขึ้นโดยบริษัท Not a Number หรือที่รู้จักกันในชื่อของ NaN ในปัจจุบันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	G5,Powerbook G4,iMac G5 Linux glibc 2.3.6,includes FFMPG Suits
- ความเร็วซีพียู	700 MHz หรือสูงกว่า
- RAM	512 MB
- พื้นที่สำหรับลงโปรแกรม	27 MB
- การ์ดแสดงผล	1024x768 16-bit color ,RAM 8 MB (แนะนำให้รองรับมาตรฐาน OpenGL)
- ระบบและอุปกรณ์อื่นๆ	เมาส์แบบ 3 ปุ่ม

2.9.3 ฟังก์ชันทางกราฟิก (Graphic Functions)

การทำงานของระบบกราฟิก ผู้ใช้ไม่จำเป็นต้องทราบว่าการทำงานต้องทำอย่างไร รู้เพียงแต่อินพุต (input) และเอาต์พุต (output) ก็เพียงพอ โดยในระบบกราฟิกจะมีฟังก์ชันที่ถูกเรียกใช้ โดยมีการรับค่าของอินพุตจากอุปกรณ์ต่างๆ เช่น เมาส์หรือคีย์บอร์ด หรืออุปกรณ์อื่นๆ เช่น แมสเสจจากระบบปฏิบัติการ หรืออินเทอร์รัพท์ก็ได้ ผลลัพธ์ที่ได้จากการทำงานจะแสดงออกไปยังอุปกรณ์ในการแสดงผล เช่น จอภาพ CRT หรือจอ LCD ทั้งนี้เอาต์พุตอาจจะมองอินพุตเป็นฟังก์ชันที่ผู้ใช้เรียกใช้และเอาต์พุตคือ ผลของการทำงานที่แสดงไปยังจอภาพ ดังรูปด้านล่าง

รูปที่ 2.15 ระบบกราฟิกที่เปรียบเสมือนกล่องดำ (Black Box) [8]

2.10 โปรแกรมภาษา Python

Python คือชื่อภาษาที่ใช้ในการเขียนโปรแกรมภาษาหนึ่งที่มีอยู่ในตอนนี้ โดยความสามารถของภาษานี้สูงไม่แพ้ภาษาอื่นๆ Python นั้นเป็นภาษาชนิด Open Source ทำให้ทุกคนสามารถที่จะนำ Python มาพัฒนาโปรแกรมได้ฟรีๆโดยไม่ต้องเสียค่าใช้จ่าย เนื่องจากความเป็น Open Source จึงทำให้มีคนเข้ามาช่วยกันพัฒนาให้ Python มีความสามารถสูงขึ้น และสามารถใช้งานได้ครอบคลุมกับทุกลักษณะงาน

Python ถูกสร้างขึ้นมาโดย Guido Van Rossumโค้ดของ Python ถูกสร้างขึ้นมาจากภาษาซี การประมวลผลจะทำในแบบอินเทอร์พรีเตอร์คือจะประมวลผลไปทีละบรรทัดและปฏิบัติตามคำสั่งที่ได้รับ Python เวอร์ชันแรกคือเวอร์ชัน 0.9.0 ออกมาเมื่อปี 2533 และเวอร์ชันปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือ 2.5.2 ออกเมื่อวันที่ 21 กุมภาพันธ์ 2551 และได้มีการพัฒนา Python ในรุ่นที่ 3 คือ Python 3000 หรือ Py3k โดยจะมีการปรุงปรุงใหม่เกือบหมด และตอนนี้อยู่ในระหว่างการทดลองอยู่

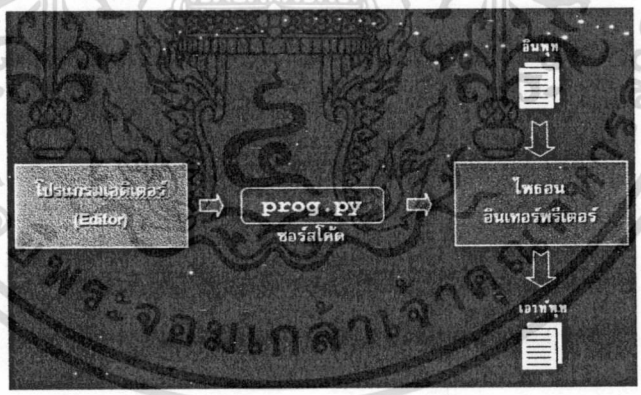
2.10.1 จุดเด่นของโปรแกรมภาษา Python

โปรแกรมภาษา Python เป็นโปรแกรมภาษาแบบ Open Source ที่สามารถดาวน์โหลดมาใช้งานได้ฟรี และยังมีความสามารถทัดเทียมกับโปรแกรมภาษาอื่นๆโดยจุดเด่นที่น่าสนใจของโปรแกรมมีดังนี้

- ภาษา Python สนับสนุนแนวคิดแบบออปเจกต์โอเรียนเทด หรือ OOP (Object Oriented Programming)
- Python เป็นภาษาคอมพิวเตอร์ที่ไม่คิดมูลค่าการใช้งานและเป็นภาษาที่มีความยืดหยุ่นสูงมาก
- โค้ดที่เขียนด้วย Python สามารถนำไปรันบนระบบปฏิบัติการอื่นๆ ได้ เช่น Linux, Ms-windows (95, 98, NT, 2000, XP), Amiga, Be-OS, OS/2, VMS, QNX, และระบบอื่นๆอีกมากมาย
- Python สนับสนุนเทคโนโลยี COM ของ Ms-windows
- Python รวมมาตรฐานการอินเตอร์เฟส Tkinter ซึ่งสนับสนุนบนระบบ Windows, Ms-windows และ Macintosh การใช้คำสั่ง Tkinter API ช่วยให้โปรแกรมเมอร์ไม่ต้องแก้ไขโค้ดเมื่อนำไปรันบนระบบปฏิบัติการอื่นๆ
- Python เป็น Dynamic typing คือ สามารถเปลี่ยนชนิดข้อมูลได้ง่ายและสะดวก
- Python มี Buil-in Object Types คือ โครงสร้างของข้อมูลที่สามารถใช้ได้ใน Python ประกอบด้วย ลิสต์, ดิกชันนารี, สตริง ที่ง่ายต่อการใช้งานและมีประสิทธิภาพสูง
- Pythonมีเครื่องมือต่างๆ มากมาย เช่น การประมวลผลเท็กซ์ไฟล์ การเรียงข้อมูล การเชื่อมต่อสตริง การตรวจสอบเงื่อนไขของข้อความ การแทนคำ เป็นต้น
- Python มีโมดูลสำหรับจัดการ Regular Expresion
- Python มีโมดูลที่สร้างขึ้นจากนักพัฒนาสนับสนุนมากมาย ได้แก่ COM, Image, CORBA, ORBs, XML เป็นต้น
- Python จัดการหน่วยความจำอย่างอัตโนมัติ สามารถจัดการพื้นที่หน่วยความจำที่ไม่ต่อเนื่องให้ทำงานได้อย่างมีประสิทธิภาพ
- Python อนุญาตให้ฝังชุดคำสั่งของ Python เอาไว้ภายในโค้ดภาษา C/C++ ได้
- Python อนุญาตให้โปรแกรมเมอร์สร้าง Dynamic Link Library (DLL) เพื่อใช้ร่วมกับ Python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Python มีโมดูลสนับสนุนเกี่ยวกับเน็ตเวิร์ก โปรเซส เธรด regular, expression, xml, GUI และอื่นๆ
- Pythonประกอบด้วยโมดูลสำหรับสร้าง Internet Script และติดต่อกับอินเทอร์เน็ตผ่าน Sockets, และทำหน้าที่เป็น CGI Script ตลอดจนใช้งานคำสั่ง FTP, Glopher, XML และอื่นๆอีกมาก
- Python สามารถประมวลผลทางด้านวิทยาศาสตร์ และวิศวกรรมศาสตร์ได้อย่างมีประสิทธิภาพ
- Python มีฟังก์ชันสนับสนุนฐานข้อมูล เช่น MySQL, Sybase, Oracle, Informix, ODBC และอื่นๆ
- Python มีไลบรารีสนับสนุนด้านการสร้างภาพกราฟิก เช่น ทำภาพเบลอ หรือภาพชัด หรือเขียนข้อความบนภาพ ตลอดจนบันทึกไฟล์ในรูปแบบต่างๆ ได้อย่างสะดวกและมีประสิทธิภาพ
- Python มีไลบรารีสนับสนุนด้านปัญญาประดิษฐ์
- Python มีไลบรารีสำหรับสร้างเอกสาร PDF โดยไม่ต้องติดตั้ง Acrobat Writer
- Python มีไลบรารีสำหรับสร้าง Shockwaves Flash (SWF) โดยไม่ต้องติดตั้ง Macromedia Flash



รูปที่ 2.16 ขั้นตอนการทำงานของโปรแกรม Python [9]

2.10.2 โปรแกรมภาษา Python ในแพลตฟอร์มต่างๆ

ผู้เขียนโปรแกรมภาษา Python สามารถเลือกใช้แพลตฟอร์มได้ดังนี้

CPython (ซีไพทอน) : เป็นแพลตฟอร์มภาษาไพทอนดั้งเดิม โปรแกรมอินเทอร์พรีเตอร์ถูกเขียนโดยภาษาซี ซึ่งคอมไพล์ใช้ได้บนหลายระบบปฏิบัติการ เช่น Windows,Linux,Unix การใช้งานสามารถทำได้โดยการติดตั้งโปรแกรมอินเทอร์พรีเตอร์และแพ็กเกจที่จำเป็นต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Jython (ไจทอน) : เป็นแพลตฟอร์มภาษาไพทอนที่ถูกพัฒนาบนแพลตฟอร์มจาวา เพื่อเพิ่มอำนวยความสะดวกในการใช้ความสามารถภาษาสคริปต์ของไพทอนลงในซอฟต์แวร์จาวาอื่นๆ การใช้งานสามารถทำได้โดยการติดตั้งจาวาและเรียนไลบรารีของไจทอนซึ่งมาในรูปไบนารีเพื่อใช้งาน

Python.NET (ไพทอนดอตเน็ต) : เป็นการพัฒนาภาษาไพทอนให้สามารถทำงานบนดอตเน็ตเฟรมเวิร์กของไมโครซอฟท์ได้ โดยโปรแกรมที่ถูกเขียนจะถูกแปลงเป็น CLR ปัจจุบันมีโครงการที่นำภาษาไพทอนมาใช้บน .NET Framework ของไมโครซอฟท์แล้วคือโครงการ IronPython

2.10.3 ไลบรารีภาษา Python

การเขียนโปรแกรมในภาษาไพทอนโดยใช้ไลบรารีต่างๆ เป็นการลดภาระของโปรแกรมเมอร์ได้เป็นอย่างดี ทำให้โปรแกรมเมอร์ไม่ต้องเสียเวลากับการเขียนคำสั่งที่ซ้ำๆ เช่นการแสดงผลข้อมูลออกสู่หน้าจอ หรือการรับค่าต่างๆ

ไพทอนมีชุดไลบรารีมาตรฐานมาให้ตั้งแต่อินเตอร์พรีเตอร์ นอกจากนั้นยังมีผู้พัฒนาจากทั่วโลกดำเนินการพัฒนาไลบรารีซึ่งช่วยอำนวยความสะดวกในด้านต่างๆ โดยจะเผยแพร่ในรูปแบบของแพ็กเกจต่างๆ ซึ่งสามารถติดตั้งเพิ่มเติมได้ ดังนี้

- wxPython : อีกทางเลือกหนึ่งสำหรับเขียนส่วนติดต่อกับผู้ใช้กับกราฟิก ซึ่งสามารถใช้ได้หลายระบบปฏิบัติการ
- SciPy : รวมโครงสร้างข้อมูลและการคำนวณต่าง ๆ ที่จำเป็นต้องใช้ในการเขียนโปรแกรมคำนวณทางวิทยาศาสตร์
- py2exe : ใช้สำหรับแปลงโปรแกรมที่เขียนในภาษาไพทอนให้อยู่ในรูปแบบของระบบปฏิบัติการวินโดวส์
- PyWin32 : ใช้สำหรับติดต่อเรียกใช้บริการบนระบบปฏิบัติการวินโดวส์และคลาสใน Microsoft Foundation Classes: MFC
- MySQLdb : ใช้สำหรับติดต่อกับระบบฐานข้อมูล MySQL
- psycopg2 : ใช้สำหรับติดต่อกับระบบฐานข้อมูล โพสต์เกรสคิวเอล
- PyGTK : GTK+ สำหรับ Python ใช้สำหรับสร้างส่วนติดต่อกับผู้ใช้แบบกราฟิก ซึ่งสามารถใช้ได้หลายระบบปฏิบัติการ
- PyQt : Qt สำหรับ Python ใช้สำหรับสร้างส่วนติดต่อกับผู้ใช้แบบกราฟิก ซึ่งสามารถใช้ได้หลายระบบปฏิบัติการ

2.10.4 ตัวอย่างการนำไปใช้ของโปรแกรมภาษา Python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

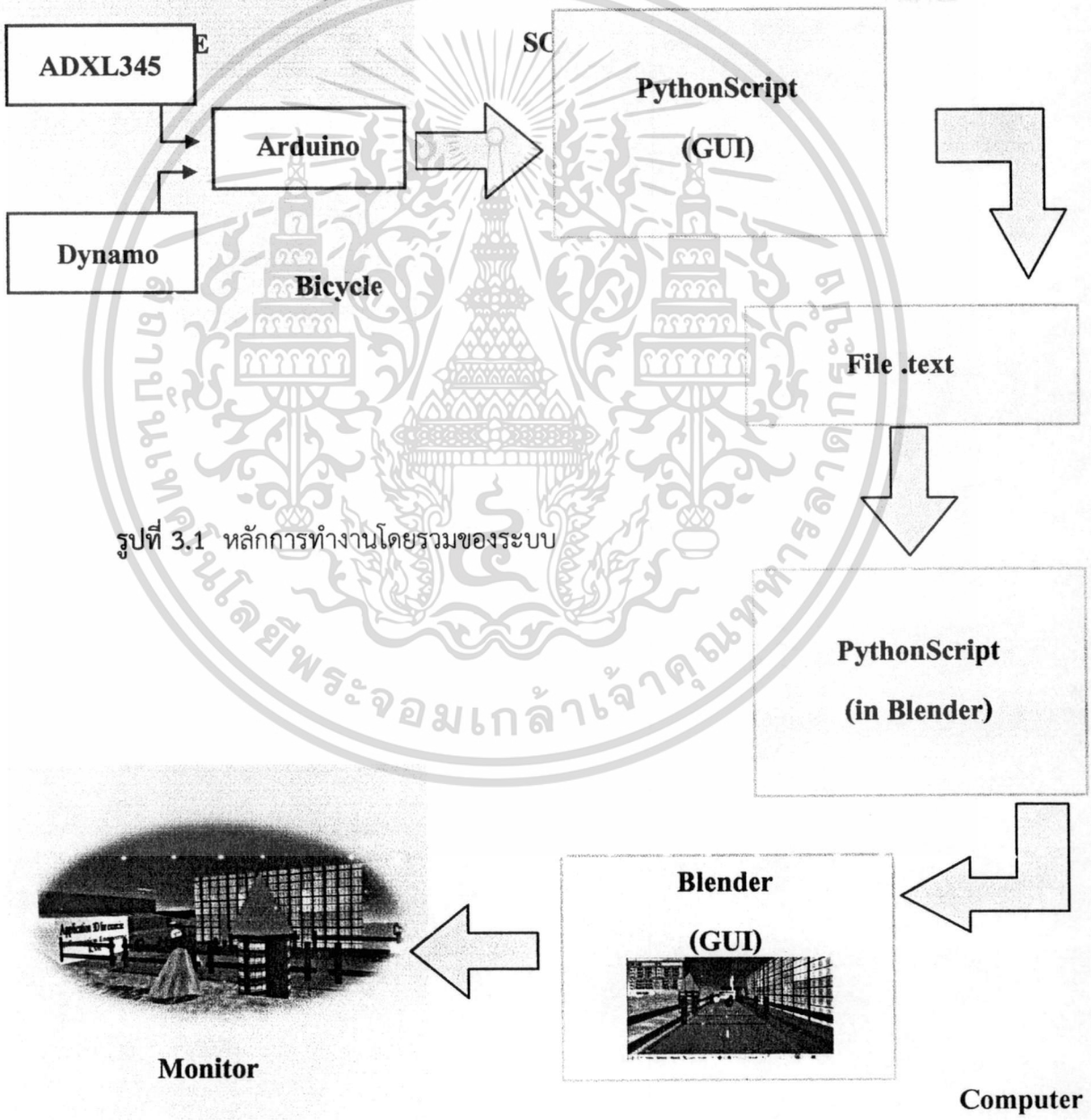
จะเห็นได้ว่าจุดเด่นของโปรแกรมภาษา Python มีอยู่มากและใช้งานได้อย่างรวดเร็ว โดยโปรแกรมภาษา Python ได้ถูกนำไปใช้ประโยชน์ในต่างๆมากมาย เช่น

- Red Hat ใช้ Python เป็นเครื่องมือสำหรับการติดตั้งระบบปฏิบัติการ Linux (Installer)
- Google สร้างขึ้นด้วย Python
- Infoseek ใช้ Python จัดการ web search products
- Yahoo! ใช้ Python ในการจัดการด้าน Internet services
- NASA ใช้ Python สำหรับ mission-control-system
- Lawrence Livermore Labs ใช้ Python สำหรับงาน numeric programming
- Industrial Light and Magic ใช้ Python สร้างภาพแอนิเมชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการสร้าง

3.1 หลักการทำงานของระบบโดยรวม



รูปที่ 3.1 หลักการทำงานโดยรวมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.1 การทำงานของระบบโดยรวมจะแบ่งเป็น 2 ส่วนใหญ่ๆ คือ ส่วนของชุดอุปกรณ์และส่วนของโปรแกรมกราฟิกสามมิติ

โดยการทำงานคือ ชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino จะทำหน้าที่ควบคุมการทำงานของอุปกรณ์ ADXL345(Accelerometer) เซนเซอร์วัดความเร่งแนว 3 แกน (แกน X, Y, Z) ที่ใช้สำหรับตรวจวัดการเคลื่อนไหวศีรษะของผู้ออกกำลังกายเพื่อให้สามารถบังคับซ้ายขวาตามความต้องการของผู้เล่นได้ และสำหรับการเคลื่อนที่ภายในเกมจะเคลื่อนที่ตามการปั่นจักรยานของผู้ออกกำลังกาย โดยมีไดนาโมจักรยานทำหน้าที่ส่งค่าอนาล็อกให้กับชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Ardunio เมื่อชุดโมดูลบอร์ดดังกล่าวได้รับค่าข้อมูลจึงจะส่งค่าไป(โดยซอฟแวร์ที่ใช้เป็นภาษา PythonScript และค่าที่ได้รับมาจะมี 4 ค่านั่นคือ X, Y, Z และ Dynamo) เก็บไว้ในรูปแบบไฟล์ .text เพื่อให้สามารถนำค่าข้อมูลดังกล่าวไปประมวลผลในส่วนของโปรแกรมกราฟิกสามมิติที่ถูกออกแบบขึ้น โดยใช้โปรแกรม Blender3D และควบคุมการใช้งานต่างๆของเกมด้วยโปรแกรมภาษา PythonScript และส่งค่าให้กับ Monitor แสดงผลการทำงานของโปรแกรมกราฟิกสามมิติต่อไป

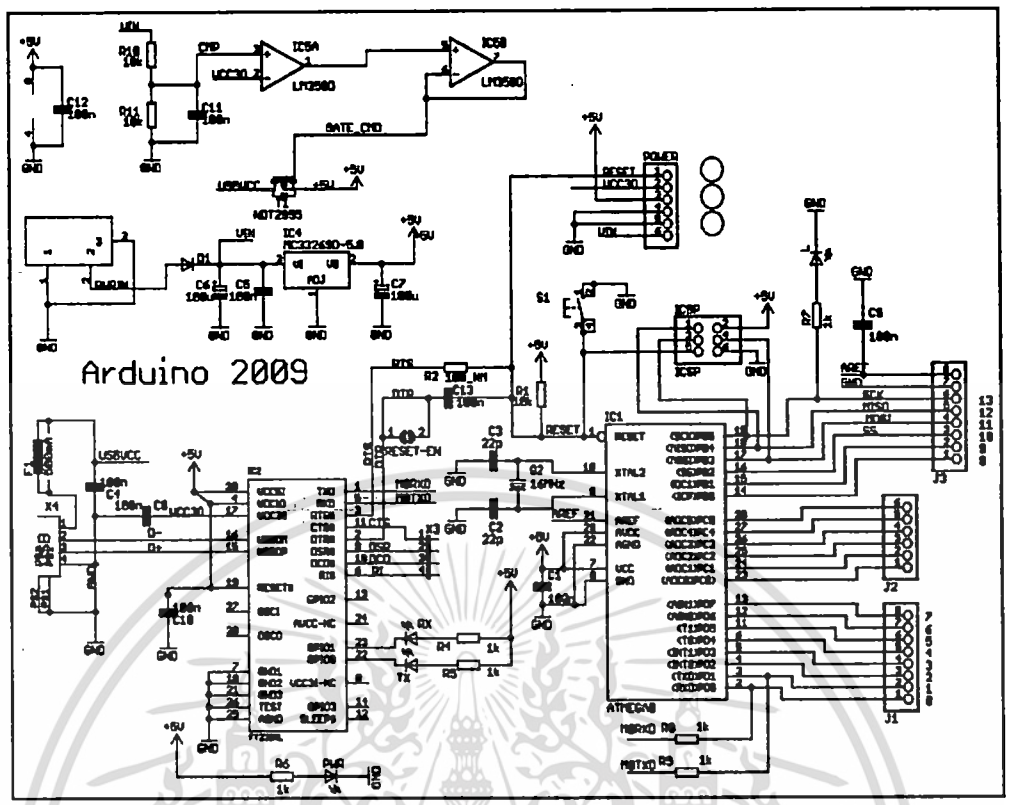
3.2 ส่วนของชุดอุปกรณ์

อุปกรณ์จะต้องอาศัยการติดต่อกันระหว่าง ชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino, ADXL345(Accelerometer) และไดนาโมจักรยาน โดยหลักการต่างๆมีดังต่อไปนี้

3.2.1 ชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino

ชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino เป็นบอร์ดทดลองสำเร็จรูปโดยใช้ MCU ในตระกูล AVR เบอร์ ATmega 328 สามารถส่งข้อมูลได้รวดเร็วและมีช่องสำหรับต่อกับอุปกรณ์ภายนอกได้อย่างสะดวก อีกทั้งสามารถใช้ไฟเลี้ยงจากการต่อพอร์ตUSB ได้อีกด้วย

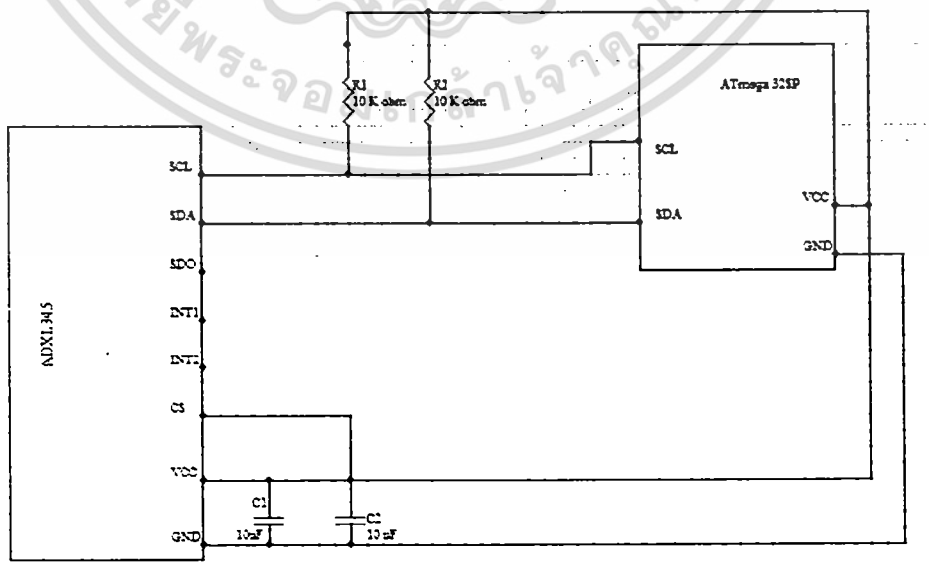
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 วงจรภายในชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove [1]

3.2.2 ADXL345 (Accelerometer)

การติดต่อเซนเซอร์ ADXL345 กับบอร์ด Arduino Duemilanove ใช้หลักการส่งข้อมูลแบบ I2C Bus ดังนั้นการออกแบบวงจรของเซนเซอร์จึงใช้เพียงสาย SDA กับ SCL ในการส่งข้อมูลระหว่างบอร์ด Arduino Duemilanove โดยใช้ไฟเลี้ยง 3.3 v จากบอร์ดได้ด้วยเช่นกัน



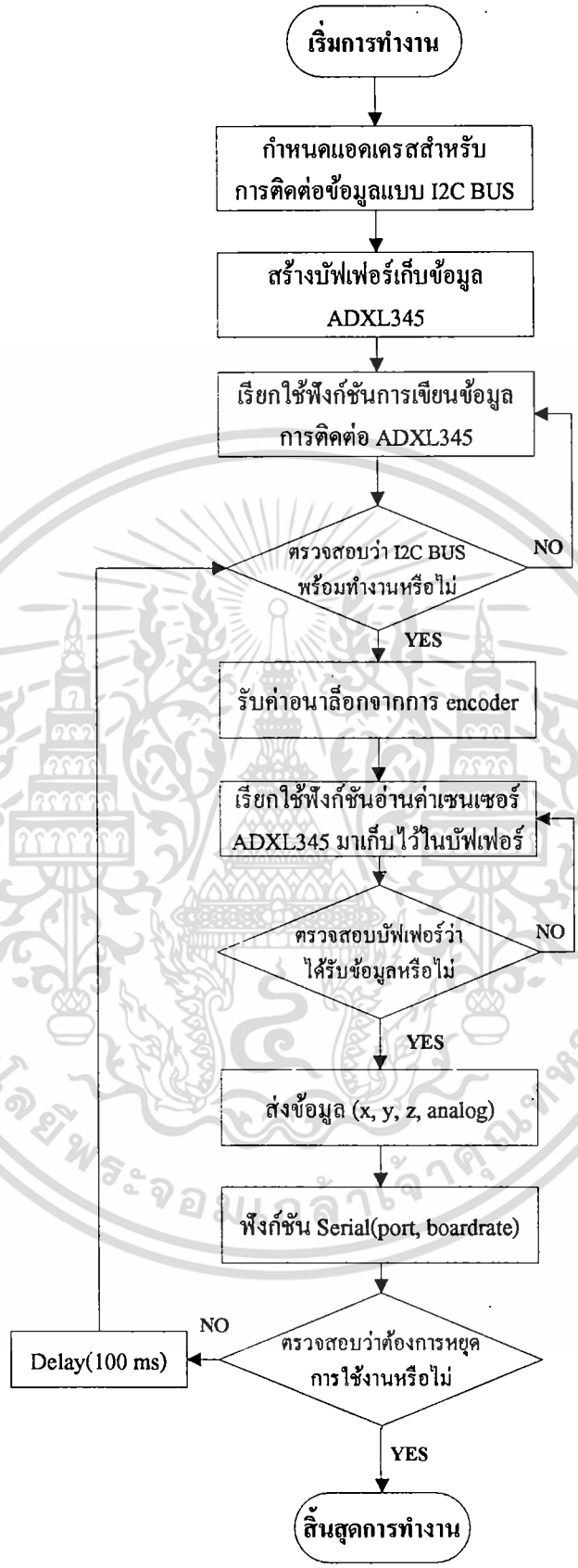
รูปที่ 3.3 การออกแบบการเชื่อมต่อข้อมูลแบบ I2C Bus กับบอร์ดทดลอง Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 หลักการทำงานของ Arduino ในการส่งข้อมูล

1. กำหนดแอดเดรสสำหรับการติดต่อข้อมูลของ Arduino โดยอาศัยการส่งแบบ I2C Bus ให้กำหนดพอร์ตอนาล็อกสำหรับรับค่าอินพุทจากไดนาโมจักรยาน
2. สร้างบัฟเฟอร์ขึ้นเพื่อสำหรับเก็บข้อมูลที่ได้รับจาก ADXL345(Accelerometer) โดยข้อมูลดังกล่าวจะถูกนำเก็บเพื่อนำไปใช้งาน
3. เรียกใช้ฟังก์ชันการเขียนข้อมูลเพื่อให้อุปกรณ์โมดูลบอร์ด Arduino ติดต่อกับ ADXL345
4. ตรวจสอบว่าการส่งข้อมูลในรูปแบบ I2C BUS พร้อมทำงานหรือไม่
    - ถ้าการส่งข้อมูลในรูปแบบ I2C BUS ไม่พร้อมทำงานจะทำการเรียกใช้ฟังก์ชันการเขียนข้อมูลติดต่อ ADXL345 (ในข้อ 3) อีกครั้ง
    - ถ้าการส่งข้อมูลในรูปแบบ I2C BUS พร้อมทำงานจะรับค่าอนาล็อกจากไดนาโมจักรยาน เพื่อส่งข้อมูลดังกล่าวต่อไป
5. ทำการรับข้อมูลชนิดอนาล็อกจากไดนาโมจักรยาน
6. เรียกใช้ฟังก์ชันการอ่านค่าข้อมูลจากเซนเซอร์ ADXL345 (ในหัวข้อ 3.2.4) และนำค่าข้อมูลดังกล่าวมาเก็บไว้ในบัฟเฟอร์
7. ตรวจสอบบัฟเฟอร์ว่าได้รับข้อมูลเรียบร้อยหรือไม่
    - ถ้าตรวจสอบบัฟเฟอร์พบว่ายังไม่มีการรับข้อมูล จะย้อนกลับไปเพื่อเรียกใช้ฟังก์ชันการอ่านค่าข้อมูลจากเซนเซอร์ ADXL345 มาเก็บในบัฟเฟอร์ (ในข้อ 6) อีกครั้ง
    - ถ้าตรวจสอบบัฟเฟอร์พบว่ามีการส่งข้อมูลเกิดขึ้น จะส่งข้อมูล (x, y, z, Analog) และฟังก์ชัน Serial (Port, Board rate) ออกไปใช้งาน
8. ตรวจสอบว่าต้องการหยุดการใช้งานหรือไม่
    - ถ้าตรวจสอบว่าไม่ต้องการหยุดการใช้งาน จะเพิ่มค่า delay เป็นขนาด 100 ms
    - ถ้าตรวจสอบว่าต้องการหยุดการใช้งาน จะสิ้นสุดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



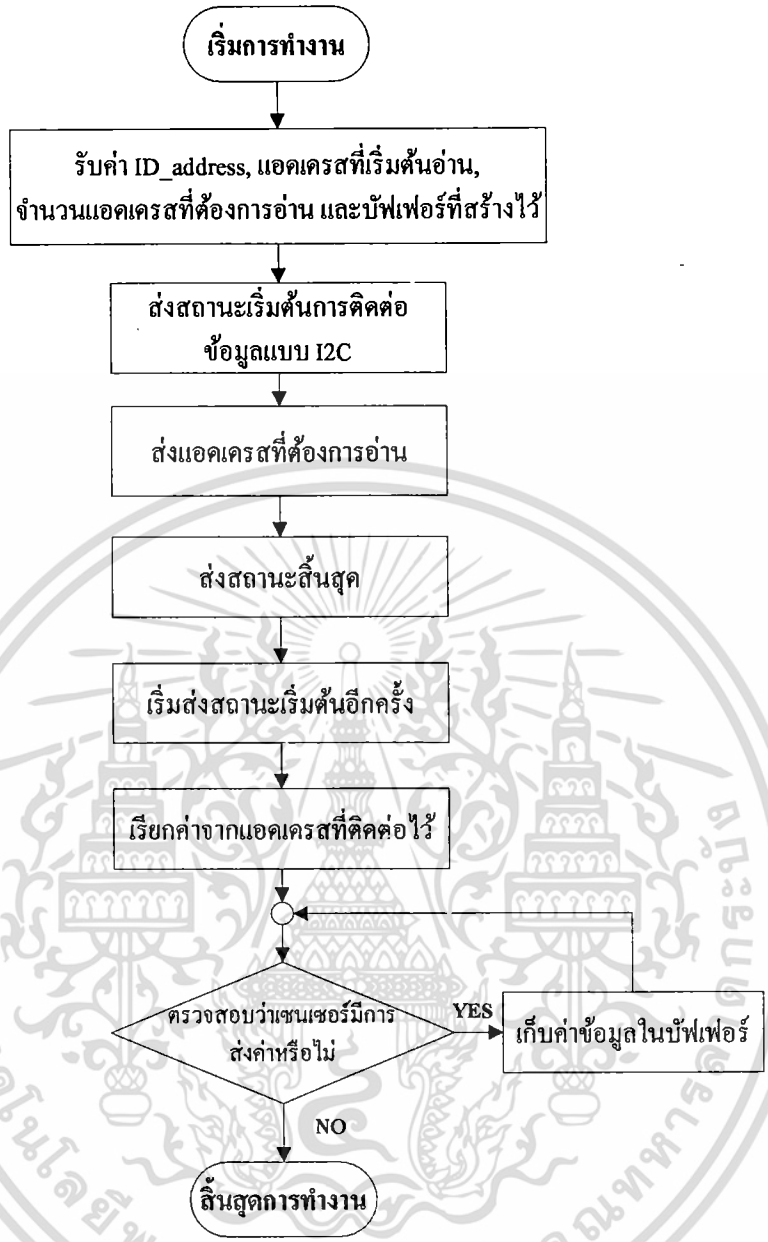
รูปที่ 3.4 Flow chat หลักการทำงานของ Arduino ในการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 หลักการทำงานของ Arduino ในการอ่านค่าข้อมูลจากเซนเซอร์ ADXL345

1. เริ่มต้นการทำงาน โดยรับค่า ID_Address, แอดเดรสที่เริ่มต้นอ่าน, จำนวนแอดเดรสที่ต้องการอ่าน และบัฟเฟอร์ที่สร้างไว้เพื่อเข้ามารองรับแอดเดรสที่อ่านได้
2. ส่งสถานะเริ่มต้นการติดต่อข้อมูลแบบ I2C
3. ส่งแอดเดรสที่ต้องการอ่าน
4. ส่งสถานะสิ้นสุด
5. เริ่มส่งสถานะเริ่มต้นใหม่อีกครั้ง
6. เรียกค่าจากแอดเดรสที่ติดต่อไว้
7. ตรวจสอบว่าเซนเซอร์ส่งค่าข้อมูลหรือไม่
   - ถ้าตรวจสอบเซนเซอร์พบว่ามีการส่งค่าข้อมูล จะทำการส่งค่าไปเก็บในบัฟเฟอร์ที่สร้างไว้ และกลับไปตรวจสอบเซนเซอร์อีกครั้ง
   - ถ้าตรวจสอบเซนเซอร์พบว่าไม่มีการส่งค่าข้อมูล จะสิ้นสุดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 Flow chat หลักการอ่านค่าข้อมูลของเซนเซอร์ ADXL345

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 หลักการทำงานของ Arduino ในการเขียนค่าข้อมูลไปยังเซนเซอร์ ADXL345

1. เริ่มต้นการทำงานโดยรับค่า ID_address ของเซนเซอร์, แอดเดรสที่ต้องการติดต่อ และค่าที่ต้องการเขียน
2. กำหนดสถานะเป็นเริ่มต้น เพื่อเริ่มการติดต่อสื่อสารระหว่าง Arduino และ ADXL345
3. ส่งรหัส Control Byte ของ ADXL345 ไปที่ (0x53)
4. ส่งค่าตำแหน่งแอดเดรสที่ต้องการเขียนข้อมูลติดต่อภายใน slave
5. ส่งค่าข้อมูลเพื่อปรับค่าใน ADXL345
6. ส่งสถานะสิ้นสุดการทำงาน

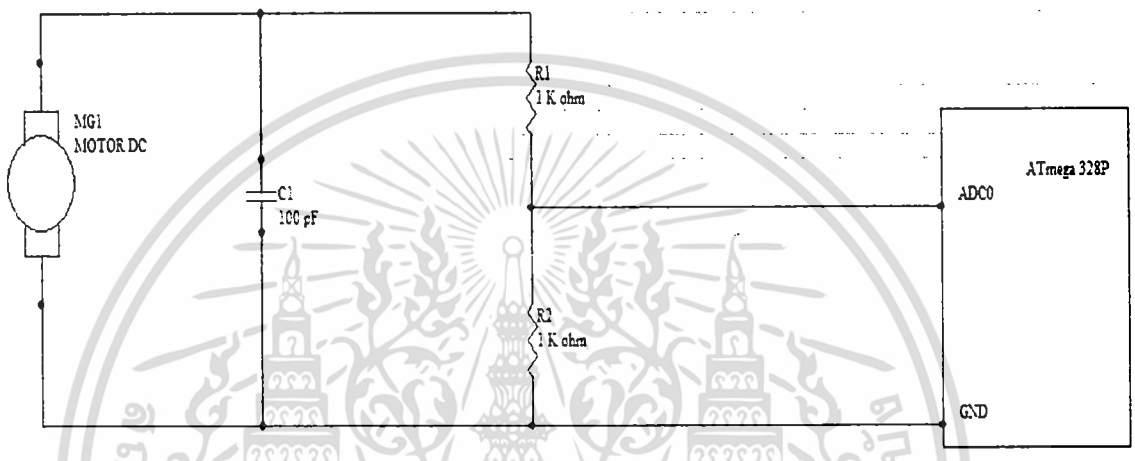
รูปที่ 3.6 Flow chat หลักการเขียนค่าข้อมูลของเซนเซอร์ ADXL345

3.2.6 หลักการทำงานของ Arduino ในการอ่านค่าข้อมูลที่ได้รับจากไดนาโมจักรยาน

ในส่วนนี้เป็นการออกแบบการติดต่อระหว่าง Arduino และไดนาโม หลักการคือ ใช้ตัวเก็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประจุแบบไม่มีขั้วต่อคร่อมกันระหว่างไดนาโมจักรยาน เพื่อป้องกันแรงดันที่อาจมีค่าสูง และใช้ตัวต้านทานสองตัวซึ่งมีค่าตัวละ 1K ต่อแบบอนุกรม เพื่อช่วยในการลดค่าแรงดันจากไดนาโมจักรยานให้เหลือเพียง 5 โวลต์ เหตุผลที่สร้างวงจรดังกล่าว เนื่องจากหากปั่นจักรยานจะเกิดการหมุนของไดนาโมจักรยานเกิดขึ้น เมื่อความเร็วในการปั่นสูงขึ้นจะเกิดแรงดันที่ทำให้ส่งเข้าไปในบอร์ดไมโครคอนโทรลเลอร์ Arduino สูงเกินกว่าบอร์ดไมโครคอนโทรลเลอร์จะรับได้ และส่งผลต่ออาจทำให้บอร์ดดังกล่าวเสียหาย



รูปที่ 3.7 วงจรไดนาโมจักรยานต่อเข้าสู่บอร์ด Arduino

3.3 ส่วนของโปรแกรมกราฟิกสามมิติ

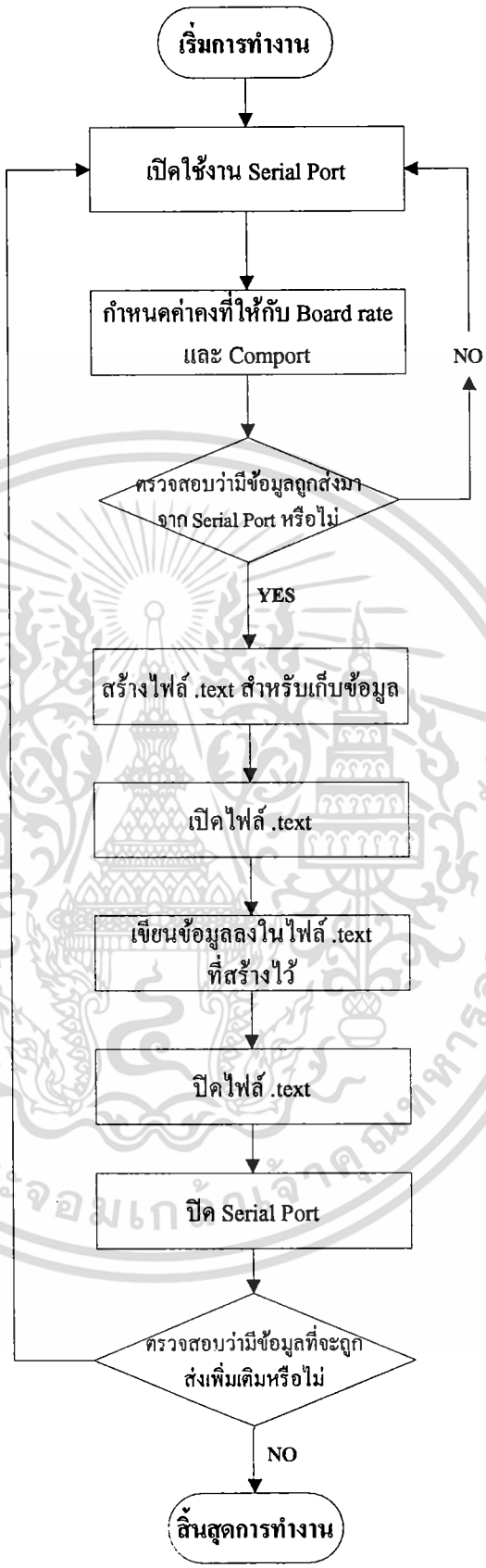
3.3.1 ส่วนของการรับข้อมูลของโปรแกรมภาษา Python จากไมโครคอนโทรลเลอร์บอร์ด Arduino โดยส่งผ่านทาง Serial Port

1. เริ่มต้นการทำงานโดยเปิดใช้งาน Serial Port
2. กำหนดค่าคงที่ให้กับ Board rate และ Comport สำหรับรับค่าจาก Serial Port
3. ตรวจสอบว่ามีข้อมูลถูกส่งมาจาก Serial Port หรือไม่
   - ถ้าตรวจสอบพบว่าส่งข้อมูลไม่ถูกส่งมา จะทำการส่งค่าว่างเปล่า(ค่า null)กลับไปเพื่อรอรับข้อมูลใหม่ที่จะเข้ามา (ในข้อ 1) อีกครั้ง
   - ถ้าตรวจสอบพบว่ามีการส่งข้อมูลเข้ามา จะนำค่าข้อมูลดังกล่าวเก็บไว้ในไฟล์ .text
4. สร้างไฟล์ .text เพื่อเก็บค่าข้อมูลที่ถูกส่งมาจาก Serial Port
5. เปิดไฟล์ .text
6. เขียนข้อมูลลงในไฟล์ .text ที่สร้างไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ปิดไฟล์ .text
8. ปิด Serial Port
9. ตรวจสอบว่าต้องการรับค่าจาก Serial Port เพิ่มเติมหรือไม่
    - ถ้าตรวจสอบพบว่าต้องการรับค่าข้อมูลเพิ่มเติมจะทำการวนลูป(ไปที่ข้อ 1) เพื่อทำการรับค่าข้อมูลใหม่
    - ถ้าตรวจสอบพบว่าไม่ต้องการรับค่าข้อมูลเพิ่มเติม จะสิ้นสุดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

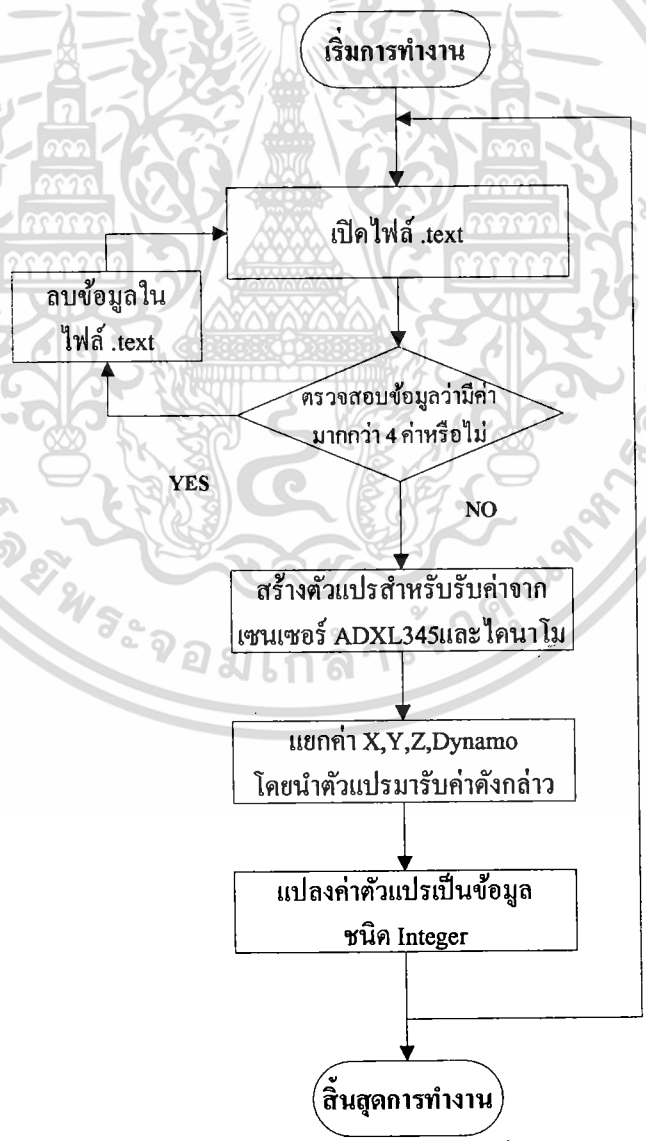


รูปที่ 3.8 Flow chat ส่วนของการรับข้อมูลโปรแกรมภาษา Python รับค่าข้อมูลจาก ไมโครคอนโทรลเลอร์บอร์ด Arduino โดยส่งผ่านทาง Serial Port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 ส่วนของการแยกค่าข้อมูลที่ส่งมาไว้ในไฟล์ .text

1. เปิดไฟล์ .text ที่ได้จัดเก็บค่าไว้
2. ตรวจสอบข้อมูลที่เข้ามาว่ามีค่ามากกว่า 4 ค่าต่อการส่งข้อมูล 1 ครั้งหรือไม่
   - ถ้าตรวจสอบพบว่าค่าที่ส่งมามีค่ามากกว่า 4 ค่า จะทำการลบข้อมูลในไฟล์ .text และรอค่าข้อมูลใหม่
   - ถ้าตรวจสอบพบว่าค่าที่ส่งมามีค่าเท่ากับ 4 ค่า จะทำการกำหนดตัวแปรมารับค่าดังกล่าวเป็นค่า dX, dY, dZ, Dynamo
3. สร้างตัวแปรสำหรับรับค่าข้อมูลจากเซนเซอร์และไดนาโมจักรยาน
4. แยกค่า X, Y, Z, Dynamo ออกจากกันและนำตัวแปรที่สร้างขึ้นมารับค่าข้อมูล
5. แปลงค่าข้อมูลดังกล่าวเป็น Integer

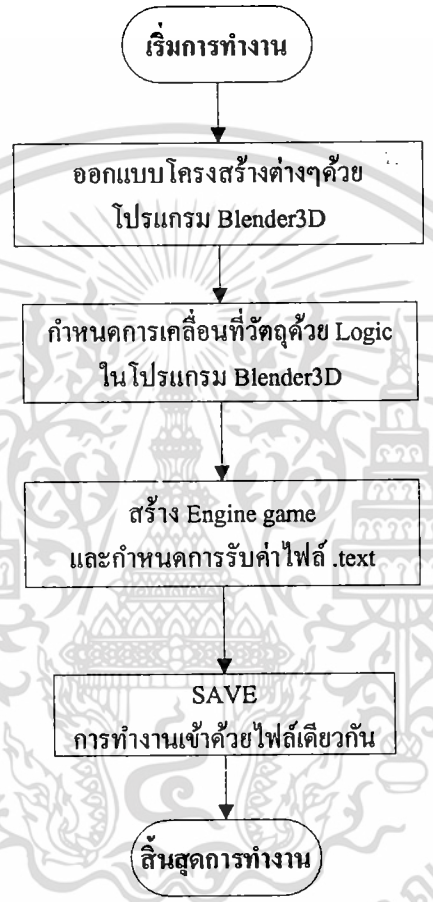


รูปที่ 3.9 Flow chat ส่วนของการแยกค่าข้อมูลที่ส่งมาไว้ในไฟล์ .text

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 ส่วนของการออกแบบรูปทรงสามมิติ

1. ออกแบบโครงสร้างต่างๆด้วยโปรแกรม Blender3D
2. กำหนดการเคลื่อนที่เบื้องต้นด้วย Logic ในโปรแกรม Blender3D
3. สร้าง Engine game และกำหนดให้มีการรับค่าจากไฟล์ .text ด้วยโปรแกรมภาษา Python
4. Save โปรแกรม Blender3D และโปรแกรมภาษา Python ลงในไฟล์เดียวกัน



รูปที่ 3.10 Flow chat ส่วนของการออกแบบรูปทรงสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

เนื้อหาในบทนี้จะกล่าวถึงรายละเอียดขั้นตอนการทดลองอ่านค่าข้อมูลจากเซนเซอร์วัดความเร่ง ADXL345 และไดนาโมจักรยาน โดยใช้บอร์ดไมโครคอนโทรลเลอร์ Arduino จากนั้นจึงส่งข้อมูลไป ประมวลผลยัง โปรแกรมกราฟิกสามมิติ (Blender3D) และควบคุมการทำงานโดยโปรแกรมภาษา Python โดยมีรายละเอียดการทดลองดังนี้

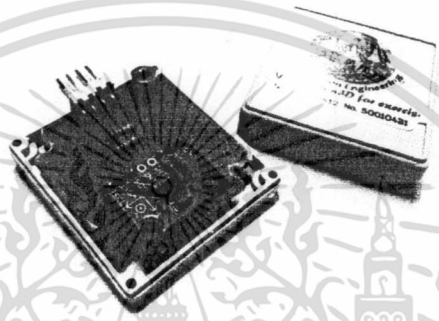
4.1 การทำงานของอุปกรณ์โดยรวม

4.1.1 ส่วนของอุปกรณ์

รูปที่ 4.1 ส่วนของอุปกรณ์

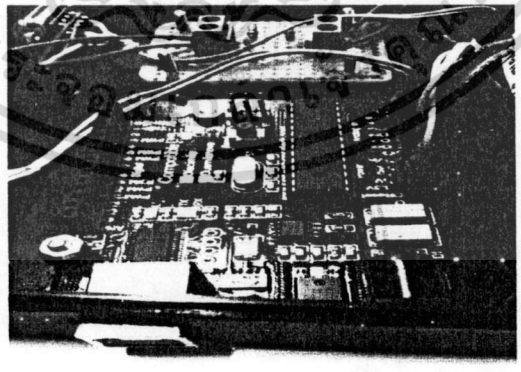
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเลข 1 คือเซนเซอร์วัดความเร่ง ADXL345 โดยจะมีการติดตั้งให้เชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์ Arduino (หมายเลข 2) และส่งค่าที่ได้ในตามแกน X, Y, Z ด้วยหลักการการส่งข้อมูลแบบ I2C Bus ผ่านทาง ขา SCL (PC.5) และ SDA (PC.4) โดยเซนเซอร์วัดความเร่ง ADXL345 มีความสัมพันธ์กับโปรแกรมกราฟิกสามมิติในกรณีที่ผู้เล่นต้องการควบคุมซ้าย–ขวา



รูปที่ 4.2 ADXL345 พร้อมกล่องสำหรับติดตั้งบนศีรษะ

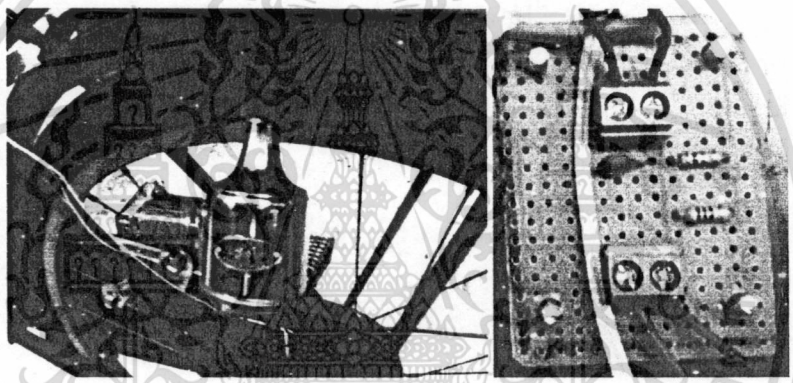
หมายเลข 2 คือบอร์ดไมโครคอนโทรลเลอร์ Arduino ที่มีการติดตั้งให้เชื่อมต่อกับเซนเซอร์วัดความเร่ง ADXL345(หมายเลข 1) และไดนาโมจักรยาน(หมายเลข 3) เพื่อรับค่าแกน X, Y, Z (จากเซนเซอร์วัดความเร่ง ADXL345) และรับค่าจากความดันในการปั่นจากไดนาโมจักรยาน(โดยแรงดันที่จ่ายเข้ามามีค่าระหว่าง 6-12 โวลต์)



รูปที่ 4.3 บอร์ดไมโครคอนโทรลเลอร์ Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเลข 3 คือไดนาโมจักรยาน โดยมีการติดตั้งให้เชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์ Arduino (หมายเลข 2) มีสายเชื่อมต่อไปยังบอร์ดดังกล่าว 2 เส้นคือ GND และขา ADC0 ก่อนการนำค่าข้อมูลเข้าไปในบอร์ดไมโครคอนโทรลเลอร์ Arduino ต้องผ่านวงจรเพื่อป้องกันการกระชากของไฟและยังช่วยกรองแรงดัน ไดนาโมจักรยานจะนำมาใช้สำหรับการปั่นเพื่อให้เกิดแรงดันและนำไปใช้ในบอร์ดไมโครคอนโทรลเลอร์ Arduino เพื่อส่งค่าดังกล่าวให้ประมวลผลทางโปรแกรมกราฟิกสามมิติ การนำค่าดังกล่าวไปใช้มีความสัมพันธ์กับโปรแกรมกราฟิกสามมิติ ในกรณีที่ผู้เล่นต้องการเคลื่อนที่ ผู้เล่นสามารถปั่นจักรยานควบคุมการเคลื่อนที่เองได้



รูปที่ 4.4 ไดนาโมจักรยานและวงจรกรองแรงดัน

หมายเลข 4 คือคอมพิวเตอร์ ที่ใช้สำหรับรับค่าข้อมูลจากบอร์ดไมโครคอนโทรลเลอร์ Arduino และนำค่าข้อมูลดังกล่าวมาใช้ในการประมวลผลในโปรแกรมเกมสามมิติ สุดท้ายผลดังกล่าวจะแสดงออกมาทางมอนิเตอร์(หน้าจอแสดงผล)

รูปที่ 4.5 หน้าโปรแกรมเกมสามมิติที่แสดงออกมาทางหน้าจอแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดสอบการทำงานของบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove

การทดสอบการทำงานของบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove ด้วยการนำโปรแกรมตัวอย่างมาคอมไพล์ และ อัพโหลดโปรแกรมไปยังโมดูล

4.2.1 การเปิดโปรแกรม Arduino

ทำการเปิดโปรแกรมโดยเข้าไปที่ โปรแกรม Arduino จะขึ้นมาดังรูปที่ 4.6

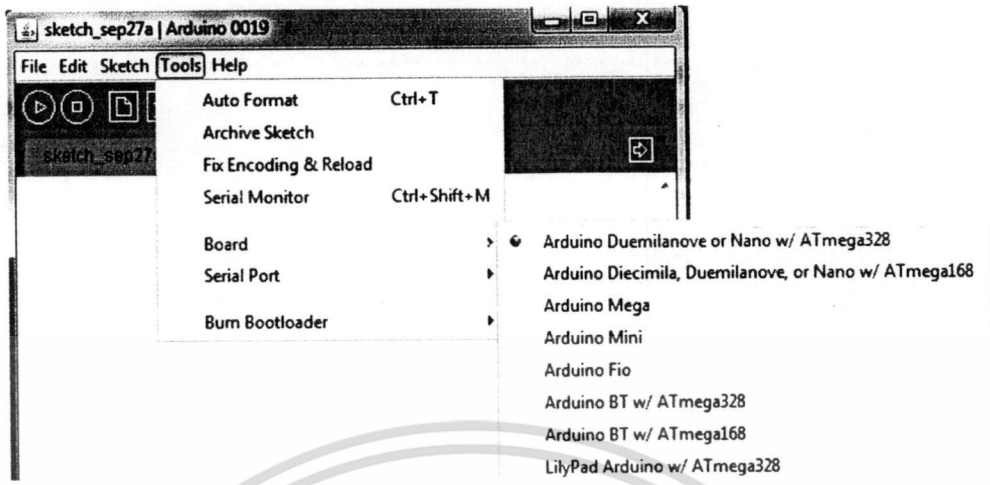
รูปที่ 4.6 หน้าต่างของโปรแกรม Arduino

4.2.2 กำหนดค่าเบื้องต้นสำหรับใช้งานกับ บอร์ดไมโครคอนโทรลเลอร์ Arduino

4.2.2.1 การเลือกบอร์ดไมโครคอนโทรลเลอร์

เลือกเมนู Tool>>Board>>Arduino Duemilanove or Nano w /ATmega328 ดังรูปที่ 4.7

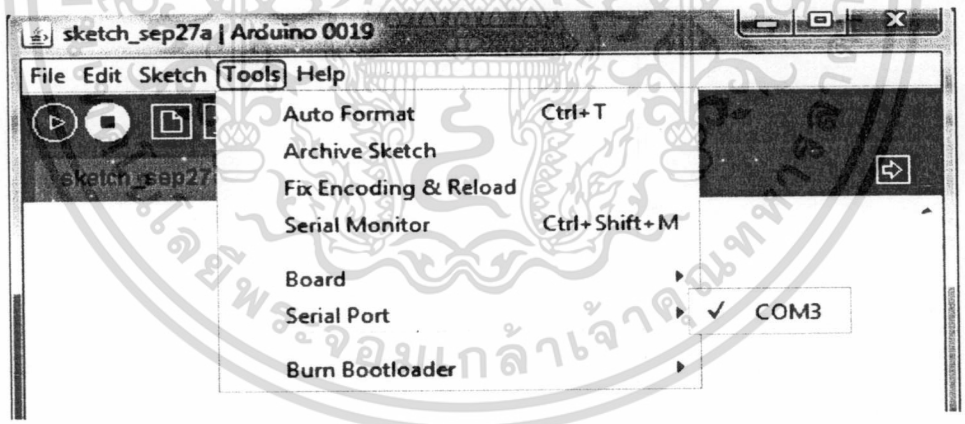
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 การเลือกไมโครคอนโทรลเลอร์ของ Arduino

4.2.2.2 การเลือกพอร์ตอนุกรมที่ใช้ในการติดต่อกับบอร์ด

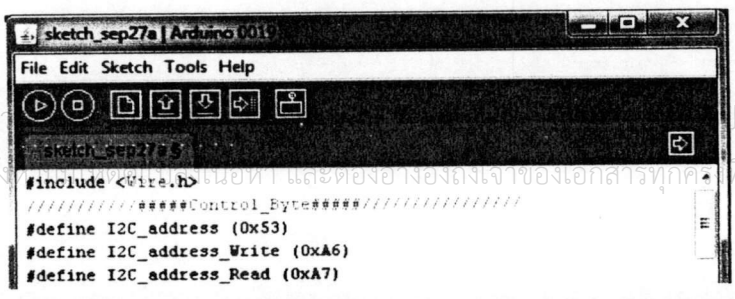
การอัพโหลดโปรแกรมจาก Arduino จะทำผ่านพอร์ตอนุกรม ซึ่งจะต้องกำหนดหมายเลขพอร์ตที่จะใช้ดังนี้ เลือกเมนู Tool>> Serial Port โปรแกรมจะแสดงพอร์ตอนุกรมที่มีในคอมพิวเตอร์ให้ผู้ใช้สามารถเลือกได้ ดังรูปที่ 4.8



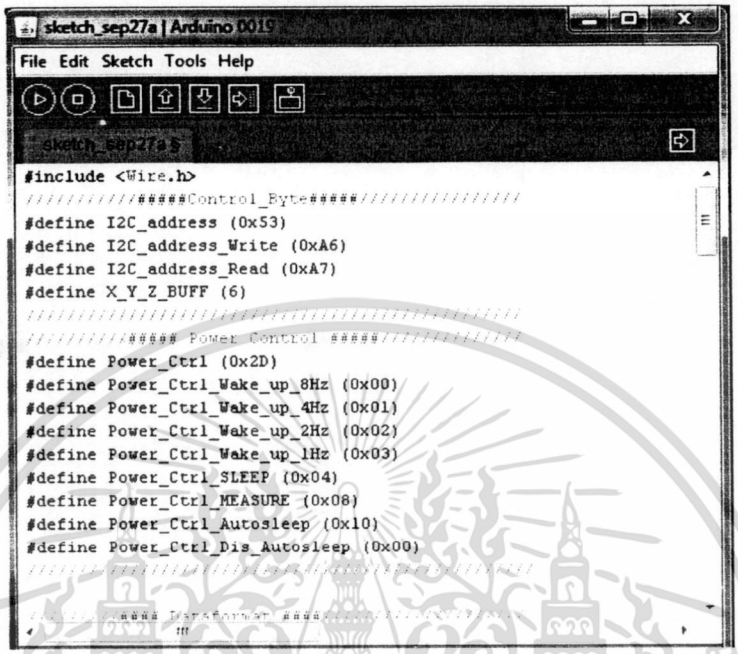
รูปที่ 4.8 การเลือกพอร์ตอนุกรมที่ใช้ติดต่อไมโครคอนโทรลเลอร์ของ Arduino

4.2.3 การทดลองเขียนโปรแกรมรับค่าจากเซนเซอร์ ADXL345 และไดนาโมจักรยาน

การเขียนโปรแกรมของ Arduino จะใช้รูปแบบของภาษา C และ C++ ซึ่งเป็นโปรแกรมที่เขียนง่ายไม่ซับซ้อนมากนัก โดยเมื่อเปิดโปรแกรม จะมีพื้นที่สำหรับเขียน code ดังรูปที่ 4.9



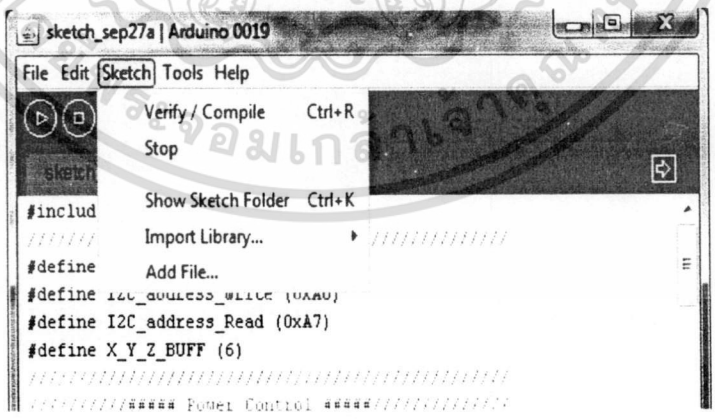
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 หน้าต่างของโปรแกรม Arduino และโปรแกรมที่เขียน

4.2.4 ทดลองการคอมไพล์โปรแกรม

เมื่อเปิดไฟล์โปรแกรมแล้วจากนั้นทำการคอมไพล์โปรแกรมเพื่อตรวจสอบความถูกต้องก่อนเขียนข้อมูลลงในบอร์ดไมโครคอนโทลเลอร์ Arduino ดังรูปที่ 4.10



รูปที่ 4.10 การเลือกเมนูคำสั่งเพื่อทำการคอมไพล์โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

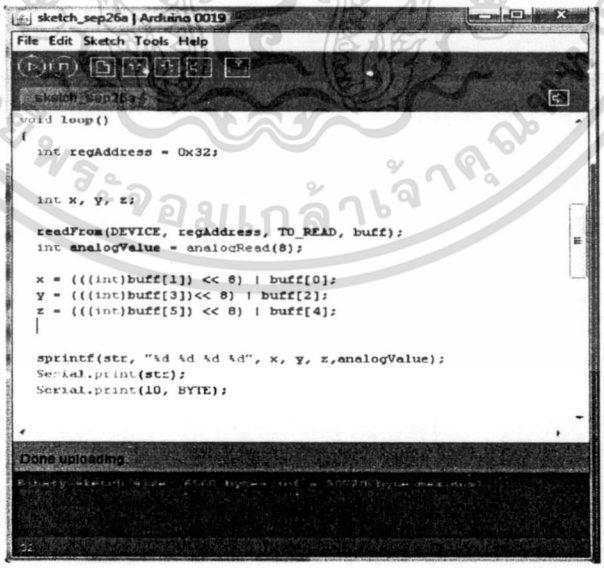
size: 1008 bytes (of a 30720 byte maximum) โดยแสดงว่าโปรแกรมที่ได้จากการคอมไพล์มีขนาด 1008 bytes จากทั้งหมด 30720 byte ที่สามารถใช้งานได้

รูปที่ 4.11 แถบแสดงสถานการณ์คอมไพล์

4.2.5 การอัพโหลดข้อมูลไปยังบอร์ด ไมโครคอนโทรลเลอร์ Arduino Duemilanove

การอัพโหลดข้อมูล ไปยังบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove มีขั้นตอนดังนี้

1. ต่อ USB พอร์ตเข้ากับบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove และคอมพิวเตอร์
2. กดสวิตช์ Reset ผลคือจะมีไฟสีแดงกระพริบขึ้น แสดงว่าบอร์ดพร้อมทำการอัพโหลดข้อมูล
3. เลือกเมนู File >> Upload to I/O Board ที่โปรแกรม Arduino
4. อัพโหลดโปรแกรมที่ต้องการเขียน
5. เมื่ออัพโหลดเสร็จสมบูรณ์ แถบแสดงสถานะของโปรแกรมจะขึ้นข้อความ Done Uploading ดังรูปที่ 4.12

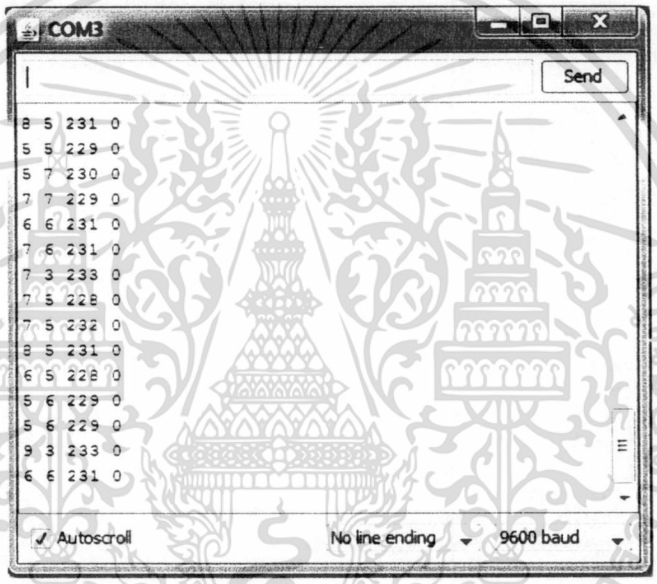


รูปที่ 4.12 หน้าต่างของโปรแกรมเมื่อทำการอัพโหลดเสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.6 การทดลองแสดงข้อมูลของ Serial Monitor ในโปรแกรม Arduino

โปรแกรม Arduino จะมี Serial Monitor สำหรับไว้แสดงค่าข้อมูลของพอร์ตที่ติดต่อเมื่อเปิดโปรแกรมแล้วสามารถเรียกใช้หน้าต่าง Serial Monitor โดยการ Ctrl +Shift + M โปรแกรมจะแสดงหน้าต่าง Serial Monitor ขึ้นมา จากนั้นจึงทำการเลือกพอร์ตและ Board Rate ที่ตั้งไว้ก็สามารถแสดงผลของโปรแกรมที่เขียนไว้ได้ตามรูปที่ 4.13 ซึ่งจะแสดงข้อมูลที่รับค่าข้อมูลจากเซนเซอร์ ADXL345 และไดนาโมจักรยานโดยจะเห็นได้ว่ามีการแสดงค่าครั้งละ 4 ค่า ลำดับค่าที่ 1-3 คือค่าในแกน X, Y, Z ตามลำดับ ลำดับค่าที่ 4 เป็นค่าข้อมูลของไดนาโมจักรยาน



รูปที่ 4.13 หน้าต่างของ Serial Monitor แสดงค่าข้อมูลของเซนเซอร์ ADXL345และไดนาโมจักรยาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การแสดงกราฟข้อมูลจากเซนเซอร์วัดความเร่ง ADXL345

การทดลองในส่วนนี้เป็นการนำข้อมูลที่อ่านได้จากเซนเซอร์วัดความเร่ง ADXL345 มาแสดงในรูปแบบกราฟเพื่อแสดงความสัมพันธ์ของ แกน X,Y,Z เมื่อทำการวางตำแหน่งของเซนเซอร์ในระนาบต่างๆ ที่เวลาต่างกัน

4.3.1 แสดงค่าที่ได้จากเซนเซอร์วัดความเร่ง ADXL345 ในระนาบ Z+

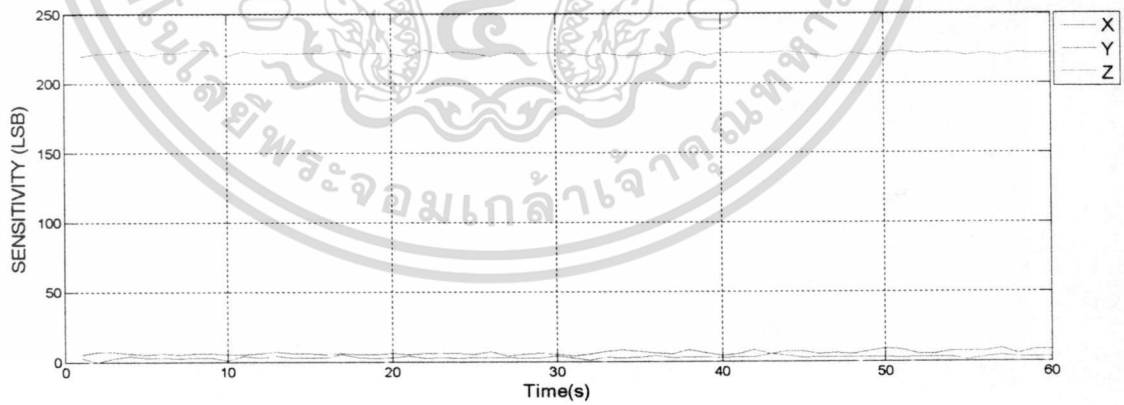
$$X_{out} = 0g$$

$$Y_{out} = 0g$$

$$Z_{out} = +1g$$

รูปที่ 4.14 การวางเซนเซอร์ระนาบ Z+

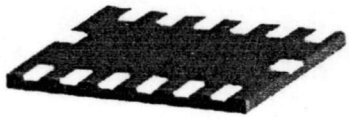
ผลการทดลองจะได้ค่าในแกน Z ที่มีค่าสูงมาก โดยค่าในแกน X และ Y มีค่าเข้าใกล้ศูนย์ ดังรูปที่ 4.15



รูปที่ 4.15 กราฟแสดงค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ Z+

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 แสดงค่าที่ได้จาก เซนเซอร์วัดความเร่ง ADXL345 ในระนาบ Z-



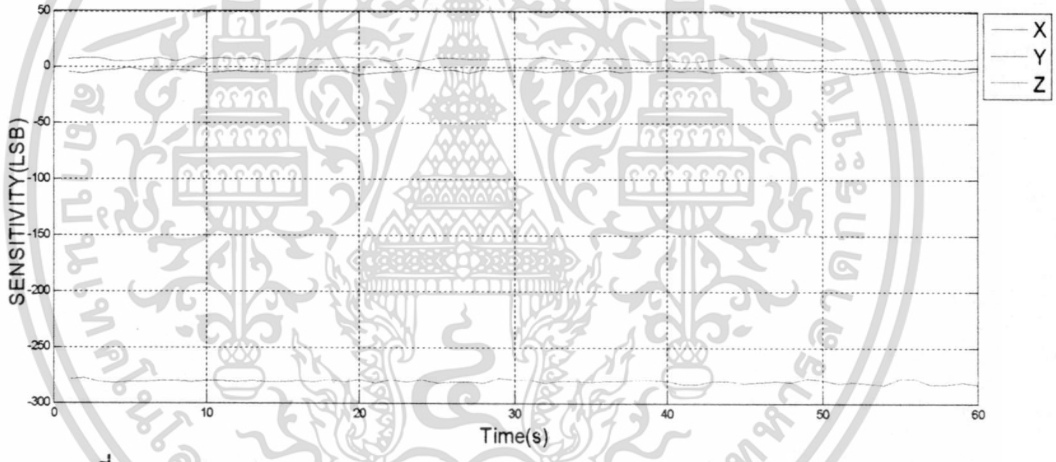
$X_{out} = 0g$

$Y_{out} = 0g$

$Z_{out} = -1g$

รูปที่ 4.16 การวางเซนเซอร์ระนาบ Z-

ผลการทดลองจะได้ค่าในแกน Z ที่มีค่าติดลบสูงมากโดยค่าในแกน X และ Y มีค่าเข้าใกล้ศูนย์ ดังรูปที่ 4.17



รูปที่ 4.17 กราฟแสดงค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ Z-

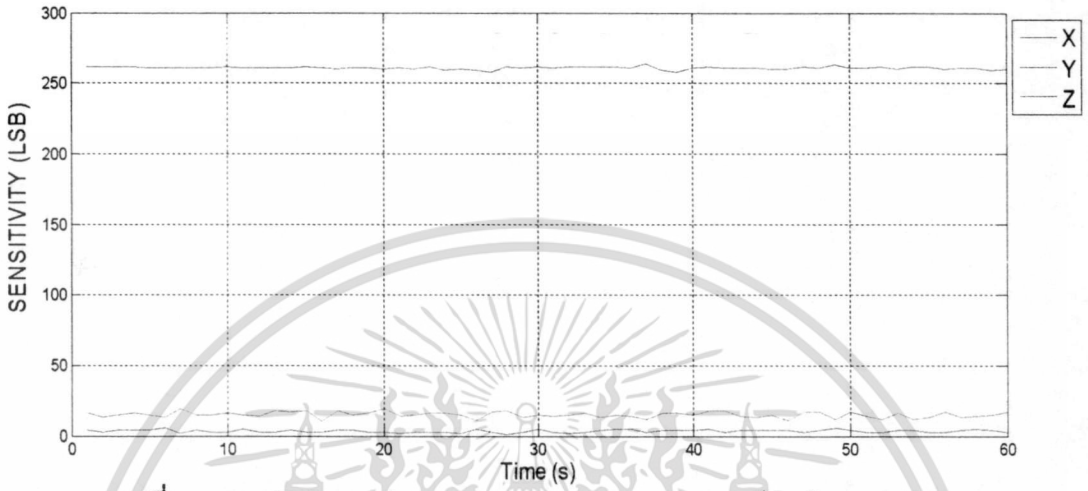
4.3.3 แสดงค่าที่ได้จาก เซนเซอร์วัดความเร่ง ADXL345 ในระนาบ X+

$X_{out} = 1g, Y_{out} = 0g, Z_{out} = 0g$

รูปที่ 4.18 การวางเซนเซอร์ระนาบ X+

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลองจะได้ค่าในแกน X จะมีค่าสูงมากโดยค่าในแกน Y และ Z มีค่าเข้าใกล้ศูนย์ รูปที่ 4.19



รูปที่ 4.19 กราฟแสดงค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ X+

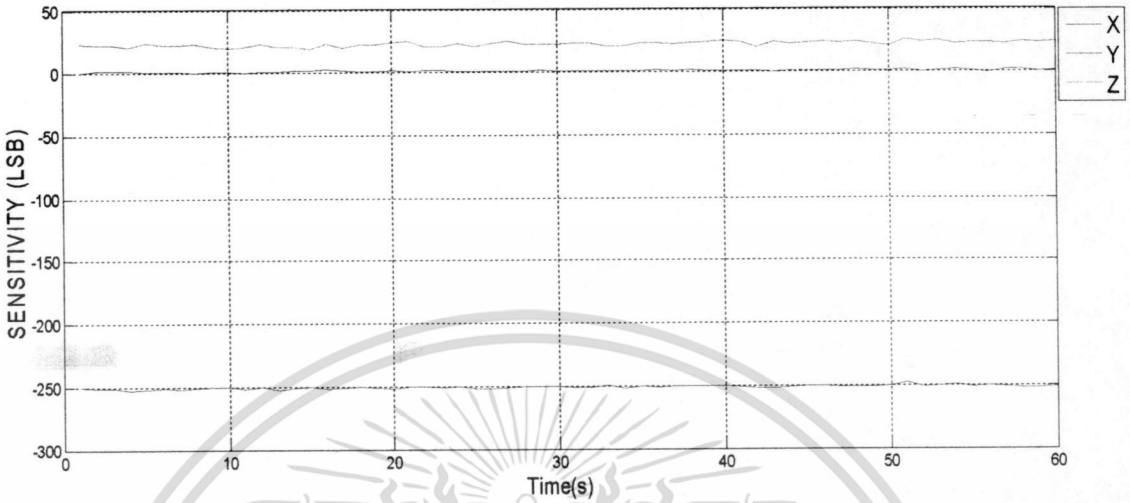
4.3.4 แสดงค่าที่ได้จาก เซนเซอร์วัดความเร่ง ADXL345 ในระนาบ X-

$X_{out} = -1g, Y_{out} = 0g, Z_{out} = 0g$

รูปที่ 4.20 การวางเซนเซอร์ระนาบ X-

ผลการทดลองจะได้ว่าค่าในแกน X จะติดลบสูงมากโดยค่าในแกน Y และ Z มีค่าเข้าใกล้ศูนย์ ดังรูปที่ 4.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 กราฟแสดงค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ X-

4.3.5 แสดงค่าที่ได้จาก เซนเซอร์วัดความเร่ง ADXL345 ในระนาบ Y+

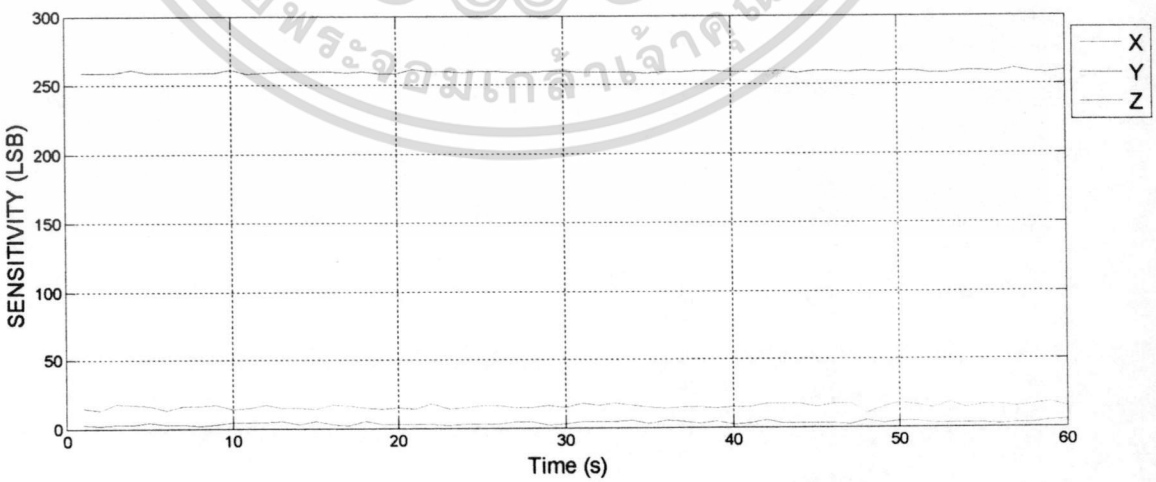
$X_{out} = 0g$

$Y_{out} = 1g$

$Z_{out} = 0g$

รูปที่ 4.22 การวางเซนเซอร์ระนาบ Y+

ผลการทดลองจะได้ค่าในแกน Y จะมีค่าสูงมากโดยค่าในแกน X และ Z มีค่าเข้าใกล้ศูนย์ รูปที่ 4.23



รูปที่ 4.23 ค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ Y+

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.6 แสดงค่าที่ได้จาก เซนเซอร์วัดความเร่ง ADXL345 ในระนาบ Y-

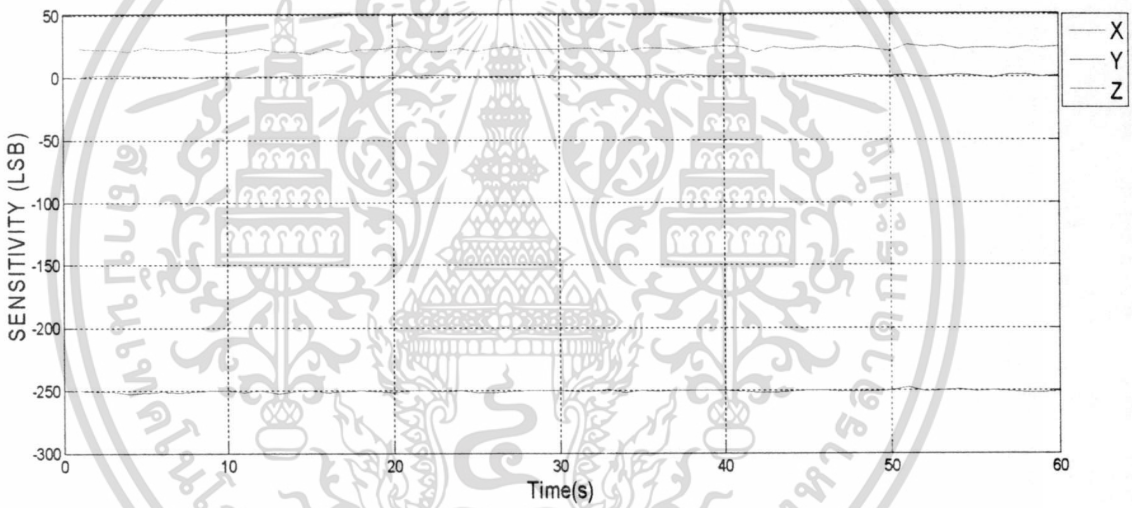
$X_{out} = 0g$

$Y_{out} = -1g$

$Z_{out} = 0g$

รูปที่ 4.24 การวางเซนเซอร์ระนาบ Y-

ผลการทดลองจะได้ว่าค่าในแกน Y จะติดลบสูงมากโดยค่าในแกน X และ Z มีค่าเข้าใกล้ศูนย์ ดังรูปที่ 4.25



รูปที่ 4.25 ค่า Sensitivity ของแกน X, Y, Z เทียบกับเวลา ระนาบ Y-

4.4 การทดลองใช้งานโปรแกรมภาษา Python สำหรับการรับค่าข้อมูลจากบอร์ดไมโครคอนโทรลเลอร์ Arduino เพื่อนำเข้าเก็บไว้ในรูปแบบไฟล์ .text

เนื่องจากโปรแกรม Blender3D เป็นโปรแกรมที่มีพื้นฐานอยู่ในระบบปฏิบัติการ Linux จึงต้องใช้ Python script ในการเขียนโปรแกรมรับค่าข้อมูลจากบอร์ดไมโครคอนโทรลเลอร์ เพื่อให้สามารถนำไปใช้งานใน โปรแกรม Blender3D ได้ ผลการทดลองคือเมื่อเปิดใช้งานโปรแกรมที่เขียนไว้จะมีค่าจากบอร์ดไมโครคอนโทรลเลอร์มาเก็บไว้ในไฟล์.text ที่ได้สร้างไว้ ทำให้สามารถนำไปใช้ประมวลผลในโปรแกรม Blender3D ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

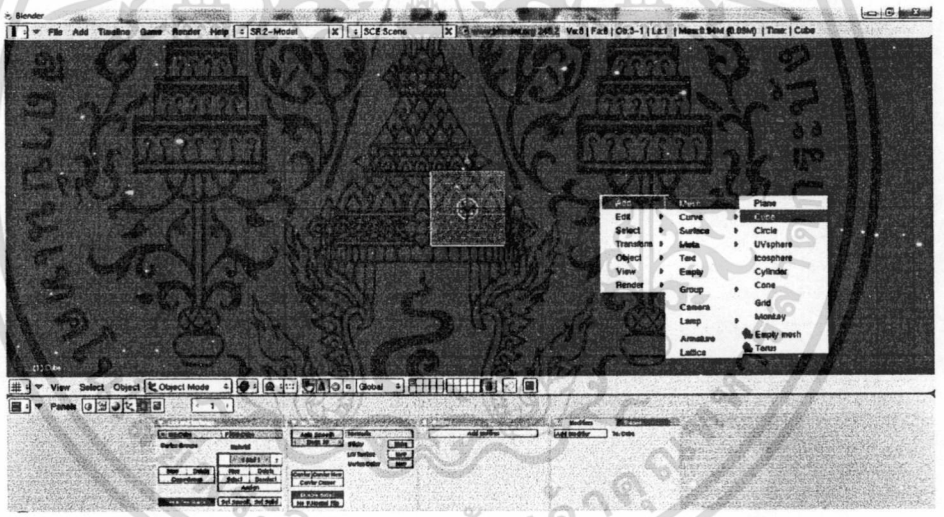
4.5 ทดลองการใช้งานโปรแกรม Blender3D

4.5.1 การทดลองใช้โปรแกรม Blender3D สำหรับออกแบบและสร้างกราฟิกสามมิติ

การออกแบบและสร้างกราฟิกสามมิติในโปรแกรม Blender3D สามารถออกแบบตามความต้องการหรือตามความคิดสร้างสรรค์ของผู้ออกแบบได้ แต่สิ่งสำคัญคือจะต้องรู้ทักษะเบื้องต้นสำหรับการออกแบบเสียก่อน เพราะโปรแกรมดังกล่าวมีการใช้งานรวมถึงไลบรารี่ที่เฉพาะตัว การทำงานต่างๆ จึงสามารถนำมาใช้ได้ง่ายหากผู้ออกแบบมีทักษะดังกล่าว

4.5.2 การออกแบบและสร้างกราฟิกสามมิติ (โมเดลบ้าน)

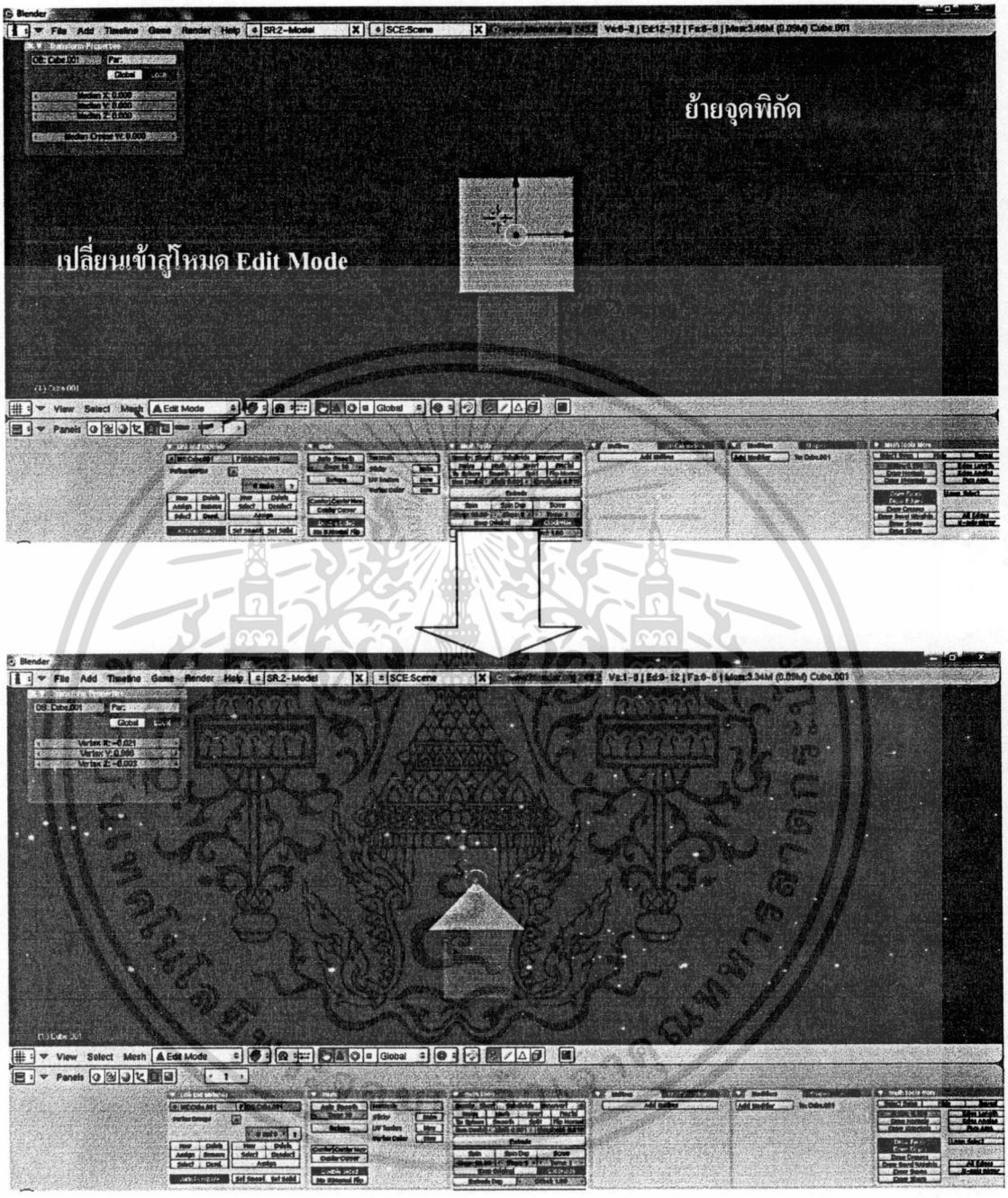
1. เปิดโปรแกรม Blender3D เพิ่มวัตถุตามรูปทรงที่ต้องการในที่นี่เลือกวัตถุชนิด Cube 2 ชิ้นและ Planeโดยเลือกที่ Spread Bar >> Add >> Mesh >> Cube และ Spread Bar >> Add >> Mesh >> Plane



รูปที่ 4.26 เลือกวัตถุในโปรแกรม Blender3D

2. ทำหลังคาโมเดลรูปบ้านโดยปรับโหมดเป็นโหมด Edit Mode และคลิกเม้าส์ที่จุดพิกัดและย้ายจุดพิกัดดังกล่าวตามความต้องการ เช่นตัวอย่างดังรูปที่ 4.22

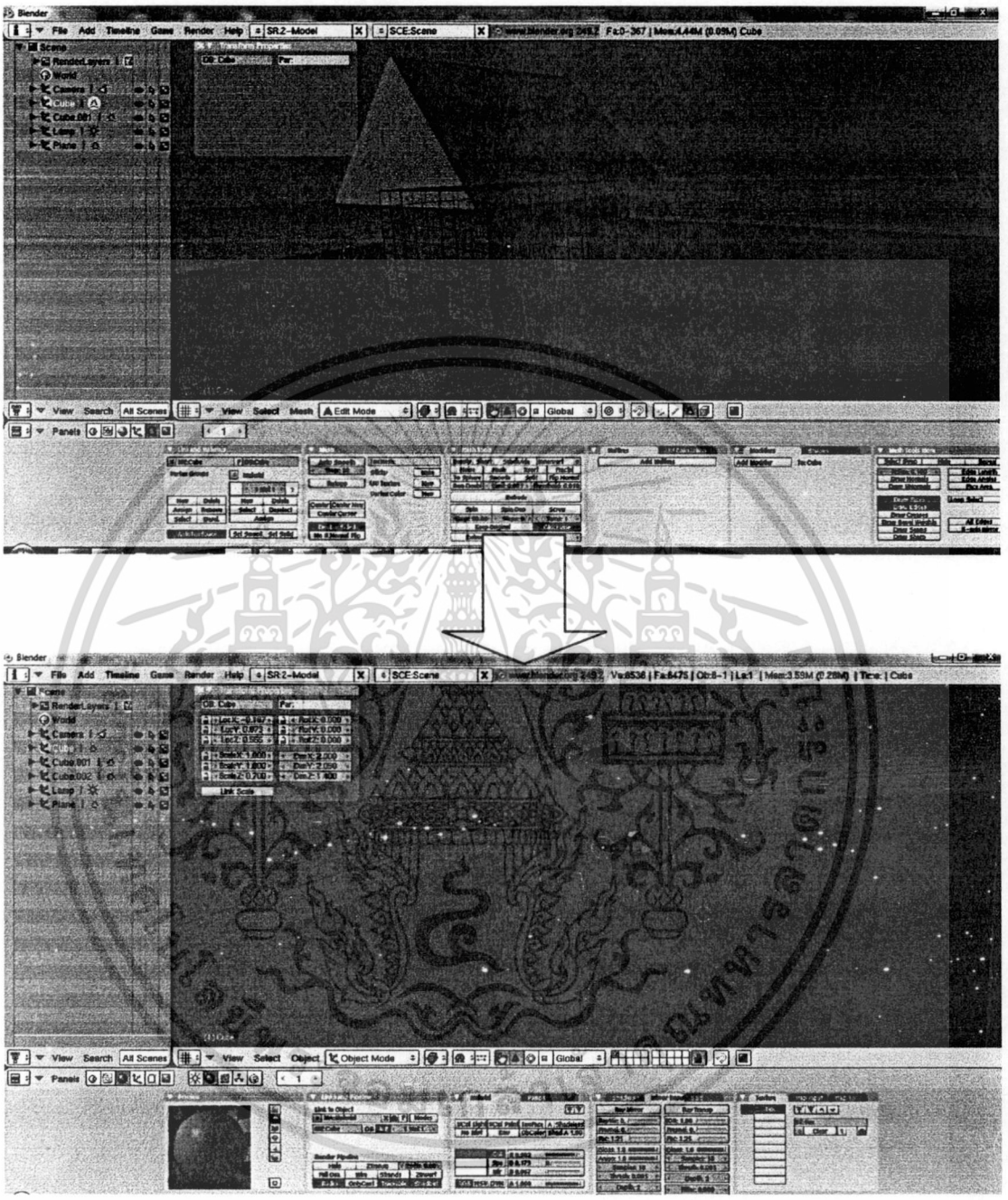
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.27 ย้ายจุดพิกัดเพื่อสร้างโมเดลรูปบ้าน

3. ตกแต่งโมเดลรูปบ้านตามความต้องการของผู้ออกแบบ ดังเช่นตัวอย่างรูปที่ 4.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.28 ตกแต่งโมเดลรูปบ้าน

4.6 การทดลองนำค่าที่ส่งเข้ามาจากไมโครคอนโทรลเลอร์ Arduino ไปประมวลผลในโปรแกรมกราฟิกสามมิติ

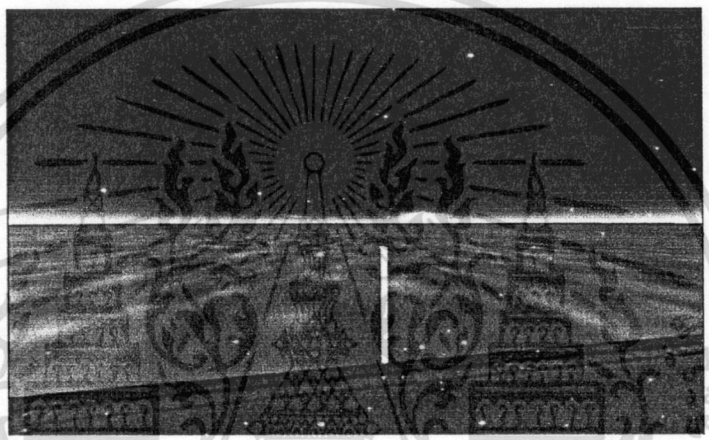
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนนี้เป็นการนำค่าที่ได้จากบอร์ดในทิศทางแกนต่างๆ มาใช้งานในโปรแกรมเกมที่ได้สร้างไว้ โดยแบ่งการรับค่าจาก 2 ส่วนคือ

4.6.1 การรับค่าจากเซนเซอร์วัดความเร่ง ADXL345

4.6.1.1 การเอียงศีรษะในระนาบ X-

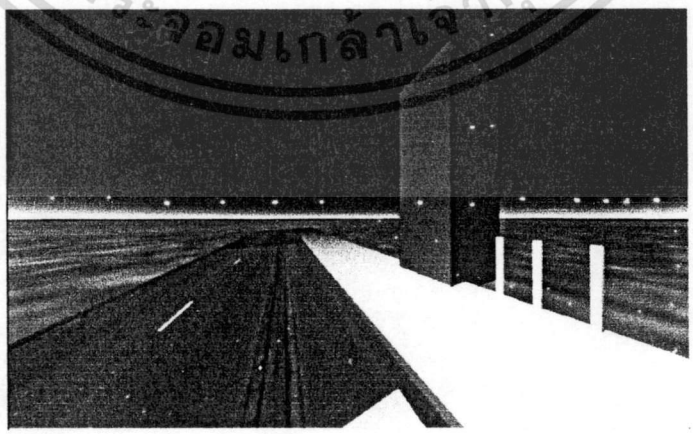
เมื่อเอียงศีรษะในระนาบ X- ค่าข้อมูลแกน X ของเซนเซอร์วัดความเร่งจะได้ค่าเป็นลบทำให้การหมุนของภาพในโปรแกรมกราฟิกสามมิติ เอียงในทิศทาง –X ดังรูปที่ 4.29



รูปที่ 4.29 ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อเอียงศีรษะในระนาบ X-

4.6.1.2 การเอียงศีรษะในระนาบ X+

เมื่อเอียงศีรษะในระนาบ X+ ค่าข้อมูลแกน X ของเซนเซอร์วัดความเร่งจะได้ค่าเป็นบวกทำให้มีการหมุนของภาพในโปรแกรมกราฟิกสามมิติ เอียงในทิศทาง +X ดังรูปที่ 4.30



รูปที่ 4.30 ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อเอียงศีรษะในระนาบ X+

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

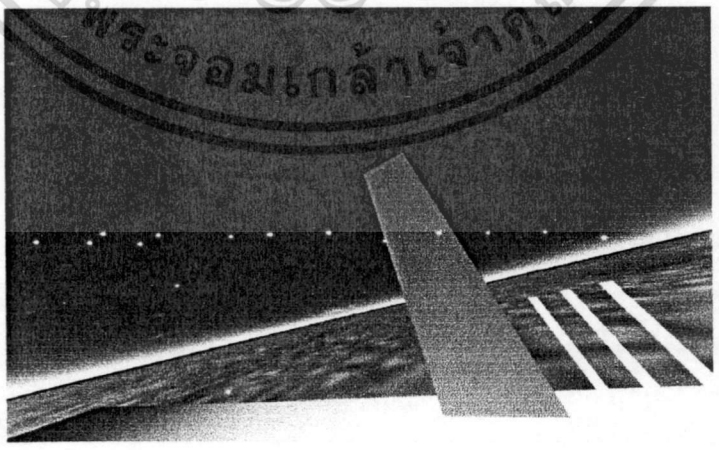
4.5.1.3 การเอียงศีรษะในระนาบ Y-

เมื่อเอียงศีรษะในระนาบ Y- ค่าข้อมูลแกน Y ของเซนเซอร์วัดความเร่งจะได้ค่าเป็นลบ แต่ได้ตั้งค่าให้ภาพกราฟิกสามมิติหมุนในระนาบ แกน Z ทำให้มีการหมุนของภาพในโปรแกรมกราฟิกสามมิติเอียงในทิศทาง -Z ดังรูปที่ 4.31

รูปที่ 4.31 ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อเอียงศีรษะในระนาบ Y-

4.5.1.4 การเอียงศีรษะในระนาบ Y+

เมื่อเอียงศีรษะในระนาบ Y+ ค่าข้อมูลแกน Y ของเซนเซอร์วัดความเร่งจะได้ค่าเป็นบวก แต่ได้ตั้งค่าให้ภาพกราฟิกสามมิติหมุนในระนาบ แกน Z ทำให้มีการหมุนของภาพในโปรแกรมกราฟิกสามมิติเอียงในทิศทาง +Z ดังรูปที่ 4.32

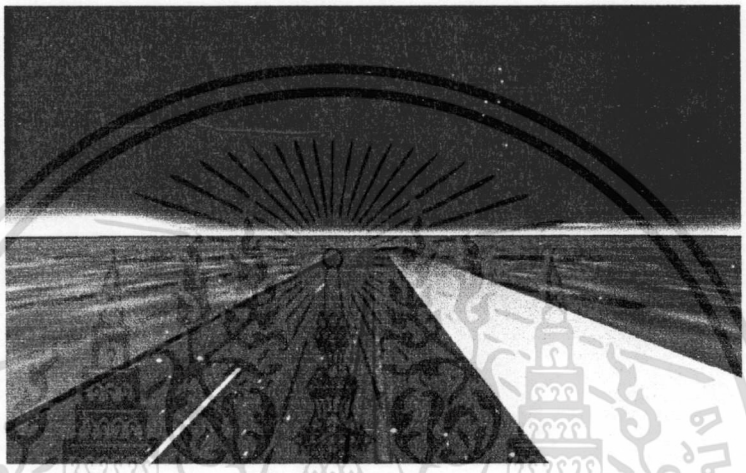


รูปที่ 4.32 ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อเอียงศีรษะในระนาบ Y+

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 การรับค่าจากไดนาโมจักรยาน

เมื่อผู้เล่นทำการปั่นจักรยานทำให้ไดนาโมทำงานส่งเป็นค่าสัญญาณอนาล็อก มาประมวลผลในโปรแกรมเกมกราฟิกสามมิติที่สร้างไว้ ผลที่ได้คือเมื่อปั่นจักรยานทำให้ภาพในโปรแกรมเกมกราฟิกสามมิติมีการเคลื่อนที่ไปข้างหน้า และเมื่อปั่นจักรยานเร็วขึ้นจะส่งผลให้การเคลื่อนที่เร็วขึ้นด้วย



รูปที่ 4.33 ภาพที่ได้จากเกมกราฟิกสามมิติ เมื่อไดนาโมทำงาน

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและอภิปรายผลการวิจัย

5.1 สรุปผลการวิจัย

การวิจัยนี้ได้ทำการออกแบบและสร้างโปรแกรมเกมกราฟิกสามมิติกับเครื่องออกกำลังกายด้วยจักรยาน เพื่อเพิ่มความเพลิดเพลินและแรงจูงใจให้กับการออกกำลังกายมากยิ่งขึ้น ทำให้ผู้ที่ไม่รักการออกกำลังกายเกิดความสนใจและมีวิสัยทัศน์ที่ดีกับการออกกำลังกายมากยิ่งขึ้น การดำเนินงานการออกแบบและสร้างดังกล่าวสามารถแบ่งได้เป็น 2 ส่วนหลักคือ

1. ส่วนการทำงานของอุปกรณ์บอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove เกิดการรับค่าข้อมูลที่ถูกส่งค่าจากเซนเซอร์วัดความเร่ง ADXL345 และไดนาโมจักรยาน(ในรูปแบบอนาล็อก) โดยค่าดังกล่าวที่รับได้ จะถูกส่งผ่านทางพอร์ตอนุกรมเพื่อแสดงผลและนำไปใช้กับส่วนของกราฟิกสามมิติ
2. ส่วนการทำงานของโปรแกรมเกมกราฟิกสามมิติ เกิดจากการออกแบบและสร้างโครงสร้างรูปทรงสามมิติด้วยโปรแกรม Blender3D และควบคุมการทำงานต่างๆของระบบด้วยโปรแกรมภาษา Python โดยการแสดงผลเกิดขึ้นได้ เพราะโปรแกรมเกมกราฟิกสามมิติสามารถรับค่าได้จากผู้เล่นโดยตรง อันเนื่องมาจากการส่งค่าข้อมูลจากบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove และไดนาโมจักรยานผ่านพอร์ตอนุกรมมาประมวลผลกับรูปทรงสามมิติที่สร้างขึ้น

5.2 อภิปรายผลการวิจัย

อุปกรณ์ที่ต้องใช้ในการดำเนินงานวิจัยบางส่วนมีราคาค่อนข้างสูง โดยปกติการรับค่าจากเซนเซอร์วัดความเร่ง ADXL345 ต้องเชื่อมต่อด้วยสายการส่งข้อมูลมายังบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove เมื่อนำไปติดบริเวณศีรษะจึงทำให้เกิดความยุ่งยากในการใช้งาน และอุปกรณ์เซนเซอร์วัดความเร่ง ADXL345 ไม่เป็นที่นิยมใช้ในประเทศไทย จึงมีความจำเป็นที่จะต้องหาข้อมูลจากต่างประเทศ ซึ่งเป็นอุปสรรคในเรื่องของภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางการพัฒนาต่อ

การออกแบบและสร้างโปรแกรมเกมกราฟิกสามมิติกับการออกกำลังกายด้วยจักรยานสามารถใช้งานได้จริง แต่ยังมีส่วนของปัญหาและข้อจำกัดการดำเนินงานในบางส่วนที่เกิดขึ้น ดังนั้นหากต้องการให้สมบูรณ์แบบจนกระทั่งสามารถใช้ในการค้าพาณิชย์ได้ต่อไป จึงต้องดำเนินแนวทางการพัฒนาซึ่งมี 2 ส่วนดังนี้

1. ส่วนอุปกรณ์ จากปัญหาการรับค่าจากเซนเซอร์วัดความเร่ง ADXL345 ที่รับข้อมูลจากบอร์ดไมโครคอนโทรลเลอร์ Arduino Duemilanove ทำให้เกิดความยุ่งยากของสายติดต่อ ดังนั้นอาจจะพัฒนาได้โดยเปลี่ยนรูปแบบการติดต่อข้อมูลเป็นแบบไร้สายเพื่อให้เกิดความสะดวกในการใช้งานมากยิ่งขึ้น
2. ส่วนโปรแกรมกราฟิกสามมิติ จากการสร้างโปรแกรมเกมกราฟิกสามมิติดังกล่าว สามารถเพิ่มเติมทางด้านกราฟิกสามมิติให้มีความหลายหลายและความน่าสนใจกับผู้เล่นมาขึ้นกว่าเดิมตามความต้องการของผู้พัฒนาได้

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ภาคผนวก ข., Datasheet หรือ http://www.alldatasheet.net/

[2] เอกชัย มะการ, เรียนรู้เข้าใจใช้งานไมโครคอนโทรลเลอร์ตระกูล AVR ด้วย Arduino, กรุงเทพฯ : บริษัท อีทีที จำกัด, 2552

[3] http://www.justusers.net/knowledges/usb.htm

[4] http://www.thaimicrotron.com/CCS-628/Referrence/I2CBUS.htm

[5] ภาคผนวก ข., Datasheet หรือ http://www.sparkfun.com/datasheets/Sensors/Accelerometer/ADXL345.pdf

[6] http://www.slideshare.net/ampair14901/ss-5093688

[7] http://lpsci.nfe.go.th/elearning/basic_science/basic64.php

[8] น.ท.ไพศาล โมลิสกุลมงคล, คอมพิวเตอร์กราฟิกส์ ใช้ OpenGL, กรุงเทพฯ : หจก.ไทยเจริญการพิมพ์, 2550

[9] cpe.ku.ac.th/~thanachat/documents/204111/01_Intro.ppt

[10] โอภาส ศิริครรชิตถาวร วรพจน์ กรแก้ววัฒนกุล ชัยวัฒน์ ลิ้มพรจิตรวิไล, เรียนรู้ควบคุมอย่างง่ายด้วยโปรแกรมภาษา C กับ Arduino และบอร์ดไมโครคอนโทรลเลอร์ POP-168, กรุงเทพฯ : บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด, 2552

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้