

ระบบเฝ้าระวังผู้สูงอายุ โดยใช้กล้องตรวจจับการเคลื่อนไหวแบบ 3 มิติ  
ตรวจจับการล้มพร้อมแจ้งเตือน

THE FALL SURVEILLANCE SYSTEM BY A 3D NON-CONTACT SENSER

วโรดม ดุสิตกุล

WARODOM DUSITKUL

สตั้มภ์ จันทร์บาง

SATAM CHANBANG

สรศักดิ์ คงอินทร์

SORASAK KONG-IN

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาอิเล็กทรอนิกส์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2557

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบเฝ้าระวังผู้สูงอายุ โดยใช้กล้องตรวจจับการเคลื่อนไหวแบบ 3 มิติ  
ตรวจจับการล้มพร้อมแจ้งเตือน

THE FALL SURVEILLANCE SYSTEM BY A 3D NON-CONTACT SENSER



T139922

โดย

วโรตม ดุสิตกุล

สตั้มภ์ จันทร์บาง

สรศักดิ์ คงอินทร์

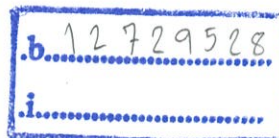
อาจารย์ที่ปรึกษา

ผศ.ดร.ภัทรพงษ์ ผาสุขกิจ

ร/พ.

ว3398  
2557

เลขหมู่.....139922  
เลขทะเบียน.....  
วัน,เดือน,ปี.....20 พ.ย. 2558



ปฏิญญานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาอิเล็กทรอนิกส์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2557

ปริญญานิพนธ์ ปีการศึกษา 2557

สาขาวิชา วิศวกรรมอิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์

เรื่อง สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ระบบเฝ้าระวังผู้สูงอายุ โดยใช้กล้องตรวจจับการเคลื่อนไหวแบบ 3 มิติ  
ตรวจจับการล้มพร้อมแจ้งเตือน

The fall surveillance system by a 3D non-contact sensor

ผู้จัดทำ

นายวโรดม ดุสิตกุล รหัส 54011160

นายสตั้มภ์ จันทร์บาง รหัส 54011310

นายสรศักดิ์ คงอินทร์ รหัส 54011335

ปริญญานิพนธ์นี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

๗๗-๗๗ 

..... อาจารย์ที่ปรึกษา  
( ผศ.ดร.ภัทรพงษ์ ผาสุขกิจ )

หัวข้อปริญญานิพนธ์	ระบบเฝ้าระวังผู้สูงอายุ โดยใช้กล้องตรวจจับการเคลื่อนไหวแบบ 3 มิติ ตรวจจับการล้ม พร้อมแจ้งเตือน		
นักศึกษา	นายวโรดม	ดุสิตกุล	54011160
	นายสัทม์ภัก	จันทร์บาง	54011310
	นายสรศักดิ์	คงอินทร์	54011335
ปริญญา	วิศวกรรมศาสตรบัณฑิต		
สาขาวิชา	วิศวกรรมอิเล็กทรอนิกส์		
ปีการศึกษา	2557		
อาจารย์ที่ปรึกษาปริญญานิพนธ์	ผศ.ดร.ภัทรพงษ์ ผาสุขกิจ		

### บทคัดย่อ

โครงงานฉบับนี้เป็นส่วนหนึ่งของวิชา Project โดยมีวัตถุประสงค์เพื่อตรวจจับการล้ม ของผู้สูงอายุ โดยใช้กล้องตรวจจับการเคลื่อนไหวแบบ 3 มิติ ทำการตรวจจับโครงร่างโครงกระดูก Skeleton Tracking ของร่างกาย โดยใช้กล้อง Kinect ทำหน้าที่เป็นเซนเซอร์ตรวจจับทำการตรวจจับจุดบริเวณระดับเอวขึ้นไป เทียบกับจุดกราวด์ (พื้น) ถ้าจุดที่เราตั้งค่าเอาไว้ระนาบเดียวกับจุดกราวด์ แสดงว่ามีการล้มเกิดขึ้น โดยมีการเชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์ (Arduino) เพื่อแจ้งเตือนให้กับคนอื่นได้ทราบในรูปแบบ ข้อความ เสียงและแสง เพื่อทำการช่วยเหลือผู้สูงอายุที่ล้มได้ทัน่วงที่

Thesis Title	The fall surveillance system by a 3D non-contact sensor		
Student	Mr.Warodom dusitkul		54011160
	Mr.Satam Chanbang		54011310
	Mr.Sorasak Sorasakkong-in		54011335
Degree	Bachelor of Engineering		
Program	Electronics Engineering		
Year	2014		
Thesis Advisor	Asst.Prof.Dr.Pattarapong Phasukkit		

### Abstract

This project is part of a course Project. The objective is to detect a fall in the elderly, used 3D motion detector of the elderly body by Skeleton Tracking of Kinect, acts as the sensor detect the waist upwards relative to ground. If the point is that we set out on the ground, it showed the elderly are falling down and connecting to the microcontroller (Arduino) to alert other person, who are hear message, sound and light from microcontroller for assist the elderly was falling immediately.

## กิตติกรรมประกาศ

ปริญญาโทเล่มนี้จะไม่สำเร็จลุล่วงไปได้เลยหากไม่ได้รับความช่วยเหลือและคำแนะนำจากบุคคลหลายท่าน

ขอขอบคุณอาจารย์ที่ปรึกษา ผศ.ดร.ภัทรพงษ์ ผาสุขกิจ ที่ได้ให้คำแนะนำต่างๆ ชี้แนะแนวทางต่างๆ อยู่เสมอ

ขอขอบคุณ พี่ๆ น้องๆ โดยเฉพาะ นายรัชชัย ไตรสและนายพัศพงศ์ เจริญพันธ์ ที่คอยเป็นกำลังใจและที่ปรึกษาในด้านต่างๆ อยู่เสมอ

และสุดท้ายขอขอบคุณบุคคลทั้งหมดอีกครั้งและขอภัยที่ไม่ได้กล่าวชื่อบุคคลทั้งหมดไว้ในที่นี้ นอกจากนี้ยังต้องขอขอบคุณผู้ปกครองของพวกเราทุกคนอีกด้วย ที่สนับสนุนด้านการเงินตลอดจนเป็นกำลังใจให้พวกเรามาโดยเสมอ

คณะผู้จัดทำ

นายวโรดม	ดุสิตกุล
นายสตั้มภ	จันทร์บาง
นายสรศักดิ์	คงอินทร์

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของโครงการ.....	2
1.3 สมมติฐานของการศึกษาวิจัย.....	2
1.4 ทฤษฎีหรือแนวคิดการวิจัย.....	2
1.5 ขอบเขตของการวิจัย.....	3
บทที่ 2 ความรู้พื้นฐาน.....	4
2.1 ทฤษฎีเบื้องต้น.....	4
2.1.1 การเปลี่ยนแปลงในวัยสูงอายุ.....	4
2.1.2 การพลัดตกหกล้ม.....	5
2.1.3 ปัจจัยส่งเสริมภาวะหกล้มในผู้สูงอายุ.....	5
2.1.4 การประเมินความเสี่ยงต่อการพลัดตกหกล้ม.....	6
2.1.5 ผลกระทบของการหกล้ม.....	6
2.1.6 การป้องกันการหกล้มของผู้สูงอายุที่บ้าน.....	6
2.2 อุปกรณ์และโปรแกรม.....	8
2.2.1 เครื่องเตือนคนล้ม.....	8
2.2.2 Arduino Uno.....	8

## สารบัญ(ต่อ)

	หน้า
2.2.3 SIM900 GSM/GPRS Arduino shield.....	10
2.2.4 Kinect X-Box 360.....	11
2.2.5 โปรแกรม Microsoft Visual Studio 2010).....	15
บทที่3 กระบวนการทำงานและขั้นตอนการทดสอบการทำงาน.....	17
3.1 กระบวนการในการทำงาน.....	17
3.2 ขั้นตอนการทดสอบ.....	20
3.3 การพัฒนาโปรแกรมส่วน Client.....	24
3.4 การพัฒนาโปรแกรมส่วน server.....	28
บทที่4 ผลการทดลอง.....	31
บทที่5 วิจารณ์สรุปผล.....	48
อ้างอิง.....	49
ภาคผนวก ก. Code Visual Basic.....	50
ภาคผนวก ข. Arduino Code.....	57
ภาคผนวก ค. Schematic.....	61
ภาคผนวก ง.ภาพการทำงาน(เก็บตก).....	63
ประวัติ	
ผู้วิจัย.....	67

## สารบัญรูป

รูปที่	หน้า
2.1 เครื่องเตือนคนล้ม.....	8
2.2 Arduino uno.....	9
2.3 SIM900 GSM/GPRS Arduino shield.....	10
2.4 ส่วนประกอบต่างๆ ของ Kinect.....	11
2.5 หลักการทำงานของKinect.....	12
2.6 หลักการทำตรวจวัดระยะของkinect .....	14
2.7 โปรแกรม Microsoft Visual Studio 2010 .....	15
3.1 ภาพแสดงกระบวนการงานในการทำงาน .....	17
3.2 ลักษณะท่าทางของคนปกติ.....	18
3.3 ลักษณะท่าทางของลักษณะคนโก่งจะล้มหรือเกือบล้ม.....	18
3.4 ลักษณะของคนที่ล้มและหมดสติ.....	19
3.5 แสดงการจับภาพจาก Kinect และค่าที่ได้จากการทดลอง.....	19
3.6 ทดสอบการใช้งาน Kinect .....	20
3.7 เขียนโค้ด Microsoft Visual Basic เพื่อทำการติดต่อกับ Kinect.....	21
3.8 เขียนโค้ดเพื่อรองรับการทำงานของตัว Kinect.....	21
3.9 ทำการทดสอบการทำงานของ Code Arduino.....	22
3.10 ทำการเชื่อมต่อระบบ.....	22
3.11 ทำการจัดอุปกรณ์ลงกล่อง.....	23
3.12 ตัวอย่างโปรแกรมส่วน Client.....	24
3.13 แสดงการทำงานของโปรแกรมส่วน Client.....	25
3.14 โค้ดการรวมค่าตำแหน่งของข้อต่อให้เป็น String.....	27
3.15 ตัวอย่าง Code การรับและส่งข้อมูล Arduino.....	28
3.16 แสดงหลักการทำงานของโปรแกรมส่วน Client.....	29

## สารบัญรูป(ต่อ)

รูปที่	หน้า
4.1 ลักษณะการล้มไปข้างหน้า.....	32
4.2 ลักษณะการล้มไปข้างหลัง.....	32
4.3 ลักษณะการล้มไปทางซ้ายมือของตัวเอง.....	33
4.4 ลักษณะการล้มไปทางขวามือของตัวเอง.....	33
4.5 แสดงกราฟการล้มไปข้างหน้า.....	37
4.6 แสดงกราฟการล้มไปข้างหลัง.....	37
4.7 แสดงกราฟการล้มไปซ้ายมือ.....	44
4.8 แสดงกราฟการล้มไปขวามือ.....	44
ค.1 แสดง schematic ของวงจรแจ้งเตือน.....	62
ง.1 แสดงด้านหลังของกล่องชิ้นงาน.....	64
ง.2 แสดงด้านหน้าของกล่องชิ้นงาน.....	64
ง.3 แสดงด้านบนของกล่องชิ้นงาน.....	65
ง.4 แสดงด้านในของกล่องชิ้นงาน.....	65
ง.5 แสดงการทำโปรเจค.....	66
ง.6 แสดงการทดสอบโปรเจค.....	66

## สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางคุณสมบัติของ Kinect.....	12
3.1 ตัวอย่างตารางบันทึกค่า .....	20
3.2 ผลการทดสอบและบันทึกผล.....	23
4.1 แสดงค่าตำแหน่ง/พิกัด ที่Z ประมาณ 1.5 ขณะยืน.....	31
4.2 แสดงค่าตำแหน่ง/พิกัด ที่Z ประมาณ 2.5 ขณะยืน.....	31
4.3 แสดงค่าตำแหน่ง/พิกัด ที่Z ประมาณ 3.5 ขณะยืน.....	31
4.4 แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปข้างหน้า.....	34
4.5 แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปข้างหลัง.....	38
4.6 แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปทางซ้ายมือตัวเอง.....	41
4.7 แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปทางขวามือตัวเอง.....	45

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันสังคมไทยได้กลายเป็นสังคมผู้สูงอายุ ด้วยสัดส่วนจำนวนประชากรผู้สูงอายุที่เพิ่มขึ้น จากอัตราการเกิดและอัตราการตายที่ลดลง จากความความเจริญทางเทคโนโลยีความก้าวหน้าทางวิทยาศาสตร์การแพทย์ มีผลทำให้ประชากรมีอายุยืนยาวส่งผลให้จำนวนผู้สูงอายุเพิ่มขึ้น ซึ่งเห็นได้จาก สำนักส่งเสริมและพิทักษ์ผู้สูงอายุ พบว่าสถิติประชากรที่มีการเปลี่ยนแปลงในระหว่างปี พ.ศ. 2553 – พ.ศ.2583 สัดส่วนของประชากรวัยเด็ก และวัยแรงงาน มีแนวโน้มลดลง ในขณะที่สัดส่วนของประชากรสูงอายุ มีแนวโน้มเพิ่มขึ้นอย่างต่อเนื่อง จากร้อยละ 13.2 ในพ.ศ. 2553 เป็นร้อยละ 32.1 ในพ.ศ.2583 ซึ่งในการเปลี่ยนแปลงของผู้สูงอายุมีการเปลี่ยนแปลงหลายด้านด้วยกันทั้งทางด้านสภาพร่างกาย จิตใจ ตลอดจนการดำรงอยู่ในสังคมของผู้สูงอายุ จะมีการเปลี่ยนแปลงไปจากวัยอื่น เนื่องจากมีความเสื่อมของการทำงานระบบต่างๆ ทั้งร่างกาย โดยเฉพาะระบบประสาท ระบบหัวใจ และหลอดเลือด ระบบขับถ่าย ระบบต่อมไร้ท่อ ระบบกระดูก และกล้ามเนื้อ ซึ่งการเปลี่ยนแปลงนี้ มักจะเกิดขึ้นช้าๆ ในภาวะปกติอวัยวะของระบบต่างๆ ยังทำหน้าที่ได้อย่างปกติแต่ในภาวะบีบคั้น ไม่ว่าจะเกิดจากทางอารมณ์ หรือทางร่างกาย หรือสังคม จะทำให้ผู้สูงอายุไม่สามารถรักษาสภาพสมดุลของร่างกายไว้มีผลทำให้เกิดโรคเรื้อรังและการเจ็บป่วยตามมา การเจ็บป่วยของผู้สูงอายุจะทำให้เกิดภาวะแทรกซ้อนมากกว่าวัยอื่น จากการศึกษาพบว่า ปัญหาที่สำคัญของผู้สูงอายุมีหลายโรคและหลายปัญหา และบางครั้งอาจเป็นปัญหาที่ไม่เกี่ยวข้องกับโรคที่ได้รับการวินิจฉัย ปัญหาหลักที่พบในผู้สูงอายุได้แก่ ภาวะสับสน ภาวะกลั้นการขับถ่ายไม่ได้ การหกล้ม ภาวะเสียสมดุล และผลข้างเคียงจากการใช้ยาหลายชนิด

กล้อง Kinect เป็นอุปกรณ์เสริมการใช้งานเครื่องเล่นเกม Xbox 360 ที่พัฒนาโดยบริษัท Microsoft มีความสามารถในการให้ภาพสี (RGB Image) ภาพความลึก (Depth Image) ที่กล้องตรวจจับ และสามารถรับฟังเสียงของผู้ใช้ได้ เมื่อนามาใช้งานร่วมกับซอฟต์แวร์คอมพิวเตอร์ที่เหมาะสม จะสามารถตรวจจับร่างกายของผู้ใช้ และสามารถจำแนกตำแหน่งข้อต่อต่างๆ ของร่างกายผู้ใช้ได้ โดยให้ข้อมูลตำแหน่งของข้อต่อในรูปแบบจุดพิกัดสามมิติ ดังนั้นจึงสามารถนำมาประยุกต์ใช้กับการตรวจจับการเคลื่อนไหวของร่างกายมนุษย์ แบบปราศจากการทาเครื่องหมายตามตำแหน่งต่างๆ บนร่างกายได้

เมื่อใช้กล้อง Kinect ในการตรวจจับการเคลื่อนไหว ข้อมูลตำแหน่งของข้อต่อของร่างกายที่ได้ จะมีความถูกต้องเฉพาะส่วนที่กล้องตรวจจับได้ หากบางส่วนของร่างกายถูกบดบัง ข้อมูลตำแหน่งของข้อต่อของส่วนนั้นจะเป็นค่าประมาณ มีความถูกต้องน้อย การทำให้ทุกส่วนของร่างกาย ถูกตรวจจับได้ตลอดเวลาจึงต้องมีการเพิ่มจำนวนกล้องในการตรวจจับ และจัดวางกล้องในหลายๆ มุมมอง เพื่อให้ครอบคลุมกับการตรวจจับการเคลื่อนไหวที่สนใจ แต่การทำเช่นนี้จะทำให้ได้ข้อมูลตำแหน่งของข้อต่อมากกว่าหนึ่งชุดในการตรวจจับคนหนึ่งคน และค่าตำแหน่งของข้อต่อที่ข้อต่อเดียวกันที่ได้จากกล้อง

แต่แต่ละตัวมีค่าไม่เท่ากัน เพราะมีตำแหน่งอ้างอิงซึ่งเป็นตำแหน่งที่ตั้งของกล้องแตกต่างกัน จึงจำเป็นต้องนำข้อมูลตำแหน่งที่ได้มาทำการปรับตำแหน่งอ้างอิง และทำการประมวลผล รวมกันให้เป็น

ข้อมูลชุดเดียว โดยการรวมข้อมูลตำแหน่งของข้อต่อนั้น จะต้องเพิ่มความถูกต้องของข้อมูลตำแหน่งของข้อต่อ และต้องทดแทนข้อมูลตำแหน่งที่ถูกบดบังในบางมุมมอง ด้วยข้อมูลตำแหน่งจากกล้องอื่นที่ไม่ถูกบดบัง เพื่อให้ได้ข้อมูลตำแหน่งของข้อต่อที่ประกอบรวมกันเป็นโครงร่างมนุษย์ที่มีความถูกต้องสมบูรณ์ สามารถนำไปใช้ในการตรวจจับท่าทางและ การเคลื่อนไหวได้ต่อไป ในส่วนของการพัฒนาระบบและโปรแกรมในการตรวจจับการเคลื่อนไหว เนื่องจากในขณะนี้ยังไม่มีวิธีการสำหรับการใช้กล้อง Kinect หลายตัว ในการตรวจจับการเคลื่อนไหวของร่างกายมนุษย์ตามเวลาจริงพร้อมกัน จึงต้องมีการออกแบบระบบและพัฒนาโปรแกรมขึ้นมาโดยเฉพาะ ซึ่งควร เป็นระบบที่รองรับการใช้กล้อง Kinect ในการตรวจจับได้ไม่จำกัด และเป็นโปรแกรมที่สามารถนำไปพัฒนาเพิ่มเติมได้

## 1.2 ความมุ่งหมายและวัตถุประสงค์ของโครงการ

1. เพื่อตรวจจับการล้ม ของผู้สูงอายุ โดยใช้กล้องตรวจจับการเคลื่อนไหวแบบ 3 มิติ ทำการตรวจจับโครงร่างโครงกระดูก Skeleton Tracking ของร่างกาย
2. เพื่อทำการช่วยเหลือและส่งสัญญาณเตือนแก่ผู้สูงอายุได้ทันที่
3. เพื่อศึกษาการเขียนโปรแกรมของบอร์ด Arduino เพื่อรับคำสั่งจาก Computer และควบคุมการแจ้งเตือนผ่าน LED ,ข้อความ และ เสียงผ่านลำโพง
4. เพื่อศึกษาการเขียนโปรแกรมการติดต่อสื่อสารระหว่าง Microsoft Visual Basic 2010 และ Arduino

## 1.3 สมมติฐานของการศึกษาวิจัย

Kinect มีลักษณะเด่นในการตรวจจับการเคลื่อนไหวของโครงสร้างกระดูกภายในร่างกาย เราจึงนำความสามารถของ Kinect มาตรวจจับลักษณะการล้มของมนุษย์ โดย Kinect จะทำการวิเคราะห์โครงกระดูกภายในร่างกาย (Skeleton) โดยไม่ต้องติดเซนเซอร์ที่ร่างกาย

## 1.4 ทฤษฎีหรือแนวคิดการวิจัย

แนวคิดมาจากผู้สูงอายุ เนื่องจากผู้สูงอายุยิ่งอายุมากขึ้นยิ่งมีสภาพร่างกายผกผันกับอายุ อาทิ เช่น โรคไขข้อกระดูกเสื่อม เบาหวาน อาจจะมีถึงสิ่งที่ไม่ใช่โรคคือ การล้ม จึงทำให้เราทำการศึกษาค้นคว้าเกี่ยวกับอุปกรณ์ในการช่วยเหลือผู้สูงอายุไม่ให้ล้ม เพราะการล้มของผู้สูงอายุอาจจะทำให้เกิดโรคอัมพฤกษ์ อัมพาต ได้

เราจึงมีแนวคิดในการหาพัฒนาโปรแกรมนี้เพื่อเป็นประโยชน์ในการช่วยเหลือผู้ป่วยและพัฒนาต่อยอดเพื่อให้สิ่งประดิษฐ์มีคุณภาพและปรับปรุงให้เหมาะสมต่อการใช้งาน

### 1.5 ขอบเขตของการวิจัย

1. เพื่อศึกษาและเฝ้าระวังการล้มของผู้สูงอายุ
2. มีการส่งสัญญาณเป็นเสียงและแสง เมื่อผู้สูงอายุเกิดการล้ม
3. ใช้ระบบจับภาพ 3 มิติ ควบคุมการทำงานและตรวจสอบ การล้มของผู้สูงอายุ ได้อย่างทันทีทัน

## บทที่ 2 ความรู้พื้นฐาน

### 2.1) ทฤษฎีเบื้องต้น

#### 2.1.1) การเปลี่ยนแปลงในวัยสูงอายุ

วัยสูงอายุจะมีการเปลี่ยนแปลงไปในลักษณะ การเสื่อมถอยทั้งทางด้านร่างกาย จิตใจ อารมณ์ และสังคม เป็นไปอย่างต่อเนื่องและมีการเปลี่ยนแปลงตลอดเวลา ซึ่งการเปลี่ยนแปลงดังกล่าว มีอิทธิพลต่อการดำเนินชีวิตของผู้สูงอายุการเปลี่ยนแปลงที่เกิดขึ้นในผู้สูงอายุโดยทั่วไป แบ่งเป็น 3 ด้าน ดังนี้

1. การเปลี่ยนแปลงทางด้านสรีรวิทยา (Physiological change) ภาวะผู้สูงอายุจะเกิดการเปลี่ยนแปลงไปในทางเสื่อมมากกว่าการเจริญเติบโต ซึ่งแต่ละคนจะเกิดไม่เท่ากัน เซลล์ต่างๆ ภายในร่างกายทำงานลดลงและมีจำนวนน้อยลง เมื่อเทียบกับวัยหนุ่มสาว ขนาดเซลล์ที่เหลือจะใหญ่ขึ้น ซึ่งจะเกิดขึ้นในทุกระบบ หน้าที่ของร่างกาย ความสามารถในการทำงานของอวัยวะในระบบต่างๆ ลดลง ผู้สูงอายุจะมีโอกาสเกิดการเจ็บป่วยเป็นโรคต่างๆได้ง่าย

2. การเปลี่ยนแปลงทางด้านจิตใจและอารมณ์ (Psychological change) การเปลี่ยนแปลงทางด้านร่างกายและสังคม จะทำให้จิตใจของผู้สูงอายุเปลี่ยนไปด้วย ผู้สูงอายุจะปรับจิตใจและอารมณ์ทั้งทางด้านบวกและด้านลบไปตามการเปลี่ยนแปลงของร่างกายและสิ่งแวดล้อม ในด้านบวกผู้สูงอายุจะมีการปรับปรุง และพัฒนาจิตใจในทางที่ดีงามมากขึ้น สามารถควบคุมจิตใจได้ดีกว่าหนุ่มสาว ในด้านลบได้แก่ การสูญเสีย บุคคลอันเป็นที่รักเช่น คู่ชีวิต การสูญเสียความสามารถทางเพศ การสูญเสียสัมพันธ์ภาพในครอบครัวจาก บุตรหลานที่แยกย้ายออกไป ทำให้ผู้สูงอายุเกิดความวิตกกังวล แต่การแสดงออกขึ้นอยู่กับลักษณะของคนๆ นั้น การศึกษา ประสบการณ์และสิ่งแวดล้อมในวัยเด็กและผู้ใหญ่ของคนๆ นั้น

3. การเปลี่ยนแปลงทางสังคม (Social change) วัยสูงอายุจะมีการปฏิสัมพันธ์กับสังคมเริ่มลดลงทั้งนี้จากภาระหน้าที่และบทบาทในสังคมเริ่มลดลง ทำให้ผู้สูงอายุมีความยากลำบากในการปรับตัว จนก่อให้เกิดความเจ็บป่วยทางร่างกายและจิตใจตามมา

สรุปได้ว่า บุคคลเมื่อมีอายุมากขึ้นจะเกิดการเปลี่ยนแปลงในทางที่เสื่อมลง โดยเฉพาะวัยสูงอายุจะ พบการเปลี่ยนแปลงไปในทางเสื่อมลงที่ชัดเจนกว่าวัยอื่น ทั้งทางด้านร่างกาย จิตใจ และสังคม แต่การ เปลี่ยนแปลงจะมากหรือน้อยยังขึ้นอยู่กับหลายๆปัจจัย เช่น เพศ อายุความสามารถในการปรับตัวของแต่ละ บุคคลการมีพฤติกรรมสุขภาพที่ดีเป็นต้น

### 2.1.2) การพลัดตกหกล้ม

การพลัดตกหกล้ม หมายถึง การที่ร่างกายเปลี่ยนท่าจากท่ายืนเป็นท่านั่ง หรือท่านอนอย่างรวดเร็วโดยไม่ได้ตั้งใจ หรือภาวะหกล้ม หมายถึง การที่บุคคลลงไปอยู่ในท่าพักโดยไม่ตั้งใจบนพื้น หรือในระดับที่ต่ำกว่าเดิม และจากผลที่ตามมาของการถูกชนหรือตีอย่างแรง การหมดสติ การเกิดอัมพาตอย่างทันทีทันใดหรือการชัก

### 2.1.3) ปัจจัยส่งเสริมภาวะหกล้มในผู้สูงอายุ

การเกิดภาวะหกล้มในผู้สูงอายุมีรายงานที่ระบุถึงปัจจัยที่เกี่ยวข้องหลายปัจจัย โดยแบ่งกลุ่มปัจจัย ออกเป็น 2 กลุ่มดังนี้

#### 1. ปัจจัยภายในบุคคล ประกอบด้วยปัจจัยดังต่อไปนี้

1.1 อายุ ผลการศึกษาพบว่าบุคคลที่มีอายุมากกว่า 65 ปี มีโอกาสเสี่ยงต่อการเกิดการหกล้มได้มาก อย่างไรก็ตามผลการศึกษายังไม่สามารถสรุปได้เนื่องจากมีบางการศึกษาที่ไม่พบความสัมพันธ์ระหว่างอายุกับอุบัติการณ์การเกิดการหกล้ม

1.2 ภาวะพร่องด้านการรับรู้และสติปัญญา เป็นปัจจัยที่สำคัญของการเกิดภาวะหกล้มในผู้สูงอายุภาวะพร่องประกอบด้วยภาวะสับสน แต่ภาวะความจำและความเข้าใจลดลงเป็นปัจจัยที่ทำให้ภาวะเสี่ยงต่อการหกล้มเพิ่มมากขึ้น

1.3 ประวัติการหกล้ม เป็นปัจจัยที่พบว่ามีความสัมพันธ์กับการเพิ่มขึ้น ของความเสี่ยงต่อการหกล้ม

1.4 การเข้ายาที่มีผลกระทบต่อระบบประสาทส่วนกลาง เช่น ยานอนหลับหรือยาระงับประสาท จะเพิ่มความเสี่ยงต่อการเกิดการหกล้ม แต่ผลการศึกษาบางการศึกษาที่เป็นการศึกษาเชิงพรรณนารายงานว่ายาบางประเภท เช่น ยาไดจอกซิน (digoxin) ยาแก้ชัก ยาขับปัสสาวะ และยาลดความดันโลหิตสูง มีส่วนในการส่งเสริมให้เกิดการหกล้มได้ แต่ยังไม่มียารายงานจากการศึกษาที่น่าเชื่อถือยืนยัน

1.5 ปัจจัยด้านร่างกาย เช่น ภาวะอ่อนแรง การเคลื่อนไหวลดลง และการประสานงานลดลง มีความสัมพันธ์กับการหกล้ม

1.6 ปัญหาการขับถ่ายปัสสาวะ เช่น ภาวะกลั้นปัสสาวะไม่อยู่ หรือปัญหาถ่ายเหลว เป็นปัจจัยเสี่ยงสำคัญของภาวะหกล้ม ในขณะที่การได้รับยาขับปัสสาวะเป็นปัจจัยส่งเสริมให้ปัญหาดังกล่าวรุนแรงมากขึ้น

1.7 ปัจจัยอื่นๆ ที่มีรายงานว่าสาเหตุของการหกล้มในผู้สูงอายุ ได้แก่ การได้รับสารละลายทางหลอดเลือดดำ ภาวะมึนงง การใช้สารเสพติด ภายหลังจากผ่าตัด ภาวะนอนไม่หลับ ผู้ป่วยโรคหลอดเลือดสมอง เป็นต้น

#### 2. ปัจจัยภายนอกบุคคล

2.1 ตำแหน่งที่มีการหกล้มรายงานพบว่าตำแหน่งที่ผู้ป่วยหกล้มมักจะเป็นตำแหน่งเดิมๆมีรายงานพบว่ามีถึงร้อยละ 43 ที่การหกล้มเกิดข้างเตียงที่เหลือเกิดในห้องน้ำห้องส้วมหรือทางเดิน

2.2 ระยะเวลาของการหกล้มระยะเวลาที่เกิดการหกล้มบ่อยมีรายงานคือเวลา กลางวัน โดยเฉพาะช่วงสายๆและบ่ายๆซึ่งมีรายงานถึงความสัมพันธ์ระหว่างช่วงเวลาหกล้มกับ ช่วงเวลาที่มีกิจกรรม

2.3 กิจกรรมขณะหกล้มมีรายงานว่า การหกล้มเกิดขณะเคลื่อนย้ายจากตำแหน่ง หนึ่งไปยังอีกตำแหน่งหนึ่งมากที่สุดเช่นการเดินการเข้าห้องน้ำการนั่งบนเก้าอี้

2.4 ลักษณะพื้นมีรายงานการเกิดการหกล้มบนพื้นไวนิลแต่อย่างไรก็ตามบนพื้น ห้องน้ำและห้องส้วมก็พบรายงานการหกล้มเช่นเดียวกับพื้นแห้ง

2.5 ปัจจัยด้านสิ่งแวดล้อมเช่นลักษณะไม้กั้นเตียงพื้นลื่นหรือแสงสว่างไม่เพียงพอ

#### 2.1.4) การประเมินความเสี่ยงต่อการพลัดตกหกล้ม

1. ประวัติการหกล้ม การกั้วการหกล้ม การกลิ้งปัสสาวะและอุจจาระไม่ได้
2. สติปัญญาและการรับรู้ลดลง ภาวะอารมณ์ ภาวะสับสน การทำหน้าที่ลดลง การใช้จ่าย
3. ปัญหาสุขภาพทางอายุรกรรม ปัจจัยเสี่ยงด้านสิ่งแวดล้อม

#### 2.1.5) ผลกระทบของการหกล้ม

การหกล้มส่งผลกระทบต่อผู้สูงอายุดังต่อไปนี้

1. ผลกระทบด้านร่างกายการหกล้มนำมาสู่การได้รับบาดเจ็บเช่นกระดูกหักเลือดคั่งได้ สมองและการบาดเจ็บของเนื้อเยื่อทำให้ผู้สูงอายุไม่เคลื่อนไหวนำมาสู่การเกิดภาวะแทรกซ้อนเช่นการ ติดเชื้อข้อติดแข็งและแผลกดทับการมีลิ้มเลือดอุดตันบริเวณหลอดเลือดดำลึกนอกจากนั้นถ้ารุนแรง อาจนำมาซึ่งการเสียชีวิตได้
2. ผลกระทบด้านจิตใจทำให้ผู้สูงอายุขาดความมั่นใจในตนเองและกลัวการหกล้มซ้ำทำให้ เกิดการพึ่งพามากขึ้นทำให้เป็นภาระของครอบครัวตามมา
3. ผลกระทบด้านเศรษฐกิจและสังคมการเสียค่าใช้จ่ายในการรักษาทำให้เป็นภาระต่อ ครอบครัว

#### 2.1.6) การป้องกันการหกล้มของผู้สูงอายุนที่บ้าน

1. ปัจจัยด้านภายในตัวบุคคล
  - 1.1 ไม่เคลื่อนไหวร่างกายโดยไม่มีผู้ช่วยเหลือขณะร่างกายอ่อนแอ
  - 1.2 หลีกเลี่ยงการไปในที่ที่เป็นอันตราย
  - 1.3 การออกกำลังกายอย่างสม่ำเสมอ
  - 1.4 การไปพบแพทย์เพื่อตรวจร่างกายประจำปี
  - 1.5 การรับประทานยาตามแผนการรักษาของแพทย์
  - 1.6 รับประทานอาหารครบส่วนและเพียงพอ
  - 1.7 การนอนหลับพักผ่อนให้เพียงพอ
  - 1.8 การเคลื่อนไหวและเปลี่ยนท่าช้าๆ
  - 1.9 สำรองสิ่งของที่เป็นอันตรายภายในบ้านเป็นประจํา

1.10 ใช้อุปกรณ์ในการเคลื่อนไหวที่เหมาะสม

1.11 สวมเครื่องแต่งกายเหมาะสม

## 2. ปัจจัยด้านสิ่งแวดล้อม

2.1 จัดให้มีแสงสว่างและสีอย่างเหมาะสมสถานที่ที่พบว่าแสงสว่างมักไม่เพียงพอ เช่นบันไดทางเข้าบ้านทางเดินภายในและภายนอกบ้านบริเวณที่มีแสงจ้ามากเกินไปเช่นบริเวณที่มีกระจกพื้นที่ขัดมันเป็นเงาสะท้อนหรือบริเวณที่แสงอาทิตย์ส่องตรงสำหรับสีที่ไม่เหมาะสมได้แก่สีกลมกลืนสีที่ประกอบกันเป็นลวดลายแบบแผนที่สับสนเช่นสีของพรมหรือกระเบื้องปูพื้นหรือสิ่งของตกแต่งผนัง เป็นต้น

2.2 จัดพื้นที่บ้านให้เหมาะสมพื้นที่เป็นอันตรายได้แก่พื้นผิวที่ไม่เรียบมีลวดลายหลอกตามีสสิ่งปกคลุมที่ทำให้ลื่นหรือสะดุดเช่นบริเวณทางเดินพื้นบ้านที่ไม่เรียบการลงน้ำมันพื้นลื่นเกินไปมีสิ่งเกะกะบนพื้นพื้นห้องน้ำที่เปียกอยู่เสมอพรมปูพื้นที่ไม่มีที่ยึดเกาะ

2.3 จัดสิ่งก่อสร้างภายในให้เหมาะสมเช่นบันไดควรมีราวยึดชั้นบันไดควรมีความสูงที่เหมาะสมสูงสม่ำเสมอและกว้างเพียงพอไม่ควรมีลวดลายควรมีขอบเขตแสดงเครื่องหมายให้ชัดเจนห้องน้ำควรมีราวเกาะยึดไม่ควรเป็นที่นั่งแบบนั่งยองๆอ่างล้างหน้าไม่ควรต่ำเกินไป

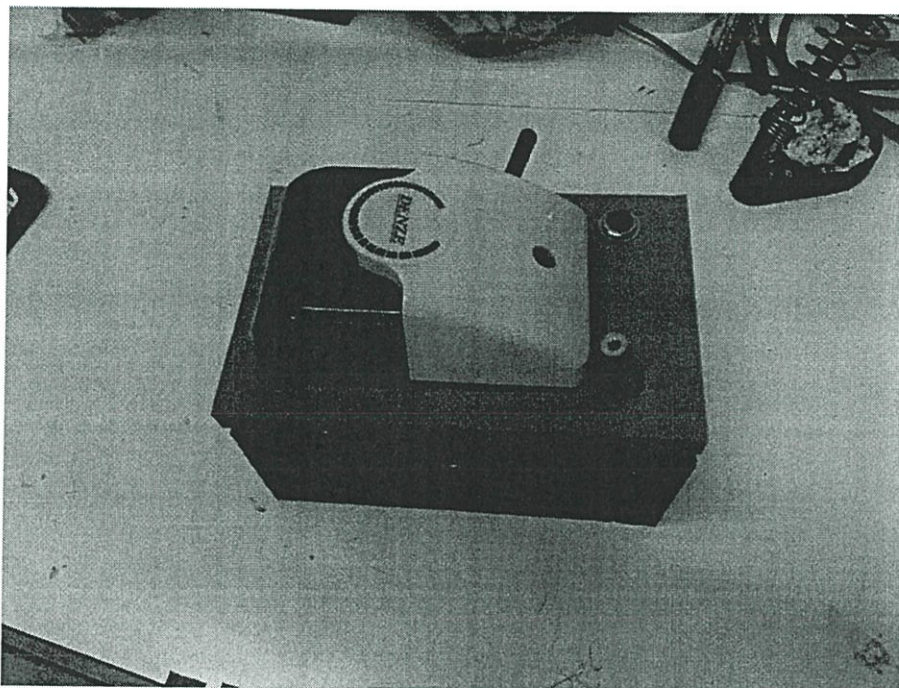
2.4 ควรจัดเครื่องเรือนให้เหมาะสมจัดให้เป็นระเบียบไม่ควรเกะกะขวางทางเดินเครื่องเรือนควรมีความมั่นคงขณะใช้งาน

2.5 เครื่องแต่งกายและเครื่องช่วยในการเคลื่อนไหวควรมีความเหมาะสมเสื้อผ้าที่ใส่ควรพอดีตัวไม่ใหญ่หรือยาวเกินไปเพราะทำให้มีโอกาสสะดุดหรือเกี่ยวดึงได้ง่ายรองเท้าควรใส่พอดีเท้าอุปกรณ์ช่วยเคลื่อนไหวต่างๆเช่นไม้เท้าล้อเข็นควรมีขนาดที่เหมาะสมและใช้ได้อย่างถูกต้อง

## 2.2) อุปกรณ์และโปรแกรม

### 2.2.1) เครื่องเตือนคนล้ม

กล่องเตือนคนล้ม เป็นกล่องที่ออกแบบด้วยโปรแกรม Solid work และนำไปทำการขึ้นรูปด้วยเครื่องขึ้นรูป 3 มิติ ดังรูปที่ 2.1



รูปที่ 2.1 เครื่องเตือนคนล้ม

### 2.2.2) Arduino Uno

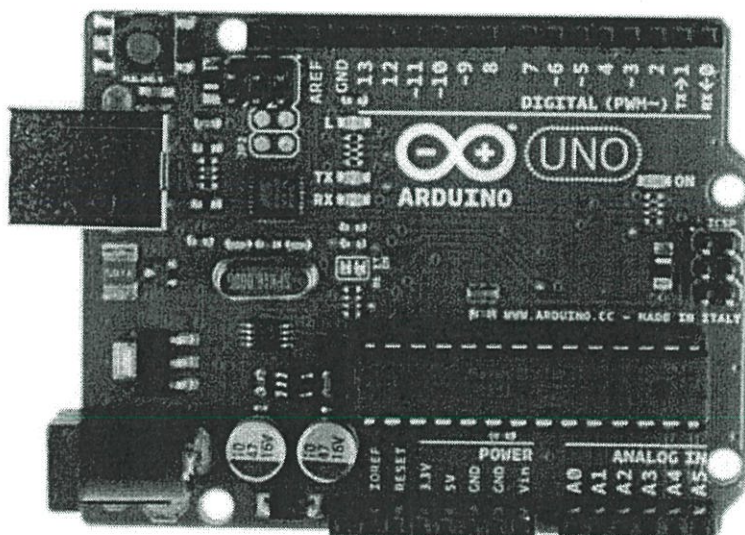
คือเครื่องมือแพลตฟอร์มเปิดที่ใช้ในการควบคุมสิ่งต่างๆโดยใช้ไมโครคอนโทรลเลอร์แบบง่ายในการเขียน โปรแกรมควบคุมโดยการเขียนโปรแกรมผ่านแพลตฟอร์ม Arduino เข้าสู่บอร์ดควบคุม

Arduino สามารถพัฒนาให้สื่อสารกับวัตถุอื่นๆสามารถรับข้อมูลจากสวิทช์หรือว่าจะเป็นข้อมูลจากเซนเซอร์ หรือว่าใช้ในการควบคุมแสงไฟมอเตอร์และอุปกรณ์อิเล็กทรอนิกส์อื่นๆอีกมากโปรเจกต์ที่พัฒนาโดยArduino สามารถทำงาน แยกโดดเดี่ยวออกมาได้หรือว่าจะสื่อสารหากันก็ทำได้

ภาษาที่ใช้ใน Arduino จะใช้ภาษา Visual Basic ในการเขียนโปรแกรมควบคุมซึ่งเหมาะสมกับผู้เริ่มต้นเขียนโปรแกรมเพราะเป็นภาษาเริ่มต้นที่ง่ายในการศึกษา

### คุณสมบัติของบอร์ด Arduino Uno

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by boot loader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz



รูปที่ 2.2 บอร์ด Arduino Uno

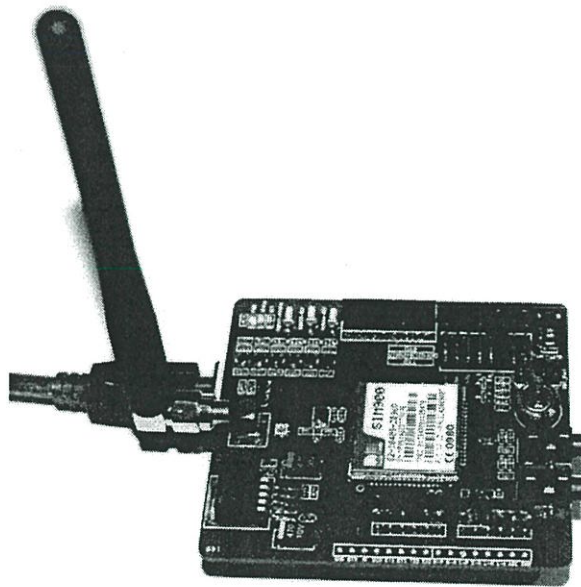
(ที่มา : <http://www.arduino.cc/en/Main/ArduinoBoardUno>)

### 2.2.3) SIM900 GSM/GPRS Arduino shield

GSM Module SIM900 Shield สำหรับ Arduino ใช้ได้เลยไม่ต้องต่อสายเพิ่ม เสียบเข้ากับ Arduino แล้วเขียนคำสั่งควบคุมโดยใช้ AT-Command สามารถควบคุมให้เป็นโทรศัพท์หรือต่ออินเทอร์เน็ตได้ตามต้องการ

#### คุณสมบัติของบอร์ด SIM900 GSM/GPRS Arduino shield

- Quad-Band 850/900/1800/1900MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 2/2+
- Class 4 (2W@850/900MHz)
- Class 1 (1W@1800/1900MHz)
- Control via commands (GSM 07.07, 07.05 and SIMCOM enhanced AT Commands)
- Short message service
- Free serial port selection
- All SIM900 pins breakout
- RTC supported with Super Cap
- Power on/off and reset function supported by Arduino interface



รูปที่ 2.3 SIM900 GSM/GPRS Arduino shield

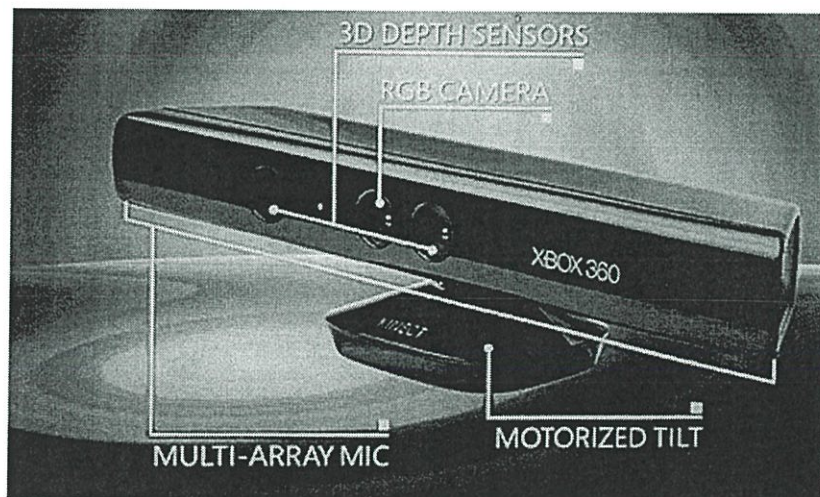
(ที่มา : <http://www.arduinoall.com>)

### 2.2.4) Kinect X-Box 360

Kinect คือ อุปกรณ์เสริมของเครื่องเล่นเกม Xbox360 จาก Microsoft โดยสามารถจดจำผู้เล่น (Facial Recognition) และให้ผู้เล่นควบคุมเกมส์ผ่านทางท่าเคลื่อนไหวร่างกายของผู้เล่น โดยตรง(3D Motion Recognition)โดยไม่จำเป็นต้องมี joystickอีกต่อไป และสามารถจดจำเสียงของผู้เล่นได้ (Voice Recognition) และนอกจากใช้ฟังก์ชันพวกนี้เล่นเกมแล้วเรายังใช้ Kinect ควบคุมการดูหนัง ฟังเพลง

ส่วนประกอบต่างๆ ของ Kinect

1. กล้อง CMOS RGB ธรรมดา
2. กล้อง IR (Infra Red) 1 ตัว
3. ตัวส่งแสง IR (Infra Red)
4. ไมโครโฟน 4 ตัว
5. มีมอเตอร์ไว้ปรับมุมกล้อง



รูปที่ 2.4 ส่วนประกอบต่างๆ ของ Kinect

(ที่มา : <https://kinectasia.wordpress.com>)

หลักการทำงานของ Kinect : Kinect มีการหน้าที่ใหญ่ๆอยู่ 3 หน้าที่ ดังนั้นเราจะเชื่อมโยงระหว่าง Hardware กับหน้าที่การทำงานของ Kinect ดังนี้

1. จดจำผู้เล่น : จดจำโดยใช้ ข้อมูลจากกล้อง CMOS RGB (ข้อมูล แดง เขียว น้ำเงิน) และประมวลผลโดยวิธี Facial Recognition

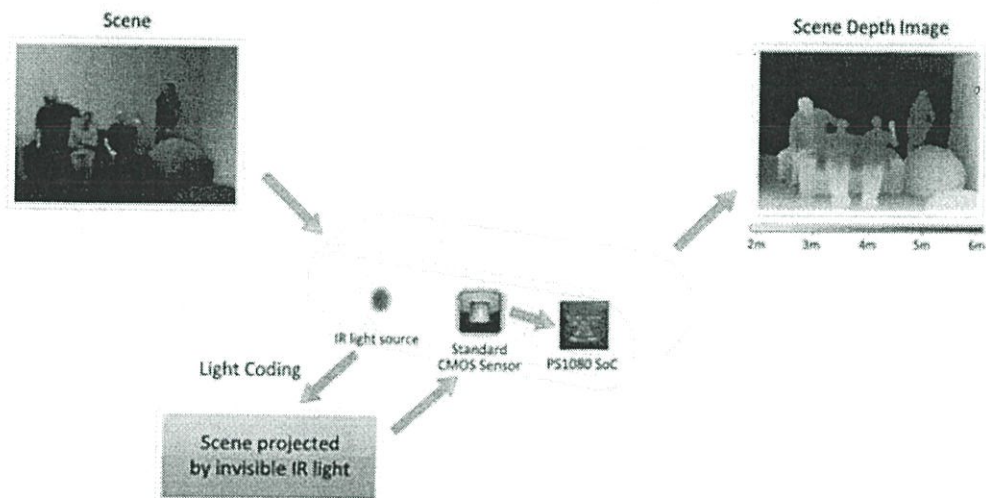
2. จดจำการเคลื่อนไหวร่างกายของผู้เล่นเป็นแบบ 3 มิติ: แบ่งเป็น 3 อย่างย่อย

2.1 จดจำวัตถุแบบ 3 มิติ : โดยใช้ตัวส่งแสง IR และกล้อง IR ตัวส่งแสง IR จะส่งแสงไปกระทบกับวัตถุ ไม่ว่าจะเป็นคนหรือสิ่งของ และแสง IR จะสะท้อนจากวัตถุนั้นๆกลับไปทีกล้อง IR และ Kinect จะใช้ข้อมูลในเรื่อง “เวลาในการสะท้อนกลับ” และ “ความยาวคลื่นแสง” ในการประมวลผลวัตถุต่างๆเป็น 3 มิติ (ยกตัวอย่างเช่น ใช้เวลานานในการสะท้อนกลับแสดงว่าวัตถุอยู่ไกลของที่มีสีต่างกันจะดูดกลืนและสะท้อนแสงกลับไปในความยาวคลื่นที่ต่างกัน)

2.2 จัดจำและแบ่งแยกประเภทของคนว่าเป็นเพศไหน อายุประมาณเท่าไร ขนาด สัดส่วนตัวประมาณไหน ใส่เสื้อผ้ายังไง โดยใช้เทียบกับฐานข้อมูลที่มีอยู่

2.3 จัดจำการเคลื่อนไหวของคน โดยหลังจากแบ่งแยกประเภทของคนแล้ว จะใช้วิธี skeletal movements (การเคลื่อนไหวตามลักษณะกระดูก) เข้าไปวิเคราะห์คนประเภทนั้นว่า ส่วนไหนคือหัว ตัว แขน ขา ข้อศอก โดยใช้สัดส่วน ลักษณะ หรือข้อจำกัดต่างๆ

3. จัดจำเสียงผู้เล่น ใช้ไมโครโฟน 4 ตัว โดยไมโครโฟนนั้นจะเป็นแบบ Wide-Field, conic audio capture รับเสียงในพื้นที่กว้าง และให้ความสำคัญกับเสียงเป็นรูปกรวย (สนใจเสียงไกลๆ เสียงใกล้ๆ อย่างเสียงที่มาจาก Xbox เอง Kinect จะไม่สนใจ) หลังจากนั้นจะใช้วิธี Voice Recognition เพื่อจัดจำและแปลความหมายของสิ่งที่ผู้เล่นแต่ละคนพูด โดย Kinect สามารถแยกแยะเสียงของแต่ละผู้เล่นได้



รูปที่ 2.5 หลักการทำงานของ Kinect

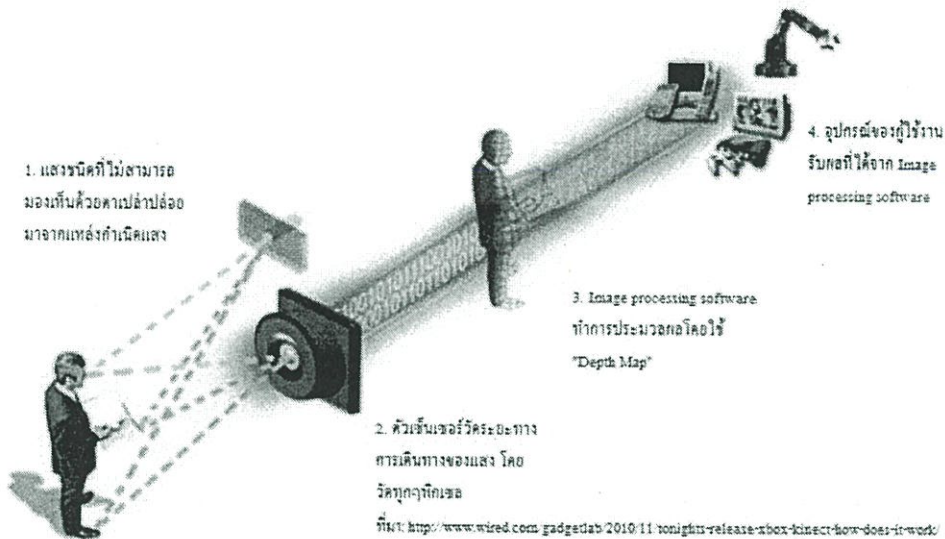
(ที่มา : <https://kinectasia.wordpress.com>)

### ตารางที่ 2.1 ตารางคุณสมบัติของ Kinect

Property	Spec
Field of View (Horizontal, Vertical, Diagonal)	58° H, 45° V, 70° D
Depth image size	VGA (640×480)
Spatial x/y resolution (@ 2m distance from sensor)	3mm
Depth z resolution (@ 2m distance from sensor)	1cm

Property	Spec
Maximum image throughput (frame rate)	60fps
Operation range	0.8m – 3.5m
Color image size	UXGA (1600x1200)
Audio: built-in microphones	Two microphones
Audio: digital inputs	Four inputs
Data interface	USB 2.0
Power supply	USB 2.0
Power consumption	2.25W
Dimensions (Width x Height x Depth)	14cm x 3.5cm x 5cm
Operation environment (every lighting condition)	Indoor
Operating temperature	0°C – 40°C

## หลักการการทำงานของระบบตรวจวัดระยะ



รูปที่ 2.6 หลักการทำตรวจวัดระยะทางของ Kinect

(ที่มา : <https://kinectasia.wordpress.com>)

ในการวัดระยะหรือความลึกของวัตถุต่าง ๆ นั้นอาศัยการทำงานของอุปกรณ์เซ็นเซอร์วัดระยะ (Depth Sensor) ซึ่งประกอบอุปกรณ์ 2 ชนิดคือ Infrared Laser Projector และ Monochrome CMOS Sensor โดยตัว Infrared Laser Projector จะส่งแสงอินฟราเรดไปกระทบกับวัตถุที่อยู่ในแนวรัศมี เมื่อแสงอินฟราเรดกระทบกับวัตถุก็จะสะท้อนกลับมาที่ตัวรับ Monochrome CMOS Sensor จากนั้นซอฟต์แวร์ก็จะคำนวณระยะทางอาศัยเวลาการเดินทางของแสงอินฟราเรด

### Kinect สำหรับ Windows

ทางบริษัท Microsoft ได้ปล่อยชุดพัฒนา (Software Development Kit, SDK) สำหรับ Windows 7 เมื่อวันที่ 16 มิถุนายน 2554 ซึ่งตัว SDK จะมีไดรเวอร์ต่างๆ ของ Kinect สำหรับคอมพิวเตอร์พีซีรวมอยู่ด้วย ซึ่งโปรแกรมที่สามารถนำมาใช้พัฒนาแอปพลิเคชัน Kinect ได้แก่ C++ C# และ Visual Basic ส่วนฟีเจอร์ต่างๆ ที่ถูกรวมไว้ใน SDK ด้วย ได้แก่

1. Raw Sensor Streams สำหรับการนำข้อมูลดิบที่ได้จากเซ็นเซอร์ต่างๆ ไปใช้งาน
2. Skeletal Tracking สำหรับติดตามโครงร่างของมนุษย์ ซึ่งสามารถติดตามได้สูงสุด 2 คนพร้อมกัน ความโดดเด่นของฟีเจอร์นี้คือ สามารถนำไปพัฒนาเป็นแอปพลิเคชันควบคุมด้วยท่าทาง (Gestures Control)
3. Advanced Audio Capabilities ใช้ในการจำแนกเสียงต่างๆ ซึ่งสามารถกำจัดเสียงรบกวนต่างๆ (Noise Suppression and Echo Cancellation) และแยกเสียงของแต่ละบุคคลได้ ซึ่งฟีเจอร์นี้สามารถนำไปพัฒนาเป็นการควบคุมด้วยเสียง (Voice Recognition)
4. โค้ดโปรแกรมตัวอย่าง และ เอกสารประกอบ

## System Requirements

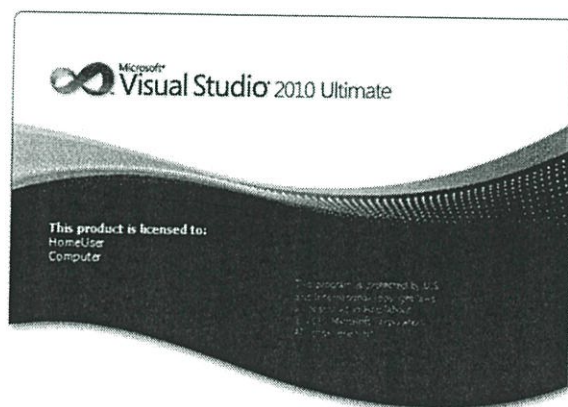
1. ระบบประมวลผลแบบ 32 หรือ 64 บิต
2. ระบบประมวลผลขั้นต่ำ Dual-core 2.66 GHz
3. ช่อง USB 2.0 (ห้ามใช้ร่วมกับอุปกรณ์อื่น)
4. หน่วยความจำอย่างต่ำ 2 GB
5. ระบบปฏิบัติการ Windows 7

การพัฒนาแอปพลิเคชันนั้นสามารถใช้ได้ทั้ง Kinect ที่มากับเครื่อง Xbox 360 หรือจะใช้ Kinect สำหรับ Windows โดยราคาของ Kinect สำหรับ Xbox นั้นจะมีราคาถูกกว่ามาก หากใช้ Kinect ที่มาพร้อม Xbox รุ่นที่เป็น Xbox 360 (slim + kinect) หรือ Xbox 360 (slim) ต้องใช้อุปกรณ์เสริมคือ Kinect Power Cable เพราะไม่มีแถมมาให้ แต่ถ้าเป็น Xbox 360 รุ่นเก่าก็จะมาพร้อมสาย USB และ สาย Power ซึ่งพร้อมใช้งานได้ทันที

ส่วนความแตกต่างระหว่างของเวอร์ชันสำหรับ Xbox กับ Windows คือ Kinect สำหรับ Windows นั้นสามารถนำไปพัฒนาเพื่อธุรกิจได้ และ ระบบ Near Mode นั้นจะใช้ได้แต่ในรุ่นที่เป็น Windows เท่านั้น เพราะรุ่น Windows สามารถใช้งานได้ระยะไกลที่สุดคือ 40 เซนติเมตร แต่รุ่น 360 ใช้ได้ไกลสุดที่ระยะ 80 เซนติเมตร ส่วนฮาร์ดแวร์อื่นๆนั้นแตกต่างกันเล็กน้อย

### 2.2.5) โปรแกรม Microsoft Visual Studio 2010

Microsoft Visual Studio คือ Integrated Development Environment พัฒนาขึ้นโดยไมโครซอฟท์ ซึ่งเป็นเครื่องมือที่ช่วยนักพัฒนาซอฟต์แวร์พัฒนาโปรแกรมคอมพิวเตอร์ web site, web application และ Web service ระบบที่รองรับการทำงานนั้นมี Microsoft Windows Pocket Pc, Smartphone และ Web browser ในปัจจุบัน Visual Studio นั้นสามารถใช้ภาษาโปรแกรมที่เป็น .Net ในโปรแกรมเดียวกันเช่น VB.NET C++ C# J# เป็นต้น



รูปที่ 2.7 โปรแกรม Microsoft Visual Studio 2010

(ที่มา : [www.thaicreate.com](http://www.thaicreate.com))

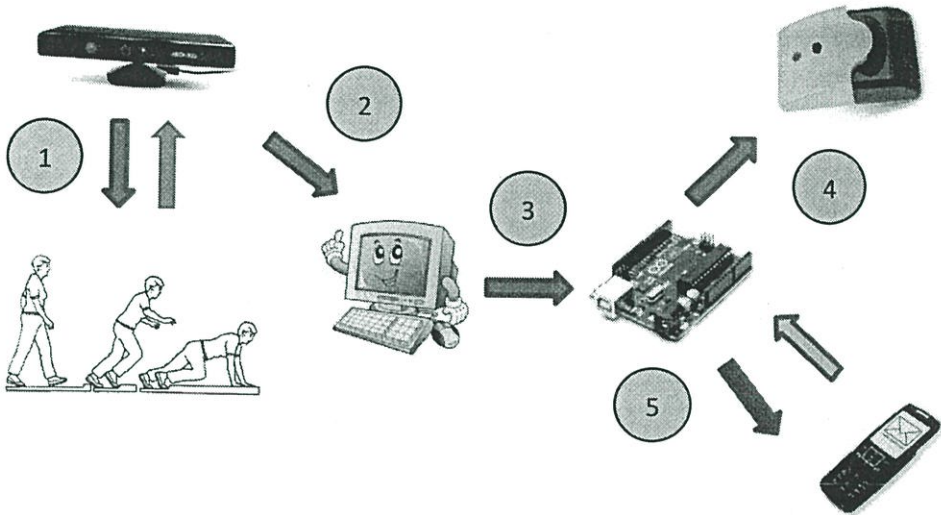
### ขั้นตอนการเขียนโปรแกรมด้วย Visual Basic .NET

1. ออกแบบหน้าจอของโปรแกรม โดยนำคอนโทรลต่างๆ จากtoolbox มาวางบนฟอร์ม แล้วปรับขนาดและตำแหน่งตามที่ต้องการ
2. กำหนดProperty ของฟอร์มและออบเจ็กต์ต่างๆ บนฟอร์ม โดยใช้วินโดว์ Properties
3. การเขียนโค้ดควบคุมการทำงานของโปรแกรม
4. รันโปรแกรมเพื่อทดสอบว่าโปรแกรมทำงานได้ตามที่ต้องการหรือไม่
5. ค้นหาข้อผิดพลาด ถ้าหากโปรแกรมเกิดข้อผิดพลาดหรือทำงานไม่ได้ตามที่ต้องการ
6. คอมไพล์Project ไปเป็นไฟล์โปรแกรมที่รันได้ Executable File

บทที่ 3

กระบวนการทำงานและขั้นตอนการทดสอบการทำงาน

3.1) กระบวนการในการทำงาน



รูปที่ 3.1 ภาพแสดงกระบวนการในการทำงาน

- จาก
- (1) แสดงการทำงานของ Kinect โดยการจับคนล้ม และมีการส่งค่ากลับไปยัง Kinect
  - (2) ค่าที่ได้จาก Kinect ส่งต่อไปยัง Computer เพื่อทำการวิเคราะห์และประมวลผลข้อมูล
  - (3) Arduino รับค่าจาก Computer เพื่อทำการแจ้งเตือน
  - (4) Arduino รับค่า และทำการแจ้งเตือนผ่าน Siren เมื่อมีคนล้ม
  - (5) Arduino รับค่าและทำการแจ้งเตือนผ่าน SMS และสามารถโทรเข้าเพื่อเช็คสถานะของคนที่ล้มได้

กระบวนการทำงานของโปรแกรมมีดังต่อไปนี้

- 1) Kinect จะทำการตรวจจับลักษณะท่าทางของบุคคลนั้นๆ เพื่อตรวจหาลักษณะของคนล้มในลักษณะท่าทางต่างๆ
- 2) ลักษณะท่าทางคนล้ม เราจะแบ่งออกได้ 3 แบบ ดังต่อไปนี้
  - 2.1) ลักษณะท่าทางของคนปกติ



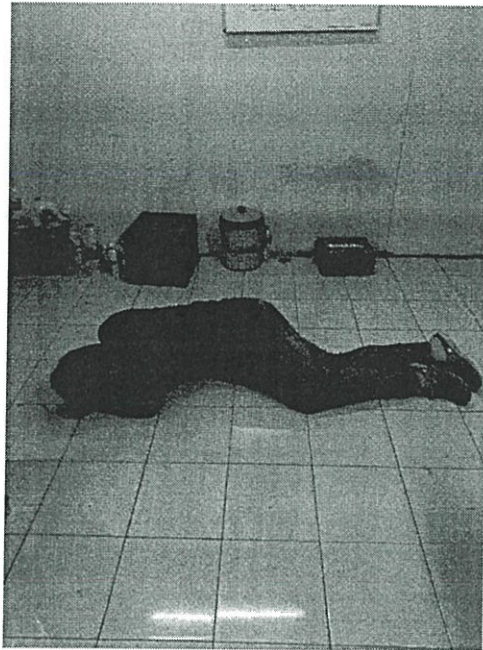
รูปที่ 3.2 ลักษณะท่าทางของคนปกติ

- 2.2) ลักษณะท่าทางของคนไถ่จะล้มหรือเกือบล้ม



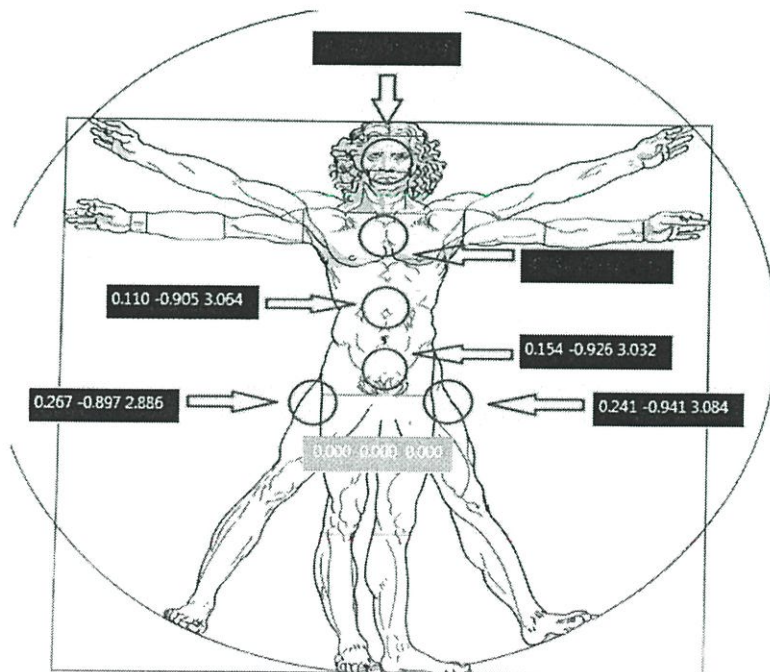
รูปที่ 3.3 ลักษณะท่าทางของลักษณะคนไถ่จะล้มหรือเกือบล้ม

### 2.3) ลักษณะท่าทางของคนที่ยิ้มและหมดสติ



รูปที่ 3.4 ลักษณะของคนที่ยิ้มและหมดสติ

- 3) Computer เมื่อ Kinect ทำการ จับภาพได้ ภาพที่ได้จะถูกนำไปประมวลผลข้อมูลและทำการบันทึกค่าที่ได้ นำไปเขียนโปรแกรม Computer โดยใช้โปรแกรม Microsoft Visual Studio 2010 ในการเขียนโปรแกรม ดังภาพตัวอย่าง



รูปที่ 3.5 แสดงการจับภาพจาก Kinect และค่าที่ได้จากการทดลอง

ตารางที่ 3.1 ตัวอย่างตารางบันทึกค่า หน่วยเป็นเมตร

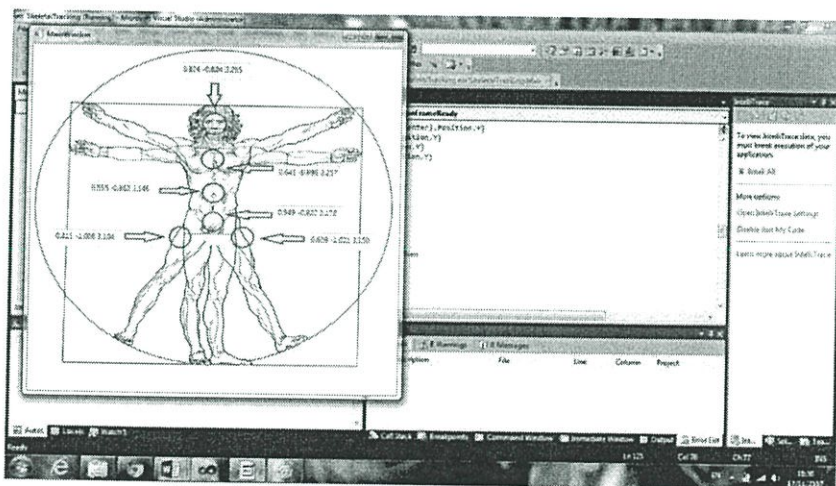
ที่ Z ประมาณ 1.5 ขณะยืน									
	ครั้งที่ 1			ครั้งที่ 2			ครั้งที่ 3		
ตำแหน่ง/พิกัด	X	Y	Z	X	Y	Z	X	Y	Z
Head	-0.16	-0.778	3.156	-0.194	-0.745	3.138	0.08	-0.798	3.205
ShoulderCenter	-0.078	-0.937	3.033	-0.117	-0.896	2.99	0.04	-0.938	3.035

หมายเหตุ x คือความกว้าง , y คือความสูงต่ำ และ z คือระยะห่างจากตัว Kinect

- 4) Arduino รับข้อมูลหรือคำสั่งจากคอมพิวเตอร์เพื่อทำการแจ้งเตือนต่อไปยังไฟ LED และ ลำโพง
- 5) Siren จะทำหน้าที่เป็นเสียงเตือนเพื่อให้บุคคลรอบข้างๆ ได้ยิน เพื่อเข้ามาช่วยเหลือให้ได้ทันที
- 6) โทรศัพท์มือถือ ใช้รับข้อความจาก Arduino เมื่อมีคนล้ม และ โทรแจ้งสภาพการล้มของผู้สูงอายุว่าล้มจริงหรือไม่
- 7) ลำโพง ใช้สำหรับสื่อสารการได้ยินเสียงจากโทรศัพท์มือถือ

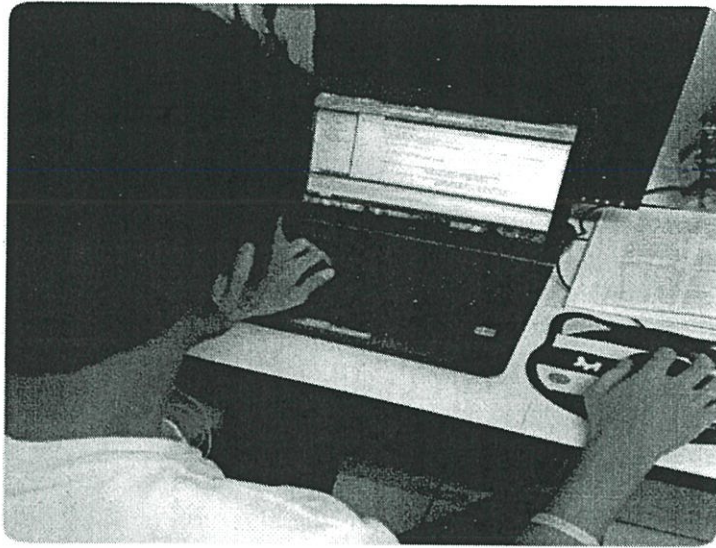
### 3.2) ขั้นตอนการทดสอบ

- 1) ศึกษาหาข้อมูลต่างๆ เกี่ยวกับโปรเจค
- 2) ทดสอบการทำงานของ Kinect โดยใช้โปรแกรม SDK Kinect



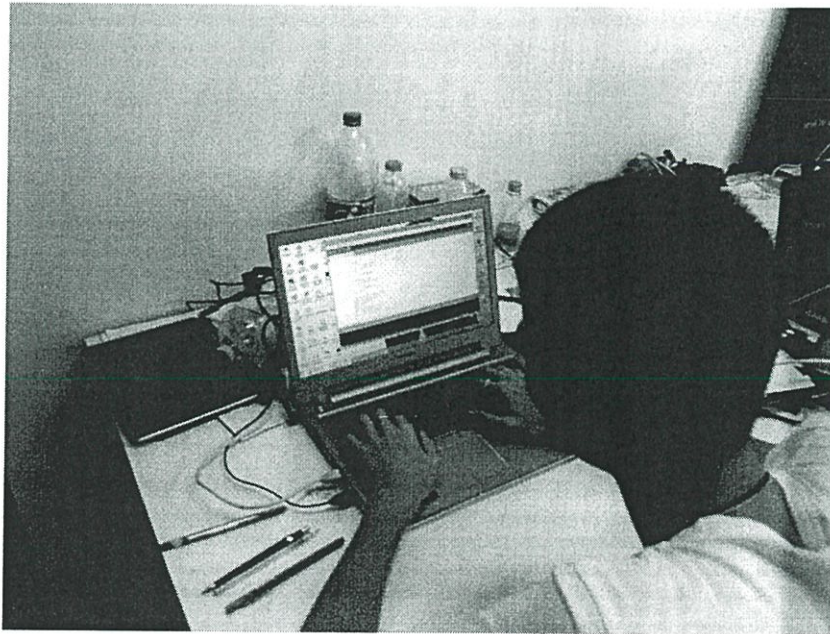
รูปที่ 3.6 ทดสอบการใช้งาน Kinect

- 3) เขียนโค้ดเพื่อติดต่อกับ Kinect โดยใช้โปรแกรม Microsoft Visual Basic



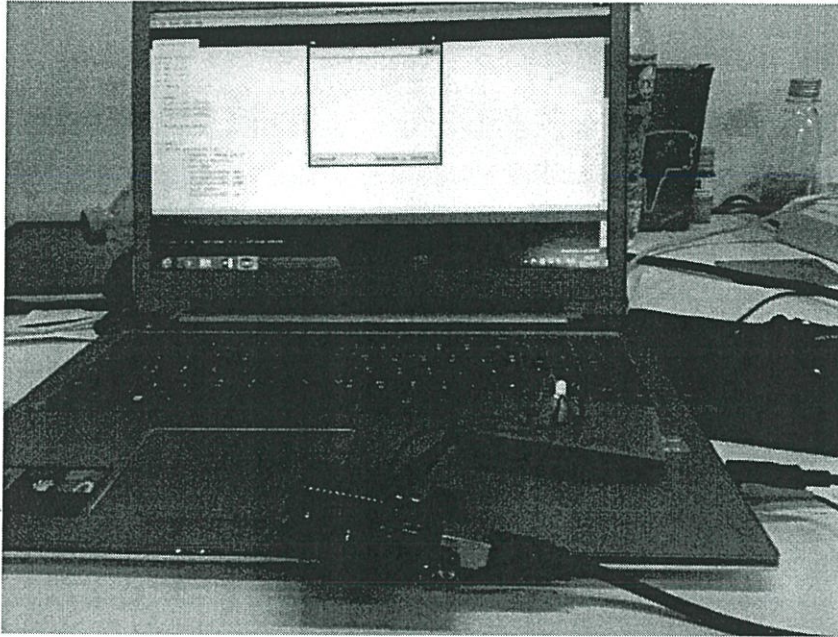
รูปที่ 3.7 เขียนโค้ด Microsoft Visual Basic เพื่อทำการติดต่อกับ Kinect

- 4) เขียนโค้ดรองรับข้อมูลจาก Kinect โดยใช้โปรแกรม Arduino



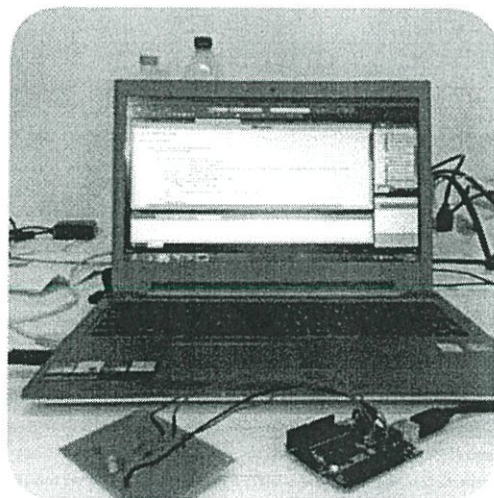
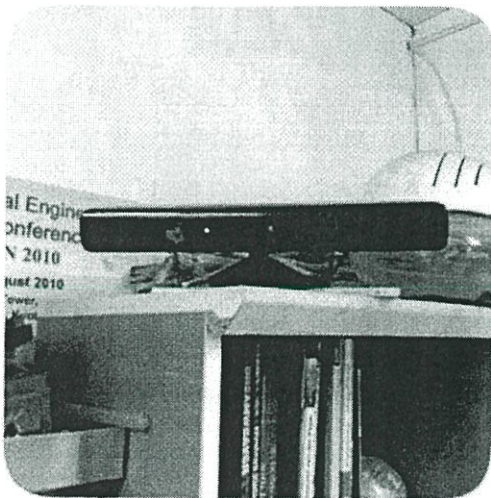
รูปที่ 3.8 เขียนโค้ดเพื่อรองรับการทำงานของตัว Kinect

5) ทำการทดสอบ Code ของ Arduino



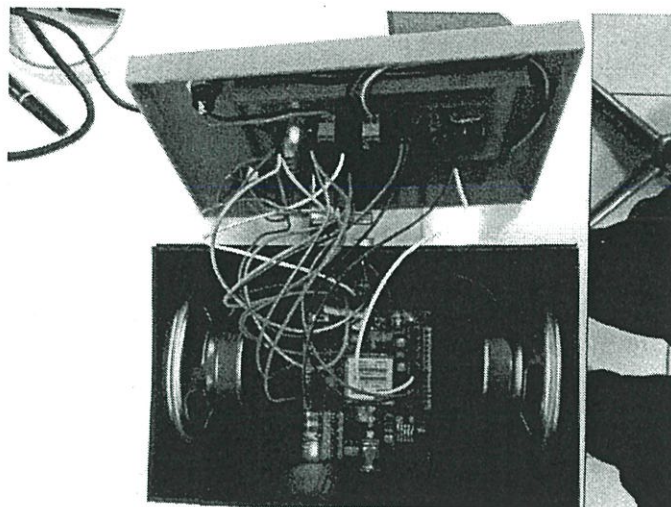
รูปที่ 3.9 ทำการทดสอบการทำงานของ Code Arduino

6) ทำการเชื่อมต่อ Arduino และ Kinect เข้ากับ Computer



รูปที่ 3.10 ทำการเชื่อมต่อระบบ

## 7) ทำการจัดอุปกรณ์ลงกล่อง



รูปที่ 3.11 ทำการจัดอุปกรณ์ลงกล่อง

## 8) ทำการทดสอบและบันทึกเก็บค่า

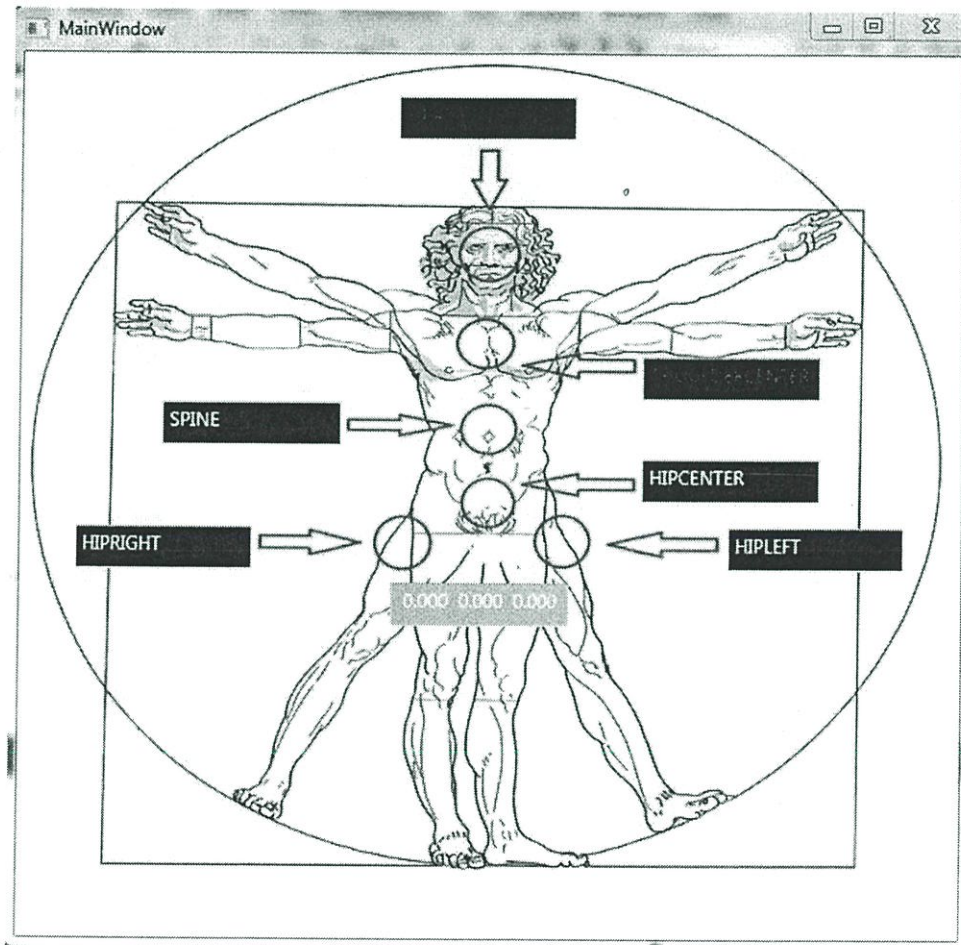
ตาราง 3.2 ตัวอย่างตารางผลการทดสอบและบันทึกผล หน่วยเป็นเมตร

	ตำแหน่ง/พิกัด	X	Y	Z	ลัม
ครั้งที่ 1	Head	0.25	-0.808	3.458	✓
	ShoulderCenter	0.277	-0.905	3.739	
ครั้งที่ 2	Head	0.209	-0.636	3.19	✓
	ShoulderCenter	0.15	-0.857	3.265	
ครั้งที่ 3	Head	0.351	-0.804	3.796	✓
	ShoulderCenter	0.273	-0.952	3.514	
ครั้งที่ 4	Head	0.494	-0.771	3.629	✓
	ShoulderCenter	0.296	-0.949	3.461	
ครั้งที่ 5	Head	0	-0.91	3.682	✓
	ShoulderCenter	-0.176	-0.971	3.453	

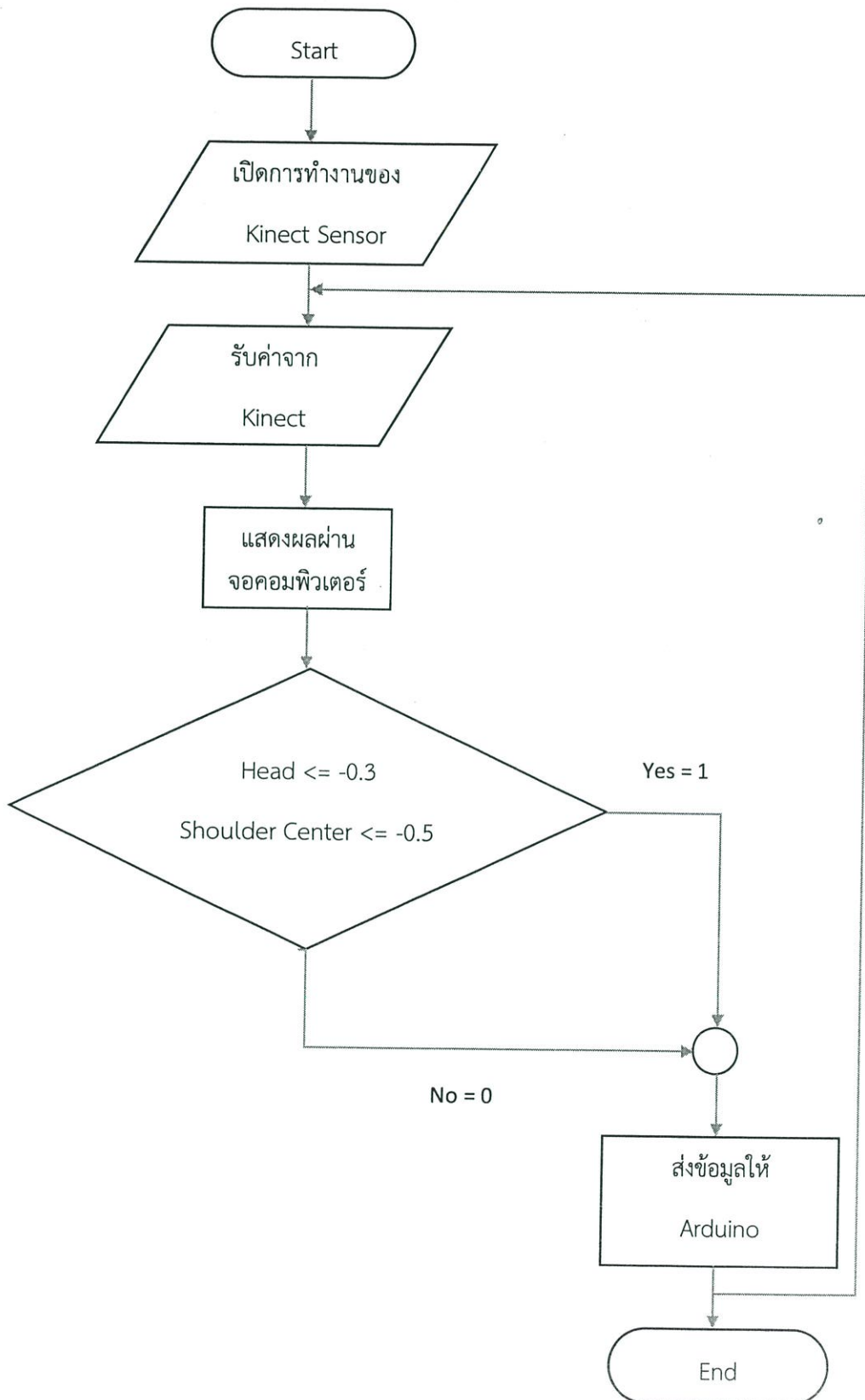
### 3.3 การพัฒนาโปรแกรมส่วน Client

โปรแกรมส่วน Client นี้จะเป็นโปรแกรมที่ทำการตรวจจับการเคลื่อนไหวและแสดงภาพโครงร่างของผู้ใช้ จากนั้นจะส่งข้อมูลการตรวจจับการเคลื่อนไหวไปยังโปรแกรม Server การพัฒนาโปรแกรมส่วน นี้จะพัฒนาเพิ่มเติมจากโปรแกรมตัวอย่างที่มีให้มา กับ SDK ของ Kinect โดยในส่วน User Interface จะเพิ่มรูปแบบฟอร์มของร่างกายมนุษย์เข้าไป เพื่อแสดงค่าต่างๆ ที่ได้รับมาจากตัว Kinect แล้วจะส่งข้อมูลการตรวจจับการเคลื่อนไหวไปให้ Arduino ตามที่แสดงในรูปที่ 3.12

ข้อมูลการตรวจจับการเคลื่อนไหวที่จะส่งไปยังโปรแกรม Server จะข้อมูลตำแหน่งของข้อต่อทั้งหมด 2 ข้อต่อ จะประกอบไปด้วยค่า Confidence และค่าพิกัด X, Y และ Z โดยนำมาเรียงต่อกันเป็นสาย String คั่นระหว่างข้อมูลแต่ละค่าด้วยการเว้นวรรค



รูปที่ 3.12 ตัวอย่างโปรแกรมส่วน Client



รูปที่ 3.13 แสดงการทำงานของโปรแกรมส่วน Client

จากรูปที่ 3.13 แสดงหลักการทำงานของโปรแกรม VB โดยจะเปิดการทำงานของกล้อง Kinect แล้วรับค่าที่ได้จากกล้อง มาแสดงค่าต่างๆบนหน้าจอ ค่าที่ได้จากกล้อง เราจะนำมาสร้างเป็นค่าเงื่อนไขต่างๆ (ค่าที่สร้างเป็นเงื่อนไขมาจากผลการทดลองตารางที่ 4.1-4.3) เพื่อส่งข้อมูลต่อให้กับ Arduino

ตัวอย่างของโค้ดในการรวมค่าตำแหน่งของข้อต่อให้เป็น String แสดงในรูปที่ 3.13

```

Private Sub sensor_SkeletonFrameReady(ByVal sender As Object, ByVal e As
Microsoft.Kinect.SkeletonFrameReadyEventArgs) Handles sensor.SkeletonFrameReady
    Dim skFrame As SkeletonFrame
    Dim skeletonSlot As Integer = 0
    Dim playerSkeleton As Skeleton
    Dim displayerSkeleton As Skeleton
    skFrame = e.OpenSkeletonFrame
    If Not skFrame Is Nothing Then
        Dim skeletonData(skFrame.SkeletonArrayLength - 1) As Skeleton
        skFrame.CopySkeletonDataTo(skeletonData)
        playerSkeleton = (From s In skeletonData Where s.TrackingState =
SkeletonTrackingState.Tracked Select s).FirstOrDefault()
        displayerSkeleton = (From s In skeletonData Where s.TrackingState =
SkeletonTrackingState.NotTracked Select s).FirstOrDefault()

        If Not playerSkeleton Is Nothing Then

            Dim strFormat = "{0:0.000} {1:0.000} {2:0.000}"

HeadKinect.Text =
String.Format(strFormat,playerSkeleton.Joints(JointType.Head).Position.X,
playerSkeleton.Joints(JointType.Head).Position.Y,
playerSkeleton.Joints(JointType.Head).Position.Z)

ShoulderCenterKinect.Text =
String.Format(strFormat,playerSkeleton.Joints(JointType.ShoulderCenter).Position.X,
playerSkeleton.Joints(JointType.ShoulderCenter).Position.Y,
playerSkeleton.Joints(JointType.ShoulderCenter).Position.Z)

```

```

SpineKinect.Text = String.Format(strFormat,
playerSkeleton.Joints(JointType.Spine).Position.X,

playerSkeleton.Joints(JointType.Spine).Position.Y,

playerSkeleton.Joints(JointType.Spine).Position.Z)

HipCenterKinect.Text =
String.Format(strFormat,playerSkeleton.Joints(JointType.HipCenter).Position.X,
                playerSkeleton.Joints(JointType.HipCenter).Position.Y,
                playerSkeleton.Joints(JointType.HipCenter).Position.Z)

HipRightKinect.Text =
String.Format(strFormat,playerSkeleton.Joints(JointType.HipRight).Position.X,
                playerSkeleton.Joints(JointType.HipRight).Position.Y,
                playerSkeleton.Joints(JointType.HipRight).Position.Z)

HipLeftKinect.Text =
String.Format(strFormat,playerSkeleton.Joints(JointType.HipLeft).Position.X,
                playerSkeleton.Joints(JointType.HipLeft).Position.Y,
                playerSkeleton.Joints(JointType.HipLeft).Position.Z)

Elseif Not displayerSkeleton Is Nothing Then

        HeadKinect.Text = "HEAD"
        ShoulderCenterKinect.Text = "SHOULDERCENTER"
        SpineKinect.Text = "SPINE"
        HipCenterKinect.Text = "HIPCENTER"
        HipRightKinect.Text = "HIPRIGHT"
        HipLeftKinect.Text = "HIPLEFT"

Else
End If

```

รูปที่ 3.14 โค้ดการรวมค่าตำแหน่งของข้อต่อให้เป็น String

หลังจากที่รวมค่าตำแหน่งของข้อต่อให้เป็นข้อมูลการตรวจจับการเคลื่อนไหวที่พร้อมจะส่งไปยังโปรแกรม Server

### 3.4 การพัฒนาโปรแกรมส่วน Server

โปรแกรมส่วน Server จะเป็นโปรแกรมที่ทำการรับข้อมูลการตรวจจับการเคลื่อนไหวที่ส่งมาจากโปรแกรม Client แสดงภาพโครงร่างมนุษย์ที่กล้องแต่ละตัวตรวจจับได้ และทำการประกอบข้อมูลการตรวจจับการเคลื่อนไหวให้เป็นโครงร่างมนุษย์ประกอบรวมตลอดเวลาที่ได้รับข้อมูล เพื่อจะแสดงสถานการณ์ถัดไป

```

if (Serial.available() > 0)
{
  computer[0] = Serial.read()-'0';

  for(int i=1;i<10000;i++){

  computer[1] = Serial.read()-'0';

  }

  Serial.print("c = ");

  Serial.println(computer[0]);

  Serial.flush();

  // kinnect value

  switch (computer[0]){

  // man down

  case (0):

  break;

  case (1):

  sms.SendSMS ("0833639693", "Help me plzzz");

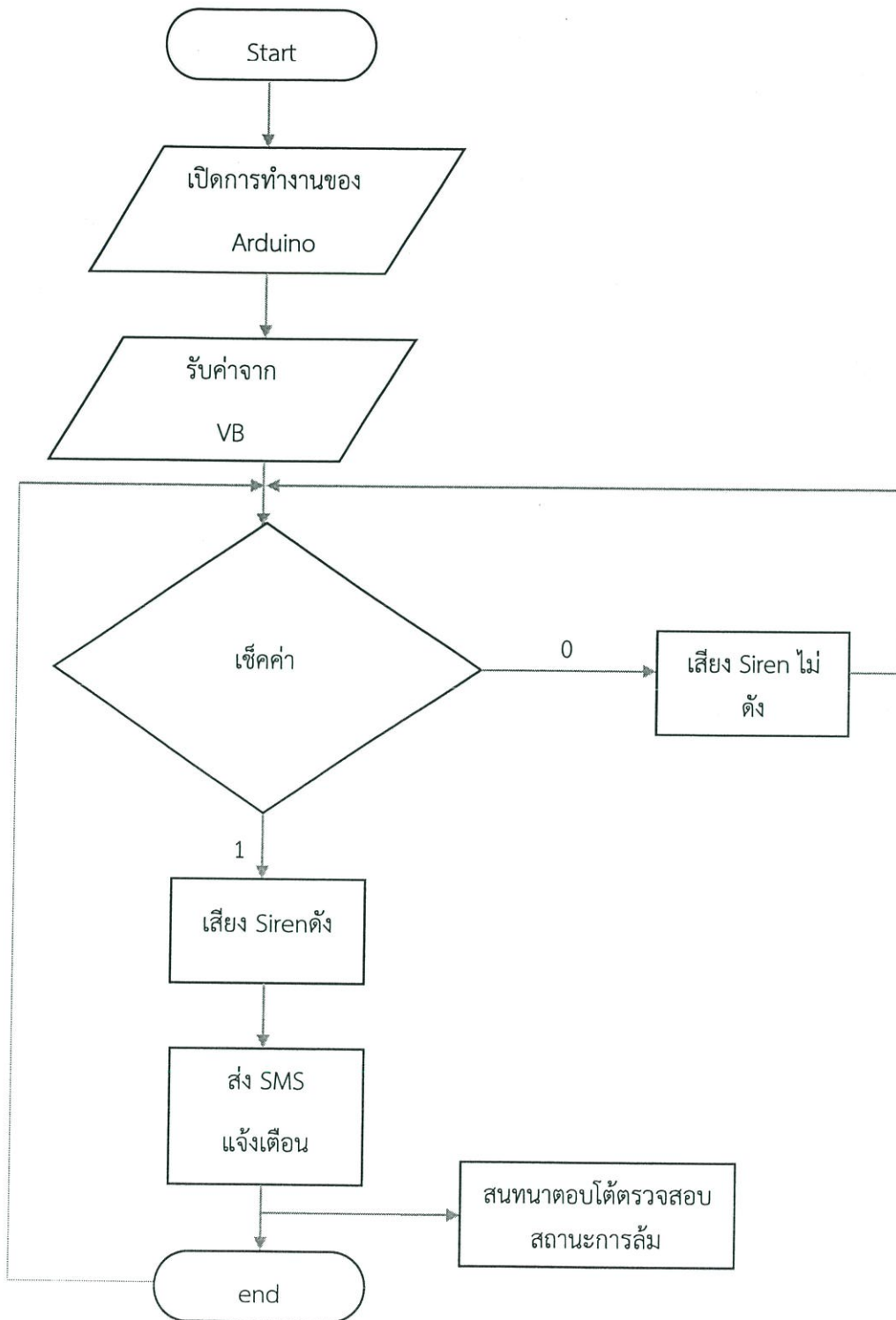
  digitalWrite(LED2, LOW);

  digitalWrite(LED3, LOW);

  digitalWrite(LED1, HIGH);

```

รูปที่ 3.15 ตัวอย่าง Code การรับและส่งข้อมูลของ Arduino



รูปที่ 3.16 แสดงหลักการทำงานของโปรแกรมส่วน Server

จากรูปที่ 3.16 แสดงหลักการทำงาน โดย Arduino จะทำการรับค่าจาก Computer (VB) จากนั้นเช็คค่าที่ได้รับจาก Computer เพื่อเช็คสถานะคนล้ม

ค่า 0 แสดงถึงไม่มีคนล้มในบริเวณนั้น Arduino จะไม่มีการแจ้งเตือนและจะวนไปตรวจสอบการรับค่าจาก Computer

ค่า 1 แสดงว่ามีคนล้มในบริเวณนั้น Arduino จะทำการแจ้งเตือนผ่าน Siren และ SMS โดยมี 2 กรณี

กรณีที่ 1 เมื่อข้อความถูกส่งไปยังปลายทางผู้รับ ผู้รับทำการโทรกลับมาเพื่อเช็คสถานะว่าบุคคลนั้นล้มจริงหรือไม่

กรณีที่ 2 เมื่อข้อความถูกส่งไปยังปลายทางผู้รับ ผู้รับไม่มีการตอบกลับหรือตอบสนอง Arduino จะทำการแจ้งเตือนผ่าน SMS ไปยังอีกบุคคลหนึ่งที่เราได้ระบุเลขหมายไว้

## บทที่ 4

### ผลการทดลอง

ตารางที่ 4.1 แสดงค่าตำแหน่ง/พิกัด ที่ Z ประมาณ 1.5 ขณะยืน หน่วยเป็นเมตร

ที่ Z ประมาณ 1.5 ขณะยืน									
	ครั้งที่ 1			ครั้งที่ 2			ครั้งที่ 3		
ตำแหน่ง/พิกัด	X	Y	Z	X	Y	Z	X	Y	Z
Head	-0.16	-0.778	3.156	-0.194	-0.745	3.138	0.08	-0.798	3.205
ShoulderCenter	-0.078	-0.937	3.033	-0.117	-0.896	2.99	0.04	-0.938	3.035

ตารางที่ 4.2 แสดงค่าตำแหน่ง/พิกัด ที่ Z ประมาณ 2.5 ขณะยืน หน่วยเป็นเมตร

ที่ Z ประมาณ 2.5 ขณะยืน									
	ครั้งที่ 1			ครั้งที่ 2			ครั้งที่ 3		
ตำแหน่ง/พิกัด	X	Y	Z	X	Y	Z	X	Y	Z
Head	-0.332	-0.853	3.258	0.82	-0.865	2.665	0.083	-0.877	3.127
ShoulderCenter	-0.141	-0.949	3.153	0.635	-0.883	2.821	-0.109	-0.905	3.271

ตารางที่ 4.3 แสดงค่าตำแหน่ง/พิกัด ที่ Z ประมาณ 3.5 ขณะยืน หน่วยเป็นเมตร

ที่ Z ประมาณ 3.5 ขณะยืน									
	ครั้งที่ 1			ครั้งที่ 2			ครั้งที่ 3		
ตำแหน่ง/พิกัด	X	Y	Z	X	Y	Z	X	Y	Z
Head	0.532	-0.44	3.788	0.403	-0.374	3.894	0.076	-0.443	3.999
ShoulderCenter	0.523	-0.623	3.679	0.35	-0.565	3.887	0.08	-0.607	4.006

## ลักษณะท่าทางการล้ม



รูปที่ 4.1 ลักษณะการล้มไปข้างหน้า



รูปที่ 4.2 ลักษณะการล้มไปข้างหลัง



รูปที่ 4.3 ลักษณะการล้มไปทางซ้ายมือของตัวเอง



รูปที่ 4.4 ลักษณะการล้มไปทางขวามือของตัวเอง

ตารางที่ 4.4 แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปข้างหน้า หน่วยเป็นเมตร

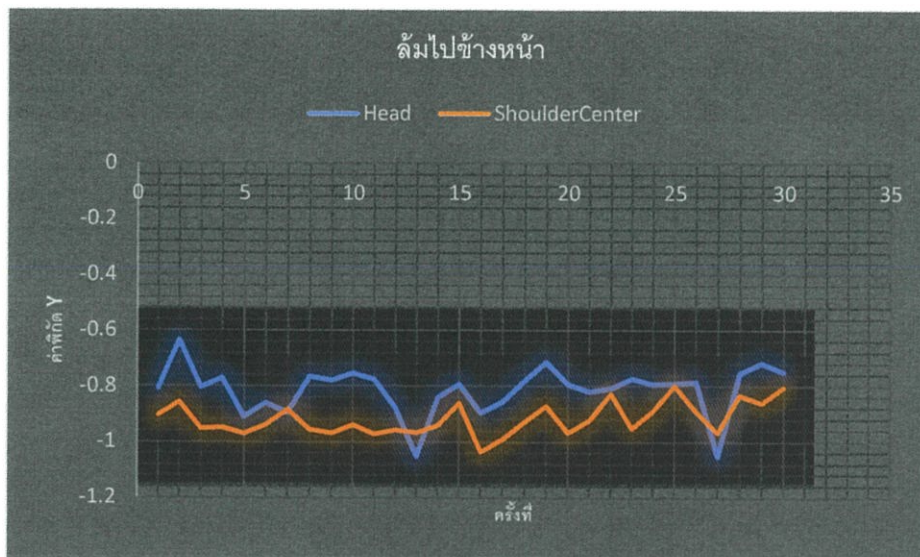
		ล้มไปข้างหน้า		
		ตำแหน่ง/พิกัด	X	Y
ครั้งที่ 1	Head	0.25	-0.808	3.458
	ShoulderCenter	0.277	-0.905	3.739
ครั้งที่ 2	Head	0.209	-0.636	3.19
	ShoulderCenter	0.15	-0.857	3.265
ครั้งที่ 3	Head	0.351	-0.804	3.796
	ShoulderCenter	0.273	-0.952	3.514
ครั้งที่ 4	Head	0.494	-0.771	3.629
	ShoulderCenter	0.296	-0.949	3.461
ครั้งที่ 5	Head	0	-0.91	3.682
	ShoulderCenter	-0.176	-0.971	3.453
ครั้งที่ 6	Head	0.395	-0.861	3.311
	ShoulderCenter	0.267	-0.939	3.426
ครั้งที่ 7	Head	0.168	-0.895	2.752
	ShoulderCenter	0.132	-0.885	2.998
ครั้งที่ 8	Head	0.428	-0.766	3.461
	ShoulderCenter	0.351	-0.956	3.335
ครั้งที่ 9	Head	0.224	-0.779	3.568
	ShoulderCenter	0.266	-0.969	3.428
ครั้งที่ 10	Head	0.389	-0.754	3.381
	ShoulderCenter	0.32	-0.939	3.473
ครั้งที่ 11	Head	0.447	-0.776	3.549
	ShoulderCenter	0.308	-0.972	3.429

ตารางที่ 4.4(ต่อ) แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปข้างหน้า หน่วยเป็นเมตร

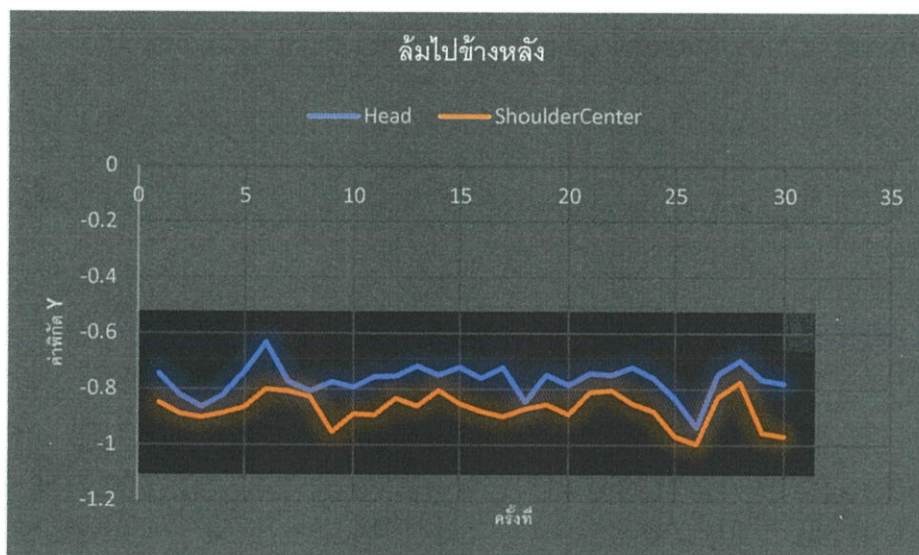
		ล้มไปข้างหน้า		
		ตำแหน่ง/พิกัด	X	Y
ครั้งที่ 12	Head	0.377	-0.878	3.677
	ShoulderCenter	0.323	-0.958	3.367
ครั้งที่ 13	Head	-0.218	-1.052	3.403
	ShoulderCenter	-0.05	-0.968	3.413
ครั้งที่ 14	Head	0.337	-0.841	3.264
	ShoulderCenter	0.321	-0.942	3.461
ครั้งที่ 15	Head	0.474	-0.794	3.135
	ShoulderCenter	0.32	-0.861	3.008
ครั้งที่ 16	Head	0.415	-0.896	2.989
	ShoulderCenter	0.447	-1.036	3.13
ครั้งที่ 17	Head	0.539	-0.856	3.142
	ShoulderCenter	0.427	-0.991	3.133
ครั้งที่ 18	Head	-0.006	-0.78	3.352
	ShoulderCenter	0.041	-0.928	3.231
ครั้งที่ 19	Head	-0.057	-0.714	3.089
	ShoulderCenter	-0.002	-0.872	2.98
ครั้งที่ 20	Head	0.508	-0.793	3.364
	ShoulderCenter	0.495	-0.968	3.366
ครั้งที่ 21	Head	0.06	-0.82	3.709
	ShoulderCenter	0.01	-0.924	3.475
ครั้งที่ 22	Head	0.116	-0.808	3.29
	ShoulderCenter	-0.093	-0.828	3.185

ตารางที่ 4.4(ต่อ) แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปข้างหน้า หน่วยเป็นเมตร

		ล้มไปข้างหน้า		
		ตำแหน่ง/พิกัด	X	Y
ครั้งที่ 23	Head	-0.012	-0.774	3.31
	ShoulderCenter	-0.121	-0.953	3.261
ครั้งที่ 24	Head	0.286	-0.793	3.489
	ShoulderCenter	0.207	-0.889	3.312
ครั้งที่ 25	Head	0.209	-0.79	3.203
	ShoulderCenter	0.181	-0.799	3.008
ครั้งที่ 26	Head	-0.107	-0.786	3.451
	ShoulderCenter	-0.067	-0.891	3.275
ครั้งที่ 27	Head	-0.218	-1.052	3.403
	ShoulderCenter	-0.05	-0.968	3.413
ครั้งที่ 28	Head	0.065	-0.754	3.265
	ShoulderCenter	-0.034	-0.833	3.086
ครั้งที่ 29	Head	0.044	-0.717	3.5
	ShoulderCenter	-0.012	-0.861	3.423
ครั้งที่ 30	Head	-0.026	-0.749	3.332
	ShoulderCenter	0.005	-0.804	3.068



รูปที่ 4.5 แสดงกราฟการล้มไปข้างหน้า



รูปที่ 4.6 แสดงกราฟการล้มไปข้างหลัง

ตารางที่ 4.5 แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปข้างหลัง หน่วยเป็นเมตร

		ล้มไปข้างหลัง		
		ตำแหน่ง/พิกัด	X	Y
ครั้งที่ 1	Head	0.109	-0.745	3.573
	ShoulderCenter	0.087	-0.848	3.357
ครั้งที่ 2	Head	0.108	-0.821	3.463
	ShoulderCenter	0.11	-0.888	3.21
ครั้งที่ 3	Head	0.095	-0.864	3.291
	ShoulderCenter	0.229	-0.901	3.063
ครั้งที่ 4	Head	0.083	-0.823	3.125
	ShoulderCenter	0.199	-0.886	2.887
ครั้งที่ 5	Head	0.094	-0.738	3.445
	ShoulderCenter	0.106	-0.865	3.295
ครั้งที่ 6	Head	-0.004	-0.632	3.154
	ShoulderCenter	-0.013	-0.8	3.098
ครั้งที่ 7	Head	0.16	-0.772	3.186
	ShoulderCenter	0.017	-0.808	2.977
ครั้งที่ 8	Head	0.116	-0.808	3.29
	ShoulderCenter	-0.093	-0.828	3.185
ครั้งที่ 9	Head	-0.012	-0.774	3.31
	ShoulderCenter	-0.111	-0.953	3.261
ครั้งที่ 10	Head	0.286	-0.793	3.489
	ShoulderCenter	0.207	-0.889	3.312
ครั้งที่ 11	Head	0.145	-0.756	3.084
	ShoulderCenter	0.13	-0.893	3.254

ตารางที่ 4.5(ต่อ) แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปข้างหลัง หน่วยเป็นเมตร

		ล้มไปข้างหน้า		
		ตำแหน่ง/พิกัด	X	Y
ครั้งที่ 12	Head	0.065	-0.754	3.265
	ShoulderCenter	-0.034	-0.833	3.086
ครั้งที่ 13	Head	0.044	-0.717	3.5
	ShoulderCenter	-0.012	-0.861	3.423
ครั้งที่ 14	Head	-0.026	-0.749	3.332
	ShoulderCenter	0.005	-0.804	3.068
ครั้งที่ 15	Head	-0.011	-0.721	3.191
	ShoulderCenter	0.073	-0.852	2.972
ครั้งที่ 16	Head	-0.041	-0.76	3.519
	ShoulderCenter	-0.02	-0.88	3.345
ครั้งที่ 17	Head	-0.142	-0.723	3.293
	ShoulderCenter	-0.102	-0.898	3.253
ครั้งที่ 18	Head	-0.215	-0.845	3.521
	ShoulderCenter	-0.095	-0.871	3.357
ครั้งที่ 19	Head	-0.15	-0.749	3.558
	ShoulderCenter	-0.08	-0.855	3.4
ครั้งที่ 20	Head	-0.107	-0.786	3.451
	ShoulderCenter	-0.067	-0.891	3.275
ครั้งที่ 21	Head	-0.069	-0.744	3.599
	ShoulderCenter	-0.05	-0.811	3.336
ครั้งที่ 22	Head	-0.026	-0.749	3.332
	ShoulderCenter	0.005	-0.804	3.068

ตารางที่ 4.5(ต่อ) แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปข้างหลัง หน่วยเป็นเมตร

		ล้มไปข้างหน้า			
		ตำแหน่ง/พิกัด	X	Y	Z
ครั้งที่ 23	Head	-0.011	-0.721	3.191	
	ShoulderCenter	0.073	-0.852	2.972	
ครั้งที่ 24	Head	-0.041	-0.76	3.519	
	ShoulderCenter	-0.02	-0.88	3.345	
ครั้งที่ 25	Head	0.496	-0.834	3.093	
	ShoulderCenter	0.359	-0.969	2.907	
ครั้งที่ 26	Head	-0.086	-0.936	2.959	
	ShoulderCenter	0.204	-0.996	2.913	
ครั้งที่ 27	Head	1.024	-0.745	2.825	
	ShoulderCenter	0.882	-0.827	2.941	
ครั้งที่ 28	Head	0.822	-0.697	2.947	
	ShoulderCenter	0.583	-0.774	2.991	
ครั้งที่ 29	Head	0.428	-0.766	3.461	
	ShoulderCenter	0.351	-0.956	3.335	
ครั้งที่ 30	Head	0.224	-0.779	3.568	
	ShoulderCenter	0.266	-0.969	3.428	

ตารางที่ 4.6 แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปทางซ้ายมือตัวเอง หน่วยเป็นเมตร

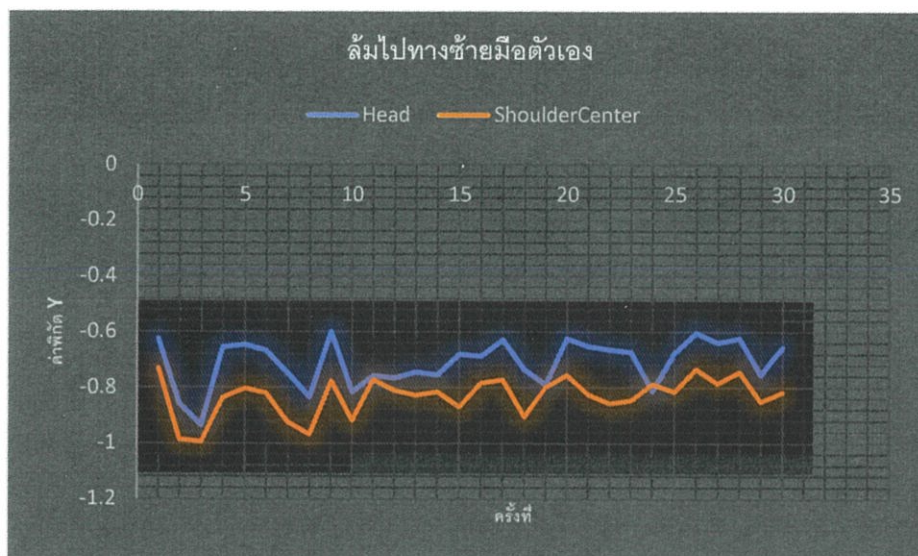
	ล้มไปทางซ้ายมือตัวเอง			
	ตำแหน่ง/พิกัด	X	Y	Z
ครั้งที่ 1	Head	-0.015	-0.626	3.059
	ShoulderCenter	0.157	-0.732	3.181
ครั้งที่ 2	Head	-0.599	-0.859	2.954
	ShoulderCenter	-0.459	-0.987	2.917
ครั้งที่ 3	Head	-0.086	-0.936	2.959
	ShoulderCenter	0.204	-0.996	2.913
ครั้งที่ 4	Head	0.444	-0.655	2.899
	ShoulderCenter	0.289	-0.837	2.883
ครั้งที่ 5	Head	0.344	-0.646	3.018
	ShoulderCenter	0.217	-0.803	2.957
ครั้งที่ 6	Head	0.34	-0.667	3.039
	ShoulderCenter	0.192	-0.82	2.963
ครั้งที่ 7	Head	0.237	-0.749	2.659
	ShoulderCenter	0.12	-0.925	2.775
ครั้งที่ 8	Head	0.496	-0.834	3.093
	ShoulderCenter	0.369	-0.969	2.97
ครั้งที่ 9	Head	0.587	-0.6	3.133
	ShoulderCenter	0.529	-0.777	3.054
ครั้งที่ 10	Head	0.562	-0.815	2.622
	ShoulderCenter	0.397	-0.918	2.724
ครั้งที่ 11	Head	-0.233	-0.758	3.258
	ShoulderCenter	-0.418	-0.773	3.386

ตารางที่ 4.6(ต่อ) แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปทางซ้ายมือตัวเอง หน่วยเป็นเมตร

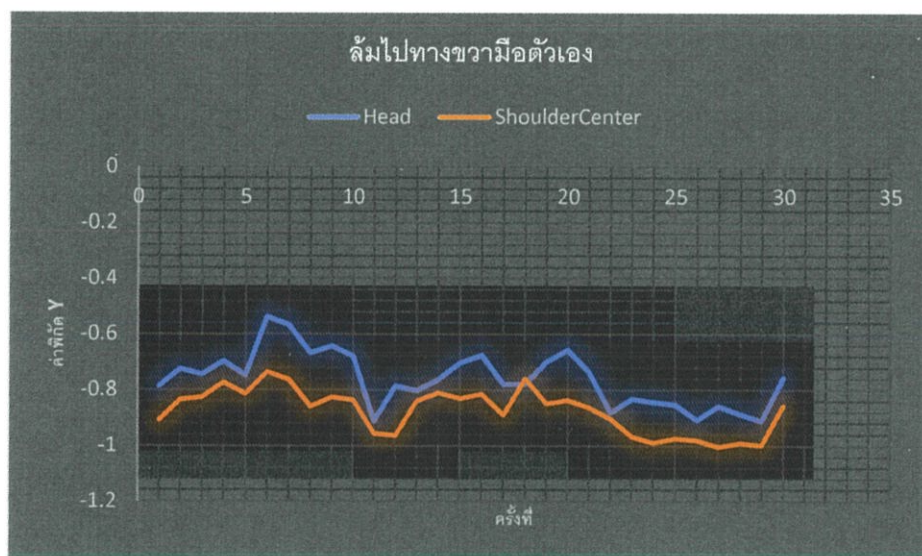
		ล้มไปข้างหน้า		
		ตำแหน่ง/พิกัด	X	Y
ครั้งที่ 12	Head	-0.333	-0.766	2.845
	ShoulderCenter	-0.24	-0.811	2.966
ครั้งที่ 13	Head	-0.728	-0.745	2.975
	ShoulderCenter	-0.585	-0.828	2.945
ครั้งที่ 14	Head	-0.268	-0.755	2.827
	ShoulderCenter	-0.417	-0.816	2.739
ครั้งที่ 15	Head	-0.571	-0.681	3.308
	ShoulderCenter	-0.499	-0.87	3.266
ครั้งที่ 16	Head	-0.394	-0.687	2.837
	ShoulderCenter	-0.282	-0.783	2.924
ครั้งที่ 17	Head	-0.236	-0.629	2.995
	ShoulderCenter	-0.346	-0.773	2.897
ครั้งที่ 18	Head	0.147	-0.736	2.736
	ShoulderCenter	0.199	-0.905	2.728
ครั้งที่ 19	Head	0.209	-0.79	3.203
	ShoulderCenter	0.181	-0.799	3.008
ครั้งที่ 20	Head	-0.444	-0.625	3.162
	ShoulderCenter	-0.346	-0.756	3.082
ครั้งที่ 21	Head	0.463	-0.652	2.972
	ShoulderCenter	0.378	-0.825	2.961
ครั้งที่ 22	Head	0.408	-0.664	2.812
	ShoulderCenter	0.317	-0.858	2.852

ตารางที่ 4.6(ต่อ) แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปทางซ้ายมือตัวเอง หน่วยเป็นเมตร

		ล้มไปข้างหน้า		
		ตำแหน่ง/พิกัด	X	Y
ครั้งที่ 23	Head	0.572	-0.673	2.832
	ShoulderCenter	0.495	-0.846	2.844
ครั้งที่ 24	Head	0.449	-0.812	3.097
	ShoulderCenter	-0.687	-0.788	3.032
ครั้งที่ 25	Head	0.213	-0.676	2.849
	ShoulderCenter	0.061	-0.816	2.95
ครั้งที่ 26	Head	0.619	-0.604	3.193
	ShoulderCenter	0.538	-0.734	3.112
ครั้งที่ 27	Head	0.804	-0.639	3.04
	ShoulderCenter	0.703	-0.786	2.994
ครั้งที่ 28	Head	0.728	-0.623	3.254
	ShoulderCenter	0.616	-0.744	3.19
ครั้งที่ 29	Head	-0.036	-0.754	2.96
	ShoulderCenter	-0.014	-0.851	3.072
ครั้งที่ 30	Head	0.302	-0.657	2.953
	ShoulderCenter	0.183	-0.817	3.022



รูปที่ 4.7 แสดงกราฟการล้มไปซ้ายมือ



รูปที่ 4.8 แสดงกราฟการล้มไปขวามือ

ตารางที่ 4.7 แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปทางขวามือตัวเอง หน่วยเป็นเมตร

		ล้มไปทางขวามือตัวเอง		
		ตำแหน่ง/พิกัด	X	Y
ครั้งที่ 1	Head	0.603	-0.787	2.94
	ShoulderCenter	0.437	-0.908	2.909
ครั้งที่ 2	Head	0.883	-0.725	2.902
	ShoulderCenter	0.707	-0.835	2.916
ครั้งที่ 3	Head	1.024	-0.745	2.825
	ShoulderCenter	0.882	-0.827	2.941
ครั้งที่ 4	Head	0.822	-0.697	2.947
	ShoulderCenter	0.583	-0.774	2.991
ครั้งที่ 5	Head	0.952	-0.748	2.806
	ShoulderCenter	0.82	-0.814	2.916
ครั้งที่ 6	Head	0.414	-0.538	3.174
	ShoulderCenter	0.405	-0.736	3.083
ครั้งที่ 7	Head	0.249	-0.567	2.996
	ShoulderCenter	0.299	-0.762	2.95
ครั้งที่ 8	Head	0.074	-0.668	2.984
	ShoulderCenter	0.088	-0.86	2.92
ครั้งที่ 9	Head	0.25	-0.643	3.031
	ShoulderCenter	0.236	-0.826	2.951
ครั้งที่ 10	Head	0.013	-0.68	2.876
	ShoulderCenter	0.001	-0.836	2.823
ครั้งที่ 11	Head	-0.054	-0.911	3.137
	ShoulderCenter	0.122	-0.957	3.076

ตารางที่ 4.7(ต่อ) แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปทางขวามือตัวเอง หน่วยเป็นเมตร

	ล้มไปข้างหน้า			
	ตำแหน่ง/พิกัด	X	Y	Z
ครั้งที่ 12	Head	0.209	-0.787	3.082
	ShoulderCenter	0.24	-0.964	2.977
ครั้งที่ 13	Head	0.647	-0.803	2.998
	ShoulderCenter	0.384	-0.84	2.937
ครั้งที่ 14	Head	0.262	-0.764	3.154
	ShoulderCenter	0.443	-0.811	3.036
ครั้งที่ 15	Head	0.471	-0.704	3.077
	ShoulderCenter	0.397	-0.831	2.894
ครั้งที่ 16	Head	-0.03	-0.676	2.876
	ShoulderCenter	0.105	-0.815	2.817
ครั้งที่ 17	Head	0.032	-0.78	2.726
	ShoulderCenter	0.08	-0.891	2.619
ครั้งที่ 18	Head	0.07	-0.779	2.878
	ShoulderCenter	0.221	-0.76	2.796
ครั้งที่ 19	Head	0.236	-0.7	3.015
	ShoulderCenter	0.205	-0.85	2.883
ครั้งที่ 20	Head	0.229	-0.658	2.865
	ShoulderCenter	0.215	-0.838	2.793
ครั้งที่ 21	Head	0.089	-0.737	2.873
	ShoulderCenter	0.175	-0.864	2.736
ครั้งที่ 22	Head	-0.066	-0.882	3.051
	ShoulderCenter	0.095	-0.904	2.946

ตารางที่ 4.7(ต่อ) แสดงค่าตำแหน่ง/พิกัด ขณะล้มไปทางขวามือตัวเอง หน่วยเป็นเมตร

		ล้มไปข้างหน้า		
		ตำแหน่ง/พิกัด	X	Y
ครั้งที่ 23	Head	0.676	-0.833	2.79
	ShoulderCenter	0.591	-0.968	2.685
ครั้งที่ 24	Head	0.518	-0.844	2.874
	ShoulderCenter	0.568	-0.99	2.717
ครั้งที่ 25	Head	0.588	-0.853	2.719
	ShoulderCenter	0.476	-0.975	2.616
ครั้งที่ 26	Head	0.691	-0.908	2.696
	ShoulderCenter	0.508	-0.981	2.657
ครั้งที่ 27	Head	0.693	-0.858	3.072
	ShoulderCenter	0.749	-1.004	3.134
ครั้งที่ 28	Head	0.712	-0.887	2.811
	ShoulderCenter	0.558	-0.991	2.657
ครั้งที่ 29	Head	0.682	-0.911	2.747
	ShoulderCenter	0.519	-0.999	2.675
ครั้งที่ 30	Head	0.534	-0.758	2.659
	ShoulderCenter	0.415	-0.857	2.734

## บทที่ 5

### วิจารณ์สรุปผล

#### วิจารณ์และสรุปผล

จากการทดลองจะเห็นได้ว่าค่าที่เราได้จากการทดลองนั้น จะมีอยู่ 3 แกน โดยการตรวจจับของส่วนต่างๆ ของร่างกาย อาทิเช่น หัว หน้าอก หัวไหล่ เป็นต้น นั่นคือ  $x, y, z$  โดยแกน  $x$  คือความกว้าง  $y$  คือความสูงต่ำ และ  $z$  คือระยะห่างจากตัว Kinect ซึ่งแกนที่เราใช้ในการวัดคนล้มคือแกน  $y$  เมื่อเราเก็บค่าแกนจากการล้มในท่าต่างๆ ก็พบว่า แกน  $y$  จะมีการเปลี่ยนแปลงที่แน่นอนจากคนยืนแล้วล้มลง ซึ่งจากการทดสอบล้ม เราจะได้ว่าเมื่อแกน  $y$  มีค่า น้อยกว่า  $-0.3$  ที่ตำแหน่งหัว และมีค่าน้อยกว่า  $-0.5$  ที่ตำแหน่งหน้าอก จะแสดงให้เห็นถึงลักษณะของคนล้ม และเครื่องจะทำการแจ้งเตือนผ่าน SMS และ Siren ตามลำดับ จากนั้นก็จะมีการโทรศัพท์ไปยังหมายเลขของเครื่องตรวจจับและทำการสนทนาในที่สุด

อุปสรรคในการทำงานช่วงแรกๆ จะเกิดจากการศึกษาการทำงานของ Code ว่าทำงานอย่างไรซึ่งใช้เวลานานในการศึกษา การหาค่าจุดต่างๆ การศึกษาการเขียนโปรแกรม จนกระทั่งความเข้าใจในการเขียนโปรแกรม แต่ยังมีบางส่วนที่ไม่สามารถแก้ไขได้อย่างสมบูรณ์

ปัญหาที่เกิดขึ้นในการทำชิ้นงานนี้

1. เมื่อมีการติดต่อและเกิดการสนทนากันแล้วที่เครื่องตรวจจับจะได้ยินเสียงพูดเบาเกินไป จนบางครั้งถ้าคนที่ล้มมีอาการหุนหวก จะทำให้ไม่ได้ยินเสียงเลย จึงทำการแก้ปัญหา โดยการ ต่อลำโพงภายนอกเข้ามาเสริม
2. คนนอนกับคนล้ม จะมีท่าทางที่คล้ายกันมาก จึงไม่สามารถแยกแยะได้ระหว่างนอนกับล้ม

แนะนำในการแก้ปัญหา ของคนล้มกับคนนอน โดยใช้ความเร็วของศีรษะในแนวแกน  $Y$  มาเป็นเงื่อนไขในการเขียนโปรแกรม ซึ่งคนล้มจะมีความเร็วการล้มแตกต่างจากคนนอน

## เอกสารอ้างอิง

- [1] Microsoft, Kinect for Xbox 360 - Xbox.com [Online], Available : <http://www.xbox.com/en-GB/kinect> [2012, April 1].
- [2] นราวุฒิ พัฒโนทัย, พรชัย มงคลนาม และบัณฑิต วรรณภา, 2554, “การค้นหาคำแทน่งและทิศทางของกล้อง Kinect อัตโนมติ”, The 2011 International Computer Science and Engineering Conference, September 7-9, Bangkok, Thailand, pp. 251-255.
- [3] รายงานผลการดำเนินงานสร้างเสริมสุขภาพผู้สูงอายุ, ทาปลาตุก, คณะพยาบาลศาสตร์, มหาวิทยาลัยเชียงใหม่

ภาคผนวก ก.  
Code Visual Basic

```
Imports System.IO
```

```
Imports System.IO.Ports
```

```
Imports System.Threading
```

```
Imports System.Text
```

```
Imports Microsoft.Kinect
```

```
Imports Coding4Fun.Kinect.Wpf
```

```
Namespace SkeletalTracking
```

```
    Partial Public Class MainWindow
```

```
        Inherits Window
```

```
        Shared _continue As Boolean
```

```
        Shared _serialPort As SerialPort
```

```
        Private WithEvents sensor As KinectSensor
```

```
        Public Sub New()
```

```
            InitializeComponent()
```

```
        End Sub
```

```
        Private Property control As Object
```

```
        Private Property components As ComponentModel.Container
```

```
        Private Property Timer As Object
```

```
Private Sub Window_Loaded(ByVal sender As Object, ByVal e As
RoutedEventArgs)

    DiscoverKinectSensor()

End Sub

Private Sub sensor_SkeletonFrameReady(ByVal sender As Object, ByVal e As
Microsoft.Kinect.SkeletonFrameReadyEventArgs) Handles sensor.SkeletonFrameReady
    Dim skFrame As SkeletonFrame
    Dim skeletonSlot As Integer = 0
    Dim playerSkeleton As Skeleton
    Dim displayerSkeleton As Skeleton

    skFrame = e.OpenSkeletonFrame

    If Not skFrame Is Nothing Then
        Dim skeletonData(skFrame.SkeletonArrayLength - 1) As Skeleton
        skFrame.CopySkeletonDataTo(skeletonData)
        playerSkeleton = (From s In skeletonData Where s.TrackingState =
SkeletonTrackingState.Tracked Select s).FirstOrDefault()
        displayerSkeleton = (From s In skeletonData Where s.TrackingState =
SkeletonTrackingState.NotTracked Select s).FirstOrDefault()

        If Not playerSkeleton Is Nothing Then

            Dim strFormat = "{0:0.000} {1:0.000} {2:0.000}"

            HeadKinect.Text = String.Format(strFormat,
playerSkeleton.Joints(JointType.Head).Position.X,
                                playerSkeleton.Joints(JointType.Head).Position.Y,
```

```
playerSkeleton.Joints(JointType.Head).Position.Z)
```

```
ShoulderCenterKinect.Text = String.Format(strFormat,  
playerSkeleton.Joints(JointType.ShoulderCenter).Position.X,
```

```
playerSkeleton.Joints(JointType.ShoulderCenter).Position.Y,  
playerSkeleton.Joints(JointType.ShoulderCenter).Position.Z)
```

```
SpineKinect.Text = String.Format(strFormat,  
playerSkeleton.Joints(JointType.Spine).Position.X,
```

```
playerSkeleton.Joints(JointType.Spine).Position.Y,
```

```
playerSkeleton.Joints(JointType.Spine).Position.Z)
```

```
HipCenterKinect.Text =
```

```
String.Format(strFormat,playerSkeleton.Joints(JointType.HipCenter).Position.X,  
playerSkeleton.Joints(JointType.HipCenter).Position.Y,  
playerSkeleton.Joints(JointType.HipCenter).Position.Z)
```

```
HipRightKinect.Text = String.Format(strFormat,  
playerSkeleton.Joints(JointType.HipRight).Position.X,
```

```
playerSkeleton.Joints(JointType.HipRight).Position.Y,
```

```
playerSkeleton.Joints(JointType.HipRight).Position.Z)
```

```
HipLeftKinect.Text = String.Format(strFormat,  
playerSkeleton.Joints(JointType.HipLeft).Position.X,
```

```
playerSkeleton.Joints(JointType.HipLeft).Position.Y,
```

```
playerSkeleton.Joints(JointType.HipLeft).Position.Z)

    Me.components = New System.ComponentModel.Container()
    Me.control = New Windows.Controls.Control()
    Me.Timer = New Timers.Timer()
    MainWindow._serialPort = New
System.IO.Ports.SerialPort(Me.components())
    MainWindow._serialPort.PortName = "COM3"
    MainWindow._serialPort.BaudRate = 4800
    MainWindow._serialPort.StopBits = StopBits.One
    Dim Head = CStr(playerSkeleton.Joints(JointType.Head).Position.Y)
    Dim ShoulderCenter =
CStr(playerSkeleton.Joints(JointType.ShoulderCenter).Position.Y)

    If Head <= -0.3 And ShoulderCenter <= -0.5 Then

        _serialPort.Open()
        _serialPort.Write("0")
        _serialPort.Write("1")
        _serialPort.Close()

    Else

        _serialPort.Open()
        _serialPort.Write("0")
        _serialPort.Close()

    End If

    ElseIf Not displayerSkeleton Is Nothing Then

        HeadKinect.Text = "HEAD"
```

```
        ShoulderCenterKinect.Text = "SHOULDERCENTER"
        SpineKinect.Text = "SPINE"
        HipCenterKinect.Text = "HIPCENTER"
        HipRightKinect.Text = "HIPRIGHT"
        HipLeftKinect.Text = "HIPLEFT"

    Else
    End If

    Else
    End If

End Sub

Private Sub Window_Closed(ByVal sender As Object, ByVal e As EventArgs)

    sensor.Stop()
    sensor.Dispose()

End Sub

Private Sub DiscoverKinectSensor()

    For Each iterSensor As KinectSensor In KinectSensor.KinectSensors
        If iterSensor.Status = KinectStatus.Connected Then
            sensor = iterSensor
            Exit For
        End If
    Next

    If sensor Is Nothing Then
        MsgBox("Could not find any valid Kinect Connected. Please restart the
program", vbCritical)
    End Sub
```

```
End If

Select Case sensor.Status
    Case KinectStatus.Connected
        ' kinect connected

    Case KinectStatus.Disconnected
        MsgBox("Kinect is Disconnected", vbExclamation)
    Case KinectStatus.NotPowered
        MsgBox("Kinect is Not Powered. Connect to Power", vbExclamation)

    Case Else
        MsgBox("Unkown Kinect Status", vbExclamation)
End Select

If (sensor.Status = KinectStatus.Connected) Then
    InitializeKinect()
End If
End Sub

Private Sub InitializeKinect()

    Dim parameters = New TransformSmoothParameters With {.Smoothing =
0.75F, .Correction = 0.0F, .Prediction = 0.0F, .JitterRadius = 0.05F, .MaxDeviationRadius
= 0.04F}

    sensor.SkeletonStream.Enable(parameters)

    sensor.Start()

End Sub
End Class
End Namespace
```

ภาคผนวก ข.  
Arduino Code

<pre> #include "SIM900.h" #include &lt;SoftwareSerial.h&gt; //We don't need the http functions. So we can disable the next line. //#include "inetGSM.h" #include "sms.h" #include "call.h"  SoftwareSerial SIM900 (7, 8);  //To change pins for Software Serial, use the two lines in GSM.cpp.  //GSM Shield for Arduino //www.open-electronics.org //this code is based on the example of Arduino Labs.  //Simple sketch to check if an incoming call is from an authorized //number and in this case, send to this number an SMS with the value //of a digital input.  //We have to create the classes for SMSs and calls. CallGSM call; SMSGSM sms;  int y = 0; int st = 0; int stbut = 0; int LED1 = 10; int LED2 = 11; int LED3 = 12; int computer[4]; digitalWrite(LED3, HIGH); digitalWrite(LED1, LOW); </pre>	<pre> int computer1 = 0; int chk = 0;  int pang1; int pang2 = 5; int pang3 = 0;  char number[20]; byte stat=0; int value=0; int pin=4; char value_str[5];  void setup() {     pinMode(LED1,OUTPUT);     pinMode(LED2,OUTPUT);     pinMode(LED3,OUTPUT);     pinMode(pin,INPUT);     //Serial connection.     Serial.begin(4800);     Serial.println("GSM Shield testing.");     //Start configuration of shield with baudrate.     //For http uses is raccomanded to use 4800 or slower.     if (gsm.begin(4800))         Serial.println("\nstatus=READY");     else Serial.println("\nstatus=IDLE"); };  void loop() {     digitalWrite(LED2, LOW);         digitalWrite(LED1, LOW);         st = 0; </pre>
---	--

```

stbut = digitalRead(pin);
Serial.println(stbut);
// check kinnect

if (Serial.available() > 0)
{
  computer[0] = Serial.read()-'0';
  for(int i=1;i<10000;i++){
  computer[1] = Serial.read()-'0';
  }
  Serial.print("c = ");
  Serial.println(computer[0]);
  Serial.flush();
  // kinnect value
  switch (computer[0]){
  // man down
  case (0):
    break;
  case (1):
    sms.SendSMS("0833639693", "Help
me plzzz");
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, LOW);
    digitalWrite(LED1, HIGH);

    for(int i=0;i<40;i++)
    {

      stbut = digitalRead(pin);
      //Chekcs status of call
      stat=call.CallStatus();
      if (stbut == HIGH)
      {
        digitalWrite(LED2, LOW);
        digitalWrite(LED3, HIGH);
        digitalWrite(LED1, LOW);
        st = 0;

```

```

        break;
      }

      else if (stat==1)
      {
        call.PickUp();
        st = 1;
        break;
      }

      st = 2;
      delay(100);
    }

    if(st == 0)
    {
      Serial.println("not ting");
      sms.SendSMS("0833639693",
"every think ok");
    }

    else if(st == 1)
    {
      Serial.println("call in comming");
    }

    else if(st == 2)
    {
      Serial.println("no call send sms
to hotpeeton");
      sms.SendSMS("0833639693",
"hotpeetall");
    }
    break;
  }
}

```

```
        break;
    }

    else if (stat==1)
    {
        call.PickUp();
        st = 1;
        break;
    }

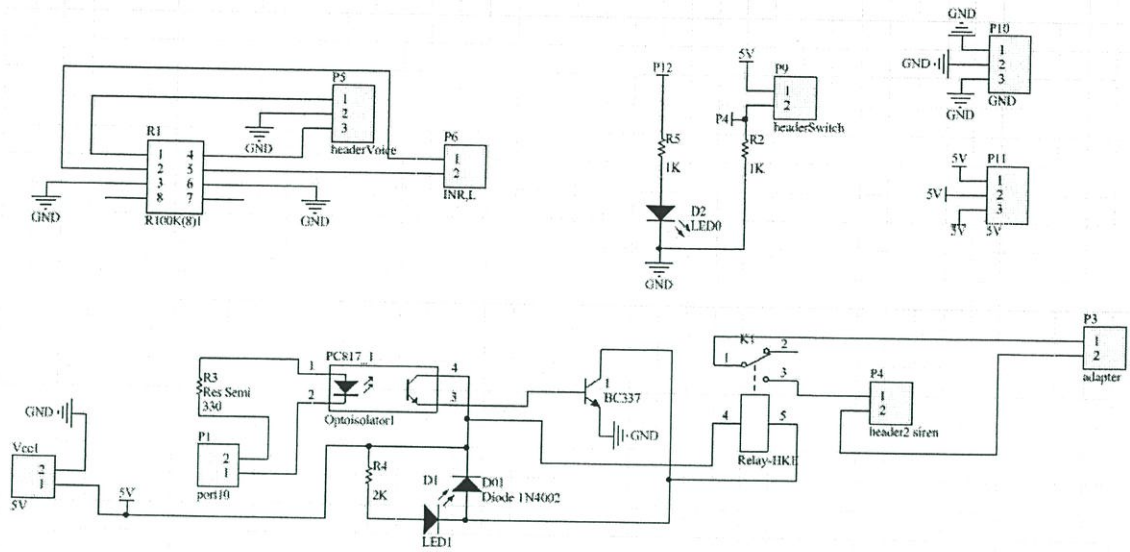
    st = 2;
    delay(100);
}

if(st == 0)
{
    Serial.println("not ting");
    sms.SendSMS("0833639693",
"every think ok");
}

else if(st == 1)
{
    Serial.println("call in comming");
}

else if(st == 2)
{
    Serial.println("no call send sms
to hotpeeton");
    sms.SendSMS("0833639693",
"hotpeetall");
}
break;
};
```

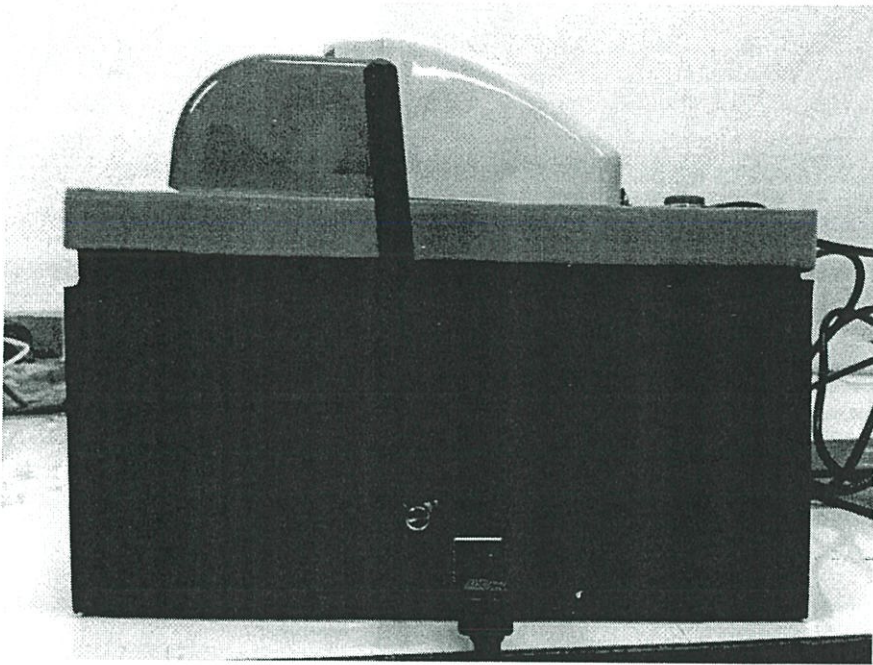
ภาคผนวก ค.  
Schematic



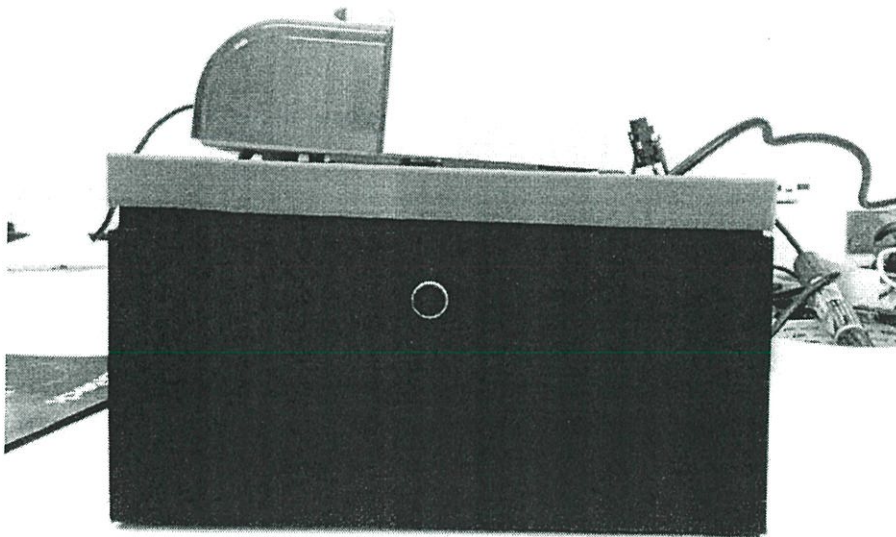
รูปที่ ค.1 แสดง schematic ของวงจรแจ้งเตือน

จากรูปที่ ค.1 แสดงschematic ของวงจร โดยมีDelay จะใช้ในการควบคุมไฟ 12V เข้ามาเพื่อมาเลี้ยงให้กับไซเรน พอตส์(P4)เป็นพอตสวิตเอาไวรั้เซรระบบใหม่ และมีRปรับค่าได้เอาไว้เพิ่มลดเสียงเวลาคุยสนทนากัน

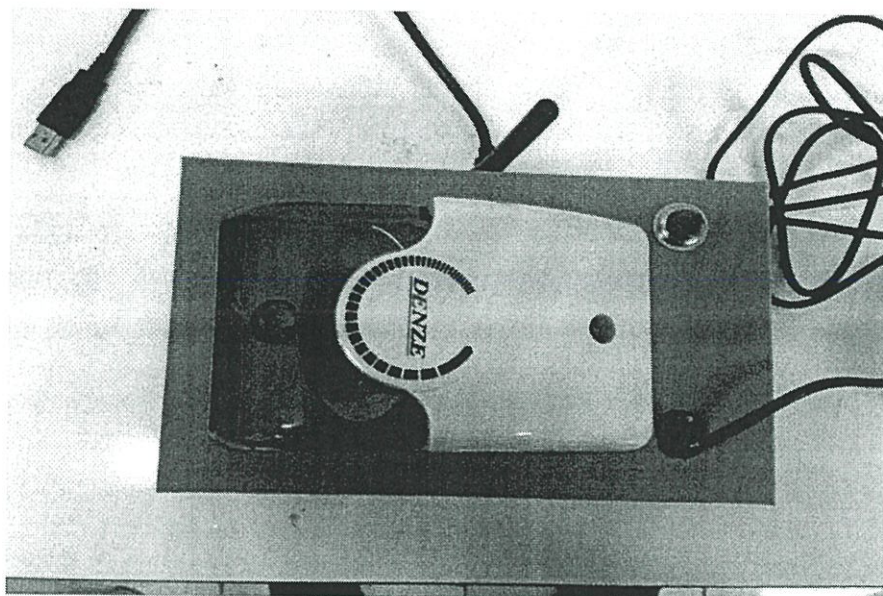
ภาคผนวก ง.  
ภาพการทำงาน ( เก็บตก )



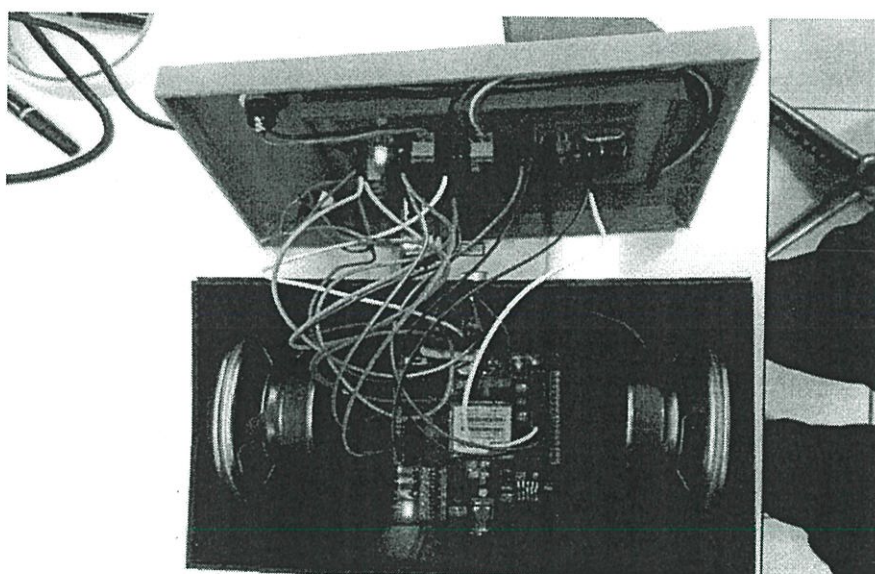
รูปที่ ง.1 แสดงด้านหลังของกล่องชิ้นงาน



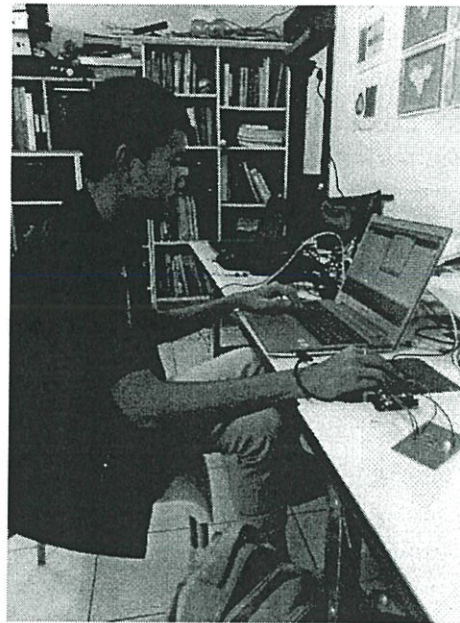
รูปที่ ง.2 แสดงด้านหน้าของกล่องชิ้นงาน



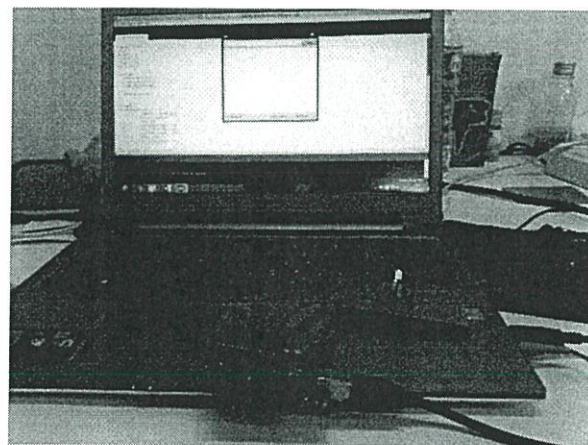
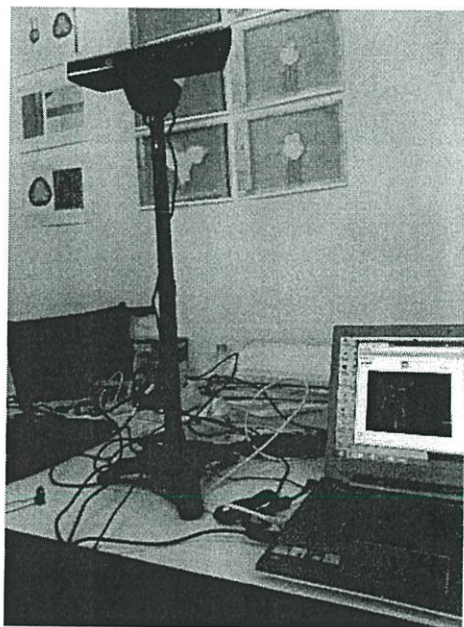
รูป ง.3 แสดงด้านบนของกล่องชิ้นงาน



รูปที่ ง.4 แสดงด้านในของกล่องชิ้นงาน



รูปที่ ๖.5 แสดงการทำโปรเจค



รูปที่ ๖.6 แสดงการทดสอบโปรเจค

## ประวัติผู้วิจัย



ชื่อ - นามสกุล  
วัน เดือน ปีเกิด  
ที่อยู่

ประวัติการศึกษา

นายโรตม ดุสิตกุล  
28 มิถุนายน 2535  
20/155 ม.5 ตำบลปากนคร อำเภอเมือง  
จังหวัดนครศรีธรรมราช 80000  
พ.ศ.2553 ประกาศนียบัตรวิชาชีพ  
สาขาวิชาอิเล็กทรอนิกส์  
วิทยาลัยเทคนิคนครศรีธรรมราช  
โทรศัพท์ 086-9029190  
Email : dusitkul@hotmail.com



ชื่อ - นามสกุล  
วัน เดือน ปีเกิด  
ที่อยู่

ประวัติการศึกษา

นายสตั้มภ์ จันทร์บาง  
11 สิงหาคม 2536  
1 หมู่ 8 ตำบลหนองเล็ง อำเภอเมืองบึงกาฬ  
จังหวัดบึงกาฬ 38000  
พ.ศ.2553 มัธยมศึกษาตอนปลาย  
สาขาวิชาวิทย์-คณิต  
โรงเรียนบึงกาฬ  
โทรศัพท์ 083-3639693  
Email : satam.chanbang@gmail.com



ชื่อ - นามสกุล  
วัน เดือน ปีเกิด  
ที่อยู่

ประวัติการศึกษา

นายสรศักดิ์ คงอินทร์  
10 กรกฎาคม 2535  
115/10 หมู่ 9 ตำบลปะเหลียน อำเภอปะเหลียน  
จังหวัดตรัง 92180  
พ.ศ.2553 มัธยมศึกษาตอนปลาย  
สาขาวิชาวิทย์-คณิต  
โรงเรียนบูรณราษฎร์  
โทรศัพท์ 097-0297100  
Email : new\_s-k@hotmail.com