การวัดการเคลื่อนไหวของร่างกายมนุษย์โดยไม่ใช้การมองเห็น
Non-optical measurement for human motion detection

ธนาธิป อินทรชูศรี
Thanathip Intarashusre
วัชระ พุ่มดอกไม้
Watchara Phumdokmai

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาชีวการแพทย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การวัดการเคลื่อนไหวของร่างกายมนุษย์โดยไม่ใช้การมองเห็น

Non-optical measurement for human motion detection

T144474

โดย

นายธนาธิป อินทรชูศรี 55010535

นายวัชระ พุ่มดอกไม้ 55011123

อาจารย์ที่ปรึกษา

ดร.สุรเดช ตรีไตรลักษณะ

วพ
ธ2431
2558
เลขหมู่.....
เลขทะเบียน.....144474
วัน,เดือน,ปี.....24 พ.ย. 2559

b.....12818926
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาชีวการแพทย์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558

ปริญญานิพนธ์ปีการศึกษา 2558

สาขาวิชา วิศวกรรมชีวการแพทย์

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การวัดการเคลื่อนไหวของร่างกายมนุษย์โดยไม่ใช้การมองเห็น

Non-optical measurement for human motion detection

ผู้จัดทำ นายธนาธิป อินทรชูศรี รหัสนักศึกษา 55010535

นายวัชระ พุ่มดอกไม้ รหัสนักศึกษา 55011123

ปริญญานิพนธ์ฉบับนี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

อาจารย์ที่ปรึกษา

หัวข้อวิทยานิพนธ์	การวัดการเคลื่อนไหวของร่างกายมนุษย์โดยไม่ใช้การมองเห็น
นักศึกษา	นายธนาธิป อินทรชูศรี 55010535
	นายวัชระ พุ่มดอกไม้ 55011123
ปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมชีวการแพทย์
พ.ศ.	2558
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ดร. สุรเดช ตรีไตรลักษณะ

บทคัดย่อ

ในการให้การรักษาทางกายภาพบำบัดได้อย่างมีประสิทธิภาพนั้น ผู้ให้การรักษาควรมีความสามารถในการตรวจประเมินที่ดี งานวิจัยนี้มีจุดประสงค์เพื่อพัฒนาอุปกรณ์ที่สามารถตรวจวัดการเคลื่อนไหวที่ส่วนต่าง ๆ ของร่างกายได้อย่างมีประสิทธิภาพ โดยใช้ข้อมูลความเร่งจาก IMU เซ็นเซอร์มาคำนวณและเปรียบเทียบผลที่ได้กับเครื่อง Electromagnetic Motion Tracking System ซึ่งมีใช้จริงภายในห้องทดลองกายภาพบำบัด คณะกายภาพบำบัด มหาวิทยาลัยมหิดล ผลการวิจัยพบว่าลักษณะของกราฟมีความคล้ายคลึงกัน

Project title	Non-optical measurement for human motion detection	
Students	Thanathip Intarashusre	55010535
	Watchara Phumdokmai	55011123
Degree	Bachelor of Engineering	
Program	Biomedical Engineering	
Year	2558	
Thesis Advisor	Dr.Suradej Tretriluxana	

Abstract

A measurement of human motion is necessary to give a good treatment for patient in physical therapy. In this paper, we develop a device for detect movement in any part of human body by Integrated acceleration data of IMU sensor. Using an acceleration data to integrate velocity data, so we can analyze a range of motion in distance. This experiment compares the result with Electromagnetic Tracking System in Physiotherapy lab at Mahidol University. The result was a bit difference because there are two difference methods to measure a human motion -- acceleration and magnetic field. So we integrated the acceleration data to velocity and the result was shown in graph.

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จได้ลุล่วงด้วยความกรุณาจากดร. สุรเดช ตรีไตรลักษณะ อาจารย์ที่ปรึกษาปริญญานิพนธ์ที่ได้ให้คำปรึกษา ข้อเสนอแนะ และ การอนุเคราะห์ด้านอุปกรณ์ ตลอดจนการแก้ไขข้อบกพร่องต่างๆ จนปริญญานิพนธ์เล่มนี้เสร็จสมบูรณ์

ขอขอบคุณอาจารย์และเจ้าหน้าที่จากคณะกายภาพบำบัด มหาวิทยาลัยมหิดลสำหรับสถานที่และการใช้งานอุปกรณ์เพื่อทำการเก็บข้อมูล เพื่อนำมาใช้ในปริญญานิพนธ์เล่มนี้

ขอขอบคุณแหล่งข้อมูลต่างๆ ที่เป็นประโยชน์และเปิดให้ผู้วิจัยสามารถเข้าถึงได้

นายธนาธิป อินทรชูศรี
นายวัชระ พุ่มดอกไม้

สารบัญ

	หน้า
บทคัดย่อ	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VI
สารบัญตาราง	IX
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์การวิจัย	1
1.3 สมมติฐานการวิจัย	2
1.4 ขอบเขตของงานวิจัย	2
1.5 ขั้นตอนการศึกษาวิจัย	2
1.6 โครงสร้างวิทยานิพนธ์	3
บทที่ 2 ทฤษฎีและความรู้พื้นฐานที่เกี่ยวข้อง	4
2.1 งานวิจัยที่เกี่ยวข้อง	4
2.2 เวชศาสตร์ฟื้นฟู	4
2.3 ออร์โธปิดิคส์	5
2.4 เวชศาสตร์ฟื้นฟูทางออร์โธปีดิคส์	5
2.5 การวัดพิสัยการเคลื่อนไหว	6
2.6 การออกกำลังกายเพื่อการบำบัดรักษา	8
2.7 การออกกำลังกายเพื่อป้องกันข้อติดหรือเพิ่มพิสัยการเคลื่อนไหวของข้อ	8
2.8 ค่าปกติของพิสัยการเคลื่อนไหว	9
2.9 การเขียนโปรแกรมโดย Arduino	11
2.10 การเขียนโปรแกรมโดย Processing	12
2.11 การเขียนโปรแกรมโดย Microsoft Visual Studio	13
2.12 การใช้โปรแกรม Microsoft Excel	18
2.13 การใช้โปรแกรม Matlab	22
2.14 หน่วยวัดแรงเฉื่อย	29
2.15 การเคลื่อนที่แนวตรง	41
2.16 แคลคูลัส	49

สารบัญ(ต่อ)

	หน้า
บทที่ 3 ระเบียบวิธีวิจัย	51
3.1 ขั้นตอนการดำเนินงาน	51
3.2 แผนการดำเนินงาน	58
บทที่ 4 ผลการทดลอง	59
4.1 การทดลองแบบให้ผู้ทดลองยื่นมือไปข้างหน้าเป็นระยะทาง 30 เซนติเมตรโดยวัดผลขณะเคลื่อนไหวไปด้วยจำนวน 5 รอบ	59
4.2 การทดลองแบบให้ผู้ทดลองยื่นมือไปข้างหน้าเป็นระยะทาง 30 เซนติเมตรแล้วจับวัตถุที่ตั้งไว้จากนั้นยกขึ้นแล้ววางลงโดยวัดผลขณะเคลื่อนไหวไปด้วยเช่นกันจำนวน 5 รอบ	69
บทที่ 5 สรุปผลการทดลอง	79
เอกสารอ้างอิง	84
ภาคผนวก	85

สารบัญรูป

รูปที่	หน้า
2.1 การใช้โกนิโอมิเตอร์ในการวัดพิสัยการเคลื่อนไหวของสะโพก	7
2.2 การใช้โกนิโอมิเตอร์ในการวัดพิสัยการเคลื่อนไหวของข้อเข่า	7
2.3 โปรแกรม Arduino IDE	11
2.4 โปรแกรม Processing	12
2.5 โปรแกรม Microsoft Visual Studio 2010	13
2.6 หน้าจอเริ่มต้นการใช้งานของโปรแกรม	14
2.7 การสร้างโปรเจคใหม่	15
2.8 หน้าจอ Windows Form Application	16
2.9 ส่วนต่าง ๆ ของโปรแกรม Visual Studio 2010	17
2.10 ส่วนต่าง ๆ ของโปรแกรม Microsoft Excel	18
2.11 หน้าต่างบันทึกไฟล์	20
2.12 รูปแบบกราฟชนิดต่าง ๆ	21
2.13 โปรแกรม Matlab	22
2.14 9DoF Razor IMU	29
2.15 การต่อ IMU เข้ากับ Arduino	30
2.16 การเชื่อมต่อ IMU โดยต่อผ่าน FDTI Basic	31
2.17 แสดงชนิดของเซนเซอร์วัดความเร่ง	31
2.18 แสดงหลักการของเซนเซอร์วัดความเร่ง	32
2.19 เครื่องวัดสภาพความเอียง	34
2.20 ไจโรสโคปรักษาเสถียรภาพของเรือ	35
2.21 วงล้อหมุนตามกฎของนิวตัน	36
2.22 เครื่องวัดสนามแม่เหล็ก	38
2.23 การเคลื่อนที่แนวตรง	41
2.24 แสดงการเคลื่อนที่แนวตรงของรถยนต์คันหนึ่ง	42
2.25 เครื่องเคาะสัญญาณเวลา	43
3.1 รูปการแสดงค่า raw data จาก IMU บนโปรกรม Arduino	51
3.2 แสดง Flow chart การทำงาน	52
3.3 โปรแกรมสำหรับ plot กราฟ	53
3.4 การแสดงค่าข้อมูลที่ได้รับจาก IMU	54
3.5 การติด sensor ตรวจวัด	55
3.6 การยื่นมือออกไปด้านหน้าตรงๆในระยะ 30 เซนติเมตรจากนั้นดึงมือกลับ	55

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.7 การยื่นมือออกไปข้างหน้าตรงๆ แล้วหยิบวัตถุที่วางไว้ยกขึ้นและวางลงจากนั้นดึงมือกลับ	56
3.8 กราฟที่ได้จากการเปรียบเทียบกัน	57
3.9 ภาพที่ได้จากโปรแกรม Processing	57
4.1 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 1	59
4.2 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 1	60
4.3 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 1	60
4.4 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 2	61
4.5 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 2	62
4.6 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 2	62
4.7 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 3	63
4.8 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 3	64
4.9 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 3	64
4.10 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 4	65
4.11 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 4	66
4.12 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 4	66
4.13 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 5	67
4.14 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 5	68
4.15 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 5	68
4.16 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลอง2ครั้งที่ 1	69
4.17 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลอง2ครั้งที่ 1	70
4.18 กราฟที่ได้จากการทดลองของผู้จัดทำในการทดลอง2ครั้งที่ 1	70
4.19 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลอง2ครั้งที่ 2	71
4.20 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลอง2ครั้งที่ 2	72
4.21 กราฟที่ได้จากการทดลองของผู้จัดทำในการทดลอง2ครั้งที่ 2	72
4.22 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลอง2ครั้งที่ 3	73
4.23 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลอง2ครั้งที่ 3	74
4.24 กราฟที่ได้จากการทดลองของผู้จัดทำในการทดลอง2ครั้งที่ 3	74
4.25 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลอง2ครั้งที่ 4	75
4.26 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลอง2ครั้งที่ 4	76
4.27 กราฟที่ได้จากการทดลองของผู้จัดทำในการทดลอง2ครั้งที่ 4	76

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.28 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลอง2ครั้งที่ 5	77
4.29 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลอง2ครั้งที่ 5	78
4.30 กราฟที่ได้จากการทดลองของผู้จัดทำในการทดลอง2ครั้งที่ 5	78

สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางแสดงค่าปกติของพิสัยของข้อในบริเวณต่างๆ	9
2.2 ตัวดำเนินการเครื่องหมายพิเศษการคำนวณ	25
2.3 ตัวดำเนินการเปรียบเทียบและตรรกะลักษณะดำเนินการ	26
2.4 ฟังก์ชั่นที่ใช้ในการหาค่าทางตรีโกณมิติ	27
2.5 ฟังก์ชั่นที่ใช้ในการหาค่า Exponential	28
3.1 ตารางแผนการดำเนินงาน	58
5.1 แสดงการเปรียบเทียบผลการทดลองที่ 1	80
5.2 แสดงการเปรียบเทียบผลการทดลองที่ 2	82

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในการให้การรักษาทางกายภาพบำบัดได้อย่างมีประสิทธิภาพนั้น ผู้ให้การรักษาควรมีความสามารถในการตรวจประเมิน วิเคราะห์ปัญหาของการเคลื่อนไหวที่ทำให้เกิดความบกพร่อง ในการทำกิจกรรมประจำวัน ทั้งในส่วนของกิจกรรมที่บ้าน ที่ทำงาน การเล่นกีฬา หรือนันทนาการ การตรวจประเมินอย่างถูกต้อง เที่ยงตรง จะทำให้ผู้ทำการรักษาสามารถค้นหาปัญหาได้อย่างแม่นยำ ให้การรักษาได้อย่างเหมาะสม มีประสิทธิภาพ เพื่อให้ผู้ป่วยกลับคืนสู่ภาวะปกติ

การตรวจประเมินต้องเชื่อถือได้ มีจุดมุ่งหมาย และเหตุผลในการตรวจแต่ละประเภท ข้อมูลที่ได้ต้องมีความน่าเชื่อถือนั่นคือเมื่อมีการตรวจสอบซ้ำจะได้ค่าคงที่ เช่น ในการวัดช่วงการเคลื่อนไหว (range of motion - ROM) ของข้อไหล่ ในผู้ป่วยที่มีข้อไหล่ติด โดยนักกายภาพบำบัดหนึ่งคนหรือมากกว่า ได้ค่าของการวัดเท่ากันทุกครั้ง ถือว่าการทดสอบนี้เชื่อถือได้ หลักสำคัญของความเชื่อถือได้ในการตรวจประเมินนั้น ขึ้นอยู่กับผู้ทำการรักษาได้ปฏิบัติตามขั้นตอนมาตรฐานทุกครั้งที่ทำการตรวจวัด ทำการทดสอบในลักษณะเดิมทุกครั้ง และในลักษณะเดียวกันกับผู้ให้การรักษาคนอื่น ๆ ความน่าเชื่อถือจะเพิ่มมากขึ้นถ้าผู้ให้การรักษาได้ให้คำอธิบายแก่ผู้ป่วยอย่างชัดเจน

ข้อมูลที่ได้จากการวัดช่วงการเคลื่อนไหวของข้อต่อในส่วนต่าง ๆ ของร่างกายนั้น สามารถนำไปใช้ประโยชน์ในการรักษาอาการสำคัญต่าง ๆ ของผู้ป่วยได้ เช่น ข้อต่อล็อค การอักเสบจากการผ่าตัด ความตึงตัวของกล้ามเนื้อ ซึ่งอาการดังกล่าวมักจะส่งผลให้การเคลื่อนไหวในบริเวณต่าง ๆ ของร่างกายถูกจำกัด ดังนั้น การจัดลำดับกระบวนการรักษาที่มีประสิทธิภาพ รวมถึงการให้คำแนะนำแก่ผู้ป่วยในการจัดท่าทางที่ถูกต้อง จะช่วยให้องศาการเคลื่อนไหวในส่วนต่าง ๆ ของร่างกายมีประสิทธิภาพมากขึ้น เพิ่มคุณภาพชีวิตให้แก่ผู้ป่วยให้สามารถกลับเข้าร่วมกิจกรรมต่าง ๆ ได้

1.2 วัตถุประสงค์การวิจัย

1.2.1. เพื่อศึกษาลักษณะของการเคลื่อนไหวที่ส่วนต่าง ๆ ของร่างกาย

1.2.2. เพื่อความสะดวกในการเก็บข้อมูลการเคลื่อนไหวของร่างกาย

1.2.3. เป็นการเสริมสร้างและพัฒนาอุปกรณ์ทางการแพทย์ให้มีประสิทธิภาพยิ่งขึ้น

1.3 สมมติฐานการวิจัย

1.3.1. ผู้ทำการรักษาสามารถเก็บข้อมูลการเคลื่อนไหวได้อย่างมีประสิทธิภาพมากขึ้น

1.3.2. ผู้ทำการรักษาสามารถนำข้อมูลที่ได้ไปใช้ในการวิเคราะห์เพื่อวางแผนการรักษาต่อไปได้

1.4 ขอบเขตของงานวิจัย

ศึกษาและออกแบบอุปกรณ์สำหรับวัดระยะการเคลื่อนไหวของร่างกาย เก็บข้อมูลการเคลื่อนไหวบริเวณต่าง ๆ ของร่างกาย สรุปผลที่ได้แล้วนำมาพล็อตข้อมูลเป็นกราฟนำข้อมูลที่ได้มาแสดงผลบนแอพพลิเคชั่นตรวจสอบความถูกต้องของข้อมูลและนำไปใช้ในการรักษาทางคลินิค

1.5 ขั้นตอนการศึกษาวิจัย

1.5.1. ศึกษาข้อมูลการทำงานวิจัยจากงานวิจัยที่เกี่ยวข้อง

ค้นคว้างานวิจัยที่เกี่ยวข้องกับการวัดการเคลื่อนไหวของร่างกาย และการวิเคราะห์ความผิดปกติของการเคลื่อนไหว ศึกษากระบวนการสร้างระบบและการวิเคราะห์เพื่อนำมาประยุกต์ใช้ในงานวิจัย

1.5.2. การสร้างระบบการเก็บข้อมูลของเซ็นเซอร์

ศึกษาและค้นหาวัสดุที่ต้องนำมาใช้ในการสร้าง ได้แก่ เซ็นเซอร์ IMU

1.5.3. การสร้างโปรแกรมสำหรับเก็บข้อมูลของเซ็นเซอร์ที่มีลักษณะเป็นความเร่ง

ศึกษาโปรแกรมที่มีประสิทธิภาพ ใช้งานสะดวกและสามารถเข้าใจง่าย ศึกษาการเขียนคำสั่งพื้นฐานและนำมาประยุกต์ใช้

1.5.4. การเก็บข้อมูลเพื่อวินิจฉัย

เมื่อสร้างระบบการเก็บข้อมูลเรียบร้อยแล้ว จะต้องทำการเก็บข้อมูลเพื่อนำมาวินิจฉัย ซึ่งเป็นการทดสอบความถูกต้องและแม่นยำของระบบที่สร้างขึ้น

1.5.5. การวิเคราะห์ผล

นำข้อมูลที่ได้จากการเก็บรวบรวมมาวิเคราะห์เพื่อหาความถูกต้องและความแม่นยำของระบบ เพื่อนำข้อผิดพลาดไปแก้ไขระบบให้มีประสิทธิภาพมากขึ้น

1.6 โครงสร้างวิทยานิพนธ์

โครงสร้างของวิทยานิพนธ์เล่มนี้ประกอบไปด้วย

1.6.1 บทนำ ความเป็นมาและความสำคัญ

1.6.2 ทฤษฎีและความรู้ที่เกี่ยวข้อง

1.6.3 ระเบียบวิธีวิจัย

1.6.4 ผลการทดลอง

1.6.5 สรุปผลการทดลอง

1.6.6 ภาคผนวก

บทที่ 2
ทฤษฎีและความรู้พื้นฐานที่เกี่ยวข้อง

2.1 งานวิจัยที่เกี่ยวข้อง

งานวิจัยเรื่อง An analysis of full range of motion vs. partial range of motion training in the development of strength in untrained men.จัดทำโดย Massey CD, Vincent J, Maneval M, Moore M และ Johnson JT. [1]เป็นงานวิจัยเกี่ยวกับการเคลื่อนไหวโดยการเปรียบเทียบระหว่างเคลื่อนไหวจนสุดกับเคลื่อนไหวแค่บางส่วนเพื่อเพิ่มความแข็งแรงของกล้ามเนื้อ ซึ่งวิธีดังกล่าวมีข้อดี คือ สามารถเปรียบเทียบพัฒนาการของกล้ามเนื้อระหว่างเคลื่อนไหวจนสุดกับเคลื่อนไหวเพียงบางส่วนได้ และข้อมูลที่ได้นั้นสามารถนำมาใช้ในการวางแผนเพื่อการรักษาคนไข้ในอนาคตได้

2.2 เวชศาสตร์ฟื้นฟู

เป็นการบริการทางการแพทย์ชนิดหนึ่ง เพื่อตรวจวินิจฉัยโรค ประเมิน รักษา ฟื้นฟูสมรรถภาพ ด้วยวิธีการใช้ยา การทำหัตถการ การใช้เครื่องมือ การออกกำลังกายจำเพาะ การให้คำแนะนำทางการแพทย์ การใช้อุปกรณ์ช่วยเหลือหรือทดแทน หรือวิธีการอื่นๆ อีกทั้งยังมุ่งส่งเสริมสุขภาพ และป้องกันการเป็นซ้ำหรือภาวะแทรกซ้อนให้กับบุคคลทั่วไป และผู้ป่วยที่มีความพิการหรือสมรรถภาพเสื่อมถอย ทั้งทางร่างกาย ทางสติปัญญา ทางการเรียนรู้ ทางการสื่อความหมาย และทางจิตใจ โดยใช้บุคลากรที่เกี่ยวข้องจากหลายๆสาขา ร่วมกันให้การรักษาและฟื้นฟู เพื่อส่งเสริมศักยภาพที่เหลืออยู่ของผู้ป่วยนั้นๆ ให้สามารถดำรงชีวิตในสภาวะแวดล้อมที่เหมาะสมได้ เพื่อให้เป็นภาระต่อคนรอบข้างและสังคมให้น้อยที่สุด อีกทั้งยังช่วยสร้างชื่อเสียง (เช่น เป็นนักกีฬา) หรือพัฒนาประเทศต่อไปได้ตามความสามารถ

2.3 ออร์โธปิดิคส์

ออร์โธปิดิกส์ เป็นชื่อสาขาวิชาแพทย์เฉพาะทางสาขาหนึ่ง ซึ่งเป็นผู้ดูแลรักษาคนไข้ที่มีปัญหาเรื่องเกี่ยวกับ กระดูกและข้อ เอ็นยึดกระดูก กล้ามเนื้อและเส้นประสาท การดูแลรักษาจะประกอบไปด้วย การวิเคราะห์โรค การให้การรักษาทางยา การให้การรักษาเพื่อฟื้นฟูสมรรถภาพของการทำงานของกล้ามเนื้อกระดูกและข้อให้กลับคืนมา โดยร่วมมือกับแพทย์เฉพาะทางสาขาเวชศาสตร์ฟื้นฟู และผู้เชี่ยวชาญทางด้านกายภาพบำบัด ตลอดจนการให้การรักษาโดยการผ่าตัดแพทย์ผู้เชี่ยวชาญในสาขานี้บางทีเรียกว่า ศัลยแพทย์ออร์โธปิดิกส์ หรือศัลยแพทย์โรคกระดูกและข้อซึ่งต้องเข้าฝึกศึกษาอบรมต่อภายหลังจบเป็นแพทย์แล้วอีก 4 ปี นอกจากนี้ ถ้าหากฝึกอบรมต่อเพื่อให้เกิดความชำนาญในสาขาวิชาย่อยลงไปอีก จะต้องฝึกอบรมต่ออีก 1-2 ปี สาขาย่อยของออร์โธปิดิกส์ ได้แก่ การดูแลรักษาคนไข้ออร์โธปิดิกส์ที่มีความชำนาญในเรื่องเด็ก ชำนาญในเรื่องมือ ชำนาญในเรื่องกระดูกสันหลัง ชำนาญในเรื่องเวชศาสตร์การกีฬา ชำนาญในเรื่องการเปลี่ยนข้อเทียม เป็นต้น

2.4 เวชศาสตร์ฟื้นฟูทางออร์โธปีดิคส์

นิยามของคำว่าเวชศาสตร์ฟื้นฟูทางออร์โธบีดิคส์ หมายถึงการทำให้หน้าที่ของระบบกระดูกและกล้ามเนื้อกลับคืนดีเหมือนเดิม ตลอดจนถึงข้อและส่วนที่สัมพันธ์กัน โดยความหมายนี้ผู้ให้การรักษาต้องรับผิดชอบบำบัดให้คนไข้หายกลับไปใช้ชีวิตได้ตามปกติ หลังจากการเจ็บป่วย หรือประสบอุบัติเหตุ หรือให้เด็กที่เกิดมามีความผิดปกติในระบบกระดูกกล้ามเนื้อ ได้เติบโตพร้อมกับใช้ส่วนที่มีอยู่ทำหน้าที่ทดแทนตามความสามารถที่มีอยู่ ให้เขาเหล่านี้ดำเนินชีวิตอยู่ได้อย่างมีความสุข อย่างไรก็ตามคนที่ปกติดีอยู่แล้วกลับไร้ความสามารถเนื่องจากภยันตรายหรือโรคก็ตาม การที่จะให้ร่างกายกลับทำหน้าที่ได้เต็มที่คืนดีดังเดิมนั้นมักจะต้องใช้ความพยายามอย่างมากในการใช้วิธีต่างๆ ที่เป็นประโยชน์และเพื่อช่วยฟื้นสภาพ ให้คนไข้เหล่านี้สามารถช่วยตัวเองได้

เวชศาสตร์ฟื้นฟูทางออร์โธปีดิคส์มีคำศัพท์ที่ใช้อยู่เสมอคือ**พิสัยการเคลื่อนไหว** (Range of Motion, R.O.M.)หมายถึงการเคลื่อนไหวหรือความสามารถที่จะเคลื่อนไหวที่ข้อตามแนวการเคลื่อนไหวที่ข้อใดข้อหนึ่ง โรคระบบกล้ามเนื้อกระดูกหลายอย่าง ทำให้หรือมีส่วนร่วมทำให้การเคลื่อนไหวข้อเสียไป ในการตรวจสอบการเคลื่อนไหวที่ข้อผู้ให้การรักษาต้องเปรียบเทียบกับการเคลื่อนไหวเป็นปกติที่แต่ละข้อด้วยเสมอ สิ่งที่ใช้วัดพิสัยการเคลื่อนไหวที่ข้อนี้เรียกว่าโกนิโอมิเตอร์ และใช้หน่วยวัดเป็นองศา

2.5 การวัดพิสัยการเคลื่อนไหว

ในการวินิจฉัยความผิดปกติของการเคลื่อนไหวนั้นจะใช้โกนิโอมิเตอร์ในการวัดพิสัยการเคลื่อนไหวของร่างกาย โดยค่าที่ได้จะเป็นค่าของมุมระหว่างข้อต่อที่กระทำต่อกันโดยเริ่มจาก 0° ค่าของมุมที่ได้มานั้นสามารถนำไปวินิจฉัยความผิดปกติของกล้ามเนื้อได้ โดยมีนิยามของการวัดพิสัยการเคลื่อนไหวดังนี้

2.5.1 การงอข้อ (Flexion) หมายถึง การเคลื่อนไหวของข้อจากตำแหน่งเริ่มต้นไปจนงอข้อเต็มที่

2.5.2 การเหยียดข้อ (Extension) หมายถึง การเคลื่อนไหวในทิศทางที่ตรงกันข้ามกับการงอข้อ (Flexion)

2.5.3 การเหยียดข้อไม่สุด (Extension Lag) หมายถึง ข้อเหยียดได้ไม่เต็มที่จากตำแหน่งงอข้อคือเหยียดไปไม่ถึงจุดเริ่มต้น0^0 ใช้แทนด้วยเครื่องหมาย – (ลบ)

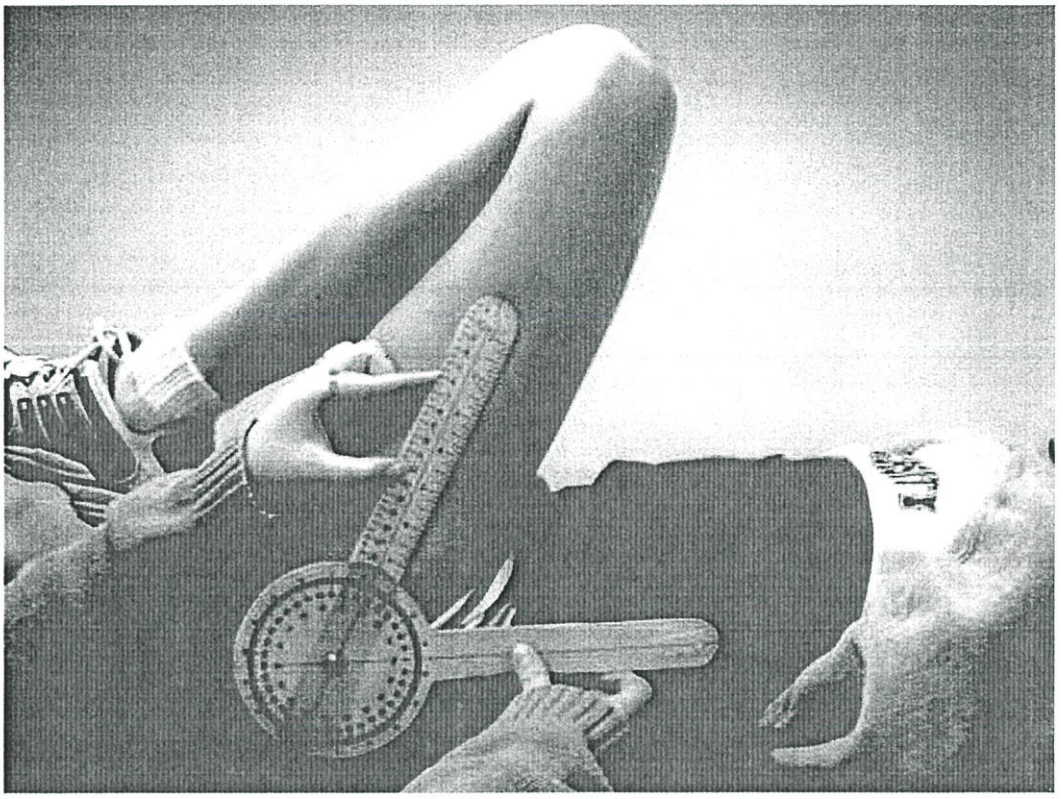
2.5.4 การเหยียดข้อเกิน (Hyperextension) หมายถึง ข้อที่เหยียดได้เกินกว่าจุดเริ่มต้น 0^0 ซึ่งพบได้เป็นปกติที่ข้อโคนนิ้ว (Metacarpophalangeal Joint) ที่ข้อศอก ที่ข้อเข่า ใช้แทนด้วยเครื่องหมาย + (บวก)

2.5.5 ข้อติดแข็ง (Ankylosis) หมายถึง ข้อที่ไม่มีการเคลื่อนไหวเลย

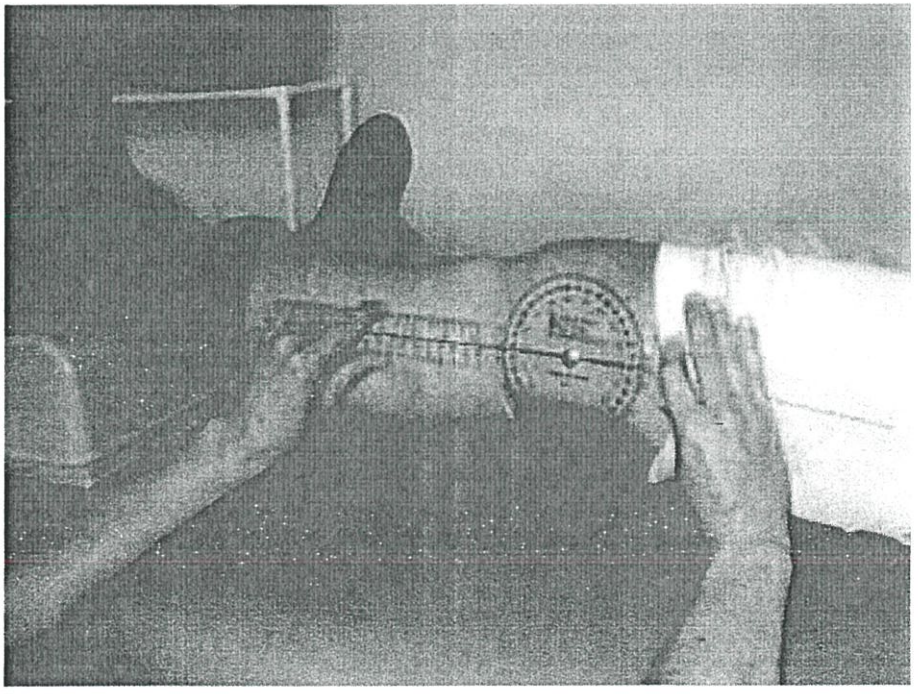
2.5.6 ข้อยึดในท่างอ (Flexion Contracture) หมายถึง ข้อที่ยังมีการเคลื่อนไหวได้บ้างแต่ไม่ถึงตำแหน่งกลาง เช่น ถ้าข้อนิ้วมือติดในท่างอ 15^0 และสามารถงอไปถึง 45^0 ก็ให้เขียนลงบันทึกว่า -15^0 ถึง 45^0

2.5.7 มุมที่ข้อเคลื่อนไหว คือ จำนวนองศาที่อยู่ระหว่างตำแหน่งเริ่มต้นและตำแหน่งสุดท้ายของการเคลื่อนไหวของข้อในระนาบนั้น ๆ เช่นมุมการเคลื่อนไหวของข้อกลางนิ้วจากตำแหน่งงอเต็มที่ จนถึงเหยียดเต็มที่ ถ้าข้อมีการเคลื่อนไหวมากกว่าหนึ่งระนาบ การเคลื่อนไหวในแต่ละระนาบหรือแต่ละชนิดให้ เรียกว่าหน่วยการเคลื่อนไหว (Unit of Motion) ตัวอย่างเช่น ข้อมือมีหน่วยการเคลื่อนไหว 2 หน่วย คืองอ/เหยียด ในระนาบหน้าหลัง และเบนไปทางนิ้วหัวแม่มือ / นิ้วก้อย ในระนาบด้านข้าง เป็นต้น

2.5.8 ตำแหน่งที่ข้อใช้งานได้ดี (Functional Position of a Joint) หมายถึง ตำแหน่งที่ข้อติดแข็งหรือหลังการผ่าตัดเชื่อมข้อแล้วทำให้ข้อนั้นมีการสูญเสียสมรรถภาพการทำงานน้อยที่สุด ถ้าข้อมีหน่วยการเคลื่อนไหวมากกว่าหนึ่งหน่วยขึ้นไป แต่ละหน่วยการเคลื่อนไหวก็จะมีตำแหน่งที่ข้อใช้งานได้ดี ตัวอย่างเช่น ตำแหน่งที่ข้อศอกใช้งานได้ดี คือ งอ 80^0 และคว่ำฝ่ามือ 20^0



รูปที่ 2.1 การใช้โกนิโอมิเตอร์ในการวัดพิสัยการเคลื่อนไหวของสะโพก



รูปที่ 2.2 การใช้โกนิโอมิเตอร์ในการวัดพิสัยการเคลื่อนไหวของข้อเข่า

2.6 การออกกำลังกายเพื่อการบำบัดรักษา

การออกกำลังกายเพื่อการบำบัดรักษา (Therapeutic exercise) คือการเคลื่อนไหวส่วนใดส่วนหนึ่งหรือทุกส่วนของร่างกายเพื่อการบำบัดรักษา ลดอาการของผู้ป่วย หรือช่วยเพิ่มประสิทธิภาพการทำงานของร่างกายให้ดีขึ้นการออกกำลังกายเพื่อการบำบัดรักษาเริ่มมีมานานตั้งแต่ก่อนประวัติศาสตร์ มีการนำการออกกำลังกายมาใช้ในการแพทย์อย่างจริงจังสมัยกรีกต่อมาในสมัยโรมันการออกกำลังกายเป็นที่นิยมมากจนมีการแข่งขันกีฬาโอลิมปิก เมื่อการแพทย์เจริญมากขึ้นสามารถอธิบายกลไก ประโยชน์และข้อระวังของการออกกำลังกายจนในปัจจุบันเป็นที่ยอมรับและเข้าใจว่าการออกกำลังกายเป็นส่วนหนึ่งของการบำบัดรักษาผู้ป่วยการนำการออกกำลังกายไปใช้ในการบำบัดรักษาก่อนอื่นต้องรู้วัตถุประสงค์ที่ต้องการจากการออกกำลังกาย แล้วจึงพิจารณาชนิด (mode) ความแรง (intensity) จำนวนครั้ง-เวลา (repetition-time)ความถี่(frequency) ให้เหมาะกับผู้ป่วยแต่ละคน

2.7 การออกกำลังกายเพื่อป้องกันข้อติด หรือเพิ่มพิสัยการเคลื่อนไหวของข้อ

เมื่อมีการอักเสบ บาดเจ็บ หรือขาดการเคลื่อนไหว (immobilization) จะมีผลทำให้พิสัยการเคลื่อนไหวของข้อค่อยๆลดลง ในผู้ที่ยังไม่มีข้อติดแต่เสี่ยงต่อภาวะข้อติดเช่น กล้ามเนื้ออ่อนแรง มีภาวะกล้ามเนื้อเกร็ง หรือบวมรอบๆข้อ การขยับข้อให้สุดพิสัยการเคลื่อนไหวทำเพื่อป้องกันข้อติด แต่ในผู้ป่วยที่มีข้อติดแล้วการขยับข้อทำเพื่อเพิ่มพิสัยการเคลื่อนไหวของข้อการยึดติดของข้อเกิดจากการหดสั้นของกล้ามเนื้อและเนื้อเยื่อรอบข้อ การหล่อลื่นของข้อลดลง การเพิ่มของเนื้อเยื่อชนิด collagen และ reticulin ทำให้ connective tissue แปลงสภาพจาก loose connective tissue กลายเป็น dense connective tissue ซึ่งจะเกิดในข้อที่ขาดการเคลื่อนไหว (immobility) นานเกิน 1 สัปดาห์ หรือถ้ามีการอักเสบหรือการขาดเลือดมาเลี้ยง การเปลี่ยนแปลงนี้จะยิ่งเร็วขึ้น

Range of motion exercise (ROM exercise) แบ่งออกได้เป็น 4 แบบคือ

1. Active exercise คือให้ผู้ป่วยออกแรงขยับข้อเองทั้งหมด

2. Acitive-assistive exercise คือให้ผู้ป่วยออกแรงขยับให้เต็มที่ก่อนแล้วใช้แรงจากภายนอกหรือผู้อื่นช่วยขยับต่อจนสุดพิสัยการเคลื่อนไหวของข้อ

3.Passive exercise คือให้ผู้ช่วยหรือใช้แรงจากภายนอกเป็นผู้ขยับตลอดพิสัยการเคลื่อนไหวของข้อ โดยผู้ป่วยไม่ได้ออกแรงเลย

4.Passive stretching exercise คือผู้ช่วยหรือผู้บำบัดช่วยดัดยืดเพื่อเพิ่มพิสัยการเคลื่อนไหวข้อในกรณีที่มีข้อติด

2.8 ค่าปกติของพิสัยการเคลื่อนไหว

การวัดพิสัยการเคลื่อนไหวของข้อสามารถบันทึกค่าตามคำบรรยายในตำราหรือใช้แบบที่เป็นตัวเลขหรือจำนวนการวัดแบบเป็นตัวเลขนั้นบังคับใช้ในบางรัฐของประเทศสหรัฐอเมริกาและบางประเทศในทวีปยุโรปเพื่อให้เกิดความง่ายสะดวกมีประสิทธิภาพเป็นแบบเดียวกันในการบันทึกและการนำข้อมูลกลับมาใช้ จึงนิยมการบันทึกแบบเป็นตัวเลขที่เรียกว่า SFTR Method (Standard Figure Technique for Record Method) สำหรับใช้ในการบันทึกพิสัยการเคลื่อนไหวของข้อ การบันทึกแบบ SFTR Method จะบันทึกพิสัยการเคลื่อนไหวของข้อโดยใช้อักษรหนึ่งตัว และตัวเลขสามจำนวน การบันทึกแบบตัวเลขเป็นการย่อที่สมบูรณ์ ลดความผิดพลาดจากการคัดลอกหรือลดความเข้าใจผิดและช่วยให้การติดต่อกันง่ายขึ้นด้วยตัวเลขที่เป็นภาษาสากลการวัดแบบ SFTR Method ยังช่วยทำให้การแปลข้อมูลจากรายงานไปเป็นระบบคอมพิวเตอร์ได้ง่าย ดังแสดงให้เห็นในตารางที่ 2.1

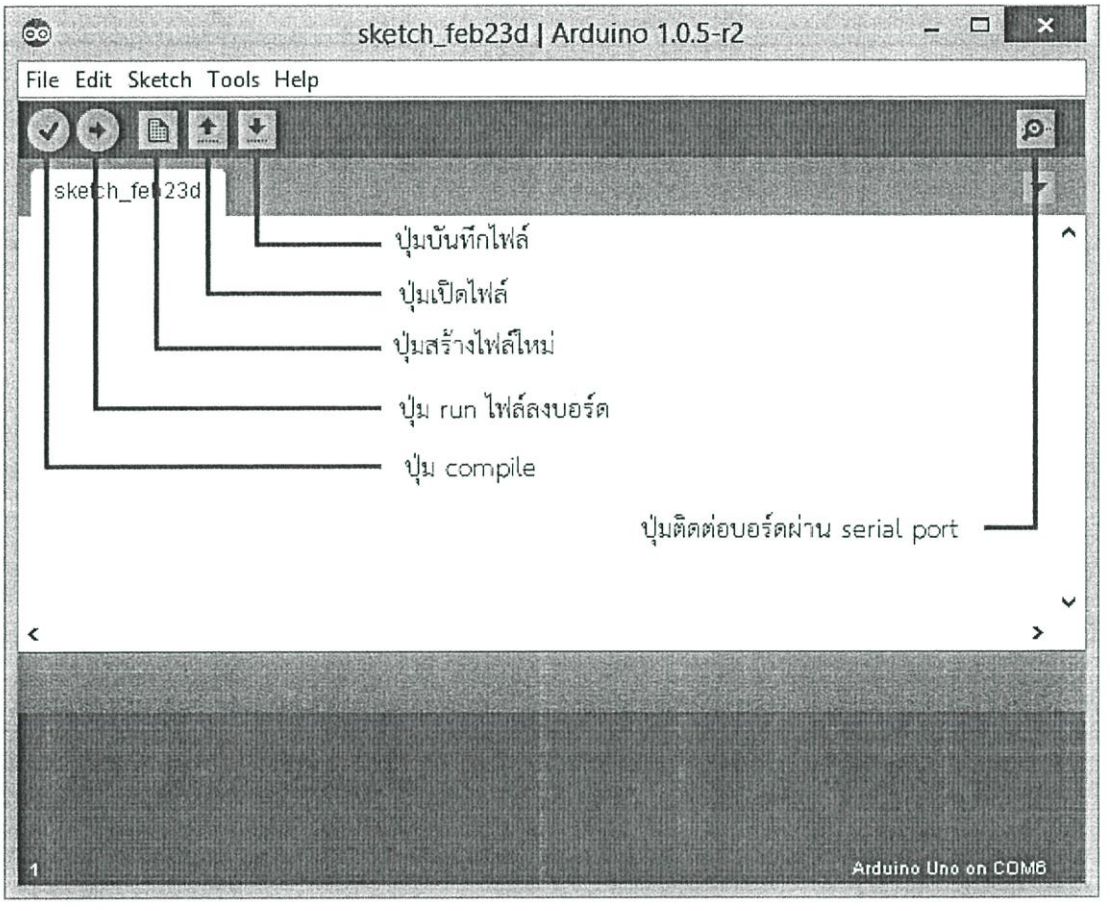
ข้อ	ระนาบ	ค่าปกติของพิสัยการเคลื่อนไหวของข้อที่ทำได้ด้วยตัวเอง (องศา)
Shoulder	Sagittal	Extension – 0 – Flexion (40 – 0 – 180)
Shoulder	Frontal	Abduction – 0 Adduction (180 – 0 – 30)
Shoulder	Rotation	Eversion – 0 – Inversion (90 – 0 – 80)
Elbow	Sagittal	Extension – 0 – Flexion (0 – 0 – 150)
Lower Arm	Rotation	Supination – 0 – Pronation (80 – 0 – 80)
Wrist	Sagittal	Elevation – 0 - Depression (60 – 0 – 60)
Wrist	Frontal	Eversion – 0 – Inversion (20 – 0 - 30)
Cervical (Spine)	Sagittal	Extension – 0 - Flexion (60 – 0 – 50)
Cervical (Spine)	Frontal	Lt. Flexion – 0 – Rt. Flexion (45 – 0 – 45)
Cervical (Spine)	Rotation	Lt. Rotation – 0 – Rt. Rotation (80 – 0 – 80)

Thoracic (Spine)	Sagittal	Extension – 0 – Flexion (0 – 0 – 45)
Thoracic (Spine)	Frontal	Lt. Flexion – 0 – Rt. Flexion (45 – 0 – 45)
Thoracic (Spine)	Rotation	Lt. Rotation – 0 – Rt. Rotation (30 – 0 – 30)
Lumbar (Spine)	Sagittal	Extension – 0 – Flexion (25 – 0 – 60)
Lumbar (Spine)	Frontal	Lt. Flexion – 0 – Rt. Flexion (25 – 0 – 25)
Knee	Sagittal	Extension – 0 - Flexion (0 – 0 – 150)
Ankle (Talocrural)	Sagittal	Elevation – 0 – Depression (20 – 0 – 40)
Ankle (Subtalar)	Frontal	Eversion – 0 –Inversion (20 – 0 – 30)

ตารางที่ 2.1 ตารางแสดงค่าปกติของพิสัยของข้อในบริเวณต่าง ๆ

2.9 การเขียนโปรแกรมโดย Arduino

อาดูโน่เป็นโอเพ่นซอร์สทั้งฮาร์ดแวร์และซอฟท์แวร์ ดังนั้นในการเขียนโปรแกรมควบคุมการทำงานของบอร์ดอาดูโน่จึงมีเครื่องมือสำหรับเขียนโปรแกรมมาให้ใช้สำหรับบอร์ดอาดูโน่โดยเฉพาะ เครื่องมือที่ใช้สำหรับเขียนโปรแกรมนี้คือ Arduino IDE (Arduino integrated development environment (IDE))หน้าตาของโปรแกรมและไอคอนที่ใช้งานดังรูปที่ 2.1

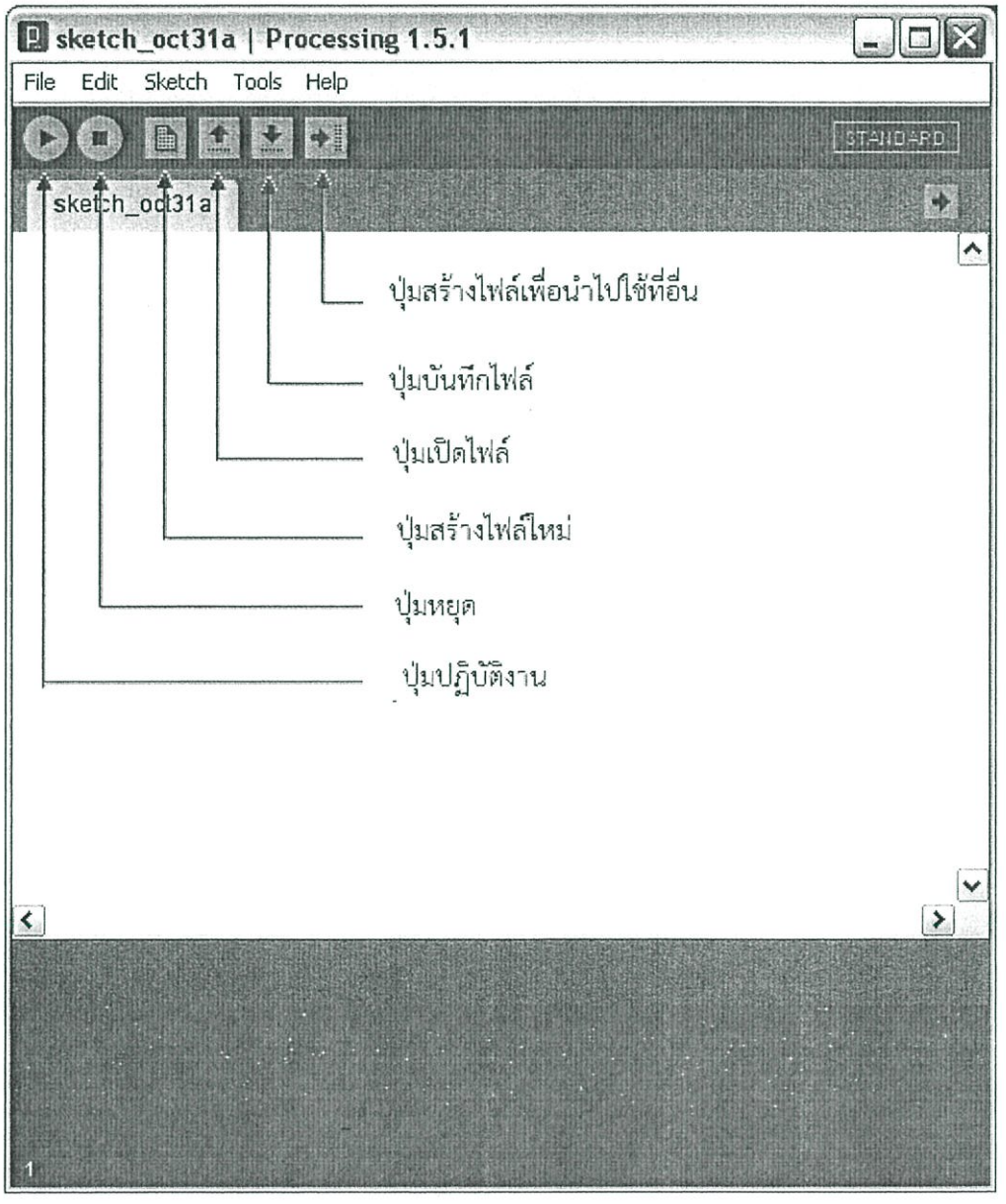


รูปที่ 2.3 โปรแกรม Arduino IDE

โปรแกรมของ Arduino แบ่งได้ เป็นสองส่วนคือvoid setup() และ void loop() โดยฟังก์ชั่น setup() เมื่อโปรแกรมทำงานจะทำคำสั่งของฟังก์ชั่นนี้เพียงครั้งเดียวใช้ในการกำหนดค่าเริ่มต้นของการทำงาน ส่วนฟังก์ชั่น loop() เป็นส่วนทำงาน โปรแกรมจะทำคำสั่งในฟังก์ชั่งนี้ต่อเนื่องตลอดเวลา โดยปกติใช้ กำหนดโหมดการทำงานของขาต่างๆ กำหนดการสื่อสารแบบอนุกรม ฯลฯ ส่วนของ loop() เป็นโค้ดโปรแกรมที่ทำงาน เช่น อ่านค่าอินพุต ประมวลผลสั่งงาน เอาต์พุต ฯลฯ โดยส่วนกำหนดค่าเริ่มต้น เช่น ตัวแปรจะต้องเขียนที่ส่วนหัวของโปรแกรม ก่อนถึงตัวฟังก์ชั่นนอกจากนั้นยังต้อง คำนึงถึงตัวพิมพ์ เล็ก-ใหญ่ ของตัวแปรและชื่อฟังก์ชั่นให้ถูกต้อง

2.10 การเขียนโปรแกรมโดย Processing

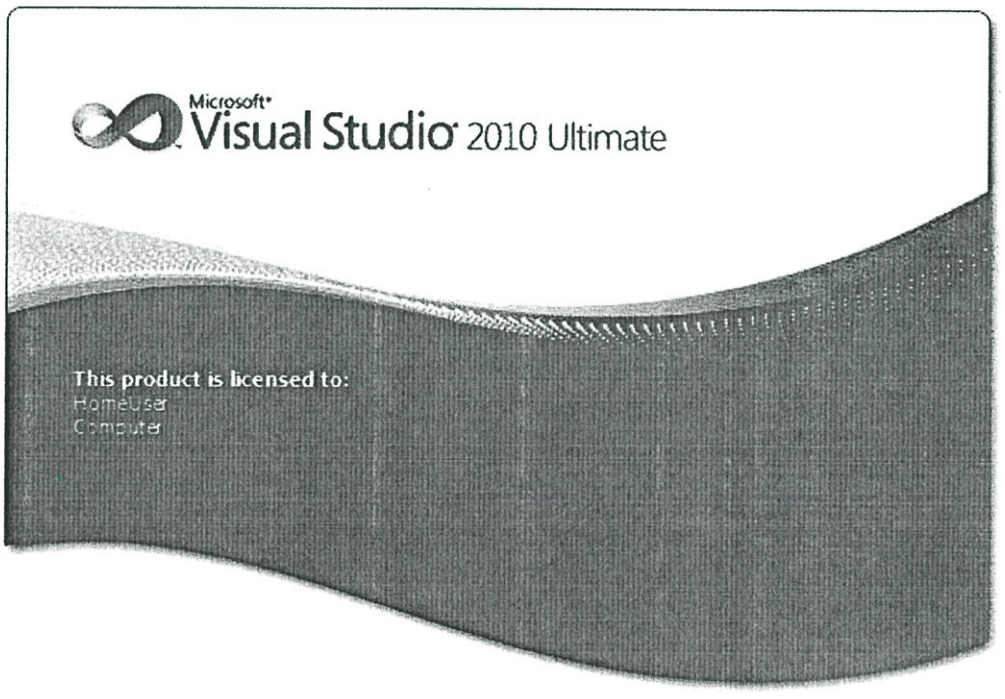
โปรแกรม Processing เป็นซอฟท์แวร์ระบบเปิด เหมาะสำหรับผู้ต้องการพัฒนาโปรแกรมเกี่ยวกับการสร้างภาพเคลื่อนไหวและการเชื่อมต่อกับอุปกรณ์ต่าง ๆ โดยรูปร่างหน้าตาของโปรแกรม Processing นั้นจะมีความคล้ายคลึงกับ Arduino มาก นอกจากนั้นโปรแกรมทั้งสองยังใช้หลักการในการเขียนโปรแกรมเหมือนกัน โดยมีพื้นฐานมาจากภาษา C/C++



รูปที่ 2.4 โปรแกรม Processing

2.11 การเขียนโปรแกรมโดย Microsoft Visual Studio

Microsoft Visual Studioเป็นเครื่องมือที่ช่วยให้เราเขียนโปรแกรมด้วยภาษา Visual Basic .NET ได้ Microsoft Visual Studio เป็นชุดพัฒนาแบบ IDE Integrated Development Environment ซึ่งหมายถึง สภาพแวดล้อมที่รวบรวมเครื่องมือและคุณสมบัติทุกอย่างที่จำเป็นสำหรับการพัฒนาโปรแกรมเข้าไว้ด้วยกันในที่เดียวกัน ไม่ว่าจะเป็นการออกแบบหน้าจอ เขียนโค้ด รันเพื่อทดสอบการทำงาน ค้นหาและแก้ไขข้อผิดพลาด เผยแพร่โปรแกรม ช่วยให้ผู้พัฒนาโปรแกรมสามารถเขียนโปรแกรมด้วยความสะดวกสบายขึ้น สามารถแก้ไขข้อผิดพลาดในการเขียนโปรแกรมได้ง่าย รวดเร็วนอกจาก Visual Basic .NET แล้ว Visual Studio ยังมีภาษาอื่นๆ ให้ใช้เขียนโปรแกรมได้อีกเช่น ภาษา C# และภาษา C++



รูปที่ 2.5 โปรแกรม Microsoft Visual Studio 2010

โดยเครื่องคอมพิวเตอร์ที่จะติดตั้งและสามารถใช้งานโปรแกรม Visual Studio 2010ได้จะต้องมีคุณสมบัติตรงตามที่กำหนด ดังนี้

ระบบปฏิบัติการที่โปรแกรม Visual Studio 2010รองรับ

- Windows 8
- Windows 7 SP1
- Windows Server 2012 R2

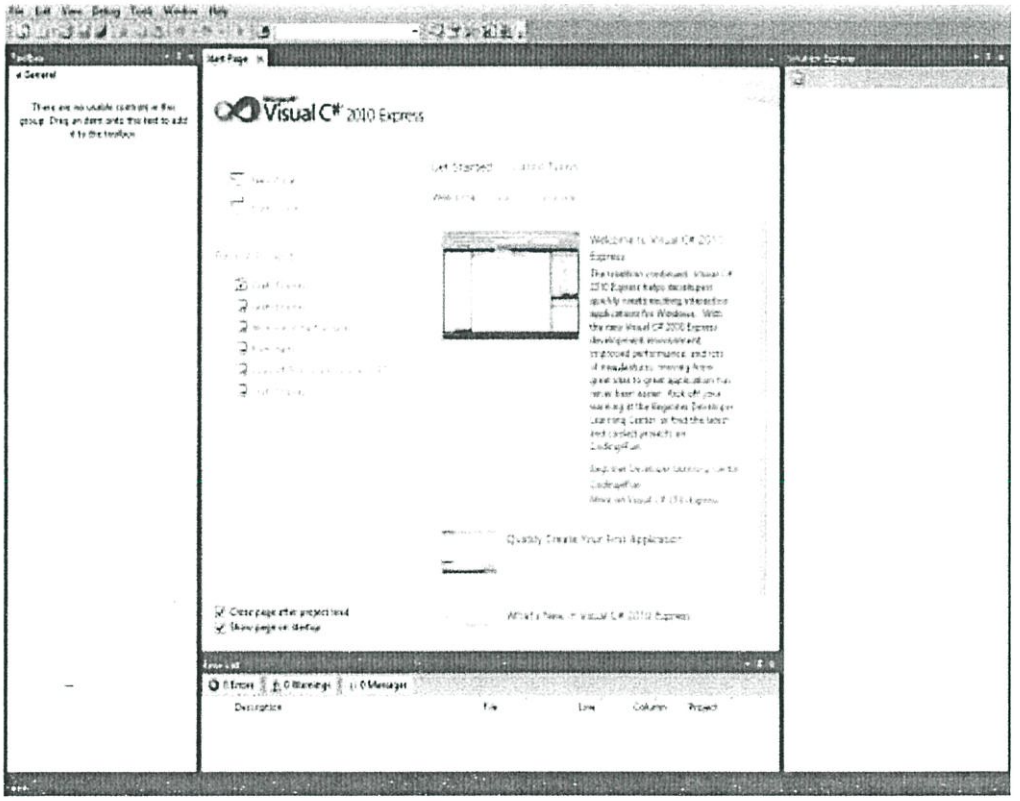
- Windows Server 2012
- Windows Server 2008 R2 SP1

ฮาร์ดแวร์ขั้นต่ำที่สามารถใช้งานร่วมกับโปรแกรม Visual Studio 2010ได้

- ซีพียูความเร็ว 1.6 GHz ขึ้นไป
- แรมไม่ต่ำกว่า 1 GB
- พื้นที่ในฮาร์ดดิสก์ไม่ต่ำกว่า 20 GB
- หน้าจอแสดงผลความละเอียดไม่ต่ำกว่า 1024 x 768 พิกเซล

การเริ่มต้นใช้งานโปรแกรม Visual Studio 2010

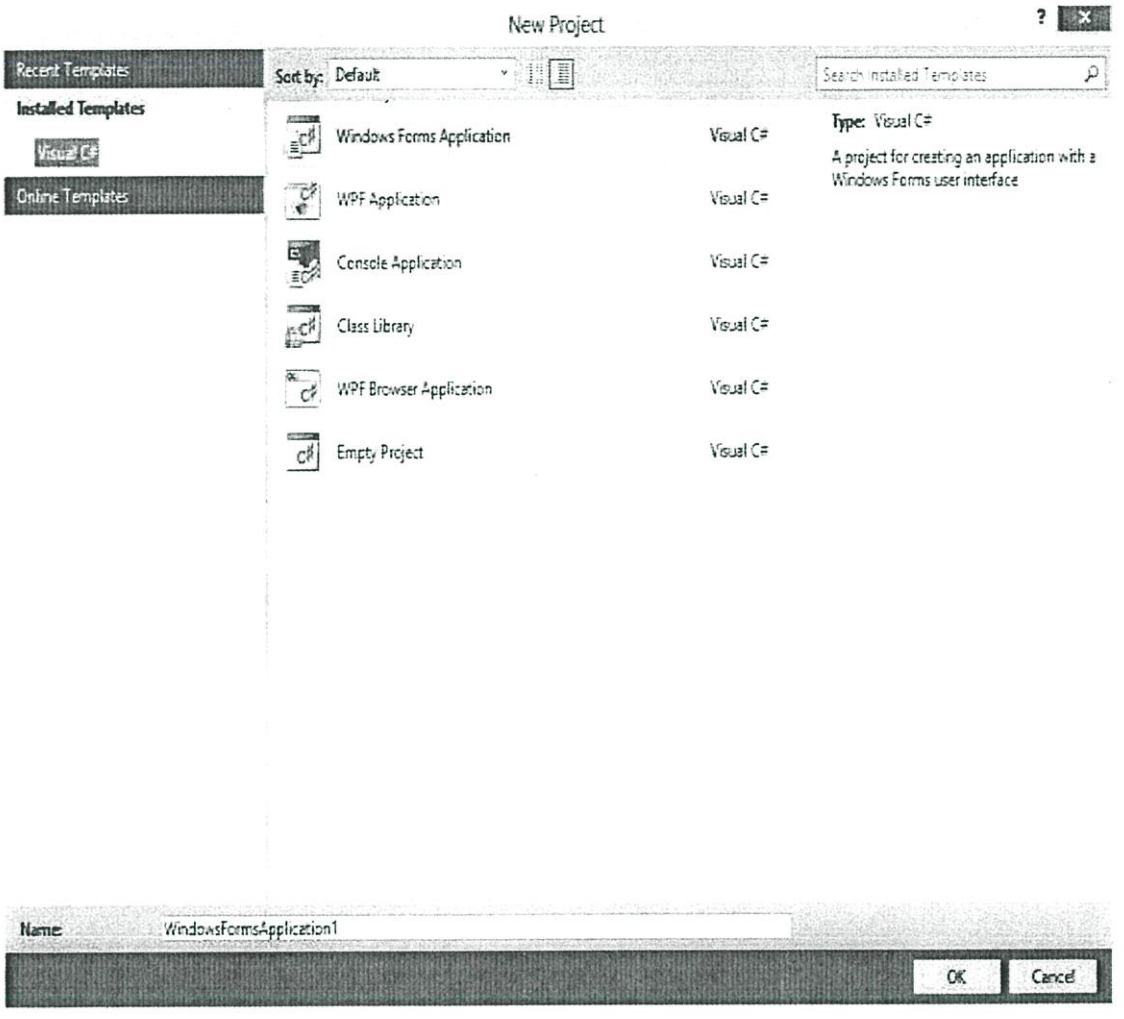
เมื่อติดตั้งโปรแกรม Visual Studio 2010เสร็จเรียบร้อยแล้ว สามารถเรียกใช้โปรแกรม โปรแกรมได้จากไอคอนบนหน้าเดสท็อปหรือปุ่ม Start Menu ของ Windows ได้ทันที ซึ่งหน้าโปรแกรมหลักจะเป็นดังภาพด้านล่าง



รูปที่ 2.6หน้าจอเริ่มต้นการใช้งานของโปรแกรม

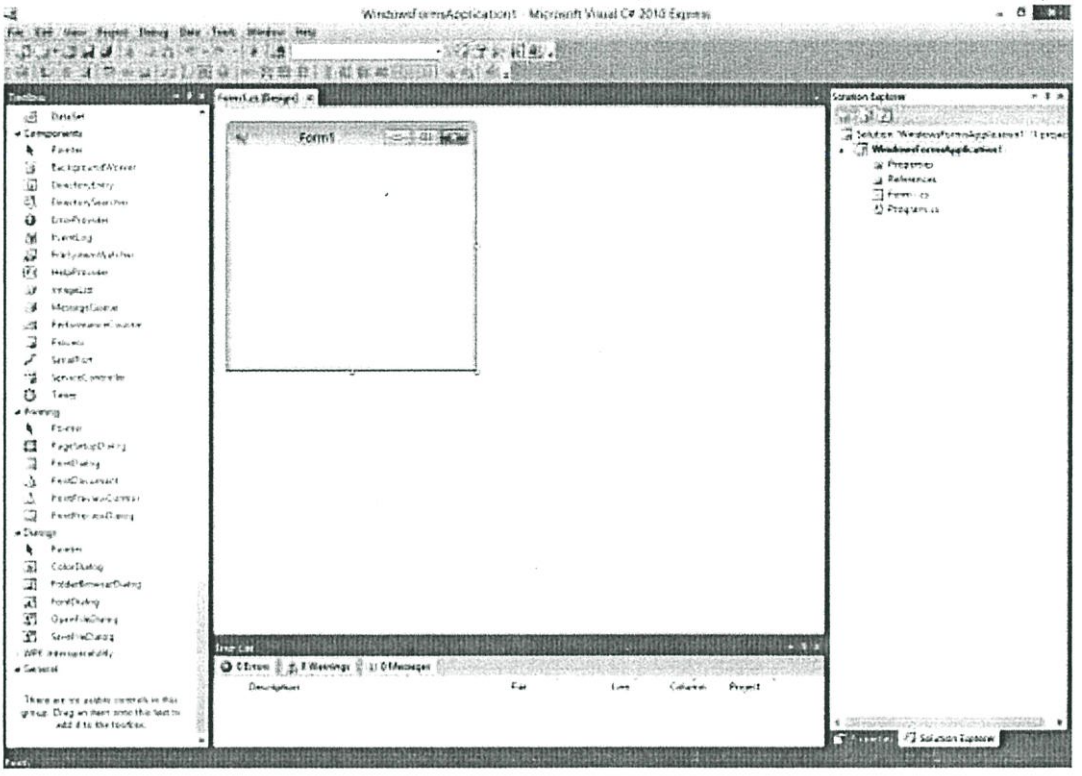
การสร้าง Project ใหม่

การเขียนโปรแกรมด้วย Visual Studio 2010จะต้องเป็นการเขียนโปรแกรมใน Project ซึ่งสามารถสร้าง Project ใหม่ได้ โดยคลิกที่เมนู File > New > Project จากนั้นจะพบกับหน้าต่าง ดังภาพ



รูปที่ 2.7 การสร้างโปรเจคใหม่

ในเมนูแถบด้านซ้าย ให้เลือกที่ Visual C# เพื่อกำหนดลักษณะของ Project ว่าเป็นการเขียนโปรแกรมด้วยภาษา C# ส่วนรายการตรงกลาง ให้เลือกที่ Windows Form Application เพื่อกำหนดว่าเป็นการสร้างโปรแกรมที่รันบนระบบปฏิบัติการ Windows จากนั้นกดปุ่ม OK

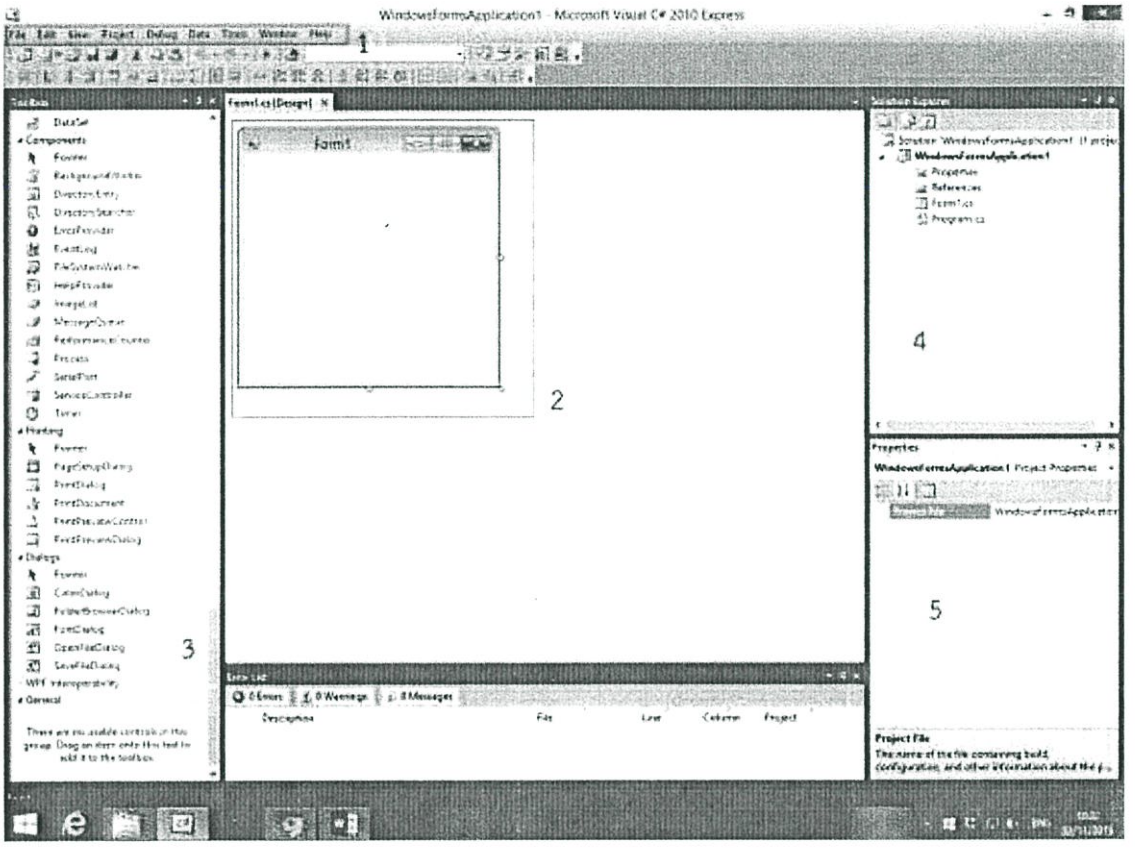


รูปที่ 2.8 หน้าจอ Windows Form Application

จะพบกับหน้าจอของ Windows Form Application ขึ้นมา สามารถเริ่มเขียนโปรแกรมได้ทันที

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ส่วนต่างๆ ของโปรแกรม Visual Studio 2010



รูปที่ 2.9 ส่วนต่าง ๆ ของโปรแกรม Visual Studio 2010

1. **แถบเมนู** (Menu bar) เป็นแถบแสดงรายการคำสั่งต่างๆ ของโปรแกรม เช่น การสร้างโปรเจ็คใหม่ การตั้งค่าโปนแกรม หรือการเรียกใช้เครื่องมืออื่นๆ เป็นต้น
2. **พื้นที่สำหรับเขียนโปรแกรม** (Workspace) เป็นพื้นที่สำหรับการออกแบบและเขียนคำสั่งต่างๆ ให้โปรแกรม
3. **กล่องเครื่องมือ** (Toolbox) เป็นกลุ่มของเครื่องมือต่างๆ สำหรับใช้ในการออกแบบหน้าจอของโปรแกรม เช่น Button, Listbox, Checkbox, Radio button Label เป็นต้น
4. **ส่วนควบคุมการทำงาน** (Solution Explorer) เป็นส่วนสำหรับแสดงโปรเจ็กหรือไฟล์ที่กำลังเปิดอยู่
5. **หน้าต่างแสดงคุณสมบัติของเครื่องมือ** (Properties) เป็นส่วนที่กำหนดคุณสมบัติต่างๆ ให้กับเครื่องมือแต่ละตัว เช่น การกำหนดสี กำหนดขนาดตัวอักษร และอื่นๆ

144474

2.12 การใช้โปรแกรม Microsoft Excel

โปรแกรมExcel เป็นโปรแกรมประเภท Spread sheetใช้สำหรับงานตาราง คำนวณ และนำเสนอข้อมูลตัวเลขด้วยกราฟแบบต่างๆ

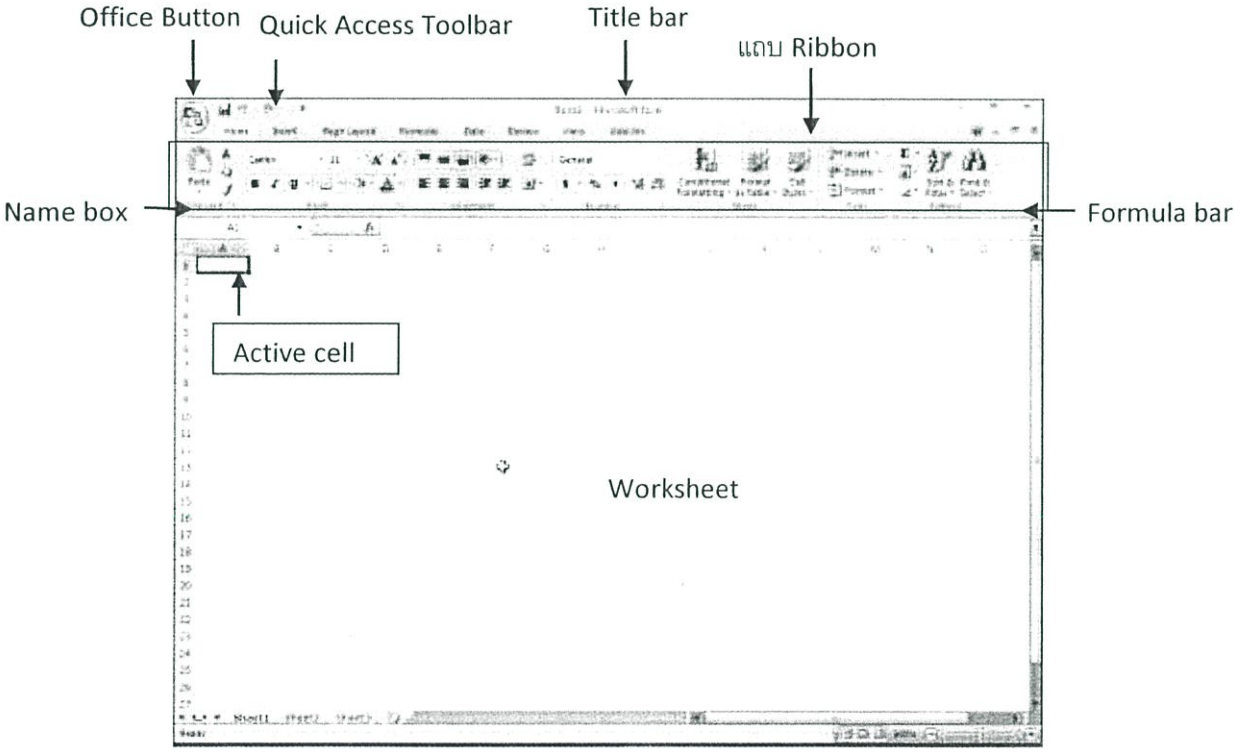
การเริ่มใช้งานโปรแกรม

เริ่มใช้งานโปรแกรม Microsoft Office Excel

1. คลิกปุ่ม Start บนแถบ Task bar
2. เลือก All Programs → Microsoft Office
3. เลือก Microsoft Office Excel 2007 จะเปิดให้ใช้งานได้ทันที

ส่วนประกอบของหน้าจอโปรแกรม

ก่อนที่จะทำงานกับโปรแกรม Microsoft Office Excel จะต้องรู้จักกับส่วนประกอบของหน้าจอโปรแกรมก่อน เพื่อจะได้เข้าใจถึงส่วนต่างๆ ได้ง่ายขึ้น



รูปที่ 2.10 ส่วนต่าง ๆ ของโปรแกรม Microsoft Excel

- Office Button เป็นปุ่มคำสั่งเกี่ยวกับการทำงานของแฟ้มงาน เช่น New, Open, Save, Save As, Print, Close, ฯลฯ
- Quick Access Toolbar เป็นแถบเครื่องมือให้เรียกใช้งานได้อย่างรวดเร็ว ผู้ใช้สามารถเพิ่มปุ่มคำสั่งที่ใช้งานบ่อยๆ ไว้ในแถบเครื่องมือนี้ได้

- Title bar แถบแสดงชื่อโปรแกรมและชื่อไฟล์ปัจจุบันที่คุณเปิดใช้งานอยู่
- แถบRibbon เป็นแถบที่รวบรวมคำสั่งต่างๆ ของเมนูหรือทูลบาร์ เพื่อให้ผู้ใช้เลือกใช้งานง่ายขึ้น
- Name box แสดงตำแหน่งเซลล์ปัจจุบัน
- Formula bar แถบแสดงสูตรคำนวณหรือข้อมูลที่พิมพ์
- Active cell เซลล์ปัจจุบันที่กำลังทำงานอยู่
- Worksheet พื้นที่ใช้งานของโปรแกรม

การป้อนข้อมูลมี 3 วิธี

1. พิมพ์ข้อมูลที่ต้องการ เสร็จแล้วกดปุ่ม Enter
2. พิมพ์ข้อมูลที่ต้องการ เสร็จแล้วกดปุ่ม Arrow key (ลูกศรสี่ทิศ) ที่คีย์บอร์ด
3. พิมพ์ข้อมูลที่ต้องการเสร็จแล้วคลิกปุ่ม Enter บน Formula bar

การยกเลิกข้อมูลที่กำลังพิมพ์ มี 2 วิธี

- วิธีที่ 1 กดปุ่ม Esc ที่คีย์บอร์ด
- วิธีที่ 2 คลิกปุ่ม Cancel บน Formula bar

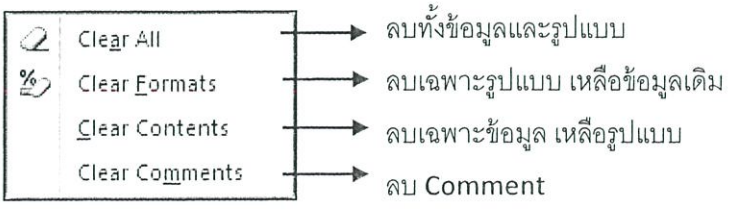
การแก้ไขข้อมูลมี 3 วิธี

1. เลือกตำแหน่งเซลล์ที่ต้องการแก้ไขข้อมูล
   วิธีที่ 1 กดปุ่ม F2 ที่คีย์บอร์ด
   วิธีที่ 2 คลิกตำแหน่งที่จะแก้ไขบน Formula bar
   วิธีที่ 3 ดับเบิ้ลคลิกที่ตำแหน่งเซลล์นั้น
2. ทำการแก้ไขข้อมูล เสร็จแล้วกด Enter

การลบข้อมูล

มี 3 วิธี

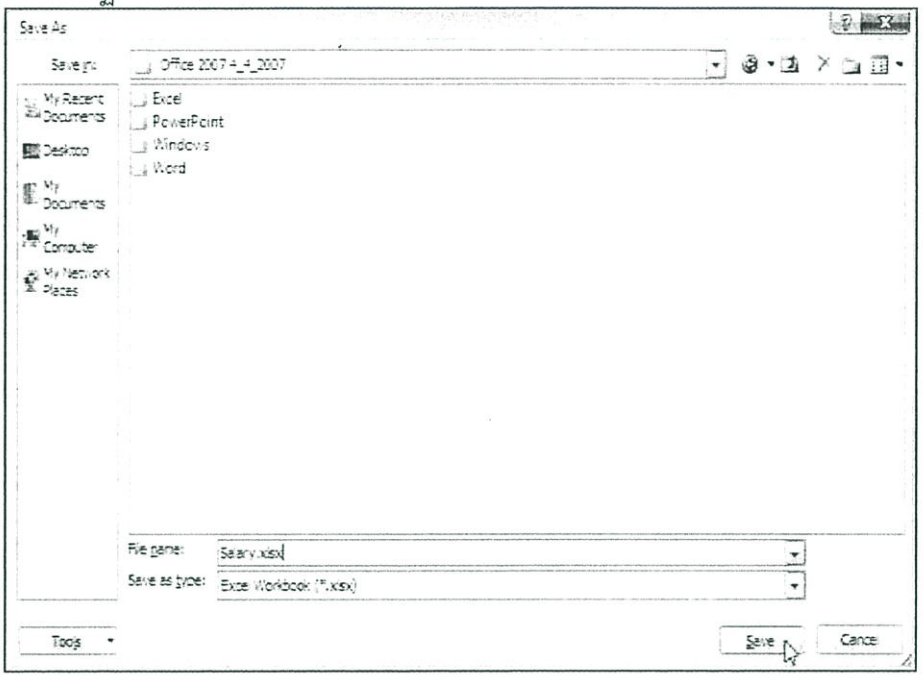
- เลือกตำแหน่งเซลล์ที่ต้องการลบข้อมูล
  วิธีที่ 1 กดปุ่ม Delete ที่คีย์บอร์ด
  วิธีที่ 2 คลิกขวาเลือกคำสั่ง Clear Contents
  วิธีที่ 3 ที่แท็บ Home คลิกลูกศรลงของปุ่ม Clear จะปรากฏคำสั่งให้เลือกใช้



การบันทึกเอกสาร

เมื่อสร้างแฟ้มงานเสร็จแล้ว ต้องทำการบันทึกข้อมูลลงใน disk มีขั้นตอนดังนี้

1. คลิกปุ่ม Save บนแถบ Quick Access Toolbar หรือคลิกปุ่ม Office Button เลือกคำสั่งSave หรือ Save As ก็ได้
2. จะปรากฏไดอะล็อกบ็อกซ์ของ Save As ให้กำหนดรายละเอียด



รูปที่ 2.11 หน้าต่างบันทึกไฟล์

3. ที่ช่อง Save in เลือกตำแหน่งไดร์ฟและโฟลเดอร์ที่ต้องการเก็บข้อมูล
4. ที่ช่อง File name พิมพ์ชื่อไฟล์ →คลิกปุ่ม Save จะได้ไฟล์นามสกุล .xlsx

การสร้างกราฟ

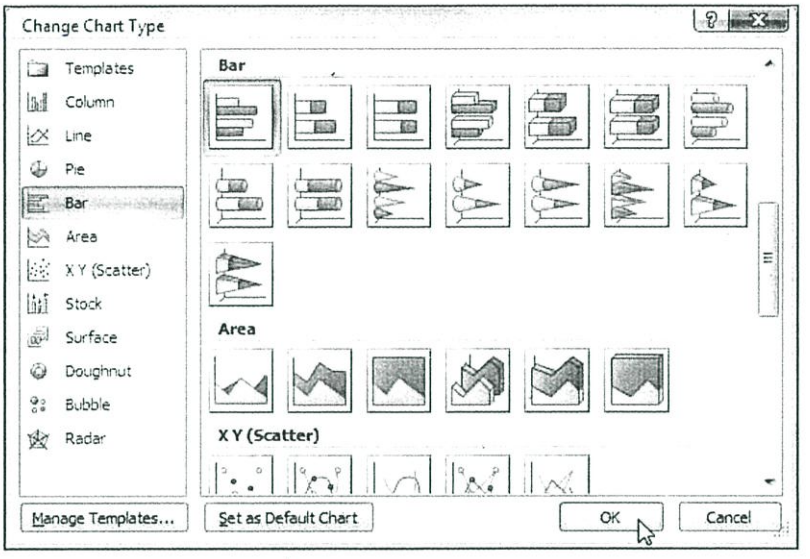
ข้อมูลในโปรแกรม Excel ส่วนใหญ่จะเป็นข้อมูลเกี่ยวกับตัวเลข เมื่อนำตัวเลขเหล่านั้นมาแสดงเป็นกราฟจะทำให้ข้อมูลดูง่ายขึ้น มีขั้นตอนดังนี้

1. เลือกช่วงข้อมูลที่ต้องการนำมาสร้างกราฟ
2. คลิกแท็บ Insert เลือกประเภทของกราฟจากหัวข้อ Chart
3. จะปรากฏรูปแบบย่อยให้คลิกเลือกได้ทันที
4. จากนั้น จะแสดงรูปกราฟที่สร้างไว้ พร้อมกับแถบ Ribbon ชื่อ Chart Tools หัวข้อ Design ให้กำหนดรายละเอียดเพิ่มเติม

การเปลี่ยนชนิดของกราฟ

หลังจากสร้างกราฟเสร็จแล้ว ต้องการเปลี่ยนแบบกราฟใหม่ สามารถทำได้ มีขั้นตอนดังนี้

1. เลือกกราฟที่ต้องการเปลี่ยนรูปแบบ
2. แถบ Chart Tools → Design หัวข้อ Type
3. จะปรากฏไดอะล็อกบ็อกซ์ Change Chart Type ให้คลิกเลือกแบบที่ต้องการ



รูปที่ 2.12 รูปแบบกราฟชนิดต่าง ๆ

2.13 การใช้โปรแกรม Matlab

รูปที่ 2.13 โปรแกรม Matlab

Matlab (matrix laboratory) เป็นภาษาคอมพิวเตอร์ชั้นสูง (High-level Language) สำหรับการคำนวณทางเทคนิคที่ประกอบด้วยการคำนวณเชิงตัวเลข กราฟิกที่ซับซ้อน และการจำลองแบบเพื่อให้มองเห็นภาพพจน์ได้ง่ายและชัดเจน เดิมโปรแกรม MATLAB ได้เขียนขึ้นเพื่อใช้ในการคำนวณทาง matrix หรือเป็น matrix software ที่พัฒนาจากโพรเจ็กที่ชื่อ LINKPACK และ EISPACK

ต่อมาได้พัฒนามาด้วยการแก้ปัญหาที่ส่งมาจากหลายๆ ผู้ใช้เป็นระยะเวลาหลายปีจึงทำให้โปรแกรม MATLAB มีฟังก์ชันต่างๆ ให้เลือกใช้มากมาย ในบางมหาวิทยาลัยได้ใช้โปรแกรม MATLAB เป็นหลักสูตรพื้นฐานในการศึกษาทางด้วนคณิตศาสตร์ วิศวกรรม และวิทยาศาสตร์แขนงต่างตลอดจนใจด้วนอุตสาหกรรมได้ใช้โปรแกรม MATLAB เป็นเครื่องมือสำหรับใช้ในงานวิจัย พัฒนาและวิเคราะห์ โปรแกรม MATLAB จะมีกล่องเครื่องมือที่ใช้ในการหาคำตอบเรียกว่า Toolbox โดยโปรแกรม MATLAB จะมี toolbox ในแต่ละสาขา เช่น การประมวลผลสัญญาณ (Signal processing toolbox) การประมวลผลภาพ (image processing toolbox) ระบบควบคุม (control system toolbox) โครงข่ายประสาท (neural networks toolbox) ฟัซซี่ลอจิก (fuzzy logic toolbox) เวฟเลท (wavelet toolbox) การติดต่อสื่อสาร (communication toolbox) สถิติ (Statistics toolbox) และสาขาอื่นๆ มากมาย ภายใน toolbox แต่ละสาขาก็จะมีฟังก์ชันต่างๆ ที่เกี่ยวข้องกับการแก้ปัญหาในสาขานั้นๆ ให้เลือกประยุกต์ใช้งานเป็นจำนวนมาก

ข้อดีของโปรแกรม matlab

1. มีฟังก์ชันคณิตศาสตร์ให้เลือกใช้ในการคำนวณมากมายตลอดจนเราสามารถสร้างฟังก์ชันขึ้นมาใช้งานได้เองในสาขาที่ต้องการ โดยฟังก์ชันที่สร้างขึ้น (M-File) จะมีนามสกุลเป็น .M
2. Algorithm พัฒนาได้ง่ายไม่ยุ่งยาก สามารถแก้ไขปัญหาทางด้วนคณิตศาสตร์ที่มีความซับซ้อนได้ง่าย และรวดเร็วกว่าโปรแกรมภาษาอื่นๆ เช่น C Fortran Basic เป็นต้น
3. มีโครงสร้างแบบจำลอง (Simulink) ซึ่งเป็น Package ที่เรานำไปสร้างบล็อกไดอะแกรมเพื่อใช้ทดสอบ และประเมินผลระบบ Dynamic ต่างๆ ก่อนนำไปใช้งานจริง
4. สามารถวิเคราะห์และตรวจสอบข้อมูลได้ง่ายและรวดเร็ว

5. นำไปใช้งานในทางด้วนกราฟิกได้เป็นอย่างดีทั้งในด้านการแสดงภาพตั้งแต่สองมิติที่เป็น rectangular polar stair bar รวมทั้งภาพสมมิติในรูปแบบพื้นผิว (surface) และระดับสูงต่ำ (contour) ตลอดจนสามารถนำภาพมาต่อกัน และเก็บไว้เพื่อที่จะสร้างเป็นภาพเคลื่อนไหวได้อีกด้วย
6. ประยุกต์ใช้ในการสร้างรูปแบบ Graphical User Interface ได้โดยการเลือกใช้ object และเมนูต่างๆ โดยโปรแกรม MATLAB จะมีเครื่องมือให้เลือกใช้ เช่น เมนู รายการ ปุ่มกด และ fields object ต่างๆ เพื่อให้ผู้ใช้สามารถเลือกนำไปใช้ในการทำงานปฏิสัมพันธ์กันระหว่างผู้ใช้กับเครื่องคอมพิวเตอร์ได้
7. ทำการประมวลผลร่วมกับโปรแกรมอื่นได้ เช่น Fortran, Borland C/C++, Microsoft Visual C++ และ Watcom C/C++ ด้วยการเขียนฟังก์ชันที่เป็น mex ไฟล์โดยโปรแกรม MATLABจะเรียกใช้รูทีนจากโปรแกรมภาษา C และ Fortran
8. โปรแกรม MATLAB เป็นระบบ interactive ซึ่งส่วนของข้อมูลพื้นฐานเป็นอาร์เรย์ที่ไม่ต้องการมิติ ทำให้โปรแกรม MATLAB สามารถทำการแก้ปัญหาทางเทคนิคต่างๆ ได้มากใช้เวลาในการประมวลผลน้อย และดีกว่าโปรแกรมภาษา C และ Fortran

ข้อเสียของโปรแกรม matlab

1. การป้อนข้อมูลแบบ spreadsheet จะทำได้ลำบากมาก

โครงสร้างของ MATLAB

MATLAB ประกอบด้วย 5 ส่วนใหญ่ คือ

1. ภาษาโปรแกรม MATLAB (The MATLAB language)MATLAB เป็นโปรแกรมภาษาชั้นสูงที่ใช้ควบคุม flow statement ฟังก์ชัน โครงสร้างข้อมูลอินพุท/เอาท์พุท และลักษณะโปรแกรม Object-Oriented Programming ทำให้การเขียนโปรแกรมไม่ยุ่งยากเมื่อเทียบกับการเขียนโปรแกรมด้วยภาษาอื่นๆ เช่น C, Fortran, Basic เป็นต้น
2. สถาปัตยกรรมในการทำงานของ MATLAB (The MATLAB working environment) MATLAB จะมีกลุ่มของเครื่องมือที่เป็นประโยชน์สำหรับการทำงานของผู้ใช้โปรแกรม หรือโปรแกรมเมอร์ประโยชน์ที่กล่าวนี้ก็คือการจัดการตัวแปรใน workspace การนำข้อมูลหรือการผ่านค่าตัวแปรเข้า/ออกและกลุ่มของเครื่องมือต่างๆ นี้ก็จะใช้สำหรับพัฒนา จัดการตรวจสอบความผิดพลาดของโปรแกรม (debugging) ที่ได้เขียนขึ้น
3. ฟังก์ชันในการคำนวณทางคณิตศาสตร์ (The MATLAB mathematical function library) MATLAB จะมีไลบรารีทั่วไปที่ใช้ในการคำนวณอย่างกว้าง เช่น sine, cosine และพีชคณิตเชิงซ้อนโดยสามารถนำไปประยุกต์ใช้เป็นฟังก์ชันหรือไลบรารีเพิ่มเติมขึ้นจากไลบรารีที่ใช้กันโดยทั่วไป เช่น ฟังก์ชันในการหา eigenvalues และ eigenvectors การแยกตัวประกอบและส่วนประกอบของเมตริกซ์ด้วยวิธีต่างๆ การวิเคราะห์ข้อมูล การหาความน่าจะเป็น และการแก้ปัญหาระบบของสมการเชิงเส้นที่เป็นพื้นฐานของสาขาวิชาต่างๆ เป็นต้น ทำให้โปรแกรม MATLAB มีฟังก์ชันสำหรับใช้งานค่อนข้างมากและครอบคลุมในรายละเอียดของการคำนวณสาขาต่างๆ ได้มากขึ้น

4. Handle Graphics
ระบบกราฟิกของ MATLAB จะประกอบด้วยคำสั่งชั้นสูงสำหรับการพล็อตกราฟโดยมีพื้นฐานอยู่บนแนวความคิดที่ว่าทุกๆ สิ่งบนหน้าต่างรูปภาพของโปรแกรม MATLAB จะเป็นวัตถุ (Object) ซึ่งมีเอกลักษณ์เฉพาะตัว Handle Graphics ประกอบด้วยคำสั่งชั้นสูงให้คุณได้เลือกใช้ในการสร้าง Graphic User Interface บนพื้นฐานการประยุกต์ใช้งานของคุณ นอกจากนี้โปรแกรม MATLAB ยังมีฟังก์ชันที่ใช้สำหรับการแสดงภาพสองมิติ ภาพสามมิติ และการสร้างภาพเคลื่อนไหว
5. The MATLAB Application Program Interface (API) API จะใช้เพื่อสนับสนุนการติดต่อจากภายนอกโดยใช้โปรแกรมที่เป็น mex ไฟล์ซึ่งเป็นไฟล์ซึ่งเป็นไฟล์ที่เขียนขึ้นโดยใช้ mex ฟังก์ชันใน MATLAB ซึ่งจะเรียกใช้รูทีนจากโปรแกรมภาษา C และ Fortran หรืออาจกล่าวได้ว่า API เป็นไลบรารีที่เขียนด้วยโปรแกรมภาษา C และ Fortran ที่มีการเชื่อมต่อกับโปรแกรม MATLAB ด้วยไฟล์ที่เป็น mex ฟังก์ชันอีกทั้ง MATLAB API นี้ยังมีความสามารถสำหรับการเรียก routine จาก MATLAB (dynamic linking) ก็ได้

นอกจากลักษณะเด่นของโปรแกรม MATLAB ทั้ง 5 ข้อที่ได้กล่าวมาข้างต้นแล้วโปรแกรม MATLAB ยังมีเครื่องมือที่ใช้สำหรับการวิเคราะห์และทดสอบระบบโดยการจำลองขึ้นมาซึ่งก็คือ Simulink

Simulink เป็นโปรแกรมที่ควบคู่กับ MATLAB ซึ่งเป็นระบบ Interactive สำหรับการจำลองและวิเคราะห์ระบบไดนามิกต่างๆ ที่เป็นระบบเชิงเส้น (Linear) ระบบไม่เชิงเส้น (nonlinear) Simulink เป็นโปรแกรม mouse-driver ที่ให้คุณใช้ระบบโมเดลโดยการวาดบล็อกไดละแกรมบนจอภาพด้วยการใช้เมาส์ทำให้โปรแกรม MATLAB สามารถทำการจำลองระบบได้หลายรูปแบบ น เชิงเส้น (linear) ไม่เชิงเส้น (nonlinear) เวลาต่อเนื่อง (continuous-times) เวลาไม่ต่อเนื่อง (discrete-time) และระบบเหลายอัตรา (multirate) ซึ่งแต่ละรูปแบบที่นำมาสร้างแบบจำลองในการวิเคราะห์นี้ผู้ใช้จะต้องมีความเข้าใจพื้นฐานการทำงานของบล็อกแต่ละบล็อกได้เป็นอย่างดี ตลอดจนเข้าใจระบบโดยรวมของงานที่จะกระทำด้วย

Blocksets เป็นสิ่งที่เพิ่มเติมใน Simulink โดยจะเป็นไลบรารีของบล็อกสำหรับการประยุกต์เฉพาะ เช่น การติดต่อสื่อสาร (Communications) การประมวลผลข้อมูล (Signal processing) และระบบไฟฟ้ากำลัง (power systems)

Real-time Workshop เป็นโปรแกรมที่ให้คุณสร้าง c code จากบ็อกไดอะแกรมของคุณและสามารถกระทำกับบล็อกไดอะแกรมได้หลากหลายด้วยระบบเวลาจริง (real-time systems)

โปรแกรม MATLAB มีอยู่หลาย version ซึ่ง version ดั้งเดิมของโปรแกรม MATLAB จะใช้งานบน DOS ที่มีการคำนวณไม่ยุ่งยากเหมาะสำหรับผู้เริ่มศึกษา คอมพิวเตอร์ที่ใช้ก็ไม่จำเป็นต้องมีพื้นที่หน่วยความจำมาก ใช้ได้กับ CPU ที่มีความเร็วต่ำ แต่มีข้อเสียคือฟังก์ชันที่นำมาใช้งานมีน้อยทำให้เขียนโปรแกรมที่มีความซับซ้อนได้ไม่ดีเท่าที่ควรเพราะมีประสิทธิภาพและความเร็วในการประมวลผลต่ำ ต่อมาเมื่อระบบเลือกใช้ได้มากมายจึงทำให้โปรแกรม MATLAB มีประสิทธิภาพและมีความสามารถในการประมวลผลที่เร็วขึ้น version ใหม่ที่ได้ทำการปรับปรุงใหม่ให้ดีขึ้นนี้จะใช้งานบน Windows ทำให้ผู้ใช้มีความสะดวกในการใช้งานมากขึ้น ข้อดีของ version ใหม่นี้คือมีประสิทธิภาพที่

ดีขึ้น การประมวลผลโปรแกรมที่ซับซ้อนมีความเร็วสูงขึ้น และมีฟังก์ชันต่างๆ ให้เลือกใช้ในการสาขาต่างๆ มากมาย แต่ก็ต้องใช้กับคอมพิวเตอร์ที่มีพื้นที่หน่วยความจำมาก CPU มีความเร็วสูง และต้องการ Co-processor ในการช่วยคำนวณ

คอมพิวเตอร์ที่เหมาะสมสำหรับโปรแกรมMATLAB

เนื่องจากโปรแกรม MATLAB เป็นโปรแกรมที่ใช้สำหรับการคำนวณทางคณิตศาสตร์และกราฟิกที่ซับซ้อนดังนั้นจึงจำเป็นต้องใช้เครื่องคอมพิวเตอร์ที่มีความเร็วสูง คอมพิวเตอร์พีซีที่เหมาะสมกับโปรแกรม MATLAB คือ มีซีพียูรุ่นเพนเทียมขึ้นไป แรมควรมีอย่างต่ำ 32 เมกกะไบต์ ส่วนฮาร์ดดิสก์ควรมีเนื้อที่ว่างเกินกว่า 80 เมกกะไบต์ขึ้นไป

ตัวดำเนินการเครื่องหมายพิเศษและตัวดำเนินการเปรียบเทียบและตรรกะ

ลักษณะการดำเนินการ	ตัวดำเนินการ	รูปแบบทางคณิตศาสตร์	รูปแบบทาง MATLAB
การบวก	+	a+b	a+b
การลบ	-	a-b	a-b
การคูณ	*	axb	a*b
การคูณเชิงสมาชิก	.*	-	a.*b
การหารทางขวา	/	a/b	a/b
การหารทางซ้าย	\	b/a	a\b
การหารเชิงสมาชิก	./	-	a./b
การยกกำลัง	Ab	ab	a^b
การยกกำลังเชิงสมาชิก	A.b	-	a.^b

ตารางที่ 2.2 **ตัวดำเนินการเครื่องหมายพิเศษการคำนวณ**

ในทาง MATLAB คือการนำเอา Matrix a และ Matrix bมา Operate กัน เช่น + ,-,*,/,หรือยกกำลังกัน เพราะฉะนั้นเงื่อนไขการ Operate ต่างๆ จึงเป็นวิธีการทาง Matrix ส่วนเครื่องหมายที่มีจุด (.) แสดงว่าจะกระทำเฉพาะแถวกับหลักเดียวกัน

เช่น

a = 1 2 3
    4 5 6

b = 2 2 2
    2 2 2

a.^b = 1 4 9
       16 25 36

ตัวดำเนินการเปรียบเทียบและตรรกะลักษณะดำเนินการ ตัวดำเนินการเปรียบเทียบและตรรกะ ตัวอย่าง

ลักษณะดำเนินการ	ตัวดำเนินการเปรียบเทียบและตรรกะ	ตัวอย่าง
น้อยกว่า	<	x<10
น้อยกว่าหรือเท่ากับ	<=	x<=10
มากกว่า	>	x>10
มากกว่าหรือเท่ากับ	>=	x>=10
เท่ากับ	==	x==1
ไม่เท่ากับ	~=	x~=5
และ	&	x>2 & y<1
หรือ	|	x>2 | y<1
ไม่	~	~x

ตารางที่ 2.3 **ตัวดำเนินการเปรียบเทียบและตรรกะ**ลักษณะดำเนินการ

ฟังก์ชั่นที่ใช้ในการหาค่าทางตรีโกณมิติ

คำสั่ง	รายละเอียด
sin(x)	ใช้คำนวณหาค่า Sine ของ x
sinh(x)	ใช้คำนวณหาค่า Hyperbolic sine.ของ x
asin(x)	ใช้คำนวณหาค่า Inverse sine. ของ x
asinh(x)	ใช้คำนวณหาค่า Inverse hyperbolic sine. ของ x
cos(x)	ใช้คำนวณหาค่า cos ของ x
cosh(x)	ใช้คำนวณหาค่า Hyperbolic cos.ของ x
acos(x)	ใช้คำนวณหาค่า Inverse cos ของ x
acosh(x)	ใช้คำนวณหาค่า Inverse hyperbolic cos. ของ x
tan(x)	ใช้คำนวณหาค่า Tangent ของ x
tanh(x)	ใช้คำนวณหาค่า Hyperbolic Tangent.ของ x
atan(x)	ใช้คำนวณหาค่า Inverse Tangent. ของ x
atan2(x)	ใช้คำนวณหาค่าFour quadrant inverse tangent ของ x
atanh(x)	ใช้คำนวณหาค่า Inverse hyperbolic Tangent ของ x
sec(x)	ใช้คำนวณหาค่า Secant. ของ x
sech(x)	ใช้คำนวณหาค่า Hyperbolic Secant.ของ x
asec(x)	ใช้คำนวณหาค่า Inverse Secant. ของ x
asech (x)	ใช้คำนวณหาค่า Inverse hyperbolic Secant. ของ x
csc(x)	ใช้คำนวณหาค่า Cosecant ของ x
csch(x)	ใช้คำนวณหาค่า Hyperbolic Cosecant.ของ x
acsc(x)	ใช้คำนวณหาค่า Inverse Cosecant. ของ x
acsch(x)	ใช้คำนวณหาค่า Inverse hyperbolic Cosecant ของ x
cot(x)	ใช้คำนวณหาค่า Cotangent ของ x
coth(x)	ใช้คำนวณหาค่า Hyperbolic Cotangent.ของ x
acot (x)	ใช้คำนวณหาค่า Inverse Cotangent. ของ x
acoth(x)	ใช้คำนวณหาค่า Inverse hyperbolic Cotangent ของ x

ตารางที่ 2.4 **ฟังก์ชั่นที่ใช้ในการหาค่าทางตรีโกณมิติ**

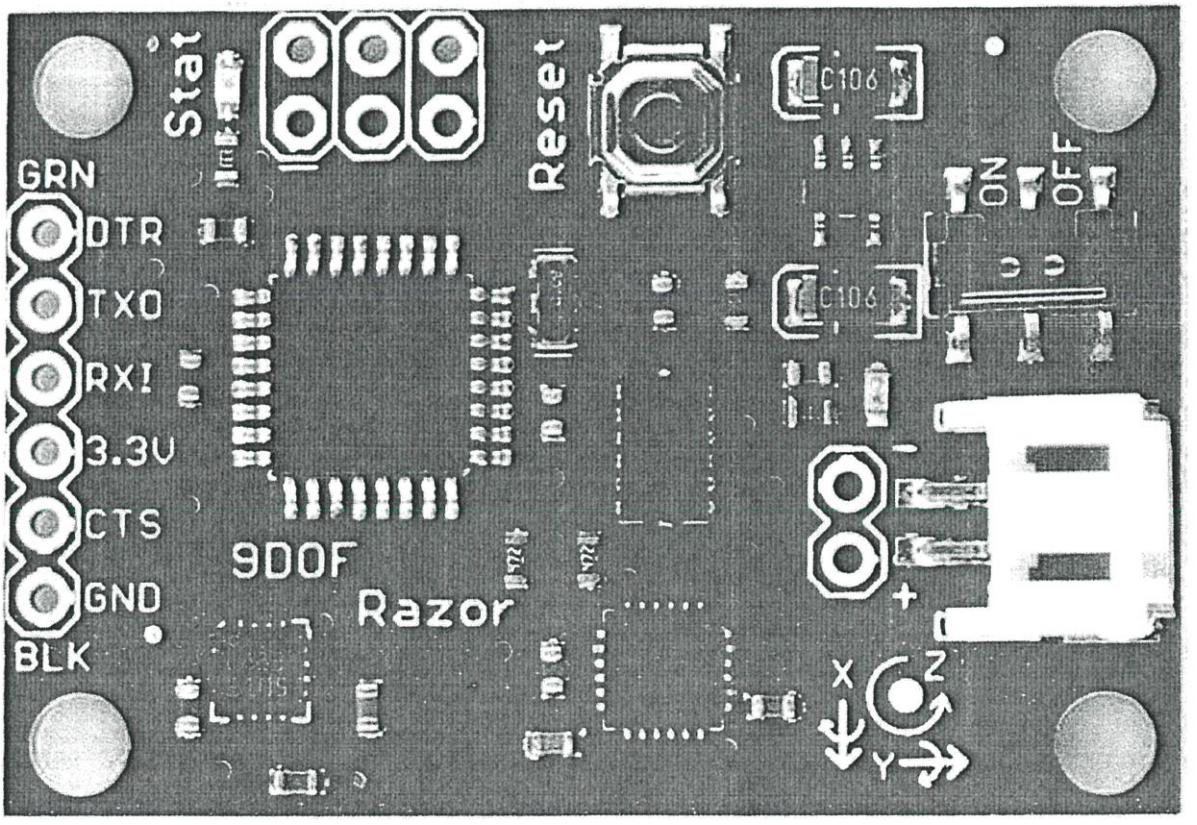
ฟังก์ชั่นที่ใช้ในการหาค่า Exponential.

คำสั่ง	รายละเอียด
exp(x)	ใช้คำนวณหาค่า Exponential. ของ x
log(x)	ใช้คำนวณหาค่า Natural logarithm.ของ x
log10(x)	ใช้คำนวณหาค่า Common (base 10) logarithm. ของ x
log2(x)	ใช้คำนวณหาค่า Base 2 logarithm and dissect floating point number. ของx
pow2(x)	ใช้คำนวณหาค่า เลขยกกำลังที่มี 2 เป็นฐาน ของ x
nextpow2(x)	ใช้คำนวณหาค่า ตัวเลขยกกำลังที่มี 2 เป็นฐานที่มีค่าเท่ากับหรือมากกว่า ของ x
abs(x)	ใช้คำนวณหาค่า Absolute ของ x
ceil(x)	ใช้คำนวณหาค่าจำนวนเต็มใดๆที่ใกล้เคียงกับค่าของ x โดยให้ค่านั้นเข้าใกล้ ? มาก
floor(x)	ที่สุด
round(x)	ใช้คำนวณหาค่าจำนวนเต็มใดๆที่ใกล้เคียงกับค่าของ x โดยให้ค่านั้นเข้าใกล้-? มาก
sqrt (x)	ที่สุด
rem(x,y)	ใช้คำนวณหาค่าจำนวนเต็มใดๆที่ใกล้เคียงกับค่าของ x โดยถ้าป็นทศนิยมเกิน 0.5
sign(x)	ปัดขึ้น
	ใช้คำนวณหาค่า Square root. ของ x
	ใช้คำนวณหาค่าเศษที่เหลือจากการหารกันระหว่าง x และ y
	ใช้สำหรับกำหนดค่าของ x ใดๆ ให้มีค่าเป็น -1,0,1
	ถ้าเป็น -1 แสดงว่าค่าของ x > 0
	ถ้าเป็น 0 แสดงว่าค่าของ x = 0
	ถ้าเป็น 1 แสดงว่าค่าของ x < 0

ตารางที่ 2.5 ฟังก์ชั่นที่ใช้ในการหาค่า Exponential

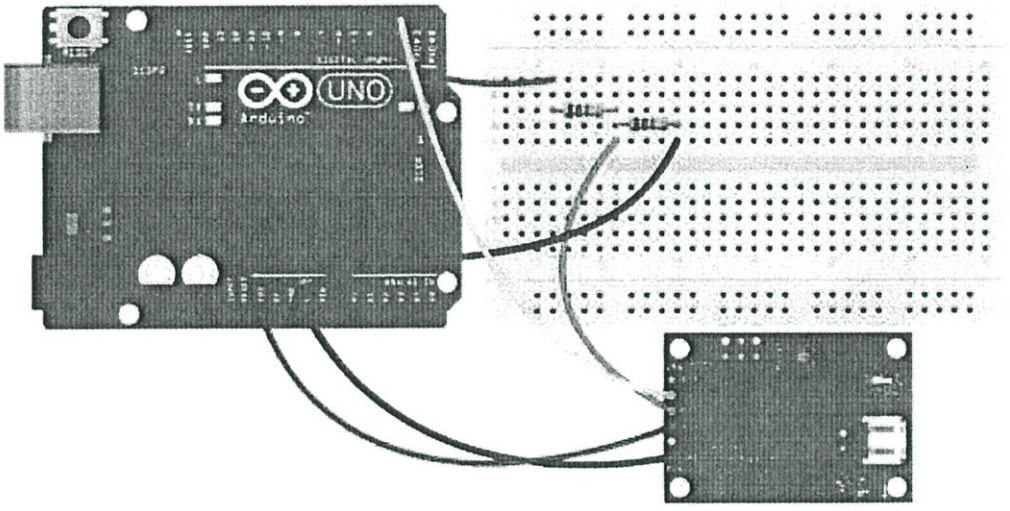
2.14 หน่วยวัดแรงเฉื่อย

หน่วยวัดแรงเฉื่อย (Inertial Measurement Unit หรือ IMU) เป็นอุปกรณ์อิเล็กทรอนิคส์ที่ใช้วัดและรายงานผลของแรงที่กระทำ อัตราเชิงมุม และสนามแม่เหล็ก ที่มีต่อวัตถุนั้นๆ มักใช้ในการสร้างเครื่องบินรวมถึงอากาศยานอื่นๆ IMU มีส่วนประกอบหลักที่สำคัญ คือ เครื่องวัดความเร่ง ซึ่งวัดทั้งความเร่งเชิงมุมและความเร่งเชิงเส้น (สำหรับการเปลี่ยนแปลงตำแหน่ง) และ ไจโรสโคป (เพื่อรักษาให้อยู่ในแนวอ้างอิงที่ถูกต้อง) ซึ่งโดยปกติจะต้องมี sensor ตรวจวัดอย่างน้อยหนึ่งตัวในแต่ละแนวแกน ลักษณะการออกแบบโดยทั่วไป IMU จะมีลักษณะเป็นกล่อง ภายในบรรจุเครื่องวัดความเร่ง 3 ตัว และ ไจโรสโคป อีก 3 ตัว เครื่องวัดความเร่งจะถูกติดตั้งเข้าไปในแต่ละแนวแกนที่ต้องการวัด โดยที่แต่ละแนวแกนตั้งฉากกัน ไจโรสโคปอีก 3 ตัวจะถูกติดตั้งในลักษณะตั้งฉากเช่นเดียวกันจะทำการวัดอาการหมุนที่เบี่ยงเบนไปจากแนวอ้างอิงในระบบพิกัดที่กำหนดไว้ จะเห็นได้ว่า เซนเซอร์ตรวจวัดหลักๆของระบบ IMU เซนเซอร์สำหรับตรวจวัดความเร่ง และ เซนเซอร์สำหรับ ตรวจวัดอาการเอียงในแนวแกนต่างๆ ซึ่งก็คือ เครื่องวัดความเร่ง (Accelerometer) ไจโรสโคป (Gyroscope)และเครื่องวัดสนามแม่เหล็ก (Magnetometer)



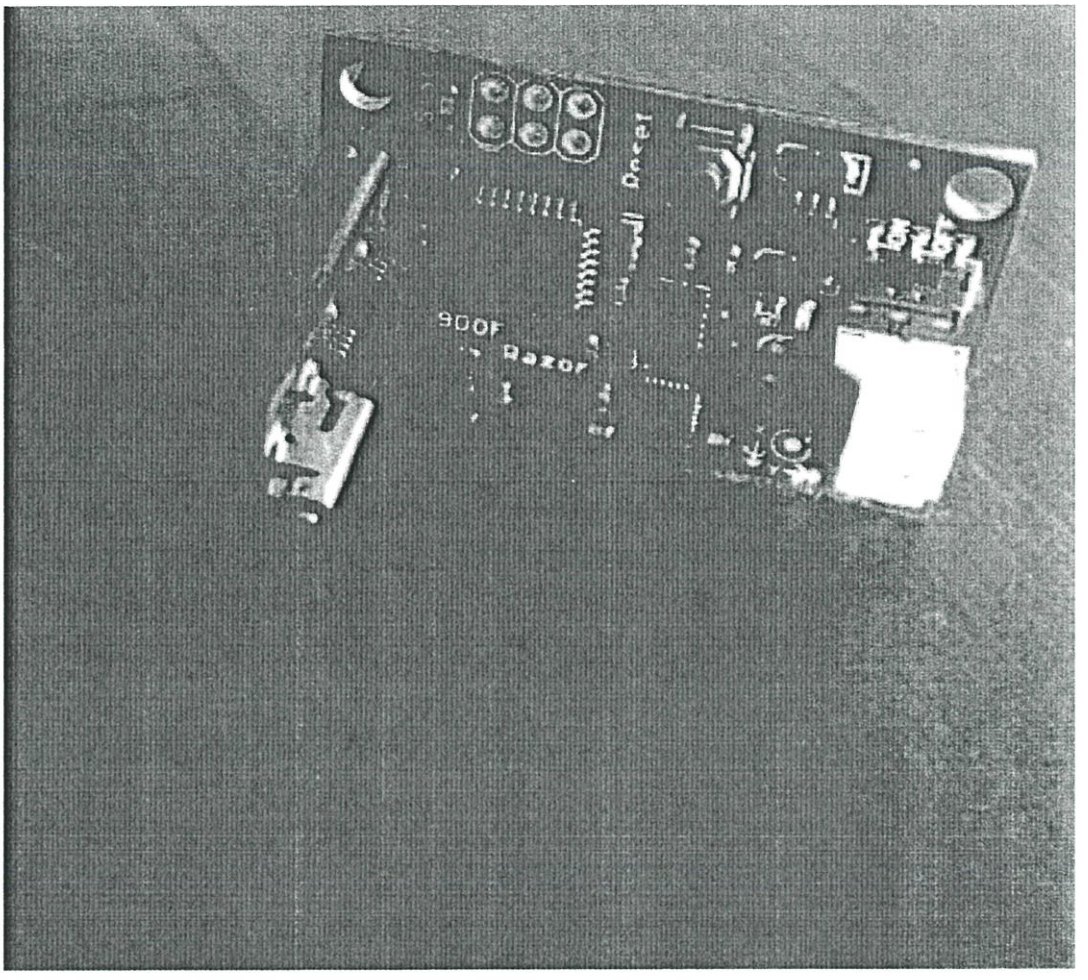
รูปที่ 2.14 9DoF Razor IMU

ในวิทยานิพนธ์ฉบับนี้ผู้จัดทำได้เลือกใช้ 9 Degree of freedom Razor IMU ในการทำวิจัย โดยประกอบไปด้วยความเร่ง 3 แกน ความเร็วเชิงมุม 3 แกน และสนามแม่เหล็ก 3 แกน นอกจากนั้น ยังสามารถต่อเข้ากับบอร์ด Arduino ได้โดยผ่าน library มาตรฐานที่ชื่อว่า wire.hแต่เนื่องจากบอร์ด Arduino กับตัว IMU ใช้แรงดันไฟไม่เท่ากัน ปัญหานี้แก้ไขได้โดยต่อผ่าน Protoboard โดยมีตัวต้านทานลดแรงดันไฟให้เท่ากัน



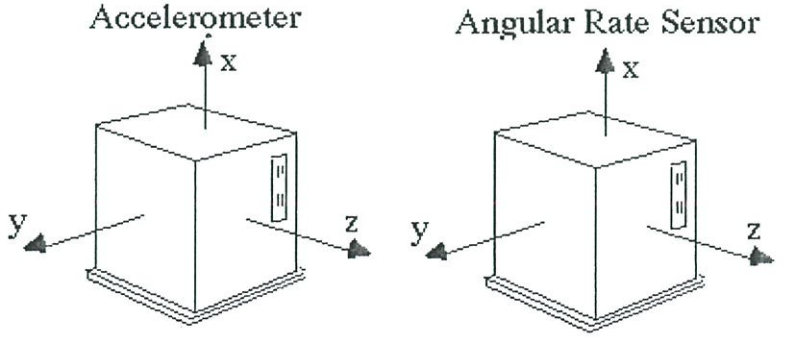
รูปที่ 2.15 การต่อ IMU เข้ากับ Arduino

นอกจากนั้นยังสามารถเชื่อมต่อกับคอมพิวเตอร์ได้โดยต่อผ่าน FDTI Basic Breakout Board



รูปที่ 2.16 การเชื่อมต่อ IMU โดยต่อผ่าน FDTI Basic

2.14.1 เซนเซอร์วัดความเร่ง (Accelerometer)

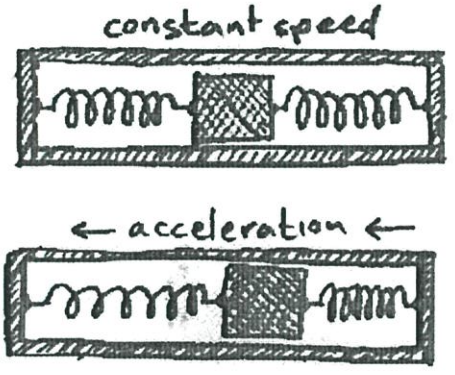


รูปที่ 2.17 แสดงชนิดของเซนเซอร์วัดความเร่ง

ในปัจจุบันอุปกรณ์ Sensor ต่างๆ เริ่มมีการติดตั้งอยู่ในอุปกรณ์ทั่วไป เช่น GPS ในรถยนต์ หรือโมดูล GPRS ใช้ทำโทรศัพท์ หรือว่าจะเป็น Ultrasonic ที่ใช้ทำsensor จับวัตถุที่ท้ายรถยนต์ มี Sensor ชนิดหนึ่ง ที่ใช้วัดความเอียง นั้นก็คือ Accelerometer และ Gyroscope

1.1 เครื่องวัดอัตราเร่ง (Accelerometer)

หากแปลตามตัวอักษรแล้ว Accelerometer มาจาก Acceleration + Meter หรือมิเตอร์ความเร่ง ตามนิยามก็คือ Sensor วัดความเร่งเพิ่มขึ้น หรือลดลง (ในหน่วย m/s2)



รูปที่ 2.18 แสดงหลักการของเซนเซอร์วัดความเร่ง

โดยหลักการทำงาน ให้นึกถึงห้องสี่เหลี่ยมเล็กๆ ที่ทุกด้านของกำแพงจะมีสปริงติดอยู่ เวลาที่ห้องนี้เอียงไปทางใดทางหนึ่ง สปริงก็จะยุบไปด้านนั้นๆ โดยสมมติว่าแรงดันของสปริงมีน้อยกว่าแรงโน้มถ่วงของโลก และใช้วงจรไฟฟ้าในการดึง Output Analog ออกมาใช้งาน (หรือ Output Digital ซึ่งก็แล้วแต่ตัว Sensor เอง) เรา จะใช้ Accelerometer สำหรับเป็นตัวชี้ว่าอยู่ในสถานะ Static (นิ่งเฉย) หรือ Dynamic (เคลื่อนไหวทันทีทันใด หรือหยุดทันทีทันใด) นั่นทำให้ Accelerometer เป็น sensor สำหรับบอกสถานะการเอียงได้เป็นอย่างดี (Tilt Sensor) Sensor ยกตัวอย่างเช่นในโทรศัพท์สมาร์ทโฟนในปัจจุบันจะมีการตรวจวัดอัตราเร่งและการตรวจวัดอัตราเร่ง โดยอัตราเร่ง คือ อัตราการเปลี่ยนแปลงของความเร็วเทียบกับเวลา เครื่องมือที่ใช้วัดอัตราเร่งก็ คือ มิเตอร์วัดอัตราเร่ง หรือแอกเซเลอโรมอเตอร์ (accelerometer) โดยที่สามารถแบ่งลักษณะการ ตรวจวัดได้ 2 ลักษณะ

1. การตรวจวัดการช็อก (shock) และการสั่นสะเทือน (vibration) ซึ่งการช็อก คืออัตราเร่งขนาดมหาศาลที่เกิดขึ้นในช่วงเวลาสั้นๆ การสั่นสะเทือน คือ อัตราเร่งขนาดเล็กที่เกิดขึ้นซ้ำกันไป เรื่อยๆ

2. การตรวจวัดอัตราเร่งของวัตถุ เพื่อนำข้อมูลไปใช้ในการระบุตำแหน่ง ความเร็ว และ ระยะทางที่ได้จากการเคลื่อนที่

1.2 ประเภทของมิเตอร์วัดความเร่ง

โดยหลักๆแล้วจะแบ่งเป็น 2 ชนิด คือ

1.2.1 มิเตอร์วัดอัตราเร่งแบบไซซมิกแมส (seismic mass accelerometer)

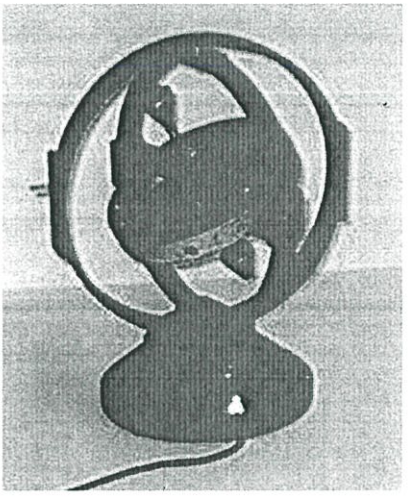
มิเตอร์ชนิดนี้อาศัยหลักการตรวจวัดระยะขจัดเชิงเส้นแล้วนำไปคำนวณหาอัตราเร่งที่เกิดขึ้น โดยเทคนิคดังกล่าวสามารถอธิบายง่ายๆ ได้ก็คือ วัตถุชิ้นหนึ่งจะมีความเร่งได้ ก็จะต้องมีแรงมากระทำ ยิ่งมีแรงมากระทำมาก ก็จะยิ่งมีความเร่งมาก ในขณะเดียวกันแรงต้านการเคลื่อนที่ก็จะมากด้วย นอกจากนี้เมื่อมีแรงมาทำให้วัตถุเกิดการเคลื่อนที่ก็จะมีระยะขจัด ซึ่งก็จะแปรผันตรงกับแรงที่มากระทำที่วัตถุ ยิ่งแรงมากระยะขจัดยิ่งมาก จากความสัมพันธ์ดังกล่าวได้นำไป ใช้เป็นหลักการพื้นฐานของมิเตอร์วัดอัตราเร่งแบบไซซมิกแมสในการตรวจวัดอัตรา เร่งของวัตถุในเทอมของระยะขจัดที่เกิดขึ้น มิเตอร์วัดอัตราเร่งแบบไซซมิกแมสนี้จะนิยมใช้ในการตรวจวัดลักษณะการช็อกและลักษณะการสั่นสะเทือนที่มีความถี่ต่ำมากๆ เช่น ในเครื่องมือตรวจวัดแผ่นดินไหว หรือในเครื่องมือตรวจวัดการปะทุใต้ดินของภูเขาไฟ ฯลฯ

1.2.2 มิเตอร์วัดอัตราเร่งแบบเพียโซอิเล็กทริก (piezoelectric accelerometer)

คุณสมบัติพื้นฐานทางไฟฟ้าของผลึก เพียโซอิเล็กทริก (piezoelectric crystal) ถูกค้นพบโดย Pierre และ Jacques Curie ในราวปี ค.ศ.1880 ซึ่ง piezoelectric crystal นี้มีคุณสมบัติพิเศษคือ เมื่อถูกแรงทางกลมากระทำจะสร้างประจุไฟฟ้าขึ้นมา โดยเป็นสัดส่วนกับแรงกระทำนั้น ซึ่งจากคุณสมบัติพิเศษนี้ได้ถูกดัดแปลงนำไปใช้สร้างอุปกรณ์ต่างๆมากมาย เช่น ใช้เป็นแบตเตอรี่จ่ายพลังงานไฟฟ้าให้กับนาฬิกาข้อมือดิจิตอลที่เราใช้ทั่วไป และยังใช้สร้างมิเตอร์วัดอัตราเร่งแบบเพียโซอิเล็กทริกอีกด้วย โครงสร้างของมอเตอร์วัดอัตราเร่ง แบบเพียโซอิเล็กทริกจะประกอบด้วย seismic mass ยึดติดกับ piezoelectric crystal และบรรจุอยู่ในตัวถังป้องกัน โดย piezoelectric crystal ที่นิยมนำมาใช้งาน ได้แก่ ผลึกควอตซ์ และผลึกโซเดียมโปตัสเซียมตาเตรต (sodium potassium tartrate) เพราะมีความทนทานต่อแรงกระทำและราคาไม่แพงมากนัก

2.14.2 เครื่องวัดสภาพการเอียง (Gyroscope)

Gyroscope คืออุปกรณ์ที่นำเอามาใช้สำหรับตรวจเช็คสภาพการเอียงการโคลงเคลงของวัตถุที่ติดตั้งตัวมันอยู่ เช่น เครื่องบิน เรือหุ่นยนต์ จรวด จานดาวเทียมซึ่งของเหล่านี้ล้วนแต่ต้องการที่จะทราบสถานะความสมดุลของมันอยู่ตลอดเวลา

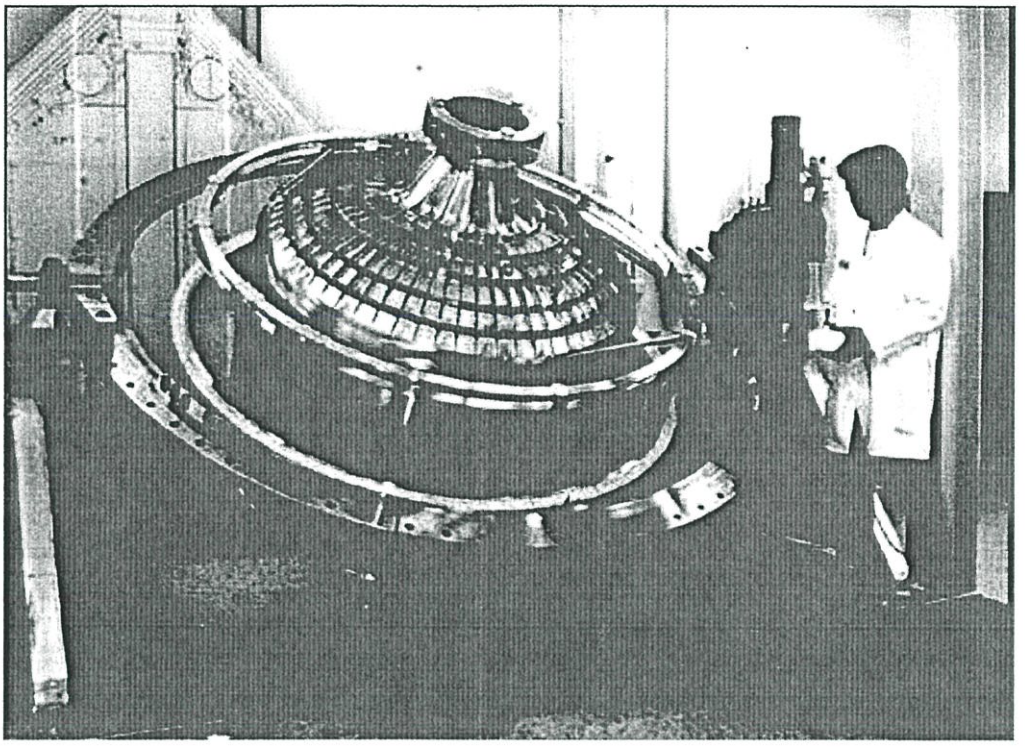


รูปที่ 2.19 เครื่องวัดสภาพความเอียง

Gyroscope ได้กำเนิดขึ้นเมื่อปี ค.ศ. 1852 ครับ โดย นักวิทยาศาสตร์ ชาวฝรั่งเศสที่ชื่อ ฌอง โบนาปาต์ เลอง ฟูโกลต์ (Leon Foucault) ได้คิดประดิษฐ์ล้อที่ติดตั้ง อยู่ในวงแหวนที่หมุนได้ เมื่อหมุนล้อหรือโรเตอร์แล้วเจ้าล้อมันจะหมุนในทิศทางเดิมของมันในอากาศ โดยไม่ขึ้นกับการเอียงไปเอียงมาของกรอบหรือวงแหวนที่ล้อมรอบมัน ซึ่งเขาได้ตั้งชื่ออุปกรณ์ตัวนี้ว่าGyroscope ครับ ซึ่งจากคุณสมบัติในข้อนี้ทำให้มันสามารถที่จะเป็นเครื่องบ่งชี้ทิศทางได้

การใช้ไจโรสโคปครั้งแรกนั้น เกิดขึ้น ในปี ค.ศ. 1910 ได้นำไปติดตั้งในเรือรบของเยอรมนี ต่อมาในปี ค.ศ. 1909 นายเอลเมอร์ เอ. เสปอร์รี (Elmer A. Sperry) ได้สร้างเครื่องบินที่บินอัตโนมัติที่ใช้คุณสมบัติการรักษาทิศทาง ของไจโรสโคปเพื่อให้เครื่องบิน บินได้ตรงทิศทาง

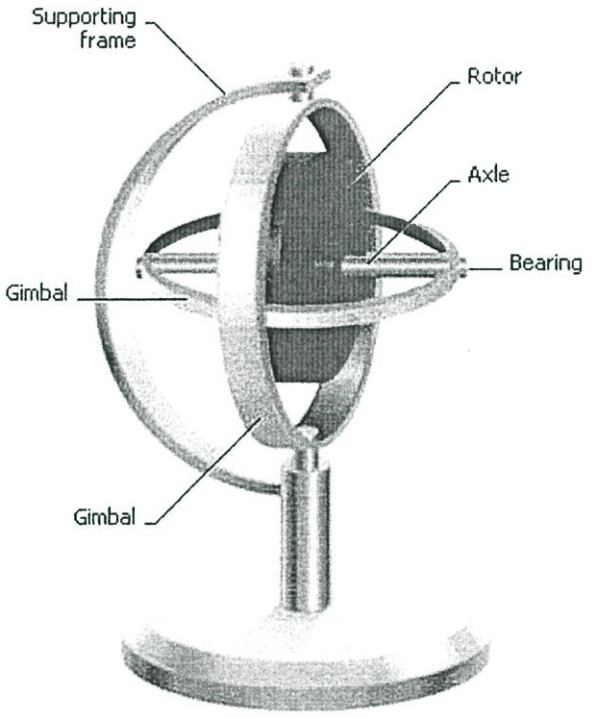
สำหรับการติดตั้งบนเรือนั้นเริ่มขึ้นในปี ค.ศ.1915 บริษัท อันชิทช ตั้งอยู่ที่ เมืองคีล ประเทศเยอรมนี ได้ติดตั้งบนเรือโดยสาร ของเดนมาร์ก ซึ่งเป็นอุปกรณ์ที่เอาไว้รักษา เสถียรภาพการของเรือเพื่อให้ ลดการโคลง ของเรือ ซึ่งทำให้สินค้ามีความเสียหายน้อยลง และทำให้ผู้โดยสารรู้สึกสบายขึ้น ซึ่งระบบนี้เรียกว่า Gyrostabilizer แต่ใช้วิธีนี้มีข้อเสีย คือ เรือนั้นมีน้ำหนักมาก และเสียเนื้อที่มาก เพราะต้องสร้างไจโรสโคปขนาดใหญ่



รูปที่ 2.20 ไจโรสโคปรักษาเสถียรภาพของเรือ

ต่อมาในปี 1916 มีผู้นำหลักการของไจโรสโคปนี้ไปทำอุปกรณ์ที่เรียกว่า ขอบฟ้าประดิษฐ์ เพื่อใช้ในเครื่องบินสำหรับบอก ระดับการโคลงเคลง ไปซ้ายขวา หน้าหลัง ซึ่งมีประโยชน์อย่างยิ่ง ในเวลาที่สภาพอากาศไม่ดีเวลาที่มองไม่เห็นขอบฟ้าหลังจากนั้นวิศวกรก็มีได้มีการนำไจโรสโคปไปใช้อย่างมากมายแพร่หลาย เช่น นำไปใช้กับจรวดนำวิถี เพื่อบังคับทิศทางโดยอัตโนมัติ หรือนำไปใช้ ในเครื่องบิน เป็นต้น

การทำงานของ Gyroscope นั้น จะเป็นไปตามกฎของนิวตันคือ มวลจะเคลื่อนที่เป็นเส้นตรงด้วยความเร็วคงที่ ถ้าไม่มีแรงภายนอกมากระทำ เมื่อตัวไจโรหมุนไป 90 องศา จุดบนจะหมุนเปลี่ยนตำแหน่งไป 90 องศา และยังเคลื่อนที่ไปทางซ้ายเช่นเดียวกับจุดล่าง เมื่อหมุนขึ้นมา 90 องศา มันยังคงเคลื่อนที่ไปทางขวา ทำให้ล้อเกิดการหมุนควง ขณะที่จุดบนและจุดล่างเปลี่ยนตำแหน่งไป 90 องศา การเคลื่อนที่ในครั้งแรก จะถูกยกเลิกไป ไม่เกิดการพลิกของล้อ ดังนั้นแกนหมุนของไจโรจะเหมือนกับห้อยอยู่กับที่ตลอดเวลา



รูปที่ 2.21 วงล้อหมุนตามกฎของนิวตัน

ดังนั้นเมื่อเอาคุณสมบัติแบบนี้ไปสร้างอุปกรณ์วัดระดับเราก็จะสามารถรู้ได้เลยว่าอุปกรณ์ที่ติดตั้งGyro เอียงไปจากเดิมเท่าไร ซึ่งเมื่อโรเตอร์หมุนแล้ว มันก็จะคงตำแหน่งอยู่แบบนั้น กรอบที่เป็นแหวนซึ่งอยู่ล้อมรอบเมื่อเราได้ติดตั้งตัววัดระยะ วัดการหมุนก็สามารถเช็คมุมที่เปลี่ยนไปได้

การนำ Gyroscope มาประยุกต์ใช้

1. Demonstration Gyroscope

Demonstration Gyroscope มักจะพบในแหล่งการศึกษาเพื่อศึกษาลักษณะทางกายภาพของGyroscope ลักษณะทั่วไปของมันมักจะมีลักษณะเหมาะที่จะใช้ในการรับแรงผ่านวงแหวนเพื่อให้ผู้ใช้ทดสอบการหาตำแหน่งใน1 มิติ

2. Computer Pointing Devices

มีการนำ Gyroscope มาใช้เป็นอุปกรณ์เพื่อบอกตำแหน่งในคอมพิวเตอร์เช่นการควบคุมเมาส์ในอากาศ ดังรูป เรียกว่า Gyromouse ซึ่งประกอบด้วย 3 เทคโนโลยี คือ Gyroscope ,การทำงานของเมาส์ (mouse ball mechanism) ,สัญญาณความถี่วิทยุ (radio frequency) ในการเลื่อนของ Gyroscope ภายในตัวเมาส์จะทำให้ทราบตำแหน่ง cursorได้ โดยหลักการทำงานของเมาส์ที่ใช้งานทั่วไป ส่วน radio เทคโนโลยีใช้สำหรับสร้าง Gyromouse แบบไร้สายช่วยควบคุมระยะได้ประมาณ40 ฟุต

3. Racing Car

คุณลักษณะของ Gyroscopeเชิงพฤติกรรม (Gyroscopic behavior) มักจะใช้ในอุตสาหกรรมรถแข่ง รถจะเลี้ยวในทางเดียวเท่านั้น เมื่อรถเกิดการเลี้ยวจะมีลักษณะเป็นพฤติกรรมทาง Gyro ดังนี้เครื่องยนต์จะกดตัวรถด้านหน้าให้ต่ำลงและยกด้านหลังให้สูงขึ้นไปในอากาศทำให้รถสามารถทรงตัวอยู่ได้

4. Motor Bikes

เช่นเดียวกับ Racing Carโดยพิจารณาที่การทำงานของล้อเคลื่อนที่แบบGyroโดยอาศัยมุมในการเลี้ยวช่วยด้วย

5. Toy

ของเล่นที่ใช้การหมุนเป็นรูปแบบหนึ่งของ Gyroscope เมื่อด้านหมุนของวัตถุนั้นหมุนจะเกิดแรงยกวัตถุนั้นขึ้นไม่ให้ล้มลง

6. Gyrocompasses

Gyroscope ช่วยในการบอกทิศทางมีการใช้งานที่ง่ายสามารถนำไปติดที่อุปกรณ์เพื่อให้เคลื่อนที่ได้อย่างอิสระเนื่องจากการเปลี่ยนทิศทางของอุปกรณ์ไม่ทำให้ Gyroscope เปลี่ยนทิศไปด้วยการหมุนไปของ Gyroscope โดยแกนของ Gyroไม่เปลี่ยนนี้สามารถนำไปวัดความเปลี่ยนแปลงของทิศทางได้ มักจะใช้ในเรือหรือยานอวกาศในการบอกทิศทาง

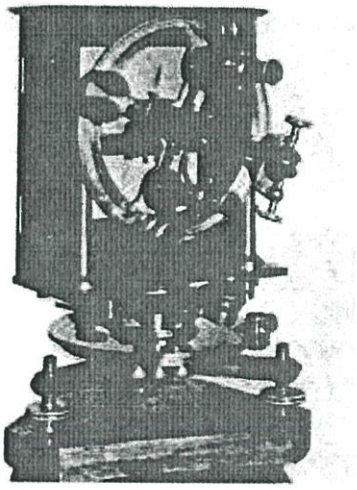
7. Gyroscopic anti-roll devices/stabiliser

นำไปใช้ในรถไฟรางเดียวหรือเรือโดยอาศัยหลักการทรงตัวของมันทำให้รถไฟทรงตัวอยู่ได้และทำให้เรือไม่ถูกซัดล้มลงโดยคลื่นเนื่องจากไม่สามารถเปลี่ยนทิศทางได้ทัน

8. อื่น ๆ

เช่นการใช้งานทางทหาร,การสร้างHelicopter แบบGyro เป็นต้น

2.14.3 เครื่องวัดสนามแม่เหล็ก (Magnetometer)



รูปที่ 2.22 เครื่องวัดสนามแม่เหล็ก

แม่เหล็ก (magnet) เป็นสิ่งที่สามารถดูวัสดุบางชนิดได้ เช่น เหล็ก นิกเกิล โคบอลต์ เป็นต้น การที่แม่เหล็กดูดสารบางอย่างได้ เนื่องจากมีสนามแม่เหล็ก (magnetic field) ในบริเวณโดยรอบแม่เหล็ก เราสามารถตรวจสอบว่าบริเวณใดมีสนามแม่เหล็กหรือไม่ โดยใช้เข็มทิศ แต่เราไม่สามารถทราบได้ว่ามีค่าเท่าใด นักวิทยาศาสตร์พยายามวัดสนามแม่เหล็กด้วยวิธีการต่าง ๆ แต่ในปัจจุบันเราสามารถวัดสนามแม่เหล็กได้สะดวกและรวดเร็วโดยใช้ตัวรับรู้ฮอลล์ (Linear Hall sensor) ซึ่งทำงานโดยอาศัยหลักการของปรากฏการณ์ฮอลล์ (Hall effect) ตัวรับรู้ฮอลล์เป็นวงจรรวมที่ทำให้เกิดความต่างศักย์ซึ่งเป็นสัดส่วนตรงกับความเข้มของสนามแม่เหล็กที่ผ่านในแนวดิ่ง เมื่อนำตัวรับรู้ฮอลล์ไปต่อกับโวลต์มิเตอร์ แล้วนำไปวางใกล้บริเวณที่มีสนามแม่เหล็กก็จะทำให้ทราบค่าความเข้มของสนามแม่เหล็กได้

ความเป็นมาของการสำรวจสนามแม่เหล็ก

ความเปนมาของการสำรวจวัดสนามแมเหล็ก การพบสารแมเหล็กหรือวัตถุที่มีความสามารถดูดสารที่เปนเหล็กไดนั้น มีมาตั้งแตสมัยกรีก โบราณ หรือกวาสองพันปมาแลว โดยพบเปนกอนหินที่มีแรแมกนีไทตประกอบ เรียกชื่อหินนั้นวา โลดสโตน (lodestone) ซึ่งคำวา lodestone มาจาก lode + stone ซึ่ง stone หมายถึง หิน lode มาจาก lead or attract หมายถึง นำหรือดึงดูด (lead or attract) ดังนั้นความหมายของ lodestone คือ หินที่มีแรงดึงดูด หินกอนนี้พบที่เมือง Magnesia จึงเปนที่มาของคำวา magnetism อันเปนชื่อของเมืองที่พบกอนหินที่มี คุณสมบัติดึงดูดวัตถุที่เปนเหล็กไดนั่นเอง ประโยชนที่เห็นชัดเจนหลังจากพบหินที่มีแรงดึงดูดวัตถุ ประเภทที่เปนเหล็กได คือ การนำมาใชเปนหินนำทาง หรือทำเปนเข็มทิศเพื่อใชเดินเรือ โดยเริ่มครั้งแรก ที่ประเทศจีนประมาณ ศตวรรษที่ 12 นับจากมีการพบหินโลดสโตนก็ไดมีการพยายามคิดคนหาประโยชนจากหินแมเหล็กนอกเหนือจากการนำมาทำเปนเข็มทิศเรื่อยมา รวมทั้งการพยายามหาคำตอบที่นาเชื่อถือ หรือมีความเปนไปไดวาทำไมเข็มทิศจึงชี้ไปในทิศทางเดียวไมเปลี่ยนแปลง คือ ชี้ไปที่ขั้วโลกเหนือ สวนการ ประยุกตสำรวจทางธรณีฟสิกสเพื่อศึกษาธรณีวิทยาใตผิวดิน ดวยการวัดสนามแมเหล็กเริ่มประมาณตนศตวรรษที่ 16 โดย กิลเบิรต (William Gilbert, 1544-1603) ที่ไดอธิบายถึงเหตุผลของเข็มทิศที่ชี้ไปที่ขั้ว โลกเหนือ เพราะโลกเสมือนมีแทงแมเหล็กขนาดใหญมีขั้วเหนือแมเหล็กอยูทางขั้วใตภูมิศาสตรของ โลก และขั้วใตแมเหล็กอยูทางขั้วเหนือภูมิศาสตรของโลก จึงทำใหมีสนามแมเหล็กของโลกเกิดขึ้น สงผลใหวัตถุที่มีอำนาจแมเหล็ก มีการดูดและผลักตามความแรงของสนามแมเหล็กโลก

ในชวงตนศตวรรษที่ 19 เกาส (Karl Fredrick Gauss, 1777-1855) ไดแสดงเหตุผลยืนยันความคิดของ กิลเบิรต โดยแสดงใหเห็นวาสนามแมเหล็กโลกไมไดเกิดจากผลภายนอกโลก แตเปนผลจากภายในโลกเอง ซึ่งอธิบายวาเปนชั้นที่เกิดจากแกนโลกชั้นนอกที่เปนของเหลวเคลื่อนที่ ทำใหเกิดกระแสไฟฟาไหลรอบๆ แกนโลกชั้นในที่เปนสวนผสมของเหล็กและนิเกล เปนผลทำใหเกิด สนามแมเหล็กไฟฟาขนาดใหญ ความสำเร็จของการประยุกตสำรวจวัดสนามแมเหล็กเพื่อแปล ความหมายหาสภาพธรณีวิทยาใตผิวดิน ยกใหเปนความสำเร็จของ von Wrede ที่ทำการตอยอดจากผล การศึกษาของกิลเบิรต เมื่อประมาณป ค.ศ. 1843 ซึ่ง von Wrede ซึ่งไดประยุกตใชหาตำแหนงของแร เหล็กดวยการวัดสนามแมเหล็ก ตอมาใน ป ค.ศ. 1879 Thalen ไดตีพิมพรายงานการสำรวจทางธรณี ฟสิกส โดยใหชื่อเรื่องวา “The Examination of Iron Ore Deposits by Magnetic Measurement” จากนั้นมี การประยุกตสำรวจเพื่อหาแหลงแรเหล็ก หินฐาน และดานอื่นๆ เรื่อยมา

ทฤษฎีและหลักการพื้นฐานของการสำรวจวัดสนามแมเหล็ก

1. แมเหล็ก (Magnet) เราเรียกสารหรือวัตถุที่มีคุณสมบัติดูดเหล็ก หรือเหนี่ยวนำสารที่เปนเหล็กใหมีสภาพกลายเปน สารที่ดูดเหล็กไดหรือมีอำนาจแมเหล็กวา “แมเหล็ก (magnet)” ซึ่งแมเหล็กจะมีขั้ว 2 ขั้ว คือ ขั้วเหนือ และขั้วใต หรือขั้วบวกและขั้วลบ เมื่อนำเอาวัตถุที่มีสภาพเปนแมเหล็กเชนเดียวกัน มาวางใกลกัน ใน กรณีที่ขั้วตางกันวางอยูใกลกันจะเกิดแรงดูดกัน ในกรณีขั้วเหมือนกันอยูใกลกันจะเกิดแรงผลักกัน

2. แรงแมเหล็ก (Magnetic Force, Fm) แรงแมเหล็กเปนแรงที่เกิดขึ้นเองตามธรรมชาติ เชนเดียวกับแรงโนมถวง แรงทางไฟฟา หรือแรงนิวเคลียร การคำนวณหาแรงแมเหล็กสามารถใชกฎของคูลอมบ (Coulomb) หรือกฎแรงดึงดูด ระหวางมวลของนิวตันมาประยุกตอธิบายได

เครื่องมือการสำรวจ

เราเรียกเครื่องมือที่ใชในการวัดสนามแมเหล็กวา แมกนิโทมิเตอร (magnetometer) หรือ เกาส มิเตอร (gauss meter) เครื่องมือนี้ใชสำหรับวัดความเขมและทิศทางของสนามแมเหล็ก แบงออกเปน 2 ประเภทใหญๆ คือ

(1) สเกลารแมกนิโทมิเตอร (scalar magnetometer) สำหรับวัดความเขมของ สนามแมเหล็กโลกรวม ตัวอยางเครื่องมือประเภทนี้เชน โปรตอนพรีเซสชัน แมกนิโทมิเตอร (Proton precession magnetometer) และ ออฟติคัลลีพัมพ แมกนิโทมิเตอร (Optically pumped magnetometer)

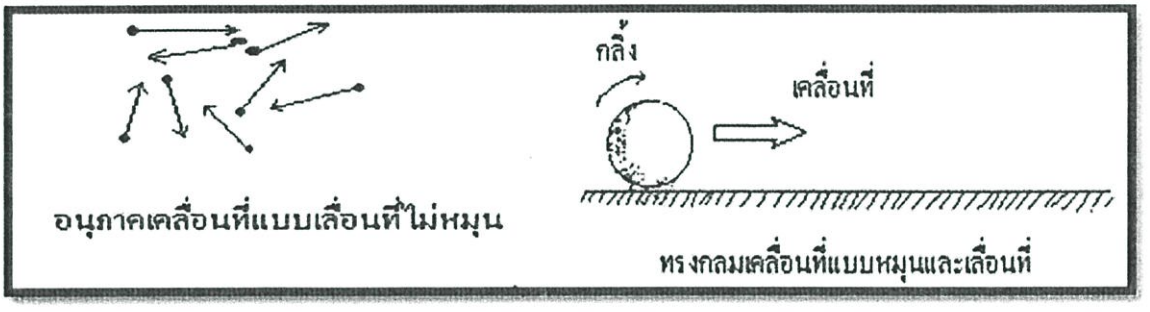
(2) เวคเตอรแมกนิโทมิเตอร (vector magnetometer) สำหรับวัดเวคเตอรองคประกอบ เชน เวคเตอร แนวดิ่ง หรือแนวนอน แนวใดแนวหนึ่งหรือทั้งสองแนวของเวคเตอรสนามแมเหล็กโลก ตัวอยางของ เครื่องมือประเภทนี้ เชน ฟลักซเกท แมกนิโทมิเตอร (Fluxgate magnetometer) และ สคิด แมกนิโท มิเตอร (SQUID (Superconduction Quantum Interference Device) magnetometer)

2.15 การเคลื่อนที่แนวตรง

ลักษณะการเคลื่อนที่

ในธรรมชาติมีการเคลื่อนที่หลายลักษณะ เช่น รถยนต์แล่นไปตามถนน การหมุนของวงล้อจักรยานการกระเพื่อมขึ้นลงของผิวน้ำ การเคลื่อนที่ทั้งหลายเหล่านี้ล้วนเกี่ยวข้องกับตำแหน่งและการเปลี่ยนตำแหน่งในช่วงเวลาหนึ่ง ๆ ถ้าเป็นการเคลื่อนที่ของรถยนต์แล่นไปตามถนนลักษณะที่จะเกี่ยวกับตำแหน่งและการเปลี่ยนตำแหน่งของรถยนต์ เป็นต้น

การเคลื่อนที่ของวัตถุต่าง ๆ สามารถแบ่งเป็นการเคลื่อนที่แบบเลื่อนที่และการเคลื่อนที่แบบหมุนอนุภาค สามารถเคลื่อนที่แบบเลื่อนที่ได้เท่านั้นโดยไม่สามารถเคลื่อนที่แบบหมุน แต่วัตถุแข็งเกร็งจะสามารถเคลื่อนที่แบบเลื่อนที่และแบบหมุน



รูปที่ 2.23 การเคลื่อนที่แนวตรง

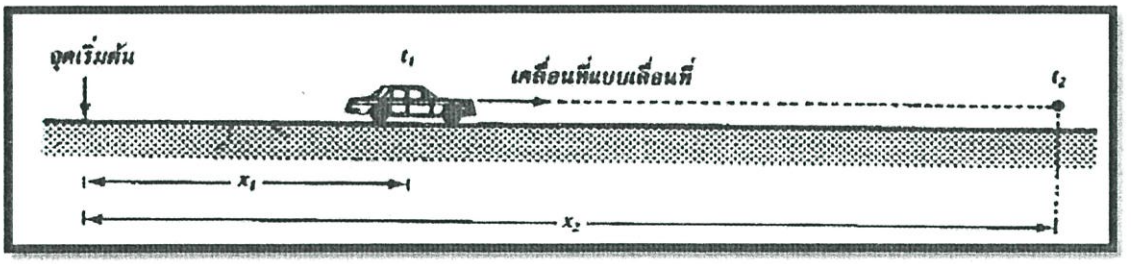
การศึกษาการเคลื่อนที่ของวัตถุในธรรมชาติแบ่งเป็นการศึกษาใน 2 ลักษณะ คือ kinematics และ dynamics สำหรับ kinematics เป็นการศึกษาการเคลื่อนที่ของวัตถุโดยไม่คำนึงถึงสาเหตุที่ทำให้วัตถุเคลื่อนที่ไป สำหรับ dynamics จะเป็นการศึกษาการเคลื่อนที่ของวัตถุโดยศึกษาถึงสาเหตุที่ทำให้วัตถุเคลื่อนที่ไป สำหรับการศึกษาในบทนี้จะเป็นการศึกษาในแนวของ kinematics และในบทที่ 7 เรื่องเกี่ยวกับนิวตันจะเป็นการศึกษาในแนวของ dynamics

ระยะทางการเคลื่อนที่

ระยะทาง หมายถึง ระยะที่วัตถุเคลื่อนที่ได้จริง ๆ โดยจะต้องตำแหน่งเริ่มต้นของวัตถุ ตำแหน่งสุดท้ายของวัตถุและเส้นทางการเคลื่อนที่ของวัตถุ ระยะทางเป็นปริมาณสเกลาร์ มีหน่วยเป็นเมตร (m)

อัตราของวัตถุ

อัตราเร็ว หมายถึง ระยะทางที่วัตถุเคลื่อนที่ได้ในหนึ่งหน่วยเวลา เป็นปริมาณสเกลาร์ มีหน่วยเป็น เมตร/วินาทีพิจารณาการเคลื่อนที่ของรถยนต์คันหนึ่งในแนวตรง ดังรูป



รูปที่ 2.24 แสดงการเคลื่อนที่แนวตรงของรถยนต์คันหนึ่ง

เมื่อสิ้นสุดเวลา t1 วินาที หรือ ณ เวลา t1 รถยนต์เคลื่อนที่ได้ระยะทาง x1 จากจุดเริ่มต้น และเมื่อสิ้นสุดเวลา t2 วินาที หรือ ณ เวลา t2 รถยนต์เคลื่อนที่ได้ระยะทาง x2 จากจุดเริ่มต้น และการเคลื่อนที่ตำแหน่งเดิมต่อไปเรื่อย ๆ

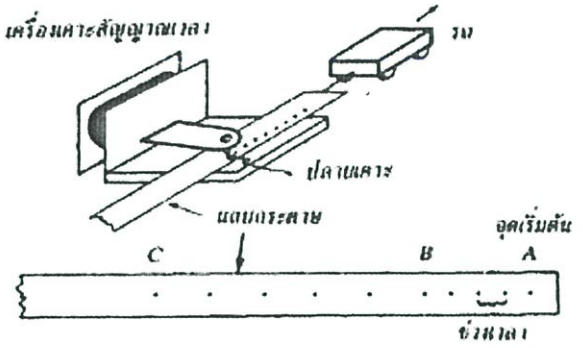
อัตราเร็วเฉลี่ย หมายถึง อัตราส่วนระหว่างระยะทางทั้งหมดที่เคลื่อนที่ได้กับช่วงเวลาที่ใช้ในการเคลื่อนที่นั้นโดยจะเขียนได้ว่า

$$v_{\text{เฉลี่ย}} = \frac{\Delta X}{\Delta t}$$

อัตราเร็วขณะหนึ่ง หมายถึง อัตราเร็วของการเคลื่อนที่ของวัตถุ ณ เวลาที่พิจารณา เช่น จากการเคลื่อนที่ของรถยนต์ ดังรูปที่ 2 เราได้กราฟระยะทางกับเวลา เป็น ดังรูปที่ 3 ถ้าต้องการหาอัตราเร็วของเวลา t สามารถหาได้จากสมการที่ 1 โดยให้เวลา t เป็นจุดกึ่งกลางของช่วงเวลา Δt และต้องคิดที่กรณีที่Δt มีค่าน้อยมาก

การวัดอัตราเร็วของการเคลื่อนที่ในแนวตรง

เครื่องเคาะสัญญาณเวลา ใช้วัดอัตราเร็วเฉลี่ยของวัตถุซึ่งเคลื่อนที่ในเวลา ๆ ดังรูป 5 รถจะลากแถบกระดาษไปในขณะที่ปลายเคาะจะเคาะกระดาษให้ปรากฏเป็นรอยด้วยอัตราการเคาะที่ 50 ครั้ง/วินาทีทำให้เราสามารถศึกษาอัตราเร็วเฉลี่ยของรถได้จากการศึกษาแถบกระดาษ



รูปที่ 2.25 เครื่องเคาะสัญญาณเวลา

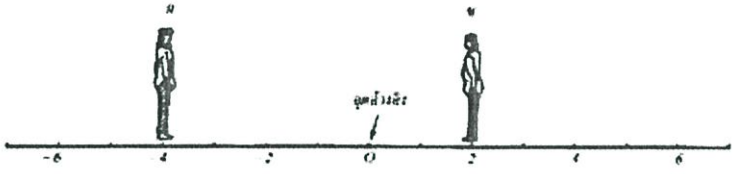
ระยะห่างจากจุดหนึ่งถัดไปบนแถบกระดาษเรียกว่า ช่วงเวลาจะมีค่าเท่ากับ 1/50 เสมอ ไม่ว่าจุดจะใกล้กันมากหรือไกลกัน ถ้า VAB เป็นอัตราเร็วเฉลี่ยของรถในช่วง AB จะได้

$$V_{เฉลี่ย} = \frac{\Delta x}{n\left(\frac{1}{50}\right)}$$

การบอกตำแหน่งของวัตถุ

เนื่องจากการเคลื่อนที่ของวัตถุเกี่ยวข้องกับตำแหน่งและการเปลี่ยนตำแหน่งของวัตถุ ดังนั้นจึงต้องทราบวิธีบอกตำแหน่งของวัตถุก่อน ดังนี้

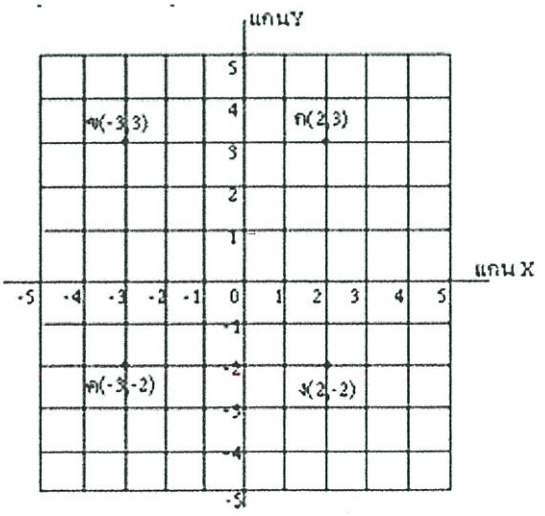
ก. การบอกตำแหน่งของวัตถุในแนวเส้นตรง (1มิติ)



จากรูปนาย ก ยืนอยู่ตรงตำแหน่ง –4 หน่วยหมายความว่า นาย ก อยู่ห่างจากจุดอ้างอิงไปทางซ้ายเป็นระยะ 4 หน่วย นาย ข ยืนอยู่ตรงตำแหน่ง +2 หน่วย จะหมายความว่า นาย ข อยู่ห่างจากจุดอ้างอิงไปทางขวาเป็นระยะ 2 หน่วยการบอกตำแหน่งของวัตถุกรณีนี้จะใช้ศึกษาการเคลื่อนที่ในแนวตรง

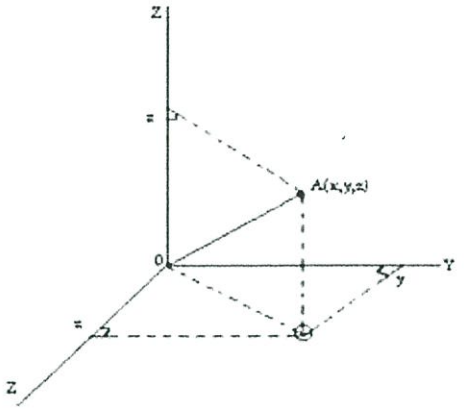
ข. การบอกตำแหน่งของวัตถุในระนาบ (2มิติ)

จะใช้เส้นตรง 2 เส้นตัดกันที่จุดกำเนิดโดยให้เส้นตรงทั้งสองตั้งฉากซึ่งกันและกันดังรูป 7 คือ จุดกำเนิด ระยะที่วัดไปทางขวาและเหนือจุดกำเนิดกำหนดให้เป็นบวก ส่วนระยะที่วัดไปทางซ้ายและล่างของจุดกำเนิดกำหนดให้เป็นลบ ตำแหน่งของวัตถุที่อยู่ในระนาบบอกได้ด้วยคู่ลำดับ (X, Y) X คือระยะจากจุดกำนิดในแกน X, y คือระยะจากจุดกำเนิดในแกนy เช่น วัตถุที่อยู่ที่จุด ก. ข. ค. และ ง. ดังรูป6.7 จะตรงอยู่ตำแหน่ง (2.3) (-3,3) ,(-3,-2) ,(2,-2) เป็นต้น



ค. การบอกตำแหน่งของวัตถุในอากาศ (3 มิติ)

จะใช้เส้นตรง 3 เส้น เรียกว่าแกน x แกน y และ แกน z ตั้งฉากซึ่งกันและกัน ตัดกันที่จุดกำเนิด O ดังรูป

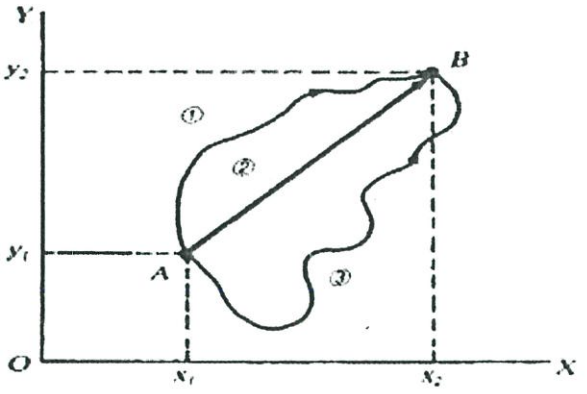


วัตถุที่ตำแหน่ง A (x, y, z) หมายความว่า ถ้าฉายไฟด้านบนในแนวแกน z จะเห็นเงาของ A ปรากฏบนระนาบ xy ที่ A' โดยที่ A จะอยู่ห่างจากแกนY เป็นระยะ x และห่างจากแกน X เป็นระยะ y จาก A ถ้าลากเส้นตรงขนานกับ A/O จะไปตัดที่แกน Z ที่ z ระยะจากจุดกำเนิด O ถึงตำแหน่ง A สามารถหาได้จาก

$$OA = \sqrt{x^2 + y^2 + z^2}$$

การกระจัด

วางวัตถุไว้ที่จุด A มีคู่ลำดับเป็น(x1,y2) ต่อมาย้ายวัตถุไปยังจุด B ซึ่งมีคู่ลำดับเป็น(x2,y2) ในการย้ายตำแหน่งจากจุด A ไปจุด B เราสามารถกระทำได้หลายทาง อาจจะใช้ทาง1,2 และ 3 ก็สามารถย้ายจาก A ไป B ได้ทั้งนั้น แต่จะมีเส้นทางหนึ่งที่ใช้ระยะทางสั้นที่สุด เส้นทางนั้นคือเส้นตรงที่ต่อระหว่างจุด A กับ B จากรูปคือ เส้นทาง 2 ลูกศรที่ชี้จาก A ไป B และมีขนาดความยาวเท่ากับ AB เรียกว่าการกระจัด(displacement) ดังนั้น การกระจัดจึงเป็นปริมาณเวกเตอร์ มีหน่วยเป็น เมตร



การกระจัดแตกต่างกันระยะทาง (distance) ตรงที่ระยะทางสนใจเพียงขนาด ไม่สนใจทิศทางและระยะทางจะเป็นระยะจริงๆ เกิดจากการย้ายตำแหน่ง เช่น ในรูป15 ถ้าเราย้ายวัตถุจากตำแหน่ง A ไปยัง B ตามเส้นทาง 1 ระยะทางจะหมายถึงระยะจริงๆ วัดตามเส้นโค้งไปมาจนถึง B ส่วนการกระจัดจะเท่ากับความยาว AB และทิศพุ่งจาก A ไป B เป็นต้น

หากจะนิยามกรกระจัดอาจกล่าวว่า “การกระจัด คือ ระยะทางที่สั้นที่สุดในการย้ายตำแหน่งจุดคู่หนึ่ง”

ความเร็ว

ความเร็ว (velocity) นิยามว่า “เป็นอัตราการเปลี่ยนแปลงการกระจัด “ พิจารณาการเคลื่อนที่ของรถยนต์ในแนวเส้นตรง (หรือวัตถุอื่นใดที่เคลื่อนที่ในแนวเส้นตรง) เริ่มออกจากจุด O เมื่อนำค่าการกระจัดของรถยนต์ ที่เวลาต่าง ๆ กันไปเขียนกราฟ โดยเขียนระหว่างการกระจัดกับเวลา

สมมติว่าได้กราฟ ดังรูป 6.16 จากกราฟการกระจัด –เวลา สามารถสรุปได้เป็นข้อ ๆ ดังนี้

$$\bar{V}_{\text{เฉลี่ย}} = \frac{\Delta \bar{s}}{\Delta t} = \frac{\bar{s}_2 - \bar{s}_1}{t_2 - t_1}$$

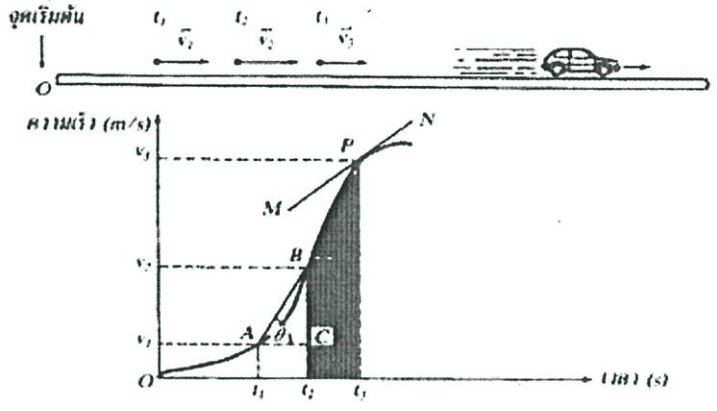
ความเร็วเฉลี่ย ตามนิยามความเร็วเราสามารถคำนวณค่าความเร็วในช่วงเวลาจาก t1 ถึง t2 ได้ดังนี้

$$\bar{V}_{\text{ขณะหนึ่ง}} = \left(\frac{\Delta \bar{s}}{\Delta t}\right) \Delta t \to 0$$

ความเร็วขณะใดขณะหนึ่ง เป็นความเร็ว ณ เวลาใดเวลาหนึ่ง นิยามว่า

ความเร่ง

ความเร่ง นิยามว่า"เป็นอัตราเปลี่ยนความเร็ว" พิจารณาการเคลื่อนที่ของรถยนต์ในแนวเส้นตรง (หรือวัตถุอื่นใดในแนวเส้นตรง) เริ่มต้นจากจุดหยุดนิ่ง ที่ O วิ่งออกไปเมื่อนำความเร็วของรถยนต์ที่เวลาต่าง ๆกันไปเขียนกราฟจะได้กราฟ



จากกราฟ ความเร็ว-เวลา สามารถสรุปได้เป็นข้อ ๆ ดังต่อไปนี้

ความเร่งเฉลี่ย ตามนิยามความเร่งเราสามารถคำนวณค่าความเร่งเฉลี่ยในช่วงเวลาจาก t1 ถึง t2 ได้ดังนี้

$$\bar{a}_{\text{เฉลี่ย}} = \frac{\bar{v}_2 - \bar{v}_1}{t_2 - t_1} = \frac{\Delta \bar{v}}{\Delta t}$$

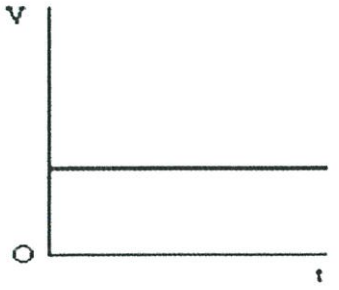
เมื่อ a เป็นความเร่งในช่วงเวลาดังกล่าว มีหน่วยเป็น เมตร/วินาที2; (m/s2) มีทิศทางไปทางเดียวกับ และจากกราฟจะพบว่าค่า นี้มีค่าเท่ากับความชัน ของเส้นตรง AB เพราะ

$$\text{ความชัน} = \tan\theta = \frac{BC}{AC} = \frac{\bar{v}_2 - \bar{v}_1}{t_2 - t_1}$$

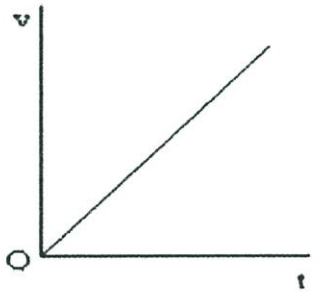
ความเร่งขณะใดขณะหนึ่ง เช่น ขณะเวลา t3 สามารถหาได้โดยการลากเส้นตรง MN ให้สัมผัสเส้นกราฟที่จุด P ค่าความชันของเส้นตรง MN ที่ได้จะเป็นค่าความเร็ว ขณะเวลา t3 เป็นต้น หรือคิดคำนวณจากสมการ 12 โดยให้ t3 เป็นจุดกึ่งกลางเวลา Δt และ $\Delta t \longrightarrow 0$

พื้นที่ใต้กราฟความเร็ว-เวลาคือการกระจัด จากกราฟ การเคลื่อนที่ของรถยนต์ในเวลา t2 ไป t3 เราสามารถหาการกระจัดในช่วงเวลาดังกล่าวได้ด้วยการหาพื้นที่ใต้กราฟส่วนที่แรเงา

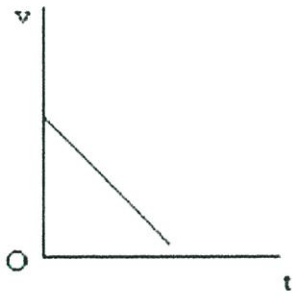
ลักษณะพิเศษของกราฟความเร็ว-เวลากับความหมาย สำหรับวัตถุที่เคลื่อนที่เป็นเส้นตรง



แสดงว่า ความเร็วคงที่ และวิ่งออกจากจุดอ้างอิงไปทางขวา



แสดงว่า ความเร็วคงที่มีค่าเป็นบวก เมื่อเวลาเพิ่มขึ้นความเร็วก็เพิ่มขึ้น และวิ่งออกจากจุดอ้างอิงไปทางขวา เพราะ ความเร็วมีค่าเป็นบวก



แสดงว่า ความเร็วคงที่มีค่าเป็นลบ เมื่อเวลาเพิ่มขึ้นความเร็วลดลง กรณีนี้มีชื่อเรียกเฉพาะว่า ความหน่วง วัตถุวิ่งออกจากจุดอ้างอิงไปทางขวา เพราะ ความเร็วมีค่าเป็นบวก

การเคลื่อนที่ของวัตถุที่ตกแบบเสรี

ถ้าเราทิ้งวัตถุก้อนหนึ่งจากที่สูง วัตถุนั้นจะตกลงสู่ผิวโลกด้วยความเร่งที่มีขนาด 9.8 เมตรต่อวินาที 2 (เพื่อความสะดวกมักใช้ 10เมตรต่อวินาที2) ใช้สัญลักษณ์ g แทนและเรียกว่า ความเร่งเนื่องจากแรงโน้มถ่วงของโลก สมการ (6-14) , (6-15) และ (6-16) สามารถนำมาใช้ได้ในความหมายเดียวกัน เพียงแต่เปลี่ยน a เป็น g

2.16 แคลคูลัส

แคลคูลัส เป็นสาขาหลักของคณิตศาสตร์ซึ่งพัฒนามาจากพีชคณิต เราขาคณิต และปัญหาทางฟิสิกส์ แคลคูลัสมีต้นกำเนิดจากสองแนวคิดหลักดังนี้

แนวคิดแรกคือ แคลคูลัสเชิงอนุพันธ์ (Differential Calculus) เป็นทฤษฎีที่ว่าด้วยอัตราการเปลี่ยนแปลง และเกี่ยวข้องกับการหาอนุพันธ์ของฟังก์ชั่นทางคณิตศาสตร์ ตัวอย่างเช่น การหาความเร็ว, ความเร่ง หรือความชันของเส้นโค้ง บนจุดที่กำหนดให้. ทฤษฎีของอนุพันธ์หลายส่วนได้แรงบันดาลใจจากปัญหาทางฟิสิกส์

แนวคิดที่สองคือ แคลคูลัสเชิงปริพันธ์ (Integral Calculus) เป็นทฤษฎีที่ได้แรงบันดาลใจจากการคำนวณหาพื้นที่หรือปริมาตรของรูปทรงทางเรขาคณิตต่าง ๆ. ทฤษฎีนี้ใช้กราฟของฟังก์ชันแทนรูปทรงทางเรขาคณิต และใช้ทฤษฎีปริพันธ์ (หรืออินทิเกรด) เป็นหลักในการคำนวณหาพื้นที่และปริมาตร**แคลคูลัสเชิงอนุพันธ์**

อนุพันธ์ (derivative) คือการหาค่าความเปลี่ยนแปลงของตัวแปรหนึ่ง เมื่ออีกตัวแปรหนึ่งเปลี่ยนแปลงในปริมาณที่น้อยมากๆ บางทีอนุพันธ์ที่เราจะได้พบครั้งแรกในโรงเรียนคือ สูตร *อัตราเร็ว = ระยะทาง/เวลา* สำหรับวัตถุที่เคลื่อนที่ด้วยอัตราเร็วคงที่ อัตราเร็วของคุณซึ่งเป็นอนุพันธ์ที่บอกการเปลี่ยนแปลงตำแหน่งในระยะเวลาหนึ่ง วิชาแคลคูลัสพัฒนาขึ้น เพื่อจัดการกับปัญหาที่ซับซ้อนและเป็นธรรมชาติกว่านี้ ซึ่งอัตราเร็วของคุณอาจเปลี่ยนแปลงได้

เมื่อเรากล่าวถึงรายละเอียดแล้ว แคลคูลัสเชิงอนุพันธ์ นิยามอัตราการเปลี่ยนแปลงในขณะใดขณะหนึ่ง (อนุพันธ์) ระหว่างค่าของฟังก์ชั่น กับตัวแปรของฟังก์ชัน นิยามจริงๆ ของอนุพันธ์คือ ลิมิตของอัตราส่วนในการเปลี่ยนแปลง (difference quotient). อนุพันธ์คือหัวใจของวิทยาศาสตร์กายภาพ กฎการเคลื่อนที่ของนิวตัน *แรง = มวล×ความเร่ง* มีความหมายในแคลคูลัส เพราะว่าความเร่งเป็นอนุพันธ์ค่าหนึ่ง ทฤษฎีแม่เหล็กไฟฟ้าของแมกซ์เวล และทฤษฎีแรงโน้มถ่วงของไอน์สไตน์ (สัมพัทธภาพทั่วไป) นั่นได้กล่าวถึงด้วยภาษาของแคลคูลัสเชิงอนุพันธ์เช่นเดียวกันกับทฤษฎีพื้นฐานของวงจรไฟฟ้า

อนุพันธ์ของฟังก์ชัน กล่าวถึงกราฟของฟังก์ชันนั้นในช่วงสั้น ๆ ซึ่งทำให้เราสามารถหาจุดสูงสุด และจุดต่ำสุด ของฟังก์ชันได้ เพราะว่าที่จุดเหล่านั้นกราฟจะขนานกับแกนราบ ดิเฟอเรนเชียล แคลคูลัสยังมีการประยุกต์ใช้อื่นๆอีก เช่น ระเบียบวิธีของนิวตัน (Newton's Method) ซึ่งเป็นวิธีในการหาค่ารากของฟังก์ชัน โดยการประมาณค่าโดยเส้นสัมผัส ดังนั้นแคลคูลัสเชิงอนุพันธ์ จึงสามารถนำไปประยุกต์ใช้กับหลากหลายคำถาม ซึ่งถ้ามองแค่ผิวเผินอาจคิดว่า ไม่อาจใช้แคลคูลัสจัดการได้**แคลคูลัสเชิงปริพันธ์**

แคลคูลัสเชิงปริพันธ์ศึกษาวิธีการหาปริพันธ์ (อินทิกรัล, Integral) ของฟังก์ชัน ซึ่งอาจนิยามจากลิมิตของผลรวมของพจน์ (ซึ่งเรียกว่าลิมิตของผลรวมรีมันน์) แต่ละพจน์นั้นคือพื้นที่ที่เป็นสี่เหลี่ยมผืนผ้าแต่ละแถบใต้กราฟของฟังก์ชัน ทำให้การอินทิเกรตเป็นวิธีที่ได้ผลวิธีหนึ่งในการหาพื้นที่ใต้กราฟ และพื้นที่ผิว และปริมาตรของแข็งเช่นทรงกลมและทรงกระบอกพื้นฐานของแคลคูลัส

พื้นฐานที่เคร่งครัดของแคลคูลัส มีฐานมาจาก แนวคิดของฟังก์ชั่น และลิมิต มันรวมเทคนิคของพีชคณิตพื้นฐาน และการอุปนัยเชิงคณิตศาสตร์ การศึกษาพื้นฐานของแคลคูลัสสมัยใหม่ รู้จักกันในชื่อ การวิเคราะห์เชิงจริง ซึ่งประกอบด้วย นิยามที่เคร่งครัด และบทพิสูจน์ของทฤษฎีของแคลคูลัส เช่นทฤษฎีการวัด และการวิเคราะห์เชิงฟังก์ชั่นทฤษฎีบทมูลฐานของแคลคูลัส เบื้องต้น

ทฤษฎีบทมูลฐานของแคลคูลัสกล่าวว่า การหาอนุพันธ์และการหาปริพันธ์เป็นวิธีการที่ตรงกันข้ามกัน กล่าวคือ ถ้าเราสร้างฟังก์ชันที่เป็นปริพันธ์ของฟังก์ชันหนึ่งขึ้นมา อนุพันธ์ของฟังก์ชันที่เราสร้าง ก็จะเท่ากับฟังก์ชันนั้นนอกจากนี้เรายังหาปริพันธ์จำกัดเขตได้ด้วยการกำหนดค่าให้กับปฏิยานุพันธ์

ทฤษฎีบทมูลฐานของแคลคูลัสเขียนในรูปสัญลักษณ์คณิตศาสตร์ได้ดังนี้: ถ้า f เป็นฟังก์ชันที่มีความต่อเนื่องบนช่วง $[a, b]$ และ F เป็นปฏิยานุพันธ์ของ f บนช่วง $[a, b]$ แล้ว

$$\int_a^b f(x)\,dx = F(b) - F(a)$$

และสำหรับทุก x ในช่วง $[a, b]$ จะได้ว่า

$$\frac{d}{dx}\int_a^x f(t)\,dt = f(x)$$

ความจริงข้อนี้ปรากฏแก่ทั้งนิวตัน และไลบ์นิซ ซึ่งเป็นกุญแจนำไปสู่การขยายผลลัพธ์เชิงวิเคราะห์อย่างมากมายหลังจากงานของทั้งสองเป็นที่รู้จัก. ความเชื่อมโยงนี้ ทำให้เราสามารถย้อนความเปลี่ยนแปลงทั้งหมดในฟังก์ชันในช่วงหนึ่ง จากอัตราการเปลี่ยนแปลงในขณะใดขณะหนึ่ง โดยการหาปริพันธ์ของส่วนหลัง. ทฤษฎีบทมูลฐานนี้ยังให้วิธีในการคำนวณหา ปริพันธ์จำกัดเขต ด้วยวิธีทางพีชคณิตเป็นจำนวนมาก โดยไม่ต้องใช้วิธีการหาลิมิต ด้วยการหาปฏิยานุพันธ์ ทฤษฎีบทนี้ยังอนุญาตให้เราแก้สมการเชิงอนุพันธ์ ซึ่งคือสมการที่เกี่ยวข้องกันระหว่าง ฟังก์ชันที่ไม่ทราบค่า และอนุพันธ์ของมัน. สมการเชิงอนุพันธ์นั้นมีอยู่ทั่วไปในวิทยาศาสตร์

การประยุกต์นำมาใช้

การพัฒนาและการใช้แคลคูลัสได้ขยายผลไปแทบทุกส่วนของการใช้ชีวิตในยุคใหม่ มันเป็นพื้นฐานของวิทยาศาสตร์เกือบทุกสาขาโดยเฉพาะ ฟิสิกส์ การพัฒนาสมัยใหม่เกือบทั้งหมด เช่น เทคนิคการก่อสร้าง การบิน และเทคโนโลยีอื่น ๆ เกือบทั้งหมด มีพื้นฐานมาจากแคลคูลัส

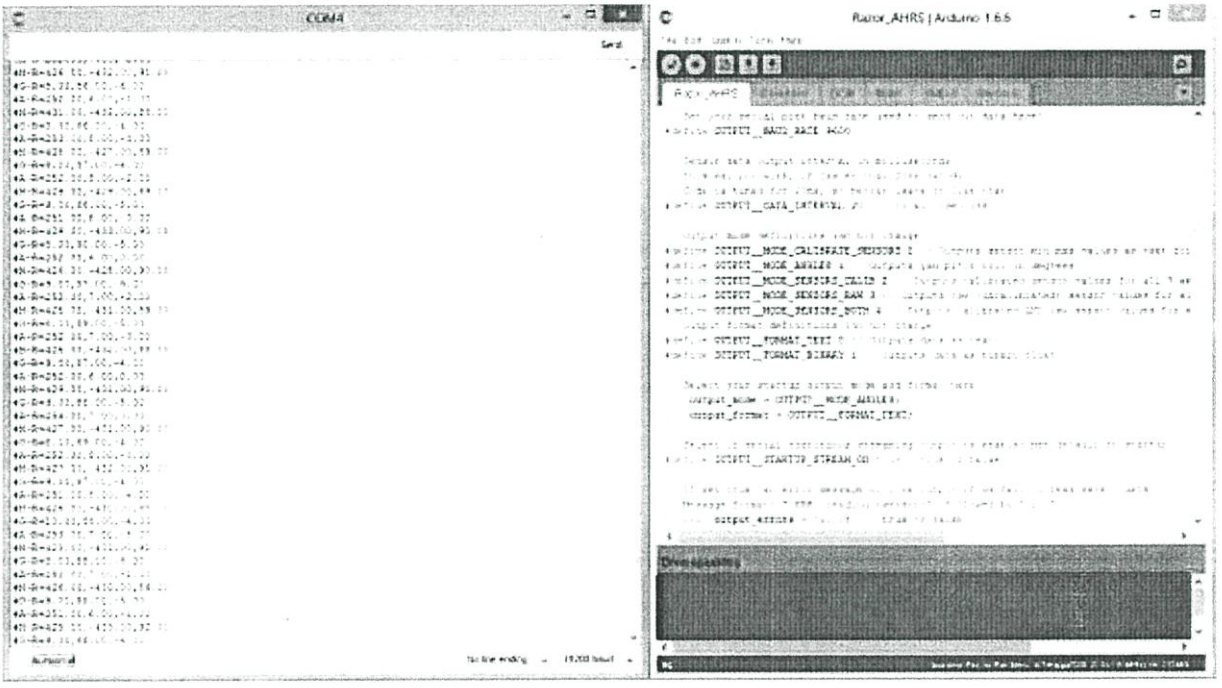
แคลคูลัสได้ขยายไปสู่ สมการเชิงอนุพันธ์ แคลคูลัสเวกเตอร์ แคลคูลัสของการเปลี่ยนแปลง การวิเคราะห์เชิงซ้อน แคลคูลัสเชิงเวลา แคลคูลัสกณิกนันต์ และ ทอพอโลยีเชิงอนุพันธ์

บทที่3

ระเบียบวิธีวิจัย

3.1 ขั้นตอนการดำเนินงาน

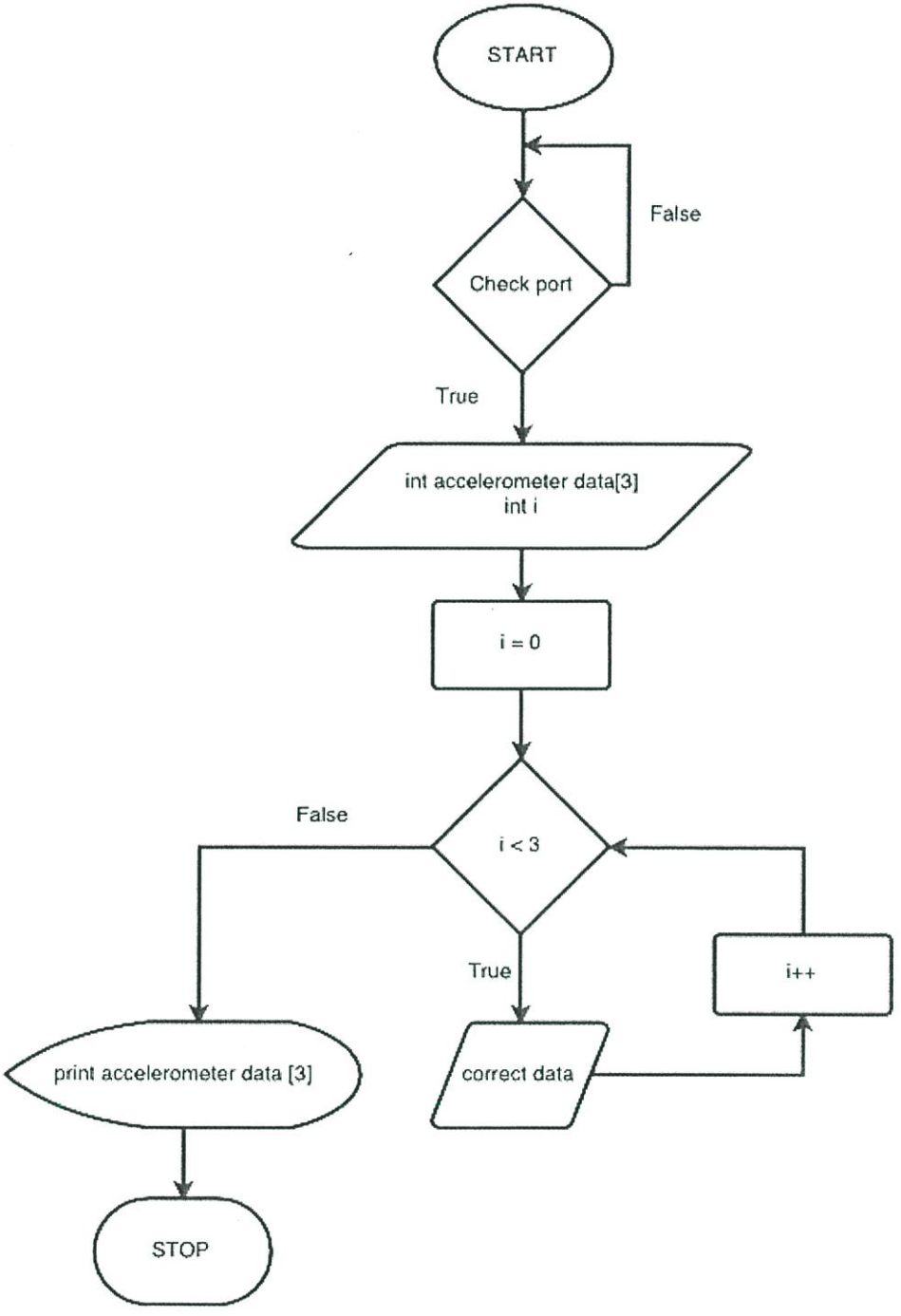
(1) ทดสอบการทำงานของ IMU โดยการศึกษา code ต้นแบบของ IMU พร้อมทดลองใช้งานในโปรแกรม Arduino ซึ่งจะได้ค่าของ sensor ออกมาเป็นค่า yaw, pitch, row และสามารถเปลี่ยนเป็นค่า raw dataได้ดังรูปที่3.1



รูปที่ 3.1 รูปการแสดงค่า raw data จาก IMU บนโปรกรม Arduino

ซึ่ง Razor IMU เป็นอุปกรณ์ที่มี sensor 3 ชนิดอยู่ภาพในตัวซึ่งคือ Accelerometer, Magnetometer, Gyroscope ซึ่งเราต้องการเพียงแค่ค่าของAccelerometerจึงต้องทำการเขียนโค้ดขึ้นใหม่เพื่อให้ทำการแสดงและส่งเฉพาะค่าของ Accelerometer

(2) นำcodeตัวอย่างมาปรับแก้เพื่อแสดงค่าในส่วนของ Accelerometer ทั้ง 3 แกน(x, y, z)



รูปที่ 3.2 แสดง Flow chart การทำงาน

(3) ใช้แอพพลิเคชั่นจากโปรแกรม Visual Studio เพื่อทำการแสดงค่าของแกน x, y, z ของ accelerometer และplotกราฟ

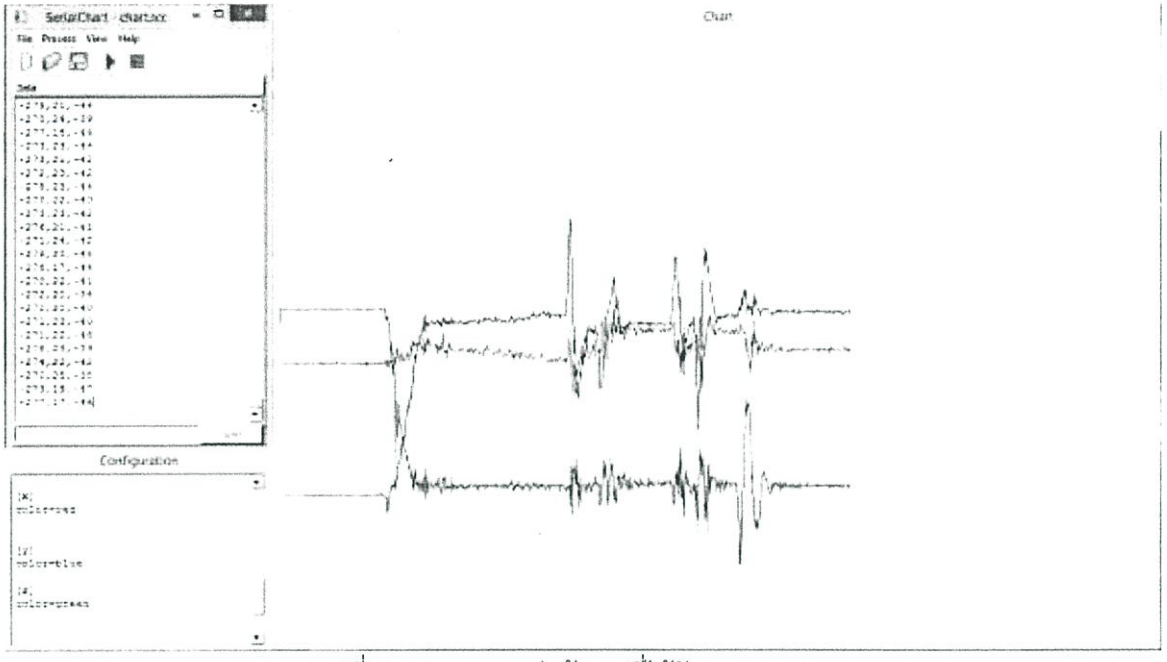
รูปที่ 3.3 โปรแกรมสำหรับ plot กราฟ

(4)การตั้งค่าในแอพพลิเคชั่นจากโปรแกรม Visual Studio สามารถทำได้โดย ปรับเปลี่ยนค่าต่างๆดังนี้

[_setup_]

Port = COM4	เลือกportที่เชื่อมต่ออยู่กับ IMU
Baud rate = 19200	ตั้งแต่rateให้สูงกว่าหรือเท่ากับในcodeของเรา
Width = 1000	ตั้งค่าความกว้างของหน้าจอกราฟ
Height = 201	ตั้งค่าความสูงของหน้าจอกราฟ
Min = -20	ตั้งค่าต่ำสุดของกราฟ
Max = 20	ตั้งค่าสูงสุดของกราฟ
[x] Color = lime	ใส่ชื่อของกราฟที่1,ตั้งค่สีของกราฟที่1
[y] Color = blue	ใส่ชื่อของกราฟที่2,ตั้งค่สีของกราฟที่2
[z] Color = red	ใส่ชื่อของกราฟที่3,ตั้งค่สีของกราฟที่3

(5) เมื่อตั้งค่าเสร็จเรียบร้อย ก็นำ IMU มาต่อเข้ากับเครื่องคอมพิวเตอร์และกดเริ่มทำงานแอพพลิเคชั่นจากโปรแกรม Visual Studio ก็จะได้ข้อมูลจากตัว IMU เป็นตัวเลขในช่อง Data และกราฟในช่อง Chartดังรูปที่ 3.4



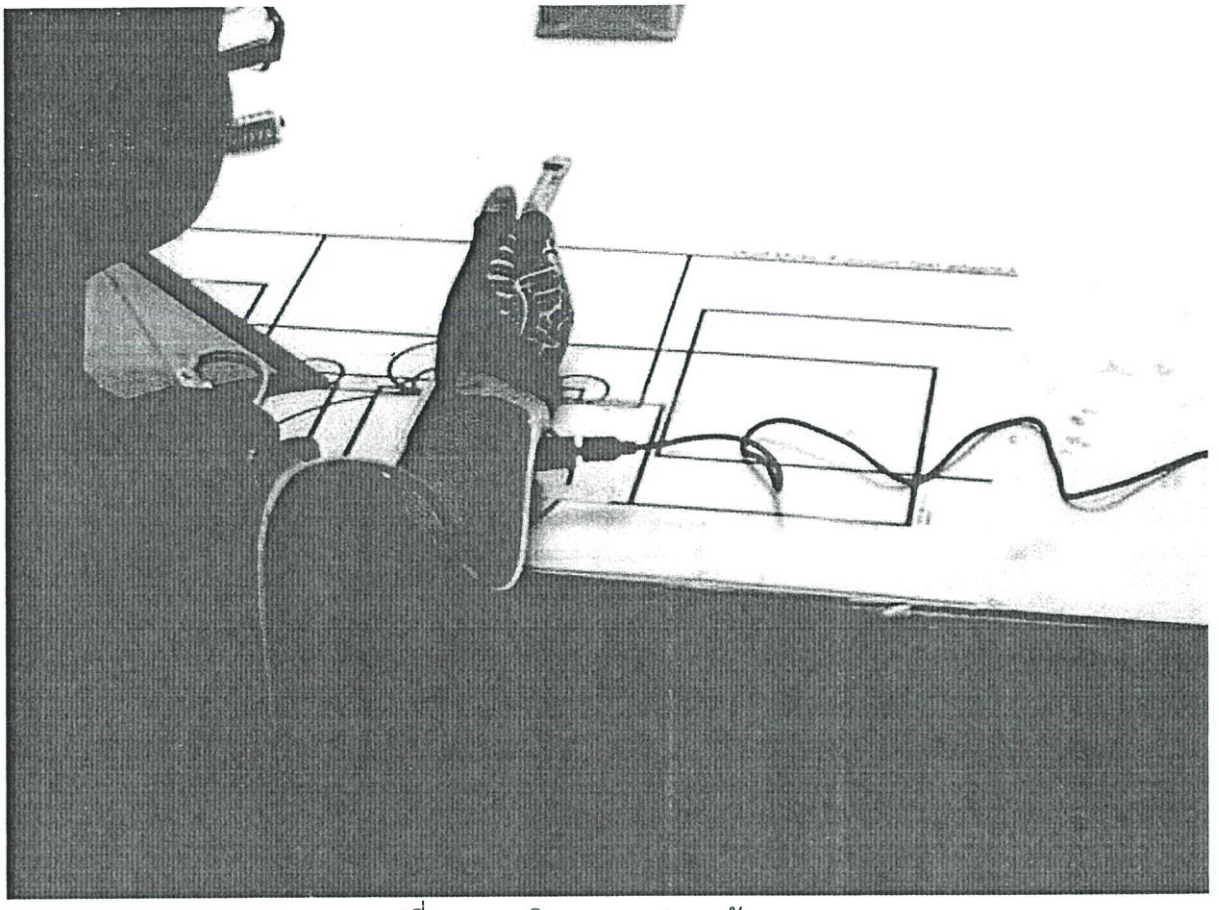
รูปที่ 3.4 การแสดงค่าข้อมูลที่ได้รับจาก IMU

(6) นำตัวชิ้นงานที่เสร็จแล้วไปใช้ในการวัดผลจริง เพื่อนำผลที่ได้มาเปรียบเทียบกับอุปกรณ์ที่ใช้อยู่จริง

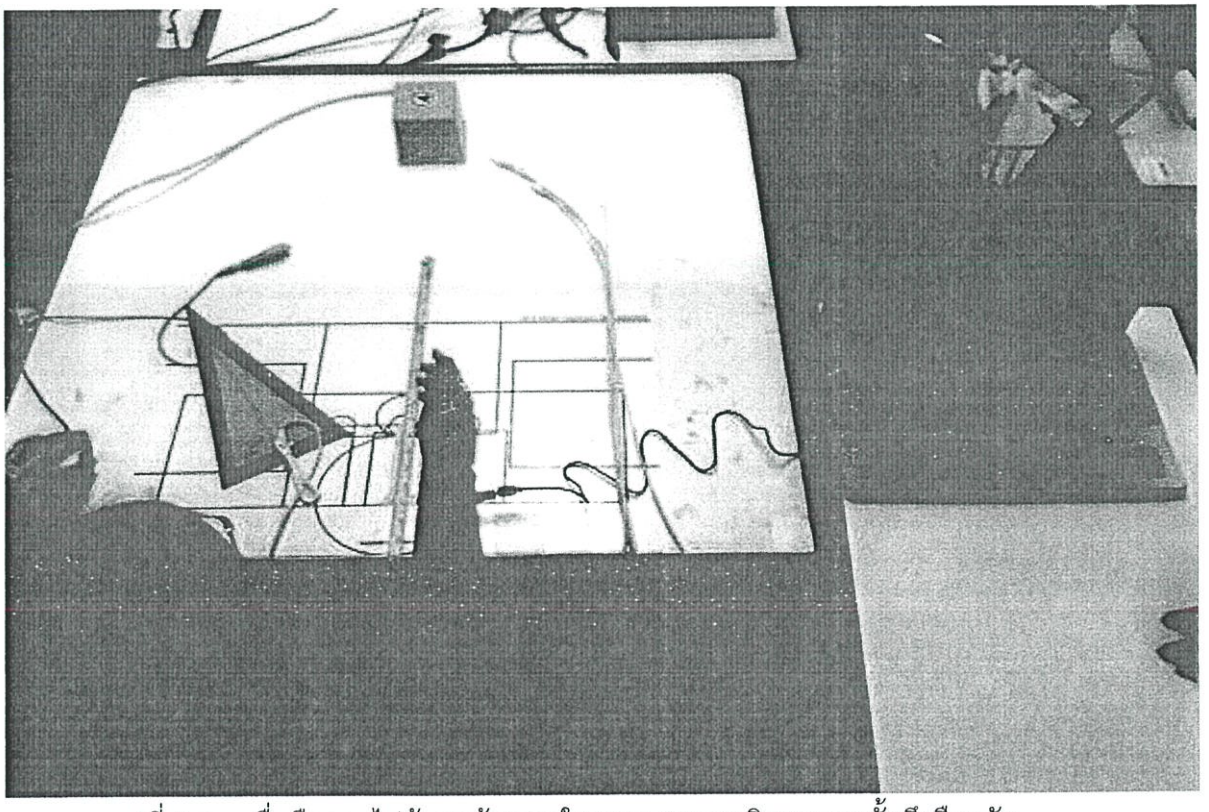
โดยที่การทดลองโดยการติดตัวIMU และ sensor ของเครื่องไว้ที่จุดเดียวกันบนข้อมือผู้ทดลองดังรูปที่ 3.5 และมีการเคลื่อนไหว 2 แบบคือ

1.การยื่นมือออกไปด้านหน้าตรง ๆ ในระยะ 30 เซนติเมตร จากนั้นดึงมือกลับ

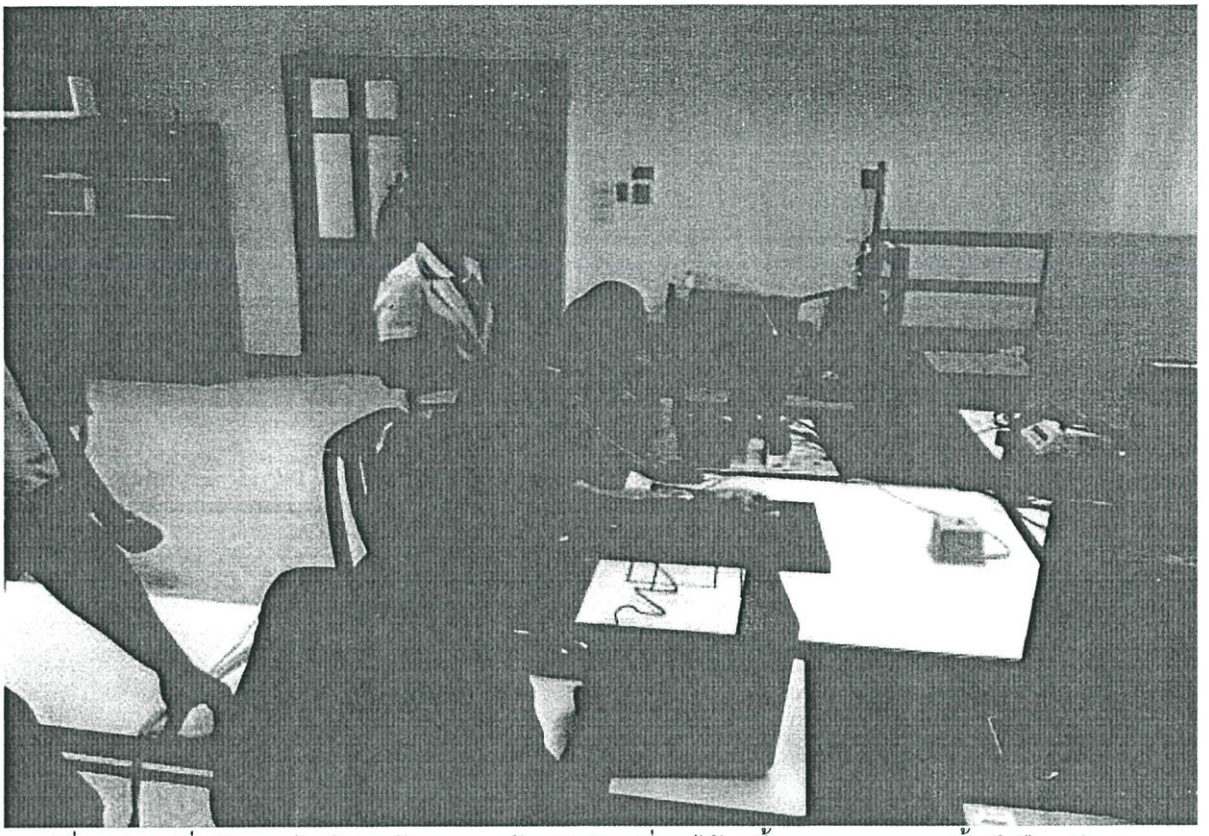
2.การยื่นมือออกไปข้างหน้าตรง ๆ หยิบวัตถุที่วางไว้ยกขึ้นและวางลงจากนั้นดึงมือกลับ



รูปที่ 3.5 การติด sensor ตรวจวัด



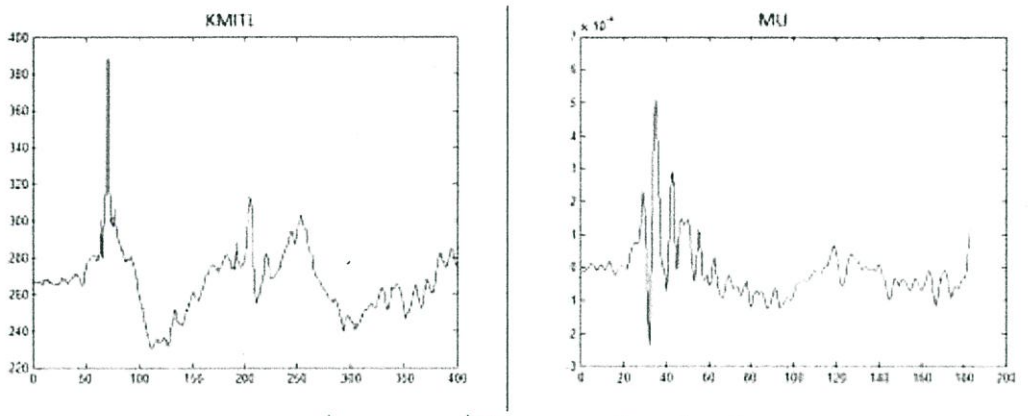
รูปที่ 3.6การยื่นมือออกไปด้านหน้าตรงๆในระยะ 30 เซนติเมตรจากนั้นดึงมือกลับ



รูปที่ 3.7 การยื่นมือออกไปข้างหน้าตรงๆ แล้วหยิบวัตถุที่วางไว้ยกขึ้นและวางลงจากนั้นดึงมือกลับ

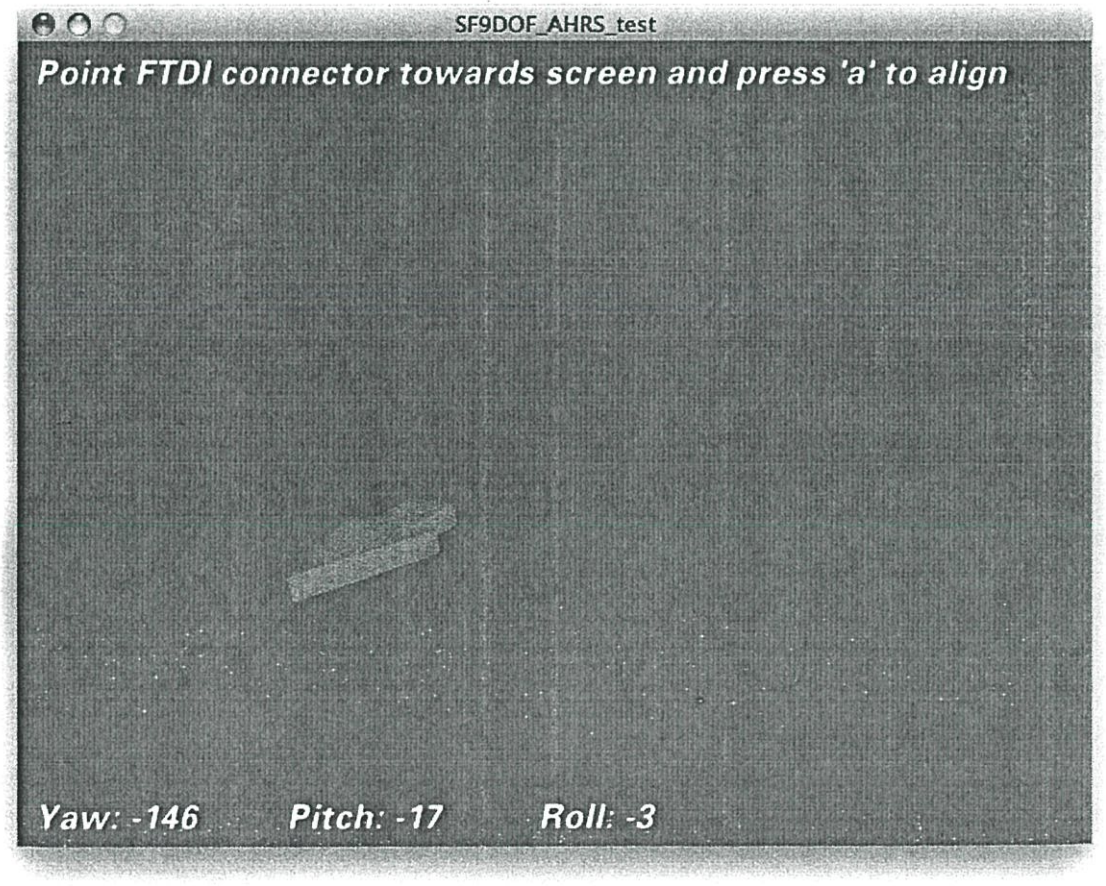
(7)นำผลที่ได้จากการทดลองมาทำการคำนวณ โดยค่าที่ได้จาก IMU เป็นค่าของความเร่งแต่ค่าที่ได้จากเครื่องที่ใช้จริงเป็นค่าของระยะทาง จึงต้องทำการแปลงค่าให้เป็นหน่วยเดียวกัน โดยทำการอนุพันธ์(Derivertives)กับค่าที่ได้จากเครื่องจริง 2 ครั้ง ให้ค่าจากระยะทาง(เมตร) เป็นค่าความเร่ง(เมตร/วินาที^{2})

(8)นำค่าที่ได้มาวาดกราฟเพื่อเปรียบเทียบกัน



รูปที่ 3.8 กราฟที่ได้จากการเปรียบเทียบกัน

(9) ทดลองใช้ใช้งานโค้ดในการวาดIMUขึ้นในโปรแกรม processing เพื่อติดตามการเคลื่อนไหวของIMU



รูปที่ 3.9ภาพที่ได้จากโปรแกรม Processing

3.2แผนการดำเนินงาน

หัวข้อ	เดือน			
	สิงหาคม	กันยายน	ตุลาคม	พฤศจิกายน
ศึกษาข้อมูลและหาอุปกรณ์ที่ใช้ในการวัดค่าความเร่ง				
สั่งซื้ออุปกรณ์ที่ใช้				
เชื่อมต่อ IMU เข้ากับ port ของคอมพิวเตอร์				
ศึกษา code ต้นแบบของ IMU พร้อมทดลองใช้งานในโปรแกรม Arduino				
นำ code ตัวอย่างมาปรับแก้เพื่อแสดงค่าในส่วนของ Accelerometer ทั้ง 3 แกน (x, y, z)				
ทดลองใช้งานโค้ดในการวาด IMU ขึ้นในโปรแกรม processing เพื่อติดตามการเคลื่อนไหวของIMU				
หาโปรแกรมและเขียนคำสั่งเพื่อทำแอปพลิเคชั่นที่แสดงกราฟของทั้ง 3 แกนของ IMU				
หัวข้อ	เดือน			
	ธันวาคม	มกราคม	กุมภาพันธุ์	มีนาคม
ปรับแต่งตัวโปรแกรมแอปพลิเคชั่น				
คำนวรและปรับแก้ค่าของข้อมูลให้ได้ตัวข้อมูลที่สามารถเข้าใจได้ง่าย				
นำชิ้นงานที่เสร็จเรียบร้อยออกเก็บข้อมูล				
นำข้อมูลที่ได้จากการทดลองมาวิเคราะห์				

ตารางที่ 3.1 ตารางแผนการดำเนินงาน

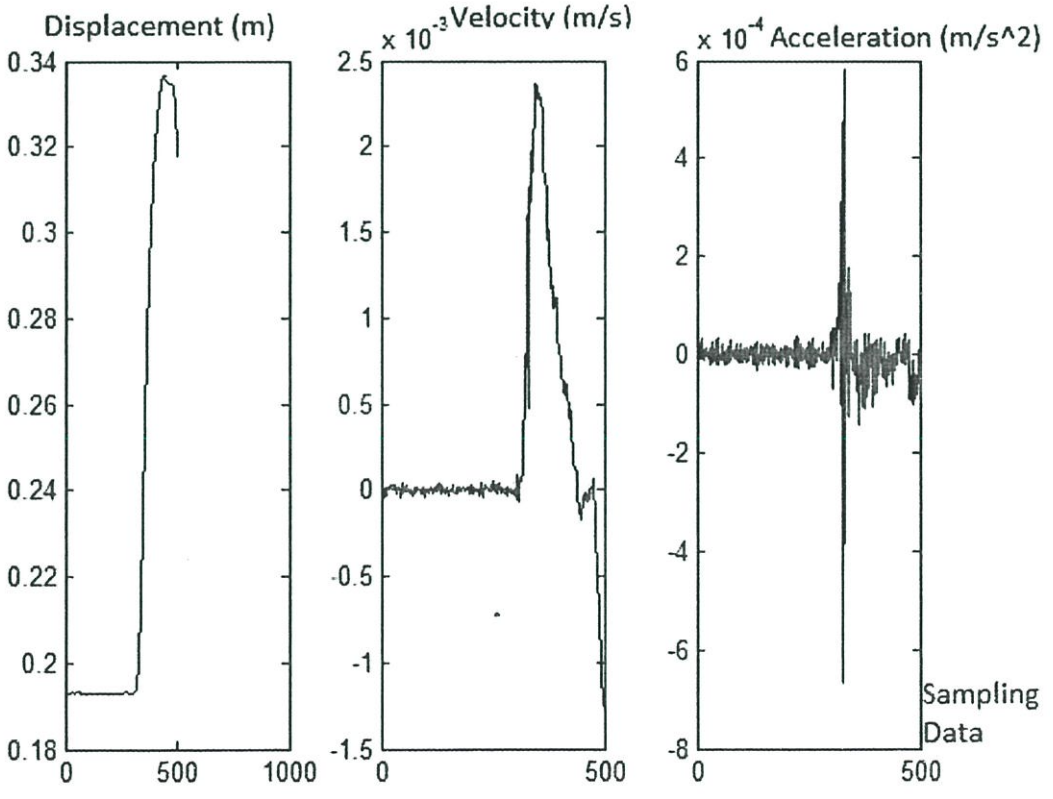
บทที่ 4

ผลการทดลอง

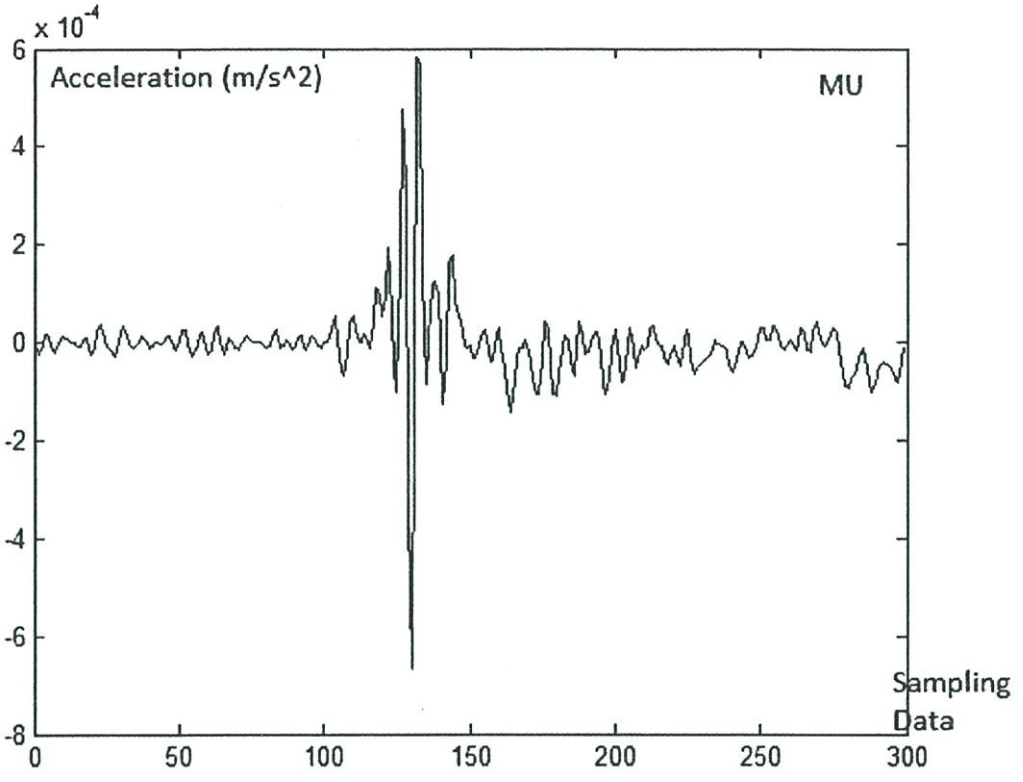
การวิเคราะห์ข้อมูลสาหรับงานวิจัยเรื่อง “การวัดการเคลื่อนไหวของร่างกายโดยวิธีแบบไม่ใช้การมองเห็น” ผู้วิจัยได้เก็บข้อมูลตัวอย่างโดยนำเอาตัว IMU มาติดเข้ากับข้อมือของผู้ทดลองเพื่อเปรียบเทียบผลกับเครื่องที่ใช้อยู่จริงที่ติดอยู่ในตำแหน่งเดียวกันโดยมีการออกแบ่งการทดลองออกเป็น 2 แบบคือ แบบที่ให้ผู้ทดลองยื่นมือไปข้างหน้าเป็นระยะทาง 30 เซนติเมตรโดยวัดผลขณะเคลื่อนไหวไปด้วยจำนวน 5 รอบ และ แบบที่ให้ผู้ทดลองยื่นมือไปข้างหน้าระยะ 30 เซนติเมตรแล้วจับวัตถุที่ตั้งไว้จากนั้นยกขึ้นแล้ววางลงโดยวัดผลขณะเคลื่อนไหวไปด้วยเช่นกันจำนวน 5 รอบ และนำผลข้อมูลที่ได้เป็นตัวเลขมาวาดกราฟโดยนำค่าข้อมูลของทั้ง3แกนมารวมเวคเตอร์กันให้ได้ค่าตัวเลขเพียงค่าเดียวและวาดกราฟเพื่อความสะดวกในการเปรียบเทียบโดยผลการทดลองที่ได้ออกมาดังนี้

4.1 การทดลองแบบให้ผู้ทดลองยื่นมือไปข้างหน้าเป็นระยะทาง 30 เซนติเมตรโดยวัดผลขณะเคลื่อนไหวไปด้วยจำนวน 5 รอบ

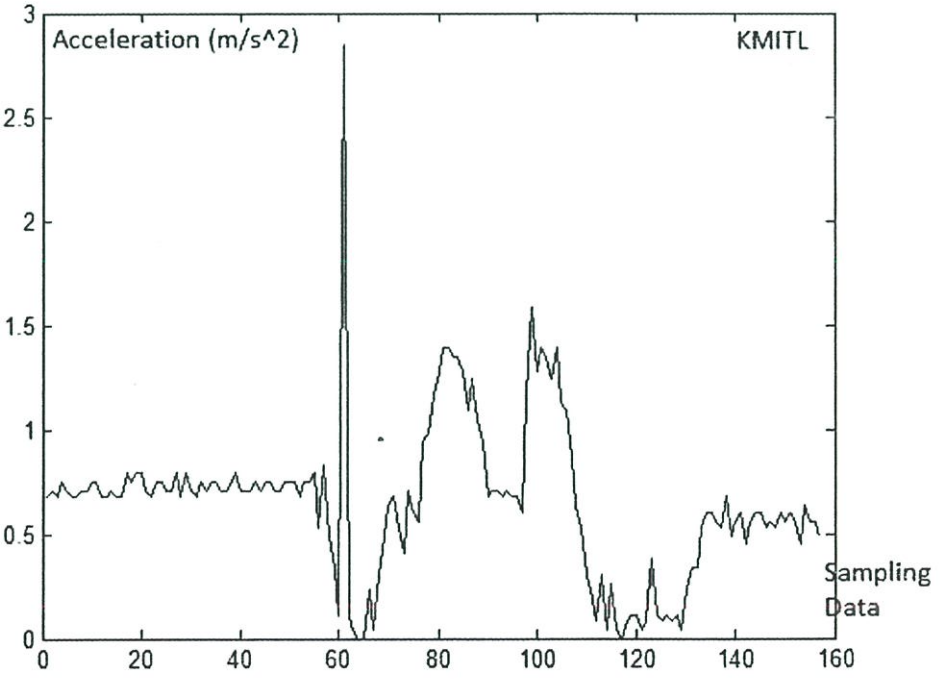
4.1.1ครั้งที่1



รูปที่ 4.1 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 1

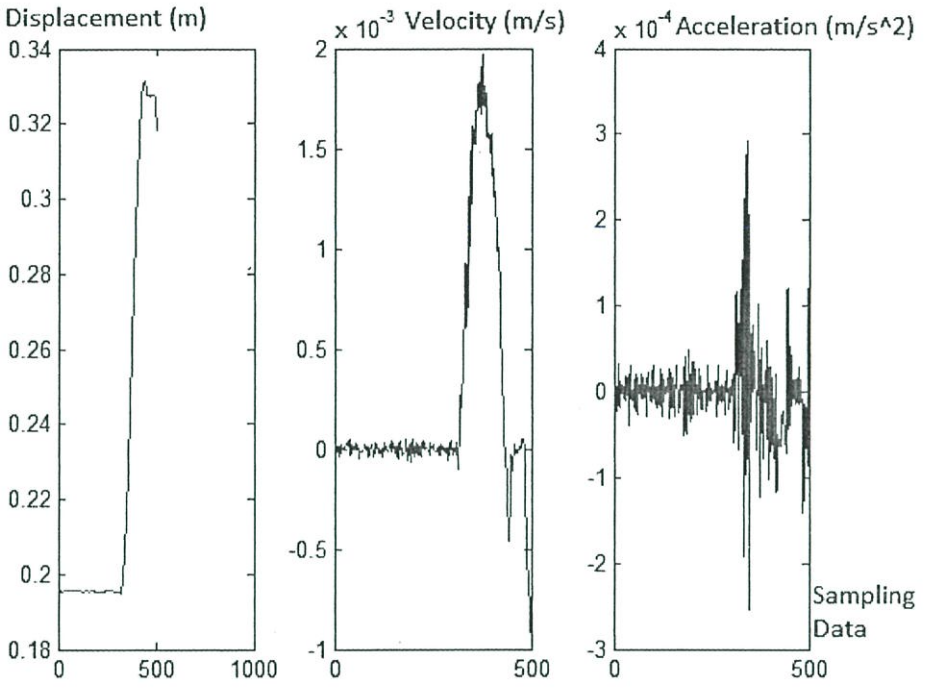


รูปที่ 4.2 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 1

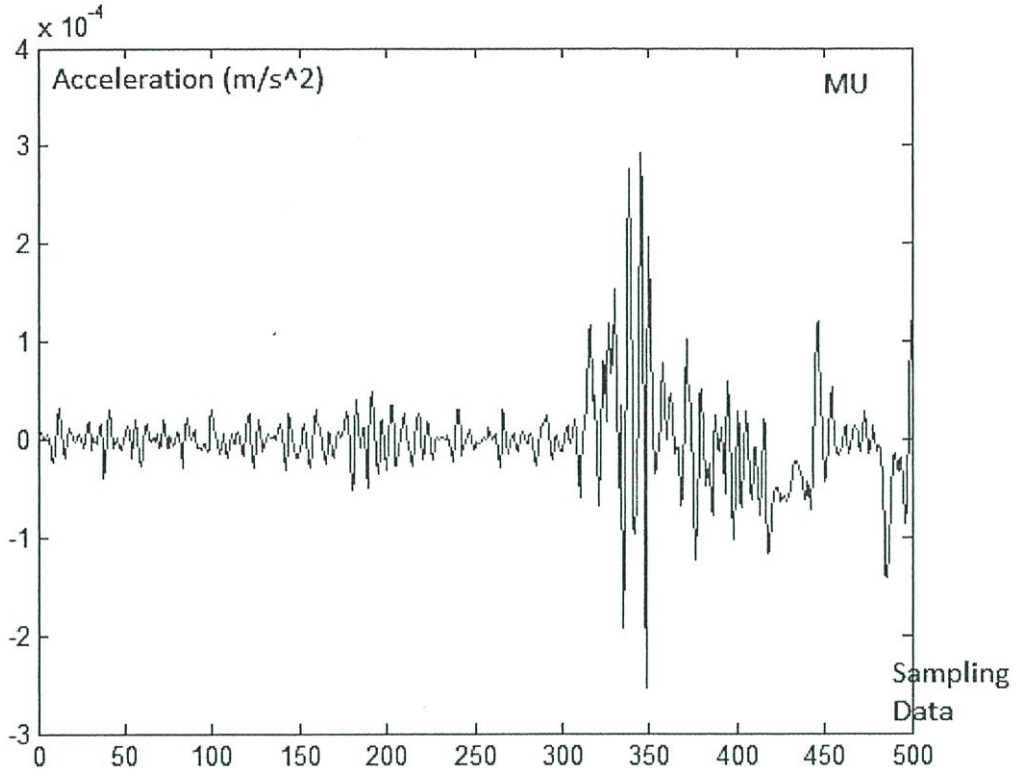


รูปที่ 4.3 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 1

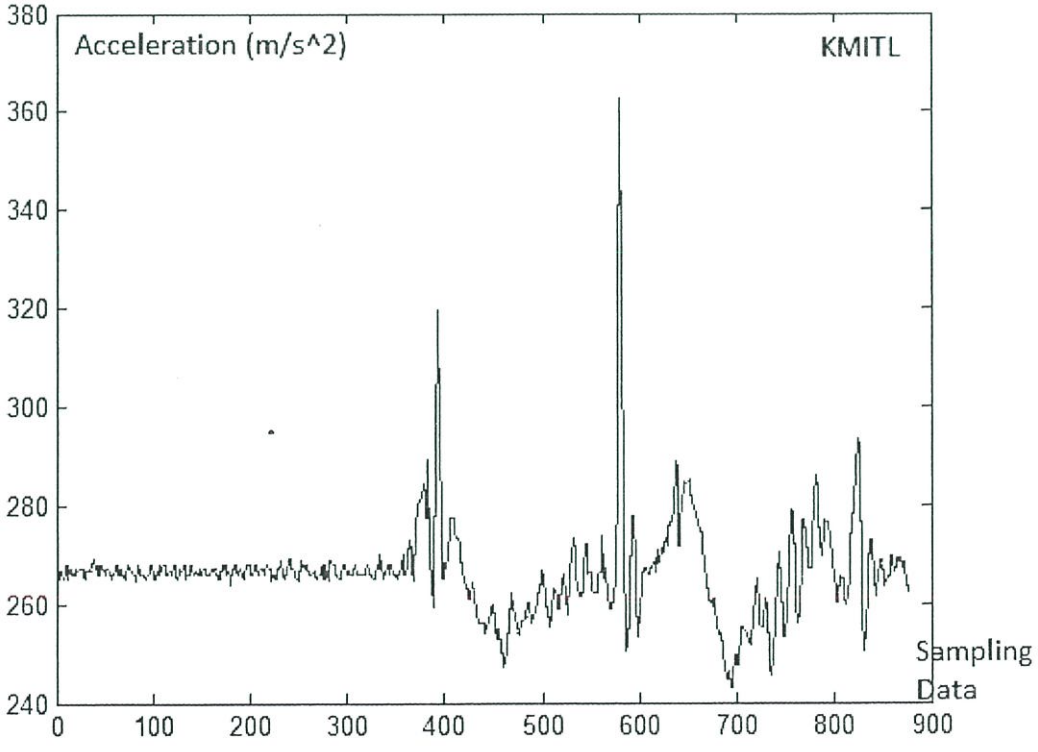
4.1.2 ครั้งที่2



รูปที่ 4.4 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 2

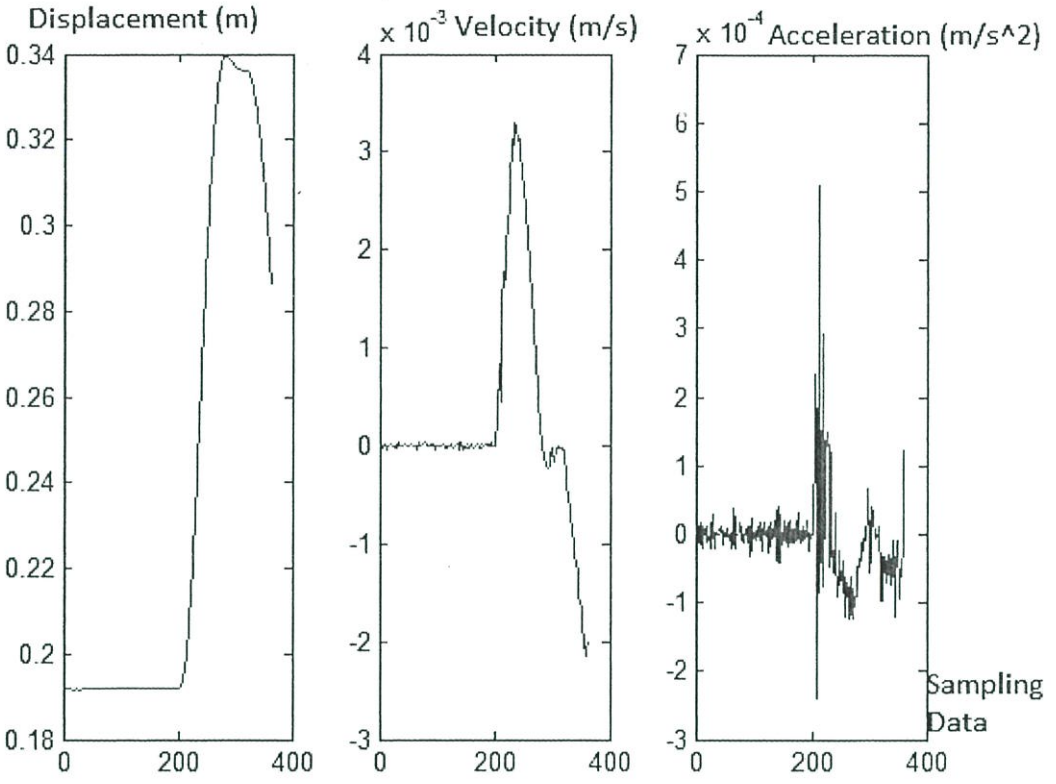


รูปที่ 4.5 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 2

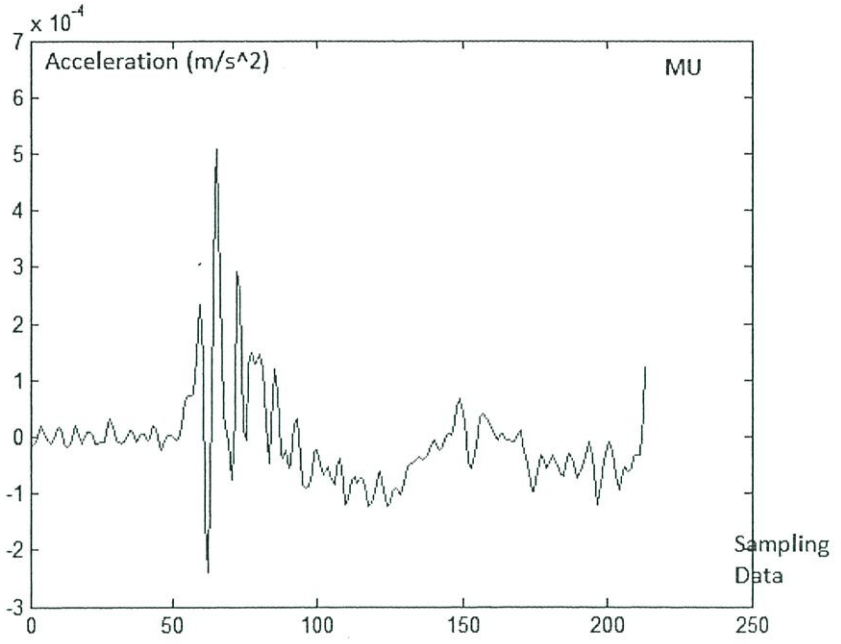


รูปที่ 4.6 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 2

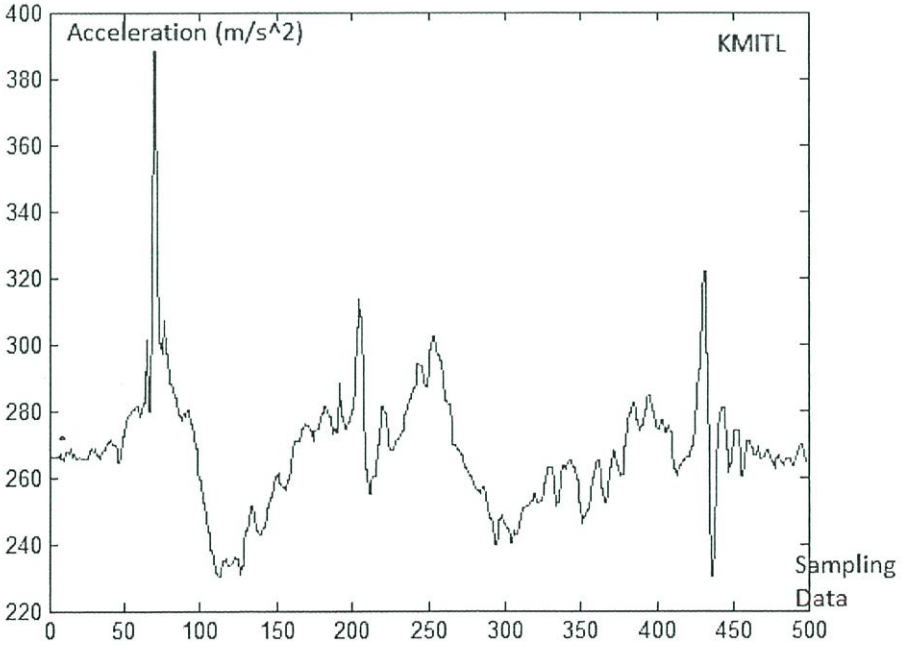
4.1.3 ครั้งที่3



รูปที่ 4.7 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 3

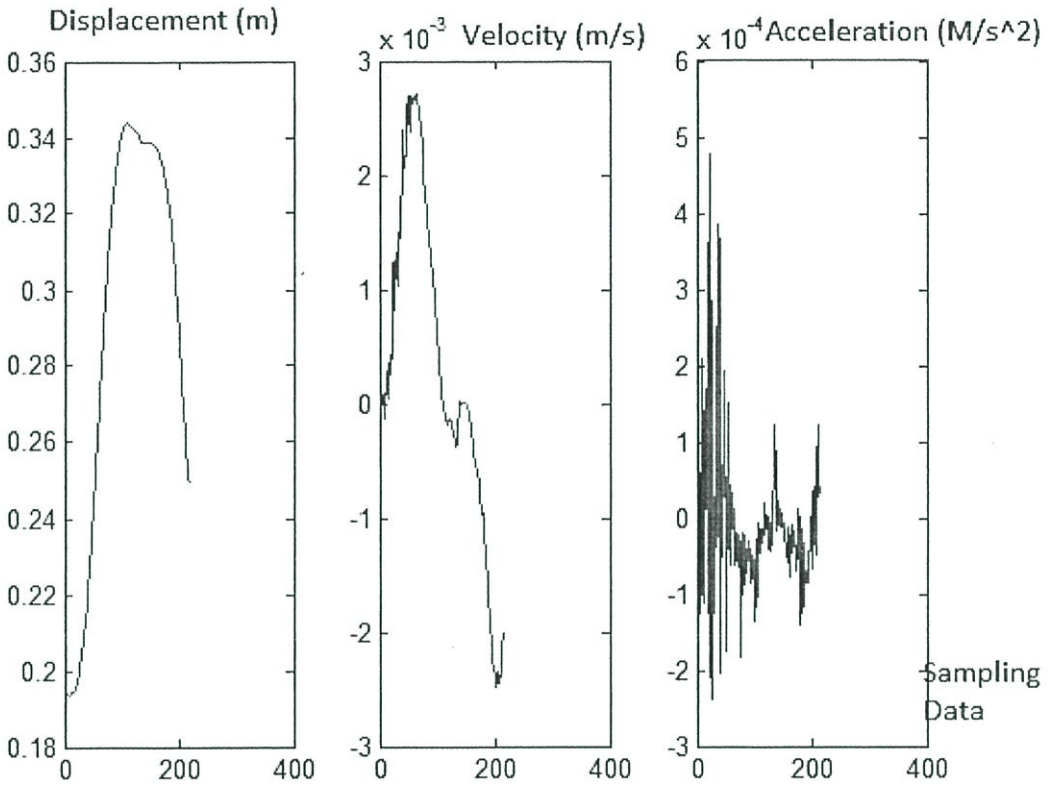


รูปที่ 4.8 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 3

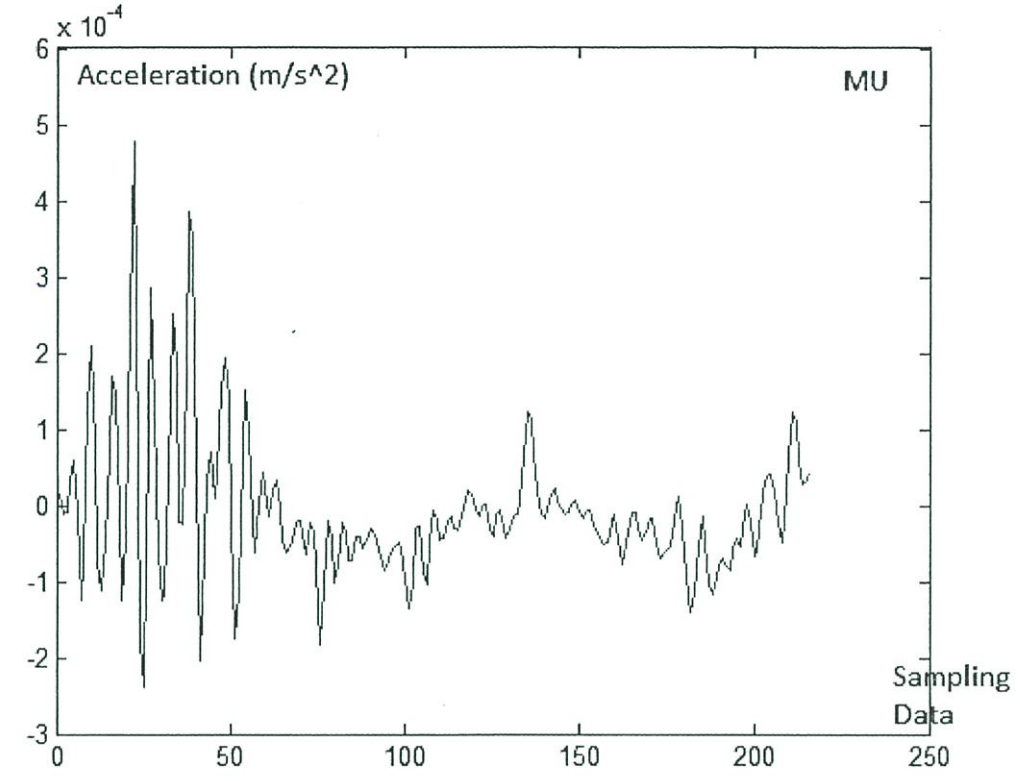


รูปที่ 4.9 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 3

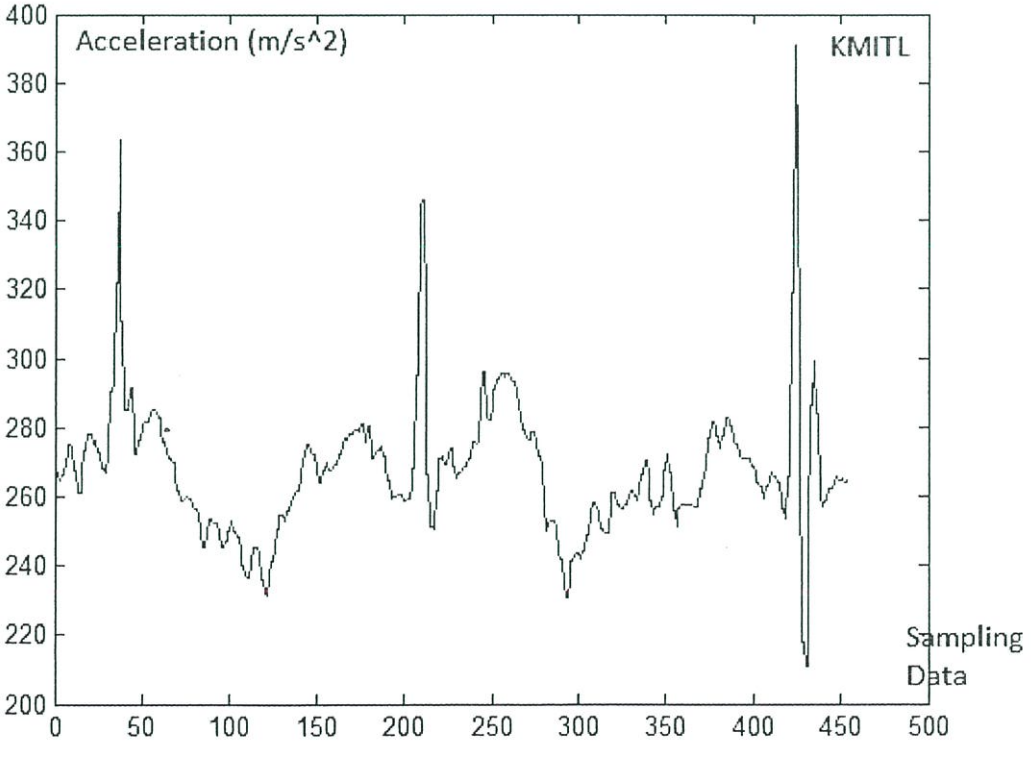
4.1.4 ครั้งที่4



รูปที่ 4.10 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 4

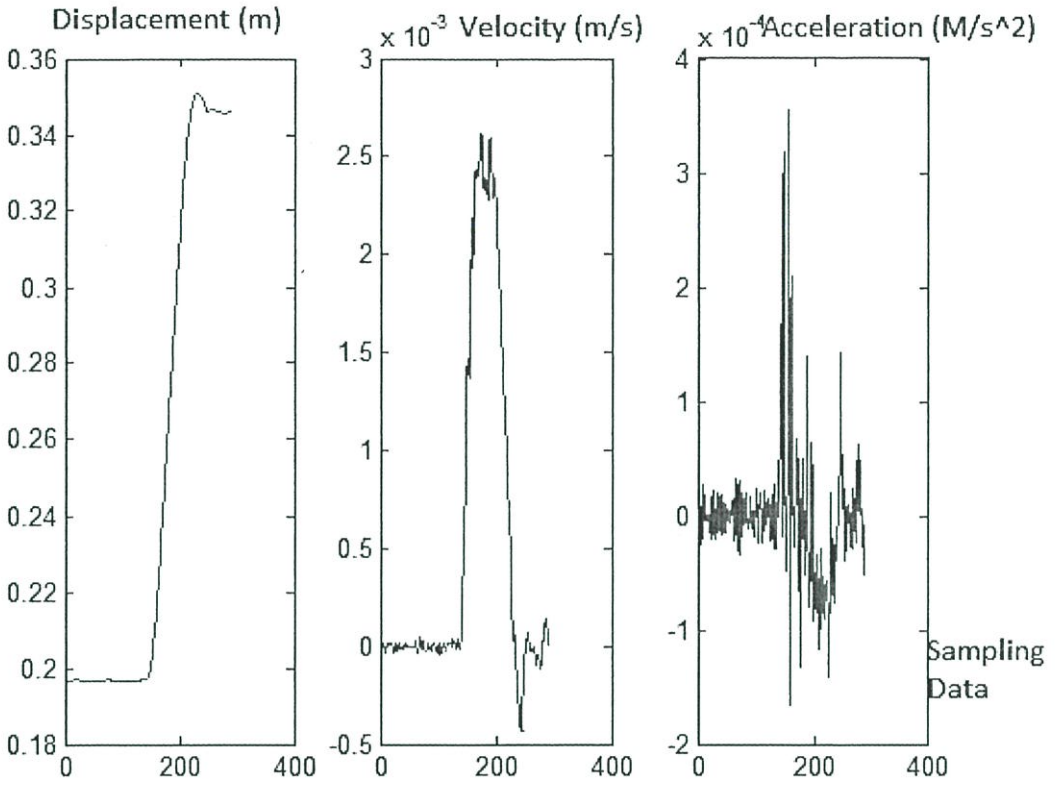


รูปที่ 4.11 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 4

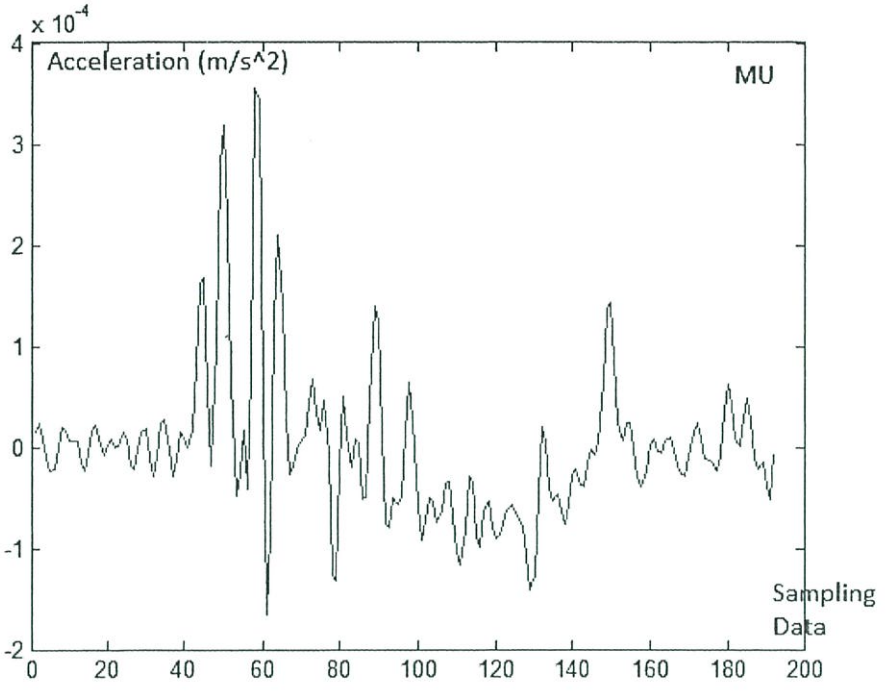


รูปที่ 4.12 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 4

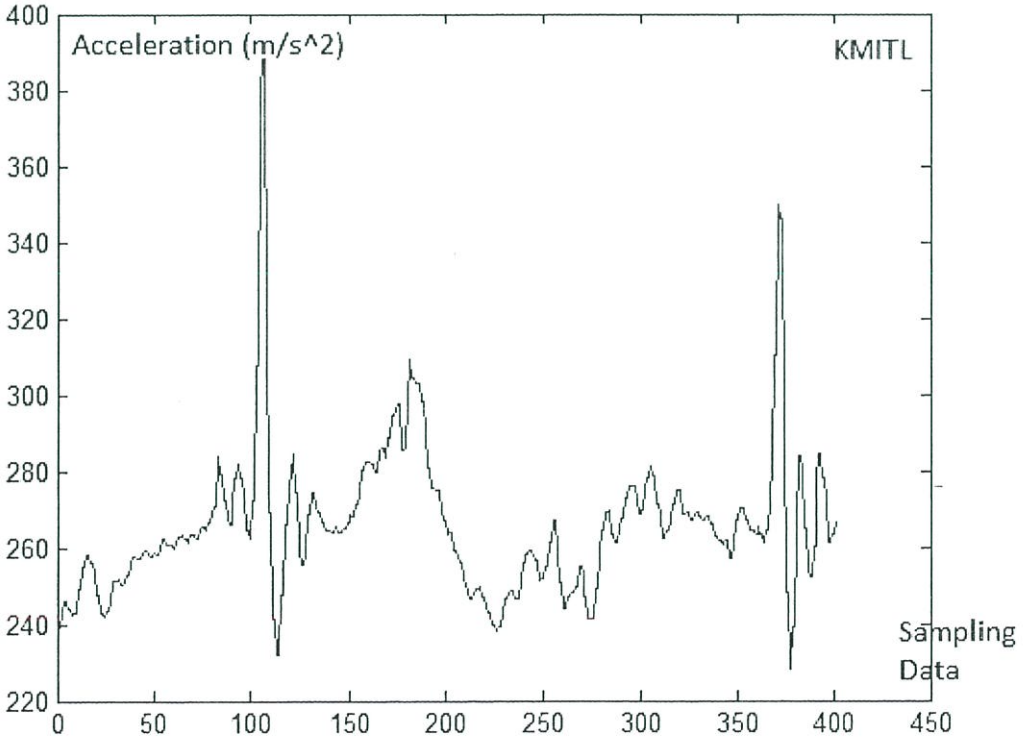
4.1.5 ครั้งที่ 5



รูปที่ 4.13 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 5



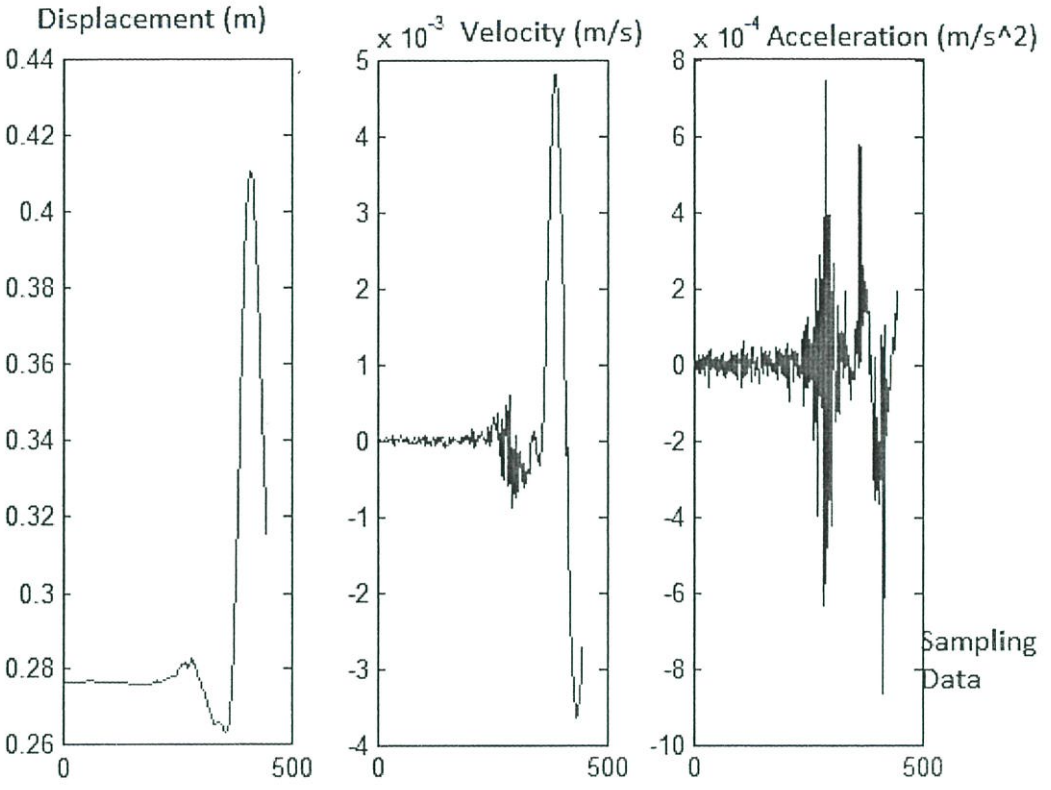
รูปที่ 4.14 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 5



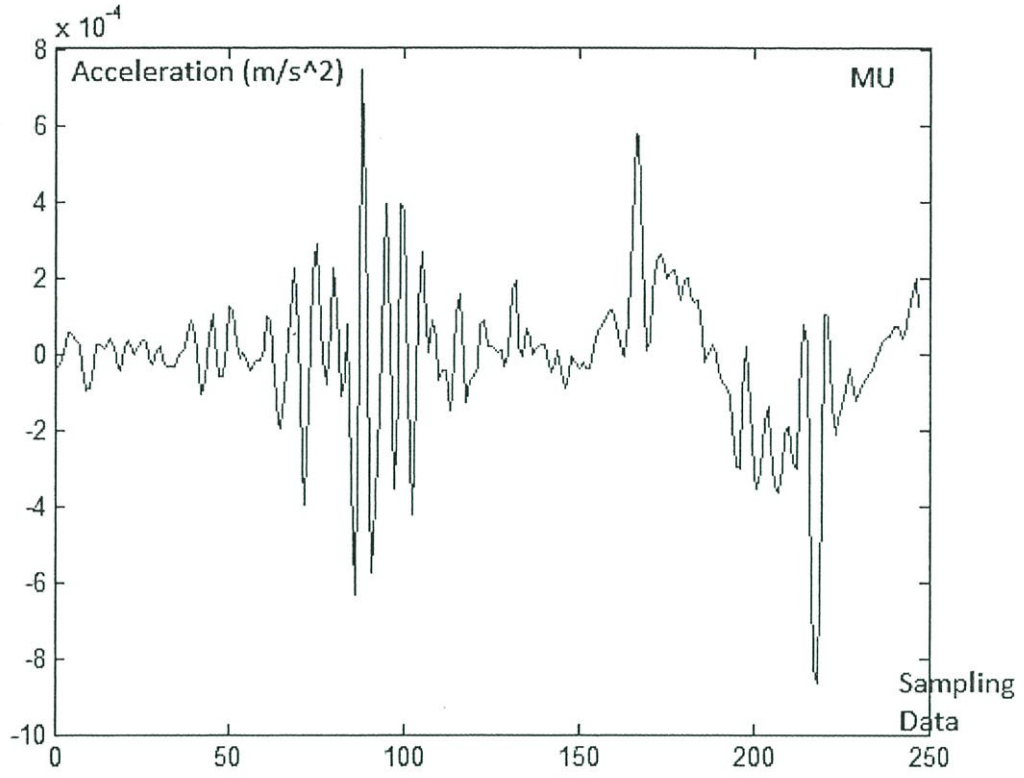
รูปที่ 4.15 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 5

4.2 การทดลองแบบให้ผู้ทดลองยื่นมือไปข้างหน้าเป็นระยะทาง 30 เซนติเมตรแล้วจับวัตถุที่ตั้งไว้ จากนั้นยกขึ้นแล้ววางลงโดยวัดผลขณะเคลื่อนไหวไปด้วยเช่นกันจำนวน 5 รอบ

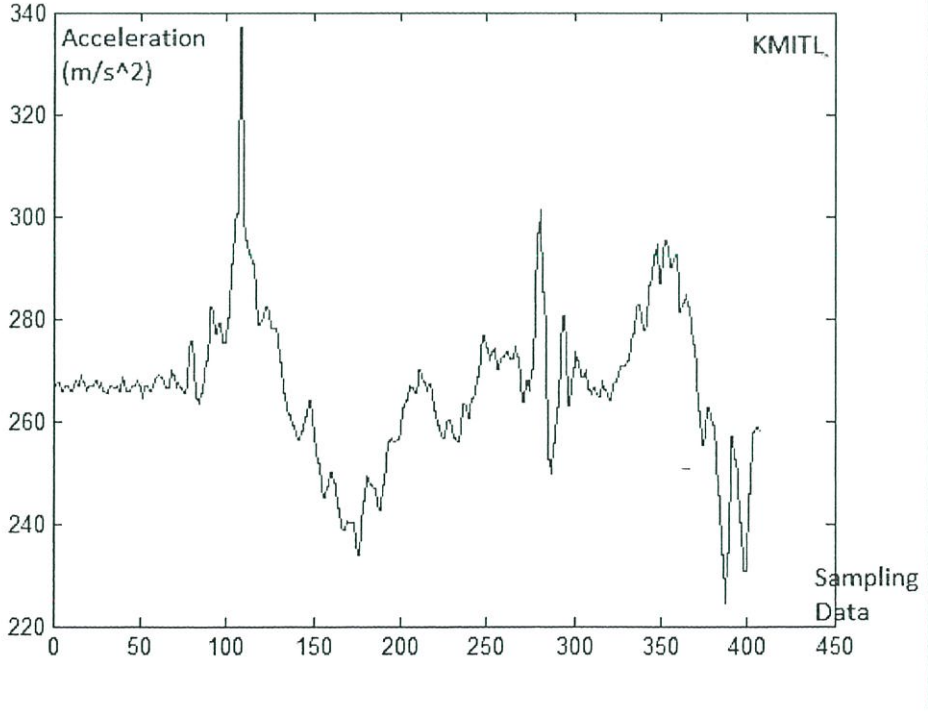
4.2.1 ครั้งที่1



รูปที่ 4.16 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 1

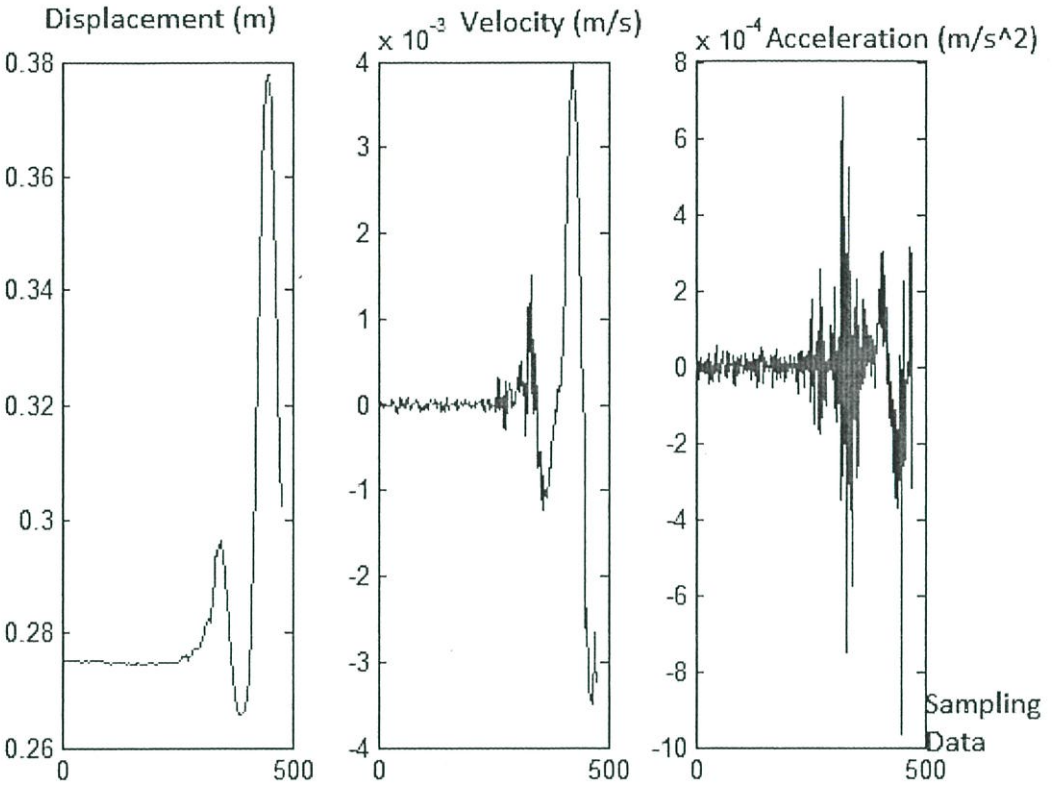


รูปที่ 4.17 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 1

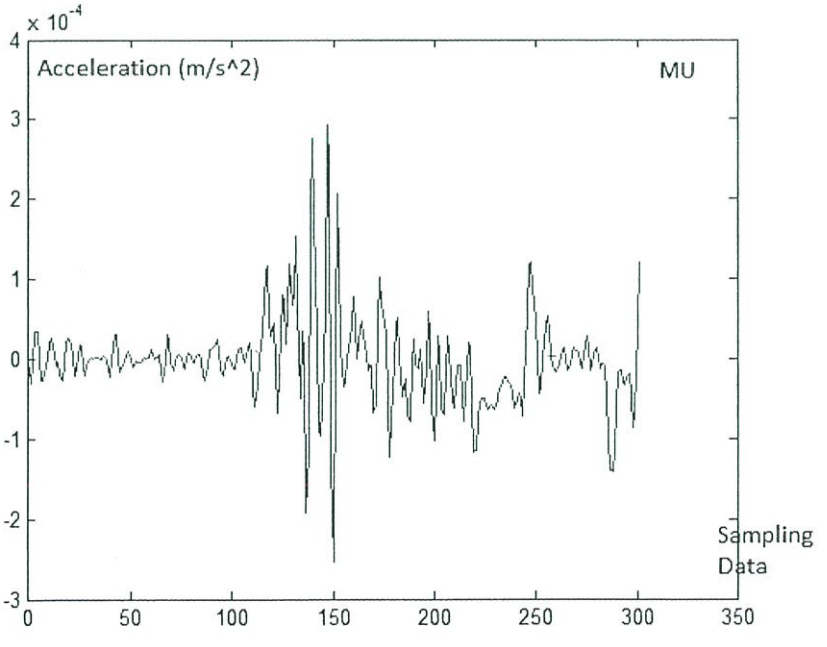


รูปที่ 4.17 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 1

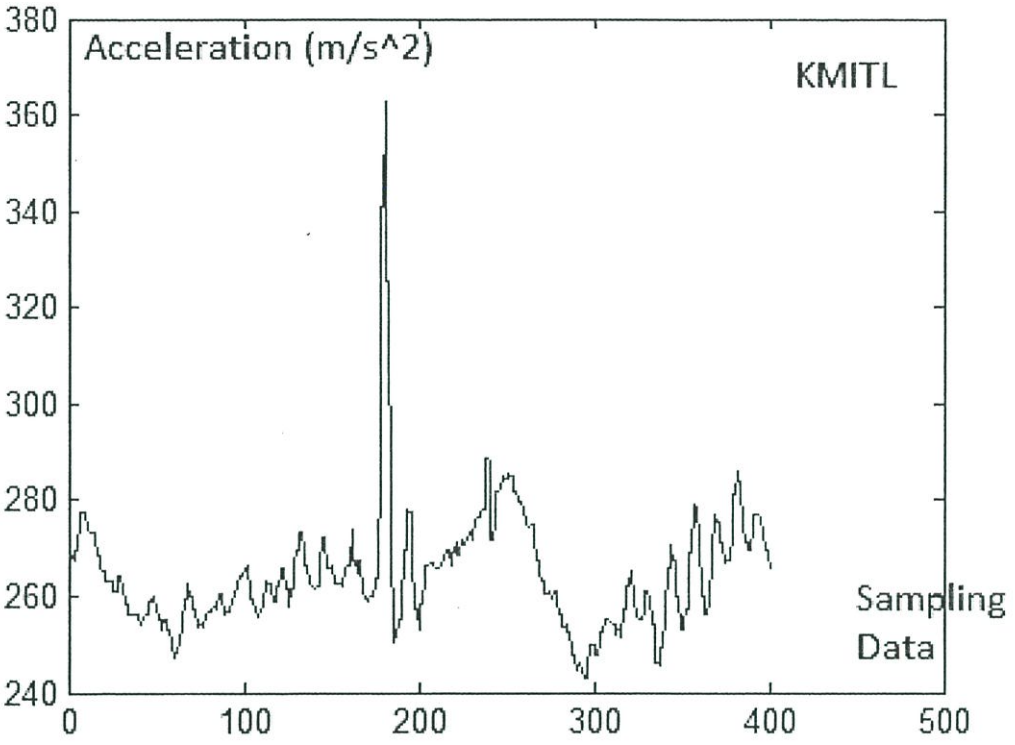
4.2.2 ครั้งที่2



รูปที่ 4.19 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 2

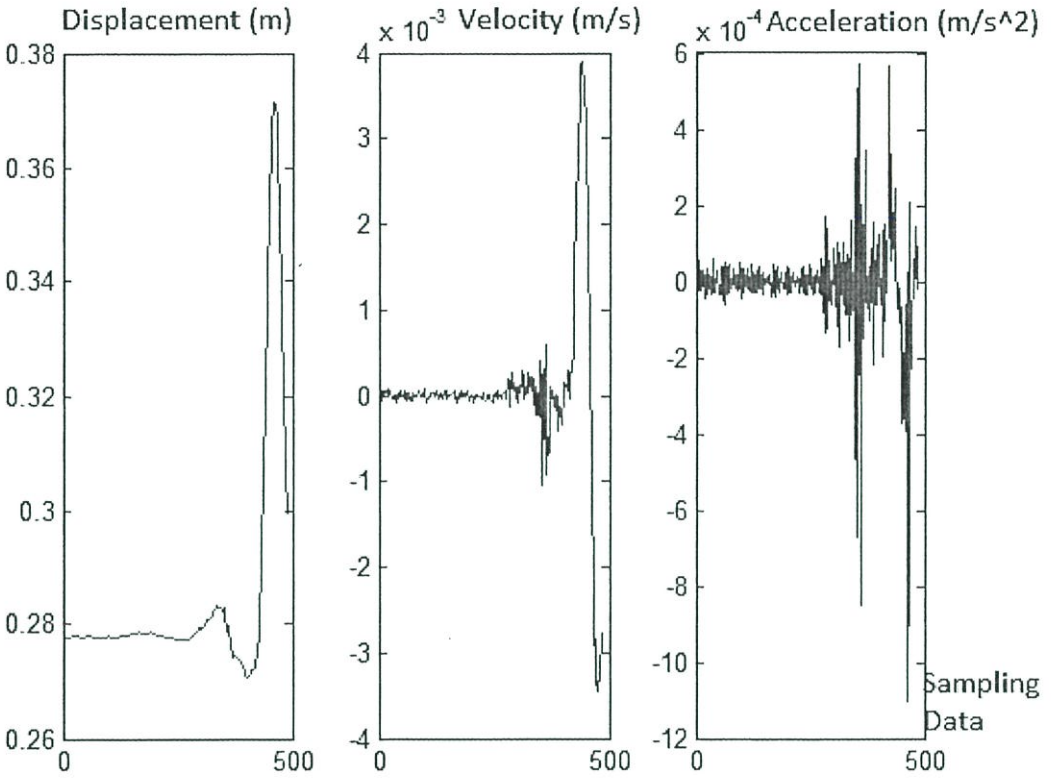


รูปที่ 4.20 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 2

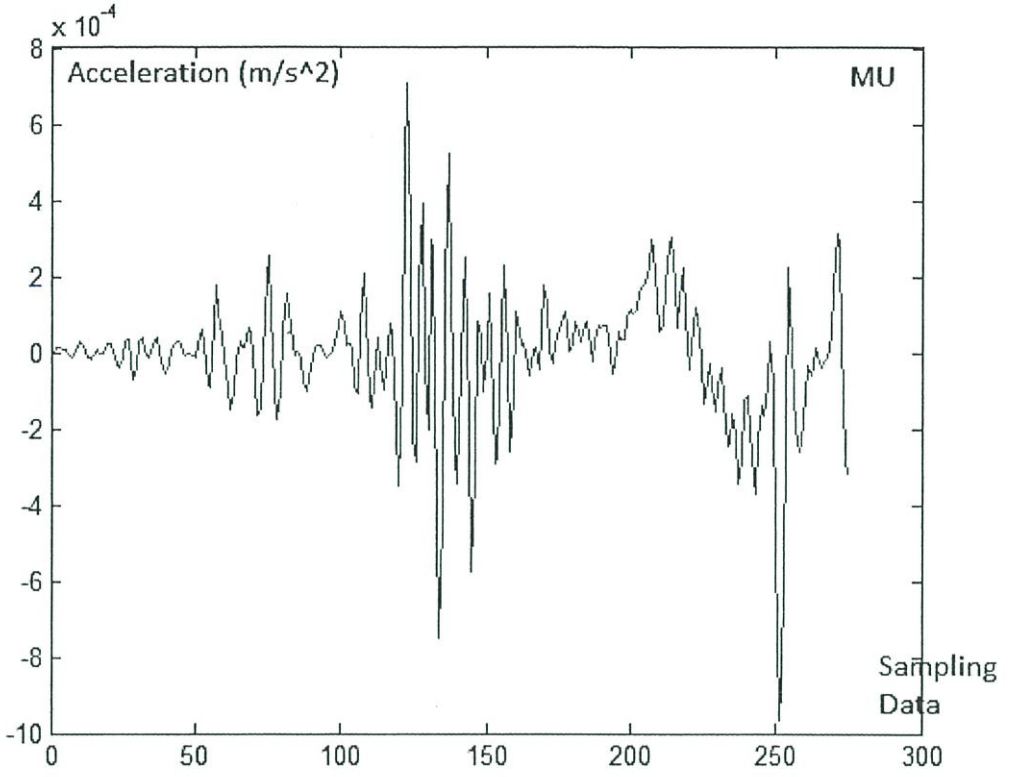


รูปที่ 4.21 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 2

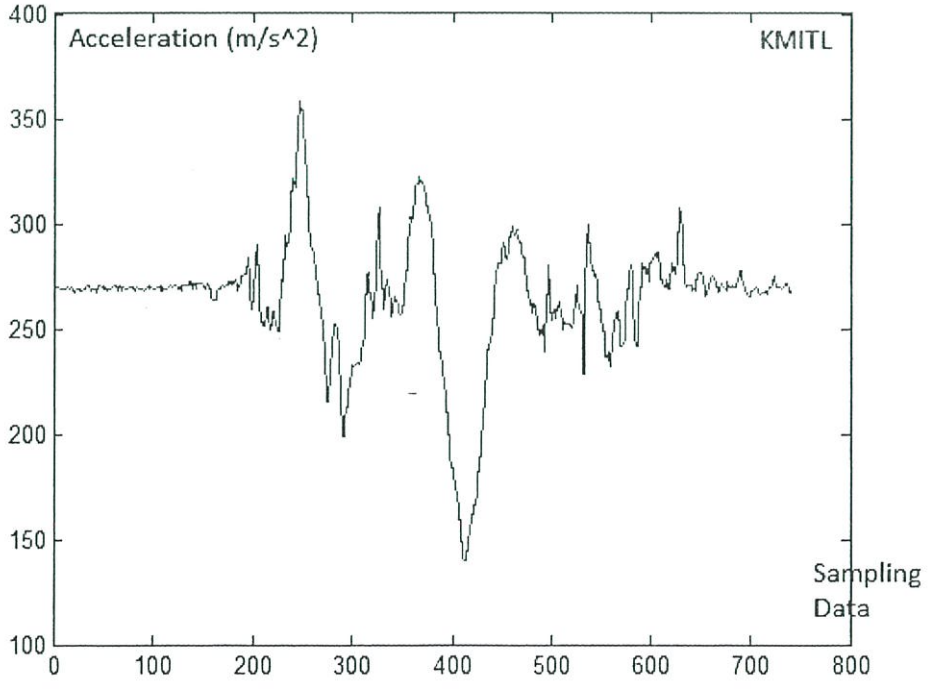
4.2.3ครั้งที่3



รูปที่ 4.22 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 3

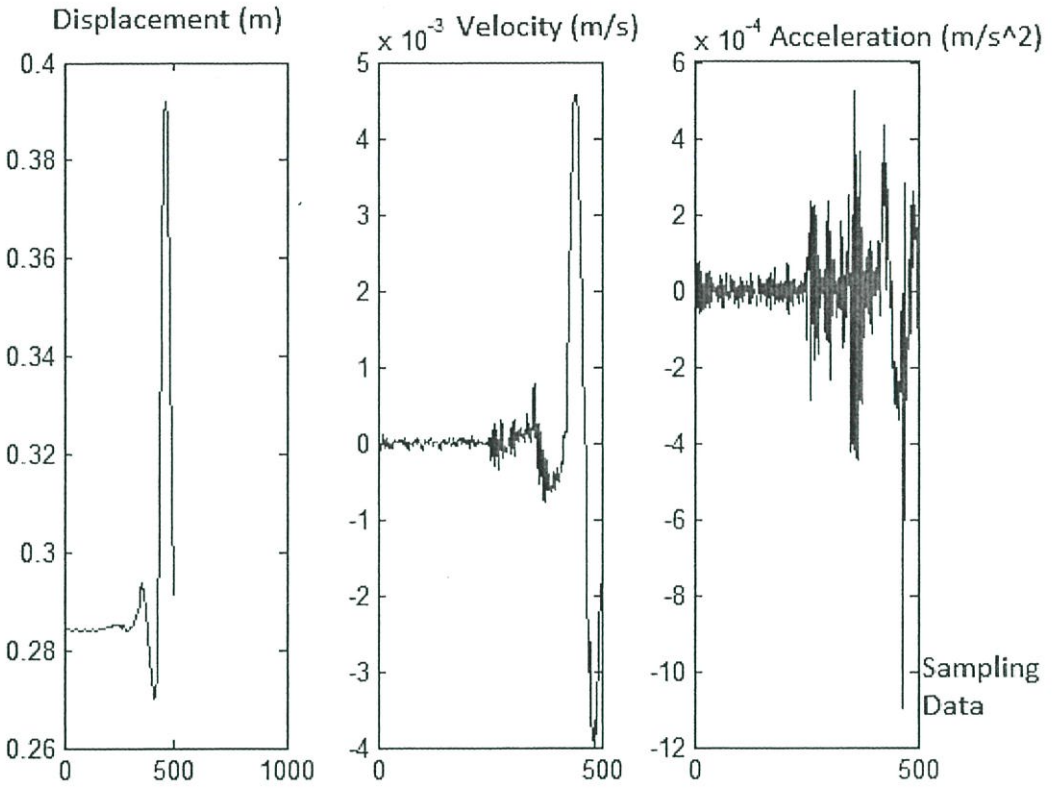


รูปที่ 4.23 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 3

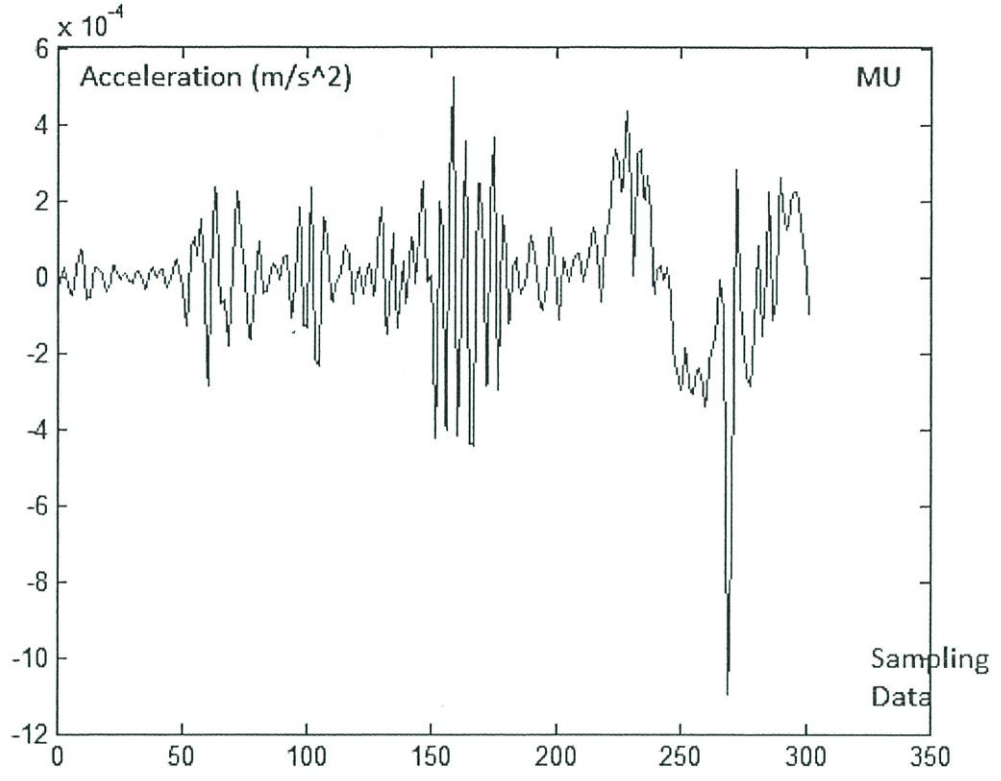


รูปที่ 4.24 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 3

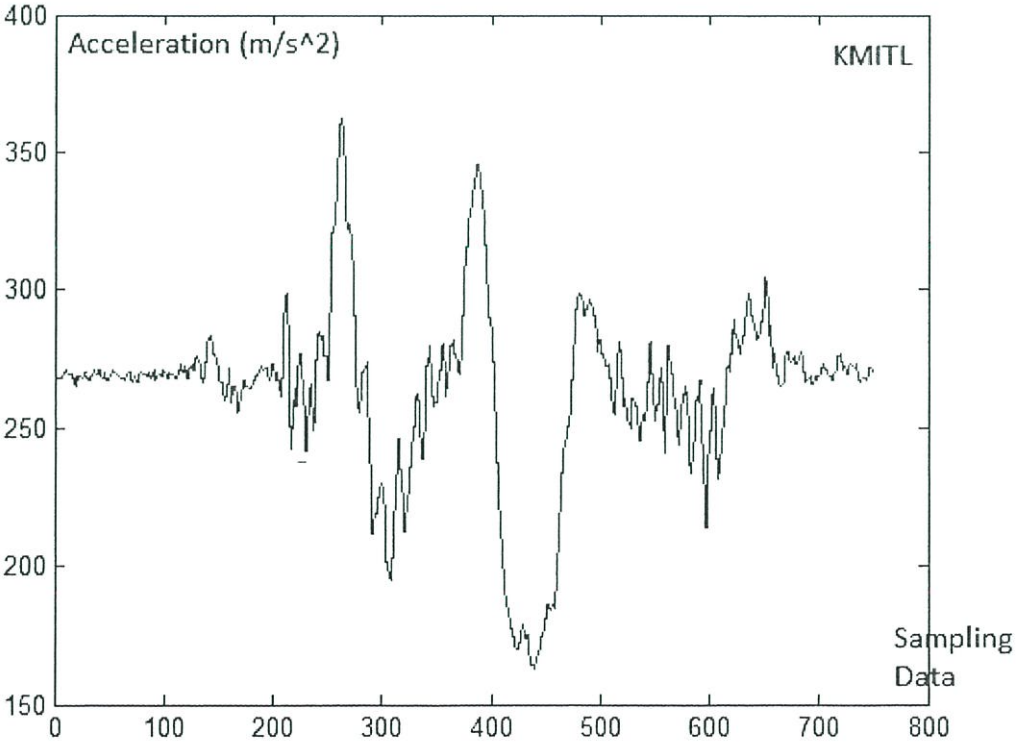
4.2.4ครั้งที่4



รูปที่ 4.25 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 4

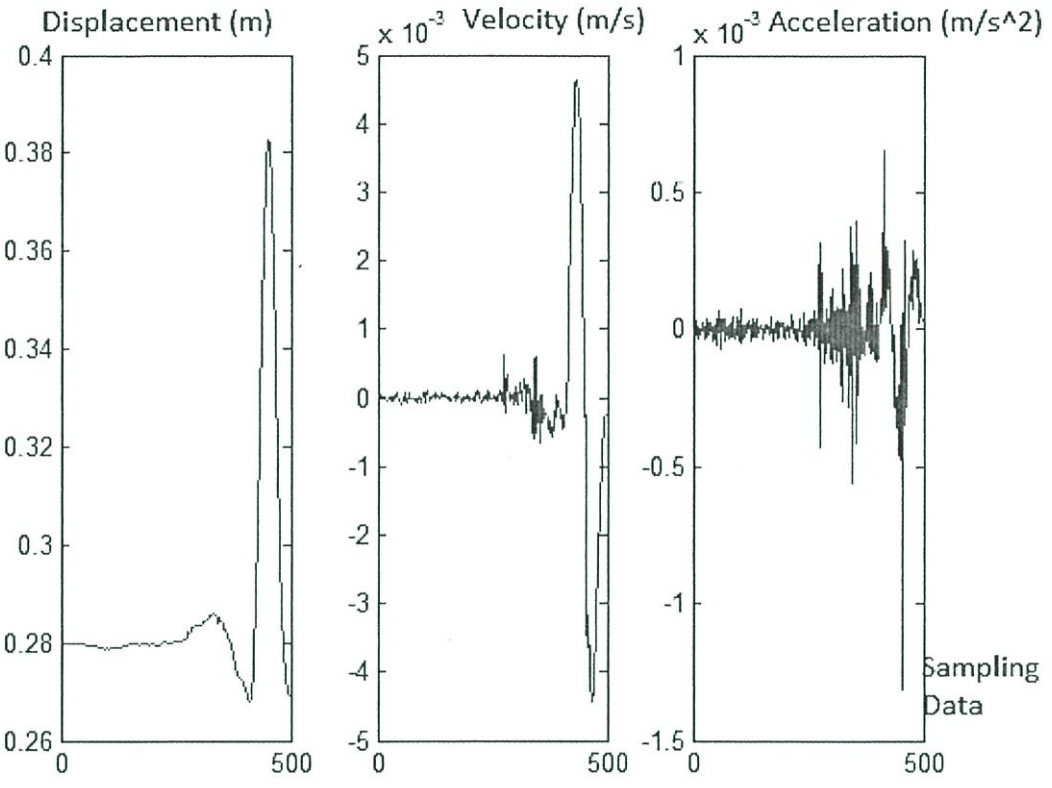


รูปที่ 4.26 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 4

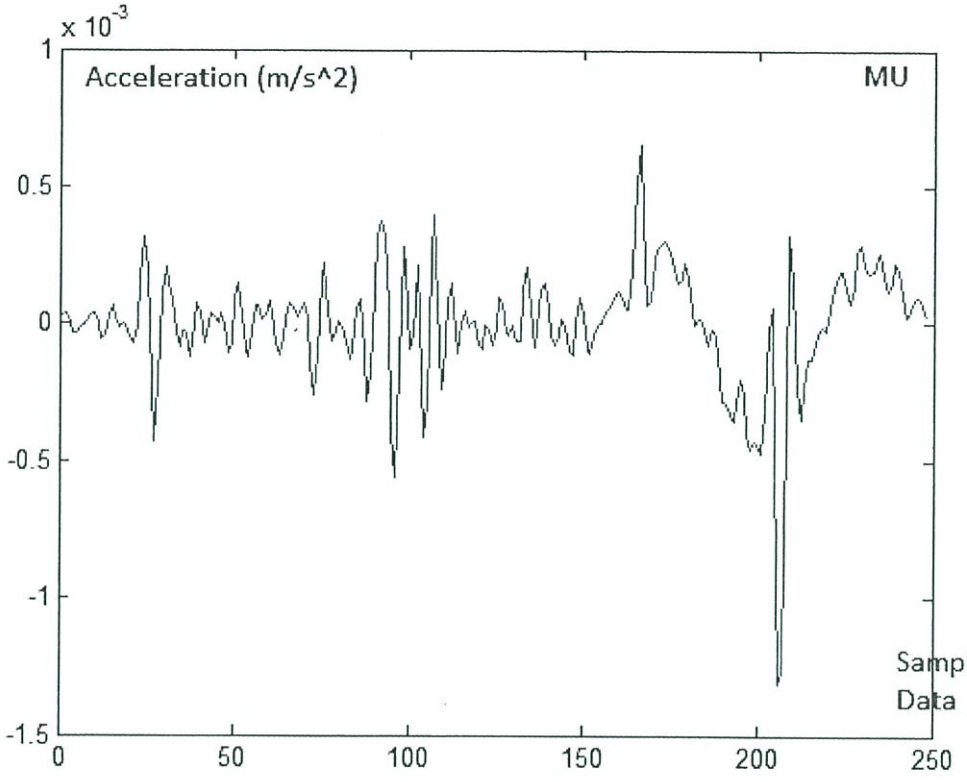


รูปที่ 4.27 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 4

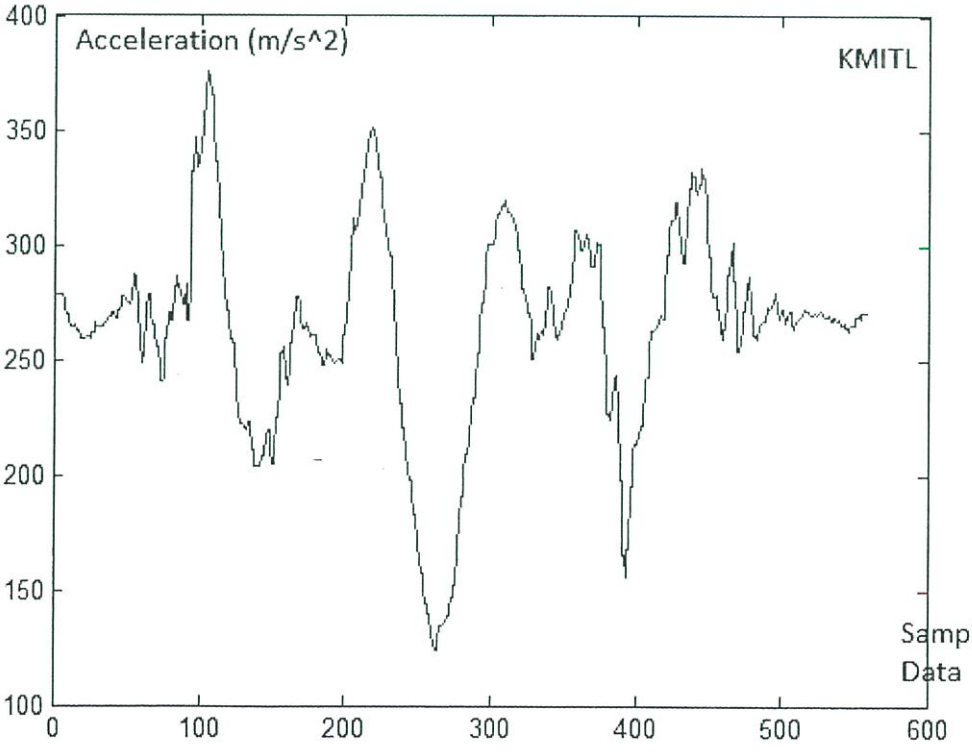
4.2.5 ครั้งที่5



รูปที่ 4.28 การแปลงหน่วยการกระจัดให้อยู่ในรูปความเร่งของการทดลองครั้งที่ 5



รูปที่ 4.29 กราฟที่ได้จากการแปลงหน่วยให้อยู่ในรูปของความเร่งของการทดลองครั้งที่ 5



รูปที่ 4.30 กราฟที่ได้จากการทดลองของผู้จัดทำครั้งที่ 5

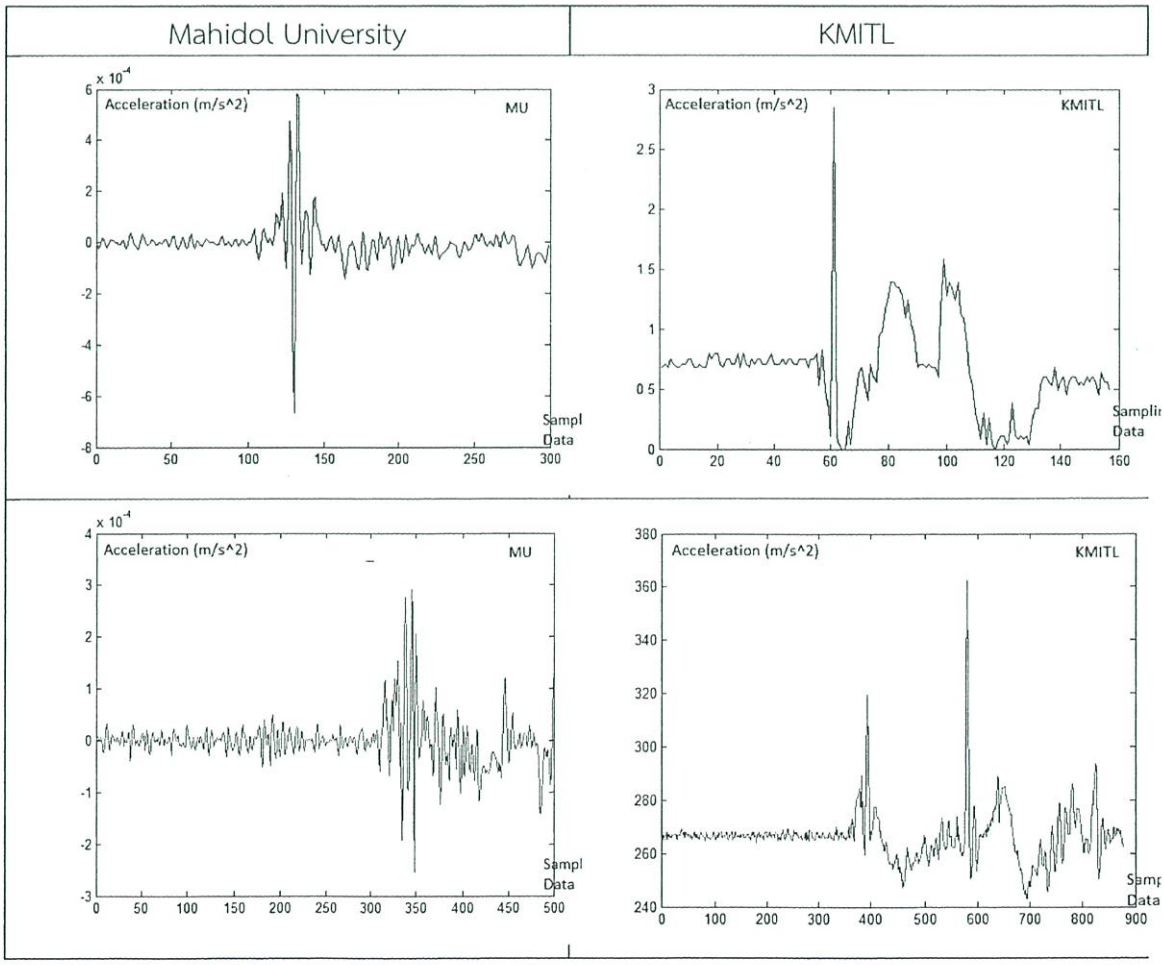
บทที่ 5

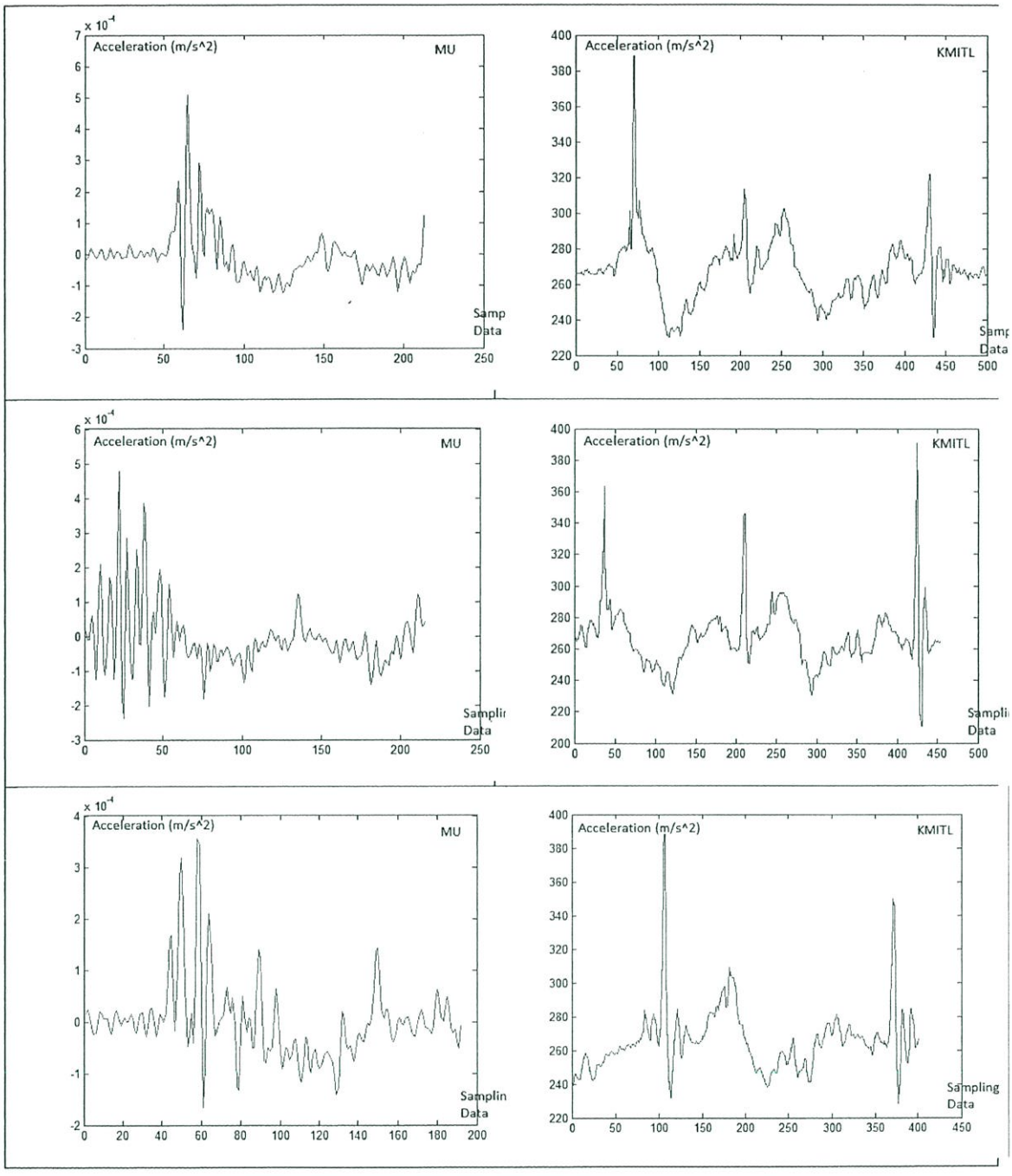
สรุปผลการทดลอง

การวิจัยครั้งนี้มีวัตถุประสงค์เพื่อพัฒนาอุปกรณ์สำหรับการวัดช่วงความเคลื่อนไหวให้สะดวกมากขึ้น โดยอุปกรณ์ที่พัฒนาขึ้นมานั้นมีค่าใช้จ่ายน้อยเมื่อเทียบกับอุปกรณ์ที่มีอยู่เดิมในห้องทดลอง นอกจากนั้นยังสามารถใช้อุปกรณ์ใหม่นี้ทำการวัดคนไข้ได้ทุกสถานที่ เนื่องจากมีขนาดเล็ก พกพาสะดวก

ในการทดลองสำหรับงานวิจัยเรื่อง "การวัดการเคลื่อนไหวของร่างกายโดยวิธีแบบไม่ใช้การมองเห็น" ผู้วิจัยได้เก็บข้อมูลตัวอย่างโดยนำเอาอุปกรณ์มาติดเข้ากับข้อมือของผู้ทดลองเพื่อเปรียบเทียบผลกับเครื่องที่ใช้อยู่จริงที่ติดอยู่ในตำแหน่งเดียวกันโดยมีการออกแบ่งการทดลองออกเป็น 2 แบบคือ แบบที่ให้ผู้ทดลองยื่นมือไปข้างหน้าเป็นระยะทาง 30 เซนติเมตรโดยวัดผลขณะเคลื่อนไหวไปด้วยจำนวน 5 รอบ และ แบบที่ให้ผู้ทดลองยื่นมือไปข้างหน้าระยะ 30 เซนติเมตรแล้วจับวัตถุที่ตั้งไว้จากนั้นยกขึ้นแล้ววางลงโดยวัดผลขณะเคลื่อนไหวไปด้วยเช่นกันจำนวน 5 รอบ และนำผลข้อมูลที่ได้เป็นตัวเลขมาวาดกราฟโดยนำค่าข้อมูลของทั้ง 3 แกนมารวมเวคเตอร์กันให้ได้ค่าตัวเลขเพียงค่าเดียวและวาดกราฟเพื่อความสะดวกในการเปรียบเทียบ โดยผลการทดลองที่ได้ออกมาดังนี้

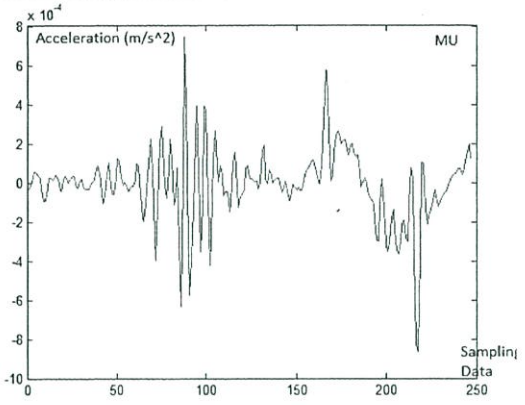
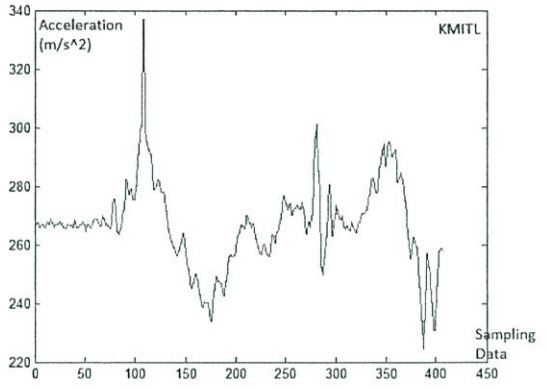
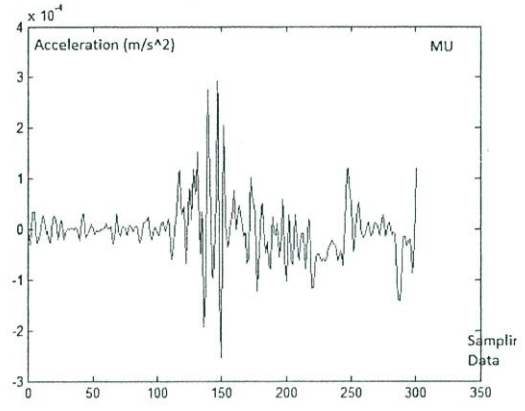
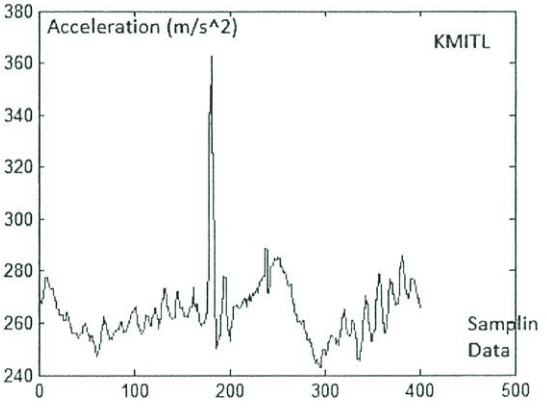
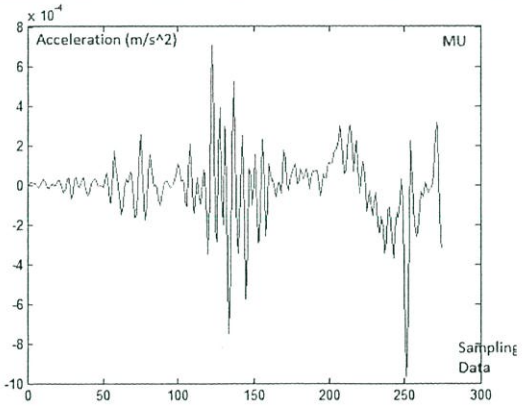
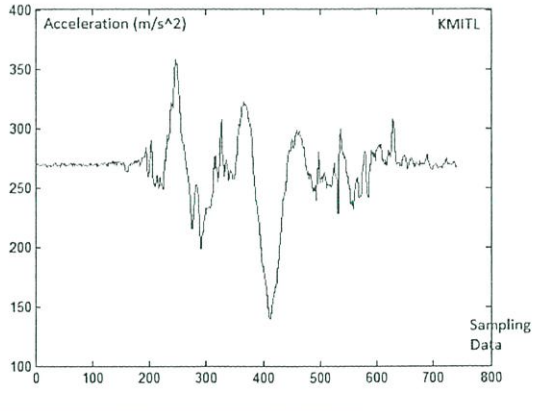
การทดลองที่ 1

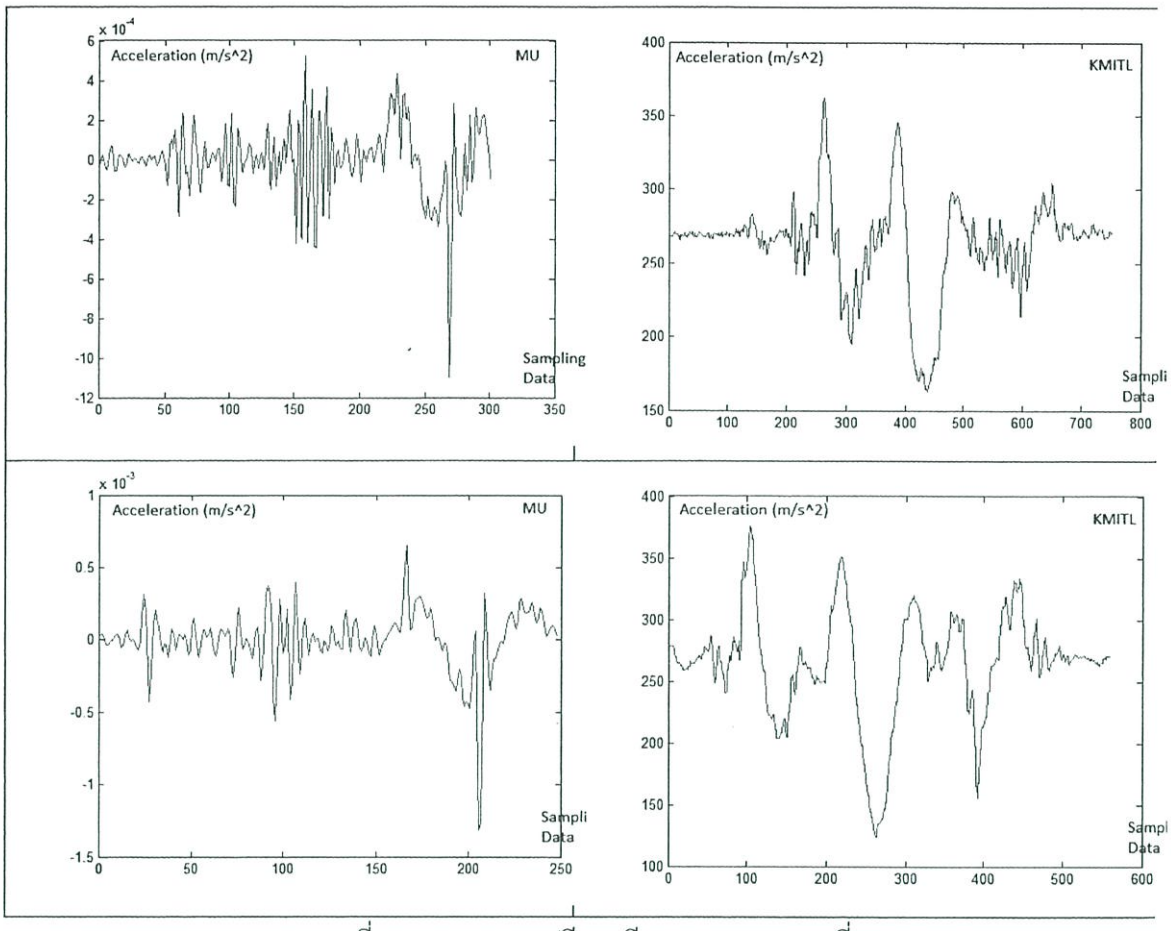




ตารางที่ 5.1 แสดงการเปรียบเทียบผลการทดลองที่ 1

การทดลองที่ 2

Mahidol University	KMITL
	
	
	



ตารางที่ 5.2 แสดงการเปรียบเทียบผลการทดลองที่ 2

จากการทดลองทั้งสองนี้จะเห็นว่ากราฟทั้งสองมีลักษณะการขึ้นลงคล้ายคลึงกัน หมายความว่า อุปกรณ์ที่ผู้วิจัยได้พัฒนาขึ้นมานั้นสามารถนำไปใช้แทนกับอุปกรณ์ที่มีอยู่เดิมได้ โดยมีข้อดีคือ อุปกรณ์ชิ้นนี้มีราคาที่ถูกกว่า นอกจากนั้นยังสามารถพกพาไปใช้ที่ไหนก็ได้ในขณะที่อุปกรณ์อันเดิมไม่สามารถเคลื่อนย้ายได้

เอกสารอ้างอิง

[1]Massey CD, Vincent J, Maneval M, Moore M, Johnson JT, "An analysis of full range of motion vs. partial range of motion training in the development of strength in untrained men.", University of Southern Mississippi, Hattiesburg 39406, USA. DMassey@usm.edu,Aug;18(3):518-21. 2011

[2] Peter Bartz, "Tutorial of using the SparkFun "9DOF Razor IMU"

https://github.com/ptrbrtz/razor-9dof-ahrs/wiki/Tutorial, April 1955.

[3] Programming Knowledge, C# Tutorial :How to use Chart /Graph in VisualC#,https://www.youtube.com/channel/UCs6nmQViDpUw0nuIx9c_WvA.

[4] Spark fun electronics[US], 9 Degrees of Freedom - Razor IMU Features, https://www.sparkfun.com/products/10736.

[5] Spark fun electronics[US], SparkFun FTDI Basic Breakout - 3.3V, https://www.sparkfun.com/products/9873.

[6] Merck manual, "Physical Therapy (PT)", http://www.merckmanuals.com/professional/special subjects/rehabilitation/physical-therapy-pt.

[7] About health, "What Is Range of Motion, or ROM?", http://sportsmedicine.about.com/od/glossary/g/ROM_def.htm.

ภาคผนวก

- Code ตัวอย่างของ IMU

```
#define HW__VERSION_CODE 10736
#define OUTPUT__BAUD_RATE 9600
#define OUTPUT__DATA_INTERVAL 20
#define OUTPUT__MODE_CALIBRATE_SENSORS 0
#define OUTPUT__MODE_ANGLES 1
#define OUTPUT__MODE_SENSORS_CALIB 2
#define OUTPUT__MODE_SENSORS_RAW 3
#define OUTPUT__MODE_SENSORS_BOTH 4
#define OUTPUT__FORMAT_TEXT 0
#define OUTPUT__FORMAT_BINARY 1
int output_mode = OUTPUT__MODE_ANGLES;
int output_format = OUTPUT__FORMAT_TEXT;
#define OUTPUT__STARTUP_STREAM_ON true
boolean output_errors = false;
#define ACCEL_X_MIN ((float) -250)
#define ACCEL_X_MAX ((float) 250)
#define ACCEL_Y_MIN ((float) -250)
#define ACCEL_Y_MAX ((float) 250)
#define ACCEL_Z_MIN ((float) -250)
#define ACCEL_Z_MAX ((float) 250)
#define DEBUG__NO_DRIFT_CORRECTION false
#define DEBUG__PRINT_LOOP_TIME false
#ifndef HW__VERSION_CODE
#endif
#include <Wire.h>
#define ACCEL_X_OFFSET ((ACCEL_X_MIN + ACCEL_X_MAX) / 2.0f)
#define ACCEL_Y_OFFSET ((ACCEL_Y_MIN + ACCEL_Y_MAX) / 2.0f)
#define ACCEL_Z_OFFSET ((ACCEL_Z_MIN + ACCEL_Z_MAX) / 2.0f)
#defineACCEL_X_SCALE(GRAVITY/(ACCEL_X_MAX-
ACCEL_X_OFFSET))
#define                    ACCEL_Y_SCALE(GRAVITY/(ACCEL_Y_MAX-
ACCEL_Y_OFFSET))
#define                    ACCEL_Z_SCALE(GRAVITY/(ACCEL_Z_MAX-
ACCEL_Z_OFFSET))
```

```
35  #define STATUS_LED_PIN 13
36  #define GRAVITY 256.0f
37  #define TO_RAD(x) (x * 0.01745329252)
38  #define TO_DEG(x) (x * 57.2957795131)
39  float accel[3];
40  float accel_min[3];
41  float accel_max[3];
42  float Accel_Vector[3]= {0, 0, 0};
43  float Gyro_Vector[3]= {0, 0, 0};
44  float Omega_Vector[3]= {0, 0, 0};
45  float Omega_P[3]= {0, 0, 0};
46  float Omega_I[3]= {0, 0, 0};
47  float Omega[3]= {0, 0, 0};
48  float DCM_Matrix[3][3] = {{1, 0, 0}, {0, 1, 0}, {0, 0, 1}};
49  float Update_Matrix[3][3] = {{0, 1, 2}, {3, 4, 5}, {6, 7, 8}};
50  float Temporary_Matrix[3][3] = {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}};
51  unsigned long timestamp;
52  unsigned long timestamp_old;
53  float G_Dt;
54  boolean output_stream_on;
55  boolean output_single_on;
56  int curr_calibration_sensor = 0;
57  boolean reset_calibration_session_flag = true;
58  int num_accel_errors = 0;
59  int num_magn_errors = 0;
60  int num_gyro_errors = 0;
61
62  void read_sensors() {
63            Read_Accel();
64          }
65      void turn_output_stream_on()
66          {
67            output_stream_on = true;
68            digitalWrite(STATUS_LED_PIN, HIGH);
69          }
70          void turn_output_stream_off()
71          {
72            output_stream_on = false;
```

```
73              digitalWrite(STATUS_LED_PIN, LOW);
74            }
75            char readChar()
76            {
78              while (Serial.available() < 1) { } // Block
79              return Serial.read();
80            }
81            void setup()
82            {
83              Serial.begin(OUTPUT__BAUD_RATE);
84              pinMode (STATUS_LED_PIN, OUTPUT);
85              digitalWrite(STATUS_LED_PIN, LOW);
86              delay(50);  // Give sensors enough time to start
87              I2C_Init();
88              Accel_Init();
89              delay(20);  // Give sensors enough time to collect data
90            void loop()
91            {
92              if (Serial.available() >= 2)
93              {
94                if (Serial.read() == '#') // Start of new control message
95                {
96                  int command = Serial.read(); // Commands
97                  if (command == 'f') // request one output _f_rame
98                    output_single_on = true;
99                  else if (command == 's') // _s_ynch request
10                {
101                   byte id[2];
102                   id[0] = readChar();
103                   id[1] = readChar();
14                    Serial.print("#SYNCH");
105                   Serial.write(id, 2);
106                   Serial.println();
107               }
18                else if (command == 'o') // Set _o_utput mode
109               {
110                   char output_param = readChar();
111                   if (output_param == 'n')  // Calibrate _n_ext sensor
```

```
112             {
113               curr_calibration_sensor = (curr_calibration_sensor +
114           1) % 3;
115               reset_calibration_session_flag = true;
116             }
117             else if (output_param == 't') // Output angles as
118           _t_ext
119             {
120               output_mode = OUTPUT__MODE_ANGLES;
121               output_format = OUTPUT__FORMAT_TEXT;
122             }
123           else if (output_param == 'b')
124             {
125               output_mode = OUTPUT__MODE_ANGLES;
126               output_format = OUTPUT__FORMAT_BINARY;
127             }
128           else if (output_param == 'c')
129             {
130               output_mode                                          =
131           OUTPUT__MODE_CALIBRATE_SENSORS;
132               reset_calibration_session_flag = true;
133             }
134            else if (output_param == 's')
135               char values_param = readChar();
136               char format_param = readChar();
137               if (values_param == 'r')
138           output_mode = OUTPUT__MODE_SENSORS_RAW;
139            else if (values_param == 'c')
140           output_mode = OUTPUT__MODE_SENSORS_CALIB;
141               else if (values_param == 'b')          o
142           utput_mode = OUTPUT__MODE_SENSORS_BOTH;
143
144               if (format_param == 't')
145                 output_format = OUTPUT__FORMAT_TEXT;
146               else if (format_param == 'b')          output_format
147           = OUTPUT__FORMAT_BINARY;
148             }
149             else if (output_param == '0')
```

```
150                 {
151                   turn_output_stream_off();
152                   reset_calibration_session_flag = true;
153                 }
154                 else if (output_param == '1')
155                 {
156                   reset_calibration_session_flag = true;
157                   turn_output_stream_on();
158                 }
159                 else if (output_param == 'e'
160                 {
161                   char error_param = readChar();
162                   if (error_param == '0') output_errors = false;
163                   else if (error_param == '1') output_errors = true;
164                   else if (error_param == 'c') // get error count
165                   {
166                     Serial.print("#AMG-ERR:");
167                     Serial.print(num_accel_errors); Serial.print(",");
168                     Serial.print(num_magn_errors); Serial.print(",");
169                     Serial.println(num_gyro_errors);
170                   }
171                 }
172               }
173             }
174             else
175             { }
176           }
177           if((millis() - timestamp) >= OUTPUT__DATA_INTERVAL)
178           {
179             timestamp_old = timestamp;
180             timestamp = millis();
181             if (timestamp > timestamp_old)
182               G_Dt = (float) (timestamp - timestamp_old) / 1000.0f;
183         else G_Dt = 0;
184             read_sensors();
185
186   if (output_mode== OUTPUT__MODE_CALIBRATE_SENSORS)
187             {
```

```
188                 check_reset_calibration_session();
189               if(output_stream_on||output_single_on)
190               output_calibration(curr_calibration_sensor);
191                 }
192                 else if (output_mode == OUTPUT__MODE_ANGLES   {
193                   compensate_sensor_errors();
194                   Compass_Heading();
195                   Matrix_update();
196                   Normalize();
197                   Drift_correction();
198                   Euler_angles();
199
200               if (output_stream_on || output_single_on) output_angles();
201                 }
202                 else
203                 {
204               if (output_stream_on || output_single_on) output_sensors();
205                 }
206                 output_single_on = false;
207
208               #if DEBUG__PRINT_LOOP_TIME == true
209                 Serial.print("loop time (ms) = ");
210                 Serial.println(millis() - timestamp);
211               #endif
212                 }
213               #if DEBUG__PRINT_LOOP_TIME == true
214                 else
215                 {
216                   Serial.println("waiting...");
217                 }
218               #endif
219     }
```

- Code IMU สำหรับส่งค่าความเร่ง

```
1   #include <Wire.h>
2   #define  ADXL345_ADDRESS (0xA6 >> 1)
3   #define ADXL345_REGISTER_XLSB (0x32)
4   #define ADXL_REGISTER_PWRCTL (0x2D)
5   #define ADXL_PWRCTL_MEASURE (1 << 3)
6   #define ITG3200_ADDRESS (0xD0 >> 1)
7   #define ITG3200_REGISTER_XMSB (0x1D)
8   #define ITG3200_REGISTER_DLPF_FS (0x16)
9   #define ITG3200_FULLSCALE (0x03 << 3)
0   #define ITG3200_42HZ (0x03)
11  #define HMC5843_ADDRESS (0x3C >> 1)
12  #define HMC5843_REGISTER_XMSB (0x03)
13  #define HMC5843_REGISTER_MEASMODE (0x02)
14  #define HMC5843_MEASMODE_CONT (0x00)
15  int accelerometer_data[3];
16  int gyro_data[3];
17  int magnetometer_data[3];
18  int x;
19  int y;
20  int z;
21  int v1;
22  int s1;
23  unsigned long time;
24  char c;
25  void setup() {
26    Wire.begin();
27    Serial.begin(9600);
28    for(int i = 0; i < 3; ++i) {
29      accelerometer_data[i] = magnetometer_data[i] = gyro_data[i] =
30  0;
31    }
32    init_adxl345();
33    init_hmc5843();
34    init_itg3200();
35  }
36  void loop() {
37     read_adxl345();
38  int x=accelerometer_data[0];
```

```
39    int y=accelerometer_data[1];
40    int z=accelerometer_data[2];
41      Serial.print(x,DEC);
42      Serial.print(",");
43      Serial.print(y,DEC);
44      Serial.print(",");
45      Serial.print(z,DEC);
46      Serial.println("");
47      delay(80);
48    }
49    void i2c_write(int address, byte reg, byte data) {
50     Wire.beginTransmission(address);
51     Wire.write(reg);
52     Wire.write(data);
53     Wire.endTransmission();
54    }
55    void i2c_read(int address, byte reg, int count, byte* data) {
56     int i = 0;
57     Wire.beginTransmission(address);
58     Wire.write(reg);
59     Wire.endTransmission();
60     Wire.beginTransmission(address);
61     Wire.requestFrom(address,count);
62     while(Wire.available()){
63       c = Wire.read();
64       data[i] = c;
65       i++;
66     }
67     Wire.endTransmission();
68    }
69    void init_adxl345() {
70      byte data = 0;
71    i2c_write(ADXL345_ADDRESS, ADXL_REGISTER_PWRCTL,
72    ADXL_PWRCTL_MEASURE);
73    i2c_read(ADXL345_ADDRESS, ADXL_REGISTER_PWRCTL, 1, &data);
74      Serial.println((unsigned int)data);
75    }
76    void read_adxl345() {
```

```
77      byte bytes[6];
78      memset(bytes,0,6);
79      i2c_read(ADXL345_ADDRESS, ADXL345_REGISTER_XLSB, 6, bytes);
80     for (int i=0;i<3;++i) {
81      accelerometer_data[i] = (int)bytes[2*i] + (((int)bytes[2*i + 1]) << 8);
82      }
83     }
84     void init_itg3200() {
85       byte data = 0;
86      i2c_write(ITG3200_ADDRESS, ITG3200_REGISTER_DLPF_FS,
87     ITG3200_FULLSCALE | ITG3200_42HZ);
88      i2c_read(ITG3200_ADDRESS, ITG3200_REGISTER_DLPF_FS, 1, &data);
89     Serial.println((unsigned int)data);
90     }
91     void read_itg3200() {
92       byte bytes[6];
93       memset(bytes,0,6);
94      i2c_read(ITG3200_ADDRESS, ITG3200_REGISTER_XMSB, 6, bytes);
95       for (int i=0;i<3;++i) {
96       gyro_data[i] = (int)bytes[2*i + 1] + (((int)bytes[2*i]) << 8);
97       }
98     }
99     void init_hmc5843() {
100      byte data = 0;
101      i2c_write(HMC5843_ADDRESS, HMC5843_REGISTER_MEASMODE,
102    HMC5843_MEASMODE_CONT);
103    i2c_read(HMC5843_ADDRESS, HMC5843_REGISTER_MEASMODE, 1,
104    &data);     Serial.println((unsigned int)data);
105    }
106    void read_hmc5843() {byte bytes[6];
107     memset(bytes,0,6);
108     i2c_read(HMC5843_ADDRESS, HMC5843_REGISTER_XMSB, 6, bytes);
109    for (int i=0;i<3;++i) {
110     magnetometer_data[i] = (int)bytes[2*i + 1] + (((int)bytes[2*i]) << 8);
111     }
112    }
```

- Code โปรแกรมแอพพลิเคชั่นจากโปรแกรม visual studio
  - Form1.cs

```
1   using System;
2   using System.Collections.Generic;
3   using System.ComponentModel;
4   using System.Data;
5   using System.Drawing;
6   using System.Linq;
7   using System.Text;
8   using System.Threading.Tasks;
9   using System.Windows.Forms;
10  using System.IO.Ports;
11
12  namespace WindowsFormsApplication1
13  {
14      public partial class Form1 : Form
15      {
16          public Form1()
17          {
18              InitializeComponent();
19              Control.CheckForIllegalCrossThreadCalls = false;
20              foreach (string s in SerialPort.GetPortNames())
21              { comboBox1.Items.Add(s); }
22          }
23          private void button1_Click(object sender, EventArgs e)
24          {
25              timer1.Start();
26              serialPort1.Open();
27          }
28          double rt = 0;
29          private void timer1_Tick(object sender, EventArgs e)
30          {
31              rt = rt + 0.1;
32          }
33
34       private void serialPort1_DataReceived(object sender,
35      SerialDataReceivedEventArgs e)
36          {
37              chart1.Series[0].Points.AddXY(rt, serialPort1.ReadLine());
38              label1.Text = serialPort1.ReadLine();
```

```
39              }
40          }
41      }
```

○ Form.Designer

```
1       namespace WindowsFormsApplication1
2       {
3           partial class Form1
4           {
5               /// <summary>
6               /// Required designer variable.
7               /// </summary>
8               private System.ComponentModel.IContainer components = null;
9
10              /// <summary>
11              /// Clean up any resources being used.
12              /// </summary>
13              /// <param name="disposing">true if managed resources should be
14      disposed; otherwise, false.</param>
15              protected override void Dispose(bool disposing)
16              {
17                  if (disposing && (components != null))
18                  {
19                      components.Dispose();
20                  }
21                  base.Dispose(disposing);
22              }
23
24              #region Windows Form Designer generated code
25
26              /// <summary>
27              /// Required method for Designer support - do not modify
28              /// the contents of this method with the code editor.
29              /// </summary>
30              private void InitializeComponent()
31              {
32                  this.components = new System.ComponentModel.Container();
33                  System.Windows.Forms.DataVisualization.Charting.ChartArea
34      chartArea2 = new
35      System.Windows.Forms.DataVisualization.Charting.ChartArea();
36                  System.Windows.Forms.DataVisualization.Charting.Legend
37      legend2 = new
38      System.Windows.Forms.DataVisualization.Charting.Legend();
39                  System.Windows.Forms.DataVisualization.Charting.Series series4
40      = new System.Windows.Forms.DataVisualization.Charting.Series();
```

```
41          System.Windows.Forms.DataVisualization.Charting.Series series5
42      = new System.Windows.Forms.DataVisualization.Charting.Series();
43          System.Windows.Forms.DataVisualization.Charting.Series series6
44      = new System.Windows.Forms.DataVisualization.Charting.Series();
45          this.chart1 = new
46      System.Windows.Forms.DataVisualization.Charting.Chart();
47          this.button1 = new System.Windows.Forms.Button();
48          this.backgroundWorker1 = new
49      System.ComponentModel.BackgroundWorker();
50          this.label1 = new System.Windows.Forms.Label();
51          this.serialPort1 = new
52      System.IO.Ports.SerialPort(this.components);
53          this.timer1 = new
54      System.Windows.Forms.Timer(this.components);
55          this.comboBox1 = new System.Windows.Forms.ComboBox();
56
57      ((System.ComponentModel.ISupportInitialize)(this.chart1)).BeginInit();
58          this.SuspendLayout();
59          //
60          // chart1
61          //
62          chartArea2.Name = "ChartArea1";
63          this.chart1.ChartAreas.Add(chartArea2);
64          legend2.Name = "Legend1";
65          this.chart1.Legends.Add(legend2);
 6          this.chart1.Location = new System.Drawing.Point(48, 27);
67          this.chart1.Name = "chart1";
68          series4.ChartArea = "ChartArea1";
69          series4.ChartType =
70      System.Windows.Forms.DataVisualization.Charting.SeriesChartType.Splin
71      e;
72          series4.Legend = "Legend1";
73          series4.Name = "X";
74          series5.ChartArea = "ChartArea1";
75          series5.ChartType =
76      System.Windows.Forms.DataVisualization.Charting.SeriesChartType.Splin
77      e;
78          series5.Legend = "Legend1";
79          series5.Name = "Y";
80          series6.ChartArea = "ChartArea1";
81          series6.ChartType =
82      System.Windows.Forms.DataVisualization.Charting.SeriesChartType.Splin
83      e;
84          series6.Legend = "Legend1";
85          series6.Name = "Z";
86          this.chart1.Series.Add(series4);
87          this.chart1.Series.Add(series5);
88          this.chart1.Series.Add(series6);
89          this.chart1.Size = new System.Drawing.Size(389, 300);
90          this.chart1.TabIndex = 0;
91          this.chart1.Text = "chart1";
```

```
92              //
93              // button1
94              //
95              this.button1.Location = new System.Drawing.Point(472, 27);
96              this.button1.Name = "button1";
97              this.button1.Size = new System.Drawing.Size(75, 23);
98              this.button1.TabIndex = 1;
99              this.button1.Text = "button1";
100             this.button1.UseVisualStyleBackColor = true;
101             this.button1.Click += new
102     System.EventHandler(this.button1_Click);
103             //
104             // label1
105             //
106             this.label1.AutoSize = true;
107             this.label1.Location = new System.Drawing.Point(469, 70);
108             this.label1.Name = "label1";
109             this.label1.Size = new System.Drawing.Size(35, 13);
110             this.label1.TabIndex = 2;
111             this.label1.Text = "label1";
112             //
113             // serialPort1
114             //
115             this.serialPort1.DataReceived += new
116     System.IO.Ports.SerialDataReceivedEventHandler(this.serialPort1_DataR
117     eceived);
118             //
119             // timer1
120             //
121             this.timer1.Tick += new System.EventHandler(this.timer1_Tick);
123             //
124             // comboBox1
125             //
126             this.comboBox1.FormattingEnabled = true;
127             this.comboBox1.Location = new System.Drawing.Point(472, 102);
128             this.comboBox1.Name = "comboBox1";
129             this.comboBox1.Size = new System.Drawing.Size(121, 21);
130             this.comboBox1.TabIndex = 3;
131             //
132             // Form1
133             //
134             this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
135             this.AutoScaleMode =
136     System.Windows.Forms.AutoScaleMode.Font;
137             this.ClientSize = new System.Drawing.Size(915, 372);
138             this.Controls.Add(this.comboBox1);
139             this.Controls.Add(this.label1);
140             this.Controls.Add(this.button1);
141             this.Controls.Add(this.chart1);
142             this.Name = "Form1";
143             this.Text = "Form1";
```

```
145
146     ((System.ComponentModel.ISupportInitialize)(this.chart1)).EndInit();
147             this.ResumeLayout(false);
148             this.PerformLayout();
149
150         }
151
152         #endregion
153
154         private System.Windows.Forms.DataVisualization.Charting.Chart
155     chart1;
156         private System.Windows.Forms.Button button1;
157         private System.ComponentModel.BackgroundWorker
158     backgroundWorker1;
159         private System.Windows.Forms.Label label1;
160         private System.IO.Ports.SerialPort serialPort1;
161         private System.Windows.Forms.Timer timer1;
162         private System.Windows.Forms.ComboBox comboBox1;
163       }
164     }
```