เครื่องช่วยเตือนจัดบุคลิกท่าทางของหลัง

MONITORING AND WARNING DEVICE FOR BACK SHAPING

โดย

นายนทีกานต์ มูลทองน้อย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2556

เครื่องช่วยเตือนจัดบุคลิกท่าทางของหลัง
MONITORING AND WARNING DEVICE FOR BACK SHAPING

โดย

นายนทีกานต์ มูลทองน้อย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2556

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องช่วยเตือนจัดบุคลิกท่าทางของหลัง
MONITORING AND WARNING DEVICE FOR BACK SHAPING

โดย

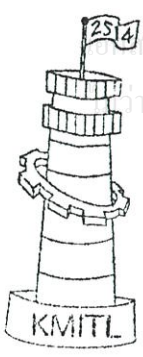
นายนทีกานต์ มูลทองน้อย 53010778

อาจารย์ที่ปรึกษา

รศ.ดร. ปราโมทย์ วาดเขียน

รศ.ดร. จีรสุดา โกษียาภรณ์

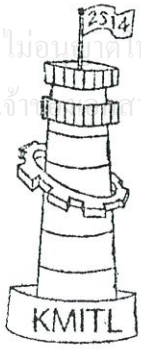
ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2556



ผ่านการตรวจรูปเล่มแล้ว

(.....ปราโมทย์.....)
อาจารย์ที่ปรึกษา
15/5/57

วิศวกรรมโทรคมนาคม
Telecommunications Engineering



ผ่านการตรวจชิ้นงานแล้ว

(.....)
กรรมการผู้ตรวจชิ้นงาน
14/5/57.

วิศวกรรมโทรคมนาคม
Telecommunications Engineering

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2556
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง เครื่องช่วยเตือนจัดบุคลิกท่าทางของหลัง
MONITORING AND WARNING DEVICE FOR BACK SHAPING

ผู้จัดทำ
1. นายนทีกานต์ มูลทองน้อย 53010778

..... อาจารย์ที่ปรึกษา
(รศ.ดร.ปราโมทย์ วาดเขียน)

..... อาจารย์ที่ปรึกษาร่วม
(รศ.ดร.จีรสุดา โกษียาภรณ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้ สามารถสำเร็จลุล่วงได้ด้วยดีเพราะได้รับความช่วยเหลืออย่างดียิ่งจากท่านอาจารย์ที่ปรึกษา รศ.ดร. ปราโมทย์ วาดเขียน และ รศ.ดร. จีรสุดา โกษียาภรณ์ ที่ให้คำแนะนำ คำสั่งสอน ให้ความรู้ความเข้าใจในเรื่องที่ทำปริญญานิพนธ์นี้

ขอขอบคุณเพื่อนๆ ทุกๆคน ที่เป็นกำลังใจและให้ความช่วยเหลืออย่างเต็มที่ตลอดมา

ขอขอบคุณพี่คุณานนต์ กิตติพุฒิ ที่คอยให้คำปรึกษาปัญหาต่างๆ มาโดยตลอด

ขอขอบคุณพี่ภากร ศิริโชติดำรงค์ ที่คอยช่วยเหลือในการทำปริญญานิพนธ์นี้จนแล้วเสร็จ

ขอขอบคุณพี่ปฐมพล ไชยาคำ ที่คอยช่วยเหลือในยามที่ชีวิตเกิดความยากลำบากท้อแท้ และสิ้นหวัง บุญคุณครั้งนี้ไม่อาจที่จะลืมได้จนวันสิ้นลมหายใจ

ขอขอบคุณบทเพลงของนายอาทิวราห์ คงมาลัย ศิลปิน นักร้องวงบอดี้สแลม ที่เติมเต็มกำลังใจให้เต็มเปี่ยมไปด้วยศรัทธาแรงกล้าที่จะทำปริญญานิพนธ์นี้ให้สำเร็จลุล่วงได้ไปด้วยดี

ขอขอบคุณบิดามารดา ที่ให้ความสนับสนุนทางด้านทุนทรัพย์และที่สำคัญที่สุดคอยเป็นแรงผลักดันและกำลังใจทำให้ปริญญานิพนธ์นี้ สำเร็จลุล่วงไปได้

นายนทีกานต์ มูลทองน้อย
ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องช่วยเตือนจัดบุคลิกท่าทางของหลัง
MONITORING AND WARNING DEVICE FOR BACK SHAPING

โดย นทีกานต์ มูลทองน้อย 53010778

อาจารย์ที่ปรึกษา รศ.ดร. ปราโมทย์ วาดเขียน
รศ.ดร. จีรสุดา โกษียาภรณ์

บทคัดย่อ

ปริญญานิพนธ์นี้นำเสนอการออกแบบและสร้างเครื่องช่วยเตือนจัดบุคลิกท่าทางของหลัง เพื่อช่วยลดอาการหลังคด หลังงอ เครื่องจะประกอบไปด้วย วงจรกำเนิดสัญญาณ เซนเซอร์ระยะยืด สายนำสัญญาณ โทรศัพท์มือถือระบบปฏิบัติแอนดรอยด์ เมื่อผู้สวมใส่ขยับลำตัว เซนเซอร์จะตรวจจับท่าทางการห่อไหล่ที่ผิดรูปทรงไปจากปกติ จากนั้นวงจรกำเนิดสัญญาณจะสร้างสัญญาณที่มีความถี่ค่าหนึ่งออกมาเพื่อส่งเข้าโทรศัพท์มือถือระบบปฏิบัติแอนดรอยด์โดยผ่านสายนำสัญญาณ โปรแกรมแอปพลิเคชั่นบนโทรศัพท์มือถือระบบปฏิบัติแอนดรอยด์จะทำการประมวลผลเพื่อตรวจจับท่าการห่อไหล่ที่ผิดปกติไปจากค่าความถี่ของสัญญาณที่ได้รับ โดยจะทำการบันทึกค่าจำนวนครั้งที่ผิดปกติลงในหน่วยความจำไมโครเอสดีการ์ด

ABSTRACT

The project is to design and build a monitoring and warning device. It aims to physical therapy the character and appearance of the back for decreasing the flex and muscle aches. The device consists of oscillator, sensor-stretch, transmission line, android phone. The machine will work when the wearer moves his/her body where sensor detects, the abnormal shape of the back. The oscillator will send the signal to an android mobile phone. Android application will process frequency of the received signal to detect the abnormal posture. The number of times that the abnormal posture occurred will be recorded to the microSD memory card.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
กิตติกรรมประกาศ		I
บทคัดย่อ		II
สารบัญ		III
สารบัญรูป		V
สารบัญตาราง		XI
บทที่ 1	**บทนำ**	1
	1.1 ความเป็นมาและความสำคัญของปัญหา	1
	1.2 วัตถุประสงค์	1
	1.3 ขอบเขตของปริญญานิพนธ์	1
บทที่ 2	**ทฤษฎีและหลักการที่เกี่ยวข้อง**	3
	2.1 ความรู้พื้นฐานเกี่ยวกับภาษาคอมพิวเตอร์ (Computer language)	3
	2.2 ภาษาจาวา (Java programming language)	3
	2.3 ระบบปฏิบัติการแอนดรอยด์ (Android)	4
	2.4 วงรอบการทำงานของแอปพลิเคชั่น (Android Activity Lifecycle)	7
	2.5 การแปลงจากสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (Analog to Digital Conversion ADC, A/D)	8
	2.6 วงจรกำเนิดสัญญาณ (Oscillator)	10
	2.7 วงจรประยุกต์ใช้งานออปแอมป์ที่มีการป้อนกลับแบบลบ	17
	2.8 การแปลงฟูเรียร์ (Fourier Transform)	18
	2.9 ตัวต้านทาน (Resistor)	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

		หน้า
บทที่ 3	**การออกแบบและการจัดทำปริญญานิพนธ์**	26
	3.1 การออกแบบ	26
	3.2 เครื่องมือที่ใช้ในการทดลอง	57
	3.3 การจัดเก็บผลการทดลอง	58
บทที่ 4	**ผลการทดลอง**	59
	4.1 การทดลองวงจรกำเนิดสัญญาณ	59
	4.2 การทดลองการทำงานของโปรแกรมชุดคำสั่งบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์	66
	4.3 ทดสอบอุปกรณ์บนตัวคน และแสดงผลค่าความถี่บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์โดยผ่านทางช่องสัญญาณไมโครโฟนด้วยสายนำสัญญาณเพื่อทำการเตือนผู้สวมใส่ด้วยการสั่น	73
บทที่ 5	**สรุปผลและข้อเสนอแนะ**	101
	5.1 สรุปผล	101
	5.2 ข้อเสนอแนะ	101
บรรณานุกรม		102
ภาคผนวก	SOURCE CODE	103

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	โครงสร้างของระบบปฏิบัติการแอนดรอยด์	5
2.2	วงรอบการทำงานของแอพพลิเคชั่น	8
2.3	กระบวนการของมอดูเลตโดยการเข้ารหัสพัลส์	9
2.4	การป้อนกลับแบบบวกของวงจรกำเนิดสัญญาณ	10
2.5	สัญญาณพัลส์ของวงจรมัลติไวเบรเตอร์	12
2.6	วงจรกำเนิดสัญญาณแบบรีแล็กเซชั่น	13
2.7	สัญญาณของวงจรอะสเตเบิ้ลมัลติไวเบรเตอร์แบบรีแล็กเซชั่น	15
2.8	วงจรสมมูลเพื่อวิเคราะห์หาค่าคาบเวลาของสัญญาณ	15
2.9	วงจรขยายสัญญาณผลรวม	17
2.10	แผนภาพการแตกตัวของวิธีการสลับตำแหน่งและการรวม	20
2.11	แสดง 1 บัตเตอร์ฟลายจะมีการคูณของเลขเชิงซ้อน 2 ครั้ง	20
2.12	การแบ่งทางเวลาที่อธิบายด้วยวิธีการบัตเตอร์ฟลาย	21
2.13	แสดงการหาจำนวนบัตเตอร์ฟลายต่อคมลัมน์	21
2.14	การปรับปรุงบัตเตอร์ฟลาย	22
3.1	บล็อกการทำงานรวมของระบบ	26
3.2	เซนเซอร์ระยะยืด	27
3.3	วงจรกำเนิดสัญญาณแบบอะสเตเบิ้ลมัลติไวเบรเตอร์โดยใช้ไอซีเบอร์ 351 ที่เซนเซอร์ถูกติดอยู่ในแนวตามขวางของแผ่นหลัง	27
3.4	วงจรกำเนิดสัญญาณแบบอะสเตเบิ้ลมัลติไวเบรเตอร์โดยใช้ไอซีเบอร์ 351 ที่เซนเซอร์ถูกติดอยู่ในแนวตั้งตามแนวของกระดูกสันหลัง	28
3.5	วงจรกำเนิดสัญญาณแบบอะสเตเบิ้ลมัลติไวเบรเตอร์โดยใช้ไอซีเบอร์ 351 ที่เซนเซอร์ถูกติดอยู่ในแนวตั้งทางด้านหน้าของลำตัว	29
3.6	วงจรขยายสัญญาณผลรวมโดยใช้ไอซีเบอร์ 741 ที่นำไปใช้ในส่วนของเซนเซอร์ 2 ตัว ที่ติดอยู่ที่ด้านหลังของลำตัว	30
3.7	วงจรขยายสัญญาณผลรวมโดยใช้ไอซีเบอร์ 741 ที่นำไปใช้ในส่วนของเซนเซอร์ด้านหน้าลำตัวตามแนวตั้ง	31
3.8	แผนผังของโปรแกรมชุดคำสั่งในส่วนของการพลอตกราฟบนแอปพลิเคชั่น	32
3.9	แผนผังของโปรแกรมชุดคำสั่งส่วนของการคำนวณหาค่าแอมปลิจูดสูงสุด	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.10	แผนผังของโปรแกรมชุดคำสั่งในส่วนของการคำนวณหาค่า X ใหม่	35
3.11	แผนผังของโปรแกรมชุดคำสั่งในส่วนของการคำนวณหาค่า Y ใหม่	36
3.12	แผนผังของโปรแกรมชุดคำสั่งการพลอตกราฟสัญญาณรูปสี่เหลี่ยม	38
3.13	แผนผังของโปรแกรมชุดคำสั่งเพื่อจำลองสร้างสัญญาณสี่เหลี่ยมบนแอปพลิเคชั่น	39
3.14	แผนผังโปรแกรมชุดคำสั่งของแอปพลิเคชั่นรับสัญญาณเสียงแบบเรียลไทม์	40
3.15	แผนผังโปรแกรมชุดคำสั่งการเก็บค่าแอมปลิจูดของสัญญาณดิจิตอล	43
3.16	แผนผังโปรแกรมชุดคำสั่งการยกระดับสัญญาณดิจิตอล	44
3.17	แผนผังโปรแกรมแอปพลิเคชั่นตรวจจับความถี่สัญญาณแบบเรียลไทม์	46
3.18	แผงผังโปรแกรมชุดคำสั่งของการกำหนดค่าจำนวนครั้งที่นั่งผิดปกติ	54
4.1	เซนเซอร์ระยะยืดขนาดความยาว 4.5 เซนติเมตร	59
4.2	ค่าความต้านทานของเซนเซอร์ระยะยืดขนาดความยาว 4.5 เซนติเมตร	59
4.3	วงจรกำเนิดสัญญาณสี่เหลี่ยมโดยใช้ไอซีเบอร์ 351	60
4.4	ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อไม่มีการยืด	60
4.5	สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ความถี่ 2.5 กิโลเฮิรตซ์ เมื่อไม่มีการยืด	60
4.6	ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 0.5 เซนติเมตร	61
4.7	สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.7 กิโลเฮิรตซ์ เมื่อยืด 0.5 เซนติเมตร	61
4.8	ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 1 เซนติเมตร	61
4.9	สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.5 กิโลเฮิรตซ์ เมื่อยืด 1 เซนติเมตร	62
4.10	ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 1.5 เซนติเมตร	62
4.11	สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.36 กิโลเฮิรตซ์เมื่อยืด 1.5 เซนติเมตร	62
4.12	ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 2 เซนติเมตร	63
4.13	สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.27 กิโลเฮิรตซ์ เมื่อยืด 2 เซนติเมตร	63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.14	ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 2.5 เซนติเมตร	63
4.15	สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.2 กิโลเฮิรตซ์ เมื่อยืด 2.5 เซนติเมตร	64
4.16	ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 3 เซนติเมตร	64
4.17	สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.17 กิโลเฮิรตซ์ เมื่อยืด 3 เซนติเมตร	64
4.18	กราฟเปรียบเทียบค่าความถี่จากการคำนวณทางทฤษฎีกับค่าความถี่ที่มาจากการปฏิบัติ	65
4.19	อุปกรณ์ที่ใช้ในการทดลองเพื่อติดตั้งโปรแกรมแอปพลิเคชั่นระบบปฏิบัติการแอนดรอยด์	66
4.20	การเชื่อมต่อโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์กับคอมพิวเตอร์	67
4.21	การเลือกโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์เพื่อติดตั้งโปรแกรมแอปพลิเคชั่น	67
4.22	แสดงผลการพลอตกราฟสี่เหลี่ยมบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์	68
4.23	หูฟังสมอลทอร์คที่ดัดแปลงเพื่อนำมาใช้ในการป้อนสัญญาณอินพุตผ่านทางช่องสัญญาณไมโครโฟน	69
4.24	อุปกรณ์ที่ใช้ในการเก็บผลการทดลองของแอปพลิเคชั่นรับสัญญาณเสียงแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์	69
4.25	สัญญาณสี่เหลี่ยมอินพุตที่แสดงบนโปรแกรมแอปพลิเคชั่นแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์	69
4.26	สัญญาณเอาต์พุตจากเครื่องกำหนดสัญญาณความถี่ 1 กิโลเฮิรตซ์	70
4.27	การแสดงผลของโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ที่ความถี่ 1 กิโลเฮิรตซ์	71
4.28	สัญญาณเอาต์พุตจากเครื่องกำหนดสัญญาณความถี่ 2 กิโลเฮิรตซ์	71
4.29	การแสดงผลของโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ที่ความถี่ 2 กิโลเฮิรตซ์	72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.30	สัญญาณเอาต์พุตจากเครื่องกำหนดสัญญาณความถี่ 3 กิโลเฮิรตซ์	72
4.31	การแสดงผลของโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ที่ความถี่ 3 กิโลเฮิรตซ์	73
4.32	กระดุมแป๊กตัวเมียที่ถูกดัดแปลงให้ถูกเชื่อมติดกับหางปลากลมเปือยเพื่อยึดเซนเซอร์	73
4.33	แผ่นอิเล็คโทรดที่มีกระดุมแป๊กตัวผู้	74
4.34	แผ่นเจลซิลิโคน	74
4.35	สายนำสัญญาณที่ใช้เชื่อมต่อกับโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ผ่านทางช่องสัญญาณไมโครโฟน	74
4.36	บอร์ดวงจรรวมของระบบ	75
4.37	แหล่งจ่ายไฟที่มีไฟเลี้ยง ±5 โวลต์ และกราวด์	75
4.38	ลักษณะการติดฐานด้วยแผ่นเจลซิลิโคนเพื่อนำมาใช้ยึดเซนเซอร์ระยะยืดไว้บนแผ่นหลัง	75
4.39	ลักษณะการติดเซนเซอร์ 2 ตัว ที่ด้านหลังของร่างกาย	76
4.40	เมื่อผู้สวมใส่นั่งในลักษณะหลังตรงค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 1.85 กิโลโอห์ม	76
4.41	สัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณของเซนเซอร์ที่ถูกติดอยู่ในแนวตามขวางของแผ่นหลังเมื่อผู้สวมใส่นั่งลักษณะหลังตรง	77
4.42	เมื่อผู้สวมใส่นั่งในลักษณะหลังตรงค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 962 โอห์ม	77
4.43	สัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณของเซนเซอร์ที่ถูกติดอยู่ในแนวตั้งตามแนวกระดูกสันหลังเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง	78
4.44	สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง	78
4.45	แอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง	79
4.46	หน้าจอแอปพลิเคชั่นเมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะหลังตรง	79

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.47	เมื่อผู้สวมใส่นั่งในลักษณะหลังค่อมค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 2.11 กิโลโอห์ม	80
4.48	เมื่อผู้สวมใส่นั่งในลักษณะหลังค่อมค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 1.164 กิโลโอห์ม	80
4.49	สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมเมื่อผู้สวมใส่นั่งในลักษณะหลังค่อม	81
4.50	แอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณเมื่อผู้สวมใส่นั่งในลักษณะหลังค่อม	82
4.51	หน้าจอแอปพลิเคชั่นเมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะหลังค่อม	82
4.52	สัญญาณเสียงจากการสั่นเตือนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์	83
4.53	ลักษณะการติดเซนเซอร์ที่ด้านหน้าของร่างกาย	83
4.54	เมื่อผู้สวมใส่นั่งในลักษณะหลังตรงค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 2.14 กิโลโอห์ม	84
4.55	สัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณของเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตั้งทางด้านหน้าของลำตัวเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง	84
4.56	สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง	85
4.57	แอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณเมื่อผู้สวมใส่นั่งในลักษณหลังตรง	85
4.58	หน้าจอแอปพลิเคชั่นเมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะหลังตรง	86
4.59	เมื่อผู้สวมใส่นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้าค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 2.35 กิโลโอห์ม	86
4.60	สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมเมื่อผู้สวมใส่นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้า	87
4.61	แอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณเมื่อผู้สวมใส่นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้า	88
4.62	หน้าจอแอปพลิเคชั่นเมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้า	88

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.63	สัญญาณเสียงจากการสั่นเตือนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์	89
4.64	ไฟล์ที่ถูกเขียนลงในไมโครเอสดีการ์ด	89
4.65	ข้อมูลที่ถูกเขียนลงไปในไฟล์ที่ชื่อ 9-5-2014.txt	90
4.66	จำนวนครั้งที่นั่งในลักษณะหลังค่อมเท่ากับ 613 ครั้ง	91
4.67	ไฟล์ที่ถูกเขียนลงในไมโครเอสดีการ์ดของวันที่ 13 พฤษภาคม 2014	91

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่		หน้า
4.1	เปรียบเทียบค่าความถี่ในทางทฤษฎีและในทางปฏิบัติที่ระยะยืดต่างๆ	65
4.2	แสดงเวลาของไฟล์ชื่อ 13-5-2014.TXT ในไมโครเอสดีการ์ด	92
4.3	จำนวนครั้งที่สามารถตรวจจับได้และไม่ได้	95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันอายุขัยโดยเฉลี่ยของประชากรภายในประเทศมีแนวโน้มเพิ่มขึ้น เนื่องจากประชากรสามารถเข้าถึงระบบสาธารณูปโภคได้อย่างทั่วถึง การส่งเสริมคุณภาพชีวิตของบุคคลในสังคม ซึ่งการนำเอาเทคโนโลยีมาประยุกต์ใช้ในชีวิตประจำวันนั้น จะสามารถตอบสนองความต้องการของบุคคลในสังคมในการปฏิบัติภารกิจในชีวิตประจำวันได้อย่างมีประสิทธิภาพ อีกทั้งยังช่วยยกระดับคุณภาพชีวิตของบุคคลในสังคมให้มีความเป็นอยู่ที่ดีขึ้น

อาการปวดหลังพบได้เป็นอันดับสองของอาการปวดในร่างกาย รองมาจากอาการปวดหัว และเป็นอาการที่พบบ่อยที่สุดของผู้ป่วยที่มาพบแพทย์ทางศัลยกรรมกระดูก อาการปวดหลังโดยทั่วไปไม่มีอันตรายร้ายแรง แต่เป็นปัญหากับผู้คนในวัยทำงาน และการดำรงชีวิตประจำวัน ในต่างประเทศเคยมีการวิจัยถึงค่าใช้จ่ายที่ต้องสูญเสียไปเนื่องจากอาการปวดหลัง ปวดเอว การเข้ารับการตรวจรักษา รวมถึงการหยุดงานเพื่อพักฟื้น พบว่าต้องสูญเสียเงินและเวลาเป็นจำนวนมาก ดังนั้นการตรวจรักษาที่ถูกต้อง รวมไปถึงการป้องกัน และปฏิบัติตัวที่ถูกวิธี จะเป็นวิธีจัดการกับอาการนี้ได้ถูกต้อง

การปฏิบัติภารกิจหน้าที่ในชีวิตประจำวันของบุคคลทั่วไปในสังคมนั้น มีการใช้กิริยาท่าทางมากมายในชีวิตประจำวัน เช่น การทำงานยกของหนัก การนั่ง หรือการยืนในท่าที่ไม่ถูกต้อง ก่อให้เกิดอาการปวดหลัง ปวดเอว ซึ่งนำไปสู่การเสื่อมสภาพของกระดูกสันหลัง แสดงให้เห็นว่าระดับคุณภาพชีวิตของบุคคลทั่วไปในสังคมไม่มีความมั่นคงในด้านสาธารณสุข อีกทั้งยังอยู่ในเกณฑ์ที่เสี่ยงต่อการเกิดโรคภัยไข้เจ็บในอนาคต

1.2 วัตถุประสงค์

1) เพื่อออกแบบและสร้างอุปกรณ์ต้นแบบที่มีเซนเซอร์ที่สามารถตรวจจับท่าทางที่ผิดปกติได้

2) เพื่อพัฒนาโปรแกรมแอปพลิเคชั่นบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ด้วยภาษาจาวา

3) เพื่อศึกษาการทำงานของเซนเซอร์ที่สามารถนำไปพัฒนาใช้ในเชิงการแพทย์ของโรคที่เกี่ยวข้องกับวิชาแพทย์สาขาเวชศาสตร์ฟื้นฟู

1.3 ขอบเขตของการทำปริญญานิพนธ์

1) สามารถสร้างเซนเซอร์ระยะยืดที่มีการยืดออกได้ 3 เซนติเมตร

2) สามารถออกแบบและสร้างแอปพลิเคชั่นที่ใช้แจ้งเตือนเมื่อผู้สวมใส่มีบุคลิกท่าทางที่ผิดปกติบนโทรศัพท์มือถือบนระบบปฏิบัติการแอนดรอยด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) สามารถเขียนโปรแกรมแอปพลิเคชั่นตรวจจับสัญญาณที่มีความถี่ค่าต่างๆ ได้ ไม่เกิน 4 กิโลเฮิรตซ์ ผ่านทางช่องสัญญาณไมโครโฟนด้วยสายนำสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 ความรู้พื้นฐานเกี่ยวกับภาษาคอมพิวเตอร์ (Computer language)

ภาษาคอมพิวเตอร์แบ่งออกเป็น

2.1.1 ภาษาเครื่องเป็นภาษาระดับต่ำสุด

เป็นภาษาที่อยู่ในรูปเลขฐานสิบหกโดยที่ภาษาเครื่องจะประกอบด้วยเลข 0 กับ 1 ดังนั้นภาษาเครื่องจึงเป็นภาษาที่สั่งคอมพิวเตอร์ทำงานได้ทันที แต่มีข้อเสียคือทำการพัฒนาภาษาได้ยากเนื่องจากมีลักษณะที่เป็นเลขฐานสอง

2.1.2 ภาษาแอสเซมบลี (Assembly language)

เป็นกึ่งภาษาเครื่อง การทำงานของภาษาจะทำงานทีละบรรทัด โดยเรียงจากบรรทัดบนสุดไปจนถึงบรรทัดสุดท้ายที่เขียนโปรแกรม

2.1.3 ภาษาขั้นสูง

ภาษาขั้นสูงเป็นภาษาที่ใกล้เคียงกับภาษาของมนุษย์เช่น ภาษาซี เป็นต้น ซึ่งภาษาชนิดนี้จะมีการแปลงเป็นภาษาเครื่องก่อนโดยผ่านโปรแกรมคอมไพเลอร์

2.2 ภาษาจาวา (Java programming language)

ภาษาจาวา คือภาษาโปรแกรมเชิงวัตถุ พัฒนาโดย เจมส์ กอสลิง และวิศวกรคนอื่นๆ ที่บริษัท ซัน ไมโครซิสเต็มส์ ภาษานี้มีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัส (C++) โดยรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาอ็อบเจกต์ทีฟซี (Objective-C) แต่เดิมภาษานี้เรียกว่า ภาษาโอ๊ก (Oak) ซึ่งตั้งชื่อตามต้นโอ๊กใกล้ที่ทำงานของ เจมส์ กอสลิง แล้วภายหลังจึงเปลี่ยนไปใช้ชื่อ "จาวา" ซึ่งเป็นชื่อกาแฟแทน จุดเด่นของภาษาจาวา อยู่ที่ผู้เขียนโปรแกรมสามารถใช้หลักการของ Object-Oriented Programming มาพัฒนาโปรแกรมของตนด้วยภาษาจาวาได้

ภาษาจาวาเป็นภาษาสำหรับเขียนโปรแกรมที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (OOP: Object-Oriented Programming) โปรแกรมที่เขียนขึ้นถูกสร้างภายในคลาส ดังนั้นคลาสคือที่เก็บเมทอด (Method) หรือพฤติกรรม (Behavior) ซึ่งมีสถานะ (State) และรูปพรรณ (Identity) ประจำพฤติกรรม (Behavior)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษาจาวาเป็นภาษาที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุแบบสมบูรณ์ ซึ่งเหมาะสำหรับพัฒนาระบบที่มีความซับซ้อน การพัฒนาโปรแกรมแบบวัตถุจะช่วยให้เราสามารถใช้คำหรือชื่อ ต่าง ๆ ที่มีอยู่ในระบบงานนั้นมาใช้ในการออกแบบโปรแกรมได้ ทำให้เข้าใจได้ง่ายขึ้น

โปรแกรมที่เขียนขึ้นโดยใช้ภาษาจาวาจะมีความสามารถทำงานได้ในระบบปฏิบัติการที่แตกต่างกัน ไม่จำเป็นต้องดัดแปลงแก้ไขโปรแกรม

ภาษาจาวามีการตรวจสอบข้อผิดพลาดทั้งตอนที่คอมไพล์ และขณะที่โปรแกรมกำลังทำงาน ทำให้สามารถลดข้อผิดพลาดที่อาจเกิดขึ้นในโปรแกรม และช่วยให้ดีบั๊ก (debug) โปรแกรมได้ง่าย

ภาษาจาวามีความซับซ้อนน้อยกว่าภาษาซีพลัสพลัส เมื่อเปรียบเทียบชุดคำสั่งของโปรแกรมที่เขียนขึ้นโดยภาษาจาวา กับภาษาซีพลัสพลัส พบว่า โปรแกรมที่เขียนโดยภาษาจาวา จะมีจำนวนชุดคำสั่งน้อยกว่าโปรแกรมที่เขียนโดยภาษาซีพลัสพลัส ทำให้ใช้งานได้ง่ายกว่าและลดความผิดพลาดได้มากขึ้น

ภาษาจาวาถูกออกแบบมาให้มีความปลอดภัยสูง ทำให้โปรแกรมที่เขียนขึ้นด้วยภาษาจาวามีความปลอดภัยมากกว่าโปรแกรมที่เขียนขึ้นด้วยภาษาอื่น เพราะจาวามีการป้องกันทั้ง low level และ high level ได้แก่ electronic signature, public and private key management

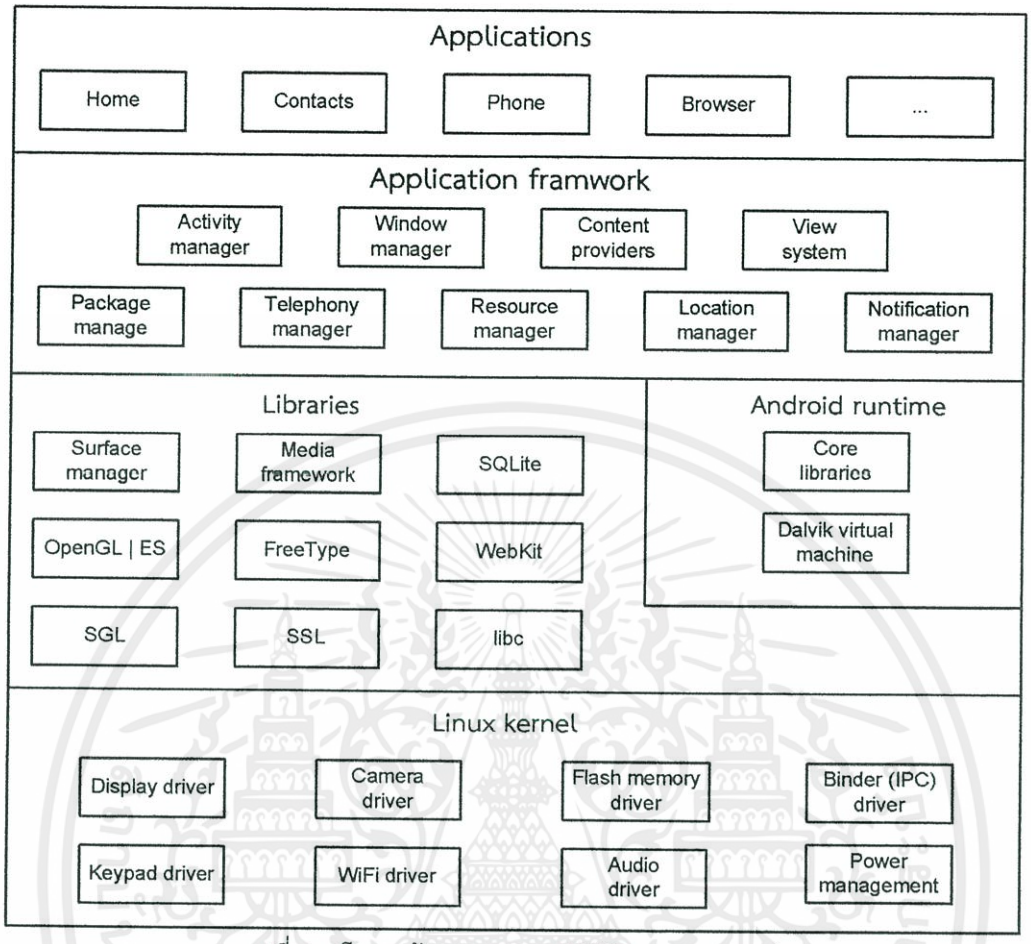
2.3 ระบบปฏิบัติการแอนดรอยด์ (Android)

ระบบปฏิบัติการแอนดรอยด์ เป็นระบบปฏิบัติการที่พัฒนามาจากการนำเอาแกนกลางของระบบปฏิบัติการลินุกซ์ (Linux Kernel) ซึ่งเป็นระบบปฏิบัติการที่ออกแบบมาเพื่อทำงานเป็นเครื่องให้บริการ (Server) มาพัฒนาต่อเพื่อให้กลายเป็นระบบปฏิบัติการบนอุปกรณ์พกพา (Mobile Operating System)

2.3.1 โครงสร้างของระบบปฏิบัติการแอนดรอยด์ (Android Architecture)

โครงสร้างของระบบปฏิบัติการแอนดรอยด์มีโครงสร้างดังรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 โครงสร้างของระบบปฏิบัติการแอนดรอยด์

โครงสร้างของระบบปฏิบัติการแอนดรอยด์แบ่งออกมาเป็นส่วนๆ โดยส่วนบนสุดจะเป็นส่วนที่ผู้ใช้งานทำการติดต่อโดยตรงซึ่งก็คือส่วนของ (Applications) จากนั้นก็จะลำดับลงมาเป็นองค์ประกอบอื่นๆตามลำดับ และสุดท้ายจะเป็นส่วนที่ติดต่อกับอุปกรณ์โดยผ่านทาง Linux Kernel สามารถอธิบายโครงสร้างของแอนดรอยด์ได้ดังนี้

2.3.1.1 แอปพลิเคชั่น (Applications)

ส่วนแอปพลิเคชั่น หรือส่วนของโปรแกรมที่มีมากับระบบปฏิบัติการ หรือเป็นกลุ่มของโปรแกรมที่ผู้ใช้งานได้ทำการติดตั้งไว้ โดยผู้ใช้งานสามารถเรียกใช้โปรแกรมต่างๆได้โดยตรง ซึ่งการทำงานของแต่ละโปรแกรมจะเป็นไปตามที่ผู้พัฒนาโปรแกรมได้ออกแบบและเขียนคำสั่งโปรแกรมเอาไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1.2 Application Framework

เป็นส่วนที่มีการพัฒนาขึ้นเพื่อให้นักพัฒนาสามารถพัฒนาโปรแกรมได้สะดวก และมีประสิทธิภาพมากยิ่งขึ้น โดยนักพัฒนาไม่จำเป็นต้องพัฒนาในส่วนที่มีความยุ่งยากมากๆ เพียงแค่ทำการศึกษาถึงวิธีการเรียกใช้งาน Application Framework ในส่วนที่ต้องการใช้งาน แล้วนำส่วนนั้นมาใช้งาน ในปริญญานิพนธ์นี้มีการใช้งานดังนี้

1) Activities Manager เป็นกลุ่มของชุดคำสั่งที่จัดการเกี่ยวกับวงจรการทำงานของหน้าต่างโปรแกรม (Activity)

2) Content Providers เป็นกลุ่มของชุดคำสั่ง ที่ใช้ในการเข้าถึงข้อมูลของโปรแกรมอื่นและสามารถแบ่งปันข้อมูลให้โปรแกรมอื่นเข้าถึงได้

3) View System เป็นกลุ่มของชุดคำสั่งที่เกี่ยวกับการจัดการโครงสร้างของหน้าจอที่แสดงผลในส่วนที่ติดต่อกับผู้ใช้งาน (User Interface)

4) Resource Manager เป็นกลุ่มของชุดคำสั่งในการเข้าถึงข้อมูลที่เป็น ข้อความ, รูปภาพ

2.3.1.3 Libraries

เป็นส่วนของชุดคำสั่งที่พัฒนาด้วยภาษาซี และภาษาซีพลัสพลัส โดยแบ่งชุดคำสั่งออกเป็นกลุ่มตามวัตถุประสงค์ของการใช้งาน ในปริญญานิพนธ์นี้ใช้ Library ดังนี้

1) os เป็นส่วนการทำงานของ Thread ใช้สำหรับแอคทิวิตี้ที่ทำงานอย่างต่อเนื่อง

2) view เป็นส่วนของการตั้งค่าหลักในการแสดงผล

3) app เป็นส่วนของการใช้งานในแอปพลิเคชั่น

4) util เป็นส่วนบรรจุการพัฒนาการกำหนดค่าตัวแปรต่างๆ และการแสดงค่าต่างๆ บน Console ทำให้สะดวกต่อการเอาไปใช้งานมากขึ้น

5) content เป็นส่วนที่ใช้สำหรับการเข้าถึงข้อมูล และการแสดงผลข้อมูล

6) media เป็นส่วนที่ใช้งานในส่วนของออดิโอ และวิดีโอ

7) widget เป็นส่วนสำหรับการควบคุมหรือสั่งการทำงานต่างๆ บนแอปพลิเคชั่น เช่น ปุ่มกด อักขระ

8) io เป็นส่วนที่ใช้เมื่อมีการรับอินพุต และส่งเอาต์พุต เช่น การบันทึกไฟล์ลงบนไมโครเอสดีการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1.4 Android Runtime

จะมี Darvik Virtual Machine ที่ถูกออกแบบมา เพื่อให้ทำงานบนอุปกรณ์ที่มีหน่วยความจำ(Memmory), หน่วยประมวลผลกลาง(CPU) และพลังงาน(Battery) ที่จำกัด ซึ่งการทำงานของ Darvik Virtual Machine จะทำการแปลงไฟล์ที่ต้องการทำงาน ไปเป็นไฟล์ .DEX ก่อนการทำงาน เพื่อให้มีประสิทธิภาพเพิ่มขึ้นเมื่อใช้งานกับหน่วยประมวลผลกลางที่มีความเร็วไม่มาก ส่วนต่อมาคือ Core Libraries ที่เป็นส่วนรวบรวมคำสั่งและชุดคำสั่งสำคัญซึ่งถูกพัฒนาด้วยภาษาจาวา

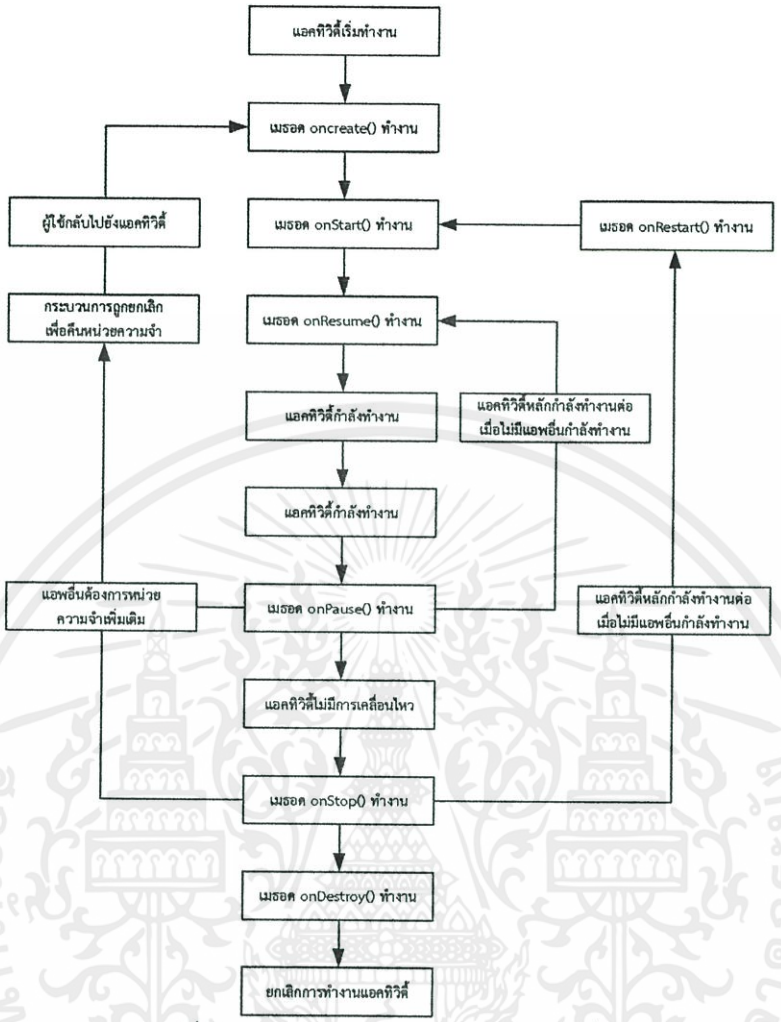
2.3.1.5 Linux Kernel

เป็นส่วนที่ทำหน้าที่หัวใจสำคัญในจัดการกับบริการหลักของระบบปฏิบัติการ เช่น เรื่องหน่วยความจำ พลังงาน ติดต่อกับอุปกรณ์ต่างๆ ความปลอดภัย เครือข่าย โดยระบบปฏิบัติการณ์แอนดรอยด์ได้นำเอาส่วนนี้มาจากระบบปฏิบัติการลินุกซ์ รุ่น 2.6 (Linux 2.6 Kernel)

2.4 วงรอบการทำงานของแอปพลิเคชั่น (Android Activity Lifecycle)

ระบบปฏิบัติการณ์แอนดรอยด์ถูกออกแบบมาเพื่อใช้งานกับอุปกรณ์บนโทรศัพท์มือถือ ซึ่งจะมีทรัพยากรของระบบที่จำกัด จึงจำเป็นที่ต้องมีกลไกสำหรับจัดการบริหารทรัพยากร เรียกว่า วงรอบการทำงานของแอพพลิเคชั่น ซึ่งจะกำหนดสถานะที่แอคทิวิตีต้องผ่านตั้งแต่ตอนที่ถูกสร้างขึ้นจนกว่าจะเสร็จสิ้นการทำงาน การทำงานของวงรอบการทำงานจะมีหลักการทำงานหลักๆดังแสดงในรูปที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 วงรอบการทำงานของแอพพลิเคชั่น

2.5 การแปลงจากสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (Analog to Digital Conversion, ADC, A/D)

2.5.1 การมอดูเลตแบบพัลส์ (Pulse Modulation)

การแปลงจากสัญญาณอุปมานเป็นสัญญาณเชิงเลขหรือสัญญาณดิจิตอลขบวนการแปลงนี้เรียกว่า การแปลงจากสัญญาณอนาลอกเป็นสัญญาณดิจิตอล ซึ่งบางทีเรียกวาการมอดูเลต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วยพัลส์ (Digital Pulse Modulation) ซึ่งมีสองเทคนิคหลักที่ใช้กัน คือ การมอดูเลตแบบเข้ารหัสพัลส์ หรือพีซีเอ็ม (Pulse Code Modulation, PCM) และ การมอดูเลตแบบเดลต้า (Delta Modulation) การมอดเลตแบบเข้ารหัสพัลส์จะประกอบด้วยสามขั้นตอนคือ ขั้นตอนการสุ่มสัญญาณ หรือการแซมปลิ้ง (Sampling) ขั้นตอนการจัดระดับสัญญาณ หรือการควอนไทซ์ และขั้นตอนการเข้ารหัส (Coding) ดังแสดงในรูปที่ 2.3

รูปที่ 2.3 กระบวนการของมอดูเลตโดยการเข้ารหัสพัลส์

2.5.2 การสุ่มตัวอย่างสัญญาณอุปมาน (Sampling of Analog Signal)

การสุ่มตัวอย่างก็คือกระบวนการที่สัญญาณที่มีความต่อเนื่องถูกวัดค่าขนาดที่เวลาขณะ หนึ่งๆ ที่ไม่มีความต่อเนื่อง ขนาดของสัญญาณที่สุ่มได้จะถูกแปลงเป็นขนาดที่สอดคล้องกับสัญญาณเดิม แต่ไม่มีความต่อเนื่อง ด้วยกระบวนการจัดขนาดและเข้ารหัส หากการสุ่มข้อมูลนี้เป็นไปตามทฤษฎีการสุ่มแล้วก็สามารถที่จะทำกลับคืนจากสัญญาณดิจิตอลมาเป็นสัญญาณต่อเนื่องได้อย่างถูกต้อง

2.5.3 การควอนไทซ์ (Quantizing)

สัญญาณที่ได้จากการสุ่ม ยังไม่สามารถที่จะเข้ารหัสเป็นเลขไบนารีได้ หากไม่ทราบจำนวนบิตที่ต้องการซึ่งการระบุจำนวนบิตที่ต้องการจะเป็นการกำหนดการเก็บรายละเอียดของสัญญาณ เนื่องจากจำนวนบิตที่ใช้มีจำกัด การจัดระดับสัญญาณก็จำกัดไปด้วย ปกติแล้วก่อนสัญญาณจะถูกสุ่ม ก็จะมีการกำหนดระดับสูงสุดเรียกว่าระดับเต็มสเกล (Full scale voltage, VFS) การแทนแรงดันต่อสัญญาณที่สุ่มได้ ด้วยเลขฐานสองจานวน m บิตนั้น ทำให้สามารถแยกแรงดันออกได้เป็น 2^m ระดับแรงดันที่สุ่มมาแล้วใกล้เคียงกับระดับใดมากที่สุด ก็จะถูกจัดให้เป็นระดับนั้น กระบวนการนี้จะเรียกว่า การจัดระดับ หรือ Quantization

2.5.4 การมอดูเลตแบบเข้ารหัสพัลส์ (Pulse Code Modulation, PCM)

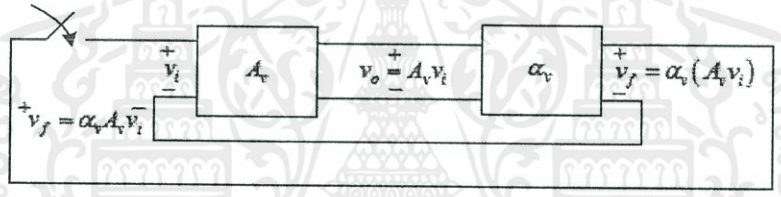
สัญญาณที่ถูกจัดระดับหรือควอนไทซ์มาแล้ว ในแต่ละระดับของการควอนไทซ์ ก็จะกำหนดใหม่รหัสประจำระดับนั้นๆ แต่ละระดับก็จะแทนด้วยเลขไบนารีจำนวน m บิต ซึ่งสามารถที่จะใช้ได้ตั้งแต่ ค่าศูนย์ (0000000.....0) จนถึง $2^m - 1$ (111111.....1) จะเห็นว่าข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวน m นั้นควรจะถูกส่งไปหมดก่อนที่จะมีการสุ่มตัวอย่างครั้งต่อไป หากความถี่ในการสุ่มคือ f_s จะได้ว่าความเร็วในการส่งหรือ บิตเรท ก็ควรจะเป็น nf_s บิตต่อวินาที

2.6 วงจรกำเนิดสัญญาณ (Oscillator)

วงจรกำเนิดสัญญาณเป็นวงจรหนึ่งที่มีความสำคัญทางวิศวกรรมอิเล็กทรอนิกส์และวิศวกรรมการสื่อสาร สามารถอธิบายได้จากทฤษฎีการแกว่ง (Oscillation) การแกว่งเป็นการเคลื่อนที่กลับไปกลับมาบนเส้นทางเดิม เรียกว่าการเคลื่อนที่ที่มีคาบ (Period) โดยหลักการพื้นฐานสำคัญประการหนึ่งของวงจรกำเนิดสัญญาณก็คือ จะต้องมีการป้อนกลับแบบบวก ทั้งนี้ก็เพื่อเสริมสัญญาณที่กำลังจะหายไปอันเนื่องมาจากกำลังงานของสัญญาณลดลง อย่างไรก็ตามการป้อนกลับของสัญญาณนั้นจะต้องกระทำอย่างมีเงื่อนไข เพราะการป้อนกลับอย่างไม่เหมาะสมอาจเป็นผลให้สัญญาณคลื่นรูปไซน์ที่กำเนิดนั้นมีองค์ประกอบความถี่มากกว่า 1 ความถี่ นอกเหนือจากความถี่ที่ต้องการ เพื่อความเข้าใจในหลักการพื้นฐานการกำเนิดสัญญาณโดยพื้นฐานของวงจรกำเนิดสัญญาณสามารถอธิบายจากบล็อกไดอะแกรมของระบบได้ดังรูปที่ 2.4



รูปที่ 2.4 การป้อนกลับแบบบวกของวงจรกำเนิดสัญญาณ

จากรูปที่ 2.4 นั้นหากสวิตช์ยังเปิดวงจรอยู่จะไม่มีสัญญาณเอาต์พุตของวงจรเกิดขึ้น แต่เมื่อสวิตช์ปิดวงจรจะทำให้ระบบครบลูปการทำงาน ซึ่งเป็นเหมือนการป้อนสัญญาณอินพุตให้กับวงจรในเวลาอันสั้น เมื่อมีสัญญาณอินพุตจะส่งผลให้เกิดสัญญาณเอาต์พุต $v_o = A_v v_i$ ขึ้นและจะได้ $v_f = \alpha_v A_v v_i$ โดยค่าผลคูณ $\alpha_v A_v$ นี้เรียกว่าลูปเกน (Loop Gain) ซึ่งหากค่าลูปเกนของวงจรทำให้สัญญาณ v_f ที่เกิดขึ้นมีขนาดและเฟสเท่ากับสัญญาณ v_i เดิม ซึ่งเมื่อสวิตช์ได้ปิดลง สัญญาณ v_i เดิมจะหายไป สัญญาณ v_f ก็จะกลับมาเป็นสัญญาณ v_i ใหม่เหมือนเดิม จึงทำให้ยังคงมีสัญญาณเอาต์พุต v_o ของวงจรเกิดขึ้นอย่างไม่มีที่สิ้นสุด ดังนั้นสามารถสรุปเงื่อนไขที่สำคัญของวงจรกำเนิดสัญญาณได้ดังนี้

1) วงจรต้องมีการป้อนกลับแบบบวก คือ สัญญาณที่เกิดจากในส่วนของการขยายและในส่วนของการป้อนกลับจะต้องมีเฟสรวมกันเท่ากับ 360°

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) วงจรต้องได้รับสัญญาณทริก (trig) เริ่มต้นในช่วงเวลาสั้นๆ จากภายในวงจรหรือจากภายนอกวงจร เพื่อเริ่มการทำงานของวงจร

3) วงจรจะต้องมีคุณสมบัติตามเงื่อนไขบาร์คฮูเซ่น (Barkhausen Criterion) กล่าวคือ ผลคูณของค่าอัตราขยายของวงจร A_v และอัตราการป้อนกลับของวงจร α_v จะต้องมีค่าเท่ากับ 1

ซึ่งถ้าหาก $\alpha_v A_v < 1$ จะทำให้สัญญาณที่ได้มีขนาดเล็กลงไปเรื่อยๆ จนกระทั่งสัญญาณนั้นหายไป แต่ถ้า $\alpha_v A_v > 1$ จะทำให้สัญญาณมีขนาดค่อยๆ ใหญ่ขึ้นจนถึงระดับสูงสุดและต่ำสุดตามระดับแรงดันไฟเลี้ยง วงจรกำเนิดสัญญาณสามารถแบ่งออกได้เป็นสองกลุ่มใหญ่ๆ คือ วงจรกำเนิดสัญญาณรูปไซน์ (Sinusoidal waveform generator) และวงจรกำเนิดสัญญาณรูปอื่นๆ ที่ไม่ใช่สัญญาณไซน์ (Non-sinusoidal waveform generator)

2.6.1 วงจรกำเนิดสัญญาณรูปไซน์

การสร้างวงจรกำเนิดสัญญาณรูปไซน์อย่างง่ายสามารถสร้างได้โดยใช้อุปกรณ์ต่างๆ เช่น ออปแอมป์ (Op-amp) โดยที่สามารถต่อในรูปแบบของเวนบริดจ์ออสซิลเลเตอร์ (Wien-bridge oscillator) ซึ่งสามารถปรับความถี่และกำลังของสัญญาณได้โดยตัวต้านทานและตัวเก็บประจุ หรือต่อเข้ากับขดลวดเหนี่ยวนำและตัวเก็บประจุ ซึ่งการปรับจะต้องมีความสอดคล้องกับเงื่อนไขเฉพาะที่จะทำให้เกิดการแกว่ง ขนาดของสัญญาณไซน์ที่ถูกสร้างขึ้นจะถูกจำกัดโดยย่านการทำงานเชิงเส้นของวงจรขยาย วงจรกำเนิดสัญญาณรูปไซน์ลักษณะนี้จึงถูกเรียกว่าเป็นวงจรกำเนิดสัญญาณแบบเชิงเส้น (Linear oscillator)

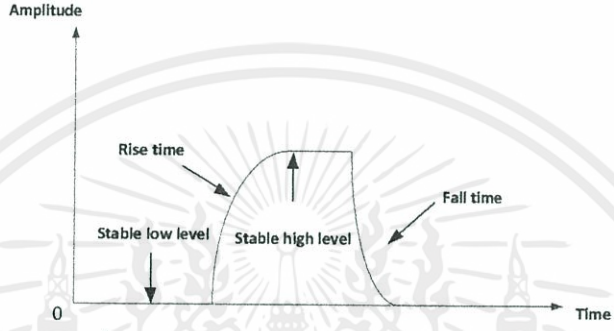
2.6.2 วงจรกำเนิดสัญญาณที่ไม่ใช่รูปไซน์

วงจรกำเนิดสัญญาณแบบอื่นๆ ที่ไม่ใช่สัญญาณรูปไซน์ ซึ่งก็คือวงจรกำเนิดสัญญาณรูปสี่เหลี่ยม (Squarewave waveform generator) วงจรกำเนิดสัญญาณรูปฟันเลื่อย (Sawtooth waveform generator) และวงจรกำเนิดสัญญาณรูปสามเหลี่ยม (Triangular waveform generator) วงจรกำเนิดสัญญาณที่ไม่ใช่รูปไซน์มีชื่อเรียกอีกอย่างหนึ่งว่า วงจรกำเนิดสัญญาณแบบผ่อนคลาย (Relaxation oscillator) ทั้งนี้เพราะอุปกรณ์ขยายที่ใช้ในวงจร จะทำงานอยู่ในย่านคัตออฟ (Cut-off) หรืออิ่มตัว (Saturation) เท่านั้น ในระหว่างที่อุปกรณ์ขยายทำงานอยู่ในย่านคัตออฟ จะไม่มีกระแสไหล จึงเสมือนกับว่าอุปกรณ์นั้นอยู่ในลักษณะผ่อนคลาย (Relaxing) โดยที่วงจรกำเนิดสัญญาณที่ไม่ใช่รูปไซน์ลักษณะนี้จึงถูกเรียกว่าเป็นวงจรกำเนิดสัญญาณแบบไม่เชิงเส้น (Non-linear oscillator)

2.6.2.1 วงจรมัลติไวเบรเตอร์ (Multivibrator)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรมัลติไวเบรเตอร์เป็นวงจรสวิตช์แบบหนึ่งซึ่งสามารถสร้างพัลส์ โดยทำหน้าที่เป็นตัวตั้งเวลาและสามารถกำเนิดความถี่ต่างๆออกมาได้ด้วยตัวเองหรือใช้สัญญาณภายนอกมาควบคุมการทำงานของวงจร ซึ่งวงจรจะสร้างจากอุปกรณ์อิเล็กทรอนิกส์คือทรานซิสเตอร์ ตัวความต้านทาน และตัวเก็บประจุ ลักษณะของรูปคลื่นสี่เหลี่ยมที่สร้างจากวงจรมัลติไวเบรเตอร์มีคุณสมบัติที่สำคัญ 4 ส่วน คือความเสถียรของด้านล่างรูปคลื่น (Stable low level) ความเสถียรด้านบนของรูปคลื่น (Stable high level) ช่วงเวลาไต่ขึ้นจากด้านล่างถึงด้านบนของรูปคลื่น (Rise time) และช่วงเวลาตกลงจากด้านบนถึงด้านล่างของรูปคลื่น (Fall time) ดังรูปที่ 2.5



รูปที่ 2.5 สัญญาณพัลส์ของวงจรมัลติไวเบรเตอร์

วงจรจะควบคุมเวลาทั้ง 4 ส่วนให้สามารถทำงานได้ด้วยความเร็วสูงและมีเสถียรภาพ วงจรมัลติไวเบรเตอร์แบ่งตามลักษณะของวงจรได้ 4 ชนิด คือ

1) วงจรโมโนสเตเบิ้ลมัลติไวเบรเตอร์ (Monostable Multivibrator)

วงจรโมโนสเตเบิ้ลมัลติไวเบรเตอร์หรือวันช็อตมัลติไวเบรเตอร์ (One-shot Multivibrator) เป็นวงจรที่ทำงาน 2 สภาวะ คือสภาวะที่มีเสถียรภาพและสภาวะกึ่งเสถียรภาพ วงจรที่อยู่ในสภาวะปกติจะอยู่ในสภาวะเสถียรภาพจนกว่าจะมีสัญญาณจากภายนอกเข้ามาการกระตุ้น วงจรจึงกลับไปอยู่ในสภาวะกึ่งเสถียรภาพ และเวลาที่อยู่ในสภาวะกึ่งเสถียรภาพจะนานแค่ไหนจะขึ้นอยู่กับค่าคงที่ของเวลาและจะกลับคืนสภาวะเสถียรภาพ จนกว่าจะมีสัญญาณจากภายนอกมากระตุ้นใหม่ ดังนั้นสภาวะกึ่งเสถียรภาพจึงเป็นสภาวะชั่วขณะ

2) วงจรไบสเตเบิ้ลมัลติไวเบรเตอร์ (Bistable Multivibrator)

วงจรไบสเตเบิ้ลมัลติไวเบรเตอร์หรือวงจรฟลิบฟลอบ (Flip-flop circuit) เป็นวงจรที่ทำงาน 2 สภาวะ คือส่วนหนึ่งจะอยู่สภาวะทำงาน ส่วนอีกสภาวะจะไม่ทำงาน

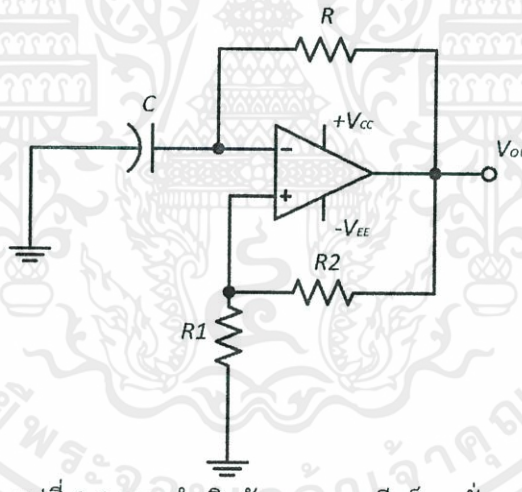
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการเปลี่ยนสภาวะการทำงานของวงจร จะเปลี่ยนสภาวะก็ต่อเมื่อมีสัญญาณพัลส์จากภายนอกมาควบคุมการทำงาน และสภาวะการทำงานของวงจรจะเปลี่ยนเป็นตรงกันข้ามเมื่อมีสัญญาณพัลส์จากภายนอกมากระตุ้นในครั้งที่สอง สภาวะการทำงานของวงจรจึงจะกลับไปสู่สภาวะเดิม สรุปได้ว่าทรานซิสเตอร์ทั้งสองจะเปลี่ยนสภาวะเมื่อมีสัญญาณพัลส์บวกจากภายนอกเข้ามากระตุ้นนั่นเอง

3) วงจรอะสเตเบิ้ลมัลติไวเบรเตอร์ (Astable Multivibrator)

วงจรอะสเตเบิ้ลมัลติไวเบรเตอร์หรือวงจรฟรี-รันนิ่ง (Free-running multivibrator) คุณสมบัติของวงจรคือไม่ต้องมีสัญญาณอินพุตมากระตุ้น วงจรมีการทำงานกึ่งเสถียรภาพ 2 สภาวะ ระยะเวลาของแต่ละสภาวะกึ่งเสถียรภาพทั้งสองจะขึ้นอยู่กับตัวต้านทานและตัวเก็บประจุในวงจร ถึงแม้ว่าจะไม่มีสัญญาณพัลส์จากอินพุตป้อนเข้ามากระตุ้นการทำงาน วงจรอะสเตเบิ้ลมัลติไวเบรเตอร์ก็จะสามารถกำเนิดเอาต์พุตสัญญาณรูปสี่เหลี่ยมออกมาได้

วงจรอะสเตเบิ้ลมัลติไวเบรเตอร์สามารถสร้างได้โดยใช้วงจรเปรียบเทียบแรงดัน (Comparator) ดังรูปที่ 2.6 ซึ่งวงจรจัดเป็นวงจรกำเนิดสัญญาณแบบรีแล็กเซชั่น (Relaxation Oscillator) คือ การทำงานของวงจรจะอาศัยการเกิดวนซ้ำของการเก็บประจุ (Charging) และการคายประจุ (Discharging) ของตัวเก็บประจุ



รูปที่ 2.6 วงจรกำเนิดสัญญาณแบบรีแล็กเซชั่น

ในการวิเคราะห์วงจรในรูปที่ 2.6 นั้น จะสมมติให้ก่อนสภาวะเริ่มต้น $t<t_0$ แรงดันตกคร่อมเริ่มต้นของตัวเก็บประจุซึ่งก็คือแรงดันที่ขาลบของออปแอมป์ v^- มีค่าเท่ากับ 0 โวลต์ และโดยอาศัยหลักการพื้นฐานของออปแอมป์คือ $v_o = Av_d = A\left(v^+ - v^-\right); A > 10^6$ ทำให้ผลต่างของสัญญาณขาบวกและขาลบของออปแอมป์จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถูกขยายด้วยค่าอัตราขยายที่สูงมาก โดยจะมีค่าเป็นบวกหรือลบขึ้นอยู่กับเครื่องหมาย v_d แต่เนื่องจากแอมปลิจูดของสัญญาณเอาต์พุต v_o ของออปแอมป์จะถูกจำกัดด้วยไฟเลี้ยงของวงจรซึ่งมีค่าเท่ากับ $\pm V_{CC}$ ดังนั้น v_o จึงมีค่าเท่ากับ $+V_{CC} = +V_{\max}$ หรือ $-V_{CC} = -V_{\max}$ สมมติให้ว่าสัญญาณเอาต์พุต v_o เปรียบเทียบแรงดันมีค่าเท่ากับ $+V_{\max}$ ดังนั้นแรงดันที่ขาบวกของออปแอมป์จะเป็นดังสมการที่ 2.1

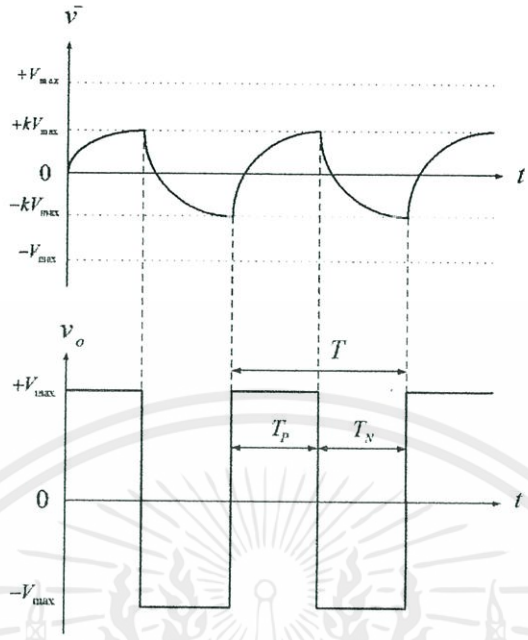
$$v^+ = \left(\frac{R_1}{R_1 + R_2}\right) V_{\max} = kV_{\max} \tag{2.1}$$

ซึ่งตัวเก็บประจุจะทำการเก็บประจุ ทำให้แรงดัน v^- ค่อยๆ เพิ่มขึ้นจาก 0 โวลต์ จนกระทั่งถึง $kV_{\max}$ จะทำให้สัญญาณเอาต์พุต v_o ของวงจรเปรียบเทียบแรงดันเริ่มเปลี่ยนระดับเป็น $-V_{\max}$ เพราะแรงดันที่ขาลบของออปแอมป์ v^- เริ่มมีค่ามากกว่าแรงดันที่ขาบวกของออปแอมป์ v^+ จึงทำให้ค่าผลต่าง $v^+ - v^-$ มีค่าเป็นลบ เมื่อสัญญาณเอาต์พุต v_o มีค่าเป็น $-V_{\max}$ จึงทำให้แรงดันที่ขาบวกของออปแอมป์จะเป็นดังสมการที่ 2.2

$$v^+ = \left(\frac{R_1}{R_1 + R_2}\right) V_{\max} = -kV_{\max} \tag{2.2}$$

ดังนั้นตัวเก็บประจุจะเริ่มทำการคายประจุ จากนั้นแรงดัน v^- จะค่อยๆ ลดลงจาก $kV_{\max}$ จนถึง $-kV_{\max}$ ทำให้สัญญาณเอาต์พุต v_o ของวงจรเปรียบเทียบแรงดันเริ่มเปลี่ยนระดับกลับไปเป็น $+V_{\max}$ เพราะแรงดันที่ขาลบของออปแอมป์ v^- เริ่มมีค่ามากกว่าแรงดันที่ขาบวกของออปแอมป์ v^+ หรือ ค่าผลต่าง $v^+ - v^-$ มีค่าเป็นบวก โดยอาศัยหลักการรีแล็กเซชั่นของตัวเก็บประจุ จึงสามารถแสดงรูปสัญญาณที่ขาลบ v^- และสัญญาณเอาต์พุต v_o เป็นสัญญาณรูปสี่เหลี่ยมได้ดังรูปที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

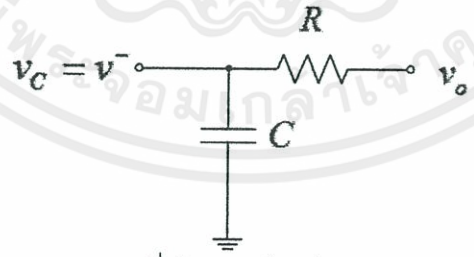


รูปที่ 2.7 สัญญาณของวงจรอะสเตเบิ้ลมัลติไวเบรเตอร์แบบรีแล็กเซชั่น

จากวงจรในรูปที่ 2.6 และรูปสัญญาณเอาต์พุต v_o ในรูปที่ 2.7 สามารถแสดงการวิเคราะห์หาค่าคาบเวลาของสัญญาณรูปสี่เหลี่ยมได้ดังสมการที่ 2.3

$$T = T_P + T_N \tag{2.3}$$

เมื่อ T_P และ T_N คือ เวลาของช่วงเวลาที่พัลส์เป็นบวกและช่วงเวลาที่พัลส์เป็นลบ ซึ่งมีค่าเท่ากับเวลาที่ใช้ในการเก็บประจุและคายประจุของตัวเก็บประจุตามลำดับ โดยวงจรสมมูลที่ใช้ในการวิเคราะห์หาค่าคาบของสัญญาณสามารถแสดงได้ดังรูปที่ 2.8



รูปที่ 2.8 วงจรสมมูลเพื่อวิเคราะห์หาค่าคาบเวลาของสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อพิจารณาในช่วงเวลา T_P ซึ่ง v_C มีค่าเปลี่ยนแปลงจาก $-kV_{\max}$ ถึง $+kV_{\max}$ และ v_o มีค่าเท่ากับ $+V_{\max}$ จะสามารถหาสมการของ T_P ได้ดังสมการที่ 2.4

$$\frac{v_C - v_o}{R} + C\frac{dv_C(t)}{dt} = 0$$

$$-C\frac{dv_C(t)}{dt} = \frac{v_C - v_o}{R}$$

$$\frac{dv_C(t)}{v_C - v_o} = -\frac{dt}{RC}$$

$$\int_{-kV_{\max}}^{+kV_{\max}} \frac{dv_C(t)}{v_C - v_o} = -\frac{1}{RC}\int_0^{T_P} dt$$

$$\ln\left|v_C - v_o\right|\Big|_{-kV_{\max}}^{+kV_{\max}} = -\frac{T_P}{RC}; \qquad v_o = +V_{\max}$$

$$\ln\left|-(1-k)V_{\max}\right| - \ln\left|-(1+k)V_{\max}\right| = -\frac{T_P}{RC}$$

$$\ln\left|-(1+k)V_{\max}\right| - \ln\left|-(1-k)V_{\max}\right| = \frac{T_P}{RC}$$

$$\ln\left(\frac{\left|-(1+k)V_{\max}\right|}{\left|-(1-k)V_{\max}\right|}\right) = \frac{T_P}{RC}$$

$$\ln\left(\frac{(1+k)}{(1-k)}\right) = \frac{T_P}{RC}$$

$$T_P = RC\ln\left(\frac{(1+k)}{(1-k)}\right); \quad k = \left(\frac{R_1}{R_1 + R_2}\right) \qquad (2.4)$$

เมื่อพิจารณาในช่วงเวลา T_N ซึ่ง v_C มีค่าเปลี่ยนแปลงจาก $+kV_{\max}$ ถึง $-kV_{\max}$ และ v_o มีค่าเท่ากับ $-V_{\max}$ จะหาช่วงเวลา T_N ได้ดังสมการที่ 2.5

$$T_N = RC\ln\left(\frac{(1+k)}{(1-k)}\right) \qquad (2.5)$$

ดังนั้นค่าคาบเวลาของสัญญาณรูปสี่เหลี่ยม T จะเป็นดังสมการที่ 2.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T = 2RC\ln\left(\frac{(1+k)}{(1-k)}\right); \quad k = \frac{R_1}{R_1+R_2}$$
$$= 2RC\ln\left(1+\frac{2R_1}{R_2}\right) \tag{2.6}$$

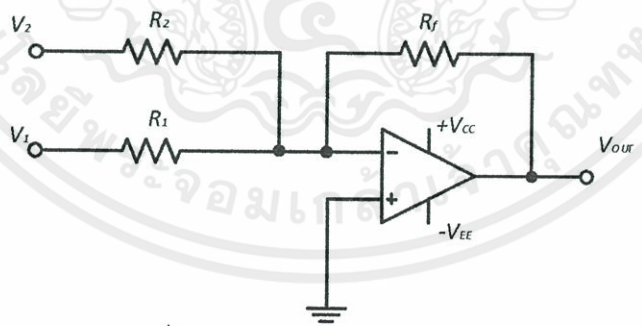
และหาค่าดิวตี้ไซเคิล (Duty cycle) ได้ดังสมการที่ 2.7

$$D = \frac{T_P}{T}$$
$$= \frac{RC\ln\left(\frac{(1+k)}{(1-k)}\right)}{2RC\ln\left(\frac{(1+k)}{(1-k)}\right)} = 50\% \tag{2.7}$$

2.7 วงจรประยุกต์ใช้งานออปแอมป์ที่มีการป้อนกลับแบบลบ

2.7.1 วงจรขยายสัญญาณผลรวม (Summing Amplifier)

สำหรับวงจรขยายสัญญาณผลรวมนั้น มีลักษณะการต่อวงจรดังรูปที่ 2.9 จะเห็นว่าสัญญาณอินพุตจะถูกกำหนดไว้ทั้งหมด 2 สัญญาณ เมื่อทำการวิเคราะห์หาความสัมพันธ์ของกระแสที่โหนดขาลบของออปแอมป์ จะได้สมการสัญญาณเอาต์พุตเป็นดังสมการที่ 2.8



รูปที่ 2.9 วงจรขยายสัญญาณผลรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$V_{OUT} = -R_f\left(\frac{V_1}{R_1}+\frac{V_2}{R_2}\right) \tag{2.8}$$

ถ้าให้ $R_1 = R_2 = R_f$ จะได้ว่า

$$V_{OUT} = -(V_1+V_2) \tag{2.9}$$

2.8 การแปลงฟูเรียร์ (Fourier Transform)

การแปลงฟูเรียร์เป็นการแปลงฟังก์ชันในโดเมนเวลาให้ไปอยู่ในโดเมนความถี่ โดยมีนิยามดังสมการที่ 2.10

$$X(\omega) = \int_{-\infty}^{\infty} x(t)\cdot e^{-j\omega t}\,dt \tag{2.10}$$

โดยที่ ω มีค่าอยู่ระหว่าง $(-\infty,\infty)$

2.8.1 การแปลงฟูเรียร์ไม่ต่อเนื่อง (Discrete Fourier Transform)

การแปลงฟูเรียร์ไม่ต่อเนื่องเป็นการแปลงฟูเรียร์ของข้อมูลดิจิตอล ซึ่งเป็นฟังก์ชันไม่ต่อเนื่องทางเวลา $x[n]$ โดยที่ n คือ ลำดับของข้อมูลที่ได้จากการสุ่มชักข้อมูล โดยข้อมูลที่ได้จากการสุ่มชักจะมีจำนวน N ค่า ซึ่งการแปลงฟูเรียร์ไม่ต่อเนื่องจะสามารถเขียนนิยามได้ดังสมการที่ 2.11

$$\begin{aligned} X(k) &= X(\omega_k)|_{\omega_k = 2\pi k/N} \\ &= \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N} \end{aligned} \tag{2.11}$$

โดยที่ k คือ ค่าความถี่ต่างๆ ซึ่งมีค่าอยู่ในช่วง 0 ถึง $N-1$ และ $X[k]$ คือ ค่าขนาดบนโดเมนความถี่ที่อยู่ในช่วงของ k จากสมการที่ 2.11 จะเห็นว่าในการคำนวณเพื่อหาค่า $X[k]$ นั้น มีจำนวนการทำงาน(Operation) ค่อนข้างมากถ้าจำนวนข้อมูลมีมาก

2.8.2 การแปลงฟูเรียร์แบบเร็ว (Fast Fourier Transform)

จากนิยามของการแปลงฟูเรียร์ไม่ต่อเนื่องจะแทนค่าเลขเชิงซ้อนเป็น $W_N \triangleq e^{-j\frac{2\pi}{N}}$ จะได้สมการใหม่ดังสมการที่ 2.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \tag{2.12}$$

โดยที่ k คือ ค่าความถี่ต่างๆ ซึ่งมีค่าอยู่ในช่วง 0 ถึง $N-1$ และถ้าคำนวณ W_N ไว้ก่อนที่ $n \cdot k$ เท่ากับ 0 จะได้ค่าเลขเชิงซ้อนเท่ากับ 1 ดังนี้

$$W_N^0 \triangleq e^0 = 1$$

และเมื่อค่า $n \cdot k$ เท่ากับ $N/2$ จะได้ค่าเลขเชิงซ้อนเท่ากับ -1 ดังนี้

$$W_N^{N/2} = e^{-j\left(\frac{2\pi}{N}\right)\left(\frac{N}{2}\right)} = e^{-j\pi} = -1$$

ซึ่งจะทำให้ลดจำนวนการคูณของเลขเชิงซ้อนลงได้อย่างมาก โดยวิธีการแปลงฟูเรียร์แบบเร็วนั้นจะเป็นการสลับตำแหน่งและการรวม(Recomposite) ดังสมการที่ 2.13

$$X(k) = x(0)W_N^0 + x(1)W_N^{1 \cdot k} + \ldots + x(n)W_N^{nk} \tag{2.13}$$

จากสมการที่ 2.13 เมื่อทำการสลับตำแหน่งใหม่ โดยแบ่ง n เป็น 2 ชุด ซึ่งได้แก่ชุดที่เป็นเลขคี่และชุดที่เป็นเลขคู่ ซึ่งถ้า n เป็นเลขคู่จะได้สมการที่ 2.14

$$X_{even}(k) = x(0)W_N^0 + x(2)W_N^{2 \cdot k} + \ldots + x(n)W_N^{nk} \tag{2.14}$$

ถ้า n เป็นเลขคี่ จะได้สมการที่ 2.15 ดังนี้

$$X_{odd}(k) = x(1)W_N^k + x(3)W_N^{3 \cdot k} + \ldots + x(n)W_N^{nk} \tag{2.15}$$

และเนื่องจาก

$$W_N^2 = e^{-j\left(\frac{2\pi}{N}\right)(2)} = e^{-j\left(\frac{2\pi}{\frac{N}{2}}\right)} = W_{N/2}$$

จึงทำให้เขียนสมการที่ 2.14 ได้ใหม่เป็นสมการที่ 2.16 ดังนี้

$$X_{even}(k) = x(0)W_N^0 + x(2)W_{N/2}^k + \ldots + x(n)W_{N/2}^{nk/2} \tag{2.16}$$

และเขียนสมการที่ 2.15 ได้ใหม่เป็นสมการที่ 2.17 ดังนี้

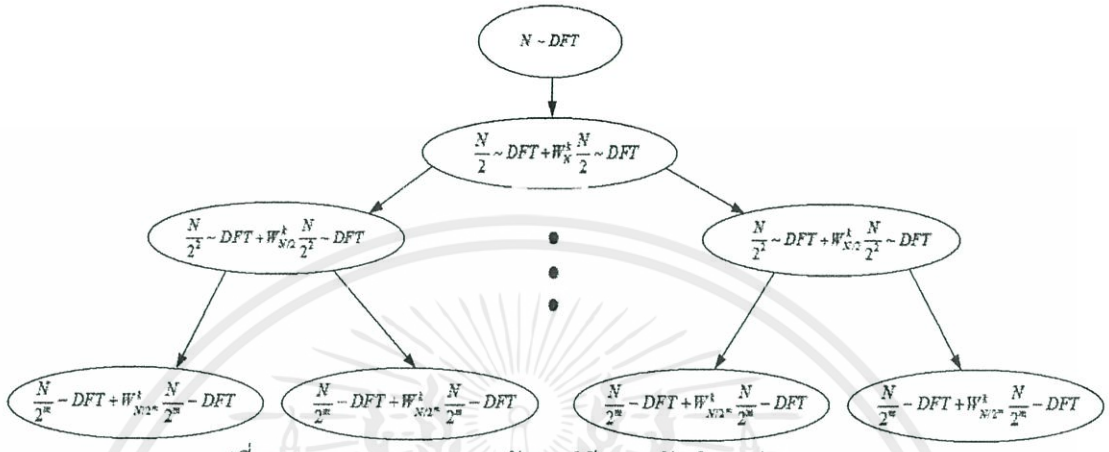
$$X_{odd}(k) = W_N^{1 \cdot k}[x(1)W_N^0 + x(3)W_{N/2}^k + \ldots + x(n)W_{N/2}^{(n-1)k/2}] \tag{2.17}$$

ดังนั้นจะเขียนสมการการหา $X(k)$ ใหม่ได้ดังสมการที่ 2.18 ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$X(k) = X_{even}(k) + W_N^k X_{odd}(k) \tag{2.18}$$

การอธิบายการแตกตัวของการแปลงฟูเรียร์ไม่ต่อเนื่องแบบ 2 จุด 2 ชุด ของการแปลงฟูเรียร์แบบเร็ว ด้วยวิธีการสลับตำแหน่งและการรวม จะแสดงแผนภาพได้ดังรูปที่ 2.10



รูปที่ 2.10 แผนภาพการแตกตัวของวิธีการสลับตำแหน่งและการรวม

เมื่อ m คือจำนวนขั้น(Stage) และมีค่าเท่ากับ $\log_2 N - 1$ ในขั้นสุดท้ายนั้นจะแตกตัวออกเป็นการแปลงฟูเรียร์ไม่ต่อเนื่องแบบ 2 จุด เนื่องจากขั้นสุดท้ายนั้นจะมีค่าเท่ากับ $\frac{N}{2^m}$ ซึ่งมีค่าเท่ากับ 2 ทำให้สามารถลดจำนวนตัวคูณและตัวบวกของเลขเชิงซ้อนลงได้อย่างมาก นอกจากนั้นยังมีการใช้ประโยชน์จากคุณสมบัติของความเป็นคาบของสัญญาณดังนี้

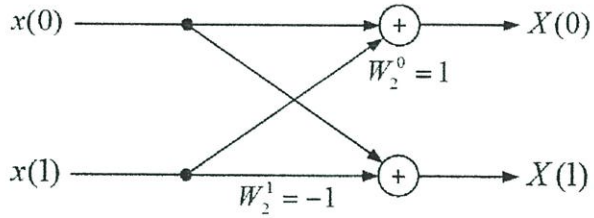
$$X_{even}(k) = X_{even}(k + N/2)$$

และ

$$X_{odd}(k) = X_{odd}(k + N/2)$$

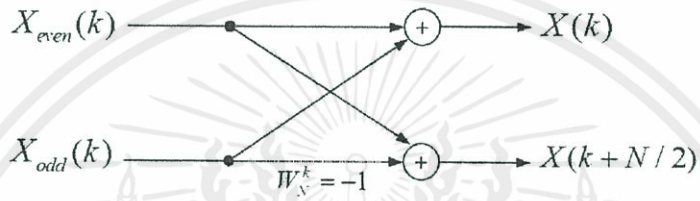
ทำให้สามารถลดช่วงของค่า k จาก $N-1$ ลงเหลือเพียง $(N/2)-1$ อย่างไรก็ตามได้มีการนำวิธีการอธิบายการคำนวณในรูปแบบต่างๆ มาใช้ ทำให้สามารถคำนวณหาจำนวนตัวคูณเลขเชิงซ้อนได้ง่ายขึ้น วิธีการอธิบายที่ถูกนำมาใช้คือ การใช้กราฟการไหลของสัญญาณ(Signal flow graph) หรือที่เรียกว่า บัตเตอร์ฟลาย(Butterfly) มาใช้อธิบาย โดย 1 บัตเตอร์ฟลายจะมีการคูณของเลขเชิงซ้อน 2 ครั้ง รูปที่ 2.11 แสดงแผนภาพบัตเตอร์ฟลายเมื่อ N เท่ากับ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



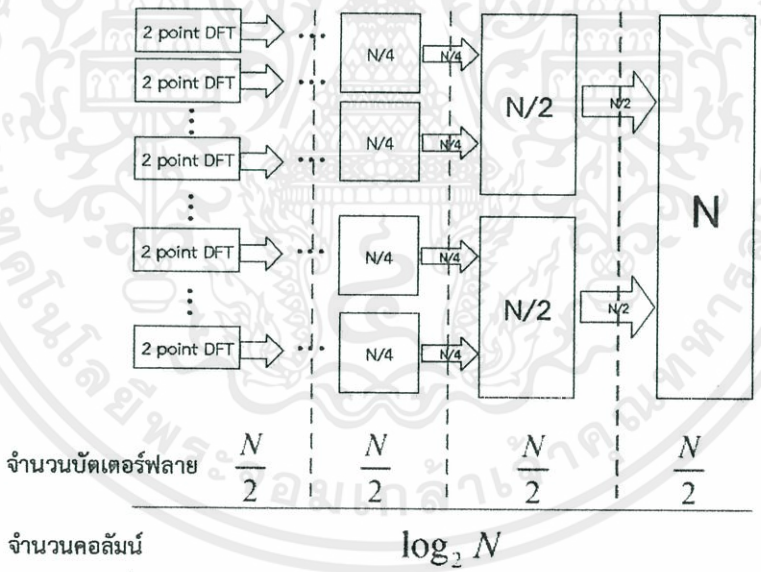
รูปที่ 2.11 แสดง 1 บัตเตอร์ฟลายจะมีการคูณของเลขเชิงซ้อน 2 ครั้ง

เมื่อนำบัตเตอร์ฟลายมาใช้ จะสามารถแสดงการแบ่งทางเวลาของค่า $X_{even}(k)$ และ $X_{odd}(k)$ ได้ดังรูปที่ 2.12



รูปที่ 2.12 การแบ่งทางเวลาที่อธิบายด้วยวิธีการบัตเตอร์ฟลาย

จากนั้นทำการแบ่งชุดของบัตเตอร์ฟลายออกเป็นคอลัมม์ได้ดังรูปที่ 2.13



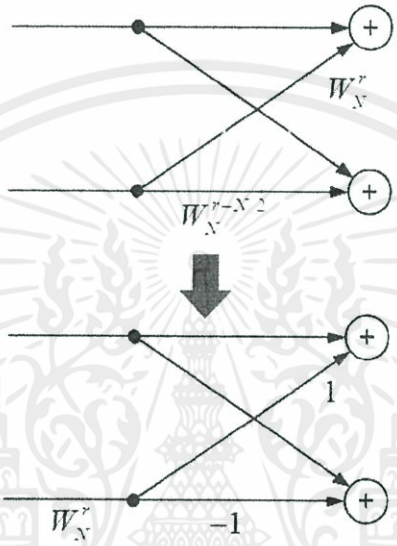
รูปที่ 2.13 แสดงการหาจำนวนบัตเตอร์ฟลายต่อคอลัมน์

จากรูปที่ 2.13 สามารถคำนวณหาจำนวนการคูณเลขเชิงซ้อนซึ่งจะเท่ากับจำนวนบัตเตอร์ฟลายต่อคอลัมน์คูณกับจำนวนคอลัมน์ และคูณกับ 2 เนื่องจาก 1 บัตเตอร์ฟลายจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีการคูณ 2 ครั้ง ทำให้ได้จำนวนครั้งการคูณเลขเชิงซ้อนของการแปลงฟูเรียร์แบบเร็วเท่ากับ $N\log_2 N$ ซึ่งลดลงจากจำนวนครั้งของการแปลงฟูเรียร์ไม่ต่อเนื่องที่มีค่าเท่ากับ N^2 อย่างมาก

ในปริญญานิพนธ์เล่มนี้ได้ใช้อัลกอริทึมการแปลงฟูเรียร์แบบเร็วของคูลลี่ ทูกี้ (Cooley Tukey Algorithm) ซึ่งวิธีการของคูลลี่ทูกี้นั้น จะเป็นการลดจำนวนของบัตเตอร์ฟลายทั้งหมดลงครึ่งหนึ่งด้วยวิธีการที่เรียกว่า การปรับปรุงบัตเตอร์ฟลาย วิธีการปรับปรุงบัตเตอร์ฟลายเกิดจากเลขเชิงซ้อนที่มี $n \cdot k$ เท่ากับ $N/2$ ทำให้เลขเชิงซ้อน $W_N^{N/2}$ มีค่าเท่ากับ –1 เพราะฉะนั้นจึงเกิดเงื่อนไขดังรูปที่ 2.14 เมื่อ r คือเลขใดๆ



รูปที่ 2.14 การปรับปรุงบัตเตอร์ฟลาย

จากรูปที่ 2.14 วิธีการปรับปรุงบัตเตอร์ฟลายนี้ทำให้เหลือจำนวนตัวคูณเลขเชิงซ้อนลดลงครึ่งหนึ่ง นั่นก็คือเหลือเพียงแค่ $\frac{N}{2}\log_2 N$ เท่านั้น

2.9 ตัวต้านทาน (Resistor)

ความต้านทานไฟฟ้าเป็นการบอกคุณสมบัติของสารในการต้านกระแสไฟฟ้าที่จะผ่านได้มากน้อยเพียงใด สารที่มีความต้านทานมากกระแสไฟฟ้าก็จะไหลผ่านไปได้น้อย ส่วนสารที่มีความต้านทานน้อยกระแสไฟฟ้าก็จะไหลผ่านได้มาก ตัวต้านทานเป็นอุปกรณ์ที่ช่วยปรับความต้านทานให้กับวงจร เพื่อช่วยปรับให้กระแสไฟฟ้าให้กับวงจร ตัวต้านทานในวงจรไฟฟ้าและวงจรอิเล็กทรอนิกส์แบ่งออกเป็น 3 ชนิด ได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.1 ตัวต้านทานชนิดค่าคงที่ (Fixed Value Resistor)

ตัวต้านทานชนิดค่าคงที่ จะเป็นตัวต้านทานที่มีค่าความต้านทานคงที่ โครงสร้างภายในทำจากวัสดุซึ่งมีคุณสมบัติเป็นตัวต้านทาน โดยที่ปลายทั้งสองข้างจะต่อลวดตัวนำออกมาและบริเวณผิวด้านนอกจะฉาบด้วยฉนวน มีรูปร่างเป็นทรงกระบอก ขนาดของตัวต้านทานจะแสดงถึงพลังงานที่ตัวต้านทานสามารถทนได้ ซึ่งอยู่ในรูปของความร้อนที่แพร่ออกมา จากการที่ตัวต้านทานทำหน้าที่จำกัดการไหลของกระแสไฟฟ้าหรืออิเล็กตรอน ดังนั้นสภาวะของการต้านทาน หรือขัดขวางการไหลของกระแสไฟฟ้านี้จึงเป็นสาเหตุทำให้เกิดความร้อนขึ้น โดยปริมาณความร้อนที่แพร่กระจายออกมาเมื่อเปรียบเทียบกับหน่วยเวลา จะมีหน่วยเป็นวัตต์ (Watts) และตัวต้านทานแต่ละตัวจะมีค่าอัตราทนกำลัง (Wattage Rating) แตกต่างกันออกไป โดยตัวต้านทานขนาดใหญ่จะสามารถแพร่กระจายความร้อนได้ดีกว่าตัวต้านทานขนาดเล็ก ส่วนค่าความคลาดเคลื่อนเป็นปริมาณความผิดพลาดของค่าความต้านทานที่แตกต่างกันออกไปจากค่าที่กำหนดไว้ ตัวต้านทานชนิดค่าคงที่สามารถแบ่งชนิดตามสารที่ใช้ผลิตได้ดังนี้

2.9.1.1 ตัวต้านทานชนิดคาร์บอนผสม (Carbon Composition Resistor)

การที่เรียกตัวต้านทานชนิดนี้ว่าตัวต้านทานแบบคาร์บอนผสม วัสดุที่นำมาใช้ทำตัวต้านทานนี้เกิดจากการผสมกันระหว่าง ผงคาร์บอนและผงของฉนวน ซึ่งการเปลี่ยนอัตราส่วนผสมของวัสดุทั้งสองชนิดนี้ จะให้ค่าความต้านทานที่ได้เปลี่ยนแปลงไป

2.9.1.2 ตัวต้านทานชนิดฟิล์มคาร์บอน (Carbon Film Resistor)

ตัวต้านทานชนิดนี้ถูกสร้างโดยการเคลือบแผ่นฟิล์มคาร์บอน ที่มีคุณสมบัติของความต้านทานลงบนแกนเซรามิค ซึ่งทำหน้าที่เป็นฉนวน หลังจากนั้นให้ทำการตัดแต่งฟิล์มคาร์บอนที่ได้ให้เป็นรูปวงแหวนรอบแกนเซรามิค ถ้ามีอัตราส่วนของปริมาณเนื้อคาร์บอน มากกว่าฉนวนจะทำให้ค่าความต้านทานที่ได้มีค่าต่ำ แต่ถ้าฉนวนมีอัตราส่วนมากกว่าเนื้อของคาร์บอน ความต้านทานที่ได้ก็จะมีค่าสูง ตัวต้านทานแบบฟิล์มคาร์บอนจะมีค่าความคลาดเคลื่อนต่ำ และสามารถทนต่อการเปลี่ยนแปลงของอุณหภูมิสูงได้ โดยไม่ทำให้ค่าความต้านทานเปลี่ยนแปลงไป นอกจากนั้นสัญญาณรบกวนที่เกิดจากการใช้ตัวต้านทานชนิดนี้ก็มีค่าน้อยกว่า เมื่อเปรียบเทียบกับตัวต้านทานชนิดคาร์บอนผสม

2.9.1.3 ตัวต้านทานชนิดฟิล์มโลหะ (Metal Film Resistor)

ตัวต้านทานชนิดฟิล์มโลหะสร้างได้โดยการพ่นฟิล์มโลหะให้เป็นแผ่นบางๆ ลงบนเซรามิครูปทรงกระบอก จากนั้นจึงตัดแผ่นฟิล์มนี้ โดยให้มีส่วนที่เป็นแผ่นฟิล์มคั่นอยู่กับฉนวนซึ่งเป็นเซรามิค ตัวต้านทานชนิดฟิล์มโลหะนี้จะมีค่าความคลาดเคลื่อนน้อยมาก และยังทนต่อการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่ยนแปลงอุณหภูมิภายนอกได้ดี นอกจากนี้ ยังเกิดสัญญาณรบกวนได้น้อยเมื่อเทียบกับตัวต้านทานชนิดอื่นๆ

2.9.1.4 ตัวต้านทานชนิดไวร์วาวด์ (Wire Wound Resistor)

โครงสร้างภายในของตัวต้านทานชนิดนี้เกิดจากพันขดลวดรอบๆ แกนเซรามิค ซึ่งทำหน้าที่เป็นฉนวน จากนั้นจึงต่อเข้าด้วยลวดตัวนำ จากส่วนหัวและท้ายออกมา สำหรับค่าความต้านทานสามารถเปลี่ยนแปลงได้ โดยขึ้นอยู่กับความยาวและขนาดของขดลวดที่ใช้พันตัวต้านทานแบบไวร์วาวด์ ส่วนมากนิยมใช้ในงานที่ต้องการค่าความต้านทานต่ำๆ ทั้งนี้เพื่อให้กระแสไหลผ่านได้ดี ดังนั้นจึงมีการออกแบบให้มีขนาดใหญ่เพื่อช่วยให้สามารถกระจายความร้อนได้มากกว่า ตัวต้านทานแบบไวร์วาวด์นี้จะมีค่าความคลาดเคลื่อนประมาณ 1%

2.9.1.5 ตัวต้านทานชนิดออกไซด์ของโลหะ (Metal Oxide Resistor)

ตัวต้านทานชนิดนี้มีโครงสร้างตัวต้านทานที่เคลือบด้วยออกไซด์โลหะประเภทดีบุกลงบนวัสดุที่ใช้เป็นฉนวน โดยอัตราส่วนของออกไซด์โลหะ จะเป็นตัวกำหนดค่าความต้านทานให้กับตัวต้านทานชนิดนี้ คุณสมบัติพิเศษสำหรับตัวต้านทานชนิดออกไซด์ของโลหะ คือ สามารถทนต่อการเปลี่ยนแปลงอุณหภูมิได้ดี

2.9.1.6 ตัวต้านทานชนิดแผ่นฟิล์มหนา (Thick - Film Resistor)

ตัวต้านทานแบบฟิล์มหนามีอยู่ 2 แบบ คือ แบบ SIP (Single in - line Package) และ DIP (Dual in - Line Package) ตัวต้านทานแบบ SIP จะต่อลวดตัวนำออกจากความต้านทานภายในเพียงแถวเดียว ส่วนตัวต้านทานแบบ DIP จะมีลวดตัวนำ 2 แถว ต่อออกมาภายนอก ซึ่งตัวต้านทานแบบฟิล์มหนาทั้งสองแบบจะได้รับการปรับแต่งให้ค่าความคลาดเคลื่อนประมาณ 2% โดยค่าความต้านทานที่ใช้ในงานทั่วไปของตัวต้านทานชนิดนี้อยู่ระหว่าง 22 โอห์ม ถึง 2.2 เมกะโอห์ม และมีอัตราทนกำลัง ประมาณ ½ วัตต์

2.9.1.7 ตัวต้านทานชนิดแผ่นฟิล์มบาง (Thin - Film Resistor)

โครงสร้างของตัวต้านทานแบบนี้ทำมาจากแผ่นฟิล์มบาง มีลักษณะรูปร่างเหมือนกับตัวไอซี (Integreate Circuit) ใช้เทคโนโลยี SMT (Surface Mount Technology) ในการผลิตเช่นเดียวกับตัวต้านทานแบบแผ่นฟิล์มหนา โดยส่วนใหญ่จะมีขาทั้งหมด 16 ขา การใช้งานต้องบัดกรีเข้ากับแผ่นลายวงจร อัตราทนกำลัง 50 มิลลิวัตต์ มีค่าความคลาดเคลื่อนบวกลบ 0.1 % และอัตราทนกำลัง 100 มิลลิวัตต์ จะมีค่าความคลาดเคลื่อนบวกลบ 5 % ที่แรงดันไฟฟ้ากระแสตรงสูงสุดไม่เกิน 50 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.2 ตัวต้านทานชนิดปรับค่าได้ (Variable Value Resistor)

โครงสร้างของตัวต้านทานแบบนี้มีลักษณะคล้ายกับแบบไวร์วาวด์ แต่โดยส่วนใหญ่บริเวณลวดตัวนำจะไม่เคลือบด้วยสารเซรามิค และมีช่องว่างทำให้มองเห็นเส้นลวดตัวนำ เพื่อทำการลัดเข็มขัดค่อมตัวต้านทาน โดยจะมีขาปรับให้สัมผัสเข้ากับจุดใดจุดหนึ่ง บนเส้นลวดของความต้านทาน ตัวต้านทานแบบนี้ส่วนใหญ่มีค่าความต้านทานต่ำ แต่อัตราทนกำลังวัตต์สูง

2.9.3 ตัวต้านทานชนิดเปลี่ยนค่าได้ (Variable Resistor)

โครงสร้างภายในทำมาจากคาร์บอน เซรามิค หรือพลาสติกตัวนำ ใช้ในงานที่ต้องการเปลี่ยนค่าความต้านทานบ่อยๆ เช่น โพเทนชิโอมิเตอร์ หรือ พอต (Pot) สำหรับชนิดที่มีแกนเลื่อนค่าความต้านทาน หรือแบบที่มีแกนหมุนเปลี่ยนค่าความต้านทานคือโวลลุ่ม (Volume) ส่วนอีกแบบหนึ่งเป็นแบบที่ไม่มีแกนปรับโดยทั่วไปจะเรียกว่า โวลลุ่มเกือกม้า หรือทิมพอต (Trimpot)

2.9.4 ตัวต้านทานชนิดพิเศษ (Special Resistor)

2.9.4.1 ตัวต้านทานชนิดเปลี่ยนตามอุณหภูมิ (Thermistor)

ตัวต้านทานชนิดนี้ จะมีค่าความต้านทานเปลี่ยนแปลงไปตามอุณหภูมิ

2.9.4.2 ตัวต้านทานชนิดเปลี่ยนค่าตามความสว่าง (Light Dependent Resistor–LDR)

ตัวต้านทานชนิดนี้ เป็นตัวต้านทานที่ไวต่อแสง โดยค่าความต้านทานจะลดลง เมื่อความเข้มของแสงที่ตกกระทบมีค่ามาก ส่วนใหญ่จะใช้เป็นตัวเปลี่ยนแปลงความเข้มของแสงให้เป็นสัญญาณทางไฟฟ้า

2.9.4.3 ตัวต้านทานที่ทำจากคาร์บอนดำผสมซิลิโคน (Carbon Black Filled Silicone Rubber)

ตัวต้านทานชนิดนี้ เป็นตัวต้านทานที่มีส่วนประกอบของคาร์บอนดำกับซิลิโคน ซึ่งตัวต้านทานชนิดนี้จะมีลักษณะคล้ายยาง และมีค่าความต้านทานที่เปลี่ยนไปอย่างสม่ำเสมอเมื่อมีแรงกดในทิศทางเดียว ดังนั้นสารประกอบชนิดนี้จึงมีความไวต่อการยืดหยุ่น จึงทำให้นำมาใช้ในงานทางด้านวิศวกรรม โดยเมื่อต้องการให้ค่าความต้านทานของเซนเซอร์ชนิดนี้เปลี่ยนไปนั่นจะหมายถึงการเปลี่ยนของค่าความเค้น จะต้องทำให้เกิดความเครียดอย่างต่อเนื่อง และเมื่อคลายความเครียดออกก็จะได้ค่าความต้านทานกลับมาคงเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

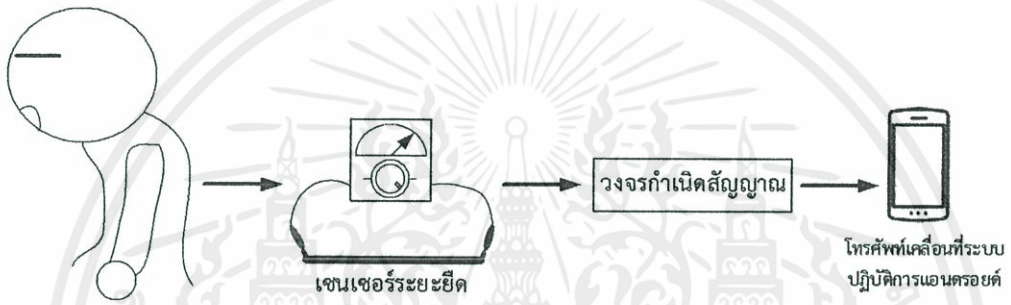
การออกแบบและการจัดปริญญานิพนธ์

ในระบบการทำงานของเครื่องช่วยเตือนจัดบุคลิกท่าทางของหลังสามารถแบ่งได้เป็น 3 ส่วนหลักๆ โดยมีบล็อกการทำงานดังรูปที่ 3.1 ดังนี้

1) เซนเซอร์ระยะยืด

2) วงจรกำเนิดสัญญาณ

3) โปรแกรมแอปพลิเคชั่นตรวจจับความถี่บนโทรศัพท์มือถือระบบปฏิบัติการแอน-ดรอยด์



รูปที่ 3.1 บล็อกการทำงานรวมของระบบ

เมื่อผู้สวมใส่ขยับลำตัว เซนเซอร์ระยะยืดจะตรวจจับท่าทางการห่อไหล่ที่ผิดรูปทรงไปจากปกติ จากนั้นวงจรกำเนิดสัญญาณจะสร้างสัญญาณที่มีความถี่ค่าหนึ่งออกมาเพื่อส่งเข้าโทรศัพท์มือถือระบบปฏิบัติแอนดรอยด์โดยผ่านสายนำสัญญาณ โปรแกรมแอปพลิเคชั่นบนโทรศัพท์มือถือระบบปฏิบัติแอนดรอยด์จะทำการประมวลผลเพื่อตรวจจับท่าการห่อไหล่ที่ผิดปกติไปจากค่าความถี่ของสัญญาณที่ได้รับ โดยจะทำการบันทึกค่าจำนวนครั้งที่ผิดปกติลงในหน่วยความจำไมโครเอสดีการ์ด

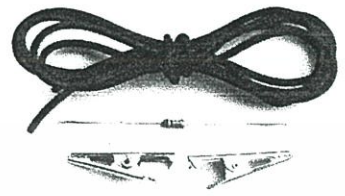
3.1 การออกแบบ

3.1.1 การออกแบบเซนเซอร์ระยะยืด

ผู้จัดทำปริญญานิพนธ์ได้เลือกใช้ตัวต้านทานชนิดพิเศษ เป็นตัวต้านทานที่ทำจากคาร์บอนดำผสมซิลิโคนนำมาใช้ติดกับแผ่นหลังของผู้สวมใส่ และในปริญญานิพนธ์นี้จะเรียกอุปกรณ์ชิ้นนี้ว่า “เซนเซอร์ระยะยืด” ซึ่งอุปกรณ์ดังกล่าวเป็นเสมือนตัวต้านทานปรับค่าได้ชนิดหนึ่ง ที่มีค่าความต้านทานเปลี่ยนไปตามระยะที่ยืดออก โดยปกติค่าความต้านทานของเซนเซอร์ระยะยืดจะมีค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

350 โอห์ม ถึง 400 โอห์มต่อนิ้ว หรือ 140 โอห์ม ถึง 160 โอห์มต่อเซนติเมตร มีลักษณะดังรูปที่ 3.2

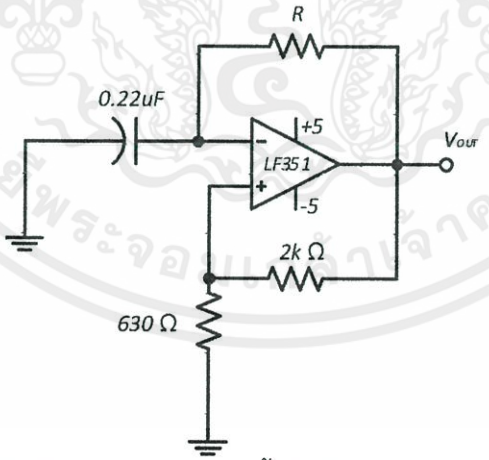


รูปที่ 3.2 เซนเซอร์ระยะยืด

3.1.2 การออกแบบวงจรกำเนิดสัญญาณ

ในการทดลองนั้นก่อนที่จะนำสัญญาณเข้าโทรศัพท์มือถือ จะต้องกำหนดขนาดสัญญาณอินพุตให้เท่ากับ 100 มิลลิโวลต์ พีคทูพีค และยกระดับสัญญาณขึ้น 1 โวลต์ เพื่อทำให้แอปพลิเคชั่นที่สร้างไว้บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์สามารถตรวจจับสัญญาณอินพุตได้ จึงทำการออกแบบและสร้างวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมโดยใช้ไอซีเบอร์ 351 ซึ่งจะกำหนดให้ตำแหน่ง R คือเซนเซอร์ระยะยืด

3.1.2.1 การออกแบบวงจรกำเนิดสัญญาณในส่วนที่มีรูปแบบของการติดเซนเซอร์อยู่ในแนวตามขวางของแผ่นหลัง จะกำหนดค่าตัวต้านทานและตัวเก็บประจุได้ดังรูปที่ 3.3



รูปที่ 3.3 วงจรกำเนิดสัญญาณแบบอะสเตเบิ้ลมัลติไวเบรเตอร์โดยใช้ไอซีเบอร์ 351 ที่เซนเซอร์ถูกติดอยู่ในแนวตามขวางของแผ่นหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยผู้จัดทำปริญญานิพนธ์จะออกแบบให้เซนเซอร์ระยะยืดมีค่าความต้านทานเริ่มต้นเมื่อไม่มีการยืดมีค่าเท่ากับ 1.85 กิโลโอห์ม จึงทำให้สามารถคำนวณหาค่าความถี่ของสัญญาณเอาต์พุตจากสมการที่ 2.6 ได้ดังนี้

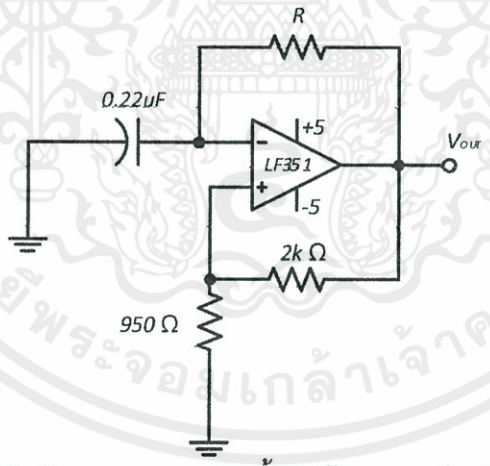
$$T = 2\times1.85\times10^{3}\times0.22\times10^{-6}\ln\left(\frac{1+\dfrac{630}{630+2\times10^{3}}}{1-\dfrac{630}{630+2\times10^{3}}}\right)$$

$$= 0.397\ ms$$

$$f = \frac{1}{0.397\times10^{-3}}$$

$$= 2.51\ kHz$$

3.1.2.2 การออกแบบวงจรกำเนิดสัญญาณในส่วนที่มีรูปแบบของการติดเซนเซอร์อยู่ในแนวตั้งตามแนวกระดูกสันหลัง จะกำหนดค่าตัวต้านทานและตัวเก็บประจุได้ดังรูปที่ 3.4



รูปที่ 3.4 วงจรกำเนิดสัญญาณแบบอะสเตเบิ้ลมัลติไวเบรเตอร์โดยใช้ไอซีเบอร์ 351 ที่เซนเซอร์ถูกติดอยู่ในแนวตั้งตามแนวของกระดูกสันหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยผู้จัดทำปริญญานิพนธ์จะออกแบบให้เซนเซอร์ระยะยืดมีค่าความต้านทานเริ่มต้นเมื่อไม่มีการยืดมีค่าเท่ากับ 962 โอห์ม จึงทำให้สามารถคำนวณหาค่าความถี่ของสัญญาณเอาต์พุตจากสมการที่ 2.6 ได้ดังนี้

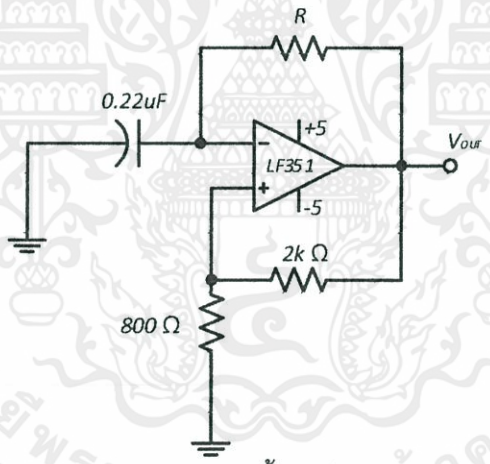
$$T = 2\times962\times0.22\times10^{-6}\ln\left(\frac{1+\frac{950}{950+2\times10^{3}}}{1-\frac{950}{950+2\times10^{3}}}\right)$$

$$= 0.283\ ms$$

$$f = \frac{1}{0.283\times10^{-3}}$$

$$= 3.54\ kHz$$

3.1.2.3 การออกแบบวงจรกำเนิดสัญญาณในส่วนที่มีรูปแบบของการติดเซนเซอร์อยู่ในแนวตั้งทางด้านหน้าของลำตัว จะกำหนดค่าตัวต้านทานและตัวเก็บประจุได้ดังรูปที่ 3.5



รูปที่ 3.5 วงจรกำเนิดสัญญาณแบบอะสเตเบิ้ลมัลติไวเบรเตอร์โดยใช้ไอซีเบอร์ 351 ที่เซนเซอร์ถูกติดอยู่ในแนวตั้งทางด้านหน้าของลำตัว

โดยผู้จัดทำปริญญานิพนธ์จะออกแบบให้เซนเซอร์ระยะยืดมีค่าความต้านทานเริ่มต้นเมื่อไม่มีการยืดมีค่าเท่ากับ 2.14 กิโลโอห์ม จึงทำให้สามารถคำนวณหาค่าความถี่ของสัญญาณเอาต์พุตจากสมการที่ 2.6 ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T = 2 \times 2.1 \times 10^{3} \times 0.22 \times 10^{-6} \ln \left(\frac{1 + \frac{800}{800 + 2 \times 10^{3}}}{1 - \frac{800}{800 + 2 \times 10^{3}}} \right)$$

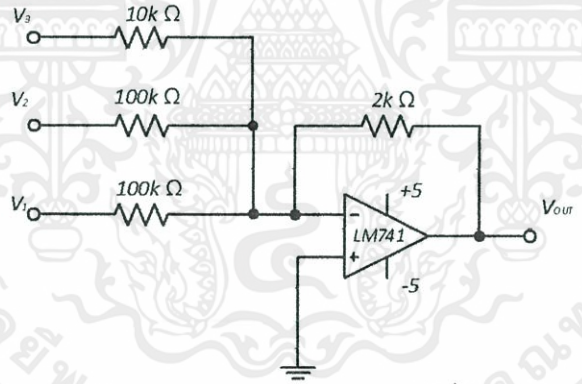
$$= 0.553 \; ms$$

$$f = \frac{1}{0.553 \times 10^{-3}}$$

$$= 1.81 \; kHz$$

3.1.3 การออกแบบวงจรขยายสัญญาณผลรวม

3.1.3.1 การออกแบบวงจรขยายสัญญาณผลรวมที่นำไปใช้ในส่วนของเซนเซอร์ 2 ตัว ที่ติดอยู่ที่ด้านหลังของลำตัวทั้งในแนวตามขวาง และแนวตั้งตามแนวของกระดูกสันหลัง โดยออกแบบวงจรขยายสัญญาณผลรวมเพื่อรวมสัญญาณขนาดแอมปลิจูดประมาณ 100 มิลลิโวลต์ พีคทูพีค 2 สัญญาณ กับสัญญาณไฟตรง 1 โวลต์ ให้เป็นสัญญาณเอาต์พุตที่ถูกยกระดับสัญญาณให้มีค่าเป็น -1 โวลต์ ซึ่งจะกำหนดค่าตัวต้านทานและตัวเก็บประจุได้ดังรูปที่ 3.6



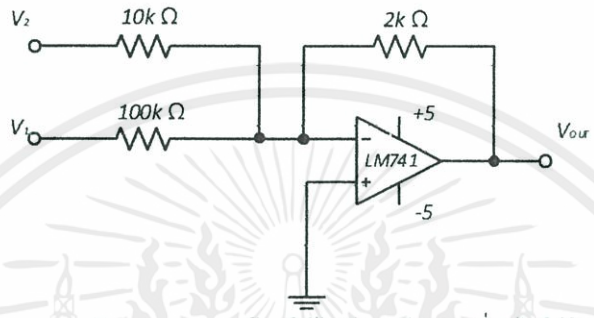
รูปที่ 3.6 วงจรขยายสัญญาณผลรวมโดยใช้ไอซีเบอร์ 741 ที่นำไปใช้ในส่วนของเซนเซอร์ 2 ตัว ที่ติดอยู่ที่ด้านหลังของลำตัว

โดยที่ V_1 และ V_2 มีค่าประมาณ 100 มิลลิโวลต์ พีคทูพีค V_3 มีค่า 5 โวลต์ และสามารถคำนวณหาสัญญาณเอาต์พุตได้จากสมการที่ 2.9 ดังนี้

$$V_{OUT} = -(V_1 + V_2) \tag{2.9}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3.2 การออกแบบวงจรขยายสัญญาณผลรวมที่นำไปใช้ในส่วนของเซนเซอร์ตัวที่ติดอยู่ที่ด้านหน้าของลำตัวตามแนวตั้ง โดยออกแบบวงจรขยายสัญญาณผลรวมเพื่อรวมสัญญาณขนาดแอมปลิจูดประมาณ 100 มิลลิโวลต์ พีคทูพีค กับสัญญาณไฟตรง 1 โวลต์ ให้เป็นสัญญาณเอาต์พุตที่ถูกยกระดับสัญญาณให้มีค่าเป็น -1 โวลต์ ซึ่งจะกำหนดค่าตัวต้านทานและตัวเก็บประจุได้ดังรูปที่ 3.7



รูปที่ 3.7 วงจรขยายสัญญาณผลรวมโดยใช้ไอซีเบอร์ 741 ที่นำไปใช้ในส่วนของเซนเซอร์ด้านหน้าลำตัวตามแนวตั้ง

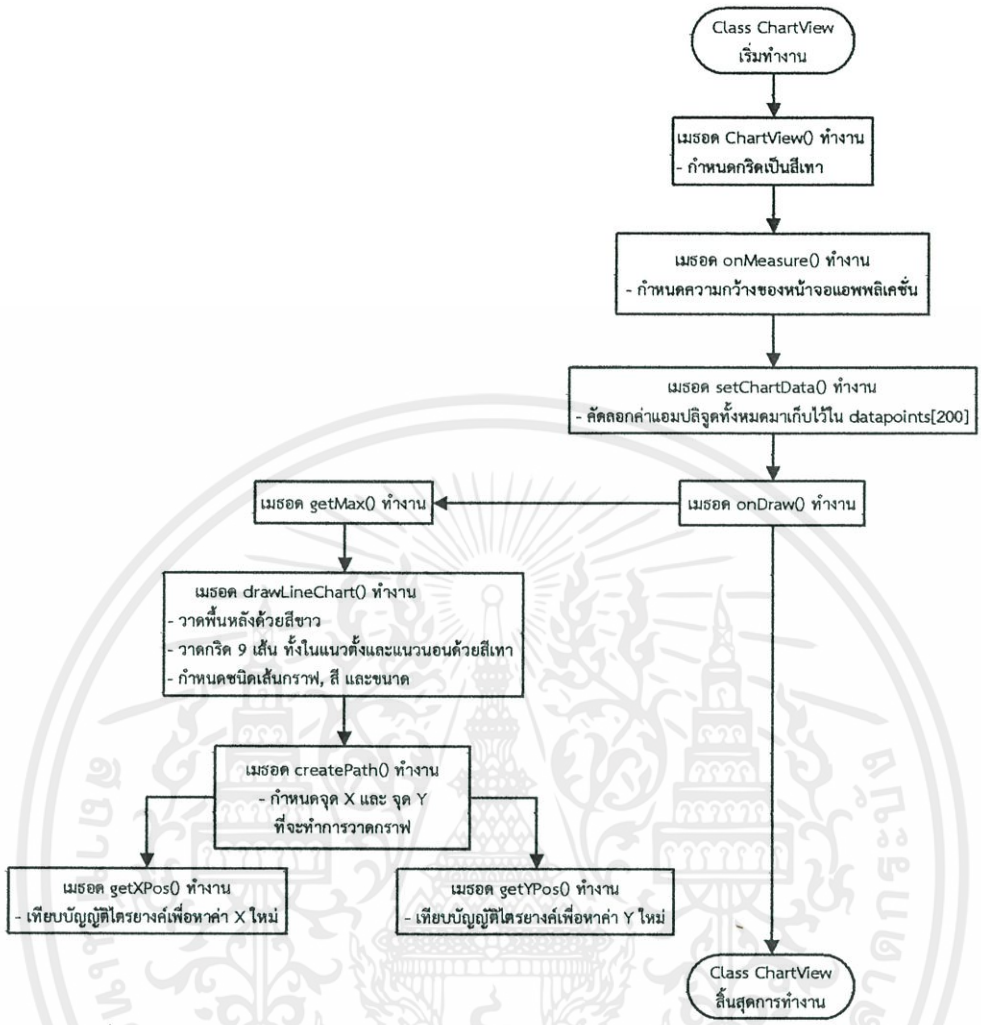
โดยที่ V_1 มีค่าประมาณ 100 มิลลิโวลต์ พีคทูพีค V_2 มีค่า 5 โวลต์ และสามารถคำนวณหาสัญญาณเอาต์พุตได้จากสมการที่ 2.9

3.1.4 การออกแบบโปรแกรมแอปพลิเคชั่น

3.1.4.1 การออกแบบโปรแกรมชุดคำสั่งในส่วนของการพลอตกราฟบนแอปพลิเคชั่น

ผู้จัดทำได้เลือกใช้คลาส ChartView เพื่อใช้เมธอดและฟังก์ชั่นต่างๆ ในการพลอตกราฟ โดยมีการออกแบบแผนผังการทำงานของโปรแกรมชุดคำสั่งดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แผนผังของโปรแกรมชุดคำสั่งในส่วนของการพลอตกราฟบนแอปพลิเคชั่น

จากแผนผังที่แสดงในรูปที่ 3.8 สามารถอธิบายขั้นตอนการทำงานของโปรแกรมชุดคำสั่งได้ดังนี้

1) เมธอด ChartView() จะกำหนดสีของเส้นตารางกริด (Grid) โดยใช้คำสั่ง grid_paint.setColor(Color.*rgb*(int red, int green, int blue)) กริดดังกล่าวจะถูกสร้างเพื่อความสวยงามเท่านั้น โดยมีโปรแกรมชุดคำสั่งดังนี้

```
private Paint grid_paint = new Paint();
public ChartView(Context context, AttributeSet attrs) {
        super(context, attrs);
                grid_paint.setColor(Color.rgb(100, 100, 100)); }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) เมธอด onMeasure() เป็นเมธอดที่ใช้กำหนดความกว้างของหน้าจอแอปพลิเคชั่น กำหนดให้มีขนาดความกว้างและความสูง 320x240 พิกเซล ซึ่งใช้ฟังก์ชัน MeasureSpec.*getSize*() เพื่อกำหนดความสูงและความกว้าง โดยมีโปรแกรมชุดคำสั่งดังนี้

```
private int width = 320;
private int heigth = 240;
protected void onMeasure(int widthMeasureSpec, int heightMeasureSpec) {
      super.onMeasure(widthMeasureSpec, heightMeasureSpec);
      width = MeasureSpec.getSize(widthMeasureSpec);
      heigth = MeasureSpec.getSize(heightMeasureSpec);
      this.setMeasuredDimension(width , heigth);
   }
```

3) เมธอด setChartData() เป็นเมธอดที่รับข้อมูล จากนั้นจะทำการคัดลอกข้อมูลจากอาร์เรย์ที่เก็บค่ามาจากสัญญาณอินพุต โดยจะคัดลอกข้อมูลทั้งชุดด้วยเมธอด clone() และรีเฟรชค่าข้อมูลเพื่อรับค่าใหม่จะใช้เมธอด invalidate() โดยมีโปรแกรมชุดคำสั่งดังนี้

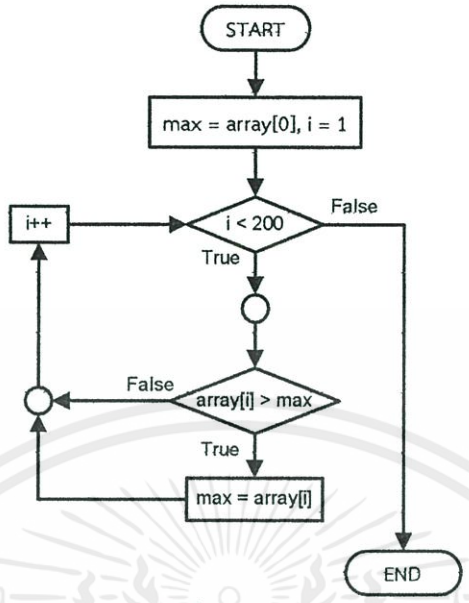
```
private float[] datapoints = new float[] {};
public void setChartData(float[] datapoints) {
      this.datapoints = datapoints.clone();
      invalidate();   }
```

4) เมธอด onDraw() เป็นเมธอดที่ประกอบด้วยเมธอด getMax() และเมธอด drawLineChart() โดยมีโปรแกรมชุดคำสั่งดังนี้

```
protected void onDraw(Canvas canvas) {
        float maxValue = getMax(datapoints);
        drawLineChart(canvas, maxValue);
   }
```

โดยเมธอด getMax() เป็นเมธอดที่จะคำนวณหาค่าแอมปลิจูดที่สูงที่สุด โดยเมื่อรับค่าแอมปลิจูดมาเก็บไว้ใน array[i] แล้วจะทำการเช็คค่าถัดไปเช่น ถ้าค่า array[i] ตัวถัดไปมากกว่าค่าเดิมก็จะทำการเก็บค่าดังกล่าวไว้ที่ตัวแปร max และเมื่อเก็บถึงค่าสุดท้ายโปรแกรมก็จะส่งค่าแอมปลิจูดสูงสุดกลับไปยังเมธอด onDraw() โดยมีแผนผังโปรแกรมชุดคำสั่งดังรูปที่ 3.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แผนผังของโปรแกรมชุดคำสั่งส่วนของการคำนวณหาค่าแอมปลิจูดสูงสุด

โดยมีโปรแกรมชุดคำสั่งดังนี้

```
private float getMax(float[] array) {
      float max = array[0];
      for (int i = 1; i < array.length; i++) {
        if (array[i] > max) {
          max = array[i]; }
      }
      return max; }
```

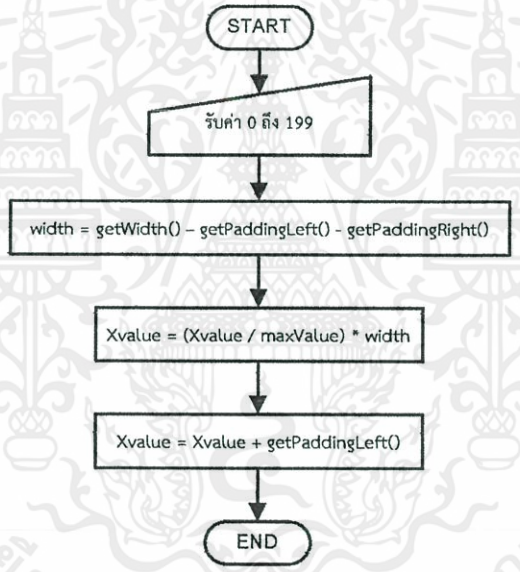
โดยเมธอด drawLineChart() เป็นเมธอดที่จะทำการวาดพื้นหลังแอปพลิเคชั่นด้วยสีขาวด้วยเมธอด drawColor() วาดกริด 9 เส้น ทั้งในแนวตั้งและแนวนอน canvas.drawLine() กำหนดชนิดเส้นกราฟจะใช้เมธอด setStyle() ขนาดจะใช้เมธอด setStrokeWidth() สีจะใช้เมธอด setColor() และใช้เมธอด drawPath() เพื่อทำการพลอตกราฟจากคลาส Canvas โดยมีโปรแกรมชุดคำสั่งดังนี้

```
private void drawLineChart(Canvas canvas, float maxValue) {
        Path path = createPath(maxValue);
        canvas.drawColor(Color.WHITE);
          for(int vertical = 1; vertical<10; vertical++){
        canvas.drawLine(vertical*32+1, 1, vertical*32+1, 241, grid_paint); }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for(int horizontal = 1; horizontal<10; horizontal++) {
        canvas.drawLine(1,horizontal*24+1, 320, horizontal*24+1, grid_paint); }
    paint.setStyle(Style.STROKE);
    paint.setStrokeWidth(2);
    paint.setColor(Color.BLACK);
    canvas.drawPath(path, paint);
}
```

นอกจากนี้จะเห็นว่าภายในเมธอด drawLineChart() เป็นเมธอดที่สั่งให้เมธอด createPath() ทำงาน จากนั้นเมธอดที่ใช้กำหนดจุด X และ Y คือ เมธอด moveTo() โดยจะใช้ออปเจ็คของคลาส Path เพื่อเก็บค่า X และ Y มาพลอตกราฟบนหน้าจอโปรแกรมแอปพลิเคชั่น โดยเมธอด moveTo() จะประกอบไปด้วยเมธอด getXPos() และ getYPos() โดยเมธอด getXPos() จะเป็นเมธอดที่ใช้กำหนดจุด X โดยมีแผนผังโปรแกรมชุดคำสั่งดังรูปที่ 3.10



รูปที่ 3.10 แผนผังของโปรแกรมชุดคำสั่งในส่วนของการคำนวณหาค่า X ใหม่

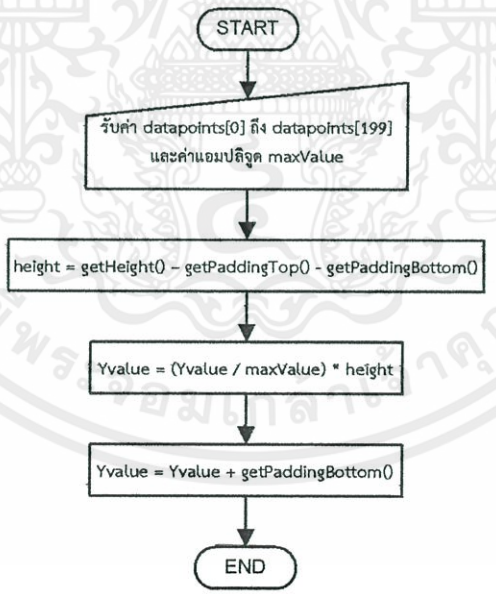
เมธอด getXPos() จะรับค่า 0 ถึง 199 มาเก็บในตัวแปร value เพื่อกำหนดจุด X เมธอดนี้ถูกใช้เพื่อกำหนด X จุดใหม่ โดยจะเว้นระยะขอบในแนวนอนของหน้าจอแอปพลิเคชั่นไว้ค่าหนึ่ง โดยจะใช้เมธอด getWidth() ซึ่งเป็นค่าความกว้างของจอแอปพลิเคชั่นทั้งหมดนั่นก็คือ 320 พิกเซล ส่วนเมธอด getPaddingLeft() และเมธอด getPaddingRight() เป็นเมธอดที่จะนำมาใช้คำนวณเพื่อกำหนดระยะขอบของจอแอปพลิเคชั่น ซึ่งการกำหนดความกว้างของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าจอใหม่นั้นจะทำโดยการนำเมธอด getWidth() มาลบกับเมธอด getPaddingLeft() และเมธอด getPaddingRight() ก็จะได้ความกว้างของหน้าจอแอปพลิเคชั่นใหม่ จากนั้นจะกำหนดค่า maxValue ใหม่ โดยค่า datapoints.length จะมีค่าเท่ากับ 200 ซึ่งจะกำหนดค่า value ใหม่โดยให้เท่ากับ (value / maxValue) * width ซึ่งจะเป็นการเทียบบัญญัติไตรยางค์เพื่อให้ได้ค่า value ใหม่ ที่จะนำไปใช้กำหนดจุดพลอตกราฟ จากนั้นนำค่า value ไปรวมกับค่าระยะขอบด้านซ้าย getPaddingLeft() จะได้ค่า value ที่เว้นระยะจากขอบซ้ายออกมา จากนั้นจะส่งค่า value กลับไปยังเมธอด createPath() เพื่อกำหนดจุดทั้งหมดก่อนที่จะนำมาพลอตกราฟ โดยมีโปรแกรมชุดคำสั่งดังนี้

```
private float getXPos(float value) {
    float width = getWidth() - getPaddingLeft() - getPaddingRight();
    float maxValue = datapoints.length;
    Xvalue = (value / maxValue) * width;
    Xvalue += getPaddingLeft();
    return Xvalue;
}
```

ในส่วนของเมธอด getYPos() จะกำหนดจุด Y โดยมีแผนผังโปรแกรมชุดคำสั่งดังรูปที่ 3.11



รูปที่ 3.11 แผนผังของโปรแกรมชุดคำสั่งในส่วนของการคำนวณหาค่า Y ใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

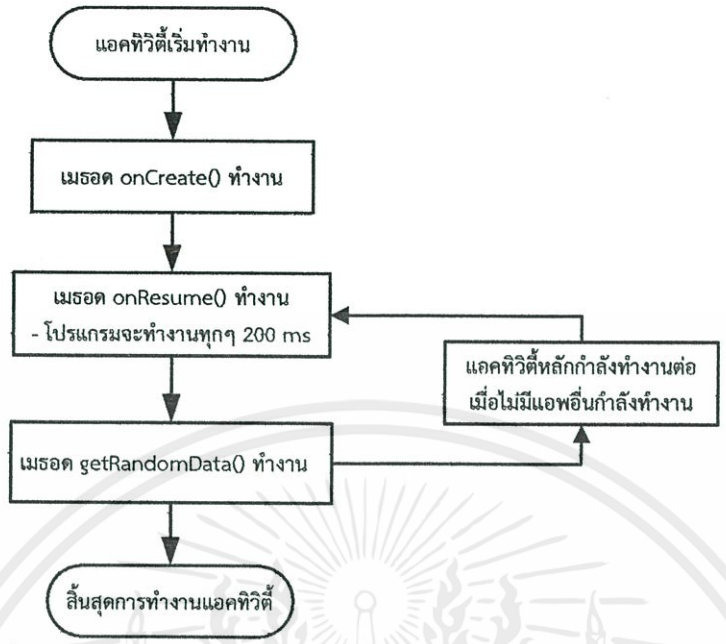
เมธอด getYPos() จะรับค่า datapoints[0] มาเก็บไว้ในตัวแปร value และรับค่าแอมปลิจูดที่สูงที่สุดคือ MaxValue เพื่อกำหนดจุด Y เมธอดนี้ถูกใช้เพื่อกำหนดจุด Y ใหม่ โดยจะเว้นระยะขอบในแนวตั้งของหน้าจอแอปพลิเคชั่น โดยจะใช้เมธอด getHeight() ซึ่งเป็นค่าความสูงของจอแอปพลิเคชั่นทั้งหมดนั่นก็คือ 240 พิกเซล ส่วนเมธอด getPaddingTop() และเมธอด getPaddingBottom() เป็นเมธอดที่จะนำมาใช้คำนวณเพื่อกำหนดระยะขอบของจอแอปพลิเคชั่น ซึ่งการกำหนดความสูงของหน้าจอใหม่นั้นจะทำโดยการนำเมธอด getWidth() มาลบกับเมธอด getPaddingTop() และเมธอด getPaddingBottom() ก็จะได้ความสูงของหน้าจอแอปพลิเคชั่นใหม่ จากนั้นจะใช้ค่า maxValue ซึ่งเป็นค่าแอมปลิจูดสูงสุด และจะใช้ค่าแอมปลิจูดของแต่ละตัวอย่างมาเก็บไว้ในตัวแปร value เพื่อนำไปเทียบบัญญัติไตรยางค์นั่นก็คือ (value / maxValue) * heigth เพื่อให้ได้ค่า value ใหม่ที่จะนำไปใช้กำหนดจุดพลอตกราฟ จากนั้นนำค่า value ไปรวมกับค่าระยะขอบด้านซ้าย getPaddingBottom() จะได้ค่า value ที่เว้นระยะจากขอบล่างออกมา จากนั้นจะส่งค่า value กลับไปยังเมธอด createPath() เพื่อกำหนดจุดทั้งหมดก่อนที่จะนำไปพลอตกราฟ ซึ่งมีโปรแกรมชุดคำสั่งดังนี้

```
private float getYPos(float value, float maxValue) {
    float height = getHeight() - getPaddingTop() - getPaddingBottom();
    Yvalue = (value / maxValue) * height;
    Yvalue += getPaddingBottom();
    return value; }
```

3.1.4.2 การออกแบบโปรแกรมชุดคำสั่งการพลอตกราฟสัญญาณรูปสี่เหลี่ยม

ในการออกแบบแอปพลิเคชั่นพลอตกราฟสัญญาณรูปสี่เหลี่ยม ต้องการออกแบบเพื่อใช้ในทดลองพลอตกราฟบนแอปพลิเคชั่น เพื่อแสดงว่าโปรแกรมแอปพลิเคชั่นสามารถพลอตกราฟจากข้อมูลเพื่อมาแสดงผลบนหน้าจอโทรศัพท์มือถือได้จริง รูปที่ 3.12 จะแสดงแผนผังของโปรแกรมชุดคำสั่งการพลอตกราฟสัญญาณรูปสี่เหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 แผนผังของโปรแกรมชุดคำสั่งการพลอตกราฟสัญญาณรูปสี่เหลี่ยม

เมื่อแอคทิวิตี้เริ่มการทำงานสามารถอธิบายขั้นตอนการทำงานของโปรแกรมชุดคำสั่งได้ดังนี้

1) เมธอด onCreate() จะถูกตั้งค่าการพลอตกราฟไว้ด้วยออป-เจ็ค lineChart ซึ่งจะใช้คลาส ChartView โดยจะกำหนดให้ออปเจ็ค lineChart ทำงานบนแอปพลิเคชั่นโดยใช้คำสั่ง findViewById(R.id.*lineChartView1*) โดย *lineChartView1* จะเป็นส่วนที่สร้างใน activity_main.xml เพื่อใช้แสดงผลต่างๆบนแอปพลิเคชั่น ส่วนเมธอด setChartData() จะถูกกำหนดค่าในเมธอด onCreate() โดยจะเป็นเมธอดที่ใช้รับค่าข้อมูลทั้งชุดเพื่อนำไปพลอตกราฟ โดยจะมีโปรแกรมชุดคำสั่งดังนี้

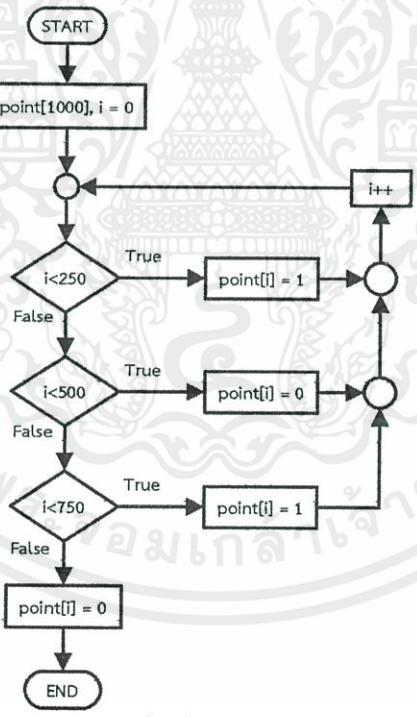
```
ChartView lineChart;
protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        lineChart = (ChartView) findViewById(R.id.lineChartView1);
        lineChart.setChartData(getRandomData());
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) เมธอด onResume() จะเป็นเมธอดที่ใช้ตั้งค่าดีเลย์สำหรับการทำงานของแอคทิวิตี้ใน 1 รอบ ซึ่งผู้จัดทำได้ตั้งค่าให้โปรแกรมทำงานทุกๆ 200 มิลลิวินาที โดยมีโปรแกรมชุดคำสั่งดังนี้

```
private final Handler mHandler = new Handler();
private Runnable mTimer1;
public void onResume() {
        super.onResume();
        mTimer1 = new Runnable() {
                public void run() {
                        lineChart.setChartData(getRandomData());
                        mHandler.postDelayed(this, 200); } };
        mHandler.postDelayed(mTimer1, 200); }
```

3) เมธอด getRandomData() จะสร้างค่าลอจิก 0 และลอจิก 1 เพื่อทดลองพลอตกราฟสี่เหลี่ยมบนโปรแกรมแอปพลิเคชั่น ซึ่งมีแผนผังโปรแกรมชุดคำสั่งดังรูปที่ 3.13



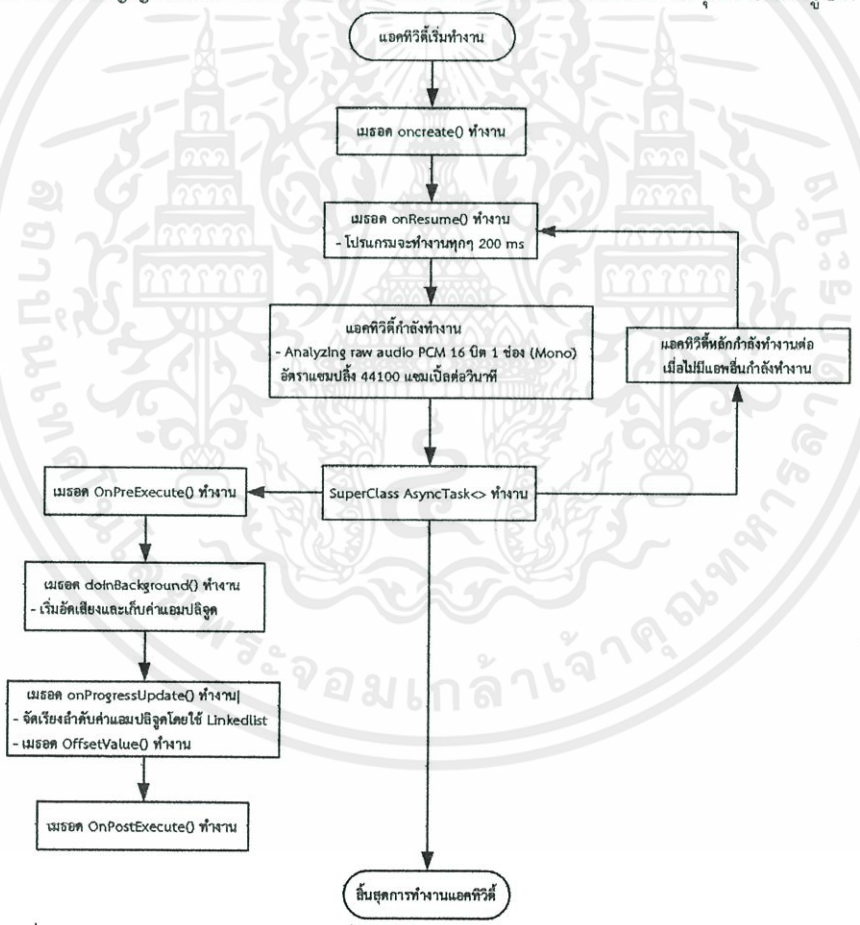
รูปที่ 3.13 แผนผังของโปรแกรมชุดคำสั่งเพื่อจำลองสร้างสัญญาณสี่เหลี่ยมบนแอปพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจะมีโปรแกรมชุดคำสั่งดังนี้

```
private float[] getRandomData(){
        float[] point = new float[1000];
        for (int i = 0; i < point.length; i++) {
                if (i < point.length / 4) point[i] = 1;
                else if (i < point.length / 2) point[i] = 0;
                else if (i < point.length / 1.33) point[i] = 1;
                else    point[i] = 0; }
return point; }
```

3.1.4.3 การออกแบบการทำงานของโปรแกรมแอปพลิเคชั่นรับสัญญาณเสียงแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ ในการออกแบบโปรแกรมแอปพลิเคชั่นรับสัญญาณเสียงในส่วนของแอคทิวิตี้หลัก มีแผนผังโปรแกรมชุดคำสั่งดังรูปที่ 3.14



รูปที่ 3.14 แผนผังโปรแกรมชุดคำสั่งของแอปพลิเคชั่นรับสัญญาณเสียงแบบเรียลไทม์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อแอคทิวิตี้หลักเริ่มการทำงานสามารถอธิบายขั้นตอนการทำงานของโปรแกรมชุดคำสั่งได้ดังนี้

1) เมธอด onCreate() จะถูกตั้งค่าการพลอตกราฟไว้ด้วยออปเจ็คชื่อlineChart ซึ่งจะใช้คลาส ChartView โดยจะกำหนดให้ออปเจ็ค lineChart ทำงานบนแอปพลิเคชั่นโดยใช้คำสั่ง findViewById(R.id.*lineChartView1*) โดยการจะใช้ออปเจ็ค การแสดงผลชื่อ *lineChartView1* จะถูกกำหนดค่าอยู่ใน activity_main.xml เพื่อแสดงผลบนแอปพลิเคชั่น ส่วนเมธอด setChartData() จะถูกกำหนดค่าในเมธอด onCreate() โดยจะเป็นเมธอดที่ใช้รับค่าข้อมูลทั้งชุดเพื่อนำไปพลอตกราฟ ซึ่งในเมธอด onCreate() นี้จะมีการตั้งค่าพารามิเตอร์การอัดเสียงแบบ RAW ออดิโอ แบบ 1 ช่องสัญญาณใช้คำสั่ง AudioFormat.*CHANNEL_IN_MONO* การมอดูเลตแบบเข้ารหัสพัลส์ 16 บิต ใช้คำสั่ง AudioFormat.*ENCODING_PCM_16BIT* ที่อัตราการสุ่มชัก 44100 ตัวอย่างต่อวินาที จากตัวแปรชื่อ *sampling* ซึ่งในการอัดเสียงแบบนี้จะมีการสร้างตัวแปรชื่อ buffersize เพื่อเก็บจำนวนค่าตัวอย่างทั้งหมดที่โปรแกรมชุดคำสั่ง AudioRecord.*getMinBufferSize*() สามารถบันทึกได้ จากนั้นจะนำเอาจำนวนค่าตัวอย่างทั้งหมดที่เก็บได้มาเป็นตัวจำนวนอิลิเมนต์ของตัวแปร *buffer* ชนิดอาเรย์ โดยจะใช้ตัวแปร **short** เนื่องจากเป็นตัวแปรที่เก็บค่าได้ 16 บิต จากนั้นนำค่าพารามิเตอร์ต่างๆ ไปใส่ในเมธอด AudioRecord() เพื่อนำค่าไปใส่ในออปเจ็ค audioRecord ของคลาส AudioRecord จากนั้นออปเจ็ค _initTask ของคลาส initTask ซึ่งเป็นคลาสลูกของคลาส AsyncTask โดยเมธอด onCreate() จะมีโปรแกรมชุดคำสั่งดังนี้

```
protected InitTask _initTask;
public AudioRecord audioRecord;
public int buffersize;
public static short[] buffer;
public static final int sampling = 44100;
protected void onCreate(Bundle savedInstanceState) {
                super.onCreate(savedInstanceState);
                setContentView(R.layout.activity_main);
                 lineChart = (ChartView) findViewById(R.id.lineChartView1);
                 lineChart.setChartData(points);
                 buffersize = AudioRecord.getMinBufferSize(sampling,
AudioFormat.CHANNEL_IN_MONO,AudioFormat.ENCODING_PCM_16BIT);
                 buffer = new short[buffersize];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            audioRecord = new AudioRecord(AudioSource.MIC, sampling,
        AudioFormat.CHANNEL_IN_MONO,AudioFormat.ENCODING_PCM_16BIT,
            buffersize);
            _initTask = new InitTask();
            _initTask.execute(this);
}
```

2) เมธอด onResume() จะเป็นเมธอดที่ใช้ตั้งค่าดีเลย์สำหรับการทำงานของแอคทิวิตี้ใน 1 รอบ ซึ่งผู้จัดทำได้ตั้งค่าให้โปรแกรมทำงานทุกๆ 200 มิลลิวินาที โดยมีโปรแกรมชุดคำสั่งดังนี้

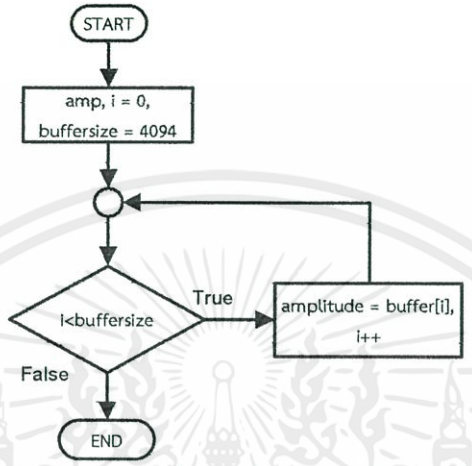
```
private final Handler mHandler = new Handler();
private Runnable mTimer1;
public void onResume() {
    super.onResume();
    mTimer1 = new Runnable() {
        public void run() {
            mHandler.postDelayed(this, 200);
        }
    };
    mHandler.postDelayed(mTimer1, 200);
}
```

3) SuperClass AsyncTask จะถูกนำมาใช้เพื่อทำการประมวลผลในการทำงานส่วนเบื้องหลังพร้อมๆ กับในส่วนของแสดงผลการพลอตกราฟ เพื่อไม่ให้แอปพลิชั่นมีอาการค้างหรืออาการหน่วงที่มากจนเกินไป คลาส AsyncTask จะประกอบด้วย 4 เมธอด นั่นคือ onPreExecute(),doInBackground(Context...params),onProgressUpdate-(Integer... values) และonPostExecute(String result)เมธอด onPreExecute() เป็นเมธอดที่จะทำงานก่อนทำงานในเมธอด doInBackground

เมธอด doInBackground() เป็นเมธอดที่จะให้ทำงานที่เบื้องหลังของ Thread หลัก จึงไม่สามารถใช้คำสั่งเฉพาะบน Thread หลักได้ในนี้ โดยส่วนมากจะใช้สำหรับเตรียมข้อมูลที่โหลดหนักๆให้พร้อมก่อนแล้วไปใช้คำสั่งเฉพาะบน Thread หลัก ที่ฟังก์ชัน onPostExecute แทน ดังนั้นในส่วนนี้จะใช้สำหรับการประมวลผล โดยการเริ่มต้นอัดเสียงโดยใช้เมธอด startRecording() และเก็บจำนวนค่าที่สุ่มชักด้วยเมธอด read() มาเก็บค่าไว้ในตัวแปร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

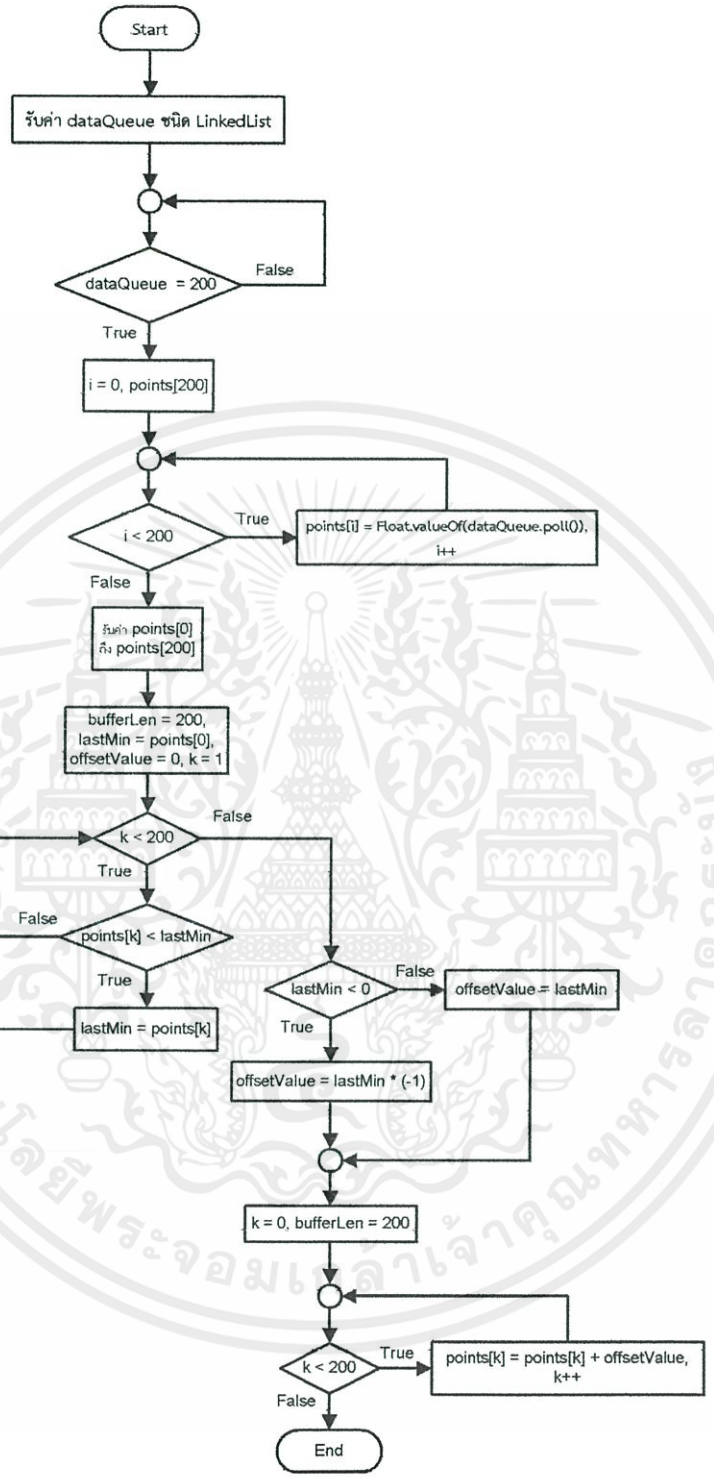
mSamplesRead และสร้างตัวแปร amp เพื่อเก็บค่าแอมปลิจูดของสัญญาณที่แปลงจากสัญญาณอนาลอกไปเป็นสัญญาณดิจิตอล จะมีการเก็บค่าทั้งหมดไว้ในตัวแปรอาเรย์ *buffer* โดยมีแผนผังการวนลูปเก็บค่าดังรูปที่ 3.15 จากนั้นจะส่งค่าแอมปลิจูด amp ไปที่เมธอด onProgressUpdate() ต่อไป ด้วยเมธอด publishProgress()



รูปที่ 3.15 แผนผังโปรแกรมชุดคำสั่งการเก็บค่าแอมปลิจูดของสัญญาณดิจิตอล

เมธอด onProgressUpdate() เป็นเมธอดที่ใช้อัพเดทสถานะการทำงาน ซึ่งจะนำค่าแอมปลิจูดจากเมธอด doInBackground() มาเก็บไว้ในตัวแปร graph2LastYValue จากนั้นจะทำการแปลงจากตัวแปรชนิด **double** ไปเป็นตัวแปร **String** จากนั้นจะเพิ่มเข้าไปใน dataQueue ชนิด LinkedList ซึ่งเปรียบเสมือนกับอาเรย์ที่มีการเรียงลำดับ มีการจัดคิวขบวนชุดข้อมูล จากนั้นเมธอด OffsetValue() จะทำงานเพื่อยกระดับค่าสัญญาณดิจิตอล โดยจะทำการหาค่าดิจิตอลที่ต่ำที่สุดของสัญญาณดิจิตอลเก็บไว้ที่ตัวแปร lastMin จากนั้นนำค่า lastMin ไปยกระดับสัญญาณดิจิตอลทุกค่าด้วยการรวมกับค่าสัญญาณดิจิตอลทั้งหมด ซึ่งแผนผังโปรแกรมชุดคำสั่งจะแสดงดังรูปที่ 3.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 แผนผังโปรแกรมชุดคำสั่งการยกระดับสัญญาณดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

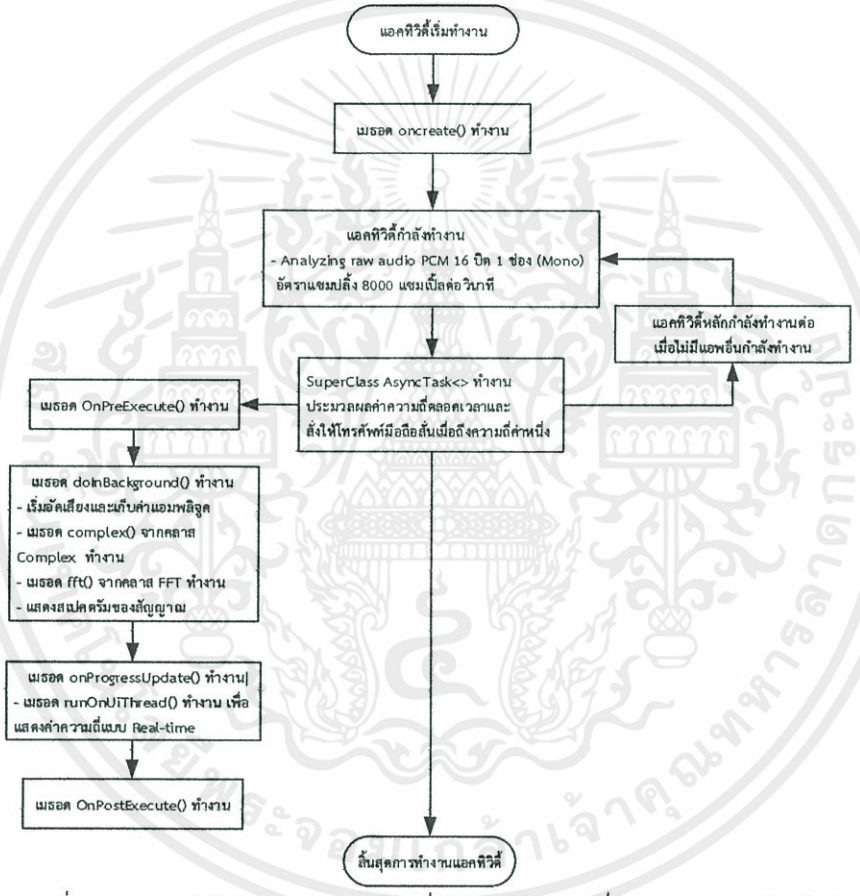
เมธอด onPostExecute() เป็นเมธอดที่ทำงานเมื่อเมธอด doInBackground() ทำงานเสร็จ โปรแกรมชุดคำสั่งทั้งหมดที่ใช้ใน **SuperClass** AsyncTask จะแสดงได้ดังนี้

```
private double graph2LastYValue = 0d;
Queue<String> dataQueue = new LinkedList<String>();
public int mSamplesRead;
public static short[] buffer;
protected class InitTask extends AsyncTask<Context, Integer, String> {
        protected void onPreExecute() {
            super.onPreExecute();
        }
        protected String doInBackground(Context... params) {
            audioRecord.startRecording();
            while (true) {
                try {
                    mSamplesRead = audioRecord.read(buffer, 0,
buffersize);
                    int amp;
                    for (int i = 0; i < buffersize - 1; i++) {
                        amp = (int) buffer[i];
                        publishProgress(amp);
                    }
                } catch (Exception e) {}
            }
        }
        protected void onProgressUpdate(Integer... values) {
            super.onProgressUpdate(values);
            graph2LastYValue = (double) values[0];
            dataQueue.add(String.valueOf(graph2LastYValue));
            OffsetValue(); }
        protected void onPostExecute(String result) {
            super.onPostExecute(result); }
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4.4 การออกแบบการทำงานของโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

ในการออกแบบแอปพลิเคชั่นตรวจจับความถี่ ต้องมีอัลกอริทึมเพื่อคำนวณค่าความถี่ และจะต้องเป็นวิธีที่ง่ายต่อการวิเคราะห์ความถี่ของสัญญาณ ซึ่งในการออกแบบโปรแกรมแอปพลิเคชั่นตรวจจับความถี่นี้ ได้นำอัลกอริทึมการแปลงฟูเรียร์แบบเร็วของคูลลี่ ทูกี้ มาใช้ อัลกอริทึมดังกล่าวนั้นผู้จัดทำได้นำคลาสที่มีผู้พัฒนาไว้อยู่แล้วมาใช้ โดยแผนผังโปรแกรมแอปพลิเคชั่นตรวจจับความถี่สัญญาณแบบเรียลไทม์จะแสดงได้ดังรูปที่ 3.17



รูปที่ 3.17 แผนผังโปรแกรมแอปพลิเคชั่นตรวจจับความถี่สัญญาณแบบเรียลไทม์

เมื่อแอคทิวิตี้หลักเริ่มการทำงานสามารถอธิบายขั้นตอนการทำงานของโปรแกรมชุดคำสั่งได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) เมธอด onCreate() จะถูกตั้งค่าไว้ด้วยออปเจ็คชื่อ lineChart ซึ่งจะใช้คลาส ChartView โดยจะกำหนดให้ออปเจ็ค lineChart ทำงานบนแอปพลิเคชั่นโดยใช้คำสั่ง findViewById(R.id.*lineChartView1*) โดยที่ออปเจ็คการแสดงผลชื่อ *lineChartView1* จะถูกกำหนดค่าใน activity_main.xml เพื่อแสดงรูปสเปคตรัมของสัญญาณบนหน้าจอแอปพลิเคชั่น ซึ่งในเมธอด onCreate() นี้จะมีการตั้งค่าพารามิเตอร์การอัดเสียงแบบ RAW ออดิโอ แบบ 1 ช่องสัญญาณใช้คำสั่ง AudioFormat.*CHANNEL_IN_MONO* ซึ่งใช้การเข้ารหัสสัญญาณของการมอดูเลตแบบรหัสพัลส์ 16 บิต ซึ่งใช้คำสั่ง AudioFormat.*ENCODING_PCM_16BIT* ที่อัตราการสุ่มชัก 8000 ตัวอย่างต่อวินาที ซึ่งในการอัดเสียงแบบนี้จะมีการสร้างตัวแปร buffersize เพื่อเก็บจำนวนตัวอย่างทั้งหมดที่โปรแกรมชุดคำสั่ง AudioRecord.*getMinBufferSize*() สามารถบันทึกได้ จากนั้นจะนำเอาจำนวนตัวอย่างทั้งหมดที่เก็บได้มาเป็นตัวจำนวนอิลิเมนต์ของตัวแปร *buffer* ชนิดอาเรย์ โดยจะใช้ตัวแปร **short** เนื่องจากเป็นตัวแปรที่เก็บค่าได้ 16 บิต จากนั้นนำค่าพารามิเตอร์ต่างๆ ไปใส่ในเมธอด AudioRecord() เพื่อนำค่าไปใส่ในออปเจ็ค audioRecord ของคลาส AudioRecord จากนั้นออปเจ็ค _initTask ของคลาส initTask ซึ่งเป็นคลาสลูกของคลาส AsyncTask ภายในเมธอด onCreate() ซึ่งมีโปรแกรมชุดคำสั่งดังนี้

```
protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        lineChart = (ChartView) findViewById(R.id.lineChartView1);
        lineChart.setChartData(points);
        buffersize = AudioRecord.getMinBufferSize(sampling,
AudioFormat.CHANNEL_IN_MONO,AudioFormat.ENCODING_PCM_16BIT);
        buffer = new short[buffersize];
        audioRecord = new AudioRecord(AudioSource.MIC, sampling,
    AudioFormat.CHANNEL_IN_MONO,AudioFormat.ENCODING_PCM_16BIT,
        buffersize);
        _initTask = new InitTask();
        _initTask.execute(this);
    }
```

2) **SuperClass** AsyncTask จะถูกนำมาใช้เพื่อทำการประมวลผลในการทำงานส่วนเบื้องหลัง เพื่อไม่ให้แอปพลิชั่นมีอาการค้างหรืออาการหน่วงที่มากจนเกินไป คลาส AsyncTask จะประกอบด้วย 4 เมธอด นั่นคือ onPreExecute(), doInBackground(Context... params), onProgressUpdate(Integer... values) และ onPostExecute(String result)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมธอด onPreExecute() เป็นเมธอดที่จะทำงานก่อนการทำงานในเมธอด doInBackground

เมธอด doInBackground() เป็นเมธอดที่จะทำงานที่เบื้องหลังของ Thread หลัก จึงไม่สามารถใช้คำสั่งเฉพาะบน Thread หลักได้ในนี้ โดยส่วนมากจะใช้สำหรับเตรียมข้อมูลที่โหลดหนักๆให้พร้อมก่อนแล้วไปใช้คำสั่งเฉพาะบน Thread หลัก ที่เมธอด onPostExecute แทน ดังนั้นในส่วนนี้จะใช้สำหรับการประมวลผล โดยการเริ่มต้นอัดเสียงโดยใช้เมธอด startRecording() และเก็บจำนวนค่าสุ่มซักด้วยเมธอด read() มาเก็บค่าไว้ในตัวแปรชื่อ mSamplesRead ซึ่งจะมีการเก็บค่าทั้งหมดไว้ในตัวแปรอาเรย์ *buffer* จากนั้นจะนำค่าข้อมูลดังกล่าวที่ได้สุ่มซักไปทำเป็นจำนวนเชิงซ้อนด้วยใช้เมธอด Complex() จากคลาส Complex โดยวนลูปเก็บค่าสัญญาณ 512 ตัวอย่าง ซึ่งจะเก็บข้อมูลเป็นจำนวนเชิงซ้อนในตัวแปรอาร์เรย์ชื่อ fftTempArray โดยสร้างจากคลาส Complex ดังนี้

```
for (int i = 0; i <512; i++) {
fftTempArray[i] = new Complex((double) buffer[i], 0); }
```

จากนั้นจะนำค่าตัวแปรอาร์เรย์ fftTempArray ไปแปลงฟูเรียร์แบบเร็วของคูลลี่ ทูกี้ โดยใช้เมธอด fft() จากคสาส FFT จะกำหนดออปเจคชื่อ dFT จากนั้นจะนำค่าที่แปลงฟูเรียร์แบบเร็วของคูลลี่ ทูกี้เก็บไว้ในตัวแปรอาร์เรย์จำนวนเชิงซ้อนจากคลาส Complex ที่ชื่อ fftArray แล้วทำการวนลูป 256 ครั้ง เพื่อหาค่าแมกนิจูดมาเก็บในตัวแปรอาร์เรย์ชื่อ bufrealA ดังนี้

```
fftArray = dFT.fft(fftTempArray);
for (int i = 0; i < 256; i++) {
bufrealA[i] = (float) Math.sqrt((fftArray[i].re) * (fftArray[i].re) + (fftArray[i].im) *
(fftArray[i].im)); }
```

จากนั้นนำค่าแมกนิจูดแต่ละค่าที่ได้ไปพลอตกราฟแสดงสเปคตรัมของสัญญาณบนหน้าจอแอปพลิเคชั่นด้วยเมธอด onProgressUpdate() โดยใช้เมธอด publishProgress() เพื่อสั่งให้เมธอด setChartData1() ทำงาน จากนั้นค่าจากเมธอด setChartData1() จะถูกส่งไปพลอตสเปคตรัมโดยใช้คลาส ChartView ดังนี้

```
for (int i = 0; i < 256; i++) {
publishProgress((float) bufrealA[i]);
 }
protected void onProgressUpdate(Float... values) {
        super.onProgressUpdate(values);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        graph2LastYValue = (float) values[0];
        dataQueue.add(String.valueOf(graph2LastYValue));
setChartData1(); }
```

3.1.4.5 การออกแบบการทำงานของโปรแกรมแอปพลิเคชั่นตรวจจับค่าความถี่เมื่อมีการนั่งผิดไปจากลักษณะปกติแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ โดยเซนเซอร์แต่ละตัวจะถูกกำหนดช่วงความถี่ให้มีค่าความถี่เมื่อนั่งในลักษณะปกติอยู่ค่าหนึ่ง และค่าความถี่จะเปลี่ยนไปเมื่อนั่งในลักษณะที่ผิดปกติอีกค่าหนึ่ง ซึ่งจะมีการแบ่งช่วงความถี่ของเซนเซอร์แต่ละตัวออกเป็นช่วงๆ เพื่อทำให้โปรแกรมแอปพลิเคชั่นสามารถแยกการทำงานเพื่อประมวลผลค่าความถี่ของเซนเซอร์แต่ละตัวได้

3.1.4.5.1 เงื่อนไขตรวจจับค่าของเซนเซอร์ตัวที่ถูกติดอยู่ในแนวตามขวางของแผ่นหลัง จากจำนวนตัวแปรอาร์เรย์ที่ใช้เก็บค่าแมกนิจูด 256 ค่านั้น จะกำหนดให้จำนวนของค่าแมกนิจูดที่จะนำมาประมวลผลจาก 256 ค่า จะวิเคราะห์เพียงแค่ค่าที่อยู่ในช่วงที่ 115 ถึงค่าที่ 185 ซึ่งก็คือการกำหนดช่วงค่าความถี่ที่จะประมวลผลเซนเซอร์ตัวนี้ให้อยู่ในช่วงความถี่ที่ 1.8 กิโลเฮิรตซ์ ถึง 2.89 กิโลเฮิรตซ์ โดยใช้จะเมธอด arraycopy() คัดลอกค่าแมกนิจูดจากตัวแปรอาร์เรย์ bufrealA ไปเก็บไว้ในตัวแปรอาร์เรย์สำหรับการประมวลผลเซนเซอร์ตัวที่ถูกติดอยู่ในแนวตามขวางของแผ่นหลังชื่อ bufrealB และใช้เมธอด getMax() เพื่อหาค่าแมกนิจูดที่สูงสุด ซึ่งนั่นก็คือการตรวจจับสเปคตรัมของสัญญาณที่อยู่ในช่วงความถี่ที่เซนเซอร์ทำงานไปเก็บไว้ที่ตัวแปร integer ชื่อ maxbufB โดยมีโปรแกรมชุดคำสั่งดังนี้

```
System.arraycopy(bufrealA, 115, bufrealB, 0, 70);
maxbufB = getMax(bufrealB);
```

เนื่องจากตัวแปรอาร์เรย์ bufrealB มาจากการนำค่าแมกนิจูดจาก bufrealA ซึ่งมีค่าอิลิเมนต์เริ่มต้นที่ 0 นั่นก็คือจำนวนอิลิเมนต์ที่นำมาประมวลผลจึงมีเพียงแค่ 70 อิลิเมนต์ และจะกำหนดเงื่อนไขเมื่อค่าความถี่จากวงจรกำเนิดสัญญาณที่มีค่าความถี่เปลี่ยนแปลงไปตามระยะยืดของเซนเซอร์ โดยถ้าค่า maxbufB น้อยกว่าหรือเท่ากับ 32 และเงื่อนไขนี้จะทำงานร่วมกับค่า maxbufC ซึ่งจะเป็นส่วนของการประมวลผลของเซนเซอร์ที่ถูกติดอยู่ในแนวตั้งตามแนวของกระดูกสันหลัง จากนั้นเมื่อเข้าเงื่อนไขดังกล่าว ก็จะสั่งให้โทรศัพท์มือถือสั่นเตือนด้วยเมธอด getSystemService(Context.*VIBRATOR_SERVICE*) และสร้างระยะดีเลย์ในการสั่นด้วยเมธอด vibrate จากคลาส Vibrator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4.5.2 เงื่อนไขตรวจจับค่าของที่เซนเซอร์ถูกติดอยู่ในแนวตั้งตามแนวของกระดูกสันหลัง จากจำนวนตัวแปรอาร์เรย์ที่ใช้เก็บค่าแมกนิจูด 256 ค่านั้น จะกำหนดให้จำนวนของค่าแมกนิจูดที่จะนำมาวิเคราะห์จาก 256 ค่า จะวิเคราะห์เพียงแค่ค่าที่อยู่ในช่วงที่ 186 ถึงค่าที่ 255 ซึ่งก็คือการกำหนดช่วงค่าความถี่ที่จะประมวลผลเซนเซอร์ตัวนี้ให้อยู่ในช่วงความถี่ 2.91 กิโลเฮิรตซ์ ถึง 3.99 กิโลเฮิรตซ์ โดยใช้เมธอด arrraycopy() คัดลอกค่าแมกนิจูดจากตัวแปรอาร์เรย์ bufrealA ไปเก็บไว้ในตัวแปรอาร์เรย์ชื่อ bufrealC ซึ่งไว้สำหรับการประมวลผลของเซนเซอร์ตัวที่ถูกติดอยู่ในแนวตั้งตามแนวของกระดูกสันหลัง และใช้เมธอด getMax() เพื่อหาความแมกนิจูดที่สูงสุด ซึ่งนั่นก็คือการตรวจจับสเปคตรัมของสัญญาณที่อยู่ในช่วงความถี่ที่เซนเซอร์ทำงานไปเก็บไว้ที่ตัวแปร integer ชื่อ maxbufC โดยมีโปรแกรมชุดคำสั่งดังนี้

```
System.arraycopy(bufrealA, 186, bufrealC, 0, 70);
maxbufC = getMax(bufrealC);
```

เนื่องจากตัวแปรอาร์เรย์ bufrealC มาจากการนำค่าแมกนิจูดจาก bufrealA มีค่าอิลิเมนต์เริ่มต้นที่ 0 นั่นก็คือจำนวนอิลิเมนต์ที่นำมาประมวลผลมีเพียงแค่ 70 อิลิเมนต์ และจะกำหนดเงื่อนไขเมื่อค่าความถี่จากวงจรกำเนิดสัญญาณที่ค่าความถี่เปลี่ยนแปลงไปตามระยะยืดของเซนเซอร์ โดยถ้าค่า maxbufC น้อยกว่าหรือเท่ากับ 19 และเงื่อนไขนี้จะทำงานร่วมกับค่า maxbufB ซึ่งจะเป็นส่วนของการประมวลผลของเซนเซอร์ตัวที่ถูกติดอยู่ในแนวตามขวางของแผ่นหลัง จากนั้นเมื่อเข้าเงื่อนไขดังกล่าว ก็จะสั่งให้โทรศัพท์มือถือสั่นเตือนด้วยเมธอด getSystemService(Context.*VIBRATOR_SERVICE*) และสร้างระยะดีเลย์ในการสั่นด้วยเมธอด vibrate จากคลาส Vibrator

3.1.4.5.3 เงื่อนไขตรวจจับค่าของเซนเซอร์ที่ถูกติดอยู่ในแนวตั้งทางด้านหน้าของลำตัว จากจำนวนตัวแปรอาร์เรย์ที่ใช้เก็บค่าแมกนิจูด 256 ค่านั้น จะกำหนดให้จำนวนของค่าแมกนิจูดที่จะนำมาวิเคราะห์จาก 256 ค่า จะวิเคราะห์เพียงแค่ค่าที่ 96 ถึงค่าที่ 114 ซึ่งก็คือการกำหนดช่วงค่าความถี่ที่จะประมวลผลเซนเซอร์ตัวนี้ให้อยู่ในช่วงความถี่ 1.5 กิโลเฮิรตซ์ ถึง 1.78 กิโลเฮิรตซ์ โดยใช้เมธอด arrraycopy() คัดลอกค่าแมกนิจูดจากตัวแปรอาร์เรย์ bufrealA ไปเก็บไว้ในตัวแปรอาร์เรย์ชื่อ bufrealD ซึ่งไว้สำหรับการประมวลผลของเซนเซอร์ตัวที่ถูกติดอยู่ในแนวตั้งตามแนวของกระดูกสันหลัง และใช้เมธอด getMax() เพื่อหาความแมกนิจูดที่สูงสุด ซึ่งนั่นก็คือการตรวจจับสเปคตรัมของสัญญาณที่อยู่ในช่วงความถี่ที่เซนเซอร์ทำงานไปเก็บไว้ที่ตัวแปร integer ชื่อ maxbufD โดยมีโปรแกรมชุดคำสั่งดังนี้

```
System.arraycopy(bufrealA, 96, bufrealD, 0, 18);
maxbufD = getMax(bufrealD);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากตัวแปรอาร์เรย์ bufrealD มาจากการนำค่าแมกนิจูดจากตัวแปรอาร์เรย์ bufrealA ซึ่งจะมีค่าอิลิเมนต์เริ่มต้นที่ 0 นั่นก็คือจำนวนอิลิเมนต์ที่นำมาประมวลผลมีเพียงแค่ 18 อิลิเมนต์ และจะกำหนดเงื่อนไขเมื่อค่าความถี่จากวงจรกำเนิดสัญญาณที่ค่าความถี่เปลี่ยนแปลงไปตามระยะยืดของเซนเซอร์ โดยถ้าค่า maxbufD น้อยกว่าหรือเท่ากับ 6 เมื่อเข้าเงื่อนไขดังกล่าว ก็จะสั่งให้โทรศัพท์มือถือสั่นเตือนด้วย เมธอด getSystemService(Context.*VIBRATOR_SERVICE*) และสร้างระยะดีเลย์ในการสั่นด้วย เมธอด vibrate จากคลาส Vibrator

3.1.4.6 การออกแบบการทำงานของโปรแกรมแอปพลิเคชั่นบันทึกค่าลงในหน่วยความจำไมโครเอสดีการ์ดและแสดงผลแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

3.1.4.6.1 ขั้นแรกจะทำเขียนโปรแกรมแอปพลิเคชั่นเพื่อบันทึกตัวเลขลงในหน่วยความจำไมโครเอสดีการ์ดของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ โดยจะต้องเพิ่มชุดคำสั่งลงไปในไฟล์ที่ชื่อว่า AndroidManifest.xml เสียก่อน ถึงจะทำให้สามารถบันทึกค่าลงในไมโครเอสดีการ์ดได้โดยมีชุดคำสั่งดังนี้

<uses-permission android:name="*android.permission.WRITE_EXTERNAL_STORAGE*"/>

จะกำหนดชื่อไฟล์ที่จะทำการบันทึกให้เป็นเวลามาตรฐาน ซึ่งจะกำหนดชื่อไฟล์ให้ตรงกับวันและเวลาจริงในปัจจุบันของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ ด้วยเมธอด setToNow() และมีการกำหนดตัวแปรชนิด Integer ชื่อ currentTime ซึ่งจะเป็นตัวแปรที่ใช้กำกับเวลาวันเดือนปีปัจจุบัน ซึ่งจะบันทึกชื่อไฟล์เป็นตัวแปร String ที่ชื่อ filename โดยมีโปรแกรมชุดคำสั่งดังนี้

```
public String filename;
public String dottxt = ".txt";
filename = String.valueOf(currentTime.monthDay) +
        "-"+ String.valueOf(currentTime.month+1) +
        "-"+ String.valueOf(currentTime.year) + dottxt;
```

โดยแอปพลิเคชั่นจะบันทึกไฟล์ลงในไมโครเอสดีการ์ดด้วยนามสกุล .txt และกำหนดตัวแปรชนิด Integer ชื่อ counter ซึ่งเป็นตัวแปรที่จะนำไปใช้แสดงค่าจำนวนครั้งของการนั่งผิดลักษณะไปจากปกติ แต่การเก็บค่าลงในไฟล์นามสกุล .txt จะต้องแปลงตัวแปรชนิด Integer ไปเป็นตัวแปรชนิด String เสียก่อน จึงจะสามารถเก็บค่าลงในไฟล์นามสกุล .txt ได้

กำหนดไดเรกทอรี่ของไฟล์ที่จะบันทึกลงในไมโครเอสดีการ์ดด้วยออปเจ็ค file จากคลาส File ซึ่งจะใช้กำหนดการสร้างไฟล์ด้วยคำสั่ง File(file *Directory*, String name) โดยในตำแหน่ง file *Directory* จะกำหนดให้บันทึกลงในไมโครเอสดีการ์ดซึ่งจะต้องใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมธอด *getExternalStorageDirectory()* จากคลาส Environment และในส่วนของ String name จะเป็นส่วนของการกำหนดชื่อไฟล์ โดยจะใช้ชื่อไฟล์ที่กำหนดไว้ในตัวแปร String ที่ชื่อ filename เนื่องจากในการบันทึกลงไมโครเอสดีการ์ดจะต้องใช้ตัวแปรชนิด Integer หรือ Byte อาร์เรย์ เท่านั้น ซึ่งในรูปแบบของการบันทึกไฟล์นั้นจะต้องใช้คลาส FileOutputStream และจะสร้างออปเจ็คขึ้นมาชื่อ fos เพื่อที่จะใช้งานในคลาสดังกล่าวได้ ในส่วนของคำสั่งที่ใช้เขียนไฟล์ลงไปที่ไมโครเอสดีการ์ด จะใช้เมธอด write(byte[] buffer) โดยที่ในตำแหน่ง byte[] buffer ของเมธอด write จะต้องแปลงตัวแปรชนิด String นั่นคือ startTime, stopTime , tPeriod(s) ที่สร้างไว้ แปลงไปเป็นตัวแปรชนิด byte อาร์เรย์ด้วยเมธอด getBytes() ก่อน จึงจะสามารถนำไปใช้งานในเมธอด write() ได้ และใช้เมธอด flush() เพื่อทำให้บันทึกไวขึ้น จากนั้นจะใช้เมธอด close() เพื่อหยุดการบันทึกไฟล์ลงไมโครเอสดีการ์ด และสร้างเงื่อนไขเพื่อตรวจเช็คนั่นก็คือ !file.exists() เนื่องจากอาจจะมีชื่อไฟล์ที่ชื่อซ้ำกันอยู่ในไมโครเอสดีการ์ดอยู่แล้ว เพื่อไม่ให้โปรแกรมแอปพลิเคชั่นเขียนไฟล์ทับไฟล์เดิมที่มีอยู่ก่อนหน้านี้แล้ว ซึ่งจะมีโปรแกรมชุดคำสั่งดังนี้

```
File file = new File(Environment.getExternalStorageDirectory(), filename);
if (!file.exists()) {
byte[] data = new String("\nstartTime\t\tstopTime\t\t\tPeriod(s)\n").getBytes();
        try     {
                FileOutputStream fos = new FileOutputStream(file);
                fos.write(data);
                fos.flush();
                fos.close();
                }
                catch (FileNotFoundException e) {}
                catch (IOException e) {}
}}
```

นอกจากนั้นได้ทำการสร้างเงื่อนไขเมื่อเกิดกรณีที่วันเปลี่ยน ก็จะเขียนชื่อไฟล์ใหม่เป็นค่าวันใหม่ โดยการสร้างตัวแปร Integer เพื่อเก็บค่าวันเก่า ชื่อ dayTemp และสร้างตัวแปร Integer เพื่อเก็บค่าวันใหม่ชื่อ Today โดยจะตรวจเช็คว่า Today != dayTemp ดังโปรแกรมชุดคำสั่งต่อไปนี้

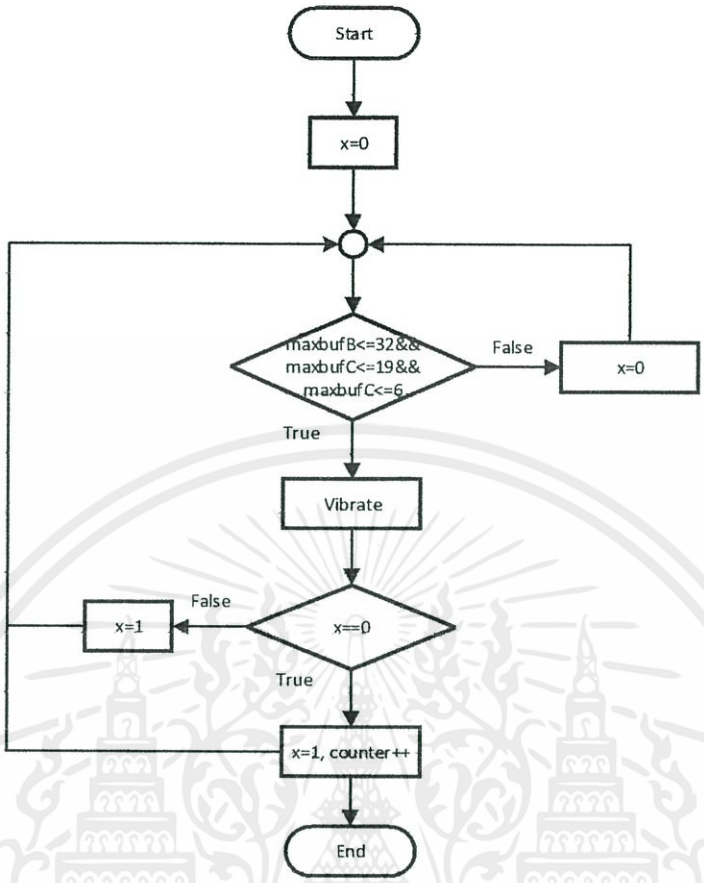
```
if (today != dayTemp) {
try { filename = String.valueOf(currentTime.monthDay) +
        "-" + String.valueOf(currentTime.month) +
        "-" + String.valueOf(currentTime.year) + dottxt;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
File file = new File(Environment.getExternalStorageDirectory(),filename);
byte[] data = new String("\nstartTime\tstopTime\t Period(s)\n").getBytes();
fos = new FileOutputStream(file);
fos.write(data);
fos.flush();
fos.close(); }
                catch (FileNotFoundException e) {}
                catch (IOException e) {} } }
```

3.1.4.6.2 ขั้นที่สองในส่วนของการบันทึกค่าลงในไมโครเอสดีการ์ด ซึ่งในการเขียนโปรแกรมแอปพลิเคชั่น จะเขียนโปรแกรมในส่วนของการบันทึกค่าลงในไมโครเอสดีการ์ดและการแสดงผลค่าตัวเลขแบบเรียลไทม์ ซึ่งในกรณีที่ค่า maxbufB น้อยกว่าหรือเท่ากับ 32 และค่า maxbufC น้อยกว่าหรือเท่ากับ 19 และค่า maxbufD น้อยกว่าหรือเท่ากับ 6 นั่นคือสเปคตรัมของสัญญาณอินพุตที่รับเข้ามาจากช่องสัญญาณไมโครโฟนมีค่าความถี่น้อยกว่า 2.3 กิโลเฮิรตซ์ 3.2 กิโลเฮิรตซ์ และ 1.6 กิโลเฮิรตซ์ ตามลำดับ จะทำให้การทำงานในส่วนของการบันทึกลงไมโครเอสดีการ์ดและการแสดงผลค่าตัวเลขแบบเรียลไทม์ทำงาน เนื่องจากการทำงานของโปรแกรมแอปพลิเคชั่นตรวจจับสเปคตรัมของสัญญาณมีการทำงานตลอดเวลา ทำให้ตัวแปรที่จะนับจำนวนครั้งเมื่อมีสัญญาณอินพุตที่มีความถี่น้อยกว่า 2.3 กิโลเฮิรตซ์ และ 3.2 กิโลเฮิรตซ์ ตามลำดับ มีค่าเพิ่มขึ้นเรื่อยๆอยู่ตลอดเวลาเมื่อมีการเข้าเงื่อนไขดังกล่าว และเพื่อไม่ให้ค่าตัวแปรนี้ผิดพลาด ซึ่งตัวแปรนี้จะต้องเพิ่มขึ้นทีละ 1 เท่านั้น ต่อการเข้าสู่เงื่อนไข จึงต้องออกแบบแผนผังของโปรแกรมชุดคำสั่งสำหรับตรวจสอบค่าความถี่ดังรูปที่ 3.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 แผงผังโปรแกรมชุดคำสั่งของการกำหนดค่าจำนวนครั้งที่นั่งผิดปกติ

จากนั้นจะต้องกำหนดค่าที่จะเขียนลงไปในไฟล์ โดยจะกำหนดระยะเวลาที่นั่งผิดลักษณะรูปทรงไปจากปกติในท่าต่างๆ นั่นก็คือ เริ่ม ณ เวลาใด หยุดการนั่งผิดลักษณะเมื่อใด และเป็นระยะเวลานานเท่าไหร่ การเริ่มต้นการบันทึกในข้างต้น เมื่อเข้าสู่เงื่อนไขค่าที่ maxbufB น้อยกว่าหรือเท่ากับ 32 และค่า maxbufC น้อยกว่าหรือเท่ากับ 19 และค่า maxbufD น้อยกว่าหรือเท่ากับ 6 จะนับว่าเกิดการนั่งที่ผิดรูปทรงไปจากปกติทันที กำหนดตัวแปรเช็คสถานะดังกล่าวเป็นตัวแปร Integer ชื่อ x ให้มีค่าเท่ากับ 0 ซึ่งจะทำให้เข้าเงื่อนไข x เท่ากับ 0 และจะเพิ่มค่า counter ขึ้น 1 ซึ่งก็คือจำนวนครั้งการนั่งผิดลักษณะ และกำหนดค่าเวลาแบบเป็นวินาทีเก็บไว้ในตัวแปร Integer ชื่อ epochStart จากนั้นกำหนดเวลาเริ่มต้นอยู่ในรูปแบบเวลาชั่วโมง, นาที และวินาที ไว้ที่ตัวแปร Interger ชื่อ startHour, startMinute และ startSecond ตามลำดับ และกำหนดให้พ้นจากเงื่อนไขนี้ด้วยการเปลี่ยนค่าตัวแปร x จาก 0 เป็น 1 โดยมีโปรแกรมชุดคำสั่งดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (x == 0) {
counter++;
currentTime.setToNow();
epochStart = currentTime.toMillis(true) / 1000;
startHour = currentTime.hour;
startMinute = currentTime.minute;
startSecond = currentTime.second;
x = 1; } }
```

จากนั้นจะเข้าเงื่อนไข x เท่ากับ 1 ซึ่งจะกำหนดค่าเวลาแบบเป็นวินาทีเก็บไว้ในตัวแปร Integer ชื่อ epochEnd เพื่อนำไปลบกับค่า epochStart ก็จะได้ระยะเวลาการสิ้นสุดการนั่งผิดลักษณะเป็นวินาที จากนั้นกำหนดเวลาสิ้นสุดในรูปแบบเวลาเป็นชั่วโมง, นาที และวินาที ไว้ที่ตัวแปร Interger ชื่อ endHour, endMinute และ endSecond ตามลำดับ และกำหนดให้พ้นจากเงื่อนไขนี้ด้วยเปลี่ยนตัวแปร x จาก 1 เป็น 0 โดยมีโปรแกรมชุดคำสั่งดังนี้

```
else if (x == 1) {
x = 0;
currentTime.setToNow();
epochEnd = currentTime.toMillis(true) / 1000;
long diffTime = epochEnd - epochStart;
try {
String startHourStr, startMinuteStr, startSecondStr, endHourStr, endMinuteStr,
endSecondStr;
endHour = currentTime.hour;
endMinute = currentTime.minute;
endSecond = currentTime.second;
```

เนื่องจากการกำหนดค่าเวลาต่างๆ นั่นคือ เดือน, วัน และวินาที เป็นหลักหน่วย จะทำให้มีปัญหาเรื่องการเขียนค่าบันทึกลงในไฟล์ จึงแก้ไขเรื่องดังกล่าวด้วยการเช็คค่าชั่วโมง ค่านาที ค่าวินาที ทั้งในเวลาที่เริ่มต้นการเข้าเงื่อนไข และในเวลาที่สิ้นสุด ว่าเป็นหลักหน่วยหรือไม่ ถ้าเป็นหลักหน่วยจะทำการเพิ่มตัวแปร String เท่ากับ 0 เพิ่มไปข้างหน้า โดยจะมีโปรแกรมชุดคำสั่งดังนี้

```
if (startHour < 10) { startHourStr = "0" + String.valueOf(startHour); }
else startHourStr = String.valueOf(startHour);
if (startMinute < 10) { startMinuteStr = "0" + String.valueOf(startMinute); }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else startMinuteStr = String.valueOf(startMinute);
if (startSecond < 10) { startSecondStr = "0"+String.valueOf(startSecond); }
else startSecondStr = String.valueOf(startSecond);
if (endHour < 10) { endHourStr = "0" + String.valueOf(endHour); }
else endHourStr = String.valueOf(endHour);
if (endMinute < 10) { endMinuteStr = "0" + String.valueOf(endMinute); }
else endMinuteStr = String.valueOf(endMinute);
if (endSecond < 10) { endSecondStr = "0" + String.valueOf(endSecond); }
```

จากนั้นทำการบันทึกค่าเวลาต่างๆ ลงไปด้วยโปรแกรมชุดคำสั่งเหมือนเดิมดังนี้

```
byte[] data = new String("\n" + startHourStr + ":"+ startMinuteStr + ":" +
startSecondStr + "\t  " + endHourStr + ":" + endMinuteStr + ":" + endSecondStr +
"\t\t\t\t"+ String.valueOf(diffTime)).getBytes();
File file = new File( Environment.getExternalStorageDirectory(),filename);
fos = new FileOutputStream(file, true);
fos.write(data);
fos.flush();
fos.close();}
catch (FileNotFoundException e) {}
catch (IOException e) {} }
```

โดยจะนำเมธอด runOnUiThread() มาใช้ทำงานในส่วนของการแสดงผลแบบเรียลไทม์ โดยจะเพิ่มชุดคำสั่งเข้าไปในคลาส AsyncTask ซึ่งเป็น SuperClass ของคลาส InitTask ที่สร้างขึ้น โดยที่จะกำหนดให้ทำงานอยู่ในส่วนของเมธอด doInBackground(Context... params) ซึ่งในส่วนของการแสดงผลบนแอปพลิเคชั่นจะใช้เมธอด setText() จากคลาส TextView เพื่อแสดงผลแบบเรียลไทม์ โดยจะทำการตั้งค่าเวลาให้เป็นปัจจุบัน ด้วยเมธอด setToNow() ซึ่งจะแสดงเวลาเป็นชั่วโมง, นาที และวินาที บนหน้าจอแอปพลิเคชั่น โดยจะนำค่าเวลาแบบเรียลไทม์มาเก็บไว้ในตัวแปร Integer ชื่อ timeshowHour, timeshowMinute และ timeshowSecond ตามลำดับ เพื่อความสวยงามในการแสดงผลบนแอปพลิเคชั่นจึงได้ทำการเติมเลข 0 ไปข้างหน้าเวลาที่เป็นหลักหน่วย และแสดงจำนวนครั้งที่นั่งลักษณะผิดปกติด้วยค่าของตัวแปร counter รวมถึงแสดงข้อความเตือนผู้สวมใส่บนหน้าจอแอปพลิเคชั่นว่า Your sitting not posture disorders ถ้าค่า maxbufB น้อยกว่าหรือเท่ากับ 32 และค่า maxbufC น้อยกว่าหรือเท่ากับ 19 และค่า maxbufD น้อยกว่าหรือเท่ากับ 6 โดยมีโปรแกรมชุดคำสั่งดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
runOnUiThread(new Runnable() {
public void run() {
currentTime.setToNow();
timeshowHour = currentTime.hour;
timeshowMinute = currentTime.minute;
timeshowSecond = currentTime.second;
String showHourStr, showMinuteStr, showSecondStr;
if (timeshowHour < 10) {
showHourStr = "0" + String.valueOf(timeshowHour); }
else showHourStr = String.valueOf(timeshowHour);
if (timeshowMinute < 10) {
showMinuteStr = "0" + String.valueOf(timeshowMinute); }
else showMinuteStr = String.valueOf(timeshowMinute);
if (timeshowSecond < 10) {
showSecondStr = "0" + String.valueOf(timeshowSecond); }
else showSecondStr = String.valueOf(timeshowSecond);
ShowNumber.setText(showHourStr + ":" + showMinuteStr + ":"showSecondStr);
cout.setText(String.valueOf(counter));
if(maxbufB<=32&&maxbufC<=19)
ShowText.setText("Your sitting posture disorders");
else ShowText.setText("Your sitting not posture disorders"); } );
```

3.2 เครื่องมือที่ใช้ในการทดลอง

ในการทดลองได้มีการใช้อุปกรณ์ต่างๆ ดังนี้

1) อุปกรณ์จ่ายไฟ (Power supply) 1 เครื่อง
2) ออสซิลโลสโคป (Oscilloscope) 1 เครื่อง
3) เครื่องกำเนิดสัญญาณ (Function generator) 1 เครื่อง
4) วงจรกำเนิดสัญญาณ, สายนำสัญญาณ, โทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การจัดเก็บผลการทดลอง

สำหรับปริญญานิพนธ์ฉบับนี้ ได้ออกแบบการจัดเก็บผลการทดลอง ดังนี้

3.3.1 การทดลองวงจรกำเนิดสัญญาณ

3.3.2 การทดลองการทำงานของโปรแกรมชุดคำสั่งบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

3.3.3 การทดสอบอุปกรณ์บนตัวคน และแสดงผลค่าความถี่บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์โดยผ่านทางช่องสัญญาณไมโครโฟนด้วยสายนำสัญญาณเพื่อทำการเตือนผู้สวมใส่ด้วยการสั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

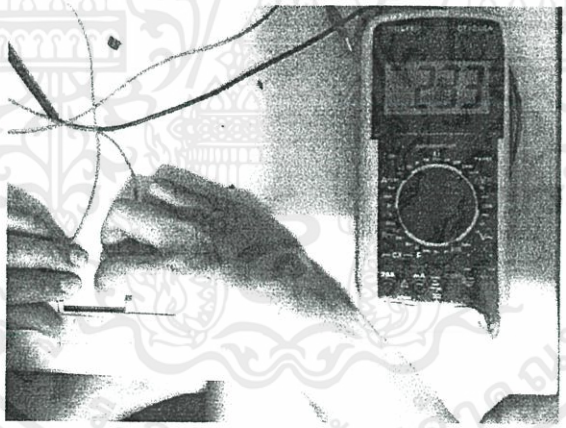
ผลการทดลอง

4.1 ผลการทดลองวงจรกำเนิดสัญญาณ

ผู้จัดปริญญานิพนธ์ทำได้นำเซนเซอร์ระยะยืดมาใช้กับวงจรกำเนิดสัญญาณสี่เหลี่ยม โดยใช้ไอซีเบอร์ 351 ในตำแหน่ง R ในวงจรกำเนิดความถี่ของรูปที่ 3.3 และเพื่อความสะดวกในการทำการทดลอง ผู้จัดทำได้นำเซนเซอร์ระยะยืดมาตัดออกเพียง 1 ส่วน มีความยาวประมาณ 4.5 เซนติเมตร ดังรูปที่ 4.1

รูปที่ 4.1 เซนเซอร์ระยะยืดขนาดความยาว 4.5 เซนติเมตร

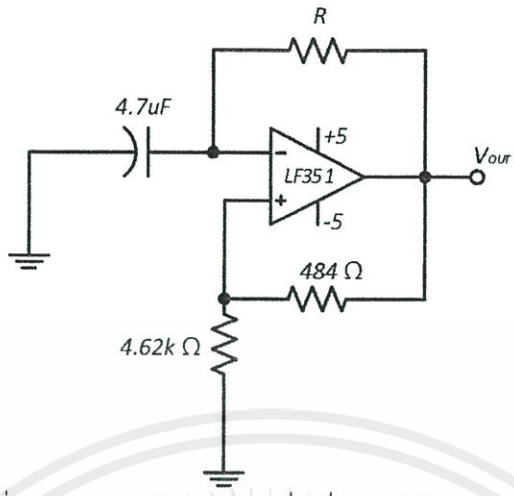
วัดค่าความต้านทานของเซนเซอร์ระยะยืดที่ตัดออกมาได้ 233 โอห์ม ดังรูปที่ 4.2



รูปที่ 4.2 ค่าความต้านทานของเซนเซอร์ระยะยืดขนาดความยาว 4.5 เซนติเมตร

ในการทดลองนี้ ได้ทำการออกแบบและสร้างวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมโดยใช้ไอซีเบอร์ 351 โดยจะกำหนดค่าตัวต้านทานและตัวเก็บประจุดังรูปที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

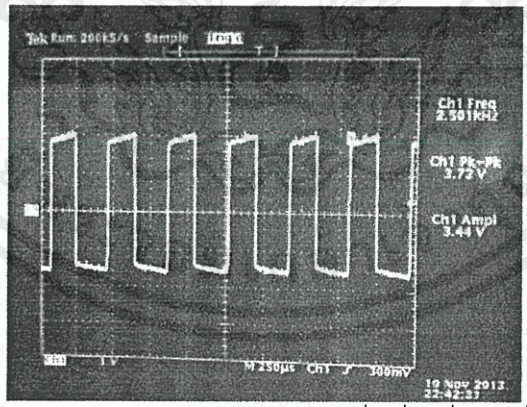


รูปที่ 4.3 วงจรกำเนิดสัญญาณสี่เหลี่ยมโดยใช้ไอซีเบอร์ 351

จากนั้นวัดค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อมีความยาว 4.5 เซนติเมตร ได้ 233.8 โอห์ม

รูปที่ 4.4 ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อไม่มีการยืด

ได้ค่าความถี่ของสัญญาณเอาต์พุตประมาณ 2.5 กิโลเฮิรตซ์ ดังรูปที่ 4.5



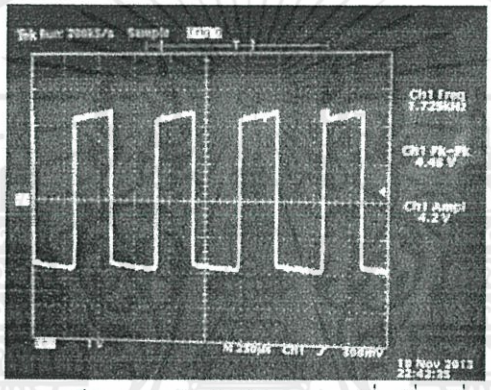
รูปที่ 4.5 สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ความถี่ 2.5 กิโลเฮิรตซ์ เมื่อไม่มีการยืด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเซนเซอร์ระยะยืดถูกยืดออก 0.5 เซนติเมตร วัดค่าความต้านทานได้ 322 โอห์ม ดังรูปที่ 4.6

รูปที่ 4.6 ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 0.5 เซนติเมตร

และได้สัญญาณเอาต์พุตของวงจรกำเนิดสัญญาณรูปสี่เหลี่ยม 1.7 กิโลเฮิรตซ์ ดังรูปที่ 4.7



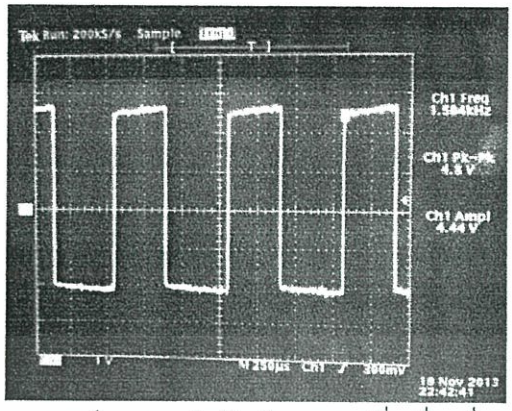
รูปที่ 4.7 สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.7 กิโลเฮิรตซ์ เมื่อยืด 0.5 เซนติเมตร

เมื่อเซนเซอร์ระยะยืดถูกยืดออก 1 เซนติเมตร วัดค่าความต้านทานได้ 377 โอห์ม ดังรูปที่ 4.8

รูปที่ 4.8 ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 1 เซนติเมตร

ได้สัญญาณเอาต์พุตของวงจรกำเนิดสัญญาณสี่เหลี่ยม 1.5 กิโลเฮิรตซ์ ดังรูปที่ 4.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

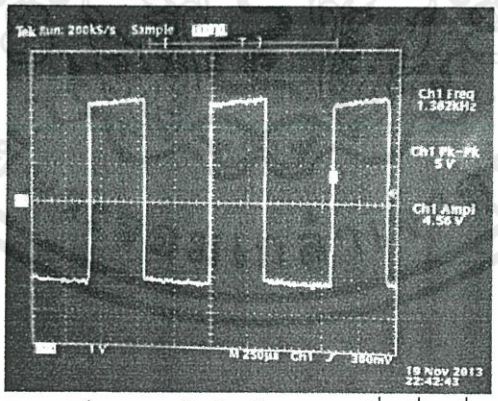


รูปที่ 4.9 สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.5 กิโลเฮิรตซ์ เมื่อยืด 1 เซนติเมตร

เมื่อเซนเซอร์ระยะยืดถูกยืดออก 1.5 เซนติเมตร วัดค่าความต้านทานได้ 406 โอห์ม ดังรูปที่ 4.10

รูปที่ 4.10 ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 1.5 เซนติเมตร

ได้สัญญาณเอาต์พุตของวงจรกำเนิดสัญญาณรูปสี่เหลี่ยม 1.36 กิโลเฮิรตซ์ ดังรูปที่ 4.11



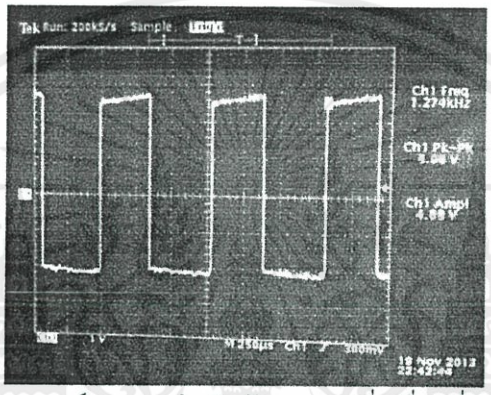
รูปที่ 4.11 สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.36 กิโลเฮิรตซ์ เมื่อยืด 1.5 เซนติเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเซนเซอร์ระยะยืดถูกยืดออก 2 เซนติเมตร วัดค่าความต้านทานได้ 437 โอห์ม ดังรูปที่ 4.12

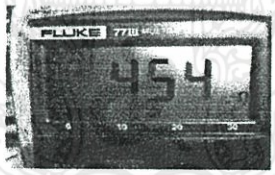
รูปที่ 4.12 ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 2 เซนติเมตร

ได้สัญญาณเอาต์พุตของวงจรกำเนิดสัญญาณรูปสี่เหลี่ยม 1.27 กิโลเฮิรตซ์ ดังรูปที่ 4.13



รูปที่ 4.13 สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.27 กิโลเฮิรตซ์ เมื่อยืด 2 เซนติเมตร

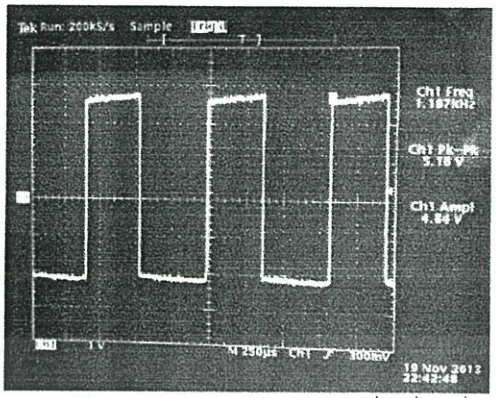
เมื่อเซนเซอร์ระยะยืดถูกยืดออก 2.5 เซนติเมตร วัดค่าความต้านทานได้ 454 โอห์ม ดังรูปที่ 4.14



รูปที่ 4.14 ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 2.5 เซนติเมตร

ได้สัญญาณเอาต์พุตของวงจรกำเนิดสัญญาณรูปสี่เหลี่ยม 1.2 กิโลเฮิรตซ์ ดังรูปที่ 4.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

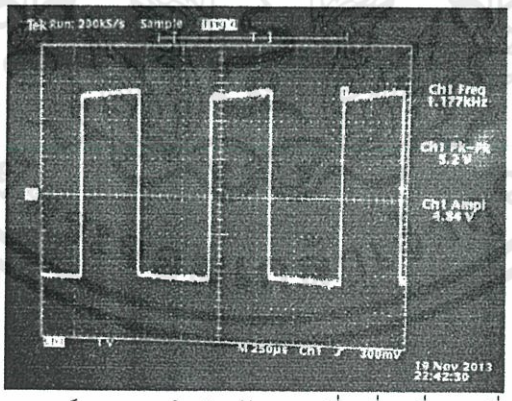


รูปที่ 4.15 สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณรูปสี่เหลี่ยมที่ความถี่ 1.2 กิโลเฮิรตซ์ เมื่อยืด 2.5 เซนติเมตร

และเมื่อเซนเซอร์ระยะยืดถูกยืดออก 3 เซนติเมตร วัดค่าความต้านทานได้ 494 โอห์ม ดังรูปที่ 4.16

รูปที่ 4.16 ค่าความต้านทานของเซนเซอร์ระยะยืดเมื่อยืดไป 3 เซนติเมตร

และได้สัญญาณเอาต์พุตของวงจรกำเนิดสัญญาณรูปสี่เหลี่ยม 1.18 กิโลเฮิรตซ์ ดังรูปที่ 4.17



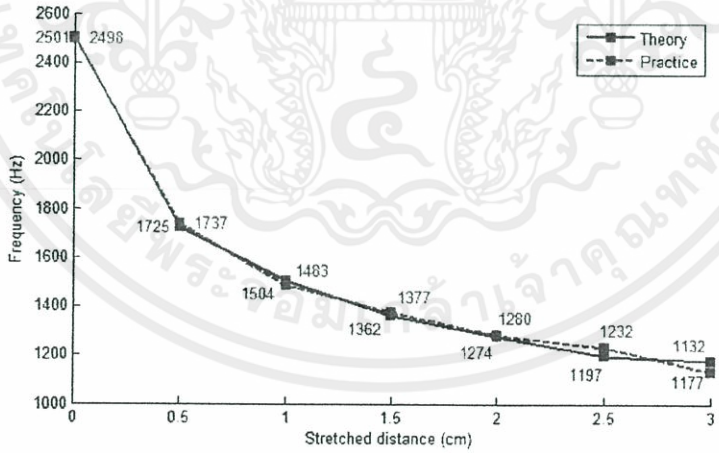
รูปที่ 4.17 สัญญาณเอาต์พุตวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ความถี่ 1.17 กิโลเฮิรตซ์ เมื่อยืด 3 เซนติเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนำค่าความถี่จากวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ได้จากการทดลองมาเปรียบเทียบกับค่าความถี่ที่คำนวณจากสมการที่ 2.6 จะได้ดังตารางที่ 4.1 จากนั้นนำค่าในตารางดังกล่าวไปพลอตกราฟเพื่อเปรียบเทียบค่าที่มาจากการคำนวณค่าความถี่ทางทฤษฎีกับค่าความถี่ที่มาจากการปฏิบัติดังรูปที่ 4.18

ตารางที่ 4.1 เปรียบเทียบค่าความถี่ในทางทฤษฎีและในทางปฏิบัติที่ระยะยืดต่างๆ

ระยะยืด (เซนติเมตร)	ค่าความต้านทานของเซนเซอร์ (โอห์ม)	ความถี่ทฤษฎี (กิโลเฮิรตซ์)	ความถี่ปฏิบัติ (กิโลเฮิรตซ์)
0	223.8	2.498	2.501
0.5	322	1.737	1.725
1.0	377	1.483	1.504
1.5	406	1.377	1.362
2.0	437	1.280	1.274
2.5	454	1.232	1.197
3.0	494	1.132	1.177

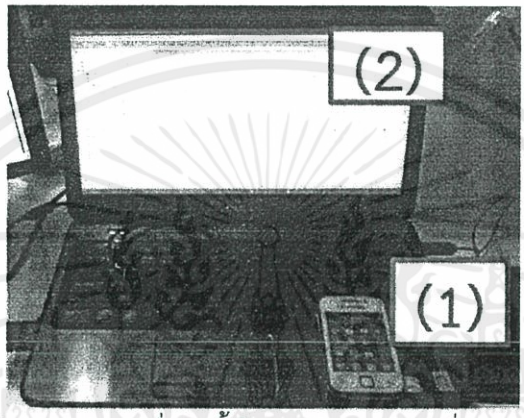


รูปที่ 4.18 กราฟเปรียบเทียบค่าความถี่จากการคำนวณทางทฤษฎีกับค่าความถี่ที่มาจากการปฏิบัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลการทดลองการทำงานของโปรแกรมชุดคำสั่งบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

ในการติดตั้งโปรแกรมแอปพลิเคชั่นของปริญญานิพนธ์ตามบล็อกไดอะแกรมการทำงานของเครื่องช่วยเตือนจัดบุคลิกท่าทางของหลัง จะต้องจัดเตรียมอุปกรณ์ในการทดลองดังรูปที่ 4.19 โดยอุปกรณ์ประกอบไปด้วยโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ (1) และโปรแกรม Eclipse (2)

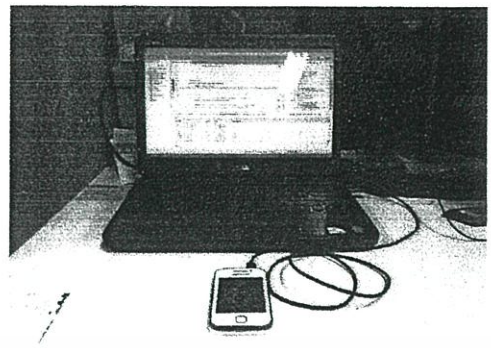


รูปที่ 4.19 อุปกรณ์ที่ใช้ในการทดลองเพื่อติดตั้งโปรแกรมแอปพลิเคชั่นระบบปฏิบัติการแอนดรอยด์

4.2.1 การเชื่อมต่อโทรศัพท์มือถือกับคอมพิวเตอร์ที่ใช้ในการเขียนโปรแกรมชุดคำสั่ง

ในการจัดเก็บผลการทดลองการแสดงผลบนหน้าจอแอปพลิเคชั่น จำเป็นที่จะต้องทำการติดตั้งโปรแกรมชุดคำสั่งที่ได้ทำการออกแบบไว้ตามบล็อกไดอะแกรมการทำงานของโปรแกรมชุดคำสั่งของเครื่องจัดบุคลิกท่าทางของหลังดังรูปที่ 3.1 สำหรับการติดตั้งแอปพลิเคชั่นสำหรับการบันทึกผลการทดลอง สามารถทำได้โดยการเชื่อมต่อโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์กับคอมพิวเตอร์ผ่านสายยูเอสบี (USB : Universal Serial Bus) ดังรูปที่ 4.20

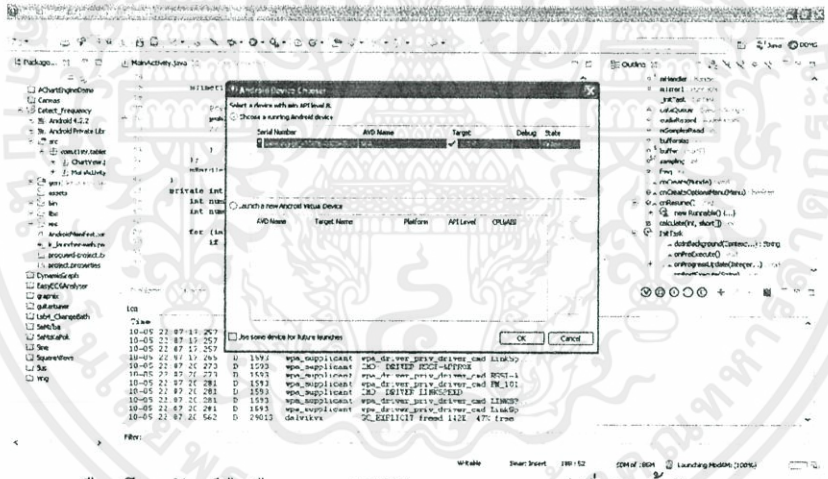
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 การเชื่อมต่อโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์กับคอมพิวเตอร์

4.2.2 การติดตั้งแอปพลิเคชั่นสำหรับบันทึกผลการทดลอง

โปรแกรม Eclipse เป็นโปรแกรมที่ใช้ในการพัฒนาโปรแกรมชุดคำสั่งของแอปพลิเคชั่น เมื่อทำการเชื่อมต่ออุปกรณ์โทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์และคอมพิวเตอร์ที่มีโปรแกรมที่ใช้พัฒนาชุดคำสั่งดังรูปที่ 4.20 เมื่อทำการสั่งรันโปรแกรมจะมีหน้าต่างปรากฏขึ้นมาเพื่อให้เลือกอุปกรณ์ปลายทางเป็นโทรศัพท์มือถือที่ได้ทำการเชื่อมต่อไว้ดังรูปที่ 4.21

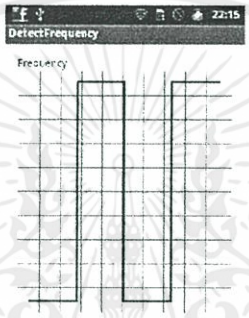


รูปที่ 4.21 การเลือกโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์เพื่อติดตั้งโปรแกรมแอปพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 ผลการทดลองโปรแกรมแอปพลิเคชั่นพลอตกราฟสัญญาณสี่เหลี่ยมบนแอปพลิเคชั่นโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

จากการออกแบบแผนผังโปรแกรมชุดคำสั่งในส่วนของการพลอตกราฟบนแอปพลิเคชั่นดังรูปที่ 3.8 และแผนผังของโปรแกรมชุดคำสั่งการพลอตกราฟสัญญาณสี่เหลี่ยมดังรูปที่ 3.12 ก่อนที่จะทำการรันโปรแกรมแอปพลิเคชั่นตามแผนผังของโปรแกรมชุดคำสั่งทั้ง 2 ส่วน จะต้องเชื่อมต่อสายยูเอสบีดังรูปที่ 4.20 เสียก่อน หลังจากนั้นทำการรันโปรแกรมเพื่อติดตั้งโปรแกรมแอปพลิเคชั่นพลอตกราฟสัญญาณสี่เหลี่ยมบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ ซึ่งจะได้ผลการรันโปรแกรมแอปพลิเคชั่นแสดงได้ดังรูปที่ 4.22

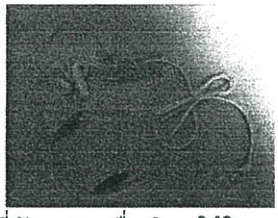


รูปที่ 4.22 แสดงผลการพลอตกราฟสี่เหลี่ยมบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

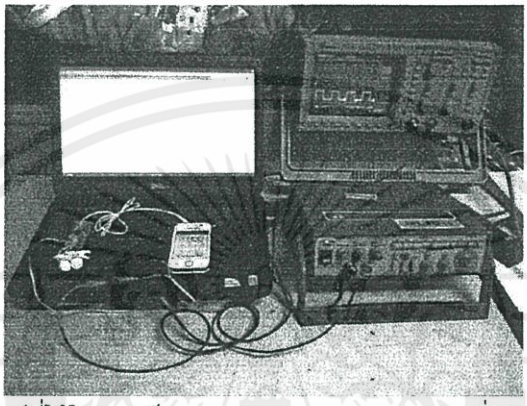
4.2.4 ผลการทดลองโปรแกรมแอปพลิเคชั่นรับสัญญาณเสียงแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

จากการออกแบบแผนผังโปรแกรมชุดคำสั่งในส่วนของการพลอตกราฟบนแอปพลิเคชั่นดังรูปที่ 3.8 และแผนผังของโปรแกรมแอปพลิเคชั่นรับสัญญาณเสียงแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ดังรูปที่ 3.14 ก่อนที่จะทำการรันโปรแกรมแอปพลิเคชั่นตามแผนผังของโปรแกรมชุดคำสั่งทั้ง 2 ส่วน จะต้องเชื่อมต่อสายยูเอสบีดังรูปที่ดังรูปที่ 4.20 และทำการป้อนสัญญาณอินพุตสี่เหลี่ยมจากเครื่องกำเนิดสัญญาณที่ค่าแรงดันประมาณ 100 มิลลิโวลต์ พีคทูพีค ยกระดับที่ -1 โวลต์ โดยมีความถี่เท่ากับ 1.5 กิโลเฮิรตซ์ ผ่านสายหูฟังสมอลทอร์คที่ถูกดัดแปลงเพื่อนำมาใช้ในการป้อนสัญญาณอินพุตผ่านทางช่องสัญญาณไมโครโฟนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ดังรูปที่ 4.23 ซึ่งอุปกรณ์ในการเก็บผลการทดลองทั้งหมดจะแสดงได้ดังรูปที่ 4.24 จากนั้นทำการรันโปรแกรมเพื่อติดตั้งโปรแกรมแอปพลิเคชั่นรับสัญญาณเสียงแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ ซึ่งจะได้ผลการรันโปรแกรมแอปพลิเคชั่นแสดงได้ดังรูปที่ 4.25

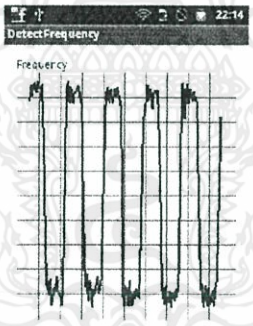
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 หูฟังสมอลทอร์คที่ดัดแปลงเพื่อนำมาใช้ในการป้อนสัญญาณอินพุตผ่านทางช่องสัญญาณไมโครโฟน



รูปที่ 4.24 อุปกรณ์ที่ใช้ในการเก็บผลการทดลองของแอปพลิเคชั่นรับสัญญาณเสียงแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์



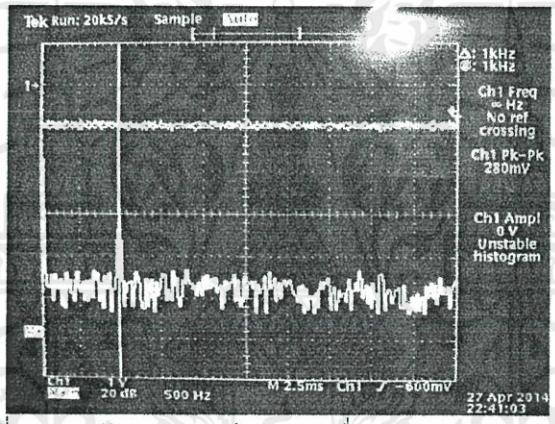
รูปที่ 4.25 สัญญาณสี่เหลี่ยมอินพุตที่แสดงบนโปรแกรมแอปพลิเคชั่นแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.5 ผลการทดลองโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

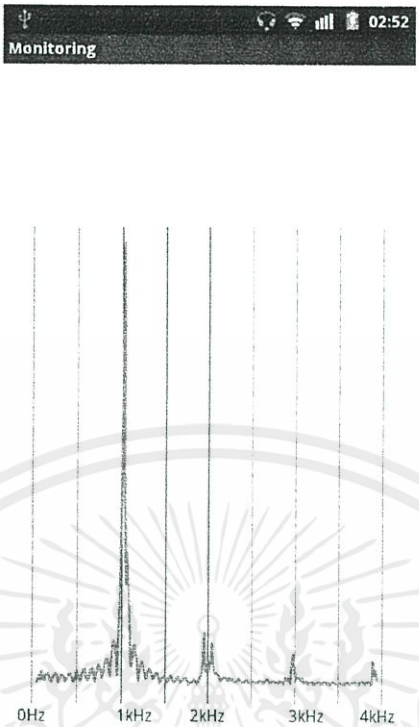
จากการออกแบบแผนผังโปรแกรมชุดคำสั่งแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณแบบเรียลไทม์ดังรูปที่ 3.17 ก่อนที่จะทำการรันโปรแกรมแอปพลิเคชั่นตามแผนผังของโปรแกรมชุดคำสั่งนั้น จะต้องเชื่อมต่อสายยูเอสบีดังรูปที่ 4.20 และป้อนสัญญาณอินพุตไซน์จากเครื่องกำเนิดสัญญาณโดยมีขนาดแอมปลิจูดของสัญญาณประมาณ 100 มิลลิโวลต์ พีคทูพีค ที่ถูกยกระดับสัญญาณ -1 โวลต์ ที่ความถี่ 1 กิโลเฮิรตซ์ 2 กิโลเฮิรตซ์ และ 3 กิโลเฮิรตซ์ ตามลำดับ โดยจะป้อนสัญญาณอินพุตผ่านสายหูฟังสมอลทอร์คผ่านเข้าไปในช่องสัญญาณไมโครโฟนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ จากนั้นทำการรันโปรแกรมเพื่อติดตั้งโปรแกรมแอปพลิเคชั่นรับสัญญาณเสียงแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

เมื่อป้อนสัญญาณอินพุตที่มีความถี่เท่ากับ 1 กิโลเฮิรตซ์ ดังรูปที่ 4.26 ซึ่งจะได้ผลการรันโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณดังรูปที่ 4.27



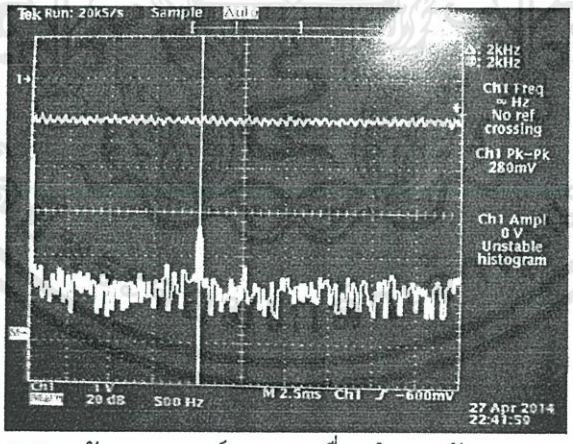
รูปที่ 4.26 สัญญาณเอาต์พุตจากเครื่องกำหนดสัญญาณความถี่ 1 กิโลเฮิรตซ์
ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตไซน์ที่ความถี่ 1 กิโลเฮิรตซ์
ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตไซน์ที่ความถี่ 1 กิโลเฮิรตซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



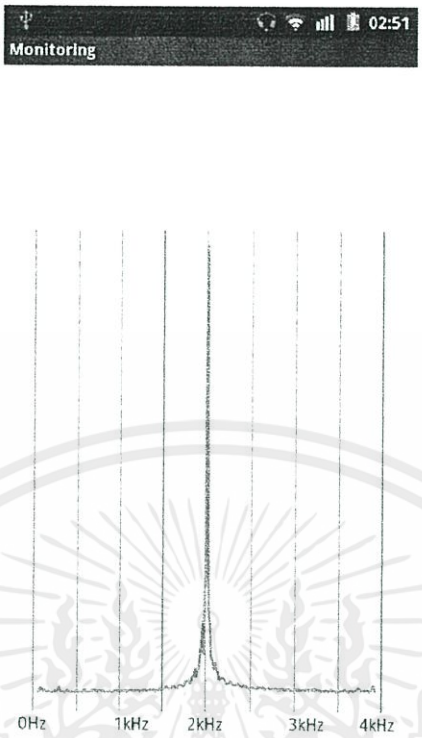
รูปที่ 4.27 การแสดงผลของโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ที่ความถี่ 1 กิโลเฮิรตซ์

เมื่อป้อนสัญญาณอินพุตที่มีความถี่เท่ากับ 2 กิโลเฮิรตซ์ ดังรูปที่ 4.28 ซึ่งจะได้ผลการรันโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณดังรูปที่ 4.29



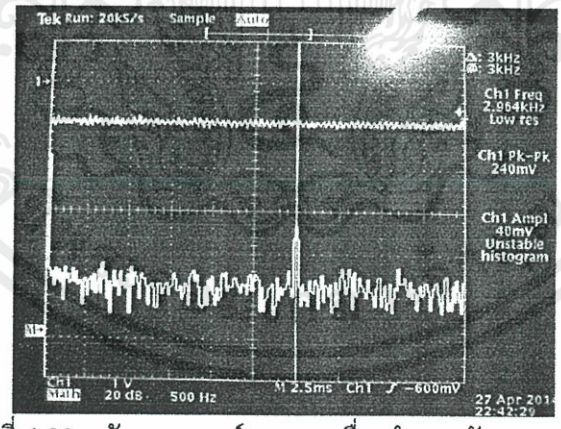
รูปที่ 4.28 สัญญาณเอาต์พุตจากเครื่องกำหนดสัญญาณความถี่ 2 กิโลเฮิรตซ์
ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตไซน์ที่ความถี่ 2 กิโลเฮิรตซ์
ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตไซน์ที่ความถี่ 2 กิโลเฮิรตซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



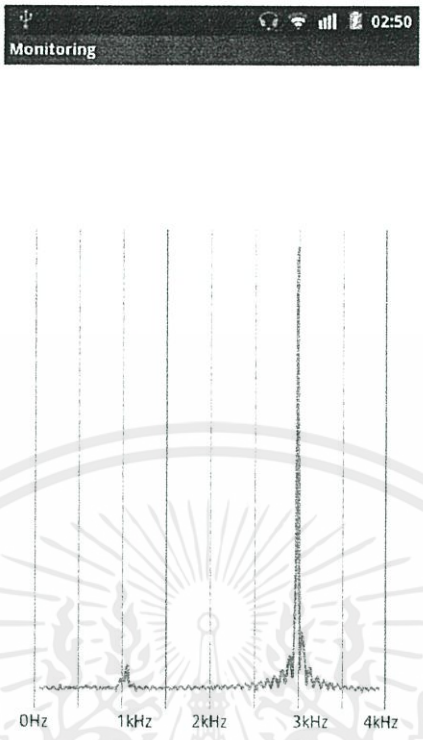
รูปที่ 4.29 การแสดงผลของโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ที่ความถี่ 2 กิโลเฮิรตซ์

เมื่อป้อนสัญญาณอินพุตที่มีความถี่เท่ากับ 3 กิโลเฮิรตซ์ ดังรูปที่ 4.30 ซึ่งจะได้ผลการรันโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณดังรูปที่ 4.31



รูปที่ 4.30 สัญญาณเอาต์พุตจากเครื่องกำหนดสัญญาณความถี่ 3 กิโลเฮิรตซ์
ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตไซน์ที่ความถี่ 3 กิโลเฮิรตซ์
ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตไซน์ที่ความถี่ 3 กิโลเฮิรตซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.31 การแสดงผลของโปรแกรมแอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ที่ความถี่ 3 กิโลเฮิรตซ์

4.3 ทดสอบอุปกรณ์บนตัวคน และแสดงผลค่าความถี่บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์โดยผ่านทางช่องสัญญาณไมโครโฟนด้วยสายนำสัญญาณเพื่อทำการเตือนผู้สวมใส่ด้วยการสั่น

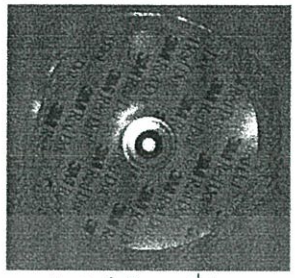
4.3.1 อุปกรณ์ที่นำมาใช้ในการทดสอบ

ทำการดัดแปลงกระดุมแป๊กตัวเมียบัดกรีติดเข้ากับหางปลารูปที่ 4.32 เพื่อนำมายึดกับเซนเซอร์ระยะยืด ซึ่งจะทำให้เซนเซอร์ระยะยืดสามารถยืดหดได้ตามสรีระของแต่ละบุคคล

รูปที่ 4.32 กระดุมแป๊กตัวเมียที่ถูกดัดแปลงให้ถูกเชื่อมติดกับหางปลาเพื่อยึดเซนเซอร์

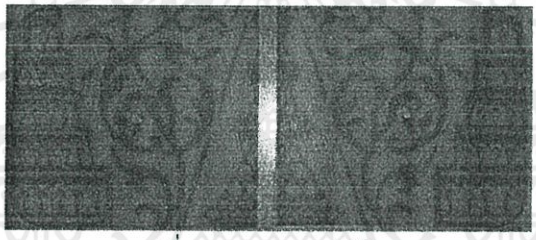
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ที่จะนำมาทำฐานยึดเซนเซอร์ระยะยืดไว้กับแผ่นหลังจะใช้แผ่นอิเล็คโทรดที่มีกระดุมแป๊กตัวผู้ดังรูปที่ 4.33



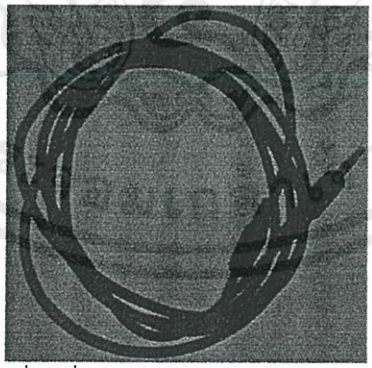
รูปที่ 4.33 แผ่นอิเล็คโทรดที่มีกระดุมแป๊กตัวผู้

อุปกรณ์ที่จะนำมาใช้ยึดฐานไว้สำหรับติดกับกระดุมแป๊กตัวผู้ โดยจะนำมาใช้ติดที่ร่างกาย อุปกรณ์ดังกล่าวมีลักษณะคล้ายๆ แผ่นสุญญากาศทำให้สามารถยึดแผ่นอิเล็คโทรดที่มีกระดุมแป๊กตัวผู้ไว้บนผิวหนังได้อย่างแน่นและมั่นคงพอที่จะทำให้เซนเซอร์ระยะยืดทำงานได้ คือ แผ่นเจลซิลิโคน(Silicone Gel Sheet) แสดงได้ดังรูปที่ 4.34



รูปที่ 4.34 แผ่นเจลซิลิโคน

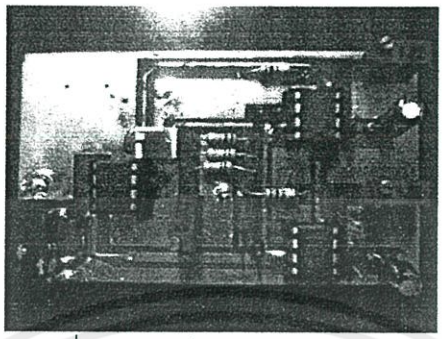
โดยจะใช้สายนำสัญญาณที่สามารถส่งสัญญาณผ่านช่องสัญญาณไมโครโฟนไปยังโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ได้ดังรูปที่ 4.35



รูปที่ 4.35 สายนำสัญญาณที่ใช้เชื่อมต่อกับโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ผ่านทางช่องสัญญาณไมโครโฟน

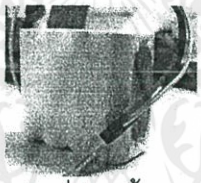
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.36 คือบอร์ดวงจรรวมของระบบซึ่งประกอบไปด้วย วงจรกำเนิดสัญญาณรูปสี่เหลี่ยม 2 วงจรและ วงจรขยายสัญญาณผลรวม 1 วงจร



รูปที่ 4.36 บอร์ดวงจรรวมของระบบ

และได้ใช้แหล่งจ่ายไฟที่มีไฟเลี้ยง ±5 โวลต์ และกราวด์ ดังรูปที่ 4.37



รูปที่ 4.37 แหล่งจ่ายไฟที่มีไฟเลี้ยง ±5 โวลต์ และกราวด์

4.3.2 วิธีการติดเซนเซอร์บนร่างกายผู้สวมใส่

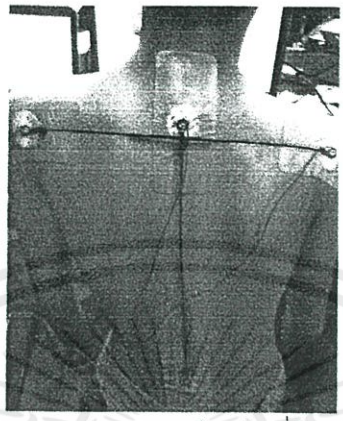
โดยจะทำการติดแผ่นอิเล็คโทรดที่มีกระดุมแป๊กตัวผู้เพื่อเป็นฐานในการยึดเซนเซอร์ระยะยืดบนแผ่นหลัง แล้วใช้แผ่นเจลซิลิโคนแปะทับฐานกระดุมแป๊กตัวผู้ให้แน่นดังรูปที่ 4.38 ก็จะทำให้ยึดเซนเซอร์ระยะยืดให้สามารถยืดหดได้ตามลักษณะที่เคลื่อนไหว

รูปที่ 4.38 ลักษณะการติดฐานด้วยแผ่นเจลซิลิโคนเพื่อนำมาใช้ยึดเซนเซอร์ระยะยืดไว้บนแผ่นหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 ลักษณะการติดเซนเซอร์ 2 ตัว ที่ด้านหลังของร่างกายผู้สวมใส่

ที่ด้านหลังของร่างกายผู้สวมใส่ ได้มีการวางการติดเซนเซอร์ 2 ตำแหน่ง ดังรูปที่ 4.39



รูปที่ 4.39 ลักษณะการติดเซนเซอร์ 2 ตัว ที่ด้านหลังของร่างกาย

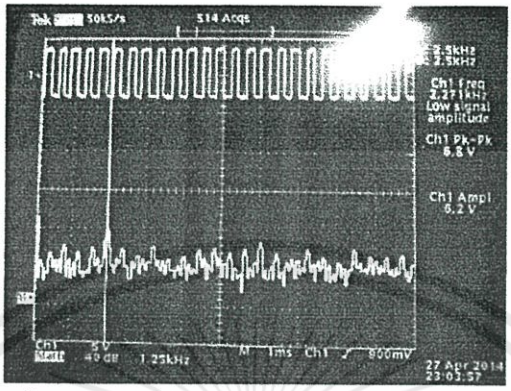
4.3.4 ผลการทดลองเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง เมื่อเซนเซอร์ถูกติดไว้ที่ด้านหลังของร่างกายผู้สวมใส่

4.3.4.1 ผลการทดลองวัดค่าเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตามขวางของแผ่นหลังเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง ค่าความต้านของเซนเซอร์ระยะยืดจะมีค่าเท่ากับ 1.85 กิโลโอห์ม ดังรูปที่ 4.40

รูปที่ 4.40 เมื่อผู้สวมใส่นั่งในลักษณะหลังตรงค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 1.85 กิโลโอห์ม

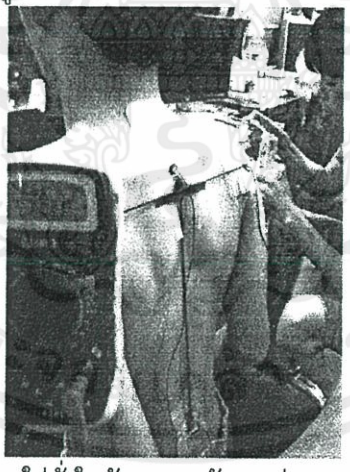
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4.2 ผลการทดลองของวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ความถี่เปลี่ยนแปลงไปตามระยะยืดของเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตามขวางของแผ่นหลังเมื่อผู้สวมใส่นั่งในลักษณะหลังตรงค่าความถี่ของวงจรกำเนิดสัญญาณจึงมีค่าเท่ากับ 2.5 กิโลเฮิรตซ์ ดังรูปที่ 4.41



รูปที่ 4.41 สัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณของเซนเซอร์ที่ถูกติดอยู่ในแนวตามขวางของแผ่นหลังเมื่อผู้สวมใส่นั่งลักษณะหลังตรง
ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณสี่เหลี่ยม
ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณสี่เหลี่ยม

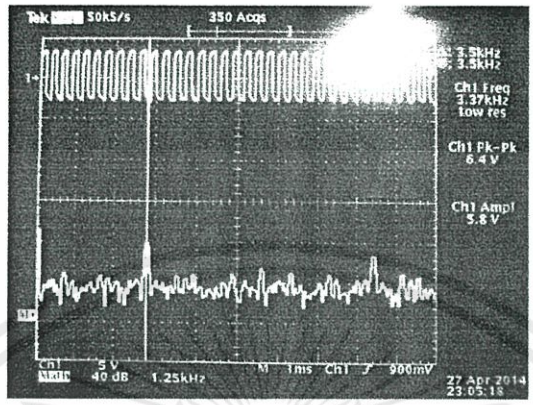
4.3.4.3 ผลการทดลองวัดค่าเซนเซอร์ระยะยืดที่เซนเซอร์ถูกติดอยู่ในแนวตั้งตามแนวกระดูกสันหลังเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง ค่าความต้านของเซนเซอร์ระยะยืดจะมีค่าเท่ากับ 962 โอห์ม ดังรูปที่ 4.42



รูปที่ 4.42 เมื่อผู้สวมใส่นั่งในลักษณะหลังตรงค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 962 โอห์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4.4 ผลการทดลองของวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ความถี่เปลี่ยนแปลงไปตามการยืดของเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตั้งตามแนวกระดูกสันหลังเมื่อผู้สวมใส่นั่งในลักษณะหลังตรงค่าความถี่ของวงจรกำเนิดสัญญาณมีค่าเท่ากับ 3.5 กิโลเฮิรตซ์ ดังรูปที่ 4.43

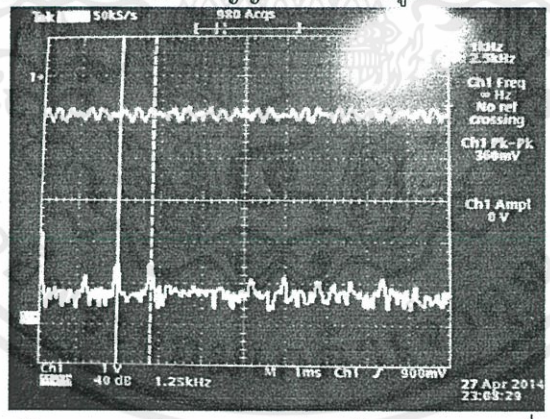


รูปที่ 4.43 สัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณของเซนเซอร์ที่ถูกติดอยู่ในแนวตั้งตามแนวกระดูกสันหลังเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง

ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณสี่เหลี่ยม

ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณสี่เหลี่ยม

4.3.4.5 ผลการทดลองของวงจรขยายสัญญาณผลรวม เมื่อใช้สัญญาณอินพุตที่ได้จากการทดลองที่ 4.3.4.2 และ 4.3.4.3 โดยจะทำการยกระดับสัญญาณด้วยไฟตรง 1 โวลต์ ได้เอาต์พุตสเปคตรัมของวงจรขยายสัญญาณผลรวมดังรูปที่ 4.44



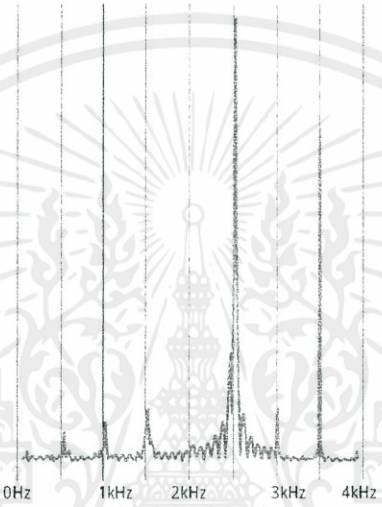
รูปที่ 4.44 สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง

ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวม

ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวม

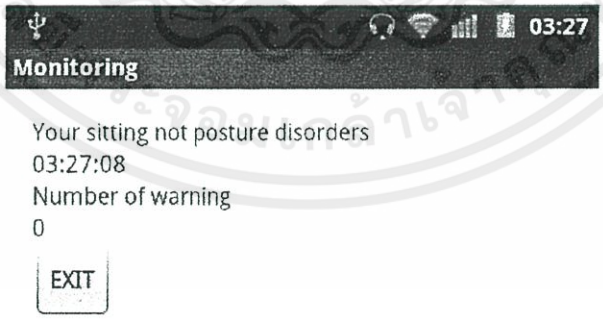
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4.6 ผลการทดลองส่งสัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมของผลการทดลองที่ 4.3.4.5 เข้าทางช่องสัญญาณไมโครโฟนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ เพื่อนำไปพลอตสเปคตรัมและประมวลผลเพื่อตรวจจับการนั่งในลักษณะหลังตรงดังรูปที่ 4.45



รูปที่ 4.45 แอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง

4.3.4.7 ผลการทดลองการแสดงผลบนหน้าจอแอปพลิเคชั่นตรวจจับค่าแมกนิจูดเมื่อมีการนั่งผิดไปจากลักษณะปกติแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ เมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะหลังตรง แอปพลิเคชั่นจะแสดงข้อความว่า “Your sitting not posture disorders” ดังรูปที่ 4.46

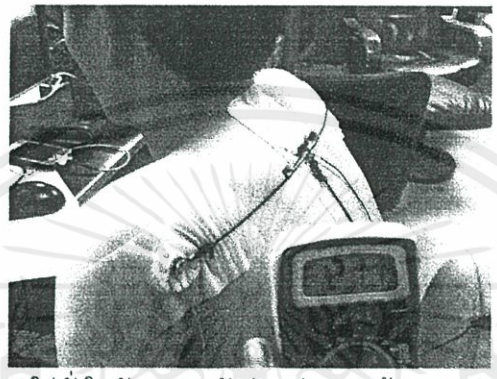


รูปที่ 4.46 หน้าจอแอปพลิเคชั่นเมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะหลังตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.5 ผลการทดลองเมื่อผู้สวมใส่นั่งในลักษณะหลังค่อม เมื่อเซนเซอร์ถูกติดไว้ที่ด้านหลังของร่างกายผู้สวมใส่

4.3.5.1 ผลการทดลองวัดค่าเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตามขวางของแผ่นหลังเมื่อผู้สวมใส่นั่งในลักษณะหลังค่อม ค่าความต้านของเซนเซอร์ระยะยืดจะมีค่าเท่ากับ 2.11 กิโลโอห์ม ดังรูปที่ 4.47



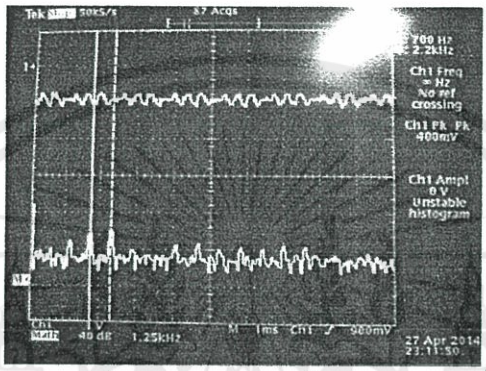
รูปที่ 4.47 เมื่อผู้สวมใส่นั่งในลักษณะหลังค่อมค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 2.11 กิโลโอห์ม

4.3.5.2 ผลการทดลองวัดค่าเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตั้งตามแนวกระดูกสันหลังเมื่อผู้สวมใส่นั่งในลักษณะหลังค่อม ค่าความต้านของเซนเซอร์ระยะยืดจะมีค่าเท่ากับ 1.164 กิโลโอห์ม ดังรูปที่ 4.48

รูปที่ 4.48 เมื่อผู้สวมใส่นั่งในลักษณะหลังค่อมค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 1.164 กิโลโอห์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.5.3 ผลการทดลองของวงจรขยายสัญญาณผลรวมของวงจรกำเนิดสัญญาณทั้ง 2 วงจร จากการทดลองที่ 4.4.3.2 และ 4.4.3.3 เมื่อผู้สวมใส่เปลี่ยนท่านั่งเป็นท่านั่งในลักษณะหลังค่อม โดยจะทำการยกระดับสัญญาณด้วยไฟตรง 1 โวลต์ ได้สัญญาณเอาต์พุตและสเปคตรัมของสัญญาณเอาต์พุตของวงจรขยายสัญญาณผลรวมที่ความถี่ 2.2 กิโลเฮิรตซ์ ซึ่งเป็นสเปคตรัมของสัญญาณในส่วนของวงจรกำเนิดสัญญาณที่ใช้เซนเซอร์ตัวที่ติดอยู่ในแนวตามขวางของแผ่นหลัง และที่ความถี่ 2.7 กิโลเฮิรตซ์ เป็นสเปคตรัมของสัญญาณในส่วนของวงจรกำเนิดสัญญาณที่ใช้เซนเซอร์ตัวที่ติดอยู่ในแนวตั้งตามแนวกระดูกสันหลังดังรูปที่ 4.49



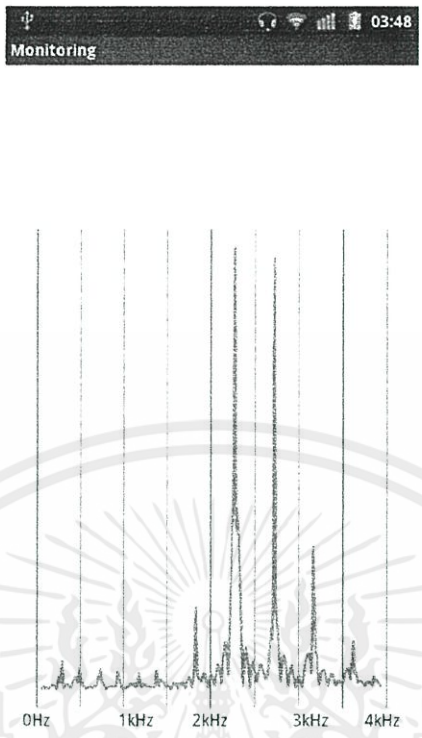
รูปที่ 4.49 สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมเมื่อผู้สวมใส่นั่งในลักษณะหลังค่อม

ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวม

ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวม

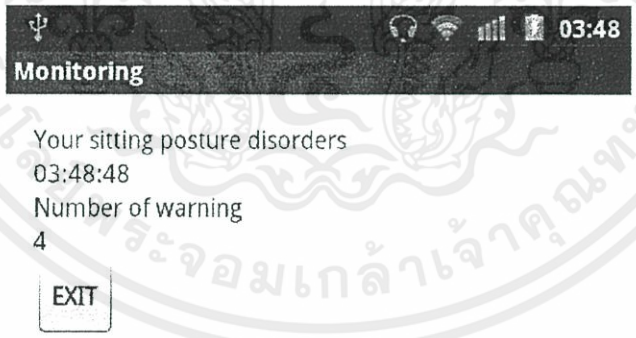
4.3.5.4 ผลการทดลองส่งสัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมจากผลการทดลองที่ 4.3.5.3 เข้าทางช่องสัญญาณไมโครโฟนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ เพื่อนำไปพลอตสเปคตรัมและประมวลผลเพื่อตรวจจับการนั่งในลักษณะหลังค่อมดังรูปที่ 4.50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.50 แอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณเมื่อผู้สวมใส่นั่งในลักษณะหลังค่อม

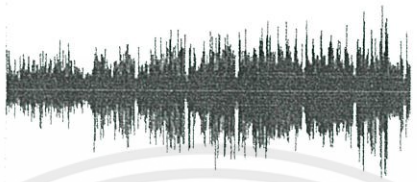
4.3.5.5 ผลการทดลองการแสดงผลบนหน้าจอแอปพลิเคชั่นตรวจจับค่าแมกนิจูดเมื่อมีการนั่งผิดไปจากลักษณะปกติแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ เมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะหลังค่อมแอปพลิเคชั่นจะแสดงข้อความว่า “Your sitting posture disorders” ดังรูปที่ 4.51



รูปที่ 4.51 หน้าจอแอปพลิเคชั่นเมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะหลังค่อม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

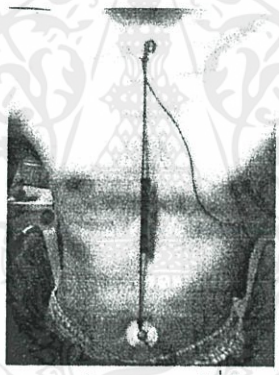
4.3.5.6 ผลการทดลองการสั่นเตือนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ โดยในการเก็บผลการทดลองจะใช้วิธีการอัดเสียงจากการสั่นของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ จากคอมพิวเตอร์ได้รูปสัญญาณเสียงจากการสั่นเตือนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ดังรูปที่ 4.52



รูปที่ 4.52 สัญญาณเสียงจากการสั่นเตือนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

4.3.6 ลักษณะการติดเซนเซอร์หนึ่งตัวที่ด้านหน้าของร่างกายผู้สวมใส่

ที่ด้านหน้าของร่างกายผู้สวมใส่ได้มีการวางเซนเซอร์ดังรูปที่ 4.53

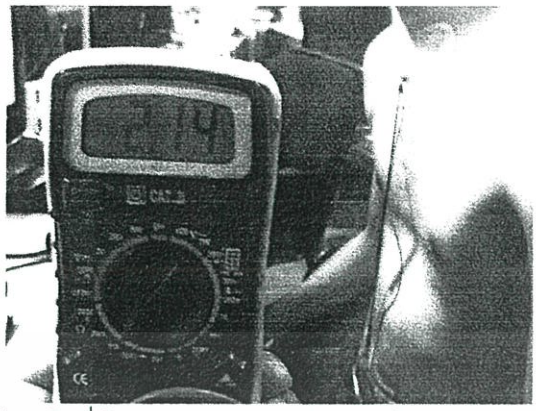


รูปที่ 4.53 ลักษณะการติดเซนเซอร์ที่ด้านหน้าของร่างกาย

4.3.7 ผลการทดลองเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง เมื่อเซนเซอร์ถูกติดไว้ด้านหน้าของร่างกาย

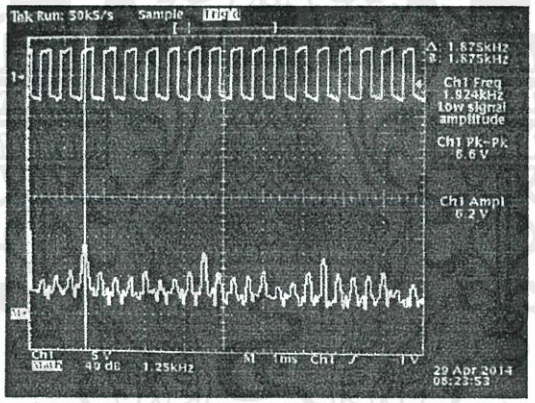
4.3.7.1 ผลการทดลองวัดค่าเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตั้งทางด้านหน้าของลำตัวเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง ค่าความต้านของเซนเซอร์ระยะยืดจะมีค่าเท่ากับ 2.14 กิโลโอห์ม ดังรูปที่ 4.54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.54 เมื่อผู้สวมใส่นั่งในลักษณะหลังตรงค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 2.14 กิโลโอห์ม

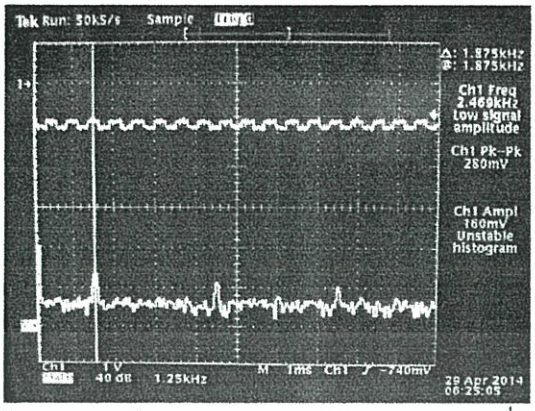
4.3.7.2 ผลการทดลองของวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ความถี่เปลี่ยนแปลงไปตามการยืดของเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตั้งทางด้านหน้าของลำตัวเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง ค่าความถี่ของวงจรกำเนิดสัญญาณจะมีค่าเท่ากับ 1.875 กิโลเฮิรตซ์ ดังรูปที่ 4.55



รูปที่ 4.55 สัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณของเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตั้งทางด้านหน้าของลำตัวเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง
ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณสี่เหลี่ยม
ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตจากวงจรกำเนิดสัญญาณสี่เหลี่ยม

4.3.7.3 ผลการทดลองของวงจรขยายสัญญาณผลรวม เมื่อใช้สัญญาณเอาต์พุตที่ได้จากการทดลองที่ 4.3.7.2 โดยจะทำการยกระดับสัญญาณด้วยไฟตรง 1 โวลต์ ได้สัญญาณเอาต์พุตและสเปคตรัมของสัญญาณเอาต์พุตวงจรขยายสัญญาณผลรวมดังรูปที่ 4.56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.56 สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง

ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวม

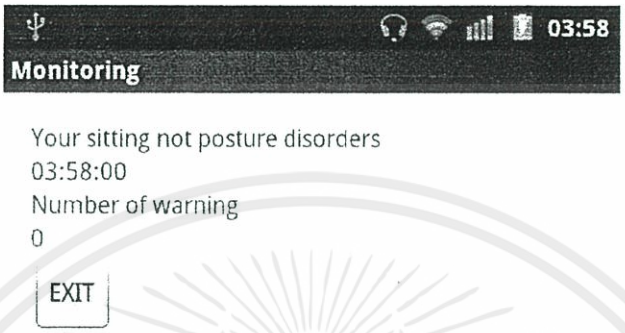
ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวม

4.3.7.4 ผลการทดลองส่งสัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมของผลการทดลองที่ 4.3.7.3 เข้าช่องสัญญาณไมโครโฟนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ เพื่อนำไปพลอตสเปคตรัมและประมวลผลเพื่อตรวจจับการนั่งในลักษณะหลังตรงดังรูปที่ 4.57

รูปที่ 4.57 แอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณเมื่อผู้สวมใส่นั่งในลักษณะหลังตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

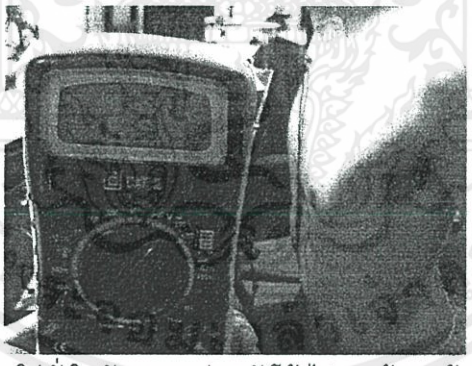
4.3.7.5 ผลการทดลองการแสดงผลบนหน้าจอแอปพลิเคชั่นตรวจจับค่าแมกนิจูดเมื่อมีการนั่งผิดไปจากลักษณะปกติแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ เมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะหลังตรงแอปพลิเคชั่นจะแสดงข้อความว่า “Your sitting not posture disorders” ดังรูปที่ 4.58



รูปที่ 4.58 หน้าจอแอปพลิเคชั่นเมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะหลังตรง

4.3.8 ผลการทดลองเมื่อผู้สวมใส่นั่งในลักษณะที่แผ่นหลังโค้งไปทางด้านหน้า เมื่อเซนเซอร์ถูกติดไว้ที่ด้านหน้าของร่างกายผู้สวมใส่

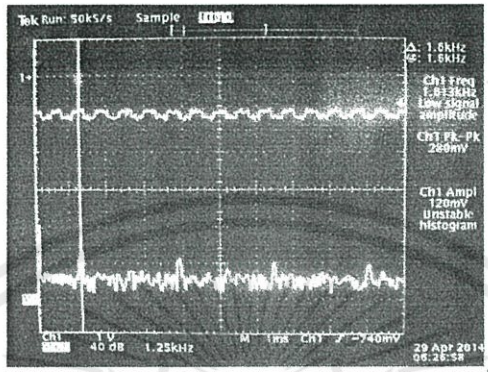
4.3.8.1 ผลการทดลองวัดค่าเซนเซอร์ระยะยืดที่ถูกติดอยู่ในแนวตั้งทางด้านหน้าของลำตัวเมื่อผู้สวมใส่นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้า ค่าความต้านของเซนเซอร์ระยะยืดจะมีค่าเท่ากับ 2.35 กิโลโอห์ม ดังรูปที่ 4.59



รูปที่ 4.59 เมื่อผู้สวมใส่นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้าค่าความต้านของเซนเซอร์ระยะยืดมีค่าเท่ากับ 2.35 กิโลโอห์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.8.2 ผลการทดลองของวงจรขยายสัญญาณผลรวมที่ใช้เอาต์พุตจากวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ได้จากการทดลองที่ 4.3.7.3 เมื่อผู้สวมใส่นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้า โดยจะทำการยกระดับสัญญาณด้วยไฟตรง 1 โวลต์ ได้สัญญาณเอาต์พุตและสเปคตรัมของสัญญาณเอาต์พุตของวงจรขยายสัญญาณผลรวมดังรูปที่ 4.60



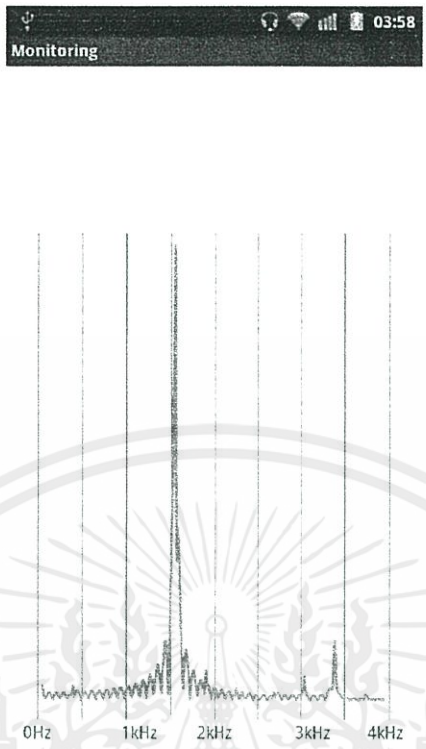
รูปที่ 4.60 สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมเมื่อผู้สวมใส่นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้า

ช่องสัญญาณที่ 1 สัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวม

ช่องสัญญาณ Math สเปคตรัมของสัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวม

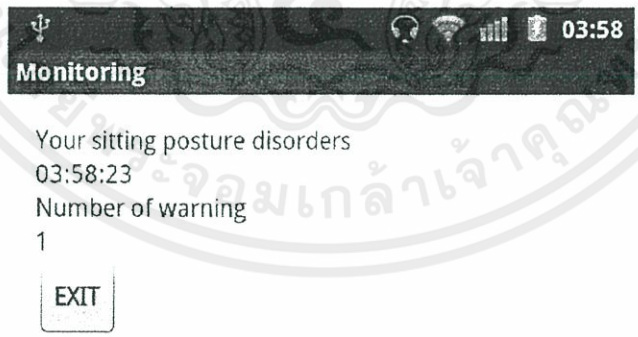
4.3.8.3 ผลการทดลองส่งสัญญาณเอาต์พุตจากวงจรขยายสัญญาณผลรวมที่ได้จากผลการทดลองที่ 4.3.8.2 เข้าช่องสัญญาณไมโครโฟนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ เพื่อนำไปพลอตสเปคตรัมและประมวลผลเพื่อตรวจจับการนั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้า ดังรูปที่ 4.61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.61 แอปพลิเคชั่นแสดงสเปคตรัมของสัญญาณเมื่อผู้สวมใส่นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้า

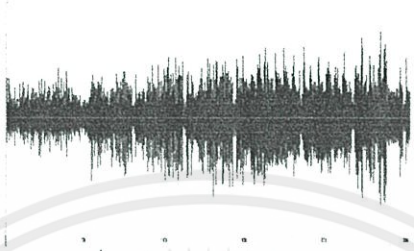
4.3.8.4 ผลการทดลองการแสดงผลบนหน้าจอแอปพลิเคชั่นตรวจจับค่าแมกนิจูดเมื่อมีการนั่งผิดปกติไปจากลักษณะปกติแบบเรียลไทม์บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ เมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้าแอปพลิเคชั่นจะแสดงข้อความว่า “Your sitting posture disorders” ดังรูปที่ 4.62



รูปที่ 4.62 หน้าจอแอปพลิเคชั่นเมื่อผู้สวมใส่อุปกรณ์นั่งในลักษณะแผ่นหลังโค้งไปทางด้านหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

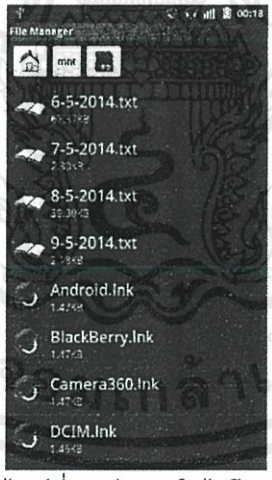
4.3.8.5 ผลการทดลองการสั่นเตือนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ โดยการเก็บผลการทดลองจะใช้วิธีการอัดเสียงจากการสั่นของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ จากคอมพิวเตอร์ได้รูปสัญญาณเสียงจากการสั่นเตือนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ดังรูปที่ 4.63



รูปที่ 4.63 สัญญาณเสียงจากการสั่นเตือนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

4.3.9 ผลการทดลองโปรแกรมแอปพลิเคชั่นบันทึกค่าลงในหน่วยความจำไมโครเอสดีการ์ดและแสดงผลแบบเรียลไทม์

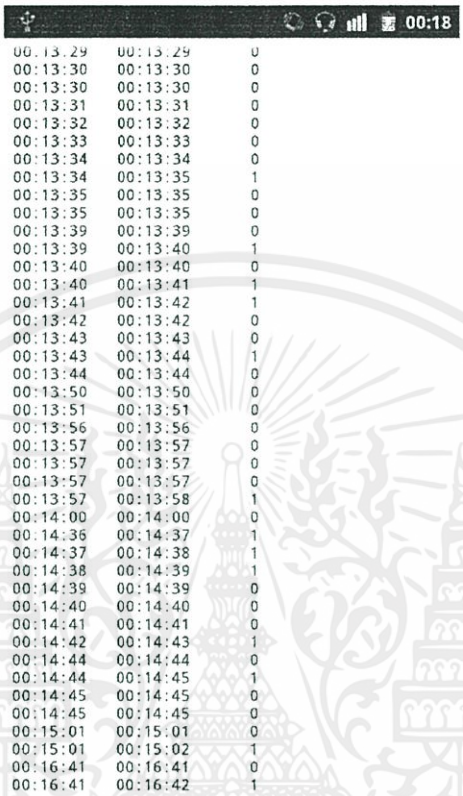
ก่อนที่จะทำการรันโปรแกรมแอปพลิเคชั่นตามแผนผังของโปรแกรมชุดคำสั่งนั้นจะต้องเชื่อมต่อสายยูเอสบีดังรูปที่ 4.20 ซึ่งได้ผลการทดลองการเขียนโปรแกรมแอปพลิเคชั่นเพื่อบันทึกตัวเลขลงในหน่วยความจำไมโครเอสดีการ์ดของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ได้ดังรูปที่ 4.64



รูปที่ 4.64 ไฟล์ที่ถูกเขียนลงในไมโครเอสดีการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเปิดไฟล์ชื่อ 9-5-2014.txt ในรูปที่ 4.64 จะเห็นได้ว่าสามารถบันทึกเวลาเมื่อผู้สวมใส่นั่งในลักษณะท่าทางที่ผิดปกติว่า เริ่ม ณ เวลาใด สิ้นสุดเมื่อไหร่ และเป็นระยะเวลานานเท่าใด ลงในหน่วยความจำไมโครเอสดีการ์ดของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ได้ดังรูปที่ 4.65

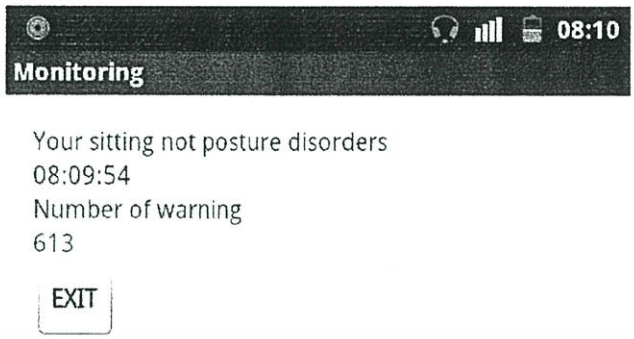


รูปที่ 4.65 ข้อมูลที่ถูกเขียนลงไปในไฟล์ที่ชื่อ 9-5-2014.txt

4.3.10 ผลการทดสอบอุปกรณ์บนตัวคนด้วยการนั่งในลักษณะหลังตรง และการนั่งในลักษณะหลังค่อมสลับกันเป็นระยะเวลา 2 ชั่วโมง 19 นาที 9 วินาที

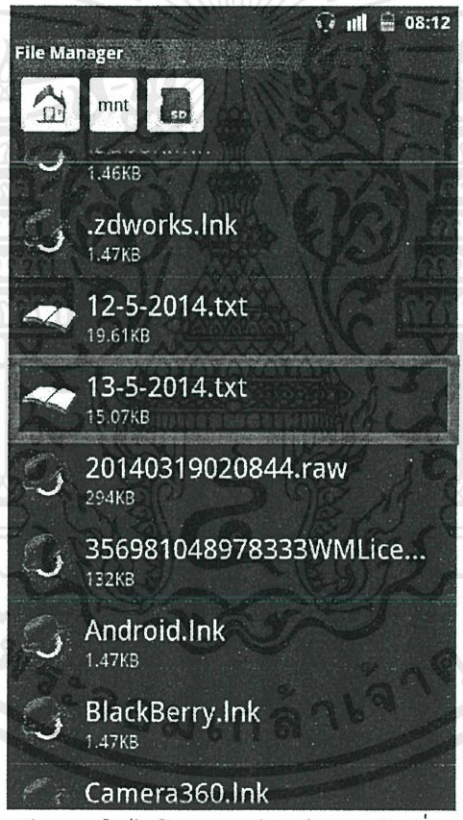
ในการทดสอบอุปกรณ์ ได้ทำการจับเวลาการนั่งในลักษณะหลังตรงและการนั่งในลักษณะหลังค่อมสลับกันเป็นระยะเวลาประมาณ 2 ชั่วโมง ได้จำนวนครั้งที่นั่งในลักษณะหลังค่อมเท่ากับ 613 ครั้ง ดังรูปที่ 4.66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.66 จำนวนครั้งที่นั่งในลักษณะหลังค่อมเท่ากับ 613 ครั้ง

ได้ผลการทดลองการเขียนโปรแกรมแอปพลิเคชั่นเพื่อบันทึกตัวเลขลงในหน่วยความจำไมโครเอสดีการ์ดของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ ได้ดังรูปที่ 4.67



รูปที่ 4.67 ไฟล์ที่ถูกเขียนลงในไมโครเอสดีการ์ดของวันที่ 13 พฤษภาคม 2014

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเปิดไฟล์ชื่อ 13-5-2014.txt จากในรูปที่ 4.68 จะเห็นได้ว่าเวลาที่บันทึกเมื่อผู้สวมใส่นั่งในลักษณะท่าทางที่ผิดปกติว่า เริ่ม ณ เวลาใด สิ้นสุดเมื่อไหร่ และเป็นระยะเวลานานเท่าใด ลงในหน่วยความจำไมโครเอสดีการ์ดของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ได้ดังตารางที่ 4.2

ตารางที่ 4.2 แสดงเวลาของไฟล์ชื่อ 13-5-2014.txt ในไมโครเอสดีการ์ด

08:49

startTime	stopTime	Period(s)
05:50:12	05:50:15	3
05:50:20	05:50:23	3
05:50:32	05:50:32	0
05:50:43	05:50:49	6
05:50:56	05:51:02	6
05:51:11	05:51:12	1
05:51:15	05:51:15	0
05:51:30	05:51:34	4
05:51:48	05:51:50	2
05:52:25	05:52:27	2
05:52:42	05:52:43	1
05:53:00	05:53:57	57
05:53:57	05:53:58	1
05:53:58	05:53:59	1
05:53:59	05:53:59	0
05:53:59	05:53:59	0
05:54:00	05:54:07	7
05:54:07	05:54:09	2
05:54:10	05:54:14	4
05:55:02	05:58:42	220
05:58:42	05:58:49	7
05:59:01	05:59:39	38
05:59:39	05:59:40	1
05:59:40	05:59:40	0
05:59:45	06:00:01	16
06:00:01	06:00:02	1
06:00:02	06:00:02	0
06:00:07	06:00:08	1
06:00:10	06:00:28	18
06:00:29	06:00:29	0
06:00:30	06:00:31	1
06:00:32	06:00:32	0
06:00:32	06:00:33	1
06:00:33	06:00:41	8
06:00:41	06:00:42	1
06:00:43	06:00:43	0
06:00:44	06:00:45	1
06:00:47	06:00:52	5
06:00:58	06:01:00	2
06:01:00	06:01:00	0

09:21

06:09:33	06:09:34	1
06:09:34	06:09:42	8
06:09:42	06:11:23	101
06:11:23	06:11:24	1
06:11:24	06:11:24	0
06:11:26	06:11:35	9
06:11:35	06:11:35	0
06:11:38	06:11:39	1
06:12:11	06:14:29	138
06:14:30	06:14:30	0
06:14:31	06:14:31	0
06:14:31	06:14:37	6
06:14:37	06:14:37	0
06:14:39	06:14:40	1
06:14:40	06:16:42	122
06:16:43	06:16:43	0
06:16:43	06:16:43	0
06:16:44	06:16:45	1
06:16:45	06:16:45	0
06:16:48	06:16:56	8
06:16:56	06:16:56	0
06:16:56	06:17:04	8
06:17:05	06:17:32	27
06:17:32	06:17:32	0
06:17:32	06:17:33	1
06:17:34	06:17:52	18
06:17:57	06:18:01	4
06:18:04	06:18:07	3
06:18:07	06:18:07	0
06:18:08	06:18:08	0
06:18:08	06:18:08	0
06:18:09	06:18:09	0
06:18:13	06:18:23	10
06:18:29	06:21:26	177
06:21:32	06:23:30	118
06:23:30	06:23:30	0
06:23:30	06:23:36	6

09:23

06:25:37	06:25:37	0
06:25:38	06:25:39	1
06:26:03	06:26:03	0
06:26:03	06:26:16	13
06:26:16	06:26:17	1
06:26:17	06:26:17	0
06:26:17	06:26:18	1
06:26:18	06:26:18	0
06:26:18	06:26:19	1
06:26:19	06:26:33	14
06:27:18	06:27:36	18
06:27:37	06:27:38	1
06:27:38	06:27:39	1
06:27:40	06:27:40	0
06:29:11	06:29:14	3
06:29:15	06:29:39	24
06:29:46	06:29:46	0
06:29:46	06:30:57	71
06:31:01	06:31:02	1
06:31:10	06:31:10	0
06:31:13	06:31:13	0
06:31:13	06:31:13	0
06:31:14	06:31:14	0
06:31:16	06:31:16	0
06:31:16	06:31:17	1
06:31:18	06:31:18	0
06:31:24	06:31:54	30
06:31:56	06:33:05	69
06:33:09	06:37:53	284
06:37:53	06:37:53	0
06:37:54	06:37:54	0
06:38:01	06:38:05	4
06:38:55	06:39:24	29
06:39:24	06:39:24	0
06:39:24	06:39:25	1

09:21

06:01:00	06:01:00	0
06:01:01	06:01:01	0
06:01:01	06:01:05	4
06:01:05	06:01:05	0
06:01:07	06:01:07	0
06:01:07	06:01:07	0
06:01:08	06:01:09	1
06:01:09	06:01:15	6
06:01:33	06:01:33	0
06:01:34	06:01:35	1
06:01:40	06:01:41	1
06:01:42	06:01:48	6
06:01:48	06:01:48	0
06:01:48	06:01:49	1
06:01:52	06:02:17	25
06:02:21	06:02:32	11
06:02:33	06:02:33	0
06:02:33	06:02:34	1
06:02:34	06:02:36	2
06:02:36	06:02:39	3
06:02:39	06:02:39	0
06:02:47	06:02:49	2
06:02:56	06:03:00	4
06:03:00	06:03:00	0
06:03:01	06:03:01	0
06:03:01	06:03:01	0
06:03:02	06:03:02	0
06:03:29	06:03:30	1
06:03:31	06:03:32	1
06:03:35	06:03:35	0
06:05:09	06:08:00	171
06:08:00	06:08:01	1
06:08:01	06:08:01	0
06:08:01	06:08:02	1
06:08:03	06:08:03	0
06:08:03	06:08:06	3
06:08:06	06:08:10	4
06:08:10	06:08:51	41
06:08:51	06:08:52	1
06:08:52	06:08:54	2
06:09:09	06:09:09	0
06:09:10	06:09:31	21
06:09:32	06:09:33	1

09:22

06:23:37	06:23:37	0
06:23:38	06:23:38	0
06:23:38	06:23:39	1
06:23:39	06:23:40	1
06:23:40	06:23:41	1
06:23:41	06:23:41	0
06:23:42	06:23:42	0
06:23:42	06:23:42	0
06:23:43	06:23:43	0
06:23:43	06:23:44	1
06:23:44	06:23:45	1
06:23:45	06:23:46	1
06:23:46	06:23:47	1
06:23:48	06:23:48	0
06:23:48	06:23:48	0
06:23:49	06:23:49	0
06:23:50	06:23:50	0
06:23:50	06:23:51	1
06:23:51	06:23:51	0
06:23:52	06:23:52	0
06:23:52	06:23:52	0
06:23:52	06:23:54	2
06:23:54	06:23:54	0
06:23:55	06:23:55	0
06:23:55	06:23:56	1
06:23:56	06:23:57	1
06:23:57	06:23:57	0
06:23:57	06:23:58	1
06:23:58	06:24:36	38
06:24:36	06:25:28	52
06:25:28	06:25:28	0
06:25:29	06:25:29	0
06:25:30	06:25:32	2
06:25:37	06:25:37	0
06:25:37	06:25:37	0

09:23

06:39:34	06:39:34	0
06:39:39	06:39:41	2
06:40:24	06:40:27	3
06:41:22	06:41:38	16
06:41:38	06:41:38	0
06:41:38	06:41:39	1
06:41:39	06:41:39	0
06:41:39	06:41:40	1
06:41:40	06:41:41	1
06:41:41	06:41:41	0
06:41:42	06:41:42	0
06:41:42	06:41:42	0
06:41:43	06:41:43	0
06:41:44	06:41:44	0
06:41:44	06:43:00	76
06:43:00	06:43:00	0
06:43:01	06:43:02	1
06:43:20	06:43:43	23
06:43:44	06:43:46	2
06:43:46	06:43:47	1
06:43:47	06:43:47	0
06:43:47	06:43:48	1
06:43:48	06:43:48	0
06:43:48	06:43:49	1
06:43:50	06:43:50	0
06:43:50	06:43:50	0
06:43:51	06:43:51	0
06:43:51	06:43:51	0
06:43:52	06:43:52	0
06:43:54	06:43:54	0
06:43:57	06:43:57	0
06:44:09	06:44:09	0
06:44:10	06:44:10	0
06:44:11	06:44:11	0
06:44:11	06:44:11	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2(ต่อ) แสดงเวลาของไฟล์ชื่อ 13-5-2014.txt ในไมโครเอสดีการ์ด

09:23

06:44:11	06:44:11	0
06:44:12	06:44:12	0
06:44:12	06:45:15	63
06:45:15	06:45:15	0
06:45:15	06:45:16	1
06:45:17	06:45:17	0
06:45:18	06:45:18	0
06:45:19	06:45:19	0
06:45:19	06:45:19	0
06:45:20	06:45:20	0
06:45:21	06:45:22	1
06:45:24	06:45:24	0
06:45:26	06:45:26	0
06:45:26	06:45:26	0
06:45:27	06:45:27	0
06:45:27	06:46:04	37
06:46:04	06:46:05	1
06:46:05	06:46:06	1
06:46:08	06:46:08	0
06:46:09	06:46:09	0
06:46:10	06:46:13	3
06:46:13	06:46:13	0
06:46:13	06:46:14	1
06:46:42	06:46:43	1
06:46:43	06:46:44	1
06:46:44	06:46:45	1
06:46:55	06:46:56	1
06:46:57	06:46:57	0
06:46:59	06:47:00	1
06:47:00	06:47:00	0
06:47:00	06:47:01	1
06:47:01	06:47:09	8
06:47:11	06:47:11	0
06:47:11	06:47:12	1
06:47:12	06:47:13	1

09:24

06:51:08	06:51:10	2
06:51:13	06:51:14	1
06:51:22	06:51:23	1
06:51:31	06:51:32	1
06:51:32	06:51:54	22
06:51:54	06:51:54	0
06:51:55	06:51:55	0
06:51:55	06:51:55	0
06:51:55	06:52:03	8
06:52:03	06:52:03	0
06:52:03	06:52:04	1
06:52:04	06:52:04	0
06:52:18	06:52:37	19
06:52:37	06:52:38	1
06:52:39	06:52:39	0
06:52:39	06:52:39	0
06:52:52	06:52:52	0
06:52:53	06:52:53	0
06:52:53	06:52:54	1
06:52:54	06:52:54	0
06:52:55	06:52:55	0
06:52:56	06:52:58	2
06:52:58	06:53:01	3
06:53:01	06:53:01	0
06:53:01	06:53:03	2
06:53:03	06:53:03	0
06:53:03	06:53:04	1
06:53:04	06:53:05	1
06:53:05	06:53:06	1
06:53:10	06:54:21	71
06:54:21	06:54:21	0
06:54:22	06:54:22	0
06:54:24	06:54:47	23
06:54:47	06:54:48	1
06:54:49	06:54:50	1

09:25

06:59:50	06:59:51	1
06:59:51	06:59:52	1
06:59:52	07:00:04	12
07:00:05	07:00:19	14
07:00:19	07:00:29	10
07:00:29	07:00:43	14
07:00:43	07:00:44	1
07:00:45	07:00:45	0
07:00:45	07:00:47	2
07:00:47	07:00:47	0
07:00:47	07:00:48	1
07:00:48	07:00:48	0
07:00:48	07:00:49	1
07:00:50	07:00:51	1
07:00:51	07:00:55	4
07:00:55	07:00:56	1
07:00:58	07:01:36	38
07:01:36	07:01:37	1
07:01:38	07:01:38	0
07:01:39	07:01:39	0
07:01:40	07:01:40	0
07:02:42	07:02:58	16
07:02:58	07:02:58	0
07:03:00	07:03:21	21
07:03:21	07:03:21	0
07:03:23	07:03:24	1
07:03:24	07:03:24	0
07:03:25	07:03:26	1
07:03:27	07:03:27	0
07:03:27	07:03:27	0
07:03:27	07:03:28	1
07:03:29	07:03:29	0
07:03:29	07:03:29	0
07:03:30	07:03:31	1
07:03:32	07:03:32	0

09:24

06:47:15	06:47:15	0
06:47:22	06:47:22	0
06:47:24	06:47:29	5
06:47:30	06:47:30	0
06:47:31	06:47:32	1
06:47:33	06:47:34	1
06:47:39	06:47:43	4
06:47:44	06:47:45	1
06:48:01	06:48:01	0
06:48:02	06:48:06	4
06:48:06	06:48:06	0
06:48:06	06:48:11	5
06:49:09	06:49:09	0
06:49:12	06:49:48	36
06:49:48	06:49:48	0
06:49:50	06:49:50	0
06:49:50	06:49:50	0
06:49:51	06:49:52	1
06:50:01	06:50:02	1
06:50:19	06:50:19	0
06:50:22	06:50:23	1
06:50:25	06:50:30	5
06:50:31	06:50:32	1
06:50:32	06:50:32	0
06:50:32	06:50:32	0
06:50:36	06:50:36	0
06:50:39	06:50:40	1
06:50:42	06:50:46	4
06:50:47	06:50:49	2
06:50:49	06:50:51	2
06:50:51	06:50:52	1
06:50:52	06:50:53	1
06:50:53	06:50:53	0
06:50:56	06:50:56	0
06:50:57	06:50:57	0

09:24

06:55:03	06:55:03	0
06:55:06	06:55:26	20
06:55:26	06:55:27	1
06:55:27	06:55:27	0
06:55:28	06:55:28	0
06:55:29	06:55:33	4
06:55:34	06:55:34	0
06:55:35	06:55:35	0
06:55:35	06:55:35	0
06:55:41	06:55:41	0
06:55:41	06:55:42	1
06:55:49	06:55:49	0
06:55:58	06:55:59	1
06:55:59	06:55:59	0
06:56:00	06:56:00	0
06:56:01	06:56:01	0
06:56:10	06:56:10	0
06:56:11	06:56:12	1
06:56:12	06:56:13	1
06:56:13	06:56:15	2
06:56:15	06:56:16	1
06:56:17	06:56:17	0
06:56:17	06:57:06	49
06:57:08	06:57:08	0
06:57:20	06:57:21	1
06:57:21	06:57:22	1
06:57:22	06:58:04	42
06:58:05	06:58:05	0
06:58:05	06:58:05	0
06:58:06	06:58:06	0
06:58:08	06:58:09	1
06:58:21	06:59:29	68
06:59:29	06:59:29	0
06:59:29	06:59:30	1
06:59:43	06:59:49	6

09:25

07:03:33	07:03:34	1
07:03:34	07:03:35	1
07:03:51	07:04:33	42
07:04:34	07:04:36	2
07:04:38	07:04:38	0
07:04:44	07:05:11	27
07:05:11	07:05:13	2
07:05:13	07:05:13	0
07:05:14	07:05:14	0
07:05:15	07:05:34	19
07:05:34	07:05:35	1
07:05:36	07:05:36	0
07:05:36	07:05:37	1
07:05:37	07:05:37	0
07:05:38	07:05:38	0
07:05:41	07:05:41	0
07:05:41	07:05:41	0
07:05:43	07:05:57	14
07:05:57	07:05:57	0
07:05:58	07:05:58	0
07:05:58	07:05:59	1
07:05:59	07:05:59	0
07:06:00	07:06:00	0
07:06:00	07:06:01	1
07:06:01	07:06:02	1
07:06:02	07:06:02	0
07:06:03	07:06:03	0
07:06:03	07:06:03	0
07:06:04	07:06:05	1
07:06:05	07:06:06	1
07:06:06	07:06:06	0
07:06:07	07:06:08	1
07:06:09	07:06:09	0
07:06:10	07:06:10	0
07:06:10	07:06:10	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2(ต่อ) แสดงเวลาของไฟล์ชื่อ 13-5-2014.txt ในไมโครเอสดีการ์ด

09:26

07:06:10	07:06:10	0
07:06:11	07:06:11	0
07:06:11	07:06:12	1
07:06:12	07:06:12	0
07:06:13	07:06:14	1
07:06:16	07:06:16	0
07:06:25	07:06:25	0
07:06:26	07:06:26	0
07:06:27	07:06:31	4
07:06:32	07:06:39	7
07:06:39	07:06:39	0
07:06:39	07:06:40	1
07:06:40	07:06:41	1
07:06:41	07:06:42	1
07:06:43	07:06:43	0
07:06:50	07:06:59	9
07:07:07	07:07:07	0
07:07:08	07:07:08	0
07:07:08	07:07:08	0
07:07:17	07:07:17	0
07:07:30	07:07:30	0
07:07:30	07:07:31	1
07:07:31	07:07:31	0
07:07:32	07:07:32	0
07:07:40	07:08:41	61
07:08:41	07:08:42	1
07:08:42	07:08:43	1
07:08:43	07:08:44	1
07:08:44	07:08:50	6
07:08:50	07:08:51	1
07:08:51	07:08:51	0
07:08:51	07:08:57	6
07:08:57	07:08:57	0
07:08:58	07:08:58	0
07:08:58	07:08:59	1

09:27

07:11:36	07:11:36	0
07:11:36	07:11:36	0
07:11:37	07:11:38	1
07:12:04	07:12:06	2
07:12:08	07:12:09	1
07:12:11	07:12:11	0
07:12:12	07:12:45	33
07:12:45	07:12:49	4
07:12:50	07:12:53	3
07:12:53	07:12:53	0
07:12:54	07:12:54	0
07:12:54	07:12:59	5
07:12:59	07:13:00	1
07:13:00	07:13:20	20
07:13:21	07:13:21	0
07:13:21	07:15:08	107
07:15:08	07:15:09	1
07:15:09	07:15:09	0
07:15:10	07:15:10	0
07:15:10	07:15:11	1
07:15:11	07:15:11	0
07:15:12	07:15:16	4
07:15:19	07:15:19	0
07:15:23	07:15:23	0
07:15:26	07:15:28	2
07:15:35	07:15:39	4
07:15:48	07:15:48	0
07:15:59	07:16:00	1
07:16:01	07:16:01	0
07:16:12	07:16:13	1
07:16:13	07:16:14	1
07:16:24	07:16:25	1
07:16:25	07:16:26	1
07:16:32	07:16:32	0
07:16:34	07:16:34	0

09:27

07:39:09	07:39:09	0
07:39:11	07:39:11	0
07:39:15	07:39:15	0
07:39:16	07:39:25	9
07:39:26	07:39:26	0
07:39:28	07:39:47	19
07:39:53	07:39:53	0
07:39:59	07:40:01	2
07:40:28	07:40:28	0
07:41:19	07:47:47	388
07:48:25	07:48:27	2
07:48:50	07:50:10	80
07:50:11	07:50:13	2
07:50:16	07:51:17	61
07:51:23	07:51:23	0
07:51:23	07:51:25	2
07:51:36	07:52:05	29
07:52:09	07:53:12	63
07:53:15	07:53:20	5
07:53:23	07:53:24	1
07:53:26	07:55:35	129
07:55:36	07:55:36	0
07:55:56	07:59:56	240
07:59:57	08:01:20	83
08:01:20	08:01:20	0
08:01:21	08:01:21	0
08:01:22	08:01:22	0
08:01:54	08:01:54	0
08:01:55	08:01:55	0
08:01:55	08:01:57	2
08:01:58	08:04:09	131
08:04:11	08:04:11	0
08:04:11	08:08:34	263
08:08:35	08:08:36	1
08:08:54	08:08:54	0

09:26

07:08:59	07:08:59	0
07:09:11	07:09:16	5
07:09:16	07:09:18	2
07:09:18	07:09:19	1
07:09:26	07:09:26	0
07:09:27	07:09:44	17
07:09:44	07:09:44	0
07:09:45	07:09:45	0
07:09:45	07:09:46	1
07:09:46	07:09:46	0
07:09:46	07:09:47	1
07:09:49	07:10:00	11
07:10:00	07:10:00	0
07:10:00	07:10:01	1
07:10:01	07:10:01	0
07:10:01	07:10:01	0
07:10:02	07:10:02	0
07:10:02	07:10:02	0
07:10:03	07:10:03	0
07:10:03	07:10:05	2
07:10:06	07:10:06	0
07:10:06	07:10:08	2
07:10:08	07:10:09	1
07:10:09	07:10:10	1
07:10:11	07:10:11	0
07:10:11	07:10:13	2
07:10:13	07:10:13	0
07:11:23	07:11:26	3
07:11:27	07:11:27	0
07:11:27	07:11:28	1
07:11:29	07:11:29	0
07:11:29	07:11:29	0
07:11:30	07:11:30	0
07:11:30	07:11:30	0
07:11:32	07:11:35	3

09:27

07:16:35	07:18:07	92
07:18:08	07:18:10	2
07:18:12	07:18:13	1
07:18:14	07:18:14	0
07:18:14	07:18:14	0
07:18:22	07:18:23	1
07:18:31	07:18:33	2
07:18:35	07:18:37	2
07:18:40	07:18:50	10
07:19:28	07:19:28	0
07:19:29	07:20:39	70
07:20:39	07:20:39	0
07:20:40	07:21:06	26
07:21:06	07:21:07	1
07:21:07	07:21:07	0
07:21:08	07:23:41	153
07:23:43	07:23:46	3
07:23:50	07:28:15	265
07:28:31	07:28:36	5
07:29:20	07:30:20	60
07:30:22	07:30:22	0
07:30:23	07:30:23	0
07:30:55	07:31:01	6
07:31:32	07:31:33	1
07:31:33	07:31:33	0
07:31:35	07:31:35	0
07:31:50	07:31:50	0
07:31:50	07:31:52	2
07:31:53	07:31:54	1
07:32:04	07:32:05	1
07:32:24	07:32:25	1
07:32:25	07:35:01	156
07:35:03	07:35:03	0
07:35:06	07:35:06	0
07:35:24	07:39:07	223

00:01

Click here to open file from the beginning or just a little before current position (15329)...

08:08:54	08:08:56	2
08:08:57	08:08:57	0

ผลการทดลองการนับจำนวนครั้งที่โปรแกรมแอปพลิเคชั่นตรวจจับการนั่งในลักษณะหลังค่อมไม่ได้เป็นดังตารางที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 จำนวนครั้งที่สามารถตรวจจับได้และไม่ได้

ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ
1	✓	30	✗	59	✓	88	✗
2	✓	31	✗	60	✓	89	✓
3	✗	32	✗	61	✗	90	✗
4	✓	33	✗	62	✓	91	✓
5	✓	34	✓	63	✓	92	✓
6	✓	35	✗	64	✗	93	✗
7	✗	36	✗	65	✗	94	✗
8	✓	37	✗	66	✗	95	✓
9	✓	38	✓	67	✗	96	✗
10	✓	39	✓	68	✓	97	✓
11	✓	40	✗	69	✓	98	✓
12	✗	41	✗	70	✗	99	✗
13	✗	42	✗	71	✓	100	✗
14	✓	43	✓	72	✓	101	✓
15	✗	44	✗	73	✗	102	✓
16	✗	45	✗	74	✓	103	✓
17	✓	46	✗	75	✗	104	✗
18	✓	47	✓	76	✓	105	✓
19	✓	48	✓	77	✓	106	✓
20	✓	49	✗	78	✓	107	✗
21	✓	50	✓	79	✓	108	✓
22	✓	51	✓	80	✓	109	✓
23	✓	52	✓	81	✗	110	✓
24	✗	53	✗	82	✓	111	✓
25	✓	54	✓	83	✓	112	✗
26	✗	55	✓	84	✓	113	✗
27	✗	56	✓	85	✓	114	✗
28	✗	57	✗	86	✓	115	✗
29	✓	58	✗	87	✓	116	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3(ต่อ) จำนวนครั้งที่สามารถตรวจจับได้และไม่ได้

ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ
117	✓	146	✓	175	✓	204	✗
118	✓	147	✓	176	✗	205	✗
119	✗	148	✗	177	✗	206	✓
120	✓	149	✓	178	✗	207	✗
121	✓	150	✓	179	✗	208	✓
122	✗	151	✓	180	✗	209	✓
123	✗	152	✗	181	✓	210	✓
124	✓	153	✗	182	✗	211	✓
125	✓	154	✓	183	✓	212	✗
126	✓	155	✗	184	✓	213	✓
127	✗	156	✗	185	✓	214	✗
128	✗	157	✗	186	✗	215	✓
129	✗	158	✓	187	✗	216	✗
130	✗	159	✗	188	✓	217	✗
131	✓	160	✓	189	✓	218	✗
132	✓	161	✓	190	✗	219	✗
133	✓	162	✗	191	✓	220	✗
134	✓	163	✓	192	✗	221	✗
135	✗	164	✗	193	✓	222	✗
136	✗	165	✓	194	✓	223	✗
137	✗	166	✓	195	✓	224	✗
138	✗	167	✓	196	✗	225	✗
139	✓	168	✓	197	✓	226	✗
140	✗	169	✓	198	✗	227	✗
141	✗	170	✗	199	✓	228	✗
142	✗	171	✓	200	✓	229	✓
143	✓	172	✓	201	✗	230	✗
144	✗	173	✗	202	✗	231	✓
145	✗	174	✓	203	✗	232	✗

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3(ต่อ) จำนวนครั้งที่สามารถตรวจจับได้และไม่ได้

ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ
233	✗	262	✗	291	✓	320	✗
234	✗	263	✗	292	✓	321	✓
235	✗	264	✗	293	✓	322	✗
236	✗	265	✗	294	✗	323	✓
237	✓	266	✓	295	✗	324	✓
238	✗	267	✓	296	✗	325	✓
239	✗	268	✓	297	✓	326	✓
240	✗	269	✓	298	✓	327	✗
241	✗	270	✗	299	✓	328	✗
242	✓	271	✓	300	✓	329	✓
243	✓	272	✗	301	✓	330	✓
244	✓	273	✓	302	✗	331	✓
245	✗	274	✗	303	✗	332	✗
246	✗	275	✓	304	✗	333	✓
247	✓	276	✗	305	✓	334	✓
248	✗	277	✗	306	✗	335	✗
249	✓	278	✗	307	✓	336	✗
250	✓	279	✓	308	✗	337	✓
251	✓	280	✓	309	✓	338	✓
252	✓	281	✗	310	✓	339	✗
253	✓	282	✓	311	✗	340	✗
254	✗	283	✓	312	✗	341	✗
255	✓	284	✓	313	✗	342	✓
256	✗	285	✗	314	✗	343	✗
257	✓	286	✗	315	✓	344	✓
258	✓	287	✗	316	✗	345	✗
259	✗	288	✓	317	✗	346	✗
260	✓	289	✓	318	✓	347	✗
261	✓	290	✓	319	✓	348	✗

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3(ต่อ) จำนวนครั้งที่สามารถตรวจจับได้และไม่ได้

ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ
349	✓	378	✗	407	✓	436	✗
350	✓	379	✓	408	✓	437	✗
351	✓	380	✓	409	✗	438	✗
352	✓	381	✓	410	✗	439	✓
353	✗	382	✓	411	✓	440	✗
354	✓	383	✓	412	✓	441	✓
355	✗	384	✓	413	✗	442	✗
356	✓	385	✗	414	✓	443	✗
357	✓	386	✗	415	✗	444	✗
358	✓	387	✗	416	✗	445	✓
359	✗	388	✓	417	✗	446	✓
360	✗	389	✗	418	✗	447	✗
361	✗	390	✓	419	✓	448	✓
362	✓	391	✗	420	✗	449	✓
363	✓	392	✓	421	✗	450	✓
364	✗	393	✗	422	✓	451	✗
365	✓	394	✓	423	✗	452	✓
366	✓	395	✗	424	✗	453	✗
367	✓	396	✗	425	✓	454	✗
368	✓	397	✓	426	✓	455	✗
369	✓	398	✗	427	✗	456	✗
370	✓	399	✗	428	✗	457	✗
371	✓	400	✓	429	✗	458	✓
372	✓	401	✗	430	✓	459	✗
373	✓	402	✓	431	✓	460	✓
374	✗	403	✓	432	✗	461	✓
375	✓	404	✓	433	✓	462	✓
376	✗	405	✓	434	✗	463	✓
377	✓	406	✗	435	✗	464	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3(ต่อ) จำนวนครั้งที่สามารถตรวจจับได้และไม่ได้

ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ
465	✓	494	✓	523	✓	552	✓
466	✓	495	✓	524	✗	553	✗
467	✗	496	✗	525	✗	554	✓
468	✓	497	✓	526	✓	555	✓
469	✗	498	✗	527	✗	556	✗
470	✗	499	✓	528	✓	557	✓
471	✓	500	✗	529	✗	558	✓
472	✗	501	✓	530	✗	559	✓
473	✓	502	✗	531	✓	560	✓
474	✓	503	✗	532	✓	561	✓
475	✓	504	✗	533	✗	562	✗
476	✗	505	✗	534	✓	563	✗
477	✓	506	✓	535	✗	564	✓
478	✗	507	✗	536	✓	565	✓
479	✗	508	✗	537	✓	566	✗
480	✓	509	✓	538	✓	567	✗
481	✗	510	✓	539	✓	568	✗
482	✓	511	✓	540	✗	569	✓
483	✓	512	✗	541	✗	570	✓
484	✗	513	✓	542	✓	571	✓
485	✓	514	✓	543	✓	572	✓
486	✗	515	✓	544	✓	573	✓
487	✗	516	✗	545	✗	574	✗
488	✗	517	✗	546	✗	575	✗
489	✗	518	✓	547	✓	576	✓
490	✗	519	✓	548	✓	577	✗
491	✓	520	✓	549	✓	578	✗
492	✗	521	✗	550	✓	579	✗
493	✓	522	✓	551	✗	580	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3(ต่อ) จำนวนครั้งที่สามารถตรวจจับได้และไม่ได้

ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ	ครั้งที่	การตรวจจับ
581	✗	589	✓	597	✓	605	✗
582	✓	590	✓	598	✗	606	✓
583	✗	591	✗	599	✓	607	✓
584	✓	592	✓	600	✓	608	✗
585	✗	593	✓	601	✗	609	✓
586	✓	594	✓	602	✗	610	✓
587	✓	595	✓	603	✗	611	✗
588	✓	596	✓	604	✗	612	✓
						613	✗

จากนั้นนำมาหาค่าความผิดพลาดโดยนับจำนวนครั้งที่ตรวจจับได้เท่ากับ 327 ครั้ง และจำนวนครั้งที่ไม่สามารถตรวจจับได้เท่ากับ 286 ครั้ง ได้เปอร์เซ็นต์ค่าความผิดพลาดดังนี้

$$Error = \left|\frac{theory - practice}{theory}\right| \times 100$$

$$= \left|\frac{613 - 327}{613}\right| \times 100$$

$$= 46.66\%$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ปริญญานิพนธ์นี้นำเสนอการออกแบบและสร้างเครื่องช่วยเตือนจัดบุคลิกท่าทางของหลัง เพื่อช่วยลดอาการหลังคด หลังงอ เครื่องจะประกอบไปด้วย วงจรกำเนิดสัญญาณ เซนเซอร์ระยะยืด สายนำสัญญาณ โทรศัพท์มือถือระบบปฏิบัติแอนดรอยด์ เมื่อผู้สวมใส่ขยับลำตัว เซนเซอร์จะตรวจจับท่าทางการห่อไหล่ที่ผิดรูปทรงไปจากปกติ จากนั้นวงจรกำเนิดสัญญาณจะสร้างสัญญาณที่มีความถี่ค่าหนึ่งออกมาเพื่อส่งเข้าโทรศัพท์มือถือระบบปฏิบัติแอนดรอยด์โดยผ่านสายนำสัญญาณ โปรแกรมแอปพลิเคชั่นบนโทรศัพท์มือถือระบบปฏิบัติแอนดรอยด์จะทำการประมวลผลเพื่อตรวจจับท่าการห่อไหล่ที่ผิดปกติไปจากค่าความถี่ของสัญญาณที่ได้รับ โดยจะทำการบันทึกค่าจำนวนครั้งที่ท่าทางของหลังผิดปกติพร้อมทั้งระยะเวลาที่เกิดอาการผิดปกตินั้นลงในหน่วยความจำไมโครเอสดีการ์ด

5.2 ข้อเสนอแนะ

อุปกรณ์นี้จะทำงานได้อย่างมีประสิทธิภาพมากขึ้น ถ้าสามารถออกแบบและสร้างเซนเซอร์ระยะยืดที่สามารถติดอยู่บนเสื้อรัดรูปทรงได้ แต่ทั้งนี้ไม่สามารถออกแบบและสร้างขึ้นได้ สาเหตุเกิดขึ้นจากข้อจำกัดทางด้านอุปกรณ์ที่เป็นตัวเสื้อ ซึ่งเสื้อที่จะนำมาใช้ร่วมกับเซนเซอร์นั้นควรใช้เสื้อที่สามารถรัดรูปแนบสนิทกับลำตัวของผู้สวมใส่ และข้อจำกัดอีกข้อหนึ่ง คือช่องสัญญาณไมโครโฟนของโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์สามารถรองรับสัญญาณอินพุตได้ในช่วงความถี่ 0 เฮิรตซ์ ถึง 4 กิโลเฮิรตซ์ เท่านั้น ทำให้มีปัญหาทางด้านการออกแบบช่วงความถี่ที่เซนเซอร์ระยะยืดทำงานร่วมกับวงจรกำเนิดสัญญาณมีไม่เพียงพอ นอกจากนี้ข้อจำกัดทางด้านช่วงกว้างของความถี่บนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์นี้ ยังทำให้เป็นการจำกัดจำนวนของเซนเซอร์ระยะยืดที่จะนำไปใช้งานอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] Greg Milette, Adam Stroud, “Professinal Android Sensor Programming”, John Wiley & Sons, Inc, 2012.

[2] ผศ.ดร. จีรสุดา โกษียาภรณ์. วิศวกรรมอิเล็กทรอนิกส์. พิมพ์ครั้งที่ 1. กรุงเทพฯ : ห้างหุ้นส่วนจำกัด วี.เจ. พริ้นติ้ง, 2551.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

SOURCE CODE

โปรแกรมชุดคำสั่งของแอปพลิเคชั่นในปริญญานิพนธ์นี้

โปรแกรมชุดคำสั่งส่วนของการประมวลผล

```
package com.t109.tabletest;

import java.util.ArrayList;

import java.util.LinkedList;

import java.util.Queue;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

import java.io.IOException;

import java.lang.System;

import java.lang.Object;

import android.media.AudioFormat;

import android.media.AudioRecord;

import android.os.AsyncTask;

import android.os.Bundle;

import android.os.Environment;

import android.os.Handler;

import android.os.Vibrator;

import android.app.Activity;

import android.app.Notification;

import android.app.NotificationManager;

import android.content.Context;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
import android.text.format.Time;

import android.util.Log;

import android.view.Menu;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.TextView;

public class MainActivity extends Activity {

private double graph2LastXValue = 0d;

private double graph2LastYValue = 0d;

public float points[] = new float[256];

ChartView lineChart;

private final Handler mHandler = new
Handler();

private final Handler mHandler2 = new
Handler();

private Runnable mTimer1;

private Runnable mTimer2;

protected InitTask _initTask;

Queue<String> dataQueue = new
LinkedList<String>();

Queue<String> chartDataQueue = new
LinkedList<String>();

public AudioRecord audioRecord;

public int mSamplesRead; // how many
samples read

public int recordingState;

public int buffersizebytes;

public int channelConfiguration =
AudioFormat.CHANNEL_IN_MONO;

public int audioEncoding =
AudioFormat.ENCODING_PCM_16BIT;

public static short[] buffer;

public static final int SAMPPERSEC = 8000;

public float[] bufrealA = new float[512];

public float[] bufrealB = new float[128];

public float[] bufrealC = new float[128];

public FFT dFT;

public Complex[] fftTempArray = new
Complex[512];

public Complex[] fftArray = new
Complex[512];

public int power28 = 256;

public int power29 = 512;

public int power210 = 1024;

public float max = 0;

public int imax;

public byte[] data;

public Time currentTime = new Time();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
public int startHour, startMinute,
startSecond, endHour, endMinute,
endSecond, timeshowHour,
timeshowMinute, timeshowSecond;

public long epochStart, epochEnd;

public String dottxt = ".txt";

public int clock = 2;

public int x = 0;

public int counter = 0;

public String filename;

public FileOutputStream fos;

Vibrator vibretor;

TextView ShowText,ShowNumber,cout;

public int maxbufB;

public int maxbufC;

Button on, off;

int LED_NOTIFICATION_ID = 0;

 Context context;

@Override

protected void onCreate(Bundle
savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_main);

ShowText = (TextView)
findViewById(R.id.ShowText);

cout = (TextView) findViewById(R.id.cout);

ShowNumber = (TextView)
findViewById(R.id.ShowNumber);

buffersizebytes =
AudioRecord.getMinBufferSize(SAMPPERSEC,c
hannelConfiguration, audioEncoding);

buffer = new short[buffersizebytes];

audioRecord = new
AudioRecord(android.media.MediaRecorder.
AudioSource.MIC, SAMPPERSEC,
channelConfiguration, audioEncoding,
buffersizebytes);

View btnExit = findViewById(R.id.exit1);

btnExit.setOnClickListener(new
OnClickListener(){

@Override

public void onClick(View v) {

switch(v.getId()){

case R.id.exit1:

finish();

break; }} });

initTask = new InitTask();

initTask.execute(this); }

@Override

public boolean onCreateOptionsMenu(Menu
menu) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
getMenuInflater().inflate(R.menu.main,
menu);

return true; }

private int getMax(float[] bufreal) {

float max = bufreal[0];

int counter = 0;

int Y = bufreal.length;

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

if (bufreal[i] > max) {

max = bufreal[i];

counter = i; } }

return counter; }

private void setChartData1() {

if (dataQueue.size() == power28) {

for (int i = 0; i < power28; i++) {

points[i] = Float.valueOf(dataQueue.poll()); }


float lastMin = points[0];

float offsetValue = 0;

for (int k = 1; k < power28; k++) {

if (points[k] < lastMin)

lastMin = points[k]; }

if (lastMin < 0) {

offsetValue = lastMin * (-1);

} else offsetValue = lastMin;

for (int k = 0; k < power28; k++) {

points[k] = points[k] + offsetValue;

}

lineChart.setChartData(points); } }

protected class InitTask extends
AsyncTask<Context, Float, String> {

@Override

protected String doInBackground(Context...
params) {

audioRecord.startRecording();

currentTime.setToNow();

try {

filename =
String.valueOf(currentTime.monthDay) + "-"

+ String.valueOf(currentTime.month+1) + "-
"+ String.valueOf(currentTime.year) + dottxt;

File file = new
File(Environment.getExternalStorageDirectory
(),filename);

if (!file.exists()) {

byte[] data = new
String("\nstartTime\t\tstopTime\t\t\tPeriod(s)
\n").getBytes();

fos = new FileOutputStream(file);

fos.write(data);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
fos.flush();

fos.close();}

} catch (FileNotFoundException e) {}

catch (IOException e) {}

int dayTemp = currentTime.monthDay;

int today;

while (true) {

try {

mSamplesRead = audioRecord.read(buffer,
0, buffersizebytes);

for (int i = 0; i < power29; i++) {

Complex(buffer[i], 0);

fftTempArray[i] = new Complex((double)
buffer[i], 0); }

fftArray = dFT.fft(fftTempArray);

for (int i = 0; i < 512; i++) {

bufrealA[i] = (float) Math.sqrt((fftArray[i].re) *
(fftArray[i].re) + (fftArray[i].im)* (fftArray[i].im));
        }

System.arraycopy(bufrealA, 115, bufrealB, 0,
70);

maxbufB = getMax(bufrealB);

System.arraycopy(bufrealA, 186, bufrealC, 0,
70);

maxbufC = getMax(bufrealC);

currentTime.setToNow();

today = currentTime.monthDay;

if (today != dayTemp) {

try {

filename =
String.valueOf(currentTime.monthDay) + "-"
+ String.valueOf(currentTime.month) + "-" +
String.valueOf(currentTime.year) + dottxt;

File file = new
File(Environment.getExternalStorageDirectory
(),filename);

byte[] data = new
String("\nstartTime\tstopTime\t
Period(s)\n").getBytes();

fos = new FileOutputStream(file);

fos.write(data);

fos.flush();

fos.close();

} catch (FileNotFoundException e) {

} catch (IOException e) {}

dayTemp = today; }

if (maxbufB<=32&&maxbufC<=19) {

vibretor = (Vibrator)
getSystemService(Context.VIBRATOR_SERVIC
E);

vibretor.vibrate(100);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (x == 0) {

counter++;

currentTime.setToNow();

epochStart = currentTime.toMillis(true) /
1000;

startHour = currentTime.hour;

startMinute = currentTime.minute;

startSecond = currentTime.second;

x = 1; }

} else if (x == 1) {

x = 0;

currentTime.setToNow();

epochEnd = currentTime.toMillis(true) /
1000;

long diffTime = epochEnd - epochStart;

try {


String startHourStr, startMinuteStr,
startSecondStr, endHourStr, endMinuteStr,
endSecondStr;

endHour = currentTime.hour;

endMinute = currentTime.minute;

endSecond = currentTime.second;

if (startHour < 10) {

startHourStr = "0" + String.valueOf(startHour);

} else startHourStr =
String.valueOf(startHour);

if (startMinute < 10) {

startMinuteStr = "0"+
String.valueOf(startMinute);

} else startMinuteStr =
String.valueOf(startMinute);

if (startSecond < 10) {

startSecondStr = "0"+
String.valueOf(startSecond);

} else startSecondStr =
String.valueOf(startSecond);

if (endHour < 10) {

endHourStr = "0" + String.valueOf(endHour);

} else endHourStr = String.valueOf(endHour);

if (endMinute < 10) {

endMinuteStr = "0" +
String.valueOf(endMinute);

} else endMinuteStr =
String.valueOf(endMinute);

if (endSecond < 10) {

endSecondStr = "0" +
String.valueOf(endSecond);

} else endSecondStr =
String.valueOf(endSecond);

byte[] data = new String("\n" + startHourStr
+ ":"+ startMinuteStr + ":" + startSecondStr+
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
"\t  " + endHourStr + ":" + endMinuteStr+ ":"
+ endSecondStr + "\t\t\t\t"+
String.valueOf(diffTime)).getBytes();

File file = new
File(Environment.getExternalStorageDirectory
(),filename);

fos = new FileOutputStream(file, true);

fos.write(data);

fos.flush();

fos.close();

} catch (FileNotFoundException e) {}

catch (IOException e) {} }

runOnUiThread(new Runnable() {

public void run() {

currentTime.setToNow();

timeshowHour = currentTime.hour;

timeshowMinute = currentTime.minute;

timeshowSecond = currentTime.second;

String showHourStr, showMinuteStr,
showSecondStr;

if (timeshowHour < 10) {

showHourStr = "0" +
String.valueOf(timeshowHour);

} else showHourStr =
String.valueOf(timeshowHour);

if (timeshowMinute < 10) {

showMinuteStr = "0" +
String.valueOf(timeshowMinute);

} else showMinuteStr =
String.valueOf(timeshowMinute);

if (timeshowSecond < 10) {

showSecondStr = "0" +
String.valueOf(timeshowSecond);

} else showSecondStr =
String.valueOf(timeshowSecond);

ShowNumber.setText(showHourStr + ":" +
showMinuteStr + ":"+ showSecondStr);

cout.setText(String.valueOf(counter));

if(maxbufB<=32&&maxbufC<=19)

ShowText.setText("Your sitting posture
disorders");

else ShowText.setText("Your sitting not
posture disorders");}});

} catch (Exception e) {} } }

@Override

protected void onPreExecute() {

super.onPreExecute(); }

@Override

protected void onProgressUpdate(Float...
values) {

super.onProgressUpdate(values); }

@Override
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
protected void onPostExecute(String result) {

super.onPostExecute(result);  } }

@Override

public void onPause() {

super.onPause();

if (audioRecord != null) {

audioRecord.release();

audioRecord = null; } } }
```

โปรแกรมชุดคำสั่งส่วนของ ChartView

```
package com.t109.tabletest;

import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.util.AttributeSet;
import android.view.View;

public class ChartView extends View {
    .
        private int width = 320;
        private int heigth = 240;

   public ChartView(Context context,
AttributeSet attrs) {
      super(context, attrs);
   }
   @Override
   protected void onMeasure(int
widthMeasureSpec, int heightMeasureSpec) {
      super.onMeasure(widthMeasureSpec,
heightMeasureSpec);
      width =
MeasureSpec.getSize(widthMeasureSpec);
      heigth =
MeasureSpec.getSize(heightMeasureSpec);
      this.setMeasuredDimension(width ,
heigth);
   }
   public void setChartData(float[]
datapoints) {
      invalidate();
   }
   @Override
   protected void onDraw(Canvas canvas) {
         drawLineChart(canvas);
   }
   private void drawLineChart(Canvas
canvas) {
      canvas.drawColor(Color.WHITE);
   }
}
```

โปรแกรมชุดคำสั่งส่วนของ Activity_main.xml

```
<LinearLayout
xmlns:android="http://schemas.android.com
/apk/res/android"

xmlns:tools="http://schemas.android.com/t
ools"
   android:id="@+id/LinearLayout1"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:orientation="vertical"

android:paddingBottom="@dimen/activity_v
ertical_margin"

android:paddingLeft="@dimen/activity_horiz
ontal_margin"

android:paddingRight="@dimen/activity_hori
zontal_margin"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

    <TextView
        android:id="@+id/ShowText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="0" />

    <TextView
        android:id="@+id/ShowNumber"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="0" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Number of warning" />

    <TextView
        android:id="@+id/cout"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="0" />

    <Button
        android:id="@+id/exit1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="EXIT" />

        <com.t109.tabletest.ChartView
        android:id="@+id/lineChartView1"
         android:layout_width="fill_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:background="#444"
        android:padding="16dp" />

</LinearLayout>
```

โปรแกรมชุดคำสั่งส่วนของ Manifest

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.t109.tabletest"
    android:versionCode="1"
    android:versionName="1.0" >
    <uses-permission
android:name="android.permission.RECORD_AUDIO"/>
    <uses-permission
android:name="android.permission.WAKE_LOCK"/>
    <uses-permission
android:name="android.permission.VIBRATE" />
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission
android:name="android.permission.READ_EXTERNAL_STORAGE"/>
    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="17" />
    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
android:name="com.t109.tabletest.MainActi
vity"
        android:label="@string/app_name"
>
        <intent-filter>
            <action
android:name="android.intent.action.MAIN"
/>
            <category
android:name="android.intent.category.LAU
NCHER" />
        </intent-filter>
      </activity>
   </application>
</manifest>
```

คลาส FFT cooley tukey

```
package com.t109.tabletest;


public class FFT {
        // compute the FFT of x[], assuming
its length is a power of 2
   public static Complex[] fft(Complex[] x) {
      int N = x.length;

      // base case
      if (N == 1) return new Complex[] { x[0]
};

      // radix 2 Cooley-Tukey FFT
      if (N % 2 != 0) { throw new
RuntimeException("N is not a power of 2"); }

      // fft of even terms
      Complex[] even = new Complex[N/2];
      for (int k = 0; k < N/2; k++) {
         even[k] = x[2*k];
      }
      Complex[] q = fft(even);

      // fft of odd terms
      Complex[] odd  = even;  // reuse the
array
      for (int k = 0; k < N/2; k++) {
         odd[k] = x[2*k + 1];
      }
      Complex[] r = fft(odd);

      // combine
      Complex[] y = new Complex[N];
      for (int k = 0; k < N/2; k++) {
         double kth = -2 * k * Math.PI / N;
         Complex wk = new
Complex(Math.cos(kth), Math.sin(kth));
         y[k]       = q[k].plus(wk.times(r[k]));
         y[k + N/2] =
q[k].minus(wk.times(r[k]));
      }
      return y;
   }


  // compute the inverse FFT of x[],
assuming its length is a power of 2
   public static Complex[] ifft(Complex[] x) {
      int N = x.length;
      Complex[] y = new Complex[N];

      // take conjugate
      for (int i = 0; i < N; i++) {
         y[i] = x[i].conjugate();
      }

      // compute forward FFT
      y = fft(y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        // take conjugate again
        for (int i = 0; i < N; i++) {
            y[i] = y[i].conjugate();
        }

        // divide by N
        for (int i = 0; i < N; i++) {
            y[i] = y[i].times(1.0 / N);
        }

        return y;

    }

    // compute the circular convolution of x
and y
    public static Complex[]
cconvolve(Complex[] x, Complex[] y) {

        // should probably pad x and y with
0s so that they have same length
        // and are powers of 2
        if (x.length != y.length) { throw new
RuntimeException("Dimensions don't agree");
}

        int N = x.length;

        // compute FFT of each sequence
        Complex[] a = fft(x);
        Complex[] b = fft(y);

        // point-wise multiply
        Complex[] c = new Complex[N];
        for (int i = 0; i < N; i++) {
            c[i] = a[i].times(b[i]);
        }

        // compute inverse FFT
        return ifft(c);
    }


    // compute the linear convolution of x
and y
    public static Complex[]
convolve(Complex[] x, Complex[] y) {
        Complex ZERO = new Complex(0, 0);

        Complex[] a = new
Complex[2*x.length];
        for (int i = 0;       i <   x.length; i++)
a[i] = x[i];
        for (int i = x.length; i < 2*x.length; i++)
a[i] = ZERO;

        Complex[] b = new
Complex[2*y.length];
        for (int i = 0;       i <   y.length; i++)
b[i] = y[i];
        for (int i = y.length; i < 2*y.length; i++)
b[i] = ZERO;

        return cconvolve(a, b);
    }

    // display an array of Complex numbers
to standard output
    public static void show(Complex[] x,
String title) {
        System.out.println(title);
        System.out.println("----------");
        for (int i = 0; i < x.length; i++) {
          System.out.println(x[i]);
        }
        System.out.println();
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/**************************************************
***********************
   * Test client and sample execution
   *
   * % java FFT 4
   * x
   * ------------------
   * -0.03480425839330703
   * 0.07910192950176387
   * 0.7233322451735928
   * 0.1659819820667019
   *
   * y = fft(x)
   * ------------------
   * 0.9336118983487516
   * -0.7581365035668999 +
0.08688005256493803i
   * 0.44344407521182005
   * -0.7581365035668999 -
0.08688005256493803i
   *
   * z = ifft(y)
   * ------------------
   * -0.03480425839330703
   * 0.07910192950176387 +
2.6599344570851287E-18i
   * 0.7233322451735928
   * 0.1659819820667019 -
2.6599344570851287E-18i
   *
   * c = cconvolve(x, x)
   * ------------------
   * 0.5506798633981853
   * 0.23461407150576394 -
4.033186818023279E-18i
   * -0.016542951108772352
   * 0.10288019294318276 +
4.033186818023279E-18i
   *
   * d = convolve(x, x)
   * ------------------
   * 0.001211336402308083 -
3.122502256758253E-17i
   * -0.005506167987577068 -
5.058885073636224E-17i
   * -0.044092969479563274 +
2.1934338938072244E-18i
   * 0.10288019294318276 -
3.6147323062478115E-17i
   * 0.5494685269958772 +
3.122502256758253E-17i
   * 0.240120239493341 +
4.655566391833896E-17i
   * 0.02755001837079092 -
2.1934338938072244E-18i
   * 4.01805098805014E-17i
   *
**************************************************
**********************/

   public static void main(String[] args) {
      int N = Integer.parseInt(args[0]);
      Complex[] x = new Complex[N];

      // original data
      for (int i = 0; i < N; i++) {
         x[i] = new Complex(i, 0);
         x[i] = new Complex(-
2*Math.random() + 1, 0);
      }
      show(x, "x");

      // FFT of original data
      Complex[] y = fft(x);
      show(y, "y = fft(x)");

      // take inverse FFT
      Complex[] z = ifft(y);
      show(z, "z = ifft(y)");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        // circular convolution of x with itself
        Complex[] c = cconvolve(x, x);
        show(c, "c = cconvolve(x, x)");

        // linear convolution of x with itself
        Complex[] d = convolve(x, x);
        show(d, "d = convolve(x, x)");
    }
}
```

คลาส Complex

```
package com.t109.tabletest;

public class Complex {
        public final double re;   // the real
part
        public final double im;   // the
imaginary part

        // create a new object with the
given real and imaginary parts
        public Complex(double real,
double imag) {
            re = real;
            im = imag;
        }

        // return a string representation
of the invoking Complex object
        public String toString() {
            if (im == 0) return re + "";
            if (re == 0) return im + "i";
            if (im <  0) return re + " - " + (-
im) + "i";
            return re + " + " + im + "i";
        }

        // return
abs/modulus/magnitude and
angle/phase/argument
        public double abs()   { return
Math.hypot(re, im); }  // Math.sqrt(re*re +
im*im)
        public double phase() { return
Math.atan2(im, re); }  // between -pi and pi

        // return a new Complex object
whose value is (this + b)
        public Complex plus(Complex b)
{
            Complex a = this;             //
invoking object
            double real = a.re + b.re;
            double imag = a.im + b.im;
            return new Complex(real,
imag);
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        // return a new Complex object
whose value is (this - b)
        public Complex minus(Complex
b) {
            Complex a = this;
            double real = a.re - b.re;
            double imag = a.im - b.im;
            return new Complex(real,
imag);
        }

        // return a new Complex object
whose value is (this * b)
        public Complex times(Complex
b) {
            Complex a = this;
            double real = a.re * b.re - a.im
* b.im;
            double imag = a.re * b.im +
a.im * b.re;
            return new Complex(real,
imag);
        }

        // scalar multiplication
        // return a new object whose
value is (this * alpha)
        public Complex times(double
alpha) {
            return new Complex(alpha *
re, alpha * im);
        }

        // return a new Complex object
whose value is the conjugate of this
        public Complex conjugate() {
return new Complex(re, -im); }

        // return a new Complex object
whose value is the reciprocal of this
        public Complex reciprocal() {
            double scale = re*re + im*im;
            return new Complex(re /
scale, -im / scale);
        }

        // return the real or imaginary
part
        public double re() { return re; }
        public double im() { return im; }

        // return a / b
        public Complex divides(Complex
b) {
            Complex a = this;
            return a.times(b.reciprocal());
        }

        // return a new Complex object
whose value is the complex exponential of
this
        public Complex exp() {
            return new
Complex(Math.exp(re) * Math.cos(im),
Math.exp(re) * Math.sin(im));
        }

        // return a new Complex object
whose value is the complex sine of this
        public Complex sin() {
            return new
Complex(Math.sin(re) * Math.cosh(im),
Math.cos(re) * Math.sinh(im));
        }

        // return a new Complex object
whose value is the complex cosine of this
        public Complex cos() {
            return new
Complex(Math.cos(re) * Math.cosh(im), -
Math.sin(re) * Math.sinh(im));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }

    // return a new Complex object
whose value is the complex tangent of this
    public Complex tan() {
        return sin().divides(cos());
    }



    // a static version of plus
    public static Complex
plus(Complex a, Complex b) {
        double real = a.re + b.re;
        double imag = a.im + b.im;
        Complex sum = new
Complex(real, imag);
        return sum;
    }



    // sample client for testing
    public static void main(String[]
args) {
        Complex a = new Complex(0,
1.0);
        Complex b = new Complex(-
3.0, 4.0);
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้