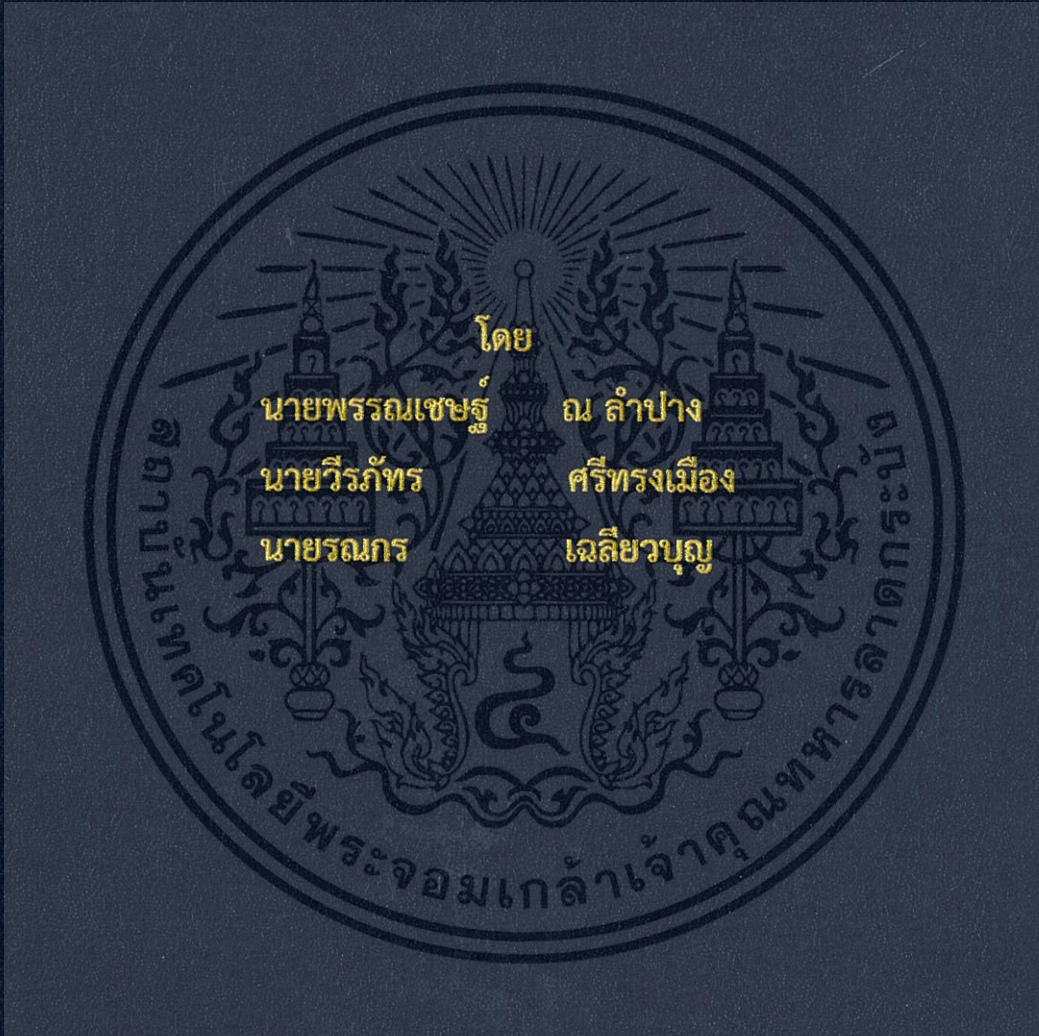


อุปกรณ์ควบคุมโทรทัศน์โดยอาศัยการประมวลผลภาพของท่าทางมือ
TV REMOTE CONTROL BASED ON HAND GESTURE IMAGE PROCESSING



ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

อุปกรณ์ควบคุมโทรทัศน์โดยอาศัยการประมวลผลภาพของท่าทางมือ

TV REMOTE CONTROL BASED ON HAND GESTURE IMAGE PROCESSING



T144342

โดย

นายพรรณเชษฐ์ ณ ลำปาง 55010824

นายวีรภัทร ศรีทรงเมือง 55011169

นายรณกร เฉลียวบุญ 55011014

เลขหมู่.....
เลขทะเบียน 144342
วันเดือน ปี 24 พ.ย. 2559

อาจารย์ที่ปรึกษา

รศ.ดร. ปราโมทย์ วาดเขียน

รศ.ดร. จีรสุดา โกษิยาภรณ์

b. 144342055b
i.

ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558

ผ่านการตรวจรูปเล่มแล้ว

(ปราโมทย์)

อาจารย์ที่ปรึกษา

7/6/2559

ผ่านการตรวจชิ้นงานแล้ว

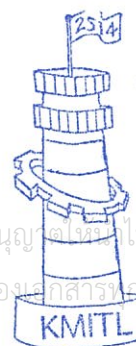
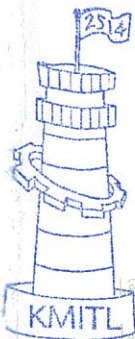
(จิรสุดา)

กรรมการผู้ตรวจชิ้นงาน

7/6/2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาติให้นำไปใช้ประโยชน์อื่นใด
วิศวกรรมโทรคมนาคม
Telecommunications Engineering

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
วิศวกรรมโทรคมนาคม
Telecommunications Engineering



ปริญญาโทปีการศึกษา 2558

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง อุปกรณ์ควบคุมโทรทัศน์โดยอาศัยการประมวลผลภาพของท่าทางมือ

TV REMOTE CONTROL BASED ON HAND GESTURE IMAGE PROCESSING

ผู้จัดทำ

1. นายพรณเชษฐ์ ณ ลำปาง 55010824

2. นายวีรภัทร ศรีทรงเมือง 55011169

3. นายรณกร เฉลียวบุญ 55011014



..... อาจารย์ที่ปรึกษา

รศ.ดร. ปราโมทย์ วาดเขียน



..... อาจารย์ที่ปรึกษาร่วม

รศ.ดร. จีรสุตา โกษิยามารณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์เรื่องระบบควบคุมการเปลี่ยนช่องโทรทัศน์โดยใช้การประมวลภาพเคลื่อนไหวของฝ่ามือฉบับนี้ได้สำเร็จลุล่วงไปได้ด้วยดีเกิดจากการสนับสนุนเกื้อกูลจาก รศ.ดร. ปราโมทย์ วาดเขียน อาจารย์ที่ปรึกษาที่กรุณาให้คำปรึกษาที่เป็นประโยชน์ตลอดจนแก้ไขข้อบกพร่องต่างๆด้วยความเอาใจใส่ดี และ รศ.ดร. จีรสุดา โกษิยาภรณ์ อาจารย์ที่ปรึกษาร่วมที่ช่วยให้คำแนะนำ คำปรึกษาที่เป็นประโยชน์ต่อปริญญานิพนธ์นี้ ผู้จัดทำตระหนักถึงความตั้งใจจริงและความทุ่มเทของอาจารย์และขอกราบขอบพระคุณไว้เป็นอย่างสูง ที่สำคัญที่สุดขอขอบคุณภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

อนึ่ง ผู้จัดทำหวังว่าปริญญานิพนธ์ฉบับนี้จะมีประโยชน์อยู่ไม่น้อยจึงขอมอบส่วนดีทั้งหมดนี้ให้แก่เหล่าคณาจารย์ที่ได้ประสิทธิประสาทวิชาจนทำให้ปริญญานิพนธ์เป็นประโยชน์ต่อผู้ที่เกี่ยวข้องและขอมอบความกตัญญูตเวทิตาคุณแต่บิดา มารดา และผู้มีพระคุณทุกท่าน สำหรับข้อบกพร่องต่างๆ ที่อาจจะเกิดขึ้นนั้นผู้จัดทำยินดีที่จะรับฟังคำแนะนำจากทุกท่านที่ได้เข้ามาศึกษาเพื่อเป็นประโยชน์ในการพัฒนาต่อไป หากมีข้อผิดพลาดประการใดผู้จัดทำขออภัยมา ณ ที่นี้

นายพรณเชษฐ์ ณ ลำปาง
นายวีรภัทร ศรีทรงเมือง
นายรณกร เฉลียวบุญ
ผู้จัดทำ

อุปกรณ์ควบคุมโทรทัศน์โดยอาศัยการประมวลผลภาพของท่าทางมือ
TV CHANNEL CONTROL BASED ON HAND GESTURE IMAGE PROCESSING

โดย นายพรณเชษฐ์ ฅลำปาง 55010824
นายวีรภัทร ศรีทรงเมือง 55011169
นายรณกร เฉลียวบุญ 55011014

อาจารย์ที่ปรึกษา รศ.ดร. ปราโมทย์ วาดเขียน
อาจารย์ที่ปรึกษาร่วม รศ.ดร. จีรสุดา โกษิยาวรณ

บทคัดย่อ

ปฏิญานพนธ์นี้นำเสนออุปกรณ์ควบคุมการเปลี่ยนช่องและปรับลดระดับเสียงของโทรทัศน์ โดยใช้ภาพท่าทางของมือที่ได้จากกล้องประมวลผลด้วยบอร์ดประมวลผลเพื่อทำการแปลงภาพท่าทางของมืองดกล่าวเป็นค่าที่บ่งชี้รูปแบบของการควบคุมซึ่งค่าที่ได้นั้นจะถูกส่งต่อไปยังไมโครคอนโทรลเลอร์เพื่อสร้างสัญญาณเสมือนรีโมท สัญญาณดังกล่าวจะถูกส่งผ่านหลอดอินฟราเรดเพื่อไปทำการควบคุมโทรทัศน์ให้เปลี่ยนช่องหรือปรับลดระดับเสียงตามสัญลักษณ์ทางภาษามือที่ได้รับมา อุปกรณ์นี้จึงช่วยเพิ่มความสะดวกสบายในการควบคุมโทรทัศน์มากยิ่งขึ้น

ABSTRACT

This thesis presents TV remote controller for control channel shifting and volume adjusting by using hand gesture. The image of hand gesture obtained from the camera will be processed by the processing board to translate into the controlling code. After the controlling code is detected, the microcontroller will generate the virtual remote control signal corresponding to the detected controlling code. This signal is then send through infrared LED to control the television. This device thus makes TV controlling more convenient

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า	
กิตติกรรมประกาศ	I	
บทคัดย่อ	II	
สารบัญ	III	
สารบัญรูป	V	
สารบัญตาราง	XII	
บทที่ 1	บทนำ	
1.1	ความเป็นมาและความสำคัญของปัญหา	1
1.2	วัตถุประสงค์	1
1.3	ขอบเขตของปริญญาานิพนธ์	1
บทที่ 2	ทฤษฎีและหลักการที่เกี่ยวข้อง	
2.1	การประมวลผลภาพ (Image processing)	2
2.2	ราสเบอร์รี่พาย (Raspberry Pi)	19
2.3	รีโมท (Remote control)	21
2.4	การส่งและการรับสัญญาณอินฟราเรด (Infrared)	22
2.5	ไดโอดเปล่งแสง (Light-emitting Diode)	25
2.6	ทฤษฎีเบื้องต้นของไอซีเบอร์ 555	27
บทที่ 3	การออกแบบและการจัดทำปริญญาานิพนธ์	
3.1	การออกแบบระบบ	30
3.2	เครื่องมือที่ใช้ในการทดลอง	43
3.3	การเก็บผลการทดลอง	43

สารบัญ (ต่อ)

	หน้า
บทที่ 4	ผลการทดลอง
4.1	ผลการทดลองการตรวจจับภาพฝ่ามือ 45
4.2	ผลการทดลองการสร้างสัญญาณรีโมทเสมือน 49
4.3	ผลการทดลองการควบคุมโทรศัพท์ด้วยลักษณะท่าทางมือ 61

บทที่ 5	สรุปผลและข้อเสนอแนะ
5.1	สรุปผล 64
5.2	ข้อเสนอแนะ 64

บรรณานุกรม	65
-------------------	----

ภาคผนวก



สารบัญรูป

รูปที่		หน้า
2.1	ตำแหน่งของพิกเซล [16]	2
2.2	การทำสำเนาภาพ และการควอนไทซ์ภาพ [17]	3
2.3	แบบจำลองสี RGB [17]	6
2.4	แบบจำลองสี HSV [17]	7
2.5	ตัวอย่างภาพและฮิสโตแกรมของภาพที่มี 256 กลุ่มสี [17]	8
2.6	ตัวอย่างภาพและฮิสโตแกรมของภาพที่มี 32 กลุ่มสี [17]	8
2.7	ตัวอย่างภาพที่มีฮิสโตแกรมเหมือนกัน [17]	10
2.8	ส่วนของการขยาย [17]	13
2.9	ส่วนของการกร่อนขนาด [17]	13
2.10	แนวความคิดการสร้างสีเหลี่ยมในการปิดล้อมวัตถุ [15]	14
2.11	แบบจุด 4 จุด เชื่อมกันและแบบจุด 8 จุด เชื่อมกัน [17]	15
2.12	อักษรตำแหน่งจุดภาพ [17]	16
2.13	หมายเลขของแต่ละจุดภาพตามชั้นตอนที่ 1 [17]	16
2.14	กลุ่มรวมที่หมายเลขเทียบเท่ากัน [17]	17
2.15	หมายเลขของแต่ละจุดภาพตามชั้นที่ 3 [17]	17
2.16	ตัวอย่างจุดภาพและตำแหน่ง [17]	17
2.17	ภาพการทำ Opening (ก) ภาพต้นฉบับ (ข) ผลลัพธ์จากการทำ Opening [17]	18
2.18	ภาพการทำ Closing (ก) ภาพต้นฉบับ (ข) ผลลัพธ์จากการทำ Closing [17]	18
2.19	ส่วนประกอบของราสเบอร์รี่พาย [20]	19
2.20	การจัดเรียงขาของพอร์ตอินพุตเอาต์พุตเนกประสงค์ [20]	21
2.21	Electromagnetic Spectrum [2]	23
2.22	ข้อมูลบิตซิมซุงแบบมาตรฐาน [7]	24
2.23	ข้อมูลเฟรมซิมซุงแบบมาตรฐาน [7]	24
2.24	รูปแสดงลักษณะของหลอด LED [2]	26
2.25	วงจรและส่วนประกอบของไอซีเบอร์ 555 [2]	27
2.26	แสดงวงจร Astable multivibrator [8]	28

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.1	30
3.2	31
3.3	32
3.4	33
3.5	34
3.6	35
3.7	36
3.8	37
3.9	38
3.10	39
3.11	40
3.12	42
3.13	42
4.1	45
4.2	46
4.3	46
4.4	47
4.5	47
4.6	48
4.7	49
4.8	50
4.9	50
4.10	51
4.11	52

สารบัญรูป (ต่อ)

รูปที่	หน้า	
4.12	สัญญาณพัลส์ของปั๊มหมายเลข 2	52
4.13	สัญญาณพัลส์ของปั๊มหมายเลข 3	52
4.14	สัญญาณพัลส์ของปั๊มหมายเลข 4	53
4.15	สัญญาณพัลส์ของปั๊มหมายเลข 5	53
4.16	สัญญาณพัลส์ที่สร้างจากวงจรสร้างสัญญาณพัลส์โดยใช้ไอซีเบอร์ 555	54
4.17	สัญญาณรีโมทเสมือนของปั๊มเพิ่มช่องของรีโมทโทรทัศน์ยี่ห้อซัมซุง	55
4.18	สัญญาณรีโมทเสมือนของปั๊มลดช่องของรีโมทโทรทัศน์ยี่ห้อซัมซุง	55
4.19	สัญญาณรีโมทเสมือนของปั๊มเพิ่มเสียงของรีโมทโทรทัศน์ยี่ห้อซัมซุง	55
4.20	สัญญาณรีโมทเสมือนของปั๊มลดเสียงของรีโมทโทรทัศน์ยี่ห้อซัมซุง	56
4.21	สัญญาณรีโมทเสมือนของปั๊มหมายเลข 1 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง	56
4.22	สัญญาณรีโมทเสมือนของปั๊มหมายเลข 2 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง	56
4.23	สัญญาณรีโมทเสมือนของปั๊มหมายเลข 3 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง	57
4.24	สัญญาณรีโมทเสมือนของปั๊มหมายเลข 4 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง	57
4.25	สัญญาณรีโมทเสมือนของปั๊มหมายเลข 5 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง	57
4.26	การเปรียบเทียบสัญญาณปั๊มเพิ่มช่องโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)	58
4.27	การเปรียบเทียบสัญญาณปั๊มลดช่องโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)	58
4.28	การเปรียบเทียบสัญญาณปั๊มเพิ่มเสียงโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)	59
4.29	การเปรียบเทียบสัญญาณปั๊มลดเสียงโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)	59
4.30	การเปรียบเทียบสัญญาณปั๊มหมายเลข 1 ของโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)	59
4.31	การเปรียบเทียบสัญญาณปั๊มหมายเลข 2 ของโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)	60
4.32	การเปรียบเทียบสัญญาณปั๊มหมายเลข 3 ของโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)	60

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.33	การเปรียบเทียบสัญญาณปุ่มหมายเลข 4 ของโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)	60
4.34	การเปรียบเทียบสัญญาณปุ่มหมายเลข 5 ของโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)	61
4.35	การเปลี่ยนช่องโทรทัศน์ไปช่องหมายเลข 1 เมื่อทำท่าทางของฝ่ามือแสดงหมายเลข 1	61
4.36	การเปลี่ยนช่องโทรทัศน์ไปช่องหมายเลข 2 เมื่อทำท่าทางของฝ่ามือแสดงหมายเลข 2	62
4.37	การเปลี่ยนช่องโทรทัศน์ไปช่องหมายเลข 3 เมื่อทำท่าทางของฝ่ามือแสดงหมายเลข 3	62
4.38	การเปลี่ยนช่องโทรทัศน์ไปช่องหมายเลข 4 เมื่อทำท่าทางของฝ่ามือแสดงหมายเลข 4	63
4.39	การเปลี่ยนช่องโทรทัศน์ไปช่องหมายเลข 5 เมื่อทำท่าทางของฝ่ามือแสดงหมายเลข 5	63

สารบัญตาราง

ตารางที่	หน้า	
2.1	ส่วนประกอบต่างๆที่สำคัญของราสเบอร์รี่พาย	20
2.2	ช่วงสีพลังงานของไดโอดเปล่งแสง	25



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในอดีตจนถึงปัจจุบันโทรทัศน์นั้นมีบทบาทเกี่ยวข้องกับชีวิตผู้คนแทบทุกครัวเรือน ซึ่งได้รับการพัฒนาอย่างต่อเนื่องที่ผ่านมา เช่น การพัฒนาจากจอภาพขาว-ดำเป็นจอสี การเพิ่มความสามารถเชื่อมต่ออินเทอร์เน็ตได้ มีระบบสั่งการด้วยเสียง เป็นต้น ดังนั้นปัญญาประดิษฐ์จึงได้นำเสนออุปกรณ์ควบคุมการเปลี่ยนช่องและปรับลดระดับเสียงของโทรทัศน์ โดยใช้ภาพท่าทางของมือที่ได้จากกล้องมาประมวลผลด้วยบอร์ดประมวลผลเพื่อทำการแปลภาพท่าทางของมืองด่งกล่าวเป็นค่าที่บ่งชี้รูปแบบของการควบคุมซึ่งค่าที่ได้นั้นจะถูกส่งต่อไปยังไมโครคอนโทรลเลอร์เพื่อสร้างสัญญาณเสมือนรีโมทสัญญาณดังกล่าวจะถูกส่งผ่านหลอดอินฟราเรด เพื่อไปทำการควบคุมโทรทัศน์ให้เปลี่ยนช่องหรือปรับลดระดับเสียงตามสัญลักษณ์ทางภาษามือที่ได้รับมา แทนการใช้รีโมทกดเปลี่ยนช่อง ทั้งนี้เพื่อเพิ่มความสะดวกสบายให้กับผู้ใช้งาน

1.2 วัตถุประสงค์

1. เพื่อนำเทคโนโลยีการประมวลผลภาพไปประยุกต์ใช้ในการตรวจจับการเคลื่อนไหวของฝ่ามือ เพื่อนำมาควบคุมการเปลี่ยนช่องของโทรทัศน์
2. เพื่อศึกษาระบบประมวลผลภาพ
3. เพื่อศึกษาสัญญาณคำสั่งของรีโมทโทรทัศน์และการสร้างสัญญาณเลียนแบบเพื่อใช้ในการควบคุมโทรทัศน์

1.3 ขอบเขตของปัญญาประดิษฐ์

1. สามารถควบคุมโทรทัศน์ได้ในระยะ 1-2 เมตร
2. สามารถสั่งการเพื่อควบคุมโทรทัศน์โดยการตรวจจับภาพท่าทางของมือได้ 9 รูปแบบ คือ เพิ่มเสียง, ลดเสียง, เพิ่มช่อง, ลดช่อง และเปลี่ยนช่องโทรทัศน์ตั้งแต่ช่อง 1 – 5 ได้ด้วยลักษณะท่ามือในรูปแบบต่างๆ ซึ่งรายละเอียดของรูปแบบคำสั่งจะแสดงในปัญญาประดิษฐ์
3. สามารถสร้างสัญญาณเสมือนรีโมทเพื่อควบคุมการเปลี่ยนช่องและปรับระดับเสียงของโทรทัศน์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

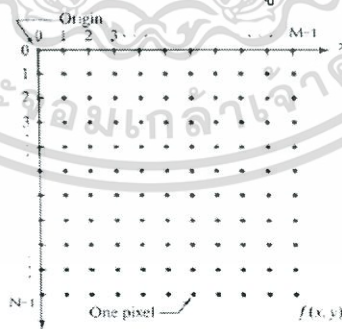
2.1 การประมวลผลภาพ (Image processing)

การประมวลผลภาพ คือ การประยุกต์ใช้งานการประมวลผลสัญญาณบนสัญญาณ 2 มิติ เช่น ภาพนิ่ง ภาพถ่าย วิดีโอ และยังรวมถึงสัญญาณ 2 มิติอื่นๆ ที่ไม่ใช่ภาพ ซึ่งในอดีตการประมวลผลภาพนั้น จะอยู่ในรูปของการประมวลผลสัญญาณแอนาล็อก โดยใช้อุปกรณ์ปรับแต่งแสง ซึ่งวิธีเหล่านั้นยังมีใช้เป็นส่วนสำคัญสำหรับการประยุกต์ใช้งานบางอย่าง เช่น ฮอโลกราฟี (Holography) แต่เนื่องจากอุปกรณ์คอมพิวเตอร์ในปัจจุบัน ราคาถูกลง และมีความเร็วเพิ่มขึ้นมาก การประมวลผลภาพดิจิทัลจึงได้รับความนิยมมากกว่าการประมวลผลภาพรูปแบบเดิม เพราะสามารถทำการประมวลผลภาพที่ซับซ้อนได้

2.1.1 ภาพและความหมายของพิกเซล

2.1.1.1 พิกเซล (Pixel)

พิกเซล (Pixel) คือ ความเข้มแสงที่รวมกันทำให้เกิดเป็นภาพ ภาพหนึ่ง ๆ จะประกอบด้วยพิกเซลมากมาย ซึ่งภาพแต่ละภาพที่สร้างขึ้นจะมีความหนาแน่นของพิกเซลเหล่านี้แตกต่างกันไป ความหนาแน่นนี้เป็นตัวบอกถึงความละเอียด (Resolution) ของภาพซึ่งมีหน่วยเป็น ppi (Pixel Per Inch) คือจำนวนพิกเซลต่อนิ้ว ซึ่งโดยทั่วไปถือว่าภาพที่มีความละเอียดสูงหรือคุณภาพดีจะมีความละเอียด 300 x 300 ppi ขึ้นไป ค่า ppi ยิ่งสูงขึ้น ภาพก็จะมี ความละเอียดและคมชัดมากขึ้น ซึ่งจะแสดงตำแหน่งของพิกเซล ดังรูปที่ 2.1



รูปที่ 2.1 ตำแหน่งของพิกเซล [16]

N = จำนวนพิกเซลที่มากที่สุดในแกน y

M = จำนวนพิกเซลที่มากที่สุดในแกน x

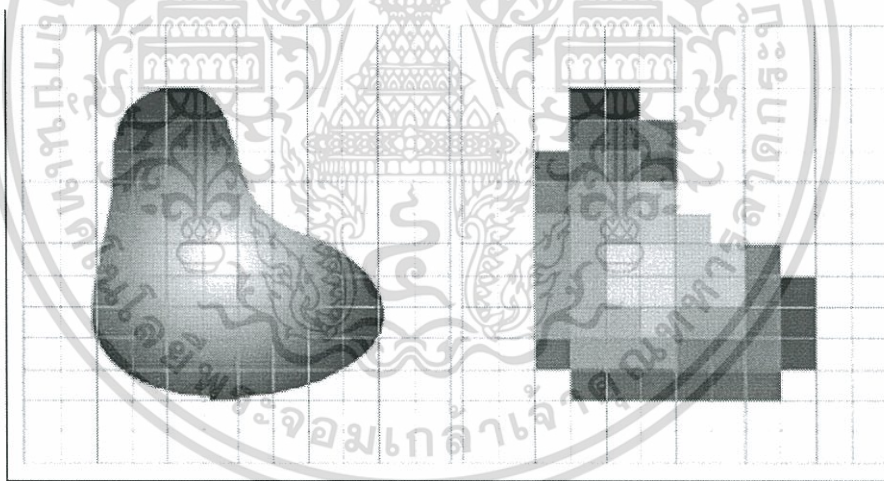
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 การประมวลผลภาพเชิงตัวเลข (Digital Image Processing)

2.1.2.1 การแทนภาพด้วยภาพแบบดิจิทัลภาพแบบดิจิทัล (Digital Image)

เป็นภาพที่ถูกแปลงมาจากแอนาล็อก ให้อยู่ในรูปของตัวเลขโดยภาพแอนาล็อกถูกแบ่งเป็นพื้นที่สี่เหลี่ยมเล็ก ๆ ที่เรียกว่าพิกเซล ในแต่ละพิกเซลจะถูกระบุตำแหน่งด้วยคู่อพิกเซล x, y และค่าระดับความเข้มของแสงของพิกเซลนั้นๆ โดยเราสามารถแปลงภาพเป็นแบบดิจิทัล โดยมีขั้นตอนและวิธีการดังนี้

เมื่อเรานำสัญญาณแอนาล็อกที่ต้องการประมวลผลผ่านส่วนที่เรียกว่าดิจิไทเซอร์ (Digitizer) ซึ่งจะทำหน้าที่ในการแปลงสัญญาณแอนาล็อกให้เป็นสัญญาณดิจิทัล จากนั้นทำการควอนไทซ์ (Quantizing) เพื่อแปลงค่าความเข้มของแสงให้เป็นตัวเลข พังก์ชันของภาพ $f(x, y)$ จะถูกทำให้เป็นสัญญาณไม่ต่อเนื่องทั้งระนาบของภาพซึ่งเรียกว่า การสุ่มภาพ (Image Sampling) ของฟังก์ชันที่ได้เรียกว่า การควอนไทซ์ ระดับความเข้มของแสง (Greasy Level Quantization) ก็จะได้ข้อมูลที่เป็นดิจิทัลดังรูปที่ 2.2



รูปที่ 2.2 การทำสุ่มภาพ และการควอนไทซ์ภาพ [17]

2.1.2.2 ลักษณะการจัดเก็บข้อมูลภาพแบบดิจิทัล

โดยทั่วไปแล้วภาพจะมีความเข้มตั้งแต่ 2 ระดับขึ้นไป แต่ที่นิยมใช้กันมาก คือ ค่าระดับความเข้มของพิกเซลที่เท่ากับ 256 ระดับ ซึ่งจะทำให้ค่าของพิกเซลอยู่ในช่วง (0-255) โดยใช้เนื้อที่ในการเก็บข้อมูลขนาด 1 ไบต์หรือ 8 บิต สำหรับข้อมูล 1 พิกเซล ในกรณีที่ต้องการภาพที่มีความเข้มสูงอาจจะต้องการจำนวนบิตสำหรับการเก็บข้อมูลมากกว่า 8 บิต คืออาจจะเป็น 16 หรือ 24 บิตโดยจะแยกความแตกต่างของภาพแต่ละประเภทให้เห็นอย่างชัดเจนได้ดังนี้

- ภาพ 2 ระดับ คือ มีพิกเซลสีขาวกับสีดำเท่านั้น โดยแต่ละพิกเซลจะมีขนาดของข้อมูลเท่ากับ 1 บิต

- ภาพ 16 ระดับ คือ ในแต่ละพิกเซลจะมีขนาดของข้อมูล 4 บิต ซึ่งสามารถแสดงภาพได้ความเข้มถึง 16 ระดับ

- ภาพ 256 ระดับ คือ ในแต่ละพิกเซลจะมีขนาดของข้อมูล 8 บิต ซึ่งทำให้สามารถแสดงภาพได้ความเข้มถึง 256 ระดับ

- ภาพทิวทัศน์ (True Color) คือในแต่ละพิกเซลจะมีขนาดของข้อมูล 24 บิต ทำให้สามารถแสดงภาพออกมาได้เฉพาะภาพสีเท่านั้น ไม่สามารถแสดงเป็นภาพขาวดำได้

โดยทั่วไปวิธีการประมวลผลภาพเชิงตัวเลขที่ทำให้คอมพิวเตอร์สามารถรู้วัตถุในภาพได้นั้น แบ่งออกได้เป็นสองระดับด้วยกันคือ การประมวลผลภาพในระดับต่ำ (Low-Level Image) และการประมวลผลภาพในระดับสูง (High-Level Image Processing) การประมวลผลภาพในระดับต่ำจะเป็นการประมวลผลเชิงตัวเลขเกือบทั้งหมด เพื่อหาตัวแปรต่าง ๆ มาอธิบายข้อมูลภาพ โดยมีจุดประสงค์ที่จะนำตัวแปรเหล่านั้นไปใช้ในการประมวลผลภาพระดับสูงต่อไปโดยทั่วไปแล้วการประมวลผลภาพระดับต่ำจะประกอบด้วย การประมวลผลภาพก่อน (Preprocessing) เช่น การกำจัดสัญญาณรบกวน หรือการทำให้ภาพคมชัด การหาขอบภาพ เป็นต้น

การประมวลผลระดับสูงเป็นการนำผลลัพธ์ หรือสัญลักษณ์ที่ได้จากการประมวลผลภาพระดับต่ำมาตีความหรือประมวลผลเพื่อให้คอมพิวเตอร์สามารถรู้จักและเข้าใจภาพได้ สำหรับความแตกต่างของการประมวลผลภาพ ทั้ง 2 ประเภทนั้นคือ การประมวลผลภาพระดับต่ำจะใช้ค่าความสว่างของจุดภาพ (พิกเซล) ส่วนการประมวลผลภาพระดับสูงนั้นข้อมูลภาพที่นำมาประมวลผลจะถูกแสดงในรูปสัญลักษณ์ ซึ่งสัญลักษณ์เหล่านี้จะแสดงถึงสิ่งต่าง ๆ ที่อยู่ในภาพเช่น ขนาดของวัตถุ รูปร่าง และความสัมพันธ์กันระหว่างวัตถุภาพ

2.1.3 การสร้างภาพไบนารี

การสร้างภาพไบนารีสามารถทำได้โดยใช้เทคนิคการทำเทรชโฮลด์ (Thresholding Technique) โดยการพิจารณาว่าพิกเซลนั้นๆจะเป็นสีขาวหรือสีดำ จะกระทำได้โดยใช้การเปรียบเทียบระหว่างพิกเซลของภาพกับค่าคงที่หนึ่งเรียกว่า ค่าเทรชโฮลด์ (Threshold Value) เทคนิคนี้ใช้กันมากในกรณีที่ข้อมูลภาพมีลักษณะที่ต่างกันระหว่างวัตถุ (Object) และพื้นหลัง (Background) โดยค่าของพิกเซลใด ๆ ของภาพที่มีค่ามากกว่าหรือเท่ากับค่าเทรชโฮลด์จะถูกเปลี่ยนเป็น 0 (ดำ) และ พิกเซลของภาพที่มีค่าน้อยกว่าค่าเทรชโฮลด์จะถูกเปลี่ยนเป็น 1 (ขาว) เป็นต้น

ในการสร้างภาพไบนารีโดยใช้เทคนิคเทรชโฮลด์เพื่อให้ได้ผลลัพธ์ที่เหมาะสมและคมชัด สิ่งที่สำคัญที่สุดคือ ค่าเทรชโฮลด์ เนื่องจากถ้าเลือกค่าเทรชโฮลด์ที่ไม่เหมาะสม (ค่าเทรช

โพลต์ที่มีค่าน้อยเกินไปหรือมากเกินไป) ภาพที่ได้อาจจะสว่างน้อยเกินไป หรือสว่างมากเกินไป หรือภาพที่ได้มีสิ่งรบกวน (Noise) เกิดขึ้น อันเป็นผลทำให้ภาพที่ได้ไม่ดีเท่าที่ควร ดังนั้นปัญหาของการสร้างภาพไบนารี ซึ่งมีวิธีคำนวณหาค่าเทรชโพลต์โดยกำหนดค่าล่วงหน้า (Pre-assigned Threshold Value) การหาค่าเทรชโพลต์จากค่ากลาง (Mid-Range Threshold Value) ซึ่งแต่ละวิธีอธิบายได้ดังนี้

- การหาค่าเทรชโพลต์โดยกำหนดค่าล่วงหน้า เป็นการกำหนดค่าเทรชโพลต์โดยการกำหนดเองจากผู้ใช้งาน ซึ่งการกำหนดนี้จะขึ้นอยู่กับประสบการณ์ของผู้ใช้คนนั้นๆ โดยการเลือกค่าคงที่ค่าหนึ่ง ซึ่งเรียกว่า ค่าเทรชโพลต์ โดยค่าที่เลือกมานี้จะเป็นค่าที่อยู่ระหว่างค่าต่ำและค่าสูงสุดของระดับความเข้มแสงของภาพ เช่นภาพอินพุทมีระดับความเข้มแสง 256 ระดับ จะมีค่าได้ตั้งแต่ 0-255

- การหาค่าเทรชโพลต์จากค่ากลาง เป็นการหาเทรชโพลต์ที่แตกต่างจากการหาค่าเทรชโพลต์วิธีแรก สำหรับวิธีนี้จะเป็นการคำนวณหาค่าเทรชโพลต์โดยอัตโนมัติโดยไม่ต้องให้ผู้ใช้เป็นผู้กำหนดโดยการหาค่าเทรชโพลต์นี้ใช้วิธีทางสถิติในเรื่องการหาค่ากลางหรือค่าเฉลี่ย (Mean) ค่าเทรชโพลต์ที่คำนวณได้จากค่ากึ่งกลางที่อยู่ระหว่างค่าระดับความเข้มสูงสุด (Maximum Level) และระดับความเข้มต่ำสุด (Minimum Level) ของภาพ

2.1.4 แบบจำลองสี (Color Model)

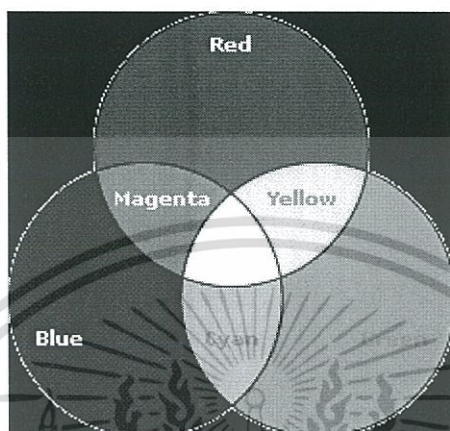
แบบจำลองสี (Color Model) เป็นสิ่งที่ใช้อ้างอิงถึงสีต่าง ๆ สำหรับคอมพิวเตอร์แล้วจะไม่ใช้แบบจำลองที่เป็น Analytical Model เหมือนกับที่ใช้ในทางวิทยาศาสตร์ ซึ่งใช้วิธีการวัดซึ่งอยู่ในรูปของพลังงานช่วงของสเปกตรัม (Spectrum) แต่จะเป็น Empirical Model ที่ได้รับสัมพันธ์ของค่าที่ใช้อ้างอิงกับสีใด ๆ จากการทดลองที่เป็นการศึกษาแบบ Psychophysical ที่มีการรับรู้ของมนุษย์เข้ามาเกี่ยวข้อง

แบบจำลองสีมีหลายแบบด้วยกัน เช่น แบบจำลองสี RGB แบบจำลองสี CMY แบบจำลองสี CMYK แบบจำลองสี HSV แบบจำลองสี HIS แบบจำลองสี HLS แบบจำลองสี YIQ และแบบจำลองสี YUV แบบจำลองสี YcbCR เป็นต้น

2.1.4.1 แบบจำลองสี RGB (RGB Color Model)

แบบจำลองสี RGB (RGB Color Model) เป็นแบบจำลองสีที่เฉพาะเจาะจงกับจอภาพคอมพิวเตอร์ เนื่องจาก RGB Model ได้ทำการสร้างสีต่าง ๆ ขึ้นโดยใช้แหล่งกำเนิดแสดงจำนวนสามสี ได้แก่ สีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) ที่เกิดจากการเรียงแสงที่มีคุณสมบัติที่ต่างกันอย่างลำดับ ซึ่งแสงทั้งสามสีจะมีค่าไม่เท่ากันในแต่ละอุปกรณ์ นอกเสียจากรวมคุณสมบัติของสารเรืองแสงและการตั้งค่าจอภาพ และสภาพแวดล้อมที่จอภาพคอมพิวเตอร์

เหมือนกันทุกประการ ซึ่งโดยปกติแล้วจะมีค่าที่แตกต่างกันออกไปตามรูปที่ 2.3 แบบจำลองสี RGB ประกอบด้วยข้อมูลจำนวนสามส่วนคือ ค่า Intensity ของสีทั้งสามซึ่งได้แก่สีแดง สีเขียว และสีน้ำเงิน



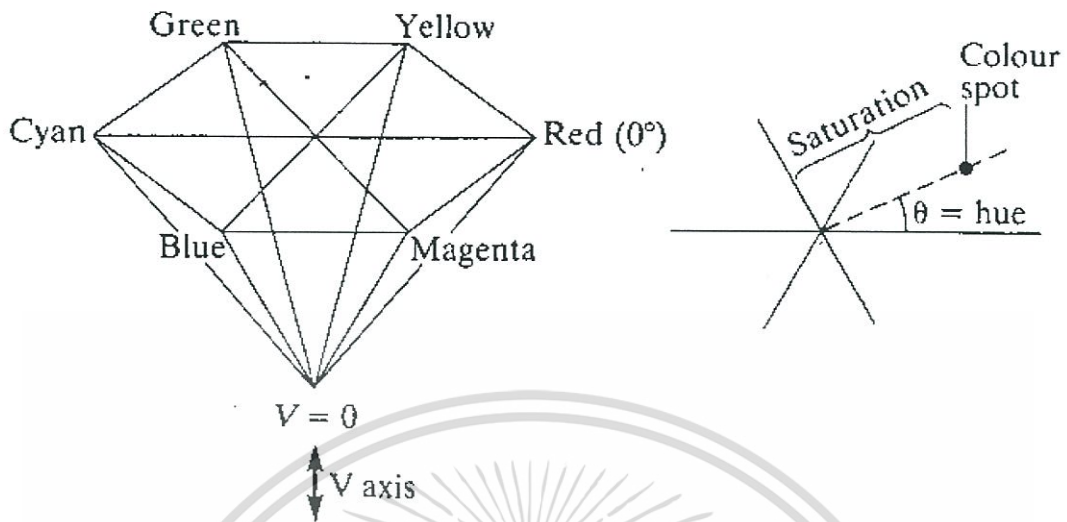
รูปที่ 2.3 แบบจำลองสี RGB [17]

2.1.4.2 แบบจำลองสี HSV (HSV Color Model)

แบบจำลองสี HSV (HSV Color Model) เป็นแบบจำลองสีที่ถูกสร้างขึ้นเพื่อเป็นทางเลือกโดยแบบจำลองสี HSV แสดงในรูปที่ 2.4 จะให้ความหมายที่ดีกว่าเมื่อกกล่าวถึง สีต่าง ๆ ในเชิงศิลปะ เช่น เมื่อพูดถึงสีเหลืองในทางศิลปะจะมีความแตกต่างกัน เมื่อพิจารณาสีเหลืองอ่อน สีเหลืองแก่ หรือสีน้ำตาลว่ามีความแตกต่างกันอย่างไร จะพบว่าทุกสี คือสีเหลืองนั่นเอง ที่มีระดับความเข้มหรือมีความอิ่มตัวที่แตกต่างกัน ดังนั้นสีในแบบจำลองสี HSV จึงให้ความรู้สึกที่เข้าใจได้มากกว่าสำหรับมนุษย์ซึ่งจำลองสี HSV ประกอบด้วยสามส่วนคือ

H หมายถึง Hue หรือสีที่มีค่าที่แตกต่างออกไปตามความถี่ของแสง เช่น แดง เหลือง เขียว น้ำเงิน หรือ ม่วง เป็นต้น

S หมายถึง Saturation หรือความอิ่มตัวของ Hue นั้น ๆ เช่น สีแดง และสีชมพู คือ สีแดงเพียงแต่สีชมพูมีความอิ่มตัวน้อยกว่า แบบจำลองสี HSV หมายถึง Value หรือค่าความสว่างของสี โดยที่ค่า Value ต่ำสุดหมายถึง สีดำ ไม่ว่าจะ Hue หรือ Saturation เท่าใด และค่า Value สูงสุดหมายถึง สีขาว ซึ่งเป็นสีที่สว่างที่สุดของ Hue และ Saturation นั้นๆ เช่น Hue ใดๆ มีค่า Saturation มีค่าเท่ากับ 0 เมื่อ Value สูงสุดคือสีเหลืองและ Value ต่ำสุดคือสีดำ



รูปที่ 2.4 แบบจำลองสี HSV [17]

2.1.5 การแยกลักษณะเฉพาะของภาพ (Image Feature Extraction)

การแยกลักษณะเฉพาะของภาพเป็นการแยกหรือสกัดเอาข้อมูลที่สำคัญของภาพออกมาซึ่งลักษณะเฉพาะของภาพเป็นคุณสมบัติที่สามารถหาได้โดยใช้ขั้นตอนวิธีการประมวลผลภาพ โดยที่ลักษณะเฉพาะพื้นฐานของภาพประกอบด้วย 3 ส่วนคือ สี, รูปร่างและพื้นผิว

- สี (Color) เป็นลักษณะเฉพาะของภาพที่มีบทบาทสำคัญในระบบค้นคืนภาพ เช่น ฮิสโตแกรมสีซึ่งเป็นลักษณะเฉพาะของสีที่ถูกนำมาใช้บ่อยๆ เนื่องจากสีเป็นสิ่งที่สามารถมองเห็นได้ง่ายและเป็นสิ่งแรกที่สามารถสังเกตเห็นได้จากการมองภาพ นอกจากนี้สียังสามารถใช้ในการแยกแยะกลุ่มของภาพออกตามเนื้อหาได้เป็นอย่างดี เช่น สีฟ้าของน้ำทะเล สีแดงของดอกไม้ สีเขียวของต้นไม้ เป็นต้น

- รูปร่าง (Shape) เป็นลักษณะเฉพาะของภาพที่ใช้อธิบายถึงรูปร่างและลักษณะรวมถึงขนาดของวัตถุภายในภาพซึ่งทำให้สามารถแยกวัตถุออกจากพื้นหลังหรือแยกแยะระหว่างวัตถุที่มีรูปร่างแตกต่างกันออกจากกันได้

- พื้นผิว (Texture) เป็นลักษณะเฉพาะที่ใช้อธิบายความหยาบความละเอียดหรือความซับซ้อนของวัตถุภายในภาพซึ่งแต่ละภาพอาจจะประกอบด้วยวัตถุที่มีลักษณะพื้นผิวที่ต่างกันออกไป การวิเคราะห์พื้นผิวจะช่วยให้สามารถแยกแยะความแตกต่างของวัตถุได้ดียิ่งขึ้น การค้นคืนภาพที่ใช้พื้นผิวเป็นลักษณะเฉพาะของภาพส่วนใหญ่จะถูกนำไปใช้ในการค้นหาภาพจากกลุ่มภาพพื้นผิว เช่น ชุดภาพพื้นผิวของหิน ชุดภาพพื้นผิวของใบไม้ เป็นต้น

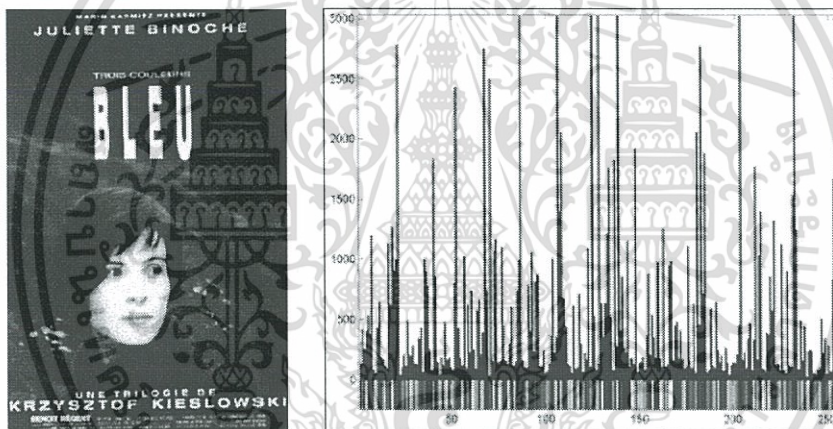
2.1.6 การสร้างดัชนีภาพ (Image Indexing)

การสร้างดัชนีภาพมีด้วยกันหลากหลายวิธีซึ่งในที่นี้ขอยกตัวอย่างวิธีการนิยมและได้การยอมรับมา 4 วิธี คือ

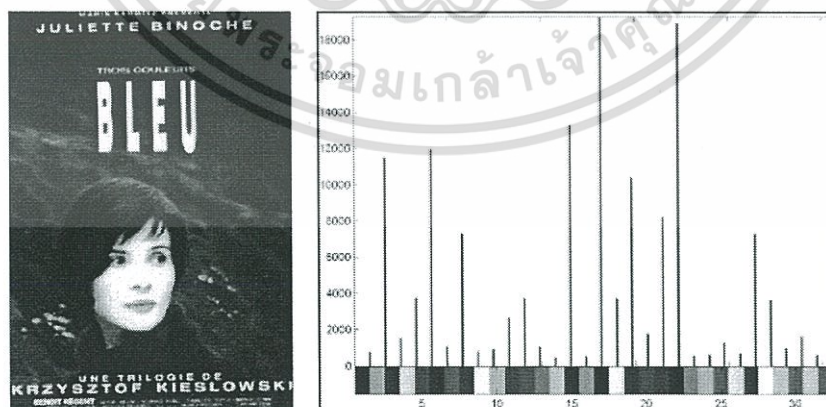
2.1.6.1 ฮิสโตแกรมสี

ฮิสโตแกรมสีเป็นกราฟแสดงความถี่หรือจำนวนจุดภาพที่ระดับค่าสีแต่ละค่าโดยแกนนอนแทนระดับค่าสีต่าง ๆ ส่วนแกนตั้งแทนจำนวนจุดภาพที่ระดับค่าสีนั้น ๆ เป็นลักษณะเฉพาะของภาพในลักษณะของการวัดการแจกแจงของสีในภาพซึ่งสามารถพิจารณาได้จากภาพตัวอย่างในรูปที่ 2.5 และ 2.6

สำหรับภาพสี RGB ขนาด 24 บิตต่อจุดภาพแต่ละ Channel คือ แดง เขียว และ น้ำเงิน นั้นจะมีค่าความสว่าง 8 บิตหรือ 256 ระดับโดยมีค่าตั้งแต่ 0-255 ดังนั้นจุดภาพแต่ละจุดจึงสามารถแสดงสีได้มากถึง 16.7 ล้านสี



รูปที่ 2.5 ตัวอย่างภาพและฮิสโตแกรมของภาพที่มี 256 กลุ่มสี [17]



รูปที่ 2.6 ตัวอย่างภาพและฮิสโตแกรมของภาพที่มี 32 กลุ่มสี [17]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการคำนวณค่าฮิสโตแกรมสีภาพแต่ละภาพจะถูกควอนไทซ์สีภายในภาพเพื่อลดมิติของเวกเตอร์และลดความซับซ้อนในการคำนวณลงโดยการแบ่งกลุ่มสีออกเป็น m ถังสี (Bins) ซึ่งส่วนใหญ่นิยมใช้ 32, 64 หรือ 256 ถังสี เนื่องจากการแยกแยะความแตกต่างของระดับค่าสีของสายตามนุษย์มีความละเอียดไม่มากนัก กำหนดให้ภาพ I มีขนาด $n_1 \times n_2$ จุดภาพและ H_{ci} แทนจำนวนจุดภาพที่มีสี c_i ของภาพ I ดังนั้นสามารถคำนวณฮิสโตแกรมสีได้ดังสมการที่ 2.1

$$h_{ci}(I) = \frac{H_{ci}}{n_1 \times n_2} \quad (2.1)$$

โดยที่ $h_{ci}(I)$ คือฮิสโตแกรมของสี c_i ของภาพ

M คือจำนวนสีภายในภาพหลังการควอนไทซ์สี

n_1 คือความกว้างของภาพ

n_2 คือความยาวของภาพ

การหารด้วย $n_1 \times n_2$ หรือจำนวนจุดภาพทั้งหมดภายในภาพมีวัตถุประสงค์เพื่อให้ค่าฮิสโตแกรมสีเป็นบรรทัดฐานและเพื่อให้สามารถนำฮิสโตแกรมสีของภาพซึ่งมีขนาดแตกต่างกันมาเปรียบเทียบกันได้

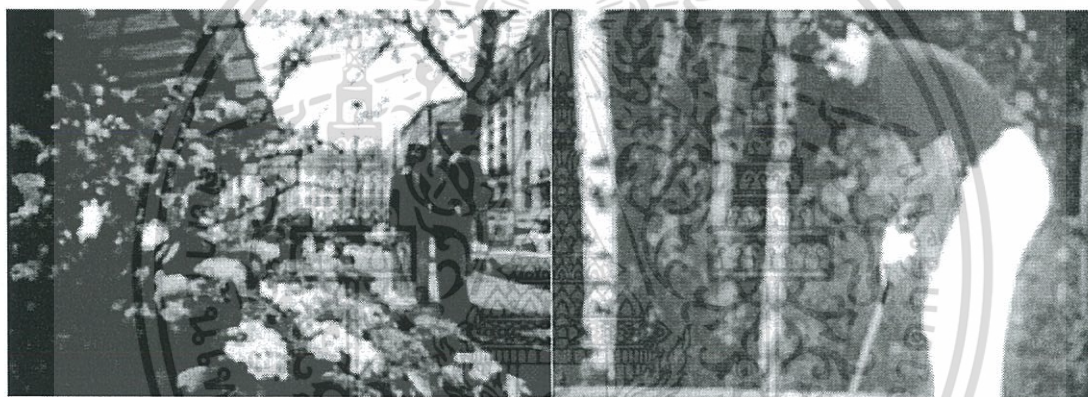
ดังนั้นจะได้เวกเตอร์แทนฮิสโตแกรมสีของภาพ I ดังสมการที่ 2.2

$$H(I) = (h_{c_1}, h_{c_2}, \dots, h_{c_m}) \quad (2.2)$$

ฮิสโตแกรมสีเป็นลักษณะเฉพาะทางสีของภาพที่นิยมนำมาใช้ในระบบการค้นคืนภาพส่วนใหญ่เนื่องจากสามารถคำนวณได้ง่ายและรวดเร็วแต่มีข้อเสียคือมีเฉพาะข้อมูลเกี่ยวกับการกระจายของสีภายในภาพเท่านั้นไม่มีข้อมูลเชิงตำแหน่ง (Spatial Information)

2.1.6.2 Color Coherence Vector (CCV)

การกำหนดการเชื่อมติดกันของสี (Color Coherence) ในลักษณะที่เป็นระดับซึ่งจุดภาพของสีนั้น ๆ เป็นสมาชิกของบริเวณขนาดใหญ่ที่มีสีเหมือนกันเรียกบริเวณสำคัญนี้ว่าบริเวณที่เชื่อมติดกัน (Coherent Regions) มีความสำคัญในการแสดงลักษณะเฉพาะของภาพตัวอย่างเช่น ภาพ 2 ภาพดังแสดงในรูปที่ 2.7 มีค่าฮิสโตแกรมของสีเหมือนกันทั้ง ๆ ที่ภาพทั้งสองมีลักษณะที่แตกต่างกันจะเห็นว่ามิสีแดงปรากฏอยู่ทั้ง 2 ภาพในปริมาณที่ใกล้เคียงกันภาพทางด้านซ้ายมีจุดภาพที่เป็นสีแดง (จากดอกไม้) กระจายอยู่อย่างกว้างๆ แต่ในขณะที่ภาพทางด้านขวามีจุดภาพที่เป็นสีแดง (จากเสื้อของนักกอล์ฟ) รวมกลุ่มอยู่ด้วยกันจากปัญหาดังกล่าว Pass และ Zabih [17] จึงได้คิดค้นลักษณะเฉพาะของภาพรูปแบบใหม่ขึ้นมาคือ Color Coherence Vector หรือ CCV



รูปที่ 2.7 ตัวอย่างภาพที่มีฮิสโตแกรมเหมือนกัน [17]

CCV คือฮิสโตแกรมสีที่มีการปรับปรุงโดยการเพิ่มเติมข้อมูลเชิงตำแหน่งเข้ามา หลักการของ CCV ก็คือการแยกกลุ่มจุดภาพออกเป็น 2 กลุ่มคือจุดภาพที่เชื่อมติดกัน (Coherent Pixel) และจุดภาพที่ไม่เชื่อมติดกัน (Incoherent Pixel) โดยจุดภาพใด ๆ จะเป็นจุดภาพที่เชื่อมติดกันก็ต่อเมื่อองค์ประกอบที่เชื่อมติดกัน (Connected Component) ของจุดภาพนั้นมีค่ามากกว่าค่าคงที่ที่กำหนดไว้ซึ่งในงานวิจัยของ Pass และ Zabih กำหนดให้มีค่าโดยประมาณเท่ากับ 1% ของจำนวนจุดภาพทั้งหมดภายในภาพส่วนจุดภาพที่เหลือก็จะเป็นจุดภาพที่ไม่เชื่อมติดกัน CCV เป็นการนำเสนอการแยกกลุ่มสำหรับแต่ละสีในภาพเพื่อกำจัดจุดภาพที่ไม่ได้เป็นองค์ประกอบสำคัญของภาพทิ้งไปโดยที่ CCV สามารถแยกจุดภาพที่เชื่อมติดกันออกจากจุดภาพที่ไม่เชื่อมติดกันและป้องกันการจับคู่ที่ผิดพลาดซึ่งสิ่งนี้เป็นลักษณะเด่นที่ฮิสโตแกรมสีไม่สามารถทำได้

CCV ของภาพใดๆคือฮิสโตแกรมสีของจุดภาพในกลุ่มจุดภาพที่เชื่อมติดกันของภาพนั้นกำหนดให้ $H_{ccv}(I)$ คือเวกเตอร์ขนาด m มิติแทน CCV ของภาพ I ซึ่งคำนวณได้จากสมการที่ 2.3

$$H_{ccv}(I) = \langle (\alpha_1, \beta_2), (\alpha_2, \beta_2), \dots, (\alpha_m, \beta_m) \rangle \quad (2.3)$$

ข้อดีของ CCV เมื่อเปรียบเทียบกับฮิสโตแกรมสีคือมีประสิทธิภาพในการค้นคืนภาพมากกว่าเนื่องจากการกำจัดจุดภาพที่ไม่ได้เป็นองค์ประกอบสำคัญของภาพทิ้งไปแต่จะใช้เวลาในการประมวลผลมากกว่าฮิสโตแกรมสี

2.1.7 การจับคู่ (Matching)

กระบวนการจับคู่ภาพที่เหมือนกัน (Matching) เป็นเหมือนการสื่อสารกันระหว่างภาพ และขั้นตอนการพัฒนาการจับคู่รูปภาพนั้นขึ้นอยู่กับขั้นตอนการ Localization ถ้าหากผู้ใช้แสดงให้เห็นว่าการทำ Localization นั้นไม่ถูกต้อง การเรียนรู้จะเกิดขึ้นโดยการใช้คุณสมบัติหรือลักษณะเด่นของภาพทั้งหมดที่มีอยู่สำหรับทำ Localization และกำหนดตำแหน่งโดยผู้ใช้ การเรียนรู้การจับคู่ภาพเป็นกระบวนการเรียนรู้ที่มีการสืบทอด คือ การสร้างคลังคำศัพท์ (Dictionary) และการรวบรวมข้อมูลสำหรับการจำแนกจะเพิ่มมากขึ้น คือ จำนวนของคำที่ได้จากการเรียนรู้จะมีเพิ่มมากขึ้นเพียงเวลาแค่ไม่นาน

การสร้าง Dictionary จะขึ้นอยู่กับเพิ่มของ Nearest Neighbor Classifier สำหรับคำหรือคุณสมบัติใหม่ที่ไม่อยู่ใน Dictionary ถ้าถูกนำเข้ามาในระบบ แล้วมีการจับคู่ภาพและพบว่ามีส่วนที่คล้ายกับข้อมูลอื่นที่อยู่ใน Dictionary ในขั้นตอนนี้หากระยะห่างระหว่างคำกับลักษณะเด่นของภาพอยู่ต่ำกว่า ค่าเกณฑ์ที่กำหนด จะทำให้มีการรู้จำคำนั้นเกิดขึ้น คือระบบสามารถระบุได้ว่าภาพที่มีการจับคู่เป็นภาพที่หือหรือรุ่นไหน แต่ถ้าค่าระยะห่างสูงกว่าเกณฑ์ที่กำหนด คำหรือชิ้นส่วนของภาพนั้น จะถูกเพิ่มเข้าไปเป็นคุณสมบัติและกำหนดเป็นค่าเริ่มต้นใหม่ของข้อมูลหรือ Dictionary และนำไปใช้ในการ Matching ภาพในครั้งต่อไปที่มีภาพแบบนี้เข้ามา

การหาจุดเด่นและการ Matching ภาพนั้นจะต้องใช้เวลานานขึ้นมาก และอาจมีประสิทธิภาพที่ไม่ดี ดังนั้นภาพที่จะนำมาหาจุดเด่นจึงต้องมีการปรับขนาดซึ่งอาจผ่านการแยกแยะรูปร่างมาก่อนแล้ว จะถูกนำมาแปลงขนาดให้เท่ากับขนาดของ Template ที่เก็บไว้ในฐานข้อมูล

หลังจากนั้นจึงทำการหมุนภาพให้มี Orientation ที่สอดคล้องกับภาพใน Template โดยอ้างอิงจากรูปร่างของรถยนต์เป็นหลัก

2.1.8 SIFT (Scale Invariant Feature Transform)

Scale Invariant Feature Transform หรือ SIFT เป็นอัลกอริทึมหนึ่งในคอมพิวเตอร์วิชันสำหรับคำนวณหาจุดสนใจ (Key point) ในรูปภาพหนึ่ง ๆ และคำนวณหาคุณลักษณะของ Key point หนึ่งๆ ที่หาได้ SIFT คือการเอาจุดเด่นในรูปที่ไม่ขึ้นอยู่กับการกำหนดทิศทาง ตำแหน่ง มุมการมอง แสงสว่าง เงา ซึ่งจะทำให้สามารถนำมาใช้ในการเปรียบเทียบจุดเด่นในรูปอื่น ๆ ได้ง่ายและถูกต้องแม่นยำมากยิ่งขึ้น อัลกอริทึมนี้ถูกคิดค้นโดย ศร.เดวิด โวลส์ (David Lowe) แห่ง University British Columbia ในปี 1999 ภาประยุกต์ใช้งาน SIFT มีหลากหลาย ดังเช่น การรู้จำวัตถุ (Object recognition) การสร้างแผนที่สำหรับนำทางหุ่นยนต์ (Robotic mapping and navigation, image stitching), การสร้างโมเดล 3 มิติของวัตถุหรือทัศนียภาพ (3D modeling of object and scenes) โดยทั่วไปแล้ว Key point จะหมายถึงจุดพิกเซลในภาพที่มีการเปลี่ยนแปลงเชิงสองมิติ (Two -Dimensional) ของระดับความสว่าง (Pixel Intensity) รอบ ๆ Key point นั้น ๆ

2.1.9 ส่วนประกอบของอัลกอริทึมในการหา SIFT Key Point ในภาพหนึ่ง

ส่วนของการตรวจหา Key Point ออกมาจากภาพอินพุต (Key Point Detection) ซึ่งในส่วนนี้จะได้ชุดของพิกัด x, y ของจุดที่จะเป็น Key Point ซึ่งจะใช้ในการให้คำอธิบายของ Key Point นั้น ๆ ในขั้นตอนถัดไป

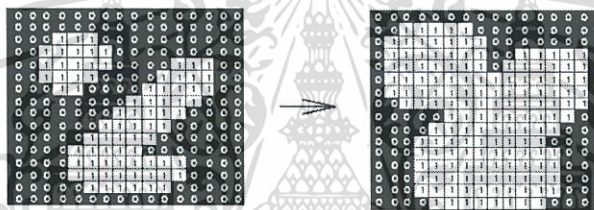
ส่วนของการให้คำอธิบายแก่ Key Point หนึ่งๆ (Key Point Description) ในส่วนนี้อัลกอริทึมจะคำนวณหาเวกเตอร์อธิบาย (Descriptor Vector) ซึ่งคำนวณมาจากค่าความสว่างของพิกเซลในอาณาบริเวณรอบๆ Key Point เวกเตอร์อธิบายเหล่านี้จะใช้ในการเปรียบเทียบเพื่อวัตถุประสงค์ของการรู้จำสัญลักษณ์ (Identity) เมื่อมันไปปรากฏอยู่ในภาพอื่น

2.1.10 การเปลี่ยนแปลงลักษณะรูปร่างหรือโครงร่างของภาพ (Morphological processing)

การเปลี่ยนแปลงลักษณะรูปร่างหรือโครงร่างของภาพ คือการประมวลผลของภาพทางด้านโครงร่างเกี่ยวกับการแยกส่วนประกอบของภาพออกเพื่อใช้ในการแสดงรูปร่าง ในเมทริกซ์จะประกอบไปด้วยค่าระดับขาว-ดำ 2 ค่า คือ 0 และ 1

2.1.10.1 การขยาย (Dilation)

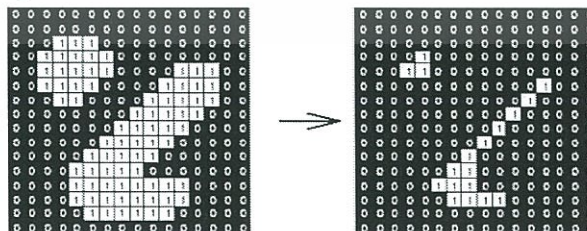
การขยาย (Dilation) จะพิจารณาข้อมูลภาพซึ่งเป็นภาพขาว-ดำ เป็นการขยายภาพให้ใหญ่ขึ้น เพื่อเพิ่มสีให้กับวัตถุที่แสดงผลในขั้นตอนสุดท้าย ซึ่งการขยายวัตถุจะทำได้โดยการกำหนดส่วนประกอบโครงร่าง (Structuring Element) และนำส่วนประกอบโครงร่างไปกราดบนข้อมูลภาพตามลำดับตลอดทั้งภาพ โดยเมื่อจุดเริ่มต้นของส่วนประกอบโครงร่างหรือจุดกำเนิดตรงกับตำแหน่งข้อมูลภาพที่เท่ากับ 1 จะทำการยูเนียนส่วนประกอบโครงร่าง เข้ากับข้อมูลภาพดังรูปที่ 2.8



รูปที่ 2.8 ส่วนของการขยาย [17]

2.1.10.2 การกร่อนขนาด (Erosion)

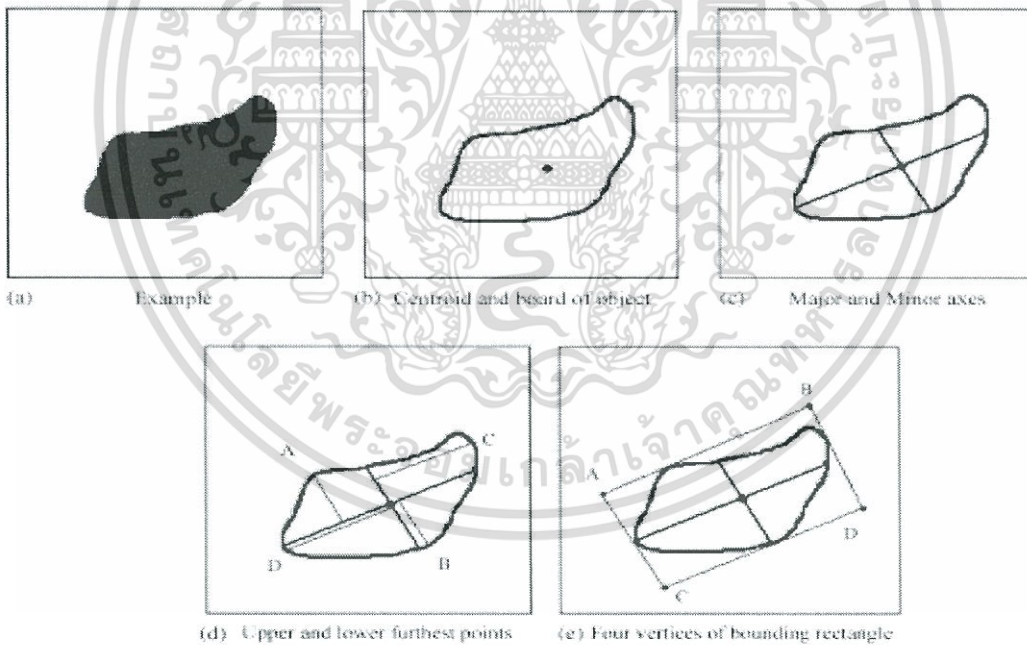
การกร่อนขนาด (Erosion) เป็นการกร่อนขนาดบริเวณขอบของวัตถุ ซึ่งการกร่อนมีวิธีคล้ายกับการขยายคือ สร้างส่วนประกอบโครงร่างขึ้นมาแล้วนำไปกระจายตามข้อมูลภาพ โดยจะเลื่อนไปทุกตำแหน่งเปรียบเทียบกับข้อมูลภาพ ถ้าข้อมูลมีค่าเหมือนกับส่วนประกอบโครงร่างจะทำการกำหนดค่าข้อมูลภาพที่ตรงกับตำแหน่งที่ตรงกับจุดเริ่มต้นหรือ จุดกำเนิดของส่วนประกอบโครงร่างให้เท่ากับ 1 ดังรูปที่ 2.9



รูปที่ 2.9 ส่วนของการกร่อนขนาด [17]

2.1.11 สี่เหลี่ยมในการปิดล้อมวัตถุ (Bounding Box)

สี่เหลี่ยมในการปิดล้อมวัตถุ (Bounding Box) คือกล่องสี่เหลี่ยมสำหรับปิดล้อมรอบบริเวณนั้น ๆ หรือบริเวณที่ถูกกำหนดหมายเลข ซึ่งกล่องสี่เหลี่ยมนี้จะช่วยในการคำนวณจุดศูนย์กลาง พื้นที่ ความยาวแกนเอก-แกนโท จุดพิกัด ๆ ของบริเวณหรือวัตถุในภาพที่สนใจ โดยจากรูปที่ 2.10 จะแสดงแนวความคิดการสร้างกรอบภาพ โดยรูป 2.10(a) คือภาพวัตถุสีขาว-ดำ ส่วนในรูปที่ 2.10(b) คือขอบและจุดศูนย์กลาง (Centroid) ของวัตถุ ต่อจากนั้นจะมีการหาแกนเอกและแกนโทของวัตถุดังรูปที่ 2.10(c) จากนั้นจะมีการหาจุดที่มีระยะห่างจากจุดศูนย์กลางวัตถุถึงขอบวัตถุที่ยาวที่สุดโดยพิจารณาจากส่วนด้านบนของแกนเอกและส่วนล่างของแกนเอก (A และ B) รวมทั้งการหาจุดที่มีระยะห่างจากจุดศูนย์กลางวัตถุถึงขอบวัตถุที่ยาวที่สุดโดยพิจารณาจากบริเวณส่วนซ้ายของแกนโทและส่วนขวาของแกนโท (C และ D) สุดท้ายก็จะได้จุดพิกัดในการสร้างกล่องสี่เหลี่ยมในการปิดล้อมวัตถุ (Minimum bounding box) นั้น

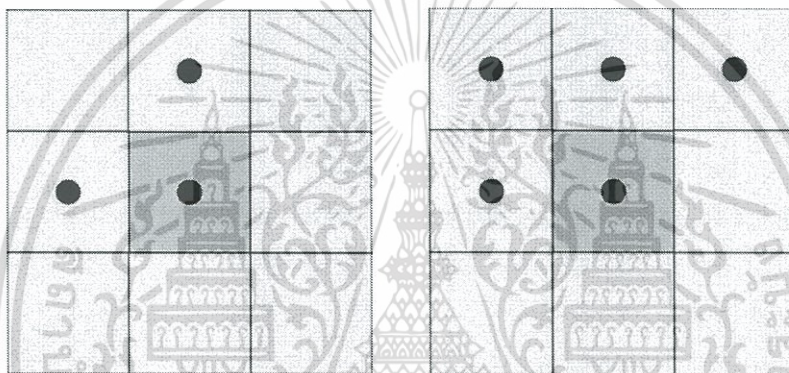


รูปที่ 2.10 แนวความคิดการสร้างสี่เหลี่ยมในการปิดล้อมวัตถุ [15]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.12 การกำหนดหมายเลขให้ส่วนที่เชื่อมกัน (Connected-component labeling)

การวิเคราะห์ส่วนประกอบหรือบริเวณที่มีการเชื่อมติดกันและใช้ในการแยกบริเวณต่าง ๆ ออกจากกันซึ่งไม่เหมือนกับวิธีตัดแยก โดยวิธีการจะกำหนดหมายเลขให้ส่วนที่เชื่อมกัน (Connected-component labeling) ซึ่งเป็นวิธีที่ใช้ในคอมพิวเตอร์วิทัศน์ (Computer vision) เพื่อ ตรวจสอบบริเวณที่เชื่อมติดกันของภาพขาวดำ ภาพสี แต่โดยทั่วไปใช้ในภาพขาวดำที่ผ่านการประมวลผลมาแล้ว อาจใช้ในการนับจำนวน การกรองและการติดตาม ดังรูปที่ 2.11



รูปที่ 2.11 แบบจุด 4 จุด เชื่อมกันและแบบจุด 8 จุด เชื่อมกัน [17]

จากรูปที่ 2.11 จุดที่อยู่รอบ ๆ จุดที่อยู่ตำแหน่งตรงกลางคือ เพื่อนบ้านมีทั้งแบบจุด 4 จุด เชื่อมกันและแบบจุด 8 จุด เชื่อมกันโดยเพื่อนบ้านที่เชื่อมกันก็คือขอบนั่นเอง การกำหนดหมายเลขมี 2 แบบคือแบบจุด 4 จุด เชื่อมกันและแบบจุด 8 จุด เชื่อมกัน มีข้อแตกต่างกันคือ ถ้าเป็นแบบจุด 4 จุด เชื่อมกันจะเลือกจุดที่เชื่อมกันเฉพาะบน ล่าง ซ้าย ขวา แต่หากเป็น แบบจุด 8 จุด เชื่อมกันจะเลือกรวมจุดที่อยู่มุมทแยงจากตำแหน่งกลางอีก 4 จุดด้วย ดังรูปที่ 2.12 กำหนดให้ p (pixel) แทนจุดภาพที่กำลังพิจารณา u แทนจุดภาพที่อยู่ตำแหน่งเหนือจุด p l แทนจุดภาพที่อยู่ตำแหน่งซ้ายของจุด p จะเริ่มทำจากซ้ายไปขวาและบนลงล่าง จากนั้นทำการกำหนดเลขหมายตามขั้นตอนดังนี้

ขั้นที่ 1. จากรูปที่ 2.12 ถ้าจุด p ไม่ใช่จุดภาพให้เลื่อนจุดในตำแหน่งถัดไป ถ้า p เป็นจุดภาพให้ตรวจสอบสถานะของ u และ l ถ้าไม่มีจุดใดเป็นจุดภาพให้ กำหนดหมายเลขใหม่ให้กับจุด p ถ้ามี 1 จุดเป็นจุดภาพให้นำหมายเลขของ จุดนั้นมากำหนดให้จุด p แต่ถ้ามีมากกว่า 1 จุด

เป็นจุดภาพสามารถหาหมายเลขของจุดใดก็ได้มากำหนดให้จุด p โดยถือว่าทุกหมายเลขเทียบเท่ากัน

	u	
l	p	

รูปที่ 2.12 อักษรตำแหน่งจุดภาพ [17]

ขั้นที่ 2. เมื่อสิ้นสุดการสแกนในขั้นที่ 1 จุดภาพทุก ๆ จุดจะมีหมายเลขกำหนดดังรูปที่ 2.13 แต่บางหมายเลขจะเทียบเท่ากัน ให้รวมกลุ่มหมายเลขที่เทียบเท่ากันดังรูปที่ 2.14 จากนั้นกำหนดหมายเลขให้แต่ละกลุ่ม

ขั้นที่ 3. นำหมายเลขของแต่ละกลุ่มจากขั้นที่ 2 ไปแทนหมายเลขของจุดภาพที่อยู่ในกลุ่มเดียวกัน ดังรูปที่ 2.15 ดังนั้นเราสามารถแบ่งตำแหน่งของแต่ละจุดที่เชื่อมกันออกเป็น 4 กลุ่มได้ดังรูปที่ 2.16

จากผลลัพธ์ที่ได้จากรูปที่ 2.13 จะเห็นว่าในแต่ละบริเวณจะมีหมายเลขกำกับไม่ซ้ำกัน ดังนั้นหมายเลขสูงสุดก็คือจำนวนบริเวณทั้งหมดที่เชื่อมกันนั่นเอง

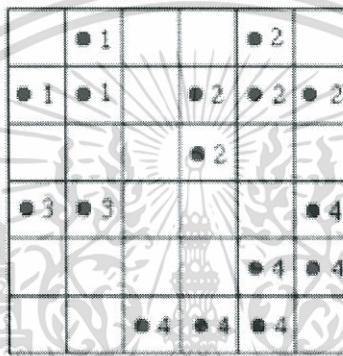
	1		2	
3	1		4	2
			4	
5	5			6
			7	6
		8	8	7

รูปที่ 2.13 หมายเลขของแต่ละจุดภาพตามขั้นตอนที่ 1 [17]

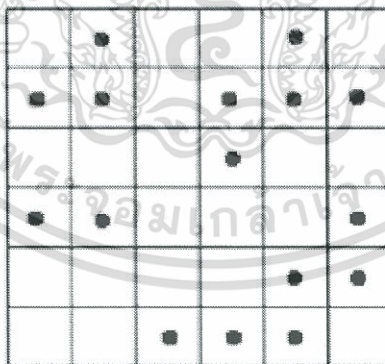
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Set ID	Equivalent Labels
1	1,3
2	2,4
3	5
4	6,7,8

รูปที่ 2.14 กลุ่มร่วมที่หมายเลขเทียบเท่ากัน [17]



รูปที่ 2.15 หมายเลขของแต่ละจุดภาพตามชั้นที่ 3 [17]

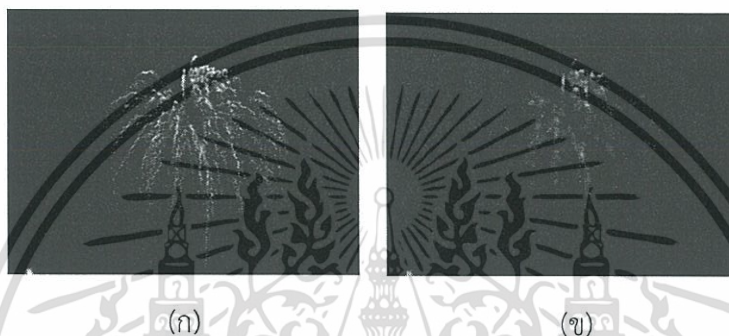


รูปที่ 2.16 ตัวอย่างจุดภาพและตำแหน่ง [17]

2.1.13 Opening and Closing

2.1.13.1 Opening

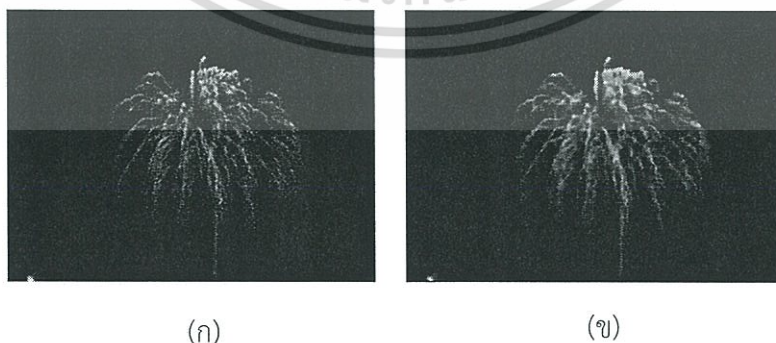
ใช้เพื่อกำจัดรายละเอียดขนาดเล็กของภาพ และการทำ Opening จะทำให้พิกเซลของภาพจะถูกเปิดกว้างมากขึ้นและวิธีการของ Opening คือการทำ Erosion ก่อนจากนั้นจึงทำ Dilation ดังสมการ $A \circ B = (A \ominus B) \oplus B$ เรียก B ว่าเป็น Structuring element ซึ่งจะแสดงตัวอย่างการทำ Opening เปรียบเทียบภาพต้นฉบับกับผลลัพธ์ของภาพหลังจากการทำ Opening ดังรูปที่ 2.17



รูปที่ 2.17 ภาพการทำ Opening (ก) ภาพต้นฉบับ (ข) ผลลัพธ์จากการทำ Opening [17]

2.1.13.2 Closing

เป็นวิธีตรงข้ามกับ Opening จะเป็นการทำให้ภาพมีการเชื่อมต่อกันมากขึ้นและ การทำ Closing จะทำให้พิกเซลของภาพจะถูกปิดเชื่อมต่อกันมากขึ้น วิธีการทำ Closing คือการทำ Dilation ก่อนจึงทำ Erosion ดังสมการ $A \bullet B = (A \oplus B) \ominus B$ เรียก B ว่าเป็น Structuring element ซึ่งจะแสดงตัวอย่างการทำ Closing เปรียบเทียบภาพต้นฉบับกับผลลัพธ์ของภาพหลังจากการทำ Closing ดังรูปที่ 2.18



รูปที่ 2.18 ภาพการทำ Closing (ก) ภาพต้นฉบับ (ข) ผลลัพธ์จากการทำ Closing [17]

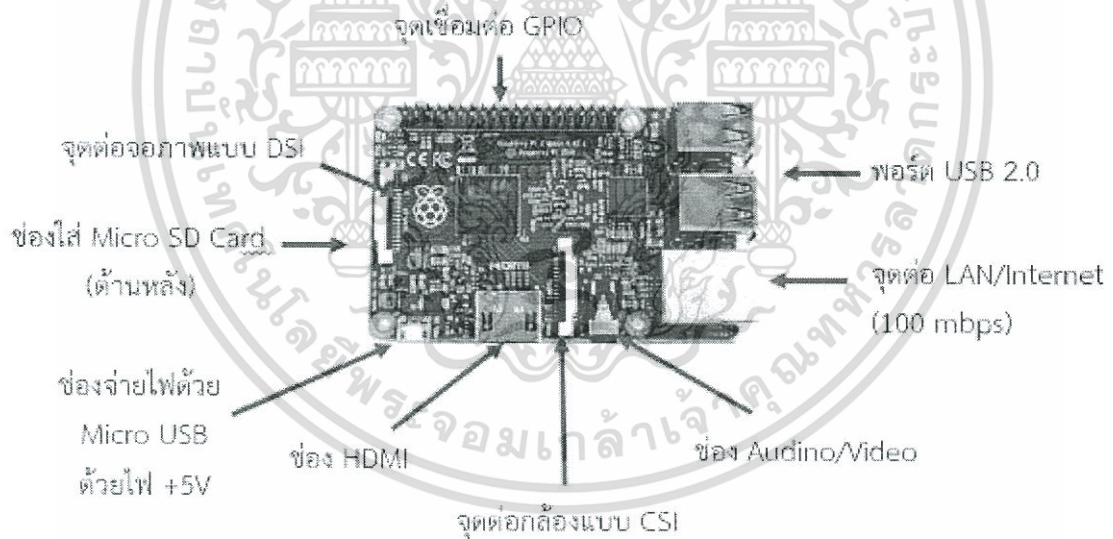
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ราชเบอร์รี่พาย (Raspberry Pi)

ราชเบอร์รี่พาย คือ คอมพิวเตอร์ขนาดจิ๋ว เป็นคอมพิวเตอร์ในบอร์ดเดียว (Single Board Computer) ถูกพัฒนาขึ้นในประเทศสหรัฐอเมริกา โดย ดร.อีเบน อัปตัน (Eben Upton) ภายใต้มูลนิธิราชเบอร์รี่พาย (Raspberry Pi Foundation) แต่เดิมนั้นราชเบอร์รี่พายได้ถูกพัฒนาขึ้นมาให้มีราคาต่ำ เพื่อใช้สำหรับการเรียนของนักศึกษา แต่ด้วยที่มีขนาดเล็ก สามารถพกพาได้สะดวก เนื่องจากเมื่อต่อกับจอคอมพิวเตอร์ เมาส์ และคีย์บอร์ด ก็จะเปรียบเสมือนคอมพิวเตอร์ที่ใช้อยู่ทั่วไป จึงทำให้ปัจจุบันได้รับความนิยมอย่างสูงในการนำมาฝึกพัฒนาโปรแกรมเบื้องต้น อีกทั้งสามารถใช้งานร่วมกับระบบอินเทอร์เน็ตได้อีกด้วย

2.2.1 โครงสร้างของราชเบอร์รี่พาย

ปัจจุบันราชเบอร์รี่พาย ได้ถูกผลิตออกมาเป็น 3 โมเดล คือโมเดล A, โมเดล B และ Pi 2 โมเดล B แต่ด้วยว่าประสิทธิภาพของโมเดล A นั้นน้อยกว่าโมเดล B และ Pi 2 Model B จึงทำให้ได้รับความนิยมน้อยกว่า ซึ่งปัญหานี้จะกล่าวถึงส่วนประกอบของโมเดล Pi 2 เป็นหลัก แสดงได้ดังตารางที่ 2.1 และแสดงส่วนประกอบของราชเบอร์รี่พาย ดังรูปที่ 2.19



รูปที่ 2.19 ส่วนประกอบของราชเบอร์รี่พาย [20]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 ส่วนประกอบต่างๆที่สำคัญของราสเบอร์รี่พาย

ส่วนประกอบหลัก	รายละเอียด
หน่วยประมวลผลกลาง (Central Processing Unit :CPU)	Broadcom BCM2836 900 MHz, Quad-core
กราฟฟิก (GPU)	Broadcom VideoCore IV
ระบบปฏิบัติการ (OS)	Windows/Linux
แรม(RAM)	1 GB
ระบบวิดีโอ (Video)	สามารถเชื่อมต่อระบบภาพวิดีโอผ่านทางพอร์ต HDMI, Composite RCA
ระบบเสียง (Audio)	สามารถเชื่อมต่อระบบเสียงผ่านทางพอร์ต 3.5 mm jack หรือ ผ่านทางพอร์ต HDMI
พอร์ตยูเอสบี 2.0 (USB 2.0 Port)	มี 4 พอร์ต ใช้ในการเชื่อมต่ออุปกรณ์ภายนอก เช่น คีย์บอร์ด เมาส์ เป็นต้น
ส่วนเชื่อมต่อเน็ตเวิร์ค (Networking)	ผ่านทางพอร์ต USB LAN โดยจะใช้มาตรฐาน RJ45
ความจุข้อมูล (Storage)	จะใช้การเก็บข้อมูลลงบนการ์ด ได้แก่ Micro Sd Card
ส่วนเชื่อมต่อไฟเลี้ยงวงจร (Power Source)	ใช้ไฟกระแสตรงที่ระดับแรงดัน 5 โวลต์ โดยผ่านทางพอร์ตไมโครยูเอสบี (Micro USB)
ขนาด	8.6 x 5.4 x 1.7 cm.

2.2.2 พอร์ตอินพุตเอาต์พุตเนกประสงค์ (General Purpose Input Output : GPIO)

คุณสมบัติที่โดดเด่นอีกอย่างของราสเบอร์รี่พายก็คือ ความยืดหยุ่นในการเชื่อมต่อกับอุปกรณ์ภายนอก โดยเฉพาะอย่างยิ่งกับอุปกรณ์อิเล็กทรอนิกส์ได้อย่างสะดวก โดยราสเบอร์รี่พายมีพอร์ตอินพุตเอาต์พุตเนกประสงค์ สำหรับ Pi 2 Model B นั้นจะมีจำนวนทั้งสิ้น 26 ขา โดยประกอบด้วยขาพอร์ตอินพุตเอาต์พุตดิจิทัล 17 ขา, ขาเชื่อมต่อบัสไอส์แควร์ซี และ เอสพีไอ จึงทำให้ราสเบอร์รี่พายสามารถเชื่อมต่ออุปกรณ์อิเล็กทรอนิกส์ได้อย่างหลากหลาย ซึ่งการจัดเรียงขาของพอร์ตอินพุตเอาต์พุตเนกประสงค์แสดงได้ดังรูปที่ 2.20

Pin#	NAME	Connection	Connection	NAME	Pin#
01	3.3V		5V (Cupcade)	5V	02
03	GPIO 2		5V (Powerboost)	5V	04
05	GPIO 3		GND (Powerboost)	Ground	06
07	GPIO 4	START		GPIO 14	08
09	Ground	GND (Cupcade)		GPIO 16	10
11	GPIO 17	UP	SELECT	GPIO 18	12
13	GPIO 27	DOWN	GND (Select/Start)	Ground	14
15	GPIO 22	LEFT	RIGHT	GPIO 23	16
17	3.3V		A	GPIO 24	18
19	GPIO 10	B	GND (ABXYR)	Ground	20
21	GPIO 09	X	Y	GPIO 25	22
23	GPIO 11	L Shoulder	R Shoulder	GPIO 08	24
25	Ground	GND (L)		GPIO 07	26
27	ID_SD			ID_SC	28
29	GPIO 05			Ground	30
31	GPIO 06			GPIO 12	32
33	GPIO 13			Ground	34
35	GPIO 19			GPIO 16	36
37	GPIO 26			GPIO 20	38
39	Ground			GPIO 21	40

รูปที่ 2.20 การจัดเรียงขาของพอร์ตอินพุตเอาต์พุตต่อเนกประสงค์ [20]

2.3 รีโมท (Remote control)

รีโมท (Remote Control) เป็นอุปกรณ์อิเล็กทรอนิกส์ชนิดหนึ่ง ใช้ทำหน้าที่สั่งงานอุปกรณ์เครื่องใช้ไฟฟ้าต่างๆ จากระยะไกล อาจจะห่างจากเครื่องใช้ไฟฟ้าที่ต้องการควบคุมเพียงเล็กน้อย เช่น ภายในระยะ 1-30 เมตร หรือรีโมทบางตัวสามารถสั่งงานได้ไกลเป็นระยะทางหลายกิโลเมตรก็ได้ ขึ้นอยู่กับการออกแบบตัวส่งรีโมทจะสั่งงานได้ ต้องประกอบด้วย 2 สิ่ง คือ รหัส และตัวพาสัญญาณรหัสออกไป

2.3.1 รหัส (Code)

รหัส (Code) จะเป็นสัญญาณแบบดิจิทัล ซึ่งจะส่งออกไปได้ต้องมีตัวพาออกไป

2.3.2 ตัวพาสัญญาณออกไป (Carrier)

ตัวพาสัญญาณออกไป (Carrier) เป็นสื่อกลาง หรือตัวเชื่อมพาราส่งออกไปสู่ตัวรับ เพื่อถอดรหัสและใช้รหัสที่ถอดได้ ไปควบคุมเครื่องใช้ไฟฟ้าต่าง ๆ หรือ หากเป็นเครื่องรับวิทยุ ก็คือเสียงที่เราได้ยินออกมาจากลำโพงนั่นเอง

ตัวพาสัญญาณ ในปัจจุบันมักพบอยู่ใน 3 ลักษณะ คือ

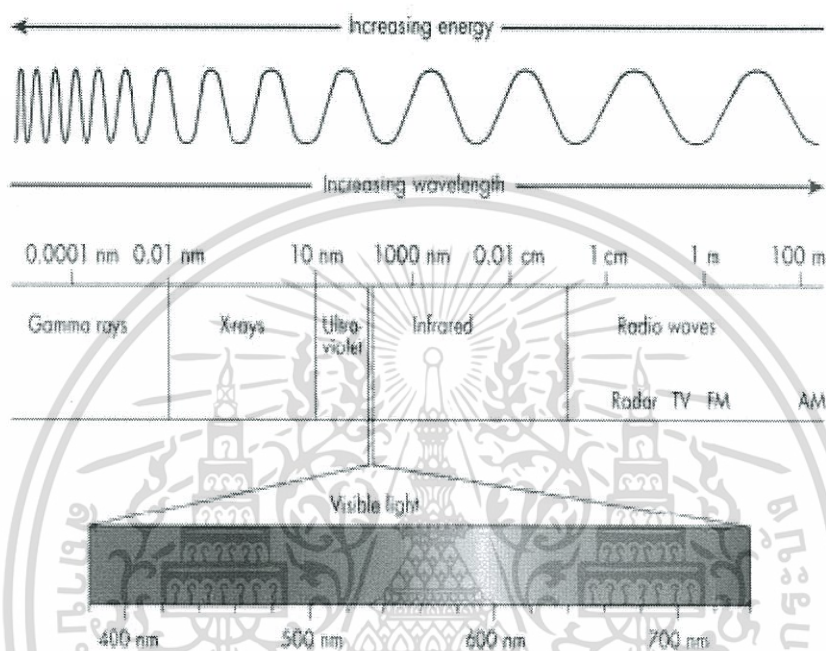
- แบบใช้สายนำสัญญาณ วิธีนี้ใช้ส่งรหัสผ่านสายตัวนำ หรือสายนำสัญญาณโดยตรง
- แบบใช้แสงเป็นตัวพาไป วิธีนี้ใช้การผสมรหัสเข้ากับแสง อาจจะเป็นแสงที่ตามองเห็น เช่น การสื่อสารผ่านใยแก้ว หรือแสงที่ตามองไม่เห็น คือแสงอินฟราเรด (Infrared) เช่น รีโมทของทีวี
- แบบใช้คลื่นวิทยุเป็นตัวพาไป วิธีนี้ใช้การผสมรหัสเข้ากับคลื่นวิทยุ เช่น สถานีส่งวิทยุ ทั้งแบบเอเอ็ม (AM) และ เอฟเอ็ม (FM) การส่งโทรทัศน์ รีโมทควบคุมการเปิดปิดประตูไฟฟ้าที่บ้านและรถยนต์ รีโมทที่ควบคุมของเล่นที่ใช้คลื่นวิทยุ เป็นต้น

2.4 การส่งและการรับสัญญาณอินฟราเรด (Infrared)

2.4.1 รังสีอินฟราเรด

คลื่นแม่เหล็กไฟฟ้าที่มีความถี่อยู่ในช่วง 10^{11} - 10^{14} เฮิรตซ์หรือความยาวคลื่นประมาณ 1mm-1 μ m เรียกว่า รังสีอินฟราเรด หรือเรียกอีกอย่างหนึ่งว่า คลื่นความถี่สั้น (Millimeter waves) และยังเป็นคลื่นแม่เหล็กไฟฟ้าที่มีความถี่ต่ำกว่าแสงที่ตามองเห็น ลำแสงอินฟราเรดเดินทางเป็นเส้นตรง ไม่สามารถผ่านวัตถุทึบแสง และสามารถสะท้อนแสงในวัสดุผิวเรียบได้เหมือนกับแสงทั่วไปใช้มากในการสื่อสารระยะใกล้ คุณสมบัติเด่นๆของแสงอินฟราเรดคือ เป็นคลื่นสั้น การเดินทางของแสงเป็นเส้นตรง ราคาถูก ง่ายต่อการผลิต ปลอดภัยต่อการดักสัญญาณ ไม่ทะลุผ่านวัตถุทึบแสงทำให้สามารถติดตั้งในพื้นที่จำกัดหรือห้องติดกันได้นั่นเอง แสดงดังรูปที่ 2.21 เนื่องด้วยคุณสมบัติต่างๆของรังสีอินฟราเรดที่เหมาะสมแก่การใช้งานจึงมีการประยุกต์นำมาใช้กันอย่างแพร่หลายในปัจจุบันโดยส่วนใหญ่เป็นอุปกรณ์ทางด้านอิเล็กทรอนิกส์ และ การสื่อสารเช่น รีโมทคอนโทรลของ โทรทัศน์, โทรศัพท์มือถือ และ กล้องส่องทางไกล เป็นต้น การที่อินฟราเรด เป็นคลื่น

สั้นดังที่ได้กล่าวมาแล้วทำให้ไม่มีปัญหาในเรื่องการรบกวนของสัญญาณในการส่งและการควบคุม แต่สัญญาณอินฟราเรดก็มีข้อเสีย คือ การถูกแสงจากแหล่งอื่นรบกวนเช่น แสงอาทิตย์ แสงไฟ เป็นต้น



รูปที่ 2.21 Electromagnetic Spectrum [2]

2.4.2 การส่งและการรับสัญญาณจากรีโมทคอนโทรลผ่านอินฟราเรด

การส่งสัญญาณอินฟราเรดจากรีโมทคอนโทรลผ่านอินฟราเรด คือ การประยุกต์ใช้คลื่นอินฟราเรดจะเป็นการประยุกต์ใช้ในการสื่อสารแบบไร้สาย (Wireless communication) ในการควบคุมอุปกรณ์เครื่องใช้ไฟฟ้า โดยการส่งสัญญาณผ่านทาง LED (Light emitting diode) โดยตัวส่ง (Transmitter) หรือ laser diode และจะมีตัวรับ (receiver) ทำการแปลงสัญญาณข้อมูลให้กลับไปเป็นเหมือนข้อมูลเริ่มแรกหรือข้อมูลต้นทางนั่นเอง

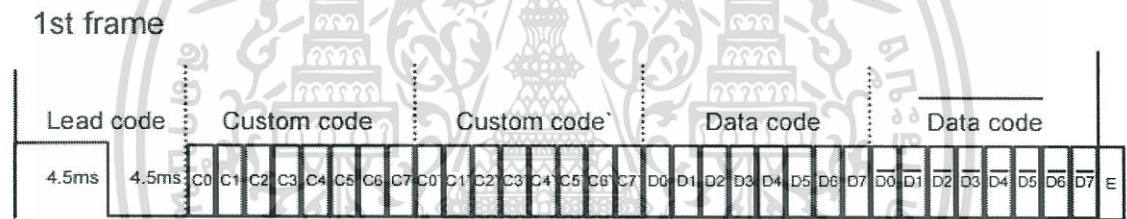
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 สัญญาณรีโมทคอนโทรลมาตรฐานซัมซุง

รีโมทยี่ห้อซัมซุงมีมาตรฐานสำหรับการควบคุมโดยใช้รีโมทคอนโทรลผ่านอินฟราเรดในผลิตภัณฑ์ของตนเองทั้งหมด ดังแสดงในรูปที่ 2.22 และ 2.23 โดยแบ่งออกได้เป็น 3 รูปแบบคือ

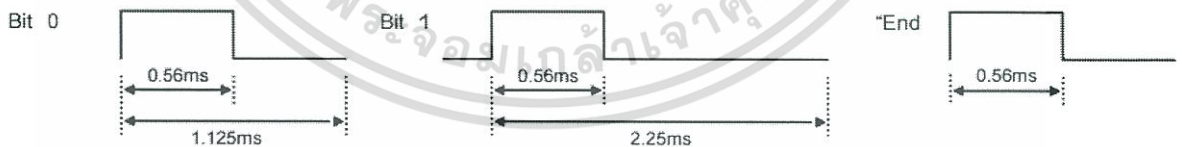
1. Start bit มีค่าช่วงเวลา 4.5ms ON และ 4.5ms OFF
2. 32 data bits stream
ประกอบด้วย bit "1" มีค่าช่วงเวลา 560 μ s ON และ 1690 μ s OFF
bit "0" มีค่าช่วงเวลา 560 μ s ON และ 560 μ s OFF
3. Stop bit มีค่าช่วงเวลา 560 μ s ON และ 560 μ s OFF

- Configuration of Frame



รูปที่ 2.22 ข้อมูลบิตซัมซุงแบบมาตรฐาน [7]

- Bit Description



รูปที่ 2.23 ข้อมูลเฟรมซัมซุงแบบมาตรฐาน [7]

ในการส่งข้อมูลสัญญาณจะมีสัญญาณคลื่นพาห์ขนาด 37.9 KHz รวมไปถึงสัญญาณข้อมูล โดยที่ข้อมูลทั้งหมด 32 บิต เมื่อจะส่งก็จะเริ่มด้วยบิต START ขนาด 9 ms จากนั้นก็จะตามด้วย ข้อมูลขนาด 32 บิต ลักษณะโดยที่สัญญาณของข้อมูล "0" คือจะมีสัญญาณ low นาน 560 us และสัญญาณ high นาน 560 us ส่วนข้อมูล "1" จะมีสัญญาณ low นาน 1.69 ms และสัญญาณ high นาน 560 us

2.5 ไดโอดเปล่งแสง (Light-emitting Diode)

ไดโอดเปล่งแสงเป็นอุปกรณ์สารกึ่งตัวนำอย่างหนึ่ง จัดอยู่ในจำพวกไดโอดที่สามารถเปล่งแสงในช่วงสเปกตรัมแคบ เมื่อถูกไบอัสทางไฟฟ้าในทิศทางไปข้างหน้า ปรากฏการณ์นี้อยู่ในรูปของ electroluminescence สีของแสงที่เปล่งออกมานั้นขึ้นอยู่กับองค์ประกอบทางเคมีของวัสดุกึ่งตัวนำที่ใช้ และเปล่งแสงได้ใกล้ช่วงอัลตราไวโอเล็ต ช่วงแสงที่มองเห็น และช่วงอินฟราเรด ผู้พัฒนาไดโอดเปล่งแสงขึ้นเป็นคนแรก คือ นิก โฮโลนยัค (Nick Holonyak Jr.) (เกิด ค.ศ. 1928) แห่งบริษัทเจเนรัล อิเล็กทริก (General Electric Company) โดยได้พัฒนาไดโอดเปล่งแสงในช่วงแสงที่มองเห็น และสามารถใช้งานได้ในช่วงปฏิบัติการเป็นครั้งแรก เมื่อ ค.ศ. 1962 หลักการทำงานของหลอด LED หลอด LED หรือไดโอดเปล่งแสง โครงสร้างประกอบไปด้วยสารกึ่งตัวนำสองชนิด (สารกึ่งตัวนำชนิด N และสารกึ่งตัวนำชนิด P) ประคบเข้าด้วยกัน มีผิวข้างหนึ่งเรียกคล้ายกระจกเมื่อจ่ายไฟฟ้ากระแสตรงผ่านตัว LED โดยจ่ายไฟบวกให้ขาแอนโนด (A) จ่ายไฟลบให้ขาแคโทด (K) ทำให้อิเล็กตรอนที่สารกึ่งตัวนำชนิด N มีพลังงานสูงขึ้น จนสามารถวิ่งข้ามรอยต่อจากสารชนิด N ไปรวมกับโฮลในสารชนิด P การที่อิเล็กตรอนเคลื่อนที่ผ่านรอยต่อ PN ทำให้เกิดกระแสไหล เป็นผลให้ระดับพลังงานของอิเล็กตรอนเปลี่ยนไปและคายพลังงานออกมาในรูปคลื่นแสง สีของแสงที่เกิดจากรอยต่อจะขึ้นอยู่กับชนิดของวัสดุที่นำมาใช้ในการสร้าง LED ทั้งชนิดที่เป็นของเหลวและก๊าซ เช่น ใช้แกลเลียมฟอสไฟด์ (GALLIUM PHOSPHIDE, GaP) ทำให้เกิดแสงสีแดง ใช้แกลเลียมอาร์เซไนด์ ฟอสไฟด์ (GALLIUM ARSENIDE PHOSPHIDE, GaAsP) เกิดแสงสีเหลืองและเขียว การควบคุมปริมาณแสงสว่างจะควบคุมกระแสที่ไหลผ่านหลอด LED หากกระแสที่ไหลสูงเกินไปจะทำให้หลอดมีความสว่างมาก แต่หากป้อนกระแสสูงเกินไปจะทำให้บริเวณรอยต่อของสารกึ่งตัวนำเกิดความร้อนปริมาณมากจนทำให้โครงสร้างหลอดเสียหายไม่สามารถใช้งานได้ อีก ความยาวคลื่นของสีต่างๆของไดโอดเปล่งแสงแสดงดังตารางที่ 2.2

ตารางที่ 2.2 ช่วงสีพลังงานของไดโอดเปล่งแสง

สีของ LED	ความยาวคลื่น (นาโนเมตร)	แรงดันตกคร่อม (โวลต์)
อินฟราเรด	> 760	$\Delta V < 1.9$
แดง	610 ถึง 760	1.63 ถึง 2.03
ส้ม	590 ถึง 610	2.03 ถึง 2.10
เหลือง	570 ถึง 590	2.10 ถึง 2.18
เขียว	500 ถึง 570	1.9 ถึง 4.0
น้ำเงิน	450 ถึง 500	2.48 ถึง 3.7
ม่วง	400 ถึง 450	2.76 ถึง 4.0
อัลตราไวโอเล็ต	< 400	3.1 ถึง 4.4
ขาว	ย่านกว้าง	3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 ลักษณะของตัว LED

LED จะทำจากสารกึ่งตัวนำ P และ N โดยจะมี 2 ขาในการใช้งาน (ยกเว้นบางประเภท เช่น LED แบบให้สีสองสีในหลอดเดียวกันอาจจะมี 3 ขาได้) โดยขาของ LED จะมีชื่อเรียกดังนี้

- ขา A หรือที่เรามักเรียกว่าขา อนโอด โดยขานี้จะต้องป้อนไฟบวก (+) ให้เท่านั้น
- ขา K หรือที่เรามักเรียกว่าขา แคโทด โดยขานี้จะต้องป้อนไฟลบ (-) ให้เท่านั้น

ที่ตัว LED แบบหลอดจะสังเกตว่าจะมีรอยบากอยู่ด้านหนึ่ง โดยทั่วไปตำแหน่งรอยบากนี้จะแสดงตำแหน่งขา K แต่ มันก็ไม่จำเป็นเสมอไปครับทางที่ดีเราควรตรวจสอบด้วยตัวเองจะดีกว่า ซึ่งจะอยู่ในแสดงรูปตัวอย่างลักษณะหลอด LED ดังรูปที่ 2.24



รูปที่ 2.24 รูปแสดงลักษณะของหลอด LED [2]

2.5.2 ข้อดีของหลอด LED

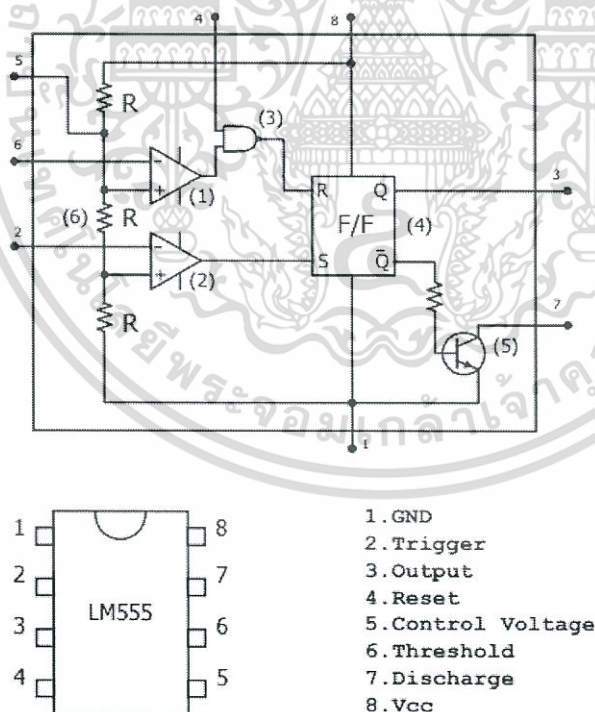
มีประสิทธิภาพการให้แสงสว่างสูง และทิศทางแสงสว่างของ LED จะส่องไปเฉพาะด้านหน้าเท่านั้นลดการสูญเสียของแสงสว่าง ใช้พลังงานน้อย ทนต่อการสั่นสะเทือนและแรงกระแทก จึงเหมาะสมสำหรับติดตั้งในเครื่องบินหรือรถยนต์ สามารถเปิดปิดได้บ่อยครั้ง และเมื่อเปิดจะให้แสงสว่างโดยทันที อายุการใช้งานยาวนานถึง 100,000 ชั่วโมง สามารถควบคุมคุณภาพของแสงที่ปล่อยออกมาได้จึงสามารถนำไปใช้ให้แสงสว่างในบางสถานที่ได้ เช่น การให้แสงสว่างกับภาพเขียน เนื่องจากสามารถควบคุมแสงสว่างจาก LED ไม่ให้มีส่วนผสมของแสงที่เป็นอันตรายต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพเขียน เช่น แสงอินฟราเรด และแสงอุลตราไวโอเล็ต ปล่อยความร้อนออกมาน้อยมาก ทำให้ลดการสูญเสียพลังงานไฟฟ้าในส่วนเครื่องปรับอากาศ การดูแลรักษาต่ำ น้ำหนักเบา,ขนาดเล็ก ข้อจำกัดของหลอด LED ในการนำหลอด LED มาใช้งานต้องมีการทดสอบสีว่า แสงที่ออกมาเป็นแสงสีที่ถูกต้องหรือไม่ ราคาหลอด LED ยังสูงกว่าหลอดฟลูออเรสเซนต์อยู่มาก

2.6 ทฤษฎีเบื้องต้นของไอซีเบอร์ 555

ไอซีเบอร์ 555 เป็นไอซีที่นิยมใช้กันมากในการนำไปสร้างสัญญาณรูปคลื่นแบบต่างๆ เช่น สัญญาณสแควร์เวฟ (Square wave Signal), สัญญาณพัลส์ (Pulse Signal), สัญญาณลาด (Ramp Signal) และวงจรตั้งเวลา ไอซีเบอร์ 555 เป็นอุปกรณ์วงจรรวมที่มีอุปกรณ์อิเล็กทรอนิกส์อื่นๆ อยู่ภายใน และมีส่วนที่ต้องต่อภายนอกเพื่อควบคุมการทำงาน และใช้งานเป็นลักษณะต่างๆ ซึ่งง่ายต่อการออกแบบ และง่ายต่อการสร้างสัญญาณพัลส์ความถี่ต่างๆ อีกทั้งยังสามารถเข้าใจการทำงานได้ง่าย รูปที่ 2.25 จะแสดงวงจรและส่วนประกอบของไอซีเบอร์ 555



รูปที่ 2.25 วงจรและส่วนประกอบของไอซีเบอร์ 555 [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

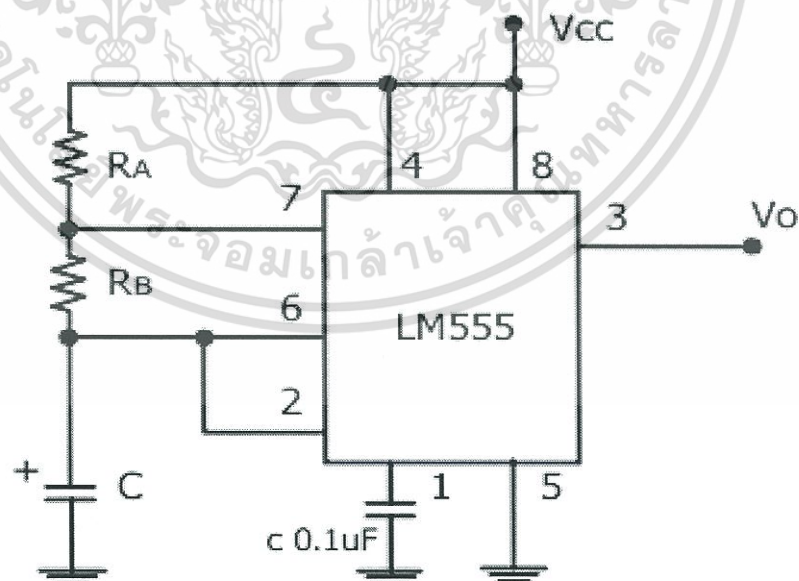
การทำงานของวงจรที่สร้างจากไอซีเบอร์ 555 แบ่งออกเป็น 3 วงจรคือ

1. วงจรโมนอสเตเบิล (Monostable Multivibrator)
2. วงจรอะสเตเบิล (Astable Multivibrator)
3. วงจรไบสเตเบิลมีลต์ติไวเบรเตอร์ (Bistable Multivibrator)

ซึ่งในปริยญาณิพนธ์นี้นั้นได้เลือกใช้เพียงแค่ วงจรอะสเตเบิล (Astable Multivibrator) ในการสร้างควมถี่เพียงรูปแบบเดียว

2.6.2 วงจรอะสเตเบิล (Astable Multivibrator)

วงจรอะสเตเบิลคือวงจรสร้างสัญญาณสแควร์เวฟ (Square wave Signal) มีความถี่ที่สามารถกำหนดได้ตามความต้องการ ด้วยค่าคาปาซิเตอร์ และค่าความต้านทาน R_A และ R_B โดยในวงจรในรูปที่ 2.26 นอกจากจะสามารถกำหนดความถี่ของสัญญาณสแควร์เวฟ (Square wave Signal) ด้วยค่า C และค่า $R_T = R_A + R_B$ แล้วยังสามารถที่จะกำหนดช่วงเวลา “ON” และ “OFF” ของสัญญาณด้วยค่าที่แตกต่างกันของ R_A และ R_B ได้อีก



รูปที่ 2.26 แสดงวงจร Astable multivibrator [8]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการทํางาน

1. ใช้หลักการเก็บประจุและคายประจุของคาปาซิเตอร์โดยจะทำการเก็บประจุผ่าน R_A , R_B และจะคายประจุผ่าน R_B และ T_r ภายในลงกราวด์
2. ขณะที่คาปาซิเตอร์เก็บประจุจนถึงระดับแรงดัน $2/3V_{CC}$ ขณะนั้นแรงดัน ของ $V_0 = V_{CC}$ ขณะที่คาปาซิเตอร์คายประจุออกเกิดแรงดันตกคร่อมมีช่วงจาก $2/3V_{CC}$ ถึง $V_{CC}/3$ ขณะนั้นแรงดัน ของ $V_0 = 0$



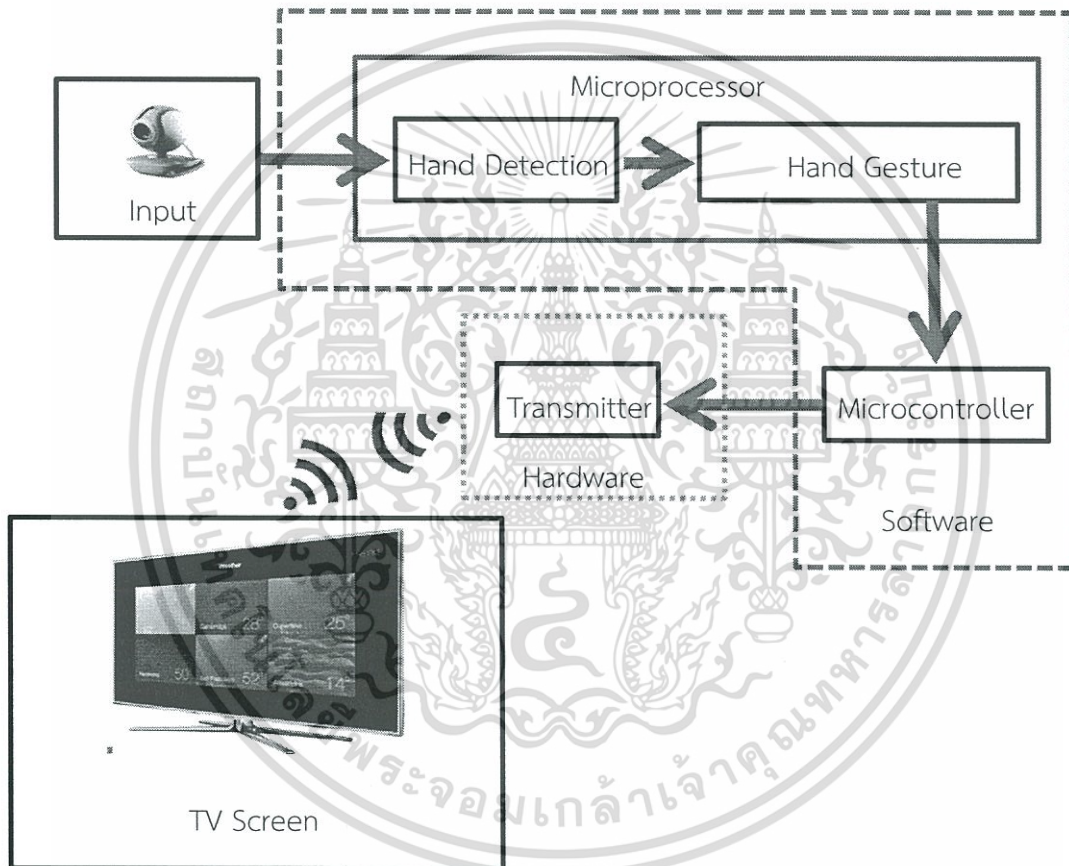
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการจัดทำปฏิญญานิพนธ์

3.1 การออกแบบระบบ

บล็อกไดอะแกรมการทำงานของอุปกรณ์ควบคุมโทรทัศน์โดยอาศัยการประมวลผลภาพของท่าทางมือ แสดงดังรูปที่ 3.1



รูปที่ 3.1 บล็อกไดอะแกรมการทำงานของอุปกรณ์ควบคุมโทรทัศน์

โดยอาศัยการประมวลผลภาพของท่าทางมือ

จากรูปที่ 3.1 แสดงบล็อกไดอะแกรมการทำงานของอุปกรณ์ควบคุมโทรทัศน์ โดยขั้นตอนบอร์ดราสเบอร์รี่พายจะรับข้อมูลภาพสีจากกล้องเว็บแคมมาทำการประมวลผลภาพ จากนั้นจะทำการแปลงเป็นข้อมูลคำสั่งในรูปแบบต่างๆ แล้วส่งต่อไปยังไมโครคอนโทรลเลอร์เพื่อทำการสร้างสัญญาณพัลส์ของแต่ละรูปแบบคำสั่งที่ได้รับมาจากการประมวลผลภาพ พร้อมส่งสัญญาณผ่านไดโอดเปล่งแสงเพื่อควบคุมโทรทัศน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 การออกแบบด้านซอฟต์แวร์

จากรูปที่ 3.1 เป็นการออกแบบระบบซอฟต์แวร์และ ฮาร์ดแวร์ของปริญญาโทฉบับนี้ ซึ่งจะแบ่งการทำงานเป็นส่วนของการประมวลผลภาพ และส่วนการควบคุมโทรทัศน์ โดยโฟลว์ชาร์ตการทำงานของระบบโดยรวมจะแสดงดังรูปที่ 3.2



รูปที่ 3.2 โฟลว์ชาร์ตการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.2 นั้นคือโฟลว์ชาร์ตการทำงานของระบบโดยรวม โดยเริ่มต้นไมโครโปรเซสเซอร์ จะทำการรับข้อมูลภาพสีภาพจากกล้องเว็บแคมมาประมวลผลภาพฝ่ามือแล้วแปลงเป็นชุดคำสั่งแต่ละรูปแบบของลักษณะท่าทางฝ่ามือ จากนั้นส่งชุดข้อมูลคำสั่งไปยังไมโครคอนโทรลเลอร์เพื่อสร้างสัญญาณพัลส์ตามรูปแบบข้อมูลทำการคูณกับสัญญาณพัลส์ความถี่ 38 กิโลเฮิร์ตซ์ที่สร้างจากวงจระอสเตเบิล (Astable Multivibrator) ด้วย AND gate พร้อมส่งสัญญาณผ่านไดโอดเปล่งแสงเพื่อไปควบคุมทรานซิสต์

3.1.1.1 การออกแบบด้านซอฟต์แวร์สำหรับประมวลผลภาพ

จากรูปที่ 3.1 ในส่วนของ Hand Detection ได้ใช้การประมวลผลภาพดิจิทัลจากภาพฝ่ามือที่ได้รับมาจากกล้องมาทำการเป็นการวิเคราะห์ภาพฝ่ามือมนุษย์ โดยจะแบ่งลักษณะการทำงานออกเป็นสองส่วนคือ ส่วนแรกระบุหาตำแหน่งของฝ่ามือ และส่วนของการวิเคราะห์ลักษณะของมือที่บ่งชี้ถึงคำสั่งควบคุมทรานซิสต์ โดยมีลำดับขั้นตอนการทำงานดังรูปที่ 3.3



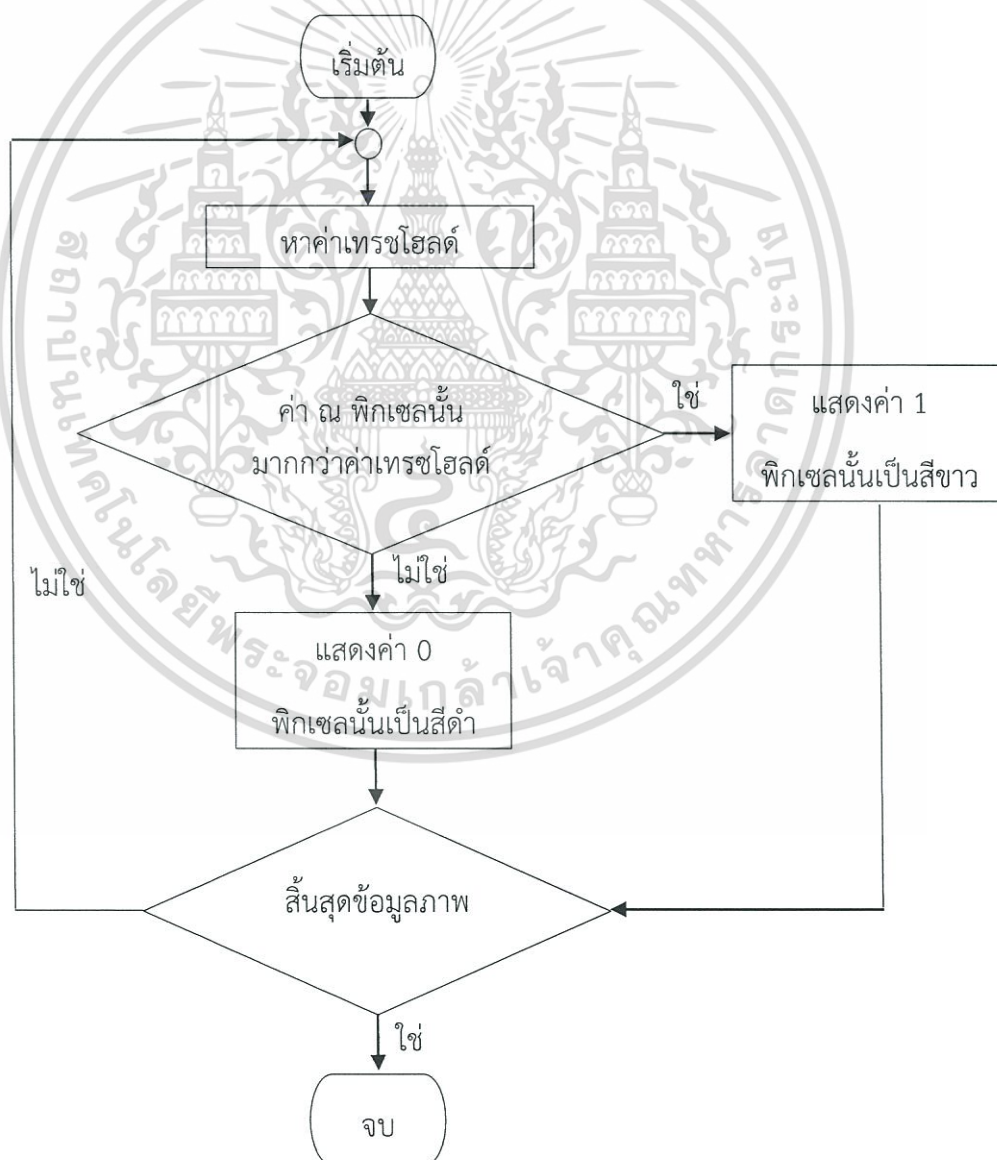
รูปที่ 3.3 โฟลว์ชาร์ตการหาตำแหน่งของฝ่ามือในภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.3 นำภาพอินพุตที่ได้จากกล้อง Webcam มาทำการแปลงให้เป็นภาพระบบสี HSV และใช้เกาส์เซียนฟิลเตอร์มาควบคุมระดับความละเอียดของขอบภาพ พร้อมทั้งช่วยลดสัญญาณรบกวน

ขั้นตอนการแปลงภาพให้เป็นภาพไบนารีโดยวิธีการเทรชโฮลด์

จากรูปที่ 3.1 ในส่วนของ Hand Detection ได้มีการกระบวนการ background subtraction หรือวิธีการหาเทรชโฮลด์ ซึ่งเป็นกระบวนการหนึ่งที่ใช้ในการแยกวัตถุที่สนใจออกจากภาพพื้นหลัง โดยวัตถุที่สนใจในปริญญานิพนธ์นี้นั้นคือภาพฝ่ามือ ซึ่งจะแสดงขั้นตอนการทำงานของกระบวนการนี้ดังรูปที่ 3.4

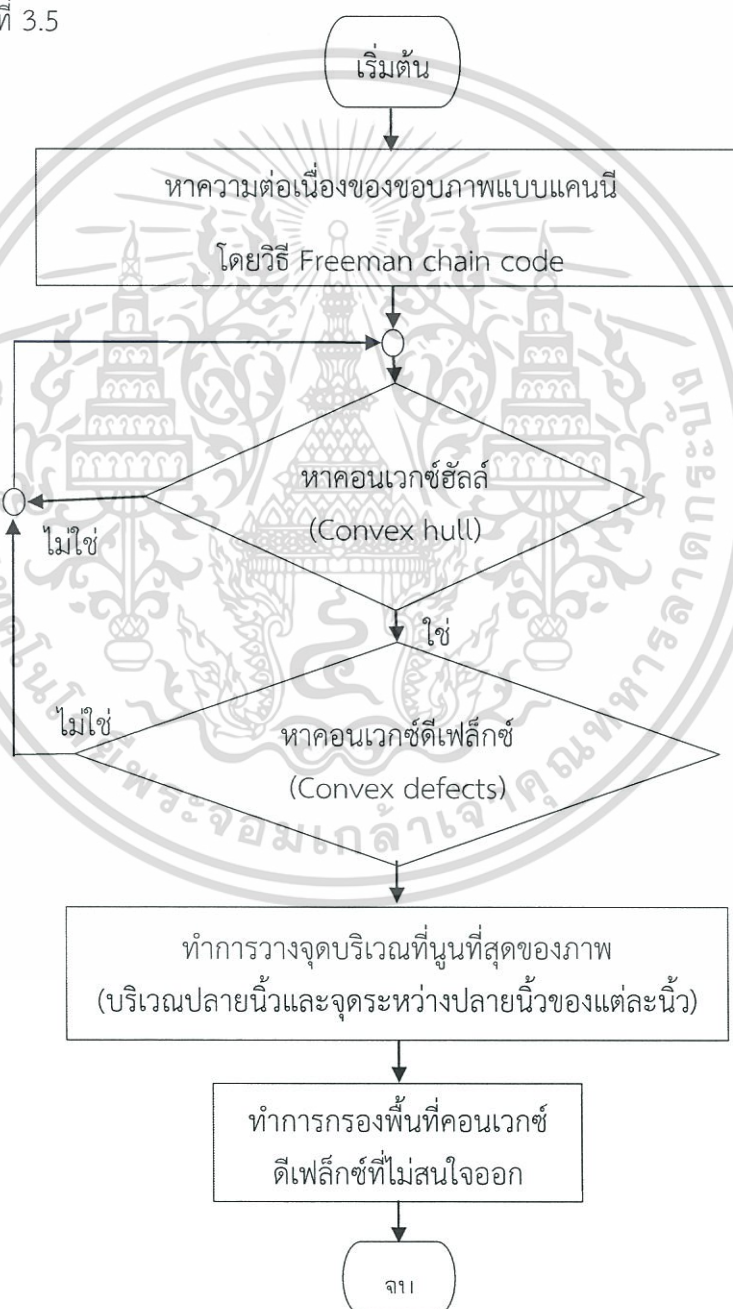


รูปที่ 3.4 โฟลว์ชาร์ตการทำงานของกระบวนการแปลงภาพให้เป็นภาพไบนารีโดยวิธีการเทรชโฮลด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.4 ได้แสดงโฟลว์ชาร์ตการทำงานของกระบวนการแปลงภาพให้เป็นภาพไบนารีโดยวิธีการเทรซโฮสต์ โดยเริ่มจากการหาขอบภาพโดยวิธีวิธีการหาขอบของภาพ และแปลงภาพเป็นภาพไบนารี เพื่อหาตำแหน่งของฝ่ามือจากวิธีการของฟรีแมนเซนโค้ด
ขั้นตอนการตรวจจบลักษณะของฝ่ามือ

จากรูปที่ 3.1 ในส่วนของ Hand Gesture เป็นกระบวนการหนึ่งที่ใช้ในการตรวจจบลักษณะของฝ่ามือเพื่อใช้เป็นคำสั่งภาษามือที่ควบคุมโทรทัศน์ในรูปแบบต่างๆ ซึ่งจะแสดงขั้นตอนการทำงานของกระบวนการนี้ดังรูปที่ 3.5



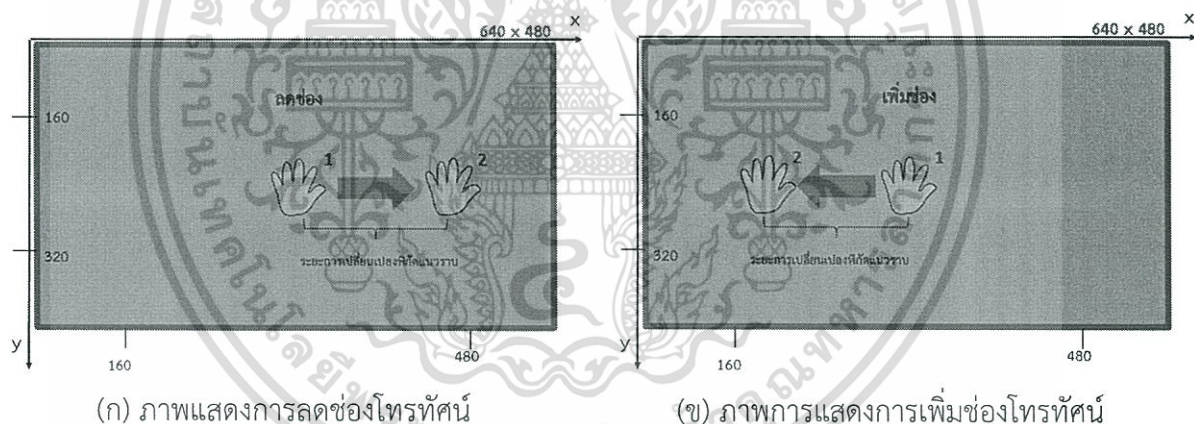
รูปที่ 3.5 โฟลว์ชาร์ตการตรวจจบลักษณะของฝ่ามือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.5 ได้แสดงโฟลว์ชาร์ตการตรวจจับลักษณะของฝ่ามือ โดยเริ่มต้นจากหาตำแหน่งของฝ่ามือโดยพิจารณาเฉพาะขอบภาพอาศัยโดยกระบวนการของฟรีแมนเซนโค้ด และทำการพรอตจุดที่แต่ละของปลายนิ้ว และระหว่างปลายนิ้วพร้อมตั้งชื่อจุดปลายนิ้วของแต่ละนิ้วมือ ดังรูปที่ 3.5

3.1.1.2 การออกแบบด้านซอฟต์แวร์สำหรับควบคุมโทรศัพท์

จากรูปที่ 3.1 ในส่วนของ Microprocessor เมื่อได้ทำการแยกฝ่ามือออกจากพื้นหลัง และทำการหาเส้นคอนทัวร์ของฝ่ามือ เพื่อนำไปหาจุดเซนทรอยด์แล้วนำมาเปรียบเทียบพิกัดตามแนวแกนราบของภาพเฟรมใหม่และพิกัดตามแนวแกนราบของภาพเฟรมเก่า จากนั้นนำค่าพิกัดทั้งสองค่าลบกันเพื่อหาค่าส่วนต่างของพิกัดตามแนวแกนราบที่เปลี่ยนไปของภาพทั้งสองเฟรม ซึ่งถ้าค่าส่วนต่างของพิกัดที่เปลี่ยนแปลงไปมากกว่า 150 พิกเซล จะทำการส่งสัญญาณปุ่มลดช่องโทรศัพท์ แต่ถ้าค่าส่วนต่างของพิกัดที่เปลี่ยนแปลงไปมีค่ามากกว่า -150 พิกเซลจะส่งสัญญาณปุ่มเพิ่มช่อง โดยจะแสดงภาพตัวอย่างการทำงานและโฟลว์ชาร์ต ดังรูปที่ 3.6 - 3.8 ตามลำดับ



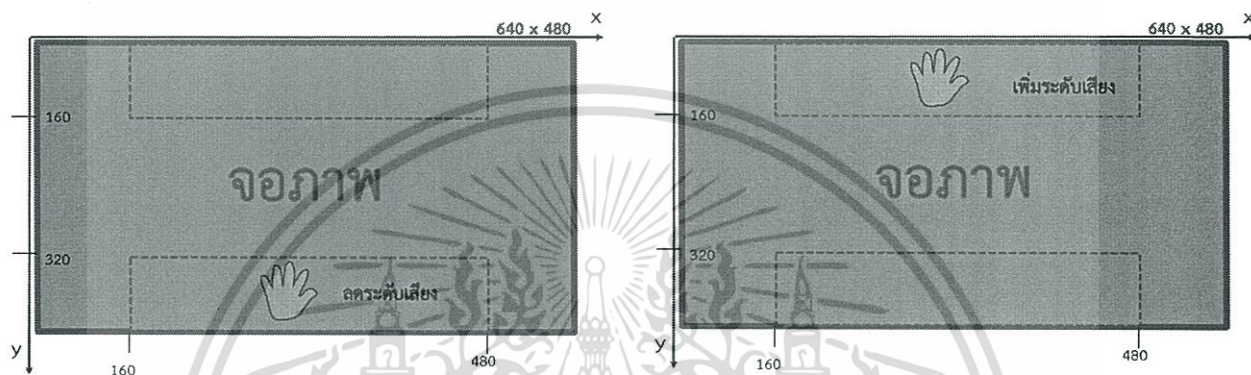
รูปที่ 3.6 รูปแบบการทำงานของ การเพิ่ม-ลดช่องโทรศัพท์

ตำแหน่งที่ 1 แทนตำแหน่งพิกัดตามแนวราบของภาพเฟรมเก่า

ตำแหน่งที่ 2 แทนตำแหน่งพิกัดตามแนวราบของภาพเฟรมใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อภาพฝ่ามืออยู่ในพิกัดตามแนวแกนตั้ง 0-160 พิกเซลของจอภาพ โปรแกรมจะทำการส่งสัญญาณปุ่มเพิ่มระดับเสียง แต่ถ้าภาพฝ่ามืออยู่ในพิกัดตามแนวแกนตั้ง 360 - 480 พิกเซลของจอภาพ โปรแกรมจะทำการส่งสัญญาณลดระดับเสียง ซึ่งจะแสดงภาพตัวอย่างการทำงาน และโพล์ชาร์ต ดังรูปที่ 3.7 และ 3.8 ตามลำดับ

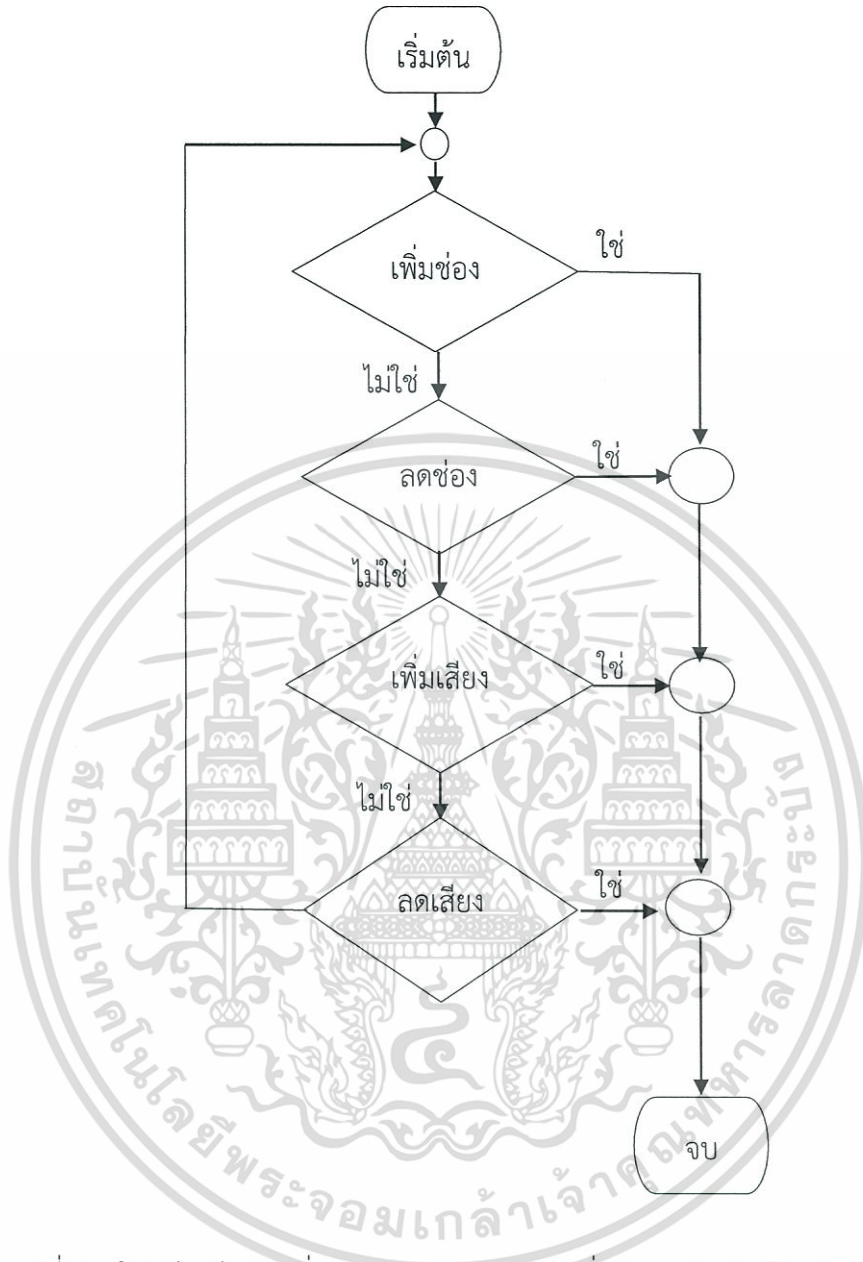


(ก) ภาพแสดงการลดระดับเสียง

(ข) ภาพแสดงการเพิ่มระดับเสียง

รูปที่ 3.7 รูปแบบการทำงานของปุ่มเพิ่ม-ลดระดับเสียงโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

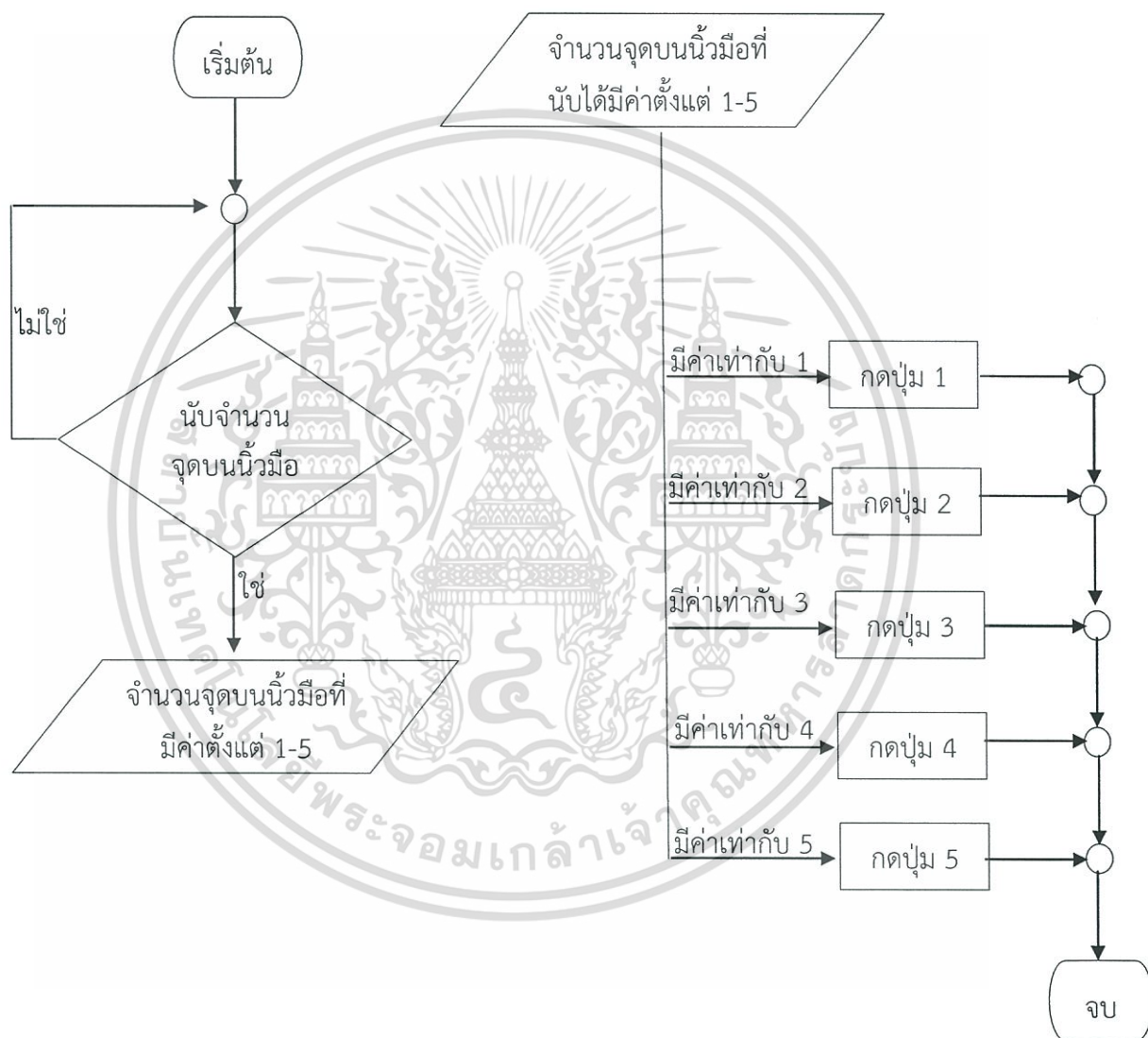


รูปที่ 3.8 โฟลว์ชาร์ตการเพิ่ม-ลดช่อง และการปรับเพิ่ม-ลดระดับเสียงโทรทัศน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำงานของกระบวนการพิจารณาภาพฝ่ามือเพื่อส่งคำสั่งไปยังไมโครคอนโทรลเลอร์

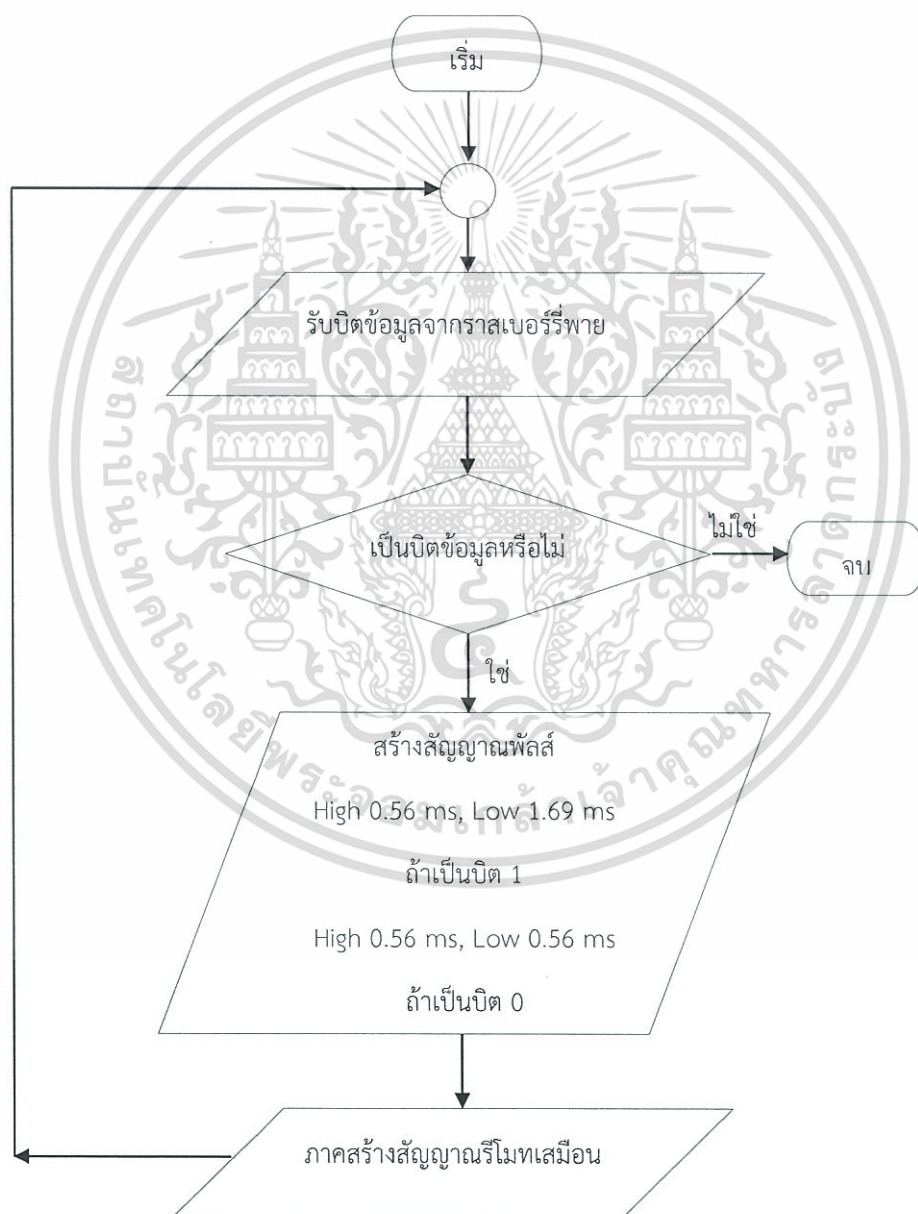
จากรูปที่ 3.1 ในส่วนของ Microprocessor เป็นขั้นตอนการวิเคราะห์ภาพลักษณะท่าทางของมือนั้นได้ทำการนำจุดที่วางบนปลายนิ้วมานับจำนวนจุด ซึ่งจำนวนจุดที่ได้บนปลายนิ้วนั้นจะนำมารวมกัน โดยผลลัพธ์ที่ได้จะเป็นตัวบ่งชี้เพื่อใช้ในการควบคุมโทรศัพท์ ดังรูปที่ 3.9



รูปที่ 3.9 โฟลว์ชาร์ตกระบวนการพิจารณาภาพฝ่ามือ เพื่อส่งคำสั่งไปยังไมโครคอนโทรลเลอร์

การออกแบบด้านซอฟต์แวร์สำหรับการควบคุมโทรทัศน์

จากรูปที่ 3.1 ในส่วนของ Microcontroller นั้นหลังจากระบบประมวลผลภาพได้ทำการประมวลผลภาพสำเร็จจะมีการส่งบิตข้อมูลจากบอร์ดทรานสเบอร์รี่ไปยังบอร์ดอาร์ดูโนเพื่อทำการสร้างสัญญาณตามชุดข้อมูลที่ระบบข้างต้นบ่งชี้มาเป็นสัญญาณพัลส์ซึ่งจะนำไปมอดูเลตกับสัญญาณพัลส์ความถี่ 38 กิโลเฮิร์ตซ์ แสดงดังรูปที่ 3.10 และแสดงตัวอย่างการเชื่อมต่อระหว่างอุปกรณ์ในส่วน Microprocessor และ Microcontroller เข้าด้วยกัน ดังรูปที่ 3.11



รูปที่ 3.10 โฟลว์ชาร์ตการสร้างสัญญาณพัลส์ของบอร์ด Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 การออกแบบด้านฮาร์ดแวร์

3.1.2.1 การออกแบบวงจรความถี่ผลิตสัญญาณ 38 kHz

ในการออกแบบวงจรสร้างสัญญาณพัลส์ในปริภูมิตอนอนนี้ได้ใช้ไอซีเบอร์ 555 เพื่อทำการสร้างความถี่ความถี่ 38 kHz ซึ่งเป็นคลื่นพาห์ของสัญญาณรีโมทเหมือน แสดงดังรูปที่ 3.11 ทำการวิเคราะห์เวลาในการ charge และ discharge Capacitor เวลาในการ charge- C จาก 0 ถึง $V_{CC}/3$ ใช้เวลา t_1 จะได้ว่า

จากสมการ

$$V_C(t) = V_{CC}(1 - e^{-t/RC}) \quad (3.1)$$

แทน $V_C(t)$ ด้วยเวลาในการ charge- C จาก 0 ถึง $V_{CC}/3$

$$\text{ดังนั้นจะได้} \quad t_1 = 0.405 RC \quad (3.2)$$

และเวลาในการ Charge-C จาก 0 ถึง $2/3V_{CC}$ ใช้เวลา t_2 คือช่วงเวลาในการ Charge-C จาก 0 ถึง $2/3V_{CC}$ ซึ่งจะได้ว่า

แทน $V_C(t)$ ด้วยเวลาในการ Charge-C จาก 0 ถึง $2/3V_{CC}$

$$\text{ดังนั้นจะได้} \quad t_2 = 1.098 RC \quad (3.3)$$

และจาก $T_1 = t_2 - t_1$ ทำการแทนค่า t_1 และ t_2

$$\text{ดังนั้นจะได้} \quad T_1 = 0.69(R_A + R_B)C \quad (3.4)$$

เวลาในการ Discharge-C จาก $2/3V_{CC}$ ถึง $V_{CC}/3$ โดยที่ T_2 เป็นเวลาในการ Discharge-c จาก $2/3V_{CC}$ ถึง $V_{CC}/3$

จากสมการที่ (3.1)

แทน $V_C(t)$ และ V_{CC} ด้วยเวลาในการ Discharge-C จาก $2/3V_{CC}$ ถึง $V_{CC}/3$ และแก้สมการ

$$\text{ดังนั้นจะได้} \quad T_2 = 0.69(R_B)C \quad (3.5)$$

และจาก $T = T_1 + T_2$ ทำการแทนค่า T_1 และ T_2 จะได้

$$T = 0.69(R_A + 2R_B)C \quad (3.6)$$

โดยที่ T คือเวลาทั้งหมดที่ใช้ในการ Charge-Discharge-C

ดังนั้นสมการความถี่คือ

$$F = 1 / 0.69(R_A + 2R_B)C \quad (3.6)$$

ในที่นี้ต้องการสัญญาณพัลส์ความถี่ 38 kHz จึงได้กำหนดค่า C, R_A, R_B เพื่อให้ได้ความถี่ตามที่ต้องการ และค่าที่ได้คือ

$$C_1 = 1 \text{ nF}$$

$$R_A = 7.5 \text{ k}$$

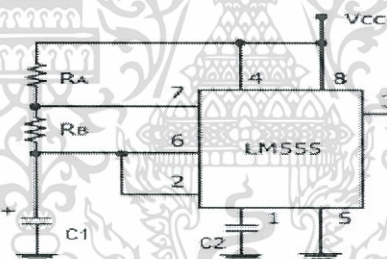
$$R_B = 15 \text{ k}$$

และทำการแทนค่าในสมการ $F = 1 / 0.69(R_A + 2R_B)C$ จะได้เป็น

$$F = 1 / 0.69(7500 + (2 \times 15000)) \times 1 \times 10^{-9}$$

$$F = 38647.343$$

$$F = 38.6 \text{ kHz}$$

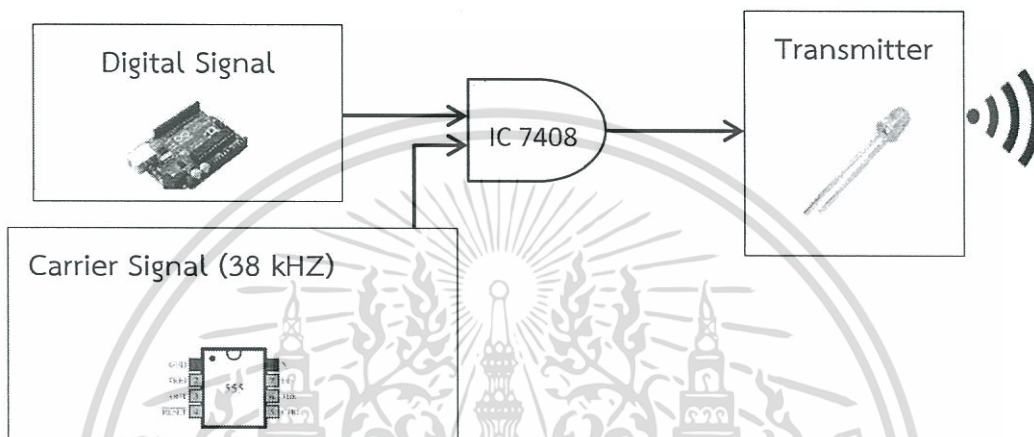


รูปที่ 3.11 วงจรสร้างสัญญาณพัลส์ด้วยไอซีเบอร์ 555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2.2 วงจรสร้างสัญญาณรีโมทเหมือน

จากรูปที่ 3.1 ในส่วนของ Transmitter สำหรับวงจรสร้างสัญญาณของรีโมทเหมือนนั้น ใช้ไอซีเบอร์ 7408 ทำหน้าที่เป็น AND gate ระหว่างสัญญาณพัลส์ความถี่ 38 kHz ที่สร้างจากวงจรในรูปที่ 3.11 และสัญญาณพัลส์เลียนแบบสัญญาณรีโมทโทรทัศน์ที่สร้างจากบอร์ดอาร์ดูโน สัญญาณรีโมทเหมือนที่ได้จะถูกส่งผ่านไดโอดเปล่งแสงเพื่อควบคุมโทรทัศน์ซึ่งจะแสดงดังรูปที่ 3.12



รูปที่ 3.12 วงจรสร้างสัญญาณรีโมท

3.2 เครื่องมือที่ใช้ในการทดลอง

1. ชุดโมดูลบอร์ดไมโครคอนโทรลเลอร์อาร์ดูโน รุ่น UNO R3
2. บอร์ดราสเบอร์รี่พาย (Raspberry Pi)
3. ไอซีเบอร์ 555
4. ไอซีเบอร์ 7408
5. กล้องเว็บแคม
6. รีโมทโทรทัศน์ยี่ห้อซัมซุง
7. ตัวรับสัญญาณอินฟราเรดสามขา TSOP 4808
8. ตัวส่งสัญญาณอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การเก็บผลการทดลอง

3.3.1 การทดสอบการตรวจจับภาพฝ่ามือ

ในการทดลองตรวจจับภาพฝ่ามือจากกล้อง จะทำการรับภาพอินพุตจากกล้องเว็บแคม แล้วแปลงเป็นภาพ HSV จากนั้นหาค่าเทรซโฮลด์โดยการกำหนดขอบเขตของค่า H , S และ V เพื่อแยกภาพฝ่ามือออกจากพื้นหลัง จากนั้นนำภาพฝ่ามือที่ได้มาปรับปรุงคุณภาพผ่านกระบวนการเปลี่ยนแปลงโครงสร้างของภาพด้วย Erosion และ Dilation เพื่อลดสัญญาณรบกวน หลังจากผ่านกระบวนการปรับปรุงคุณภาพแล้ว นำภาพฝ่ามือที่ได้ ไปหาเส้นคอนทัวร์และจุดเซนทรอยด์เพื่อหาพิกัดของฝ่ามือ จากนั้นทำการหา Convex hull และ Convexity defects เพื่อหาจุดปลายของนิ้วมือและทำการนับจำนวนนิ้ว

3.3.2 การทดสอบการสร้างสัญญาณรีโมทเสมือน

การสร้างสัญญาณรีโมทเสมือนเพื่อทำการควบคุมโทรทัศน์เริ่มจากการเก็บข้อมูลสัญญาณรีโมทของปุ่มโทรทัศน์แต่ละปุ่มเพื่อนำมาสร้างพัลส์ให้มีดีเลย์ตรงกับชุดของสัญญาณนั้นแล้วมาทำการมอดูเลตกับสัญญาณพัลส์ความถี่ 38 kHz จะได้สัญญาณรีโมทเสมือนเพื่อส่งผ่านไดโอดเปล่งแสงไปทำการควบคุมโทรทัศน์ต่อไป

3.3.3 การทดสอบการควบคุมโทรทัศน์ด้วยลักษณะท่าทางมือ

การประมวลผลภาพฝ่ามือแต่ละลักษณะท่าทางของมือนั้น บอร์ดราสเบอร์รี่พายจะรับภาพฝ่ามือจากกล้องเว็บแคมมาทำการประมวลผลภาพเพื่อบ่งชี้ท่าทาง จากนั้นจะส่งบิตข้อมูลของคำสั่งที่สอดคล้องกับท่าทางของฝ่ามือไปยังบอร์ดไมโครคอนโทรลเลอร์เพื่อสร้างสัญญาณรีโมทเสมือนไปควบคุมโทรทัศน์

3.3.4 ขั้นตอนการเก็บผลการทดลอง

ในการจัดเก็บผลการทดลองนั้นได้อาศัยเครื่องมือด้วยกัน 6 อย่างคือ ออสซิลโลสโคป, โทรทัศน์ยี่ห้อซัมซุง, รีโมทโทรทัศน์, บอร์ดอาดูโน่, บอร์ดราสเบอร์รี่พาย, กล้องเว็บแคม และวงจรรีโมทเสมือนโดยมีลำดับขั้นตอนดังต่อไปนี้

1. นำออสซิลโลสโคปมาวัดสัญญาณที่ได้จากการกดปุ่มเพื่อเรียนรู้รูปแบบของสัญญาณ พร้อมทั้งนำไปเขียนโปรแกรมในบอร์ดอาดูโน่

2. หลังจากเขียนโปรแกรมเสร็จแล้วนั้น นำสัญญาณที่ได้จากการเขียนโปรแกรมจากบอร์ดอาดูโนมา AND gate กับความถี่ 38 กิโลเฮิร์ตซ์ ที่ได้จากวงจรรีโมทเสมือน แล้วนำออสซิลโลสโคปมาวัดและเปรียบเทียบค่าสัญญาณว่าตรงกลับสัญญาณเอาต์พุตของรีโมทหรือไม่

3. เมื่อได้ค่าสัญญาณที่ตรงกันแล้ว จึงนำไปทดสอบกลับโทรศัพท์มือถือซัมซุงว่าสามารถควบคุมโทรศัพท์นี้ได้ถูกต้องจริงหรือไม่ พร้อมจัดเก็บผลการทดลอง

4. ในขั้นตอนการประมวลผลภาพ นำกล้องเว็บแคมเชื่อมต่อเข้ากับบอร์ดราสเบอร์รี่พาย พร้อมทำการเขียนโปรแกรม

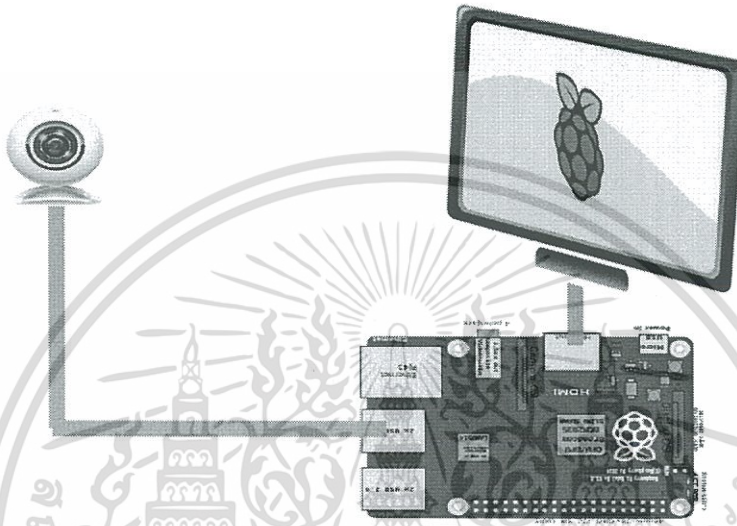
5. เมื่อทำสำเร็จทั้ง 4 ขั้นตอนจึงเริ่มทำการทดสอบระบบโดยรวมพร้อมจัดเก็บผลการทดลอง โดยนำกล้องเว็บแคมและบอร์ดราสเบอร์รี่พายเชื่อมต่อเข้ากับบอร์ดอาดูโนและวงจรรีโมทเสมือน พร้อมทดสอบด้วยการทำลักษณะท่าทางของฝ่ามือในรูปแบบต่างๆ แล้วนำไปทดสอบกับโทรศัพท์ว่าสามารถควบคุมได้ตามที่ต้องการหรือไม่

6. ในส่วนขั้นตอนการเก็บผลทางวิศวกรรมได้นำออสซิลโลสโคปมาวัดสัญญาณเอาต์พุตระหว่างสัญญาณของรีโมทโทรศัพท์ และสัญญาณของอุปกรณ์ พร้อมทั้งเปรียบเทียบค่าสัญญาณว่าผลลัพธ์ที่ได้นั้นมีค่าถูกต้องและเหมือนกันหรือไม่

บทที่ 4

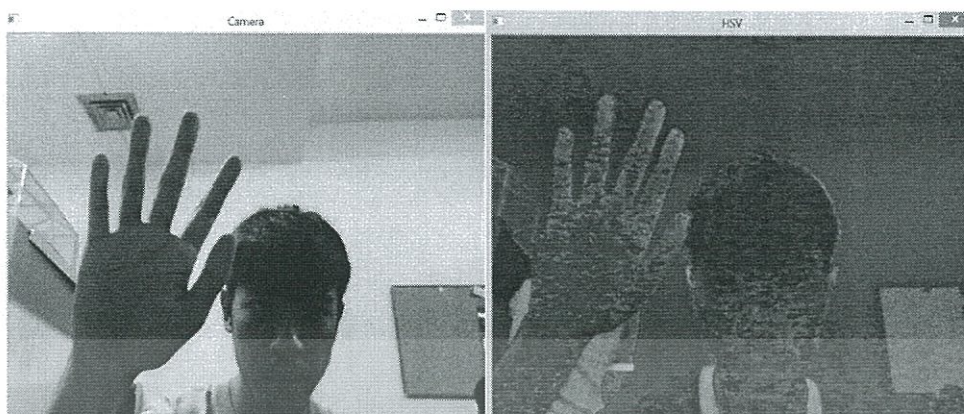
ผลการทดลอง

4.1 ผลการทดลองการตรวจจับภาพฝ่ามือ



รูปที่ 4.1 ตัวอย่างการต่อกล้องกับราสเบอร์รี่พายและจอมอนิเตอร์

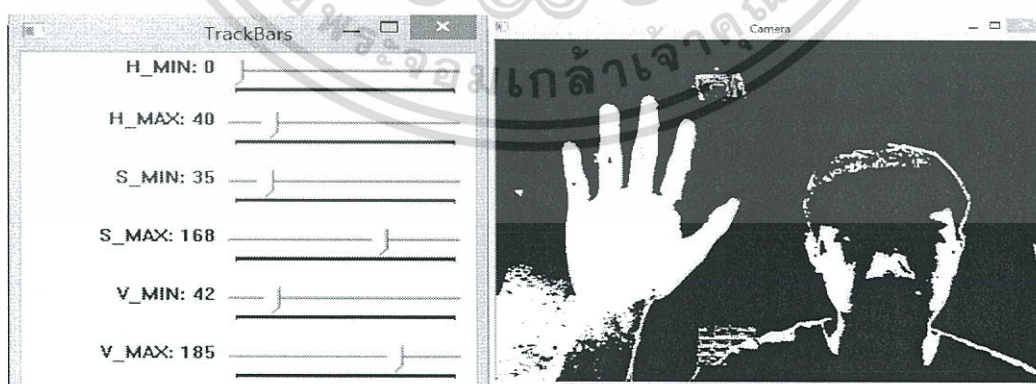
4.1.1 ในการทดลองตรวจจับภาพฝ่ามือ จะทำการต่อกล้องกับราสเบอร์รี่พายแล้วแสดงผลผ่านทางหน้าจอมอนิเตอร์ดังรูปที่ 4.1 จากนั้นทดลองจับภาพจากกล้องโดยใช้คำสั่ง `VideoCapture cap(0)` และแสดงภาพที่จับได้จากกล้องโดยใช้คำสั่ง `imshow("Camera", frame)` ดังรูปที่ 4.2 (ก) จากนั้นจะทำการแปลงภาพที่ได้เป็นภาพ HSV โดยใช้คำสั่ง `cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)` ของ OpenCV ซึ่งภาพ HSV นั้นเหมาะกับการประมวลผลภาพที่แยกวัตถุออกจากพื้นหลัง เนื่องจากค่าความสว่างของแสงมีผลกระทบกับภาพ HSV น้อยกว่าภาพ RGB ซึ่งแสดงดังรูปที่ 4.2 (ข)



(ก) ภาพอินพุตที่ได้จากกล้องโดยเป็นระบบสี RGB (ข) ภาพเอาต์พุตที่ได้จากการแปลงระบบสีภาพเป็น HSV

รูปที่ 4.2 การแปลงภาพจากระบบสี RGB เป็น HSV

4.1.2 จากนั้นหาค่าเทรซโฮลด์โดยการกำหนดขอบเขตของค่า H, S และ V จากแทรคบาร์ที่สร้างขึ้นจากคำสั่ง CreateTrackbars() เพื่อแยกภาพฝ่ามือออกจากพื้นหลัง โดยที่ค่า H คือค่าของสีที่ต้องการจะแยกออกจากพื้นหลัง, S คือค่าความเข้มของสี และ V คือค่าความสว่างของแสง ซึ่งในการทำงานของกระบวนการแยกภาพฝ่ามือนั้น จะต้องปรับค่า H ให้ตรงกับสีของฝ่ามือ ส่วนค่าของ S และ V นั้นจะขึ้นอยู่กับสถานะของแสงในแต่ละสถานที่นั้นๆ โดยการปรับค่า S และ V นั้นจะต้องเลือกให้อยู่ในช่วงที่ค่าความเข้มสีตรงกับสีของฝ่ามือ และค่าความเข้มของแสงตรงกับแสงที่ตกกระทบกับฝ่ามือ ดังรูปที่ 4.3



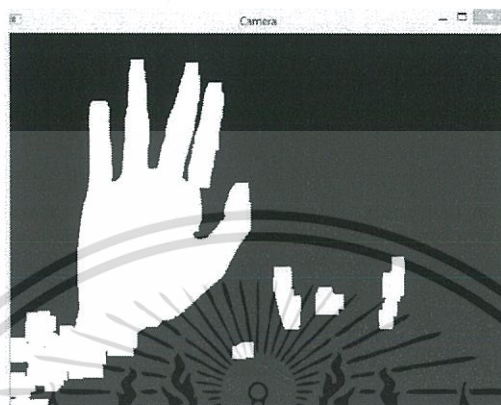
(ก) แสดงขอบเขตของค่า H, S และ V

(ข) ภาพฝ่ามือที่ได้หลังจากหาค่าเทรซโฮลด์

รูปที่ 4.3 การปรับค่าเทรซโฮลด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 จากนั้นนำภาพฝ่ามือที่ได้มาปรับปรุงคุณภาพผ่านกระบวนการเปลี่ยนแปลงโครงร่างของภาพโดยใช้คำสั่ง Erosion() และ Dilation() เพื่อลดสัญญาณรบกวนและวัตถุที่ไม่ได้สนใจออกจากเฟรมภาพเพื่อให้ง่ายต่อการประมวลผลภาพในขั้นตอนต่อไป ดังรูปที่ 4.4



รูปที่ 4.4 ภาพที่ผ่านกระบวนการเปลี่ยนแปลงโครงร่างของภาพ

4.1.4 หลังจากผ่านกระบวนการปรับปรุงคุณภาพแล้ว นำภาพฝ่ามือที่ได้ ไปหาเส้นคอนทัวร์และจุดเซนทรอยด์โดยใช้วิธีโมเมนต์เมธอดเพื่อใช้ในการตรวจจับการเคลื่อนไหวของฝ่ามือโดยการเปรียบเทียบพิกัดของฝ่ามือในแต่ละเฟรมภาพ หากฝ่ามือเคลื่อนที่ไปทางซ้ายกำหนดให้เป็นการเพิ่มช่องโทรทัศน์ หากฝ่ามือเคลื่อนที่ไปทางขวากำหนดให้เป็นการลดช่องโทรทัศน์ จากนั้นทำการหา Convex hull และ Convexity defects เพื่อหาจุดปลายของนิ้วมือ ดังรูปที่ 4.5



รูปที่ 4.5 ภาพที่ได้หลังจากการหาเส้นคอนทัวร์ จุดเซนทรอยด์ และจุดปลายของนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.5 จากนั้นนำภาพจุดปลายของนิ้วมือมาทำการนับจำนวนนิ้วโดยหาจุดที่อยู่เหนือจากจุดเซนทรอยด์แล้วทำการวนซ้ำจนครบทั้งเฟรมภาพเริ่มจากคอลัมน์ทางซ้ายไปขวา เมื่อนับจุดปลายของนิ้วมือครบทั้งเฟรมภาพแล้วทำการแสดงผลออกมาเป็นตัวเลขเพื่อใช้เป็นสัญลักษณ์ในการส่งงานโทรทัศน ดังรูปที่ 4.6



รูปที่ 4.6 แสดงตัวอย่างการนับจำนวนนิ้วมือ 3 และ 5 นิ้ว

4.2 ผลการทดลองการสร้างสัญญาณรีโมทเสมือน

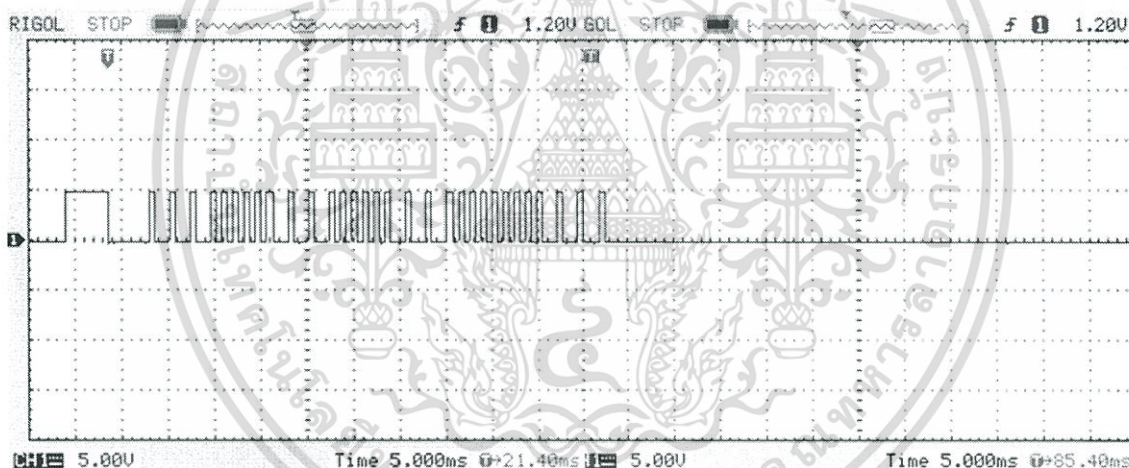
การสร้างสัญญาณรีโมทเสมือนเพื่อทำการควบคุมโทรทัศนเริ่มจากการเก็บข้อมูลสัญญาณรีโมทของปุ่มโทรทัศนแต่ละปุ่ม นำมาวิเคราะห์รูปแบบของแต่ละปุ่มเพื่อนำสัญญาณที่ได้มาสร้างสัญญาณพัลส์เลียนแบบและนำไปมอดูเลตกับสัญญาณคลื่นพาห้ที่ความถี่ 38 กิโลเฮิร์ตซ์ แล้วส่งสัญญาณผ่านหลอดอินฟราเรดเพื่อทำการควบคุมโทรทัศน โดยระบบที่กล่าวมาทั้งหมดจะรับคำสั่งในการควบคุมโทรทัศนผ่านรูปแบบของลักษณะมือที่บ่งชี้ถึงรูปแบบคำสั่งต่างๆ แล้วส่งข้อมูลคำสั่งมาให้ไมโครคอนโทรลเลอร์ทำการสร้างรีโมทเสมือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 ผลการทดลองสร้างสัญญาณพัลส์เลียนแบบสัญญาณรีโมททรทัศน์ด้วย ไมโครคอนโทรลเลอร์

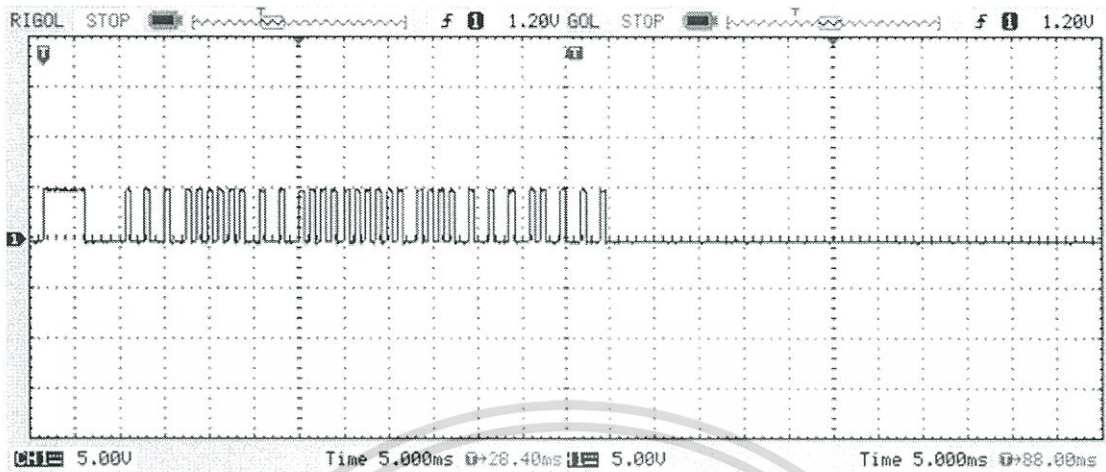
ในการสร้างสัญญาณพัลส์เลียนแบบรีโมททรทัศน์นั้น โดยในการทดลองจะใช้สัญญาณรีโมทต้นฉบับจากรีโมททรทัศน์ยี่ห้อซัมซุงทำการอ่านข้อมูลบิตและตีเลยจากข้อมูลที่ได้ทำการเก็บมาเพื่อนำมาสร้างพัลส์ให้มีตีเลยตรงกับชุดของสัญญาณนั้นและทำการสร้างโดยตัวไมโครคอนโทรลเลอร์เพื่อที่จะนำไปผ่านการมอดูเลตกับคลื่นพาห้ซึ่งเป็นสัญญาณพัลส์ความถี่ 38 kHz ซึ่งสัญญาณพัลส์เลียนแบบที่ได้นั้นมีรูปแบบตรงตามสัญญาณรีโมทมาตรฐานอ้างอิงจากหัวข้อที่ 2.4.3 สัญญาณรีโมทคอนโทรลมาตรฐานซัมซุง จากบทที่ 2

4.2.1.1 สัญญาณพัลส์เลียนแบบของปุ่มเพิ่มช่องรีโมททรทัศน์ยี่ห้อซัมซุงที่สร้างโดย ไมโครคอนโทรลเลอร์มีลักษณะของสัญญาณดังรูป 4.7



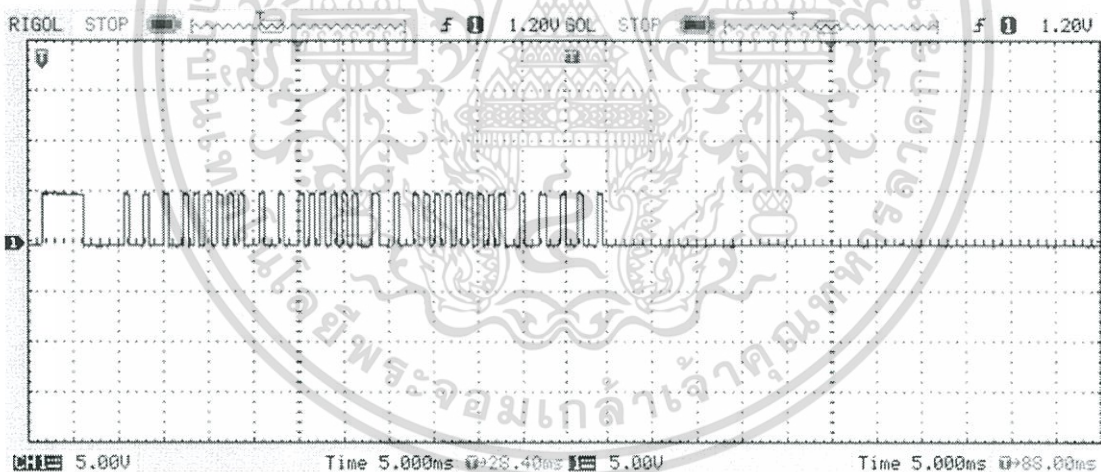
รูปที่ 4.7 สัญญาณพัลส์เลียนแบบของปุ่มเพิ่มช่องที่สร้างโดยไมโครคอนโทรลเลอร์

4.2.1.2 สัญญาณพัลส์เลียนแบบของปุ่มลดช่องรีโมททรทัศน์ยี่ห้อซัมซุงที่สร้างโดย ไมโครคอนโทรลเลอร์มีลักษณะของสัญญาณดังรูป 4.8



รูปที่ 4.8 สัญญาณพัลส์เลียนแบบของปั๊มลดช่องที่สร้างโดยไมโครคอนโทรลเลอร์

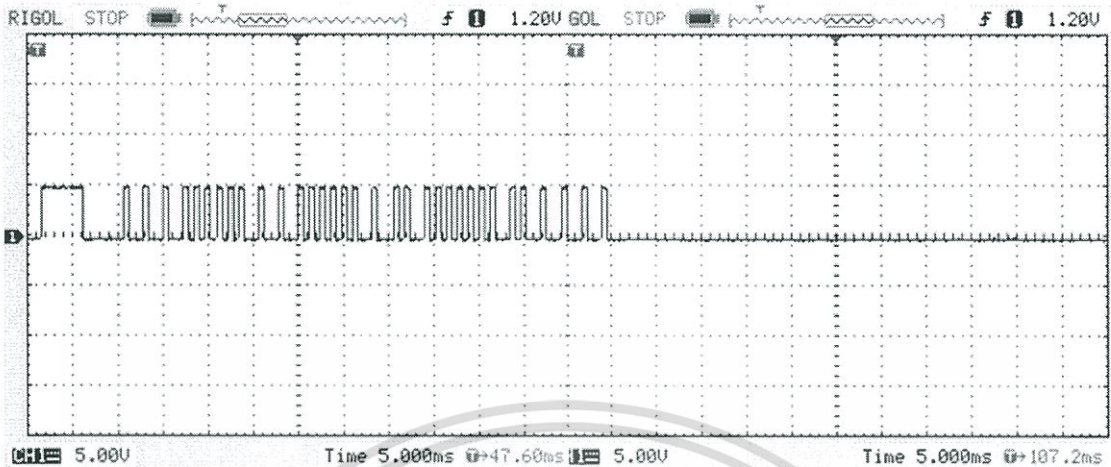
4.2.1.3 สัญญาณพัลส์เลียนแบบของปั๊มเพิ่มเสียงรีโมทโทรทัศน์ยี่ห้อซัมซุงที่สร้างโดยไมโครคอนโทรลเลอร์มีลักษณะของสัญญาณดังรูป 4.9



รูปที่ 4.9 รูปสัญญาณพัลส์ของปั๊มเพิ่มเสียงรีโมทโทรทัศน์ยี่ห้อซัมซุง

4.2.1.4 สัญญาณพัลส์เลียนแบบของปั๊มลดเสียงรีโมทโทรทัศน์ยี่ห้อซัมซุงที่สร้างโดยไมโครคอนโทรลเลอร์มีลักษณะของสัญญาณดังรูป 4.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 รูปสัญญาณพัลส์ของปั๊มลดเสียงรีโมทโทรทัศน์ยี่ห้อซัมซุง

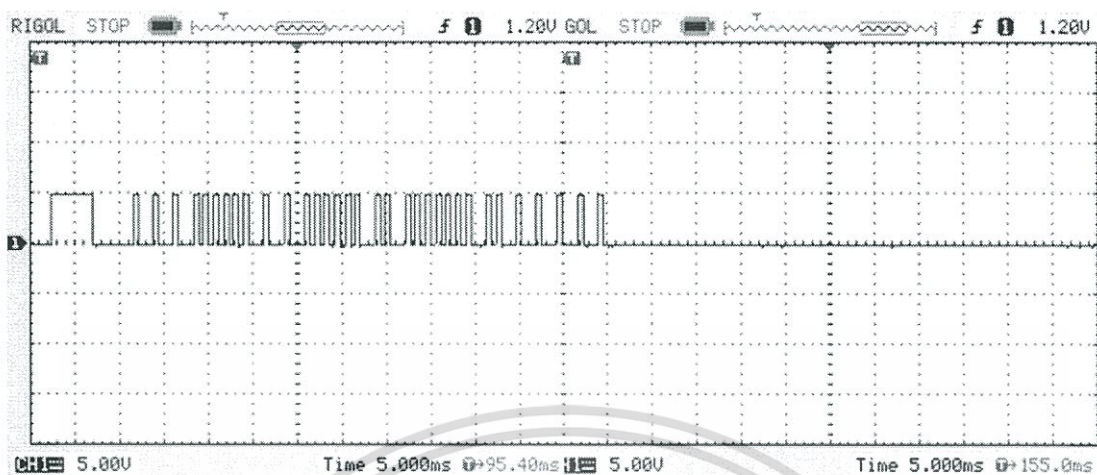
4.2.1.5 สัญญาณพัลส์เลียนแบบของปั๊มหมายเลข 1 รีโมทโทรทัศน์ยี่ห้อซัมซุงที่สร้างโดยไมโครคอนโทรลเลอร์มีลักษณะของสัญญาณดังรูปที่ 4.11



รูปที่ 4.11 สัญญาณพัลส์เลียนแบบของปั๊มหมายเลข 1

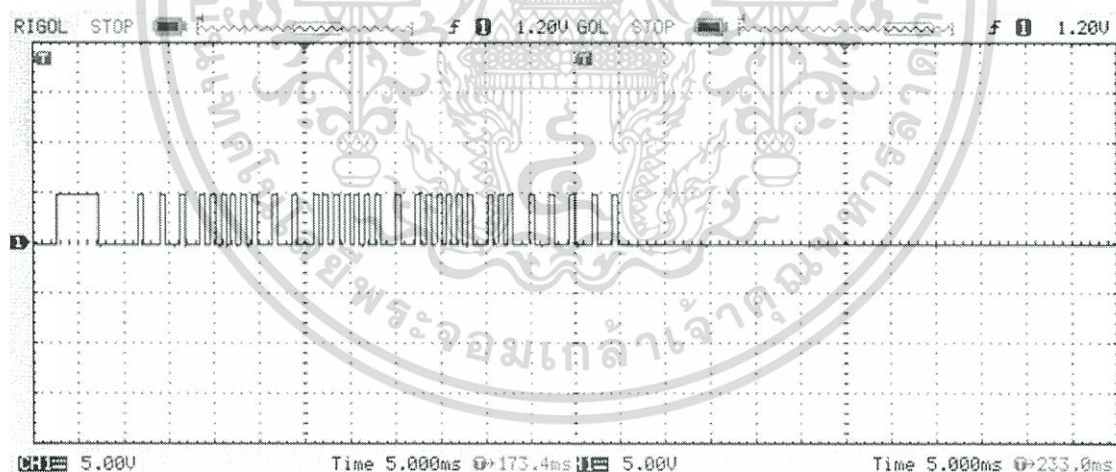
4.2.1.6 สัญญาณพัลส์เลียนแบบของปั๊มหมายเลข 2 รีโมทโทรทัศน์ยี่ห้อซัมซุงที่สร้างโดยไมโครคอนโทรลเลอร์มีลักษณะของสัญญาณดังรูปที่ 4.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 สัญญาณพัลส์ของปั๊มหมายเลข 2

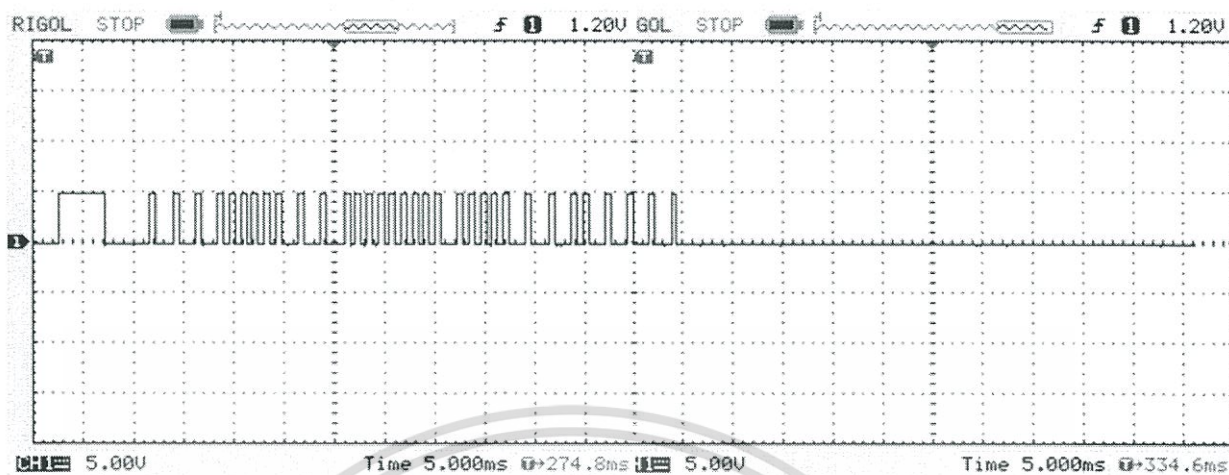
4.2.1.7 สัญญาณพัลส์เลียนแบบของปั๊มหมายเลข 3 รีโมททรานส์มิตเตอร์ที่ห่อหุ้มซึ่งที่สร้างโดยไมโครคอนโทรลเลอร์มีลักษณะของสัญญาณดังรูปที่ 4.13



รูปที่ 4.13 สัญญาณพัลส์ของปั๊มหมายเลข 3

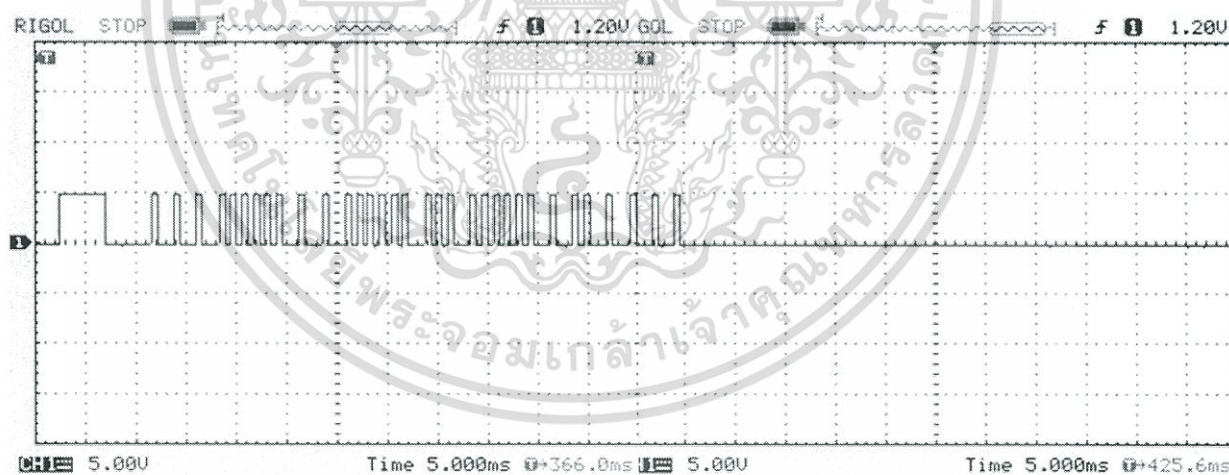
4.2.1.8 สัญญาณพัลส์เลียนแบบของปั๊มหมายเลข 4 รีโมททรานส์มิตเตอร์ที่ห่อหุ้มซึ่งที่สร้างโดยไมโครคอนโทรลเลอร์มีลักษณะของสัญญาณดังรูปที่ 4.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 สัญญาณพัลส์ของปั๊มหมายเลข 4

4.2.1.9 สัญญาณพัลส์เลียนแบบของปั๊มหมายเลข 5 รีโมททรานส์คีนีย์ห่อซมซุงที่สร้างโดยไมโครคอนโทรลเลอร์มีลักษณะของสัญญาณดังรูปที่ 4.15

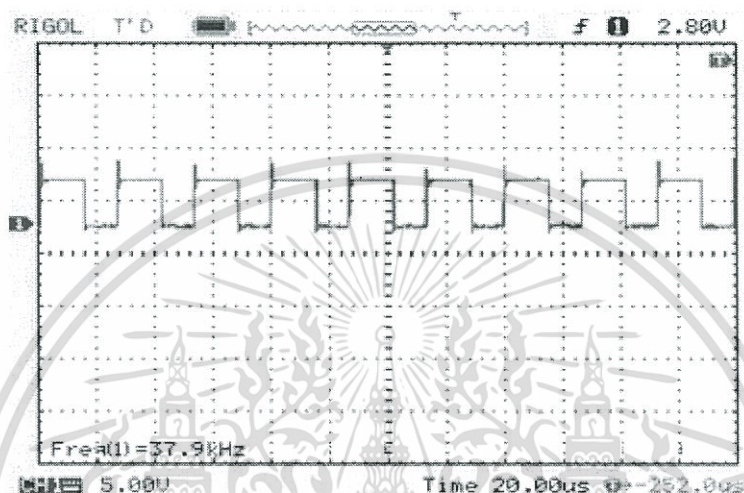


รูปที่ 4.15 รูปสัญญาณพัลส์ของปั๊มหมายเลข 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 ผลการทดลองสร้างสัญญาณพัลส์ความถี่ 38 kHz

ในปริิณญาณินพณธ์นั้ได้ทำการสร้างสัญญาณพัลส์ซึ่งเป็นคลื่นพาร์ควมถึเท่ากับ 38 kHz และมีแอมพลิจูด 5 Vpp ซึ่งถูกใช้เป็นสัญญาณความถี่มาตรฐานของสัญญาณโทรทัศน์ยี่ห้อซ้มซุง ซึ่งผลการทดลองจะได้สัญญาณความถี่ดังรูปที่ 4.16

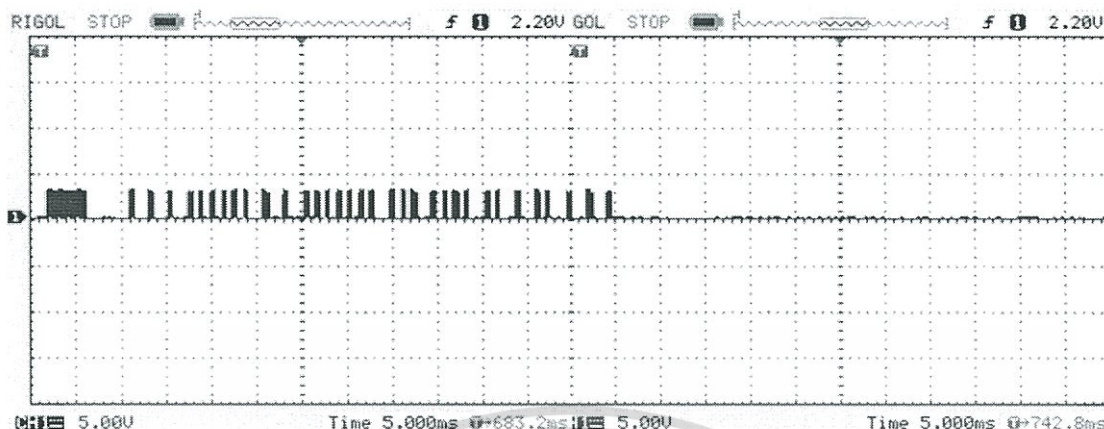


รูปที่ 4.16 สัญญาณพัลส์ที่สร้างจากวงจรสร้างสัญญาณพัลส์โดยใช้ไอซีเบอร์ 555

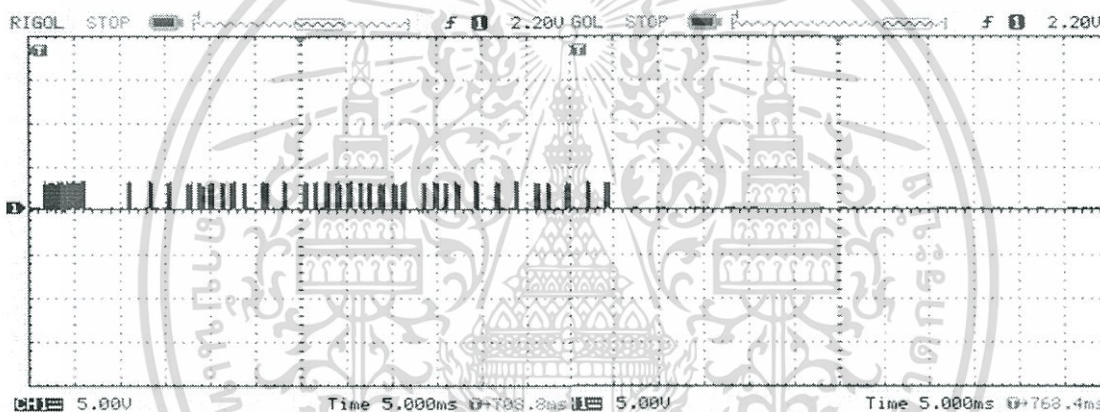
หลังจากที่ได้สร้างสัญญาณความถี่สำเร็จแล้วก็จะนำไปผ่านการมอดูเลตโดยไอซีเบอร์ 7408 เพื่อให้ได้สัญญาณเสมือนที่พร้อมส่งไปยังภาคของโทรทัศน์

4.2.3 ผลการทดลองการสร้างสัญญาณรีโมทเสมือน

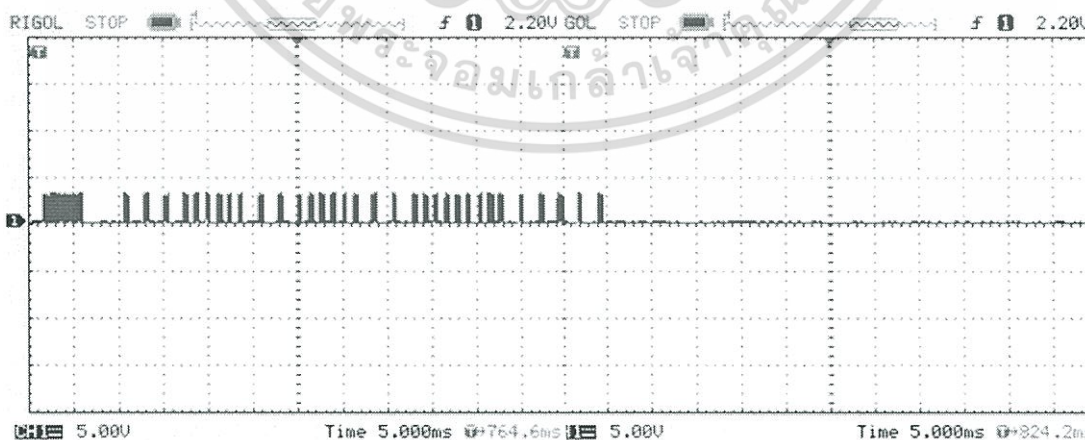
จากนั้นนำสัญญาณพัลส์เลียนแบบของการกดปุ่มที่สร้างโดยไมโครคอนโทรลเลอร์มาทำการมอดูเลตกับสัญญาณพัลส์ความถี่ 38 kHz จะได้สัญญาณรีโมทเสมือนซึ่งมีรูปแบบตรงตามสัญญาณรีโมทมาตรฐานอ้างอิงจากหัวข้อที่ 2.4.3 สัญญาณรีโมทคอนโทรลมาตรฐานซ้มซุงจากบทที่ 2 เพื่อส่งผ่านไดโอดเปล่งแสงไปทำการควบคุมโทรทัศน์ต่อไป ซึ่งผลสัญญาณรีโมทเสมือนของการกดปุ่มเพิ่มช่อง, ลดช่อง, เพิ่มเสียง, ลดเสียง และหมายเลข 1 ถึง 5 แสดงดังรูปที่ 4.17 – 4.25 ตามลำดับ



รูปที่ 4.17 สัญญาณรีโมทเสมือนของปุ่มเพิ่มช่องของรีโมทโทรทัศน์ยี่ห้อซัมซุง

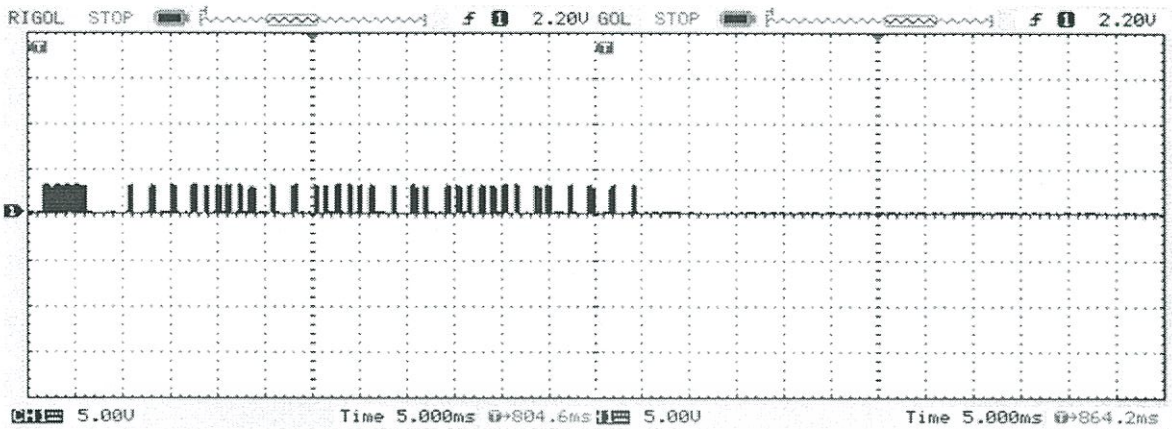


รูปที่ 4.18 สัญญาณรีโมทเสมือนของปุ่มลดช่องของรีโมทโทรทัศน์ยี่ห้อซัมซุง

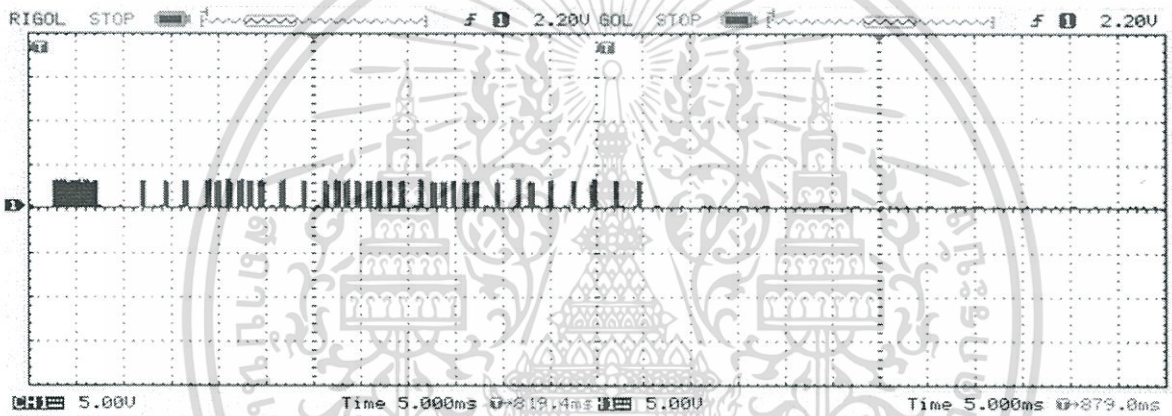


รูปที่ 4.19 สัญญาณรีโมทเสมือนของปุ่มเพิ่มเสียงของรีโมทโทรทัศน์ยี่ห้อซัมซุง

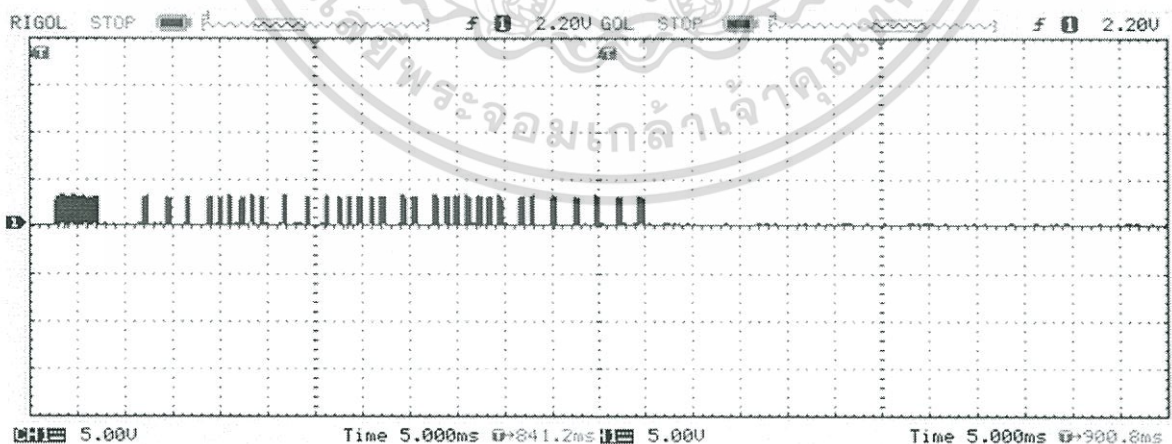
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 สัญญาณรีโมทเหมือนของปุ่มลดเสียงของรีโมทโทรทัศน์ยี่ห้อซัมซุง

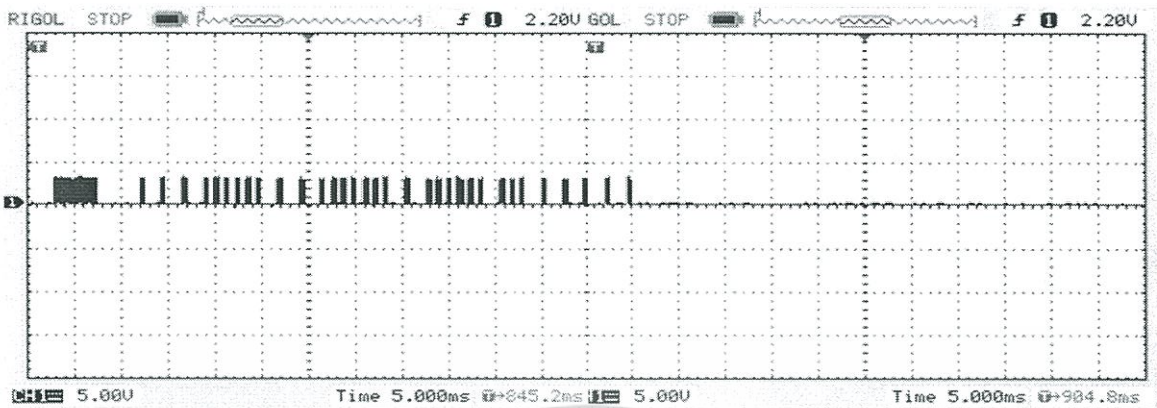


รูปที่ 4.21 สัญญาณรีโมทเหมือนของปุ่มหมายเลข 1 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง

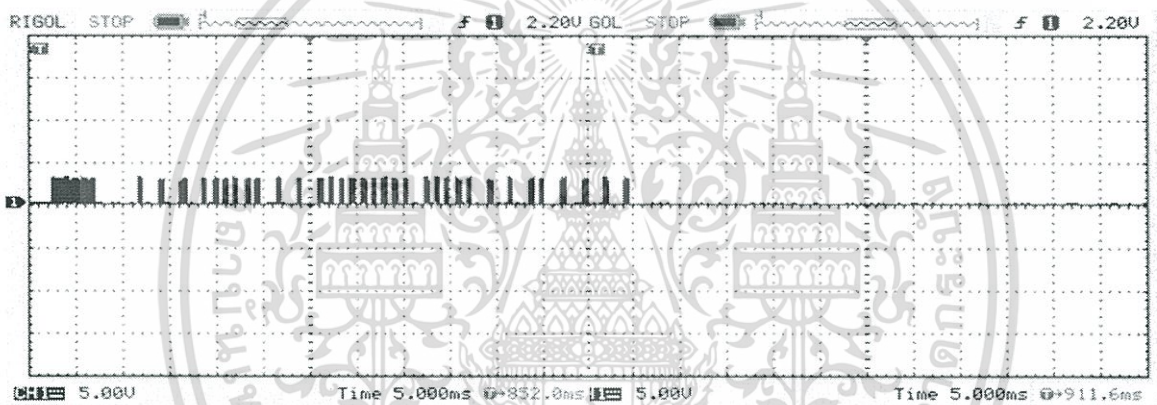


รูปที่ 4.22 สัญญาณรีโมทเหมือนของปุ่มหมายเลข 2 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง

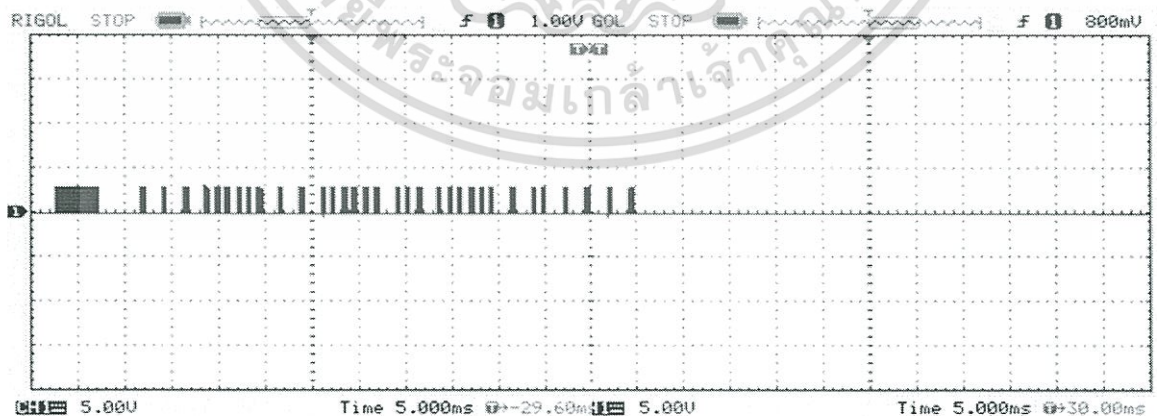
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 สัญญาณรีโมทเสมือนของปุ่มหมายเลข 3 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง



รูปที่ 4.24 สัญญาณรีโมทเสมือนของปุ่มหมายเลข 4 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง

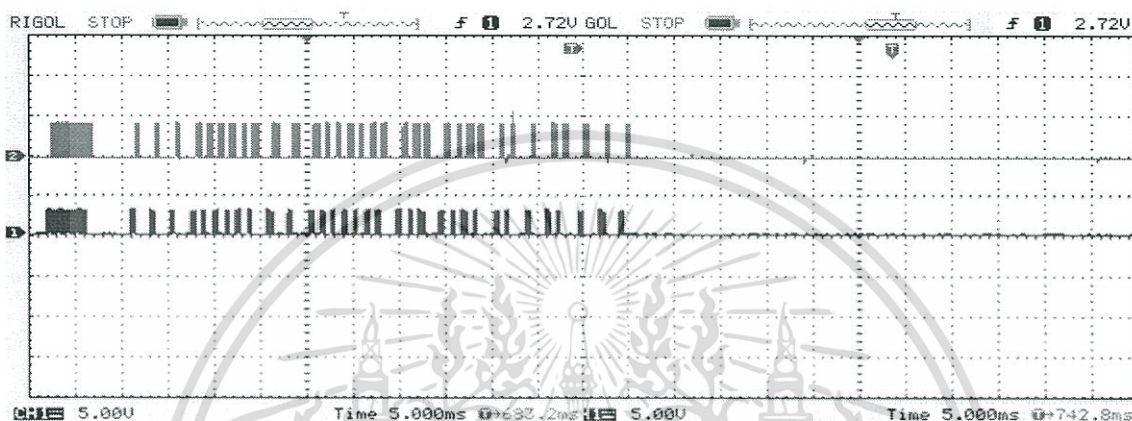


รูปที่ 4.25 สัญญาณรีโมทเสมือนของปุ่มหมายเลข 5 ของรีโมทโทรทัศน์ยี่ห้อซัมซุง

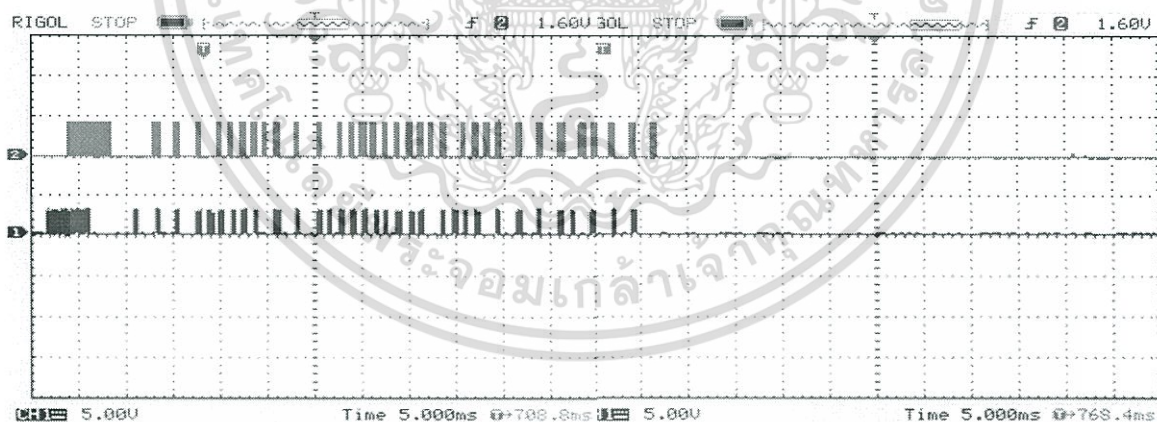
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 ผลการทดลองเปรียบเทียบสัญญาณรีโมทต้นฉบับกับสัญญาณรีโมทเสมือน

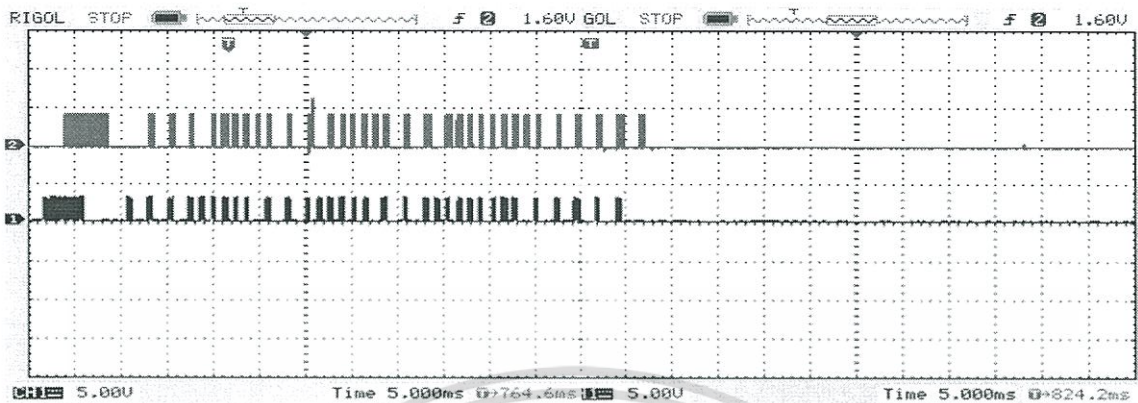
ผู้จัดทำได้ทำการเปรียบเทียบสัญญาณระหว่างสัญญาณรีโมทเสมือนที่สร้างขึ้นและสัญญาณที่ได้จากรีโมทโทรทัศน์ยี่ห้อซัมซุงซึ่งผลการเปรียบเทียบแสดงดังรูปที่ 4.26 – 4.34 ของการกดปุ่มเพิ่มช่อง, ลดช่อง, เพิ่มเสียง, ลดเสียง และหมายเลข 1 – 5 ตามลำดับ



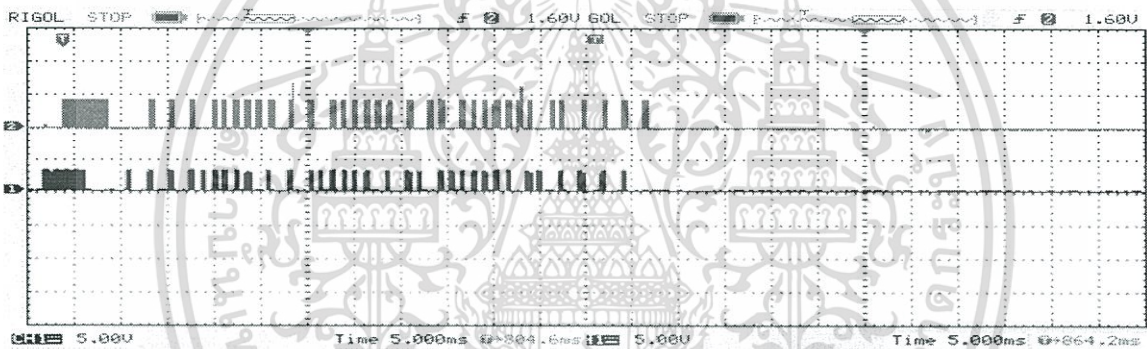
รูปที่ 4.26 การเปรียบเทียบสัญญาณปุ่มเพิ่มช่องโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)



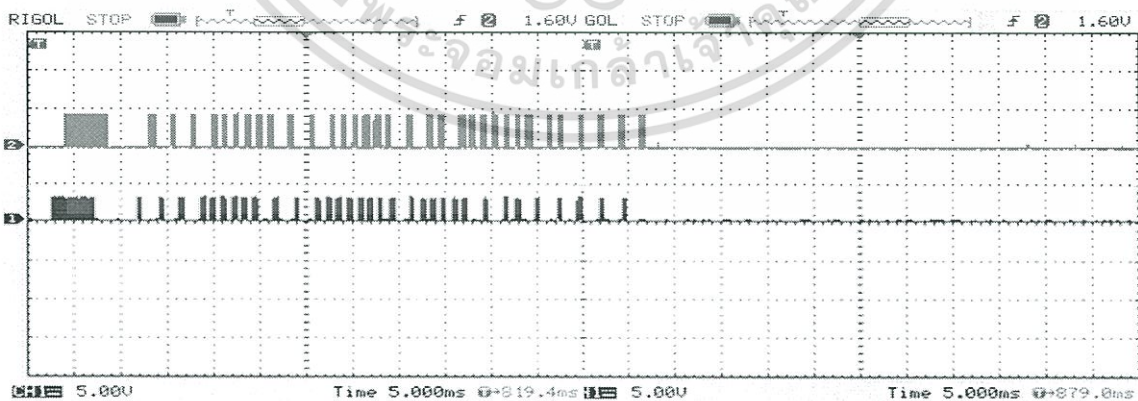
รูปที่ 4.27 การเปรียบเทียบสัญญาณปุ่มลดช่องโทรทัศน์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)



รูปที่ 4.28 การเปรียบเทียบสัญญาณป้อนเพิ่มเสียงโทรศัพท์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)

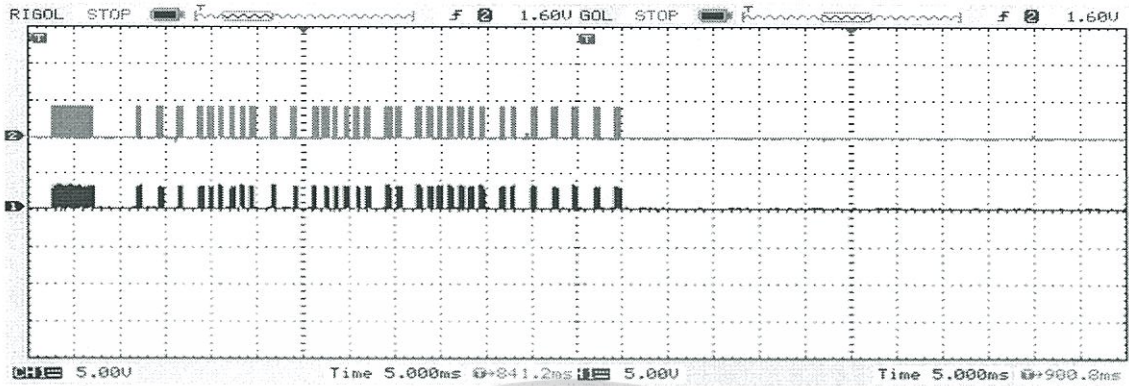


รูปที่ 4.29 การเปรียบเทียบสัญญาณป้อนลดเสียงโทรศัพท์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)

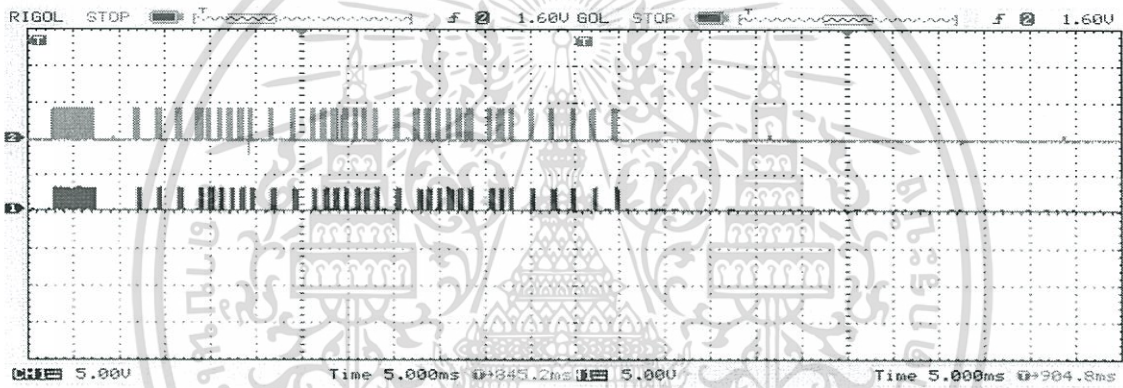


รูปที่ 4.30 การเปรียบเทียบสัญญาณป้อนหมายเลข 1 ของโทรศัพท์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)

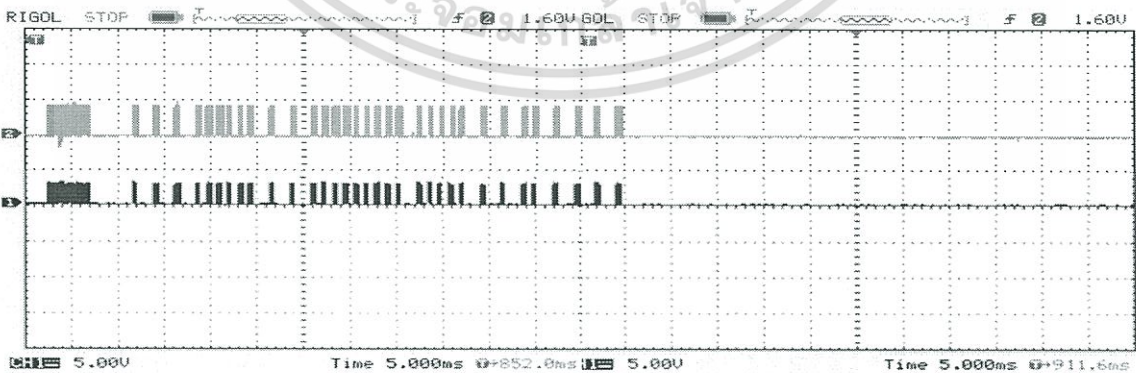
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.31 การเปรียบเทียบสัญญาณปุ่มหมายเลข 2 ของโทรศัพท์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)

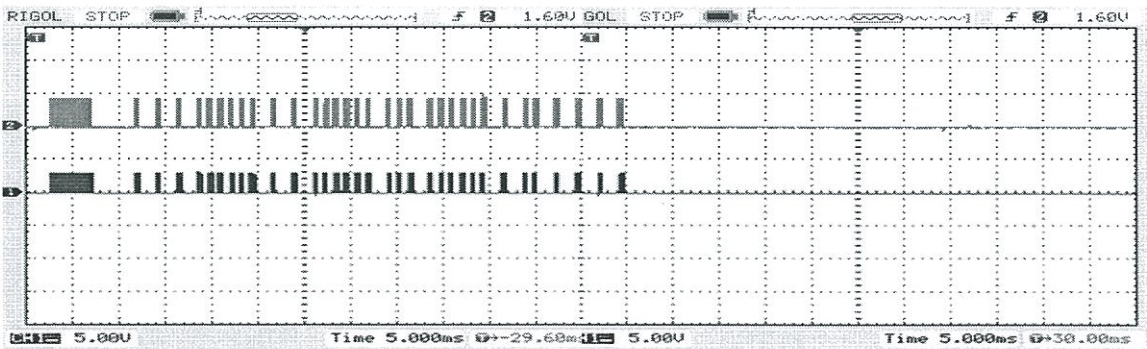


รูปที่ 4.32 การเปรียบเทียบสัญญาณปุ่มหมายเลข 3 ของโทรศัพท์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)



รูปที่ 4.33 การเปรียบเทียบสัญญาณปุ่มหมายเลข 4 ของโทรศัพท์ที่ได้จากสัญญาณรีโมทเสมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)

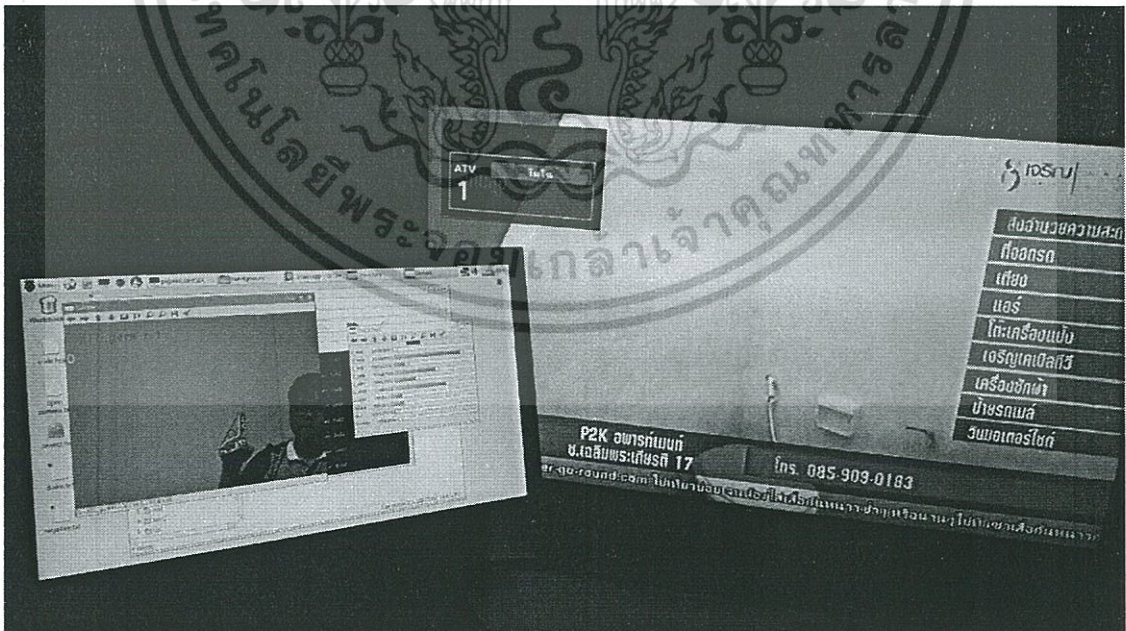
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.34 การเปรียบเทียบสัญญาณปุ่มหมายเลข 5 ของโทรศัพท์ที่ได้จากสัญญาณรีโมทเหมือนที่สร้างขึ้น (บน) และ สัญญาณรีโมทต้นฉบับ (ล่าง)

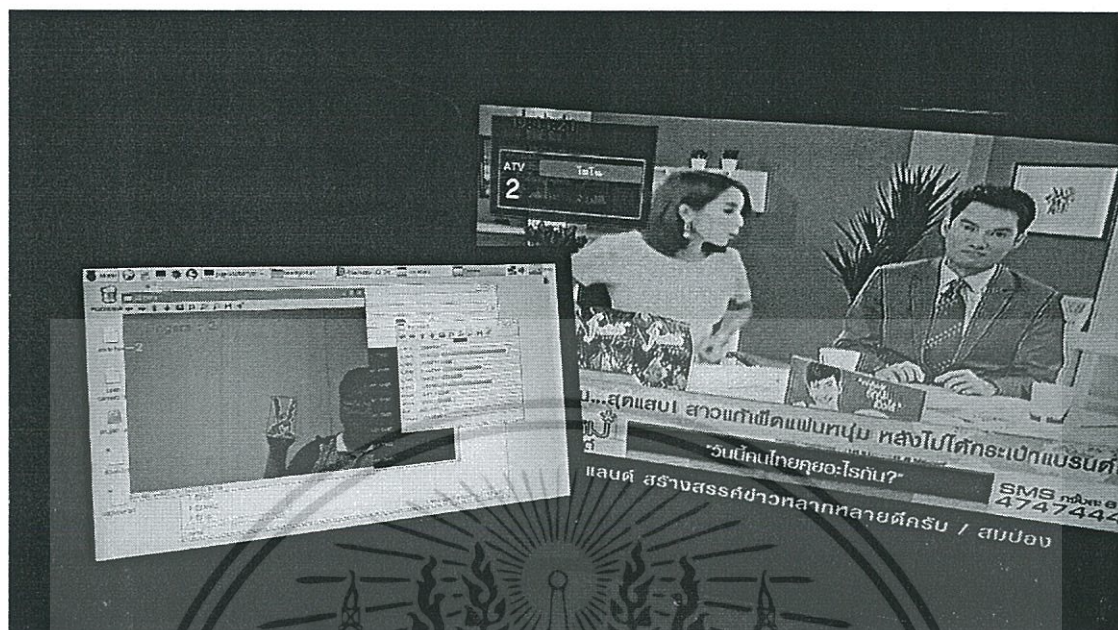
4.3 ผลการทดลองการควบคุมโทรศัพท์ด้วยลักษณะท่าทางมือ

จากการประมวลผลภาพฝ่ามือแต่ละลักษณะท่าทางของมือนั้น บอร์ดราสเบอร์รี่พายจะรับภาพฝ่ามือจากกล้องเว็บแคมมาทำการประมวลผลภาพเพื่อบ่งชี้ท่าทาง จากนั้นจะส่งบิตข้อมูลของคำสั่งที่สอดคล้องกับท่าทางของฝ่ามือไปยังบอร์ดไมโครคอนโทรลเลอร์เพื่อสร้างสัญญาณพัลส์เลียนแบบไปมอดูเลตกับคลื่นพาห้ซึ่งเป็นสัญญาณพัลส์ความถี่ 38 KHz เพื่อสร้างสัญญาณรีโมทเสมือนผ่านหลอดอินฟราเรดทำการควบคุมโทรศัพท์ ซึ่งผลลัพธ์การควบคุมโทรศัพท์แสดงดังรูปที่ 4.35 – 4.39 เมื่อท่าทางของฝ่ามือบ่งชี้ช่องหมายเลข 1 – 5 ตามลำดับ

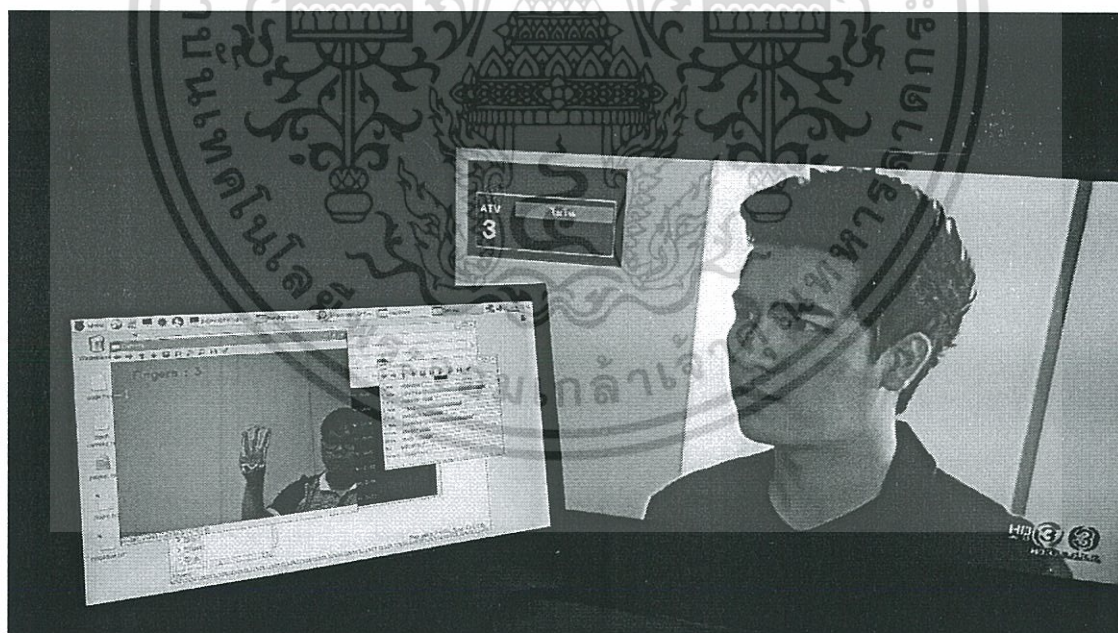


รูปที่ 4.35 การเปลี่ยนช่องโทรศัพท์ไปช่องหมายเลข 1 เมื่อท่าทางของฝ่ามือแสดงหมายเลข 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

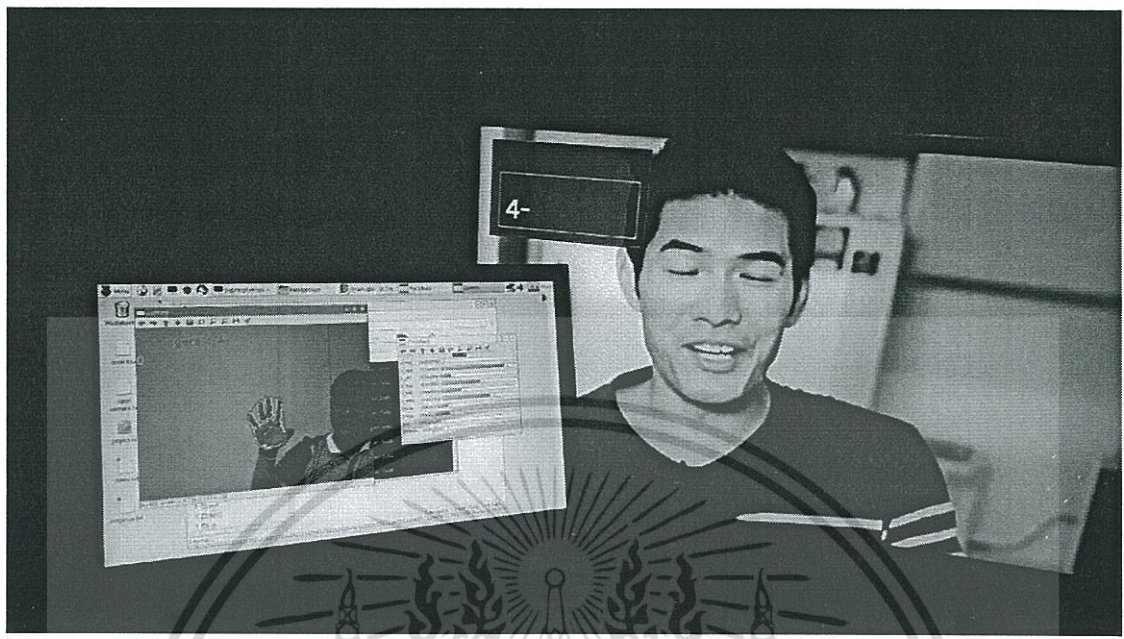


รูปที่ 4.36 การเปลี่ยนช่องโทรทัศน์ไปช่องหมายเลข 2 เมื่อทำท่าทางของฝ่ามือแสดงหมายเลข 2



รูปที่ 4.37 การเปลี่ยนช่องโทรทัศน์ไปช่องหมายเลข 3 เมื่อทำท่าทางของฝ่ามือแสดงหมายเลข 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.38 การเปลี่ยนช่องโทรทัศน์ไปช่องหมายเลข 4 เมื่อทำท่าทางของฝ่ามือแสดงหมายเลข 4



รูปที่ 4.39 การเปลี่ยนช่องโทรทัศน์ไปช่องหมายเลข 5 เมื่อทำท่าทางของฝ่ามือแสดงหมายเลข 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ปริญญานิพนธ์นี้นำเสนอระบบการประมวลผลภาพส่งผ่านไปยังไม่โครคอนโทรเลอร์เพื่อทำการเปลี่ยนช่องและปรับลดระดับเสียงของโทรทัศน์ โดยใช้ภาพท่าทางของมือ แทนการใช้รีโมทกดเปลี่ยนช่อง สามารถสรุปผลปริญญานิพนธ์ได้ดังนี้

1. ระบบสามารถตรวจจับฝ่ามือจากลักษณะท่าทางของมือได้โดยการรับภาพระบบสีจากกล้องเว็บแคมแล้วทำการประมวลผลด้วยบอร์ดราสเบอร์รี่พาย

2. ระบบสามารถตรวจจับทิศทางของฝ่ามือได้ทำให้สามารถประมวลผลได้ว่าจะเพิ่มช่องหรือลดช่องโดยการนำพิกัดตามแนวแกน x จากภาพเฟรมเก่าลบค่า x เฟรมปัจจุบันแล้วกำหนดขอบเขตของผลลัพธ์ที่ได้ว่าจะให้เอาท์พุตออกมาเป็นเพิ่มช่อง, ลดช่อง, ไม่เปลี่ยนแปลง

3. สามารถสร้างสัญญาณรีโมทเสมือนของรีโมทซัมซุง และเปลี่ยนช่องได้เสมือนรีโมทโทรทัศน์จริงๆ

4. ระบบสามารถเปลี่ยนช่องโทรทัศน์ได้โดยการรับค่าเอาท์พุตจากการประมวลผลภาพ (เพิ่มช่อง, ลดช่อง, ไม่เปลี่ยนแปลง, หมายเลขช่อง) มาทำการสร้างสัญญาณพัลส์ที่มีความถี่ 38kHz ผ่านอินฟราเรดไปยังโทรทัศน์เพื่อทำการควบคุมโทรทัศน์ได้ตามการเคลื่อนไหวของฝ่ามือ

5.2 ข้อเสนอแนะ

1. หลักการในการประมวลผลภาพในปริญญานิพนธ์นี้นั้นใช้การตรวจจับจากสีของฝ่ามือ ทำให้มีข้อเสียในการตรวจจับเพราะอาจจะมีสีของผิวหนังส่วนอื่นเข้ามาด้วยทำให้เกิดการผิดพลาดในการตรวจจับในบางครั้ง

2. ในการตรวจจับนั้นถ้าบริเวณที่มีแสงสว่างมากๆจะส่งผลให้มีการตรวจจับได้ยากขึ้นเนื่องจากความสว่างมากในบริเวณพื้นที่ๆใช้งานนั้นมีผลต่อค่าภาพระบบสี HSV ส่งผลให้กระบวนการในการทำเทรซโฮลด์แยกวัตถุออกจากฉากหลังได้ไม่สมบูรณ์จึงตรวจจับภาพไม่ได้

3. เพื่อไม่เกิดข้อจำกัดในการนำระบบนี้ไปใช้งานใช้กับโทรทัศน์รุ่นอื่นควรจัดทำสร้างวงจรเรียนรู้สัญญาณรีโมทพร้อมนำค่าที่เรียนรู้ไปใช้งานต่อกับระบบได้เลย

บรรณานุกรม

- [1] Gary Bradski and Adrian Kaehler. Learning OpenCV. United States of America : O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, 2008.
- [2] <http://docs.opencv.org/2.4/modules/refman.html>
- [3] <http://opencv-srf.blogspot.com/>
- [4] <http://docs.opencv.org/2.4/doc/tutorials/tutorials.html>
- [5] <http://www.tutorialspoint.com/csharp/>
- [6] https://www.youtube.com/playlist?list=PL36i_9J1VEAErCbxcIRtm5rlgk6BirupC
- [7] https://www.youtube.com/playlist?list=PLS1lqxOwNjOaE0gy_Q6tzU_QhZOB593Tv
- [8] <http://www.introprogramming.info>
- [9] ฉัตรชัย สุมามัลย์, การสื่อสารข้อมูลคอมพิวเตอร์และระบบเครือข่าย, กรุงเทพฯ : ไทยเจริญการพิมพ์, ไม้ระบุรีที่พิมพ์.
- [10] ชนวัฒน์ ศรีสอ้าน และสุทธิชัย มณีรัตน์รุ่งโรจน์, เอกสารประกอบการบรรยาย : การสื่อสารทางไกลและเครือข่าย, นครราชสีมา : มหาวิทยาลัยเทคโนโลยีสุรนารี, 2542.
- [11] เชลลี, การี บี, การสื่อสารข้อมูลระดับพื้นฐาน, กรุงเทพฯ : ทอมสัน, 2544.
- [12] http://www.one-2-win.com/radio_tech_02.htm
- [13] <http://devvertical.blogspot.com/2014/10/analog-transmission.html>
- [14] <http://praphut.blogspot.com/p/radio-frequency.html>
- [15] <https://www.youtube.com/watch?v=oH0ZkfFoeYU>
- [16] <https://www.youtube.com/watch?v=bSeFrPrqZ2A>
- [17] <https://www.youtube.com/watch?v=s5ROKdRUKul>
- [18] <http://stackoverflow.com/questions/9168785/hand-detection-using-opencv>
- [19] <http://anikettatipamula.blogspot.com/2012/02/hand-gesture-using-opencv.html>
- [20] <http://th.rs-online.com/web/generalDisplay.html?id=raspberrypi>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OpenCV Reference

createTrackbar

Creates a trackbar and attaches it to the specified window.

`createTrackbar(const string& trackbarname, const string& winname, int* value, int count, TrackbarCallback onChange=0, void* userdata=0)`

- `trackbarname` – Name of the created trackbar.
- `winname` – Name of the window that will be used as a parent of the created trackbar.
- `value` – Optional pointer to an integer variable whose value reflects the position of the slider. Upon creation, the slider position is defined by this variable.
- `count` – Maximal position of the slider. The minimal position is always 0.
- `onChange` – Pointer to the function to be called every time the slider changes position. This function should be prototyped as `void Foo(int,void*);`, where the first parameter is the trackbar position and the second parameter is the user data (see the next parameter). If the callback is the NULL pointer, no callbacks are called, but only value is updated.
- `userdata` – User data that is passed as is to the callback. It can be used to handle trackbar events without using global variables.

The function `createTrackbar` creates a trackbar (a slider or range control) with the specified name and range, assigns a variable value to be a position synchronized with the trackbar and specifies the callback function `onChange` to be called on the trackbar position change. The created trackbar is displayed in the specified window `winname`.

imshow

Displays an image in the specified window.

`imshow(const string& winname, InputArray mat)`

- `winname` – Name of the window.
- `image` – Image to be shown.

The function `imshow` displays an image in the specified window. If the window was created with the `CV_WINDOW_AUTOSIZE` flag, the image is shown with its original size.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Otherwise, the image is scaled to fit the window. The function may scale the image, depending on its depth:

- If the image is 8-bit unsigned, it is displayed as is.
- If the image is 16-bit unsigned or 32-bit integer, the pixels are divided by 256. That is, the value range [0,255*256] is mapped to [0,255].
- If the image is 32-bit floating-point, the pixel values are multiplied by 255. That is, the value range [0,1] is mapped to [0,255].

namedWindow

Creates a window.

`namedWindow(const string& winname, int flags=WINDOW_AUTOSIZE)`

- name – Name of the window in the window caption that may be used as a window identifier.
- flags – Flags of the window. The supported flags are:
 - WINDOW_NORMAL If this is set, the user can resize the window (no constraint).
 - WINDOW_AUTOSIZE If this is set, the window size is automatically adjusted to fit the displayed image (see `imshow()`), and you cannot change the window size manually.
 - WINDOW_OPENGL If this is set, the window will be created with OpenGL support.

waitKey

Waits for a pressed key.

`waitKey(int delay=0)`

- delay – Delay in milliseconds. 0 is the special value that means “forever”.

The function `waitKey` waits for a key event infinitely (when `\texttt{delay}\leq 0`) or for `delay` milliseconds, when it is positive. Since the OS has a minimum time between switching threads, the function will not wait exactly `delay` ms, it will wait at least `delay` ms, depending on what else is running on your computer at that time. It returns the code of the pressed key or -1 if no key was pressed before the specified time had elapsed.

Imread

- filename – Name of file to be loaded.
- flags – Flags specifying the color type of a loaded image:
 - CV_LOAD_IMAGE_ANYDEPTH - If set, return 16-bit/32-bit image when the input has the corresponding depth, otherwise convert it to 8-bit.
 - CV_LOAD_IMAGE_COLOR - If set, always convert image to the color one
 - CV_LOAD_IMAGE_GRAYSCALE - If set, always convert image to the grayscale one
 - >0 Return a 3-channel color image.
 - =0 Return a grayscale image.
 - <0 Return the loaded image as is (with alpha channel).

VideoCapture

class VideoCapture

Class for video capturing from video files or cameras. The class provides C++ API for capturing video from cameras or for reading video files.

Erode

Erodes an image by using a specific structuring element.

`erode(InputArray src, OutputArray dst, InputArray kernel, Point anchor=Point(-1,-1), int iterations=1, int borderType=BORDER_CONSTANT, const Scalar& borderValue=morphologyDefaultBorderValue())`

- src – input image; the number of channels can be arbitrary, but the depth should be one of CV_8U, CV_16U, CV_16S, CV_32F or CV_64F.
- dst – output image of the same size and type as src.
- element – structuring element used for erosion; if element=Mat(), a 3 x 3 rectangular structuring element is used.
- anchor – position of the anchor within the element; default value (-1, -1) means that the anchor is at the element center.
- iterations – number of times erosion is applied.
- borderType – pixel extrapolation method (see borderInterpolate() for details).
- borderValue – border value in case of a constant border (see createMorphologyFilter() for details).

Dilate

Dilates an image by using a specific structuring element.

`dilate(InputArray src, OutputArray dst, InputArray kernel, Point anchor=Point(-1,-1), int iterations=1, int borderType=BORDER_CONSTANT, const Scalar& borderValue=morphologyDefaultBorderValue())`

- `src` – input image; the number of channels can be arbitrary, but the depth should be one of `CV_8U`, `CV_16U`, `CV_16S`, `CV_32F` or `CV_64F`.
- `dst` – output image of the same size and type as `src`.
- `element` – structuring element used for dilation; if `element=Mat()`, a 3×3 rectangular structuring element is used.
- `anchor` – position of the anchor within the element; default value `(-1, -1)` means that the anchor is at the element center.
- `iterations` – number of times dilation is applied.
- `borderType` – pixel extrapolation method (see `borderInterpolate()` for details).
- `borderValue` – border value in case of a constant border (see `createMorphologyFilter()` for details).

cvtColor

Converts an image from one color space to another.

`cvtColor(InputArray src, OutputArray dst, int code, int dstCn=0)`

- `src` – input image: 8-bit unsigned, 16-bit unsigned (`CV_16UC...`), or single-precision floating-point.
- `dst` – output image of the same size and depth as `src`.
- `code` – color space conversion code (see the description below).
- `dstCn` – number of channels in the destination image; if the parameter is 0, the number of the channels is derived automatically from `src` and `code`.

The function converts an input image from one color space to another. In case of a transformation to-from RGB color space, the order of the channels should be specified explicitly (RGB or BGR). Note that the default color format in OpenCV is often referred to as RGB but it is actually BGR (the bytes are reversed). So the first byte in a standard (24-bit) color image will be an 8-bit Blue component, the second

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

byte will be Green, and the third byte will be Red. The fourth, fifth, and sixth bytes would then be the second pixel (Blue, then Green, then Red), and so on.

The conventional ranges for R, G, and B channel values are:

- 0 to 255 for CV_8U images
- 0 to 65535 for CV_16U images
- 0 to 1 for CV_32F images

findContours

Finds contours in a binary image.

`findContours(InputOutputArray image, OutputArrayOfArrays contours, OutputArray hierarchy, int mode, int method, Point offset=Point())`

- `image` – Source, an 8-bit single-channel image. Non-zero pixels are treated as 1's. Zero pixels remain 0's, so the image is treated as binary. You can use `compare()`, `inRange()`, `threshold()`, `adaptiveThreshold()`, `Canny()`, and others to create a binary image out of a grayscale or color one. The function modifies the image while extracting the contours. If mode equals to `CV_RETR_CCOMP` or `CV_RETR_FLOODFILL`, the input can also be a 32-bit integer image of labels (`CV_32SC1`).
- `contours` – Detected contours. Each contour is stored as a vector of points.
- `hierarchy` – Optional output vector, containing information about the image topology. It has as many elements as the number of contours. For each *i*-th contour `contours[i]`, the elements `hierarchy[i][0]`, `hierarchy[i][1]`, `hierarchy[i][2]`, and `hierarchy[i][3]` are set to 0-based indices in contours of the next and previous contours at the same hierarchical level, the first child contour and the parent contour, respectively. If for the contour *i* there are no next, previous, parent, or nested contours, the corresponding elements of `hierarchy[i]` will be negative.
- `mode` – Contour retrieval mode (if you use Python see also a note below).
 - `CV_RETR_EXTERNAL` retrieves only the extreme outer contours. It sets `hierarchy[i][2]=hierarchy[i][3]=-1` for all the contours.
 - `CV_RETR_LIST` retrieves all of the contours without establishing any hierarchical relationships.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- CV_RETR_CCOMP retrieves all of the contours and organizes them into a two-level hierarchy. At the top level, there are external boundaries of the components. At the second level, there are boundaries of the holes. If there is another contour inside a hole of a connected component, it is still put at the top level.
- CV_RETR_TREE retrieves all of the contours and reconstructs a full hierarchy of nested contours. This full hierarchy is built and shown in the OpenCV contours.c demo.
- method – Contour approximation method (if you use Python see also a note below).
 - CV_CHAIN_APPROX_NONE stores absolutely all the contour points. That is, any 2 subsequent points (x1,y1) and (x2,y2) of the contour will be either horizontal, vertical or diagonal neighbors, that is, $\max(\text{abs}(x1-x2), \text{abs}(y2-y1))=1$.
 - CV_CHAIN_APPROX_SIMPLE compresses horizontal, vertical, and diagonal segments and leaves only their end points. For example, an up-right rectangular contour is encoded with 4 points.
 - CV_CHAIN_APPROX_TC89_L1, CV_CHAIN_APPROX_TC89_KCOS applies one of the flavors of the Teh-Chin chain approximation algorithm. See [TehChin89] for details.
- offset – Optional offset by which every contour point is shifted. This is useful if the contours are extracted from the image ROI and then they should be analyzed in the whole image context.

drawContours

Draws contours outlines or filled contours.

```
drawContours(InputOutputArray image, InputArrayOfArrays contours, int contourIdx,
const Scalar& color, int thickness=1, int lineType=8, InputArray hierarchy=noArray(),
int maxLevel=INT_MAX, Point offset=Point() )
```

- image – Destination image.
- contours – All the input contours. Each contour is stored as a point vector.
- contourIdx – Parameter indicating a contour to draw. If it is negative, all the contours are drawn.

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- color – Color of the contours.
- thickness – Thickness of lines the contours are drawn with. If it is negative (for example, thickness=CV_FILLED), the contour interiors are drawn.
- lineType – Line connectivity. See line() for details.
- hierarchy – Optional information about hierarchy. It is only needed if you want to draw only some of the contours (see maxLevel).
- maxLevel – Maximal level for drawn contours. If it is 0, only the specified contour is drawn. If it is 1, the function draws the contour(s) and all the nested contours. If it is 2, the function draws the contours, all the nested contours, all the nested-to-nested contours, and so on. This parameter is only taken into account when there is hierarchy available.
- offset – Optional contour shift parameter. Shift all the drawn contours by the specified (dx,dy) .
- contour – Pointer to the first contour.
- externalColor – Color of external contours.
- holeColor – Color of internal contours (holes).

convexHull

Finds the convex hull of a point set.

convexHull(InputArray points, OutputArray hull, bool clockwise=false, bool returnPoints=true)

- points – Input 2D point set, stored in std::vector or Mat.
- hull – Output convex hull. It is either an integer vector of indices or vector of points. In the first case, the hull elements are 0-based indices of the convex hull points in the original array (since the set of convex hull points is a subset of the original point set). In the second case, hull elements are the convex hull points themselves.
- hull_storage – Output memory storage in the old API (cvConvexHull2 returns a sequence containing the convex hull points or their indices).
- clockwise – Orientation flag. If it is true, the output convex hull is oriented clockwise. Otherwise, it is oriented counter-clockwise. The assumed coordinate system has its X axis pointing to the right, and its Y axis pointing upwards.

- orientation – Convex hull orientation parameter in the old API, CV_CLOCKWISE or CV_COUNTERCLOCKWISE.
- returnPoints – Operation flag. In case of a matrix, when the flag is true, the function returns convex hull points. Otherwise, it returns indices of the convex hull points. When the output array is std::vector, the flag is ignored, and the output depends on the type of the vector: std::vector<int> implies returnPoints=true, std::vector<Point> implies returnPoints=false.

convexityDefects

Finds the convexity defects of a contour.

convexityDefects(InputArray contour, InputArray convexhull, OutputArray convexityDefects)

- contour – Input contour.
- convexhull – Convex hull obtained using convexHull() that should contain indices of the contour points that make the hull.
- convexityDefects – The output vector of convexity defects. In C++ and the new Python/Java interface each convexity defect is represented as 4-element integer vector (a.k.a. cv::Vec4i): (start_index, end_index, farthest_pt_index, fixpt_depth), where indices are 0-based indices in the original contour of the convexity defect beginning, end and the farthest point, and fixpt_depth is fixed-point approximation (with 8 fractional bits) of the distance between the farthest contour point and the hull. That is, to get the floating-point value of the depth will be fixpt_depth/256.0. In C interface convexity defect is represented by CvConvexityDefect structure - see below.
- storage – Container for the output sequence of convexity defects. If it is NULL, the contour or hull (in that order) storage is used.

General-purpose single bipolar timers

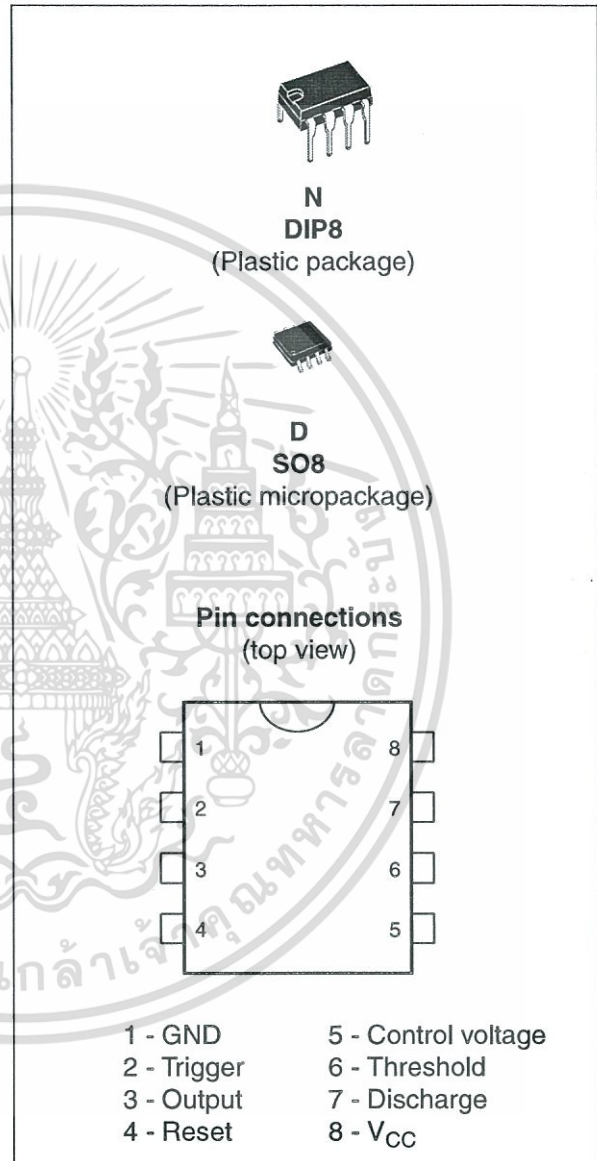
Features

- Low turn-off time
- Maximum operating frequency greater than 500 kHz
- Timing from microseconds to hours
- Operates in both astable and monostable modes
- Output can source or sink up to 200 mA
- Adjustable duty cycle
- TTL compatible
- Temperature stability of 0.005% per °C

Description

The NE555 monolithic timing circuit is a highly stable controller capable of producing accurate time delays or oscillation. In the time delay mode of operation, the time is precisely controlled by one external resistor and capacitor. For a stable operation as an oscillator, the free running frequency and the duty cycle are both accurately controlled with two external resistors and one capacitor.

The circuit may be triggered and reset on falling waveforms, and the output structure can source or sink up to 200 mA.



1 Schematic diagrams

Figure 1. Block diagram

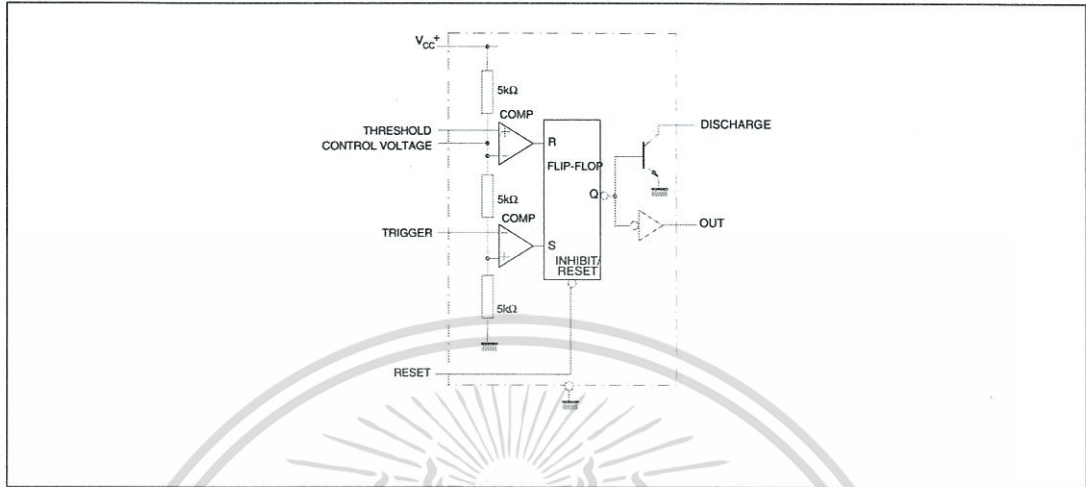
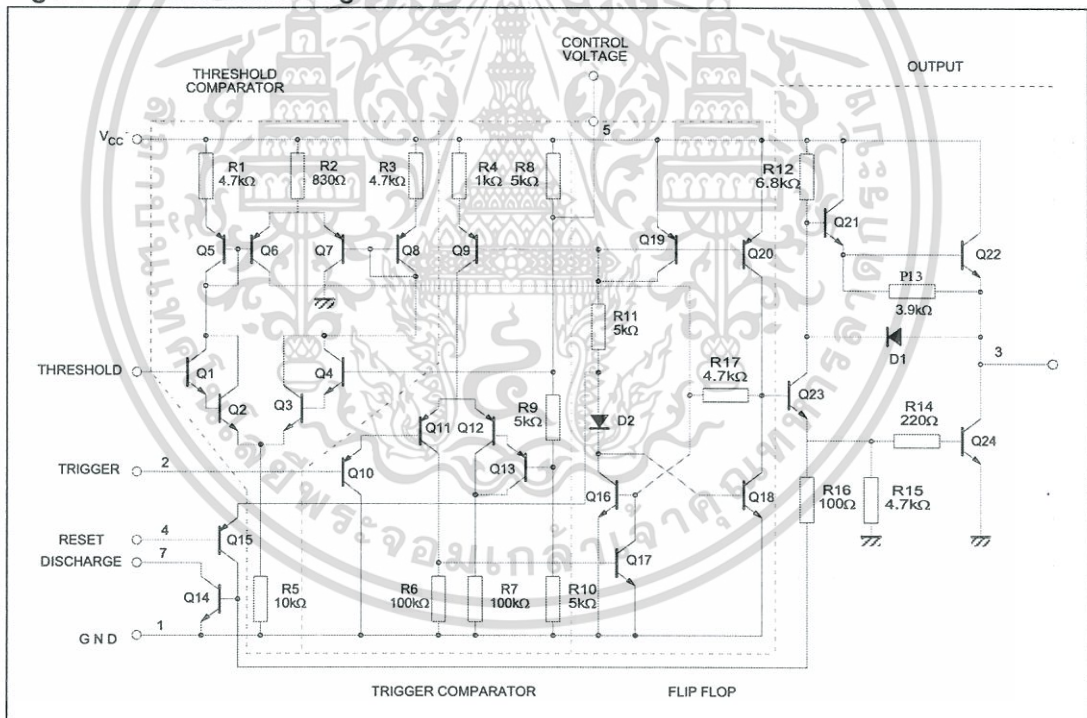


Figure 2. Schematic diagram



2 Absolute maximum ratings and operating conditions

Table 1. Absolute maximum ratings

Symbol	Parameter	Value	Unit
V_{CC}	Supply voltage	18	V
I_{OUT}	Output current (sink & source)	± 225	mA
R_{thja}	Thermal resistance junction to ambient ⁽¹⁾		$^{\circ}\text{C}/\text{W}$
	DIP8 SO-8	85 125	
R_{thjc}	Thermal resistance junction to case ⁽¹⁾		$^{\circ}\text{C}/\text{W}$
	DIP8 SO-8	41 40	
ESD	Human body model (HBM) ⁽²⁾	1000	V
	Machine model (MM) ⁽³⁾	100	
	Charged device model (CDM) ⁽⁴⁾	1500	
	Latch-up immunity	200	mA
T_{LEAD}	Lead temperature (soldering 10 seconds)	260	$^{\circ}\text{C}$
T_j	Junction temperature	150	$^{\circ}\text{C}$
T_{stg}	Storage temperature range	-65 to 150	$^{\circ}\text{C}$

1. Short-circuits can cause excessive heating. These values are typical.
2. Human body model: a 100 pF capacitor is charged to the specified voltage, then discharged through a 1.5k Ω resistor between two pins of the device. This is done for all couples of connected pin combinations while the other pins are floating.
3. Machine model: a 200 pF capacitor is charged to the specified voltage, then discharged directly between two pins of the device with no external series resistor (internal resistor < 5 Ω). This is done for all couples of connected pin combinations while the other pins are floating.
4. Charged device model: all pins and the package are charged together to the specified voltage and then discharged directly to the ground through only one pin. This is done for all pins.

Table 2. Operating conditions

Symbol	Parameter	Value	Unit
V_{CC}	Supply voltage		V
	NE555	4.5 to 16	
	SA555 SE555	4.5 to 16 4.5 to 18	
V_{th} , V_{trig} , V_{cl} , V_{reset}	Maximum input voltage	V_{CC}	V
I_{OUT}	Output current (sink and source)	± 200	mA
T_{oper}	Operating free air temperature range		$^{\circ}\text{C}$
	NE555	0 to 70	
	SA555 SE555	-40 to 105 -55 to 125	

3 Electrical characteristics

Table 3. $T_{amb} = +25^{\circ}C$, $V_{CC} = +5V$ to $+15V$ (unless otherwise specified)

Symbol	Parameter	SE555			NE555 - SA555			Unit	
		Min.	Typ.	Max.	Min.	Typ.	Max.		
I_{CC}	Supply current ($R_L = \infty$)							mA	
	Low state $V_{CC} = +5V$		3	5		3	6		
	$V_{CC} = +15V$		10	12		10	15		
	High state $V_{CC} = +5V$		2			2			
	Timing error (monostable) ($R_A = 2k\Omega$ to $100k\Omega$, $C = 0.1\mu F$)							%	
	Initial accuracy ⁽¹⁾		0.5	2		1	3		
	Drift with temperature		30	100		50			ppm/ $^{\circ}C$
	Drift with supply voltage		0.05	0.2		0.1	0.5		%/V
	Timing error (astable) ($R_A, R_B = 1k\Omega$ to $100k\Omega$, $C = 0.1\mu F$, $V_{CC} = +15V$)							%	
	Initial accuracy ⁽¹⁾		1.5			2.25			
	Drift with temperature		90			150			ppm/ $^{\circ}C$
	Drift with supply voltage		0.15			0.3			%/V
V_{CL}	Control voltage level							V	
	$V_{CC} = +15V$	9.6	10	10.4	9	10	11		
	$V_{CC} = +5V$	2.9	3.33	3.8	2.6	3.33	4		
V_{th}	Threshold voltage							V	
	$V_{CC} = +15V$	9.4	10	10.6	8.8	10	11.2		
	$V_{CC} = +5V$	2.7	3.33	4	2.4	3.33	4.2		
I_{th}	Threshold current ⁽²⁾		0.1	0.25		0.1	0.25	μA	
V_{trig}	Trigger voltage							V	
	$V_{CC} = +15V$	4.8	5	5.2	4.5	5	5.6		
	$V_{CC} = +5V$	1.45	1.67	1.9	1.1	1.67	2.2		
I_{trig}	Trigger current ($V_{trig} = 0V$)		0.5	0.9		0.5	2.0	μA	
V_{reset}	Reset voltage ⁽³⁾	0.4	0.7	1	0.4	0.7	1	V	
I_{reset}	Reset current							mA	
	$V_{reset} = +0.4V$		0.1	0.4		0.1	0.4		
	$V_{reset} = 0V$		0.4	1		0.4	1.5		
V_{OL}	Low level output voltage							V	
	$V_{CC} = +15V$ $I_{O(sink)} = 10mA$		0.1	0.15		0.1	0.25		
	$I_{O(sink)} = 50mA$		0.4	0.5		0.4	0.75		
	$I_{O(sink)} = 100mA$		2	2.2		2	2.5		
	$I_{O(sink)} = 200mA$		2.5			2.5			
	$V_{CC} = +5V$ $I_{O(sink)} = 8mA$		0.1	0.25		0.3	0.4		
$I_{O(sink)} = 5mA$		0.05	0.2		0.25	0.35			
V_{OH}	High level output voltage							V	
	$V_{CC} = +15V$ $I_{O(sink)} = 200mA$		13	12.5		12.5			
	$I_{O(sink)} = 100mA$		3	13.3		13.3			
	$V_{CC} = +5V$ $I_{O(sink)} = 100mA$		3	3.3		3.3			

Table 3. $T_{amb} = +25^{\circ} C$, $V_{CC} = +5 V$ to $+15 V$ (unless otherwise specified) (continued)

Symbol	Parameter	SE555			NE555 - SA555			Unit
		Min.	Typ.	Max.	Min.	Typ.	Max.	
$I_{dis(off)}$	Discharge pin leakage current (output high) $V_{dis} = 10V$		20	100		20	100	nA
$V_{dis(sat)}$	Discharge pin saturation voltage (output low) ⁽⁴⁾ $V_{CC} = +15V, I_{dis} = 15mA$ $V_{CC} = +5V, I_{dis} = 4.5mA$		180 80	480 200		180 80	480 200	mV
t_r t_f	Output rise time Output fall time		100 100	200 200		100 100	300 300	ns
t_{off}	Turn off time ⁽⁵⁾ ($V_{reset} = V_{CC}$)		0.5			0.5		μs

1. Tested at $V_{CC} = +5 V$ and $V_{CC} = +15 V$.
2. This will determine the maximum value of $R_A + R_B$ for 15 V operation. The maximum total ($R_A + R_B$) is 20 M Ω for +15 V operation and 3.5 M Ω for +5 V operation.
3. Specified with trigger input high.
4. No protection against excessive pin 7 current is necessary, providing the package dissipation rating is not exceeded.
5. Time measured from a positive pulse (from 0 V to $0.8 \times V_{CC}$) on the Threshold pin to the transition from high to low on the output pin. Trigger is tied to threshold.

Figure 3. Minimum pulse width required for triggering

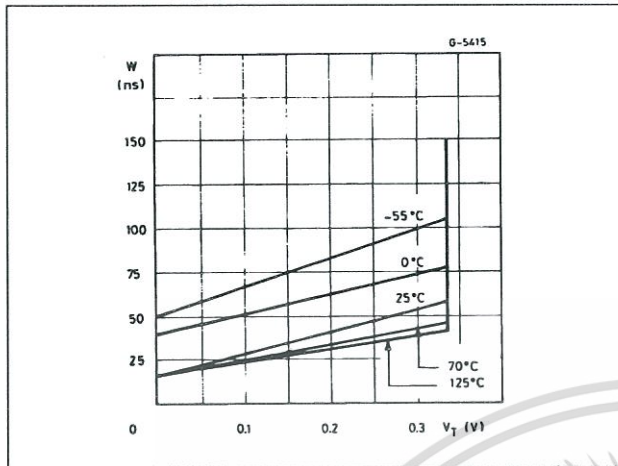


Figure 4. Supply current versus supply voltage

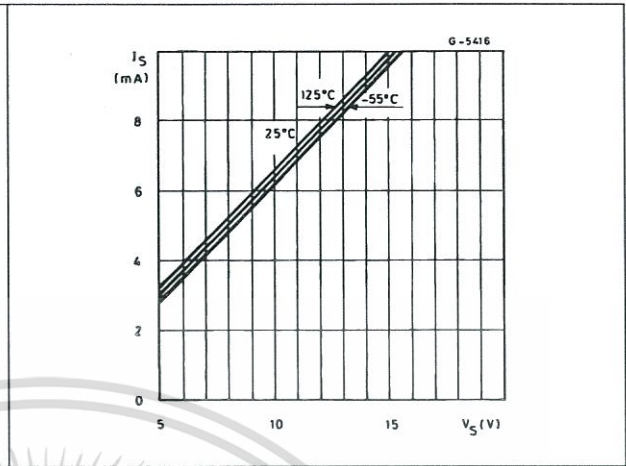


Figure 5. Delay time versus temperature

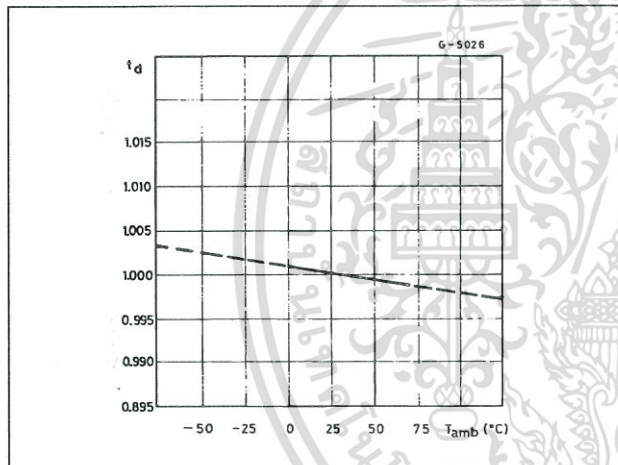


Figure 6. Low output voltage versus output sink current

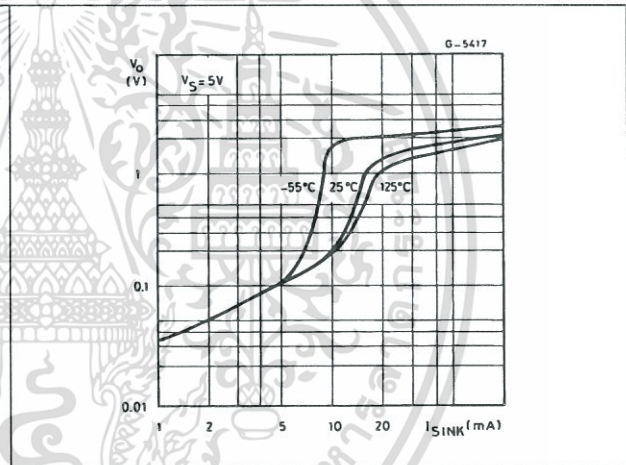


Figure 7. Low output voltage versus output sink current

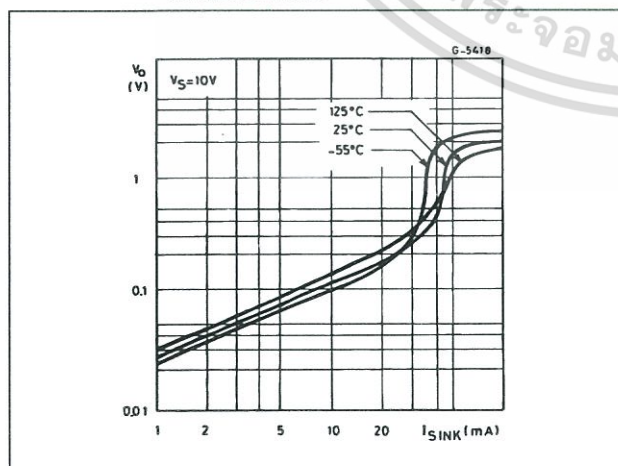


Figure 8. Low output voltage versus output sink current

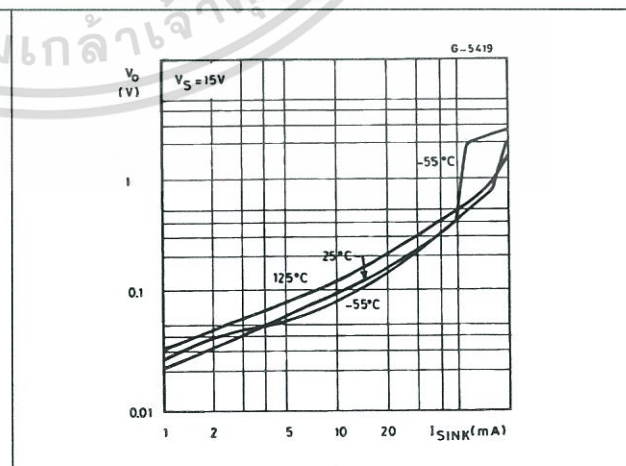


Figure 9. High output voltage drop versus output

Figure 10. Delay time versus supply voltage

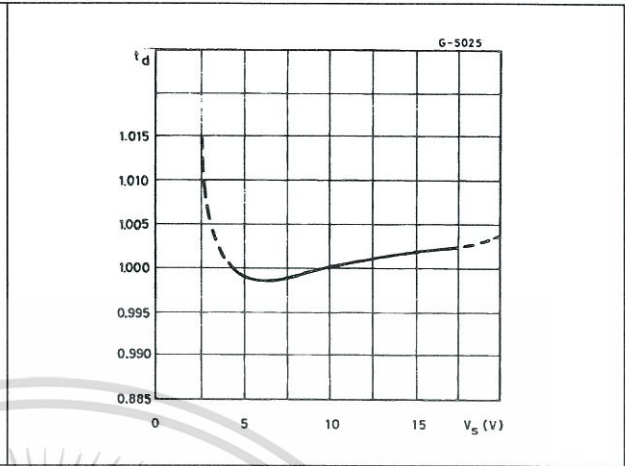
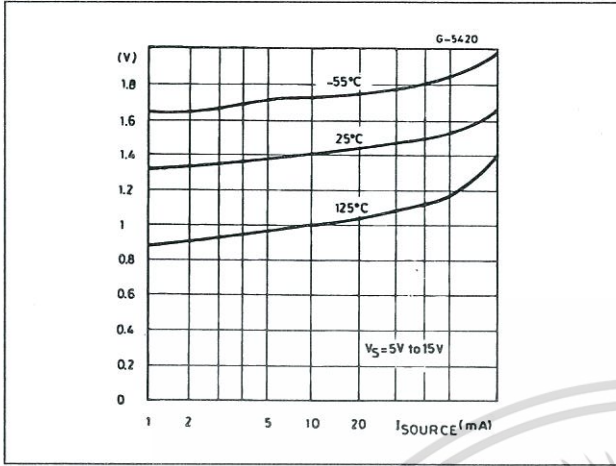
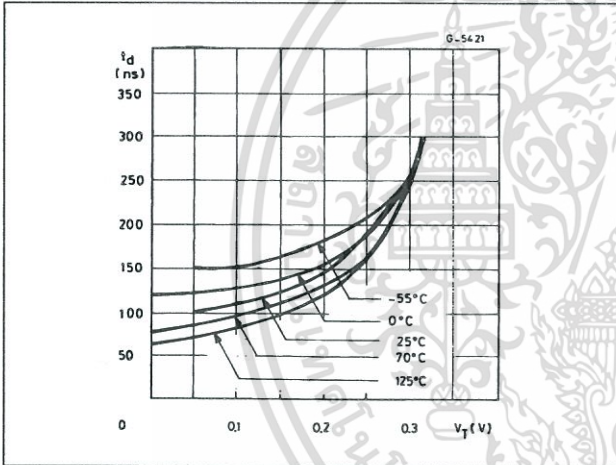


Figure 11. Propagation delay versus voltage level of trigger value

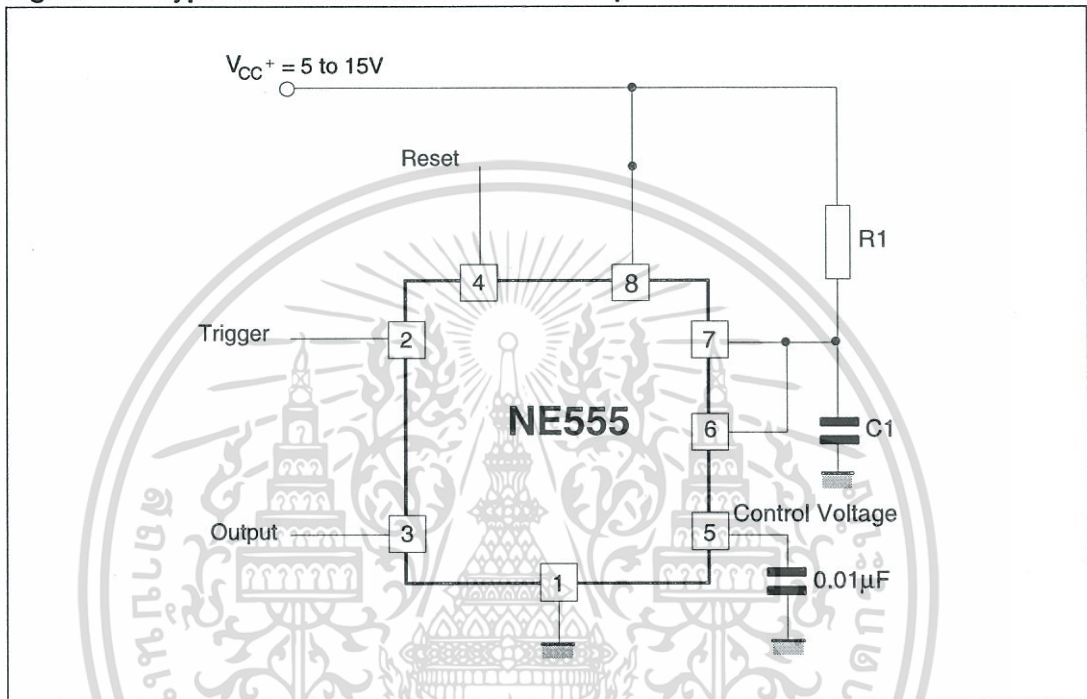


4 Application information

4.1 Monostable operation

In the monostable mode, the timer generates a single pulse. As shown in *Figure 12*, the external capacitor is initially held discharged by a transistor inside the timer.

Figure 12. Typical schematics in monostable operation



The circuit triggers on a negative-going input signal when the level reaches $1/3 V_{CC}$. Once triggered, the circuit remains in this state until the set time has elapsed, even if it is triggered again during this interval. The duration of the output HIGH state is given by $t = 1.1 R_1 C_1$ and is easily determined by *Figure 14*.

Note that because the charge rate and the threshold level of the comparator are both directly proportional to supply voltage, the timing interval is independent of supply. Applying a negative pulse simultaneously to the reset terminal (pin 4) and the trigger terminal (pin 2) during the timing cycle discharges the external capacitor and causes the cycle to start over. The timing cycle now starts on the positive edge of the reset pulse. During the time the reset pulse is applied, the output is driven to its LOW state.

When a negative trigger pulse is applied to pin 2, the flip-flop is set, releasing the short-circuit across the external capacitor and driving the output HIGH. The voltage across the capacitor increases exponentially with the time constant $t = R_1 C_1$. When the voltage across the capacitor equals $2/3 V_{CC}$, the comparator resets the flip-flop which then discharges the capacitor rapidly and drives the output to its LOW state.

Figure 13 shows the actual waveforms generated in this mode of operation.

When Reset is not used, it should be tied high to avoid any possibility of unwanted triggering.

Figure 13. Waveforms in monostable operation

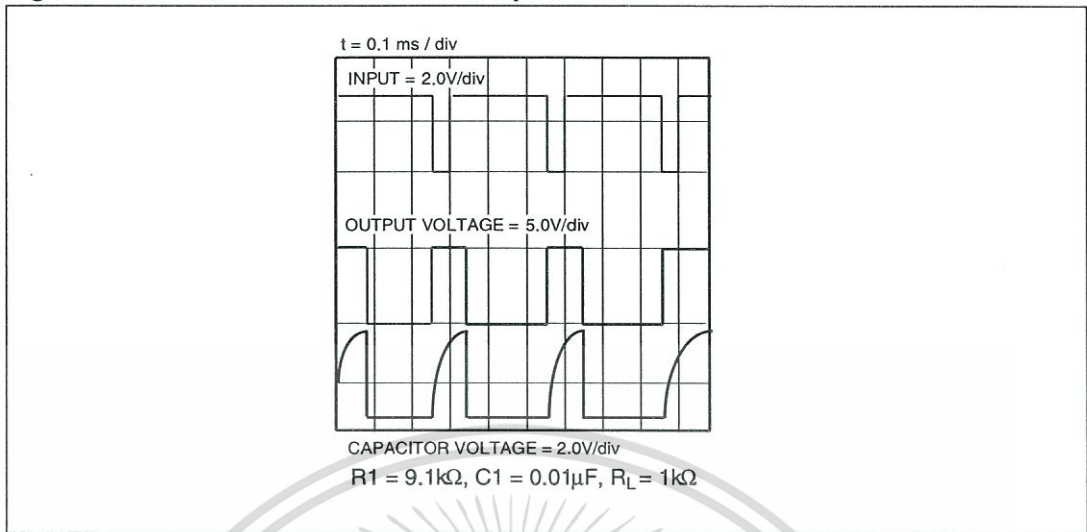
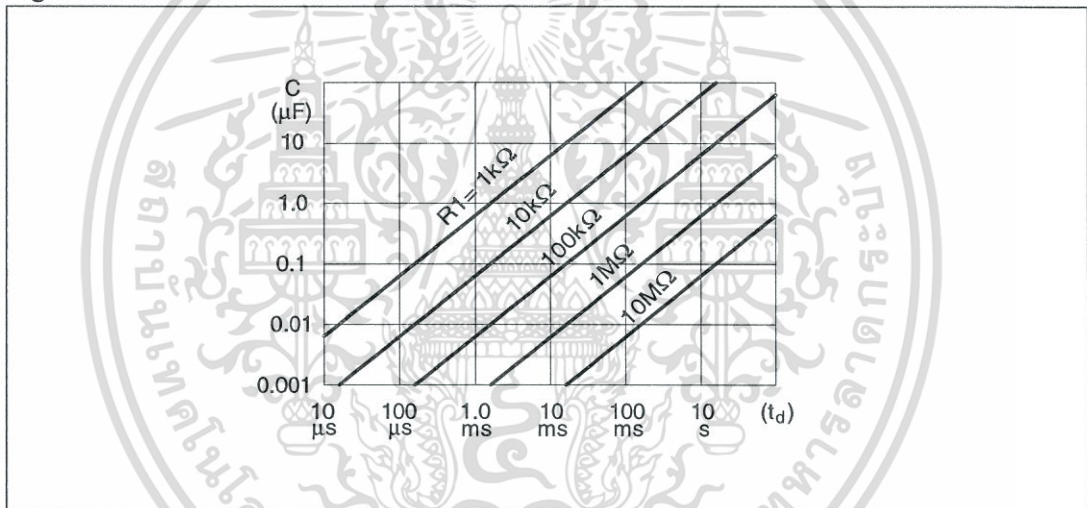


Figure 14. Pulse duration versus R1C1



4.2 Astable operation

When the circuit is connected as shown in *Figure 15* (pins 2 and 6 connected) it triggers itself and free runs as a multi-vibrator. The external capacitor charges through R_1 and R_2 and discharges through R_2 only. Thus the duty cycle can be set accurately by adjusting the ratio of these two resistors.

In the astable mode of operation, C_1 charges and discharges between $1/3 V_{CC}$ and $2/3 V_{CC}$. As in the triggered mode, the charge and discharge times and, therefore, frequency are independent of the supply voltage.

Figure 15. Typical schematics in astable operation

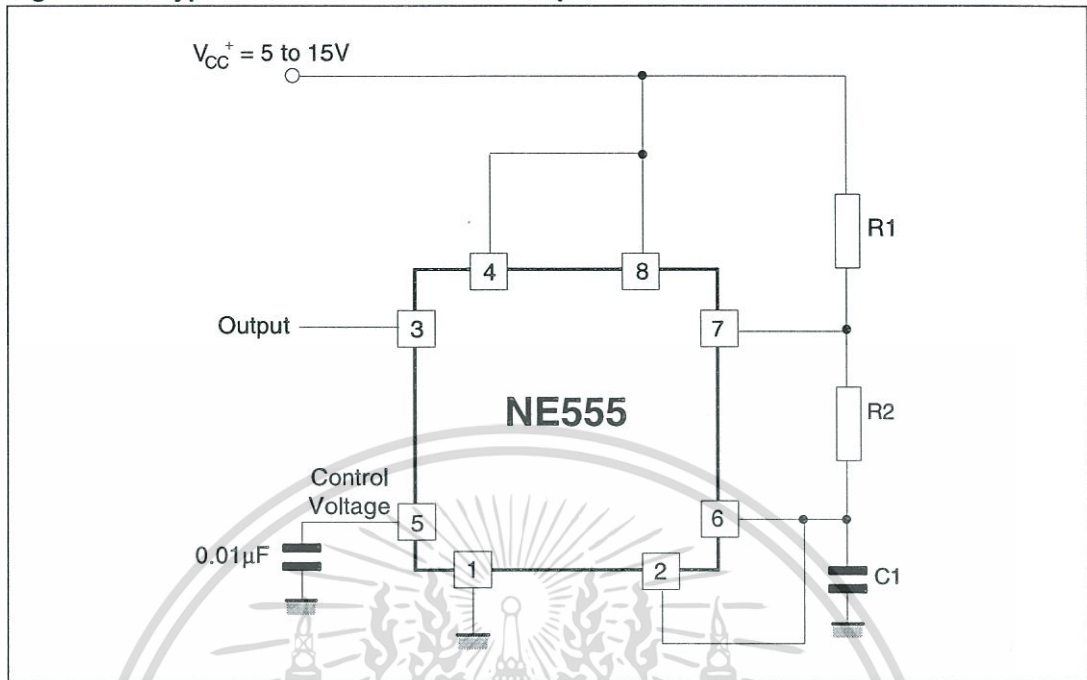


Figure 16 shows the actual waveforms generated in this mode of operation.

The charge time (output HIGH) is given by:

$$t_1 = 0.693 (R_1 + R_2) C_1$$

and the discharge time (output LOW) by:

$$t_2 = 0.693 (R_2) C_1$$

Thus the total period T is given by:

$$T = t_1 + t_2 = 0.693 (R_1 + 2R_2) C_1$$

The frequency of oscillation is then:

$$f = \frac{1}{T} = \frac{1.44}{(R_1 + 2R_2)C_1}$$

It can easily be found from Figure 17.

The duty cycle is given by:

$$D = \frac{R_2}{R_1 + 2R_2}$$

Figure 16. Waveforms in astable operation

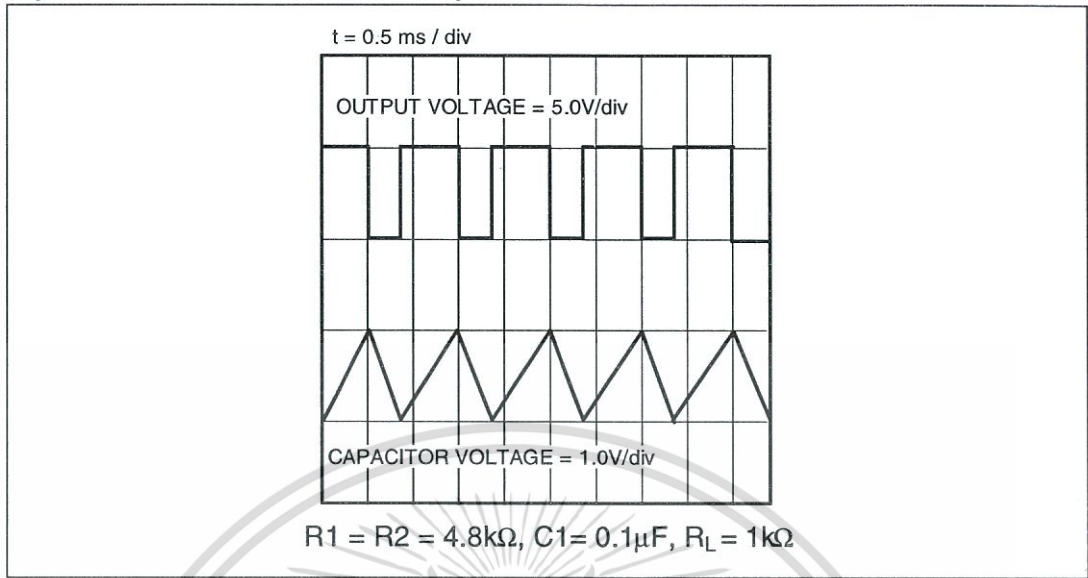
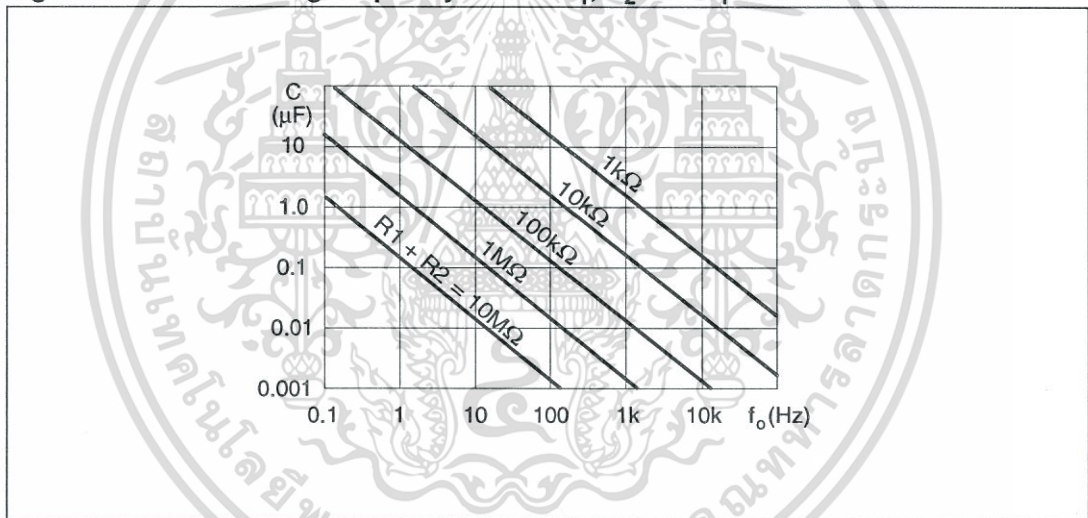


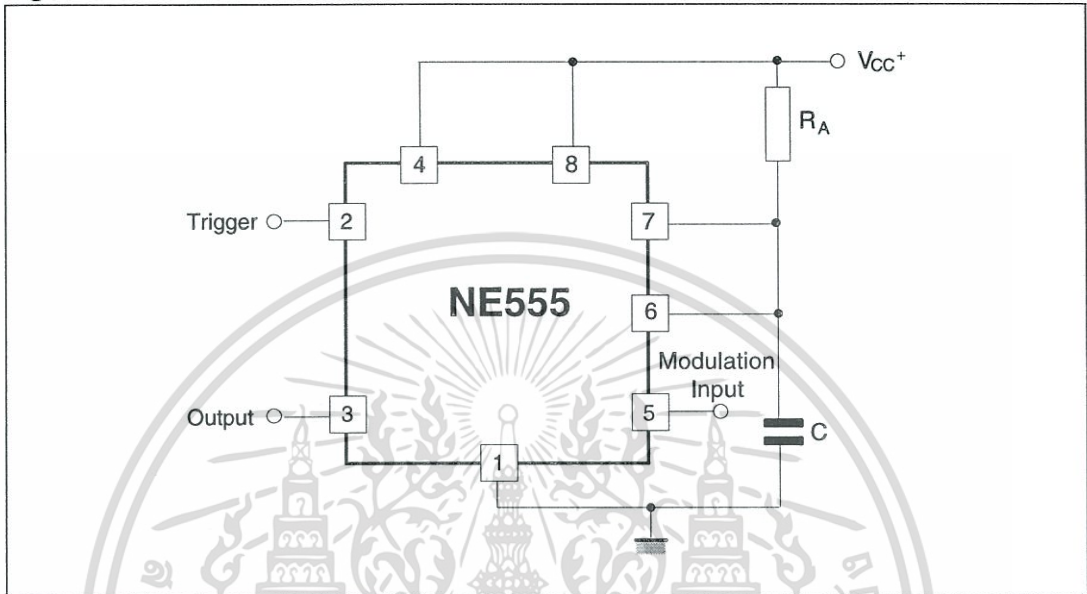
Figure 17. Free running frequency versus R_1, R_2 and C_1



4.3 Pulse width modulator

When the timer is connected in the monostable mode and triggered with a continuous pulse train, the output pulse width can be modulated by a signal applied to pin 5. *Figure 18* shows the circuit.

Figure 18. Pulse width modulator



4.4 Linear ramp

When the pull-up resistor, R_A , in the monostable circuit is replaced by a constant current source, a linear ramp is generated. *Figure 19* shows a circuit configuration that will perform this function.

Figure 19. Linear ramp

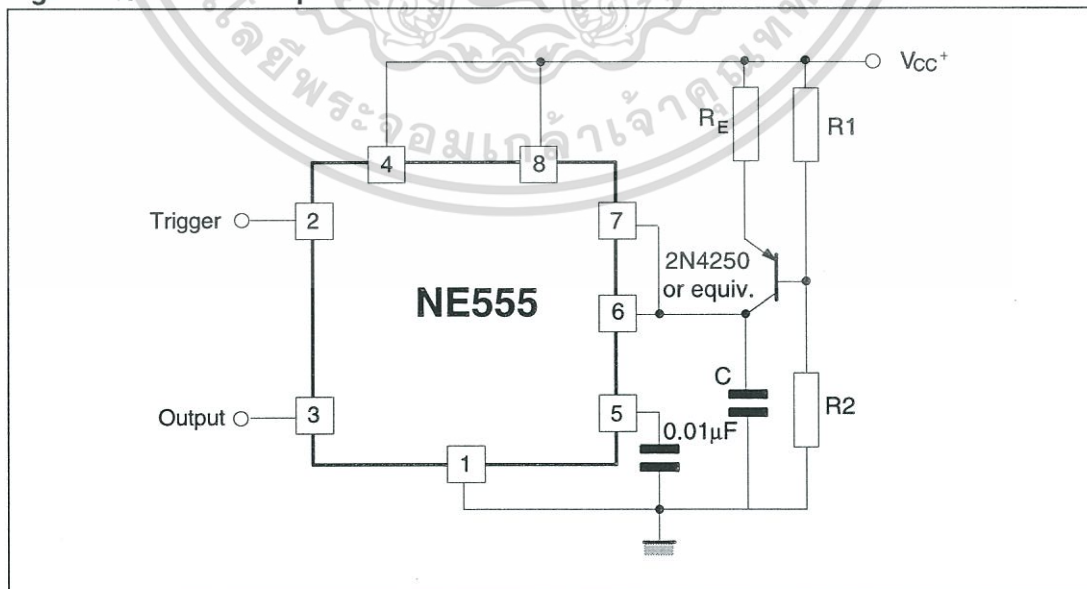
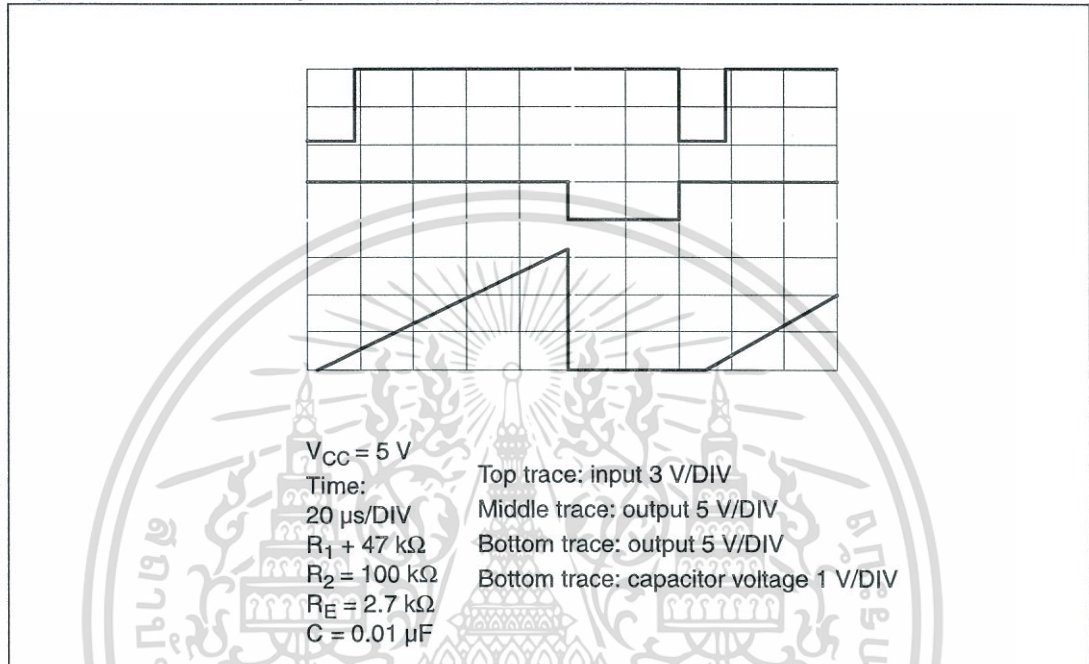


Figure 20 shows the waveforms generated by the linear ramp.

The time interval is given by:

$$T = \frac{(2/3 V_{CC} R_E (R_1 + R_2) C}{R_1 V_{CC} - V_{BE} (R_1 + R_2)} \quad V_{BE} = 0.6V$$

Figure 20. Linear ramp



4.5 50% duty cycle oscillator

For a 50% duty cycle, the resistors R_A and R_B can be connected as in Figure 21. The time period for the output high is the same as for astable operation (see Section 4.2 on page 9):

$$t_1 = 0.693 R_A C$$

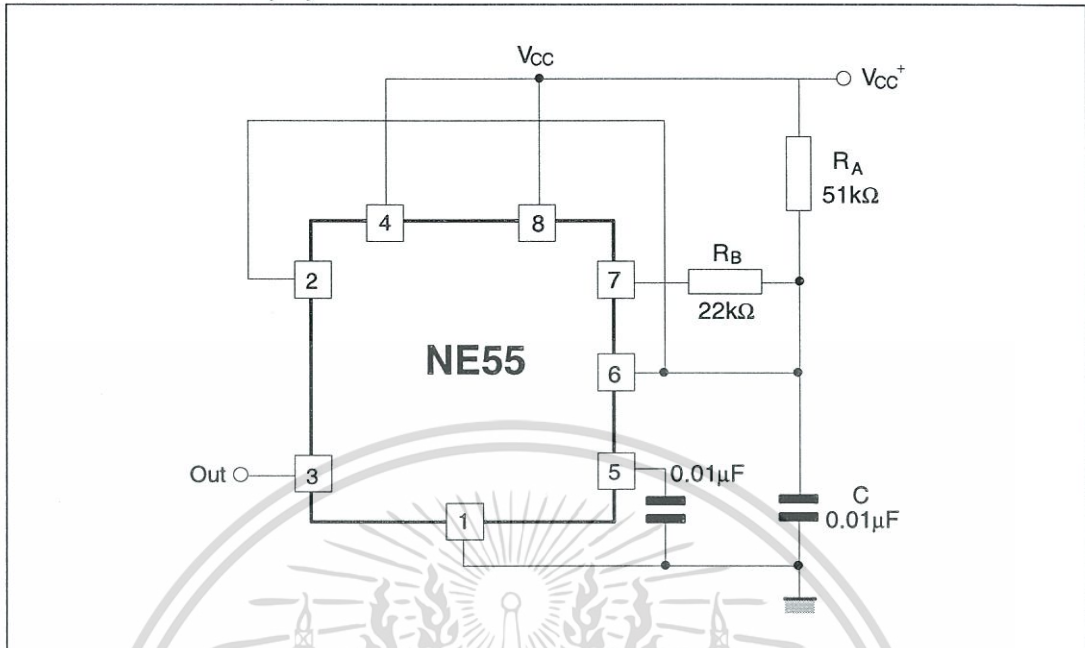
For the output low it is

$$t_2 = \left[\frac{R_B}{R_A + R_B} \right] C \cdot \ln \left[\frac{R_B - 2R_A}{2R_B - R_A} \right]$$

Thus the frequency of oscillation is:

$$f = \frac{1}{t_1 + t_2}$$

Figure 21. 50% duty cycle oscillator



Note that this circuit will not oscillate if R_B is greater than $1/2 R_A$ because the junction of R_A and R_B cannot bring pin 2 down to $1/3 V_{CC}$ and trigger the lower comparator.

4.6 Additional information

Adequate power supply bypassing is necessary to protect associated circuitry. The minimum recommended is $0.1 \mu\text{F}$ in parallel with $1 \mu\text{F}$ electrolytic.

5 Package information

In order to meet environmental requirements, STMicroelectronics offers these devices in ECOPACK® packages. These packages have a lead-free second level interconnect. The category of second level interconnect is marked on the package and on the inner box label, in compliance with JEDEC Standard JESD97. The maximum ratings related to soldering conditions are also marked on the inner box label. ECOPACK is an STMicroelectronics trademark. ECOPACK specifications are available at: www.st.com.



5.1 DIP8 package information

Figure 22. DIP8 package mechanical drawing

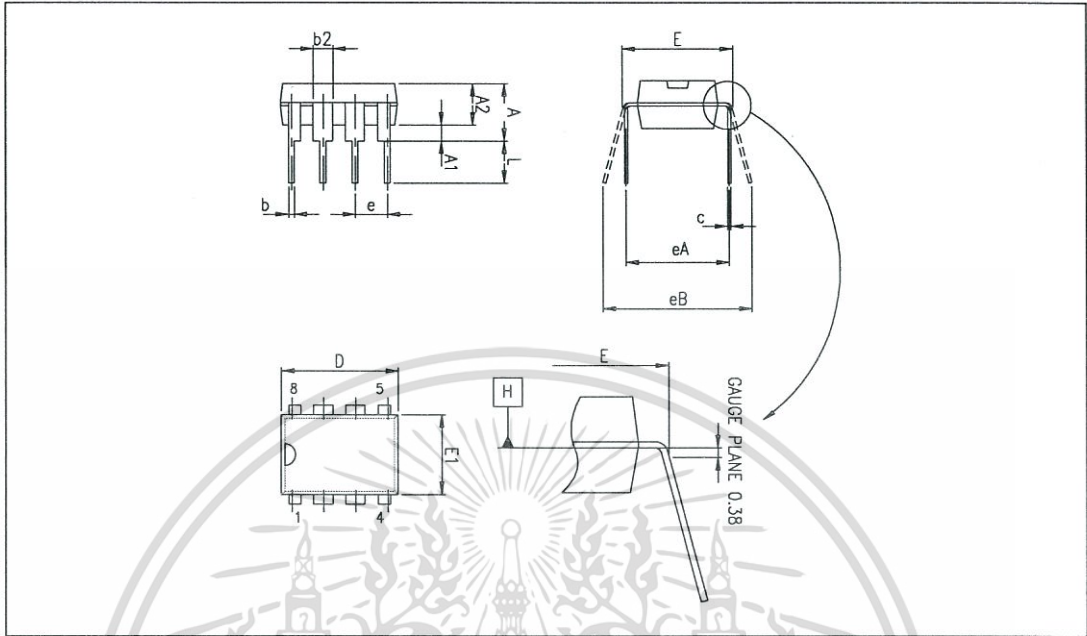


Table 4. DIP8 package mechanical data

Ref.	Dimensions					
	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A			5.33			0.210
A1	0.38			0.015		
A2	2.92	3.30	4.95	0.115	0.130	0.195
b	0.36	0.46	0.56	0.014	0.018	0.022
b2	1.14	1.52	1.78	0.045	0.060	0.070
c	0.20	0.25	0.36	0.008	0.010	0.014
D	9.02	9.27	10.16	0.355	0.365	0.400
E	7.62	7.87	8.26	0.300	0.310	0.325
E1	6.10	6.35	7.11	0.240	0.250	0.280
e		2.54			0.100	
eA		7.62			0.300	
eB			10.92			0.430
L	2.92	3.30	3.81	0.115	0.130	0.150

5.2 SO-8 package information

Figure 23. SO-8 package mechanical drawing

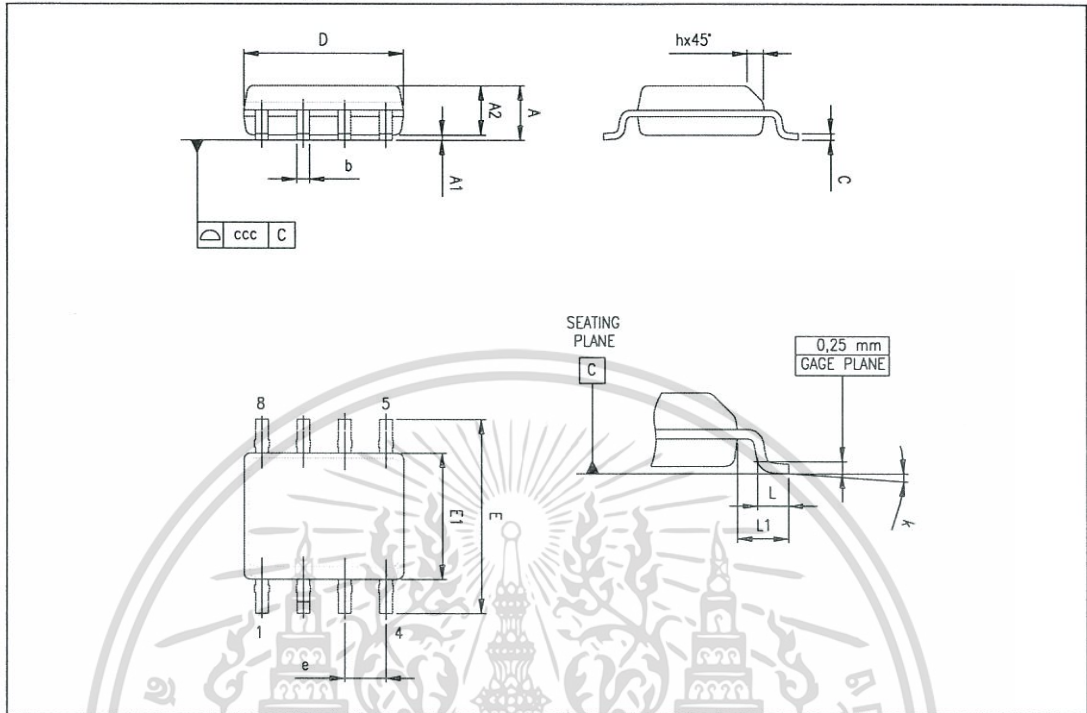


Table 5. SO-8 package mechanical data

Ref.	Dimensions					
	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A			1.75			0.069
A1	0.10		0.25	0.004		0.010
A2	1.25			0.049		
b	0.28		0.48	0.011		0.019
c	0.17		0.23	0.007		0.010
D	4.80	4.90	5.00	0.189	0.193	0.197
E	5.80	6.00	6.20	0.228	0.236	0.244
E1	3.80	3.90	4.00	0.150	0.154	0.157
e		1.27			0.050	
h	0.25		0.50	0.010		0.020
L	0.40		1.27	0.016		0.050
L1		1.04			0.040	
k	0		8°	1°		8°
ccc			0.10			0.004

6 Ordering information

Table 6. Order codes

Part number	Temperature range	Package	Packing	Marking
NE555N	0°C, +70°C	DIP8	Tube	NE555N
NE555D/DT		SO-8	Tube or tape & reel	NE555
SA555N	-40°C, +105°C	DIP8	Tube	SA555N
SA555D/DT		SO-8	Tube or tape & reel	SA555
SE555N	-55°C, + 125°C	DIP8	Tube	SE555N
SE555D/DT		SO-8	Tube or tape & reel	SE555



7 Revision history

Table 7. Document revision history

Date	Revision	Changes
01-Jun-2003	1	Initial release.
2004-2006	2-3	Internal revisions
15-Mar-2007	4	Expanded order code table. Template update.
06-Nov-2008	5	Added I_{OUT} value in <i>Table 1: Absolute maximum ratings</i> and <i>Table 2: Operating conditions</i> . Added ESD tolerance, latch-up tolerance, R_{thja} and R_{thjc} in <i>Table 1: Absolute maximum ratings</i> .



Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST's terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST'S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED ST REPRESENTATIVE, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER'S OWN RISK.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

© 2008 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com

DM7408

Quad 2-Input AND Gates

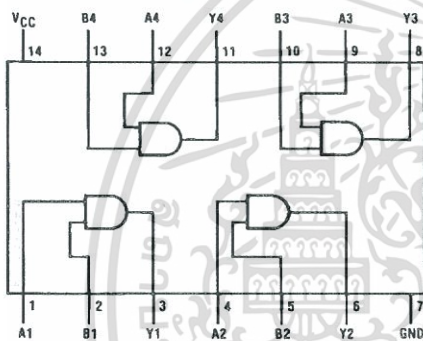
General Description

This device contains four independent gates each of which performs the logic AND function.

Ordering Code:

Order Number	Package Number	Package Description
DM7408N	N14A	14-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300" Wide

Connection Diagram



Function Table

$Y = AB$

Inputs		Output
A	B	Y
L	L	L
L	H	L
H	L	L
H	H	H

H = HIGH Logic Level
L = LOW Logic Level

Absolute Maximum Ratings(Note 1)

Supply Voltage	7V
Input Voltage	5.5V
Operating Free Air Temperature Range	0°C to +70°C
Storage Temperature Range	-65°C to +150°C

Note 1: The "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at these limits. The parametric values defined in the Electrical Characteristics tables are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Recommended Operating Conditions

Symbol	Parameter	Min	Nom	Max	Units
V _{CC}	Supply Voltage	4.75	5	5.25	V
V _{IH}	HIGH Level Input Voltage	2			V
V _{IL}	LOW Level Input Voltage			0.8	V
I _{OH}	HIGH Level Output Current			-0.8	mA
I _{OL}	LOW Level Output Current			16	mA
T _A	Free Air Operating Temperature	0		70	°C

Electrical Characteristics

over recommended operating free air temperature range (unless otherwise noted)

Symbol	Parameter	Conditions	Min	Typ (Note 2)	Max	Units
V _I	Input Clamp Voltage	V _{CC} = Min, I _I = -12 mA			-1.5	V
V _{OH}	HIGH Level Output Voltage	V _{CC} = Min, I _{OH} = Max V _{IL} = Max	2.4	3.4		V
V _{OL}	LOW Level Output Voltage	V _{CC} = Min, I _{OL} = Max V _{IH} = Min		0.2	0.4	V
I _I	Input Current @ Max Input Voltage	V _{CC} = Max, V _I = 5.5V			1	mA
I _{IH}	HIGH Level Input Current	V _{CC} = Max, V _I = 2.4V			40	μA
I _{IL}	LOW Level Input Current	V _{CC} = Max, V _I = 0.4V			-1.6	mA
I _{OS}	Short Circuit Output Current	V _{CC} = Max (Note 3)	-18		-55	mA
I _{CCH}	Supply Current with Outputs HIGH	V _{CC} = Max		11	21	mA
I _{CCL}	Supply Current with Outputs LOW	V _{CC} = Max		20	33	mA

Note 2: All typicals are at V_{CC} = 5V, T_A = 25°C.

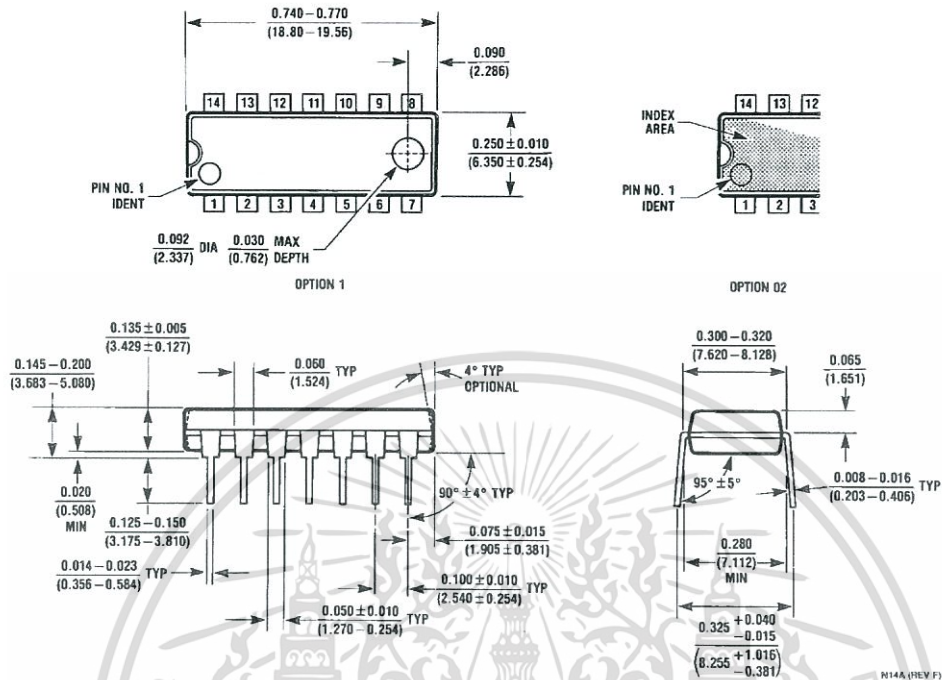
Note 3: Not more than one output should be shorted at a time.

Switching Characteristics

at V_{CC} = 5V and T_A = 25°C

Symbol	Parameter	Conditions	Min	Max	Units
t _{PLH}	Propagation Delay Time LOW-to-HIGH Level Output	C _L = 15 pF R _L = 400Ω		27	ns
t _{PHL}	Propagation Delay Time HIGH-to-LOW Level Output			19	ns

Physical Dimensions inches (millimeters) unless otherwise noted



14-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300" Wide Package Number N14A

Fairchild does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and Fairchild reserves the right at any time without notice to change said circuitry and specifications.

LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF FAIRCHILD SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component in any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

www.fairchildsemi.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมประมวลผลภาพ

```
#include <sstream>
#include <string>
#include <iostream>
#include <highgui.h>
#include <stdio.h>
#include <cv.h>
#include <cxcore.hpp>
#include <wiringPi.h>
// #include <imgproc.hpp>
// #include <opencv2/video/background_segm.hpp>
// #include <window.h>
const int Pin0 = 0;
const int Pin1 = 1;
const int Pin2 = 2;
const int Pin3 = 3;

using namespace cv;
using namespace std;

const int FRAME_WIDTH = 1280;
const int FRAME_HEIGHT = 720;

const int MIN_OBJECT_AREA = 20*20;
const int MAX_OBJECT_AREA = FRAME_HEIGHT*FRAME_WIDTH/1.5;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//functions prototypes  
void on_trackbar(int, void*);  
void createTrackbars();  
void showimgcontours(int &x, int &y, Mat &thresedimg, Mat &original);  
void toggle(int key);  
void morphit(Mat &img);  
void blurthresh(Mat &img);  
void condefects(vector<Vec4i> convexityDefectsSet, vector<Point> mycontour, Mat &frame);  
string intToString(int number);
```

```
//function prototypes ends here
```

```
//boolean toggles
```

```
bool domorph = false;
```

```
bool doblurthresh = false;
```

```
bool showchangedframe = false;
```

```
bool showcontours = false;
```

```
bool showhull = false;
```

```
bool showcondefects = false;
```

```
//boolean toggles end
```

```
int H_MIN = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int H_MAX = 255;
int S_MIN = 0;
int S_MAX = 255;
int V_MIN = 0;
int V_MAX = 255;
```

```
int kerode = 1;
int kdilate = 1;
int kblur = 1;
int threshval = 0;
```

```
int main(void)
{
    wiringPiSetup ();
    pinMode (Pin0, OUTPUT) ;
    pinMode (Pin1, OUTPUT) ;
    pinMode (Pin2, OUTPUT) ;
    pinMode (Pin3, OUTPUT) ;
```

```
int x, y;
createTrackbars();
on_trackbar(0, 0);
```

```
Mat frame, hsvframe, rangeframe;
```

```
int key;
```

```
VideoCapture cap(0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

while ((key = waitKey(30)) != 27)
{

toggle(key);

//cap.set(CV_CAP_PROP_FPS,5);

//cap.set(CV_CAP_PROP_FRAME_WIDTH,FRAME_WIDTH);

//cap.set(CV_CAP_PROP_FRAME_HEIGHT,FRAME_HEIGHT);

cap.read(frame);

//cap >> frame;

//flip(frame, frame, 180);
cvtColor(frame, hsvframe, COLOR_BGR2HSV);

inRange(hsvframe, Scalar(H_MIN, S_MIN, V_MIN), Scalar(H_MAX, S_MAX, V_MAX),
rangeframe);

//imshow("HSV", hsvframe);

if (domorph)
    morphit(rangeframe);

if (doblurthresh)
    blurthresh(rangeframe);

if (showcontours)
    showingcontours(x, y, rangeframe, frame);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (showchangedframe)
    imshow("Camera", frame);
else
    imshow("Camera", rangeframe);
}
}

```

```

void on_trackbar(int, void*)

```

```

//This function gets called whenever a
// trackbar position is changed

```

```

if (kerode == 0)
    kerode = 1;
if (kdilate == 0)
    kdilate = 1;
if (kblur == 0)
    kblur = 1;
}

```

```

void createTrackbars()

```

```

{
    String trackbarWindowName = "TrackBars";
    namedWindow(trackbarWindowName, WINDOW_NORMAL);
    createTrackbar("H_MIN", trackbarWindowName, &H_MIN, H_MAX, on_trackbar);
    createTrackbar("H_MAX", trackbarWindowName, &H_MAX, H_MAX, on_trackbar);
    createTrackbar("S_MIN", trackbarWindowName, &S_MIN, S_MAX, on_trackbar);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

createTrackbar("S_MAX", trackbarWindowName, &S_MAX, S_MAX, on_trackbar);
createTrackbar("V_MIN", trackbarWindowName, &V_MIN, V_MAX, on_trackbar);
createTrackbar("V_MAX", trackbarWindowName, &V_MAX, V_MAX, on_trackbar);
createTrackbar("Erode", trackbarWindowName, &kerode, 31, on_trackbar);
createTrackbar("Dilate", trackbarWindowName, &kdilate, 31, on_trackbar);
createTrackbar("Blur", trackbarWindowName, &kblur, 255, on_trackbar);
createTrackbar("Thresh", trackbarWindowName, &threshval, 255, on_trackbar);

```

```

}

void morphit(Mat &img)
{
    erode(img, img, getStructuringElement(MORPH_RECT, Size(kerode, kerode)));
    dilate(img, img, getStructuringElement(MORPH_RECT, Size(kdilate, kdilate)));
}

void blurthresh(Mat &img)
{
    //medianBlur(img, img, kblur%2+3+kblur);
    blur(img, img, Size(kblur, kblur), Point(-1, -1), BORDER_DEFAULT);
    threshold(img, img, threshval, 255, THRESH_BINARY_INV);
}

```

```

void toggle(int key)
{

```

```

//toggle line start

```

```

if (key == 'm')

```

```

    domorph = !domorph;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (key == 'b')
    doblurthresh = !doblurthresh;

if (key == 'r')
    showchangedframe = !showchangedframe;

if (key == 'c')
    showcontours = !showcontours;

if (key == 'h')
    showhull = !showhull;

if (key == 'd')
    showcondefects = !showcondefects;

//toggle line end
}

string intToString(int number){

    std::stringstream ss;
    ss << number;
    return ss.str();
}

void showimgcontours(int &x, int &y, Mat &threshedimg, Mat &original)
{
    int a, b, dx, dy;

    a = x;
    b = y;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
digitalWrite(Pin3, 0); // sets the LED on
digitalWrite(Pin2, 0); // sets the LED on
digitalWrite(Pin1, 0); // sets the LED on
digitalWrite(Pin0, 0); // sets the LED on
delay(10);
```

```
vector<vector<Point> > contours;
vector<Vec4i> hierarchy;
int largest_area = 0;
int largest_contour_index = 0;

findContours(threshedimg, contours, hierarchy, CV_RETR_TREE, CV_CHAIN_APPROX_SIMPLE);

double refArea = 0;
if (hierarchy.size() > 0) {

for (int index = 0; index >= 0; index = hierarchy[index][0]) {

    Moments moment = moments((cv::Mat)contours[index]);

    double area = moment.m00;

    if(area>MIN_OBJECT_AREA && area<MAX_OBJECT_AREA && area>refArea){

        x = moment.m10/area;

        y = moment.m01/area;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    refArea = area;
}
}

if(x > 100 && x < 540){
    dx = x - a;
    dy = y - b;
    putText(original,intToString(dx),Point(0,100),2,1,Scalar(0,255,0),2);
}
}

/// Find the convex hull,contours and defects for each contour
vector<vector<Point> >hull(contours.size());
vector<vector<int> >inhull(contours.size());
vector<vector<Vec4i> >defects(contours.size());
for (int i = 0; i < contours.size(); i++)
{
    convexHull(Mat(contours[i]), hull[i], false);
    convexHull(Mat(contours[i]), inhull[i], false);
    if (inhull[i].size()>3)
        convexityDefects(contours[i], inhull[i], defects[i]);
}

//find hull and contour and defects end here
//this will find largest contour
for (int i = 0; i < contours.size(); i++) // iterate through each contour.
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

double a = contourArea(contours[i], false); // Find the area of contour
if (a>largest_area)
{
    largest_area = a;
    largest_contour_index = i;          //Store the index of largest contour
}
}
//search for largest contour has end

if (contours.size() >0)
{
    drawContours(original, contours, largest_contour_index, CV_RGB(0, 255, 0), 2, 8, hierarchy);
    //if want to show all contours use below one
    //drawContours(original,contours,-1, CV_RGB(0, 255, 0), 2, 8, hierarchy);
    if (showhull)
        drawContours(original, hull, largest_contour_index, CV_RGB(0, 0, 255), 2, 8, hierarchy);
    //if want to show all hull, use below one
    //drawContours(original,hull,-1, CV_RGB(0, 255, 0), 2, 8, hierarchy);
    if (showcondefects){

        if( refArea > 20*20 ){

            if(dx>100){

                putText(original,"Increase CH",Point(0,150),2,1,Scalar(255,0,0),2);

                digitalWrite(Pin3, 0); // sets the LED on

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

digitalWrite(Pin2, 1); // sets the LED on
digitalWrite(Pin1, 1); // sets the LED on
digitalWrite(Pin0, 0); // sets the LED on

delay(10);

dx=0;

//increaseCH();

}

else if(dx<-100){

  putText(original,"Decrease CH",Point(0,150),2,1,Scalar(255,0,0),2);

  digitalWrite(Pin3, 0); // sets the LED on
  digitalWrite(Pin2, 1); // sets the LED on
  digitalWrite(Pin1, 1); // sets the LED on
  digitalWrite(Pin0, 1); // sets the LED on

  delay(10);

  dx=0;

  //decreaseCH();

}

else if(y < 160 && x > 160 && x < 480){

  putText(original,"Increase Vol",Point(0,150),2,1,Scalar(255,0,0),2);

  digitalWrite(Pin3, 1); // sets the LED on

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

digitalWrite(Pin2, 0); // sets the LED on
digitalWrite(Pin1, 0); // sets the LED on
digitalWrite(Pin0, 0); // sets the LED on

delay(10);

y=240;

//increaseVol();
}

else if(y > 320 && x > 160 && x < 480){
  putText(original,"Decrease Vol",Point(0,150),2,1,Scalar(255,0,0),2);

  digitalWrite(Pin3, 1); // sets the LED on
  digitalWrite(Pin2, 0); // sets the LED on
  digitalWrite(Pin1, 0); // sets the LED on
  digitalWrite(Pin0, 1); // sets the LED on

  delay(10);

  y=240;

  //decreaseVol();
}

else if(x > 160 && x < 480){
  condefects(defects[largest_contour_index], contours[largest_contour_index], original);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else {
```

```
    digitalWrite(Pin3, 0); // sets the LED on
```

```
    digitalWrite(Pin2, 0); // sets the LED on
```

```
    digitalWrite(Pin1, 0); // sets the LED on
```

```
    digitalWrite(Pin0, 0); // sets the LED on
```

```
    delay(10);
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
void condefects(vector<Vec4i> convexityDefectsSet, vector<Point> mycontour, Mat &original)
```

```
{
```

```
    Point2f mycenter;
```

```
    float radii;
```

```
    int fingers = 0;
```

```
    minEnclosingCircle(mycontour, mycenter, radii);
```

```
    circle(original, mycenter, 10, CV_RGB(0, 0, 255), 2, 8);
```

```
    cout << "===start===" << endl;
```

```
    for (int cDeflt = 0; cDeflt < convexityDefectsSet.size(); cDeflt++) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int startIdx = convexityDefectsSet[cDeflt].val[0]; Point ptStart(mycontour[startIdx]);

int endIdx = convexityDefectsSet[cDeflt].val[1]; Point ptEnd(mycontour[endIdx]);

int farIdx = convexityDefectsSet[cDeflt].val[2]; Point ptFar(mycontour[farIdx]);

double depth = static_cast<double>(convexityDefectsSet[cDeflt].val[3]) / 256;

//cout << "depth" << depth << endl;

//display start points of finger

if (depth>11 && ptStart.y<mycenter.y)
{
    circle(original, ptStart, 4, CV_RGB(255, 0, 0), 4);
    fingers++;
}
}

//display fingers on image
putText(original, "Fingers : " + intToString(fingers), Point(50, 50), 2, 1, CV_RGB(255, 0, 0), 4, 8);

if (fingers == 1){

    digitalWrite(Pin3, 0); // sets the LED on
    digitalWrite(Pin2, 0); // sets the LED on
    digitalWrite(Pin1, 0); // sets the LED on
    digitalWrite(Pin0, 1); // sets the LED on

    delay(10);

    fingers = 0;

}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (fingers == 2){

    digitalWrite(Pin3, 0); // sets the LED on
    digitalWrite(Pin2, 0); // sets the LED on
    digitalWrite(Pin1, 1); // sets the LED on
    digitalWrite(Pin0, 0); // sets the LED on

    delay(10);

    fingers = 0;
}

if (fingers == 3){

    digitalWrite(Pin3, 0); // sets the LED on
    digitalWrite(Pin2, 0); // sets the LED on
    digitalWrite(Pin1, 1); // sets the LED on
    digitalWrite(Pin0, 1); // sets the LED on

    delay(10);

    fingers = 0;
}
```

```
if (fingers == 4){
```

```
    digitalWrite(Pin3, 0); // sets the LED on
    digitalWrite(Pin2, 1); // sets the LED on
    digitalWrite(Pin1, 0); // sets the LED on
    digitalWrite(Pin0, 0); // sets the LED on
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
delay(10);

fingers = 0;

}

if (fingers == 5){

    digitalWrite(Pin3, 0); // sets the LED on
    digitalWrite(Pin2, 1); // sets the LED on
    digitalWrite(Pin1, 0); // sets the LED on
    digitalWrite(Pin0, 1); // sets the LED on
    delay(10);
    fingers = 0;

}

cout << "=="end line==" << endl;
} // condefacts ends here
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้