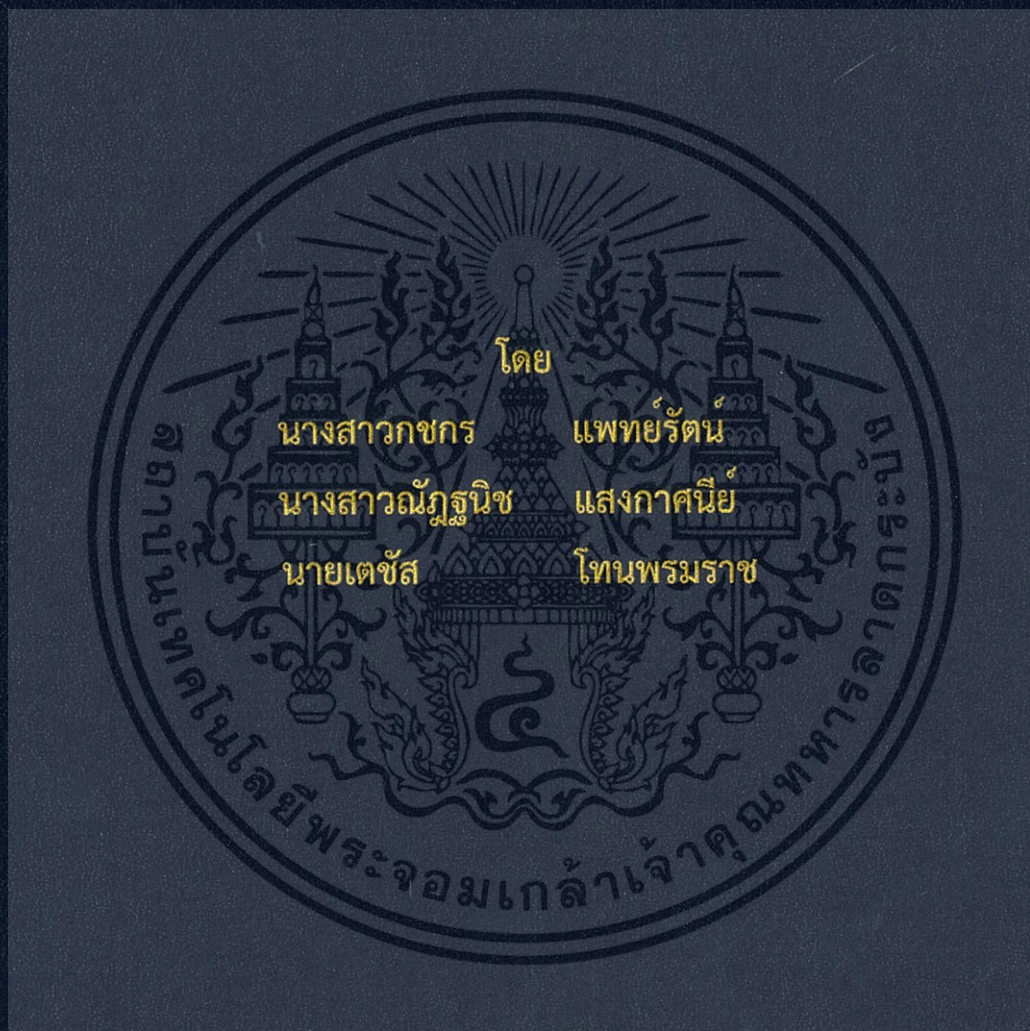


แบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์  
AUTOMATIC PROGRAM LAWNMOWER MODEL USING  
MICROCONTROLLER



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมโทรคมนาคม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2558

# สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

แบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์  
AUTOMATIC PROGRAM LAWNMOWER MODEL USING  
MICROCONTROLLER



T144417

โดย  
นางสาว กชกร แพทย์รัตน์  
นางสาว ณัฐนิช แสงภาคนิย์  
นาย เตชัส โทณพรมราช

สาขา.....  
เลขทะเบียน 144417  
วันเดือนปี 24 พ.ย. 2559

b. 144417 21/11/59  
i. ....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมโทรคมนาคม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2558

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์  
Automatic Program Lawnmower Model Using Microcontroller

โดย

นางสาว กชกร	แพทย์รัตน์	55010002
นางสาว ณีฐฐนิช	แสงภาคนีย์	55010340
นาย เตชัส	โชนพรมราช	55010441

อาจารย์ที่ปรึกษา

รศ.ดร. พิพัฒน์ พรหมมี

ผศ.ดร. มนตรี คำเงิน

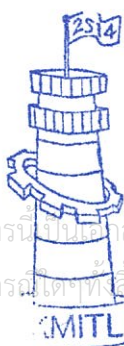
ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558

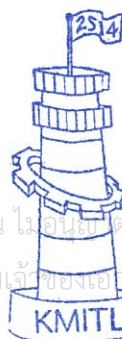


ผ่านการตรวจชิ้นงานแล้ว

(.....)  
กรรมการผู้ตรวจชิ้นงาน

16/05/59

วิศวกรรมโทรคมนาคม  
Telecommunications Engineering



ผ่านการตรวจรูปเล่มแล้ว

(.....)  
อาจารย์ที่ปรึกษา

16/05/59

วิศวกรรมโทรคมนาคม  
Telecommunications Engineering

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์อื่น  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2558

สาขาวิชาวิศวกรรมโทรคมนาคม


คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง แบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์

Automatic Program Lawnmower Model Using Microcontroller

ผู้จัดทำ

1. นางสาวกชกร แพทย์รัตน์
2. นางสาวณัฐนิช แสงกาศนีย์
3. นายเตชส์ โทนมพรราชา

  
..... อาจารย์ที่ปรึกษา

(รศ.ดร. พิพัฒน์ พรหมมี)

  
..... อาจารย์ที่ปรึกษาร่วม

(ผศ.ดร. มนตรี คำเงิน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ดำเนินงานสำเร็จลุล่วงได้เป็นอย่างดีโดยได้รับความร่วมมือและความช่วยเหลือจากหลายท่านในการให้คำแนะนำแนวทางและวิธีการแก้ไขปัญหา

คณะผู้จัดทำขอกราบขอบพระคุณ รศ.ดร.พิพัฒน์ พรหมมี และ ผศ.ดร.มนตรี คำเงิน อาจารย์ที่ปรึกษา ที่ให้คำปรึกษาและแนะนำทางด้านทฤษฎีและด้านปฏิบัติพร้อมทั้งยังให้ความอนุเคราะห์ในด้านเครื่องมือและอุปกรณ์ที่ใช้ในการทดสอบ ตลอดจนชี้แนะแนวทางในการแก้ไขปัญหาดังกล่าว ที่เกิดขึ้นในการทำปริญญานิพนธ์ครั้งนี้ได้เป็นอย่างดีมาโดยตลอด

คุณอาจารย์ทุกท่าน ที่ประสิทธิ์ประสาทวิชาความรู้และพี่ เพื่อนนักศึกษาทุกท่านที่คอยช่วยเหลือให้คำปรึกษาและแนะนำในหลายๆ ด้าน คณะผู้จัดทำขอขอบพระคุณมา ณ ที่นี้

นางสาวชกร แพทย์รัตน์  
นางสาวณัฐนิช แสงกาศนีย์  
นายเดชาส โทนมพรราช  
ผู้จัดทำ

แบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้  
ไมโครคอนโทรลเลอร์  
Automatic Program Lawnmower Model Using  
Microcontroller

โดย นางสาวกชกร แพทย์รัตน์ 55010002  
นางสาวณัฐนิช แสงภาคนีย์ 55010340  
นายเตชัส โทนมพรราช 55010441

อาจารย์ที่ปรึกษา รศ.ดร.พิพัฒน์ พรมมี  
ผศ.ดร.มนตรี คำเงิน

#### บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ ได้จัดทำแบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์ วัตถุประสงค์เพื่อศึกษาการทำงานของไมโครคอนโทรลเลอร์ พร้อมกับการทำงานร่วมกันของเซนเซอร์อินฟราเรด, เซ็มทิส และ มอเตอร์ เพื่อเป็นต้นแบบของรถตัดหญ้าในอนาคต โดยปริญญานิพนธ์นี้ทำการสร้างวงจรตัวส่งอินฟราเรด ที่มีความถี่ 38 kHz มัลติเพล็กซ์กับสัญญาณ 500 Hz จำนวน 4 ตัว เพื่อวางให้อยู่ในมุมสี่เหลี่ยมสำหรับกำหนดพื้นที่การทำงาน ในส่วนของทิศทางการเคลื่อนที่ของแบบจำลองรถตัดหญ้า ได้ทำการโปรแกรมในไมโครคอนโทรลเลอร์ สั่งการโมดูล L298N ให้ขับมอเตอร์และ ประมวลผลร่วมกับโมดูลเซ็มทิส เพื่อควบคุมความเร็วและทิศทาง ในระหว่างที่แบบจำลองรถตัดหญ้าเคลื่อนที่ ระบบจะทำการตรวจสอบสัญญาณอินฟราเรดโดยใช้ตัวรับสัญญาณ TSOP4838 ซึ่งสามารถอธิบายการทำงานได้เป็นลำดับดังนี้ เริ่มจากทำการวางตัวส่งอินฟราเรดไว้ 4 มุม ของสนามสี่เหลี่ยม และวางแบบจำลองรถตัดหญ้าในมุมล่างซ้ายของสนามในทิศที่ต้องการเพื่อเคลื่อนที่ไปข้างหน้า จากนั้น กดปุ่มเพื่อเก็บค่าองศาทิศทางจากโมดูลเซ็มทิส ใน 4 ทิศ เมื่อเริ่มการทำงาน แบบจำลองรถตัดหญ้าอัตโนมัติ จะทำการเคลื่อนที่ไปข้างหน้า จนกระทั่งตัวรับสัญญาณ TSOP4838 ได้รับสัญญาณอินฟราเรดจากตัวส่ง จะทำการเลี้ยวไปในเส้นทางใหม่ ในพื้นที่สนาม และหยุดการทำงาน เมื่อเคลื่อนที่ครบในพื้นที่ที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ABSTRACT

This project introduces an automatic programmable lawnmower model using microcontroller. The purposes of this project are studying the work of microcontroller and its interface to infrared-sensor, compass and motor for further lawnmower prototype. This project has created infrared transmitters circuits with frequency 38 kHz multiplex 500 Hz for setting area (4 circuits for 4 corners of rectangular). The direction of lawnmower movement is programmed in microcontroller for cooperate between L298N motor driver and compass module for controlling the speed and direction. During lawnmower moving, the system will checking the infrared signal by using TSOP4838 module for arriving the border of desired area. The operation of the project can be explained by the following steps. Firstly, placing infrared transmitters in four corners of the rectangular field and placing Automatic program lawnmower model in lower left corners with forward directions. Secondly, press button collect the data from compass module for four directions. Finally, when it starts, Automatic program lawnmower model is moving forward until TSOP 4838 receiving the infrared signal from infrared transmitters and will turn to new lane in setting area. It will stop automatically when the target is finished.

## สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	IV
สารบัญรูป	VII
สารบัญตาราง	X
<b>บทที่ 1 บทนำ</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของปริญญาานิพนธ์	1
<b>บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง</b>	<b>2</b>
2.1 การส่งและรับสัญญาณอินฟราเรด	2
2.1.1 รังสีอินฟราเรด (INFRARED RAYS)	2
2.1.2 การส่งและการรับสัญญาณจาก INFRARED REMOTE CONTROL	3
2.2 บอร์ด ARDUINO MEGA 2560	4
2.2.1 หลักการทำงานของ ARDUINO MEGA 2560	5
2.3 บอร์ด ARDUINO NANO 3.0	6
2.4 มอเตอร์ไฟฟ้ากระแสตรง	7
2.4.1 ความหมายของมอเตอร์ไฟฟ้ากระแสตรง	7
2.4.2 ส่วนประกอบของมอเตอร์ไฟฟ้ากระแสตรง	7
2.4.3 หลักการของมอเตอร์ไฟฟ้ากระแสตรง	8
2.5 ตัวรับแสงอินฟราเรด TSOP 4838	9
2.6 โปรแกรมจำลองการทำงาน PROTEUS	9
2.7 โมดูลควมคุมมอเตอร์ L298N	10
2.8 ไอซีเบอร์ 555	11
2.8.1 โหมดการทำงานของไอซี 555	11
2.9 โพรโทบอร์ด (PROTOBOARD)	12

## สารบัญ (ต่อ)

	หน้า
2.10 ตัวเก็บประจุชนิดอิเล็กโตรไลต์	13
2.11 ตัวเก็บประจุชนิดเซรามิค	13
2.12 ตัวต้านทานแบบฟิล์มโลหะ	14
2.13 ตัวต้านทานแบบโพเทนซิโอมิเตอร์แบบปรับละเอียด	14
2.14 ทรานซิสเตอร์ BD136	15
2.15 ไดโอดเปล่งแสงอินฟราเรด	15
2.16 โมดูลเข็มทิศ HMC5883L	16
2.17 ถ่านไฟฉายขนาด 2A	16
2.18 แบตเตอรี่ขนาด 12V	17
2.19 แบตเตอรี่สำรอง (POWER BANK)	18
2.20 สายไฟจัมป์เปอร์	18
<b>บทที่ 3 การออกแบบและการจัดทำปริญญานิพนธ์</b>	<b>19</b>
3.1 การออกแบบ	19
3.1.1 การออกแบบด้านฮาร์ดแวร์	20
3.1.1.1 ส่วนเซนเซอร์อินฟราเรด	20
3.1.1.2 ส่วนขับเคลื่อนแบบจำลองรถตัดหญ้าอัตโนมัติ	21
3.1.1.3 ส่วนแบบจำลองการตัดหญ้า	26
3.1.2 การออกแบบด้านซอฟต์แวร์	27
3.2 เครื่องมือที่ใช้ในการทดลอง	32
3.3 การจัดเก็บผลการทดลอง	32
3.3.1 ส่วนเซนเซอร์อินฟราเรด	32
3.3.2 ส่วนของตัวรับสัญญาณอินฟราเรด	34
3.3.3 ส่วนของการควบคุมทิศทาง	36
<b>บทที่ 4 ผลการทดลอง</b>	<b>37</b>
4.1 ผลการทดลองส่วนของซอฟต์แวร์	37
4.2 ผลการทดลองส่วนของฮาร์ดแวร์	42

## สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผลและข้อเสนอแนะ	45
5.1 สรุปผล	45
5.2 ข้อเสนอแนะ	45
บรรณานุกรม	46
ภาคผนวก ก CODE ภาษา C	48
ภาคผนวก ข DATASHEET	64



## สารบัญรูป

	หน้า
รูปที่ 2.1 Electromagnetic Spectrum	2
รูปที่ 2.2 วงจรการส่งสัญญาณอินฟราเรด	3
รูปที่ 2.3 มอเตอร์ไฟฟ้ากระแสตรง	8
รูปที่ 2.4 ตัวรับแสงอินฟราเรด TSOP 4838	9
รูปที่ 2.5 จำลองการทำงานของวงจรส่งอินฟราเรดบนโปรแกรม Proteus	10
รูปที่ 2.6 ส่วนประกอบของโมดูลควบคุมมอเตอร์	10
รูปที่ 2.7 ไอซีเบอร์ 555	12
รูปที่ 2.8 โพรโทบอร์ด	12
รูปที่ 2.9 ตัวเก็บประจุชนิดอิเล็กโทรไลต์	13
รูปที่ 2.10 ตัวเก็บประจุชนิดเซรามิก	13
รูปที่ 2.11 ตัวต้านทานแบบฟิล์มโลหะ	14
รูปที่ 2.12 ตัวต้านทานแบบโพเทนชิโอมิเตอร์แบบปรับละเอียด	14
รูปที่ 2.13 ทรานซิสเตอร์ BD136	15
รูปที่ 2.14 ไดโอดเปล่งแสงอินฟราเรด	16
รูปที่ 2.15 โมดูลเข็มทิศ HMC5883L	16
รูปที่ 2.16 ส่วนประกอบของถ่านไฟฉาย	17
รูปที่ 2.17 ถ่านไฟฉายขนาด 2A	17
รูปที่ 2.18 แบตเตอรี่ขนาด 12V	17
รูปที่ 2.19 แบตเตอรี่สำรอง	18
รูปที่ 2.20 สายไฟจัมพ์เปอร์สำเร็จรูป	18
รูปที่ 3.1 บล็อกไดอะแกรมรวมของรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์	19
รูปที่ 3.2 การจำลองการทำงานของวงจรตัวส่งสัญญาณอินฟราเรดบนโปรแกรม Proteus	20
รูปที่ 3.3 ออกแบบลายวงจรตัวส่งสัญญาณอินฟราเรด	20
รูปที่ 3.4 วงจรตัวส่งสัญญาณอินฟราเรด	21
รูปที่ 3.5 ตัวส่งสัญญาณอินฟราเรดสำเร็จรูป	21
รูปที่ 3.6 ทดสอบการควบคุมมอเตอร์โดยใช้แรงดันไฟฟ้า 12V และใช้กระแสไฟฟ้า 90 MA	22

## สารบัญรูป (ต่อ)

	หน้า
รูปที่ 3.7 ทดสอบการเคลื่อนที่ของรถจำลอง	22
รูปที่ 3.8 การทำงานของบอร์ด Arduino ประมวลผลการทำงานร่วมกับโมดูลเข็มทิศ	23
รูปที่ 3.9 คำสั่งการใช้งานการเก็บค่าองศาทิศทางของโมดูลเข็มทิศ	23
รูปที่ 3.10 คำสั่งแสดงผลการเก็บค่าองศาทิศทางจากโมดูลเข็มทิศ	24
รูปที่ 3.11 ไดโอดปรากฏแสงสีเหลืองขึ้นแสดงการเก็บค่าองศาทิศทางเอาไว้	24
รูปที่ 3.12 ไดโอดสีแดงและสีเขียวปรากฏแสงขึ้นแสดงถึงการเริ่มการทำงาน	25
รูปที่ 3.13 ไดโอดเปล่งแสงทำงานพร้อมกันเมื่อตัวรับสัญญาณได้รับสัญญาณอินฟราเรด	25
รูปที่ 3.14 ส่วนของมอเตอร์และใบมีดตัดหญ้า	26
รูปที่ 3.15 การทำงานของแบบจำลองรถตัดหญ้าโดยวิ่งตัดหญ้าในบริเวณที่กำหนด	26
รูปที่ 3.16 ภาพรวมโครงสร้างทั้งหมดในส่วนฮาร์ดแวร์	27
รูปที่ 3.17 คำสั่งการกำหนดเงื่อนไขเมื่อมีการเคลื่อนที่	27
รูปที่ 3.18 คำสั่งหยุดการทำงานเมื่อได้รับสัญญาณอินฟราเรด	28
รูปที่ 3.19 คำสั่งการเก็บค่าองศาจากโมดูลเข็มทิศ	28
รูปที่ 3.20 คำสั่งควบคุมความเร็วและทิศทางของมอเตอร์	29
รูปที่ 3.21 คำสั่งควบคุมการเลี้ยวซ้ายและเลี้ยวขวา	29
รูปที่ 3.22 คำสั่งการแสดงผล LED เพื่อควบคุมการใช้งาน	29
รูปที่ 3.23 คำสั่งหยุดการทำงานเมื่อเคลื่อนที่ครบตามพื้นที่ที่กำหนดไว้	30
รูปที่ 3.24 การถอดค่าสัญญาณอินฟราเรดเพื่อประมวลผล	30
รูปที่ 3.25 Flow Chart การทำงานของแบบจำลองรถตัดหญ้าอัตโนมัติ	31
รูปที่ 3.26 ทดสอบการทำงานของวงจรตัวส่งสัญญาณอินฟราเรดบนโปรแกรม Proteus	32
รูปที่ 3.27 สัญญาณด้านวงจรตัวส่งสัญญาณอินฟราเรดที่ความถี่ 39.88 KHZ	33
รูปที่ 3.28 สัญญาณดิจิตอล 500HZ ที่มีดิเฟล็กเข้าไปในวงจรตัวส่งสัญญาณอินฟราเรด	33
รูปที่ 3.29 วัฏระยะการส่งสัญญาณอินฟราเรดได้ 10 เมตร	34
รูปที่ 3.30 วงจรตัวรับสัญญาณอินฟราเรด	34
รูปที่ 3.31 สัญญาณดิจิตอล 500 HZ ที่ได้จากวงจรตัวส่งอินฟราเรดโดยไม่มีสิ่งกีดขวาง มาบ่งระหว่างตัวรับกับตัวส่ง	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

	หน้า
รูปที่ 3.32 สัญญาณที่ได้รับจากวงจรตัวส่งอินฟราเรดเมื่อมีสิ่งกีดขวางระหว่างตัวรับกับตัวส่ง	35
รูปที่ 3.33 อพโทลด์คำสั่งควบคุมการทำงานของไมโครเคมีทิส	36
รูปที่ 3.34 ผลการทำงานเบื้องต้นของไมโครเคมีทิสที่เปลี่ยนองศาทิศทางเมื่อลองหมุนเคมีทิส	36
รูปที่ 4.1 ผลลัพธ์แสดงการเก็บค่าองศาทิศทางทั้งหมด 4 ทิศ	37
รูปที่ 4.2 ค่าองศาทิศทางที่ไมโครเคมีทิสอ่านได้เมื่อเริ่มการทำงาน	38
รูปที่ 4.3 ผลลัพธ์แสดงการเคลื่อนที่ไปข้างหน้า	38
รูปที่ 4.4 ผลลัพธ์แสดงการหยุดเมื่อตรวจรับเจอสัญญาณอินฟราเรด	39
รูปที่ 4.5 ผลลัพธ์แสดงการเคลื่อนที่เมื่อเลี้ยวไปในทิศทางใหม่	39
รูปที่ 4.6 ผลลัพธ์แสดงการเคลื่อนที่ไปข้างหน้าในเส้นทางใหม่เป็นมุมองศา 233.54	40
รูปที่ 4.7 ผลลัพธ์แสดงการเคลื่อนที่เมื่อได้รับสัญญาณอินฟราเรด	41
รูปที่ 4.8 ผลลัพธ์แสดงการเคลื่อนที่เมื่อแบบจำลองรถตัดหญ้าเลี้ยวซ้าย	41
รูปที่ 4.9 ผลลัพธ์แสดงการเคลื่อนที่เมื่อแบบจำลองรถตัดหญ้าทำงานเสร็จสมบูรณ์	42
รูปที่ 4.10 เริ่มการเคลื่อนที่ของแบบจำลองรถตัดหญ้าอัตโนมัติ	42
รูปที่ 4.11 การเคลื่อนที่กลับไปอีกด้านหนึ่ง	43
รูปที่ 4.12 แบบจำลองรถตัดหญ้าอัตโนมัติเลี้ยวซ้ายเพื่อเคลื่อนที่ไปยังพื้นที่ถัดไป	43
รูปที่ 4.13 การทำงานของแบบจำลองรถตัดหญ้าอัตโนมัติเสร็จสิ้นสมบูรณ์	44
รูปที่ 4.14 ภาพจำลองแสดงการเคลื่อนที่ทั้งหมดของแบบจำลองรถตัดหญ้าอัตโนมัติ	44

## สารบัญตาราง

	หน้า
ตารางที่ 2.1 คุณสมบัติของ Arduino Mega 2560	4
ตารางที่ 2.2 คุณสมบัติของ Arduino Nano 3.0	6
ตารางที่ 2.3 หน้าที่ของขาไอซี 555	11



## บทที่ 1

### บทนำ

#### 1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันการใช้เทคโนโลยีเข้ามามีส่วนเกี่ยวข้องในชีวิตประจำวันของเรามากขึ้นเพื่ออำนวยความสะดวกไม่ว่าจะเป็นเทคโนโลยีที่ใช้ภายในบ้านหรือนอกบ้าน รถตัดหญ้าเป็นเครื่องมือหนึ่งที่อำนวยความสะดวกให้กับผู้ใช้งานในการตัดสนามหญ้าให้เป็นไปตามที่ต้องการ ซึ่งโดยทั่วไปรถตัดหญ้าที่ทำงานอัตโนมัติจะทำงานแบบไร้ทิศทางและไม่มีกำหนดพื้นที่ที่แน่นอน โครงการนี้จึงได้จัดทำแบบจำลองรถตัดหญ้าอัตโนมัติขึ้นเพื่อจำลองการใช้งานของรถตัดหญ้าจริง ซึ่งสามารถโปรแกรมให้ทำงานได้อัตโนมัติเคลื่อนที่ไปในทิศทางที่เรากำหนดได้โดยประมวลผลการทำงานร่วมกับไมโครคอนโทรลเลอร์ ทำให้มีความแม่นยำมากขึ้นในการเคลื่อนที่ได้ตามมุมมองทิศทางที่เราเก็บค่าไว้ และสามารถทำงานภายในพื้นที่ที่กำหนดจากการใช้งานของอินฟราเรดเซนเซอร์ที่เป็นตัวกำหนดขอบเขตพื้นที่ที่ใช้งาน โดยแบบจำลองรถตัดหญ้าจะมีตัวรับสัญญาณ TSOP4838 เป็นตัวตรวจจับสัญญาณอินฟราเรด

โดยการทำงานทั้งหมดของแบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติทำงานภายใต้การควบคุมของชุดคำสั่งไมโครคอนโทรลเลอร์ที่เขียนโค้ดเก็บไว้บน Arduino เพื่อที่จะสามารถทำงานให้เป็นไปตามเงื่อนไขที่กำหนดไว้

#### 1.2 วัตถุประสงค์

1. เพื่อศึกษาการเขียนชุดคำสั่งไมโครคอนโทรลเลอร์ในการบังคับทิศทางรถตัดหญ้า
2. เพื่อศึกษาหลักการการทำงานของรถตัดหญ้าอัตโนมัติ
3. เพื่อศึกษาหลักการการทำงานของเซนเซอร์อินฟราเรดและไมโครคอนโทรลเลอร์
4. สามารถสร้างแบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติให้ทำงานในพื้นที่ที่กำหนดได้

#### 1.3 ขอบเขตของปริญญานิพนธ์

ปริญญานิพนธ์นี้ได้ศึกษาหลักการการทำงานของรถตัดหญ้าอัตโนมัติแล้วนำมาประยุกต์สร้างแบบจำลองรถตัดหญ้าขึ้นซึ่งทำงานได้โดยอัตโนมัติตามที่เราได้โปรแกรมไว้ผ่านชุดควบคุมไมโครคอนโทรลเลอร์ให้สามารถเคลื่อนที่ไปตามทิศทางที่ต้องการและในพื้นที่ที่กำหนดไว้

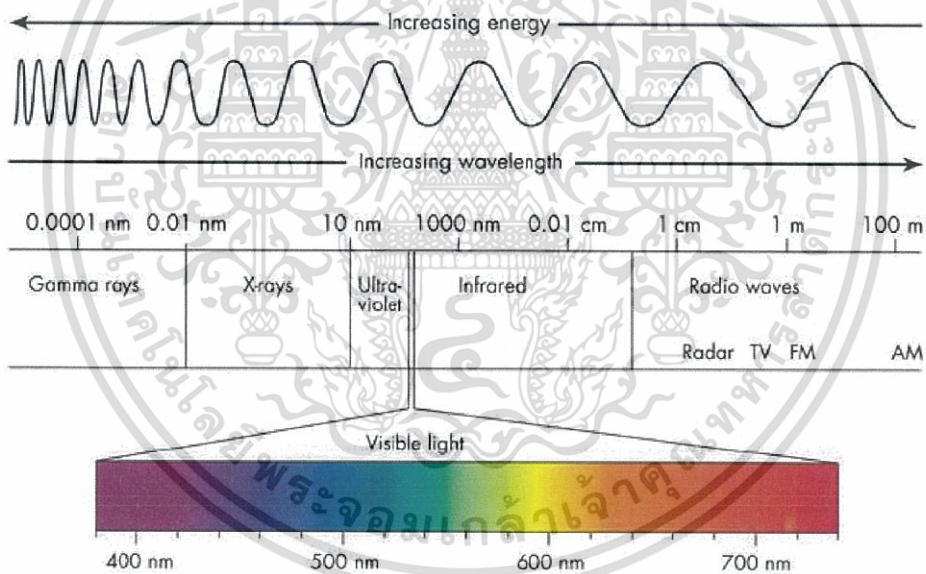
## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

#### 2.1 การส่งและรับสัญญาณอินฟราเรด

##### 2.1.1 รังสีอินฟราเรด (infrared rays)

คลื่นแม่เหล็กไฟฟ้าที่มีความถี่อยู่ในช่วง  $10^{11}$  -  $10^{14}$  เฮิร์ตซ์หรือความยาวคลื่นประมาณ 1mm-1 $\mu$ m เรียกว่า รังสีอินฟราเรด หรือเรียกอีกอย่างหนึ่งว่า คลื่นความถี่สั้น (Millimeter waves) และยังเป็นคลื่น แม่เหล็กไฟฟ้าที่มีความถี่ ต่ำกว่าแสงที่ตามองเห็น ลำแสงอินฟราเรดเดินทางเป็นเส้นตรงไม่สามารถผ่านวัตถุทึบแสง และสามารถสะท้อนแสงในวัสดุผิวเรียบได้เหมือนกับแสงทั่วไปใช้มากในการสื่อสารระยะใกล้ คลื่นสั้นนั้นจะไม่มีปัญหาในเรื่องการ Interference (การรบกวนของสัญญาณ)



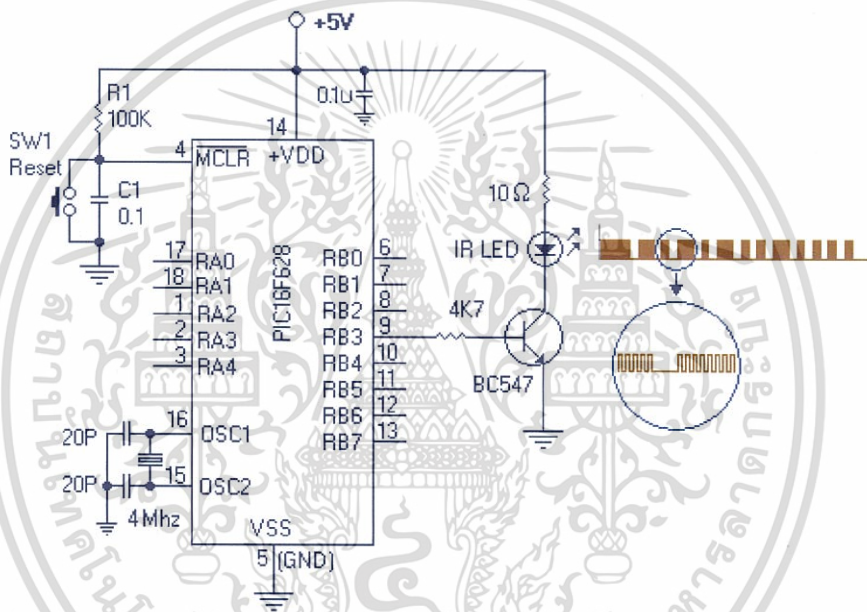
รูปที่ 2.1 Electromagnetic Spectrum [1]

เหมาะแก่การใช้งานจึงมีการประยุกต์นำมาใช้กันอย่างแพร่หลายในปัจจุบันโดยส่วนใหญ่เป็นอุปกรณ์ทางด้าน Electronics และการสื่อสารเช่น Remote Controller ของ TV โทรศัพท์มือถือ และกล้องส่องทางไกล เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.2 การส่งและการรับสัญญาณจาก Infrared Remote Control

การส่งสัญญาณ Infrared จาก Infrared Sensor การประยุกต์ใช้คลื่นอินฟราเรดจะเป็นการประยุกต์ใช้ในการสื่อสารแบบไร้สาย (Wireless Communication) ในการควบคุมอุปกรณ์เครื่องใช้ไฟฟ้า โดยการส่งสัญญาณผ่านทาง LED (Light Emitting Diode) โดยตัวส่ง (Transmitter) หรือ Laser Diode และจะมีตัวรับ (Receiver) ทำการแปลงสัญญาณข้อมูลให้กลับไปเป็นเหมือนข้อมูลเริ่มแรกหรือข้อมูลต้นทางนั่นเอง



รูปที่ 2.2 วงจรการส่งสัญญาณอินฟราเรด [2]

จากรูปสัญญาณอินฟราเรดจะถูกส่งออกทาง IR LED โดยในการส่งสัญญาณนั้น เพื่อที่จะหลีกเลี่ยงการถูกรบกวนจากแสงแหล่งต่างๆ หรือเพื่อไม่ให้เกิดการซ้ำกันในการส่งสัญญาณ Infrared จึงมีการกำหนด หรือสร้างรูปแบบของสัญญาณที่จะส่งออกไปเรียกว่า การ Coding ซึ่งสัญญาณที่ได้จะถูกส่งออกไปในรูปของ Digital Pulse ซึ่งมีความถี่ที่แน่นอนอยู่ค่าหนึ่งอยู่ในช่วง

30 – 60 kHz

## 2.2 บอร์ด Arduino Mega 2560

Arduino Mega 2560 เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ทำงานบนพื้นฐานของ ATmega2560 โดยเป็นไมโครคอนโทรลเลอร์ที่จำเป็นต้องใช้การต่อไฟเลี้ยงซึ่งสามารถทำได้ทั้งการเชื่อมต่อเข้ากับ USB cable หรือ จ่ายไฟด้วย AC - DC adapter หรือ การใช้แบตเตอรี่ ซึ่ง Mega 2560 เป็นบอร์ดที่เข้ากันได้กับ shield ที่ออกแบบมาเพื่อ Arduino Duemilanove หรือ Diecimila ซึ่งประกอบด้วย [3]

- 54 digital input/output pins (15 pin สามารถใช้เป็น PWM output ได้)
- 16 analog inputs
- 4 UARTs
- 16 MHz crystal oscillator (ใช้สำหรับกรองความถี่ให้กับบอร์ดไมโครคอนโทรลเลอร์)
- USB connection
- ช่องเสียบแหล่งจ่าย
- ICSP header : In-Circuit Serial Programming (ส่วนที่เป็น AVR ขนาดเล็กสำหรับ  
การโปรแกรม Arduino ซึ่งประกอบด้วย MOSI, MISO, SCK, RESET, VCC,GND )
- ปุ่มกด reset

ตารางที่ 2.1 คุณสมบัติของ Arduino Mega 2560 [4]

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 kB of which 8 kB used by bootloader
SRAM	8 kB
EEPROM	4 kB
Clock Speed	16 MHz
Length	101.52 mm
Width	53.3 mm
Weight	37 g

### 2.2.1 หลักการทำงานของ Arduino Mega 2560

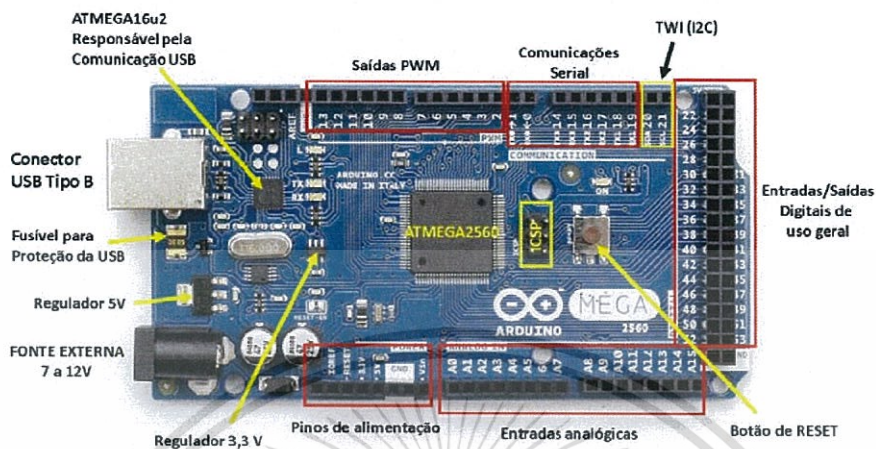
Arduino Mega สามารถเชื่อมรับพลังงานโดยการเชื่อมต่อ micro USB connector หรือ จาก power supply จากภายนอกได้ โดยแหล่งพลังงานจะถูกเลือกโดยอัตโนมัติแหล่งจ่ายจากภายนอกสามารถมาได้จาก AC – to - DC adapter หรือจากแบตเตอรี่ โดยต่อเข้ากับ 2.1 mm center-positive plug ไปยังช่องเสียบแหล่งจ่าย และการต่อเข้ากับแบตเตอรี่สามารถทำได้โดยการต่อเข้ากับ GND และ Vin pin header ของ power connector

บอร์ดสามารถทำงานได้ในช่วงแรงดัน 6 ถึง 20 Volts ถ้า แหล่งจ่ายมีค่าต่ำกว่า 7 V อาจส่งผลให้ 5 V pin มีแรงดันที่ต่ำกว่า 5V และ บอร์ดอาจจะไม่เสถียร แต่ถ้าหากแรงดันมีค่าสูงกว่า 12 V อาจส่งผลให้บอร์ด Overheat และอาจทำให้บอร์ดเสียหายได้ ดังนั้นช่วงแรงดันที่เหมาะสมกับบอร์ด คือ 7 V ถึง 12V

- VIN เป็น input voltage ของบอร์ด Arduino โดยใช้แหล่งจ่ายจากภายนอก
- 5V เป็น output pin ที่ควบคุม 5 V จากบอร์ด
- 3V3 เป็น 3.3 volt supply ที่สร้างขึ้นจาก regulator บนบอร์ด และให้กระแสได้สูงสุด 50 mA
- GND เป็น ground pin
- IOREF เป็น pin ที่ให้ voltage reference กับไมโครคอนโทรลเลอร์ เพื่อเลือกค่าแรงดันให้กับ shield ที่มาเชื่อมต่อกับบอร์ด

ATmega2560 มีหน่วยความจำ 256 kB (8 kB ใช้สำหรับ bootloader ) นอกจากนี้ยังมีอีก 8 kB สำหรับ SRAM และ 4 kB สำหรับ EEPROM ในแต่ละ digital pins ทั้ง 54 pins บนบอร์ด Arduino Uno สามารถเป็นได้ทั้ง input และ output โดยจะทำงานที่แรงดัน 5 V และให้กระแสสูงสุด 40 mA

Arduino Uno สามารถสื่อสารกับคอมพิวเตอร์, Arduino ตัวอื่นๆ หรือ microcontroller ได้ โดยที่ไมโครคอนโทรลเลอร์บนบอร์ด คือ ATmega32U4 จะให้การสื่อสารแบบอนุกรม UART TTL (5 V) ซึ่งมีอยู่ใน pins 0 (Rx) และ 1 (Tx) นอกจากนี้ 32U4 สามารถใช้การสื่อสารแบบอนุกรมผ่าน USB และจะปรากฏเป็น COM port เสมือนไปยัง Software แต่อย่างไรก็ตามต้องใช้ไฟล์ .inf บนระบบปฏิบัติการ Windows แต่ OSX และ Linux สามารถ recognize ได้โดยอัตโนมัติ [5]



รูปที่ 2.3 ส่วนประกอบต่างๆ บนบอร์ด Arduino mega 2560 [5]

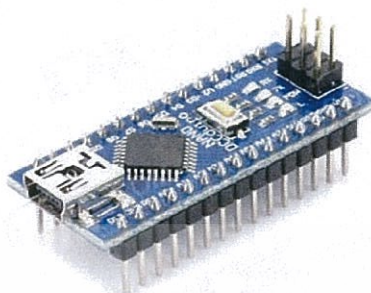
### 2.3 บอร์ด Arduino Nano 3.0

Arduino Nano 3.0 ใช้ชิพ CH340G เป็นชิพ USB Driver เป็น Arduino ที่ใช้หน่วยประมวลผล ATmega328 Arduino Nano สามารถโปรแกรมได้โดยตรงผ่าน USB port มีจำนวน input/output น้อย

ตารางที่ 2.2 คุณสมบัติของ ARDUINO NANO 3.0

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage	5 - 12 V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	8
DC Current per I/O Pin	40 mA
Flash Memory	32 kB (of which 2 kB used by bootloader)
SRAM	2 kB
EEPROM	1 kB
Clock Speed	16 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 Arduino Nano 3.0 [6]

## 2.4 มอเตอร์ไฟฟ้ากระแสตรง

### 2.4.1 ความหมายของมอเตอร์ไฟฟ้ากระแสตรง

มอเตอร์ไฟฟ้ากระแสตรง เป็นต้นกำลังขับเคลื่อนที่สำคัญอย่างหนึ่งในโรงงานอุตสาหกรรม เพราะมีคุณสมบัติที่เด่นในด้านการปรับความเร็วได้ตั้งแต่ความเร็วต่ำสุดจนถึงสูงสุด นิยมใช้กันมากในโรงงานอุตสาหกรรม เช่น โรงงานทอผ้า โรงงานเส้นใยโพลีเอสเตอร์ โรงงานถลุงโลหะหรือให้ เป็นต้นกำลังในการขับเคลื่อนรถไฟฟ้า เป็นต้นในการศึกษาเกี่ยวกับมอเตอร์ไฟฟ้ากระแสตรงจึงควรรู้จัก อุปกรณ์ต่าง ๆ ของมอเตอร์ไฟฟ้ากระแสตรงและเข้าใจถึงหลักการทำงานของมอเตอร์ไฟฟ้ากระแสตรงแบบต่าง ๆ

### 2.4.2 ส่วนประกอบของมอเตอร์ไฟฟ้ากระแสตรง

มอเตอร์ไฟฟ้ากระแสตรงมีส่วนประกอบที่สำคัญ 2 ส่วนดังนี้

#### 1. ส่วนที่อยู่กับที่หรือที่เรียกว่าสเตเตอร์ (Stator) ประกอบด้วย

1.1 เฟรมหรือโยค (Frame or Yoke) เป็นโครงภายนอกทำหน้าที่เป็นทางเดินของเส้นแรงแม่เหล็กจากขั้วเหนือไปขั้วใต้ให้ครบวงจรและยึดส่วนประกอบอื่นๆ ให้แข็งแรงทำด้วยเหล็กหล่อหรือเหล็กแผ่นหนาม้วนเป็นรูปทรงกระบอก

1.2 ขั้วแม่เหล็ก (Pole) ประกอบด้วย 2 ส่วนคือแกนขั้วแม่เหล็กและขดลวดส่วนแรกแกนขั้ว (Pole Core) ทำด้วยแผ่นเหล็กบางๆ กั้นด้วยฉนวนประกบกันเป็นแท่งยึดติดกับเฟรมส่วนปลายที่ทำเป็นรูปโค้งนั้นเพื่อโค้งรับรูปกลมของตัวโรเตอร์เรียกว่า ขั้วแม่เหล็ก (Pole Shoes) มีวัตถุประสงค์ให้ขั้วแม่เหล็กและโรเตอร์ใกล้ชิดกันมากที่สุดเพื่อให้เกิดช่องอากาศน้อยที่สุด เพื่อให้เกิดช่องอากาศน้อยที่สุดจะมีผลให้เส้นแรงแม่เหล็กจากขั้วแม่เหล็กจากขั้วแม่เหล็กผ่านไปยังโรเตอร์มากที่สุดแล้วทำให้เกิดแรงบิดหรือกำลังบิดของโรเตอร์มากเป็นการทำให้มอเตอร์ มีกำลังหมุน (Torque )

ส่วนที่สอง ขดลวดสนามแม่เหล็ก (Field Coil) จะพันอยู่รอบๆ แกนขั้วแม่เหล็กขดลวดนี้ทำหน้าที่รับกระแสจากภายนอกเพื่อสร้างเส้นแรงแม่เหล็กให้เกิดขึ้น และเส้นแรงแม่เหล็กนี้จะเกิดการหักล้างและเสริมกันกับสนามแม่เหล็กของอาเมเจอร์ทำให้เกิดแรงบิดขึ้น

2. ตัวหมุน (Rotor) ตัวหมุนหรือเรียกว่าโรเตอร์ตัวหมุนนี้ทำให้เกิดกำลังงานมีแกนวางอยู่ในตลับลูกปืน (Ball Bearing) ซึ่งประกอบอยู่ในแผ่นปิดหัวท้าย (End Plate) ของมอเตอร์ ตัวโรเตอร์ประกอบด้วย 4 ส่วนด้วยกัน คือ

2.1 แกนเพลลา (Shaft)

2.2 แกนเหล็กอาร์มาเจอร์ (Armature Core)

2.3 คอมมิวเตเตอร์ (Commutator)

2.4 ขอลวดอาร์มาเจอร์ (Armature Winding)

#### 2.4.3 หลักการของมอเตอร์ไฟฟ้ากระแสตรง

หลักการของมอเตอร์ไฟฟ้ากระแสตรง (Motor Action) เมื่อเป็นแรงดันกระแสไฟฟ้าตรงเข้าไปในมอเตอร์ ส่วนหนึ่งจะ แปรงผ่านผ่านคอมมิวเตเตอร์เข้าไปในขดลวดอาร์มาเจอร์สร้างสนามแม่เหล็กขึ้น และกระแสไฟฟ้าอีกส่วนหนึ่งจะไหลเข้าไปในขดลวดสนามแม่เหล็ก (Field coil) สร้างขั้วเหนือ-ใต้ขึ้น จะเกิดสนามแม่เหล็ก 2 สนาม ในขณะเดียวกัน ตามคุณสมบัติของเส้นแรงแม่เหล็ก จะไม่ตัดกันทิศทางตรงข้าม จะหักล้างกัน และทิศทางเดียวจะเสริมแรงกัน ทำให้เกิดแรงบิดในตัวอาร์มาเจอร์ ซึ่งวางแกนเพลลาและแกนเพลลานี้สวมอยู่กับตลับลูกปืนของมอเตอร์ ทำให้อาร์มาเจอร์นี้หมุนได้ ขณะที่ตัวอาร์มาเจอร์ทำหน้าที่หมุนได้นี้เรียกว่า โรเตอร์ (Rotor) ซึ่งหมายความว่าตัวหมุน การที่อำนาจเส้นแรงแม่เหล็กทั้งสองมีปฏิกิริยาต่อกันทำให้ขดลวดอาร์มาเจอร์ หรือโรเตอร์หมุนไปนั้นเป็นไปตามกฎซ้ายของเฟลมมิ่ง (Fleming's left hand rule)



รูปที่ 2.3 มอเตอร์ไฟฟ้ากระแสตรง [7]

## 2.5 ตัวรับแสงอินฟราเรด TSOP 4838

อุปกรณ์ตัวรับสำหรับแสงอินฟราเรด เช่น รุ่น TSOP4838 ทำงานในลักษณะคล้ายโฟโตทรานซิสเตอร์ (Phototransistor) มี 3 ขา คือ OUT, GND, VCC สามารถใช้กับแรงดัน 5V ได้ เมื่อได้รับแสงอินฟราเรด จะให้เอาต์พุตเป็น LOW แต่ถ้าไม่ได้รับแสง จะให้เอาต์พุตเป็น HIGH เพื่อป้องกันแสงรบกวน อุปกรณ์ประเภทนี้ได้ถูกออกแบบมาใช้งานสำหรับแสงอินฟราเรดที่เป็นสัญญาณแบบ Pulse Train ความถี่ เช่น 38 kHz



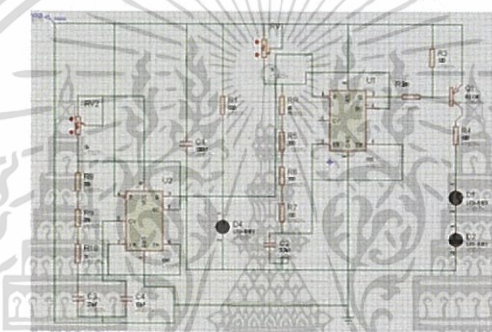
รูปที่ 2.4 ตัวรับแสงอินฟราเรด TSOP 4838 [8]

## 2.6 โปรแกรมจำลองการทำงาน PROTEUS

โปรแกรม Proteus เป็นโปรแกรมที่มีความสามารถมากอีกโปรแกรมหนึ่ง ในงานด้านอิเล็กทรอนิกส์ เพราะสามารถออกแบบวงจรไฟฟ้า พร้อมทั้งจำลองการทำงานของวงจรได้ ทั้งยังสามารถออกแบบลายวงจรพิมพ์ได้ ความสามารถที่โดดเด่นของ Proteus นั้น จะกล่าวได้ว่าเป็นโปรแกรมที่สามารถจำลองพฤติกรรม (Simulator) การทำงานของวงจรที่ใช้ Microcontroller เบอร์ต่าง ๆ ได้มากมาย โดยไม่ต้องประกอบวงจรให้เสียเวลา เพื่อพิสูจน์ว่าโปรแกรมที่เขียนขึ้นใช้งานได้หรือไม่

โปรแกรม Proteus หรือ Proteus VSM (Virtual System Modeling) เป็นโปรแกรมที่พัฒนาขึ้น โดยบริษัท แล็บเซ็นเตอร์อิเล็กทรอนิกส์ จำกัด (Labcenter Electronics Ltd.) ที่ประเทศอังกฤษ โปรแกรม Proteus มีชื่อเต็มว่า Labcenter Electronics Proteus ซึ่งภายในโปรแกรมจะประกอบด้วยส่วนประกอบหลัก 2 ส่วน คือ ISIS และ ARES โปรแกรม Proteus จะมีอยู่หลายเวอร์ชันให้เลือกใช้งานความสามารถในการทำงานของโปรแกรม Proteus ก็คือ สามารถจำลองการทำงานของวงจรอิเล็กทรอนิกส์ได้หลากหลายรูปแบบ ไม่ว่าจะเป็นแบบอนาล็อกและแบบดิจิตอล หรือทั้งแบบอนาล็อกและดิจิตอลผสมกัน นอกจากนี้ Proteus ยังสามารถออกแบบลายวงจรพิมพ์ (PCB) ได้อีกด้วย จุดเด่นของโปรแกรม Proteus ที่เป็นที่นิยมและชื่นชอบก็คือ การจำลองการทำงานของวงจรอิเล็กทรอนิกส์ ที่ใช้

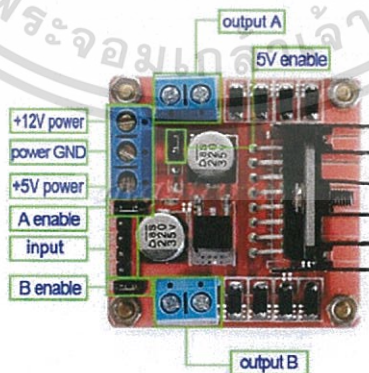
ไมโครคอนโทรลเลอร์ตระกูลต่าง ๆ ไม่ว่าจะเป็น PIC, MCS-51, AVR และ ARM เป็นต้น ทำให้นักเขียนโปรแกรมหรือโปรแกรมเมอร์สามารถตรวจสอบได้ว่าโปรแกรม หรือซอสโค้ด (Source Code) ที่เขียนขึ้นมา นั้น สามารถสนับสนุนกับวงจรฮาร์ดแวร์ที่ต่อได้หรือไม่ ถ้าโปรแกรม (Source Code) ที่เขียนขึ้น ไม่สนับสนุนกับวงจรฮาร์ดแวร์ที่ต่อโปรแกรมเมอร์ก็จะทำการพัฒนาโปรแกรม (Source Code) ที่เขียนขึ้นใหม่ หรือปรับปรุงวงจรฮาร์ดแวร์ใน Proteus จนกว่าโปรแกรมที่เขียนขึ้นและฮาร์ดแวร์ที่ต่อ สามารถสนับสนุนซึ่งกันและกัน ทำให้การสร้างโครงการต่าง ๆ สามารถประหยัดเวลาและค่าใช้จ่ายเป็นอย่างมาก เพราะในอดีตการเขียนโปรแกรมขึ้นมา นั้น จะต้องต่อวงจรจริงเพื่อทดสอบ ทำให้เสียเวลาและค่าใช้จ่ายมาก ในกรณีที่วงจรฮาร์ดแวร์และโปรแกรมที่เขียนขึ้นไม่สนับสนุนซึ่งกันและกัน



รูปที่ 2.5 จำลองการทำงานของวงจรส่งอินฟาเรดบนโปรแกรม Proteus

## 2.7 โมดูลควบคุมมอเตอร์ L298N

โมดูลที่ใช้ในการควบคุมความเร็วและทิศทางของมอเตอร์เชื่อมต่อกับ Arduino ที่เขียนโปรแกรมไว้



รูปที่ 2.6 ส่วนประกอบของโมดูลควบคุมมอเตอร์ [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.8 ไอซีเบอร์ 555

ไอซีไทเมอร์ 555 นับเป็นวงจรรวมที่สามารถใช้งานได้หลากหลายและเป็นที่ยอมรับมากที่สุดตัวหนึ่ง ภายในประกอบด้วยทรานซิสเตอร์ 23 ตัว, ไดโอด 2 ตัว และรีซิสเตอร์อีก 16 ตัว เรียงกันบนชิปซิลิกอนแผ่นเดียว โดยติดตั้งในตัวถัง 8 ขา แบบมินิ DIP (dual-in-line package)

### 2.8.1 โหมดการทำงานของไอซี 555

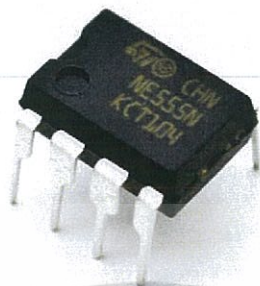
1. โมโนสเตเบิล (Monostable) ในโหมดนี้ การทำงานของ 555 จะเป็นแบบซิงเกิ้ลช็อต หรือวันช็อต (one-shot) โดยการสร้างสัญญาณครั้งเดียว ประยุกต์การใช้งานสำหรับการนับเวลา การตรวจสอบพัลส์ สวิตช์สัมผัส ฯลฯ

2. ออสเตเบิล (Astable) ในโหมดนี้ การทำงานจะเป็นออสซิลเลเตอร์ การใช้งานได้แก่ ทำไฟกะพริบ, กำเนิดพัลส์, กำเนิดเสียง, เตือนภัย ฯลฯ

3. ไบสเตเบิล (Bistable) ในโหมดนี้ ไอซี 555 สามารถทำงานเป็นฟลิปฟล็อป (flip-flop) ถ้าไม่ต่อขา DIS และไม่ใช้คาปาซิเตอร์ ใช้เป็นสวิตช์ bouncefree latched switches เป็นต้น

ตารางที่ 2.3 หน้าที่ของขาไอซี 555

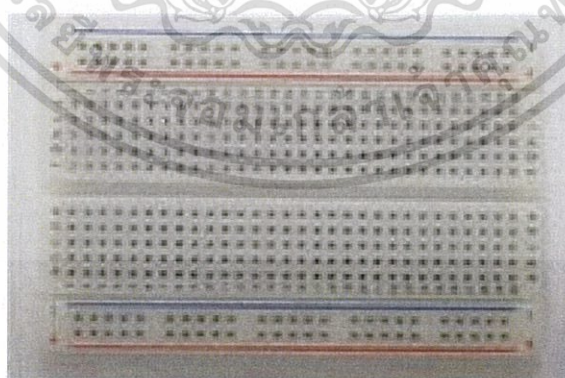
ขา	ชื่อ	หน้าที่
1	GND	กราวด์ หรือ คอมมอนส์
2	TR	พัลส์สั้นกระตุ้นทริกเกอร์เพื่อเริ่มนับเวลา
3	Q	ช่วงการนับเวลา เอาต์พุตจะอยู่ที่ $+V_{CC}$
4	R	ช่วงเวลานับ อาจหยุดโดยการใช้พัลส์รีเซ็ต
5	CV	แรงดันควบคุมยอมให้เข้าถึงตัวหารแรงดันภายใน ( $2/3 V_{CC}$ )
6	THR	เทรชโฮลด์ที่จุดช่วงเวลานับ
7	DIS	เชื่อมต่อกับคาปาซิเตอร์ตัวหนึ่ง ซึ่งเวลาคายประจุของมันจะมีผลต่อช่วงเวลานับ
8	$V+, V_{CC}$	แรงดันจ่ายไฟบวก ซึ่งต้องอยู่ในช่วง $+5$ ถึง $+15$ V



รูปที่ 2.7 ไอซีเบอร์ 555 [10]

## 2.9 โพรโทบอร์ด (Protoboard)

ในการสร้างชิ้นงานทางอิเล็กทรอนิกส์โดยเฉพาะในงานไมโครคอนโทรลเลอร์หลังจากทำการอัดโปรแกรมชิปโปรแกรมไมโครคอนโทรลเลอร์แล้วก็จำเป็นต้องมีการต่อวงจรไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอก เพื่อทดสอบว่าโปรแกรมที่เขียนขึ้นทำให้ไมโครคอนโทรลเลอร์สามารถทำงานได้หรือไม่ ซึ่งโพรโทบอร์ดนี้จะอำนวยความสะดวกในการใช้ทดลองวงจรอิเล็กทรอนิกส์ ลักษณะเป็นแผ่นพลาสติกหนาสีขาว บนแผ่นมีรูเรียงกันจำนวนมาก ภายในรูมีตัวนำไฟฟ้าซึ่งเชื่อมต่อกันในรูปแบบที่มีการกำหนดไว้ เวลาทดลองก็เสียบขาของอุปกรณ์ อิเล็กทรอนิกส์ลงไปให้ตัวนำภายในเชื่อมวงจรถึงกัน และอาจใช้สายไฟเสียบลงรูเพื่อเชื่อมวงจรไฟฟ้าได้เช่นกัน ข้อดีของโพรโทบอร์ดคือ ไม่ต้องออกแบบแผงวงจรและไม่ต้องบัดกรี แต่มีข้อเสียคือใช้ทดลองวงจรที่ทำงานที่ความถี่สูง ๆ ไม่ได้เนื่องจากมีปัญหาเรื่องสัญญาณรบกวนในวงจร



รูปที่ 2.8 โพรโทบอร์ด [11]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.10 ตัวเก็บประจุชนิดอิเล็กโทรไลต์

ตัวเก็บประจุ หรือ คาปาซิเตอร์ เป็นอุปกรณ์อิเล็กทรอนิกส์อย่างหนึ่ง ทำหน้าที่เก็บพลังงานในสนามไฟฟ้า ที่สร้างขึ้นระหว่างคู่ฉนวน โดยมีค่าประจุไฟฟ้าเท่ากัน แต่มีชนิดของประจุตรงข้ามกัน บางครั้งเรียกตัวเก็บประจุนี้ว่า คอนเดนเซอร์ (Condenser) เป็นอุปกรณ์พื้นฐานสำคัญในงานอิเล็กทรอนิกส์ โดยในโครงงานนี้ใช้ตัวเก็บประจุชนิดอิเล็กโทรไลต์ซึ่งเป็นที่นิยมใช้กันมากเพราะให้ค่าความจุสูง มีขั้วบวกลบโดยขาที่ยาวจะเป็นขั้วบวก และขาที่สั้นจะเป็นขั้วลบ เวลาใช้งานต้องติดตั้งให้ถูกขั้ว โครงสร้างภายในคล้ายกับแบตเตอรี่ นิยมใช้กับงานความถี่ต่ำหรือใช้สำหรับไฟฟ้ากระแสตรง มีข้อเสียคือ กระแสรั่วไหลและความผิดพลาดสูงมาก



รูปที่ 2.9 ตัวเก็บประจุชนิดอิเล็กโทรไลต์ [12]

## 2.11 ตัวเก็บประจุชนิดเซรามิก

ตัวเก็บประจุที่มีค่าไม่เกิน 1 ไมโครฟารัด (mF) นิยมใช้กันทั่วไปเพราะมีราคาถูก เหมาะสำหรับวงจรประเภทคัปปลิงความถี่วิทยุ ข้อเสียของตัวเก็บประจุชนิดเซรามิกคือมีการสูญเสียมาก



รูปที่ 2.10 ตัวเก็บประจุชนิดเซรามิก [13]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.12 ตัวต้านทานแบบฟิล์มโลหะ

ตัวต้านทานแบบฟิล์มโลหะทำมาจากแผ่นฟิล์มบางของแก้วและโลหะหลอมเข้าด้วยกันแล้วนำไปเคลือบที่เซรามิค ทำเป็นรูปทรงกระบอก แล้วตัดแผ่นฟิล์มที่เคลือบออกให้ได้ค่าความต้านทานตามที่ต้องการ ขั้นตอนสุดท้ายจะทำการเคลือบด้วยสารอีพ็อกซี (Epoxy) ตัวต้านทานชนิดนี้มีค่าความผิดพลาดบวกลบ 0.1% ถึงประมาณบวกลบ 2% ซึ่งถือว่ามีค่าความผิดพลาดน้อยมาก นอกจากนี้ยังทนต่อการเปลี่ยนแปลงอุณหภูมิจากภายนอกได้ดี สัญญาณรบกวนน้อยเมื่อเทียบกับตัวต้านทานชนิดอื่น ๆ



รูปที่ 2.11 ตัวต้านทานแบบฟิล์มโลหะ [14]

## 2.13 ตัวต้านทานแบบโพเทนซิโอมิเตอร์แบบปรับละเอียด

ตัวต้านทานแบบนี้ส่วนมากมักใช้ประกอบในวงจรประเภทเครื่องมือวัดและทดสอบ เพราะสามารถปรับหมุน เพื่อต้องการเปลี่ยนค่าความต้านทานได้ที่ละเอียด และสามารถหมุนได้ 15 รอบหรือมากกว่า



รูปที่ 2.12 ตัวต้านทานแบบโพเทนซิโอมิเตอร์แบบปรับละเอียด [15]

## 2.14 ทรานซิสเตอร์ BD136

ทรานซิสเตอร์เป็นอุปกรณ์ซึ่งถูกควบคุมด้วยกระแสไฟฟ้าที่ผ่านขา B หรือเรียกว่า กระแสเบส นั่นคือ เมื่อกระแสเบสเปลี่ยนแปลงเพียงเล็กน้อยก็จะทำให้กระแสไฟฟ้าในขา E (กระแสอิมิตเตอร์) และ กระแสไฟฟ้าในขา C (กระแสคอลเล็กเตอร์) เปลี่ยนแปลงไปด้วย ซึ่งทำให้ทรานซิสเตอร์ทำหน้าที่เป็นสวิตช์ ปิดหรือเปิดวงจร โดยถ้าไม่มีกระแสไฟฟ้าผ่านขา B ก็จะทำให้ไม่มีกระแสไฟฟ้าผ่านขา E และ C ด้วย ซึ่ง เปรียบเสมือนปิดไฟ (วงจรเปิด) แต่ถ้าให้กระแสไฟฟ้าเพียงเล็กน้อยผ่านขา B จะสามารถควบคุม กระแสไฟฟ้าที่มากกว่าให้ผ่านทรานซิสเตอร์แล้วผ่านไปยังขา E และผ่านไปยังอุปกรณ์อื่นที่ต่อจากขา C



รูปที่ 2.13 ทรานซิสเตอร์ BD136 [16]

## 2.15 ไดโอดเปล่งแสงอินฟราเรด

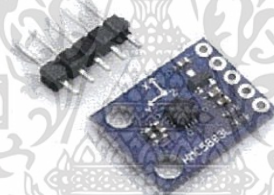
การรับส่งข้อมูลด้วยแสงอินฟราเรด (Infrared: IR) เป็นวิธีการหนึ่งในการสื่อสารข้อมูล (Data Communication) แบบไร้สาย ตัวอย่างการประยุกต์ใช้งานที่พบเห็นได้แพร่หลายคือ อุปกรณ์รีโมท ควบคุม (Remote Control) บทความนี้จะกล่าวถึง การสร้างวงจรพื้นฐานในการรับส่งข้อมูลด้วยแสง อินฟราเรด และเขียนโปรแกรม Arduino เพื่อส่งและรับข้อมูลผ่านวงจรดังกล่าว ไดโอดเปล่งแสง อินฟราเรดหรือ IR LED เป็นอุปกรณ์สำคัญของตัวส่ง ให้แสงในช่วงคลื่นอินฟราเรด (มองด้วยตาเปล่าไม่ เห็น) และให้ความเข้มแสงสูงสุดที่เฉพาะค่าความถี่เท่านั้น LED ประเภทนี้มีลักษณะเหมือน LED ทั่วไป มี 2 ขา คือ แอนโอด กับ แคโทด ดังนั้นการต่อใช้งาน ก็เหมือนกรณี LED ทั่วไป LED ที่ให้แสงอินฟราเรดแต่ ละชนิด สามารถทนกระแสสูงสุด (MA) ได้แตกต่างกัน



รูปที่ 2.14 ไดโอดเปล่งแสงอินฟราเรด [17]

## 2.16 โมดูลเข็มทิศ HMC5883L

เข็มทิศอิเล็กทรอนิกส์ที่ใช้เชื่อมต่อกับไมโครคอนโทรลเลอร์ เพื่ออ่านค่าทิศทางโดยเทียบเคียงกับสนามแม่เหล็กโลก เหมาะอย่างยิ่งกับการประยุกต์ใช้งานกับหุ่นยนต์หรืออุปกรณ์ควบคุมต่างๆ ที่ต้องมีการเคลื่อนที่ ให้ผลลัพธ์ 2 รูปแบบ คือ สัญญาณ PWM พัลส์วidthมอดูเลเตอร์ ดูผลจากการเปลี่ยนแปลงค่าความกว้างของพัลส์ ให้ผลเป็นตัวเลข 0-255 แทนค่าองศา 0-359.9 องศา และข้อมูลอนุกรมสื่อสารผ่านระบบบัส I2C ความละเอียดของข้อมูล 8 บิต

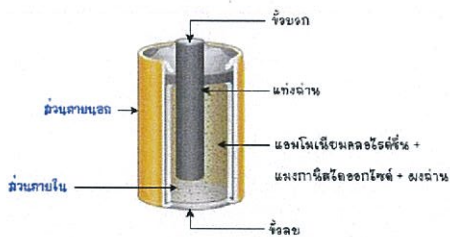


รูปที่ 2.15 โมดูลเข็มทิศ HMC5883L [18]

## 2.17 ถ่านไฟฉายขนาด 2A

ถ่านไฟฉาย เป็นเซลล์ไฟฟ้าเคมีที่ไม่ใช้สารละลายที่เป็นของเหลว จึงเรียกว่า เซลล์แห้ง (Dry Cell) ผู้ที่สร้างเซลล์ไฟฟ้าเคมีชนิดนี้คือ เลอ คังเซ George Leclanché ผู้ที่สร้างถ่านไฟฉาย ดังนั้น จึงอาจเรียกอีกชื่อหนึ่งว่า เซลล์เลอคังเซ ถ่านไฟฉาย 1 ก้อน หรือ 1 เซลล์ มีความต่างศักย์ประมาณ 1.5 โวลต์ เราไม่ควรแกะดูชิ้นส่วนของถ่านไฟฉาย เพราะสารเคมีที่ใช้ทำแบตเตอรี่นั้นอันตรายมาก

ถ่านไฟฉายประกอบด้วยแท่งแกรไฟต์อยู่ตรงกลางเป็นขั้วแคโทดมีอิเล็กโทรไลต์เป็นส่วนผสมของแมงกานีส (IV) ออกไซด์ ( $MnO_2$ ) แอมโมเนียมคลอไรด์ ( $NH_4Cl$ ) ซิงค์คลอไรด์ ( $ZnCl_2$ ) แป้งเปียก ผงคาร์บอน และน้ำคลุกเคล้าอยู่ด้วยกันในลักษณะเป็นอิเล็กโทรไลต์ชั้น สารทั้งหมดบรรจุอยู่ในกล่องสังกะสี อาจหุ้มด้วยกระดาษ แผ่นพลาสติกหรือโลหะสแตนเลส เพื่อป้องกันไม่ให้สารภายในรั่วออกมา ด้านบนของแท่งแกรไฟต์ประกอบด้วยโลหะสังกะสีอีกชั้นหนึ่ง ส่วนด้านล่างมีแผ่นสังกะสีทำหน้าที่เป็นขั้วแอโนด



รูปที่ 2.16 ส่วนประกอบของถ่านไฟฉาย [19]



รูปที่ 2.17 ถ่านไฟฉายขนาด 2A [20]

### 2.18 แบตเตอรี่ขนาด 12V

แบตเตอรี่ (Battery) คือ อุปกรณ์ที่ทำหน้าที่จัดเก็บพลังงานเพื่อไว้ใช้ต่อไป ถือเป็นอุปกรณ์ที่สามารถแปลงพลังงานเคมีให้เป็นไฟฟ้าได้โดยตรงด้วยการใช้เซลล์กัลวานิก (Galvanic cell) ที่ประกอบด้วยขั้วบวกและขั้วลบ พร้อมกับสารละลายอิเล็กโทรไลต์ (Electrolyte Solution) แบตเตอรี่อาจประกอบด้วยเซลล์กัลวานิกเพียง 1 เซลล์ หรือมากกว่า

แบตเตอรี่เป็นอุปกรณ์สำหรับจัดเก็บไฟฟ้าเท่านั้น ไม่ได้ผลิตไฟฟ้า สามารถประจุไฟฟ้าเข้าไปใหม่ (Recharge) ได้หลายครั้งและประสิทธิภาพจะไม่เต็ม 100% จะอยู่ที่ประมาณ 80% เพราะมีการสูญเสียพลังงานบางส่วนไปในรูปความร้อนและปฏิกิริยาเคมีจากการประจุหรือจ่ายประจุ แบตเตอรี่จัดเป็นอุปกรณ์ที่มีราคาแพง และเสียหายได้ง่าย หากดูแลรักษาไม่ดีเพียงพอหรือใช้งานผิดวิธี รวมถึง อายุการใช้งานของแบตเตอรี่แต่ละชนิดจะแตกต่างกันไป เนื่องด้วยวิธีการใช้, การบำรุงรักษา, การประจุ และอุณหภูมิ ฯลฯ



รูปที่ 2.18 แบตเตอรี่ขนาด 12V [21]

## 2.19 แบตเตอรี่สำรอง (Power Bank)

แบตเตอรี่สำรองหรือ Power Bank เป็นอุปกรณ์ที่จะเพิ่มพลังไฟให้กับเครื่องโทรศัพท์มือถือถึง Smart Phone, Tablet โดยผ่านแบตเตอรี่สำรองแบบพกพาที่มีขนาดกระทัดรัด น้ำหนักเบา แต่มีความจุมาก เช่น 5,000, 6,600, ไปจนถึง 30,000 MAH โดย Power Bank จะจ่ายไฟไปเข้าอุปกรณ์ เช่น Smart Phone เป็นการเติมไฟ หรือชาร์จไฟเข้าไป เช่นแบตเตอรี่ของ Smart Phone มีความจุ 2,100 MAH หากไฟใกล้หมด ก็เชื่อมต่อกับ Power Bank ใช้ต่อไปได้อีกนาน และเมื่อไฟใน Power Bank ใกล้จะหมด เราก็นำไปชาร์จไฟบ้าน 220 โวลต์ใหม่ได้



รูปที่ 2.19 แบตเตอรี่สำรอง [22]

## 2.20 สายไฟจัมป์เปอร์

สายไฟสำเร็จรูปเป็นอุปกรณ์ที่ใช้ส่งพลังงานไฟฟ้าจากที่หนึ่งไปยังอีกที่หนึ่งโดยกระแสไฟฟ้าจะเป็นตัวนำพลังงานไฟฟ้าผ่านไปตามสายไฟจนถึงอุปกรณ์อีกชนิดหนึ่ง สายไฟทำด้วยสารที่ยอมให้กระแสไฟฟ้าผ่านได้ เรียกว่าตัวนำไฟฟ้า และตัวนำไฟฟ้าที่ใช้ทำสายไฟเป็นโลหะที่ยอมให้กระแสไฟฟ้าผ่านได้ดี ลวดตัวนำแต่ละชนิดยอมให้กระแสไฟฟ้าผ่านได้ต่างกัน ตัวนำไฟฟ้าที่ยอมให้กระแสไฟฟ้าผ่านได้มากเรียกว่ามีความนำไฟฟ้ามาก ลวดตัวนำจะมีความต้านทานไฟฟ้าอยู่ด้วย โดยลวดตัวนำที่มีความต้านทานไฟฟ้ามากจะยอมให้กระแสไฟฟ้าผ่านได้น้อยเป็นสายไฟที่ใช้กับแรงดันไม่เกิน 750 V เป็นสายหุ้มฉนวน ทำด้วยทองแดงหรืออลูมิเนียม โดยทั่วไปเป็นสายทองแดงสายขนาดเล็กจะเป็นตัวนำเดี่ยววัสดุฉนวนที่ใช้กับสายแรงดันต่ำคือ Polyvinyl Chloride (PVC) และ Cross-Linked Polyethylene (XLPE)



รูปที่ 2.20 สายไฟจัมป์เปอร์สำเร็จรูป [23]

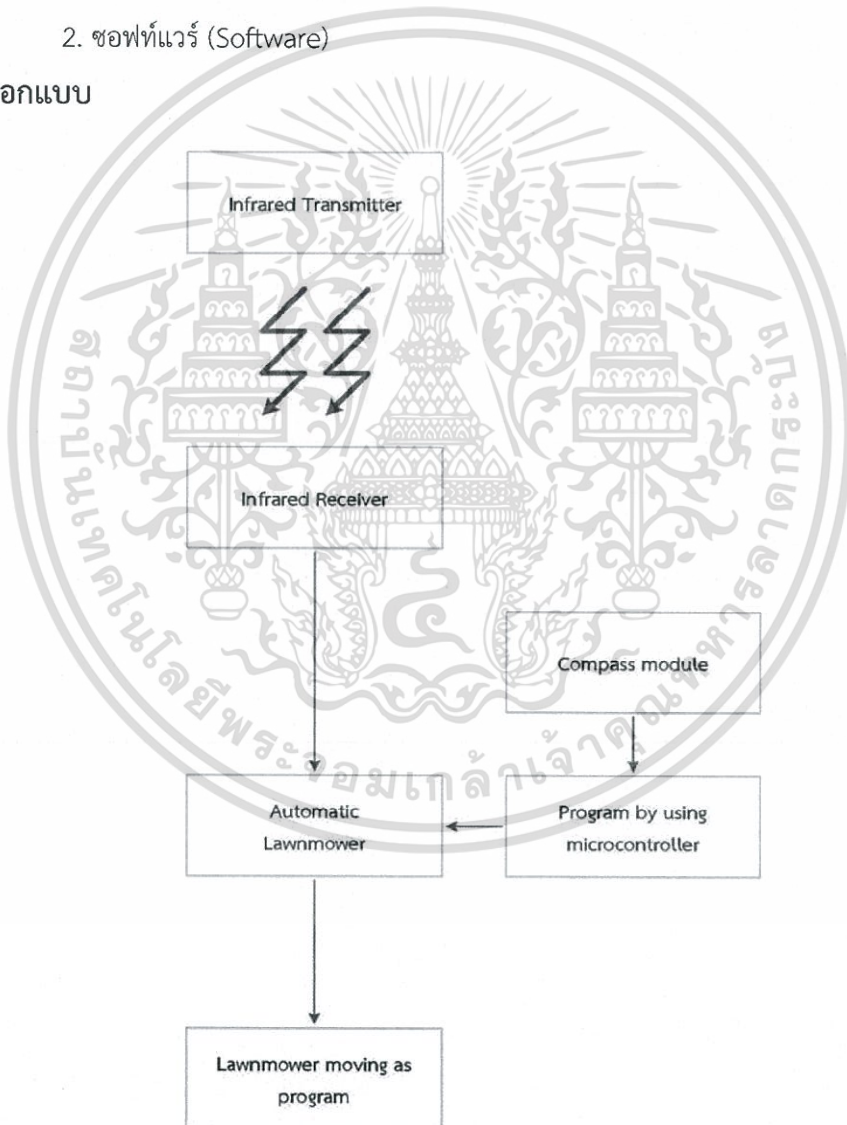
### บทที่ 3

#### การออกแบบและการจัดทำปฏิญานิพนธ์

การออกแบบและจัดทำปฏิญานิพนธ์ “แบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์” จะแบ่งการออกแบบเป็นสองส่วนหลัก คือ

1. ฮาร์ดแวร์ (Hardware)
2. ซอฟต์แวร์ (Software)

#### 3.1 การออกแบบ



รูปที่ 3.1 บล็อกไดอะแกรมรวมของรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์

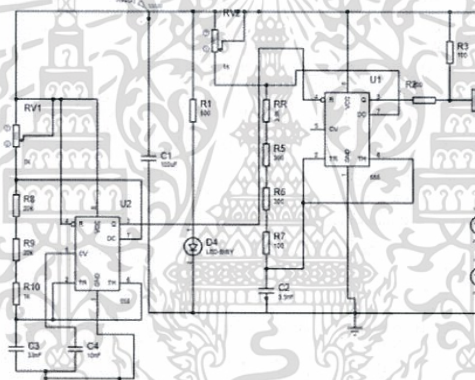
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.1 การออกแบบด้านฮาร์ดแวร์

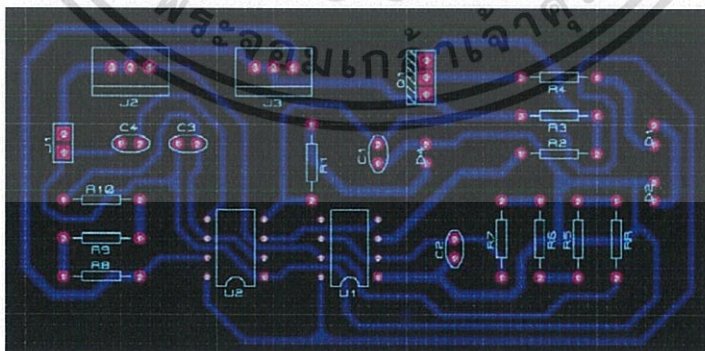
การออกแบบด้านฮาร์ดแวร์ของปฏิญาณนิพนธ์นี้แบ่งออกเป็น 2 ส่วนหลักๆ คือ ส่วนของวงจรตัวส่งอินฟราเรดที่ทำหน้าที่เป็นเซนเซอร์กำหนดขอบเขตพื้นที่ที่ใช้งานและส่วนขับเคลื่อนแบบจำลองรถตัดหญ้าอัตโนมัติ

#### 3.1.1.1 ส่วนเซนเซอร์อินฟราเรด

ส่วนของเซนเซอร์ที่เป็นวงจรตัวส่งสัญญาณอินฟราเรด ใช้ในการติดต่อในระยะทางใกล้ๆ การประยุกต์ใช้คลื่นอินฟราเรดจะเป็นการประยุกต์ใช้การสื่อสารแบบไร้สาย โดยใช้ IC555 ออกแบบให้วงจรมีความถี่ 38 kHz และได้เพิ่มเติมวงจรให้สามารถส่งสัญญาณได้ระยะทางไกลมากขึ้น โดยมีลิตเติ้ลสัญญาณ 500 Hz เข้าไปในวงจร ได้ทำการใช้โปรแกรม Proteus ในการออกแบบวงจรซึ่งสามารถสร้างสัญญาณ ความถี่ 38 kHz มัลติเพล็กซ์กับสัญญาณ ความถี่ 500 Hz



รูปที่ 3.2 การจำลองการทำงานวงจรตัวส่งสัญญาณอินฟราเรดบนโปรแกรม Proteus

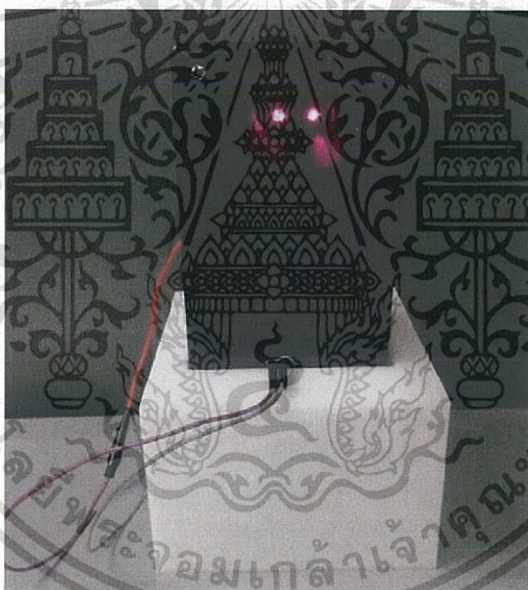


รูปที่ 3.3 ออกแบบลายวงจรตัวส่งสัญญาณอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 วงจรตัวส่งสัญญาณอินฟราเรด

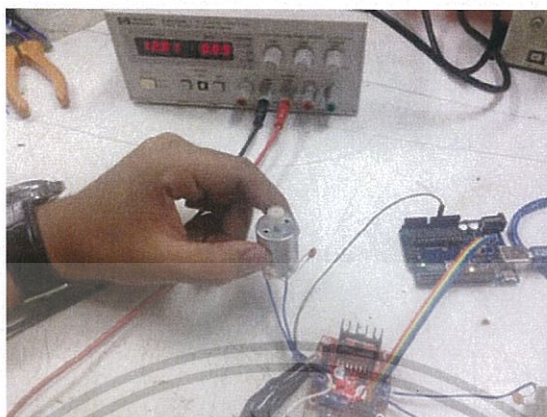


รูปที่ 3.5 ตัวส่งสัญญาณอินฟราเรดสำเร็จรูป

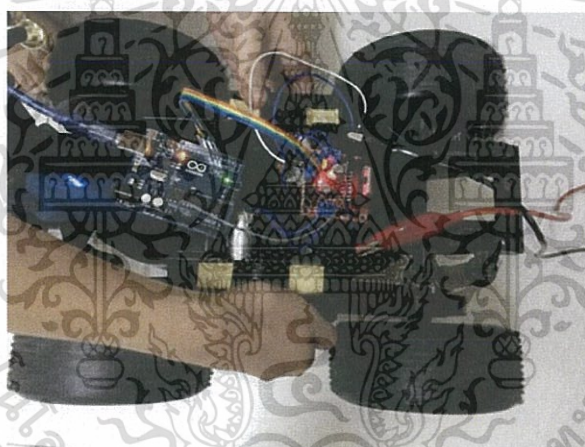
### 3.1.1.2 ส่วนขับเคลื่อนแบบจำลองรถตัดหญ้าอัตโนมัติ

ส่วนของการขับเคลื่อนแบบจำลองรถตัดหญ้านั้นมีกลไกในการทำงานคือการขับมอเตอร์ให้เกิดแรงเพื่อไปหมุนให้ล้อรถทำงาน โดยใช้ DC Motor Drive Controller Board Module L298N เป็นโมดูลที่ใช้ในการควบคุมความเร็วและทิศทางของมอเตอร์ซึ่งเชื่อมต่อกับบอร์ด Arduino ที่เขียนโปรแกรมไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



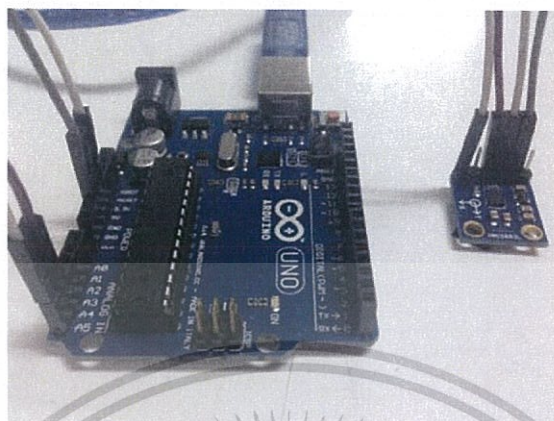
รูปที่ 3.6 ทดสอบการควบคุมมอเตอร์โดยใช้แรงดันไฟฟ้า 12V และใช้กระแสไฟฟ้า 90mA



รูปที่ 3.7 ทดสอบการเคลื่อนที่ของรถจำลอง

การขับเคลื่อนของรถตัดหญ้าอัตโนมัตินั้นสามารถเคลื่อนที่ไปตามทิศทางที่เรากำหนดไว้โดยประมวลผลการทำงานร่วมกับโมดูลเข็มทิศ HMC5883L ซึ่งจะทำให้รถตัดหญ้าสามารถเคลื่อนที่ไปได้อย่างแม่นยำและควบคุมองศาของทิศทางได้จากการเขียนโปรแกรมในชุดคำสั่งไมโครคอนโทรลเลอร์บนบอร์ด Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 การทำงานของบอร์ด arduino ประมวลผลการทำงานร่วมกับโมดูลเซ็นเซอร์

```

Serial.println("LOAD DEGREES");
Lowbuffer=EEPROM.read(eeAddress);
Highbuffer=EEPROM.read(eeAddress+1);
degree_Buffer_1=float(int(Lowbuffer)+int(Highbuffer));

delay(50);
Lowbuffer=EEPROM.read(eeAddress+2);
Highbuffer=EEPROM.read(eeAddress+3);
degree_Buffer_2=float(int(Lowbuffer)+int(Highbuffer));
delay(50);
Lowbuffer=EEPROM.read(eeAddress+4);
Highbuffer=EEPROM.read(eeAddress+5);
degree_Buffer_3=float(int(Lowbuffer)+int(Highbuffer));
delay(50);
Lowbuffer=EEPROM.read(eeAddress+6);
Highbuffer=EEPROM.read(eeAddress+7);
degree_Buffer_4=float(int(Lowbuffer)+int(Highbuffer));

```

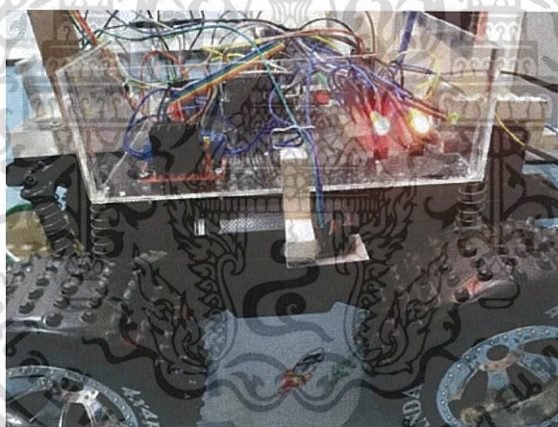
รูปที่ 3.9 คำสั่งการใช้งานการเก็บค่าองศาทิศทางของโมดูลเซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประมวลผลจากการทำงานของโมดูลเข็มทิศโดยโปรแกรมคำสั่งให้เข็มทิศไหลดค่าองศาทิศทางทั้งหมด 4 ทิศมาเก็บค่าไว้ การเก็บค่าองศาทิศทางนั้นให้กดปุ่มที่ไอโอดเปล่งแสงอันที่ 2 ค่าจึงจะน้อยกว่าจะมีแสงสีเหลืองปรากฏขึ้นโดยจะต้องทำการเก็บค่าทั้งหมด 4 ครั้งเพื่อเป็นองศาอ้างอิงในการเคลื่อนที่ของแบบจำลองรถตดหญาอัตโนมัติ

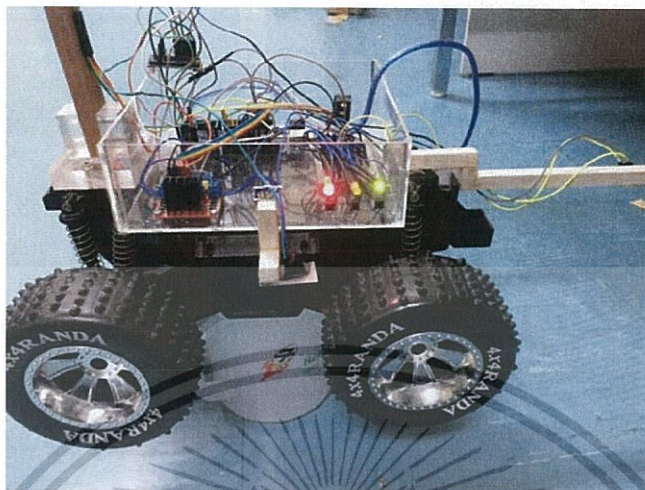
```
Serial.print("degree1 : "); Serial.print(degree_Buffer_1);
delay(100);
Serial.print(" degree2 : ");Serial.print(degree_Buffer_2);
delay(100);
Serial.print(" degree3 : ");Serial.print(degree_Buffer_3);
delay(100);
Serial.print(" degree4 : ");Serial.println(degree_Buffer_4);
delay(100);
degree=degree_Buffer_1;
Serial.begin(9600);
C=120;
```

รูปที่ 3.10 คำสั่งแสดงผลการเก็บค่าองศาทิศทางจากโมดูลเข็มทิศ

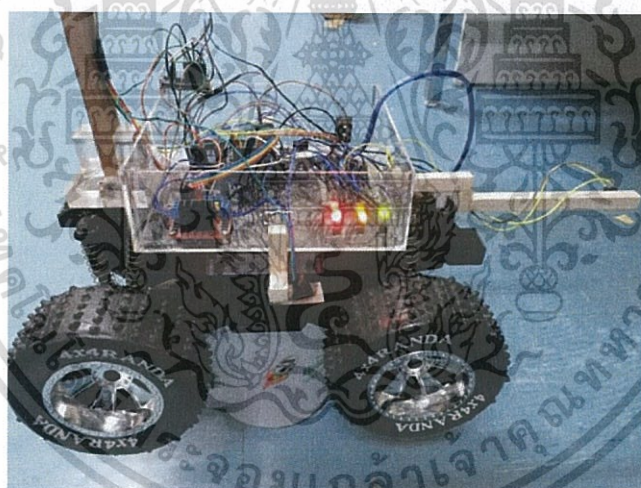


รูปที่ 3.11 ไอโอดปรากฏแสงสีเหลืองขึ้นแสดงการเก็บค่าองศาทิศทางเอาไว้

เมื่อเก็บค่าองศาทิศทางครบทั้ง 4 ทิศ สามารถเริ่มคำสั่งการทำงานของแบบจำลองรถตดหญาอัตโนมัติได้โดยกดปุ่มที่ไอโอดเปล่งแสงตัวที่ 1 และตัวที่ 3 พร้อมกันจะเริ่มการทำงานให้แบบจำลองรถตดหญาเคลื่อนที่ไปในพื้นที่ที่กำหนดไว้ เมื่อเคลื่อนที่ถึงขอบเขตที่กำหนดด้วยเซนเซอร์อินฟราเรดแล้ว ตัวรับสัญญาณอินฟราเรดจะทำงานและสั่งให้ไอโอดเปล่งแสงตัวที่ 2 เปล่งแสงสีเหลืองขึ้นมา โปรแกรมจะสั่งให้แบบจำลองรถตดหญาเคลื่อนที่ไปในทิศทางใหม่ที่อยู่ภายในพื้นที่ที่กำหนด



รูปที่ 3.12 ไดโอดสีแดงและสีเขียวปรากฏแสงขึ้นแสดงถึงการเริ่มการทำงาน



รูปที่ 3.13 ไดโอดเปล่งแสงทำงานพร้อมกันเมื่อตัวรับสัญญาณได้รับสัญญาณอินฟราเรด

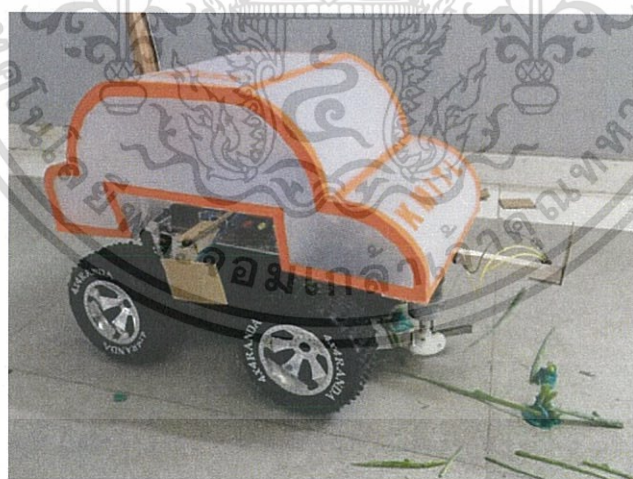
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.1.3 ส่วนแบบจำลองการตัดหญ้า

ส่วนแบบจำลองการตัดหญ้าได้ออกแบบการทำงานโดยใช้มอเตอร์ขนาด 12V ซึ่งต่อกับแหล่งจ่ายไฟโดยตรงจากแบตเตอรี่ขนาด 12V ที่ตัวมอเตอร์นั้นมีการติดใบมีดไว้ 2 ใบซึ่งเมื่อนำไปใช้งานแล้วนั้นสามารถตัดหญ้าได้จริง

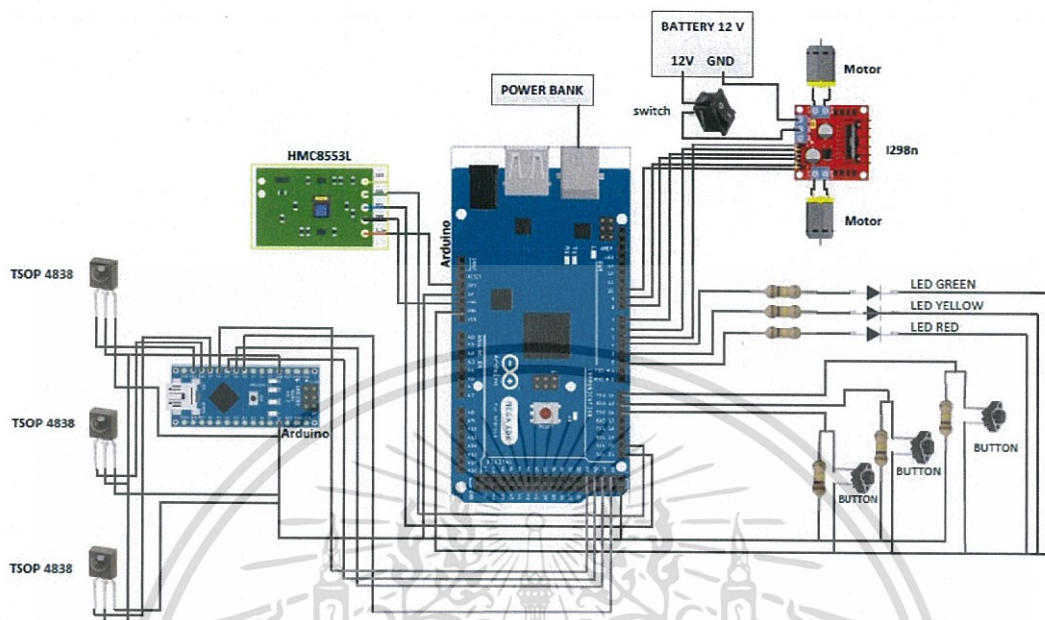


รูปที่ 3.14 ส่วนของมอเตอร์และใบมีดตัดหญ้า



รูปที่ 3.15 การทำงานของแบบจำลองรถตัดหญ้าโดยวิ่งตัดหญ้าในบริเวณที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 ภาพรวมโครงสร้างทั้งหมดในส่วนฮาร์ดแวร์

### 3.1.2 การออกแบบด้านซอฟต์แวร์

การออกแบบด้านซอฟต์แวร์ใช้โปรแกรม Arduino เป็นโปรแกรมที่ใช้เขียนภาษา C ลงในตัวไมโครคอนโทรลเลอร์ที่เรียกว่า Arduino ซึ่งการเขียนโปรแกรมนั้นจะเป็นการควบคุมการทำงานทั้งหมดของแบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติให้ทำงานเป็นไปตามที่กำหนดไว้

```

void loop()
{
  Read_Degree();
  Cal_Deg();
  if (Deg < -0.5) {D=-1; TURN();}
  if (Deg > 0.5) {D=1; TURN();}
  if ((Deg > -1.5) && (Deg < 1.5)) {D=0; TURN();}
  if (irrecv.decode(&results)) {
    Serial.println(results.value, HEX);
    irrecv.resume();
    irrecv.enableIRin();
  }
  if (results.value != 0)
  {
    if ((degree_Buffer_1 == degree)){
      degree = degree_Buffer_2;
      Read_Degree();
      IRrecv irrecv(IRpin1);
      decode_results results;
      irrecv.enableIRin();
      while(!((Cal_Deg()>5) || (Cal_Deg()<-5) )
      {
        if (Cal_Deg()>5)D=-2;
        if (Cal_Deg()<-5)D=2;
        TURN();
        delay(100);
        D=5; TURN();
        delay(100);
      }
    }
  }
}

```

รูปที่ 3.17 คำสั่งการกำหนดเงื่อนไขเมื่อมีการเคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if(digitalRead(button6)==LOW) STOP();
    }

    D=0; TURN();
    delay(700);
    D=5; TURN();
    delay(1000);
    degree = degree_Buffer_1;

Read_Degree();
while((Cal_Deg()>5) || (Cal_Deg()<-5) )
{
    Read_Degree();
    if(Cal_Deg()>5)D=-2;
    if(Cal_Deg()<5)D=2;
    TURN();
    delay(100);
    D=5; TURN();
    delay(100);
    Read_Degree();
    if(digitalRead(button6)==LOW) STOP();
}
}

    TimeC = mille();
    C=200; }
results.value =0;
C=200; irrecv.resume();
}
if(digitalRead(button6)==LOW) STOP();

```

รูปที่ 3.18 คำสั่งหยุดการทำงานเมื่อได้รับสัญญาณอินฟราเรด

ส่วนของคำสั่งที่ควบคุมการทำงานของแบบจำลองรถตัดหญ้าที่ให้เคลื่อนที่ตามทิศทางที่เรา กำหนดซึ่งการทำงานนั้นชุดคำสั่งไมโครคอนโทรลเลอร์จะประมวลผลร่วมกับโมดูลเข็มทิศและสามารถ โปรแกรมคำสั่งภาษา C ได้ดังนี้

```

float Read_Degree(void)
{
    i=0;
    MagnetometerRaw raw = compass.ReadRawAxis();
    float heading = atan2(raw.YAxis, raw.XAxis);
    if(heading < 0)
    heading += 2*PI;
    float declinationAngle = 0.26459;
    heading += declinationAngle;
    headingDegree_S = (heading * 180/M_PI);
    while(i<C){
    MagnetometerRaw raw = compass.ReadRawAxis();
    float heading = atan2(raw.YAxis, raw.XAxis);
    float declinationAngle = 0.26459;

    heading += declinationAngle;
    if(heading < 0)
    heading += 2*PI;
    headingDegrees = (heading * 180/M_PI);
    headingDegree_S=headingDegree_S+headingDegrees;
    i++;
    }
    degree_CAL=(headingDegree_S/(C+1));
    Serial.print(" HeadingDegree : ");
    Serial.println(degree_CAL);
    HEADING_BUFFER = degree_CAL;
}

```

รูปที่ 3.19 คำสั่งการเก็บค่าองศาจากโมดูลเข็มทิศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void TURN(void) {
  Serial.print(" Direction : ");
  if(D==1)
  {
    Serial.print("LEFT");
    analogWrite(enA, 255);
    analogWrite(enB, 0); //AAA
    digitalWrite(in1, HIGH);
    digitalWrite(in2, LOW);
    digitalWrite(in3, HIGH);
    digitalWrite(in4, LOW);
  }
}

```

รูปที่ 3.20 คำสั่งควบคุมความเร็วและทิศทางของมอเตอร์

```

if(D==2)
{ Serial.print(" ");
  Serial.println("OVER TURN LEFT");
  analogWrite(enA, 255); //AAA
  analogWrite(enB, 255);
  digitalWrite(in1, LOW);
  digitalWrite(in2, HIGH);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
}
if(D==2)
{
  Serial.print(" ");
  Serial.println("OVER TURN RIGHT");
  analogWrite(enA, 255);
  analogWrite(enB, 255); //AAA
  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
}

```

รูปที่ 3.21 คำสั่งควบคุมการเลี้ยวซ้ายและเลี้ยวขวา

```

void setting_degree(void)
{
  C=1000;
  digitalWrite(LEDGREEN,LOW);
  digitalWrite(LEDYELLOW,HIGH);delay(500);digitalWrite(LEDYELLOW,LOW);delay(500);
  digitalWrite(LEDYELLOW,HIGH);delay(500); digitalWrite(LEDYELLOW,LOW);delay(500);digitalWrite(LEDYELLOW,HIGH);
  while (digitalRead(button5)==HIGH){
    if(digitalRead(LEDYELLOW)==HIGH)
    digitalWrite(LEDYELLOW,LOW); else digitalWrite(LEDYELLOW,LOW);
    Read_Degree();
  }
  digitalWrite(LEDYELLOW,LOW);delay(300);
  digitalWrite(LEDYELLOW,HIGH);delay(300);

  degree_Buffer_1=HEADING_BUFFER;
  Serial.print(" Degree1 :"); Serial.println(degree_Buffer_1);
  delay(2000);
  while (digitalRead(button5)==HIGH){
    if(digitalRead(LEDYELLOW)==HIGH)
    digitalWrite(LEDYELLOW,LOW); else digitalWrite(LEDYELLOW,LOW);
    Read_Degree();
  }
  digitalWrite(LEDYELLOW,LOW);delay(300);
  digitalWrite(LEDYELLOW,HIGH);delay(300);
  digitalWrite(LEDYELLOW,LOW);delay(300);
  digitalWrite(LEDYELLOW,HIGH);delay(500);
}

```

รูปที่ 3.22 คำสั่งการแสดงผล LED เพื่อควบคุมการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void STOP(void)
{
digitalWrite(LEDGREEN,LOW);
Serial.println("STOP");
D=5; TURN();
delay(2000);
while(digitalRead(button6)==HIGH);
delay(2000);
digitalWrite(LEDGREEN,HIGH);
return ;
}
void loaddegrees(void)
{
}

```

รูปที่ 3.23 คำสั่งหยุดการทำงานเมื่อเคลื่อนที่ครบตามพื้นที่ที่กำหนดไว้

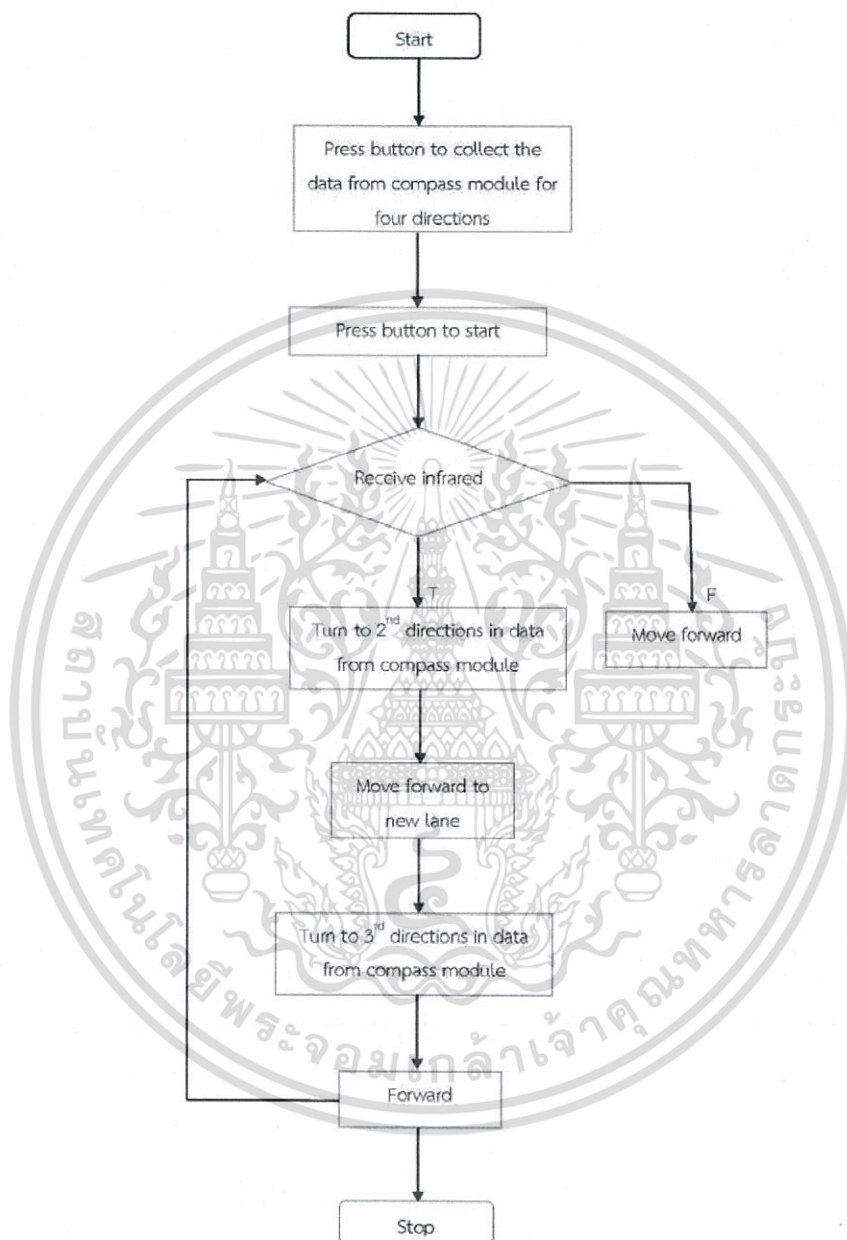
การทำงานในส่วนของ Arduino Nano 3.0 เป็นการทำงานในส่วนของการรับสัญญาณอินฟราเรด เพื่อนำมาประมวลผลและส่งคำสั่งไปยัง Arduino Mega 2560 เพื่อนำไปควบคุมการเคลื่อนที่ของแบบจำลองรถตัดหญ้า

```

void loop()
{
Serial.println("STATUS=0");
STATUS=0;
delay(200);
IR_BUFFER =IRpin1;
Set_UP();
if (irrecv.decode(&results)) {
Serial.println(results.value, HEX);
Serial.println("1");
irrecv.resume();
STATUS1=1;
}
check_TSOP();
delay(200);
IR_BUFFER =IRpin2;
Set_UP();
if (irrecv.decode(&results)) {
Serial.println(results.value, HEX);
Serial.println("2");
irrecv.resume();
STATUS1=1;
}
}

```

รูปที่ 3.24 การถอดค่าสัญญาณอินฟราเรดเพื่อประมวลผล



รูปที่ 3.25 Flow chart การทำงานของแบบจำลองรถตัดหญ้าอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

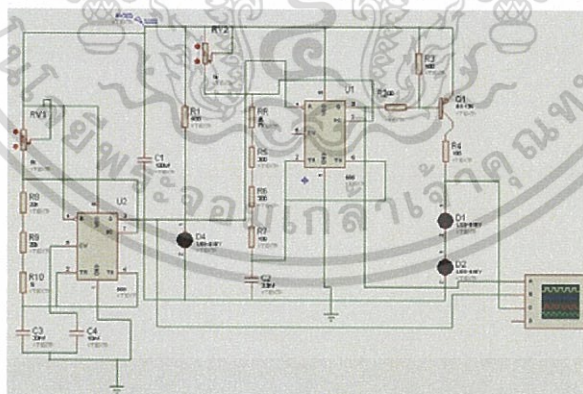
### 3.2 เครื่องมือที่ใช้ในการทดลอง

1. คอมพิวเตอร์
2. ออสซิลโลสโคป (Oscilloscope)
3. แหล่งจ่ายแรงดันไฟฟ้า (Power Supply)
4. มัลติมิเตอร์
5. บอร์ด Arduino Mega 2560
6. บอร์ด Arduino Nano 3.0
7. โมดูลเข็มทิศ HMC5883L
8. โมดูล L298N ควบคุมมอเตอร์
9. โพรโทบอร์ด
10. สายสำหรับต่ออุปกรณ์ต่างๆ

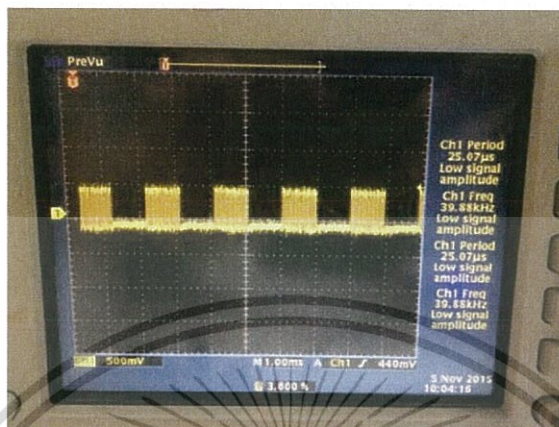
### 3.3 การจัดเก็บผลการทดลอง

#### 3.3.1 ส่วนเซนเซอร์อินฟราเรด

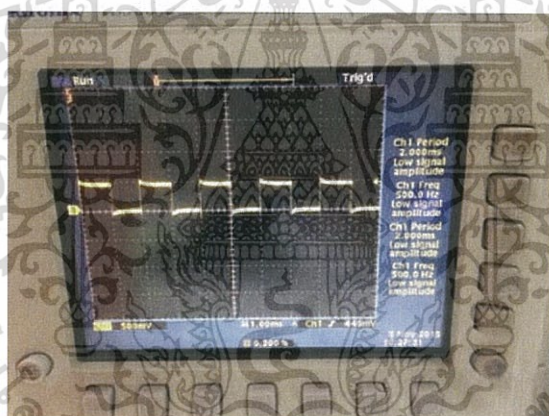
การจัดเก็บผลการทดลองในส่วนแรกคือ เซนเซอร์อินฟราเรดที่ออกแบบวงจรโดยใช้ IC555 ให่วงจรมีความถี่ 38 kHz และมีลติเพิลิกสัญญาณ 500 Hz เข้าไปในวงจรเพื่อให้สามารถส่งสัญญาณอินฟราเรดได้ระยะทางไกลขึ้น



รูปที่ 3.26 ทดสอบการทำงานของวงจรตัวส่งสัญญาณอินฟราเรดบนโปรแกรม Proteus



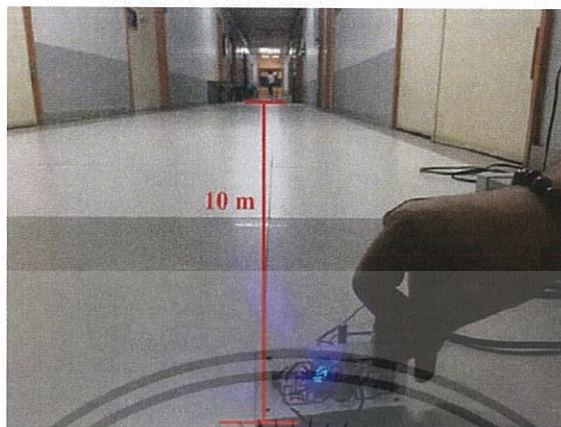
รูปที่ 3.27 สัญญาณด้านวงจรตัวส่งสัญญาณอินฟราเรดที่ความถี่ 39.88 kHz



รูปที่ 3.28 สัญญาณดิจิทัล 500Hz ที่มีลติเพล็กซ์เข้าไปในวงจรตัวส่งสัญญาณอินฟราเรด

การวัดระยะทางระหว่างตัวรับกับตัวส่งสัญญาณอินฟราเรด ซึ่งระยะทางที่วัดได้สัญญาณสามารถส่งได้ไกล 10-15 เมตร แต่ระยะทางที่นำมาใช้งานจริงต้องการเพียง 5-10 เมตร เพราะฉะนั้นวงจรตัวส่งสัญญาณอินฟราเรดที่มีการมัลติเพล็กซ์สัญญาณดิจิทัลความถี่ 500 Hz เข้าไปด้วยสามารถส่งสัญญาณได้ระยะทางไกลขึ้นและเสถียรมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



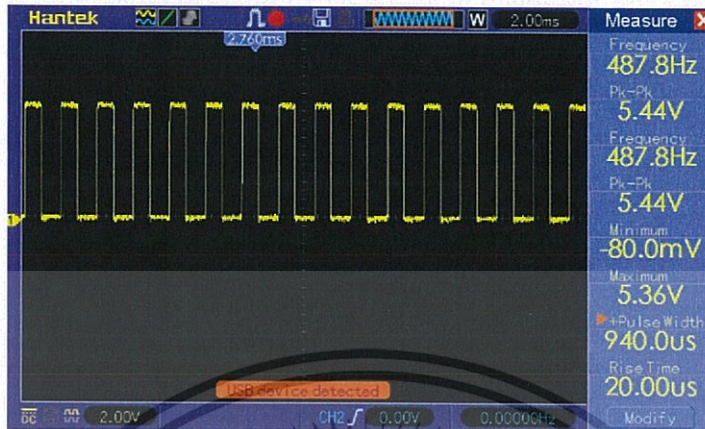
รูปที่ 3.29 วัดระยะการส่งสัญญาณอินฟราเรดได้ 10 เมตร

### 3.3.2 ส่วนของตัวรับสัญญาณอินฟราเรด

การเก็บผลการทดลองจากส่วนของตัวรับสัญญาณอินฟราเรด ซึ่งใช้ตัวรับสัญญาณ TSOP4838 โดยตัวรับสัญญาณนี้จะติดอยู่กับแบบจำลองรถตัดหญ้าอัตโนมัติเพื่อคอยตรวจรับสัญญาณอินฟราเรดไม่ให้เคลื่อนที่ออกนอกพื้นที่ที่กำหนด โดยจะรับสัญญาณดิจิทัลที่มีความถี่ 500 Hz และสามารถแสดงผลสัญญาณในขณะที่มีสิ่งกีดขวางและไม่มีสิ่งกีดขวางมาบังได้ดังรูปที่ 3.29 และรูปที่ 3.30



รูปที่ 3.30 วงจรตัวรับสัญญาณอินฟราเรด



รูปที่ 3.31 สัญญาณดิจิทัล 500 Hz ที่ได้จากวงจรตัวส่งอินฟราเรดโดยไม่มีสิ่งกีดขวางมาบังระหว่างตัวรับกับตัวส่ง

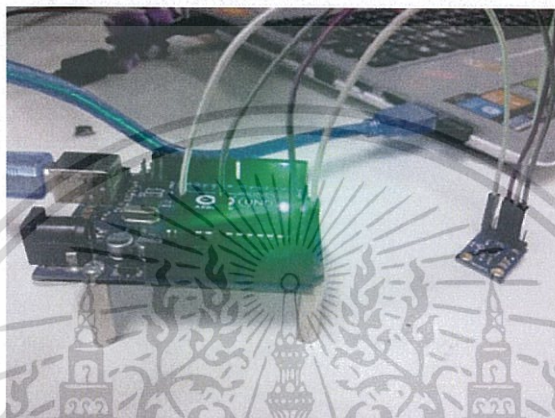


รูปที่ 3.32 สัญญาณที่ได้รับจากวงจรตัวส่งอินฟราเรดเมื่อมีสิ่งกีดขวางระหว่างตัวรับกับตัวส่ง

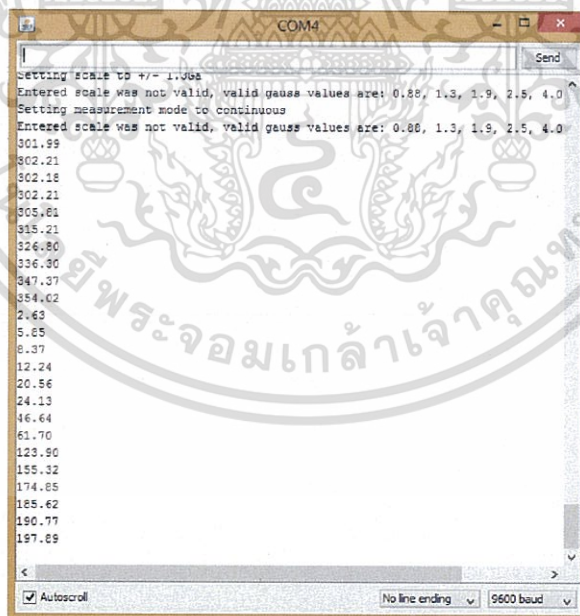
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.3 ส่วนของการควบคุมทิศทาง

การเคลื่อนที่ของแบบจำลองรถดัดหญ้าอัตโนมัติอาศัยการควบคุมทิศทางจากโมดูลควบคุมมอเตอร์ L298N และโมดูลเข็มทิศที่เป็นฟังก์ชันเสริมที่ทำงานประมวลผลทิศทางให้เกิดความแม่นยำมากขึ้น



รูปที่ 3.33 ออฟโหลดคำสั่งควบคุมการทำงานของโมดูลเข็มทิศ



รูปที่ 3.34 ผลการทำงานเบื้องต้นของโมดูลเข็มทิศที่เปลี่ยนองศาทิศทางเมื่อลงหมุนเข็มทิศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการทดลอง

จากการออกแบบทั้งหมดในบทที่ 3 เมื่อทดสอบการทำงานของแบบจำลองรถตัดหญ้า โปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์สามารถแสดงผลได้ 2 ส่วน คือ ส่วนของซอฟต์แวร์และส่วนของฮาร์ดแวร์

#### 4.1 ผลการทดลองส่วนของซอฟต์แวร์

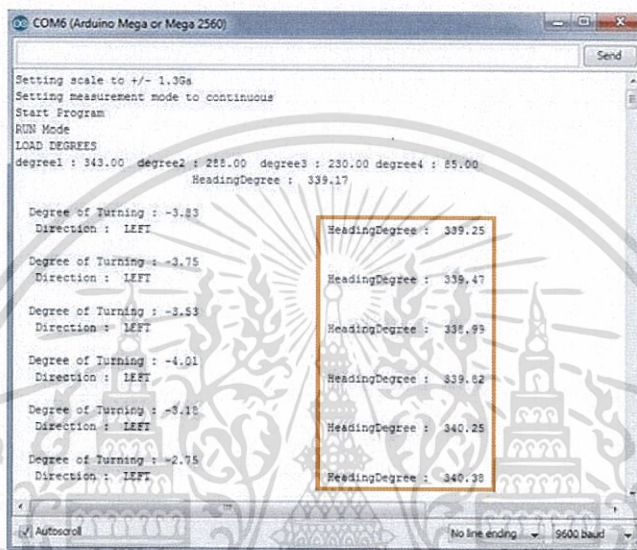
ผลการทดลองในส่วนของซอฟต์แวร์เป็นตัวแสดงผลการทำงานของโปรแกรมว่าเป็นไปตามหลักการที่ได้เขียนโปรแกรมลงในไมโครคอนโทรลเลอร์ ซึ่งผลลัพธ์การทำงานจะแสดงผลบนหน้าจอมอนิเตอร์ของโปรแกรม Arduino

เริ่มจากการตั้งค่าโหมดการทำงานของแบบจำลองรถตัดหญ้าโดยเก็บค่าองศาทิศทางทั้งหมด 4 ทิศเพื่อเป็นข้อมูลอ้างอิงในการเคลื่อนที่ซึ่งการเก็บค่าในครั้งนี้ได้มุมองศาทิศทางคือ องศาที่ 1 : 343.00, องศาที่ 2 : 288.00, องศาที่ 3 : 230.00 และองศาที่ 4 : 85.00

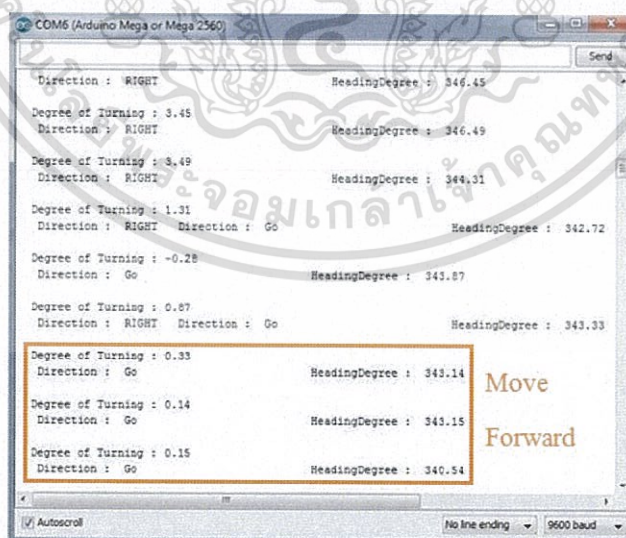
```
Setting scale to +/- 1.30g
Setting measurement mode to continuous
Start Program
Setup Mode
HeadingDegree : 343.96
HeadingDegree : 343.76
HeadingDegree : 343.95
HeadingDegree : 343.99
Degree1 : 343.00
HeadingDegree : 343.63
HeadingDegree : 343.68
HeadingDegree : 336.37
HeadingDegree : 325.07
HeadingDegree : 307.38
HeadingDegree : 294.10
HeadingDegree : 288.41
HeadingDegree : 288.82
HeadingDegree : 288.82
Degree2 : 288.00
HeadingDegree : 287.60
HeadingDegree : 281.91
HeadingDegree : 262.88
HeadingDegree : 235.88
HeadingDegree : 230.36
HeadingDegree : 230.47
HeadingDegree : 230.34
Degree3 : 230.00
HeadingDegree : 230.17
HeadingDegree : 217.09
HeadingDegree : 164.95
HeadingDegree : 105.62
HeadingDegree : 86.19
HeadingDegree : 85.02
Degree4 : 85.00
EEPROM VALUE : 343.00 EEPROM VALUE : 288.00 EEPROM VALUE : 230.00 EEPROM VALUE : 85.00
```

รูปที่ 4.1 ผลลัพธ์แสดงการเก็บค่าองศาทิศทางทั้งหมด 4 ทิศ

จากการเก็บค่าองศาทิศทางแล้วนั้นเพื่อเป็นการกำหนดให้แบบจำลองรถตัดหญ้าเคลื่อนที่ภายในพื้นที่ที่กำหนด โดยลักษณะการเคลื่อนที่จะเริ่มจากมุมสนามด้านซ้ายวิ่งไปด้านหน้าจนถึงขอบเขตที่มีสัญญาณอินฟราเรด แล้วแบบจำลองรถตัดหญ้าจะเลี้ยวขวาไปเป็นมุม 90 องศาเพื่อเปลี่ยนเส้นทางในการเคลื่อนที่

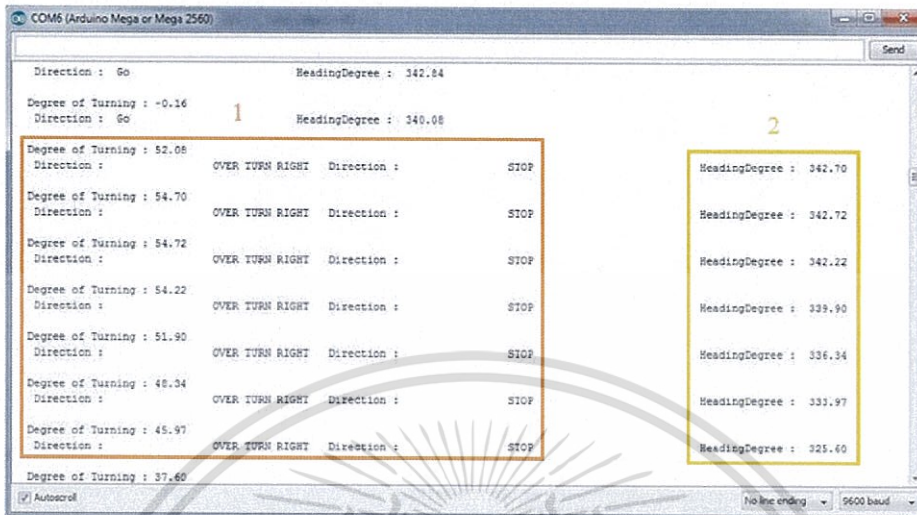


รูปที่ 4.2 ค่าองศาทิศทางที่โมดูลเข็มทิศอ่านได้เมื่อเริ่มการทำงาน



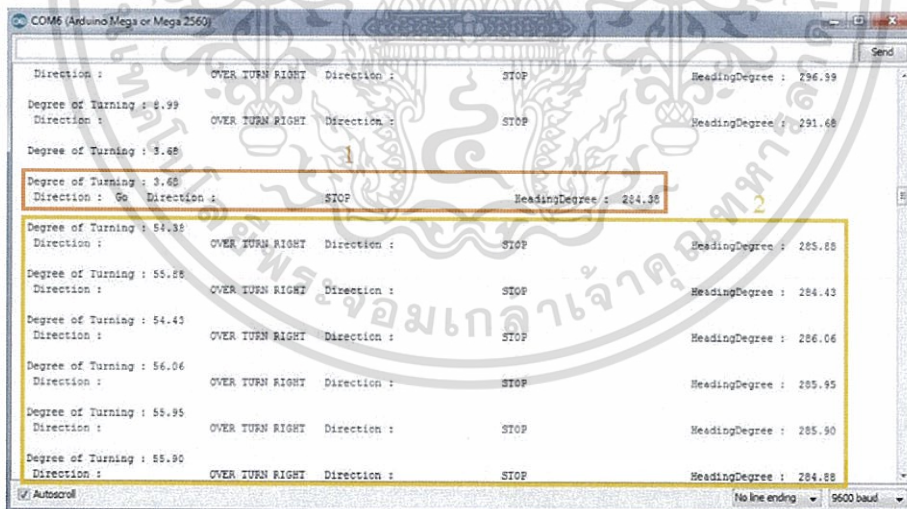
รูปที่ 4.3 ผลลัพธ์แสดงการเคลื่อนที่ไปข้างหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 ผลลัพธ์แสดงการหยุดเมื่อตรวจรับเจอสัญญาณอินฟราเรด

จากรูปที่ 4.4 ส่วนที่ 1 เป็นการแสดงผลเมื่อแบบจำลองรถตัดหญ้าได้รับสัญญาณอินฟราเรดที่กำหนดขอบเขตพื้นที่ไว้แล้วจะทำการค่อยๆ เลี้ยวขวาเพื่อเปลี่ยนทิศทาง และส่วนที่ 2 แสดงมุมมองค่าที่โมดูลเข็มทิศอ่านได้ในขณะที่กำลังเลี้ยวขวา



รูปที่ 4.5 ผลลัพธ์แสดงการเคลื่อนที่เมื่อเลี้ยวไปในทิศทางใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.5 ส่วนที่ 1 เป็นการแสดงผลเมื่อเคลื่อนที่ไปข้างหน้าเล็กน้อยเพื่อเปลี่ยนเส้นทางใหม่ และส่วนที่ 2 แสดงการเลี้ยวขวาของแบบจำลองรถตัดหญ้าอีกครั้งเพื่อที่จะเคลื่อนที่ไปข้างหน้าในเส้นทางใหม่

```

COM5 (Arduino Mega or Mega 2560)
Send

Degree of Turning : 10.17
Direction : OVER TURN RIGHT Direction : STOP HeadingDegree : 238.81

Degree of Turning : 8.81
Direction : OVER TURN RIGHT Direction : STOP HeadingDegree : 233.92

Degree of Turning : 3.92
Degree of Turning : 3.92
HeadingDegree : 233.54

Degree of Turning : 3.54
Direction : RIGHT HeadingDegree : 231.07

Degree of Turning : 1.07
Direction : RIGHT Direction : Go HeadingDegree : 229.10

Degree of Turning : -0.99
Direction : LEFT Direction : Go HeadingDegree : 230.16

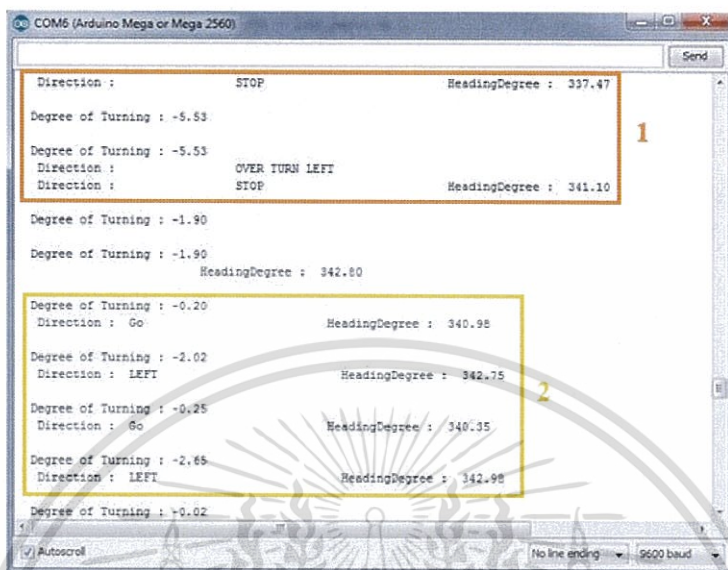
Degree of Turning : 0.16
Direction : Go HeadingDegree : 230.42

Degree of Turning : 0.48
Direction : Go HeadingDegree : 228.30

Autoscroll No line ending 9600 baud
  
```

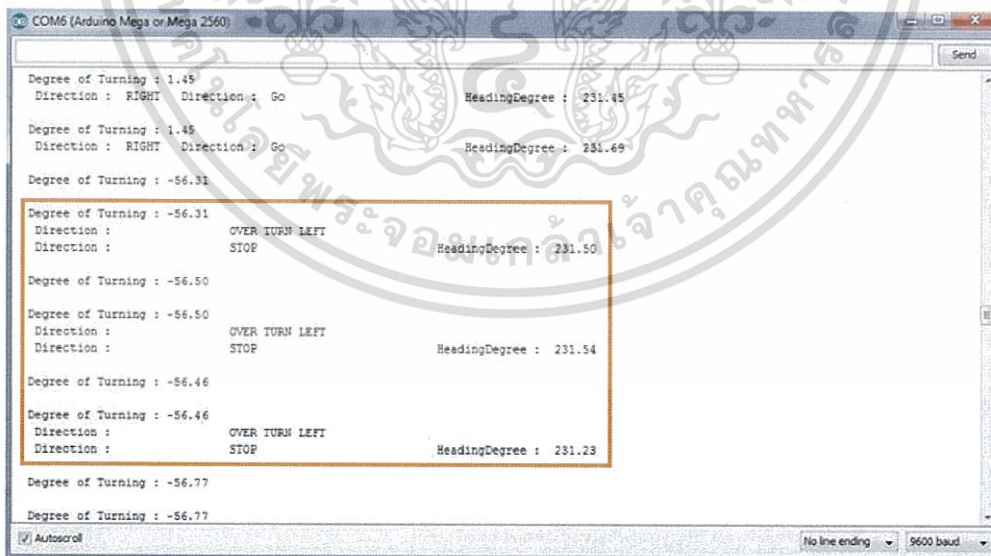
รูปที่ 4.6 ผลลัพธ์แสดงการเคลื่อนที่ไปข้างหน้าในเส้นทางใหม่เป็นมุมองศา 233.54

เมื่อแบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติเคลื่อนที่ทำงานในเส้นทางใหม่ โปรแกรมการทำงานที่เขียนไว้ในไมโครคอนโทรลเลอร์จะมีคำสั่งให้เคลื่อนที่ไปข้างหน้าอีกครั้งซึ่งจะเป็นทิศทางตรงกันข้ามกับการเคลื่อนที่ในครั้งแรกและเมื่อวิ่งไปจนถึงขอบเขตที่กำหนดไว้ตัวรับสัญญาณอินฟราเรด TSOP4838 ตรวจรับสัญญาณได้จะทำให้แบบจำลองรถตัดหญ้าหยุดและเคลื่อนที่เลี้ยวไปทางซ้ายเพื่อเคลื่อนที่ไปในเส้นทางใหม่และทำงานต่อไปจนครบบริเวณพื้นที่ของสนามหญ้าที่กำหนดไว้



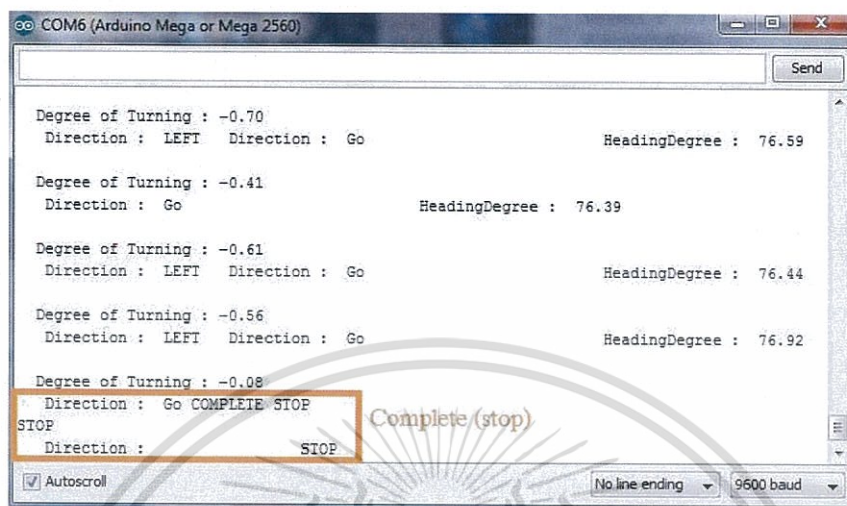
รูปที่ 4.7 ผลลัพธ์แสดงการเคลื่อนที่เมื่อได้รับสัญญาณอินฟราเรด

จากรูปที่ 4.7 ส่วนที่ 1 เป็นการแสดงผลลัพธ์เมื่อแบบจำลองรถตัดหญ้าเคลื่อนที่ถึงขอบเขตที่มีสัญญาณอินฟราเรดจึงหยุดและเลี้ยวซ้ายเพื่อเปลี่ยนองศาทิศทาง ส่วนที่ 2 เป็นการเคลื่อนที่ไปข้างหน้าอีกเพียงเล็กน้อยเพื่อเคลื่อนที่เขยิบไปทางซ้าย



รูปที่ 4.8 ผลลัพธ์แสดงการเคลื่อนที่เมื่อแบบจำลองรถตัดหญ้าเลี้ยวซ้าย

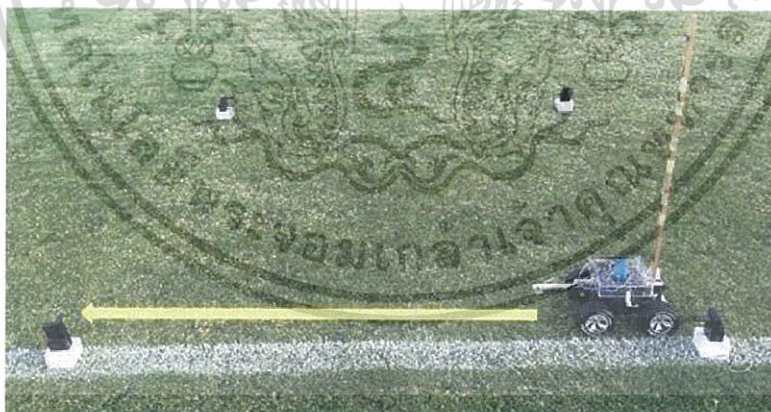
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 ผลลัพธ์แสดงการเคลื่อนที่เมื่อแบบจำลองรถตัดหญ้าทำงานเสร็จสมบูรณ์

## 4.2 ผลการทดลองส่วนของฮาร์ดแวร์

การทดสอบการทำงานของโครงงานนี้ในส่วนของฮาร์ดแวร์ได้นำแบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติมาวิ่งทดสอบบนสนามหญ้าขนาด 3x3 เมตร โดยมีความเร็วมอเตอร์ในการหมุนต่อรอบสูงสุด ที่ค่า PWM เท่ากับ 255

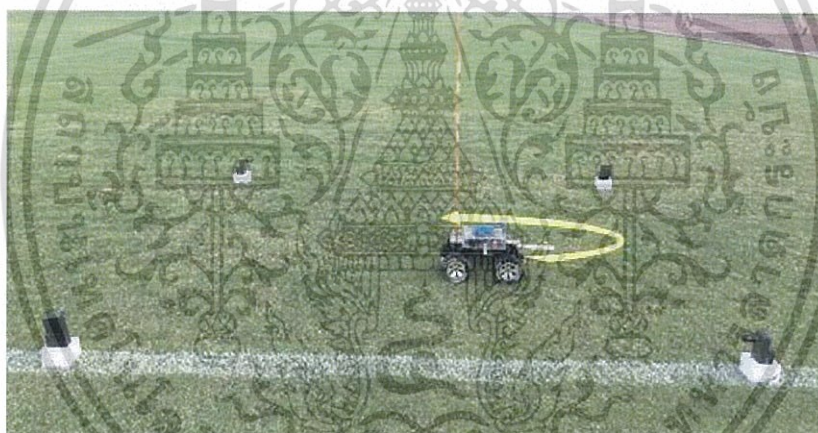


รูปที่ 4.10 เริ่มการเคลื่อนที่ของแบบจำลองรถตัดหญ้าอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 การเคลื่อนที่กลับไปอีกด้านหนึ่ง

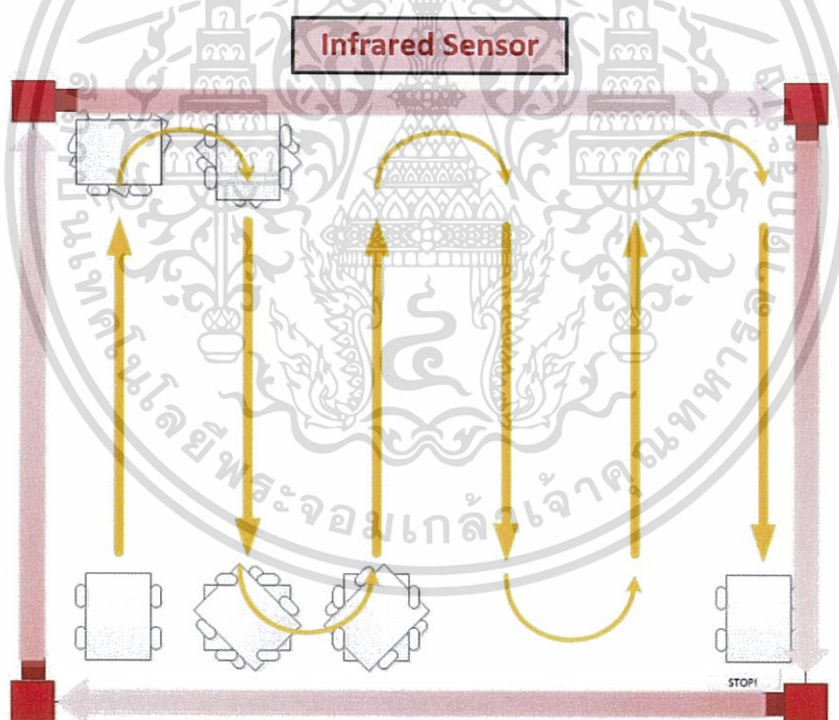


รูปที่ 4.12 แบบจำลองรถตัดหญ้าอัตโนมัติเลี้ยวซ้ายเพื่อเคลื่อนที่ไปยังพื้นที่ถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 การทำงานของแบบจำลองรถตัดหญ้าอัตโนมัติเสร็จสิ้นสมบูรณ์



รูปที่ 4.14 ภาพจำลองแสดงการเคลื่อนที่ทั้งหมดของแบบจำลองรถตัดหญ้าอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผลและข้อเสนอแนะ

#### 5.1 สรุปผล

ปริญญาโทขั้นต้นนี้นำเสนอแบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์ โดยเริ่มจากการทำเซนเซอร์ที่เป็นวงจรตัวส่งสัญญาณอินฟราเรด ใช้ในการติดต่อในระยะทางไกลๆ โดยใช้ IC555 ออกแบบให้วงจรมีความถี่ 38 kHz มัลติเพล็กซ์สัญญาณ 500 Hz เพื่อให้สามารถส่งสัญญาณได้ระยะทางไกลมากขึ้น ในส่วนของการขับเคลื่อนแบบจำลองรถตัดหญ้า มีกลไกในการทำงานคือ การขับมอเตอร์ให้เกิดแรงเพื่อหมุนให้ล้อรถทำงาน โดยใช้ Motor Drive Controller L298N เป็นโมดูลที่ใช้ในการควบคุมความเร็วและทิศทางของมอเตอร์ซึ่งเชื่อมต่อกับบอร์ด Arduino Mega 2560 ประมวลผลการทำงานร่วมกับโมดูลเข็มทิศ HMC5883L ซึ่งจะทำให้แบบจำลองรถตัดหญ้าสามารถเคลื่อนที่ไปได้อย่างแม่นยำ ส่วนของการตัดหญ้าได้ออกแบบการทำงานโดยใช้มอเตอร์ขนาด 12V ซึ่งต่อกับแบตเตอรี่ขนาด 12V ที่ตัวมอเตอร์นั้นมีการติดใบมีดไว้ 2 ใบซึ่งสามารถนำไปใช้ตัดหญ้าได้จริง

การออกแบบด้านซอฟต์แวร์ใช้โปรแกรม Arduino เป็นโปรแกรมที่ใช้เขียนภาษา C โดยใช้ Arduino Mega 2560 ให้โปรแกรมควบคุมการเคลื่อนที่ของแบบจำลองส่วนของ Arduino Nano 3.0 เป็นการทำงานในส่วนของการรับสัญญาณอินฟราเรดโดยใช้ TSOP4838 เป็นตัวรับสัญญาณอินฟราเรด จำนวน 3 ตัว ได้แก่ ด้านหน้า ด้านขวา และด้านซ้าย เพื่อนำมาประมวลผลและส่งคำสั่งไปยัง Arduino Mega 2560 เพื่อควบคุมการเคลื่อนที่ของแบบจำลองรถตัดหญ้า

ผลการทำงานของระบบพบว่า การขับเคลื่อนของแบบจำลองรถตัดหญ้าอัตโนมัตินั้นสามารถเคลื่อนที่ไปตามทิศทางที่เรากำหนดโดยสามารถเคลื่อนที่ตามค่าองศาทิศทางที่เก็บค่าไว้ และเคลื่อนที่ภายในบริเวณที่กำหนดโดยมีเซนเซอร์อินฟราเรดเป็นตัวกำหนดขอบเขต รวมไปถึงแบบจำลองรถตัดหญ้าอัตโนมัติสามารถตัดหญ้าใช้งานได้จริงตามวัตถุประสงค์ที่ได้กล่าวไว้ข้างต้น

#### 5.2 ข้อเสนอแนะ

แบบจำลองรถตัดหญ้าอัตโนมัติที่จัดทำขึ้นในโครงงานนี้ สามารถเพิ่มเติมในส่วนซอฟต์แวร์ให้มีการทำงานได้หลากหลายฟังก์ชันมากขึ้น เช่น เคลื่อนที่ในทิศทางใหม่ๆ ปรับความเร็วในการเคลื่อนที่ รวมถึงการออกแบบตัวรับสัญญาณอินฟราเรด สามารถออกแบบให้มีความทนทานมากขึ้น เช่น บรรจุให้ใช้งานในกล่องที่ทนต่อทุกสภาพอากาศ แต่เนื่องจากระบบรถเป็นแบบจำลองเครื่องตัดหญ้า จึงไม่ได้ใช้อุปกรณ์ที่มีราคาสูงและมีประสิทธิภาพมากนัก เมื่อส่วนซอฟต์แวร์ทำงานร่วมกับฮาร์ดแวร์ จึงยังคงมีความคลาดเคลื่อนได้

## บรรณานุกรม

- [1] Webscaping. “Electromagnetic Spectrum.”  
<http://www.cyberphysics.co.uk/topics/light/emspect.htm>
- [2] Thaimicrotron. “วงจรการส่งสัญญาณอินฟราเรด.”  
<http://www.thaimicrotron.com/CCS-628/EXAM/TVRemote.htm>
- [3] Siritwimon Sunthon. “บอร์ด Arduino Mega 2560.”  
<http://mbeddedweekly.blogspot.com/2014/08/arduino-mega2560.html>
- [4] ArduinoAll. “คุณสมบัติของ Arduino Mega 2560.”  
<http://www.arduinoall.com/product/17/arduino-mega-2560-r3>
- [5] Tokopedia. “ส่วนประกอบต่างๆ บนบอร์ด Arduino Mega 2560.”  
<https://www.tokopedia.com/multicopter/arduino-mega-2560-open-source>
- [6] ArduinoAll. “Arduino Nano 3.0.”  
<http://www.arduinoall.com/product/18/arduino-nano-3-0>
- [7] John Boxall. “การควบคุมมอเตอร์.”  
<http://tronixlabs.com/news/tutorial-l298n-dual-motor-controller-module-2a-and-arduino/>
- [8] Go Tronic. “ตัวรับส่งอินฟราเรด TSOP4838.”  
<http://www.gotronic.fr/art-module-ir-38-khz-tsop4838-2323.htm>
- [9] ArduinoAll. “ส่วนประกอบของโมดูลมอเตอร์”  
<http://www.arduitronics.com/product/748/l298n-h-bridge-dual-motor-controller-module>
- [10] Eazyelec. “ไอซีเบอร์555.”  
<http://www.eazyelec.com/article/36-555-ic-timer-B2.html>
- [11] บ้านอิเล็กทรอนิกส์. “โพโทบอร์ด.”  
<http://www.semi-shop.com/knowledge/knowledge.php>
- [12] พิสิกส์ราชมงคล. “ตัวเก็บประจุชนิดอิเล็กโทรไลต์.”  
<http://www.electron.rmutphysics.com/c/>
- [13] Alibaba. “ตัวเก็บประจุชนิดเซรามิก .”  
[https://www.alibaba.com/product-detail/104-Low-Voltage-Ceramic-Capacitor\\_499720320.html](https://www.alibaba.com/product-detail/104-Low-Voltage-Ceramic-Capacitor_499720320.html)

- [14] Commandronestore. “ตัวต้านทานแบบฟิล์มโลหะ.”  
<http://www.amazon.com/Watt-3-9K-Carbon-Film-Resistor/dp/B009XOP02M>
- [15] RungPhung. “ตัวต้านทานแบบโพเทนซิโอมิเตอร์แบบปรับละเอียด.”  
<http://www.diyaudiovillage.net/index.php?topic=18318.120>
- [16] Amazon. “ทรานซิสเตอร์ BD136.”  
<http://www.amazon.com/BD136-1-5A-Power-Transistor-126/dp/B00161B9J4>
- [17] Rapid Electronics. “ไดโอดเปล่งแสงอินฟราเรด.”  
<http://www.rapidonline.com/kingbright-l-7113f3bt-5mm-blue-lens-infrared-diode>
- [18] AliExpress. “โมดูลเข็มทิศ HMC5883L.”  
<http://th.aliexpress.com/item/3V-5V-HMC5883L-Triple-Axis-Compass-Magnetometer-Sensor-Module-For-Arduino-Hot-sale/32259879923.html>
- [19] Myfirstbrain. “ส่วนประกอบของถ่านไฟฉาย.”  
[http://www.myfirstbrain.com/student\\_view.aspx?id=69302](http://www.myfirstbrain.com/student_view.aspx?id=69302)
- [20] Rspssupply. “ถ่านไฟฉายขนาด 2A.”  
<http://www.weloveshopping.com/shop/shop.php?shopid=276334>
- [21] อีสต์เทิร์น เอ็นเนอร์ยี “แบตเตอรี่ขนาด 12V.” <http://eastern-energy.tarad.com/product-th>
- [22] Khonkaenlink. “แบตเตอรี่สำรอง.”  
<http://forum.khonkaenlink.info/index.php?topic=17212607.0>
- [23] Arduitrronics. “สายไฟจัมพ์เปอร์สำเร็จรูป.”  
<http://www.arduitronics.com/product/jumper-male-to-female>



ภาคผนวก ก  
CODE ภาษา C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## CODE ทดสอบการทำงานทั้งหมดของแบบจำลองรถตัดหญ้าโปรแกรมอัตโนมัติ

### ส่วนของ ARDUINO MEGA 2560

```
#include <IRremote.h>
#include <Wire.h>
#include <HMC5883L.h>
#include <EEPROM.h>
HMC5883L compass;
int K =0;
int eeAddress=0;
int IRpin1 = 22;
int IRpin2 = 24;
int IROUT = 28;
int enA = 11;
int in1 = 9;
int in2 = 8;
int enB = 5;
int in3 = 7;
int in4 = 6;
int button4 = 14;
int button5 = 15;
int button6 = 16;
int LEDRED = 2;
int LEDYELLOW = 3;
int LEDGREEN = 4;
int check =0 ;
int checkstatus =0 ;
int i=0,D=0;
int C=20;
int setting=0;
float heading,degree, Deg;
float headingDegree_S=0,headingDegrees=0;
float declinationAngle = 0.26459;
int HEADING_BUFFER=0;
float degreeset=205 ;
unsigned long TimeC=0;
IRrecv irrecv(IRpin1);
decode_results results;
byte Lowbuffer,Highbuffer;
boolean LEDon = true;
float
degree_Buffer_1,degree_Buffer_2,degree_Buffer_3,degree_Buffer_4,d
egree_CAL=0;
int Running = 0 ;
unsigned int READ=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void setup()
{
  Serial.begin(9600);
  Wire.begin();
  compass = HMC5883L();

  Serial.println("Start Program");
  delay(1000);
  pinMode(LEDRED, OUTPUT);pinMode(LEDYELLOW,
OUTPUT);pinMode(LEDGREEN, OUTPUT);pinMode(button4, INPUT);
  pinMode(button4, INPUT);pinMode(button4, INPUT);
  digitalWrite(LEDRED,LOW); delay(10);
  digitalWrite(LEDYELLOW,LOW); delay(10);
  digitalWrite(LEDGREEN,LOW);

  delay(500);
  digitalWrite(LEDRED,HIGH); delay(200);
  digitalWrite(LEDYELLOW,HIGH); delay(200);
  digitalWrite(LEDGREEN,HIGH);
  delay(200); digitalWrite(LEDRED,LOW); delay(200);
  digitalWrite(LEDYELLOW,LOW); delay(200);
  digitalWrite(LEDGREEN,LOW);
  delay(1000);
  digitalWrite(LEDRED,HIGH);

  while(digitalRead(button4)==HIGH ||
digitalRead(button6)==HIGH){

    if(digitalRead(button5)==LOW &&
digitalRead(button4)==LOW){digitalWrite(LEDGREEN,HIGH);
    digitalWrite(LEDYELLOW,LOW);Serial.println("Setup
Mode");setting_degree(); setting=0;}
    if(digitalRead(button5)==LOW){setting=1;
digitalWrite(LEDYELLOW,HIGH); digitalWrite(LEDGREEN,LOW);}
    if(digitalRead(button6)==LOW){digitalWrite(LEDGREEN,HIGH);
digitalWrite(LEDYELLOW,LOW);setting=0;}

  };

  if(setting==0){

digitalWrite(LEDRED,LOW);delay(500);digitalWrite(LEDRED,HIGH);del
ay(500);digitalWrite(LEDRED,LOW);delay(500);digitalWrite(LEDRED,H
IGH);
  delay(500);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

digitalWrite(LEDGREEN,HIGH);delay(300);digitalWrite(LEDGREEN,LOW)
;delay(300); digitalWrite(LEDGREEN,HIGH);delay(300);
digitalWrite(LEDGREEN,LOW);
Serial.println("RUN Mode");
delay(1000);
digitalWrite(LEDGREEN,HIGH);

}
pinMode(IRpin1, INPUT);
pinMode(IRpin2, INPUT);
pinMode(enA, OUTPUT);
pinMode(enB, OUTPUT);
pinMode(in1, OUTPUT);
pinMode(in2, OUTPUT);
pinMode(in3, OUTPUT);
pinMode(in4, OUTPUT);
Wire.begin();
compass = HMC5883L();
Serial.println("Setting scale to +/- 1.3Ga");
int error = compass.SetScale(1.3);
if(error != 0)
Serial.println(compass.GetErrorText(error));

Serial.println("Setting measurement mode to continuous");
error = compass.SetMeasurementMode(Measurement Continuous);
if(error != 0)
Serial.println(compass.GetErrorText(error));
digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
digitalWrite(in3, LOW);
digitalWrite(in4, LOW);

TimeC = millis();
C=200;

Serial.println("LOAD DEGREES");
Lowbuffer=EEPROM.read(eeAddress);
Highbuffer=EEPROM.read(eeAddress+1);
degree_Buffer_1=float(int(Lowbuffer)+int(Highbuffer));
delay(50);
Lowbuffer=EEPROM.read(eeAddress+2);
Highbuffer=EEPROM.read(eeAddress+3);
degree_Buffer_2=float(int(Lowbuffer)+int(Highbuffer));
delay(50);
Lowbuffer=EEPROM.read(eeAddress+4);
Highbuffer=EEPROM.read(eeAddress+5);
degree_Buffer_3=float(int(Lowbuffer)+int(Highbuffer));
delay(50);
Lowbuffer=EEPROM.read(eeAddress+6);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Highbuffer=EEPROM.read(eeAddress+7);
degree_Buffer_4=float(int(Lowbuffer)+int(Highbuffer));

Serial.print("degree1 : "); Serial.print(degree_Buffer_1);
delay(100);
Serial.print(" degree2 : ");Serial.print(degree_Buffer_2);
delay(100);
Serial.print(" degree3 : ");Serial.print(degree_Buffer_3);
delay(100);
Serial.print(" degree4 : ");Serial.println(degree_Buffer_4);
delay(100);
degree=degree_Buffer_1;

Serial.begin(9600);
C=120;
}

void loop()
{
  Read_Degree();
  Cal_Deg();
  if(Deg < -0.5){D=-1; TURN();}
  if(Deg > 0.5) {D=1; TURN();}
  if((Deg > -1.5)&&(Deg < 1.5)){D=0; TURN();}
  if (irrecv.decode(&results)) {
    Serial.println(results.value, HEX);
    irrecv.resume();
    irrecv.enableIRIn();
  }
  if (results.value != 0)
  {
    if((degree_Buffer_1 == degree)){
      degree = degree_Buffer_2;
      Read_Degree();
      IRrecv irrecv(IRpin1);
      decode_results results;
      irrecv.enableIRIn();
      while((Cal_Deg()>5) || (Cal_Deg()<-
5) )
      {
        if(Cal_Deg()>5)D=-2;
        if(Cal_Deg()<5)D=2;
        TURN();

        delay(100);
        D=5; TURN();
      }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

delay(100);
Read_Degree();
}

D=0; TURN();
delay(700);
D=5; TURN();
delay(1000);
degree =

degree_Buffer_3 ;

Read_Degree();

while((Cal_Deg()>5) || (Cal_Deg()<-5) )
{

if(Cal_Deg()>5)D=-2;
if(Cal_Deg()<5)D=2;

TURN();
delay(100);
D=5; TURN();
delay(100);
Read_Degree();
}

TimeC = millis();
} else if((degree==degree_Buffer_3)){
IRrecv irrecv(IRpin2);
decode_results results;
irrecv.enableIRIn();
degree = degree_Buffer_4;
Read_Degree();
while((Cal_Deg()>5) || (Cal_Deg()<-
5) )
{
Read_Degree();

if(Cal_Deg()>5)D=-2;

if(Cal_Deg()<5)D=2;

TURN();
delay(100);
D=5; TURN();
delay(100);
Read_Degree();

if(digitalRead(button6)==LOW) STOP();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

D=0; TURN();
delay(700);
D=5; TURN();
delay(1000);
degree =
degree_Buffer_1;
Read_Degree();
while((Cal_Deg()>5) || (Cal_Deg()<-
5) )
{
Read_Degree();
if(Cal_Deg()>5)D=-2;
if(Cal_Deg()<5)D=2;
TURN();
delay(100);
D=5; TURN();
delay(100);
Read_Degree();
if(digitalRead(button6)==LOW) STOP();
}
TimeC = millis();
C=200;
}
results.value =0;
C=200; irrecv.resume();
}
if(digitalRead(button6)==LOW) STOP();
}
float Read_Degree(void)
{
Wire.begin();
compass = HMC5883L();
i=0;
MagnetometerRaw raw = compass.ReadRawAxis();
float heading = atan2(raw.YAxis, raw.XAxis);
if(heading < 0)
heading += 2*PI;
float declinationAngle = 0.26459;
heading += declinationAngle;
headingDegree_S = (heading * 180/M_PI);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

while(i<C){
    MagnetometerRaw raw = compass.ReadRawAxis();

    float heading = atan2(raw.YAxis, raw.XAxis);
    float declinationAngle = 0.26459;
    heading += declinationAngle;
    (heading < 0)
    heading += 2*PI;

    headingDegrees = (heading * 180/M_PI);

headingDegree_S=headingDegree_S+headingDegrees;
    i++;

}

degree_CAL=(headingDegree_S/(C+1));
Serial.print("          HeadingDegree : ");
Serial.println(degree_CAL);
HEADING_BUFFER = degree_CAL;
}

void CALCULATOR(void)
{
    MagnetometerRaw raw = compass.ReadRawAxis();
    heading = atan2(raw.YAxis, raw.XAxis);
    if(heading < 0)
        heading += 2*PI;
    headingDegrees = heading * 180/M_PI;
    declinationAngle = 0.26459;
    heading += declinationAngle;

    headingDegree_S =
headingDegree_S+headingDegrees;

}

void TURN(void){
    Serial.print("    Direction : ");
    if(D==--1)
    {
        Serial.print("LEFT");
        analogWrite(enA, 200);
        analogWrite(enB, 0);
        digitalWrite(in1, HIGH);
        digitalWrite(in2, LOW);
        digitalWrite(in3, HIGH);
        digitalWrite(in4, LOW);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }

    if(D==1)
    {
        Serial.print("RIGHT");
        analogWrite(enA, 0);
        analogWrite(enB, 200);
        digitalWrite(in1, HIGH);
        digitalWrite(in2, LOW);
        digitalWrite(in3, HIGH);
        digitalWrite(in4, LOW);
    }

    if(D==0)
    {
        Serial.print("Go");
        analogWrite(enB, 150);
        analogWrite(enA, 150);
        digitalWrite(in1, HIGH);
        digitalWrite(in2, LOW);
        digitalWrite(in3, HIGH);
        digitalWrite(in4, LOW);
    }

    if(D==2)
    {
        Serial.print(" ");
        Serial.println("OVER TURN RIGHT");
        analogWrite(enA, 250);
        analogWrite(enB, 250);
        digitalWrite(in1, LOW);
        digitalWrite(in2, HIGH);
        digitalWrite(in3, HIGH);
        digitalWrite(in4, LOW);
    }

    if(D== -2)
    {
        Serial.print(" ");
        Serial.print("OVER TURN LEFT");
        analogWrite(enA, 250);
        analogWrite(enB, 250);
        digitalWrite(in1, HIGH);
        digitalWrite(in2, LOW);
        digitalWrite(in3, LOW);
        digitalWrite(in4, HIGH);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if (D==5)
        {
            Serial.print("                ");
            Serial.print("STOP");
            analogWrite(enA, 0);
            analogWrite(enB, 0);
            digitalWrite(in1, HIGH);
            digitalWrite(in2, HIGH);
            digitalWrite(in3, HIGH);
            digitalWrite(in4, HIGH);
        }
    }

float Cal_Deg() {
    if (degree >= 360);
    Deg = (degree_CAL) - (int (degree) % 360);
    Serial.println("");
    Serial.print(" Degree of Turning : ");
    Serial.println(Deg);
    return Deg;
}

void setting_degree(void)
{
    C=1000;
    digitalWrite(LEDGREEN, LOW);

    digitalWrite(LEDYELLOW, HIGH); delay(500); digitalWrite(LEDYELLOW, LOW);
    delay(500); digitalWrite(LEDYELLOW, HIGH);
    delay(500);
    digitalWrite(LEDYELLOW, LOW); delay(500); digitalWrite(LEDYELLOW, HIGH);
    delay(500);

    while (digitalRead(button5) == HIGH) {
        if (digitalRead(LEDYELLOW) == HIGH)
            digitalWrite(LEDYELLOW, LOW); else
        digitalWrite(LEDYELLOW, LOW);
        Read_Degree();
    }
    digitalWrite(LEDYELLOW, LOW); delay(300);
    digitalWrite(LEDYELLOW, HIGH); delay(300);

    degree_Buffer_1 = HEADING_BUFFER;
    Serial.print(" Degree1 :"); Serial.println(degree_Buffer_1);
    delay(2000);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

while(digitalRead(button5)==HIGH) {
    if(digitalRead(LEDYELLOW)==HIGH)
        digitalWrite(LEDYELLOW,LOW); else
digitalWrite(LEDYELLOW,LOW);
    Read_Degree();
}
digitalWrite(LEDYELLOW,LOW);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(300);
digitalWrite(LEDYELLOW,LOW);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(500);
degree_Buffer_2=HEADING_BUFFER;
Serial.print(" Degree2 :"); Serial.println(degree_Buffer_2);
delay(2000);

while(digitalRead(button5)==HIGH) {
    if(digitalRead(LEDYELLOW)==HIGH)
        digitalWrite(LEDYELLOW,LOW); else
digitalWrite(LEDYELLOW,LOW);
    Read_Degree();
}
digitalWrite(LEDYELLOW,LOW);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(300);
digitalWrite(LEDYELLOW,LOW);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(300);
digitalWrite(LEDYELLOW,LOW);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(300);
degree_Buffer_3=HEADING_BUFFER;
Serial.print(" Degree3 :"); Serial.println(degree_Buffer_3);
delay(2000);

while(digitalRead(button5)==HIGH) {
    if(digitalRead(LEDYELLOW)==HIGH)
        digitalWrite(LEDYELLOW,LOW); else
digitalWrite(LEDYELLOW,LOW);
    Read_Degree();
}
digitalWrite(LEDYELLOW,LOW);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(300);
digitalWrite(LEDYELLOW,LOW);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(300);
digitalWrite(LEDYELLOW,LOW);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(300);
digitalWrite(LEDYELLOW,LOW);delay(300);
digitalWrite(LEDYELLOW,HIGH);delay(300);
degree_Buffer_4=HEADING_BUFFER;
Serial.print(" Degree4 :"); Serial.println(degree_Buffer_4);
delay(1000);
int K =sizeof(float);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

byte highbit,lowbit;
eeAddress=0;

READ= degree_Buffer_1;
highbit = byte(READ%255);
if(READ<255) highbit = 0;
lowbit = byte(READ);
if(READ/255>=1) {lowbit = 255; }
EEPROM.write(eeAddress,lowbit);
EEPROM.write(eeAddress+1,highbit);
Lowbuffer=EEPROM.read(eeAddress);
Highbuffer=EEPROM.read(eeAddress+1);
degree_Buffer_1=(int(Lowbuffer)+int(Highbuffer));
Serial.print(" EEPROM VALUE : ");
Serial.print(degree_Buffer_1);

READ= degree_Buffer_2;
highbit = byte(READ%255);
if(READ<255) highbit = 0;
lowbit = byte(READ);

EEPROM.write(eeAddress+2,lowbit);
EEPROM.write(eeAddress+3,highbit);
Lowbuffer=EEPROM.read(eeAddress+2);
Highbuffer=EEPROM.read(eeAddress+3);
degree_Buffer_2=(int(Lowbuffer)+int(Highbuffer));
Serial.print(" EEPROM VALUE : ");
Serial.print(degree_Buffer_2);

READ= degree_Buffer_3;
highbit = byte(READ%255);
if(READ<255) highbit = 0;
lowbit = byte(READ);
if(READ/255>=1) {lowbit = 255; }
EEPROM.write(eeAddress+4,lowbit);
EEPROM.write(eeAddress+5,highbit);
Lowbuffer=EEPROM.read(eeAddress+4);
Highbuffer=EEPROM.read(eeAddress+5);
degree_Buffer_3=(int(Lowbuffer)+int(Highbuffer));
Serial.print(" EEPROM VALUE : ");
Serial.print(degree_Buffer_3);

READ= degree_Buffer_4;
highbit = byte(READ%255);
if(READ<255) highbit = 0;
lowbit = byte(READ);
if(READ/255>=1) {lowbit = 255; }
EEPROM.write(eeAddress+6,lowbit);
EEPROM.write(eeAddress+7,highbit);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Lowbuffer=EEPROM.read(eeAddress+6);
Highbuffer=EEPROM.read(eeAddress+7);
degree_Buffer_4=(int(Lowbuffer)+int(Highbuffer));
Serial.print(" EEPROM VALUE : ");
Serial.print(degree_Buffer_4);
digitalWrite(LEDGREEN,HIGH);delay(300);
digitalWrite(LEDGREEN,LOW);delay(300);
digitalWrite(LEDGREEN,HIGH);delay(300);
digitalWrite(LEDGREEN,LOW);delay(300);
digitalWrite(LEDYELLOW,LOW);delay(300);
return ;
}
void STOP(void)
{
digitalWrite(LEDGREEN,LOW);
Serial.println("STOP");
D=5; TURN();
delay(2000);
while(digitalRead(button6)==HIGH);
delay(2000);
digitalWrite(LEDGREEN,HIGH);
return ;
}
void loaddegrees(void)
{
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ส่วนของ ARDUINO NANO 3.0

```
#include <IRremote.h>
int IRpin1 = 9;
int IRpin2 = 10;
int IRpin3 = 11;
int IRpinIN = 8;
int IROUT1=5;
int IROUT2=6;
int IR_BUFFER_1 = 0;
int IR_BUFFER_2 = 0;
int IR_BUFFER_3 = 0;
int STATUS=0;
int STATUS1=0;
int check =0 ;
decode_results results[3];
boolean LEDon = true;
void setup()
{
  Serial.begin(9600);
  pinMode(IROUT1, OUTPUT);
  pinMode(IROUT2, OUTPUT);
  digitalWrite(IROUT1,HIGH);
  digitalWrite(IROUT2,HIGH);
}
void loop()
{
  STATUS=0;
  {
    IRrecv irrecv_1(IRpin1);
    irrecv_1.enableIRIn();
    delay(120);
    if (irrecv_1.decode(&results[0])) {
      Serial.println(results[0].value, HEX);
      Serial.println("1");
      irrecv_1.resume();
      //STATUS1=1;
    }
  }

  {
    IRrecv irrecv_1(IRpin2);
    irrecv_1.enableIRIn();
    delay(120);
    if (irrecv_1.decode(&results[0])) {
      Serial.println(results[0].value, HEX);
      Serial.println("2");
      irrecv_1.resume();
    }
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        STATUS=1;
    }
}
{
    IRrecv irrecv_1(IRpin3);
    irrecv_1.enableIRIn();
    delay(120);
    if (irrecv_1.decode(&results[0])) {
        Serial.println(results[0].value, HEX);
        Serial.println("3");
        irrecv_1.resume();
        STATUS1=1;
    }
}

check_TSOP();
Serial.print(STATUS);
Serial.print(",");
Serial.println(STATUS1);
void check_TSOP (void){
    if(STATUS1==1 && STATUS==1){
        Serial.println("STATUS1=2");

        digitalWrite(IROUT2,LOW);
        delay(3000);
        while(digitalRead(IRpinIN)==LOW)
        {
            digitalWrite(IROUT2,HIGH);
        }
        STATUS=0;
        STATUS1=0;
        return;
    }
    if(STATUS==1){
        Serial.println("STATUS=1");
        digitalWrite(IROUT1,LOW);
        delay(3000);
        while(digitalRead(IRpinIN)==LOW)
        {
            digitalWrite(IROUT1,HIGH);
            STATUS=0;
        }
        return;
    }

    if(STATUS1==1){
        Serial.println("STATUS1=1");
        digitalWrite(IROUT1,LOW);
        delay(3000);
        while(digitalRead(IRpinIN)==LOW)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {  
        digitalWrite(IROUT1,LOW);  
    }  
    STATUS1=0;  
    //irrecv.resume();  
    //STATUS=0;  
    //STATUS1=0;  
    return;  
}  
  
return;  
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Photo Modules for PCM Remote Control Systems

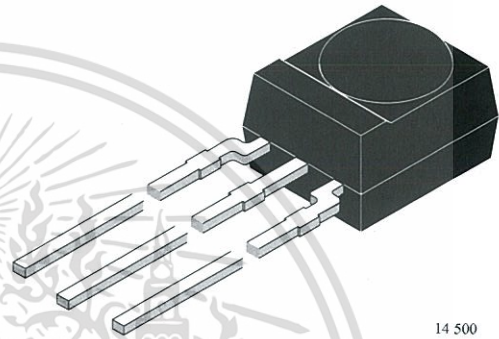
## Available types for different carrier frequencies

Type	fo	Type	fo
TSOP4830	30 kHz	TSOP4833	33 kHz
TSOP4836	36 kHz	TSOP4837	36.7 kHz
TSOP4838	38 kHz	TSOP4840	40 kHz
TSOP4856	56 kHz		

## Description

The TSOP48.. – series are miniaturized receivers for infrared remote control systems. PIN diode and preamplifier are assembled on lead frame, the epoxy package is designed as IR filter.

The demodulated output signal can directly be decoded by a microprocessor. TSOP48.. is the standard IR remote control receiver series, supporting all major transmission codes.

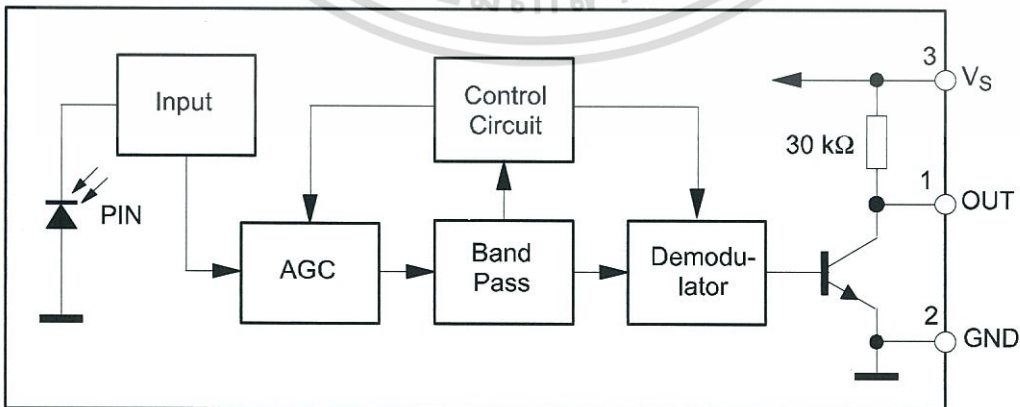


14 500

## Features

- Photo detector and preamplifier in one package
- Internal filter for PCM frequency
- Improved shielding against electrical field disturbance
- TTL and CMOS compatibility
- Output active low
- Low power consumption
- High immunity against ambient light
- Continuous data transmission possible (800 bit/s)
- Suitable burst length  $\geq 10$  cycles/burst

## Block Diagram



9612226

### Absolute Maximum Ratings

$T_{amb} = 25^{\circ}\text{C}$

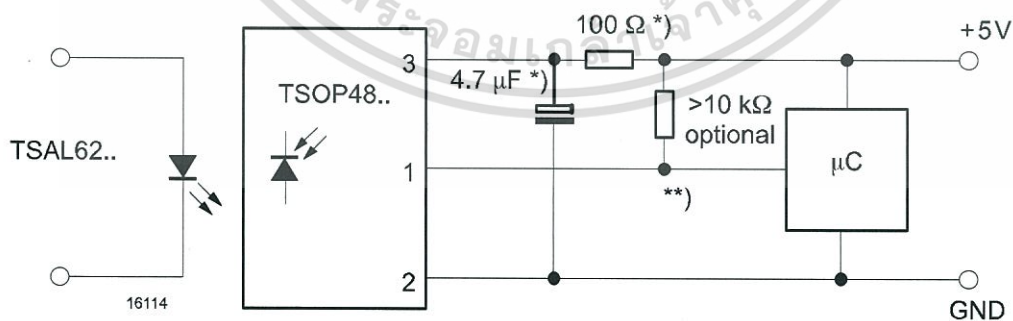
Parameter	Test Conditions	Symbol	Value	Unit
Supply Voltage	(Pin 3)	$V_S$	-0.3...6.0	V
Supply Current	(Pin 3)	$I_S$	5	mA
Output Voltage	(Pin 1)	$V_O$	-0.3...6.0	V
Output Current	(Pin 1)	$I_O$	5	mA
Junction Temperature		$T_j$	100	$^{\circ}\text{C}$
Storage Temperature Range		$T_{stg}$	-25...+85	$^{\circ}\text{C}$
Operating Temperature Range		$T_{amb}$	-25...+85	$^{\circ}\text{C}$
Power Consumption	( $T_{amb} \leq 85^{\circ}\text{C}$ )	$P_{tot}$	50	mW
Soldering Temperature	$t \leq 10\text{ s}$ , 1 mm from case	$T_{sd}$	260	$^{\circ}\text{C}$

### Basic Characteristics

$T_{amb} = 25^{\circ}\text{C}$

Parameter	Test Conditions	Symbol	Min	Typ	Max	Unit
Supply Current (Pin 3)	$V_S = 5\text{ V}$ , $E_v = 0$	$I_{SD}$	0.8	1.1	1.5	mA
	$V_S = 5\text{ V}$ , $E_v = 40\text{ klx}$ , sunlight	$I_{SH}$		1.4		mA
Supply Voltage (Pin 3)		$V_S$	4.5		5.5	V
Transmission Distance	$E_v = 0$ , test signal see fig.7, IR diode TSAL6200, $I_F = 250\text{ mA}$	$d$		35		m
Output Voltage Low (Pin 1)	$I_{OSL} = 0.5\text{ mA}$ , $E_e = 0.7\text{ mW/m}^2$	$V_{OSL}$			250	mV
Irradiance (30 – 40 kHz)	Pulse width tolerance: $t_{pi} - 5/f_o < t_{po} < t_{pi} + 6/f_o$ , test signal see fig.7	$E_{e\ min}$		0.2	0.4	$\text{mW/m}^2$
Irradiance (56 kHz)	Pulse width tolerance: $t_{pi} - 5/f_o < t_{po} < t_{pi} + 6/f_o$ , test signal see fig.7	$E_{e\ min}$		0.3	0.6	$\text{mW/m}^2$
Irradiance	$t_{pi} - 5/f_o < t_{po} < t_{pi} + 6/f_o$	$E_{e\ max}$	30			$\text{W/m}^2$
Directivity	Angle of half transmission distance	$\phi_{1/2}$		$\pm 45$		deg

### Application Circuit



\*) recommended to suppress power supply disturbances

\*\*) The output voltage should not be hold continuously at a voltage below 3.3V by the external circuit.

## Suitable Data Format

The circuit of the TSOP48.. is designed in that way that unexpected output pulses due to noise or disturbance signals are avoided. A bandpassfilter, an integrator stage and an automatic gain control are used to suppress such disturbances.

The distinguishing mark between data signal and disturbance signal are carrier frequency, burst length and duty cycle.

The data signal should fulfill the following condition:

- Carrier frequency should be close to center frequency of the bandpass (e.g. 38kHz).
- Burst length should be 10 cycles/burst or longer.
- After each burst which is between 10 cycles and 70 cycles a gap time of at least 14 cycles is necessary.
- For each burst which is longer than 1.8ms a corresponding gap time is necessary at some time in the data stream. This gap time should be at least 4 times longer than the burst.
- Up to 800 short bursts per second can be received continuously.

Some examples for suitable data format are:

NEC Code, Toshiba Micom Format, Sharp Code, RC5 Code, RC6 Code, R-2000 Code.

When a disturbance signal is applied to the TSOP48.. it can still receive the data signal. However the sensitivity is reduced to that level that no unexpected pulses will occur.

Some examples for such disturbance signals which are suppressed by the TSOP48.. are:

- DC light (e.g. from tungsten bulb or sunlight)
- Continuous signal at 38kHz or at any other frequency
- Signals from fluorescent lamps with electronic ballast with high or low modulation (see Figure A or Figure B).

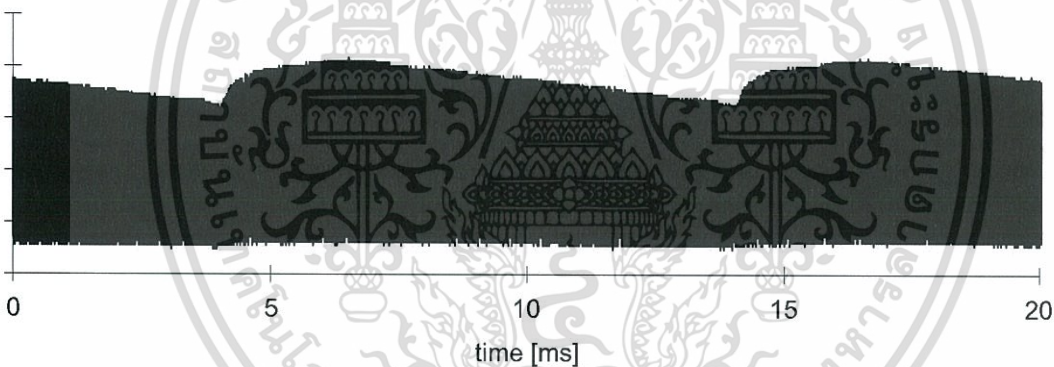


Figure A: IR Signal from Fluorescent Lamp with low Modulation

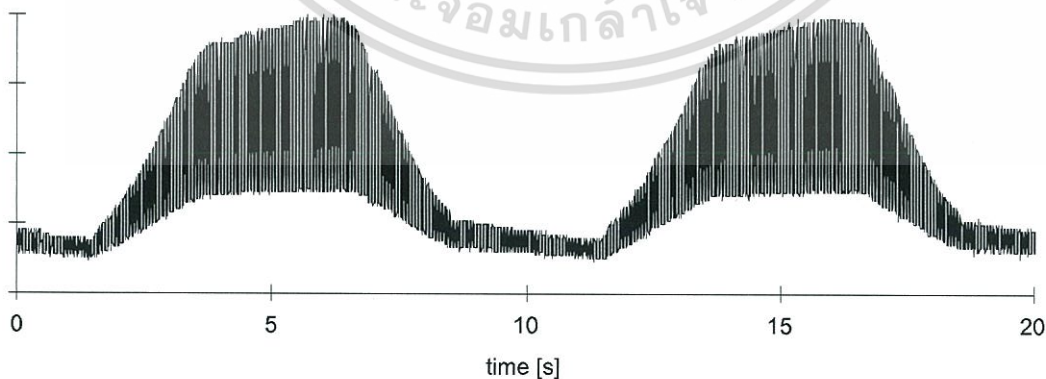


Figure B: IR Signal from Fluorescent Lamp with high Modulation

Typical Characteristics ( $T_{amb} = 25^{\circ}C$  unless otherwise specified)

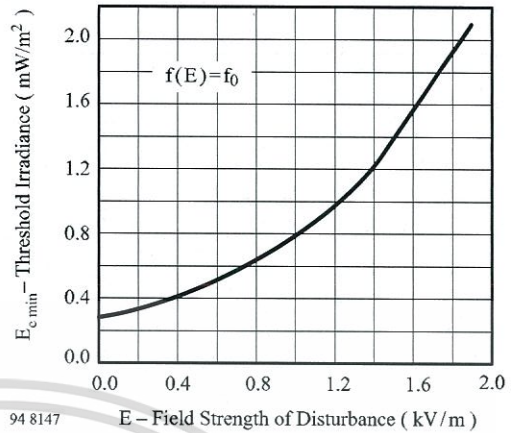
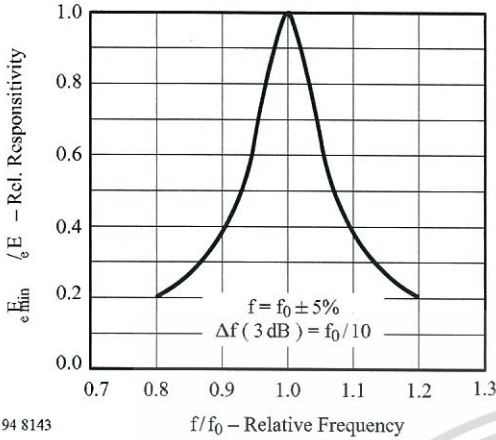


Figure 1. Frequency Dependence of Responsivity

Figure 4. Sensitivity vs. Electric Field Disturbances

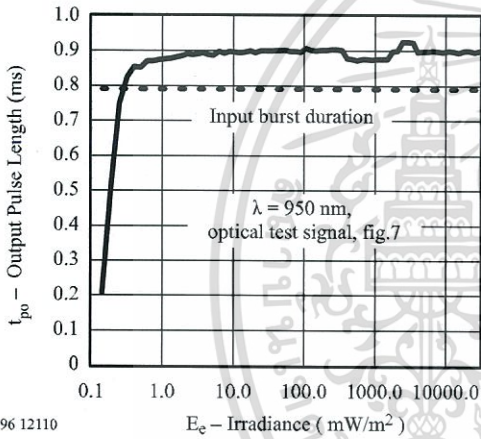


Figure 2. Sensitivity in Dark Ambient

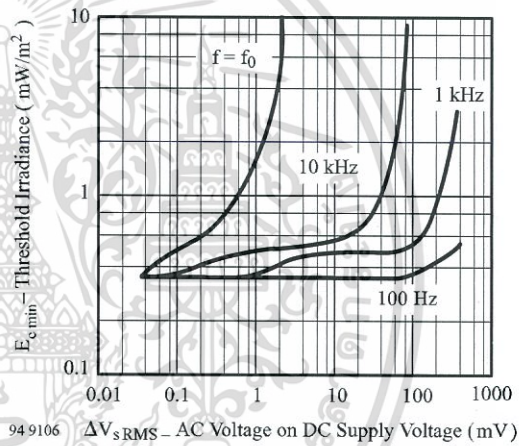


Figure 5. Sensitivity vs. Supply Voltage Disturbances

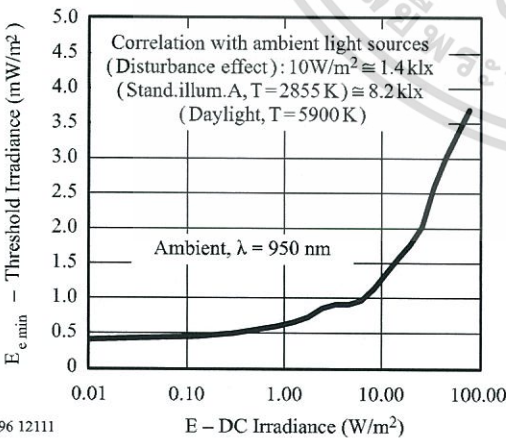


Figure 3. Sensitivity in Bright Ambient

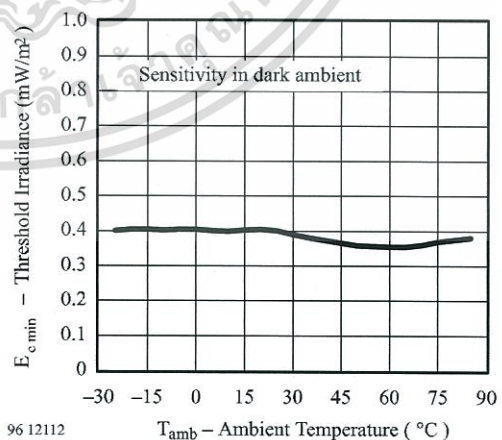


Figure 6. Sensitivity vs. Ambient Temperature

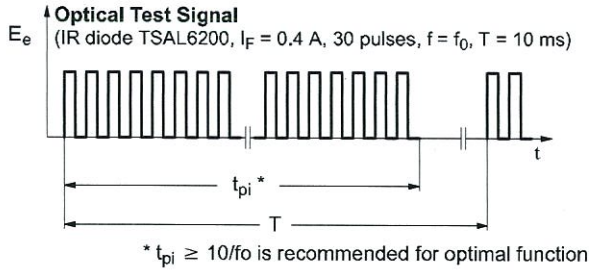


Figure 7.

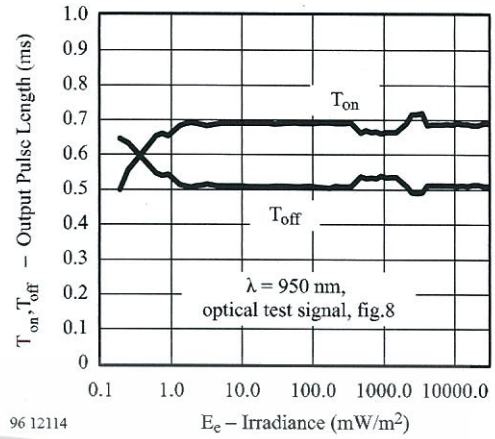


Figure 10. Output Pulse Diagram

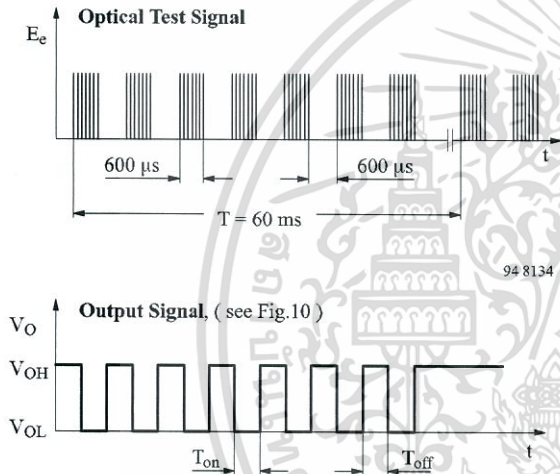


Figure 8. Output Function

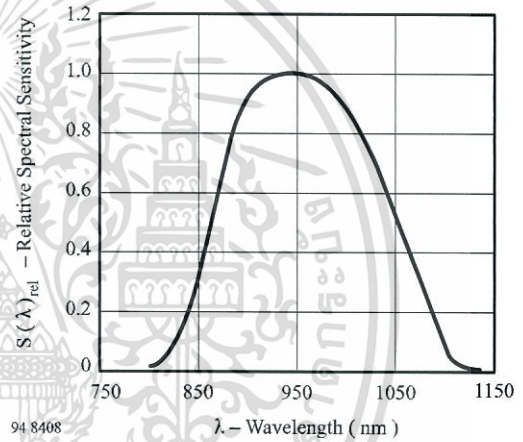


Figure 11. Relative Spectral Sensitivity vs. Wavelength

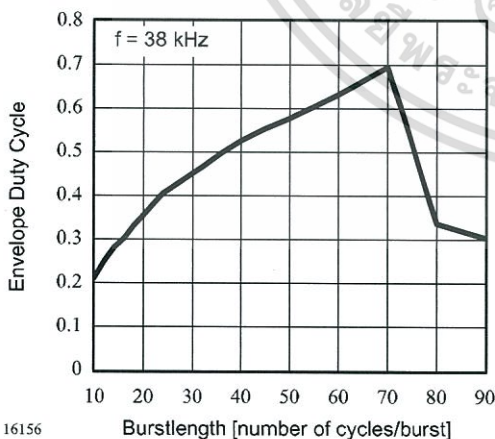


Figure 9. Max. Envelope Duty Cycle vs. Burstlength

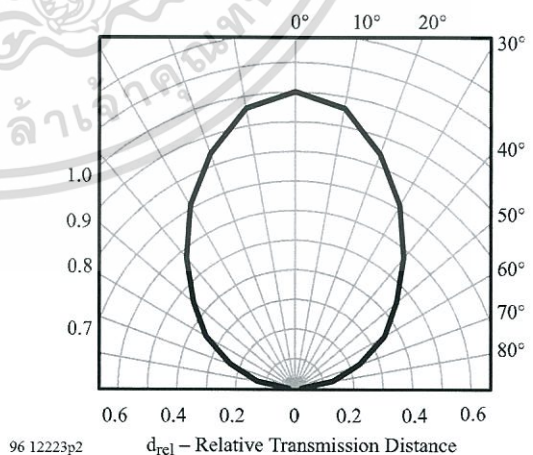


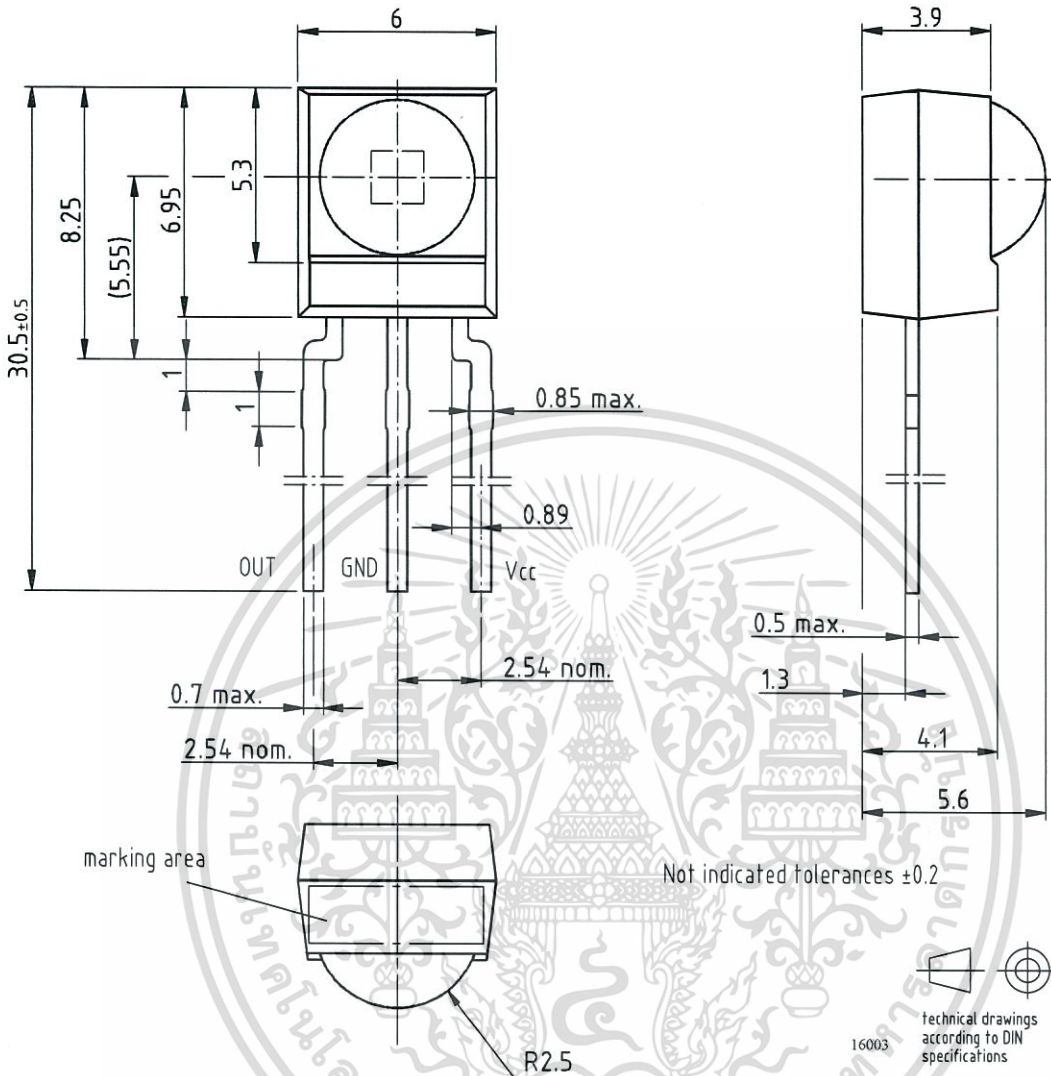
Figure 12. Directivity

# TSOP48..

Vishay Telefunken



Dimensions in mm



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Ozone Depleting Substances Policy Statement

It is the policy of **Vishay Semiconductor GmbH** to

1. Meet all present and future national and international statutory requirements.
2. Regularly and continuously improve the performance of our products, processes, distribution and operating systems with respect to their impact on the health and safety of our employees and the public, as well as their impact on the environment.

It is particular concern to control or eliminate releases of those substances into the atmosphere which are known as ozone depleting substances (ODSs).

The Montreal Protocol (1987) and its London Amendments (1990) intend to severely restrict the use of ODSs and forbid their use within the next ten years. Various national and international initiatives are pressing for an earlier ban on these substances.

**Vishay Semiconductor GmbH** has been able to use its policy of continuous improvements to eliminate the use of ODSs listed in the following documents.

1. Annex A, B and list of transitional substances of the Montreal Protocol and the London Amendments respectively
2. Class I and II ozone depleting substances in the Clean Air Act Amendments of 1990 by the Environmental Protection Agency (EPA) in the USA
3. Council Decision 88/540/EEC and 91/690/EEC Annex A, B and C (transitional substances) respectively.

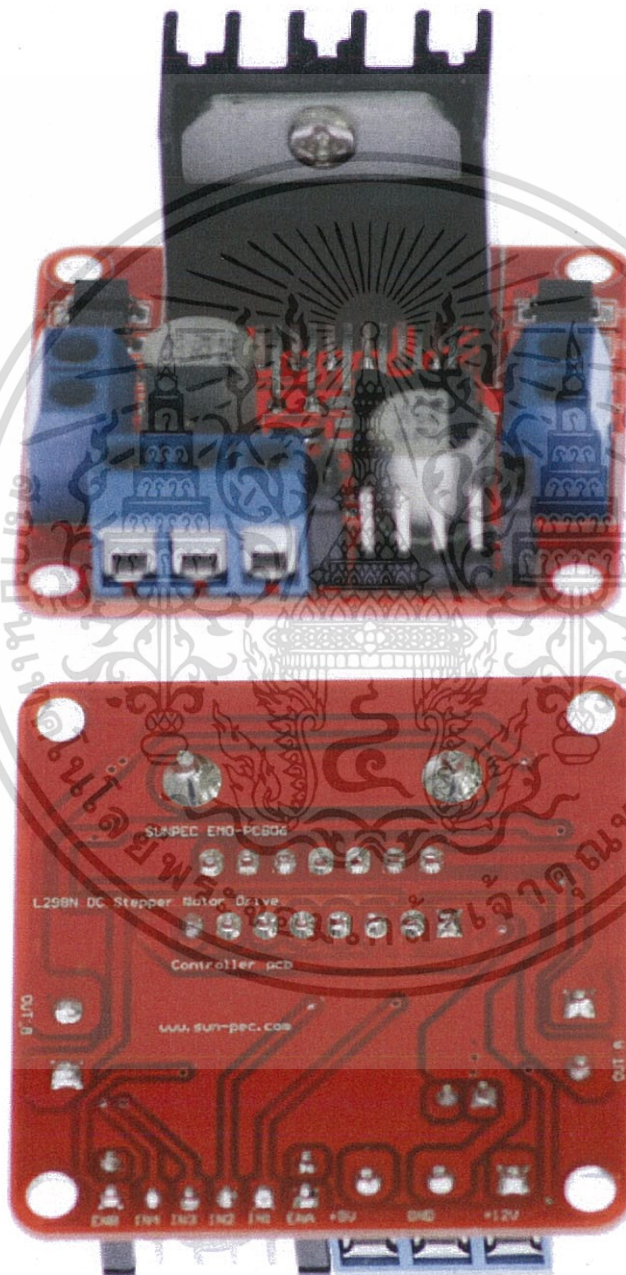
**Vishay Semiconductor GmbH** can certify that our semiconductors are not manufactured with ozone depleting substances and do not contain such substances.

**We reserve the right to make changes to improve technical design and may do so without further notice.**

Parameters can vary in different applications. All operating parameters must be validated for each customer application by the customer. Should the buyer use Vishay-Telefunken products for any unintended or unauthorized application, the buyer shall indemnify Vishay-Telefunken against all claims, costs, damages, and expenses, arising out of, directly or indirectly, any claim of personal damage, injury or death associated with such unintended or unauthorized use.

Vishay Semiconductor GmbH, P.O.B. 3535, D-74025 Heilbronn, Germany  
Telephone: 49 (0)7131 67 2831, Fax number: 49 (0)7131 67 2423

## Overview

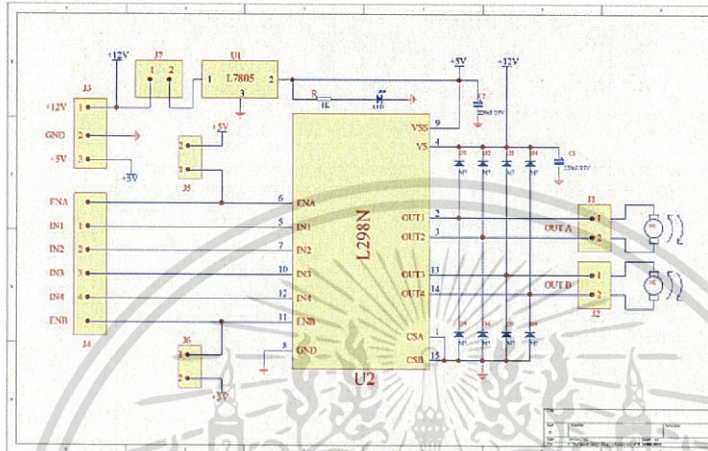


[www.sun-pec.com](http://www.sun-pec.com)

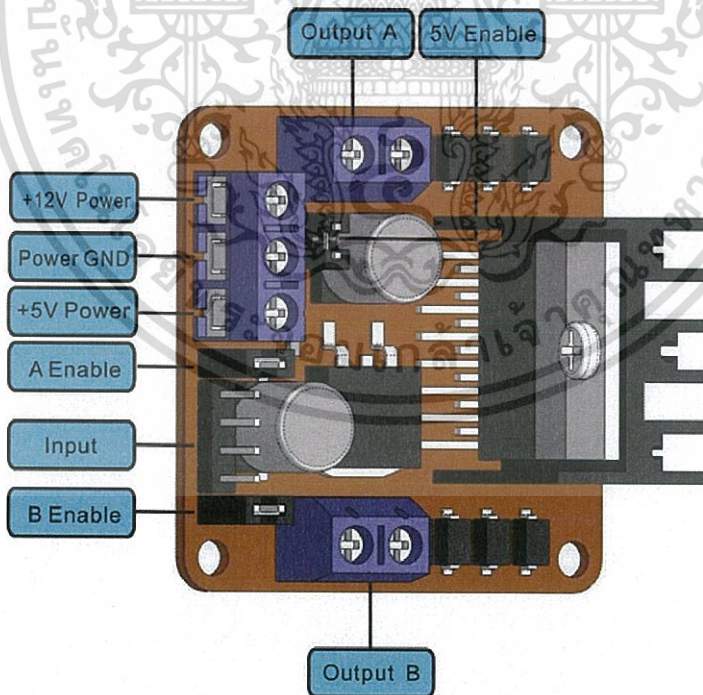
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Electrical Drawing

### Schematic



### Diagram



## Description

Double H Bridge Motor Driver Module	<b>Working mode</b>	Driven by H bridge (double lines)	
<b>Control chip</b>	L298N (ST)		
<b>Logical voltage</b>	5V	<b>Driving voltage</b>	5V—35V
<b>Logical current</b>	0mA-36mA	<b>Driving current</b>	2A (MAX single bridge)
<b>Storage temperature</b>	(-20°C) — (+135°C)	<b>Maximum power</b>	25W
<b>weight</b>	30g	<b>Periphery dimension</b>	43*43*27mm

### Product features:

1. Using L298N made by ST company as the control chip, the module has such characteristics as strong driving ability, low calorific value and strong anti-interference ability.
2. This module can use built-in 78M05 for electric work via a driving power supply part. But to avoid the damage of the voltage stabilizing chip, please use an external 5V logic supply when using more than 12V driving voltage.
3. Using large capacity filter capacitor, this module can follow current to protect diodes, and improve there liability

This module is integrated with a built-in 5V power. When the drive voltage is 7V-35V, it can enable the onboard 5V logic power supply; after the power supply, don't input voltage in the interface+5V power supply, but you can lead the 5V for external use.

When ENA enable IN1 IN2 control OUT1 OUT2

When ENB enable IN3 IN4 control OUT3 OUT4

[www.sun-pec.com](http://www.sun-pec.com)

## Description

### Applied cases:

#### 1 Driving stepper motor

The connection of driving a common 4 line 2 phase electric motor is shown in below figure

after enable ENA ENB

Input the following driving timing from IN1-IN4, then the speed and direction of the stepper

motor can be controlled

stepper motor	signal input	step 1	step 2	step 3	step 4	return to step1
corotation	IN1	0	1	1	1	return
	IN2	1	0	1	1	return
	IN3	1	1	0	1	return
	IN4	1	1	1	0	return
reversal	IN1	1	1	1	0	return
	IN2	1	1	0	1	return
	IN3	1	0	1	1	return
	IN4	0	1	1	1	return

#### 2 Driving DC motor

Because the module is driven by double H bridge, it can drive two motors simultaneously.

The connecting method is shown in below figure

after enable ENA ENB

You can input the speed and direction of PWM signal drive motor 1 from IN1 IN2

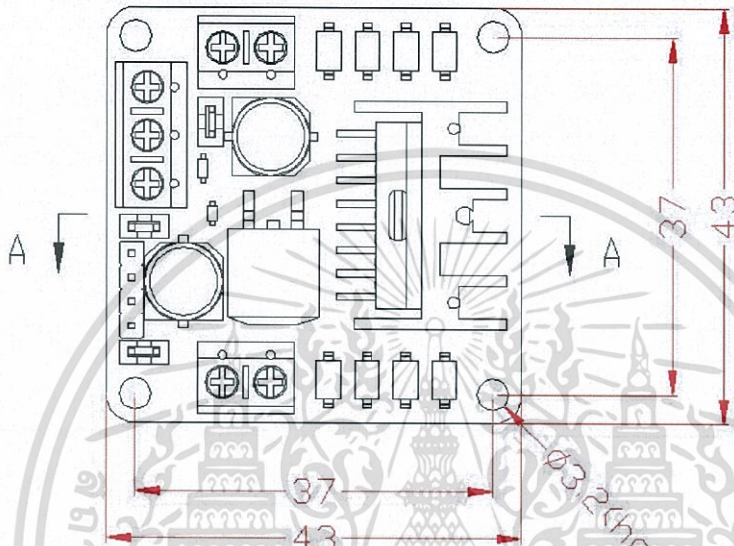
You can input the speed and direction of PWM signal drive motor 2 from IN1 IN2

The signal is shown in the figure

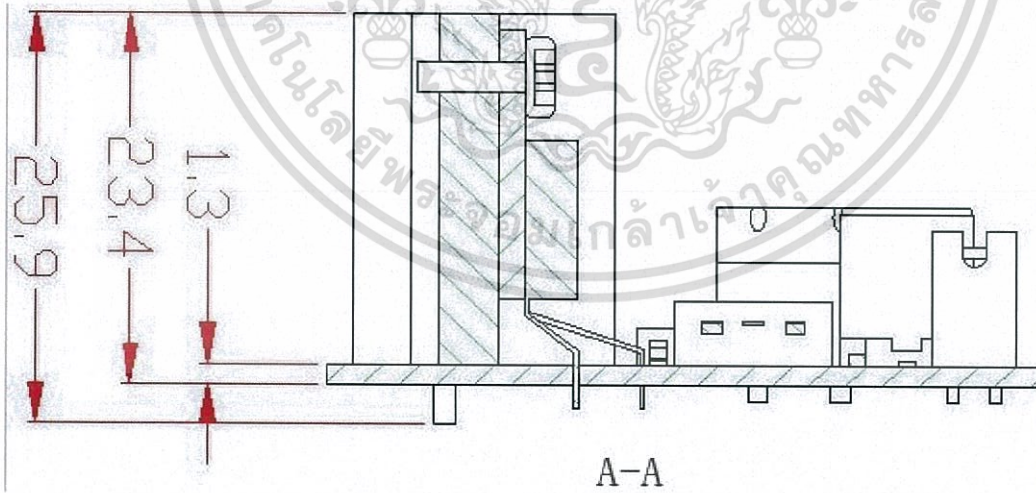
DC motor	rotate	IN1	IN2	IN3	IN4	speed adjust PWM signal	
						end	end
M1	corotation	high	low	/	/	high	/
	reversal	low	high	/	/	high	/
	stop	low	low	/	/	high	/
M2	corotation	/	/	high	low	/	high
	reversal	/	/	low	high	/	high
	stop	/	/	low	low	/	high

## Mechanical Drawing

### Top Size



### Side size



---

---

## Contact us

---

---

SUN-Pec Co., Ltd .

Unit E, 17/F, Success Ind. Building, 17 Sheung Hei Street, San  
Po Kong, HK

Tel: +852-23549977

Fax: +852-23549355

Website: [www.sun-pec.com](http://www.sun-pec.com)

Email: [contact@sun-pec.com](mailto:contact@sun-pec.com)



---

[www.sun-pec.com](http://www.sun-pec.com)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# LM555/NE555/SA555

## Single Timer

### Features

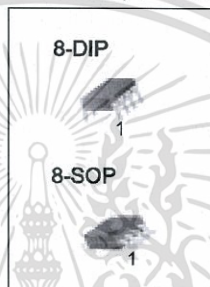
- High Current Drive Capability (200mA)
- Adjustable Duty Cycle
- Temperature Stability of 0.005%/°C
- Timing From  $\mu$ Sec to Hours
- Turn off Time Less Than 2 $\mu$ Sec

### Description

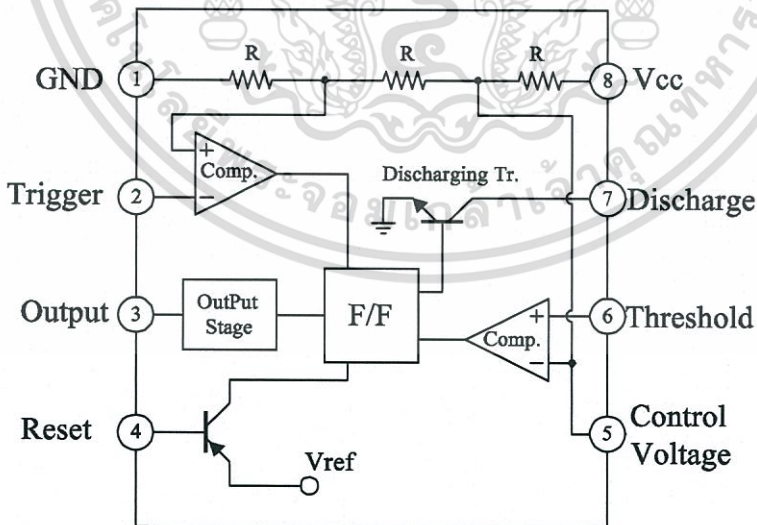
The LM555/NE555/SA555 is a highly stable controller capable of producing accurate timing pulses. With a monostable operation, the time delay is controlled by one external resistor and one capacitor. With an astable operation, the frequency and duty cycle are accurately controlled by two external resistors and one capacitor.

### Applications

- Precision Timing
- Pulse Generation
- Time Delay Generation
- Sequential Timing



### Internal Block Diagram



Rev. 1.0.3

**Absolute Maximum Ratings (TA = 25°C)**

Parameter	Symbol	Value	Unit
Supply Voltage	VCC	16	V
Lead Temperature (Soldering 10sec)	TLEAD	300	°C
Power Dissipation	PD	600	mW
Operating Temperature Range LM555/NE555 SA555	TOPR	0 ~ +70 -40 ~ +85	°C
Storage Temperature Range	TSTG	-65 ~ +150	°C



## Electrical Characteristics

( $T_A = 25^\circ\text{C}$ ,  $V_{CC} = 5 \sim 15\text{V}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Supply Voltage	$V_{CC}$	-	4.5	-	16	V
Supply Current (Low Stable) (Note1)	$I_{CC}$	$V_{CC} = 5\text{V}$ , $R_L = \infty$	-	3	6	mA
		$V_{CC} = 15\text{V}$ , $R_L = \infty$	-	7.5	15	mA
Timing Error (Monostable) Initial Accuracy (Note2) Drift with Temperature (Note4) Drift with Supply Voltage (Note4)	ACCUR $\Delta t/\Delta T$ $\Delta t/\Delta V_{CC}$	$R_A = 1\text{k}\Omega$ to $100\text{k}\Omega$ $C = 0.1\mu\text{F}$	-	1.0 50 0.1	3.0 - 0.5	% ppm/ $^\circ\text{C}$ %/V
Timing Error (Astable) Initial Accuracy (Note2) Drift with Temperature (Note4) Drift with Supply Voltage (Note4)	ACCUR $\Delta t/\Delta T$ $\Delta t/\Delta V_{CC}$	$R_A = 1\text{k}\Omega$ to $100\text{k}\Omega$ $C = 0.1\mu\text{F}$	-	2.25 150 0.3	-	% ppm/ $^\circ\text{C}$ %/V
Control Voltage	$V_C$	$V_{CC} = 15\text{V}$	9.0	10.0	11.0	V
		$V_{CC} = 5\text{V}$	2.6	3.33	4.0	V
Threshold Voltage	$V_{TH}$	$V_{CC} = 15\text{V}$	-	10.0	-	V
		$V_{CC} = 5\text{V}$	-	3.33	-	V
Threshold Current (Note3)	$I_{TH}$	-	-	0.1	0.25	$\mu\text{A}$
Trigger Voltage	$V_{TR}$	$V_{CC} = 5\text{V}$	1.1	1.67	2.2	V
		$V_{CC} = 15\text{V}$	4.5	5	5.6	V
Trigger Current	$I_{TR}$	$V_{TR} = 0\text{V}$	-	0.01	2.0	$\mu\text{A}$
Reset Voltage	$V_{RST}$	-	0.4	0.7	1.0	V
Reset Current	$I_{RST}$	-	-	0.1	0.4	mA
Low Output Voltage	$V_{OL}$	$V_{CC} = 15\text{V}$ $I_{SINK} = 10\text{mA}$ $I_{SINK} = 50\text{mA}$	-	0.06 0.3	0.25 0.75	V V
		$V_{CC} = 5\text{V}$ $I_{SINK} = 5\text{mA}$	-	0.05	0.35	V
High Output Voltage	$V_{OH}$	$V_{CC} = 15\text{V}$ $I_{SOURCE} = 200\text{mA}$ $I_{SOURCE} = 100\text{mA}$	12.75	12.5 13.3	-	V V
		$V_{CC} = 5\text{V}$ $I_{SOURCE} = 100\text{mA}$	2.75	3.3	-	V
Rise Time of Output (Note4)	$t_R$	-	-	100	-	ns
Fall Time of Output (Note4)	$t_F$	-	-	100	-	ns
Discharge Leakage Current	$I_{LKG}$	-	-	20	100	nA

### Notes:

- When the output is high, the supply current is typically 1mA less than at  $V_{CC} = 5\text{V}$ .
- Tested at  $V_{CC} = 5.0\text{V}$  and  $V_{CC} = 15\text{V}$ .
- This will determine the maximum value of  $R_A + R_B$  for 15V operation, the max. total  $R = 20\text{M}\Omega$ , and for 5V operation, the max. total  $R = 6.7\text{M}\Omega$ .
- These parameters, although guaranteed, are not 100% tested in production.

## Application Information

Table 1 below is the basic operating table of 555 timer:

Table 1. Basic Operating Table

Threshold Voltage (V <sub>th</sub> )(PIN 6)	Trigger Voltage (V <sub>tr</sub> )(PIN 2)	Reset(PIN 4)	Output(PIN 3)	Discharging Tr. (PIN 7)
Don't care	Don't care	Low	Low	ON
$V_{th} > 2V_{cc} / 3$	$V_{th} > 2V_{cc} / 3$	High	Low	ON
$V_{cc} / 3 < V_{th} < 2 V_{cc} / 3$	$V_{cc} / 3 < V_{th} < 2 V_{cc} / 3$	High	-	-
$V_{th} < V_{cc} / 3$	$V_{th} < V_{cc} / 3$	High	High	OFF

When the low signal input is applied to the reset terminal, the timer output remains low regardless of the threshold voltage or the trigger voltage. Only when the high signal is applied to the reset terminal, the timer's output changes according to threshold voltage and trigger voltage.

When the threshold voltage exceeds 2/3 of the supply voltage while the timer output is high, the timer's internal discharge Tr. turns on, lowering the threshold voltage to below 1/3 of the supply voltage. During this time, the timer output is maintained low. Later, if a low signal is applied to the trigger voltage so that it becomes 1/3 of the supply voltage, the timer's internal discharge Tr. turns off, increasing the threshold voltage and driving the timer output again at high.

### 1. Monostable Operation

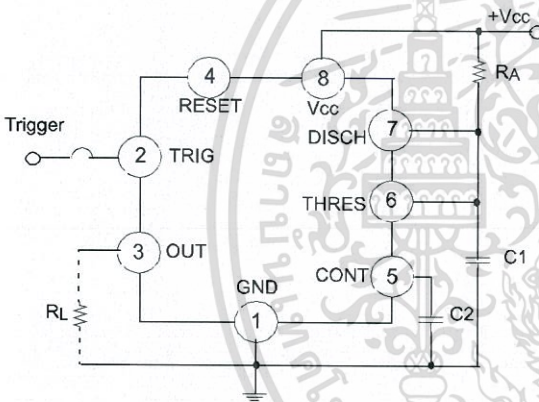


Figure 1. Monoatable Circuit

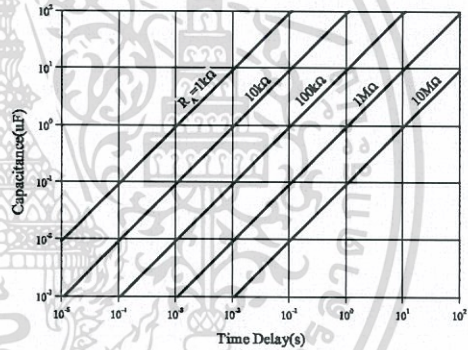
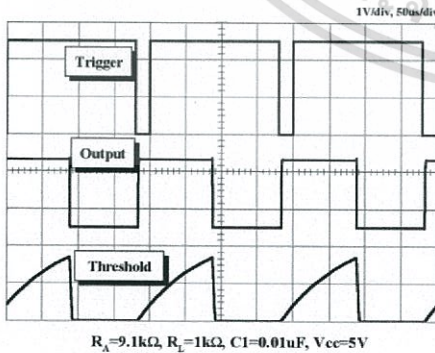


Figure 2. Resistance and Capacitance vs. Time delay( $t_d$ )



$R_A=9.1k\Omega, R_T=1k\Omega, C1=0.01\mu F, V_{cc}=5V$

Figure 3. Waveforms of Monostable Operation

Figure 1 illustrates a monostable circuit. In this mode, the timer generates a fixed pulse whenever the trigger voltage falls below  $V_{cc}/3$ . When the trigger pulse voltage applied to the #2 pin falls below  $V_{cc}/3$  while the timer output is low, the timer's internal flip-flop turns the discharging Tr. off and causes the timer output to become high by charging the external capacitor C1 and setting the flip-flop output at the same time.

The voltage across the external capacitor C1,  $V_{C1}$  increases exponentially with the time constant  $\tau = R_A * C$  and reaches  $2V_{cc}/3$  at  $t_d = 1.1R_A * C$ . Hence, capacitor C1 is charged through resistor  $R_A$ . The greater the time constant  $R_A C$ , the longer it takes for the  $V_{C1}$  to reach  $2V_{cc}/3$ . In other words, the time constant  $R_A C$  controls the output pulse width.

When the applied voltage to the capacitor C1 reaches  $2V_{cc}/3$ , the comparator on the trigger terminal resets the flip-flop, turning the discharging Tr. on. At this time, C1 begins to discharge and the timer output converts to low.

In this way, the timer operating in the monostable repeats the above process. Figure 2 shows the time constant relationship based on  $R_A$  and C. Figure 3 shows the general waveforms during the monostable operation.

It must be noted that, for a normal operation, the trigger pulse voltage needs to maintain a minimum of  $V_{cc}/3$  before the timer output turns low. That is, although the output remains unaffected even if a different trigger pulse is applied while the output is high, it may be affected and the waveform does not operate properly if the trigger pulse voltage at the end of the output pulse remains at below  $V_{cc}/3$ . Figure 4 shows such a timer output abnormality.

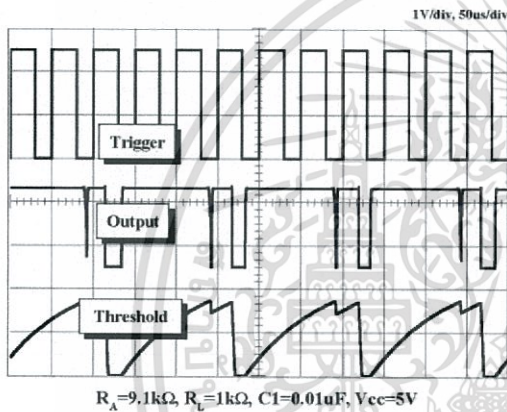


Figure 4. Waveforms of Monostable Operation (abnormal)

## 2. Astable Operation

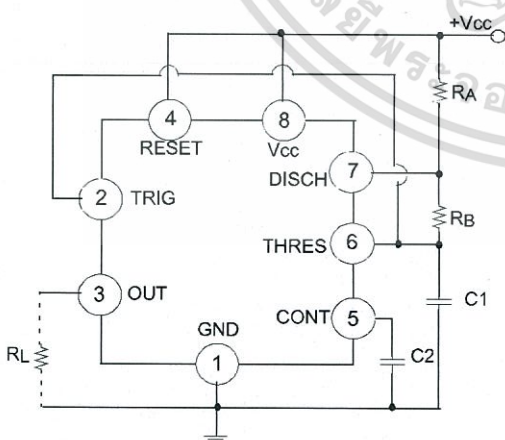


Figure 5. Astable Circuit

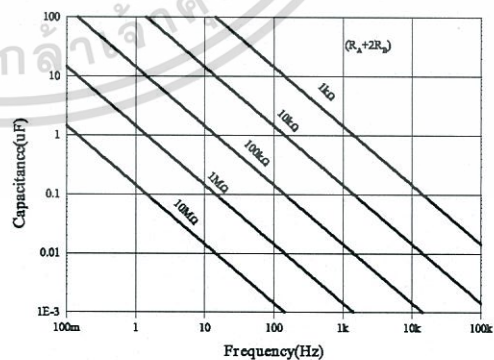


Figure 6. Capacitance and Resistance vs. Frequency

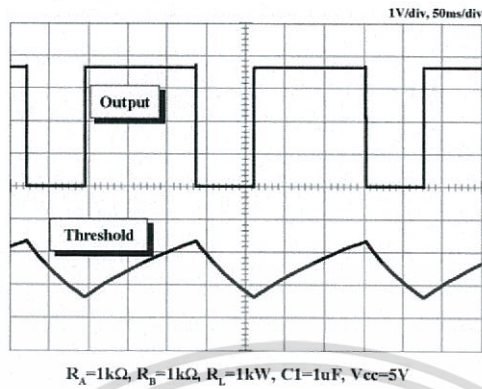
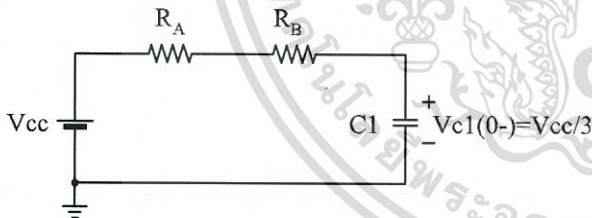


Figure 7. Waveforms of Astable Operation

An astable timer operation is achieved by adding resistor  $R_B$  to Figure 1 and configuring as shown on Figure 5. In the astable operation, the trigger terminal and the threshold terminal are connected so that a self-trigger is formed, operating as a multi vibrator. When the timer output is high, its internal discharging  $Tr$  turns off and the  $V_{C1}$  increases by exponential function with the time constant  $(R_A+R_B)*C$ .

When the  $V_{C1}$ , or the threshold voltage, reaches  $2V_{CC}/3$ , the comparator output on the trigger terminal becomes high, resetting the F/F and causing the timer output to become low. This in turn turns on the discharging  $Tr$  and the  $C1$  discharges through the discharging channel formed by  $R_B$  and the discharging  $Tr$ . When the  $V_{C1}$  falls below  $V_{CC}/3$ , the comparator output on the trigger terminal becomes high and the timer output becomes high again. The discharging  $Tr$  turns off and the  $V_{C1}$  rises again.

In the above process, the section where the timer output is high is the time it takes for the  $V_{C1}$  to rise from  $V_{CC}/3$  to  $2V_{CC}/3$ , and the section where the timer output is low is the time it takes for the  $V_{C1}$  to drop from  $2V_{CC}/3$  to  $V_{CC}/3$ . When timer output is high, the equivalent circuit for charging capacitor  $C1$  is as follows:



$$C_1 \frac{dv_{c1}}{dt} = \frac{V_{CC} - V(0-)}{R_A + R_B} \quad (1)$$

$$V_{C1}(0+) = V_{CC}/3 \quad (2)$$

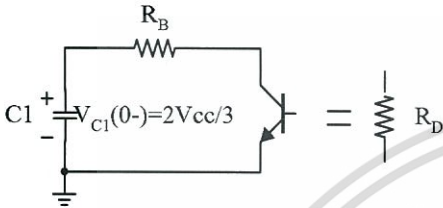
$$V_{C1}(t) = V_{CC} \left( 1 - \frac{2}{3} e^{-\left( \frac{t}{(R_A + R_B)C_1} \right)} \right) \quad (3)$$

Since the duration of the timer output high state( $t_H$ ) is the amount of time it takes for the  $V_{C1}(t)$  to reach  $2V_{CC}/3$ ,

$$V_{C1}(t) = \frac{2}{3}V_{CC} = V_{CC} \left( 1 - \frac{2}{3} e^{-\left(\frac{t_H}{(R_A + R_B)C_1}\right)} \right) \quad (4)$$

$$t_H = C_1(R_A + R_B) \ln 2 = 0.693(R_A + R_B)C_1 \quad (5)$$

The equivalent circuit for discharging capacitor C1, when timer output is low is, as follows:



$$C_1 \frac{dV_{C1}}{dt} + \frac{1}{R_A + R_B} V_{C1} = 0 \quad (6)$$

$$V_{C1}(t) = \frac{2}{3} V_{CC} e^{-\frac{t}{(R_A + R_D)C_1}} \quad (7)$$

Since the duration of the timer output low state ( $t_L$ ) is the amount of time it takes for the  $V_{C1}(t)$  to reach  $V_{CC}/3$ ,

$$\frac{1}{3} V_{CC} = \frac{2}{3} V_{CC} e^{-\frac{t_L}{(R_A + R_D)C_1}} \quad (8)$$

$$t_L = C_1(R_B + R_D) \ln 2 = 0.693(R_B + R_D)C_1 \quad (9)$$

Since  $R_D$  is normally  $R_B \gg R_D$  although related to the size of discharging  $T_r$ ,  
 $t_L = 0.693R_B C_1$  (10)

Consequently, if the timer operates in astable, the period is the same with ' $T = t_H + t_L = 0.693(R_A + R_B)C_1 + 0.693R_B C_1 = 0.693(R_A + 2R_B)C_1$ ' because the period is the sum of the charge time and discharge time. And since frequency is the reciprocal of the period, the following applies.

$$\text{frequency, } f = \frac{1}{T} = \frac{1.44}{(R_A + 2R_B)C_1} \quad (11)$$

### 3. Frequency divider

By adjusting the length of the timing cycle, the basic circuit of Figure 1 can be made to operate as a frequency divider. Figure 8. illustrates a divide-by-three circuit that makes use of the fact that retriggering cannot occur during the timing cycle.

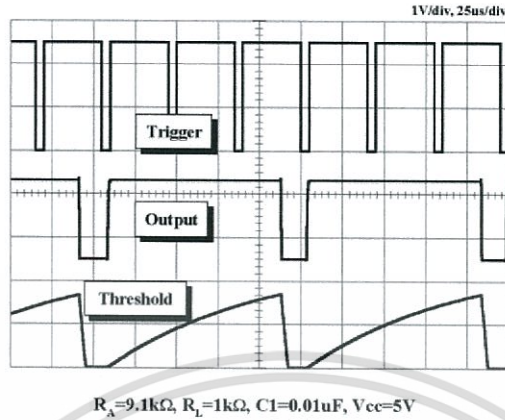


Figure 8. Waveforms of Frequency Divider Operation

#### 4. Pulse Width Modulation

The timer output waveform may be changed by modulating the control voltage applied to the timer's pin 5 and changing the reference of the timer's internal comparators. Figure 9 illustrates the pulse width modulation circuit.

When the continuous trigger pulse train is applied in the monostable mode, the timer output width is modulated according to the signal applied to the control terminal. Sine wave as well as other waveforms may be applied as a signal to the control terminal. Figure 10 shows the example of pulse width modulation waveform.

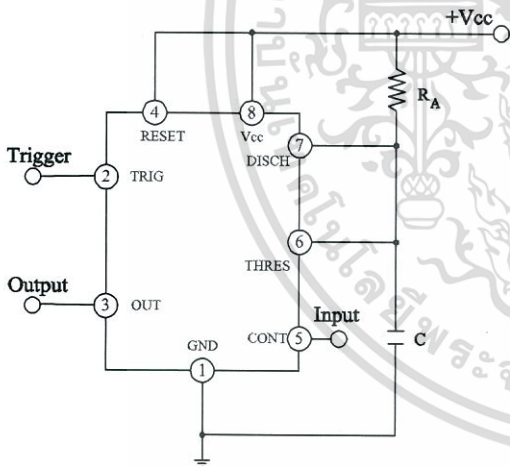


Figure 9. Circuit for Pulse Width Modulation

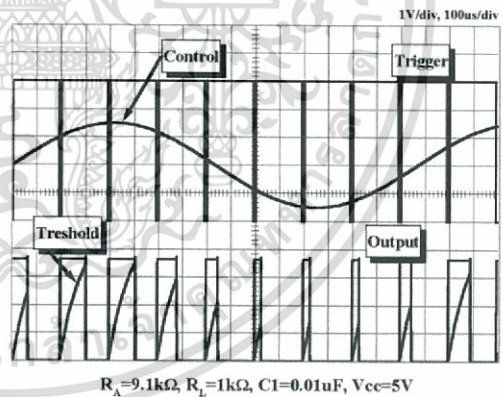


Figure 10. Waveforms of Pulse Width Modulation

#### 5. Pulse Position Modulation

If the modulating signal is applied to the control terminal while the timer is connected for the astable operation as in Figure 11, the timer becomes a pulse position modulator.

In the pulse position modulator, the reference of the timer's internal comparators is modulated which in turn modulates the timer output according to the modulation signal applied to the control terminal.

Figure 12 illustrates a sine wave for modulation signal and the resulting output pulse position modulation : however, any wave shape could be used.

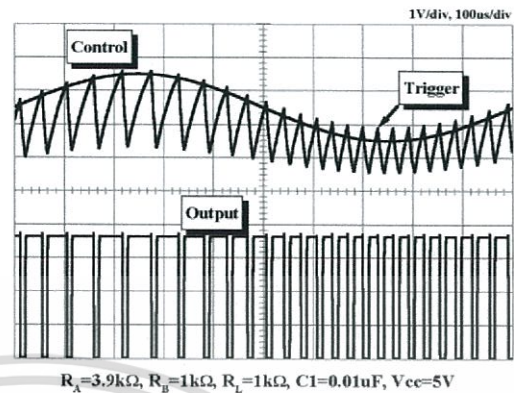
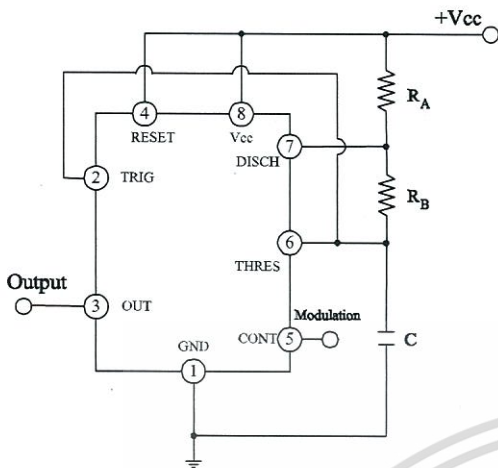


Figure 11. Circuit for Pulse Position Modulation

Figure 12. Waveforms of pulse position modulation

## 6. Linear Ramp

When the pull-up resistor  $R_A$  in the monostable circuit shown in Figure 1 is replaced with constant current source, the  $V_{C1}$  increases linearly, generating a linear ramp. Figure 13 shows the linear ramp generating circuit and Figure 14 illustrates the generated linear ramp waveforms.

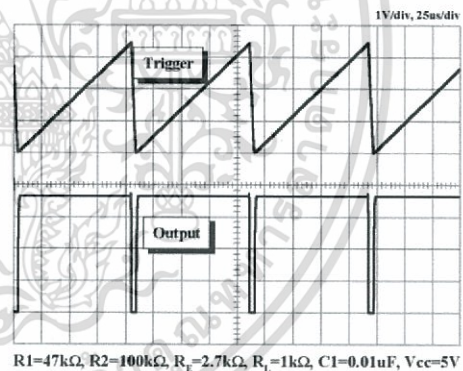
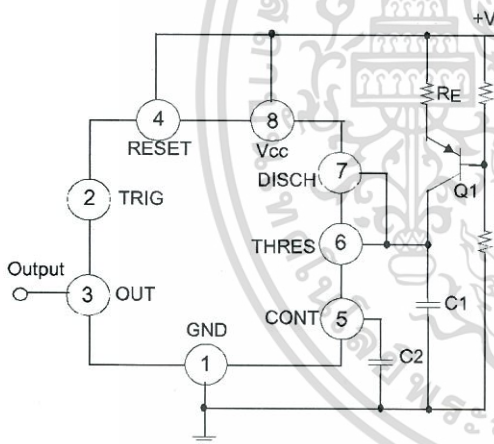


Figure 13. Circuit for Linear Ramp

Figure 14. Waveforms of Linear Ramp

In Figure 13, current source is created by PNP transistor  $Q_1$  and resistor  $R_1$ ,  $R_2$ , and  $R_E$ .

$$I_C = \frac{V_{CC} - V_E}{R_E} \quad (12)$$

Here,  $V_E$  is

$$V_E = V_{BE} + \frac{R_2}{R_1 + R_2} V_{CC} \quad (13)$$

For example, if  $V_{CC}=15V$ ,  $R_E=20k\Omega$ ,  $R_1=5k\Omega$ ,  $R_2=10k\Omega$ , and  $V_{BE}=0.7V$ ,  
 $V_E=0.7V+10V=10.7V$   
 $I_C=(15-10.7)/20k=0.215mA$

When the trigger starts in a timer configured as shown in Figure 13, the current flowing through capacitor C1 becomes a constant current generated by PNP transistor and resistors.

Hence, the VC is a linear ramp function as shown in Figure 14. The gradient S of the linear ramp function is defined as follows:

$$S = \frac{V_{p-p}}{T} \quad (14)$$

Here the  $V_{p-p}$  is the peak-to-peak voltage.

If the electric charge amount accumulated in the capacitor is divided by the capacitance, the VC comes out as follows:

$$V=Q/C \quad (15)$$

The above equation divided on both sides by T gives us

$$\frac{V}{T} = \frac{Q/T}{C} \quad (16)$$

and may be simplified into the following equation.

$$S=I/C \quad (17)$$

In other words, the gradient of the linear ramp function appearing across the capacitor can be obtained by using the constant current flowing through the capacitor.

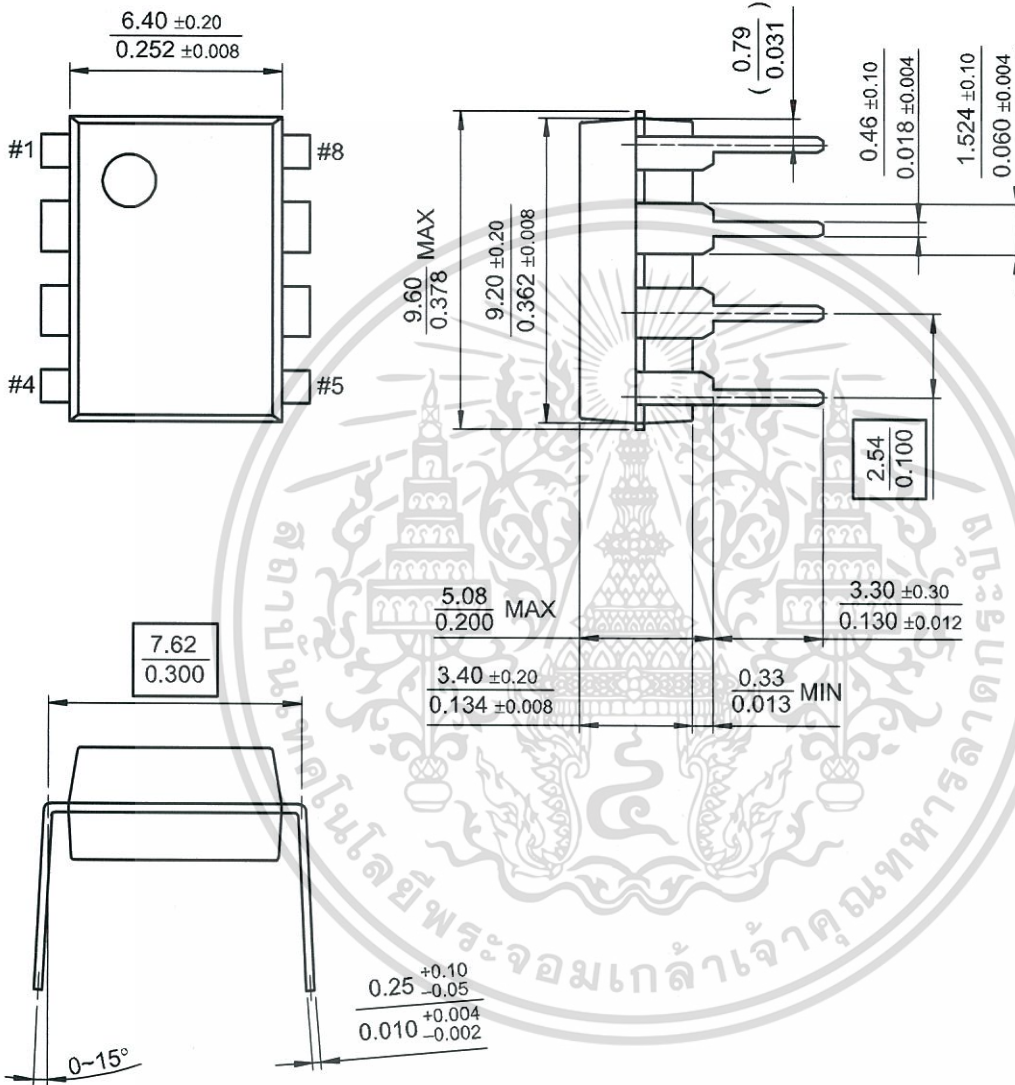
If the constant current flow through the capacitor is 0.215mA and the capacitance is 0.02 $\mu$ F, the gradient of the ramp function at both ends of the capacitor is  $S = 0.215\text{m}/0.022\mu = 9.77\text{V/ms}$ .

## Mechanical Dimensions

## Package

Dimensions in millimeters

## 8-DIP

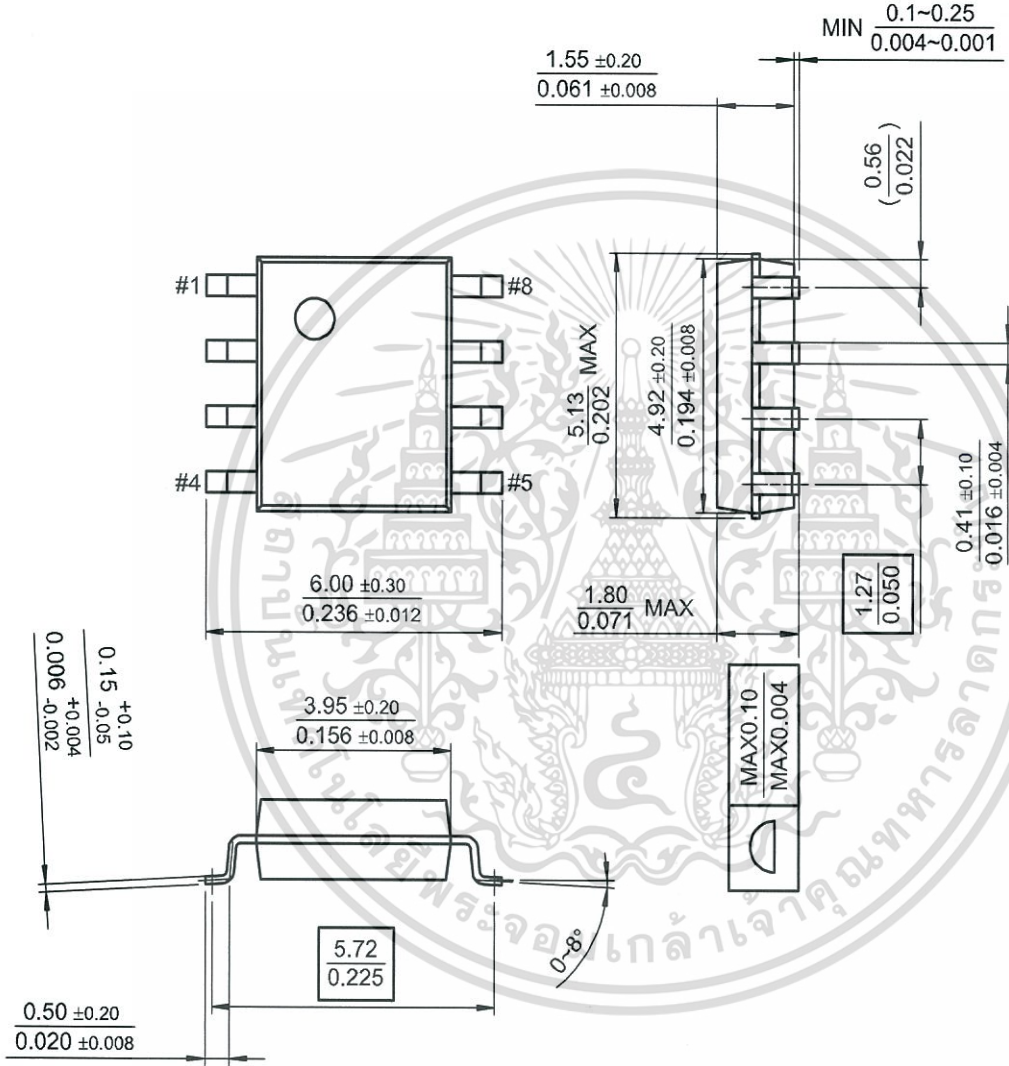


Mechanical Dimensions (Continued)

Package

Dimensions in millimeters

8-SOP



## Ordering Information

Product Number	Package	Operating Temperature
LM555CN	8-DIP	0 ~ +70°C
LM555CM	8-SOP	
Product Number	Package	Operating Temperature
NE555N	8-DIP	0 ~ +70°C
NE555D	8-SOP	
Product Number	Package	Operating Temperature
SA555	8-DIP	-40 ~ +85°C
SA555D	8-SOP	





**DISCLAIMER**

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS.

**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 of 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](http://www.fairchildsemi.com)

11/29/02 0.0m 001  
Stock#DSxxxxxxx

© 2002 Fairchild Semiconductor Corporation

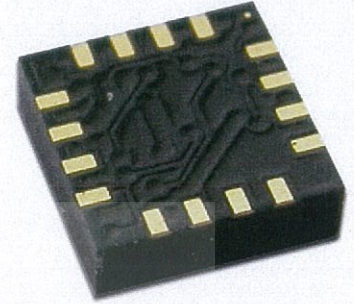
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อการพิมพ์  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# 3-Axis Digital Compass IC HMC5883L

**Honeywell**

Advanced Information

The Honeywell HMC5883L is a surface-mount, multi-chip module designed for low-field magnetic sensing with a digital interface for applications such as low-cost compassing and magnetometry. The HMC5883L includes our state-of-the-art, high-resolution HMC118X series magneto-resistive sensors plus an ASIC containing amplification, automatic degaussing strap drivers, offset cancellation, and a 12-bit ADC that enables 1° to 2° compass heading accuracy. The I<sup>2</sup>C serial bus allows for easy interface. The HMC5883L is a 3.0x3.0x0.9mm surface mount 16-pin leadless chip carrier (LCC). Applications for the HMC5883L include Mobile Phones, Netbooks, Consumer Electronics, Auto Navigation Systems, and Personal Navigation Devices.



The HMC5883L utilizes Honeywell's Anisotropic Magneto-resistive (AMR) technology that provides advantages over other magnetic sensor technologies. These anisotropic, directional sensors feature precision in-axis sensitivity and linearity. These sensors' solid-state construction with very low cross-axis sensitivity is designed to measure both the direction and the magnitude of Earth's magnetic fields, from milli-gauss to 8 gauss. Honeywell's Magnetic Sensors are among the most sensitive and reliable low-field sensors in the industry.

## FEATURES

- ▶ 3-Axis Magneto-resistive Sensors and ASIC in a 3.0x3.0x0.9mm LCC Surface Mount Package
- ▶ 12-Bit ADC Coupled with Low Noise AMR Sensors Achieves 5 milli-gauss Resolution in ±8 Gauss Fields
- ▶ Built-In Self Test
- ▶ Low Voltage Operations (2.16 to 3.6V) and Low Power Consumption (100 µA)
- ▶ Built-In Strap Drive Circuits
- ▶ I<sup>2</sup>C Digital Interface
- ▶ Lead Free Package Construction
- ▶ Wide Magnetic Field Range (+/-8 Oe)
- ▶ Software and Algorithm Support Available
- ▶ Fast 160 Hz Maximum Output Rate

## BENEFITS

- ▶ Small Size for Highly Integrated Products. Just Add a Micro-Controller Interface, Plus Two External SMT Capacitors Designed for High Volume, Cost Sensitive OEM Designs Easy to Assemble & Compatible with High Speed SMT Assembly
- ▶ Enables 1° to 2° Degree Compass Heading Accuracy
- ▶ Enables Low-Cost Functionality Test after Assembly in Production
- ▶ Compatible for Battery Powered Applications
- ▶ Set/Reset and Offset Strap Drivers for Degaussing, Self Test, and Offset Compensation
- ▶ Popular Two-Wire Serial Data Interface for Consumer Electronics
- ▶ RoHS Compliance
- ▶ Sensors Can Be Used in Strong Magnetic Field Environments with a 1° to 2° Degree Compass Heading Accuracy
- ▶ Compassing Heading, Hard Iron, Soft Iron, and Auto Calibration Libraries Available
- ▶ Enables Pedestrian Navigation and LBS Applications

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# HMC5883L

## SPECIFICATIONS (\* Tested at 25°C except stated otherwise.)

Characteristics	Conditions*	Min	Typ	Max	Units
<b>Power Supply</b>					
Supply Voltage	VDD Referenced to AGND	2.16		3.6	Volts
	VDDIO Referenced to DGND	1.71	1.8	VDD+0.1	Volts
Average Current Draw	Idle Mode	-	2	-	µA
	Measurement Mode (7.5 Hz ODR; No measurement average, MA1:MA0 = 00) VDD = 2.5V, VDDIO = 1.8V	-	100	-	µA

## Performance

Field Range	Full scale (FS) – total applied field (Typical)	-8		+8	gauss
Mag Dynamic Range	3-bit gain control	±1		±8	gauss
Resolution	VDD=3.0V, GN=2		5		milli-gauss
Linearity	±2.0 gauss input range			0.1	±% FS
Hysteresis	±2.0 gauss input range		±25		ppm
Cross-Axis Sensitivity	Test Conditions: Cross field = 0.5 gauss, Applied = ±3 gauss		±0.2%		%FS/gauss
Output Rate (ODR)	Continuous Measurement Mode	0.75		75	Hz
	Single Measurement Mode			160	Hz
Measurement Period	From receiving command to data ready		6		msec
Turn-on Time	Ready for I2C commands		200		µs
Gain Tolerance	All gain/dynamic range settings		±5		%
I <sup>2</sup> C Address	7-bit address		0x1E		hex
	8-bit read address		0x3D		hex
	8-bit write address		0x3C		hex
I <sup>2</sup> C Rate	Controlled by I <sup>2</sup> C Master			400	kHz
I <sup>2</sup> C Hysteresis	Hysteresis of Schmitt trigger inputs on SCL and SDA - Fall (VDDIO=1.8V) Rise (VDDIO=1.8V)			0.2*VDDIO	Volts
				0.8*VDDIO	Volts
Self Test	X & Y Axes		±1.16		gauss
	Z Axis		±1.08		
	X & Y Axes (GN=100) Z Axis (GN=100)		510		LSb

## General

ESD Voltage	Human Body Model (all pins) CDM			2000	Volts
				750	
Operating Temperature	Ambient	-30		85	°C
Storage Temperature	Ambient, unbiased	-40		125	°C
Reflow Classification	MSL 3, 260 °C Peak Temperature				
Package Size	Length and Width	2.85	3.00	3.15	mm
Package Height		0.8	0.9	1.0	mm

# HMC5883L

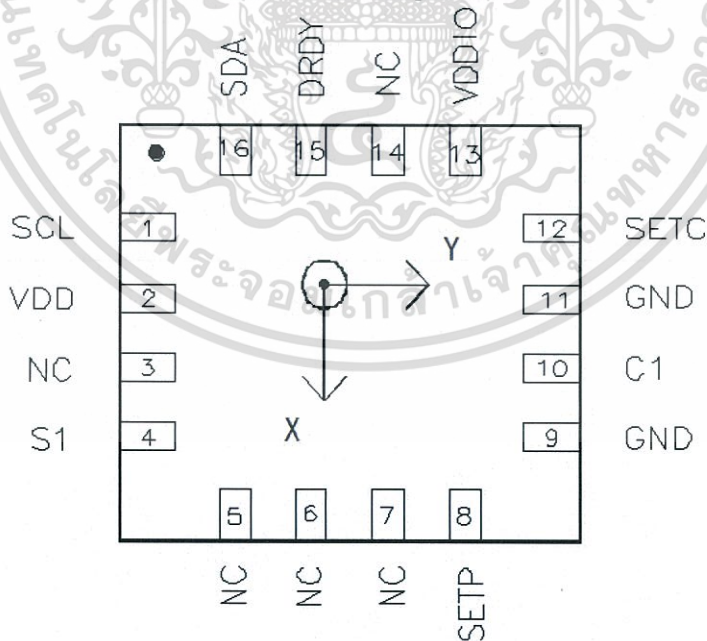
## Absolute Maximum Ratings (\* Tested at 25°C except stated otherwise.)

Characteristics	Min	Max	Units
Supply Voltage VDD	-0.3	4.8	Volts
Supply Voltage VDDIO	-0.3	4.8	Volts

## PIN CONFIGURATIONS

Pin	Name	Description
1	SCL	Serial Clock – I <sup>2</sup> C Master/Slave Clock
2	VDD	Power Supply (2.16V to 3.6V)
3	NC	Not to be Connected
4	S1	Tie to VDDIO
5	NC	Not to be Connected
6	NC	Not to be Connected
7	NC	Not to be Connected
8	SETP	Set/Reset Strap Positive – S/R Capacitor (C2) Connection
9	GND	Supply Ground
10	C1	Reservoir Capacitor (C1) Connection
11	GND	Supply Ground
12	SETC	S/R Capacitor (C2) Connection – Driver Side
13	VDDIO	IO Power Supply (1.71V to VDD)
14	NC	Not to be Connected
15	DRDY	Data Ready, Interrupt Pin. Internally pulled high. Optional connection. Low for 250 usec when data is placed in the data output registers.
16	SDA	Serial Data – I <sup>2</sup> C Master/Slave Data

Table 1: Pin Configurations



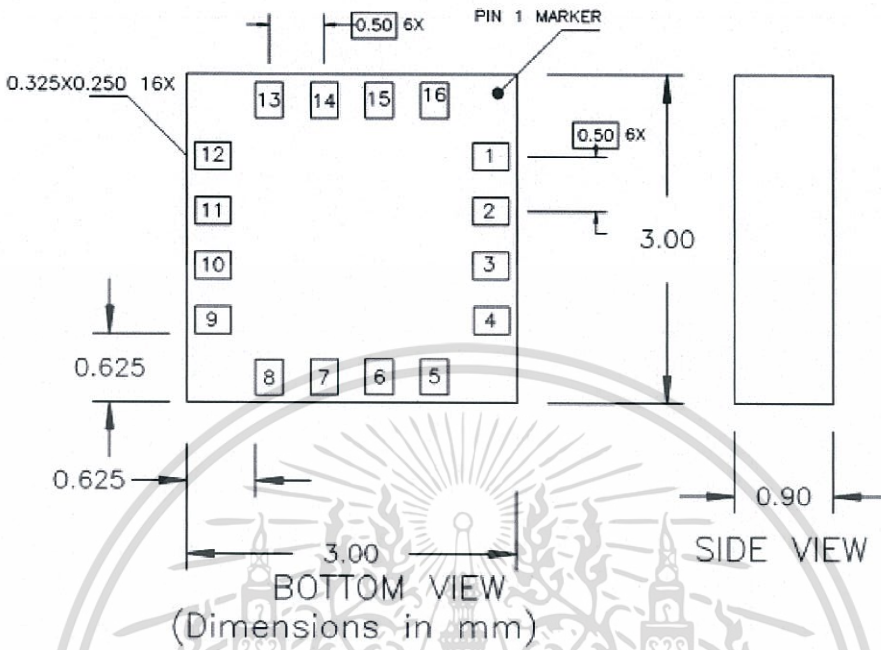
TOP VIEW (looking through)

Arrow indicates direction of magnetic field that generates a positive output reading in Normal Measurement configuration.

# HMC5883L

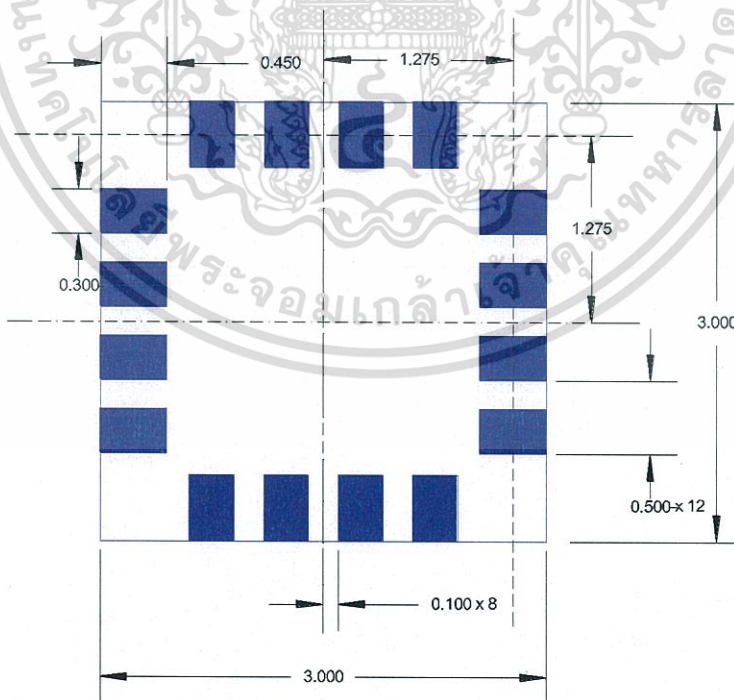
## PACKAGE OUTLINES

### PACKAGE DRAWING HMC5883L (16-PIN LPCC, dimensions in millimeters)



## MOUNTING CONSIDERATIONS

The following is the recommend printed circuit board (PCB) footprint for the HMC5883L.



HMC5883 Land Pad Pattern  
(All dimensions are in mm)

# HMC5883L

## PCB Pad Definition and Traces

The HMC5883L is a fine pitch LCC package. Refer to previous figure for recommended PCB footprint for proper package centering. Size the traces between the HMC5883L and the external capacitors (C1 and C2) to handle the 1 ampere peak current pulses with low voltage drop on the traces.

## Stencil Design and Solder Paste

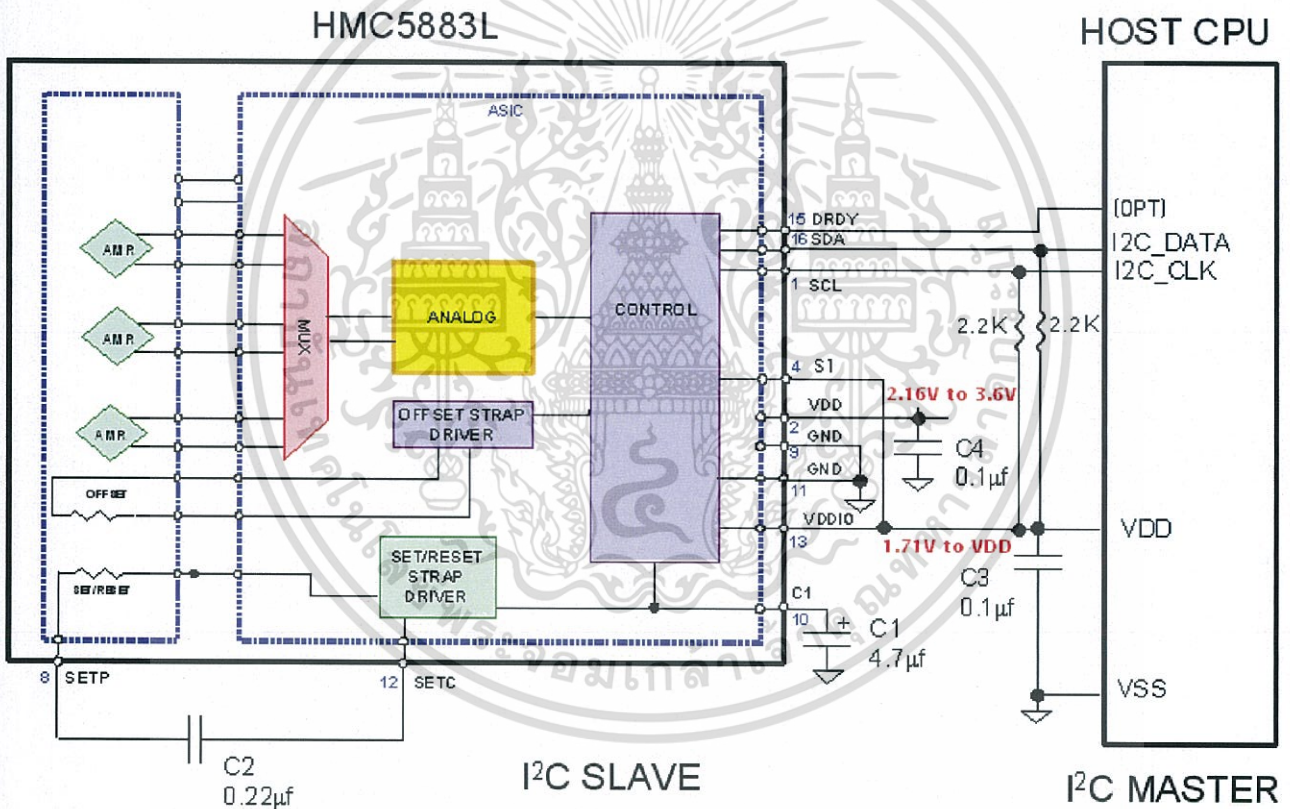
A 4 mil stencil and 100% paste coverage is recommended for the electrical contact pads.

## Reflow Assembly

This device is classified as MSL 3 with 260°C peak reflow temperature. A baking process (125°C, 24 hrs) is required if device is not kept continuously in a dry (< 10% RH) environment before assembly. No special reflow profile is required for HMC5883L, which is compatible with lead eutectic and lead-free solder paste reflow profiles. Honeywell recommends adherence to solder paste manufacturer's guidelines.

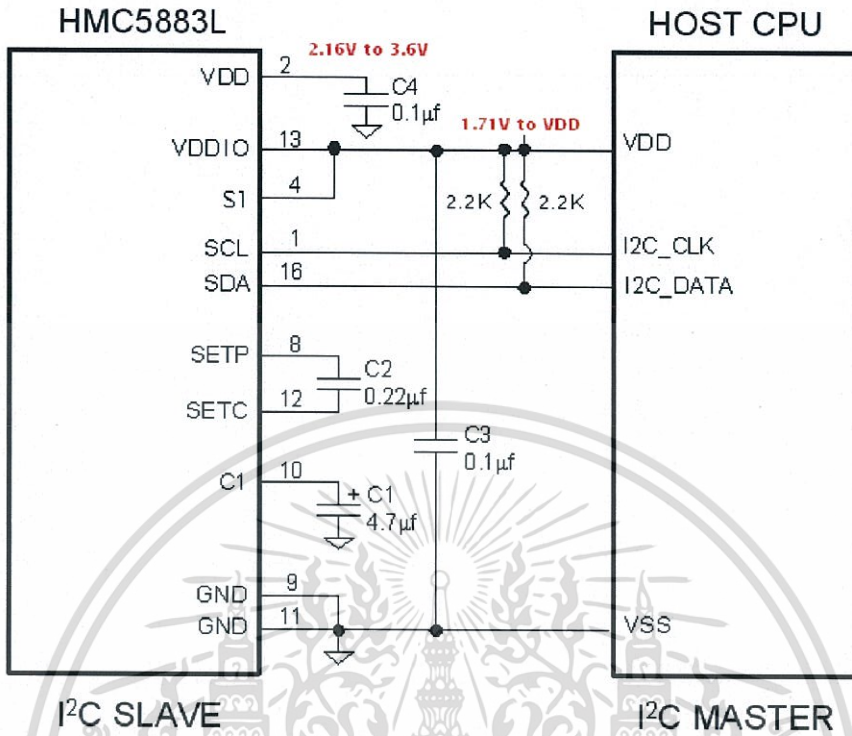
## INTERNAL SCHEMATIC DIAGRAM

### HMC5883L

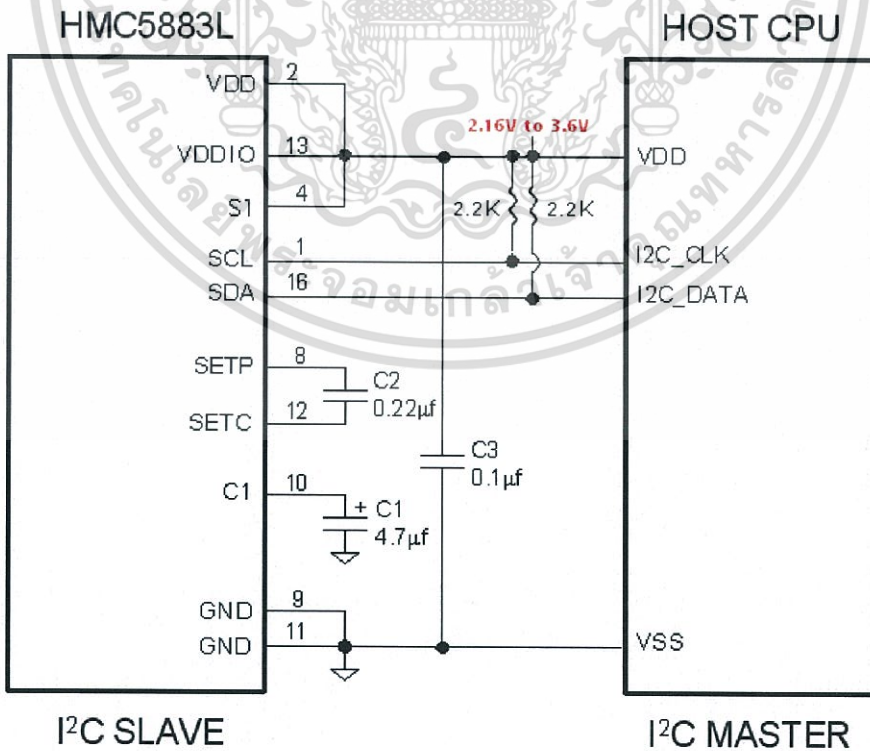


# HMC5883L

## DUAL SUPPLY REFERENCE DESIGN



## SINGLE SUPPLY REFERENCE DESIGN

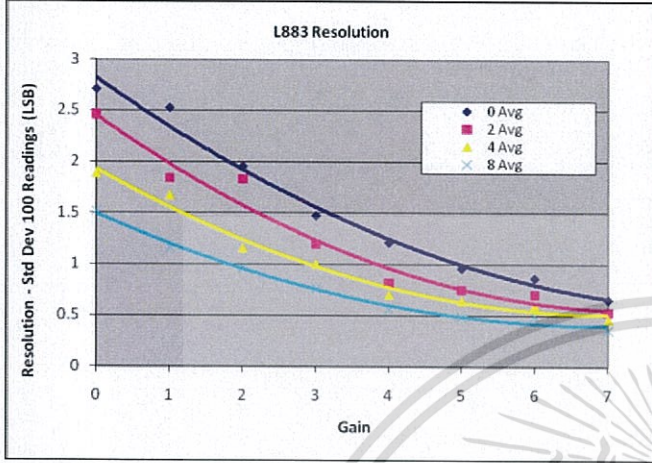


# HMC5883L

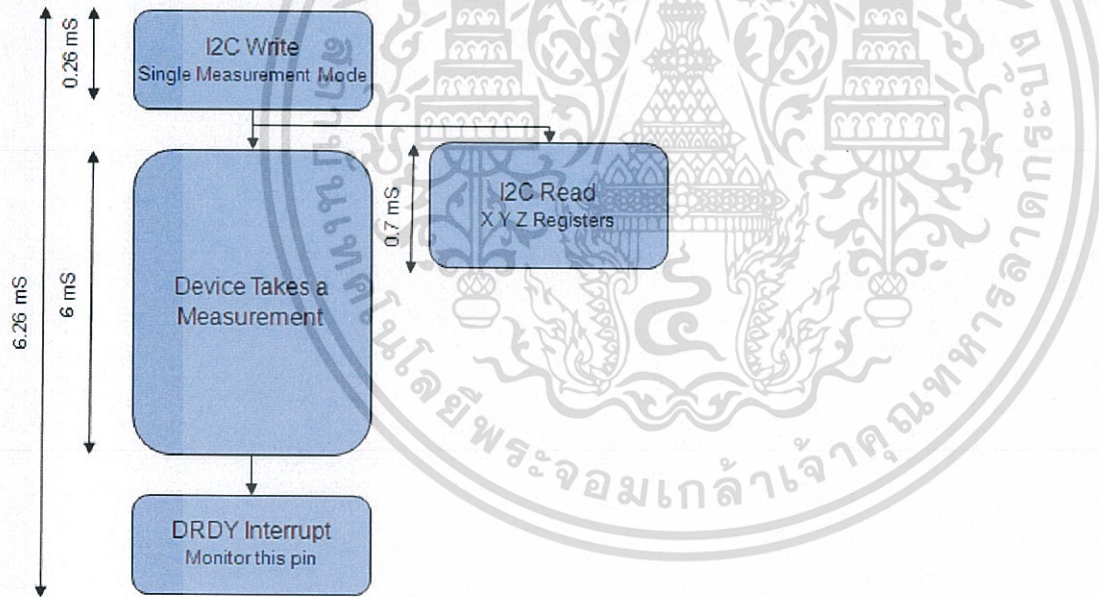
## PERFORMANCE

The following graph(s) highlight HMC5883L's performance.

### Typical Resolution



### Typical Measurement Period in Single-Measurement Mode



\* Monitoring of the DRDY Interrupt pin is only required if maximum output rate is desired.

# HMC5883L

## BASIC DEVICE OPERATION

### Anisotropic Magneto-Resistive Sensors

The Honeywell HMC5883L magnetoresistive sensor circuit is a trio of sensors and application specific support circuits to measure magnetic fields. With power supply applied, the sensor converts any incident magnetic field in the sensitive axis directions to a differential voltage output. The magnetoresistive sensors are made of a nickel-iron (Permalloy) thin-film and patterned as a resistive strip element. In the presence of a magnetic field, a change in the bridge resistive elements causes a corresponding change in voltage across the bridge outputs.

These resistive elements are aligned together to have a common sensitive axis (indicated by arrows in the pinout diagram) that will provide positive voltage change with magnetic fields increasing in the sensitive direction. Because the output is only proportional to the magnetic field component along its axis, additional sensor bridges are placed at orthogonal directions to permit accurate measurement of magnetic field in any orientation.

### Self Test

To check the HMC5883L for proper operation, a self test feature is incorporated in which the sensor is internally excited with a nominal magnetic field (in either positive or negative bias configuration). This field is then measured and reported. This function is enabled and the polarity is set by bits MS[n] in the configuration register A. An internal current source generates DC current (about 10 mA) from the VDD supply. This DC current is applied to the offset straps of the magnetoresistive sensor, which creates an artificial magnetic field bias on the sensor.

See SELF TEST OPERATION section later in this datasheet for additional details.

### Power Management

This device has two different domains of power supply. The first one is VDD that is the power supply for internal operations and the second one is VDDIO that is dedicated to IO interface. It is possible to work with VDDIO equal to VDD; Single Supply mode, or with VDDIO lower than VDD allowing HMC5883L to be compatible with other devices on board.

### I<sup>2</sup>C Interface

Control of this device is carried out via the I<sup>2</sup>C bus. This device will be connected to this bus as a slave device under the control of a master device, such as the processor.

This device is compliant with *I<sup>2</sup>C-Bus Specification*, document number: 9398 393 40011. As an I<sup>2</sup>C compatible device, this device has a 7-bit serial address and supports I<sup>2</sup>C protocols. This device supports standard and fast modes, 100kHz and 400kHz, respectively, but does not support the high speed mode (Hs). External pull-up resistors are required to support these standard and fast speed modes.

Activities required by the master (register read and write) have priority over internal activities, such as the measurement. The purpose of this priority is to not keep the master waiting and the I<sup>2</sup>C bus engaged for longer than necessary.

### Internal Clock

The device has an internal clock for internal digital logic functions and timing management.

### H-Bridge for Set/Reset Strap Drive

The ASIC contains large switching FETs capable of delivering a large but brief pulse to the Set / Reset strap of the sensor. This strap is largely a resistive load. There is no need for an external Set/Reset circuit. The controlling of the Set/Reset function is done automatically by the ASIC for each measurement. One half of the difference from the measurements taken after a set pulse and after a reset pulse will be put in the data output register for each of the three axes. By doing so, the sensor's internal offset and its temperature dependence is removed/cancelled for all measurements.

### Charge Current Limit

The current that reservoir capacitor (C1) can draw when charging is limited for both single supply and dual supply

## HMC5883L

configurations. This prevents drawing down the supply voltage (VDD).

### MODES OF OPERATION

This device has several operating modes whose primary purpose is power management and is controlled by the Mode Register. This section describes these modes.

#### Continuous-Measurement Mode

During continuous-measurement mode, the device continuously makes measurements, at user selectable rate, and places measured data in data output registers. Data can be re-read from the data output registers if necessary; however, if the master does not ensure that the data register is accessed before the completion of the next measurement, the data output registers are updated with the new measurement. To conserve current between measurements, the device is placed in a state similar to idle mode, but the Mode Register is not changed to Idle Mode. That is, MD[n] bits are unchanged. Settings in the Configuration Register A affect the data output rate (bits DO[n]), the measurement configuration (bits MS[n]), when in continuous-measurement mode. All registers maintain values while in continuous-measurement mode. The I<sup>2</sup>C bus is enabled for use by other devices on the network in while continuous-measurement mode.

#### Single-Measurement Mode

This is the default power-up mode. During single-measurement mode, the device makes a single measurement and places the measured data in data output registers. After the measurement is complete and output data registers are updated, the device is placed in idle mode, and the Mode Register is changed to idle mode by setting MD[n] bits. Settings in the configuration register affect the measurement configuration (bits MS[n]) when in single-measurement mode. All registers maintain values while in single-measurement mode. The I<sup>2</sup>C bus is enabled for use by other devices on the network while in single-measurement mode.

#### Idle Mode

During this mode the device is accessible through the I<sup>2</sup>C bus, but major sources of power consumption are disabled, such as, but not limited to, the ADC, the amplifier, and the sensor bias current. All registers maintain values while in idle mode. The I<sup>2</sup>C bus is enabled for use by other devices on the network while in idle mode.

# HMC5883L

## REGISTERS

This device is controlled and configured via a number of on-chip registers, which are described in this section. In the following descriptions, *set* implies a logic 1, and *reset* or *clear* implies a logic 0, unless stated otherwise.

### Register List

The table below lists the registers and their access. All address locations are 8 bits.

Address Location	Name	Access
00	Configuration Register A	Read/Write
01	Configuration Register B	Read/Write
02	Mode Register	Read/Write
03	Data Output X MSB Register	Read
04	Data Output X LSB Register	Read
05	Data Output Z MSB Register	Read
06	Data Output Z LSB Register	Read
07	Data Output Y MSB Register	Read
08	Data Output Y LSB Register	Read
09	Status Register	Read
10	Identification Register A	Read
11	Identification Register B	Read
12	Identification Register C	Read

Table2: Register List

### Register Access

This section describes the process of reading from and writing to this device. The device uses an address pointer to indicate which register location is to be read from or written to. These pointer locations are sent from the master to this slave device and succeed the 7-bit address plus 1 bit read/write identifier.

To minimize the communication between the master and this device, the address pointer updates automatically without master intervention. This automatic address pointer update has two additional features. First when address 12 or higher is accessed the pointer updates to address 00 and secondly when address 08 is reached, the pointer rolls back to address 03. Logically, the address pointer operation functions as shown below.

*If (address pointer = 08) then address pointer = 03*  
*Else if (address pointer >= 12) then address pointer = 0*  
*Else (address pointer) = (address pointer) + 1*

The address pointer value itself cannot be read via the I<sup>2</sup>C bus. Any attempt to read an invalid address location returns 0's, and any write to an invalid address location or an undefined bit within a valid address location is ignored by this device.

To move the address pointer to a random register location, first issue a "write" to that register location with no data byte following the command. For example, to move the address pointer to register 10, send 0x3C 0x0A.

# HMC5883L

## Configuration Register A

The configuration register is used to configure the device for setting the data output rate and measurement configuration. CRA0 through CRA7 indicate bit locations, with CRA denoting the bits that are in the configuration register. CRA7 denotes the first bit of the data stream. The number in parenthesis indicates the default value of that bit.

CRA7	CRA6	CRA5	CRA4	CRA3	CRA2	CRA1	CRA0
(1)	MA1(1)	MA0(1)	DO2 (1)	DO1 (0)	DO0 (0)	MS1 (0)	MS0 (0)

Table 3: Configuration Register A

Location	Name	Description
CRA7	CRA7	This bit must be cleared for correct operation.
CRA6 to CRA5	MA1 to MA0	Select number of samples averaged (1 to 8) per measurement output. 00 = 1; 01 = 2; 10 = 4; 11 = 8 (Default)
CRA4 to CRA2	DO2 to DO0	Data Output Rate Bits. These bits set the rate at which data is written to all three data output registers.
CRA1 to CRA0	MS1 to MS0	Measurement Configuration Bits. These bits define the measurement flow of the device, specifically whether or not to incorporate an applied bias into the measurement.

Table 4: Configuration Register A Bit Designations

The Table below shows all selectable output rates in continuous measurement mode. All three channels shall be measured within a given output rate. Other output rates with maximum rate of 160 Hz can be achieved by monitoring DRDY interrupt pin in single measurement mode.

DO2	DO1	DO0	Typical Data Output Rate (Hz)
0	0	0	0.75
0	0	1	1.5
0	1	0	3
0	1	1	7.5
1	0	0	15 (Default)
1	0	1	30
1	1	0	75
1	1	1	Not used

Table 5: Data Output Rates

MS1	MS0	Measurement Mode
0	0	Normal measurement configuration (Default). In normal measurement configuration the device follows normal measurement flow. The positive and negative pins of the resistive load are left floating and high impedance.
0	1	Positive bias configuration for X, Y, and Z axes. In this configuration, a positive current is forced across the resistive load for all three axes.
1	0	Negative bias configuration for X, Y and Z axes. In this configuration, a negative current is forced across the resistive load for all three axes..
1	1	This configuration is reserved.

Table 6: Measurement Modes

# HMC5883L

## Configuration Register B

The configuration register B for setting the device gain. CRB0 through CRB7 indicate bit locations, with CRB denoting the bits that are in the configuration register. CRB7 denotes the first bit of the data stream. The number in parenthesis indicates the default value of that bit.

CRB7	CRB6	CRB5	CRB4	CRB3	CRB2	CRB1	CRB0
GN2 (0)	GN1 (0)	GN0 (1)	(0)	(0)	(0)	(0)	(0)

Table 7: Configuration B Register

Location	Name	Description
CRB7 to CRB5	GN2 to GN0	Gain Configuration Bits. These bits configure the gain for the device. The gain configuration is common for all channels.
CRB4 to CRB0	0	These bits must be cleared for correct operation.

Table 8: Configuration Register B Bit Designations

The table below shows nominal gain settings. Use the "Gain" column to convert counts to Gauss. Choose a lower gain value (higher GN#) when total field strength causes overflow in one of the data output registers (saturation).

GN2	GN1	GN0	Recommended Sensor Field Range	Gain (LSB/Gauss)	Output Range
0	0	0	± 0.88 Ga	1370	0xF800–0x07FF (-2048–2047)
0	0	1	± 1.3 Ga	1090 (default)	0xF800–0x07FF (-2048–2047)
0	1	0	± 1.9 Ga	820	0xF800–0x07FF (-2048–2047)
0	1	1	± 2.5 Ga	660	0xF800–0x07FF (-2048–2047)
1	0	0	± 4.0 Ga	440	0xF800–0x07FF (-2048–2047)
1	0	1	± 4.7 Ga	390	0xF800–0x07FF (-2048–2047)
1	1	0	± 5.6 Ga	330	0xF800–0x07FF (-2048–2047)
1	1	1	± 8.1 Ga	230	0xF800–0x07FF (-2048–2047)

Table 9: Gain Settings

# HMC5883L

## Mode Register

The mode register is an 8-bit register from which data can be read or to which data can be written. This register is used to select the operating mode of the device. MR0 through MR7 indicate bit locations, with MR denoting the bits that are in the mode register. MR7 denotes the first bit of the data stream. The number in parenthesis indicates the default value of that bit.

MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0
(1)	(0)	(0)	(0)	(0)	(0)	MD1 (0)	MD0 (1)

Table 10: Mode Register

Location	Name	Description
MR7 to MR2	0	These bits must be cleared for correct operation. Bit MR7 bit is set internally after each single-measurement operation.
MR1 to MR0	MD1 to MD0	Mode Select Bits. These bits select the operation mode of this device.

Table 11: Mode Register Bit Designations

MD1	MD0	Operating Mode
0	0	Continuous-Measurement Mode. In continuous-measurement mode, the device continuously performs measurements and places the result in the data register. RDY goes high when new data is placed in all three registers. After a power-on or a write to the mode or configuration register, the first measurement set is available from all three data output registers after a period of $2/f_{DO}$ and subsequent measurements are available at a frequency of $f_{DO}$ , where $f_{DO}$ is the frequency of data output.
0	1	Single-Measurement Mode (Default). When single-measurement mode is selected, device performs a single measurement, sets RDY high and returned to idle mode. Mode register returns to idle mode bit values. The measurement remains in the data output register and RDY remains high until the data output register is read or another measurement is performed.
1	0	Idle Mode. Device is placed in idle mode.
1	1	Idle Mode. Device is placed in idle mode.

Table 12: Operating Modes

# HMC5883L

## Data Output X Registers A and B

The data output X registers are two 8-bit registers, data output register A and data output register B. These registers store the measurement result from channel X. Data output X register A contains the MSB from the measurement result, and data output X register B contains the LSB from the measurement result. The value stored in these two registers is a 16-bit value in 2's complement form, whose range is 0xF800 to 0x07FF. DXRA0 through DXRA7 and DXRB0 through DXRB7 indicate bit locations, with *DXRA* and *DXRB* denoting the bits that are in the data output X registers. DXRA7 and DXRB7 denote the first bit of the data stream. The number in parenthesis indicates the default value of that bit.

In the event the ADC reading overflows or underflows for the given channel, or if there is a math overflow during the bias measurement, this data register will contain the value -4096. This register value will clear when after the next valid measurement is made.

DXRA7	DXRA6	DXRA5	DXRA4	DXRA3	DXRA2	DXRA1	DXRA0
(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
DXRB7	DXRB6	DXRB5	DXRB4	DXRB3	DXRB2	DXRB1	DXRB0
(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)

Table 13: Data Output X Registers A and B

## Data Output Y Registers A and B

The data output Y registers are two 8-bit registers, data output register A and data output register B. These registers store the measurement result from channel Y. Data output Y register A contains the MSB from the measurement result, and data output Y register B contains the LSB from the measurement result. The value stored in these two registers is a 16-bit value in 2's complement form, whose range is 0xF800 to 0x07FF. DYRA0 through DYRA7 and DYRB0 through DYRB7 indicate bit locations, with *DYRA* and *DYRB* denoting the bits that are in the data output Y registers. DYRA7 and DYRB7 denote the first bit of the data stream. The number in parenthesis indicates the default value of that bit.

In the event the ADC reading overflows or underflows for the given channel, or if there is a math overflow during the bias measurement, this data register will contain the value -4096. This register value will clear when after the next valid measurement is made.

DYRA7	DYRA6	DYRA5	DYRA4	DYRA3	DYRA2	DYRA1	DYRA0
(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
DYRB7	DYRB6	DYRB5	DYRB4	DYRB3	DYRB2	DYRB1	DYRB0
(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)

Table 14: Data Output Y Registers A and B

## Data Output Z Registers A and B

The data output Z registers are two 8-bit registers, data output register A and data output register B. These registers store the measurement result from channel Z. Data output Z register A contains the MSB from the measurement result, and data output Z register B contains the LSB from the measurement result. The value stored in these two registers is a 16-bit value in 2's complement form, whose range is 0xF800 to 0x07FF. DZRA0 through DZRA7 and DZRB0 through DZRB7 indicate bit locations, with *DZRA* and *DZRB* denoting the bits that are in the data output Z registers. DZRA7 and DZRB7 denote the first bit of the data stream. The number in parenthesis indicates the default value of that bit.

In the event the ADC reading overflows or underflows for the given channel, or if there is a math overflow during the bias measurement, this data register will contain the value -4096. This register value will clear when after the next valid measurement is made.

DZRA7	DZRA6	DZRA5	DZRA4	DZRA3	DZRA2	DZRA1	DZRA0
(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
DZRB7	DZRB6	DZRB5	DZRB4	DZRB3	DZRB2	DZRB1	DZRB0
(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)

Table 15: Data Output Z Registers A and B

**Data Output Register Operation**

When one or more of the output registers are read, new data cannot be placed in any of the output data registers until all six data output registers are read. This requirement also impacts DRDY and RDY, which cannot be cleared until new data is placed in all the output registers.

**Status Register**

The status register is an 8-bit read-only register. This register is used to indicate device status. SR0 through SR7 indicate bit locations, with SR denoting the bits that are in the status register. SR7 denotes the first bit of the data stream.

SR7	SR6	SR5	SR4	SR3	SR2	SR1	SR0
(0)	(0)	(0)	(x)	(0)	(0)	LOCK (0)	RDY(0)

Table 16: Status Register

Location	Name	Description
SR7 to SR2	0	These bits are reserved. SR4 may be 0 or 1 depending on internal activity. Disregard activity on this bit.
SR1	LOCK	Data output register lock. This bit is set when this some but not all for of the six data output registers have been read. When this bit is set, the six data output registers are locked and any new data will not be placed in these register until one of three conditions are met: one, all six bytes have been read or the mode changed, two, the mode is changed, or three, the measurement configuration is changed.
SR0	RDY	Ready Bit. Set when data is written to all six data registers. Cleared when device initiates a write to the data output registers and after one or more of the data output registers are written to. When RDY bit is clear it shall remain cleared for a 250 μs. DRDY pin can be used as an alternative to the status register for monitoring the device for measurement data.

Table 17: Status Register Bit Designations

# HMC5883L

## Identification Register A

The identification register A is used to identify the device. IRA0 through IRA7 indicate bit locations, with *IRA* denoting the bits that are in the identification register A. IRA7 denotes the first bit of the data stream. The number in parenthesis indicates the default value of that bit.

The identification value for this device is stored in this register. This is a read-only register. Register values. ASCII value *H*

IRA7	IRA6	IRA5	IRA4	IRA3	IRA2	IRA1	IRA0
0	1	0	0	1	0	0	0

Table 18: Identification Register A Default Values

## Identification Register B

The identification register B is used to identify the device. IRB0 through IRB7 indicate bit locations, with *IRB* denoting the bits that are in the identification register A. IRB7 denotes the first bit of the data stream.

Register values. ASCII value 4

IRB7	IRB6	IRB5	IRB4	IRB3	IRB2	IRB1	IRB0
0	0	1	1	0	1	0	0

Table 19: Identification Register B Default Values

## Identification Register C

The identification register C is used to identify the device. IRC0 through IRC7 indicate bit locations, with *IRC* denoting the bits that are in the identification register A. IRC7 denotes the first bit of the data stream.

Register values. ASCII value 3

IRC7	IRC6	IRC5	IRC4	IRC3	IRC2	IRC1	IRC0
0	0	1	1	0	0	1	1

Table 20: Identification Register C Default Values

## I<sup>2</sup>C COMMUNICATION PROTOCOL

The HMC5883L communicates via a two-wire I<sup>2</sup>C bus system as a slave device. The HMC5883L uses a simple protocol with the interface protocol defined by the I<sup>2</sup>C bus specification, and by this document. The data rate is at the standard-mode 100kbps or 400kbps rates as defined in the I<sup>2</sup>C Bus Specifications. The bus bit format is an 8-bit Data/Address send and a 1-bit acknowledge bit. The format of the data bytes (payload) shall be case sensitive ASCII characters or binary data to the HMC5883L slave, and binary data returned. Negative binary values will be in two's complement form. The default (factory) HMC5883L 8-bit slave address is 0x3C for write operations, or 0x3D for read operations.

The HMC5883L Serial Clock (SCL) and Serial Data (SDA) lines require resistive pull-ups (Rp) between the master device (usually a host microprocessor) and the HMC5883L. Pull-up resistance values of about 2.2K to 10K ohms are recommended with a nominal VDDIO voltage. Other resistor values may be used as defined in the I<sup>2</sup>C Bus Specifications that can be tied to VDDIO.

The SCL and SDA lines in this bus specification may be connected to multiple devices. The bus can be a single master to multiple slaves, or it can be a multiple master configuration. All data transfers are initiated by the master device, which is responsible for generating the clock signal, and the data transfers are 8 bit long. All devices are addressed by I<sup>2</sup>C's

www.honeywell.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## HMC5883L

unique 7-bit address. After each 8-bit transfer, the master device generates a 9<sup>th</sup> clock pulse, and releases the SDA line. The receiving device (addressed slave) will pull the SDA line low to acknowledge (ACK) the successful transfer or leave the SDA high to negative acknowledge (NACK).

Per the I<sup>2</sup>C spec, all transitions in the SDA line must occur when SCL is low. This requirement leads to two unique conditions on the bus associated with the SDA transitions when SCL is high. Master device pulling the SDA line low while the SCL line is high indicates the Start (S) condition, and the Stop (P) condition is when the SDA line is pulled high while the SCL line is high. The I<sup>2</sup>C protocol also allows for the Restart condition in which the master device issues a second start condition without issuing a stop.

All bus transactions begin with the master device issuing the start sequence followed by the slave address byte. The address byte contains the slave address; the upper 7 bits (bits7-1), and the Least Significant bit (LSb). The LSb of the address byte designates if the operation is a read (LSb=1) or a write (LSb=0). At the 9<sup>th</sup> clock pulse, the receiving slave device will issue the ACK (or NACK). Following these bus events, the master will send data bytes for a write operation, or the slave will clock out data with a read operation. All bus transactions are terminated with the master issuing a stop sequence.

I<sup>2</sup>C bus control can be implemented with either hardware logic or in software. Typical hardware designs will release the SDA and SCL lines as appropriate to allow the slave device to manipulate these lines. In a software implementation, care must be taken to perform these tasks in code.

## OPERATIONAL EXAMPLES

The HMC5883L has a fairly quick stabilization time from no voltage to stable and ready for data retrieval. The nominal 6 milli-seconds with the factory default single measurement mode means that the six bytes of magnetic data registers (DXRA, DXRB, DZRA, DZRB, DYRA, and DYRB) are filled with a valid first measurement.

To change the measurement mode to continuous measurement mode, after the power-up time send the three bytes:

0x3C 0x02 0x00

This writes the 00 into the second register or mode register to switch from single to continuous measurement mode setting. With the data rate at the factory default of 15Hz updates, a 67 milli-second typical delay should be allowed by the I<sup>2</sup>C master before querying the HMC5883L data registers for new measurements. To clock out the new data, send:

0x3D, and clock out DXRA, DXRB, DZRA, DZRB, DYRA, and DYRB located in registers 3 through 8. The HMC5883L will automatically re-point back to register 3 for the next 0x3D query. All six data registers must be read properly before new data can be placed in any of these data registers.

## SELF TEST OPERATION

To check the HMC5883L for proper operation, a self test feature is incorporated in which the sensor offset straps are excited to create a nominal field strength (bias field) to be measured. To implement self test, the least significant bits (MS1 and MS0) of configuration register A are changed from 00 to 01 (positive bias) or 10 (negative bias), e.g. 0x11 or 0x12.

Then, by placing the mode register into single-measurement mode (0x01), two data acquisition cycles will be made on each magnetic vector. The first acquisition will be a set pulse followed shortly by measurement data of the external field. The second acquisition will have the offset strap excited (about 10 mA) in the positive bias mode for X, Y, and Z axes to create about a ±1.1 gauss self test field plus the external field. The first acquisition values will be subtracted from the second acquisition, and the net measurement will be placed into the data output registers.

Since self test adds ~1.1 Gauss additional field to the existing field strength, using a reduced gain setting prevents sensor from being saturated and data registers overflowed. For example, if the configuration register B is set to 0x60 (Gain=3), values around +766 LSB (1.16 Ga \* 660 LSB/Ga) will be placed in the X and Y data output registers and around +713 (1.08 Ga \* 660 LSB/Ga) will be placed in Z data output register. To leave the self test mode, change MS1 and MS0 bit of the configuration register A back to 00 (Normal Measurement Mode), e.g. 0x10.

# HMC5883L

## SCALE FACTOR CALIBRATION

Using the self test method described above, the user can scale sensors' sensitivity to match each other. Since placing device in positive bias mode (or alternatively negative bias mode) applies a known artificial field on all three axes, the resulting ADC measurements in data output registers can be used to scale the sensors. For example, if the expected self test value for X-axis is 766 and the actual value is 750 then a scale factor of  $(766/750)$  should be multiplied to all future readings of X-axis. Doing so for all three axes will ensure their sensitivity are well matched,


The built-in self test can also be used to periodically compensate the scaling errors due to temperature variations. A compensation factor can be found by comparing the self test outputs with the ones obtained at a known temperature. For example, if the self test output is 750 at room temperature and 700 at the current temperature then a compensation factor of  $(750/700)$  should be applied to all current magnetic readings. A temperature sensor is not required using this method.

## EXTERNAL CAPACITORS

The two external capacitors should be ceramic type construction with low ESR characteristics. The exact ESR values are not critical but values less than 200 milli-ohms are recommended. Reservoir capacitor C1 is nominally 4.7  $\mu\text{F}$  in capacitance, with the set/reset capacitor C2 nominally 0.22  $\mu\text{F}$  in capacitance. Low ESR characteristics may not be in many small SMT ceramic capacitors (0402), so be prepared to up-size the capacitors to gain Low ESR characteristics.

## ORDERING INFORMATION

Ordering Number	Product
HMC5883L-TR	Tape and Reel 4k pieces/reel



**Caution**  
This part is sensitive to damage by electrostatic discharge. Use ESD precautionary procedures when touching, removing or inserting.

## FIND OUT MORE

For more information on Honeywell's Magnetic Sensors visit us online at [www.honeywell.com/magneticsensors](http://www.honeywell.com/magneticsensors) or contact us at 800-323-8295 (763-954-2474 internationally).

The application circuits herein constitute typical usage and interface of Honeywell product. Honeywell does not warranty or assume liability of customer-designed circuits derived from this description or depiction.

Honeywell reserves the right to make changes to improve reliability, function or design. Honeywell does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others.

U.S. Patents 4,441,072, 4,533,872, 4,569,742, 4,681,812, 4,847,584 and 6,529,114 apply to the technology described

**CAUTION: ESDS CAT. 1B**

Honeywell  
12001 Highway 55  
Plymouth, MN 55441  
Tel: 800-323-8295  
[www.honeywell.com/magneticsensors](http://www.honeywell.com/magneticsensors)

Form # 900405 Rev B  
October 2010  
©2010 Honeywell International Inc.

**Honeywell**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้