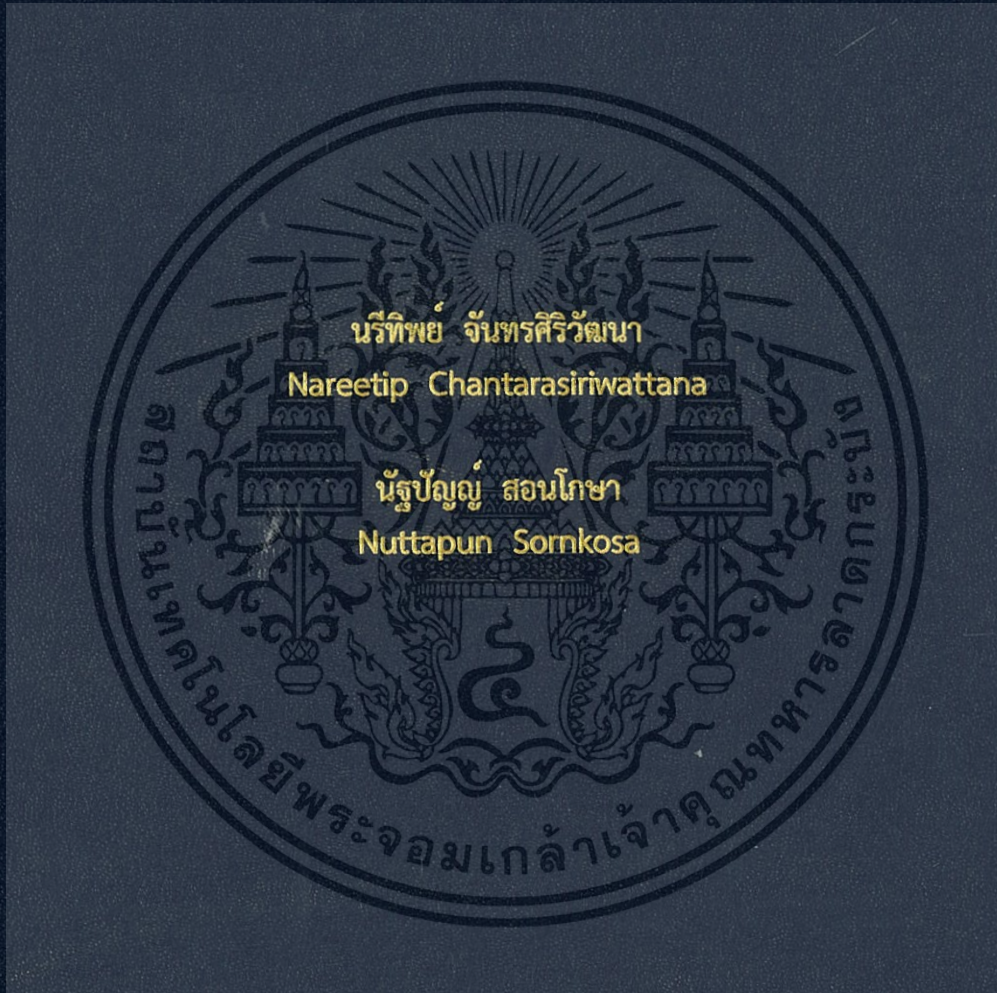


การศึกษาระบบควบคุมไฟจราจรอัจฉริยะแบบตรวจจับยานพาหนะบนท้องถนน
STUDY OF TRAFFIC SIGNAL SYSTEM
USING LOOP DETECTOR FOR VEHICLES



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2558

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การศึกษาระบบควบคุมไฟจราจรอัจฉริยะแบบตรวจจับยานพาหนะบนท้องถนน

STUDY OF TRAFFIC SIGNAL SYSTEM
USING LOOP DETECTOR FOR VEHICLES

โดย



T143916

นริทธิพย์ จันทศิริวัฒนา รหัส 55010629

นัฐปัญญา สอนโกษา รหัส 55010642

อาจารย์ที่ปรึกษา

อาจารย์ชินภัทร นันทจิวงกรชัย



เลขหมู่.....
เลขทะเบียน 143916
วันเดือนปี 04 ต.ค. 2559

12810873
b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2558

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2558

ภาควิชา วิศวกรรมอิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การศึกษาระบบควบคุมไฟจราจรอัจฉริยะแบบตรวจจับยานพาหนะบนท้องถนน

STUDY OF TRAFFIC SIGNAL SYSTEM USING LOOP DETECTOR FOR VEHICLES

ผู้จัดทำ นางสาวนริศทิพย์ จันทศิริวัฒนา รหัส 55010629

นายณัฐปัญญา สอนโกษา รหัส 55010642

ปริญญาานิพนธ์นี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว



(อ.ชินภัทร นันทจิวารชัย)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญาานิพนธ์	การศึกษาระบบสัญญาณไฟ แบบตรวจจับยานพาหนะบนท้องถนน	
นักศึกษา	นางสาวนริทิพย์ จันทศิริวัฒนา	รหัส 55010629
	นายณัฐปัญญา สอนโกษา	รหัส 55010642
ปริญญา	วิศวกรรมศาสตรบัณฑิต	
สาขาวิชา	วิศวกรรมอิเล็กทรอนิกส์	
ปีการศึกษา	2558	
อาจารย์ที่ปรึกษาปริญญาานิพนธ์	อาจารย์ชินภัทร นันทจิวงกรชัย	

บทคัดย่อ

ปริญญาานิพนธ์นี้ศึกษาาระบบสัญญาณไฟ แบบตรวจจับยานพาหนะบนท้องถนน เนื่องจากปัญหาการจราจรติดขัด นับเป็นปัญหาสำคัญระดับชาติที่ปัจจุบันได้มีความรุนแรงเพิ่มมากขึ้น และขยายปัญหาจากพื้นที่ในเขตกรุงเทพมหานครไปยังพื้นที่อื่นๆ ทั่วประเทศ จึงเกิดมีแนวคิดที่จะพัฒนาระบบควบคุมสัญญาณไฟจราจรสำหรับทางแยกเดี่ยวแบบมีการตรวจวัดปริมาณการสัญจรของยานพาหนะ โดยระบบนี้สามารถตรวจจับยานพาหนะบนถนน เพื่อประมวลผลและควบคุมการกำหนดสัญญาณไฟที่เหมาะสมได้ด้วยตนเอง ไม่จำเป็นต้องรอเวลาครบรอบสัญญาณไฟแบบกำหนดเวลาคงที่ เช่น ถ้าไม่มียานพาหนะในแยกที่ตรวจสอบ สัญญาณไฟเขียวจะไม่ถูกปล่อยให้แยกนั้น เป็นต้น

โดยระบบนี้ประกอบไปด้วย ส่วนสำคัญสามส่วน คือ 1. ตัวตรวจด้วยลูป 2. ตัวควบคุมระบบโครงข่ายลูป 3. ตัวควบคุมสัญญาณไฟ ผลจากการทดลองนำไปใช้ในการควบคุมสัญญาณไฟ และเก็บข้อมูลปริมาณยานพาหนะและความยาวแถว เพื่อวัดประสิทธิภาพของระบบ ให้สามารถเพิ่มประสิทธิภาพในการให้สัญญาณไฟจราจรมากที่สุด เมื่อเปรียบเทียบกับระบบควบคุมสัญญาณไฟแบบกำหนดเวลาคงที่

Thesis Title	Study of traffic signal system using loop detector for vehicles.
Student	Ms.Nareetip Chantarasiriwattana Mr.Nuttapun Sornkosa
Degree	Bachelor of Engineering
Program	Electronics Engineering
Year	2015
Thesis Advisor	Mr.Chinnapat Nuntajiwakornchai

Abstract

The traffic congestion is considered a severe national problem. Not only is the difficulty in the Metropolitan area itself, but it has expanded across the nation. Therefore, a traffic detector system was tested. This machine will automatically calculate the amount of vehicles in order to set up the appropriate traffic signal without waiting for a pre-timed traffic signal system. For example, if the traffic detector finds that there is not enough vehicles to be released, it will not show the releasing signal.

This system is composed of 3 parts 1) Inductive Loop Detector 2) Inductive Loop Network System 3) Traffic Lights Controller. When the machine was installed at the junction to gather the information of the amount of vehicles in the queue to evaluate the system, it will have more efficient in releasing vehicles than the pre-timed traffic signal system.

กิตติกรรมประกาศ

โครงการระบบควบคุมไฟจราจรอัจฉริยะแบบตรวจจับยานพาหนะบนท้องถนนนี้ สำเร็จลุล่วงไปได้ด้วยดีโดยได้รับความอนุเคราะห์และความช่วยเหลือของอาจารย์ชินภัทร นันทจิวารัชย์ อาจารย์ที่ปรึกษาโครงการ และจากรุ่นพี่หลายๆคน

คณะผู้วิจัยจึงขอขอบพระคุณอาจารย์ชินภัทร นันทจิวารัชย์ ที่คอยให้ความรู้ คำปรึกษา และคำแนะนำที่เป็นประโยชน์ต่อโครงการนี้ ขอขอบพระคุณรุ่นพี่ทุกคนที่คอยแนะแนวทางในการดำเนินงาน ให้ความช่วยเหลือในการแก้ไขปัญหาที่เกิดขึ้น อีกทั้งยังคอยให้กำลังใจและติดตามเอาใจใส่เป็นอย่างดีมาโดยตลอด ตั้งแต่เริ่มต้นโครงการ จนกระทั่งโครงการนี้สำเร็จไปได้ด้วยดี



นริทธิพย์ จันทศิริวัฒนา

นัฐปัญญา สอนโกษา

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญรูป.....	VI
บทที่ 1 บทนำ	
1.1 ที่มาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของการศึกษา.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 หลักการและทฤษฎีของระบบควบคุมสัญญาณไฟจราจร	
2.1 Loop detector.....	3
2.1.1 วงจรออสซิลเลเตอร์.....	4
2.1.2 วงจรเรโซแนนซ์แบบขนาน.....	5
2.1.3 วงจรเรียงกระแส.....	6
2.1.4 วงจรเปรียบเทียบแรงดัน.....	7
2.2 STM32F429I-DISCOVERY.....	8
2.2.1 คุณสมบัติของบอร์ด.....	8
2.2.2 Key features of the Cortex-M4 core.....	9
บทที่ 3 หลักการพื้นฐานของอุปกรณ์	
3.1 การทำงานของอุปกรณ์.....	10
3.2 Flow Chart.....	11
3.3 Block Diagram.....	12
บทที่ 4 ผลการทดลอง	
4.1 จุดประสงค์การทดลอง.....	13

เอกสาร 4.2 วิธีการทดลองที่ส่งมอบไว้ส่วนแรกจะใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านกา

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา IV จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
4.3 ผลการทดลอง.....	14
4.3.1 ผลการทดลองในส่วนของออสซิลเลเตอร์.....	14
4.3.2 ผลการทดลองในส่วนของขดลวดเหนี่ยวนำ.....	14
4.3.3 ผลการทดลองในส่วนของเรกติไฟเออร์.....	15
4.3.4 ผลการทดลองในส่วนของคอมพาราเตอร์ ขณะที่ไม่พบโลหะ.....	15
4.3.5 ผลการทดลองในส่วนของคอมพาราเตอร์ ขณะที่มีพบโลหะ.....	16
4.3.6 ผลการทดลองตรวจจับยานพาหนะ.....	16
4.3.7 รูปชิ้นงาน.....	17
บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ	
5.1 สรุปผลการทดลอง.....	19
5.2 ข้อเสนอแนะ.....	19
เอกสารอ้างอิง.....	20
ภาคผนวก	
ภาคผนวก ก. โค้ดที่ใช้ในการประมวลผล.....	21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 Loop Detector.....	3
2.2 Schematic.....	3
2.3 Astable Oscillator.....	4
2.4 Series Resonance.....	5
2.5 แผนภาพเฟเซอร์ของวงจร RLC แบบขนาน.....	5
2.6 วงจรเรียงกระแสแบบครึ่งคลื่น.....	6
2.7 วงจรเปรียบเทียบแรงดันที่ $V_{out} = V_{OH}$ เมื่อ $V_{in} > V_{REF}$	7
2.8 วงจรเปรียบเทียบแรงดันที่ $V_{out} = V_{OL}$ เมื่อ $V_{in} > V_{REF}$	7
2.9 STM32F429I-DISCOVERY Board.....	8
3.1 Flow Chart แสดงการทำงานของระบบ.....	11
3.2 Block Diagram แสดงการทำงานของวงจร.....	12
4.1 ผลการทดลองส่วนออสซิลเลเตอร์.....	14
4.2 ผลการทดลองส่วนขดลวดเหนี่ยวนำ.....	14
4.3 ผลการทดลองส่วนเรกติไฟเออร์.....	15
4.4 ผลการทดลองส่วนคอมพาราเตอร์ เมื่อไม่พบโลหะ.....	15
4.5 ผลการทดลองส่วนคอมพาราเตอร์ เมื่อพบโลหะ.....	16
4.6 เมื่อตรวจจับไม่พบยานพาหนะ ไฟแสดงสถานะดับ.....	16
4.7 เมื่อตรวจจับพบยานพาหนะ ไฟแสดงสถานะติด.....	17
4.8 วงจร Loop Detector.....	17
4.9 Inductive Loop.....	18
4.10 โมเดลจำลองเมื่อตรวจจับรถได้ ไฟแสดงสถานะติด ไฟเขียวมีเวลาเพิ่มขึ้น.....	18

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

โครงการฉบับนี้เป็นการจัดทำขึ้นด้วยแนวคิดที่ต้องการจะพัฒนาระบบควบคุมการใช้สัญญาณไฟจราจร เพื่อให้ตอบสนองกับปริมาณการสัญจรของพาหนะในปัจจุบันที่มีปริมาณเพิ่มมากขึ้น และมีปัญหาการจราจรติดขัดที่รุนแรง จึงเกิดเป็นโครงการนี้ขึ้น จากการวัดปริมาณยานพาหนะที่สัญจรบนถนนสายนั้นๆ ด้วยระบบ Loop Detector เพื่อนำข้อมูลที่ได้จาก Loop Detector นี้มาวิเคราะห์และประมวลผลเพื่อควบคุมการกำหนดสัญญาณไฟที่เหมาะสมกับปริมาณยานพาหนะในแต่ละแยกโดยเพิ่มเวลาปล่อยรถในฝั่งที่มีการจราจรติดขัดให้นานขึ้น ส่วนแยกที่มีการจราจรปกติ ก็ให้คงการปล่อยสัญญาณไฟจราจรตามเดิม รวมถึงการบริหารจัดการไฟจราจรในบริเวณที่มีการกั้บรถ เพื่ออำนวยความสะดวกแก่ผู้ใช้ทางกั้บรถ ทำให้สามารถบริหารการจราจรได้อย่างมีประสิทธิภาพมากขึ้น

1.2 วัตถุประสงค์ของโครงการ

ระบบสัญญาณไฟแบบตรวจจับยานพาหนะบนท้องถนนด้วย Loop Detector จะทำงานด้วยการวางอุปกรณ์ตรวจจับยานพาหนะไว้เป็นระยะ เพื่อตรวจสอบความหนาแน่นของยานพาหนะในแต่ละแยก รวมถึงการตรวจจับว่ามียานพาหนะมารออยู่ที่เลนกั้บรถหรือไม่ ในส่วนของทางกั้บรถ โดยอาศัยการเปลี่ยนแปลงค่าความเหนี่ยวนำของสนามแม่เหล็กไฟฟ้า แล้วนำข้อมูลที่ได้มาวิเคราะห์ปริมาณยานพาหนะและประมวลผล เพื่อกำหนดสัญญาณไฟจราจรอย่างเหมาะสมและมีประสิทธิภาพ

1.3 ขอบเขตของโครงการ

1. ศึกษาพฤติกรรมในการยานพาหนะบนท้องถนนในแต่ละแยก เพื่อหาสาเหตุของปัญหาการจราจรที่ติดขัด
2. ศึกษาคอนโทรลเลอร์ที่ใช้ในการควบคุมสัญญาณไฟจราจร ซึ่งได้รับข้อมูลมาจากเซนเซอร์ (Loop Detector) เพื่อนำมาประมวลผล
3. ศึกษาและออกแบบวงจรเพื่อใช้เป็นเซนเซอร์ที่ใช้ในการตรวจจับยานพาหนะ
4. ค้นคว้าหาข้อมูลในการออกแบบขดลวดเหนี่ยวนำสำหรับใช้ในวงจรให้มีค่าที่เหมาะสม

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาปัญหาที่เกิดจากการใช้ระบบสัญญาณไฟจราจรแบบปกติ และหาวิธีแก้ไขปัญหา
2. ศึกษากระบวนการของอุปกรณ์ตรวจจับยานพาหนะบนท้องถนนซึ่งมีส่วนประกอบหลักแบ่งเป็น 3 ส่วนดังนี้
 - 2.1 ศึกษาการทำงานและการประยุกต์ใช้งานของวงจรรอสซิลเลเตอร์

วงจรรอสซิลเลเตอร์ ทำหน้าที่ให้กำเนิดสัญญาณพัลส์ เนื่องจากวงจรรอสซิลเลเตอร์สามารถจ่ายสัญญาณออกมาได้ทั้งๆที่ไม่มีการป้อนอินพุท จึงเป็นวงจรที่มีลักษณะการทำงานแบบไม่เชิงเส้น โดยพื้นฐานจะประกอบด้วยวงจรขยายที่มีการป้อนกลับแบบบวก
 - 2.2 ศึกษาการทำงานและการออกแบบของขดลวดเหนี่ยวนำ

ขดลวดเหนี่ยวนำ ทำหน้าที่เมื่อมีวัตถุที่ทำด้วยโลหะเคลื่อนตัวมาอยู่ใกล้ระยะตรวจจับ โดยทั่วไปจะอยู่ระหว่าง 4 - 40 เมตร จะทำงานโดยการเหนี่ยวนำสนามแม่เหล็ก ซึ่งสายลูปจะถูกฝังและกลบใต้พื้นถนน ลึกประมาณ 3 - 5 ซม. ซึ่งเมื่อมีวัตถุซึ่งประกอบด้วยโลหะอยู่เหนือสายลูปดังกล่าว จะส่งผลให้ศักดาไฟฟ้าที่ตกคร่อมระหว่างลูปดีเทคเตอร์มีปริมาณลดลง ระบบจึงตรวจพบการเปลี่ยนแปลงของ ฟลักซ์แม่เหล็กที่ลดลง
3. ศึกษาการทำงานของวงจรถ่ายแปลงแรงดัน
 - 1.3 ศึกษาการทำงานของวงจรถ่ายแปลงแรงดัน

วงจรถ่ายแปลงแรงดัน ทำหน้าที่เปรียบเทียบแรงดันอินพุทที่ป้อนให้วงจรกับแรงดันอ้างอิงที่ตั้งไว้แล้ว ทำให้เกิดแรงดันเอาต์พุทของวงจรเปลี่ยนแปลงอยู่สองสถานะ คือ สถานะสูง (high) และสถานะต่ำ (low)
3. ศึกษาการใช้งานคอนโทรลเลอร์ เพื่อนำไปออกแบบระบบในส่วนประมวลผล และส่วนควบคุมสัญญาณไฟจราจร

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เพิ่มประสิทธิภาพในการจัดการระบบควบคุมสัญญาณไฟจราจร
2. ลดปริมาณยานพาหนะที่ติดขัดในแต่ละแยก ด้วยการควบคุมการปล่อยสัญญาณไฟจราจรระบายปริมาณยานพาหนะ ในฝั่งที่มีความหนาแน่นให้นานมากขึ้น
3. ช่วยบริหารการจราจรในบริเวณของทางกลับรถ ให้สอดคล้องกับปริมาณยานพาหนะที่มารอ
4. ทำให้มีทักษะการเขียนโปรแกรมบนคอนโทรลเลอร์ (ARM) มากขึ้น

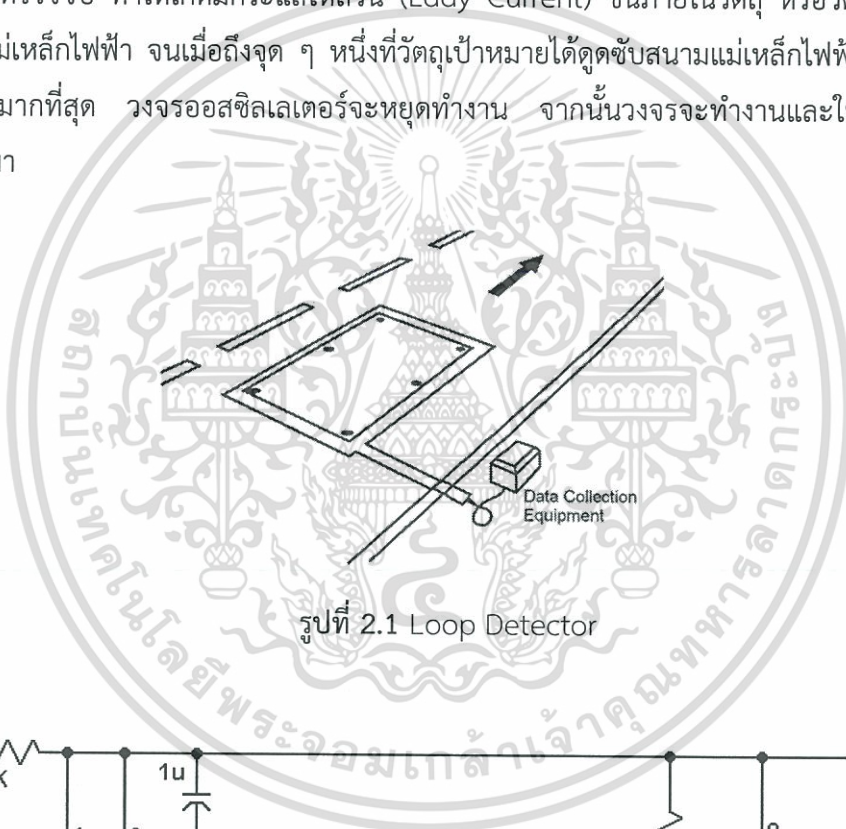
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

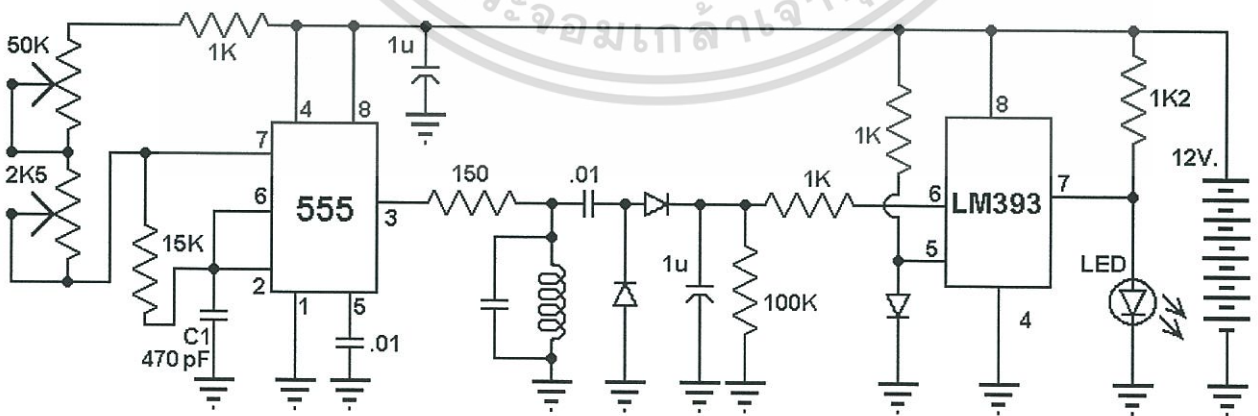
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 Loop Detector

เป็นเซนเซอร์ที่ใช้ตรวจจับวัตถุที่เป็นโลหะเท่านั้น เช่น เหล็ก สแตนเลส เป็นต้น โครงสร้างประกอบด้วย สนามแม่เหล็กไฟฟ้า ขดลวดเหนี่ยวนำ ส่วนวงจรออสซิลเลเตอร์ และส่วนแสดงผล ทำงานโดยอาศัยหลักการเหนี่ยวนำของสนามแม่เหล็กไฟฟ้าที่กำเนิดขึ้นจากวงจรออสซิลเลเตอร์ โดยกำเนิดสัญญาณส่งให้ขดลวดซึ่งฝังอยู่ในพื้น ทำให้เกิดสนามแม่เหล็กไฟฟ้าบริเวณด้านหน้าของอุปกรณ์ เรียกบริเวณนี้ว่า "ส่วนตรวจจับ" เมื่อมีวัตถุเป้าหมายซึ่งต้องเป็นโลหะเท่านั้นเคลื่อนที่เข้ามาบริเวณส่วนตรวจจับ สนามแม่เหล็กไฟฟ้าจะเหนี่ยวนำในวัตถุที่ต้องการตรวจจับ ทำให้เกิดมีกระแสไหลวน (Eddy Current) ขึ้นภายในวัตถุ หรือวัตถุเป้าหมายทำการดูดซับสนามแม่เหล็กไฟฟ้า จนเมื่อถึงจุด ๆ หนึ่งที่วัตถุเป้าหมายได้ดูดซับสนามแม่เหล็กไฟฟ้าจนหมด หรือเกิดการเหนี่ยวนำมากที่สุด วงจรออสซิลเลเตอร์จะหยุดทำงาน จากนั้นวงจรจะทำงานและให้สัญญาณทางด้านเอาต์พุตออกมา



รูปที่ 2.1 Loop Detector



รูปที่ 2.2 Schematic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1 วงจรออสซิลเลเตอร์ (Oscillator)

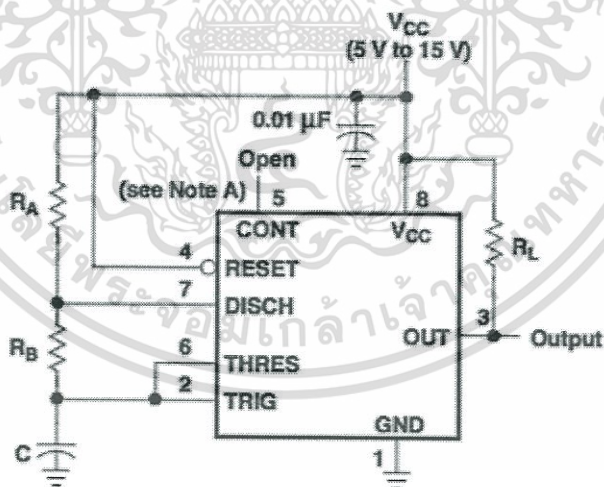
วงจรที่ถูกสร้างขึ้นเพื่อทำหน้าที่ให้กำเนิดสัญญาณพัลส์ เรียกว่า วงจรออสซิลเลเตอร์ เนื่องจากวงจรออสซิลเลเตอร์สามารถจ่ายสัญญาณออกมาได้ทั้งๆที่ไม่มีป้อนอินพุต จึงเป็นวงจรที่มีลักษณะการทำงานแบบไม่เชิงเส้น การศึกษาเกี่ยวกับวงจรออสซิลเลเตอร์จึงไม่ได้อยู่ที่การคำนวณวงจรหาค่าของปริมาณต่างๆเช่นในการศึกษาวิเคราะห์วงจรขยายเชิงเส้น แต่อยู่ที่การวิเคราะห์ห้วงจรหาเงื่อนไขของการออสซิลเลต

วงจรออสซิลเลเตอร์โดยพื้นฐานจะประกอบด้วยวงจรขยายที่มีการป้อนกลับแบบบวก ดังนั้นจึงเป็นวงจรที่ไม่สามารถนำมาใช้เป็นวงจรขยายเชิงเส้นได้ แต่สามารถนำมาใช้เป็นวงจรออสซิลเลเตอร์ได้

ซึ่งอุปกรณ์ที่ใช้ คือ ไอซี 555 มีโหมดการทำงาน 3 โหมด ดังนี้

- โมโนสเตเบิล (Monostable) ในโหมดนี้ การทำงานของ 555 จะเป็นแบบชิ่งเกิ้ลช็อต หรือวันช็อต (one-shot) โดยการสร้างสัญญาณครั้งเดียว ประยุกต์การใช้งานสำหรับการนับเวลา การตรวจสอบพัลส์ สวิตช์สัมผัส ฯลฯ
- ออสเตเบิล (Astable) ในโหมดนี้ การทำงานจะเป็นออสซิลเลเตอร์ การใช้งาน ได้แก่ ทำไฟกะพริบ, กำเนิดพัลส์, กำเนิดเสียง, เตือนภัย ฯลฯ
- ไบสเตเบิล (Bistable) ในโหมดนี้ ไอซี 555 สามารถทำงานเป็นฟลิปฟล็อป (flip-flop) ถ้าไม่ต่อขา DIS และไม่ใช้คาปาซิเตอร์ ใช้เป็นสวิตช์ bouncefree latched switches เป็นต้น

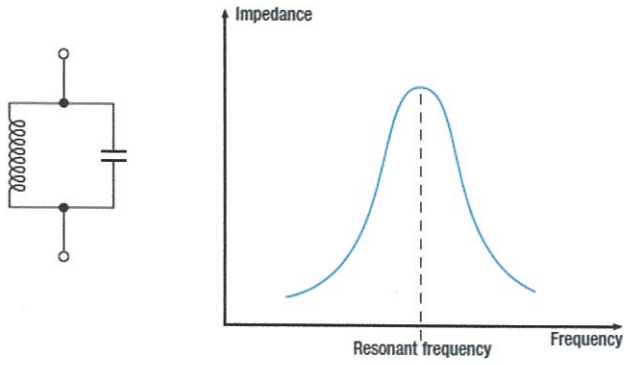
โดยในวงจร Loop detector จะใช้โหมดออสเตเบิล (Astable) เป็นออสซิลเลเตอร์ ในการสร้างสัญญาณพัลส์ให้แก่ส่วนของวงจรรีเซตแชนซ์



รูปที่ 2.3 Astable Oscillator

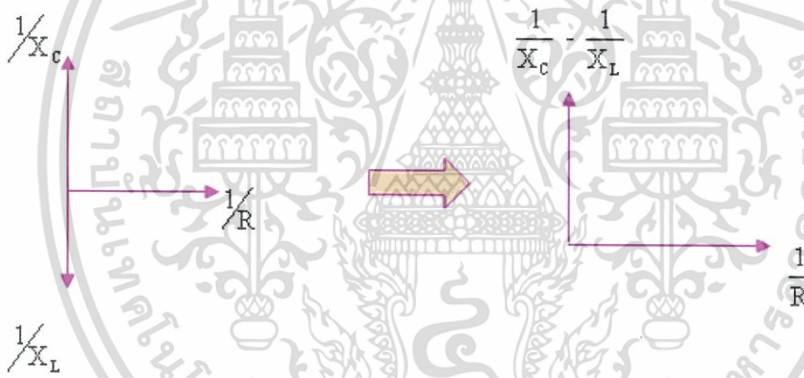
ไอซี 555 เป็นที่นิยมและใช้งานง่ายสำหรับการใช้งานทั่วไป วัตถุประสงค์ในการใช้งาน เริ่มตั้งแต่ 10us - ชั่วโมง หรือจากน้อยกว่า 1mHz - 100kHz ในส่วนของโหมดออสเตเบิล (Astable) ความถี่และอัตราส่วนความกว้างของพัลส์ในวงจรสามารถควบคุมได้อย่างอิสระ โดยมีตัวต้านทานภายนอก 2 ตัว และตัวเก็บประจุภายนอก 1 ตัว เอาท์พุทสูงสุด จะมีค่ามากกว่า Vcc ส่วนกระแสดีซิสชาร์จ จะมีค่าน้อยกว่า Vcc เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 วงจรเรโซแนนซ์แบบขนาน (Series Resonance)



รูปที่ 2.4 Series Resonance

ตัวต้านทาน ขดลวด และตัวเก็บประจุต่อขนานกันแล้วนำไปต่อกับแหล่งจ่ายไฟกระแสสลับ ดังนั้น ความต่างศักย์ไฟฟ้าที่คร่อมอุปกรณ์แต่ละชิ้นจึงมีค่าเท่ากัน แต่เฟสของกระแสไฟฟ้าที่ผ่านตัวต้านทาน ขดลวด และตัวเก็บประจุไม่เหมือนกัน



รูปที่ 2.5 แผนภาพเฟเซอร์ของวงจร RLC แบบขนาน

กระแสที่ไหลผ่านตัวต้านทาน (R) ; $I_R = V/R$ มีเฟสตรงกับ V (2.1)

กระแสที่ไหลผ่านขดลวด (L) ; $I_L = V/X_L$ ตามหลังความต่างศักย์ 90 องศา (2.2)

กระแสที่ไหลผ่านตัวเก็บประจุ (C) ; $I_C = V/X_c$ นำหน้าความต่างศักย์ 90 องศา (2.3)

วงจรจึงมีคุณสมบัติ คือ จะมีค่า Impedance สูงที่ความถี่เรโซแนนซ์ ซึ่งทำให้กระแสไหลได้ต่ำสุด ในกรณีที่ $I_C = I_L$ ค่ากระแสไฟฟ้าในวงจรจะมีค่าเท่ากับ I_R เพราะ I_C มีเฟสต่างจาก I_L อยู่ 180 องศา ทำให้ได้ค่าความถี่ f เรียกว่า “ค่าความถี่เรโซแนนซ์”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่เรโซแนนซ์ ค่าตัวเก็บประจุ และค่าการเหนี่ยวนำ สามารถหาได้จากสูตร

$$f = \frac{1}{2\pi\sqrt{LC}} \quad \text{or} \quad C = \frac{1}{4\pi^2 f^2 L} \quad \text{or} \quad L = \frac{1}{4\pi^2 f^2 C} \quad (2.4)$$

f = ความถี่เรโซแนนซ์

C = ค่าตัวเก็บประจุ

L = ค่าการเหนี่ยวนำ

การคำนวณค่าความเหนี่ยวนำของขดลวด หาได้จากสูตร

$$L = \frac{r^2 N^2}{20r + 28d} \quad (2.5)$$

L = ค่าการเหนี่ยวนำ (μH)

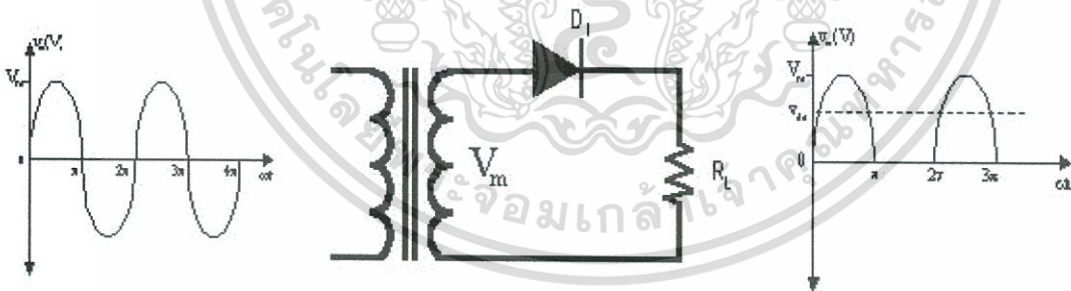
r = รัศมีเฉลี่ยของคอยล์ (cm)

N = จำนวนรอบ

d = ความลึกของคอยล์ (รัศมีนอกลบด้วยรัศมีใน) (cm)

2.1.3 วงจรเรียงกระแส (Rectifier Circuit)

วงจรเรียงกระแสหรือเรียกอีกอย่างหนึ่งว่า เรกติไฟเออร์ คือวงจรไฟฟ้าที่มีคุณสมบัติในการแปลงสัญญาณกระแสสลับให้กลายเป็นสัญญาณไฟฟ้ากระแสตรงหรือมีคุณสมบัติยอมให้ไฟฟ้าไหลผ่านไปในทิศทางใดทิศทางหนึ่ง อุปกรณ์ที่นิยมใช้ในการแปลงสัญญาณได้แก่ ไดโอด



รูปที่ 2.6 วงจรเรียงกระแสแบบครึ่งคลื่น

วงจรเรียงกระแสแบบครึ่งคลื่น (Half Wave Rectifier Circuit)

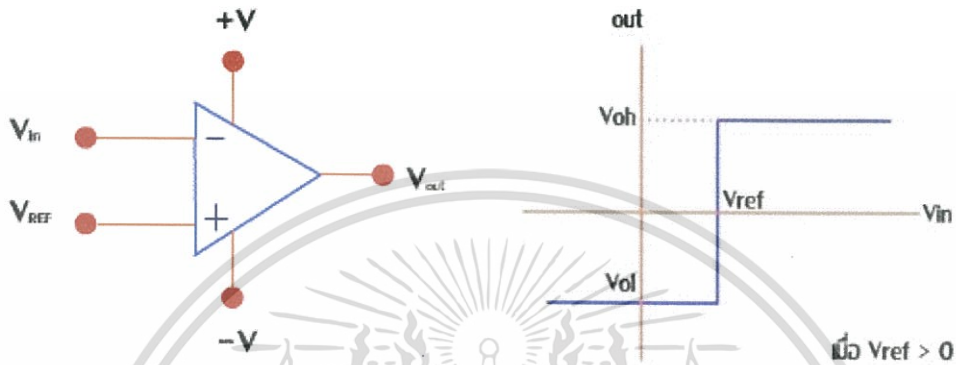
จากรูป 2.4 อธิบายการทำงานของวงจรได้ดังนี้ เมื่อมีสัญญาณครึ่งบวกเข้ามา ไดโอดจะถูกไบอัสตรง ทำให้กระแสไหลในวงจร เกิดแรงดันที่ตกคร่อมตัวต้านทาน R_L ตามลักษณะของสัญญาณอินพุต แต่เมื่อมีสัญญาณครึ่งลบเข้ามาไดโอดจะถูกไบอัสกลับ จึงทำให้ไม่มีกระแสไหลในวงจร แรงดันที่ตกคร่อม R_L จึงมีค่าเป็นศูนย์ดังนั้นสัญญาณเอาต์พุตที่ไหลจึงมีค่าเป็นสัญญาณรูปครึ่งไซเคิล ที่เรียกว่า ฮาล์ฟเวฟ (Half wave)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

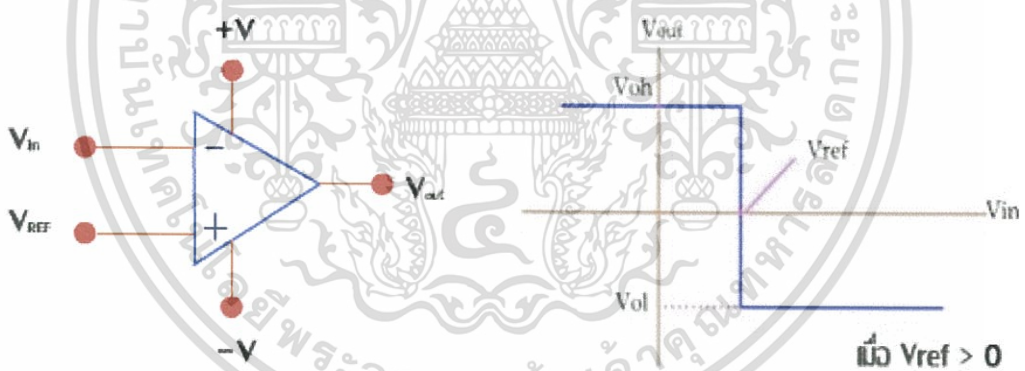
2.1.4 วงจรเปรียบเทียบแรงดัน (Voltage Comparator)

หน้าที่ของวงจรเปรียบเทียบแรงดัน คือ ทำการเปรียบเทียบแรงดันอินพุตที่ป้อนให้วงจรกับแรงดันอ้างอิงที่ตั้งไว้แล้ว ทำให้เกิดแรงดันเอาต์พุตของวงจรเปลี่ยนแปลงอยู่สองสถานะ คือ สถานะสูง (high) และ สถานะต่ำ (low) เท่านั้น

วงจรเปรียบเทียบแรงดันและกราฟคุณสมบัติของวงจรแสดงได้ดังรูปที่ 2.7 และ 2.8 ซึ่งการทำงานของวงจรมีความสัมพันธ์สรุปได้ดังนี้คือ



รูปที่ 2.7 วงจรเปรียบเทียบแรงดันที่ $V_{out} = V_{OH}$ เมื่อ $V_{in} > V_{REF}$



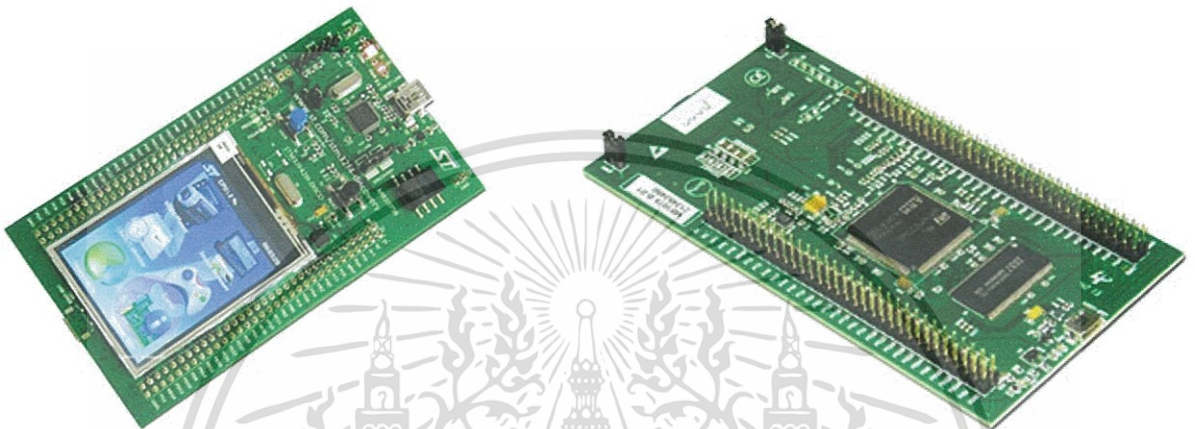
รูปที่ 2.8 วงจรเปรียบเทียบแรงดันที่ $V_{out} = V_{OL}$ เมื่อ $V_{in} > V_{REF}$

เมื่อ V_{OH} และ V_{OL} คือแรงดัน เอาต์พุตอิ่มตัวของออปแอมป์ (OP – AMP saturation voltages) ในสถานะสูงและสถานะต่ำ ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 STM32F429I-DISCOVERY

STM32F429 DISCOVERY บอร์ดไมโครคอนโทรลเลอร์ ARM Cortex-M4 เบอร์ STM32F429ZIT6 ประสิทธิภาพสูง ตัวบอร์ดออกแบบให้ใช้งานได้ง่าย เหมาะสำหรับผู้เริ่มต้นและนักพัฒนา นอกจากนี้ MCU เบอร์ STM32F429ZIT6 บนบอร์ดยังมีส่วน ST-LINK/V2 สำหรับ Download และ Debug โปรแกรมพร้อม หน้าจอ LCD ขนาด 2.4 นิ้ว ความละเอียดระดับ QVGA แรมภายนอกขนาด 64 Mbits เซ็นเซอร์ gyroscope LED และปุ่มกด



รูปที่ 2.9 STM32F429I-DISCOVERY Board

2.2.1 คุณสมบัติของบอร์ด

1. STM32F429ZIT6 32 BIT ARM CORTEX-M4F, 2MB FLASH, 256KB RAM , LQFP144
2. ตัวบอร์ดสามารถใช้ไฟเลี้ยง +5VDC จากขั้วต่อ USB ได้ หรือจาก POWER 5V หรือ 3V ได้จากภายนอก
3. 2.4 “ QVGA TFT LCD พร้อมส่วน TOUCH SCREEN บนบอร์ด
4. SD RAM 64 MBITS
5. SENSOR 3-AXIS DIGITAL OUTPUT เบอร์ L3GD20
6. 6 LED, 2 SW. ใช้งาน และ RESET
7. USB แบบ OTG (ขั้วแบบ MICRO-AB)
8. ขั้วต่อ PIN HEADER ใต้ PCB แบบ 2.54 mm. ตัวผู้ 32 x 2 จำนวน 2 ชุด ต่อใช้งานภายนอกบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 Key features of the Cortex-M4 core

1. ARMv7E-M architecture
2. 3-stage pipeline with branch speculation.
3. 32-bit hardware multiply with 32-bit or 64-bit result, signed or unsigned, add or subtract after the multiply.
4. 32-bit hardware divide (2-12 cycles)
5. Saturated math support.
6. DSP extension: Single cycle 16/32-bit MAC, single cycle dual 16-bit MAC, 8/16-bit SIMD arithmetic.
7. 1 to 240 interrupts, plus NMI.
8. 12 cycle interrupt latency.
9. Integrated sleep modes



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

หลักการพื้นฐานของอุปกรณ์

ระบบควบคุมสัญญาณไฟจราจร ประกอบด้วย ส่วนที่ทำหน้าที่สร้างสัญญาณ (Oscillator) ให้แก่ ส่วนตรวจจับโลหะ (Inductance loop) และส่วนวิเคราะห์ผลที่ได้ (Comparator) เพื่อจะส่งค่าเอาต์พุตที่ได้ ไปยังระบบคอนโทรลเลอร์และประมวลผลควบคุมไฟจราจร

3.1 การทำงานของอุปกรณ์

1. วงจรออสซิลเลเตอร์จะสร้างสัญญาณรูปขายน้ออกมา (AC) เพื่อป้อนให้กับส่วนตรวจจับโลหะ
2. ส่วนตรวจจับโลหะ เมื่อได้รับกระแสไฟฟ้า จะสร้างสนามแม่เหล็กไฟฟ้าออกมา และเมื่อมีโลหะ (ยานพาหนะ) เข้ามาใกล้อยู่ในบริเวณนั้น ฟลักซ์แม่เหล็กจะถูกเหนี่ยวนำ ทำให้ศักดาไฟฟ้าที่ตกคร่อมส่วนตรวจจับโลหะมีค่าลดลง
3. ส่วนวิเคราะห์ผล เมื่อได้รับสัญญาณจากส่วนตรวจจับโลหะ ที่มีค่าเปลี่ยนแปลงจากเดิม จะนำมาวิเคราะห์ผลเป็นค่า High หรือ Low ตามที่ได้ตั้งค่าอ้างอิงไว้ และมีไฟแสดงสถานะบอกสถานะการทำงาน จาก นั้นจึงส่งผลที่ได้ ไปยังคอนโทรลเลอร์

4. เมื่อคอนโทรลเลอร์ได้รับผลมาจากส่วน Comparator

4.1 ระบบสัญญาณไฟจราจร

ระบบสัญญาณไฟจราจร จะตรวจหาค่าที่ได้จากส่วนวิเคราะห์ผล ว่าได้รับเป็นค่า High หรือ Low ก่อนจะปล่อยเป็นสัญญาณไฟเขียวในแต่ละฝั่ง โดยแต่ละฝั่งของแยกจะปล่อยสัญญาณไฟเขียว ฝั่งละเท่าๆกัน

4.1.1 เมื่อได้รับอินพุตเป็น Low (ตรวจไม่พบยานพาหนะ) สัญญาณไฟแยกนั้น จะปล่อยสัญญาณไฟจราจรตามเวลาปกติที่ได้ตั้งค่าไว้

4.1.2 เมื่อได้รับอินพุตเป็น High (ตรวจพบยานพาหนะ) สัญญาณไฟแยกนั้น จะมีการเพิ่มเวลาเฉพาะไฟเขียวให้นานขึ้น

4.2 สัญญาณไฟกลับรถ

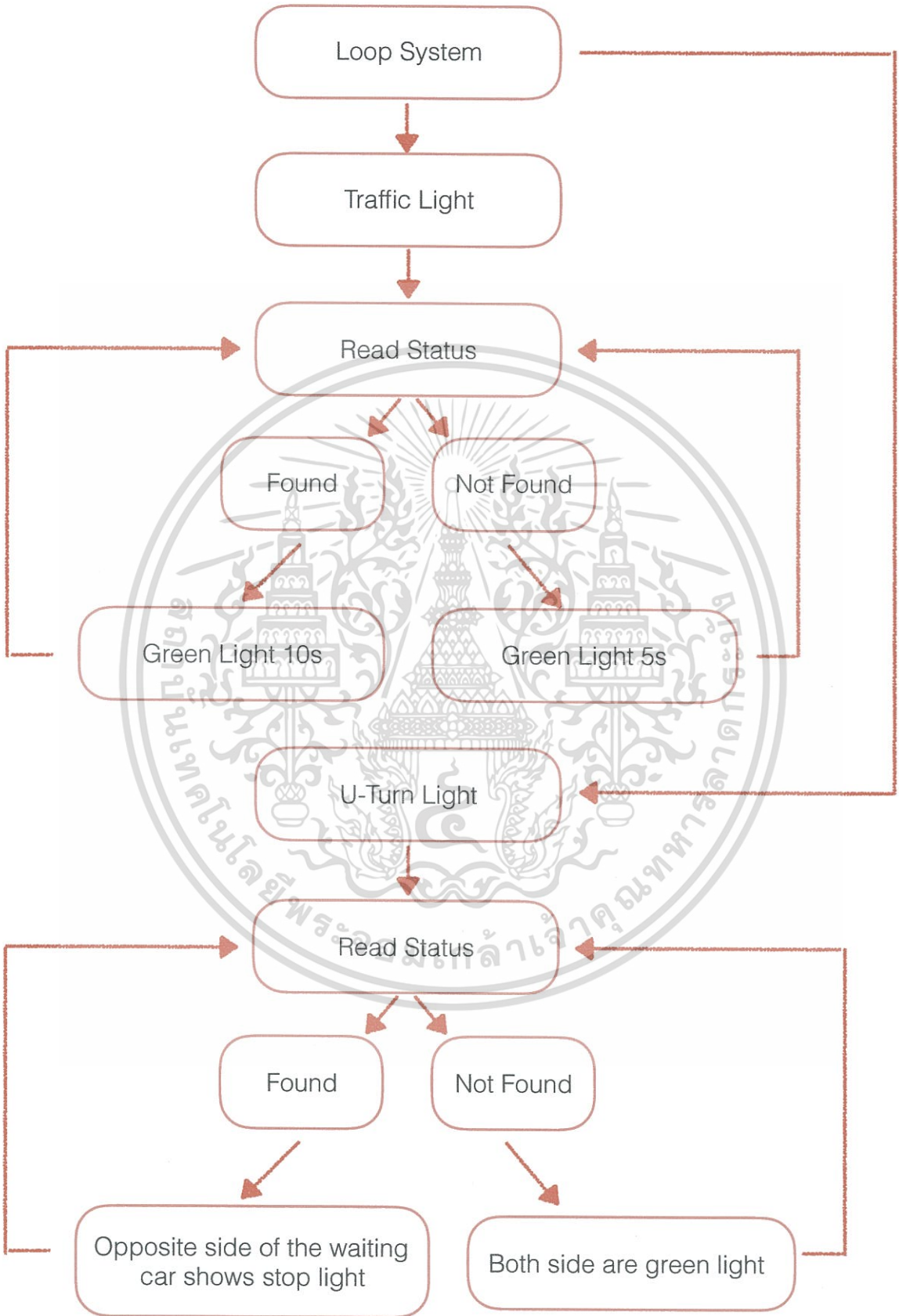
สัญญาณไฟกลับรถ จะมีการตรวจจับหายานพาหนะเป็นระยะ ตามเวลาที่ตั้งไว้

4.2.1 เมื่อได้รับอินพุตเป็น Low (ตรวจไม่พบยานพาหนะ) สัญญาณไฟทั้ง 2 ฝั่ง จะแสดงเป็นไฟเขียวตลอด

4.2.2 เมื่อได้รับอินพุตเป็น High (ตรวจพบยานพาหนะ) สัญญาณไฟเขียวฝั่งตรงข้ามของฝั่งที่ตรวจจับได้ จะกระพริบก่อนที่จะเปลี่ยนจากไฟเขียวเป็นไฟแดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

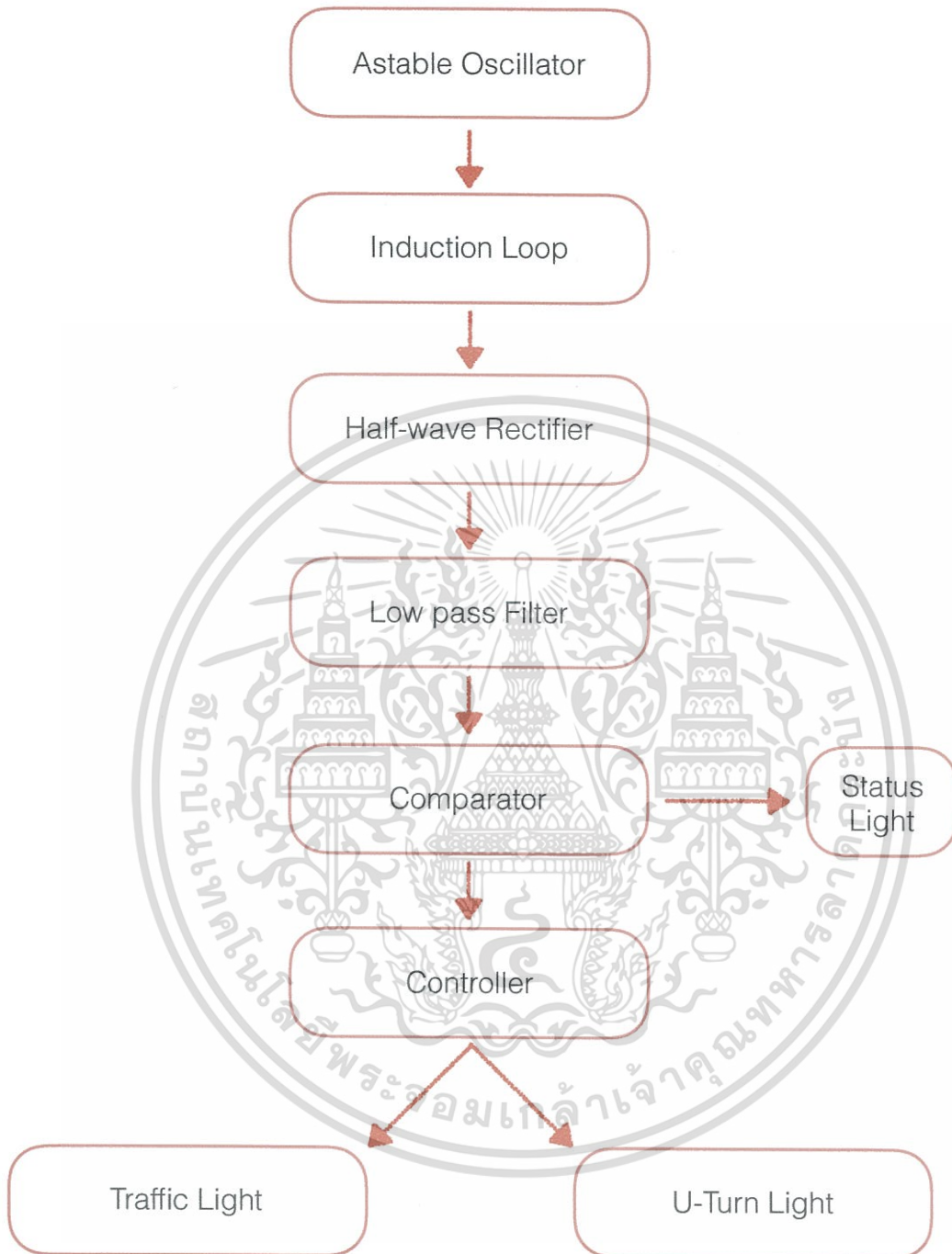
3.2 Flow Chart



รูปที่ 3.1 Flow Chart แสดงการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 Block Diagram



รูปที่ 3.2 Block Diagram แสดงการทำงานของวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 จุดประสงค์การทดลอง

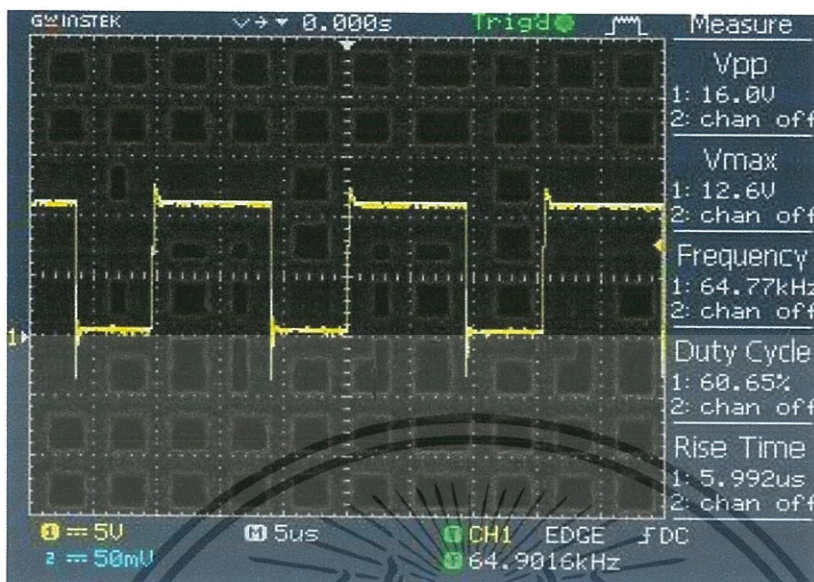
ทำการทดลองเพื่อควบคุมสัญญาณไฟจราจรด้วยคอนโทรลเลอร์ เพื่อนำไปบริหารการจราจรของแต่ละแยก หรือในบริเวณที่มีการกลับรถ โดยมีส่วนเซนเซอร์ตรวจจับโลหะเข้ามาช่วยในการประมวลผลเพื่อการตัดสินใจในการแสดงสัญญาณไฟจราจร ซึ่งเซนเซอร์ที่ใช้ในการตรวจจับโลหะคือ Loop Detector สามารถแบ่งได้เป็น 3 ส่วนหลักๆ คือ วงจรสร้างสัญญาณกระแสสลับ ขดลวดเหนี่ยวนำ วงจรเปรียบเทียบแรงดัน

4.2 วิธีการทดลอง

1. ทำลูปจากสายไฟเพื่อนำไปใช้ทดสอบ
2. นำลูปที่ได้มาเชื่อมต่อกับวงจรส่วนที่เหลือ แล้วนำไปใช้ทดสอบกับบอร์ด
3. แก้ไขความคลาดเคลื่อนที่เกิดจากอุปกรณ์ส่วนของขดลวดเหนี่ยวนำ
4. เขียนโค้ดลงคอนโทรลเลอร์
 - 4.1 ทดลองเซตค่าต่างๆที่ต้องใช้บนคอนโทรลเลอร์
 - 4.2 ทดลองเขียนโค้ดไฟกระพริบ เพื่อหัดใช้ขาเอาต์พุตของบอร์ด
 - 4.3 ทดลองเขียนโค้ดรับค่าอินพุต แล้วสั่งให้ไฟติด
 - 4.4 ออกแบบโลจิกของสัญญาณไฟจราจร และไฟกลับรถ
5. ทำโมเดลเพื่อทดสอบโลจิกที่ได้ออกแบบไว้
6. แก้ไขปัญหาที่เกิดขึ้นจากโค้ด

4.3 ผลการทดลอง

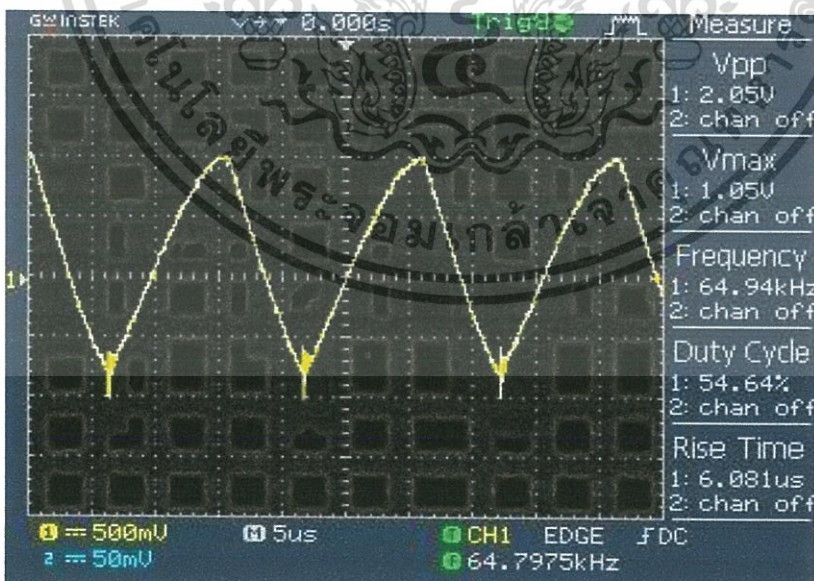
4.3.1 ผลการทดลองในส่วนของออสซิลเลเตอร์



รูปที่ 4.1 ผลการทดลองส่วนออสซิลเลเตอร์

จากการทดลองในส่วนของออสซิลเลเตอร์ จะได้สัญญาณเป็นคลื่นพัลส์ ขนาด 12.6 V และมีความถี่ 64.77 kHz ซึ่งอยู่ในช่วงการทำงานของ IC 555 ที่ 1 mHz - 100 kHz

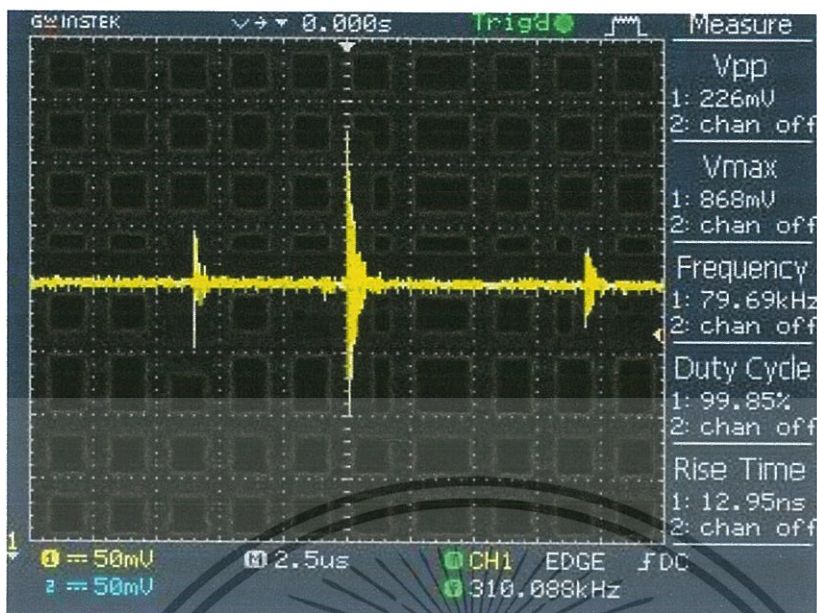
4.3.2 ผลการทดลองในส่วนของขดลวดเหนี่ยวนำ



รูปที่ 4.2 ผลการทดลองส่วนขดลวดเหนี่ยวนำ

จากการทดลอง แสดงให้เห็นกราฟในส่วนของ Inductive loop ซึ่งเมื่อขดลวดตรวจจับโลหะได้ V_{pp} จะมีขนาดลดลงอย่างมาก
 ใบเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะวิธีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 ผลการทดลองในส่วนของเรคตีไฟเออร์



รูปที่ 4.3 ผลการทดลองส่วนเรคตีไฟเออร์

จากการทดลองในส่วนของเรคตีไฟเออร์จะนำสัญญาณที่ได้จากส่วนของขดลวดมาแปลงเป็นไฟกระแสตรง เพื่อนำไปใช้งานในส่วนต่อไป

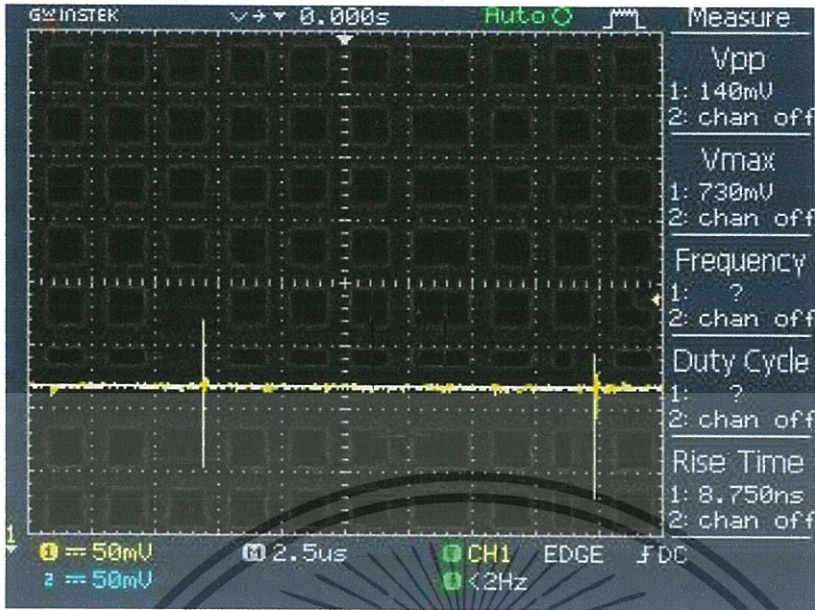
4.3.4 ผลการทดลองในส่วนของคอมพาราเตอร์ ขณะที่ไม่พบโลหะ



รูปที่ 4.4 ผลการทดลองส่วนคอมพาราเตอร์ เมื่อไม่พบโลหะ

จากการทดลองส่วนคอมพาราเตอร์ เมื่อตรวจไม่พบโลหะ พบว่า จะมีโวลต์ตกคร่อมขาลบอยู่ที่ประมาณ 814 mV ออกสารนี้เป็เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

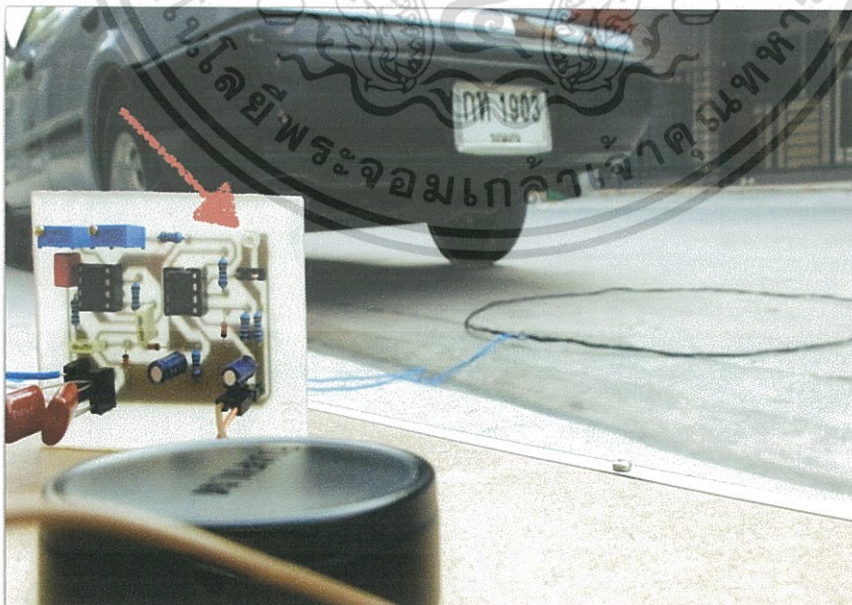
4.3.5 ผลการทดลองในส่วนของคอมพาราเตอร์ ขณะทีพบโลหะ



รูปที่ 4.5 ผลการทดลองส่วนคอมพาราเตอร์ เมื่อพบโลหะ

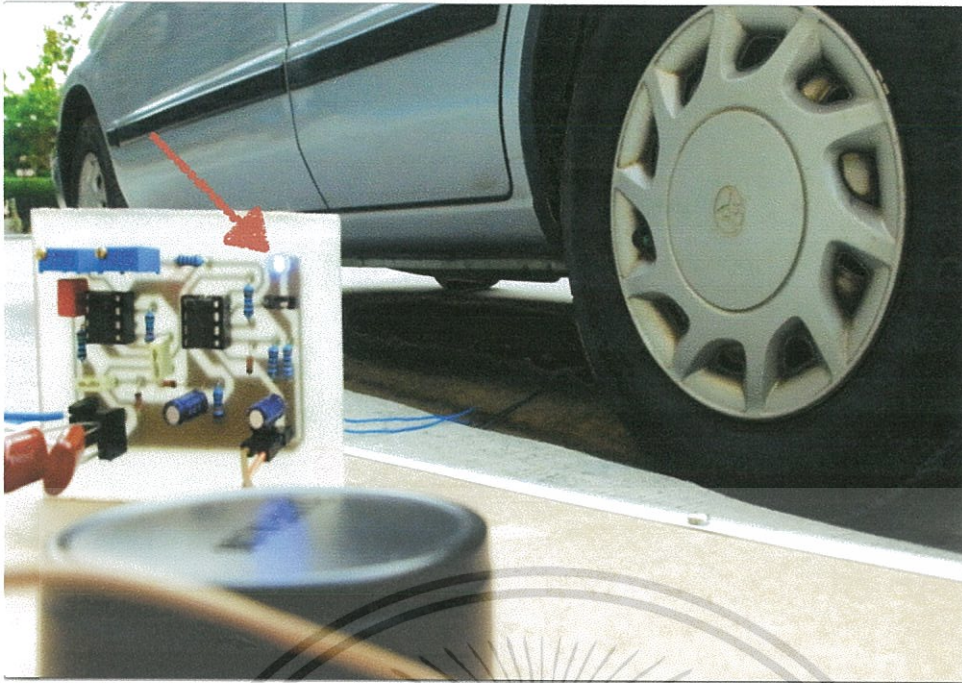
จากการทดลองส่วนคอมพาราเตอร์ เมื่อตรวจพบโลหะ โวลต์ที่ตกคร่อมขาลบของคอมพาราเตอร์จะมีขนาดลดลงเล็กน้อย จากก่อนหน้านี้นี้ที่ไม่ตรวจพบโลหะจะมีโวลต์ตกคร่อมอยู่ 814 mV โดยหลังจากตรวจพบโลหะ ก็จะมีโวลต์ตกคร่อมเหลือ 730 mV

4.3.6 ผลการทดลองตรวจจับยานพาหนะ



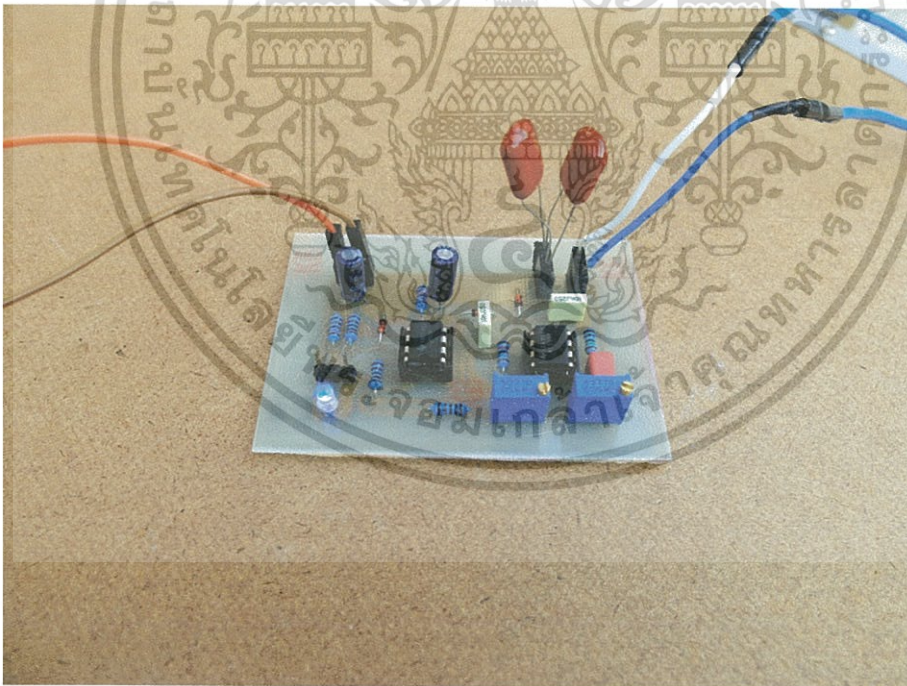
รูปที่ 4.6 เมื่อตรวจจับไม่พบยานพาหนะ ไฟแสดงสถานะดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 เมื่อตรวจจับพบยานพาหนะ ไฟแสดงสถานะติด

4.3.7 รูปชิ้นงาน



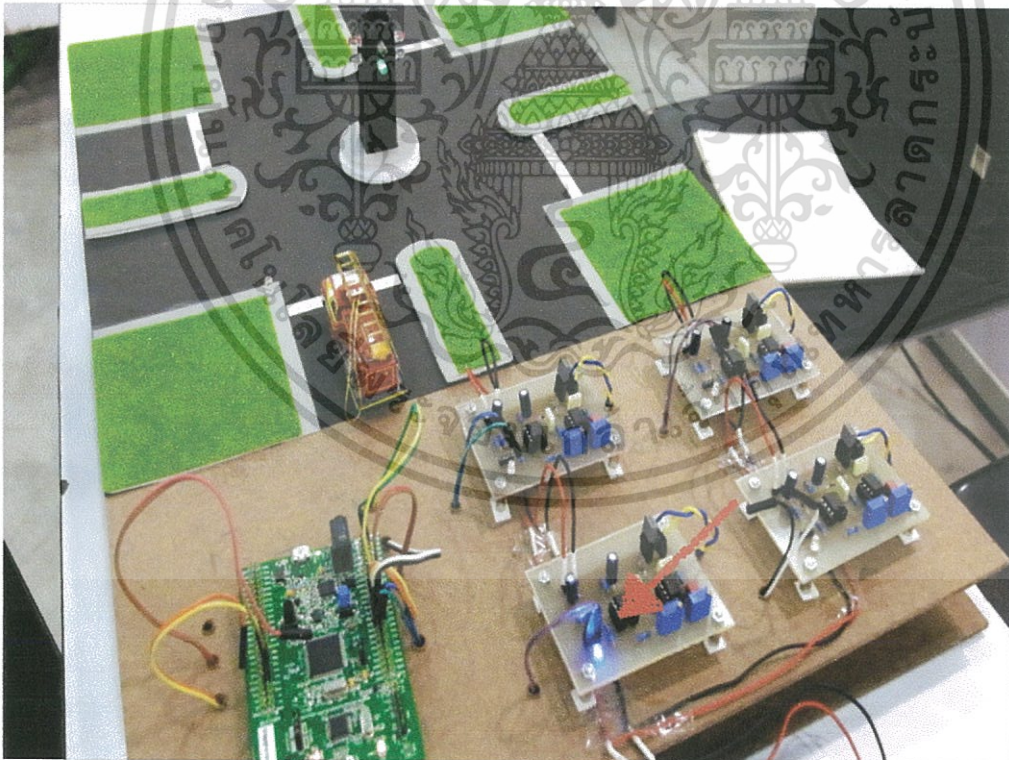
รูปที่ 4.8 วงจร Loop Detector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

143916



รูปที่ 4.9 Inductive Loop



รูปที่ 4.10 โมเดลจำลองเมื่อตรวจจ้บรถได้ ไฟแสดงสถานะติด ไฟเขียวมีเวลาเพิ่มขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

จากการทดลองอุปกรณ์ตรวจจับยานพาหนะด้วยเซนเซอร์ตรวจจับโลหะ (Loop Detector) พบว่า

1. วงจรที่ออกแบบสามารถตรวจจับรถได้จริง โดยเมื่อตรวจจับรถได้ ไฟแสดงสถานะจะติด และไฟจะดับเมื่อรถเคลื่อนที่ผ่านลูบไปแล้ว

2. ระบบสัญญาณไฟจราจรที่ได้ออกแบบโลจิกไว้ ทำงานไปตามคำสั่งที่เขียนไว้ โดยเมื่อมีค่าเอาต์พุตเป็น Low สัญญาณไฟจราจรจะปล่อยรถเป็นเวลาตามปกติ และเมื่อได้รับค่าเอาต์พุตเป็น High คอนโทรลเลอร์จะเพิ่มเวลาในการปล่อยรถให้แก่สัญญาณไฟจราจรในฝั่งนั้นๆ ส่วนสัญญาณไฟสำหรับกลับรถ เมื่อมีค่าเอาต์พุตเป็น Low สัญญาณไฟจะแสดงเป็นไฟเขียวตลอดทั้ง 2 ฝั่ง และเมื่อได้รับค่าเอาต์พุตเป็น High สัญญาณไฟเขียวของฝั่งตรงข้าม จะกระพริบ และเปลี่ยนจากไฟเขียวเป็นไฟแดง

5.2 ข้อเสนอแนะ

1. ค่าของขดลวดเหนี่ยวนำไม่คงที่ เนื่องจากรูปร่างของขดลวดเหนี่ยวนำที่ใช้ในการทดลองไม่คงที่
2. ปัญหาในการปรับค่าของความต้านทานปรับค่าได้ เนื่องจากเมื่อเปิดการทำงานของวงจรใหม่ ต้องมีการปรับค่าใหม่ แก้ไขโดยการเปลี่ยนตัวความต้านทานปรับค่าได้จาก 25 รอบ เป็น 50 รอบ เพื่อเพิ่มความละเอียดในวงจรออสซิลเลเตอร์
3. มีปัญหาในการเชื่อมต่อคอนโทรลเลอร์เข้ากับโปรแกรม และการตั้งค่าอินพุต เอาต์พุต
4. ปัญหาเกิดภายในโมเดล เนื่องจากโมเดลมีขนาดเล็ก ทำให้ขดลวดเหนี่ยวนำแต่ละจุดอยู่ใกล้กัน จึงเกิดการเหนี่ยวนำซึ่งกันและกัน ทำให้ผลที่ได้ไม่ตรงกับความเป็นจริง
5. มีความสับสนในการวัดความต่างศักย์บนอุปกรณ์ในวงจรส่วนของ Loop Detector ทำให้บอร์ด ARM เกิดการลัดวงจร

เอกสารอ้างอิง

- [1] A Simple Vehicle Loop Detector. ค้นเมื่อ 9 กุมภาพันธ์ 2558, จาก <http://chemelec.com/Projects/Loop-Detector-1/Loop-Detector.htm>
- [2] Induction loop. ค้นเมื่อ 8 มีนาคม 2558, จาก https://en.wikipedia.org/wiki/Induction_loop
- [3] STM32F429 Discovery. ค้นเมื่อ 14 มีนาคม 2558, จาก <http://www.thaieasyelec.com/en/home/stm32f429-discovery-detail.html?tmpl=component&flexiblelayout=print>
- [4] ออปแอมป์และวงจรขยายสัญญาณพื้นฐานโดยใช้ออปแอมป์. ค้นเมื่อ 20 มีนาคม 2558, จาก http://living-electronics.blogspot.com/2010/05/blog-post_5115.html
- [5] NE555 Datasheet. ค้นเมื่อ 20 มีนาคม 2558, จาก http://www.doctrionics.co.uk/pdf_files/ne555.pdf
- [6] LM393 Datasheet. ค้นเมื่อ 20 มีนาคม 2558, จาก <http://www.ti.com.cn/cn/lit/ds/symlink/lm2903-n.pdf>
- [7] วงจร RESONANCE. ค้นเมื่อ 25 มีนาคม 2558, จาก http://www.hs8jyx.com/html/resonance_circuit.html
- [8] การต่อวงจรRLCแบบอนุกรมและขนาน. ค้นเมื่อ 25 มีนาคม 2558, จาก <https://sites.google.com/site/physicsbydreammy/kar-tx-wngcrrlc-baeb-xnukrmlaea-khnan>
- [9] Resonance in RLC Series Circuit. ค้นเมื่อ 25 มีนาคม 2558, จาก http://www.il.mahidol.ac.th/e-media/electromagnetism/sub_lesson/11_5.htm
- [10] Fundamentals: Analog Electronic Circuits. ค้นเมื่อ 27 มีนาคม 2558, จาก <https://www.scribd.com/doc/30941292/Engineering-Electronic-Circuits>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include "stm32f4xx_hal.h"
```

```
void SystemClock_Config(void);
```

```
static void MX_GPIO_Init(void);
```

```
int main(void)
```

```
{
```

```
    HAL_Init();
```

```
    SystemClock_Config();
```

```
    MX_GPIO_Init();
```

```
    uint8_t A,B,C,D,P;
```

```
    int i,x;//P>>Program
```

```
    while (1)
```

```
    {
```

```
        P = HAL_GPIO_ReadPin(GPIOC, GPIO_PIN_13);
```

```
        if(P==0)
```

```
        {
```

```
            i=1;
```

```
            if(i==1)//A
```

```
            {
```

```
                A = HAL_GPIO_ReadPin(GPIOE, GPIO_PIN_0);
```

```
                if(A==0)
```

```
                {
```

```
                    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, 1);
```

```
                    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น การใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0|GPIO_PIN_2, 0);

HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 0);

HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7, 0);

for(x=0;x<=9;x++)//5s

    {

        HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_10);//Status

        HAL_Delay(500);

    }

}

if(A==1)
{
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, 1);
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 1);
    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_8|GPIO_PIN_9, 1);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0|GPIO_PIN_2, 0);
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 0);
    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7, 0);

    for(x=0;x<=19;x++)//10s

        {

            HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_10);//Status

            HAL_Delay(500);

        }

}

```

เอกสารนี้เป็นเอกสารที่++ในไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

if(i==2)//B

{

    //B = HAL_GPIO_ReadPin(GPIOB, GPIO_PIN_0);

    B=0;

    if(B==0)

    {

        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, 1);

        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 1);

        HAL_GPIO_WritePin(GPIOE, GPIO_PIN_8| GPIO_PIN_9, 1);

        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1| GPIO_PIN_2, 0);

        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 0);

        HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7, 0);

        for(x=0;x<=9;x++)//5s

        {

            HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_10);//Status

            HAL_Delay(500);

        }

    }

}

if(B==1)

{

    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, 1);

    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 1);

    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_8| GPIO_PIN_9, 1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1| GPIO_PIN_2, 0);

HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 0);

HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7, 0);

for(x=0;x<=19;x++)//10s

{

    HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_10);//Status

    HAL_Delay(500);

}

}

i++;

}

if(i==3)//C

{

    C = HAL_GPIO_ReadPin(GPIOE, GPIO_PIN_6);

    if(C==0)

    {

        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, 1);

        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 1);

        HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7|GPIO_PIN_8, 1);

        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0|GPIO_PIN_2, 0);

        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 0);

        HAL_GPIO_WritePin(GPIOE, GPIO_PIN_9, 0);

        for(x=0;x<=9;x++)//5s

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_10);//Status

        HAL_Delay(500);

    }

}

if(C==1)

{

    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, 1);

    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 1);

    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7|GPIO_PIN_8, 1);

    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0|GPIO_PIN_2, 0);

    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 0);

    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_9, 0);

    for(x=0;x<=19;x++)//10s

    {

        HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_10);//Status

        HAL_Delay(500);

    }

}

i++;

}

if(i==4)//D

{

    //D = HAL_GPIO_ReadPin(GPIOF, GPIO_PIN_15);

```

เอกสารนี้เป็นเอกสารที่ D=0, อนุญาตให้สามารถใช้งานได้ฟรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(D==0)
{
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1|GPIO_PIN_2, 1);

    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 1);

    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_9, 1);

    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, 0);

    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 0);

    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7|GPIO_PIN_8, 0);

    for(x=0;x<=9;x++)//5s
    {
        HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_10);//Status
        HAL_Delay(500);
    }
}
if(D==1)
{
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1|GPIO_PIN_2, 1);

    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 1);

    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_9, 1);

    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, 0);

    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 0);

    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7|GPIO_PIN_8, 0);

    for(x=0;x<=19;x++)//10s

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_10); //Status

        HAL_Delay(500);

    }

}

}

if(P==1)
{

    A = HAL_GPIO_ReadPin(GPIOE, GPIO_PIN_0);

    C = HAL_GPIO_ReadPin(GPIOE, GPIO_PIN_6);

    if(A==0 && C==0)
    {

        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 1);

        HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7, 1);

        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0|GPIO_PIN_1|GPIO_PIN_2, 0);

        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 0);

        HAL_GPIO_WritePin(GPIOE, GPIO_PIN_8|GPIO_PIN_9, 0);

        HAL_Delay(2000);

    }

    if(A==1)

    {

        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 1);

        HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7, 1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ห้ามเผยแพร่หรือแจกจ่ายโดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 0);

HAL_GPIO_WritePin(GPIOE, GPIO_PIN_8|GPIO_PIN_9, 0);

for(x=0;x<=9;x++)//5s

{

    HAL_GPIO_TogglePin(GPIOC, GPIO_PIN_8);//Status

    HAL_Delay(500);

}

HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 1);

HAL_GPIO_WritePin(GPIOE, GPIO_PIN_9, 1);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0|GPIO_PIN_1|GPIO_PIN_2, 0);

HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 0);

HAL_GPIO_WritePin(GPIOE, GPIO_PIN_8|GPIO_PIN_7, 0);

for(x=0;x<=9;x++)//5s

{

    HAL_GPIO_TogglePin(GPIOC, GPIO_PIN_8);//Status

    HAL_Delay(500);

}

}

if(C==1)

{

    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 1);

    HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7, 1);

    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0|GPIO_PIN_1|GPIO_PIN_2, 0);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่หรือแจกจ่ายโดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

HAL_GPIO_WritePin(GPIOE, GPIO_PIN_8|GPIO_PIN_9, 0);

HAL_Delay(5000);

for(x=0;x<=9;x++)//5s

    {

        HAL_GPIO_TogglePin(GPIOC, GPIO_PIN_8);//Status

        HAL_Delay(500);

    }

```

```

HAL_GPIO_WritePin(GPIOC, GPIO_PIN_5, 1);

HAL_GPIO_WritePin(GPIOE, GPIO_PIN_7, 1);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0|GPIO_PIN_1|GPIO_PIN_2, 0);

HAL_GPIO_WritePin(GPIOC, GPIO_PIN_4, 0);

HAL_GPIO_WritePin(GPIOE, GPIO_PIN_8|GPIO_PIN_9, 0);

for(x=0;x<=9;x++)//5s

    {

        HAL_GPIO_TogglePin(GPIOC, GPIO_PIN_8);//Status

        HAL_Delay(500);

    }

```

```

}

```

```

}

```

```

}

```

```

}

```

```

void SystemClock_Config(void)

```

```

{

```

RCC_OscInitTypeDef RCC_OscInitStruct;การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RCC_ClkInitTypeDef RCC_ClkInitStruct;

__PWR_CLK_ENABLE();

__HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE1);

RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSE;

RCC_OscInitStruct.HSEState = RCC_HSE_BYPASS;

RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;

RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;

RCC_OscInitStruct.PLL.PLLM = 4;

RCC_OscInitStruct.PLL.PLLN = 192;

RCC_OscInitStruct.PLL.PLLP = RCC_PLLP_DIV4;

RCC_OscInitStruct.PLL.PLLQ = 8;

HAL_RCC_OscConfig(&RCC_OscInitStruct);

RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_SYSCLK|RCC_CLOCKTYPE_PCLK1;

RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;

RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;

RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV4;

RCC_ClkInitStruct.APB2CLKDivider = RCC_HCLK_DIV1;

HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_3);

HAL_SYSTICK_Config(HAL_RCC_GetHCLKFreq()/1000);

HAL_SYSTICK_CLKSourceConfig(SYSTICK_CLKSOURCE_HCLK);

```

} เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void MX_GPIO_Init(void)
{
    GPIO_InitTypeDef GPIO_InitStructure;

    __GPIOE_CLK_ENABLE();

    __GPIOC_CLK_ENABLE();

    __GPIOH_CLK_ENABLE();

    __GPIOA_CLK_ENABLE();

    __GPIOB_CLK_ENABLE();

    __GPIOD_CLK_ENABLE();

    GPIO_InitStructure.Pin = GPIO_PIN_2|GPIO_PIN_6|GPIO_PIN_0;
    GPIO_InitStructure.Mode = GPIO_MODE_INPUT;
    GPIO_InitStructure.Pull = GPIO_NOPULL;
    HAL_GPIO_Init(GPIOE, &GPIO_InitStructure);

    GPIO_InitStructure.Pin = GPIO_PIN_3|GPIO_PIN_7|GPIO_PIN_8|GPIO_PIN_9|GPIO_PIN_10;
    GPIO_InitStructure.Mode = GPIO_MODE_OUTPUT_PP;
    GPIO_InitStructure.Pull = GPIO_NOPULL;
    GPIO_InitStructure.Speed = GPIO_SPEED_LOW;
    HAL_GPIO_Init(GPIOE, &GPIO_InitStructure);

    GPIO_InitStructure.Pin = GPIO_PIN_4|GPIO_PIN_5|GPIO_PIN_1;
    GPIO_InitStructure.Mode = GPIO_MODE_EVT_RISING;
    เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
    ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
}

```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
HAL_GPIO_Init(GPIOE, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_13;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_0|GPIO_PIN_4|GPIO_PIN_5;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_LOW;
```

```
HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_3;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_AF_PP;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_LOW;
```

```
GPIO_InitStruct.Alternate = GPIO_AF5_SPI2;
```

```
HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_0;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_EVT_RISING;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_4;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_AF_PP;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_LOW;
```

```
GPIO_InitStruct.Alternate = GPIO_AF6_SPI3;
```

```
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_5|GPIO_PIN_6|GPIO_PIN_7;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_AF_PP;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_HIGH;
```

```
GPIO_InitStruct.Alternate = GPIO_AF5_SPI1;
```

```
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_0|GPIO_PIN_1|GPIO_PIN_2;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_LOW;
```

```
HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_10|GPIO_PIN_12;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_AF_PP;
```

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_LOW;
```

```
GPIO_InitStruct.Alternate = GPIO_AF5_SPI2;
```

```
HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_12|GPIO_PIN_13|GPIO_PIN_14|GPIO_PIN_15|GPIO_PIN_4;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_LOW;
```

```
HAL_GPIO_Init(GPIOD, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_7|GPIO_PIN_10|GPIO_PIN_12;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_AF_PP;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_LOW;
```

```
GPIO_InitStruct.Alternate = GPIO_AF6_SPI3;
```

```
HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_9;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
GPIO_InitStruct.Pin = GPIO_PIN_10|GPIO_PIN_11|GPIO_PIN_12;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_AF_PP;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_HIGH;
```

```
GPIO_InitStruct.Alternate = GPIO_AF10_OTG_FS;
```

```
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_5;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
```

```
GPIO_InitStruct.Pull = GPIO_NOPULL;
```

```
HAL_GPIO_Init(GPIOD, &GPIO_InitStruct);
```

```
GPIO_InitStruct.Pin = GPIO_PIN_6|GPIO_PIN_9;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_AF_OD;
```

```
GPIO_InitStruct.Pull = GPIO_PULLUP;
```

```
GPIO_InitStruct.Speed = GPIO_SPEED_LOW;
```

```
GPIO_InitStruct.Alternate = GPIO_AF4_I2C1;
```

```
HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);
```

```
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้