ระบบเตือนทางข้ามรถไฟโดยใช้บอร์ด Raspberry Pi และตัวตรวจรู้ Ultrasonic

โดย

นางสาววิไลลักษณ์	กุลรัตน์	รหัสนักศึกษา	B5515097
นายจักรพงศ์	กาสี	รหัสนักศึกษา	B5525072
นายสุรสิทธิ์	ทุมวรรณ์	รหัสนักศึกษา	B5540891

รายงานนี้เป็นส่วนหนึ่งของการศึกษาวิชา 527499 โครงงานวิศวกรรมโทรคมนาคม

หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม หลักสูตรปรับปรุง พ.ศ. 2554

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

ประจำภาคการศึกษาที่ 3 ปีการศึกษา 2558

ระบบเตือนทางข้ามรถไฟโดยใช้บอร์ด Raspberry Pi และตัวตรวจรู้ Ultrasonic

คณะกรรมการสอบโครงงาน

.....

(รองศาสตราจารย์ ดร.พีระพงษ์ อุฑารสกุล)

กรรมการ

.....

(ผู้ช่วยศาสตราจารย์ ดร.ปิยาภรณ์ มีสวัสดิ์)

กรรมการ

มหาวิทยาลัยเทคโนโลยีสุรนารี

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้ให้นับรายงานโครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมโทรคมนาคม วิชา 527499 โครงงานวิศวกรรมโทรคมนาคม ประจำปีการศึกษา 2558

โครงงาน	ระบบเตือนทางข้ามรถไฟโดยใช้บอร์ด Raspberry Pi และตัวตรวจรู้ Ultrasonic
จัดทำโดย	นางสาววิไลลักษณ์ กุลรัตน์ นายจักรพงศ์ กาสี นายสุรสิทธิ์ ทุมวรรณ์
อาจารย์ที่ปรึกษา	อาจารย์เศรษฐวิทย์ ภูฉายา
สาขาวิชา	วิศวกรรมโทรคมนาคม
ภาคการศึกษาที่	3/ 2558

บทคัดย่อ

ด้วยปัจจุบัน อุบัติเหตุที่เกิดขึ้นกับรถไฟมีความถี่ในการเกิดขึ้นบ่อยครั้ง ส่งผลกระทบต่อประชาชนที่สัญจรผ่านทางรถไฟ โดยเฉพาะจุดลักผ่านหรือทางลักผ่าน ก่อให้เกิดความเสียหายทั้งชีวิตและทรัพย์สิน ด้วยเหตุนี้ทางการรถไฟแห่งประเทศไทยจึงได้จัดสร้างระบบแจ้งเตือนการมาของรถไฟตามบริเวณสำคัญจุดต่างๆ หรือแยกใหญ่ๆตามหัวเมือง แต่ระบบดังกล่าวมีงบประมาณในการก่อสร้างสูง อีกทั้งงบประมาณของรัฐบาลที่มีอยู่อย่างจำกัด จึงทำให้มีบริเวณที่ไม่มีระบบแจ้งเตือนการมาของรถไฟโดยเฉพาะทางลักผ่าน

คณะทำงานได้ทำการสร้างแบบจำลองทางลักผ่าน เพื่อจำลองสถานการณ์ทั้งเมื่อมีรถไฟวิ่งผ่าน แล้วระบบไฟทำการเตือน และเตือนไปยังรถไฟ กรณีมีสิ่งกีดขวางบนรางรถไฟ ในบริเวณทางลักผ่าน จากการทดสอบชิ้นงานบนแบบจำลองทางลักผ่าน จะเห็นได้ว่าอุปกรณ์ทำงานได้ตามจุดประสงค์ที่วางไว้ พร้อมทั้งแจ้งเตือนว่ากำลังมีรถไฟมา และแจ้งเตือนกรณีที่มีรถค้างอยู่บนรางรถไฟ ทำให้สามารถเพิ่มความปลอดภัยบริเวณทางลักผ่านได้

ในการวัดระยะทางด้ายตัวตรวจรู้ Ultrasonic ระยะทางที่วัดได้มีความผิดพลาดจากค่ามาตรฐาน ดังนั้นจะต้องทำการปรับเทียบ (Calibration) เพื่อให้ระยะทางที่วัดด้วยตัวตรวจรู้ Ultrasonic มีค่าใกล้เคียงกับค่ามาตรฐานมากที่สุด

กิตติกรรมประกาศ

จากการที่คณะผู้จัดทำรายงานได้รับมอบหมายให้ทำโครงงานเรื่อง ระบบเตือนภัยทางลักผ่านโดยใช้บอร์ด Raspberry pi และตัวตรวจรู้ Ultrasonic ส่งผลให้คณะผู้จัดทำรายงานได้รับความรู้และประสบการณ์ต่างๆ เกี่ยวกับการเพิ่มความปลอดภัยบริเวณทางลักผ่าน บัดนี้โครงงานดังกล่าวพร้อมทั้งรายงานได้สำเร็จลงแล้ว ทั้งนี้ด้วยความร่วมมือและสนับสนุนจากอาจารย์เศรษฐวิทย์ ภูฉายา อาจารย์ที่ปรึกษาโครงงาน คณาจารย์ บิดา มารดา และนักศึกษาปริญญาตรีสาขาวิศวกรรมโทรคมนาคม มหาวิทยาลัยเทคโนโลยีสุรนารีทุกคน

ข้าพเจ้าใคร่ขอขอบพระคุณผู้ที่มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูลและเป็นที่ปรึกษาในการทำรายงานฉบับนี้จนเสร็จสมบูรณ์ ตลอดจนให้การดูแลและให้ความเข้าใจเกี่ยวกับพื้นฐานการใช้งานโปรแกรม ซึ่งข้าพเจ้าขอขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ด้วย

คณะผู้จัดทำ

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญรูป	จ
สารบัญตาราง	ช
บทที่1 บทนำ	
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตงาน	2
1.4 ขั้นตอนการดำเนินงาน	2
1.5 ผลที่คาดว่าจะได้รับ	2
บทที่2 ทฤษฎีและหลักการที่เกี่ยวข้อง	
2.1 รถไฟ	3
2.1.1 ข้อมูลทางกายภาพ	3
2.1.2 จุดตัดทางรถไฟ	4
2.1.3 กฎหมายที่เกี่ยวข้อง	5
2.2 Raspberry Pi 512MB Model B+	6
2.3 Ultrasonic ranging module HC-SR04	8
2.4 ภาษา Python	11
บทที่3 การออกแบบและการจำลอง	
3.1 บทนำ	14
3.2 การออกแบบและติดตั้งตัวตรวจรู้ Ultrasonic	15
3.2.1 การคำนวณหาระยะทางระหว่างตัวตรวจรู้ Ultrasonic กับทางลักผ่าน	15
3.2.2 การคำนวณหาระยะเวลาที่รถยนต์ค้างอยู่บนรางรถไฟ	16
3.2.3 การคำนวณระยะทางระหว่างตัวตรวจรู้ Ultrasonic กับรางรถไฟ	17

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ (ต่อ)

เรื่อง	**หน้า**
3.3 การใช้งาน Raspberry Pi	18
3.3.1 การติดตั้งระบบปฏิบัติการ Raspbian ให้บอร์ด Raspberry Pi	18
3.3.2 การเชื่อมต่อและแชร์อินเตอร์เน็ตจากคอมพิวเตอร์ให้กับ Raspberry Pi	18
3.3.3 การใช้งาน SSH ด้วยโปรแกรม PuTTY	22
3.3.4 การเขียนโปรแกรมโดยใช้บอร์ด Raspberry pi	26
3.3.5 การเริ่มต้นใช้งานภาษา Python	32
3.4 การใช้งานตัวตรวจรู้วัดระยะ Ultrasonic HC-SR04	32
3.4.1 การแบ่งแรงดันให้ GPIO	33
3.5 บทสรุป	35
บทที่4 การทดสอบชิ้นงาน	
4.1 การปรับเทียบตัวตรวจรู้ Ultrasonic	36
4.2 การทดสอบชิ้นงาน	38
4.3 บทสรุป	41
บทที่5 บทสรุปของโครงงาน	
5.1 ปัญหาและการแก้ไข	42
5.2 ข้อเสนอแนะ	42
5.3 แนวทางในการพัฒนาต่อไป	42
5.4 บทสรุป	42
บรรณานุกรม	44
ภาคผนวก	
Ultrasonic Ranging Module HC – SR04 Specification	46
Raspberry B+ Specification	49
ประวัติผู้เขียน	50

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูปภาพ

รายการ **หน้า**

รูป 2.1 แสดงทางลักผ่าน (Illegal Crossing) 4
รูป 2.2 Raspberry Pi 512MB Model B+ 6
รูป 2.3 แสดง Ultrasonic ranging module HC-SR04 8
รูป 2.4 การต่อใช้งานโมดูล Ultrasonic HC-SR04 9
รูป 2.5 แสดง ภาษา Python 11
รูป 3.1 การจำลองชิ้นงาน 14
รูป 3.2 ระยะห่างระหว่างตัวตรวจรู้ Ultrasonic กับ บอร์ด Raspberry 15
รูป 3.3 แสดงบริเวณที่สามารถตรวจสอบได้ว่ามีรถยนต์ค้างอยู่บนราง 16
รูป 3.4 การคำนวณระยะห่างระหว่างตัวตรวจรู้ Ultrasonic กับรางรถไฟ 17
รูป 3.5 ระยทางระหว่างตัวตรวจรู้ Ultrasonic 17
รูป 3.6 Network Connections 18
รูป 3.7 เลือก Properties 19
รูป 3.8 การเลือกที่ช่อง Allow other network users to connection through this computer's internet connection 19
รูป 3.9 Unidentified network 20
รูป 3.10 เลือก Status 20
รูป 3.11 Detail 21
รูป 3.12 IP Address ของ Ethernet 21
รูป 3.13 หา IP Address ของบอร์ด Raspberry Pi 22
รูป 3.14 เปิดโปรแกรม PuTTy และพิมพ์ IP Address 22
รูป 3.15 Login 23
รูป 3.16 โหมด SSH 23
รูป 3.17 ใช้งานโปรแกรม xrdp 24
รูป 3.18 หมายเลข IP ของบอร์ด Raspberry Pi ที่ได้จากโปรแกรม Advance IP scan 24
รูป 3.19 เข้าสู่หน้า Login xrdp เข้าใช้งานบอร์ด Raspberry Pi 25
รูป 3.20 Desktop ของระบบปฏิบัติการ Raspbian 25
รูป 3.21 การเขียนโปรแกรมให้บอร์ด Raspberry Pi โดยใช้ภาษา Python 26

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูปภาพ(ต่อ)

รายการ **หน้า**

รูป 3.22 การรันบนโหมด SSH ในในโปรแกรม PuTTY 32
รูป 3.23 ตัวอย่างการแบ่งแรงดัน 33
รูป 3.24 ตัวอย่างวงจรแบ่งแรงดัน 34
รูป 4.1 การฟแสดงการปรับเทียบตัวตรวจรู้ Ultrasonic 36
รูป 4.2 แผนภูมิการเปรียบเทียบการวัดระยะทาง 37
รูป 4.3 แบบจำลองทางข้ามรถไฟ 38
รูป 4.4 รถไฟวิ่งผ่านตัวตรวจรู้ Ultrasonic ตัวที่ 1 38
รูป 4.5 รถไฟวิ่งผ่านตัวตรวจรู้ Ultrasonic ตัวที่ 2 39
รูป 4.6 มีรถค้างอยู่บนทางลักผ่าน 39
รูป 4.7 กราฟแสดงการแจ้งเตือนรถยนต์เมื่อมีรถไฟมา 40
รูป 4.8 กราฟแสดงการแจ้งเตือนรถไฟเมื่อมีรถยนต์ค้างอยู่บนรางรถไฟ 41

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญตาราง

เรื่อง	หน้า
ตารางที่ 2.1 แสดงการเปรียบเทียบข้อดีและข้อจำกัดของ ตัวตรวจรู้ Infrared และตัวตรวจรู้Ultrasonic	10
ตารางที่ 4.1 แสดงการวัดระยะโดยใช้ตัวตรวจรู้ Ultrasonic กับระยะทางจริง	36
ตารางที่ 4.2 แสดงค่า R^2 (R-Squared)	37
ตารางที่ 4.3 แสดงผลการทดสอบจากแบบจำลอง	40

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ด้วยปัจจุบัน อุบัติเหตุที่เกิดขึ้นกับรถไฟมีความถี่ในการเกิดขึ้นบ่อยครั้ง ส่งผลกระทบต่อประชาชนที่สัญจรผ่านทางรถไฟ โดยเฉพาะจุดลักผ่านหรือทางลักผ่าน ก่อให้เกิดความเสียหายทั้งชีวิตและทรัพย์สิน ด้วยเหตุนี้ทางการรถไฟแห่งประเทศไทยจึงได้จัดสร้างระบบแจ้งเตือนการมาของรถไฟตามบริเวณสำคัญจุดต่างๆ หรือแยกใหญ่ๆตามหัวเมือง แต่ระบบดังกล่าวมีงบประมาณในการก่อสร้างสูง อีกทั้งงบประมาณของรัฐบาลที่มีอยู่อย่างจำกัด จึงทำให้มีบริเวณที่ไม่มีระบบแจ้งเตือนการมาของรถไฟโดยเฉพาะบริเวณทางลักผ่าน

ซึ่งทางลักผ่านนี้เกิดจากการก่อสร้างด้วยงบประมาณขององค์การบริหารส่วนท้องถิ่น เพื่อให้ประชาชนสัญจรข้ามทางรถไฟได้อย่างสะดวก แต่อย่างไรก็ตาม ณ บริเวณทางข้ามนี้ยังมีความเสี่ยงในการเกิดอุบัติเหตุอยู่ เนื่องจากประชาชนไม่สามารถมองเห็นการมาของรถไฟหรืออาจชะลอหรือหยุดรถไม่ทันท่วงที

ดังนั้นคณะทำงานจึงเห็นความสำคัญของการเพิ่มความปลอดภัยของประชาชนในการสัญจรผ่านทางลักผ่านดังกล่าว คณะทำงานจึงเสนอการเพิ่มความปลอดภัยบริเวณทางลักผ่านใช้บอร์ด Raspberry Pi ร่วมกับตัวตรวจรู้ Ultrasonic ทั้งนี้หากเกิดอุบัติเหตุต่อประชาชนจะก่อให้เกิดความเสียหายที่เกิดมูลค่าสูง เช่น การซ่อมหัวจักรของรถไฟ และรถยนต์ของประชาชน **“คณะผู้จัดทำมิได้ส่งเสริมการสัญจรบริเวณจุดลักผ่าน หากเพียงแต่ต้องการเพิ่มความปลอดภัยแก่ประชาชนที่สัญจรผ่านทางลักผ่านด้วยงบประมาณที่มีอยู่อย่างจำกัด”**

1.2 วัตถุประสงค์

1. เพื่อสร้างระบบเตือนการมาของรถไฟมาบริเวณทางลักผ่านโดยใช้บอร์ด Raspberry Pi และตัวตรวจรู้ Ultrasonic
2. เพื่อเพิ่มความปลอดภัยบริเวณทางลักผ่านหรือจุดลักผ่าน แก่ประชาชนผู้สัญจรไปมาบริเวณจุดดังกล่าว
3. เพื่อสร้างระบบแจ้งเตือนรถไฟ เมื่อมีรถยนต์ค้างอยู่บนราง

1.3 ขอบเขตงาน

1. ศึกษาระบบทางข้ามรถไฟเฉพาะบริเวณทางลักผ่าน
2. ศึกษาวิธีการทำงานของ Raspberry Pi เพื่อทำหน้าที่สั่งงานตัวตรวจรู้ Ultrasonic
3. สามารถแสดงข้อมูลที่ตัวตรวจรู้ Ultrasonic วัดได้

1.4 ขั้นตอนการดำเนินงาน

1. ปรึกษาอาจารย์ที่ปรึกษาโครงงานเกี่ยวกับขอบเขตของโครงงานที่จะทำ
2. จัดหาอุปกรณ์ที่เกี่ยวข้องกับโครงงาน
3. ศึกษาการใช้โปรแกรม Raspberry Pi และวิธีการเขียนภาษา Python
4. ศึกษาหลักการทำงานและวิธีการใช้งานของตัวตรวจรู้ Ultrasonic
5. สร้างโมเดลสำหรับการทดลอง
6. ทดสอบการทำงานของอุปกรณ์และแก้ไขปรับปรุงให้เป็นไปตามวัตถุประสงค์ของโครงงาน
7. สรุปผล เขียนรายงานโครงงานและนำเสนอโครงงาน

1.5 ผลที่คาดว่าจะได้รับ

1. ระบบสามารถเพิ่มความปลอดภัยให้กับประชาชนที่สัญจรผ่านทางลักผ่านดังกล่าวได้
2. เป็นแนวทางในการสร้างเครื่องแจ้งเตือนการมาของรถไฟที่มีราคาประหยัด ทำให้ประชาชนทุกคนเข้าถึงได้หากเกิดอุบัติเหตุที่รุนแรง ส่งผลต่อชีวิตและทรัพย์สิน

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 รถไฟ

รถไฟ (อังกฤษ: Train)[1] เป็นกลุ่มของยานพาหนะที่เคลื่อนที่ไปตามรางเพื่อการขนส่งสินค้าหรือผู้โดยสารจากที่หนึ่งไปยังอีกที่หนึ่ง รางส่วนใหญ่มักจะประกอบด้วยราง 2 เส้นขนานกัน แต่ยังหมายรวมถึงประเภทรางเดี่ยวหรือประเภทที่ใช้พลังงานแม่เหล็กด้วย รถไฟจะขับเคลื่อนด้วยหัวรถจักรหรือขับเคลื่อนด้วยเครื่องยนต์หลายๆตัวที่ติดอยู่ใต้ท้องรถ รถไฟสมัยใหม่จะใช้กำลังจากเครื่องยนต์ดีเซลหรือจากไฟฟ้าที่ส่งมาตามสายไฟที่อยู่เหนือตัวรถหรือตามรางสาม (Third Rail) เดิมรถไฟขับเคลื่อนโดยใช้หม้อต้มน้ำทำให้เกิดไอน้ำ ไอน้ำทำให้เกิดแรงดัน แรงดันจะทำการขับเคลื่อนกลไกทำให้ล้อรถไฟเคลื่อนที่ได้ การที่ใช้ฟืนเป็นแหล่งพลังงานในการต้มน้ำ และฟืนที่ทำให้เกิดเปลวไฟ ทำให้เรียกรถชนิดนี้ว่า รถจักรไอน้ำ

2.1.1 ข้อมูลทางกายภาพ

ขนาดความกว้างรางรถไฟ (อังกฤษ: rail gauge หรือ track gauge)[2] คือระยะห่างของรางรถไฟ โดยวัดจากหัวรางด้านในข้างซ้ายถึงหัวรางด้านในข้างขวา สแตนดาร์ดเกจ (standard gauge) เป็นชื่อของขนาดความกว้างรางที่นิยมใช้มากที่สุดทั่วโลก โดยประมาณ 60% ของรางรถไฟทั้งหมด โดยมีขนาด 1,435 มิลลิเมตร (1.435 เมตร) โดยในเมืองไทยรางรถไฟส่วนใหญ่มีขนาดความกว้างที่เรียกว่า มีเตอร์เกจ ที่มีขนาดความกว้าง 1,000 มิลลิเมตร (1เมตร) ซึ่งมีแผนการจะพัฒนาปรับปรุงเพื่อใช้สำหรับรถไฟความเร็วสูง

รถไฟความเร็วสูง หรือ **ระบบรางความเร็วสูง** (อังกฤษ: High-Speed Rail (HSR))[3] เป็นระบบขนส่งทางรางที่วิ่งด้วยความเร็วสูงกว่าระบบขนส่งทางรางทั่วไปอย่างมาก โดยการใช้ระบบล้อเลื่อน (อังกฤษ: rolling stock) พิเศษรวมกับระบบรางที่ออกแบบมาให้ใช้โดยเฉพาะ ส่วนใหญ่วิ่งบนรางที่มีขนาด 1.435 เมตร (สแตนดาร์ดเกจ (อังกฤษ: standard gauge))

---

[1] รถไฟ (อังกฤษ: Train) แหล่งที่มา : https://th.wikipedia.org/wiki/รถไฟ

[2] ขนาดความกว้างรางรถไฟ (อังกฤษ: rail gauge หรือ track gauge) แหล่งที่มา: https://th.wikipedia.org/wiki/ขนาดความกว้างรางรถไฟ

[3] รถไฟความเร็วสูง หรือ ระบบรางความเร็วสูง (อังกฤษ: High-Speed Rail (HSR)) แหล่งที่มา: https://th.wikipedia.org/wiki/รถไฟความเร็วสูง

2.1.2 จุดตัดทางรถไฟ

จุดตัดทางรถไฟที่พบเห็นในปัจจุบันแบ่งได้เป็น 4 ประเภท คือ จุดตัดทางรถไฟแบบต่างระดับ จุดตัดทางรถไฟแบบมีเครื่องกั้น จุดตัดทางรถไฟแบบควบคุมด้วยเครื่องหมายจราจร และจุดตัดทางรถไฟที่เป็นทางลักผ่าน

รูป 2.1 แสดงทางลักผ่าน (Illegal Crossing)

http://www.accident.or.th/datacenter/index.php/2015-10-13-04-59-28/88-2015-10-16-03-27-42

ทางลักผ่าน (Illegal Crossing)[4] คือ ทางตัดผ่านทางรถไฟที่เป็นทางเข้า-ออกประจำของเอกชนหรือผู้อยู่อาศัยบริเวณนั้นๆ ผู้ทำทางตัดผ่านอาจจะเป็นประชาชนหรือองค์กรปกครองส่วนท้องถิ่น เช่น เทศบาล องค์การบริหารส่วนตำบล **"แต่ไม่ได้มีการขออนุญาตทำทางตัดผ่านจากการรถไฟแห่งประเทศไทยหรือไม่ได้รับอนุญาตจากการรถไฟแห่งประเทศไทย"** จึงเป็นทางตัดผ่านที่ไม่มีการควบคุมด้านความปลอดภัย ซึ่งในปัจจุบันนี้พบว่ามีจำนวนมากและยากต่อการควบคุมให้อยู่ในมาตรฐานความปลอดภัย เมื่อพิจารณาค่าเฉลี่ยของการเกิดอุบัติเหตุบริเวณจุดตัดทางรถไฟพบว่าอุบัติเหตุที่เกิดขึ้น

[4] ทางลักผ่าน (Illegal Crossing) แหล่งที่มา: http://www.accident.or.th/datacenter/index.php/2015-10-13-04-59-28/88-2015-10-16-03-27-42

ประมาณร้อยละ 87 เป็นอุบัติเหตุที่เกิดขึ้นบริเวณจุดตัดทางรถไฟที่ไม่มีเครื่องกั้น ได้แก่ จุดตัดทางรถไฟประเภททางลักผ่านและจุดตัดทางรถไฟที่ควบคุมด้วยเครื่องหมายจราจรเพียงอย่างเดียว

หากแต่สิ่งเดียวที่การรถไฟฯ มีขอบเขตในการป้องกันได้คือการติดตั้งป้ายเตือน และป้ายหยุดก่อนข้ามทางรถไฟ แต่สิ่งป้องกันเหล่านั้นก็ป้องกันได้ไม่เต็ม 100% จากพฤติกรรมการใช้รถใช้ถนนของผู้ขับขี่ยวดยาน ซึ่งนอกจากจะเกิดอุบัติเหตุแล้ว รถไฟยังไม่สามารถทำความเร็วได้เสถียร ทำให้รถไฟล่าช้ากว่ากำหนดเวลาเดินรถ

2.1.3 กฎหมายที่เกี่ยวข้อง

มาตรา 62 ในทางเดินรถตอนใดที่มีทางรถไฟผ่านถ้าปรากฏว่า
(1) มีเครื่องหมายหรือสัญญาณระวังรถไฟแสดงว่ารถไฟกำลังจะผ่าน
(2) มีสิ่งปิดกั้นหรือมีเจ้าหน้าที่ให้สัญญาณแสดงว่ารถไฟกำลังจะผ่าน
(3) มีเสียงสัญญาณของรถไฟหรือรถไฟกำลังแล่นผ่านเข้ามาใกล้อาจเกิดอันตรายในเมื่อจะขับรถผ่านไปผู้ขับขี่ต้องลดความเร็วของรถและหยุดรถให้ห่างจากทางรถไฟไม่น้อยกว่าห้าเมตร เมื่อรถไฟผ่านไปแล้วและมีเครื่องหมายหรือสัญญาณให้รถผ่านได้ ผู้ขับขี่จึงจะขับรถผ่านไปได้

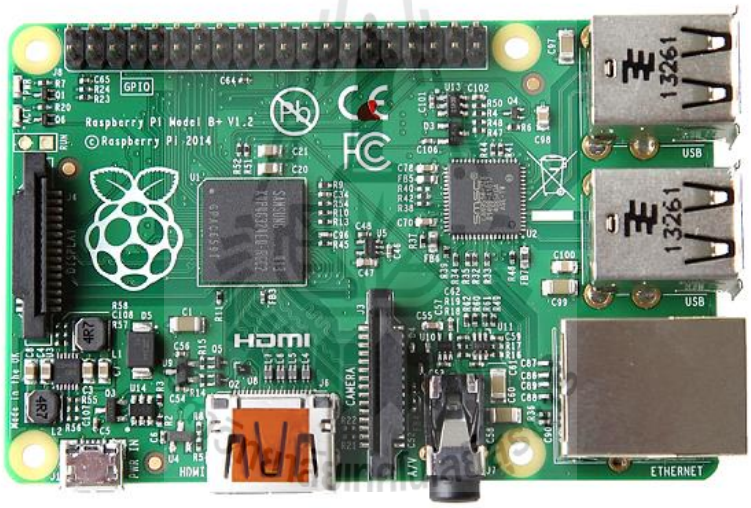
มาตรา 63 ในทางเดินรถตอนใดที่มีทางรถไฟผ่านไม่ว่าจะมีเครื่องหมายระวังรถไฟหรือไม่ ถ้าทางรถไฟนั้นไม่มีสัญญาณระวังรถไฟหรือสิ่งปิดกั้น **ผู้ขับขี่ต้องลดความเร็วของรถและหยุดรถห่างจากทางรถไฟในระยะไม่น้อยกว่าห้าเมตร เมื่อเห็นว่าปลอดภัยแล้วจึงจะขับรถผ่านไปได้**[5]

มหาวิทยาลัยเทคโนโลยีสุรนารี

[5] กฎหมายที่เกี่ยวข้อง แหล่งที่มา: http://www.accident.or.th/datacenter/index.php/2015-10-13-04-59-28/88-2015-10-16-03-27-42

2.2 Raspberry Pi 512MB Model B+[6]

Model B+ ยังคงใช้ชิพ BCM2835 ของ Broadcom ในตระกูล Application Processor ของ ARM บนสถาปัตยกรรม ARM11 มีความเร็วในการทำงานที่ 700MHz เท่าเดิม ใช้ชิพช่วยประมวลผลกราฟฟิกของ VideoCore IV เช่นเดิม มีหน่วยความจำหลัก (Primary Memory: RAM) ขนาด 512MB แบบเดิม (บอร์ดลง RAM ครอบ CPU ไว้) และยังคงไม่มีหน่วยความจำสำรอง (Secondary Storage Memory) คือไม่มีพวกแฟลชเมมโมรี่ต่างๆ บนบอร์ดเหมือนเดิม ขนาดความกว้างและความยาวบอร์ดเท่าเดิม ใช้แหล่งจ่ายไฟผ่านคอนเน็คเตอร์ microUSB แบบเดิม สามารถรันระบบปฏิบัติการ Raspbian หรือระบบอื่นๆ ได้จากไฟล์ตัวเดิมที่ออกหลังเดือนมิถุนายน 2557 ขาสัญญาณบนพอร์ต GPIO ในส่วน 26 ขาแรกจัดเรียงเหมือนเดิม ใช้พอร์ต HDMI แบบ Full-Size ขนาดเท่าเดิม พอร์ตสำหรับส่วนของ Audio Out ผ่าน Audio Jack ยังคงทำหน้าที่ได้เหมือนเดิม คอนเน็คเตอร์สำหรับต่อกล้อง Raspberry Pi Camera และ DSI Display ยังคงมีอยู่เช่นเดิม



รูป 2.2 Raspberry Pi 512MB Model B+

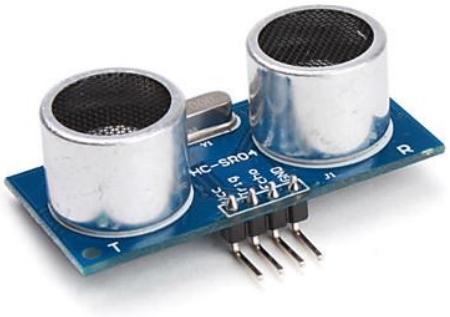
http://www.raspberrypi.org/introducing-raspberry-pi-model-b-plus/

[6] Raspberry Pi 512MB Model B+ แหล่งที่มา: http://www.raspberrypi.org/introducing-raspberry-pi-model-b-plus/

- เพิ่มจำนวนขาสัญญาณของ GPIO มากขึ้น จากเดิม 26 ขา เป็น 40 ขา
- เพิ่มจำนวนช่องเสียบ USB ให้มากขึ้น จากเดิมที่เคยมี 2 ช่องใน Model B ได้เพิ่มเป็น 4 ช่องใน Model B+
- เปลี่ยนมาใช้ช่องเสียบ Micro SD Card แทน Standard Full-Size SD Card
- ปรับปรุงภาคจ่ายไฟใหม่ ส่งผลต่อสเถียรภาพโดยรวมของบอร์ด ทั้งช่วยยืดอายุการใช้งานของอุปกรณ์ต่างๆ จากไฟที่นิ่งขึ้น แล้วยังช่วยลดโอกาสที่ระบบจะล่มได้จากการดึงไฟเวลาถอด-เสียบ USB
- วงจรเสียงได้รับการปรับปรุงด้านการป้องกันสัญญาณรบกวนให้ดีขึ้นอีกเล็กน้อย ซึ่งเป็นผลพวงจากการปรับปรุงภาคจ่ายไฟของบอร์ด
- ตำแหน่งของจุดยึดน็อตบนแผ่นวงจรและตำแหน่งคอนเน็คเตอร์ต่างๆ เปลี่ยนไป ดังนั้นกล่องที่เคยใช้งานอยู่จะใช้ไม่ได้ต้องเปลี่ยนกล่องใหม่
- LED แสดงผลบนบอร์ดก็เปลี่ยนตำแหน่งไปและลดจำนวนลง

มหาวิทยาลัยเทคโนโลยีสุรนารี

2.3 Ultrasonic ranging module HC-SR04[7]



รูป 2.3 แสดง Ultrasonic ranging module HC-SR04

http://www.arduitronics.com/product/20/ultrasonic-sensor-module-hc-sr04

HC-SR04 เป็นเซนเซอร์โมดูลสำหรับตรวจจับวัตถุและวัดระยะทางแบบไม่สัมผัส โดยใช้คลื่น Ultrasonic ซึ่งเป็นคลื่นเสียงความถี่สูงเกินกว่าการได้ยินของมนุษย์ วัดระยะได้ตั้งแต่ 2 – 400 เซนติเมตร หรือ 1 – 156 นิ้ว สามารถต่อใช้งานกับไมโครคอนโทรลเลอร์ได้ง่าย ใช้พลังงานต่ำ เหมาะกับการนำไปประยุกต์ใช้งานด้านระบบควบคุมอัตโนมัติ หรืองานด้านหุ่นยนต์ หลักการทำงาน จะเหมือนกันกับการตรวจจับวัตถุด้วยเสียงของค้างคาว โดยจะประกอบไปด้วยตัว รับ-ส่ง Ultrasonic ตัวส่งจะส่งคลื่นความถี่ 40 kHz ออกไปในอากาศด้วยความเร็วประมาณ 346 เมตรต่อวินาที และตัวรับจะคอยรับสัญญาณที่สะท้อนกลับจากวัตถุ เมื่อทราบความเร็วในการเคลื่อนที่ของคลื่น, เวลาที่ใช้ในการเดินทางไป-กลับ (t) ก็จะสามารถคำนวณหาระยะห่างของวัตถุ (S) ได้

[7] Ultrasonic ranging module HC-SR04 แหล่งที่มา:http://www.arduitronics.com/product/20/ultrasonic-sensor-module-hc-sr04

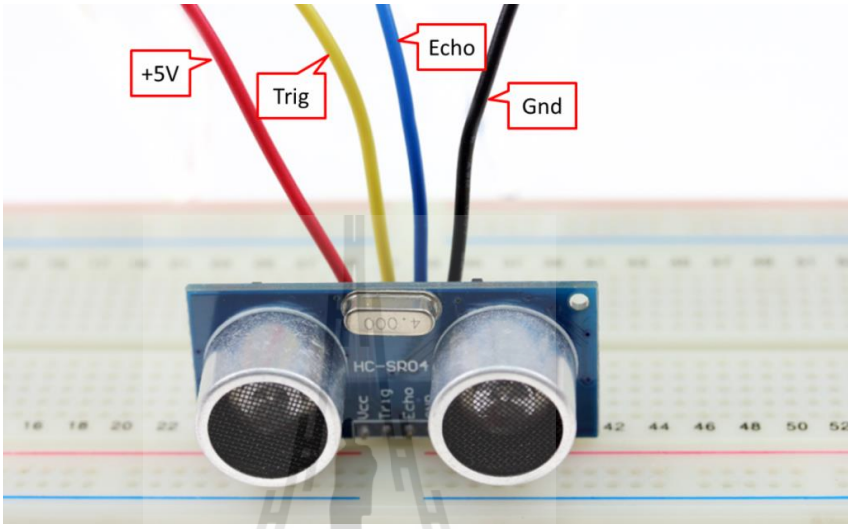
การต่อเพื่อใช้งานทำได้โดยต่อ **PIN** ดังนี้

PIN ที่ 1 -- 5V Supply

PIN ที่ 2 -- Trigger Pulse Input

PIN ที่ 3 -- Echo Pulse Output

PIN ที่ 4 -- GND



รูป 2.4 การต่อใช้งานโมดูล Ultrasonic HC-SR04

http://aimagin.com/blog/wp-content/uploads/2014/02/pic02_HC-SR04.png

Specifications:

Working voltage: DC 5 V

Static current: 3 mA

Working temperature: 0 ~ + 70

Output way: GPIO

Induction Angle: Less than 15

Detection range:2 cm to 4 m

Detecting precision: 0.3 cm + 1%

Sensor size: Approx. 45 x 20 x 1.6mm

มหาวิทยาลัยเทคโนโลยีสุรนารี

ตารางที่1 แสดงการเปรียบเทียบข้อดีและข้อจำกัดของ ตัวตรวจรู้ Infrared และตัวตรวจรู้ Ultrasonic

ตัวตรวจรู้ Infrared		**ตัวตรวจรู้ Ultrasonic**	
ข้อดี	**ข้อจำกัด**	**ข้อดี**	**ข้อจำกัด**
1.สามารถนำไปใช้ตรวจความร้อนของร่างกายได้ 2.ความยาวคลื่นมีขนาดสั้น ทางเดินของแสงเป็นแนวตรง 3.ราคาถูก 4.ง่ายต่อการผลิต 5.ปลอดภัยต่อการตรวจจับสัญญาณ 6.ไม่สามารถทะลุผ่านวัตถุ ทำให้สามารถติดตั้งในห้องทำงานติดกันได้ 7.มีความปลอดภัยในการส่งข้อมูลดีกว่าคลื่นวิทยุ	1.แสงที่กระทบต่อผลการตรวจจับทำให้ข้อมูลที่ได้มาผิดพลาด 2.ไม่สามารถผ่านวัตถุทึบแสงได้	1.ตรวจจับวัตถุโดยการใช้คลื่นเสียงที่มีความถี่สูง ซึ่งมนุษย์ไม่ได้ยิน 2.มีระยะตรวจจับสูง 3.ตรวจจับวัตถุโดยไม่สนใจสีหรือพื้นผิวของวัตถุ 4.ใช้ได้กับของแข็งและของเหลว ทั้งเปียกและแห้ง 5.ตรวจจับวัตถุที่มีไอฝุ่นได้	1.Ultrasonicไม่สามารถทำงานในที่แคบได้ 2.ในสถานที่ที่เป็นสุญญากาศ Ultrasonic จะไม่สามารถใช้งานได้ เพราะคลื่นเสียงต้องเดินทางผ่านตัวกลาง 3.ควรหลีกเลี่ยงการติดตั้งUltrasonic ในบริเวณใกล้ๆกัน 4.ไม่เหมาะกับวัตถุผิวนุ่มและผิวโค้ง

มหาวิทยาลัยเทคโนโลยีสุรนารี

2.4 ภาษา Python

รูป 2.5 แสดง ภาษา Python

http://bbee.exteen.com/20080313/python

Python[8] คือชื่อภาษาที่ใช้ในการเขียนโปรแกรมภาษาหนึ่งที่มีความสามารถสูง ไม่แพ้ภาษาอื่นๆที่มีอยู่ในตอนนี้ Python นั้นเป็นภาษาที่ Open Source ทำให้ทุกคนสามารถที่จะนำ Python มาพัฒนาโปรแกรมของเราได้ฟรีๆโดยไม่ต้องเสียค่าใช้จ่าย และความเป็น Open Source ทำให้มีคนเข้ามาช่วยกันพัฒนาให้ Python มีความสามารถสูงขึ้น และใช้งานได้ครบคุมกับทุกลักษณะงาน
Python ถูกสร้างขึ้นมาโดย **Guido Van Rossum** โค้ดของ Python ถูกสร้างขึ้นมาจากภาษาซี การประมวลผลจะทำในแบบอินเทอร์พรีเตอร์ คือจะประมวลผลไปทีละบรรทัดและปฏิบัติตามคำสั่งที่ได้รับ Python เวอร์ชันแรกคือ เวอร์ชัน 0.9.0 ออกมาเมื่อปี 2533 และเวอร์ชันปัจจุบันคือ 2.5.2 ออกเมื่อวันที่ 21 กุมภาพันธ์ 2551 และได้มีการพัฒนา Python ในรุ่นที่ 3 คือ Python 3000 หรือ Py3k โดยจะมีการปรุงปรุงใหม่เกือบหมด และตอนนี้อยู่ในระหว่างการทดลองอยู่

คุณลักษณะเด่นของภาษา Python

- ภาษา Python สนับสนุนแนวแบบคิดออปเจกต์โอเรียนเทด หรือ OOP (Object Oriented Programming)
- Pythonเป็นภาษาคอมพิวเตอร์ที่ไม่คิดมูลค่าการใช้งานและเป็นภาษาที่มีความยืดหยุ่นสูงมาก
- โค้ดที่เขียนด้วย Python สามารถนำไปรันบนระบบปฏิบัติการอื่นๆ ได้(Portable)เช่น Linux, Ms-windows (95, 98, NT, 2000, XP), Amiga, Be-OS, OS/2, VMS, QNX, และระบบอื่นๆอีกมากมาย
- Python สนับสนุนเทคโนโลยี COM ของ Ms-windows

[8] http://bbee.exteen.com/20080313/python

- Python รวมมาตรฐานการอินเตอร์เฟส Tkinter ซึ่งสนับสนุนบนระบบ X windows, Ms-windows และ Macintosh การใช้คำสั่ง Tkinter API ช่วยให้โปรแกรมเมอร์ไม่ต้องแก้ไขโค้ดเมื่อนำไปรันบนระบบปฏิบัติการอื่นๆ
- Python เป็น Dynamic typing คือ สามารถเปลี่ยนชนิดข้อมูลได้ง่ายและสะดวก
- Python มี Buil-in Object Types คือ โครงสร้างของข้อมูลที่สามารถใช้ได้ใน Python ประกอบด้วย ลิสต์, ดิกชันนารี, สตริง ที่ง่ายต่อการใช้งานและมีประสิทธิภาพสูง
- Pythonมีเครื่องมือต่างๆ มากมาย เช่น การประมวลผลเท็กซ์ไฟล์ การเรียงข้อมูล การเชื่อต่อสตริง การตรวจสอบเงื่อนไขของข้อความ การแทนคำ เป็นต้น
- Python มีมอดูลสำหรับจัดการ Regular Expresion
- Python มีมอดูลที่สร้างขึ้นจากนักพัฒนาสนับสนุนมากมาย ได้แก่ COM, Image, CORBA, ORBs, XML เป็นต้น
- Python จัดการหน่วยความจำอย่างอัตโนมัติ สามารถจักการพื้นที่หน่วยความจำที่ไม่ต่อเนื่องให้ทำงานได้อย่างมีประสิทธิภาพ
- Python อนุญาตให้ฝังชุดคำสั่งของ Python เอาไว้ภายในโคดภาษา C/C++ ได้
- Python อนุญาตให้โปรแกรมเมอร์สร้าง Dynamic Link Libray (DLL) เพื่อใช้ร่วมกับ Python
- Python มีมอดูลสนับสนุนเกี่ยวกับเน็ตเวิร์ก โปรเซส เธรด regular, expression, xml, GUI และอื่นๆ
- Pythonประกอบด้วยมอดูลสำหรับสร้าง Internet Script และติดต่อกับอินเทอร์เน็ตผ่าน Sockets, และทำหน้าที่เป็น CGI Script ตรอดจนใช้งานคำสั่ง FTP, Glopher, XML และอื่นๆอีกมาก
- Python สามารถประมาลผมทางด้านวิยาศาสตร์ และวิศวกรรมศาสตร์ได้อย่างมีประสิทธิภาพ
- Python มีฟังก์ชันสนับสนุนฐานข้อมูล เช่น MySQL, Sybase, Oracle, Informix, ODBC และอื่นๆ
- Python มีไลบรารีสนับสนุนด้านการสร้างภาพกราฟฟิก เช่น ทำภาพเบลอ หรือภาพชัด หรือเขียนข้อความบนภาพ ตลอดจนบันถึกไฟล์ในรูปแบบต่างๆ ได้อย่างสะดวกและมีประสิทธิภาพ
- Python มีไลบรารีสนับสนุนด้านปัญยาประดิษฐ์
- Python มีไลบรารีสำหรับสร้างเอกสาร PDF โดยไม่ต้องติดตั้ง Acrobat Writer
- Python มีไลบรารีสำหรับสร้าง Shockwaves Flash (SWF) โดยไม่ต้องติดตั้ง Macromedia Flash

องค์กรที่นำภาษา Python มาประยุกต์ใช้

- Red Hat ใช้ Python เป็นเครื่องมือสำหรับการติดตั้งระบบปฏิบัติการ Linux (Installer)
- Google สร้างขึ้นด้วย Python
- Infoseek ใช้ Python จัดการ web search products
- Yahoo! ใช้ Python ในการจักการด้าน Internet services
- NASA ใช้ Python สำหรับ mission-control-system
- Lawrence Livermore Labs ใช้ Python สำหรับงาน numeric programming
- Industrial Light and Magic ใช้ Python สร้างภาพแอนิเมชัน

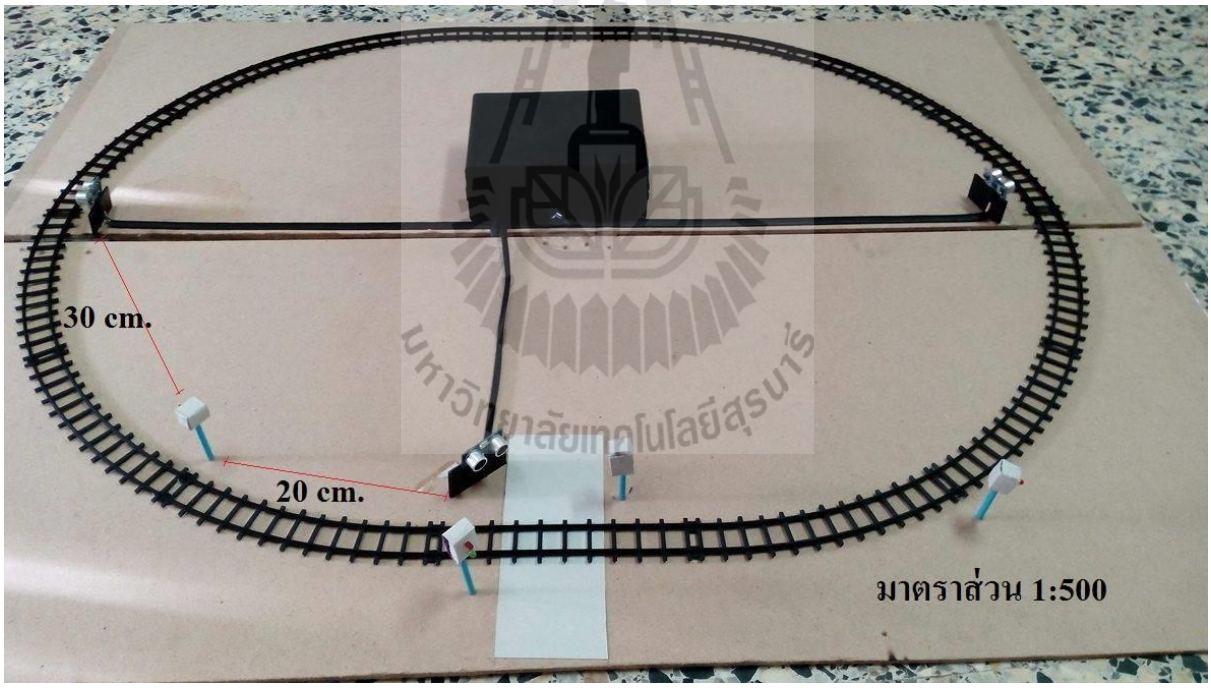
มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 3

การออกแบบและการจำลอง

3.1 บทนำ

ในบทนี้จะกล่าวถึงวิธีการออกแบบรวมถึงการจำลองชิ้นงาน เพื่อสามารถนำไปเป็นแนวทางในการสร้างชิ้นงานจริง โดยเริ่มจากการเตรียมอุปกรณ์ การใช้งานโปรแกรมสำหรับออกแบบหรือโปรแกรมสำหรับควบคุมบอร์ด Raspberry Pi โดยทำการจำลองสถานีรถไฟบริเวณทางลักผ่านให้ตัวตรวจรู้ Ultrasonic ตรวจสอบระยะทางและส่งผลกลับมายังไฟเตือน ดังรูป 3.1



รูป3.1 การจำลองชิ้นงาน

3.2 การออกแบบและติดตั้งตัวตรวจรู้ Ultrasonic

3.2.1. การคำนวณหาระยะห่างระหว่างตัวตรวจรู้ Ultrasonic กับทางลักผ่าน

กำหนดให้รถไฟวิ่งด้วยความเร็ว 60 กิโลเมตรต่อชั่วโมง หรือ 16.67 เมตรต่อวินาที

ถ้าต้องการให้ไฟแจ้งเตือน เตือนก่อนรถไฟมาถึงทางลักผ่าน 15 วินาที จะหาระยะทางที่ต้องติดตั้งตัวตรวจรู้ Ultrasonic จาก

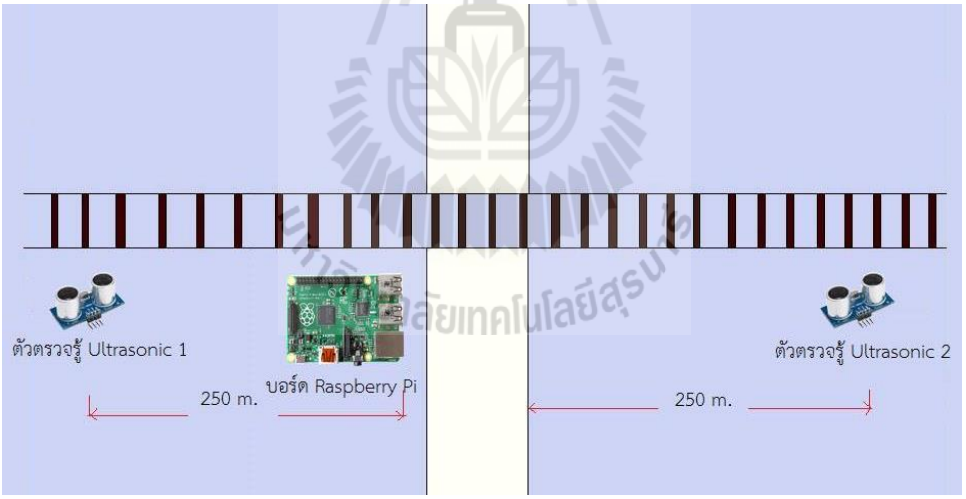
จากสมการ $V = \frac{S}{T}$

จะได้ว่า $S = VT$

$= (16.67\ m/s)(15s)$

$S = 250\ m$

ดังนั้น จะต้องติดตั้งตัวตรวจรู้ Ultrasonic ห่างจากทางลักผ่าน 250 เมตร



รูป3.2 ระยะทางระหว่างตัวตรวจรู้ Ultrasonic กับ บอร์ด Raspberry

3.2.2. การคำนวณหาระยะเวลาที่รถยนต์ค้างอยู่บนรางรถไฟ

กำหนดให้รถยนต์ที่วิ่งข้ามทางรถไฟ มีความเร็ว 20 กิโลเมตรต่อชั่วโมง หรือ 5.55 เมตรต่อวินาที

ระยะทางที่รถต้องวิ่งผ่านตัวตรวจรู้และทางรถไฟ เท่ากับ 3.435 เมตร โดยที่รถยนต์ยาว 5 เมตร

ดังนั้น ระยะทางที่รถยนต์จะวิ่งข้ามทางรถไฟพ้นทั้งคัน เท่ากับ เท่ากับ 8.435 เมตร

เวลาที่รถยนต์ใช้ข้ามทางรถไฟ หาได้จาก

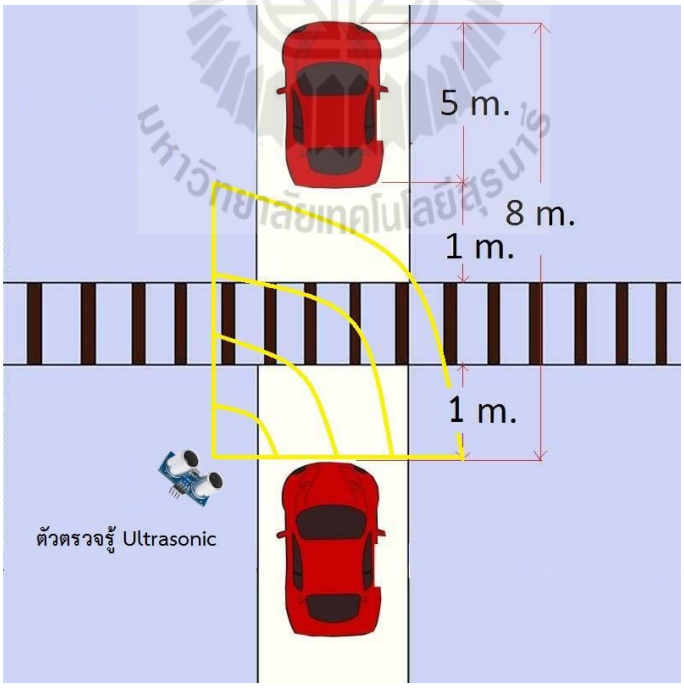
จากสมการ $V = \frac{S}{T}$

จะได้ว่า $T = \frac{S}{V}$

$$= \frac{8.435\ \mathrm{m}}{5.5\ \mathrm{m/s}}$$

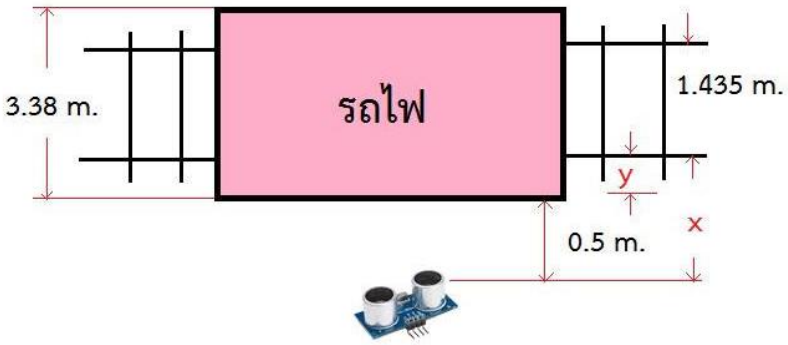
$$= 1.53 \approx 2\ \mathrm{s}$$

ดังนั้น ถ้าเวลาที่รถอยู่บนรางรถไฟ มากกว่า 2 วินาที แสดงว่ามีรถยนต์ค้างอยู่บนราง



รูป3.3 แสดงบริเวณที่สามารถตรงจสอบได้ว่ามีรถยนต์ค้างอยู่บนรางรถไฟ

3.2.3 การคำนวณหาระยะห่างระหว่างตัวตรวจรู้ Ultrasonic กับรางรถไฟ



รูป3.4 การคำนวณระยะทางระหว่างตัวตรวจรู้ Ultrasonic กับ รางรถไฟ

กำหนดให้ x = ระยะห่างจากตัวตรวจรู้ Ultrasonic ถึง ขอบรางรถไฟ

กำหนดให้ y = ระยะห่างจากขอบรางรถไฟ ถึง ขอบตู้รถไฟ

จากรูป 2y + 1.435 m. = 3.38 m.

จะได้ว่า y = $\frac{3.38-1.435}{2}$ = 0.9725 m. ≈ 1 m.

จากรูป x = 0.5 m. + y

จะได้ว่า x = 1.5 m.

ดังนั้น ระยะห่างระหว่างตัวตรวจรู้ Ultrasonic กับ รางรถไฟ เท่ากับ 1.5 เมตร

รูป3.5 ระยะทางระหว่างตัวตรวจรู้ Ultrasonic กับ รางรถไฟ

3.3 การใช้งาน Raspberry Pi

3.3.1 การติดตั้งระบบปฏิบัติการ Raspbian ให้บอร์ด Raspberry Pi

1. ดาวน์โหลดโปรแกรมสำหรับทำ SD card image โดยใช้โปรแกรม Win 32 Disk Imager จาก http://sourceforge.net/project/win32diskimager/

2. ดาวโหลดไฟล์ Image ระบบปฏิบัติการ Raspbian wheezy จาก http://rapberrypi.org/downloads/

3. แตกไฟล์ Zip ที่ดาวน์โหลดมาจะได้ไฟล์ Image ชื่อ 2015-05-05-wheezy-raspbian.img

4. เชื่อมต่อ SD card เข้ากับคอมพิวเตอร์ เปิดโปรแกรม Win 32 Disk Imager เลือกตำแหน่งไฟล์ Image ระบบปฏิบัติการและตำแหน่ง SD card

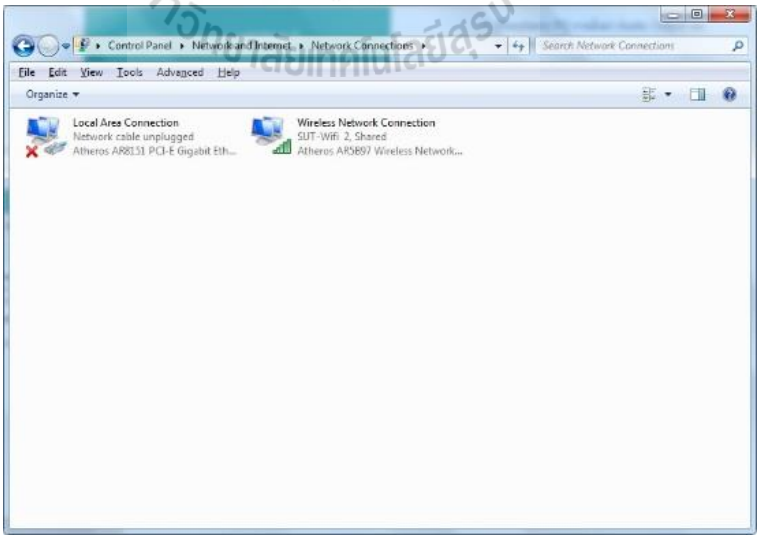
5. คลิกที่ปุ่ม Write เพื่อเขียนข้อมูลลงใน SD card โดยโปรแกรมจะมีหน้าต่างแจ้งเตือนการ Format ให้คลิก Yes เพื่อเริ่มดำเนินการ

6. รอจนโปรแกรมเขียนข้อมูลลงบน SD card เสร็จเรียบร้อย

7. ถอด SD card จากคอมพิวเตอร์ แล้วนำมาเชื่อมต่อกับบอร์ด Raspberry Pi

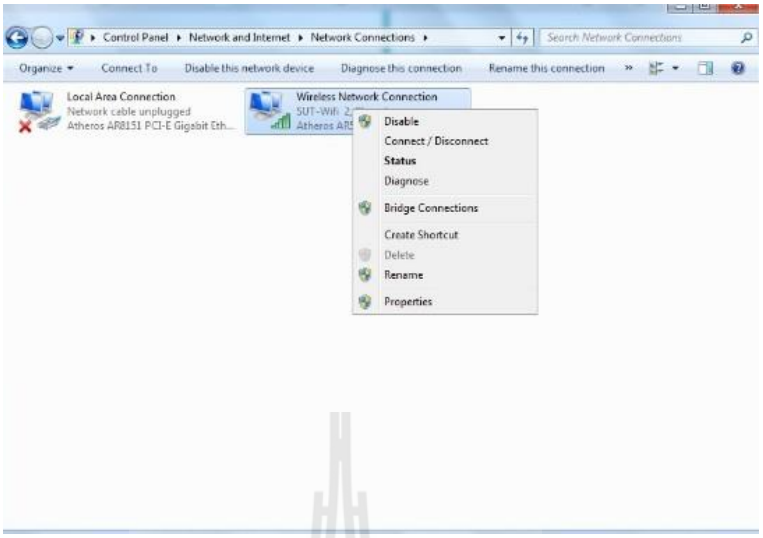
3.3.2 การเชื่อมต่อและแชร์อินเตอร์เน็ตจากคอมพิวเตอร์ให้กับ Raspberry Pi

1. เปิดหน้าต่าง Network Connections ขึ้นมาก็จะเห็นอุปกรณ์ที่เชื่อมต่อกับเครือข่าย



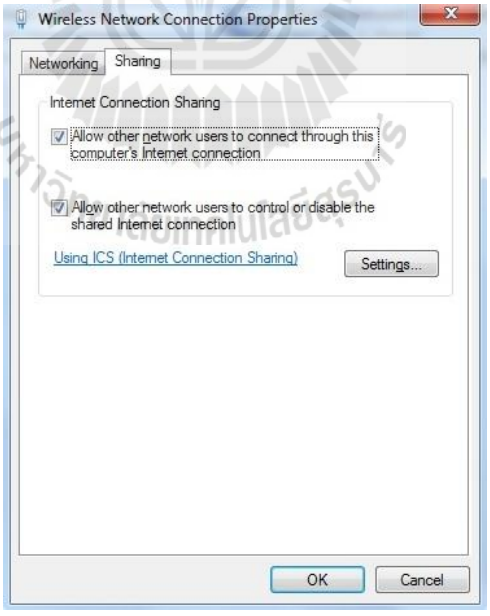
รูป3.6 Network Connections

2. คลิกขวาที่ Wi-Fi แล้วเลือก Properties



รูป3.7 เลือก Properties

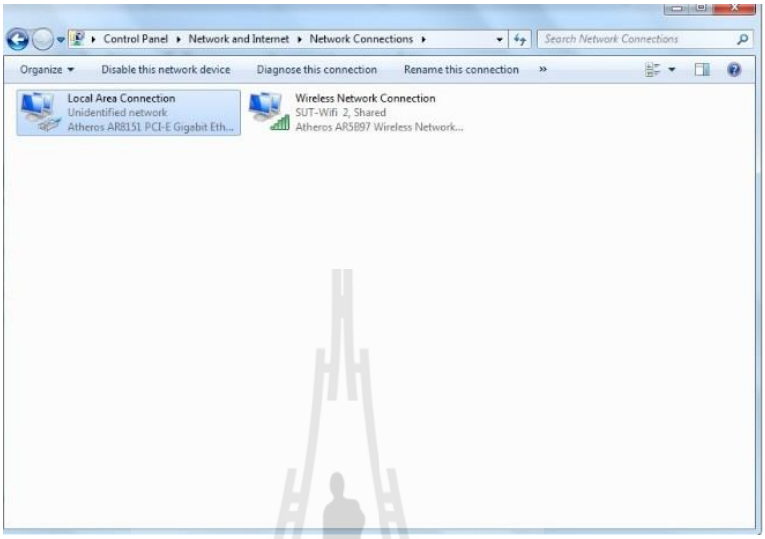
3. เลือกไปที่ Sharing แล้วเลือกที่ช่อง Allow other network users to connection through this computer's internet connection แล้วกดปุ่ม OK



รูป3.8 การเลือกที่ช่อง Allow other network users to connection through this computer's internet connection

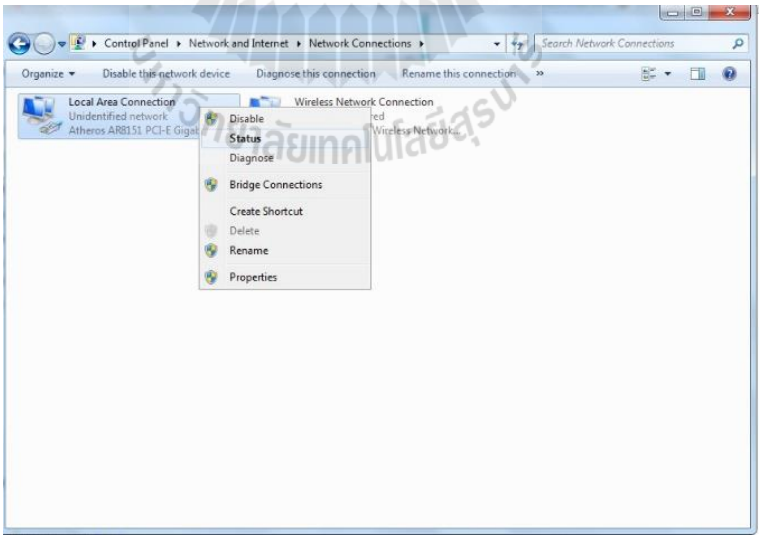
4. เชื่อมต่อสาย LAN ระหว่างคอมพิวเตอร์กับ Raspberry Pi โดยที่ฝั่ง Raspberry Pi จะต้องกำหนด IP Address เป็น DHCP

5. เชื่อมต่อสาย LAN แล้วจะเห็นว่าที่ Ethernet ขึ้นเชื่อมต่อสถานะเป็น Unidentified network



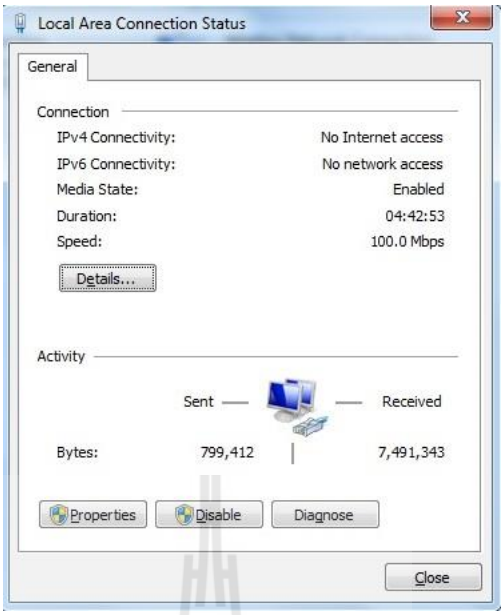
รูป3.9 Unidentified network

6. คลิกขวาที่ Ethernet แล้วเลือก Status



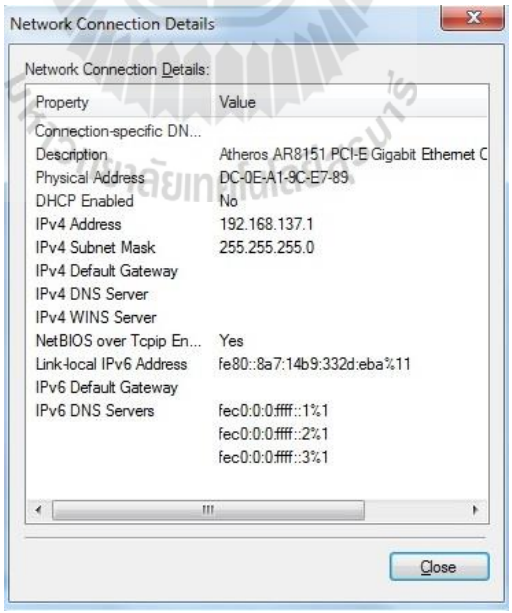
รูป3.10 เลือก Status

7. กดที่ปุ่ม Detail



รูป3.11 Detail

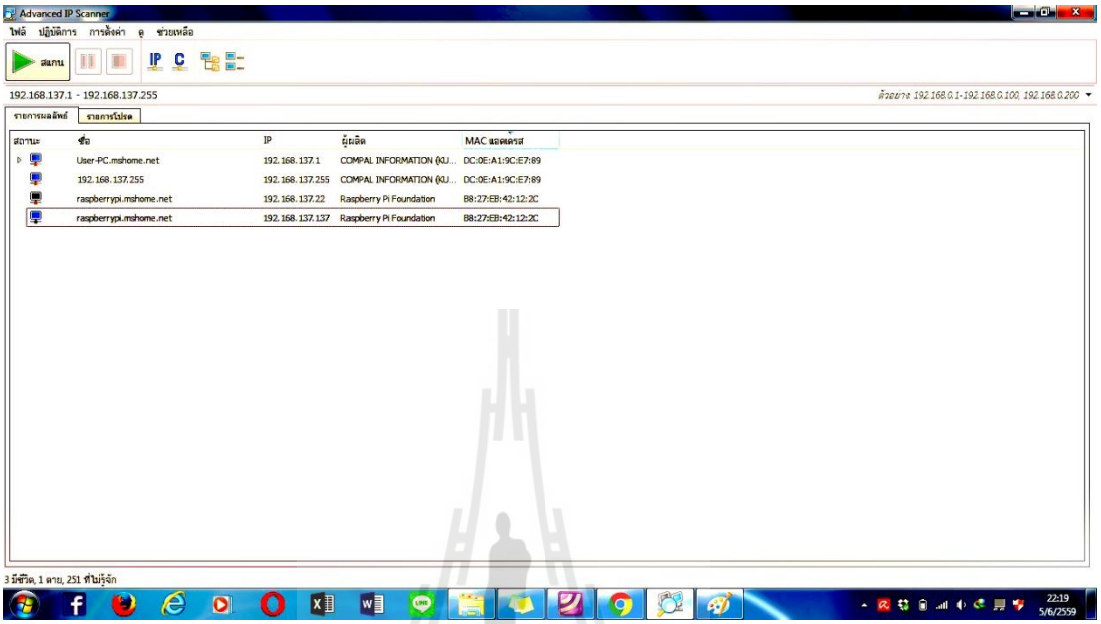
8. ที่ IPv4 Address จะขึ้น IP Address ของ Ethernet



รูป3.12 IP Address ของ Ethernet

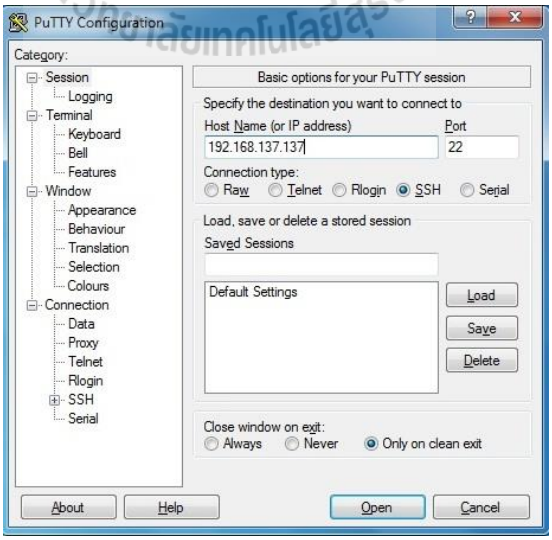
3.3.3 การใช้งาน SSH ด้วยโปรแกรม PuTTY

1. ใช้โปรแกรม Advance IP Scan เพื่อหา IP Address ของบอร์ด Raspberry Pi



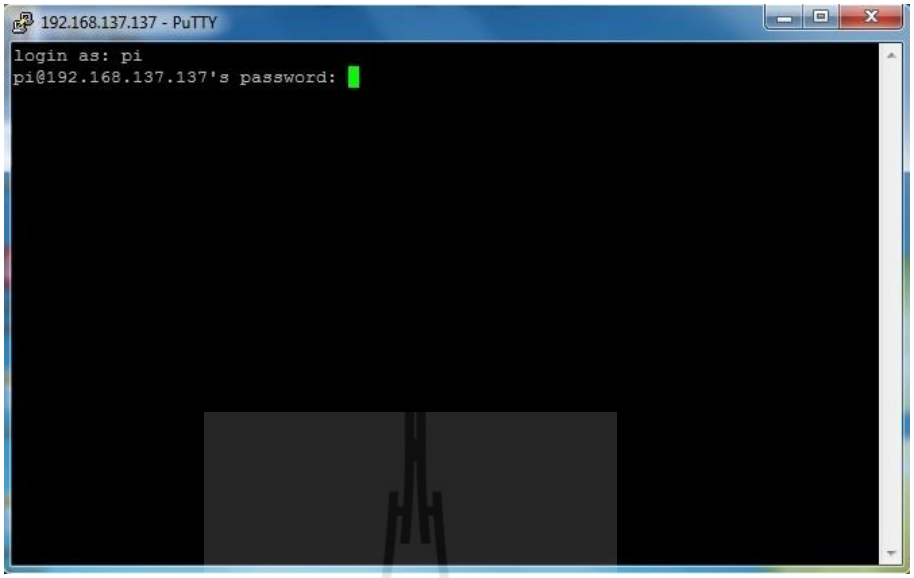
รูป3.13 หา IP Address ของบอร์ด Raspberry Pi

2. เปิดโปรแกรม PuTTy และพิมพ์ IP Address



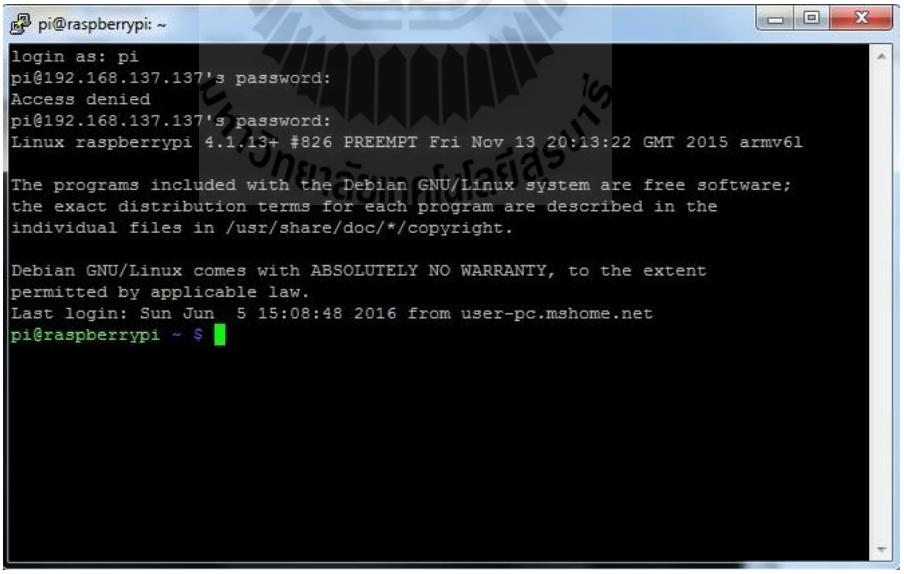
รูป3.14 เปิดโปรแกรม PuTTy และพิมพ์ IP Address

3. ใส่ Username และ Password เพื่อ Login



รูป3.15 Login

4. เข้าสู่ Raspberry Pi โหมด SSH



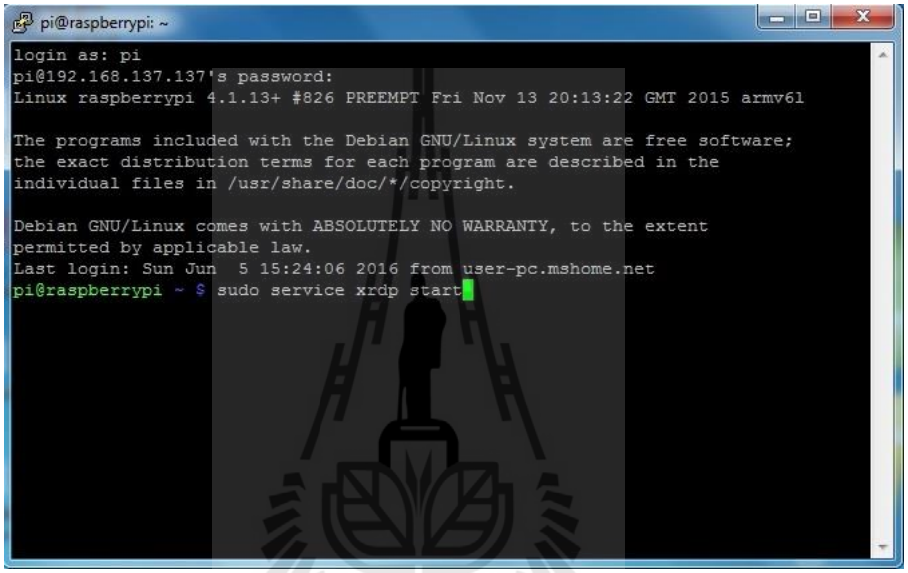
รูป3.16 โหมด SSH

5. ติดตั้งโปรแกรม xrdp เพื่อเป็นตัวกลางในการเชื่อมต่อ Remote ผ่าน LXterminal โดยใช้คำสั่งดังนี้

```
sudo apt-get install xrdp
```

6. เปิดใช้งานโปรแกรม xrdp โดยใช้คำสั่งดังนี้

```
sudo service xrdp start
```



รูป3.17 ใช้งานโปรแกรม xrdp

7. เปิดโปรแกรม Remote Desktop บนอุปกรณ์ปลายทาง ในที่นี้คือคอมพิวเตอร์และกรอกหมายเลข IP ของบอร์ด Raspberry Pi ที่ได้จากโปรแกรม Advance IP scan

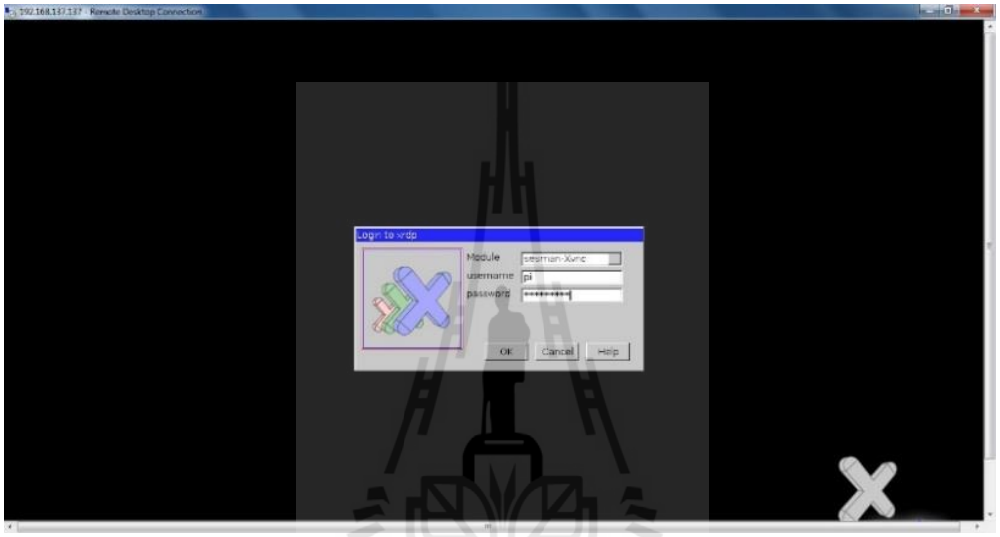
รูป3.18 หมายเลข IP ของบอร์ด Raspberry Pi ที่ได้จากโปรแกรม Advance IP scan

8. เข้าสู่หน้า Login xrdp เข้าใช้งานบอร์ด Raspberry Pi ดังรูป 3.15 โดยใช้

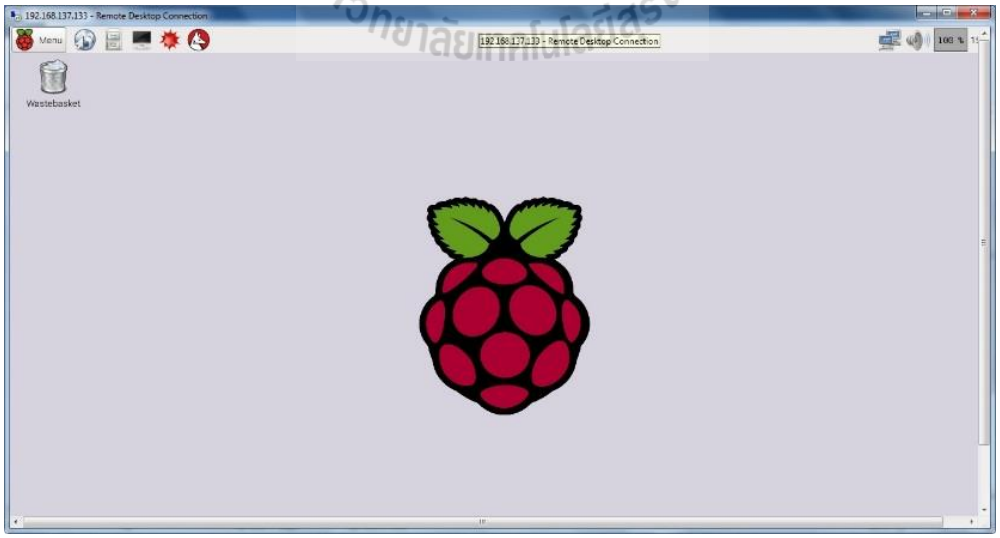
   Username: pi

   Password: Raspberry

   เหมือนการ Login เข้าใช้งานปกติ จากนั้นจะเข้าสู่หน้า Desktop ของระบบปฏิบัติการ Raspbian ดังรูป 3.19 ตามลำดับ



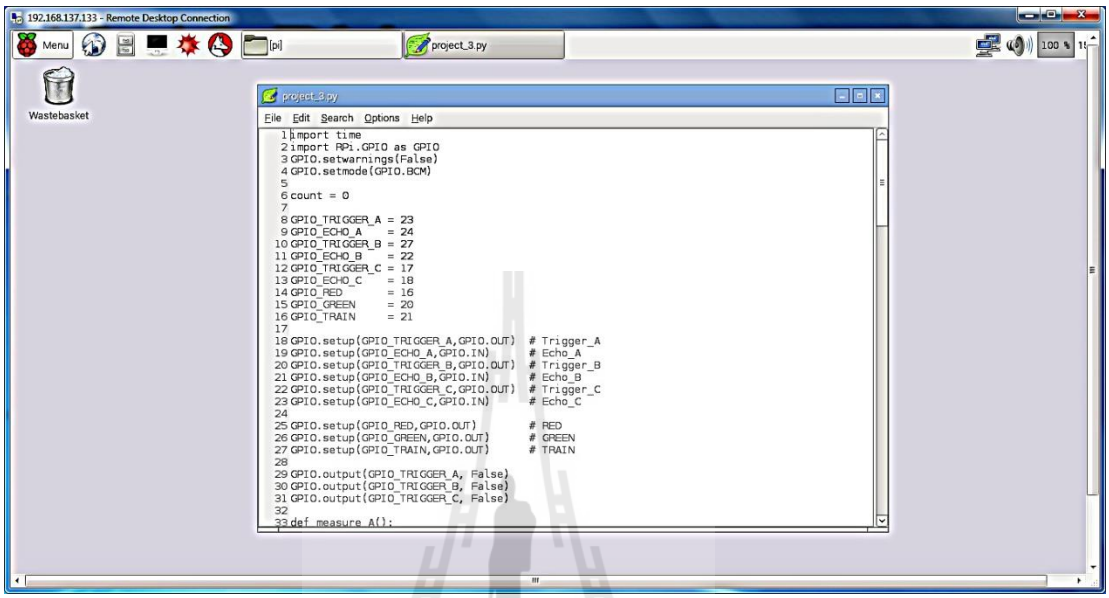
รูป3.19 เข้าสู่หน้า Login xrdp เข้าใช้งานบอร์ด Raspberry Pi



รูป3.20 Desktop ของระบบปฏิบัติการ Raspbian

3.3.4 การเขียนโปรแกรมโดยใช้บอร์ด Raspberry pi

ในการเขียนโปรแกรมให้บอร์ด Raspberry Pi โดยใช้ภาษา Python สามารถใช้ Text editor ใดๆ ก็ได้ ในที่นี้ใช้โปรแกรม Python2 ที่มาพร้อมกับระบบปฏิบัติการ Raspbian ที่มีรูปแบบของโปรแกรมตามรูป 3.17



รูป3.21 การเขียนโปรแกรมให้บอร์ด Raspberry Pi โดยใช้ภาษา Python

โดยจะเขียนภาษา Python ควบคุมบอร์ดดังนี้

ส่วนเริ่มต้นเป็นการนำเข้า Libraries ที่จำเป็น ได้แก่ เวลา, Socket GPIO

```
import time

import RPi.GPIO as GPIO
```

การเรียกใช้ BMC GPIO

```
GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)
```

กำหนดตัวแปร count

```
count = 0
```

กำหนดขาพินที่ใช้เชื่อมต่อตัวตรวจรู้ Ultrasonic HC-SR04 และขาพินที่ใช้เชื่อมต่อไฟแจ้งเตือน

```
GPIO_TRIGGER_A      = 23
GPIO_ECHO_A         = 24
GPIO_TRIGGER_B      = 27
GPIO_ECHO_B         = 22
GPIO_TRIGGER_C      = 17
GPIO_ECHO_C         = 18
GPIO_RED            = 16
GPIO_GREEN          = 20
GPIO_TRAIN          = 21
```

กำหนดสัญญาณอินพุตและสัญญาณเอ้าท์พุตให้ขาพิน

```
GPIO.setup(GPIO_TRIGGER_A,GPIO.OUT)
GPIO.setup(GPIO_ECHO_A,GPIO.IN)
GPIO.setup(GPIO_TRIGGER_B,GPIO.OUT)
GPIO.setup(GPIO_ECHO_B,GPIO.IN)
GPIO.setup(GPIO_TRIGGER_C,GPIO.OUT)
GPIO.setup(GPIO_ECHO_C,GPIO.IN)
GPIO.setup(GPIO_RED,GPIO.OUT)
GPIO.setup(GPIO_GREEN,GPIO.OUT)
GPIO.setup(GPIO_TRAIN,GPIO.OUT)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

กำหนดค่าเริ่มต้นให้ขา Trigger คือ False (Low)

```
GPIO.output(GPIO_TRIGGER_A, False)
GPIO.output(GPIO_TRIGGER_B, False)
GPIO.output(GPIO_TRIGGER_C, False)
```

การส่งสัญญาณ Pulse 10 us ไปให้ Trigger ตัวตรวจรู้ Ultrasonic ตัวที่ 1 เพื่อส่งออกไปกระทบวัตถุ

```
def measure_A():
  GPIO.output(GPIO_TRIGGER_A, True)
  time.sleep(0.00001)
  GPIO.output(GPIO_TRIGGER_A, False)
  startA = time.time()
  while GPIO.input(GPIO_ECHO_A)==0:
  startA = time.time()
  while GPIO.input(GPIO_ECHO_A)==1:
  stopA = time.time()
```

การคำนวณความยาว Pulse ที่เดินทางในอากาศคูณกับความเร็วของเสียง คือ 34000 เซนติเมตรต่อวินาที ระยะที่ได้เป็นระยะที่เดินทางจาก Trigger มากระทบวัตถุแล้วสะท้อนกลับเข้าที่ Echo ดังนั้นจึงต้องหาร 2 เพื่อให้ได้ระยะทางที่แท้จริง

```
elapsedA = stopA-startA
distanceA = (elapsedA * 34300)/2
return distanceA
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

การส่งสัญญาณ Pulse 10 us ไปให้ Trigger ตัวตรวจรู้ Ultrasonic ตัวที่ 2 เพื่อส่งออกไปกระทบวัตถุ

```
def measure_B():
  GPIO.output(GPIO_TRIGGER_B, True)
  time.sleep(0.00001)
  GPIO.output(GPIO_TRIGGER_B, False)
  startB = time.time()
  while GPIO.input(GPIO_ECHO_B)==0:
  startB = time.time()
  while GPIO.input(GPIO_ECHO_B)==1:
  stopB = time.time()
```

การคำนวณความยาว Pulse ที่เดินทางในอากาศคูณกับความเร็วของเสียง คือ 34000 เซนติเมตรต่อวินาที ระยะที่ได้เป็นระยะที่เดินทางจาก Trigger มากระทบวัตถุแล้วสะท้อนกลับเข้าที่ Echo ดังนั้นจึงต้องหาร 2 เพื่อให้ได้ระยะทางที่แท้จริง

```
elapsedB = stopB-startB
distanceB = (elapsedB * 34300)/2
return distanceB
```

การส่งสัญญาณ Pulse 10 us ไปให้ Trigger ตัวตรวจรู้ Ultrasonic ตัวที่ 3 เพื่อส่งออกไปกระทบวัตถุ

```
def measure_C():
  GPIO.output(GPIO_TRIGGER_C, True)
  time.sleep(0.00001)
  GPIO.output(GPIO_TRIGGER_C, False)
  startC = time.time()
  while GPIO.input(GPIO_ECHO_C)==0:
  startC = time.time()
  while GPIO.input(GPIO_ECHO_C)==1:
  stopC = time.time()
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

การคำนวณความยาว Pulse ที่เดินทางในอากาศคูณกับความเร็วของเสียง คือ 34000 เซนติเมตรต่อวินาที ระยะที่ได้เป็นระยะที่เดินทางจาก Trigger มากระทบวัตถุแล้วสะท้อนกลับเข้าที่ Echo ดังนั้นจึงต้องหาร 2 เพื่อให้ได้ระยะทางที่แท้จริง

```
elapsedC = stopC-startC
distanceC = (elapsedB * 34300)/2
return distanceC
```

นำค่าที่วัดได้จากตัวตรวจรู้ Ultrasonic ทั้ง 3 ตัว มาแสดงผล

```
try:
  while True:

    distanceA = measure_A()
    distanceB = measure_B()
    distanceC = measure_C()

    print "DistanceA : %d" % distanceA
    print "DistanceB : %d" % distanceB
    print "DistanceC : %d" % distanceC
    print "count : %d" % count
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

ถ้าตัวตรวจรู้ Ultrasonic ตัวที่ 1 วัดระยะทางได้น้อยกว่า 10 เซนติเมตร ไฟแจ้งเตือนแดงติด ไฟแจ้งเตือนสีเขียวดับ

```
if distanceA<10:
        GPIO.output(GPIO_RED, GPIO.HIGH)
        GPIO.output(GPIO_GREEN, GPIO.LOW)
```

ถ้าตัวตรวจรู้ Ultrasonic ที่ 2 วัดระยะทางได้น้อยกว่า 10 เซนติเมตร ไฟแจ้งเตือนสีแดงดับ ไฟแจ้งเตือนสีเขียวติด

```
if distanceB<10:
        GPIO.output(GPIO_RED, GPIO.LOW)
        GPIO.output(GPIO_GREEN, GPIO.HIGH)
```

ถ้าตัวตรวจรู้ Ultrasonic ตัวที่ 3 วัดระยะทางได้น้อยกว่า 10 เซนติเมตร count = count + 1

```
if distanceC<10:
        count = count + 1
```

ถ้าตัวตรวจรู้ Ultrasonic ตัวที่ 3 วัดระยะทางได้มากกว่า 10 เซนติเมตร count = 0 ไฟแจ้งเตือนรถไฟดับ

```
if distanceC>10:
        count = 0
        GPIO.output(GPIO_TRAIN, GPIO.LOW)
        time.sleep(0.5)
```

เมื่อ count มีค่าเท่ากับ 5 เคลียร์ count ไฟแจ้งเตือนรถไฟติด

```
while count == 5:
        count = 0
        GPIO.output(GPIO_TRAIN, GPIO.HIGH)
```

เคลียร์ GPIO

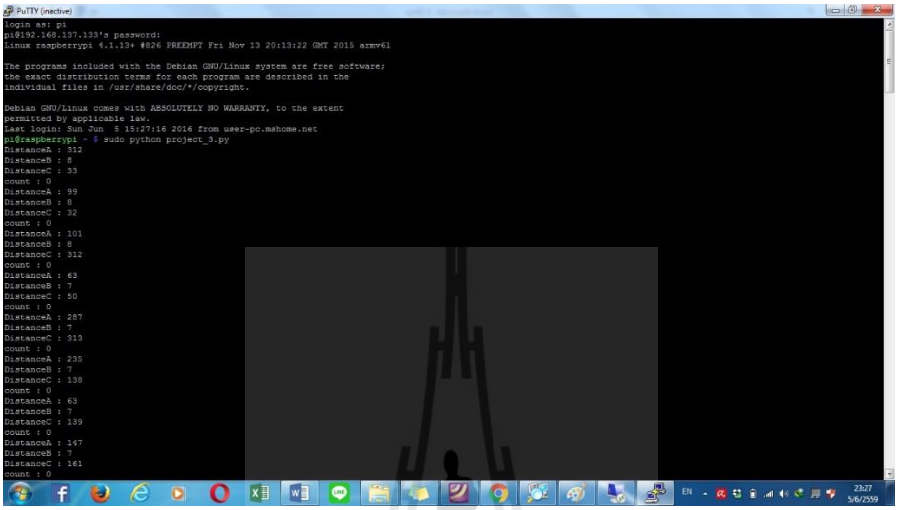
```
except KeyboardInterrupt:
    GPIO.cleanup()
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.3.5 การเริ่มต้นทำงานภาษา Python

เมื่อเขียนภาษา Python เรียบร้อยแล้วจะได้ไฟล์นามสกุล *.py สามารถนำมาเริ่มต้นทำงานบนโหมด SSH ในในโปรแกรม PuTTY ได้ โดยใช้คำสั่งดังนี้

```
sudo python project_3.py
```



รูป3.22 การรันบนโหมด SSH ในในโปรแกรม PuTTY

3.4 การใช้งานตัวตรวจรู้วัดระยะ Ultrasonic HC-SR04

ตัวตรวจรู้วัดระยะ Ultrasonic ใช้คลื่นเสียงที่เดินทางโดยอาศัยอากาศเป็นตัวกลางในการเคลื่อนที่ เสียงตามช่วงความยาวที่มนุษย์สามารถได้ยินจะเรียกว่าช่วงอะคูสติก ซึ่งอยู่ในช่วงความถี่ 20-20,000 Hz เสียงที่มีความถี่ที่ต่ำกว่าจะเรียกว่า Infrasound ส่วนเสียงที่มีความถี่สูงกว่าจะเรียกว่า Ultrasound ตัวตรวจรู้ Ultrasonic จะถูกออกแบบให้เมื่อกระทบกับวัตถุแล้วจะมีการสะท้อนกลับ คล้ายกับเรดาห์ จะนิยมใช้คลื่นเสียงความถี่สูงที่เรียกว่า Ultrasound ที่มนุษย์ไม่ได้ยินนี้มาใช้งาน ซึ่งค่อนข้างแม่นยำในระยะทางสั้นๆ

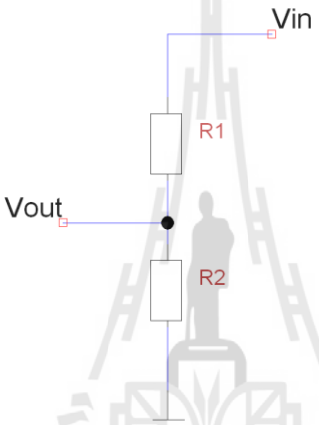
ตัวตรวจรู้ Ultrasonic โดยพื้นฐานประกอบด้วย เครื่องส่งสัญญาณ Ultrasonic (ลำโพง) เครื่องรับและวงจรควบคุมเมื่อส่งสัญญาณออกไปกระทบวัตถุที่มีความทึบสัญญาณจะถูกสะท้อนกลับมาที่เครื่องรับ แล้วประมวลผลสัญญาณด้วยวงจรประมวลผล ด้วยการใช้ความแตกต่างของเวลาที่ส่งออกกับที่รับได้ จะได้ระยะทางระหว่างตัวตรวจรู้กับวัตถุนั้นๆ

ตัวตรวจรู้วัดระยะ Ultrasonic HC-SR04 จะเชื่อมต่อกับบอร์ด Raspberry Pi โดยใช้ขาพิน GPIO จำนวน 4 ขา ได้แก่ กระแส 5 โวลต์ (VCC),Echo Pulse Output (ECHO),Trigger Pulse Input (TRIG) และกราวด์ (GND) ตามลำดับ

3.4.1 การแบ่งแรงดันให้ GPIO

เนื่องจากสัญญาณที่ส่งจากตัวตรวจรู้วัดระยะ Ultrasonic กลับมาที่พิน GPIO ของบอร์ด Raspberry Pi มีความแรงของสัญญาณที่ 5 โวลต์ แต่ที่พินของ GPIO รองรับอยู่ที่ 3.3 โวลต์ จึงต้องทำการลดแรงดันของสัญญาณลง เพื่อเป็นการป้องกันอันตรายให้กับบอร์ด Raspberry Pi โดยจะใช้การคำนวณจากกฎการแบ่งแรงดันดังนี้

การแบ่งแรงดันจะใช้ตัวต้านทาน 2 ตัวมาต่อแบบอนุกรมกันกับแรงดันอินพุต (V_{in}) เพื่อลดแรงดันเอาต์พุต (V_{out}) ให้เหลือ 3.3 โวลต์ ในวงจรแรงดันอินพุตจะเป็นแรงดันจาก ECHO จะได้ตัวอย่างการแบ่งแรงดัน ตามรูป 3.19



รูป3.23 ตัวอย่างการแบ่งแรงดัน

http://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi

จากสมการ[1]

$$V_{out} = \frac{R_2}{R_1 + R_2} V_{in}$$

$$\frac{V_{out}}{V_{in}} = \frac{R_2}{R_1 + R_2}$$

[1] การหาค่า R_2 แหล่งข้อมูล : http://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi

เมื่อ input voltage = 5V, output voltage = 3.3V, R_1 = 1000 Ω

$$\frac{3.3}{5} = \frac{R_2}{1000\Omega + R_2}$$

$$0.66 = \frac{R_2}{1000\Omega + R_2}$$

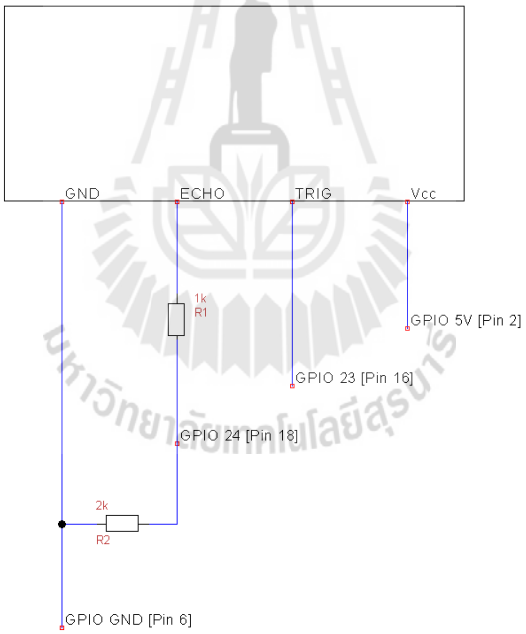
$$0.66(1000 + R_2) = R_2$$

$$660 + 0.66R_2 = R_2$$

$$0.34R_2 = 660$$

$$R_2 = 1941\Omega \approx 2000\ \Omega$$

ดังนั้น เมื่อใช้ R_1 =1 kΩ จะได้ R_2 =2 kΩ เมื่อนำมาใช้กับตัวตรวจรู้จะได้ตัวอย่างวงจร ตามรูป3.22



รูป3.24 ตัวอย่างวงจรแบ่งแรงดัน

http://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi

3.5 บทสรุป

เนื้อหาในบทนี้ได้รวบรวมวิธีการ การออกแบบและติดตั้งตัวตรวจรู้ Ultrasonic การติดตั้งระบบปฏิบัติการ Raspbian ให้บอร์ด Raspberry pi, การเชื่อมต่อและแชร์อินเตอร์เน็ตจากคอมพิวเตอร์ให้กับ Raspberry pi, การใช้งาน SSH ด้วยโปรแกรม PuTTY, การเขียนโปรแกรมโดยใช้บอร์ด Raspberry pi, การรันโค้ดภาษา Python, และการใช้งานตัวตรวจรู้วัดระยะ Ultrasonic HC-SR04 เพื่อใช้ในการออกแบบและการจำลอง

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 4

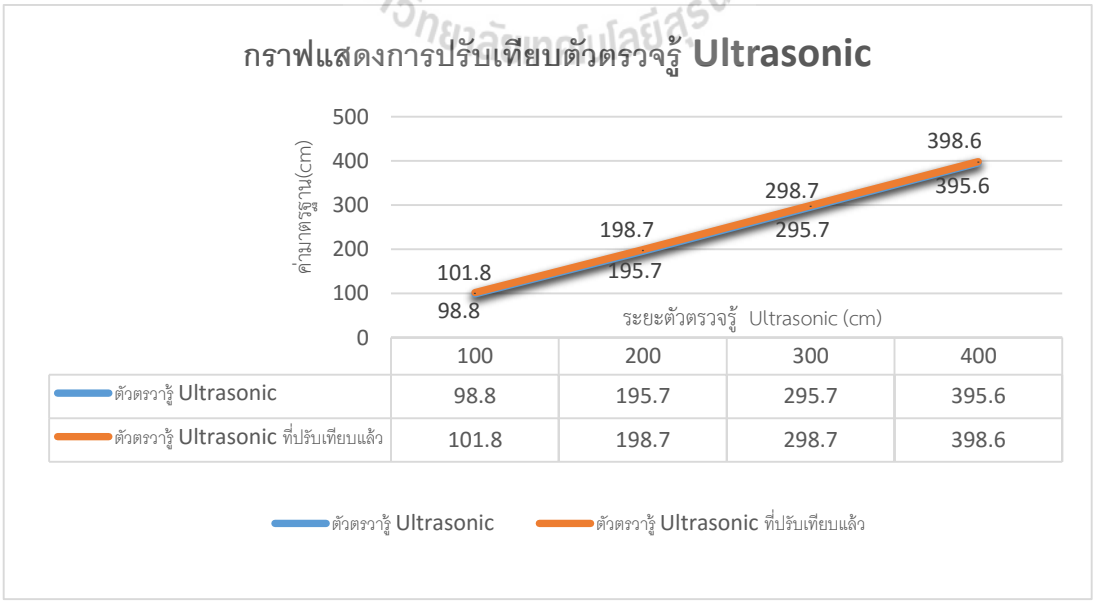
การทดสอบชิ้นงาน

4.1 การปรับเทียบตัวตรวจรู้ Ultrasonic

ในการวัดระยะทางด้วยตัวตรวจรู้ Ultrasonic ระยะทางที่วัดได้มีความผิดพลาดจากค่ามาตรฐาน ดังนั้นจะต้องทำการปรับเทียบ (Calibration) เพื่อให้ระยะทางที่วัดด้วยตัวตรวจรู้ Ultrasonic มีค่าใกล้เคียงกับค่ามาตรฐานมากที่สุด

ตารางที่ 4.1 แสดงการวัดระยะโดยใช้ตัวตรวจรู้ Ultrasonic กับระยะทางจริง

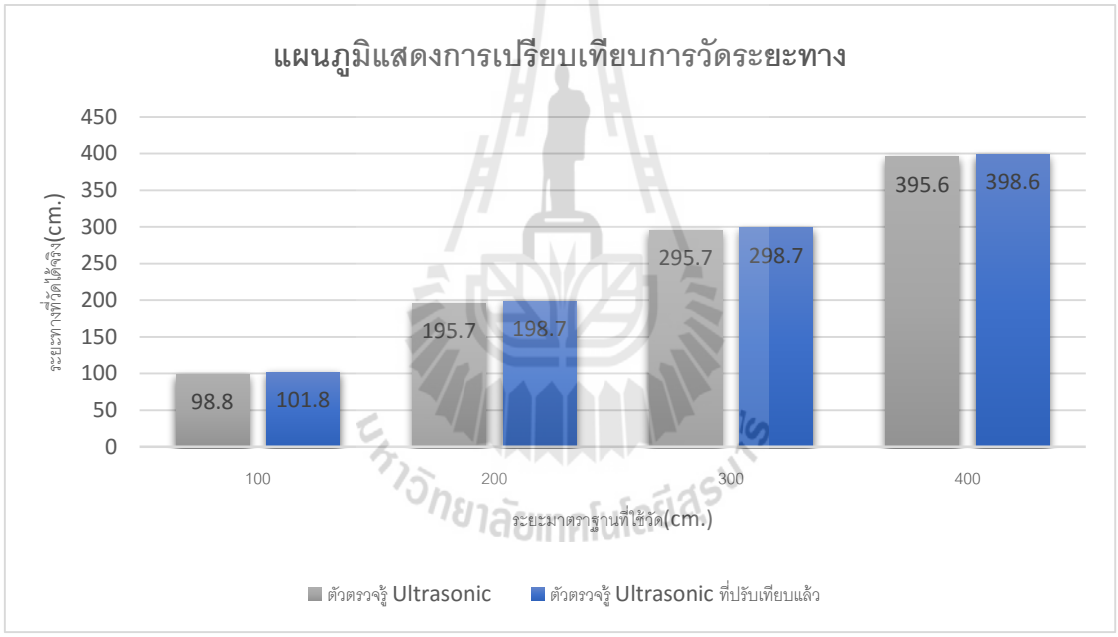
ระยะทางจริง	ตัวตรวจรู้ Ultrasonic
100	98.8
200	195.7
300	295.7
400	395.6



รูปที่ 4.1 กราฟแสดงการ ปรับเทียบตัวตรวจรู้ Ultrasonic

ตารางที่ 4.2 แสดงค่า R^2 (R-Squared)

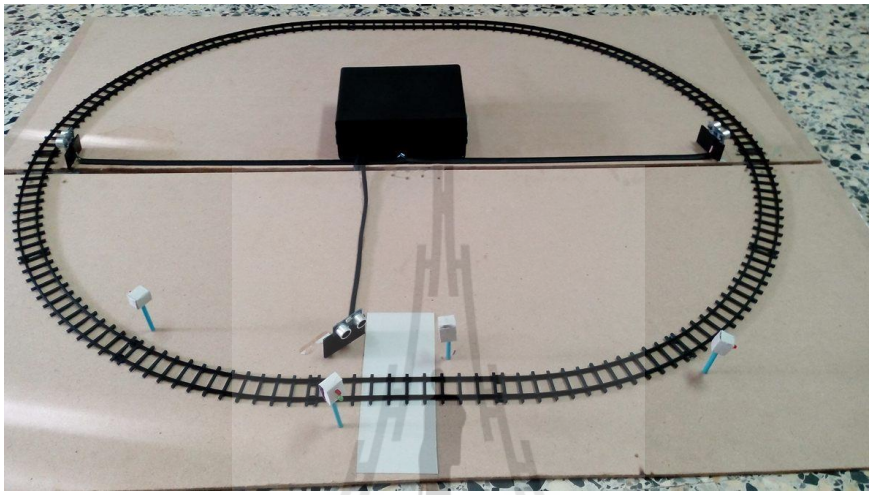
ระยะทาง	ตัวตรวจรู้ Ultrasonic	ตัวตรวจรู้ Ultrasonic ที่ปรับเทียบแล้ว	R²ก่อนการปรับเทียบ	R²หลังการปรับเทียบ
100	98.8	101.8	0.9761	0.9649
200	195.7	198.7	0.9574	0.9870
300	295.7	298.7	0.9715	0.9914
400	395.6	398.6	0.9781	0.9930



รูปที่ 4.2 แผนภูมิการเปรียบการวัดระยะทาง

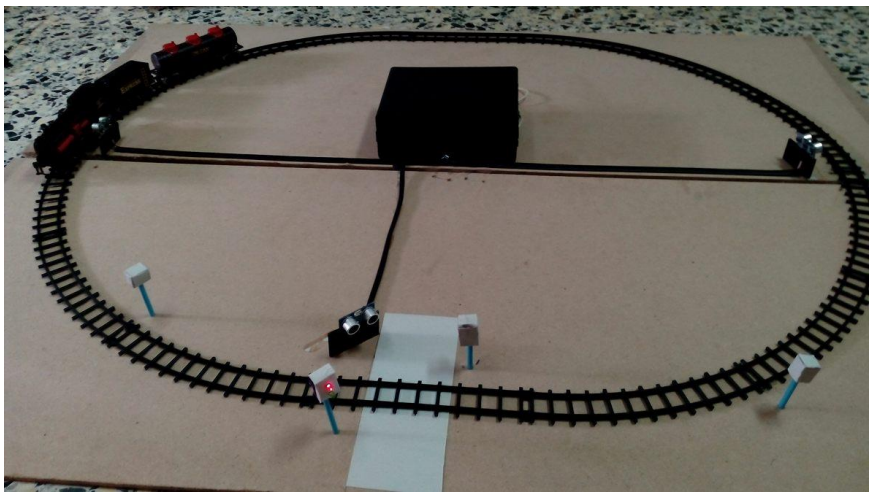
4.2 การทดสอบชิ้นงาน

คณะทำงานได้ทำการสร้างแบบจำลองทางลักผ่าน เพื่อจำลองสถานการณ์ทั้งเมื่อมีรถไฟวิ่งผ่านแล้วระบบไฟทำการเตือน และเตือนไปยังรถไฟ กรณีมีสิ่งกีดขวางบนรางรถไฟ ในบริเวณทางลักผ่าน



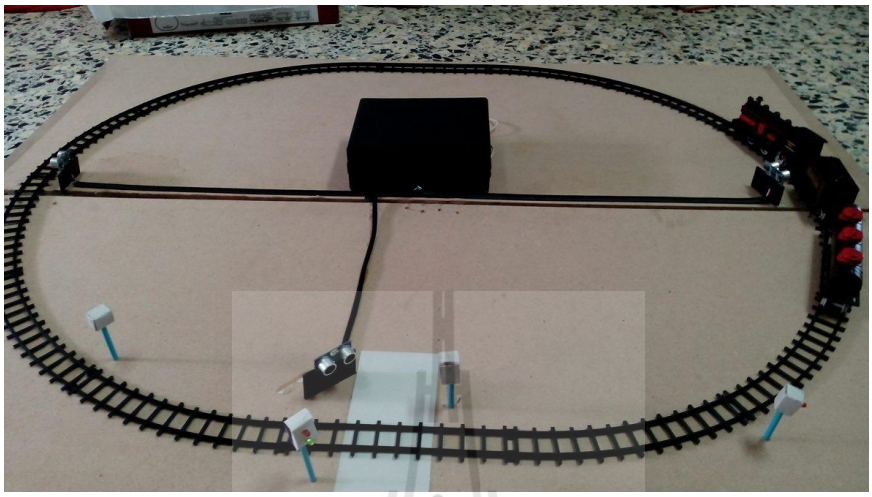
รูปที่ 4.3 แบบจำลองทางข้ามรถไฟ

กรณีมีรถไฟวิ่งผ่านตัวตรวจรู้ Ultrasonic ตัวที่ 1 ระยะทางที่วัดได้จากตัวตรวจรู้ตัวที่ 1 จะมีค่าน้อยกว่า 10 เซนติเมตร จากคำสั่ง ถ้าตรวจรู้ Ultrasonic ตัวที่ 1 วัดระยะทางได้น้อยกว่า 10 เซนติเมตร ไฟแจ้งเตือนสีแดงจะติด ทำให้รถยนต์ที่สัญจรข้ามทางลักผ่านรู้ได้ว่ารถไฟกำลังมา และสามารถหยุดรถก่อนถึงทางข้ามรถไฟได้



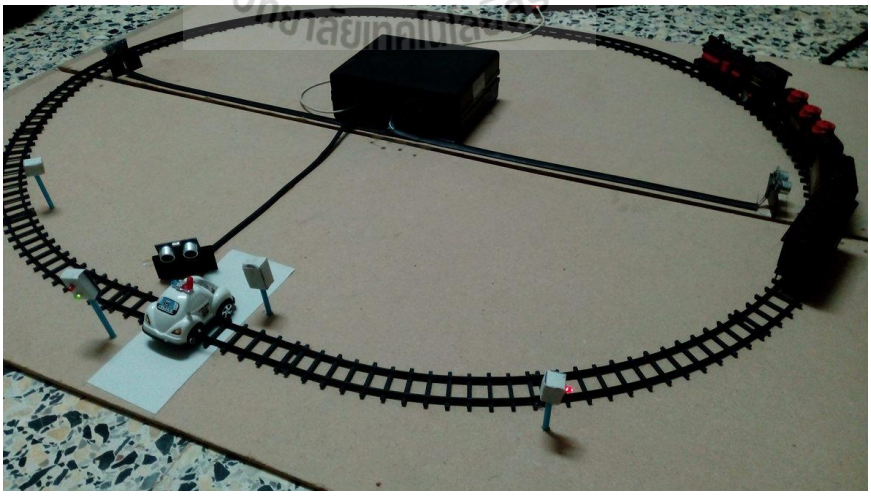
รูปที่ 4.4 รถไฟวิ่งผ่านตัวตรวจรู้ Ultrasonic ตัวที่ 1

กรณีมีรถไฟวิ่งผ่านตัวตรวจรู้ Ultrasonic ตัวที่ 2 ระยะทางที่วัดได้จากตัวตรวจรู้ตัวที่ 2 จะมีค่าน้อยกว่า 10 เซนติเมตร จากคำสั่ง ถ้าตรวจรู้ Ultrasonic ตัวที่ 2 วัดระยะทางได้น้อยกว่า 10 เซนติเมตร ไฟแจ้งเตือนสีเขียวจะติด ทำให้รถยนต์ที่สัญจรข้ามทางลักผ่านรู้ได้ว่าไม่มีรถไฟมา และสามารถข้ามทางข้ามรถไฟได้



รูปที่ 4.5 รถไฟวิ่งผ่านตัวตรวจรู้ Ultrasonic ตัวที่ 2

กรณีที่มีรถค้างอยู่บนทางลักผ่าน ระยะทางที่วัดได้จากตัวตรวจรู้ตัวที่ 3 จะมีค่าน้อยกว่า 10 เซนติเมตร จากคำสั่ง ถ้าตรวจรู้ Ultrasonic ตัวที่ 3 วัดระยะทางได้น้อยกว่า 10 เซนติเมตร ตัวแปร count จะเพิ่มค่าขึ้นไปเรื่อยๆ เมื่อตัวแปร count มีค่าเท่ากับ 5 แสดงว่า มีสิ่งกีดขวางอยู่บนรางรถไฟ ไฟแจ้งเตือนรถไฟก็จะติด ทำให้รถไฟที่วิ่งมารู้ว่ามีสิ่งกีดขวางทางรถไฟอยู่ ให้รถไฟชะลอความเร็ว และระวังเพื่อไม่ให้เกิดอุบัติเหตุ

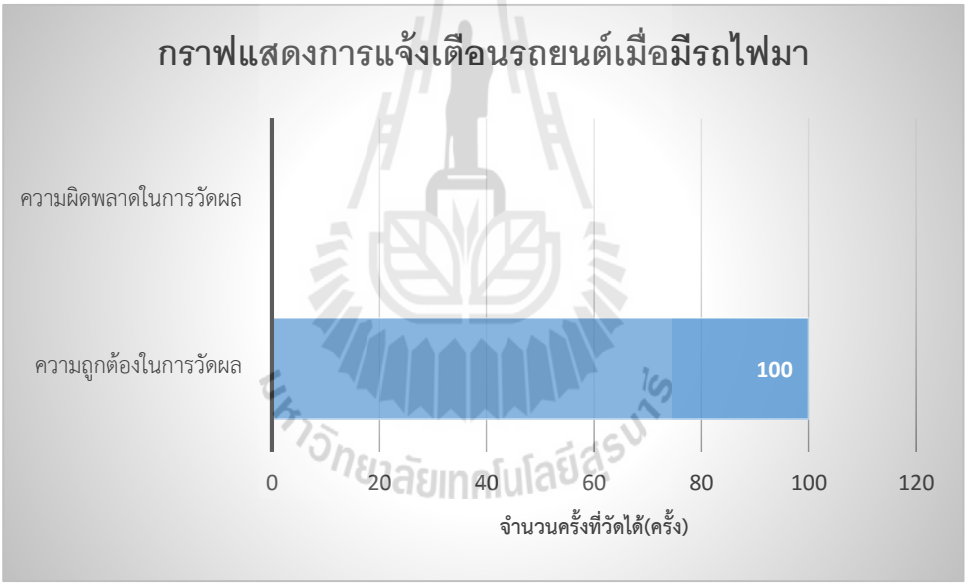


รูปที่ 4.6 มีรถค้างอยู่บนทางลักผ่าน

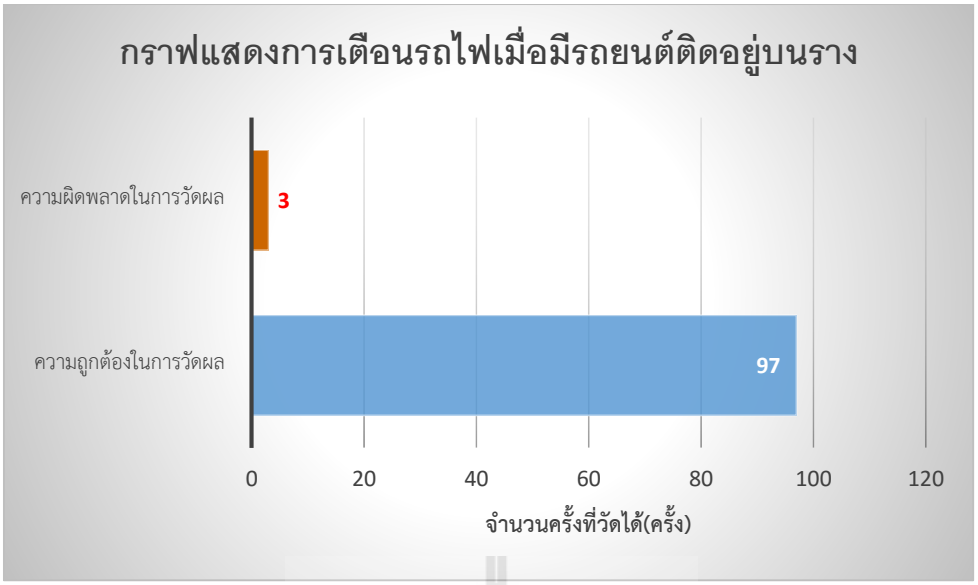
จากการทดสอบจากแบบจำลอง จำนวน 100 ครั้ง ได้ผลที่เป็นไปตามตาราง ดังนี้

ตารางที่ 4.3 แสดงผลการทดสอบจากแบบจำลอง

กรณีในการแจ้งเตือน	จำนวนครั้งในการวัดผล	ผลการวัด	
		ความถูกต้องในการวัดผล	ความผิดพลาดในการวัดผล
แจ้งเตือนรถยนต์เมื่อมีรถไฟมา	100	100	0
แจ้งเตือนรถไฟเมื่อมีรถยนต์ติดอยู่บนราง	100	97	3



รูปที่ 4.7 กราฟแสดงการแจ้งเตือนรถยนต์เมื่อมีรถไฟมา



รูปที่ 4.8 กราฟแสดงการแจ้งเตือนรถไฟเมื่อมีรถยนต์ค้างอยู่บนราง

4.3 บทสรุป

จากการปรับเทียบ(Calibration) ตัวตรวจรู้ Ultrasonic พบว่า ค่าที่วัดได้จะมีค่าใกล้เคียงกับค่ามาตราฐานมากกว่าก่อนการปรับเทียบ ทำให้การตรวจจับรถไฟและรถยนต์มีความแม่นยำมากขึ้น ส่งผลให้ระบบการแจ้งเตือนมีความน่าเชื่อถือ

ในการทดสอบชิ้นงาน จะต้องทำการทดสอบจำนวนหลายครั้ง เพื่อแสดงให้เห็นว่าระบบแจ้งเตือนนี้สามารถทำงานได้จริง และสามารถตรวจสอบความผิดพลาดของระบบแจ้งนี้ได้

จากการทดสอบชิ้นงานบนแบบจำลองทางลักผ่าน จะเห็นได้ว่าอุปกรณ์ทำงานได้ตามจุดประสงค์ที่วางไว้ พร้อมทั้งแจ้งเตือนว่ากำลังมีรถไฟมา และแจ้งเตือนกรณีที่มีรถค้างอยู่บนรางรถไฟ ทำให้สามารถเพิ่มความปลอดภัยบริเวณทางลักผ่านได้

บทที่ 5
บทสรุปของโครงงาน

5.1 ปัญหาและการแก้ไข

1. ตัวตรวจรู้ Ultrasonic ที่อยู่ใกล้กัน จะส่งคลื่นมารบกวนกัน ทำให้การวัดระยะทางผิดพลาด
   <u>การแก้ไข</u> ติดตั้งตัวตรวจรู้ Ultrasonic ให้ทิศทางของลำคลื่นไม่เกิดซ้อนทับกัน

5.2 ข้อเสนอแนะ

ในการติดตั้งตัวตรวจรู้ Ultrasonic จะเกิดการรบกวนกันของคลื่น จึงควรคำนวณระยะในการติดตั้ง และศึกษามุมในการสะท้อนของคลื่น เพื่อลดปัญหาที่อาจจะเกิดขึ้น

5.3 แนวทางในการพัฒนาต่อไป

โครงงานชิ้นนี้เป็นตัวต้นแบบระบบเตือนภัยทางลักผ่านโดยใช้บอร์ด Raspberry pi และตัวตรวจรู้ Ultrasonic เพียง 3 ตัว โดย 2 ตัวใช้เพื่อตรวจจับการมาของรถไฟเพื่อไปแจ้งเตือนเตือนผู้สัญจรบริเวณทางลักผ่าน และอีก 1 ตัวใช้ตรวจจับรถยนต์ที่ค้างอยู่บนทางข้ามรถไฟเพื่อเตือนรถไฟ เมื่อมีรถหรือสิ่งกีดขวางติดอยู่บนรางรถไฟ ทั้งนี้สามารถพัฒนาโดยการแจ้งเตือนบนเซิร์ฟเวอร์ไปยังหน่วยงานที่เกี่ยวข้องในบริเวณใกล้เคียงทางลักผ่านนั้น ในกรณีที่มีรถยนต์ค้างอยู่บนรางรถไฟหรือกรณีเกิดอุบัติเหตุ เพื่อเป็นการต่อยอดผลงานต่อไป

5.4 บทสรุป

เนื้อหาในบทนี้เป็นการกล่าวถึงบทสรุปของโครงงานระบบแจ้งเตือนทางข้ามรถไฟ โดยใช้บอร์ด Raspberry Pi และตัวตัวตรวจรู้ Ultrasonic จะประกอบไปด้วยปัญหาที่พบในขณะดำเนินงาน วิธีการแก้ไข ข้อเสนอแนะและวิธีพัฒนาโครงงานต่อไป

โครงงานระบบแจ้งเตือนทางข้ามรถไฟ โดยใช้บอร์ด Raspberry Pi และตัวตรวจรู้ Ultrasonic มีส่วนประกอบดังนี้

1. บอร์ดคอมพิวเตอร์ขนาดเล็ก Raspberry Pi ทำหน้าที่เป็น Controller
2. ตัวตรวจรู้ Ultrasonic ทำหน้าที่เป็นตัวตรวจจับวัตถุ
3. แบบจำลองสถานที่ทางข้ามรถไฟ

สรุปผลของโครงงาน จากที่บอร์ด Raspberry Pi สามารถตรวจสอบการมาของรถไฟ และการค้างของรถยนต์อยู่บนรางรถไฟได้ ทำให้เพิ่มความปลอดภัยในการสัญจรข้ามบริเวณทางลักผ่าน และป้องกันการเกิดอุบัติเหตุการชนกันของรถไฟกับรถยนต์ที่ค้างอยู่บนรางรถไฟได้

โครงงานเรื่องนี้ คณะผู้จัดทำมิได้ส่งเสริมให้สัญจรข้ามบริเวณทางลักผ่าน แต่จัดทำขึ้นเพื่อเพิ่มความปลอดภัยในการสัญจรข้ามบริเวณทางลักผ่านเท่านั้น

มหาวิทยาลัยเทคโนโลยีสุรนารี

บรรณานุกรม

วิกิพิเดีย. 2016. รถไฟ. (ออนไลน์). แหล่งที่มา: https://th.wikipedia.org/wiki/รถไฟ. 6 มิถุนายน 2559

วิกิพิเดีย. 2016. ขนาดความกว้างรางรถไฟ. (ออนไลน์). แหล่งที่มา: https://th.wikipedia.org/wiki/ขนาดความกว้างรางรถไฟ. 6 มิถุนายน 2559

วิกิพิเดีย. 2016. รถไฟความเร็วสูง. (ออนไลน์). แหล่งที่มา: https://th.wikipedia.org/wiki/รถไฟความเร็วสูง. 6 มิถุนายน 2559

ศูนย์กลางข้อมูลข่าวสารความปลอดภัยทางถนน. 2016. ทางลักผ่าน. (ออนไลน์). แหล่งที่มา: http://www.accident.or.th/datacenter/index.php/2015-10-13-04-59-28/88-2015-10-16-03-27-42.6 มิถุนายน 2559

ศูนย์กลางข้อมูลข่าวสารความปลอดภัยทางถนน. 2016. กฎหมายที่เกี่ยวข้อง. (ออนไลน์). แหล่งที่มา: http://www.accident.or.th/datacenter/index.php/2015-10-13-04-59-28/88-2015-10-16-03-27-42.6 มิถุนายน 2559

วิกิพิเดีย. 2016. ภาษาไพทอน. (ออนไลน์). แหล่งที่มา: https://th.wikipedia.org/wiki/ภาษาไพทอน. 6 มิถุนายน 2559

_____. Ultrasonic Sensor Module (HC-SR04). Retrieved June7,2016, from arduitronics Web site: http://www.arduitronics.com/product/20/ultrasonic-sensor-module-hc-sr04

ภาคผนวก

มหาวิทยาลัยเทคโนโลยีสุรนารี

Ultrasonic Ranging Module HC - SR04

□ Product features:

Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:

(1) Using IO trigger for at least 10us high level signal,

(2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.

(3) IF the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning.

Test distance = (high level time×velocity of sound (340M/S) / 2,

□ Wire connecting direct as following:

5V Supply

Trigger Pulse Input

Echo Pulse Output

0V Ground

Electric Parameter

Working Voltage	**DC 5 V**
Working Current	**15mA**
Working Frequency	**40Hz**
Max Range	**4m**
Min Range	**2cm**

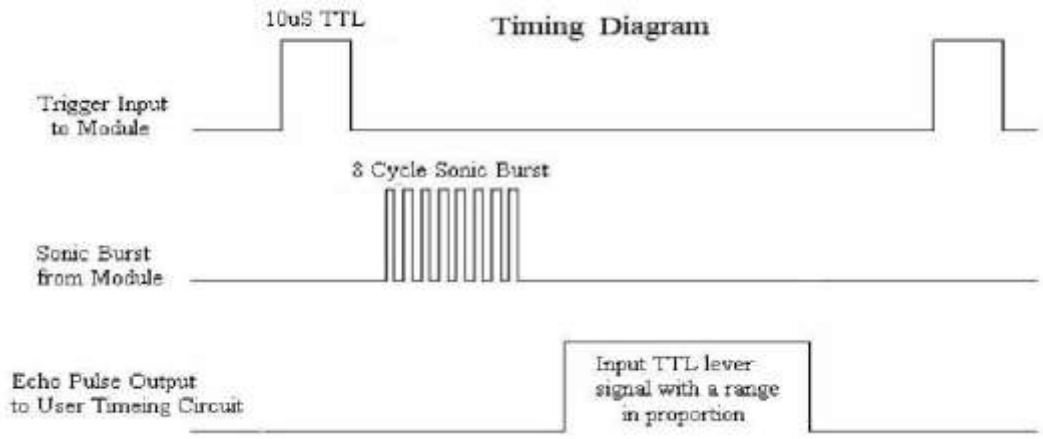
MeasuringAngle	**15 degree**
Trigger Input Signal	**10uS TTL pulse**
Echo Output Signal	**Input TTL lever signal and the range in proportion**
Dimension	**45*20*15mm**

Vcc Trig Echo GND

มหาวิทยาลัยเทคโนโลยีสุรนารี

Timing diagram

The Timing diagram is shown below. You only need to supply a short 10uS pulse to the trigger input to start the ranging, and then the module will send out an 8 cycle burst of ultrasound at 40 kHz and raise its echo. The Echo is a distance object that is pulse width and the range in proportion .You can calculate the range through the time interval between sending trigger signal and receiving echo signal. Formula: uS / 58 = centimeters or uS / 148 =inch; or: the range = high level time * velocity (340M/S) / 2; we suggest to use over 60ms measurement cycle, in order to prevent trigger signal to the echo signal.



□ Attention:

□ The module is not suggested to connect directly to electric, if connected electric, the GND terminal should be connected the module first, otherwise, it will affect the normal work of the module.

□ When tested objects, the range of area is not less than 0.5 square meters and the plane requests as smooth as possible, otherwise ,it will affect the results of measuring.

www.Elecfreaks.com

มหาวิทยาลัยเทคโนโลยีสุรนารี

Raspberry Pi

MODEL B+

Product Name	**Raspberry Pi Model B+**
Product Description	The Raspberry Pi Model B+ incorporates a number of enhancements and new features. Improved power consumption, increased connectivity and greater IO are among the improvements to this powerful, small and lightweight ARM based computer.

Specifications

Chip	Broadcom BCM2835 SoC
Core architecture	ARM11
CPU	700 MHz Low Power ARM1176JZFS Applications Processor
GPU	Dual Core VideoCore IV® Multimedia Co-Processor Provides Open GL ES 2.0, hardware-accelerated OpenVG, and 1080p30 H.264 high-profile decode Capable of 1Gpixel/s, 1.5Gtexel/s or 24GFLOPs with texture filtering and DMA infrastructure
Memory	512 MB SDRAM
Operating System	Boots from Micro SD card, running a version of the Linux operating system
Dimensions	85 x 56 x 17mm
Power	Micro USB socket 5V, 2A

Connectors:

Ethernet	10/100 BaseT Ethernet socket
Video Output	HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC)
Audio Output	3.5 mm jack, HDMI
USB	4 x USB 2.0 Connector
GPIO Connector	40- pin 2.54 mm (100 mil) expansion header: 2x20 strip Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines
Camera Connector	15- pin MIPI Camera Serial Interface (CSI -2)
JTAG	Not populated
Display Connector	Display Serial Interface (DSI) 15 way flat flex cable connector with two data lanes and a clock lane
Memory Card Slot	SDIO

ประวัติผู้เขียน

นางสาววิไลลักษณ์ กุลรัตน์ เกิดเมื่อวันที่ 29 เมษายน พ.ศ. 2535 ภูมิลำเนาอยู่ที่ ตำบลลำดวน อำเภอลำดวน จังหวัดสุรินทร์ สำเร็จการศึกษาระดับมัธยมปลายจากโรงเรียนสุรพินท์พิทยา อำเภอลำดวน จังหวัดสุรินทร์ เมื่อปี พ.ศ. 2554 ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

นายจักรพงศ์ กาสี เกิดเมื่อวันที่ 8 พฤศจิกายน พ.ศ. 2536 ภูมิลำเนาอยู่ที่ ตำบลม่วงหวาน อำเภอน้ำพอง จังหวัดขอนแก่น สำเร็จการศึกษาระดับมัธยมปลายจากโรงเรียนขอนแก่นวิทยายน 3 อำเภอเมือง จังหวัดขอนแก่น เมื่อปี พ.ศ. 2554 ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

นายสุรสิทธิ์ ทุมวรรณ์ เกิดเมื่อวันที่ 6 กุมภาพันธ์ พ.ศ. 2536 ภูมิลำเนาอยู่ที่ ตำบลสร้างก่อ อำเภอกุดจับ จังหวัดอุดรธานี สำเร็จการศึกษาระดับมัธยมปลายจากโรงเรียนอุดรพิชัยรักษ์พิทยา อำเภอเมืองจังหวัดอุดรธานี เมื่อปี พ.ศ. 2554 ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี