ระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์
ALARM SYSTEM FOR CHILDREN TRAPPED IN CAR

โดย

นายศุภธนา อุดมกิจโอฬาร

นายสิรวิชญ์ หอมเอนก

นายสุทธิพงษ์ สมัครสมาน

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

ระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์
ALARM SYSTEM FOR CHILDREN TRAPPED IN CAR

โดย

นายศุภธนา	อุดมกิจโอฬาร	56011242
นายสิรวิชญ์	หอมเอนก	56011318
นายสุทธิพงษ์	สมัครสมาน	56011342

อาจารย์ที่ปรึกษา
ผศ.ดร. สมเกียรติ ฤกษ์วรัญญู

อาจารย์ที่ปรึกษาร่วม
รศ.ดร.พิพัฒน์ พรหมมี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

ผ่านการตรวจรูปเล่มแล้ว
(.....)
อาจารย์ที่ปรึกษา
...../...../.....

ผ่านการตรวจชิ้นงานแล้ว
(.....)
กรรมการผู้ตรวจชิ้นงาน
...../...../.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น และห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2559
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง ระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์
ALARM SYSTEM FOR CHILDREN TRAPPED IN CAR

ผู้จัดทำ

1.	นายศุภธนา	อุดมกิจโอฬาร	56011242
2.	นายสิรวิชญ์	หอมเอนก	56011318
3.	นายสุทธิพงษ์	สมัครสมาน	56011342

..... อาจารย์ที่ปรึกษา
(ผศ.ดร. สมเกียรติ ฤกษ์วรัญญู)

..... อาจารย์ที่ปรึกษาร่วม
(รศ.ดร.พิพัฒน์ พรหมมี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การดำเนินปริญญานิพนธ์ “ระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์” จะไม่สามารถสำเร็จลุล่วงไปได้ด้วยดีได้เลย หากขาดการสนับสนุน และกำลังใจจากหลายๆ ฝ่าย เช่น ผศ.ดร.สมเกียรติ ฤกษ์วรัญญู อาจารย์ที่ปรึกษาโครงงาน รศ.ดร.พิพัฒน์ พรหมมี สำหรับคำปรึกษา คำแนะนำและแนวทางการแก้ไขเมื่อเกิดปัญหาในการจัดทำปริญญานิพนธ์ รวมทั้งสนับสนุนสถานที่ เครื่องมือ และอุปกรณ์ต่าง ๆ ที่จำเป็นต้องใช้ในระหว่างการจัดทำปริญญานิพนธ์ คุณพ่อ คุณแม่ และครอบครัวที่คอยห่วงใย และมอบกำลังใจให้ในยามที่เกิดปัญหา รวมทั้งคอยสนับสนุนการทำงานจนเสร็จสมบูรณ์ พี่ๆ และเพื่อนๆ ทุกคนในห้อง T112 รวมทั้งห้อง ECC510 ที่ได้คอยให้คำแนะนำคำปรึกษาและให้ความรู้เพิ่มเติมเกี่ยวกับการจัดทำปริญญานิพนธ์

ผู้จัดทำขอขอบพระคุณทุก ๆ ท่านเป็นอย่างสูง ณ ที่นี้ ที่ได้ช่วยแนะนำและให้คำปรึกษาทำให้การจัดทำปริญญานิพนธ์ในครั้งนี้สำเร็จลุล่วงไปได้ด้วยดี

นายศุภธนา อุดมกิจโอฬาร
นายสิรวิชญ์ หอมเอนก
นายสุทธิพงษ์ สมัครสมาน
ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์
ALARM SYSTEM FOR CHILDREN TRAPPED IN CAR

โดย นายศุภธนา อุดมกิจโอฬาร 56011242
นายสิรวิชญ์ หอมเอนก 56011318
นายสุทธิพงษ์ สมัครสมาน 56011342

อาจารย์ที่ปรึกษา ผศ.ดร. สมเกียรติ ฤกษ์วรัญญู
อาจารย์ที่ปรึกษาร่วม รศ.ดร. พิพัฒน์ พรหมมี

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการสร้างระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์ เพื่อป้องอุบัติเหตุจากการติดอยู่ในรถของเด็ก ด้วยความรู้เท่าไม่ถึงการณ์ของคนขับรถหรือผู้ปกครองที่ปล่อยให้เด็กหรือบุตรหลานนอนหลับอยู่บนรถ ซึ่งระบบออกแบบให้มีการป้องกันและแจ้งเตือน โดยมีการใช้ระบบอาร์เอฟไอดีในการตรวจนับคนขึ้นลงรถยนต์ ซึ่งถ้ามีการขึ้นลงของคนภายในรถยนต์ไม่เท่ากันจะมีการส่งเสียงแจ้งเตือนผ่านทางไซเรนของรถยนต์ กรณีหากไม่ได้ยินเสียงไซเรนของรถยนต์จะทำการติดเซ็นเซอร์ตรวจจับการเคลื่อนไหวไว้ภายในรถ เพื่อตรวจสอบอีกครั้งเมื่อมีการเคลื่อนไหวภายในรถยนต์ เซ็นเซอร์ตรวจจับการเคลื่อนไหวจะการส่งเสียงร้องทันทีพร้อมทั้งเปิดกระจกรถยนต์ในกรณีที่เป็นกระจกไฟฟ้า และโทรหาเจ้าของรถทันทีเพื่อแจ้งเตือน โดยอุปกรณ์ทั้งหมดจะใช้การควบคุมผ่านไมโครคอนโทรลเลอร์

ABSTRACT

This project invents an alert system for children trapped inside the car. It can prevent the tragedy accident in case of forgotten the children in the car. The system is designed by using a microcontroller, frequency identification (RFID), GPRS module, alerting subsystem and several sensors. The system uses RFID to count the total of children in the car. If the discrepancy of children rest in the car has been detected, the system will send alert via the siren. If user cannot hear the sound, the system will rechecked again by using motion sensor. After that the system will be open the window automatically if motion has been detected. The system also sends the message to user for the situation.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
กิตติกรรมประกาศ		I
บทคัดย่อ		II
สารบัญ		III
สารบัญรูป		V
บทที่ 1	**บทนำ**	**1**
	1.1 ความเป็นมาและความสำคัญของปัญหา	1
	1.2 วัตถุประสงค์	1
	1.3 ขอบเขตของโครงงาน	1
บทที่ 2	**ทฤษฎีและหลักการที่เกี่ยวข้อง**	**3**
	2.1 Arduino	3
	2.2 RFID Card Reader and Detector Module	5
	2.3 LCD Module	5
	2.4 RFID (Radio Frequency Identification)	6
	2.5 อุปกรณ์พื้นฐานทางไฟฟ้ารถยนต์	14
	2.6 PIR Motion Sensor	18
	2.7 GSM Module SIM800L	21
บทที่ 3	**การออกแบบและการจัดทำปริญญานิพนธ์**	**22**
	3.1 การออกแบบ	22
	3.2 เครื่องมือที่ใช้ในการทดลอง	27
	3.3 การจัดเก็บผลการทดลอง	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

		หน้า
บทที่ 4	**ผลการทดลอง**	**33**
	4.1 ทดสอบ สถานะ IN และ OUT เมื่อมีการแตะ Card เข้ามา	33
	4.2 ทดสอบเมื่อ มีการแตะการ์ดซ้ำ	34
	4.3 ทดสอบเมื่อดับสวิตช์	35
	4.4 ทดสอบการเขียนลงบน SD CARD	36
	4.5 ทดสอบ PIR สถานะทำงาน และ ไม่ทำงาน	37
	4.6 ทดสอบอุปกรณ์รวมภายในรถยนต์	38
บทที่ 5	**สรุปผลและข้อเสนอแนะ**	**42**
	5.1 สรุปผล	42
	5.2 ข้อเสนอแนะ	42
บรรณานุกรม		**43**
ภาคผนวก	ชุดคำสั่งการทำงานของ MICROCONTROLLER	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่

1.1	บล็อกไดอะแกรมการทำงานของระบบ	2
2.1	ส่วนประกอบของ Arduino board รุ่น MEGA 2560 R3	3
2.2	Arduino MEGA	4
2.3	RFID Card Reader and Detector Module	5
2.4	LCD Module	5
2.5	ภาพรวมของระบบ RFID	7
2.6	RFID แท็กในรูปแบบต่างๆ	7
2.7	บล็อกไดอะแกรมของ Passive Tag	8
2.8	ตัวอย่าง Active Tag ที่มีแบตเตอรี่ Lithium 2 ก้อนอยู่ภายนอก	9
2.9	โครงสร้างภายในเครื่องอ่าน	9
2.10	รูปตัวอย่างเครื่องอ่านแบบต่างๆ	10
2.11	รูปคลื่นของสัญญาณระหว่างแท็กและเครื่องอ่านแบบ AM	10
2.12	ซ้ายลูปตัวนำที่ถูกเหนี่ยวนำด้วยสนามแม่เหล็ก ขวาวงจรทีสมมูลกับรูปทางซ้ายมือ	11
2.13	การสัญญาณรูปคลื่นที่เข้ารหัสแบบแมนเชสเตอร์ (Manchester)	12
2.14	ตัวอย่างของอัลกอริทึมในการป้องกันการชนของข้อมูล(Anti-Collision) ในแท็ก	13
2.15	ความถี่ย่านที่ระบบ RFID ถูกใช้งาน	13
2.16	ลักษณะของฟิวส์แบบหลอด	15
2.17	ลักษณะของฟิวส์แบบเสียบ	15
2.18	ลักษณะของฟิวส์สายแบบสายอ่อนและการต่อไปใช้งาน	16
2.19	โครงสร้างของเซอร์กิตเบรกเกอร์	16
2.20	ลักษณะของรีเลย์ที่ใช้ในรถยนต์โดยทั่วไป	17
2.21	โครงสร้างของแบตเตอรี่	17
2.22	การทำงานของ PIR Sensor	18
2.23	PIR Motion Sensor SB-0061	18
2.24	Output ในโหมด H ของ PIR Motion Sensor	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่

2.25	Output ในโหมด L ของ PIR Motion Sensor	19
2.26	GSM Module SIM800L	21
3.1	การต่อ LM7805	22
3.2	การใช้ Transistor เบอร์ 2N3904 เพื่อขับกระแสให้ Relay	23
3.3	การต่อ Digital Input	23
3.4	การออกแบบส่วนขยายใช้งาน ในโปรแกรม EAGLE	24
3.5	ลายวงจรสำหรับลง PCB	24
3.6	บล็อกไดอะแกรมการทำงานรวมของระบบ	25
3.7	Flowchart การทำงานของอุปกรณ์	26
3.8	บอร์ด ARDUINO MEGA	27
3.9	LCD ขนาด 20*4	28
3.10	Module I2C สำหรับต่อกับจอ LCD	28
3.11	RFID (RC522)	29
3.12	Module Data Logger	29
3.13	Active Buzzer Module	30
3.14	Dc-to-Dc Step Down LM2596 Module	30
3.15	GSM Module SIM800L	31
3.16	PIR Motion Sensor Module	31
4.1	Serial monitor เมื่อมีการแตะการ์ดเข้ามา	33
4.2	ค่าบน LCD เมื่อมีการแตะการ์ดเข้ามา	33
4.3	ค่าบนหน้าจอเมื่อมีการแตะการ์ดซ้ำเข้ามา	34
4.4	Serial monitor เมื่อมีการดับสวิตช์ไฟรถยนต์ กรณี In = out	35
4.5	Serial monitor เมือมีการดับสวิตช์รถยนต์ กรณี In != out	35
4.6	การเก็บประวัติขึ้น-ลงไว้ภายใน SD CARD	36
4.7	สัญญาณ Output จาก PIR Sensor	37
4.8	เมื่อมีการแตะ การ์ด	38
4.9	เมื่อจำนวน In = Out	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.10	กระจกไฟฟ้าถูกเปิดเองอัตโนมัติ	39
4.11	LCD แสดงผลบอกว่า Module GSM กำลังโทรออก	39
4.12	มีการโทรเข้าจาก Module GSM	40
4.13	การกดวางสายจากตัวอุปกรณ์	40
4.14	Serial monitor เมื่อดับกุญแจรถและจำนวน In และ Out เป็น ศูนย์	41
4.15	UID ของ Tag Card	41

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

จากปัญหากรณีเด็กติดอยู่ในรถแล้วเสียชีวิต เรื่องราวที่เกิดขึ้นส่วนหนึ่งล้วนมาจากความประมาทเลินเล่อของผู้ใหญ่ที่ดูแลเด็ก คนส่วนใหญ่มักเข้าใจว่าเป็นเพราะขาดอากาศหายใจเนื่องจากประตูหน้าต่างปิดสนิท แต่ความจริงแล้วอากาศภายในรถสามารถนอนได้นานเป็นชั่วโมง แต่ส่วนใหญ่ที่เด็กจะเสียชีวิตเป็นเพราะความร้อนภายในที่สูงขึ้น สิ่งที่ตามมา เด็กอาจหยุดหายใจ และอวัยวะทุกอย่างหยุดทำงาน กรมควบคุมโรค ได้เผยถึงข้อมูลจากการเฝ้าระวังข่าวจากสื่อของสำนักโรคไม่ติดต่อ กรมควบคุมโรค ว่าในช่วง 5 ปี (พ.ศ.2555-2559) พบเด็กถูกลืมในรถถึง 13 ครั้ง เสียชีวิต 6 ราย ในจำนวนเด็กที่เสียชีวิตเป็นชาย 4 ราย หญิง 2 ราย ราย โดยเด็กที่เสียชีวิตถูกลืมทิ้งไว้ในรถรับ-ส่งนักเรียน 5 ราย และรถยนต์ส่วนบุคคล 1 ราย ทั้งหมดถูกลืมทิ้งไว้นานกว่า 6 ชั่วโมงขึ้นไป นอกจากนี้ยังพบว่าเด็กที่เสียชีวิต จำนวน 4 รายนอนหลับอยู่บนรถขณะรถกำลังไปส่งที่โรงเรียน ส่วนเด็กที่ช่วยไว้ได้ทัน 7 คน ทั้งหมดผู้ปกครองไม่ได้ลืม แต่ตั้งใจทิ้งเด็กไว้ในรถเอง เพราะคิดว่าลงไปทำธุระไม่นาน (โดยติดเครื่องยนต์และเปิดแอร์ทิ้งไว้) ซึ่งทั้งหมดเป็นเด็กอายุ 3-4 ปี ที่ยังช่วยเหลือตัวเองได้ไม่เต็มที่ ด้วยปัญหาที่กล่าวมานี้ ปริญญานิพนธ์นี้จึงเป็นปริญญานิพนธ์ที่แก้ไขปัญหาดังกล่าว โดยการพัฒนาระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์ และประยุกต์ใช้เทคโนโลยีระบบสื่อสารไร้สาย ประหยัดค่าใช้จ่าย ใช้งานง่ายสะดวกและมีประสิทธิภาพ

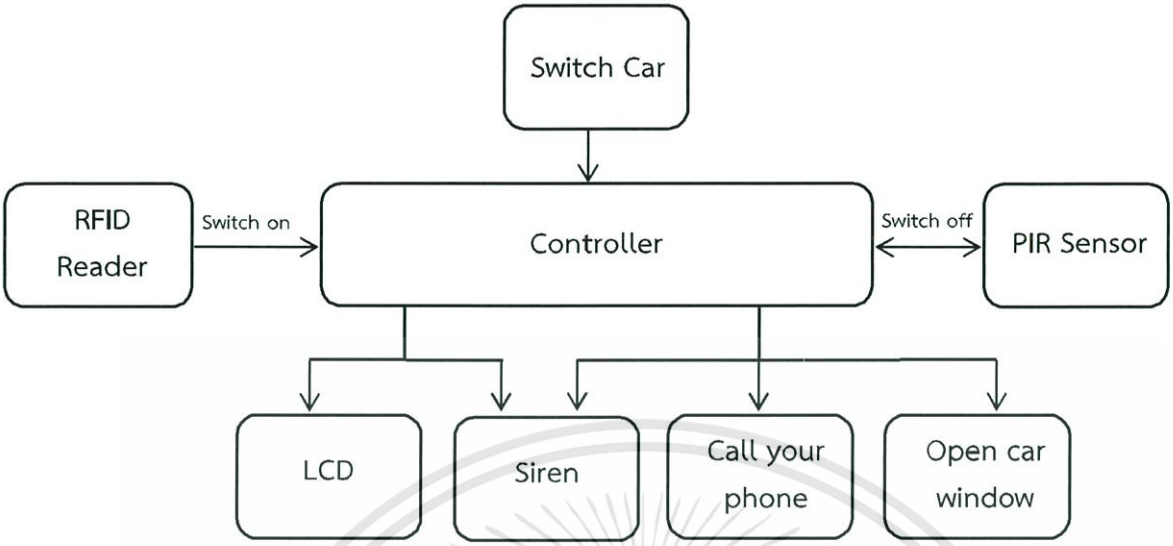
1.2 วัตถุประสงค์

1) เพื่อสร้างอุปกรณ์ป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์ ให้มีความปลอดภัยในรถยนต์
2) เพื่อความปลอดภัย ป้องกัน ชีวิตให้กับเด็กที่นอนหลับ หรือ ถูกลืม ไว้ภายในรถยนต์
3) เพื่อศึกษาระบบไฟฟ้าภายในรถยนต์ ประเมินพฤติกรรมของบุคคลที่ขับรถยนต์
4) เพื่อศึกษาการฝึกใช้งาน Microcontroller ชนิด Arduino และศึกษาเขียนโปรแกรมควบคุม

1.3 ขอบเขตของปริญญานิพนธ์

สร้างระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์ ซึ่งระบบออกแบบให้มีการป้องกันและแจ้งเตือน โดยมีการใช้ระบบอาร์เอฟไอดีในการตรวจนับคนขึ้นลงรถยนต์ ซึ่งถ้ามีการขึ้นลงของคนภายในรถยนต์ไม่เท่ากันจะมีการส่งเสียงแจ้งเตือนผ่านทางไซเรน กรณีหากไม่ได้ยินเสียงไซเรนของรถยนต์ จะทำการติดเซ็นเซอร์ตรวจจับการเคลื่อนไหวไว้ภายในรถ เพื่อตรวจสอบอีกครั้งเมื่อมีการเคลื่อนไหวภายในรถยนต์ ซึ่งเมื่อมีการเคลื่อนไหวอุปกรณ์จะทำการสั่งให้ไซเรนดังทันทีพร้อมทั้งเปิดกระจกรถยนต์ในกรณีที่เป็นกระจกไฟฟ้า และโทรหาเจ้าของรถทันทีเพื่อแจ้งเตือน โดยอุปกรณ์ทั้งหมดจะใช้การควบคุมผ่านไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.1 บล็อกไดอะแกรมการทำงานของระบบ

จากรูปที่ 1.1 เมื่อเปิดสวิตช์ไฟรถ จะใช้การป้องกันการติดในรถของเด็กโดยใช้ RFID ในการเก็บข้อมูลการขึ้น-ลงรถของเด็กแต่ละคน เมื่อจำนวนขึ้นไม่เท่ากันจำนวนลง เมื่อดับสวิตช์รถจะทำการแจ้งเตือนเป็นเสียงไซเรนทันที เมื่อสวิตช์ไฟดับลงจะเริ่มใช้ PIR Sensor เพื่อทำการตรวจจับการเคลื่อนไหวของเด็กที่ติดในรถแล้วทำการส่งเสียงเตือน พร้อมทั้งโทรศัพท์หาเจ้าของรถและเปิดกระจกทันทีเพื่อป้องกันการเสียชีวิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 Arduino

เป็นบอร์ดไมโครคอนโทรลเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัว บอร์ด Arduino ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย

ความง่ายของบอร์ด Arduino ในการต่ออุปกรณ์เสริมต่าง ๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิกส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขา I/O ของบอร์ด หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม (Arduino Shield) ประเภทต่าง ๆ เช่น Arduino XBee Shield, Arduino Music Shield, Arduino Relay Shield, Arduino Wireless Shield, Arduino GPRS Shield เป็นต้นมา เสียบกับบอร์ดบนบอร์ด Arduino แล้วเขียนโปรแกรมพัฒนาต่อได้เลย

2.1.1 Layout & Pin out Arduino Board (Model: Arduino MEGA 2560)

โดยอุปกรณ์ที่ใช้จะมีลักษณะดังรูปที่ 2.1 โดย Arduino นั้นจะมีบอร์ดหลากหลายชนิดการเลือกใช้ขึ้นอยู่กับลักษณะงานของแต่ละบุคคล

รูปที่ 2.1 ส่วนประกอบของ Arduino board รุ่น MEGA 2560 [1]

รายละเอียดอุปกรณ์

1. USB Port ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
2. Reset Button เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่
3. ICSP Port ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. I/O Port Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin 0, 1 เป็นขา Tx, Rx Serial, Pin 3, 5, 6, 9, 10 และ 11 เป็นขา PWM
5. ICSP Port Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
6. MCU Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino
7. I/O Port นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็น ช่องรับสัญญาณอนาล็อกตั้งแต่ขา A0-A5
8. Power Port ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง +3.3 V, +5V, GND, V_{in}
9. Power Jack รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V
10. MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่าน Atmega16U2

2.1.2 จุดเด่นของบอร์ด Arduino

1. ง่ายต่อการพัฒนา มีรูปแบบคำสั่งพื้นฐาน ไม่ซับซ้อน เหมาะสำหรับผู้เริ่มต้นงาน
2. มี community กลุ่มคนที่ร่วมกันพัฒนาที่แข็งแรง
3. Open hardware ทำให้ผู้ใช้สามารถนำบอร์ดไปต่อยอดใช้งานได้หลายด้าน
4. ราคาไม่แพง
5. Cross platform สามารถพัฒนาโปรแกรมบน OS ใดก็ได้

2.1.3 รายละเอียดคุณลักษณะ

บอร์ด Arduino ที่ได้เลือกศึกษาเพื่อนำมาใช้งานคือรุ่น Arduino MEGA 2560 ดังรูปที่ 2.2 ซึ่งเป็นรุ่นที่อัพเกรดจากรุ่นพื้นฐานโดยการเพิ่มทั้งปริมาณ RAM และ ROM และจำนวนพอร์ตดิจิตอลที่เพิ่มมากขึ้น

รูปที่ 2.2 Arduino MEGA [1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 RFID Card Reader and Detector Module

ใช้สำหรับทำงานที่ย่านความถี่ HF หรือ 13.56MHz สามารถอ่านและเขียนแท็ก RFID ได้ โดยหน้าที่ของเครื่องอ่านป้ายคือ จะทำการเชื่อมต่อกับป้ายเพื่อทำการอ่านหรือเขียนข้อมูลลงในป้ายโดยใช้สัญญาณวิทยุ ซึ่งภายในเครื่องอ่านจะประกอบด้วย เสาอากาศ เพื่อใช้รับ-ส่งสัญญาณ, ภาครับ-ภาคส่งสัญญาณวิทยุ, วงจรควบคุมการอ่าน-เขียนข้อมูล และส่วนที่ติดต่อกับคอมพิวเตอร์ แท็กทำงานได้โดยไม่จำเป็นต้องใช้แหล่งจ่ายไฟจากภายนอก เพราะภายในบัตรมีวงจรกำเนิดไฟฟ้าเหนี่ยวนำ เป็นแหล่งพลังงานในตัวอยู่แล้ว ระยะการอ่านข้อมูลได้ในระยะสั้นๆ โดย RFID ที่เลือกใช้งานจะทำการรับส่งข้อมูลผ่าน SPI โดยจะมีลักษณะอุปกรณ์ดังรูปที่ 2.3

รูปที่ 2.3 RFID Card Reader and Detector Module [2]

2.3 LCD Module

จอ Liquid Crystal Display (LCD) เป็นจอแสดงผลรูปแบบหนึ่งที่นิยมนำมาใช้งานกันกับระบบสมองกลฝังตัวอย่างแพร่หลาย จอ LCD มีทั้งแบบแสดงผลเป็นตัวอักขระเรียกว่า Character LCD ซึ่งมีการกำหนดตัวอักษรหรืออักขระที่สามารถแสดงผลไว้ได้อยู่แล้ว และแบบที่สามารถแสดงผลเป็นรูปภาพหรือสัญลักษณ์ได้ตามความต้องการของผู้ใช้งานเรียกว่า Graphic LCD นอกจากนี้บางชนิดเป็นจอที่มีการผลิตขึ้นมาใช้เฉพาะงาน ทำให้มีรูปแบบและรูปร่างเฉพาะเจาะจงในการแสดงผล เช่น นาฬิกาดิจิตอล เครื่องคิดเลข หรือ หน้าปัดวิทยุ เป็นต้น โดยอุปกรณ์ที่ใช้จะเป็นจอ LCD ขนาด 20*4 ดังรูปที่ 2.4 เชื่อมต่อกับ Arduino ผ่าน Module I2C

รูปที่ 2.4 LCD Module [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 RFID (Radio Frequency Identification)

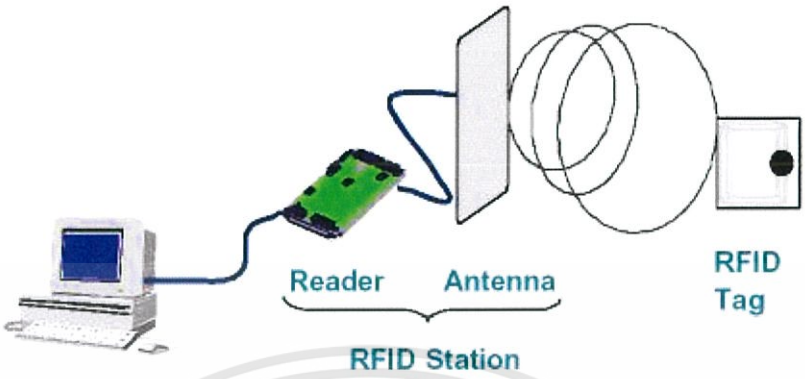
2.4.1 RFID คืออะไร

RFID ย่อมาจาก Radio Frequency Identification เป็น ระบบระบุลักษณะของวัตถุด้วยคลื่นความถี่วิทยุที่ได้ถูกพัฒนามาตั้งแต่ปี ค.ศ. 1980 มีวัตถุประสงค์หลักเพื่อนำไปใช้งานแทนระบบบาร์โค้ด (Barcode) โดยจุดเด่นของ RFID อยู่ที่การอ่านข้อมูลจากแท็ก (Tag) ได้หลายๆ แท็กแบบไร้สัมผัสและสามารถอ่านค่าได้แม้ในสภาพที่ทัศนวิสัยไม่ดี ทนต่อความเปียกชื้นแรงสั่นสะเทือนการกระทบกระแทก สามารถอ่านข้อมูลได้ด้วยความเร็วสูง โดยข้อมูลจะถูกเก็บไว้ในไมโครชิปที่อยู่ในแท็ก ในปัจจุบันได้มีการนำ RFID ไปประยุกต์ใช้งานในด้านอื่นๆ นอกเหนือจากนำมาใช้แทนระบบบาร์โค้ดแบบเดิม เช่น ใช้ในบัตรชนิดต่างๆ เช่น บัตรสำหรับใช้ผ่านเข้าออกสถานที่ต่างๆ บัตรที่จอดรถ ตามศูนย์การค้าต่างๆ ที่เราอาจพบเห็นอยู่ในรูปของแท็กสินค้า มีขนาดเล็กจนสามารถแทรกลงระหว่างชั้นของเนื้อกระดาษได้ หรือเป็นแคปซูลขนาดเล็กฝังเอาไว้ในตัวสัตว์เพื่อบันทึกประวัติต่างๆ เป็นต้น น่าสนใจกันแล้วใช่ไหมครับว่า RFID มีหลัก ทำงานอย่างไร มีส่วนประกอบต่างๆ อะไรบ้างและสามารถนำไปประยุกต์ใช้งานอะไรได้บ้าง และประเทศไทยของเรามีการพัฒนาระบบนี้บ้างหรือไม่

2.4.2 ส่วนประกอบของระบบ RFID

ในระบบ RFID จะมีองค์ประกอบหลักๆ อยู่ 2 ส่วนด้วยกัน ส่วนแรกคือทรานสปอนเดอร์หรือแท็ก (Transponder/Tag) ที่ใช้ติดกับวัตถุต่างๆ ที่เราต้องการ โดยแท็กที่ว่าจะบันทึกข้อมูลเกี่ยวกับวัตถุชิ้นนั้นๆ เอาไว้ ส่วนที่สองก็คือเครื่องสำหรับอ่าน/เขียนข้อมูลภายในแท็ก (Interrogator/Reader) ด้วยคลื่นความถี่วิทยุ เพื่อความเข้าใจผมขอเปรียบเทียบกับระบบบาร์โค้ดเพื่อให้เห็นภาพชัดเจน แท็กในระบบ RFID ก็คือ ตัวบาร์โค้ดที่ติดกันฉลากของสินค้า และเครื่องอ่านในระบบ RFID ก็คือเครื่องอ่านบาร์โค้ด (Scanner) โดยข้อแตกต่างของทั้งสองระบบคือ ระบบ RFID จะใช้คลื่นความถี่วิทยุในการอ่าน/เขียนส่วนระบบบรหัสแท่งจะใช้แสงเลเซอร์ ในการอ่านโดยข้อเสียของระบบบาร์โค้ด คือหลักการอ่านเป็นการใช้แสงในการอ่านแท็กบาร์โค้ด ซึ่งจะต้องอ่านแท็กที่ไม่อะไรกับปกปิดหรือต้องอยู่ในเส้นตรงเดียวกับลำแสงที่ยิงจากเครื่องสแกน และอ่านได้ทีละแท็กในระยะใกล้ๆแต่ระบบ RFID จะแตกต่างโดยสามารถอ่านแท็กได้ โดยไม่ต้องเห็นแท็ก หรือแท็กนั้นซ้อนอยู่ภายในวัตถุและไม่จำเป็นต้องอยู่ในเส้นตรงกับคลื่น เพียงอยู่ในบริเวณที่สามารถรับคลื่นวิทยุได้ก็สามารถอ่านข้อมูล ได้ และการอ่านแท็กในระบบ RFIDยังสามารถอ่านได้หลายๆ แท็กในเวลาเดียวกัน โดยระยะในการอ่านข้อมูลได้ไกลกว่าระบบบาร์โค้ดอีกด้วย โดยแสดงให้เห็นถึงหลักการทำงานอย่างง่ายดังรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ภาพรวมของระบบ RFID [4]

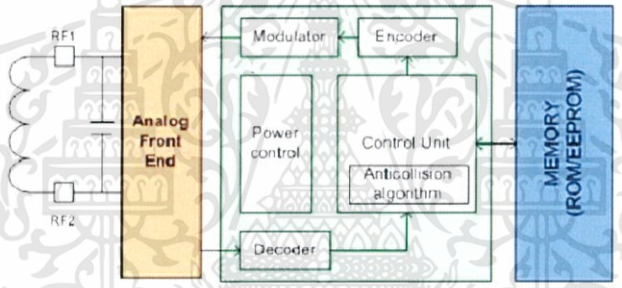
2.4.2.1 แท็ก (Tag)

โครงสร้างภายในของแท็กจะประกอบด้วย 2 ส่วนใหญ่ๆ ได้แก่ขดลวดขนาดเล็กซึ่งทำหน้าที่เป็นสายอากาศ (Antenna) สำหรับรับส่งสัญญาณคลื่นความถี่วิทยุ และสร้างพลังงาน ป้อนให้ส่วนของไมโครชิป (Microchip) ที่ทำหน้าที่เก็บข้อมูลของวัตถุเช่นรหัสสินค้า โดยทั่วไปตัวแท็กอาจอยู่ในชนิดทั้งเป็นกระดาษแผ่นฟิล์ม พลาสติก มีขนาดและรูปร่างต่างๆ กันไป ทั้งนี้ขึ้นอยู่กับวัสดุที่จะนำเอาไปติด และมีหลายรูปแบบเช่น ขนาดเท่าบัตรเครดิต เหรียญ กระดุม ฉลากสินค้า แคปซูล เป็นต้น ดังรูปที่ 2.6 แต่โดยหลักการอาจแบ่งแท็กที่มีการใช้งานกันอยู่ 2 ชนิดใหญ่ๆ แต่ละชนิดก็จะมีความแตกต่างกันในแง่ของการใช้งานราคาโครงสร้างและหลักการทำงานอยู่ ซึ่งจะขอกล่าวถึงและอธิบายแยกเป็นหัวข้อดังนี้

รูปที่ 2.6 RFID แท็กในรูปแบบต่างๆ [4]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

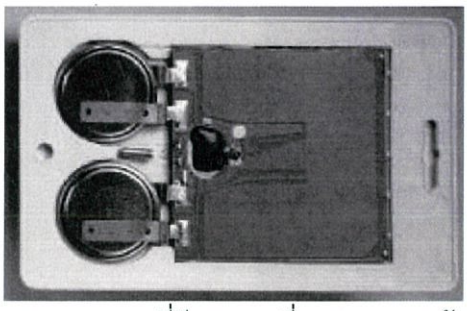
1) Passive RFID Tags แท็กชนิดนี้ไม่ต้องอาศัยแหล่งจ่ายไฟภายนอกใดๆ เพราะภายในแท็กจะมีวงจรกำเนิดไฟฟ้าเหนี่ยวนำขนาดเล็กเป็นแหล่งจ่ายไฟในตัวอยู่ ทำให้การอ่านข้อมูลทำได้ไม่ไกลมากนักระยะอ่านสูงสุดประมาณ 1 เมตร ขึ้นอยู่กับความแรงของเครื่องส่งและคลื่นความถี่วิทยุที่ใช้ ปกติแท็กชนิดนี้มักมีหน่วยความจำขนาดเล็กโดยทั่วไปประมาณ 16 ถึง 1,024 ไบต์ มีขนาดเล็กและน้ำหนักเบา ราคาต่อหน่วยต่ำไอซีของแท็กชนิดพาสซีฟที่มีการผลิตออกมา จะมีทั้งขนาดและรูปร่างเป็นแท่งหรือแผ่นขนาดเล็กจนแทบไม่สามารถมองเห็นได้ไปจนถึงขนาดใหญ่สะดุดตา ซึ่งต่างก็มีความเหมาะสมกับชนิดงานที่แตกต่างกัน ส่วนโครงสร้างภายในที่เป็นไอซีของแท็กนั้น ก็จะประกอบด้วย 3 ส่วนหลักๆ ได้แก่ ส่วนควบคุมการทำงานของภาครับ-ส่งสัญญาณวิทยุ (AnalogFront-End) ส่วนควบคุมภาคลอจิก (Digital Control Unit) ส่วนของหน่วยความจำ (Memory) ซึ่งอาจจะเป็นแบบ ROMหรือ EEPROM ดังรูปที่ 2.7



รูปที่ 2.7 บล็อกไดอะแกรมของ Passive Tag [4]

2) Active RFID Tags แท็กชนิดนี้จะต้องอาศัยแหล่งจ่ายไฟจากแบตเตอรี่ภายนอก ดังรูปที่ 2.8 เพื่อจ่ายพลังงานให้กับวงจรภายในทำงาน แท็กชนิดนี้มีหน่วยความจำภายในขนาดใหญ่ได้ถึง 1 เมกะไบต์ และสามารถอ่านได้ในระยะไกลสูงสุดประมาณ 10 เมตร แม้ว่าแท็กจะมีข้อดีอยู่หลายข้อแต่ก็มีข้อเสียด้วยเช่นกัน เช่น มีราคาต่อหน่วยแพง มีขนาดค่อนข้างใหญ่ และมีระยะเวลาในการทำงานที่จำกัด ดังรูปที่ 2.8 นอกจากการแบ่งจากชนิดที่ว่ามาแล้วแท็กก็ยังถูกแบ่งประเภทจากรูปแบบในการใช้งานได้เป็น 3 แบบ คือ แบบที่สามารถถูกอ่านและเขียนข้อมูลได้อย่างอิสระ (Read-Write),แบบเขียนได้เพียงครั้งเดียวเท่านั้นแต่อ่านได้อย่างอิสระ (Write-Once Read-Many หรือ WORM) และแบบอ่านได้เพียงอย่างเดียว (Read-Only) ด้วย อย่างไรก็ตามแท็กชนิดพาสซีฟจะนิยมใช้มากกว่า ดังนั้นจึงจะขอกล่าวถึงเฉพาะแท็กชนิดนี้เป็นหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

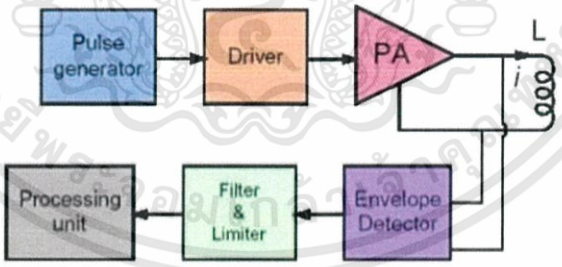


รูปที่ 2.8 ตัวอย่าง Active Tag ที่มีแบตเตอรี่ Lithium 2 ก้อนอยู่ภายนอก [4]

2.4.2.2 เครื่องอ่าน (Reader)

โดยหน้าที่ของเครื่องอ่านก็คือ การเชื่อมต่อเพื่อเขียนหรืออ่านข้อมูลลงในแท็กด้วยสัญญาณความถี่วิทยุภายในเครื่องอ่านจะประกอบด้วย เสาอากาศที่ทำจากขดลวดทองแดงเพื่อใช้รับส่งสัญญาณภาครับและภาคส่งสัญญาณวิทยุและวงจรควบคุมการอ่าน-เขียนข้อมูล จำพวกไมโครคอนโทรลเลอร์และส่วนของการติดต่อกับคอมพิวเตอร์ ดังรูปที่ 2.9 โดยทั่วไปเครื่องอ่านจะประกอบด้วยส่วนประกอบหลักดังนี้

- ภาครับและส่งสัญญาณวิทยุ
- ภาคสร้างสัญญาณพาหะ
- ขดลวดที่ทำหน้าที่เป็นสายอากาศ
- วงจรจูนสัญญาณ
- หน่วยประมวลผลข้อมูล และภาคติดต่อกับคอมพิวเตอร์



รูปที่ 2.9 โครงสร้างภายในเครื่องอ่าน [4]

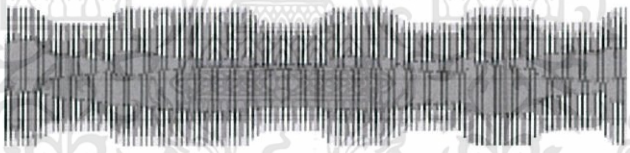
หน่วยประมวลข้อมูลที่อยู่ภายในเครื่องอ่านมักใช้เป็นไมโครคอนโทรลเลอร์ ซึ่งอัลกอริทึมที่อยู่ภายในโปรแกรมจะทำหน้าที่ถอดรหัสข้อมูล (Decoding) ที่ได้รับและทำหน้าที่ติดต่อกับคอมพิวเตอร์ ลักษณะขนาดและรูปร่างของเครื่องอ่านจะแตกต่างกันไปตามประเภทของการใช้งาน เช่น แบบมือถือขนาดเล็กหรือติดผนัง จนไปถึงขนาดใหญ่เท่าประตู (Gate size) เป็นต้น ดังแสดงในรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.10 รูปตัวอย่างเครื่องอ่านแบบต่างๆ [4]

2.4.3 หลักการและเทคนิคที่ใช้รับและส่งข้อมูลระหว่าง แท็กและเครื่องอ่าน

โดยมากเทคนิคในการรับส่งข้อมูลระหว่างเครื่องอ่านและแท็ก จะใช้หลักการมอดูเลตทางแอมปลิจูด (Amplitude Modulation: AM) หรือใช้การมอดูเลตทางแอมปลิจูดบวกกับการเข้ารหัสแมนเชสเตอร์ (Manchesterencoded AM) แต่ทว่าในปัจจุบันก็มีแท็กที่ใช้การมอดูเลตแบบอื่นๆ ด้วย เช่นการมอดูเลชั่นแบบเฟสชิฟคีย์อิง (Phase Shift Keying: PSK), ฟรีเควนซี่ชิฟคีย์อิง (Frequency Shift Keying: FSK) หรือการใช้การมอดูเลตทางความถี่ (Frequency Modulation: FM) ดังรูปที่ 2.11

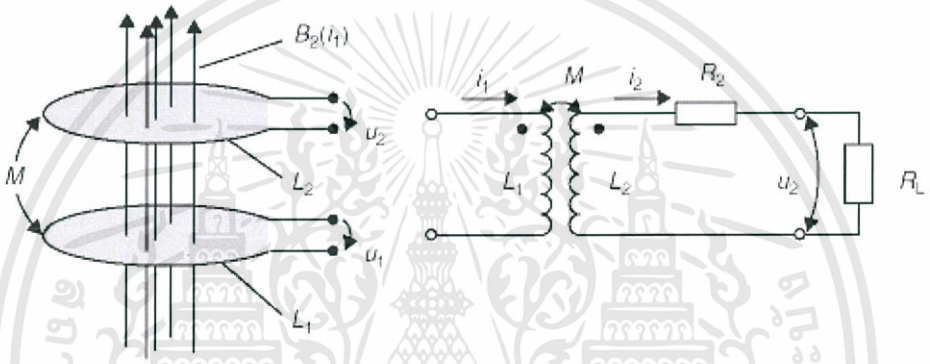


รูปที่ 2.11 รูปคลื่นของสัญญาณระหว่างแท็กและเครื่องอ่านแบบ AM [4]

ในการรับส่งข้อมูลหรือสัญญาณวิทยุระหว่างแท็กกับเครื่องอ่านจะทำได้อย่างมีประสิทธิภาพต่อเมื่อสายอากาศมีความยาวที่เหมาะสมกับความถี่พาหะที่ใช้งาน เช่น เมื่อความถี่ใช้งานเป็น 13.56 เมกะเฮิรตซ์ ความยาวของเสาอากาศ (เป็นเส้นตรง) ที่เหมาะสมก็คือ 22.12 เมตร แน่นอนว่าในทางปฏิบัติเราคงไม่สามารถนำเสาอากาศที่ใหญ่ขนาดนั้นมาใช้งานกับแท็กขนาดเล็กของเราได้ สายอากาศที่ดูเหมาะจะใช้ร่วมกับแท็กมากที่สุดก็คือสายอากาศที่เป็นขดลวดขนาดเล็กหรือที่มีชื่ออย่างเป็นทางการว่าสายอากาศแบบแมกเนติกไดโพล (magnetic dipole antenna) รูปแบบของสายอากาศแบบนี้ก็จะมีอยู่หลากหลาย ทั้งแบบที่เป็นขดลวดพันบนแกน อากาศหรือแกนเฟอร์ไรต์ แบบที่เป็นวงลูปที่ทำขึ้นจากลายทองแดง บนแผ่นวงจรพิมพ์ ทั้งที่เป็นลูปแบบวงกลมและสี่เหลี่ยม ทั้งนี้ความเหมาะสมในการใช้งานก็แตกต่างกันไปตามความถี่พาหะ และประเภทของงานด้วยเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากการรับส่งข้อมูลแล้วสายอากาศก็ยังทำหน้าที่เป็นแหล่งจ่ายพลังงานให้กับแท็กด้วย โดยอาศัยหลักการทำงานตามแนวคิดของไมเคิล ฟาราเดย์ เรื่องแรงดันเหนี่ยวนำในขดลวดที่เกิดขึ้นจากเส้นแรงแม่เหล็ก (จากเครื่องอ่าน) ที่มีค่าเปลี่ยนแปลงไปตามเวลา (Time-varyingmagnetic field) พุ่งผ่านสายอากาศของแท็กเมื่อแท็กและเครื่องอ่านตั้งอยู่ห่างกันในระยะ 0.16 เท่าของความยาวคลื่นพาหะที่ใช้เรียกปรากฏการณ์ที่เกิดขึ้นนี้ว่า transformer-type coupling ซึ่งเป็นปรากฏ-การณ์แบบเดียวกับการเกิดแรงดันไฟฟ้าเหนี่ยวนำขึ้นระหว่างขดลวดปฐมภูมิ (primary) และขดลวดทุติยภูมิ (secondary) ในหม้อแปลงไฟฟ้า (transformer) ดังรูปที่ 2.12 จะเป็นวงจรพื้นฐานสำหรับอธิบายกลไกที่เกิดขึ้นในการส่งข้อมูลของแท็ก

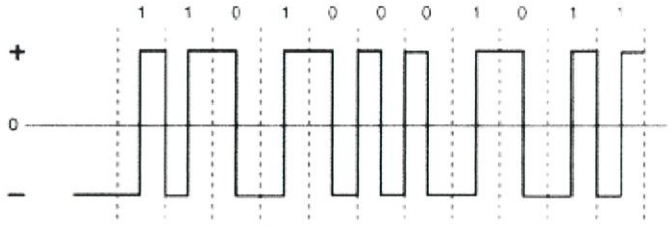


รูปที่ 2.12 ซ้ายลูปตัวนำที่ถูกเหนี่ยวนำด้วยสนามแม่เหล็ก ขวาวงจรทีสมมูลกับรูปทางซ้ายมือ [4]

2.4.3.1 การเข้ารหัสแบบแมนเชสเตอร์

เป็นการเข้ารหัสข้อมูลดิจิทัลวิธีหนึ่ง ก่อนที่ข้อมูลซึ่งผ่านการเข้ารหัสแล้วจะถูกส่งไปมอดูเลต เพื่อแก้ปัญหาเกี่ยวกับการซิงโครไนซ์ของข้อมูล เนื่องจากการส่งกระจายสัญญาณตามปกตินั้นหากมีการส่งสัญญาณดิจิทัลในระดับเดียวติดต่อกันเป็นช่วงยาว เช่น ส่งสัญญาณดิจิทัลที่มีค่าลอจิกเป็น 1 ออกไป 20บิตติดต่อกัน จะทำให้การซิงโครไนซ์ของข้อมูลเกิดการคลาดเคลื่อน (โดยปกติวงจรดิจิทัลจะปรับการซิงโครไนซ์ของข้อมูลได้เฉพาะในช่วงที่มีการเปลี่ยนระดับของข้อมูลจาก 1 เป็น 0 หรือจาก 0 เป็น 1) และทำให้รับข้อมูลผิดพลาดเพื่อป้องกันปัญหาดังกล่าวจึงจะต้องมีการนำสัญญาณดิจิทัลปกติไปผ่านเข้ารหัสเสียก่อน โดยการเข้ารหัสแบบแมนเชสเตอร์ จะเปลี่ยนให้สัญญาณดิจิทัลลอจิก 0 ถูกแทนด้วยการเปลี่ยนค่าจากลอจิก 1 เป็น 0 และสัญญาณดิจิทัลลอจิก 1 แทนด้วยการเปลี่ยนค่าจากลอจิก 0 เป็น 1 ข้อดีของการเข้ารหัสแบบนี้ก็คือ ทำให้การเปลี่ยนระดับของข้อมูลทุกๆ ครั้งเป็นไปอย่างแน่นอน หรือ เกิดการเข้าจังหวะ (synchronize) กันของข้อมูลนั่นเอง แต่ว่าการเข้ารหัสแบบนี้ก็มีข้อเสียอยู่กล่าวคือช่วงความถี่ที่ใช้ในการส่งข้อมูลต้องเพิ่มขึ้นเป็น 2 เท่า ดังรูปที่ 2.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 การสัญญาณรูปคลื่นที่เข้ารหัสแบบแมนเชสเตอร์ (Manchester) [4]

2.4.3.2 ขั้นตอนการทำงานระหว่างเครื่องอ่านกับแท็ก

1. ตัวเครื่องอ่านจะทำการส่งสัญญาณวิทยุอย่างต่อเนื่องหรือเป็นจังหวะ และรอคอยสัญญาณตอบจากตัวแท็ก

2. เมื่อแท็กได้รับสัญญาณคลื่นวิทยุที่ส่งมาจากเครื่องอ่านในระดับที่เพียงพอ ก็จะทำเหนี่ยวนำเพื่อสร้างพลังงานป้อนให้แท็กทำงาน โดยแท็กจะสร้างสัญญาณนาฬิกาเพื่อกระตุ้นให้วงจรภาคดิจิทัลในแท็กทำงาน

3. วงจรภาคดิจิทัลจะไปอ่านข้อมูลจากหน่วยความจำภายในและเข้ารหัสข้อมูลแล้วส่งไปยังภาคแอนะล็อกที่ทำหน้าที่มอดูเลตข้อมูล

4. ข้อมูลที่ถูกมอดูเลตจะถูกส่งไปส่งขดลวดที่ทำหน้าที่เป็นสายอากาศ เพื่อส่งไปยังเครื่องอ่าน

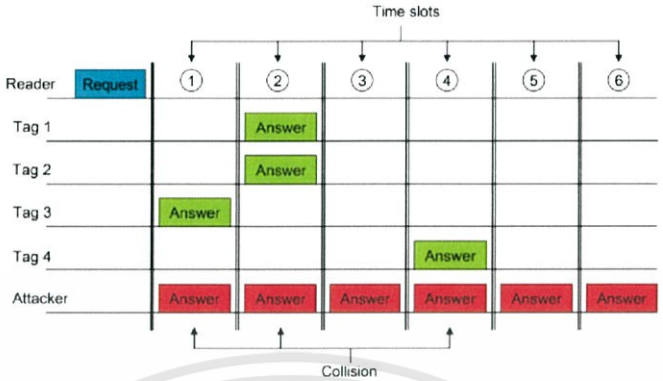
5. เครื่องอ่านจะสามารถตรวจจับสัญญาณการเปลี่ยนแปลงของแอมปลิจูด (Envelope Detector) และใช้พีก ดีเทกเตอร์ (Peak Detector) ในการแปลงสัญญาณข้อมูลที่มอดูเลตแล้วจากแท็ก

6. เครื่องอ่านจะถอดรหัสข้อมูลและส่งไปยังคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมต่อไป

2.4.3.3 การป้องกันการชนกันของสัญญาณข้อมูล (Anti-Collision)

การอ่านข้อมูลจากแท็กได้หลายๆ แท็กในเวลาเดียวกันเป็นข้อดีข้อหนึ่งของ RFID จะทำให้การอ่านข้อมูลของแท็กจำนวนมากทำได้อย่างรวดเร็ว ซึ่งสิ่งที่ทำให้การอ่านข้อมูลจากแท็กได้พร้อมๆ กัน นั้นก็คือ อัลกอริทึมที่ใช้ในการป้องกันการชนของข้อมูล (Anti-Collision) ที่อยู่ภายในระบบ RFID นั้นเอง จากรูปที่ 2.14 แสดงอัลกอริทึมที่ใช้ป้องกันการชนข้อมูลของแท็กบางชนิด โดยหลักการของการอ่านข้อมูลจากแท็กจะอ่านเป็นลำดับในเวลาที่กำหนด แต่ละแท็กจะไม่ส่งข้อมูลไปยังเครื่องอ่านทันทีจะมีการจัดสรรลำดับเวลา (TimeSlot) ในการส่งข้อมูลที่เวลาต่างๆ กัน ตามอัลกอริทึมที่กำหนดทำให้ข้อมูลที่เครื่องอ่านรับได้ไม่มีการชนของข้อมูลที่ส่งมาจากแท็กหลายแท็กพร้อมกันนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



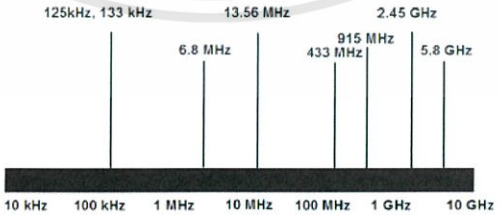
รูปที่ 2.14 ตัวอย่างของอัลกอริทึมในการป้องกันการชนของข้อมูล(Anti-Collision) ในแท็ก [4]

2.4.4 คลื่นพาหะในระบบ RFID

ในปัจจุบันคลื่นพาหะที่ใช้งานกันในระบบ RFID จะอยู่ในย่านความถี่ ISM (Industrial-Scientific-Medical) ซึ่งเป็นย่านความถี่ที่กำหนดการใช้งานในเชิงอุตสาหกรรม วิทยาศาสตร์และการแพทย์ สามารถใช้งานได้โดยไม่ตรงกับย่านความถี่ที่ใช้งานในการสื่อสารทั่วไป สำหรับคลื่นพาหะที่ใช้กันในระบบRFID อาจแบ่งออกได้เป็น 3 ย่านความถี่ใช้งานหลัก ได้แก่

- ย่านความถี่ต่ำ (Low Frequency: LH) ต่ำกว่า 150 kHz
- ย่านความถี่สูง (High Frequency: HF) 13.56 MHz
- ย่านความถี่สูงยิ่ง (Ultra High Frequency: UHF) 433/868/915 MHz

การใช้งาน 2 ย่านความถี่แรกจะเหมาะสำหรับใช้กับงานที่มีระยะการสื่อสารข้อมูลในระยะใกล้ (LH ระยะอ่านประมาณ 10-20 เซนติเมตร และ HF ระยะอ่านประมาณ 1เมตร) เช่น การตรวจสอบการผ่านเข้าออกพื้นที่การตรวจหาและเก็บประวัติในสัตว์ ส่วนย่านความถี่สูงยิ่ง จะถูกใช้กับงานที่มีระยะการสื่อสารข้อมูลในระยะไกล (UHF ระยะอ่านประมาณ1-10 เมตร) เช่น ระบบเก็บค่าบริการทางด่วน และในปัจจุบันระบบ RFID กำลังถูกวิจัยและพัฒนาในย่านความถี่ไมโครเวฟที่ความถี่ 2.4 GHz และความถี่ 5.8 GHz เพื่อใช้งานที่ต้องการระยะอ่านที่ไกลกว่า 10 เมตร เป็นต้น ดังรูปที่ 2.15



รูปที่ 2.15 ความถี่ย่านที่ระบบ RFID ถูกใช้งาน [4]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในแง่ของราคาและความเร็วในการสื่อสารข้อมูล เมื่อเทียบกันแล้ว RFID ซึ่งใช้คลื่นพาหะย่านความถี่สูงเป็นระบบที่มีความเร็วในการส่งข้อมูลสูงสุด และมีราคาแพงที่สุดด้วย เช่นกัน ส่วน RFID ที่ใช้คลื่นพาหะในอีก 2 ย่านความถี่จะมีระดับราคาและความเร็วลดหลั่น กันไป

2.5 อุปกรณ์พื้นฐานทางไฟฟ้ารถยนต์

ตามที่ทราบแล้วว่าไฟฟ้าที่เรานำมาใช้ประโยชน์คือไฟฟ้ากระแส ไฟฟ้าที่ใช้ตามบ้านและโรงงานอุตสาหกรรมจะเป็นไฟฟ้ากระแสสลับ (Alternating current) ส่วนในรถยนต์จะใช้ไฟฟ้ากระแสตรง(Direct current) ดังนั้นเวลาเลือกใช้อุปกรณ์ไฟฟ้าจะต้องเลือกให้ถูกกับชนิดของไฟฟ้าที่ใช้ เช่นถ้าซื้ออุปกรณ์ไฟฟ้ามาใช้ในบ้านจะต้องตรวจดูว่าเป็นชนิดที่ใช้กับไฟฟ้ากระแสสลับและที่สำคัญคือขนาดแรงเคลื่อนไฟฟ้าของอุปกรณ์นั้นใช้กับแรงเคลื่อนไฟฟ้ากี่โวลต์ ในประเทศไทยแรงเคลื่อนไฟฟ้าที่ใช้ตามบ้านนั้นมีขนาดแรงดัน 220 โวลต์ ความถี่ 50 เฮิรตซ์ (Hertz, Hz) แต่ในโรงงานอุตสาหกรรมมีทั้งขนาดแรงเคลื่อนไฟฟ้า 220 โวลต์และ 380 โวลต์ สำหรับในรถจะใช้ไฟฟ้ากระแสตรง ในรถยนต์นั่ง (รถเก๋ง) รถปิกอัพและรถบรรทุกขนาดเล็ก (รถบรรทุกไม่เกิน 6 ล้อ) จะใช้แรงเคลื่อนไฟฟ้าขนาด 12 โวลต์ รถบรรทุกขนาดใหญ่ (รถบรรทุก10 ล้อขึ้นไป) และรถโดยสารจะใช้แรงเคลื่อนไฟฟ้าขนาด 24 โวลต์ อุปกรณ์พื้นฐานทางไฟฟ้ารถยนต์นับว่ามีความจำเป็นในงานตรวจซ่อมระบบไฟฟ้าในรถยนต์ที่ช่างซ่อมจะต้องรู้จักและสามารถเลือกใช้ได้อย่างถูกต้อง รวมทั้งจะต้องทราบถึงผลเสียที่อาจเกิดขึ้นถ้าใช้อุปกรณ์นั้นๆ ผิดขนาดหรือผิดประเภทในการตรวจซ่อม ผู้เป็นช่างจะต้องเป็นผู้มีจิตสำนึกที่ดี มีความซื่อสัตย์สุจริต มีความรับผิดชอบและตรงต่อเวลา ให้ความสำคัญกับคุณภาพของผลงานที่ซ่อมเป็นอันดับแรก คุณภาพของงานจะบอกถึงความรู้ความสามารถของช่างซ่อมได้เป็นอย่างดี การตรงต่อเวลาและความซื่อสัตย์สุจริตจะสร้างความเชื่อมั่นให้ผู้มาใช้บริการมากยิ่งขึ้น สิ่งเหล่านี้จะทำให้ช่างซ่อมทั้งหลายประสบความสำเร็จในอาชีพได้รับความไว้วางใจจากผู้มาใช้บริการอุปกรณ์พื้นฐานทางไฟฟ้ารถยนต์ที่ควรทราบ

2.5.1 เต้าเสียบขั้วสายไฟ (Latch)

เต้าเสียบขั้วสายไฟหรือปลั๊กเสียบขั้วสายไฟ เป็นอุปกรณ์ที่เก็บรวมขั้วสายไฟในกลุ่มเดียวกันไว้ด้วยกันทำให้สามารถเก็บชุดสายไฟได้เรียบร้อยสวยงามและสะดวกในการตรวจซ่อม เต้าเสียบขั้วสายไฟทำจากพลาสติกมีหลายแบบหลายขนาดแล้วแต่จำนวนของสายไฟที่ใช้ในวงจรนั้นๆ เต้าเสียบขั้วสายไฟจะมีร่องสำหรับล็อกขั้วสายไฟไม่ให้หลุดออกจากเต้าเสียบ สำหรับตัวเต้าเสียบเองจะมีตำแหน่งบังคับระหว่างตัวผู้กับตัวเมียให้เสียบเข้าหากันได้ตำแหน่งเดียวเพื่อป้องกันความผิดพลาดในขณะที่ถอดและใส่เข้าตำแหน่งเดิมในการตรวจซ่อมและยังมีครีบสำหรับล็อกไม่ให้ตัวเต้าเสียบหลุดออกจากกันด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 ฟิวส์ (Fuse)

ฟิวส์มีหน้าที่ตัดทางไฟฟ้าในวงจรเมื่อเกิดการลัดวงจรในวงจรนั้นๆหรือเมื่อเกิดกระแสไฟฟ้ามากเกินในวงจร วงจรไฟฟ้าทุกวงจรในรถยนต์ควรจะต้องมีฟิวส์ต่อไว้ ถ้าวงจรใดไม่มีฟิวส์ต่อไว้เมื่อเกิดการลัดวงจรจะทำให้วงจรนั้นเกิดความเสียหายและยังทำให้วงจรไฟฟ้าอื่นๆเกิดความเสียหายตามไปด้วยเพราะมีสายไฟจำนวนมากที่พันรวมอยู่ด้วยกัน ดังนั้นวงจรไฟฟ้าต่างๆจะต้องมีฟิวส์ต่อไว้และต้องใช้ฟิวส์ที่มีค่าทนกระแสไฟฟ้าที่เหมาะสม การใช้ฟิวส์ที่มีค่าทนกระแสไฟฟ้าต่ำเกินไปตัวฟิวส์จะขาดบ่อยแต่ถ้าใช้ฟิวส์ที่มีค่าทนกระแสไฟฟ้าสูงเกินไปตัวฟิวส์จะไม่ขาดเพื่อตัดวงจรเมื่อเกิดการลัดวงจรซึ่งจะเป็นอันตรายกับวงจรนั้นๆได้

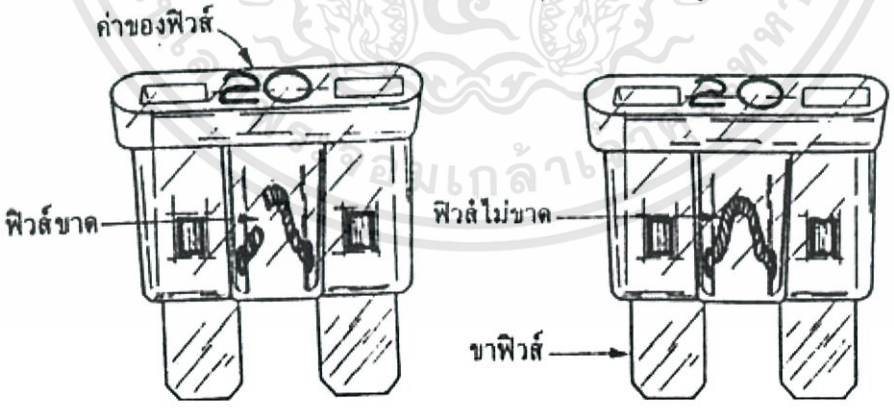
ประเภทของฟิวส์

ฟิวส์ที่ใช้ในรถยนต์มี 3 แบบ คือ

- **ฟิวส์แบบหลอด** ดังรูปที่ 2.16 ส่วนใหญ่จะใช้ในรถยนต์รุ่นเก่า

รูปที่ 2.16 ลักษณะของฟิวส์แบบหลอด [5]

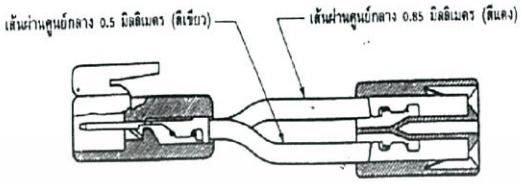
- **ฟิวส์แบบเสียบ** เป็นฟิวส์ที่รถยนต์ปัจจุบันนิยมใช้ ดังรูปที่ 2.17



รูปที่ 2.17 ลักษณะของฟิวส์แบบเสียบ [5]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

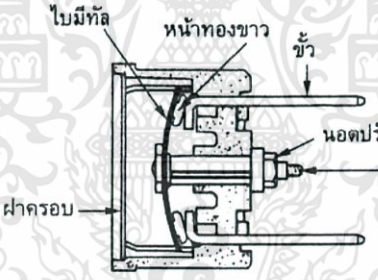
- **ฟิวส์สาย** ดังรูปที่ 2.18 จะใช้เป็นฟิวส์เมนที่ต่อก่อนจะแยกเป็นฟิวส์ย่อยในวงจรต่าง ๆ



รูปที่ 2.18 ลักษณะของฟิวส์สายแบบสายอ่อนและการต่อไปใช้งาน [5]

2.5.3 เซอร์กิตเบรกเกอร์ (Circuit breaker)

เป็นอุปกรณ์ป้องกันวงจรไฟฟ้าทำหน้าที่แทนฟิวส์เพื่อป้องกันในวงจรที่ซับซ้อนและใช้กระแสไฟฟ้าจำนวนมากเช่นวงจรกระจกไฟฟ้า วงจรซันรูฟ เป็นต้นและจะมีใช้ในรถยนต์บางรุ่นเท่านั้นในเซอร์กิตเบรกเกอร์จะมีคอนแทก (หน้าทองขาว) ซึ่งจะทำหน้าที่ตัดวงจรในทันทีที่มีกระแสไฟฟ้าเกินและจะดีดตัวกลับมาต่อวงจรไฟฟ้าอีกครั้งเมื่อไบมีทัลมีความร้อนลดลงโดยมีลักษณะดังรูปที่ 2.19

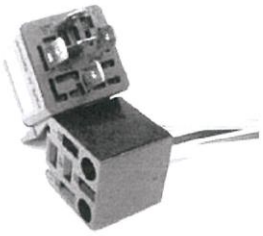


รูปที่ 2.19 โครงสร้างของเซอร์กิตเบรกเกอร์ [5]

2.5.4 รีเลย์ (Relay)

รีเลย์ ดังรูปที่ 2.20 เป็นอุปกรณ์ไฟฟ้าที่นิยมใช้กันมากในรถยนต์ปัจจุบัน เนื่องจากรีเลย์จะช่วยลดแรงเคลื่อนไฟฟ้าตกคร่อมภายในวงจร ทำให้วงจรนั้นๆมีกระแสไฟฟ้าผ่านไปใช้งานได้อย่างเต็มที่และยังเป็นการยืดอายุการใช้งานของสวิตช์เนื่องจากกระแสไฟฟ้าส่วนใหญ่ที่ไปเลี้ยงอุปกรณ์ไฟฟ้า (Load) จะผ่านทางรีเลย์ จะมีกระแสไฟฟ้าส่วนน้อยที่เลี้ยงคอยล์ของรีเลย์เท่านั้นที่ผ่านสวิตช์ รีเลย์ที่นิยมใช้ในปัจจุบันมีทั้งแบบ 3 ขั้ว 4 ขั้วและ5 ขั้ว ซึ่งแต่ละแบบมีหลักการทำงานที่คล้ายกันขึ้นอยู่กับว่าในวงจรไฟฟ้าของรถยนต์รุ่นนั้นจะออกแบบให้ใช้กับรีเลย์แบบใด

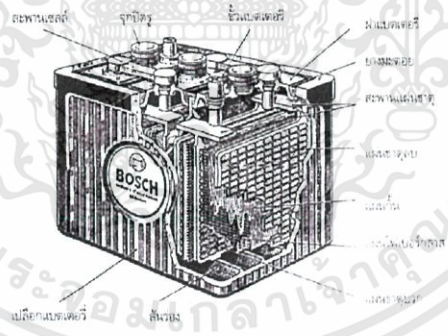
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 ลักษณะของรีเลย์ที่ใช้ในรถยนต์โดยทั่วไป [5]

2.5.5 แบตเตอรี่ (Battery)

แบตเตอรี่เป็นอุปกรณ์ไฟฟ้าที่ทำหน้าที่จ่ายพลังงานไฟฟ้าให้กับอุปกรณ์ไฟฟ้าต่างๆในรถยนต์แบตเตอรี่นับว่าเป็นอุปกรณ์พื้นฐานทางไฟฟ้ารถยนต์ที่สำคัญที่สุดอย่างหนึ่ง เพราะถ้าไม่มีแบตเตอรี่รถยนต์จะไม่สามารถสตาร์ตติดได้หรือถ้าแบตเตอรี่อยู่ในสภาพที่ไม่สมบูรณ์จะทำให้รถยนต์คันนั้นไม่อยู่ในสภาพที่ใช้งานได้อย่างสะดวกสบาย เพราะฉะนั้นการดูแลและบำรุงรักษาแบตเตอรี่ให้อยู่ในสภาพที่สมบูรณ์จึงต้องกระทำอยู่เป็นประจำและอย่างถูกวิธีจึงจะทำให้แบตเตอรี่มีอายุการใช้งานได้ยาวนานแบตเตอรี่ที่ใช้ในรถยนต์โดยทั่วไปจะเป็นเซลล์ไฟฟ้าแบบเปียก ดังรูปที่ 2.21 (มีสารเคมีที่เป็นของเหลวอยู่ภายใน) เมื่อแบตเตอรี่ถูกใช้งานไปจนไม่สามารถจ่ายกระแสไฟฟ้าได้อีกสามารถจะนำแบตเตอรี่ไปประจุไฟใหม่ได้ ซึ่งเซลล์ไฟฟ้าที่สามารถนำไปประจุไฟฟ้าใหม่ได้อีกเรียกว่า“เซลล์ทุติยภูมิ”ส่วนเซลล์ไฟฟ้าที่ไม่สามารถจะนำไปประจุไฟฟ้าใหม่ได้อีกเรียกว่า“เซลล์ปฐมภูมิ”เช่นถ่านไฟฉายที่ใช้โดยทั่วไปเป็นต้น



รูปที่ 2.21 โครงสร้างของแบตเตอรี่ [5]

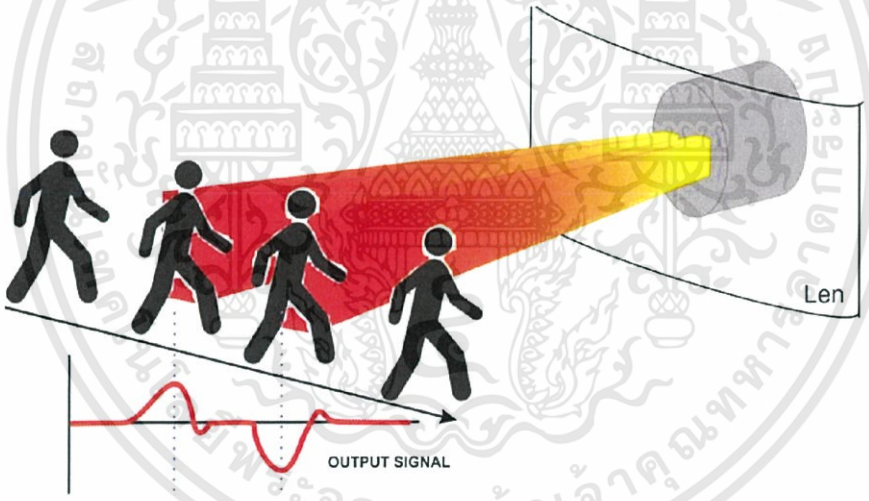
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 PIR Motion Sensor

อุปกรณ์ตรวจจับการเคลื่อนไหวเมื่อมีคนอยู่ในบริเวณเป้าหมาย (detection zone) มีอยู่หลายประเภท (เช่น ทำงานโดยใช้แสงอินฟราเรด หรือ ใช้คลื่นไมโครเวฟ ในการตรวจจับการเคลื่อนไหว) แต่ประเภทหนึ่งที่พบเห็นได้บ่อยเรียกว่า PIR (Passive Infrared) human detection sensor ซึ่งทำงานโดยการตรวจจับการเปลี่ยนแปลงของรังสีอินฟราเรดที่แผ่ออกมาจากร่างกายของมนุษย์ (หรือแหล่งความร้อนอื่นๆ) และเปรียบเทียบกับพื้นหลัง ตัวอย่างการนำไปประยุกต์ใช้งาน เช่น การตรวจจับผู้บุกรุกในอาคาร การตรวจจับผู้อยู่ในห้องและกำหนดพฤติกรรมการทำงานของเครื่องปรับอากาศ เป็นต้น

PIR Motion Sensor คือ อุปกรณ์ Sensor ชนิดหนึ่งที่ใช้ตรวจจับคลื่นรังสี Infrared ที่แพร่จาก มนุษย์ หรือ สัตว์ ที่มีการเคลื่อนไหว ทำให้มีการนำเอา PIR มาประยุคใช้งานกันเป็นอย่างมากใช้เพื่อตรวจจับการเคลื่อนไหวของสิ่งมีชีวิต หรือ ตรวจจับการบุกรุกในงานรักษาความปลอดภัย

การทำงานของ PIR Sensor

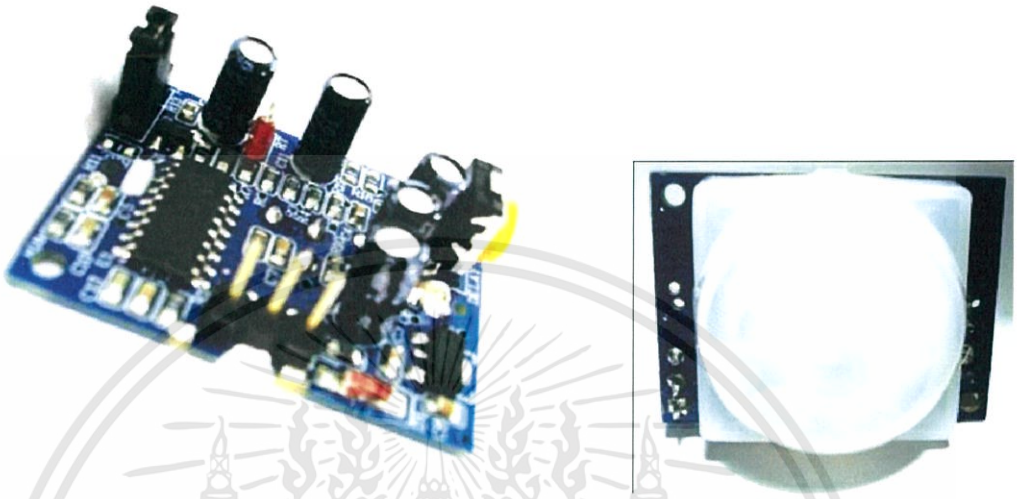


รูปที่ 2.22 การทำงานของ PIR Sensor [7]

ภายใน PIR จะมีอุปกรณ์ตรวจจับรังสี Infrared อยู่ 2 ชุดด้วยกันดังรูป เมื่อมี คน หรือ สัตว์ที่มีความอบอุ่นในร่างกายเคลื่อนที่ผ่านเข้ามาในพื้นที่โซนที่ PIR สามารถตรวจจับคลื่นรังสี Infrared ที่แพร่ออกมาจากสิ่งมีชีวิตได้ PIR จะเปลี่ยนคลื่นรังสี Infrared ให้กลายเป็นกระแสไฟฟ้าดังรูปที่ 2.22 จะเห็นว่าเมื่อมีสิ่งมีชีวิตเคลื่อนที่ผ่าน อุปกรณ์ตรวจจับรังสี Infrared ตัวที่ 1 จะได้สัญญาณ Output ออกมาสูงกว่าแรงดันปกติ และเมื่อสิ่งมีชีวิตเคลื่อนที่ผ่านอุปกรณ์ตรวจจับรังสี Infrared ตัวที่ 2 จะได้แรงดัน Output ต่ำกว่าค่าแรงดันปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

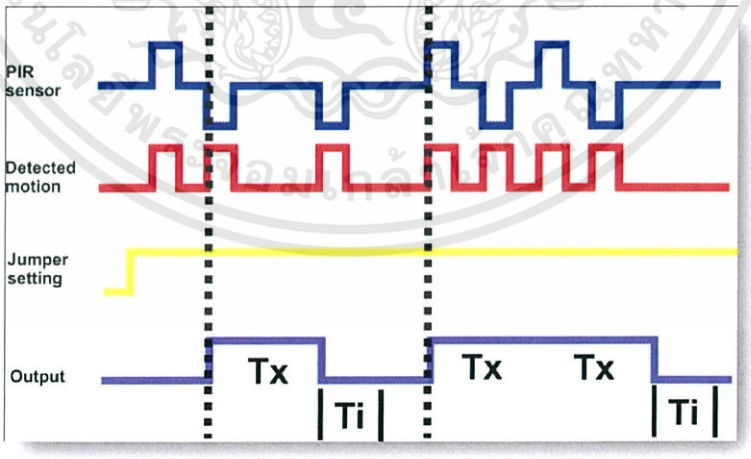
SB-0061 เป็นโมดูล PIR Motion Sensor ดังรูปที่ 2.23 ที่มีการเพิ่มเอา Cds เข้ามาภายในโมดูล สามารถเลือกลักษณะของ Output ได้ 2 รูปแบบ และ สามารถปรับค่า Delay ได้



รูปที่ 2.23 PIR Motion Sensor SB-0061 [7]

Cds คืออุปกรณ์ Electronic ที่ใช้ตรวจวัดแสง (LDR) ใช้เพื่อวัดแสงและนำค่าแสงเข้ามาช่วยตัดสินใจในการตรวจจับการเคลื่อนไหว SB-0061 สามารถเลือกลักษณะของ Output ได้ 2 ลักษณะโดยการเลือก Jumper ให้ไปอยู่ที่ตัวอักษร H หรือ L

Output ในโหมด H (Retriggering)

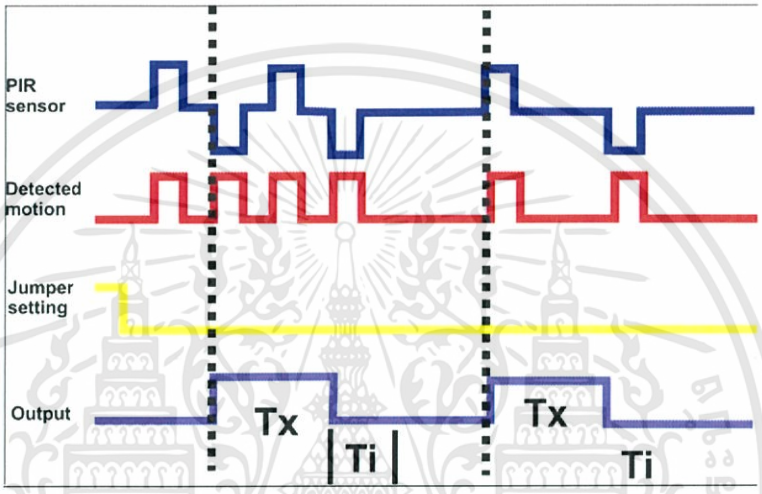


รูปที่ 2.24 Output ในโหมด H ของ PIR Motion Sensor [7]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ PIR สามารถตรวจจับการเคลื่อนไหวได้จะมีการส่งสัญญาณ Output ออกมาโดยจะส่ง Output ออกมาค้างสถานะเอาไว้จนกว่าจะไม่มีการเคลื่อนไหวจึงจะเปลี่ยนสถานะกลับไปสู่สถานะปกติ และจะเข้าสู่ช่วง Ti ดังรูปที่ 2.24 ซึ่งในช่วงนี้ตัว PIR จะไม่ตอบสนองการเคลื่อนไหวใดๆ

Output ในโหมด L (Non - Retriggering)



รูปที่ 2.25 Output ในโหมด L ของ PIR Motion Sensor [7]

ในโหมด L (Non-Retriggering) เมื่อตรวจจับการเคลื่อนไหวได้ PIR จะส่ง Output ออกมาและจะส่งออกมาค้างเอาไว้ในช่วงเวลาหนึ่ง (Tx) และจะเปลี่ยนสถานะกลับตามเดิม ถึงแม้ว่าในขณะนั้นจะยังคงมีการเคลื่อนไหวอยู่หรือไม่ และจะเข้าสู่สถานะ Ti คือจะไม่รับรู้การเคลื่อนไหวไปชั่วขณะหนึ่ง ดังรูปที่ 2.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 GSM Module SIM800L

บอร์ด SIM800L Module นี้มีขนาดเล็ก ดังรูปที่ 2.26 มาพร้อมเสาอากาศ พร้อมใช้งานกับ microSIM SIM800L Module สามารถใช้ในการรับส่ง SMS หรือโทรศัพท์หาเบอร์ที่ต้องการ เชื่อมต่ออินเตอร์เน็ตและอื่นๆ ได้เหมือนบอร์ด GSM รุ่นอื่นๆ แต่ไม่สามารถคุยได้เพราะตัดขาลำโพงกับไมออก จึงเหมาะกับงานเฉพาะด้านที่ต้องการความคุ้มค่า การใช้งานไลบารีเหมือน SIM900 การต่อขาใช้เพียง 2 เส้นคือ RX และ TX ใช้ไฟที่ 3.7-4.2 โวลต์กระแสสูงสุด 2A

รูปที่ 2.26 GSM Module SIM800L [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

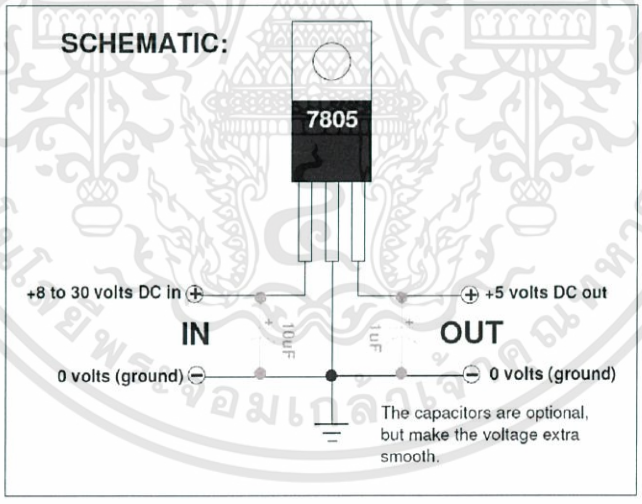
การออกแบบและการจัดทำปริญญานิพนธ์

ปริญญานิพนธ์นี้เป็นการสร้างสร้างอุปกรณ์ป้องกันเด็กติดภายในรถโดยสารไม่ประจำทาง โดยในเทอมนี้จะทำในส่วนที่เป็นตัวนับจำนวนขึ้น-ลงโดยใช้ RFID พร้อมทั้งเก็บข้อมูลการขึ้น-ลงไว้ใน Data Logger และแสดงค่าจำนวนคนขึ้น-ลง ผ่านหน้าจอ LCD และแสดงค่าเมื่อจำนวนขึ้น-ลง ไม่เท่ากันผ่าน LED พร้อมทั้งแจ้งเตือนผ่านเสียงไซเรนรถยนต์เมื่อจำนวนขึ้น-ลง ไม่เท่ากันเมื่อมีการดับสวิตช์ไฟรถไปแล้ว

3.1 การออกแบบ

3.1.1 การออกแบบฮาร์ดแวร์

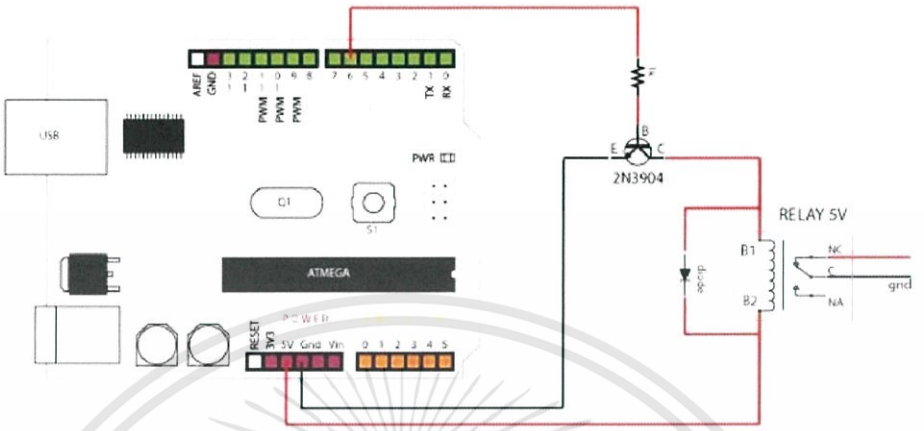
การออกแบบวงจรเสริม สำหรับใช้กับ ARDUINO UNO เนื่องจาก เราต้องใช้ไฟ 12 V จากแบตเตอรี่รถยนต์ แต่ ARDUINO ต้องการไฟเพียง 5 V ที่พอร์ต USB กับออกแบบส่วนที่เป็นวงจร relay เนื่องจาก Port Digital ของ ARDUINO จ่ายไฟไม่เพียงพอสำหรับการขับกระแสให้ Relay ทำงาน



รูปที่ 3.1 การต่อ LM7805 [1]

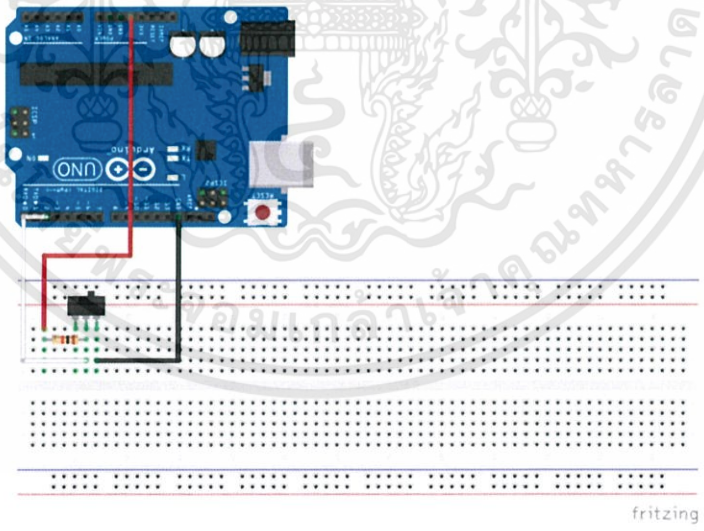
เนื่องจากไฟจากรถยนต์มีค่า 12V DC เราจึงต้องใช้ LM7805 เพื่อลดแรงดันไฟลงเหลือเพียง 5 V โดยให้ Cin = 100 uF และ Cout = 10 uF ดังรูปที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 การใช้ Transistor เบอร์ 2N3904 เพื่อขับกระแสให้ Relay [1]

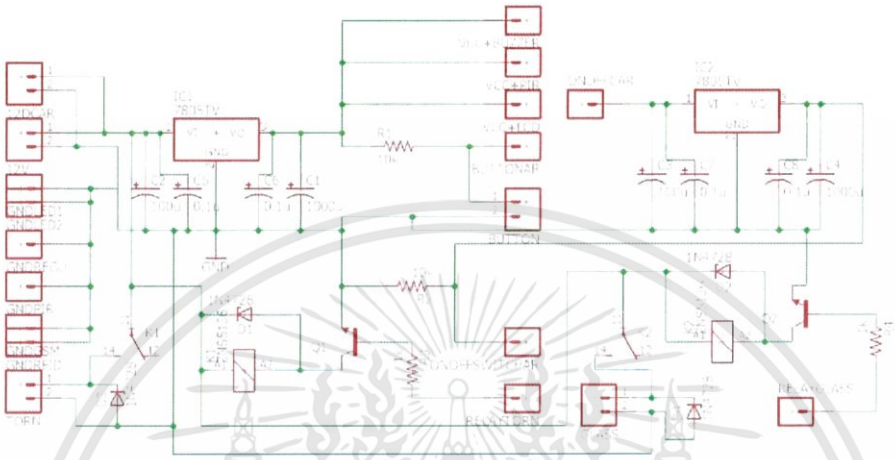
เนื่องจาก Relay ต้องการกระแสมากเวลามีการสั่งให้มันทำงาน อาจจะทำให้ ARDUINO จ่ายไฟไม่พอสำหรับการทำงานหลายๆอย่างเมื่อมีการดึงไฟ มาจ่ายให้ Relay ดังนั้น เราต้องใช้ Transistor เป็นตัวขับกระแสให้มากขึ้นเพื่อพอสำหรับการทำงานหลายๆอย่างดังรูปที่ 3.2



รูปที่ 3.3 การต่อ Digital Input [1]

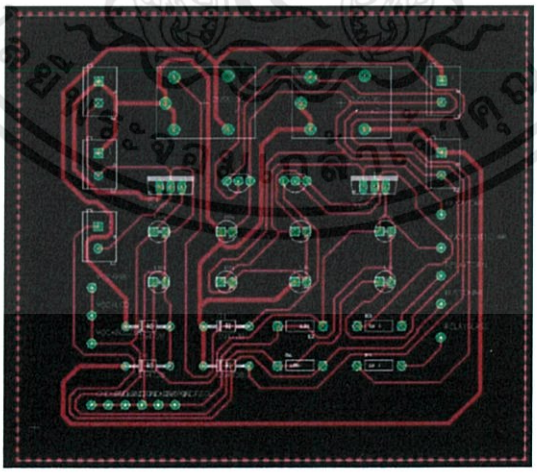
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากทำเป็นแบบจำลองดังนั้น เราจะใช้ Digital Input สำหรับการจำลองการเปิด-ปิด สวิตช์รถยนต์ และการกดหยุดเสียงแตรเมื่อจำนวนคน ขึ้น-ลง ไม่เท่ากัน โดยเราจะต่ออุปกรณ์ดังรูปที่ 3.3 เพื่อใช้เป็น Digital Input



รูปที่ 3.4 การออกแบบส่วนขยายใช้งาน ในโปรแกรม EAGLE

เนื่องจากต้องมีการใช้ Relay, Digital Input, LED, หน้าจอ LCD 20*4, BUZZLE, GSM module และ RFID อีกทั้งต้องการแปลงแรงดันไฟ 12 V DC เป็น 5 V DC ดังนั้นจึงต้องออกแบบส่วนขยายเพื่อช่วยให้มี Ground รวมทั้ง VCC สำหรับ module ต่างๆ ที่เพิ่มเข้ามาเพื่อให้เพียงพอต่อการทำงาน โดยจะเขียนลายวงจรเพิ่มเติม ดังรูปที่ 3.4 เพื่อใช้ในส่วนขยาย และทำการกัดลายวงจรตามรูปที่ 3.5 ลงบนแผ่น PCB

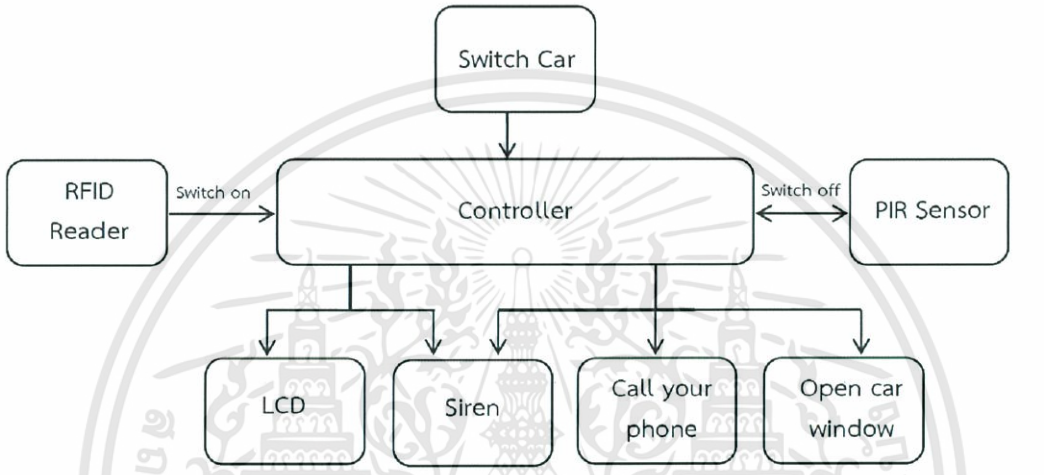


รูปที่ 3.5 ลายวงจรสำหรับลง PCB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 การออกแบบโปรแกรมการทำงาน

โดยการควบคุมการทำงานของ Microcontroller Atmega328p บนบอร์ด ARDUINO UNO นั้นจะใช้พื้นฐานการเขียนภาษา C ในการเขียนการควบคุมการทำงานของอุปกรณ์ทั้งหมด ด้วยโปรแกรม ARDUINO IDE โดยเริ่มสร้างแนวคิดของอุปกรณ์จาก Block Diagram ดังรูปที่ 3.6

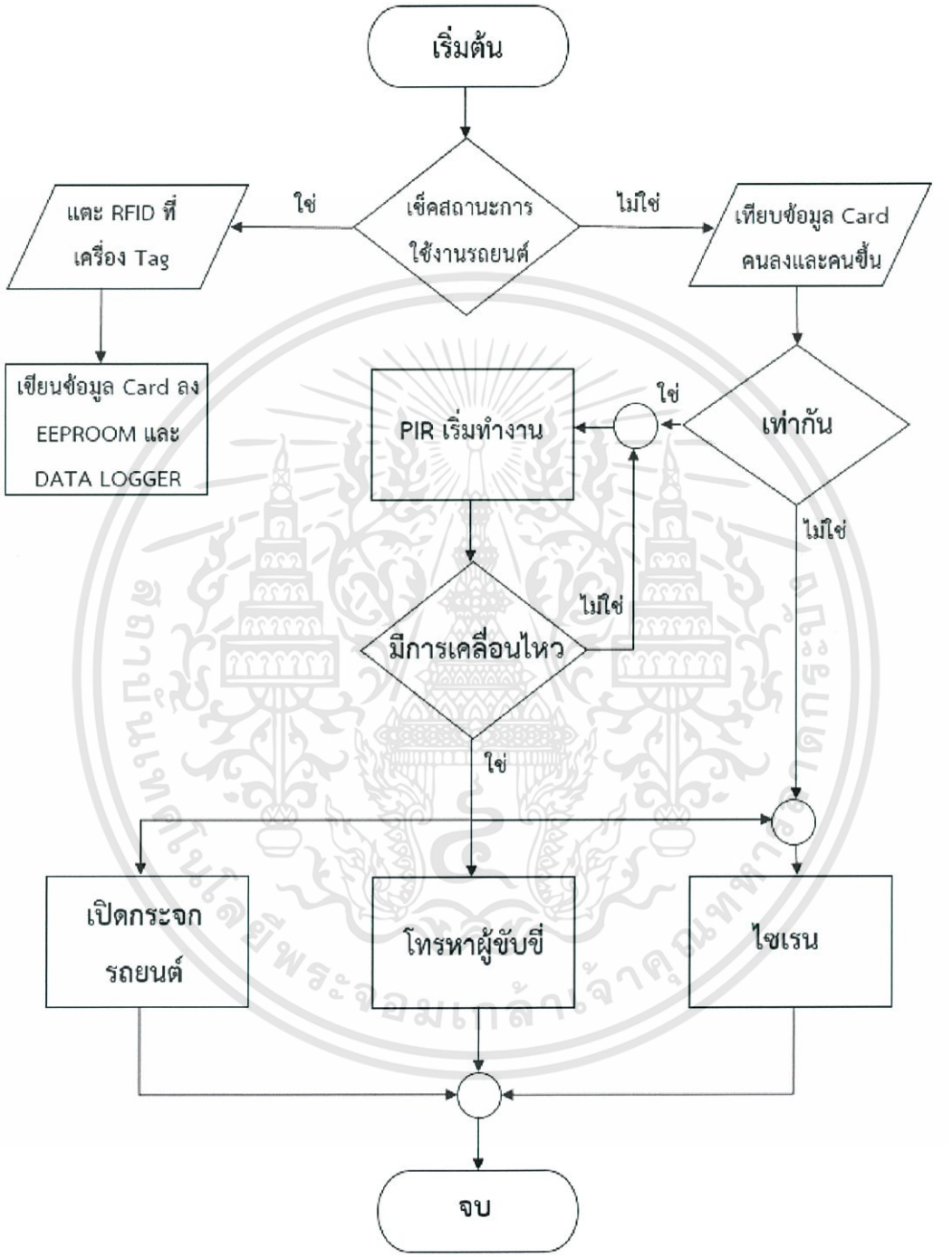


รูปที่ 3.6 บล็อกไดอะแกรมการทำงานรวมของระบบ

ปริญญานิพนธ์เป็นการสร้างอุปกรณ์ป้องกันเด็กติดภายในรถโดยสารไม่ประจำทาง โดยจะออกแบบและสร้างอุปกรณ์ตรวจนับจำนวนคนที่ขึ้น-ลง และเก็บประวัติการขึ้นลงของการ์ดแต่ละใบหรือแต่ละคนลง SD Card เมื่อจำนวนไม่เท่ากับจะทำการส่งเสียงเพื่อเตือนทันทีในกรณีเกิดความผิดพลาด เมื่อมีการลืม Tag ID ซึ่งอาจจะทำให้จำนวนคนขึ้น และลง เป็นศูนย์ จะทำการติด Sensor ตรวจจับการเคลื่อนไหวไว้ภายในรถ เพื่อตรวจสอบอีกขั้นเมื่อมีการติดภายในรถจะทำการเปิดกระจกรถในกรณีที่เป็นกระจกไฟฟ้า และโทรหาเจ้าของรถทันทีเผื่อแจ้งเตือน โดยอุปกรณ์ทั้งหมดจะใช้การควบคุมผ่าน microcontroller โดยการเขียนโปรแกรมควบคุมอุปกรณ์นั้นได้ทำตามรูปที่ 3.7 ซึ่งเป็นภาพรวมของการทำงานทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 Flow Chart



รูปที่ 3.7 Flowchart การทำงานของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 เครื่องมือที่ใช้ในการทดลอง

3.2.1 ARDUINO MEGA

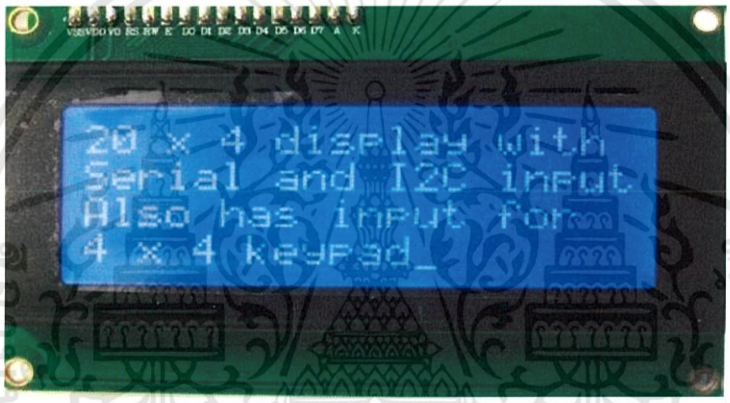
เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัวบอร์ด Arduino ดังรูปที่ 3.8 ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย ความง่ายของบอร์ด Arduino ในการต่ออุปกรณ์เสริมต่างๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิคส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขา I/O ของบอร์ด หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม (Arduino Shield) ประเภทต่างๆ แล้วเขียนโปรแกรมพัฒนาต่อได้เลย โดยใช้สำหรับควบคุมอุปกรณ์ครั้งนี้ทั้งหมด

รูปที่ 3.8 บอร์ด ARDUINO MEGA [1]

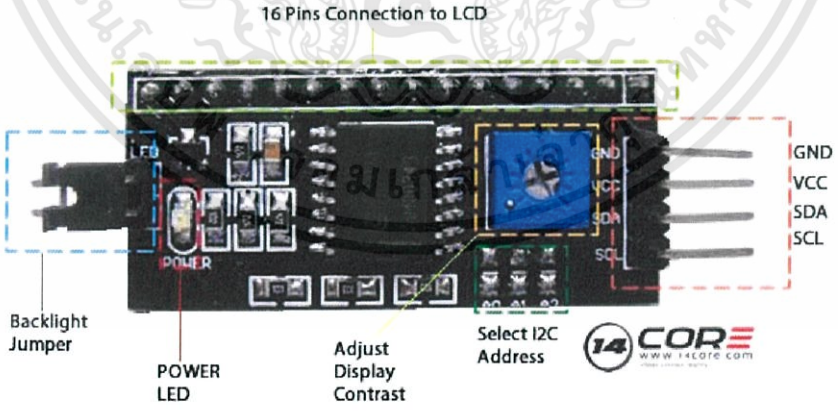
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 LCD 20*4 I2C

การทดลองครั้งนี้ใช้ เลือกใช้หน้าจอ LCD 20*4 ดังรูปที่ 3.9 เชื่อมต่อกับ ARDUINO ด้วย I2C ดังรูปที่ 3.10 เพื่อลดจำนวนการใช้งาน Port Digital ลง โดยใช้เพื่อแสดงผลจำนวนคนขึ้น-ลง ว่า ณ ปัจจุบันมีจำนวนเท่าใด I2C เป็นวิธีการส่งข้อมูลกันระหว่างไอซีหรือ Microcontroller (ทีละ 2 ตัว) มันสามารถติดต่อกันได้หลายตัว แต่คุยกันได้ทีละคู่โดยมีตัวออกคำสั่งว่าจะคุยกับใคร คือ Master และ ตัวที่สั่งให้ต้องตอบคือ Slave เป็นการสื่อสารอนุกรม แบบซิงโครนัส (Synchronous) เพื่อใช้ติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์ (MCU) กับอุปกรณ์ภายนอก โดยใช้สายสัญญาณเพียง 2 เส้นเท่านั้น คือ serial data (SDA) และสาย serial clock (SCL) ซึ่งสามารถเชื่อมต่ออุปกรณ์จำนวนหลายๆ ตัว เข้าด้วยกันได้ ทำให้ MCU ใช้พอร์ตเพียง 2 พอร์ตเท่านั้น



รูปที่ 3.9 LCD ขนาด 20*4 [3]

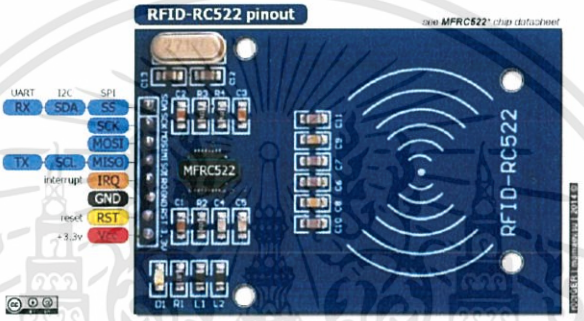


รูปที่ 3.10 Module I2C สำหรับต่อกับจอ LCD [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 RFID

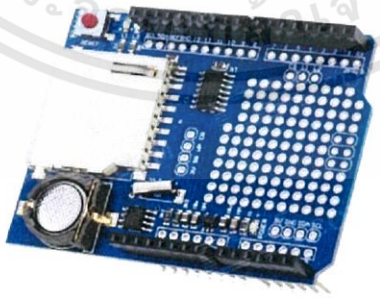
โดย module ที่เลือกใช้เป็นการสื่อสาร ระหว่าง Microcontroller ด้วย SPI คือ เป็นการเชื่อมต่อสื่อสารแบบอนุกรมโดยอาศัยสัญญาณนาฬิกาเป็นตัวกำหนดจังหวะการรับส่งข้อมูล (Synchronous) ที่สามารถส่งข้อมูลไปยังปลายทางและรับข้อมูลจากปลายทางกลับมาในครั้งเดียวกัน (Full Duplex) SPI แบ่งอุปกรณ์ออกเป็น 2 ฝั่ง คือ Master เป็นตัวควบคุมการรับส่งข้อมูล โดยในที่นี้คือไมโครคอนโทรลเลอร์ กับ Slave เป็นอุปกรณ์ที่รอรับคำสั่งจาก Master โดย Slave มีได้มากกว่า 1 ตัว โดย RC522 อ่านการ์ด ที่ความถี่ 13.56 MHz ระยะอ่านเขียนประมาณ 10 CM โดย RFID ที่ใช้มีลักษณะดังรูปที่ 3.11



รูปที่ 3.11 RFID (RC522) [2]

3.2.4 DATA LOGGER

ใช้ เพื่อเก็บข้อมูลการแตะการ์ด ของ RFID เพื่อเก็บประวัติการขึ้น-ลง ลงใน memory card โดยเก็บเป็นข้อมูล ID card และวันเวลาการแตะการ์ดของแต่ละใบด้วย โดย Module ที่ใช้เป็นการรวมระหว่าง Module RTC (Real time clock) กับ Module SD Card โดยเป็น Module สำเร็จ ดังรูปที่ 3.12



รูปที่ 3.12 Module Data logger [1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 Active Buzzer Module

โมดูล Active Buzze ดังรูปที่ 3.13 ใช้ไฟเลี้ยง 3.3 - 5V สามารถสร้างเสียงเตือนได้อย่างง่าย ๆ เพียงแค่จ่ายไฟ เข้าไปที่ ขา I/O โมดูลนี้มีทรานซิสเตอร์เบอร์ 9012 ช่วยขยายสัญญาณจึงมีความดังเป็นพิเศษ

รูปที่ 3.13 Active Buzzer Module [1]

3.2.6 DC DC-to-DC Step Down LM2596 Module

เป็นวงจรแปลงไฟ ดังรูปที่ 3.14 สำหรับลดแรงดันไฟฟ้า ระบบ Switching ควบคุมด้วย IC เบอร์ LM2596S รับไฟตั้งแต่ 4-40V เลือกปรับลดได้ตั้งแต่ 1.5V ถึง 35V กระแสใช้งานปกติ 2A วงจรสามารถตั้งค่าไฟขาออกตามต้องการได้ ด้วยการปรับค่า R-Trim port หลังจากตั้งค่าแล้ว วงจรจะทำการปรับแรงดันไฟฟ้าขาออกให้คงที่อัตโนมัติแม้แรงดันขาเข้าจะแปรผันก็ตาม

รูปที่ 3.14 DC-to-DC Step Down LM2596 Module [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.7 GSM Module SIM800L

บอร์ด SIM800L Module นี้มีขนาดเล็ก มาพร้อมเสาอากาศใช้งานกับ microSIM ดังรูปที่ 3.15 SIM800L Module สำหรับใช้ในการรับส่ง SMS โทรศัพท์หาเบอร์ที่ต้องการ เชื่อมต่ออินเตอร์เน็ตและอื่นๆ ได้เหมือนบอร์ด GSM รุ่นอื่นๆ แต่ไม่สามารถคุยได้ เพราะตัดขาลำโพงกับไมออก การต่อขาใช้เพียง 2 เส้นคือ RX และ TX ใช้ไฟที่ 3.7-4.2 โวลต์กระแสสูงสุด 2A ทำงานทันทีที่จ่ายไฟ มี LED แสดงผลสัญญาณ ถ้าจับสัญญาณโทรศัพท์ได้จะกระพริบช้า ๆ แต่ถ้าจับไม่ได้จะกระพริบถี่ ๆ บอร์ด SIM800L Module สามารถ ต่อกับ Battery Li-ion ได้โดยตรง

รูปที่ 3.15 GSM Module SIM800L [9]

3.2.8 PIR Motion Sensor Module

ตรวจจับความเคลื่อนไหวจากความร้อน เมื่อมีคนเดินผ่านก็จะจับค่าความร้อนที่เปลี่ยนแปลง แล้วส่งค่าสัญญาณมีไฟ ออกมา ในกรณีที่ไม่มีการเปลี่ยนแปลงส่งค่าไม่มีไฟ ออกมา เราสามารถนำค่านี้ไปสั่งควบคุม Arduino ได้ สามารถปรับเวลาหน่วงเวลาในการตรวจจับครั้งต่อไปได้ ปรับระยะทางการตรวจจับได้ 3-7 เมตร มีช่องให้ต่อ LDR เพิ่ม เพื่อให้ทำงานตรวจจับแค่ตอนกลางคืน โดยอุปกรณ์ที่ใช้จะมีลักษณะดังรูปที่ 3.16

รูปที่ 3.16 PIR Motion Sensor Module [7]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การจัดเก็บผลการทดลอง

ปริญญานิพนธ์นี้ได้ออกแบบขึ้นเพื่อ ป้องกันการเสียชีวิตของการติดอยู่ภายในรถ โดยการทดลองทั้งหมดนั้นได้ทดลองบน โฟโต้บอร์ดรวมทั้งกับทดลองภายในรถจริงโดยจะเก็บผลตามรายละเอียดดังต่อไปนี้

3.3.1 การเก็บผลเมื่อทดลองบนโฟโต้บอร์ด

โดยทำการเก็บการแตะการ์ด การแสดงผลบนหน้าจอ LCD สามารถแสดงจำนวน IN และ Out บนหน้าจอ LCD ตามการ์ดที่แตะเข้ามาได้ พร้อมทั้งทดสอบการแตะซ้ำของการ์ดใบเดิมภายในระยะเวลา 30 วินาที จะต้องไม่สามารถแตะการ์ดใบนั้นได้ และมีการจำลองการดับสวิตซ์รถเพื่อตรวจสอบสถานะของ Relay และ Module GSM โดย ไซเรนจะดังเมื่อจำนวน In != Out ส่วน PIR จะทำงานเมื่อ จำนวน In และ Out เป็นศูนย์

3.3.2 การเก็บผลเมื่อทดลองในรถจริง

ทดสอบอุปกรณ์ภายในรถยนต์โดยใช้ไฟเลี้ยงอุปกรณ์จากแบตเตอรี่รถยนต์ ขนาด 12 V ทดสอบกรณีดับสวิตซ์ และเปิดสวิตซ์ โดยกรณีเปิดสวิตซ์จะสามารถแตะการ์ดได้ ส่วนปิดสวิตซ์จะแตะการ์ดไม่ได้ จำลองสถานการณ์การลืมแตะการ์ดเมื่อลงจากรถ เมื่อจำนวน In != Out เมื่อดับสวิตซ์ จะทำให้ไซเรนดัง และจำลองสถานการณ์การติดภายในรถ เมื่อ In และ Out เป็นศูนย์ แต่ PIR ตรวจจับการเคลื่อนไหว กระจกรถต้องเปิด และมีการโทรออกจาก Module GSM ไปหาเจ้าของรถ

3.3.3 การเก็บผล PIR Output

นำ Oscilloscope วัดที่ขา Output ของ PIR Sensor เมื่อไม่มีการเคลื่อนไหวเกิดขึ้น Output จะออกมาเป็นแรงดัน ไฟ 0 V แต่ถ้ามีการเคลื่อนไหวเกิดขึ้น Output จะแสดงออกมาเป็นแรงดันไฟ 3.3 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

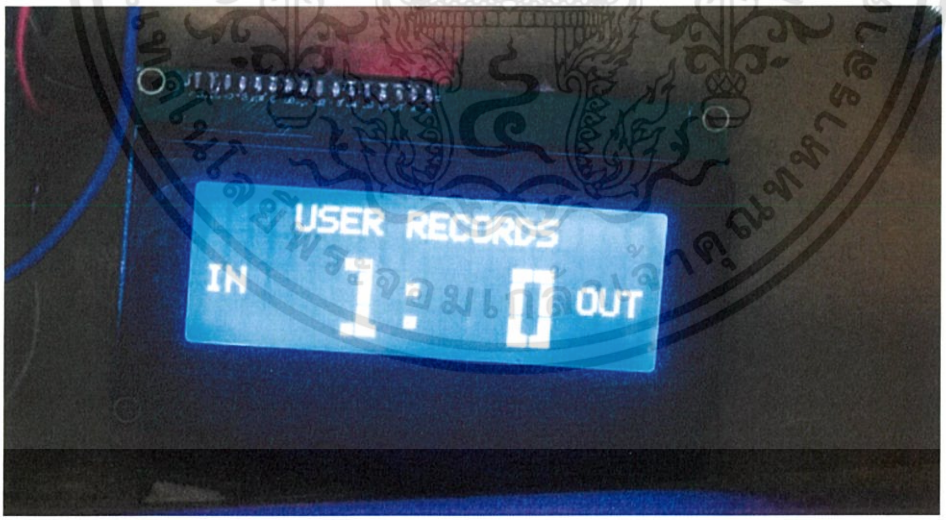
บทที่ 4

ผลการทดลอง

4.1 ทดสอบ สถานะ IN และ OUT เมื่อมีการแตะ Card เข้ามา

ทดสอบการ Tag card จะสามารถ Tag ได้ เมื่อดับสวิตช์จะไม่สามารถ Tag id card ได้ และจะไม่สามารถแตะการ์ดใบเดิมซ้ำภายในระยะเวลา 30 วินาที โดยแสดงผลบน Serial monitor ดังรูปที่ 4.1 และ แสดงค่าคนขึ้นและลงบนหน้าจอ LCD ดังรูปที่ 4.2

รูปที่ 4.1 Serial monitor เมื่อมีการแตะการ์ดเข้ามา

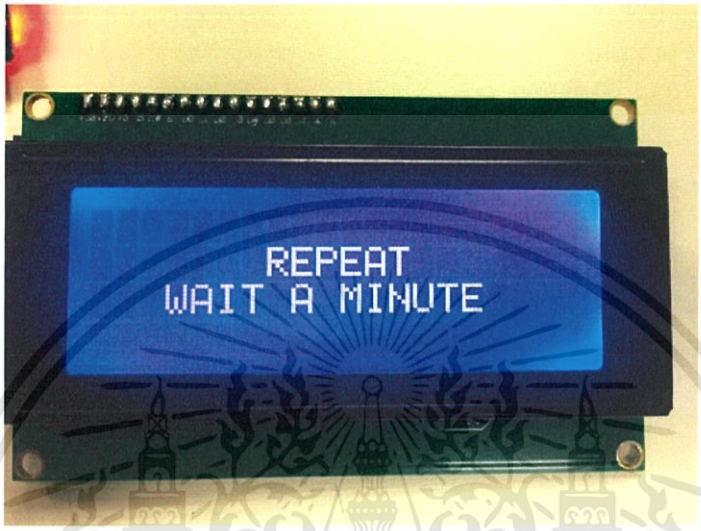


รูปที่ 4.2 ค่าบน LCD เมื่อมีการแตะการ์ดเข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ทดสอบเมื่อ มีการแตะการ์ดซ้ำ

เมื่อมีการ Tag Card ใบเดิมภายในระยะเวลา 30 วินาที จะไม่ยอมให้แตะ Card ใบนั้นและแสดงผลบนหน้าจอ LCD เพื่อเตือนว่ามีการซ้ำของ Card ดังรูปที่ 4.3



รูปที่ 4.3 ค่าบนหน้าจอเมื่อมีการแตะการ์ดซ้ำเข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ทดสอบเมื่อดับสวิตซ์

เมื่อมีการดับสวิตซ์กุญแจรถยนต์จะไม่สามารถแตะการ์ดเข้ามาได้ เมื่อดับสวิตซ์รถยนต์จะแบ่งเป็นสองกรณี กรณีแรกเมื่อจำนวน In = Out จะเริ่มสั่งให้ PIR ทำงาน ดังรูปที่ 4.4 เพื่อป้องกันการติดอยู่ในรถ กรณีที่สอง เมื่อ In != Out จะสั่งให้ ไซเรนดัง แสดง Serial print เป็นคำว่า Horn ดังรูปที่ 4.5

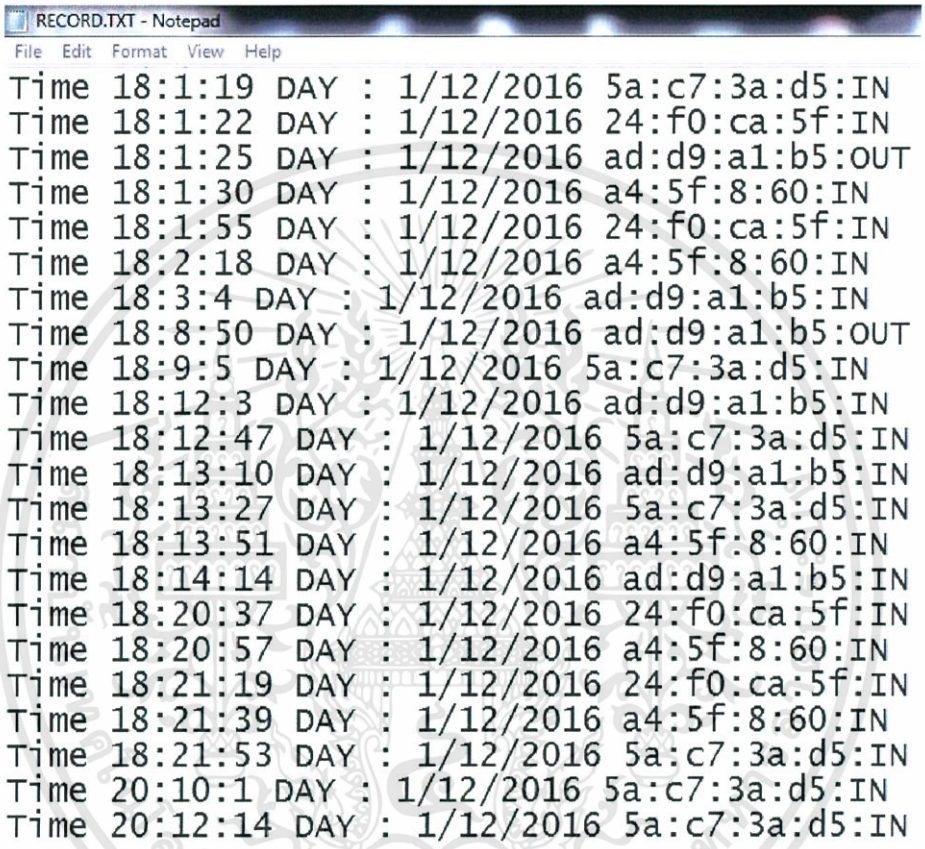
รูปที่ 4.4 Serial monitor เมื่อมีการดับสวิตซ์ไฟรถยนต์ กรณี In = out

รูปที่ 4.5 Serial monitor เมื่อมีการดับสวิตซ์รถยนต์ กรณี In != out

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ทดสอบการเขียนลงบน SD CARD

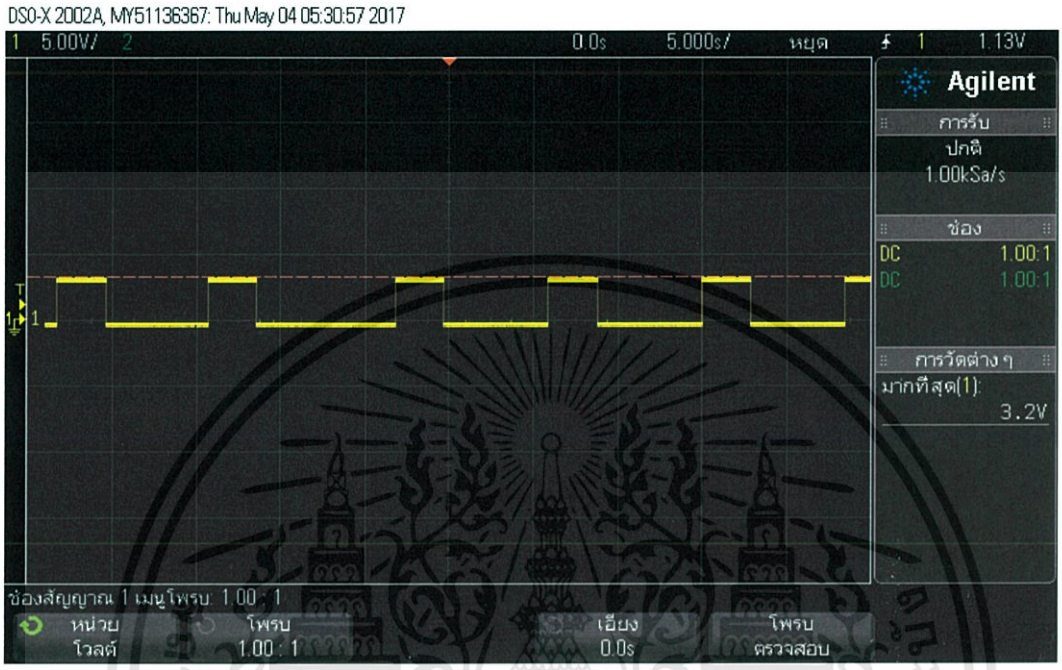
ทำการเก็บประวัติข้อมูลวันเวลาการขึ้น-ลง ของ Card ทุกใบ โดยเก็บรหัสของ Card แต่ละใบ วัน เวลา รวมทั้งสถานะ IN หรือ OUT ไว้ภายใน SD Card เพื่อเก็บไว้ทำเป็นสถิติ โดยจะเขียนข้อมูลลงไฟล์ ประเภท .TXT ลงใน SD CARD ดังรูปที่ 4.6



รูปที่ 4.6 การเก็บประวัติขึน-ลงไว้ภายใน SD CARD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ทดสอบ PIR สถานะทำงาน และ ไม่ทำงาน



รูปที่ 4.7 สัญญาณ Output จาก PIR Sensor

จากรูปที่ 4.7 ตามทฤษฎี output ของ PIR จะแสดง ออกมาเป็น TTL ขนาด 3.2 V โดยเลือกโหมดการ Trig สัญญาณ แบบ Repeat Trigger หมายถึงเมื่อมีการเคลื่อนไหว จะทำการ Trig ตลอดเวลา จากรูป จะเห็นว่ามีสัญญาณสองแบบ คือกรณีที่ สัญญาณมีขนาด 3.2 V และ 0 V โดยเมื่อมีการเคลื่อนไหว PIR จะมี Output เป็นแรงดัน ไฟ 3.2 V และเมื่อไม่มีการเคลื่อนไหว PIR จะมี Output ออกมาเป็น 0 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

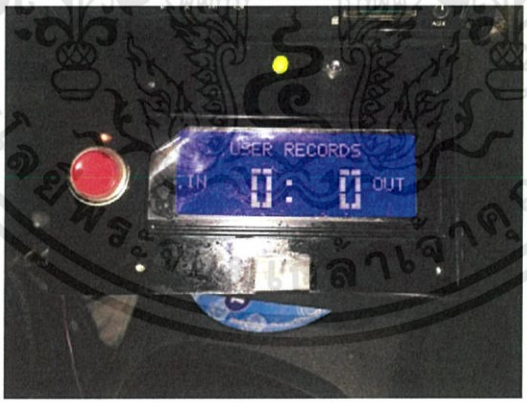
4.6 ทดสอบอุปกรณ์รวมภายในรถยนต์

การทดสอบอุปกรณ์ที่ทำเสร็จแล้วภายในรถจริง เพื่อตรวจสอบความพร้อมอุปกรณ์เพื่อให้เกิดความผิดพลาดน้อยที่สุด

4.6.1 การทดสอบ RFID เมื่อมีการเปิด switch รถ

รูปที่ 4.8 เมื่อมีการแตะการ์ด

จากรูปที่ 4.8 เมื่อเปิดสวิตช์กุญแจรถ จะสามารถแตะการ์ดได้ LCD จะแสดงจำนวน In และ Out เมื่อ In != Out LED จะแสดงสถานะสีแดง ดังรูปที่ 4.8 เมื่อเราดับสวิตช์กุญแจ ในกรณีตามภาพ ไซเรนจะดังทันทีถ้าต้องการหยุดเสียงแตรจะต้องทำการกดปุ่มสีแดง ซึ่งเป็นการเคลียร์ค่า In และ Out ให้เป็นศูนย์ หรือ มีการแตะการ์ดทำให้ In = Out



รูปที่ 4.9 เมื่อจำนวน In = Out

เมื่อจำนวนการแตะการ์ด ทำให้ In = Out Led จะเปลี่ยนสถานะจากแดงเป็นเขียว และรอสักพักจำนวน In และ Out จะเปลี่ยนเป็นศูนย์ ดังรูปที่ 4.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6.2 การทดสอบเมื่อจำนวนคนขึ้นและลง เป็นศูนย์ แล้วมีการเคลื่อนไหวเกิดขึ้นบนรถ

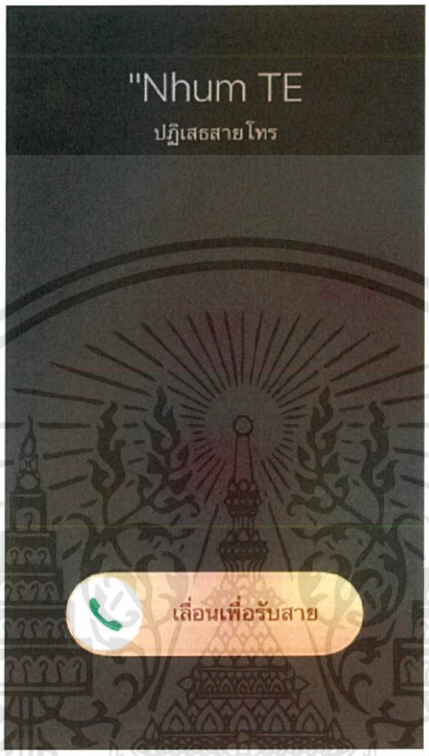
เมื่อมีการดับสวิตช์กุญแจรถยนต์แล้ว จำนวน In และ Out เป็นศูนย์ PIR จะเริ่มทำงานเพื่อป้องกันการติดอยู่ภายในรถอีกชั้นหนึ่ง เมื่อมีการเคลื่อนไหวภายในรถ อุปกรณ์จะสั่งให้ไซเรนดัง เปิดกระจกรถ ดังรูปที่ 4.10 และโทรหาเจ้าของรถทันที ดังรูปที่ 4.11

รูปที่ 4.10 กระจกไฟฟ้าถูกเปิดเองอัตโนมัติ

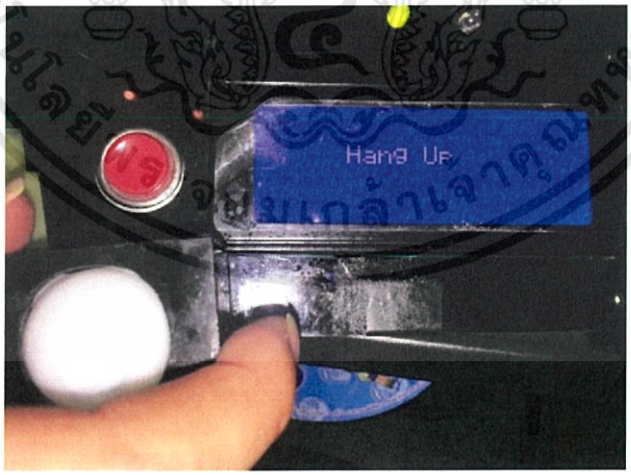
รูปที่ 4.11 LCD แสดงผลบอกว่า Module GSM กำลังโทรออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อมีการโทรหาผู้ขับขี่ดังรูปที่ 4.12 การที่จะหยุดการโทรออกจากอุปกรณ์ได้นั้น ผู้ขับขี่ต้องกลับมาที่ตัวอุปกรณ์เพื่อกดปุ่มบนอุปกรณ์เพื่อหยุดเสียงไซเรนและโทรออก ดังรูปที่ 4.13

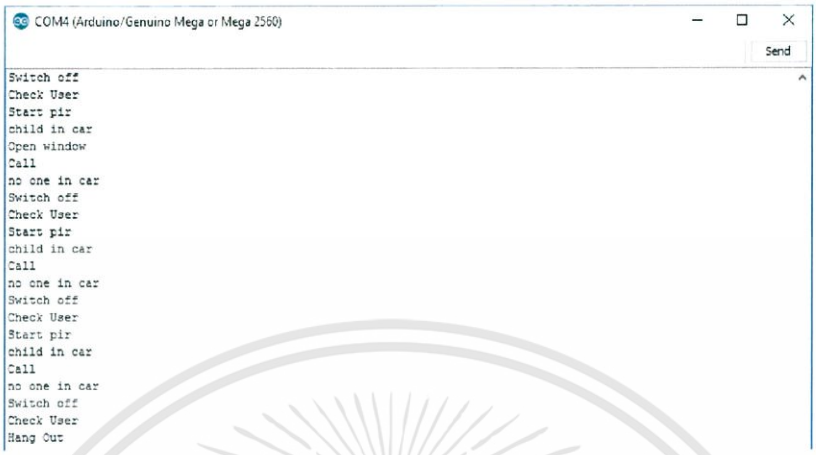


รูปที่ 4.12 มีการโทรเข้าจาก Module GSM



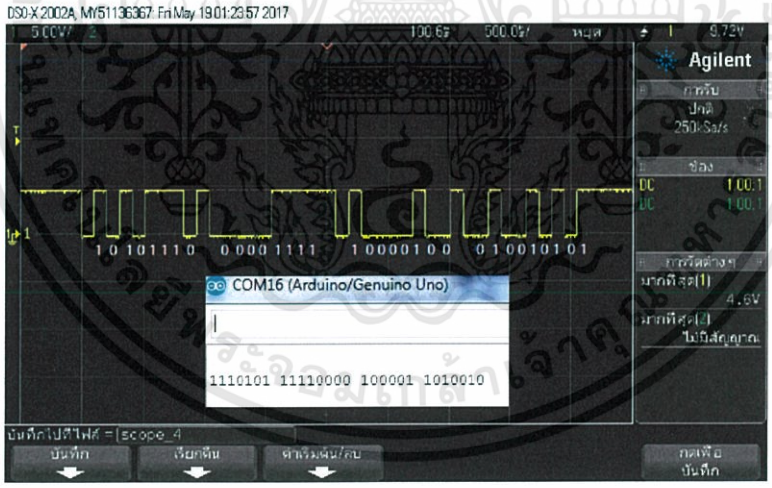
รูปที่ 4.13 การกดวางสายจากตัวอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 Serial monitor เมื่อดับกุญแจรถและจำนวน In และ Out เป็นศูนย์

จากรูปที่ 4.14 จะเห็นได้ชัดเจนของการทำงานของอุปกรณ์เมื่อดับกุญแจรถ และจำนวน In และ Out เป็นศูนย์เท่ากัน จะเห็นว่า PIR จะเริ่มทำงานเมื่อมีการเคลื่อนไหวจะสั่งเปิดกระจกรถก่อน หลังจากนั้นจะโทรหาคนขับทันที โดยจะหยุดการโทรได้นั้นต้องมากดปุ่มที่ตัวอุปกรณ์เท่านั้น



รูปที่ 4.15 UID ของ Tag Card

จากรูปที่ 4.15 แสดง bit ข้อมูลของ UID ของการ์ดแต่ละใบโดยข้อมูล 1 ชุดมี 10 bit มี 1 start bit และ 1 stop bit และข้อมูล 8 bit โดย bit ซ้ายสุดจะเป็น LSB ส่วนขวาสุดจะเป็น MSB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ปริญญานิพนธ์เรื่องนี้เป็นการสร้าง ระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์ ประสบความสำเร็จในการจัดทำระบบ โดยระบบนั้นได้มีการป้องกันและแจ้งเตือน โดยมีการใช้ระบบอาร์เอฟไอดีในการตรวจนับคนขึ้นลงรถยนต์ ซึ่งถ้ามีการขึ้นลงของคนภายในรถยนต์ไม่เท่ากัน จะมีการส่งเสียงแจ้งเตือนผ่านทางไซเรนที่ติดเพิ่มและจะหยุดเมื่อมีการรีเซตระบบที่ตัวอุปกรณ์ กรณีที่จำนวนคนขึ้นและลงเท่ากัน ทางผู้จัดทำได้ใส่ PIR sensor เพื่อป้องการลืม Tag id และกรณีที่มีการติดภายในรถจริงๆ จะทำการเปิดกระจกรถ และทำการโทรไปแจ้งเจ้าของรถผ่านโมดูล GSM และจะหยุดโทรเมื่อมีการกดรีเซตที่ตัวอุปกรณ์เพื่อจะสามารถให้เจ้าของรถกลับมาเช็คว่ามีคนติดอยู่ในรถจริงหรือไม่

5.2 ข้อเสนอแนะ

ระบบป้องกันและแจ้งเตือนเด็กติดภายในรถยนต์ ที่ถูกออกแบบและจัดทำขึ้นมาในปริญญานิพนธ์นี้ นั้นยังสามารถนำไปพัฒนาและปรับปรุงคุณภาพให้ดียิ่งขึ้นได้อีก ซึ่งจากการทดลองการทำงานของระบบนั้น ถ้าผู้ใช้งานเกิดปัญหาลืมแท็กไอดีมา จะไม่สามารถใช้งานระบบนี้ได้ อาจก่อเกิดปัญหาเรื่องความปลอดภัยของผู้ใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] บริษัท วีนัส ซัพพลาย จำกัด. “Arduino คืออะไร.” http://www.thaieasyelec.com/article-wiki/basic-electronics/บทความ-arduino-คืออะไร-เริ่มต้นใช้งาน-arduino.html.

[2] บริษัท วีนัส ซัพพลาย จำกัด. “RFID คืออะไร? ตอนที่ 1 ความรู้เบื้องต้นเกี่ยวกับ RFID.” http://thaieasyelec.com/article-wiki/basic-electronics/rfid-basic.html.

[3] บริษัท วีนัส ซัพพลาย จำกัด. “การใช้งาน Character LCD Display กับ Arduino.” http://thaieasyelec.com/article-wiki/basic-electronics/การใช้งาน-character-lcd-display-กับ-arduino-ตอนที่1-รูปแบบการเชื่อมต่อแบบ-parallel.html.

[4] วัชรากร หนูทอง. “RFID เทคโนโลยีสารพัดประโยชน์.” สาร NECTEC.1 (กันยายน 2547) : 15-22.

[5] แผนกวิชาช่างยนต์. “อุปกรณ์พื้นฐานทางไฟฟ้ารถยนต์.” ใบความรู้ที่ 1.2 .1 (มีนาคม 2555) : 1-28.

[6] แผนกวิชาช่างยนต์. “วงจรแตรไฟเบรกและไฟถอย.” ใบความรู้ที่ 3.2 .1 (ตุลาคม 2556) : 1-7.

[7] บริษัท วีนัส ซัพพลาย จำกัด. “PIR Motion Sensor Getting Started.” http://www.thaieasyelec.com/article-wiki/review-product-article/pir-motion-sensor-getting-started.html.

[8] ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ. “การตรวจจับการเคลื่อนไหวด้วย AMN34112 PIR Motion Sensor.” http://cpre.kmutnb.ac.th/esl/learning/index.php?article=amn34112-pir-motion-sensor.

[9] บริษัท Arduitronics จำกัด. “Introduction to Arduino GPRS/GSM Module.” https://www.arduitronics.com/article/44/introduction-to-arduino-gprs-gsm-module.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

ชุดคำสั่งการทำงานของ MICROCONTROLLER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <EEPROM.h>     // We are going to read and write PICC's UIDs from/to EEPROM
#include <SPI.h>        // RC522 Module uses SPI protocol
#include <MFRC522.h>  // Library for Mifare RC522 Devices
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DS1307RTC.h>
#include<TimeLib.h>
#include <SD.h>
#include "RTClib.h"
#include "SIM900.h"
#include <SoftwareSerial.h>
#include "call.h"

CallGSM call;

RTC_DS1307 rtc;
File myFile;
const int chipSelect = 10;
String nhum;
String time ;
String file;
int rec;
volatile bool interrupt = false;
volatile bool mirror = true;
volatile bool checktime = true;
volatile bool checkpir = true;
tmElements_t tm;
// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x3F, 20, 4);
byte x10[8] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x07, 0x07};
byte x11[8] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x1C, 0x1C, 0x1C};
byte x12[8] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x1F, 0x1F, 0x1F};
byte x13[8] = {0x07, 0x07, 0x07, 0x07, 0x07, 0x1F, 0x1F, 0x1F};
byte x14[8] = {0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1F, 0x1F, 0x1F};
byte x15[8] = {0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C};
byte x16[8] = {0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07, 0x07};
byte x17[8] = {0x00, 0x00, 0x0E, 0x0E, 0x0E, 0x00, 0x00, 0x00};

byte row = 0, col = 0;
int button = 18, LED1 = 22, LED2 = 23, buzzer = 24, relay1 = 25, relay2 = 26, digitalin = 27,
digitalin1 = 28;



volatile int yy, mm, dd, hh, mi, ss;

int comein, comeout, comein1, comein2;


boolean match = false;          // initialize card match to false
boolean programMode = false;  // initialize programming mode to false
int successRead;    // Variable integer to keep if we have Successful Read from Reader
int cardname[20];
byte storedCard[4];   // Stores an ID read from EEPROM
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
byte readCard[4];   // Stores scanned ID read from RFID Module
// Stores master card's ID read from EEPROM
//////////////////time
// Date and time functions using a DS1307 RTC connected via I2C and Wire lib


char daysOfTheWeek[7][12] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday"};

//////////////////time
/*
  We need to define MFRC522's pins and create instance
  Pin layout should be as follows (on Arduino Uno):
  MOSI: Pin 11 / ICSP-4
  MISO: Pin 12 / ICSP-1
  SCK : Pin 13 / ICSP-3
  SS : Pin 10 (Configurable)
  RST : Pin 9 (Configurable)
  look MFRC522 Library for
  other Arduinos' pin configuration
*/

#define SS_PIN 9
#define RST_PIN 8
MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance.
int num = EEPROM.read(0);
int glass = EEPROM.read(4092);
int glass1 = EEPROM.read(4093);

int in = EEPROM.read(4094);
int out = EEPROM.read(4095);


/////////////////////////////////////// Setup /////////////////////////////////////
void setup() {
  Serial.begin(9600);  // Initialize serial communications with PC
  lcd.begin();
  lcd.clear();
  lcd.setCursor(5, 1);
  lcd.print(F("Wait GSM"));
  gsm.begin(2400);
  lcd.clear();
  lcd.setCursor(5, 1);
  lcd.print(F("GSM Start"));
  delay(1000);
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(buzzer, OUTPUT);
  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);
  pinMode(digitalin, INPUT);
  pinMode(digitalin1, INPUT);
  pinMode(button, INPUT_PULLUP);
  /*pinMode(button,OUTPUT);*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
attachInterrupt(digitalPinToInterrupt(button), Inter, FALLING);
Serial.begin(9600);

while (!Serial) ; // wait for serial
delay(200);
Serial.println(F("Alarm system for children trapped in car"));
Serial.println("");
Serial.println(F("--------------------"));
pinMode(SS, OUTPUT);
SPI.begin();
mfrc522.PCD_Init();

if (!SD.begin(10, 11, 12, 13)) {
  Serial.println("SD Card initialization failed!");
  return;
}
Serial.println("SD Card OK.");
lcd.setCursor(5, 1);
lcd.print(F("SD Card OK"));
delay(1000);
delay(1000);
if (! rtc.begin()) {
  Serial.println("Couldn't find RTC");
  Serial.println("RTC is NOT running!");
  lcd.clear();
lcd.setCursor(5, 1);
lcd.print(F("Check RTC With Computer"));
  while (1);
}

if (! rtc.isrunning()) {
  Serial.println("RTC is NOT running!");
  lcd.clear();
lcd.setCursor(5, 1);
lcd.print(F("Check RTC With Computer"));
  // following line sets the RTC to the date & time this sketch was compiled
  // rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  // This line sets the RTC with an explicit date & time, for example to set
  // January 21, 2014 at 3am you would call:
  //rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
}


//////////////////////time
// setup LCD  for 16 columns and 2 rows
lcd.createChar(0, x10);                 // digit piece
lcd.createChar(1, x11);                 // digit piece
lcd.createChar(2, x12);                 // digit piece
lcd.createChar(3, x13);                 // digit piece
lcd.createChar(4, x14);                 // digit piece
lcd.createChar(5, x15);                 // digit piece
lcd.createChar(6, x16);                 // digit piece
lcd.createChar(7, x17);                 // digit piece (colon)

row = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
void doNumber(byte num, byte r, byte c) {
  lcd.setCursor(c, r);
  switch (num) {
    case 0: lcd.write(byte(2)); lcd.write(byte(2));
      lcd.setCursor(c, r + 1); lcd.write(byte(5)); lcd.write(byte(6));
      lcd.setCursor(c, r + 2); lcd.write(byte(4)); lcd.write(byte(3)); break;

    case 1: lcd.write(byte(0)); lcd.write(byte(1));
      lcd.setCursor(c, r + 1); lcd.print(" "); lcd.write(byte(5));
      lcd.setCursor(c, r + 2); lcd.write(byte(0)); lcd.write(byte(4)); break;

    case 2: lcd.write(byte(2)); lcd.write(byte(2));
      lcd.setCursor(c, r + 1); lcd.write(byte(2)); lcd.write(byte(3));
      lcd.setCursor(c, r + 2); lcd.write(byte(4)); lcd.write(byte(2)); break;

    case 3: lcd.write(byte(2)); lcd.write(byte(2));
      lcd.setCursor(c, r + 1); lcd.write(byte(0)); lcd.write(byte(3));
      lcd.setCursor(c, r + 2); lcd.write(byte(2)); lcd.write(byte(3)); break;

    case 4: lcd.write(byte(1)); lcd.write(byte(0));
      lcd.setCursor(c, r + 1); lcd.write(byte(4)); lcd.write(byte(3));
      lcd.setCursor(c, r + 2); lcd.print(" "); lcd.write(byte(6)); break;

    case 5: lcd.write(byte(2)); lcd.write(byte(2));
      lcd.setCursor(c, r + 1); lcd.write(byte(4)); lcd.write(byte(2));
      lcd.setCursor(c, r + 2); lcd.write(byte(2)); lcd.write(byte(3)); break;
    case 6: lcd.write(byte(1)); lcd.print(" ");
      lcd.setCursor(c, r + 1); lcd.write(byte(4)); lcd.write(byte(2));
      lcd.setCursor(c, r + 2); lcd.write(byte(4)); lcd.write(byte(3)); break;

    case 7: lcd.write(byte(2)); lcd.write(byte(2));
      lcd.setCursor(c, r + 1); lcd.print(" "); lcd.write(byte(6));
      lcd.setCursor(c, r + 2); lcd.print(" "); lcd.write(byte(6)); break;

    case 8: lcd.write(byte(2)); lcd.write(byte(2));
      lcd.setCursor(c, r + 1); lcd.write(byte(4)); lcd.write(byte(3));
      lcd.setCursor(c, r + 2); lcd.write(byte(4)); lcd.write(byte(3)); break;

    case 9: lcd.write(byte(2)); lcd.write(byte(2));
      lcd.setCursor(c, r + 1); lcd.write(byte(4)); lcd.write(byte(3));
      lcd.setCursor(c, r + 2); lcd.print(" "); lcd.write(byte(6)); break;

    case 10: lcd.setCursor(c, r + 1); lcd.write(byte(7)); lcd.setCursor(c, r + 2);
lcd.write(byte(7)); break;

  }
}
int getID() {

  // Getting ready for Reading PICCs
  if ( ! mfrc522.PICC_IsNewCardPresent()) { //If a new PICC placed to RFID reader continue
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    return 0;
  }
  if ( ! mfrc522.PICC_ReadCardSerial()) {   //Since a PICC placed get Serial and continue
    return 0;
  }
  // There are Mifare PICCs which have 4 byte or 7 byte UID care if you use 7 byte PICC
  // I think we should assume every PICC as they have 4 byte UID
  // Until we support 7 byte PICCs
  for (int i = 0; i < 4; i++) {  //
    readCard[i] = mfrc522.uid.uidByte[i];

  }
  mfrc522.PICC_HaltA(); // Stop reading
  return 1;
}
void writeID( byte a[] ) {

  DateTime now = rtc.now();
  yy = (now.year() % 100);
  mm = (now.month());
  dd = (now.day());
  hh = (now.hour());
  mi = (now.minute());
  ss = (now.second());



  if (!findID( a ))
  { // Before we write to the EEPROM, check to see if we have seen this card before!
    digitalWrite(buzzer, HIGH);
    delay(100);
    digitalWrite(buzzer, LOW);
    delay(100);
    digitalWrite(buzzer, HIGH);
    delay(100);
    digitalWrite(buzzer, LOW);

    num = EEPROM.read(0);
    in = EEPROM.read(4094);
    out = EEPROM.read(4095);
    int start = ( num * 12 ) + 1;  // Figure out where the next slot starts
    ////////////////////////////////////timec=ShowSeconds();
    comein = 1;
    comeout = 0;
    EEPROM.write((num * 12) + 5, yy);
    EEPROM.write((num * 12) + 6, mm);
    EEPROM.write((num * 12) + 7, dd);
    EEPROM.write((num * 12) + 8, hh);
    EEPROM.write((num * 12) + 9, mi);
    EEPROM.write((num * 12) + 10, ss);
    EEPROM.write((num * 12) + 11, comein);
    EEPROM.write((num * 12) + 12, comeout);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    num = EEPROM.read(0);     // Get the numer of used spaces, position 0 stores the
 number of ID cards
    num++;                // Increment the counter by one
    in++;
    if (((in % 10) > 0) && ((in / 10) >= 1))
    {
      doNumber( 1, row, 3);                    // static
      doNumber((in % 10), row, 6);
      doNumber(10, row, 9);
    }
    else
    {
      doNumber(in, row, 6);
      doNumber(10, row, 9);
    }
    if (((out % 10) > 0) && ((out / 10) >= 1))
    {
      doNumber( 1, row, 11);
      doNumber((out % 10), row, 14);
      doNumber(10, row, 9);
    }
    else
    {
      doNumber(out, row, 14);
      doNumber(10, row, 9);
    }

    EEPROM.write( 4094, in );


    EEPROM.write( 0, num );     // Write the new count to the counter
    file = "";
    for ( int j = 0; j < 4; j++ )
    { // Loop 4 times
      EEPROM.write( start + j, a[j] );  // Write the array values to EEPROM in the right
 position
      rec = a[j];
      file += String(rec, HEX);
      file += ":";
    }
    file += "IN";
    gettime(file);


    successRead = 0;


  } else
  {
    deleteID(readCard);
    successRead = 0;

  }
}
void deleteID( byte a[] ) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  DateTime now = rtc.now();
  num = EEPROM.read(0);   // Get the numer of used spaces, position 0 stores the number
of ID cards
  in = EEPROM.read(4094);
  out = EEPROM.read(4095);
  int slot;       // Figure out the slot number of the card
  int start;      // = ( num * 4 ) + 6; // Figure out where the next slot starts
  int looping;    // The number of times the loop repeats
  int j, tr;
  slot = findIDSLOT( a );   // Figure out the slot number of the card to delete
  int yy1 = (now.year() % 100);
  int mm1 = (now.month());
  int dd1 = (now.day());
  int hh1 = (now.hour());
  int mi1 = (now.minute());
  int ss1 = (now.second());

  int yy2 = EEPROM.read((slot * 12) + 5);
  int mm2 = EEPROM.read((slot * 12) + 6);
  int dd2 = EEPROM.read((slot * 12) + 7);
  int hh2 = EEPROM.read((slot * 12) + 8);
  int mi2 = EEPROM.read((slot * 12) + 9);
  int ss2 = EEPROM.read((slot * 12) + 10);
  int comein1 = EEPROM.read((slot * 12) + 11);
  int comeout1 = EEPROM.read((slot * 12) + 12);
  int total = (yy1 * 365 * 24 * 60 * 60) + (mm1 * 30 * 24 * 60 * 60) + (dd1 * 24 * 60 * 60)
+ (hh1 * 60 * 60) + (mi1 * 60) + ss1;
  int total1 = (yy2 * 365 * 24 * 60 * 60) + (mm2 * 30 * 24 * 60 * 60) + (dd2 * 24 * 60 * 60)
+ (hh2 * 60 * 60) + (mi2 * 60) + ss2;



  if ((total - total1) > 30)
  {
    digitalWrite(buzzer, HIGH);
    delay(100);
    digitalWrite(buzzer, LOW);
    delay(100);
    digitalWrite(buzzer, HIGH);
    delay(100);
    digitalWrite(buzzer, LOW);

    if ((comein1 == 1))
    {
      comeout1 = 1;
      comein1 = 0;
      EEPROM.write((slot * 12) + 5, yy1);
      EEPROM.write((slot * 12) + 6, mm1);
      EEPROM.write((slot * 12) + 7, dd1);
      EEPROM.write((slot * 12) + 8, hh1);
      EEPROM.write((slot * 12) + 9, mi1);
      EEPROM.write((slot * 12) + 10, ss1);
      EEPROM.write((slot * 12) + 11, comein1);
      EEPROM.write((slot * 12) + 12, comeout1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  out++;
  EEPROM.write( 4095, out );
  file = "";
  for ( int j = 0; j < 4; j++ )
  { // Loop 4 times
    // Write the array values to EEPROM in the right position
    rec = a[j];
    file += String(rec, HEX);
    file += ":";
  }
  file += "OUT";
  gettime(file);
  if (((in % 10) > 0) && ((in / 10) >= 1))
  {
    doNumber( 1, row, 3);                    // static
    doNumber((in % 10), row, 6);
    doNumber(10, row, 9);

  }
  else
  {
    doNumber(in, row, 6);
    doNumber(10, row, 9);

  }
  if (((out % 10) > 0) && ((out / 10) >= 1))
  {
    doNumber( 1, row, 11);
    doNumber((out % 10), row, 14);
    doNumber(10, row, 9);

  }
  else
  {
    doNumber(out, row, 14);
    doNumber(10, row, 9);

  }

}
else if (comein1 == 0)
{
  comein1 = 1;
  comeout1 = 0;
  EEPROM.write((slot * 12) + 5, yy1);
  EEPROM.write((slot * 12) + 6, mm1);
  EEPROM.write((slot * 12) + 7, dd1);
  EEPROM.write((slot * 12) + 8, hh1);
  EEPROM.write((slot * 12) + 9, mi1);
  EEPROM.write((slot * 12) + 10, ss1);
  EEPROM.write((slot * 12) + 11, comein1);
  EEPROM.write((slot * 12) + 12, comeout1);
  in++;
  EEPROM.write( 4094, in );
  file = "";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  for ( int j = 0; j < 4; j++ )
  { // Loop 4 times
    rec = a[j];
    file += String(rec, HEX);
    file += ":";
  }
  file += "IN";
  gettime(file);

  if (((in % 10) > 0) && ((in / 10) >= 1))
  {
    doNumber( 1, row, 3);                    // static
    doNumber((in % 10), row, 6);
    doNumber(10, row, 9);

  }
  else
  {
    doNumber(in, row, 6);
    doNumber(10, row, 9);

  }
  if (((out % 10) > 0) && ((out / 10) >= 1))
  {
    doNumber( 1, row, 11);
    doNumber((out % 10), row, 14);
    doNumber(10, row, 9);

  }
  else
  {
    doNumber(out, row, 14);
    doNumber(10, row, 9);

  }

}
else
{
  if (((in % 10) > 0) && ((in / 10) >= 1))
  {
    doNumber( 1, row, 3);                    // static
    doNumber((in % 10), row, 6);
    doNumber(10, row, 9);

  }
  else
  {
    doNumber(in, row, 6);
    doNumber(10, row, 9);

  }
  if (((out % 10) > 0) && ((out / 10) >= 1))
  {
    doNumber( 1, row, 11);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      doNumber((out % 10), row, 14);
      doNumber(10, row, 9);

    }
    else
    {
      doNumber(out, row, 14);
      doNumber(10, row, 9);

    }
    Serial.println("");
    Serial.print(" user record : ");
    Serial.print(num);
    Serial.println(" Card");

  }

} else
{
  digitalWrite(buzzer, HIGH);
  delay(100);
  digitalWrite(buzzer, LOW);
  delay(100);
  digitalWrite(buzzer, HIGH);
  delay(100);
  digitalWrite(buzzer, LOW);
  delay(100);
  digitalWrite(buzzer, HIGH);
  delay(100);
  digitalWrite(buzzer, LOW);
  delay(100);
  digitalWrite(buzzer, HIGH);
  delay(100);
  digitalWrite(buzzer, LOW);
  if (((in % 10) > 0) && ((in / 10) >= 1))
  {
    doNumber( 1, row, 3);                    // static
    doNumber((in % 10), row, 6);
    doNumber(10, row, 9);

  }
  else
  {
    doNumber(in, row, 6);
    doNumber(10, row, 9);

  }
  if (((out % 10) > 0) && ((out / 10) >= 1))
  {
    doNumber( 1, row, 11);
    doNumber((out % 10), row, 14);
    doNumber(10, row, 9);

  }
  else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
    doNumber(out, row, 14);
    doNumber(10, row, 9);

  }
  if (((in % 10) > 0) && ((in / 10) >= 1))
  {
    doNumber( 1, row, 3);                    // static
    doNumber((in % 10), row, 6);
    doNumber(10, row, 9);

  }
  else
  {
    doNumber(in, row, 6);
    doNumber(10, row, 9);

  }
  if (((out % 10) > 0) && ((out / 10) >= 1))
  {
    doNumber( 1, row, 11);
    doNumber((out % 10), row, 14);
    doNumber(10, row, 9);

  }
  else
  {
    doNumber(out, row, 14);
    doNumber(10, row, 9);

  }

  lcd.clear();
  lcd.setCursor(7, 1);
  lcd.print(F("REPEAT"));
  lcd.setCursor(3, 2);
  lcd.print(F("WAIT A MINUTE"));
  delay(1000);
  lcd.clear();
}
if ((in == out))
{
  if (in >= 1)

  {
    if (((in % 10) > 0) && ((in / 10) >= 1))
    {
      doNumber( 1, row, 3);                  // static
      doNumber((in % 10), row, 6);
      doNumber(10, row, 9);

    }
    else
    {
      doNumber(in, row, 6);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      doNumber(10, row, 9);

    }
    if (((out % 10) > 0) && ((out / 10) >= 1))
    {
      doNumber( 1, row, 11);
      doNumber((out % 10), row, 14);
      doNumber(10, row, 9);

    }
    else
    {
      doNumber(out, row, 14);
      doNumber(10, row, 9);

    }
    digitalWrite(LED2, HIGH);
    digitalWrite(LED1, LOW);
    delay(5000);
    EEPROM.write(4092, 0);
    EEPROM.write(4093, 0);
    EEPROM.write(4094, 0);
    EEPROM.write(4095, 0);

    in = EEPROM.read(4094);
    out = EEPROM.read(4095);

  }
 }
}
boolean findID( byte find[] ) {
 int count = EEPROM.read(0);      // Read the first Byte of EEPROM that
 for ( int i = 0 ; i < count; i++ ) {    // Loop once for each EEPROM entry
  readID(i);          // Read an ID from EEPROM, it is stored in storedCard[4]
  if ( checkTwo( find, storedCard ) ) {  // Check to see if the storedCard read from EEPROM
   return true;
   break;  // Stop looking we found it
  }
  else {    // If not, return false
  }
 }
 return false;
}
void readID( int number ) {
 int start = (number * 12 ) + 1;     // Figure out starting position
 for ( int i = 0; i < 4; i++ ) {     // Loop 4 times to get the 4 Bytes
  storedCard[i] = EEPROM.read(start + i);   // Assign values read from EEPROM to array
 }
}
int findIDSLOT( byte find[] ) {
 int count = EEPROM.read(0);       // Read the first Byte of EEPROM that
 for ( int i = 0; i < count; i++ ) {    // Loop once for each EEPROM entry
  readID(i);                // Read an ID from EEPROM, it is stored in storedCard[4]
  if ( checkTwo( find, storedCard ) ) {  // Check to see if the storedCard read from EEPROM
   // is the same as the find[] ID card passed
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      return i;         // The slot number of the card
      break;          // Stop looking we found it
    }
   }
}

boolean checkTwo ( byte a[], byte b[] ) {
  if ( a[0] != 0 )       // Make sure there is something in the array first
    match = true;       // Assume they match at first
  for ( int k = 0; k < 4; k++ ) {   // Loop 4 times
    if ( a[k] != b[k] )     // IF a != b then set match = false, one fails, all fail
      match = false;
  }
  if ( match ) {      // Check to see if if match is still true
    return true;      // Return true
  }
  else  {
    return false;       // Return false
  }
}



void checkuser()
{
  in = EEPROM.read(4094);
  out = EEPROM.read(4095);
  if ((in == out) && (in != 0))

  {
    if (in >= 1)

    {
      if (((in % 10) > 0) && ((in / 10) >= 1))
      {
        doNumber( 1, row, 3);                    // static
        doNumber((in % 10), row, 6);
        doNumber(10, row, 9);

      }
      else
      {
        doNumber(in, row, 6);
        doNumber(10, row, 9);

      }
      if (((out % 10) > 0) && ((out / 10) >= 1))
      {
        doNumber( 1, row, 11);
        doNumber((out % 10), row, 14);
        doNumber(10, row, 9);

      }
      else
      {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      doNumber(out, row, 14);
      doNumber(10, row, 9);

    }
    delay(5000);
    EEPROM.write(4094, 0);
    EEPROM.write(4095, 0);

    in = EEPROM.read(4094);
    out = EEPROM.read(4095);
    if (((in % 10) > 0) && ((in / 10) >= 1))
    {
      doNumber( 1, row, 3);                    // static
      doNumber((in % 10), row, 6);
      doNumber(10, row, 9);
    }
    else
    {
      doNumber(in, row, 6);
      doNumber(10, row, 9);
    }
    if (((out % 10) > 0) && ((out / 10) >= 1))
    {
      doNumber( 1, row, 11);
      doNumber((out % 10), row, 14);
      doNumber(10, row, 9);
    }
    else
    {
      doNumber(out, row, 14);
      doNumber(10, row, 9);
    }


  }
 }
 else if ((in == out) && (in == 0))
 {
   Serial.println("In = out");
   Serial.println("Check User");
    DateTime now = rtc.now();
    yy = (now.year() % 100);
    mm = (now.month());
    dd = (now.day());
    hh = (now.hour());
    mi = (now.minute());
    ss = (now.second());
    int yy2 = EEPROM.read(4086);
    int mm2 = EEPROM.read(4087);
    int dd2 = EEPROM.read(4088);
    int hh2 = EEPROM.read(4089);
    int mi2 = EEPROM.read(4090);
    int ss2 = EEPROM.read(4091);
   int
total=(yy*365*24*60*60)+(mm*30*24*60*60)+(dd*24*60*60)+(hh*60*60)+(mi*60)+ss;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    int
total1=(yy2*365*24*60*60)+(mm2*30*24*60*60)+(dd2*24*60*60)+(hh2*60*60)+(mi2*60
)+ss2;
    int pir = digitalRead(digitalin1);
    if (interrupt)
    {
     Serial.println("Hang Out");
     lcd.clear();
     lcd.setCursor(5, 1);
     lcd.print(F("Hang Up"));
     call.HangUp();
     num = EEPROM.read(0);
     for (int i = 0; i < num; i++)
     {
       EEPROM.write((i * 12) + 11, 0);
       EEPROM.write((i * 12) + 12, 0);
     }
     EEPROM.write(4092, 0);
     EEPROM.write(4093, 0);
     EEPROM.write(4094, 0);
     EEPROM.write(4095, 0);
     digitalWrite(relay2, LOW);
     in = EEPROM.read(4094);
     out = EEPROM.read(4095);
     interrupt = false;
     mirror = true;
     delay(2000);
     lcd.clear();
     lcd.setCursor(4, 0);
     lcd.print(F("USER RECORDS"));
     lcd.setCursor(0, 2);
     lcd.print(F("IN"));
     lcd.setCursor(17, 2);
     lcd.print(F("OUT"));
     if (((in % 10) > 0) && ((in / 10) >= 1))
     {
       doNumber( 1, row, 3);                    // static
       doNumber((in % 10), row, 6);
       doNumber(10, row, 9);
     }
     else
     {
       doNumber(in, row, 6);
       doNumber(10, row, 9);
     }
     if (((out % 10) > 0) && ((out / 10) >= 1))
     {
       doNumber( 1, row, 11);
       doNumber((out % 10), row, 14);
       doNumber(10, row, 9);
     }
     else
     {
       doNumber(out, row, 14);
       doNumber(10, row, 9);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    EEPROM.write(4086, 0);
    EEPROM.write(4087, 0);
    EEPROM.write(4088, 0);
    EEPROM.write(4089, 0);
    EEPROM.write(4090, 0);
    EEPROM.write(4091, 0);
   checkpir = false;
   }
   else if(checkpir)
   {
   Serial.println(pir);
   Serial.println(total-total1);
   if ((pir == 1)&&((total-total1)>30))
   {
    Serial.println("Start pir");
    Serial.println("child in car");
    if (mirror)
    {
     Serial.println("Open window");
     digitalWrite(relay2, HIGH);
     delay(4000);
     digitalWrite(relay2, LOW);
     mirror = false;
    }
    Serial.println("Call");
    call.Call("0824875897");
    lcd.clear();
    lcd.setCursor(8, 1);
    lcd.print(F("CALL"));

   }
   }
   Serial.println("no one in car");
}
else if (in != out)
{

  if (interrupt)
  {
    Serial.println("Clear LCD");
    digitalWrite(relay1, LOW);
    num = EEPROM.read(0);
    int start = ( num * 12 ) + 1;
    interrupt = false;
    lcd.clear();
    lcd.setCursor(5, 1);
    lcd.print(F("Clear User"));
    for ( int j = 0; j < start; j++ )
    { // Loop 4 times
      EEPROM.write( j, 0 );  // Write the array values to EEPROM in the right position

    }
    EEPROM.write(0, 0);
    EEPROM.write(4094, 0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      EEPROM.write(4095, 0);
      in = EEPROM.read(4094);
      out = EEPROM.read(4095);
      delay(1000);
      lcd.clear();
      lcd.setCursor(4, 0);
      lcd.print(F("USER RECORDS"));
      lcd.setCursor(0, 2);
      lcd.print(F("IN"));
      lcd.setCursor(17, 2);
      lcd.print(F("OUT"));
      if (((in % 10) > 0) && ((in / 10) >= 1))
      {
        doNumber( 1, row, 3);                    // static
        doNumber((in % 10), row, 6);
        doNumber(10, row, 9);
      }
      else
      {
        doNumber(in, row, 6);
        doNumber(10, row, 9);
      }
      if (((out % 10) > 0) && ((out / 10) >= 1))
      {
        doNumber( 1, row, 11);
        doNumber((out % 10), row, 14);
        doNumber(10, row, 9);
      }
      else
      {
        doNumber(out, row, 14);
        doNumber(10, row, 9);
      }
      interrupt = false;
    } else
    {
      Serial.println("In != out");
      Serial.println("Horn");
      digitalWrite(relay1, HIGH);

    }
  }
}

void gettime(String p) {
  time = "Time " + Now();
  time += " ";
  time += p ;
  Serial.println(time);
  WriteText(time);

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void WriteText(String txt) {
  myFile = SD.open("RECORD.txt", FILE_WRITE);
  if (myFile) {
    myFile.println(txt);
    myFile.close();
  }
  else {
    // if the file didn't open, print an error:
    Serial.println("error opening test.txt");
  }
}


String Now() {
  String time = "";
  String file;
  String nhum = file ;
  if (RTC.read(tm)) {
    //    time = String(tm.Hour+":"+tm.Minute+":"+tm.Secnd+" DAY :
"+tm.Day+"/"+tm.Month+"/"+tmYearToCalendar(tm.Year));
    time += tm.Hour;
    time += ":";

    time += tm.Minute;
    time += ":";

    time += tm.Second;
    time += " DAY : ";

    time += tm.Day;
    time += "/";

    time += tm.Month;
    time += "/";

    time += tmYearToCalendar(tm.Year);



  }
  else {
    time = "NO";
    if (RTC.chipPresent()) {
      Serial.println("The DS1307 is stopped.  Please run the SetTime");

    }
    else {
      Serial.println("DS1307 read error!  Please check the circuitry.");

    }
  }
  return time;
}
void Inter()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  interrupt = true;
}



////////////////////////////////////// Main Loop ////////////////////////////////
void loop () {
  in = EEPROM.read(4094);
  out = EEPROM.read(4095);
  lcd.clear();
  lcd.setCursor(4, 0);
  lcd.print(F("USER RECORDS"));
  lcd.setCursor(0, 2);
  lcd.print(F("IN"));
  lcd.setCursor(17, 2);
  lcd.print(F("OUT"));

  DateTime now = rtc.now();

  if (((in % 10) > 0) && ((in / 10) >= 1))
  {
    doNumber( 1, row, 3);                    // static
    doNumber((in % 10), row, 6);
    doNumber(10, row, 9);
  }
  else
  {
    doNumber(in, row, 6);
    doNumber(10, row, 9);
  }
  if (((out % 10) > 0) && ((out / 10) >= 1))
  {
    doNumber( 1, row, 11);
    doNumber((out % 10), row, 14);
    doNumber(10, row, 9);
  }
  else
  {
    doNumber(out, row, 14);
    doNumber(10, row, 9);
  }
  Serial.println("Wait Card");
  //Serial.println(now.second(), DEC);

  do {

    if ((in != out)) // Ã¤Â¿ÃiÂ´Â§
    {
      digitalWrite(LED1, HIGH);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    else
    {
      digitalWrite(LED1, LOW);
    }
    if (in == out) //Ã¤Â¿Ã Â¢Ã•Ã‚Ã‡
    {
      digitalWrite(LED2, HIGH);
    }
    else
    {
      digitalWrite(LED2, LOW);
    }
    int val = digitalRead(digitalin);

    if (val == 0)
    {
      int in = EEPROM.read(4094);
      Serial.println("Switch off");

      if(checktime&&(in==0))
      {

        DateTime now = rtc.now();
        yy = (now.year() % 100);
        mm = (now.month());
        dd = (now.day());
        hh = (now.hour());
        mi = (now.minute());
        ss = (now.second());
      EEPROM.write(4086, yy);
      EEPROM.write(4087, mm);
      EEPROM.write(4088, dd);
      EEPROM.write(4089, hh);
      EEPROM.write(4090, mi);
      EEPROM.write(4091, ss);
      checktime = false;
      }

      /*digitalWrite(button,LOW);*/
      checkuser();

    }
    else
    {
      digitalWrite(relay1, LOW);
      interrupt = false;
      successRead = getID();   // sets successRead to 1 when we get read from reader
otherwise 0
    }
  }
  while (!successRead);  // sets successRead to 1 when we get read from reader otherwise 0

  // sets successRead to 1 when we get read from reader otherwise 0
  if (successRead == 1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {

    writeID(readCard);
    successRead = 0;
  }
  checktime = true;
  checkpir = true;

}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้