รหัสโครงการวิจัย SUT7-709-52-24-45

รายงานการวิจัย

เครื่องควบคุมสัญญาณไฟจราจรโดยอัตโนมัติ
(Automatic Traffic Light Controller)

ได้รับทุนอุดหนุนการวิจัยจาก
มหาวิทยาลัยเทคโนโลยีสุรนารี

ผลงานวิจัยเป็นความรับผิดชอบของหัวหน้าโครงการวิจัยแต่เพียงผู้เดียว

รหัสโครงการวิจัย SUT7-709-52-24-45

รายงานการวิจัย

เครื่องควบคุมสัญญาณไฟจราจรโดยอัตโนมัติ
(Automatic Traffic Light Controller)

คณะผู้วิจัย

หัวหน้าโครงการ
ผู้ช่วยศาสตราจารย์ ดร. รังสรรค์ ทองทา
สาขาวิชาวิศวกรรมโทรคมนาคม
สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี

ผู้ร่วมวิจัย
นายปัญญา หันตุลา

ได้รับทุนอุดหนุนการวิจัยจากมหาวิทยาลัยเทคโนโลยีสุรนารี ปีงบประมาณ พ.ศ. 2553-2554
ผลงานวิจัยเป็นความรับผิดชอบของหัวหน้าโครงการวิจัยแต่เพียงผู้เดียว
พฤษภาคม 2555

กิตติกรรมประกาศ

ผู้วิจัยขอขอบคุณ มหาวิทยาลัยเทคโนโลยีสุรนารีในการสนับสนุนงบประมาณวิจัย และขอขอบคุณศูนย์ความเป็นเลิศทางด้านชีวมวล มทส ในการช่วยเหลือออกแบบและติดตั้ง loop detector รูปทรงกลม ซึ่งน่าจะเป็นตัวแรกที่ทำการติดตั้งและใช้งานจริงบนพื้นถนนในประเทศไทย

ผู้วิจัย
พฤษภาคม 2555

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทคัดย่อ

งานวิจัยนี้ได้นำเสนอการออกแบบ สร้าง และทดสอบ เครื่องควบคุมสัญญาณไฟจราจรโดยอัตโนมัติแบบกระตุ้นด้วยยานพาหนะ ซึ่งประกอบด้วยส่วนประกอบที่สำคัญสองส่วนคือ ส่วนตรวจจับยานพาหนะ (Vehicle Detector) และส่วนควบคุมสัญญาณไฟจราจร (Traffic Light Controller) ส่วนตรวจจับยานพาหนะเป็นชนิดลูปดีเทคเตอร์ โดยวงจรกำเนิดความถี่จะสร้างความถี่ประมาณ 30-60 kHz ร่วมกับขดลวดที่ฝังอยู่ภายใต้พื้นถนน เมื่อมียานพาหนะมาอยู่เหนือขดลวด จะทำให้ความถี่ที่มันสร้างขึ้นเปลี่ยนแปลงไป การวัดความถี่ที่เปลี่ยนแปลงนี้กระทำโดยการนับโดยตรงด้วยไมโครโปรเซสเซอร์ การตัดสินใจว่ามียานพาหนะปรากฏอยู่สามารถหาได้จากเปอร์เซ็นต์การเปลี่ยนแปลงความถี่ โดยสามารถกำหนดได้ 3 ระดับคือ 0.5% 1% และ 3% วงจรกำเนิดความถี่ถูกแยกออกจากขดลวดด้วยหม้อแปลงความถี่เพื่อป้องกันการนำไฟฟ้าจากภายนอกเข้าไปวงจรภายใน ส่วนควบคุมสัญญาณไฟจราจรอาศัยข้อมูลจากส่วนตรวจจับมาตัดสินใจเพื่อควบคุมสัญญาณไฟตามที่ผู้ใช้งานกำหนด ส่วนควบคุมนี้มีส่วนป้องกันสัญญาณไฟเขียวทั้งสองด้านของทิศทางที่ขัดแย้งกัน โดยส่วนป้องกันนี้เป็นวงจรที่แยกต่างหากจากวงจรควบคุม ระบบทั้งหมดสามารถทำงานได้ดีภายใต้สภาวะแวดล้อมจริง สามารถตรวจจับยานพาหนะขนาดเล็ก เช่น รถจักยานยนต์ที่ความไว 0.5% ได้

มหาวิทยาลัยเทคโนโลยีสุรนารี

Abstract

This research presents design, build and test an automatic traffic light controller system. The system consists of two parts; vehicle detector and traffic light controller. The vehicle detector is a loop detector type. The oscillator circuit with the loop under the traffic lane creates frequency between 30-60 kHz. When a vehicle presents, the frequency will be changed from the original. To make decision, threshold level in percentage of changing frequency of 0.5%, 1% and 3% are available. The oscillator circuit is separated from the loop detector by high frequency transformer. The traffic light controller uses information from the detector circuit to operate the traffic light accordingly to the users. The controller has separate circuit to prevent both green lights between two conflict directions. The overall systems can operate well under real situations and can detect small vehicle such as a motorcycle at 0.5% sensitivity.

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ

เรื่อง	หน้า
กิตติกรรมประกาศ	ก
บทคัดย่อภาษาไทย	ข
บทคัดย่อภาษาอังกฤษ	ค
สารบัญ	ง
สารบัญตาราง	ช
สารบัญภาพ	ฌ
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มาของปัญหา	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของงานวิจัย	2
1.4 ขั้นตอนการวิจัย	2
1.5 ประโยชน์ที่คิดว่าจะได้รับ	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 ความรู้ทั่วไปเกี่ยวกับสัญญาณไฟจราจร	3
2.2 คุณลักษณะของผู้ใช้รถใช้ถนน	6
2.2.1 ระยะเวลาการรับรู้และตอบสนอง (Perception and reaction time)	6
2.2.2 การมองเห็นและการขับขี่	7
2.2.3 คุณลักษณะอื่นๆ ของผู้ขับขี่	9
2.3 คุณลักษณะของยานพาหนะ	9
2.3.1 รูปร่างยานพาหนะมาตรฐาน (Design Vehicles)	9
2.3.2 อัตราการเร่งของยานพาหนะ	10
2.3.3 การชะลอตัวของยานพาหนะ	11
2.3.4 คุณลักษณะในการขับเคลื่อน	12
2.3.5 ตัวอย่าการคำนวณระยะทางการรับรู้และตอบสนอง และระยะทางการชะลอความเร็ว	13
2.4 ทฤษฎีการวิเคราะห์การออกแบบสัญญาณไฟจราจร	14
2.4.1 รอบเวลาสัญญาณไฟ (Signal Cycle)	14
2.4.2 เวลาล่าช้าของการเริ่มต้นและสุดท้าย (Start and End Lag Time)	14
2.4.3 การสูญเปล่าของการเคลื่อนที่ (Movement Lost Time)	15
2.4.4 การเคลื่อนที่วิกฤต (Critical Movement)	15
2.4.5 เวลาสูญเปล่าของทางแยก (Intersection Lost Time)	16
2.5 ทฤษฎีการวิเคราะห์ประสิทธิภาพการทำงานของทางแยกที่ควบคุมด้วยสัญญาณไฟ	16
2.5.1 การพิจารณาการเคลื่อนที่ (Movement Consideration)	16

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ(ต่อ)

เรื่อง		หน้า
	2.5.2 ความล่าช้า (Delay)	17
2.6	ระบบควบคุมสัญญาณไฟจราจร	17
2.7	องค์ประกอบของเครื่องควบคุมไฟจราจร	20
2.8	ระบบตัวตรวจวัด (Detectors)	22
2.9	วิวัฒนาการของระบบตรวจวัดยานพาหนะ	25
	2.9.1 อุปกรณ์ตรวจวัดการจราจรที่ติดตั้งอยู่บนหรือใต้ผิวทาง (In-roadway sensors)	26
	2.9.2 อุปกรณ์ตรวจวัดการจราจรทีติดตั้งอยู่เหนือผิวทาง (Over-roadway sensors)	30
	2.9.3 เทคโนโลยีการตรวจวัดการจราจรที่ใช้ในประเทศไทย	37
2.10	ไมโครคอนโทรเลอร์ ARM7	38
	2.10.1 ไมโครคอนโทรลเลอร์ ARM7 Philips LCP2148	38
	2.10.2 ฮาร์ดแวร์และซอฟต์แวร์ของไมโคคอนโทรเลอร์	43
	2.10.3 การกำหนดค่าเริ่มต้นให้กับไมโครคอนโทรลเลอร์ ARM7	45
	2.10.4 การต่อ GPIO เป็นเอาต์พุต	52
	2.10.5 อินเตอร์รัปต์ไมโครคอนโทรลเลอร์ ARM7	58
	2.10.6 การกำหนดค่าเริ่มต้นสำหรับ UART0	66
บทที่ 3	การออกแบบระบบควบคุมสัญญาณไฟจราจร	72
3.1	ส่วนของการออกแบบฮาร์ดแวร์ (Hardware) ของส่วนควบคุมสัญญาณไฟจราจร (Controller)	73
	3.1.1 บอร์ดไมโครคอนโทลเลอร์ ARM7 LPC2148	75
	3.1.2 วงจรเชื่อมต่อตัวตรวจจับยานพาหนะ (Detector Socket)	76
	3.1.3 วงจรแสดงผล (Display)	78
	3.1.4 วงจรป้องกัน (Protection Circuit)	79
	3.1.5 วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร (Driver 220 VAC)	84
	3.1.6 ออกแบบวงจรรวมของส่วนควบคุมสัญญาณไฟจราจร	86
3.2	ส่วนของการออกแบบซอร์ฟแวร์ (Software) ของส่วนควบคุมสัญญาณไฟจราจร (Controller)	88
3.3	ส่วนของการออกแบบฮาร์ดแวร์ตัวตรวจจับยานพาหนะ (Detector)	100
	3.3.1 ตัวกำเนิดสัญญาณ (Oscillator)	100
	3.3.2 ลูปเหนี่ยวนำ (Loop Detector)	103
	3.3.2 ไมโครคอนโทลเลอร์ ARM7 LPC2148	105
	3.3.3 ออกแบบวงจรรวมของตัวตรวจจับยานพานะ	107

สารบัญ(ต่อ)

เรื่อง	หน้า
3.4 ส่วนของการออกแบบซอร์ฟแวร์ตัวตรวจจับยานพาหนะ (Detector)	108
บทที่ 4 การใช้งานและผลการทดสอบ	114
4.1 การทดสอบความถี่เรโซแนนซ์ของหม้อแปลงที่เชื่อมต่อกับลูปเหนี่ยวนำ	117
4.2 การทดสอบวงจรกำเนิดความถี่ในห้องปฏิบัติการ	119
4.3 การทดสอบโปรแกรมนับความถี่สำหรับตรวจวัดยานพาหนะกับเครื่องมือวัดในห้องปฏิบัติการ	120
4.4 การใช้งานตัวตรวจวัดยานพาหนะ (Detector)	123
4.5 การทดสอบการตรวจวัดยานพาหนะ	126
4.6 การใช้งานตัวควบคุมสัญญาณไฟจราจร (Controller)	137
4.7 การทดสอบตัวควบคุมสัญญาณไฟจราจร (Controller)	137
บทที่ 5 สรุปผลการทดลอบและข้อเสนอแนะ	138
บรรณนุกรม	140
ภาคผนวก	141
ประวัติผู้วิจัย	168

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญตาราง

ตาราง **หน้า**

ตารางที่ 2.1 ขนาดของยานพาหนะมาตรฐานที่ใช้สำหรับออกแบบ โดย AASHTO 10
ตารางที่ 2.2 อัตราการเร่งยานพาหนะแต่ละประเภท 10
ตารางที่ 2.3 ค่าสัมประสิทธิ์ความต้านทานการลื่นไถลที่ใช้ในการออกแบบ 12
ตารางที่ 2.4 แสดงข้อดี-ข้อเสียของระบบ VA และ ATC 18
ตารางที่ 2.5 PLL Register 47
ตารางที่ 2.6 PLLCFG Register 48
ตารางที่ 2.7 PLLCON Register 48
ตารางที่ 2.8 ค่าแต่ละบิตของรีจิสเตอร์ PLLSTAT 49
ตารางที่ 2.9 โหมดการทำงานของวงจรPLL 49
ตารางที่ 2.10 ค่าแต่ละบิตของรีจิสเตอร์ VPBDIV 50
ตารางที่ 2.11 MAM Register 50
ตารางที่ 2.12 การกำหนดค่าให้กับ MAMCR Register 51
ตารางที่ 2.13 การกำหนดค่าให้กับ MAMTIM Register 51
ตารางที่ 2.14 รีจิสเตอร์ PINSEL0, PINSEL1, PINSEL2 52
ตารางที่ 2.15 ค่าประจำบิตของรีจิสเตอร์ PINSEL0 52
ตารางที่ 2.16 ค่าประจำบิตของรีจิสเตอร์ PINSEL1 54
ตารางที่ 2.17 ค่าประจำบิตของรีจิสเตอร์ PINSEL2 55
ตารางที่ 2.18 ชื่อและแอดเดรสของรีจิสเตอร์ที่ใช้ในการควบคุม GPIO 55
ตารางที่ 2.19 แหล่งกำเนิดอินเตอร์รัปต์ทั้ง 32 ตัว 58
ตารางที่ 2.20 รีจิสเตอร์ที่เกี่ยวข้องกับการอินเตอร์รัปต์ 60
ตารางที่ 2.21 รีจิสเตอร์ที่เกี่ยวข้องกับการอินเตอร์รัปต์จากภายนอก 63
ตารางที่ 2.22 ค่าประจำบิตของรีจิสเตอร์ EXTINT 63
ตารางที่ 2.23 ค่าประจำบิตของรีจิสเตอร์ EXTMODE 64
ตารางที่ 2.24 ค่าประจำบิตของรีจิสเตอร์ EXTPOLAR 65
ตารางที่ 2.25 ค่าประจำบิตของรีจิสเตอร์ EXTPILAR 66
ตารางที่ 2.26 รีจิสเตอร์ที่เกี่ยวข้องกับ UART0 67
ตารางที่ 2.27 ค่าประจำบิตของรีจิสเตอร์ UART0 Line Control Register (U0LCR) 68
ตารางที่ 2.28 แสดงค่าประจำบิตของ UART0 Line Status Register (U0LSR) 69
ตารางที่ 3.1 LED แสดงสถานะการทำงานและการเชื่อมต่อพอร์ตของไมโครคอนโทลเลอร์ 78
ตารางที่ 3.2 แสดงสถานะการเชื่อมต่อพอร์ตของไมโครคอนโทลเลอร์ 79
ตารางที่ 3.3 ตารางแสดงการทำงานของการ RS Flip flop 83
ตารางที่ 3.4 ผลการทดสอบเมื่อทำการปรับสวิตช์ 101
ตารางที่ 3.5 การตั้งค่าอัตราการเปลี่ยนแปลงความถี่ (Sensitivity) 113
ตารางที่ 4.1 ความถี่ของตัวตรวจวัดยานพาหนะ 123

สารบัญตาราง(ต่อ)

ตาราง		หน้า
ตารางที่ 4.2	ความไวสำหรับตรวจวัดยานพาหนะ	124
ตารางที่ 4.3	ตารางแสดงผลการตรวจวัดประเภทรถยนต์ส่วนบุคคล	136
ตารางที่ 4.4	ตารางแสดงผลการตรวจวัดประเภทรถจักรยานยนต์	136
ตารางที่ 4.5	สรุปผลการตรวจวัดยานพาหนะ	136

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญภาพ

รูปภาพ **หน้า**

รูปที่ 2.1 ขอบเขตการมองเห็น 10
รูปที่ 2.2 แสดงเวลาล่าช้าของการเริ่มต้นและสุดท้าย (Start and End Lag Time) 15
รูปที่ 2.3 การสูญเปล่าของการเคลื่อนที่ (Movement Lost Time) 16
รูปที่ 2.4 ลักษณะพื้นฐานของเครื่องควบคุมสัญญาณไฟจราจร 21
รูปที่ 2.5 ลักษณะพื้นฐานของส่วนประมวลผลกับตัวตรวจจับยานพาหนะ 22
รูปที่ 2.6 แสดงลำดับความสูงที่เหมาะสมในการตรวจวัด 23
รูปที่ 2.7 แสดงผลการกระทำระหว่างยานพาหนะกับขดลวดเหนี่ยวนำ 24
รูปที่ 2.8 ลักษณะวงจรพื้นฐานของส่วนเครื่องตรวจจับยานพาหนะแบบลูปเหนี่ยวนำ 24
รูปที่ 2.9 การตรวจวัดยานพาหนะโดยใช้ Micro Phone 25
รูปที่ 2.10 การตรวจวัดยานพาหนะโดยการกดทับของแผ่นเหล็ก 25
รูปที่ 2.11 การตรวจวัดยานพาหนะโดยอาศัยเสียง 26
รูปที่ 2.12 การตรวจวัดยานพาหนะโดยใช้ท่อลม 27
รูปที่ 2.13 การตรวจวัดยานพาหนะโดยใช้ Inductive loop 27
รูปที่ 2.14 Inductive loop บนถนน 28
รูปที่ 2.15 การติดตั้ง Magnetic Sensors 29
รูปที่ 2.16 การตรวจวัดยานพาหนะโดยใช้ Video Image Processors 30
รูปที่ 2.17 คลื่นแม่เหล็กไฟฟ้าที่ความถี่ต่างๆ 31
รูปที่ 2.18 การตรวจวัดยานพาหนะโดย Microwave radar sensors 32
รูปที่ 2.19 การตรวจวัดยานพาหนะโดยใช้ Active infrared sensors 33
รูปที่ 2.20 การตรวจวัดการจราจรแบบ Microwave Radar Sensors 33
รูปที่ 2.21 เครื่องตัววัดจราจร แบบ Active Infrared Sensors 33
รูปที่ 2.22 การตรวจวัดยานพาหนะโดยใช้ Passive infrared sensors 34
รูปที่ 2.23 การตรวจวัดยานพาหนะโดยใช้ Ultrasonic sensors 36
รูปที่ 2.24 การตรวจวัดยานพาหนะโดยใช้ Passive acoustic sensors 37
รูปที่ 2.25 Block Diagram LPC2148 39
รูปที่ 2.26 memory map LPC 2148 41
รูปที่ 2.27 LPC2148 pinning 42
รูปที่ 2.28 ลักษณะโครงสร้างของบอร์ด CP-JR ARM7 USB-LPC2148 / EXP 44
รูปที่ 2.29 ผังวงจร PLL ภายในไมโครคอนโทรลเลอร์ ARM 7 46
รูปที่ 2.30 Block Diagram วงจรกำเนิดสัญญาณภายใน ARM7 30
รูปที่ 3.1 แสดงโครงสร้างระบบควบคุมสัญญาณไฟจราจร 72
รูปที่ 3.2 วงจรสำหรับควบคุมสัญญาณไฟจราจร 74
รูปที่ 3.3 วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร (Driver 220 VAC) 74
รูปที่ 3.4 ไมโครคอนโทลเลอร์ ARM7 LPC2148 75
รูปที่ 3.5 การออกแบบไมโครคอนโทลเลอร์ ARM7 LPC2148 76

สารบัญภาพ(ต่อ)

รูปภาพ		หน้า
รูปที่ 3.6	การออกแบบวงจรแปลงระดับสัญญาณ	76
รูปที่ 3.7	วงจรเชื่อมต่อตัวตรวจจับยานพาหนะ (Detector Socket)	77
รูปที่ 3.8	การเชื่อมต่อระหว่างไมโครคอนโทลเลอร์ ARM7 LPC2148 และตัวตรวจจับยานพาหนะ	78
รูปที่ 3.9	วงจรแสดงผล (Display)	79
รูปที่ 3.10	วงจรป้องกัน (Protection Circuit)	80
รูปที่ 3.11	ออกแบบวงจรสำหรับตรวจสอบสัญญาณไฟจราจร	80
รูปที่ 3.12	วงจรตรวจสอบสัญญาณไฟจราจร	81
รูปที่ 3.13	วงจรอินพุตของวงจรป้องกันการเกิดสัญญาณไฟเขียว 2 ทิศทาง	82
รูปที่ 3.14	การออกแบบวงจรดิจิตอล	82
รูปที่ 3.15	วงจรควบคุมการเปิดและปิดแหล่งจ่ายไฟ	84
รูปที่ 3.16	วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร	84
รูปที่ 3.17	ลักษณะการวางตำแหน่งของอุปกรณ์ของบอร์ดควบคุมสัญญาณไฟจราจร	86
รูปที่ 3.18	ลักษณะการวางตำแหน่งของอุปกรณ์วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร	87
รูปที่ 3.19	ลักษณะแยกการจราจรที่ติดตั้งระบบควบคุมสัญญาณไฟจราจร	88
รูปที่ 3.20	รูปแบบการทำงานของฟังก์ชันการทำงานหลัก (Main Function)	89
รูปที่ 3.21	การทำงานของ Main Function	90
รูปที่ 3.22	การทำงานของ Set Time Wait Function	91
รูปที่ 3.23	การทำงานของ Check Loop Present Function	92
รูปที่ 3.24	การทำงานของ Present Loop Function	94
รูปที่ 3.25	ฟังก์ชันตรวจสอบตัวตรวจจับยานพาหนะ (Check Detector Function)	95
รูปที่ 3.26	การทำงานของ Absent Loop Function	96
รูปที่ 3.27	การทำงานของ Change Cross to Main Function	97
รูปที่ 3.28	การทำงานของตัวตรวจจับยานพาหนะ (Detector)	99
รูปที่ 3.29	ตัวตรวจจับยานพาหนะ (Detector)	100
รูปที่ 3.30	วงจรกำเนิดสัญญาณ	101
รูปที่ 3.31	วงจรกำเนิดสัญญาณ (Oscillator)	102
รูปที่ 3.32	การต่อลูปเหนี่ยวนำกับวงจรกำเนิดความถี่โดยผ่านหม้อแปลงไฟฟ้า	102
รูปที่ 3.33	รูปแบบของขดลวดแบบต่าง ๆ	103
รูปที่ 3.34	การติดตั้งลูปเหนี่ยวนำ	103
รูปที่ 3.35	การติดตั้งลูปเหนี่ยวนำที่พื้นถนน	104
รูปที่ 3.36	การเชื่อมต่อวงจรกำเนิดความถี่กับไมโครคอนโทลเลอร์ ARM7 LPC218	105
รูปที่ 3.37	การเชื่อมต่อไมโครคอนโทลเลอร์ ARM7 LPC218 กับวงจรภายนอก	106
รูปที่ 3.38	ลักษณะการวางตำแหน่งของอุปกรณ์ต่างๆ ของบอร์ดตรวจจับยานพานะ	107

สารบัญภาพ(ต่อ)

รูปภาพ		หน้า
รูปที่ 3.39	การทำงานของตัวตรวจจับยานพาหนะ	108
รูปที่ 3.40	วิธีการนับสัญญาณในช่วงเวลา 1 วินาที	108
รูปที่ 3.41	สัญญาณ Pulse ที่เข้ามา Logic1 และ Logic0	109
รูปที่ 3.42	สัญญาณ Pulse ที่จับได้ในช่วงเวลา 100 ms	109
รูปที่ 3.43	การทำงานของ Main Function ตัวตรวจจับยานพาหนะ	110
รูปที่ 3.44	การทำงานของ Check Present Function	112
รูปที่ 4.1	ระบบควบคุมสัญญาณไฟจราจรแบบกึ่งอัตโนมัติ	114
รูปที่ 4.2	ภาพถ่ายดาวเทียมแยกการจราจรที่ใช้ทดสอบ	115
รูปที่ 4.3	แยกการจราจรที่ใช้ทดสอบ (1)	115
รูปที่ 4.4	แยกการจราจรที่ใช้ทดสอบ (2)	115
รูปที่ 4.5	ตู้ควบคุมสัญญาณไฟจราจรที่ติดตั้งตรงบริเวณแยกการจราจร	116
รูปที่ 4.6	การทดลองหม้อแปลงด้วยความถี่ต่างๆ	117
รูปที่ 4.7	กราฟความถี่เรโซแนนซ์ของหม้อแปลง (1:3)	117
รูปที่ 4.8	กราฟความถี่เรโซแนนซ์ของหม้อแปลง (1:1) 150 รอบ	118
รูปที่ 4.9	กราฟความถี่เรโซแนนซ์ของหม้อแปลง (1:1) 100 รอบ	118
รูปที่ 4.10	วงจรกำเนิดความถี่และหม้อแปลง	119
รูปที่ 4.11	วัดความถี่โดยใช้ Oscilloscope	119
รูปที่ 4.12	อุปกรณ์การทดสอบโปรแกรมนับความถี่สำหรับตรวจวัดยานพาหนะในห้องปฏิบัติการ	120
รูปที่ 4.13	เครื่อง Multi Function Generator สร้างความถี่ที่ 50 kHz	120
รูปที่ 4.14	เครื่อง Universal Counter อ่านความถี่ได้ที่ 50 kHz	121
รูปที่ 4.15	Oscilloscope วัดความถี่ได้ 50 kHz	121
รูปที่ 4.16	โปรแกรมนับความถี่สำหรับตรวจวัดยานพาหนะอ่านความถี่ที่สร้างมาจากเครื่อง Multi Function Generator ที่ 50 kHz	122
รูปที่ 4.17	ลูปเหนี่ยวนำที่ติดตั้งที่พื้นถนน	123
รูปที่ 4.18	Dip Switch สำหรับเลือกความถี่ของตัวตรวจวัดยานพาหนะ	124
รูปที่ 4.19	Dip Switch สำหรับเลือกความไวของตัวตรวจวัดยานพาหนะ	125
รูปที่ 4.20	แสดงตัวอย่างของการแสดงผลความถี่ของตัวตรวจวัดยานพาหนะผ่านทางโปรแกรม Hyper Terminal	125
รูปที่ 4.21	อุปกรณ์สำหรับทดสอบตรวจจับยานพาหนะ	126
รูปที่ 4.22	ผลการทดสอบวัดความถี่ 50 kHz ในขณะที่ไม่มียานพาหนะ	127
รูปที่ 4.23	ผลการทดสอบวัดความถี่ 32 kHz ในขณะที่ไม่มียานพาหนะ	127
รูปที่ 4.24	ผลการทดสอบวัดความถี่ 26 kHz ในขณะที่ไม่มียานพาหนะ	128
รูปที่ 4.25	ลักษณะยานพาหนะขนาดเล็กอยู่บนลูปเหนี่ยวนำ	129

สารบัญภาพ(ต่อ)

รูปภาพ **หน้า**

รูปที่ 4.26 ลักษณะยานพาหนะประเภทรถยนต์ส่วนบุคคลอยู่บนลูปเหนี่ยวนำ 129
รูปที่ 4.27 ความถี่ที่เกิดการเปลี่ยนแปลงเมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (50 kHz) 130
รูปที่ 4.28 อัตราการเปลี่ยนแปลงความถี่เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (50 kHz) 130
รูปที่ 4.29 ความถี่ที่เกิดการเปลี่ยนแปลงเมื่อมีรถจักรยานยนต์อยู่ที่ลูปเหนี่ยวนำ (50 kHz) 131
รูปที่ 4.30 อัตราการเปลี่ยนแปลงความถี่เมื่อมีรถจักรยานยนต์อยู่ที่ลูปเหนี่ยวนำ (50 kHz) 131
รูปที่ 4.31 ความถี่ที่เกิดการเปลี่ยนแปลงเมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (32 kHz) 132
รูปที่ 4.32 อัตราการเปลี่ยนแปลงความถี่เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (32 kHz) 132
รูปที่ 4.33 ความถี่ที่เกิดการเปลี่ยนแปลงเมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (32 kHz) 133
รูปที่ 4.34 อัตราการเปลี่ยนแปลงความถี่เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (32 kHz) 133
รูปที่ 4.35 ความถี่ที่เกิดการเปลี่ยนแปลงเมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (26 kHz) 136
รูปที่ 4.36 อัตราการเปลี่ยนแปลงความถี่เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (26 kHz) 136
รูปที่ 4.37 ความถี่ที่เกิดการเปลี่ยนแปลงเมื่อมีรถจักรยานยนต์อยู่ที่ลูปเหนี่ยวนำ (26 kHz) 135
รูปที่ 4.38 อัตราการเปลี่ยนแปลงความถี่เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (26 kHz) 135

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของปัญหา

ในปัจจุบันความต้องการใช้ยานพาหนะเพื่อการเดินทางเพิ่มมากขึ้น จึงทำให้สภาพการจราจรเพิ่มขึ้นอย่างรวดเร็วก่อให้เกิดปัญหาที่สำคัญสำหรับชมชุนเมือง และนอกจากนี้ทำให้เกิดปัญหาอื่นๆ ตามมาอย่างมากมาย เช่น ความแออัดของการจราจรทำให้เกิดความล้าช้าของการเดินทาง เกิดมลพิษทางอากาศ ปัญหาคุณภาพชีวิตลดลง เป็นต้น การควบคุมสัญญาณไฟจราจรเป็นวิธีการแก้ไขปัญหาความหนาแน่นของการจราจรที่เกิดขึ้นได้ดีที่สุด แต่ในปัจจุบันจะเป็นการควบคุมสัญญาณไฟแบบเวลาคงที่ (Fixed-time Control) คือ การควบคุมจราจรที่ถูกกำหนดเวลาของสัญญาณไฟในแต่ละทิศทางเอาไว้แน่นอน โดยจะต้องมีการเก็บข้อมูลมาก่อนเพื่อนำมาพิจารณาในการออกแบบ การควบคุมสัญญาณไฟจราจรแบบดังกล่าวจะสามารถแก้ไขปัญหาการจราจรที่มีความหนาแน่นในทุกทิศทางได้ดี แต่สำหรับสภาพการจราจรที่มีทางเอกและทางโทจะทำให้การควบคุมสัญญาณไฟจราจรมีประสิทธิภาพต่ำเพราะจะต้องให้สัญญาณไฟจราจรตามรอบเวลาที่กำหนดเอาไว้ ดังนั้นในงานวิจัยนี้จึงได้ออกแบบการควบคุมสัญญาณไฟจราจรแบบกระตุ้นด้วยยานพาหนะ (Vehicle Actuated: VA) คือ ยานพาหนะในทางเอกสามารถผ่านได้ตลอดเวลาเมื่อไม่มียานพาหนะมารอที่ทางโท แต่เมื่อมียานพาหนะมารอในทางโทจะเปลี่ยนสัญญาณไฟจราจรให้ยานพาหนะในทางโทสามารถผ่านไปได้ ซึ่งเหมาะสมสำหรับการควบคุมสัญญาณไฟจราจรที่มียานพาหนะในทางโทมีปริมาณการจราจรที่ต่ำ การควบคุมสัญญาณไฟจราจรแบบ VA นี้จะให้ประสิทธิภาพมากกว่าแบบกำหนดเวลาคงที่

การตรวจจับว่ามียานพาหนะอยู่หรือไม่ เป็นหัวใจสำคัญของระบบ VA นี้ วิธีการตรวจจับยานพาหนะที่ใช้กันอย่างแพร่หลายคือการใช้ Loop Detector ซึ่งอาศัยคุณสมบัติการเปลี่ยนค่าความเหนี่ยวนำของ Loop ซึ่งเป็นส่วนหนึ่งของวงจรกำเนิดความถี่ เมื่อค่าความเหนี่ยวนำของ Loop เปลี่ยนไปเนื่องจากยานพาหนะ ค่าความถี่ที่ได้จากวงจรกำเนิดความถี่ก็จะเปลี่ยนไปด้วย การตรวจนับความถี่ที่เปลี่ยนไปกระทำได้โดยใช้วงจรที่สามารถทำการคำนวณได้อย่างซับซ้อน เช่น Digital Signal Processing: DSP เพื่อคำนวณหาค่า FFT ของวงจรเพื่อทำการเปรียบเทียบและตัดสินใจว่าขณะนี้มียานพาหนะอยู่หรือไม่

นอกจากการใช้ Loop Detector แล้ว ปัจจุบันยังมีการนำเอาเทคนิค Digital Image Processing มาใช้ทำการประมวลผลภาพที่ได้จากกล้องที่จับอยู่บนถนน เพื่อตัดสินใจว่าในขณะนี้มียานพาหนะอยู่หรือไม่ หรือแม้กระทั่งใช้ Fiber Optic Sensors วางบนถนนเพื่อตรวจจับรถที่วิ่งผ่าน อย่างไรก็ตาม เมื่อพิจารณาถึงความทนทานและราคาของตัวตรวจจับแล้วจะพบว่า Loop Detectors ยังมีความได้เปรียบระบบอื่นอยู่มาก

เนื่องจากระบบนี้ต้องทำงานที่เกี่ยวข้องกับความปลอดภัยทั้งชีวิตและทรัพย์สินของคนหมู่มาก ความทนทานและความเชื่อถือได้ของระบบมีความสำคัญที่ต้องนำมาพิจารณาในลำดับต้นๆ ทั้งที่ต้องทนทานต่อสภาพการใช้งานตามปกติ เช่น การที่ตัวตรวจจับต้องถูกยานพาหนะวิ่งทับอยู่เป็นประจำ การทนทานต่ออุณหภูมิที่สูงกว่าปกติเมื่อต้องทำงานอยู่กลางแดด เป็นต้น และต้องทนต่อสภาพการใช้งานที่ไม่ปกติ เช่น

สภาพแรงดันไฟฟ้าที่สูงหรือต่ำกว่าปกติ ซึ่งบ่อยครั้งจะพบว่าผู้ออกแบบจะคำนึงถึงแต่สภาพการณ์ที่ปกติเท่านั้น ซึ่งเมื่อมีเหตุการณ์ที่ไม่ปกติเกิดขึ้นปรากฏว่าระบบทำงานล้มเหลวอย่างสิ้นเชิง หรือแม้กระทั้งอาจก่อให้เกิดความเสียหายอย่างคาดไม่ถึงในลำดับต่อมา

1.2 วัตถุประสงค์

1. พัฒนา (สร้าง) เครื่องต้นแบบระบบควบคุมสัญญาณไฟจราจรแบบกระตุ้นด้วยยานพาหนะ โดยใช้อุปกรณ์ประมวลสัญญาณซึ่งมีความไว และเสถียรภาพสูง สามารถทนต่อสภาพแวดล้อมได้ดี
2. พัฒนา (สร้าง) ส่วนตรวจจับ (Detector) ยานพาหนะประเภท Loop Detector เพื่อใช้ร่วมกับระบบควบคุมที่ได้พัฒนาขึ้น โดยส่วนตรวจจับนี้ต้องมีความไวเพียงพอที่จะใช้สำหรับตรวจจับยานพาหนะประเภทรถจักยานยนต์ได้

1.3 ขอบเขตของงานวิจัย

- ตัวตรวจจับที่ใช้จะเป็น Loop Detector ทั้งนี้เนื่องจากเป็นที่นิยม มีความทนทาน และราคาถูก
- ส่วนควบคุมต้องมีการป้องกันไม่ให้เกิดเหตุการณ์ที่เป็นอันตราย เช่น ให้ไฟเขียวทุกด้าน
- มีการป้องกันจากอันตรายที่อาจจะเกิดขึ้นจากฟ้าฝ่า เช่น การแยก Loop Detector ของจากวงจร
- มีความสามารถพื้นฐาน เช่นสามารถเปลี่ยนความถี่ได้ในช่วง 50 kHz เพื่อป้องกันการรบกวนกันหรือตั้งระดับความไวในการตรวจจับได้

1.4 ขั้นตอนการวิจัย

- สืบค้นข้อมูลเกี่ยวกับงานวิจัยที่เกี่ยวข้อง
- ศึกษาข้อมูลทางเทคนิคของอุปกรณ์ที่จะนำมาใช้
- พัฒนาส่วนควบคุมที่มีความสามารถตามต้องการ และมีความปลอดภัยภายใต้เงื่อนไขที่ต้องการ
- พัฒนาส่วนตรวจจับที่มีความไว และมีความปลอดภัยภายใต้เงื่อนไขที่ต้องการ
- ติดตั้งระบบควบคุมสัญญาณไฟจราจรที่ออกแบบเพื่อควบคุมสัญญาณไฟจราจร
- ติดตั้งขดลวดเหนี่ยวนำใต้พื้นถนนตรงจุดบริเวณทางโท โดยใช้ตัวตรวจจับยานพาหนะที่ออกแบบเป็นตัวตรวจสอบยานพาหนะ
- ทำการทดสอบวัดค่าความถี่ที่เปลี่ยนไปกับยานพาหนะแบบต่างๆ
- วิเคราะห์และสรุปผลการทดสอบ

1.5 ประโยชน์ที่คิดว่าจะได้รับ

1. ได้ระบบควบคุมสัญญาณไฟแบบกึ่งอัตโนมัติที่ให้ประสิทธิภาพการทำงานที่ดีกว่าระบบเดิม
2. ได้ตัวตรวจจับยานพาหนะที่สามารถทำงานร่วมกับระบบควบคุมสัญญาณไฟจราจร
3. ได้ตัวตรวจจับยานพาหนะที่สามารถตรวจจับยานพาหนะขนาดเล็กได้
4. เพิ่มประสิทธิภาพการควบคุมสัญญาณไฟจราจร โดยสามารถตรวจจับปริมาณการจราจรได้
5. ลดการพึ่งพาเทคโนโลยีจากต่างประเทศ

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ความรู้ทั่วไปเกี่ยวกับสัญญาณไฟจราจร

สัญญาณไฟจราจร (Traffic Signalization) หมายถึง สัญญาณไฟสำหรับให้ผู้ใช้ทางปฏิบัติตามสัญญาณนั้น อาจเป็นสัญญาณไฟทางแยก หรือสัญญาณไฟทางข้ามหรือสัญญาณไฟกระพริบ หรือสัญญาณไฟควบคุมช่องเดินรถ หรือสัญญาณไฟเตือนอื่นๆ (ตามข้อกำหนดกรมตำรวจ เรื่องสัญญาณจราจร เครื่องหมายจราจร และความหมายของสัญญาณจราจรและเครื่องหมายจราจร ข้อ 3 เมื่อวันที่ 24 เมษายน 2522 ออกตาม พ.ร.บ.จราจรทางบกพ.ศ.2522 “**สัญญาณไฟจราจร** หมายถึง โคมสัญญาณไฟที่ใช้ควบคุมการจราจรมีขนาดและติดตั้งหรือทำให้ปรากฏไว้ในทาง ในลักษณะที่ทำให้ผู้ขับขี่หรือผู้ที่ต้องปฏิบัติตามสัญญาณมองเห็นได้โดยชัดเจน การติดตั้งโคมสัญญาณไฟจราจรต้องประกอบด้วยดวงโคมอย่างน้อยสามดวง โดยมีโคมสัญญาณไฟจราจรสีแดงอยู่ตอนบนหรือด้านขวาของผู้ขับขี่หรือผู้ต้องปฏิบัติตามสัญญาณโคมสัญญาณไฟจราจรสีเหลืองอำพันอยู่ตอนกลางและโคมสัญญาณไฟจราจรสีเขียวอยู่ตอนล่างหรือด้านซ้ายมือของผู้ขับขี่ หรือผู้ที่ต้องปฏิบัติตามสัญญาณ ในบางกรณีอาจมีโคมสัญญาณไฟจราจรลูกศรเขียวประกอบได้”)

สัญญาณไฟจราจร

สัญญาณไฟจราจรประกอบด้วย 3 สี คือ

- สีเหลืองอำพัน (Amber) หมายถึง การเตือนให้ผู้ขับขี่ชะลอความเร็วและเพิ่มความระมัดระวังให้มากยิ่งขึ้นในบางครั้งให้หยุดเมื่อเห็นว่าจะไม่ปลอดภัย
- สีแดง (Red) หมายถึง การบังคับให้ยานพาหนะทุกคันหยุด
- สีเขียว (Green) หมายถึง การอนุญาตให้ยานพาหนะแล่นผ่านไปได้

จุดประสงค์ในการติดตั้งสัญญาณไฟจราจร

การติดตั้งสัญญาณไฟจราจรควบคุมที่ทางแยกจะช่วยลดปัญหาการจราจรติดขัดและทำให้การใช้รถใช้ถนนให้ปลอดภัยมากยิ่งขึ้น และเป็นแนวทางที่สำคัญอันหนึ่งที่ใช้ควบคุมการใช้รถใช้ถนน ในโครงข่าย จุดประสงค์หลักในการติดตั้งสัญญาณไฟจราจรอาจจะเป็นข้อหนึ่งข้อใดหรือหลายข้อดังต่อไปนี้

- ลดความขัดแย้งในการจราจร (TRAFFIC CONFLICTS)
- ลดความล่าช้าการจราจร
- เพิ่มความจุถนน
- ลดอุบัติเหตุ
- ลดกำลังเจ้าหน้าที่ตำรวจ
- อำนวยความสะดวกในการข้ามถนนของคนเดินเท้าและรถจักรยาน
- ให้สิทธิพิเศษแก่ยานพาหนะบางประเภทในบางโอกาส เช่น รถโดยสารสาธารณะรถจักรยาน

- ส่งเสริมให้มีการใช้ถนนตรงตามชนิดของถนนที่กำหนดไว้ เช่น ลดจุดขัดแย้งและความล่าช้าบนถนนบางสาย เพื่อดึงดูดให้ผู้ขับขี่หันมาใช้ถนนนั้น เพื่อลดจำนวนการขับขี่ตามซอยต่าง ๆ ที่ผ่านพื้นที่อยู่อาศัย
- ลดความยากลำบาก และความกดดันในการขับขี่ผ่านทางแยกที่มีการจราจรหนาแน่น

ข้อดีของการติดตั้งสัญญาณไฟจราจร

- การจราจรเป็นไปอย่างมีระเบียบ ลดปัญหาการติดขัด
- ลดอุบัติเหตุบางชนิด เช่น การประสานงาน อุบัติเหตุของคนเดินถนน
- ให้ความปลอดภัยแก่รถทางโทในการผ่านหรือเข้าสู่ทางเอก
- เสริมความมั่นใจให้กับผู้ใช้ยานพาหนะ

ข้อเสียของการติดตั้งสัญญาณไฟจราจร

- ทำให้เกิดความล่าช้าบริเวณทางแยก โดยเฉพาะนอกเหนือเวลาเร่งด่วน
- เพิ่มอุบัติเหตุบางประเภท เช่น ชนด้านหลังของรถยนต์
- การติดตั้งสัญญาณไฟในบริเวณไม่เหมาะสมอาจเพิ่มความล่าช้าโดยไม่จำเป็น
- การติดตั้งสัญญาณไฟที่ระยะเวลาไม่เหมาะสมอาจสร้างความล่าช้าและความเบื่อหน่าย

การไหลอิ่มตัว (Saturation Flow)

การไหลอิ่มตัว หมายถึง ปริมาณการจราจรสูงสุดที่ผ่านเส้นหยุดที่ทางแยกไปได้ในช่วงสัญญาณไฟเขียวมีหน่วยนับเป็นปริมาณยานพาหนะต่อชั่วโมงหรือหน่วยยานพาหนะเทียบรถยนต์ส่วนบุคคล เริ่มจากการเคลื่อนที่ การเร่งความเร็วเพื่อให้ความเร็วเพิ่มขึ้นคงที่ ปริมาณการจราจรที่สามารถผ่านทางแยกส่วนหนึ่งจะเสียไปจากประสิทธิภาพของทางแยก

ปัจจัยที่ใช้พิจารณากำหนดให้มีสัญญาณไฟจราจร

1. ความเร็วของการจราจรที่คาดได้ ปริมาณการจราจร และจำนวนคนเดินเท้าในช่วงเวลา 16 ชั่วโมง นอกจากนี้จะต้องพิจารณาถึงสัดส่วนของยานพาหนะในช่วงเวลาเร่งด่วน ความสมดุลระหว่างปริมาณการจราจรในถนนสายหลักกับปริมาณจราจรในถนนสายรอง และจุดขัดแย้ง (CONFLICT) ระหว่างรถเลี้ยวกับคนข้ามถนน
2. สถิติของอุบัติเหตุ (จำนวน อัตรา และคุณลักษณะ) และความสัมพันธ์ของอุบัติเหตุกับรูปแบบของพื้นที่บริเวณนั้น
3. ความเป็นไปได้ในการจัดรูปแบบทางแยกในลักษณะต่างๆ และแผนผังของทางแยก
4. ทางแยกนั้นอยู่ในพื้นที่การควบคุมสัญญาณไฟเป็นโครงข่าย UPC หรือไม่ และอยู่ใกล้กับทางแยกถัดไปมากน้อยเพียงใด

การควบคุมสัญญาณไฟ

การควบคุมสัญญาณไฟ มี 2 ประเภท คือ

การควบคุมสัญญาณไฟแบบเดี่ยว (Isolated Signal Control) ทางแยกที่มีสัญญาณไฟจราจรควบคุมสามารถที่จะทำงานได้เดี่ยวๆ ไม่ว่าจะเป็นการทำงานแบบกำหนดช่วงเวลา ควบคุมสัญญาณไฟจราจร สามารถที่จะทำงานได้เดี่ยวๆ ไม่ว่าจะเป็นการทำงานแบบกำหนดช่วงเวลาสัญญาณไฟล่วงหน้า (Fixed Time Operation) หรือจัดสัญญาณไฟตามสภาพจริง การควบคุมในลักษณะนี้ จะใช้ในกรณีที่ทางแยกนั้นเป็นทางแยกที่มีสัญญาณไฟจราจรอยู่แห่งเดียวในบริเวณนั้นหรือทางแยกอื่นๆ อยู่ห่างออกไปมากเกินกว่า 500 เมตร เป็นต้น

การควบคุมสัญญาณไฟแบบต่อเนื่อง (Linked Signal Control) บริเวณในเมืองไม่ควรใช้การควบคุมสัญญาณไฟแบบเดี่ยว ดังนั้นควรพิจารณาออกแบบสัญญาณไฟแบบต่อเนื่อง ซึ่งจะให้มีประสิทธิภาพสูงสุด สัญญาณไฟแบบนี้จะเชื่อมต่อกันกับทางแยกใกล้เคียงภายใต้ระบบควบคุมสัญญาณแบบโครงข่าย UPC (Urban Traffic Control)

การใช้งานอื่น ๆ ของสัญญาณไฟจราจร

สัญญาณไฟกระพริบที่เรียกว่า “WIG-WAG” ใช้ในการหยุดการจราจรในบริเวณต่าง ๆ เช่นทางรถไฟ บริเวณใกล้กับสนามบิน สถานีดับเพลิง และสถานีฉุกเฉินต่างๆ สัญญาณไฟนี้จะประกอบด้วยไฟแดง 2 ดวง ที่อยู่ในแนวขนานกันและกระพริบไป-มา มีไฟเหลือง 1 ดวง อยู่ตรงกลางระหว่างไฟแดงทั้งสองขณะที่เริ่มทำงานจะเปิดไฟเหลืองก่อนที่จะมีการเปิดไฟกระพริบสีแดง โดยทั่วไปสัญญาณนี้จะปิดไว้ตลอดจนกว่าจะมีการสั่งให้ทำงานโดยมือหรือโดยอุปกรณ์อัตโนมัติก็ได้

ในถนนที่มีความเร็วสูงการควบคุมช่องทางจราจรแบบพิเศษและสัญญาณบอกเหตุอื่นๆ สามารถใช้เพื่อหยุดการจราจรเมื่อมีเหตุการณ์บางอย่างเกิดขึ้นและอาจจะบังคับให้เปลี่ยนช่องทาง ,ลดความเร็ว, เตือนอันตรายหรือบังคับให้ออกที่ทางออกข้างหน้า

สัญญาณอาจจะวางไว้ที่ข้างถนนหรือวางบนทั้งบนช่องทางที่ต้องการ ส่วนมากการดำเนินการแบบนี้จะใช้กับแผนจัดการจราจรตามปริมาณการจราจรซึ่งทิศทางจราจรจะเปลี่ยนไปตามชั่วโมงเร่งด่วน การเปลี่ยนทิศทางนี้จะใช้สัญญาณไฟที่ติดตั้งไว้กับโครงเหล็กที่ติดตั้งไว้เหนือช่องทางจราจรเป็นตัวบ่งบอก ในงานก่อสร้างหรือบำรุงรักษาถนนอาจจะเป็นที่จะต้องใช้สัญญาณไฟจราจรชั่วคราว (สามารถเคลื่อนย้ายได้) เพื่อช่วยจัดการจราจร เช่น จัดเดินรถทางเดียว เป็นต้น

หลักในการควบคุมสัญญาณไฟจราจร

1. วงรอบเวลาในการปิด-เปิดสัญญาณไฟจราจร แต่ละรอบต้องไม่นานเกินไป โดยแต่ละรอบไม่ควรเกิน 4 นาที โดยประมาณ
2. การปิด-เปิดสัญญาณไฟจราจร แต่ละด้านต้องไม่นานเกินไป แต่ละด้านไม่ควรเกิน 1 นาที
3. การปล่อยแต่ละด้านไม่ควรข้ามจังหวะการปล่อยปกติ
4. การควบคุมการปิด-เปิด ต้องประสานสัมพันธ์กับแยกใกล้เคียงทุกด้าน

5. ควบคุมทิศทางตามความเหมาะสม เช่น เร่งระบายรถด้านเข้าเมืองในช่วงเช้าในชั่วโมงเร่งด่วน และระบายรถขาออกเมืองในช่วงเย็น

การจราจร เป็นระบบที่ค่อนข้างซับซ้อน มีองค์ประกอบหลักอยู่ 4 ส่วน ได้แก่ ผู้ใช้รถใช้ถนน (Drivers) ยานพาหนะ (Vehicles) ถนนหนทาง (Roadways) และอุปกรณ์ควบคุมการจราจร (Traffic Controls) ซึ่งปัจจัยทั้งสี่มีความเกี่ยวเนื่องสัมพันธ์กันจนก่อให้เกิดเป็นสภาพการจราจร จึงจำเป็นต้องทำความเข้าใจในองค์ประกอบทั้งสี่อย่างลึกซึ้งเพื่อสามารถนำความรู้ดังกล่าวไปใช้ในการออกแบบ วางแผน จัดการควบคุมระบบการจราจรได้อย่างเหมาะสมต่อไป

2.2 คุณลักษณะของผู้ใช้รถใช้ถนน

ผู้ใช้รถใช้ถนน คือ คนที่เดินทางอยู่บนท้องถนนทุกคน ไม่ว่าจะเป็นผู้ขับขี่ยานพาหนะ (รถยนต์ รถมอเตอร์ไซค์ รถเมล์) ผู้โดยสาร ผู้ขับขี่จักรยาน คนเดินเท้า และคนพิการ ผู้ใช้รถใช้ถนนนี้เป็นองค์ประกอบของระบบจราจรที่มีความซับซ้อนมากที่สุด เพราะคนแต่ละคนมีความรู้ความนึกคิด ความรู้ ความพร้อม สภาพร่างกายและอารมณ์แตกต่างกันไปในแต่ละบุคคลและยังแตกต่างกันไปในแต่ละช่วงเวลาด้วย ดังนั้นการออกแบบอุปกรณ์ควบคุมสภาพการจราจรต่างๆ ให้ใช้งานได้อย่างเหมาะสมกับผู้ใช้รถใช้ถนนทุกคนได้อย่างมีประสิทธิภาพและปลอดภัย ซึ่งต้องคำนึงถึงบุคคลกลุ่มต่างๆ ไม่ว่าจะเป็นผู้ใหญ่ เด็ก คนชรา คนพิการ และผู้ใช้รถใช้ถนนอื่นๆ ด้วย

คุณลักษณะของผู้ใช้รถใช้ถนนที่สำคัญบางตัวถูกนำมาใช้ในการวิเคราะห์วิศวกรรมจราจร ได้แก่ ระยะเวลาการรับรู้และตอบสนอง (Perception and reaction time) และ visual acuity ซึ่งคุณลักษณะเหล่านี้สามารถทำการตรวจวัดได้ไม่ยากนักสำหรับคุณลักษณะอื่นๆ เช่น ความสมบูรณ์แข็งแรงของร่างกาย การได้ยิน ความเหนื่อยล้า และสภาพจิตใจ ก็มีผลต่อการตัดสินใจของผู้ใช้รถใช้ถนนเช่นกัน แต่ปัจจัยเหล่านี้ทำการตรวจวัดได้ยากจึงไม่ค่อยนิยมนำมาใช้ในการวิเคราะห์มากนัก

2.2.1 ระยะเวลาการรับรู้และตอบสนอง (Perception and reaction time)

การขับขี่บนท้องถนนเป็นการกระทำการการรับรู้ทางสายตาและเสี่ยงอย่างต่อเนื่อง โดยผู้ขับขี่จะต้องคอยสนใจอยู่ตลอดเวลาและมีปฏิกิริยาตอบสนองอย่างทันท่วงที ถ้าวิเคราะห์กันอย่างละเอียดแล้วการรับรู้และตอบสนองจะประกอบด้วย 4 ขั้นตอน คือ

1. **การรับรู้ (Perception)** - การรับรู้ว่ามีเหตุการณ์หนึ่งเกิดขึ้น
2. **การเข้าใจหรือจำได้ (Intellection or Identification)** - ผู้ขับขี่จำได้ว่าเหตุการณ์นั้นคืออะไร จากการสะสมประสบการณ์ในอดีต
3. **การตัดสินใจ (Decision or Emotion)** – ผู้ขับขี่วิเคราะห์หาวิธีการตอบสนองที่เหมาะสมกับเหตุการณ์นั้นๆ
4. **ปฏิกิริยาตอบสนอง (Reaction or Volition)** – การดำเนินการอย่างใดอย่างหนึ่งตามการตัดสินใจข้างต้นตัวอย่างเช่น ถ้าผู้ขับขี่กำลังขับรถยนต์มาถึงทางแยกสัญญาณไฟจราจรแห่งหนึ่ง ซึ่งเมื่อกำลังเข้ามาใกล้ทางแยกนั้น สัญญาณไฟได้เปลี่ยนจากสีเขียว

เป็นสีแดง ผู้ขับขี่เริ่มต้นจากการรับรู้ นั่นคือการมองเห็นสัญญาณไฟจราจรเปลี่ยนจากสีเขียวเป็นสีแดง จำได้ว่าสัญญาณไฟสีแดงเป็นการให้สัญญาณหยุดรถ ณ เส้นหยุดรถ เพื่อให้รถยนต์จากทางด้านอื่นไปก่อน ซึ่งเป็นสิ่งที่ผู้ขับขี่ได้สั่งสมประสบการณ์เรื่อยมา จากนั้นจึงตัดสินใจที่จะชะลอความเร็วเพื่อหยุดรถ ณ เส้นหยุดรถ เนื่องจากคาดว่าจะไม่สามารถข้ามทางแยกได้ทันการณ์ จึงทำปฏิกิริยาตอบสนอง โดยการเหยียบเบรกเพื่อชะลอรถยนต์จนรถหยุดที่เส้นหยุด

ระยะเวลาตลอดกระบวนการสี่ขั้นตอนนี้ รวมกันเรียกว่าระยะเวลาการรับรู้และตอบสนอง หรือค่าเวลา **PIEV** (ตัวย่อของขั้นตอนทั้งสี่) ซึ่งค่านี้ได้นำไปใช้ในการออกแบบด้านวิศวกรรมจราจรมากมายจากกรณีดังกล่าวข้างต้นจึงสังเกตว่า ในขณะที่ผู้ขับขี่กำลังรับรู้และตอบสนองอยู่นั้น รถยนต์ก็ยังคงวิ่งต่อไปเรื่อยๆ ด้วยความเร็วคงที่ (เพราะกำลังรับรู้และตัดสินใจอยู่ แต่ยังไม่ได้เหยียบเบรกจริง) ซึ่งในกรณีที่ยานพาหนะใช้ความเร็วสูงหรือผู้ขับขี่บางกลุ่มที่มีเวลา PIEV มาก รถยนต์คันดังกล่าวก็จะวิ่งเป็นระยะทางอีกไกลพอสมควรจนผู้ขับขี่ไม่สามารถหยุดรถได้ทันการณ์ อาจจะก่อให้เกิดอุบัติเหตุได้ ระยะทางรวมที่รถยนต์กำลังวิ่งต่อเนื่องไปในขณะที่ผู้ขับขี่อยู่ในกระบวนการรับรู้และตอบสนองนั้น จะเรียกว่า **ระยะทาง PIEV**

โดยทั่วไป **ระยะทาง PIEV** (ในหน่วยเมตร) มีสูตรคำนวณดังนี้

$$d_p = \frac{1}{3.6} vt \qquad (2\text{-}1)$$

โดยที่ d_p – ระยะทางการรับรู้และการตอบสนอง (เมตร)

v – ความเร็วของยานพาหนะขณะนั้น (กม./ชม.)

t – ระยะเวลาการรับรู้และตอบสนอง หรือค่าเวลา PIEV (วินาที)

ในสภาพการณ์จราจรจริง ระยะเวลารับรู้และตอบสนองของแต่ละบุคคลจะแตกต่างกันไป ทั้งนี้ค่าดังกล่าวจะเพิ่มสูงขึ้นสำหรับเหตุการณ์ที่มีความซับซ้อน

โดยปรกติแล้ว ระยะเวลาการรับรู้และตอบสนอง จะขึ้นอยู่กับปัจจัยหลายประการ เช่น

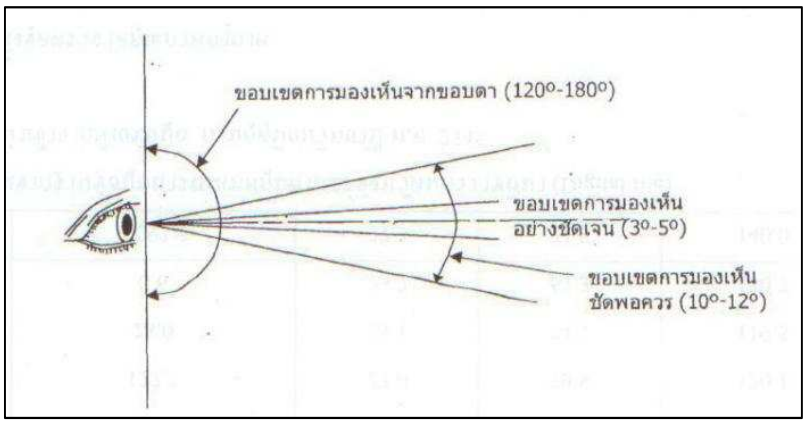
1. อายุ
2. ความล้า
3. ความซับซ้อนของเหตุการณ์
4. ความพิการทางกายภาพ
5. การอยู่ในสภาพมึนเมา

American Association of State Highway and Transportation Officials (AASHTO) ได้แนะนำให้ใช้ค่าเวลาการรับรู้และตอบสนอง (หรือค่า PIEV) เท่ากับ 2.5 วินาที ในการคำนวณระยะการเบรกหรือระยะการหยุดรถในสภาพการณ์ต่างๆ

2.2.2 การมองเห็นและการขับขี่

เป็นที่ทราบกันดีว่า ผู้ขับขี่ต้องสามารถมองเห็นท้องถนนจึงสามารถขับขี่รถยนต์ได้ อย่างไรก็ตามที่ผ่านมาไม่ค่อยมีการวิเคราะห์เกี่ยวกับรายละเอียดการขับขี่และ Visual Acuity กันมากนัก โดยส่วนใหญ่ในปัจจุบันการตรวจสอบสายตาในการสอบใบขับขี่ของกรมการขนส่งทางบกจะดำเนินการตรวจสอบเฉพาะความคมชัดในการมองวัตถุหยุดนิ่ง เช่น การอ่านค่าตัวเลขขนาดต่างๆ จากโปสเตอร์ที่ติดบนฝาผนัง ซึ่งในความเป็นจริงแล้ว การมองวัตถุหยุดนิ่งจะมีผลต่อการขับขี่น้อยกว่าการมองเห็นในรูปแบบอื่นๆ เช่น ความชัดเจนในการมองวัตถุเคลื่อนที่ (Dynamic visual acuity) การเห็นความใกล้ไกลของวัตถุ (Depth perception) การที่สายตากลับคืนสู่สภาพปกติหลังจากมองแสงจ้า (Glare recovery) และการมองเห็นวัตถุบริเวณขอบตา (Peripheral vision) เป็นต้น คุณลักษณะทางสายตาที่สำคัญเหล่านี้ยังไม่ได้รับการศึกษาวิจัยกันมากนัก ทั้งนี้เพราะว่าโดยปกติแล้วคนส่วนใหญ่ที่มีปัญหาทางด้านสายตาเหล่านี้จะตระหนักถึงความบกพร่องของตนเองและทำการปรับเปลี่ยนพฤติกรรมเพื่อให้แก้ปัญหาในชีวิตประจำวันอยู่แล้ว เช่น ผู้มีปัญหาทางด้านสายตากลับคืนสู่สภาพปกติหลังจากมองแสงจ้าจะต้องใช้เวลาในการคืนสภาพก็จะขับขี่รถยนต์ช้าลง และเพิ่มความระมัดระวังในการขับขี่ในเวลากลางคืน

รูปที่.2.1 แสดงขอบเขตการมองเห็นที่สำคัญสามประเภท ที่มีความสำคัญในการขับขี่ยานพาหนะ ประเภทแรกคือขอบเขตการมองเห็นอย่างชัดเจน (Field of clear or acute vision) ซึ่งจะครอบคลุมพื้นที่ประมาณ 3-5 องศาโดยรอบของแนวการมองของของตาดำ (พื้นที่การมองเห็นจะมีลักษณะเป็น cone) ผู้ขับขี่สามารถอ่านตัวอักษรหรือสัญลักษณ์ต่างๆ ในขอบเขตการมองเห็นนี้ได้อย่างชัดเจน ประเภทที่สองได้แก่ ขอบเขตการมองเห็นชัดพอควร (Field of fairly clear vision) ซึ่งจะมีพื้นที่การมองเห็นกว้างกว่าการมองเห็นประเภทแรก นั่นคือครอบคลุมพื้นที่ประมาณ 10-12 องศารอบแนวการมองของตาดำภายในขอบเขตนี้ ผู้ขับขี่สามารถระบุได้ว่าวัตถุมีสีอะไรและมีรูปร่างอย่างไร แต่ไม่สามารถอ่านตัวอักษรได้ ส่วนประเภทสุดท้าย ได้แก่ ขอบเขตการมองเห็นจากขอบตา (Field of peripheral vision) ซึ่งโดยปกติ จะครอบคลุมพื้นที่ประมาณ 120-180 องศาจากศูนย์กลางจากแนวการมองของสายตาดำในขอบเขตการมองเห็นดังกล่าวผู้ขับขี่ไม่สามารถบอกสีและรูปร่างของวัตถุได้ แต่สามารถรับรู้ถึงการเคลื่อนที่ของวัตถุในขอบเขตดังกล่าวได้



รูปที่.2.1 ขอบเขตการมองเห็น

ขอบเขตการมองเห็นนี้ มีความสำคัญในการออกแบบติดตั้งป้ายจราจร สัญญาณไฟจราจร และอุปกรณ์ควบคุมการจราจรประเภทต่างๆ เช่นในการวางป้ายเตือนที่มีตัวอักษรกำกับอยู่ ก็ควรจะติดตั้งที่ผู้ขับขี่สามารถมองเห็นด้วยขอบเขตการมองเห็นอย่าชัดเจน หรือ 3-5 องศาจากศูนย์กลางตาดำเพื่อให้ผู้สัญจรไปมาสามารถมองเห็นได้อย่างชัดเจนโดยที่ไม่ต้องละสายตาจากถนนเบื้องหน้า

สีและรูปร่างของวัตถุสามารถมองเห็นได้จากขอบเขตการมองเห็นอย่างชัดเจนพอควร ซึ่งมีมุมมองกว้างขึ้น ดังนั้น ป้ายจราจรต่างๆ จึงได้ถูกออกแบบให้มีสีและรูปร่างที่เป็นมาตรฐาน เช่น ป้ายหยุด จะมีพื้นสีแดงและมีรูปร่างแปดเหลี่ยม เป็นต้น

2.2.3 คุณลักษณะอื่นๆ ของผู้ขับขี่

การได้ยินก็เป็นปัจจัยอันหนึ่งที่ช่วยให้การขับขี่เป็นไปอย่างมีประสิทธิภาพและปลอดภัยมากยิ่งขึ้น อย่างไรก็ตามการศึกษาสถิติอุบัติเหตุในอดีตก็ไม่มีการยืนยันว่าผู้ที่พิการทางหูจะมีสถิติอุบัติเหตุจราจรมากกว่าผู้ขับขี่อื่นๆ ในปัจจุบันผู้พิการทุพลภาพ ก็สามารถขับขี่ยานพาหนะได้ โดยจะต้องมีการดัดแปลงสภาพรถยนต์เพื่อให้มีความเหมาะสมกับความพิการนั้นๆ ได้

2.3 คุณลักษณะของยานพาหนะ

คุณลักษณะของยานพาหนะบนท้องถนนมีความแตกต่างกัน มียานพาหนะหลายๆ ประเภทซึ่งมีรูปร่างและสมรรถนะที่แตกต่างกันไป ตั้งแต่ขนาดเล็ก เช่น รถมอเตอร์ไซค์ รถเก๋งสี่ล้อ รถเก๋งอเนกประสงค์ รถกระบะ รถสองแถว จนไปถึงยานพาหนะขนาดใหญ่ เช่น รถประจำทาง รถบรรทุก 6 ล้อ 10 ล้อ 18 ล้อ และรถพ่วง ดังนั้น วิศวกรจราจรจำเป็นจะต้องคำนึงถึงปัจจัยเหล่านี้ ในการออกแบบและจัดการระบบการจราจรบนท้องถนน

2.3.1 รูปร่างยานพาหนะมาตรฐาน (Design Vehicles)

ในประเทศสหรัฐอเมริกา American Association of State Highway and Transportation Officials (AASHTO) ได้กำหนดมาตรฐานรูปร่างยานพาหนะ (Design Vehicles) สำหรับยานพาหนะไว้ทั้งหมด 10 ประเภท สำหรับใช้ในการออกแบบคุณลักษณะทางเรขาคณิตของท้องถนน (Geometric Design) และโครงสร้างที่อำนวยความสะดวกต่างๆ เช่น ความกว้างช่องจราจร ความกว้างไหล่ทาง รัศมีของขอบทางและมุมถนน ระดับความสูงของโครงสร้างคร่อมช่องทางจราจรต่างๆ (Clearance heights) และองค์ประกอบอื่นๆ

ตารางที่ 2.1 แสดงขนาดและรูปร่างของยานพาหนะมาตรฐาน 10 ประเภท ที่กำหนดโดย AASHTO ดังแสดงในตาราง จะเห็นได้ว่ายานพาหนะแต่ละประเภทก็จะมีคุณลักษณะความกว้าง ความสูง ความยาวและขนาดอื่นๆ ที่กำหนดเป็นมาตรฐานสำหรับการออกแบบ ทั้งนี้ในกระแสการจราจรจริงยานพาหนะแต่ละประเภทก็มีรูปร่าง และขนาดแตกต่างกันอยู่บ้างเล็กน้อย รูปร่างของยานพาหนะมาตรฐานที่กำหนดขึ้น ถือว่าเป็นตัวแทนของยานพาหนะนั้นๆ สำหรับใช้ในการออกแบบเท่านั้น

ตารางที่ 2.1 ขนาดของยานพาหนะมาตรฐานที่ใช้สำหรับออกแบบ โดย AASHTO

ประเภทยานพาหนะ	สัญลักษณ์	ขนาด (ฟุต)				
		โดยรวม			เฉพาะส่วนที่ยื่น	
		ความสูง	ความกว้าง	ความยาว	ด้านหน้า	ด้านหลัง
รถยนต์ส่วนบุคคล	P	4.25	7	19	3	5
รถบรรทุก	SU	13.5	8.5	30	4	6
รถประจำทาง	BUS	13.5	8.5	40	7	8
รถประจำทางพ่วง	A-BUS	10.5	8.5	60	8.5	9.5
รถบรรทุก						
รถบรรทุกกึ่งพ่วง - กลาง	WB-40	13.5	8.5	50	4	6
รถบรรทุกกึ่งพ่วง - ใหญ่	WB-50	13.5	8.5	55	3	2
รถบรรทุกพ่วง	WB-60	13.5	8.5	65	2	3
ยานพาหนะสำหรับพักผ่อน						
รถแบบอยู่อาศัยได้	MH		8	30	4	6
รถพ่วงบ้านแค้มป์	P/T		8	49	3	10
รถพ่วงเรือ	P/B		8	42	2	8

2.3.2 อัตราการเร่งของยานพาหนะ

ยานพาหนะแต่ละประเภท จะมีขนาด น้ำหนักและกำลังขับเคลื่อนที่แตกต่างกันไป ทำให้มีสมรรถภาพการเร่งความเร็วที่แตกต่างกันด้วย ซึ่งสรุปไว้ใน ตารางที่ 2.2

ตารางที่ 2.2 อัตราการเร่งยานพาหนะแต่ละประเภท

ประเภท	น้ำหนักสุทธิ	อัตราความเร่งสูงสุด		
ยานพาหนะ	(ปอนด์)	0-15 ไมล์/ชม.	จาก 40ไมล์/ชม.	จาก 60ไมล์/ชม.
รถยนต์ - ใหญ่	4,800	10.0	4.0	2.5
รถยนต์ - กลาง	4,000	8.0	4.0	2.0
รถคอมแพกต์	3,000	8.0	3.0	1.1
รถยนต์-เล็ก	2,100	6.0	1.2	0.7
รถปิคอัพ	5,000	8.0	1.8	1.5
รถบรรทุก	12,000	2.0	0.6	0.6
รถบรรทุก - ใหญ่	45,000	2.0	0.4	-

จากตารางจะเห็นได้ว่า โดยทั่วไปแล้ว ยานพาหนะขณะที่ใช้ความเร็วต่ำ สามารถเร่งความเร็วได้ดีกว่าขณะที่ใช้ความเร็วสูง และยานพาหนะที่มีขนาดเล็ก และน้ำหนักเบากว่าสามารถเร่งความเร็วได้ดีกว่า

ยานพาหนะที่มีขนาดใหญ่และมีน้ำหนักมาก ซึ่งคุณลักษณะนี้มีผลต่อการออกแบบท้องถนนเป็นอย่างมาก เพราะค่าที่ต้องใช้ในการคำนวณระยะทางที่สามารถหยุดรถได้โดยปลอดภัย โดยใช้สมการ

$$d_a = \frac{1}{7.2}at^2 \quad (2\text{-}2)$$

โดยที่ d_a – ระยะทางที่รถยนต์เคลื่อนที่ไปขณะที่เร่งความเร็ว (เมตร)
a – อัตราการเร่งความเร็ว (กม./ชม./วินาที)
t – ระยะเวลาในการเร่งความเร็ว (วินาที)

ความแตกต่างในการเร่งสูงสุดของยานพาหนะแต่ละประเภท ส่งผลให้ประสิทธิภาพของการจราจรในระบบลดต่ำลง ตัวอย่างเช่น รถบรรทุกขนาดใหญ่ ติดรอสัญญาณไฟจราจร ณ ทางแยกแห่งหนึ่ง เมื่อได้รับสัญญาณไฟเขียว ก็จะเร่งความเร็วได้ช้า ทำให้ยานพาหนะประเภทอื่นที่ตามหลังมาเกิดความล่าช้าไปด้วย

2.3.3 การชะลอตัวของยานพาหนะ

การชะลอตัวของยานพาหนะถือเป็นคุณลักษณะของยานพาหนะที่มีความสำคัญมากที่สุด ในด้านการออกแบบบนท้องถนน และความปลอดภัยการจราจร ทั้งระยะเวลา และระยะทางที่จำเป็นต้องใช้ในการชะลอรถยนต์จนหยุดนิ่งสนิท จะนำมาใช้ประกอบการออกแบบและปฏิบัติการของการจราจรในแทบทุกๆ ด้าน

ประสิทธิภาพในการชะลอตัวของยานพาหนะ จะขึ้นอยู่กับปัจจัยหลายๆ ประการ ได้แก่ ระบบเบรกของยานพาหนะ ชนิดและสภาพของยางรถยนต์ ตลอดจนประเภทและสภาพของพื้นผิวถนน โดยปรกติการคำนวณที่จำเป็นต้องใช้ในการชะลอรถยนต์ จะใช้สมการ

$$d_b = \frac{v^2 - u^2}{254.0(f+g)} \quad (2\text{-}3)$$

โดยที่ d_b – ระยะทางที่รถยนต์เคลื่อนที่ไประหว่างการชะลอความเร็ว (เมตร)
v – ความเร็วเริ่มต้นก่อนการชะลอความเร็ว (กม./ชม.)
u – ความเร็วสุดท้ายหลังจากชะลอความเร็ว (กม./ชม.)
f – ค่าสัมประสิทธิ์ของความต้านทานการลื่นไถล (Coefficient of skidding friction)
g – ความชันของถนน (Decimal)

ทั้งนี้ ระยะทางชะลอรถยนต์ที่คำนวณได้จากสมการ (2-3) ข้างต้นนี้ เป็นระยะทางในช่วงระหว่างการชะลอความเร็วหลังจากเริ่มเหยียบเบรกแล้วเท่านั้น จะไม่รวมถึงเวลารับรู้ตอบสนอง ในกรณีที่ความชันเป็นแบบลงเขา ก็ให้ใส่ค่าความลาดชันพร้อมด้วยเครื่องหมายลบ

ตารางที่ 2.3 แสดงค่าสัมประสิทธิ์ความต้านทานการลื่นไถลที่นิยมใช้ในการออกแบบ ซึ่งจะแปรผันตามชนิดของพื้นผิวถนน ความเร็วก่อนเริ่มชะลอความเร็ว สภาพยางของรถยนต์ และสภาพพื้นผิวถนนว่าแห้งหรือเปียก

ตารางที่ 2.3 ค่าสัมประสิทธิ์ความต้านทานการลื่นไถลที่ใช้ในการออกแบบ

ประเภทของพื้นผิว	พื้นผิวแห้ง		พื้นผิวเปียก
	ยางใหม่ที่ได้มาตรฐาน	ยางที่เสื่อมสภาพแล้ว	ค่าออกแบบที่แนะนำโดย AASHTO
	ความเร็ว = 10 mph (17.7 km/h)		
Dry bit. Conc.	0.74	0.61	-
Sand asphalt	0.75	0.66	-
Rock asphalt	0.78	0.73	-
Port. Cem. Conc.	0.76	0.68	-
	ความเร็ว = 20 mph (32.2 km/h)		
Dry bit. Conc.	0.76	0.60	0.40
Sand asphalt	0.75	0.57	0.40
Rock asphalt	0.76	0.65	0.40
Port. Cem. Conc.	0.73	0.50	0.40
	ความเร็ว = 30 mph (48.2 km/h)		
Dry bit. Conc.	0.79	0.57	0.36
Sand asphalt	0.79	0.48	0.36
Rock asphalt	0.74	0.59	0.36
Port. Cem. Conc.	0.78	0.47	0.36
	ความเร็ว = 40 mph (64.4 km/h)		
Dry bit. Conc.	0.75	0.48	0.33
Sand asphalt	0.75	0.39	0.33
Rock asphalt	0.74	0.50	0.33
Port. Cem. Conc.	0.76	0.33	0.33
ทุกพื้นผิว	ความเร็ว = 50 mph (80.5 km/h)		0.31
	ความเร็ว = 60 mph (96.5 km/h)		0.30
	ความเร็ว = 70 mph (112.6 km/h)		0.29
	ความเร็ว = 80 mph (128.7 km/h)		0.27

2.3.4 คุณลักษณะในการขับเคลื่อน

ในขณะที่ยานพาหนะเคลื่อนที่นั้น จะมีแรงต้านทานภายนอกต่างๆ มาทำให้รถช้าลง ได้แก่ แรงต้านทานจากอากาศ (Air resistance) แรงต้านทานจากความลาดชัน (Grade resistance) แรงต้านทานในการ

ขับเคลื่อน (Rolling resistance) แรงต้านทานจากการเข้าโค้ง (Curve resistance) แรงต้านทานจากความต้านทานบนผิวถนน (Friction resistance)

2.3.5 ตัวอย่าการคำนวณระยะทางการรับรู้และตอบสนอง และระยะทางการชะลอความเร็ว

ดังที่กล่าวมาแล้ว ระยะทางการชะลอรถยนต์ถือเป็นตัวแปรหลักในการออกแบบท้องถนน เพราะมีความสำคัญทางด้านความปลอดภัย ยานพาหนะจะต้องชะลอตัวได้ทันท่วงทีเพื่อหลีกเลี่ยงการเกิดอุบัติเหตุ ระยะทางดังกล่าวจะประกอบด้วย 2 ส่วน คือ ระยะทางการรับรู้และตอบสนอง ซึ่งเริ่มต้นขึ้นจากการที่ผู้ขับขี่สามารถมองเห็นเหตุการณ์ข้างหน้า จนกระทั่งเริ่มเหยียบเบรก และระยะทางที่ใช้ในการชะลอความเร็วยานพาหนะภายหลังจากการเหยียบเบรกแล้ว ดังนั้น สามารถคำนวณได้โดยใช้สูตร

$$d_s = d_p + d_b$$

$$d_s = \frac{1}{3.6} vt + \frac{v^2 - u^2}{254.0(f+g)} \quad (2\text{-}4)$$

ระยะทางการหยุดรถอย่างปลอดภัย

โดยหลักการแล้วการออกแบบท้องถนนนั้น จะต้องให้ผู้ขับขี่มีระยะการมองเห็นได้ไม่น้อยกว่าระยะการหยุดรถอย่างปลอดภัย ทั้งนี้เพื่อให้ผู้ขับขี่ เมื่อพบเห็นสิ่งที่กีดขวางถนนข้างหน้า เช่นอุบัติเหตุ ของตกหล่น เป็นต้น ก็สามารถชะลอความเร็วยานพาหนะได้อย่างทันท่วงที

ระยะเวลาการเปลี่ยนสัญญาณไฟ ณ ทางแยก

โดยทั่วไปจะเห็นว่าการเปลี่ยนสัญญาณไฟจราจรจากไฟเขียวเป็นไฟแดงนั้น จะต้องให้สัญญาณไฟเหลืองในระยะเวลาสั้นๆ ก่อนเสมอ ทั้งนี้ เพื่อให้ยานพาหนะคันที่ไม่สามารถหยุดรถที่เส้นหยุดได้ทัน สามารถเร่งความเร็วให้ผ่านทางแยกได้อย่างปลอดภัย ก่อนที่สัญญาณไฟเขียวทางด้านอื่นจะเริ่มต้น ถ้าไม่มีสัญญาณไฟเหลือง กล่าวคือ เปิดไฟเขียวให้ด้านอื่นไปทันที จะส่งผลให้เกิดอุบัติเหตุขึ้นมากมาย ดังนั้น จึงต้องมีการพิจารณาระยะเวลาการเปลี่ยนสัญญาณไฟ (Clearance interval) ในการออกแบบไฟสัญญาณจราจรด้วย

ในการวิเคราะห์หาระยะเวลาการเปลี่ยนสัญญาณไฟจราจร (ช่วงเวลาสั้นๆ ที่แสดงสัญญาณไฟเหลือง และไฟแดงทุกด้าน) นั้น จะพิจารณาจากระยะทางการหยุดรถอย่างปลอดภัย ในขณะที่สัญญาณไฟจราจรเปลี่ยนจากสีเขียวเป็นสีแดงนั้น ยานพาหนะที่ยังคงอยู่ห่างจากทางแยกมาก ระยะห่างจากทางแยกมากกว่าระยะการหยุดรถอย่างปลอดภัยนั้น จะสามารถชะลอความเร็วและจอดหยุดนิ่งที่เส้นหยุดได้ทัน ส่วนยานพาหนะที่กำลังเข้าสู่ทางแยก ระยะห่างจากทางแยกน้อยกว่าระยะการหยุดรถอย่างปลอดภัย จะไม่สามารถหยุดรถที่ทางแยกได้ทัน ดังนั้น วิศวกรจะต้องออกแบบสัญญาณไฟจราจรให้รถยนต์เหล่านี้ สามารถเคลื่อนตัวผ่านทางแยกได้อย่างปลอดภัย ก่อนที่จะเปิดสัญญาณไฟเขียวให้กับขาอื่นของทางแยก

2.4 ทฤษฎีการวิเคราะห์การออกแบบสัญญาณไฟจราจร

การวิเคราะห์การออกแบบสัญญาณไฟจราจรมีด้วยกันหลายวิธี ซึ่งได้แก่ รอบเวลาสัญญาณไฟ (Signal Cycle) เวลาล่าช้าของการเริ่มต้นและสุดท้าย (Start and End Lag Time) การสูญเปล่าของการเคลื่อนที่ (Movement Lost Time) การเคลื่อนที่วิกฤต (Critical Movement) เวลาสูญเปล่าของทางแยก (Intersection Lost Time) สามารถสรุปได้ดังนี้

2.4.1 รอบเวลาสัญญาณไฟ (Signal Cycle)

ลำดับเวลาจังหวะสัญญาณไฟช่วงเวลาที่ครบ 1 รอบของวงรอบ เรียกว่า รอบเวลาสัญญาณไฟ (c) ผลรวมของเวลาระหว่างไฟเขียวและเวลาแสดงไฟเขียวของทุกเฟส คือรอบเวลาสัญญาณไฟ (Cycle Time) สามารถคำนวณได้จากสมการ

$$c = \sum(I + \mathrm{G}) \tag{2-5}$$

เมื่อ c คือ รอบเวลาสัญญาณไฟ (วินาที)

I คือ เวลาระหว่างไฟเขียว (วินาที)

G คือ เวลาแสดงไฟเขียว (วินาที)

2.4.2 เวลาล่าช้าของการเริ่มต้นและสุดท้าย (Start and End Lag Time)

Start Lag (a) คือ ผลรวมของเวลาช่วง Intergreen บวกกับเวลาที่สูญเสียช่วงเริ่มต้น

End Lag (b) คือ เวลาล่าช้าของการสิ้นสุด

ดังนั้นจะมีสมการดังนี้

$$\text{Start Lag, } a = I + ee' \tag{2-6}$$

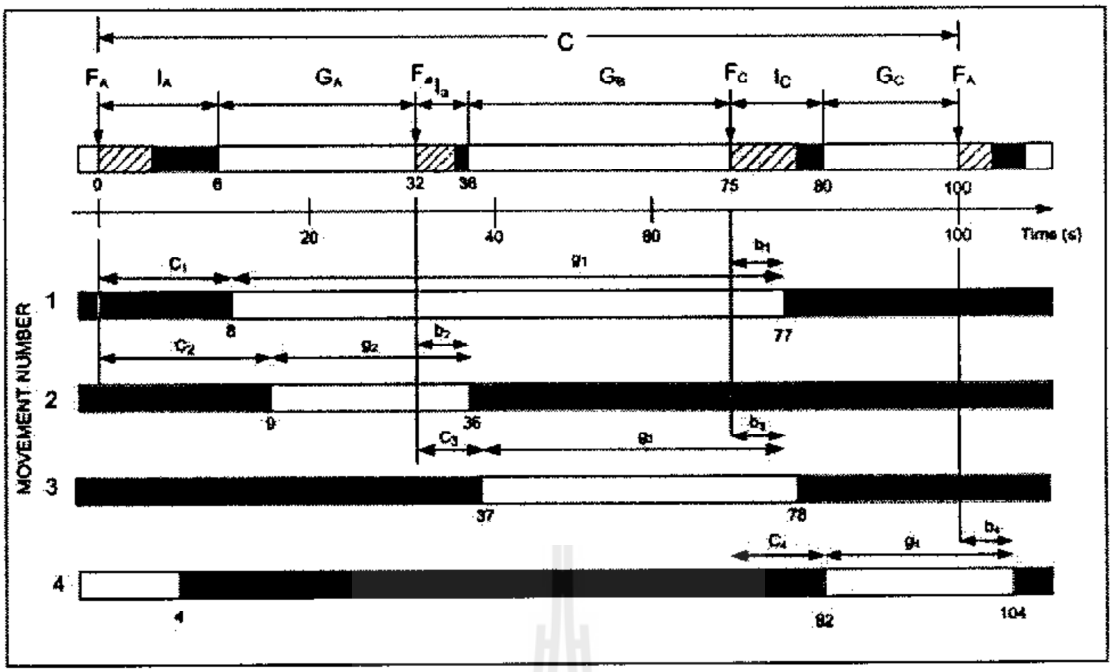
$$\text{End Lag, } b = ff' \tag{2-7}$$

เมื่อ I คือ เวลาระหว่างไฟเขียว (วินาที)

ee' คือ เวลาล่าช้าของการเริ่มต้น (วินาที)

ff' คือ เวลาล่าช้าการสิ้นสุด (วินาที)

สำหรับ เวลาระหว่างไฟเขียวของการเคลื่อนที่ ก็คือ เวลาระหว่างไฟเขียวของเฟสที่กำลังจะเริ่มต้นของการเคลื่อนที่ โดยช่วงประสิทธิภาพเขียวจะเริ่ม (F_i + a) และเวลาสิ้นสุด คือ (F_k + b) เมื่อ F_i และ F_k เป็นเวลาของการเปลี่ยนเฟส ซึ่งเริ่มต้นและหยุดตามลำดับ



รูปที่.2.2 แสดงเวลาล่าช้าของการเริ่มต้นและสุดท้าย (Start and End Lag Time)

2.4.3 การสูญเปล่าของการเคลื่อนที่ (Movement Lost Time)

การสูญเปล่าของการเคลื่อนที่ (Movement Lost Time) คือ ผลต่างระหว่าง Start Lag Time กับ End Lag Time สามารถคำนวณได้ดังสมการ

$$l = a - b$$

$$= I + ee' - ff' \qquad (2\text{-}8)$$

เมื่อ l คือ เวลาสูญเปล่าของการเคลื่อนที่ (วินาที)

ee' คือ เวลาล่าช้าของการเริ่มต้น (วินาที)

ff' คือ เวลาล่าช้าการสิ้นสุด (วินาที)

จากการสูญเสียเปล่าของการเคลื่อนที่ สามารถสร้างความสัมพันธ์ระหว่าง Movement Lost Time กับ Display Green Time (G) และเวลาไฟเขียวประสิทธิภาพ (g) ได้ดังสมการ

$$g + l = G + I \qquad (2\text{-}9)$$

2.4.4 การเคลื่อนที่วิกฤต (Critical Movement)

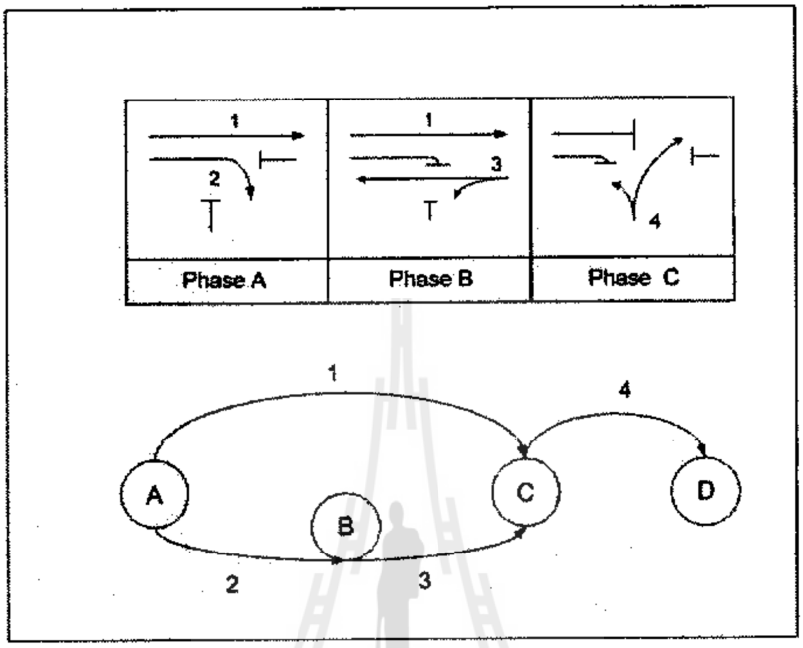
เป็นการเคลื่อนที่ที่ใช้ในการคำนวณความจุและจังหวะสัญญาณไฟของทางแยกจากสมการ 2-5 การเคลื่อนที่วิกฤตได้อธิบายรอบสัญญาณไฟในรูปผลบวก เวลาไฟเขียวประสิทธิผลและเวลาที่สูญเสียไฟของการเคลื่อนที่ ในขณะการเคลื่อนที่วิกฤตสามารถคำนวณได้จากสมการ

$$c = \sum(g + l) \quad (2\text{-}10)$$

เมื่อ c คือ รอบสัญญาณไฟ (วินาที)

g คือ เวลาไฟเขียวประสิทธิภาพ สำหรับการเคลื่อนที่วิกฤต (วินาที)

l คือ เวลาสูญเปล่าไฟของการเคลื่อนที่สำหรับการเคลื่อนที่วิกฤต (วินาที)



รูปที่.2.3 การสูญเปล่าของการเคลื่อนที่ (Movement Lost Time)

2.4.5 เวลาสูญเปล่าของทางแยก (Intersection Lost Time)

เวลาสูญเปล่าของทางแยก (L) สามารถคำนวณได้จากสมการ

$$L = \sum l \quad (2\text{-}11)$$

เมื่อ L คือ เวลาสูญเปล่าของทางแยก (วินาที)

l คือ เวลาสูญเปล่าไปของการเคลื่อนที่สำหรับการเคลื่อนที่วิกฤต (วินาที)

2.5 ทฤษฎีการวิเคราะห์ประสิทธิภาพการทำงานของทางแยกที่ควบคุมด้วยสัญญาณไฟ

2.5.1 การพิจารณาการเคลื่อนที่ (Movement Consideration)

ความจุของทางแยกขึ้นอยู่กับอัตราสูงสุดที่รถยนต์สามารถเคลื่อนที่ออกจากทางแยก คือ ปริมาณจราจรอิ่มตัว (Saturation Flow, s) ความจุของการเคลื่อนที่ที่ทางแยก คำนวณได้จาก

$$Q = s(\frac{g}{c}) \quad (2\text{-}12)$$

เมื่อ Q คือ ความจุ (คัน/ชั่วโมง)
s คือ ปริมาณจราจรอิ่มตัว (คัน/ชั่วโมง)
c คือ รอบสัญญาณไฟ (วินาที)
g คือ ระยะเวลาไฟเขียว (วินาที)

2.5.2 ความล่าช้า (Delay)

ความล่าช้าโดยประมาณสำหรับการเคลื่อนที่ที่ทางแยกเดี่ยวแบบกำหนดเวลาแน่นอน สามารถคำนวณจากสมการการจำลองความล่าช้าสม่ำเสมอของ Webster ดังสมการที่ 2-13

$$UD = \frac{0.50C\left[1-\frac{g}{c}\right]^2}{1-\left(\frac{g}{c}\right)x} \quad (2\text{-}13)$$

เมื่อ UD คือ ความล่าช้าเฉลี่ยสม่ำเสมอต่อคัน (วินาที/คัน)
C คือ รอบเวลาสัญญาณไฟ (วินาที)
g คือ ประสิทธิภาพไฟเขียว (วินาที)
x คือ ความจุความอิ่มตัว
c คือ ความจุของทางแยก (คัน/ชั่วโมง)

2.6 ระบบควบคุมสัญญาณไฟจราจร

รูปแบบระบบควบคุมสัญญาณไฟจราจรมี 2 ประเภท

1. Fixed Time Plan (ระบบสัญญาณไฟจราจรแบบตั้งเวลาไว้ล่วงหน้า)

เป็นระบบที่ใช้การเก็บข้อมูลสถิติสภาพจราจรที่ผ่านมาในอดีต เพื่อมาคำนวณรอบสัญญาณไฟ ตั้งเวลาของตู้ควบคุม โดยสามารถแยกได้เป็นช่วงเวลาในแต่ละวัน และสามารถแยกได้ระหว่างวันทำงาน (จันทร์ – ศุกร์) และวันหยุด (เสาร์ – อาทิตย์)

2. Vehicle Actuated (ระบบสัญญาณไฟจราจรแบบปรับตามยานพาหนะ)

เป็นระบบที่มีการติดตั้ง Detector บริเวณทางแยกเพื่อประมวลการเดินทางของรถ โดยทำการตรวจสอบรถที่เดินทางเข้ามาในบริเวณทางแยกว่ามีหรือไม่ ถ้ามีจะทำการปล่อยสัญญาณไฟจนกระทั่งหมดรถหรือ ถึงค่าสัญญาณไฟเขียวสูงสุดที่ตั้งไว้ ในกรณีที่มีมีรถเดินทางเข้ามาระบบจะทำการตัดเวลาไฟเขียวดังกล่าวออกไป มีการทำงานดังนี้

2.1 ระบบจะใช้ค่าตั้งต้นจากแผนที่ตั้งไว้ในแต่ละช่วงเวลาของวัน โดยการกำหนดเวลาไฟเขียวต่ำสุดและเวลาไฟเขียวมากสุด

2.2 ระบบมีการติดตั้งลูป (Loop Detector) ในช่องจราจรที่มีความต้องการในการผ่านทางแยกน้อยที่สุด (มุ่งสู่ทางโท) เพื่อส่งข้อมูลไปคำนวณการให้เวลารอบสัญญาณไฟเขียวแก่ทิศทางนั้น โดยระบบจะเริ่มให้จากให้เวลาไฟเขียวต่ำสุด และจะเพิ่มเวลาให้ทีละน้อยในกรณีที่ยังมีรถทับลูปอยู่เรื่อยๆ จนกระทั่งถึงค่าสูงสุดของเวลาที่ตั้งไว้ แต่ถ้า

ไม่มีรถทับลูปนานเกินช่วงเวลาหนึ่ง ระบบจะหยุดการให้ไฟเขียวในจังหวะนั้น เพื่อรถการสูญเสียเวลาโดยเปล่าประโยชน์

2.3 ความยาวรอบสัญญาณไฟ (Cycle Length) จะเป็นผลจากการเพิ่มหรือลดเวลาสัญญาณไฟในแต่ละจังหวะเพื่อตอบสนองสภาพจราจร ไม่มีการควบคุม

2.4 เนื่องจากไม่มีการควบคุมรอบสัญญาณไฟ จึงสามารถพิจารณาค่าเหลื่อมเวลา (Offset) ได้เมื่อสัญญาณไฟในแต่ละทางแยกไม่เท่ากัน สัญญาณไฟจราจรของทั้งระบบจะไม่มีรูปแบบ (Pattern) ทำให้การพยากรณ์การมาถึงของรถจากแยกที่แล้วทำได้ยาก ระบบจึงไม่สามารถคำนวณค่าเหลื่อมเวลาได้

3. Area Traffic Control (ระบบสัญญาณไฟจราจรแบบเป็นพื้นที่)

เป็นระบบควบคุมสัญญาณไฟจราจรที่สามารถปรับเปลี่ยนได้เองอัตโนมัติ (Adaptive Signal Control) การควบคุมในลักษณะนี้อาศัยการตรวจวัดปริมาณการจราจรที่เข้ามาที่ทางแยกโดยผ่านเครื่องตรวจนับรถ (Detector) โดยจะปรับสัญญาณไฟให้เหมาะสมกับสภาพการจราจรที่วัดได้ในขณะนั้น

ตารางที่ 2.4 แสดงข้อดี-ข้อเสียของระบบ VA และ ATC

	VA (Vehicle Actuated)	ATC (Area Traffic Control)
ความซับซ้อนของระบบ	เรียบง่าย เนื่องจากสามารถเชื่อมต่อข้อมูลจากลูปโดยตรงไปใช้การประมวลผลจาก CPU ของตู้ควบคุมสัญญาณไฟจราจรได้โดยตรง	ซับซ้อนกว่า เนื่องจากในแต่ละทางแยกจะต้องเพิ่มคอมพิวเตอร์ เพื่อประมวลผลเพิ่มเติมจาก CPU ของตู้ควบคุม (CPU ตู้จราจรมีประสิทธิภาพพอจะคำนวณได้)
อุปกรณ์เพิ่มเติมจากตู้จราจรปกติ	การ์ด CPU รุ่นที่สามารถทำงานในโหมด VA ได้ + การ์ดเชื่อมต่อข้อมูลลูป + (อุปกรณ์เชื่อมโยงสัญญาณ)	การ์ด CPU รุ่นที่รองรับระบบ ATC + Embedded Computer + การ์ดเชื่อมต่อข้อมูลลูป + (อุปกรณ์เชื่อมโยงสัญญาณ) * CPU ที่รองรับ ATC สามารถทำงานในโหมด VA ได้เช่นกัน
แนวคิดของระบบ	ลดการสูญเสียเวลาไฟเขียวจากการไม่ได้ใช้ประโยชน์ (ปกติจะใช้เฉพาะในทิศทางรองที่มีการจราจรเบาบาง)	ลดเวลาการรอคอยให้น้อยที่สุดในทุกทิศทางของทางแยก และให้รถติดไฟแดงน้อยที่สุดในทิศทางหลักของทางแยกที่เชื่อมต่อกัน
หลักการทำงาน	ระบบจะเริ่มจากให้เวลาไฟเขียวต่ำที่สุด และจะเพิ่มเวลาให้ทีละน้อยในกรณีที่ยังคงมีรถทับลูปอยู่เรื่อยๆ จนกระทั่งถึงค่าสูงสุด แต่ถ้าหากไม่มีรถทับลูปนานเกินช่วงเวลาหนึ่ง ระบบจะหยุดการให้ไฟเขียวในจังหวะนั้น	ประเมินสภาพจราจรในทุกทิศทางในระยะเวลาหนึ่ง เพื่อคำนวณและปรับปรุงสัญญาณไฟจราจรทีละน้อย (Adaptive) เพื่อให้ได้สัญญาณไฟจราจรที่ดีที่สุดของทั้งทางแยกและของทั้งเส้นทาง (กรณีที่มีการประสานสัมพันธ์ทางแยก)

	VA (Vehicle Actuated)	ATC (Area Traffic Control)
การใช้ข้อมูลจากลูป	จะดูในช่วงเวลาก่อนหน้าสั้นๆ ว่ามีรถทับลูปหรือไม่ เพื่อเพิ่มเวลาไฟเขียวไปเรื่อยๆ	มีการเก็บเก็บข้อมูลจากลูปทั้งช่วงเวลาไฟเขียวในทุกจังหวะ เพื่อใช้ประเมินสภาพจราจรนำมาคำนวณการปรับปรุงรอบสัดส่วนของสัญญาณไฟเมื่อสิ้นสุดรอบสัญญาณไฟ
การชี้วัดสภาพจราจร	ไม่มีตัวชี้วัดที่สามารถเปรียบเทียบสภาพจราจรระหว่างสองทิศทางได้	ชี้วัดสภาพจราจรได้จากค่า Degree of Saturation ทำให้สามารถปรับสัดส่วนสัญญาณไฟจราจรให้เหมาะสมในแต่ละทิศทางได้
การเพิ่มและตัดเวลาไฟเขียวเฉพาะหน้า	จะเพิ่มเมื่อยังมีรถทับลูป และจะตัดเวลาไฟเขียวเมื่อไม่มีรถในช่วงเวลาหนึ่ง	มีเฉพาะการตัดเวลาไฟเขียวเมื่อไม่มีรถทับลูปในช่วงเวลาหนึ่ง การเพิ่มเวลาจะเกิดขึ้นจากการคำนวณรอบสัญญาณไฟจราจร
เวลารอคอย	สามารถลดเวลารอคอยได้จากการตัดเวลาไฟเขียวเมื่อไม่มีรถทับลูป แต่จะไม่คำนึงถึงผลกระทบต่อการรอคอยของรถในจังหวะอื่น จากการเพิ่มเวลาไฟเขียวไปเรื่อยๆ ในกรณีที่ยังมีรถทับลูป	ประมวลผลโดยคำนึงถึงภาพรวมของรอบสัญญาณไฟและสัดส่วนของจังหวะสัญญาณไฟจราจร มีการคำนึงว่าการเพิ่มเวลาไฟเขียวในทิศทางหนึ่งจะมีผลกระทบต่อเวลารอคอยรถในอีกทิศทาง (ระบบพยายามลดเวลารอคอยของทุกทิศทาง)
การประสานสัมพันธ์ทางแยก	ไม่สามารถทำได้เนื่องจากไม่มีการควบคุมสัญญาณไฟ รอบสัญญาณไฟเปลี่ยนแปลงไปตามการมีหรือไม่มีทับลูปทำให้รอบสัญญาณไฟไม่เท่ากันในแต่ละทางแยก จึงไม่สามารถจัด Offset ได้	สามารถประสานสัมพันธ์ทางแยกได้ เนื่องจากมีระบบคำนวณกลาง (อยู่ที่ศูนย์ควบคุมที่ตู้ควบคุมสัญญาณไฟจราจรก็ได้) ที่คอยคำนวณรอบสัญญาณไฟกลาง (Common Cycle Length) และค่าเหลื่อมเวลา Offset และแต่ละทางแยกสามารถปรับรอบสัญญาณไฟให้เข้ากับรอบสัญญาณไฟกลางได้
ความเหมาะสมกับสภาพจราจร	เหมาะสมกับสภาพจราจรที่ไม่สมดุล (เช่น จังหวะหนึ่งรถมาก อีกจังหวะหนึ่งรถน้อย) หรือสภาพจราจรที่ไม่คับคั่ง เนื่องจากเมื่อมีรถมาก จะมีรถทับลูปตลอดเวลาไฟเขียวจะไม่เกิดขึ้นซึ่งจะเท่ากับการใช้ระบบ Fixed time	เหมาะสมกับทุกสภาพจราจร โดยสามารถรองรับสภาพการจราจรติดขัดได้ในระดับหนึ่ง เนื่องจากระบบสามารถปรับปรุงรอบและสัดส่วนไฟสัญญาณได้เอง เพื่อตอบสองต่อสภาพจราจรได้ในจังหวะนั้น ทั้งนี้ในกรณีที่สภาพการจราจรเบาบางและไม่มีปัญหามาก การใช้ระบบ Fixed time จะคุ้มค่ากว่า
การเปลี่ยนแปลงสภาพการจราจร	ไม่รองรับสภาพการจราจรที่เปลี่ยนแปลง (เช่น ในวันหยุดติดต่อยาว) เนื่องจากพื้นฐานของระบบคือแผนตามช่วงเวลา และระบบไม่สามารถชี้วัดสภาพจราจรได้	มีการปรับปรุงเวลาสัญญาณไฟให้เหมาะสมกับสภาพจราจรในแต่ละช่วงเวลา (Adaptive) ทำให้สัญญาณไฟสามารถปรับได้ตามสภาพจราจรที่เปลี่ยนแปลงไป

การควบคุมสัญญาณไฟจราจร โดยระบบ VA และ ATC เป็นที่ยอมรับอย่างกว้างขวางว่าสามารถเพิ่มประสิทธิภาพในการควบคุมการจราจรได้ โดยนิยมเรียกระบบการควบคุมสัญญาณไฟจราจรแบบนี้ว่า **ระบบควบคุมสัญญาณไฟจราจรอัจฉริยะ** และสามารถลดการใช้พลังงานในภาคขนส่งได้ โดยเฉพาะอย่างยิ่งบริเวณจุดตัดถนนสายหลักและถนนสายรอง (ตรอก/ซอย) โดยใช้หลักการคือ การเพิ่มประสิทธิภาพการใช้ประโยชน์จากไฟเขียวให้มากที่สุด โดยที่การควบคุมสัญญาณไฟจราจร โดยระบบ Fixed Time Plan ไม่สามารถทำได้ เพราะบ่อยครั้งจะพบว่าจะมีการเปิดสัญญาณไฟเขียวโดยไม่มีรถยนต์จอดติดสัญญาณไฟจราจรอยู่เลย

ระบบควบคุมสัญญาณไฟจราจรอัจฉริยะ(ATC) ได้รับการติดตั้งอยู่หลายจุดทั้งกรุงเทพฯ ปริมณฑล และเมืองใหญ่ๆ เช่น พัทยา หาดใหญ่ เชียงใหม่ ฯลฯ และยังไม่มีการรายงานและประเมินผลถึงการประหยัดพลังงานอย่างชัดเจน อันเนื่องมาจากสาเหตุ คือ

1. บริเวณกรุงเทพมหานครและปริมณฑล เป็นบริเวณที่มีจราจรหนาแน่น มีการใช้ประโยชน์จากสัญญาณไฟเขียวเต็มที่
2. การจราจรในตัวเมืองของประเทศไทยมียานพาหนะประเภท รถมอเตอร์ไซค์ จำนวนมากและอุปกรณ์ตรวจจับ (Detector) มีความไวไม่เพียงพอ

2.7 องค์ประกอบของเครื่องควบคุมไฟจราจร

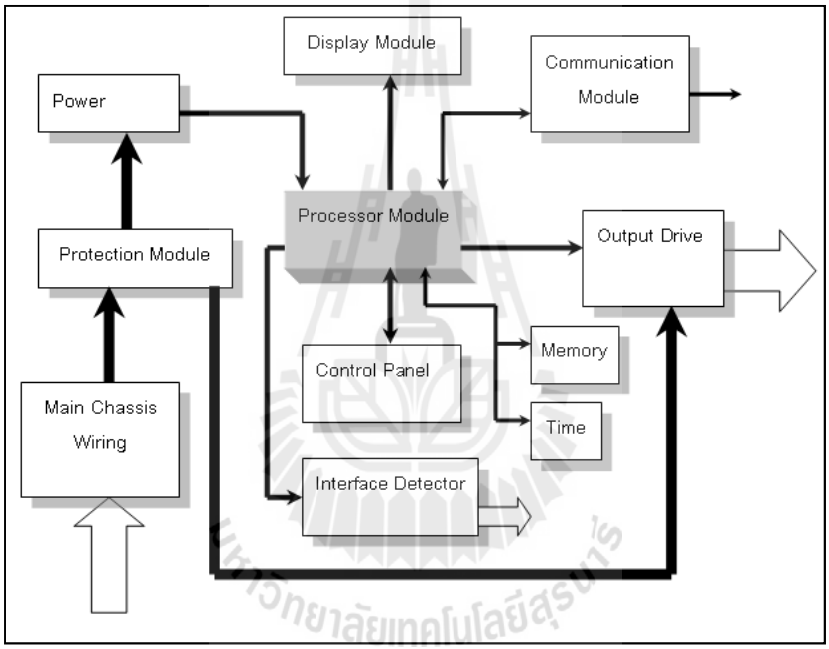
จากการทบทวนถึงงานวิจัยที่เกี่ยวข้องกับเครื่องควบคุมการจราจรในประเทศไทยสามารถออกแบบหลักการทำงานของฮาร์ดแวร์ไว้ดังรูปที่.2.4

- ส่วนโปรเซสเซอร์ (Processor Module) ทำหน้าที่เก็บรวบรวมข้อมูลและโปรแกรมควบคุม ทั้งยังเป็นเสมือนสมองของเครื่องควบคุมสัญญาณไฟจราจร ที่คอยสั่งการ และรับรู้เงื่อนไขต่างๆ เช่น เงื่อนไขทางเวลาเพื่อนำไปเลือกโปรแกรมทำงาน, เงื่อนไขจากปุ่มควบคุมเพื่อกำหนดรูปแบบการจราจรตามที่คนควบคุมต้องการ เป็นต้น
- ส่วนแผงควบคุม (Control Panel Module) เป็นส่วนที่รวบรวมปุ่มควบคุม และเป็นสื่อกลางในการติดต่อระหว่างผู้ใช้กับเครื่อง (Man-Machine Interface) ส่วนขับหลอดไฟจราจร (Output Drive Module) ทำหน้าที่แปลงสัญญาณ ที่ได้รับจากส่วนโปรเซสเซอร์ให้เป็นไฟฟ้ากระแสสลับ 220 โวลต์ เพื่อในไปขับหลอดไฟจราจร
- ส่วนติดต่อสื่อสาร (Communication Module) ทำหน้าที่ในการเชื่อมโยงเครื่องควบคุมกับอุปกรณ์สื่อสารเช่น โมเด็ม (Modem) เพื่อการส่งข้อมูลระยะไกล หรือติดต่อกับส่วนป้อนโปรแกรมแบบมือถือ (Hand Held Programmer) หรือกับคอมพิวเตอร์เพื่อป้อนข้อมูลต่างๆ รวมทั้งโปรแกรมควบคุมให้กับควบคุม
- ส่วนบันทึกข้อมูล (Memory Module) เพื่อนำมาใช้ประมวลผล
- ส่วนฐานเวลา (Time Module)
- ส่วนแสดงผล (Display Module) ใช้ในการแสดงข้อมูลการทำงาน และตั้งข้อมูล
- ส่วนจ่ายกำลังไฟฟ้า (Power Supply Module)

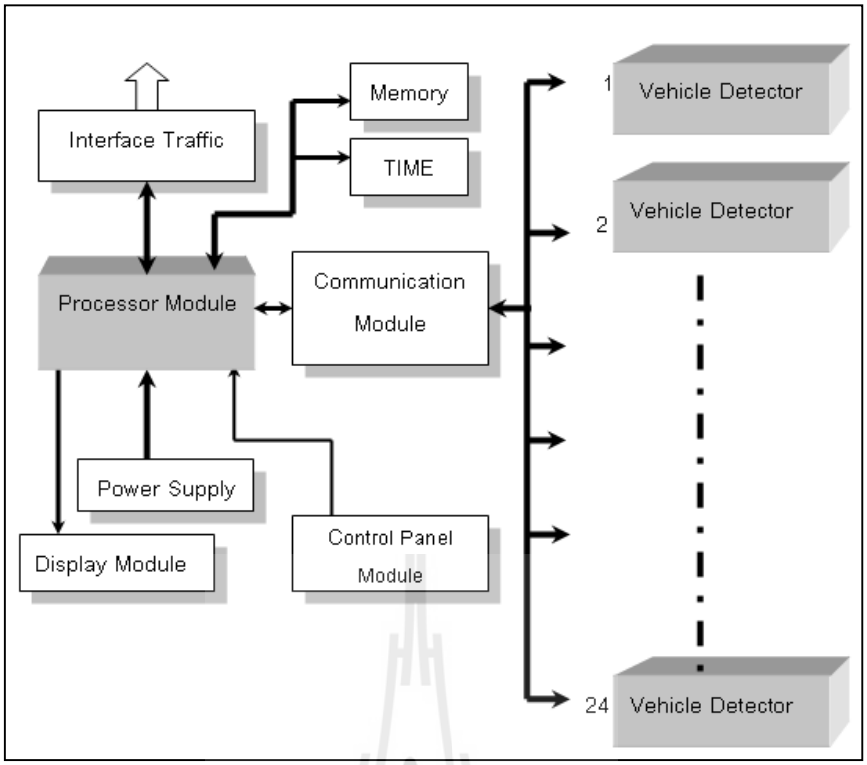
นอกจากนี้ยังมี ส่วนของวงจรป้องกันแบบทรานเซี้ยต์ และส่วนรวบรวมสายไฟเข้าสู่ตู้ควบคุม (Main Chassis Wiring Module) ซึ่งถือเป็นส่วนสนับสนุนการทำงานของส่วนอื่นของเครื่องควบคุมสัญญาณไฟจราจร

ส่วนที่ต้องมีเพิ่มเติมการทำงานของฮาร์ดแวร์คือ ส่วนโปรเซสเซอร์ (Processor Interface Module) ที่ใช้ในการดำเนินงานทางด้านเครือข่ายตัวตรวจวัด ส่วนติดต่อสื่อสาร (Communication Module) ใช้เชื่อมต่อกับตัวตรวจวัดด้วยมาตรฐาน RS-485

ส่วนประกอบสุดท้ายที่สำคัญ ตัวตรวจวัด ซึ่งมีหน้าที่สำคัญในการตรวจดูปริมาณจราจร การเข้ามาบริเวณทางแยกของยานพาหนะต่างๆ ซึ่งในแยกหนึ่งๆ อาจมี ตัวตรวจวัด ตั้งแต่ 1 ตัวจนถึง 24 ตัวดูรูปที่.2.5 ประกอบ



รูปที่.2.4 ลักษณะพื้นฐานของเครื่องควบคุมสัญญาณไฟจราจร



รูปที่.2.5 ลักษณะพื้นฐานของส่วนประมวลผลกับตัวตรวจจับยานพาหนะ

2.8 ระบบตัวตรวจวัด (Detectors)

เป็นปัจจัยสำคัญในการตรวจสอบยานพาหนะบนท้องถนน ตัวตรวจวัดจะใช้หลักการทำงานของขดลวดเหนี่ยวนำ จากขดลวดหรือสายไฟมาพันเป็นรูปสี่เหลี่ยมผืนผ้า, สี่เหลี่ยมจัตุรัส, หรือเป็นรูปวงกลม โดยมีจำนวนรอบตั้งแต่หนึ่งรอบขึ้นไปจะทำให้ขดลวดหรือสายไฟเส้นนั้นมีคุณสมบัติเป็นตัวเหนี่ยวนำ (Inductor) ค่าความเหนี่ยวนำ (Inductance) ที่ได้อยู่ที่ขนาด และจำนวนรอบการคำนวณทั่วไปของค่าเหนี่ยวนำคือ

$$L = P (t^2 + t) / 4 \qquad (2\text{-}14)$$

โดย L = ค่าความเหนี่ยวนำ (ไมโครเฮนรี)

P = เส้นรอบรูป (ฟุต)

t = จำนวนรอบ

ค่าความไวในการตรวจวัดยานพาหนะ

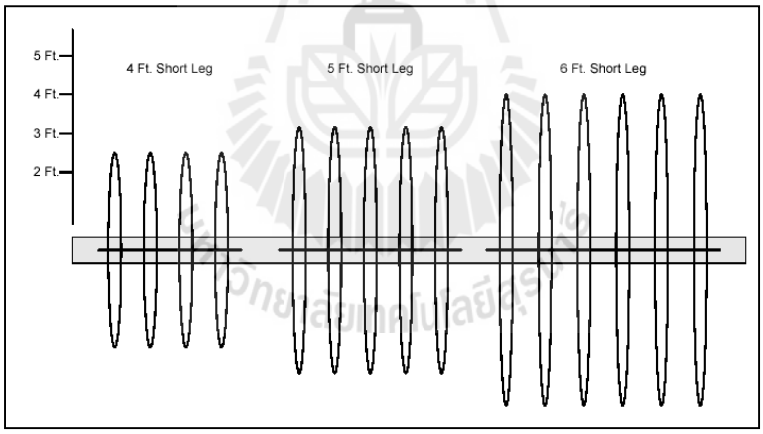
จากสมการ

$$\text{Amount of Change} \approx \frac{\text{Vehicle Size}}{\text{Caused by Vehicle} \quad (\text{Loop Size}) \times (\text{Vehicle Height})} \qquad (2\text{-}15)$$

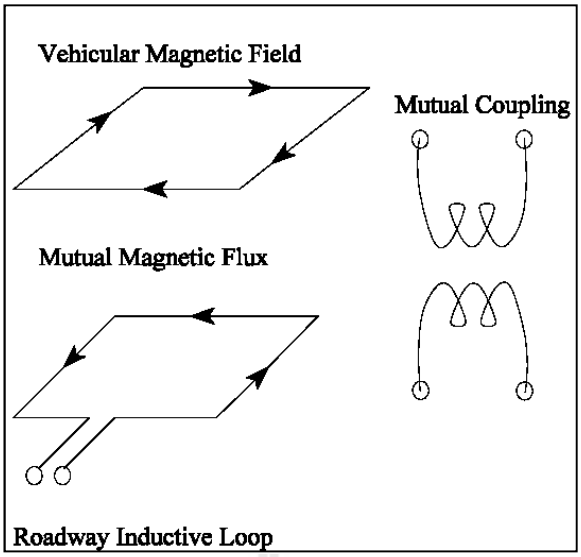
โดย Vehicle Size = ขนาดของยานพาหนะ
Loop Size = ขนาดของลูป
Vehicle Height = ความสูงของยานพาหนะ

ดังนั้นจากสมการที่ (2-14) และ (2-15) จะทำให้ได้ว่า

- การเพิ่มขนาดของลูป จะลดการเปลี่ยนแปลงที่กระทำจากยานพาหนะ
- ถ้ายานพาหนะมีความสูง ความเหนี่ยวนำที่เกิดจากยานพาหนะก็จะลดลง
- ถ้ายานพาหนะมีขนาดเล็กการเปลี่ยนแปลงที่กระทำจากยานพาหนะก็จะน้อยลง
- การเพิ่มจำนวนรอบไม่มีผลต่อความสูงที่ต้องการตรวจวัด
- ขนาดของลูปมีผลต่อความสูงในการตรวจวัด



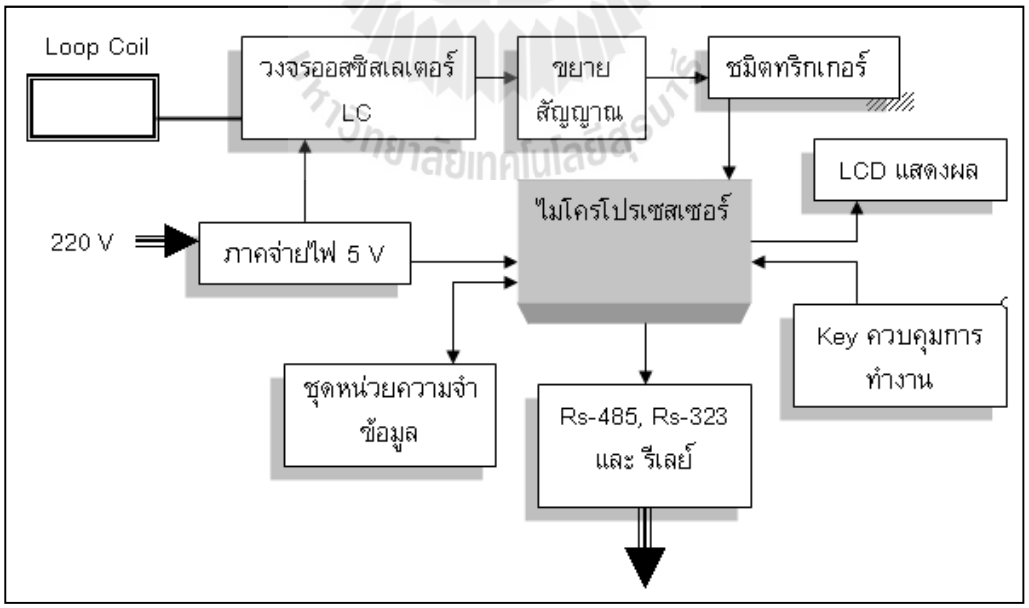
รูปที่.2.6 แสดงลำดับความสูงที่เหมาะสมในการตรวจวัด



รูปที่.2.7 แสดงผลการกระทำระหว่างยานพาหนะกับขดลวดเหนี่ยวนำ

วงจรการทำงาน

ขดลวดจะถูกฝังลงบริเวณพื้นดิน และเมื่อตัวตรวจจับได้ไปพบกับวัตถุที่เป็นยานพาหนะ จะทำให้ความถี่ในภาคผลิตความถี่อิสระดังที่ได้กล่าวมาแล้วแต่ตอนต้นเกิดการเปลี่ยนแปลงความถี่ การเปลี่ยนแปลงความถี่ดังกล่าวนี้จะถูกขยายสัญญาณและแปลงสัญญาณเป็นรูปสี่เหลี่ยมเข้าไปในส่วนวิเคราะห์ข้อมูลด้วยอัลกอลิทึมในไมโครคอนโทรลเลอร์ บล็อกไดอะแกรมดังแสดงในรูปที่.2.8



รูปที่.2.8 ลักษณะวงจรพื้นฐานของส่วนเครื่องตรวจจับยานพาหนะแบบลูปเหนี่ยวนำ

2.9 วิวัฒนาการของระบบตรวจวัดยานพาหนะ

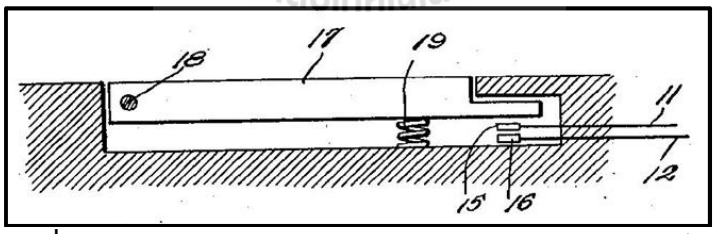
อุปกรณ์ตรวจวัดการจราจร (Traffic flow sensors) คือ อุปกรณ์ที่ใช้ในการตรวจวัดการมีอยู่ (presence) หรือการผ่าน (passage) ของยานพาหนะ ณ บริเวณจุดสำรวจให้ข้อมูลที่ใช้ในการจัดการจราจร และข้อมูลที่ใช้ในการวางแผน

พัฒนาการของเทคโนโลยีตรวจวัดการจราจร

ค.ศ. 1928 Horn-activated traffic signal (developed by Charles Adler,Jr.) เป็นเครื่องมือวัดปริมาณรถโดยอาศัยการติดตั้ง Micro Phone

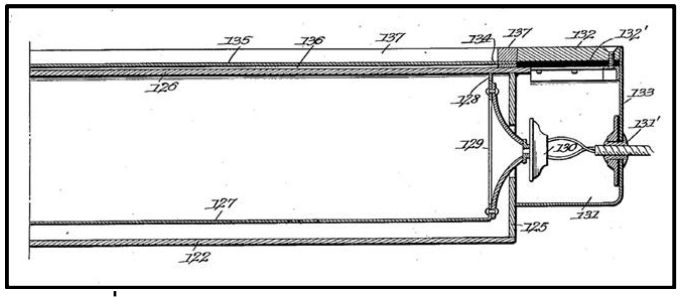
รูปที่.2.9 การตรวจวัดยานพาหนะโดยใช้ Micro Phone

ค.ศ. 1928 In-roadway pressure sensitive sensor (developed by Henry A. Haugh) เป็นเครื่องมือวัดปริมาณรถโดยอาศัยกระแสไฟฟ้าที่ได้จากแรงกดทับของแผ่นเหล็กที่ติดตั้งไว้ใต้ผิวถนน



รูปที่.2.10 การตรวจวัดยานพาหนะโดยการกดทับของแผ่นเหล็ก

ค.ศ. 1930 In-roadway pressure sensitive sensor (developed by Charles Adler,Jr.)) เป็นเครื่องมือวัดปริมาณรถโดยอาศัยเสียงซึ่งได้มาจากการติดตั้งเครื่องวัดเสียงที่ติดตั้งไว้ใต้ผิวถนน



รูปที่.2.11 การตรวจวัดยานพาหนะโดยอาศัยเสียง

ซึ่ง In-roadway pressure sensitive sensor ได้รับความนิยมมากแต่ก็ยังมีปัญหาซึ่งสรุปได้ดังนี้

1. อุปกรณ์มีราคาแพง
2. ต้องติดตั้งอุปกรณ์ใหม่หากมีการปรับปรุงผิวถนน
3. ความเสียหายเนื่องจากรถกวาดหิมะ

2.9.1 อุปกรณ์ตรวจวัดการจราจรที่ติดตั้งอยู่บนหรือใต้ผิวทาง (In-roadway sensors)

1. Pneumatic road tube

เริ่มใช้ตั้งแต่ปี 1920s เป็นเครื่องตรวจวัดการจราจรที่ติดตั้งโดยอาศัยท่อยางเพื่อตรวจวัด

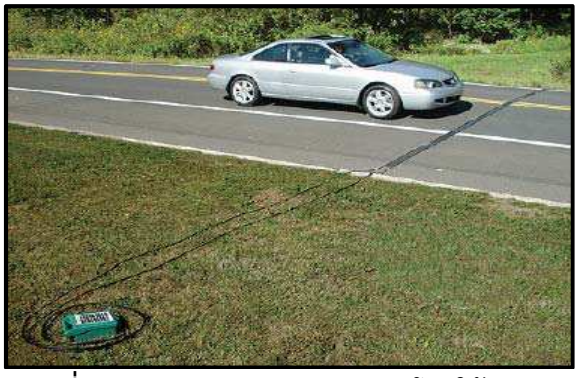
หลักการทำงาน

น้ำหนักรถกดทับท่อลมเป็นเครื่องตรวจวัดการจราจรที่ติดตั้งบนผิวทางวางขวางช่องทางเดินรถ เคลื่อนที่อาศัยน้ำหนักรถกดทับท่อลม ซึ่งทำด้วยยางแล้วเกิดแรงดันกระจายไปในแนวท่อยางโดยปลายด้านหนึ่งของท่อยางปิดตันและอีกด้านต่อเข้ากับอุปกรณ์ระบบไฟฟ้า ซึ่งทำงานด้วยแรงดันที่เปลี่ยนแปลงในเรือท่อทำให้ทราบจำนวนการกดทับและวิ่งผ่านท่อลมซึ่งสามารถจัดเก็บข้อมูลได้ดังนี้

- ปริมาณจราจร
- ความเร็วรถ
- แยกประเภทรถโดยใช้จำนวนเพลาและระยะห่างระหว่างเพลา,เวลาห่างระหว่างรถ (Gap)

การติดตั้ง Pneumatic road tube

- เลือกตำแหน่งที่ต้องการติดตั้ง
- ติดตั้ง End plug เพื่อกันน้ำหรือฝุ่นเข้า
- ยึดปลายด้านหนึ่งของท่อลมให้แน่นกับพื้นถนน
- ติดท่อลมกับผิวถนนด้วยเทปกาวแอสฟัลท์
- วัดระยะเพื่อติดตั้งท่อลมอีกเส้น (ถ้ามี)
- ติดปลายท่ออีกด้านเข้ากับเครื่องนับและถ้าทำการติดตั้ง Two-pneumatic road tubes connected together จะช่วยลดข้อผิดพลาดจากการวัดระยะห่างท่อลมขณะติดตั้ง



รูปที่.2.12 การตรวจวัดยานพาหนะโดยใช้ท่อลม

ข้อดี

- ใช้งานง่าย
- ราคาถูก

ข้อเสีย

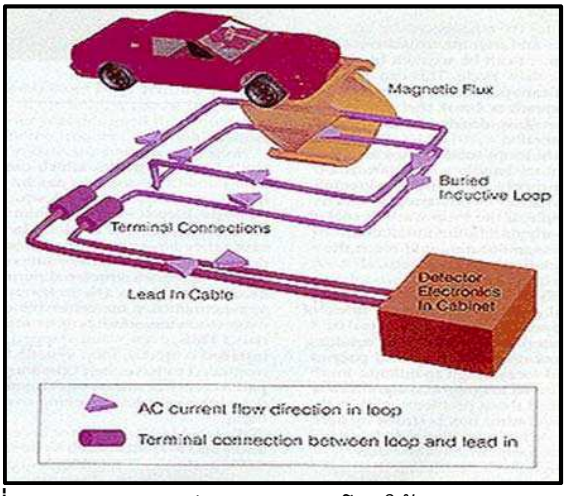
- ท่อลมชำรุดได้ง่ายถ้ามีปริมาณการจราจรสูง
- การติดตั้งเหมาะสมกับการติดตั้งแบบชั่วคราวและเป็นบริเวณที่
- ไม่ซับซ้อน
- การติดตั้งต้องปิดการจราจรถึงจะทำการติดตั้งได้

2. Inductive loop detectors

เริ่มใช้ตั้งแต่ปี 1960s เป็นเครื่องมือสำหรับจัดการจราจรที่ตรวจวัดการเปลี่ยนแปลงของสนามแม่เหล็กไฟฟ้า

หลักการทำงาน

เปลี่ยนแปลงค่าความต้านทานการเหนี่ยวนำของกระแสไฟฟ้าในขดลวดเหนี่ยวนำ



รูปที่.2.13 การตรวจวัดยานพาหนะโดยใช้ Inductive loop

มหาวิทยาลัยเทคโนโลยีสุรนารี

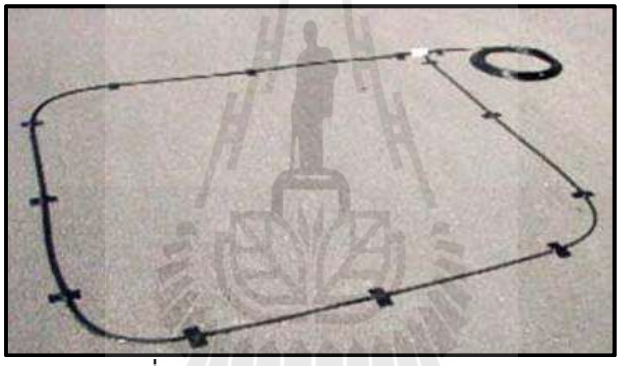
ซึ่งประเภทของ Inductive loop detectors มีดังนี้

- Saw cut loops เป็นการกัดพื้นผิวของถนนแล้วทำการฝังสาย Loop ซึ่งมี

วิธีการติดตั้ง Saw-cut loops ดังนี้

1. วาง layout ของ loop บนผิวถนน
2. กัดผิวถนน
3. ติดตั้งสาย Loop
4. ปิด Loop ด้วย Sealant

- Trenched-in loops เป็นการวางสาย Loop ก่อนการทำถนน
- Pre-formed loops เป็นการทำสาย Loop สำเร็จรูปแล้วนำมาวางบนพื้นถนนแล้วนำเทปกาวติดเพื่อยึดสาย Loop ไว้บนถนนซึ่งมีข้อเสียคือจะไม่มั่นคงแต่จะมีข้อดีตรงที่สามารถเคลื่อนย้ายง่ายเมื่อต้องการนำไปติดตั้ง



รูปที่.2.14 Inductive loop บนถนน

หมายเหตุ ตัวอย่าง Layout ของ Loop detectors ต่างๆ เช่น รูปวงกลม, รูปสี่เหลี่ยม, รูปหกเหลี่ยมเป็นต้น

เครื่องตรวจวัดสามารถจัดเก็บข้อมูลได้ดังนี้

- ปริมาณจราจร
- ความเร็วรถ
- การมีรถบริเวณนั้น (Presence)
- ระยะเวลาครอบครอง (Occupancy)
- ระยะห่างของรถแต่ละคัน (Gap)
- Headway
- แยกประเภทรถโดยใช้ vehicle signature

ข้อดี

- เป็นเทคโนโลยีที่มีการพัฒนามานานมีคนใช้เยอะ
- ให้ข้อมูลการนับรถที่ถกต้องที่สุด
- สามารถนำไปประยุกต์ใช้กับงานได้หลายรูปแบบ
- ไม่ไวต่อสภาพอากาศที่เปลี่ยนแปลงใช้งานง่าย

ข้อเสีย

- การติดตั้งจำเป็นต้องมีการกัดผิวทาง
- การติดตั้งที่ไม่ดีทำให้ผิวทางมีอายุการใช้งานสั้นลง
- ต้องมีการปิดการจราจรขณะติดตั้งและซ่อมแซม
- ในแต่ละจุดต้องใช้ loop หลายตัวในการตรวจวัดการจราจร
- น้ำหนักกดทับและอุณหภูมิส่งผลต่อ loop ที่ฝังไว้

3. Magnetic sensors

หลักการทำงาน

การเปลี่ยนแปลงของสนามแม่เหล็กโลกซึ่งประเภทของ Magnetic sensors มีดังนี้

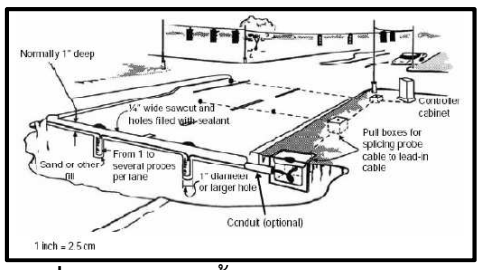
- Two-axis fluxgate magnetometer
- Induction or search coil magnetometer

เครื่องตรวจวัดสามารถจัดเก็บข้อมูลได้ดังนี้

- ปริมาณจราจร
- ความเร็วรถ
- การมีรถบริเวณนั้น (Presence)
- ระยะเวลาครอบครอง (Occupancy)
- แยกประเภทรถโดยใช้ vehicle magnetic signature

การติดตั้ง Magnetic Sensors

เจาะรูปบนผิวถนนความลึกประมาณ 5 cm แล้วนำ Portable magnetic sensor วางไว้แล้วนำแผ่นยางปิดบริเวณดังกล่าวโดยใช้ตะปูตอกเพื่อยึดแผ่นยางไม่ได้แผ่นยางเปิดวิธีนี้ง่ายต่อการติดตั้ง



รูปที่.2.15 การติดตั้ง Magnetic Sensors

ข้อดี

- สามารถใช้ในบริเวณที่ไม่สามารถติดตั้ง loop ได้
- ไม่ไวต่อสภาพอากาศ
- ผลจากแรงกดทับจากปริมาณจราจรมีน้อยกว่า loop

ข้อเสีย

- การติดตั้งจำเป็นต้องมีการกัดผิวทางหรือเจาะเป็นช่องใต้ผิวทาง
- ไม่สามารถตรวจจับรถที่จอดนิงอยู่กับที่ได้หากไม่มีการวาง layout
- ของอุปกรณ์ตรวจวัดแบบพิเศษและใช้ software เสริม

2.9.2 อุปกรณ์ตรวจวัดการจราจรทีติดตั้งอยู่เหนือผิวทาง (Over-roadway sensors)

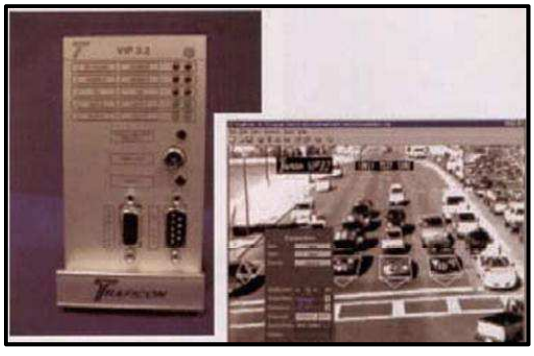
1. Video Image Processors

เริ่มใช้ตั้งแต่ปี 1970s

หลักการทำงาน

วิเคราะห์ภาพจาก video frames เครื่องตรวจวัดสามารถให้ข้อมูลได้ดังนี้

- ปริมาณจราจร
- ความเร็วรถ
- การมีรถบริเวณนั้น (Presence)
- ระยะเวลาครอบครอง (Occupancy)
- ระยะห่างของรถแต่ละคัน (Gap)
- Headway
- แยกประเภทรถโดยใช้ความยาว
- ความหนาแน่นของปริมาณรถ
- ความยาวของแถวรถที่ติดอยู่บนท้องถนน



รูปที่.2.16 การตรวจวัดยานพาหนะโดยใช้ Video Image Processors

ข้อดี

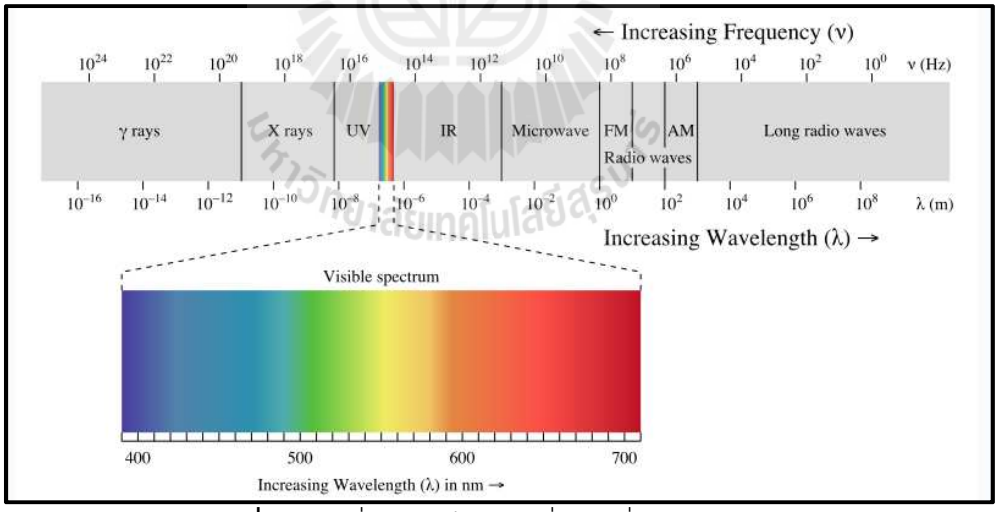
- สามารถตรวจวัดได้หลายช่องจราจรพร้อมๆกัน
- ง่ายในการเพิ่มพื้นที่ตรวจวัด
- ให้ข้อมูลจราจรหลายอย่าง
- สามารถตรวจวัดได้ในบริเวณกว้างเมื่อมีการเชื่อมข้อมูลระหว่างกล้อง

ข้อเสีย

- ต้องมีการปิดช่องจราจรหากติดตั้งกึ่งกลางถนน
- ต้องมีการทำความสะอาดเลนส์ตามระยะเวลา
- สภาพอากาศและความสั่นสะเทือนมีผลต่อประสิทธิภาพของ กล้อง
- ต้องมีไฟส่องสว่างที่เพียงพอในเวลากลางคืน
- ต้องติดตั้งกล้องจากมุมสูง
- จะคุ้มค่าก็ต่อเมื่อมีการตรวจวัดหลายพื้นที่พร้อมกันในกล้องเดียวหรือต้องการข้อมูลบางอย่างเป็นพิเศษ

2. Microwave radar sensors

คลื่นไมโครเวฟ (Microwaves) เป็นคลื่นแม่เหล็กไฟฟ้าที่มีความถี่อยู่ในช่วง 3000 MHz คลื่นไมโครเวฟได้ถูกนำมาประยุกต์ใช้ในอุปกรณ์ต่างๆ เช่น เตาไมโครเวฟโทรศัพท์มือถือรวมถึงเครื่องมือในการตรวจวัดการจราจร



รูปที่.2.17 คลื่นแม่เหล็กไฟฟ้าที่ความถี่ต่างๆ

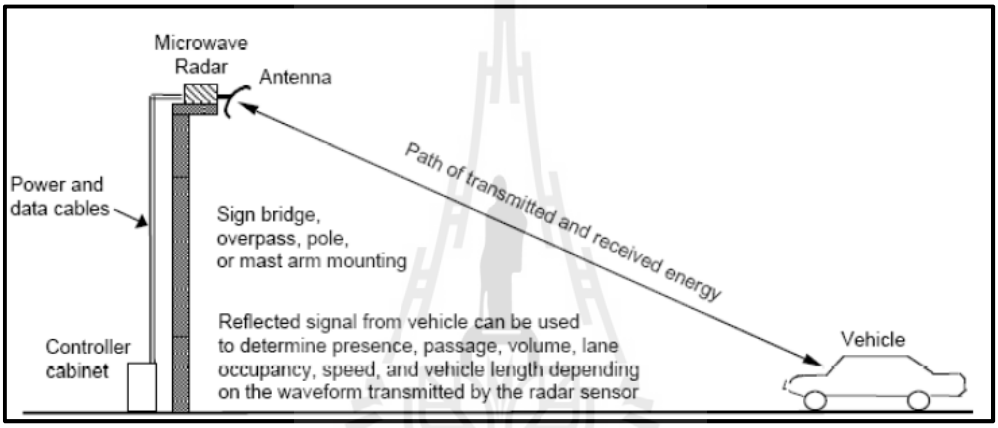
เครื่องมือตรวจวัดการจราจรแบบ Microwave Radar Sensors มี 2 ประเภทคือ

1. แบบ Doppler Microwave
2. แบบ Frequency – modulated continuous wave

หลักการทำงาน

เครื่องตรวจวัดจะส่งสัญญาณและสะท้อนกลับของคลื่น Microwave ออกมาเพื่อตรวจจับรถยนต์ที่วิ่งผ่านแนวสัญญาณเครื่องตรวจวัดประมวลผลออกมาเป็นข้อมูลดังนี้

- ปริมาณจราจร
- ความเร็วรถ
- การมีรถบริเวณนั้น (Presence)
- ระยะเวลาครอบครอง (Occupancy)
- ระยะห่างของรถแต่ละคัน (Gap)
- Headway
- ประเภทรถยนต์จำแนกตามความยาวรถแต่ละคัน



รูปที่.2.18 การตรวจวัดยานพาหนะโดย Microwave radar sensors

ข้อดี

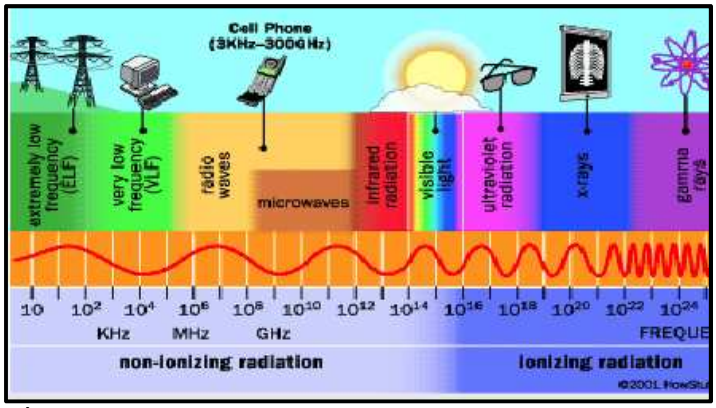
- ไม่ไวต่อสภาพอากาศสำหรับการใช้งานตรวจจับในช่วงสั้น
- สามารถวัดความเร็วได้โดยตรง
- สามารถใช้ตรวจวัดการจราจรได้หลายช่องทางในเวลาเดียวกัน

ข้อเสีย

- Continuous Wave Doppler Sensors ไม่สามารถตรวจวัดรถที่จอดนิ่งอยู่กับที่

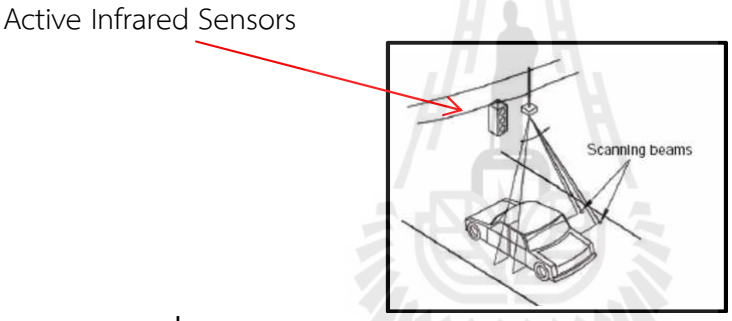
3. Active infrared sensors (laser sensors)

คลื่นอินฟราเรด (Infrared) เป็นคลื่นแม่เหล็กไฟฟ้าที่มีความถี่อยู่ในช่วง 1011 Hz –1014 Hz ซึ่งมีความถี่สูงกว่าคลื่นไมโครเวฟเครื่องตรวจวัดการจราจรแบบ Active Infrared Sensors

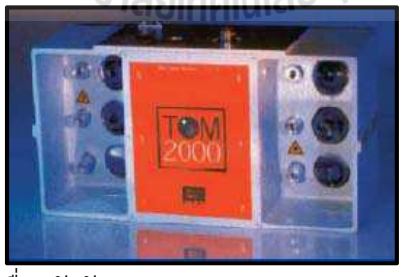


รูปที่.2.19 การตรวจวัดยานพาหนะโดยใช้ Active infrared sensors

เครื่องตรวจวัดการจราจรแบบ Active Infrared Sensors มีลักษณะการทำงานคล้ายๆ กับการทำงานของเครื่องตรวจวัดการจราจรแบบ Microwave Radar Sensors คือมีการปล่อยคลื่นอินฟราเรดออกมาเพื่อตรวจจับรถยนต์ที่วิ่งผ่านแนวสัญญาณดังรูป



รูปที่.2.20 การตรวจวัดการจราจรแบบ Microwave Radar Sensors



รูปที่.2.21 เครื่องตัววัดจราจร แบบ Active Infrared Sensors

หลักการทำงาน

เป็นอุปกรณ์ลำแสง 2 ชุดกวาดผ่านช่องจราจรที่จะทำการตรวจวัดแล้วอุปกรณ์ตรวจจับพลังงานแสงที่เกิดการกระจายเมื่อยานพาหนะเคลื่อนที่ผ่านแล้วแปรสัญญาณส่งผ่านอุปกรณ์ประมวลผลซึ่งเครื่องตรวจวัดสามารถจัดเก็บข้อมูลได้ดังนี้

- ปริมาณการจราจร
- ความเร็วรถ
- การมีรถบริเวณนั้น (Presence)
- ประเภทรถยนต์จำแนกตามความยาวรถแต่ละคัน

ข้อดี

- สัญญาณที่ส่งออกมามีหลายลำทำให้สามารถตรวจจับตำแหน่งรถความเร็ว
- และประเภทของได้อย่างแม่นยำสูง
- สามารถใช้ตรวจวัดการจราจรได้หลายช่องทางในเวลาเดียวกัน

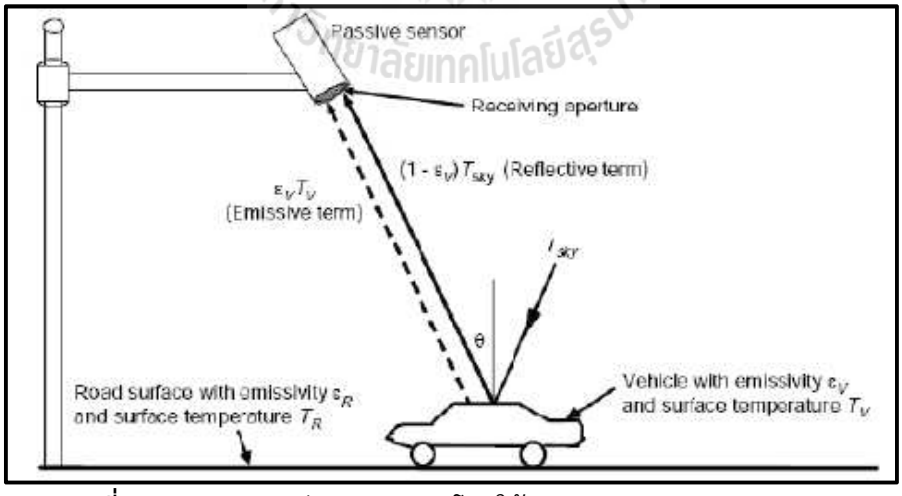
ข้อเสีย

- ประสิทธิภาพในการทำงานลดหากมีหมอกหรือมีหิมะตก
- ต้องมีระยะเวลาในการทำความสะอาดเลนส์เป็นประจำ
- ต้องมีการปิดช่องการจราจรเพื่อทำการติดตั้ง

4. Passive infrared sensors

หลักการทำงาน

เป็นเครื่องมือวัดการจราจรตรวจจับพลังงานที่สะท้อนจากวัตถุเข้าสู่ตัวกล้องที่เกิดจากวัตถุที่เคลื่อนที่ผ่านหรือวัตถุที่อยู่ในบริเวณพื้นที่ตรวจวัด เช่น ผิวจราจรรถยนต์และพลังงานที่เกิดจากบรรยากาศบริเวณนั้นสะท้อนผ่านวัตถุที่เคลื่อนที่เครื่องมือจะตรวจจับการเปลี่ยนแปลงของ Infrared Spectrum แปรข้อมูลส่งผ่านเครื่องประมวลผล



รูปที่.2.22 การตรวจวัดยานพาหนะโดยใช้ Passive infrared sensors

ซึ่งประเภทของ Passive infrared sensors มีดังนี้

- Non-imaging
- Imaging

เครื่องตรวจวัดสามารถให้ข้อมูลได้ดังนี้

- ปริมาณจราจร
- ความเร็วรถ
- การมีรถบริเวณนั้น (Presence)
- แยกประเภทรถโดยใช้ความยาว
- ความยาวของแถวรถที่ติดอยู่บนท้องถนน

ข้อดี

- ระบบตรวจวัดแบบหลายพื้นที่ใช้วัดความเร็ว

ข้อเสีย

- ประสิทธิภาพลดลงช่วงฝนตกหิมะตกหมอกลง
- บางรุ่นไม่สามารถตรวจวัดการมีรถในบริเวณที่กำหนดได้

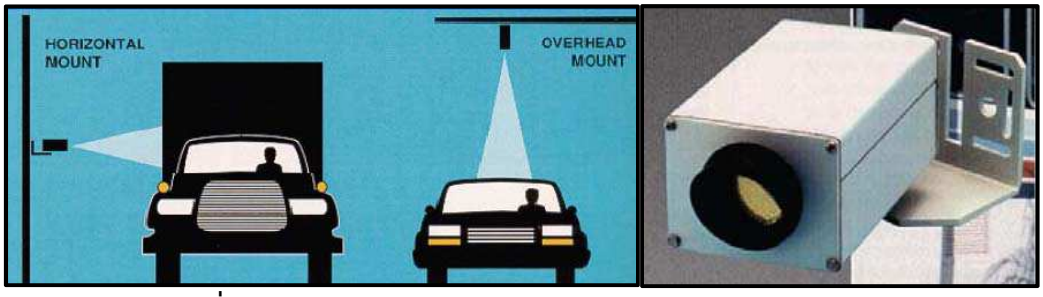
5. Ultrasonic sensors

เริ่มใช้ตั้งแต่ปี 1950s เป็นเครื่องมือตรวจวัดการจราจรที่ใช้การส่งสัญญาณเสียงในระดับความถี่ช่วง 25-50 KHz ใช้ตรวจจับการสะท้อนกลับของคลื่นเสียง

หลักการทำงาน

เป็นการส่งสัญญาณเสียงในระดับความถี่ดังที่กล่าวมาโดยวัดระยะเวลาในการส่งสัญญาณจนถึงระยะเวลานี้เสียงสะท้อนกลับมาซึ่งจะสามารถแสดงผลเป็นระยะห่างระหว่างรถกับสัญญาณเครื่องตรวจวัดสามารถให้ข้อมูลได้ดังนี้

- ปริมาณจราจร
- ความเร็วรถ
- การมีรถบริเวณนั้น (Presence)
- แยกประเภทรถโดยใช้ความยาว



รูปที่.2.23 การตรวจวัดยานพาหนะโดยใช้ Ultrasonic sensors

ข้อดี

- ตรวจวัดได้หลายช่องจราจรพร้อมๆกัน
- สามารถตรวจวัดรถที่มีความสูงเกินกำหนดได้
- ใช้งานแพร่หลายในประเทศญี่ปุ่น

ข้อเสีย

- อุณหภูมิและลมมีผลต่อประสิทธิภาพ
- คลื่นช่วงกว้างส่งผลต่อความถูกต้องในการวัด occupancy บน freeway

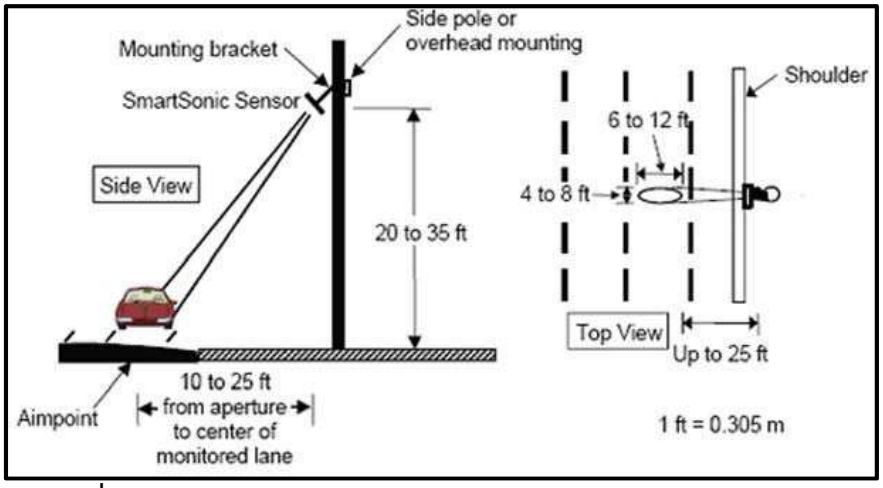
6. Passive acoustic sensors

เป็นเครื่องมือตรวจวัดการจราจรที่ใช้พลังงานเสียง

หลักการทำงาน

ทำงานโดยไม่ต้องมีการส่งสัญญาณเสียงออกไป แต่จะต้องอาศัยการตรวจจับเสียงที่เกิดขึ้นจากยานพาหนะที่เคลื่อนที่ผ่าน เช่น เสียงเครื่องยนต์เสียงล้อรถเมื่อมีเสียงเกิดขึ้นในพื้นที่การตรวจวัดซึ่งจะสามารถจำกัดเป็นช่องทางจราจรได้ โดยเสียงที่เกิดนอกเขตพื้นที่ตรวจวัดจะไม่ส่งผลต่ออุปกรณ์การวัดจากระดับเสียงที่ตรวจวัดได้จะถูกแปรสัญญาณโดยเครื่องประมวลผลได้ข้อมูลดังนี้

- ปริมาณจราจร
- ความเร็วรถ
- การมีรถบริเวณนั้น (Presence)
- การผ่านของรถบริเวณนั้น (Passage)



รูปที่.2.24 การตรวจวัดยานพาหนะโดยใช้ Passive acoustic sensors

ข้อดี

- เป็นเทคโนโลยีแบบ passive
- ไม่ไวต่อสภาพฝน
- บางรุ่นสามารถใช้ตรวจวัดได้หลายช่องจราจรพร้อมๆกัน

ข้อเสีย

- ความถูกต้องของการนับรถลดลงที่อุณหภูมิต่ำ
- บางรุ่นไม่เหมาะกับสภาพการจราจรติดขัด

2.9.3 เทคโนโลยีการตรวจวัดการจราจรที่ใช้ในประเทศไทย

1. Inductive loop detector: ทล, กทม, สนข
2. Video image processor: ทล, สนข, กทพ, ทช
3. Magnetic sensor: ทล
4. Microwave radar: สนข (ระหว่างดำเนินการ)
5. Ultrasonic sensor: สตช (ระหว่างดำเนินการ)

2.10 ไมโครคอนโทรเลอร์ ARM7

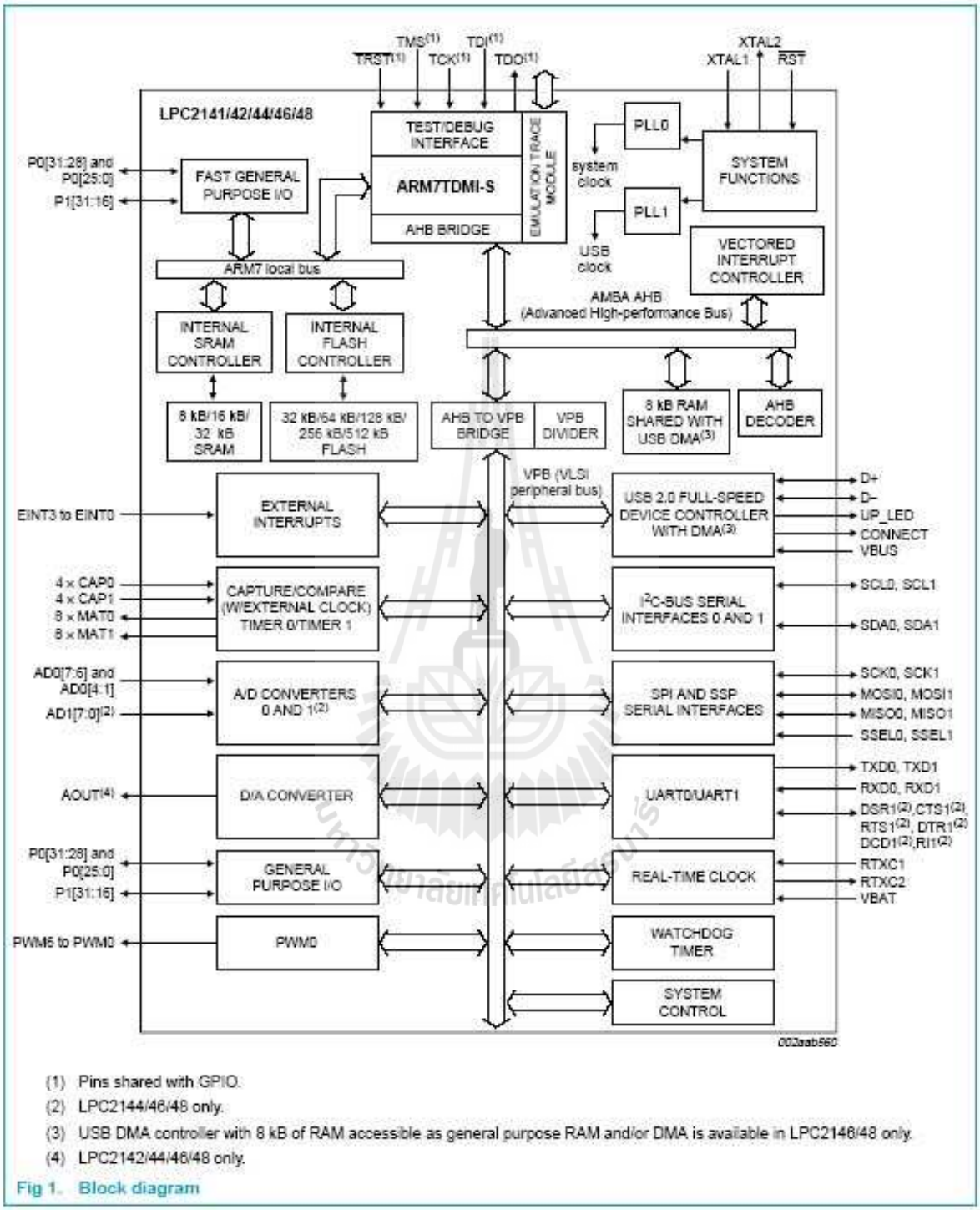
2.10.1 ไมโครคอนโทรลเลอร์ ARM7 Philips LCP2148

ความสามารถของไมโครคอนโทรลเลอร์ Philips LCP2148 มีดังนี้

- ไมโครคอนโทรลเลอร์ขนาด 16/32 บิต ARM7TDMI-S มี LQFP 64 ขา
- หน่วยความจำ Static RAM มีขนาด 40kB
- หน่วยความจำ Flash Program Memory มีขนาด 512kB อยู่ภายในชิปที่สามารถ ลบ/เขียนซ้ำได้ถึง 10000 ครั้ง โปรแกรมชิปสามารถได้ทันทีผ่าน In-System Programming (ISP) และ In-Application Programming (IAP) โดยใช้ซอฟต์แวร์ boot-loader ที่อยู่ภายในชิปตัวควบคุมอุปกรณ์ USB 2.0 Full-speed โดยมี ARM สำหรับ endpoint ขนาด 8 kB สำหรับการติดต่อแบบ DMAวงจรแปลงแอนะลอกเป็นดิจิตอลความละเอียด 10 บิตจำนวน 2 ชุด ที่รับอินพุตได้ถึง 14 อินพุต โดยมีเวลาในการแปลงค่าต่ำถึง 2.44 μsวงจรแปลงดิจิตอลเป็นแอนะลอกความละเอียด 10 บิต 1 ตัววงจรไทเมอร์ขนาด 32 บิต 2 ชุด(มี 4 capture และ 4 compare channel)PWM (Pulse width modulation) 6 เอาต์พุทโมดูลนาฬิกาเวลาจริง (Real Time Clock) ที่สามารถต่อกับคริสตอลความถี่ 32 kHz และแบตเตอรี่ภายนอกได้ และวอชดอกซ์ (Watchdog)
- วงจรสื่อสารอนุกรม UART(16C550) จำนวน 2 ชุดวงจรสื่อสารอนุกรม I^2C ความเร็วสูง (400Kbits/s)วงจรสื่อสารอนุกรม SPI และ SSP มีวงจร Phase lock loop ภายในเพื่อคูณค่าให้สัญญาณนาฬิกาภายในทำงานที่ความถี่สูงสุด 60 MHz Vectored Interrupt Controller ที่สามารถกำหนดลำดับความสำคัญ และกำหนดแอดเดรสของเวกเตอร์ได้ใช้กับแหล่งจ่ายไฟชุดเดียวขนาด 3.0 V และ 3.6V (3.3±10%)
- มี I/O pin อเนกประสงค์ที่สามารถใช้กับระดับแรงดัน 5 V ได้สูงสุด 45 ขาโดยสามารถจัดเป็นขาอินเตอร์รัปต์จากภายนอกได้สูงสุด 21 ขามีโหมดประหยัดพลังงาน 2 โหมดได้แก่ Idle และ Power-down

มหาวิทยาลัยเทคโนโลยีสุรนารี

Block Diagram LPC2148



(1) Pins shared with GPIO.
(2) LPC2144/46/48 only.
(3) USB DMA controller with 8 kB of RAM accessible as general purpose RAM and/or DMA is available in LPC2148/48 only.
(4) LPC2142/44/46/48 only.

Fig 1. Block diagram

รูปที่.2.25 Block Diagram LPC2148

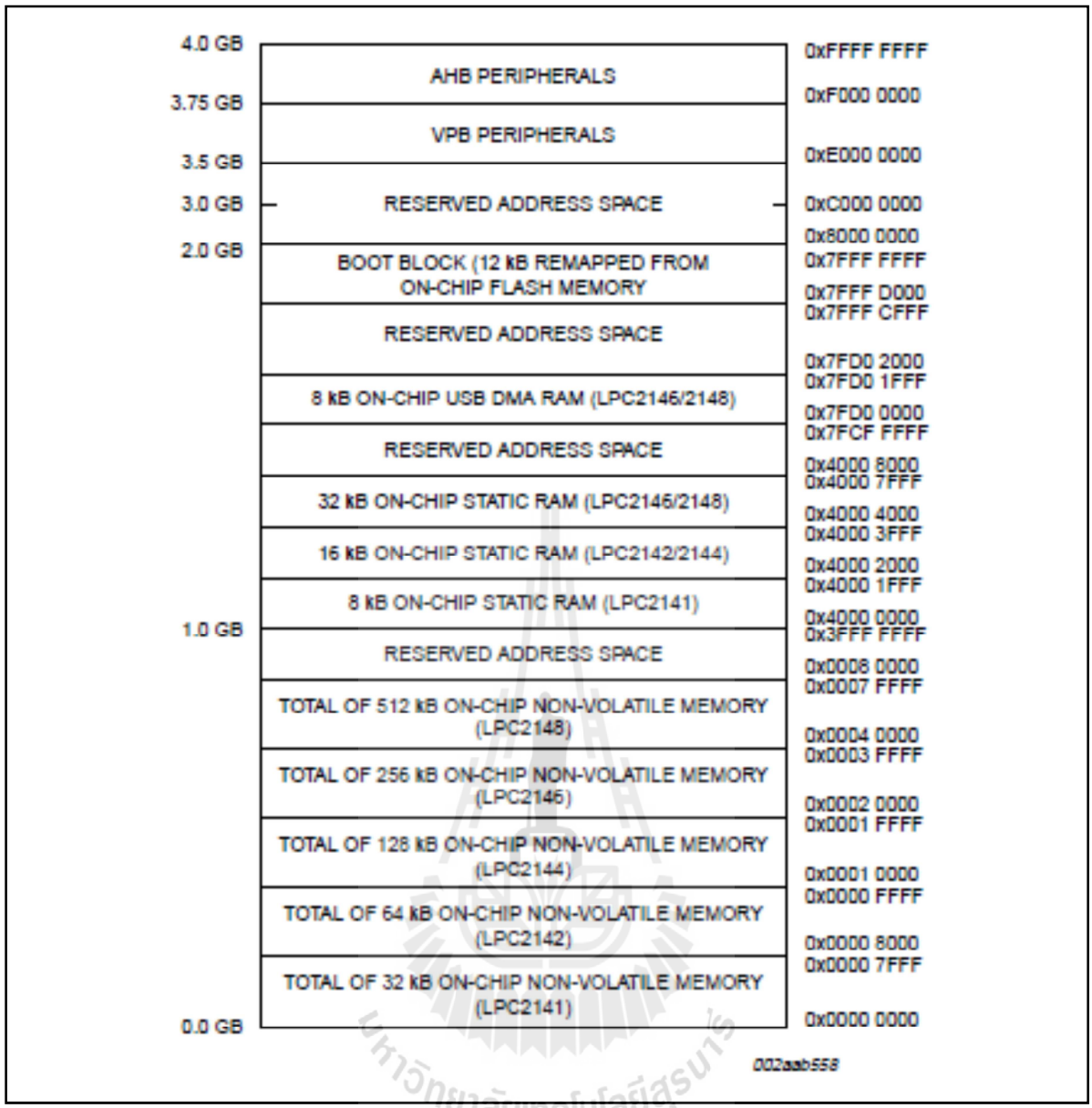
จากรูปที่.2.25 เป็นไมโครโปรเซสเซอร์ ARM7-TDMI-S ซึ่งเป็นปัจจัยหลัก ด้านซ้ายมือเป็นส่วนของ AMR7 Local Bus ที่ใช้ในการติดต่อกับหน่วยความจำแบบ Flash ที่ใช้เก็บโปรแกรมและหน่วยความจำ SRAM ที่ใช้เก็บข้อมูล ส่วนที่ใช้ในการติดต่อกับหน่วยความจำภายนอกมีการติดต่อผ่านบัส AMBA AHB (Advanced High-performance Bus) ซึ่งใน LPC2148 ไม่สามารถต่อกับหน่วยความจำภายนอกได้

ในการติดต่อกับอุปกรณ์ที่เกี่ยวข้อง เช่น GPIO, I^2C, SPI, UART ฯลฯ จะติดต่อผ่านบัส VPB (VLSI peripheral BUS) ซึ่ง VPB บัสต่อกับ AHB บัส ผ่าน AHB to VPB Bridge โดยสามารถปรับลดค่าความถี่ของ VPB บัสให้ทำงานช้ากว่าความถี่ของซีพียูได้เพื่อให้ทำงานร่วมกับอุปกรณ์เสริมต่างๆ ที่มีความเร็วต่ำกว่าได้

การจัดหน่วยความจำของ LPC2148

เนื่องจาก ARM7 เป็นซีพียูขนาด 32 บิต ที่มีขาแอดเดรสต่อกับหน่วยความจำ จำนวน 32 ขา ทำให้สามารถอ้างถึงหน่วยความจำได้ 4 GB (2^{32} = 4GB) อุปกรณ์หลักของ ARM7TDMI จะมีสถาปัตยกรรมแบบ Von Neumann ที่ใช้บัสขนาด 32 บิต ชุดเดียวกันสำหรับตัวคำสั่งของโปรแกรม และข้อมูล โดยมีแค่คำสั่ง load, store, swap เท่านั้น ที่ใช้การเรียกข้อมูลที่เก็บในหน่วยความจำ การติดต่อกับคอมพิวเตอร์อินพุต หรือ เอาต์พุตก็จะใช้คำสั่งเดียวกันกับการใช้คำสั่งจัดการเกี่ยวกับหน่วยความจำในไมโครคอนโทรลเลอร์ LPC2148 ได้จัดสรรหน่วยความจำดังรูป 2.26 เมื่อซีพียูถูกรีเซตจะเริ่มต้นทำงานที่หน่วยความจำ 0x000 0000 จากหน่วยความจำทั้งหมด 4GB ได้แบ่งออกเป็น 4 ส่วน ดังนี้

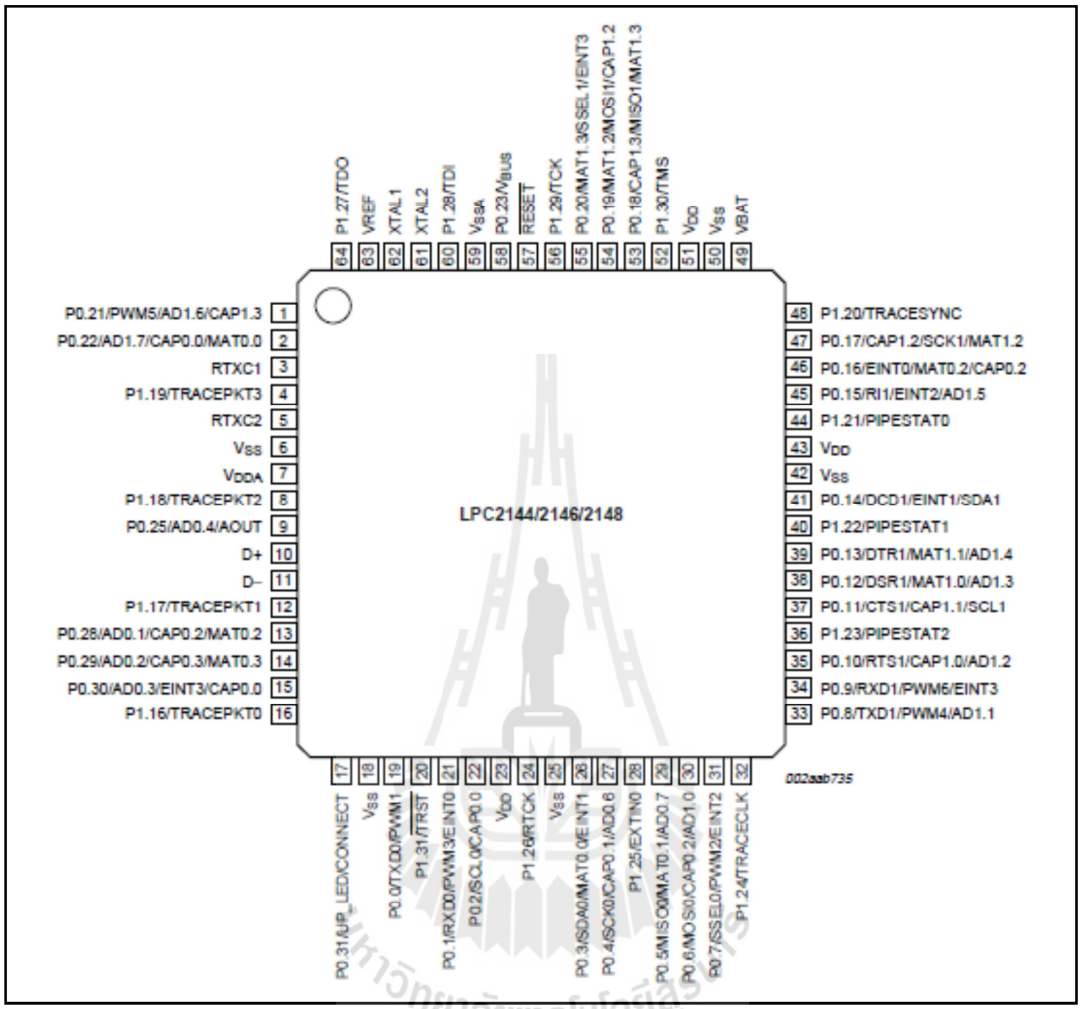
- 1 GB แรก (แอดเดรส 0x0000 0000 – 0x3FFF FFFF) จัดเป็นส่วนของหน่วยความจำสำหรับเก็บโปรแกรม คือ หน่วยความจำ Flash memory ขนาด 512kB ซึ่งมีแอดเดรส 0x0000 0000 – 0x0007 FFFF
- หน่วยความจำในช่วง 1 GB - 2 GB (แอดเดรส 0x4000 0000 – 0x7FFF FFFF) จัดเป็นส่วนของหน่วยความจำ ARM จะเป็น SRAM ขนาด 40kB โดยแบ่งออกเป็น 2 ส่วน
- ส่วนที่ 1 คือ 32kB แรกอยู่ที่แอดเดรส 0x4000 0000 – 0x4000 7FFFF
- ส่วนที่ 2 คือ 8kB เป็นหน่วยความจำใช้สำหรับ USB โดยมีแอดเดรส
- 0x7FD0 0000- 0x7FD0 1FFF ในกรณีที่ไม่ใช้การติดต่อกับ USB สามารถนำหน่วยความจำ ARM ส่วนนี้มาใช้เป็น ARM สำหรับงานทั่วไปได้ หน่วยความจำที่ใกล้ 2 GB จะเป็นส่วนของ Boot Block ขนาด 12 kB ซึ่งเป็นโปรแกรมที่ทำงานเพื่อเขียนโปรแกรมลงในหน่วยความจำ Flash
- หน่วยความจำ 2GB - 3 GB (แอดเดรส 0x8000 0000 – 0xDFFF FFFF) สงวนไว้สำหรับต่อกับหน่วยความจำภายนอก ซึ่งไม่ได้ใช้งาน
- หน่วยความจำ 2GB - 3GB จะเป็นพื้นที่สำหรับติดต่อกับอุปกรณ์อื่นๆ ที่อยู่ในชิป โดยแบ่งเป็น อุปกรณ์ที่ต่อกับ VPB บัส จะติดต่อกับหน่วยความจำช่วง
- 0xE000 00000 – 0xEFFF FFFF ถ้าเป็นอุปกรณ์ที่ติดต่อกับผ่านทาง AHB จะมีช่วงแอดเดรส 0xF000 0000 – 0xFFFF FFFF ไม่สามารถติดต่อกับหน่วยความจำภายนอก จึงไม่ได้ใช้หน่วยความจำส่วนนี้



รูปที่.2.26 memory map LPC 2148

การจัดขาของ LPC2148

มีจำนวนขาทั้งหมด 64 ขา โดยมีการจัดขาแสดงดังรูปที่.2.27 ดังนี้



รูปที่.2.27 LPC2148 pinning

หลังจากการรีเซตขาพอร์ตทั้งหมดจะถูกกำหนดให้ทำหน้าที่เป็นอินพุตขาแต่ละขา จะมีการทำงาน เช่นขาที่ 14 สามารถทำหน้าที่ได้ 4 หน้าที่ คือ

- เป็น P0.29/AD0.2/CAP0.3/MAT0.3
- ถ้าเป็นอินพุตเอาต์พุตจะเรียกว่า General Purpose Input Output : GPIO ก็คือขา P0.23
- ถ้าใช้วงจรแปลงแอนะลอกเป็นดิจิตอลขานี้คือ AD0.2 เป็นขาอินพุตสำหรับสัญญาณแอนะลอก ADC0 อินพุตสอง
- ถ้าใช้งาน Time0 ขานี้จะเป็น CAP0.3 คือ Capture input for timer 0, Channel3 หรือ MAT0.3 ซึ่งเป็น Match output for timer0, Channel3

2.10.2 ฮาร์ดแวร์ และซอฟต์แวร์ของไมโคคอนโทรเลอร์

ฮาร์ดแวร์ และซอฟต์แวร์ที่ใช้ในการเขียนไมโครคอนโทรลเลอร์ Philips LPC 2148 เป็นไมโครคอนโทรลเลอร์ตระกูล ARM7TDMI-S ฮาร์ดแวร์ที่ใช้ประกอบด้วย แผงวงจร CP-JR ARM7 USB-LPC2148 EXP บริษัท อีทีที จำกัด ส่วนของซอฟต์แวร์คอมไพเลอร์เพื่อแปลภาษาซีเป็นภาษาเครื่องของ LPC2148 จะใช้ชุดพัฒนา RealView Microcontroller Development Kit Version 3.02a โดยใช้คอมไพเลอร์ CARMของบริษัท Keill เป็นรุ่น Evaluation ที่อนุญาตให้ดาวน์โหลดมาใช้ทดสอบโปรแกรมฟรี โดยมีข้อจำกัดว่าโปรแกรมที่แปลเป็นภาษาเครื่องแล้ว จะมีขนาดไม่เกิน 16kB หลังจากคอมไพล์เป็นภาษาเครื่องจะทำการลงโปรแกรมชิปไมโครคอนโทรลเลอร์ด้วยโปรแกรม LPC2000 Flash Utility ของบริษัท Philips ซึ่งอนุญาตให้ใช้งานได้ฟรี

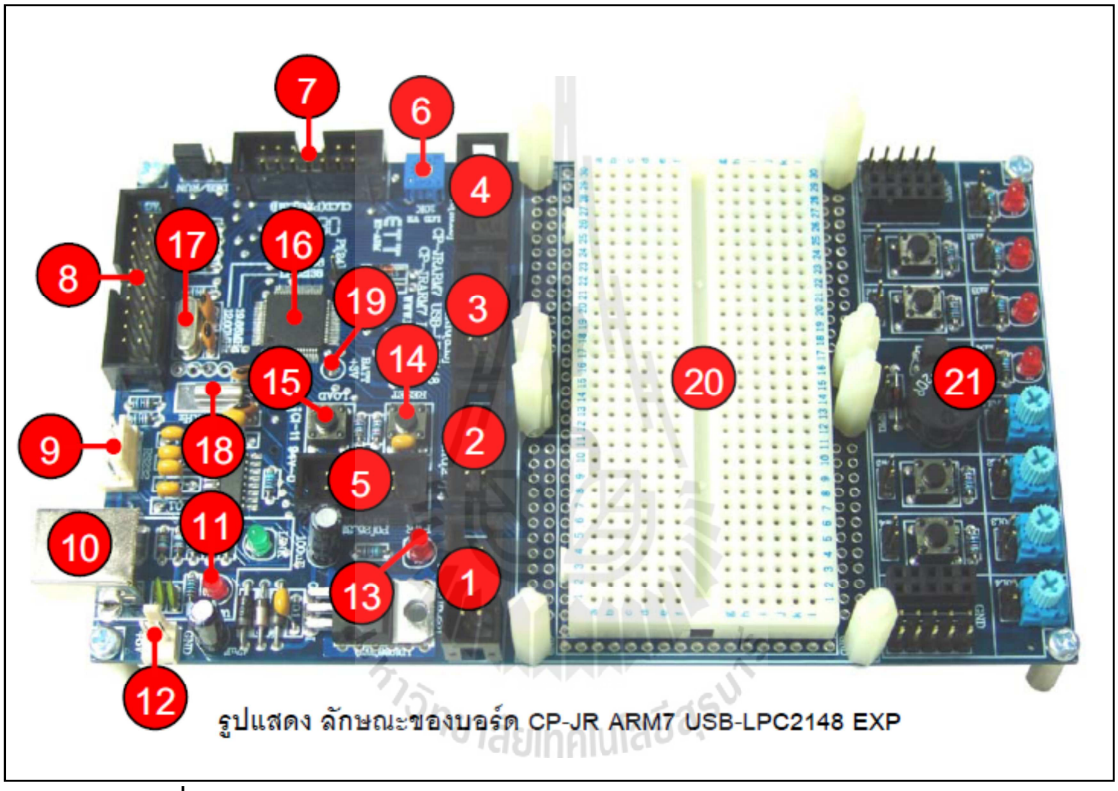
คุณสมบัติของบอร์ดไมโครคอนโทรเลอร์ CP-JR ARM7 USB-LPC2148

1. ใช้ MCU ตระกูล ARM7TDMI-S เบอร์ LPC2148 ของ Philips ซึ่งเป็น MCU ขนาด 16/32-Bit
2. ใช้ Crystal 12.00MHz โดย MCU สามารถประมวลผลด้วยความเร็วสูงสุดที่ 60MHz เมื่อใช้งานร่วมกับ Phase-Locked Loop (PLL) ภายในตัว MCU เอง
3. รองรับการโปรแกรมแบบ In-System Programming (ISP) และ In-Application Programming (IAP) ผ่านทาง On-Chip Boot-Loader Software ทางพอร์ต UART0 (RS232)
4. Power Supply ใช้แรงดันไฟฟ้า +5VDC โดยใช้ขั้วต่อแบบ CPA-2PIN จากภายนอก หรือ ใช้พลังงานจาก USB Port ได้ (ในกรณีใช้กระแสไม่เกิน 500mA)
5. ภายใน MCU มีหน่วยความจำโปรแกรมแบบ Flash ขนาด 512KB, Static RAM ขนาด 40KB
6. มีวงจร USB มาตรฐาน 2.0 แบบ Full Speed ภายในตัว (USB Function มี 32 End Point)
7. จำนวน GPIO สูงสุดถึง 47 I/O Pins สามารถเชื่อมต่อกับระบบ I/O ที่เป็นสัญญาณ 5V ได้ ซึ่งขาสัญญาณ GPIO จะมีการใช้งานร่วมกันของ Function อื่นๆอีกดังนี้วงจรสื่อสารอนุกรมแบบ SPI จำนวน 2 ช่อง และ วงจรสื่อสารอนุกรมแบบ I2C จำนวน 2 ช่องวงจร ADC ขนาด 10 Bit จำนวน 14 ชุด และ วงจร DAC ขนาด 10 Bit จำนวน 1 ชุดวงจร UART แบบ Full-Duplex จำนวน 2 ช่อง คือ UART-0 มาตรฐาน 4 Pin ETT เป็นสัญญาณระดับ RS232 Level และ UART-1 เป็นสัญญาณระดับ TTL LevelTimer 32-bit จำนวน 2 ช่อง (4 Input Capture / 4 Output Compare), 6-Channel PWMOutput, Watchdog Timer และ Real Time Clock
8. มีวงจรเชื่อมต่อกับ Character LCD โดยใช้วงจรการเชื่อมต่อแบบ 4 บิต จาก GPIO1[25..31] พร้อมวงจรปรับความสว่างหน้าจอ
9. มีวงจรเชื่อมต่อกับ JTAG ARM ขนาด 20 Pin มาตรฐาน เพื่อทำการ Debug แบบ Real Time ได้
10. มีวงจรทดลองขั้นพื้นฐานสำหรับสนับสนุนการใช้งานและทดลองเรียนรู้ ขั้นพื้นฐานอย่างครบถ้วน จัดเตรียมไว้ภายในบอร์ด (ติดตั้งไว้เฉพาะรุ่น CP-JR ARM7 USB-LPC2148 EXP) ซึ่ง ได้แก่
    - LED Output แบบ Sink Current สำหรับแสดงสถานะของ Output จำนวน 4 ชุด
    - Push Button Switch แบบ Active Logic “0” สำหรับทดสอบ Input Logic จำนวน 4 ชุด
    - Volume ปรับค่าแรงดัน 0-3.3V สำหรับทดสอบการทำงานของ ADC จำนวน 4 ชุด

- ชุดกำเนิดสัญญาณเสียง Mini Speaker สำหรับทดสอบการเสียงแบบต่างๆ จำนวน 1 ชุด
- แผงต่อวงจร Project Board รุ่น AD-100 ขนาด 360 จุด สำหรับเป็นพื้นที่ต่อทดลองวงจรขนาด
- เล็กๆ เพื่อใช้งานร่วมกับ CPU ได้อย่างอิสระ
- จุดต่อแหล่งจ่ายไฟ +3.3V และ GND สำหรับเชื่อมต่อไปยังวงจรภายนอกอื่นๆ

11.ทนอุณหภูมิใช้งานระหว่าง -40 to +85°C

โครงสร้างบอร์ดไมโครคอนโทรเลอร์ CP-JR ARM7 USB-LPC2148



รูปที่.2.28 ลักษณะโครงสร้างของบอร์ด CP-JR ARM7 USB-LPC2148 / EXP

จากรูปที่.2.28 รายละเอียดของการจัดตำแหน่งส่วนต่าง ๆ ของบอร์ดไมโครคอนโทลเลอร์ มีดังนี้

หมายเลข 1 คือ ขั้วต่อ Port1[16..23] จำนวน 8 บิต

หมายเลข 2 คือ ขั้วต่อ Port0[2..7] จำนวน 6 บิต

หมายเลข 3 คือ ขั้วต่อ Port0[8..15] จำนวน 8 บิต

หมายเลข 4 คือ ขั้วต่อ Port0[16..23] จำนวน 8 บิต

หมายเลข 5 คือ ขั้วต่อ Port0[25..31] จำนวน 7 บิต

หมายเลข 6 คือ ตัวต้านทานสำหรับปรับค่าความสว่าง (Contrast) ของหน้าจอ LCD

หมายเลข 7 คือ ขั้วต่อ Character LCD โดยใช้สัญญาณ Port1[25..31] ในการเชื่อมต่อ

หมายเลข 8 คือ ขั้วต่อ JTAG โดยใช้สัญญาณ Port1[26..31] และ Reset ของ CPU

หมายเลข 9 คือ ขั้วต่อ RS232 สำหรับใช้งาน และ Download Hex File ให้ CPU

หมายเลข 10 คือ ขั้วต่อ USB สำหรับเชื่อมต่อกับ USB Hub รุ่น 2.0

หมายเลข 11 คือ LED แสดงสถานะ Power จาก USB และ สถานของการเชื่อมต่อกับ USB

หมายเลข 12 คือ ขั้วต่อ Power ขนาด +5VDC และ GND เพื่อจ่ายให้กับบอร์ด

หมายเลข 13 คือ LED แสดงสถานะของแหล่งจ่ายไฟ Power ของบอร์ด

หมายเลข 14 คือ Switch RESET สำหรับสั่ง Reset การทำงานของ CPU

หมายเลข 15 คือ Switch LOAD ใช้ร่วมกับ Switch RESET เพื่อ Download Hex ให้ CPU

หมายเลข 16 คือ CPU เบอร์ LPC2148 ของ Philips ซึ่งเป็น CPU ประจำบอร์ด

หมายเลข 17 คือ Crystal 12.00 MHz สำหรับป้อนให้เป็นสัญญาณนาฬิกาของ LPC2148

หมายเลข 18 คือ Crystal 32.768 KHz สำหรับ Real Time Clock (RTC) ในตัวของ LPC2148

หมายเลข 19 คือ จุดเชื่อมต่อ ลังถ่าน Battery ขนาด +3V (อยู่ด้านใต้บอร์ด) สำหรับต่อให้กับRTC เพื่อเก็บรักษาค่าเวลาของ RTC ในขณะที่ไม่ได้จ่ายไฟเลี้ยงให้กับบอร์ด

หมายเลข 20 คือ แผง Project Board รุ่น AD-100 ขนาด 360 จุด สำหรับต่อวงจร (มีติดตั้งไว้เฉพาะในรุ่น CP-JR ARM7 USB-LPC2148 EXP)

หมายเลข 21 คือ ส่วนของวงจร I/O พื้นฐาน สำหรับใช้ทดสอบการทำงานของ Function ต่างๆของ CPU (มีติดตั้งไว้เฉพาะในรุ่น CP-JR ARM7 USB-LPC2148 EXP) โดยมีรายละเอียดดังนี้คือ

LED สำหรับแสดงผลการทำงานของ Output แบบ Sink Current มีทั้งหมด 4 ชุด

Push Button Switch สำหรับกำเนิด Logic เพื่อทดสอบการทำงาน Input มีทั้งหมด 4 ชุด

Volume สำหรับปรับค่าแรงดัน 0-3.3V เพื่อใช้ทดสอบการทำงานของ A/D มีทั้งหมด 4 ชุด

Mini Speaker สำหรับใช้กำเนิดเสียง เช่น Beep จำนวน 1 ชุด

จุดต่อแหล่งจ่ายไฟ +3.3V และ GND

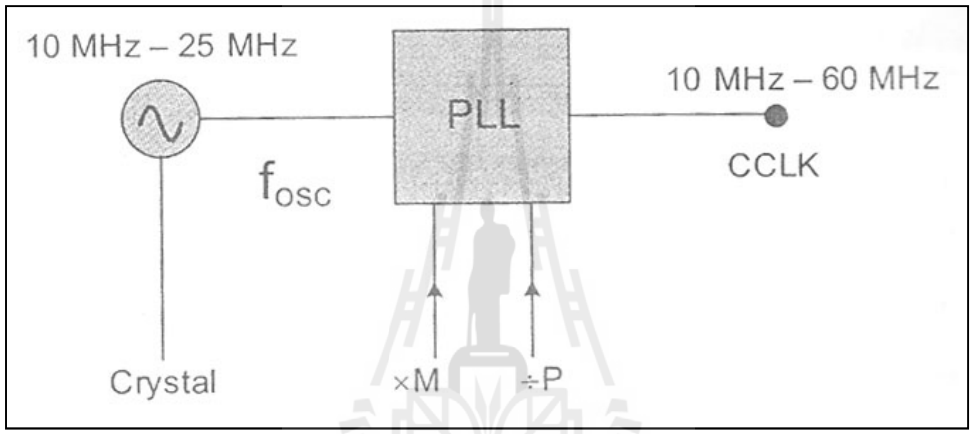
2.10.3 การกำหนดค่าเริ่มต้นให้กับไมโครคอนโทรลเลอร์ ARM7

การกำหนดค่าเริ่มต้นให้ก่อนโดยค่าที่ต้องการกำหนด คือ ส่วนของหน่วยความจำ RAM ที่ใช้เป็น stack pointer การกำหนดค่าของ stack pointer ต้องเขียนคำสั่งเป็นโปรแกรมภาษาแอสแซมบลีภายในของไมโครคอนโทรลเลอร์ ARM มีวงจร Phase Lock Loop : PLL สำหรับคูณค่าความถี่ของสัญญาณนาฬิกาจากภายนอก เพื่อให้ตัวไมโครคอนโทรลเลอร์สามารถทำงานที่ความถี่สูงๆ ได้โดยการกำหนดค่าตัวคูณความถี่ สำหรับ LPC2148 ภายในจะมีวงจร PLL อยู่สองวงจรโดยตั้งชื่อเป็น PLL0 และ PLL1 โดย PLL0 ใช้คูณค่าความถี่ของสัญญาณนาฬิกาภายนอกให้กับซีพียู ARM 7 และ PLL1 ที่ทำหน้าที่คูณวามถี่ให้กับวงจรส่วนของ USB โดยต้องคูณค่าให้ได้ความถี่ 48 MHz หลังจากที่ไมโครคอนโทรลเลอร์ทำงานที่ความถี่สูงแล้วอุปกรณ์ประกอบ(Peripheral)ต่างๆ ที่อยู่ภายนอกไมโครคอนโทรลเลอร์ เช่น UART , I^2C ฯลฯ จะทำงานไม่ทัน ดังนั้นภายในไมโครคอนโทรลเลอร์ ARM7 จะมีวงจรหารความถี่เพื่อลดความถี่ที่ป้อนให้อุปกรณ์ต่างๆหน่วยความจำแบบ Flash ภายในของ LPC2148 มีการทำงานช้ากว่าตัวซีพียู ดังนั้น

ภายในของ LPC2148 จะมีวงจร Memory Accelerator Module (MAM) เพื่อทำหน้าที่เร่งความเร็วของการติดต่อกับหน่วยความจำแฟลช(Flash)ภายในชิป

วงจร Phase Lock Loop : PLL

วงจรคริสตัลออสซิลเลเตอร์ภายในของไมโครคอนโทรลเลอร์ตระกูล LPC2000 ใช้ได้กับคริสตัลค่าความถี่ 1 MHz – 30MHz เอาต์พุตของวงจรเรียกว่า f_{osc} ส่วนความถี่สัญญาณนาฬิกาของไมโครคอนโทรลเลอร์จะมีชื่อเรียกว่า CCLK โดยปกติถ้าไม่ได้เปิดใช้วงจร PLL ไมโครคอนโทรลเลอร์ ARM7 จะนำความถี่ f_{osc} ไปใช้กับไมโครคอนโทรลเลอร์ ARM7 เช่นเดียวกัน (กรณีนี้ค่าของ f_{osc} และ CCLK จะมีค่าเท่ากัน)



รูปที่.2.29 ผังวงจร PLL ภายในไมโครคอนโทรลเลอร์ ARM 7

วงจร PLL จะรับค่าความถี่สัญญาณนาฬิกาจากคริสตัลออสซิเลเตอร์ที่ถูกควบคุมค่าโดยคริสตัลภายนอก แล้วนำมาคูณด้วยค่าคงที่ M ให้เป็นความถี่ 10 MHz – 60 MHz โดยใช้วงจร Current Controlled Oscillator (CCO) ทำหน้าที่คูณความถี่ ค่าตัวคูณ M จะมีค่าตั้งแต่ 1 ถึง 32 วงจร CCO ทำงานในช่วงความถี่ 156 MHz - 320MHz ดั้งนั้นภายในลูปของ CCO จะต้องมีวงจรหารค่าอีกหนึ่งตัวเพื่อให้เอาต์พุตของ PLL สร้างความถี่ให้ได้ค่าตามที่ต้องการ ค่าตัวหาร คือ P กำหนดค่าได้เป็น 2,4,8 หรือ 16 เมื่อไมโครคอนโทรลเลอร์ถูกรีเซต วงจร PLL จะถูกปิดการทำงาน ซึ่งต้องให้ซอฟต์แวร์สั่งเปิดการทำงานของโปรแกรม ตัวโปรแกรมจะต้องกำหนดค่าตัวคูณ M ตัวหาร P และกระตุ้นการทำงานของ PLL และรอให้ PLL ล็อกความถี่ได้ก่อน จึงจะสั่งต่อให้ PLL เป็นสัญญาณนาฬิกาของไมโครคอนโทรลเลอร์ ARM7 ค่าเวลาในการเซต PLL เป็น 10μs เอาต์พุตของ PLL คำนวณได้ดังนี้

$CCLK = M * f_{osc}$ หรือ $CCLK = fcco/(2 * P)$

ค่าความถี่ CCO คำนวณได้ดังนี้

$fcco = CCLK * 2 * P$ หรือ $fcco = f_{osc} * 2 * M * P$

ค่า P เมื่อนำไปคูณแล้ว fcco จะมีเงื่อนไขดังนี้

156MHz < fcco < 320MHz

แผงวงจร CP-JR ARM7 USB-LPC2148 EXPใช้คริสตัลความถี่ 12.00MHz ดังนั้น f_{osc} = 12MHz ต้องการคูณให้ได้ความถี่สูงสุดไม่เกิน 60MHz จะได้ตัวคูณ M = 5 ดังนั้น

CCLK = 5*12.00 = 60.00MHz

ในการกำหนดค่า PLL1 เพื่อกำเนิดสัญญาณนาฬิกาสำหรับวงจร USB จะต้องคำนวณ ค่า M ที่นำไปคูณกับความถี่ของสัญญาณนาฬิกาให้ได้ความถี่ 48 MHz จาก แผงวงจร CP-JR ARM7 USB-LPC2148 EXP จะคำนวณได้ M = 48MHz / f_{osc} = 48MHz / 12MHz =4

การหาค่าของ P ให้ใช้ตามค่าที่ได้จากกรณีของ PLL0 คือให้ค่า P = 2 ได้ fcco =48MHz * 2 *2 = 192MHz ซึ่งตรงตามเงื่อนไขในการกำหนดค่าการทำงานของวงจร PLL จะสั่งผ่านทางรีจิสเตอร์ ที่เกี่ยวข้องกับ PLL ดังแสดงในตารางที่ 2.5

ตารางที่ 2.5 PLL Register

ชื่อทั่วไป	**ความหมาย**	**ติดต่อ**	**ค่าหลังรีเซต**	**ชื่อและแอดเดรสของ PLL0**	**ชื่อและแอดเดรสของ PLL1**
PLLCON	PLL Control Register เก็บค่ารีจิสเตอร์สำหรับการอัพเดทค่า PLL control bit ค่าที่เขียนให้ยังไม่มีผลจนกว่าจะเขียน PLL feed sequence ที่ถูกต้องให้	อ่าน/เขียน	0	PLL0CON 0xE01FC080	PLL1CON 0xE01FC0A0
PLLCFG	PLL Configuration Register เก็บค่าสำหรับอัพเดทค่าของ PLL Configuration ค่าทีเขียนให้ยังไม่มีผลจนกว่าจะเขียน PLL feed sequence	อ่าน/เขียน	0	PLL0CFG 0xE01FC084	PLL1CFG 0xE01FC0A4
PLLSTAT	PLL Configuration อ่านค่าสถานะของ PLL Control และ configuration	อ่าน/เขียน	0	PLL0STAT 0xE01FC088	PLL1STAT 0xE01FC0A8
PLLFEED	PLL Feed Register เป็นการสั่งโหลดค่าที่เก็บใน	อ่าน/เขียน	ไม่มี	PLL0FEED	PLL1FEED

ชื่อทั่วไป	ความหมาย	ติดต่อ	ค่าหลังรีเซต	ชื่อและแอดเดรสของ PLL0	ชื่อและแอดเดรสของ PLL1
	PLLCON และ PLLCFG เพื่อสั่งให้ทำงานตามค่าที่กำหนดให้			0xE01FC08C	0xE01FC0AC

การกำหนดค่าของ P , M กำหนดที่รีจิสเตอร์ PLLCFG ซึ่งแสดงรายละเอียดแต่ละบิตของรีจิสเตอร์ PLLCFG ได้ดังตาราง 2.6

ตารางที่ 2.6 PLLCFG Register

ค่าบิตของ PLLCFG	หน้าที่	ความหมาย	ค่าหลังรีเซต
4:0	MSEL 4:0	ค่าตัวคูณ M ของวงจร PLL 00000 คือ M=1 , 00001 คือ M=2,.....,11110 คือ M =31 และ 11111 คือ M=32	0
6:5	PSEL 1:0	ค่าตัวหาร P ของวงจร PLL 00 คือ P=1 ,01 คือ P=2 10 คือ P=2 , 11 คือ P=4	0
7	สงวนไว้	ห้ามเขียนค่า 1 ให้กับบิตนี้	ไม่ได้กำหนด

เมื่อกำหนดค่าตัวหาร P และตัวคูณ M แล้วคำสั่งให้วงจร PLL ทำงานและต่อ PLL เป็นสัญญาณนาฬิกาหลักได้ที่รีจิสเตอร์ PLLCON ซึ่งมีค่าดังตารางที่ 2.7

ตารางที่ 2.7 PLLCON Register

ค่าบิตของ PLLCON	หน้าที่	ความหมาย	ค่าหลังรีเซต
0	PLLE	PLL Enable สั่งให้ PLL ทำงานโดยเขียนค่าเป็น 1 และต้องเขียนค่า PLLFEED ที่เหมาะสมด้วย	0
1	PLLC	PLL Connect เมื่อ PLLC และ PLLE มีค่าเป็น 1 และหลังจากเขียนค่า PLLFEED ที่เหมาะสมจะเป็นการสั่งให้ต่อ PLL เป็นสัญญาณนาฬิกาหลังของวงจร	0
7:2	สงวนไว้	ห้ามเขียนค่า 1 ให้กับบิตนี้	ไม่ได้กำหนด

ค่าแต่ละบิตของรีจิสเตอร์ PLLSTAT เพื่อดูสถานะของ PLL แสดงได้ในตารางที่ 2.8

ตารางที่ 2.8 ค่าแต่ละบิตของรีจิสเตอร์ PLLSTAT

ค่าบิตของ PLLSTAT	หน้าที่	ความหมาย	ค่าหลังรีเซต
4:0	MSEL 4:0	อ่านค่าว่าปัจจุบันวงจร PLL ใช้ค่าตัวคูณความถี่เท่ากับเท่าไร	0
6:5	PSEL 1:0	อ่านค่าว่าปัจจุบันวงจร PLL ใช้ค่าตัวหารความถี่เท่ากับเท่าไร	0
7	สงวนไว้	ห้ามเขียนค่า 1 ให้กับบิตนี้	ไม่กำหนด
8	PLLE	อ่านค่าว่าปัจจุบันบิต PLLE มีค่าเป็น 0 หรือ 1	0
9	PLLC	อ่านค่าว่าปัจจุบันบิต PLLC มีค่าเป็น 0 หรือ 1	0
10	PLOCK	ถ้าเป็น 0 แสดงว่า PLL ยังไม่สามารถล็อคค่าความถี่ ถ้าเป็น 1 แสดงว่าสามารถล็อคค่าความถี่ตามที่กำหนดได้แล้ว	0
15:11	สงวนไว้	ห้ามเขียนค่า 1 ให้กับบิตนี้	ไม่กำหนด

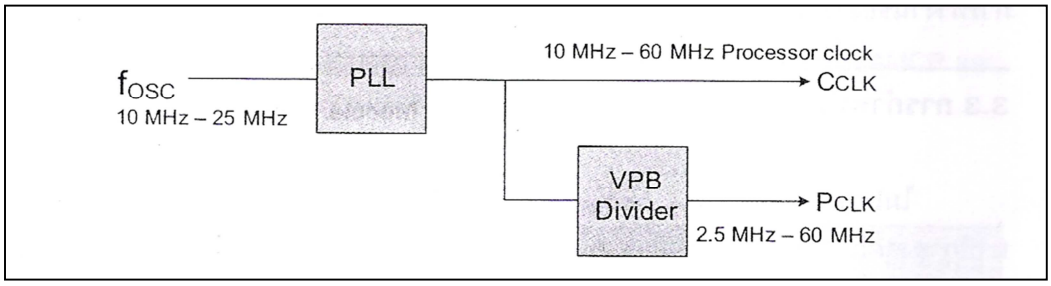
เราสามารถกำหนดโหมดการทำงานของวงจร PLL ได้จากบิต PLLC และ PLLE ของรีจิสเตอร์ PLLCON ได้ดังตารางที่ 2.9

ตารางที่ 2.9 โหมดการทำงานของวงจร PLL

บิต PLLC	บิต PLLE	การทำงานของ PLL
0	0	PLL ไม่ทำงาน ไม่ได้ต่อเข้ากับไมโครคอนโทรเลอร์
0	1	PLL ทำงานแต่ไม่ได้ต่อกับไมโครคอนโทรเลอร์
1	0	เหมือนกับกรณีค่า 00 เพื่อป้องกันไม่ให้ต่อ PLL เข้ากับระบบถ้ายังไม่สั่งให้มันทำงาน
1	1	PLL ทำงาน และต่อเป็นสัญญาณนาฬิกาหลัก

การกำหนดค่าความถี่ให้กับ VLSI Peripheral Bus

เมื่อกำหนดให้กับวงจร PLL ภายในไมโครคอนโทรลเลอร์ ARM7 ทำงานสร้างความถี่สูง และต่อให้เป็นสัญญาณนาฬิกาของโปรเซสเซอร์แล้ว อุปกรณ์ประกอบต่างๆ เช่น UART, GPIO, I^2C ที่ต่อกับบัส VPB จะทำงานไม่ทัน ดังนั้นภายในไมโครคอนโทรลเลอร์ ARM7 จะมีวงจรหารลดความถี่ของ CCLK ให้เป็นความถี่ของ VPB Bus ที่เรียกว่า PCLK โดยบล็อกไดอะแกรมภายในแสดงดังรูป 2.30



รูปที่. 2.30 Block Diagram วงจรกำเนิดสัญญาณภายใน ARM7

รีจิสเตอร์ที่เกี่ยวข้องคือ VPBDIV ที่อยู่แอดเดรส 0xE10FC100 โดยสามารถอ่านและเขียนได้ ค่าของรีจิสเตอร์ VPBDIV จะมีค่า 2 บิตสุดท้าย โดยกำหนดได้ดังตารางที่ 2.10

ตารางที่ 2.10 ค่าแต่ละบิตของรีจิสเตอร์ VPBDIV

ค่าบิตของ VPBDIV	หน้าที่	ความหมาย	ค่าหลังรีเซต
1:0	VPBDIV	VPB bus clock = CCLK/4 VPB bus clock = CCLK VPB bus clock = CCLK/2 สงวนไว้	0
7:2	สงวนไว้	ห้ามเขียนค่า 1 ให้บิตเหล่านี้	0

การกำหนดค่าให้กับ Memory Accelerator Module (MAM)

การกำหนดค่าการทำงาน MAM มีรีจิสเตอร์ที่เกี่ยวข้องสองตัว คือ MAMCR, MAMTIM ดังแสดงในตารางที่ 2.11

ตารางที่ 2.11 MAM Register

แอดเดรส	ชื่อ	ความหมาย	ติดต่อ
0xE01FC000	MAMCR	Memory Accelerator Module Control Register ใช้กำหนดโหมดการทำงานของ MAM	อ่าน/เขียน
0xE01FC004	MAMTIM	Memory Accelerator Module Timing Control ใช้กำหนดจำนวนสัญญาณนาฬิกาที่ใช้ในการติดต่อกับหน่วยความจำ Flash	อ่าน/เขียน

การกำหนดค่าโหมดการทำงานของ MAM ผ่านทาง MAMCR ได้ดังตาราง 2.12

ตารางที่ 2.12 การกำหนดค่าให้กับ MAMCR Register

MAMCR bit	ฟังก์ชัน	ความหมาย
1:0	MAN mode control	00-MAM Disable หยุดการทำงานของ MAM 01-MAM partially enable เปิดการทำงาน MAM บางส่วน 10-MAM fully enable เปิดการทำงาน MAMสมบรูณ์แบบ 11-สงวนไว้
7:2	สงวนไว้	สงวนไว้ ห้ามเขียนค่า 1 ให้บิตเหล่านี้

การกำหนดค่ารีจิสเตอร์ MAMTIM กำหนดได้ดังตารางที่ 2.13

ตารางที่ 2.13 การกำหนดค่าให้กับ MAMTIM Register

MAMTIM bit	ฟังก์ชัน	ความหมาย
2:0	MAN Fetch Cycle timing	000-สงวนไว้ 001-MAM fetch cycles = 1 * CCLK 010-MAM fetch cycles = 2 * CCLK 011-MAM fetch cycles = 3 * CCLK 100-MAM fetch cycles = 4 * CCLK 101-MAM fetch cycles = 5 * CCLK 110-MAM fetch cycles = 6 * CCLK 111-MAM fetch cycles = 7 * CCLK
7:3	สงวนไว้	สงวนไว้ ห้ามเขียนค่า 1 ให้บิตเหล่านี้

ในการกำหนดค่าเริ่มต้นการทำงานของ MAM เริ่มต้นด้วยการหยุดทำงานของ MAM ด้วยการโหลดค่า 0 ให้กับ MAMCR แล้วจึงกำหนดค่าของ MAMTIM ตามที่ต้องการแล้วจึงเปิดการทำงานของ MAM ด้วยการเขียนค่า 1 หรือ 2 ให้กับ MAMCR

ถ้าซีพียูมีความถี่ CCLK น้อยกว่า 20MHz ให้ใช้ MAMTIM = 1*CCLK ถ้า CCLKมีค่าระหว่าง 20 –40 MHz ให้กำหนด MAMTIM = 2*CCLK ถ้าความถี่มากกว่า 40MHz ให้กำหนดค่า MAMTIM = 3*CCLK

2.10.4. การต่อ GPIO เป็นเอาต์พุต

ภายในไมโครคอนโทรลเลอร์ มีพอร์ตอเนกประสงค์ ขนาด 32 บิต ให้ใช้งาน 2 พอร์ต คือ

Port 0 , Port1 โดยให้ Port0 มีขาต่อใช้งานได้ 29 ขา คือ P0.0 –P0.31 โดยไม่มีขา P0.24, P0.26, P0.27 ให้ใช้งาน และขา P0.31 ทำหน้าที่เป็นเอาต์พุตได้อย่างเดียว ส่วน Port1 จะมีต่อให้ใช้งานแค่ 16 ขา คือ P1.16-P1.31 โดยแต่ละขาของพอร์ต P0 และ P1 มีหน้าที่การทำงานได้หลายประเภท ซึ่งต้องกำหนดรีจิสเตอร์ PINSEL

การกำหนดหน้าที่การทำงานของพอร์ต

หลังจากเกิดการรีเซตไมโครคอนโทรลเลอร์ ARM7 วงจรภายในสั่งรีเซตค่ารีจิสเตอร์ PINSEL ทุกตัวกำหนดค่าให้ Port0 ,Port1 ทุกขาเป็น GPIO และให้ทุกขาเป็นอินพุต

ขาแต่ละขาของ Port0,Port1 มีหน้าที่การทำงานได้หลายหน้าที่ โดยกำหนดการทำงานของ Port ที่รีจิสเตอร์ PINSEL0, PINSEL1 และกำหนดหน้าที่การทำงานของ Port1 ที่รีจิสเตอร์ PINSEL2 โดยรีจิสเตอร์แต่ละตัวที่มีแอดเดรสแสดงในตารางที่ 2.14

ตารางที่ 2.14 รีจิสเตอร์ PINSEL0, PINSEL1, PINSEL2

ชื่อ	ความหมาย	การติดต่อ	แอดเดรส
PINSEL0	Pin function select register กำหนดการทำงานของ P0.0-P0.15	อ่าน/เขียน	0xE002 C000
PINSEL1	Pin Function select register 1 กำหนดการทำงานของ P0.16-P0.31	อ่าน/เขียน	0xE002 C004
PINSEL2	Pin Function select register 2 กำหนดการทำงานของ P01.0 –P1.31	อ่าน/เขียน	0xE002 C014

ตารางที่ 2.15 ค่าประจำบิตของรีจิสเตอร์ PINSEL0

PINSEL0	ชื่อขา	การทำงาน เมื่อมีค่า 00	การทำงาน เมื่อมีค่า 01	การทำงาน เมื่อมีค่า 10	การทำงาน เมื่อมีค่า 11	ค่าหลังรีเซต
1:0	P0.0	GPIO Port0.0	TxD0(UART0)	PWM1	Reserved	00
3:2	P0.1	GPIO Port0.1	RxD0(UART0)	PWM3	EINT0	00
5:4	P0.2	GPIO Port0.2	SCL0(I^2C)	Capture 0.0(TIMER0)	Reserved	00
7:6	P0.3	GPIO Port0.3	SDA0(I^2C)	Match0.0 (TIMER0)	EINT1	00
9:8	P0.4	GPIO Port0.4	SCK0(SPI0)	Capture 0.1(TIMER0)	AD0.6	00
11:10	P0.5	GPIO Port0.5	MISO0(SPI0)	Match0.1 (TIMER0)	AD0.7	00

PINSEL0	ชื่อขา	การทำงาน เมื่อมีค่า 00	การทำงาน เมื่อมีค่า 01	การทำงาน เมื่อมีค่า 10	การทำงาน เมื่อมีค่า 11	ค่าหลัง รีเซต
13:12	P0.6	GPIO Port0.6	MISI0(SPI0)	Capture 0.2(TIMER0)	AD1.0	00
15:14	P0.7	GPIO Port0.7	SSEL0(SPI0)	PWM2	EINT2	00
17:16	P0.8	GPIO Port0.8	TxD1 (UART1)	PWM4	AD1.1	00
19:18	P0.9	GPIO Port0.9	RxD1 (UART1)	PWM6	EINT3	00
21:20	P0.10	GPIO Port0.10	RTS1(UART1)	Capture 1.0(TIMER1)	AD1.2	00
23:22	P0.11	GPIO Port0.11	CTS1(UART1)	Capture 1.1(TIMER1)	SCL1(I^2C1)	00
25:24	P0.12	GPIO Port0.12	DSR1(UART1)	Match1.0 (TIMER1)	AD1.3	00
27:26	P0.13	GPIO Port0.13	DTR1(UART1)	Match1.1 (TIMER1)	AD1.4	00
29:28	P0.14	GPIO Port0.14	DCD1(UART1)	EINT1	SDA1(I^2C1)	00
31:30	P0.15	GPIO Port0.15	RI1(UART1)	EINT2	AD1.5	00

หมายเหตุ :

- ขา P0.0 และ P0.1 ทำหน้าที่เป็น UART0 เพื่อติดต่อกับคอมพิวเตอร์ในการเขียนโปรแกรม
- การกำหนด ค่ารีจิสเตอร์ PINSEL0 ต้องระวังให้ P0.0 , P0.1 มีหน้าที่การทำงานเป็นขา
- TxD0 , RxD0, ของ UART0 เสมอ มิฉะนั้นอาจจะทำให้วงจรเสียได้ ดังนั้นจึงกำหนดค่าเริ่มต้นของ PINSEL0 ได้ดังนี้
- PINSEL0 = 0x00000005 ; // set P0.2 –P0.15 to GPIO Function

ตารางที่ 2.16 ค่าประจำบิตของรีจิสเตอร์ PINSEL1

PINSEL 1	ชื่อขา	การทำงาน เมื่อมีค่า 00	การทำงาน เมื่อมีค่า 01	การทำงาน เมื่อมีค่า 10	การทำงาน เมื่อมีค่า 11	ค่าหลัง รีเซต
1:0	P0.16	GPIO Port0.16	EINT0	Match0.2(TIMER0)	Capture0.2 (TIMER0)	00
3:2	P0.17	GPIO Port0.17	Capture1.2 (TIMER1)	SCK1(SPI1)	Match1.2 (TIMER1)	00
5:4	P0.18	GPIO Port0.18	Capture1.3 (TIMER1)	MISO1(SPI1)	Match1.3 (TIMER1)	00
7:6	P0.19	GPIO Port0.19	Match1.2 (TIMER1)	MOIS1(SPI1)	Capture1.2 (TIMER1)	00
9:8	P0.20	GPIO Port0.20	Match1.3 (TIMER1)	SSEL(SPI1)	EINT3	00
11:10	P0.21	GPIO Port0.21	PWM5	AD1.6	Capture1.3 (TIMER1)	00
13:12	P0.22	GPIO Port0.22	V_{BUS}	Capture0.0 (TIMER0)	Match0.0 (TIMER0)	00
15:14	P0.23	GPIO Port0.23	AD0.7	สงวนไว้	สงวนไว้	00
17:16	P0.24	สงวนไว้				00
19:18	P0.25	GPIO Port0.25	AD0.4	AOUT	สงวนไว้	00
21:20	P0.26	สงวนไว้				00
23:22	P0.27	สงวนไว้				00
25:24	P0.28	GPIO Port0.28	AD0.1	Capture0.2 (TIMER0)	Match0.2 (TIMER0)	00
27:26	P0.29	GPIO Port0.29	AD0.2	Capture0.3 (TIMER0)	Match0.3 (TIMER0)	00
29:28	P0.30	GPIO Port0.30	AD0.3	EINT3	Match0.0 (TIMER0)	00
31:30	P0.31	GPIO Port0.31	UP_LED	CONNECT		00

ตารางที่ 2.17 ค่าประจำบิตของรีจิสเตอร์ PINSEL2

PINSEL2	ความหมาย	ค่าหลังรีเซต
1:0	สงวนไว้	00
2	เป็น 0 ขา P1.31:26 จะใช้เป็น GPIO ถ้าเป็น1ขา P1.31:26จะเป็น Debug port	P1.26/RTCK
3	เป็น 0 ขา P1.25:16 จะใช้เป็น GPIO ถ้าเป็น1ขา P1.25:16จะเป็นTrace port	P1.20/ TRACESYNC

ตัวอย่างการกำหนดค่าขาให้ ขา P0.16-P0.31 มีการทำงานเป็น GPIO ทำโดยการเขียนค่า 0 ทุกบิตให้กับ PINSEL 1 ดังนี้

PINSEL1 = 0x00000000; //set P0.16-P0.31 to GPIO Function

ค่าแต่ละบิตของริจิสเตอร์ PINSEL2 ที่ใช้กำหนดหน้าที่การทำงาน Port P1 แสดงได้ในตารางที่ 2.2.14 ตัวอย่างการกำหนดค่าให้ขา P1.16-P1.31 มีการทำงานเป็น GPIO ทำการเขียนค่า 0 ทุกบิตให้กับ PINSEL1

PINSEL2 = 0x000000000; //set P1.16-P1.31 to GPIO Function

การกำหนดค่าควบคุมการทำงานพอร์ตอเนกประสงค์ (GPIO)

หลังจากที่กำหนดค่าให้พอร์ตแต่ละตัวมีการทำงานเป็น GPIO แล้ว ควบคุมการทำงานของ GPIO จะสั่งงานผ่านทางรีจีสเตอร์ 4 ตัวได้แก่ IOPN , IOSET, IOCKR, IODIRโดยที่รีจิสเตอร์แต่ละตัวจะมีแอดเดรสแสดงในตารางที่ 2.18

ตารางที่ 2.18 ชื่อและแอดเดรสของรีจิสเตอร์ที่ใช้ในการควบคุม GPIO

ชื่อทั่วไป	ความหมาย	การติดต่อ	ชื่อ และ แอดเดรสของ Port0	ชื่อ และ แอดเดรสของPort0
IOPIN	GPIO Port Pin value register ใช้อ่านสถานะปัจจุบันของพอร์ต	อ่าน	0XE002 8000 IOPIN0	0XE002 8010 IOPIN1
IOSET	GPIO Port Output set register ใช้กำหนดค่าให้ขาของพอร์ตมีค่าเป็น 1 ถ้าบิตใดเป็น 1 ขาของพอร์ตที่ตรงกันจะมีค่าเป็น 1	อ่าน/ เขียน	0XE002 8004 IOSET0	0XE002 8014 IOSET1
IODIR	GPIO Port Direction control	อ่าน/	0XE002 8008	0XE002 8018

	register ใช้กำหนดว่าขาใดเป็นอินพุตขาใดเป็นเอาต์พุต	เขียน	IODIR0	IODIR1
IOCLR	GPIO Port Output clear register ใช้กำหนดให้ขาของพอร์ตมีค่าเป็น 0	เขียน	0XE002 800C IOCLR0	0XE002 801C IOCLR1

ในไมโครคอนโทรลเลอร์ LPC2148 มีพอร์ตอเนกประสงค์ที่ให้ทำงานได้เร็วขึ้นจะเรียกว่าเป็น Fast GPIO โดยให้กำหนดรีจิสเตอร์เพิ่มเติมขึ้นอีก แต่การทำงานที่รวดเร็วขึ้นจะต้องเขียนโปรแกรมเป็นภาษาแอสแซมบลีเท่านั้น

การสั่งงานของ GPIOเริ่มต้นด้วยการกำหนดทิศทางของพอร์ตก่อนว่าจะให้เป็นอินพุตหรือเอาท์พุต โดยกำหนดค่ารีสเตอร์ IODIR โดยค่าบิตใดเป็น 0 คือให้ขาของบิตนั้นเป็นอินพุต ถ้าให้ค่าเป็น 1 ขาของบิตนั้นจะเป็นเอาต์พุต

ตัวอย่างเช่น ต้องการให้ขา P0.22, P0.20, P0.19, และ P0.16 เป็นเอาท์พุต โดยขาของ P0 ที่เหลือเป็นอินพุต จะต้องกำหนดค่าให้กับรีจิสเตอร์ IODIR0 แต่ละบิตดังนี้

บิตที่	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ค่า	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	1
	0				0				5				9			

บิตที่	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ค่า	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0				0				0				0			

เขียนเป็นโปรแกรมภาษาซีได้ดังนี้

IODIR 0 = 0X00590000;

หลังจากการกำหนดให้พอร์ตเป็นเอาต์พุตแล้ว ถ้าต้องการสั่งให้พอร์ตที่เป็นเอาต์พุตนี้มีค่าเป็น 1 ต้องสั่งที่รีสเตอร์ IOSET จากตัวอย่างต้องสั่งให้ขา P0.16 และ P.19 เป็น 1 จะต้องกำหนดให้ค่ารีจิสเตอร์ IOSET0 ดังนี้

บิตที่	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

ค่า	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
	0				0				0				9			

บิตที่	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ค่า	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0				0				0				0			

เขียนเป็นโปรแกรมภาษาซีได้ดังนี้

IOSET 0 = 0X00090000;

การเขียนค่าเป็น 0 ให้รีจิสเตอร์ IOSET จะไม่มีผลต่อขาของพอร์ตที่ตรงกับบิตนั้น ขาของพอร์ตจะมีค่าคงเดิม ถ้าต้องการส่งให้พอร์ตที่เป็นเอาต์พุตนี้มีค่าเป็น 0 ต้องสั่งที่รีจิสเตอร์ IOCLR ด้วยจากตัวอย่าง ต้องการสั่งให้ขา P0.22 และ P0.20 เป็น 0 จะต้องกำหนดค่าให้กับ รีจิสเตอร์ IOCLR0 ดังนี้

บิตที่	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ค่า	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
	0				0				5				0			

บิตที่	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ค่า	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0				0				0				0			

เขียนโดยใช้โปรแกรมภาษาซีได้ดังนี้

IOCLR = 0X00500000;

<u>ข้อควรระวัง</u> ในการสั่งให้ขาพอร์ตมีค่าเป็น 0 นี้ไม่ได้สั่งให้เขียนค่า 0 ให้กับรีจิสเตอร์ IOCLR ต้องเขียนค่าเป็น 1 เท่านั้น

ในการเขียนค่าให้พอร์ต จะต้องแยกข้อมูล ถ้าบิตใดเป็น 0 จะต้องเขียนสั่งที่รีจิสเตอร์ IOCLR ถ้าบิตใดเป็น 1 จะต้องเขียนสั่งที่รีจิสเตอร์ IOSET

มหาวิทยาลัยเทคโนโลยีสุรนารี

2.10.5 อินเตอร์รัปต์ไมโครคอนโทรลเลอร์ ARM7

ในไมโครคอนโทรลเลอร์ ARM7 มีโมดูล Vectored Interrupt Controller (VIC) เป็นตัวควบคุมกลไกลของการอินเตอร์รัปต์ โดยสามารถรับอินเตอร์รัปต์จากอุปกรณ์ทั้งภายในและภายนอกไมโครคอนโทรลเลอร์ ARM7 ได้ทั้งหมด 32 อินพุตตามที่แสดงในตารางที่ 2.19

ตารางที่ 2.19 แหล่งกำเนิดอินเตอร์รัปต์ทั้ง 32 ตัว

อุปกรณ์	แหล่งกำเนิดอินเตอร์รัปต์	VIC Channel#
WDT	Watchdog Interupt (WDINT)	0
-	Reserved for software interrupts only	1
ARM Core	Embeded ICE, DbgCommRx	2
ARM Core	Embeded ICE, DbgcommTx	3
TIME0	Match 0-3 (MR0, MR1, MR2, MR3) Capture0-3 (CR0, CR1, CR2, CR3)	4
TIME1	Match0-3 (MR0, MR1, MR2, MR3) Capture 0-3 (CR0,CR1,CR2 CR3)	5
UART0	Rx Line Status (RLS) Transmit Holding Register Empty(THRE) Rx Data Available (RDA) Charater Time-out Indicator(CTI)	6
UART1	Rx Line Status (RLS) Transmit Holding Register Empty(THRE) Rx Data Available (RDA) Charater Time-out Indicator(CTI) Modem Status Interrupt(MSI)	7
PWM0	Match 0-6 (MR0, MR1, MR2, MR3, MR4, MR5, MR6)	8
I^2C0	SI(state change)	9
SPI0	SPI Interrupt Flag (SPIF) Mode Fault(MODF)	10
SPI1(SSP)	SPI Interrupt Flag(SPIF) Mode Fault (MODF)	11
PLL	PLL Lock (PLOCK)	12
RTC	Counter Indrement(RTCCIF) Alarm(RTCALF)	13

อุปกรณ์	แหล่งกำเนิดอินเตอร์รัปต์	VIC Channel#
System Control	External Interrupt 0 (EINT0)	14
System Control	External Interrupt 1 (EINT1)	15
System Control	External Interrupt 2 (EINT2)	16
System Control	External Interrupt 3 (EINT3)	17
ADC0	A/D Converter 0 end of conversion	18
I^2C1	SI(state change)	19
BOD	Brown our detect	20
ADC1	A/D Converter 1 end of conversion	21
USB	USB interrupt, DMA Interrupt	22
	Reserved	23-31

VIC จะนำอินพุตจากการอินเตอร์รัปต์ทั้ง 32 แหล่งมาจัดแบ่งตามประเภทได้เป็น 3 ประเภท ทำให้สามารถปรับลำดับความสำคัญของอินเตอร์รัปต์จากอุปกรณ์ประกอบต่างๆ ได้ตามต้องการ คือ

- Fast Interrupt request(FIQ) จะลำดับความสำคัญสูงสุดโดยรีบตอบสนองเร็วสุด
- Vectored IRQs จะมีการลำดับความสำคัญอยู่กึ่งกลาง โดยสามารถนำอินเตอร์รัปต์แค่ 16 บิต หรือ 32 แหล่งมาจัดให้เป็น Vectored IRQs ได้ 16 ตัว โดย Slot 0 จะมีความสำคัญสูงสุด Slot 15 มีความสำคัญต่ำสุด
- Non-vectored IRQ มีความสำคัญต่ำสุด

รีจิสเตอร์ที่เกี่ยวข้องกับการอินเตอร์รัปต์มีทั้งหมด 43 ตัว ดังแสดงในตารางที่ 2.19 (ภาคผนวก) โดยในหัวข้อนี้จะเป็นการเลือกเฉพาะรีจิสเตอร์ที่เกี่ยวกับการอินเตอร์รัปต์ในแบบ Vector IRQ คือรีจิสเตอร์ VICVectAddr0-15 , VICVectCntl0-15 และ VICIntEanble

รีจิสเตอร์ VICVectAddr0-15 เป็นรีจิสเตอร์ที่เก็บค่าแอดเดรสของโปรแกรมที่ตอบสนองต่ออินเตอร์รัปต์โดย VICVectAddr0 จะเป็นอินเตอร์รัปต์ Slot 0 ที่มีความสำคัญสูงสุด ตามลำดับไปจนถึง VICVectAddr15 จะเป็นอินเตอร์รัปต์ Slot 15 ที่มีความสำคัญต่ำสุด

VICVectCntl 0- 15 ใช้ในการควบคุม คือ Slot หมายเลขที่ตรงกับ VICVectCntl นี้จะรีบอินเตอร์รัปต์จากแหล่งใด จากทั้งหมด 32 แหล่ง ตามตารางที่ 2.19 โดยกำหนดค่าในบิตที่ 4:0 โดยบิตที่ 5 ถ้าเป็น 1 หมายถึง อนุญาตให้อินเตอร์รัปต์จากอุปกรณ์ที่กำหนดใน Slot นี้

VICInEnable ใช้เปิดอินเตอร์รัปต์จากแหล่งต่างๆ ทั้ง 32 แหล่งตามตารางที่ 2.19 โดยบิตใดมีค่าเป็น 1 หมายถึงอนุญาตให้อินเตอร์รัปต์ได้ ถ้าบิตใดเป็น 0 ไม่อนุญาตให้อุปกรณ์นั้นๆ อินเตอร์รัปต์

ตารางที่ 2.20 รีจิสเตอร์ที่เกี่ยวข้องกับการอินเตอร์รัปต์

ชื่อ	ความหมาย	การติดต่อ	ค่าหลังรีเซต	แอดเดรส
VICIRQStatus	IRQ Status Request อ่านค่าสถานะว่าอนุญาตให้มี interrupt request จากตัวใด และถูกจัดเป็น IRQ	อ่าน	0	0xFFFF F000
VICFIQStatus	FIQ Status Request อ่านค่าสถานะว่าอนุญาตให้มีอินเตอร์รัปต์รีเควสจากตัวใด และถูกจัดเป็น FIQ	อ่าน	0	0xFFFF F004
VICRawltr	Raw Interrupt Status Register ใช้อ่านสถานะของอินเตอร์รัปต์จากทั้ง 32 แหแหล่งหรือจากซอฟต์แวร์โดยไม่สนว่าถูกเปิดใช้หรือถูกจัดประเภทหรือไม่	อ่าน/เขียน	0	0xFFFF F008
VICintselect	Interrupt Select Register ใช้ในการระบุว่าอินเตอร์รัปต์ทั้ง 32 แหล่งแต่ละตัวเป็น FIQ หรือ IRQ	เขียน	0	0xFFFF F00C
VICIntEnable	Inerrupt E able Register ควบคุมว่าให้อินเตอร์รัปต์จากทั้ง32 แหล่งตัวไหนทำงาน และให้เป็น FIQ หรือ IRQ	อ่าน/เขียน	0	0xFFFF F010
VICIntEnClr	Interrupt Enable Clear Register ใช้ล้างค่าบิตหนึ่งบิตหรือมากกว่าหนึ่งบิตของรีจิสเตอร์ Interrupt Enable Register	เขียน	0	0xFFFF F014
VicSoftlnt	Software Interrupt Register ค่าของรีจิสเตอร์นี้จะถูกจำไป OR กับอินเตอร์รัปต์จากอุปกรณ์ต่างๆ ทั้ง 32 แหล่ง	อ่าน/เขียน	0	0xFFFF F018
VICSoftlntClear	Software Interrupt Clear Register ใช้ล้างค่าบิตหนึ่งบิตหรือมากกว่าหนึ่งบิตของรีจิสเตอร์ Software Interrupt Register	เขียน	0	0xFFFFF01C
VICProtection	Protection enable register ใช้จำกัดการเข้าถึง VIC Register	อ่าน/เขียน	0	0xFFFF F020

ชื่อ	ความหมาย	การติดต่อ	ค่าหลังรีเซต	แอดเดรส
VICVectAddr	Vector Address Register เมื่อเกิดการอินเตอร์รัปต์แบบ IRQ ตัว IRQ Sevice routine จะอ่านค่าจากรีจิสเตอร์เพื่อกระโดดไปยังแอดเดรสที่ระบุไว้	อ่าน/เขียน	0	0xFFFF F030
VICDefVectAddr	Default Vector Address Register ใช้เก็บค่าแอดเดรสของ Interrupt service routine(ISR) สำหรับ IRQ non-vectored IRQ	อ่าน/เขียน	0	0xFFFF F100
VICVEctAddr1	Vector address 1 register	อ่าน/เขียน	0	0xFFFF F104
VICVEctAddr2	Vector address 2 register	อ่าน/เขียน	0	0xFFFF F108
VICVEctAddr3	Vector address 3 register	อ่าน/เขียน	0	0xFFFFF10C
VICVEctAddr4	Vector address 4 register	อ่าน/เขียน	0	0xFFFF F110
VICVEctAddr5	Vector address 5 register	อ่าน/เขียน	0	0xFFFF F114
VICVEctAddr6	Vector address 6 register	อ่าน/เขียน	0	0xFFFF F118
VICVEctAddr7	Vector address 7 register	อ่าน/เขียน	0	0xFFFFF11C
VICVEctAddr8	Vector address 8 register	อ่าน/เขียน	0	0xFFFF F120
VICVEctAddr9	Vector address 9 register	อ่าน/เขียน	0	0xFFFF F124
VICVEctAddr10	Vector address 10 register	อ่าน/เขียน	0	0xFFFF F128
VICVEctAddr11	Vector address 11 register	อ่าน/เขียน	0	0xFFFFF12C
VICVEctAddr12	Vector address 12 register	อ่าน/เขียน	0	0xFFFF F130
VICVEctAddr13	Vector address 13 register	อ่าน/เขียน	0	0xFFFF F134
VICVEctAddr14	Vector address 14 register	อ่าน/เขียน	0	0xFFFF F138
VICVEctAddr15	Vector address 15 register	อ่าน/เขียน	0	0xFFFFF13C
VICVectCntl0	Vector control 0 register รีจิสเตอร์ Vector Control Register0-15 แต่ละตัวควบคุม IRQ Slot แต่ละตัว โดย Slot 0 มีความสำคัญสูงสุดและ Slot 15 มีความสำคัญต่ำสุด	อ่าน/เขียน	0	0xFFFF F200
VICVectCntl1	Vector control 1 register	อ่าน/เขียน	0	0xFFFF F204
VICVectCntl2	Vector control 2 register	อ่าน/เขียน	0	0xFFFF F208

ชื่อ	ความหมาย	การติดต่อ	ค่าหลังรีเซต	แอดเดรส
VICVectCntl3	Vector control 3 register	อ่าน/เขียน	0	0xFFFF F20C
VICVectCntl4	Vector control 4 register	อ่าน/เขียน	0	0xFFFF F210
VICVectCntl5	Vector control 5 register	อ่าน/เขียน	0	0xFFFF F214
VICVectCntl6	Vector control 6 register	อ่าน/เขียน	0	0xFFFF F218
VICVectCntl7	Vector control 7 register	อ่าน/เขียน	0	0xFFFFF21C
VICVectCntl8	Vector control 8 register	อ่าน/เขียน	0	0xFFFF F220
VICVectCntl9	Vector control 9 register	อ่าน/เขียน	0	0xFFFF F224
VICVectCntl10	Vector control10 register	อ่าน/เขียน	0	0xFFFF F228
VICVectCntl11	Vector control 11 register	อ่าน/เขียน	0	0xFFFFF22C
VICVectCntl12	Vector control 12 register	อ่าน/เขียน	0	0xFFFF F230
VICVectCntl13	Vector control 13 register	อ่าน/เขียน	0	0xFFFF F234
VICVectCntl14	Vector control 14 register	อ่าน/เขียน	0	0xFFFF F238
VICVectCntl15	Vector control 15 register	อ่าน/เขียน	0	0xFFFFF23C

การอินเตอร์รัปต์จากอินพุตภายนอกของไมโครคอนโทรลเลอร์ ARM7

ในไมโครคอนโทรลเลอร์ LPC2148 จะรับอินเตอร์รัปต์จากภายนอกได้สูงสุด 4 แหล่ง คือ EINT0, ENIT1, ENIT2 และ ENIT3 โดยคิดเป็นอินพุตได้ทั้งหมด 9 ตัว

- EINT0 อยู่ที่ขา P0.1 และ P0.16
- EINT1 อยู่ที่ขา P0.3 และ P0.14
- EINT2 อยู่ที่ขา P0.7 และ P0.15
- EINT3 อยู่ที่ขา P0.9, P0.20 และ P0.30

โดยสามารถนำอินพุตจากการอินเตอร์รัปต์ทั้งสี่ตัวนี้ มาใช้ในการกระตุ้นให้ไมโครคอนโทรลเลอร์ออกจากการทำงานในโหมด Power Down ได้

รีจิสเตอร์ที่เกี่ยวข้องกับอินเตอร์รัปต์จากภายนอก

รีจิสเตอร์ที่เกี่ยวข้องกับการอินเตอร์รัปต์จากภายนอกมีทั้งหมด 4 ตัว คือ EXTINT, EXTWAKE, EXXTMODE และ EXTPOLAR ดังแสดงในตารางที่ 2.21

ตารางที่ 2.21 รีจิสเตอร์ที่เกี่ยวข้องกับการอินเตอร์รัปต์จากภายนอก

แอดเดรส	ชื่อ	ความหมาย	การติดต่อ
0xE01FC140	EXTINT	External Interrupt Flag Register เก็บค่าอินเตอร์รัปต์แฟล็กของ EINT0, EINT1 และ EINT2	อ่าน/เขียน
0xE01FC144	EXTWAKE	External Interrupt Wakeup Register เก็บค่าบิตที่ควบคุมว่าจะให้อินเตอร์รัปต์จากภายนอกนี้สามารถกระตุ้นซีพียูจาก Power mode หรือไม่	อ่าน/เขียน
0xE01FC148	EXTMODE	External Interrupt Mode Register ควบคุมให้แต่ละระดับจะตอบสนองต่อระดับลอกจิกหรือขอบสัญญาณ	อ่าน/เขียน
0xE01FC14C	EXTPOLAR	External Interrupt Polarity Register ควบคุมว่าแต่ละขาจะตอบสอนงระดับลอกจิก 0 หรือ1 หรือตอบสนองต่อขอบขาขึ้นขอบขาลงของสัญญาณ	อ่าน/เขียน

เมื่อเกิดการอินเตอร์รัปต์จากภายนอกเกิดขึ้น จะทำให้ค่าบิตบางบิตของรีจิสเตอร์ EXTINT มีค่าเป็น 1 และส่งต่อการอินเตอร์รัปต์ไปยัง VIC เพื่อสร้างอินเตอร์รัปต์ขัดการทำงานของไมโครคอนโทรลเลอร์ เมื่อโปรแกรมตอบสนองต่อการอินเตอร์รัปต์ ทำงานเสร็จแล้วจะต้องเขียนค่า 1 ให้กับบิตเหล่านี้เพื่อหยุดการอินเตอร์รัปต์ ค่าแต่ละบิตของรีจิสเตอร์ EXTINT แสดงในตารางที่ 2.22

ตารางที่ 2.22 ค่าประจำบิตของรีจิสเตอร์ EXTINT

ค่าบิตของ EXTINT	หน้าที่	ความหมาย	ค่าหลังรีเซ็ต
0	EINT0	เมื่อขา EINT0 ทำงานมีระดับลอจิกหรือมีขอบของสัญญาณตามที่กำหนดเกิดขึ้น บิตนี้จะมีค่าเป็น 1 เมื่อเขียนค่า 1 ให้มันเป็นการล้างค่า ยกเว้นกรณีที่ทำงานตามระดับลอจิกต้องรอให้ค่าลอจิกหยุดการทำงานก่อน ขาที่เป็น EINT0 คือ P0.1 และ P0.16	0
1	EINT1	เมื่อขา EINT1 ทำงานมีระดับลอจิกหรือมีขอบของสัญญาณตามที่กำหนดเกิดขึ้น บิตนี้จะมีค่าเป็น 1 เมื่อ	0

		เขียนค่า 1 ให้มนเป็นการล้างค่า ยกเว้นกรณีที่ทำงานตามระดับลอจิกต้องรอให้ค่าบอจิกหยุดการทำงานก่อน ขาที่เป็น EINT1 คือ P0.3 และ P0.14	
2	EINT2	เมื่อขา EINT2 ทำงานมีระดับลอจิกหรือมีขอบของสัญญาณตามที่กำหนดเกิดขึ้น บิตนี้จะมีค่าเป็น 1 เมื่อเขียนค่า 1 ให้มนเป็นการล้างค่า ยกเว้นกรณีที่ทำงานตามระดับลอจิกต้องรอให้ค่าบอจิกหยุดการทำงานก่อน ขาที่เป็น EINT2 คือ P0.7 และ P0.15	0
3	EINT3	เมื่อขา EINT3 ทำงานมีระดับลอจิกหรือมีขอบของสัญญาณตามที่กำหนดเกิดขึ้น บิตนี้จะมีค่าเป็น 1 เมื่อเขียนค่า 1 ให้มนเป็นการล้างค่า ยกเว้นกรณีที่ทำงานตามระดับลอจิกต้องรอให้ค่าบอจิกหยุดการทำงานก่อน ขาที่เป็น EINT3 คือ P0.9, P0.20 และ P0.30	0
7:4	สงวนไว้	ห้ามเขียนค่า 1 ให้กับบิตเหล่านี้ ค่าที่อ่านได้ไม่มีความหมาย	ไม่กำหนด

ในการกำหนดว่าสัญญาณที่อินเตอร์รัปต์นี้ทำงานที่ระดับสัญญาณหรือที่ขาขอบของสัญญาณกำหนดได้ที่รีจิสเตอร์ EXTMODE ซึ่งมีค่าดังตารางที่ 2.23

ตารางที่ 2.23 ค่าประจำบิตของรีจิสเตอร์ EXTMODE

ค่าบิตของ EXTWAKE	**หน้าที่**	**ความหมาย**	**ค่าหลังรีเซ็ต**
0	EXTWAKE0	โหมดการทำงานของ EINT0 เมื่อมีค่าเป็น 0 ทำงานที่ระดับสัญญาณ เมื่อมีค่าเป็น 1 ทำงานที่ขอบของสัญญาณ	0
1	EXTWAKE1	โหมดการทำงานของ EINT1 เมื่อมีค่าเป็น 0 ทำงานที่ระดับสัญญาณ เมื่อมีค่าเป็น 1 ทำงานที่ขอบของสัญญาณ	0
2	EXTWAKE2	โหมดการทำงานของ EINT2 เมื่อมีค่าเป็น 0 ทำงานที่ระดับสัญญาณ เมื่อมีค่าเป็น 1 ทำงานที่ขอบของสัญญาณ	0
3	EXTWAKE3	โหมดการทำงานของ EINT3 เมื่อมีค่าเป็น 0 ทำงานที่ระดับสัญญาณ เมื่อมีค่าเป็น 1 ทำงานที่ขอบของสัญญาณ	0

7:4	สงวนไว้	ห้ามเขียนค่า 1 ให้กับบิตเหล่านี้ ค่าที่อ่านได้ไม่มีความหมาย	ไม่กำหนด

หลังจากที่กำหนดโหมดการทำงานของสัญญาณที่ใช้อินเตอร์รัปต์ว่าเป็นระดับของสัญญาณหรือขอบสัญญาณแล้วที่รีจิสเตอร์ EXTMODE แล้วต้องกำหนดว่าสัญญาณที่อินเตอร์รัปต์นี้ทำงานที่ระดับลอจิก 1 หรือ 0 หรือทำงานที่ขอบขาขึ้นหรือขาลงของสัญญาณ โดยกำหนดที่รีจิสเตอร์ EXTPOLAR ซึ่งแสดงรายละเอียดดังตารางที่ 2.24

ตารางที่ 2.24 ค่าประจำบิตของรีจิสเตอร์ EXTPOLAR

ค่าบิตของ EXTPOLAR	หน้าที่	ความหมาย	ค่าหลังรีเซ็ต
0	EXTPOLAR0	เป็น 0 EINT0 ทำงานที่ลอจิก 0 หรือขอบขาลงของสัญญาณ เป็น1 EINT0 ทำงานที่ลอจิก 1 หรือขอบขาขึ้นของสัญญาณ	0
1	EXTPOLAR1	เป็น 0 EINT1 ทำงานที่ลอจิก 0 หรือขอบขาลงของสัญญาณ เป็น1 EINT1 ทำงานที่ลอจิก 1 หรือขอบขาขึ้นของสัญญาณ	0
2	EXTPOLAR2	เป็น 0 EINT2 ทำงานที่ลอจิก 0 หรือขอบขาลงของสัญญาณ เป็น1 EINT2 ทำงานที่ลอจิก 1 หรือขอบขาขึ้นของสัญญาณ	0
3	EXTPOLAR3	เป็น 0 EINT3 ทำงานที่ลอจิก 0 หรือขอบขาลงของสัญญาณ เป็น1 EINT3 ทำงานที่ลอจิก 1 หรือขอบขาขึ้นของสัญญาณ	0
7:4	สงวนไว้	ห้ามเขียนค่า 1 ให้กับบิตเหล่านี้ ค่าที่อ่านได้ไม่มีความหมาย	ไม่กำหนด

ถ้าต้องการให้สัญญาณที่ขา EINT0-EINT3 สามารถกระตุ้นให้ไมโครคอนโทรลเลอร์ ARM7 ออกจากการทำงานในโหมด Power Down จะต้องสั่งที่รีจิสเตอร์ EXTWAKE ซึ่งมีค่บิตของรีจิสเตอร์แสดงในตารางที่ 2.25

ตารางที่ 2.25 ค่าประจำบิตของรีจิสเตอร์ EXTPILAR

ค่าบิตของ EXTMODE	หน้าที่	ความหมาย	ค่าหลังรีเซ็ต
0	EXTMODE0	เป็น 1 สัญญาณที่ขา EINT0 สามารถกระตุ้นให้ออกจาก Power Down ได้	0
1	EXTMODE1	เป็น 1 สัญญาณที่ขา EINT1 สามารถกระตุ้นให้ออกจาก Power Down ได้	0
2	EXTMODE2	เป็น 1 สัญญาณที่ขา EINT2 สามารถกระตุ้นให้ออกจาก Power Down ได้	0
3	EXTMODE3	เป็น 1 สัญญาณที่ขา EINT0 สามารถกระตุ้นให้ออกจาก Power Down ได้	0
7:4	สงวนไว้	ห้ามเขียนค่า 1 ให้กับบิตเหล่านี้ ค่าที่อ่านได้ไม่มีความหมาย	ไม่กำหนด

2.10.6 การกำหนดค่าเริ่มต้นสำหรับ UART0

UART0 มีขาสัญญาณสองขา คือ

1. ขาสัญญาณ TxD0 ไว้สำหรับส่งข้อมูลอยู่ที่ขา
2. ขาสัญญาณ RxD0 ไว้สำหรับรับข้อมูลอยู่ที่ขา P0.1

การใช้งาน UART0 เริ่มต้นด้วยการเขียนค่ายังรีจิสเตอร์ PINSEL0 เพื่อกำหนดให้ขา P0.0 และ P0.1 มีการทำงานเป็น UART0 โดยต้องกำหนดบิตที่ 3-0 ของ PINSEL0 ให้มีค่าเป็น 1010 ซึ่งเขียนเป็นคำสั่งได้ดังนี้

```
PINSEL0 = 0x00000005;
```

รีจิสเตอร์ที่เกี่ยวข้องกับ UART0 มีทั้งหมด 10 ตัวโดยแต่ละตัวมีขนาด 8 บิต ดังแสดงในตารางที่ 2.26

ตารางที่ 2.26 รีจิสเตอร์ที่เกี่ยวข้องกับ UART0

ชื่อ	ความหมาย	บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0	การติดต่อ	ค่าหลังรีเซต	แอดเดรส
U0RBR	Receiver Buffer Register	MSB	อ่านข้อมูล						LSB	อ่าน	ไม่กำหนด	0xE000C000 DLAB=0
U0THR	Transmit Holding Register	MSB	เขียนข้อมูล						LSB	เขียน	ไม่กำหนด	0xE000C000 DLAB=0
U0IER	Interrupt Enable Register	0	0	0	0	0	Enable Rx Line status Interrupt	Enable THRE Interrupt	Enable Rx Data Available Interrupt	อ่าน/ เขียน	0	0xE000C004 DLAB=0
U0IIR	Interrupt ID Register	FIFOs Enable		0	0	IIR3	IIR2	IIR1	IIR0	อ่าน	0x01	0xE000C008
U0FCR	FIFO Control Register	Rx Trigger		สงวนไว้		-	Tx FIFO Reset	Rx FIFO Reset	FIFO Enable	เขียน	0	0xE000C008
U0LCR	Line Control Register	DLAB	Set Break	Stick Parity	Even Parity Select	Parity Enable	Number of Stop Bit	World Length Select		อ่าน / เขียน	0	0xE000C00C
U0LSR	Line Status Register	Rx FIFO Error	TEMT	THRE	BI	FE	PE	OE	DR	อ่าน	0X60	0xE000C014
U0SCR	Scratch Pad Register	MSB LSB								อ่าน / เขียน	0	0xE000C01C
U0DLL	Divisor Latch LSB	MSB LSB								อ่าน / เขียน	0x01	0xE000C000 DLAB=1
U0DLM	Divisor Latch MSB	MSB LSB								อ่าน / เขียน	0	0xE000C004 DLAB=1

หลังจากที่กำหนดให้ขา P0.0 และ P0.1 ทำงานเป็น UART0 แล้วถัดมาเป็นการกำหนดรูปแบบการติดต่อ เช่น ติดต่อแบบ 8 ใช้การตรวจสอบบิตพลาดแบบใด เช่น even parity จำนวนของ Stop bit โดยการกำหนดค่าผ่านทางรีจิสเตอร์ UART Line Control Register : LCR : ซึ่งมีรายละเอียดของรีจิสเตอร์ดังแสดงในตารางที่ 2.27

ตารางที่ 2.27 ค่าประจำบิตของรีจิสเตอร์ UART0 Line Control Register (U0LCR)

ค่าของบิต U0LCR	หน้าที่	ความหมาย	ค่าหลังรีเซ็ต
1:0	World Length Select	ตัวอักษรขนาด 5 บิต ตัวอักษรขนาด 6 บิต ตัวอักษรขนาด 7 บิต ตัวอักษรขนาด 8 บิต	0
2	Stop Bit Select	0 Stop bit 1 บิต 1 Stop Bit 2 บิต(1.5บิต ถ้า U0LCR[1:0]=00)	0
3	Parity Enable	0 ยกเลิกการเพิ่มพาริตี้บิตและการตรวจพาริตี้ 1 ใช้งานการเพิ่มพาริตี้บิตและการตรวจพาริตี้	0
5:4	Parity Select	Odd parity Even Parity ให้พาริตี้บิตมีค่าเป็น 1 ตลอดเวลา ให้พาริตี้บิตมีค่าเป็น 0 ตลอดเวลา	0
6	Break Control	ยกเลิกการหยุดการสั่ง อนุญาตให้หยุดการส่งให้ ขาที่เอาต์พุตของ UART0 TxD จะมีค่าลอจิก 0	0
7	Divisor Latch Access Bit	ไม่อนุญาตให้แก้ไขค่าตัวหาร Divisor Latch อนุญาตให้แก้ไขค่าตัวหาร Divisor Latch	0

ในรีจิสเตอร์ LCR มีบิตที่เรียกว่า DLAB (Divisor Latch Access bit) ถ้าต้องการปรับค่าของวงจรกำเนิด Broad Rate ต้องเซตบิตนี้ให้เป็น 1 ค่าของ Baud rate generator เป็นค่าของตัวหารขนาด 16 บิต เพื่อนำไปใช้หารค่าของ PCLK เพื่อให้ได้ความถี่สูงกว่าความเร็ว Broad Rate 16 เท่า ทำให้สมการค่าตัวหารเป็นดังนี้

$$Divisor = PCLK/(16*Baud)$$

แผงวงจร CP-JR ARM7 USB-LPC2148 EXP ใช้คริสตัลความถี่ 12.000MHz สั่ง PLL คูณ 5 จะได้ CCLK = 60.0 MHz และกำหนดให้ VPBDIV = 2 จะได้ PCLK = 30.0 MHz ด้วย ถ้าต้องการอัตราบอดที่ 9600 bps ตัวหารจะมีค่าเท่ากับ

Divisor = 30,000,000/(60*9600) = 195.3125 ปัดเศษลง

Divisor = 195 หรือ 0xC3 นำค่าที่ได้ไปคำนวณหาอัตราบอดจะได้

Baud = PCLK/(16*Divisor) = 30,000,000/(16*195) = 9615 bps

ซึ่งผิดพลาดไป 0.156% สามารถให้งานได้ เนื่องมาตรฐานของการสื่อสารแบบอนุกรมสามารถรับอัตราบอดที่ผิดพลาดได้ถึง 5%

นำค่าหารที่ได้ไปเก็บลงในรีจิสเตอร์ขนาด 8 บิตสองตัว คือ Divisor Latch MAB(DLM) และ Divisor Latch LSB (DLL) ในขณะที่เขียนค่าเก็บในรีจิสเตอร์ DLM และ DLL ค่าบิต DLAB ต้องมีค่าเป็น 1 เมื่อเขียนเสร็จแล้วต้องรีเซตค่าบิตนี้ให้กลับเป็น 0 ซึ่งเขียนเป็นคำสั่งภาษาซีได้ดังนี้

```
U0DLM = 0x00;
U0DLL = 0xC3;
U0LCR = 0x7F;
```

ในการเรียกใช้ฟังก์ชัน uart0_init() จะต้องส่งค่าอัตราบอดที่ต้องการให้ฟังก์ชัน ตัวอย่างเช่นต้องการอัตราบอดที่ 9600 bps จะต้องเรียกใช้ฟังก์ชันดังนี้

```
uart0_init(9600);
```

เมื่อกำหนดการทำงานให้กับ UART แล้ว จะสามารถรับส่งค่าผ่านพอร์ตอนุกรมได้ในการส่งข้อมูลต้องเขียนข้อมูลไปยังรีจิสเตอร์ Transmit Holding Register (THR) ถ้าต้องการอ่านข้อมูลที่รับพอร์ตอนุกรมต้องอ่านค่าจากรีจิสเตอร์ Receiver Buffer Register (RBR) จากตารางที่ 2.2.24 (ภาคผนวก) จะพบว่าค่าแอดเดรสของรีจิสเตอร์ทั้งสอง มีค่าอยู่ที่ตำแหน่งเดียวกัน แสดงว่าการเขียนค่าให้กับ THR เป็นการเขียนค่าลงในบัฟเฟอร์แบบ FIFO ของ UART0 การอ่านค่าจากรีจิสเตอร์ RBR เป็นการอ่านค่าจากบัฟเฟอร์แบบ FIFO

ก่อนที่จะอ่านหรือเขียนค่าลงในรีจิสเตอร์ THR หรือ RBR จะต้องอ่านค่าสถานะของ UART ก่อนว่ามีการผิดพลาดหรือไม่ โดยอ่านค่าสถานะที่รีจิสเตอร์ Line Status Register (LSB) ก่อน โดยค่าประจำบิตของรีจิสเตอร์แสดงได้ในตารางที่ 2.28

ตารางที่ 2.28 แสดงค่าประจำบิตของ UART0 Line Status Register (U0LSR)

ค่าบิตของ U0LSR	หน้าที่	ความหมาย	ค่าหลังรีเซต
0	Receiver Data Ready (RDR)	0 : U0RBR ว่าง 1 : U0RBR มีรับข้อมูลที่ถูกต้อง U0LSR0 เป็น 1 เมื่อ U0RBR มีค่าตัวอักษรที่ยังไม่อ่านและถูกล้างค่าเมื่อ UART0 RBR FIFO ว่าง	0
1	Overrun Error (OE)	0 : ไม่มี Overrun error 1 : มี Overrun error เกิดขึ้น	0

ค่าบิตของ U0LSR	หน้าที่	ความหมาย	ค่าหลังรีเซต
		Overrun error เกิดขึ้นเมื่อ UART0 RSR ได้รับตัวอักษรตัวใหม่ในขณะที่บัฟเฟอร์ FIFO เต็ม ในกรณีนี้ข้อมูลตัวใหม่ใน UART0 RSR จะสูญหายไป เมื่ออ่านค่าของ U0LSR จะล้างค่าบิตนี้	
2	Parity Error (PE)	0 : ไม่มี Parity Error 1 : มี Parity Error เกิดขึ้น ใช้ตรวจสอบว่าข้อมูลที่รับมาใน UART0 RBR FIFO มีการผิดพลาดที่พาริตี้บิตหรือไม่ เมื่ออ่านค่าของ U0LSR จะล้างค่าของบิตนี้	0
3	Framing Error (FE)	0 : ไม่มี Framing Error 1 : มี Framing Error เกิดขึ้น เมื่อ Stop bit ของข้อมูลที่รับมามีค่าเป็น 0 แสดงว่าเกิดการผิดพลาดที่เฟรมข้อมูลในขณะที่เกิดการผิดพลาดที่เฟรมข้อมูลนี้ UART0 จะพยายามรับข้อมูลโดยนำ Stop bit ที่ผิดพลาดนี้มาใช้เป็น Start bit ข้อมูลตัวถัดไป ซึ่งอาจจะอ่านข้อมูลได้ถูกต้องหรือไม่ถูกต้อง	0
4	Break Interrupt (BI)	0 : ไม่ใช้ Break Interrupt 1 : ใช้ Break Interrupt เมื่อ RxD0 ได้รับข้อมูลที่เป็น 0 ทุกบิต (start, data, parity, stop) จะเกิด Break Interrupt ภาครับจะหยุดทำงาน จนกว่าจะได้รับข้อมูลที่เป็น 1 ทุกบิตอีกครั้งหนึ่ง	0
5	Transmitter Holding Register Empty (THRE)	0 : U0THR มีข้อมูลที่ถูกต้อง 1 : U0THR ว่าง บิต THRE จะถูกเซ็ตค่าเป็น 1 ทันทีที่พบว่า UART0 THR ว่าง และถูกล้างค่าทันทีมีการเขียนค่าไปยัง U0THR	1
6	Transmitter Empty (TEMT)	0 : U0THRและ/หรือ U0TSR มีข้อมูลที่ถูกต้อง 1 : U0THR และ U0TSR ว่าง บิตนี้จะถูกล้างค่าเมื่อ U0THR หรือ U0TSR มีข้อมูลที่ถูกต้อง	1

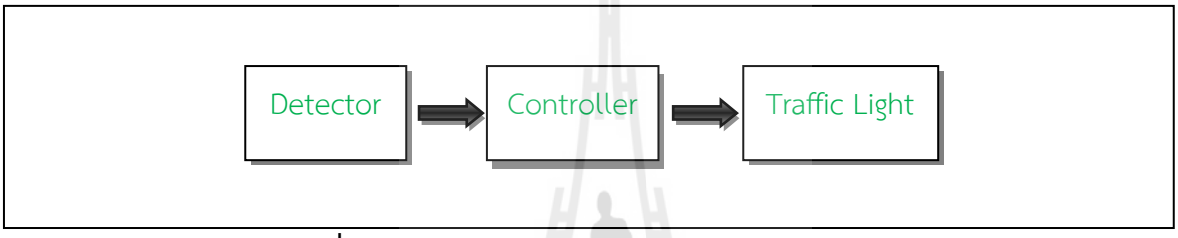
ค่าบิตของ U0LSR	หน้าที่	ความหมาย	ค่าหลังรีเซต
7	Error in Rx FIFO (RXFE)	0 : ข้อมูลที่เก็บใน UART0 RBR FIFO ไม่ผิดพลาด 1 : ข้อมูลที่เก็บใน UART0 RBR FIFO อย่างน้อยหนึ่งตัวมีการผิดพลาด	0

ก่อนที่จะเขียนข้อมูลให้ UART ต้องตรวจสอบดูที่บิต Transmitter Empty (TEMT) ของรีจิสเตอร์ LSR ก่อนว่าบัฟเฟอร์สำหรับส่งว่างหรือไม่ ถ้าว่างจะได้ค่าเป็น 1 จึงส่งข้อมูลได้ก่อนที่จะอ่านข้อมูลจาก UART ต้องตรวจสอบบิต Receiver Data Ready (RDR) ก่อน ถ้ามีข้อมูลพร้อมแล้วบิตนี้จะมีค่าเป็น 1 จึงอ่านค่าจาก UART ได้

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 3
การออกแบบระบบควบคุมสัญญาณไฟจราจร

การควบคุมสัญญาณไฟจราจรนี้ได้มีการออกแบบระบบโดยแบ่งส่วนของการออกแบบเป็น 2. ส่วนคือ 1. การควบคุมสัญญาณไฟจราจร (Controller) 2. ตัวตรวจจับยานพาหนะ (Detector) โดยในแต่ละส่วนจะมีหน้าที่การทำงานที่แตกต่างกันออกไปโดยที่ส่วนการควบคุมไฟสัญญาณจราจรจะรับข้อมูลจากตัวตรวจจับยานพาหนะเพื่อนำข้อมูลที่ได้มาประมวลผลในการควบคุมสัญญาณไฟจราจรที่เหมาะสม ซึ่งสามารถอธิบายได้ดังรูปที่.3.1



รูปที่.3.1 แสดงโครงสร้างระบบควบคุมสัญญาณไฟจราจร

ระบบควบคุมสัญญาณไฟจราจรในงานวิจัยนี้จะเป็นระบบการควบคุมแบบกึ่งอัตโนมัติ (Semi-Actuated) ใช้ควบคุมสัญญาณไฟจราจรที่มีลักษณะแยกจราจรที่มีทางเอกและทางโท โดยให้รถทางเอกสามารถผ่านได้ตลอดเวลาเมื่อไม่มียานพาหนะมารอที่ทางโท แต่เมื่อมียานพาหนะมารอในทางโทจะเปลี่ยนสัญญาณไฟจราจรให้ยานพาหนะในทางโทสามารถผ่านไปได้ ซึ่งเหมาะสมสำหรับการควบคุมจราจรที่มียานพาหนะในทางโทที่มีปริมาณการจราจรที่ต่ำ

มหาวิทยาลัยเทคโนโลยีสุรนารี

ส่วนควบคุมสัญญาณไฟจราจร (Controller)

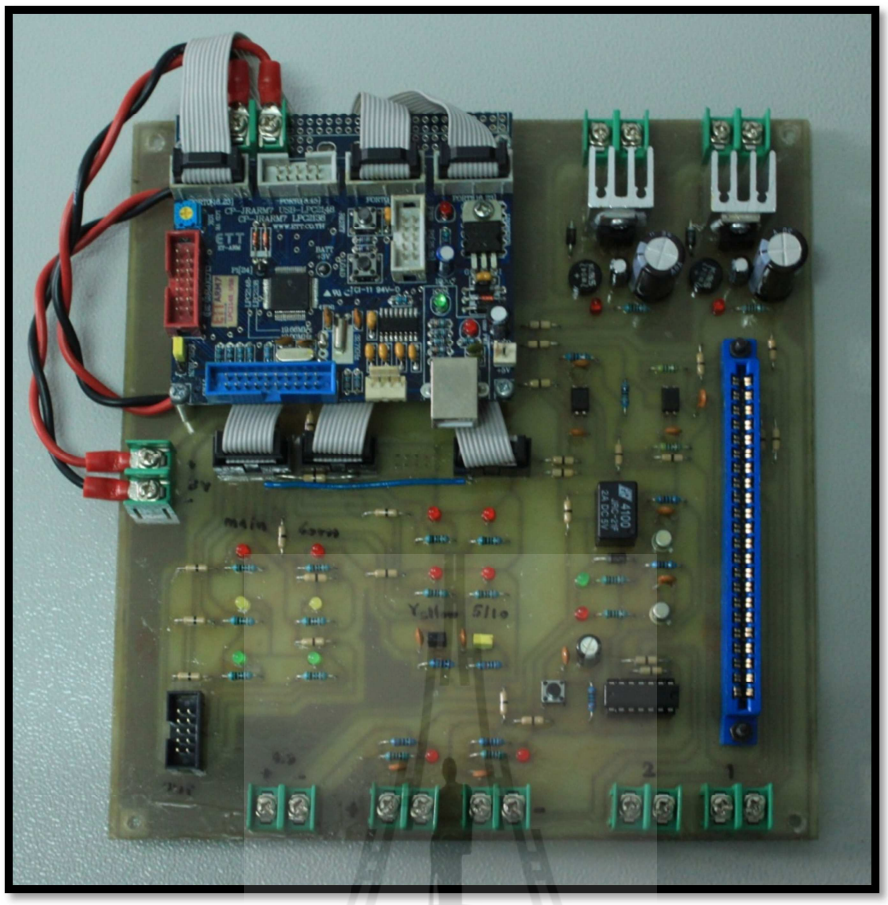
ในส่วนนี้จะเป็นส่วนการควบคุมหลักของระบบควบคุมสัญญาณไฟจราจร โดยการออกแบบออกเป็น 2 ส่วน คือ 1. ส่วนของการออกแบบฮาร์ดแวร์ (Hardware) 2. ส่วนของการออกแบบซอร์ฟแวร์ (Software) ส่วนของการออกแบบฮาร์ดแวร์จะมีองค์ประกอบคือวงจรควบคุมต่าง ๆ ที่ใช้ในระบบการควบคุมสัญญาณไฟจราจร ส่วนการออกแบบซอร์ฟแวร์จะเป็นส่วนของโปรแกรมที่ใช้ควบคุมและประมวลผลของระบบควบคุมสัญญาณไฟจราจรโดยอาศัยรูปแบบการควบคุมแบบกึ่งอัตโนมัติ นอกจากนี้ยังเป็นส่วนของการกำหนดรูปแบบการทำงานต่าง ๆ ในระบบควบคุม

3.1 ส่วนของการออกแบบฮาร์ดแวร์ (Hardware) ของส่วนควบคุมสัญญาณไฟจราจร (Controller)

ส่วนการออกแบบฮาร์ดแวร์ของระบบควบคุมสัญญาณไฟจราจรจะออกแบบวงจรอิเล็กทรอนิกส์เพื่อให้สามารถควบคุมการทำงานของระบบควบคุมสัญญาณไฟจราจรได้ ในส่วนนี้จะเป็นตัวควบคุมการทำงานหลักของระบบ โดยจะประกอบด้วยส่วนการทำงานต่างๆ ดังนี้

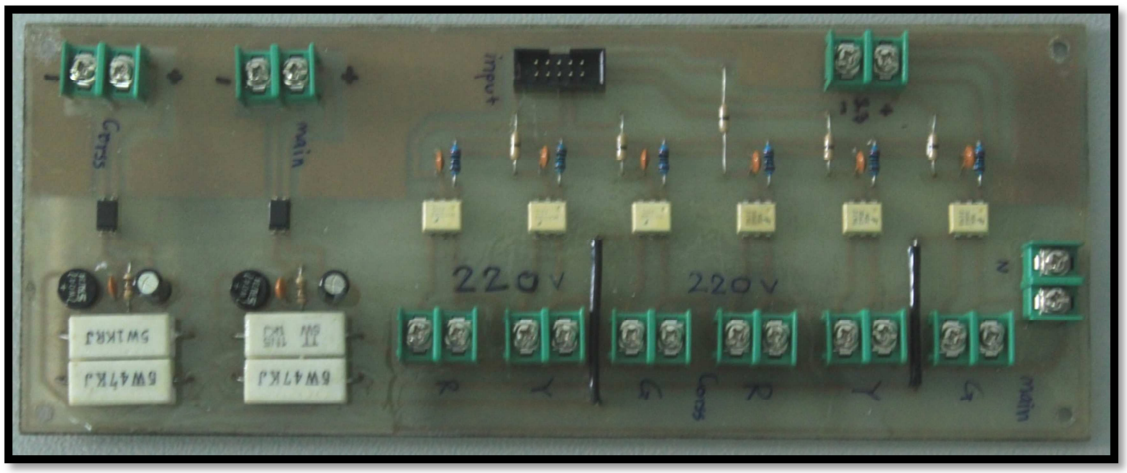
1. บอร์ดไมโครคอนโทลเลอร์ ARM7 LPC2148
2. วงจรเชื่อมต่อตัวตรวจจับยานพาหนะ (Detector Socket)
3. วงจรแสดงผล (Display)
4. วงจรป้องกัน (Protection Circuit)
5. วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร (Driver 220 VAC)

มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่.3.2 วงจรสำหรับควบคุมสัญญาณไฟจราจร

รูปที่.3.2 คือ วงจรควบคุมสัญญาณไฟจราจรที่ออกแบบสำหรับงานวิจัยนี้ โดยมีขนาด 18 x 20 เซนติเมตร จะทำหน้าที่ควบคุมสัญญาณไฟจราจรในรูปแบบการควบคุมแบบกึ่งอัตโนมัติ โดยจะรับสัญญาณจากตัวตรวจยานพาหนะเพื่อนำมาประมวลผลให้ได้ตามที่กำหนดเอาไว้

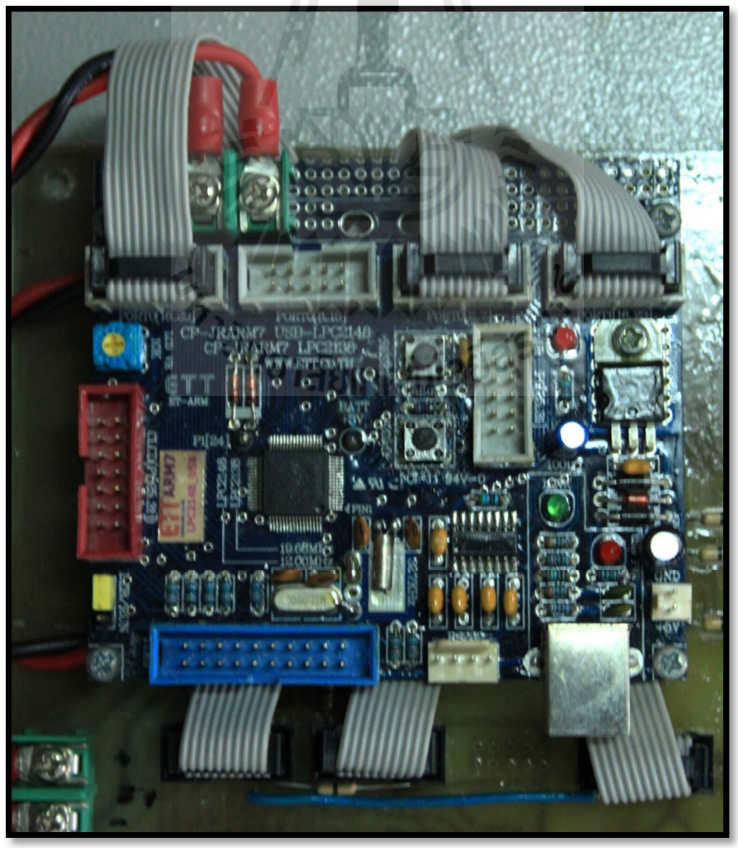


รูปที่.3.3 วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร (Driver 220 VAC)

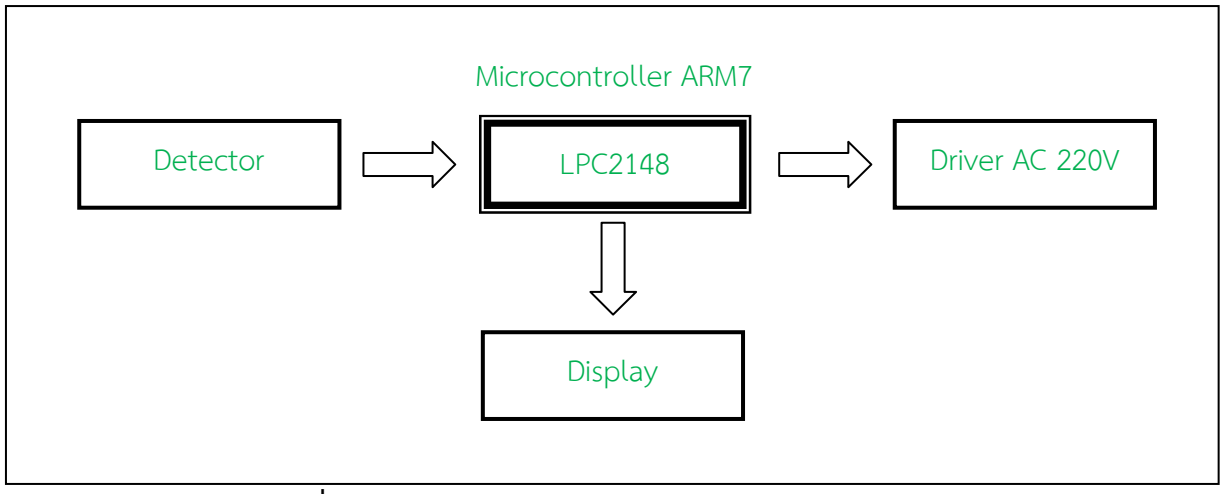
รูปที่.3.3 คือวงจรสำหรับควบคุมการเปิดและปิดสัญญาณไฟจราจรมีขนาด 8 x 23 เซนติเมตร สัญญาณไฟจราจรที่ติดตั้งอยู่บนถนนนั้นจะใช้แรงดันไฟฟ้า 220VAC ดังนั้นจึงจำเป็นต้องมีวงจรสำหรับการเปิดปิดสัญญาณไฟซึ่งการเปิดปิดสัญญาณไฟนั้นจะสั่งงานจากวงจรควบคุมโดยตรง ในวงจรวงจรสำหรับควบคุมการเปิดและปิดสัญญาณไฟจราจรจะสามารถสั่งงานสัญญาณไฟได้สองทิศทางและมีวงจรสำหรับการตรวจสอบสัญญาณไฟเขียว เพื่อจะนำมาเป็นอินพุตให้กับวงจรป้องกัน

3.1.1 ไมโครคอนโทลเลอร์ ARM7 LPC2148

การควบคุมการทำงานหลักของระบบควบคุมสัญญาณไฟจราจรจะใช้ไมโครคอนโทลเลอร์ ARM7 LPC2148 เป็นหน่วยประมวลผลหลัก ซึ่งเป็นบอร์ดไมโครคอนโทรเลอร์ในตระกูล ARM7TDMI-S Core สามารถเลือกการประมวลผลเป็นแบบ 16 บิต หรือ 32 บิตก็ได้ ซึ่งในงานวิจัยนี้ได้เลือกใช้การประมวลผลแบบ 32 บิต มีขนาด 64 Pin ใช้พลังงานต่ำที่มีแรงดันในการประมวลผลที่ 3.3V ภายในบอร์ดจะมี Connector RS232 (UART0) จำนวน 1 Port สำหรับทำการ Download Hex File หรือใช้งานในการสื่อสาร RS232 ในโปรแกรม Application ที่เขียนขึ้นเอง บอร์ดไมโครคอนโทลเลอร์ ARM7 LPC2148 ที่อยู่ในบอร์ดควบคุมสัญญาณไฟจราจรจะแสดงดังรูปที่.3.4



รูปที่.3.4 ไมโครคอนโทลเลอร์ ARM7 LPC2148

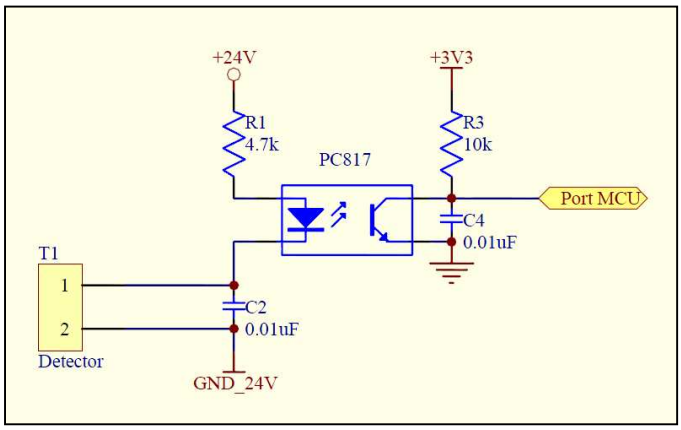


รูปที่.3.5 การออกแบบไมโครคอนโทลเลอร์ ARM7 LPC2148

จากรูปที่.3.5 จะเห็นได้ว่ามีอยู่ 3 ส่วนหลักที่เชื่อมต่อกับไมโครคอนโทลเลอร์ ARM7 LPC2148 เพื่อทำหน้าที่ควบคุมสัญญาณไฟจราจรคือ ส่วนของวงจรตรวจจับยานพาหนะ, ส่วนการแสดงผลและส่วนสำหรับเปิดปิดสัญญาณไฟจราจร

3.1.2 วงจรเชื่อมต่อตัวตรวจจับยานพาหนะ (Detector Socket)

จากคุณสมบัติของตัวตรวจยานพาหนะที่มีระดับของการประมวลผลสัญญาณที่ต่างกันโดยที่ไมโครคอนโทลเลอร์ ARM7 LPC2148 จะมีระดับการประมวลผลสัญญาณที่ 3.3VDC และตัวตรวจจับยานพาหนะประมวลผลสัญญาณที่ 24VDC ดังนั้น การเชื่อมต่อของวงจรตรวจจับยานพาหนะจึงจำเป็นต้องใช้วงจรแปลงระดับสัญญาณ เพื่อให้สามารถทำงานรวมกันได้ โดยการออกแบบนั้นจะใช้ Optoisolators เบอร์ PC817 เพื่อแปลงระดับสัญญาของตัวตรวจจับยานพาหนะ ให้สามารถใช้งานร่วมกันกับไมโครคอนโทลเลอร์ ARM7 LPC2148 ซึ่งจะมีการออกแบบวงจรตามรูปที่.3.6

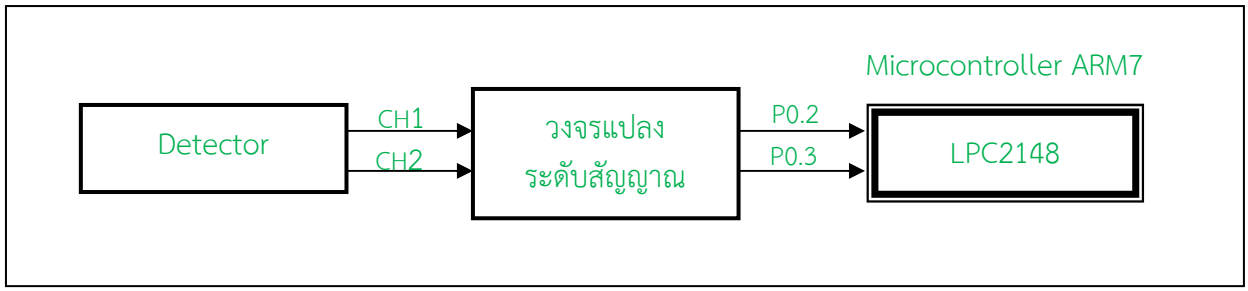


รูปที่.3.6 การออกแบบวงจรแปลงระดับสัญญาณ

จากรูปที่.3.6 จะเห็นว่าจุดที่เชื่อมต่อกับตัวตรวจจับยานพาหนะคือ Connector 2 Pin (T1) จากที่ตัวตรวจจับยานพาหนะที่มีพอร์ตการเชื่อมต่อสำหรับส่งสัญญาณ 2 พอร์ต คือ Open Collector และ Open Emitter เพื่อให้สามารถนำไปใช้งานได้สะดวก เมื่อตัวตรวจจับยานพาหนะไม่พบยานพาหนะไมโครคอนโทลเลอร์จะมีสัญญาณเป็น Logic 1 แต่เมื่อตัวตรวจจับยานพาหนะสามารถตรวจพบยานพาหนะพอร์ต Open Collector และ Open Emitter จะเกิดการทำงาน ทำให้เกิดมีกระแสไหลทางด้านแรงดัน 24V ส่งผลให้ Optoisolators ทำงาน ทำให้ไมโครคอนโทลเลอร์จะมีสัญญาณเป็น Logic 0

รูปที่.3.7 วงจรเชื่อมต่อตัวตรวจจับยานพาหนะ (Detector Socket)

ตัวตรวจจับยานพาหนะสามารถใช้งานได้ 2 Channel ที่แยกอิสระต่อกัน ดังนั้นจึงออกแบบให้ไมโครคอนโทลเลอร์ ARM7 LPC2148 ให้สามารถรับข้อมูลได้ทั้ง 2 Channel โดยที่ให้พอร์ต P0.2 เชื่อมต่อกับ Channel 1 และพอร์ต P0.3 เชื่อมต่อกับ Channel 2 และเชื่อมต่อทั้ง 2 Channel ผ่านวงจรแปลงระดับสัญญาดังรูปที่.3.6 ดังนั้นเราสามารถออกแบบวงจรที่เชื่อมต่อกันระหว่างไมโครคอนโทลเลอร์ ARM7 LPC2148 และตัวตรวจจับยานพาหนะได้ดังรูปที่.3.8



รูปที่.3.8 การเชื่อมต่อระหว่างไมโครคอนโทลเลอร์ ARM7 LPC2148
และตัวตรวจจับยานพาหนะ

3.1.3 วงจรแสดงผล (Display)

วงจรแสดงผลจะประกอบด้วย LED ที่ใช้แสดงสถานะต่าง ๆ ของระบบควบคุมสัญญาณไฟจราจร (Controller) วงจร LED ที่ใช้แสดงสถานะการทำงานของระบบ จะเชื่อมต่อกับไมโครคอนโทลเลอร์ ARM7 LPC2148 ซึ่งเป็นหน่วยประมวลผลหลัก โดยการทำงานจะเป็นแบบ Active Low การเชื่อมต่อ LED กับไมโครคอนโทลเลอร์เพื่อใช้แสดงสถานะต่าง ๆ ของระบบ โดยจะมีการเชื่อมต่อตามตารางที่ 3.1

ตารางที่ 3.1 LED แสดงสถานะการทำงานและการเชื่อมต่อพอร์ตของไมโครคอนโทลเลอร์

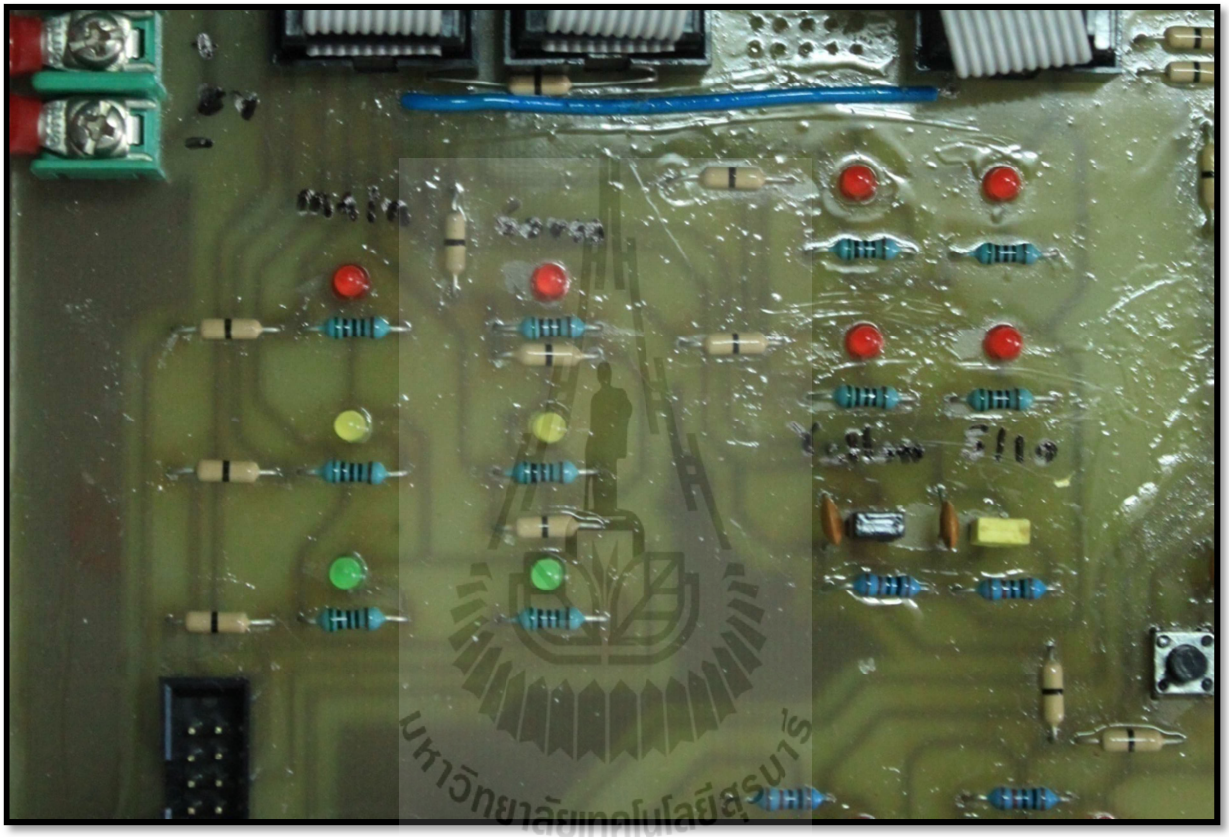
การเชื่อมต่อของไมโครคอนโทลเลอร์	
พอร์ตของไมโครคอนโทลเลอร์	แสดงสถานะ
P0.4	เมื่อตรวจพบยานพาหนะ
P0.5	เมื่อใช้โหมด Fast/Slow **
P0.6	เมื่อตัวตรวจจับยานพาหนะ (Detector) **Channel 1** เกิดความ
P0.7	ผิดพลาด
P1.16	เมื่อตัวตรวจจับยานพาหนะ(Detector) **Channel 2** เกิดความ
P1.17	ผิดพลาด
P1.18	แสดงสถานะ ไฟเขียว ของเส้นทางเอก
P1.19	แสดงสถานะ ไฟเหลือง ของเส้นทางเอก
P1.20	แสดงสถานะ ไฟแดง ของเส้นทางเอก
P1.21	แสดงสถานะ ไฟเขียว ของเส้นทางโท
	แสดงสถานะ ไฟเหลือง ของเส้นทางโท
	แสดงสถานะ ไฟแดง ของเส้นทางโท

หมายเหตุ โหมด Fast/Slow จะกล่าวถึงในส่วนของการออกแบบซอฟแวร์ (Software)

นอกจากจะมีการแสดงสถานะต่าง ๆ ของส่วนการควบคุมสัญญาณไฟจราจรด้วย LED แล้ว ยังมีส่วนของการเลือกโหมดการทำงานต่าง ๆ ซึ่งจะออกแบบในลักษณะเป็นจุดเชื่อมต่อซึ่งในส่วนของการควบคุมสัญญาณไฟจราจรนั้นมีจุดเชื่อมต่อ 2 จุด คือ 1. สำหรับเลือกเป็นแบบไฟเหลืองกระพริบ 2. สำหรับการเลือกโหมด Fast/Slow โดยจะมีการเชื่อมต่อตามตารางที่ 3.2

ตารางที่ 3.2 แสดงสถานะการเชื่อมต่อพอร์ตของไมโครคอนโทลเลอร์

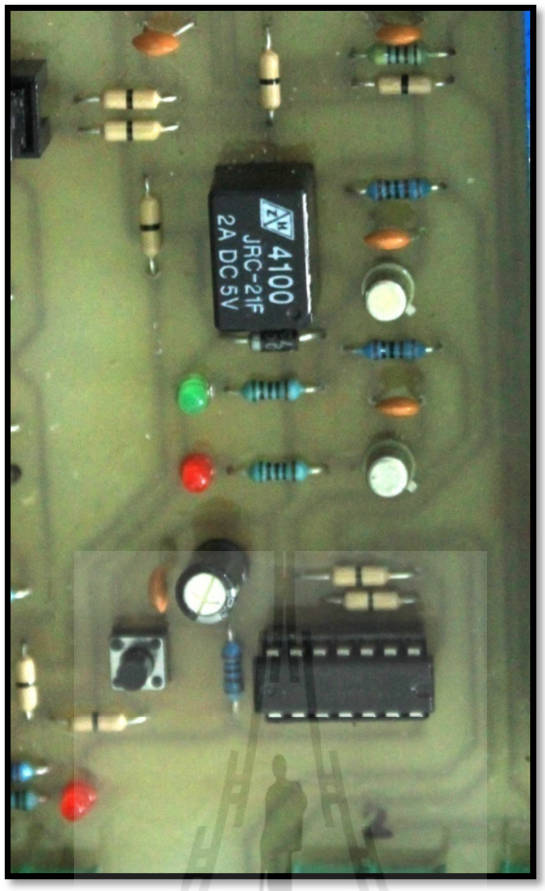
การเชื่อมต่อของไมโครคอนโทลเลอร์	
พอร์ตของไมโครคอนโทลเลอร์	โหมด
P0.16	ไฟเหลืองกระพริบ
P1.23	โหมด Fast/Slow



รูปที่.3.9 วงจรแสดงผล (Display)

3.1.4 วงจรป้องกัน (Protection Circuit)

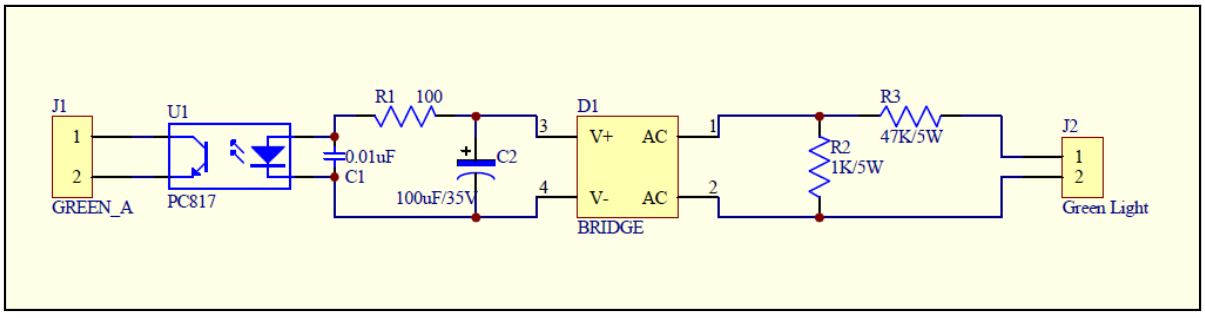
ในการควบคุมสัญญาณไฟจราจรนั้นสำคัญที่สุดคือการให้ทิศทางของสัญญาณไฟเขียวในทิสทางที่ไม่เกิดความขัดแย้งกัน ดังนั้นในส่วนของการควบคุมสัญญาณไฟจราจรนั้นจะมีวงจรป้องกันไม่ให้เกิดสัญญาณไฟเขียวในทิศทางที่ขัดแย้งกัน โดยในระบบควบคุมสัญญาณไฟจราจรนี้จะมีทิศทางของสัญญาณไฟเขียวเพียง 2 ทิศทาง จึงได้ออกแบบวงจรป้องกันเพื่อไม่ให้เกิดไฟเขียวพร้อมกันทั้ง 2 ทิศทางโดยการออกแบบวงจรต่าง ๆ ดังนี้



รูปที่.3.10 วงจรป้องกัน (Protection Circuit)

1. **วงจรตรวจสอบสถานะของไฟจราจร**

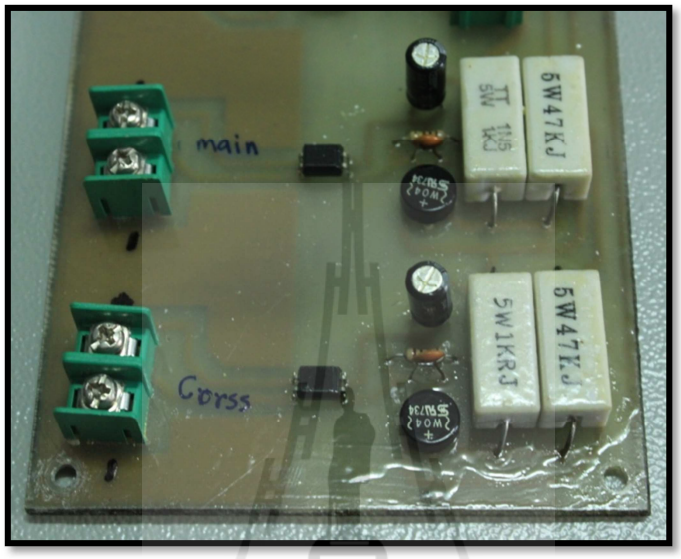
ในส่วนนี้จะเป็นวงจรที่ใช้ตรวจสอบสถานะของไฟจราจรที่แสดงอยู่บนท้องถนน โดยสัญญาณไฟดังกล่าวเป็นไฟกระแสสลับ 220VAC และได้ออกแบบวงจรเพื่อตรวจสอบสัญญาณไฟจราจรว่ามีสถานะเปิดหรือปิดอยู่ในขณะนี้ ซึ่งจะออกแบบวงจรตามรูปที่.3.11



รูปที่.3.11 ออกแบบวงจรสำหรับตรวจสอบสัญญาณไฟจราจร

จากรูปที่.3.11 จะเห็นว่าสัญญาณไฟจราจรที่จะเชื่อมต่อกับ Connector J2 ซึ่งสัญญาณไฟที่เชื่อมต่อนั้นเป็นไฟกระแสสลับแรงดัน 220VAC ซึ่งใช้กับหลอดไฟของสัญญาณไฟจราจรที่อยู่บนท้องถนน เมื่อมี

การเปิดสัญญาณไฟ จะทำให้เกิดมีกระแสไฟเข้ามาที่ Connector J2 และใช้วงจรแบ่งแรงดันเพื่อลดขนาดของแรงดันลงให้เหลือ 5 VAC จากนั้นแปลงจากกระแสไฟสลับให้เป็นกระแสตรงโดยการนำแรงดัน 5 VAC ผ่านวงจรเรียงกระแสแบบเต็มคลื่น (Bridge) เพื่อให้ได้ไฟกระแสตรง 5 VDC โดยไฟกระแสตรงที่ได้จะเป็นแหล่งจ่ายไฟให้กับ Optoisolators เบอร์ PC817 เพื่อให้ Optoisolators ทำงาน ซึ่งในวงจรนี้เป็นวงจรที่ออกแบบให้เป็น Open Collector และ Open Emitter เพื่อความสะดวกนำไปใช้งานต่อไป

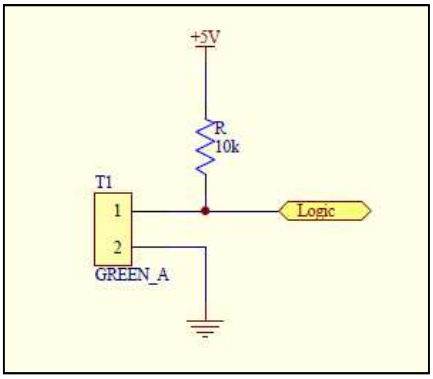


รูปที่.3.12 วงจรตรวจสอบสัญญาณไฟจราจร

2. วงจรป้องกันการเกิดสัญญาณไฟเขียว 2 ทิศทาง

ในส่วนนี้คือวงจรที่ใช้ป้องกันการเกิดสัญญาณไฟเขียวพร้อมกันใน 2 ทิศทาง ซึ่งในวงจรได้ออกแบบเพื่อควบคุมแหล่งจ่ายไฟของส่วนควบคุมการเปิดและปิดสัญญาณไฟจราจร เมื่อตรวจสอบพบว่ามีสัญญาณไฟเขียวพร้อมกันทั้ง 2 ทิศทาง จะทำการปิดแหล่งจ่ายไฟของส่วนควบคุมการเปิดและปิดสัญญาณไฟจราจรเพื่อไม่ให้สัญญาณไฟที่อยู่บนท้องถนนสามารถแสดงได้ ซึ่งการออกแบบส่วนต่าง ๆ ดังนี้

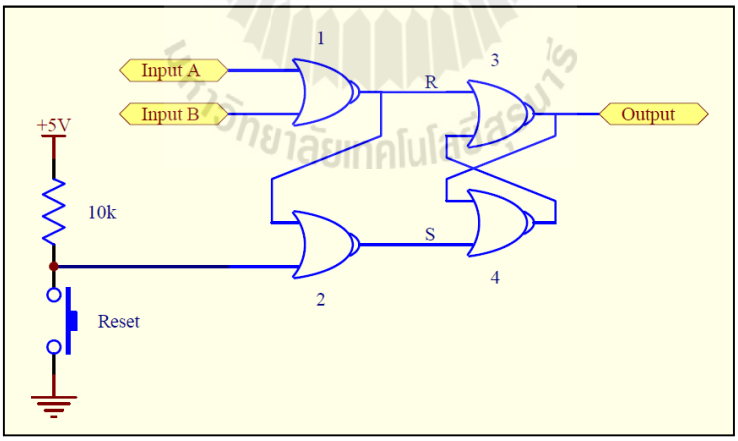
การรับข้อมูลจากวงจรตรวจสอบสถานะของไฟจราจร



รูปที่.3.13 วงจรอินพุตของวงจรป้องกันการเกิดสัญญาณไฟเขียว 2 ทิศทาง

จากรูปที่.3.13 คือวงจรอินพุตของวงจรป้องกันการเกิดสัญญาณไฟเขียว 2 ทิศทาง โดย Connector T1 จะรับสัญญาณจากวงจรตรวจสอบสถานะของไฟจราจร ซึ่งจะเป็น Open Collector และ Open Emitter ของ Optoisolators เบอร์ PC817 เมื่อสัญญาณไฟจราจรที่อยู่บนท้องถนนมีการทำงานจะทำให้ Optoisolators ทำงานจะทำให้ได้ Logic 0 แต่เมื่อสัญญาณไฟจราจรที่อยู่บนท้องถนนไม่มีการทำงานจะทำให้ได้ Logic 1

การออกแบบทางดิจิตอล



รูปที่.3.14 การออกแบบวงจรดิจิตอล

จากรูปวงจรที่ 3.14 เป็นการออกแบบดิจิตอลสำหรับการป้องกันการเกิดสัญญาณไฟจราจรพร้อมกันทั้ง 2 ทิศทาง จากวงจรจะเห็นว่ามี Input อยู่ด้วยกัน 3 Input คือ Input A สัญญาณจากไฟจราจรในทิศทางที่ 1 Input B สัญญาณจากไฟจราจรในทิศทางที่ 2 และ Input ที่ได้จาก สวิตซ์ Reset และมี 1 Output สำหรับควบคุมแหล่งจ่ายของส่วนควบคุมการเปิดและปิดสัญญาณไฟจราจร (Traffic Light Control)

ซึ่งจากวงจรดังกล่าว NOR Gate 3 และ 4 จะทำหน้าที่เหมือนกับ RS Flip flop ซึ่งจะมีตารางการทำงานของ RS Flip flop ตามตารางที่ 3.3

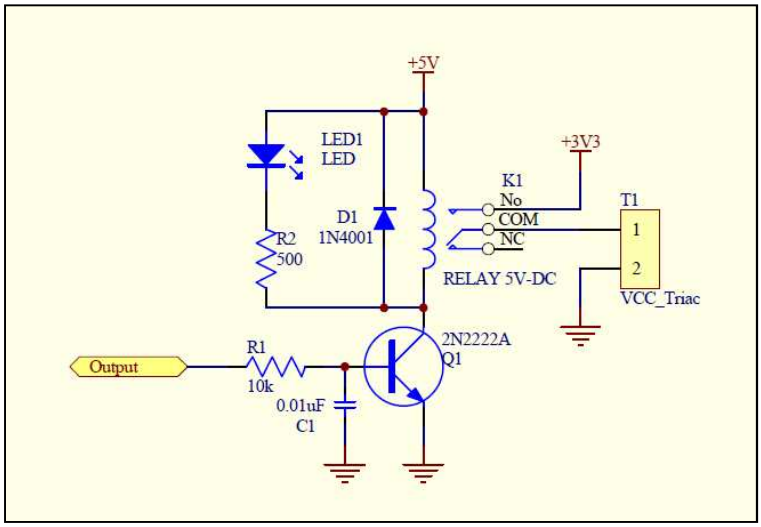
ตารางที่ 3.3 แสดงการทำงานของการ RS Flip flop

RS Flip flop		
S	R	Q
0	0	No Change
0	1	0
1	0	1
1	1	0

Logic ของ Output ที่เกิดขึ้นจะเป็นตัวกำหนดการเปิดและปิดแหล่งจ่ายไฟให้กับส่วนควบคุมการเปิดและปิดของสัญญาณไฟจราจร โดยที่ Output มี Logic 1 จะทำการจ่ายไฟให้กับวงจรควบคุมการเปิดและปิดของสัญญาณไฟจราจรซึ่งทำให้สัญญาณไฟจราจรที่อยู่บนท้องถนนสามารถแสดงได้ แต่ถ้าหาก Output มี Logic 0 จะปิดแหล่งจ่ายไฟวงจรควบคุมการเปิดและปิดของสัญญาณไฟจราจรทำให้ไม่สามารถแสดงสัญญาณไฟจราจรได้ จากรูปที่.3.14 เริ่มต้นเมื่อสัญญาณไฟจราจรทั้ง 2 ทิศทางไม่มีการทำงาน (Input A และ Input B มี Logic 1) , Output มี Logic 1 และสวิตซ์ Reset ยังไม่ถูกกด ในสภาวะการทำงานที่เป็นปกติของการเปิดสัญญาณไฟเขียวที่เปิดไม่พร้อมกัน (Input A และInput B ไม่ได้มี Logic 0 พร้อมกัน) จากตารางการทำงานของ RS Flip flop จะเห็นว่าเมื่อ S ถูกกำหนดให้เป็น Logic 0 และ R ถูกกำหนดให้เป็น Logic 0 จะไม่เกิดการเป็นแปลงสถานะของ Output แต่ถ้าหากเกิดมีการทำงานของไฟเขียวทำงานพร้อมกัน 2 ทิศทาง (Input A และInput B มี Logic 0 พร้อมกัน) จะทำให้ R มี Logic 1 ซึ่งจะทำให้ Output มี Logic 0 นั้นคือจะปิดแหล่งจ่ายไฟให้กับส่วนควบคุมการเปิดและปิดของสัญญาณไฟจราจรทำให้ไฟสัญญาณจราจรที่อยู่บนท้องถนนไม่มีการทำงาน และเมื่อสัญญาณไฟเขียวทั้ง 2 ทิศทางไม่มีการทำงาน จะทำให้ Input A และInput B กลับไปมี Logic 1 ทั้งคู่ และจะทำให้ R มี Logic 0 แต่ Output ยังคงเป็น Logic 0 ดังนั้นในขณะนี้จะได้ว่า R มี Logic 0 , S มี Logic 0 เมื่อพิจารณาจากตารางการทำงานของ RS Flip flop พบว่าจะไม่มีการเปลี่ยนแปลง Output แต่เมื่อต้องการให้ Output มี Logic 1 จะต้องทำให้ S มี Logic 1 ซึ่งสามารถทำได้โดยการกดสวิตซ์ Reset ซึ่งจากรูปที่.3.14 เมื่อมีการกด Reset จะทำให้ S มี Logic เป็น 1 ซึ่งจะทำให้ Output มี Logic 1 ตามไปด้วย

การเปิดและปิดแหล่งจ่ายไฟ

จากที่ได้กล่าวมาแล้วว่า Output ของวงจรป้องกันจะทำหน้าที่ควบคุมการเปิดและปิดแหล่งจ่ายไฟของส่วนควบคุมการเปิดและปิดของสัญญาณไฟจราจร ดังนั้นจะมีการการออกแบบวงจรสำหรับการเปิดและปิดแหล่งจ่ายไฟดังรูปที่.3.15

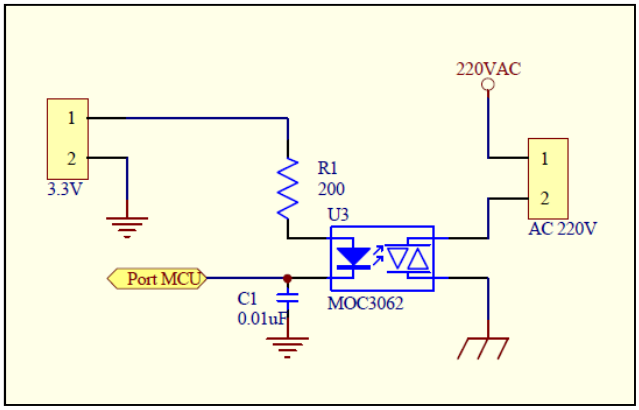


รูปที่.3.15 วงจรควบคุมการเปิดและปิดแหล่งจ่ายไฟ

จากรูปที่.3.15 จะเป็นว่าเมื่อ Output มี Logic 1 จะทำให้ทรานซิสเตอร์ทำงานเกิดกระแสไหลส่งผลให้ Relay ทำงานเพื่อเปิดแหล่งจ่ายไฟกระแสตรง 3.3V ให้กับส่วนควบคุมการเปิดและปิดของสัญญาณไฟจราจรเพื่อให้สัญญาณไฟจราจรที่อยู่บนท้องถนนสามารถทำงานได้ตามปกติ แต่เมื่อ Output มี Logic 0 จะทำให้ Relay ไม่ทำงานและแหล่งจ่ายไฟจะไม่ถูกจ่ายให้กับส่วนควบคุมการเปิดและปิดของสัญญาณไฟจราจรทำให้ไฟสัญญาณจราจรไม่สามารถทำงานได้

3.1.5 วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร (Traffic Light Control)

สัญญาณไฟจราจรที่อยู่บนท้องถนนนั้นจะทำงานที่แรงดัน 220VAC ดังนั้นจึงจะเป็นต้องมีวงจรควบคุมการเปิดและปิดสัญญาณไฟจราจรเพื่อให้ไมโครคอนโทลเลอร์สามารถสั่งงานเปิดและปิดสัญญาณไฟจราจรได้โดยตรง ในระบบควบคุมสัญญาณไฟจราจรนั้นจะมีการป้องกันไม่ให้เกิดสัญญาณไฟเขียวที่ทำให้เกิดความขัดแย้งกัน ดังนั้นแหล่งจ่ายไฟที่จ่ายให้กับวงจรควบคุมการเปิดและปิดของสัญญาณไฟจราจร จะได้มาจากวงจรป้องกันในหัวข้อ 3.1.5 การออกแบบวงจรสำหรับเปิดและปิดสัญญาณไฟจราจรจะเป็นดังรูปที่.3.16



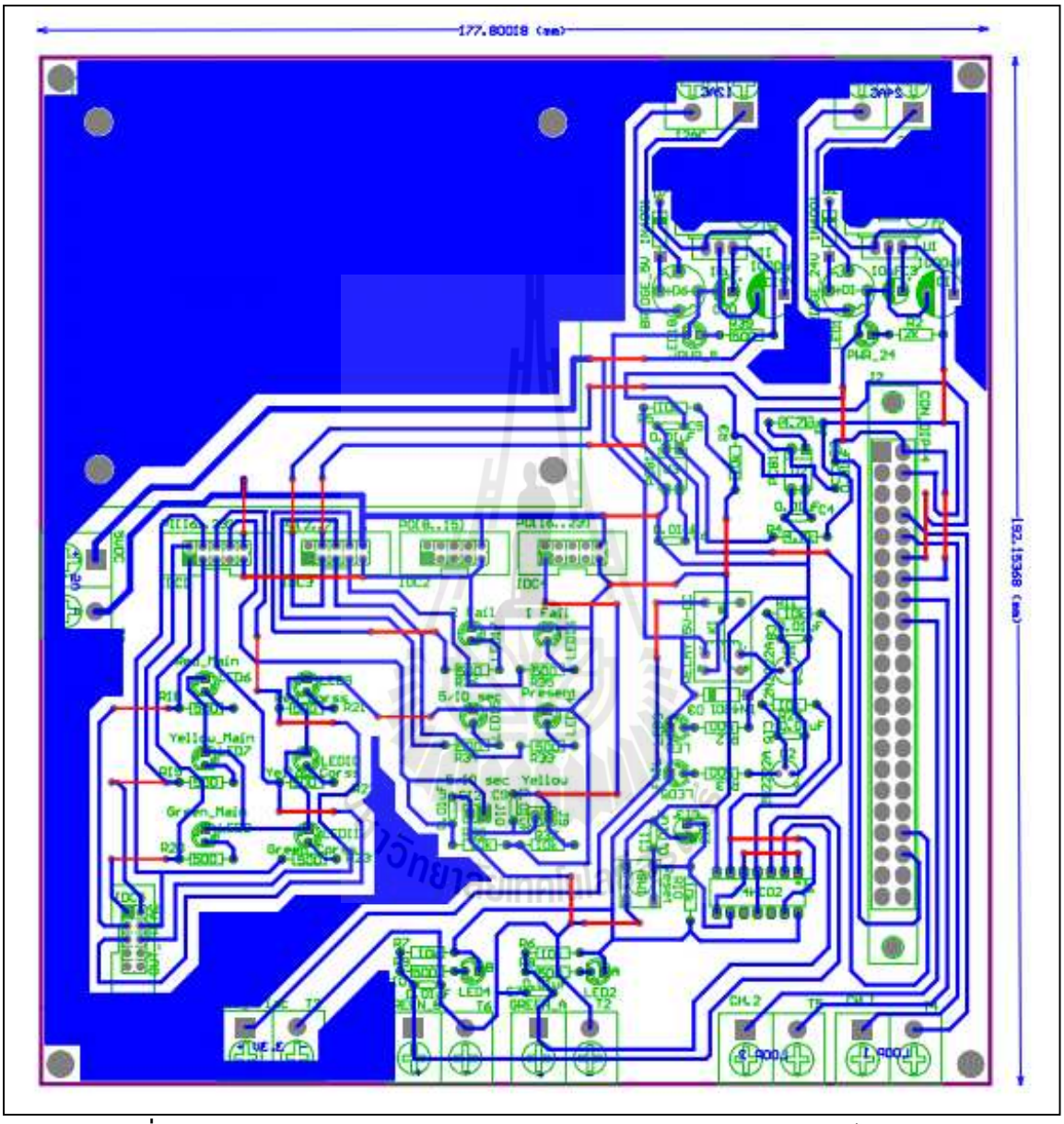
รูปที่.3.16 วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร

จากรูปที่.3.16 จะเห็นว่าการควบคุมการเปิดและปิดของสัญญาณไฟจราจรนั้นจะใช้ Optoisolators Triac Driver เบอร์ MOC3062 เพื่อให้แปลงระดับแรงดันจาก 3.3 VDC ให้เป็น 220VAC ได้ โดยเมื่อไมโครคอนโทรเลอร์มี Logic 0 จะเกิดกระแสไหลทางด้าน 3.3VDC ทำให้เกิดการทำงานของ Optoisolators Triac Driver จะสามารถเปิดสัญญาณไฟจราจรได้ แต่ถ้าไมโครคอนโทรเลอร์มี Logic 1 Optoisolators Triac Driver จะไม่ทำงานและจะทำให้ไฟสัญญาณจราจรถูกปิดตามไปด้วย

มหาวิทยาลัยเทคโนโลยีสุรนารี

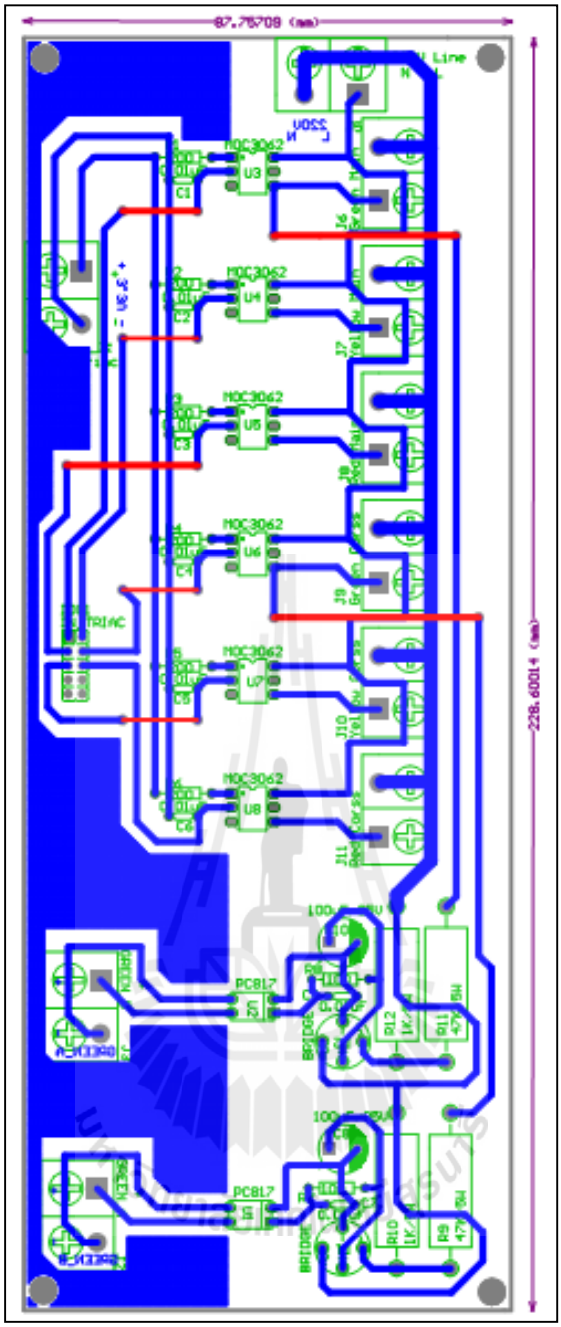
3.1.6 ออกแบบวงจรรวมของส่วนควบคุมสัญญาณไฟจราจร

จากส่วนต่างๆ ที่ได้กล่าวมาแล้วนั้นจะนำมาออกแบบวงจรรวมเพื่อให้ได้บอร์ดควบคุมสัญญาณไฟจราจร โดยจะมีการออกแบบวงจรต่าง ๆ ดังนี้



รูปที่.3.17 ลักษณะการวางตำแหน่งของอุปกรณ์ของบอร์ดควบคุมสัญญาณไฟจราจร

จากรูปที่.3.17 จะเห็นได้ว่ามีการจัดวางตำแหน่งของส่วนต่างๆ ดังภาพ ซึ่งขนาดของบอร์ดควบคุมสัญญาณไฟจราจรจะมีขนาด 18 x 20 เซนติเมตร

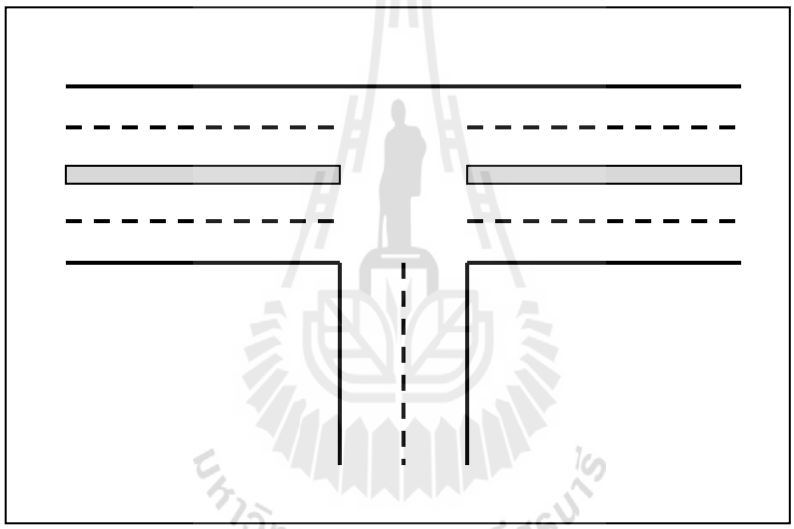


รูปที่.3.18 ลักษณะการวางตำแหน่งของอุปกรณ์
วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร

จากรูปที่.3.18 จะเห็นได้วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจรสามารถที่จะสั่งงานสำหรับการเปิด-ปิดสัญญาณไฟจราจรได้ทั้งหมด 6 ช่อง ซึ่งบอร์ดนี้นำมาใช้งานกับแยกการจราจรที่มีลักษณะสามแยกการจราจร และลักษณะของบอร์ดจะมีขนาด 23 x 9 เซนติเมตร

3.2 ส่วนของการออกแบบซอร์ฟแวร์ (Software) ของส่วนควบคุมสัญญาณไฟจราจร (Controller)

ในส่วนของการออกแบบซอร์ฟแวร์ของระบบควบคุมสัญญาณไฟจราจรนั้น จะออกแบบตามแยกการจราจรที่ติดตั้ง โดยจะเป็นการควบคุมสัญญาณไฟจราจรแบบกึ่งอัตโนมัติซึ่งจะติดตั้งในทางแยกจราจรที่มีทางเอกและทางโท โดยติดตั้งตัวตรวจจับยานพาหนะไว้ที่ทางโทของทางแยกการจราจร โดยในสภาพการจราจรที่เป็นปกติ (ไม่มียานพาหนะในทางโท) ทางเอกจะได้สัญญาณไฟเขียวโดยตลอด และทางโทจะมีสัญญาณไฟแดงตลอดเวลาเช่นกัน เว้นแต่จะมียานพาหนะมารออยู่ที่ทางโทจึงจะทำให้ที่ทางโทรได้สัญญาณไฟเขียว เพื่อให้ยานพาหนะในทางโทผ่านได้ ซึ่งในงานวิจัยนี้ได้ติดตั้งระบบควบคุมสัญญาณไฟจราจรในแยกการจราจรที่มีลักษณะดังรูปที่.3.19



รูปที่.3.19 ลักษณะแยกการจราจรที่ติดตั้งระบบควบคุมสัญญาณไฟจราจร

ในระบบควบคุมสัญญาณไฟจราจรแบบกึ่งอัตโนมัติจำเป็นจะต้องมียานพาหนะมากระตุ้นหรือมารอที่ทางโทจึงจะเกิดการเปลี่ยนแปลงสัญญาณไฟจราจร ซึ่งจะต้องมีช่วงระยะเวลาของยานพาหนะที่มารอที่ทางโทก่อนที่จะเกิดการเปลี่ยนสัญญาณไฟซึ่งระยะเวลาของการรอดังกล่าวในระบบการควบคุมสัญญาณไฟจราจรได้ทำการออกแบบเป็นโหมดการทำงาน 2 โหมด คือ 1. โหมด Fast คือระยะเวลาของการรอ 5 วินาที 2. โหมด Slow คือระยะเวลาของการรอ 10 วินาที นอกจากระยะเวลาของการรอในทางโทแล้ว โหมดการทำงาน Fast และ Slow จะควบคุมระยะเวลาของสัญญาณไฟเขียวที่ทางเอก นั้นคือหลังจากที่ทางโทได้สัญญาณไฟเขียวแล้วรอบต่อไปที่จะได้สัญญาณไฟเขียวอีกนั้นจะต้องรอเป็นระยะเวลา 30 วินาที สำหรับโหมด Fast และ 60 วินาที สำหรับโหมด Slow ดังนั้นสามารถสรุปการทำงานของระบบควบคุมสัญญาณไฟจราจรได้ดังนี้

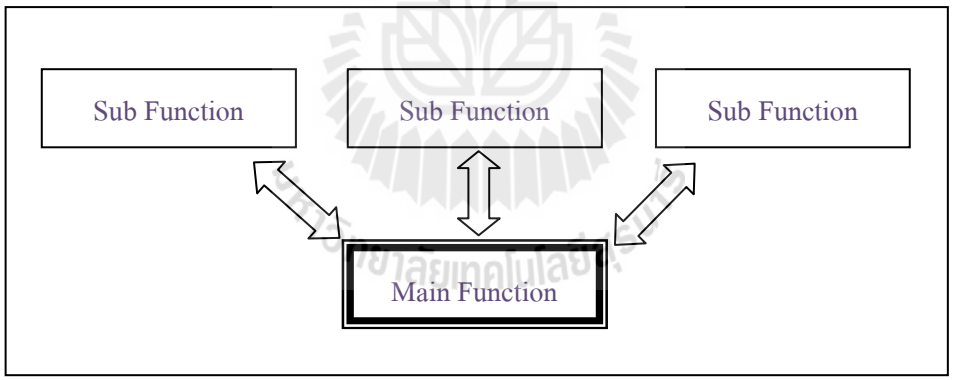
โหมด Fast : ระยะเวลารอของยานพาหนะในทางโท 5 วินาที และจะได้ไฟเขียวรอบต่อไปหลังจาก 30 วินาที

โหมด Slow : ระยะเวลารอของยานพาหนะในทางโท 10 วินาที และจะได้ไฟเขียวรอบต่อไปหลังจาก 60 วินาที

ในระบบควบคุมสัญญาณไฟจราจรจะใช้ไมโครคอนโทรเลอร์ ARM7 LPC2148 เป็นตัวควบคุมการทำงานหลักโดยจะต้องมีซอฟต์แวร์ที่ใช้ในการเขียนคำสั่งควบคุม ดังนั้นในงานวิจัยนี้ได้เลือกใช้ภาษาซีในการเขียนคำสั่ง โดยเขียนคำสั่งควบคุมผ่านโปรแกรม Keil uVision3 เนื่องจากภาษาซีเป็นภาษาที่มีโครงสร้างง่ายต่อการทำความเข้าใจและสามารถปรับปรุงพัฒนา นอกจากนั้นภาษาซียังเป็นภาษามาตรฐานไม่ขึ้นกับฮาร์ดแวร์ (ไมโครคอนโทรเลอร์) มีความยืดหยุ่นในการใช้งานกับไมโครคอนโทรลเลอร์ตระกูลอื่นได้

การทำงานของระบบควบคุมสัญญาณไฟจราจร

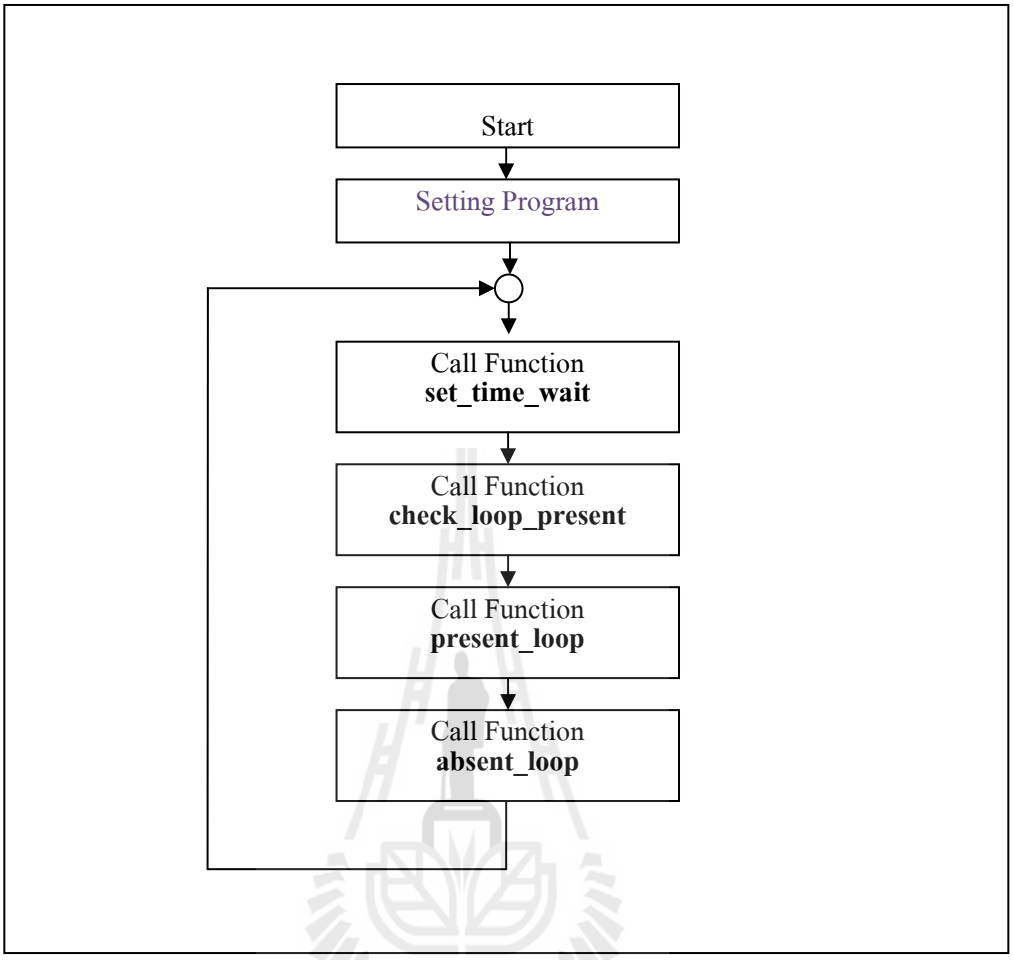
ระบบการควบคุมสัญญาณไฟจราจรแบบกึ่งอัตโนมัติที่นำมาใช้เป็นรูปแบบการทำงานหลักในงานวิจัยนี้ โดยมีฟังก์ชันการทำงานหลัก (Main Function) เป็นส่วนควบคุมการทำงานทั้งหมดของระบบ ซึ่งฟังก์ชันการทำงานหลัก (Main Function) ของระบบควบคุมสัญญาณไฟจราจรจะประกอบด้วยการทำงานของฟังก์ชันย่อย ๆ ดังรูปที่.3.20 โดยที่ฟังก์ชันการทำงานหลักจะทำการเรียกใช้งานในแต่ละฟังก์ชัน เพื่อให้ระบบเกิดความไม่ซับซ้อน และสามารถหาจุดบอกพร่องและแก้ไขได้ง่าย



รูปที่.3.20 รูปแบบการทำงานของฟังก์ชันการทำงานหลัก (Main Function)

ซึ่งการทำงานของฟังก์ชันต่างๆ สามารถอธิบายดังนี้

3.2.1 Main Function



รูปที่.3.21 การทำงานของ Main Function

จากรูปที่.3.21 เป็นขั้นตอนการทำงานของ Main Function โดยจะเริ่มต้นจากการตั้งค่าต่าง ๆ ของไมโครคอนโทลเลอร์ ARM7 LPC2148 ซึ่งดังนี้

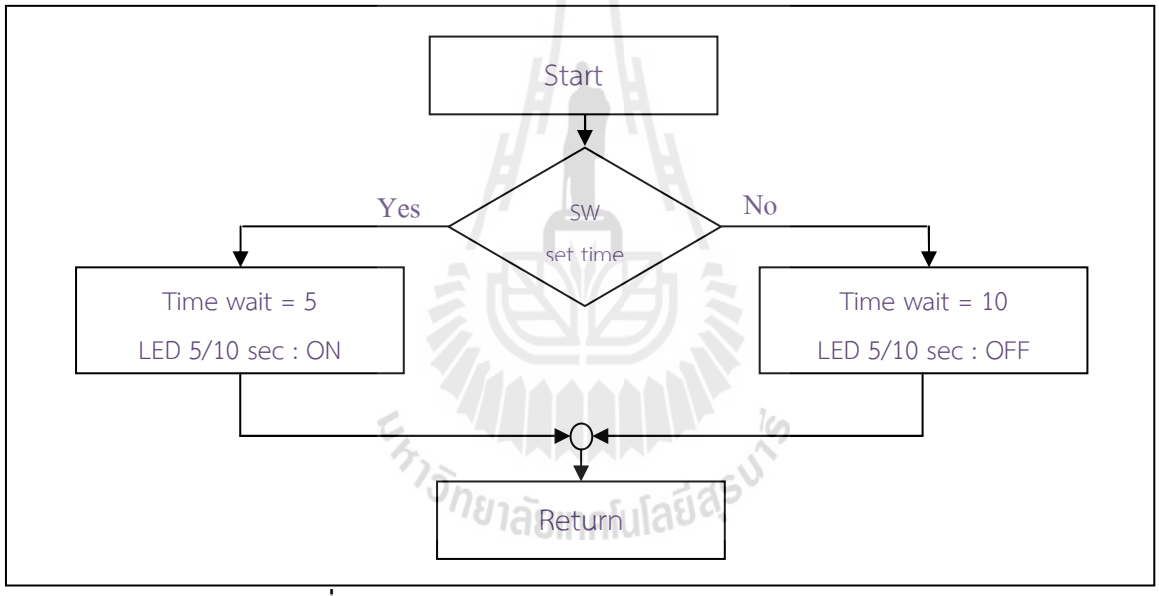
- เปิดใช้งานระบบเวลาจริง (Real Time Clock) ซึ่งจะใช้งานแบบอินเตอร์รัป (Interrupt) ที่เกิดการอินเตอร์รัปทุก ๆ 1 วินาที เพื่อนำมาใช้งานในการนับเวลาของการควบคุมสัญญาณไฟจราจร
- เปิดใช้งานอินเตอร์รัปภายนอก (External Interrupt 0) ที่พอร์ต P0.16 สำหรับการเปิดใช้งานสัญญาณไฟเหลืองกระพริบ

หลังจากตั้งค่าต่าง ๆ ของไมโครคอนโทลเลอร์ ARM7 LPC2148 แล้วจะทำการตั้งค่าของระยะเวลาการรอของยานพาหนะในทางโทหรือการเลือกโหมด Fast / Slow นั้นเอง หลังจากเลือกโหมดการทำงานแล้วจะทำการเรียกใช้งาน Check Loop Present Function เพื่อตรวจสอบยานพาหนะในทางโทว่ามียานพาหนะมารออยู่หรือไม่ ละจะส่งผลของการตรวจสอบยานพาหนะที่ได้ให้กับ Present Loop Function ถ้ามียานพาหนะมารอที่ทางโท จะให้สัญญาณไฟเขียวเพื่อให้ยานพาหนะสามารถผ่านไปได้ แต่ถ้าไม่มีก็จะไม่

เกิดการทำงานของฟังก์ชันนี้และต่อจากนั้นจะทำเรียกใช้งาน Absent Loop Function เพื่อตรวจสอบว่าไม่มียานพาหนะมารออยู่ที่ทางโทจริงจะทำการเปลี่ยนแปลงสัญญาณไฟจราจรโดยให้สัญญาณไฟเขียวที่ทางเอก แต่ถ้าหากพบว่ายังมียานพาหนะรออยู่ก็จะยังคงให้สัญญาณไฟเขียวต่อไปจนกว่าจะเกินเวลาที่กำหนดเอาไว้

3.2.2 Set Time Wait Function

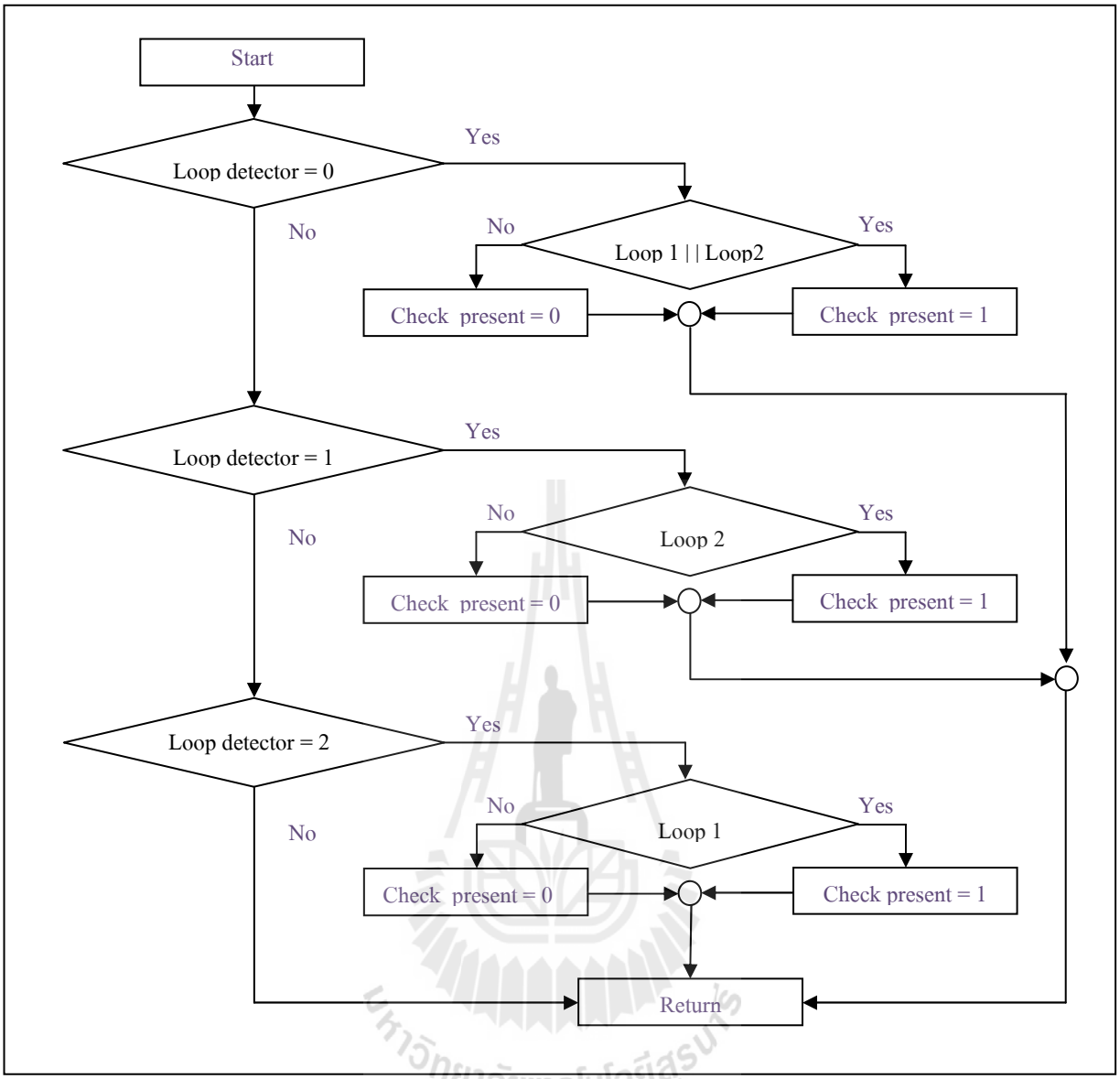
จากที่ได้กล่าวมาแล้วว่า ในระบบควบคุมสัญญาณไฟจราจรในงานวิจัยนี้ได้ออกแบบระยะเวลาของยานพาหนะที่มารอในทางโทแบ่งออกเป็น 2 โหมด คือ โหมด Fast คือระยะเวลาสำหรับการรอเปลี่ยนสัญญาณไฟเป็น 5 วินาที และโหมด Slow คือระยะเวลาสำหรับการรอเปลี่ยนสัญญาณไฟเป็น 10 วินาที ในฟังก์ชันนี้จะทำการตรวจสอบจุดเชื่อมต่อที่พอร์ต P1.23 เพื่อเป็นตัวเลือกโหมดการทำงานของระบบ โดยเมื่อจุดเชื่อมต่อ เมื่อการเชื่อมต่ออยู่จะตั้งค่า Time_wait = 5 และจะทำให้ LDE ที่พอร์ต P0.5 (LED 5/10 sec) ติดเพื่อแสดงว่าใช้โหมด Fast แต่เมื่อไม่มีการเชื่อมต่อจะตั้งค่า Time_wait = 10 และให้ LDE ที่พอร์ต P0.5 (LED 5/10 sec) ดับเพื่อแสดงว่าใช้โหมด Slow ซึ่งจะสามารถอธิบายการทำงานของฟังก์ชันได้ดังรูปที่.3.22



รูปที่.3.22 การทำงานของ Set Time Wait Function

3.2.3 Check Loop Present Function

Check Loop Present Function เป็นฟังก์ชันที่จะรับสัญญาณจากตัวตรวจจับยานพาหนะ ซึ่งมีทั้งหมด 2 Channel เมื่อตัวตรวจจับยานพาหนะตรวจพบยานพาหนะใน Channel 1 หรือ Channel 2 ฟังก์ชันนี้จะทำการส่งค่ากลับโดยที่จะส่งค่า check_persent = 1 และถ้าตรวจสอบไม่พบยานพาหนะใน Channel 1 และ Channel 2 จะทำการส่งค่า check_persent = 0 นอกจากนี้ในฟังก์ชันตรวจสอบยานพาหนะ (Check Loop Present Function)จะทำการตรวจสอบจาก Channel ที่ไม่เกิดความผิดพลาดเท่านั้น เช่น ถ้าหากตัวตรวจจับยานพาหนะที่ Channel 1 เกิดความผิดพลาดก็จะไม่รับสัญญาณจาก Channel 1 จะรับสัญญาณจาก Channel 2 เท่านั้น ซึ่งจะพิจารณาการเกิดความผิดพลาดจาก Check Detector Function การทำงานของฟังก์ชันตรวจสอบยานพาหนะสามารถอธิบายได้ดังรูปที่.3.23



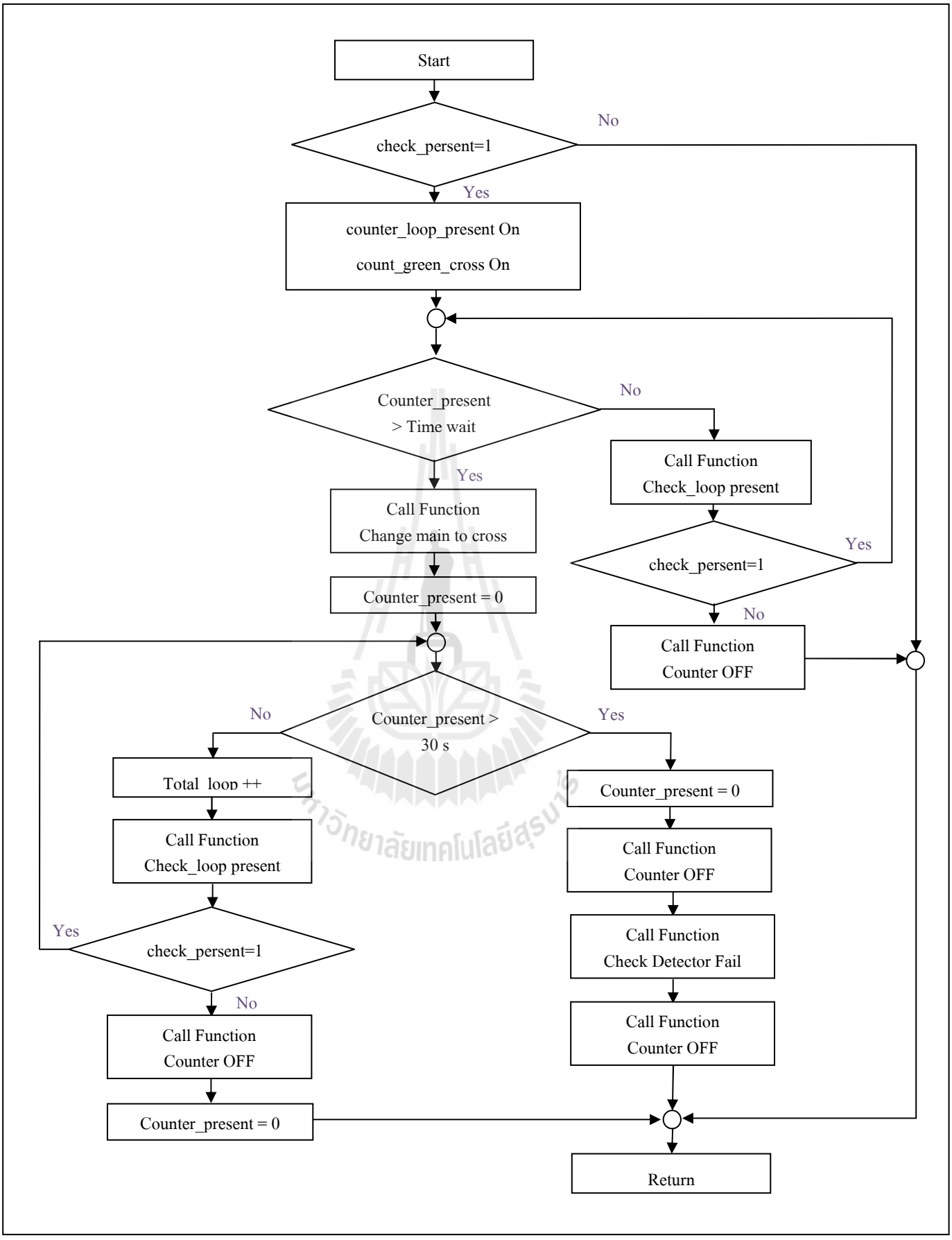
รูปที่.3.23 การทำงานของ Check Loop Present Function

3.2.4 Present Loop Function

ในฟังก์ชันนี้จะทำงานก็ต่อเมื่อตัวตรวจจับยานพาหนะสามรถตรวจพบยานพาหนะในทางโท โดยจะตรวจสอบจาก Present Loop Function โดยเริ่มต้นเมื่อตรวจพบยานพาหนะจะให้เริ่มนับเวลาการรอของยานพาหนะที่เข้ามารอในทางโท โดยเวลาที่รอนั้นจะต้องมากกว่าเวลาที่ถูกตั้งค่าเอาไว้จาก Set Time Wait Function ซึ่งในช่วงระยะเวลาที่รออยู่นั้นจะทำการตรวจสอบยานพาหนะอยู่ตลอดเวลา ว่ายังมียานพาหนะรออยู่หรือไม่ ถ้าไม่พบยานพาหนะรออยู่จะหยุดการนับเวลาและหยุดการทำงานของฟังก์ชัน แต่ถ้าหากยังพบยานพาหนะรออยู่จนกระทั่งหมดช่วงเวลาที่ต้องรอของยานพาหนะในทางโท จะทำการเปลี่ยนสัญญาณไฟจราจรเพื่อให้ยานพาหนะในทางโทรสามารถผ่านได้ เมื่อเปลี่ยนสัญญาณไฟจราจรในทางโทแล้วจะเริ่มนับเวลาของสัญญาณไฟเขียวในทางโทเพื่อไม่ให้สัญญาณไฟเขียวในทางโทเกิน 30 วินาที หากระยะเวลาของสัญญาณไฟเขียวในทางโทยังไม่เกิน 30 วินาทีจะทำการตรวจสอบว่ายังมียานพาหนะรออยู่ที่ทางโทหรือไม่ ถ้ายังมี

ยานพาหนะมารออยู่จะยังคงให้สัญญาณไฟเขียวต่อไปจนกว่าจะเกินเวลาที่กำหนด แต่ถ้าไม่มียานพาหนะหรือระยะเวลาของสัญญาณไฟเขียวเกิน 30 วินาที จะทำการหยุดการนับเวลาและตรวจสอบว่าตัวตรวจจับยานพาหนะทำงานผิดพลาดหรือไม่ และจะเปลี่ยนสัญญาณไฟจราจรจากสัญญาณไฟเขียวเป็นสัญญาณไฟแดงในทางโท และให้สัญญาณไฟเขียวในทางเอก โดยการทำงานของฟังก์ชันจะสามารถอธิบายได้ดังรูปที่.3.24

มหาวิทยาลัยเทคโนโลยีสุรนารี

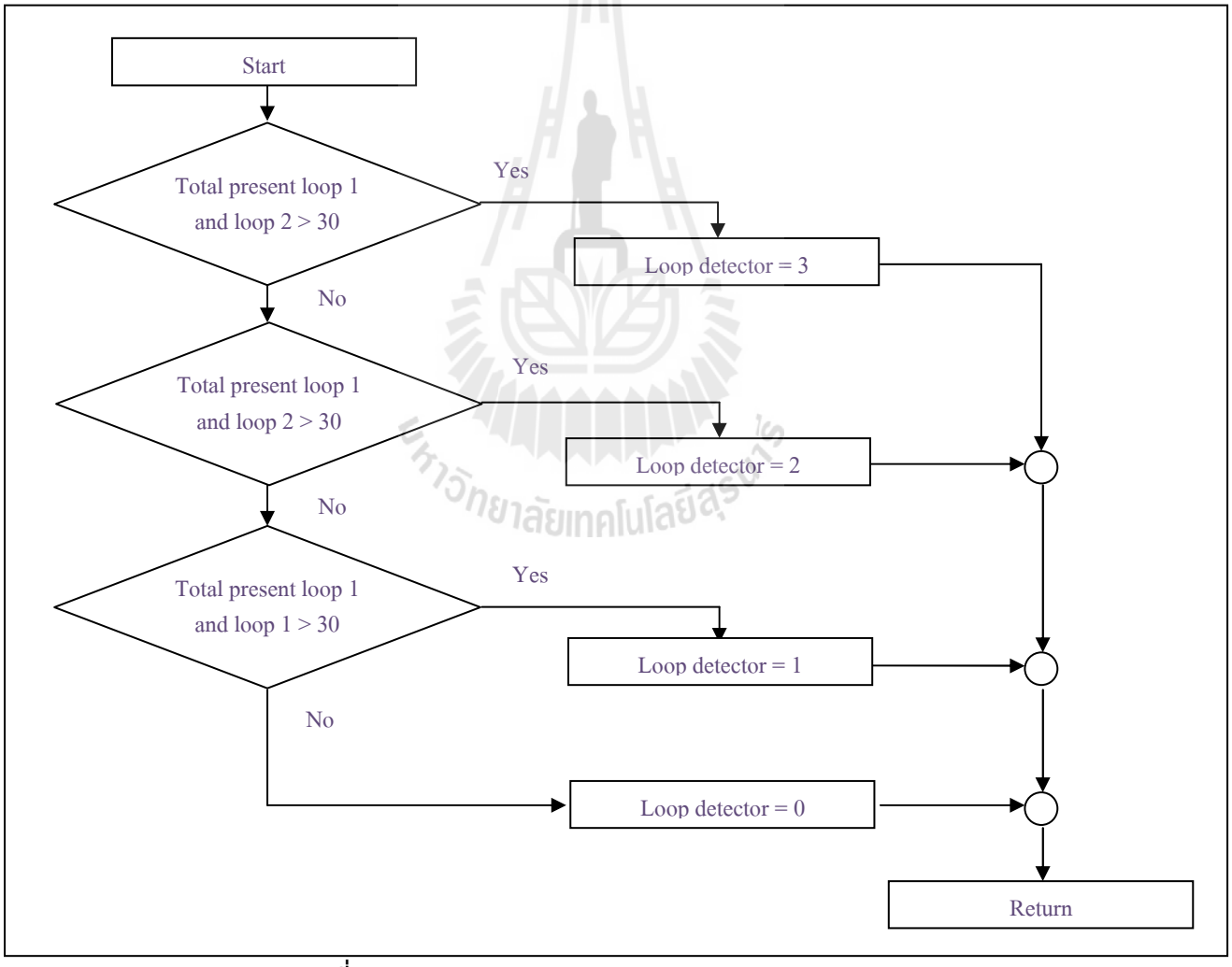


รูปที่.3.24 การทำงานของ Present Loop Function

3.2.5 Check Detector Fail Function

ในฟังก์ชันนี้จะทำการตรวจสอบตัวตรวจจับยานพาหนะในแต่ละ Channel ว่าเกิดความผิดพลาดหรือไม่ ซึ่งวิธีการตรวจสอบคือ การนับเวลาของตัวตรวจจับยานพาหนะ (Detector) ที่ส่งสัญญาณออกมา ถ้าระยะเวลาที่ส่งสัญญาณของตัวตรวจจับยานพาหนะ เกิน 30 วินาที แสดงว่าตัวตรวจจับยานพาหนะส่งสัญญาณออกมาตลอดเวลาซึ่งเป็นความผิดพลาดที่เกิดขึ้น ดังนั้นระบบจะไม่ทำการรับสัญญาณที่ส่งเข้ามาจาก Channel นั้น ซึ่งใน Check Detector Function จะทำการตรวจสอบในแต่ละ Channel ว่าเกิดความผิดพลาดหรือไม่ ซึ่งค่าที่ส่งกลับมาจากฟังก์ชันมีดังนี้

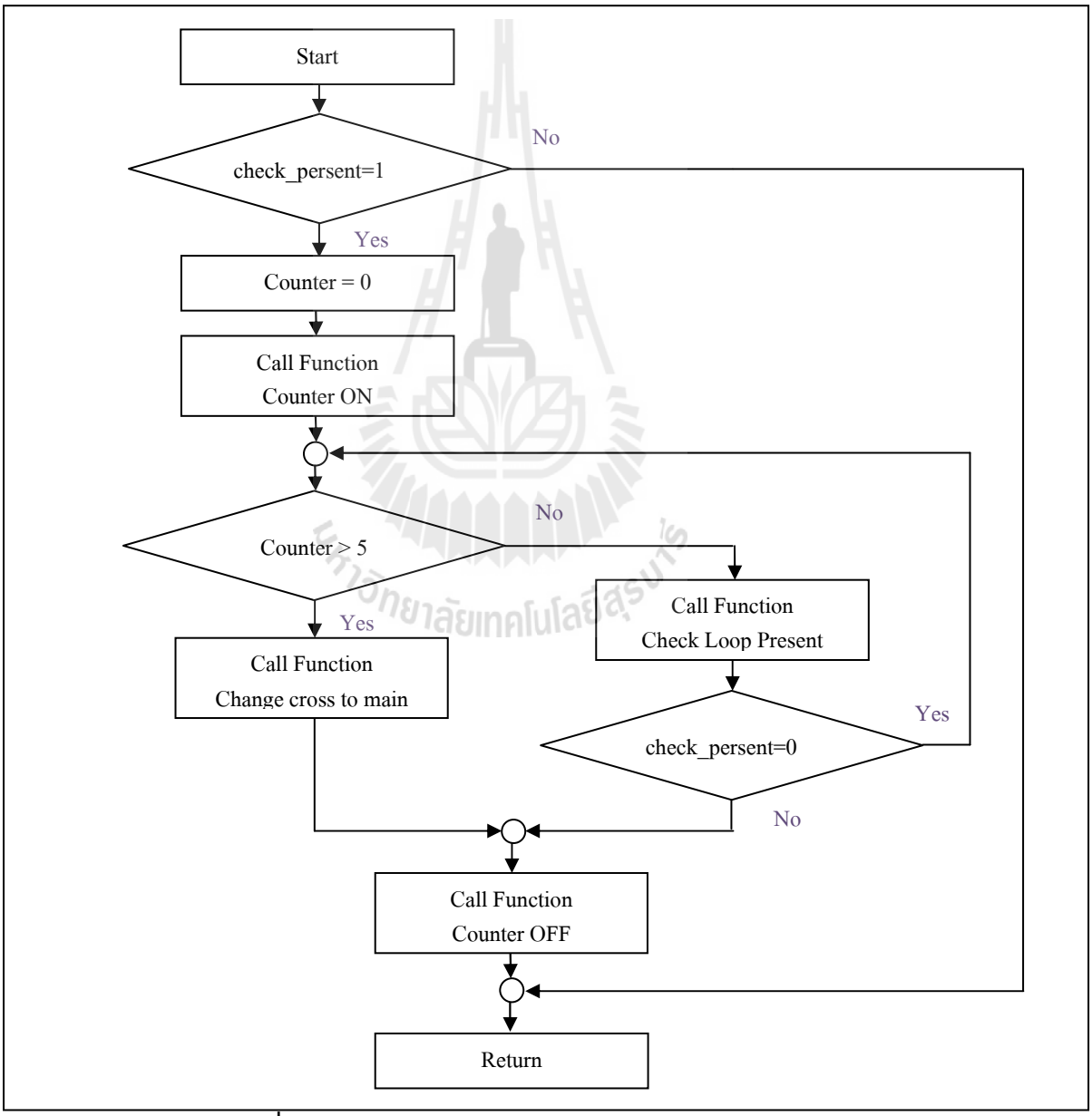
- ไม่เกิดความผิดพลาดจะส่งค่า loop_detector = 0
- Channel 1 เกิดความผิดพลาดจะส่งค่า loop_detector = 1
- Channel 2 เกิดความผิดพลาดจะส่งค่า loop_detector = 2
- Channel 1 และ Channel 2 เกิดความผิดพลาดจะส่งค่า loop_detector = 3



รูปที่.3.25 การทำงานของ Check Detector Function

3.2.6 Absent Loop Function

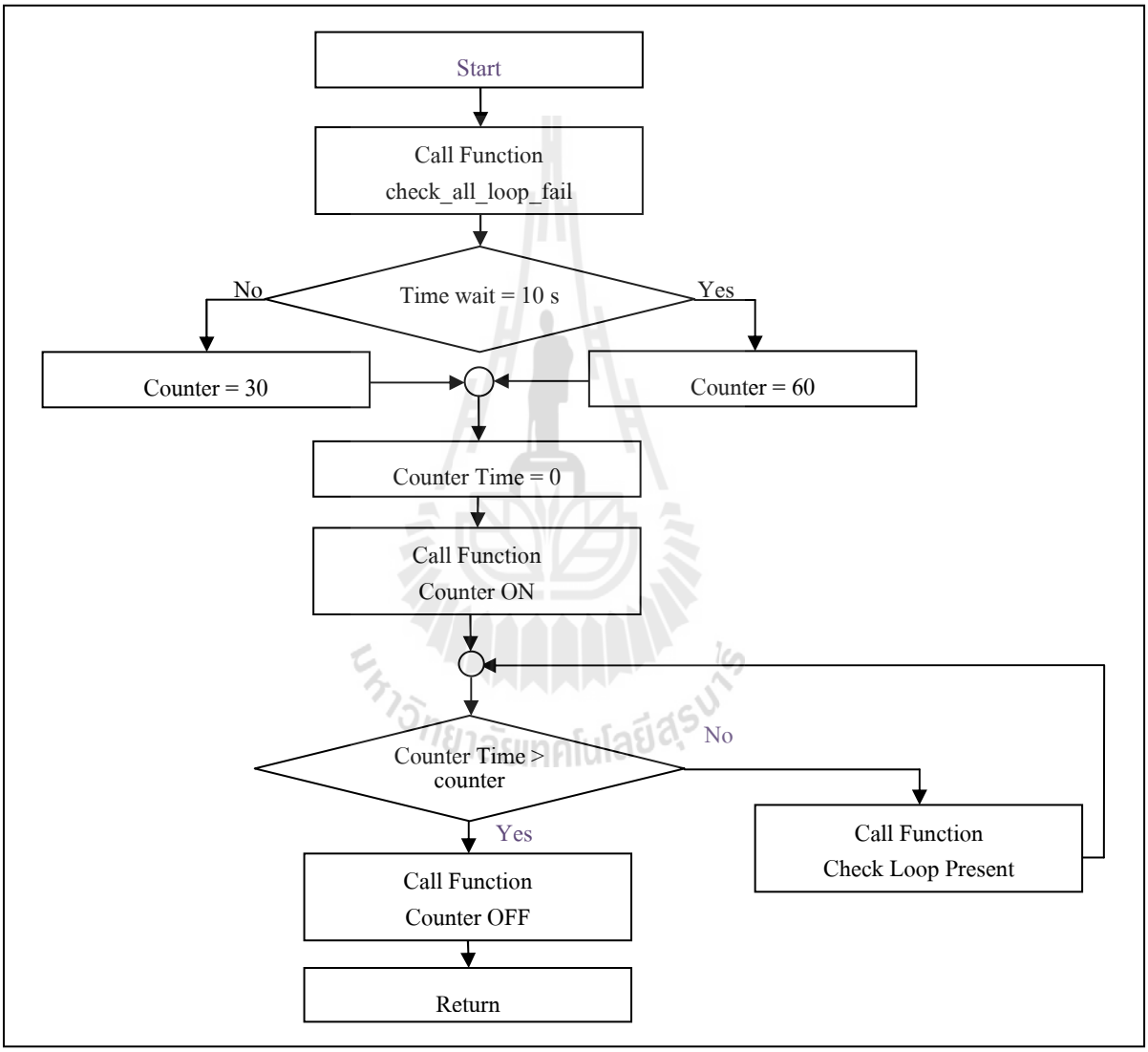
การทำงานของฟังก์ชันเมื่อตรวจไม่พบยานพาหนะในทางโท โดยจะตรวจสอบจาก Check Loop Present Function ในฟังก์ชันนี้จะเริ่มทำงานหลังจากที่มียานพาหนะเข้ามารอที่ทางโทแล้วได้สัญญาณไฟเขียวในช่วงเวลาที่ทางเอกได้สัญญาณไฟเขียวฟังก์ชันนี้จะไม่ทำงาน โดยหลังจากที่ให้สัญญาณไฟเขียวกับทางโทและยานพาหนะคันแรกผ่านไป ระบบจะทำการตรวจสอบว่าสามารถตรวจพบยานพาหนะคันต่อไปภายในเวลา 5 วินาทีหรือไม่ ถ้าตรวจพบภาพใน 5 วินาทีจะยังคงให้สัญญาณไฟเขียวที่ทางโทต่อไปจนกว่าจะเกินเวลาที่กำหนด แต่ถ้าหากตรวจไม่พบยานพาหนะภายในระยะเวลา 5 วินาที ระบบจะเปลี่ยนสัญญาณไฟจราจรในทางโทเป็นสัญญาณไฟแดง และให้สัญญาณไฟเขียวในทางเอก Absent Loop Function สามารถอธิบายการทำงานได้ดังรูปที่.3.26



รูปที่.3.26 การทำงานของ Absent Loop Function

3.2.7 Change Cross to Main Function

Change Cross to Main Function คือการเปลี่ยนสถานะจากสัญญาณไฟเขียวเป็นสัญญาณไฟเหลืองและสัญญาณไฟแดง ตามลำดับ โดยที่การเปลี่ยนสถานะสัญญาณไฟเขียวของทางโทไปทางเอกนั้น ระบบจะทำการเปิดสัญญาณไฟเขียวในทางเอกจนครบตามระยะเวลาที่กำหนดในโหมดการทำงาน จึงจะทำการตรวจสอบยานพาหนะในทางโทอีกครั้ง แต่ถ้าหากตรวจสอบไม่พบยานพาหนะจะยังคงเป็นสัญญาณไฟเขียวที่ทางเอกต่อไป แต่ถ้าหากตรวจพบยานพาหนะจะดำเนินการตามเงื่อนไปของการทำงานตามที่ได้กล่าวมาแล้ว ซึ่งสามารถอธิบายการทำงานของ Change Cross to Main Function ดังรูปที่.3.27



รูปที่.3.27 การทำงานของ Change Cross to Main Function

3.2.8 Change Main to Cross Function

Change Main to Cross Function คือฟังก์ชันการเปลี่ยนสถานะจากสัญญาณไฟเขียว เป็นสัญญาณไฟเหลือง และสัญญาณไฟแดง ตามลำดับ โดยที่จะมีสัญญาณไฟเหลือง 3 วินาที และไฟแดงพร้อมกันทั้งทางเอกและทางโท 2 วินาที

3.2.9 Counter On Function

Counter On Function จะทำงานแบบอินเตอร์รัป (Interrupt) ทุก ๆ 1 วินาที เพื่อเพิ่มค่าตัวแปรที่ฟังก์ชันต่าง ๆ ต้องการนับเวลาในทุก ๆ 1 วินาที เพื่อให้ได้ค่าเวลาที่ใช้ในการประมวลผลในฟังก์ชันนั้น ๆ

3.2.10 Counter Off Function

Counter Off Function จะทำการหยุดนับเวลาทั้งหมดของระบบควบคุมสัญญาณไฟจราจร

3.2.11 Tuning Loop Function

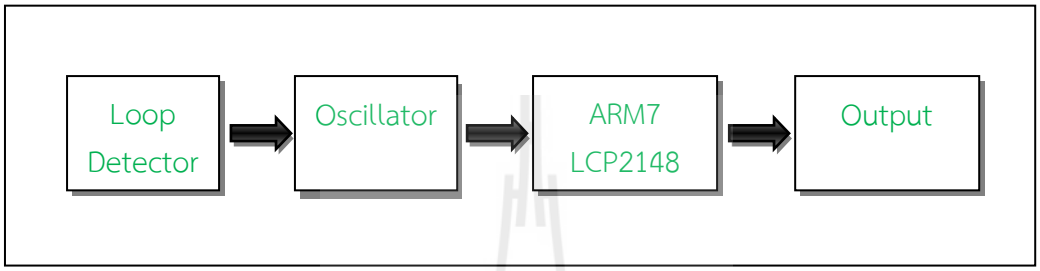
ในกรณีที่เกิดความผิดพลาดของตัวตรวจจับยานพาหนะใน Channel ต่าง ๆ จะทำการตรวจสอบ Channel ของตัวตรวจจับยานพาหนะที่ทำงานผิดพลาดอีกครั้ง เมื่อพบว่าสามารถทำงานได้ตามปกติจะนำตัวตรวจจับยานพาหนะ Channel นั้นมาพิจารณาในการตรวจจับยานพาหนะใหม่ แต่เมื่อพบว่า Channel นั้นยังคงทำงานผิดพลาดอยู่จะไม่นำมาพิจารณาในการตรวจสอบยานพาหนะ และจะยังคงตรวจสอบ Channel นั้นไปเรื่อย ๆ จนกว่าจะทำงานไม่ผิดพลาด

3.2.12 All Loop Fail Function

ในกรณีที่เกิดความผิดพลาดของตัวตรวจจับยานพาหนะทำงานผิดพลาดทั้ง 2 Channel ระบบจะเปลี่ยนเป็นสัญญาณไฟเหลืองกระพริบเตือน โดยจะเป็นสัญญาณไฟเหลืองอย่างต่อเนื่อง จนกว่าตัวตรวจจับยานพาหนะใน Channel ใดสามารถใช้งานได้ จึงจะกลับมาสู่โหมดการทำงานปกติโดยจะมีการตรวจสอบความผิดพลาดของตัวตรวจจับยานพาหนะอยู่ตลอดเวลา

ตัวตรวจจับยานพาหนะ (Detector)

ในระบบควบคุมสัญญาณไฟจราจรแบบกึ่งอัตโนมัติ (Semi - Actuated) ส่วนที่สำคัญอย่างยิ่งคือตัวตรวจจับยานพาหนะซึ่งเป็นส่วนที่คอยตรวจสอบยานพาหนะที่มารอในทางโท ซึ่งถ้าหากตัวตรวจจับยานพาหนะทำงานผิดพลาด ยานพาหนะที่รออยู่ในทางโทจะไม่ได้สัญญาณไฟเขียวเลย ดังนั้นตัวตรวจจับยานพาหนะจึงเป็นหัวใจหลักในการควบคุมระบบสัญญาณไฟจราจรแบบกึ่งอัตโนมัติ ในงานวิจัยนี้ได้พัฒนาตัวตรวจจับยานพาหนะสำหรับใช้ในระบบควบคุมสัญญาณไฟจราจร ซึ่งใช้หลักการของขวดลาดเหนี่ยวนำเพื่อให้เกิดความถี่เปลี่ยนแปลงไปเมื่อมียานพาหนะ โดยส่วนการทำงานของตัวตรวจจับยานพาหนะสามารถอธิบายได้โดยรูปที่.3.28



รูปที่.3.28 การทำงานของตัวตรวจจับยานพาหนะ (Detector)

จากรูปที่.3.28 จะมีส่วนการทำงานแบ่งออกเป็น 4 ส่วนคือ 1. ลูปเหนี่ยวนำ (Loop Detector) 2. ตัวกำเนิดความถี่ (Oscillator) 3. หน่วยประมวลผล (ARM7 LPC2148) 4. ผลการตรวจสอบ (Output) จากรูปลูปเหนี่ยวนำถูกเชื่อมต่อกับตัวกำเนิดความถี่ ซึ่งตัวกำเนิดสัญญาณจะสร้างความถี่ที่คงที่ออกมาค่าหนึ่ง แต่เมื่อลูปเหนี่ยวนำถูกเหนี่ยวนำด้วยยานพาหนะจะทำให้ตัวกำเนิดความถี่สร้างความถี่ที่มีค่าเปลี่ยนไป และความถี่ที่ได้จากตัวกำเนิดความถี่จะถูกอ่านค่าความถี่ออกมาโดยใช้ไมโครคอนโทลเลอร์ARM7 LPC2148 ทำหน้าที่เป็นเครื่องนับความถี่ (Frequency Counter) และหาอัตราการเปลี่ยนแปลงของความถี่เพื่อนำมาตัดสินใจว่าความถี่ที่เปลี่ยนแปลงไปนั้นเกิดจากการเหนี่ยวนำความถี่ที่เกิดขึ้นจากยานพาหนะหรือไม่และจะส่งผลที่ได้ออกมา การออกแบบการทำงานของตัวตรวจจับยานพาหนะสามารถแบ่งออกเป็น 2 ส่วนคือ 1. ส่วนของการออกแบบฮาร์ดแวร์ (Hardware) 2. ส่วนของการออกแบบซอร์ฟแวร์ (Software) ซึ่งจะมีการทำงานของส่วนต่าง ๆ ดังนี้

รูปที่.3.29 ตัวตรวจจับยานพาหนะ (Detector)

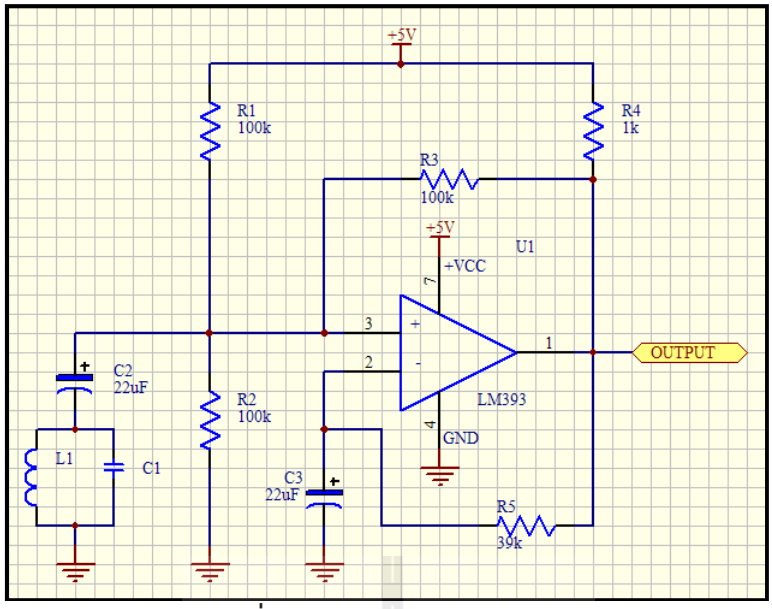
รูปที่.3.29 คือ ตัวตรวจจับยานพาหนะที่ออกแบบในงานวิจัยนี้ โดยตัวตรวจจับยานพานหะที่ออกแบบมีขนาด 9 x 17 เซนติเมตร สามารถตัวจับยานพาหนะได้ 2 Channel สามารถปรับความถี่ได้ 4 ความถี่เพื่อไม่ให้เกิดการรบกวนกันของตัวตรวจจับยานพาหนะตัวอื่น ๆ และนอกจากนี้ยังสามารถเลือกความไว (Sensitivity) เพื่อให้สามารถตรวจจับยานพาหนะขนาดเล็กได้

3.3 ส่วนของการออกแบบฮาร์ดแวร์ตัวตรวจจับยานพาหนะ (Detector)

ในส่วนของการออกแบบฮาร์ดแวร์ (Hardware) ของของตัวตรวจจับยานพาหนะนั้นสามารถแบ่งออกเป็นส่วนย่อย ๆ คือ ลูปเหนี่ยวนำ, ตัวกำเนิดสัญญาณและไมโครคอนโทลเลอร์ ARM7 LPC2148 ซึ่งจะมีการทำงานของส่วนต่าง ๆ ดังนี้

3.3.1 ตัวกำเนิดสัญญาณ (Oscillator)

ตัวกำเนิดสัญญาณที่นำมาใช้ในงานวิจัยนี้จะเป็นวงจรกำเนิดสัญญาณแบบ LC Oscillator สามารถกำเนิดความถี่อยู่ระหว่าง 20 - 100 kHz ซึ่งค่าความเหนี่ยวนำที่ทำให้เกิดความถี่จะได้มาจากลูปเหนี่ยวนำที่อยู่บนท้องถนน เมื่อยานพาหนะมาหยุดอยู่บนลูปเหนี่ยวนำที่ติดตั้งเอาไว้ จะทำให้เกิดค่าความเหนี่ยวนำเปลี่ยนไป ซึ่งจะส่งผลให้ความถี่เปลี่ยนแปลงด้วย วงจรกำเนิดความถี่ที่ออกแบบจะเป็นดังรูปที่.3.30



รูปที่.3.30 วงจรกำเนิดสัญญาณ

จากรูปที่.3.30 เป็นวงจำกำเนิดสัญญาณแบบ LC Oscillator ซึ่งจะออกแบบโดยการ IC Op-Amp เบอร์ LM393 ในการกำเนิดความถี่ ซึ่งสามารถคำนวณความถี่ที่ได้จากสมการ (1)

$$f = \frac{1}{2\pi\sqrt{L_1 C_1}} \tag{1}$$

จากสมการ (1) ความถี่ที่เกิดขึ้นจะขึ้นอยู่กับค่าความเหนี่ยวนำ L1 และตัวเก็บประจุ C1 ซึ่งในงานวิจัยนี้ค่าความเหนี่ยวนำ L1 จะได้มาจากลูปเหนี่ยวนำที่ติดตั้งอยู่บนถนนซึ่งจะมีค่าความเหนี่ยวนำประมาณ 100 µH ดังนั้นเพื่อให้ได้ความถี่ 50 kHz สามารถเราคำนวณหาค่าตัวเก็บประจุ C1 ได้จากสมการ (1) ได้ค่าตัวเก็บประจุ C1 = 0.10 µF นอกจากนี้เราสามารถที่จะเปลี่ยนความถี่ของตัวกำเนิดสัญญาณได้โดยการเปลี่ยนค่าตัวเก็บประจุ C1 เนื่องจากบริเวณที่ตรวจวัดยานพาหนะอาจมีลูปเหนี่ยวนำหลายตัวทำให้เกิดการรบกวนเกิดขึ้นได้ ดังนั้นจึงได้ออกแบบสวิตช์ปรับความถี่เพื่อให้ได้ความถี่ในช่วง 20 – 100 kHz โดยการนำตัวเก็บประจุมาขนานกับวงจรกำเนิดสัญญาณ ซึ่งตัวเก็บประจุที่อยู่กับวงจรมีขนาด 0.1 µF ทำให้วงจรกำเนิดความถี่ได้เพียงความถี่เดียว แต่เมื่อเพิ่มตัวเก็บประจุขนาด 0.1 µF และ 0.2 µF เข้าไปในวงจรกำเนิดความถี่โดยการต่อแบบขนานกับตัวเก็บประจุในวงจร ได้ผลดังตารางต่อไปนี้

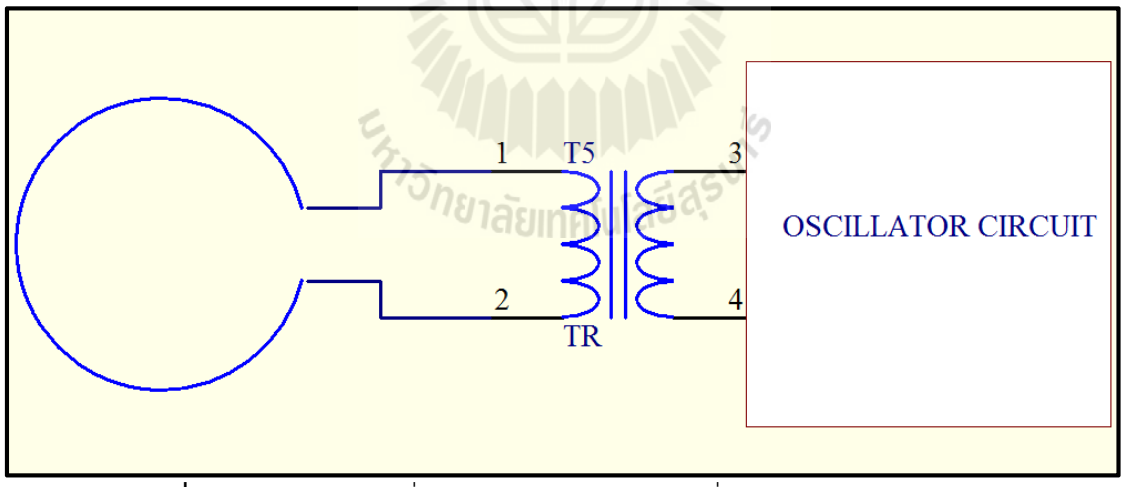
ตารางที่ 3.4 ผลการทดสอบเมื่อทำการปรับสวิตช์

กรณีที่	สวิตช์ 1	สวิตช์ 2	ค่าตัวเก็บประจุรวม (µF)	ความถี่ที่ได้จากการคำนวณ (kHz)
1	เปิด	เปิด	0.1	46.13
2	ปิด	เปิด	0.2	32.62
3	เปิด	ปิด	0.3	26.63
4	ปิด	ปิด	0.4	23.06

ซึ่งการเลือกความถี่ดังกล่าวสามารถเลือกโดยการเลือกผ่าน Dip Switch ซึ่งจะติดตั้งอยู่ในตัวตรวจจับยานพาหนะ ดังรูปที่.3.31

รูปที่.3.31 วงจรกำเนิดสัญญาณ (Oscillator)

ลูปเหนี่ยวนำที่ถูกฝังอยู่บนถนนทำหน้าที่เป็นอินพุตของวงจร มีลักษณะเป็นโลหะชนิดหนึ่งเมื่อถูกฝังไว้ใต้พื้นถนนกลายเป็นสายล่อฟ้าได้และถ้าเกิดฟ้าผ่าที่ลูปเหนี่ยวนำจะส่งผลทำให้วงจรกำเนิดความถี่พังเสียหายได้ เพราะลูปเหนี่ยวนำถูกต่อเข้ากับบอร์ดวงจรกำเนิดความถี่โดยตรง จึงได้ออกแบบให้ลูปเหนี่ยวนำและวงจรกำเนิดความถี่เชื่อมต่อกันโดยใช้หม้อแปลงไฟฟ้าเป็นตัวกลาง



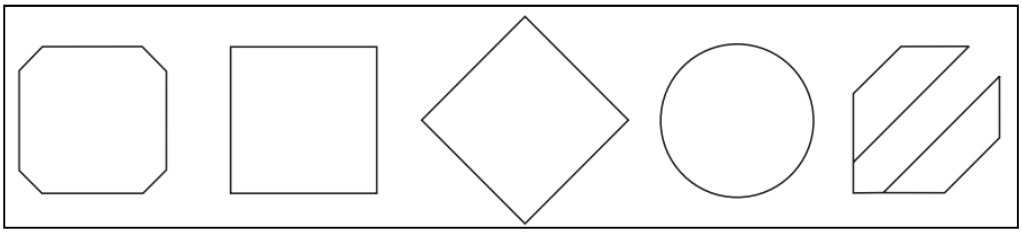
รูปที่.3.32 การต่อลูปเหนี่ยวนำกับวงจรกำเนิดความถี่โดยผ่านหม้อแปลงไฟฟ้า

หม้อแปลงไฟฟ้าที่ด้านปฐมภูมิจะต่อกับลูปเหนี่ยวนำและด้านทุติยภูมิจะต่อกับวงจรกำเนิดความถี่ เมื่อมียานพาหนะมาหยุดบนลูปเหนี่ยวนำจะทำให้ขดลวดของหม้อแปลงด้านปฐมภูมิเหนี่ยวนำขดลวดของหม้อแปลงด้านทุติยภูมิ ทำให้ความเหนี่ยวนำของวงจรเปลี่ยนส่งผลให้ความถี่ของวงจรกำเนิดความถี่เปลี่ยนไป

หม้อแปลงไฟฟ้าที่ใช้เกิดจากการพันเส้นลวดรอบแกนเฟอร์ไรท์ โดยจะพันรอบด้านปฐมภูมิ 100 รอบ และด้านทุติยภูมิ 100 รอบ นั่นคือ หม้อแปลงจะมีอัตราส่วนเป็น 1:1 เพื่อที่จะให้การเปลี่ยนแปลงเกิดขึ้น

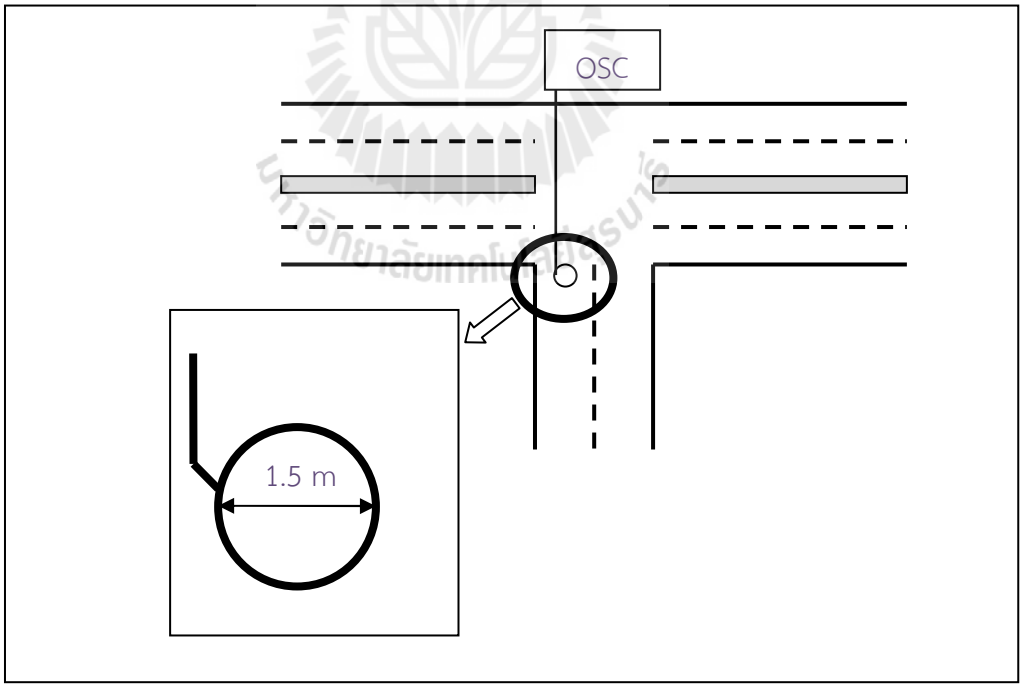
ด้านทุติยภูมิ (ด้านวงจรกำเนิดความถี่) มีการเปลี่ยนแปลงเช่นเดียวกันกับด้านปฐมภูมิ (ด้านลูปเหนี่ยวนำ) เสมือนว่าลูปเหนี่ยวนำได้ต่อกับวงจรกำเนิดความถี่โดยตรง

3.3.2 ลูปเหนี่ยวนำ (Loop Detector)



รูปที่.3.33 รูปแบบของขดลวดแบบต่าง ๆ

ลูปเหนี่ยวนำจะมีลักษณะเป็นขดลวดเหนี่ยวนำซึ่งจะเชื่อมต่ออยู่กับตัวกำเนิดสัญญาณ จะมีรูปแบบต่างๆ ดังรูปที่.3.33 โดยในงานวิจัยนี้ได้เลือกใช้รูปแบบวงกลมเพราะมีอัตราการเปลี่ยนแปลงความถี่สูงเมื่อมียานพาหนะเข้ามาเหนี่ยวนำและได้ออกแบบให้ลูปเหนี่ยวนำแบบวงกลมที่มีเส้นผ่านศูนย์กลาง 1.5 เมตร ติดตั้งลูปเหนี่ยวนำโดยการฝังอยู่บนท้องถนนและนำมาเชื่อมต่อกับตัวกำเนิดสัญญาณ ระยะทางของลูปเหนี่ยวนำกับตัวกำเนิดสัญญาณประมาณ 30 เมตร การติดตั้งลูปเหนี่ยวนำจะติดตั้งตามรูปที่.3.34



รูปที่.3.34 การติดตั้งลูปเหนี่ยวนำ

รูปที่.3.35 การติดตั้งลูปเหนี่ยวนำที่พื้นถนน

การหาค่าความเหนี่ยวนำของลูปเหนี่ยวนำ

จากสมการที่ (1) เพื่อให้ได้ความถี่ที่ถูกต้องจำเป็นต้องรู้ค่าความเหนี่ยวนำของลูปเหนี่ยวนำ และนำมาคำนวณหาค่าความถี่ได้ ดังนั้นจากสมการที่ (1) เมื่อให้ค่าตัวเก็บประจุมีค่าคงที่และนำลูปเหนี่ยวนำมาขนานกับตัวเหนี่ยวนำในวงจร โดยจะเลือกใช้ตัวเก็บประจุขนาด 1 µF (C1) และตัวเหนี่ยวนำขนาด 100 µH (L1) ทำการวัดค่าความถี่จะได้มีค่าเท่ากับ 21.6 kHz (F) และเมื่อทำการแก้สมการ (1) แล้ว จะสามารถทราบค่าความเหนี่ยวนำทั้งหมดได้ คือ

$$L_{รวม} = \left(\frac{1}{2\pi F\sqrt{C1}}\right)^2 \quad (2)$$

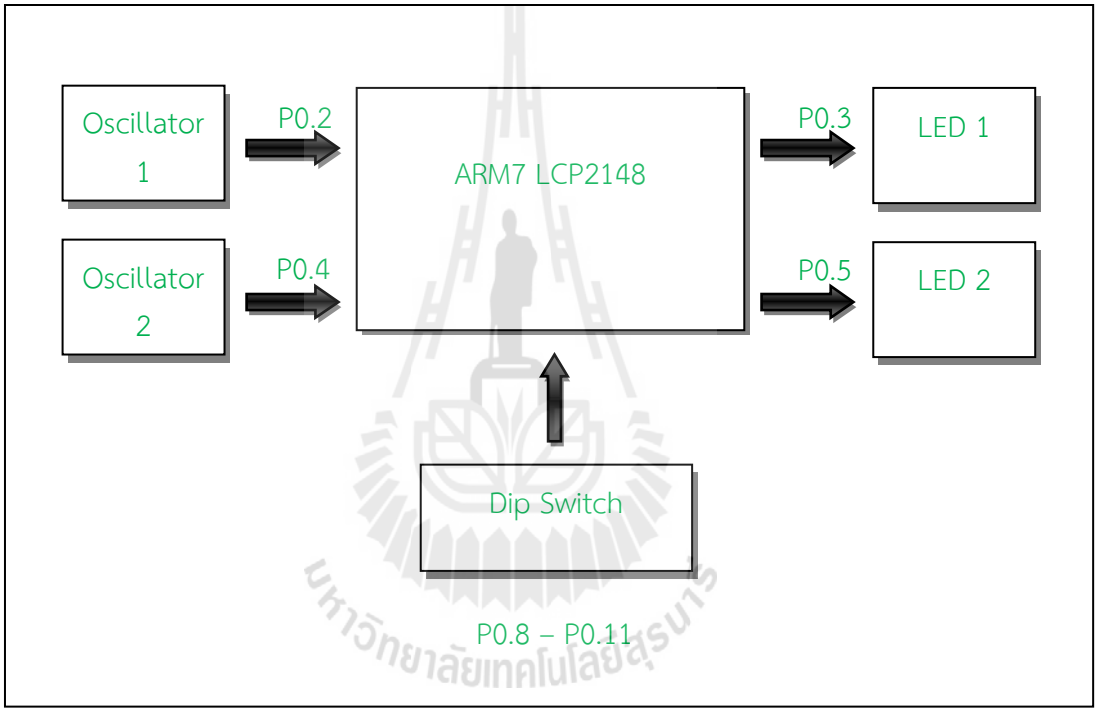
จากการแก้สมการได้ $L_{รวม}$ = 54.34 µH ทำให้ทราบความเหนี่ยวนำของลูปเหนี่ยวนำได้จาก

$$L_{รวม} = \text{Loop} \;//\; 100\ \mu H$$

ดังนั้นค่าความเหนี่ยวนำของลูปเหนี่ยวนำจะมีค่าเท่ากับ 119.03 µH

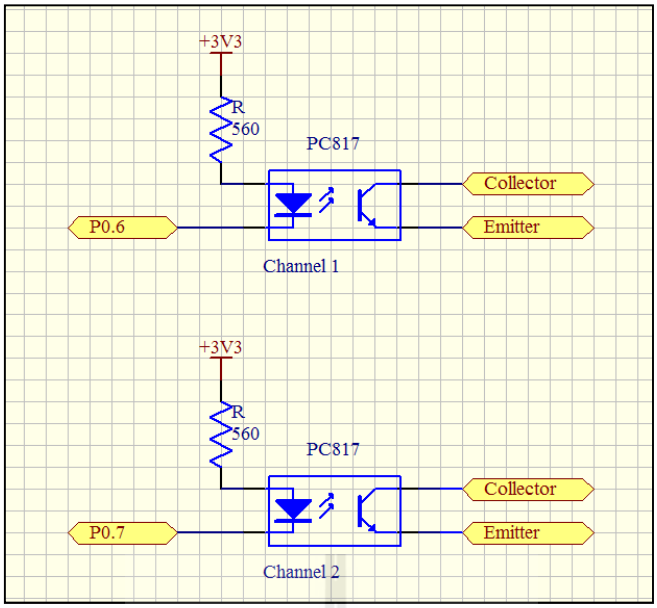
3.3.2 ไมโครคอนโทลเลอร์ ARM7 LPC2148

สำหรับการประมวลผลของตัวตรวจจับยานพาหนะจะใช้ไมโครคอนโทลเลอร์ ARM7 LPC2148 ซึ่งจะทำหน้าที่เป็นเครื่องนับความถี่และประมวลผลของสัญญาณที่ได้ ในการนับความถี่จากวงจรกำเนิดสัญญาณไมโครคอนโทลเลอร์จะใช้ฟังก์ชันการนับ (Counter) เพื่อตรวจสอบสัญญาณที่เข้ามาในช่วงเวลา 0.1 วินาที และแปลงสัญญาณที่ได้ให้อยู่ในรูปของความถี่ ไมโครคอนโทลเลอร์ ARM7 LPC2148 จะมีฟังก์ชันการนับที่ชื่อว่า Capture ดังนั้นจะออกแบบให้สามารถนับความถี่ได้ 2 Channel เพื่อให้ตรงกับการออกแบบตัวตรวจจับยานพาหนะที่ต้องการให้สามารถตรวจจับยานพาหนะได้ 2 Channel จึงเปิดใช้งานฟังก์ชัน Capture0.0 (Port P0.2) และ Capture0.1 (Port P0.4) ซึ่งการเชื่อมต่อของวงจรกำเนิดสัญญาณกับไมโครคอนโทลเลอร์ ARM7 LPC2148 จะเชื่อมต่อดังรูปที่.3.36



รูปที่.3.36 การเชื่อมต่อวงจรกำเนิดความถี่กับไมโครคอนโทลเลอร์ ARM7 LPC218

นอกจากวงจรกำเนิดสัญญาณที่เชื่อมต่อกับไมโครคอนโทลเลอร์แล้ว ยังมี LED เพื่อแสดงสถานะของตัวตรวจจับยานพาหนะ โดยที่ LED1 แสดงสถานการณ์ตรวจจับยานพาหนะของ Channel 1 และLED2 แสดงสถานการณ์ตรวจจับยานพาหนะของ Channel 2 และยังมี Dip Switch 4 ช่อง สำหรับเลือกความไว (Sensitivity) ของแต่ละ Channel โดยที่ Dip Switch 1 และ 2 สำหรับตัวตรวจจับยานพาหนะ Channel 1 , Dip Switch 3 และ 4 สำหรับตัวตรวจจับยานพาหนะ Channel 2



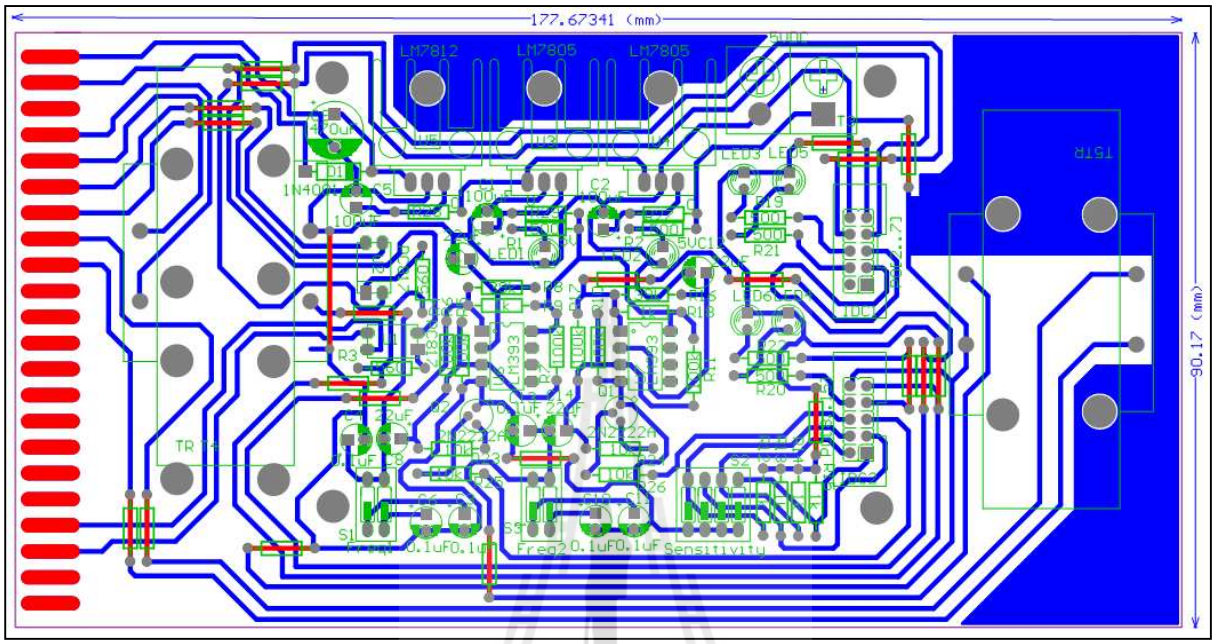
รูปที่.3.37 การเชื่อมต่อไมโครคอนโทลเลอร์ ARM7 LPC218 กับวงจรภายนอก

จากรูปวงจรที่ 3.37 เป็นการเชื่อมต่อไมโครคอนโทลเลอร์ ARM7 LPC218 กับวงจรภายนอก เพื่อนำสัญญาณที่ได้จากตัวตรวจจับยานพาหนะไปประมวลผล โดยเมื่อตัวตรวจจับยานพาหนะตรวจสอบพบยานพาหนะจะทำการส่งสัญญาณออกมาโดยการให้พอร์ตของไมโครคอนโทรเลอร์มี Logic 0 เพื่อให้ Optoisolators เบอร์ PC817 ทำงาน จากรูปที่.3.37 จะเห็นว่าพอร์ต P0.6 จะทำหน้าที่ควบคุมการส่งสัญญาณของตัวตรวจจับยานพาหนะ Channel 1 และพอร์ต P0.7 จะทำหน้าที่ควบคุมการส่งสัญญาณของตัวตรวจจับยานพาหนะ Channel 2

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.3.3 ออกแบบวงจรรวมของตัวตรวจจับยานพาหนะ

จากส่วนต่างๆ ที่ได้กล่าวมาแล้วนั้นจะนำมาออกแบบวงจรรวมเพื่อให้ได้บอร์ดควบคุมสัญญาณไฟจราจร โดยจะมีการออกแบบวงจรต่าง ๆ ดังนี้

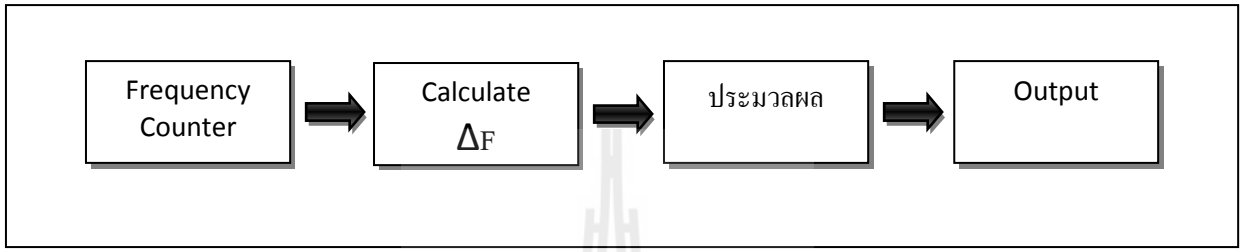


รูปที่.3.38 ลักษณะการวางตำแหน่งของอุปกรณ์ต่างๆ ของบอร์ดตรวจจับยานพาหนะ

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.4 ส่วนของการออกแบบซอร์ฟแวร์ตัวตรวจจับยานพาหนะ (Detector)

ส่วนของการออกแบบซอร์ฟแวร์ของตัวตรวจจับยานพาหนะ จะออกแบบให้ไมโครคอนโทรเลอร์ทำหน้าที่เป็นเครื่องนับความถี่จากตัวกำเนิดสัญญาณ และนำความถี่ที่ได้มาคำนวณอัตราการเปลี่ยนแปลงความถี่ (ΔF) เมื่อมียานพาหนะเข้ามาในลูปเหนี่ยวนำ อัตราการเปลี่ยนแปลงความถี่ที่เกิดขึ้นจะถูกตีความว่ามียานพาหนะจริงหรือไม่โดยส่วนการประมวลผลและจะส่งข้อมูลที่จากการตรวจสอบยานาพาหนะให้กับส่วนที่เชื่อมต่อกับวงจรภายนอก ซึ่งการทำงานของตัวตรวจจับยานพาหนะสามารถอธิบายได้จากรูปที่.3.39

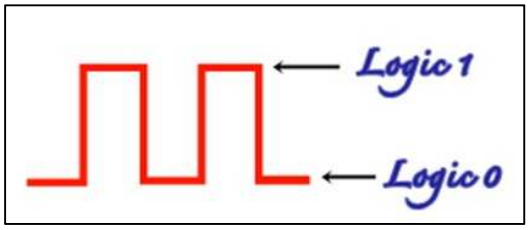


รูปที่.3.39 การทำงานของตัวตรวจจับยานพาหนะ

วิธีการนับความถี่

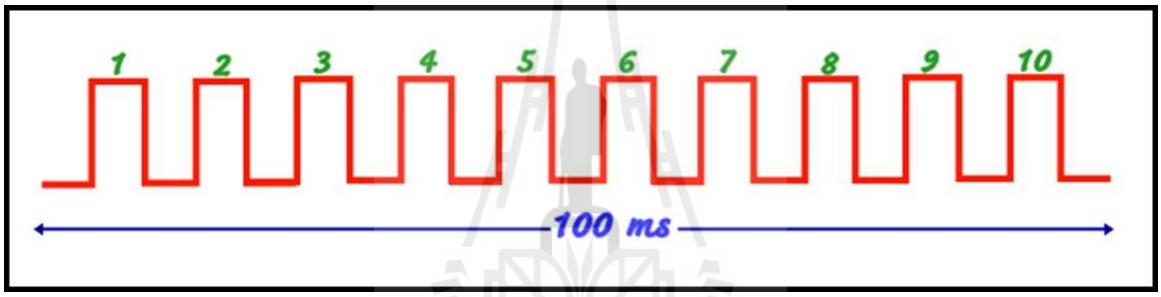
รูปที่.3.40 วิธีการนับสัญญาณในช่วงเวลา 1 วินาที

ในการนับความถี่สำหรับตรวจวัดยานพานะนั้น เราจะนับความถี่แบบนับในช่วงเวลา ซึ่งความถี่ที่เข้ามาจะมีลักษณะเป็นสัญญาณ Pulse โดยจะนับความถี่ในช่วงเวลา 1 วินาที (sec) เพื่อดูว่าสัญญาณที่เข้ามานั้นมีกี่ลูกคลื่น แล้วค่าที่ได้จะเป็นค่าของความถี่ ซึ่งในไมโครคอนโทรลเลอร์นั้น จะใช้ Interrupt Capture (รายละเอียดในบทที่ 2) ในการนับความถี่



รูปที่.3.41 สัญญาณPulse ที่เข้ามา Logic1 และ Logic0

จากที่ได้กล่าวมาแล้วว่าการนับความถี่จะนับความในช่วงระยะเวลา 1 วินาที ซึ่งจะทำให้ใช้เวลาในการนับแต่ละรอบใช้เวลานาน ดังนั้นการนับความถึงจึงจะนับในช่วงเวลาเพียง 1 ใน 10 หรือ 100 ms เท่านั้น โดยไมโครคอนโทรลเลอร์นั้นจะเริ่มทำงานนับความถี่ที่สัญญาณPulse มีค่าเท่ากับ Logic 1 ซึ่งจะนับในช่วงเวลาที่เราทำการ Delay ไว้ 100 ms ว่าในช่วงเวลานี้นั้น Logic1 เท่ากับเท่าไหร่ จากนั้นนำค่าที่ได้มาคำนวณหาค่าความถี่ที่แท้จริง โดย นำค่าที่อ่านได้ x 10 เพื่อให้ได้ความถี่ที่แท้จริง



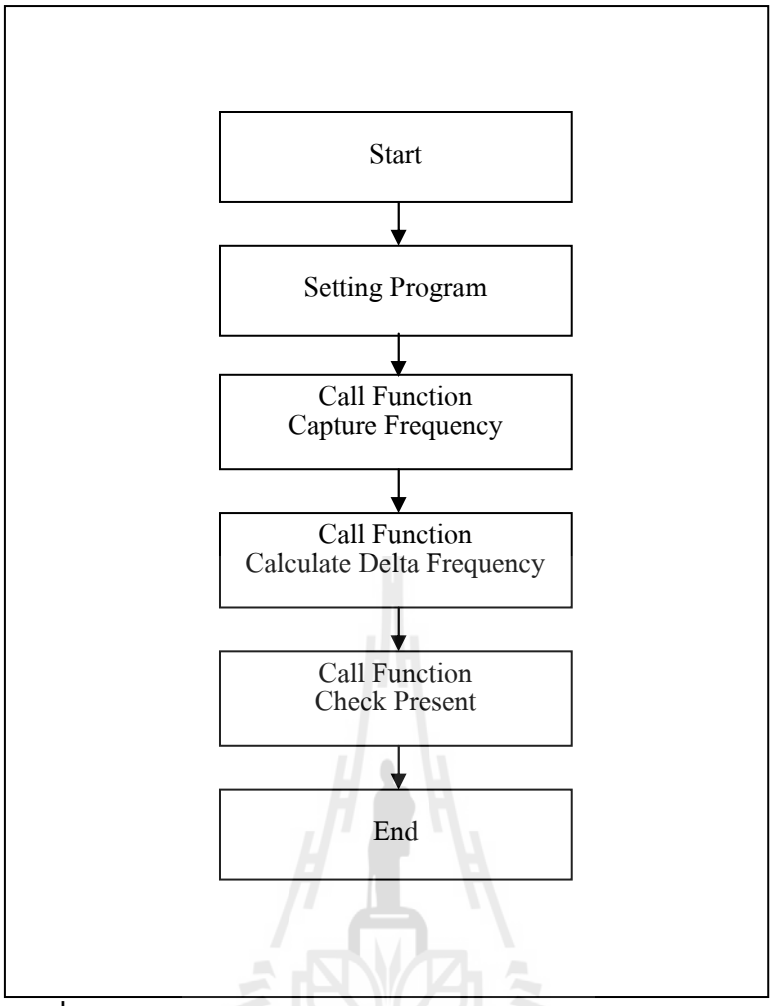
รูปที่.3.42 สัญญาณ Pulse ที่จับได้ในช่วงเวลา 100 ms

จากรูปจะเห็นว่าสัญญาณ Pulse ที่เข้ามานั้น ในช่วงเวลา 100 ms มีค่าที่เป็น Logic 1 จำนวน 10 ค่า ดังนั้นค่าความถี่ที่ได้จากช่วงเวลานี้คือ 10 x 10 = 100 Hz

การออกแบบซอร์ฟแวร์ของตัวตรวจจับยานพหานะนั้นจะแบ่งฟังก์ชันการทำงานออกเป็นย่อยๆ เพื่อให้สามารถสะดวกในการทำงานและการตรวจสอบแก้ไขซอร์ฟแวร์ ซึ่งจะมีฟังก์ชันการทำงานต่างๆ ดังนี้

3.4.1 Main Function

เป็นฟังก์ชันการทำงานหลักของตัวตรวจจับยานพหานะโดยจะมีการเรียกให้ฟังก์ชันย่อย ๆ เพื่อทำการประมวลผล การทำงานเริ่มต้นจะทำการตั้งค่าการทำงานต่าง ๆ ของไมโครคอนโทรเลอร์ เมื่อทำการตั้งค่าเรียบร้อยแล้วจะอ่านค่าความถี่ของ Channel 1 และChannel 1 หลังจากที่ได้อ่านค่าความถี่แล้วจะทำการคำนวณอัตราการเปลี่ยนแปลงความถี่เพื่อที่จะนำมาตรวจสอบว่าตรวจสอบพบยานพาหนะหรือไม่ โดยการตรวจสอบดังกล่าวจะนำผลการตรวจสอบยานพาหนะที่ได้ส่งสัญญาณออกไปให้กับส่วนการเชื่อมต่อภายนอก การทำงานของ Main Function จะสามารถอธิบายได้ดังรูปที่.3.43



รูปที่.3.43 การทำงานของ Main Function ตัวตรวจจับยานพาหนะ

3.4.2 Setting Program

ในโปรแกรมการทำงานของตัวตรวจจับยานพาหนะจะทำการตั้งค่าต่างๆ ของไมโครคอนโทรเลอร์ ARM7 LPC2148 เพื่อให้สามารถทำงานได้ตามวัตถุประสงค์ โดยมีการตั้งค่าต่าง ๆ ดังนี้

- เปิดใช้งานระบบเวลาจริง (Real Time Clock) ซึ่งจะใช้งานแบบอินเตอร์รัป (Interrupt) ที่เกิดการอินเตอร์รัปทุก ๆ 1 นาที เพื่อนำมาใช้งานในการนับเวลา
- เปิดใช้งานโหมด Timer0 ที่ทำงานเป็น Counter เพื่อนับสัญญาณที่เข้ามาจากตัวกำเนิดความถี่

3.4.3 Capture Frequency Function

ในการทำงานของ Capture Frequency จะทำการอ่านค่าความถี่ที่ได้จากตัวกำเนิดสัญญาณโดยการใช้เทคนิคการนับสัญญาณ pulse การนับสัญญาณนั้นจะใช้งานโหมด Counter ในการนับสัญญาณ ซึ่งจะมีระยะเวลาในการนับสัญญาณ 1 วินาที จำนวนสัญญาณ pulse ที่ได้นั้นคือความถี่ที่เกิดขึ้นจากตัวกำเนิดสัญญาณ

3.4.4 Calculate Delta Frequency Function

ในฟังก์ชันนี้จะทำการคำนวณอัตราการเปลี่ยนแปลงความถี่ หรือความไวในการตอบสนองของตัวตรวจจับยานพาหนะโดยจะทำการคำนวณจากสมการ (2)

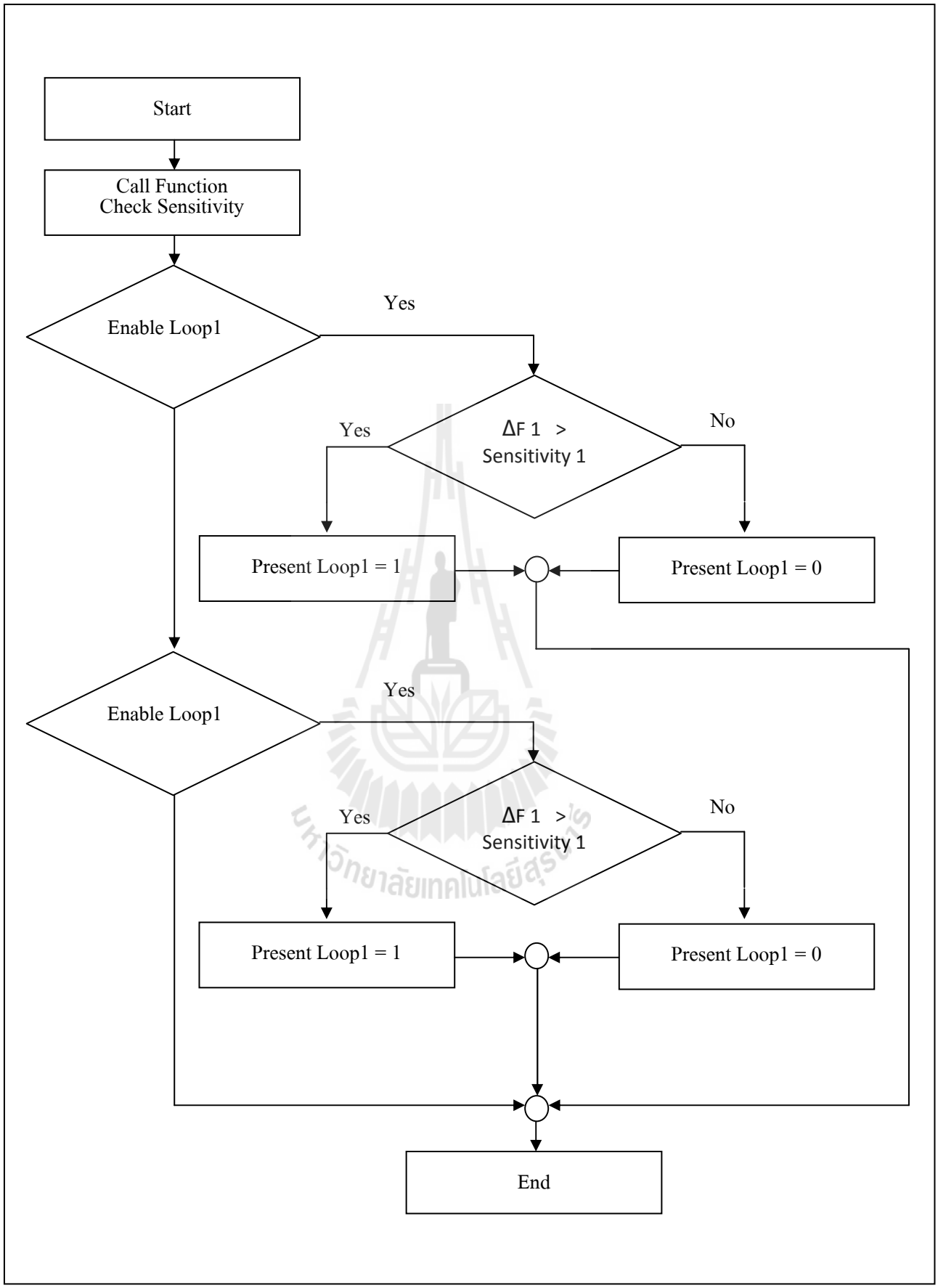
$$\Delta F = \frac{\left| f_B - f_R \right|}{f_B} \times 100\% \qquad (3)$$

จากสามาการที่ (2) คือสมการอัตราการเปลี่ยนแปลงความถี่ โดยที่ f_B คือ Frequency Base ความถี่ที่เป็นความถี่ในขณะที่ไม่มียานพาหนะอยู่ที่ลูปเหนี่ยวนำ, f_R คือ Frequency Real ความถี่ที่เกิดขึ้นจากตัวกำเนิดสัญญาณ ซึ่งเมื่อมียานพาหนะเข้ามาที่ลูปเหนี่ยวนำจะทำให้ความถี่ที่ได้เปลี่ยนแปลงไป ดังนั้นในฟังก์ชันการทำงานนี้จะทำการคำนวณอัตราการเปลี่ยนแปลงความถี่ของตัวตรวจจับยานพาหนะ

3.4.5 Check Present Function

ในฟังก์ชันการทำงานนี้จะทำการตรวจสอบค่าอัตราการเปลี่ยนแปลงความถี่ที่ได้ ว่าอัตราการเปลี่ยนแปลงที่ได้มากกว่าที่กำหนดหรือไม่ ถ้ามีมากกว่าที่กำหนดเอาไว้จะตีความได้ว่ามียานพาหนะอยู่บนลูปเหนี่ยวนำ ซึ่งในการกำหนดค่าอัตราการเปลี่ยนแปลงนั้นในงานวิจัยนี้ได้มีค่าที่กำหนดเอาไว้ 3 ค่าคือ 0.5%, 1% และ 2% ซึ่งสามารถเลือกค่าที่กำหนดได้จาก Dip Switch ที่อยู่บนตัวตรวจจับยานพาหนะ โดยการทำงานของฟังก์ชันคือต้องตรวจสอบ Dip Switch ก่อนเพื่อดูว่าในขณะนี้เลือกกำหนดค่าอัตราการเปลี่ยนแปลงเท่าไหร่ หลังจากนั้นจึงจะตรวจสอบอัตราการเปลี่ยนแปลงความถี่ที่คำนวณได้ ฟังก์ชันการทำงานนี้สามารถอธิบายได้ดังรูปที่.3.44

มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่.3.44 การทำงานของ Check Present Function

3.4.6 Check Sensitivity Function

เป็นฟังก์ชันการเลือกตัวกำหนดค่าอัตราการเปลี่ยนแปลงความถี่ (Sensitivity) โดยทำการตรวจสอบ Dip Switch ที่อยู่บนตัวตรวจจับยานพาหนะแต่ละ Channel จะมี Dip Switch 2 ช่องเพื่อให้สามารถเลือกค่าอัตราการเปลี่ยนแปลงความถี่ได้ซึ่งการเลือกดังกล่าว ตัวตรวจจับยานพาหนะทั้ง 2 Channel จะมีการตั้งค่าของอัตราการเปลี่ยนแปลงความถี่ที่แยกออกจากกัน ซึ่งพิจารณา Dip Switch 2 จะมีการตั้งค่าการใช้งานได้ 4 แบบดังนี้

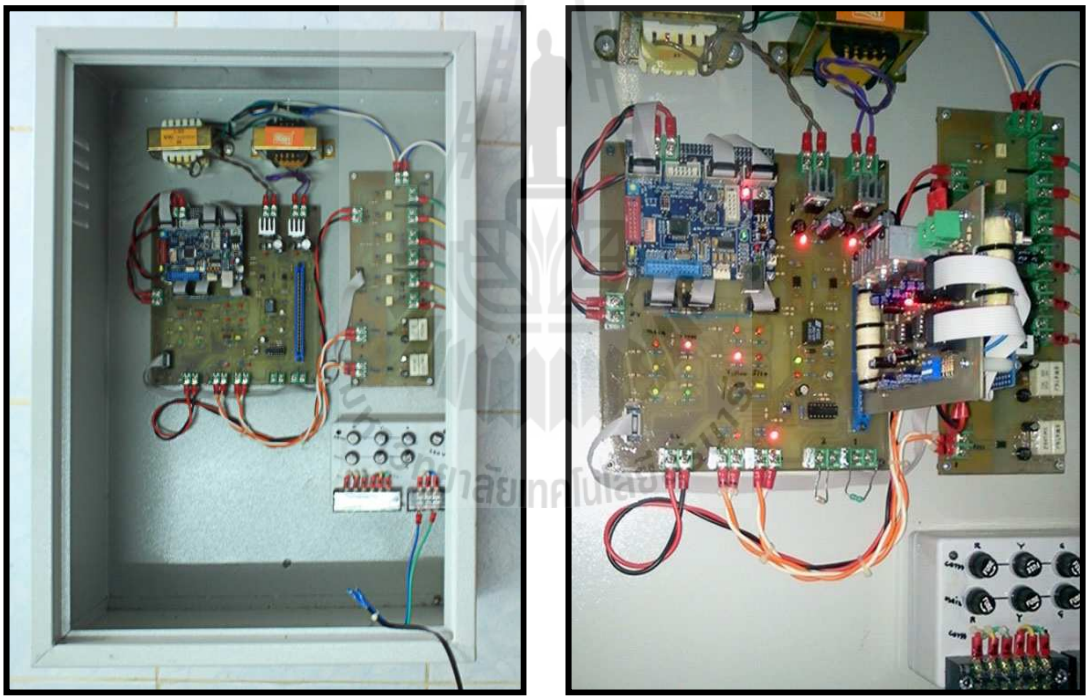
ตารางที่ 3.5 การตั้งค่าอัตราการเปลี่ยนแปลงความถี่ (Sensitivity)

Dip Switch 1	Dip Switch 2	Sensitivity
0	0	Disable Channel
0	1	2%
1	0	1%
1	1	0.5%

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 4
การใช้งานและผลการทดสอบ

การทดสอบระบบควบคุมสัญญาณไฟจราจรนั้นจะแบ่งการทดสอบออกเป็น 2 คือ ส่วนของระบบควบคุมและส่วนของตัวตรวจวัดยานพาหนะแต่การทำงานของระบบทั้งหมดจะต้องมี 2 ส่วนนี้ทำงานร่วมกัน ระบบควบคุมสัญญาณไฟจราจรในงานวิจัยนี้ได้อ้างอิงระบบการควบคุมแบบกึ่งอัตโนมัติ (Sami-Actuated) ใช้ควบคุมสัญญาณไฟจราจรที่มีลักษณะแยกการจราจรที่มีทางเอกและทางโท โดยเลือกสามแยกตรงบริเวณหน้ามหาวิทยาลัยเทคโนโลยีสุรนารี (แยกมาบเอื้อง) เป็นจุดทดสอบผล ซึ่งการทดสอบนั้นจะทำการทดสอบตัวตรวจวัดยานพาหนะและตัวควบคุมสัญญาณไฟจราจร เพื่อจะนำผลการทดสอบที่ได้มาวัดประสิทธิภาพการทำงานเมื่อเปรียบเทียบกับระบบการควบคุมสัญญาณไฟจราจรแบบกำหนดเวลาคงที่ ซึ่งเป็นการควบคุมสัญญาณไฟจราจรแบบเดิม



รูปที่.4.1 ระบบควบคุมสัญญาณไฟจราจรแบบกึ่งอัตโนมัติ

รูปที่.4.2 ภาพถ่ายดาวเทียมแยกการจราจรที่ใช้ทดสอบ

รูปที่.4.3 แยกการจราจรที่ใช้ทดสอบ (1)

รูปที่.4.4 แยกการจราจรที่ใช้ทดสอบ (2)

รูปที่.4.5 ตู้ควบคุมสัญญาณไฟจราจรที่ติดตั้งตรงบริเวณแยกการจราจร

4.1 การทดสอบความถี่เรโซแนนซ์ของหม้อแปลงที่เชื่อมต่อกับลูปเหนี่ยวนำ

ในการใช้หม้อแปลงเพื่อเชื่อมต่อลูปเหนี่ยวนำและวงจรกำเนิดความถี่นั้น จะต้องหาหม้อแปลงที่เหมาะสม เพื่อไม่ได้เกิดการสูญเสียของสัญญาณในการเชื่อมต่อ ซึ่งการทดสอบจะมีขั้นตอนดังนี้

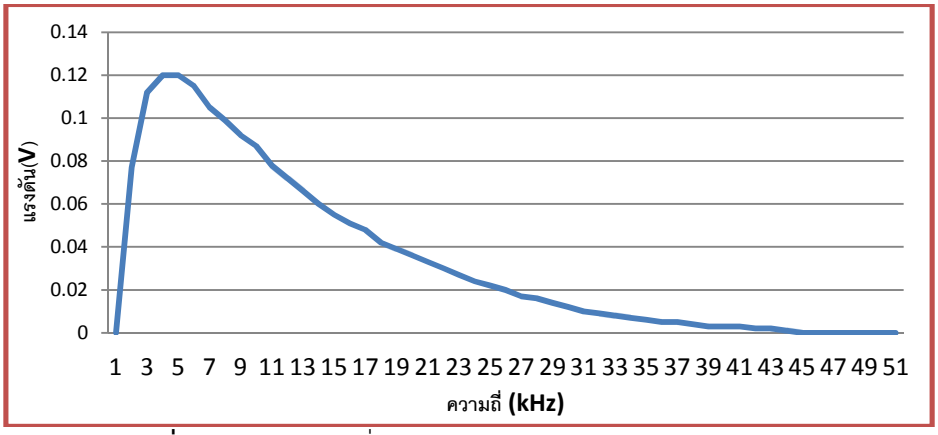
1. ใช้ Function Generator จ่ายไฟฟ้ากระแสสลับด้วยแรงดัน 5 Vp-p เข้ากับด้านปฐมภูมิของหม้อแปลง แล้วใช้ตัวต้านทาน 1 kΩ มาเป็นโหลด ซึ่งจะต่อเข้ากับด้านทุติยภูมิของหม้อแปลง และทำการวัดค่าแรงดันที่ได้จากโหลดตัวต้านทาน 1 kΩ
2. เปลี่ยนความถี่ของ Function Generator โดยจะป้อนความถี่ตั้งแต่ 1 – 100 kHz เพื่อที่จะหาว่าความถี่ที่อยู่ในช่วงใดเป็นความถี่เรโซแนนซ์ของหม้อแปลง

รูปที่.4.6 การทดลองหม้อแปลงด้วยความถี่ต่างๆ

ผลการทดสอบ

หม้อแปลงตัวที่ 1 (1:3)

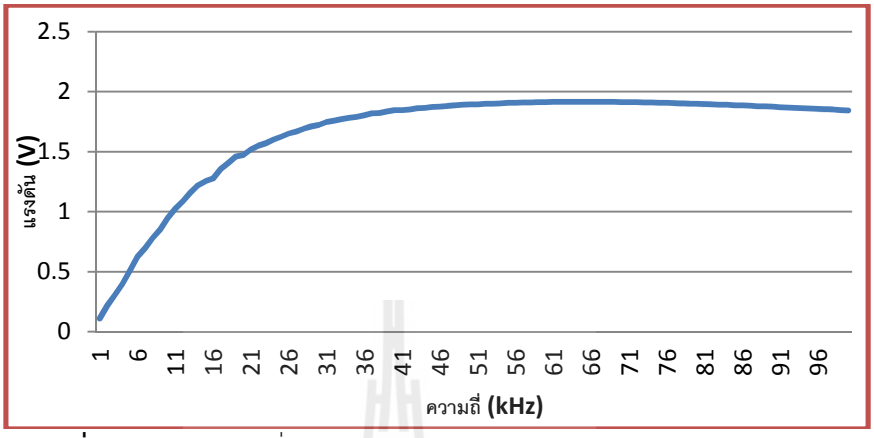
เมื่อต่อโหลดด้วย ตัวต้านทาน 1 kΩ ป้อนอินพุตด้วยความถี่ต่างๆ แล้ววัดแรงดันตกคร่อมที่โหลด ซึ่งจำนวนรอบในการพันขดลวดเท่ากับ 100 : 300 รอบ จะได้ผลดังกราฟ



รูปที่.4.7 กราฟความถี่เรโซแนนซ์ของหม้อแปลง (1:3)

หม้อแปลงตัวที่ 3 (1:1)

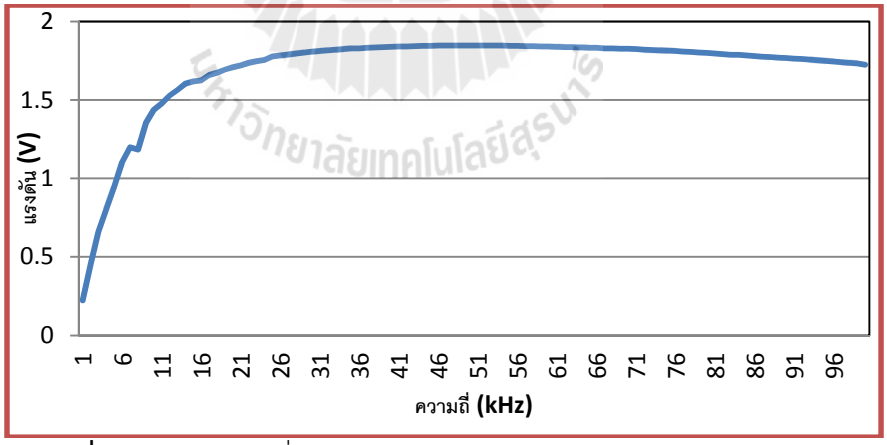
เมื่อต่อโหลดด้วยตัวต้านทาน 1 kΩ ป้อนอินพุตด้วยความถี่ต่างๆ แล้ววัดแรงดันตกคร่อมที่โหลด ซึ่งจำนวนรอบในการพันขดลวดเท่ากับ 150 : 150 รอบ จะได้ผลดังกราฟ



รูปที่.4.8 กราฟความถี่เรโซแนนซ์ของหม้อแปลง (1:1) 150 รอบ

หม้อแปลงตัวที่ 4 (1:1)

เมื่อต่อโหลดด้วย ตัวต้านทาน 1 kΩ ป้อนอินพุตด้วยความถี่ต่างๆ แล้ววัดแรงดันตกคร่อมที่โหลด ซึ่งจำนวนรอบในการพันขดลวดเท่ากับ 100 : 100 รอบ จะได้ผลดังกราฟ

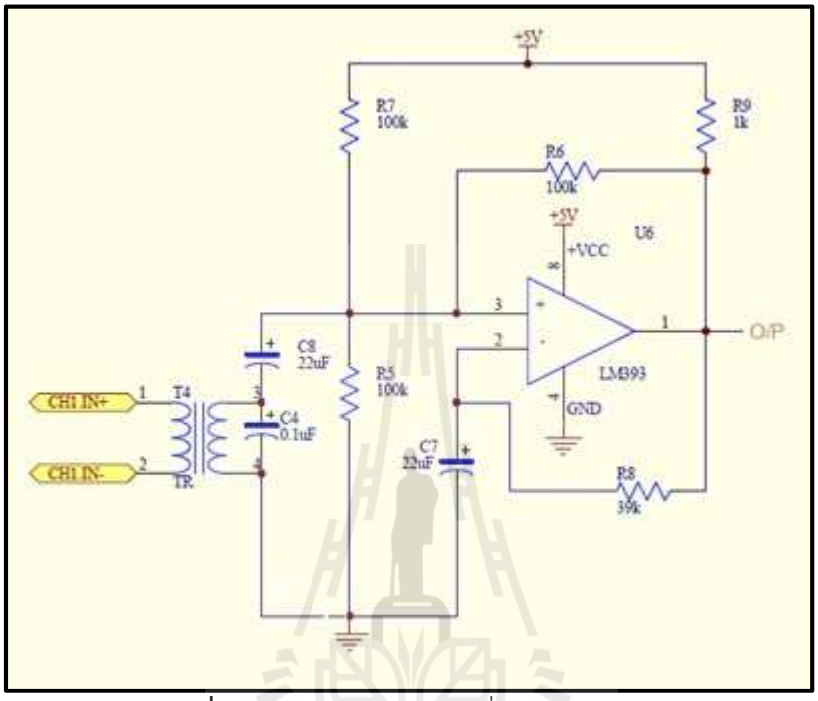


รูปที่.4.9 กราฟความถี่เรโซแนนซ์ของหม้อแปลง (1:1) 100 รอบ

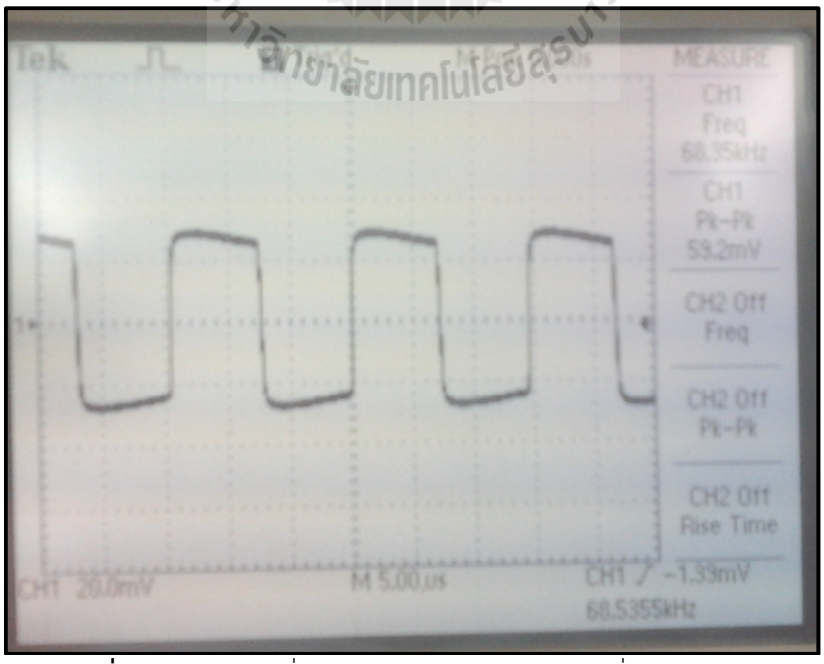
จากการทดสอบพบว่า จำนวนรอบนั้นมีผลต่อช่วงความถี่ที่ใช้งาน โดยที่การพันขดลวดแบบ 1:1 ทำให้เกิดการสูญเสียของช่วงความถี่สูงที่น้อยกว่า ดังนั้นจึงเลือกการพันหม้อแปลงเป็นแบบ 1:1 และจำนวนรอบนั้นเมื่อมีจำนวนรอบที่น้อยกว่า การสูญเสียทางด้านความถี่ต่ำก็จะน้อยกว่าด้วย โดยในงานวิจัยนี้ได้เลือกการพันหม้อแปลงเป็นแบบ 1:1 จำนวน 100 รอบ

4.2 การทดสอบวงจรกำเนิดความถี่ในห้องปฏิบัติการ

การทดสอบวงจรกำเนิดความถี่นั้นจะทำการทดสอบในห้องปฏิบัติการ โดยการใช้ตัวเหนี่ยวนำที่มีค่า 100 μH เพื่อให้เป็นค่าความเหนี่ยวนำแทนลูปเหนี่ยวนำโดยที่วงจรกำเนิดความถี่ที่ได้นั้นจะมีลักษณะเป็นสัญญาณสี่เหลี่ยม (Square wave) ที่ความถี่ 68.35 kHz



รูปที่.4.10 วงจรกำเนิดความถี่และหม้อแปลง

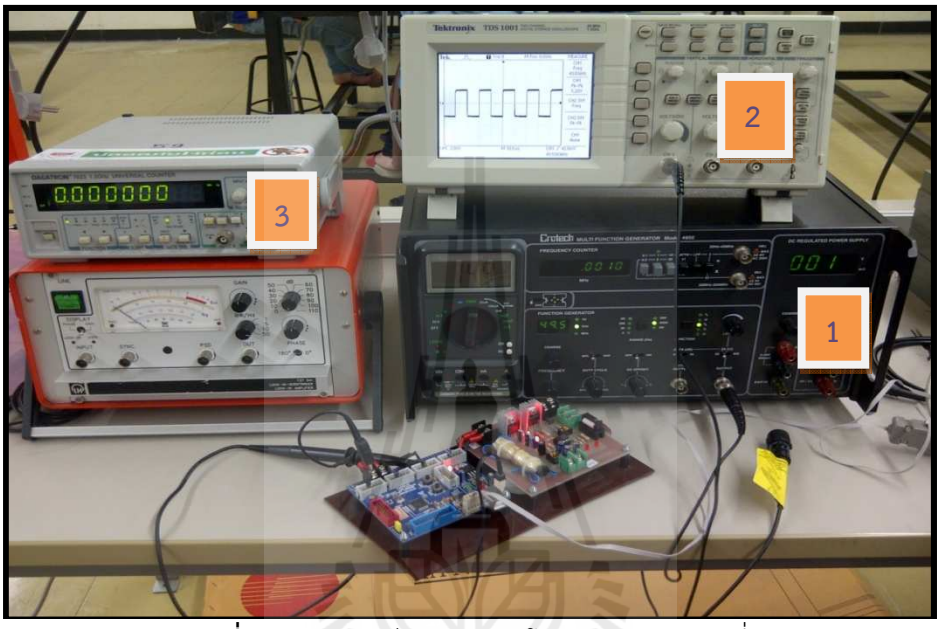


รูปที่.4.11 วัดความถี่โดยใช้ Oscilloscope ความถี่ 68.35 kHz

4.3 การทดสอบโปรแกรมนับความถี่สำหรับตรวจวัดยานพาหนะกับเครื่องมือวัดในห้องปฏิบัติการ

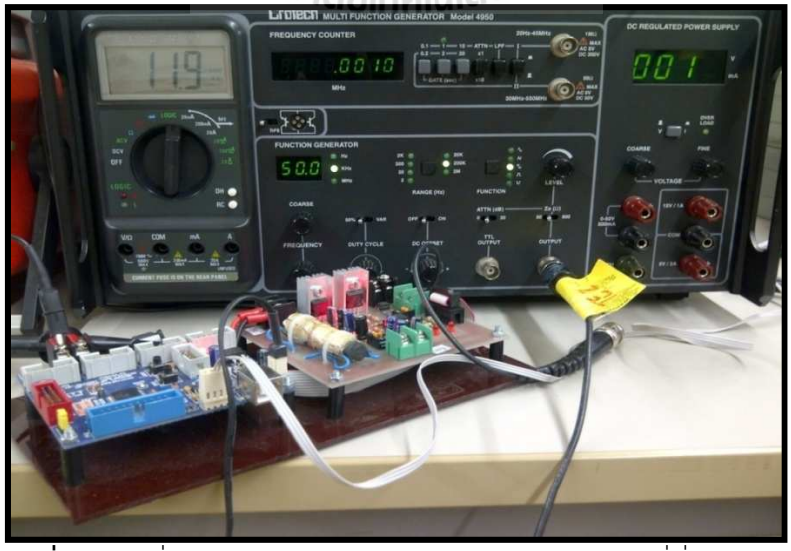
การทดสอบโปรแกรมนับความถี่สำหรับตรวจวัดยานพาหนะกับเครื่องมือวัดในห้องปฏิบัติการประกอบด้วยเครื่องมือต่างๆ ดังนี้

1. เครื่อง Multi Function Generator ใช้สำหรับสร้างความถี่ที่ต้องการทดสอบ
2. เครื่อง Oscilloscope ใช้วัดระดับของสัญญาณและความถี่ที่สามารถจับได้
3. เครื่อง Universal Counter ใช้สำหรับตรวจนับความถี่



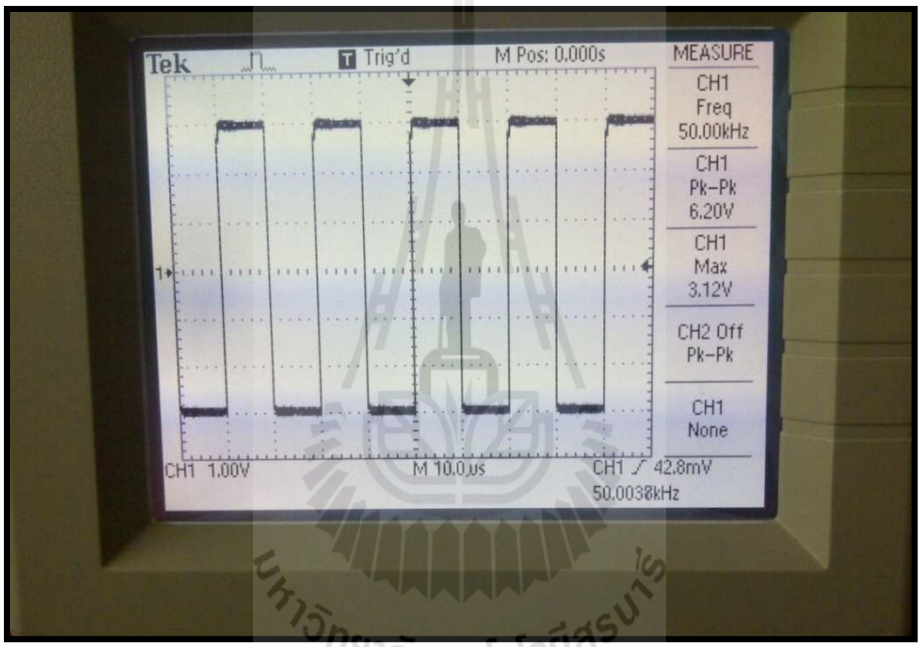
รูปที่.4.12 อุปกรณ์การทดสอบโปรแกรมนับความถี่
สำหรับตรวจวัดยานพาหนะในห้องปฏิบัติการ

ทดสอบโปรแกรมนับความถี่สำหรับตรวจวัดยานพาหนะที่ความถี่ 50 kHz

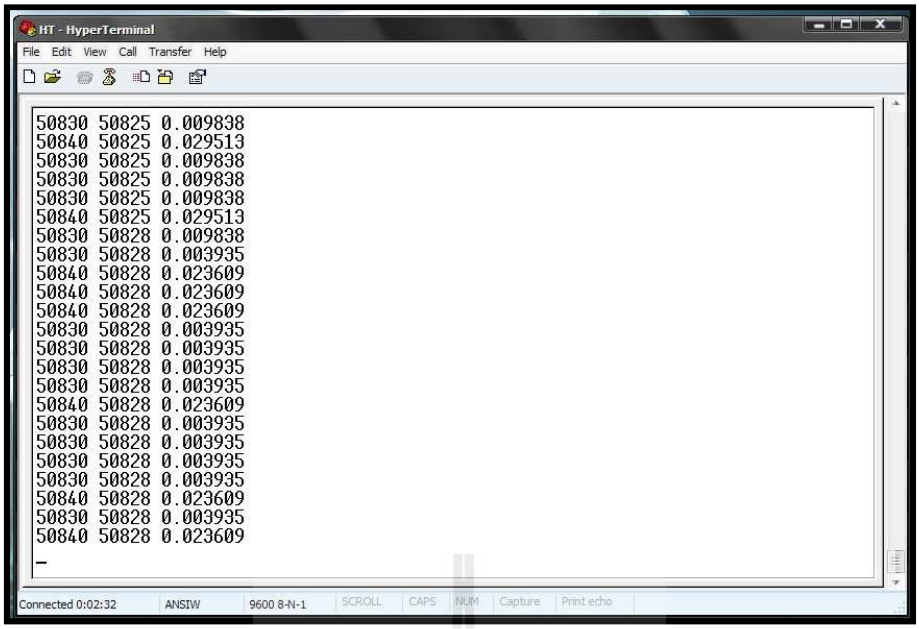


รูปที่.4.13 เครื่อง Multi Function Generator สร้างความถี่ที่ 50 kHz

รูปที่.4.14 เครื่อง Universal Counter อ่านความถี่ได้ที่ 50 kHz



รูปที่.4.15 Oscilloscope วัดความถี่ได้ 50 kHz



รูปที่. 4.16 โปรแกรมนับความถี่สำหรับตรวจวัดยานพาหนะ
อ่านความถี่ที่สร้างมาจากเครื่อง Multi Function Generator ที่ 50 kHz

จากการทดสอบโปรแกรมนับความถี่สำหรับตรวจวัดยานพาหนะ กับเครื่องมือวัดในห้องปฏิบัติการ ทำให้สามารถพิสูจน์ได้ว่า โปรแกรมนับความถี่สามารถอ่านความถี่ได้ใกล้เคียงกับเครื่องมือวัดต่างๆ ซึ่งสามารถนำไมโครคอนโทรลเลอร์มาอ่านความถี่จากวงจรกำเนิดความถี่เพื่อมาวิเคราะห์การตรวจจับยานพาหนะ

มหาวิทยาลัยเทคโนโลยีสุรนารี

4.4 การใช้งานตัวตรวจวัดยานพาหนะ (Detector)

ตัวตรวจวัดยานพาหนะที่ได้ออกแบบนั้นจะใช้หลักการของขดลวดเหนี่ยวนำ โดยจะติดตั้งลูปเหนี่ยวนำ (Loop Detector) ไว้ที่ใต้พื้นถนนตรงจุดที่เราต้องการจะตรวจสอบยานพาหนะ โดยลูปเหนี่ยวนำจะมีลักษณะเป็นวงกลมเส้นผ่านศูนย์กลาง 1.5 เมตร และห่างจากตู้ควบคุมสัญญาณไฟจราจรประมาณ 30 เมตร

รูปที่.4.17 ลูปเหนี่ยวนำที่ติดตั้งที่พื้นถนน

ความถี่ของตัวตรวจวัดยานพาหนะ

ตัวตรวจวัดยานพาหนะที่ออกแบบนั้นจะสามารถเลือกความถี่ได้ 4 ความถี่ โดยการเลือก Dip Switch เพื่อให้ได้ความถี่ตามต้องการและความถี่ที่สามารถเลือกใช้งานมีทั้งหมดดังนี้

ตารางที่ 4.1 ความถี่ของตัวตรวจวัดยานพาหนะ

Dip Switch 1	Dip Switch 2	ความถี่ (kHz)
เปิด	เปิด	46.13
ปิด	เปิด	32.62
เปิด	ปิด	26.63
ปิด	ปิด	23.06

รูปที่.4.18 Dip Switch สำหรับเลือกความถี่ของตัวตรวจวัดยานพาหนะ

ความไวของตัวตรวจวัดยานพาหนะ

ในการตรวจวัดยานพาหนะ ความไวเป็นพารามิเตอร์ที่สำคัญอย่างยิ่งที่จะบ่งบอกถึงประสิทธิภาพในการตรวจวัดยานพาหนะ เพราะถ้ายิ่งมีความไวในการตรวจสอบมาก จะทำให้สามารถตรวจวัดยานพาหนะขนาดเล็กได้ ซึ่งในงานวิจัยนี้ได้ออกแบบความไวในการตรวจสอบยานพาหนะไว้ 3 ระดับซึ่งสามารถเลือกความไวในการตรวจสอบโดยการเลือกจาก Dip Swatch ซึ่งจะสามารถเลือกได้ดังนี้

ตารางที่ 4.2 ความไวสำหรับตรวจวัดยานพาหนะ

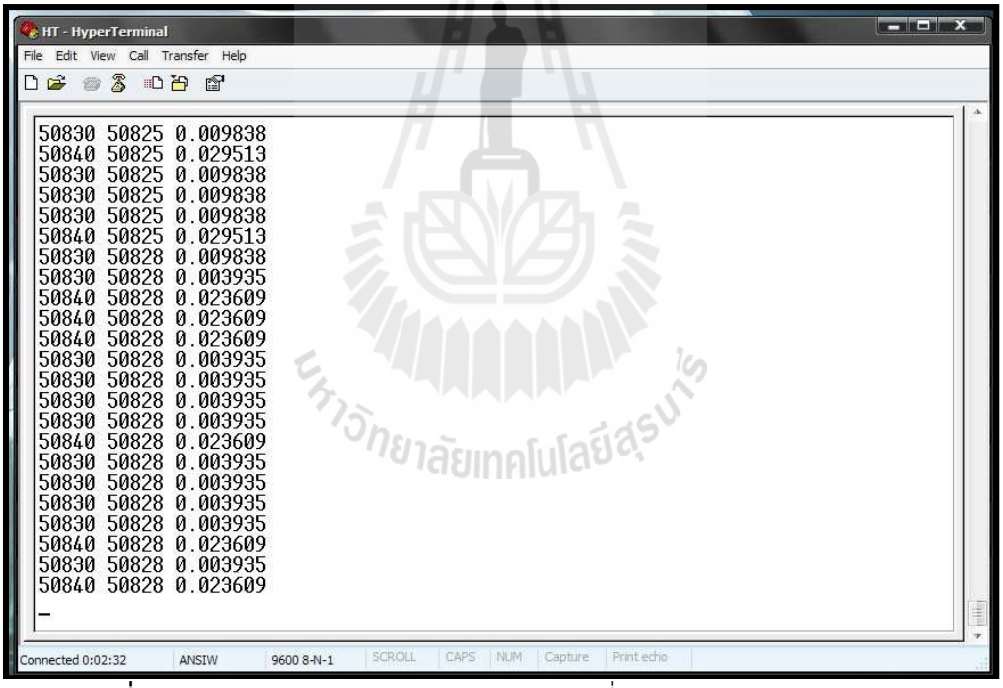
Dip Switch				Sensitivity	
1	2	3	4	Channel 1	Channel 2
OFF	OFF	-	-	Disable	-
OFF	ON	-	-	2%	-
ON	OFF	-	-	1%	-
ON	ON	-	-	0.5%	-
-	-	OFF	OFF	-	Disable
-	-	OFF	ON	-	2%
-	-	ON	OFF	-	1%
-	-	ON	ON	-	0.5%

ความไว 0.5% หมายถึง การตรวจสอบอัตราการเปลี่ยนแปลงความถี่ของตัวตรวจวัดยานพาหนะ ซึ่งความไวที่ 0.5% ถือว่าเป็นความไวที่สูงที่สุด เพราะว่าสามารถตรวจวัดยานพาหนะ ขนาดเล็กได้

รูปที่.4.19 Dip Switch สำหรับเลือกความไวของตัวตรวจวัดยานพาหนะ

การเชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์และโปรแกรม Hyper Terminal

บอร์ดวงจรกำเนิดความถี่จะใช้ไมโครคอนโทรลเลอร์ในการนับความถี่และแสดงผลออกทางคอมพิวเตอร์ผ่านโปรแกรม Hyper Terminal ซึ่งสามารถที่จะเก็บข้อมูลความถี่ที่ได้จากตัวตรวจวัดยานพานะนั้นเพื่อมาวิเคราะห์ผลต่อไปได้

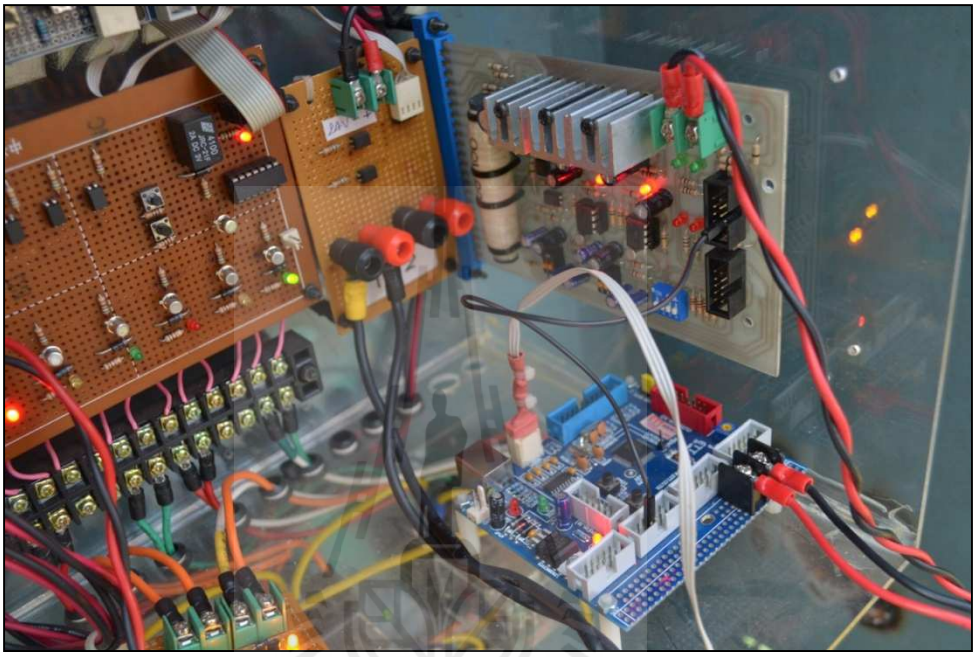


รูปที่.4.20 แสดงตัวอย่างของการแสดงผลความถี่ของตัวตรวจวัดยานพาหนะ
ผ่านทางโปรแกรม Hyper Terminal

มหาวิทยาลัยเทคโนโลยีสุรนารี

4.5 การทดสอบการตรวจวัดยานพาหนะ

หลักการทำงานของตัวตรวจวัดยานพาหนะ คือ การสร้างความถี่ที่มีค่าคงที่เมื่อไม่มียานพาหนะ แต่เมื่อมียานพาหนะเข้ามาที่ลูปเหนี่ยวนำก็จะทำให้ความถี่เปลี่ยนไป ซึ่งลูปเหนี่ยวนำจะเป็นเซ็นเซอร์ในการตรวจวัดยานพาหนะ โดยอัตราการเปลี่ยนแปลงความถี่ที่เกิดขึ้นจะนำมาตัดสินใจว่ามียานพาหนะมารอจริงหรือไม่

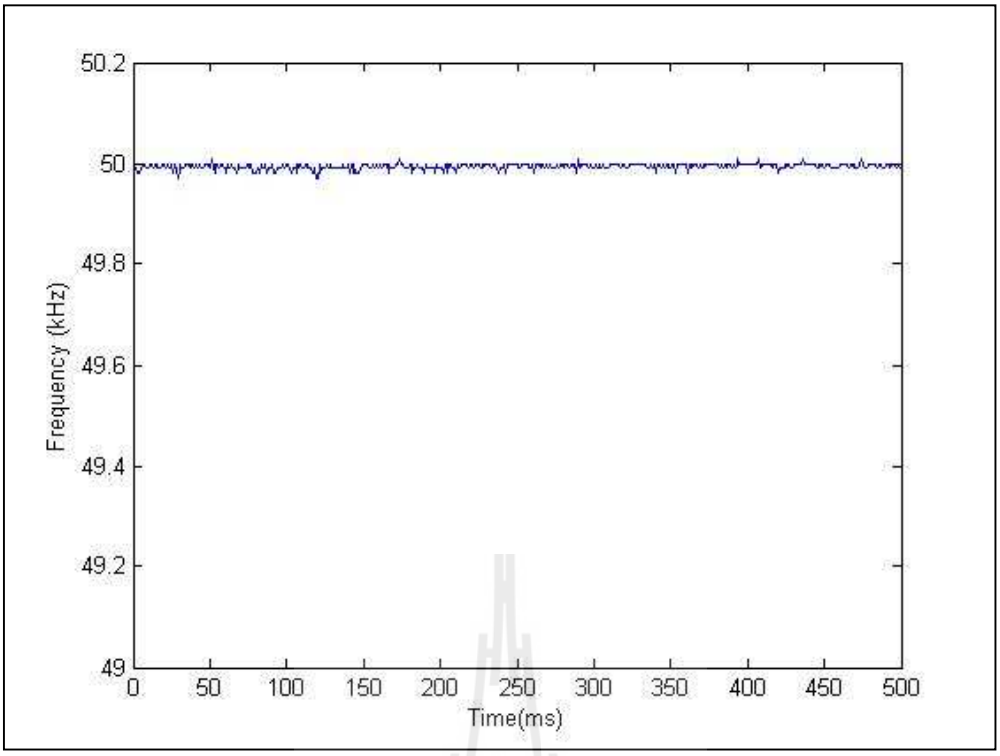


รูปที่.4.21 อุปกรณ์สำหรับทดสอบตรวจจับยานพาหนะ

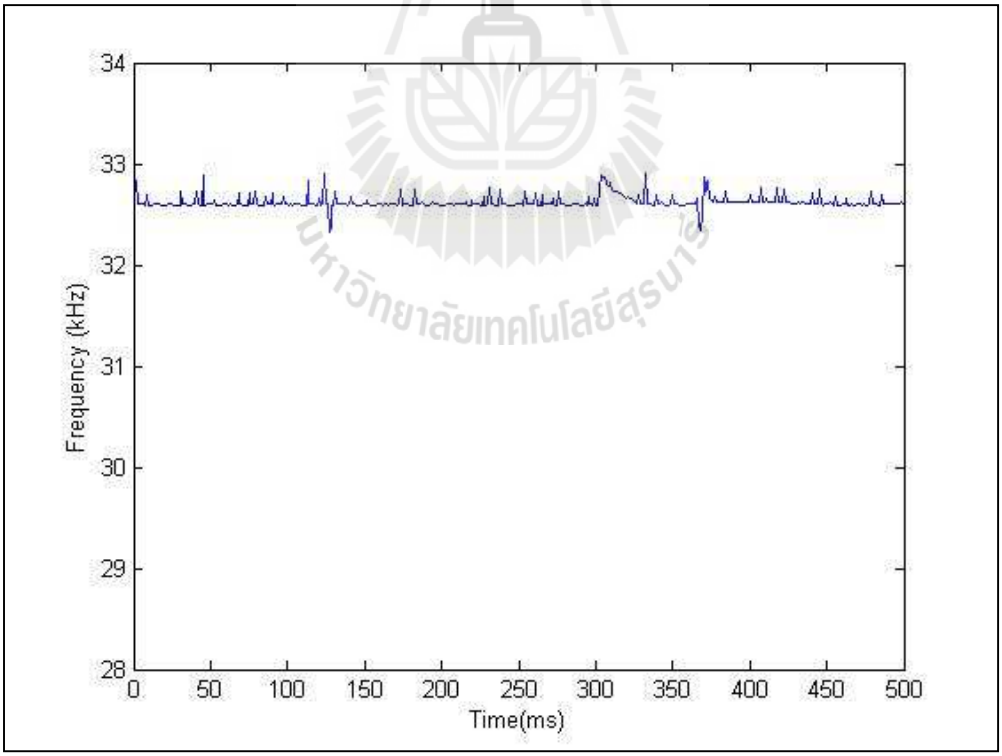
จากการทดสอบวงจรกำเนิดความถี่ขณะที่ทำการเชื่อมต่อเข้ากับลูปเหนี่ยวนำขนาดขนาด 119.03 μH ความถี่ที่ได้จากการปรับสวิตช์ทั้ง 4 ระดับ คือ

1. เปิด-เปิด (00) จะได้ความถี่ประมาณ 48.85 kHz
2. เปิด-ปิด (01) จะได้ความถี่ประมาณ 33.98 kHz
3. ปิด-เปิด (10) จะได้ความถี่ประมาณ 26.26 kHz
4. ปิด-ปิด (11) จะได้ความถี่ประมาณ 22.86 kHz

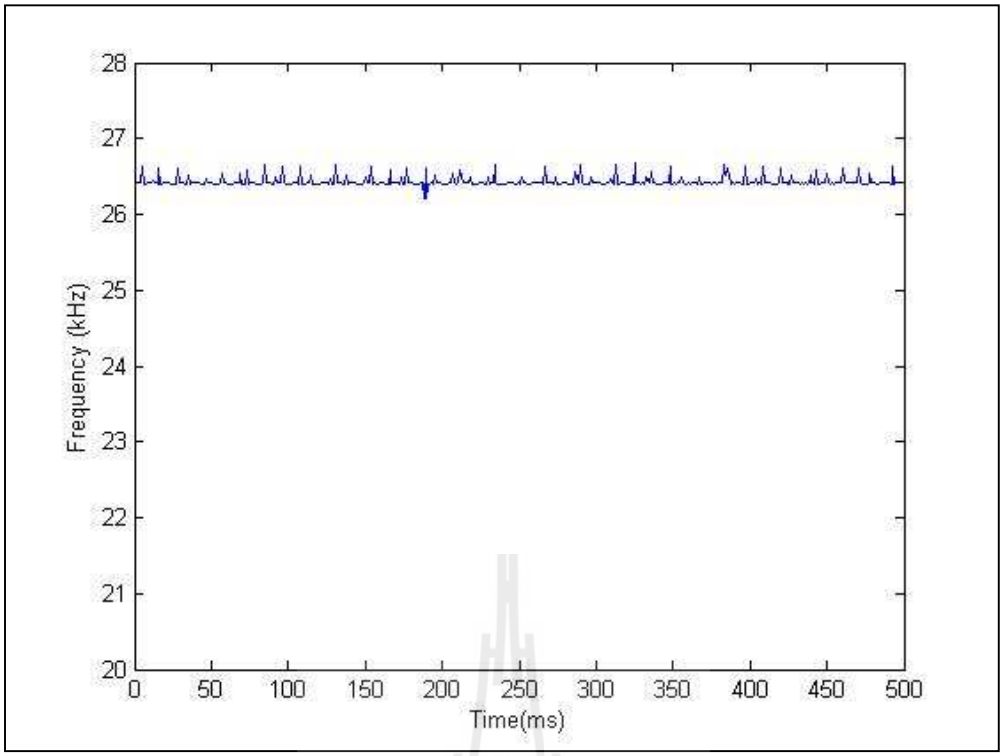
จากหลักการการทำงานของตัวตรวจวัดยานพาหนะจะเห็นได้ว่าถ้าความถี่ที่ได้จากวงจรกำเนิดสัญญาณในขณะที่ไม่มียานพาหนะไม่มีค่าคงที่จะทำให้เกิดการเปลี่ยนแปลงความถี่อยู่ตลอดเวลาทำให้อัตราการเปลี่ยนแปลงความถี่ (ΔF) มีค่าสูง ไม่สามารถที่จะตรวจสอบได้ ดังนั้นจึงทำการทดสอบความถี่ที่เกิดขึ้นในขณะที่ไม่มียานพาหนะเพื่อหาอัตราการเปลี่ยนแปลงความถี่ในขณะที่ไม่มียานพาหนะ



รูปที่.4.22 ผลการทดสอบวัดความถี่ 50 kHz ในขณะที่ไม่มียานพาหนะ



รูปที่.4.23 ผลการทดสอบวัดความถี่ 32 kHz ในขณะที่ไม่มียานพาหนะ



รูปที่.4.24 ผลการทดสอบวัดความถี่ 26 kHz ในขณะที่ไม่มียานพาหนะ

จากผลการทดสอบพบว่า ความถี่ทั้ง 3 ความถี่มีอัตราการเปลี่ยนแปลงความถี่ในขณะที่ไม่มียานพาหนะที่ต่ำ ดังนั้นจึงสามารถใช้งานสำหรับการตรวจวัดยานพาหนะได้

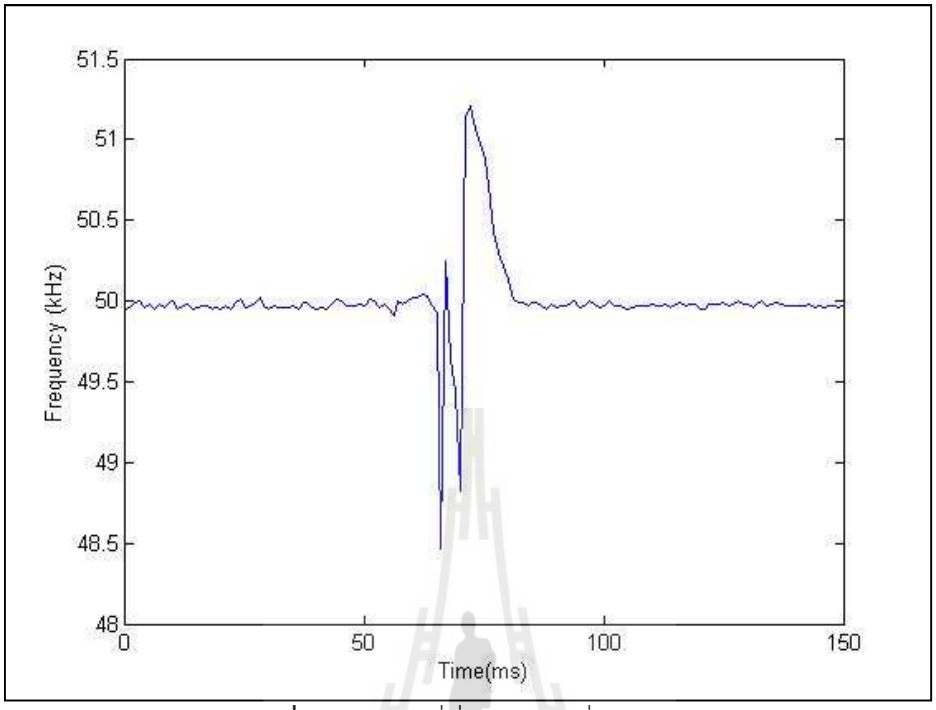
ทดสอบความถี่ในขณะที่มียานพาหนะ

จากความถี่ที่เกิดการเปลี่ยนแปลงไปเมื่อมียานพาหนะเข้ามาอยู่ที่ลูปเหนี่ยวนำ ซึ่งความถี่ที่ได้จะนำมาหาอัตราการเปลี่ยนแปลงความถี่ (ΔF) เพื่อที่จะนำมาตีความว่าเป็นยานพาหนะจริงหรือไม่ ความถี่ที่เปลี่ยนแปลงนั้นเกิดจากยานพาหนะเข้ามาอยู่ตรงจุดที่ติดตั้งลูปเหนี่ยวนำเอาไว้ ทำให้ค่าความเหนี่ยวนำเพิ่มขึ้นและจะทำให้ความถี่เปลี่ยนแปลงตามไปด้วย ซึ่งการทดสอบจะทำการทดสอบกับยานพาหนะประเภทรถยนต์ส่วนบุคคลและยานพาหนะขนาดเล็ก เช่น รถจักรยานยนต์ ซึ่งอัตราการเปลี่ยนแปลงที่เกิดขึ้นจากรถยนต์ส่วนบุคคลจะมีค่ามากกว่า เพราะขนาดของยานพาหนะมีขนาดใหญ่กว่า

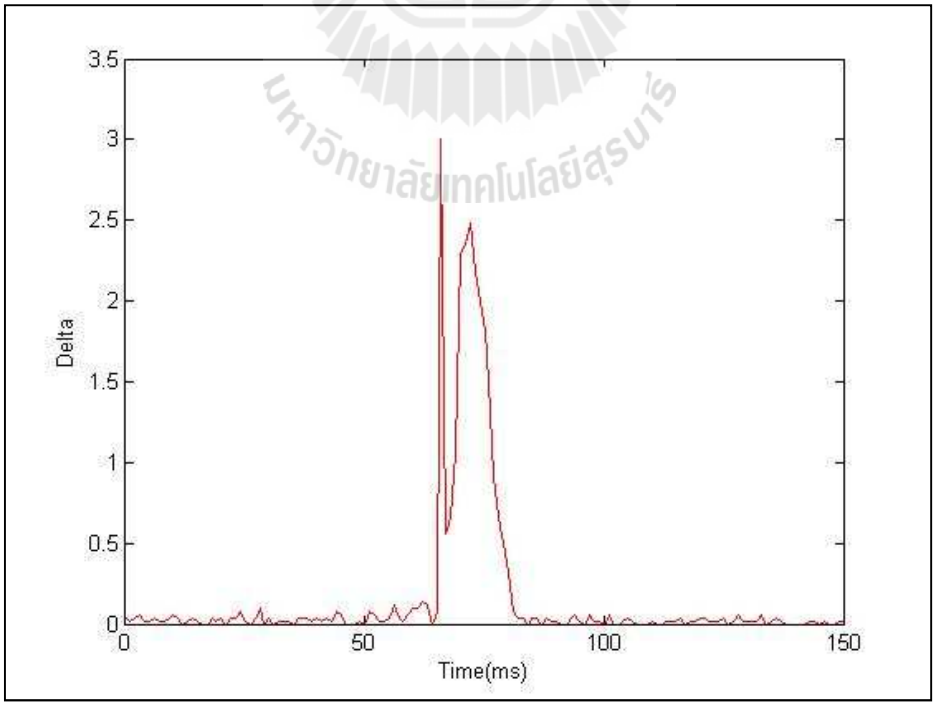
รูปที่.4.25 ลักษณะยานพาหนะขนาดเล็กอยู่บนลูปเหนี่ยวนำ

รูปที่.4.26 ลักษณะยานพาหนะประเภทรถยนต์ส่วนบุคคล
อยู่บนลูปเหนี่ยวนำ

ทดสอบตรวจวัดยานพาหนะประเภทรถยนต์ส่วนบุคคลที่ความถี่ 50 kHz

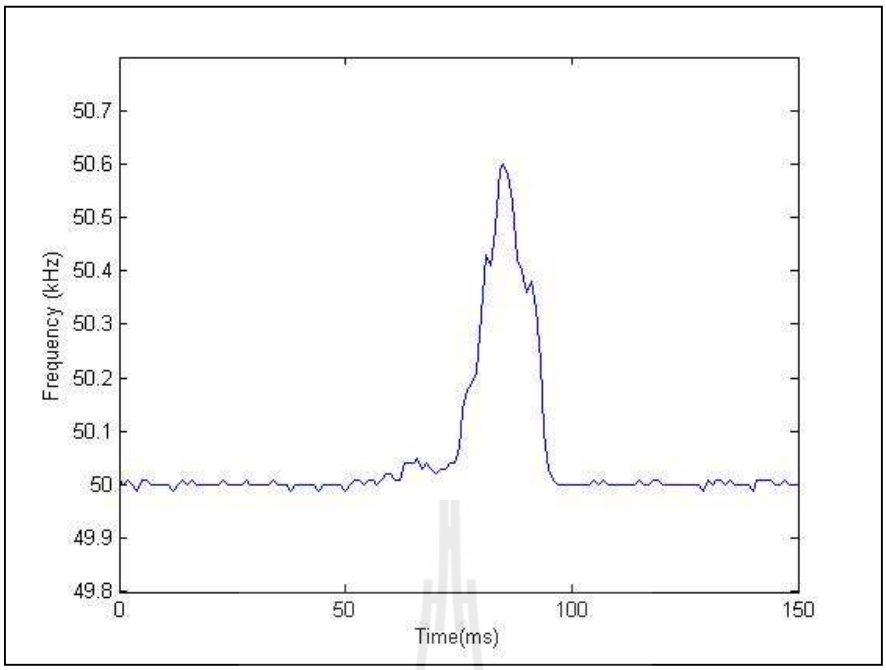


รูปที่.4.27 ความถี่ที่เกิดการเปลี่ยนแปลง
เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (50 kHz)

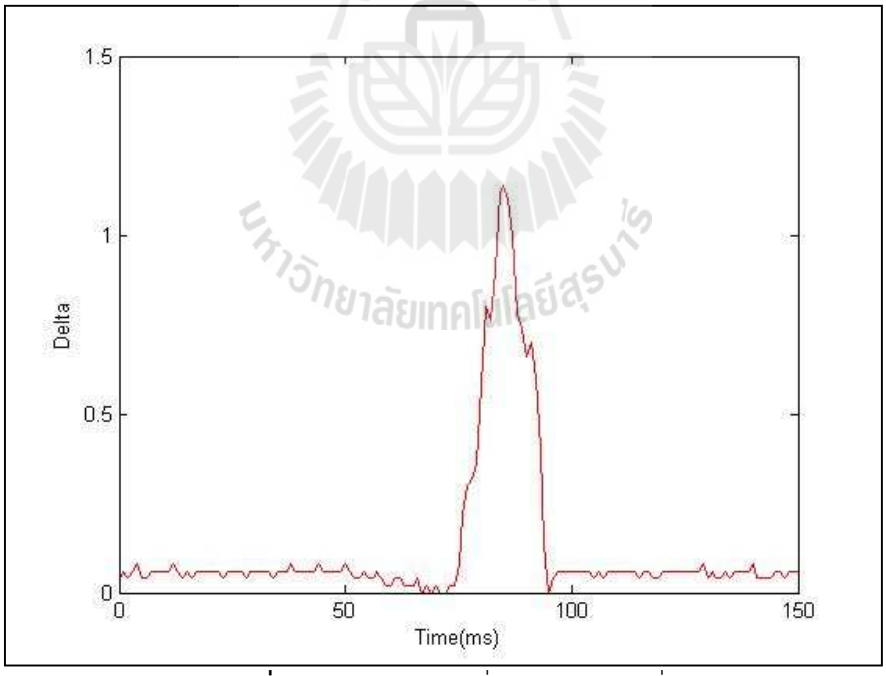


รูปที่.4.28 อัตราการเปลี่ยนแปลงความถี่
เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (50 kHz)

ทดสอบตรวจวัดยานพาหนะประเภทรถจักรยานยนต์ที่ความถี่ 50 kHz

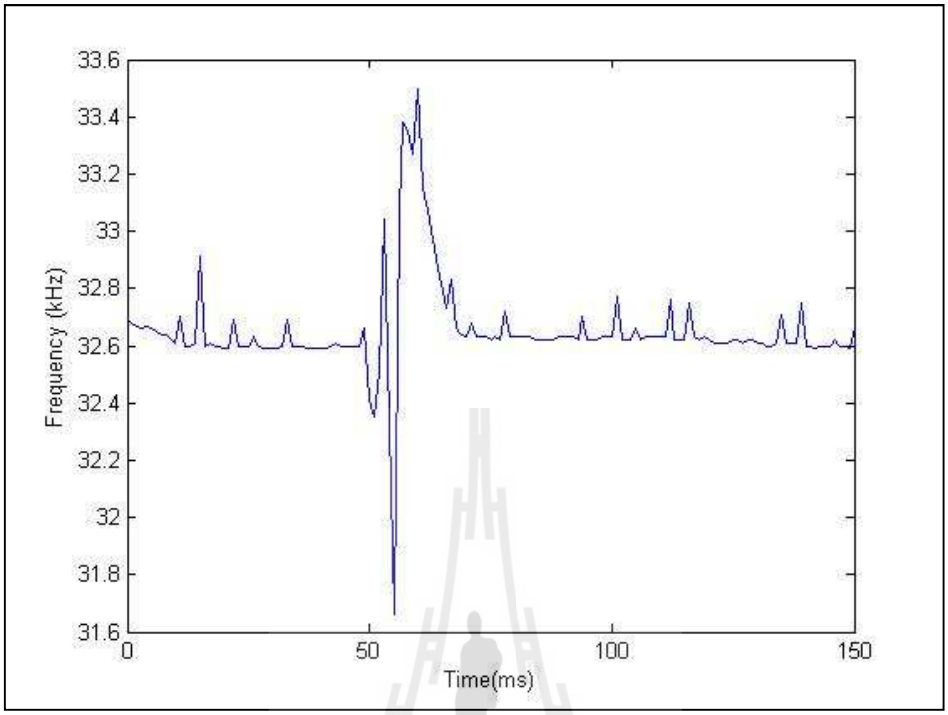


รูปที่.4.29 ความถี่ที่เกิดการเปลี่ยนแปลง
เมื่อมีรถจักรยานยนต์อยู่ที่ลูปเหนี่ยวนำ (50 kHz)

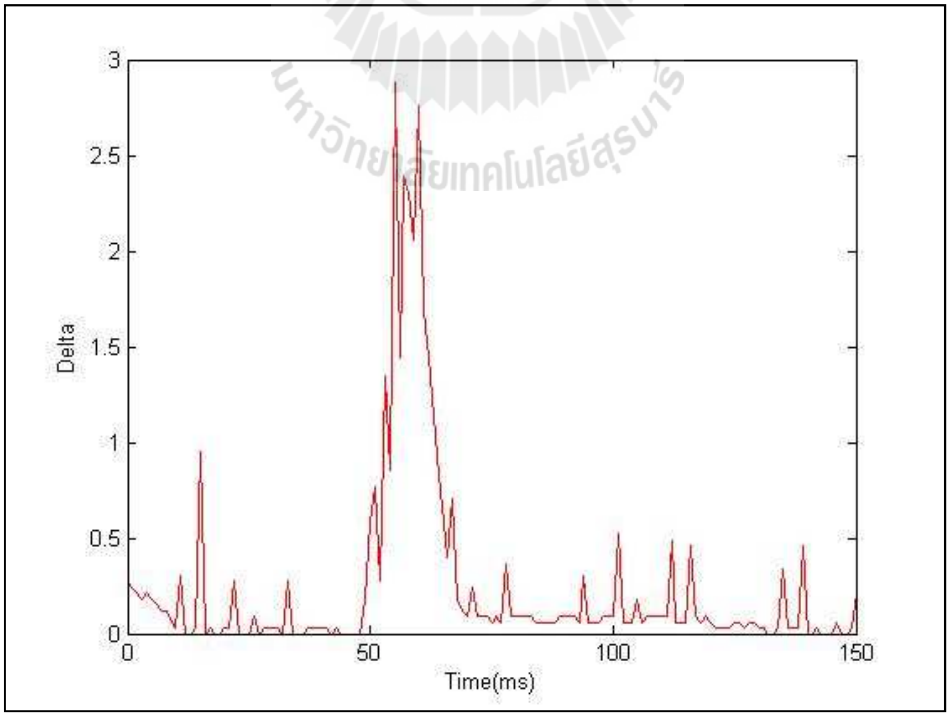


รูปที่.4.30 อัตราการเปลี่ยนแปลงความถี่
เมื่อมีรถจักรยานยนต์อยู่ที่ลูปเหนี่ยวนำ (50 kHz)

ทดสอบตรวจวัดยานพาหนะประเภทรถยนต์ส่วนบุคคลที่ความถี่ 32 kHz

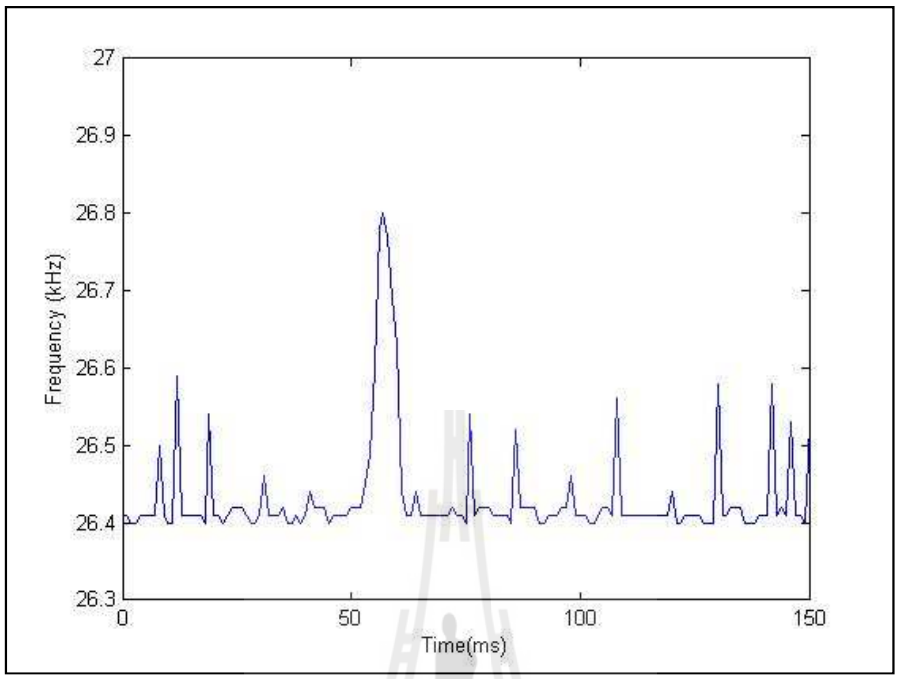


รูปที่.4.31 ความถี่ที่เกิดการเปลี่ยนแปลง
เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (32 kHz)

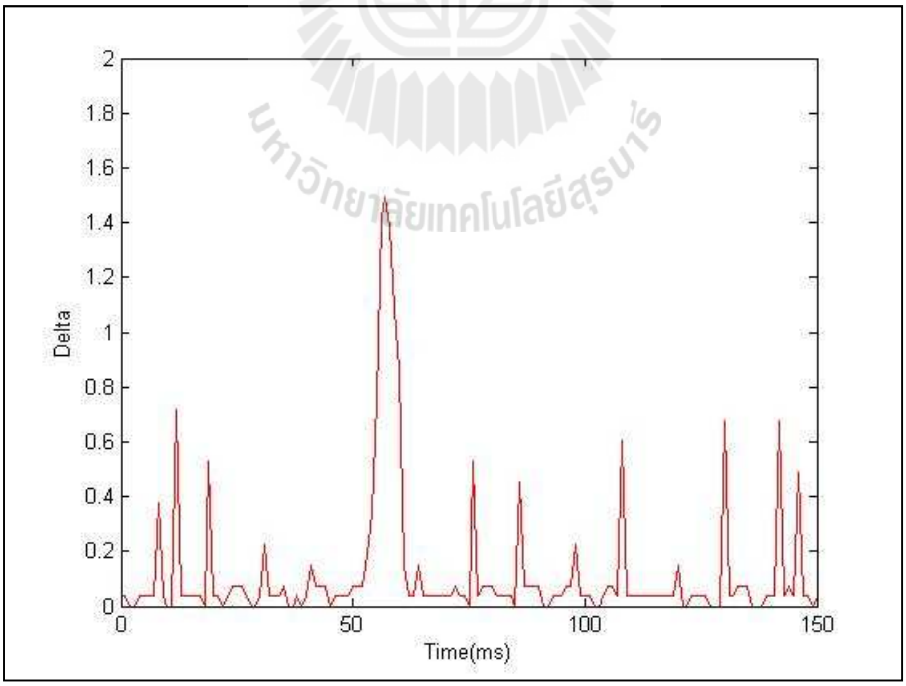


รูปที่.4.32 อัตราการเปลี่ยนแปลงความถี่
เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (32 kHz)

ทดสอบตรวจวัดยานพาหนะประเภทรถจักรยานยนต์ที่ความถี่ 32 kHz

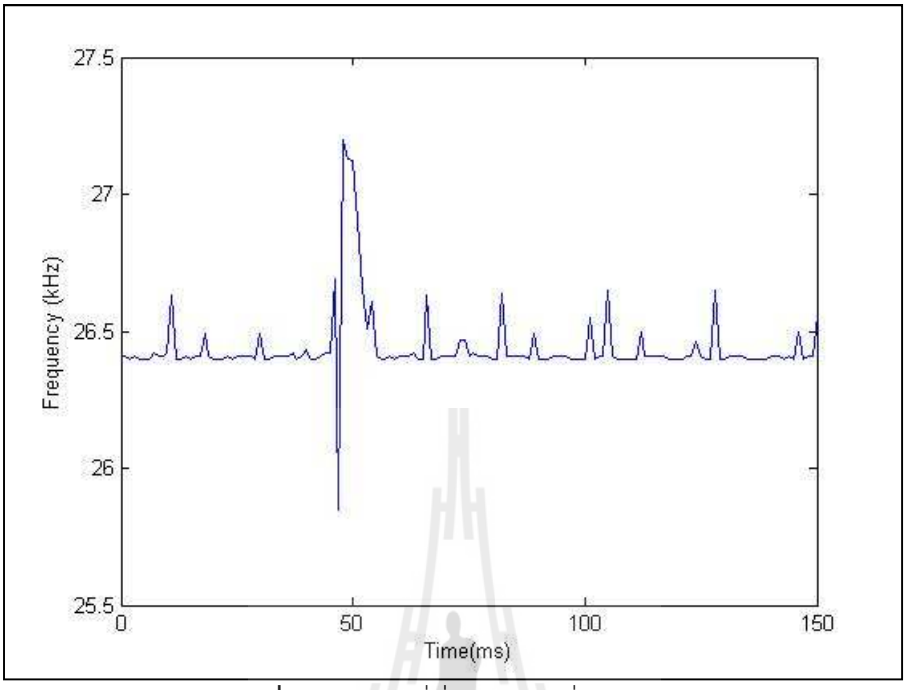


รูปที่.4.33 ความถี่ที่เกิดการเปลี่ยนแปลง
เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (32 kHz)

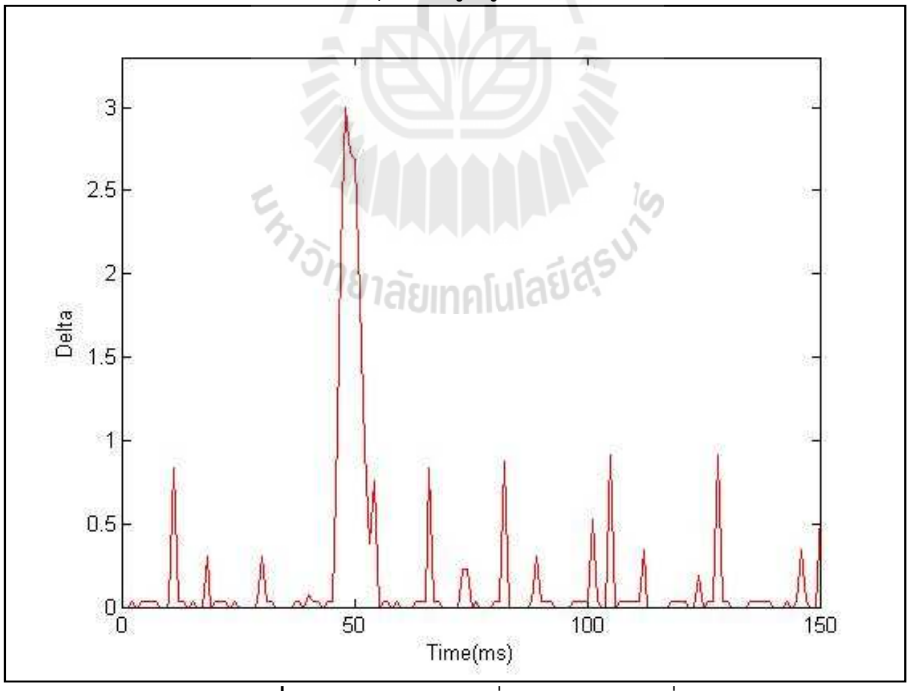


รูปที่.4.34 อัตราการเปลี่ยนแปลงความถี่
เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (32 kHz)

ทดสอบตรวจวัดยานพาหนะประเภทรถยนต์ส่วนบุคคลที่ความถี่ 26 kHz

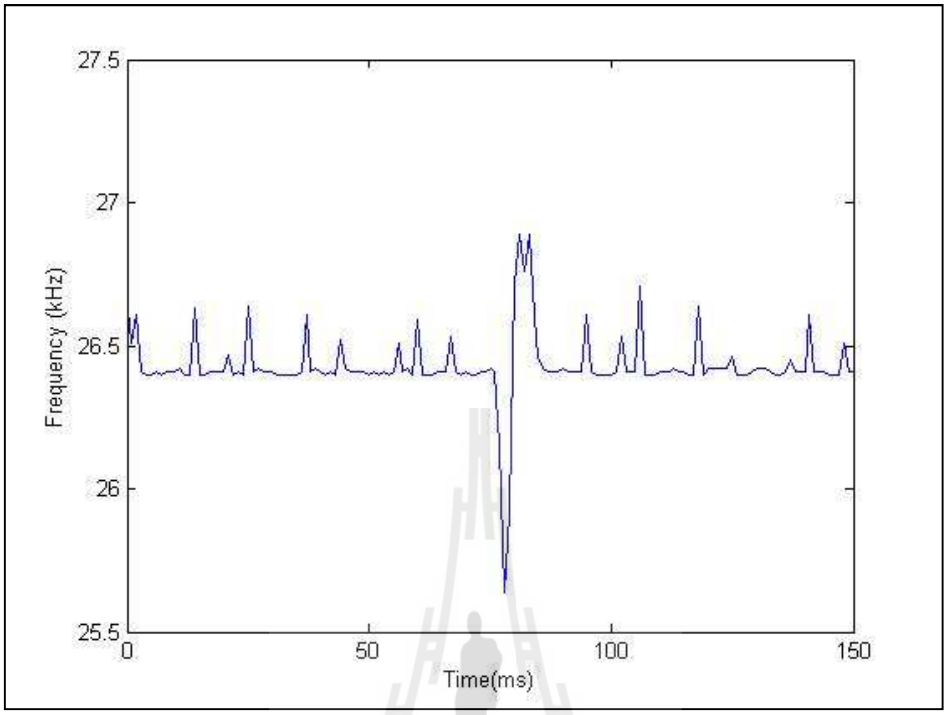


รูปที่.4.35 ความถี่ที่เกิดการเปลี่ยนแปลง
เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (26 kHz)

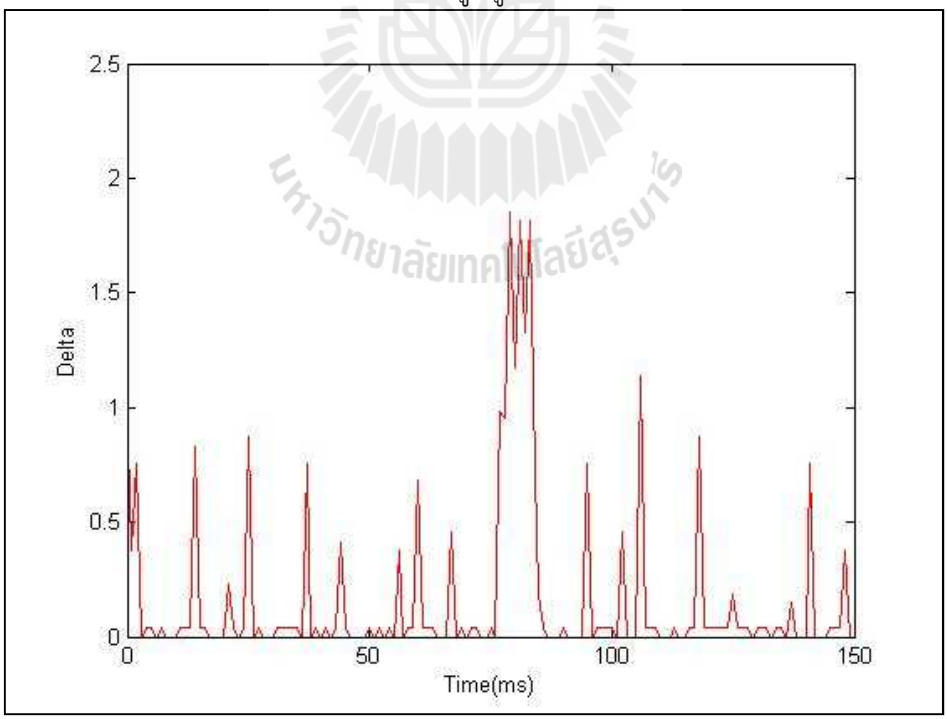


รูปที่.4.36 อัตราการเปลี่ยนแปลงความถี่
เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (26 kHz)

ทดสอบตรวจวัดยานพาหนะประเภทรถจักรยานยนต์ที่ความถี่ 26 kHz



รูปที่.4.37 ความถี่ที่เกิดการเปลี่ยนแปลง
เมื่อมีรถจักรยานยนต์อยู่ที่ลูปเหนี่ยวนำ (26 kHz)



รูปที่.4.38 อัตราการเปลี่ยนแปลงความถี่
เมื่อมีรถยนต์ส่วนบุคคลอยู่ที่ลูปเหนี่ยวนำ (26 kHz)

จากการทดสอบตรวจวัดยานพานะทั้งหมดสามารถสรุปได้ตามตารางดังนี้

ตารางที่ 4.3 ตารางแสดงผลการตรวจวัดประเภทรถยนต์ส่วนบุคคล

ความถี่ตอนไม่มียานพาหนะ (kHz)	**ความถี่เมื่อมียานพาหนะ (kHz)**	**เปอร์เซ็นต์การเปลี่ยนแปลง (%)**
26.26	26.4-26.91	0.53-2.47
33.98	34.19-34.56	0.62-1.71
48.85	49.14-50.02	0.59-2.4

ตารางที่ 4.4 ตารางแสดงผลการตรวจวัดประเภทรถจักรยานยนต์

ความถี่ตอนไม่มียานพาหนะ (kHz)	**ความถี่เมื่อมียานพาหนะ (kHz)**	**เปอร์เซ็นต์การเปลี่ยนแปลง (%)**
26.26	26.35-26.41	0.34-0.57
33.95	34.13-34.19	0.53-0.71
48.85	49.10-49.18	0.51-0.67

จากการทดสอบทั้ง 3 ความถี่กับยานพาหนะขนาดเล็กและยานพาหนะประเภทรถยนต์ส่วนบุคคล พบว่าเมื่อยานพาหนะเข้ามาอยู่บนจุดที่ติดตั้งลูปเหนี่ยวนำจะทำให้เกิดความถี่เปลี่ยนแปลงไปจากเดิม เมื่อคำนวณหาอัตราการเปลี่ยนแปลงความถี่พบว่า ยานพาหนะประเภทรถยนต์ส่วนบุคคลจะมีอัตราการเปลี่ยนแปลงเฉลี่ยอยู่ที่ 1.39% และยานพาหนะขนาดเล็กจะมีอัตราการเปลี่ยนแปลงเฉลี่ยอยู่ที่ 0.56% ดังนั้นเมื่อเราพิจารณาความไวในการตรวจวัดยานพาหนะเราสามารถตรวจวัดยานพาหนะขนาดเล็กได้ ทั้งนี้เนื่องจากความสามารถออกอุปกรณ์ได้ออกไว้ให้มีความไวสูงสุดที่ 0.5%

จากการทดสอบสรุปผลการทดสอบของจำนวนยานพาหนะที่ตรวจวัดได้ดังนี้

ตารางที่ 4.5 สรุปผลการตรวจวัดยานพาหนะ

ความถี่ (kHz)	รถยนต์ส่วนบุคคล		รถจักรยานยนต์	
	ทั้งหมด	ตรวจวัดได้	ทั้งหมด	ตรวจวัดได้
48.85	7 คัน	7 คัน	4 คัน	3 คัน
33.98	7 คัน	7 คัน	4 คัน	2 คัน
26.26	7 คัน	7 คัน	4 คัน	2 คัน

จากตารางสรุปผลการตรวจวัดยานพาหนะสามารถสรุปได้สามารถตรวจวัดยานพาหนะได้ แต่ความสามารถในการตรวจวัดก็มีข้อจำกัด คือ ในกรณีรถจักรยานยนต์ไม่สามารถตรวจวัดได้ทั้งหมดเนื่องจากรถจักรยานยนต์นั้นมีส่วนประกอบที่เป็นโลหะน้อย ทำให้ค่าความเหนี่ยวนำของลูปเหนี่ยวนำเปลี่ยนไปน้อย ส่งผลให้ความถี่ที่ได้จากวงจรกำเนิดความถี่เปลี่ยนไปน้อยด้วย ซึ่งถ้าความถี่ที่เปลี่ยนไปมีค่าน้อย ไมโครคอนโทรลเลอร์จะไม่สามารถวิเคราะห์ได้ว่าขณะนั้นมียานพาหนะมาหยุดหรือผ่านจุดที่ทดสอบ

4.6 การใช้งานควบคุมสัญญาณไฟจราจร (Controller)

ตัวควบคุมสัญญาณไฟจราจรนั้นเป็นส่วนควบคุมหลัก ซึ่งจะทำหน้าที่สั่งงานสัญญาณไฟจราจรที่อยู่บนท้องถนน ซึ่งการควบคุมสัญญาณไฟจราจรนั้นอ้างอิงระบบควบคุมแบบกึ่งอัตโนมัติ โดยจะใช้ตัวตรวจวัดยานพาหนะติดตั้งเอาไว้ที่ทางโทเพื่อตรวจวัดยานพาหนะที่มารอ จากที่ได้กล่าวมาแล้วในบทที่ 3 ระบบควบคุมสัญญาณไฟจราจรที่ได้ออกแบบนั้นได้แบ่งโหมดการทำงานออกเป็น 2 โหมด คือ โหมด Fast และโหมด Slow ในการทำงานของทั้ง 2 โหมดนั้นจะมีการทำงานที่เหมือนกันคือ จะต้องมียานพาหนะในทางโทมากระตุ้นระบบจึงจะเกิดการทำงาน สิ่งที่แตกต่างกันของทั้ง 2 โหมดคือ ระยะเวลาของยานพานะที่มารอในทิศทางโทเพื่อที่จะผ่านแยกการจราจรและระยะเวลาของการสัญญาณไฟเขียวรอบต่อไปในทางโท

การเลือกโหมดการทำงานของตัวควบคุมสัญญาณไฟจราจรสามารถเลือกได้จากการเลือกจุดเชื่อมต่อ (Jumper) Fast / Slow บนบอร์ดควบคุมสัญญาณไฟจราจรโดยที่เมื่อมีการเชื่อมต่อจะเป็นการเลือกใช้งานโหมด Fast (LED Fast/Slow สว่าง) แต่เมื่อไม่ได้มีการเชื่อมต่อจะเป็นการใช้งานโหมด Slow (LED Fast/Slow ดับ)

4.7 การทดสอบตัวควบคุมสัญญาณไฟจราจร (Controller)

จากที่ได้ออกแบบระบบควบคุมสัญญาณไฟจราจรเป็นแบบกึ่งอัตโนมัติเพื่อที่จะใช้งานควบคุมแยกการจราจรที่เป็นทางเอกและทางโทนั้นผลการทดสอบเป็นดั้งนี้

1. เมื่อไม่มียานพาหนะในทางโท ทางเอกจะได้สัญญาณไฟเขียวอยู่ตลอดเวลา
2. เมื่อมียานพาหนะมาหยุดรอที่ทางโทเป็นเวลามากกว่า 5 วินาที (สำหรับโหมด Fast) สัญญาณไฟจราจรจะเกิดการเปลี่ยนแปลง โดยเป็นมาให้สัญญาณไฟเขียวที่ทางโทเพื่อให้ยานพาหนะที่มารอสามารถผ่านไปได้
3. เมื่อมียานพาหนะมารอที่ทางโท แต่หยุดรอน้อยกว่า 5 วินาที (กรณีฝ่าฝืนกฎสัญญาณไฟจราจร) ระบบควบคุมจะให้สัญญาณไฟเขียวที่ทางเอกต่อไป จนกว่าจะมียานพาหนะมารอที่ทางโท
4. ถ้ายานพาหนะคันถัดไปมาถึงแยกทางโทก่อนเวลา 5 วินาทีหลังจากยานพาหนะคันก่อนหน้าได้ผ่านไปแล้ว จะยังคงให้สัญญาณไฟจราจรที่ทางโทต่อไปจนกว่าจะไม่มียานพาหนะที่ทางโทหรือเกินระยะเวลาของสัญญาณไฟเขียวในทางโท (ระยะเวลาสูงสุด 30 วินาที) แต่ถ้ายานพาหนะคันถัดไปมาถึงแยกทางโทหลังจากเวลาผ่านไปแล้ว 5 วินาที ระบบจะเปลี่ยนเป็นให้สัญญาณไฟเขียวที่ทางเอกทันที
5. หลังจากที่ระบบให้สัญญาณไฟแดงที่ทางโทและให้สัญญาณไฟเขียวที่ทางเอกแล้ว ทางโทจะต้องรอเป็นระยะเวลา 30 วินาที (สำหรับโหมด Fast) จึงจะได้สัญญาณไฟเขียวที่ทางโทอีกครั้งถ้าหากมียานพาหนะมารออยู่ แต่ถ้าไม่มียานพาหนะมารอ ทางเอกจะยังคงได้สัญญาณไฟเขียวต่อไป

บทที่ 5
สรุปผลการทดลอบและข้อเสนอแนะ

งานวิจัยนี้ประกอบด้วยส่วนที่สำคัญ 2 ส่วนคือส่วนตรวจจับ (Detector) ซึ่งเป็นประเภท Loop Detector และส่วนควบคุม โดยในงานวิจัยได้ทำการพัฒนาทั้ง 2 ส่วนนี้โดยอิงจากความสามารถของอุปกรณ์ที่มีขายในต่างประเทศ นอกจากความสามารถในการทำงานแล้ว ความปลอดภัยเนื่องจากสภาวะที่อุปกรณ์อาจจะทำงานผิดพลาดก็เป็นเรื่องสำคัญ ทั้งนี้เพราะระบบนี้จำเป็นต้องเกี่ยวข้องกับประชาชนเป็นจำนวนมาก ดั้งนั้นการทำงานที่ผิดพลาดของอุปกรณ์ (ทั้งจากที่ออกแบบผิดพลาดหรือผิดพลาดจากความไม่ปกติของอุปกรณ์) จำเป็นจะต้องได้รับการดูแลเพื่อไม่ให้ส่งผลต่อการใช้งานในที่สุด ซึ่งในการออกแบบเราก็ได้ป้องกันเรื่องนี้เป็นอย่างดี เช่น ต้องไม่มีเหตุการณ์ไฟเขียวทั้งสองด้านที่ตัดกัน

โดยปกติแล้วอุปกรณ์ควบคุมสัญญาณไฟจราจรจำเป็นต้องทำงานภายนอกห้องปรับอากาศ ดังนั้นอุณหภูมิและความชื้นก็เป็นสิ่งที่ต้องทดสอบ การแยกแรงดันของวงจรออกจากกันจึงเป็นเรื่องที่สำคัญที่ทำให้งานวิจัยนี้สามารถสำเร็จลุล่วงด้วยดี โดยเฉพาะส่วนที่เป็น Loop Detector ควรแยกจากวงจรทั้งหมดเพราะส่วนนี้อาจจะไปนำกระแสจากฟ้าฝ่าเข้ามาทำอันตรายแก่วงจรได้

ผลการทดสอบส่วน Detector นั้น วงจรสามารถทำงานได้เป็นอย่างดีที่ความไวตั้งแต่ 0.5% ขึ้นไป ทั้งนี้ก็เพียงพอที่จะใช้สำหรับตรวจสอบยานพาหนะประเภทรถจักรยานยนต์ได้เป็นอย่างดี ส่วน Detector นี้ประกอบด้วยวงจรที่เหมือนกัน 2 วงจร และสามารถปรับความถี่ที่แตกต่างกันได้ 3 ความถี่เพื่อป้องกันการรบกวนกันเองระหว่าง 2 วงจรนี้

จากผลการทดสอบที่ประเภทยานพาหนะเดียวกัน เมื่อตั้งไว้ที่ความถี่สูงจะให้การเปลี่ยนแปลงที่มากกว่า ทั้งนี้เนื่องจาก Q ของวงจรที่สูงขึ้นนั่นเอง (L เท่าเดิม แต่ C ลดค่าลง) ดังนั้นหากพิจารณาในแง่นี้ การตั้งความถี่ที่สูงกว่าดูเหมือนว่าจะได้เปรียบ แต่ถ้าหากสายนำสัญญาณจาก Loop Detector มาถึงวงจรมีค่ายาวมากๆ (มากกว่า 100 เมตร) การลดทอนเนื่องจากความถี่สูงจะทำให้วงจรไม่สามารถทำงานที่ความถี่สูงได้ ดังนั้นในทางปฏิบัติเราจะให้ Loop Detector ที่อยู่ใกล้ที่สุดมีความถี่สูงสุดที่จะตั้งได้ แล้วตั้งให้ Loop Detector ตัวอื่นมีความถี่ลดหลั่นกันลงไปตามลำดับ

ปัญหาและอุปสรรค

1. ในงานวิจัยนี้สามารถตรวจจับยานพาหนะขนาดเล็กได้จะต้องปรับความไวในการตรวจสอบเป็น 1 % จึงจะสามารถตรวจสอบได้
2. เมื่อมีความถี่ในการตรวจจับลดลงจะทำเกิดสัญญาณรบกวนมากขึ้น ซึ่งจะทำให้อัตราการเปลี่ยนแปลงเพิ่มมาขึ้นด้วย ดังนั้นการตรวจจับยานพาหนะจึงจะต้องมีค่ามากกว่าอันตราการเปลี่ยนแปลงที่เกิดขึ้น
3. การควบคุมสัญญาณไฟจราจรสามารถควบคุมแยกการจราจรได้เพียงแค่ 2 ทิศทางเท่านั้น
4. การติดตั้งตัวเหนียวนำความถี่จะต้องติดตั้งไว้ที่พื้นถนน ซึ่งจะมีความลำบากในการติดตั้ง

มหาวิทยาลัยเทคโนโลยีสุรนารี

บรรณนุกรม

[1] นคร ภักดีชาติ, อรรถพล บุญยะโภคาม, โอภาส ศิริครรชิตถาวร และชัยวัฒน์ ลิ้มพรวิไล (ม.ป.ป.). คู่มือทดลองไมโครคอนโทรลเลอร์ 32 บิตตระกูล ARM7 เบื้องต้น ฉบับ LPC2148. บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด.

[2] บริษัทอีทีที จำกัด. คู่มือการใช้งานบอร์ด CP-JR ARM7 USB-LPC2148 EXP [On line] จาก : http://www.ett.co.th/product/ARM/images/CP_JR_ARM7_LPC2138/MAN_CP_JR_ARM7_LPC2138.pdf

[3] บริษัทอีทีที จำกัด. คู่มือการใช้งานบอร์ด ET-MINI SD/MMC [On line] จาก :http://www.ett.co.th/product/InterfaceBoard/P-ET-A-00299/MAN_MINI_mp3_SD_AMP_LOGIC_PWR33.pdf

[4] รังสรรค์ ทองทา และ ประโยชน์ คำสวัสดิ์. 2545 ความไวของตัวตรวจจับแบบวงกลมที่เนื่องมาจากยานพาหนะขนาดเล็ก. รายงานฉบับสมบูรณ์ มหาวิทยาลัยเทคโนโลยีสุรนารี

[5] โครงการสาธิตการประยุกต์ใช้งานระบบควบคุมสัญญาณไฟจราจรอัจฉริยะเพื่อลดการใช้พลังงานในภาคขนส่ง. 2553. ศูนย์ความเป็นเลิศด้านชีวมวล มหาวิทยาลัยเทคโนโลยีสุรนารี

[6] โครงการวิจัยและพัฒนาระบบ Intelligent Transportation system. 2550. ศูนย์เทคโนโลยีอิเล็กทรอนิกส์ และคอมพิวเตอร์แห่งชาติ

[7] การพัฒนาเครื่องควบคุมสัญญาณไฟจราจรแบบตรวจจับยานพาหนะบนท้องถนนเพื่อตัดสินใจในการกำหนดสัญญาณไฟจราจรที่เหมาะสม (The development of traffic signal controller using the vehicle detector to set up proper traffic signals.) ชญานนท์ มินเสน ,ผศ.ดร. จิตรเกษม งามนิล และวฐา มินเสน มหาวิทยาลัยเทคโนโลยีมหานคร และมหาลัยวิทยาเชียงใหม่

[8] เครื่องตรวจจับและนับรถยนต์ (Car detector and counter) กิตติพงศ์ ศิลปวงศ์เจริญ, อภิศักดิ์ วงศ์ศรีรักษ์, สถาพร อุดมสิน คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น

[9] Traffic Engineering Handbook , Fifth Edition , James L.Pline ,Institute of Transportation Engineerings 1099 14th Street, NW Suite 300 West Washington, DC 20005-3438 USA.

[10] Intelligent Transportation System. Smart and Green Infrastructure Design. Second Edition , Sumit Ghosh , Tony S.Lee

[11] Eric Norum. 2002. Vehicle Detector, Lab (Online). Available URL: http://www.engr.usask.ca/classes/EP/424/lab/EP414LabManual/node5.html.

[12] Paul Consentino and Barry Grossman. 1996. Development of Fiber Optic Dynamic Weight-in-Motion Systems. Report Submitted to Transportation Statistic Office, Florida Department of Transportation, and Tallahassee, Florida.

[13] U.S. Department of Transportation. 1990. Traffic Detector Handbook. 2nd Edition. McLean,
Virginia

ภาคผนวก

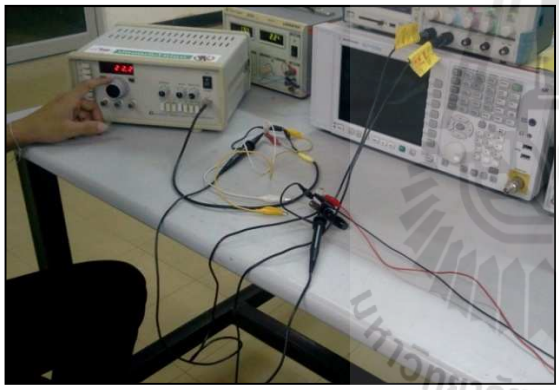
มหาวิทยาลัยเทคโนโลยีสุรนารี

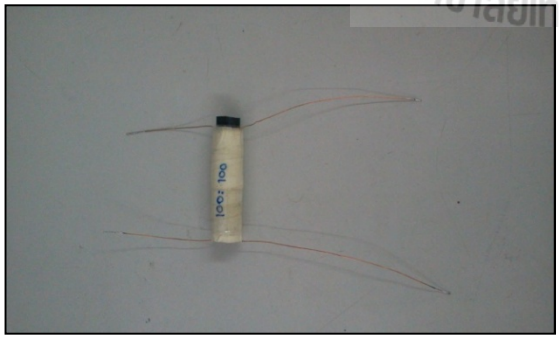
การจำลองลูปเหนี่ยวนำ (Loop Detector) ก่อนที่จะติดตั้งบนท้องถนน

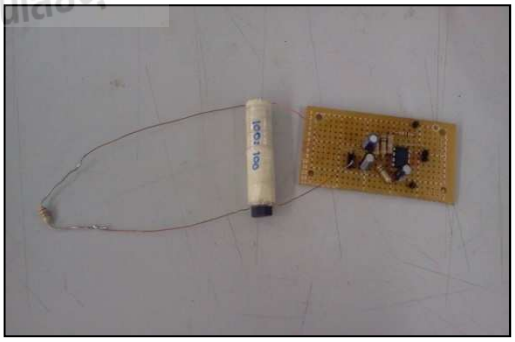
การทดสอบด้วยยานพาหนะขนาดเล็กก่อนที่จะติดตั้งบนท้องถนน

การทดสอบความถี่เรโซแนนซ์ของหม้อแปลงที่เชื่อมต่อกับลูปเหนี่ยวนำ

มหาวิทยาลัยเทคโนโลยีสุรนารี

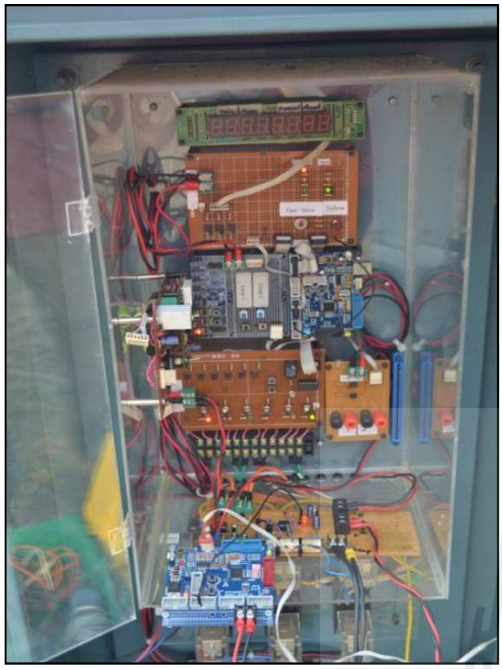
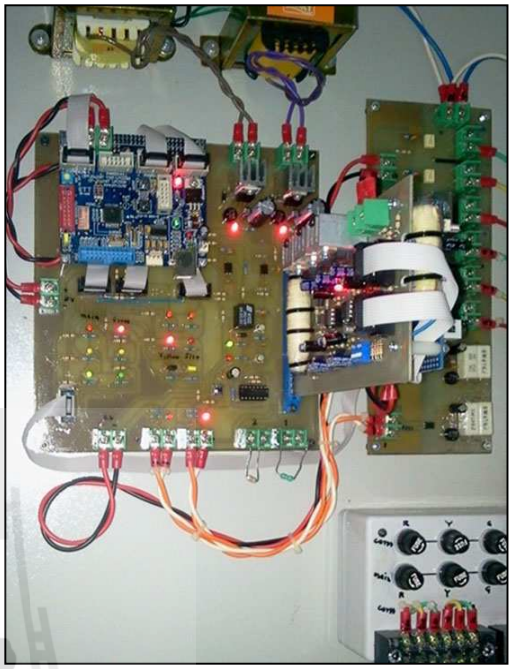




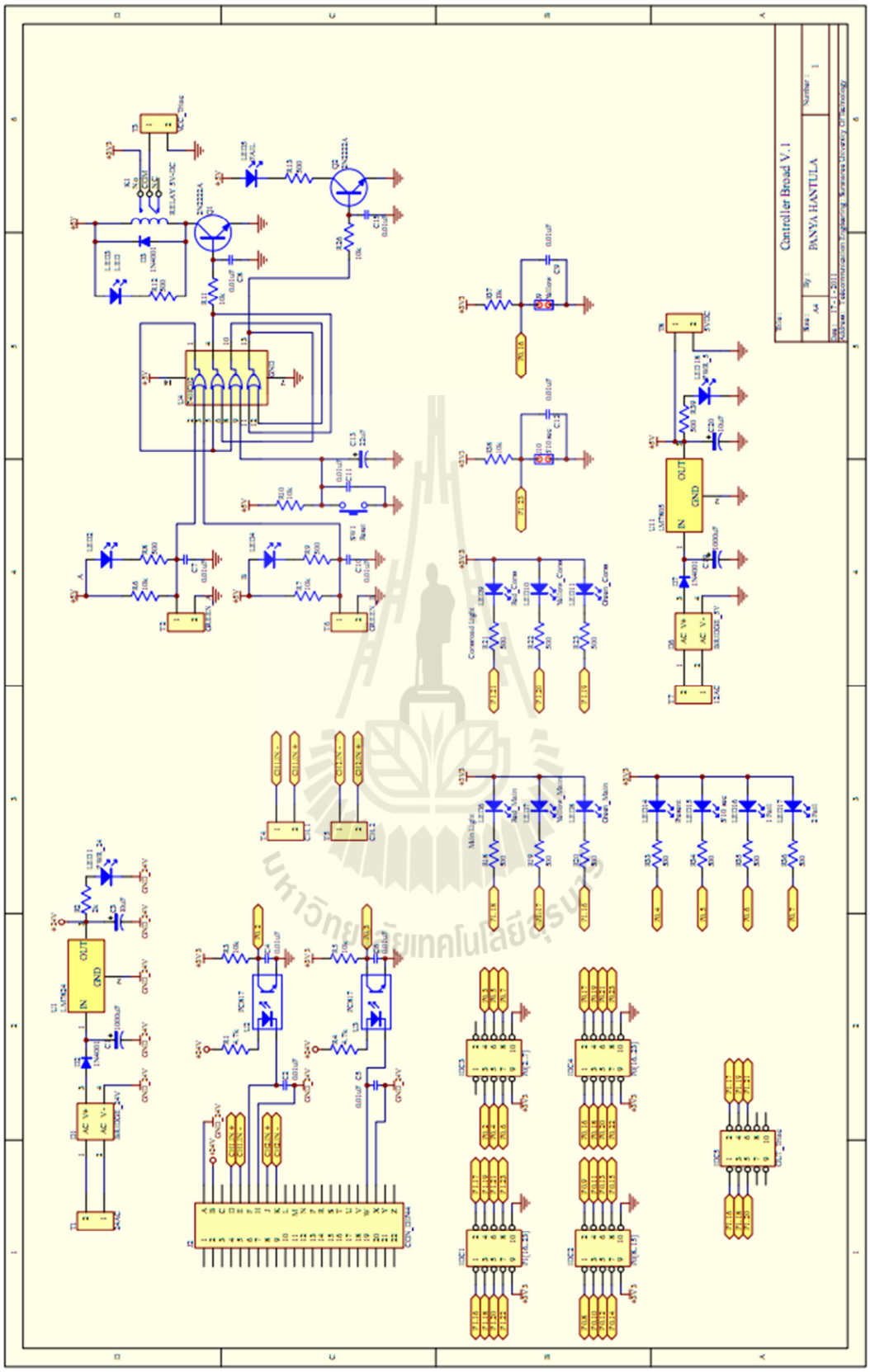


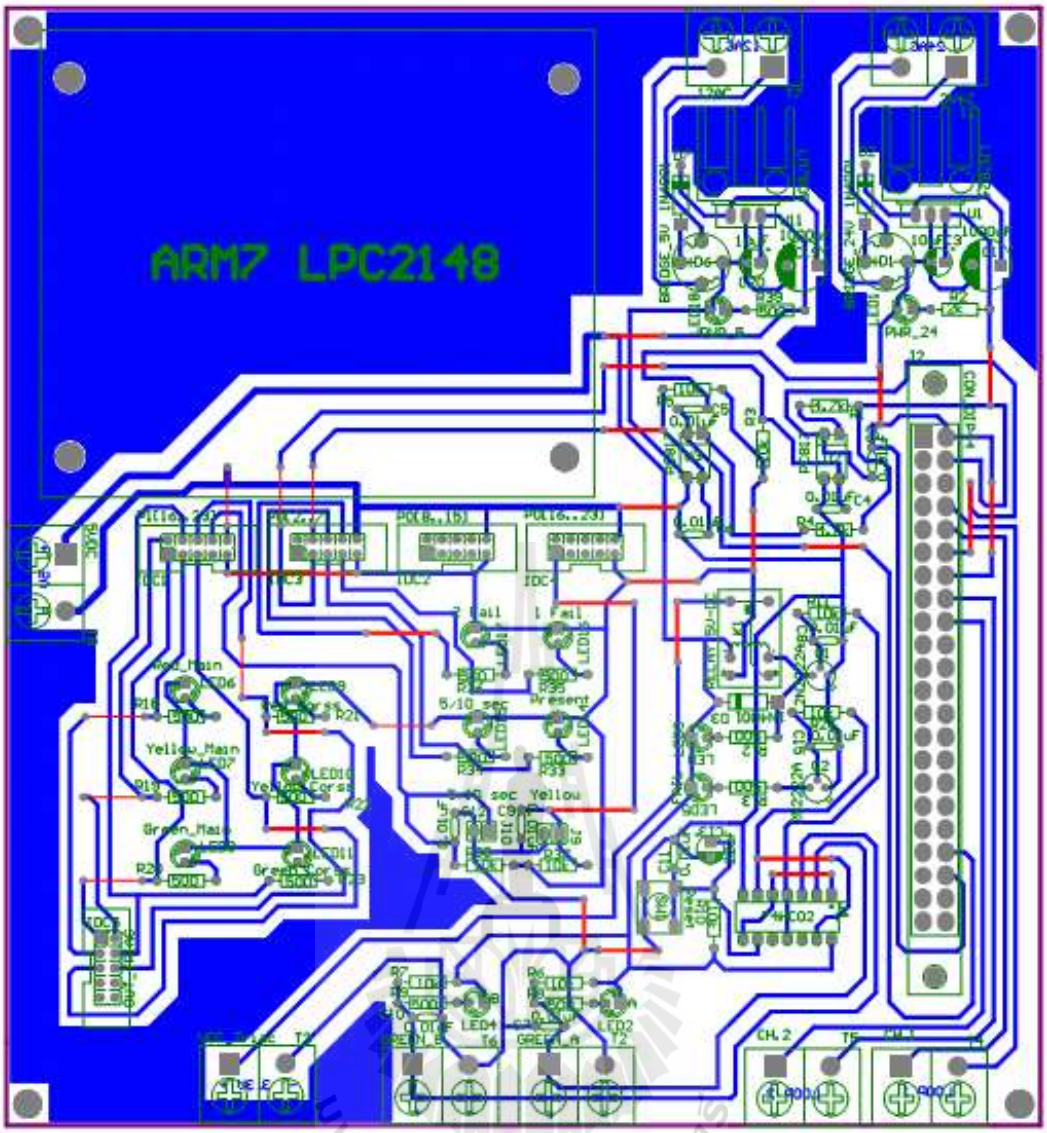
มหาวิทยาลัยเทคโนโลยีสุรนารี

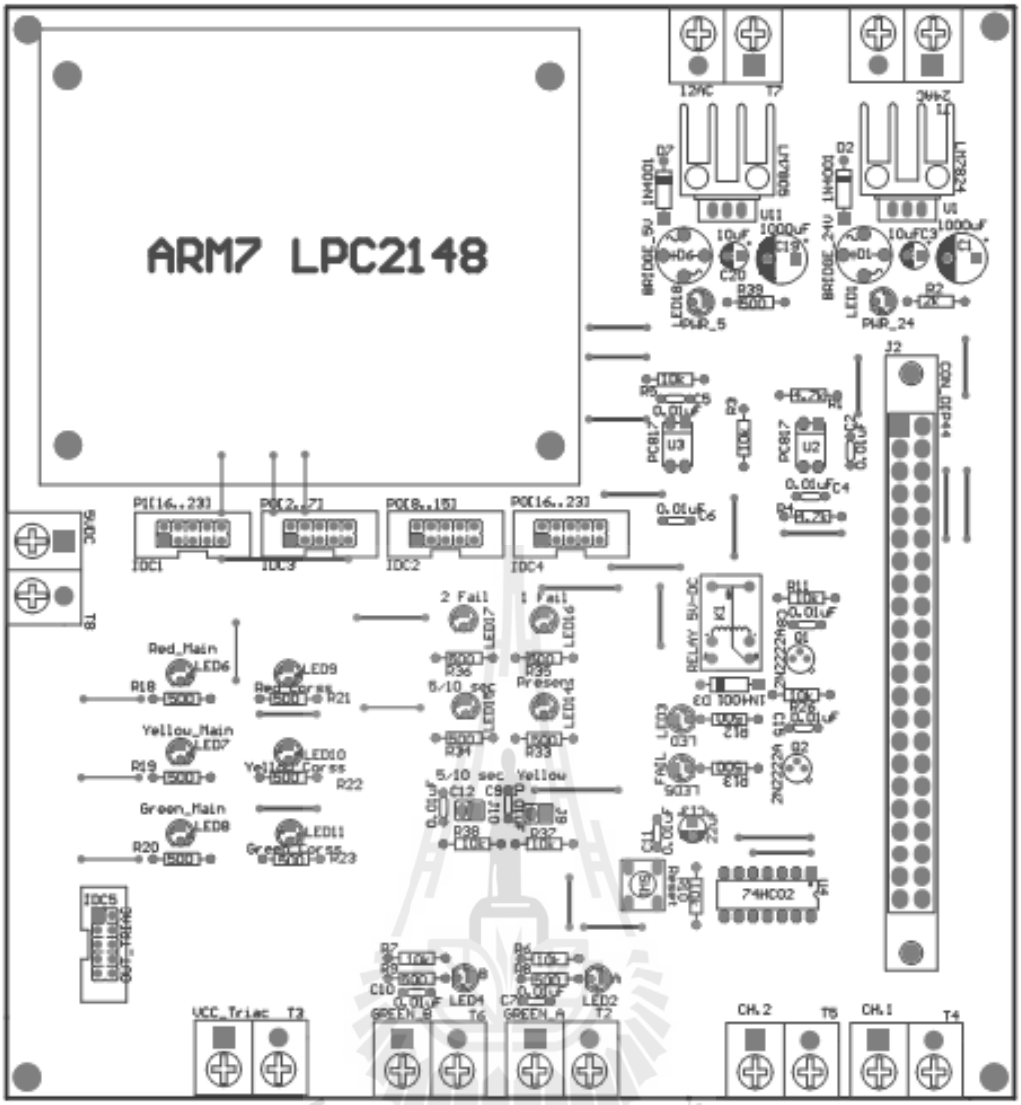
ทดสอบตัวตรวจจับยานพาหนะกับลูปเหนียวนำที่ติดตั้งอยู่บนท้องถนน

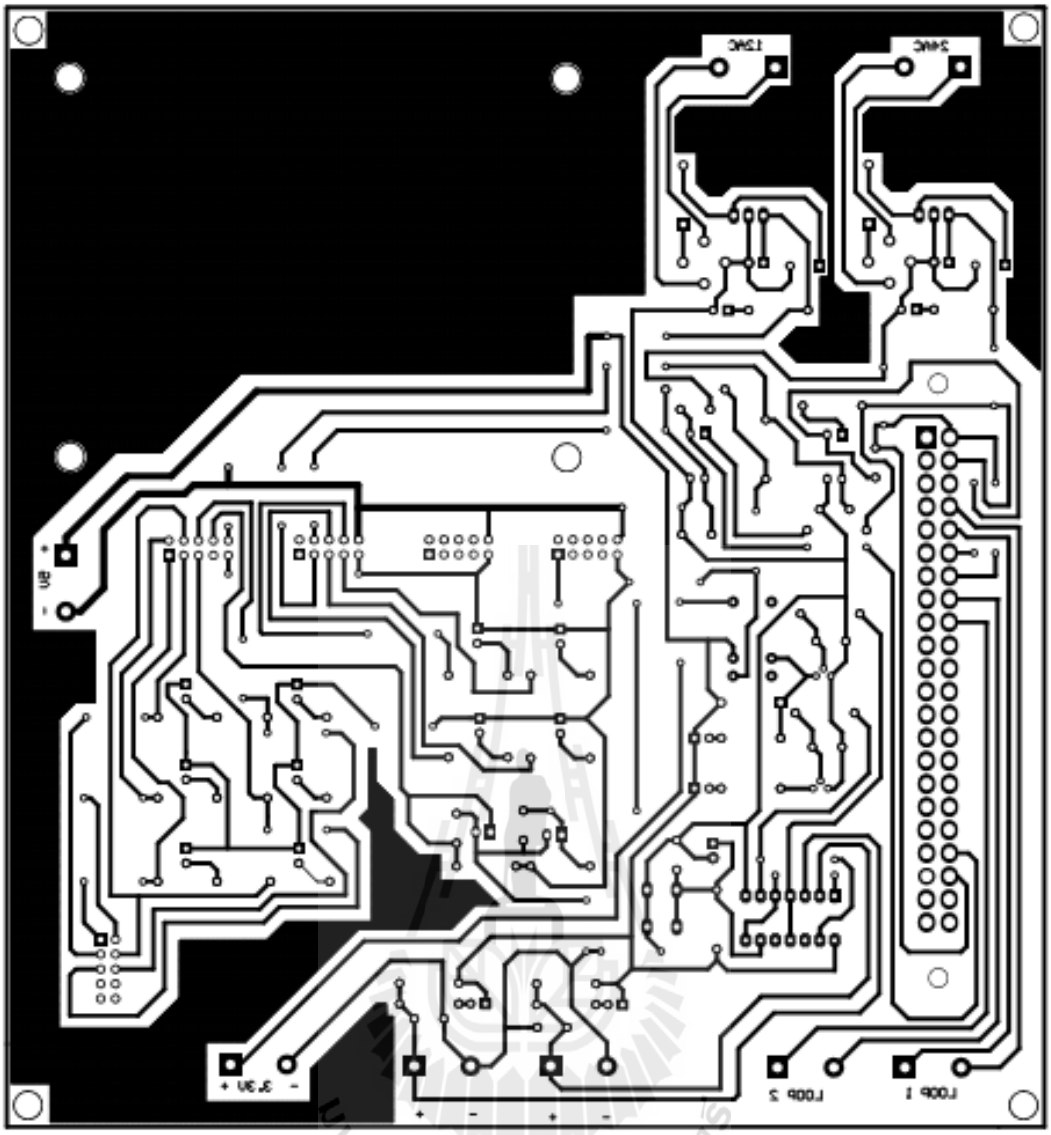



DATASHEET ระบบควบคุมสัญญาณไฟจราจร (Controller)



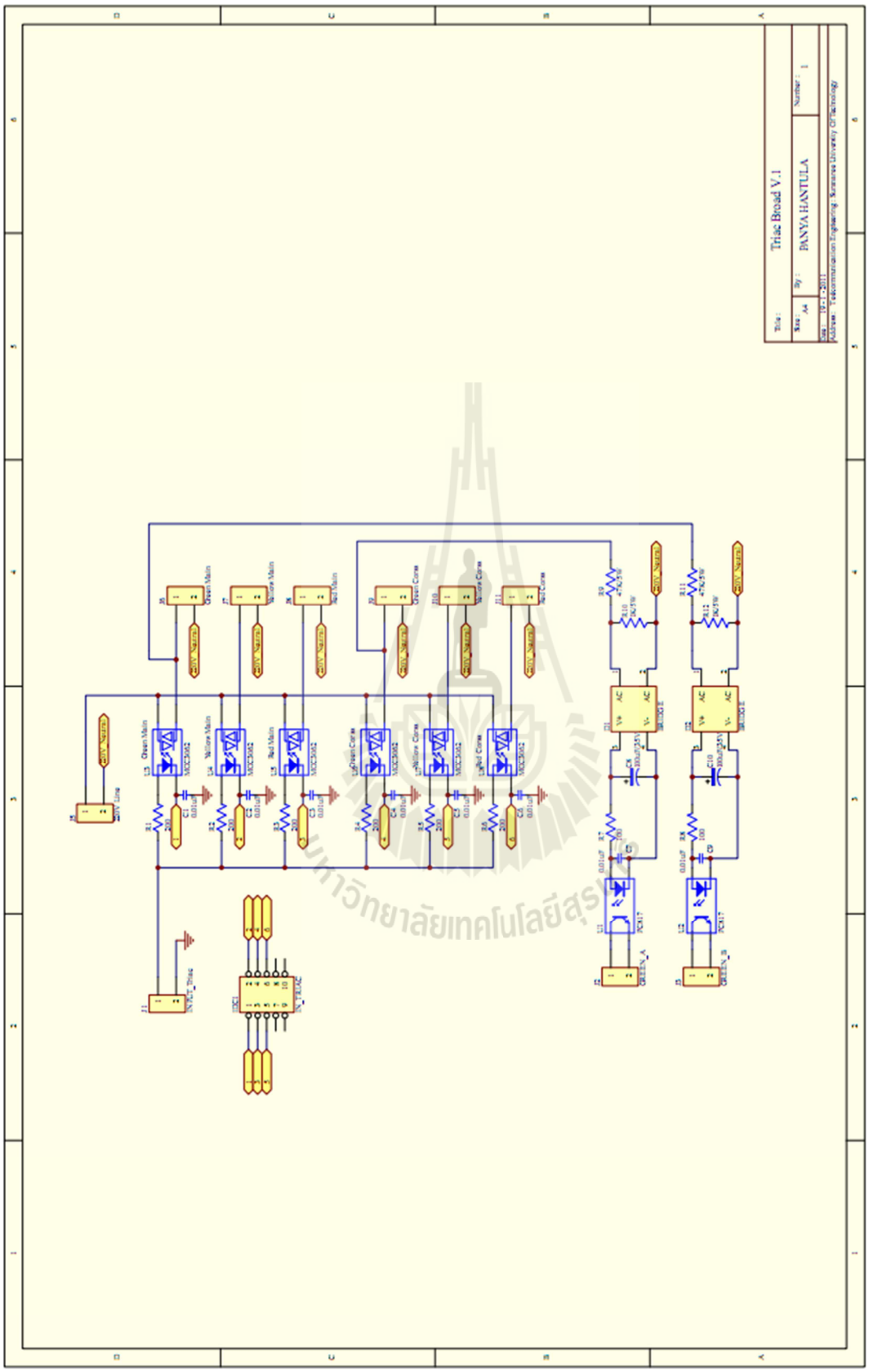

ARM7 LPC2148

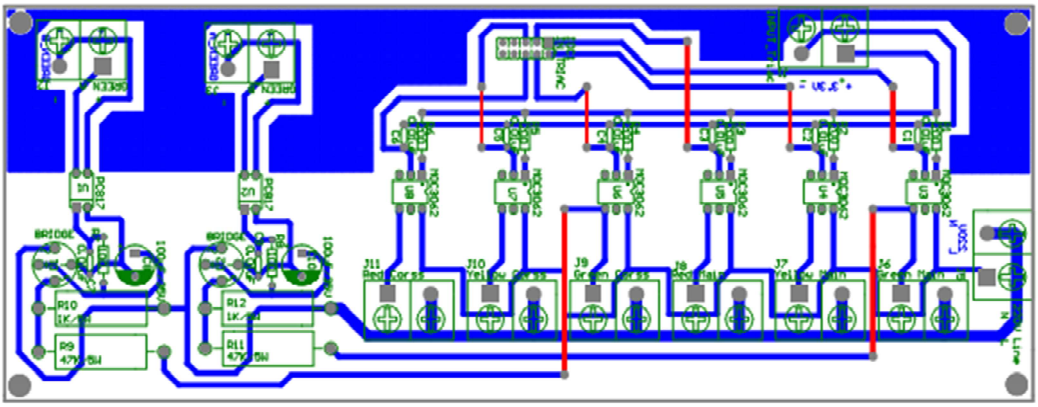

ARM7 LPC2148
12VAC
24VAC
T7
T1
LM7805
LM7824
1N4001
D1
D2
BRIDGE_5V
BRIDGE_24V
10uF
1000uF
C20
C19
C3
C1
R38
R2
PWR_5
PWR_24
J2
CON_DIP44
PC817
R5
R3
R4
R8
C5
C4
0.01uF
0.1uF
30VAC
T8
P1[16..23]
P0[2..7]
P0[8..15]
P0[16..23]
IDC1
IDC3
IDC2
IDC4
2 Fail
1 Fail
LED17
LED16
R36
R35
Red_Main
LED6
LED9
R18
R21
5/10 sec
Present
LED15
LED14
R34
R33
Yellow_Main
LED7
LED10
R19
R22
5/10 sec Yellow
C12
C9
R38
R32
Green_Main
LED8
LED11
R20
R23
RELAY 5V-DC
K1
1N4001 D3
2N2222A
Q3
Q2
R11
R10
R9
C18
C13
R13
R12
FAIL
LED5
LED3
C11
C10
R7
R6
C10
C7
0.01uF
LED4
LED2
GREEN_B
GREEN_A
T6
T2
74HC02
U4
IDC5
OUT_TRIAC
VCC_Triac
T3
CH.2
T5
CH.1
T4

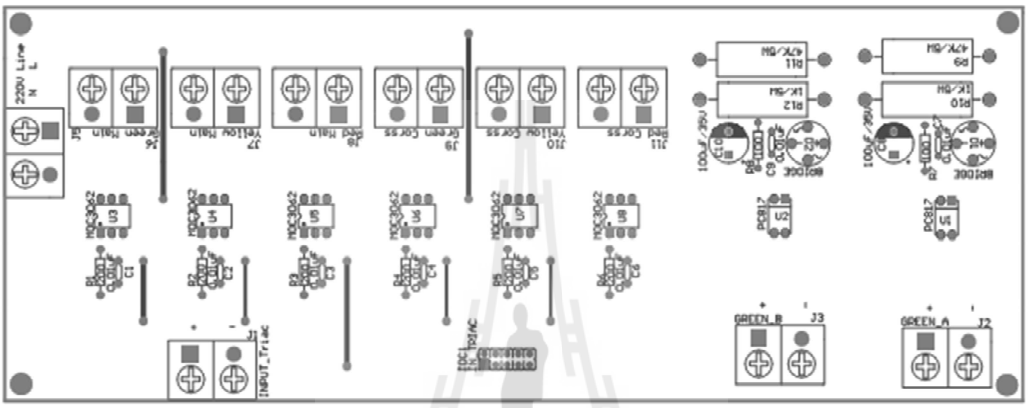


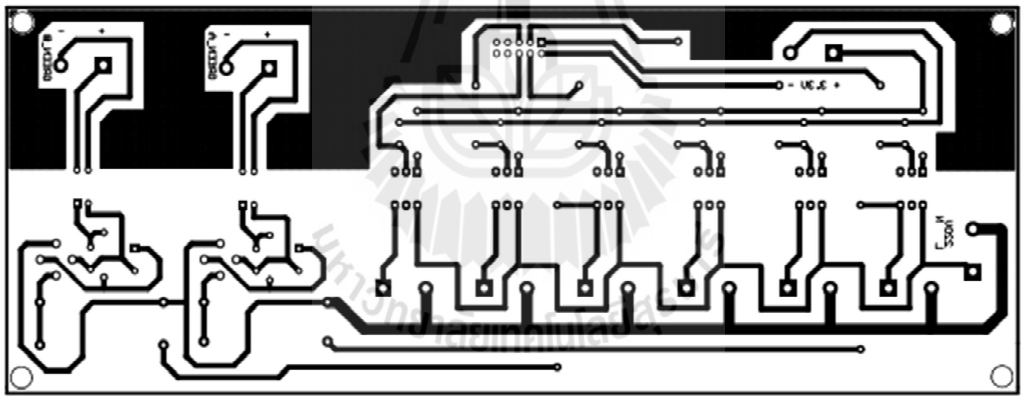
มหาวิทยาลัยเทคโนโลยีสุรนารี

DATASHEET วงจรควบคุมการเปิดและปิดสัญญาณไฟจราจร

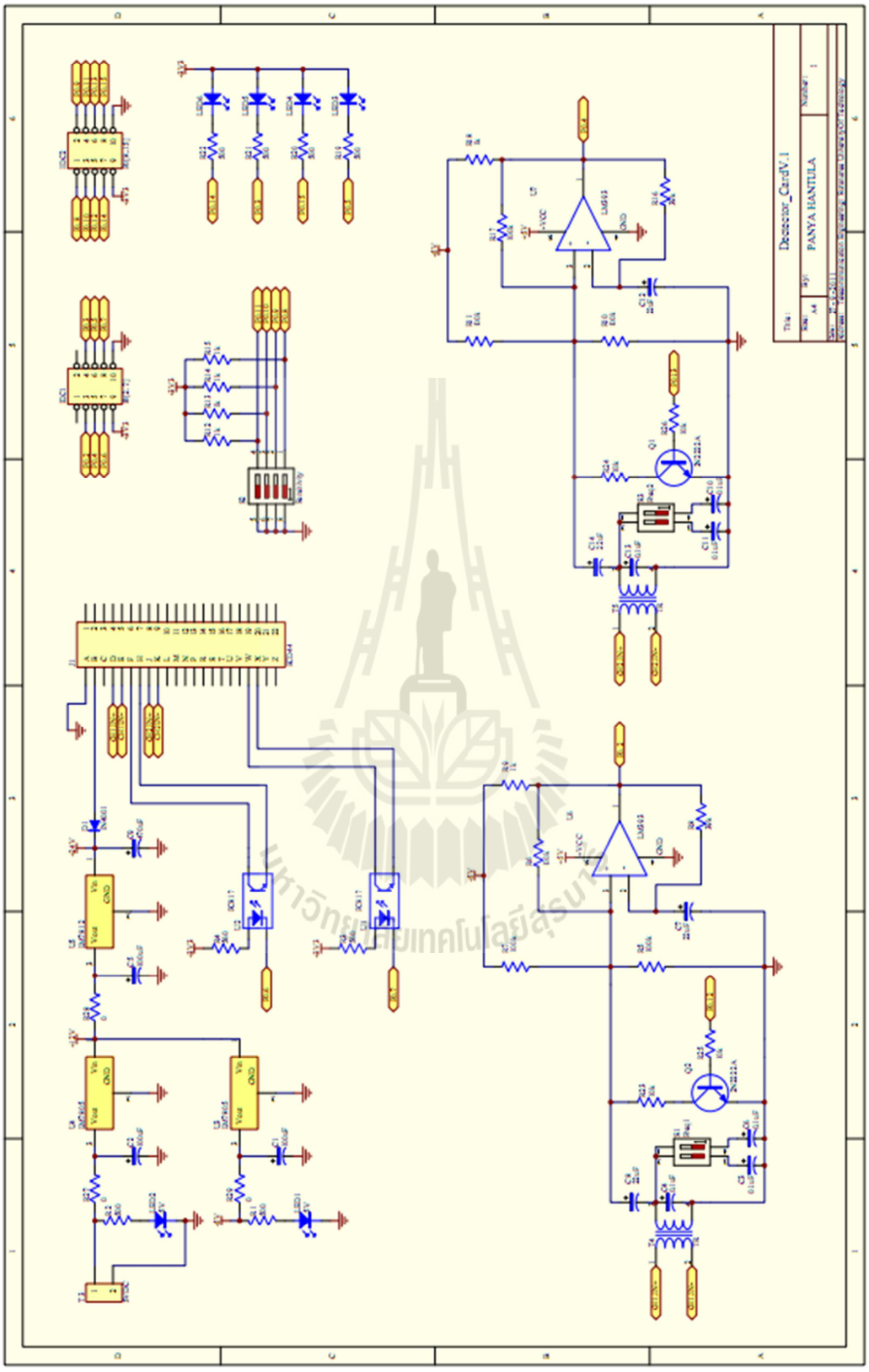


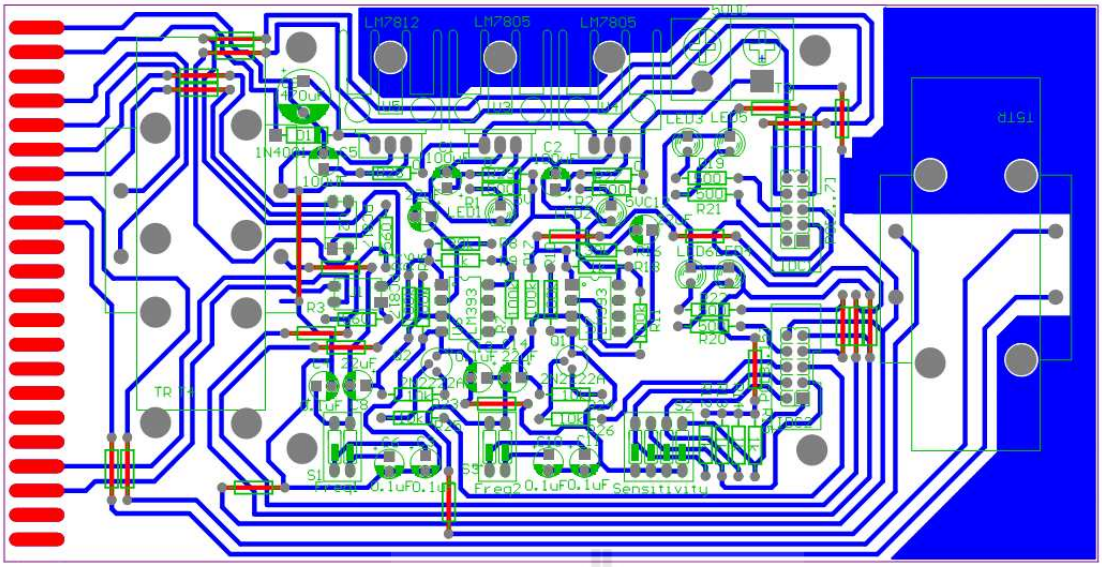


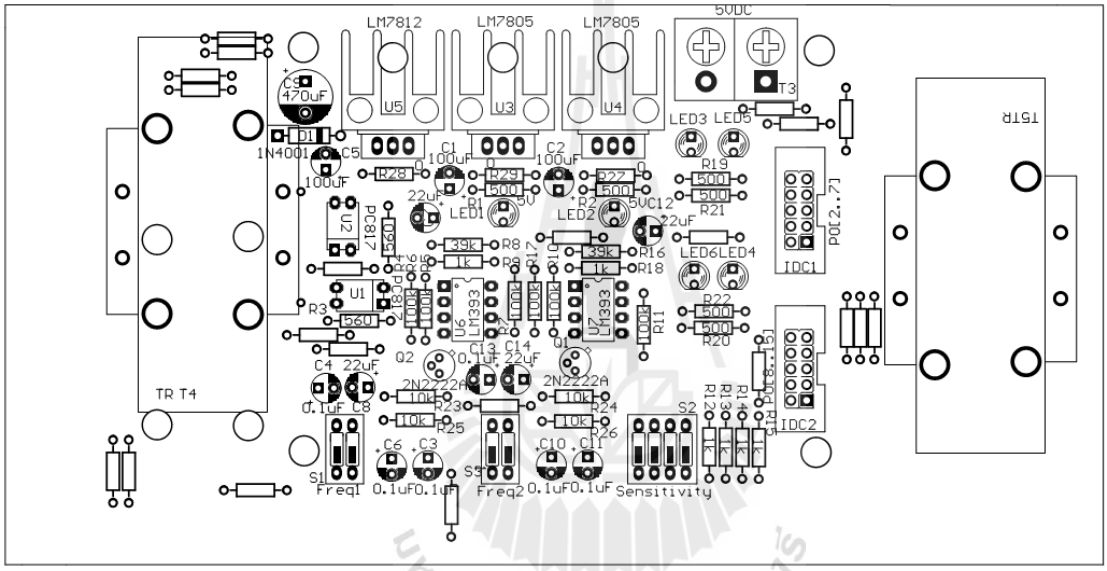


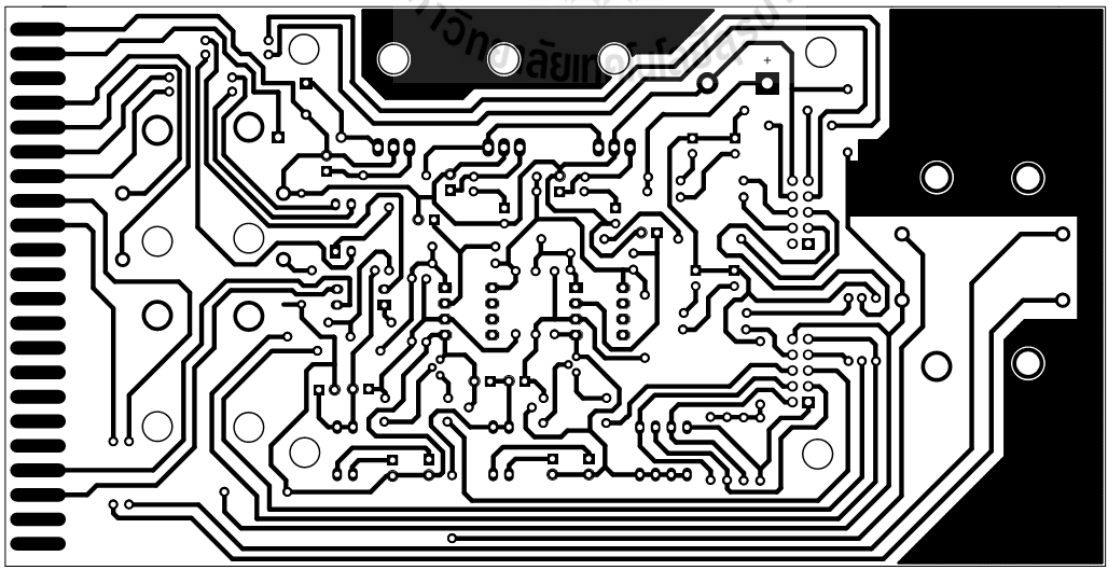


DATASHEET ตัวตรวจจับยานพาหนะ (Detector)

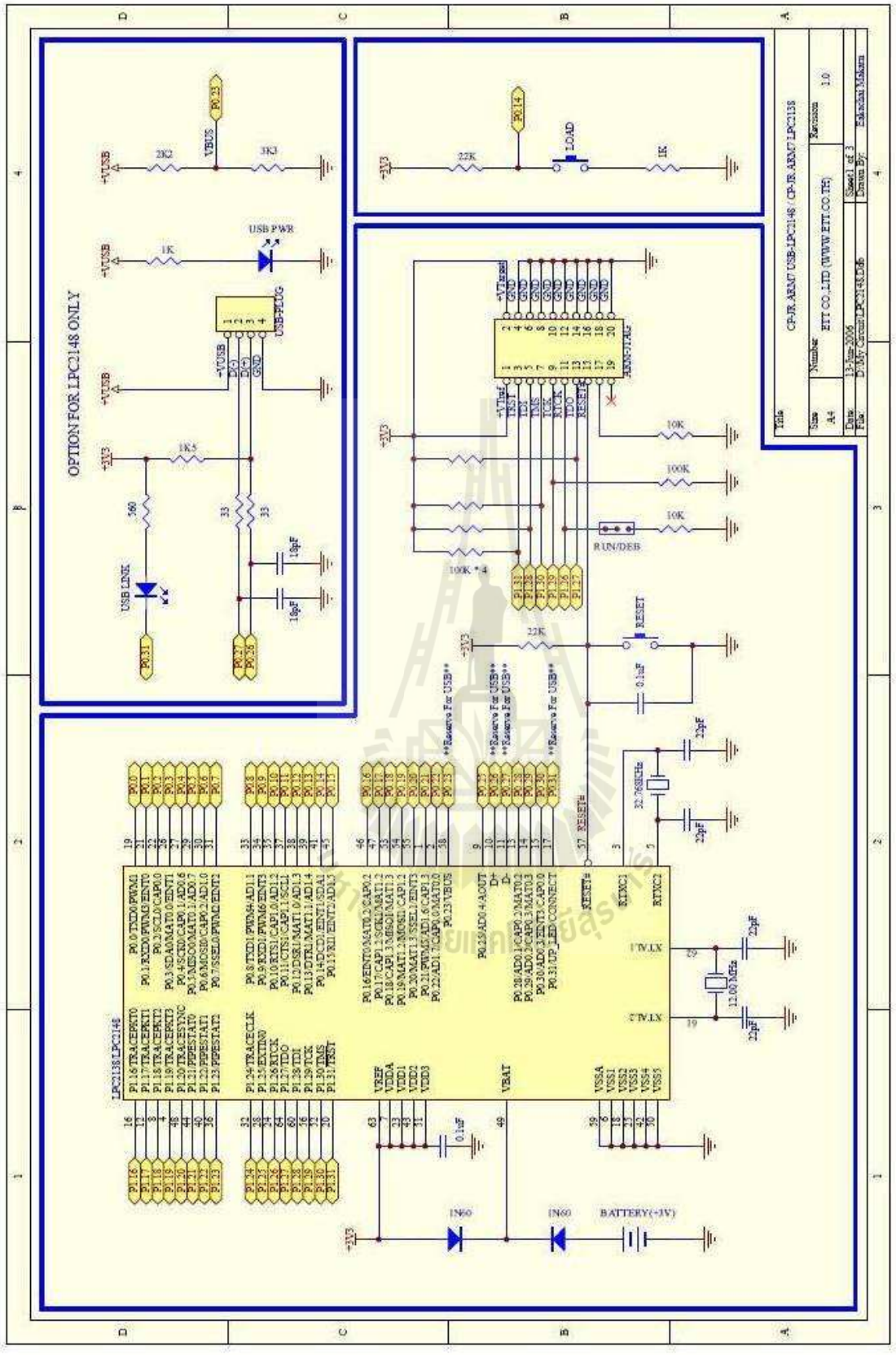


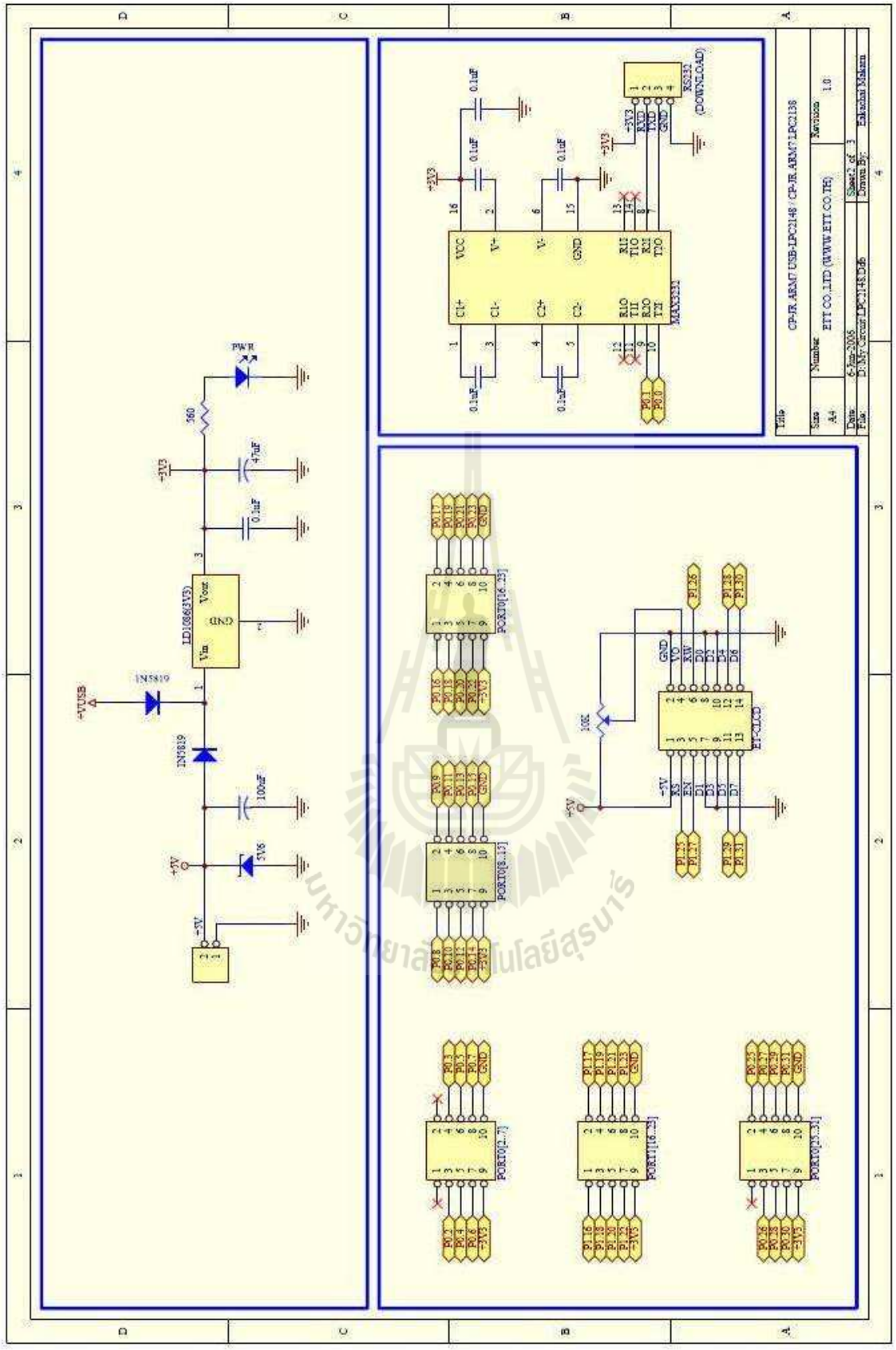


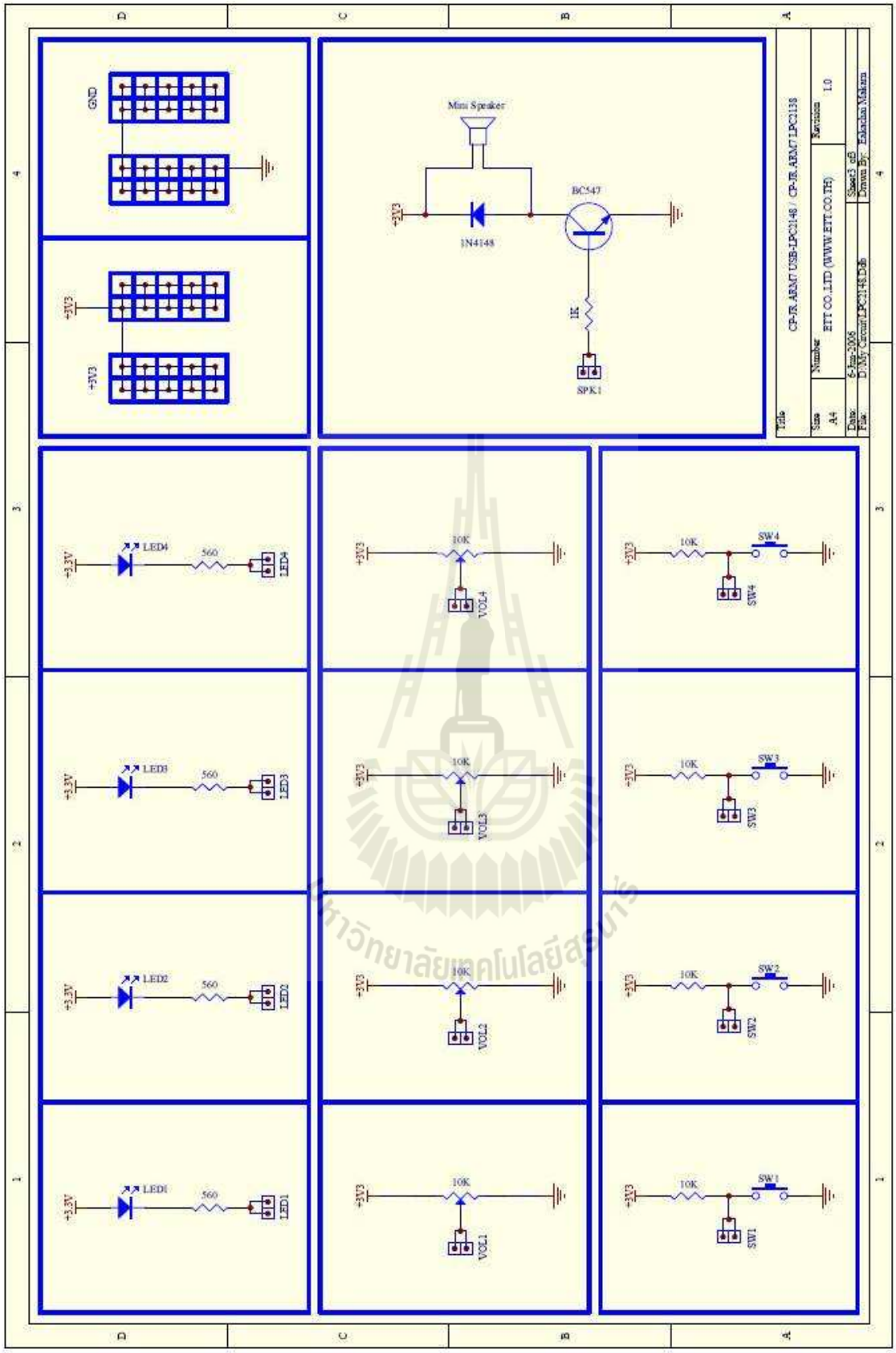

LM7812
LM7805
LM7805
50VDC
C5
470uF
D1
1N4001
C1
100uF
C2
100uF
T3
LED3 LED5
R19
500
R21
500
15TR
POT2..7
IDC1
LED1
LED2
LED6 LED4
R22
500
R20
500
IDC2
TR T4
C4 22uF
0.1uF C8
2N2222A
2N2222A
10k
10k
R25
10k
10k
C6 .C3
.C10 .C11
S1
S2
S3
Freq1 0.1uF 0.1uF
Freq2 0.1uF 0.1uF
Sensitivity



DATASHEET MICROCONTROLLER (CP_JR_ARM7_USB_LPC2148)



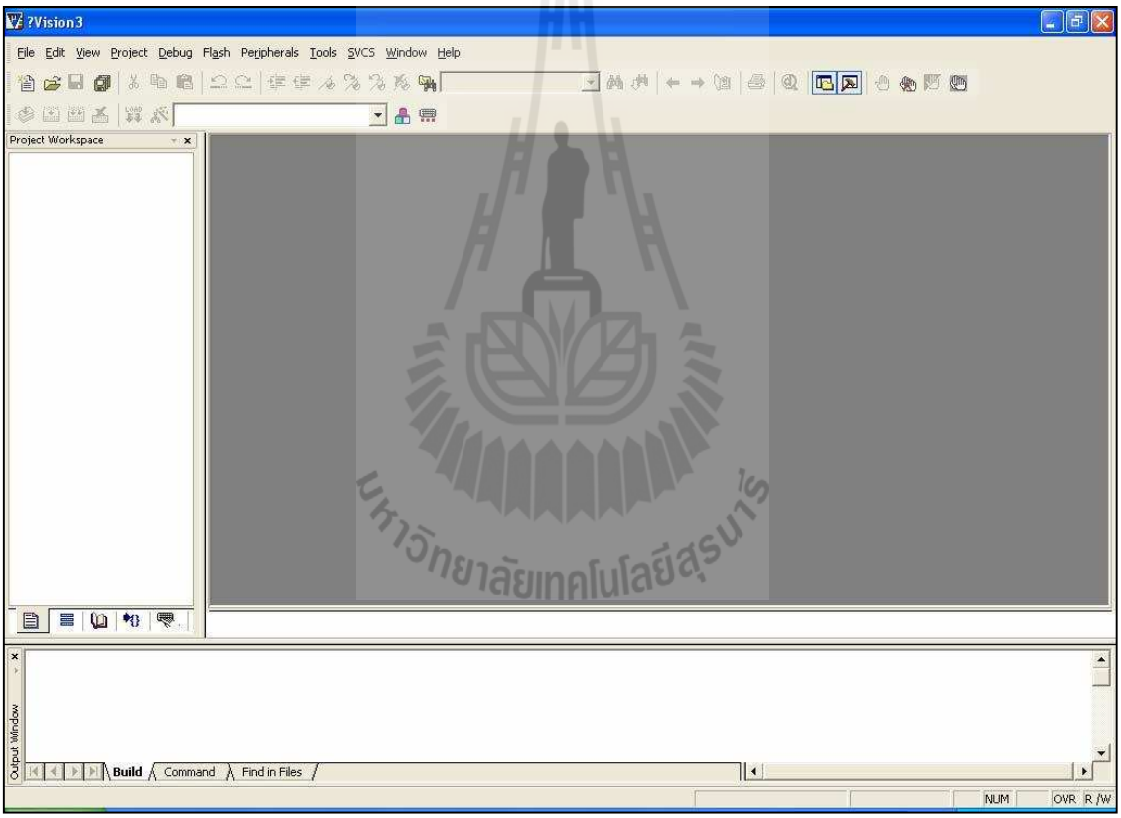

CP-JR.ARM7 USB-LPC2148 / CP-JR.ARM7-LPC2138
ETT CO.,LTD (WWW.ETT.CO.TH)
Number
Revision
1.0
Sheet 2 of 3
Drawn By: Ekachai Makarn
Date: 6-Jun-2006
File: D:\My Circuit\LPC2148\Dc6
Size: A4
Title
MAX3232
LD1086(3V3)
LM7805
LM7812
LM7912
LCD
PWR
IN5819
POWER
RS232
(DOWNLOAD)
PORT0[16..23]
PORT0[8..15]
PORT0[2..7]
PORT1[16..23]
PORT0[25..31]


GND
+3V3
Mini Speaker
BC547
1N4148
1K
SPK1
LED4
LED3
LED2
LED1
560
10K
VOL4
VOL3
VOL2
VOL1
SW4
SW3
SW2
SW1
CP-IX ARM7 USB-LPC2148 / CP-IX ARM7 LPC2138
ETT CO.,LTD (WWW.ETT.CO.TH)
Revision 1.0
Size A4
Number
Date: 6-Jan-2008
File: D:\My Circuit\LPC2148\D4b
Sheet3 of5
Drawn By: Eakachai Maksra

การใช้ Keil uVision3 ในการสร้าง Project File ของ Keil ARM

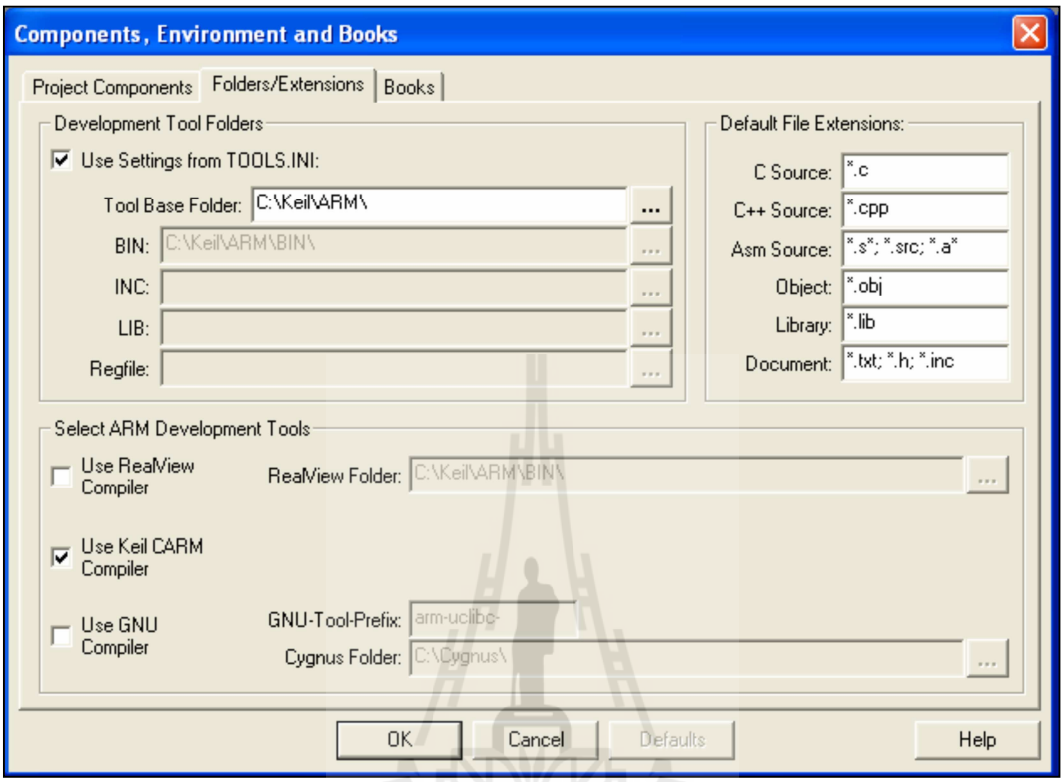
ในที่นี้จะขอแสดงแนวทางการเขียนโปรแกรมภาษาซีโดยใช้ Keil-CARMในการแปลคำสั่งภายใต้โปรแกรม Text Editor ของ Keil (Keil uVision3) โดยจะขออธิบายถึงเฉพาะวิธีการกำหนดค่า Option สำหรับเชื่อมโยงคำสั่งในการสั่งแปลโปรแกรมด้วย Keil-CARM ผ่านทาง Keil uVision3 เท่านั้น ส่วนรายละเอียดคำสั่งและการใช้งานฟังก์ชันต่างๆในการเขียนโปรแกรมของ Keil-CARM นั้นขอให้ผู้ใช้ศึกษาจากคู่มือคำสั่งของ Keil-CARM เอง โดยวิธีการกำหนดค่าตัวเลือกของ Keil uVision3 ให้ใช้งานกับ Keil-CARM นั้นมีขั้นตอนพอสังเขปดังนี้คือ

1. เปิดโปรแกรม Keil uVision3 ซึ่งเป็นโปรแกรม Text Editor ของ Keil-CARM ใช้สำหรับใช้ในการเขียนโปรแกรมที่เป็น Source Code ภาษาซี โดยจะมีลักษณะดังรูป

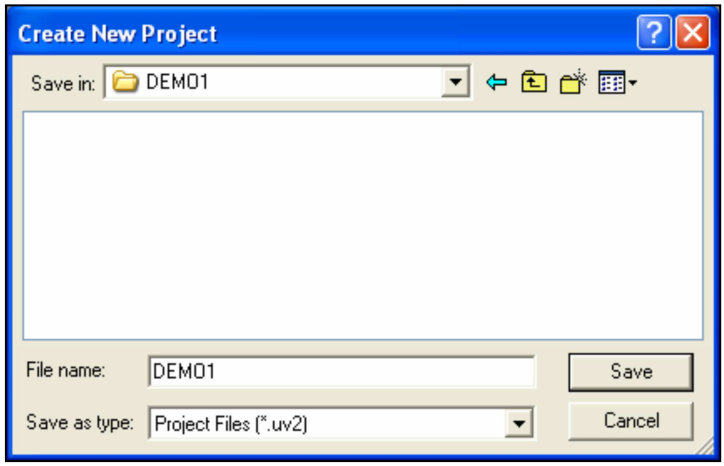


2. ทำการกำหนดค่าตัวเลือกในการแปลคำสั่งของ uVision3 ให้ใช้งานกับโปรแกรม Keil uVision3 และ Keil-CARM โดยให้เลือกคลิกเมาส์ที่เมนูคำสั่ง Project ➔ Components ,Environment, Books... จากนั้นให้เลือกค่าตัวเลือกสำหรับกำหนดการใช้งาน Complier จากหัวข้อ Select ARM Development Tools ซึ่งจะมีค่าตัวเลือกอยู่ 3 แบบ คือ Use Keil-CARM Tools ,Use GNU Tools และ Use ARM Tools โดยให้เลือกเป็น “Use Keil ARM Tools” จากนั้นให้ทำการกำหนดตำแหน่ง Folder

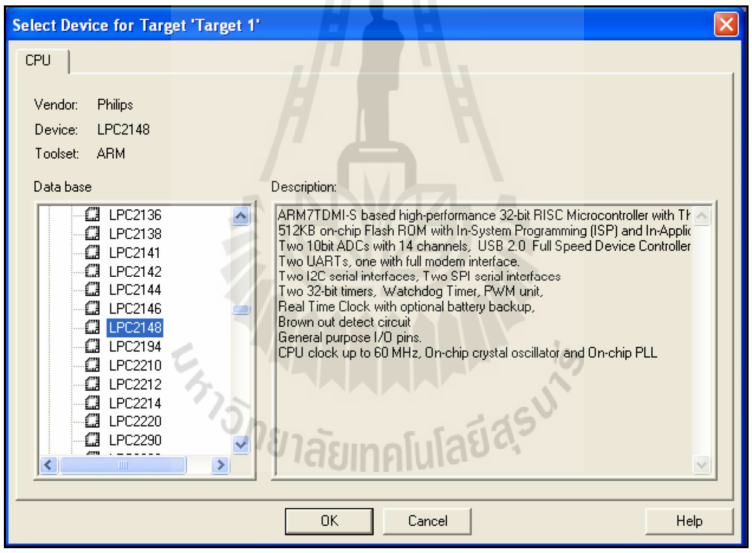
สำหรับเก็บค่าตัวเลือกการทำงานของโปรแกรม Keil ARM ซึ่งตามปรกติ แล้วจะเป็น “C:\Keil\ARM\” แต่ถ้าติดตั้ง Keil ไว้ที่อื่นก็ต้องปรับเปลี่ยนให้ถูกต้องตามความเป็นจริงด้วยดังรูป



3. ทำการสร้าง Project File ขึ้นมาใหม่ โดยเรียกเมนูคำสั่ง Project ➔ New Project จากนั้นให้เลือกกำหนดหรือสร้างตำแหน่ง Folder ที่จะบันทึก Project File พร้อมกับกำหนดชื่อ Project File ตามต้องการ เช่น ถ้าต้องการสร้าง Project File ชื่อ DEMO1 โดยเก็บไว้ใน Folder ชื่อ DEMO1 ก็สามารถกำหนดตำแหน่ง Folder และชื่อ Project Fileได้เอง โดยเมื่อกำหนดชื่อในช่อง File name แล้วให้เลือก Save เพื่อบันทึก Project File ไว้ ดังรูป



หลังจากกำหนดชื่อและสั่ง Save Project File แล้ว โปรแกรมจะรอให้ผู้ใช้ทำการกำหนดเบอร์ MCU ที่จะใช้งานใน Project ที่สั่ง Save นั้น ซึ่งในกรณีที่ใช้งานกับบอร์ด CP-JR ARM7 USB-LPC2148 นั้น ให้เลือกกำหนดเบอร์ของ MCU เป็น LPC2148 ของ Philips แล้วเลือก OK ดังรูป



หลังจากเลือกกำหนดเบอร์ของ MCU เป็นที่เรียบร้อยแล้ว ในขั้นตอนนี้โปรแกรมจะรอให้ผู้ใช้ยืนยันว่าต้องการจะทำการ Copy ไฟล์ Startup ของ Keil เพื่อใช้งานกับ MCU ของ Philips มาใช้ใน Project ด้วยหรือไม่ โดยStartup ไฟล์จะเป็นส่วนของการกำหนดค่าเริ่มต้นการทำงานให้กับ MCU เช่น การกำหนดค่า Stack และการกำหนดค่าการทำงานให้กับ Phase-Lock-Loop ต่างๆ ก่อนที่จะเริ่มต้นทำงานตามโปรแกรมที่เราเขียนขึ้น ไม่เช่นนั้นแล้วโปรแกรมที่เราเขียนขึ้นมานั้นจะต้องเพิ่มคำสั่งในการเตรียมการทำงานส่วนเหล่านี้ให้ MCU เองทั้งหมดแต่เนื่องจากไฟล์ Startup ของ Keil-ARM นั้น เป็นไฟล์ภาษา แอสแซมบลี้(Assembly) ซึ่งกำหนดค่าการทำงานไว้กับชุดพัฒนาของ Keil เอง ดังนั้นข้อกำหนดและการกำหนดค่าบางอย่างจะมีความแตกต่างกันอยู่กับค่าที่ต้องการสำหรับบอร์ด “CP-JR ARM7 USB-LPC2148” ไม่สามารถใช้งานไฟล์ Startup ได้ทันที ต้องมีการแก้ไขค่าตัวเลือกใหม่ดังนั้นก่อนที่จะใช้โปรแกรม Keil-CARM ในการแปลคำสั่งให้นั้น ผู้ใช้

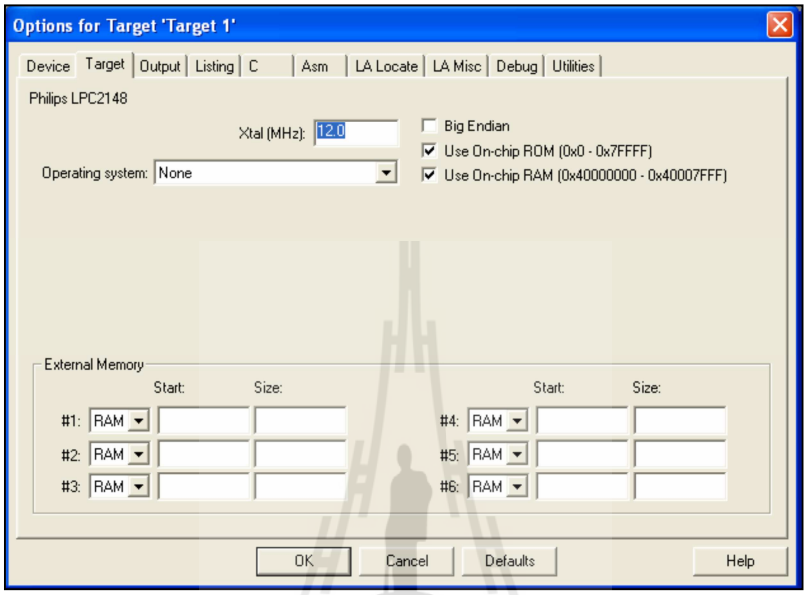
จะต้องเข้าไปแก้ไขไฟล์ Startup ใหม่โดยต้องกำหนดรูปแบบให้ถูกต้องตรงกับความต้องการของบอร์ดด้วย ดังนั้นในที่นี้ขอแนะนำให้เลือก “No” เพื่อไม่ให้ Keil uVision3 ทำการ Copy ไฟล์ Startup ของ Keil-CARM มาใช้ใน Project นี้ด้วย

4. ให้ทำการ Copy File ชื่อ “Startup.s” ที่ทาง ผู้จัดทำจัดเตรียมไว้ใน CD-ROM ชื่อ “Startup.s” มาไว้ในตำแหน่ง Folder เดียวกันกับ Project File ที่สร้างขึ้นมาใหม่นี้โดยไฟล์ “Startup.s” จะเป็นไฟล์ซึ่งบรรจุคำสั่งภาษาแอสแซมบลี้ของ ARM7 สำหรับทำหน้าที่กำหนดค่าเริ่มต้นการทำงานที่จำเป็นให้กับ MCU ซึ่งได้แก่การ กำหนดค่า Stack ให้กับ MCU การ Initial Phase-Lock-Loopการกำหนดค่าให้กับ MAM Function และการกำหนดตำแหน่ง Vector ต่างๆของ MCU สำหรับใช้งานร่วมกับบอร์ด “CP-JR ARM7 USB-LPC2148” ซึ่งถ้าสั่ง Add ไฟล์ “Startup.s” จาก Keil หรือ Copy ไฟล์ดังกล่าวมาจากแหล่งอื่นๆ อาจมีการทำงานของโปรแกรมใน Startup ไม่เหมือนกัน ซึ่งจะส่งผลต่อการทำงานของโปรแกรมด้วย

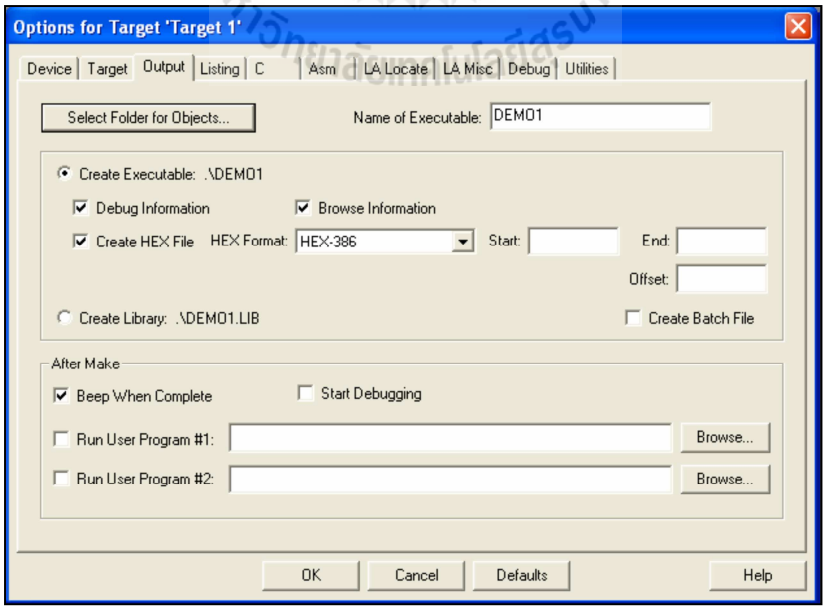
มหาวิทยาลัยเทคโนโลยีสุรนารี

5. ให้ทำการกำหนดค่า Option ของ Project File โดยเลือกเมนูคำสั่ง Project ➔ Option for Target 'Target 1' จากนั้นเลือกที่ Tab ของ Target เพื่อกำหนดค่าของ MCU Target โดยให้กำหนดดังนี้

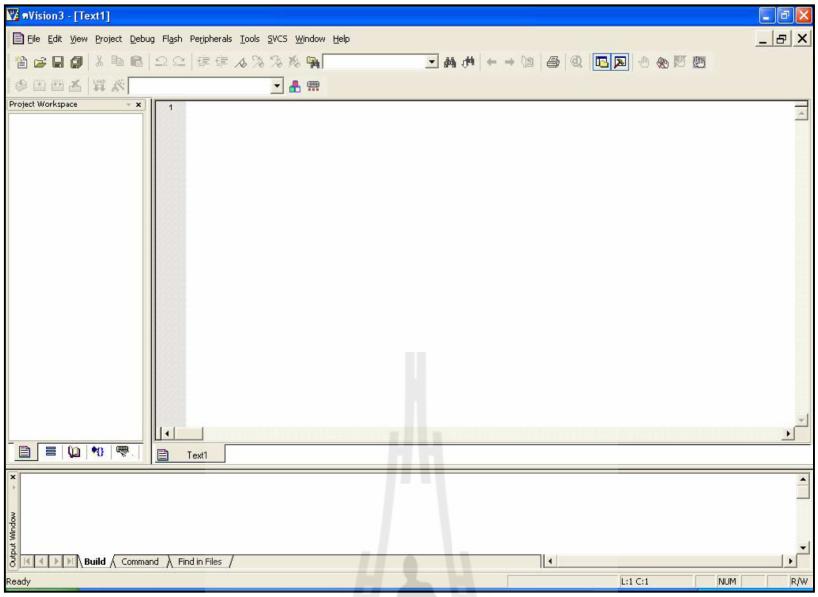
5.1 X-TAL ให้กำหนดเป็น 12 MHz พร้อมกับเลือกกำหนดให้ใช้หน่วยความจำที่มีอยู่ใน MCU เป็นเงื่อนไข ในการแปลโปรแกรมของ Keil-CARM ด้วย ดังรูป



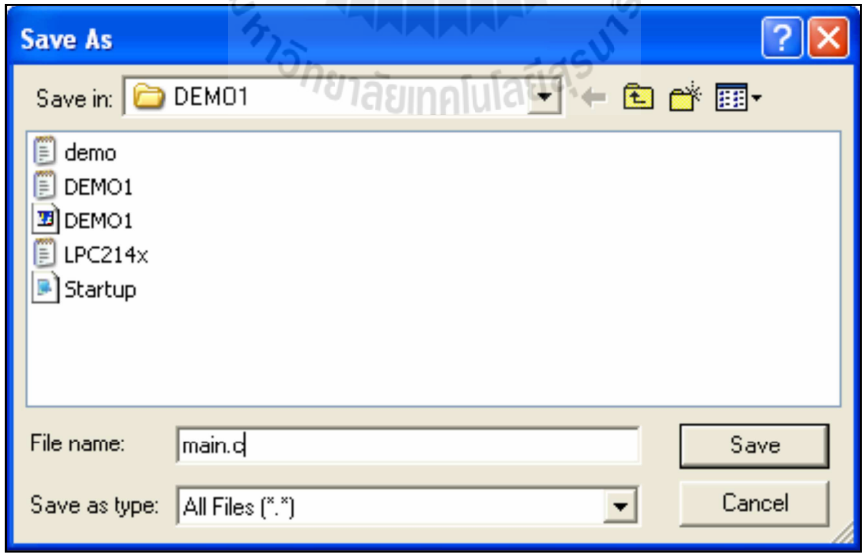
5.2 เลือกที่ Output ให้เลือกคลิกเมาส์ที่ค่าตัวเลือก Create HEX File พร้อมกับเลือกกำหนดรูปแบบของ Hex ให้เป็นแบบ HEX-386 แล้วเลือก OK ดังรูป



6. เริ่มต้นเขียน Source Code ภาษาซี โดยให้เลือกคลิกเมาส์ที่เมนูคำสั่ง File ➔ New... ซึ่งจะได้พื้นที่ในการเขียน Text File เกิดขึ้นมา โดยในครั้งแรกจะกำหนดชื่อตามค่า Default เป็น “Text1” ดังรูป



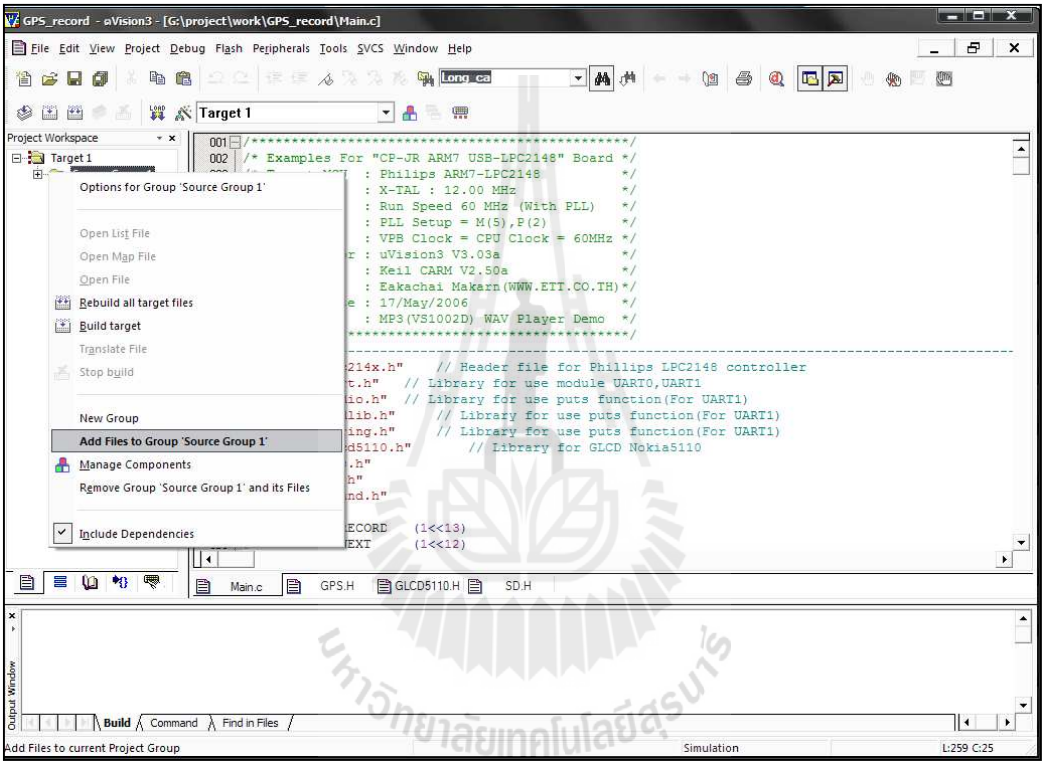
ในขั้นตอนนี้ให้ทำการพิมพ์ Source Code ภาษาซี ตามข้อกำหนดของ Keil-CARM ในพื้นที่เขียนโปรแกรมตามต้องต้องการ หลังจากพิมพ์คำสั่งภาษาซีเสร็จเรียบร้อยตามต้องการแล้ว ให้สั่ง Save ไฟล์ดังกล่าว โดยต้องกำหนดเป็นไฟล์ที่มีนามสกุลเป็น “.C” ในที่นี้ขอแนะนำให้สั่ง Save โดยใช้คำสั่ง File ➔ Save As... แล้วกำหนดชื่อและนามสกุลของไฟล์เป็น .main.c” ดังรูป



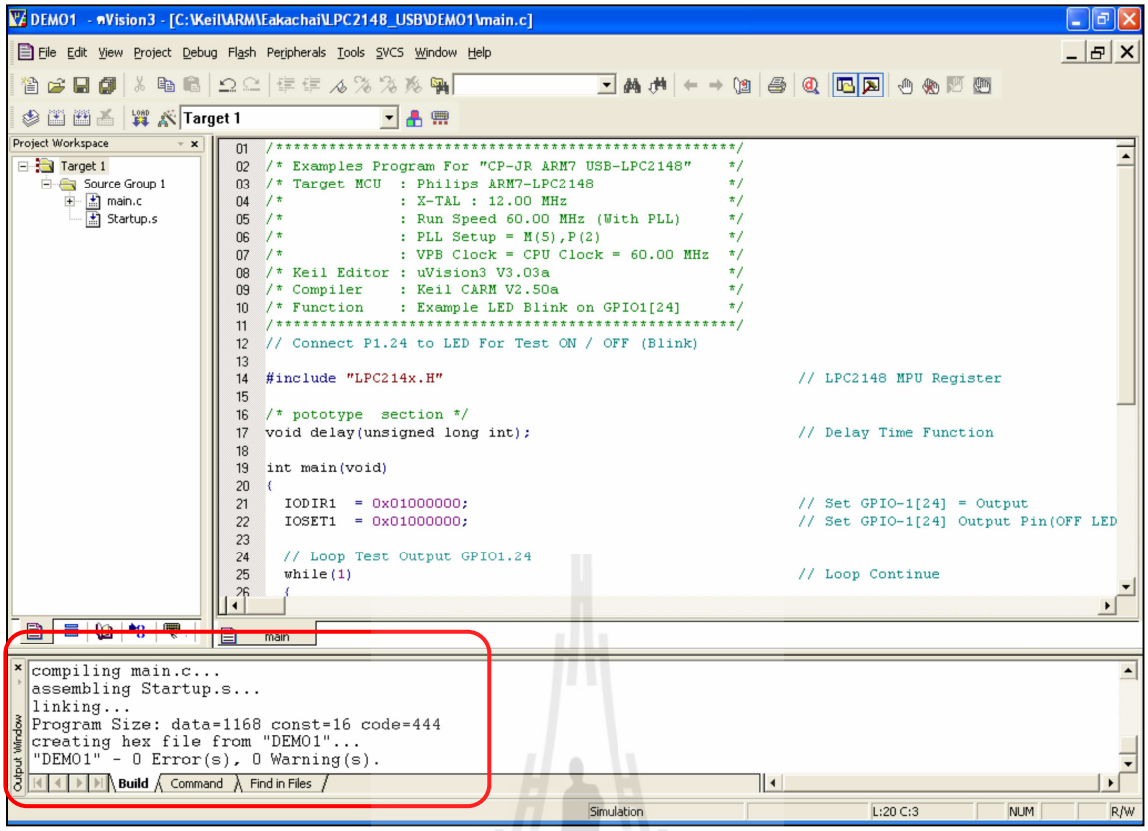
ซึ่งหลังจากที่สั่ง Save ไฟล์เป็น “main.c” แล้วจะเห็นว่าลักษณะสีของตัวอักขระต่างๆในโปรแกรมจะเกิดการเปลี่ยนแปลงไปตามหน้าที่ เช่น Comment, ตัวแปร และ คำสั่ง เป็นต้น ซึ่งส่วนนี้เป็นข้อดี

ของ Keil uVision3 ซึ่งสามารถแยกและแสดงตัวอักขระได้อย่างเป็นหมวดหมู่ ทำให้ง่ายต่อการอ่านโปรแกรมด้วย

7. ทำการสั่ง Add File ต่างๆเข้ากับ Project File โดยให้เลือกคลิกเมาส์ที่ด้านซ้ายของหน้าต่าง จากนั้นให้เลือกที่ Add Files to Group “Source Group 1 “ แล้วเลือกที่ Add File ที่ต้องการจะเพิ่มเข้าไปใช้งานร่วมกับ Project Fileโดยในครั้งแรกให้เลือก Files of type เป็น “C Source files(*.c)” ซึ่งจะปรากฏชื่อไฟล์ต่างๆที่เป็น Source Code ภาษาซีให้เห็น โดยในที่นี้ให้เลือกคลิกเมาส์ที่ไอคอนของไฟล์ชื่อ “main.c” แล้วเลือก Add เพื่อสั่งเพิ่มไฟล์ชื่อ “Startup.s” เข้าไปรวมกับ Project Files ที่เราสร้างไว้



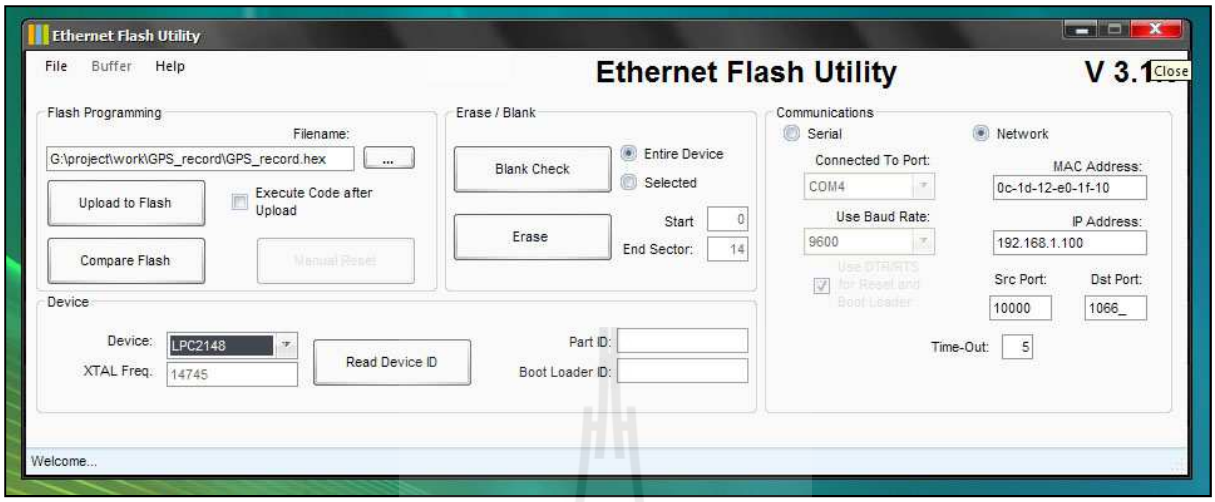
8. ให้ทำการสั่งแปลโปรแกรมที่เราเขียนขึ้นเรียบร้อยแล้ว โดยให้คลิกเมาส์ที่เมนูคำสั่ง Projects ➔ Rebuild all target files ซึ่งโปรแกรม Keil uVision3 จะทำการสั่งให้โปรแกรม Keil-CARM ทำการแปลคำสั่งให้ทันที ซึ่งหลังจากสั่งแปลโปรแกรมแล้วได้ผลถูกต้องและไม่เกิดข้อผิดพลาดใดๆขึ้น (0 Error และ 0 Warning) ก็ได้ Hex File ซึ่งมีชื่อเหมือนกันกับชื่อของ Project File ที่สร้างไว้ ซึ่งผู้ใช้สามารถนำ Hex File ดังกล่าวไปทำการ Download ให้กับ MCU ได้ทันที



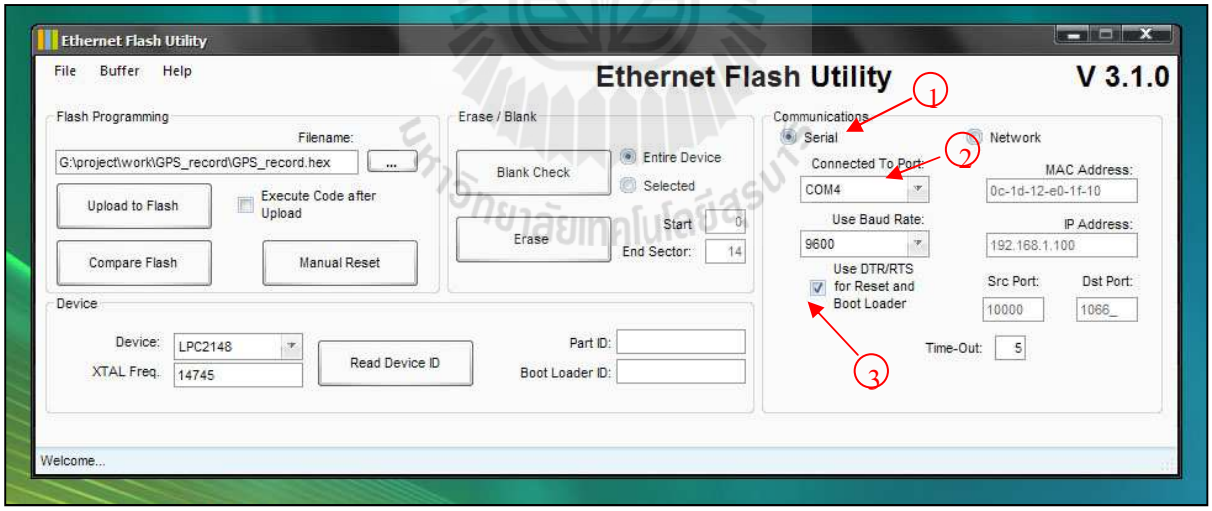
มหาวิทยาลัยเทคโนโลยีสุรนารี

การโหลดโปรแกรมทดสอบ

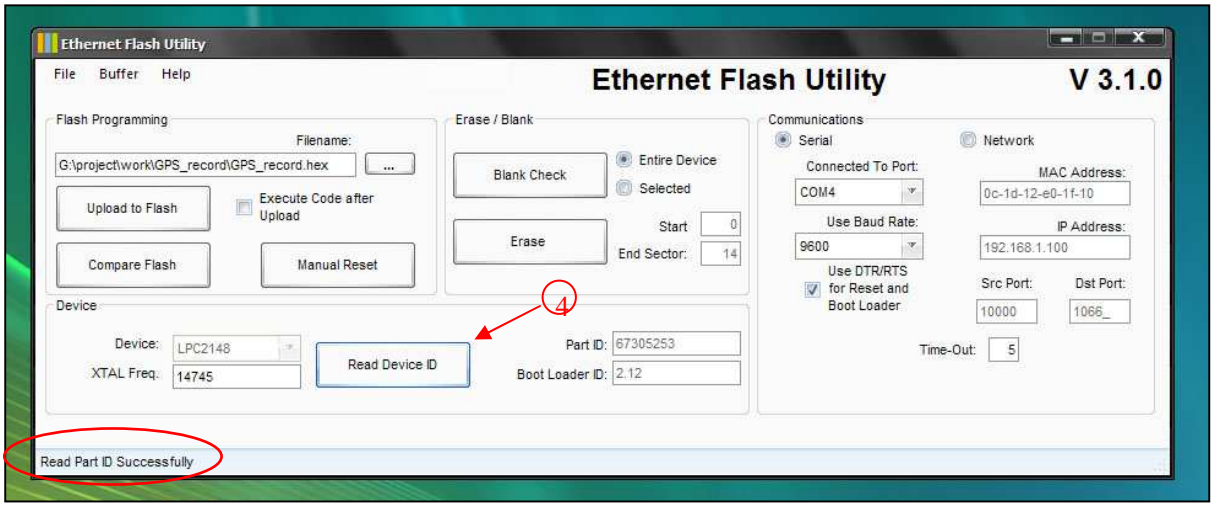
เปิดโปรกแกรม Ethernet Flash Utility จะได้หน้าต่างดังรูป



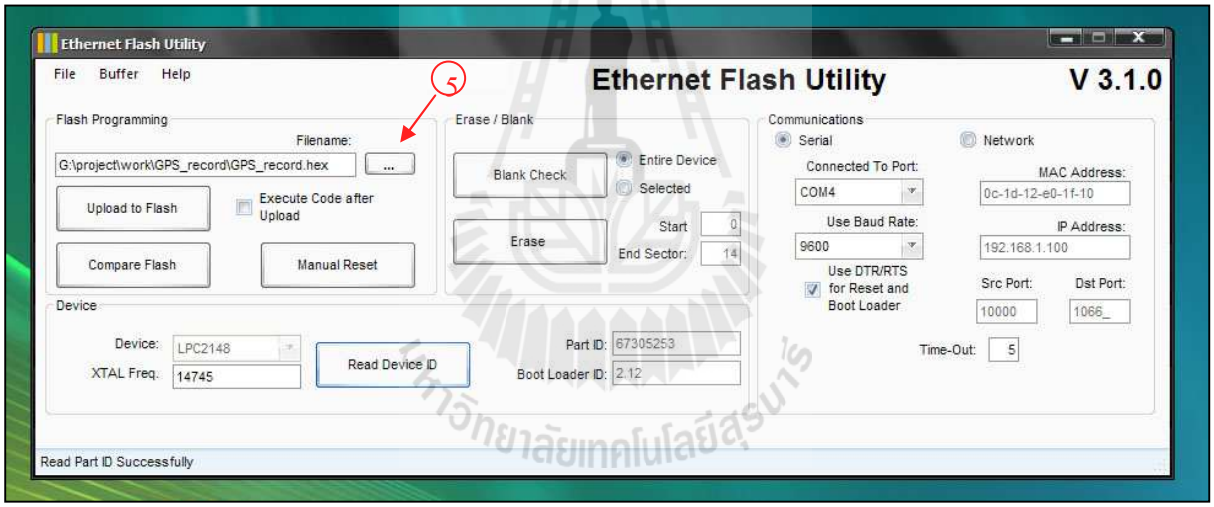
เลือกรูปแบบการเชื่อมต่อเป็นแบบ Serial ➔ เลือก COM ให้ถูกต้อง ➔ Baud Rate 9600 ➔ เลือกที่ช่อง Use DTR/RTS for Reset and Boot Loader

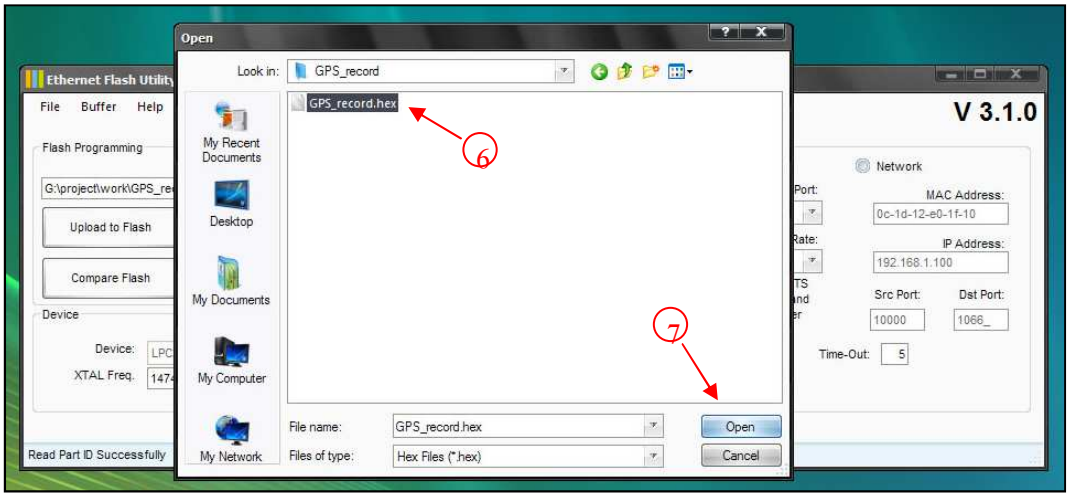


กด Switch Reset ที่ บอร์ดไมโครคอนโทรลเลอร์ แล้วเลือกที่ Read Device ID หากเชื่อมต่อกับไมโครคอนโทรลเลอร์สำเร็จจะแสดงข้อความ Read Part ID Successfully

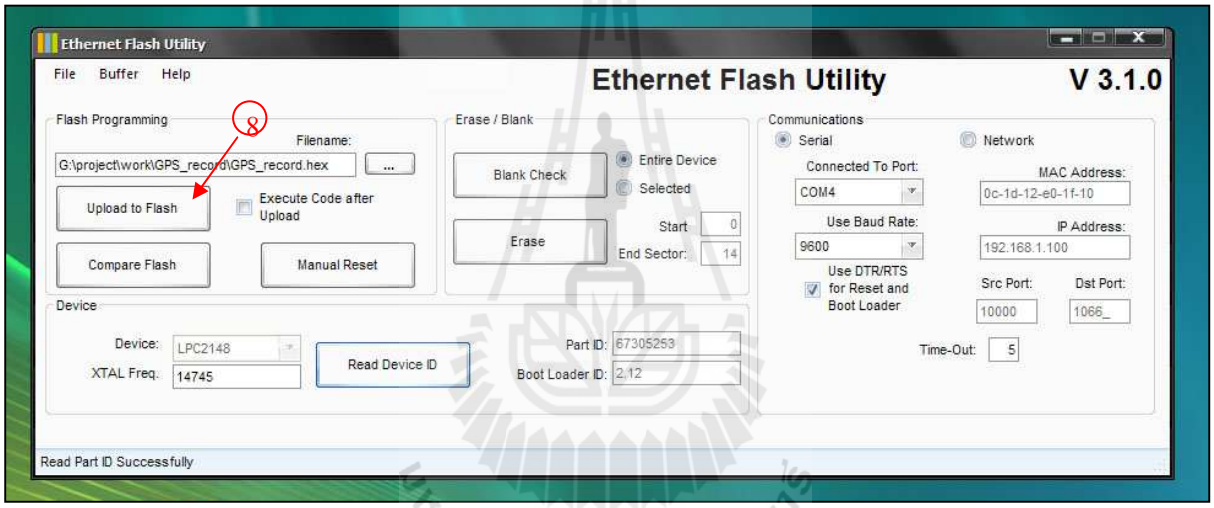


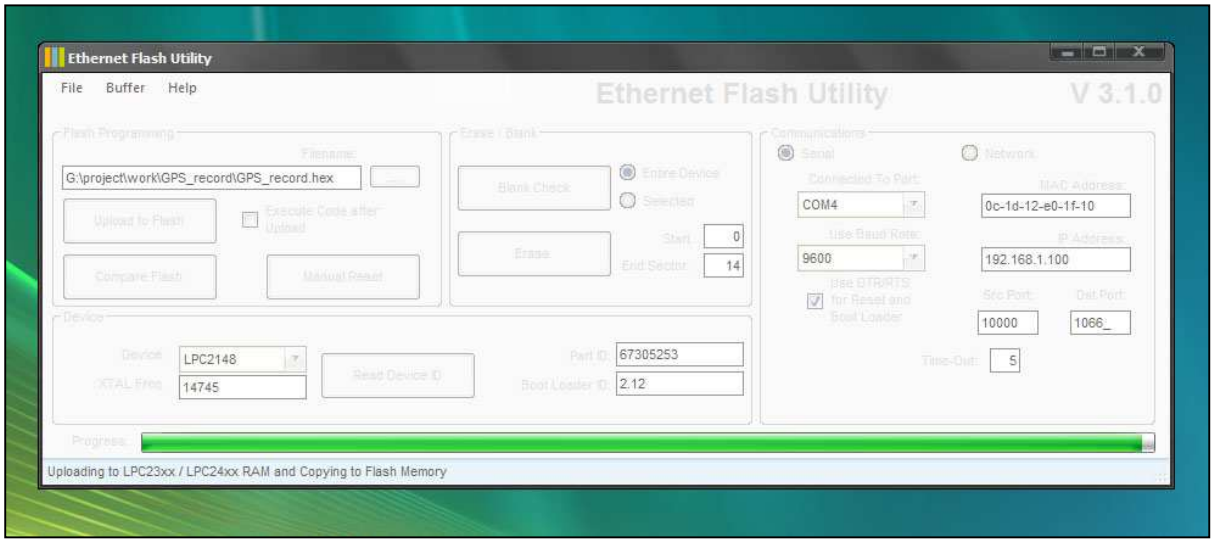
เมื่อเชื่อมต่อเรียบร้อยแล้ว ทำการโหลดไฟล์ .HEX โดยเลือกที่ Brown เลือกไฟล์ .HEX ที่ต้องการ แล้วเลือก Open ดังรูป





เลือกที่ Upload to Flash และรอจนกว่าโปรแกรมจะทำการโหลดเสร็จ ดังรูป





ประวัติผู้วิจัย

ผู้ช่วยศาสตราจารย์ ดร. รังสรรค์ ทองทา สำเร็จการศึกษาวิศวกรรมไฟฟ้า จากมหาวิทยาลัยขอนแก่น เมื่อปี พ.ศ. 2531 และ MS และ PH.D. จาก Florida Institute of Technology, USA เมื่อปี 1993 และ 1998 ตามลำดับ เริ่มงานในตำแหน่งวิศวกรที่บริษัท Elcom Research เมื่อปี พ.ศ. 2531 และบริษัท การบินไทย จำกัด ในปี พ.ศ. 2532 เมื่อปี พ.ศ. 2533 ได้เริ่มในตำแหน่งอาจารย์ที่ภาควิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยขอนแก่น และมหาวิทยาลัยเทคโนโลยีสุรนารีในปี พ.ศ. 2536

มหาวิทยาลัยเทคโนโลยีสุรนารี