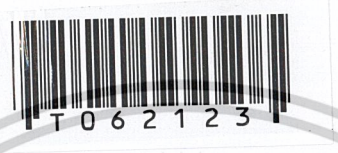
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การประยุกต์ใช้ rfPIC สำหรับโครงข่ายเซนเซอร์แบบไร้สาย

Application of rfPIC for Wireless Sensor Networks



โดย

นายไพฑูรย์ สุขล้าน

นายยุทธภูมิ เทียมเมืองแพน

เลขหมู่.....
เลขทะเบียน..... 62123
วัน,เดือน,ปี..... 31 ม.ค. 2549
b..... 11571925
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงาน	การประยุกต์ใช้ rfPIC สำหรับโครงข่ายเซนเซอร์แบบไร้สาย	
ชื่อนักศึกษา	นายไพฑูรย์ สุขล้าน	รหัสนักศึกษา 45015808
	นายยุทธภูมิ เทียมเมืองแพน	รหัสนักศึกษา 45015811
อาจารย์ที่ปรึกษา	อาจารย์พิชญ สุพรรณกูล	
	ผศ. มนต์ชัย แช่มช้อย	
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ภาควิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2547	

โครงงานนี้ได้รับการตรวจสอบเรียบร้อยแล้ว พร้อมที่จะได้รับการสอบ

.....

(อาจารย์พิชญ สุพรรณกูล)

อาจารย์ที่ปรึกษา

.....

(ผศ. มนต์ชัย แช่มช้อย)

อาจารย์ที่ปรึกษา

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การประยุกต์ใช้ rfPIC สำหรับโครงข่ายเซนเซอร์แบบไร้สาย	
ชื่อนักศึกษา	นายไพฑูรย์ สุขล้าน	รหัสนักศึกษา 45015808
	นายยุทธภูมิ เทียมเมืองแพน	รหัสนักศึกษา 45015811
อาจารย์ที่ปรึกษา	อาจารย์พิชญ สุพรรณกูล	
	ผศ. มนต์ชัย แช่มช้อย	
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ภาควิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2547	

บทคัดย่อ

โครงงานนี้เป็นการศึกษาและ ประยุกต์ใช้ rfPIC ในงานโครงข่ายเซนเซอร์แบบไร้สาย (Wireless Sensor Network) โดย rfPIC นี้เป็นไมโครคอนโทรลเลอร์ที่ใช้ ในการประมวลผลการติดต่อสื่อสารแบบไร้สายในย่านความถี่วิทยุ โครงงานนี้จะประกอบด้วยโหนดต่างๆ ซึ่งแต่ละโหนด จะประกอบด้วยภาครับ ภาคส่งและภาคเซนเซอร์อยู่ในโหนดตัวเดียวกัน ซึ่งลักษณะการทำงานจะเป็นการติดต่อสื่อสารไร้สาย ระหว่างโหนดโดยมีตัวเซ็นเซอร์เป็นตัวควบคุมการส่งสัญญาณข้อมูล และจะมีการส่งข้อมูลต่อไปยังโฮสต์เพื่อบอกสถานะและหมายเลขประจำตัวของโหนดนั้นและ จะมีโปรโตคอลในการสื่อสารระหว่างโหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Application of rfPIC for Wireless Sensor Network	
Student	Mr. Paitoon Sukhlan	ID 45015808
	Mr. Yutthapoom Tieammungpan	ID 45015811
Advisor	Mr. Pichaya Spanakoon	
Co-Advisor	Asst.Prof. Monchai Chamchoy	
Graduate Level	Bachelor Degree of Information Engineering	
Department	Information Engineering	
Acadamic Year	2003	

Abstract

This project is education and application of rfPIC for wireless sensor networks. This rfPIC is microcontroller use for operate wireless communication in radio frequency. This project consist of node which each node consist of receiver transmitter and sensor that is in the same node which operating system is wireless communication between each node which is controlled by sensor and forward packet to host for indicate status and identify number and it has protocol which use for communication between each node.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานการประยุกต์ใช้ rfPIC สำหรับโครงข่ายเซนเซอร์แบบไร้สาย สำเร็จลุล่วงได้เนื่องด้วยการสนับสนุนของท่านอาจารย์ พิชญ สุพรรณกูล และ ท่าน ผศ. มนต์ชัย แช่มช้อย ซึ่งได้ให้คำปรึกษาเกี่ยวกับโครงงานเป็นอย่างดี จึงขอขอบคุณมา ณ.โอกาสนี้ และที่ขาดไม่ได้ก็ขอขอบคุณ คุณพ่อ คุณแม่ที่ได้ให้กำเนิดเกิดมา ครู อาจารย์ทุกท่าน ที่ได้ประสิทธิประศาสตร์วิชาจนมีความรู้ประดับตัว และขอขอบคุณเพื่อนๆที่ให้กำลังใจและข้อเสนอแนะที่ดี

ไพฑูรย์ สุขล้าน

ยุทธภูมิ เทียมเมืองแพน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	จ
สารบัญรูป	ฉ
สารบัญตาราง	ซ
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มาของปัญหา	1
1.2 วัตถุประสงค์ของโครงงาน	1
1.3 ขอบเขตของโครงงาน	2
1.4 ผลที่คาดว่าจะได้รับ	2
บทที่ 2 ไมโครคอนโทรลเลอร์แบบ rfPIC	3
2.1 rfPIC12F675	3
2.2 rfRXDO420	10
บทที่ 3 การออกแบบ	49
3.1 การออกแบบโครงงานโดยรวม	49
3.2 ภาครับ (Receiver)	50
3.3 ภาคส่ง (Transmitter)	51
3.4 เซนเซอร์	52
3.5 โหนด (Node)	53
3.6 โปรโตคอล (Protocol)	54
บทที่ 4 ผลการทดลอง	55
4.1 วัตถุประสงค์การทดลอง	55
4.2 อุปกรณ์การทดลอง	55
4.3 การทดลองกับเซนเซอร์	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
4.4 การทดลอง รับ- ส่ง ข้อมูล	58
4.5 การทดลองเชื่อมต่อเซนเซอร์กับภาคส่ง	59
บทที่ 5 สรุปผลการทดลองและปัญหาที่พบ	62
เอกสารอ้างอิง	
ภาคผนวก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
รูปที่ 1.1 โครงข่ายเซนเซอร์ไร้สาย	1
รูปที่ 1.2 โครงสร้างของโครงงาน	2
รูปที่ 2.1 การจัดขาของไมโครคอนโทรลเลอร์ rfPIC12F675	5
รูปที่ 2.2 วงจรASK Transmitter 433.93 MHz	5
รูปที่ 2.3 แผนที่หน่วยความจำโปรแกรมและสแต็ค (STACK) สำหรับ rfPIC12F675	7
รูปที่ 2.4 การจัดสรรพื้นที่ของหน่วยความจำข้อมูลแรม และรีจีสเตอร์ในของ rfPIC12F675	8
รูปที่ 2.5 วงจร ASK Transmitter 433.93 MHz	9
รูปที่ 2.6 การจัดขาของไมโครคอนโทรลเลอร์ rfRXD0420	10
รูปที่ 2.7 บล็อกไดอะแกรมของ rfRXD0420	11
รูปที่ 2.8 ไดอะแกรมลอจิกของการเกิดสัญญาณอินเตอร์รัพต์	24
รูปที่ 2.9 ไดอะแกรมการทำงานของไทเมอร์ 0	29
รูปที่ 2.10 ไดอะแกรมการทำงานของไทเมอร์ 1	32
รูปที่ 2.11 ไดอะแกรมการทำงานของโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล	36
รูปที่ 2.12 วงจรสมมูลอินพุตอนาลอกของโมดูลแปลงสัญญาณอนาลอก เป็นดิจิตอลใน rfPIC12F675	36
รูปที่ 2.13 โหมดรีเซต(CM2-CM2: 000)	41
รูปที่ 2.14 โหมดวงจรเปรียบเทียบอิสระและค่าส่งออกเอาต์พุต(CM2-CM0: 001)	41
รูปที่ 2.15 โหมดวงจรเปรียบเทียบอิสระ(CM2-CM0: 010)	42
รูปที่ 2.16 โหมดการวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายใน และค่าส่งออกเอาต์พุต (CM2-CM1: 011)	42
รูปที่ 2.17 โหมดการวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายใน (CM2-CM1: 100)	43
รูปที่2.18 โหมดวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายในแบบเลือกอินพุตได้ และส่งค่าออกเอาต์พุต(CM2-CM0:101)	43
รูปที่ 2.19 โหมดวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายในแบบเลือกอินพุตได้(CM2-CM1: 110)	44
รูปที่ 2.20 โหมดหยุดการทำงาน (CM2-CM0: 111)	44
รูปที่ 2.21 วงจรส่วนเอาต์พุตของโมดูลเปรียบเทียบแรงดันอนาลอกใน rfPIC12F675	45
รูปที่ 2.22 ไดอะแกรมการทำงานของโมดูลสร้างแรงดันอ้างอิงภายใน rfPIC12F675	46
รูปที่ 3.1 ลำดับขั้นตอนการทำงาน	49
รูปที่ 3.2 ภาครับ	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
รูปที่ 3.3 ภาคส่ง	51
รูปที่ 3.4 เซนเซอร์	52
รูปที่ 3.5 โหนด	53
รูปที่ 3.6 โปรโตคอลการสื่อสารระหว่างโหนด	54
รูปที่ 4.1 การปรับค่าแรงดันอ้างอิงให้ได้1.25 โวลต์	56
รูปที่ 4.2 การวัดเปรียบเทียบค่าความชื้น	57
รูปที่ 4.3 ผลการทดลองส่งค่า 0xFF	58
รูปที่ 4.4 ผลการทดลองส่งค่า 0x3A	58
รูปที่ 4.5 เซนเซอร์ตรวจจับความชื้นที่ระดับ 5 และ ส่งค่า	59
รูปที่ 4.6 ภาครับการส่งค่าเมื่อเซนเซอร์ตรวจจับที่ระดับ 5	59
รูปที่ 4.7 เซนเซอร์ตรวจจับความชื้นที่ระดับ 9 และ ส่งค่า	60
รูปที่ 4.8 ภาครับการส่งค่าเมื่อเซนเซอร์ตรวจจับที่ระดับ 9	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 2.1รายละเอียดขาพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ rfPIC12F675	6
ตารางที่ 2.2 รายละเอียดขาพอร์ตทั้งหมดของ rfRXD0420 PINOUT I/O	12
ตารางที่ 2.3 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ STATUS	13
ตารางที่ 2.4 รายละเอียดของการบิตต่างๆ ในรีจีสเตอร์ OPTION	14
ตารางที่ 2.5 การกำหนดอัตราส่วนของปรีสเกลเลอร์	15
ตารางที่ 2.6 รายละเอียดของการบิตต่างๆ ในรีจีสเตอร์ PCON	16
ตารางที่ 2.7 รายละเอียดของการบิตต่างๆ ในรีจีสเตอร์ OSCCAL	16
ตารางที่ 2.8 รายละเอียดของการบิตต่างๆ ในรีจีสเตอร์ WPU	19
ตารางที่ 2.9 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ INTCON	21
ตารางที่ 2.10 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ PIE1	23
ตารางที่ 2.11 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ EECON1	26
ตารางที่ 2.12 รายละเอียดของการบิตต่างๆ ในรีจีสเตอร์ OPTION	29
ตารางที่ 2.13 อัตราการทำงานของปรีสเกลเลอร์เมื่อทำงานกับไทม์เมอร์ 0 และวอตช์ด็อกไทม์เมอร์	31
ตารางที่ 2.14 การเลือกค่าของตัวเก็บประจุที่มีความถี่สัญญาณนาฬิกาต่างๆ ของไทเมอร์1	34
ตารางที่ 2.15 ความสัมพันธ์ของคาบเวลาสัญญาณนาฬิกาของโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล กับความถี่สูงสุดของสัญญาณนาฬิกาหลักของ rfPIC12F675	38
ตารางที่ 2.16 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ EECON	39
ตารางที่ 4.1 ผลจากการทดลอง	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

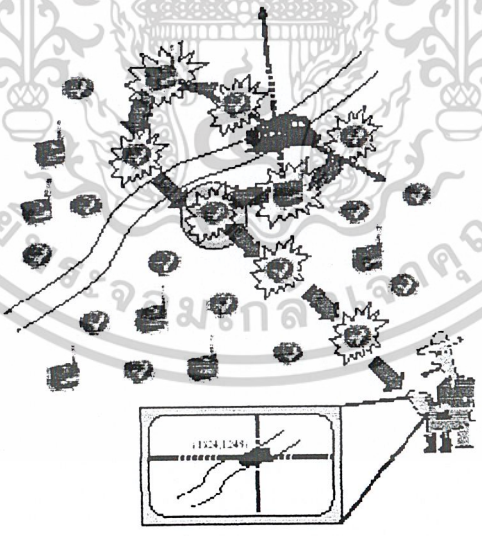
1.1 ความสำคัญและที่มา

1.1.1 ที่มาของปัญหา

สำหรับเซนเซอร์ที่มีการเชื่อมต่อกันโดยสายนำสัญญาณนั้น อาจจะมีความยุ่งยากในเรื่องของการติดตั้ง และการซ่อมบำรุง ประกอบกับในปัจจุบันนี้ เทคโนโลยีไร้สายได้เข้ามามีส่วนในชีวิตประจำวันเรามากขึ้น จึงได้มีแนวคิดที่จะพัฒนาเซนเซอร์ให้มีการเชื่อมต่อกันแบบไร้สายขึ้นมา เพื่อให้สะดวกในการติดตั้งและการใช้งาน

1.1.2 แนวคิดในการแก้ไขปัญหา

ประยุกต์เอาไมโครคอนโทรลเลอร์แบบ rfPIC มาใช้ในการเชื่อมต่อเซนเซอร์แต่ละตัวแบบไร้สายดังรูปที่ 1.1 ซึ่งจะช่วยอำนวยความสะดวก เนื่องจากมีประสิทธิภาพในการใช้งาน มีขนาดที่เล็กและ มีอายุการใช้งานที่ยาวนาน



รูปที่ 1.1 โครงข่ายเซนเซอร์ไร้สาย

1.2 วัตถุประสงค์ของโครงงาน

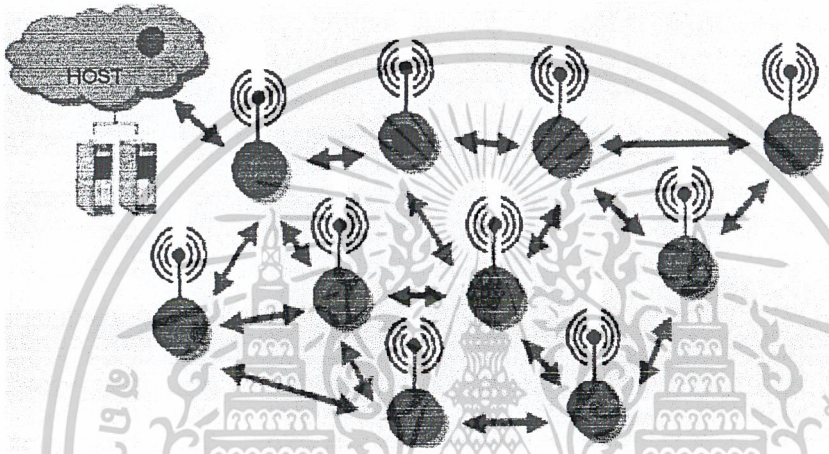
- เพื่อศึกษาการใช้งานไมโครคอนโทรลเลอร์ rfPIC ในโครงข่ายเซนเซอร์ไร้สาย
- เพื่อพัฒนาให้มีขนาดที่เล็กลง
- เพื่อพัฒนาโปรโตคอลที่ใช้ในการสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน

- ภาครับและภาคส่งสามารถติดต่อสื่อสารกันได้
- สามารถประยุกต์ใช้กับเซนเซอร์ในงานควบคุม อย่างใดอย่างหนึ่งได้ เช่น เซนเซอร์ความชื้น หรือ เซนเซอร์อุณหภูมิ
- มีโปรโตคอลในการสื่อสาร

ดังแสดงในรูปที่ 1.2



รูปที่ 1.2 โครงสร้างของโครงงาน

1.4 ผลที่คาดว่าจะได้รับ

- โครงงานนี้สามารถประยุกต์ใช้กับงานจริงได้
- นักศึกษาสามารถนำเอาความรู้ที่ได้มานี้ประยุกต์ใช้ในงานอย่างอื่นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ไมโครคอนโทรลเลอร์แบบ rfPIC

บทนี้กล่าวถึงลักษณะคุณสมบัติของไมโครคอนโทรลเลอร์ แบบ rfPIC ทั้งของภาครับและของภาคส่ง ตลอดจนการประยุกต์ใช้งาน

2.1 rfPIC12F675

ลักษณะเฉพาะ

- มี 20 ขา
- มีขนาดค่อนข้างเล็กมาก
- ความจำแรมข้อมูล Sram 64 x 8 ไบต์
- หน่วยความจำข้อมูลอีอีพรอม 128 x 8 ไบต์
- มีวอตซ์ด็อกไทเมอร์ (WDT) ที่มีวงจรออสซิลเลเตอร์ในตัวทำให้มีความน่าเชื่อถือในการทำงาน
- ความทนทานในการเขียน 100,000 ครั้ง ของ FLASH
- ความทนทานเขียน 1,000,000 ครั้ง ของ EEPROM
- FLASH/data EEPROM หน่วยความจำสามารถคงไว้ได้มากกว่า 40 ปี
- ป้องกันการโปรแกรม
- 6 ขา input และ output ในการควบคุมและอินเตอร์รัพต์
- กระแสสูงในการขับ LED ได้โดยตรง
- การเปรียบเทียบสัญญาณอนาลอกอ้างอิงได้ 16 level ภายใน
- สัญญาณอนาลอกแปลงไปสู่สัญญาณดิจิตอล มี 10 bits 4 channel
- Timer 0 มี 8 bits timer/counter กับ 8 bits-prescaler
- Timer 1 มี 16 bits timer / counter 3 bits-prescaler
- Timer 1 สามารถใช้ LP ooscillator ใน INTOSC mode
- กินกำลังต่ำที่ Vdd = 3 v.
- FLASH/data EEPROM หน่วยคามจำสามารถคงไว้ได้มากกว่า 40 ปี
- 14 mA ในการส่ง +6 dBm ที่ 434 MHz
- 4 mA ในการส่ง -15 dBm ที่ 434 MHz
- 500 uA , 4.0 MHz INTOSC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

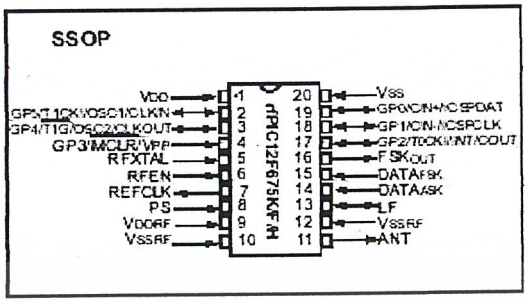
- 0.6 uA SLEEP กับการควบคุมด้วย enabled
- 0.1 uA standby current
- แรงดันที่ใช้ 2.0 V. – 5.5 V.
- ความถี่สัญญาณนาฬิกา ตั้งไฟตรงถึง 20MHz
- ขนาดความจำโปรแกรม 1 กิโลเวิร์ด x 14
- เลือกป้องกันข้อมูลทั้งในหน่วยความจำโปรแกรมและหน่วยความจำข้อมูลเกิน 4MHz
- โหมด HS ใช้คริสตอลความถี่สูง สูงสุดไม่เกิน 20 MHz
- สามารถโปรแกรมในวงจรได้
- กระแสซิงค์และกระแสซอร์สของพอร์ต 25 mA
- มีโมดูลเปรียบเทียบแรงดันอนาลอก 1 ชุด
- มีโมดูลสร้างแรงดันอ้างอิงสำหรับวงจรเปรียบเทียบแรงดันอนาลอก
- มีวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล 10 บิต จำนวน 4 ช่อง
- อัตราข้อมูล Ask : 0 – 40 kbps
- อัตราข้อมูล Fsk : 0 – 40 kbps โดยการดึงของ Crystal
- กำลังทาง Output : +10 dBm ถึง -12 dBm ใน 4 steps
- ความถี่ในการส่งถูกตั้งโดย Crystal oscillator

รายละเอียดทั้งหมดแสดงในรูปที่ 2.1 และตารางที่ 2.1

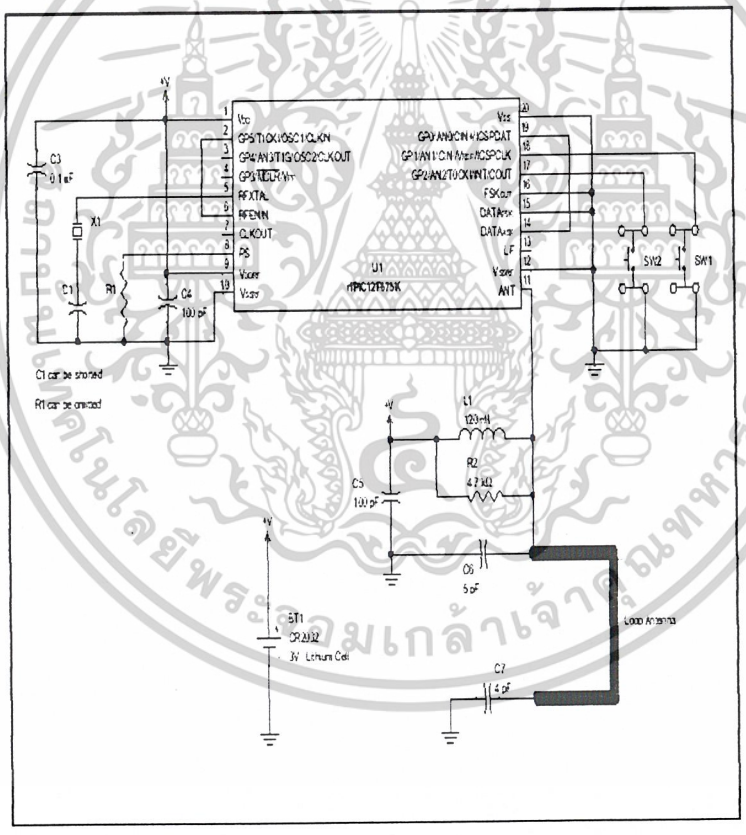
การประยุกต์ใช้งาน

- ระบบป้องกันขโมย
- เซนเซอร์ความดัน
- เซนเซอร์ไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 การจัดขาของไมโครคอนโทรลเลอร์ rfPIC12F675



รูปที่ 2.2 วงจร ASK Transmitter 433.93 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

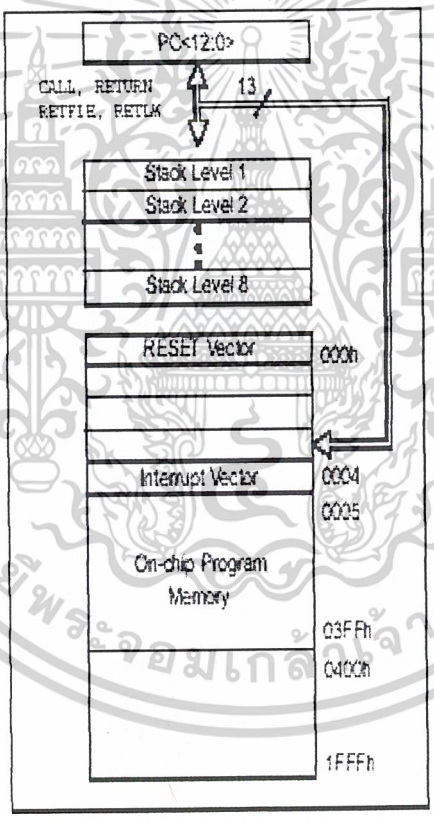
ตารางที่ 2.1 รายละเอียดขาพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ rfPIC12F675

PIN		BUFFER		WEAK	DESCRIPTION
		IN	OUT	PULL-UP	
1	V_{DD}	Direct	---	---	ไฟเลี้ยง
2	GP5	TIL	CMOS	Prog	ส่วนควบคุมของการอินเตอร์รัพต์
	T1CKI	ST	---	---	ไทเมอร์ 1สัญญาณนาฬิกา
	OSC1	X tal	---	Bias	ขาสัญญาณ XTAL
	CLKIN	ST	---	---	สัญญาณ RC ภายนอก หรือ สัญญาณนาฬิกาเข้า
3	GP4	TIL	CMOS	Prog	ส่วนควบคุมของการอินเตอร์รัพต์
	T1G	ST	---	---	ไทเมอร์ 1 Gate
	AN2	Analog	---	---	A/D ช่อง 3 สัญญาณเข้า
	OSC2	---	X tal	Bias	ขาสัญญาณ XTAL
	CLKOUT	---	CMOS	---	Tosc/4 สัญญาณนาฬิกาอ้างอิง
4	GP3	TIL	---		ส่วนควบคุมของการอินเตอร์รัพต์
	MCLR	ST	---	No	ขารีเซตหลัก
	Vpp	HV	---	---	แรงดันของโปรแกรม
5	RFXTAL	Xtal	Xtal	Bias	RF คริสตอล
6	RFEN	TIL	---	---	RF Enable
7	REFCLK	---	CMOS	---	ส่วนควบคุมของการอินเตอร์รัพต์
8	PS	Analog	---	Bias	เลือกแรงดัน
9	V_{DDRF}	Direct	---	---	RF แหล่งจ่ายความถี่
10	V_{SSRF}	Direct	---	---	RF กราวนด์
11	ANT	---	OD	---	RF แหล่งจ่ายกระแสให้สายอากาศ
12	V_{SSRF}	Direct	---	---	RF กราวนด์
13	LF	Analog	Analog	---	ลูป Filter
14	$DATA_{ASK}$	TTL	---	---	ASK Modulation Data
15	$DATA_{FSK}$	TIL	---	---	FSK Modulation Data
16	FSKout	---	OD	---	FSK output to modulate referent crytal
17	GP2	ST	CMOS	Prog	ส่วนควบคุมของการอินเตอร์รัพต์
	AN2	Analog	---	---	A/D ช่อง 3 สัญญาณเข้า
	COUT	---	CMOS	---	เปรียบเทียบสัญญาณออก
	T0CKI	ST	---	---	สัญญาณนาฬิกา สำหรับ ไทเมอร์ 0
	INT	ST	---	---	สัญญาณอินเตอร์รัพต์ภายนอก
18	GP1	TIL	CMOS	Prog	ส่วนควบคุมของการอินเตอร์รัพต์
	AN1	Analog	---	---	A/D ช่อง 1 สัญญาณเข้า
	CIN-	Analog	---	---	เปรียบเทียบสัญญาณอินพุต -
	V_{REF}	Analog	---	---	แรงดันอ้างอิงภายนอก
	ICSPCLK	ST	---	---	ข้อมูลอนุกรมโปรแกรมนาฬิกา
19	GP0	TIL	CMOS	Prog	ส่วนควบคุมของการอินเตอร์รัพต์
	AN0	Analog	---	---	A/D ช่อง 0 สัญญาณเข้า
	CIN+	Analog	---	---	เปรียบเทียบสัญญาณอินพุต +
	ICSPDAT	TIL	CMOS	---	ข้อมูลอนุกรมโปรแกรม I/O
20	Vss	Direct	---	---	กราวนด์
Legend : TTL = TTL inter buffer, ST = Schmitt Trigger input buffer, OD = Open Drain output					

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

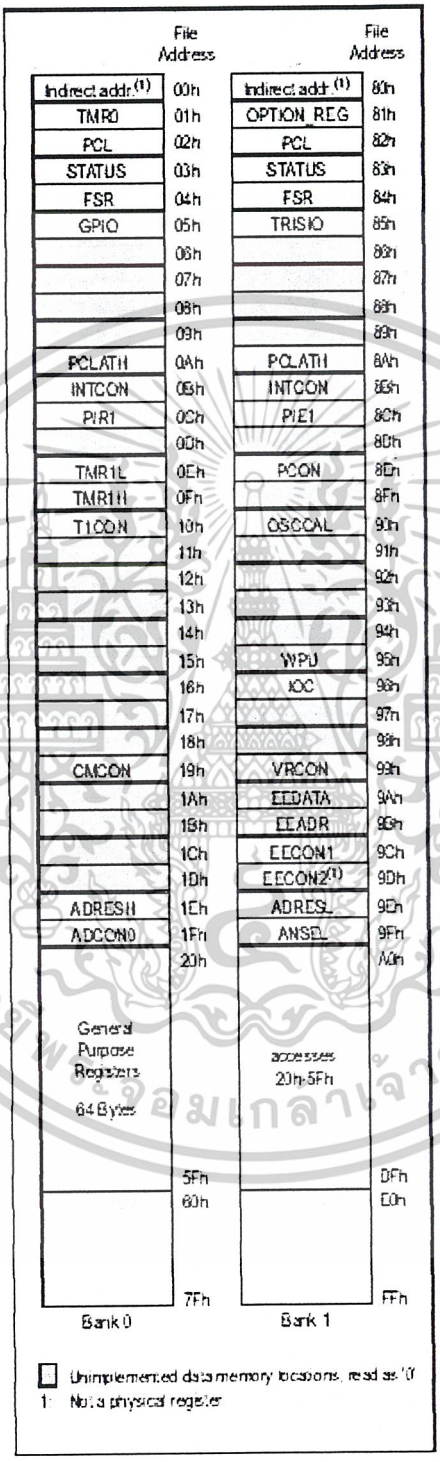
การจัดการสรรหน่วยความจำการจัดการสรรหน่วยความจำ

หน่วยความจำโปรแกรมของ rfPIC12F675 เป็นหน่วยความจำแบบเฟลช (Flash Memory) มีขนาด 1 กิโลเวิร์ด ใช้โปรแกรมเคาน์เตอร์ (PC) ขนาด 13 บิตเพื่อกำหนดหน่วยจำโปรแกรมในรูปที่ 2.6 เป็นการจัดสรรหน่วยความจำโปรแกรมของ PIC12F65 ซึ่งอยู่ในตำแหน่งแอดเดรส 0x0000-0x03FF แอดเดรสแรกสำหรับเก็บค่ารีเซตเวกเตอร์ และที่ตำแหน่ง 0x0004 เป็นที่สำหรับเก็บค่าแอดเดรสอินเตอร์รัพต์เวกเตอร์ ในการเขียนโปรแกรมจึงต้องเริ่มต้นการทำงานที่แอดเดรส 0x0005 จะเหมาะสมที่สุด และยังมีพื้นที่หน่วยความจำสำหรับเก็บค่าของโปรแกรมเคาน์เตอร์ (PC) ชั่วคราวขนาด 13 บิต เรียกว่า สแต็ก (Stack) สแต็กใน rfPIC12F675 สามารถเก็บค่าของ PC ได้ 8 ระดับ ดังแสดงในรูปที่ 2.3



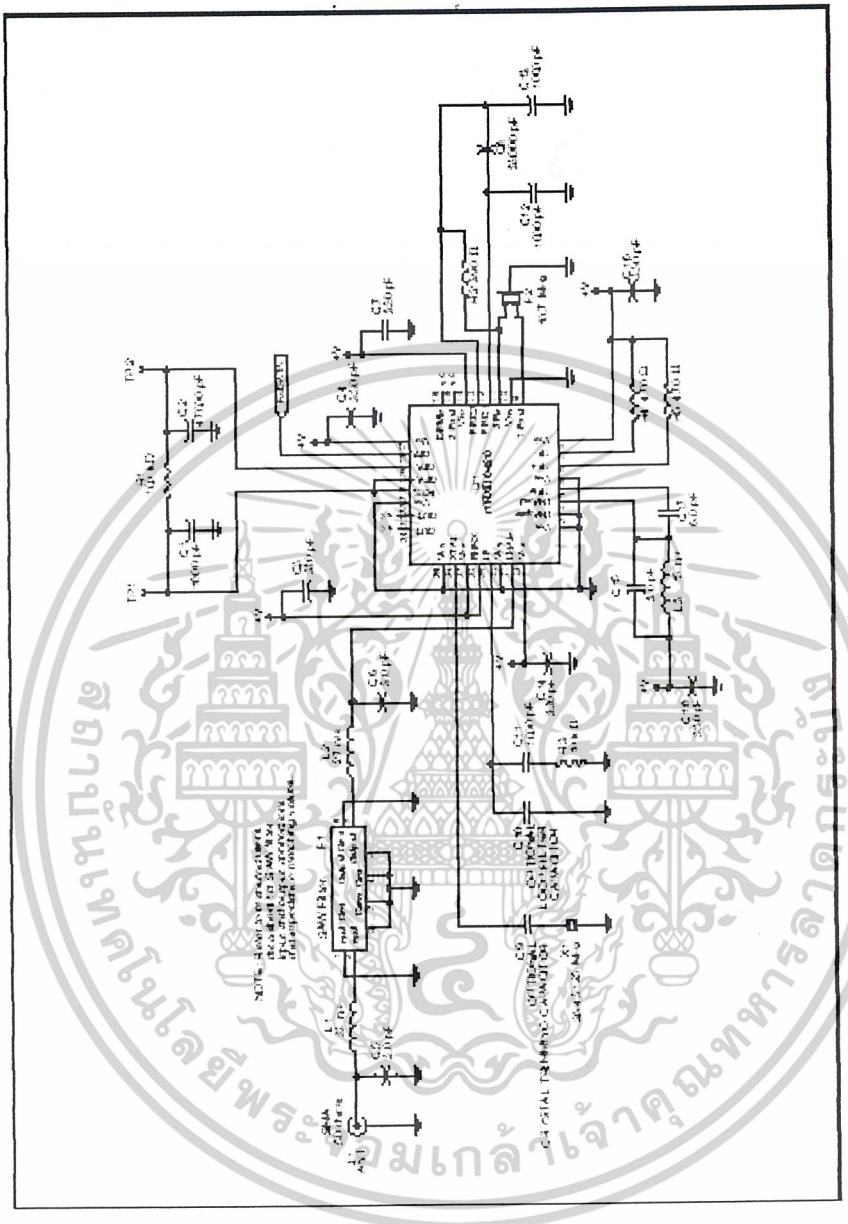
รูปที่ 2.3 แผนที่หน่วยความจำโปรแกรมและสแต็ค(STACK) สำหรับ rfPIC12F675

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การจัดสรรพื้นที่ของหน่วยความจำข้อมูลแรมและรีจีสเตอร์ในของ rfPIC12F675

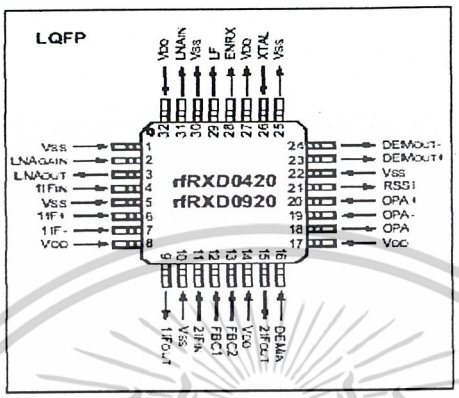
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 วงจร Receiver rfRXDO420

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 rfRXDO420 UHF ASK/FSK/FM Receiver



รูปที่ 2.6 การจัดขาของไมโครคอนโทรลเลอร์ rfRXD0420

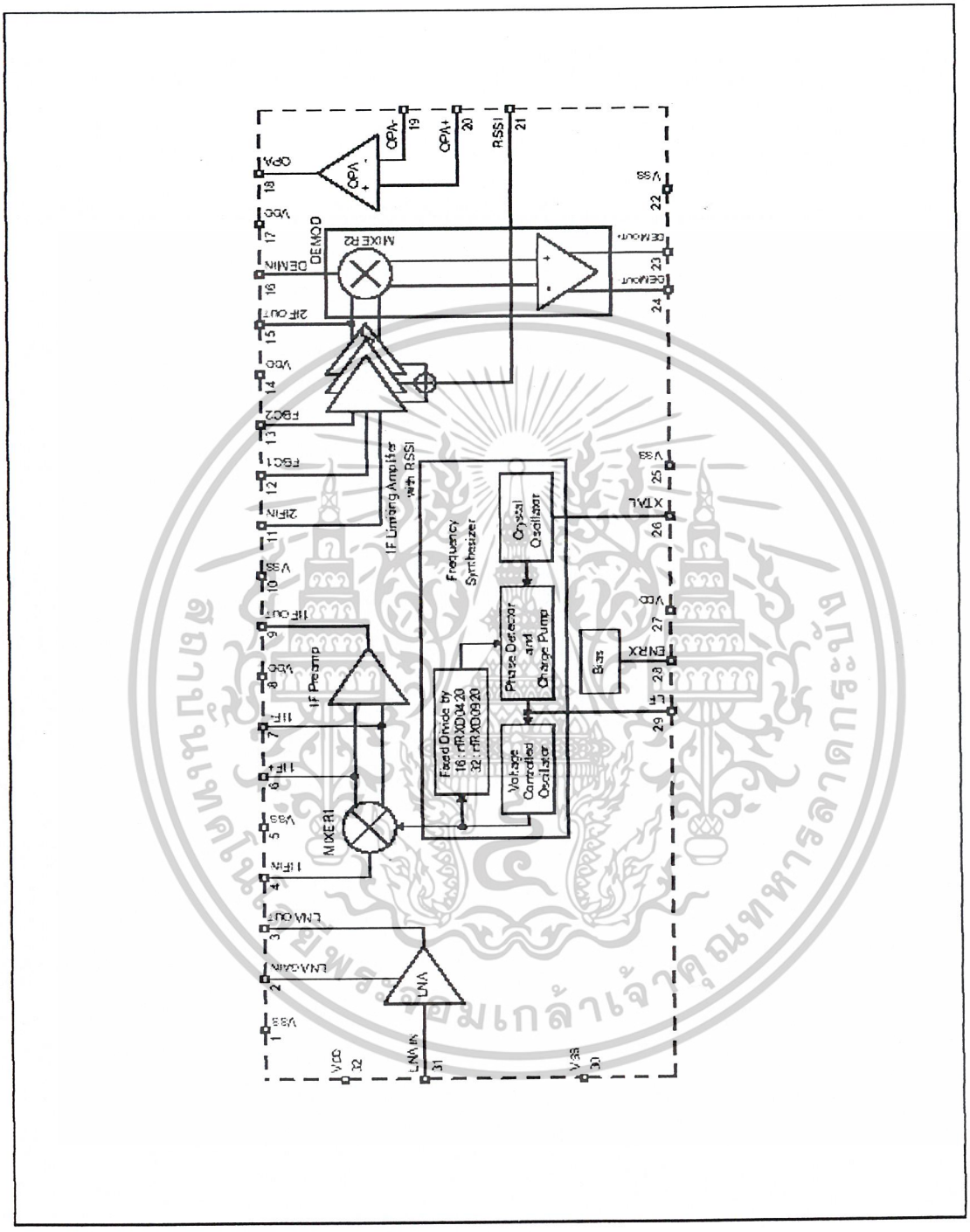
ลักษณะเฉพาะ

- มีขนาดที่เล็กมาก
- มี 32 ขา LQFP
- ใช้งานในย่านความถี่ 300 MHz ถึง 450 MHz
- ความถี่ที่รับถูกกำหนดโดยความถี่ของ crystal
- กำหนดความกว้างของความถี่ IF โดยอุปกรณ์ภายนอกที่เรียกว่า ceramic IF filter
- มีราคาค่อนข้างถูก
- ความถี่ IF อยู่ในย่าน 450 kHz ถึง 21.4 MHz
- ความถี่อยู่ในย่าน ±5 kHz ถึง ±120 kHz
- อัตราข้อมูลสูงสุด
- ASK 80 Kbps NRZ
- FSK 40 Kbps NRZ

การประยุกต์ใช้งาน

- รีโมทควบคุมไร้สาย
- ระบบรักษาความปลอดภัยไร้สาย
- บ้านอัตโนมัติ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 บล็อกไดอะแกรมของ rfRXD0420

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 รายละเอียดขาพอร์ตทั้งหมดของ rfRXD0420 PINOUT I/O

ชื่อขา	ขาที่	ชนิด	ชนิดบัฟเฟอร์	รายละเอียด
LNA_{GAIN}	2	I	ซีมอส	LNA gain ควบคุม
LNA_{OUT}	3	O	อนาล็อก	LNA เอาต์พุต
$1IF_{IN}$	4	I	อนาล็อก	1^{st} IF สเตจอินพุต
1IF+	6	-	อนาล็อก	MIXER1 ไบอัส (open collector)
1IF-	7	-	อนาล็อก	MIXER1 ไบอัส (open collector)
$1IF_{OUT}$	9	O	อนาล็อก	1^{st} IF สเตจเอาต์พุต
$2IF_{IN}$	11	I	อนาล็อก	2^{nd} IF สเตจอินพุต
FB C1	12	-	อนาล็อก	จำกัดIF ตอบกลับภายนอกของ capacitor
FB C2	13	-	อนาล็อก	จำกัด IF ตอบกลับภายนอกของ capacitor
$2IF_{OUT}$	15	O	อนาล็อก	2^{nd} IF สเตจเอาต์พุต
DEM_{IN}	16	I	อนาล็อก	สัญญาณ Demodulation อินพุต
OPA	18	O	อนาล็อก	ขยายสัญญาณเอาต์พุต
OPA+	19	I	อนาล็อก	ขยายสัญญาณอินพุตลบ
OPA-	20	I	อนาล็อก	ขยายสัญญาณอินพุตบวก
RSSI	21	O	อนาล็อก	รับสัญญาณ โดยตรงชี้ไปที่เอาต์พุต
DEM_{OUT+}	23	O	อนาล็อก	สัญญาณ Demodulation เอาต์พุตบวก
DEM_{OUT-}	24	O	อนาล็อก	สัญญาณ Demodulation เอาต์พุตลบ
XTAL	26	I	อนาล็อก	อินพุต crystal oscillator
ENRX	28	I	ซีมอส	อินพุต Receiver enable
LF	29	I	อนาล็อก	ต่อกับ loop filter ภายนอก
LNA_{IN}	30	I	อนาล็อก	อินพุต LNA
V_{DD}	8,14,17,27,32	P		ไฟเลี้ยง +
V_{SS}	1,5,10,25,30	P		กราวด์อ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

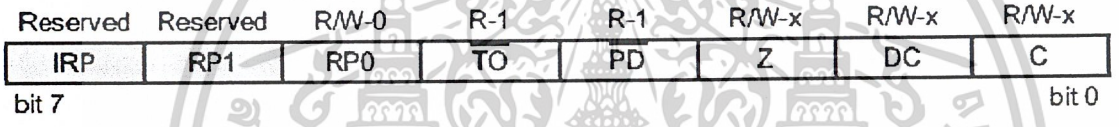
การจัดสรรหน่วยความจำและรีจีสเตอร์ควบคุมของ rfPIC12F675

ใน rfPIC12F675 มีรีจีสเตอร์ควบคุมที่มีบทบาทสำคัญอยู่ 6 ตัว คือ STATUS OPTION INTCON PIE1 PIR1 และ PCON

รีจีสเตอร์ STATUS

เป็นรีจีสเตอร์ที่ใช้แสดงสถานะทางคณิตศาสตร์ของหน่วยประมวลผลทางคณิตศาสตร์ สถานะ การทำงานของ rfPIC12F675 และใช้เป็นตัวกำหนดเลือกแบงค์ของหน่วยความจำข้อมูลการเข้าถึงรีจีสเตอร์ STATUS เพื่ออ่านและเขียนข้อมูล สามารถการกระทำได้ด้วยวิธการเดียวกันกับการอ่านและเขียนรีจีสเตอร์ตัวอื่นๆค่าพารามิเตอร์ในตารางที่ที่ 2.3

ตาราง 2.3 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ STATUS



บิต 7 : IRP ต้องกำหนดให้เป็น “0”

บิต 6 : RP1 ต้องกำหนดให้เป็น “0”

บิต 5 : RP0 บิตเลือกแบงค์ของรีจีสเตอร์

“1” = แบงค์ 1 (แอดเดรส 0x00-0x7F)

“0” = แบงค์ 0 (แอดเดรส 0x80-0xFF)

บิต 4 : TO (Time Out Bit) บิตขอบเขตเวลา บิตนี้สามารถอ่านค่าได้อย่างเดียวเท่านั้น

“1” เมื่อไฟเลี้ยงให้ rfPIC12F675 หรือมีการกระทำคำสั่ง Clrwdt หรือ Sleep

“0” เมื่อวอตซ์ด็อกไทม์เมอร์ภายในการทำงานครบเวลาหรือเกิดไทม์เอาท์

บิต 3 : PD (Power down Bit) บิตเพาเวอร์ดาวน์ ทำงานในโหมดสลีป (Sleep)

“1” เมื่อไฟเลี้ยงให้ rfPIC12F675 หรือมีการกระทำคำสั่ง Clrwdt

“0” เมื่อมีการกระทำคำสั่ง Sleep

บิต 2 : Z (Zero Bit) บิตศูนย์เป็นแสดงผลการกระทำทางคณิตศาสตร์

“1” เมื่อมีการทำคำสั่งทางคณิตศาสตร์และลอจิกแล้ว เกิดค่าศูนย์ (0) ขึ้น

“0” เมื่อมีการทำคำสั่งทางคณิตศาสตร์และลอจิกแล้วไม่ เกิดค่าศูนย์ (0) ขึ้น

บิต 1 : DC (Digit Carry/Borrow Bit) บิตทดหรือบิตระหว่างหลัก เป็นแสดงผลทางคณิตศาสตร์ ในกรณีกระทำคำสั่ง Addwf และ Addlw บิต DC จะเกิดผลดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"1" เมื่อการเกิดทดจากบิตล่างที่ 4 ไปยังกลุ่มบิตบน (จากบิตที่ 4 ไปยังบิตที่ 5)

"0"เมื่อไม่การเกิดทดจากบิตล่างที่ 4 ไปยังกลุ่มบิตบน (จากบิตที่ 4 ไปยังบิตที่ 5)

ในกรณีที่กระทำคำสั่ง Subwf และ Sublw บิต DC จะเกิดผลดังนี้

"1" เมื่อไม่เกิดการยืมจากบิตล่างที่ 4 โดยกลุ่มบิตบน (บิตที่ 5 ยืมค่าจากบิตที่ 4)

"0" เมื่อเกิดการยืมจากบิตล่างที่ 4 โดยกลุ่มบิตบน (บิตที่ 5 ยืมค่าจากบิตที่ 4)

บิต 0 : C (Carry Bit / Borrow Bit) บิตทดหรือบิตยืม ทำงานคล้ายกับบิต DC แต่จะเกิดการเปลี่ยนแปลงค่าก็ต่อเมื่อเกิดการทดหรือยืมค่าจากบิตนัยสำคัญสูงสุด (Most Significant Bit)

ในกรณีกระทำคำสั่ง Addwf และ Addlw บิต C จะเกิดผลดังนี้

"1" เมื่อการกระทำคำสั่งทางคณิตศาสตร์และลอจิกแล้ว เกิดการทด

"0" เมื่อการกระทำคำสั่งทางคณิตศาสตร์และลอจิกแล้ว ไม่เกิดการทด

ในกรณีกระทำคำสั่ง Subwf และ Sublw บิต C จะเกิดผลดังนี้

"1" เมื่อการกระทำคำสั่งทางคณิตศาสตร์และลอจิกแล้ว ไม่เกิดการยืม

"0" เมื่อการกระทำคำสั่งทางคณิตศาสตร์และลอจิกแล้ว เกิดการยืม

รีจีสเตอร์ OPTION

เป็นรีจีสเตอร์ที่ควบคุมการอินเตอร์รัพต์จากสัญญาณภายนอกค่าพารามิเตอร์ในตารางที่ 2.4

ตารางที่ 2.4 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ OPTION

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{GPPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

บิต 7 : GPPU: GPIO Pull-up Enable bit

"1" GPIO ดิสเอเบิลการพลูอัพ

"0" GPIO อีนาเบิลการพลูอัพ

บิต 6 : INTEDG (Interrupt Edge Select Bit) บิตเลือกขอบขาสัญญาณอินเตอร์รัพต์จากภายนอกใช้เลือกขอบขาสัญญาณที่ทำให้เกิดการอินเตอร์รัพต์ที่ขา RB0 / INT

"1" เลือกขอบขาขึ้นของสัญญาณ GP2 / INT

"0" เลือกขอบขาลงของสัญญาณ GP2 / INT

บิต 5 : TOCS (TMRO Clock Source Select Bit) บิตเลือกแหล่งจ่ายสัญญาณนาฬิกาให้แก่ไทม์เมอร์เคาร์เตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"1" เลือกจากการเปลี่ยนแปลงระดับสัญญาณที่ขา GP2 / TOCKI

"0" เลือกจากสัญญาณนาฬิกาที่ใช้กำหนดไซเคิลการทำงานภายใน PIC2F675 (CLKOUT)

บิต 4 : TOSE (TMRO Source Edge Select bit) ใช้เลือกขอบขาของสัญญาณนาฬิกาที่จ่ายเข้ายังขา GP2 / TOCKI ในกรณีที่เลือกแหล่งจ่ายสัญญาณนากาให้ไทม์เมอร์เคาเตอร์ผ่านทางขา GP2 / TOCKI โดยการกำหนดบิต TOCS ให้เป็น "1"

"1" TMROเพิ่มค่าเมื่อเกิดการเปลี่ยนแปลงระดับสัญญาณ จากสูงมาต่ำ

"0" TMOR เพิ่มค่าเมื่อเกิดการเปลี่ยนแปลงระดับสัญญาณ จากต่ำมาสูง

บิต 3 : PSA (Prescaler Assignment Bit) บิตกำหนดการทำงานของปรีสเกลเลอร์

"1" ปรีสเกลเลอร์ทำงานร่วมกับวอตซ์ด็อกไทม์เมอร์

"0" กำหนดให้ปรีสเกลเลอร์ทำงานร่วมกับ TMRO

บิต 2-0 : PS2-PS0 (Prescaler Rate Select Bit) บิตเลือกอัตราส่วนของปรีสเกลเลอร์ เป็นบิที่ใช้ในการกำหนดอัตราส่วนในการทำงานของปรีสเกลเลอร์ เมื่อทำงานกับ TMRO และวอตซ์ด็อกไทม์เมอร์ อัตราส่วนนี้จะมีค่าไม่เท่ากัน

ตารางที่ 2.5 การกำหนดอัตราส่วนของปรีสเกลเลอร์

PS2	PS1	PS0	อัตราส่วนของปรีสเกลเลอร์	
			เมื่อทำงานร่วมกับTMR0	เมื่อทำงานร่วมกับ WDT
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:32	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจีสเตอร์ PCON (Power control status register)

เป็นรีจีสเตอร์ที่ใช้ในการแจ้งสถานะของการรีเซตที่เกิดขึ้นของ rfPIC12F675 มีแอดเดรสอยู่ที่0x8E มีบิตการใช้งาน 2 บิต ค่าพารามิเตอร์ในตารางที่ 2.6 ดังนี้

ตารางที่ 2.6 รายละเอียดของการบิตต่างๆ ในรีจีสเตอร์ PCON

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-x
—	—	—	—	—	—	$\overline{POR}$	$\overline{BOD}$
bit 7							bit 0

บิต 2-7 ไม่ได้ใช้งาน อ่านค่าได้เป็น “0”

บิต 1 POR (Power-on reset status bit) บิตแสดงสถานะการเกิดเพาเวอร์ออนรีเซต

“1” ไม่มีเหตุการณ์เพาเวอร์ออนรีเซตเกิดขึ้น

“0” มีเกตุการณ์เพาเวอร์ออนรีเซตเกิดขึ้น

หลังจากที่เกิดเพาเวอร์ออนรีเซตเกิดขึ้น ต้องเซตบิตนี้ทันทีด้วยกระบวนการทางซอฟต์แวร์

บิต 0 BOR (Brown-out reset status bit) บิตแสดงสถานะการเกิดบราวเอาต์รีเซต

“1” ไม่มีเหตุการณ์ราวเอาต์รีเซตเกิดขึ้น

“0” มีเหตุการณ์ราวเอาต์รีเซตเกิดขึ้น

หลังจากเกิดบราวน์เอาต์รีเซตขึ้น ต้องเซตบิตนี้ทันทีด้วยกระบวนการทางซอฟต์แวร์

รีจีสเตอร์ OSCCAL

เป็นรีจีสเตอร์ที่ใช้สำหรับปรับแต่งค่าความถี่ของวงจรกำเนิดสัญญาณนาฬิกาภายใน 4MHz ของ rfPIC12F675 มีแอดเดรสอยู่ที่ 0x90 ในแบงค์ 1 มีบิตใช้งานทั้งสิ้น 6 บิตเพื่อปรับแต่งค่าความถี่ของสัญญาณนาฬิกาให้มีค่าใกล้เคียง 4 MHz มากที่สุด ค่าพารามิเตอร์แสดงในตารางที่2.7

ตารางที่ 2.7 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ OSCCAL

R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—
bit 7							bit 0

บิต 2-7 CAL5-CAL0 (6-bit Signed oscillator calibration bits) บิตปรับแต่งค่าความถี่สัญญาณนาฬิกาแบบคิดเครื่องหมาย

“111111” ค่าความถีสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

"100000" ค่าความถี่กลาง

"000000" ค่าความถี่ต่ำ

บิต 1 และ 0 : ไม่ได้ใช้งาน อ่านค่าเป็น "0"

รีจีสเตอร์ W (Working register)

เป็นรีจีสเตอร์ที่มีความสำคัญมากที่สุดของไมโครคอนโทรลเลอร์ PIC เพราะไม่ว่าจะเป็นการประมวลผลทางคณิตศาสตร์และลอจิก ต้องกระทำผ่านรีจีสเตอร์ W นี้ทั้งสิ้น หรืออาจกล่าวได้ว่า รีจีสเตอร์คือ แอกคิวมูลเลเตอร์ (Accumulator) ในการเรียกใช้งานรีจีสเตอร์ตัวนี้สามารถกระทำได้ทันทีและตลอดเวลา โดยคำสั่งที่เกี่ยวข้องกับรีจีสเตอร์ W ซึ่งสามารถสังเกตได้จากคำสั่ง หากคำสั่งมีตัวอักษร W เข้าไปเป็นส่วนประกอบของคำสั่งนั้นๆ จะต้องติดต่อกับรีจีสเตอร์ W แน่นอน

โปรแกรมเคาน์เตอร์ (PC)

เป็นรีจีสเตอร์พิเศษที่ทำหน้าที่ระบุแอดเดรสของหน่วยความจำโปรแกรมที่ซีพียูจะต้องไปทำการอ่านข้อมูลเพื่อทำงาน โปรแกรมเคาน์เตอร์มีขนาด 13 บิต สำหรับใน rfPIC12F675 ซึ่งมีขนาดหน่วยความจำโปรแกรม 1 กิโลเวิร์ดจะใช้เพียง 10 บิต

โปรแกรมเคาน์เตอร์แบ่งเป็น 2 ส่วน ส่วนแรกมีขนาด 8 บิต เป็นข้อมูลในไบต์ต่ำ (บิต 0-7) เรียกว่า รีจีสเตอร์ PCL มีแอดเดรสอยู่ที่ 0x02 อีกส่วนหนึ่งมีขนาด 5 บิต เป็นบิตข้อมูลในไบต์สูง (บิต 8-12) เรียกว่า รีจีสเตอร์ PCH สำหรับรีจีสเตอร์ PCH ไม่สามารถเข้าถึงได้โดยตรง การปรับปรุงข้อมูลใน PCH จะต้องกระทำผ่านรีจีสเตอร์ PCLATH ซึ่งมีแอดเดรสอยู่ที่ 0x0A

สแต็ก (Stack)

ใน rfPIC12F675 มีหน่วยความจำสำรองสำหรับเก็บค่าของโปรแกรมเคาน์เตอร์ (PC) ชั่วคราว ขนาด 13 บิตหรือเรียกว่า สแต็ก โดยสามารถเก็บข้อมูลได้ทั้งสิ้น 8 ระดับ โยที่พื้นทีของสแต็กนั้นจะจัดสรรแยกไว้ต่างหากไม่เกี่ยวข้องกับพื้นที่ของหน่วยความจำแต่อย่างใด ตัวชี้ตำแหน่งของสแต็กหรือเรียกว่า สแต็กพอยน์เตอร์ (Stack Pointer) ผู้ใช้งานไม่สามารถอ่านหรือเขียนข้อมูลได้ โดยค่าของสแต็กพอยน์เตอร์จะเปลี่ยนแปลงอัตโนมัติจากการกระทำคำสั่ง CALL , RETURN , RETLW และ RETFIE การเก็บค่าของสแต็กจะต่อเนื่องการเป็นวงกลม สามรถเก็บข้อมูลที่ไม่ซ้ำกันได้ 8 ค่า เมื่อมีการเก็บข้อมูลครั้งที่ 9 เข้ามาข้อมูลนั้นจะไปทับในสแต็กของข้อมูลครั้งแรกเช่นนี้ไปตลอด ดังนั้นในไมโครคอนโทรลเลอร์ PIC จึงไม่มีการแจ้งเหตุการณ์สแต็กเกิน (Stack Overflow)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
62123

นอกจากนั้นการเก็บหรืออ่านค่าในสต็กของไมโครคอนโทรลเลอร์ rfPIC12F675 จะเกิดขึ้นเมื่อมีการกระทำคำสั่ง CALL , RETURN , RETLW และ RETFIE หรือเกิดการอินเตอร์รัพต์เท่านั้น ไม่มีคำสั่ง PUSH หรือ POP เพื่อติดต่อกับสแต็กเหมือนกับไมโครคอนโทรลเลอร์เบอร์อื่นๆ

โครงสร้างพอร์ตของ rfPIC12F675

ไมโครคอนโทรลเลอร์ rfPIC12F675 จะขาพอร์ตสำหรับต่อใช้งานเพียง 6 ขา ทว่าในแต่ละขากลับมีประสิทธิภาพ และมีฟังก์ชันการทำงานที่พิเศษจนข้อด้อยเรื่องจำนวน ไม่ใช่ข้อจำกัดในการใช้งาน rfPIC12F675 ดั้งนั้นการทำความเข้าใจในโครงสร้างของขาพอร์ตแต่ละขาของ rfPIC12F675 จึงเป็นเรื่องที่น่าสนใจยิ่ง รายละเอียดของขาพอร์ตแต่ละขาเรียงตามลำดับตั้งแต่ GP0 จนถึง GP5

สถานะของพอร์ต

เมื่อเริ่มต้นการจ่ายไฟเลี้ยงหรือเกิดเพาเวอร์ออนรีเซต ขาพอร์ตของ rfPIC12F675 จะมีสถานะดังนี้

GP0-GP2 เป็นขาอินพุตรับสัญญาณอนาลอกองวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล

GP3 เป็นขาอินดิจิตอลหรือขารีเซตหลัก MCRLขึ้นอยู่กับการกำหนดที่ Configuration Word

GP4 เป็นขาสัญญาณอนาลอก หรือขาสัญญาณนาฬิกา OSC2 ขึ้นอยู่กับการกำหนด ที่ Configuration Word ในโปรแกรมหรือจากซอฟต์แวร์ในขณะทำการโปรแกรมก็ได้

GP5 เป็นขาอินพุตดิจิตอลหรือขาสัญญาณนาฬิกา OSC1 ขึ้นอยู่กับการกำหนดที่ Configuration Word ในโปรแกรมหรือซอฟต์แวร์ในขณะทำการโปรแกรมก็ได้

รีจีสเตอร์ที่เกี่ยวข้อง

ในการใช้งานขาพอร์ตของ rfPIC12F675 ในแบบดิจิตอลจะมีรีจีสเตอร์สำคัญที่เกี่ยวข้องโดยตรง 4 ตัวคือ GPIO, TRISIO, WPO และ GPIO ใช้บรรจุข้อมูลของพอร์ตแต่ละขาของ rfPIC12F675 ส่วนรีจีสเตอร์ TRISIO ใช้กำหนดลักษณะของขาพอร์ตว่าเป็นอินพุตหรือเอาท์พุต สำหรับรีจีสเตอร์ WPU ใช้กำหนดพลูอัพให้กับขาพอร์ต สุดท้ายรีจีสเตอร์ IOCB ใช้สำหรับเอ็นเอเบิลการอินเตอร์รัพต์อันเนื่องมาจากการเปลี่ยนแปลงลอจิกที่ขาพอร์ตของ rfPIC12F675

รีจีสเตอร์ GPIO และ TRISIO

รีจีสเตอร์ GPIO และ TRISIO มีขนาด 8 บิตแต่ใช้งานจริงเพียง 6 บิต โดย 2 บิตบน ไม่ใช้งาน กำหนดให้เป็น “0” ความสัมพันธ์ของรีจีสเตอร์ทั้งสอง คือ TRISIO ใช้กำหนดว่า ขาพอร์ตใดเป็นอินพุตหรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอาท์พุต โดยบิต 0 (TRISIO0) ใช้กำหนดทิศทางของขาพอร์ต GPO ไล่ตามลำดับจนถึงลำดับ 5 (TRISIO5) โดยกำหนดทิศทางของขาพอร์ต GP5

พอร์ตอินพุต เขียน “1”

พอร์ตเอาท์พุต เขียน “0”

ทุกขาพอร์ตของ rfPIC12F675 สามารถกำหนดให้เป็นอินพุตหรือเอาท์พุต เว้นเฉพาะขา GP5 จะเป็นอินพุตได้เพียงอย่างเดียว

ส่วนรีจีสเตอร์ GPIO จะบรรจุข้อมูลของขาพอร์ตไม่ว่าขณะนั้นขาพอร์ตจะเป็นขาอินพุตหรือเอาท์พุตก็ตาม กล่าวคือ หากเป็นอินพุตสัญญาณข้อมูลที่ rfPIC12F675 อ่านได้ จะถูกเก็บไว้ในรีจีสเตอร์ GPIO ก่อน หรือในกรณีส่งข้อมูลออกไปทางขาพอร์ตเมื่อกำหนดให้เป็นพอร์ตเอาท์พุต ก็ต้องเขียนข้อมูลมายังรีจีสเตอร์ GPIO เช่นกัน

เช่นเดียวกันกับรีจีสเตอร์ TRISIO บิต 0 ของ GPIO (GPIO0) ใช้บรรจุของขาพอร์ต GPO เรียงตามลำดับจนถึง GPIO5 ซึ่งใช้บรรจุข้อมูลของขาพอร์ต GP5

รีจีสเตอร์ WPU (Weak Pull-Up register)

มีแอดเดรสอยู่ที่ 0x95 ในแบงค์ 1 ค่าพารามิเตอร์แสดงในตารางที่2.8

ตารางที่ 2.8รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ WPU

U-0	U-0	R/W-1	R/W-1	U-0	R/W-1	R/W-1	R/W-1
—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0
bit 7							bit 0

บิต 7 : ไม่ใช้งานกำหนดเป็น “0”

บิต 5-4:WPU5-WPU4 (Weak pull-up register bit) บิตเลือกการพลูอัพภายในที่ขาพอร์ต GP4 และ GP5

“1” เอ็นเอเบิล

“0” ดิสเอเบิล

บิต 3 : ไม่ได้ใช้งาน กำหนดเป็น “0”

บิต 2-0 : WPU2-WPU0 (Weak pull up register bit) บิตเลือกการพลูอัพที่ขา GP2-GP0

“1” เอ็นเอเบิล

“0” ดิสเอเบิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสามารถพิเศษอื่นของขาพอร์ตของ rfPIC12F675

การพลูอัพภายใน

ในกรณีที่เลือกให้ขาพอร์ตของ rfPIC12F675 เป็นพอร์ตอินพุตดิจิตอลโดยปกติมักจะมีการต่อตัวต้านทานเพื่อพลูอัพหรือสร้างลอจิก "1" ตั้งต้นเอาไว้ สำหรับขาพอร์ตของ rfPIC12F675 จะมีความสามารถพิเศษคือ สามารถเลือกให้มีการการพลูอัพภายในขาพอร์ตได้อัตโนมัติ ทั้งนี้เพื่อช่วยลดอุปกรณ์ต่อพ่วง ทำให้สามารถลดอุปกรณ์ต่อพ่วงได้ด้วย การเลือกฟังก์ชันพิเศษนี้สามารถกระได้ทุกขาพอร์ตยกเว้น GP3 โดยการเขียนข้อมูลควบคุมไปยังรีจีสเตอร์ WPU (Weak pull-up register) และฟังก์ชั่นพิเศษจะถูกดิสเอเบิลเมื่อเกิดเพาเวอร์ออนรีเซต และเมื่อมีการเปลี่ยนแปลงการทำงานเป็นพอร์ตเอาท์พุต

การเอ็นเอเบิลความสามารถนี้ทำได้โดยเคลียร์บิต GPPU (บิต 7 ของรีจีสเตอร์ OPTION_REG) เพื่อเอ็นบิลการพลูอัพร่วมก่อน จากนั้นจึงเลือกเอ็นเอเบิลขาพอร์ตแต่ละขาผ่านรีจีสเตอร์ WPU

การอินเตอร์รัพต์เนื่องจากการเปลี่ยนแปลงลอจิกที่ขาพอร์ต

ความสามารถพิเศษที่น่าสนใจคือ การอินเตอร์รัพต์จากการเปลี่ยนแปลงลอจิกที่ขาพอร์ต ด้วยความสามารถนี้ช่วยให้การตรวจสอบการเปลี่ยนแปลงลอจิกที่พอร์ตทำได้ง่าย ช่วยลดงานของการวนตรวจสอบสถานะลอจิกของขาพอร์ต เนื่องจากหากเกิดการเปลี่ยนแปลงก็จะเกิดการอินเตอร์รัพต์ได้

ซีพียูจึงค่อยมาสนใจและตอบสนอง ดังนั้น ในภาวะปกติซีพียูจึงสามารถไปทำงานอื่นในโปรแกรมหลักได้คล่องตัว ไม่ต้องคอยตรวจสอบขาพอร์ต

การทำงานของฟังก์ชั่นนี้ใช้รีจีสเตอร์ 2 ตัว ตัวแรกคือ INTION โดยต้องเซตบิต GIE (Global Interrupt enable bit: บิต 7) และ GPIE (PORT Chang Interrupt Enable bit: บิต 3) จากนั้นเลือกขาพอร์ตที่ต้องการผ่านรีจีสเตอร์ตัวที่สอง นั่นคือ IOCB (Interrupt –On-Change GPIO Register)

การอินเตอร์รัพต์ใน rfPIC12F675

ใน rfPIC12F675 สามารถกำเนิดและตอบสนองการเกิดอินเตอร์รัพต์ได้ 7 ตำแหน่ง

- จากไทเมอร์ 3 แหล่ง คือ การเกิดค่าเกินหรือโอเวอร์โฟลว์ในไทเมอร์ 0 ไทเมอร์ 1
- จากการเขียนหน่วยความจำข้อมูลอีอีพรอมเสร็จสมบูรณ์ 1 แหล่ง
- จากการเปลี่ยนแปลงลอจิกที่ขาพอร์ต GPIO 1แหล่ง
- จากวงจรเปรียบเทียบแรงดันอนาลอก 1 แหล่ง
- จากโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล 1แหล่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- จากสัญญาณภายนอกที่ขา GP2 / INT

การกำหนดให้เกิดอินเตอร์รัพต์ขึ้น ต้องเตรียมการ 4 ขั้นตอน คือ

1. เอ็นเอเบิลการอินเตอร์รัพต์รวม โดยเซตบิต GIE ในรีจีสเตอร์ INTCON

2. เอ็นเอเบิลแหล่งกำเนิดอินเตอร์รัพต์

3. เขียนโปรแกรมบริการอินเตอร์รัพต์ ซึ่งมีค่าแอดเดรสอินเตอร์รัพต์เวกเตอร์อยู่ที่ 0x0004

4. รอคอยให้เงื่อนไขของการเกิดอินเตอร์รัพต์ในลักษณะต่างๆ เกิดขึ้น

รีจีสเตอร์ที่เกี่ยวข้องกับการอินเตอร์รัพต์

1.INTCON เป็นรีจีสเตอร์ควบคุมการอินเตอร์เป็นหลัก

2. PIE1 เป็นรีจีสเตอร์เอ็นเอเบิลแหล่งกำเนิดอินเตอร์รัพต์จากอุปกรณ์ต่อพ่วง

3. PIR1 เป็นรีจีสเตอร์แสดงสถานะเงื่อนไขของแหล่งกำเนิดอินเตอร์รัพต์ จากอุปกรณ์ต่อพ่วงทำงานร่วมกับ PIE1

รีจีสเตอร์ INTCON

เป็นรีจีสเตอร์ที่มีนัยสำคัญสูงสุดสำหรับการอินเตอร์รัพต์มีแอดเดรสอยู่ที่ 0x0B ในแบงค์ 0 และ 0x8Bแบงค์ 1 ใช้เอ็นเอเบิลการอินเตอร์รัพต์รวมของไมโครคอนโทรลเลอร์ ค่าพารามิเตอร์แสดงในตารางที่ 2.9

ตารางที่ 2.9 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ INTCON

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF
bit 7							bit 0

บิต 7 : GIE (Global Interrupt enable bit) บิตอีนาเบิลการอินเตอร์รัพต์

"1" เลือกไม่ให้มีการอินเตอร์รัพต์เกิดขึ้นทั้งหมด

"0"เลือกให้มีการอินเตอร์รัพต์เกิดขึ้นทั้งหมด

บิต 6 : PEIE (Peripheral Interrupt Enable bit)

"1" เอ็นเอเบิลอินเตอร์รัพต์ให้แก่ไทเมอร์ 1 วงจรเปรียบเทียบแรงดัน โมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอลและหน่วยความจำอีอีพรอม

"0" ดีสเบิลอินเตอร์รัพต์แบบนี้

บิต 5 : TOIE (TMRO Overflow Interrupt Enable bit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"1" เอ็นเอเบิลการอินเตอร์รัพต์

"0" ดีสเบิลการอินเตอร์รัพต์

บิต 4 : INTE (GP2 / INT External Interrupt Enable bit)

"1" เอ็นเอเบิลการอินเตอร์รัพต์

"0" ดีสเอเบิลการอินเตอร์รัพต์

บิต 3 : GPIE (Port Change Interrupt Enable bit)

"1" เอ็นเอเบิลการอินเตอร์รัพ

"0" ดีสเอเบิลการอินเตอร์รัพ

บิต 2 : TOIF (TMRO Overflow Interrupt Flag)

"1" แจ้งว่าเกิดการอินเตอร์รัพต์อันเนื่องมาจาก TMRO เกิดการ โอเวอร์โฟลว์

"0" ไม่เกิดการอินเตอร์รัพต์อันเนื่องมาจาก TMRO เกิดการ โอเวอร์โฟลว์

บิต 1 : INTF (GP2 / INT External Interrupt Flag bit)

"1" แจ้งว่าเกิดการอินเตอร์รัพต์อันเนื่องมาจากสัญญาณภายนอกที่ขา GP2 / INT

"0" ไม่เกิดการอินเตอร์รัพต์อันเนื่องมาจากสัญญาณภายนอกที่ขา GP2 / INT

บิต 0 : GPIF (Port Change Interrupt Flag bit)

"1" มีการเปลี่ยนแปลงลอจิกที่เกิดขึ้นที่ขาพอร์ต ทำให้เกิดการอินเตอร์รัพต์ ต้องเคลียร์ทางซอฟต์แวร์

"0" ไม่ มีการเปลี่ยนแปลงลอจิกที่เกิดขึ้นที่ขาพอร์ต RB4-RB7

รีจีสเตอร์ PIE1

ใช้เอ็นเอเบิลแหล่งกำเนิดสัญญาณอินเตอร์รัพต์จากอุปกรณ์ต่อพ่วงในแหล่งต่างๆ มีแอดเดรสอยู่ที่ 0x8C ในแบงค์ 1 ค่าพารามิเตอร์แสดงในตารางที่ 2.10

ตารางที่ 2.10รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ PIE1

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
EEIE	ADIE	—	—	CMIE	—	—	TMR1IE
bit 7							bit 0

บิต 7 : EEIE (EEPROM Write Operation Interrupt Enable bit) บิตเลือกการเกิดอินเตอร์รัพต์เมื่อการเขียนข้อมูลลงหน่วยความจำข้อมูลอีอีพรอมเสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"1" เอ็นเอเบิลการอินเตอร์รัพต์

"0" ดิสเอเบิลการอินเตอร์รัพต์

บิต 6 ADIE (A/D Converter Interrupt Enable bit) บิตเลือกการเกิดอินเตอร์รัพต์จากมูลแบลงสัญญาณอนาลอกเป็นดิจิตอล

"1" เอ็นเอเบิลการอินเตอร์รัพต์

"0" ดิสเอเบิลการอินเตอร์รัพต์

บิต 5-4 : ไม่มีการใช้งาน อ่านได้เป็น "0"

บิต 3 : CMIE (Comparator Interrupt Enable bit) บิตเลือกอินเตอร์รัพต์จากวงจรเปรียบเทียบแรงดัน

"1" เอ็นเอเบิลการอินเตอร์รัพต์

"0" ดิสเอเบิลการอินเตอร์รัพต์

บิต 2-1 : ไม่มีการใช้งาน อ่านได้เป็น "0"

บิต 0 : TMR1IE (TMR1 Overflow interrupt Enable bit) บิตเลือกการเกิดอินเตอร์รัพต์จากการโอเวอร์โฟลว์ของไทเมอร์ 1

"1" เอ็นเอเบิลการอินเตอร์รัพต์

"0" ดิสเอเบิลการอินเตอร์รัพต์

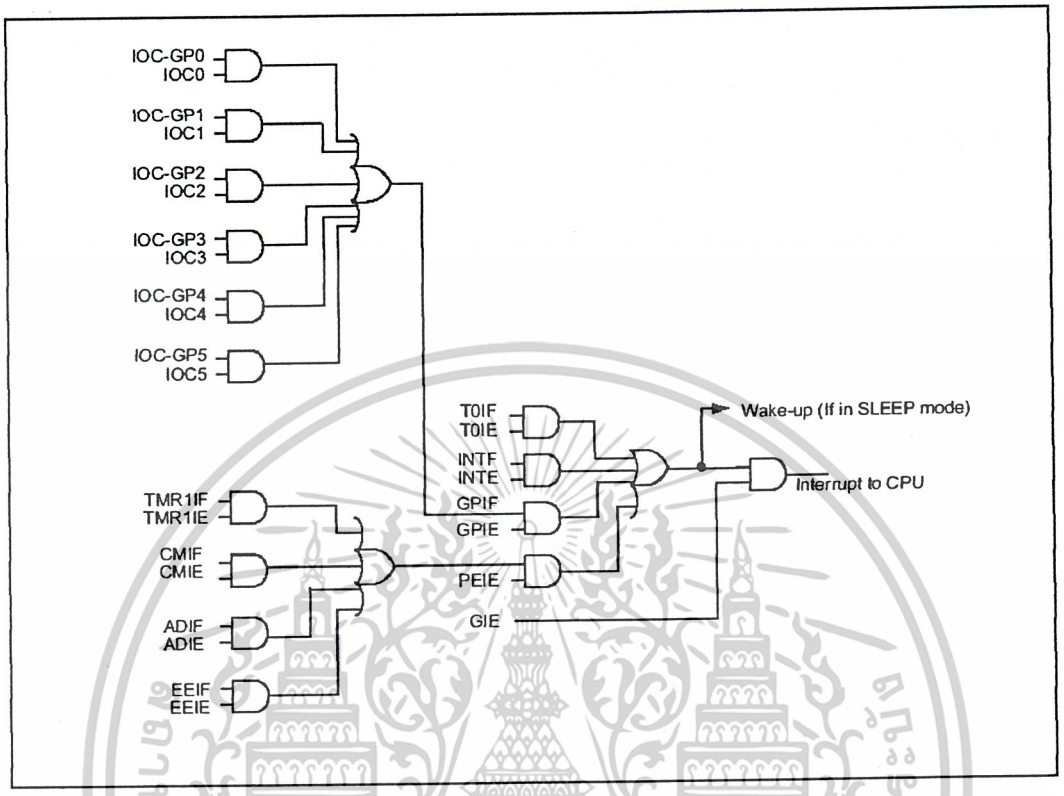
การอินเตอร์รัพต์ลอจิก (Interrupt logic)

อินเตอร์รัพต์ลอจิกแบ่งออกเป็น 3 ส่วน คือแหล่งกำเนิดสัญญาณอินเตอร์รัพต์จากอุปกรณ์ต่อพ่วง 7 แหล่ง ส่วนที่ 2 คือ ส่วนควบคุมการอินเตอร์รัพต์จากอุปกรณ์ต่อพ่วงและแหล่งกำเนิดสัญญาณอินเตอร์พื้นฐานอีก 3 แหล่ง และส่วนที่ 3 คือ ส่วนควบคุมการอินเตอร์รัพต์รวมซึ่งควบคุมโดยบิต GIE และ INTCON ดังแสดงในรูปที่ 2.8

การบริการอินเตอร์รัพต์

เมื่อเกิดการอินเตอร์รัพต์ ซีพียูจะเก็บค่าแอดเดรส ที่จะต้องกลับมา หลังจากการบริการอินเตอร์รัพต์ ซึ่งก็คือค่าของโปรแกรมเคาน์เตอร์ (PC) นั่นเองลงในสแต็ก และ เคลียร์บิต GIE ดิสเอเบิลเกิดอินเตอร์รัพต์ทั้งหมด ทั้งนี้เพื่อป้องกันการเกิดอินเตอร์รัพต์ซ้อน ในขณะที่ยังทำการอินเตอร์ไม่เสร็จ จากนั้นกระโดดไปยังอินเตอร์รัพต์เวกเตอร์ หรือแอดเดรสเริ่มต้นของการบริการอินเตอร์รัพต์ กำหนดไว้ที่แอดเดรส 0x004 ดังนั่นในการเขียนที่มีการบริการอินเตอร์รัพต์ ต้องทำสำร้องที่แอดเดรส 0x004 ไว้เพื่อบริการแอดเดรสด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 ไดอะแกรมลอจิกของการเกิดสัญญาณอินเตอร์รัพต์

การเก็บข้อมูลเมื่อเกิดการอินเตอร์รัพต์

ในโปรแกรมประกอบด้วยส่วนสำคัญส่วนหนึ่งคือ ส่วนเก็บข้อมูลของรีจีสเตอร์ เนื่องจากเมื่อเกิดการอินเตอร์รัพต์ มีค่าของโปรแกรมเคาน์เตอร์เท่านั้นที่ได้รับการเก็บไว้ในสแต็ก แต่ค่ารีจิสเตอร์ตัวอื่น ๆ จะติดตามไปยังโปรแกรมย่อยบริการอินเตอร์รัพต์ด้วยการทำงานดังนั้นก่อนที่จะทำงานใด ๆ ในโปรแกรมย่อยอินเตอร์รัพต์ ต้องทำการสำรองขอมูลเดิมของรีจีสเตอร์ที่ต้องมีการใช้งานโปรแกรมย่อยบริการอินเตอร์รัพต์เสียก่อน รวมถึงบิตสถานะหรือแฟลต่าง ๆ ในรีจีสเตอร์ STATUS ด้วย และหลังจากการทำงานในโปรแกรมบริการอินเตอร์รัพต์เรียบร้อย ก่อนออกจากโปรแกรมต้องทำการเรียกขอมูลเดิมของรีจีสเตอร์ที่เกี่ยวข้องทุกจากหน่วยความจำชั่วคราว รวมถึงค่าบิตสถานะของรีจิสเตอร์ STATUS ด้วย

การติดต่อหน่วยความจำข้อมูลอีอีพรอมใน rfPIC12F675

รายละเอียดของหน่วยความจำข้อมูลแบบอีอีพรอม (EEPROM data memory) ภายใน PIC12F6975 ซึ่งมีความจุ 128 ไบต์ หน่วยความจำส่วนนี้สามารถอ่าน และ เขียนได้ในขณะทำงานปกติภายใต้ย่านไฟเลี้ยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปกติ แต่ ไม่สามารถที่จะเข้าถึงเพื่ออ่านและเขียนได้โดยตรง ต้องกระทำผ่านรีจีสเตอร์ฟังก์ชั่นพิเศษด้วยวิธีการเข้าถึงได้โดยอ้อม (Indirect addressing)

รีจีสเตอร์ที่ใช้ในการอ่านและเขียนข้อมูลกับหน่วยความจำข้อมูลอีอีพรอมมีทั้งสิ้น 4 ตัว คือ

1. EECON1 รีจีสเตอร์ควบคุมเข้าหน่วยความจำ
2. EECON2 รีจีสเตอร์จัดลำดับการเขียนข้อมูลในหน่วยความจำ
3. EEDATA รีจีสเตอร์บัฟเฟอร์ข้อมูลสำหรับการอ่านและเขียน
4. EEADR รีจีสเตอร์แอดเดรส

ในการติดต่อหน่วยความจำข้อมูลอีอีพรอมจะเลือกแอดเดรสที่ต้องการติดต่อผ่านทางรีจีสเตอร์ EEADR ซึ่งมีแอดเดรสอยู่ที่ 0x9B โดยรีจีสเตอร์ EEADR ต้องการข้อมูลเพื่อเลือกแอดเดรสเพียง 7 บิต (บิต 7 ไม่ใช่กำหนดเป็น “0”) จากนั้นใช้รีจีสเตอร์ EECON1 และ EECON2 เพื่อกำหนดจุดประสงค์ในการติดต่อกับหน่วยความจำต้องการอ่านหรือเขียน โดยข้อมูลที่ทำการอ่านหรือเขียนจะบรรจุอยู่ในรีจีสเตอร์ EEDATA หน่วยความจำข้อมูลอีอีพรอมนี้สามารถทำการลบและเขียนใหม่ได้ถึง 10 ล้านรอบ และข้อมูลไม่สูญหายแม้ไม่มีการจ่ายไฟเลี้ยง

รีจีสเตอร์ EECON1 และ EECON2

รีจีสเตอร์ EECON1 เป็นรีจีสเตอร์ควบคุมการเข้าถึงหน่วยความจำข้อมูลอีอีพรอม ในขณะที่ EECON2 เป็นรีจีสเตอร์พิเศษที่ช่วยในการช่วยเขียนข้อมูลลงในหน่วยความจำข้อมูลอีอีพรอม สำหรับ EECON2 นี้ไม่ใช่รีจีสเตอร์ปกติที่สามารถเข้าถึงได้โดยตรง การเข้าถึงจะมีขั้นขั้นตอนและรูปแบบที่กำหนดไว้เป็นพิเศษ การเรียกใช้หรือการเข้าถึงจะเกิดขึ้นเฉพาะในตอนของการเขียนข้อมูลลงในหน่วยความจำข้อมูลอีอีพรอมเท่านั้น ในกรณีอื่นจะอ่านค่าในรีจีสเตอร์EECON2 เป็น “0”ค่าพารามิเตอร์แสดงงในตารางที่ 2.11

ตารางที่ 2.11 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ EECON1

U-0	U-0	U-0	U-0	R/W-x	R/W-0	R/S-0	R/S-0
—	—	—	—	WRERR	WREN	WR	RD
bit 7							bit 0

บิต 4-7 : ไม่ใช้งาน กำหนดเป็น “0”

บิต 3 WRERR (EEPROM error flag bit) บิตแจ้งความผิดพลาดในการเขียนข้อมูลลงในหน่วยความจำข้อมูลอีอีพรอม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"1" การเขียนข้อมูลเกิดความผิดพลาดหรือยกเลิกกะทันหัน

เกิดสัญญาณรีเซตภายนอกที่ขา MCLR

เกิดการรีเซตเนื่องจากวอร์ดอกไทเมอร์ในขณะทำงานปกติ

เกิดการรีเซตจากวงจรบราวเอาต์ดีเท็กหรือบราวเอาต์รีเซต

"0" การเขียนข้อมูลเสร็จสมบูรณ์

บิต 2 WREN (EEPROM Write Enable bit) บิตเลือกการเขียนข้อมูลหน่วยความจำอีอีพรอม

"1" เลือกให้เกิดกระบวนการเขียนข้อมูลได้

"0" เลือกไม่ให้เกิดการเขียนข้อมูลได้

บิต 1 : WR (Write control bit) บิตควบคุมการเขียนข้อมูล

บิตนี้ใช้สำหรับเริ่มต้นกระบวนการเขียนข้อมูลลงในหน่วยความจำ ข้อมูลอีอีพรอม โดยเมื่อต้องการให้เริ่มการเขียนข้อมูล ต้องเซตบิตนี้ก่อนเสมอ และ ไม่สามารถเคลียร์บิตนี้ทางซอฟต์แวร์ได้บิตนี้จะเคลียร์เป็น "0" อย่างอัตโนมัติ เมื่อกระบวนการเขียนข้อมูลเสร็จสิ้นอย่างสมบูรณ์

บิต 0: RD (Read control bit) บิตควบคุมการอ่าน

บิตนี้ใช้สำหรับเริ่มต้นกระบวนการอ่านข้อมูลจากหน่วยความจำข้อมูลอีอีพรอม โดยเมื่อต้องการให้เริ่มอ่านข้อมูล ต้องทำการเซตบิตนี้ก่อนเสมอ จะไม่สามารถเคลียร์บิตนี้ทางซอฟต์แวร์ได้ บิตนี้จะเคลียร์เป็น "0" อย่างอัตโนมัติ เมื่อกระบวนการเขียนข้อมูลเสร็จสิ้น

การอ่านหน่วยความจำข้อมูลอีอีพรอม

เริ่มด้วยการกำหนดแอดเดรสที่ต้องการอ่าน ข้อมูลลงใน รีจีสเตอร์ EEADR แล้ว เซตบิต RD ในรีจีสเตอร์ EECON1 ข้อมูลจากหน่วยความจำจะได้รับการอ่านออกมาเก็บไว้ในรีจีสเตอร์ EEDATA ในไซเกิลขอคำสั่งถัดไป และรักษาข้อมูลไว้จนกว่าจะเกิดการอ่านหรือเขียนข้อมูลใหม่ขึ้น

การเขียนข้อมูลลงในหน่วยความจำข้อมูลอีอีพรอม

เริ่มต้นการเขียนข้อมูลด้วยการกำหนดแอดเดรส ของหน่วยความจำอีอีพรอม ที่ต้องการไปยังรีจีสสเตอร์ EEADR ต่อด้วยการเขียนข้อมูลที่ต้องการไปยังรีจีสเตอร์ EEDATA เซตบิต WREN เพื่อเอ็นเอเบิลการเขียนข้อมูล และดิสเอเบิลการอินเตอร์รัพต์ทุกรูปแบบขั้นตอนต่อไปเป็นขั้นตอนที่สำคัญมาก เป็นขั้นตอนการเขียนข้อมูล 0x55 และ 0xAA ไปยังEECON2 เพื่อเริ่มต้นกระบวนการเขียนข้อมูลลงในหน่วยความจำอีอีพรอม โดยขั้นตอนพิเศษนี้มีลำดับของโปรแกรมดังนี้

MOVLW 0x55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOVWF EECON2

MOVLW 0xAA

MOVWF EECON2

BSF EECON1, WR

ในทุกครั้งที่มีการเขียนข้อมูลลงในหน่วยความจำอีอีพรอม ต้องเขียนโปรแกรมทั้ง 5 บรรทัด นี้ โดยเรียงลำดับตามขั้นตอนทุกประการ หลังจากที่กระบวนการเขียนเสร็จสิ้นลง ให้ทำการเอ็นเอเบิลตอบสนองอินเตอร์รัพต์ แล้วเข้าสู่โหมดสลีป เพื่อรอการอินเตอร์รัพต์อันเนื่องมาจากการเขียนข้อมูลลงหน่วยความจำอีอีพรอมได้อย่างสมบูรณ์ นั่นคือเกิดการเคลียร์บิต WR อันการแจ้งความสมบูรณ์ของการเขียนข้อให้ทราบและเซตบิต EEIF ซึ่งเป็นแฟลก แจ้งการอินเตอร์รัพต์อันเองมาจากการเขียนข้อมูลที่เสร็จสมบูรณ์ ขั้นตอนสุดท้ายคือ ดิสเอเบิลการเขียนข้อมูลโดยการเคลียร์บิต WREN ในรีจีสเตอร์ EECON1และเคลียร์แฟลก EEIF ก่อนออกจากกระบวนการเขียนข้อมูล

การป้องกันข้อมูลในหน่วยความจำข้อมูลอีอีพรอม

สามารถกระทำได้จากเครื่องโปรแกรมภายนอกเท่านั้น โดยเลือกฟังก์ชั่น EEPROM protection หรืออาจมีชื่ออื่นที่แตกต่างออกไป และการป้องกันนี้จะป้องกันเฉพาะการอ่านโดยใช้เครื่องโปรแกรมภายนอกเท่านั้น โดยระหว่างที่การทำงานปกติของไมโครคอนโทรลเลอร์ ซีพียูจะสามารถเข้าถึงหน่วยความจำข้อมูลอีอีพรอมได้ตามปกติ หรืออาจกล่าวได้ว่าเป็นการป้องกันการคัดลอก โดยใช้เครื่องอ่านหรือเครื่องโปรแกรมภายนอกนั่นเอง

ไทเมอร์เคาน์เตอร์ในไมโครคอนโทรลเลอร์ rfPIC12F675

มี 2 ตัว คือ ไทเมอร์ 0 (Timer0) และ ไทเมอร์ 1 (Timer 1)

ไทเมอร์ 0 (Timer 0) สามารถทำงานเป็นไทเมอร์และเคาน์เตอร์ ขนาด 8 บิต มีปรีสเกลเลอร์ขนาด 8 บิต รวมอยู่ด้วย ประกอบด้วยรีจีสเตอร์หลักขนาด 8 บิต 1 ตัว คือ รีจีสเตอร์ TMR0 มีแอดเดรสอยู่ที่ 0x01 ข้อมูลที่เกิดจากการทำงานของไทเมอร์ 0 ทั้งหมดจะได้รับการเขียนและเก็บไว้ที่รีจีสเตอร์ TMR0 นี้ และเมื่อต้องการอ่านค่าของไทเมอร์ 0 ก็ต้องมาอ่านนที่รีจีสเตอร์ TMR0

การรับสัญญาณเพื่อกระต้นให้ไทเมอร์ 0 ทำงาน มาจาก 2 แหล่งกำเนิด คือ แหล่งกำเนิดภายในจากสัญญาณนาฬิกาภายในซึ่งมีความถี่เท่ากับ ความถี่สัญญาณนาฬิกาหลักหาร 4 และจากสัญญาณกระตุ้นภายนอกที่ผ่านเข้ามาทางขา GP2/T0CKI จากการรับสัญญาณกระต้นจากแหล่งกำเนิดสัญญาณทั้งสองนี้เองจึงเป็นสิ่งที่ใช้แบ่งแยกให้ทราบว่าไทเมอร์ 0 ทำงานเป็นไทเมอร์ หรือ เคาน์เตอร์ หากรับสัญญาณนาฬิกาจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายใน แสดงว่าทำงานเป็นไทเมอร์ แต่ถ้ารับสัญญาณจากภายนอกแสดงว่าทำงานเป็นเคาน์เตอร์ และเมื่อการเพิ่มค่าของไทเมอร์ 0 มาถึงจุดสูงสุด คือ 0xFF แล้วเปลี่ยนเป็น 0x00 จะเกิดการ โอเวอร์โฟลว์ขึ้น พร้อมกับเกิดอินเตอร์รัพต์ในกรณีที่มีการเอ็นเอเบิลการอินเตอร์รัพต์แบบนี้ไว้

การทำงานเบื้องต้นของไทเมอร์ 0

โหมดการทำงานของไทเมอร์ 0 สามารถกำหนดได้ที่บิต T0CS ซึ่งเป็นบิตที่ 5 ในรีจีสเตอร์ OPTION_REG วิ่งก็คือ การเลือกรับสัญญาณอินพุตเพื่อกระตุ้นการทำงานของไทเมอร์ 0 โดย

ทำงานในโหมดไทเมอร์ เมื่อ T0CS = “0” และ

ทำงานในโหมดเคาน์เตอร์ เมื่อ T0CS= “1”

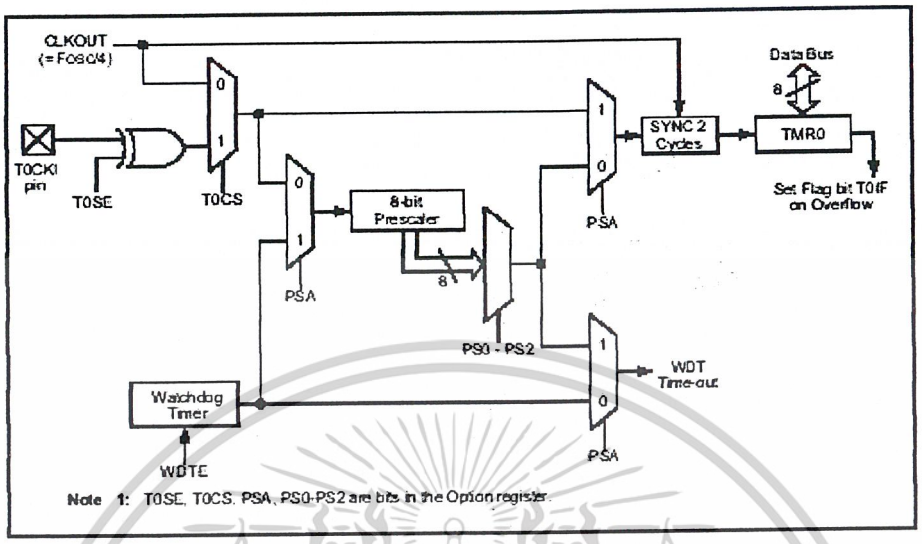
เมื่อทำงานในโหมดไทเมอร์ ค่า TMR0 จะเพิ่มขึ้นทุก ๆ ไซเคิลการทำงาน หากมีการเขียนข้อมูลลงในรีจีสเตอร์ TMR0 การเพิ่มค่าของไทเมอร์ 0 จะหยุดชะงักชั่วครู่ และกลับมาเพิ่มเป็นปกติหลังจากนั้น 2 ไซเคิลการทำงาน ดั้งนั้นในการข้อมูลของไทเมอร์ 0 มาต้องใช้ความระมัดระวัง หากมีการเขียนข้อมูลไปยัง TMR0 ต้องรออีกอย่างน้อย 2 ไซเคิลการทำงาน ค่าของ TMR0 จึงจะได้รับการปรับปรุงให้ถูกต้อง

เมื่อทำงานในโหมดเคาน์เตอร์ ต้องเลือกว่า จะให้ไทเมอร์ 0 ทำงานที่ขอบขาใดของสัญญาณถูกป้อนเข้ามา โดยเลือกบิตที่ T0SE เป็น “0” ค่าของไทเมอร์ 0 จะเพิ่มขึ้นเมื่อพบขอบขาขึ้นของสัญญาณอินพุต ในทางตรงกันข้ามหากบิต T0SE เป็น “1” จะเพิ่มค่าขึ้นเมื่อพบขอบขาลงของสัญญาณที่ขา GP2/T0CKI

การเพิ่มค่าของตัวนับในไทเมอร์ 0 ยังถูกควบคุมโดยปรีสเกลเลอร์ด้วย โดยปรีสเกลเลอร์ทำหน้าที่เป็นเหมือนตัวหารของไทเมอร์ 0 โดยที่ปรีสเกลเลอร์นี้สามารถกำหนดอัตราการหาร ได้โดยกำหนดที่บิต PS2-PS0 ในรีจีสเตอร์ OPTION_REG โดยสามารถเลือกได้ว่าต้องการให้ปรีสเกลเลอร์ทำงานกับไทเมอร์ 0 หรือไม่ ด้วยการกำหนดที่บิต PSA ในรีจีสเตอร์ OPTION_REG ทั้งนี้เนื่องจากรีสเกลเลอร์นี้มีการใช้งานร่วมกันระหว่างไทเมอร์ 0 กับวอตช์ด็อกไทเมอร์ หลังจากนั้นสัญญาณจะผ่านเข้าไปยังส่วนจัดการสัญญาณเพื่อให้การเพิ่มค่าของไทเมอร์ 0 สัมพันธ์กับสัญญาณนาฬิกาภายใน หลังจากเขียนค่าลงในรีจีสเตอร์ TMR0

ไทเมอร์ 0 จะให้ผลทางเอาต์พุต 2 ลักษณะ คือ อ่านข้อมูลที่รีจีสเตอร์ TMR0 ที่แอดเดรส 0x01 และกำเนิดสัญญาณอินเตอร์รัพต์เมื่อเกิดโอเวอร์โฟลว์ จากการเปลี่ยนแปลงข้อมูลที่ 0xFF เป็น 0x00 ดังแสดงในรูปที่2.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 ไดอะแกรมการทำงานของไทเมอร์ 0

รีจีสเตอร์ OPTION

เป็นรีจีสเตอร์ที่ควบคุมการอินเตอร์รัพต์จากสัญญาณภายนอก ค่าพารามิเตอร์แสดงในตารางที่ 2.12

ตารางที่ 2.12 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ OPTION

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

บิต 7 : GPPU: GPIO Pull-up Enable bit

"1" GPIO ดิสเอเบิลการพลูอัพ

"0" GPIO อีนาเบิลการพลูอัพ

บิต 6 : INTEDG (Interrupt Edge Select Bit) บิเลือกขอบขาสัญญาณอินเตอร์รัพต์จากภายนอกใช้เลือกขอบขาสัญญาณที่ทำให้เกิดการอินเตอร์รัพต์ที่ขา RB0 / INT

"1" เลือกขอบขาขึ้นของสัญญาณ GP2 / INT

"0" เลือกขอบขาลงของสัญญาณ GP2 / INT

บิต 5 :TOCS (TMRO Clock Source Select Bit) บิตเลือกแหล่งจ่ายสัญญาณนาฬิกาให้แก่ไทม์เมอร์เคาร์เตอร์

"1" เลือกจากการเปลี่ยนแปลงระดับสัญญาณที่ขา GP2 / TOCKI

"0" เลือกจากสัญญาณนาฬิกาที่ใช้กำหนดไซเคิลการทำงานภายใน PIC2F675 (CLKOUT)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต 4 : TOSE (TMRO Source Edge Select bit) ใช้เลือกขอบขาของสัญญาณนาฬิกาที่จ่ายเข้ายังขา GP2 / TOCKI ในกรณีที่เลือกแหล่งจ่ายสัญญาณนากาให้ไทม์เมอร์เคาร์เตอร์ผ่านทางขา GP2 / TOCKI โดยการกำหนดบิต TOCS ให้เป็น “1”

“1” TMROเพิ่มค่าเมื่อเกิดการเปลี่ยนแปลงระดับสัญญาณ จากสูงมาต่ำ

“0” TMOR เพิ่มค่าเมื่อเกิดการเปลี่ยนแปลงระดับสัญญาณ จากต่ำมาสูง

บิต 3 : PSA (Prescaler Assignment Bit) บิตกำหนดการทำงานของปรีสเกลเลอร์

“1” ปรีสเกลเลอร์ทำงานร่วมกับวอตซ์ด็อกไทม์เมอร์

“0” กำหนดให้ปรีสเกลเลอร์ทำงานร่วมกับ TMROบิต 2-0: PS2-PS0 (Prescaler Rate Select Bit) บิตเลือกอัตราส่วนของปรีสเกลเลอร์ เป็นบิที่ใช้ในการกำหนดอัตราส่วนในการทำงานของปรีสเกลเลอร์ เมื่อทำงานกับ TMRO และวอตซ์ด็อกไทม์เมอร์ อัตราส่วนนี้จะมีค่าไม่เท่ากัน

ปรีสเกลเลอร์

เป็นตัวนับขนาด 8 บิตบิต ทำงาน 2 ลักษณะ คือ เป็นปรีสเกลเลอร์หรือตัวหาร- ตัวลดทอนสำหรับโมดูลไทเมอร์ 0 และเป็นโพสต์สเกลเลอร์ (Postscaler) หรือตัวเพิ่มเวลาสำหรับวอตซ์ด็อกไทเมอร์ ไม่สามารถอ่านและขียนค่ากับปรีสเกลได้ แต่สามารถเข้าโหมดและกำหนดอัตราการทำงานของปรีสเกลเลอร์ได้ โดยกำหนดข้อมูลที่บิต PSA สำหรับเลือกปรีสเกลเลอร์ทำงานกับไทเมอร์ 0 หรือวอตซ์ไทเมอร์ และบิต PS2-PS0 เพื่อกำหนดอัตราการทำงาน โดยเลือกได้ 8 ค่า ขึ้นอยู่กับว่ากำหนดปรีสเกลเลอร์ทำงานกับไทเมอร์ 0 หรือวอตซ์ด็อกไทเมอร์ ดังแสดงอัตราการทำงานของปรีสเกลในตารางที่ 2.5

การใช้ไทเมอร์ 0 กับสัญญาณนาฬิกาภายนอก

เมื่อต้องการให้ไทเมอร์ 0 ทำงานด้วยสัญญาณนาฬิกาภายนอก สิ่งที่จะต้องใคามสำคัญมากคือจังหวะเวลาของสัญญาณจะต้องสอดคล้องกับจังหวะการทำงานภายใน rfPIC12F675 หรือเกิดการซิงโครไนเซชั่น เมื่อมีการป้อนสัญญาณนาฬิกาจากภายนอกให้แก่ไทเมอร์ 0 ผ่านทางขา GP2/T0CKI และไม่มีการใช้ปรีสเกลเลอร์ภายใน rfPIC12F675 การซิงโครไนเซชั่นระหว่างสัญญาณนาฬิกาจากภายนอกที่ขา GP2/T0CKI กับสัญญาณนาฬิกาภายในที่จะเกิดขึ้นจากการสุ่มที่ไซเคิล Q2 และ Q4 ของสัญญาณภายใน ดั้งนั้นที่ขา GP2/T0CKI จะต้องมีคาบเวลามากกว่าสัญญาณนาฬิกาภายใน หรือTose อย่างน้อย 2 เท่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.13 อัตราการทำงานของปรีสเกลเลอร์เมื่อทำงานกับไทม์เมอร์ 0 และวอตซ์ด็อกไทม์เมอร์

PS2	PS1	PS0	อัตราส่วนของปรีสเกลเลอร์	
			เมื่อทำงานร่วมกับTMR0	เมื่อทำงานร่วมกับ WDT
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:32	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

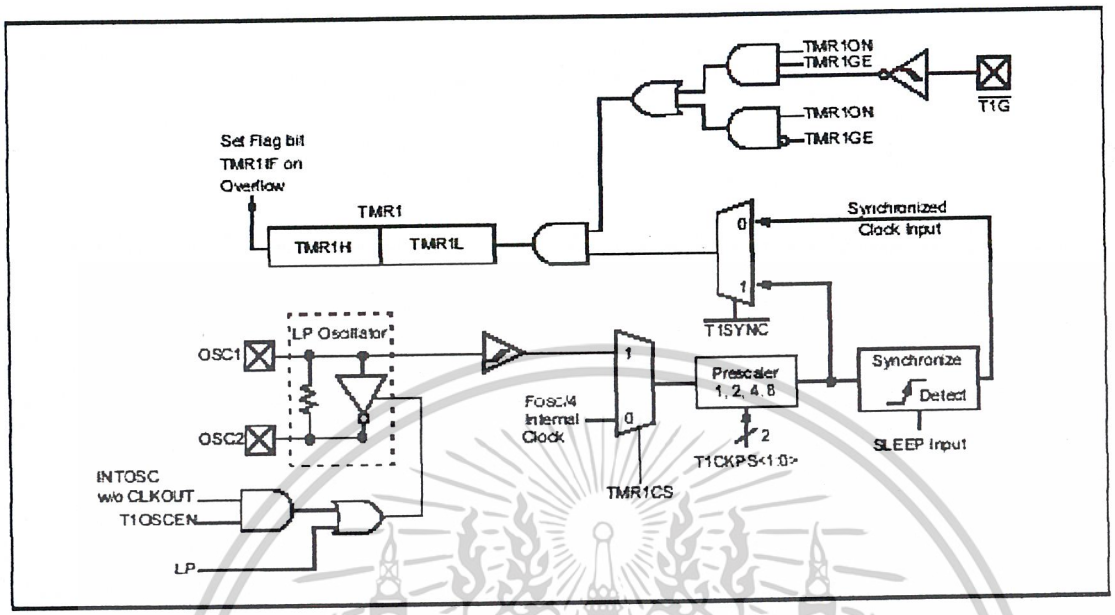
ไทเมอร์ 1

ไทเมอร์ 1 (Timer 1) มีขนาด 16 บิตพร้อมปรีสเกลเลอร์ กำหนดให้ทำงานเป็น 2 ลักษณะสามารถเลือกการทำงานให้เป็นไทเมอร์หรือเคาน์เตอร์ขนาด 16 บิต ประกอบด้วยไทเมอร์/เคาน์เตอร์ขนาด 8 บิตต่อร่วมกัน รีจิสเตอร์ที่เกี่ยวข้องกับไทเมอร์ 1 มี 2 ตัว คือ TMR1 และ T1CON โดย TMR1 เป็รีจิสเตอร์เก็บข้อมูลของไทเมอร์/เคาน์เตอร์ มีขนาด 16 บิต แบ่งเป็นรีจิสเตอร์ขนาด 8 บิต ได้อีก 2 ตัว คือ TMR1H ใช้ในการเก็บค่าบิตที่สูงหรือ 8 บิตบน และ TMR1L ใช้ในกาของไทเมอร์ 1 จึงเกิดขึ้นได้ตั้งแต่ 0x0000 - 0xFFFF และเมื่อค่าของไทเมอร์จาก0xFFFF วนกลับมาเป็น0x0000 ก็จะเกิดโอเวอร์โฟลว์ บิต TMR1F ในรีจิสเตอร์ PIR1 (บิต 0) เซต ทำให้เกิดการอินเตอร์รัพต์ขึ้นหากเอ็นเอเบิลการอินเตอร์รัพต์ด้วยไทเมอร์ 1 ซึ่งกำหนดไว้ที่บิต TMR1E ในรีจิสเตอร์ PIE

จุดแตกต่างจากไทเมอร์ 0 คือ ไทเมอร์ 1เมื่อได้รับการเอ็นเอเบิลให้ทำงานในโหมดเคาน์เตอร์แล้วจะทำงานอย่างต่อเนื่อง แม้ว่าไมโครคอนโทรลเลอร์จะทำงานในโหมดสลีบก็ตาม และไทเมอร์ 1 สามารถกำหนดได้ว่าต้องการให้มีการซิงโครไนซ์ ระหว่างสัญญาณกระตุ้นจากภายนอก กับสัญญาณนาฬิกาภายในหรือไม่ก็ได้ ในกรณีใช้สัญญาณกระตุ้นจากภายนอกรีจิสเตอร์ที่ใช้ควบคุมการทำงานของไทเมอร์ 1 คือ T1CON

นอกจากนี้สามารถควบคุมการทำงานของไทเมอร์ 1 ผ่านางฮาร์ดแวร์ได้ โดยการควบคุมที่ขา GP4/$\overline{T1G}$ โดยหกมีการเอ็นเอเบิลการทำงานที่ขานี้ โดยกำหนดที่รีจิสเตอร์ T1CON ไทเมอร์ 1 จะสามารถทำงานได้ก็ต่อเมื่อต่อขา GP4/$\overline{T1G}$ นี้ลงกราวด์หรือป้อนลอจิกเป็น “0” มาที่ขานี้ ดังแสดงในรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 ไดอะแกรมการทำงานของไทเมอร์ 1

การทำงานของไทเมอร์ 1

การทำงานโหมดไทเมอร์

เลือกให้ทำงานโดยการเคลียร์บิต TMR1CS สัญญาณใช้จากภายใน มีความถี่เท่ากับ 1 / 4 เท่าของความถี่หลักของสัญญาณนาฬิกาหรือ Fosc/4 ซึ่งเท่ากับความถี่ของสัญญาณนาฬิกาภายในของไมโครคอนโทรล เลอร์นั่นอง โดยสถานะที่บิต $\overline{T1SYNC}$ จะไม่มีผลต่อการทำงานในโหมดนี้

การทำงานในโหมดเคาน์เตอร์แบบซิงโครนัส

ไทเมอร์ 1 จะทำงานในโหมดนี้เมื่อเซตบิต TMR1CS ค่าของตัวนับในไทเมอร์ 1 จะเพิ่มขึ้นเมื่อขึ้นเมื่อพบสัญญาณขอบขาขึ้นของสัญญาณกระตุ้นจากภายนอกที่ส่งมายังขาอินพุตของไทเมอร์ 1 ซึ่งมีด้วยกัน 2 ขา คือ RC0/T1OSO/T1CKI และ RC1/T1OSI ขึ้นอยู่กับการกำหนดที่บิต T1OSCEN หากเป็น “0” หมายถึงให้รับสัญญาณที่ขา RC0/T1OSO/T1CKI หากเป็น “1” ให้รับสัญญาณอินพุต RC1/T1OSI

หากต้องการให้สัญญาณนิกาจากภายนอกที่ส่งเข้ามานับค่าไทเมอร์สัมพันธ์ หรือซิงโครไนซ์กับสัญญาณนาฬิกาภายนอกของไมโครคอนโทรลเลอร์ต้องทำการเคลียร์บิต T1SYNC และจะทำการปรีสเกลเลอร์ไม่ต้องทำงานร่วมในโหมดนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อกำหนดให้ไทเมอร์ 1 ทำงานในโหมดเคาน์เตอร์แบบซิงโครนัส หากไมโครคอนโทรลเลอร์ ต้องทำงานในโหมดสลีบ การทำงานของไทเมอร์ 1 จะหยุดลงชั่วขณะ คือ ไม่มีการเพิ่มค่าการนับขึ้น แม้ว่าจะยังคงมีสัญญาณภายนอกเข้ามาอย่างต่อเนื่องก็ตาม แต่ถ้าไม่เลือกการซิงโครไนซ์ไว้ เมื่อในโหมด สลีบ ปรีสเกลจะยังคงเพิ่มค่าต่อไป

แม้ว่าจะเลือกการซิงโครไนซ์เกิดขึ้นแล้วก็ตาม ย่อมต้องมีช่วงเวลาหน่วงเกิดขึ้น หลังจากการซิงโครไนซ์ เมื่อปรีสเกลเลอร์ถูกกำหนดให้อัตรา 1:1 นั่นหมายความว่าเอาต์พุตของปรีสเกลเลอร์จะเหมือนกับสัญญาณอินพุตที่เข้ามา การซิงโครไนซ์จะเกิดขึ้นโดยการสุ่มที่ช่วงเวลา Q2 และ Q4 ของไซเคิลการทำงานของไมโครคอนโทรลเลอร์ จำเป็นอย่างยิ่งที่สถานะของสัญญาณที่อินพุต T1CKI ต้องเป็นลอจิก "1" หรือ "0" ปรากฏเป็นสัญญาณขอบขาขึ้นหรือลง อย่างน้อย $2T_{OSC}$ รวมกับค่าหน่วงเวลาเนื่องจากตัวต้านทานและตัวเก็บประจุในอีกประมาณ 20 นาโนวินาที ในกรณีที่อัตราส่วนของปรีสเกลที่ไม่ใช่ 1:1 สัญญาณอินพุตที่เข้ามาขานั้นต้องรักษาสถานะเดิมไว้เป็นระยะที่เพิ่มขึ้นเป็น 4Tosc รวมกับค่าหน่วงเวลาเนื่องจากตัวต้านทานและตัวเก็บประจุภายในอีกประมาณ 40 วินาที

การทำงานในโหมดเคาน์เตอร์แบบอซิงโครนัส

การทำงานในโหมดนี้จะเกิดขึ้นเมื่อเซตบิต T1SYNC ส่งผลให้สัญญาณกระตุ้นจากภายนอกไม่มีการซิงโครไนซ์กับสัญญาณนาฬิกาในไมโครคอนโทรลเลอร์ค่าของไทเมอร์ 1 จะเพิ่มขึ้นโดยไม่สัมพันธ์กับนาฬิกาภายใน นั่นคือทำงานเป็นตัวนับอิสระขนาด 16 บิต ข้อเด่นของการทำงานในโหมดนี้คือ ไทเมอร์ 1 ยังคงทำงานต่อเนื่องแม้ว่าไมโครคอนโทรลเลอร์ จะเข้าสู่โหมดสลีบหรือโหมดประหยัดพลังงานแล้วก็ตาม ทั้งนี้ยังสามารถอินเตอร์รัพต์เกิดขึ้นได้ในกรณีที่เอ็นเอเบิลการอินเตอร์รัพต์ไว้ เมื่อเกิดการอินเตอร์รัพต์ขึ้นจะทำให้ไมโครคอนโทรลเลอร์เกิดการเวกอัพ (Weak-up) กลับสู่การทำงานในโหมดปกติได้

การอ่านค่าการนับของไทเมอร์ 1 ซึ่งเก็บอยู่ในรีจีสเตอร์ TMR1H และ TMR1L สามารถกระทำได้ตลอดเวลาที่ไทเมอร์ 1 ทำงาน ควรสำรองค่าของการนับ 16 บิตที่อ่านไว้ในรีจีสเตอรีไฟล์ขนาด 8 บิต 2 ตัว เพื่อป้องกันความผิดพลาดในกรณีที่เกิดการโอเวอร์โฟลว์ขึ้นขณะทำหารอ่านข้อมูลจากไทเมอร์ 1

ในกรณีที่ต้องการเขียนข้อมูลไปยังไทเมอร์ 1 ควรหยุดการทำงานของไทเมอร์เสียก่อน ไม่เช่นนั้นจะเกิดการชนกันของข้อมูลทำให้ไม่ทราบได้เลยว่า ขณะนี้ค่าของไทเมอร์ 1 ที่แท้จริงเป็นเท่าใดส่งผลทำให้เกิดความผิดพลาดไทเมอร์ 1 ออสซิลเลเตอร์ในไทเมอร์ 1 ได้บรรจุวงจรกำเนิดสัญญาณหรือ ออสซิลเลเตอร์ไว้ภายในด้วย โดยการบรรจุอยู่ที่ระหว่างอินพุต T1OSI และเอาต์พุต T1OSO สามารถเอ็นเอเบิลการทำงานได้โดยการเซต T1OSCEN วงจรออสซิลเลเตอร์ เป็นแบบพลังงานต่ำกำเนิดสัญญาณความถี่ได้สูงสุด 200 kHz และสามารถทำงานอย่างต่อเนื่องแม้ว่าไมโครคอนโทรลเลอร์จะทำงานในโหมดสลีบก็ตามนั่นคือหาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการใช้งานวงจรออสซิลเลเตอร์ควรใช้คริสตอลพลังงานต่ำ ที่มีความถี่ไม่เกิน 200 kHz เพื่อให้สัญญาณที่ออกวงจรออสซิลเลเตอร์มีความเที่ยงตรงจึงควรเลือกใช้ตัวเก็บประจุที่ต่อร่วมกับคริศตอลให้เหมาะสม ดังแสดงในตารางที่ 2.14

ตารางที่ 2.14 การเลือกค่าของตัวเก็บประจุที่มีความถี่สัญญาณนาฬิกาต่างๆ ของไทเมอร์ 1

ความถี่	C1	C2
32kHz	33pF	33pF
100kHz	15pF	15pF
200kHz	15pF	15pF

การรีเซตของรีจีสเตอร์ในไทเมอร์ 1

รีจีสเตอร์ข้อมูลทั้ง 2 ตัวของไทเมอร์ 1 คือ TMR1H และ TMR1L จะเกิดการรีเซตก็ต่อเมื่อ

1. เกิดเพาเวอร์-ออนรีเซต (รีเซตเมื่อเกิดการจ่ายไฟเลี้ยงใหม่)
2. บราวเอาต์รีเซต (รีเซตเนื่องจากแรงดันไฟเลี้ยงลดต่ำกว่าที่กำหนด)

หลังเกิดการรีเซตไทเมอร์ 1 ถูกปิด และค่าของปรีสเกลในไทเมอร์ 1 กลายเป็น 1:1

วอตช์ด็อกไทเมอร์ (Watchdog timer: WDT)

เป็นไทเมอร์ที่บรรจุวงจรกำเนิดสัญญาณอย่างต่อเนื่องหรือ ฟรีรันนิ่งออสซิลเลเตอร์ (Free Running Oscillator) แบบ RC อยู่ภายใน ในการทำงานจึงไม่ต้องวงจรภายนอก วงจรออสซิลเลเตอร์ของวอตช์ด็อกไทเมอร์นี้แยกการทำงานออกจากวงจรออสซิลเลเตอร์ที่ขา OSC1/CLKIN ดังนั้นวอตช์ ด็อกไทเมอร์ ยังคงทำงานต่อไปแม้ว่าไม่มีการป้อนสัญญาณนาฬิกาเข้ามาที่ขา OSC1/CLKIN และ OSC2/CLKOUT ก็ตาม โดยเมื่อเอ็นเอเบิลให้ทำงานแล้ววอตช์ด็อกไทเมอร์จะทำงานเท่าที่ยังมีการจ่ายไฟเลี้ยง ไมโครคอนโทรลเลอร์

คาบเวลาของวอตช์ด็อกไทเมอร์

วอตช์ด็อกไทเมอร์มีคาบเวลาไทม์เอาต์เท่ากับ 18 มิลลิวินาที ในกรณีที่ไม่มีการใชปรีสเกลเลอร์ คาบเวลานีอาจคลาดเคลื่อนได้ตามอุณหภูมิใช้งานและค่าแรงดันไฟเลี้ยง ถ้าต้องการให้คาบเวลาของวอตช์ไทเมอร์ยาวนานขึ้น จะต้องนำปรีสเกลเลอร์มาช่วยซึ่งสามารถกำหนดได้ด้วย กระบวนการซอฟต์แวร์ โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดค่าลงในรีจีสเตอร์ OPTION_REG เมื่อนำปรีสเกลเลอร์มาช่วยคาบเวลาจะเพิ่มสูงถึง 128 เท่า หรือประมาณ 2.3 วินาที

คำสั่ง CLRWDT และ SLEEP เป็นคำสั่งที่ใช้เคลียร์ค่าของวอตช์ด็อกไทเมอร์ เมื่อกระทำ 2 คำสั่งนี้ ค่าของวอตช์ด็อกไทเมอร์จะเป็น 0 โดยไม่เกิดไทม์เอาต์และกำเนิดสัญญาณเพื่อรีเซตระบบนอกจากนั้น $\overline{TO}$ ในรีจีสเตอร์ STATUS จะถูกเคลียร์เมื่อวอตช์ไทเมอร์เกิดการไทม์เอาต์

โมดูลแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

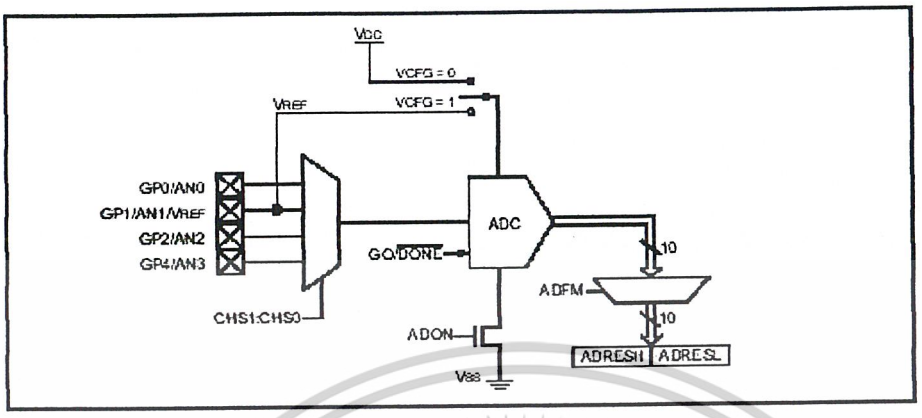
โมดูลแปลงสัญญาณแปลงสัญญาณอนาลอกเป็นดิจิตอลความละเอียด 10 บิต จำนวน 4 ช่อง โดยขาพอร์ตที่ใช้งานร่วมด้วยคือ GP0, GP1, GP2 และ GP4 การทำงานเพื่อรองรับอินพุตจำนวนมากจะใช้วีมัลติเพล็กซ์ ซึ่งควบคุมด้วยกระบวนการทางซอฟต์แวร์ สำหรับการทำงานของโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอลใน rfPIC12F675 เป็นแบบซักเซสซีฟแอปพร็อกซิเมชั่น

การทำงานของโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล

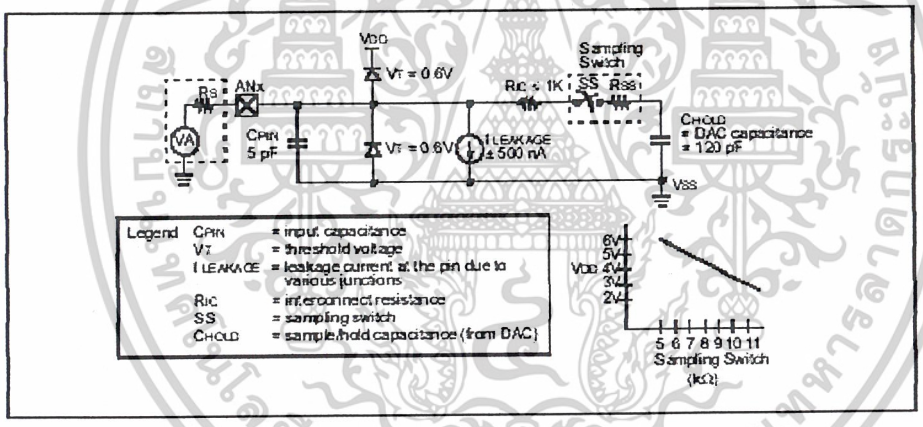
การทำงานขาพอร์ตที่ใช้เป็นอนาลอกต้องได้รับการกำหนดมาจากบิต ANS3: ANS0 ในรีจีสเตอร์ ANSEL ว่าจะให้ขาพอร์ตใดรับสัญญาณอนาลอก หรือพอร์ตใดเป็นพอร์ตดิจิตอล จากนั้นสัญญาณอนาลอกจะถูกส่งเข้ามาในวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอลผ่านวงจรเลือกช่องสัญญาณซึ่งควบคุมด้วยบิต CHS1 และ CHS0 ในรีจีสเตอร์ ADCON0

ภาคอินพุตของวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอลภายใน rfPIC12F675 ประกอบด้วย วงจรสุ่มและเก็บสัญญาณ (Sample and Hold) ที่มีตัวเก็บประจุ 25 pf ทำหน้าที่เก็บแรงดันทางด้านอินพุตไว้ให้กับวงจรแปลงดิจิตอลเป็นอนาลอกและวงจรเปรียบเทียบภายใน หรือ C_{HOLD} ซึ่งในการแปลงนั้นจะต้องรอให้ C_{HOLD} ประจุแรงดันอินพุตเต็มก่อนสัญญาณจึงจะทำให้ผลลัพธ์ที่ได้มีความถูกต้องแม่นยำ นอกจาก C_{HOLD} แล้วยังมีตัวต้านทานอีก 3 ตัว ที่มีผลต่อเวลาที่ใช้ในการเก็บประจุ นั่นคือ ความต้านทานของแหล่งจ่ายสัญญาณภายนอก หรือ R_S ค่าความต้านทานภายในของวงจรแปลงสัญญาณภายอนาลอกเป็นดิจิตอล หรือ R_{IC} และค่าความต้านทานของสวิตช์สุ่มสัญญาณหรือ R_{SS}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 ไดอะแกรมการทำงานของโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล ใน rfPIC12F675



รูปที่2.12 วงจรสมมูลย์อินพุตอนาลอกของโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอลใน rfPIC12F675

ค่าเวลาแอกควิซิชั่น (Acquisition time: TACQ)

เป็นค่าเวลารวมที่ใช้จัดการสัญญาณอนาลอกก่อนเข้าสู่การเปลี่ยนแปลงสัญญาณคำนวณได้จาก

$$T_{ACQ} = T_{AMP} + T_C + T_{COFF}$$

โดยที่

T_{AMP} คือ ค่าเวลาของการขยายและการปรับสัญญาณ (Amplifier Setting Time) มีหน่วยเป็นวินาที ปกติค่าเท่ากับ 2 ไมโครวินาที

T_C คือ เวลาในการประจุแรงดันของตัวเก็บประจุ C_{HOLD} มีหน่วยเป็นวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

T_{COFF} คือ ค่าเวลาสัมประสิทธิ์ทางอุณหภูมิของโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอลซึ่งคำนวณได้จาก

ค่าอุณหภูมิที่ใช้งาน – 25 องศาเซลเซียส x ค่าสัมประสิทธิ์ทางอุณหภูมิ (Sec/C)

หมายเหตุ เนื่องจากอุณหภูมิใช้งานปกติมีค่าเท่ากับ 25 องศาเซลเซลเซียส ทำให้ค่าเวลา T_{COFF} มีค่าเท่า 0 หรือน้อยมากๆ

การเลือกความถี่สัญญาณนาฬิกาของการแปลงสัญญาณ

การเลือกความถี่สัญญาณของสัญญาณนาฬิกาของการเปลี่ยนแปลงนี้สำคัญมาก เนื่องมีผลต่อค่าเวลารวมในการแปลงสัญญาณ โดยในโมดูลแปลงสัญญาณอนาลอกของ rfPIC12F675 ต้องใช้เวลาแปลงสัญญาณอนาลอกเป็นข้อมูลดิจิตอล 10 บิต เท่ากับ $12T_{AD}$ โดยที่ค่าเวลา T_{AD} คือค่าเวลาที่ใช้ในการแปลงสัญญาณต่อบิต และค่า T_{AD} ที่น้อยที่สุดยังคงให้มีผลการเปลี่ยนแปลงสัญญาณที่ถูกต้อง คือ 1.6 ไมโครวินาที ดังนั้น การเลือกความถี่สัญญาณนาฬิกาต้องคำนึงถึงข้อมูลนี้ การเลือกความถี่ทำได้โดยกำหนดข้อมูลในบิต ADCS2 : ADCS ของรีจีสเตอร์ ANSEL จะยังเป็นค่าเดิมก่อนหน้านี้

กระบวนการแปลงสัญญาณอนาลอกเป็นดิจิตอล

เมื่อสั่งให้โมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอลทำงาน บิต GO/$\overline{DONE}$ ที่ใช้แสดงสถานะการทำงานเซต หากมีการเคลียร์บิตที่เกิดขึ้นในขณะที่ทำการแปลงสัญญาณยังดำเนินการอยู่ การแปลงสัญญาณจะถูกหยุดลงทันที แต่ข้อมูลของการแปลงสัญญาณที่ไม่สมบูรณ์จะไม่ได้รับการนำไปใช้งานนั่นคือ หากแปลงไม่สมบูรณ์ค่าในรีจีสเตอร์ ADRESH และADRESL จะยังเป็นค่าเดิมก่อนหน้านี้

การใช้งานโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล

ขั้นตอนการทำงานมีดังนี้ คือ

1. กำหนดค่าพารามิเตอร์เบื้องต้นต่างๆ
    - กำหนดให้ขาพอร์ตที่ใช้งานเป็นอินพุต โดยเซตบิตในรีจีสเตอร์ TRISIO
    - เลือกจำนวนช่องผ่านบิต ANS3 : ANS0 ในรีจีสเตอร์ ANSEL และแหล่งจ่ายแรงดันอ้างอิงผ่านบิต VCFG ในรีจีสเตอร์ ADCCON0
    - เลือกรูปแบบผลลัพธ์จากบิต ADFM ในรีจีสเตอร์ ADCON0
    - เลือกช่องสัญญาณผ่านบิต CJS1 และ CHS0 ในรีจีสเตอร์ ADCON0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เลือกสัญญาณนาฬิกาจากข้อมูลที่บิต ADCS2 : ADCS0 ในรีจีสเตอร์ ANSEL
- เอ็นเอเบิลโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอลโดยการเซตADONในรีจีสเตอร์ ADCON0

2. การใช้งานอินเตอร์รัพต์ให้กำหนดค่าดังต่อไปนี้
   - เคลียร์บิต ADIF ในรีจีสเตอร์ PIR1
   - เซตบิต ADIE ในรีจีสเตอร์ PIE1
   - เอ็นเอเบิลการอินเตอร์รัพต์โดยการเซตบิต GIE ในรีจีสเตอร์ INTCON
3. หน่วงเวลาเพื่อให้พ้นช่วงเวลาแอกควิซิชั่น (Acquisition time)
4. เริ่มต้นการแปลงสัญญาณด้วยการเซตบิต GO/ $\overline{DONE}$ ในรีจีสเตอร์ ADCON0
5. หน่วงเวลาเพื่อให้จบกระบวนการแปลงสัญญาณ ซึ่งทำได้โดยวนค่ารอที่บิต GO/ $\overline{DONE}$

ตารางที่2.15 ความสัมพันธ์ของคาบเวลาสัญญาณนาฬิกาของโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล กับความถี่สูงสุดของสัญญาณนาฬิกาหลักของ rfPIC12F675

เงื่อนไขในการทำงานของโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล		ความถี่สัญญาณนาฬิกาหลัก			
การทำงาน	ADCS2 : ACDS0	20 MHZ	5 MHz	4 MHz	1.25 MHz
2 Tosc	000	100 ns	400ns	500ns	1.6 µs
4Tosc	100	200 ns	800 ns	1 µs	3.2 µs
8Tosc	001	400 ns	1.6 µs	2 µs	6.4 µs
16Tosc	101	800 ns	3.2 µs	4 µs	12.8 µs
32Tosc	010	1.6 µs	6.4 µs	8 µs	25.6 µs
64Tosc	110	3.2 µs	12.8 µs	16 µs	51.2 µs
RC	X11	2-6 µs	2-6µs	2-6 µs	2-6 µs

โมดูลเปรียบเทียบแรงดันอนาลอกและโมดูลสร้างแรงดันอ้างอิง rfPIC12F675

rfPIC12F675 ส่วนจัดการอนาลอกอีก 1 โมดูล นอกเหนือไปจากโมดูลแปลงสัญญาณอนาลอกดิจิตอล นั้นคือ โมดูลเปรียบเทียบแรงดันอนาลอก (Analog Comparator Module) และ โมดูลสร้างดันอ้างอิง (Voltage Reference Module) ในโมดูลเดียวกันนี้มีใน PIC12F629 ซึ่งเป็นไมโครคอนโทรลเลอร์ในอนุกรมเดียวกัน rfPIC12F675 ด้วยเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมดูลเปรียบเทียบแรงดันอนาลอก

ใน rfPIC12F675 โมดูลเปรียบเทียบอนาลอก 1 ชุด โดยอินพุตของวงจรเปรียบเทียบนั้นจะใช้ร่วมกับขาพอร์ต GP0/CIN+ และ GP1/CIN- ส่วนเอาท์พุตของโมดูลนี้ คือ ขา GP2/COUT สำหรับแรงดันอ้างอิงที่ใช้ในการเปรียบเทียบแรงดันได้มาจาก 2 ทาง คือ จากภายในผ่านโมดูลสร้างแรงดันอ้างอิง และจากภายนอกผ่านทางขา GP1/CIN-/V_{REF} โมดูลเปรียบเทียบแรงดันอนาลอกนี้สามารถเลือกโหมดการทำงานได้ 8 แบบ อินพุตของโมดูลเปรียบเทียบแรงดันอนาลอกสามารถรับแรงดันอนาลอกได้โดยตรงได้ตั้งแต่ V_{SS} (คือกราวนด์) ถึง V_{DD} (คือไฟเลี้ยง) เมื่อแรงดันที่อินพุต Vin+ มากกว่า Vin- เอาต์พุตวงจรเปรียบเทียบจะเป็นลอจิกสูง ในทางตรงข้ามหากเกิดแรงดันที่อินพุต Vin+ น้อยกว่า Vin- เอาต์พุตของวงจรเปรียบจะเป็นลอจิกต่ำ ในกรณีที่ทำงานในโหมดไม่กลับลอจิก

สำหรับเวลาในการทำงานของโมดูลเปรียบเทียบแรงอนาลอกใน rfPIC12F675 อยู่ระหว่าง 150 นาโนวินาทีถึง 10 ไมโครวินาที ขึ้นอยู่กับการเลือกแหล่งจ่ายแรงดันอ้าอิง หากใช้แรงดันอ้างอิงจากภายนอกวงจรเปรียบเทียบจะใช้เวลา 150-600 นาโนวินาที แต่ถ้าหากใช้แรงดันอ้างอิงจากภายในจะใช้เวลาสูงสุด 10 ไมโครวินาที

รีจีสเตอร์ CMCON ควบคุมการทำงานของโมดูลเปรียบเทียบแรงดันอนาลอก

ใช้กำหนดและควบคุมการทำงานของโมดูลเปรียบเทียบแรงดันอนาลอกมีแอดเดรสอยู่ที่ตำแหนง 0x19 ในแบงค์ 0 ค่าพารามิเตอร์แสดงในตารางที่ 2.16

ตารางที่ 2.16 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ EECON

U-0	R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	COUT	—	CINV	CIS	CM2	CM1	CM0
bit 7							bit 0

บิต 7 : ไม่ได้ใช้งาน อ่านค่าเป็น “0”

บิต 6 : COUT (Comparator Output bit) บิตแสดงผลเปรียบเทียบของโมดูลเปรียบเทียบแรงดัน ซึ่งสัมพันธ์กับการกำหนดสถานะที่บิต C1INV โดย

ถ้าบิต CINV เป็น “0”

COUT เป็น “0” เมื่อแรงดันที่อินพุต CIN+ น้อยกว่า CIN-

COUT เป็น “1” เมื่อแรงดันที่อินพุต CIN+ มากกว่า CIN-

ถ้าบิต CINV เป็น “1”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COUT เป็น “0” เมื่อแรงดันที่อินพุต CIN+ มากกว่า CIN-

COUT เป็น “1” เมื่อแรงดันที่อินพุต CIN+ น้อยกว่า CIN

บิต 5 : ไม่ได้ใช้งาน อ่านค่าเป็น “0”

บิต 4 : CINV (Comparator Output Inversion bit) บิตกลับสภาวะเอาต์พุตของโมดูลเปรียบเทียบแรงดัน ใช้สำหรับกำหนดการแสดงผลเอาต์พุตของโมดูลเปรียบเทียบแรงดันว่าจะใช้แสดงผลในทางตรงข้ามหรือไม่

“1” เลือกให้การแสดงผลเอาต์พุตมีการกลับสถานะ

“0” เลือกให้การแสดงผลเอาต์พุตไม่มีการกลับสถานะ

บิต 3 : CIS (Comparator input switch bit) บิตเลือกการทำงานให้แก่โมดูลเปรียบเทียบแรงดันเมื่อใช้งานโมดูลเปรียบเทียบแรงดันในโหมด “101” หรือ “110”

“1” กำหนดให้ขาอินพุต VIN- ต่อกับขา GP0/CIN+

“0” กำหนดให้ขาอินพุต VIN- ต่อกับขา GP0/CIN-

บิต 2-0 : CM2-CM0 (Comparator mode bit) บิตเลือกโหมดการทำงานของโมดูลเปรียบเทียบแรงดัน

“000” อยู่ในสภาวะรีเซต ไม่มีการทำงาน

“001” เปรียบเทียบแรงดันอิสระ ส่งค่าออกขาพอร์ต GP2/COUT

“010” เปรียบเทียบแรงดันอิสระ ส่งผลผ่านบิต COUT

“011” เปรียบเทียบแรงดันโดยใช้อ้างอิงภายใน ส่งค่าออกขาพอร์ต GP2/COUT

“100” เปรียบเทียบแรงดันโดยใช้อ้างอิงภายใน ส่งผลผ่านบิต COUT

“101” เปรียบเทียบแรงดันโดยใช้อ้างอิงภายใน ส่งค่าออกขาพอร์ต GP2/COUT และสามารถเลือกขาพอร์ตอินพุตได้ขึ้นอยู่กับการกำหนดที่บิต CIS

“110” เปรียบเทียบแรงดัน โดยใช้แรงดันอ้างอิงภายใน เลือกขาพอร์ตอินพุตได้จากบิต CIS

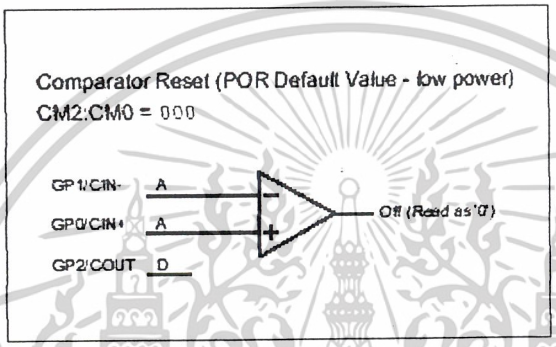
“111” วงจรเปรียบเทียบแรงดันไม่ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมดการทำงานของโมดูลเปรียบเทียบแรงดันอนาลอก

โหมดรีเซต (CM2-CM2: 000)

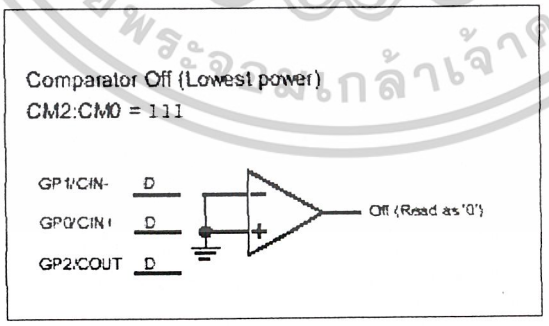
โมดูลเปรียบเทียบแรงจะไม่มีการทำงานใดๆ ในโหมดนี้ เมื่อไมโครคอนโทรลรีเซตโมดูลเปรียบเทียบแรงดันจะรีเซตด้วยเช่นกัน เอาต์พุตอ่านค่าได้"0"



รูปที่ 2.13 โหมดรีเซต(CM2-CM2: 000)

โหมดวงจรเปรียบเทียบอิสระและค่าส่งออกเอาต์พุต (CM2-CM0: 001)

แสดงการทำงานเป็นการเลือกให้เปรียบเทียบแรงดันอินพุต CP0/CIN+ กับ GP1/CIN- ผลการเปรียบเทียบจะปรากฏที่บิต COUT และขาพอร์ต GP2/COUT

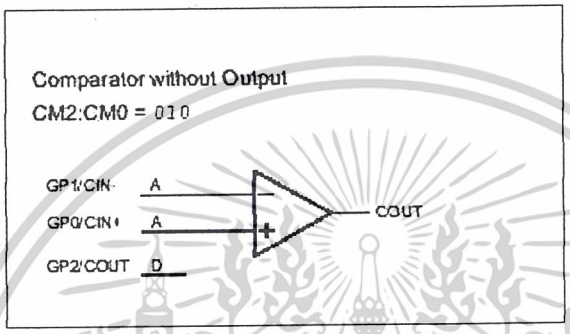


รูปที่ 2.14 โหมดวงจรเปรียบเทียบอิสระและค่าส่งออกเอาต์พุต(CM2-CM0: 001)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมดวงจรเปรียบเทียบอิสระ(CM2-CM0: 010)

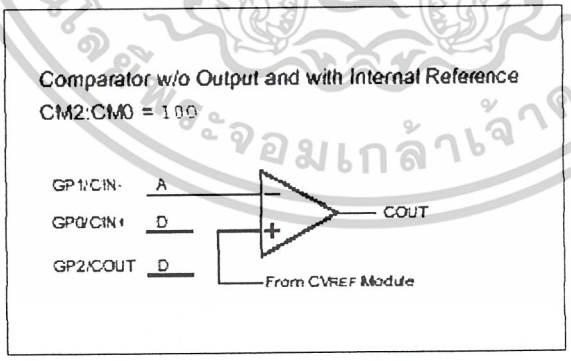
แสดงการเปรียบเทียบแรงดันระหว่างอินพุตCP0/CIN+ กับ GP1/CIN- ผลการเปรียบเทียบจะปรากฏที่บิต COUT



รูปที่ 2.15 โหมดวงจรเปรียบเทียบอิสระ(CM2-CM0: 010)

โหมดการวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายในและค่าส่งออกเอาต์พุต (CM2-CM1: 011)

แสดงการเปรียบเทียบแรงดันระหว่างอินพุต GP1/CIN- กับแรงดันอ้างอิงภายในของ rfPIC12F675 ผลการเปรียบเทียบจะปรากฏที่บิต COUT และขาพอร์ต GP2/COUT

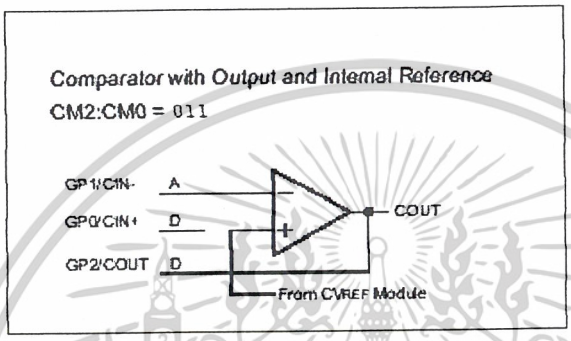


รูปที่ 2.16 โหมดการวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายในและค่าส่งออกเอาต์พุต (CM2-CM1: 011)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมดการวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายใน (CM2-CM1:100)

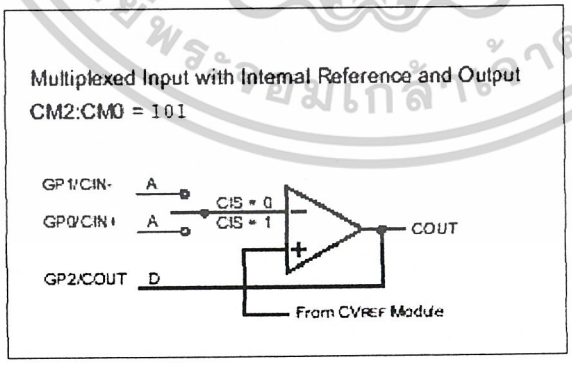
แสดงการเปรียบเทียบแรงดันระหว่างอินพุตGP1/CIN-กับแรงดันอ้างอิงภายในของ rfPIC12F675 ผลการเปรียบเทียบจะปรากฏที่บิต COUT



รูปที่ 2.17โหมดการวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายใน (CM2-CM1: 100)

โหมดวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายในแบบเลือกอินพุตได้และส่งค่าออกเอาต์พุต (CM2-CM0:101)

สามารถเลือกขาอินพุตของโมดูลเปรียบเทียบแรงดันได้โดยจะเปรียบเทียบแรงดันระหว่าง อินพุต GP0/CIN+ หรือ GP1/CIN- ถ้าบิตกับแรงดันอ้างอิงภายใน ขึ้นกับการกำหนดบิต CIS หากบิต CIS เป็น “0” จะเลือกขา GP1/CIN- ถ้าบิต CIS เป็น “1” จะเลือกแรงดันอินพุต GP0/CIN+ มาเปรียบเทียบแทน ผลการเปรียบเทียบจะปรากฏที่บิต COUT และขาพอร์ต GP2/COUT

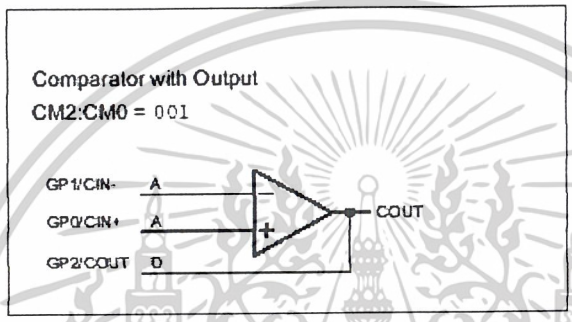


รูปที่ 2.18 โหมดวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายในแบบเลือกอินพุตได้และส่งค่าออกเอาต์พุต (CM2-CM0:101)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมดวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายในแบบเลือกอินพุตได้(CM2-CM1: 110)

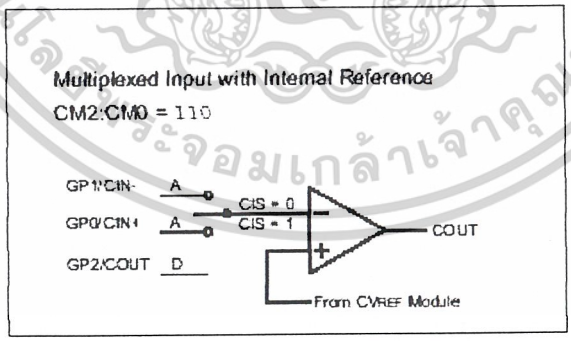
สามารถเลือกขาอินพุตของโมดูลเปรียบเทียบแรงดันได้ โดยจะเปรียบเทียบแรงดันระหว่างอินพุต GP0/CIN+ หรือ GP1/CIN- ถ้าบิตกับแรงดันอ้างอิงภายใน ขึ้นกับการกำหนดบิต CIS หากบิต CIS เป็น “0” จะเลือกขา GP1/CIN- ถ้าบิต CIS เป็น “1” จะเลือกแรงดันอินพุต GP0/CIN+ มาเปรียบเทียบแทน ผลการเปรียบเทียบจะปรากฏที่บิต COUT



รูปที่ 2.19 โหมดวงจรเปรียบเทียบใช้แรงดันอ้างอิงภายในแบบเลือกอินพุตได้(CM2-CM1: 110)

โหมดหยุดการทำงาน (CM2-CM0: 111)

ในโหมดนี้วงจรเปรียบเทียบทั้งหมดจะหยุดการทำงาน นั้นหมายความว่าขาพอร์ต A ทั้งหมดจะถูกกำหนดให้เป็นขาพอร์ตดิจิตอล ที่บิต COUT อ่านค่าได้เป็น “0”



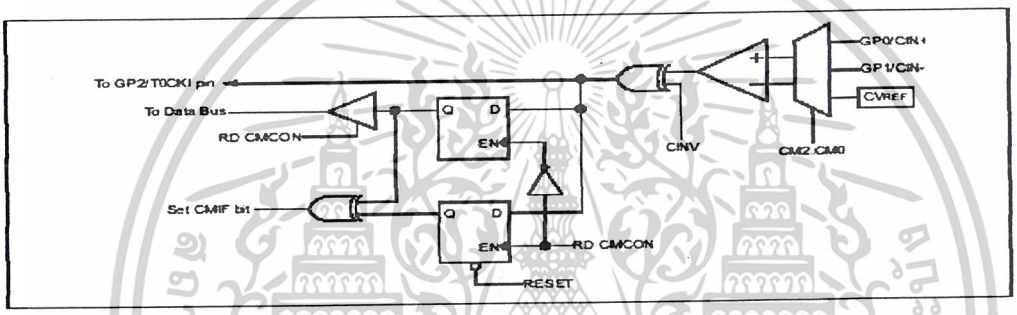
รูปที่ 2.20 โหมดหยุดการทำงาน (CM2-CM0: 111)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนเอาต์พุตของโมดูลเปรียบเทียบแรงดัน

ใน rfPIC12F675 แสดงออก 2 ทาง คือ ผ่านทางรีจีสเตอร์ CMCON ที่บิต COUT อีกท่งหนึ่ง คือ ส่งค่าออกทางขาพอร์ต GP2 นอกจากนั้นที่ขาพอร์ต GP2 ยังมีหน้าที่อื่นทับซ้อนอยู่ด้วย แต่จะไม่มีการทำงานพร้อมกันในแต่ละหน้าที่

เมื่อเลือกให้โมดูลเปรียบเทียบแรงดันทำงานในโหมดที่ส่งค่าออกทางขาพอร์ต การกำหนดทิศทางของขาพอร์ตที่รีจีสเตอร์ TRISIO ยังจำเป็นต้องกระทำอยู่ โดยหากต้องการให้มีสัญญาณออกมาที่ขาพอร์ตต้องกำหนดให้ขาพอร์ต GP2 เป็นเอาต์พุตด้วยเสมอ



รูปที่ 2.21 วงจรส่วนเอาต์พุตของโมดูลเปรียบเทียบแรงดันอนาลอกใน rfPIC12F675

การอินเตอร์รัพต์ในโมดูลเปรียบเทียบแรงดันอนาลอก

จะเกิดขึ้นเมื่อมีการเปลี่ยนแปลงค่าเอาต์พุตของโมดูลเปรียบแรงดัน ดังนั้นในการเขียนโปรแกรมเพื่อใช้งานโมดูลเปรียบเทียบแรงดันอนาลอกจำเป็นอย่างยิ่งที่ต้องดูแลสถานะที่บิต COUT บิตที่ 6 ในรีจีสเตอร์ CMCON เนื่องจากเป็นตัวกำหนดเงื่อนไขในการเปลี่ยนแปลง

เมื่อเกิดการเปลี่ยนแปลงขึ้นที่ขาเอาต์พุตวงจรเปรียบเทียบ จะเกิดการเซตบิต CMIF บิต 3 ในรีจิสเตอร์ PIR1 ถ้าหากมีการเอ็นเอเบิลอินเตอร์รัพต์แบบนี้ไว้ ก็จะเกิดการอินเตอร์รัพต์ขึ้นและซีพียูในไมโครคอนโทรลเลอร์ก็ต้องกระโดดไปทำงานอินเตอร์ต่อไป

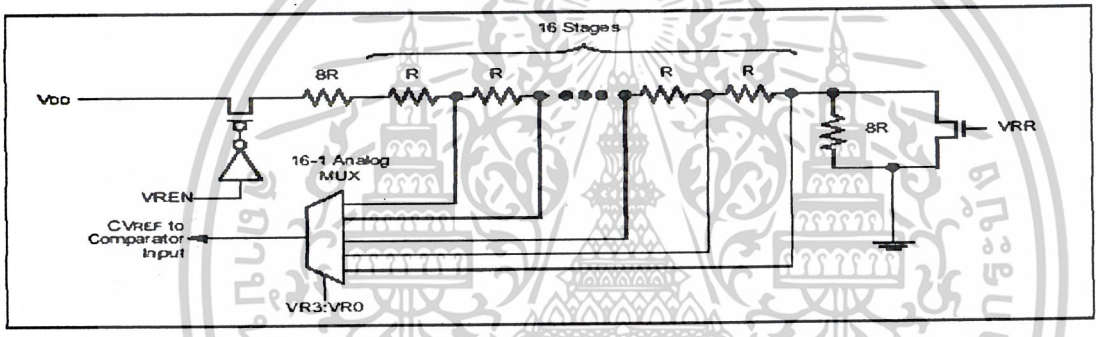
การเอ็นเอเบิลการอินเตอร์อันเนื่องจากโมดูลเปรียบเทียบแรงดันอนาลอกทำได้โดยเซตบิต CMIE บิตที่ 3 ในรีจีสเตอร์ PIE1 ร่วมกับบิต PEIE บิตที่ 6ในรีจีสเตอร์ INTCON และบิต GIE หากเงื่อนไขใดเงื่อนไขหนึ่งไม่ครบ การอินเตอร์รัพต์ไม่เกิดขึ้น และทุกครั้งที่อินเตอร์รัพต์ไปแล้วต้องเคลียร์บิต CMIF เสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมดูลสร้างแรงดันอ้างอิงภายในสำหรับเปรียบเทียบแรงดัน

ภายในโมดูลนี้ประกอบด้วยตัวต้านทาน 2 ค่าที่ต่อร่วมกันในลักษณะคล้ายบันได 16 จุดต่อ โดยต่อเข้ากับวงจรอนาลอกมัลติเพล็กซ์แบบเข้า 16 ออก 1 ค่าของตัวต้านทานในโมดูนี้มีความแตกต่างกัน 8 เท่า คือ R และ 8R ค่าความต้าน R กำหนดให้เท่ากับ 2 kΩ ค่าแรดันเอาต์พุตและการทำงานทั้งหมดของโมดูลนี้ขึ้นอยู่กับรีจีสเตอร์ VRCON ซึ่งมีแอดเดรสอยู่ที่ 0x99

แรงดันอ้างอิงที่สร้างขึ้นนี้จะถูกต่อเข้ากับอินพุต VIN+ ของโมดูลเปรียบเทียบแรงดันอนาลอกและไม่มีการต่อออกมาใช้งานภายนอก สามารถเลือกได้ 2 ย่านคือ 0-0.75 V_{DD} และ 0.25 V_{DD} – 0.75 V_D ขึ้นกับการกำหนดที่ รีจีสเตอร์ VRCON



รูปที่ 2.22 ไดอะแกรมการทำงานของโมดูลสร้างแรงดันอ้างอิงภายใน rfPIC12F675

คุณสมบัติอื่นๆ ของโมดูลสร้างแรงดันอ้างอิง

ค่าความละเอียดและค่าความผิดพลาด

เนื่องจากปลายทั้งสองของวงจรตัวต้านทานภายในโมดูลสร้างแรงดันอ้างอิงมีแฟตต่ออยู่ ทำให้แรงดันที่เกิดขึ้นไม่สามารถสร่างได้เต็มย่านตั้งแต่ V_{SS} – V_{DD} โดยแรงดันที่เดขึ้นจะเป็นการลดทอนจากแรงดัน V_{DD} หรือ ไฟเลี้ยงของไมโครคอนโทรลเลอร์ rfPIC12F675 ความละเอียดของแรงอ้างอิงที่สร้างขึ้นนี้มีค่าต่ำสุด V_{DD}/24LSb และสูงสุดเท่ากับ V_{DD}/32LSb

ความเร็วในการทำงาน

หลังจากที่เอ็นเอเบิลให้โมดูลสร้างแรงดันอ้างอิงทำงาน และกำหนดพารามิเตอร์ครบถ้วนแล้วไมโครคอนโทรลเลอร์จะใช้เวลาในการสร้างแรงดันอ้างอิง 10 ไมโครวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลกระทบในโหมดสลีป

เมื่อเข้าสู่โหมดสลีป โมดูลนี้จะหยุดทำงานด้วย แต่เมื่อเกิดการเวกอัปให้กับทำงานปกติไม่ว่าด้วยสาเหตุใดก็ตาม ค่าของรีจีสเตอร์ VRCON จะไม่เปลี่ยนแปลง

ผลกระทบเมื่อเกิดการรีเซต

เมื่อเกิดการรีเซตเกิดขึ้น โมดูลสร้างแรงดันอ้างอิงนี้จะถูกดิสเอเบิลตามไปด้วย เนื่องจาก VREN บิตที่ 7 ของ VRCON จะถูกเคลียร์เป็น "0" เช่นเดียวกันกับ VRR และบิต VR0-VR3 ก็จะเคลียร์เป็น "0" เมื่อเกิดการรีเซต

รีจีสเตอร์ VRCON ควบคุมโมดูลสร้างแรงดันอ้างอิงสำหรับเปรียบเทียบแรงดัน

เป็นรีจีสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ตำแหน่ง 0x99 ในแบงก์ 1 ค่าพารามิเตอร์แสดงในตารางที่ 2.17

ตารางที่ 2.17 รายละเอียดของบิตต่างๆ ในรีจีสเตอร์ VRCON รีจีสเตอร์ควบคุมโมดูลสร้างแรงดันอ้างอิงสำหรับเปรียบเทียบแรงดัน

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
VREN	—	VRR	—	VR3	VR2	VR1	VR0
bit 7							bit 0

บิต 7 : VREN (Comparator Voltage Reference Enable bit) บิตเลือกการทำงานของโมดูลสร้างแรงดันอ้างอิง

"1" จ่ายแรงดันให้แก่โมดูลสร้างแรงดัน

"0" ปิดการทำงานของโมดูลนี้

บิต 6 : ไม่ได้ใช้งานกำหนดเป็น "0"

บิต 5 : VRR (Comparator V_{REF} Value Selection) บิตเลือกย่านแรงดันอ้างอิง

"1" เลือกแรงดันอ้างอิงย่าน 0-0.75 V_{DD} มีระดับการเปลี่ยนแปลงค่าเท่ากับ $V_{DD}/32$ ต่อระดับ

"0" เลือกแรงดันอ้างอิงย่าน 0.25 V_{DD}- 0.75 V_{DD}มีระดับการเปลี่ยนแปลงเท่ากับ$V_{DD}/32$ ต่อระดับ

บิต 4 : ไม่มีการใช้งาน อ่านค่าเป็น "0"

บิต 3-0 : VR3: VR0 (Comparator V_{REF} Value Selection) บิตเลือกค่าแรงดันอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรงดันอ้างอิงที่เกิดขึ้นของโมดูลนี้จะขึ้นอยู่กับการกำหนดค่าของบิต V_R ทั้ง 4 รวมกับค่าของบิต V_{RR} โดยค่าของบิต V_R ทั้ง 4 บิตจะเกิดเป็นค่าของเลขฐาน 10 ได้ตั้งแต่ 0-15 ค่าของแรงดันสามารถคำนวณได้จาก

1. ในกรณีบิต VRR เท่ากับ "0" แรงดันอ้างอิงเท่ากับ $\frac{1}{4}V_{DD} + \{ \ VR_{10} \times V_{DD} \ \}/32$
2. ในกรณีบิต VRR เท่ากับ "1" แรงดันอ้างอิงเท่ากับ $\{ \ VR_{10} \times V_{DD} \ \}/24$

โดยที่ VR_{10} คือ ค่าของบิต VR3-VR0 ในรูปเลขฐาน 10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

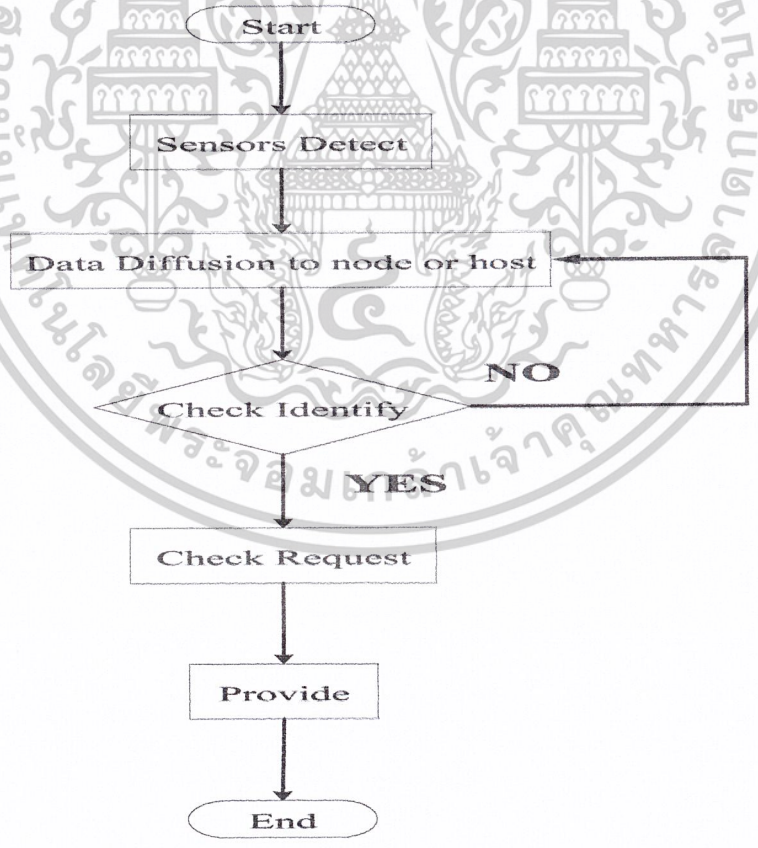
บทที่ 3

การออกแบบ

บทนี้กล่าวถึงการออกแบบซึ่งจะประกอบด้วยการออกแบบภาคส่ง ภาครับภาคเซนเซอร์ตรวจจับความชื้น และยังรวมถึงการออกแบบโปรโตคอลที่ใช้ในการติดต่อสื่อสาร

3.1 การออกแบบโครงงานโดยรวม

โครงงานนี้จะประกอบด้วย 3 ภาคด้วยกันได้แก่ ภาครับ ภาคส่งและภาคเซนเซอร์โดยจะมีการทำงานร่วมกัน คือ เริ่มจากเซนเซอร์ตรวจจับความชื้นที่กำหนดไว้แล้วทำการส่งข้อมูลไปยังโอสต์โดยผ่านยังโหนดต่างๆซึ่งจะทำการตรวจสอบว่าเป็นหมายเลขประจำโหนดของตัวเองหรือไม่ ถ้าใช่ก็จะทำการตรวจสอบความต้องการแต่ถ้าไม่ใช่ก็จะส่งแพ็กเก็ตต่อไปยังเป้าหมายซึ่งก็คือโฮส์ต ดังแสดงในรูปที่ 3.1

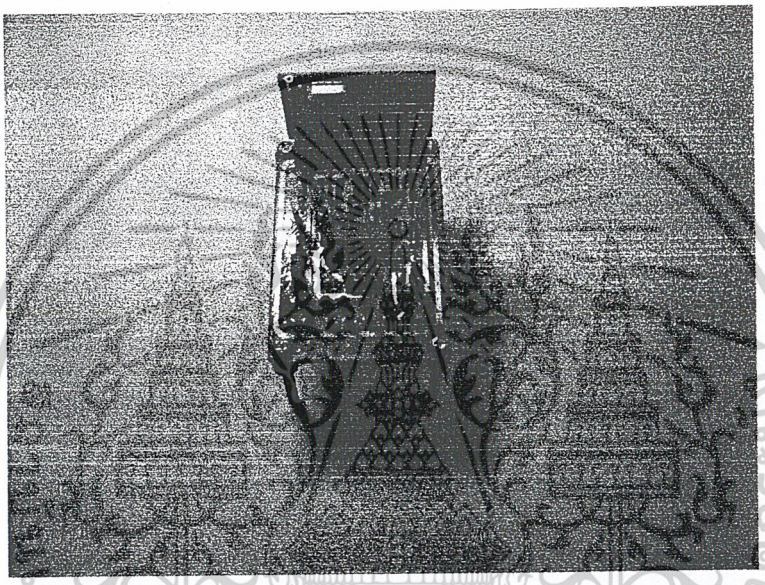


รูปที่ 3.1ลำดับขั้นการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ภาครับ (Receiver)

ภาครับซึ่งจะแสดงผลของข้อมูลที่รับมาขนาด 8 บิตซึ่งข้อมูลขนาด 8 บิตนี้จะประกอบด้วยหมายเลขประจำตัวของโหนดและสถานะของเซนเซอร์ที่ส่งข้อมูลมาของแต่ละโหนด ดังในรูปที่ 3.2 ส่วนของวงจรภาครับ โปรแกรมอินเตอร์เฟสและ PCB จะแสดงอยู่ในภาคผนวก (ข.)

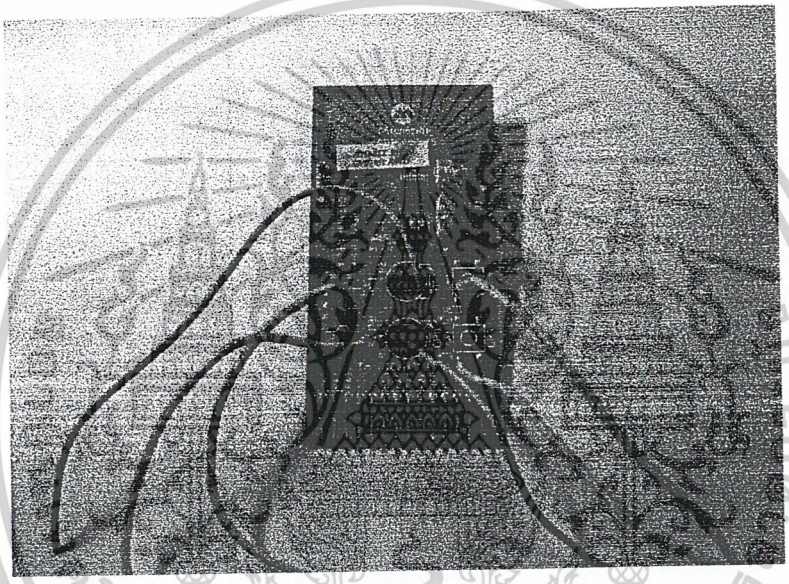


รูปที่ 3.2 ภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ภาคส่ง (Transmitter)

ภาคส่งจะส่งข้อมูลขนาด 8 บิต ซึ่งข้อมูลจะถูกส่ง ก็ต่อเมื่อเซนเซอร์ความชื้นตรวจพบ 2 สถานะคือ ชื้นและแห้ง ตามที่ผู้ใช้ได้กำหนดระดับของความชื้นที่ต้องการที่จะส่งข้อมูลไว้ ดังในรูปที่ 3.3 ส่วนของวงจรภาคส่ง โปรแกรมอินเตอร์เฟสและ PCB จะแสดงอยู่ในภาคผนวก (ค.)

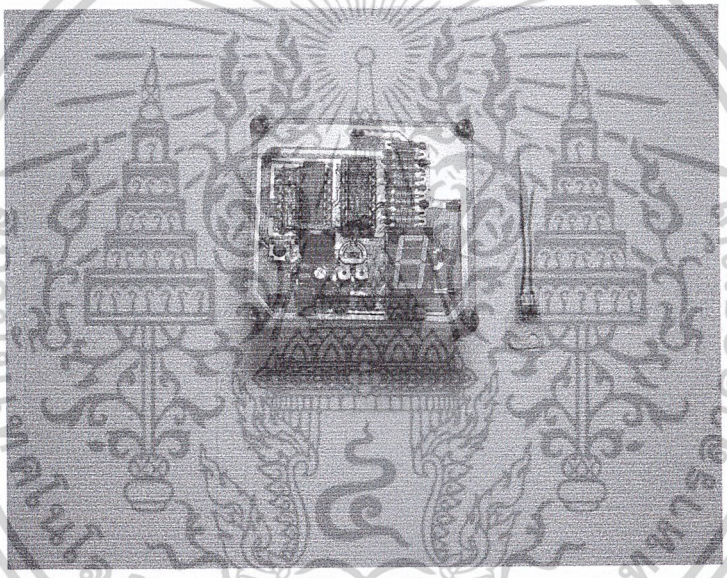


รูปที่ 3.3 ภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 เซนเซอร์ (Sensor)

เซนเซอร์ตรวจจับความชื้น ซึ่งจะแบ่งความชื้นออกเป็น 9 ระดับ คือ 0–9 ซึ่งได้กำหนดไว้ว่าที่ระดับ 5 คือ สถานะความชื้นที่น้อยที่จะต้องส่งข้อมูลออกไปและ ที่ระดับ 9 คือ สถานะความชื้นมากซึ่งอาจจะหมายถึง ตรวจพบน้ำนั่นเอง และจะต้องส่งข้อมูลออกไปเช่นกัน ดังแสดงในรูปที่ 3.4 ส่วนของวงจรภาคเซนเซอร์ โปรแกรมอินเตอร์เฟสและ PCB จะแสดงอยู่ในภาคผนวก (ก.)

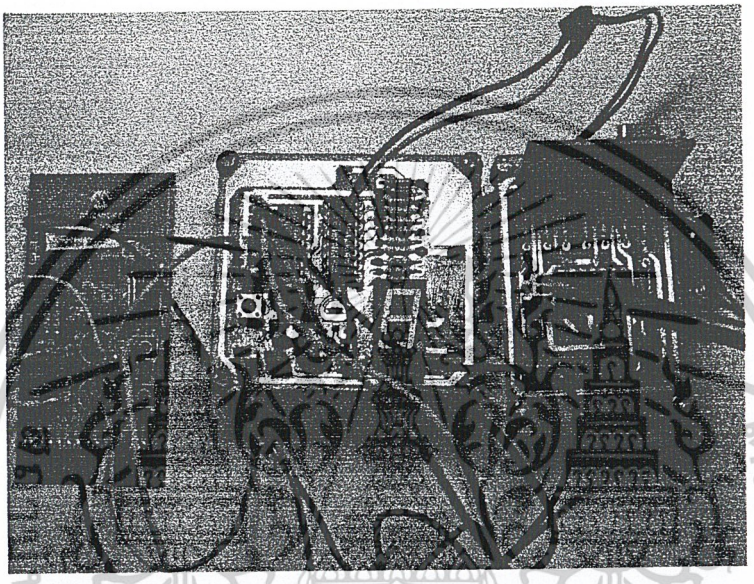


รูปที่ 3.4 เซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 โหนด (Node)

ในแต่ละโหนดจะประกอบด้วยภาคส่ง ภาครับ และ เซนเซอร์ตรวจจับความชื้น ซึ่งแต่ละโหนดจะรับและส่งข้อมูลต่อกันและกันโดยมีเป้าหมายไปยังโฮสต์ ดังแสดงในรูปที่ 3.5

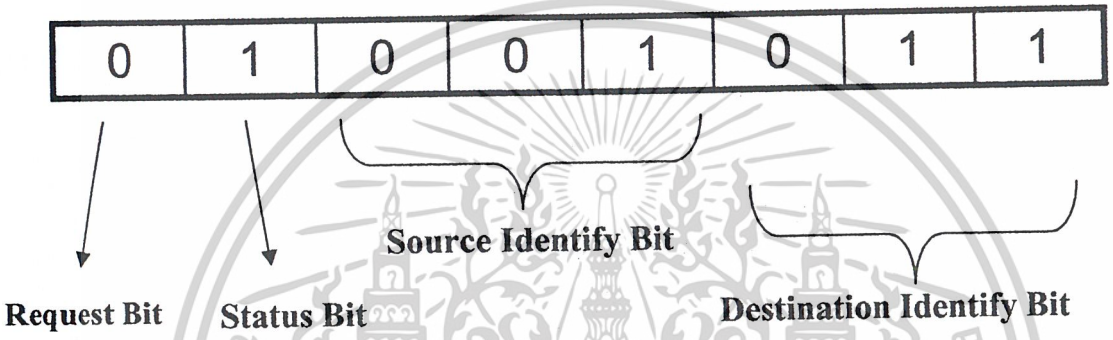


รูปที่ 3.5 โหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 โปรโตคอล (Protocol)

แพ็กเกตข้อมูลขนาด 8 บิต โดย 3 บิตแรกเป็นหมายเลขประจำโหนดเป้าหมายที่ต้องการที่จะส่งข้อมูลไป ส่วน 3 บิตต่อมาจะเป็นหมายเลขประจำโหนดต้นทางที่จะส่งข้อมูลและ 2 บิตสุดท้ายจะเป็นบิตแสดงสถานะของความชื้นและความต้องการของแต่ละโหนด ดังแสดงในรูปที่ 3.6



รูปที่ 3.6 โปรโตคอลการสื่อสารระหว่างโหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

บทนี้กล่าวถึง การทดลองและ ผลการทดลองที่ได้แล้วในโครงงานนี้โดยทำการวัดสัญญาณที่จุดต่างๆ แล้วบันทึกค่าไว้เปรียบเทียบกับค่าที่ได้ทางทฤษฎี

4.1 วัตถุประสงค์การทดลอง

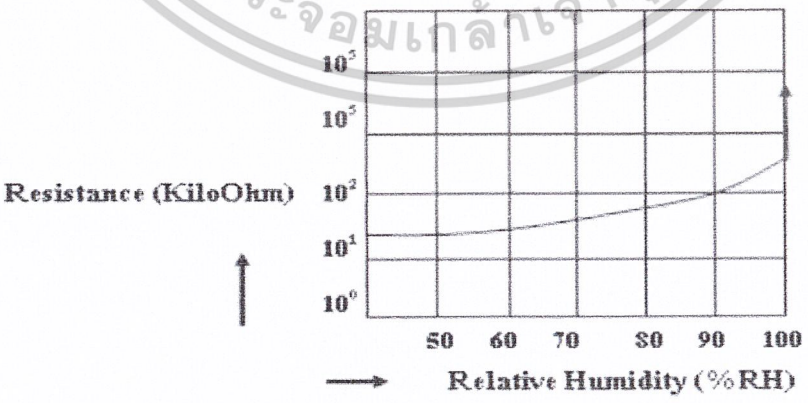
- เพื่อเปรียบเทียบค่าที่ได้จากการทดลองว่าเป็นไปตามทฤษฎีหรือไม่
- เพื่อเป็นแนวทางในการวิเคราะห์หาข้อบกพร่องของโครงงาน

4.2 อุปกรณ์การทดลอง

- เซนเซอร์ที่ได้ประกอบวงจรเรียบร้อยแล้ว	1	ชุด
- ภาคส่งที่ได้ประกอบวงจรเรียบร้อยแล้ว	1	ชุด
- ภาครับที่ได้ประกอบวงจรเรียบร้อยแล้ว	1	ชุด
- มัลติมิเตอร์	1	เครื่อง

4.3 การทดลองกับเซนเซอร์

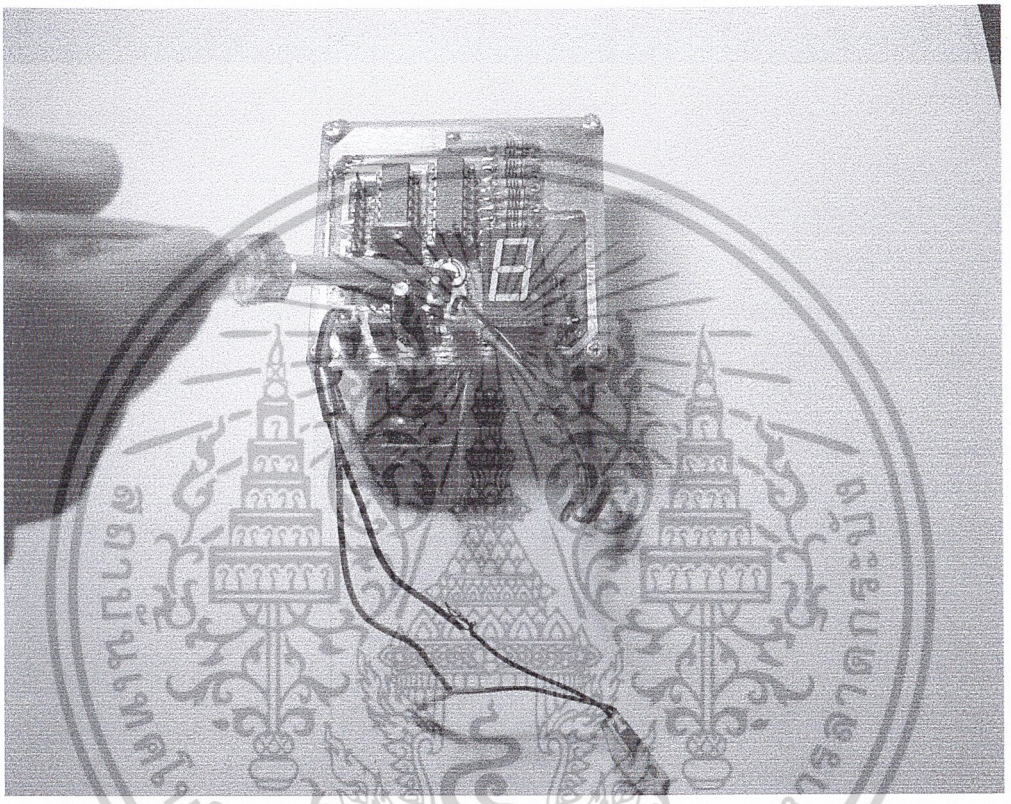
1. ทำการปรับแรงดันอ้างอิงให้ได้ 1.25 โวลต์เพราะเนื่องจากตัวเซนเซอร์รับแรงดันได้สูงสุดเพียง 0.8 โวลต์



รูปที่ 4.1คุณสมบัติของเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

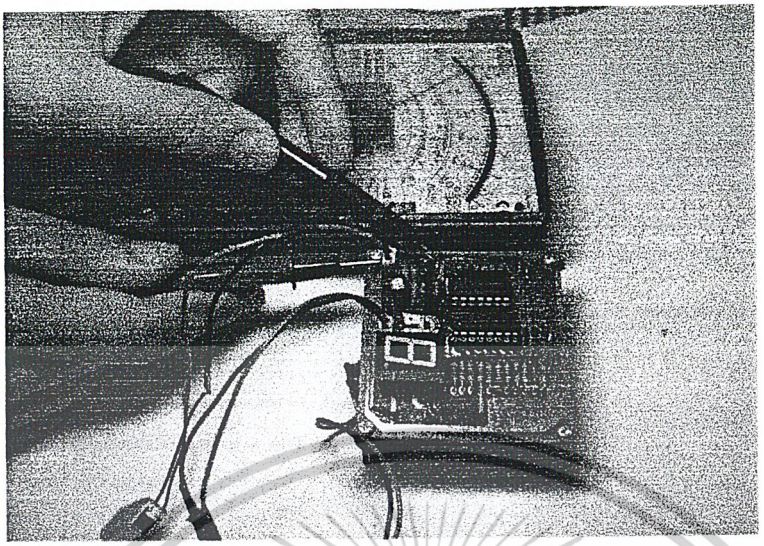
คุณสมบัติของเซนเซอร์ชนิดนี้ จะเปรียบเสมือนตัวต้านทานปรับค่าได้ คือ เมื่อค่าความชื้นสัมพัทธ์มีการเปลี่ยนแปลงสูงขึ้น ก็จะทำให้ค่าของความต้านทานมีค่าเพิ่มขึ้นเช่นกันจะแปรผันตรงซึ่งกันและกัน ดังแสดงในรูปที่ 4.1



รูปที่ 4.2 การปรับค่าแรงดันอ้างอิงให้ได้1.25 โวลต์

2. ทำการเปรียบเทียบค่าระดับของความชื้นที่เป็นดิจิตอลที่วัดได้กับค่าแรงดันอนาล็อกอินพุตที่วัดได้เพื่อที่จะได้นำไปหาค่าของความชื้นสัมพัทธ์ ดังแสดงในตารางที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 การวัดเปรียบเทียบค่าความชื้น

ตารางที่ 4.1 ผลการทดลอง

ค่าแรงดันอนาล็อกอินพุต (v)	**ค่าระดับดิจิตอล**	**ค่าความชื้นสัมพัทธ์ (%)**
0.05	4	4
0.15	5	12
0.3	6	24
0.55	7	44
0.95	8	76
1.20	9	96

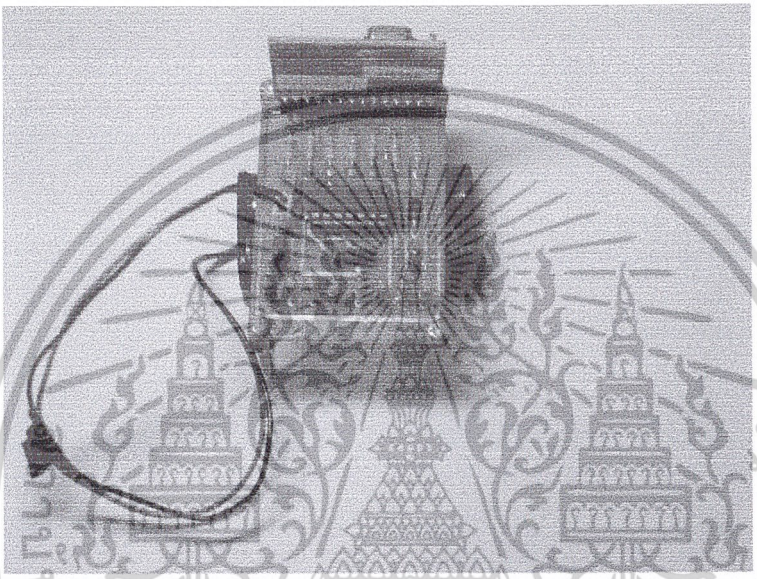
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิเคราะห์ผลการทดลอง

จากผลการทดลอง สังเกตผลการทดลองที่ได้ จะเห็นได้ว่า ค่าที่ได้ จะเป็นค่าโดยประมาณ ซึ่งจะไม่เป็นเชิงเส้นนัก และ ค่าที่ได้อาจจะมีข้อผิดพลาดเกิดขึ้นอันเนื่องมาจากสาเหตุต่างๆ ตามผลการทดลองที่ได้ตามนั้น

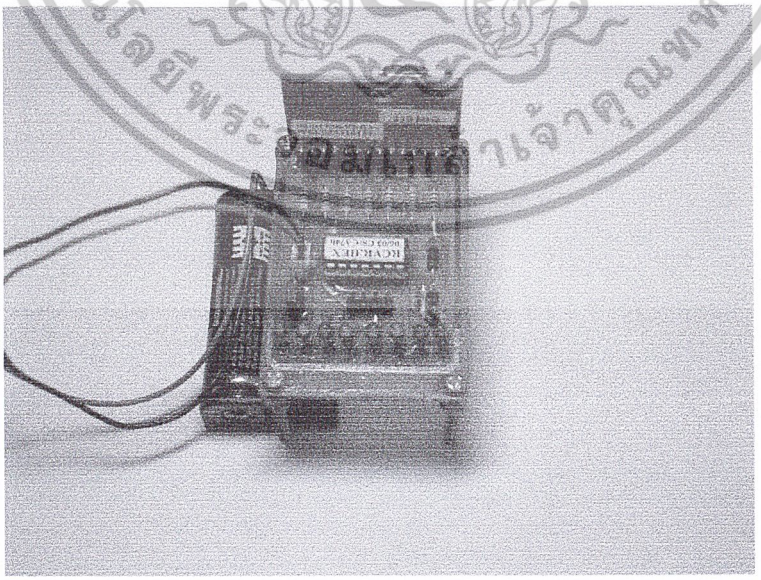
4.4 การทดลอง รับ- ส่ง ข้อมูล

4.4.1. ทำการทดลองส่งค่าขนาด 8 บิต ค่าที่ส่งคือ 0xFF



รูปที่ 4.4 ผลการทดลองส่งค่า 0xFF

4.4.2. ทำการทดลองส่งค่าขนาด 8 บิต ค่าที่ส่งคือ 0x3A



รูปที่ 4.5 ผลการทดลองส่งค่า 0x3A

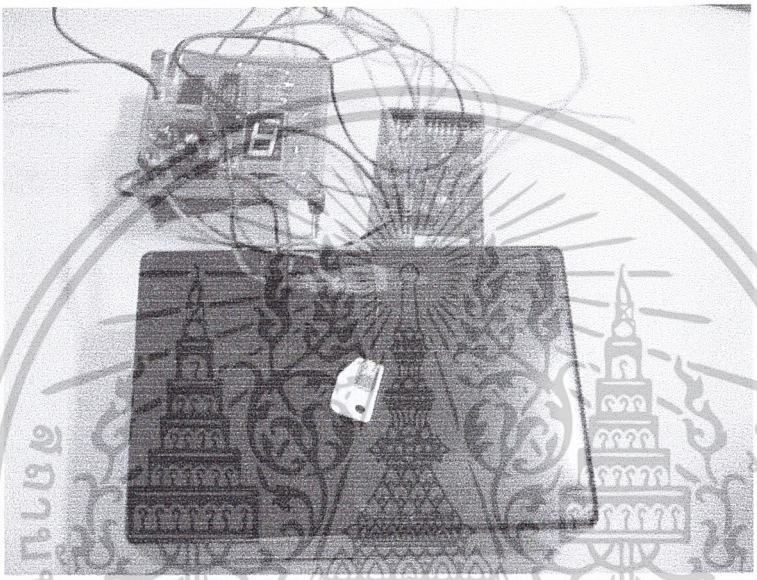
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิเคราะห์ผลการทดลอง

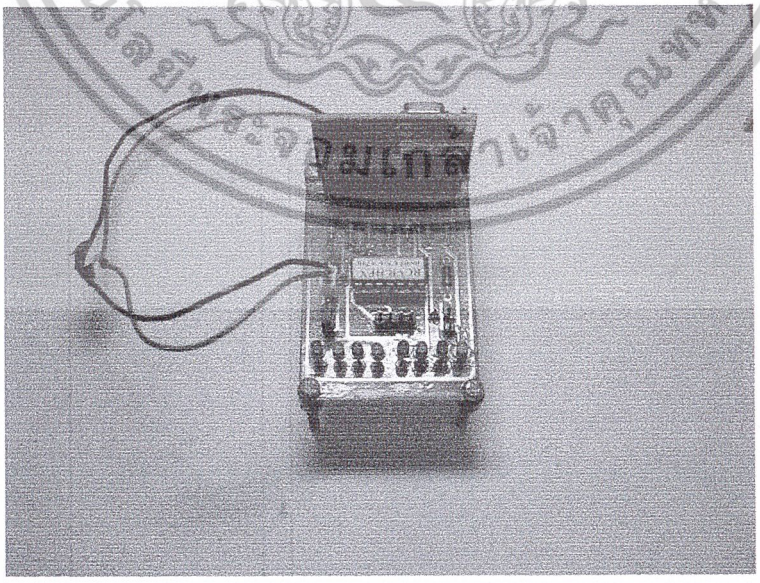
จากผลการทดลอง จะเห็นได้ว่าเมื่อทำการส่งค่า 0xFF หลอดแสดงผลสว่างทั้งหมดคือ 11111111 เมื่อทำการส่งค่า 0x3A หลอดแสดงผลจะแสดงผล 3A คือ 00111010

4.5 การทดลองเชื่อมต่อเซนเซอร์กับภาคส่ง

1. เมื่อเซนเซอร์ตรวจจับค่าความชื้นระดับที่ 5 และทำการบันทึกค่าผลการทดลองที่ได้



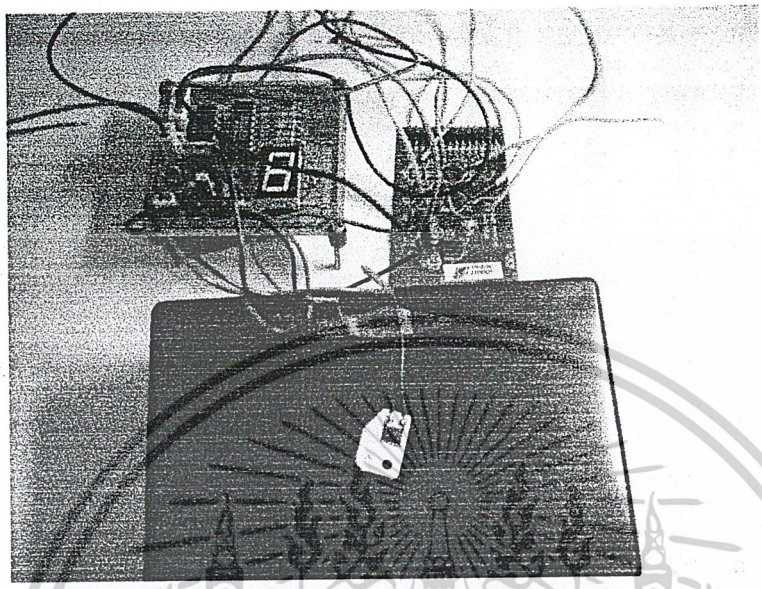
รูปที่ 4.6 เซนเซอร์ตรวจจับความชื้นที่ระดับ 5 และ ส่งค่า



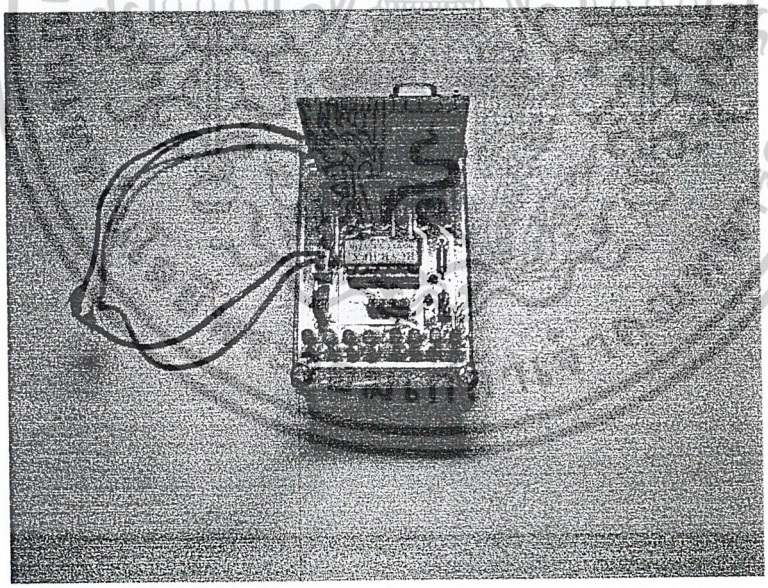
รูปที่ 4.7 ภาครับการส่งค่าเมื่อเซนเซอร์ตรวจจับที่ระดับ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เมื่อเซนเซอร์ตรวจจับค่าความชื้นระดับที่ 9 และทำการบันทึกค่าผลการทดลองที่ได้



รูปที่ 4.8 เซนเซอร์ตรวจจับความชื้นที่ระดับ 9 และ ส่งค่า



รูปที่ 4.9 ภาครับการส่งค่าเมื่อเซนเซอร์ตรวจจับที่ระดับ 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิเคราะห์ผลการทดลอง

จากการทดลองสังเกตผลการทดลองที่ได้ เมื่อเซนเซอร์ตรวจความชื้นที่ระดับ 5 ภาคส่งก็ส่งค่าไปยังภาครับเพื่อแสดงสถานะการตรวจจับที่ระดับความชื้นน้อย และเมื่อเซนเซอร์ตรวจจับที่ระดับ 9 จะส่งค่าไปยังภาครับเพื่อแสดงสถานะการตรวจจับที่ระดับความชื้นมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลองและปัญหาที่พบ

5.1 สรุปผลการทดลอง

จากการทดลองผลที่ได้ คือ ภาครับและ ภาคส่ง สามารถ รับ-ส่ง กันได้ โดยเมื่อกำหนดระดับความชื้นของเซนเซอร์ ที่ระดับ 9 ซึ่งหมายถึงการตรวจพบน้ำภาคส่งก็จะส่งค่าไปโฮสต์และเช่นเดียวกันเมื่อกำหนดระดับความชื้นของเซนเซอร์ที่ระดับ 5 ซึ่งหมายถึงมีความชื้นน้อยซึ่งเมื่อเซนเซอร์ตรวจพบก็จะส่งค่าไปยังโฮสต์เช่นเดียวกัน ซึ่งก็หมายถึงจะส่งค่า 2 สถานการณ์ คือ ค่าความชื้นมากกับค่าความชื้นน้อย แต่ยังไม่สามารถที่จะส่งข้อมูลต่อได้ ตามผลการทดลองที่ได้ตามนั้น

5.2 ปัญหาที่พบ

ปัญหาที่พบสำหรับฮาร์ดแวร์ก็คือ ความไม่มีเสถียรภาพ ซึ่งมักจะเกิดขึ้นเป็นบางครั้งกับตัวรับและตัวส่งซึ่งอาจจะมีสาเหตุมาจากจุดบัดกรีไม่แน่น หรือ จุดผิดพลาดเล็กๆน้อยๆซึ่งฮาร์ดแวร์ ที่เกี่ยวกับ RF นั้น ความผิดพลาดเล็กๆน้อยๆก็อาจจะทำให้เกิดปัญหาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. ณัฏฐพล วงศ์สุนทรชัย,ชัยวัฒน์ ลิ้มพรจิตรวิไล ปฏิบัติการไมโครคอนโทรลเลอร์ PIC12F675

2. Wireless Sensor Networks1 F. L. LEWIS

3. Microchip rfPIC12F675 Data Sheet CMOS Microcontroller with UHF ASK/FSK Transmitter

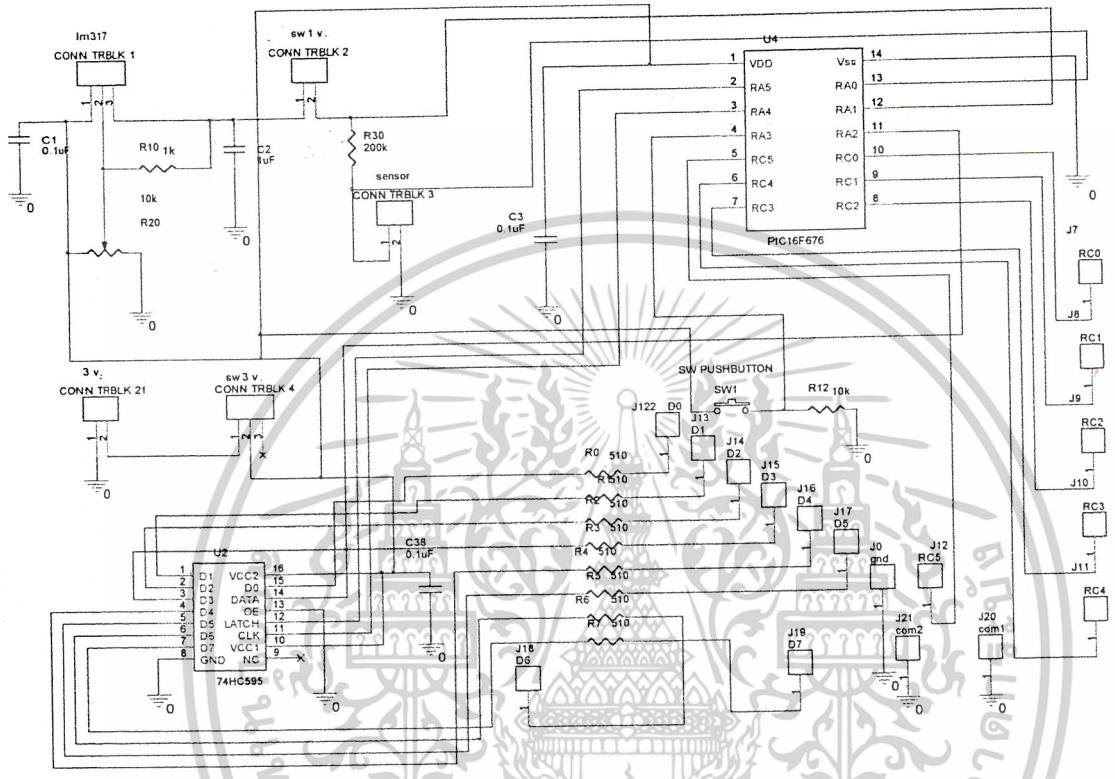
4. MPLAB PIC-micro development tool และการทดลองทางซอฟต์แวร์ของไมโครคอนโทรลเลอร์ PIC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

ภาคผนวก (ก.) เซนเซอร์

1. วงจรภาคเซนเซอร์



2. โปรแกรมอินเตอร์เฟส

```
    LIST  P=16F676,W=-302

    #INCLUDE <P16F676.INC>

    __CONFIG   _CPD_OFF & _CP_OFF & _BODEN_OFF & _MCLRE_OFF & _PWRTE_OFF & _WDT_OFF &
_INTRC_OSC_NOCLKOUT

    CBLOCK  0x20
    CNT,SEND,COUNT,DUMMY1,DUMMY2,TEMP,CHKLED,CHKLED1,CHKBIT_H,CHKBIT_L
    ENDC
#DEFINE  LAT  PORTA,5
#DEFINE  DAT  PORTA,2
#DEFINE  CLK  PORTA,4
#DEFINE  PUSH PORTA,3
#DEFINE  PORTC0 PORTC,0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#DEFINE POTC1 PORTC,1
   ORG 0x000
      GOTO   INITIAL
ADC    BCF   STATUS,RP0
      BCF   ADCON0,CHS2
      BCF   ADCON0,CHS1
      BCF   ADCON0,CHS0
      BSF   ADCON0,ADON

      movlw  D'6'
      movwf  TEMP

      decfsz TEMP, F
      goto   $-1
      bsf    ADCON0, GO
      btfsc  ADCON0, GO
      goto   $-1
      BCF    ADCON0,ADON
      RETURN
SEND_0  BCF  STATUS,RP0
      MOVLW  0x3F
      MOVWF  SEND
      GOTO   SEND_LED

SEND_1  BCF  STATUS,RP0
      MOVLW  0x06
      MOVWF  SEND
      GOTO   SEND_LED

SEND_2  BCF  STATUS,RP0
      MOVLW  0x5B
      MOVWF  SEND
      GOTO   SEND_LED

SEND_3  BCF  STATUS,RP0
      MOVLW  0x4F
      MOVWF  SEND
      GOTO   SEND_LED

SEND_4  BCF  STATUS,RP0
      MOVLW  0x66
      MOVWF  SEND
      GOTO   SEND_LED
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SEND_5  BCF  STATUS,RP0
     MOVLW  0x6D
     MOVWF  SEND
     GOTO   SEND_LED

SEND_6  BCF  STATUS,RP0
     MOVLW  0x7D
     MOVWF  SEND
     GOTO   SEND_LED

SEND_7  BCF  STATUS,RP0
     MOVLW  0x07
     MOVWF  SEND
     GOTO   SEND_LED

SEND_8  BCF  STATUS,RP0
     MOVLW  0x7F
     MOVWF  SEND
     GOTO   SEND_LED

SEND_9  BCF  STATUS,RP0
     MOVLW  0x6F
     MOVWF  SEND
     GOTO   SEND_LED
RESET   BCF  STATUS,RP0
     CLRF   SEND
     MOVLW  0x3F
     MOVWF  SEND
     GOTO   SEND_LED
 TX_DAT0    BSF   POTC0
        CALL   DELAY
        CALL   DELAY
        CALL   DELAY
        CALL   DELAY
        CALL   DELAY
        BCF    POTC0
        GOTO   SEND_5
 TX_DAT1    BSF   POTC1
        CALL   DELAY
        CALL   DELAY
        CALL   DELAY
        CALL   DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CALL   DELAY
        BCF    POTC1
        GOTO   SEND_9

INITIAL MOVLW  .7
        MOVWF  CMCON
        BANKSEL ANSEL
        CALL   0x3FF
        MOVWF  OSCCAL
        BCF    STATUS,RP0
        bcf    T1CON, TMR1ON
        bcf    T1CON, TMR1CS
        bcf    T1CON, NOT_T1SYNC
        bcf    T1CON, T1OSCEN
        bcf    T1CON, T1CKPS1
        bcf    T1CON, T1CKPS0
        bcf    T1CON, TMR1GE
        BCF    ADCON0,ADFM
        BSF    ADCON0,VCFG
        BCF    ADCON0,ADON
        BSF    STATUS,RP0
        bcf    OPTION_REG, T0CS
        bcf    OPTION_REG, T0SE
        bcf    OPTION_REG, PSA
        bcf    OPTION_REG, PS2
        bcf    OPTION_REG, PS1
        bcf    OPTION_REG, PS0
        BCF    ADCON1,ADCS2
        BCF    ADCON1,ADCS1
        BSF    ADCON1,ADCS0
        BCF    ANSEL,ANS3
        BCF    ANSEL,ANS2
        BSF    ANSEL,ANS1
        BSF    ANSEL,ANS0
        BCF    TRISC,3
        BSF    TRISA,0
        BSF    TRISA,1
        BCF    TRISA,2
        BCF    TRISA,4
        BCF    TRISA,5
        BCF    TRISC,0
        BCF    TRISC,1
        BCF    TRISC,2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    BCF   TRISC,4
    BCF   TRISC,5

MAIN  BCF   STATUS,RP0
    CLRF  SEND
    CLRF  CHKBIT_H
    CLRF  CHKBIT_L
    ;BCF   POTC0
    ;BCF   POTC1
    CALL  ADC
    BSF   STATUS,RP0
    MOVFW  ADRESH
    BCF   STATUS,RP0
    MOVWF  CHKBIT_H
    MOVFW  ADRESL
    MOVWF  CHKBIT_L
    BTFSC  PUSH
    GOTO  RESET

    BTFSS  CHKBIT_L,7
    GOTO  $+2
    GOTO  TX_DAT1
    ;GOTO   SEND_9
    BTFSC  PUSH
    GOTO  RESET

    BTFSS  CHKBIT_L,6
    GOTO  $+2
    GOTO  SEND_8
    BTFSC  PUSH
    GOTO  RESET

    BTFSS  CHKBIT_L,5
    GOTO  $+2
    GOTO  SEND_7
    BTFSC  PUSH
    GOTO  RESET

    BTFSS  CHKBIT_L,4
    GOTO  $+2
    GOTO  SEND_6
    BTFSC  PUSH
    GOTO  RESET
```

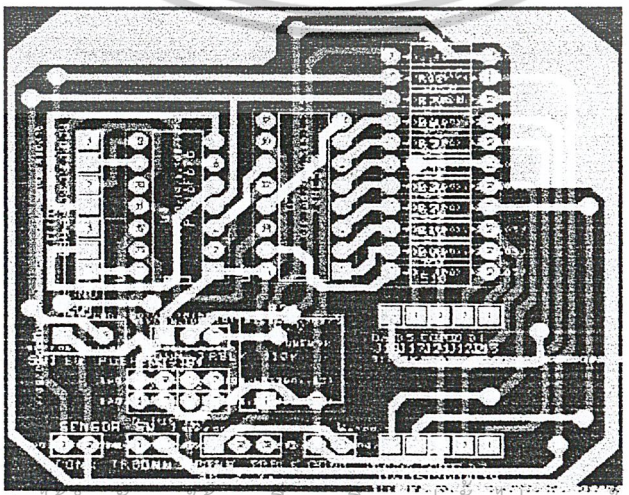
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    BTFSS  CHKBIT_L,3
    GOTO   $+2
    GOTO   TX_DAT0
    ;GOTO   SEND_5
    BTFSC  PUSH
    GOTO   RESET

    BTFSS  CHKBIT_L,2
    GOTO   $+2
    GOTO   SEND_4
    BTFSC  PUSH
    GOTO   RESET

    BTFSS  CHKBIT_L,1
    GOTO   $+2
    GOTO   SEND_3
    BTFSC  PUSH
    GOTO   RESET

    BTFSS  CHKBIT_L,0
    GOTO   $+2
    GOTO   SEND_2
    BTFSC  PUSH
    GOTO   RESET

    BTFSS  CHKBIT_H,7
    GOTO   $+2
    GOTO   SEND_1
    BTFSC  PUSH
    GOTO   RESET

    BTFSS  CHKBIT_H,6
    GOTO   $+2
    GOTO   SEND_0
    GOTO   MAIN


SEND_LED    CALL   SEND8BIT
            CALL   DELAY
            CALL   DELAY
            ;CALL   DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        GOTO  MAIN
SEND8BIT  MOVLW 0x08
      MOVWF COUNT
CHK_BIT   BCF DAT
    BTFSC SEND,7
    BSF DAT
CLOCK  BSF CLK
    BCF CLK
ROTATE RLF SEND,F
    DECFSZ COUNT,F
    GOTO CHK_BIT

LATCH  BSF LAT
    BCF LAT
    RETURN

DELAY  CLRF DUMMY1
DELAY1 CLRF DUMMY2
    DECFSZ DUMMY2,F
    GOTO $-1
    DECFSZ DUMMY1,F
    GOTO DELAY1
    RETURN

    ORG  0x3FF
    RETLW 0x80

    END
```

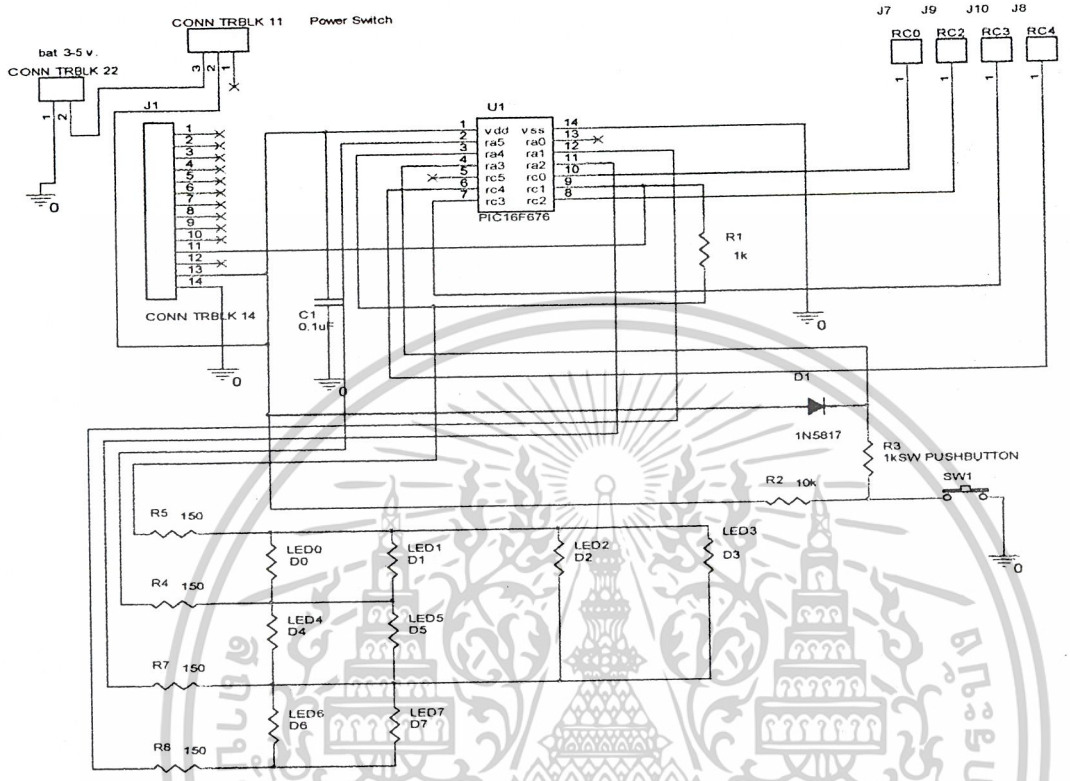
3. PCB LAYOUT



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก (ข.) ภาครับ

1. วงจรภาครับ



2. โปรแกรมอินเตอร์เฟส

```
list      p=16f676
#include   <p16f676.inc>

errorlevel  -302

 __CONFIG   _CPD_OFF & _CP_OFF & _BODEN_OFF & _MCLRE_OFF & _PWRTE_OFF & _WDT_OFF &
_INTRC_OSC_NOCLKOUT

cblock 0x20

  w_temp
  status_temp
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    w_temp
    status_temp

    TEMP
    DATA0
    DATA1
    DATA2
    DATA3
    ORIGIN
    SX1TMR
    SX2TMR
    TMRLOW
    TMRHIGH
    HIGHWDTH
    LOWWDTH
    STATECNTR
    BITCNTR
    FLAGS
    COUNTR

    LEDREG
    LEDREGBUF
    LEDSTATE

  endc


#define RA0     PORTA, 0
#define RA1     PORTA, 1
#define RA2     PORTA, 2
#define LRNPB   PORTA, 3
#define RA4     PORTA, 4
#define RA5     PORTA, 5
#define PORTATRIS  b'00111111'
#define RC0     PORTC, 0
#define RXDATA  PORTC, 1
#define RC2     PORTC, 2
#define RC3     PORTC, 3
#define RC4     PORTC, 4
#define RC5     PORTC, 5
#define PORTCTRIS  b'00100010'
#define     BEGN     0x00
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define    BEGN1    0x01
#define    HEADR    0x02
#define    HEADR1   0x03
#define    HIGHP    0x04
#define    LOWP     0x05
#define    RECRD    0x06
#define    WAIT     0x07
#define    VALID    0x08
#define    IMPLMNT          0x09
#define LRNFLG  FLAGS, 0
#define TOGFLG  FLAGS, 1
#define HLFLG   FLAGS, 2
#define LED0    LEDREG, 0
#define LED1    LEDREG, 1
#define LED2    LEDREG, 2
#define LED3    LEDREG, 3
#define LED4    LEDREG, 4
#define LED5    LEDREG, 5
#define LED6    LEDREG, 6
#define LED7    LEDREG, 7
#define LRNLED  LED3
#define LED0TRIS   b'00001111'
#define LED1TRIS   b'00001111'
#define LED2TRIS   b'00101011'
#define LED3TRIS   b'00101011'
#define LED4TRIS   b'00011011'
#define LED5TRIS   b'00011011'
#define LED6TRIS   b'00111001'
#define LED7TRIS   b'00111001'
#define LEDOFFTRIS b'00111111'
#define LED0ON     b'00010000'
#define LED1ON     b'00100000'
#define LED2ON     b'00010000'
#define LED3ON     b'00000100'
#define LED4ON     b'00100000'
#define LED5ON     b'00000100'
#define LED6ON     b'00000100'
#define LED7ON     b'00000010'

    ORG   0x000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    nop
    goto   RESET


    ORG    0x004

    movwf  w_temp
    swapf  STATUS, W
    bcf    STATUS, RP0
    movwf  status_temp
    swapf  status_temp, W
    movwf  STATUS
    swapf  w_temp, F
    swapf  w_temp, W

    retfie

DATA_EEPROM_READ

    bsf    STATUS, RP0
    movwf  EEADR
    bsf    EECON1, RD
    movf   EEDATA, W
    bcf    STATUS, RP0

    return

DATA_EEPROM_WRITE

    bsf    STATUS, RP0
    bsf    EECON1, WREN
    bcf    INTCON, GIE
    movlw  0x55
    movwf  EECON2
    movlw  0xAA
    movwf  EECON2
    bsf    EECON1, WR
    btfsc  EECON1, WR
    goto   $-1
    bcf    EECON1, WREN
    bcf    STATUS, RP0

    return
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
READ_ANALOG
    bsf   ADCON0, ADON
    bcf   ADCON0, CHS1
    bcf   ADCON0, CHS0
    movlw 0x07
    movwf TEMP
    decfsz TEMP, F
    goto  $-1
    bsf   ADCON0, GO
    btfsc ADCON0, GO
    goto  $-1
    bcf   ADCON0, ADON

    return

INITIAL

    bcf   INTCON, GIE
    bsf   STATUS, RP0
    call  0x3FF
    movwf OSCCAL
    bcf   STATUS, RP0
    bsf   STATUS, RP0
    movlw PORTATRIS
    movwf TRISA
    movlw PORTCTRIS
    movwf TRISC

    bcf   STATUS, RP0
    bsf   CMCON, CM2
    bsf   CMCON, CM1
    bsf   CMCON, CM0
    bsf   STATUS, RP0
    bcf   VRCON, VREN
    bcf   STATUS, RP0
    bcf   ADCON0, ADFM
    bcf   ADCON0, VCFG
    bcf   ADCON0, ADON
    bsf   STATUS, RP0
    bcf   ADCON1, ADCS2
    bcf   ADCON1, ADCS1
    bsf   ADCON1, ADCS0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
bcf   ANSEL, ANS7
bcf   ANSEL, ANS6
bcf   ANSEL, ANS5
bcf   ANSEL, ANS4
bcf   ANSEL, ANS3
bcf   ANSEL, ANS2
bcf   ANSEL, ANS1
bsf   ANSEL, ANS0
bcf   STATUS, RP0
bcf   T1CON, TMR1ON
bcf   T1CON, TMR1CS
bcf   T1CON, NOT_T1SYNC
bcf   T1CON, T1OSCEN
bcf   T1CON, T1CKPS1
bcf   T1CON, T1CKPS0
bcf   T1CON, TMR1GE
bsf   STATUS, RP0
movlw B'00000000'
movwf WPUA
bcf   STATUS, RP0
bsf   STATUS, RP0
bsf   OPTION_REG, NOT_GPPU
bsf   OPTION_REG, INTEDG
bcf   OPTION_REG, T0CS
bcf   OPTION_REG, T0SE
bcf   OPTION_REG, PSA
bcf   OPTION_REG, PS2
bcf   OPTION_REG, PS1
bsf   OPTION_REG, PS0
bcf   STATUS, RP0
bsf   STATUS, RP0
clrf  IOCA
bcf   STATUS, RP0
bsf   STATUS, RP0
bcf   PIE1, EEIE
bcf   PIE1, ADIE
bcf   PIE1, CMIE
bcf   PIE1, TMR1IE
bcf   STATUS, RP0
bcf   INTCON, PEIE
bcf   INTCON, T0IE
bcf   INTCON, INTE
bcf   INTCON, RAIE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    bcf   INTCON, GIE
    retlw  0
TIMER
    btfss  TOGFLG
     goto  TIMER1
    movlw  B'01111111'
    addwf  TMR0, W
    btfss  STATUS, C
     retlw 0
    bcf    TOGFLG
    incfsz SX1TMR, F
     retlw 0
     incf   SX2TMR, F
    retlw  0

TIMER1
    movlw  B'01111111'
    addwf  TMR0, W
    btfsc  STATUS, C
     retlw 0
     bsf    TOGFLG
    retlw  0
CLOCK
    movf   ORIGIN, W
    subwf  TMR0, W
    addwf  TMRLOW, F
    btfsc  STATUS, C
     incf  TMRHIGH, F

     nop
     nop
     nop
     movlw  2
     subwf  TMR0, W
     movwf  ORIGIN
     retlw  0
LRNDTCT
     btfsc  LRNFLG
     retlw 0
     movlw  BEGN
     movwf  STATECNTR
     bsf    LRNFLG
     bsf    LRNLED
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    clrf   SX1TMR
    clrf   SX2TMR
    retlw  0
LEARN
    btfss  SX2TMR, 5
     goto  LEARN1
    bcf    LRNFLG
    bcf    LRNLED
    retlw  0
LEARN1
    movlw  VALID
    xorwf  STATECNTR, W
    btfss  STATUS, Z
     retlw 0
    bsf    STATUS, RP0
    movlw  0x00
    movwf  EEADR
    movfw  DATA0
    movwf  EEDATA
    bcf    STATUS, RP0
    call   DATA_EEPROM_WRITE
    bsf    STATUS, RP0
    movlw  0x01
    movwf  EEADR
    movfw  DATA1
    movwf  EEDATA
    bcf    STATUS, RP0
    call   DATA_EEPROM_WRITE

     bcf   LRNLED
     bcf   LRNFLG
     movlw BEGN
     movwf STATECNTR
     retlw 0
SETWATCH
     movf  TMR0, W
     movwf ORIGIN
     clrf  TMRLOW
     clrf  TMRHIGH
     retlw 0
SXON
     btfss SX2TMR, 0
     retlw 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    clrf   LEDREG
    retlw  0
LEDStateMachine
    movlw  HIGH LEDStateMachineStart
    movwf  PCLATH
    movf   LEDSTATE, W
    andlw  B'00000111'
    addwf  PCL, F
LEDStateMachineStart
    goto   LEDSTATE0
    goto   LEDSTATE1
    goto   LEDSTATE2
    goto   LEDSTATE3
    goto   LEDSTATE4
    goto   LEDSTATE5
    goto   LEDSTATE6
    goto   LEDSTATE7
LEDSTATE0
    call   LITELED0
    bsf    RC0
    bcf    RC4
    bcf    RC3
    bcf    RC2
    bcf    RC0
    goto   EndLEDStateMachine
LEDSTATE1
    call   LITELED1
    bsf    RC0
    bcf    RC4
    bcf    RC3
    bsf    RC2
    bcf    RC0
    goto   EndLEDStateMachine
LEDSTATE2
    call   LITELED2
    bsf    RC0
    bcf    RC4
    bsf    RC3
    bcf    RC2
    bcf    RC0
    goto   EndLEDStateMachine
LEDSTATE3
    call   LITELED3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    bsf   RC0
    bcf   RC4
    bsf   RC3
    bsf   RC2
    bcf   RC0
    goto  EndLEDStateMachine
LEDSTATE4
    call  LITELED4
    bsf   RC0
    bsf   RC4
    bcf   RC3
    bcf   RC2
    bcf   RC0
    goto  EndLEDStateMachine
LEDSTATE5
    call  LITELED5
    bsf   RC0
    bsf   RC4
    bcf   RC3
    bsf   RC2
    bcf   RC0
    goto  EndLEDStateMachine
LEDSTATE6
    call  LITELED6
    bsf   RC0
    bsf   RC4
    bsf   RC3
    bcf   RC2
    bcf   RC0
    goto  EndLEDStateMachine
LEDSTATE7
    call  LITELED7
    bsf   RC0
    bsf   RC4
    bsf   RC3
    bsf   RC2
    bcf   RC0
EndLEDStateMachine
    return
LITELED0
    bsf   STATUS, RP0
    movlw  LED0TRIS
    movwf  TRISA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    bcf   STATUS, RP0
    movlw  LED0ON
    movwf  PORTA
    return
LITELED1
    bsf   STATUS, RP0
    movlw  LED1TRIS
    movwf  TRISA
    bcf   STATUS, RP0
    movlw  LED1ON
    movwf  PORTA
    return
LITELED2
    bsf   STATUS, RP0
    movlw  LED2TRIS
    movwf  TRISA
    bcf   STATUS, RP0
    movlw  LED2ON
    movwf  PORTA
    return
LITELED3
    bsf   STATUS, RP0
    movlw  LED3TRIS
    movwf  TRISA
    bcf   STATUS, RP0
    movlw  LED3ON
    movwf  PORTA
    return

LITELED4
    bsf   STATUS, RP0
    movlw  LED4TRIS
    movwf  TRISA
    bcf   STATUS, RP0
    movlw  LED4ON
    movwf  PORTA
    return
LITELED5
    bsf   STATUS, RP0
    movlw  LED5TRIS
    movwf  TRISA
    bcf   STATUS, RP0
    movlw  LED5ON
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    movwf  PORTA
    return
LITELED6
    bsf    STATUS, RP0
    movlw  LED6TRIS
    movwf  TRISA
    bcf    STATUS, RP0
    movlw  LED6ON
    movwf  PORTA
    return
LITELED7
    bsf    STATUS, RP0
    movlw  LED7TRIS
    movwf  TRISA
    bcf    STATUS, RP0
    movlw  LED7ON
    movwf  PORTA
    return
DISPLAY
    movf   LEDSTATE, W
    btfsc  STATUS, Z
    goto   BUFFERRESET
    btfss  LEDSTATE, 3
    goto   BUFFERLOADED
BUFFERRESET
    movf   LEDREG, W
    btfss  STATUS, Z
    goto   BUFFERRESET1

    bsf    STATUS, RP0
    movlw  LEDOFFTRIS
    movwf  TRISA
    bcf    STATUS, RP0
    clrf   PORTA
BUFFERRESET1
    movf   LEDREG, W
    movwf  LEDREGBUF
    clrf   LEDSTATE
BUFFERLOADED
    bcf    STATUS, C
    rrf    LEDREGBUF, F
    btfsc  STATUS, C
    goto   UPDATELED
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    goto   CONTINUE
UPDATELED
    call   LEDStateMachine
CONTINUE
    incf   LEDSTATE, F
    return
RESET
    clrf   FLAGS
    clrf   PORTA
    clrf   PORTC
    clrf   LEDREG
    clrf   LEDSTATE
    clrf   LEDREGBUF
    movlw  BEGN
    movwf  STATECNTR
    call   INITIAL
MAIN
    call   TIMER
    call   CLOCK
    call   DISPLAY
    movlw  B'00000111'
    andwf  LEDREG, W
    btfss  STATUS, Z
    call   SXON
    btfss  LRNPB
    call   LRNDTCT
    btfsc  LRNFLG
    call   LEARN

    movlw  HIGH STATEM
    movwf  PCLATH
    movf   STATECNTR, W
    andlw  B'00001111'
    addwf  PCL, F
STATEM
    goto   BEGIN
    goto   BEGIN1
    goto   HEADER
    goto   HEADER1
    goto   HIGHPLSE
    goto   LOWPULSE
    goto   RECORD
    goto   WAIT4END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    goto   VALIDATE
    goto   IMPLEMNT
    goto   RESET
    goto   RESET
    goto   RESET
    goto   RESET
    goto   RESET
    goto   RESET
BEGIN
    btfsc  RXDATA
    incf   STATECNTR, F
    goto   MAIN
BEGIN1
    btfsc  RXDATA
    goto   MAIN
    call   SETWATCH
    incf   STATECNTR, F
    goto   MAIN
HEADER
    btfsc  RXDATA
    goto   RESTART
    btfss  TMRHIGH, 0
    goto   MAIN
    movlw  D'64'
    andwf  TMRLOW, W
    btfsc  STATUS, Z
    goto   MAIN
    incf   STATECNTR, F

    goto   MAIN
    HEADER1
    movlw  D'6'
    subwf  TMRHIGH, W
    btfss  STATUS, C
    goto   HEADER2
    goto   RESTART
HEADER2
    btfss  RXDATA
    goto   MAIN
    call   SETWATCH
    movlw  D'32'
    movwf  BITCNTR
    incf   STATECNTR, F
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    goto   MAIN
HIGHPLSE
    btfsc  TMRHIGH, 0
    goto   RESTART
    btfsc  RXDATA
    goto   MAIN
    movf   TMRLOW, W
    movwf  HIGHWDTH
    call   SETWATCH
    incf   STATECNTR, F
    goto   MAIN
LOWPULSE
    btfsc  TMRHIGH, 0
    goto   LOW2
    btfss  RXDATA
    goto   MAIN
    movf   TMRLOW, W
    movwf  LOWWDTH
    call   SETWATCH
    incf   STATECNTR, F
    goto   MAIN
LOW2
    movlw  HEADR
    movwf  STATECNTR
    goto   MAIN
RECORD
    movf   HIGHWDTH, W
    subwf  LOWWDTH, W
    rrf    DATA3, F
    rrf    DATA2, F
    rrf    DATA1, F
    rrf    DATA0, F
    movlw  HIGHP
    movwf  STATECNTR
    decfsz BITCNTR, F
    goto   MAIN
    movlw  D'4'
    movwf  COUNTR
    movlw  DATA0
    movwf  FSR
RECORD1
    movlw  0xFF
    xorwf  INDF, W
```

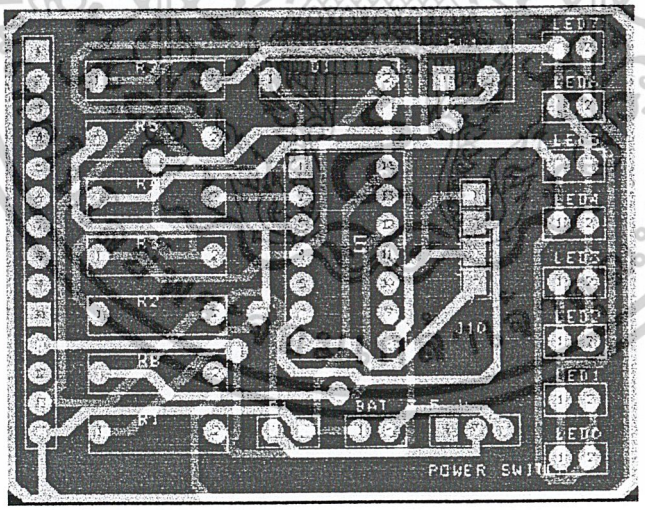
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    btfss  STATUS, Z
    goto   RECORD2
    incf   FSR, F
    decfsz COUNTR, F
    goto   RECORD1
    goto   RESTART
RECORD2
    movlw  WAIT
    movwf  STATECNTR
    goto   MAIN
WAIT4END
    btfsc  HLFLG
    goto   WAIT1
    btfsc  RXDATA
    goto   MAIN
    call   SETWATCH
    bsf    HLFLG
WAIT1
    btfss  RXDATA
    goto   WAIT2
    bcf    HLFLG
    goto   MAIN
WAIT2
    btfss  TMRHIGH, 3
    goto   MAIN
    bcf    HLFLG
    incf   STATECNTR, F
    goto   MAIN

VALIDATE
    movlw  0x00
    call   DATA_EEPROM_READ
    xorwf  DATA0, W
    btfss  STATUS, Z
    goto   RESTART
    movlw  0x01
    call   DATA_EEPROM_READ
    movwf  TEMP
    bsf    STATUS, RP0
    movf   EEDATA, W
    bcf    STATUS, RP0
    xorwf  TEMP, F
    btfsc  TEMP, 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    goto   MAIN
IMPLEMNT
    movfw  DATA3
    movwf  LEDREG
    clrf   SX1TMR
    clrf   SX2TMR
    goto   RESTART
RESTART
    movlw  BEGN
    movwf  STATECNTR
    goto   MAIN

 ORG   0x2100
    DE    0x73, 0x43, 0x02, 0x03
 org   0x3ff
    retlw  0x40

    End
```

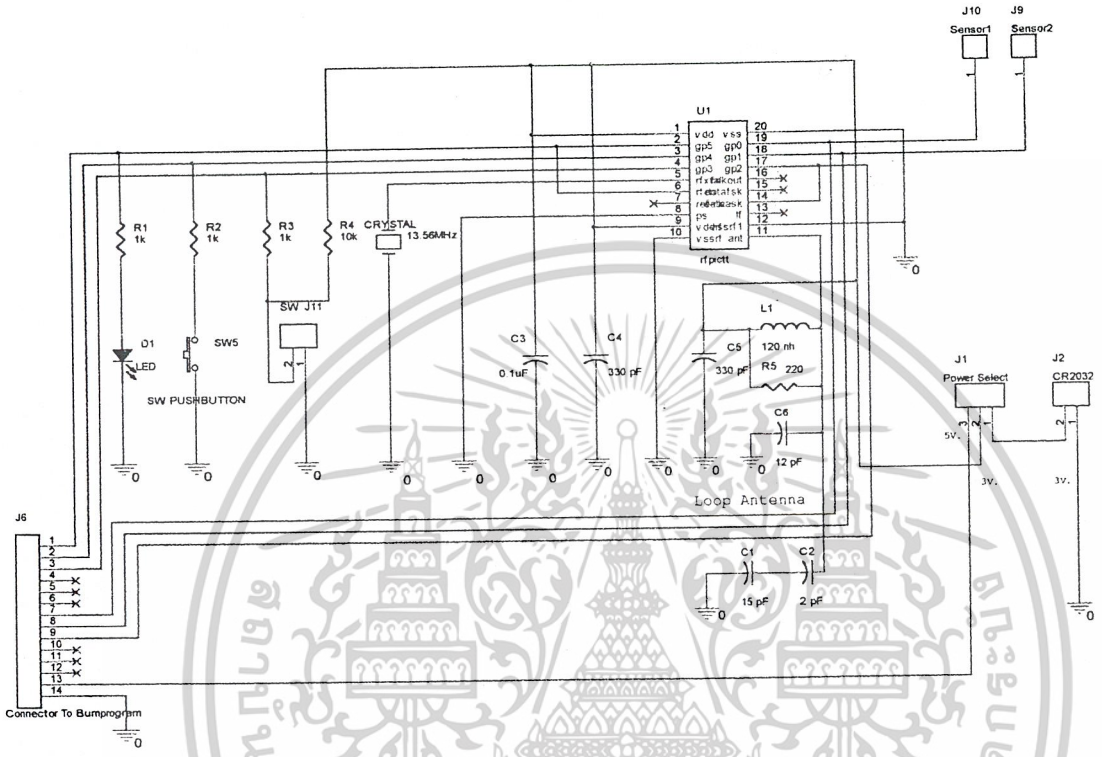
3. PCB LAYOUT



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก (ค.) ภาคส่ง

1. วงจรภาคส่ง



2. โปรแกรมอินเตอร์เฟส

```
 list      p=12f675
#include   <p12f675.inc>
errorlevel  -302
__CONFIG   _CPD_OFF & _CP_OFF & _BODEN_OFF & _MCLRE_OFF & _PWRTE_OFF & _WDT_OFF &
_INTRC_OSC_NOCLKOUT
cblock  0x20
  w_temp
  status_temp
  TEMP
  CSR0
  CSR1
  CSR2
  CSR3
  CSR4
  CSR5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    CSR7
    CSR8
    Count
    Count2
    BitCount
    TimeHi
    TimeLo
    DAT
    FuncBits
  endc
#define POT0   GPIO, 0
#define POT1   GPIO, 1
#define TXD    GPIO, 2
#define PB3    GPIO, 3
#define PB4    GPIO, 4
#define RFENA  GPIO, 5
#define GPTRIS  B'00011011'
#define TGUARD  D'46'
#define PREAMB  D'16'
    ORG    0x000
    nop
    goto   INITALIZE
    ORG    0x004
    movwf  w_temp
    swapf  STATUS, W
    bcf    STATUS, RP0
    movwf  status_temp
    movfw  GPIO
    bcf    INTCON, GPIF
    swapf  status_temp, W
    movwf  STATUS
    swapf  w_temp, F
    swapf  w_temp, W
    retfie
DATA_EEPROM_READ
    bsf    STATUS, RP0
    movwf  EEADR
    bsf    EECON1, RD
    movf   EEDATA, W
    bcf    STATUS, RP0
    return
DATA_EEPROM_WRITE
    bsf    STATUS, RP0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    bsf   EECON1, WREN
    bcf   INTCON, GIE
    movlw  0x55
    movwf  EECON2
    movlw  0xAA
    movwf  EECON2
    bsf   EECON1, WR
    bsf   INTCON, GIE
    bcf   EECON1, WREN
    bcf   STATUS, RP0
    return
READ_ANALOG_AN0
    bcf   ADCON0, CHS1
    bcf   ADCON0, CHS0
    goto   READ_ANALOG
READ_ANALOG_AN1
    bcf   ADCON0, CHS1
    bsf   ADCON0, CHS0
READ_ANALOG
    bsf   ADCON0, ADON
    movlw  D'6'
    movwf  TEMP
    decfsz  TEMP, F
    goto   $-1
    bsf   ADCON0, GO
    btfsc  ADCON0, GO
    goto   $-1
    bcf   ADCON0, ADON
    return
WaitxTE
    movwf  Count2
waitxlp
    movlw  D'79'
    movwf  Count
wait400lp
    nop
    nop
    decfsz  Count,F
     goto   wait400lp
    decfsz  Count2,F
     goto   waitxlp
    retlw  0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
INITALIZE
    bcf   INTCON, GIE
    bsf   STATUS, RP0
    call  0x3FF
    movwf OSCCAL
    bcf   STATUS, RP0
  bsf   STATUS, RP0
    movlw GPTRIS
    movwf TRISIO
    bcf   STATUS, RP0
    bsf   CMCON, CM2
    bsf   CMCON, CM1
    bsf   CMCON, CM0
    bsf   STATUS, RP0
    bcf   VRCON, VREN
    bcf   STATUS, RP0
    bcf   ADCON0, ADFM
    bcf   ADCON0, VCFG
    bcf   ADCON0, ADON
    bsf   STATUS, RP0
    bcf   ANSEL, ADCS2
    bcf   ANSEL, ADCS1
    bsf   ANSEL, ADCS0
    bcf   ANSEL, ANS3
    bcf   ANSEL, ANS2
    bsf   ANSEL, ANS1
    bsf   ANSEL, ANS0
    bcf   STATUS, RP0
    bcf   T1CON, TMR1ON
    bcf   T1CON, TMR1CS
    bcf   T1CON, NOT_T1SYNC
    bcf   T1CON, T1OSCEN
    bcf   T1CON, T1CKPS1
    bcf   T1CON, T1CKPS0
    bcf   T1CON, TMR1GE
    bsf   STATUS, RP0
    movlw B'00010000'
    movwf WPU
    bcf   STATUS, RP0
    bsf   STATUS, RP0
    bcf   OPTION_REG, NOT_GPPU
    bsf   OPTION_REG, INTEDG
    bcf   OPTION_REG, T0CS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    bcf   OPTION_REG, PSA
    bcf   OPTION_REG, PS2
    bcf   OPTION_REG, PS1
    bcf   OPTION_REG, PS0
    bcf   STATUS, RP0
    bsf   STATUS, RP0
    movlw  b'00011000'
    movwf  IOCB
    bcf   STATUS, RP0
    bsf   STATUS, RP0
    bcf   PIE1, EEIE
    bcf   PIE1, ADIE
    bcf   PIE1, CMIE
    bcf   PIE1, TMR1IE
    bcf   STATUS, RP0
    bcf   INTCON, PEIE
    bcf   INTCON, T0IE
    bcf   INTCON, INTE
    bsf   INTCON, GPIE
MAIN
    CLRF   DAT
    bcf   RFENA
    bsf   INTCON, GIE
SCANPB
    movlw  0x00
    movwf  FuncBits
    btfsc  PB3
    goto  SPB1
    movlw  0x23
    iorwf  FuncBits, F
    call   READ_ANALOG_AN0
SPB1
    btfsc  PB4
    goto  SPB2

    movlw  0x43
    iorwf  FuncBits, F
    call   READ_ANALOG_AN1

SPB2
    movlw  0xFF
    andwf  FuncBits, W
    btfss  STATUS, Z
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

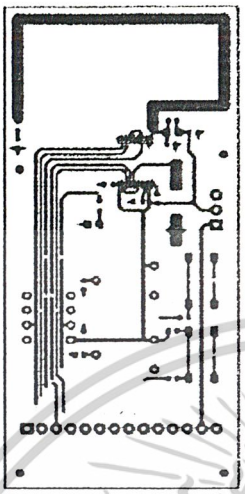
```
        goto   XMIT
        bcf    RFENA
        sleep
        goto   SCANPB
XMIT
        bsf    RFENA
        movlw  0x73
        movwf  CSR0
        movfw  FuncBits
        movwf  CSR1
        bsf    STATUS, RP0
        movfw  ADRESL
        bcf    STATUS, RP0
        movwf  CSR2
        movlw  0xAA
        movwf  CSR3
        movlw  0x56
        movwf  CSR4
        movlw  0x34
        movwf  CSR5
        movlw  0x12
        movwf  CSR6
        movlw  0x20
        movwf  CSR7
        movlw  0x55
        movwf  CSR8
TXLoop
Preamble
        movlw  PREAMB
        movwf  BitCount
PreL
        bsf    TXD
        movlw  1
        call   WaitxTE
        bcf    TXD
        movlw  1
        call   WaitxTE
        decfsz BitCount,F
        goto   PreL
TXloop
        movlw  D'10'
        call   WaitxTE
        movlw  CSR0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        movwf   FSR
TXNextByte
        movlw   D'8'
        movwf   BitCount
TXNextBit
        rrf     INDF,W
        rrf     INDF,F
        BC      ONE
ZERO
        movlw   2
        movwf   TimeHi
        movlw   1
        movwf   TimeLo
        goto    Trasm_BIT
ONE
        movlw   1
        movwf   TimeHi
        movlw   2
        movwf   TimeLo
Trasm_BIT
        bsf     TXD
        movf    TimeHi,W
        call    WaitxTE
        bcf     TXD
        movf    TimeLo,W
        call    WaitxTE
        decfsz  BitCount,F
        goto    TXNextBit
        incf    FSR,F
        movlw   CSR8+1
        xorwf   FSR,W
        andlw   0x1F
        BNZ     TXNextByte
        movlw   TGUARD
        call    WaitxTE
        goto    SCANPB
        org     0x3ff
        retlw   0x80
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. PCB LAYOUT



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้