แจ้งเตือนเมื่อใช้พลังงานเกิน

WARNING FOR EXCESSIVED ELECTRICAL POWER

ฉัตรชัย พันโท
ฉัตรชัย อินโต
พงศ์ชัย จิวะวงศ์สกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2555

แจ้งเตือนเมื่อใช้พลังงานเกิน
WARNING FOR EXCESSIVED ELECTRICAL POWER

ฉัตรชัย พันโท
ฉัตรชัย อินโต
พงศ์ชัย จิวะวงศ์สกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WARNING FOR EXCESSIVED ELECTRICAL POWER

CHATCHAI PHUNTO
CHATCHAI INTO
PONGCHAI JIWAWONGSAKUL

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERRING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUE OF TECHNOLOGY LARDKRABANG
ACADEMIC YEAR 2012

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2555
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์	แจ้งเตือนเมื่อใช้พลังงานเกิน WARNING FOR EXCESSIVED ELECTRICAL POWER			
นักศึกษาผู้จัดทำ	นายฉัตรชัย	พันโท	**รหัสนักศึกษา**	52010187
	นายฉัตรชัย	อินโต	**รหัสนักศึกษา**	52010189
	นายพงศ์ชัย	จิวะวงศ์สกุล	**รหัสนักศึกษา**	52010751
ปริญญา	วิศวกรรมศาสตรบัณฑิต			
สาขาวิชา	วิศวกรรมการวัดคุม			
ปีการศึกษา	2555			

อาจารย์ผู้ควบคุมปริญญานิพนธ์	**ลายมือชื่อ**
รศ.ดร.ฟูศักดิ์ ชีวสุวิทย์	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	แจ้งเตือนเมื่อใช้พลังงานเกิน			
	WARNING FOR EXCESSIVED ELECTRICAL POWER			
นักศึกษาผู้จัดทำ	นายฉัตรชัย	พันโท	**รหัสนักศึกษา**	52010187
	นายฉัตรชัย	อินโต	**รหัสนักศึกษา**	52010189
	นายพงศ์ชัย	จิวะวงศ์สกุล	**รหัสนักศึกษา**	52010751
อาจารย์ที่ปรึกษา	รศ.ดร.ฟูศักดิ์	ชีวสุวิทย์		
ปีการศึกษา	2555			

บทคัดย่อ

โครงงานนี้จัดทำขึ้นเพื่อนำระบบเครือข่ายไร้สายมาประยุกต์ใช้ในการจัดการพลังงาน โดยข้อดีของการนำระบบเครือข่ายไร้สายมาประยุกต์ใช้คือ สามารถลดค่าใช้จ่ายในการเดินสายเกี่ยวกับการสื่อสารข้อมูล และการติดตั้งอุปกรณ์เครื่องใช้ไฟฟ้าต่างๆ รวมถึงการรับส่งข้อมูลในเครือข่ายไร้สายสามารถทำได้สะดวก ส่วนอุปกรณ์ภาครับและภาคส่งที่นำมาประยุกต์ใช้ในโครงงานนี้คือ ซิกบี โดยมีการส่งสัญญาณไร้สายผ่านซิกบี เพื่อตรวจวัดปริมาณการใช้พลังงานไฟฟ้าภายในบ้าน และทำการควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้า ผ่านการแสดงผลหน้าจอคอมพิวเตอร์ ซึ่งจะช่วยให้ผู้อยู่อาศัยประเมินการใช้พลังงานไฟฟ้าของตนเองได้สะดวกมากยิ่งขึ้น

ผู้จัดทำโครงงานได้ทำการออกแบบ และเลือกใช้ Current Sensor เบอร์ ACS714 เป็นตัววัดค่ากระแสไฟฟ้าและเลือกใช้ไมโครคอนโทรลเลอร์ PIC18F458 เป็นตัวโปรแกรมคำนวณค่าทางไฟฟ้าซึ่งค่าที่สามารถวัดได้ มีดังนี้ กระแส แรงดัน และกำลังไฟฟ้าจริงที่ค่อนข้างแม่นยำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	WARNING FOR EXCESSIVED ELECTRICAL POWER	
Authors	Mr. Chatchai	Phunto
	Mr. Chatchai	Into
	Mr. Pongchai	Jiwawongsakull
Thesis Advisor	Mr. Fusak	Cheewasuwit
Year	2012	

ABSTRACT

This project presents an energy management by using wireless network. The wireless network can save the cost of cable installation for electrical appliances, and also for data transmission. In this project, the quantity of electrical energy for each controlled appliances will be transmitted through the ZigBee. The total quantity will be shown by the monitor for house owner to estimate the used of electrical energy. Here, the current sensor ACS714 is used to measure the consumed current, while the microcontroller PIC 18F458 in use for calculating the electrical power.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี เนื่องจากได้รับการสนับสนุนจากบุคคลหลายฝ่ายที่ให้คำปรึกษา และชี้แนะแนวทาง ทำให้ปริญญานิพนธ์ฉบับนี้บรรลุเป้าหมายเป็นไปตามวัตถุประสงค์ได้เป็นอย่างดี

ขอขอบคุณ รศ.ดร.ฟูศักดิ์ ชีวสุวิทย์ ที่กรุณาสละเวลา มาเป็นอาจารย์ที่ปรึกษาโครงงานนี้ ซึ่งได้ให้ความรู้ คำแนะนำ และข้อคิดเห็นเพิ่มเติม ตลอดจนให้ความช่วยเหลือต่างๆ ในการดำเนินงานเกี่ยวกับการพัฒนาโครงงานมาโดยตลอด จนกระทั่งเสร็จสมบูรณ์

ขอขอบคุณ ภาควิชาการวัดและควบคุม หลักสูตรวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่เอื้อเฟื้อสถานที่ และเครื่องมือวัสดุอุปกรณ์ต่างๆ สำหรับการพัฒนาโครงการนี้

ขอขอบคุณ เพื่อนๆ ที่ให้ความช่วยเหลือในการพัฒนาโครงงาน

สุดท้ายนี้ ขอขอบคุณ ผู้ที่มีส่วนเกี่ยวข้องทุกๆ ท่าน ที่ได้ให้ความช่วยเหลือให้โครงงานนี้สำเร็จลุล่วงไปได้ด้วยดี ผู้จัดทำโครงงาน จึงขอขอบคุณเป็นอย่างสูง มา ณ โอกาสนี้

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I
บทคัดย่อภาษาอังกฤษ II
กิตติกรรมประกาศ III
สารบัญ IV
สารบัญตาราง VII
สารบัญภาพ VIII

บทที่ 1 บทนำ 1
1.1 ความสำคัญของปริญญานิพนธ์ 1
1.2 วัตถุประสงค์ของปริญญานิพนธ์ 1
1.3 ขอบเขตของปริญญานิพนธ์ 1
1.4 ขั้นตอนการศึกษา 1
1.5 ประโยชน์ที่คาดว่าจะได้รับ 2

บทที่ 2 ทฤษฎี 3
2.1 Voltage and Current Sensor 3
2.1.1 Voltage Divider 3
2.1.2 เซ็นเซอร์ตรวจจับกระแส (Current Sensor) 3
2.2 ทฤษฎีกำลังไฟฟ้า 5
2.3 ทฤษฎีเกี่ยวกับการสื่อสารและรับส่งข้อมูล 5
2.3.1 เทคโนโลยีซิกบี (ZigBee) 5
2.3.2 เอ็กซ์บี (XBee) 9
2.3.3 ZX-XBeeU บอร์ดเชื่อมต่อโมดูล XBee-PRO กั บคอมพิวเตอร์ 15
2.3.4 ADX-XBee บอร์ดเชื่อมต่อโมดูล XBee-PRO กับอุปกรณ์ภายนอก 17
2.4 ไมโครคอนโทรลเลอร์ PIC18F458 18
2.4.1 คุณสมบัติของไมโครคอนโทรลเลอร์ PIC18F458 18
2.4.2 การจัดขาของ PIC18F458 20
2.4.3 สถาปัตยกรรมและโครงสร้างของไมโครคอนโทรลเลอร์ PIC18F458 23
2.4.4 โหมดสัญญาณนาฬิกา 25
2.4.4.1 การป้อนสัญญาณนาฬิกาโดยใช้คริสตอล 25
2.4.4.2 โหมดการป้อนสัญญาณนาฬิกาตัวต้านทาน และตัวเก็บประจุ RC และ RCIO 26
2.4.4.3 โหมดการป้อนสัญญาณนาฬิกาแบบ EC และ ECIO 26
2.4.4.4 โหมดการป้อนสัญญาณนาฬิกาโดยใช้วงจร LPL 27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.4.5 การรีเซตของไมโครคอนโทรลเลอร์ PIC18F458 28
2.4.5.1 เพาเวอร์ออนรีเซต (Power On Reset : POR) 28
2.4.5.2 เพาเวอร์อัปไทเมอร์ 28
2.4.5.3 ออสซิลเลเตอร์สตาร์ตอัปไทเมอร์ 28
2.4.5.4 บราวเอาต์รีเซต 28
2.4.6 ลำดับการเกิดไทม์เอาต์ (Time Out Squence) 29
2.4.7 หน่วยความจำ 29
2.4.7.1 หน่วยความจำโปรแกรม 30
2.4.7.2 สแต็ก (Stack) 31
2.4.7.3 รีจิสเตอร์โปรแกรมเคาน์เตอร์ PCL, PCLATH และ PCLATL 31
2.4.7.4 จังหวะและไซเคิลการทำงานของ PIC18F458 31
2.4.7.5 หน่วยความจำข้อมูล (Data Memory) 32
2.5 รีเลย์ (Relay) 33
2.5.1 ชนิดของรีเลย์ 33
2.5.2 รีเลย์ (HLS8L-DC6V-S-C) 34

บทที่ 3 วิธีดำเนินงาน 36
3.1 ภาพรวมของการวางแผนโครงสร้างของโครงงาน 36
3.2 ส่วนการทำงานโดยรวม 37
3.2.1 ส่วนการวัดพลังงานไฟฟ้า (Measuring Section) 37
3.2.1.1 Voltage Divider 37
3.2.1.2 Current Sensor 38
3.2.1.3 ไมโครคอนโทรลเลอร์ 38
3.2.2 ส่วนควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้า 39
3.2.2.1 Relay 39
3.2.2.2 ไมโครคอนโทรลเลอร์ 39
3.2.3 ส่วนการแสดงผลและติดต่อกับผู้ใช้ 39
3.2.4 ส่วนการสื่อสารและรับส่งข้อมูล 40
3.2.4.1 เอ็กซ์บี (XBee) 40
3.2.4.2 ADX-XBee3V บอร์ดใช้งานโมดูล XBee กับอุปกรณ์ 40
3.2.4.3 ZX-XBeeU บอร์ดเชื่อมต่อคอมพิวเตอร์ 41
3.3 การออกแบบวงจร 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 4 การทดลองและผลการทดลอง **43**

4.1 การสอบเทียบ 43

4.1.1 สมการแก้ไขค่าผิดพลาด IRMS 48

4.1.2 สมการแก้ไขค่าผิดพลาด VRMS 48

4.2 ผลจากการสอบเทียบค่าและปรับแต่งค่าด้วยสถิติ 49

4.3 การควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้าภายในบ้านผ่าน XBee 50

4.4 การแสดงผลผ่านโปรแกรม VISUAL STUDIO บนหน้าจอคอมพิวเตอร์ 51

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ **51**

5.1 สรุปผลการดำเนินงาน 51

5.2 ปัญหาที่พบในการทำวิจัย 51

5.3 ข้อเสนอแนะ 51

บรรณานุกรม **52**

ภาคผนวก **53**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

2.1 ตารางแสดงคุณสมบัติและรายละเอียดขาต่างๆของ PIC18F458 20
2.2 ตารางแสดงค่าของตัวเก็บประจุที่ใช้กับคริสตอลออสซิลเลเตอร์ 26
2.3 ตารางแสดงค่าเวลาต่างๆ ของแต่ละส่วนก่อนการเกิดไทม์เอาต์ 29
4.1 ตารางแสดงค่าเปรียบเทียบค่าพลังงานทางไฟฟ้าระหว่างค่าจริงกับค่าที่วัดได้ 45
4.2 ตารางแสดงค่า หลังจากการสอบเทียบค่าด้วยสถิติ ทำให้ได้ค่ามีถูกต้อง 49
4.3 ตารางแสดงการควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้า 51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

2.1 แสดง Voltage Divider 3
2.2 แสดง ACS714 Current Sensor Carrier -5 to +5A 4
2.3 แสดงหน้าที่การทำงานของขาต่างๆ ของ ACS714 4
2.4 แสดงวงจรต่อใช้งานของ ACS714 4
2.5 กราฟแสดง Power Triangle 5
2.6 แสดงโครงสร้างซอฟท์แวร์ของ ซิกบี (ZigBee) 6
2.7 แสดงชั้น Application ของ ซิกบี (ZigBee) 6
2.8 ย่านความถี่ใช้งาน ซิกบี (ZigBee) 7
2.9 มาตรฐานโปรโตคอลสื่อสารไร้สาย 8
2.10 แสดงส่วนประกอบต่างๆ ของซิกบี (ZigBee) 9
2.11 Star (Broadcast) Network 10
2.12 Cluster Tree (Tree) Network 11
2.13 Mesh Network 11
2.14 แสดงการจัดขาของ XBee 13
2.15 แผนภาพการไหลภายในข้อมูล 14
2.16 รายละเอียดของ ZX-XBeeU บอร์ดเชื่อมต่อโมดูล XBee/XBee-PRO กับคอมพิวเตอร์ 15
2.17 วงจรของ ZX-XBeeU บอร์ดเชื่อมต่อโมดู ล XBee/XBee-PRO กับคอมพิวเตอร์ 16
2.18 รายละเอียดของ ADX-XBee บอร์ดเชื่อมต่อโมดูล XBee-PRO กับอุปกรณ์ภายนอก 17
2.19 แสดงวงจรของบอร์ด ADX-XBee ทั้ง 2 รุ่นคือ ADX-XBee5V สำหรับระบบ +5V 17
และ ADX-XBee3V สำหรับระบบ +3.3 หรือ +3V
2.20 รูปแสดงการจัดวางขาของ PIC18F458 20
2.21 รูปแสดงสถาปัตยกรรมของไมโครคอนโทรลเลอร์ PIC18F458 24
2.22 รูปแสดงการต่อคริสตอลแบบเรโซเนเตอร์ 25
2.23 รูปแสดงการต่อออสซิลเลเตอร์แบบ RC และ RCIO 26
2.24 แสดงการเชื่อมต่อออสซิลเลเตอร์แบบ EC และ ECIO 27
2.25 แสดงบล็อกไดอะแกรมการทำงานของวงจร PLL (Phase Lock Loop) 27
2.26 แสดงการจัดสรรหน่วยความจำโปรแกรมของ PIC18F458 30
2.27 แสดงการจัดสรรหน่วยความจำข้อมูลของ PIC18F458 32
2.28 แสดงตัวอย่างรีเลย์ควบคุมแบบต่างๆ 33
2.29 แสดงตัวอย่างคอนแทกเตอร์แบบต่างๆ 33
2.30 แสดงตำแหน่งขาต่างๆ ของรีเลย์ (HLS8L-DC6V-S-C) 34
2.31 แสดงสัญลักษณ์ของรีเลย์ 34
2.32 แสดงรีเลย์ขณะที่ไม่จ่ายกระแสไฟฟ้าแก่ขดลวดรีเลย์ 35
2.33 แสดงรีเลย์ขณะจ่ายกระแสไฟฟ้าแก่ขดลวดรีเลย์ 35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

3.1 แสดงโครงสร้างของอุปกรณ์ต่างๆ 36
3.2 วงจร Voltage Divider และสูตรการคำนวณ 37
3.3 Current Sensor เบอร์ ACS714 และโครงสร้าง Pin Configuration ที่ใช้งาน 38
3.4 ไมโครคอนโทรลเลอร์ 38
3.5 ภาพแสดงตำแหน่งขาต่างๆ ของรีเลย์ และสัญลักษณ์ของรีเลย์ 39
3.6 คอมพิวเตอร์แสดงผลและติดต่อกับผู้ใช้ 39
3.7 แสดงลักษณ์ของเอ็กซ์บี และการจัดขาของเอ็กซ์บี 40
3.8 แสดง ADX-XBee3V บอร์ดใช้งานโมดูล XBee กับอุปกรณ์ 40
3.9 แสดง ZX-XBeeU บอร์ดเชื่อมต่อคอมพิวเตอร์ 41
3.10 แสดงวงจรการทำงานโดยรวม 41
3.11 แสดงวงจรการทำงานของ Relay 42
4.1 การต่ออุปกรณ์เพื่อทดสอบการทำงานของเครื่องมือ 43
4.2 โหลดที่ใช้ในการทดสอบเครื่องมือวัด 44
4.3 การอ่านข้อมูลมาแสดงผลผ่าน SERIAL PORT บนหน้าจอคอมพิวเตอร์ 44
4.4 กราฟแสดงค่า IRMS จากอุปกรณ์ตรวจวัดที่อ่านค่าได้ 46
4.5 กราฟแสดงค่า IRMS จากเครื่องสอบเทียบที่อ่านค่าได้ 46
4.6 กราฟแสดงค่า VRMS จากอุปกรณ์ตรวจวัดที่อ่านค่าได้ 47
4.7 กราฟแสดงค่า VRMS จากเครื่องสอบเทียบที่อ่านค่าได้ 47
4.8 แสดงการทำงานของอุปกรณ์ตรวจวัดร่วมกับสวิตช์ 50
4.9 แสดงภาพโปรแกรม VISUAL STUDIO ติดต่อกลับผู้ใช้งาน 51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญของปริญญานิพนธ์

ในปัจจุบัน มนุษย์มีความสนใจในความเป็นอยู่ที่สะดวกสบายมากขึ้น มีความต้องการที่จะควบคุมการใช้พลังงานภายในบ้านเพื่อให้เกิดประโยชน์สูงสุด เทคโนโลยีจึงได้เข้ามามีบทบาทสำคัญในชีวิตประจำวันมากขึ้น เดิมทีการส่งผ่านข้อมูลระหว่างสองแห่งจะต้องเชื่อมต่อผ่านสาย แต่ในปัจจุบันนี้มีการพัฒนาโดยไม่ต้องใช้สายแล้ว ด้วยเหตุนี้จึงมีการนำเทคโนโลยีการสื่อสารแบบไร้สายมาประยุกต์ใช้งานในการควบคุมอุปกรณ์ภายในบ้าน แต่ในปัจจุบันอุปกรณ์ประเภทนี้ยังมีราคาแพง โครงงานนี้จึงมีแนวคิดในการพัฒนาอุปกรณ์ในการควบคุมพลังงานภายในบ้านโดยใช้ซิกบี ซึ่งเป็นอุปกรณ์สื่อสารไร้สาย ด้วยคุณสมบัติของซิกบี ที่มีประสิทธิภาพสูง ใช้พลังงานต่ำ ราคาถูก ใช้งานได้ยาวนาน จึงเป็นทางเลือกใหม่ในการควบคุมพลังงานภายในบ้าน

ระบบแจ้งเตือนเมื่อใช้พลังงานเกินนี้ จึงจัดทำเพื่อทำการตรวจวัดพลังงานทางไฟฟ้าภายในบ้าน ควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้าจากระยะไกลร่วมกับสวิตช์ ซึ่งการตรวจวัดพลังงานทางไฟฟ้าจะเป็นการวัดค่ากระแสไฟฟ้า แรงดันไฟฟ้าที่ใช้งานจริง และแสดงผลค่าพลังงานทางไฟฟ้าที่วัดได้ผ่านทางคอมพิวเตอร์ อีกทั้งยังสามารถตั้งค่าพลังงานไฟฟ้ามากสุดไว้สำหรับการแจ้งเตือนได้อีกด้วย

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อศึกษาคุณสมบัติ การใช้งาน และระบบการส่งผ่านข้อมูลของซิกบี
2. เพื่อศึกษาการทำงานของไมโครคอนโทรลเลอร์ ตระกูล PIC ที่ต่อใช้งานร่วมกับซิกบี
3. ศึกษาการทำงานของอุปกรณ์ไฟฟ้า และควบคุมการทำงานของอุปกรณ์ไฟฟ้า
4. ศึกษาการวัดแรงดันแบบ Voltage Divider และการทำงานของ Current Sensor
5. ศึกษาการตรวจวัดพลังงานทางไฟฟ้าภายในบ้าน

1.3 ขอบเขตของปริญญานิพนธ์

1. สามารถสั่งการควบคุมการเปิด-ปิดการใช้งานของอุปกรณ์ไฟฟ้าต่างๆ แบบไร้สายโดยใช้ซิกบี
2. สามารถตรวจวัดค่าพลังงานทางไฟฟ้าภายในบ้าน และนำค่าพลังงานทางไฟฟ้า แสดงผ่านคอมพิวเตอร์
3. สามารถตั้งค่าพลังงานไฟฟ้าสูงสุดที่ใช้งาน จากคอมพิวเตอร์เพื่อแจ้งเตือนเมื่อพลังงานไฟฟ้าเกินได้

1.4 ขั้นตอนการศึกษา

1. ศึกษาหลักการทำงานเบื้องต้นของซิกบี
2. ศึกษาหลักการทำงานของไมโครคอนโทรลเลอร์ ตระกูล PIC
3. ศึกษาหลักการทำงานของเซ็นเซอร์ตรวจจับกระแส
4. ศึกษาการเขียนโปรแกรมควบคุมสั่งการ ไมโครคอนโทรลเลอร์ ตระกูล PIC ด้วยภาษา C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ศึกษาหลักการส่งข้อมูลจาก ไมโครคอนโทรลเลอร์ ตระกูล PIC ไปยัง Computer ผ่านทางซิกบี
6. เขียนโปรแกรมสำหรับตรวจวัดพลังงานทางไฟฟ้า ควบคุมการทำงานของอุปกรณ์ไฟฟ้าและแสดงผลค่าพลังงานทางไฟฟ้า
7. Run โปรแกรมทดสอบระบบ
8. วิเคราะห์การทำงานของระบบที่ใช้ควบคุมฮาร์ดแวร์ เพื่อหาข้อผิดพลาดและทำการแก้ไขปรับปรุง
9. ทำรายงาน และสรุปผลการทำงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ศึกษาการเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์ด้วยภาษา C
2. ได้ศึกษาและเรียนรู้การออกแบบลายวงจรและโปรแกรมการออกแบบลายวงจร
3. สามารถสร้างเครื่องมือวัดพลังงานทางไฟฟ้าและแจ้งเตือนเมื่อใช้พลังงานไฟฟ้าเกินที่มีคุณลักษณะตามที่ต้องการได้
4. มีพื้นฐานความรู้ที่สามารถไปต่อยอดทางด้านเครื่องมือวัดพลังงานทางไฟฟ้าและการจัดการทางด้านฐานข้อมูลด้วยไมโครคอนโทรเลอร์เป็นอย่างดี
5. ได้เรียนรู้การทำงานเป็นทีม และเรียนรู้กับการจัดการปัญหาที่เกิดขึ้นระหว่างที่ทำโครงงาน
6. คาดหวังว่าโครงงานนี้จะช่วยให้ผู้ใช้งานมีความสะดวกสบายมากขึ้นและสามารถลดการใช้พลังงานภายในบ้านได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

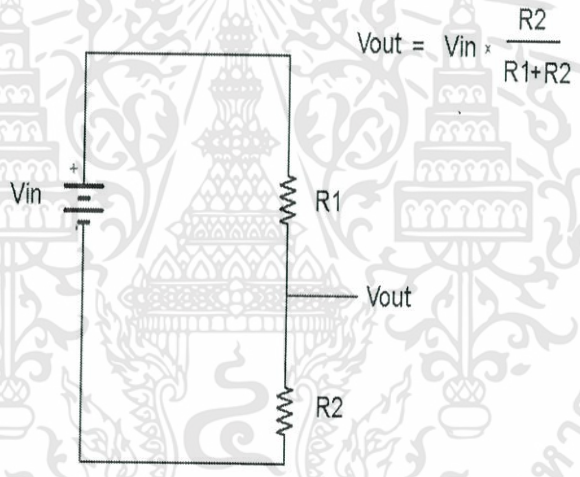
บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 Voltage and Current Sensor

เป็นส่วนของการรับค่าแรงดันและกระแสไฟฟ้า ซึ่งก่อนจะทำการส่งค่าเหล่านี้ไปคำนวณในไมโครคอนโทรลเลอร์ ตระกูล PIC จำเป็นต้องลดขนาดของสัญญาณ และปรับแต่งค่าให้เหมาะสมกับไมโครคอนโทรลเลอร์ที่ใช้ ในที่นี้ ไมโครคอนโทรลเลอร์ ที่นำมาใช้ คือ PIC18F458

2.1.1 Voltage Divider

เป็นเทคนิคในการหารแรงดันลงมา ซึ่งวิธีนี้เป็นวิธีการที่ง่ายที่สุด แต่มีข้อเสียคือ วงจรจะไม่ Isolated จากแหล่งจ่ายไฟที่ผู้จัดทำโครงงาน ได้ทำการวัดค่า แรงดันที่เราได้ออกมาทาง Output นั้นจะเป็นสัดส่วนกับ Input ดังสมการ และรูปด้านล่าง

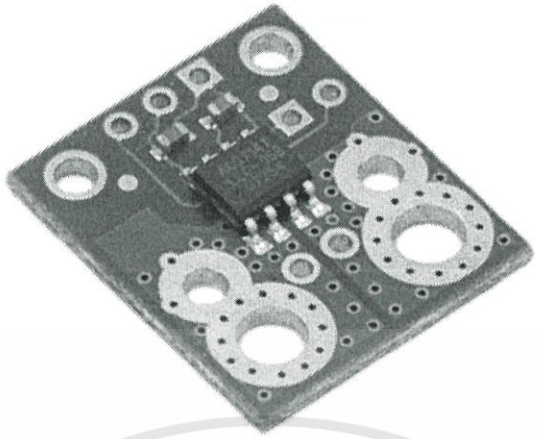


รูปที่ 2.1 แสดง Voltage Divider

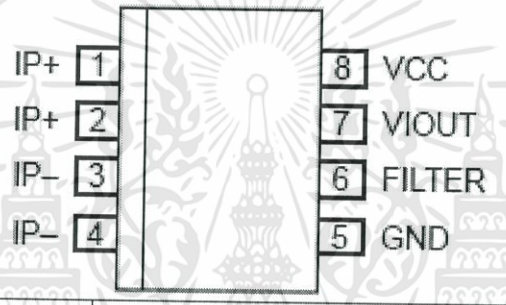
2.1.2 เซ็นเซอร์ตรวจจับกระแส (Current Sensor)

การวัดกระแสใช้เซนเซอร์วัดกระแสโดยนำเอาต์พุตจากเซนเซอร์วัดกระแสผ่านวงจรเรียงกระแสซึ่ง Current Sensor Carrier -5 to +5A เป็นเซนเซอร์ตรวจวัดกระแสใช้ IC เบอร์ ACS714 วัดได้ทั้งกระแสไฟ DC และ AC สามารถวัดกระแสที่ไหลได้ 2 ทิศทางทั้งค่าบวกและค่าลบ มีขนาด 5A (±5A) ให้ค่า Output เป็น Analog Voltage 185 mV/A โดยมี Center ที่ 2.5 V (มีค่า typical error น้อยกว่า 1.5%)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

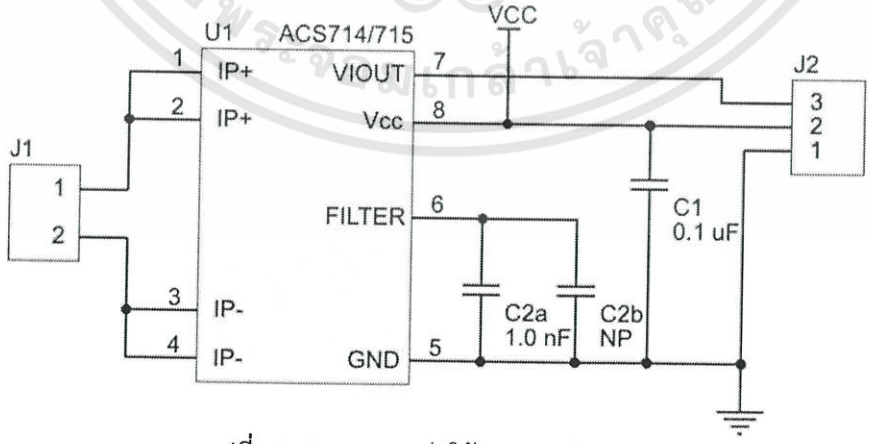


รูปที่ 2.2 แสดง ACS714 Current Sensor Carrier -5 to +5A



Number	Name	Description
1 and 2	IP+	Terminals for current being sampled; fused internally
3 and 4	IP–	Terminals for current being sampled; fused internally
5	GND	Signal ground terminal
6	FILTER	Terminal for external capacitor that sets bandwidth
7	VIOUT	Analog output signal
8	VCC	Device power supply terminal

รูปที่ 2.3 แสดงหน้าที่การทำงานของขาต่างๆ ของ ACS714

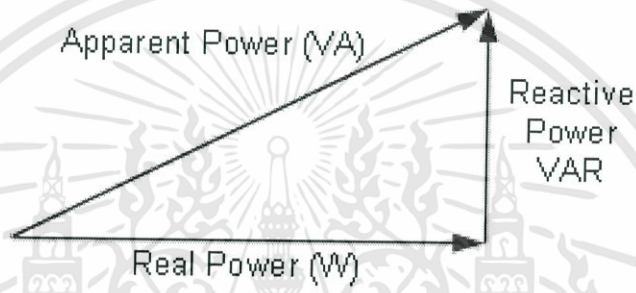


รูปที่ 2.4 แสดงวงจรต่อใช้งานของ ACS714

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ทฤษฎีกำลังไฟฟ้า

กำลังไฟฟ้า (Electrical Power) คือ การนำเอากระแสมาคูณกับแรงดัน อาจจะมีตัวประกอบกำลังมาคูณเพิ่ม ดังนั้น กำลังไฟฟ้าประกอบไปด้วย 3 ส่วนหลักๆ คือ 1. ค่ากำลังไฟฟ้าที่ปรากฏ (Apparent Power) คือ ค่าที่กระแสคูณ (Cross Product) กับแรงดัน, 2. ค่ากำลังไฟฟ้าจริง (Real Power) คือ ค่าที่กระแสคูณ (Cross Product) กับแรงดัน แล้วคูณด้วย Cosine ของมุมระหว่างแรงดันกับกระแสนั้นๆ กำลังไฟฟ้าสุดท้าย คือ ค่ากำลังไฟฟ้าประกอบ (Reactive Power) คือ ค่าที่กระแสคูณ (Cross Product) กับแรงดัน แล้วคูณด้วย Sine ของมุมระหว่างแรงดันกับกระแสนั้นๆ ซึ่งเรียก Cosine ของมุมระหว่างแรงดันกับกระแสว่า Power factor



รูปที่ 2.5 กราฟแสดง Power Triangle

2.3 ทฤษฎีเกี่ยวกับการสื่อสารและรับส่งข้อมูล

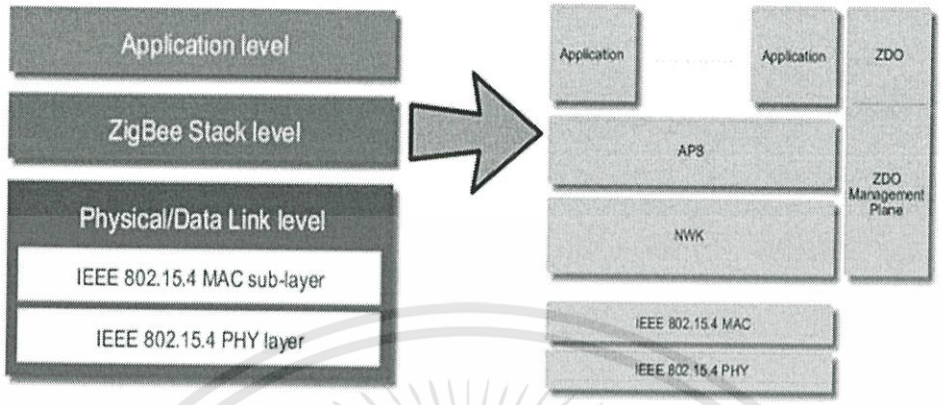
2.3.1 เทคโนโลยีซิกบี(ZigBee)

ซิกบี (ZigBee) เป็นการสื่อสารที่ออกแบบขึ้นสำหรับการสื่อสารในเครือข่ายเซ็นเซอร์แบบไร้สาย (Wireless Sensor Network) โดยเริ่มจากการกำหนดมาตรฐานการรับ-ส่งข้อมูลแบบ IEEE 802.15.4 ที่เน้นการสื่อสารแบบประหยัดพลังงาน ความเร็วการรับส่งข้อมูลต่ำและมีราคาถูก การสื่อสารลักษณะนี้ได้ถูกนำมาใช้สำหรับการสื่อสาร ระหว่างเครื่องตรวจวัดหรือเซ็นเซอร์ ที่ต้องการสื่อสารแบบไร้สายเพื่อลดความยุ่งยากซับซ้อนสำหรับการติดตั้ง

มาตรฐาน IEEE 802.15.4 มีจุดมุ่งหมายเบื้องต้นเพื่อตรวจจับและควบคุมแอพพลิเคชั่น ความสามารถในด้านการประหยัดพลังงานถือเป็นคุณสมบัติที่สำคัญในการที่จะทำให้อุปกรณ์ที่ใช้งานแบตเตอรี่สามารถใช้งานได้ยาวนานขึ้น จำนวนปริมาณข้อมูลเอาท์พุธ (แบนด์วิธ) ที่ได้ค่อนข้างต่ำเมื่อเทียบกับไวร์เลส แลน ซึ่งต้องใช้ถึง 250 กิโลบิตต่อวินาที (kbps) กับแอพพลิเคชั่นหลายตัวจึงจะมากเพียงพอ ระยะห่างระหว่างโหนด 2 โหนด อยู่ที่ 50 เมตรขึ้นไป และแต่ละโหนดที่แลกเปลี่ยนข้อมูลกันจะทำการสร้างเครือข่ายขนาดใหญ่ขึ้นเรื่อยๆ เพื่อให้ครอบคลุมเส้นทางที่เป็นไปได้ทั้งหมด

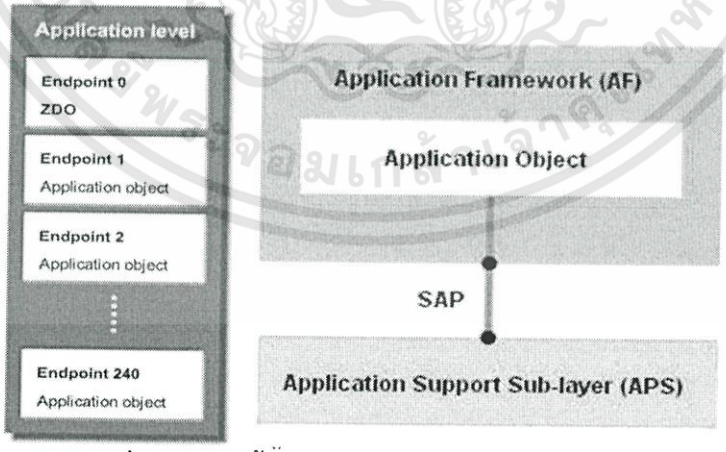
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **โครงสร้างซอฟท์แวร์**



รูปที่ 2.6 แสดงโครงสร้างซอฟท์แวร์ของ ซิกบี (ZigBee)

โครงสร้างซอฟท์แวร์ แสดง 3 ชั้นหลัก ชั้นล่าง PHY+MAC ว่าตาม IEEE802.15.4 (รับผิดชอบการรับส่งข้อมูลและ addressing) ZigBee Stack จัดเตรียมสมบัติหรือฟังก์ชั่นโครงสร้างเน็ตเวิร์กเส้นทางการส่งข้อมูล และการรักษาความปลอดภัย (อาจมองว่าคือ network layer) และ Application level ซึ่งเป็นตัวปฏิบัติการตาม app ที่ทำงานของแต่ละโหนด ถ้าแจกแจงให้ละเอียดกว่าเดิมสามารถแสดงโครงสร้างซอฟท์แวร์ของ ซิกบี ได้ดังรูปที่ 2.7 โดยทั่วไปโหนดอาจมีได้หลาย app และเราเรียก app instance บนโหนดว่า endpoint โดยที่ endpoint เป็นที่ที่ข้อความ (message) ถูกสร้างขึ้นหรือเป็นที่ปลายทางของข้อความ หนึ่งโหนดสามารถมีได้สูงสุด 240 endpoints แต่ละ endpoint ต้องมีหมายเลขกำกับ (address) ตั้งแต่ 1 ถึง 240 และที่อยู่หมายเลข 0 สงวนไว้สำหรับ app พิเศษตัวหนึ่งซึ่งเรียกว่า ZDO (ZigBee Device Objects)



รูปที่ 2.7 แสดงชั้น Application ของ ซิกบี (ZigBee)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SAP จะจัดเตรียมกลุ่มของ operation สำหรับส่ง information และ command ระหว่างชั้น มี 4 operation หลักได้แก่

- Request ซึ่ง layer บนต้องการใช้ service บางอย่างจาก layer ล่างก็จะส่งคำสั่ง request ไปยัง layer ล่าง
- Confirm เป็นการตอบสนองกลับของ layer ล่างว่ามัน accept หรือ reject ต่อ request นั้น
- Response เป็นการตอบสนองผลลัพธ์ของ request จาก layer ล่างในแบบใด ซึ่ง response นี้อาจจะ response ทันทีหรือไม่ทันที ซึ่งสามารถหน่วงเวลาออกไปก็ได้ ทั้งนี้ขึ้นอยู่กับว่าสิ่งที่ layer บน request มานั้นเป็นอะไร
- Indication จะถูกสร้างขึ้นมาเมื่อ layer ล่างมีอะไรอยากแจ้ง layer บน

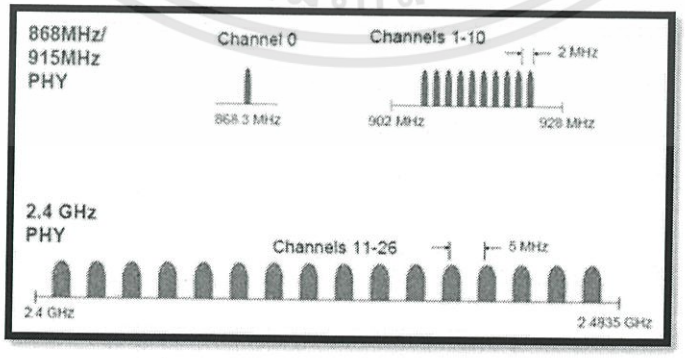
ถัดลงมาจาก APS คือ NWK (Network layer) จัดการเกี่ยวกับ network addressing กับ routing โดยไป invoke ให้เกิดการกระทำใน MAC layer สรุปหน้าที่สำคัญ

- สำหรับตัวแม่ ฟังก์ชันของ layer นี้คือการเริ่มต้นเน็ตเวิร์ก
- กำหนด network address
- เพิ่มหรือลดอุปกรณ์จากเน็ตเวิร์ก
- Routing ข้อมูลไปยังจุดหมายปลายทาง
- ใส่มาตรการรักษาความปลอดภัยให้กับข้อมูลที่จะส่งออก
- Implementing route discovery

- **คุณลักษณะทั่วไปของซิกบี (ZigBee)**

คุณลักษณะทั่วไปของซิกบี นั้นจะมีอัตราการส่งข้อมูล 3 ระดับดังนี้

- ย่านความถี่ 2.4 Ghz มี 16 ช่องสัญญาณ อัตรารับส่งข้อมูล 250 Kbps (ทั่วไป)
- ย่านความถี่ 915 Ghz มี 10 ช่องสัญญาณ อัตรารับส่งข้อมูล 40 Kbps (อเมริกาเหนือ)
- ย่านความถี่ 868 Ghz มี 1 ช่องสัญญาณ อัตรารับส่งข้อมูล 20 Kbps (ยุโรป)



รูปที่ 2.8 ย่านความถี่ใช้งาน ซิกบี (ZigBee)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **ประเภทของอุปกรณ์เชิงคุณภาพ**

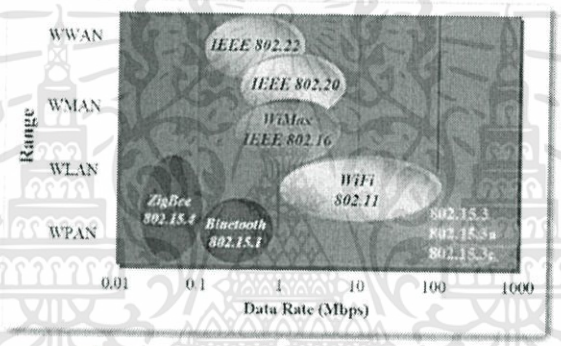
ตามมาตรฐาน IEEE ได้กำหนดประเภทของอุปกรณ์เชิงกายภาพ ไว้ 2 ประเภท คือ Reduced Function Device (RFD) และ Full Function Device (FFD)

➢ RFD (Reduced Functionality Device)

เป็นอุปกรณ์ที่ลดความสามารถลง ฉะนั้นใช้หน่วยความจำน้อยกว่า (ตัวที่ไม่ได้ลด) การประมวลผลลดลง และใช้พลังงานในการดำเนินการต่ำลง เงื่อนไขคือ RFD สามารถคุยกับ FFD ได้เพียงตัวเดียวเท่านั้นในเครือข่ายและไม่สามารถคุยกันเองกับ RFD ได้

➢ FFD (Full Functionality Device)

ต้องการทรัพยากรต่างๆ (หน่วยความจำ พลังงาน ความสามารถประมวลผล) มากกว่า RFD สามารถสื่อสารกับใครเท่าไรก็ได้ มีความสามารถที่จะเป็นได้ทั้งตัวแม่ (coordinator) หรือตัวลูก (end-device) ของเครือข่ายย่านความถี่ใช้งานซิกบี



รูปที่ 2.9 มาตรฐานโปรโตคอลสื่อสารไร้สาย

- **ลักษณะการทำงานของซิกบี**

ซิกบี (ZigBee) แบ่งออกตามลักษณะการทำงาน 3 แบบ คือ

➢ ซิกบี Coordinator

ทำหน้าที่สร้างโครงข่าย จัดการโหนดในโครงข่าย และเก็บข่าวสารของโหนดในโครงข่าย

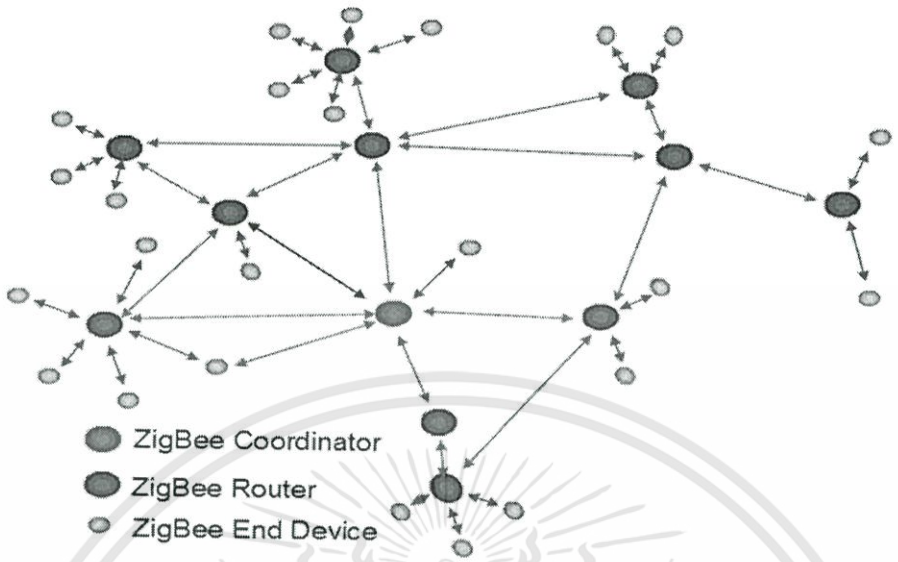
➢ ซิกบี Router

ทำหน้าที่จัดการเส้นทางของข้อมูลที่ส่งผ่านภายในโครงข่ายระหว่างโหนด

➢ ซิกบี end Device

เป็นจุดปลายของโครงสร้างเครือข่าย อยู่ในส่วนของผู้ใช้งานโดยสามารถเป็นได้ทั้ง แบบ RFD และ FFD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 แสดงส่วนประกอบต่างๆ ของซิกบี (ZigBee)

2.3.2 เอ็กซ์บี (XBee)

เอ็กซ์บี เป็นอุปกรณ์ที่มี ไมโครคอนโทลเลอร์ (Microcontroller) และ RF IC อยู่ภายใน ทำหน้าที่เป็นอุปกรณ์รับ-ส่งสัญญาณ (transceiver) แบบ Half Duplex ย่านความถี่ 2.4 GHz มีการจัดการโดยใช้พลังงานต่ำ ใช้งานง่าย มี interface ที่ใช้รับและส่งข้อมูลกับเอ็กซ์บี เป็น UART (TTL) สำหรับด้านไมโครคอนโทรลเลอร์ สามารถนำขาที่ใช้ติดต่อสื่อสาร UART ของเอ็กซ์บี ต่อเข้ากับ UART ของ ไมโครคอนโทรลเลอร์ได้เลย

เอ็กซ์บี สามารถใช้งานตามมาตรฐานซิกบีได้ โดยไม่ต้องเขียนโปรแกรมสร้าง เครือข่ายซิกบี เพราะสามารถโหลด firmware เข้าไปในตัวเอ็กซ์บีสำเร็จรูปเพื่อให้สามารถ set parameter ผ่าน software interface (X-CTU หรือโปรแกรมที่เขียนขึ้นเอง), ผ่านทาง At command (เหมือนกับการควบคุม GSM Module) โดยใช้ Hyper terminal หรือผ่านทางการรับส่งข้อมูลด้วยไมโครคอนโทรลเลอร์ได้อย่างง่าย โดยเมื่อ set เอ็กซ์บีให้ทำงานเป็นอุปกรณ์ในเครือข่ายซิกบี และเรียก เอ็กซ์บีแต่ละตัวว่าเป็นโหนด

Firmware ที่ใช้กับเอ็กซ์บีจะใช้โหลดผ่านโปรแกรม X-CTU ทั้งนี้ เอ็กซ์บีแต่ละรุ่น จะสามารถ Setting Function การใช้งานได้มากมาย ทำให้ Firmware ที่จะต้องโหลดเข้าไปนั้น มีมากมายหลายแบบต้องเลือกให้เหมาะสมกับการใช้งาน สำหรับการใช้งานเอ็กซ์บี มีรายละเอียดต่อไปนี้

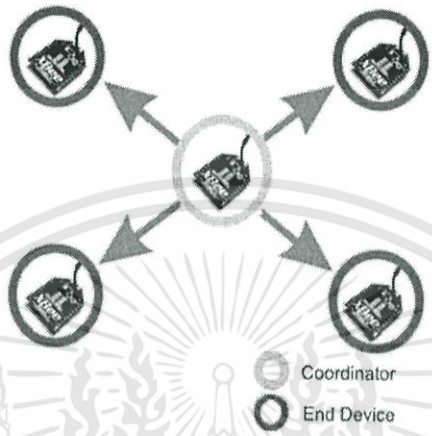
- **เอ็กซ์บีโทโพโลยี (XBee Topology)**

ในการสร้างโครงข่ายไร้สายของซิกบีนั้น จะต้องประกอบด้วย โหนดจำนวนอย่างน้อยที่สุด 2 ชนิด คือ Coordinator node และ node ลูกข่าย ชนิดใดชนิดหนึ่ง (Router/End device) จึงจะสามารถสื่อสารและทำงานในรูปแบบของ PAN (Personal area-network) ได้ โดยซิกบีสามารถแบ่งรูปแบบ เครือข่ายได้เป็น 3 รูปแบบ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

➢ Star (Broadcast)

การเชื่อมต่อ แบบ Star หรือ แบบ Broadcast เป็นการรับส่งข้อมูลแบบ ไม่เฉพาะเจาะจง จุดหมายปลายทาง หรือ เอ็กซ์บี ทุกตัวที่อยู่ในระบบเครือข่ายเดียวกันสามารถ รับข้อมูลทุกข้อมูลได้ทุกตัว



รูปที่ 2.11 Star (Broadcast) Network

จากรูป เป็นการใช้งานแบบ Star จะประกอบไปด้วย เอ็กซ์บีที่ทำงานเป็น 2 รูปแบบคือ แบบที่ 1 เป็น Coordinator ทำหน้าที่ สร้างเครือข่าย และ แบบที่ 2 เป็น End Device ทำหน้าที่เป็นลูกข่าย

- ข้อดี

การติดตั้งเครือข่ายและการดูแลรักษาทำได้ง่าย หากมีเครื่องใดเกิดความเสียหาย ก็สามารถตรวจสอบได้ง่าย และศูนย์กลางสามารถตัดเครื่องที่เสียหายนั้นออกจากการสื่อสารในเครือข่ายได้เลย โดยไม่มีผลกระทบกับระบบเครือข่าย

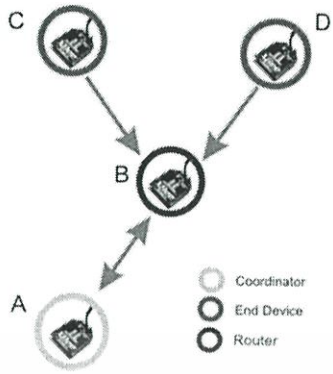
- ข้อเสีย

เสียค่าใช้จ่ายมาก ทั้งในด้านของเครื่องที่จะใช้เป็น เครื่องศูนย์กลาง หรือตัวฮับเอง และค่าใช้จ่ายในการติดตั้งสายเคเบิลในเครื่องอื่น ๆ ทุกเครื่อง การขยายระบบให้ใหญ่ขึ้นทำได้ยาก เพราะการขยายแต่ละครั้ง จะต้องเกี่ยวเนื่องกับเครื่องอื่นๆ ทั้งระบบ

➢ Cluster Tree

เป็นการ รับส่งข้อมูลแบบส่งผ่าน เช่น A ต้องการติดต่อ กับ C แต่ C อยู่ไกลจาก A จน A ไม่สามารถ ติดต่อกับ C ได้ แต่พอดีมี B ที่อยู่ระหว่าง A กับ C ดังนั้น Cluster Tree จะใช้ B เป็นเหมือน ตัวกลาง เชื่อมการติดต่อ (Repeater) ระหว่าง A กับ C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 Cluster Tree (Tree) Network

จากรูปภาพจะพบว่ามีการใช้งานเอ็กซ์บี (XBee) อยู่ 3 ลักษณะด้วยกัน คือ Coordinator, End Device และ Router

- ข้อดี

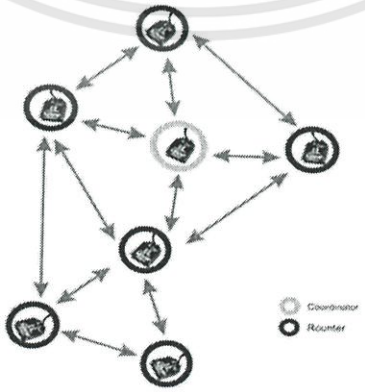
การติดตั้งเครือข่ายและการดูแลรักษาทำได้ง่าย หากมีเครื่องใด เกิดความเสียหายก็สามารถตรวจสอบได้ง่าย และศูนย์กลางสามารถตัดเครื่องที่เสียหายนั้นออกจากการสื่อสารในเครือข่ายได้เลยโดยไม่มีผลกระทบกับระบบเครือข่าย สามารถเปลี่ยนรูปแบบการวางสายได้ง่าย สามารถเพิ่มโหนดได้ง่ายและตรวจสอบจุดที่เป็นปัญหาได้ง่าย

- ข้อเสีย

เสียค่าใช้จ่ายมาก ทั้งในด้านของเครื่องที่จะใช้เป็นเครื่องศูนย์กลาง หรือตัวฮับ และค่าใช้จ่ายในการติดตั้งสายเคเบิลในเครื่องอื่น ๆ ทุกเครื่อง การขยายระบบให้ใหญ่ขึ้นทำได้ยาก เพราะการขยายแต่ละครั้ง จะต้องเกี่ยวเนื่องกับเครื่องอื่นๆ ทั้งระบบ การเชื่อมต่อจากศูนย์กลางมีโอกาสที่ระบบเครือข่าย จะล้มเหลวพร้อมกัน

➢ Mesh

การเชื่อมต่อเครือข่ายแบบ Mesh เป็นโครงข่ายที่มีประสิทธิภาพสูงเนื่องจากข้อมูลสามารส่งไปถึงเป้าหมายได้หลายทาง ทำให้ระบบนี้สามารถรับส่งข้อมูลไปยังจุดหมายปลายทางได้แม้จะเกิดความเสียหายของระบบในบางส่วนก็ตาม ระบบนี้จึงเป็นระบบที่ได้รับความนิยมเป็นอย่างมาก



รูปที่ 2.13 Mesh Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.13 จะพบว่ามีการใช้งานเอ็กซ์บี อยู่ 2 ลักษณะด้วยกันคือ Coordinator และ Router ระบบเครือข่ายแบบ Mesh สามารถค้นหาและเปลี่ยนแปลงเส้นทางการส่งข้อมูลได้เอง ทำให้เสถียรภาพของการรับส่งข้อมูลในรูปแบบนี้จึงมีประสิทธิภาพมาก

- ข้อดี

แต่ละ Link รับผิดชอบเพียงอุปกรณ์ 2 ตัว จึงไม่เกิดปัญหา เกี่ยวกับการจัดสรรช่องการสื่อสาร ถ้ามี Link ใดเสียหาย ระบบยังคงสามารถทำงานได้ ระบบรักษาความปลอดภัยข้อมูลดี และยังสามารถบ่งชี้ตำแหน่งบกพร่องใน Network และแก้ไขได้ง่าย

- ข้อเสีย

จำนวน I/O Links ทั้งหมดมีค่ามากและมีราคาแพง

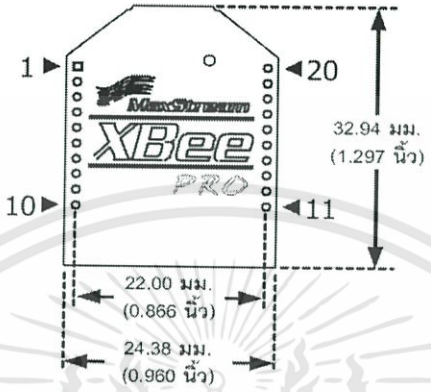
- **คุณสมบัติโดยทั่วไปของเอ็กซ์บี (XBee)**
  - ความถี่ในการทำงาน: 2.4 GHz
  - สายอากาศ: มีสายอากาศแบบ Whip
  - ระยะทำการในร่ม: สูงสุดประมาณ 100 เมตร
  - ระยะทำการในกลางแจ้ง (แบบ line-of-sight) : สูงสุดประมาณ 1,500 เมตร
  - กำลังส่ง: 60mW (18dBm)
  - ความไวในการรับสัญญาณ: -100 dBm (1% packet error rate)
  - การทำงานของพอร์ต: สามารถกำหนดผ่านทางซอฟต์แวร์ X-CTU เพื่อให้การทำงานเป็นอินพุตอะนาล็อกสำหรับวงจรแปลงสัญญาณอะนาล็อกเป็นดิจิตอล ความละเอียด 10 บิต และอินพุตเอาต์พุตดิจิตอล
  - ขนาด: 0.96x1.297 นิ้ว หรือ 2.438x3.294 cm
  - ไฟเลี้ยง: 2.8 ถึง 3.4 V
  - กระแสไฟฟ้า: เมื่อส่งข้อมูล 215mA, รับข้อมูล 55mA, น้อยกว่า 10μA ในโหมดลดพลังงาน ที่ไฟเลี้ยง +3.3V
  - อุณหภูมิใช้งาน: -40 ถึง 85°C

- **คุณสมบัติการสื่อสารข้อมูล**
  - สามารถทำงานเป็นอุปกรณ์มาสเตอร์และสเลฟได้
  - อัตราถ่ายทอดข้อมูลผ่านคลื่นวิทยุ: 250,000 bps
  - อัตราการถ่ายทอดข้อมูลอนุกรม (บอดเรต): 1,200 ถึง115, 200 บิตต่อวินาที
  - รูปแบบโครงข่ายข้อมูลที่รองรับ:จุดต่อจุด (point-to-point), จุดต่อหลายจุด (point-to-multipoint) และเข้ากันได้กับอุปกรณ์ตามมาตรฐานรหัส 802.15.4
  - ทางเลือกแอดเดรส: PAN ID, ช่อง (Channel) และแอดเดรส (Addresses) สำหรับแอดเดรส สามารถกำหนดรหัสแอดเดรสได้มากถึง 65,000 รหัส
  - เทคโนโลยีในการกระจายคลื่น: DSSS (Direct Sequence Spread Spectrum)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รองรับการทำงานทั้งแบบ API และ AT command สามารถกำหนดได้ผ่านทางซอฟต์แวร์ X-CTU

- **การจัดขาของเอ็กซ์บี (XBee)**



รูปที่ 2.14 แสดงการจัดขาของ XBee

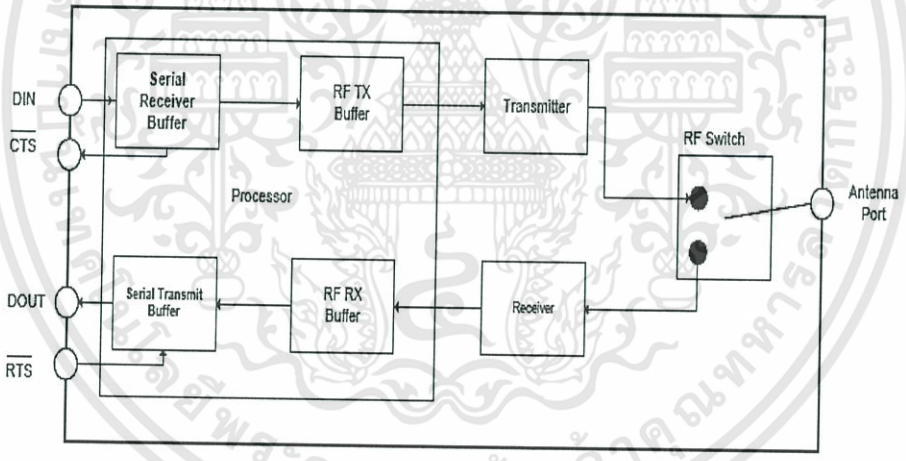
- Vcc: ขาไฟเลี้ยง +3.3V
- DOUT: ขาเอาต์พุตรับข้อมูลอนุกรม
- DIN: ขาอินพุตรับข้อมูลอนุกรม
- DO8: ขาเอาต์พุตดิจิตอลช่อง 8
- RESET: ขารีเซตหลัก (แอกตีฟ “0”)
- PWM0/RSSI : ขาเอาต์พุต PWM ช่อง 0 และขาเอาต์พุตแสดงความแรงของการรับสัญญาณ
- PWM1: ขาเอาต์พุต PWM ช่อง 1
- NC: ไม่ใช้งาน
- DTR/SLEEP_RQ/D18: ขาอินพุตรับสัญญาณให้หยุดทำงานเข้าสู่โหมดสลีปหรือเป็นขาอินพุตดิจิตอลช่อง 8
- GND: ขาต่อกราวด์
- AD4/DIO4: ขาอินพุตอะนาล็อก 4 หรือ ขาอินพุตเอาต์พุตดิจิตอล 4
- CTS/DIO7: อินพุตรับสัญญาณแจ้งการส่งข้อมูลจากโฮสต์ (Clear-To-Sent) ใช้ในการควบคุมจังหวะการรับส่งข้อมูล หรือเป็นขาอินพุตเอาต์พุตดิจิตอล 7
- ON/SLEEP: ขาแสดงสถานะการทำงาน “1” อยู่ในโหมดทำงานปกติ “0” อยู่ในโหมดสลีป
- VRE.: ขาต่อแรงดันอ้างอิงสำหรับโมดูลแปลงสัญญาณอะนาล็อกเป็นดิจิตอลภายใน XBee Series 2
- Associated/AD5/DIO5: ขาแสดงสถานะการเชื่อมต่อหรือขาอินพุตอะนาล็อก 5 หรือขา อินพุตเอาต์พุตดิจิตอล 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- RTS/AD6/DIO6: ขาเอาต์พุตแจ้งความพร้อมในการส่งข้อมูล (Ready-To-Send) ใช้ควบคุมจังหวะการรับส่งข้อมูลหรือเป็นขาอินพุตอะนาล็อก 6 หรือเป็นอินพุตเอาต์พุตดิจิตอล 6
- AD3/DIO3: ขาอินพุตอะนาล็อก 3 หรือขาอินพุตเอาต์พุตดิจิตอล 3
- AD2/DIO2: ขาอินพุตอะนาล็อก 2 หรือขาอินพุตเอาต์พุตดิจิตอล 2
- AD1/DIO1: ขาอินพุตอะนาล็อก 1 หรือขาอินพุตเอาต์พุตดิจิตอล 1
- AD0/DIO0: ขาอินพุตอะนาล็อก 0 หรือขาอินพุตเอาต์พุตดิจิตอล 0

- **การส่งผ่านข้อมูลของเอ็กซ์บี**

โดยทั่วไปการใช้งาน RF Module ควรกำหนดให้มี Buffer ด้วย เพื่อการปรับอัตรารับส่งข้อมูลระหว่างตอนที่รับส่งทางอากาศ กับตอนที่รับส่งไปยัง ไมโครคอนโทรลเลอร์ หรือ อุปกรณ์อื่น ๆ ได้อย่างเหมาะสม Data ที่รับส่งระหว่าง MCU กับ เอ็กซ์บี จะมีข้อจำกัด เรื่อง Packet อาจถูก Drop ได้ เนื่องมากจาก Data Over Flow โดยสำหรับด้านการส่งข้อมูลไปที่ เอ็กซ์บี เพื่อออกอากาศนั้น ที่ขา DI จะมี Buffer อยู่ประมาณ 202 Bytes หากส่งเกิน Buffer จะเกิดการ Drop packet ทิ้ง ซึ่งทางฝั่งรับข้อมูล ที่ขา DO ก็มี Buffer อยู่เช่นกัน โดยจะมี Parameter ที่เกี่ยวข้องกับ Data Throughput คือ RO และ BD



รูปที่ 2.15 แผนภาพการไหลภายในข้อมูล

ค่า RO คือค่า Packetization Timeout ซึ่งเป็น delay ของข้อมูลที่อยู่ใน DI Buffer ก่อนที่จะถูก encapsulate ไปที่ส่วน RF transmission เพื่อส่งข้อมูลออกอากาศ หากตั้ง RO = 0 Data ที่รับเข้ามาจาก MCU จะถูก เอ็กซ์บี Encapsulate Packet ส่งออกอากาศทันที ดังนั้น เมื่อมี Parameter RO และ BD ที่จะช่วยในการปรับ Data รับส่งให้สามารถรับส่งกันได้ทัน ไม่ให้มีการ Drop Packet ได้ ในกรณีที่ส่งข้อมูลเกิน 200 Bytes

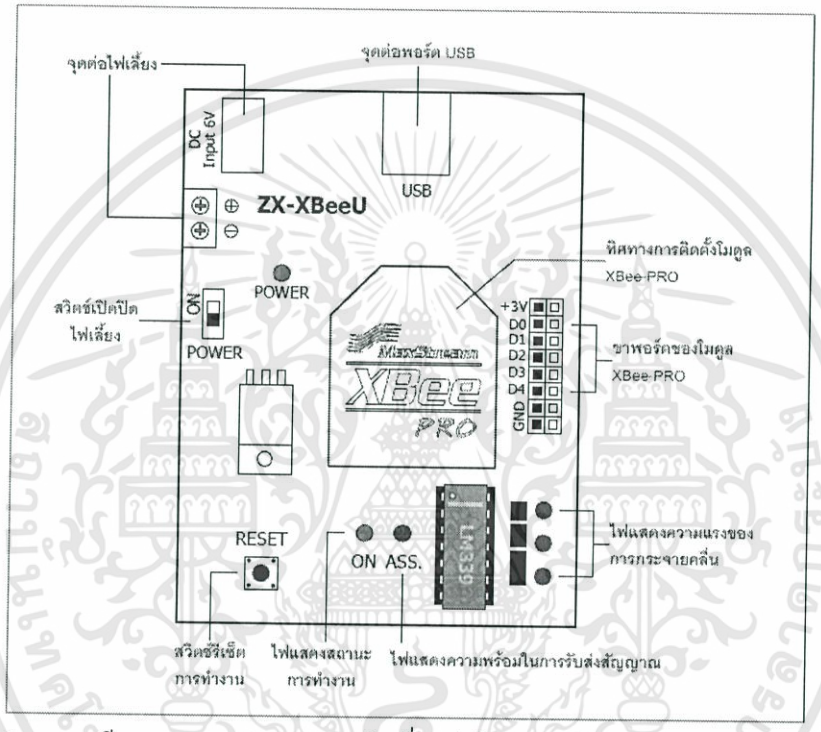
นอกจากนี้ยังมี PIN CTS (ขา12) และ RTS (ขา16) ทำหน้าที่ช่วยเตือนเวลาที่ Buffer ภายในใกล้จะเต็ม โดยในฝั่งส่ง DI Buffer จะส่ง Signal มาทาง CTS เมื่อ DI Buffer เหลือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พื้นที่จัดเก็บอยู่อีก 17 Bytes และส่ง Clear Signal ที่ CTS เมื่อ DI Buffer เหลือพื้นที่จัดเก็บมากกว่า 34 Bytes

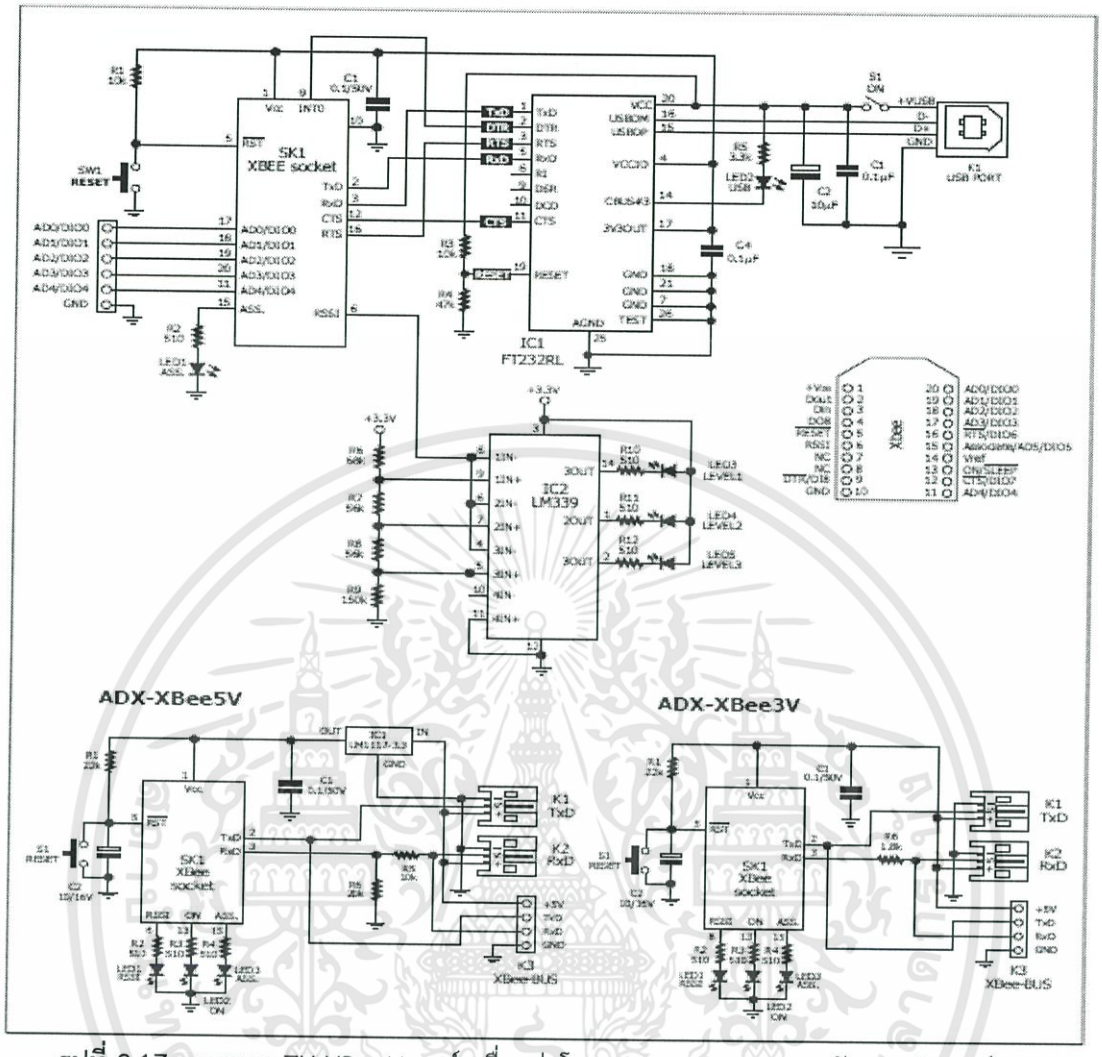
2.3.3 ZX-XBeeU บอร์ดเชื่อมต่อโมดูล XBee-PRO กับคอมพิวเตอร์

เป็นบอร์ดสำหรับติดตั้งโมดูล XBee-PRO เพื่อเชื่อมต่อกับคอมพิวเตอร์สำหรับตั้งค่าพารามิเตอร์ต่างๆ ในรูปที่ 2.16 แสดงรายละเอียดของบอร์ด ZX-XBeeU และการติดตั้งโมดูล XBee-PRO ลงบนบอร์ด สำหรับวงจรของบอร์ด ZX-XBeeU แสดงในรูปที่ 2.17



รูปที่ 2.16 รายละเอียดของ ZX-XBeeU บอร์ดเชื่อมต่อโมดูล XBee/XBee-PRO กับคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



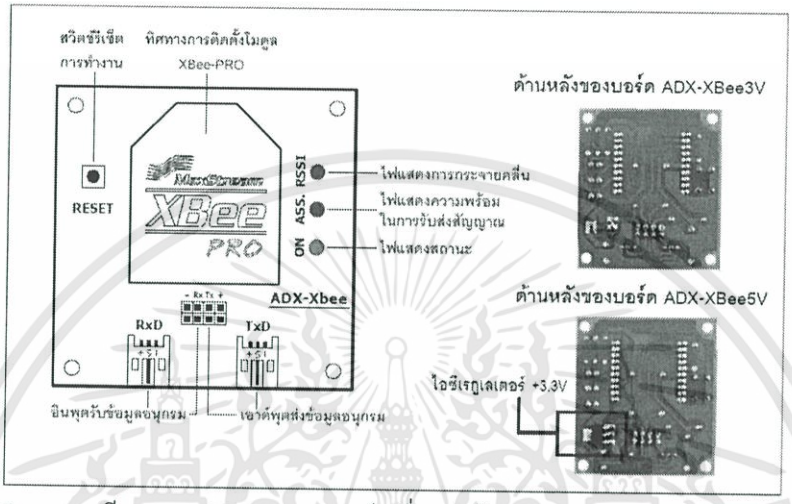
รูปที่ 2.17 วงจรของ ZX-XBeeU บอร์ดเชื่อมต่อโมดูล XBee/XBee-PRO กับคอมพิวเตอร์

- **คุณสมบัติทางเทคนิคที่สำคัญของบอร์ ด ZX-XBeeU มี ดังนี้**
  - มีคอนเน็กเตอร์ตัวเมียสำหรับติดตั้งโมดูล XBee และ XBee-PRO
  - มีจุดต่อขาพอร์ต D0 ถึง D4 ของโมดูล XBee-PRO ซึ่งสามารถกำหนดลักษณะการทำงานได้ผ่านทางซอฟต์ แวร์ X-CTU
  - เชื่อมต่อคอมพิวเตอร์ผ่านทางพอร์ต USB
  - ใช้งานกับซอฟต์แวร์ X-CTU เพื่อกำหนดค่าพารามิเตอร์และใช้ในการอัปเกรดเฟิร์มแวร์
  - ใช้ไฟเลี้ยง +5V ผ่านพอร์ต USB ของคอมพิวเตอร์
  - มี LED แสดงความพร้อมในการติดต่อสื่อสารข้อมูล ASS. และ LED แสดงความแรงของการส่งกระจายคลื่น 3 ระดับ (จะแสดงผลเมื่อมีการรับส่งข้อมูลเท่านั้น)
  - ขนาด 2.25 x 3.5 นิ้ว

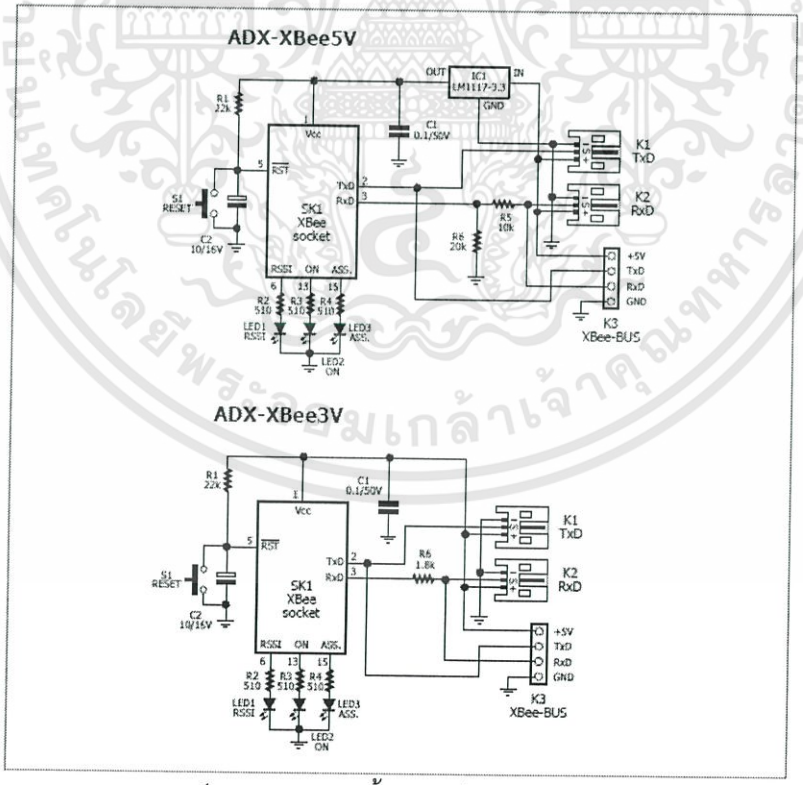
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4 ADX-XBee บอร์ดเชื่อมต่อโมดูล XBee-PRO กับอุปกรณ์ภายนอก

เป็นบอร์ดสำหรับติดตั้งโมดูล XBee-PRO เพื่อเชื่อมต่อกับอุปกรณ์ภายนอก อาทิ ไมโครคอนโทรลเลอร์ เพื่อช่วยให้ไมโครคอนโทรลเลอร์ มีระบบสื่อสารข้อมูลอนุกรมไร้สาย 2.4GHz ในรูปที่ 2.18 แสดงรายละเอียดของบอร์ด ADX-XBee และการติดตั้งโมดูล XBee ลงบนบอร์ด ส่วนวงจรของบอร์ด ADX-XBee ทั้ง 2 รุ่น คือ รุ่น 5V และ 3V แสดงในรูปที่ 2.18



รูปที่ 2.18 รายละเอียดของ ADX-XBee บอร์ดเชื่อมต่อโมดูล XBee-PRO กับอุปกรณ์ภายนอก



รูปที่ 2.19 แสดงวงจรของบอร์ด ADX-XBee ทั้ง 2 รุ่นคือ ADX-XBee5V สำหรับระบบ +5V และ ADX-XBee3V สำหรับระบบ +3.3 หรือ +3V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **คุณสมบัติทางเทคนิคที่สำคัญของบอร์ด ADX-XBee มี ดังนี้**
  - มีคอนเน็กเตอร์ตัวเมียสำหรับติดตั้งโมดูล XBee และ XBee-PRO
  - มีจุดต่อขาพอร์ต TxD และ RxD สำหรับเชื่อมต่ออุปกรณ์ภายนอกโดยจัดสัญญาณผ่านคอนเน็กเตอร์ JST 3 ขา และ IDC ทั้งแบบตัวผู้และตัวเมียทำให้อุปกรณ์ภายนอกไม่ว่าจะเป็นบอร์ดไมโครคอนโทรลเลอร์ สามารถติดต่อใช้งานกับโมดูล XBee-PRO ได้ง่ายและสะดวกขึ้น
  - ใช้ไฟเลี้ยง +5V สำหรับรุ่น ADX-XBee5V และ +3.3V สำหรับ ADX-XBee3V
  - มี LED แสดงความพร้อมในการสื่อสารข้อมูล (ASS.) และแจ้งสภาวะการรับส่งสัญญาณ (RSSI)
  - ขนาดเล็กเพียง 1.8 x 2 นิ้ว

2.4 ไมโครคอนโทรลเลอร์ PIC18F458

ไมโครคอนโทรลเลอร์ PIC18F458 เป็นไมโครคอนโทรลเลอร์อีกเบอร์หนึ่งของตระกูล PIC (Peripherral Interface Controller) ในปัจจุบันซึ่งมีศักยภาพในการทำงานสูงและในคุณสมบัติของไมโครคอนโทรลเลอร์นี้คือเพียบพร้อมไปด้วยทรัพยากรหรือฟังก์ชั่น (Function) การใช้งานต่างๆไว้ภายในตัวมันเอง เช่น มีโมดูล (Module) แปลงสัญญาณแอนาลอกเป็นดิจิตอล (Analog To Digital Converter) , USART , SPI , I2C , PWM อื่นๆ มีโมดูลที่เพิ่มขึ้นมาใหม่คือ Can Module , ECCP ซึ่งเป็นคุณสมบัติใหม่ของไมโครคอนโทรลเลอร์ตระกูล PIC โดยมีในไมโครคอนโทรลเลอร์ PIC18F458 และเบอร์อื่นๆ ของ PIC18FXXX และยังเหมาะต่อการใช้งานตรงที่หน่วยความจำโปรแกรมเป็นหน่วยความจำแบบแฟลช (Flash Program Memory) ซึ่งสามารถเขียนและลบข้อมูลได้ด้วยสัญญาณทางไฟฟ้าได้นับหลายพันครั้ง ข้อเด่นอีกประการหนึ่งของไมโครคอนโทรลเลอร์เบอร์นี้ในเรื่องความเร็ว PIC18F458 สามารถทำงานได้ที่ความถี่นาฬิกาสูงสุดถึง 40 แมกกะเฮิรตซ์ (MHz) และมีวงจร PLL (Phase Lock Loop) ซึ่งเป็นวงจรคูณความถี่อยู่ภายในโดยสามารถเลือกโดยการโปรแกรมทางซอฟต์แวร์ ซึ่งสามารถคูณค่าความถี่ที่รับเข้ามาได้ถึง 4 เท่าของสัญญาณนาฬิกาจากภายนอกทั้งยังทำงานในลักษณะไปป์ไลน์ (Pipe Line) ทำให้มีความเร็วในการทำงานมากกว่าซีพียูทั่วไปที่ค่าความถี่เดียวกันโดยลักษณะการทำงานจะใช้สัญญาณนาฬิกาเพียง 1 หรือ 2 ไซเคิล (Cycle) ต่อคำสั่งเท่านั้น และหน่วยความจำไม่ถูกแบ่งเป็นเพจ (Page) อีกต่อไป การเขียนโปรแกรมจึงง่ายโดยไม่ต้องเลือกแบงก์ (Bank)

2.4.1 คุณสมบัติของไมโครคอนโทรลเลอร์ PIC18F458

สามารถสรุปคุณสมบัติของไมโครคอนโทรลเลอร์ PIC18F458 ได้ดังนี้

1. มีชุดคำสั่ง 75 คำสั่ง
2. .ซีพียูเป็นแบบ RISC (Reduce Instruction Set Computer)
3. เป็นซีพียู 16 บิต
4. หน่วยความจำ SRAM 1536 ไบต์ (Byte)
5. หน่วยความจำโปรแกรม 32 กิโลไบต์ (Kbyte)
6. หน่วยความจำอีอีพรอม 256 ไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. รับความถี่สัญญาณนาฬิกาตั้งแต่ไฟตรงถึง 40 เมกะเฮิรตซ์

8. มีวงจร PPL (Phase Look Loop) ภายในคูณค่าความถี่ของสัญญาณนาฬิกา 4 เท่าของค่าสัญญาณนาฬิกาอินพุต

9. ตอบสนองการอินเตอร์รัพต์ (Interrupt) ได้ถึง 21 แหล่ง

10. มีขารับสัญญาณอินเตอร์รัพต์จากภายนอก 3 ขา คือ RB0/INT0, RB1/INT1, RB2/INT2

11. เลือกลำดับความสำคัญของการอินเตอร์รัพต์จากอุปกรณ์ต่อพ่วงได้

12. กระแสซิงก์ (Synchronous) และ (Source) ของพอร์ต (port) สูงสุด 25 มิลลิแอมป์

13. มีโมดูลไทเมอร์ 4 ตัว ดังนี้

13.1 ไทเมอร์ 0 ขนาด 8/16 บิต เป็นไทเมอร์/เคาน์เตอร์ พร้อมปรีสเกลเลอร์ 8 บิต

13.2 ไทเมอร์ 1 ขนาด 16 บิต เป็นไทเมอร์/เคาน์เตอร์ พร้อมปรีสเกลเลอร์ 8 บิต

13.3 ไทเมอร์ 2 ขนาด 8 บิต มีปรีสเกลเลอร์ โพสต์สเกลเลอร์ และรีจิสเตอร์คาบเวลา (Period Register) เป็นตัวเปรียบค่ากับไทเมอร์ 2 อยู่ภายใน

13.4 ไทเมอร์ 3 ขนาด 16 บิต เป็นไทเมอร์/เคาน์เตอร์

14. มีโมดูล CCP (Capture/Compare/PWM) 1 ชุด ส่วนตรวจจับสัญญาณ (Capture) ขนาด 16 บิต ความละเอียดสูงสุด 6.25 นาโนวินาที ส่วนเปรียบเทียบสัญญาณ (Compare) ความละเอียดสูงสุด 100 นาโนวินาที และส่วนวงจรมอดูเลชันทางความกว้างของพัลส์ (Pulse Width Modulation : PWM) ความละเอียดสูงสุด 10 บิต

15. มีโมดูล ECCP (Enhanced Capture/Compare/PWM) 1 ชุด ทำงานคล้ายกับโมดูล CCP แต่จะต่างกันตรงที่จะใช้งานในการควบคุมมอเตอร์

16. มีโมดูล MSSP (Master Syncronouse Serial Port) ใช้งานเป็นวงจรเชื่อมต่ออุปกรณ์ทำงานได้ 2 โหมด คือ SPI และ I2C

17. มีโมดูลการสื่อสารอนุกรม USART (Universal Syncronous Asyncronous Receiver Transmitter)

18. มีโมดูลแปลงสัญญาณแอนะลอกเป็นดิจิตอล (Analog to Digital Converter) อยู่ภายในความละเอียด 10 บิต 8 แชนแนล

19. มีโมดูลเปรียบเทียบแรงดันแอนะลอก (Comparater Voltage Referrent Module) ภายใน

20. สามารถเลือกโหมดการป้องกันข้อมูลได้ (Code Protection)

21. ทำงานที่ไฟเลี้ยง 2.0V ถึง 5.5V

22. สามารถโปรแกรมด้วยแรงดันไฟฟ้าต่ำ LVP (Low Voltage Programing)

23. ฟังก์ชั่นการโปรแกรมเป็นแบบ ICSP (In Circuit Serial Programnig)

24. มีเพาเวอร์ออนรีเซต (Power On Reset : POR) เพาเวอร์อัปไทเมอร์ (Power Up Timer : PWRT) และมีออสซิลเลเตอร์สตาร์ตอัปไทเมอร์ (Oscillator Start Up Timer : OST)

25. มีวอตช์ด็อกไทเมอร์ (Watch Dog Timer : WDT)

26. มีโหมดในการประหยัดพลังงาน (Sleep Mode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

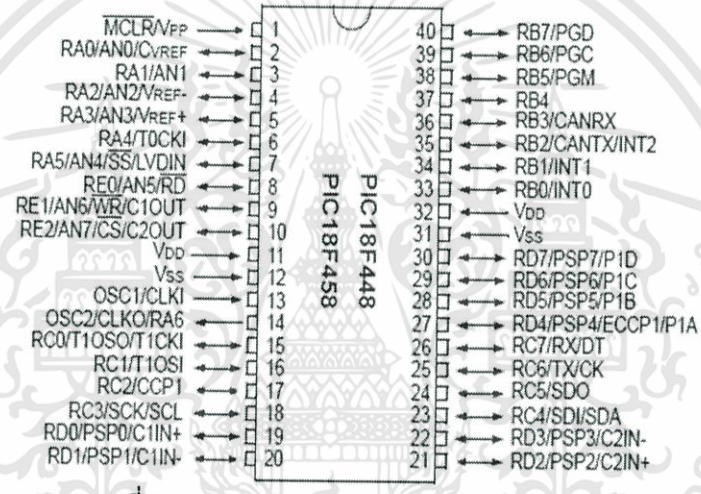
27. มีฟังก์ชั่นการตรวจสอบไฟเลี้ยง (Brown Out Reset : BOR)

28. มีสแต็ก (Stack) 31 ระดับ

29. มีพอร์ตอินพุต 5 พอร์ต คือ A, B, C, D และ E รวมแล้วพอร์ตการใช้งานทั้งหมด คือ 34 บิต โดยพอร์ต A มีจำนวน 7 บิตคือ RA0-RA6, พอร์ต B มีจำนวน 8 บิตคือ RB0-RB7, พอร์ต C และพอร์ต D มีจำนวน 8 บิตคือ RC0-RC7 และ RD0-RD7, พอร์ต E มีจำนวน 3 บิตคือ RE0-RE2

2.4.2 การจัดขาของ PIC18F458

PIC18F458 มีขาทั้งหมด 40 ขา ในตัวถังแบบ PDIP และ 44 ขา ตัวถึงแบบ TQFP ซึ่งแต่ละขาจะมีหน้าที่การใช้งานแตกต่างกันออกไป และมีขาอินพุตทั้งหมด 34 ขา ซึ่งทั้ง 34 ขาสามารถเลือกใช้งานเป็นอินพุตได้ทั้งหมดดังรูป



รูปที่ 2.20 รูปแสดงการจัดวางขาของ PIC18F458

ตารางที่ 2.1 ตารางแสดงคุณสมบัติและรายละเอียดขาต่างๆของ PIC18F458

ขาสัญญาณ	ขาที่	ชนิดของขา	ชนิดของบัฟเฟอร์	รายละเอียด
OSC1/CLK1	9	I	ST	ขารับสัญญาณนาฬิกาหลักจากภายนอก
OCS2/CLK OUT	14	O	ST	เป็นขาของสัญญาณนาฬิกา (1/4 ของ OSC1) ใช้ต่อร่วมกับขา OCS1 ในโหมด XT ถ้าใช้โหมดสัญญาณนาฬิกาแบบ RC ขานี้ปล่อยลอยไว้
MCLR/VPP	1	I/P	ST	ขารีเซตหลัก / ขาสัญญาณในการโปรแกรม
RA0/AN0/CVREF	2	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาอินพุตวงจรเปรียบเทียบแรงดันแอนะลอกช่อง AN0 / เอาต์พุตแรงดันด้านอ้างอิงของโมดูลแรงดันอ้างอิง
RA1/AN1	3	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาอินพุตวงจรเปรียบเทียบแรงดันแอนะลอกช่อง AN1
RA2/AN2/VREF-	4	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาอินพุตวงจรเปรียบเทียบแรงดันแอนะลอกช่อง AN2/ ขาสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RA3/AN3/VREF+	5	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต/ขาอินพุตวงจรเปรียบเทียบแรงดันแอนะลอกช่อง AN3/ ขาสัญญาณ
RA4/TOCK1	6	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต / สัญญาณนาฬิกาของไทเมอร์ 0
RA5/AN4/SS/ LVDIN	7	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาอินพุตวงจรเปรียบเทียบแรงดันแอนะลอกช่อง AN4 / ขาสัญญาณ Slave Select ในการสื่อสารแบบซิงโครนัส
RB0/INT0	34	I/O	TTL/ST	ขาพอร์ตอินพุตเอาต์พุต / ขาสัญญาณอินเตอร์รัพต์จากภายนอก INT0
RB1/INT1	35	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาสัญญาณอินเตอร์รัพต์จากภายนอก INT1
RB2/CANTX/ INT2	36	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้ในการส่งข้อมูลใน Can Module/ขารับสัญญาณอินเตอร์รัพต์จากภายนอก INT2
PB3/CANRX	37	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้ในการส่งข้อมูลใน Can Module
PB5/PGM	38	I/O	TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาสัญญาณการโปรแกรมด้วยไฟฟ้าแรงต่ำ
PB6/PGC	39	I/O	TTL/ST	ขาพอร์ตอินพุตเอาต์พุต / ขาสัญญาณอินเตอร์รัพต์จากการเปลี่ยนแปลงสถานะของขาสัญญาณ / ขาสัญญาณนาฬิกาในการโปรแกรม
PB7/PGD	40	I/O	TTL/ST	ขาพอร์ตอินพุตเอาต์พุต / ขาสัญญาณอินเตอร์รัพต์จากการเปลี่ยนแปลงสถานะของขาสัญญาณ/ ขาสัญญาณนาฬิกาในการโปรแกรม
RC0/T1OSO/TICK1	15	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต / ขาเอาต์พุตออสซิลเลเตอร์ของไทเมอร์ 1 / ขาอินพุตสัญญาณนาฬิกาของไทเมอร์ 1
RC1/TIOS1	16	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต / ขาเอาต์พุตออสซิลเลเตอร์ของไทเมอร์ 1
RC2/CCP1	17	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต / ขาสัญญาณ (Capture 1 Input / Compare 1 Input / PWM 1 Input)
RC3/SCK/SCL	18	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต / ขาอินพุตสัญญาณนาฬิกาในการในโหมด I2C และ SPI
RC4/SDI/SDA	23	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต/ขาสัญญาณส่งข้อมูลในโหมด SPI / เอาต์พุตสัญญาณข้อมูลในโหมด I2C
RC5/SDO	24	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต / ขุสัญญาณส่งข้อมูลในโหมด SPI
RC6/TX/CK	25	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต / ขาสัญญาณส่งข้อมูลในการสื่อสาร USART / ขาสัญญาณนาฬิกาในโหมดการสื่อสารแบบซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RC7/RX/DT	26	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต / ขาสัญญาณรับข้อมูลในการสื่อสาร USART / ขาสัญญาณข้อมูลในโหมดการสื่อสารแบบซิงโครนัส
RD0/PSP0/C11N+	19	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนานติดต่อกับระบบบัสไมโครโปรเซสเซอร์
RD1/PSP1/C11N	20	I/O	ST	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนานติดต่อกับระบบบัสไมโครโปรเซสเซอร์
RD2/PSP2/C2IN+	21	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนานติดต่อกับระบบบัสไมโครโปรเซสเซอร์
RD0/PSP0/CIIN+	19	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนาน (Parallel Slave Port : PSP) / ขาอินพุต CIIN+ ของโมดูลเปรียบเทียบแรงดันแอนะลอก
RD1/PSP1/CIIN-	20	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนาน / ขาอินพุต CIIN- ของโมดูลเปรียบเทียบแรงดันแอนะลอก
RD2/PSP2/C2IN+	21	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนาน / ขาอินพุต C2IN+ ของโมดูลเปรียบเทียบแรงดันแอนะลอก
RD3/PSP3/C2IN-	22	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนาน / ขาอินพุต C2IN- ของโมดูลเปรียบเทียบแรงดันแอนะลอก
RD4/PSP4/ECCP/PIA	27	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนาน ขาสัญญาณของโมดูล ECCP ช่องที่ PIA
RD5/PSP5/PIB	28	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนาน ขาสัญญาณของโมดูล ECCP ช่องที่ PIB
RD6/PSP6/PIC	29	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนาน ขาสัญญาณของโมดูล ECCP ช่องที่ PIC
RD7/PSP7/PID	30	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาใช้งานเป็นพอร์ตขนาน ขาสัญญาณของโมดูล ECCP ช่องที่ PIC
RE0/AN5/RD	8	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาอินพุตแอนะลอกช่องที่ AN5 / ขาควบคุมการไหลในโหมด PSP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

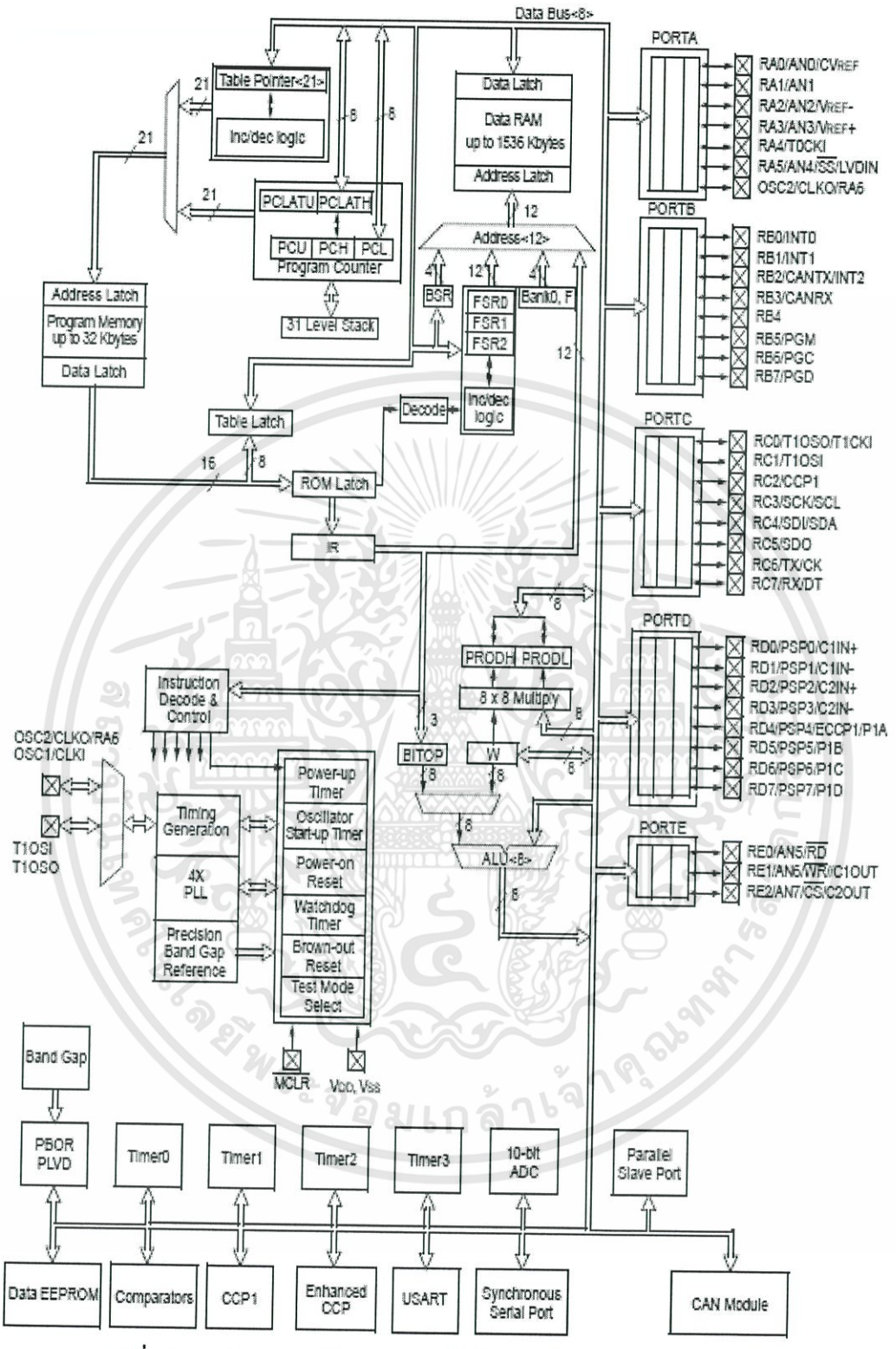
RE1/AN6/WR/ C1OUT	9	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาอินพุตแอนะลอกช่องที่ AN6 / ขาควบคุมการไหลในโหมด PSP
RE2/AN7/CS/ C2OUT	10	I/O	ST/TTL	ขาพอร์ตอินพุตเอาต์พุต / ขาอินพุตแอนะลอกช่องที่ AN7 / ขาควบคุมการไหลในโหมด PSP
Vcc , Vss	11,12	-	-	ขาต่อไฟเลี้ยงบวก 2-6V , ขา 12 ต่อกับกราวด์
หมายเหตุ : 1 = อินพุต 0 = เอาต์พุต		I/P = อินพุต / เอาต์พุต TTL = วงจรทีทีแอล		ST = วงจรชมิตต์ทริกเกอร์ P = เพาเวอร์

2.4.3 สถาปัตยกรรมและโครงสร้างของไมโครคอนโทรลเลอร์ PIC18F458

ไมโครคอนโทรลเลอร์ PIC18F458 เป็นไมโครคอนโทรลเลอร์ที่พัฒนาสถาปัตยกรรมมาจากเบอร์อื่นๆ ของตระกูล PIC เช่น PIC16F877 ซึ่งถือว่ามีคุณสมบัติที่ใกล้เคียงกันกับ PIC18F458 แต่จะต่างกันตรงที่มีฟังก์ชั่นและโมดูลต่างๆที่เพิ่มเติมเข้ามาและที่ได้เห็นชัดคือ PIC18F458 นั้นเป็นไมโครคอนโทรลเลอร์ 16 บิต ที่มีชุดคำสั่งทั้งหมด 75 คำสั่งซึ่งยังไม่รวมคำสั่งไดเร็กทีฟที่มากับโปรแกรม MPLAB ไมโครคอนโทรลเลอร์ PIC18F458 เป็นไมโครคอนโทรลเลอร์ที่มีการแยกหน่วยความจำข้อมูลออกจากกันทำให้ทำงานได้เร็วขึ้นและนอกจากการจัดสถาปัตยกรรมแบบนี้แล้ว ไมโครคอนโทรลเลอร์ตระกูล PIC ยังมีลักษณะเด่นอีกอย่างคือมีการทำงานเป็นแบบไปป์ไลน์ (Pipe Line) ซึ่งลักษณะการทำงานคือสามารถเฟตช์ (fetch) คำสั่งถัดไปได้ในขณะที่กำลังเอ็กซิคิวต์ (Execute) อยู่จึงทำให้การทำงานของไมโครคอนโทรลเลอร์ PIC18F458 มีการทำงานได้เร็วขึ้น โครงสร้างจะมีบางส่วนที่คล้ายคลึงกับ PIC16F877 ซึ่งสิ่งที่เพิ่มเข้ามาก็คือระบบ Can Module,ECCP และยังมีไทเมอร์เพิ่มขึ้นมาอีก 1 ตัว มีขนาด 16 บิต คือไทเมอร์ 3 รวมแล้วเป็นจำนวน 4 ตัว และหน่วยความจำก็ได้เพิ่มขนาดขึ้นด้วยโดย PIC16F877 มีหน่วยความจำโปรแกรมเพียง 8 กิโลไบต์ หน่วยความจำข้อมูล 368 กิโลไบต์ ส่วน PIC18F458 มีมากถึง 32 กิโลไบต์ และมีหน่วยความจำข้อมูล 1536 กิโลไบต์ สำหรับหน่วยความจำอีอีพรอมคือ 256 กิโลไบต์ สถาปัตยกรรมและการทำงานของไมโครคอนโทรลเลอร์ตระกูล PIC โดยรวมแล้วจะมีส่วนการทำงานพื้นฐานที่เหมือนกัน

การทำงานของไมโครคอนโทรลเลอร์เริ่มจากการป้อนไฟเลี้ยง และป้อนสัญญาณนาฬิกาให้แก่ตัวมันจากนั้นซีพียูจะติดต่อกับหน่วยความจำโปรแกรม เพื่อที่จะอ่านคำสั่งแล้วทำงานตามคำสั่งที่บรรจุอยู่ในหน่วยความจำโปรแกรม เมื่อไมโครคอนโทรลเลอร์ทำงานตามคำสั่งที่กำหนดให้ข้อมูลของชุดคำสั่งถูกนำไปเก็บไว้ในรีจิสเตอร์คำสั่ง (Instruction Register) จากนั้นจะถูกส่งไปยังวงจรถอดรหัสเพื่อทำการควบคุมไทเมอร์ทั้งหมดภายในตัวไมโครคอนโทรลเลอร์นอกจากนี้ยังส่งไปควบคุมหน่วยคำนวณทางคณิตศาสตร์ (Arithmetic Logic Unti : ALU) โดยผ่านทางด้านวงจรมัลติเพล็กซ์ (Multiplex) ด้วยการกำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์ PIC18F458 เป็นหน้าที่ของส่วนกำเนิดจังหวะการทำงาน (Timming Generation) ซึ่งทำงานสัมพันธ์กับไทเมอร์ 3 ตัวคือ ออสซิลเลเตอร์สตาร์อัปไทเมอร์ วอตช์ด็อกไทเมอร์ และเพาเวอร์ไทเมอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 รูปแสดงสถาปัตยกรรมของไมโครคอนโทรลเลอร์ PIC18F458

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.4 โหมดสัญญาณนาฬิกา

ในไมโครคอนโทรลเลอร์ PIC18F458 สามารถเลือกตำแหน่งสัญญาณนาฬิกาที่ป้อนให้แก่ซีพียูได้ 8 โหมดการกำหนดให้เป็นโหมดต่างๆ นั้นสามารถกระทำได้ที่รีจิสเตอร์กำหนดค่าการทำงาน (Configuration Bit) โดยกำหนดที่บิต FOSC2, FOSC1 และ FOSC0

LP: ใช้สัญญาณนาฬิกาจากคริสตอลกำลังงานต่ำ

XT: ใช้สัญญาณนาฬิกาจากคริสตอลแบบเรโซเนเตอร์

HS: ใช้สัญญาณนาฬิกาจากคริสตอลความเร็วสูง

HS4: ใช้วงจร PPL ภายในคูณค่าความถี่ของสัญญาณนาฬิกาจากความถี่สัญญาณนาฬิกาอินพุต 4 เท่าโดยใช้ร่วมกับคริสตอลความเร็วสูง

RC: ใช้สัญญาณนาฬิกาตัวต้านทานและตัวเก็บประจุภายนอก

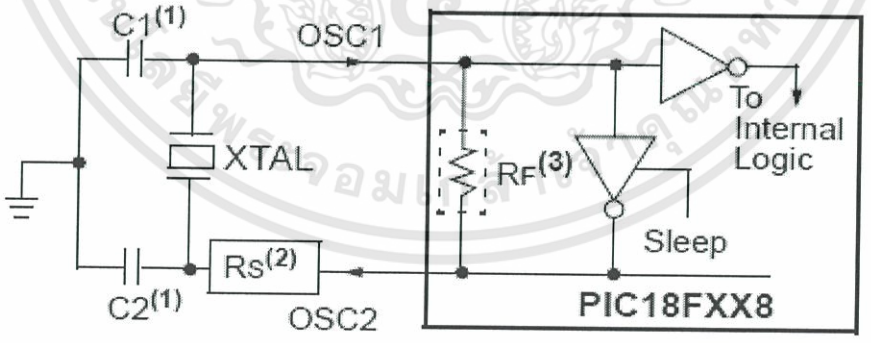
EC: ใช้สัญญาณนาฬิกาจากภายนอก

RCIO:ใช้สัญญาณนาฬิกาตัวต้านทานและตัวเก็บประจุต่อภายนอกแต่โหมดนี้จะสามารถใช้งานขา OSC2 เป็นขาอินพุตเอาต์พุตได้

ECIO: ใช้สัญญาณนาฬิกาจากภายนอกป้อนที่ขา OSC1 แต่โหมดนี้จะสามารถใช้งานขา OSC2 เป็นขาอินพุตเอาต์พุตได้

2.4.4.1. การป้อนสัญญาณนาฬิกาโดยใช้คริสตอล

โหมดนี้สัญญาณนาฬิกา LP, XT, HS, และ HS4 และมีสัญลักษณ์การต่อเหมือนกันแต่จะต่างกันตรงที่ชนิดของคริสตอลที่ใช้ ส่วนการต่อใช้งานแบบนี้จะไม่สามารถนำขา OSC2 ไปใช้งานเป็นขาอินพุตเอาต์พุตได้ถ้าหากต้องการนำขา OSC2 ไปใช้งานจะต้องใช้คริสตอลที่มีโมดูล คือ มีตัวเก็บประจุและตัวต้านทานอยู่ภายใน ซึ่งการใช้งานก็เพียงต่อไฟเลี้ยงให้เท่านั้น แล้วยังสามารถนำขา OSC2 ไปใช้งานได้โดยที่ขานี้จะมีความถี่ออกมาเป็นความถี่ของ OSC1/4 ประโยชน์ก็คือสามารถนำขา OSC2 ไปใช้งานเป็นสัญญาณนาฬิกาให้กับอุปกรณ์อื่นๆ ได้



รูปที่ 2.22 รูปแสดงการต่อคริสตอลแบบเรโซเนเตอร์

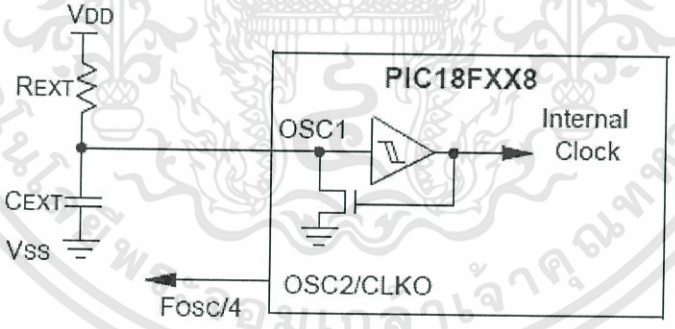
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 ตารางแสดงค่าของตัวเก็บประจุที่ใช้กับคริสตอลออสซิลเลเตอร์

ชนิดคริสตอลที่ใช้	ความถี่	C1	C2
LP	32.0 KHz	33 pF	33 pF
	200 KHz	15 pF	15 pF
XT	200 KHz	47-68 pF	47-68 pF
	1.0 KHz	15 pF	15 pF
	4.0 KHz	15 pF	15 pF
HS	4.0 KHz	15 pF	15 pF
	8.0 KHz	15-33 pF	15-33 pF
	20.0 KHz	15-33 pF	15-33 pF
	25.0 KHz	15-33 pF	15-33 pF

2.4.4.2 โหมดการป้อนสัญญาณนาฬิกาตัวต้านทานและตัวเก็บประจุ RC และ RCIO

การติดต่อออสซิลเลเตอร์ในโหมดนี้จะต่อใช้งานเพียงขาเดียวคือขา OSC1 ซึ่งการต่อใช้งานในโหมดนี้จะไม่นิยมทำมากเท่าไหร่ เนื่องด้วยการใช้งานโหมดนี้มีข้อเสียในเรื่องของอุณหภูมิซึ่งมือไมโครคอนโทรลเลอร์ทำงานเป็นเวลานานๆ อาจทำให้ไมโครคอนโทรลเลอร์ทำงานผิดพลาดได้โดยเฉพาะงานที่ต้องการความเที่ยงตรงของเวลา การใช้ออสซิลเลเตอร์แบบนี้จะทำให้เกิดความคลาดเคลื่อนได้แต่ข้อดีของการใช้ออสซิลเลเตอร์แบบนี้คือประหยัดหาซื้อง่าย ส่วนในโหมดของ RCIO จะสามารถนำขา RA6 ไปใช้งานเป็นขาอินพุตเอาต์พุตปกติได้ การต่อใช้งานแสดงดังรูป 2.4 ซึ่งจะมีตัวต้านทานและตัวเก็บประจุทำหน้าที่กำหนดค่าความถี่ โดยค่าตัวต้านทาน Rext ที่นำมาใช้นั้นจะอยู่ระหว่าง 4-100 กิโลโอห์ม ตัวเก็บประจุ Cext ที่ใช้งานได้จะต้องมากกว่า 20 pF ขึ้นไป

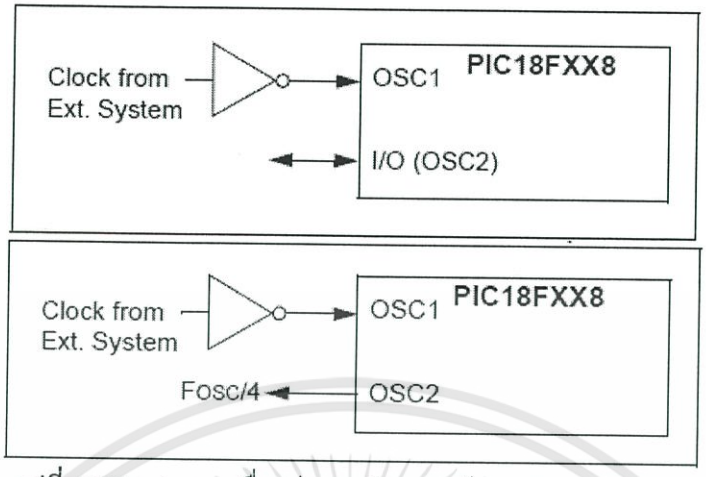


รูปที่ 2.23 รูปแสดงการต่อออสซิลเลเตอร์แบบ RC และ RCIO

2.4.4.3 โหมดการป้อนสัญญาณนาฬิกาแบบ EC และ ECIO

การทำงานในโหมดนี้จะเป็นการรับสัญญาณนาฬิกาจากภายนอกผ่านทางขา OSC1 ซึ่งขา OSC2 จะปล่อยไว้ ส่วนขา RA6/OSC2 จะมีสัญญาณนาฬิกาออกมาที่ขานี้เท่ากับความถี่สัญญาณนาฬิกาอินพุตหารด้วย 4 สามารถนำสัญญาณนาฬิกาจากขา OSC2 ไปใช้งานให้เกิดประโยชน์ได้ เช่น ใช้ในการสื่อสารซิงโครนัส (Synchronize) และอุปกรณ์ต่อพ่วงอื่นๆ แต่โหมด ECIO นั้นจะแตกต่างกันตรงที่จะสามารถนำขา RA6/OSC2 ไปใช้เป็นขาอินพุตปกติได้

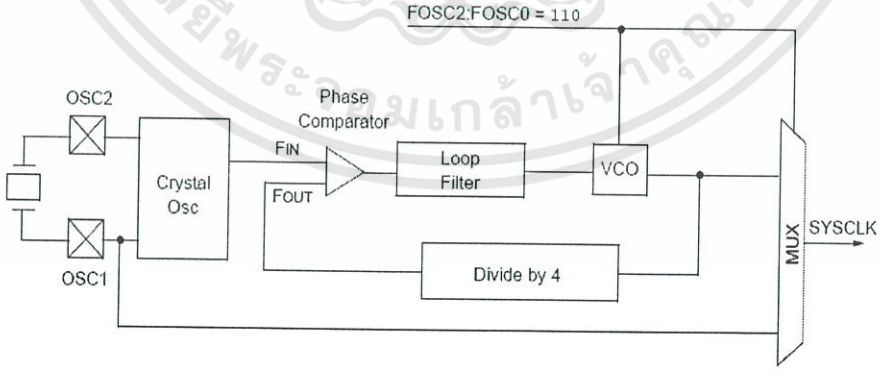
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 แสดงการเชื่อมต่อออสซิลเลเตอร์แบบ EC และ ECIO

2.4.4.4 โหมดการป้อนสัญญาณนาฬิกาโดยใช้วงจร LPL (Phase Lock Loop)

โหมดนี้เพิ่มเติมเข้ามาใหม่ในไมโครคอนโทรลเลอร์ของตระกูล PIC บางเบอร์เท่านั้นซึ่งมีไมโครคอนโทรลเลอร์ PIC18F458 การใช้งานในโหมดนี้เริ่มจากการป้อนสัญญาณนาฬิกาให้แก่ซีพียูโดยใช้วิธีต่อแบบ คริสตอลออสซิลเลเตอร์จากการกำหนดค่าให้โหมด PLL (Phase Lock Loop) ทำงานโดยการกำหนด Configuration Bit ที่บิต OSC1-OSC2 และในซอฟต์แวร์โปรแกรมซีพียู เช่นโปรแกรม Epic Win ในฟังก์ชั่น HS4 โดยซีพียูจะมีวงจร PLL อยู่ภายในทำการคูณความถี่ให้ โดยมีค่าเท่ากับสัญญาณนาฬิกาที่อินพุตคูณด้วย 4 เช่นป้อนสัญญาณนาฬิกามีค่าเท่ากับ 10 เมกะเฮิรต์ เมื่อผ่านวงจรเฟสล็อกลูปก็จะได้ค่าความถี่ออกมาเป็น 40 เมกะเฮิรต์ ซึ่งคุณสมบัติแบบนี้จะไม่มีใน PIC16F877 ซึ่งการทำงานของซีพียูจะอาศัยสัญญาณนาฬิการะบบ (Buss Clock) เป็นจุดอ้างอิงในการทำงานโดย ของไมโครคอนโทรลเลอร์จะมีค่าเท่ากับ 1 หาร 4 ของสัญญาณนาฬิกาอินพุต



รูปที่ 2.25 แสดงบล็อกไดอะแกรมการทำงานของวงจร PLL (Phase Lock Loop)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.5 การรีเซตของไมโครคอนโทรลเลอร์ PIC18F458

การรีเซตของไมโครคอนโทรลเลอร์ PIC18F458 สามารถเกิดได้หลายแหล่งดังต่อไปนี้

- การรีเซตที่เกิดขึ้นเมื่อจ่ายไฟให้ซีพียู (Power On Reset : POR)
- การรีเซตจากขารีเซตหลัก (MCLR)
- การรีเซตจากขา MCLR ในระหว่างการทำงานในโหมดประหยัดพลังงาน(Sleep Mode)
- การรีเซตจากวอตช์ด็อกไทเมอร์ (Watch Dog Timer : WDT)
- การรีเซตจากวอตช์ด็อกไทเมอร์ที่ทำให้เกิดเวกอัป (Wake Up) ในขณะทำงานในโหมดประหยัดพลังงาน(Sleep Mode)
- การรีเซตที่เกิดขึ้นจากวงจรตรวจสอบแรงดันต่ำ (Brown Out Reset : BOR)

2.4.5.1 เพาเวอร์ออนรีเซต (Power On Reset : POR)

การรีเซตแบบนี้จะเกิดขึ้นเมื่อเริ่มจ่ายไฟให้กับซีพียูโดยจะเกิดการรีเซตขึ้นหากพบว่ามีการจ่ายไฟให้กับตัวซีพียูอยู่ในระหว่างที่ซีพียูไม่สามารถทำงานได้ โดยปกติแล้วจะอยู่ในช่วง 1.2 ถึง 1.7 โวลต์ โดยในการรีเซตแบบนี้ถือว่ามีความสำคัญสูงสุดเนื่องจากเป็นการรีเซตลำดับแรกสุด ถ้าหากไม่มีการจ่ายไฟให้แก่ซีพียูก็จะไม่เกิดการรีเซตแบบนี้ขึ้น

2.4.5.2 เพาเวอร์อัปไทเมอร์

เมื่อเกิดเพาเวอร์ออนรีเซตจะมีไทเมอร์ตัวหนึ่งภายในทำงานไทเมอร์ตัวนั้นก็คือเพาเวอร์อัปไทเมอร์โดยเพาเวอร์อัปไทเมอร์จะเป็นตัวกำหนดคาบเวลาการเกิดไทม์เอาต์ที่ 72 มิลลิวินาทีหลังการเกิดเพาเวอร์ออนรีเซต ทั้งนี้ก็เพื่อที่จะไม่ให้คอนโทรลเลอร์รอให้ส่วนต่างๆ ของซีพียูพร้อมที่จะทำงานก่อนที่จะมีการรีเซตเกิดขึ้น สำหรับการปิดการทำงานของเพาเวอร์อัปไทเมอร์ตัวนี้สามารถทำได้ที่รีจิสเตอร์ Configuration ที่บิต Pwrte]

2.4.5.3 ออสซิลเลเตอร์สตาร์ตอัปไทเมอร์ (Oscillator Start Up Timer : OST)

ออสซิลเลเตอร์แบบนี้จะทำหน้าที่หน่วงเวลาทำงานของไมโครคอนโทรลเลอร์ไปอีก 1024 ไซเคิล จากอินพุตที่ขาสัญญาณนาฬิกา OSC1 โดยจะเกิดขึ้นหลังในส่วนการเกิดเพาเวอร์อัปไทเมอร์ (หากมีการเปิดทำงานไว้) ทั้งนี้เพื่อที่จะให้แน่ใจว่าส่วนกำเนิดนาฬิกาหรือวงจรออสซิลเลเตอร์นั้นทำงานได้คงที่หรือมีเสถียรภาพพอแล้ว โดยการทำงานในส่วนนี้จะเกิดขึ้นได้ก็ต่อเมื่อเราเลือกใช้โหมดสัญญาณนาฬิกาเป็นแบบ LP XT และ HS และเมื่อการใช้เพาเวอร์ออนรีเซตหรือเมื่อเกิดการเวกอัป (Wake Up) จากการทำงานในโหมดสลีป (Sleep Mode)

2.4.5.4 บราวเอาต์รีเซต (Brown Out Reset : BOR)

บราวเอาต์รีเซตนี้จะทำหน้าที่รีเซตซีพียูในขณะที่ไฟเลี้ยงจ่ายให้กับซีพียูตกลงจนต่ำกว่าแรงดันที่กำหนดไว้ (VBOR) โดยปกติแล้วจะอยู่ที่ 4 โวลต์ โดยหากมีการตรวจสอบพบว่าแรงดันที่ไฟเลี้ยง VDD มีขนาดลดลงต่ำกว่าแรงดันที่ VBOR เป็นเวลานานเกิน 100 นาโนวินาที ก็จะส่งสัญญาณไปทำการรีเซตซีพียูแต่ถ้าหากแรงดัน VDD ลดต่ำกว่า VBOR ในระยะเวลาไม่เกิน 100 นาโนวินาที กระบวนการรีเซตซีพียูจะไม่เกิดขึ้น สำหรับประโยชน์ของการรีเซตแบบนี้คือจะช่วยแก้ปัญหาการทำงานที่ผิดพลาดของซีพียูที่เกิดจากแรงดันที่ไม่คงที่เมื่อเกิดการรีเซตจากบราวเอาต์รีเซต สภาวะการรีเซตนี้จะยังคงค้างอยู่จนกว่าแรงดันไฟที่เลี้ยง VDD จะมีค่าสูงกว่าแรงดัน VBOR อีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ครั้งซึ่งหลังจากที่แรงดันที่ VDD กลับมามีค่ามากกว่าแรงดัน VBOR แล้วจะเกิดการหน่วงเวลาไปอีก 72 มิลลิวินาที จากวงจรเพาเวอร์อัปไทเมอร์ หากมีการเปิดการทำงานของเพาเวอร์อัปไทเมอร์เอาไว้ ปกติสามารถกระทำได้โดยผ่านทางซอฟต์แวร์ก่อนการโปรแกรมสำหรับโปรแกรมดังกล่าวที่มีฟังก์ชั่นนี้ ก็เช่นโปรแกรม Epic Win ซึ่งในช่วงระหว่างการหน่วงเวลา 72 มิลลิวินาที ของเพาเวอร์อัปไทเมอร์นี้ หากแรงดัน VDD ตกลงต่ำกว่า VBOR อีก การทำงานของบราวเอาต์รีเซตนี้ก็จะถูกทำการรีเซตกลับไปเริ่มใหม่ทันที

2.4.6 ลำดับการเกิดไทม์เอาต์ (Time Out Squence)

จากการทำงานในส่วนต่างๆกว่าที่ซีพียูจะเกิดการรีเซตได้นั้นจะต้องผ่านขั้นตอนต่างๆ หลายๆ ขั้นตอนนั้นก็เพื่อให้การทำงนของซีพียูถูกต้องแม่นยำและมีเสถียรภาพในหารทำงานมากขึ้น ฯนตารางที่ 2.3 แสดงค่าเวลาที่เกิดขึ้นจากการรีเซตของซีพียูที่เกิดขึ้นในแต่ละแบบโดยหลังจากเกิดเพาเวอร์ออนรีเซต หากทำการปิดการทำงานของเพาเวอร์อัปไทเมอร์สัญญาณรีเซตจะถูกหน่วงเวลาไปอีกเป็นเวลา 72 มิลลิวินาที หลังจากนั้นภาคออสซิลเลเตอร์สตาร์ตอัปไทเมอร์จะหน่วงเวลาของออสซิลเลเตอร์ไปอีกจำนวน 1024 ไซเคิล และเมื่อเสร็จสิ้นกระบวนการดังกล่าวตามที่ได้กล่าวมาแล้ว กระบวนการรีเซตของซีพียูจึงจะเกิดขึ้น ส่วนบราวเอาต์รีเซตจะมีลักษณะเดียวกันกับการรีเซตของเพาเวอร์ออนรีเซต ส่วนการเกิดเวกอัป (Wake Up) จะมีการหน่วงเวลาการทำงานของซีพียูไปเป็น 1024 ไซเคิล เพื่อรอให้ระดับกระแสและแรงดันที่จ่ายให้ซีพียูคงที่หรือมีเสถียรภาพมากขึ้น

ตารางที่ 2.3 ตารางแสดงค่าเวลาต่างๆ ของแต่ละส่วนก่อนการเกิดไทม์เอาต์

ออสซิลเลเตอร์	เพาเวอร์อัปไทเมอร์		เพาเวอร์ออนรีเซต	การเวกอัปจากโหมดสลีป
	PWRTEN = 0	PWRTEN = 1		
HS4	72 มิลลิวินาที + 1024 ไซเคิล + 2 มิลลิวินาที	1024 ไซเคิล + 2 มิลลิวินาที	72 มิลลิวินาที + 1024 ไซเคิล + 2 มิลลิวินาที	1024 ไซเคิล + 2 มิลลิวินาที
HS, XT, LP	72 มิลลิวินาที + 1024 ไซเคิล	1024 ไซเคิล	72 มิลลิวินาที + 1024 ไซเคิล	1024 ไซเคิล
RC	72 มิลลิวินาที	-	72 มิลลิวินาที	-
External RC	72 มิลลิวินาที	-	72 มิลลิวินาที	-

2.4.7 หน่วยความจำ

หน่วยความจำเป็นส่วนที่เก็บข้อมูลและโค๊ดโปรแกรมต่างๆ หน่วยความจำของไมโครคอนโทรลเลอร์ PIC18F458 มีด้วยกันอยู่ 3 ส่วนดังนี้

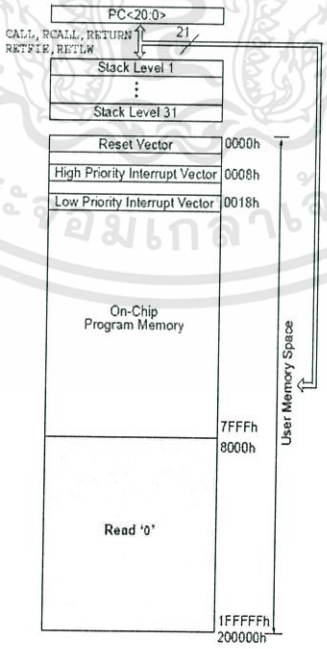
- หน่วยความจำโปรแกรม (Program Memory)
- หน่วยความจำข้อมูล (Data Memory)
- หน่วยความจำอีอีพรอม (EEPROM)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.7.1 หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรมเป็นส่วนที่ใช้เก็บซอร์สโค๊ดของโปรแกรม โดยหน่วยความจำของ PIC18F458 จะเป็นแบบแฟลช (Fash Memory) ที่สามารถทำการเขียนและลบข้อมูลได้หลายพันครั้งโดยโครงสร้างชุดคำสั่งของ PIC18F458 จะเป็น 16 บิต ซึ่งลักษณะการอ้างถึงข้อมูลจะอ้างได้ถึง 16 บิต เช่นหน่วยความจำโปรแกรมของ PIC18F458 มีขนาด 32 กิโลไบต์ซึ่งถือว่าเพียงพอต่อการใช้งานแล้วหลังจากที่ทำการเขียนในขั้นตอนของการโปรแกรมแล้วก็จะมีไว้สำหรับอ่านออกมาได้เพียงอย่างเดียวเท่านั้น รูปที่ 2.7 ได้แสดงพื้นที่การจัดสรรหน่วยความจำโปรแกรมของ PIC18F458 และ PIC18F258 ซึ่งมีโครงสร้างเหมือนกันโดยการจัดสรรนั้นได้ถูกจัดวางไว้อยู่ที่ตำแหน่ง 0000h-1FFFFFh สำหรับแอดเดรส 0000h-07FFFh จะสงวนไว้สำหรับเก็บค่ารีเซตเว็กเตอร์ (Reset Vector) และที่ตำแหน่ง 0008h และ 0018h จะมีไว้เก็บค่าอินเตอร์รัพต์เว็กเตอร์ (Interrupt Vector) ไบต์สูงและต่ำดังนั้นในการเขียนโปรแกรมจะต้องมาเริ่มต้นที่แอดเดรส 0020h จึงจะเหมาะสมที่สุดแต่ถ้าหากผู้เขียนโปรแกรมคาดว่าไม่มีการใช้งานการอินเตอร์รัพต์แล้วก็จะละเลยในส่วนของการสงวนนี้ได้

ไมโครคอนโทรลเลอร์ PIC18F458 ยังมีพื้นที่หน่วยความจำพิเศษสำหรับเก็บค่าของโปรแกรมเคาน์เตอร์ชั่วคราวเรียกว่าสแต็ก (Stack) ซึ่งจะมีบทบาทมากในการกระโดดไปทำงานยังโปรแกรมย่อยโดยหากเมื่อมีการกระทำคำสั่งให้กระโดดไปทำงานยังโปรแกรมย่อย ซีพียูจะทำการเก็บค่าโปรแกรมเคาน์เตอร์ (Program Counter : PC) ในขณะนั้นไว้สแต็ก จากนั้นจึงกระโดดไปทำงานยังโปรแกรมย่อยและเมื่อทำงานเรียบร้อยแล้วซีพียูก็จะอ่านค่าโปรแกรมเคาน์เตอร์ที่อยู่ในสแต็กกลับคืนมาแล้วไปทำงานยังโปรแกรมหลักต่อไป สำหรับสแต็กภายใน PIC18F458 นี้จะสามารถเก็บค่าได้มากถึง 31 ระดับซึ่งมีค่ามากกว่า PIC18F458 ที่มีขนาดเพียง 8 ระดับเท่านั้นและในส่วนของโปรแกรมเคาน์เตอร์ของ PIC18F458 จะมีขนาด 21 บิต ซึ่งใน PIC16F877 จะมีขนาดเพียง 13 บิต อาจกล่าวได้ว่าถ้าสแต็กมีค่ามากก็จะสามารถเก็บค่าโปรแกรมเคาน์เตอร์ได้มากนั่นเอง



รูปที่ 2.26 แสดงการจัดสรรหน่วยความจำโปรแกรมของ PIC18F458

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.7.2 สแต็ก (Stack)

ในไมโครคอนโทรลเลอร์ PIC18F458 ได้จัดสรรหน่วยความจำที่เรียกว่า สแต็ก ไว้เก็บค่าของโปรแกรมเคาน์เตอร์ชั่วคราวได้ 31 ระดับ โดยในแต่ละระดับจะมีขนาด 21 บิต ถ้าหากมีการพุช (Push) 1 ครั้งสแต็กก็จะทำการเก็บค่าของโปรแกรมเคาน์เตอร์ไว้ 1 ค่าโดยในไมโครคอนโทรลเลอร์ PIC18F458 จะสามารถเก็บค่าหรือรับรองได้ 31 ครั้ง หากมีการพุช (Push) เป็นครั้งที่ 32 ซีพียูก็จะนำค่าของโปรแกรมเคาน์เตอร์เก็บลงสแต็กในครั้งที่ 1 สำหรับการพุช (Push) หมายถึงการที่ซีพียูจะกระโดดไปทำคำสั่งถัดไปซีพียูก็จะทำการเก็บค่าของโปรแกรมเคาน์เตอร์ไว้ในหน่วยความจำชั่วคราวซึ่งหน่วยความจำชั่วคราวนั้นคือสแต็ก (Stack) นั่นเองเมื่อซีพียูกระทำคำสั่งในโปรแกรมย่อยเรียบร้อยแล้วซีพียูก็จะไปทำการอ่านค่าที่โปรแกรมเคาน์เตอร์ที่เก็บไว้ในสแต็กกลับมากระทำคำสั่งในแอดเดรสถัดไป

2.4.7.3 รีจิสเตอร์โปรแกรมเคาน์เตอร์ PCL, PCLATH และ PCLATL

สำหรับโปรแกรมเคาน์เตอร์ (Program Counter : PC) ในซีพียู PIC18F458 มีขนาด 21 บิตโดยจะแบ่งออกเป็น 2 ส่วนคือ PCL และ PCH ไบต์ต่ำสุดคือ PCL ไบต์สูงสุดคือ PCH โดยสามารถอ่านและเขียนค่าได้ในรีจิสเตอร์ PCL ส่วน PCH ในรีจิสเตอร์ PC บิตที่ 15 และ 8 จะไม่สามารถอ่านและเขียนค่าได้โดยตรงซึ่งหากต้องการที่จะอ่านและเขียนข้อมูลนั้นจะต้องกระทำผ่านรีจิสเตอร์ PCLATH และเมื่อเกิดรีเซตทุกครั้งรีจิสเตอร์ PC จะถูกเคลียร์ค่าเป็นศูนย์ทั้งหมดและมีผลทำให้ตำแหน่งเริ่มต้นที่ซีพียูทำงานนั้นไปอยู่ที่ตำแหน่ง 0x000

2.4.7.4 จังหวะและไซเคิลการทำงานของ PIC18F458

สัญญาณนาฬิกาหลักภายในของไมโครคอนโทรลเลอร์ PIC18F458 จะประกอบไปด้วยสัญญาณนาฬิกาจากภายนอกจำนวน 4 ไซเคิล (OSC/4) ประกอบไปด้วย Q1, Q2, Q3 และ Q4 โดยสัญญาณนาฬิกาหลักภายใน ของซีพียูจะมีค่าเท่ากับ ¼ ของสัญญาณนาฬิกาจากภายนอกที่รับเข้ามาโดย Tosc คือ คาบเวลาของสัญญาณนาฬิกาภายนอก และ TCY คือ คาบเวลาของสัญญาณนาฬิกาหลักภายใน โดยแต่ละช่วงการทำงานจะมีลักษณะดังต่อไปนี้

Q1 : ถอดรหัสคำสั่ง (Decode) หรืออยู่ในสภาวะ No Operation

Q2 : อ่านคำสั่งข้อมูล (Read Data Cycle) หรืออยู่ในสภาวะ No Operation

Q3 : ประมวลผลข้อมูล (Process The Data)

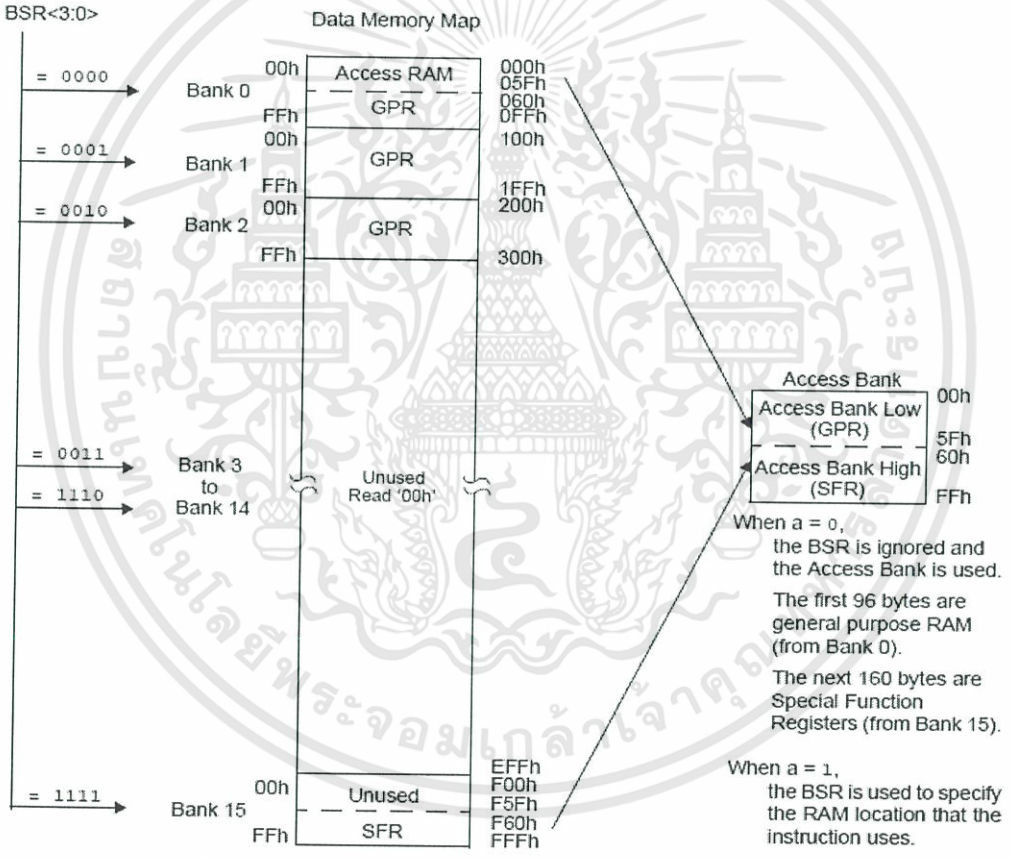
Q4 : เขียนรหัสข้อมูล (Write Data Cycle)

รีจิสเตอร์ PC (Program Counter) จะเพิ่มขึ้นทีละ 1 ค่าทุกๆไซเคิลการทำงานที่ 1 (Q1) และจะมีการอ่านข้อมูลคำสั่ง (Source Code) หรือสภาวะการเฟตช์ (Fetch) ข้อมูลขบวนการถอดรหัสคำสั่งและการปฏิบัติคำสั่ง (Execute) จะเกิดขึ้นอยู่ในช่วงการทำงานที่ 1 ถึง 4 การประมวลผลของไมโครคอนโทรลเลอร์ PIC18F458 สามารถประมวลผล 1 Clock ต่อ 1 คำสั่งนั่นหมายถึงจะเป็นลักษณะ Clock ภายในประกอบด้วยสัญญาณนาฬิกาจากภายนอก 4 ไซเคิล และเหตุที่ซีพียูในตระกูล PIC นี้สามารถประมวลผลสัญญาณ 1 ไซเคิลต่อคำสั่งได้ เพราะรูปแบบการประมวลผลนั้นจะเป็นแบบไปป์ไลน์ (Pipe line) คือจะมีลักษณะการทำงานที่เหลื่อมล้ำกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.7.5 หน่วยความจำข้อมูล (Data Memory)

หน่วยความจำข้อมูลของ PIC18F458 เป็นหน่วยความจำชนิด Static Ram ซึ่งสามารถเปลี่ยนแปลงข้อมูลได้ตลอดเวลา หน่วยความจำข้อมูลของ PIC18F458 มีขนาด 1536 ไบต์ (Byte) การจัดแบ่งพื้นที่การใช้งานต่างๆ จะแบ่งเป็นพื้นที่ที่เป็นรีจิสเตอร์ที่ใช้งานทั่วไป (General Purpose Register : GPR) ซึ่งภายใน PIC18F458 จะมีทั้งหมด 6 แบงก์ (Bank) ของพื้นที่แบงก์ข้อมูลทั้งหมดคือ 15 แบงก์ โดยใน 6 แบงก์ของ GPR นั้นจะมีขนาดแบงก์ละ 96 ไบต์ เท่ากันหมด โดยแอดเดรสเริ่มต้นคือ 000h ถึง 5FFh สำหรับรีจิสเตอร์ฟังก์ชั่นพิเศษ (Special Function Register : FSR) มีพื้นที่ทั้งหมด 160 ไบต์ โดยจะอยู่ที่ แบงก์ 15 ข้อดีอีกอย่างหนึ่งของ PIC18F458 คือในการเขียนโปรแกรมผู้เขียนไม่จำเป็นต้องเลือกแบงก์หรือสลับแบงก์ไปมาเหมือนไมโครคอนโทรลเลอร์เบอร์อื่นๆเนื่องจากว่าหน่วยความจำของไมโครคอนโทรลเลอร์ PIC18F458 ไม่ถูกแบ่งเป็นเพจ (Page) คือในการจัดสรรแบงก์ข้อมูลจะอยู่ในเพจเดียวกัน



รูปที่ 2.27 แสดงการจัดสรรหน่วยความจำข้อมูลของ PIC18F458

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

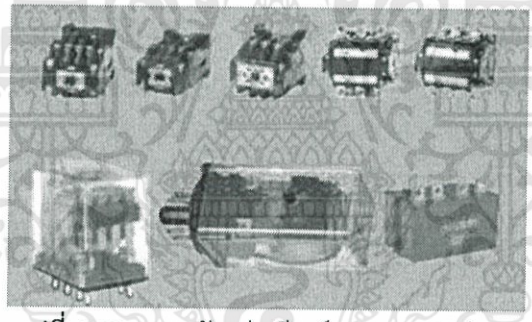
2.5 รีเลย์ (Relay)

รีเลย์ (Relay) คือ อุปกรณ์อิเล็กทรอนิกส์ที่ทำหน้าที่ ตัด-ต่อวงจร โดยอาศัยหลักการของอำนาจแม่เหล็กไฟฟ้า เพื่อควบคุมการเปลี่ยนแปลงของหน้าสัมผัส รีเลย์มีความหมายในแบบของนักอิเล็กทรอนิกส์ว่า “ตัวถ่ายทอดกำลัง” เพราะเราป้อนกำลังงานไฟฟ้าให้แก่รีเลย์เพียงเล็กน้อย ก็สามารถควบคุมวงจรกำลังงานสูงๆ ที่ต่ออยู่กับหน้าสัมผัสของรีเลย์ได้ โดยเมื่อทำการจ่ายกระแสไฟฟ้าให้กับขดลวดรีเลย์ (Coil) จะทำให้เกิดสนามแม่เหล็กรอบขดลวด ซึ่งอำนาจแม่เหล็กชั่วคราวที่เกิดขึ้นมีค่าเพียงพอที่จะชนะแรงสปริง ทำให้เกิดการเปลี่ยนแปลงที่หน้าสัมผัส รีเลย์มีรูปร่างและขนาดที่แตกต่างกัน ในการเลือกใช้งานรีเลย์ จะต้องคำนึงถึงชนิดของรีเลย์ อัตรากำลังสูงสุดที่รีเลย์สามารถทนได้ ความถี่ใช้งานและอื่นๆ เพื่อให้สามารถใช้งานรีเลย์ได้อย่างถูกต้องและเหมาะสม

2.5.1 ชนิดของรีเลย์

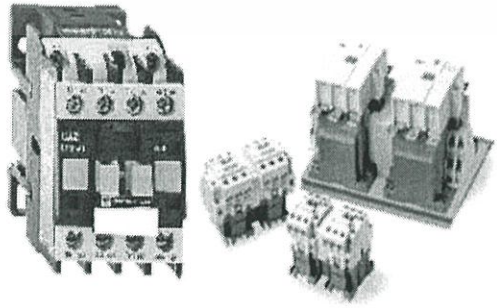
การแบ่งชนิดของรีเลย์ตามชนิดของการควบคุมที่ทำงานด้วยไฟฟ้า สามารถแบ่งได้เป็น 2 ประเภท คือ

1) รีเลย์ควบคุม (Control Relay) จะเป็นรีเลย์ที่มีขนาดเล็ก ใช้กำลังไฟฟ้าต่ำ ใช้ในวงจรควบคุมทั่วไปที่มีกำลังไฟฟ้าไม่มากนัก หรือเพื่อการควบคุมรีเลย์หรือคอนแทกเตอร์ขนาดใหญ่ รีเลย์ควบคุมนิยมเรียกกันง่าย ๆ ว่า "รีเลย์"



รูปที่ 2.28 แสดงตัวอย่างรีเลย์ควบคุมแบบต่างๆ

2) รีเลย์กำลัง (Power Relay) นิยมเรียกกันว่าคอนแทกเตอร์(Contactor or Magnetic Contactor) ซึ่งเป็นรีเลย์ที่มีขนาดใหญ่กว่ารีเลย์ควบคุม นิยมใช้งานกับกำลังไฟฟ้าสูงดังแสดงในรูปที่ 2 ส่วนใหญ่ใช้ในการควบคุมไฟฟ้ากำลังที่มีขนาดใหญ่ เช่นการควบคุมมอเตอร์ไฟฟ้าสามเฟส เป็นต้น

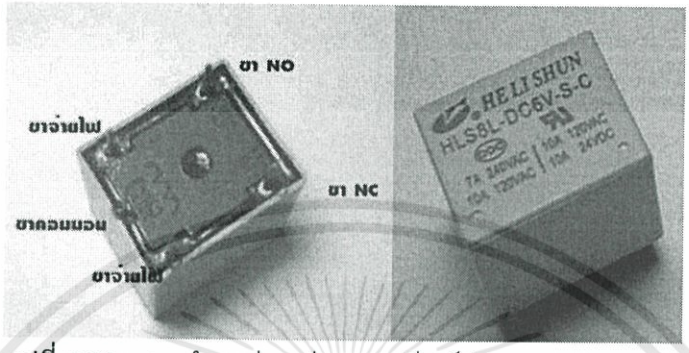


ที่รูป 2.29 แสดงตัวอย่างคอนแทกเตอร์แบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 รีเลย์ (HLS8L-DC6V-S-C)

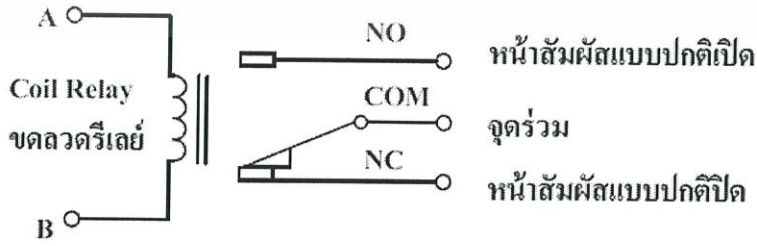
ใช้ในการควบคุมเครื่องใช้ไฟฟ้าทั้งหมด รีเลย์ที่ใช้เป็นเพาเวอร์รีเลย์แรงดัน 5 V DC ที่ทน กระแสได้ 10 A



รูปที่ 2.30 แสดงตำแหน่งขาต่างๆ ของรีเลย์ (HLS8L-DC6V-S-C)

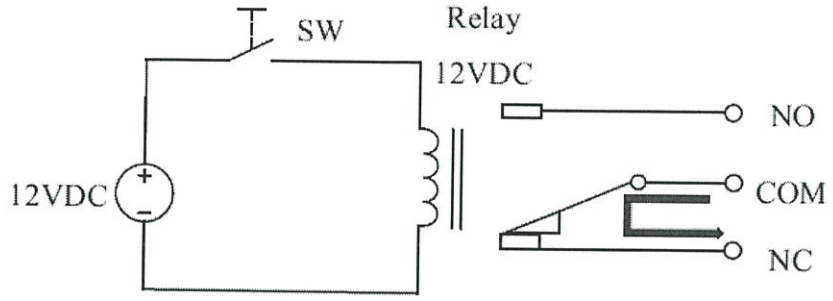
หลักการของรีเลย์ คือ รีเลย์จะทำงานตามหลักการแม่เหล็กไฟฟ้า เมื่อเรานำเอาขดลวดพัน รอบแกนเหล็กหลายรอบแล้วป้อนกระแสไฟฟ้า เข้าในขดลวดนั้น แกนเหล็กจะกลายเป็น แม่เหล็กอำนาจแม่เหล็กจะดึงหน้าสัมผัส COM มาต่อกับหน้าสัมผัส NO (Normal Open) ทำให้ กระแสไฟฟ้าไหลผ่าน NO และ COM แต่ถ้าหากไม่ป้อนกระแสไฟฟ้าเข้าไปแกนเหล็กก็จะไม่มีอำนาจแม่เหล็กทำให้หน้าสัมผัสไม่เปลี่ยนแปลงนั่นก็คือ COM ต่อกับหน้าสัมผัส NC (Normal Close)

- **ส่วนประกอบต่างๆ ของรีเลย์ (HLS8L-DC6V-S-C)**
  - ขาจ่ายแรงดันใช้งาน ซึ่งจะมีอยู่ 2 ขา จากรูปจะเห็นสัญลักษณ์ขดลวดแสดงตำแหน่งขา coil หรือ ขาต่อแรงดันใช้งาน
  - ขา C หรือ COM หรือ ขาคอมมอน จะเป็นขาต่อระหว่าง NO และ NC
  - ขา NO (Normally Open) โดยปกติขานี้จะเปิดเอาไว้ จะทำงานเมื่อเราป้อนแรงดันให้รีเลย์
  - ขา NC (Normally Close) โดยปกติขานี้จะต่อกับขา C ในกรณีที่เราไม่ได้จ่ายแรงดันหน้าสัมผัสของ C และ NC จะต่อถึงกัน

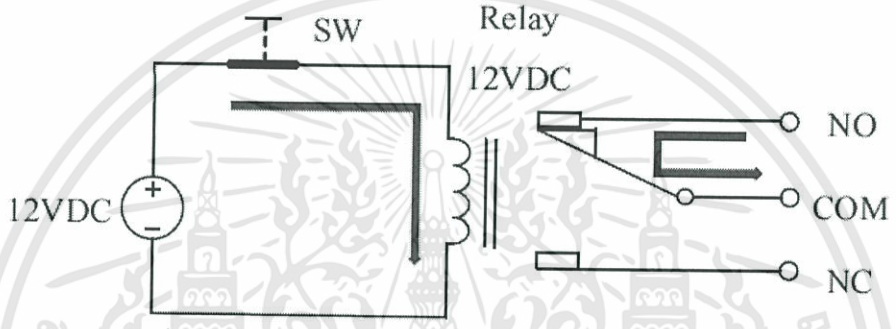


รูปที่ 2.31 แสดงสัญลักษณ์ของรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.32 แสดงรีเลย์ขณะที่ไม่จ่ายกระแสไฟฟ้าแก่ขดลวดรีเลย์



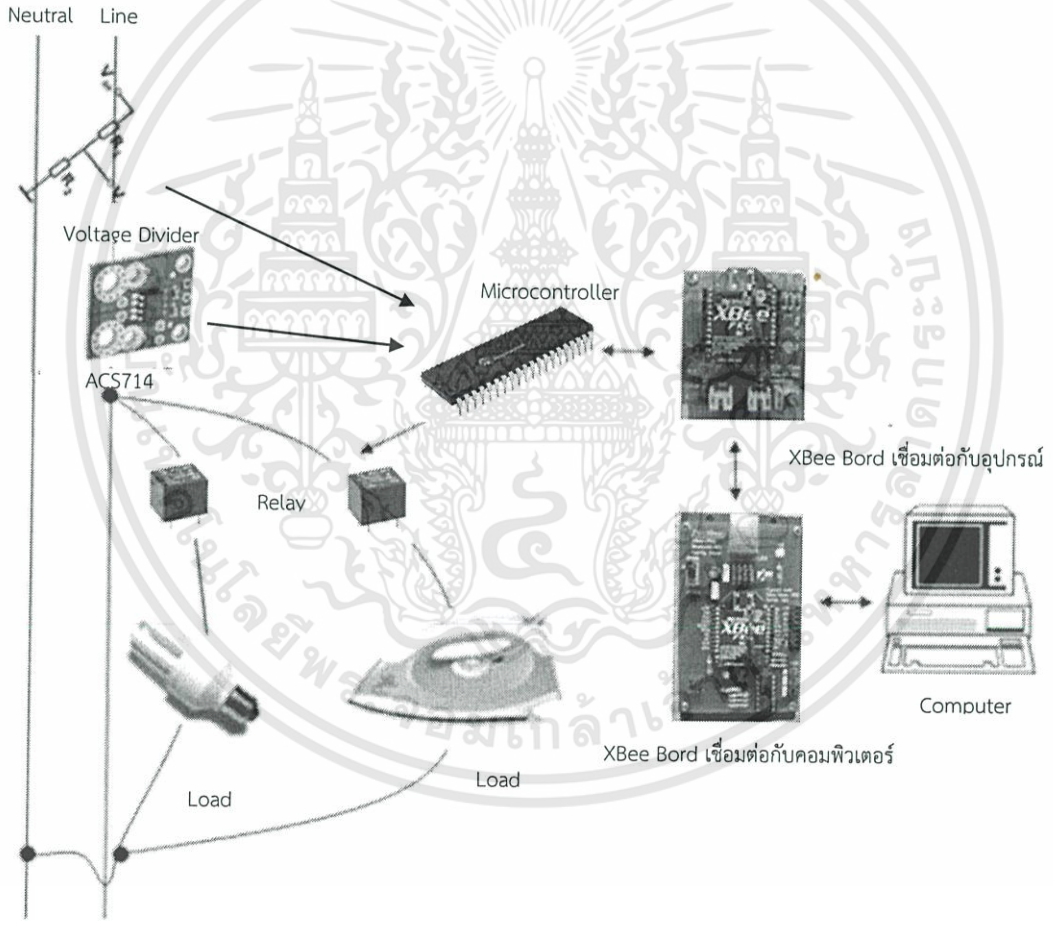
รูปที่ 2.33 แสดงรีเลย์ขณะจ่ายกระแสไฟฟ้าแก่ขดลวดรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีดำเนินงาน

3.1 ภาพรวมของการวางแผนโครงสร้างของโครงงาน

ในการออกแบบการทำงานระบบแจ้งเตือนเมื่อใช้พลังงานเกินโดยใช้ซิกบี เราใช้ทฤษฎีทั้งหมดในการออกแบบระบบ ซึ่งประกอบด้วยส่วนของการออกแบบอุปกรณ์ควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้าต่างๆ การออกแบบอุปกรณ์ตรวจวัดพลังงานทางไฟฟ้าภายในบ้าน การประมวลผลด้วยไมโครคอนโทรเลอร์ การออกแบบการสื่อสารไร้สาย และการแสดงผลการทำงานของอุปกรณ์ไฟฟ้า การตรวจวัดพลังงานทางไฟฟ้าใช้ไฟฟ้า ผ่านโปรแกรม Visual studio บนหน้าจอคอมพิวเตอร์ ซึ่งภาพรวมการทำงานได้แสดงในรูปที่ 3.1



รูปที่ 3.1 แสดงโครงสร้างของอุปกรณ์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ส่วนการทำงานโดยรวม

3.2.1 ส่วนการวัดพลังงานไฟฟ้า

3.2.1.1 Voltage Divider

3.2.1.2 Current Sensor (ACS714 Current Sensor Carrier -5 to +5A)

3.2.1.3 ไมโครคอนโทรลเลอร์ (PIC18F458)

3.2.2 ส่วนควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้า

3.2.2.1 Relay

3.2.2.2 ไมโครคอนโทรลเลอร์ (PIC18F458)

3.2.3 ส่วนการแสดงผลและติดต่อกับผู้ใช้ (คอมพิวเตอร์)

3.2.4 ส่วนการสื่อสารและรับส่งข้อมูล

3.2.2.1 เอ็กซ์บี (XBee)

3.2.2.2 ADX-XBee3V บอร์ดใช้งานโมดูล XBee กับอุปกรณ์

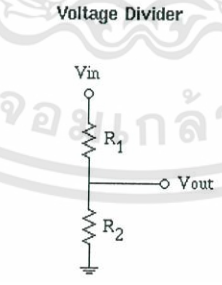
3.2.2.3 ZX-XBeeU บอร์ดเชื่อมต่อคอมพิวเตอร์

3.2.1 ส่วนการวัดพลังงานไฟฟ้า (Measuring Section)

ในส่วนนี้เป็นการวัดและคำนวณค่าของพลังงานไฟฟ้า ที่อยู่ในรูปแบบต่างๆ ดังนั้น ในส่วนนี้จึงมีอุปกรณ์ย่อยๆ ที่สำคัญต่างๆดังนี้

3.2.1.1 Voltage Divider

เนื่องจากไมโครคอนโทรลเลอร์ PIC18F458 รับ voltage input ได้ไม่เกิน 5V ในส่วนนี้จึงทำหน้าที่หารแรงดันลง โดยผู้จัดทำโครงงานจะใช้หม้อแปลงแรงดันจาก 220V AC แปลงเป็น 9V AC เพื่อให้ง่ายต่อการหารแรงดันลงมา ให้เหมาะสมกับ ไมโครคอนโทรลเลอร์ ที่นำมาใช้



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

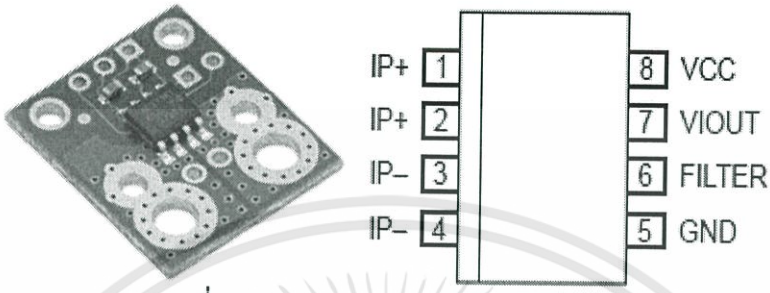
รูปที่ 3.2 วงจร Voltage Divider และสูตรการคำนวณ

คณะผู้จัดทำโครงงานได้ออกแบบ แรงดันที่จะหาร เท่ากับแรงดันที่หม้อแปลงแปลงลงมา คือ $9V_{rms}$ และใช้ R_1= 5k และ R_2= 1k เมื่อทำการหารแรงดัน จะได้ $V_{out} = 1.5V_{rms}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.2 Current Sensor

เนื่องจากไมโครคอนโทรลเลอร์ PIC18F458 รับกระแสเข้ามาตรงๆ เลยไม่ได้ ดังนั้นจึงต้องแปลงกระแสเป็นแรงดันก่อน และ voltage input นั้นก็ต้องไม่เกิน ± 5V ในส่วนนี้จึงทำหน้าที่เปลี่ยนกระแสเป็นแรงดันและหารแรงดันลงมาอีก

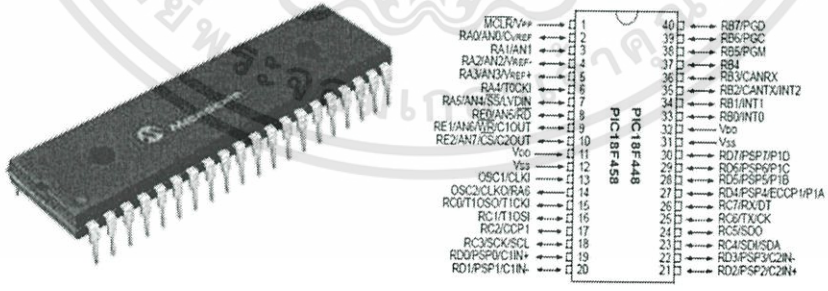


รูปที่ 3.3 Current Sensor เบอร์ ACS714
และโครงสร้าง Pin Configuration ที่ใช้งาน

ผู้จัดทำโครงงานได้ออกแบบ โดยใช้ Current Sensor Carrier -5 to +5A เป็นเซนเซอร์ตรวจวัดกระแสใช้ IC เบอร์ ACS714 ซึ่งมีค่า Sensitivity = 185mV/A มี Center 2.5V วัดได้ทั้งกระแสไฟ DC และ AC ดังนั้น Current Sensor ตัวนี้ จะส่งค่ากระแสไปให้ไมโครคอนโทรลเลอร์ PIC18F458 มากที่สุดคือ 13.5A โดยที่ ไมโครคอนโทรลเลอร์ PIC18F458 ทำงานอย่างเป็นปกติ

3.2.1.3 ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ ทำหน้าที่ในการควบคุมการทำงานทั้งหมด ในส่วนการวัดพลังงานไฟฟ้า และยังทำหน้าที่ในการคำนวณค่าต่างๆ คือ กระแส แรงดัน และกำลังไฟฟ้าจริง ไมโครคอนโทรลเลอร์ ยังสามารถส่งข้อมูลไปยังส่วนการสื่อสารและรับส่งข้อมูลเพื่อส่งต่อไปถึงส่วนการแสดงผลและติดต่อกับผู้ใช้ โดยผ่าน เอ็กบี



รูปที่ 3.4 ไมโครคอนโทรลเลอร์

ผู้จัดทำโครงงานได้เลือกใช้ไมโครคอนโทรลเลอร์ PIC18F458 ซึ่งโปรแกรมโดยภาษาซี ด้วยโปรแกรม mikroC เหตุที่เลือกใช้เบอร์นี้เพราะว่า มีราคาค่อนข้างถูก สะดวกในการใช้งาน แต่มีข้อจำกัดตรงที่ความละเอียดของการแปลงสัญญาณแอนาลอกเป็นดิจิตอลมีเพียง 10 bits

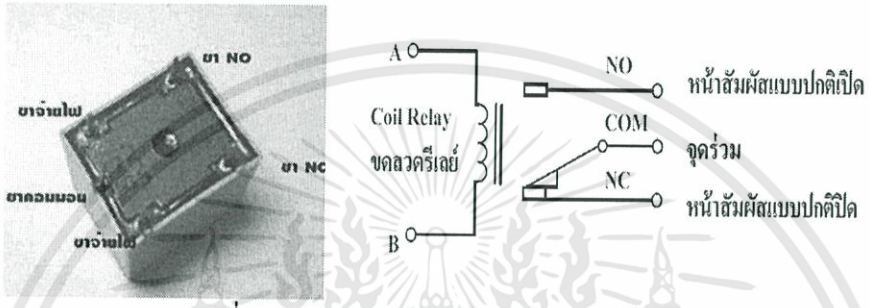
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 ส่วนควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้า

ในส่วนนี้เป็นการควบคุมการทำงานของอุปกรณ์ไฟฟ้า ซึ่งจะใช้ไมโครคอนโทรลเลอร์ในการสั่งงานให้ Relay ทำงาน เพื่อควบคุมการใช้พลังงานไฟฟ้าให้เกิดประโยชน์สูงสุด

3.2.2.1 Relay

เป็นอุปกรณ์ที่นำมาทำเป็นสวิตช์ทางด้านอิเล็กทรอนิกส์ โดยจะต้องป้อนกระแสไฟฟ้าให้ไหลผ่านขดลวดจำนวนหนึ่ง เพื่อใช้ Relay ควบคุมการทำงานของอุปกรณ์ไฟฟ้าที่ต่ออยู่กับหน้าสัมผัสหรือคอนแทกต์ของรีเลย์ แทนการใช้สวิตช์หรือใช้งานร่วมกับสวิตช์



รูปที่ 3.5 ภาพแสดงตำแหน่งขาต่างๆ ของรีเลย์ และสัญลักษณ์ของรีเลย์

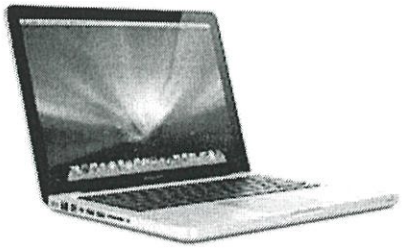
ผู้จัดทำโครงงานได้เลือกใช้ Relay 5V (HLS8L-DC6V-S-C) สำหรับควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้ากำลัง 220V AC ซึ่งใช้คำสั่งจากไมโครคอนโทรลเลอร์ในการควบคุมการทำงานของ Relay

3.2.2.2 ไมโครคอนโทรลเลอร์ (PIC18F458)

คุณสมบัติของไมโครคอนโทรลเลอร์ตัวนี้เป็นไมโครคอนโทรลเลอร์ตัวเดียวกับส่วนการวัดค่าพลังงานไฟฟ้า แต่ทำหน้าที่ในส่วนของการความคุมการทำงานของ Relay เพื่อควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้า

3.2.3 ส่วนการแสดงผลและติดต่อกับผู้ใช้

ส่วนนี้มีไว้เพื่อให้ผู้ใช้สามารถดูข้อมูลค่าพลังงานไฟฟ้าต่างๆ และควบคุมการทำงานของอุปกรณ์ไฟฟ้าได้ ซึ่งส่วนนี้ผู้จัดทำโครงงานได้เลือกใช้คอมพิวเตอร์ในการแสดงผลโดยผ่านโปรแกรม Visual Studio และควบคุมการเปิดปิดอุปกรณ์ไฟฟ้าต่างๆ ได้อีกด้วย



รูปที่ 3.6 คอมพิวเตอร์แสดงผลและติดต่อกับผู้ใช้

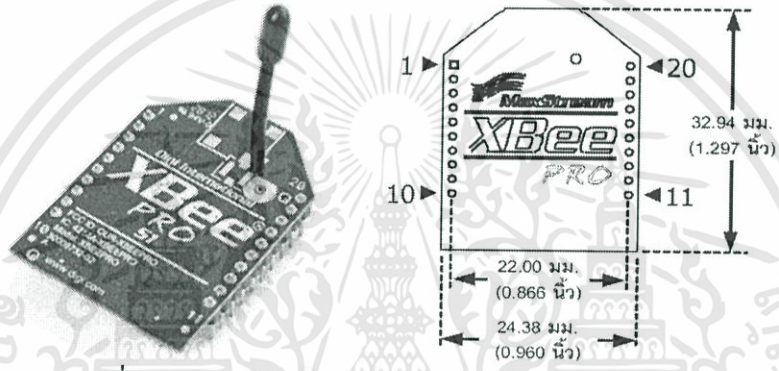
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 ส่วนการสื่อสารและรับส่งข้อมูล

ส่วนนี้เป็นส่วนที่ใช้ติดต่อสื่อสารระหว่าง ไมโครคอนโทรลเลอร์ กับ ส่วนแสดงผลและติดต่อกับผู้ใช้ เพื่อใช้ในการควบคุมสั่งการไมโครคอนโทรลเลอร์ และรับค่าจากไมโครคอนโทรลเลอร์มาแสดงผลบนหน้าจอคอมพิวเตอร์

3.2.4.1 เอ็กซ์บี (XBee)

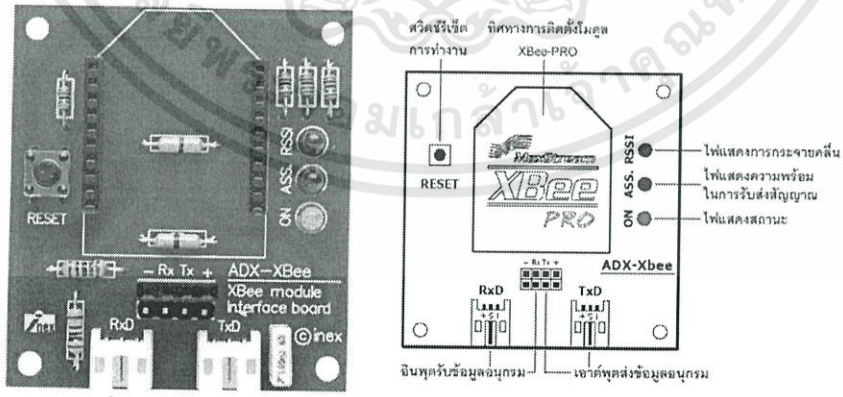
เป็นอุปกรณ์ที่ทำหน้าที่เป็นอุปกรณ์รับ-ส่งสัญญาณ (transceiver) แบบ Half Duplex ย่านความถี่ 2.4 GHz มีการจัดการโดยใช้พลังงานต่ำ ใช้ในการสื่อสารและรับส่งข้อมูลระหว่าง ส่วนการแสดงผลและติดต่อกับผู้ใช้ กับ ส่วนการวัดพลังงานไฟฟ้าและส่วนควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้า



รูปที่ 3.7 แสดงลักษณ์ของเอ็กซ์บี และการจัดขาของเอ็กซ์บี

3.2.4.2 ADX-XBee3V บอร์ดใช้งานโมดูล XBee กับอุปกรณ์

เป็นบอร์ดสำหรับติดตั้งโมดูล XBee-PRO เพื่อเชื่อมต่อกับอุปกรณ์ภายนอก คือ ไมโครคอนโทรลเลอร์ เพื่อช่วยให้ไมโครคอนโทรลเลอร์มีระบบสื่อสารข้อมูลแบบอนุกรมไร้สาย 2.4GHz

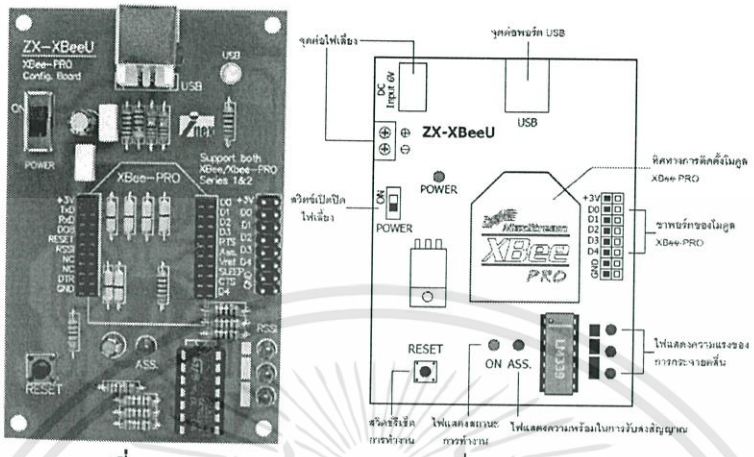


รูปที่ 3.8 แสดง ADX-XBee3V บอร์ดใช้งานโมดูล XBee กับอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4.3 ZX-XBeeU บอร์ดเชื่อมต่อคอมพิวเตอร์

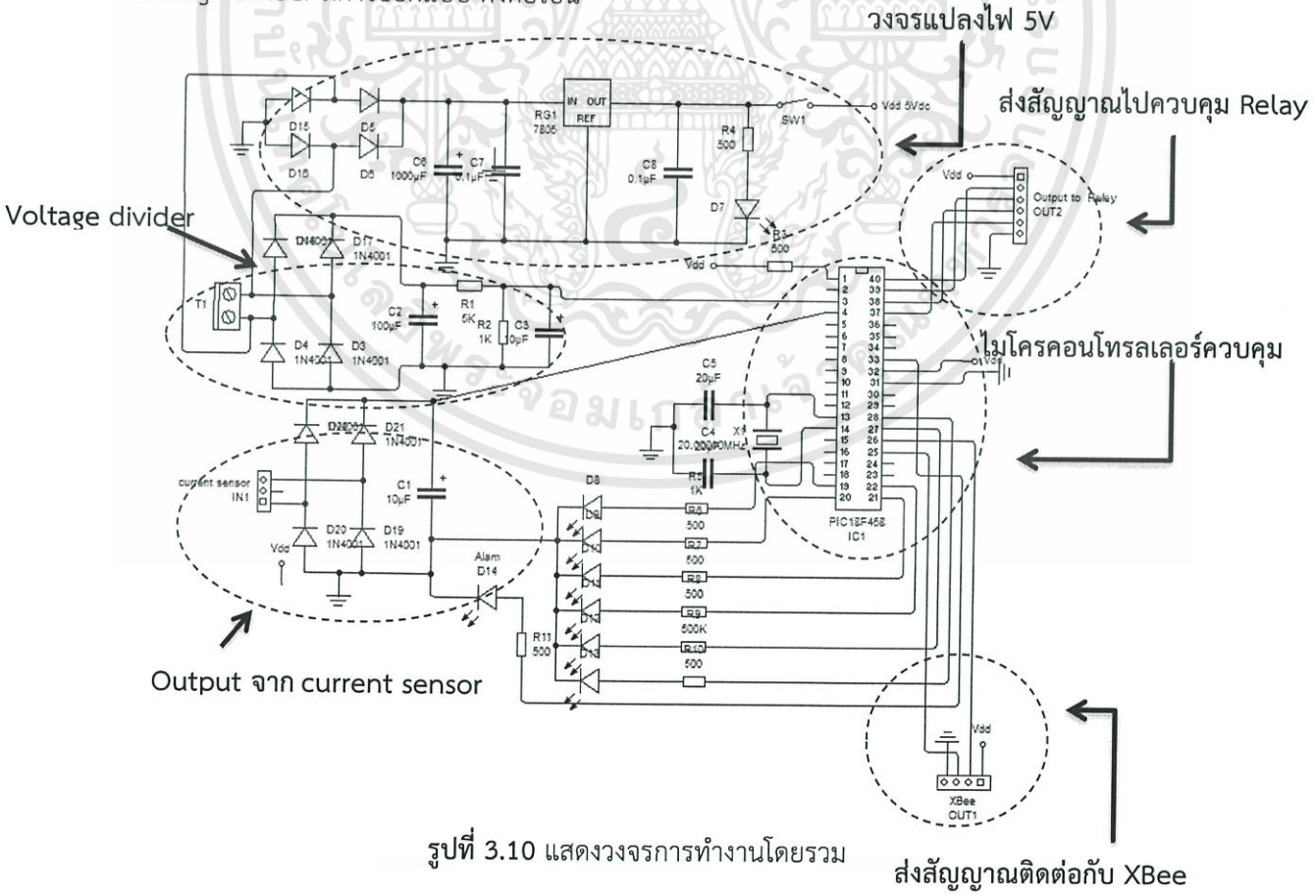
เป็นบอร์ดสำหรับติดตั้งโมดูล XBee-PRO เพื่อเชื่อมต่อกับคอมพิวเตอร์ สำหรับตั้งค่าพารามิเตอร์ ต่างๆ



รูปที่ 3.9 แสดง ZX-XBeeU บอร์ดเชื่อมต่อคอมพิวเตอร์

3.3 การออกแบบวงจร

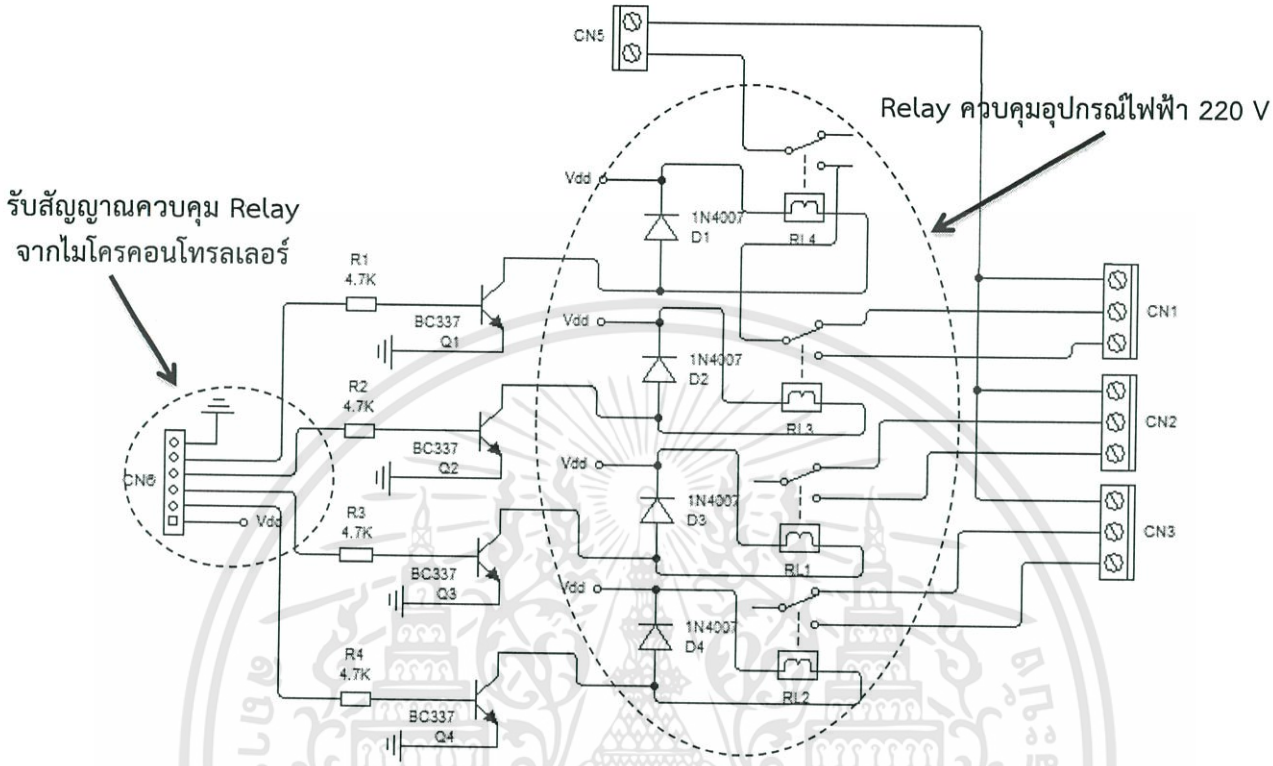
วงจรตรวจวัดพลังงานทางไฟฟ้าและควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้าต่างๆ ภายในบ้าน ซึ่งประกอบด้วย วงจรแปลงไฟ 5V ไมโครคอนโทรลเลอร์ควบคุม current sensor และ voltage divider มีการออกแบบ ดังต่อไปนี้



รูปที่ 3.10 แสดงวงจรการทำงานโดยรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรการทำงานของ Relay มีการออกแบบ ดังต่อไปนี้



รูป 3.11 แสดงวงจรการทำงานของ Relay

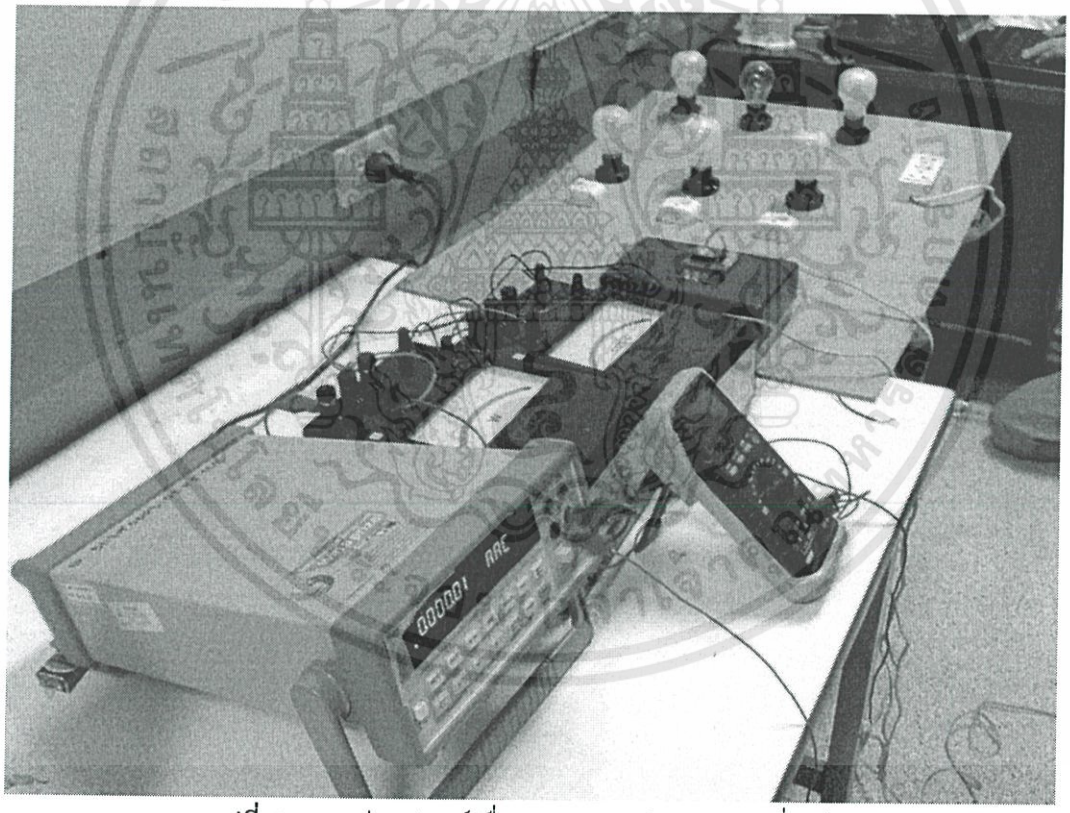
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 การสอบเทียบ

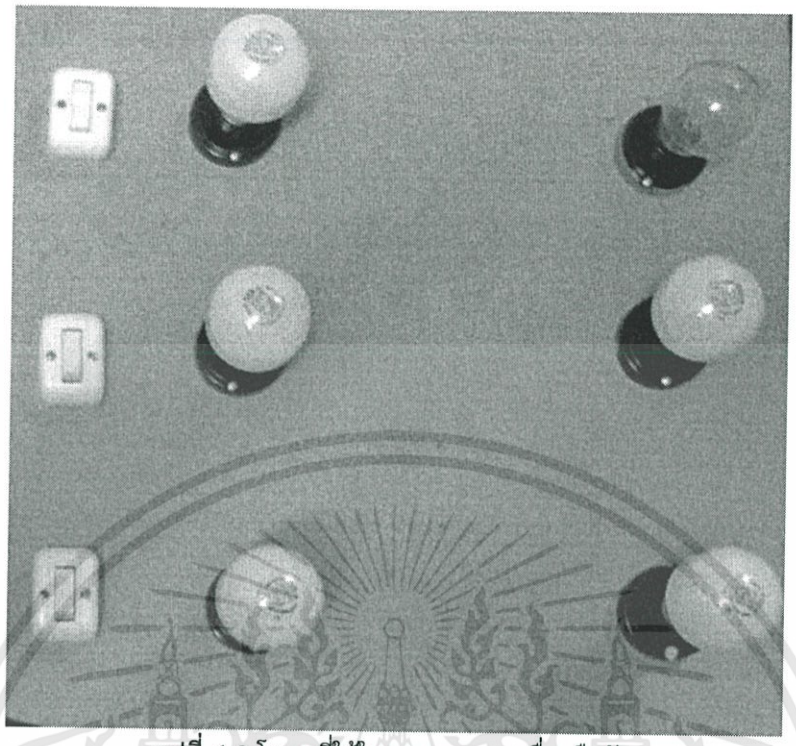
ขั้นตอนของการทดลองเพื่อทดสอบประสิทธิภาพในการทำงานของอุปกรณ์ตรวจวัดพลังงานทางไฟฟ้าภายในบ้านนั้น ได้ทำการทดลองกับโหลดที่เป็นอุปกรณ์เครื่องใช้ไฟฟ้าภายในบ้านจริง คือ หลอดไฟ โดยหลอดไฟที่นำมาใช้ในการทดสอบนั้นมีจำนวน 6 หลอด ซึ่งแต่ละหลอดใช้พลังงานไฟฟ้า 100 WATT และใช้เครื่องมือวัดแบบอะนาล็อกและเครื่องมือวัดแบบดิจิตอลที่มีความเที่ยงตรงสูงมาใช้เป็นตัวอ้างอิงค่าที่ถูกต้อง โดยการอ่านค่าจากอุปกรณ์ตรวจวัดจะทำการอ่านค่าผ่าน SERIAL PORT ของไมโครคอนโทรลเลอร์มาแสดงผลผ่านโปรแกรม VISUAL STUDIO บนหน้าจอคอมพิวเตอร์

ก่อนที่จะสอบเทียบค่านั้นผู้จัดทำโครงงานได้ไปยืมอุปกรณ์วัดค่าทางไฟฟ้าจากภาควิชาวิศวกรรมการวัดคุม คือ PORTABLE SINGLE PHASE WATTMETER, PORTABLE POWER FACTOR METER, THE MY-68 DIGITAL MULTIMETER และ DIGIT MULTIMETER



รูปที่ 4.1 การต่ออุปกรณ์เพื่อทดสอบการทำงานของเครื่องมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 โหลดที่ใช้ในการทดสอบเครื่องมือวัด

รูปที่ 4.3 การอ่านข้อมูลมาแสดงผลผ่าน SERIAL PORT บนหน้าจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้จัดทำโครงงานได้ทำการทดลองวัดค่าพลังงานทางไฟฟ้า เพื่อต้องการที่จะดูความแม่น ดังนี้

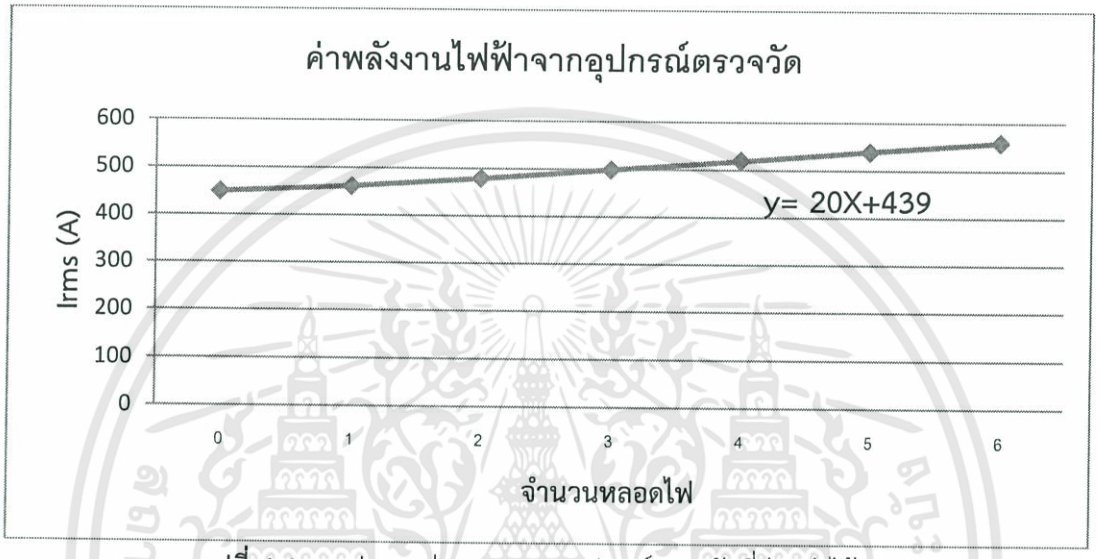
ตารางที่ 4.1 ตารางแสดงค่าเปรียบเทียบค่าพลังงานทางไฟฟ้าระหว่างค่าจริงกับค่าที่วัดได้

จำนวนหลอดไฟ	พลังงานไฟฟ้า		ค่าพลังงานไฟฟ้าจากเครื่องสอบเทียบ	ค่าพลังงานไฟฟ้าจากอุปกรณ์ตรวจวัด
0	IRMS	(A)	0.019	449
	VRMS	(V)	228.0	411
	APPARENT POWER	(VA)	0	184539
	REAL POWER	(W)	0	184539
	PF		1	1
1	IRMS	(A)	0.448	461
	VRMS	(V)	227.7	410
	APPARENT POWER	(VA)	100	189010
	REAL POWER	(W)	100	189010
	PF		1	1
2	IRMS	(A)	0.875	479
	VRMS	(V)	227.3	409
	APPARENT POWER	(VA)	200	195911
	REAL POWER	(W)	200	195911
	PF		1	1
3	IRMS	(A)	1.311	499
	VRMS	(V)	227.0	408
	APPARENT POWER	(VA)	300	203592
	REAL POWER	(W)	300	203592
	PF		1	1
4	IRMS	(A)	1.739	519
	VRMS	(V)	226.6	407
	APPARENT POWER	(VA)	400	211233
	REAL POWER	(W)	400	211233
	PF		1	1
5	IRMS	(A)	2.173	539
	VRMS	(V)	226.1	406
	APPARENT POWER	(VA)	500	240758
	REAL POWER	(W)	500	240758
	PF		1	1
6	IRMS	(A)	2.595	559
	VRMS	(V)	225.8	405

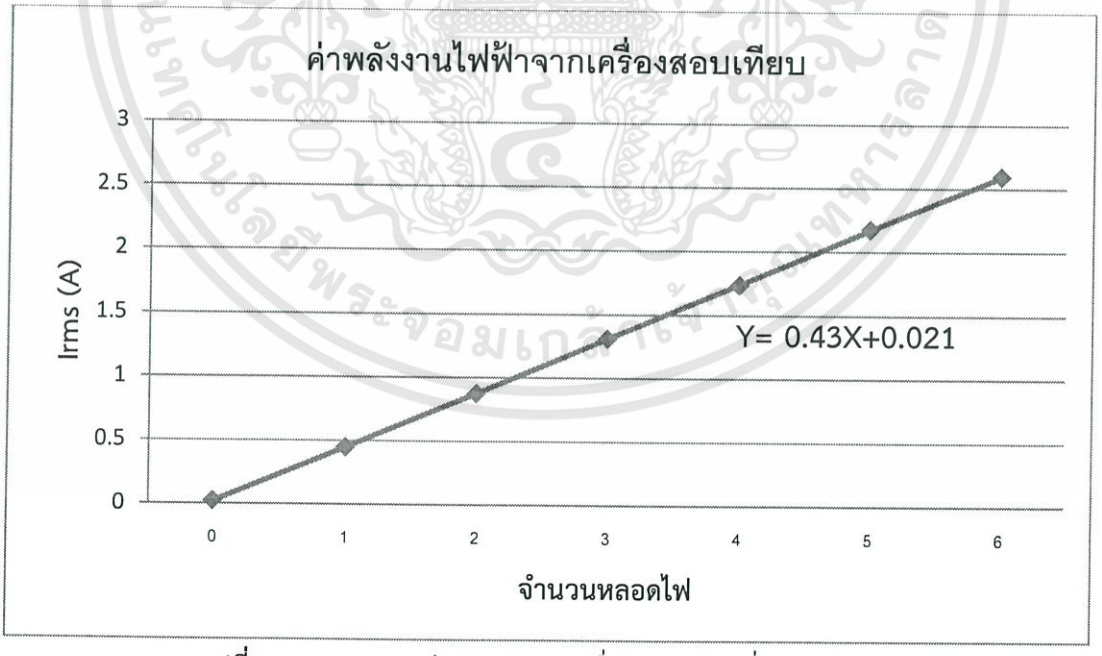
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	APPARENT POWER	(VA)	600	226395
	REAL POWER	(W)	600	226395
	PF		1	1

จากผลการทดลอง เมื่อผู้จัดทำโครงงานมาวิเคราะห์ จากนั้นก็นำค่าที่ได้มาเปรียบเทียบกับค่าที่วัดจริง ดังรูปกราฟต่อไปนี้

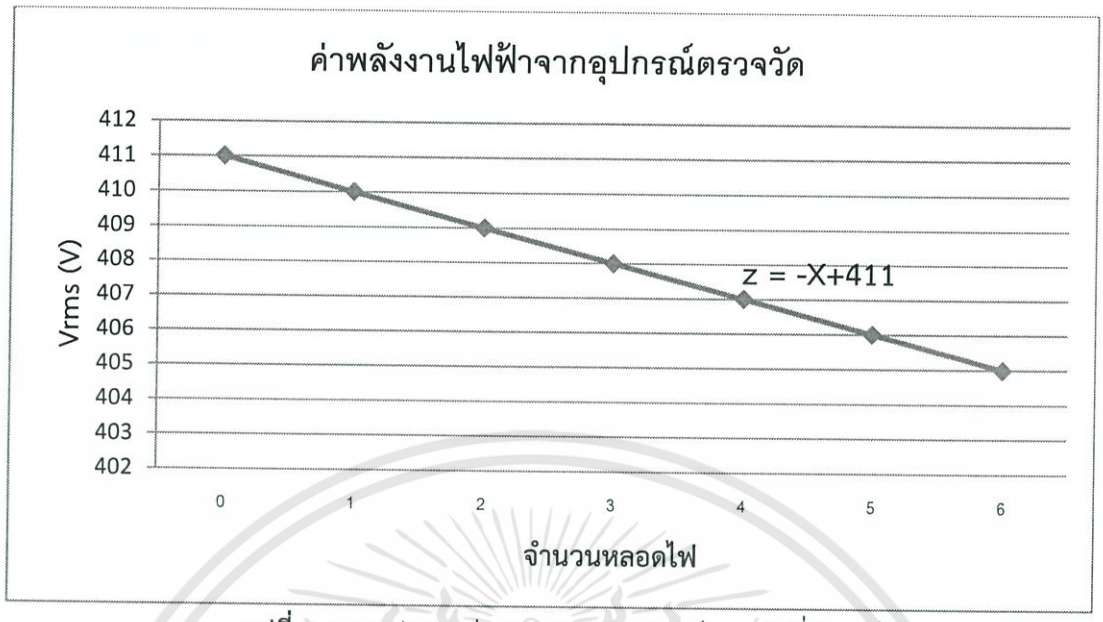


รูปที่ 4.4 กราฟแสดงค่า IRMS จากอุปกรณ์ตรวจวัดที่อ่านค่าได้

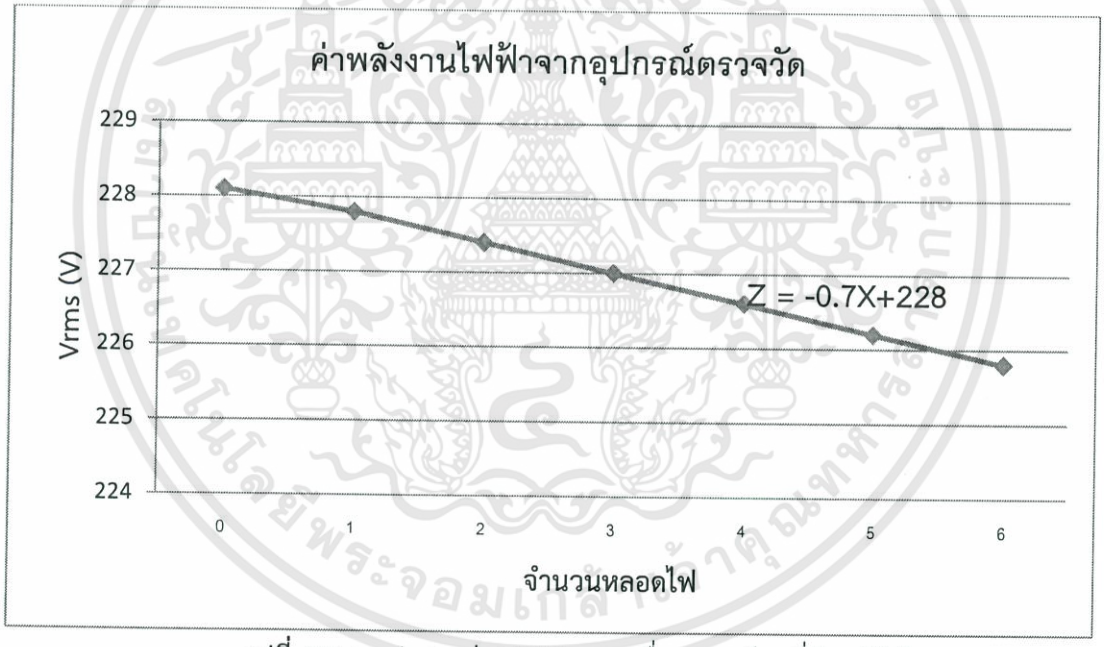


รูปที่ 4.5 กราฟแสดงค่า IRMS จากเครื่องสอบเทียบที่อ่านค่าได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 กราฟแสดงค่า VRMS จากอุปกรณ์ตรวจวัดที่อ่านค่าได้



รูปที่ 4.7 กราฟแสดงค่า VRMS จากเครื่องสอบเทียบที่อ่านค่าได้

จากข้อมูลที่ทดลองได้ เมื่อนำมาพล็อตกราฟ ทำให้ผู้จัดทำโครงงานสามารถทำการสอบเทียบได้ และสามารถหาสมการแก้ไขค่าผิดพลาด ซึ่งจะออกมาเป็นค่าที่สามารถทำให้อุปกรณ์ตรวจวัดพลังงานทางไฟฟ้านี้มีความแม่นยำได้ ผู้จัดทำโครงงานจึงขอยกตัวอย่าง ในการสอบเทียบค่ากระแส (IRMS) และค่าแรงดันไฟฟ้า (VRMS) ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1 สมการแก้ไขค่าผิดพลาด IRMS

ค่า IRMS ที่อ่านได้จากอุปกรณ์ตรวจวัดแล้วนำมาวาดกราฟ สามารถหาสมการเส้นตรงของเส้นกราฟได้ คือ

$$y = 20x + 439 \quad \text{... (1)}$$

ค่า IRMS ที่อ่านได้จาเครื่องสอบเทียบแล้วนำมาวาดกราฟ สามารถหาเส้นตรงของเส้นกราฟได้ คือ

$$Y = 0.43x + 0.021 \text{ ... (2)}$$

จากนั้น ค่า X ก็คือ จำนวนหลอดไฟ ซึ่งทั้งสองสมการจะต้องมีค่ากระแส (IRMS) เท่ากัน ณ หลอดไฟที่มีจำนวนเท่ากัน ดังนั้นจะสามารถเขียนสมการสอบเทียบใหม่ได้ คือ

$$Y = 0.43\left(\frac{y-439}{20}\right) + 0.021 \text{ (3)}$$

ดังนั้น สมการที่ ... (3) เป็นสมการที่ใช้เปลี่ยนค่าที่อ่านได้จากอุปกรณ์ตรวจวัดให้เป็นค่ากระแสไฟฟ้า (IRMS) จริง

4.1.2 สมการแก้ไขค่าผิดพลาด VRMS

ค่า VRMS ที่อ่านได้จากอุปกรณ์ตรวจวัดแล้วนำมาวาดกราฟ สามารถหาสมการเส้นตรงของเส้นกราฟได้ คือ

$$\mathbf{z = -x + 411} \text{ ... (4)}$$

ค่า VRMS ที่อ่านได้จาเครื่องสอบเทียบแล้วนำมาวาดกราฟ สามารถหาเส้นตรงของเส้นกราฟได้ คือ

$$\mathbf{Z = -0.7x + 228} \text{ ... (5)}$$

จากนั้น ค่า X ก็คือ จำนวนหลอดไฟ ซึ่งทั้งสองสมการจะต้องมีค่าแรงดัน (VRMS) เท่ากัน ณ หลอดไฟที่มีจำนวนเท่ากัน ดังนั้นจะสามารถเขียนสมการสอบเทียบใหม่ได้ คือ

$$\mathbf{Z = -0.7(411 - z) + 228} \text{ ... (6)}$$

ดังนั้น สมการที่ ... (6) เป็นสมการที่ใช้เปลี่ยนค่าที่อ่านได้จากอุปกรณ์ตรวจวัดให้เป็นค่าแรงดันไฟฟ้า (VRMS) จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลจากการสอบเทียบค่าและปรับแต่งค่าด้วยสถิติ

ตารางที่ 4.2 ตารางแสดงค่า หลังจากการสอบเทียบค่าด้วยสถิติ ทำให้ได้ค่ามีถูกต้องมากยิ่งขึ้น

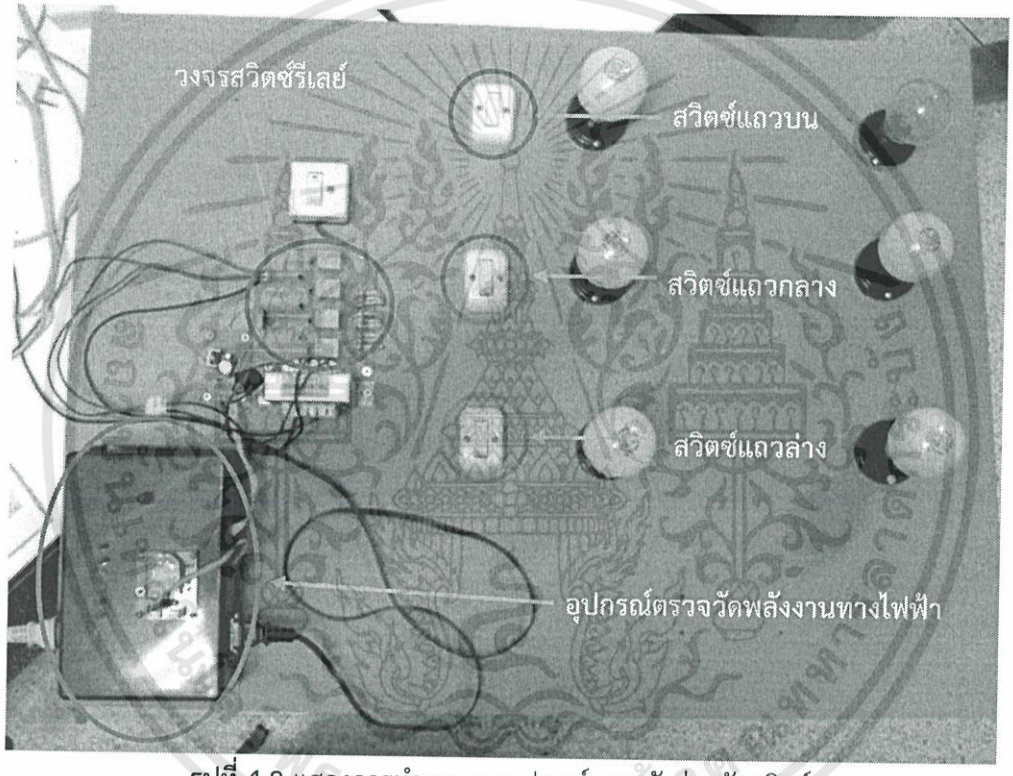
จำนวนหลอดไฟ	พลังงานไฟฟ้า		ค่าพลังงานไฟฟ้าจากเครื่องสอบเทียบ	ค่าพลังงานไฟฟ้าจากอุปกรณ์ตรวจวัด
0	IRMS	(A)	0.019	0.214
	VRMS	(V)	228.00	229.00
	APPARENT POWER	(VA)	0	0
	REAL POWER	(W)	0	0
	PF		1	1
1	IRMS	(A)	0.448	0.472
	VRMS	(V)	227.70	228.50
	APPARENT POWER	(VA)	100	107.85
	REAL POWER	(W)	100	107.85
	PF		1	1
2	IRMS	(A)	0.875	0.881
	VRMS	(V)	227.30	228.00
	APPARENT POWER	(VA)	200	200.87
	REAL POWER	(W)	200	200.87
	PF		1	1
3	IRMS	(A)	1.311	1.310
	VRMS	(V)	227.00	227.50
	APPARENT POWER	(VA)	300	298.03
	REAL POWER	(W)	300	298.03
	PF		1	1
4	IRMS	(A)	1.739	1.741
	VRMS	(V)	226.60	227.00
	APPARENT POWER	(VA)	400	395.21
	REAL POWER	(W)	400	395.21
	PF		1	1
5	IRMS	(A)	2.173	2.185
	VRMS	(V)	226.10	226.50
	APPARENT POWER	(VA)	500	494.90
	REAL POWER	(W)	500	494.90
	PF		1	1
6	IRMS	(A)	2.595	2.611
	VRMS	(V)	225.80	226.00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	APPARENT POWER (VA)	600	590.09
	REAL POWER (W)	600	590.09
	PF	1	1

หมายเหตุ เนื่องจากโหลดที่เราใช้ในการทดลองเป็นโหลดทางด้านความต้านทานจึงทำให้ค่า POWER FACTOR มีเค่าเท่ากับ 1

4.3 การควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้าภายในบ้านผ่าน XBee



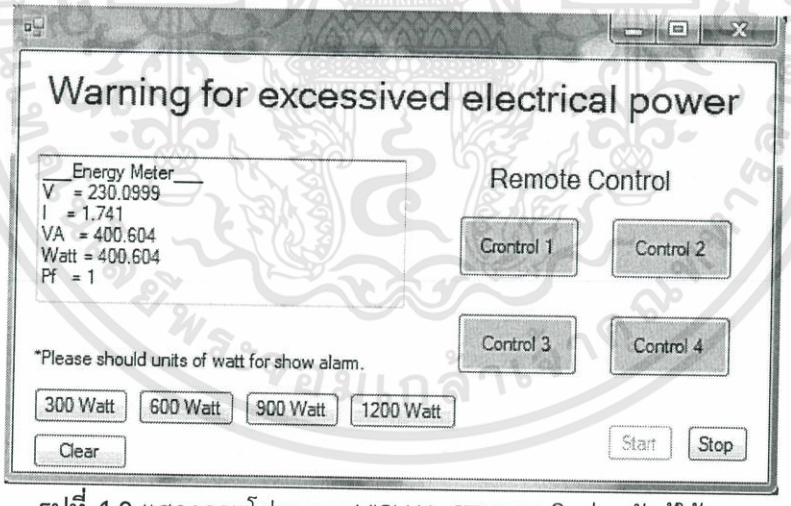
รูปที่ 4.8 แสดงการทำงานของอุปกรณ์ตรวจวัดร่วมกับสวิตช์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ตารางแสดงการควบคุมการทำงานของอุปกรณ์เครื่องใช้ไฟฟ้า

คำสั่งกด	เงื่อนไขการทำงาน
Control 1	- ควบคุมการทำงานทั้งหมดของวงจร
Control 2	- ควมคุมการทำงานของหลอดไฟแถวบนสุดร่วมกับสวิตซ์
Control 3	- ควมคุมการทำงานของหลอดไฟแถวกลางร่วมกับสวิตซ์
Control 4	- ควมคุมการทำงานของหลอดไฟแถวล่างสุดร่วมกับสวิตซ์
300 Watt	- เมื่อกำลังไฟฟ้าเกิน 300 Watt จะทำให้เกิดไฟแจ้งเตือนขึ้น
600 Watt	- เมื่อกำลังไฟฟ้าเกิน 600 Watt จะทำให้เกิดไฟแจ้งเตือนขึ้น
900 Watt	- เมื่อกำลังไฟฟ้าเกิน 900 Watt จะทำให้เกิดไฟแจ้งเตือนขึ้น
1200 Watt	- เมื่อกำลังไฟฟ้าเกิน 1200 Watt จะทำให้เกิดไฟแจ้งเตือนขึ้น

4.4 การแสดงผลผ่านโปรแกรม VISUAL STUDIO บนหน้าจอคอมพิวเตอร์



รูปที่ 4.9 แสดงภาพโปรแกรม VISUAL STUDIO ติดต่อกลับผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการสร้างระบบแจ้งเตือนเมื่อใช้พลังงานเกิน ซึ่งประมวลผลการตรวจวัดพลังงานไฟฟ้าด้วยไมโครคอนโทรลเลอร์โดยส่งผ่าน Zigbee มาแสดงผลค่าพลังงานไฟฟ้าต่างๆ ผ่านคอมพิวเตอร์นั้น ผลการดำเนินงานพบว่าค่าแรงดันไฟฟ้าที่คำนวณจากไมโครคอนโทรลเลอร์นั้น ยังไม่มีความแม่นยำมากนักแต่ก็สามารถนำมาใช้งานได้อยู่ ส่วนค่าพลังงานไฟฟ้าอื่นๆ ถือได้ว่านำมาใช้งานได้ คือ สามารถกำหนดค่าพลังงานไฟฟ้ามากสุดที่ใช้เพื่อการแจ้งเตือนด้วยหลอด LED ได้ และสามารถควบคุมการเปิดปิดหลอดไฟผ่านคอมพิวเตอร์ได้

5.2 ปัญหาที่พบในการทำวิจัย

จากการทำวิจัยพบว่าค่าของแรงดันไฟฟ้าที่ได้จากการคำนวณยังไม่มีความแม่นยำพอ อันเนื่องมาจากในการทดลองนั้นเมื่อเพิ่มโหลดการใช้งานด้วยการเปิดหลอดไฟ พบว่าค่าของแรงดันไฟฟ้าเปลี่ยนแปลงน้อยมาก และการตรวจจับแรงดันไฟฟ้าที่ส่งมายังไมโครคอนโทรลเลอร์เกิดการแกว่งของสัญญาณยากที่ไมโครคอนโทรลเลอร์จะตรวจจับได้อย่างแม่นยำ แนวทางการแก้ปัญหาคือ ศึกษาเพิ่มเติมสำหรับการใช้อุปกรณ์ตรวจวัดและประมวลผลแรงดันไฟฟ้าที่มีประสิทธิภาพมากกว่าเดิม

5.2 ข้อเสนอแนะ

1. สามารถพัฒนาระบบให้ควบคุมการเปิด-ปิดไฟฟ้าหลายๆ จุด ด้วยเครื่องคอมพิวเตอร์พร้อมกันได้ เนื่องจากชิกบีรองรับรูปแบบโครงข่ายแบบ Point–to–multiple
2. สามารถพัฒนาชิ้นงานให้มีขนาดเล็กลง หรือ นำไปสร้างเป็นส่วนประกอบของอุปกรณ์ไฟฟ้าโดยตรงได้ เพื่อเพิ่มความสะดวกสบายในการติดตั้งและใช้งาน รวมทั้งสามารถควบคุมการเปิด-ปิดไฟฟ้าจากระยะไกลได้
3. สามารถพัฒนาระบบให้ควบคุมการเปิด-ปิดไฟฟ้าที่มีแรงดันสูงเกินกว่า 220 V AC ได้
4. สามารถพัฒนาและประยุกต์โครงงานนี้ร่วมกับเซนเซอร์ชนิดต่างๆ นอกเหนือจากเซนเซอร์ตรวจจับกระแสที่ผู้จัดทำได้ใช้ในโครงงานนี้ เช่น การประยุกต์ใช้ Infrared Sensor ตรวจจับการเดินผ่านของมนุษย์ เพื่อควบคุมการเปิด-ปิดไฟ เป็นต้น
5. สามารถนำโครงงานนี้ไปต่อยอด โดยการเชื่อมโยงกับระบบโครงข่ายอินเตอร์เน็ต เพื่อให้เกิดความสะดวกสบายแก่ผู้ใช้งานมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ทีมงานสมาร์ทเลิร์นนิ่ง. **Advance PIC Microcontroller in C การประยุกต์ใช้งาน PIC ขั้นสูงด้วยภาษา C.** พิมพ์ครั้งที่ 1. กรุงเทพฯ: ห้างหุ้นส่วนสามัญสมาร์ทเลิร์นนิ่ง, 2555.

บริษัทวีนัส ซัพพลาย. **Introduction Zigbee and Xbee BASIC.** [Online]. Available : http://www.thaieasyelec.com/Review-Product-Article/zigbee-and-xbee-basic.html

บริษัทวีนัส ซัพพลาย. **ACS714 Current Sensor Carrier -5 to +5A.** [Online]. Available : http://www.thaieasyelec.com/Sensors/Voltage-Current-Sensor/ACS714-Current-Sensor-Carrier--5-to-5A.html

IT-Devzone. **Demo RS232 using C# – Serial Communication.** [Online]. Available : http://www.it-devzone.com/demo-rs232/

Instructables. **Using Visual Basic 2010 to control Arduino Uno.** [Online]. Available : http://www.instructables.com/id/Using-Visual-Basic-to-control-Arduino-Uno/

สำนักงานคณะกรรมการการอาชีวศึกษา. **ใบความรู้ วิชาอิเล็กทรอนิกส์อุตสาหกรรม 1 เรื่อง รีเลย์.**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม mikroC for PIC สำหรับ PIC18F458

```
int alam(float max,float com);
void main()
{
unsigned char i;
int a;
double vol=0,cur=0,an[3];
double max=9000,com,va;
char pvol[10],pcur[10],pva[15];
trisa.f1=1;
trisa.f2=1;
trisb=0x00;
UART1_Init(9600);
while(1)
{
    delay_ms(1000);
    an[1]=adc_read(1);vol=0.7*an[1]-59.7;
    an[2]=adc_read(2);cur=0.0215*an[2]-9.4175;
    va=vol*cur;
    com=va;
    floattostr(vol,pvol);
    floattostr(cur,pcur);
    floattostr(va,pva);
    uart1_write_text("___Energy Meter___\n");delay_ms(50);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      uart1_write_text("V    = "); uart1_write_text(pvol);uart1_write_text("\n");
  delay_ms(50);

      uart1_write_text("I    = "); uart1_write_text(pcur);uart1_write_text("\n");
  delay_ms(50);

      uart1_write_text("VA   = "); uart1_write_text(pva);uart1_write_text("\n");
  delay_ms(50);

      uart1_write_text("Watt = "); uart1_write_text(pva);uart1_write_text("\n");
  delay_ms(50);

      uart1_write_text("Pf   = 1\n");

      a=alam(max,com);

      portb.f0=a;

       if(Uart1_data_ready())

         { i = Uart1_Read();

          if(i=='1'){portb.f7=~portb.f7;}

          if(i=='2'){portb.f6=~portb.f6;}

          if(i=='3'){portb.f5=~portb.f5;}

          if(i=='4'){portb.f4=~portb.f4;}

          if(i=='A'){max=300;}

          if(i=='B'){max=600;}

          if(i=='C'){max=900;}

          if(i=='D'){max=1200;}

          if(i=='F'){max=9000;}

        }

}

}

int alam(float max,float com)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
int a;
if(com>max){a=1;return a;}
 else a=0;return a;
}
```

โปรแกรม Visual Studio สำหรับแสดงผลและควบคุมสวิตซ์รีเลย์ทางคอมพิวเตอร์

```
using System;
using System.Windows.Forms;

namespace RS232
{
    public partial class Form2 : Form
    {
        private System.IO.Ports.SerialPort serialPort1;
        //private IContainer components;
        private TextBox textBox1;
        private Button buttonStart;
        private Button buttonStop;

        // Add this variable
        string RxString;


        public Form2()
        {

            InitializeComponent();
        }

        private void buttonStart_Click(object sender, EventArgs e)
        {
            serialPort1.PortName = "COM6";
            serialPort1.BaudRate = 9600;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        serialPort1.DataBits = 8;
        serialPort1.StopBits = System.IO.Ports.StopBits.One;
        serialPort1.Parity = System.IO.Ports.Parity.None;
        serialPort1.Handshake = System.IO.Ports.Handshake.RequestToSend;
        serialPort1.Open();


        if (serialPort1.IsOpen)
        {
            buttonStart.Enabled = false;
            buttonStop.Enabled = true;
            textBox1.ReadOnly = false;
        }
    }

    private void buttonStop_Click(object sender, EventArgs e)
    {
        if (serialPort1.IsOpen)
        {
            serialPort1.Close();
            buttonStart.Enabled = true;
            buttonStop.Enabled = false;
            textBox1.ReadOnly = true;
        }

    }

    private void Form1_FormClosing(object sender, FormClosingEventArgs e)
    {
        if (serialPort1.IsOpen) serialPort1.Close();
    }

    private void textBox1_KeyPress(object sender, KeyPressEventArgs e)
    {
        // If the port is closed, don't try to send a character.
        if (!serialPort1.IsOpen) return;

        // If the port is Open, declare a char[] array with one element.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        char[] buff = new char[1];

        // Load element 0 with the key character.
        buff[0] = e.KeyChar;

        // Send the one character buffer.
        serialPort1.Write(buff, 0, 1);

        // Set the KeyPress event as handled so the character won't
        // display locally. If you want it to display, omit the next line.
        e.Handled = false;
    }

    private void DisplayText(object sender, EventArgs e)
    {
        textBox1.AppendText(RxString);
    }

    private void serialPort1_DataReceived(object sender,
System.IO.Ports.SerialDataReceivedEventArgs e)
    {
        RxString = serialPort1.ReadExisting();
        this.Invoke(new EventHandler(DisplayText));
    }

    private void button1_Click(object sender, EventArgs e)
    {
        serialPort1.Write("1");
    }

    private void button2_Click(object sender, EventArgs e)
    {
        serialPort1.Write("2");
    }

    private void button3_Click(object sender, EventArgs e)
    {
        serialPort1.Write("3");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }

    private void button4_Click(object sender, EventArgs e)
    {
        serialPort1.Write("4");
    }

    private void button5_Click(object sender, EventArgs e)
    {
        button5.Enabled = false;
        button6.Enabled = true;
        button7.Enabled = true;
        button8.Enabled = true;
        serialPort1.Write("A");
    }

    private void button6_Click(object sender, EventArgs e)
    {
        button5.Enabled = true;
        button6.Enabled = false;
        button7.Enabled = true;
        button8.Enabled = true;
        serialPort1.Write("B");
    }

    private void button7_Click(object sender, EventArgs e)
    {
        button5.Enabled = true;
        button6.Enabled = true;
        button7.Enabled = false;
        button8.Enabled = true;
        serialPort1.Write("C");
    }

    private void button8_Click(object sender, EventArgs e)
    {
        button5.Enabled = true;
        button6.Enabled = true;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            button7.Enabled = true;
            button8.Enabled = false;
            serialPort1.Write("D");
        }

        private void button9_Click(object sender, EventArgs e)
        {
            button5.Enabled = true;
            button6.Enabled = true;
            button7.Enabled = true;
            button8.Enabled = true;
            serialPort1.Write("F");
        }

        private void label4_Click(object sender, EventArgs e)
        {

        }

    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้