สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การเก็บข้อมูลแบบไร้สายด้วยเทคโนโลยี Wi-Fi

(Wi-Fi Technology Data Logging)

T119382

นารีรัตน์ สาลิกา

พิชชาภา พานเงิน

วันวิสา ช้างแดง

เลขหมู่.....
เลขทะเบียน..... 119382
วัน,เดือน,ปี..... - 7 ส.ค. 2554

b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Wi-Fi Technology Data Logging

Nareerat Salika

Pitchapa Panngern

Wanwisa Changdaeng

THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT

OF REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN CONTROL ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2010

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การเก็บข้อมูลแบบไร้สายด้วยเทคโนโลยี Wi-Fi

Wi-Fi Technology Data Logging

ผู้จัดทำ นางสาวนารีรัตน์ สาลิกา 50010810

นางสาวพิชชาภา พานเงิน 50011089

นางสาววันวิสา ช้างแดง 50011445

สังวาล บกสุวรรณ .....อาจารย์ที่ปรึกษา

(อาจารย์สังวาล บกสุวรรณ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเก็บข้อมูลแบบไร้สายด้วยเทคโนโลยี Wi-Fi

โดย

นางสาวนารีรัตน์	สาลิกา	50010810
นางสาวพิชชาภา	พานเงิน	50011089
นางสาววันวิสา	ช้างแดง	50011445

อาจารย์ที่ปรึกษา

อาจารย์สังวาล บกสุวรรณ

ปีการศึกษา 2553

บทคัดย่อ

ในปริญญานิพนธ์เล่มนี้มีวัตถุประสงค์ เพื่อการส่งข้อมูลผ่านการเชื่อมต่อแบบไร้สายด้วยเทคโนโลยีวายฟาย ซึ่งในครั้งนี้ต้องการศึกษาการรับข้อมูลซึ่งเป็นอุณหภูมิจากเซนเซอร์และใช้ไมโครคอนโทรลเลอร์เชื่อมต่อแบบไร้สายด้วยเทคโนโลยีวายฟายเข้าสู่คอมพิวเตอร์ ซึ่งใช้เซนเซอร์ TC 1047A เป็นตัววัดอุณหภูมิโดยมีวงจรกรองสัญญาณรบกวน ข้อมูลที่ได้จะถูกส่งผ่านแบบวายฟายและแสดงผลบนคอมพิวเตอร์อีกเครื่องหนึ่งด้วยโปรแกรม Wi-Fi data logging และข้อมูลที่ได้สามารถนำไปประยุกต์ใช้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Wi-Fi Technology Data Logging

By

Miss Nareerat Salika

Miss Pitchapa Panngern

Miss Wanwisa Changdaeng

Advisor

Mr. Sungwan Boksuwan

Academic Year 2010

Abstract

The purpose of this thesis is to transmit information over a Wi-Fi communication. The interesting information comes from measuring the temperature using microcontroller. And then the resulting data will be sent to the computer through Wi-Fi technology. In this project, the temperature is measured by TC1047 producing the analog output and filtered by second order Sellan key low-pass filter. The received information from filter will be sent with Wi-Fi and displayed on another computer via Wi-Fi data logging program.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์ฉบับนี้ สามารถสำเร็จลุล่วงไปได้ด้วยดี เพราะได้รับความช่วยเหลือเป็นอย่างดีจากอาจารย์ สังวาล บกสุวรรณ ที่ได้กรุณาให้คำแนะนำที่ดีมาโดยตลอด รวมไปถึงความช่วยเหลืออื่นๆ ที่เป็นประโยชน์ต่อโครงงาน ผู้จัดทำรู้สึกซาบซึ้งและขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบคุณพี่ปริญญาโทและเพื่อนๆ ที่ช่วยเหลือให้คำแนะนำได้เป็นอย่างดี รวมไปถึงเสนอแนะ ข้อผิดพลาดที่เกิดขึ้นในโครงงาน

สุดท้ายนี้ผู้จัดทำขอขอบคุณบิดา มารดา และครอบครัว ที่เป็นกำลังใจที่ดีตลอดมารวมถึงการสนับสนุนในเรื่องของงบประมาณที่ขาดเหลือ ตลอดจนเป็นแรงบันดาลใจที่ดีที่สุดในการทำให้โครงงานนี้สำเร็จลุล่วงไปได้ด้วยดี

และสุดท้าย ขอขอบคุณทุกๆท่านที่ได้มีส่วนเกี่ยวข้องกับหนังสือเล่มนี้

ผู้จัดทำ

นารีรัตน์ สาลิกา

พิชชาภา พานเงิน

วันวิสา ช้างแดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	**I**
บทคัดย่อภาษาอังกฤษ	**II**
กิตติกรรมประกาศ	**III**
สารบัญ	**IV**
สารบัญภาพ	**VI**
บทที่ 1 บทนำ	
1.1 ความสำคัญและความเป็นมา	1
1.2 เป้าหมายและวัตถุประสงค์ของโครงงาน	2
1.3 ขั้นตอนการศึกษาและการจัดทำโครงงาน	2
บทที่ 2 ทฤษฎีและความรู้พื้นฐาน	
2.1 ความรู้เกี่ยวกับ Wi-Fi	3
2.1.1 Peer - to – Peer	3
2.1.2 Client/server (Infrastructure mode)	4
2.2 อุปกรณ์ในส่วนของการเชื่อมต่อ	5
2.2.1 ไมโครคอนโทรลเลอร์ dsPIC30F6014 และ PIC24FJ128GB110	5
2.2.2 โมดูล Wi-Fi เบอร์ MRF24WB0MA	7
2.2.3 Router เบอร์ TEW-635BRM	8
2.2.4 บอร์ดที่ใช้ในการเชื่อมต่อ	8
2.3 รูปแบบของการติดต่อสื่อสารกันระหว่างไมโครคอนโทรลเลอร์ กับโมดูลWi-Fi	9
2.3.1 การเชื่อมต่อแบบ SPI	9
2.3.2 การส่งข้อมูลแบบซิงโครนัส	10
2.4 อุปกรณ์การวัดอุณหภูมิ	11
2.4.1 เซนเซอร์วัดอุณหภูมิ เบอร์ TC1047A	11
2.4.2 ออปแอมป์ เบอร์ MCP602	12
2.5 รูปแบบการติดต่อสื่อสารกันระหว่างไมโครคอนโทรลเลอร์กับเซนเซอร์วัดอุณหภูมิ	12
2.6 การใช้งานโมดูลไทเมอร์ในไมโครคอนโทรลเลอร์ dsPIC30F6014	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 3 หลักการออกแบบ

3.1การออกแบบการใช้งานไทเมอร์ 17

3.2 การเชื่อมต่อแบบ SPI 18

3.3 การทำงานของ ADC 18

3.4 การออกแบบ Low pass filter 19

3.5 การเชื่อมต่อแบบWi-Fi บนโมดูล TCP 21

บทที่ 4 การทดลองและผลการทดลอง

4.1 การทดลองการใช้งาน Timer 23

4.2 การทดลองการใช้งาน SPI 24

4.3 การทดลองการวัดอุณหภูมิจากเซนเซอร์ 25

4.4 การทดลองการเชื่อมต่อแบบ Wi-Fi 26

บทที่ 5 บทวิจารณ์และสรุป

5.1 สรุปผลการทดลอง 29

5.1.1 สรุปผลการทดลองการใช้งาน Timer 29

5.1.2 สรุปผลการทดลองการใช้งาน SPI 29

5.1.3 สรุปผลการทดลองการวัดอุณหภูมิจากเซนเซอร์ 29

5.1.4 สรุปผลการทดลองการเชื่อมต่อแบบ Wi-Fi 30

5.2 ปัญหาที่พบและแนวทางแก้ไข 30

5.3 ข้อเสนอแนะและแนวทางในการค้นคว้าพัฒนา 30

ภาคผนวก 31

เอกสารอ้างอิง 49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 1.1 ภาพรวมของการติดต่อสื่อสาร 1

รูปที่ 2.1 การเชื่อมต่อระหว่างโมดูลและไมโครคอนโทรลเลอร์ 3

รูปที่ 2.2 การเชื่อมต่อแบบ Peer to Peer 4

รูปที่ 2.3 การเชื่อมต่อแบบ Client 5

รูปที่ 2.4 ไมโครคอนโทรลเลอร์ dsPIC30F6014 6

รูปที่ 2.5 ไมโครคอนโทรลเลอร์ PIC24FJ128GB110 6

รูปที่ 2.6 โมดูล Wi-Fi เบอร์ MRF24WB0MA 7

รูปที่ 2.7 Router เบอร์ TEW-635BRM 8

รูปที่ 2.8 บอร์ดทดลอง dsPICDEM™ 1.1 Development Board 8

รูปที่ 2.9 บอร์ด Wi-Fi 9

รูปที่ 2.10 ไดอะแกรมโมดูล SPI 10

รูปที่ 2.11 การส่งข้อมูลแบบซิงโครนัส 10

รูปที่ 2.12Block Diagram การเชื่อมต่อของเซนเซอร์คุณสมบัติของเซนเซอร์ 11

รูปที่ 2.13 แผนผังวงจรวัดอุณหภูมิ 12

รูปที่ 2.14 ไดอะแกรมไทเมอร์ฐานเวลาแบบ A 16

รูปที่ 3.1 Flow Chart การทำงานของฟังก์ชั่นไทเมอร์ 17

รูปที่ 3.2 Flow Chart การทำงานของฟังก์ชั่น SPI 18

รูปที่ 3.3 แผนผังวงจรกรองสัญญาณความถี่ต่ำ 19

รูปที่ 3.4 กราฟสเปกตรัมของข้อมูล 20

รูปที่ 3.5 กราฟ Bode Diagram แสดง Frequency Response 21

รูปที่ 3.6 ภาพรวมการใช้งานของโมดูล TCP 21

รูปที่ 4.1 การเชื่อมต่อของอุปกรณ์ต่าง ๆ ในการทดลองที่4.1 -4.3 23

รูปที่ 4.2 กราฟที่ได้จากการทดลองการใช้งาน Timer1 24

รูปที่ 4.3 กราฟที่ได้จากการทดลองการใช้งาน SPI 24

รูปที่ 4.4 ผลการทดลองจากโปรแกรม HyperTerminal 25

รูปที่ 4.5 บอร์ด Wi-Fi 26

รูปที่ 4.6 การเชื่อมต่อของอุปกรณ์ต่าง ๆ ในการทดลองที่ 4.4 26

รูปที่ 4.7 โปรแกรม Wi-Fi Data Logging 27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ(ต่อ)

	หน้า
รูปที่ 4.8 กราฟผลการทดลองการเชื่อมต่อแบบ Wi-Fi	27
รูปที่ 4.9 ข้อมูลจากกราฟการทดลองการเชื่อมต่อแบบ Wi-Fi	28
รูปที่ 4.10 กราฟ Log ที่ได้ในโปรแกรม MATLAB	28
รูปที่ 5.1 อุณหภูมิที่วัดได้จากเทอร์โมมิเตอร์	29

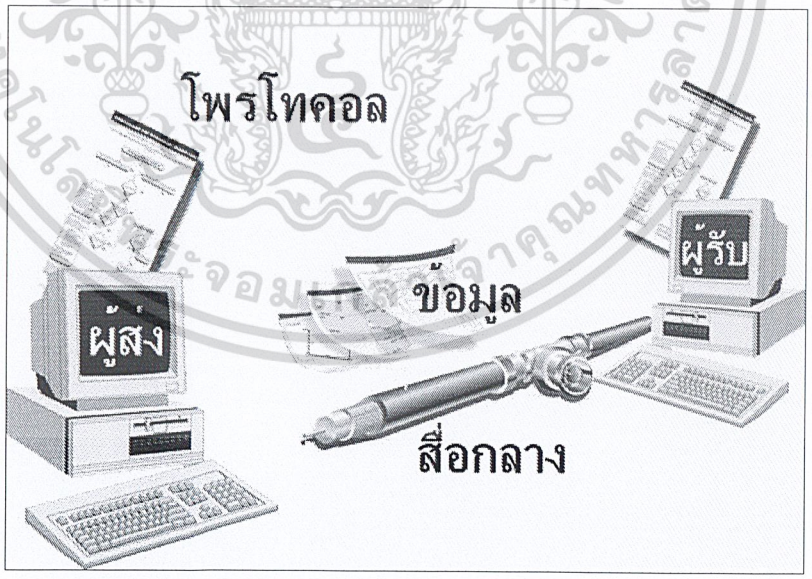
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและความเป็นมา

เนื่องด้วยตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมระบบควบคุม ภาควิชาวิศวกรรมระบบควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังได้เห็นความสำคัญของการให้นักศึกษาได้ทดลองปฏิบัติการจริงเพื่อนำความรู้ที่ได้จากการศึกษามาประยุกต์ใช้ จึงกำหนดให้นักศึกษาชั้นปีที่ 4 จัดตั้งกลุ่มขึ้นเพื่อทำการเสนอโครงงาน ซึ่งกำหนดระยะเวลาทั้งสิ้น 2 ภาคการศึกษา

ซึ่งภาควิศวกรรมระบบควบคุมนั้นมีการศึกษาที่เกี่ยวข้องกับการเขียนโปรแกรมลงไมโครคอนโทรลเลอร์เพื่อทำการควบคุมการทำงานของระบบและอุปกรณ์อิเล็คทรอนิคส์ต่างๆ ดังนั้นจึงมีแนวคิดที่จะนำไมโครคอนโทรลเลอร์มาประยุกต์ใช้ควบคุมการบันทึกข้อมูลแบบไร้สายด้วยเทคโนโลยี Wi-Fi เนื่องจากเป็นเทคโนโลยีที่ใช้กันอย่างแพร่หลายในปัจจุบันและสามารถนำไปใช้งานได้อย่างหลากหลายอีกด้วย



รูปที่ 1.1 ภาพรวมของการติดต่อสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 เป้าหมายและวัตถุประสงค์ของโครงงาน

การเก็บข้อมูลเป็นสิ่งที่สำคัญมากในการทำการทดลองเพราะจะทำให้ข้อมูลนั้นสามารถนำมาวิเคราะห์ได้ในภายหลัง ดังนั้น โครงงานนี้จึงทำการศึกษาการติดต่อสื่อสารแบบไร้สายด้วยโมดูล Wi-Fi โดยทำการอ่านค่าของอุณหภูมิจากเซนเซอร์ และส่งค่าข้อมูลที่ได้ไปยังคอมพิวเตอร์ผ่านการสื่อสารด้วยเทคโนโลยี Wi-Fi เพื่อเก็บข้อมูลดังกล่าวในรูปแบบของไฟล์อิเลคทรอนิคส์

1.3 ขั้นตอนการศึกษาและการจัดทำโครงงาน

1. ศึกษาการเชื่อมต่อข้อมูลแบบไร้สายด้วย Wi-Fi
2. ศึกษาการทำงานและการติดต่อสื่อสารของไมโครคอนโทรลเลอร์กับอุปกรณ์อื่นๆ
3. ออกแบบและทดสอบโปรแกรมของไมโครคอนโทรลเลอร์
4. ศึกษาวงจรและออกแบบโปรแกรมทำงานของวงจรวัดอุณหภูมิ
5. ทดลองการทำงานระหว่างอุปกรณ์และปรับปรุงแก้ไข
6. สรุปและอภิปรายผล

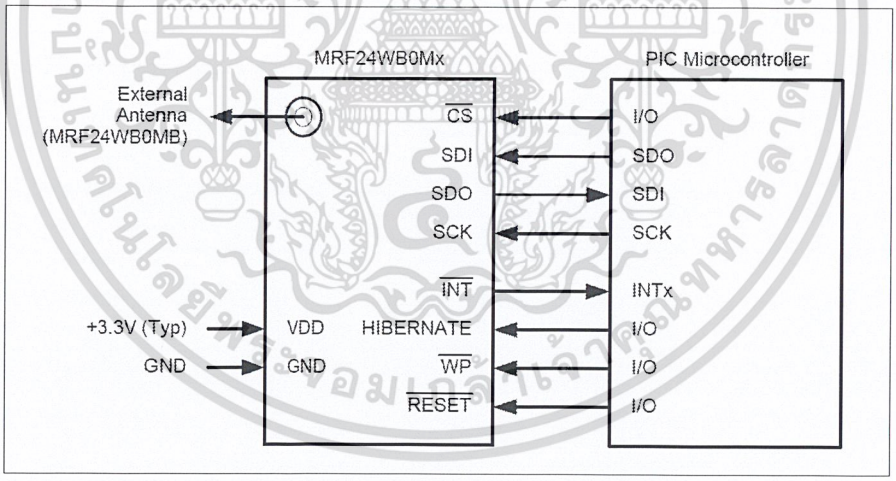
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและความรู้พื้นฐาน

จากที่ได้กล่าวมาในบทที่ 1 ก่อนทำการทดลองและการเชื่อมต่อจำเป็นต้องศึกษาหาข้อมูล ความรู้และองค์ประกอบต่างๆ เพื่อเป็นแนวทางในการศึกษาและคิดคำนวณ ซึ่งทฤษฎีและความรู้ที่จำเป็นนั้นจะกล่าวในบทนี้

2.1 ความรู้เกี่ยวกับ Wi-Fi

วายฟาย (Wi-Fi ย่อมาจาก wireless fidelity) หมายถึงชุดผลิตภัณฑ์ต่างๆ ที่สามารถใช้ได้กับมาตรฐานเครือข่ายคอมพิวเตอร์แบบไร้สาย (มาตรฐาน IEEE 802.11) ในโครงงานนี้จะใช้โมดูลเบอร์ MRF24WB0MA ทำการเชื่อมต่อแบบไร้สายกับไมโครคอนโทรลเลอร์ ผ่านทาง SPI 4 ขาสัญญาณ



รูปที่ 2.1 การเชื่อมต่อระหว่างโมดูลและไมโครคอนโทรลเลอร์

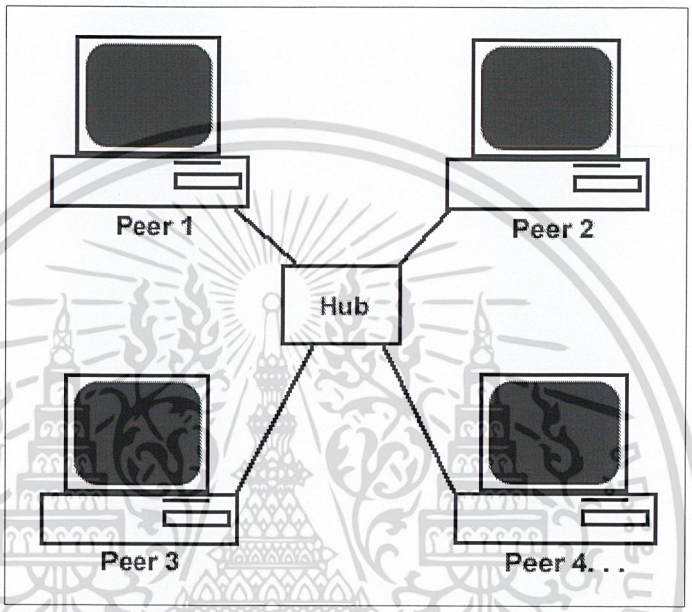
การติดต่อสื่อสารด้วย Wi-Fi ที่นิยมมี 2 แบบคือ

2.1.1 Peer to Peer

รูปแบบการเชื่อมต่อระบบแลนไร้สายแบบ Peer to Peer เป็นลักษณะ การเชื่อมต่อแบบโครงข่ายโดยตรงระหว่างเครื่องคอมพิวเตอร์ จำนวน 2 เครื่องหรือมากกว่านั้น เป็นการใช้งานร่วมกันของ wireless adapter cards โดยไม่ได้มีการเชื่อมต่อกับเครือข่ายแบบใช้สายเลย โดยที่เครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์แต่ละเครื่องจะมีความเท่าเทียมกัน สามารถทำงานของตนเองได้และขอใช้บริการเครื่องอื่นได้ เหมาะสำหรับการนำมาใช้งานเพื่อจุดประสงค์ในด้านความรวดเร็วหรือติดตั้งได้โดยง่ายเมื่อไม่มีโครงสร้างพื้นฐานที่จะรองรับ ยกตัวอย่างเช่น ในศูนย์ประชุม, หรือการประชุมที่จัดขึ้นนอกสถานที่

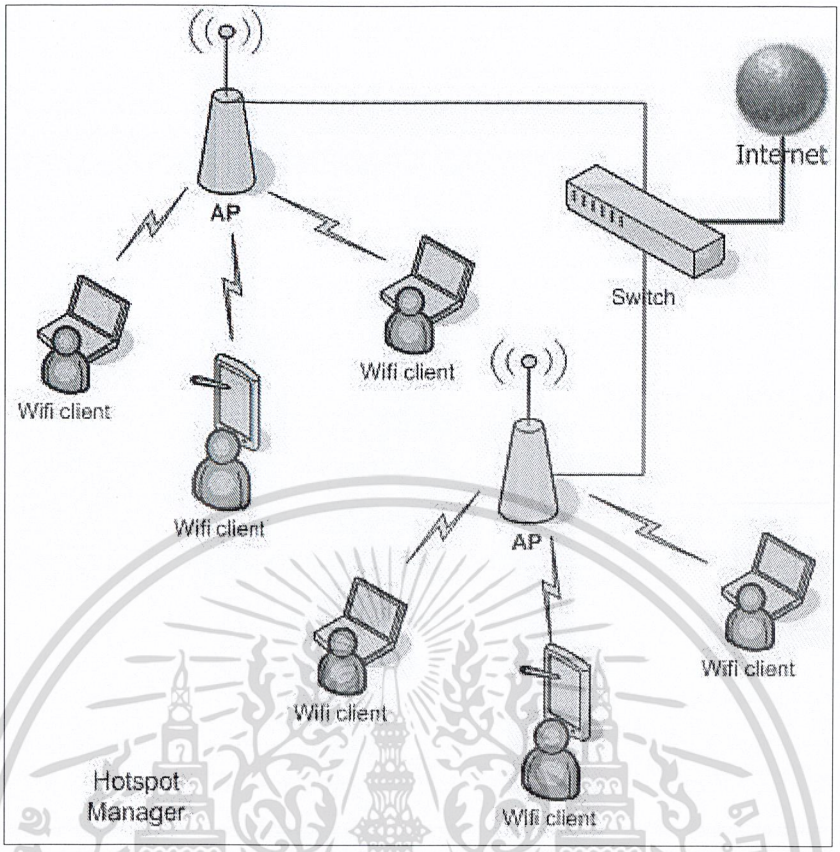


รูปที่ 2.2 การเชื่อมต่อแบบ Peer to Peer

2.1.2 Client/server (Infrastructure mode)

ระบบเครือข่ายไร้สายแบบ Client / server หรือ Infrastructure mode เป็นลักษณะการรับส่งข้อมูลโดยอาศัย Access Point (AP) หรือเรียกว่า "Hot spot" ทำหน้าที่เป็นสะพานเชื่อมต่อระหว่างระบบเครือข่ายแบบไร้สายกับเครื่องคอมพิวเตอร์ลูกข่าย (client) โดยจะกระจายสัญญาณคลื่นวิทยุเพื่อ รับ-ส่งข้อมูลเป็นรัศมีโดยรอบเครื่องคอมพิวเตอร์ที่อยู่ในรัศมีของ AP จะกลายเป็น เครือข่ายกลุ่มเดียวกันทันที โดยเครื่องคอมพิวเตอร์จะสามารถติดต่อกันหรือติดต่อกับ Server เพื่อแลกเปลี่ยนและค้นหาข้อมูลได้ โดยต้องติดต่อผ่านAP เท่านั้น ซึ่ง AP 1 จุดสามารถให้บริการเครื่องลูกข่ายได้ถึง 15-50 อุปกรณ์ ของเครื่องลูกข่าย เหมาะสำหรับการนำไปขยายเครือข่ายหรือใช้ร่วมกับระบบเครือข่ายแบบใช้สายเดิมในสำนักงาน ห้องสมุดหรือในห้องประชุม เพื่อเพิ่มประสิทธิภาพในการทำงานให้มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



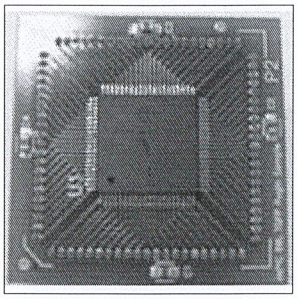
รูปที่ 2.3 การเชื่อมต่อแบบ Client

2.2 อุปกรณ์ในส่วนของการเชื่อมต่อ

2.2.1 ไมโครคอนโทรลเลอร์ dsPIC30F6014 และ PIC24FJ128GB110

ในการเชื่อมต่อกับอุปกรณ์ต่างๆ จำเป็นต้องมีการควบคุมการทำงานของอุปกรณ์นั้น ซึ่งทำได้หลายวิธี แต่วิธีที่มีประสิทธิภาพวิธีหนึ่ง คือ ใช้ไมโครคอนโทรลเลอร์ในการควบคุม ซึ่งง่ายต่อการออกแบบการทำงานให้ได้ตามที่ต้องการและในตัวไมโครคอนโทรลเลอร์เองนั้นก็มีฟังก์ชั่นที่ช่วยในการออกแบบการควบคุมทำให้สะดวกยิ่งขึ้นในการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 ไมโครคอนโทรลเลอร์ dsPIC30F6014

คุณสมบัติของ dsPIC30F6014

หน่วยประมวลผลกลาง

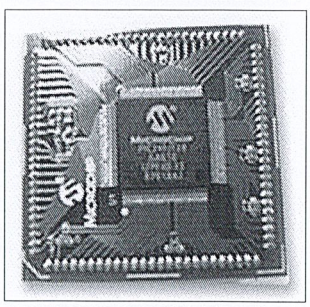
- ความเร็วระบบปฏิบัติการมากกว่า 30 MIPS ที่ 40 MHz
- สัญญาณนาฬิกา 4-10 MHz
- ริจิสเตอร์อเรย์ 16x16 bit
- ตำแหน่งของหน่วยความจำข้อมูลมีความยืดหยุ่น
- โครงสร้างภาษา C - Complier

คุณสมบัติ อะนาลอก

- 12 บิต ADC และมีอินพุทมากกว่า 16 ช่องสัญญาณ

อุปกรณ์ต่อพ่วงคอมพิวเตอร์

- รองรับโมดูล SPI (โหมด 4 เฟรม)
- URAT โมดูล
- 5 ไทเมอร์/เคาท์เตอร์ 16 บิต ซึ่งสามารถตั้งโปรแกรมได้



รูปที่ 2.5 ไมโครคอนโทรลเลอร์ PIC24FJ128GB110

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติของ PIC24FJ128GB110

หน่วยประมวลผลกลาง

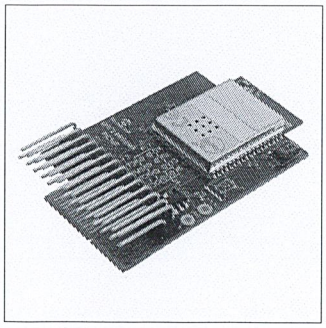
- ใช้สถาปัตยกรรมแบบฮาร์วาร์ด
- ความเร็วระบบปฏิบัติการมากกว่า 16 MIPS ที่ 32 MHz
- สัญญาณนาฬิกา 8 MHz
- ริจิสเตอร์อเรย์ 16x16 bit
- หน่วยความจำเชิงเส้น
- แยกอ่าน และเขียนตำแหน่งของหน่วยความจำข้อมูล
- โครงสร้างภาษา C - Complier

คุณสมบัติพิเศษ

- สามารถเลือกโหมดพลังงานเป็นโหมด Sleep, Idle and Doze และกลับมาทำงานได้อย่างรวดเร็ว
- มีความยืดยุ่นในการใช้วอทช์ด็อกไทเมอร์บนชิฟเซ็ต
- In-Circuit Debug(ICD) 2 pins
- ความจำแบบเฟรช

2.2.2 โมดูล Wi-Fi เบอร์ MRF24WB0MA

เป็นโมดูลแบบไร้สายที่อยู่ภายใต้มาตรฐาน IEEE 802.11 มีเสาสัญญาณภายในตัวจึงสะดวกและง่ายต่อการใช้งาน สามารถส่งสัญญาณได้มากกว่า 400 เมตร (1300ฟุต) ออกแบบใช้งานได้กับ ไมโครคอนโทรลเลอร์ในตระกูล PIC18 PIC24 และ dsPIC30 ซึ่งเป็นไมโครคอนโทรลเลอร์ที่ใช้งานอยู่ และติดต่อสื่อสารกันแบบ SPI (Serial Peripheral Interface)

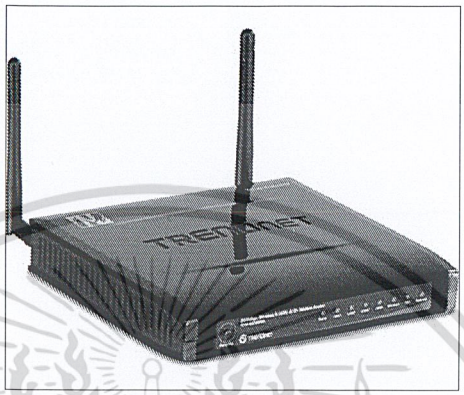


รูปที่ 2.6 โมดูล Wi-Fi เบอร์ MRF24WB0MA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 Router เบอร์ TEW-635BRM

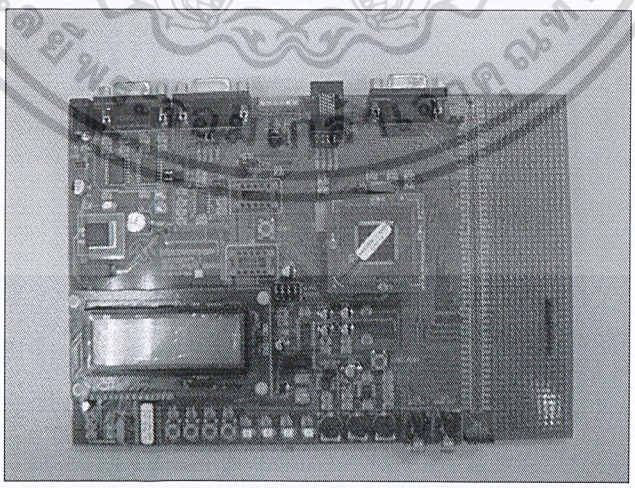
เป็นการรวมกันของโมเดม ADSL 2/2 กับ Router ไร้สายประสิทธิภาพสูง มีความเร็ว 12x4x และอยู่ภายใต้การเชื่อมต่อไร้สายแบบ G มีดับเบิ้ลไฟร์วอลและการเข้ารหัสแบบขั้นสูงช่วยให้มั่นใจในความปลอดภัยของข้อมูล



รูปที่ 2.7 Router เบอร์ TEW-635BRM

2.2.4 บอร์ดที่ใช้ในการเชื่อมต่อ

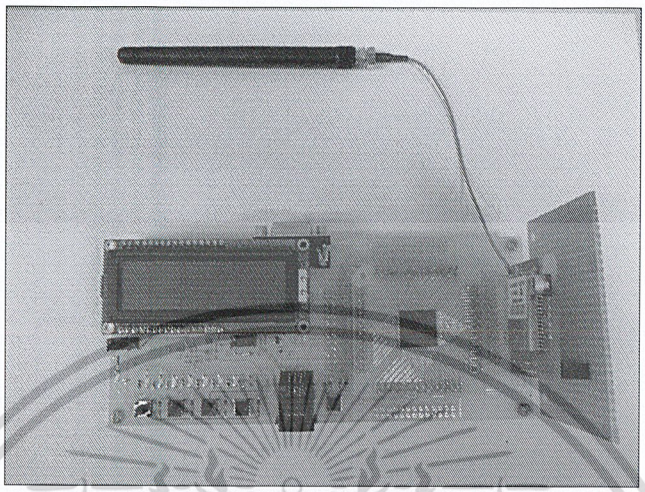
2.2.4.1 บอร์ดทดลอง dsPICDEM™ 1.1 Development Board เป็นบอร์ดที่ใช้ในการทดลองที่ 4.1 – 4.3 ซึ่งมีการเชื่อมต่อผ่านพอร์ตอนุกรม UART



รูปที่ 2.8 บอร์ดทดลอง dsPICDEM™ 1.1 Development Board

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4.2 บอร์ด Wi-Fi เป็นบอร์ดที่ใช้ในการทดลองที่ 4.4 ซึ่งมีการเชื่อมต่อผ่าน Wi-Fi



รูปที่ 2.9 บอร์ด Wi-Fi

2.3 รูปแบบของการติดต่อสื่อสารกันระหว่างไมโครคอนโทรลเลอร์กับโมดูล Wi-Fi

การติดต่อสื่อสารกันระหว่างไมโครคอนโทรลเลอร์ PIC24FJ128GB110 กับโมดูล Wi-Fi เบอร์ **MRF24WB0MA** เป็นการติดต่อสื่อสารกันแบบ SPI (Serial Peripheral Interface) ซึ่งเป็นการรับส่งข้อมูล ระหว่าง Master (ส่วนใหญ่จะเป็น MCU) กับ Slave (อาจจะเป็น MCU หรือ IC อื่นๆที่มี SPI ก็ได้) การรับส่งข้อมูลเป็นแบบ Full-Duplex คือสามารถรับ และส่งข้อมูลได้พร้อมกัน และเป็นแบบ Synchronous คือ การรับส่งข้อมูลจะเป็นไปตามจังหวะสัญญาณ Clock ที่ Master ส่งออกมา (แบบ Asynchronous จะไม่ใช้สัญญาณ Clock แต่จะใช้การตั้งค่า Baud rate ให้ตรงกัน)

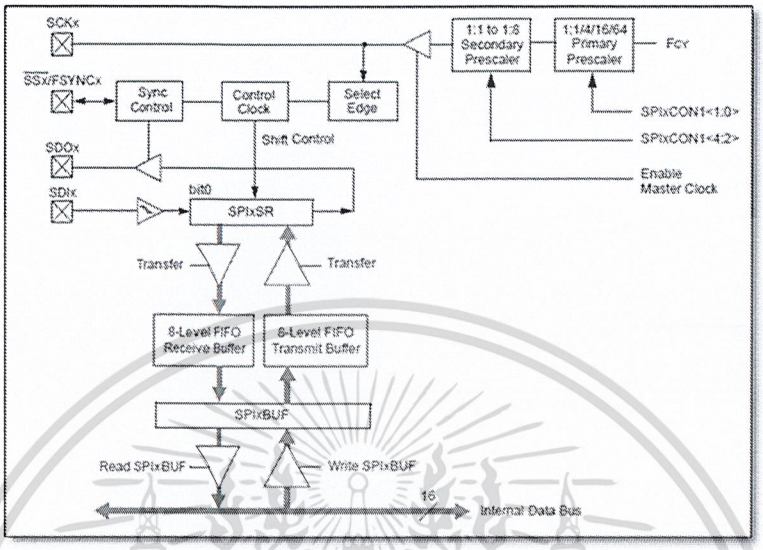
2.3.1 การเชื่อมต่อแบบ SPI

ส่วนการเชื่อมต่ออุปกรณ์อนุกรม หรือ Serial Peripheral Interface (SPI) เป็นมาตรฐานการเชื่อมต่อข้อมูลโดยอนุกรมซิงโครนัส ซึ่งถูกตั้งชื่อโดยโมโตโรล่า และ SPI นี้ทำงานแบบ Full Duplex เชื่อมต่อกันด้วยมาสเตอร์โหมดโดยตัวมาสเตอร์เป็นตัวส่งเฟรมข้อมูล แล้วใช้ตัวสเลฟเป็นตัวรับข้อมูล

การใช้งานโมดูลสื่อสารข้อมูลอนุกรม SPI โมดูล SPI เป็นโมดูลเชื่อมต่ออุปกรณ์ภายนอกแบบซิงโครนัส เพื่อใช้ในการสื่อสารกับอุปกรณ์ต่างๆ หรือกับไมโครคอนโทรลเลอร์ด้วยกัน เช่น การเชื่อต่อกับหน่วยความจำอีอีพรอมแบบอนุกรม ไอซีชิฟรีจิสเตอร์ อุปกรณ์แสดงผล อุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ดัดแปลง เผยแพร่ หรือจำหน่ายในเชิงพาณิชย์ ไม่ว่ากรณีใดๆ ทั้งสิ้น

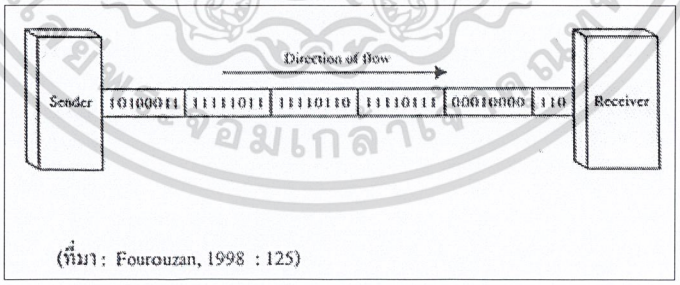
แปลงสัญญาณอะนาลอกเป็นดิจิตอล โดยอุปกรณ์เหล่านี้ต้องสามารถทำงานเข้าได้กับ SPI และการเชื่อมต่อแบบ SIOP (Simple Synchronous Serial I/O Port)



รูปที่ 2.10 ไดอะแกรมโมดูล SPI

2.3.2 การส่งข้อมูลแบบซิงโครนัส

การส่งข้อมูลแบบซิงโครนัส (Synchronous transmission) เป็นการส่งบิต 0 และ 1 ที่ต่อเนื่องกันไปโดยไม่มีการแบ่งแยก ผู้รับต้องแยกบิตเหล่านี้ออกมาเป็นไบต์ หรือเป็นตัวอักษร



รูปที่ 2.11 การส่งข้อมูลแบบซิงโครนัส

จากภาพแสดงการส่งข้อมูลแบบซิงโครนัส ผู้ส่งทำการส่งบิตติดต่อกันยาว ๆ ถ้าผู้ส่งต้องการแบ่งช่วงกลุ่มข้อมูลก็จะส่งกลุ่มบิต 0 หรือ 1 เพื่อแสดงสถานะว่าง เมื่อบิตมาถึงผู้รับ ผู้รับจะนับจำนวนบิตแล้วจับกลุ่มของบิตให้เป็นไบต์ที่มี 8 บิต การส่งข้อมูลแบบซิงโครนัสมีประสิทธิภาพมาก และใช้ความสามารถของสายสื่อสารได้เกือบทั้งหมด

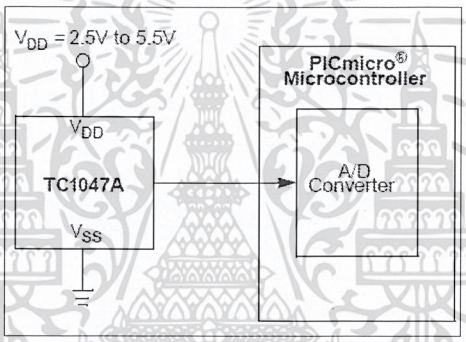
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีของการส่งข้อมูลแบบซิงโครนัส คือความเร็วในการส่งข้อมูล ทั้งนี้เพราะไม่มีบิตพิเศษหรือช่องว่างที่ไม่ได้ถูกนำไปใช้เมื่อถึงผู้รับ จึงทำให้ความเร็วของการส่งข้อมูลแบบซิงโครนัสเร็วกว่าแบบอะซิงโครนัส ด้วยเหตุนี้จึงมีการนำไปใช้งานที่ต้องการความเร็วสูง เช่น การส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์

2.4 อุปกรณ์การวัดอุณหภูมิ

ในโครงงานนี้ จะศึกษาและทดลองนำข้อมูลที่เป็นผลของอุณหภูมิที่วัดได้จากการทดลองนั้นไปประยุกต์ใช้งาน โดยนำข้อมูลไปแสดงผลด้วยการเชื่อมต่อแบบ Wi-Fi

2.4.1 เซนเซอร์วัดอุณหภูมิ เบอร์ TC1047A



รูปที่ 2.12 Block Diagram การเชื่อมต่อของเซนเซอร์คุณสมบัติของเซนเซอร์

คุณสมบัติของเซนเซอร์ที่นำมาใช้งาน

- ใช้แรงไฟฟ้า 2.5 ถึง 5.5 โวลต์
- วัดค่าอุณหภูมิได้ในช่วง -40 องศาเซลเซียส ถึง 125 องศาเซลเซียส
- มีความแม่นยำ ±2 องศาเซลเซียส
- วัดอุณหภูมิเป็นเชิงเส้น ความชัน 10 mV/ºC
- เชื่อมต่อกับไมโครคอนโทรลเลอร์ด้วย ADC

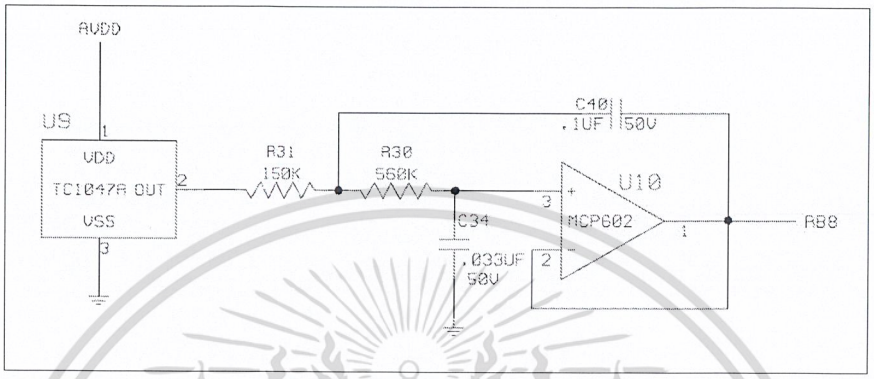
เนื่องจากค่าเอาท์พุตที่ได้จากเซนเซอร์วัดอุณหภูมิ เบอร์ TC1047A นั้นเป็นค่าแรงดันจึงต้องมีการแปลงค่าให้เป็นอุณหภูมิ ดังสมการต่อไปนี้

$$V_{OUT} = (10mV/^{\circ}C) \text{ CTemperature } ^{\circ}C) + 500mV \quad \ldots 2.1$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 ออปแอมป์ เบอร์ MCP602

ในโครงงานนี้ใช้แบบ Sallen key lowpass filter ทำหน้าที่กรองสัญญาณที่มีความถี่สูงๆออกจากวงจรวัดอุณหภูมิ



รูปที่ 2.13 แผนผังวงจรวัดอุณหภูมิ

2.5 รูปแบบการติดต่อสื่อสารกันระหว่างไมโครคอนโทรลเลอร์กับเซนเซอร์วัดอุณหภูมิ

จากที่ได้ศึกษาในส่วนของอุปกรณ์วัดอุณหภูมิและตัวไมโครคอนโทรลเลอร์พบว่า ทั้งสองส่วนนี้เชื่อมต่อกันโดยอาศัยการทำงานของฟังก์ชั่น Analog to Digital Converter ดังนั้นจึงจำเป็นต้องศึกษาการทำงานของฟังก์ชั่นตัวนี้เพื่อที่จะนำมาใช้งาน

การแปลงสัญญาณอะนาลอกเป็นดิจิตอล (Analog to Digital Converter)

สัญญาณที่ใช้ในอุปกรณ์อิเลคทรอนิคส์ มี 2 ชนิด คือ สัญญาณอนาล็อก และสัญญาณดิจิตอล สัญญาณอนาล็อก จะใช้ใน อุปกรณ์ทั่วๆ ไป และใช้ในการควบคุมแบบเก่า ซึ่งในปัจจุบันมีไมโครโปรเซสเซอร์ และไมโครคอนโทรลเลอร์ เข้ามาช่วยในการควบคุมอุปกรณ์ต่างๆ มากมาย ซึ่งทำให้การควบคุมนั้นทำได้ง่าย และรวดเร็วยิ่งขึ้น แต่ในการควบคุมนั้นเราจำเป็นต้องใช้สัญญาณดิจิตอลในการติดต่อกับไมโครโปรเซสเซอร์ หรือไมโครคอนโทรลเลอร์ แต่ในความเป็นจริงนั้น ใช้สัญญาณอนาล็อกในการควบคุม ดังนั้นจึงจำเป็น ต้องมีการเปลี่ยนสัญญาณอนาล็อก เป็นสัญญาณดิจิตอล แล้วจึงนำสัญญาณนั้นเข้ามาสู่ไมโครโปรเซสเซอร์ หรือไมโครคอนโทรลเลอร์

A หมายถึง Analog input D หมายถึง Digital I/O

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADRESH คือ ค่า register ผลลัพธ์ byte สูง ของการแปลงสัญญาณ

ADRESL คือ ค่า register ผลลัพธ์ byte ต่ำ ของการแปลงสัญญาณ

ADCON0 คือ register ควบคุมเกี่ยวกับ A/D byte ที่ 1

ADCON1 คือ register ควบคุมเกี่ยวกับ A/D byte ที่ 2

ADCON0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/(DONE\)	-	ADON

Bit7-6: ADCS1:ADCS0: เป็น bit ที่ใช้เลือกสัญญาณนาฬิกาในการแปลง A/D

00 = Fosc/2 Fosc = ความถี่ของ crystal ที่ใช้

01 = Fosc/8

00 = Fosc/32

11 = FRC (เลือกใช้ความถี่จากวงจร RC ที่อยู่ภายใน)

Bit5-3: CHS2:CHS0 เป็น bit ที่ใช้เลือก channel ของสัญญาณ A/D

000 = channel 0, (RA0/AN0)

001 = channel 1, (RA0/AN1)

000 = channel 2, (RA0/AN2)

000 = channel 3, (RA0/AN3)

000 = channel 4, (RA0/AN4)

000 = channel 5, (RA0/AN5) (ไม่มีใน mcu แบบ 28 pins)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

000 = channel 6, (RA0/AN6)

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

000 = channel 7, (RA0/AN7)

Bit 2: GO/(DONE\) : เป็นบิตที่ใช้ในการแสดงสถานะของการแปลง A/D

ถ้า ADON bit ถูก set เป็น 1 แล้ว เมื่อบิตนี้เป็น

1 หมายถึง A/D กำลังอยู่ในช่วงการแปลงค่า (ให้ set บิต นี้ในการเริ่มต้นการแปลงสัญญาณ)

0 หมายถึง A/D ไม่ได้อยู่ในช่วงการแปลงค่า (บิตนี้จะ MCU จะ clear เป็น 0 โดยอัตโนมัติ เมื่อทำการแปลงสัญญาณเสร็จเรียบร้อยแล้ว

Bit 1: ยังไม่ถูกใช้งาน

Bit 0: ADON: A/D On bit (บอกสถานะของ A/D ในขณะนั้น)

1 = A/D convertor กำลังถูกใช้งาน

0 = A/D convertor ไม่ได้ถูกใช้งาน

ADCON1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADFM	-	-	-	PCFG3	PCFG2	PCFG1	PCFG0

Bit 7: ADFM:

1 = หลังจากแปลงสัญญาณ ให้ด้านซ้ายของ ADRESH เป็น 0

ADRESH	ADRESL
000000xx	xxxxxxxx

0 = หลังจากแปลงสัญญาณให้ด้านขวาของ ADRESL เป็น 0

ADRESH	ADRESL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

xxxxxxxx	xx000000

Bit 6-4: ไม่ได้ถูกใช้

Bit 3-0: PCFG3:PCFG0 เป็นตัว set คุณสมบัติต่างๆ ในการทำงาน A/D ให้กับ PIC โดยเราสามารถเลือกว่าจะใช้ VREF แยกต่างหากหรือจะใช้จาก VDD, VSS เลยก็ได้

เมื่อการแปลง A/D เสร็จสิ้น ผลลัพธ์ของการแปลง A/D จะมีขนาด 10 bit ซึ่งจะเก็บอยู่ใน Register 2 ตัวต่อกันคือ ADRESH:ADRESL ส่วน register bit GO/DONE\ (ADCON0<2>) จะถูก cleared และ ADIF จะถูก set (A/D interrupt flag) Block diagram ของ A/D

2.6 การใช้งานโมดูลไทเมอร์ในไมโครคอนโทรลเลอร์

โมดูลไทเมอร์ภายใน มีขนาด 16 บิต สามารถทำงานร่วมกันเป็นไทเมอร์ 32 บิตได้ โดยประกอบไปด้วย timer1 timer2 และ timer3 จำนวนไทเมอร์ขึ้นอยู่กับเบอร์ที่เลือกใช้งาน (มีไทเมอร์ 5 ตัว) การทำงานของไทเมอร์จะเป็นตัวจับเวลาหรือตัวนับขึ้นอยู่กับการนับสัญญาณ หากเป็นการนับสัญญาณนาฬิกาภายในตัวไมโครคอนโทรลเลอร์ จะเรียกว่าตัวจับเวลา หรือไทเมอร์ และหากเป็นการนับสัญญาณนาฬิกาจากภายนอกจะเรียกว่า ตัวนับ หรือ เคาน์เตอร์ โดยที่แบ่งการทำงานของไทเมอร์ในรูปแบบฐานเวลา ได้ 3 รูปแบบ คือ

- ไทเมอร์ฐานเวลาแบบ A (Type A time base) สำหรับ Timer1
- ไทเมอร์ฐานเวลาแบบ B (Type B time base) สำหรับ Timer2, Timer4
- ไทเมอร์ฐานเวลาแบบ C (Type C time base) สำหรับ Timer3, Timer5

แต่ละโมดูลไทเมอร์จะมีรีจิสเตอร์ขนาด 16 บิต โดยรีจิสเตอร์สามารถอ่านเขียนได้ดังนี้

1. รีจิสเตอร์ TMRx ใช้ในการนับค่าขนาด 16 บิต
2. รีจิสเตอร์ PRx เป็นรีจิสเตอร์คาบเวลา (Period Register) ใช้กำหนดขอบเขตของการนับของ รีจิสเตอร์ TMRx
3. รีจิสเตอร์ TxCON ใช้กำหนดคุณสมบัติและควบคุมการทำงานของไทเมอร์

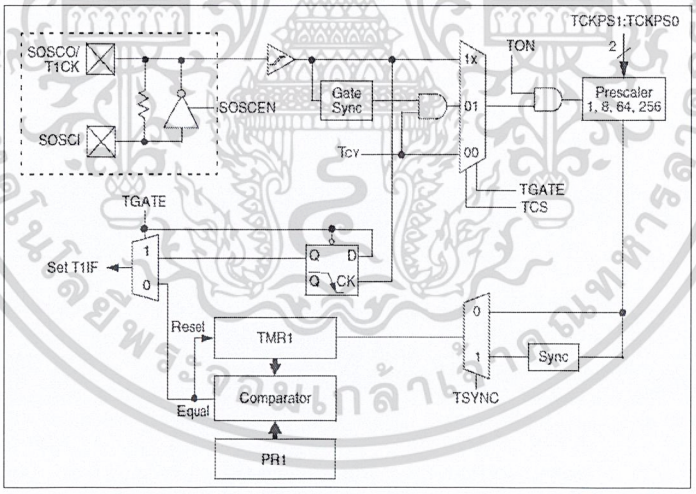
แต่ละโมดูลไทเมอร์จะมีบิตที่เกี่ยวข้องกับอินเตอร์รัปต์ของโมดูลไทเมอร์ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. บิต TxIE (Interrupt Enable Control bit) ควบคุมการเปิดใช้งานอินเตอร์รัปต์ของไทเมอร์
2. บิต TxIF (Interrupt Flag Status bit) แสดงสถานการณ์เกิดอินเตอร์รัปต์ของไทเมอร์
3. บิต TxIP (Interrupt Priority Control bits) ขนาด 3 บิต ใช้กำหนดระดับความสำคัญของอินเตอร์รัปต์ของไทเมอร์

ในโครงงานนี้เลือกใช้งานฐานเวลาแบบ A ซึ่งเป็นโมดูลไทเมอร์พื้นฐานที่มีในไมโครคอนโทรลเลอร์ มีคุณสมบัติดังนี้

- สามารถทำงานกับออสซิเลเตอร์ความถี่ต่ำที่ 32 kHz ได้ (Low Power Oscillator)
- สามารถทำงานในโหมดการนับแบบอะซิงโครนัส ได้จากแหล่งสัญญาณนาฬิกาภายนอกซึ่งมีเฉพาะในฐานเวลาแบบ A เท่านั้น
- สามารถใช้สร้างแอพพิเคชันฐานเวลานาฬิกาจริง หรือ ที่เรียกว่า Real-Time Clock(RTC)



รูปที่ 2.14 ไดอะแกรมไทเมอร์ฐานเวลาแบบ A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

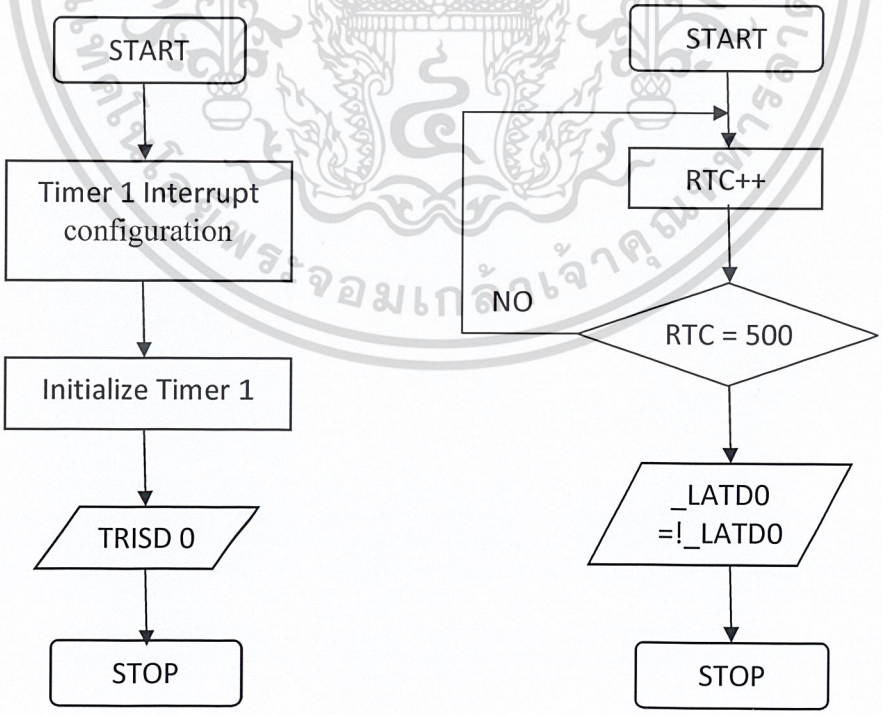
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

บทที่ 3
หลักการออกแบบ

จากที่ได้ศึกษาข้อมูลทั้งหมดในเบื้องต้นแล้ว การควบคุมและประมวลผลจำเป็นต้องสั่งผ่านไมโครคอนโทรลเลอร์ ซึ่งการส่งข้อมูลโดยพื้นฐาน ต้องอาศัยไทเมอร์เป็นตัวกำหนดเวลาในการรับส่ง จากนั้น ใช้ฟังก์ชัน SPI เป็นตัวเชื่อมต่อไมโครคอนโทรลเลอร์เข้ากับโมดูล Wi-Fi ในโครงงานนี้จะใช้กรณีศึกษาโดยนำผลอุณหภูมิที่วัดได้จากเซนเซอร์ซึ่งแสดงผลบนคอมพิวเตอร์ผ่าน Wi-Fi เนื่องด้วยข้อมูลที่ได้จากเซนเซอร์เป็นอะนาลอก เราจึงต้องใช้ฟังก์ชัน ADC ในไมโครคอนโทรลเลอร์สั่งงาน

3.1 การออกแบบการใช้งานไทเมอร์

การออกแบบในส่วนแรกจะเริ่มจากการเลือกค่าการอินเตอร์รัปต์ของไทเมอร์ซึ่งในโครงงานนี้เลือกใช้ไทเมอร์ชนิด A (Timer 1) จากนั้นเปิดการทำงานของฟังก์ชันและให้แสดงผลออกมาที่พอร์ต D ซึ่งสามารถเขียนเป็นกระบวนการทำงานได้ดังนี้



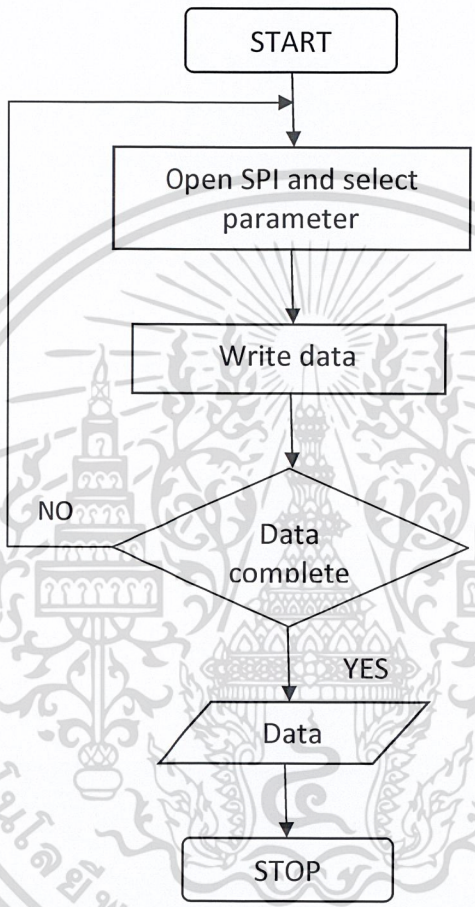
รูปที่ 3.1 Flow Chart การทำงานของฟังก์ชั่นไทเมอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

119382

3.2 การเชื่อมต่อแบบ SPI

กระบวนการทำงานของ SPI เริ่มจากการเปิดใช้งานโมดูล SPI และเลือกค่าพารามิเตอร์ต่างๆ ที่จะใช้งานจากนั้นทำการเขียนข้อมูลที่ต้องการส่ง แสดงการทำงานได้ดังขั้นตอนต่อไปนี้



รูปที่ 3.2 Flow Chart การทำงานของฟังก์ชั่น SPI

3.3 การทำงานของ ADC มีขั้นตอนการทำงานดังนี้

3.3.1 เลือก set config ของ A/D โดย

- เลือก analog pins/ voltage reference ด้วย ADCON1
- เลือก A/D Input channel ด้วย ADCON0
- เลือก A/D conversion clock (ความถี่ของสัญญาณนาฬิกาที่จะใช้ใน A/D) จาก ADCON0
- สั่งให้ A/D module ทำงาน ด้วย ADCON0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 ถ้าต้องการใช้ A/D interrupt ต้อง set flag ต่างๆ ดังนี้

- clear ADIF bit
- set ADIE bit
- set GIE bit

3.3.3 รอเวลาเพื่อให้ A/D module พร้อม (acquisition time) หาได้จากการคำนวณ

3.3.4 เริ่มทำการ A/D ด้วยการ set ON bit

3.3.5 รอจนกว่าการแปลงA/D จะเสร็จสมบูรณ์ ซึ่งดูได้ 2 วิธี

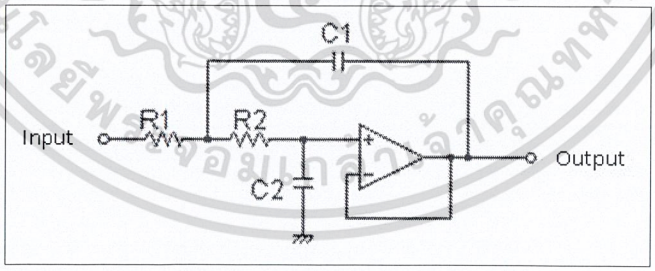
- เขียนโปรแกรมวน loop รอดู ON bit จนกว่าจะ cleared
- รอ A/D interrupt

3.3.6 อ่านค่าผลลัพธ์ของ A/D จาก ADRESH:ADRESL โดย clear ADIF bit หลังจาก การอ่านด้วยถ้าใช้ A/D interrupt

3.3.7 หลังจากอ่านค่าเรียบร้อยแล้ว ต้องรอเป็นระยะเวลา 2TAD (มีอยู่ใน data sheet) ก่อนที่จะทำการแปลง A/D อีกครั้งหนึ่ง

3.4 การออกแบบ Low pass filter

เนื่องจากเซนเซอร์วัดอุณหภูมินั้นให้เอาท์พุตเป็นสัญญาณอะนาลอก ดังนั้นเราจึงนำวงจรกรองสัญญาณมาใช้เพื่อกรองสัญญาณรบกวนซึ่งมีความถี่สูงออก

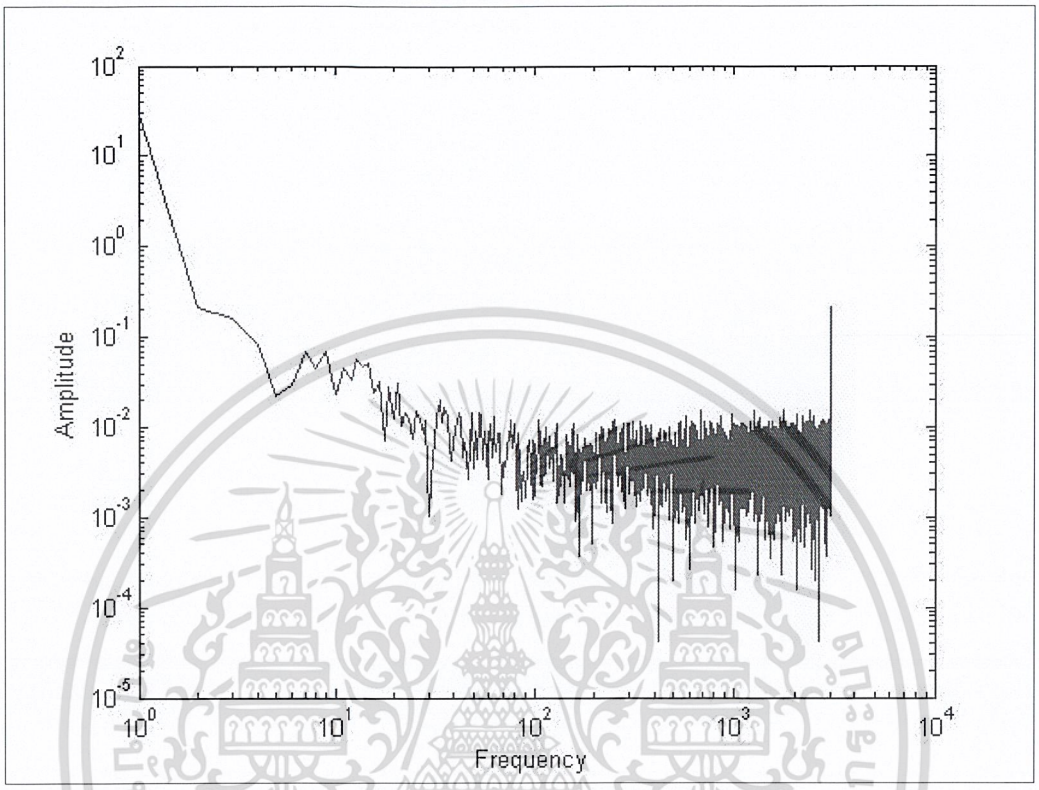


รูปที่ 3.3 แผนผังวงจรกรองสัญญาณความถี่ต่ำ

กำหนดให้ความถี่คัทออฟเท่ากับ 10 เฮิร์ต เนื่องด้วยค่าอุณหภูมิที่ได้จากการแสดงผลของเซนเซอร์มีการเปลี่ยนแปลงน้อย พิสูจน์ได้จากการหาสเปกตรัมของข้อมูลซึ่งพล็อตในโปรแกรม MATLAB จะเห็นว่าข้อมูลที่ได้ส่วนใหญ่อยู่ในช่วงความถี่ต่ำ ดังนั้นในโครงงานครั้งนี้จึงเลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่คัทออฟเท่ากับ 10 เฮิร์ต ซึ่งสามารถเลือกความถี่คัทออฟค่าต่ำกว่านี้ได้ แต่ Order ของFilter ก็ต้องสูงขึ้นตามและมีการออกแบบที่ซับซ้อนมากขึ้น



รูปที่ 3.4 กราฟสเปกตรัมของข้อมูล

ดังนั้นจึงเลือกค่า *R* และค่า *C* ดังนี้

R1 = 150 K , *R2* = 560 K

C1 = 0.1 UF , *C2* = 0.033 UF

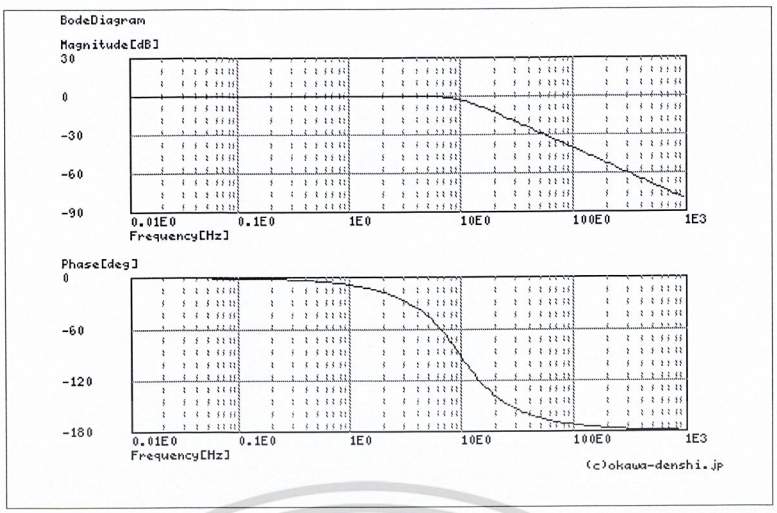
จากที่กำหนดค่าต้านทานและค่าตัวเก็บประจุแล้วจะได้ฟังก์ชันถ่ายโอน ดังสมการต่อไปนี้

$$G(s) = \frac{3608}{s^2 + 84.52s + 3608}$$

สามารถหาค่าเกนได้ดังนี้ G (0) = 1

จากฟังก์ชันถ่ายโอนนำมาพล็อตกราฟผลตอบสนองเชิงความถี่ ที่มีความถี่คัทออฟที่ประมาณ 10 เฮิร์ต ดังนี้

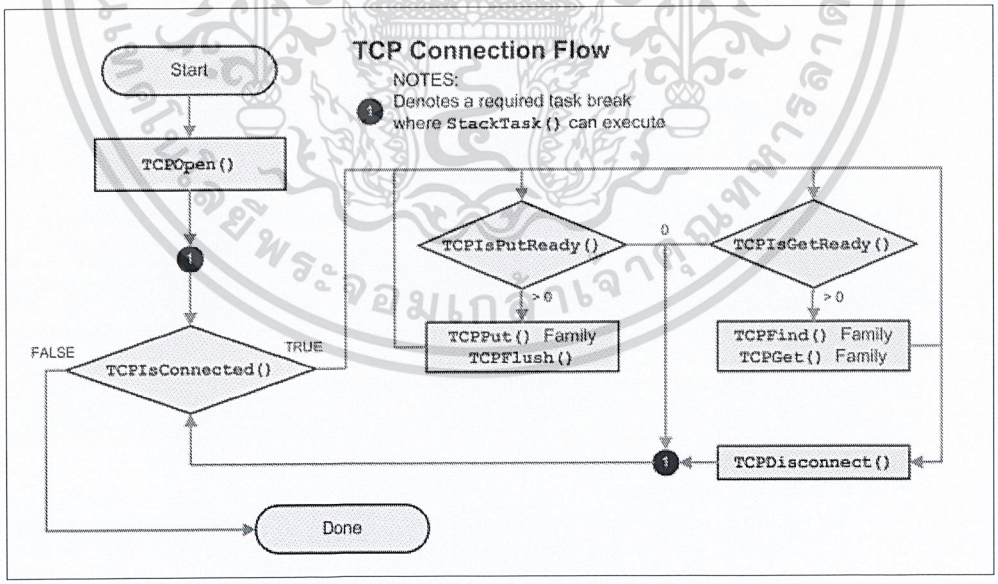
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 กราฟ Bode Diagram แสดง Frequency Response

3.5 การเชื่อมต่อแบบWi-Fi บนโมดูล TCP

TCP เป็นมาตราฐานการเชื่อมต่อโปรโตคอล ดำเนินการในบัฟเฟอร์ รูปที่ 3.6 แสดงภาพรวมการใช้งานของโมดูล TCP



รูปที่ 3.6 ภาพรวมการใช้งานของโมดูล TCP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.6 มีการทำงานดังนี้

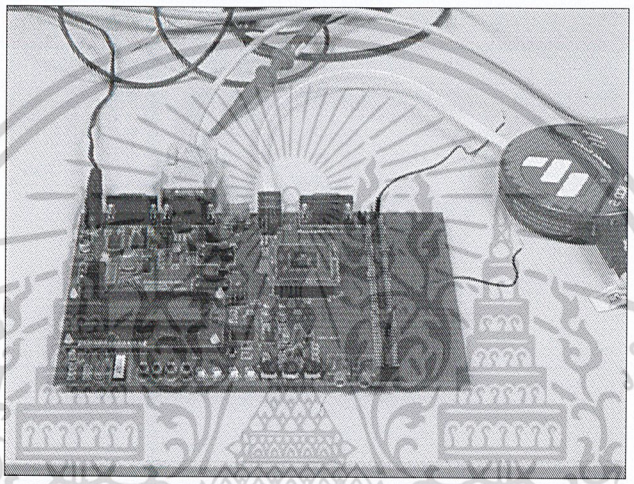
- เริ่มต้นเปิดใช้งานโมดูล TCP
- โมดูล TCP มีการเชื่อมต่อ และ พร้อมที่จะส่งข้อมูลเข้าสู่บัฟเฟอร์
- บัฟเฟอร์รับค่า และพร้อมส่งข้อมูล
- แสดงผลข้อมูลที่ถูกส่งมาจากบัฟเฟอร์
- จากนั้นทำการเชื่อมต่อและส่งข้อมูลใหม่อีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

จากที่ได้ศึกษาและออกแบบโปรแกรมการใช้งานมาแล้ว จึงทำการทดลองเพื่อให้ไมโครคอนโทรลเลอร์ประมวลผลและแสดงข้อมูลออกมา ซึ่งมีอุปกรณ์ในการเชื่อมต่อดังรูปด้านล่างนี้



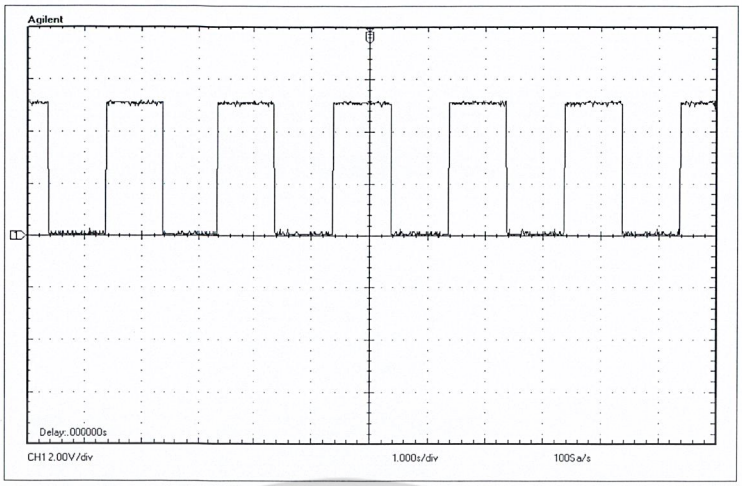
รูปที่ 4.1 การเชื่อมต่อของอุปกรณ์ต่าง ๆ ในการทดลองที่4.1 -4.3

ซึ่งเราแบ่งการทดลองเป็น 3 การทดลอง ดังนี้

4.1 การทดลองการใช้งาน Timer

เมื่อนำโปรแกรมการใช้งาน Timer ที่ออกแบบมาเบิร์นลงไมโครคอนโทรลเลอร์ แล้วเชื่อมต่อบอร์ดทดลองกับสโคป แล้วบันทึกผลการทดลอง จึงได้ผลการทดลองดังกราฟต่อไปนี้

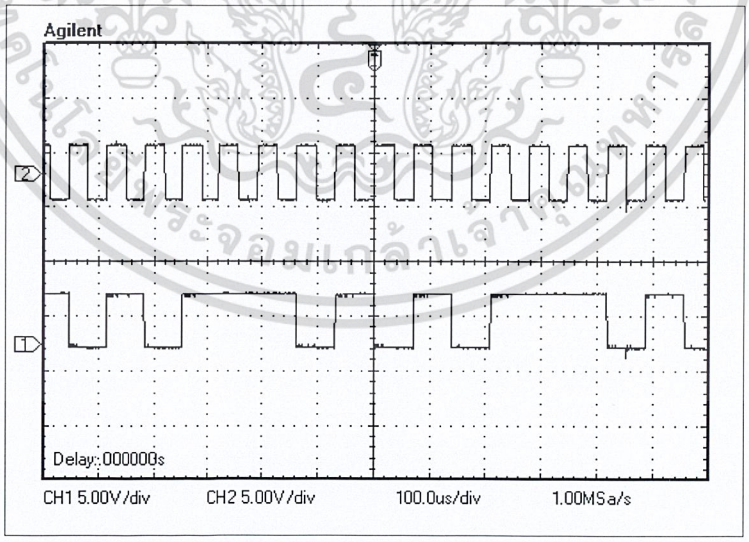
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 กราฟที่ได้จากการทดลองการใช้งาน Timer1

4.2 การทดลองการใช้งาน SPI

เมื่อนำโปรแกรมการใช้งาน SPI ที่ออกแบบมาเบิร์นลงไมโครคอนโทรลเลอร์ แล้วเชื่อมต่อบอร์ดทดลองกับสโคป แล้วบันทึกผลการทดลอง จึงได้ผลการทดลองดังกราฟต่อไปนี้



รูปที่ 4.3 กราฟที่ได้จากการทดลองการใช้งาน SPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองการวัดอุณหภูมิจากเซนเซอร์

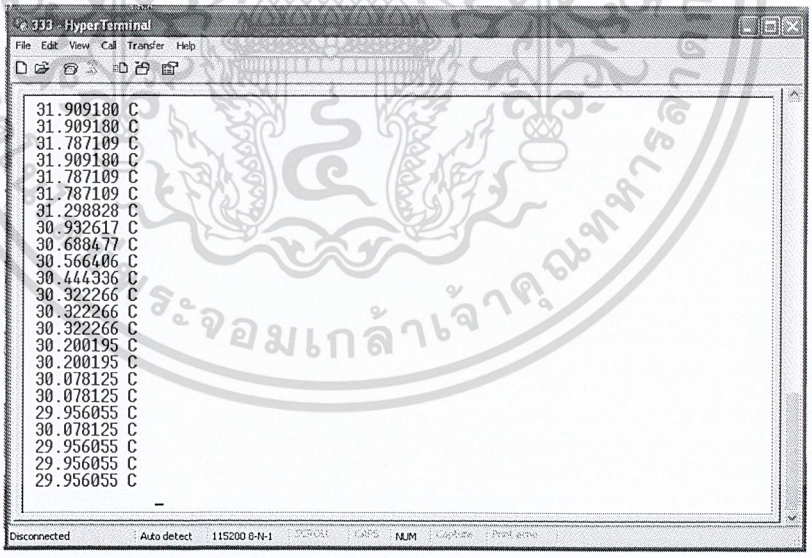
4.3.1 เบิร์นโปรแกรมที่เขียนสมบูรณ์แล้วลงในไมโครคอนโทรลเลอร์เบอร์ dsPIC60F6014 บนบอร์ดทดลอง dsPICDEM™ 1.1 Development Board

4.3.2 นำบอร์ดทดลอง dsPICDEM™ 1.1 Development Board เชื่อมต่อกับคอมพิวเตอร์ โดยต่อสายเคเบิลระหว่างพอร์ต RS232 ที่บอร์ดและพอร์ต RS232 ที่คอมพิวเตอร์เพื่อทำการติดต่อสื่อสารแบบอนุกรม UART

4.3.3 เข้าโปรแกรม Hyper Terminal ในคอมพิวเตอร์แล้วตั้งค่าการเชื่อมต่อโดยตั้งค่าช่องสัญญาณคือ COM1 และบอร์ดเรตเท่ากับ 115200 แล้วคลิกไอคอนเชื่อมต่อ (Connect)

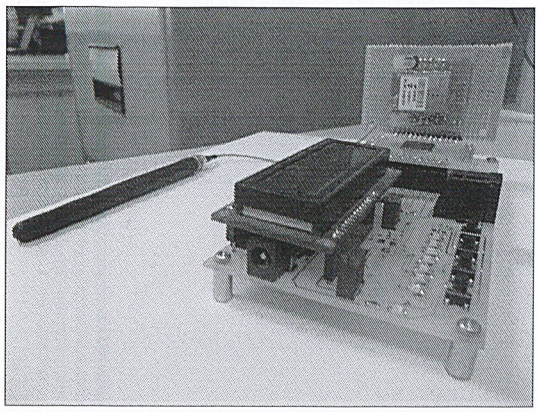
4.3.4 จากนั้นเราทดลองเปลี่ยนแปลงสภาพแวดล้อมภายนอกเพื่อให้อุณหภูมิเปลี่ยนแปลงแล้วทำการบันทึกผลการทดลอง

จากการทดลองจะได้ผลการทดลองดังรูปต่อไปนี้ จะเห็นได้ว่าที่โปรแกรม HyperTerminal จะแสดงค่าอุณหภูมิที่รับมาจากเซนเซอร์เบอร์ TC 1047A มีการเปลี่ยนแปลงค่าตามสภาพแวดล้อมภายนอกที่เปลี่ยนแปลงไป

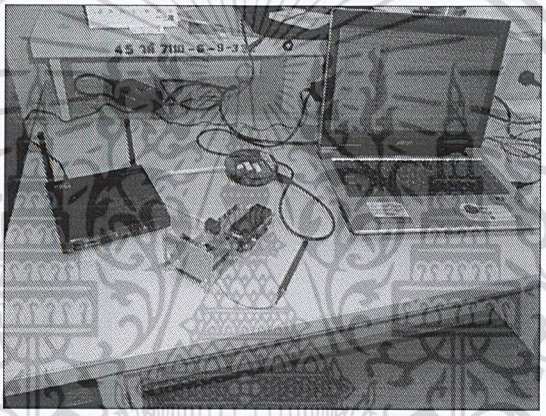


รูปที่ 4.4 ผลการทดลองจากโปรแกรม HyperTerminal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 บอร์ด Wi-Fi



รูปที่ 4.6 การเชื่อมต่อของอุปกรณ์ต่าง ๆ ในการทดลองที่ 4.4

4.4 การทดลองการเชื่อมต่อแบบ Wi-Fi

ในการทดลองครั้งนี้ ใช้บอร์ด Wi-Fi ดังรูปที่ 4.5 ในการทดลอง และมีการเชื่อมต่อดังรูปที่ 4.6 ซึ่งจากการทดลองที่ 4.3 นั้น เราได้นำวงจรเซนเซอร์วัดอุณหภูมิดังกล่าวมาเชื่อมต่อกับบอร์ด Wi-Fi แล้วทำการส่งข้อมูลผ่านโมดูล Wi-Fi โดยมีวิธีการทดลองดังนี้

4.4.1 เบิร์นโปรแกรมที่เขียนสมบูรณ์แล้วลงในไมโครคอนโทรลเลอร์เบอร์ PIC24FJ128GB110 บนบอร์ด Wi-Fi

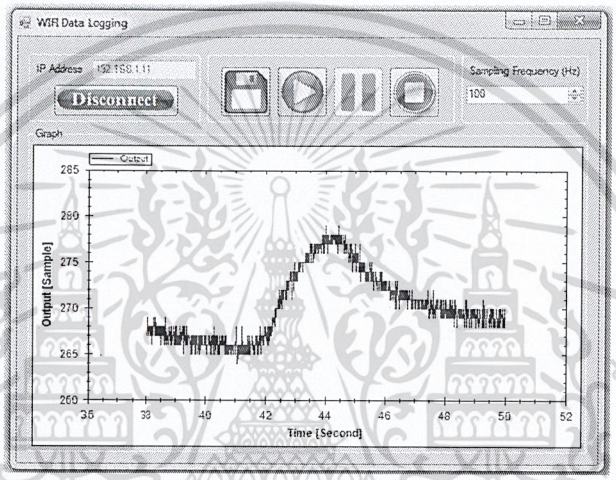
4.4.2 ตั้งค่าของ Router เบอร์ TEW-635BRM เพื่อกำหนดค่า IP Address และค่าการเชื่อมต่อต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

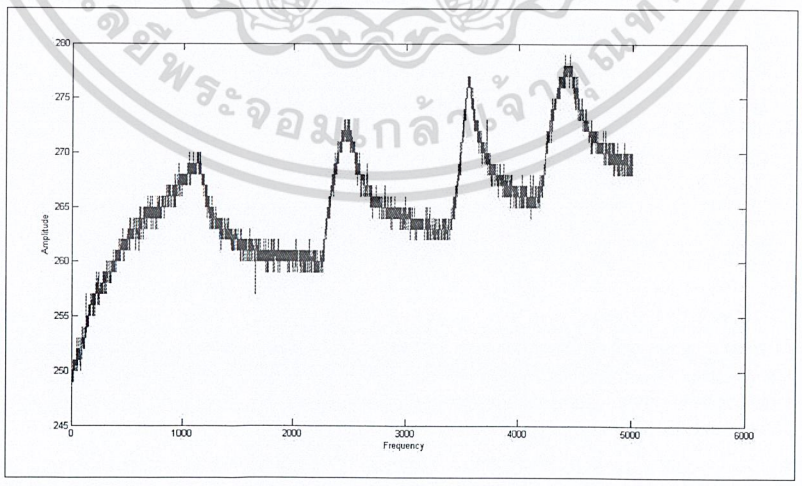
4.4.3 ส่งค่าอุณหภูมิจากเซนเซอร์ในบอร์ด Wi-Fi เข้าสู่ Notebook อีกเครื่องหนึ่ง ผ่านการเชื่อมต่อไร้สายแบบ Wi-Fi

4.4.4 เข้าโปรแกรม Wi-Fi Data logging ใน Notebook อีกเครื่องหนึ่ง เพื่อทำการรับค่าอุณหภูมิจากเซนเซอร์ในบอร์ด Wi-Fi ที่ส่งมาและบันทึกข้อมูล

4.4.5 นำค่าอุณหภูมิที่ได้จากโปรแกรม Wi-Fi Data logging ทำการแปลงค่า ADC เป็นค่าอุณหภูมิตามสมการ 2.1 ในบทที่ 2 และนำมาพล็อตกราฟในโปรแกรม MATLAB แล้วบันทึกผลการทดลอง

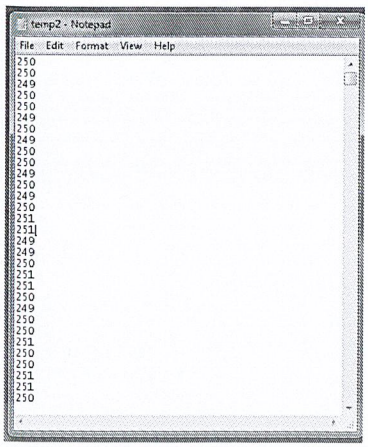


รูปที่ 4.7 โปรแกรม Wi-Fi Data Logging

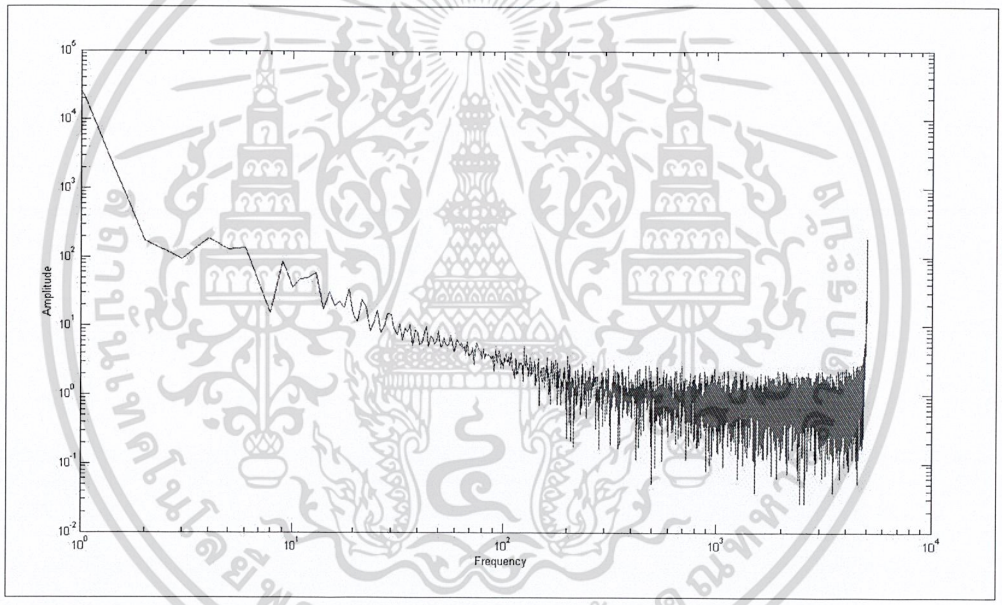


รูปที่ 4.8 กราฟผลการทดลองการเชื่อมต่อแบบ Wi-Fi

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 ข้อมูลจากกราฟการทดลองการเชื่อมต่อแบบ Wi-Fi



รูปที่ 4.10 กราฟ Log ที่ได้ในโปรแกรม MATLAB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

5.1 สรุปผลการทดลอง

5.1.1 สรุปผลการทดลองการใช้งาน Timer

จากการทดลองพบว่าสามารถกำหนดฐานเวลาได้ตามที่ต้องการ ซึ่งในที่นี้ต้องการให้อินเตอร์รัพป์ทุก ๆ 1 มิลลิเซค

5.1.2 สรุปผลการทดลองการใช้งาน SPI

จากการทดลองพบว่าสามารถเปิดใช้งานโมดูล SPI ได้ และข้อมูลที่ได้จากกราฟตรงกับข้อมูลที่ได้จากการสั่งงานในโปรแกรม ทำให้ทราบว่าสามารถติดต่อแบบ SPI ได้

5.1.3 สรุปผลการทดลองการวัดอุณหภูมิจากเซนเซอร์

จากการทดลองดังกล่าว เราสามารถวัดอุณหภูมิโดยใช้เซนเซอร์เบอร์ TC 1047A ในบอร์ดทดลอง dsPICDEM™ 1.1 Development Board แล้วได้ค่าอุณหภูมิซึ่งแสดงผลผ่านพอร์ตอนุกรม UART ด้วยโปรแกรม Hyper terminal ได้จริงและตรงกับอุณหภูมิที่เปลี่ยนแปลงไปตามสภาพแวดล้อมที่ทำการทดลอง ซึ่งสามารถพิสูจน์ได้โดยการคำนวณอุณหภูมิที่ได้จากการทดลองดังสมการต่อไปนี้

VOUT = (10mV/°C) C Temperature °C) + 500mV

รูปที่ 5.1 อุณหภูมิที่วัดได้จากเทอร์โมมิเตอร์

จากรูปที่ 5.1.3 จะเห็นว่าค่าอุณหภูมิที่วัดได้จากเทอร์โมมิเตอร์และอุณหภูมิที่ได้จากการทดลองมีค่าใกล้เคียงกัน ทำให้ทราบว่าโปรแกรมที่เราออกแบบสามารถใช้งานได้จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น และห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.4 สรุปผลการทดลองการเชื่อมต่อแบบ Wi-Fi

จากการทดลองพบว่าสามารถส่งค่าอุณหภูมิจากเซนเซอร์ในบอร์ด Wi-Fi เข้าสู่ Notebook ผ่านการเชื่อมต่อไร้สายแบบ Wi-Fi ได้จริง และสามารถบันทึกข้อมูลได้

5.2 ปัญหาที่พบและแนวทางแก้ไข

5.2.1 ปัญหาที่พบคือ เชื่อมต่อแบบอนุกรม UART กับคอมพิวเตอร์ไม่ได้ ซึ่งสามรถแก้ไขได้โดยกำหนดค่าบอร์ดเรตระหว่าง บอร์ดทดลองและคอมพิวเตอร์ไม่ตรงกัน

5.2.2 ปัญญาที่พบคือ ค่าอุณหภูมิที่ได้ให้ผลไม่เป็นจริงเนื่องจากกำหนดค่าระดับของ ADC ไม่ตรงกับที่บอร์ดกำหนด สามารถแก้ไขได้ที่โปรแกรม

5.3 ข้อเสนอแนะและแนวทางในการค้นคว้าพัฒนา

จากการศึกษาการเชื่อมต่อไร้สายแบบ Wi-Fi สามารถส่งข้อมูลชนิดอื่น ๆ ผ่านการเชื่อมต่อไร้สายแบบ Wi-Fi ได้ โดยไม่จำเป็นต้องเป็นค่าอุณหภูมิจากเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Code Timer1

```
#include<p30f6014.h>

_CONFIG2(FNOSC_PRIPLL & POSCMOD_XT)
_CONFIG1(JTAGEN_OFF & FWDTEN_OFF)

unsigned int RTC = 0;
void _ISR _T1Interrupt(void)
{
        RTC++;
        if(RTC == 500)
                {

                _LATD0 = !_LATD0;
                RTC = 0;
                }
        _T1IF = 0;
}

void Timer1(void)
{
        _NSTDIS = 1;
        _T1IF = 0;
        _T1IE = 1;
        TMR1 = 0;
        PR1 = 7373;
        T1CON = 0x8000;
}

int main(void)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    _TRISD0 = 0;
    Timer1();

    while(1);

}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Code SPI

```
#include<p30f6014.h>
#include "SPI.h"

_FOSC(CSW_FSCM_OFF & XT_PLL4);
_FWDT(WDT_OFF);
_FBORPOR(PBOR_OFF &  MCLR_EN);

#define Latch _LATF1

void SPI1Init(void){

        SPI1CON = 0;
        SPI1CONbits.DISSDO = 0; // Enable SDO PIN
        SPI1CONbits.MODE16 = 0; // 8-Bit Mode0
//      SPI1CONbits.MODE16 = 1; // 16-Bit Mode0
        SPI1CONbits.SSEN   = 0; // SS PIN not used
        SPI1CONbits.CKP = 1; // Clock Polarity
        SPI1CONbits.MSTEN = 1; // Master Mode
        SPI1CONbits.SPRE = 0b000; // Secondary Prescale
        SPI1CONbits.PPRE = 0b00; // Primary Prescale

        SPI1STAT = 0;
        SPI1STATbits.SPIEN = 1; // Enable SPI 1 Module
        //IEC0bits.SPI1IE = 1;
}

void SPIWriteData(unsigned char data){
        unsigned char dummy;
```

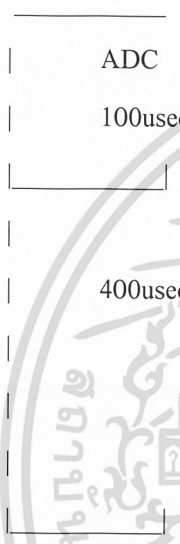
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        TRISF = 0;
        Latch = 1;
        SPI1BUF = data; //Load Data
        while(SPI1STATbits.SPIRBF == 0) //Wait for Transmission Complete
    Latch = 0;
        dummy = SPI1BUF; // Clear SPI1STATbits.SPIRBF

}

int main(void)
{
        unsigned int Data[3] = {0x00,0xEE,0xAA};
                unsigned int i;
                SPI1Init();
                while(1)
        {
                        for(i=0;i<5;i++)
                        {
                                SPIWriteData(0x01);

                }
 }

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Code ADC

```
/********************** ADCLogging.c ******************************
Hardware Nodes:          dsPIC30F4011 20I/P
                                  crystal 7.3728 MHz
UART Nodes:    Baud Rate 115200
AD Nodes: Sampling rate          100 microsecond
Analog Pin: AN2
```




```
System Sampling 500 usec
RE8 is used to control ADC.
RE0 is ligth if it properly works.
*****************************************************************/
#include <p30f6014.h>

/********************** Configuration Fuse ************************/
_FOSC(CSW_FSCM_OFF & XT_PLL4);                  /* Fuses for 7.3728MHz crystal */
_FWDT(WDT_OFF);
_FBORPOR(PBOR_OFF & BORV_27 & PWRT_16 & MCLR_EN);
_FGS(CODE_PROT_OFF);

/********************** Global variables **************************/
unsigned int RTC = 0;
float VoltageOut,T;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/********************** 500 usec Interrupt ***************************/
/*void __attribute__((__interrupt__)) _T1Interrupt(void)
{
        if (IFS0bits.T1IF) {
                IFS0bits.T1IF = 0; // reset Timer 1 interrupt flag
                RTC++;

                if (RTC == 1000){

                        printf("\r h\n ");

                        ADCON1bits.SAMP = 1;
                        while (ADCON1bits.DONE == 0);
                        ADCON1bits.DONE = 0;
                        IFS0bits.ADIF = 0;

                        VoltageOut = (5.0/1024.0)*ADCBUF0;
                        printf("\r %f  \n ", VoltageOut);
                        LATDbits.LATD1 =!LATDbits.LATD1;
                }
                RTC = 0;
        }

}*/
void _ISR _T1Interrupt(void)
{
        RTC++;
        if(RTC == 500)
        {
                ADCON1bits.SAMP = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                while (ADCON1bits.DONE == 0);
                ADCON1bits.DONE = 0;
                IFS0bits.ADIF = 0;

                VoltageOut = (5.0/4096.0)*ADCBUF0;
                //printf("\r %f C \n ",VoltageOut);
                T = (VoltageOut-0.5)/0.01;
                printf("\r %f C \n ",T);
                LATDbits.LATD0 =!LATDbits.LATD0;
                RTC = 0;
        }
        _T1IF = 0;
}
void Timer1(void)
{
        _NSTDIS = 1;
        _T1IF = 0;
        _T1IE = 1;
        TMR1 = 0;
        PR1 = 73728;
        T1CON = 0x8000;
}

/********************** Mainline **********************************/
int main(void)
{
        _TRISD0 = 0;
        Timer1();
//      INTCON1bits.NSTDIS = 1;                 /* Disable netting interrupt */
//      IFS0bits.T1IF = 0;                      /* Clear Timer1 interrupt flag
*/
//      IEC0bits.T1IE = 1;                      /* Enable Timer1 interrupt */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        /************** Initialize Timer1 ***********************************/
//      PR1 = 7373;                                          /* Interrupt every
500 microsecond
                                                             // Sampling
rate                    */
//      T1CONbits.TON = 1;                                   /* Start Timer1 */

        /************** Initialize UART *************************************/
        U1MODEbits.PDSEL = 0b00;                             /* 8 bit data, no parity */
        U1MODEbits.STSEL = 0;                                /* One stop bit */
        U1BRG = 3;                                           /* baud rate
115200 */
        U1MODEbits.UARTEN = 1;                               /* Enable UART
Module */
        U1STAbits.UTXEN = 1;                                 /* One at a time */

        /************** Initialize ADC **************************************/
        IEC0bits.ADIE = 0;                                   /* Disable Interrupt */
        /************** Set up pin for analog function **********************/
        ADPCFGbits.PCFG0 = 0;
        ADCHSbits.CH0SA = 0b1000;
        /************** Number of Sample/Hold *******************************/
        //ADCON2bits.CHPS = 0b00;
        ADCON1bits.SAMP = 0;
        /************** Select voltage reference ****************************/
        ADCON2bits.VCFG = 0b000;
        /************** Choose Time and Clock sample and convert ************/
        ADCON3bits.ADCS = 0b000010;
        /************** Channels scaning ************************************/
        ADCON2bits.CSCNA = 0;
        /************** Sample/Convert sequence *****************************/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ทั้งนี้ห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ADCON1bits.SSRC = 0b111;
        ADCON1bits.ASAM = 0;
        ADCON3bits.SAMC = 479;
        /********** Form data reading *********************************/
        ADCON1bits.FORM = 0b00;
        /********** Interrupt rate ************************************/
        ADCON2bits.SMPI = 0b0000;
        /********** Turn on A/D Module ********************************/
        ADCON1bits.ADON = 1;

        /************** Initialize Ports ********************************/
        TRISB = 0xFFF0;


        while (1) {         // loop forever
                printf("\r start\n ");
        while(1);
        } // end loop forever
} // end Mainline
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Code MATLAB for graph 3.4

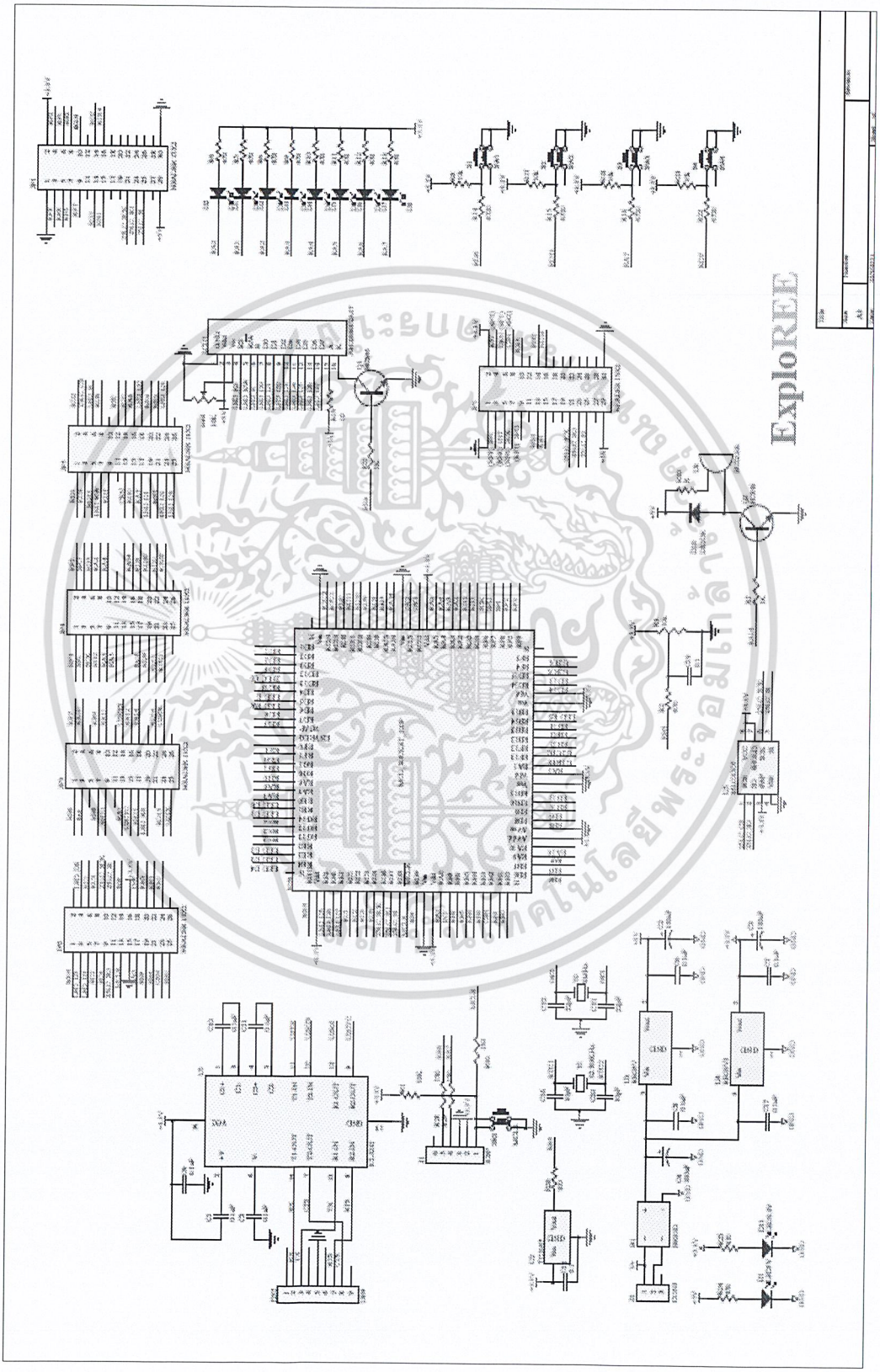
```
X = fft(temp1);
Plot (abs(X)/3046);
loglog(abs(X)/3046);
xlabel('Frequency');
ylabel('Amplitude');
```

Code MATLAB for graph 4.10

```
X = fft(temp2);
Plot (abs(X)/5004);
loglog(abs(X)/5004);
xlabel('Frequency');
ylabel('Amplitude');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Schematic ExploREE



MICROCHIP

TC1047/TC1047A

Precision Temperature-to-Voltage Converter

Features

- Supply Voltage Range:
  - TC1047: 2.7V to 4.4V
  - TC1047A: 2.5V to 5.5V
- Wide Temperature Measurement Range: -40°C to +125°C
- High Temperature Converter Accuracy: ± 2°C, Max, at 25°C
- Linear Temperature Slope 10mV/°C (typ.)
- Available in 3-Pin SOT-23B Package
- Very Low Supply Current:
  - 35µA Typical

Applications

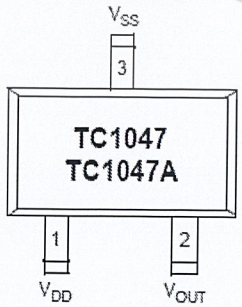
- Cellular Phones
- Power Supply Thermal Shutdown
- Temperature Controlled Fans
- Temperature Measurement/Instrumentation
- Temperature Regulators
- Consumer Electronics
- Portable Battery Powered Equipment

Device Selection Table

Part Number	Package	Temp. Range
TC1047VNB	3-Pin SOT-23B	-40°C to +125°C
TC1047AVNB	3-Pin SOT-23B	-40°C to +125°C

Package Type

3-Pin SOT-23B*



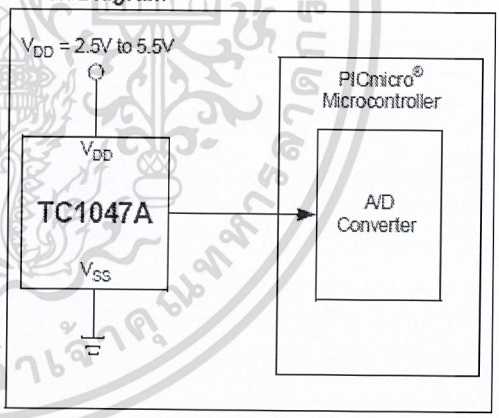
Note: * 3-Pin SOT-23B is equivalent to JEDEC TO-236

General Description

The TC1047 and TC1047A are linear voltage output temperature sensors whose output voltage is directly proportional to the measured temperature. The TC1047 and TC1047A can accurately measure temperature from -40°C to +125°C. With the TC1047, the supply voltage can vary between 2.7V and 4.4V. The power supply range of the TC1047A is from 2.5V to 5.5V.

The output voltage range for these devices is typically 100mV at -40°C, 500mV at 0°C, 750mV at +25°C, and 1.75V at +125°C. A 10mV/°C voltage slope output response allows for a predictable temperature measurement over a wide temperature range. The TC1047 and TC1047A are packaged in 3-Pin SOT-23B packages, making them ideal for space critical applications.

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น รวมทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารฉบับนี้ทุกครั้งที่มีการนำไปใช้

© 2002 Microchip Technology Inc.

TC1047/TC1047A

1.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings*

Supply Voltage ..... +7V

Voltage on Any Pin with Respect to Supplies ..... V_{SS} -0.3 to V_{DD} +0.3V

Operating Temperature ..... -40°C to +125°C

Storage Temperature Range ..... -55°C to +150°C

*Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions above those indicated in the operation sections of the specifications is not implied. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability.

TC1047/1047A ELECTRICAL SPECIFICATIONS

Electrical Characteristics: Unless otherwise specified, these specifications apply for the entire supply voltage range and for T_A = -40°C to +125°C

Symbol	Parameter	Min	Typ	Max	Units	Test Conditions
V_{DD}	Supply Voltage	2.7 2.5	— —	4.4 5.5	V	TC1047 TC1047A
I_Q	Supply Current, Operating	—	35	60	µA	
A_V	Average Slope of Output Voltage	—	10	—	mV/°C	
TMP_{ACY}	Temperature Accuracy	-2 -3 —	±0.5 ±0.5 1.0	+2 +3 —	°C °C °C	T_A = 25°C T_A = +125°C T_A = -40°C
V_{OUT}	Output Voltage	— 730 1720	100 750 1750	— 770 1780	mV mV mV	T_A = -40°C T_A = 25°C T_A = +125°C
I_{OUT}	Output Source and Sink Current	100	—	—	µA	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

© 2002 Microchip Technology Inc.

2.0 PIN DESCRIPTIONS

The descriptions of the pins are listed in Table 2-1.

TABLE 2-1: PIN FUNCTION TABLE

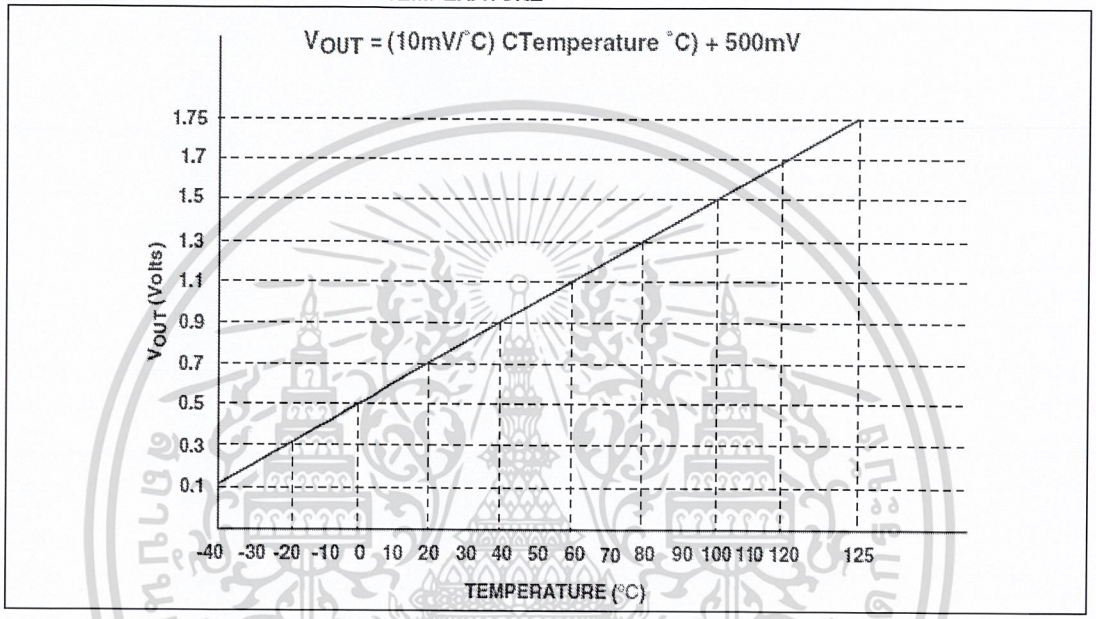
Pin No. (3-Pin SOT-23B)	Symbol	Description
1	V_{DD}	Input Supply Voltage
2	V_{OUT}	Temperature Sensor Output Terminal
3	V_{SS}	Ground Terminal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.0 DETAILED DESCRIPTION

The TC1047 and TC1047A have an output voltage that varies linearly with temperature in degrees Celsius. Figure 3-1 shows a plot of the output voltage versus temperature for the TC1047 and TC1047A. The temperature slope is fixed at 10mV/°C, and the output voltage at 0°C is 500mV.

FIGURE 3-1: OUTPUT VOLTAGE VS. TEMPERATURE



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

© 2002 Microchip Technology Inc.

4.0 TYPICAL CHARACTERISTICS

Note: The graphs provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore outside the warranted range.

FIGURE 4-1: TEMPERATURE ACCURACY VS. TEMPERATURE

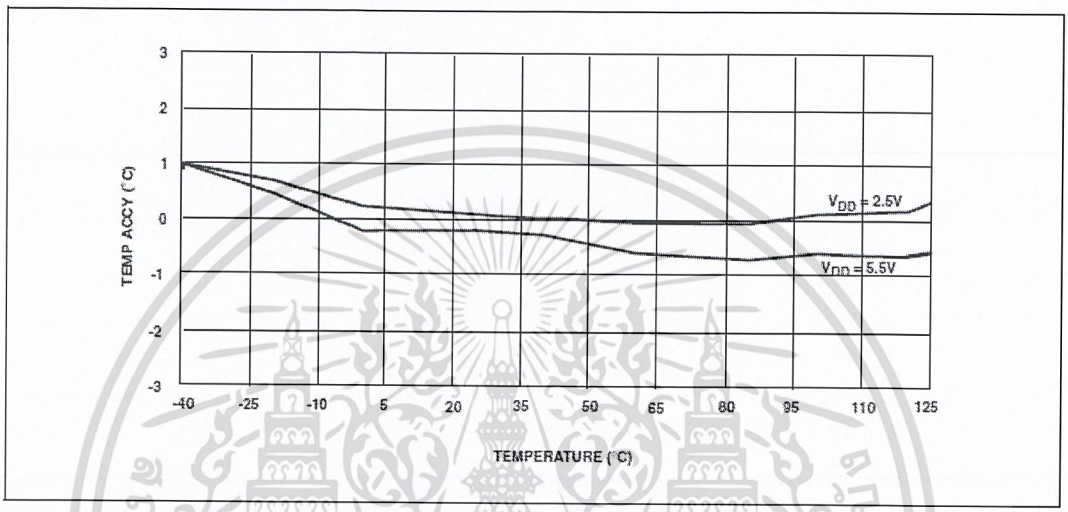
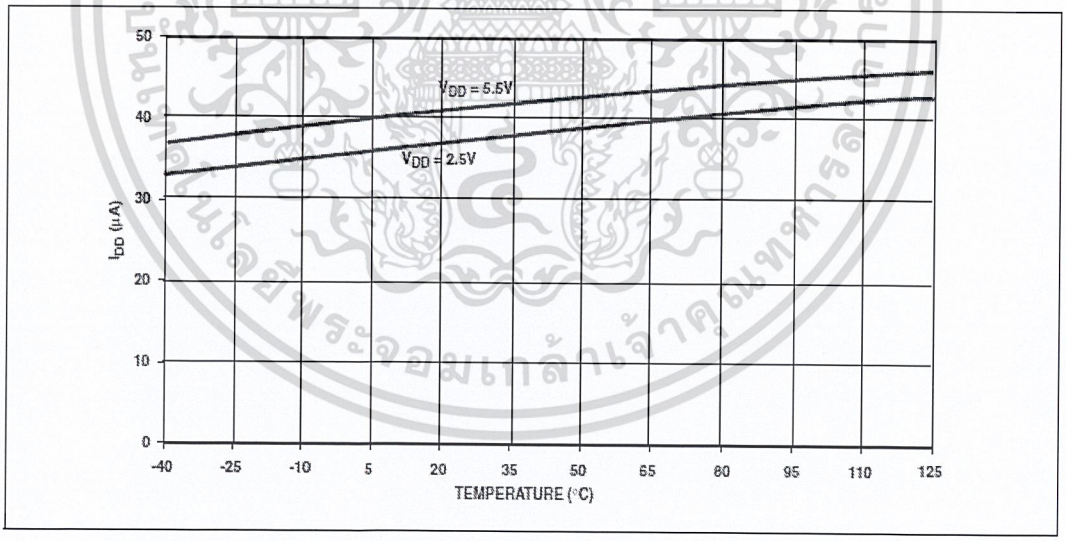


FIGURE 4-2: SUPPLY CURRENT VS. TEMPERATURE



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	X
Device	Temperature Range

Device:	DSTEMP:	High Precision Temp-to-Voltage Converter
	DSTEMPxTR:	High Precision Temp-to-Voltage Converter (Tape and Reel)
Temperature Range:	V =	-40°C to +125°C
Package:	NB =	3-pin SOT-23B

Examples:

a) DSTEMPVNBTR: High Precision Temp-to-Voltage Converter

SALES AND SUPPORT

Data Sheets

Products supported by a preliminary Data Sheet may have an errata sheet describing minor operational differences and recommended workarounds. To determine if an errata sheet exists for a particular device, please contact one of the following:

1. Your local Microchip sales office
2. The Microchip Corporate Literature Center U.S. FAX: (480) 792-7277
3. The Microchip Worldwide Site (www.microchip.com)

Please specify which device, revision of silicon and Data Sheet (include Literature #) you are using.

New Customer Notification System

Register on our web site (www.microchip.com/cn) to receive the most current information on our products.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] ประจิน พลังสันติกุล. **การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์ dsPIC30F ด้วยคอมไพเลอร์ MPLAB C.**

กรุงเทพมหานคร : บริษัท แอพซอฟต์เทค จำกัด. 2551.

[2] **"Sallen Key Lowpass Filter."** [Online]. Available :

http://sim.okawa-denshi.jp/en/OPstool.php. 2011.

[3] **"Sallen–Key topology."** [Online]. Available :

http://en.wikipedia.org/wiki/Sallen%E2%80%93Key_topology. 2011.

[4] "**การแปลงสัญญาณอนาล็อก-ดิจิตอล.**" [Online]. Available :

http://www.cpe.ku.ac.th/~yuen/204471/conversion/adc/. 2011.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้