ระบบแสดงอัตราการให้น้ำเกลือแบบไร้สาย

WIRELESS SALINE SOLUTION INJECTION RATE DISPLAYING SYSTEM

นายกรวิชญ์ ตัวลือ รหัส 51361520

นางสาวอัมวิกา ละอองเภา รหัส 51364590

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 20 ก.ค. 2558
เลขทะเบียน 16913829
เลขเรียกหนังสือ ปร.
มหาวิทยาลัยนเรศวร ก182 ร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต 2557

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2557

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อโครงงาน	ระบบแสดงอัตราการให้น้ำเกลือแบบไร้สาย			
ผู้ดำเนินโครงงาน	นายกรวิชญ์	ตัวลือ	รหัส	51361520
	นางสาวอัมวิกา	ละอองเภา	รหัส	51364590
ที่ปรึกษาโครงงาน	ดร.พรพิศุทธิ์	วรจิรันตน์		
สาขาวิชา	วิศวกรรมไฟฟ้า			
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์			
ปีการศึกษา	2557			

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมไฟฟ้า

.....ที่ปรึกษาโครงงาน

(ดร. พรพิศุทธิ์ วรจิรันตน์)

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. พนัส นัถฤทธิ์)

.....กรรมการ

(ดร.พิสุทธิ์ อภิชยกุล)

ชื่อหัวข้อโครงงาน	ระบบแสดงอัตราการให้น้ำเกลือแบบไร้สาย			
ผู้ดำเนินโครงงาน	นายกรวิชญ์	ตัวลือ	รหัส	51361520
	นางสาวอัมวิกา	ละอองเภา	รหัส	51364590
ที่ปรึกษาโครงงาน	ดร. พรพิศุทธิ์	วรจิรันตน์		
สาขาวิชา	วิศวกรรมไฟฟ้า			
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์			
ปีการศึกษา	2557			

.....

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอโครงงานเกี่ยวกับระบบแสดงอัตราการให้น้ำเกลือแบบไร้สาย โดยให้มีการรับและส่งข้อมูลแบบไร้สายผ่านทางโมดูล TRW 2.4 GHz โดยที่บอร์ดไมโครคอนโทรลเลอร์ตัวส่งจะเชื่อมต่อกับเซนเซอร์ตรวจจับสัญญาณที่ตรวจจับหยดน้ำเกลือแล้วทำการส่งข้อมูลไปยังไมโครคอนโทรลเลอร์ตัวรับที่เชื่อมต่อกับคอมพิวเตอร์ส่วนตัว ซึ่งมีระยะการรับส่งข้อมูลสูงสุด 16 เมตรในพื้นที่บริเวณปิด และมีการรับส่งข้อมูลสูงสุด 290 เมตรในพื้นที่บริเวณเปิด โดยแสดงผลข้อมูลของผู้ป่วยและจำนวนปริมาตรของน้ำเกลือที่ให้กับผู้ป่วยที่ลดลงเรื่อยๆ ผ่านโปรแกรม Visual Basic 6.0 เมื่อปริมาตรน้ำเกลือที่ให้กับผู้ป่วยใกล้หมดโปรแกรม Visual Basic 6.0 จะทำการแจ้งเตือน เพื่อให้พยาบาลหรือผู้ที่เกี่ยวข้องทำการเปลี่ยนน้ำเกลือให้กับผู้ป่วยต่อไป

Project title	Wireless Saline Solution Injection Rate Displaying System		
Name	Mr. Korawit	Tualue	ID. 51361520
	Miss Amwika	La-ongphao	ID. 51364590
Project advisor	Ponpisut	Worrajiran, Ph. D.	
Major	Electrical Engineering		
Department	Electrical and Computer Engineering		
Academic year	2014		

.....

Abstract

This thesis presented a project about the water rate system wirelessly. To accept and send data wirelessly via a module TRW 2.4 GHz Transmitter with Microcontroller Board is connected to a sensor that detects the signal drops of saline. Then send the data to a receiver connected to the microcontroller, personal computer. The distance data transmission maximum to 16 meters in close area and data transmission maximum to 290 meters in open area by display of the patient and the volume of saline to patients declining steadily through the Visual Basic 6.0 on volume of normal saline to patients near the end of a Visual Basic 6.0 will alert Nursing or related to changing the brine to the next patient.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จได้ด้วยความช่วยเหลืออย่างดีจาก ดร. พรพิศุทธ์ วรจิรันตน์ อาจารย์ที่ปรึกษาโครงงานที่กรุณาชี้แนะให้คำแนะนำในการดำเนินโครงงานและเขียนปริญญา-นิพนธ์ รวมทั้ง ผู้ช่วยศาสตราจารย์ ดร. พนัส นัถฤทธิ์ และ ดร. พิสุทธิ์ อภิชยกุล ที่กรุณาให้คำแนะนำตลอดจนแก้ไขข้อบกพร่องของปริญญานิพนธ์ด้วยการเอาใจใส่จนทำให้ปริญญานิพนธ์สำเร็จลุล่วงได้อย่างสมบูรณ์ ข้าพเจ้ารู้สึกซาบซึ้งในความกรุณาจึงขอขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้

ขอขอบคุณคณาจารย์ทุกท่านที่ประสิทธิ์ประสาทวิชาความรู้ให้กับคณะผู้ดำเนินโครงงาน

ขอขอบคุณภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร ที่เอื้อเฟื้อสถานที่ในการทดลอง และขอขอบคุณเจ้าหน้าที่ห้องปฏิบัติการทางไฟฟ้าที่ให้คำแนะนำในการใช้เครื่องมือ

สุดท้ายนี้ขอกราบขอบพระคุณบิดามารดาที่คอยเป็นกำลังใจ ให้ความรักความอบอุ่นหาที่สุดไม่ได้และให้การสนับสนุนอย่างไม่มีเงื่อนไขตลอดมา

นายกรวิชญ์ ตัวลือ

นางสาวอัมวิกา ละอองเภา

สารบัญ

หน้า

ใบรับรองปริญญานิพนธ์ ..... ก
บทคัดย่อภาษาไทย ..... ข
บทคัดย่อภาษาอังกฤษ ..... ค
กิตติกรรมประกาศ ..... ง
สารบัญ ..... จ
สารบัญตาราง ..... ซ
สารบัญรูป ..... ญ

บทที่ 1 บทนำ 1
1.1 ที่มาและความสำคัญของโครงงาน 1
1.2 วัตถุประสงค์ของโครงงาน 1
1.3 ผลที่คาดว่าจะได้รับ 2
1.4 ขอบเขตการทำโครงงานวิศวกรรม 2
1.5 ขั้นตอนการดำเนินงาน 2
1.6 งบประมาณ 3

บทที่ 2 หลักการและทฤษฎีเบื้องต้น 4
2.1 การสื่อสารไร้สาย 4
2.1.1 ความถี่ของคลื่น 4
ก. ย่านความถี่ ISM 5
ข. ย่านความถี่บลูทูธ กับ Wifi 6
2.2 ไมโครคอนโทรลเลอร์ ATTINY2313 7
2.2.1 คุณสมบัติของไมโครคอนโทรลเลอร์ ATTINY2313 7
2.3 ไมโครคอนโทรลเลอร์ ATMEGA32 8
2.3.1 คุณสมบัติของไมโครคอนโทรลเลอร์ ATMEGA32 8
2.4 ชุดเซนเซอร์ตรวจสอบการสะท้อนกลับของสัญญาณอินฟราเรด R – REFLEX S5 9
2.5 Wireless TRW 2.4 GHz 10
2.6 โปรแกรม Visual Basic 6.0 10

สารบัญ (ต่อ)

หน้า

2.7 โปรแกรมควบคุมการทำงานของไมโครคอนโทรเลอร์ 11
2.8 การส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม 12
2.8.1 การสื่อสารแบบอะซิงโครนัส (Asynchronous Transmission) 12
2.8.2 การสื่อสารแบบซิงโครนัส (Synchronous Transmission) 13
2.8.3 RS 232 (Recommended Standard 232) 15
2.9 น้ำเกลือ 15
2.9.1 การคำนวณอัตราหยดของน้ำเกลือ 16
2.10 ระบบตรวจเฝ้าระวังการให้น้ำเกลือแบบไร้สาย 17

บทที่ 3 วิธีการดำเนินโครงงาน 19
3.1 การออกแบบทางฮาร์ดแวร์ 21
3.1.1 การออกแบบบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) 21
3.1.2 การออกแบบบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) 25
3.1.3 การออกแบบส่วนเชื่อมโยงข้อมูล 27
3.2 การออกแบบโปรแกรม 28
3.2.1 การออกแบบโปรแกรมบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) 28
3.2.2 การออกแบบโปรแกรมบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) 31
3.3 โปรแกรมการแสดงผล 32
3.3.1 ปุ่ม Communication 33
3.3.2 ปุ่ม Date/Time 33
3.3.3 ปุ่ม Channel Setting 34
3.3.4 ปุ่ม Display 38
3.4 วิธีการทดสอบ 40
3.4.1 การทดสอบส่วนของซอฟแวร์ 40
3.4.2 การทดสอบการใช้งานอุปกรณ์ 40
ก. การทดสอบการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ 41
ข. การทดสอบการนับหยดน้ำเกลือ 42
ค. การทดสอบระยะทางของการรับส่งข้อมูล 44

สารบัญ (ต่อ)

หน้า

บทที่ 4 ผลการทดลองและสรุปผลการทดลอง 46

4.1 ผลการทดลองส่วนซอฟแวร์ 46

4.1.1 โปรแกรมที่ใช้ในบอร์ดไมโครคอนโทรลเลอร์ 46

4.1.2 วิธีการใช้งานเครื่องวัดหยดน้ำเกลือ 47

ก. ส่วนของการเชื่อมต่ออุปกรณ์ 47

ข. ส่วนของการติดตั้งโปรแกรม 48

ค. ส่วนของการตั้งค่าโปรแกรม 49

4.2 ผลการทดลองการใช้งานของอุปกรณ์ 51

4.2.1 ผลการทดลองการแสดงผลอัตราการให้น้ำเกลือ 51

4.2.2 ผลการทดลองจากการทดสอบการนับหยดน้ำเกลือ 53

ก. การทดลองย่อยที่ 1 53

ข. การทดลองย่อยที่ 2 59

ค. การทดลองย่อยที่ 3 65

4.2.3 ผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูล 71

ก. การทดลองย่อยที่ 1 72

ข. การทดลองย่อยที่ 2 77

4.3 สรุปผลการทดลอง 80

บทที่ 5 สรุปผลการดำเนินโครงงาน 82

5.1 ปัญหาและแนวทางการแก้ปัญหา 82

5.2 ข้อเสนอแนะในการพัฒนา 82

เอกสารอ้างอิง 84

ภาคผนวก ก 86

ภาคผนวก ข 99

ประวัติผู้เขียนโครงงาน 119

สารบัญตาราง

ตารางที่ หน้า

1.1 ขั้นตอนการดำเนินงาน 2

1.1 ขั้นตอนการดำเนินงาน (ต่อ) 3

2.1 แสดงการแบ่งความถี่ออกเป็นย่านต่างๆ 5

3.1 แสดงการเปรียบเทียบแรงดันที่จุดอ้างอิงกับแรงดันอินพุต 24

3.2 ตัวอย่าง Code ของตัวเลือก Use system time 34

3.3 ตัวอย่าง Code ของปุ่ม Save 35

3.3 ตัวอย่าง Code ของปุ่ม Save (ต่อ) 36

3.3 ตัวอย่าง Code ของปุ่ม Save (ต่อ) 37

3.4 ตัวอย่าง Code ของการแจ้งเตือนน้ำเกลือหมด 39

4.1 ตัวอย่างการตั้งค่าข้อมูลของผู้ป่วย 50

4.2 แสดงผลการนับหยดน้ำเกลือครั้งที่ 1 54

4.3 แสดงผลการนับหยดน้ำเกลือครั้งที่ 2 54

4.4 แสดงผลการนับหยดน้ำเกลือครั้งที่ 3 55

4.5 แสดงผลการนับหยดน้ำเกลือครั้งที่ 4 55

4.6 แสดงผลการนับหยดน้ำเกลือครั้งที่ 5 56

4.7 แสดงผลการนับหยดน้ำเกลือครั้งที่ 6 56

4.8 แสดงผลการนับหยดน้ำเกลือครั้งที่ 7 57

4.9 แสดงผลการนับหยดน้ำเกลือครั้งที่ 8 57

4.10 แสดงผลการนับหยดน้ำเกลือครั้งที่ 9 58

4.11 แสดงผลการนับหยดน้ำเกลือครั้งที่ 10 58

4.12 แสดงค่าเฉลี่ยของการนับหยดน้ำเกลือในการทดลองย่อยที่ 1 59

4.13 แสดงผลการนับหยดน้ำเกลือครั้งที่ 1 60

4.14 แสดงผลการนับหยดน้ำเกลือครั้งที่ 2 60

4.15 แสดงผลการนับหยดน้ำเกลือครั้งที่ 3 61

4.16 แสดงผลการนับหยดน้ำเกลือครั้งที่ 4 61

4.17 แสดงผลการนับหยดน้ำเกลือครั้งที่ 5 62

4.18 แสดงผลการนับหยดน้ำเกลือครั้งที่ 6 62

4.19 แสดงผลการนับหยดน้ำเกลือครั้งที่ 7 63

4.20 แสดงผลการนับหยดน้ำเกลือครั้งที่ 8 63

สารบัญตาราง (ต่อ)

ตารางที่ หน้า

4.21 แสดงผลการนับหยดน้ำเกลือครั้งที่ 9 64
4.22 แสดงผลการนับหยดน้ำเกลือครั้งที่ 10 64
4.23 แสดงค่าเฉลี่ยของการนับหยดน้ำเกลือในการทดลองย่อยที่ 2 65
4.24 แสดงผลการนับหยดน้ำเกลือครั้งที่ 1 66
4.25 แสดงผลการนับหยดน้ำเกลือครั้งที่ 2 66
4.26 แสดงผลการนับหยดน้ำเกลือครั้งที่ 3 67
4.27 แสดงผลการนับหยดน้ำเกลือครั้งที่ 4 67
4.28 แสดงผลการนับหยดน้ำเกลือครั้งที่ 5 68
4.29 แสดงผลการนับหยดน้ำเกลือครั้งที่ 6 68
4.30 แสดงผลการนับหยดน้ำเกลือครั้งที่ 7 69
4.31 แสดงผลการนับหยดน้ำเกลือครั้งที่ 8 69
4.32 แสดงผลการนับหยดน้ำเกลือครั้งที่ 9 70
4.33 แสดงผลการนับหยดน้ำเกลือครั้งที่ 10 70
4.34 แสดงค่าเฉลี่ยของการนับหยดน้ำเกลือในการทดลองย่อยที่ 3 71
4.35 ผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูลบริเวณพื้นที่โล่ง 72
4.35 ผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูลบริเวณพื้นที่โล่ง (ต่อ) 73
4.36 ผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูลบริเวณพื้นที่ปิด 77

สารบัญรูป

รูปที่ หน้า

2.1 ส่วนประกอบของระบบสื่อสารข้อมูล 4
2.2 ตัวไอซี ATTINY2313 7
2.3 ตัวไอซี ATMEGA32 8
2.4 ชุดเซนเซอร์ R – REFLEX S5 9
2.5 แสดงการทำงานของ PHOTO REFLECTIVE SENSOR 9
2.6 โมดูล R - REFLEX S5 10
2.7 หน้าต่างโปรแกรม Visual Basic 6.0 11
2.8 หน้าต่างโปรแกรม Code vision AVR 11
2.9 การส่งข้อมูลแบบอนุกรม 12
2.10 การสื่อสารแบบอะซิงโครนัสที่ไม่ได้ใช้พารีตี้บิต 13
2.11 การสื่อสารแบบอะซิงโครนัสที่ใช้พารีตี้บิต 13
2.12 การสื่อสารแบบซิงโครนัส 14
2.13 การสื่อสารแบบซิงโครนัสที่อาจผิดพลาดได้ 14
2.14 ตัวอย่างการใช้อักขระซิง 2 ตัวในการสื่อสารแบบซิงโครนัส 14
2.15 การเปลี่ยนระดับแรงดันของลอจิกในการเชื่อมต่อข้อมูลแบบอนุกรม 15
2.16 ระบบตรวจเฝ้าระวังการให้หยดน้ำเกลือแบบไร้สาย 18
3.1 แผนภาพโดยรวมแสดงการเชื่อมต่อวงจรและอุปกรณ์ 19
3.2 แผนภาพโดยรวมของโครงงาน 20
3.3 วงจรของบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) 23
3.4 วงจรของบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) 26
3.5 วงจรของส่วนเชื่อมโยงข้อมูล 28
3.6 แผนภาพแสดงการเชื่อมต่อวงจรของตัวส่ง (Attiny2313) ที่มีรหัสประจำเครื่องคือ # 29
3.7 แผนภาพแสดงการเชื่อมต่อวงจรของตัวส่ง (Attiny2313) ที่มีรหัสประจำเครื่องคือ $ 30
3.8 แผนภาพแสดงการเชื่อมต่อวงจรของตัวรับ (Atmega 32) 31
3.9 รูปร่างหน้าต่างหลักของอุปกรณ์ 32
3.10 รูปร่างหน้าต่างตั้งค่าการเชื่อมต่อของอุปกรณ์ 33
3.11 รูปร่างหน้าต่างตั้งค่าเวลาของอุปกรณ์ 33
3.12 รูปร่างหน้าต่างตั้งค่าการทำงานของอุปกรณ์ 34
3.13 ตัวอย่างไฟล์ฐานข้อมูลในการจัดเก็บข้อมูลจากหน้าต่าง setting 37

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.14 รูปร่างหน้าต่างแสดงผลข้อมูล 38

3.15 ตัวอย่างหน้าต่างโปรแกรมไฮเปอร์เทอร์มินอล 40

3.16 แสดงการทดสอบการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ 41

3.17 แสดงการทดสอบการนับหยดน้ำเกลือ 42

3.18 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับบริเวณพื้นที่โล่ง 44

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งบริเวณพื้นที่โล่ง 44

3.19 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับบริเวณพื้นที่ปิด 44

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งบริเวณพื้นที่ปิด 44

4.1 ผลการทดสอบการส่งข้อมูลของไมโครคอนโทรลเลอร์ 46

4.2 แสดงการเชื่อมต่อภายในของเครื่องวัดหยดน้ำเกลือในส่วนของตัวรับข้อมูล 47

4.3 แสดงการเชื่อมต่อเครื่องวัดหยดน้ำเกลือในส่วนของตัวรับข้อมูล 47

4.4 แสดงการเชื่อมต่อภายในของเครื่องวัดหยดน้ำเกลือในส่วนของตัวส่งข้อมูล....48

4.5 แสดงการเชื่อมต่อเครื่องวัดหยดน้ำเกลือในส่วนของตัวส่งข้อมูล 48

4.6 หน้าต่างหลักของโปรแกรม 48

4.7 หน้าต่าง Communication....49

4.8 หน้าต่าง Date/Time....49

4.9 หน้าต่าง Channel Setting....50

4.10 หน้าต่าง Display 51

4.11 แสดงการปรับสายน้ำเกลือให้ใกล้เคียงกับอัตราการให้น้ำเกลือที่ต้องการ 51

4.12 หน้าต่าง Display ที่แสดงผลอัตราการให้น้ำเกลือที่มีปริมาตรคงเหลือ 100 % 52

4.13 หน้าต่าง Display ที่แสดงผลอัตราการให้น้ำเกลือที่มีปริมาตรคงเหลือ 9.55 % 52

4.14 หน้าต่าง Display ที่แสดงผลอัตราการให้น้ำเกลือที่มีปริมาตรคงเหลือ 0 % 53

4.15 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 10 เมตร 74

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 10 เมตร 74

(ค) แสดงผลการทดสอบที่ระยะ 10 เมตร 74

4.16 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 50 เมตร 74

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 50 เมตร 74

(ค) แสดงผลการทดสอบที่ระยะ 50 เมตร 74

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.17 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 110 เมตร....75

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 110 เมตร....75

(ค) แสดงผลการทดสอบที่ระยะ 110 เมตร....75

4.18 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 180 เมตร....75

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 180 เมตร....75

(ค) แสดงผลการทดสอบที่ระยะ 180 เมตร....75

4.19 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 290 เมตร....76

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 290 เมตร....76

(ค) แสดงผลการทดสอบที่ระยะ 290 เมตร....76

4.20 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 291 เมตร....76

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 291 เมตร....76

(ค) แสดงผลการทดสอบที่ระยะ 291 เมตร....76

4.21 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 300 เมตร....77

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 300 เมตร....77

(ค) แสดงผลการทดสอบที่ระยะ 300 เมตร....77

4.22 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 5 เมตร....78

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 5 เมตร....78

(ค) แสดงผลการทดสอบที่ระยะ 5 เมตร....78

4.23 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 10 เมตร....79

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 10 เมตร....79

(ค) แสดงผลการทดสอบที่ระยะ 10 เมตร....79

4.24 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 15 เมตร....79

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 15 เมตร....79

(ค) แสดงผลการทดสอบที่ระยะ 15 เมตร....79

4.25 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 16 เมตร....79

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 16 เมตร....79

(ค) แสดงผลการทดสอบที่ระยะ 16 เมตร....79

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.26 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 17 เมตร 80

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 17 เมตร 80

(ค) แสดงผลการทดสอบที่ระยะ 17 เมตร 80

4.27 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 20 เมตร 80

(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 20 เมตร 80

(ค) แสดงผลการทดสอบที่ระยะ 20 เมตร 80

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ปัจจุบันความก้าวหน้าทางด้านเทคโนโลยีการสื่อสารไร้สายมีการพัฒนาไปมาก การส่งข้อมูลข่าวสารได้มีการพัฒนาอย่างรวดเร็วทั้งด้านระยะทางที่ไกลมากๆ และจำนวนข้อมูลข่าวสารหน่วยงานหรือองค์กรต่างๆ ที่อยู่ห่างไกลต้องการระบบสื่อสารในการรับส่งข้อมูลข่าวสาร เช่น ระบบโทรศัพท์ ระบบโทรทัศน์และระบบเครือข่ายคอมพิวเตอร์ ซึ่งจะพบว่าไมโครคอนโทรลเลอร์และระบบสื่อสารไร้สายได้เข้ามามีบทบาทสำคัญในการควบคุมสั่งงานหรือประมวลผลการทำงานของระบบการสื่อสารต่างๆ

ปัจจุบันสำหรับอุปกรณ์ทางการแพทย์ที่ใช้ในการรักษาผู้ป่วย เช่น ในการให้น้ำเกลือ ถือว่าเป็นการรักษาอาการของผู้ป่วยในขั้นหนึ่ง ถ้าผู้ป่วยที่รับน้ำเกลือขณะนั้น ไม่ได้รับน้ำเกลืออย่างต่อเนื่อง เนื่องจากไม่มีพยาบาลมาเปลี่ยนให้ ซึ่งอาจจะทำให้ผู้ป่วยอาการหนักมากขึ้นได้ ดังนั้นจึงจำเป็นอย่างมากในทางการแพทย์ที่จะต้องคอยดูแลในเรื่องการให้น้ำเกลือกับผู้ป่วย ในส่วนของผู้ป่วยที่ได้รับการให้น้ำเกลือในขณะนั้นจึงต้องมีการวัดระดับน้ำเกลือเพื่อแจ้งเตือนระดับของน้ำเกลือที่ใกล้จะหมดของผู้ป่วยในแต่ละจุด ซึ่งในแต่ละจุดนั้นจะมีระยะทางที่ห่างไกลจะทำให้พยาบาลต้องเสียเวลาที่ต้องคอยเดินตรวจสอบระดับน้ำเกลือของผู้ป่วย ดังนั้นถ้านำเอาเทคโนโลยีระบบการติดต่อสื่อสารใหม่ๆ เข้ามาประยุกต์ใช้ในการควบคุมและตรวจจับการทำงานต่างๆ เพื่อตอบสนองความต้องการของผู้ใช้ที่จะก่อให้เกิดประสิทธิภาพในการทำงานมากขึ้น จากข้างต้นจึงเป็นการออกแบบและสร้างเครื่องต้นแบบสำหรับนำไปใช้ตรวจวัดระดับของน้ำเกลือแบบไร้สายเพื่อใช้แจ้งเตือนแก่พยาบาลว่าน้ำเกลือของผู้ป่วยใกล้จะหมด และให้นำน้ำเกลืออันใหม่มาเปลี่ยนให้กับผู้ป่วย ซึ่งจะนำไปใช้ในทางการแพทย์ช่วยประหยัดเวลาและให้พยาบาลทำงานส่วนอื่นได้อย่างเต็มที่

1.2 วัตถุประสงค์ของโครงงาน

1. สร้างและพัฒนาระบบแสดงอัตราการให้น้ำเกลือแบบไร้สาย แสดงระดับน้ำเกลือที่เหลืออยู่พร้อมทั้งระบบแจ้งเตือนเมื่อน้ำเกลือในขวดใกล้หมด

1.3 ผลที่คาดว่าจะได้รับ

1. ผู้สนใจสามารถนำอุปกรณ์ที่พัฒนาขึ้นนี้ไปใช้เพื่ออำนวยความสะดวกในทางการแพทย์ได้

2. ผู้ศึกษาสามารถเข้าใจการทำงานของอุปกรณ์ไมโครคอนโทรลเลอร์และอุปกรณ์ต่างๆ ที่ใช้ในการสนับสนุนได้

3. ช่วยเสริมประสิทธิภาพและอำนวยความสะดวกในการทำงานของเจ้าหน้าที่ทางการแพทย์

1.4 ขอบเขตการทำโครงงานวิศวกรรม

1. สามารถใช้ไมโครคอนโทรเลอร์ (AVR) ในการควบคุมการทำงานของบอร์ด

2. สามารถใช้ Wireless TRW 2.4 GHz ในการรับส่งข้อมูลบริเวณพื้นที่โล่งระยะไกลไม่เกิน 290 เมตรและรับส่งข้อมูลบริเวณพื้นที่ปิดระยะไกลไม่เกิน 16 เมตร

3. ใช้งานกับน้ำเกลือธรรมดา (Normal Saline)

4. ใช้เซนเซอร์ตรวจสอบการสะท้อนกลับของสัญญาณอินฟราเรด R – REFLEX S5 ในการตรวจวัดระดับการไหลของน้ำเกลือ

5. สามารถแสดงระดับน้ำเกลือได้ 2 ขวดพร้อมกัน บนหน้าจอคอมพิวเตอร์

1.5 ขั้นตอนการดำเนินงาน

ตารางที่ 1.1 ขั้นตอนการดำเนินงาน

เรื่อง	ปี 2557										
	ก.พ	มี.ค	เม.ย	พ.ค	มิ.ย	ก.ค	ส.ค	ก.ย	ต.ค	พ.ย	ธ.ค
ปรึกษาและวางแผน											
เลือกหัวข้อโครงงาน											
ศึกษาอุปกรณ์ไมโครคอนโทรเลอร์ตระกูล AVR											
ศึกษาการเขียนโปรแกรมภาษาซีและโปรแกรม Visual Basic 6.0											

ตารางที่ 1.1 ขั้นตอนการดำเนินงาน (ต่อ)

เรื่อง	ปี 2557										
	ก.พ	มี.ค	เม.ย	พ.ค	มิ.ย	ก.ค	ส.ค	ก.ย	ต.ค	พ.ย	ธ.ค
สร้างเครื่องวัดหยดน้ำเกลือและส่งข้อมูลแบบไร้สาย											
เขียนโปรแกรมและแก้ไข											
ทดสอบอุปกรณ์และปรับปรุงแก้ไข											
วิเคราะห์ผลการทดลอง											
สรุปการทดลอง											
จัดทำรูปเล่มรายงาน											

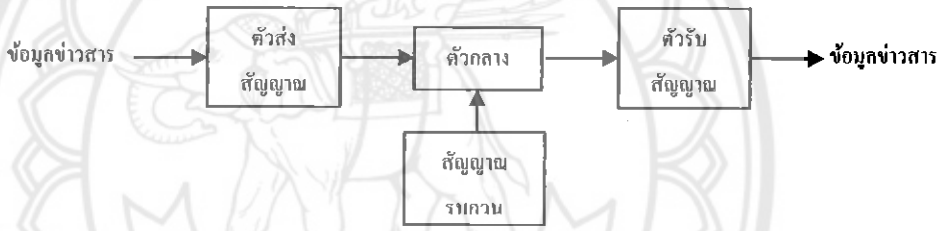
1.6 งบประมาณ

รายจ่าย	1. ค่าจัดทำเอกสาร	200	บาท
	2. ค่าวัสดุ อุปกรณ์สำนักงาน	1000	บาท
	3. ค่าจัดทำรูปเล่มโครงงาน	200	บาท
	4. อื่นๆ	600	บาท
	รวม	2,000	บาท
หมายเหตุ อนุญาตถัวเฉลี่ยทุกรายการ			
รายรับ ค่าดำเนินการโครงงานจากคณะ		**2,000**	**บาท**

บทที่ 2
หลักการและทฤษฎีเบื้องต้น

2.1 การสื่อสารไร้สาย [1]

การสื่อสารไร้สาย คือ การติดต่อสื่อสารโดยที่ไม่มีการใช้สายสัญญาณในการเชื่อมต่อกัน โดยปกติถ้าต้องการเชื่อมต่อเทอร์มินอล (Terminal) เข้ากับเบส (Base) หรือเชื่อมต่อเทอร์มินอลเข้ากับเทอร์มินอลจะต้องใช้สายสัญญาณในการเชื่อมต่อ แต่การสื่อสารแบบไร้สายจะทำการเชื่อมต่อโดยที่ไม่ใช้สายแต่จะใช้การสื่อสารแบบไร้สายแทนโดยจะใช้คลื่นวิทยุในการติดต่อสื่อสารผ่านตัวกลางต่างๆ ซึ่งองค์ประกอบของระบบสื่อสารไร้สายมีลักษณะเช่นเดียวกับระบบการสื่อสารพื้นฐานซึ่งมีอยู่ทั้งหมด 3 ส่วน คือ ตัวส่งสัญญาณ สื่อตัวกลางและตัวรับสัญญาณ ซึ่งในระบบอาจมีสัญญาณรบกวนเกิดขึ้นได้ดังแสดงในรูปที่ 2.1



รูปที่ 2.1 ส่วนประกอบของระบบสื่อสารข้อมูล

2.1.1 ความถี่ของคลื่น [2]

คลื่นวิทยุเป็นคลื่นความถี่ที่ถูกใช้ในระบบสื่อสารหลายๆ ประเภท ได้แก่ คลื่นวิทยุกระจายเสียง ซึ่งมีช่วงความถี่ตั้งแต่ 88 - 108 เมกะเฮิรตซ์ คลื่นวิทยุสมัครเล่นที่มีช่วงความถี่ 144 เมกะเฮิรตซ์ วิทยุสื่อสารที่ใช้กันในระบบราชการและคลื่นวิทยุที่ใช้ในระบบโทรศัพท์มือถือ ซึ่งมีอยู่หลายย่าน เช่น 800, 900, 1800 และ 1900 เมกะเฮิรตซ์ เป็นต้น ในสเปกตรัมของคลื่นความถี่วิทยุแบ่งออกเป็นหลายช่วง แต่ละช่วงมีลักษณะต่างกันไป ทำให้สามารถนำมาประยุกต์ใช้ให้เหมาะสมกับงานต่างๆ ตัวอย่างเช่น คลื่นที่อยู่ในช่วง 3 - 30 MHz เป็นคลื่นวิทยุที่มีคุณสมบัติสามารถเดินทางได้ในระยะไกลๆ จึงถูกนำมาใช้ในระบบวิทยุกระจายเสียงแบบ AM ทำให้สามารถส่งวิทยุข้ามจังหวัดได้ แต่สำหรับงานที่ไม่ต้องการให้คลื่นกระจายไปในระยะไกลมาก เช่น ระบบโทรศัพท์มือถือใช้คลื่นความถี่ในช่วง 300 - 3000 เมกะเฮิรตซ์ คลื่นย่านนี้เดินทางได้ในระยะใกล้จึงสามารถนำความถี่กลับมาใช้ซ้ำได้ในบริเวณใกล้เคียง ช่วยให้การใช้คลื่นความถี่ที่มีประสิทธิภาพมากขึ้นและจากตารางที่ 2.1 แสดงการแบ่งย่านความถี่เพื่อใช้งานในรูปแบบต่างๆ

ตารางที่ 2.1 แสดงการแบ่งความถี่ออกเป็นย่านต่างๆ

ย่านความถี่	ความถี่	ความยาวคลื่น
Very Low Frequency (VLF)	3 - 30 kHz	ยาวกว่า 10 Km
Low Frequency (LF)	30 - 300 kHz	10 - 1 Km
Medium Frequency (MF)	300 - 3000kHz	1000 - 100 m
High Frequency (HF)	3 - 30 MHz	100 - 10 m
Very High Frequency (VHF)	30 - 300 MHz	10 - 1 m
Ultra High Frequency (UHF)	300 - 3000 MHz	100 - 10 cm
Super High Frequency (SHF)	3 - 30 GHz	10 - 1 cm
Extreme High Frequency (EHF)	30 - 300 GHz	10 - 1 mm

Very Low Frequency (VLF) เป็นความถี่ย่าน 3 - 30 kHz ใช้ในวิทยุของเรือเดินสมุทร เนื่องจากเป็นคลื่นสั้นที่มีคุณสมบัติสามารถเดินทางได้ในระยะไกลๆ

Low Frequency (LF) and Medium Frequency (MF) เป็นคลื่นความถี่ย่าน 30 - 3000 MHz ใช้ในวิทยุ AM มีคุณสมบัติเดินทางได้ในระยะไกลมากๆ สามารถข้ามทวีปโดยไม่ต้องอาศัยสถานีทวนสัญญาณ

High Frequency (HF) เป็นคลื่นความถี่ระหว่าง 3 - 30 MHz เรียกว่า เป็นวิทยุคลื่นสั้นใช้ในระบบวิทยุกระจายเสียงแบบ AM และระบบวิทยุสมัครเล่น

Very High Frequency (VHF) เป็นคลื่นความถี่ระหว่าง 30 - 300 MHz ใช้ในระบบวิทยุแบบ FM และการกระจายสัญญาณโทรทัศน์ในช่อง 3, 5, 7, 9 และ 11

Ultra High Frequency (UHF) เป็นคลื่นความถี่ระหว่าง 300 - 3000 MHz ใช้ในโทรศัพท์มือถือ การกระจายสัญญาณของสถานี iTV และระบบ Wireless LAN

Super High Frequency (SHF) เป็นคลื่นความถี่ระหว่าง 3 - 30 GHz ใช้ในระบบสื่อสารผ่านดาวเทียม และการรับสัญญาณโทรทัศน์ของ UBC จากดาวเทียมไทยคม

Extreme High Frequency (EHF) เป็นคลื่นความถี่ระหว่าง 30 - 300 GHz ใช้ในระบบสื่อสารผ่าน ดาวเทียมและระบบเรดาร์ค้นหาเครื่องบิน

ก. ย่านความถี่ ISM

ISM Band (Industrial Scientific and Medical Band) ความถี่วิทยุเป็นทรัพยากรธรรมชาติที่มีค่ามาก ผู้ใดมีเครื่องรับ-ส่งสัญญาณสามารถนำคลื่นความถี่นี้ไปใช้ได้ ไม่ว่าจะใช้ในการสื่อสารภาพ เสียง หรือข้อมูล ดังนั้นทำให้มีผู้ต้องการนำความถี่ไปใช้งานเป็นจำนวนมาก แต่ถ้าไม่มีผู้จัดการความถี่แล้วจะมีปัญหาการรบกวนกันเกิดขึ้น ซึ่งผู้ที่มีหน้าที่จัดสรรความถี่ใน

ประเทศไทย คือ คณะกรรมการกิจการกระจายเสียงและกิจการโทรทัศน์แห่งชาติ (กสช.) มีหน้าที่วางแผนและจัดสรรความถี่ให้มีประสิทธิภาพสูงสุด และมีหน้าที่ตรวจสอบและเฝ้าระวังเพื่อไม่ให้เกิดการรบกวนกัน นั่นคือ ทำให้เกิดข้อกำหนดต่างๆ เช่น กำลังส่งที่ใช้ได้ ความแรงของสัญญาณ ความถี่ข้างเคียงของเครื่องส่งที่ผลิตออกมาได้ แบนด์วิธของเครื่องส่ง อัตราขยายของสายอากาศ (Antenna gain) ที่ใช้ได้ในแต่ละความถี่ ความถี่ที่จัดสรรส่วนใหญ่จะเป็นเพื่อการติดต่อสื่อสารทางราชการ หรือความถี่ในงานสัมปทานต่างๆ แต่มีความถี่ย่านหนึ่งที่คณะกรรมการกิจการกระจายเสียงและกิจการโทรทัศน์แห่งชาติ (กสช.) จัดสรรออกมาโดยอ้างอิงความถี่ในช่วง ISM Band เป็นความถี่เสรีที่ใช้กันทั่วโลกโดยไม่ต้องขออนุญาต มีทั้งหมด 3 ย่านคือ ย่าน 900 เมกะเฮิรตซ์ ย่าน 2.4 กิกะเฮิรตซ์และย่าน 5 กิกะเฮิรตซ์ แต่ในเมืองไทยนั้นสามารถใช้งานได้ย่านเดียวคือ 2.4 กิกะเฮิรตซ์ นั่นคือ ทำให้เครือข่ายไวร์เลสแลนมาตรฐาน 802.11a ที่ความถี่ย่าน 5 กิกะเฮิรตซ์ไม่สามารถใช้ได้ในประเทศไทย

ข. ย่านความถี่บลูทูธ กับ wifi

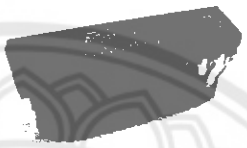
ย่านความถี่บลูทูธใช้สัญญาณวิทยุความถี่สูง 2.4 GHz. แต่จะแยกย่อยออกไปตามแต่ละประเทศ ตัวอย่างเช่น ในแถบยุโรปและอเมริกา ใช้ช่วง 2.400 ถึง 2.4835 GHz แบ่งออกเป็น 79 ช่องสัญญาณ และใช้ช่องสัญญาณที่แบ่งนี้เพื่อส่งข้อมูลสลับช่องไปมา 1600 ครั้งต่อ 1 วินาที ส่วนที่ญี่ปุ่นจะใช้ความถี่ 2.402 ถึง 2.480 GHz. แบ่งออกเป็น 23 ช่อง ระยะการทำงานของบลูทูธอยู่ที่ 5-10 เมตร โดยหลักการของบลูทูธจะถูกออกแบบมาเพื่อใช้กับอุปกรณ์ที่มีขนาดเล็ก เนื่องจากใช้การขนส่งข้อมูลในจำนวนที่ไม่มาก นอกจากนี้ยังใช้พลังงานต่ำ กินไฟน้อย และสามารถใช้งานได้นานโดยไม่ต้องนำไปชาร์จไฟบ่อยๆ

ส่วนความสามารถการส่งถ่ายข้อมูลของบลูทูธอยู่ที่ 1 Mbps และคงจะไม่มีปัญหาอะไรมากกับขนาดของไฟล์ที่ใช้กันบนโทรศัพท์มือถือหรือการใช้งานแบบทั่วไป ซึ่งถือว่าเหลือเฟือมาก แต่ถ้าเป็นข้อมูลที่มีขนาดใหญ่จะช้าเกินไป และถ้าถูกนำไปเปรียบกับ Wireless LAN (WLAN) แล้วความสามารถของบลูทูธคงจะห่างชั้นกันเยอะ ซึ่งในส่วนของ WLAN มีระยะการรับ-ส่งที่ไกลกว่า

ส่วนในการใช้งานในโครงงานนี้ใช้เป็นโมดูล Wireless TRW 2.4GHz ที่สัญญาณวิทยุความถี่ 2.4 GHz ซึ่งใช้ในช่วงความถี่ 2.4 – 2.527 GHz แบ่งออกเป็น 128 ช่อง ระยะการทำงานอยู่ในช่วง 280 เมตร มีความเร็วในการส่งข้อมูล 250 Kbps และ 150 เมตร มีความเร็วในการส่งข้อมูล 1 Mbps และสามารถทำงานได้ที่ 3 โวลต์

2.2 ไมโครคอนโทรลเลอร์ ATTINY2313 [4]

โครงงานนี้ใช้ไมโครคอนโทรลเลอร์ ATTINY2313 เป็นไมโครคอนโทรลเลอร์ในการควบคุมการทำงานของบอร์ดตัวส่ง มีสถาปัตยกรรมแบบ RISC (Reduce Instruction Set Computer) ซึ่งมีความเร็วในการประมวลผล 1 ล้านคำสั่งต่อ 1 วินาทีต่อ 1 MHz (1 MIPS/MHz) ซึ่งบอร์ดที่นำมาทดลองใช้คริสตัล 11.0592 MHz ดังนั้น จึงทำให้ไมโครคอนโทรลเลอร์มีความเร็วในการประมวลผล 11 ล้านคำสั่งต่อ 1 วินาที (11 MIPS)



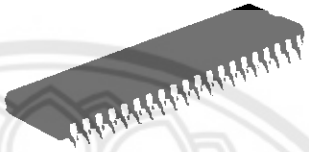
รูปที่ 2.2 ตัวไอซี ATTINY2313

2.2.1 คุณสมบัติของไมโครคอนโทรลเลอร์ ATTINY2313

1. หน่วยความจำแบบแฟลช (FLASH) สำหรับบันทึก Program memory ขนาด 2 กิโลไบต์
2. หน่วยความจำแบบอีอีพรอม (EEPROM) สำหรับบันทึก Data memory ขนาด 128 ไบต์
3. หน่วยความจำแบบแรม (SRAM) ขนาด 128 ไบต์
4. ระบบการตรวจจับระดับสัญญาณอนาล็อก (Analog Comparator)
5. ระบบการกำเนิดอินเตอร์รัพต์จากภายนอก (External Interrupt) และระบบการกำเนิดอินเตอร์รัพต์จากภายใน (Internal Interrupt)
6. การเชื่อมต่อแบบ USI : Universal Serial Interface
7. ระบบการสื่อสารข้อมูลดิจิตอลแบบซิงโครนัส (SPI)
8. ระบบการเปลี่ยนสัญญาณอนาลอกเป็นดิจิตอล ADC (Analog to Digital Converter)
9. ความเร็วในการทำงาน 1 คำสั่งต่อ 1 สัญญาณนาฬิกา
10. วงจรนับ/จับเวลา (COUNTER/TIMER)
11. ระบบตรวจจับการทำงานผิดพลาดของไมโครคอนโทรเลอร์ (Watchdog Timer)
12. ระบบการกำเนิดสัญญาณแบบ PWM 1 ช่อง
13. แรงดันใช้งาน: 4.5 - 5.5 โวลต์

2.3 ไมโครคอนโทรลเลอร์ ATMEGA32 [5]

โครงงานนี้ใช้ไมโครคอนโทรลเลอร์ Atmega32 เป็นไมโครคอนโทรลเลอร์ในการควบคุมการทำงานของบอร์ดตัวรับ ซึ่งเป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต มีสถาปัตยกรรมแบบ RISC (Reduce Instruction Set Computer) ซึ่งมีความเร็วในการประมวลผล 1 ล้านคำสั่งต่อ 1 วินาทีต่อ 1 MHz (1 MIPS/MHz) ซึ่งบอร์ดที่นำมาทดลองใช้คริสตัล 16 MHz ดังนั้นจึงทำให้ไมโครคอนโทรลเลอร์มีความเร็วในการประมวลผล 16 ล้านคำสั่งต่อ 1 วินาที (16 MIPS)



รูปที่ 2.3 ตัวไอซี ATMEGA32

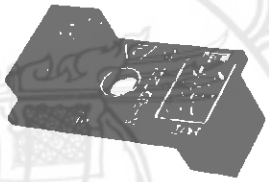
2.3.1 คุณสมบัติของไมโครคอนโทรลเลอร์ ATMEGA32

1. พอร์ตอินพุต เอาท์พุตแบบดิจิตอล 4 พอร์ต พอร์ตละ 8 บิต
2. หน่วยความจำแบบแฟลช (FLASH) สำหรับบันทึก Program memory ขนาด 32 กิโลไบต์
3. หน่วยความจำแบบอีอีพรอม (EEPROM) สำหรับบันทึก Data memory ขนาด 1 กิโลไบต์
4. หน่วยความจำแบบแรม (RAM) ขนาด 2 กิโลไบต์
5. ระบบการเปลี่ยนสัญญาณอนาลอกเป็นดิจิตอล ADC (Analog to Digital Converter) ขนาด 10 บิต จำนวน 8 ช่อง
6. คำสั่งในการควบคุมการทำงานของไมโครคอนโทรเลอร์ จำนวน 131 คำสั่ง
7. ความเร็วในการทำงาน 1 คำสั่งต่อ 1 สัญญาณนาฬิกา
8. วงจรนับ/จับเวลา (COUNTER/TIMER) ขนาด 16 บิต 1 ตัว
9. วงจรนับ/จับเวลา (COUNTER/TIMER) ขนาด 8 บิต 2 ตัว
10. ระบบการสื่อสารข้อมูลดิจิตอลแบบอะซิงโครนัส (UART) 1 ช่อง
11. ระบบการสื่อสารข้อมูลดิจิตอลแบบซิงโครนัส (SPI) 1 ช่อง
12. ระบบตรวจจับการทำงานผิดพลาดของไมโครคอนโทรเลอร์ (Watchdog Timer)
13. ระบบการกำเนิดสัญญาณแบบ PWM 1 ช่อง
14. ระบบการกำเนิดอินเตอร์รัพต์จากภายนอก (External Interrupt)
15. แรงดันใช้งาน: 4.5 - 5.5 โวลต์

ทั้งนี้การเลือกใช้ไมโครคอนโทรลเลอร์ ATTINY2313 เป็นตัวส่ง เนื่องจากมีขนาดเล็กทำให้สะดวกต่อการติดตั้ง มีหน่วยความจำที่เหมาะสมกับข้อมูลที่ต้องการส่งและมีราคาถูกในด้านการเพิ่มจำนวนของบอร์ดตัวส่ง ในส่วนของการเลือกใช้ไมโครคอนโทรลเลอร์ Atmega32 เป็นตัวรับ เนื่องจากมีขนาดของหน่วยความจำที่เพียงพอต่อความต้องการ สามารถรองรับการใช้งานในด้านของการเพิ่มจำนวนของบอร์ดตัวส่ง ซึ่งทั้งไมโครคอนโทรลเลอร์ ATTINY2313 และไมโครคอนโทรลเลอร์ Atmega32 อยู่ในตระกูลเดียวกัน ทำให้สามารถใช้ชุดคำสั่งที่เหมือนกันได้

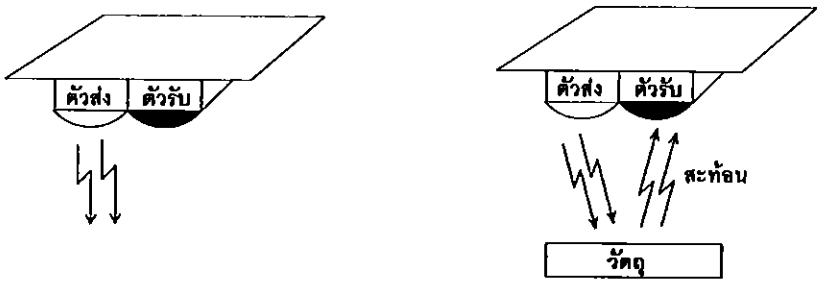
2.4 ชุดเซนเซอร์ตรวจสอบการสะท้อนกลับของสัญญาณอินฟราเรด R – REFLEX S5

โครงงานนี้ใช้วงจรชุดเซนเซอร์ตรวจสอบการสะท้อนกลับของสัญญาณอินฟราเรด R - REFLEX S5 ที่เชื่อมต่อกับบอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313)



รูปที่ 2.4 ชุดเซนเซอร์ R – REFLEX S5

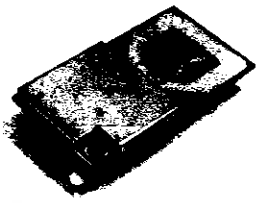
การทำงานของ R - REFLEX S5 [6] จะอาศัยการทำงานของ PHOTO REFLECTIVE SENSOR เซนเซอร์ตัวนี้จะประกอบด้วย ตัวส่งและตัวรับสัญญาณอินฟราเรด ตัวส่งมีลักษณะโครงสร้างเป็นแอลอีดีแบบอินฟราเรดและตัวรับมีลักษณะโครงสร้างเป็นแบบทรานซิสเตอร์ โดยที่ตัวส่งจะทำการส่งสัญญาณอินฟราเรดออกไปตลอดเวลา หากสัญญาณดังกล่าวไปกระทบกับวัตถุ สัญญาณก็จะสะท้อนกลับมาที่ตัวรับ ดังรูปที่ 2.5



รูปที่ 2.5 แสดงการทำงานของ PHOTO REFLECTIVE SENSOR

2.5 Wireless TRW 2.4 GHz

โครงงานนี้ใช้โมดูล TRW 2.4 GHz [7] ในการรับ – ส่งข้อมูลระหว่างบอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) และบอร์ดไมโครคอนโทรเลอร์ตัวรับ (Atmega32)

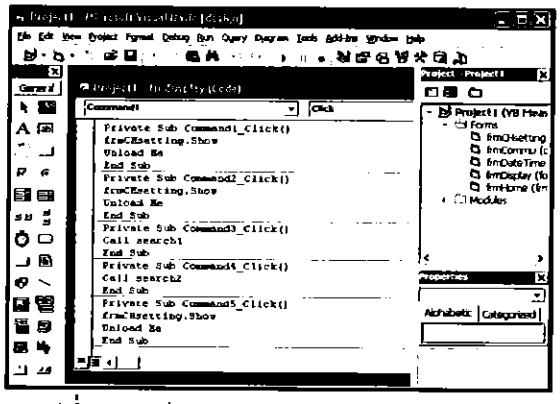


รูปที่ 2.6 โมดูล TRW 2.4 GHz

โมดูล TRW 2.4 GHz เป็นโมดูลสำเร็จรูปที่ใช้ในการรับ - ส่ง ข้อมูลในแบบอนุกรมโดยใช้กับความถี่ 2.4 GHz พร้อมมีเสาอากาศที่ปรับแต่งสำเร็จรูปในตัว ใช้งานได้ในระยะไกล 280 เมตร (ความเร็วข้อมูล 250 kbps) ระยะ 150 เมตร (ความเร็ว 1Mbps) ในพื้นที่โล่งแจ้ง สามารถรับส่งข้อมูลกันได้ในระยะทางที่ไกลกว่า RS232 หลายเท่าตัว ไม่จำเป็นต้องใช้สายสัญญาณที่เป็นตัวนำสัญญาณทางไฟฟ้าในการสื่อสารข้อมูลกัน ทำให้สามารถเปลี่ยนแปลงหรือเคลื่อนย้ายจุดรับส่งข้อมูลได้ตลอดเวลา ซึ่งถ้าเป็นการรับส่งข้อมูลด้วยระบบ RS232 แบบที่ใช้สายสัญญาณนั้นจะเกิดความยุ่งยากในการติดตั้งสายสัญญาณเป็นอย่างมาก แต่อย่างไรก็ตามการรับส่งข้อมูลโดยใช้อากาศเป็นตัวกลางในการสื่อสารนั้น ก็มีข้อจำกัดบางประการเหมือนกัน โดยเฉพาะอย่างยิ่งเรื่องความน่าเชื่อถือของข้อมูลที่รับส่งกัน ซึ่งมีโอกาสผิดพลาดหรือสูญหายได้เหมือนกัน เนื่องจากในการลำเลียงข้อมูลนั้นไม่ได้ใช้สายสัญญาณเป็นตัวกลางในการรับส่งข้อมูล แต่ใช้อากาศเป็นตัวกลางในการรับส่งข้อมูลแทน ซึ่งมีโอกาสที่ข้อมูลจะเกิดการรบกวนจากสัญญาณอื่นๆ ที่มีย่านความถี่ใกล้เคียงกันแล้วทำให้ข้อมูลผิดเพี้ยนไป

2.6 โปรแกรม Visual Basic 6.0

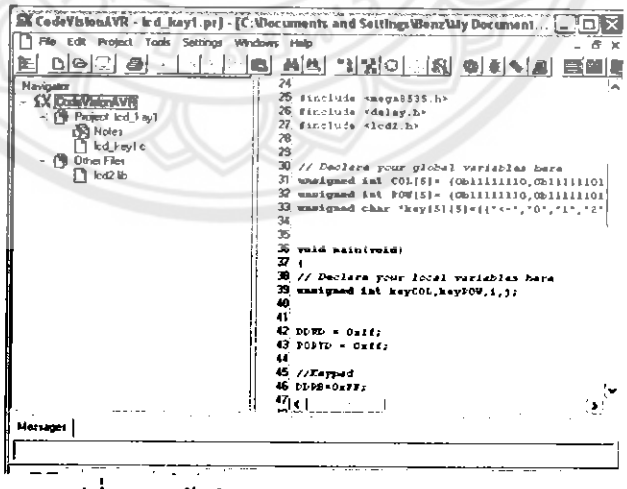
โครงงานนี้ใช้โปรแกรม Visual Basic 6.0 ซึ่งเป็นเวอร์ชันเบสิคที่ง่ายต่อการเรียนรู้และเข้าใจสำหรับผู้เริ่มต้นและเนื่องจากโครงงานนี้มีการใช้งานการแสดงผลที่ไม่ซับซ้อนมาก โปรแกรม Visual Basic 6.0 จึงเหมาะสมกับการใช้งานในโครงงานนี้



รูปที่ 2.7 หน้าต่างโปรแกรม Visual Basic 6.0

โปรแกรมVisual Basic 6.0 [8] เป็นโปรแกรมที่ได้เปลี่ยนรูปแบบการเขียนโปรแกรมใหม่ โดยมีชุดคำสั่งมาสนับสนุนการทำงานมีเครื่องมือต่าง ๆ ที่เรียกกันว่า คอนโทรล (Controls) ไว้สำหรับช่วยในการออกแบบโปรแกรม โดยเน้นการออกแบบหน้าจอแบบกราฟิกหรือที่เรียกว่า Graphic User Interface (GUI) ทำให้การจัดรูปแบบหน้าจอเป็นไปได้ง่าย และในการเขียนโปรแกรมนั้นจะเขียนแบบ Event - Driven Programming คือ โปรแกรมจะทำงานก็ต่อเมื่อเหตุการณ์ (Event) เกิดขึ้นตัวอย่างของเหตุการณ์ ได้แก่ ผู้ใช้เลื่อนเมาส์ ผู้ใช้กดปุ่มบนคีย์บอร์ด ผู้ใช้กดปุ่มเมาส์ เป็นต้น

2.7 โปรแกรมควบคุมการทำงานของไมโครคอนโทรเลอร์



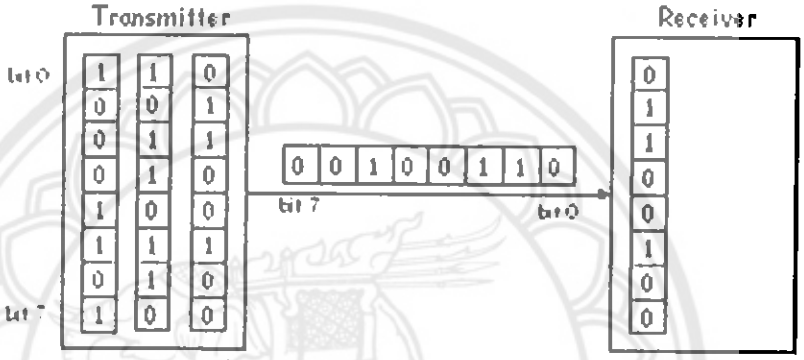
รูปที่ 2.8 หน้าต่างโปรแกรม Code vision AVR

สำหรับการเขียนโปรแกรมเพื่อควบคุมการทำงานของไมโครคอนโทรเลอร์ Atmega32 และไมโครคอนโทรเลอร์ Attiny2313 ได้เลือกใช้โปรแกรม Code Vision AVR โดยในส่วนของการเขียนโปรแกรมนั้นใช้ภาษาซี ในการเขียนซึ่งโปรแกรม Code Vision AVR นี้มีข้อดีคือ เป็น

โปรแกรมที่สามารถ compiler ได้ทั้งภาษาแอสเซมบลีและภาษาซี อีกทั้งยังมี wizard เพื่อช่วยในการเขียนโปรแกรมได้ง่ายและรวดเร็วขึ้น

2.8 การส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม

การส่งผ่านข้อมูลแบบอนุกรม (Serial Transmission) รูปแบบการส่งผ่านข้อมูล [13] ในลักษณะนี้ทุกบิตที่เข้ารหัสแทนข้อมูลหนึ่งตัวอักษรจะถูกส่งผ่านไปตามสายส่งเรียงลำดับกันไปทีละบิตในสายส่งเพียงเส้นเดียว ดังรูปที่ 2.9



รูปที่ 2.9 การส่งข้อมูลแบบอนุกรม [14]

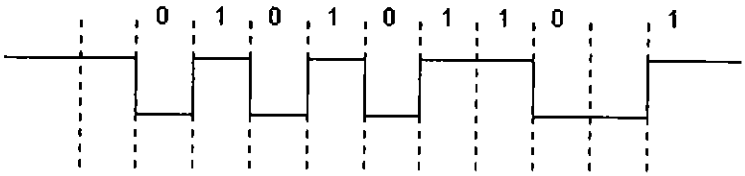
จากรูปที่ 2.9 ตัวอักษรจะประกอบด้วย 8 บิต เรียงเป็นลำดับ ข้อมูลจะถูกส่งออกมาทีละบิตระหว่างต้นทางและปลายทาง ในส่วนของปลายทางจะรวบรวมบิตเหล่านี้ทีละบิตจนครบ 8 บิต เป็น 1 ตัวอักษร จะเห็นว่าการส่งข้อมูลแบบนี้จะช้ากว่าแบบขนาน แต่ค่าใช้จ่ายจะถูกกว่าแบบขนาน ซึ่งเหมาะสำหรับการส่งระยะทางไกลๆ

โดยทั่วไปแล้วการส่งข้อมูลนั้นจะประกอบไปด้วยกลุ่มของตัวอักษร ดังนั้นในการส่งข้อมูลแบบอนุกรมนี้จึงเกิดปัญหาขึ้นว่า แล้วต้นทางและปลายทางจะทราบได้อย่างไรว่าจะแบ่งแต่ละตัวอักษรตรงบิตใด จึงเกิดวิธีการสื่อสารข้อมูลขึ้น 2 แบบ คือ การสื่อสารแบบอะซิงโครนัส (Asynchronous Transmission) และการสื่อสารแบบซิงโครนัส (Synchronous Transmission)

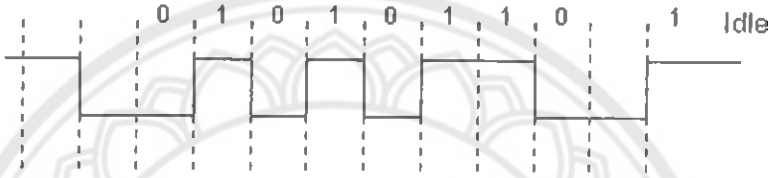
2.8.1. การสื่อสารแบบอะซิงโครนัส (Asynchronous Transmission)

UART ย่อมาจากคำว่า Universal Asynchronous Receiver Transmitter หมายถึงอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส (Asynchronous) การสื่อสารแบบอะซิงโครนัสหรือเรียกอีกอย่างหนึ่งว่าเป็นการสื่อสารแบบระบุจุดเริ่มต้นและจุดสิ้นสุด (Start - Stop Transmission) ลักษณะของสัญญาณที่ใช้ในการติดต่อสื่อสารกันจะประกอบไปด้วย บิตเริ่มต้น (start bit) บิตของข้อมูลที่สื่อสาร (transmission data) จำนวน 8 บิต บิตตรวจข้อผิดพลาด (parity

bit) และบิตสิ้นสุด (stop bit) สำหรับบิตตรวจสอบข้อผิดพลาดจะใช้หรือไม่ใช้ก็ได้ ดังนั้นสัญญาณจึงต้องประกอบด้วยส่วนประกอบอย่างน้อย 3 ส่วน ดังรูปที่ 2.10 และรูปที่ 2.11



รูปที่ 2.10 การสื่อสารแบบอะซิงโครนัสที่ไม่ได้ใช้พารีตี้บิต



รูปที่ 2.11 การสื่อสารแบบอะซิงโครนัสที่ใช้พารีตี้บิต

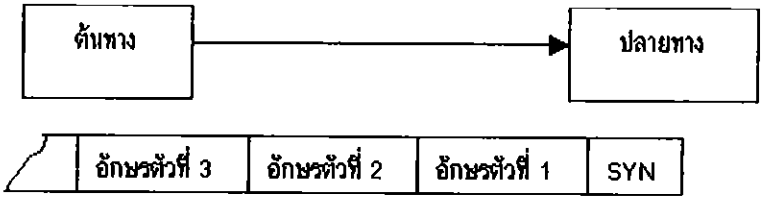
จากรูปที่ 2.10 และรูปที่ 2.11 ในขณะที่ไม่มีข้อมูลส่งออกมาสถานะของการส่งจะเป็นแบบว่าง (Idle) ซึ่งจะมีระดับของสัญญาณเป็น 1 ตลอดเวลา เพื่อความแน่ใจว่าปลายทางหรือฝ่ายรับยังคงติดต่อกับต้นทางหรือฝ่ายส่งอยู่ เมื่อเริ่มจะส่งข้อมูลสัญญาณของอะซิงโครนัสจะเป็น 0 หนึ่งช่วงสัญญาณนาฬิกา ซึ่งบิตนี้เรียกว่า บิตเริ่มต้น ตามหลังของบิตเริ่มต้นจะเป็นบิตข้อมูลสำหรับ 1 ตัวอักษร ตามหลังบิตข้อมูลก็จะเป็นบิตตรวจข้อผิดพลาดแล้วจะตามด้วยบิตสิ้นสุด ถ้าไม่ใช่บิตตรวจข้อผิดพลาด ตามหลังบิตข้อมูลก็จะเป็นบิตสิ้นสุดเลย หลังจากนั้นถ้าไม่มีข้อมูลส่งออกมาสัญญาณจะกลับไปอยู่ที่สถานะแบบว่างอีกเพื่อรอการส่งข้อมูลต่อไป

การสื่อสารแบบอะซิงโครนัสนี้ มีลักษณะเป็นไปทีละตัวอักษร และสัญญาณที่ส่งออกมา มีบางส่วนเป็นบิตเริ่มต้น บิตสิ้นสุด และบิตตรวจข้อผิดพลาด ทำให้ความเร็วในการส่งข้อมูลต่อวินาทีน้อยลงไป เนื่องจากต้องสูญเสียช่องทางการสื่อสารให้กับบิตเริ่มต้น บิตสิ้นสุด และบิตตรวจข้อผิดพลาด (ถ้ามีใช้) ตลอดเวลา การสื่อสารแบบอะซิงโครนัสนี้มักใช้ในการติดต่อระหว่างคอมพิวเตอร์กับอุปกรณ์รอบข้าง

2.8.2. การสื่อสารแบบซิงโครนัส (Synchronous Transmission)

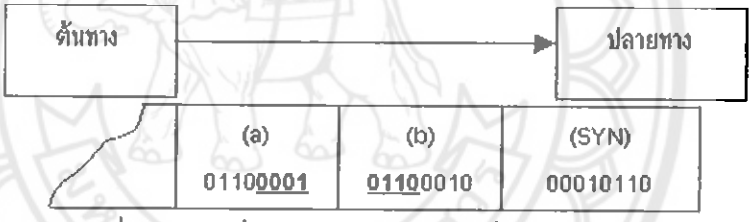
การสื่อสารแบบซิงโครนัส จะทำการจัดกลุ่มของข้อมูลเป็นกลุ่มๆ และทำการส่งข้อมูลทั้งกลุ่มไปพร้อมกันในทีเดียว เรียกกลุ่มของข้อมูลนี้ว่า บล็อกของข้อมูล (Block of Data) ซึ่งตัวอักษรตัวแรกและตัวถัดไปที่อยู่ในบล็อกเดียวกันจะไม่มีอะไรมาคั่นเหมือนอย่างแบบอะซิงโครนัส ที่ต้องใช้บิตเริ่มต้นและบิตสิ้นสุดคั่นทุกๆ ตัวอักษร แต่จะมีข้อมูลเริ่มต้นซึ่งเป็น

ลักษณะของบิตพิเศษที่ส่งมาเพื่อให้รู้ว่านั้นคือ จุดเริ่มต้นของกลุ่มตัวอักษรที่กำลังส่งเรียงกันเข้ามา เช่น อักขระซิง (SYN character) โดยที่อักขระซิงมีรูปแบบบิต คือ 00010110 ตัวอย่างของการส่งแสดงได้ดังรูปที่ 2.12



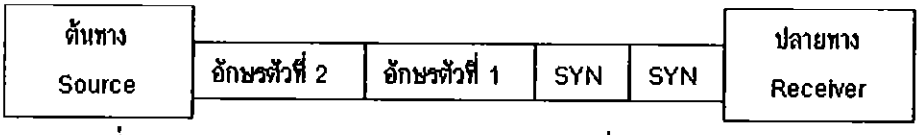
รูปที่ 2.12 การสื่อสารแบบซิงโครนัส

เมื่อปลายทางตรวจพบอักขระซิง หรือ 00010110 แล้วจะทราบได้ทันทีว่าบิตที่ตามมาคือบิตตัวอักษรแต่ละตัว แต่การใช้อักขระซิงเพียงตัวเดียวอาจเกิดข้อผิดพลาดได้ เช่น ถ้าเราส่งตัวอักษร b และตัวอักษร a ติดต่อกันไป ซึ่งตัวอักษร b มีรูปแบบบิตคือ 01100010 และตัวอักษร a มีรูปแบบบิตคือ 01100001 การส่งจะแสดงได้ดังรูปที่ 2.13



รูปที่ 2.13 การสื่อสารแบบซิงโครนัสที่อาจผิดพลาดได้

จะเห็นว่าเครื่องปลายทางจะตรวจพบอักขระซิงระหว่างบิตของตัวอักษร b และตัวอักษร a ทำให้เข้าใจว่าบิตต่อไปจะเป็นบิตของกลุ่มข้อมูล ซึ่งจะทำให้การรับข้อมูลนั้นเกิดผิดพลาดขึ้นได้ ดังนั้นจึงแก้ปัญหาด้วยการใช้อักขระซิง 2 ตัวต่อกันเป็นลักษณะของบิตพิเศษที่บอกให้ทราบว่าเป็นจุดเริ่มต้นบิตของกลุ่มข้อมูล ตัวอย่างของการใช้อักขระซิง 2 ตัวในการสื่อสารแบบซิงโครนัส และการตัดแถวของบิตข้อมูลออกเป็นกลุ่มทีละ 8 บิต เพื่อแทนข้อมูลแสดงได้ดังรูปที่ 2.14



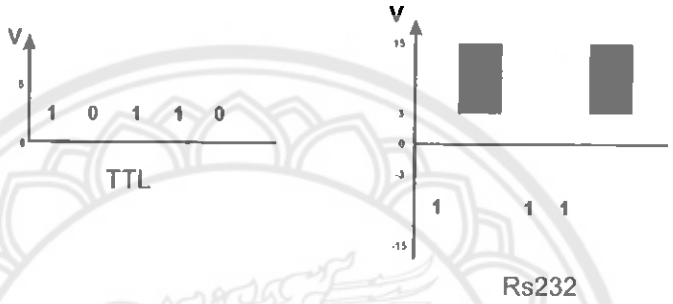
รูปที่ 2.14 ตัวอย่างการใช้อักขระซิง 2 ตัวในการสื่อสารแบบซิงโครนัส

2.8.3 RS 232 (Recommended Standard 232) [15]

RS 232 คือ มาตรฐานการเชื่อมต่อข้อมูลแบบอนุกรม ใช้เพื่อเพิ่มระยะทางในการส่งข้อมูล แบบอนุกรม ให้สามารถส่งได้ระยะทางที่มากขึ้น โดยมีการเปลี่ยนระดับแรงดันของลอจิกจากเดิมที่จะอยู่ในช่วง 0 - 5 V หรือ 0-3.3 V เป็นช่วง - 15 ถึง 15 V โดยมีรายละเอียดดังนี้

ลอจิก 0 ของ RS 232 จะอยู่ในช่วง 3 ถึง 15V

ลอจิก 1 ของ RS 232 จะอยู่ในช่วง - 3 ถึง - 15V



รูปที่ 2.15 การเปลี่ยนระดับแรงดันของลอจิกในการเชื่อมต่อข้อมูลแบบอนุกรม

2.9 น้ำเกลือ [9]

ชนิดของน้ำเกลือ ที่ใช้บ่อย ได้แก่

1. นอร์มัลซาไลน์ (Normal saline solution/NSS) หมายถึง น้ำเกลือธรรมดาที่มีความเข้มข้น 0.9 % ซึ่งเท่ากับความเข้มข้นของเกลือในกระแสเลือดของคนปกติ มีขนาด 500 มิลลิลิตรและ 1,000 มิลลิลิตร

2. 5 % เดกซ์โทรส (5 % Dextrose in water หรือ 5% D/W) หมายถึง น้ำตาลเดกซ์โทรสที่มีความเข้มข้น 5 % ไม่มีเกลือแร่ผสม มีขนาด 500 มิลลิลิตร และ 1,000 มิลลิลิตรโดยใช้ในภาวะน้ำตาลในเลือดต่ำหรืออดอาหาร สำหรับเด็ก 100 มิลลิลิตร ต่อน้ำหนักตัว 1 กิโลกรัม ใน 24 ชั่วโมง และผู้ใหญ่ 1,000 - 2,000 มิลลิลิตร ใน 12 - 24 ชั่วโมง

3. 5 % เดกซ์โทรสในนอร์มัลซาไลน์ (5 % Dextrose in NSS หรือ 5 % D/NSS) หมายถึง น้ำตาลเดกซ์โทรสเข้มข้น 5 % ผสมกับน้ำเกลือธรรมดา โดยใช้สำหรับเด็กที่มีภาวะขาดน้ำ, ช็อก, ไข้เลือดออก, อดอาหาร หรือมีภาวะน้ำตาลในเลือดต่ำโดยใช้ขนาด 100 มิลลิลิตรต่อน้ำหนักตัว 1 กิโลกรัม ใน 24 ชั่วโมง ถ้ามีภาวะขาดน้ำรุนแรง หรือช็อกใน 2 ชั่วโมง แรก ควรให้ขนาด 20 - 40 มิลลิลิตร ต่อน้ำหนักตัว 1 กิโลกรัม

4. 5 % เดกซ์โทรสใน 1/3 นอร์มัลซาไลน์ (5 % Dextrose in 1/3 NSS) หมายถึง น้ำตาลเดกซ์โทรสเข้มข้น 5 % ผสมกับน้ำเกลือที่มีความเข้มข้น 0.3 % (เข้มข้นเพียง 1/3 ของน้ำเกลือธรรมดา) มีขนาด 500 มิลลิลิตรและ 1,000 มิลลิลิตร ใช้สำหรับผู้ป่วยที่มีภาวะขาดน้ำ ช็อก อด

อาหาร หรือมีภาวะน้ำตาลในเลือดต่ำ สำหรับผู้ใหญ่ 1,000 - 2,000 มิลลิลิตรใน 12 - 24 ชั่วโมงใน ผู้ป่วยที่มีภาวะขาดน้ำรุนแรงหรือช็อก ในระยะแรกควรให้อย่างรวดเร็วจนอาการดีขึ้นจึงลดให้ช้าลง

โดยกฎทั่วๆ ไป (Rule of Thumb) พบว่า หนึ่งหยดจะประมาณ 0.04 มิลลิลิตรต่อหยดหรือ ออกมาประมาณ 20 หยดต่อ 1 มิลลิลิตร (cc.) สำหรับน้ำเกลือให้คนไข้แต่ก็ขึ้นกับปัจจัยอื่นด้วย เช่น ความข้นของน้ำยา อุณหภูมิ เป็นต้น โดยน้ำเกลือจะให้ในผู้ป่วยที่มีอาการ ดังนี้

1. ขาดน้ำ (Dehydration) เนื่องจากท้องเดิน, อาเจียนรุนแรง (เช่น กระเพาะลำไส้อุดตัน ก้อนในสมอง), หอบ (เช่น หืด ปอดอักเสบ หลอดลมฝอยอักเสบ) ควรให้น้ำเกลือที่มี NSS ผสม เช่น NSS, 5 % D/NSS, 5 % D in 1/3 NSS

2. ช็อก (Shock) เนื่องจากเสียเลือด เสียน้ำหรือจากสาเหตุอื่นๆ ควรให้น้ำเกลือที่มี NSS ผสมเช่นเดียวกับข้อ 1

3. หมดสติ หรือกินข้าวและน้ำไม่ได้นานๆ ควรให้น้ำเกลือที่มีเดกซ์โทรสผสมกับน้ำเกลือ เช่น 5 % D/NSS, 5 % D in 1/3 NSS

4. น้ำตาลในเลือดต่ำ (Hypoglycemia) เนื่องจากอดอาหารนาน ๆ, ดื่มเหล้าจัด, ใช้ยารักษา เบาหวานเกินขนาด ควรให้น้ำเกลือที่มีเดกซ์โทรสผสม เช่น 5 % D/W, 5 % D/NSS, 5 % D in 1/3 NSS

5. ผู้ป่วยที่อดอาหารและน้ำก่อนและหลังผ่าตัด ควรให้น้ำเกลือที่มีเดกซ์โทรสผสมกับ น้ำเกลือเช่นเดียวกับข้อ 3

6. ผู้ป่วยที่ต้องฉีดยาเข้าทางหลอดเลือดดำ วันละหลายๆ ครั้ง เลือกให้น้ำเกลือชนิดใดชนิด หนึ่งก็ได้ โดยให้ช้าๆ เพียงเพื่อให้มีสายน้ำเกลือคากับหลอดเลือดดำ เพื่อให้เกิดความสะดวกในการ ฉีดยา

2.9.1 การคำนวณอัตราหยดของน้ำเกลือ

การให้น้ำเกลือตามแผนการรักษานั้นจะต้องคำนวณปริมาณหยดที่แพทย์สั่งได้ว่าจะปรับ ให้แก่ผู้ป่วยอย่างไร โดยทั่วไปน้ำเกลือที่แพทย์สั่งให้ผู้ป่วยจะมีปริมาตรขวดละ 1000 มิลลิลิตร ซึ่ง การที่จะปรับหยดนั้นจะต้องทราบจำนวนหยดต่อมิลลิลิตรของเซตน้ำเกลือว่าเป็นแบบใด โดยเซต น้ำเกลือจะมี 2 แบบคือ ชนิด microdrop เป็นแบบ 60 หยดต่อมิลลิลิตร และ macrodrop เป็นแบบ 15 หรือ 20 หยดต่อมิลลิลิตร โดยดูได้จากซองใส่เซตน้ำเกลือ โดยใช้สูตรในการคำนวณดังต่อไปนี้

$$\text{mL/Hr} = \frac{\text{จำนวนน้ำเกลือที่ต้องการให้ทั้งหมด}}{\text{จำนวนชั่วโมงที่ให้}} \quad (1)$$

$$\text{mL/min} = \frac{\text{จำนวน มิลลิลิตร ต่อชั่วโมง}}{60 \text{ นาที}} \quad (2)$$

$$\text{drop/min} = \text{จำนวนมิลลิลิตรต่อนาที} \times \text{จำนวนหยดต่อ 1 มิลลิลิตร} \quad (3)$$

Ex. แพทย์สั่งให้น้ำเกลือ 1000 มิลลิลิตรให้ภายใน 8 ชั่วโมง คำนวณจำนวน mL/Hr , mL/min และ drop/min เมื่อใช้เซตน้ำเกลือแบบ 20 drops/ml

$$\text{mL/Hr} = \frac{1000\text{mL}}{8\text{Hr}} = 125\ \text{mL/Hr}$$

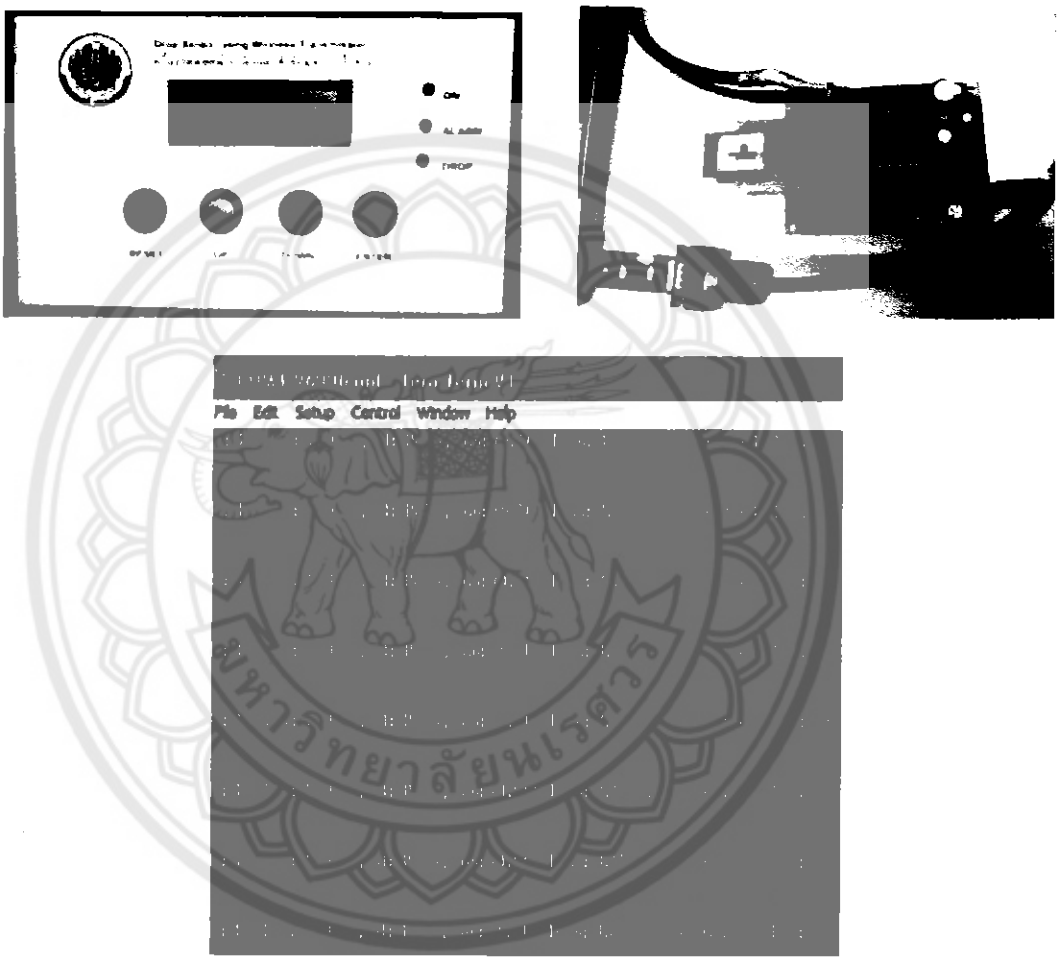
$$\text{mL/min} = \frac{125\text{mL}}{60\text{min}} = 2.1\ \text{mL/min}$$

$$\text{drop/min} = 2.1\ \text{mL/min} \times 20\ \text{drops/ml} = 42\ \text{drops/min}$$

2.10 ระบบตรวจเฝ้าระวังการให้น้ำเกลือแบบไร้สาย

จากผลงานวิจัยของ ดร. สุเมธ อ่ำชิต อาจารย์ประจำภาควิชาฟิสิกส์อุตสาหกรรมและอุปกรณ์การแพทย์ คณะวิทยาศาสตร์ประยุกต์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ พร้อม นายวุฒิชัย วรนันทกุล นั่นคือ การพัฒนาระบบตรวจเฝ้าระวังการให้น้ำเกลือแบบไร้สาย โดยมีวัตถุประสงค์เพื่อพัฒนาระบบตรวจเฝ้าระวังการให้น้ำเกลือแบบไร้สาย โดยระบบที่พัฒนาขึ้นมีความสามารถในการวัดอัตราการหยดของน้ำเกลือ (หยดต่อนาที) อัตราการไหล (มิลลิลิตรต่อนาที) ปริมาตรที่เหลือของน้ำเกลือ (มิลลิลิตร) การแจ้งเตือนเมื่ออัตราการหยดของน้ำเกลือเกิน 200 หยดต่อนาที แจ้งเตือนน้ำเกลือในถุงใกล้หมดและแจ้งเตือนการ อุดตันของสายน้ำเกลือเมื่อน้ำเกลือหยุดไหลเกิน 10 วินาที (Occlusion) และส่งข้อมูลไร้สายไปที่คอมพิวเตอร์ บริเวณเคาน์เตอร์พยาบาล เพื่อช่วยให้พยาบาลสามารถ ติดตามและเฝ้าระวังการไหลของน้ำเกลือระยะไกลได้ ซึ่งจากการทดลองเปรียบเทียบค่าอัตราการไหลของสารละลาย (มิลลิลิตรต่อชั่วโมง) ที่วัดได้จากระบบตรวจเฝ้าระวังการให้น้ำเกลือแบบไร้สายกับเครื่องควบคุมการให้สารละลาย ผ่านทางหลอดเลือดอัตโนมัติที่อัตราการหยดของน้ำเกลือ ประมาณ 5-50 หยดต่อนาที หรือประมาณ 25-200 มิลลิลิตรต่อชั่วโมง พบว่าค่าทั้งสองมีความสัมพันธ์ไปในทิศทางเดียวกัน โดยสามารถคำนวณหาค่าสัมประสิทธิ์สหสัมพันธ์ได้เท่ากับ 0.9588 และมีค่าความผิดพลาดไม่เกิน ± 6.4 เปอร์เซ็นต์ และเมื่อทำการเปรียบเทียบค่าปริมาตรของน้ำเกลือที่ให้ กับผู้ป่วยโดยวัดจากระบบตรวจเฝ้าระวังการให้น้ำเกลือแบบไร้สายเทียบกับค่าปริมาตรของน้ำเกลือที่อ่านได้จากบีกเกอร์ พบว่าค่าทั้งสองมีค่าใกล้เคียงกันมาก โดยสามารถ คำนวณหาค่าสัมประสิทธิ์สหสัมพันธ์ได้เท่ากับ 0.99984 และมีค่าความผิดพลาดไม่เกิน ± 5.6 เปอร์เซ็นต์ อีกทั้งระบบตรวจเฝ้าระวังการให้น้ำเกลือแบบไร้สายยัง

สามารถส่งข้อมูลไร้สายจากบริเวณเสาน้ำเกลือไปยังภาครับข้อมูลที่คอมพิวเตอร์บริเวณเคาน์เตอร์พยาบาลโดยมีระยะห่างไม่เกิน 50 เมตร แบบมีกำแพงกั้น และสามารถส่งข้อมูลจากภาคส่งข้อมูลแบบไร้สายไปยังภาครับข้อมูลที่คอมพิวเตอร์บริเวณเคาน์เตอร์พยาบาล ซึ่งในปัจจุบันเครื่องควบคุมการให้สารละลายผ่านทางหลอดเลือดอัตโนมัติยังไม่สามารถเฝ้าระวังและแจ้งเตือนการให้สารน้ำเกลือในระยะไกลได้



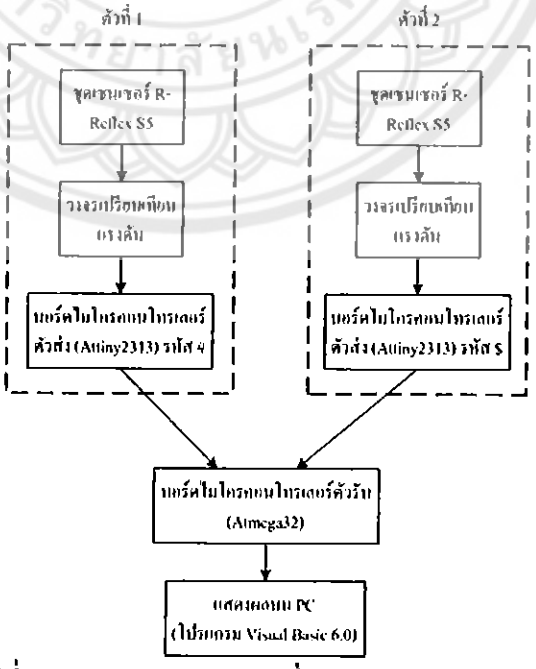
รูปที่ 2.16 ระบบตรวจเฝ้าระวังการให้น้ำเกลือแบบไร้สาย

บทที่ 3
วิธีการดำเนินโครงงาน

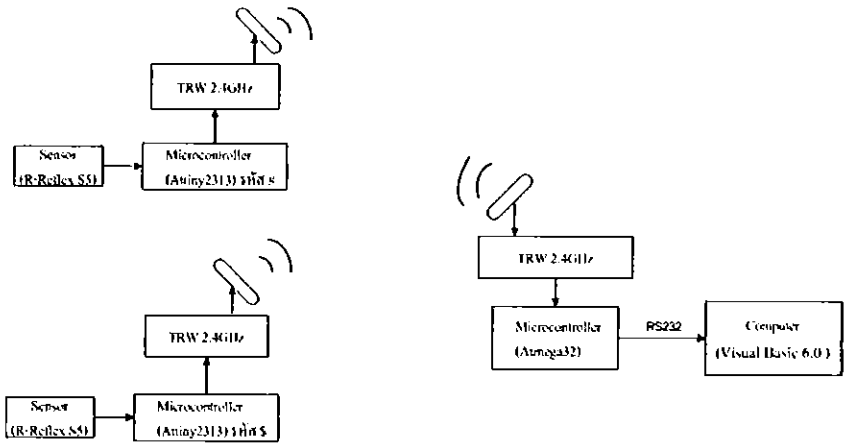
การออกแบบและพัฒนาระบบแสดงอัตราการให้น้ำเกลือแบบไร้สาย หลังจากที่ได้ศึกษาข้อมูลและทฤษฎีที่เกี่ยวข้องแล้ว ผู้ดำเนินโครงงานได้ออกแบบและพัฒนาเครื่องต้นแบบสำหรับระบบแสดงอัตราการให้น้ำเกลือแบบไร้สาย โดยการออกแบบให้สอดคล้องและเหมาะสม ซึ่งการออกแบบแบ่งออกเป็น 3 ส่วน ดังนี้

1. การออกแบบทางฮาร์ดแวร์
2. การออกแบบโปรแกรม
3. การออกแบบโปรแกรมแสดงผล

ซึ่งสามารถเขียนแผนภาพโดยรวมแสดงการเชื่อมต่อวงจรและอุปกรณ์ได้ดังรูปที่ 3.1 โดยเริ่มจากการออกแบบทางฮาร์ดแวร์และออกแบบโปรแกรมภาษาซีในการรับสัญญาณจากเซนเซอร์ (R-Reflex S5) ที่เชื่อมต่อกับชุดวงจรเปรียบเทียบแรงดันแล้วส่งสัญญาณเอาต์พุตมาที่บอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) เพื่อนำมาประมวลผล ต่อมาเขียนภาษาซีในการรับส่งข้อมูลระหว่างบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) กับบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) จากนั้นเขียนโปรแกรมภาษาซีส่งข้อมูลเหล่านั้นไปยังคอมพิวเตอร์ส่วนบุคคล เพื่อแสดงผลบนหน้าจอมอนิเตอร์ด้วยโปรแกรม Visual Basic 6.0



รูปที่ 3.1 แผนภาพแสดงการเชื่อมต่อวงจรและอุปกรณ์



รูปที่ 3.2 แผนภาพโดยรวมของโครงงาน

จากรูปที่ 3.2 แผนภาพโดยรวมของโครงงาน สามารถอธิบายการทำงานขององค์ประกอบแต่ละส่วนได้ดังนี้

เซนเซอร์ (R-Reflex S5) ทำหน้าที่ในการนับหยดน้ำเกลือ โดยเซนเซอร์จะทำการตรวจจับหยดน้ำเกลือมาเป็นสัญญาณแรงดัน แล้วส่งไปยังชุดวงจรเปรียบเทียบแรงดันที่มีการเชื่อมต่อกัน ทำหน้าที่ในการเปรียบเทียบแรงดันเป็นสัญญาณแรงดันดิจิตอลลอจิก "0" (0V) และ "1" (+5V) โดยเอาท์พุตที่ได้จากการนับหยดน้ำเกลือ 1 หยดเท่ากับ 1 บิตจะมีค่าเป็นลอจิก "0" (0V) จากนั้นส่งข้อมูลนี้ให้กับบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) เพื่อนำมาประมวลผล

บอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ทำหน้าที่ในการรับเอาต์พุตสัญญาณแรงดันดิจิตอลลอจิกจากชุดวงจรเปรียบเทียบแรงดันที่เชื่อมต่อกับตัวเซนเซอร์ (R-Reflex S5) โดยการนำค่าเอาท์พุตที่ได้จากการนับหยดน้ำเกลือ 1 หยดเท่ากับ 1 บิตจะมีค่าเป็นลอจิก "0" (0V) ทำให้อยู่ในรูปของเลขฐานสิบหก ใช้ Timer ในการจับเวลาการนับหยด 10 วินาที เมื่อครบ 10 วินาที ไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) จะทำการส่งจำนวนหยดที่นับได้ไปทีละบิต ในข้อมูลจะมีรหัสประจำเครื่องนำหน้าตามด้วยข้อมูลและปิดท้ายด้วยเครื่องหมาย "!" ตัวอย่างเช่น เครื่องที่มีรหัส # นับหยดน้ำเกลือได้ 10 หยดต่อ 10 วินาที ดังนั้นข้อมูลที่บอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) จะต้องทำการส่งคือ "#10.0!" สำหรับส่งไปยังบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

บอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) ทำหน้าที่ในการนำจำนวนหยดที่นับได้จากบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) แปลงเป็นหน่วยมิลลิลิตรต่อ 10 วินาที โดยใช้เซตน้ำเกลือแบบ 20 หยดต่อ 1 มิลลิลิตร ดังนั้นจะได้สมการ

$$\text{ปริมาตร}(ml) = \frac{\text{จำนวนหยด}}{20}$$

จากนั้นส่งค่าจำนวนหยดที่นับได้ในหน่วยมิลลิลิตรต่อ 10 วินาที โดยกำหนดให้ส่งรหัส # ตามด้วยจำนวนหยดปิดท้ายด้วย ! เป็นลำดับแรก จากนั้นส่งรหัส $ ตามด้วยจำนวนหยดปิดท้ายด้วย ! เป็นลำดับที่สอง ตัวอย่างเช่น #10.00! $15.00! ผ่านสาย RS232 เพื่อนำไปแสดงผลบนหน้าจอมอนิเตอร์ของคอมพิวเตอร์ส่วนบุคคลโดยผ่านโปรแกรม Visual Basic 6.0

โปรแกรม Visual Basic 6.0 ทำหน้าที่ในการแสดงผลข้อมูลและตั้งค่าการใช้งาน โดยโปรแกรม Visual Basic 6.0 จะทำการรับค่าจำนวนหยดที่นับได้จากบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ที่มีหน่วยเป็นหน่วยมิลลิลิตรต่อ 10 วินาที โดยการรวมค่าปริมาตรที่ได้จนครบ 6 ครั้ง (60 วินาที) มาแสดงผลทุกๆ 1 นาที ที่จอแสดงผลจะแสดงค่าเป็นอัตราการไหลของน้ำเกลือขณะนี้เป็นมิลลิลิตรต่อเวลาหนึ่งนาทีและหยดต่อเวลาหนึ่งนาที นอกจากนั้นยังมีหมายเลขเตียงข้อมูลของคนไข้ อัตราการไหลของน้ำเกลือที่ต้องการเป็นมิลลิลิตรต่อเวลาหนึ่งนาที จำนวนหยดต่อเวลาหนึ่งนาที และปริมาตรคงเหลือของน้ำเกลือในขวดน้ำเกลือคิดเป็นเปอร์เซ็นต์

3.1 การออกแบบทางฮาร์ดแวร์

การออกแบบทางฮาร์ดแวร์ทำให้ทราบถึงลักษณะและขนาดของชิ้นงานว่าเป็นอย่างไร การออกแบบวงจรการทำงานซึ่งมีความสัมพันธ์กับลักษณะและขนาดของชิ้นงาน ดังนั้นจึงต้องออกแบบให้สอดคล้องและเหมาะสม นอกจากนี้ยังมีการออกแบบเงื่อนไขการทำงานต่างๆ โดยการออกแบบทางฮาร์ดแวร์นั้นสามารถแบ่งออกเป็น 3 ส่วนใหญ่ๆ คือ

1. การออกแบบบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313)
2. การออกแบบบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)
3. การออกแบบส่วนเชื่อมโยงข้อมูล

3.1.1 การออกแบบบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313)

การออกแบบบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ให้สามารถทำงานได้อย่างถูกต้องตามที่ต้องการและมีประสิทธิภาพนั้น มีส่วนสำคัญที่ต้องคำนึงถึงอยู่ 3 ส่วน คือ

1. การออกแบบวงจร
2. การออกแบบเงื่อนไขการทำงาน

ก. การออกแบบวงจร

การออกแบบวงจรบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) เป็นการออกแบบวงจรการทำงานให้กับบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ซึ่งใช้ TRW-2.4GHz เป็นตัวส่งสัญญาณ โดยใช้ไมโครคอนโทรลเลอร์ในการควบคุมการทำงาน การเชื่อมโยงระหว่าง TRW 2.4 GHz กับ Microcontroller นั้น เป็นดังรายละเอียดที่ภาคผนวก ก คู่มือ TRW-2.4GHz จะใช้

สายสัญญาณเพียง 5 เส้น จากที่ออกแบบได้ใช้พอร์ต B ซึ่ง TRW 2.4 GHz ทำงานในช่วง 1.9 – 3.6V ดังนั้นไมโครคอนโทรลเลอร์ที่จะนำมาเชื่อมต่อจะต้องทำงานได้ในช่วงแรงดันที่ค่อนข้างกว้าง ซึ่งโดยปกติแล้วไมโครคอนโทรลเลอร์ส่วนใหญ่จะทำงานอยู่ในช่วงแรงดัน 5V ซึ่งถ้านำมาเชื่อมต่อจะต้องมีการแปลงแรงดันขึ้น-ลงทำให้วงจรซับซ้อนมากยิ่งขึ้น จึงเลือกใช้ไมโครคอนโทรลเลอร์ ATTINY2313 เนื่องจากช่วงแรงดันที่ไมโครคอนโทรลเลอร์จะมองเป็นลอจิก 1 นั้นคือตั้งแต่ 4.5V ถึง 5.5V ทำให้ทั้ง TRW 2.4 GHz และไมโครคอนโทรลเลอร์ทำงาน โดย

PB.0 ต่อกับขา CE (Chip Enable) ทำหน้าที่ในการสั่งให้ TRW 2.4 GHz ทำงานหรือไม่ทำงาน โดยถ้า CE = 1 คือให้ TRW 2.4 GHz ทำงานและถ้า CE = 0 คือไม่ให้ TRW 2.4 GHz ทำงาน

PB.1 ต่อกับขา CS (Chip Select) ทำงานร่วมกับสัญญาณ CE ซึ่งจะเป็นการกำหนดการทำงานของ TRW-2.4GHz

PB.2 ต่อกับขา CLK1 (Clock) ทำงานร่วมกับสัญญาณ DATA โดยจะทำให้ทราบได้ว่า ข้อมูลมีกี่บิตและเป็นข้อมูลอะไร

PB.3 ต่อกับขา DATA ทำหน้าที่ในการรับหรือส่งข้อมูล

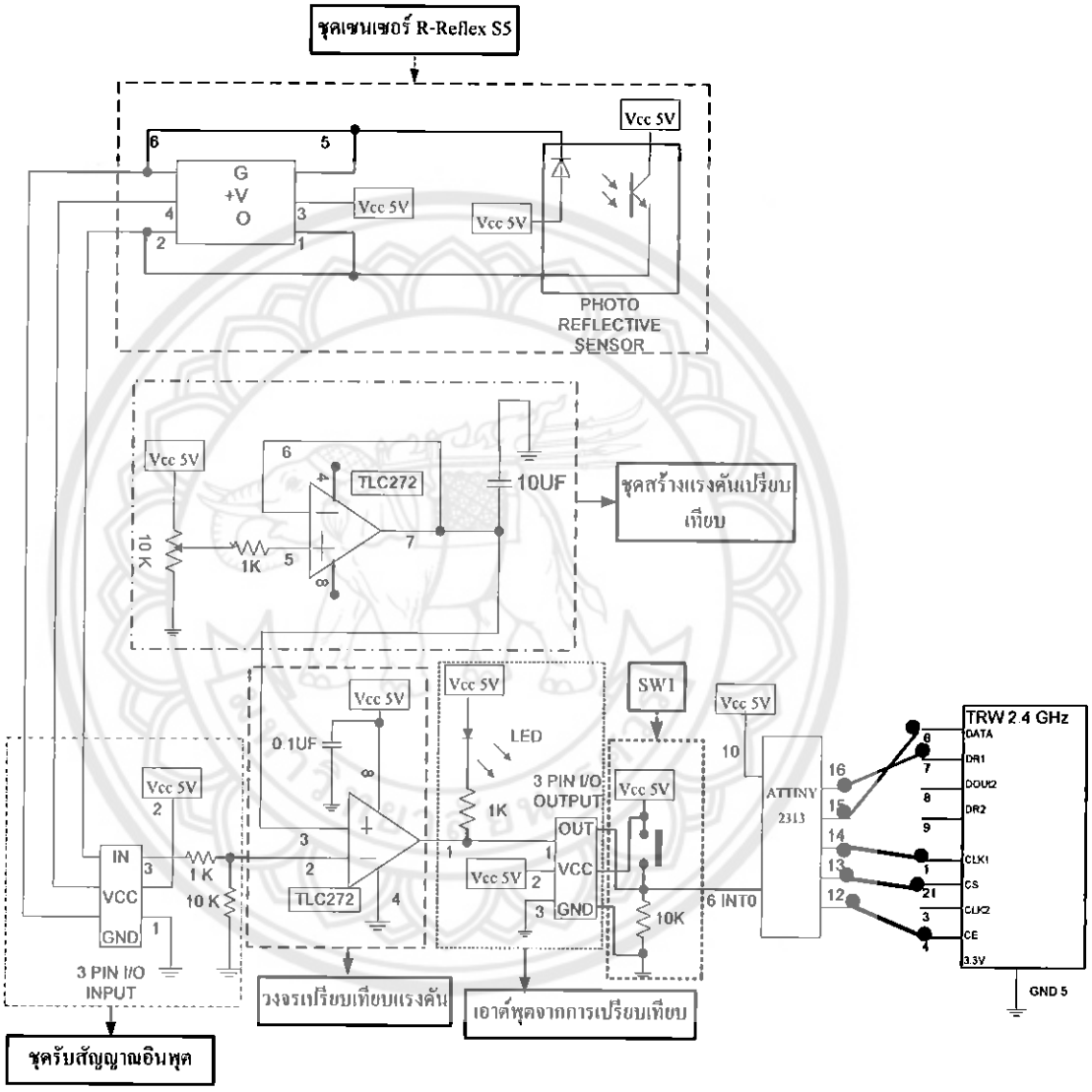
PB.4 ต่อกับขา DR1 (Data Ready) ทำหน้าที่ในการบอกสถานะของข้อมูล คือ ถ้า DR1 = 0 คือไม่มีข้อมูล แต่ถ้า DR1 = 1 คือมีข้อมูล

หลังจากที่ทำการกำหนดให้ TRW 2.4GHz เป็นตัวส่งข้อมูลแล้ว จากนั้นเมื่อไมโครคอนโทรลเลอร์มีข้อมูลที่จะส่งต้องทำให้สัญญาณ CE = 1 โดยการเซทพอร์ต PB.4 ให้เป็น 1 จากนั้นก็ทำการส่งข้อมูลและสัญญาณ Clock ออกไปได้ และหากต้องรีเซทการทำงานก็กดที่ปุ่ม Reset ได้

ข. การออกแบบเงื่อนไขในการทำงาน

เงื่อนไขการทำงานของบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) คือ บอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313)) ทำหน้าที่ในการรับเอาต์พุตสัญญาณแรงดันดิจิตอลลอจิกจากชุดวงจรเปรียบเทียบแรงดันที่เชื่อมต่อกับตัวเซนเซอร์(R-Reflex S5) โดยการนำค่าเอาท์พุตที่ได้จากการนับหยดน้ำเกลือ 1 หยดเท่ากับ 1 บิตจะมีค่าเป็นลอจิก "0" (0V) ทำให้อยู่ในรูปของเลขฐานสิบหก ใช้ Timer ในการจับเวลาการนับหยด 10 วินาที เมื่อครบ 10 วินาที ไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) จะทำการส่งจำนวนหยดที่นับได้ไปทีละบิต ทันทีที่บอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) ส่งรหัสประจำเครื่องที่ตรงกันมา ในข้อมูลจะมีรหัสประจำเครื่องนำหน้าตามด้วยข้อมูลและปิดท้ายด้วยเครื่องหมาย "!" และจะหยุดเมื่อมีการกดปุ่ม Reset จึงต้องมีวงจร Reset และมีปุ่มกดเพื่อทำให้สามารถรีเซทได้ทุกเวลาที่ต้องการ โดยปุ่มกดต้องเป็นแบบกดติด-ปล่อยดับ มีสวิทช์ SW1 แทนการหยดของน้ำเกลือเพื่อง่ายต่อการทดสอบเครื่อง และมีการ

เชื่อมต่อกับเซนเซอร์ (R-Reflex S5) นอกจากนี้ในการควบคุมการทำงานของ TRW 2.4 GHz นั้นมีการหน่วงเวลา (Time Delay) มาเกี่ยวข้องเยอะมาก และการกำหนดการหน่วงเวลาเหล่านี้ก็ขึ้นอยู่กับความถี่ของ Oscillator ด้วย ดังนั้น เพื่อความง่ายในการคำนวณTime Delay จึงควรใช้ Oscillator ที่คำนวณแล้ว ซึ่งในที่นี้ใช้ Oscillator ขนาด 11.0592 MHz



รูปที่ 3.3 วงจรของบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313)

จากรูปที่ 3.3 วงจรของบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ต่อเข้ากับวงจรเซนเซอร์ (R-Reflex S5) ทำหน้าที่ในการนับหยดน้ำเกลือ โดยเซนเซอร์จะทำการตรวจจับหยดน้ำเกลือมาเป็นสัญญาณแรงดัน แล้วส่งไปยังชุดวงจรเปรียบเทียบแรงดันที่มีการเชื่อมต่อกัน วงจรเปรียบเทียบแรงดัน (Voltage Comparator) ใช้ออปแอมป์ (Op-amp) ต่อเป็นวงจรเปรียบเทียบแรงดัน สามารถแยกส่วนประกอบของวงจรได้ดังนี้

● **วงจรชุดเซนเซอร์ตรวจสอบการสะท้อนกลับของสัญญาณอินฟราเรด**

ทำหน้าที่ในการนับหยดน้ำเกลือ โดยเซนเซอร์จะทำการตรวจจับหยดน้ำเกลือมาเป็นสัญญาณแรงดัน แล้วส่งไปยังชุดวงจรเปรียบเทียบแรงดันที่มีการเชื่อมต่อกัน ทำหน้าที่ในการเปรียบเทียบแรงดันเป็นสัญญาณแรงดันดิจิตอลลอจิก "0" (0V) และ "1" (+5V) โดยเอาท์พุตที่ได้จากการนับหยดน้ำเกลือ 1 หยดเท่ากับ 1 บิตจะมีค่าเป็นลอจิก "0" (0V) จากนั้นส่งข้อมูลนี้ให้กับบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) เพื่อนำมาประมวลผล

● **ชุดวงจรเปรียบเทียบแรงดัน**

ชุดวงจรส่วนนี้เป็นการต่อชุดรับสัญญาณอินพุต ชุดวงจรสร้างแรงดันอ้างอิงและวงจรเปรียบเทียบแรงดันเข้าด้วยกัน โดยชุดรับสัญญาณอินพุตทำหน้าที่เป็นจุดต่อรับสัญญาณแรงดันที่ได้จากชุดเซนเซอร์ R - REFLEX S5 โดยใช้คอนเน็คเตอร์แบบ 6 pin เป็นจุดต่อรับสัญญาณ สำหรับชุดวงจรสร้างแรงดันอ้างอิง วงจรส่วนนี้ใช้ตัวต้านทานปรับค่าได้ ขนาด 10 KΩ ในการปรับค่าแรงดันผ่านวงจรออปแอมป์ ที่ทำหน้าที่เป็นวงจรบัฟเฟอร์ แรงดันเอาต์พุตของวงจรนี้เป็นแรงดันอ้างอิงในการเปรียบเทียบแรงดัน โดยสามารถปรับได้ตั้งแต่ 0 โวลต์ไปจนถึงค่าแรงดัน VCC และวงจรเปรียบเทียบแรงดันทำการเปรียบเทียบแรงดันที่จุดอ้างอิงกับแรงดันอินพุตจากชุดเซนเซอร์ R - REFLEX S5 โดยค่าเอาต์พุตจากการเปรียบเทียบเป็นดังตารางต่อไปนี้

ตารางที่ 3.1 แสดงการเปรียบเทียบแรงดันที่จุดอ้างอิงกับแรงดันอินพุต

การเปรียบเทียบแรงดัน	เอาต์พุต
แรงดันอินพุต > แรงดันอ้างอิง	0 โวลต์
แรงดันอินพุต < แรงดันอ้างอิง	+VCC

● **เอาต์พุตจากการเปรียบเทียบ**

สัญญาณเอาต์พุตจากวงจรเปรียบเทียบแรงดันจะถูกต่อมาที่คอนเน็คเตอร์ 3 PIN I/O โดยมีแอลอีดีแสดงสถานะของสัญญาณเอาต์พุต ดังนี้

เอาต์พุต = 0 โวลต์ หลอดไฟแอลอีดีติดสว่าง

เอาต์พุต = +VCC หลอดไฟแอลอีดีติดดับ

ชุดเอาต์พุตจากการเปรียบเทียบต่อเข้ากับขา INT0 ของไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) โดยผ่านสวิทช์ SW1 ซึ่งทำหน้าที่เสมือนเป็นชุดเซนเซอร์ R - REFLEX S5 เพื่อสะดวกในการตรวจสอบการทำงานของบอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) เมื่อไม่มีการต่อเข้ากับสายน้ำเกลือ การกดสวิทช์ 1 ครั้งเท่ากับน้ำเกลือที่หยด 1 หยด

3.1.2 การออกแบบบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

การออกแบบบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) ให้สามารถทำงานได้อย่างถูกต้องตามที่ต้องการและมีประสิทธิภาพนั้น มีส่วนสำคัญที่ต้องคำนึงถึงอยู่ 3 ส่วน คือ

1. การออกแบบวงจร
2. การออกแบบเงื่อนไขการทำงาน

ก. การออกแบบวงจร

การออกแบบวงจรบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) เป็นการออกแบบวงจรการทำงานให้กับบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) ซึ่งใช้ TRW 2.4 GHz เป็นตัวรับสัญญาณ โดยใช้ไมโครคอนโทรลเลอร์ในการควบคุมการทำงาน การเชื่อมโยงระหว่าง TRW 2.4 GHz กับ Microcontroller เป็นดังรายละเอียดที่ภาคผนวก ก คู่มือ TRW-2.4GHz จะใช้สายสัญญาณเพียง 5 เส้น จากที่ออกแบบได้ใช้พอร์ต C ซึ่ง TRW 2.4 GHz ทำงานในช่วง 1.9 – 3.6V ดังนั้นไมโครคอนโทรลเลอร์ที่จะนำมาเชื่อมต่อจะต้องทำงานได้ในช่วงแรงดันที่ค่อนข้างกว้างซึ่งโดยปรกติแล้วไมโครคอนโทรลเลอร์ส่วนใหญ่จะทำงานอยู่ในช่วงแรงดัน 5V ซึ่งถ้าเรานำมาเชื่อมต่อจะต้องมีการแปลงแรงดันขึ้น-ลงทำให้วงจรซับซ้อนมากยิ่งขึ้น เราจึงเลือกใช้ไมโครคอนโทรลเลอร์ ATMEGA32 เนื่องจากช่วงแรงดันที่ไมโครคอนโทรลเลอร์จะมองเป็นลอจิก 1 นั้นคือตั้งแต่ 4.5V ถึง 5.5V ทำให้ทั้ง TRW 2.4 GHz และไมโครคอนโทรลเลอร์ทำงาน โดย

PC.0 ต่อกับขา CE (Chip Enable) ทำหน้าที่ในการสั่งให้ TRW 2.4 GHz ทำงานหรือไม่ทำงาน โดยถ้า CE = 1 คือให้ TRW 2.4 GHz ทำงานและถ้า CE = 0 คือไม่ให้ TRW 2.4 GHz ทำงาน

PC.1 ต่อกับขา CS (Chip Select) ทำงานร่วมกับสัญญาณ CE ซึ่งจะเป็นการกำหนดการทำงานของ TRW 2.4 GHz

PC.2 ต่อกับขา CLK1 (Clock) ทำงานร่วมกับสัญญาณ DATA โดยจะทำให้เราทราบได้ว่า ข้อมูลมีกี่บิตและเป็นข้อมูลอะไร

PC.3 ต่อกับขา DATA ทำหน้าที่ในการรับหรือส่งข้อมูล

PC.4 ต่อกับขา DR1 (Data Ready) ทำหน้าที่ในการบอกสถานะของข้อมูล คือ ถ้า DR1 = 0 คือไม่มีข้อมูล แต่ถ้า DR1 = 1 คือมีข้อมูล

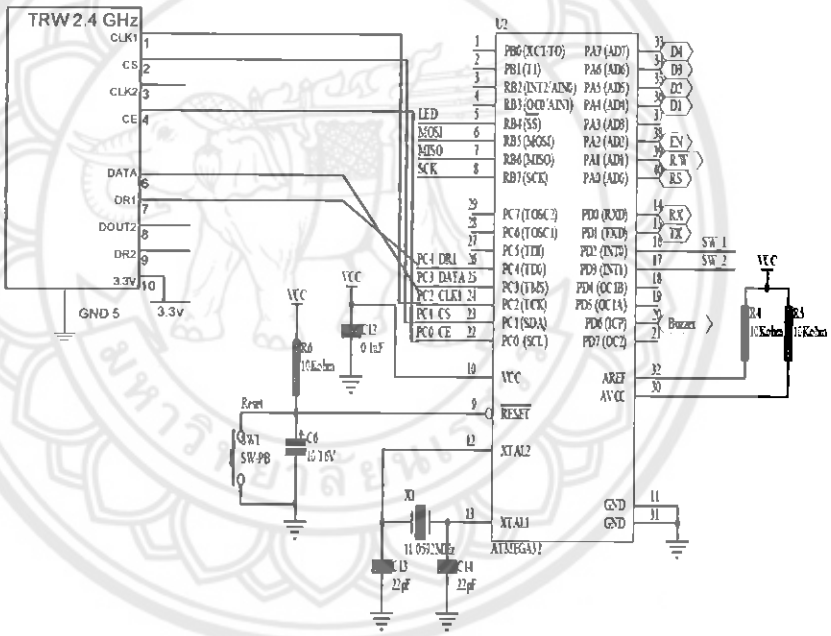
นอกจากนั้นจะมีการต่อสัญญาณจากขา Tx (พอร์ต PD.1) มาให้กับส่วนเชื่อมโยงข้อมูลเพื่อส่งข้อมูลที่รับมาได้จากเครื่องส่งมาให้กับส่วนเชื่อมโยงข้อมูล

เพื่อให้การทำงานมีประสิทธิภาพจำเป็นต้องมีบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) เนื่องจากบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) กับบอร์ด

ไมโครคอนโทลเลอร์ตัวรับ (Atmega32) รับ-ส่งข้อมูลผ่านโมดูล TRW 2.4 GHz จึงทำให้บอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ส่งข้อมูลไปที่คอมพิวเตอร์โดยตรงไม่ได้

ข. การออกแบบเงื่อนไขในการทำงาน

เงื่อนไขในการทำงานของบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) คือ ทำหน้าที่คอยนับและตรวจสอบข้อมูลที่รับมานั้นว่าถูกต้องครบถ้วนแล้วหรือยัง ถ้ายังไม่ครบก็จะรอรับจนกว่าจะครบและเมื่อรับข้อมูลมาครบแล้วจะนำจำนวนหยดที่นับได้จากบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) แปลงเป็นหน่วยมิลลิลิตรต่อ 10 วินาที ส่งข้อมูลเหล่านั้นไปยังส่วนเชื่อมโยงข้อมูลทางพอร์ตอนุกรม (Tx) จากนั้นส่งข้อมูลที่ได้ผ่านสาย RS232 เพื่อนำไปแสดงผลบนหน้าจอมอนิเตอร์ของคอมพิวเตอร์ส่วนบุคคลโดยผ่านโปรแกรม Visual Basic 6.0



รูปที่ 3.4 วงจรของบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

จากรูปที่ 3.4 สามารถอธิบายการทำงานของโปรแกรมได้ คือ เมื่อบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) รับข้อมูลจากบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ก็จะรอจนกว่าจะรับข้อมูลครบ จากนั้นเมื่อรับข้อมูลมาครบแล้ว ก็จะส่งสัญญาณที่เป็นลอจิก 1 ออกมาทางขอ DR1 (พอร์ต PC.4) เพื่อให้ไมโครคอนโทรลเลอร์รู้ว่ารับข้อมูลมาครบแล้ว พร้อมกับส่งสัญญาณข้อมูล DATA มาที่พอร์ต PC.0 และสัญญาณ Clock มาที่พอร์ต PC.3 จากนั้นไมโครคอนโทรลเลอร์ก็จะรับเอาข้อมูลที่เข้ามานั้นทำการคำนวณแปลงจำนวนหยดน้ำเกลือ

เป็นหน่วยมิลลิลิตรต่อนาที แล้วส่งออกไปทางขา Tx (พอร์ต PD.1) ไปให้กับส่วนเชื่อมโยงข้อมูลต่อไป

3.1.3 การออกแบบส่วนเชื่อมโยงข้อมูล

การออกแบบส่วนเชื่อมโยงข้อมูลให้สามารถทำงานได้อย่างถูกต้องตามที่ต้องการและมีประสิทธิภาพนั้น มีส่วนสำคัญที่ต้องคำนึงถึงอยู่ 2 ส่วนด้วยกันคือ

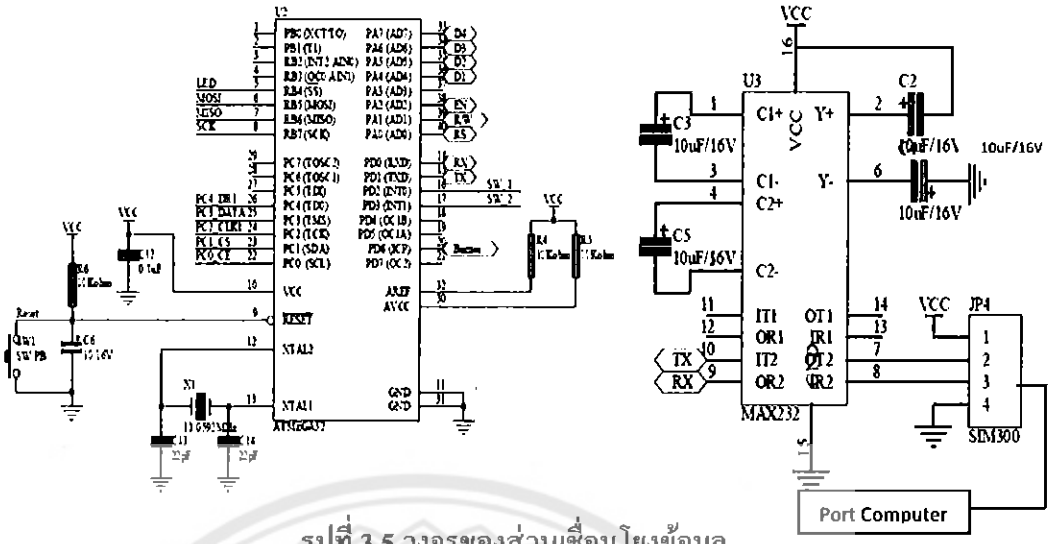
1. การออกแบบวงจร
2. การออกแบบเงื่อนไขการทำงาน

ก. การออกแบบวงจร

ในการออกแบบส่วนเชื่อมโยงข้อมูลจะประกอบไปด้วยส่วนที่เชื่อมโยงระหว่างบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) กับส่วนเชื่อมโยงข้อมูลกับคอมพิวเตอร์ ซึ่งส่วนที่เชื่อมโยงระหว่างบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) กับไมโครคอนโทรลเลอร์ของส่วนเชื่อมโยงข้อมูลสามารถเชื่อมต่อกันได้โดยตรง เนื่องจากใช้แรงดันอยู่ในช่วงเดียวกัน แต่ส่วนที่เชื่อมโยงระหว่างไมโครคอนโทรลเลอร์ของส่วนเชื่อมโยงข้อมูลกับคอมพิวเตอร์นั้น มีระดับแรงดันต่างกัน โดยที่คอมพิวเตอร์มีระดับแรงดันอยู่ที่ 15V แต่ไมโครคอนโทรลเลอร์มีระดับแรงดันอยู่ที่ 5V เท่านั้น จึงต้องมีอุปกรณ์ที่สามารถแปลงแรงดัน 5V ให้เป็นแรงดัน 15V ได้ โดยเราได้ใช้ไอซี MAX232 ซึ่งเป็นไอซีที่นิยมใช้กันทั่วไปและมีพอร์ตส่งสัญญาณและพอร์ตรับสัญญาณอย่างละ 1 คู่ จากการออกแบบต้องการส่งข้อมูลให้กับคอมพิวเตอร์เพียงอย่างเดียว ไม่มีการรับข้อมูลเข้ามา ดังนั้นจึงใช้พอร์ตในการส่งเพียงพอร์ตเดียว โดยใช้พอร์ต T2 ซึ่งจะมีขาสัญญาณ 2 ขา คือสัญญาณ IT2 และสัญญาณ OT2 โดยสัญญาณ IT2 จะต่อเข้ากับขา Tx ของไมโครคอนโทรลเลอร์และสัญญาณ OT2 จะต่อเข้ากับคอมพิวเตอร์ทาง RS-232

ข. การออกแบบเงื่อนไขในการทำงาน

เงื่อนไขการทำงานของส่วนเชื่อมโยงข้อมูล คือ จะเป็นทางผ่านของข้อมูลจากเครื่องรับไปยังคอมพิวเตอร์ โดยจะคอยรับข้อมูลจากเครื่องรับผ่านทางขา Rx แล้วส่งข้อมูลออกมาทางขา Tx ผ่าน MAX232 แล้วไปเข้าที่คอมพิวเตอร์ผ่านทาง RS-232 เพื่อแสดงผลผ่านทางโปรแกรม Visual Basic 6.0 ทั้งนี้จะต้องมีการกำหนดอัตรารับส่งข้อมูล (Baud rate) ที่ระหว่างส่วนเชื่อมโยงข้อมูลกับเครื่องรับและระหว่างส่วนเชื่อมโยงข้อมูลกับคอมพิวเตอร์ให้ตรงกัน มิฉะนั้นจะไม่สามารถรับ-ส่งข้อมูลกันได้



รูปที่ 3.5 วงจรของส่วนเชื่อมโยงข้อมูล

การออกแบบโปรแกรมจากรูปที่ 3.5 สามารถอธิบายการทำงานได้ ดังนี้ เมื่อมีข้อมูลจากบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) ส่งเข้ามาที่ขา Rx (PD.0) ของไมโครคอนโทรลเลอร์ของส่วนเชื่อมโยงข้อมูล ไมโครคอนโทรลเลอร์ก็จะรับข้อมูลนั้นเข้ามาแล้วส่งข้อมูลที่มีขนาดแรงดัน 5V ทางขา Tx (PD.1) ไปเข้าที่ขา IT2 ของ MAX232 เพื่อทำการแปลงแรงดันจาก 5V ให้เป็น 15V แล้วส่งข้อมูลที่มีขนาดแรงดัน 15V ไปให้กับคอมพิวเตอร์เพื่อแสดงผลผ่านทางโปรแกรม Visual Basic 6.0

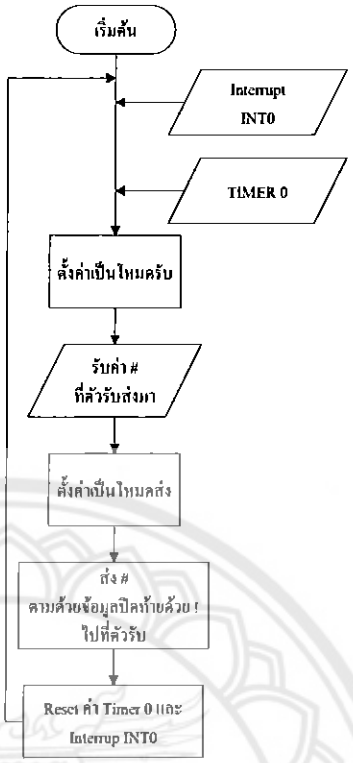
3.2 การออกแบบโปรแกรม

การออกแบบโปรแกรมเป็นการออกแบบโปรแกรมการทำงานของบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)และบอร์ดไมโครคอนโทลเลอร์ตัวส่ง(Attiny2313) ซึ่งประกอบด้วยการรับส่งข้อมูล การเชื่อมโยงข้อมูล การควบคุม การประมวลผลและการสั่งการทั้งหมด การออกแบบซอฟแวร์เหล่านี้ สามารถแบ่งเป็นส่วนๆ ได้แก่

1. การออกแบบโปรแกรมบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313)
2. การออกแบบโปรแกรมบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

3.2.1 การออกแบบโปรแกรมบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313)

การออกแบบขั้นตอนการทำงานของโปรแกรมเป็นการออกแบบโปรแกรมเพื่อควบคุมการทำงานของบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) เพื่อให้สามารถส่งสัญญาณข้อมูลออกไปได้อย่างถูกต้อง มีขั้นตอนดังนี้



รูปที่ 3.6 แผนภาพแสดงการเชื่อมต่อวงจรของตัวส่ง (Attiny2313) ที่มีรหัสประจำเครื่องคือ #

การทำงานของบอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313)

บอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) ที่มีรหัสประจำเครื่องคือ #

แผนภาพการทำงานของการเชื่อมต่อวงจรของตัวส่ง (Attiny2313) ดังรูปที่ 3.6 ที่มีรหัสประจำเครื่องคือ # สามารถอธิบายการทำงานได้ ดังนี้

1. เริ่ม start เข้ามา Interrupt INT0 และ TIMER0 จะเริ่มทำงาน โดยที่ Interrupt INT0 มาจาก เซนเซอร์ (R-Reflex S5) ทำหน้าที่ในการนับหยดน้ำเกลือ โดยเซนเซอร์จะทำการตรวจจับหยดน้ำเกลือมาเป็นสัญญาณแรงดัน แล้วส่งไปยังชุดวงจรเปรียบเทียบแรงดันที่มีการเชื่อมต่อกัน ซึ่งมีหน้าที่ในการเปรียบเทียบแรงดันเป็นสัญญาณแรงดันดิจิตอลลอจิก "0" (0V) และ "1" (+5V) โดยเอาท์พุตที่ได้จากการนับหยดน้ำเกลือ 1 หยดเท่ากับ 1 บิตจะมีค่าเป็นลอจิก "0" (0V) และ TIMER0 ทำหน้าที่เหมือนกับนาฬิกาจับเวลาทุกๆ 10 วินาที

2. บอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) จะตั้งค่าเป็นโหมดรับ

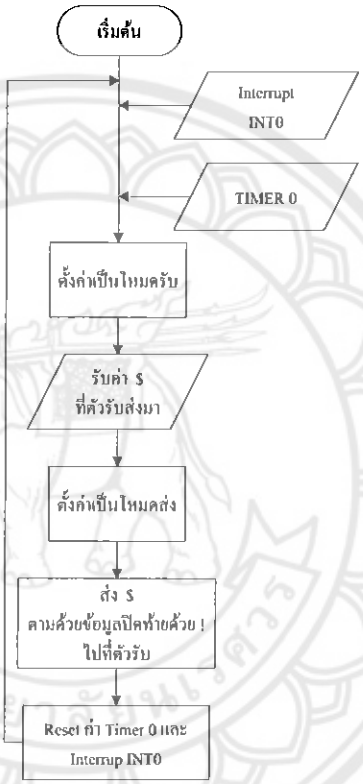
3. เมื่อรหัสที่ส่งมาจากตัวรับ (Atmega32) เป็น # ซึ่งตรงกับรหัสประจำเครื่อง บอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) จะตั้งค่าเป็นโหมดส่ง

4. ส่งข้อมูล (จำนวนหยดน้ำเกลือที่นับได้ต่อ 10 วินาที) ไปทีละบิต ในข้อมูลจะมีรหัสประจำเครื่องนำหน้าตามด้วยข้อมูล และปิดท้ายด้วยเครื่องหมาย "!" ตัวอย่างเช่น #10.0! ส่งไปยังบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

5. เมื่อมีการส่งข้อมูลครบถ้วนแล้ว จะทำการรีเซตค่า Timer 0 และ Interrupt INT0 เท่ากับ 0 เพื่อนับค่าหยดน้ำเกลือครั้งต่อไป

บอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) ที่มีรหัสประจำเครื่องคือ $

แผนภาพการทำงานของการเชื่อมต่อวงจรของตัวส่ง (Attiny2313) ดังรูปที่ 3.7 ที่มีรหัสประจำเครื่องคือ $ สามารถอธิบายการทำงานได้ ดังนี้



รูปที่ 3.7 แผนภาพแสดงการเชื่อมต่อวงจรของตัวส่ง (Attiny2313) ที่มีรหัสประจำเครื่องคือ $

1. เริ่ม start เข้ามา Interrupt INT0 และ TIMER0 จะเริ่มทำงาน โดยที่ Interrupt INT0 มาจาก เซนเซอร์ (R-Reflex S5) ทำหน้าที่ในการนับหยดน้ำเกลือ โดยเซนเซอร์จะทำการตรวจจับหยดน้ำเกลือมาเป็นสัญญาณแรงดัน แล้วส่งไปยังชุดวงจรเปรียบเทียบแรงดันที่มีการเชื่อมต่อกัน ซึ่งมีหน้าที่ในการเปรียบเทียบแรงดันเป็นสัญญาณแรงดันดิจิตอลลอจิก "0" (0V) และ "1" (+5V) โดยเอาท์พุตที่ได้จากการนับหยดน้ำเกลือ 1 หยดเท่ากับ 1 บิตจะมีค่าเป็นลอจิก "0" (0V) และ TIMER0 ทำหน้าที่เหมือนกับนาฬิกาจับเวลาทุกๆ 10 วินาที

2. บอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) จะตั้งค่าเป็นโหมดรับ

3. เมื่อรหัสที่ส่งมาจากตัวรับ (Atmega32) เป็น $ ซึ่งตรงกับรหัสประจำเครื่อง บอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) จะตั้งค่าเป็นโหมดส่ง

4. ส่งข้อมูล (จำนวนหยดน้ำเกลือที่นับได้ต่อ 10 วินาที) ไปทีละบิต ในข้อมูลจะมีรหัสประจำเครื่องนำหน้าตามด้วยข้อมูลและปิดท้ายด้วยเครื่องหมาย "!" ตัวอย่างเช่น $10.0! ส่งไปยังบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

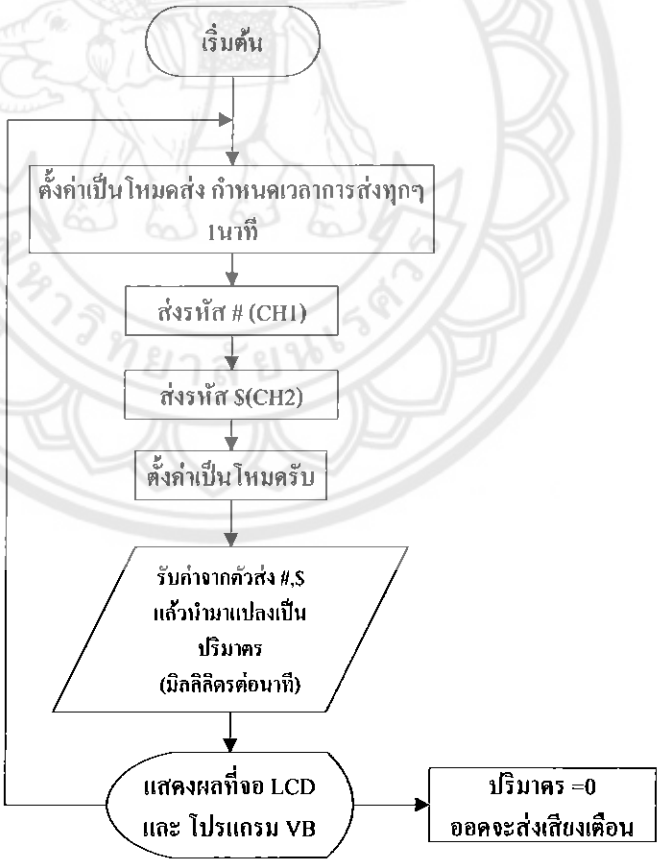
5. เมื่อมีการส่งข้อมูลครบถ้วนแล้ว จะทำการรีเซตค่า Timer 0 และ Interrupt INT0 เท่ากับ 0 เพื่อนับค่าหยดน้ำเกลือครั้งต่อไป

3.2.2 การออกแบบโปรแกรมบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

การออกแบบขั้นตอนการทำงานของโปรแกรมเป็นการออกแบบโปรแกรมเพื่อควบคุมการทำงานของบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) เพื่อให้สามารถส่งสัญญาณข้อมูลออกไปได้อย่างถูกต้อง มีขั้นตอนดังนี้

การทำงานของบอร์ดไมโครคอนโทรเลอร์ตัวรับ (Atmega32)

แผนการทำงานของบอร์ดไมโครคอนโทรเลอร์ตัวรับ (Atmega32) ดังรูปที่ 3.8 สามารถอธิบายการทำงานได้ดังนี้



รูปที่ 3.8 แผนภาพแสดงการเชื่อมต่อวงจรของตัวรับ (Atmega32)

1. เริ่ม start เข้ามา บอร์ดไมโครคอนโทรเลอร์ตัวรับ (Atmega32) จะตั้งค่าเป็นโหมดส่ง

2. ส่งรหัสประจำเครื่องทุกๆ 1 นาที ได้แก่ # ซึ่งเป็นรหัสของบอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) ตัวที่ 1 และ $ ซึ่งเป็นรหัสของบอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) ตัวที่ 2

3. บอร์ดไมโครคอนโทรเลอร์ตัวรับ (Atmega32) ตั้งค่าเป็นโหมดรับ

4. รับค่าจากไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) แล้วนำค่าปริมาตรที่ได้มาแปลงเป็นหน่วยมิลลิลิตรต่อ 10 วินาที จากสมการ

$$\text{ปริมาตร}(ml) = \frac{\text{จำนวนหยด}}{20}$$

5. นำค่ามาแสดงผลที่จอ LCD และโปรแกรม Visual Basic 6.0 ซึ่งค่าที่ได้มีหน่วยเป็นหน่วยมิลลิลิตรต่อ 10 วินาที จากนั้นทำการรวมค่าปริมาตรที่ได้จนครบ 6 ครั้ง (60 วินาที) มาแสดงผลทุกๆ 1 นาที เนื่องจากได้กำหนดให้การส่งรหัสประจำเครื่องเป็นไปตามลำดับดังนี้ ตัวแรกคือรหัส # และตัวที่สองคือรหัส $ ค่าที่ได้รับจากบอร์ดไมโครคอนโทรเลอร์ตัวส่ง (Attiny2313) ทั้งสองก็จะแสดงผลตามลำดับ จากนั้นโปรแกรม Visual Basic 6.0 จะทำการตัดเฉพาะค่าตัวเลขมาประมวลผลและแสดงผล

6. เมื่อปริมาตรมีค่าเท่ากับ 0 ออดจะส่งสัญญาณเตือน

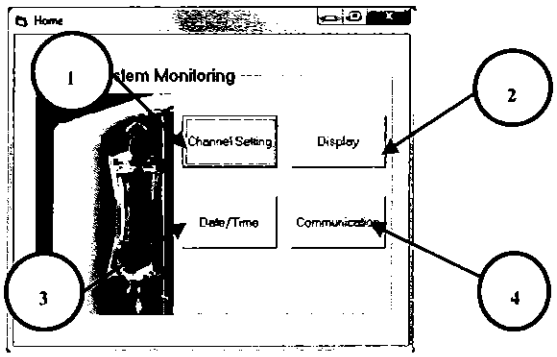
3.3 โปรแกรมการแสดงผล

ในการออกแบบการแสดงผลโดยใช้โปรแกรม Visual Basic 6.0 แบ่งออกเป็น 2 ส่วน คือ

1. ส่วนสำหรับการตั้งค่าการทำงานของอุปกรณ์

2. ส่วนแสดงผลข้อมูล

โดยมีหน้าต่างหลักแสดงดังรูปที่ 3.9



รูปที่ 3.9 รูปร่างหน้าต่างหลักของอุปกรณ์

จากรูปที่ 3.9 สามารถอธิบายรายละเอียดของแต่ละตำแหน่งได้ ดังนี้

ตำแหน่งที่ 1 คือ ปุ่ม Channel Setting ใช้แสดงหน้าต่างในการตั้งค่าต่างๆ

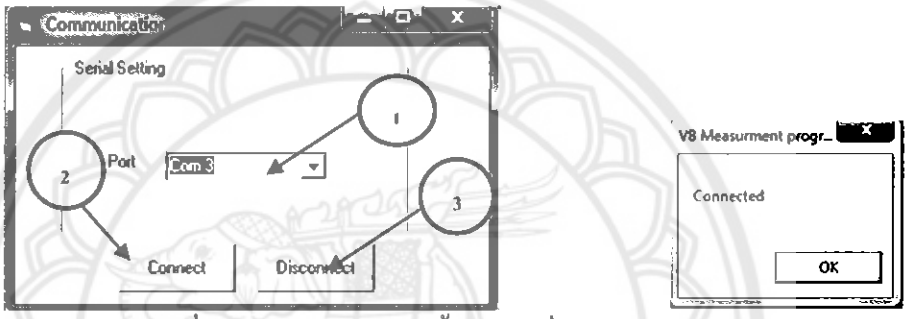
ตำแหน่งที่ 2 คือ ปุ่ม Display ใช้แสดงหน้าต่างในการแสดงผลของข้อมูล

ตำแหน่งที่ 3 คือ ปุ่ม Date/Time ใช้แสดงหน้าต่างวันที่และเวลา

ตำแหน่งที่ 4 คือ ปุ่ม Communication ใช้แสดงหน้าต่างในการตั้งค่าการเชื่อมต่อ

3.3.1 ปุ่ม Communication

ปุ่ม Communication มีรูปร่างหน้าต่างตั้งค่าการเชื่อมต่อของอุปกรณ์ดังรูปที่ 3.10



รูปที่ 3.10 รูปร่างหน้าต่างตั้งค่าการเชื่อมต่อของอุปกรณ์

จากรูปที่ 3.10 สามารถอธิบายรายละเอียดของแต่ละตำแหน่งได้ ดังนี้

ตำแหน่งที่ 1 คือ ข้อความใช้กำหนดพอร์ตและอัตราการเชื่อมต่อ

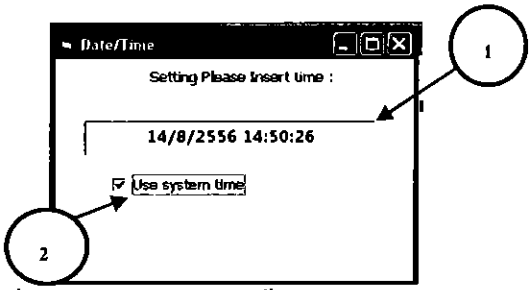
ตำแหน่งที่ 2 คือ ปุ่ม Connect ใช้กำหนดในการเชื่อมต่อระหว่าง PC กับไมโครคอนโทรลเลอร์

ตำแหน่งที่ 3 คือ ปุ่ม Disconnect ใช้เมื่อไม่ต้องการเชื่อมต่อระหว่าง PC กับไมโครคอนโทรลเลอร์

หน้าต่าง VB Measurement program เป็นหน้าต่างที่ใช้แสดงว่ามีการเชื่อมต่อระหว่าง PC กับไมโครคอนโทรลเลอร์แล้ว

3.3.2 ปุ่ม Date/Time

ปุ่ม Date/Time มีรูปร่างหน้าต่างตั้งค่าเวลาของอุปกรณ์ดังรูปที่ 3.11



รูปที่ 3.11 รูปร่างหน้าต่างตั้งค่าเวลาของอุปกรณ์

จากรูปที่ 3.11 สามารถอธิบายรายละเอียดของแต่ละตำแหน่งได้ ดังนี้

ตำแหน่งที่ 1 คือ กรอบข้อความใช้กำหนดวันที่และเวลา

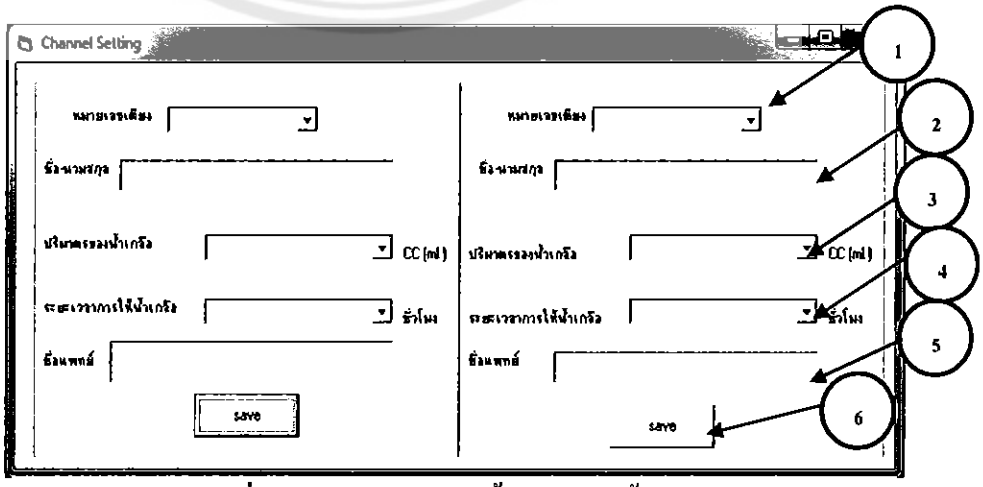
ตำแหน่งที่ 2 คือ กล่องตัวเลือกใช้เลือกเมื่อต้องการให้เวลาตรงกับเวลาในคอมพิวเตอร์

ตารางที่ 3.2 ตัวอย่าง Code ของตัวเลือก Use system time

Private Sub Check1_change()
txtTime = Now
End Sub
Private Sub Check1_Click()
If Check1 = 1 Then
txtTime = Now
Else
txtTime = "0/0/0000 0:00:00 AM"
End If
End Sub
Private Sub Form_Load()
txtTime = "0/0/0000 0:00:00 AM"
End Sub
	ใช้งานฟังก์ชั่นเมื่อมีการเลือกข้อมูล ถ้ามีการเลือกของข้อมูลที่แสดง โดยจะทำการตรวจสอบข้อมูลว่าตรงตามเงื่อนไขหรือไม่และเมื่อมีการเลือกจะทำการแสดงเวลาที่ตรงกับเวลาในคอมพิวเตอร์

3.3.3 ปุ่ม Channel Setting

ปุ่ม Channel Setting มีหน้าต่างตั้งค่าการให้น้ำเกลือ ดังรูปที่ 3.12



รูปที่ 3.12 รูปร่างหน้าต่างตั้งค่าการให้น้ำเกลือ

จากรูปที่ 3.12 สามารถอธิบายรายละเอียดของแต่ละตำแหน่งได้ ดังนี้

ตำแหน่งที่ 1 คือ กรอบข้อความใช้กำหนดหมายเลขเตียงผู้ป่วย

ตำแหน่งที่ 2 คือ กรอบข้อความชื่อผู้ป่วย

ตำแหน่งที่ 3 คือ กรอบข้อความใช้กำหนดปริมาตรของน้ำเกลือที่จะให้กับผู้ป่วย ซึ่งแพทย์เป็นผู้กำหนด

ตำแหน่งที่ 4 คือ กรอบข้อความใช้กำหนดระยะเวลาในการให้น้ำเกลือกับผู้ป่วย ซึ่งแพทย์เป็นผู้กำหนด

ตำแหน่งที่ 5 คือ กรอบข้อความชื่อแพทย์ผู้รักษา

ตำแหน่งที่ 6 คือ ปุ่ม save ใช้บันทึกข้อมูลที่ต้องการกำหนด

ตารางที่ 3.3 ตัวอย่าง Code ของปุ่ม Save

Private saveme1, saveme2 As Integer Private Sub cmdSave2_Click() saveme1 = 0 saveme2 = 1 If Combo5.Text = "" Or Combo6.Text = "" Or Combo7.Text = "" Then MsgBox "Please completed form before save!!"Else Checkrepeated End If End Sub	ใช้งานฟังก์ชั่นเมื่อมีการกดปุ่ม cmdSave2 - กำหนดให้ค่า saveme2 มีค่าเท่ากับ 1 และมีการตรวจสอบว่าข้อความถูกกรอกครบทุกช่อง

Private Sub cmdSave1_Click()
saveme1 = 1
saveme2 = 0
If Combo1.Text = "" Or Combo2.Text = "" Or Combo3.Text = "" Then
MsgBox "Please completed form before save!!" Else
Call checkrepeated
End If
End Sub
	ใช้งานฟังก์ชั่นเมื่อมีการกดปุ่ม cmdSave1
	- กำหนดให้ค่า saveme1 มีค่าเท่ากับ 1 และมีการตรวจสอบว่าข้อความถูกกรอกครบทุกช่อง
	เรียกฟังก์ชั่น checkrepeated

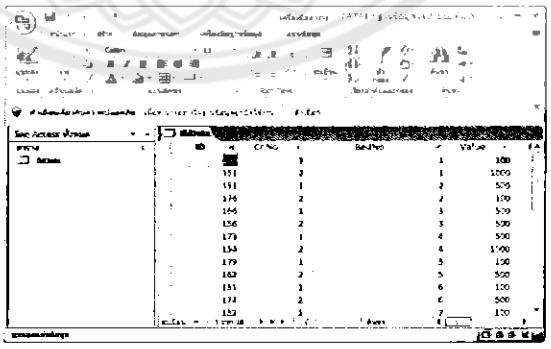
ตารางที่ 3.3 ตัวอย่าง Code ของปุ่ม Save (ต่อ)

Private Sub checkrepeated()	ใช้งานฟังก์ชั่น checkrepeated
Dim checkbednow, noch As Integer	
If saveme1 = 1 Then	ถ้าค่า saveme1 มีค่าเท่ากับ 1,noch = 1
checkbednow = Val(Combo1.Text)	
noch = 1	
ElseIf saveme2 = 1 Then	ถ้าค่า saveme2 มีค่าเท่ากับ 1,noch = 2
checkbednow = Val(Combo5.Text)	กำหนดการเก็บค่า
noch = 2	
End If Sub Save1()	
If saveme1 = 1 Then	ใช้งานฟังก์ชั่น Save1
Statement = "SELECT * FROM tblData ORDER BY ID"	กำหนด FROM tblData ORDER BY ID เป็นฟอร์มในการเก็บค่า
rs.Open Statement, ConnDB, adOpenKeyset, adLockOptimistic, adCmdText	
rs.AddNew	กำหนด
rs("ChNo") = 1	- ChNo คือ channel 1
rs("BedNo") = Combo1.Text	- BedNo คือลำดับเตียง
rs("Value") = Combo2.Text	- Rank คือ ระยะเวลาการให้น้ำเกลือ
rs("Rank") = Combo3.Text	- Name คือ ชื่อผู้ป่วย
rs("Name") = Text1.Text	- Doctor คือ ชื่อแพทย์ผู้รักษา
rs("Doctor") = Text3.Text	- DateModify คือ เวลา
rs("DateModify") = Now()	- Value คือ ปริมาตรของน้ำเกลือ
End If	
If saveme2 = 1 Then	
Statement = "SELECT * FROM tblData ORDER BY ID"	
rs.Open Statement, ConnDB, adOpenKeyset, adLockOptimistic, adCmdText	

ตารางที่ 3.3 ตัวอย่าง Code ของปุ่ม Save (ต่อ)

rs.AddNew	
rs("ChNo") = 2	
rs("BedNo") = Combo5.Text	
rs("Value") = Combo6.Text	
rs("Rank") = Combo7.Text	กำหนด - ChNo คือ channel 2
rs("Name") = Text1.Text	- BedNo คือลำดับเตียง
rs("Doctor") = Text3.Text	- Value คือ ปริมาตรของน้ำเกลือ
rs("DateModify") = Now()	- Rank คือ ระยะเวลาการให้น้ำเกลือ
End If	- Name คือ ชื่อผู้ป่วย
rs.Update	- Doctor คือ ชื่อแพทย์ผู้รักษา
rs.Close: Set rs = Nothing	- DateModify คือ เวลา
MsgBox "Data already saved",	
vbOKOnly + vbInformation, "Status	อัพเดตข้อมูลตลอดเวลา
Report"	
Call cleardata	
End Sub	

เมื่อมีการกดปุ่ม save ที่หน้าต่าง Channel setting จะมีการบันทึกข้อมูลลงในฐานข้อมูล ดังตัวอย่างไฟล์ในรูปที่ 3.13



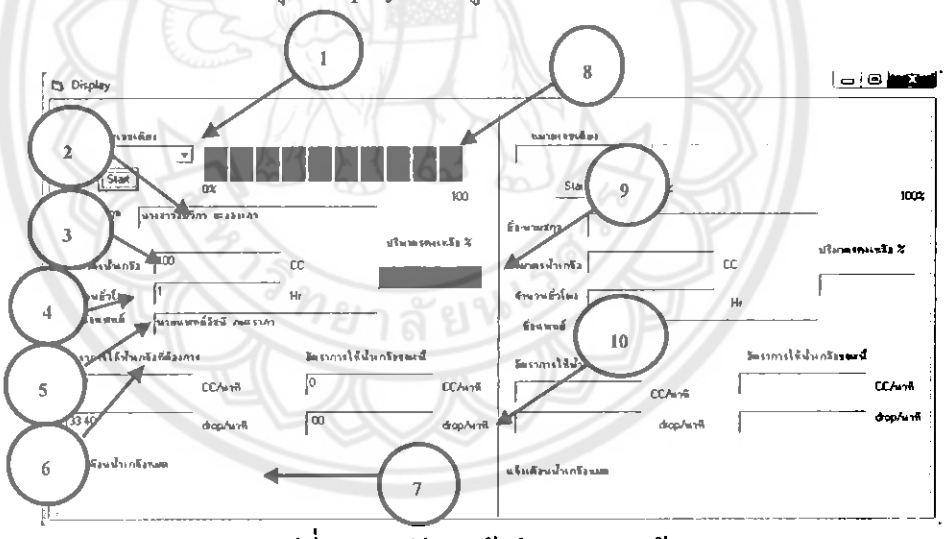
รูปที่ 3.13 ตัวอย่างไฟล์ฐานข้อมูลในการจัดเก็บข้อมูลจากหน้าต่าง setting

3.3.4 ปุ่ม Display

ในการออกแบบและพัฒนาโปรแกรม Visual Basic 6.0 มีลำดับขั้นตอนการทำงานโปรแกรมที่ใช้ในการแสดงผลสัญญาณ ดังนี้

1. รับข้อมูลที่ได้จากไมโครคอนโทรเลอร์ตัวรับ (Atmega32)
2. แสดงผลข้อมูล ดังนี้
   - 2.1 เป็นการแสดงผลข้อมูลออกมาเป็นตามข้อมูลจริง
   - 2.2 อัตราการให้น้ำเกลือที่ต้องการ (หยดต่อนาที) และ (มิลลิลิตรต่อนาที)
   - 2.3 อัตราการให้น้ำเกลือ ณ เวลาขณะนั้น (หยดต่อนาที) และ (มิลลิลิตรต่อนาที)
   - 2.4 ปริมาตรคงเหลือของน้ำเกลือ
   - 2.5 การแจ้งเตือนเมื่อน้ำเกลือที่ให้ใกล้หมด
   - 2.6 ชื่อผู้ป่วย
   - 2.7 ชื่อแพทย์ผู้รักษา

ในการออกแบบรูปร่างหน้าต่างของจอมอนิเตอร์ตั้งค่าต่างๆ ของระบบแสดงอัตราการให้น้ำเกลือแบบไร้สายเมื่อกดปุ่ม Display ดังในรูปที่ 3.14



รูปที่ 3.14 รูปร่างหน้าต่างแสดงผลข้อมูล

จากรูปที่ 3.14 สามารถอธิบายรายละเอียดของแต่ละตำแหน่งได้ ดังนี้

ตำแหน่งที่ 1 คือ กรอบข้อความที่ใช้แสดงผลของหมายเลขเตียงของผู้ป่วย

ตำแหน่งที่ 2 คือ กรอบข้อความที่ใช้แสดงชื่อผู้ป่วย

ตำแหน่งที่ 3 คือ กรอบข้อความที่ใช้แสดงผลของปริมาตรน้ำเกลือที่ผู้ป่วยต้องใช้

ตำแหน่งที่ 4 คือ กรอบข้อความที่ใช้แสดงผลของจำนวนชั่วโมงในการให้น้ำเกลือแก่ผู้ป่วย

ตำแหน่งที่ 5 คือ กรอบข้อความที่ใช้แสดงชื่อแพทย์ผู้รักษา

ตำแหน่งที่ 6 คือ ผลของอัตราการให้น้ำเกลือที่ผู้ป่วยต้องการ

ตำแหน่งที่ 7 คือ การแจ้งเตือนของน้ำเกลือที่เหลืออีก 10 % และหมดแล้ว

ตำแหน่งที่ 8 คือ กราฟแสดงผลการลดลงของปริมาตรน้ำเกลือ

ตำแหน่งที่ 9 คือ กรอบข้อความที่ใช้แสดงผลของปริมาตรคงเหลือที่เป็นเปอร์เซ็นต์

ตำแหน่งที่ 10 คือ กรอบข้อความที่ใช้แสดงผลของอัตราการให้น้ำเกลือในขณะนี้

ตารางที่ 3.4 ตัวอย่าง Code ของการแจ้งเตือนน้ำเกลือหมด

Private Sub Form_Load() Label18.Caption = "frmCommu.Myinputdata1" Label19.Caption = "frmCommu.Myinputdata2"	เป็นการกำหนดรับค่าจากบอร์ดไมโครคอนโทรเลอร์
NewA = Val(Label18.Caption) NewB = Val(Label19.Caption) Timer1.Enabled = True Text2.Text = "" & rs("Value") Rateflow = "" & rs("Rank"),Text5.Text = Rateflow Text3.Text = FormatNumber((Text2.Text / (Rateflow * 60)), 3) Text4.Text = Datach2 Text6.Text = FormatNumber((Datach2 * 20), 3) Text1.Text = ((Text2.Text - (Val(NewA))) / Text2.Text) * 100 Text13.Text = FormatNumber((Text3.Text * 20), 3)	เป็นการดึงค่าที่ฐานข้อมูลที่ได้กำหนดในหน้าต่าง setting มาแสดง โดยดึงข้อมูลต่าง ๆ มาแสดง และนำมาคำนวณค่าก่อนนำค่ามาแสดง เช่น อัตราการให้น้ำเกลือที่ผู้ป่วยต้องการอัตราการให้น้ำเกลือในขณะนี้ปริมาตรคงเหลือที่เป็นเปอร์เซ็นต์
If Text1.Text = 0 Then Command1.Visible = True MsgBox "Please refill the saline!!"	ปุ่มนี้เป็นปุ่ม total = 0 % ที่ใช้แสดงว่าน้ำเกลือหมด ให้นำน้ำเกลือขวดใหม่มาเปลี่ยน
ElseIf Text1.Text = 10 Then Command5.Visible = True MsgBox "the saline total 10%!!" End If	ปุ่มนี้เป็นปุ่ม total = 10 % ที่ใช้แสดงน้ำเกลือก่อนที่จะหมดในขณะที่น้ำเกลือขณะนี้เหลืออยู่ 10 %จำนวนชั่วโมงในการให้น้ำเกลือแก่ผู้ป่วย

3.4 วิธีการทดสอบ

เพื่อตรวจสอบฟังก์ชั่นการทำงานของระบบแสดงอัตราการให้น้ำเกลือแบบไร้สายที่พัฒนาขึ้นในด้านซอฟแวร์ จึงมีความจำเป็นต้องทดสอบประสิทธิภาพการทำงานของเครื่องวัดหยดน้ำเกลือและส่งข้อมูลแบบไร้สายที่พัฒนาขึ้นมา ในหัวข้อย่อยนี้จะกล่าวถึงการออกแบบการทดสอบอุปกรณ์ดังกล่าว 2 ส่วนหลัก คือ การทดสอบส่วนของซอฟแวร์ และการทดสอบการใช้งานของอุปกรณ์

3.4.1 การทดสอบส่วนของซอฟแวร์

เริ่มต้นด้วยการเขียนรหัสโปรแกรมภาษาซีที่ใช้สำหรับการรับส่งข้อมูลระหว่างบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) กับบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) และเขียนรหัสโปรแกรมภาษาซีที่ใช้สำหรับการรับส่งข้อมูลระหว่างบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) กับคอมพิวเตอร์ส่วนบุคคล โดยใช้โปรแกรมไฮเปอร์เทอร์มินอลแสดงผล ดังรูปที่ 3.15

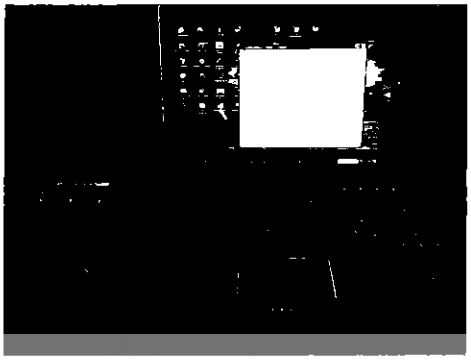
รูปที่ 3.15 ตัวอย่างหน้าต่างโปรแกรมไฮเปอร์เทอร์มินอล

เมื่อเชื่อมต่อระหว่างบอร์ดไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ได้แล้ว จึงออกแบบการเขียนโปรแกรม Visual Basic 6.0 เพื่อทำการแสดงผลข้อมูล จากนั้นทำการทดสอบโปรแกรมขั้นสุดท้ายด้วยการต่อบอร์ดไมโครคอนโทรลเลอร์เข้ากับคอมพิวเตอร์ส่วนบุคคล เพื่อแสดงผลสัญญาณผ่านทางหน้าต่างจอมอนิเตอร์ของคอมพิวเตอร์ส่วนบุคคล

3.4.2 การทดสอบการใช้งานอุปกรณ์

เมื่อทำการทดสอบในส่วนของซอฟแวร์เสร็จเรียบร้อยแล้ว จึงนำเครื่องวัดหยดน้ำเกลือและโปรแกรมที่พัฒนาขึ้นมาทั้งหมดมาทำการทดสอบ ดังนี้

ก. การทดสอบการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์



รูปที่ 3.16 แสดงการทดสอบการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์

ขั้นตอนที่ 1 ทำการเชื่อมต่อบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) กับคอมพิวเตอร์ส่วนบุคคล โดยใช้พอร์ตยูเอสบี ซึ่งใช้สายที่แปลงสัญญาณจาก RS 232 เป็นยูเอสบีนำไปต่อเข้ากับคอมพิวเตอร์ส่วนบุคคล

ขั้นตอนที่ 2 ต่อบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) กับแหล่งจ่ายไฟ กดสวิตช์เปิดเครื่องทั้งบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) และบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

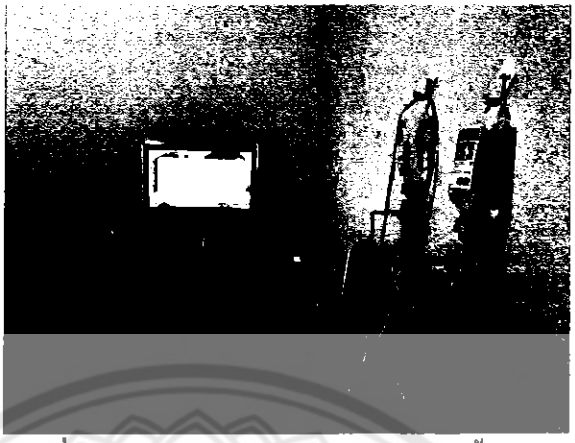
ขั้นตอนที่ 3 เปิดหน้าต่างโปรแกรมไฮเปอร์เทอร์มินอล ตั้งค่าการเชื่อมต่อที่คอมพอร์ต 2 (โดยจะกำหนดบอร์ดเรทเท่ากับ 9600) เพื่อให้อัตราการรับและส่งข้อมูลระหว่างบอร์ดไมโครคอนโทรเลอร์กับคอมพิวเตอร์ส่วนบุคคลมีค่าเท่ากัน

ขั้นตอนที่ 4 กดปุ่มสีเหลืองที่บอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) เพื่อให้เครื่องเริ่มทำงาน ถ้ามีการเชื่อมต่อระหว่างบอร์ดไมโครคอนโทรลเลอร์ของตัวรับ (Atmega32) กับบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ข้อมูลที่มีการเชื่อมต่อแล้วจะแสดงผลออกมาทางหน้าจอแอลซีดี

ขั้นตอนที่ 5 ถ้าบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) สามารถเชื่อมต่อกับคอมพิวเตอร์ส่วนบุคคลได้ ตัวไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) จะทำการส่งข้อมูลของบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) มาแสดงที่หน้าต่างของโปรแกรมไฮเปอร์เทอร์มินอลได้

การทดลองนี้เป็นการทดสอบการเชื่อมต่อระหว่างบอร์ดไมโครคอนโทรลเลอร์เข้ากับคอมพิวเตอร์ การทดสอบนี้เพื่อตรวจสอบประสิทธิภาพของบอร์ดไมโครคอนโทรลเลอร์

ข. การทดสอบการนับหยดน้ำเกลือ



รูปที่ 3.17 แสดงการทดสอบการนับหยดน้ำเกลือ

ขั้นตอนที่ 1 ต่อบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) กับแหล่งจ่ายไฟ จากนั้นต่อสายน้ำเกลือเข้ากับขวดน้ำเกลือ โดยในโครงงานนี้ใช้เซตน้ำเกลือแบบ macrodrop เป็นแบบ 20 หยดต่อมิลลิลิตร

ขั้นตอนที่ 2 ทำการเชื่อมต่อบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) กับหลอดหยดสายน้ำเกลือ

ขั้นตอนที่ 3 ทำการเชื่อมต่อบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) กับคอมพิวเตอร์ส่วนบุคคล โดยใช้พอร์ตยูเอสบี ซึ่งสายที่ใช้แปลงสัญญาณจาก RS 232 เป็นยูเอสบี

ขั้นตอนที่ 4 กดสวิตช์เปิดเครื่องทั้งบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) และบอร์ดไมโครคอนโทลเลอร์ ตัวรับ (Atmega32)

ขั้นตอนที่ 5 เปิดหน้าต่างหลักของโปรแกรมแสดงผล Visual Basic 6.0

- กดปุ่ม Communication จะแสดงหน้าต่างเพื่อตั้งค่าในการเชื่อมต่อ ทำการเชื่อมต่อที่พอร์ต 2 ถ้ามีการเชื่อมต่อที่ถูกต้องจะมีหน้าต่าง VB Measurment program แสดงข้อความว่า "Connected"
- กดปุ่ม Date/Time เพื่อตั้งค่าวันที่และเวลาให้กับอุปกรณ์ ทำการเลือกที่ Use system time เพื่อตั้งค่าวันที่และเวลาให้ตรงกับระบบเวลาในเครื่อง
- กดปุ่ม Channel Setting เพื่อตั้งค่าข้อมูลของผู้ป่วยในการให้น้ำเกลือ โดยตั้งค่าการทำงาน โดยการแบ่งเป็น 3 การทดลองย่อย ดังนี้

1. การทดลองย่อยที่ 1 กำหนดอัตราการให้ 1 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร

2. การทดลองย่อยที่ 2 กำหนดอัตราการให้ 2 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร

3. การทดลองย่อยที่ 3 กำหนดอัตราการให้ 3 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร

กดปุ่ม save เพื่อเก็บข้อมูล

● กดปุ่ม Display เลือกหมายเลขเตียงให้ตรงกับที่ทำการตั้งค่าในหน้าต่าง Channel setting จากนั้นกดปุ่ม start ที่หน้าต่างแสดงผล display ทำการปรับสายน้ำเกลือเพื่อให้ค่าที่ช่องอัตราการให้น้ำเกลือขณะนี้ ตรงหรือใกล้เคียงกับค่าอัตราการให้น้ำเกลือที่ต้องการ

ขั้นตอนที่ 6 จับเวลาและนับหยดน้ำเกลือ โดยการแบ่งเป็น 3 การทดลองย่อย ดังนี้

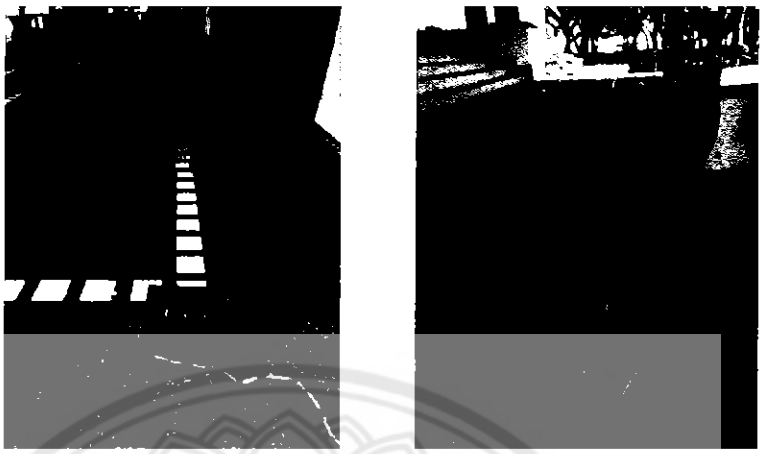
● การทดลองย่อยที่ 1 กำหนดอัตราการให้ 1 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร โดยแบ่งเป็น 6 ช่วงเวลา คือ ช่วงที่ 1: 5 – 10 นาที, ช่วงที่ 2: 15 - 20 นาที, ช่วงที่ 3: 25 - 30 นาที, ช่วงที่ 4: 35 - 40 นาที, ช่วงที่ 5: 45 - 50 นาทีและช่วงที่ 6: 55 - 60 นาที ทำการทดสอบทั้งหมด 10 ครั้งและจดบันทึกการทดสอบ

● การทดลองย่อยที่ 2 กำหนดอัตราการให้ 2 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร โดยแบ่งเป็น 6 ช่วงเวลา คือ ช่วงที่ 1: 5 – 10 นาที, ช่วงที่ 2: 25 - 30 นาที, ช่วงที่ 3: 45 - 50 นาที, ช่วงที่ 4: 75 - 80 นาที, ช่วงที่ 5: 95 - 100 นาทีและช่วงที่ 6: 115 - 120 นาที ทำการทดสอบทั้งหมด 10 ครั้งและจดบันทึกการทดสอบ

● การทดลองย่อยที่ 3 กำหนดอัตราการให้ 3 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร โดยแบ่งเป็น 6 ช่วงเวลา คือ ช่วงที่ 1: 25 – 30 นาที, ช่วงที่ 2: 55 - 60 นาที, ช่วงที่ 3: 85 - 90 นาที, ช่วงที่ 4: 115 - 120 นาที, ช่วงที่ 5: 145 - 150 นาทีและช่วงที่ 6: 175 - 180 นาที ทำการทดสอบทั้งหมด 10 ครั้ง และจดบันทึกการทดสอบ

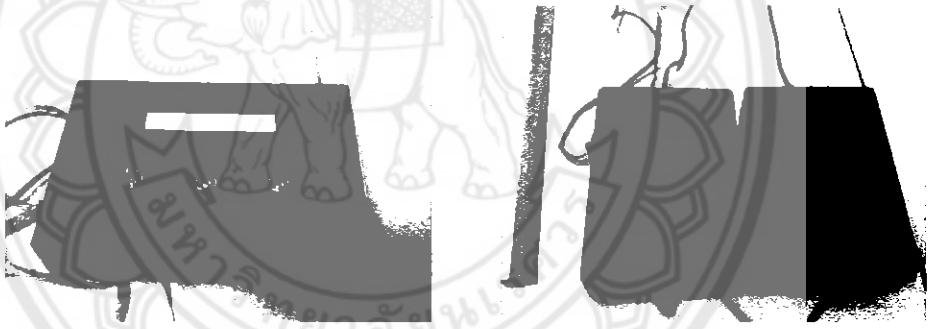
การทดลองนี้เป็นการทดสอบการนับหยดน้ำเกลือ เพื่อตรวจสอบประสิทธิภาพการวัดอัตราการไหลของน้ำเกลือ การนับจำนวนหยดของน้ำเกลือและตรวจสอบเกณฑ์ความคลาดเคลื่อนจากค่าความผิดพลาดการหยดของน้ำเกลือ

ค. การทดสอบระยะทางของการรับส่งข้อมูล



(ก) (ข)

รูปที่ 3.18 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับบริเวณ**พื้นที่โล่ง**
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งบริเวณ**พื้นที่โล่ง**



(ก) (ข)

รูปที่ 3.19 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับบริเวณ**พื้นที่ปิด**
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งบริเวณ**พื้นที่ปิด**

ขั้นตอนที่ 1 ต่อบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) กับแหล่งจ่ายไฟ

ขั้นตอนที่ 2 กดสวิตช์เปิดบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) และบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

ขั้นตอนที่ 3 กดปุ่มสีเหลืองที่บอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) เพื่อให้เครื่องทำงาน

ขั้นตอนที่ 4 ทดสอบการรับส่งข้อมูลระหว่างบอร์ดไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) โดยการแบ่งเป็น 2 การทดลองย่อย ดังนี้

● การทดลองย่อยที่ 1 ในบริเวณพื้นที่โล่ง เลื่อนตำแหน่งของบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) โดยกำหนดให้เพิ่มระยะครั้งละ 10 เมตร จนถึงระยะที่บอร์ดไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) ไม่สามารถรับส่งข้อมูลกันได้ และจดบันทึกการทดสอบ

●การทดลองย่อยที่ 2 ในบริเวณพื้นที่ปิด เลื่อนตำแหน่งของบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) โดยกำหนดให้เพิ่มระยะครั้งละ 5 เมตร จนถึงระยะที่บอร์ดไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) ไม่สามารถรับส่งข้อมูลกันได้ และจดบันทึกการทดสอบ

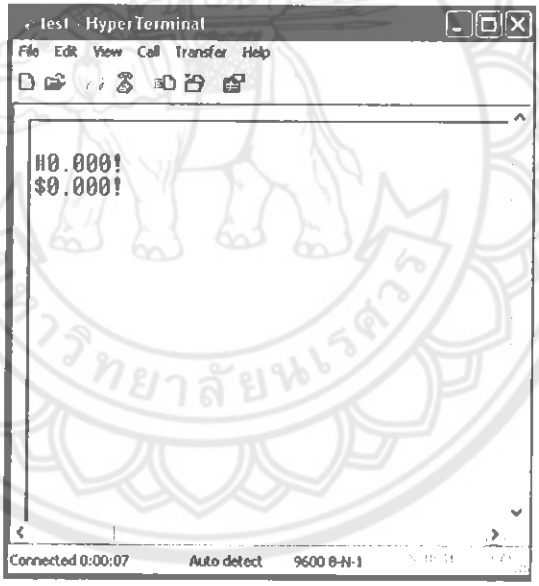
การทดลองนี้เป็นการทดสอบขอบเขตระยะทางของการรับส่งข้อมูลระหว่างบอร์ดไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) เพื่อตรวจสอบประสิทธิภาพ ขอบเขตระยะทางระหว่างบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) กับบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313)

บทที่ 4
ผลการทดลองและสรุปผลการทดลอง

บทนี้จะกล่าวถึงผลการทดลองและวิเคราะห์ผลการทดลองที่ได้รับจากการทดสอบประสิทธิภาพการทำงานของระบบแสดงอัตราการให้น้ำเกลือแบบไร้สาย ที่พัฒนาขึ้นตามที่กล่าวมาแล้วในบทที่ 3 ซึ่งแบ่งผลการทดลองและสรุปผลการทดลองออกเป็น 2 กรณี คือ ผลการทดลองส่วนซอฟแวร์ และผลการทดลองการใช้งานของอุปกรณ์เครื่องวัดหยดน้ำเกลือและส่งข้อมูลแบบไร้สาย

4.1 ผลการทดลองส่วนซอฟแวร์

4.1.1 โปรแกรมที่ใช้ทดสอบบอร์ดไมโครคอนโทรลเลอร์



รูปที่ 4.1 ผลการทดสอบการส่งข้อมูลของไมโครคอนโทรลเลอร์

พิจารณารูปที่ 4.1 เมื่อทำการเขียนโปรแกรมภาษาซีลงไปยังบอร์ดของไมโครคอนโทรลเลอร์ ผลการทดสอบการส่งข้อมูลของไมโครคอนโทรลเลอร์ โดยใช้โปรแกรมไฮเปอร์เทอร์มินอลในการแสดงผลการทดสอบของการส่งข้อมูล พบว่าไมโครคอนโทรลเลอร์ส่งข้อความ #0.000! และ $0.000! มายังโปรแกรมไฮเปอร์เทอร์มินอลซึ่งแสดงให้เห็นว่าโปรแกรมภาษาซีที่บันทึกลงในไมโครคอนโทรลเลอร์สามารถติดต่อสื่อสารมายังคอมพิวเตอร์ส่วนบุคคลได้

4.1.2 วิธีการใช้งานเครื่องวัดหยดน้ำเกลือสำหรับผู้ติดตั้งโปรแกรม

ในการใช้งานของเครื่องวัดหยดน้ำเกลือ ประกอบด้วย 3 ส่วน ดังนี้

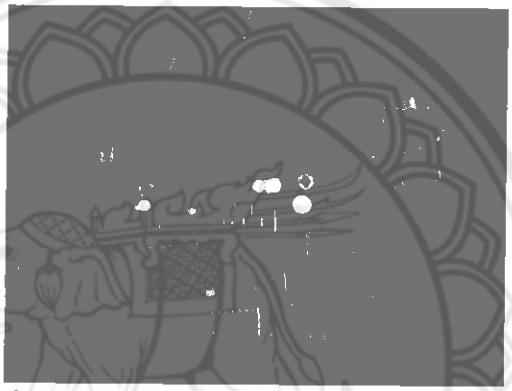
ก. ส่วนของการเชื่อมต่ออุปกรณ์

ก1. เครื่องวัดหยดน้ำเกลือในส่วนของตัวรับข้อมูล

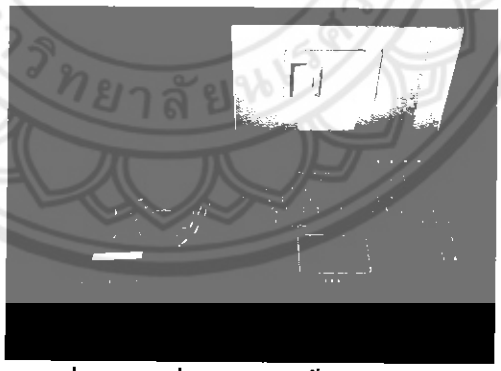
1. ทำการเชื่อมต่อบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) กับคอมพิวเตอร์ส่วนบุคคล โดยใช้พอร์ตยูเอสบี ซึ่งสายที่ใช้แปลงสัญญาณจาก RS 232 เป็นยูเอสบี

2. ต่อไฟเลี้ยง 5 โวลต์ให้ บอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32)

3. กดสวิทช์เปิด



รูปที่ 4.2 แสดงการเชื่อมต่อภายในของเครื่องวัดหยดน้ำเกลือในส่วนของตัวรับข้อมูล



รูปที่ 4.3 แสดงการเชื่อมต่อเครื่องวัดหยดน้ำเกลือในส่วนของตัวรับข้อมูล

ก2. เครื่องวัดหยดน้ำเกลือในส่วนของตัวส่งข้อมูล

1. ทำการเชื่อมต่อบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) กับหลอดหยดสายน้ำเกลือดังรูปที่ 4.4

2. ต่อไฟเลี้ยง 12 โวลต์ ให้กับบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313)

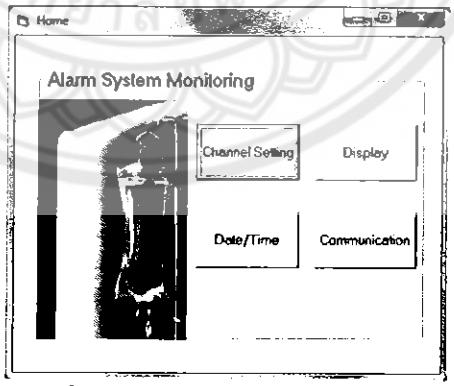
3. กดสวิทช์เปิด

รูปที่ 4.4 แสดงการเชื่อมต่อภายในของเครื่องวัดหยดน้ำเกลือในส่วนของตัวส่งข้อมูล

รูปที่ 4.5 แสดงการเชื่อมต่อเครื่องวัดหยดน้ำเกลือในส่วนของตัวส่งข้อมูล

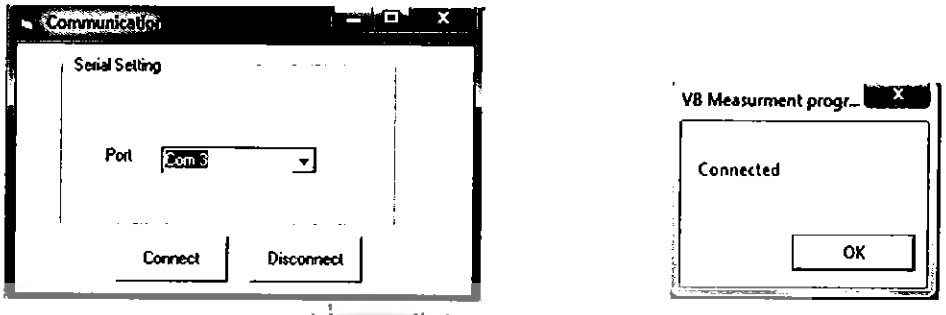
ข. ส่วนของการติดตั้งโปรแกรม

1. หลังจากลงโปรแกรมแล้ว เปิดหน้าต่างหลักของโปรแกรม ดังรูปที่ 4.6



รูปที่ 4.6 หน้าต่างหลักของโปรแกรม

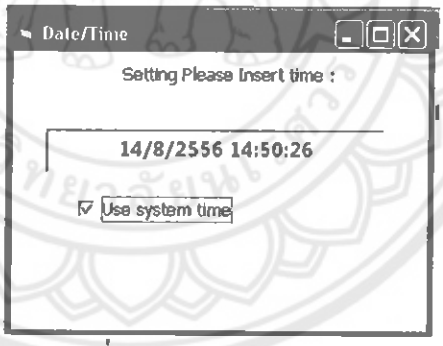
2. กดปุ่ม Communication จะแสดงหน้าต่างเพื่อตั้งค่าในการเชื่อมต่อดังรูปที่ 4.7 ทำการเชื่อมต่อครั้งแรกเพียงครั้งเดียว



รูปที่ 4.7 หน้าต่าง Communication

จากรูปที่ 4.7 ได้ทำการเชื่อมต่อที่พอร์ต 2 ถ้ามีการเชื่อมต่อที่ถูกต้องจะมีหน้าต่าง VB Measurment program แสดงข้อความว่า "Connected"

3. กดปุ่ม Date/Time จะแสดงหน้าต่างเพื่อตั้งค่าวันที่และเวลาให้กับอุปกรณ์ ดังรูปที่ 4.8 ทำการเชื่อมต่อครั้งแรกเพียงครั้งเดียว

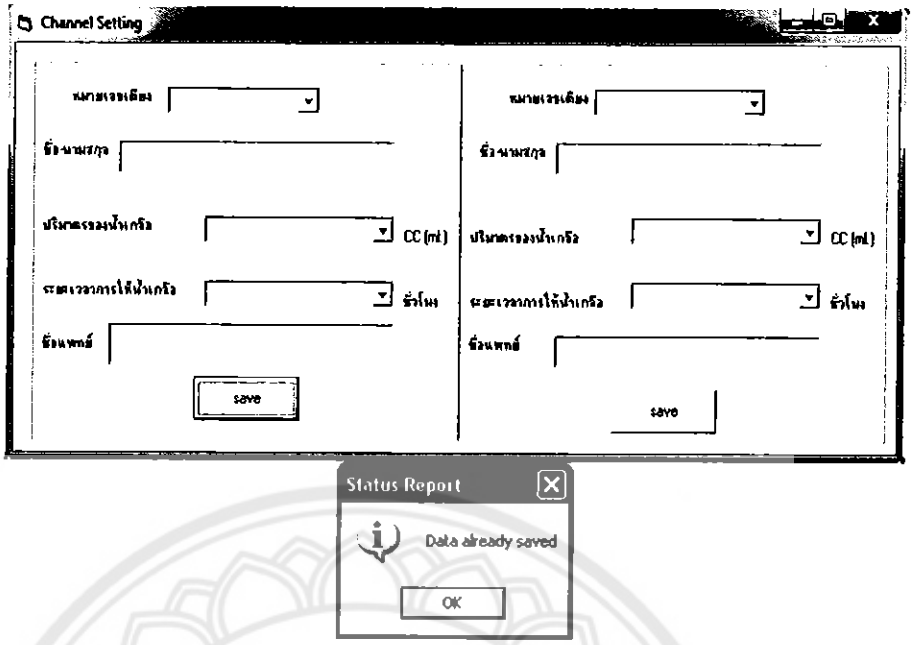


รูปที่ 4.8 หน้าต่าง Date/Time

จากรูปที่ 4.8 ทำการเลือกที่ Use system time เพื่อตั้งค่าวันที่และเวลาให้ตรงกับระบบเวลาในเครื่อง

ค. ส่วนของการตั้งค่าโปรแกรม

1. กดปุ่ม Channel Setting จะแสดงหน้าต่างเพื่อตั้งค่าอัตราการให้น้ำเกลือของผู้ป่วยในการให้น้ำเกลือ จากนั้นกดปุ่ม save ดังรูปที่ 4.9



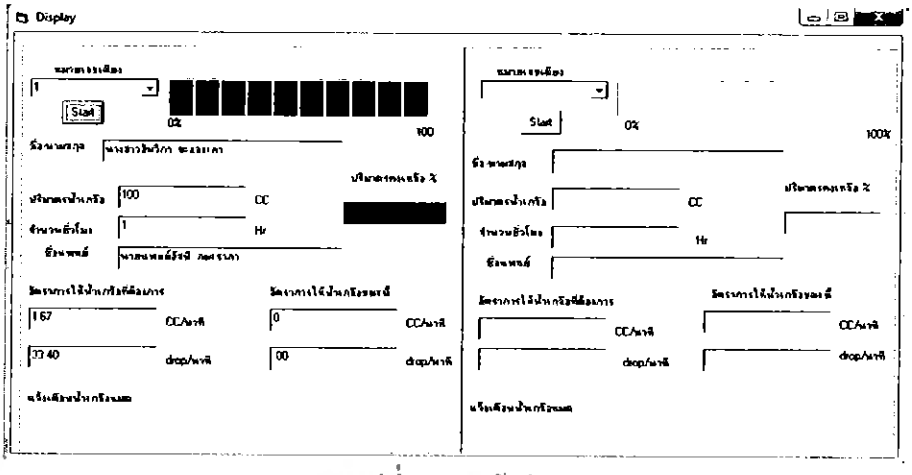
รูปที่ 4.9 หน้าต่าง Channel Setting

จากรูปที่ 4.9 ได้ทำการตั้งค่าข้อมูลของผู้ป่วย ดังนี้

ตารางที่ 4.1 ตัวอย่างการตั้งค่าข้อมูลของผู้ป่วย

หมายเลขเตียง	1	2
ชื่อ-นามสกุลผู้ป่วย	นางสาวอัมวิกา ละอองเภา	-
ปริมาตรของน้ำเกลือ	100	-
ระยะเวลาการให้น้ำเกลือ (ชั่วโมง)	1	-
แพทย์ผู้รักษา	นายแพทย์อัสนี ภมะราภา	-

2. กดปุ่ม Display จะแสดงหน้าต่าง จากนั้นเลือกหมายเลขเตียงให้ตรงกับที่ทำการตั้งค่าในหน้าต่าง Channel setting กดปุ่มสีเหลืองที่บอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) เพื่อให้บอร์ดไมโครคอนโทลเลอร์เริ่มทำงาน จากนั้นกดปุ่ม start หน้าต่าง Display จะแสดงข้อมูลและเริ่มการทำงาน



รูปที่ 4.10 หน้าต่าง Display

ค. ทำการปรับสายน้ำเกลือเพื่อให้ค่าที่ช่องอัตราการให้น้ำเกลือขณะนี้ ตรงหรือใกล้เคียงกับค่าอัตราการให้น้ำเกลือที่ต้องการ

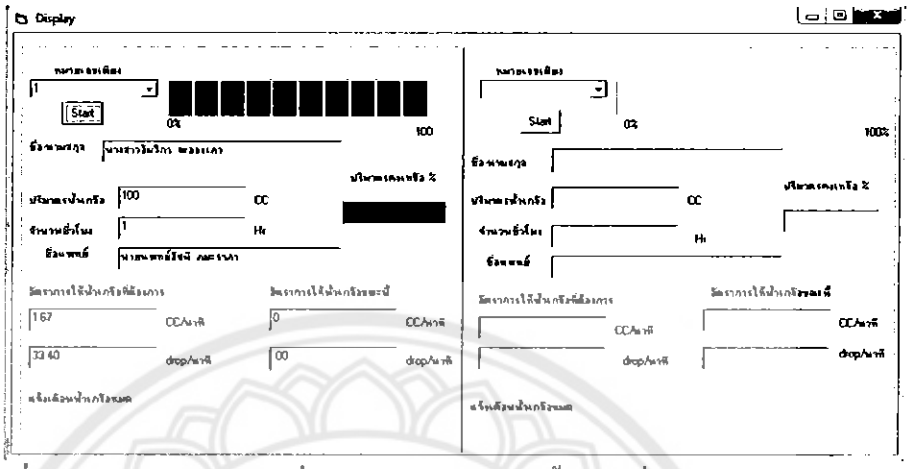
รูปที่ 4.11 แสดงการปรับสายน้ำเกลือให้ใกล้เคียงกับอัตราการให้น้ำเกลือที่ต้องการ

4.2 ผลการทดลองการใช้งานของอุปกรณ์

4.2.1 ผลการทดลองการแสดงผลอัตราการให้น้ำเกลือ

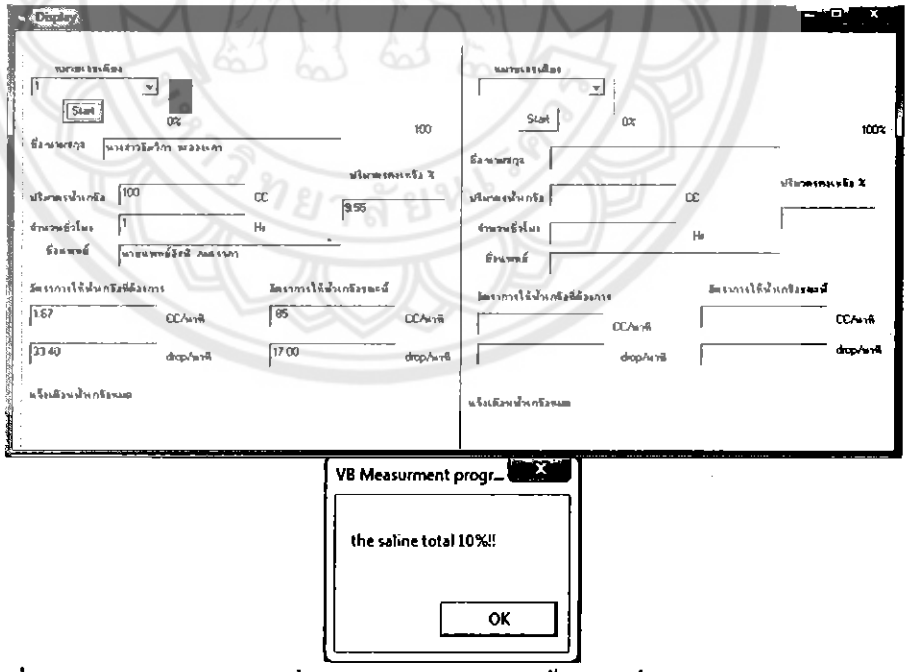
ผลการทดลองดังต่อไปนี้แสดงหน้าต่าง Display ที่แสดงผลอัตราการให้น้ำเกลือที่มีตั้งค่าเช่นเดียวกับตัวอย่างตารางที่ 4.1 การตั้งค่าข้อมูลของผู้ป่วยที่มีปริมาตรคงเหลือเท่ากับ 100 %, 9.55 %, 0 % ดังนี้

หน้าต่าง Display แสดงผลอัตราการให้น้ำเกลือที่มีปริมาตรคงเหลือเท่ากับ 100 % ดังรูปที่ 4.12



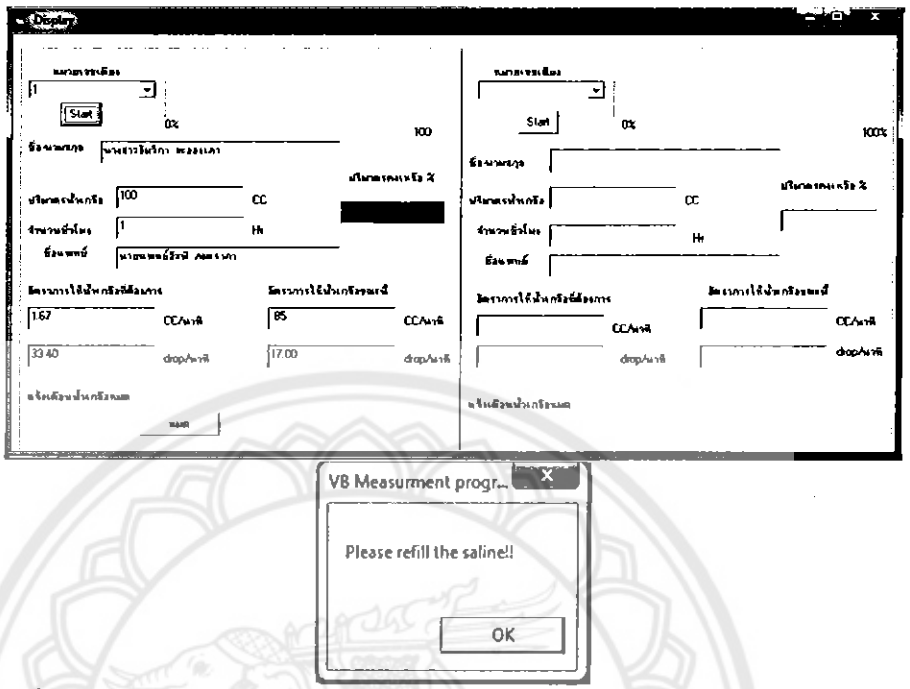
รูปที่ 4.12 หน้าต่าง Display ที่แสดงผลอัตราการให้น้ำเกลือที่มีปริมาตรคงเหลือ 100 %

หน้าต่าง Display แสดงผลอัตราการให้น้ำเกลือที่มีปริมาตรคงเหลือเท่ากับ 9.55 % ดังรูปที่ 4.13



รูปที่ 4.13 หน้าต่าง Display ที่แสดงผลอัตราการให้น้ำเกลือที่มีปริมาตรคงเหลือ 9.55 %

หน้าต่าง Display แสดงผลอัตราการให้น้ำเกลือที่มีปริมาตรคงเหลือเท่ากับ 0 % ดังรูปที่ 4.14



รูปที่ 4.14 หน้าต่าง Display ที่แสดงผลอัตราการให้น้ำเกลือที่มีปริมาตรคงเหลือ 0 %

4.2.2 ผลการทดลองจากการทดสอบการนับหยดน้ำเกลือ

ผลการทดลองดังต่อไปนี้แสดงตารางจากผลการทดสอบนับหยดน้ำเกลือ เพื่อตรวจสอบค่าความผิดพลาดจากการหยดของน้ำเกลือ ดังนี้

ก. การทดลองย่อยที่ 1 กำหนดระยะเวลาการให้ 1 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร ดังนั้นอัตราการให้น้ำเกลือ เท่ากับ 1.67 ml/min หรือ 33.4 หยด/นาที โดยแบ่งเป็น 6 ช่วงเวลา คือ ช่วงที่ 1: 5 – 10 นาที, ช่วงที่ 2: 15 - 20 นาที, ช่วงที่ 3: 25 - 30 นาที, ช่วงที่ 4: 35 - 40 นาที, ช่วงที่ 5: 45 - 50 นาทีและช่วงที่ 6: 55 - 60 นาที ในแต่ละช่วงเวลา กำหนดระยะเวลาในการทดสอบ 5 นาที ซึ่งในการจับเวลาการทดสอบจะทำการนับหยดน้ำเกลือทุกๆ 1 นาที โดยทำการทดสอบทั้งหมด 10 ครั้ง และจดบันทึกการทดสอบ

ตารางที่ 4.2 แสดงผลการนับหยดน้ำเกลือครั้งที่ 1

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	30	30	30	30	30	30	30	30	30	30
2 (นาทีที่ 15 - 20)	30	30	30	30	30	29	29	29	29	29
3 (นาทีที่ 25 - 30)	30	30	30	30	30	29	29	29	29	30
4 (นาทีที่ 35 - 40)	30	30	30	30	30	30	30	30	30	30
5 (นาทีที่ 45 - 50)	30	30	30	30	30	30	30	30	29	29
6 (นาทีที่ 55 - 60)	30	30	30	30	30	29	29	29	29	29

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 1 : นับหยดจริง = 30 หยดต่อนาที (1.50 มิลลิลิตรต่อนาที)

: จอแสดงผล = 29.47 หยดต่อนาที (1.473 มิลลิลิตรต่อนาที)

ตารางที่ 4.3 แสดงผลการนับหยดน้ำเกลือครั้งที่ 2

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	30	30	30	30	30	29	29	29	29	29
2 (นาทีที่ 15 - 20)	30	30	30	30	30	29	29	29	30	30
3 (นาทีที่ 25 - 30)	30	30	30	30	30	30	30	29	29	29
4 (นาทีที่ 35 - 40)	30	30	30	30	30	29	29	29	29	29
5 (นาทีที่ 45 - 50)	30	30	30	28	28	30	30	30	28	28
6 (นาทีที่ 55 - 60)	28	28	30	30	30	27	27	28	28	28

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 2 : นับหยดจริง = 29.733 หยดต่อนาที (1.487 มิลลิลิตรต่อนาที)

: จอแสดงผล = 28.933 หยดต่อนาที (1.447 มิลลิลิตรต่อนาที)

ตารางที่ 4.4 แสดงผลการนับหยดน้ำเกลือครั้งที่ 3

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	30	30	30	30	30	28	28	29	29	29
2 (นาทีที่ 15 - 20)	30	28	30	30	30	29	29	29	29	29
3 (นาทีที่ 25 - 30)	30	30	30	30	30	29	29	29	29	29
4 (นาทีที่ 35 - 40)	30	30	29	30	30	30	30	30	30	30
5 (นาทีที่ 45 - 50)	30	30	30	30	30	30	30	30	30	28
6 (นาทีที่ 55 - 60)	30	30	30	30	30	28	28	28	29	29

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 3 : นับหยดจริง = 29.900 หยดต่อนาที (1.495 มิลลิลิตรต่อนาที)

: จอแสดงผล = 29.100 หยดต่อนาที (1.455 มิลลิลิตรต่อนาที)

ตารางที่ 4.5 แสดงผลการนับหยดน้ำเกลือครั้งที่ 4

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	30	30	30	30	30	29	29	29	30	30
2 (นาทีที่ 15 - 20)	30	30	30	30	30	30	30	30	30	30
3 (นาทีที่ 25 - 30)	30	30	30	30	30	30	30	30	30	30
4 (นาทีที่ 35 - 40)	30	30	30	30	30	30	30	30	30	30
5 (นาทีที่ 45 - 50)	30	30	30	30	30	29	29	29	29	29
6 (นาทีที่ 55 - 60)	30	30	30	30	30	29	29	29	30	30

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 4 : นับหยดจริง = 30 หยดต่อนาที (1.5 มิลลิลิตรต่อนาที)

: จอแสดงผล = 29.633 หยดต่อนาที (1.482 มิลลิลิตรต่อนาที)

ตารางที่ 4.6 แสดงผลการนับหยดน้ำเกลือครั้งที่ 5

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	30	30	30	29	28	30	30	30	28	28
2 (นาทีที่ 15 - 20)	28	29	30	30	30	27	27	28	28	28
3 (นาทีที่ 25 - 30)	30	30	30	30	30	28	28	29	29	29
4 (นาทีที่ 35 - 40)	30	30	30	30	30	29	29	30	30	30
5 (นาทีที่ 45 - 50)	30	30	30	30	30	29	29	30	30	30
6 (นาทีที่ 55 - 60)	30	30	30	30	30	30	30	30	30	30

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 5 : นับหยดจริง = 29.8 หยดต่อนาที (1.49 มิลลิลิตรต่อนาที)

: จอแสดงผล = 29.1 หยดต่อนาที (1.455 มิลลิลิตรต่อนาที)

ตารางที่ 4.7 แสดงผลการนับหยดน้ำเกลือครั้งที่ 6

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	30	30	30	30	30	29	29	29	30	30
2 (นาทีที่ 15 - 20)	30	30	30	30	30	30	30	30	30	30
3 (นาทีที่ 25 - 30)	30	30	30	30	30	30	30	30	29	29
4 (นาทีที่ 35 - 40)	30	30	30	30	30	29	29	29	29	29
5 (นาทีที่ 45 - 50)	30	30	30	30	30	29	29	29	30	30
6 (นาทีที่ 55 - 60)	30	30	30	30	30	30	30	29	29	29

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 6 : นับหยดจริง = 30 หยดต่อนาที (1.5 มิลลิลิตรต่อนาที)

: จอแสดงผล = 29.467 หยดต่อนาที (1.473 มิลลิลิตรต่อนาที)

ตารางที่ 4.8 แสดงผลการนับหยดน้ำเกลือครั้งที่ 7

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	30	29	30	30	30	30	30	30	30	30
2 (นาทีที่ 15 - 20)	29	30	30	30	30	29	30	30	30	30
3 (นาทีที่ 25 - 30)	29	30	30	30	30	30	30	30	30	30
4 (นาทีที่ 35 - 40)	30	30	30	30	30	29	30	30	30	30
5 (นาทีที่ 45 - 50)	30	30	30	29	30	30	30	30	29	29
6 (นาทีที่ 55 - 60)	30	30	30	30	30	29	29	30	30	30

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 7 : นับหยดจริง = 29.867 หยดต่อนาที (1.494 มิลลิลิตรต่อนาที)

: จอแสดงผล = 29.8 หยดต่อนาที (1.49 มิลลิลิตรต่อนาที)

ตารางที่ 4.9 แสดงผลการนับหยดน้ำเกลือครั้งที่ 8

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	28	28	30	30	30	28	29	30	30	30
2 (นาทีที่ 15 - 20)	30	30	30	30	30	30	30	30	30	30
3 (นาทีที่ 25 - 30)	30	29	30	30	30	29	30	30	30	30
4 (นาทีที่ 35 - 40)	30	29	29	30	30	29	30	30	30	30
5 (นาทีที่ 45 - 50)	29	30	30	30	30	30	30	30	30	30
6 (นาทีที่ 55 - 60)	30	30	30	30	30	30	30	30	30	30

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 8 : นับหยดจริง = 29.733 หยดต่อนาที (1.487 มิลลิลิตรต่อนาที)

: จอแสดงผล = 29.833 หยดต่อนาที (1.492 มิลลิลิตรต่อนาที)

ตารางที่ 4.10 แสดงผลการนับหยดน้ำเกลือครั้งที่ 9

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	30	30	30	30	30	29	30	30	30	30
2 (นาทีที่ 15 - 20)	30	30	30	30	30	29	30	30	30	30
3 (นาทีที่ 25 - 30)	30	30	30	30	30	30	30	30	30	29
4 (นาทีที่ 35 - 40)	30	30	30	30	30	30	30	30	29	29
5 (นาทีที่ 45 - 50)	30	30	30	30	30	30	30	30	30	30
6 (นาทีที่ 55 - 60)	30	30	30	30	30	30	30	30	30	30

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 9 : นับหยดจริง = 30 หยดต่อนาที (1.5 มิลลิลิตรต่อนาที)

: จอแสดงผล = 29.833 หยดต่อนาที (1.492 มิลลิลิตรต่อนาที)

ตารางที่ 4.11 แสดงผลการนับหยดน้ำเกลือครั้งที่ 10

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	30	30	29	30	30	30	30	30	30	30
2 (นาทีที่ 15 - 20)	30	30	30	30	30	30	30	30	30	28
3 (นาทีที่ 25 - 30)	30	30	30	30	30	28	29	29	29	29
4 (นาทีที่ 35 - 40)	30	30	30	30	30	29	29	29	30	30
5 (นาทีที่ 45 - 50)	30	30	30	30	30	30	30	30	30	30
6 (นาทีที่ 55 - 60)	30	30	30	30	30	30	30	30	30	30

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 10 : นับหยดจริง = 29.967 หยดต่อนาที (1.498 มิลลิลิตรต่อนาที)

: จอแสดงผล = 29.633 หยดต่อนาที (1.482 มิลลิลิตรต่อนาที)

ตารางที่ 4.12 แสดงค่าเฉลี่ยของการนับหยดน้ำเกลือในการทดลองย่อยที่ 1

การนับหยดน้ำเกลือครั้งที่	ค่าเฉลี่ย				% ความคลาดเคลื่อน
	นับหยดจริง		จอแสดงผล		
	(หยด/นาที)	(มิลลิลิตร/นาที)	(หยด/นาที)	(มิลลิลิตร/นาที)	
1	30.000	1.500	29.467	1.473	1.800
2	29.733	1.487	28.933	1.447	2.689
3	29.900	1.495	29.100	1.455	2.675
4	30.000	1.500	29.633	1.482	1.200
5	29.800	1.490	29.100	1.455	2.348
6	30.000	1.500	29.467	1.473	1.800
7	29.867	1.494	29.800	1.490	0.267
8	29.733	1.487	29.833	1.492	0.336
9	30.000	1.500	29.833	1.492	0.533
10	29.967	1.498	29.633	1.482	1.068
Total	29.900	1.495	29.480	1.474	1.405

จากตารางค่าเฉลี่ยของการนับหยดน้ำเกลือในการทดลองย่อยครั้งที่ 1 กำหนดอัตราการให้ 1 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร นับหยดจริง = 29.90 หยดต่อนาที (1.495 มิลลิลิตรต่อนาที) จอแสดงผล = 29.480 หยดต่อนาที (1.474 มิลลิลิตรต่อนาที) ทำให้ค่าความคลาดเคลื่อนการหยดของน้ำเกลือระหว่างการนับจริงกับจอแสดงผลมีค่าเท่ากับ 1.405 %

ข. การทดลองย่อยที่ 2 กำหนดระยะเวลาการให้ 2 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร ดังนั้นอัตราการให้น้ำเกลือ เท่ากับ 0.83 ml/min หรือ 16.66 หยด/นาที โดยแบ่งเป็น 6 ช่วงเวลา คือ ช่วงที่ 1: 5 – 10 นาที, ช่วงที่ 2: 25 - 30 นาที, ช่วงที่ 3: 45 - 50 นาที, ช่วงที่ 4: 75 - 80 นาที, ช่วงที่ 5: 95 - 100 นาทีและช่วงที่ 6: 115 - 120 นาที ในแต่ละช่วงเวลากำหนดระยะเวลาในการทดสอบ 5 นาที ซึ่งในการจับเวลาการทดสอบจะทำการนับหยดน้ำเกลือทุกๆ 1 นาที โดยทำการทดสอบทั้งหมด 10 ครั้ง และจดบันทึกการทดสอบ

ตารางที่ 4.13 แสดงผลการนับหยดน้ำเกลือครั้งที่ 1

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	18	18	18	18	17	17	17	17	17
2 (นาทีที่ 25 - 30)	18	18	18	18	17	18	18	18	18	17
3 (นาทีที่ 45 - 50)	18	18	18	18	18	17	17	17	17	17
4 (นาทีที่ 75 - 80)	18	18	18	18	18	17	17	17	18	18
5 (นาทีที่ 95 - 100)	18	18	18	18	18	18	18	18	18	18
6 (นาทีที่ 115 - 120)	18	18	18	18	18	18	18	18	17	18

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 1 : นับหยดจริง = 17.967 หยดต่อนาที (0.898 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.5 หยดต่อนาที (0.875 มิลลิลิตรต่อนาที)

ตารางที่ 4.14 แสดงผลการนับหยดน้ำเกลือครั้งที่ 2

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	18	18	18	18	17	17	17	17	17
2 (นาทีที่ 25 - 30)	18	18	18	18	18	17	17	17	17	17
3 (นาทีที่ 45 - 50)	18	18	18	18	18	17	18	18	18	18
4 (นาทีที่ 75 - 80)	18	17	18	18	18	18	18	18	18	18
5 (นาทีที่ 95 - 100)	18	18	18	18	18	17	17	17	17	17
6 (นาทีที่ 115 - 120)	18	18	17	18	18	17	17	17	17	17

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 2 : นับหยดจริง = 17.933 หยดต่อนาที (0.897 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.3 หยดต่อนาที (0.865 มิลลิลิตรต่อนาที)

ตารางที่ 4.15 แสดงผลการนับหยดน้ำเกลือครั้งที่ 3

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	18	18	18	18	17	17	17	17	17
2 (นาทีที่ 25 - 30)	18	18	18	18	17	17	17	17	17	17
3 (นาทีที่ 45 - 50)	18	18	18	18	18	17	17	17	17	17
4 (นาทีที่ 75 - 80)	18	18	18	18	17	17	17	17	17	17
5 (นาทีที่ 95 - 100)	18	18	18	18	18	17	17	17	17	17
6 (นาทีที่ 115 - 120)	18	18	17	18	18	18	18	18	18	18

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 3 : นับหยดจริง = 17.9 หยดต่อนาที (0.895 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.167 หยดต่อนาที (0.858 มิลลิลิตรต่อนาที)

ตารางที่ 4.16 แสดงผลการนับหยดน้ำเกลือครั้งที่ 4

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	18	18	18	18	17	17	17	17	18
2 (นาทีที่ 25 - 30)	18	18	18	18	18	18	18	18	18	17
3 (นาทีที่ 45 - 50)	18	17	18	18	18	17	17	17	18	18
4 (นาทีที่ 75 - 80)	18	18	17	17	17	17	17	17	18	18
5 (นาทีที่ 95 - 100)	17	17	18	17	17	18	18	17	17	17
6 (นาทีที่ 115 - 120)	17	17	17	17	17	18	18	18	17	17

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 4 : นับหยดจริง = 17.567หยดต่อนาที (0.878 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.467 หยดต่อนาที (0.873 มิลลิลิตรต่อนาที)

ตารางที่ 4.17 แสดงผลการนับหยดน้ำเกลือครั้งที่ 5

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	18	18	18	18	18	17	17	17	17
2 (นาทีที่ 25 - 30)	18	18	18	18	18	17	18	18	18	18
3 (นาทีที่ 45 - 50)	18	18	18	18	18	18	18	18	18	18
4 (นาทีที่ 75 - 80)	18	17	18	18	17	17	17	17	17	17
5 (นาทีที่ 95 - 100)	17	16	18	17	17	17	17	18	17	17
6 (นาทีที่ 115 - 120)	17	16	18	17	17	17	17	17	18	18

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 5 : นับหยดจริง = 17.6 หยดต่อนาที (0.88 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.433 หยดต่อนาที (0.872 มิลลิลิตรต่อนาที)

ตารางที่ 4.18 แสดงผลการนับหยดน้ำเกลือครั้งที่ 6

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	18	17	17	17	18	18	18	18	18
2 (นาทีที่ 25 - 30)	18	17	18	18	18	17	18	18	18	18
3 (นาทีที่ 45 - 50)	18	18	18	18	18	17	17	17	18	18
4 (นาทีที่ 75 - 80)	18	18	18	18	17	18	18	18	18	17
5 (นาทีที่ 95 - 100)	17	18	18	18	18	17	17	17	17	18
6 (นาทีที่ 115 - 120)	18	18	18	18	18	18	17	17	17	17

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 6 : นับหยดจริง = 17.2 หยดต่อนาที (0.86 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.567 หยดต่อนาที (0.878 มิลลิลิตรต่อนาที)

ตารางที่ 4.19 แสดงผลการนับหยดน้ำเกลือครั้งที่ 7

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	18	18	18	18	18	18	18	18	18
2 (นาทีที่ 25 - 30)	17	17	17	17	17	17	17	17	17	17
3 (นาทีที่ 45 - 50)	18	18	17	18	18	18	18	18	18	18
4 (นาทีที่ 75 - 80)	18	18	18	18	18	18	18	18	18	17
5 (นาทีที่ 95 - 100)	17	17	17	17	18	17	17	17	18	18
6 (นาทีที่ 115 - 120)	18	18	17	17	17	17	17	17	17	17

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 7 : นับหยดจริง = 17.567 หยดต่อนาที (0.878 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.533หยดต่อนาที (0.877 มิลลิลิตรต่อนาที)

ตารางที่ 4.20 แสดงผลการนับหยดน้ำเกลือครั้งที่ 8

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	18	17	18	18	17	17	17	17	17
2 (นาทีที่ 25 - 30)	18	18	17	18	18	17	17	17	17	17
3 (นาทีที่ 45 - 50)	17	17	18	18	18	18	18	18	18	17
4 (นาทีที่ 75 - 80)	17	17	18	18	18	18	17	17	17	17
5 (นาทีที่ 95 - 100)	17	17	18	18	18	17	16	16	16	18
6 (นาทีที่ 115 - 120)	18	18	18	18	18	16	16	18	18	18

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 8 : นับหยดจริง = 17.733 หยดต่อนาที (0.887 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.133 หยดต่อนาที (0.857 มิลลิลิตรต่อนาที)

ตารางที่ 4.21 แสดงผลการนับหยดน้ำเกลือครั้งที่ 9

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	17	18	18	18	18	18	18	18	18
2 (นาทีที่ 25 - 30)	18	18	18	18	18	17	17	17	17	17
3 (นาทีที่ 45 - 50)	17	17	17	17	17	17	17	17	17	17
4 (นาทีที่ 75 - 80)	17	17	17	17	17	17	17	17	17	17
5 (นาทีที่ 95 - 100)	17	17	17	17	17	17	17	17	17	17
6 (นาทีที่ 115 - 120)	17	17	17	17	17	17	17	17	17	17

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 9 : นับหยดจริง = 17.3 หยดต่อนาที (0.865 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.167 หยดต่อนาที (0.858 มิลลิลิตรต่อนาที)

ตารางที่ 4.22 แสดงผลการนับหยดน้ำเกลือครั้งที่ 10

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 5 - 10)	18	18	17	17	17	17	17	17	17	17
2 (นาทีที่ 25 - 30)	17	17	17	18	18	17	17	17	17	18
3 (นาทีที่ 45 - 50)	17	17	17	18	18	17	17	17	17	18
4 (นาทีที่ 75 - 80)	17	17	18	17	17	17	18	18	18	18
5 (นาทีที่ 95 - 100)	17	17	17	17	17	17	17	17	17	17
6 (นาทีที่ 115 - 120)	18	18	17	17	18	18	18	17	17	17

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 10 : นับหยดจริง = 17.333 หยดต่อนาที (0.867 มิลลิลิตรต่อนาที)

: จอแสดงผล = 17.267 หยดต่อนาที (0.863 มิลลิลิตรต่อนาที)

ตารางที่ 4.23 แสดงค่าเฉลี่ยของการนับหยดน้ำเกลือในการทดลองย่อยที่ 2

การนับหยดน้ำเกลือครั้งที่	ค่าเฉลี่ย				% ความคลาดเคลื่อน
	นับหยดจริง		จอแสดงผล		
	(หยด/นาที)	(มิลลิลิตร/นาที)	(หยด/นาที)	(มิลลิลิตร/นาที)	
1	17.967	0.898	17.500	0.875	2.561
2	17.933	0.897	17.300	0.865	3.567
3	17.900	0.895	17.167	0.858	4.134
4	17.567	0.878	17.467	0.873	0.569
5	17.600	0.880	17.433	0.872	0.909
6	17.200	0.860	17.567	0.878	2.093
7	17.567	0.878	17.533	0.877	0.113
8	17.733	0.887	17.133	0.857	3.382
9	17.300	0.865	17.167	0.858	0.809
10	17.333	0.867	17.267	0.863	0.461
Total	17.610	0.881	17.353	0.868	1.459

จากตารางค่าเฉลี่ยของการนับหยดน้ำเกลือในการทดลองย่อยครั้งที่ 2 กำหนดอัตราการให้ 2 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร นับหยดจริง = 17.61 หยดต่อนาที (0.881 มิลลิลิตรต่อนาที) จอแสดงผล = 17.353 หยดต่อนาที (0.868 มิลลิลิตรต่อนาที) ทำให้ค่าความคลาดเคลื่อนการหยดของน้ำเกลือระหว่างการนับจริงกับจอแสดงผลมีค่าเท่ากับ 1.459 %

ค. การทดลองย่อยที่ 3 กำหนดระยะเวลาการให้ 3 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร ดังนั้นอัตราการให้น้ำเกลือ เท่ากับ 0.56 ml/min หรือ 11.2 หยด/นาที โดยแบ่งเป็น 6 ช่วงเวลา คือ ช่วงที่ 1: 25 – 30 นาที, ช่วงที่ 2: 55 - 60 นาที, ช่วงที่ 3: 85 - 90 นาที, ช่วงที่ 4: 115 - 120 นาที, ช่วงที่ 5: 145 - 150 นาทีและช่วงที่ 6: 175 - 180 ในแต่ละช่วงเวลา กำหนดระยะเวลาในการทดสอบ 5 นาที ซึ่งในการจับเวลาการทดสอบจะทำการนับหยดน้ำเกลือทุกๆ 1 นาที โดยทำการทดสอบทั้งหมด 10 ครั้ง และจดบันทึกการทดสอบ

ตารางที่ 4.24 แสดงผลการนับหยดน้ำเกลือครั้งที่ 1

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	12	12	12	12	12	12	12	12	12	12
2 (นาทีที่ 55 - 60)	12	12	12	12	12	12	12	12	12	12
3 (นาทีที่ 85 - 90)	12	12	12	12	12	12	12	12	12	12
4 (นาทีที่ 115 - 120)	12	12	12	12	12	12	12	12	12	12
5 (นาทีที่ 145 - 150)	12	12	12	12	12	12	12	12	12	12
6 (นาทีที่ 175 - 180)	12	12	12	12	12	12	12	12	12	12

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 1 : นับหยดจริง = 12 หยดต่อนาที (0.6 มิลลิลิตรต่อนาที)

: จอแสดงผล = 12 หยดต่อนาที (0.6 มิลลิลิตรต่อนาที)

ตารางที่ 4.25 แสดงผลการนับหยดน้ำเกลือครั้งที่ 2

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	12	12	12	12	12	12	12	12	12	12
2 (นาทีที่ 55 - 60)	12	12	12	12	12	12	12	12	12	12
3 (นาทีที่ 85 - 90)	12	12	12	12	12	12	12	12	12	12
4 (นาทีที่ 115 - 120)	12	12	11	11	11	12	12	10	10	10
5 (นาทีที่ 145 - 150)	12	12	12	11	11	10	10	10	10	9
6 (นาทีที่ 175 - 180)	12	12	12	12	12	9	9	9	9	9

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 2 : นับหยดจริง = 11.833 หยดต่อนาที (0.592 มิลลิลิตรต่อนาที)

: จอแสดงผล = 10.933 หยดต่อนาที (0.547 มิลลิลิตรต่อนาที)

ตารางที่ 4.26 แสดงผลการนับหยดน้ำเกลือครั้งที่ 3

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	11	11	11	12	12	10	10	10	10	10
2 (นาทีที่ 55 - 60)	10	10	10	11	12	10	10	10	11	11
3 (นาทีที่ 85 - 90)	10	10	10	10	10	10	11	11	11	11
4 (นาทีที่ 115 - 120)	11	11	11	10	10	11	11	11	10	10
5 (นาทีที่ 145 - 150)	10	11	11	10	10	10	11	11	11	11
6 (นาทีที่ 175 - 180)	10	11	10	11	11	11	11	11	11	11

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 3 : นับหยดจริง = 10.6 หยดต่อนาที (0.53 มิลลิลิตรต่อนาที)

: จอแสดงผล = 10.6 หยดต่อนาที (0.53 มิลลิลิตรต่อนาที)

ตารางที่ 4.27 แสดงผลการนับหยดน้ำเกลือครั้งที่ 4

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	11	12	11	11	11	12	12	12	12	12
2 (นาทีที่ 55 - 60)	12	11	10	11	12	12	12	12	12	11
3 (นาทีที่ 85 - 90)	11	11	12	11	11	11	11	12	12	12
4 (นาทีที่ 115 - 120)	11	12	11	11	11	11	12	12	12	12
5 (นาทีที่ 145 - 150)	11	11	11	11	11	11	12	11	11	11
6 (นาทีที่ 175 - 180)	11	11	11	12	11	12	12	11	11	11

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 4 : นับหยดจริง = 11.167 หยดต่อนาที (0.558 มิลลิลิตรต่อนาที)

: จอแสดงผล = 11.633 หยดต่อนาที (0.582 มิลลิลิตรต่อนาที)

ตารางที่ 4.28 แสดงผลการนับหยดน้ำเกลือครั้งที่ 5

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	11	11	11	12	12	11	11	11	12	12
2 (นาทีที่ 55 - 60)	10	10	10	11	12	11	11	12	12	12
3 (นาทีที่ 85 - 90)	11	12	11	11	11	11	12	12	12	11
4 (นาทีที่ 115 - 120)	11	11	11	11	11	11	11	11	11	11
5 (นาทีที่ 145 - 150)	12	10	11	11	11	11	11	11	11	11
6 (นาทีที่ 175 - 180)	11	11	11	11	11	11	11	11	11	11

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 5 6 : นับหยดจริง = 11.033 หยดต่อนาที (0.552 มิลลิลิตรต่อนาที)

: จอแสดงผล = 11.267 หยดต่อนาที (0.563 มิลลิลิตรต่อนาที)

ตารางที่ 4.29 แสดงผลการนับหยดน้ำเกลือครั้งที่ 6

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	11	11	11	11	11	11	11	11	11	12
2 (นาทีที่ 55 - 60)	11	11	11	11	11	11	11	11	12	12
3 (นาทีที่ 85 - 90)	11	11	11	11	11	11	11	11	12	12
4 (นาทีที่ 115 - 120)	11	11	11	11	11	11	11	11	11	11
5 (นาทีที่ 145 - 150)	12	12	12	12	12	12	12	12	11	11
6 (นาทีที่ 175 - 180)	12	12	12	10	11	12	12	11	11	11

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 6 : นับหยดจริง = 11.233 หยดต่อนาที (0.562 มิลลิลิตรต่อนาที)

: จอแสดงผล = 11.333 หยดต่อนาที (0.567 มิลลิลิตรต่อนาที)

ตารางที่ 4.30 แสดงผลการนับหยดน้ำเกลือครั้งที่ 7

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	11	11	11	12	12	11	11	11	11	11
2 (นาทีที่ 55 - 60)	12	12	11	11	11	11	11	11	11	11
3 (นาทีที่ 85 - 90)	11	11	11	11	11	11	11	11	11	11
4 (นาทีที่ 115 - 120)	12	12	11	12	12	12	12	12	11	11
5 (นาทีที่ 145 - 150)	12	12	11	12	12	12	12	12	11	11
6 (นาทีที่ 175 - 180)	12	12	12	12	11	12	12	12	11	11

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 7 : นับหยดจริง = 11.533 หยดต่อนาที (0.577 มิลลิลิตรต่อนาที)

: จอแสดงผล = 11.3 หยดต่อนาที (0.565 มิลลิลิตรต่อนาที)

ตารางที่ 4.31 แสดงผลการนับหยดน้ำเกลือครั้งที่ 8

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	11	11	11	12	12	11	11	11	12	12
2 (นาทีที่ 55 - 60)	11	11	11	12	12	11	11	11	12	12
3 (นาทีที่ 85 - 90)	11	11	11	11	11	11	11	11	12	12
4 (นาทีที่ 115 - 120)	11	11	11	11	11	11	11	11	11	12
5 (นาทีที่ 145 - 150)	11	11	11	11	11	11	11	11	11	12
6 (นาทีที่ 175 - 180)	11	11	11	11	11	11	11	11	11	12

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 8 : นับหยดจริง = 11.133 หยดต่อนาที (0.557 มิลลิลิตรต่อนาที)

: จอแสดงผล = 11.3 หยดต่อนาที (0.565 มิลลิลิตรต่อนาที)

ตารางที่ 4.32 แสดงผลการนับหยดน้ำเกลือครั้งที่ 9

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	11	11	11	11	11	11	11	11	11	11
2 (นาทีที่ 55 - 60)	11	11	11	11	11	11	11	11	11	11
3 (นาทีที่ 85 - 90)	11	11	11	11	12	11	11	11	11	11
4 (นาทีที่ 115 - 120)	11	11	11	11	11	11	11	11	11	11
5 (นาทีที่ 145 - 150)	11	11	11	11	11	12	11	11	11	12
6 (นาทีที่ 175 - 180)	11	11	11	11	12	12	11	11	11	12

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 9 : นับหยดจริง = 11.067 หยดต่อนาที (0.553 มิลลิลิตรต่อนาที)

: จอแสดงผล = 11.133 หยดต่อนาที (0.557 มิลลิลิตรต่อนาที)

ตารางที่ 4.33 แสดงผลการนับหยดน้ำเกลือครั้งที่ 10

ช่วงเวลาที่	นาทีที่									
	นับหยดจริง					จอแสดงผล				
	1	2	3	4	5	1	2	3	4	5
1 (นาทีที่ 25 - 30)	11	11	11	11	12	11	11	11	11	12
2 (นาทีที่ 55 - 60)	11	11	11	11	11	11	11	11	11	11
3 (นาทีที่ 85 - 90)	11	11	11	11	12	11	11	11	11	12
4 (นาทีที่ 115 - 120)	11	11	11	11	11	11	11	11	11	11
5 (นาทีที่ 145 - 150)	11	11	11	11	11	11	11	11	11	11
6 (นาทีที่ 175 - 180)	11	11	11	11	12	12	12	11	11	11

ค่าเฉลี่ยของจำนวนหยดน้ำเกลือครั้งที่ 10 : นับหยดจริง = 11.1 หยดต่อนาที (0.555 มิลลิลิตรต่อนาที)

: จอแสดงผล = 11.133 หยดต่อนาที (0.557 มิลลิลิตรต่อนาที)

ตารางที่ 4.34 แสดงค่าเฉลี่ยของการนับหยดน้ำเกลือในการทดลองย่อยที่ 3

การนับหยดน้ำเกลือครั้งที่	ค่าเฉลี่ย				% ความคลาดเคลื่อน
	นับหยดจริง		จอแสดงผล		
	(หยด/นาที)	(มิลลิลิตร/นาที)	(หยด/นาที)	(มิลลิลิตร/นาที)	
1	12.000	0.600	12.000	0.600	0
2	11.833	0.592	10.933	0.547	7.601
3	10.600	0.530	10.600	0.530	0
4	11.167	0.558	11.633	0.582	4.301
5	11.033	0.552	11.267	0.563	1.992
6	11.233	0.562	11.333	0.567	0.889
7	11.533	0.577	11.300	0.565	2.079
8	11.133	0.557	11.300	0.565	1.436
9	11.067	0.553	11.133	0.557	0.723
10	11.100	0.555	11.133	0.557	0.360
Total	11.270	0.564	11.263	0.563	0.062

จากตารางค่าเฉลี่ยของการนับหยดน้ำเกลือในการทดลองย่อยครั้งที่ 3 กำหนดอัตราการให้ 3 ชั่วโมงและปริมาตรของน้ำเกลือเท่ากับ 100 มิลลิลิตร นับหยดจริง = 11.270 หยดต่อนาที (0.564 มิลลิลิตรต่อนาที) จอแสดงผล = 11.263 หยดต่อนาที (0.563 มิลลิลิตรต่อนาที) ทำให้ค่าความคลาดเคลื่อนการหยดของน้ำเกลือระหว่างการนับจริงกับจอแสดงผลมีค่าเท่ากับ 0.062 %

ดังนั้น ค่าเฉลี่ยของการนับหยดน้ำเกลือทั้ง 3 การทดลองย่อย ทำให้เครื่องวัดหยดน้ำเกลือและส่งข้อมูลแบบไร้สาย มีการวัดอัตราการไหลของน้ำเกลือและทำการนับจำนวนหยดของน้ำเกลือได้จริง และผลที่ได้อยู่ในเกณฑ์ความคลาดเคลื่อนไม่เกิน 0.975 %

4.2.3 ผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูล

ผลการทดลองดังต่อไปนี้แสดงผลการทดสอบขอบเขตระยะทางของการรับส่งข้อมูลระหว่างบอร์ดไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) เพื่อตรวจสอบประสิทธิภาพขอบเขตระยะทางระหว่างบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) กับบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ดังนี้

ก. การทดลองย่อยที่ 1 ในบริเวณพื้นที่โล่ง

ผลจากการทดสอบขอบเขตระยะทางของการรับส่งข้อมูลระหว่างบอร์ดไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) ในบริเวณพื้นที่โล่ง มีดังนี้

ตารางที่ 4.35 ผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูลบริเวณพื้นที่โล่ง

ระยะทาง (เมตร)	ผลการทดสอบ
10	✓
20	✓
30	✓
40	✓
50	✓
60	✓
70	✓
80	✓
90	✓
100	✓
110	✓
120	✓
130	✓
140	✓
150	✓
160	✓
170	✓
180	✓
190	✓
200	✓
210	✓
220	✓
230	✓
240	✓
250	✓

ตารางที่ 4.35 ผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูลบริเวณพื้นที่โล่ง (ต่อ)

ระยะทาง (เมตร)	ผลการทดสอบ
260	✓
270	✓
280	✓
290	✓
291	✗
292	✗
293	✗
294	✗
295	✗
300	✗

หมายเหตุ ✓ คือ สามารถรับส่งข้อมูลได้

✗ คือ ไม่สามารถรับส่งข้อมูลได้

จากตารางที่ 4.36 แสดงผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูลบริเวณพื้นที่โล่ง กำหนดให้เพิ่มระยะครั้งละ 10 เมตร โดยที่ระยะทาง 290 เมตรบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) สามารถรับส่งข้อมูลถึงกันได้ (จอแอลซีดีแสดงข้อความ "CH1: DEPLETE, CH2: DEPLETE" ซึ่งหมายถึง แจ้งน้ำเกลือหมด เนื่องจากการทดสอบระยะทาง ที่บอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ไม่ได้ต่อเข้ากับหลอดหยดน้ำเกลือเพื่อความสะดวกในการทดสอบ) แต่เมื่อเพิ่มระยะทางเป็น 300 เมตรบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ไม่สามารถรับส่งข้อมูลถึงกันได้ (จอแอลซีดีแสดงข้อความ "ERROR BOARD, DISCONNECT!") จากนั้นจึงลดระยะทางการรับส่งข้อมูลโดยเพิ่มทีละ 1 เมตรจากระยะ 290 เมตร เพื่อให้มีความละเอียดมากขึ้น โดยที่ระยะ 291 - 295 เมตร บอร์ดตัวรับ (Atmega32) กับบอร์ดตัวส่ง (Attiny2313) ไม่สามารถรับส่งข้อมูลถึงกันได้ ดังนั้นทำให้ผลการทดลองการทดสอบระยะทางไกลที่สุดเท่ากับ 290 เมตรในบริเวณพื้นที่โล่ง

ผลการทดสอบการรับส่งข้อมูลในระยะ 10 เมตร ดังรูปที่ 4.15

(ก) (ข) (ค)

รูปที่ 4.15 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 10 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 10 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 10 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 50 เมตร ดังรูปที่ 4.16

(ก) (ข) (ค)

รูปที่ 4.16 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 50 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 50 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 50 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 110 เมตร ดังรูปที่ 4.17

(ก) (ข) (ค)

รูปที่ 4.17 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 110 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 110 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 110 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 180 เมตร ดังรูปที่ 4.18

(ก) (ข) (ค)

รูปที่ 4.18 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 180 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 180 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 180 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 290 เมตร ดังรูปที่ 4.19

(ก) (ข) (ค)

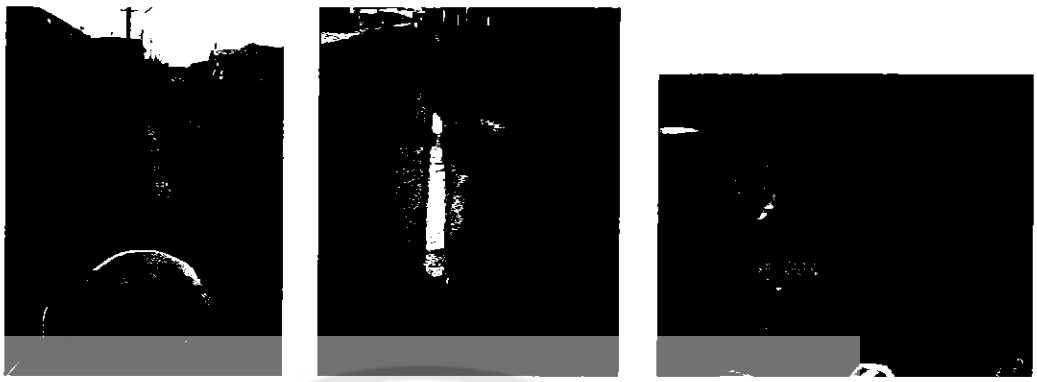
รูปที่ 4.19 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 290 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 290 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 290 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 291 เมตร ดังรูปที่ 4.20

(ก) (ข) (ค)

รูปที่ 4.20 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 291 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 291 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 291 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 300 เมตร ดังรูปที่ 4.21



(ก) (ข) (ค)

รูปที่ 4.21 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 300 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 300 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 300 เมตร

ข. การทดลองย่อยที่ 2 ในบริเวณพื้นที่ปิด

ผลจากการทดสอบขอบเขตระยะทางของการรับส่งข้อมูลระหว่างบอร์ดไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) ในบริเวณพื้นที่ปิด ที่มีสภาพเป็นห้องและมีอุปกรณ์กีดขวางการรับส่งข้อมูลระหว่างบอร์ดไมโครคอนโทรเลอร์ มีดังนี้

ตารางที่ 4.37 ผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูลบริเวณพื้นที่ปิด

ระยะทาง (เมตร)	ผลการทดสอบ
5	✓
10	✓
15	✓
16	✓
17	✗
18	✗
19	✗
20	✗

หมายเหตุ ✓ คือ สามารถรับส่งข้อมูลได้
✗ คือ ไม่สามารถรับส่งข้อมูลได้

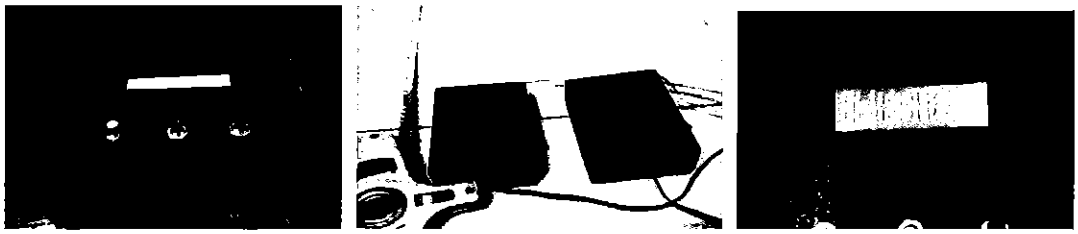
จากตารางที่ 4.37 แสดงผลการทดลองจากการทดสอบระยะทางในการรับส่งข้อมูลบริเวณพื้นที่ปิด กำหนดให้เพิ่มระยะครั้งละ 5 เมตร โดยที่ระยะทาง 15 เมตรบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) สามารถรับส่งข้อมูลถึงกันได้ (จอแอลซีดีแสดงข้อความ "CH1: DEPLETE, CH2: DEPLETE" ซึ่งหมายถึง แจ้งน้ำเกลือหมด เนื่องจากการทดสอบระยะทาง ที่บอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ไม่ได้ต่อเข้ากับหลอดหยดน้ำเกลือเพื่อความสะดวกในการทดสอบ) แต่เมื่อเพิ่มระยะทางเป็น 20 เมตรบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ไม่สามารถรับส่งข้อมูลถึงกันได้ (จอแอลซีดีแสดงข้อความ "ERROR BOARD, DISCONNECT!") จากนั้นจึงลดระยะทางการรับส่งข้อมูลโดยเพิ่มทีละ 1 เมตรจากระยะ 15 เมตร เพื่อให้มีความละเอียดมากขึ้น โดยที่ระยะ 16 เมตรบอร์ดไมโครคอนโทลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) สามารถรับส่งข้อมูลถึงกันได้ เมื่อเพิ่มระยะทางเป็น 17 – 20 เมตร บอร์ดตัวรับ (Atmega32) กับบอร์ดตัวส่ง (Attiny2313) ไม่สามารถรับส่งข้อมูลถึงกันได้ ดังนั้นทำให้ผลการทดลองการทดสอบระยะทางไกลที่สุดเท่ากับ 16 เมตรในบริเวณพื้นที่ปิด

ผลการทดสอบการรับส่งข้อมูลในระยะ 5 เมตร ดังรูปที่ 4.22

(ก) (ข) (ค)

รูปที่ 4.22 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 5 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 5 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 5 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 10 เมตร ดังรูปที่ 4.23



(ก) (ข) (ค)

รูปที่ 4.23 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 10 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 10 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 10 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 15 เมตร ดังรูปที่ 4.24

(ก) (ข) (ค)

รูปที่ 4.24 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 15 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 15 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 15 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 16 เมตร ดังรูปที่ 4.25

(ก) (ข) (ค)

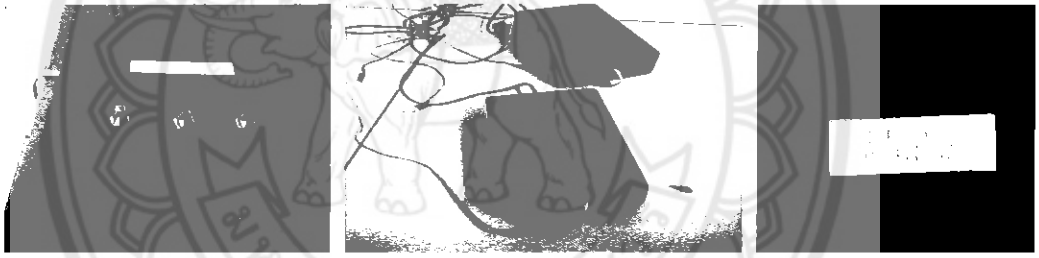
รูปที่ 4.25 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 16 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 16 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 16 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 17 เมตร ดังรูปที่ 4.26

(ก) (ข) (ค)

รูปที่ 4.26 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 17 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 17 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 17 เมตร

ผลการทดสอบการรับส่งข้อมูลในระยะ 20 เมตร ดังรูปที่ 4.27



(ก) (ข) (ค)

รูปที่ 4.27 (ก) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวรับที่ระยะ 20 เมตร
(ข) แสดงการทดสอบบอร์ดไมโครคอนโทลเลอร์ตัวส่งที่ระยะ 20 เมตร
(ค) แสดงผลการทดสอบที่ระยะ 20 เมตร

4.3 สรุปผลการทดลอง

จากการทดลองการใช้งานอุปกรณ์เครื่องวัดหยดน้ำเกลือ พบว่าระบบแสดงอัตราการให้น้ำเกลือแบบไร้สายที่พัฒนาขึ้นนั้น ช่วยให้พยาบาลหรือผู้ดูแลผู้ป่วยได้รับทราบข้อมูลและสถานะในการให้น้ำเกลือแก่ผู้ป่วยผ่านทางการส่งข้อมูลแบบไร้สาย ทำให้ไม่ต้องคอยมาตรวจสอบปริมาตรของน้ำเกลือที่เตียงผู้ป่วย โดยอุปกรณ์เครื่องวัดหยดน้ำเกลือจะทำการนับหยดน้ำเกลือผ่านเซนเซอร์ R-Reflex S5 ซึ่งทำหน้าที่ในการนับหยดน้ำเกลือ โดยเซนเซอร์จะทำการตรวจจับหยดน้ำเกลือมาเป็นสัญญาณแรงดัน แล้วส่งไปยังชุดวงจรเปรียบเทียบแรงดันที่มีการเชื่อมต่อกัน ทำหน้าที่ในการเปรียบเทียบแรงดันเป็นสัญญาณแรงดันดิจิตอลลอจิก “0” (0V) และ “1” (+5V)โดยเอาท์พุตที่ได้จากการนับหยดน้ำเกลือ 1 หยดเท่ากับ 1 บิตจะมีค่าเป็นลอจิก “0” (0V) จากนั้นส่งข้อมูลนี้ให้กับบอร์ด

ไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) เพื่อนำมาประมวลผล บอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) ทำหน้าที่ในการรับเอาต์พุตสัญญาณแรงดันดิจิตอลลอจิกจากชุดวงจรเปรียบเทียบแรงดันที่เชื่อมต่อกับตัวเซนเซอร์(R-Reflex S5) โดยการนำค่าเอาท์พุตที่ได้จากการนับหยดน้ำเกลือ 1 หยดเท่ากับ 1 บิตจะมีค่าเป็นลอจิก "0" (0V) ทำให้อยู่ในรูปของเลขฐานสิบหก ใช้ Timer ในการจับเวลาการนับหยด 10 วินาที เมื่อครบ 10 วินาที ไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) จะทำการส่งจำนวนหยดที่นับได้ไปทีละบิต ในข้อมูลจะมีรหัสประจำเครื่องนำหน้าตามด้วยข้อมูลและปิดท้ายด้วยเครื่องหมาย "!" จากนั้นส่งข้อมูลดังกล่าวไปยังบอร์ดไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) ซึ่งทำหน้าที่ในการนำจำนวนหยดที่นับได้จากบอร์ดไมโครคอนโทลเลอร์ตัวส่ง (Attiny2313) แปลงเป็นหน่วยมิลลิลิตรต่อ 10 วินาที จากนั้นส่งข้อมูลที่ได้ผ่านสาย RS232 เพื่อนำไปแสดงผลบนหน้าจอมอนิเตอร์ของคอมพิวเตอร์ส่วนบุคคลโดยผ่านโปรแกรม Visual Basic 6.0 ที่เคาท์เตอร์พยาบาล โปรแกรม Visual Basic 6.0 จะทำการรวมค่าปริมาตรที่ได้จนครบ 6 ครั้ง (60 วินาที) มาแสดงผลทุกๆ 1 นาที โดยตัวเครื่องจะบอกถึงหมายเลขเตียง ข้อมูลของคนไข้ อัตราการไหลของน้ำเกลือขณะนี้เป็นมิลลิลิตรต่อเวลาหนึ่งนาทีและหยดต่อเวลาหนึ่งนาที อัตราการไหลของน้ำเกลือที่ต้องการเป็นมิลลิลิตรต่อเวลาหนึ่งนาทีและหยดต่อเวลาหนึ่งนาที ปริมาตรคงเหลือของน้ำเกลือในขวดน้ำเกลือคิดเป็นเปอร์เซ็นต์ และส่งสัญญาณเสียงเตือน เมื่อน้ำเกลือใกล้จะหมด ซึ่งในการวัดอัตราการไหลของน้ำเกลือและทำการนับจำนวนหยดของน้ำเกลือได้จริง เนื่องจากการทำการทดลองทำให้มีการขนย้ายอุปกรณ์ ซึ่งอาจทำให้อุปกรณ์ได้รับการกระทบกระเทือน จึงส่งผลทำให้เกิดความคลาดเคลื่อน ประสิทธิภาพของอุปกรณ์ลดลง จึงทำให้ผลการทดสอบระยะการรับส่งข้อมูลมีระยะที่ 290 เมตรในบริเวณพื้นที่โล่งและมีระยะการรับส่งข้อมูลที่ 16 เมตรในบริเวณพื้นที่ปิด และเนื่องจากจังหวะการนับหยดน้ำเกลือระหว่างผู้ทำการทดลองกับบอร์ดไมโครคอนโทรเลอร์อาจไม่ตรงกันจึงเป็นผลทำให้มีเกณฑ์ความคลาดเคลื่อนสูงสุด 1.459 % และมีค่าเฉลี่ยความคลาดเคลื่อน 0.975 % ซึ่งในส่วนของการทดลองนั้นได้ทดลองเฉพาะสารละลายที่เป็นน้ำเกลือธรรมดาเท่านั้น ส่วนสารละลายอื่นจะต้องทำการทดลองใหม่อีกครั้ง เพื่อความมั่นใจต่อการใช้งานต่อไป

เมื่อทำการเชื่อมต่อเรียบร้อยแล้ว ต้องทำการตั้งค่าข้อมูลของผู้ป่วยลงในโปรแกรม Microsoft Visual basic 6.0 เพื่อให้แสดงข้อมูลของผู้ป่วยรายนั้นๆ ออกมาทางหน้าจอคอมพิวเตอร์ การแสดงผลจะแสดงค่าทุก 1 นาที อีกทั้งยังสามารถแสดงข้อมูลของผู้ป่วยได้มากถึงสองคนได้อีกด้วย

บทที่ 5
สรุปผลการดำเนินโครงงาน

ในบทนี้ได้กล่าวถึงสรุปของผลการดำเนินงานที่ได้ทำมาตลอดปีการศึกษาว่าได้เจอปัญหาหรือข้อผิดพลาดใดบ้างที่เกิดขึ้นในระหว่างการดำเนินโครงงาน เพื่อเสนอแนะแนวทางในการแก้ไขและพัฒนาให้ระบบแสดงอัตราการให้น้ำเกลือแบบไร้สายมีประสิทธิภาพที่ดีขึ้น จากโครงงานทำให้ได้ข้อสรุปผลการดำเนินงาน ดังนี้

1. ระบบแสดงอัตราการให้น้ำเกลือแบบไร้สายนี้สามารถใช้ในระยะ 290 เมตรในบริเวณพื้นที่โล่งและในระยะ 16 เมตร ในบริเวณพื้นที่ปิด สามารถนำไปพัฒนาต่อได้ในอนาคต
2. การวัดอัตราการไหลของน้ำเกลือและการนับจำนวนหยดของน้ำเกลือ ผลที่ได้อยู่ในเกณฑ์ความคลาดเคลื่อนสูงสุด 1.459 % และมีค่าเฉลี่ยความคลาดเคลื่อน 0.975 %

5.1 ปัญหาและแนวทางการแก้ปัญหา

เนื่องจากจังหวะการนับหยดน้ำเกลือระหว่างผู้ทำการทดลองกับบอร์ดไมโครคอนโทรเลอร์อาจไม่ตรงกันจึงทำให้ค่าที่ได้อยู่ในเกณฑ์ความคลาดเคลื่อนสูงสุด 1.459 % และในส่วนของอุปกรณ์ตัวส่งข้อมูล (บอร์ดไมโครคอนโทรลเลอร์ตัวส่ง Attiny2313) ที่ทำการติดตั้งกับขวดน้ำเกลือต้องมีการจ่ายไฟเลี้ยงให้กับบอร์ดไมโครคอนโทรเลอร์ตลอดเวลา ฉะนั้นจึงต้องติดตั้งบริเวณที่มีแหล่งจ่ายไฟและไม่ควรมีการเคลื่อนย้ายหลังจากที่ได้ติดตั้งอุปกรณ์แล้ว

5.2 ข้อเสนอแนะในการพัฒนา

ระบบแสดงอัตราการให้น้ำเกลือแบบไร้สายนี้สามารถปรับแต่งโปรแกรมแสดงผลได้ สามารถเพิ่มจำนวนของขวดน้ำเกลือ สามารถนำปุ่ม Date/Time ที่มีในโปรแกรมมาพัฒนาให้เชื่อมต่อกับโปรแกรมต่อไปได้ และสามารถนำโปรแกรม Visual Basic 6.0 ในโครงงานนี้พัฒนาเป็นเวอร์ชันอื่นได้

ในส่วนของโปรแกรมแสดงผลเพื่อให้มีความละเอียดมากขึ้น ในการพัฒนาควรเพิ่มการแสดงผลเพื่อรองรับการแสดงผลตามจำนวนขวดน้ำเกลือที่เพิ่มขึ้น โดยการเพิ่มโปรแกรมคำสั่งการกำหนดรับค่าจากบอร์ดไมโครคอนโทรเลอร์ตัวรับ (Atmega32) ในโปรแกรม Visual Basic 6.0

ในส่วนของการเพิ่มจำนวนของขวดน้ำเกลือ ต้องมีการเพิ่มโปรแกรมคำสั่งของบอร์ดไมโครคอนโทรลเลอร์ตัวรับ (Atmega32) และบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313)

- บอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) ทุกตัวใช้โปรแกรมคำสั่งเดียวกัน

แต่แตกต่างกันที่รหัสประจำเครื่อง ซึ่งแต่ละตัวจะต้องมีรหัสประจำเครื่องที่ไม่เหมือนกัน ในการพัฒนาเมื่อมีการเพิ่มจำนวนขวดน้ำเกลือ ควรกำหนดรหัสประจำเครื่องที่ไม่ซ้ำกัน

- บอร์ดไมโครคอนโทรลเลอร์ตัวรับ(Atmega32) ต้องเพิ่มรหัสประจำเครื่องไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) ที่เพิ่มตามจำนวนขวดน้ำเกลือและรหัสประจำเครื่องจะต้องตรงกันกับบอร์ดไมโครคอนโทรลเลอร์ตัวส่ง (Attiny2313) ที่กำหนดขึ้นมาใหม่
- ในโครงงานนี้มีการตั้งค่าการทำงานให้ 2 เครื่อง (ส่ง) ทำงานพร้อมกัน ในการพัฒนา เมื่อมีการเพิ่มจำนวนเครื่องส่งจะต้องให้เครื่องส่งทั้งหมดทำงานพร้อมกัน

เอกสารอ้างอิง

[1] อภินันท์ ก้อนมณีและสมศักดิ์ อรรคทิมากูล. (2551). การสร้างชุดควบคุมวาล์วน้ำไร้สายโดยใช้ไมโครโปรเซสเซอร์ การประชุมทางวิชาการระดับชาติด้านคุรุศาสตร์อุตสาหกรรม ครั้งที่ 1 มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 22

[2] ระบบเครือข่ายไร้สาย Wireless Lan. สืบค้นเมื่อ 1 มิถุนายน 2557, จาก http://www.ptn2000.com/wireless/wireless_files/page0001.htm

[3] Wikipedia (2554) วายฟาย. สืบค้นเมื่อ 7 กันยายน 2555, จาก http://th.wikipedia.org/wiki/วายฟาย

[4] บริษัท Atmel Corporation จำกัด.สืบค้นเมื่อ 15 กันยายน 2555, จาก http://www.atmel.com/Images/doc2543.pdf

[5] บริษัท Atmel Corporation จำกัด. สืบค้นเมื่อ 23 กันยายน 2555, จาก http://www.atmel.com/Images/doc2503.pdf

[6] บริษัท อีทีที จำกัด. สืบค้นเมื่อ 23 กันยายน 2555, จาก http:// www.etteam.com/product/robot/rsensor5.html

[7] บริษัท CSJ Electronics จำกัด.สืบค้นเมื่อ 10 ตุลาคม 2555, จากเว็บไซต์: http://www.csjelectronics.com/TRW-24G.pdf

[8] ครูวิทูลย์ งามขา. วิชาการเขียนโปรแกรม "Visual Basic 6.0".สืบค้นเมื่อ 15 ตุลาคม 2555, จาก http://www.pbps.ac.th/e_learning/vb6/contact.html

[9] บริษัท คริสตัลไดแอกนอสติกจำกัด. สืบค้นเมื่อ 20 ตุลาคม 2555, จาก http://www.thailabonline.com/drug/drug1.htm#น้ำเกลือ/Saline

[10] ดร.สุเมธ อ่ำชิต . "เครื่องวัดหยดน้ำเกลือบลูทูธ" อาจารย์ประจำภาควิชาฟิสิกส์อุตสาหกรรมและอุปกรณ์การแพทย์ คณะวิทยาศาสตร์ประยุกต์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

[11] http://cptd.chandra.ac.th/selfstud/datacom/CAI/part3-8.htm. สืบค้นเมื่อ 28 ตุลาคม 2555

[12] ผศ.ดร.ปิยะ โควินท์ทวีวัฒน์. ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (2554) บทที่ 5 "บูรณภาพของข้อมูล".สืบค้นเมื่อ 27 ตุลาคม 2555, จาก http://home.npru.ac.th/piya/RFID/file/Piya_Ch5.pdf

เอกสารอ้างอิง (ต่อ)

[13] ความรู้พื้นฐานของเทคโนโลยีเครือข่ายรูปแบบการส่งผ่านข้อมูล. สืบค้นเมื่อ 10 ตุลาคม 2555, จาก http://irrigation.rid.go.th/rid15/ppn/Knowledge/Networks%20Technology/network4.htm

[14] ความรู้พื้นฐานของการสื่อสารข้อมูลทางอิเล็กทรอนิกส์. สืบค้นเมื่อ 10 พฤศจิกายน 2555, จาก http://janney105.blogspot.com/2009/12/blog-post.html

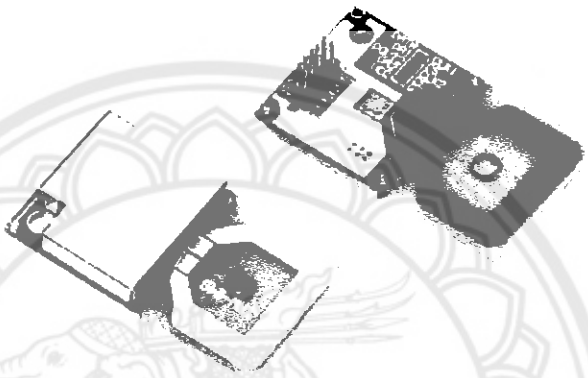
[15] บริษัท วีนัส ซัพพลาย จำกัด.สืบค้นเมื่อ 12 พฤศจิกายน 2555, จาก http://www.thaieasyelec.com/electronics-in-chapter/UART-TTL-RS232-MAX232-MAX3232.html

ภาคผนวก ก

คู่มือ TRW 2.4 GHz

TRW-24G

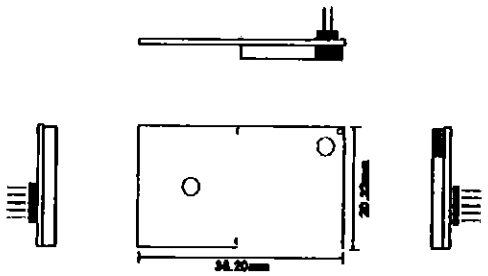
Wireless High Frequency Transceiver Module (RF GFSK)

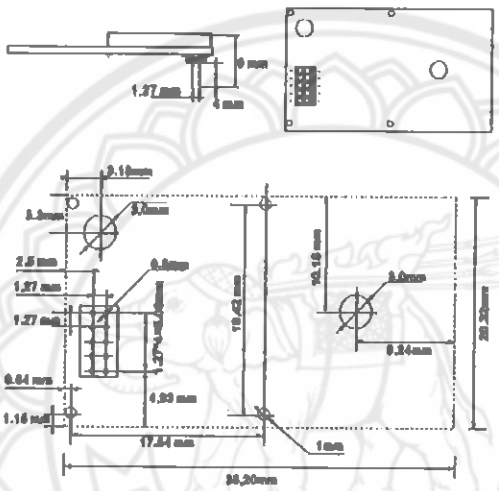


➢ **Model : TRW-24G (2.4GHz)**

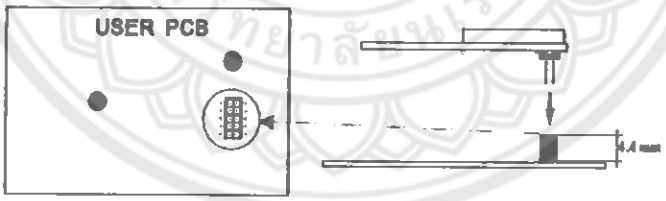
* Frequency Range: 2.4~2.527GHz
* Modulate Mode: GFSK
* Work Voltage: 3V
* Channel: 128
* Output Power: 0dBm
* Data Rate: 1Mbps; 250Kbps
* Operating Temperature: -40~+85 Centigrade
* The longest range : 280m (250Kbps); 150m (1Mbps)
* No dead spaces in reception.
* Built in antenna.
* Competitive price.
* Apply for various type of products: Wireless Joysticks, Wireless Speaker, Wireless Earphone , Wireless Cell phone , Wireless Intercom , Wireless Mouse, Wireless Keyboard and Data Communication.

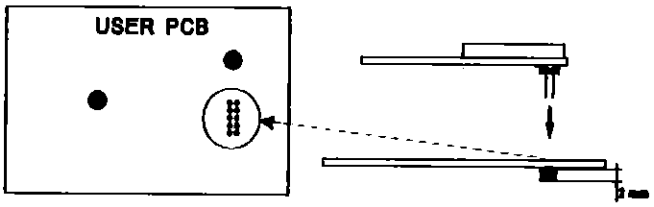
➢ **Graph:**





Reference hole position for PCB mounting(Bottom view)



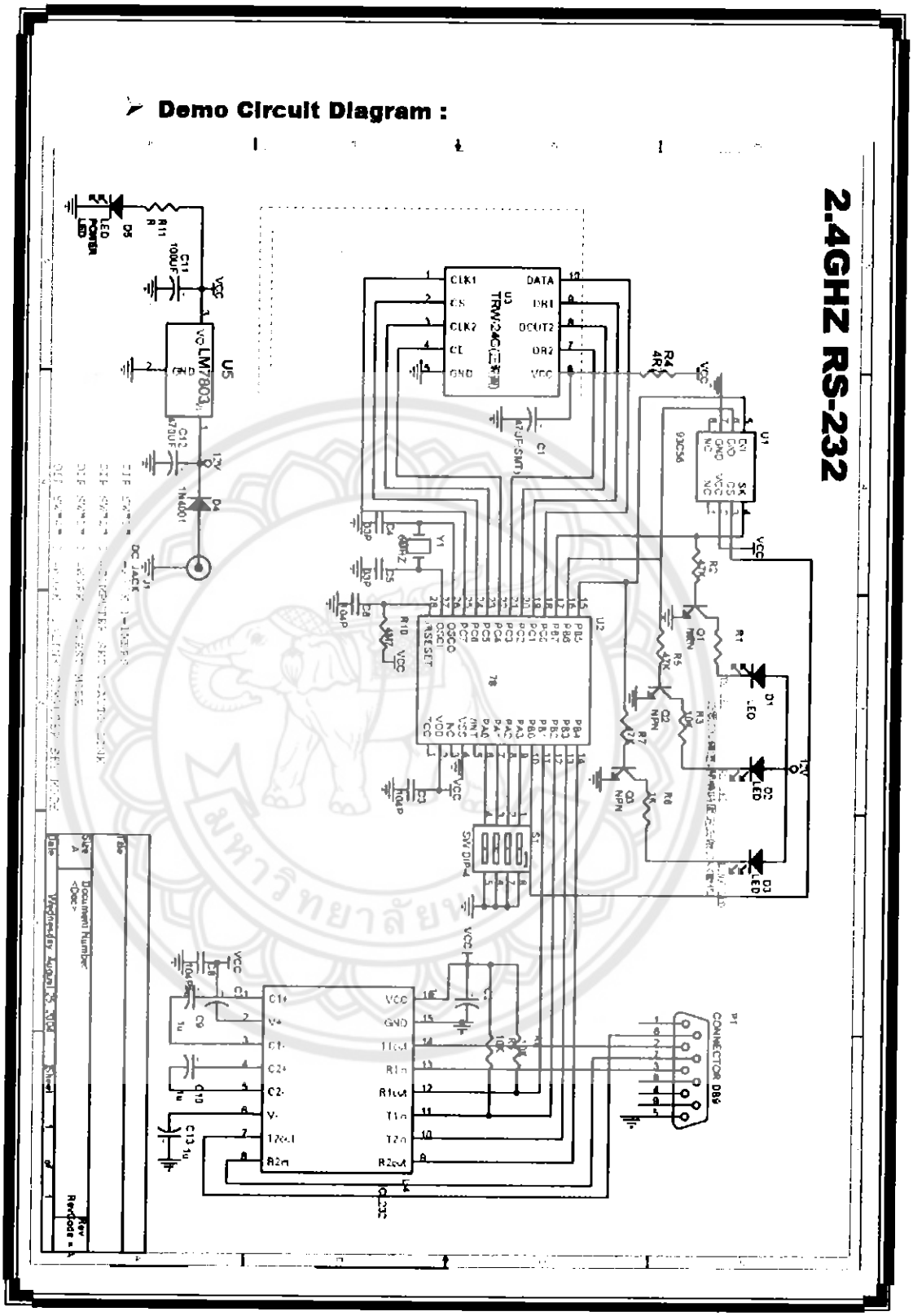


➢ Specification :

Conditions: VDD=+3V,VSS=0V,T_A=-40 centigrade to +85 centigrade

Symbol	Parameter(condition)	Min	Typ	Max	Unit
VDD	Supply voltage	1.9	3.0	3.6	V
TEMP	Operating temperature	-40	+27	+85	Centigrade
f_{op}	Operating frequency	2400		2527	MHz
R_{GFSK}	Data rate direct mode	250		1000	Kbps
$F_{CHANNEL}$	Channel spacing		1		MHz
I_{VDD}	Supply current one channel 250Kbps		18		mA
I_{VDD}	Supply current one channel 1000Kbps		19		mA
I_{VDD}	Supply current two channels 250Kbps		23		mA
I_{VDD}	Supply current two channels 1000Kbps		25		mA
RX_{SENS}	Sensitivity at 0.1%BER(@250Kbps)		-90		dBm
RX_{SENS}	Sensitivity at 0.1%BER(@1000Kbps)		-80		dBm

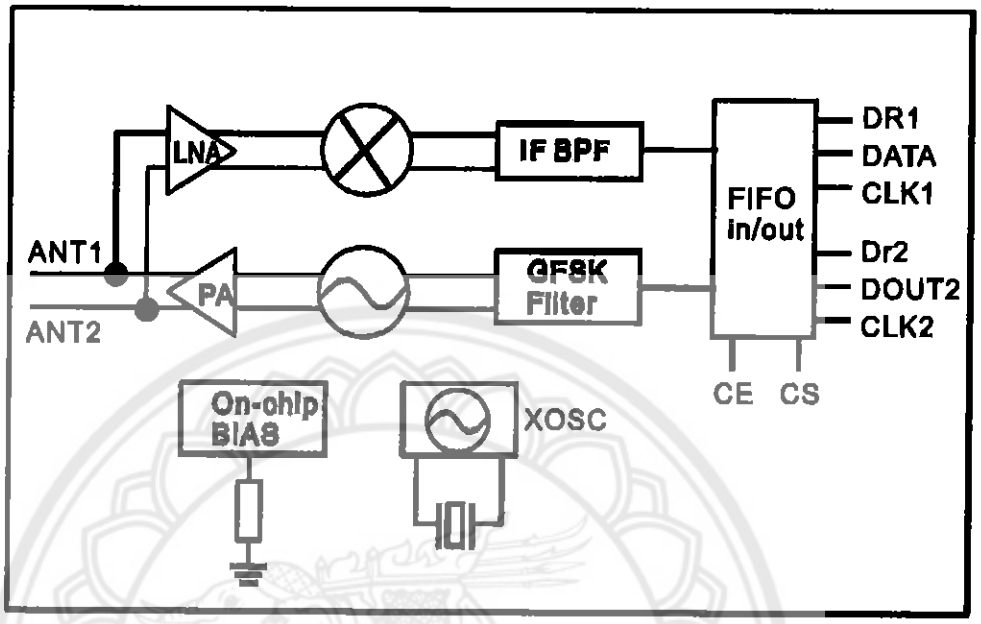
➢ **Demo Circuit Diagram :**



Conditions: VDD = +3V, VSS = 0V, T_A = -40°C to +85°C

Symbol	Parameter (condition)	Notes	Min.	Typ.	Max.	Units
	Operating conditions					
VDD	Supply voltage		1.9	3.0	3.6	V
TEMP	Operating Temperature		-40	+27	85	°C
	Digital input pin					
V_{IH}	HIGH level input voltage		VDD-0.3		VDD	V
V_{IL}	LOW level input voltage		Vss		0.3	V
	Digital output pin					
V_{OH}	HIGH level output voltage (I_{OH} -0.5mA)		VDD-0.3		VDD	V
V_{OL}	LOW level output voltage (I_{OL} 0.5mA)		Vss		0.3	V
	General RF conditions					
F_{op}	Operating frequency	1)	2400		2524	MHz
Δf	Frequency deviation			±156		kHz
R_{xBPSK}	Data rate ShockBurst™		0		1000	kbps
$F_{CHANNEL}$	Channel spacing			1		MHz
	Transmitter operation					
P_{RF}	Maximum Output Power	4)		0	+4	dBm
P_{RFC}	RF Power Control Range		16	20		dB
P_{RFCR}	RF Power Control Range Resolution				3	dB
P_{BW}	20dB Bandwidth for Modulated Carrier				1000	kHz
P_{RF2}	2nd Adjacent Channel Transmit Power 2MHz				-20	dBm
P_{RF3}	3rd Adjacent Channel Transmit Power 3MHz				-40	dBm
I_{VDD}	Supply current @ 0dBm output power	5)		13		mA
I_{VDD}	Supply current @ -20dBm output power	5)		8.8		mA
I_{VDD}	Average Supply current @ -5dBm output power, ShockBurst™	6)		0.8		mA
I_{VDD}	Average Supply current in stand-by mode	7)		12		µA
I_{VDD}	Average Supply current in power down			1		µA
	Receiver operation					
I_{VDD}	Supply current one channel 250kbps			18		mA
I_{VDD}	Supply current one channel 1000kbps			19		mA
I_{VDD}	Supply current two channels 250kbps			23		mA
I_{VDD}	Supply current two channels 1000kbps			25		mA
RX_{SENS}	Sensitivity at 0.1%BER (@ 250kbps)			-90		dBm
RX_{SENS}	Sensitivity at 0.1%BER (@ 1000kbps)			-80		dBm
C/I_{CO}	C/I Co-channel			6		dB
C/I_{1ST}	1st Adjacent Channel Selectivity C/I 1MHz			+1		dB
C/I_{2ND}	2nd Adjacent Channel Selectivity C/I 2MHz			-16		dB
C/I_{3RD}	3rd Adjacent Channel Selectivity C/I 3MHz			-26		dB
RX_B	Blocking Data Channel 2			-41		dB

➢ Circuit Description:



➢ ShockBurst™

The ShockBurst™ technology uses on-chip FIFO to clock in data at a low data rate and transmit at a very high rate thus enabling extremely power reduction.

When operation the TRW-24G in ShockBurst™ ,you gain access to the high data rates(1 Mbps)offered by the 2.4GHz band without the need of a costly, high-speed micro controller (MCU) for data processing.

By putting all high speed signal processing related to RF protocol on-chip, the TRW-24G offers the following benefits:

- Highly reduced current consumption.
- Lower system cost (facilitates use of less expensive micro controller).
- Greatly reduced risk of 'on-air' collisions due to short transmission time.

The TRW-24G can be programmed using a simple 3-wire interface where the data rate is decided by the speed of the micro controller.

By allowing the digital part of the application to run at low speed while maximizing the data rate on the RF link, the nRF ShockBurst™ mode reduces the average current consumption in applications considerably.

➢ ShockBurst™ principle:

When the TRW-24G is configured in ShockBurst™, TX or RX operation is conducted in the following way (10 kbps for the example only).

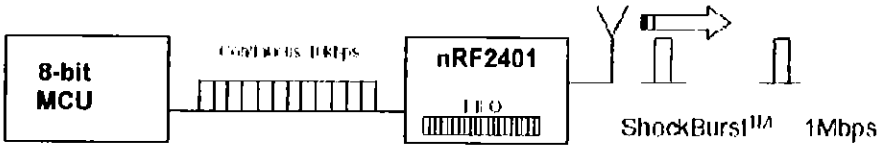


Figure 4 Clocking in data with MCU and sending with ShockBurst™ technology

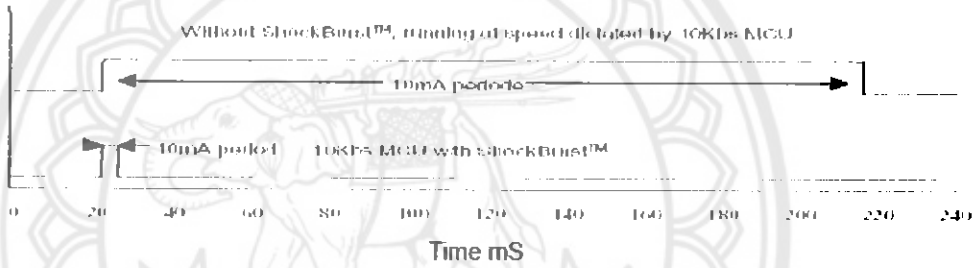


Figure1 Current consumption with & without ShockBurst™ technology

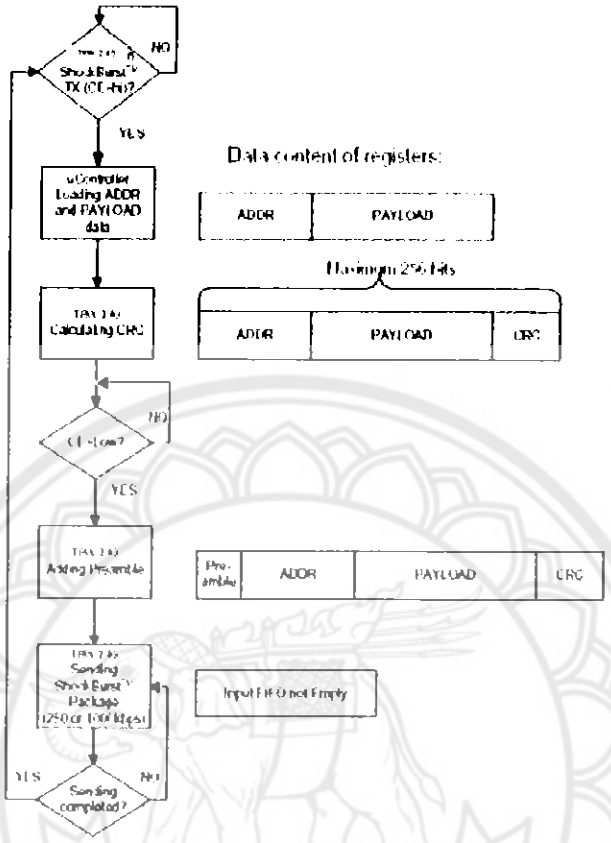


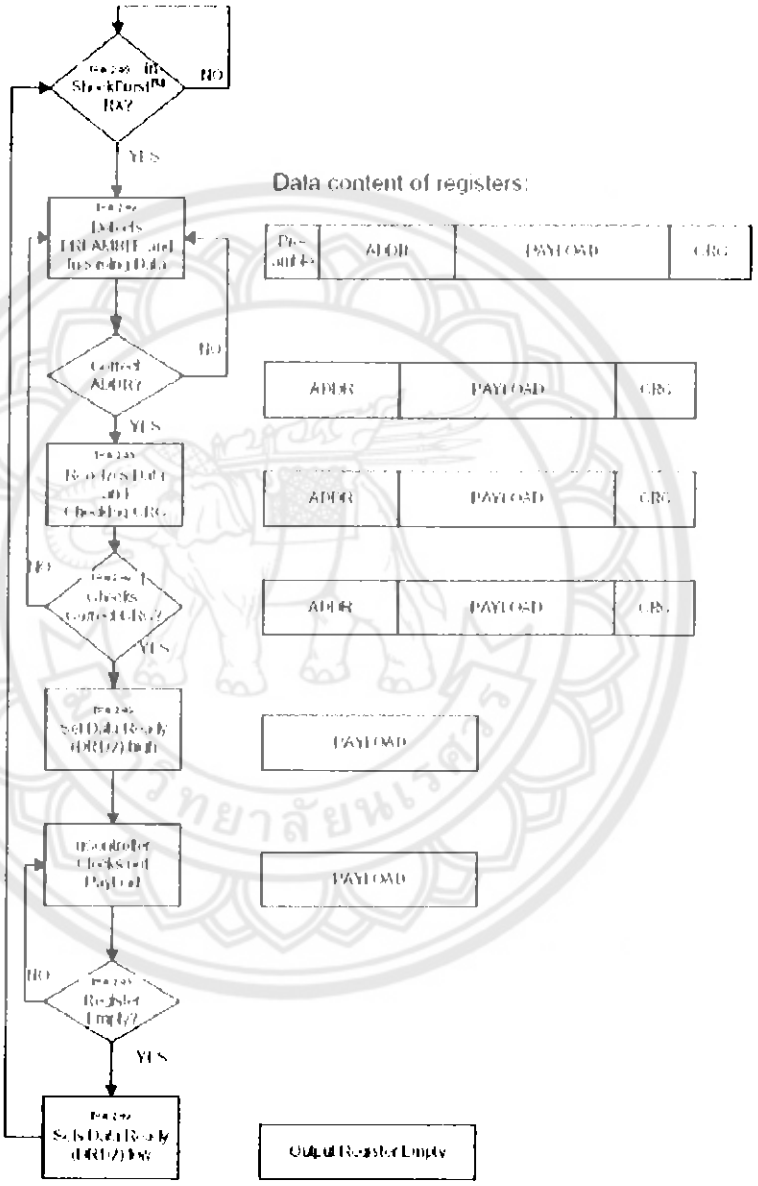
Figure 2 Flow Chart shockBurst™ Transmit of TRW-24G

➢ nRF2401 ShockBurst™ Transmit:

MCU interface pins:CE,CLK1,DATA

1. When the application MCU has data to send, set CE high. This activates TRW-24G on-board data processing.
2. The address of the receiving node(RX address) and payload data is clocked into the TRW-24G. The application protocol or MCU sets the speed <1Mbps(ex:10kbps)>.
3. MCU sets CE low, this activates a TRW-24G ShockBurst™ transmission.
4. TRW-24G ShockBurst™:

- RF front end is powered up.
- RF package is completed (preamble added, CRC calculated).
- Data is transmitted at high speed (250kbps or 1 Mbps configured by user).
- TRW-24G return to stand-by when finished.



➢ TRW-24G ShockBurst™ Receive:

MCU interface pins: CE, DR1, CLK1 and DATA (one RX channel receive)

1. Correct address and size of payload of incoming RF packages are set when TRW-24G is configured to ShockBurst™ RX.
2. To activate RX , set CE high.
3. After 200us settling, TRW-24G is monitoring the air for incoming communication.
4. When a valid package has been received (correct address and CRC found), TRW-24G removes the preamble, address and CRC bits.
5. TRW-24G then notifies (interrupts) the MCU by setting the DR1 pin high.
6. MCU may (or may not) set the CE low to disable the RF front end (low current mode).
7. The MCU will clock out just the payload data at a suitable rate (ex,10 kbps).
8. When all payload data is retrieved TRW-24G sets DR1 low again, and is ready for new incoming data package if CE is kept high during data download. If the CE was set low, a new start up sequence can begin, see Figure 12.

➢ Duoceiver™ Simultaneous Two Channel Receive Mode:

In both ShockBurst™ modes the TRW-24G can facilitate simultaneous reception of two parallel independent frequency channels at the maximum data rate.
This means:

- TRW-24G can receive data from two 1Mbps transmitters (ex: TRW-24G or TRW-24G) 8MHz (8 frequency channels) apart through one antenna interface.
- The output from the two data channels is fed to two separate MCU interfaces.
- Data channel 1:CLK1,DATA,and DR1
- Data channel 2:CLK2,DOUT2,and DR2
- DR1 and DR2 are available only in ShockBurst™.

The TRW-24G DuoCeiver™ technology provides 2 separate dedicated data channels for RX and replaces the need for two, stand alone receiver systems.

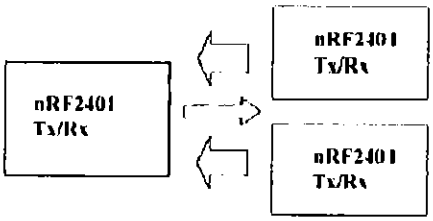


Figure 4 Simultaneous 2 channel receive on TRW-24G

There is one absolute requirement for using the second data channel. For the TRW-24G to be able to receive at the second data channel the frequency channel must be 8MHz higher than the frequency of data channel 1. The TRW-24G must be programmed to receive at the frequency of data channel 1. No time multiplexing is used in TRW-24G to fulfil this function. In direct mode the MCU must be able to handle two simultaneously incoming data packets if it is not multiplexing between the two data channels. In ShockBurst™ it is possible for the MCU to clock out one data channel at a time while data on the other data channel waits for MCU availability, without any lost data packets, and by doing so reduce the needed performance of the MCU.

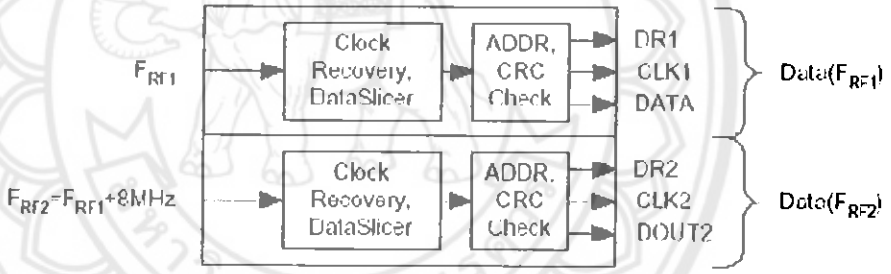


Figure 5 DuoCeiver™ with two simultaneously independent receive channels

➢ Device Configuration:

All configuration of theTRW-24G is done via 3-wire interface to a single configuration register. The configuration word can be up to 15 bytes long for ShockBurst™.

➢ Configuration or ShockBurst™ operation:

The configuration word in ShockBurst™ enables theTRW-24G to handle the RF protocol. Once the protocol is completed and loaded intoTRW-24G only one byte, bit [7:0], needs to be updated during actual operation.
The configuration blocks dedicated to ShockBurst™ is as follows:

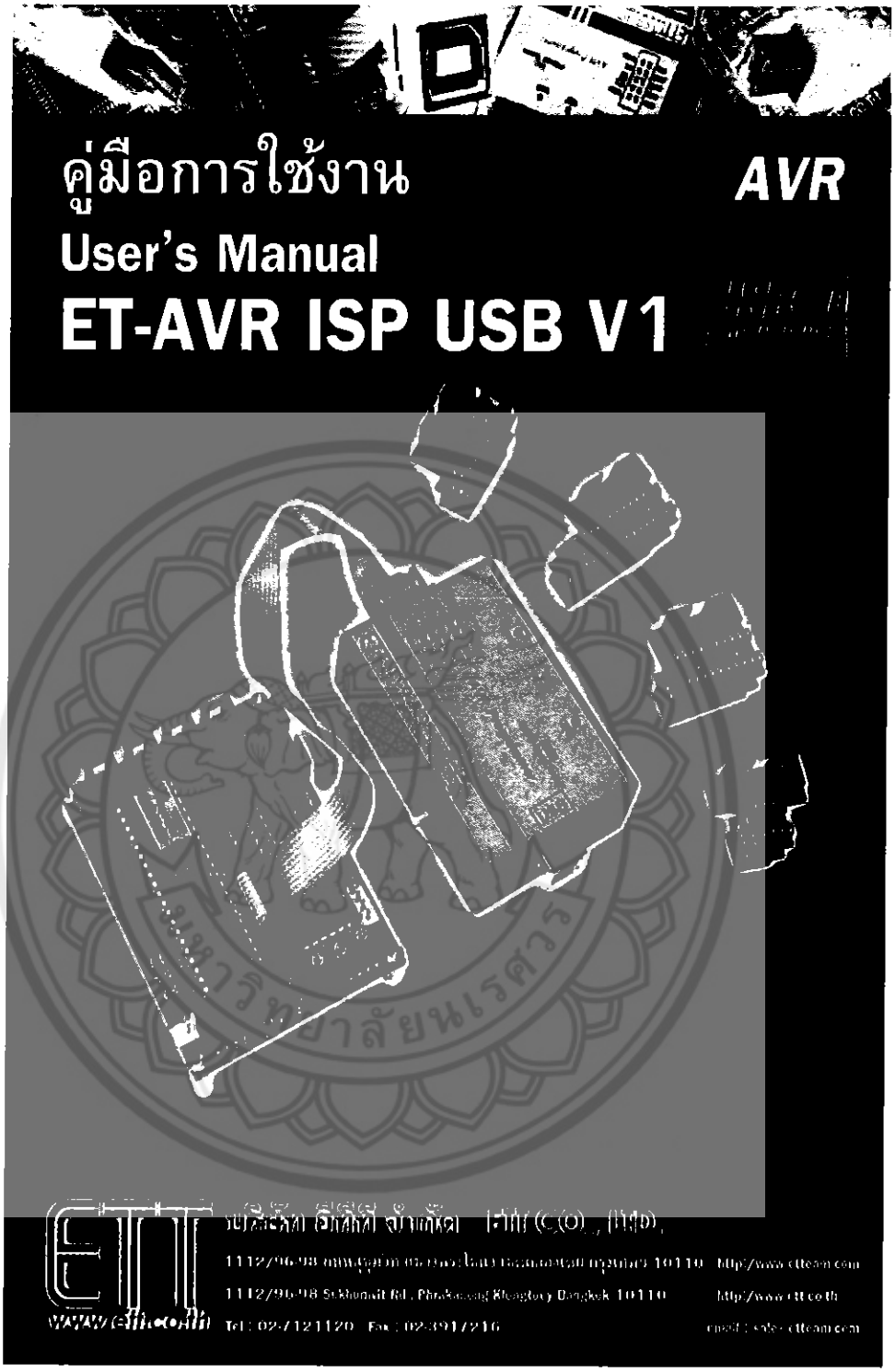
- Payload section width: Specifies the number of payload bits in a RF package. This enables theTRW-24G to distinguish between payload data and the CRC bytes in a received package.
- Address width: Sets the number of bits used for address in the RF package, This enables theTRW-24G to distinguish between address and payload data.
- Address (RX Channel 1 and 2): Destination address for received data.
- CRC: Enables TRW-24G on-chip CRC generation and de-coding.

NOTE:
These configuration blocks, with the exception of the CRC, are dedicated for the packages that a TRW-24G is to receive.
In TX mode, the MCU must generate an address and a payload section that fits the configuration of theTRW-24G that is to receive the data.
When using theTRW-24G on-chip CRC feature ensure that CRC is enabled and uses the same length for both the TX and RX devices.

Figure 10 Data packet set-up

ภาคผนวก ข

คู่มือการใช้งาน ET-AVR ISP USB V1


คู่มือการใช้งาน
AVR
User's Manual
ET-AVR ISP USB V1

ET-AVR ISP USB V1.0

ET-AVR ISP USB V1.0 เป็นบอร์ดที่ออกแบบมาเพื่อใช้ในการดาวน์โหลด Hex File ให้กับ MCU ตระกูล AVR ของ Atmel โดยผ่านทาง ISP Interface โดยต้องใช้ร่วมกับโปรแกรม AVR Studio 4.XX หรือซอฟแวร์อื่นๆ ที่รองรับ AVR ISP ซึ่งปัจจุบันมีให้เลือกใช้หลายตัวมาก

คุณสมบัติของ ET-AVR ISP USB V1.0

1. มีคุณสมบัติเทียบเท่า AVR ISP ของ ATMEL
2. โปรแกรมผ่านทาง ISP Interfacce
3. สามารถอัพเกรด Firmware โดยตรงผ่านโปรแกรม AVR Studio 4 โดยไม่ต้องใช้เครื่องโปรแกรมจากภายนอก เพื่อให้สามารถใช้กับ MCU เบอร์ใหม่ ๆ ได้ ซึ่ง Firmware จะติดมากับโปรแกรม AVR studio 4
4. สามารถใช้ได้กับระบบไฟเลี้ยงตั้งแต่ 2.7V – 5.5V
5. การติดต่อสื่อสารผ่านพอร์ต USB
6. มีสัญญาณ Clock ไว้จ่ายให้ MCU ในกรณีที่ Fuse Bit เลือกแหล่งของสัญญาณ Clock ผิด
7. มี LED แสดงการทำงาน Power , Activity , Usb
8. สามารถใช้กับซอฟแวร์ต่างๆ ที่รองรับ AVR ISP เช่น AVR Studio,WinAVR,ICC AVR ,CodeVision ,BASCOM-AVR เป็นต้น

เบอร์ของไมโครคอนโทรลเลอร์ AVR ที่สามารถใช้ได้กับ ET-AVR ISP USB V1.0 (AVR studio 4.12 SP4)

AT90S1200	ATtiny12	ATmega8	AT86RF401
AT90S2313	ATtiny13	ATmega16	AT89S51
AT90S/LS2323	ATtiny15	ATmega32	AT89S52
AT90S/LS2343	ATtiny22	ATmega48	AT90CAN32
AT90S/LS2333	ATtiny2313	ATmega48P	AT90CAN128
AT90S4414	ATtiny24	ATmega64	AT90PWM2
AT90S/LS4433	ATtiny25	ATmega88	AT90PWM3

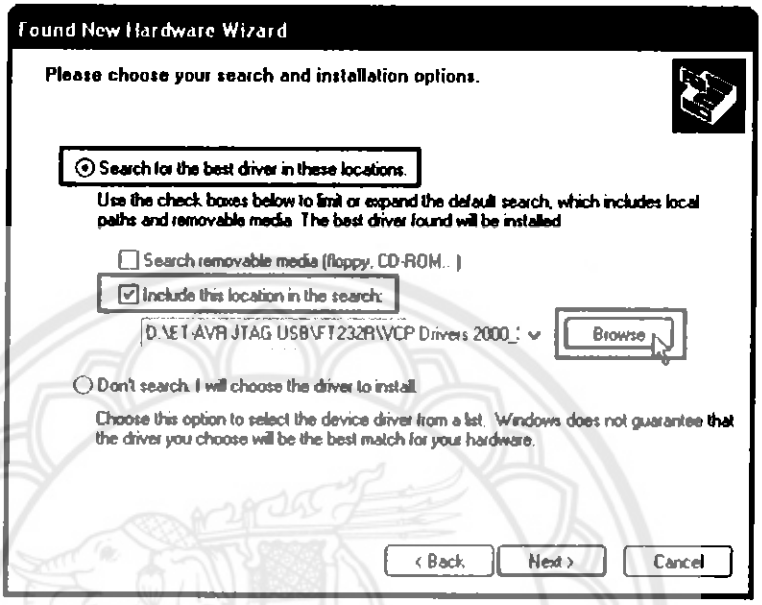
AT90S/LS4434	ATtiny26	ATmega103	
AT90S8515	ATtiny44	ATmega128	
AT90S/LS8535	ATtiny45	ATmega1280	
	ATtiny461	ATmega1281	
	ATtiny84	ATmega161	
	ATtiny85	ATmega162	
	ATtiny861	ATmega163	
		ATmega164P	
		ATmega165	
		ATmega165P	
		ATmega169	
		ATmega169P	
		ATmega323	
		ATmega324P	
		ATmega325	
		ATmega325P	
		ATmega329	
		ATmega329P	
		ATmega644P	
		ATmega645	
		ATmega649	
		ATmega2560	
		ATmega2561	
		ATmega3250	
		ATmega3250P	
		ATmega3290	
		ATmega3290P	
		ATmega6450	
		ATmega6490	
		ATmega8515	
		ATmega8535	

การติดตั้ง Driver ของ ET- AVR ISP USB V1.0

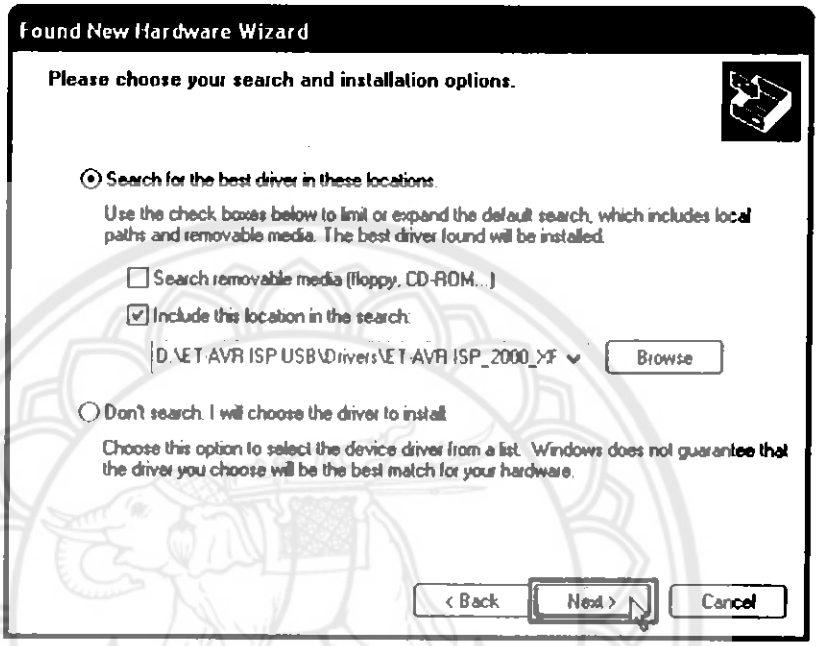
1. ทำการเชื่อมต่อ ET-AVR ISP USB V1.0 เข้ากับคอมพิวเตอร์ทางพอร์ต USB โดยไม่ต้องต่อ Target Board จากนั้นวินโดวส์จะตรวจพบฮาร์ดแวร์ใหม่ดังรูป

2. จากนั้นจะปรากฏหน้าต่าง Found New Hardware Wizard ให้เลือกที่ Install from a list or specific location (Advanced) และคลิก Next

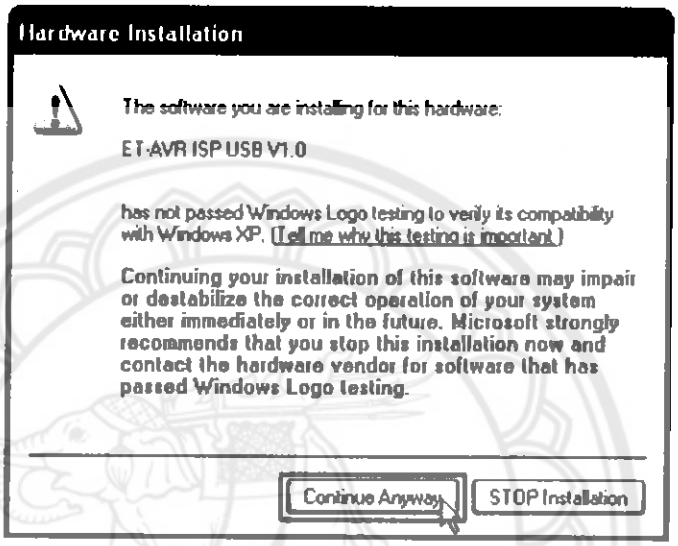
3. ทำการเลือกดังรูป และคลิกปุ่ม Browse เพื่อระบุตำแหน่งที่เก็บ Driver จากนั้นคลิก OK

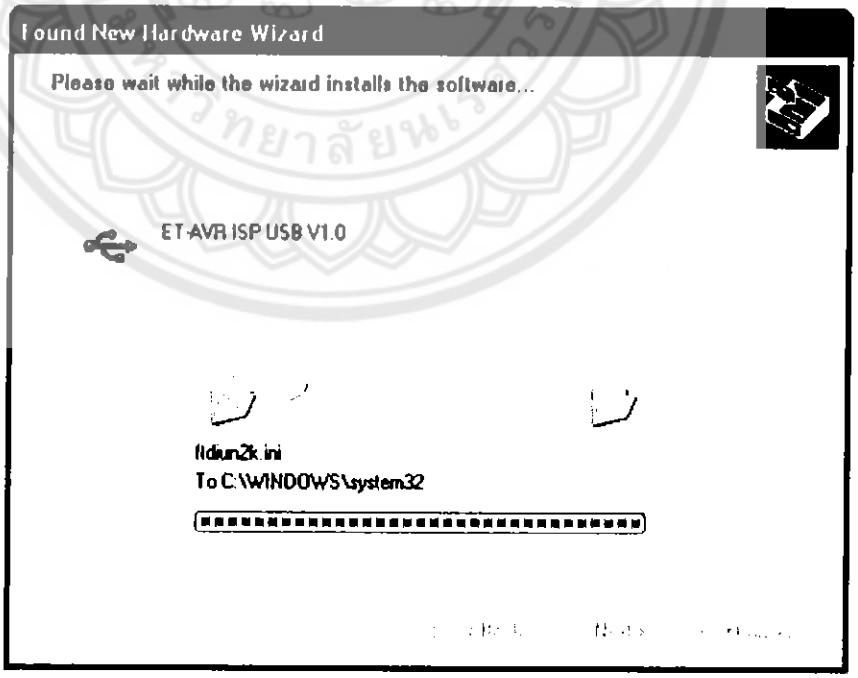


4. เมื่อทำการเลือกเรียบร้อยแล้วให้คลิกปุ่ม Next จากนั้นวินโดวส์จะทำการหาฮาร์ดแวร์เพื่อที่จะทำการติดตั้ง Driver



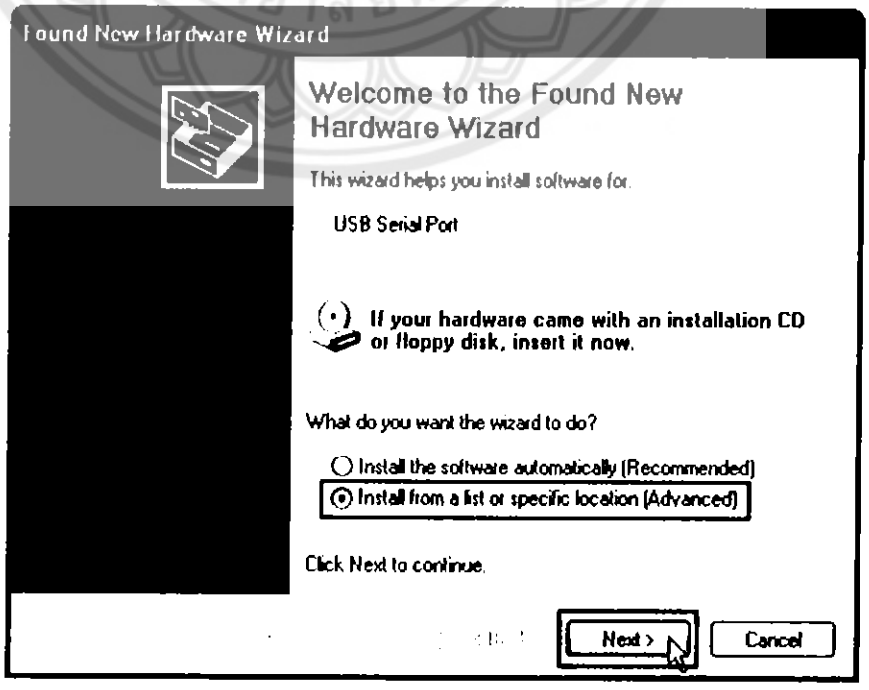
5. จากนั้นจะปรากฏหน้าต่าง Hardware Installation บอกว่าฮาร์ดแวร์ไม่ได้ผ่านการทดสอบของวินโดวส์ ให้ทำการคลิกที่ Continue Anyway ซึ่งวินโดวส์จะทำการติดตั้ง Driver ของ ET-AVR ISP USB V1.0



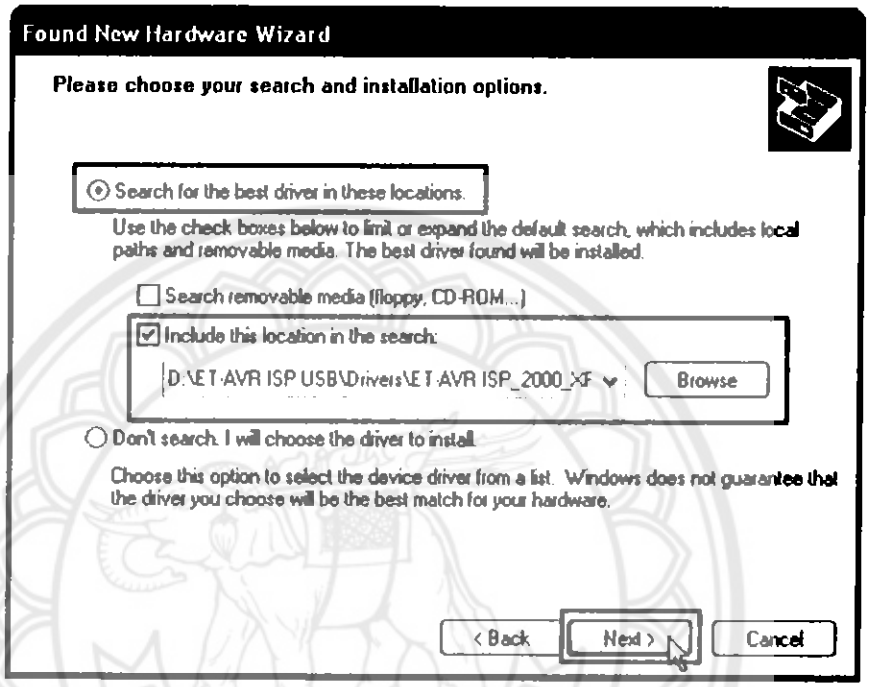


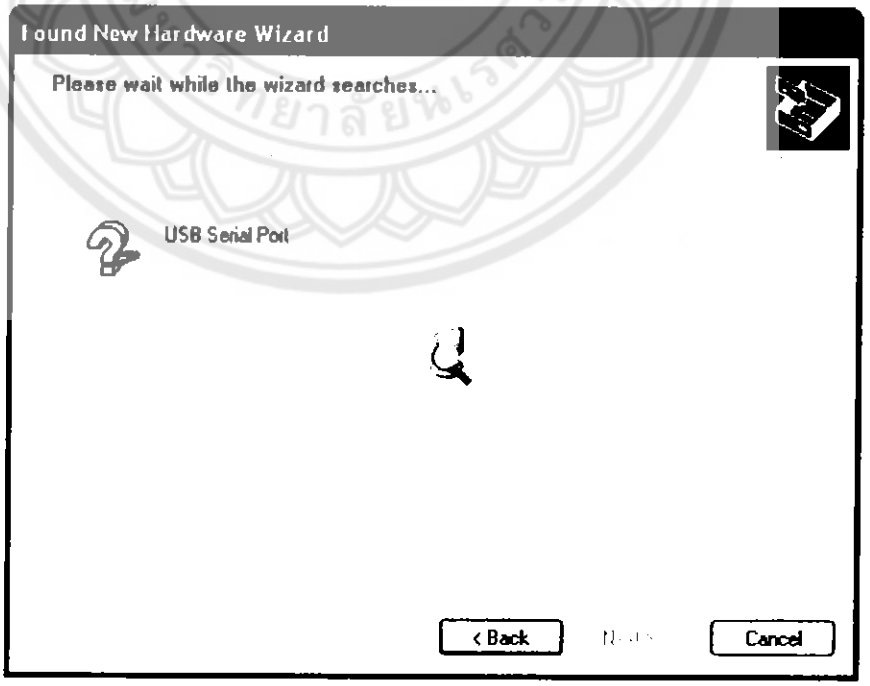
6. เมื่อปรากฏหน้าต่างว่าได้ทำการติดตั้งเรียบร้อยแล้วให้คลิก Finish

7. จากนั้นไม่นานตัว Driver จะมีการสร้างพอร์ตอนุกรมเสมือนขึ้นมาและมีหน้าต่างให้ติดตั้ง Driver ของ USB Serial Port ดังรูป ให้ทำการเลือกเหมือนข้อที่ผ่านมาจากนั้นคลิก Next

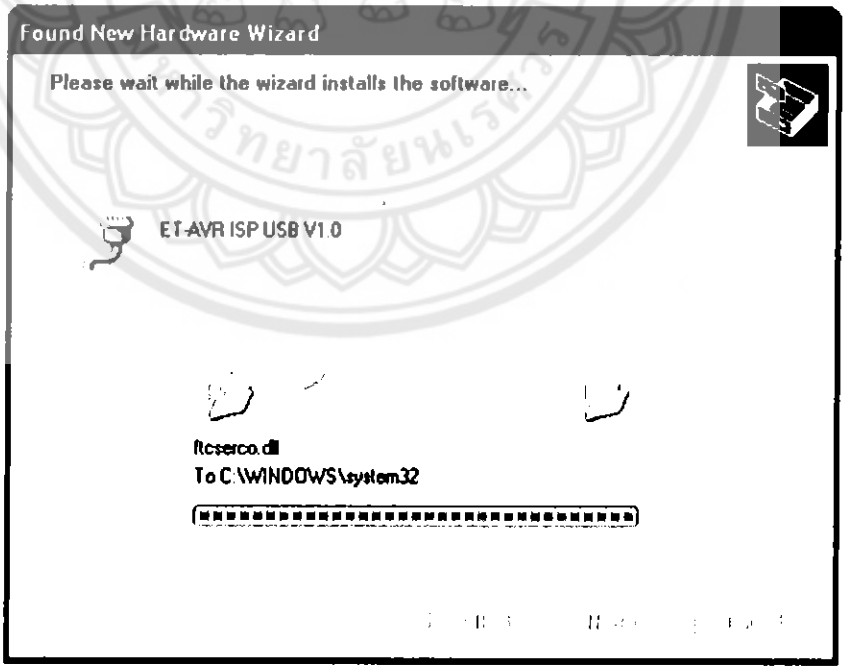


8. เลือกตำแหน่งที่ตั้งของ Driver ซึ่งปกติจะจำค่าเดิมไว้อยู่แล้วให้คลิก Next ได้เลย จากนั้นวินโดวส์จะทำการหาฮาร์ดแวร์เพื่อที่จะทำการติดตั้ง Driver

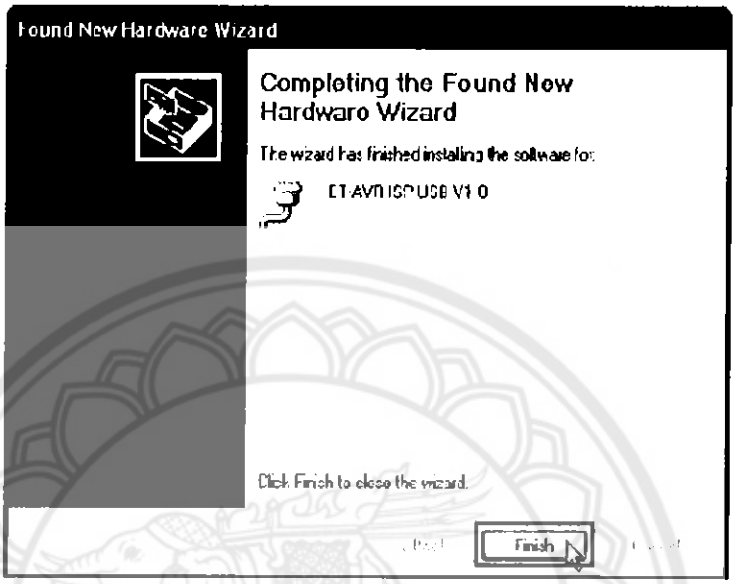




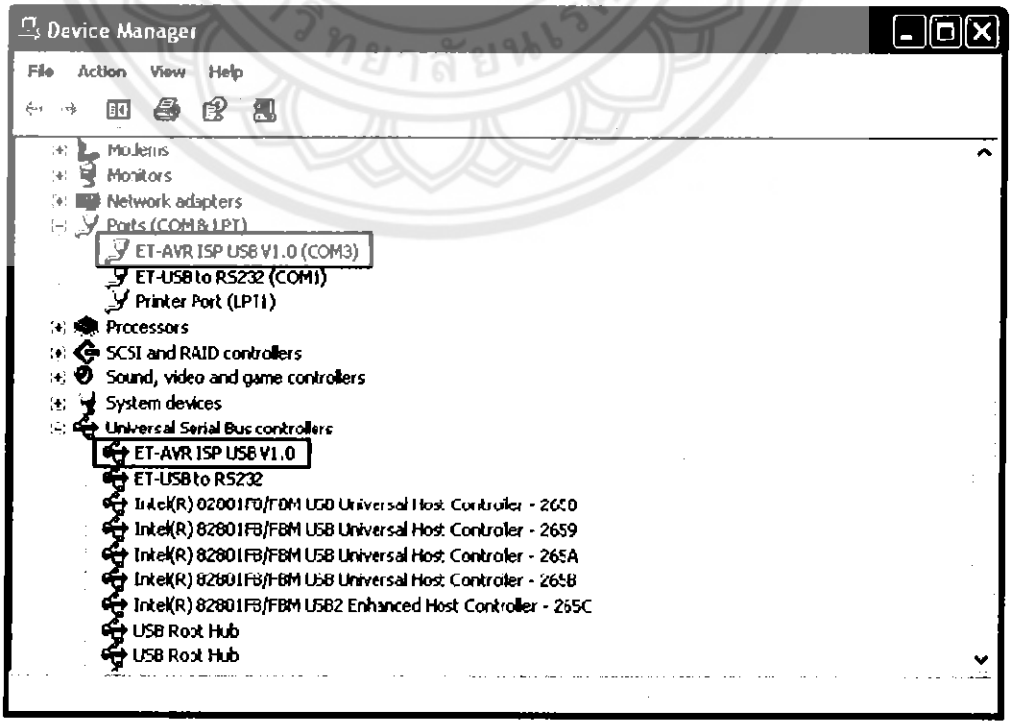
9. จากนั้นจะปรากฏหน้าต่าง Hardware Installation บอกว่าฮาร์ดแวร์ไม่ได้ผ่านการทดสอบของวินโดวส์ให้ทำการคลิกที่ Continue Anyway ซึ่งวินโดวส์จะทำการติดตั้ง Driver ของ ET-AVR ISP USB V1.0



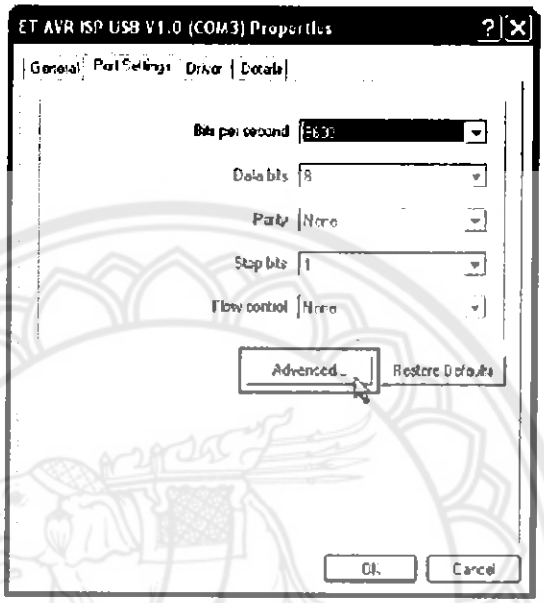
10. เมื่อปรากฏหน้าต่างว่าได้ทำการติดตั้งเรียบร้อยแล้วให้คลิก Finish



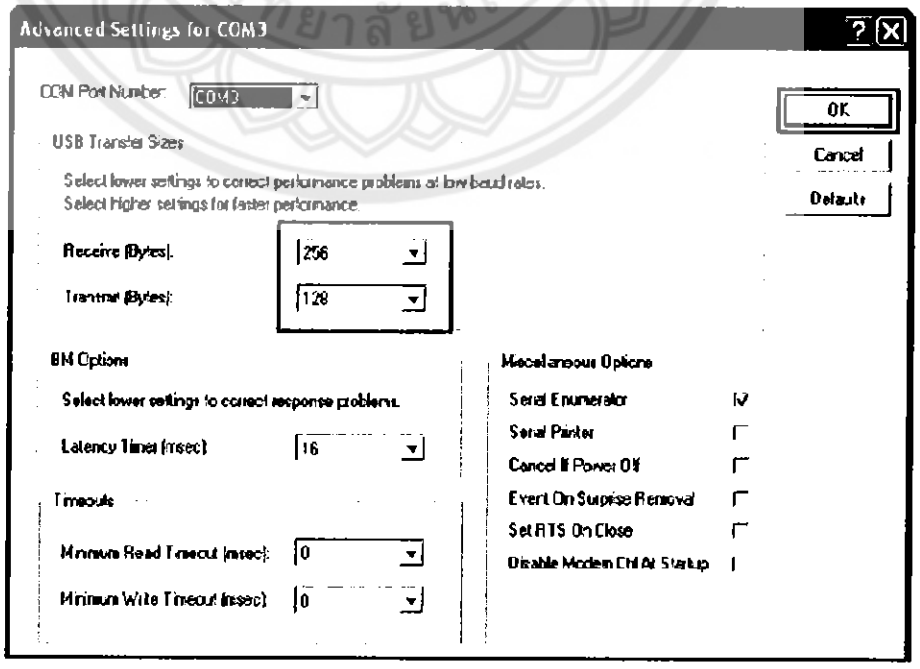
11. เราสามารถที่จะตรวจสอบว่าได้ติดตั้ง Driver ของ ET-AVR ISP USB V1.0 เสร็จสมบูรณ์หรือไม่ โดยดูที่ Control Panel → System เลือกแท็ป Hardware และเลือกที่ Device Manager ซึ่งจะเห็นรายการฮาร์ดแวร์ ET-AVR ISP USB V1.0 ดังรูป



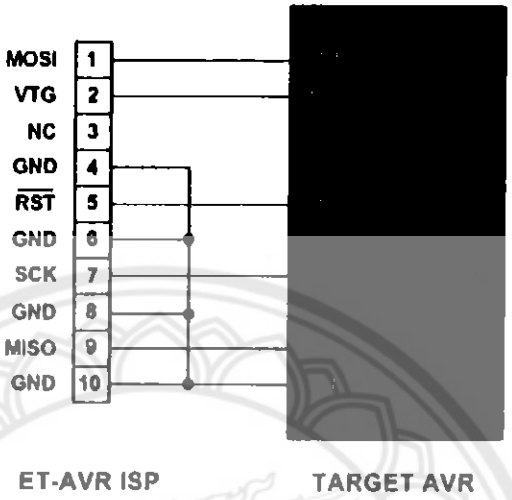
12. ดับเบิลคลิกที่ ET- AVR ISP USB V1.0 ตรงส่วนของ Ports(COM&LPT) จากนั้นจะปรากฏหน้าต่าง Properties ดังรูป ให้เลือกมาที่ Port Setting และทำการคลิกที่ปุ่ม Advanced...



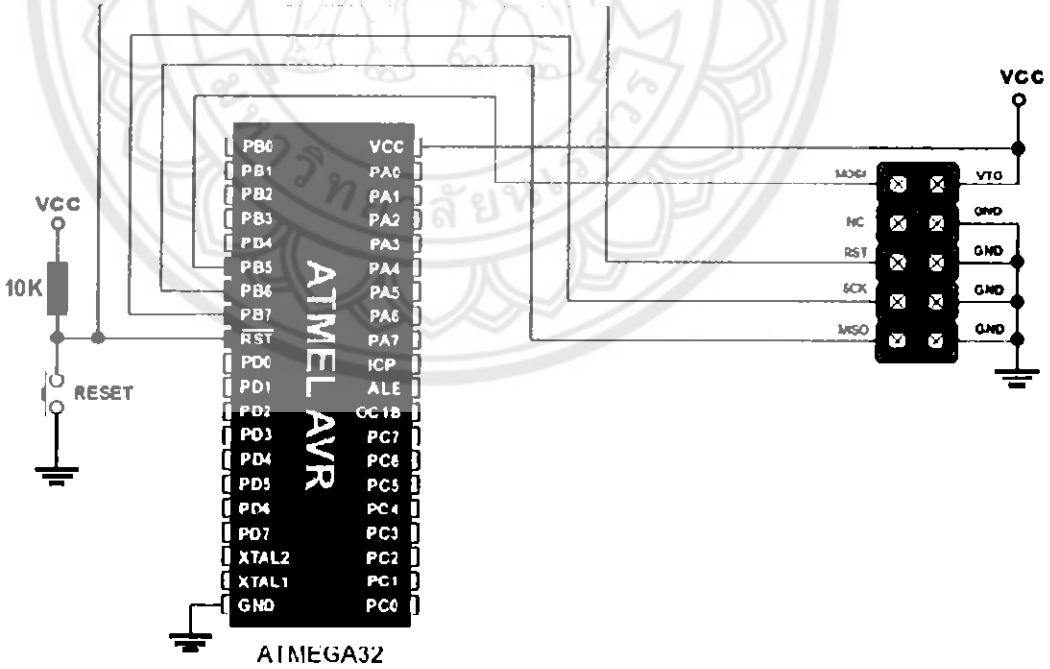
13. กำหนดค่า Receive(Bytes) และ Transmit(Bytes) ดังรูป และคลิกที่ OK เพื่อยืนยันการเปลี่ยนแปลง จากนั้นให้ทำการรีสตาร์ทเครื่องคอมพิวเตอร์หรือสแกนหาฮาร์ดแวร์ใหม่



การเชื่อมต่อ ET-AVR ISP กับไมโครคอนโทรลเลอร์ AVR

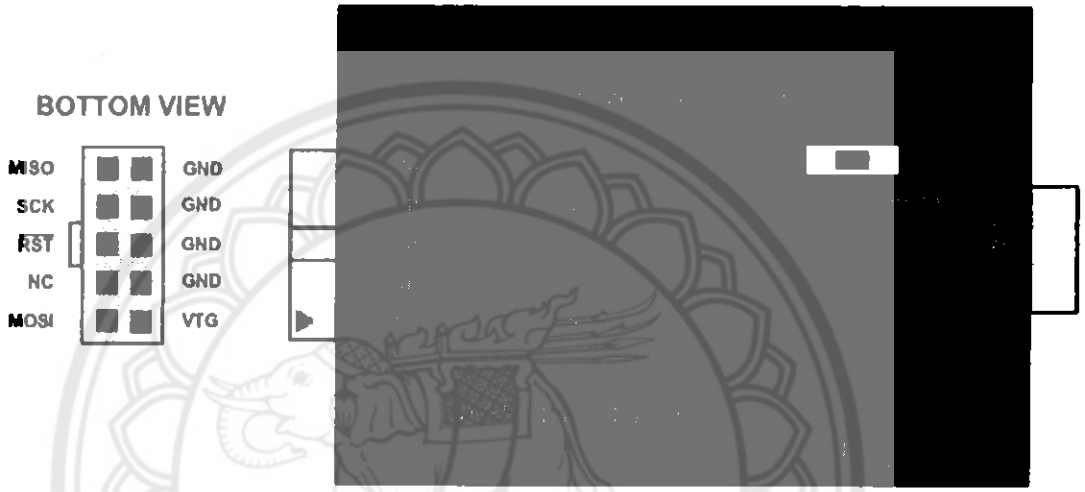


แสดงการเชื่อมต่อ ET-AVR ISP กับ ไมโครคอนโทรลเลอร์ AVR



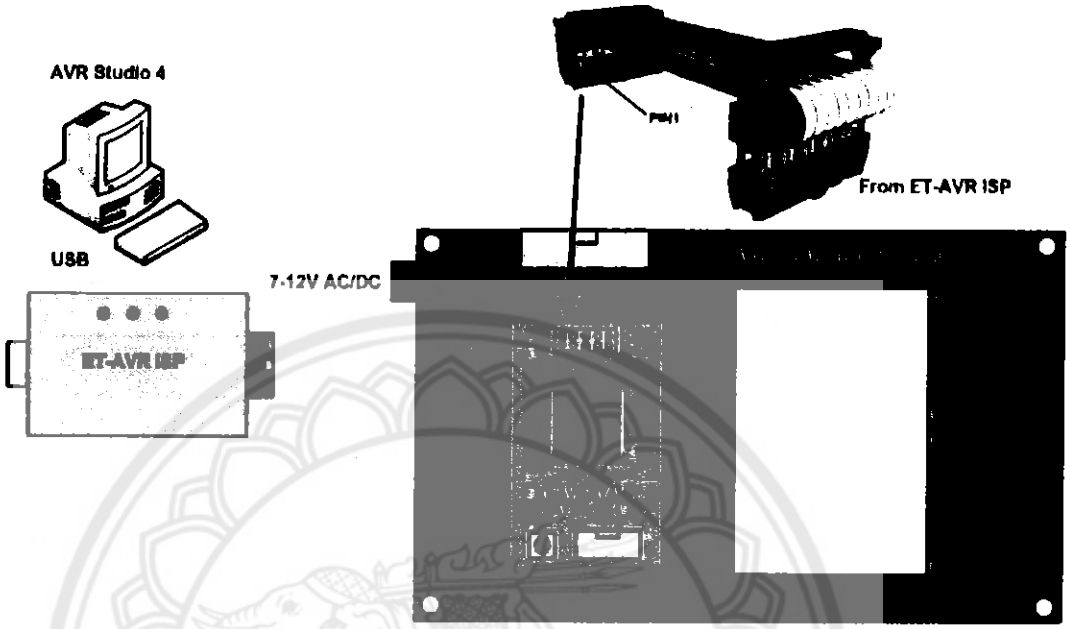
แสดงการเชื่อมต่อกับ AVR เบอร์ ATMEGA32

การต่อใช้งานจะใช้สายสัญญาณ MISO , MOSI , SCK , RST , VTG , GND ซึ่งจะต้องต่อกับขา ISP Interface ของ AVR ซึ่งท่านจะต้องมีไฟเลี้ยงบอร์ด Target ต่างหากไม่สามารถใช้ไฟเลี้ยงจาก ET-AVR ISP USB V1.0 ได้ และที่สำคัญสวิทช์เลือกโหมดการทำงานต้องอยู่ในตำแหน่ง NORMAL เท่านั้น



แสดงตำแหน่งขาสัญญาณของ ET-AVR ISP USB V1.0 โดยมองจากด้านล่าง

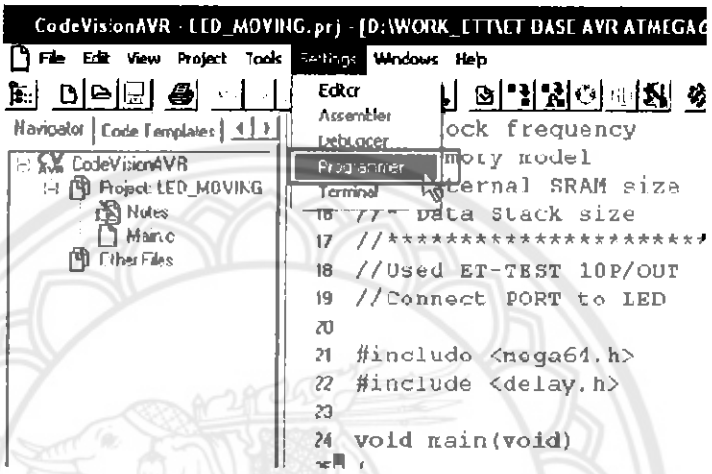
ซึ่งถ้านำ ET-AVR ISP USB V1.0 มาใช้กับบอร์ดรุ่นใหม่ๆ ของ ETT เช่น ET-AVR STAMP MEGA64/128,ET-BASE MEGA64/128,ET-BASE TINY2313 ซึ่งได้เตรียมคอนเน็คเตอร์ ISP ไว้แล้ว ก็สามารถเสียบใช้งานได้ทันที แต่ข้อควรระวังคือถ้าใช้กับบอร์ด ET-AVR STAMP MEGA64/128 เพราะว่าคอนเน็คเตอร์ ISP ของรุ่นนี้ไม่มีตัวล็อคตำแหน่งสายอาจจะทำให้เสียบกลับทางได้ โดยจะต้องสังเกตตำแหน่งขา 1 ซึ่งก็คือขา MOSI ให้ตรงกัน ซึ่งจะต้องให้เครื่องหมายสามเหลี่ยมของสายและบอร์ดตรงกันดังรูป



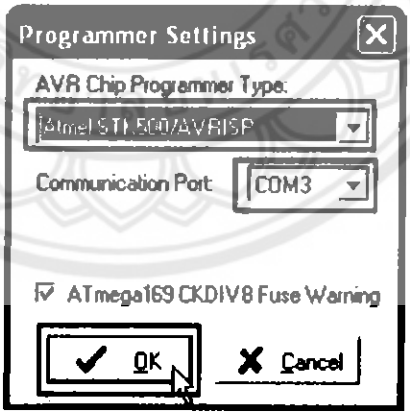
แสดงการเชื่อมต่อ ET-AVR JTAG กับ ET-AVR STAMP64

การใช้งาน ET-AVR ISP USB V1.0 ร่วมกับโปรแกรม CodeVisionAVR

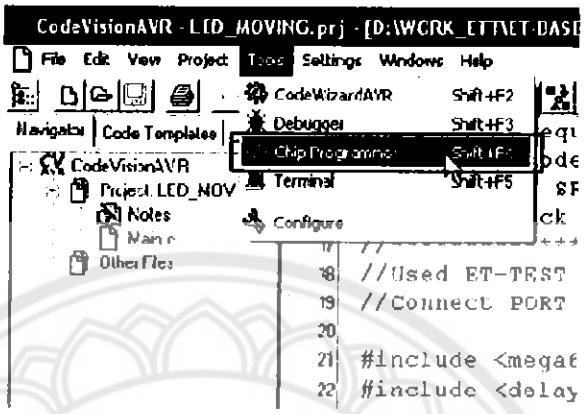
1. ไปที่เมนู Settings >>> Programmer เพื่อเลือกเครื่องมือที่จะใช้ในการโปรแกรม



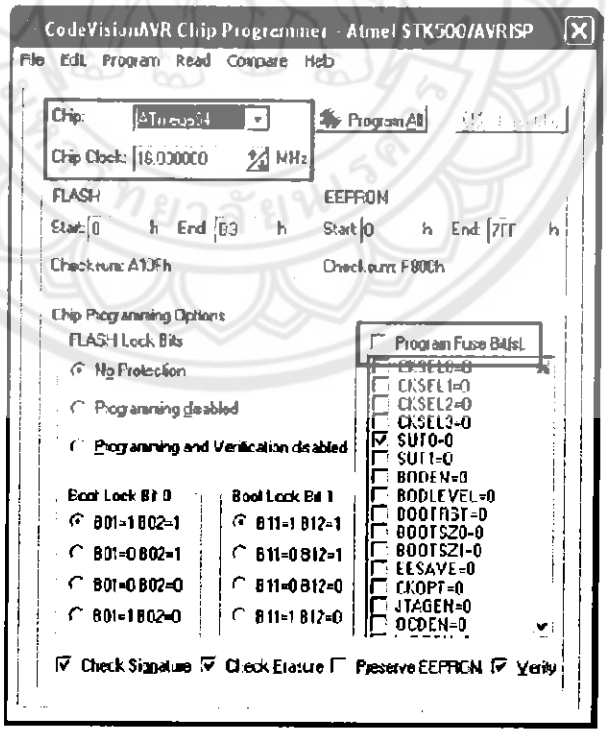
2. เลือก AVR Chip Progammer Type เป็น Atmel STK500/AVRISP และ Communication Port: ที่ ET-AVR ISP USB V1.0 ติดตั้งอยู่ จากนั้นคลิก OK



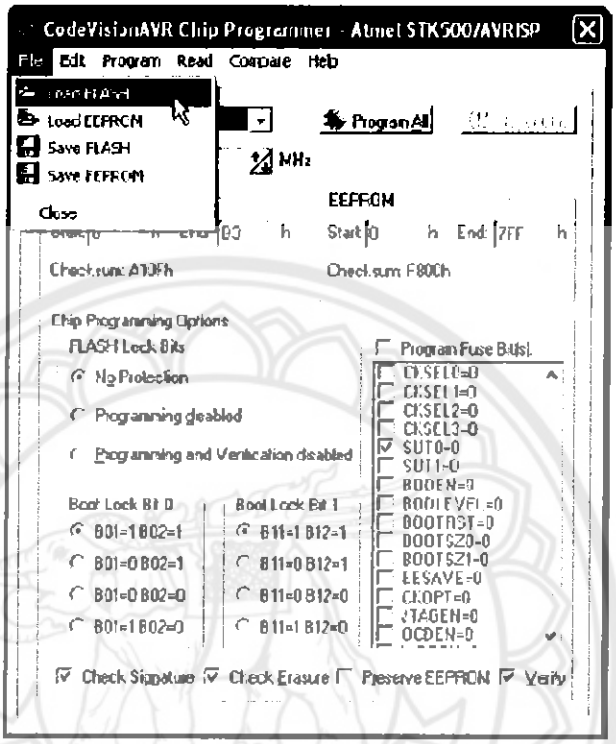
3. เลือกไปที่เมนู Tools >> Chip Programmer เพื่อทำการโปรแกรม MCU

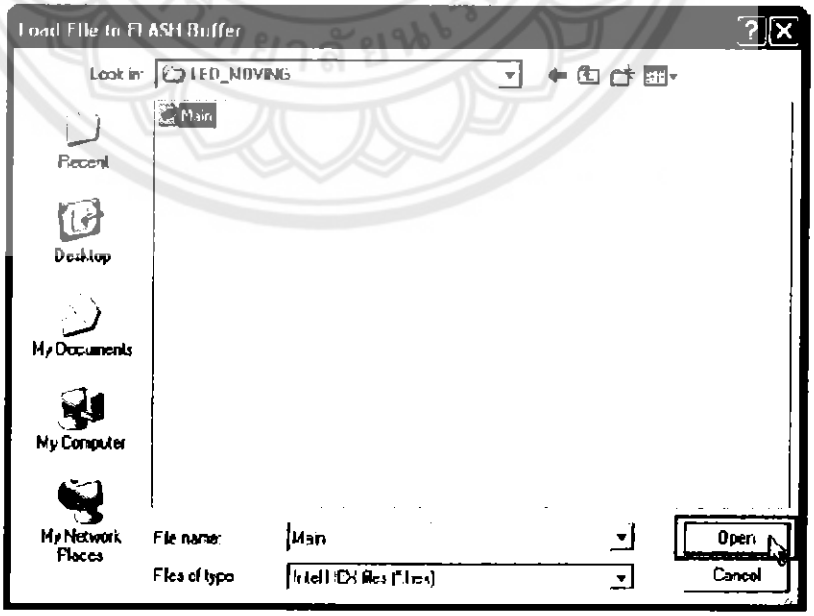


4. จากนั้นจะปรากฏหน้าต่างของโปรแกรม CodeVisionAVR Chip Programmer เลือกเบอร์ Chip และกำหนดความถี่ของคริสตอลที่ใช้งาน ส่วนช่อง Programmet Fuse Bit(s). ไม่ต้องเลือกถ้าไม่ต้องการที่จะโปรแกรม Fuse bits

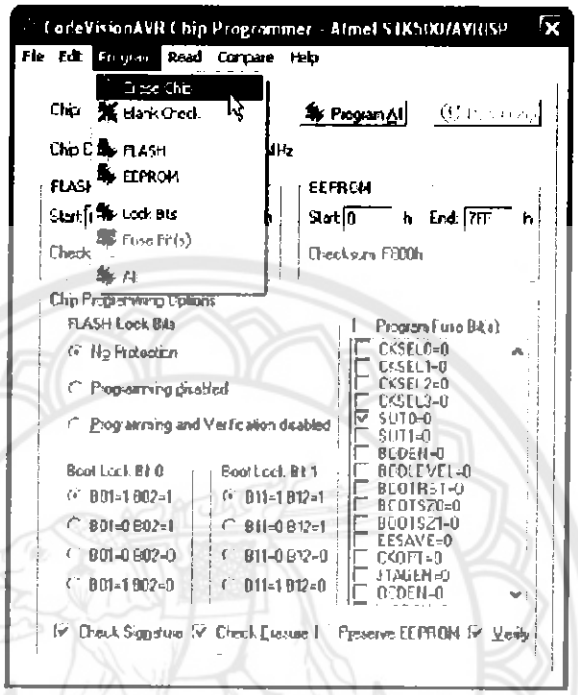


5. เลือกที่เมนู File >> Load Flash เพื่อระบุ HEX File ที่จะโปรแกรมลง MCU





6. ทำการลบข้อมูลเก่าของ MCU โดยเลือกที่เมนู Program >> Erase Chip ดังรูป



7. ทำการโปรแกรม HEX File เข้าสู่ MCU โดยเลือกที่เมนู Program >> FLASH ดังรูปเพื่อโปรแกรมเฉพาะ Flash Memory เท่านั้น