สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องวัดความดันโลหิตและอัตราการเต้นของหัวใจ ควบคุมด้วยโทรศัพท์มือถือ

โดย

นายสุรเชษฎ์ บุญงอก

ปพ.
ส848ค
2549

เลขหมู่.....
เลขทะเบียน 72703
วัน,เดือน,ปี 21 มิ.ย. 2550

b 11291598
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Digital blood pressure and pulse rate meter control by mobile phone

BY

MR.SURACHES BOONNGOG

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2006

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เครื่องวัดความดันโลหิตและอัตราการเต้นของหัวใจ ควบคุมด้วยโทรศัพท์มือถือ
ชื่อนักศึกษา	นายสุรเชษฎ์ บุญงอก รหัสประจำตัว 46010870
อาจารย์ที่ปรึกษา	ผศ. บุณย์ชนะ ภู่ระหงษ์ ดร. สมเกียรติ อุดมหรรษากุล
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2549

ปริญญานิพนธ์ฉบับนี้ ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

.....

(ผศ. บุณย์ชนะ ภู่ระหงษ์)

อาจารย์ที่ปรึกษา

.....

(ดร. สมเกียรติ อุดมหรรษากุล)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ เครื่องวัดความดันโลหิตและอัตราการเต้นของหัวใจ ควบคุมด้วยโทรศัพท์มือถือ

ชื่อนักศึกษา นายสุรเชษฎ์ บุญงอก รหัสประจำตัว 46010870

อาจารย์ที่ปรึกษา ผศ. บุณย์ชนะ ภู่ระหงษ์
ดร. สมเกียรติ อุดมหรรษากุล

ระดับการศึกษา ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมสารสนเทศ

ภาควิชา วิศวกรรมสารสนเทศ

ปีการศึกษา 2549

บทคัดย่อ

โครงงานนี้เป็นการประยุกต์ใช้ไมโครคอนโทลเลอร์ ให้ทำงานร่วมกับเครื่องวัดความดันโลหิดแบบดิจิตอล โดยส่งคำสั่งให้ทำงานผ่านโทรศัพท์มือถือและส่งผลที่ได้กลับมายังโทรศัพท์มือถือในรูปแบบ SMS ซึ่ง ที่เครื่องวัดความดันจะมีอุปกรณ์ที่ทำหน้าที่รับสัญญาณจากโทรศัพท์มือถือ โดยอุปกรณ์นี้จะเชื่อมต่อกับ ไมโครคอนโทลเลอร์ ซึ่ง ไมโครคอนโทลเลอร์นี้จะเป็นตัวควบคุมการทำงานของเครื่องวัดความดันโลหิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title Digital blood pressure and pulse rate meter control by mobile phone

Student Mr. Suraches Boonngog ID. 46010870

Advisor Mr. Boonchana Purahong

Dr. Somkait Udomhunsakul

Graduate Level Bachelor Degree of Information Engineering

Department Information Engineering

Academic Year 2006

ABSTRACT

The thesis is apply microcontroller to operate with digital blood pressure meter which assign by mobile phone and send the result as a SMS (short message service). The meter have a device to receive a signal from the mobile phone that contact with microcontroller that control the digital blood pressure meter.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณ อาจารย์ที่ปรึกษา ผศ.บุณย์ชนะ ภู่ระหงษ์ ดร. สมเกียรติ อุดมหรรษากุล และอาจารย์ประจำภาควิชาวิศวกรรมสารสนเทศทุกท่านที่ให้คำแนะนำและให้คำปรึกษา

และขอขอบคุณบริษัทเซอร์ไคล์ฟ จำกัด ที่ให้ความอนุเคราะห์ในด้านอุปกรณ์ต่างๆ และที่จะลืมไม่ได้คือเพื่อนๆ นักศึกษาภาควิชาวิศวกรรมสารสนเทศทั้งหลักสูตร 4 ปี และต่อเนื่องทุกคนที่ให้ความช่วยเหลือในหลายๆด้าน

ผู้จัดทำ

นายสุรเชษฎ์ บุญงอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	1
1.1 วัตถุประสงค์	1
1.2 องค์ประกอบของเครื่องวัดความดันโลหิตและอัตราการเต้นของหัวใจ	1
1.3 ขอบเขตของโครงงาน	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 ความดันโลหิต (blood pressure)	3
2.2 ทฤษฎีไมโครคอนโทรลเลอร์	3
2.2.1 ไมโครคอนโทรลเลอร์ตระกูล MCS-51	4
2.2.2 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51	4
2.2.3 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51	6
2.2.4 การแบ่งประเภทของหน่วยความจำ	9
2.2.5 กระบวนการอินเตอร์รัพต์	10
2.2.6 การจัดการอินเตอร์รัพต์	11
2.3 การสื่อสาร	11
2.3.1 การสื่อสารแบบขนาน	12
2.3.2 การสื่อสารแบบอนุกรม	13
2.3.3 การสื่อสารข้อมูลแบบซิงโครนัส	13
2.3.4 การสื่อสารข้อมูลแบบอะซิงโครนัส	13
2.4 การอินเตอร์เฟส	14
2.4.1 มาตรฐานพอร์ตอนุกรมแบบ RS-232	15
2.4.2 คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ	15
2.4.3 การแฮนด์เชคใน RS-232	16
2.5 GSM AT Command กับมือถือ	19
2.5.1 ทดสอบการเชื่อมต่อกับมือถือ	20
2.5.2 AT COMMAND ที่เกี่ยวข้องกับการรับ-ส่ง SMS	22
2.5.3 หลักการรับ – ส่ง SMS	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

เรื่อง	หน้า
2.5.4 โหมดของการรับ – ส่ง ข้อมูล	26
2.5.5 การรับข้อความ SMS ในพีดียูโหมด	26
2.5.6 การส่งข้อความ SMS ในพีดียูโหมด	28
2.5.7 การแปลงตัวอักษรชนิด 7 บิต เป็นข้อมูล 8 บิต(Octet)	29
2.5.8 รหัสตัวอักษรชนิด 7 บิต (7 bit default alphabet)	30
บทที่ 3 การออกแบบและการสร้าง	34
3.1 การออกแบบ	34
3.2 เครื่องวัดความดันโลหิต	35
3.3 ส่วนควบคุม	36
3.4 การใช้งานพอร์ตอนุกรมมาตราฐาน RS-232	38
3.4.1 RS-232 serial connecter pin assignment	38
3.4.2 รูปแบบของการเชื่อมต่อ	40
บทที่ 4 การทดลองและผลการทดลอง	44
4.1 การเชื่อมต่อโทรศัพท์มือถือ กับคอมพิวเตอร์	44
4.2 การเชื่อมต่อเครื่องวัดความดันโลหิดกับคอมพิวเตอร์	46
4.3 การเชื่อมต่อ Microcontroller โทรศัพท์มือถือ และเครื่องวัดความดันโลหิต	47
บทที่ 5 สรุปผลการทดลอง	50
5.1 สรุปปัญหาจากการทดลอง	50
5.2 สรุปผลการทดลอง	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ชื่อตาราง	หน้า
ตารางที่ 2.1 รายละเอียดบางส่วนของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	8
ตารางที่ 2.2 ตารางเปรียบเทียบการทำงานระหว่างการสื่อสารข้อมูลแบบขนานและแบบอนุกรม	14
ตารางที่ 2.3 การจัดขา DB-9 และ DB-25	16
ตารางที่ 2.4 อธิบายรายละเอียดของข้อความ 22 Byte ที่ส่ง	25
ตารางที่ 2.5 ส่วนประกอบของสตริงการรับข้อความ SMS	27
ตารางที่ 2.6 ส่วนประกอบของสตริงการส่งข้อความ SMS	29
ตารางที่ 2.7 รหัสตัวอักษรชนิด 7 บิต (7 bit default alphabet)	30
ตารางที่ 3.1 รายละเอียดขาสัญญาณของพอร์ตอนุกรม RS-232	39
ตารางที่ 3.2 หน้าที่การทำงานของการเชื่อมต่อแบบเต็มรูปแบบ	41
ตารางที่ 3.3 หน้าที่การทำงานของการเชื่อมต่อแบบ Partial Handshaking	42
ตารางที่ 3.4 หน้าที่การทำงานของการเชื่อมต่อแบบ Loop Back Handshaking	43
ตารางที่ 3.5 หน้าที่การทำงานของการเชื่อมต่อแบบง่าย	43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

ชื่อภาพ	หน้า
รูปที่ 2.1 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แฟลช อนุกรม AT89Sxx	5
รูปที่ 2.2 การจัดขาตามมาตรฐานของไมโครคอนโทรลเลอร์ MCS-51	6
รูปที่ 2.3 รายละเอียดโครงสร้างหลักของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	7
รูปที่ 2.4 การเลือกใช้หน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์เบอร์ AT89C51	10
รูปที่ 2.5 การสื่อสารแบบขนาน	12
รูปที่ 2.6 การสื่อสารข้อมูลแบบอนุกรม	13
รูปที่ 2.7 คอนเน็กเตอร์ของพอร์ตอนุกรม	15
รูปที่ 2.8 การเชื่อมต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์	17
รูปที่ 2.9 ลักษณะภายนอกของไอซี MAX 232	19
รูปที่ 2.10 การเชื่อมต่อมือถือเข้ากับคอมพิวเตอร์ด้วยสาย Data Link	20
รูปที่ 2.11 เลือกรูปแบบของการเชื่อมต่อจากโปรแกรม Hyper Terminal	21
รูปที่ 2.12 ทดสอบการเชื่อมต่อกับมือถือ	21
รูปที่ 2.13 ตัวอย่างคำสั่งพื้นฐานและ LIST ข้อความใน STO SENT	23
รูปที่ 2.14 ตัวอย่างการอ่านข้อความ	24
รูปที่ 3.1 Block Diagram อธิบายการทำงานโดยรวมของระบบ	34
รูปที่ 3.2 โฟล์วชาร์ตแสดงขั้นตอนการทำงานของระบบ	35
รูปที่ 3.3 เครื่องวัดความดันโลหิตแบบดิจิตอล	36
รูปที่ 3.4 วงจรรีเลย์	37
รูปที่ 3.5 การเชื่อมต่อไมโครคอนโทลเลอร์กับโทรศัพท์มือถือผ่านทาง Serial port	37
รูปที่ 3.6 วงจรส่วนควบคุม	38
รูปที่ 3.7 RS-232 คอนเน็คเตอร์แบบ DB9	38
รูปที่ 3.8 RS-232 คอนเน็คเตอร์แบบ DB9 ตัวผู้ที่อยู่ด้านหลังคอมพิวเตอร์	39
รูปที่ 3.9 ลักษณะการเชื่อมต่อแบบเต็มรูปแบบ	40
รูปที่ 3.10 ลักษณะการเชื่อมต่อแบบ Partial Handshaking	41
รูปที่ 3.11 ลักษณะการเชื่อมต่อแบบ Loop Back Handshaking	42
รูปที่ 3.12 ลักษณะการเชื่อมต่อแบบง่าย	43
รูปที่ 4.1 โทรศัพท์มือถือ และสาย Data Link	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ(ต่อ)

ชื่อภาพ	หน้า
รูปที่ 4.2 Dialog Box Port setting จากโปรแกรม Hyper Terminal	45
รูปที่ 4.3 การทดลองส่ง AT command ไปยังโทรศัพท์มือถือ	46
รูปที่ 4.4 แสดงโปรแกรม BP 3.0 ขณะเชื่อมต่อเครื่องวัดความดันกับคอมพิวเตอร์	47
รูปที่ 4.5 จอ LCD ในสถานะพร้อมทำงาน	48
รูปที่ 4.6 ภาพจอ LCD ขณะได้รับ SMS	48
รูปที่ 4.7 จอ LCD ของเครื่องวัดความดันโลหิตขณะกำลังทำงาน	49
รูปที่ 4.8 แสดงภาพหน้าจอ LCD ของเครื่องวัดความดันเมื่อทำงานเสร็จ	49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในยุคการสื่อสารที่ไร้ขีดจำกัด โทรศัพท์มือถือ นับได้ว่าเป็นอุปกรณ์พื้นฐานที่ทุกคนจะต้องมีไว้พกพา ซึ่งปัจจุบันนี้ บริษัทผู้ผลิตโทรศัพท์มือถือ ได้พัฒนาขีดความสามารถและแอพพลิเคชันต่างๆ เพิ่มเข้ามาไว้ในโทรศัพท์มือถือ เพื่อความสะดวกสบายในการดำเนินชีวิตประจำวันของผู้ใช้มากขึ้น

การนำเทคโนโลยีทางการสื่อสารเข้ามาใช้ในทางการแพทย์ จะช่วยให้การตรวจรักษามีความสะดวกและประสิทธิภาพมากยิ่งขึ้น และยังสามารถนำไปประยุกต์ใช้งานได้ในลักษณะต่างๆ ตามประเภทและรูปแบบของการตรวจรักษา

1.1. วัตถุประสงค์

1. เพื่อนำเทคโนโลยีในการสื่อสารเข้ามาใช้ในทางการแพทย์
2. เพื่อศึกษาการทำงานของโทรศัพท์มือถือ ด้วยชุดคำสั่ง AT Command
3. เพื่อประยุกต์ใช้โทรศัพท์มือถือและไมโครคอลโทลเลอร์ให้ทำงานร่วมกัน

1.2. องค์ประกอบของเครื่องวัดความดันโลหิตและอัตราการเต้นของหัวใจผ่านโทรศัพท์มือถือ

ประกอบด้วย 3 ส่วน คือ เครื่องวัดความดันโลหิตแบบดิจิตอล โทรศัพท์มือถือ และไมโครคอลโทลเลอร์

เครื่องวัดความดันโลหิตแบบดิจิตอล เป็นส่วนที่ใช้วัดความดันและชีพจร ซึ่งในเครื่องนี้จะประกอบด้วย ปั๊มลม เซนเซอร์วัดความดัน และจอแสดงผล (LCD)

โทรศัพท์มือถือ เป็นส่วนที่รอรับคำสั่งจากผู้ใช้ และส่งค่าที่วัดได้ผ่านระบบเครือข่ายโทรศัพท์ไปยังผู้ใช้

ไมโครคอลโทลเลอร์ เป็นส่วนที่ใช้ประมวลผลคำสั่งและควบคุมการทำงานของเครื่องวัดความดันและโทรศัพท์มือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3. ขอบเขตของโครงงาน

ส่งข้อความ (SMS) ไปยังโทรศัพท์มือถือที่เชื่อมต่ออยู่กับไมโครคอนโทลเลอร์ เพื่อให้ไมโครคอนโทลเลอร์ไปทำการ สวิตซ์เครื่องวัดความดันโลหิตให้ทำงาน และเมื่อเครื่องวัดความดันโลหิตทำการวัดเสร็จ ก็ให้ส่งผลที่วัดได้ กลับไปยังโทรศัพท์มือถือในรูปแบบ SMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ความดันโลหิต (blood pressure)

ความดันโลหิต (blood pressure) คือ แรงดันที่เกิดจากการบีบตัวและคลายตัวของหัวใจ ห้องล่างซ้าย โดยหัวใจบีบตัวเพื่อนำเลือดไปเลี้ยงส่วนต่างๆ ของร่างกาย เลือดจะถูกส่งผ่านไปยังเส้นเลือดแดงใหญ่และกระจายไปตามหลอดเลือดแดงไปสู่อวัยวะสำคัญต่างๆ รวมถึงแขนและขา ซึ่งโลหิตจะมีแรงกระทำต่อผนังเส้นเลือด เมื่อเวลาวัดความดันที่หลอดเลือดแดงบริเวณแขนหรือขาจะได้ค่าตัวเลข 2 ค่า เช่น 120/80 มม.ปรอท ค่าตัวบนเรียกว่า ความดันช่วงหัวใจบีบ (systolic) หมายถึงความดันเมื่อหัวใจห้องล่างซ้ายบีบตัว จากตัวอย่างวัดได้ค่าเท่ากับ120 มม.ปรอท ส่วนค่าตัวล่างเรียกว่า ความดันช่วงหัวใจคลาย (diastolic) หมายถึง ความดันเมื่อหัวใจคลายตัว ซึ่งจากตัวอย่างจะมีค่าเท่ากับ 80 มม.ปรอท

2.2 ทฤษฎีไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ (Microcontroller) เป็นอุปกรณ์อิเล็กทรอนิกส์แบบหนึ่งที่รวมเอาหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์และลอจิก วงจรรับสัญญาณอินพุท วงจรขับสัญญาณเอาท์พุท หน่วยความจำ วงจรกำเนิดสัญญาณนาฬิกาไว้ด้วยกัน ทำให้สามารถนำไปใช้งานแทนวงจรอิเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี ช่วยลดจำนวนอุปกรณ์และขนาดของระบบในขณะที่มีความสามารถสูงขึ้น ภายใต้งบประมาณที่เหมาะสม

ไมโครคอนโทรลเลอร์มาจากคำ 2 คำรวมกันคือ "ไมโคร" (micro) ซึ่งหมายถึง ไมโครโปรเซสเซอร์ (microprocessor) ซึ่งเป็นอุปกรณ์ประมวลผลข้อมูลขนาดเล็ก ภายในประกอบด้วยหน่วยประมวลผลกลางหรือซีพียู (CPU : Central Processing Unit) หน่วยคำนวณทางคณิตศาสตร์และลอจิก (ALU : Arithmetic Logic Unit) วงจรเชื่อมต่อหน่วยความจำ และวงจรสัญญาณนาฬิกา อีกคำหนึ่งคือคำว่า "คอนโทรลเลอร์" (controller) หมายถึง อุปกรณ์ควบคุม ดังนั้นไมโครคอนโทรลเลอร์จึงเป็นอุปกรณ์ที่ใช้ในการควบคุม โดยสามารถเขียนโปรแกรมเพื่อกำหนดรูปแบบการควบคุมได้อย่างอิสระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ไมโครคอนโทรลเลอร์ตระกูล MCS-51

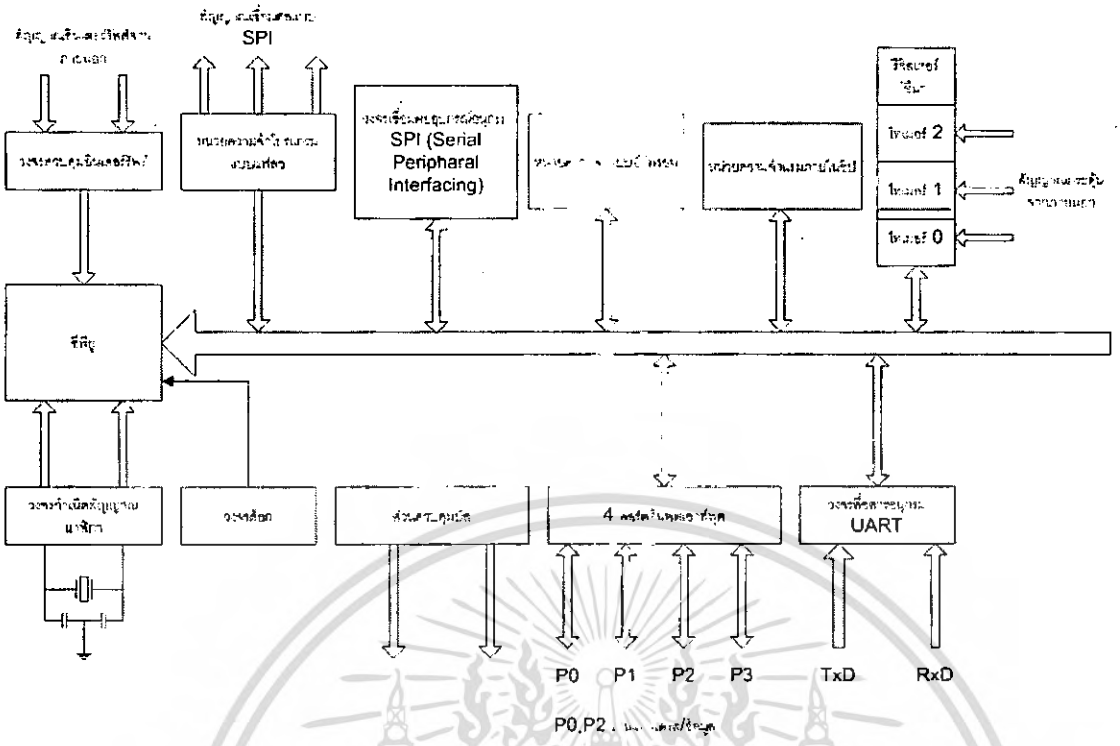
ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ที่จะใช้ในปริญญานิพนธ์นี้จะเป็นไมโครคอนโทรลเลอร์ ตระกูล MCS-51 ที่มีหน่วยความจำภายในเป็นแบบแฟลช (Flash Memory) ที่มีเบอร์ขึ้นต้นด้วย AT89 เหตุผลที่ใช้ไมโครคอนโทรลเลอร์แบบนี้ในการเรียนรู้เพื่อการใช้งานเพราะไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีด้วยกันหลายประการดังนี้

- หน่วยความจำโปรแกรมภายในตัวไมโครคอนโทรลเลอร์เป็นแบบแฟลช สามารถลบและเขียนใหม่ได้นับ 1,000 ครั้ง จึงสามารถใช้งานในรูปแบบของมโครคอนโทรลเลอร์ ชิปเดี่ยวไม่ต้องใช้หน่วยความจำภายนอก ส่งผลให้สามารถใช้งานพอร์ตอินพุต (Port Input) ของไมโครคอนโทรลเลอร์ได้อย่างเต็มประสิทธิภาพ
- ไม่จำเป็นต้องใช้เครื่องมือพัฒนา จำพวกอีมูเลเตอร์ (Emulator) และเครื่องโปรแกรมอีพรอม (EPROM)
- ไมโครคอนโทรลเลอร์ตระกูลนี้มีออกมาหลายเบอร์ และมีความสามารถแตกต่างกันออกไป ทำให้มีทางเลือกในการใช้งานที่ดีกว่า
- มีการใช้หน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ ทำให้สามารถป้องกันการคัดลอกข้อมูลของหน่วยความจำโปรแกรมได้เป็นอย่างดี

2.2.2 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51

- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลชสามารถลบและเขียนใหม่ได้ 1,000 ครั้ง
- หน่วยความจำข้อมูลพื้นฐานเป็นหน่วยความจำแบบแรม (Ram) ในบางเบอร์จะมีหน่วยความจำแบบอีพรอมเพิ่มเติม
- ขาพอร์ตเป็นแบบสองทิศทาง สามารถใช้งานได้ทั้งเป็นอินพุตและเอาต์พุต
- มีวงจรสื่อสารอนุกรมแบบ 2 ทิศทาง (Full Duplex)
- ไทเมอร์/เคาน์เตอร์ (Timer/Counter) ขนาด 16 บิตอย่างน้อย 2 ตัว
- สามารถรองรับแหล่งกำเนิดอินเตอร์รัพ (Interrupt) ได้ 6 ประเภท
- สามารถขยายหน่วยความจำภายนอกเพิ่มเติมได้สูงสุด 64 กิโลไบต์
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ภายในชิพ
- มีวงจรสื่อสารอนุกรมแบบ SPI สำหรับในอนุกรม AT89Sxx
- มีวอตช์ด็อกไทเมอร์ในตัว สำหรับในอนุกรม AT89Sxx

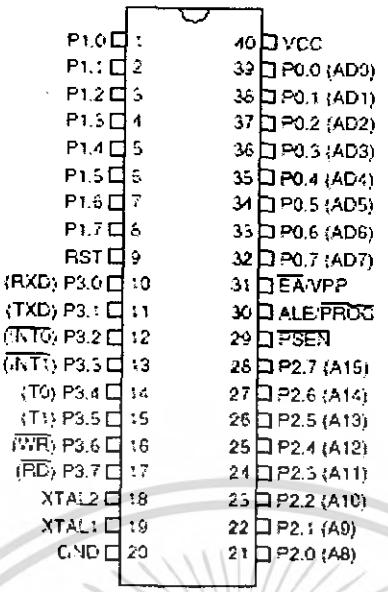
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แฟลช อนุกรม AT89Sxx

จากรูปที่ 2.1 แสดงโครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แบบอนุกรม AT89Sxx มีส่วนประกอบที่เพิ่มเติมแตกต่างจาก AT89C5x ขึ้นมาคือ วงจรเชื่อมต่อแบบ SPI ซึ่งในไมโครคอนโทรลเลอร์อนุกรมนี้ใช้ในการเขียนข้อมูลลงในหน่วยความจำโปรแกรมโดยไม่ต้องถอดตัวชิปออกไปจากระบบ หรือเรียกว่าการโปรแกรมในวงจร ไทเมอร์และเคาน์เตอร์ขนาด 16 บิต ที่เพิ่มเติมเข้าหาอีกหนึ่งตัวเป็นไทเมอร์ 2 และมีวงจรวอตช์ด็อกที่ใช้ในการตรวจสอบการทำงานผิด พลาดของซีพียู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 การจัดขาตามมาตรฐานของไมโครคอนโทรลเลอร์ MCS-51

2.2.3 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51

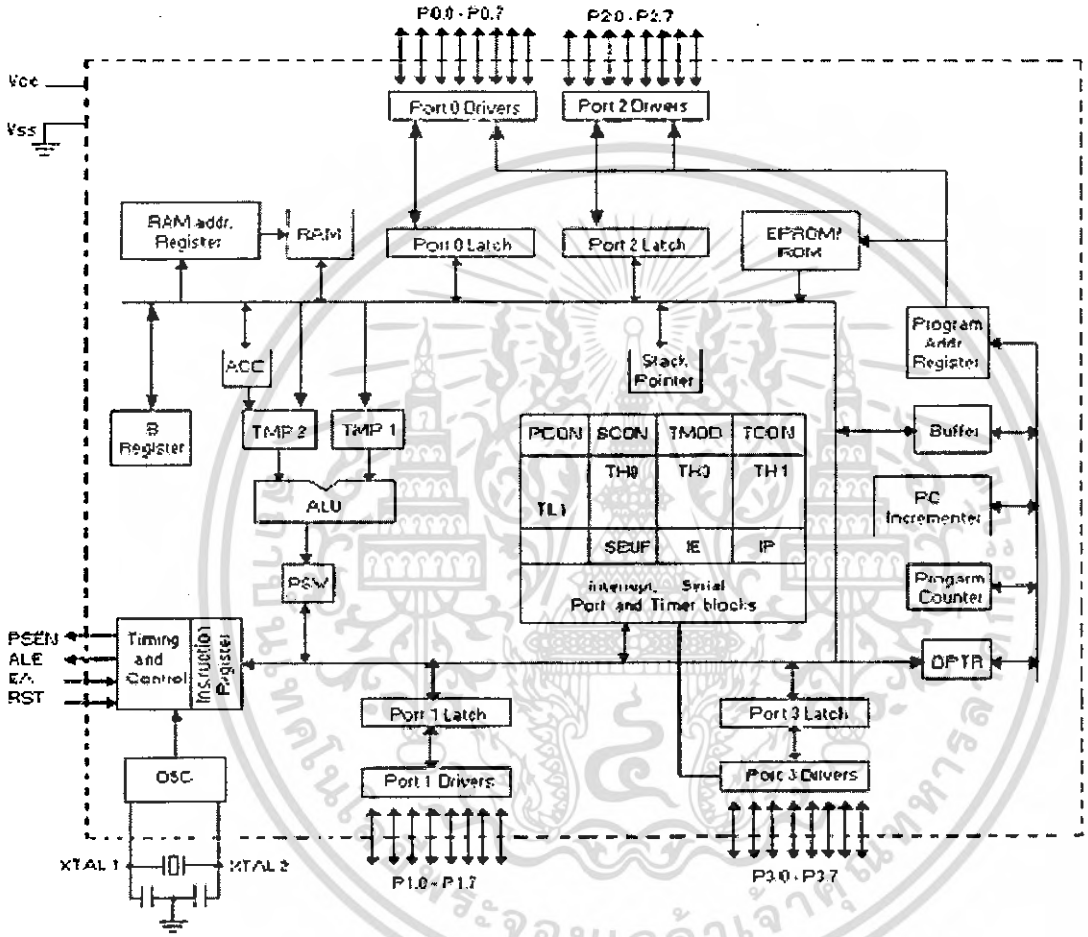
ไมโครคอนโทรลเลอร์ MCS-51 ทุกเบอร์จะมีขาใช้งานพื้นฐานเหมือนกัน ดังรูปที่ 2.2 และโครงสร้างหลักของ MCS-51 ในรูปที่ 2.3 โดยมีรายละเอียดขั้นต้นดังนี้

- ขา VCC ที่ขา 40 ใช้สำหรับต่อไฟเลี้ยง +5V
- ขา GND ที่ขา 20 เป็นขากราวด์ สำหรับต่อกับกราวด์ของระบบ
- ขาพอร์ต 0 (P0.0-P0.7) มี 8 ขา ตั้งแต่ขา 32 – 39 แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุต และเอาต์พุตสำหรับใช้งานทั่วไป ถ้าหากต้องกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุตสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย จะส่งผลให้พอร์ตนั้นมีสถานะปล่อยลอย (Float) ที่มีอินพุตอิมพีแดนซ์สูง สามารถใช้งานเป็นพอร์ตอินพุตได้ นอกจากนั้นขาพอร์ตนี้ยังใช้งานในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำภายนอก (A0-A7) และขาของข้อมูล (A0-A7) และขาของข้อมูล (D0-D7) โดยใช้กระบวนการมัลติเพลกซ์เข้าช่วย เพื่อสลับการทำงานเป็นได้ทั้งขาติดต่อกับแอดเดรส และขาข้อมูล
- ขาพอร์ต 1 (P1.0-P1.7) มี 8 ขา ตั้งแต่ขา 1 – 8 โดยมีการกำหนดสถานะของขาพอร์ตเหมือนกับการกำหนดขาพอร์ต 0 นอกจากนั้นในอนุกรม AT89Sxx จะใช้ขา P1.0 เป็นขาอินพุตสำหรับนับค่าของไทเมอร์ 2 และ P1.1 เป็นขาอินพุตทริกเกอร์ของไทเมอร์ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขณะที่ขา P1.4 ถึง P1.7 เป็นขาสำหรับเชื่อมต่อแบบ SPI เพื่อทำการโปรแกรมข้อมูลในระบบ

- ขาพอร์ต 2 (P2.0-P2.7) มี 8 ขา ตั้งแต่ขา 21 – 28 โดยมีการกำหนดสถานะของขาพอร์ตเหมือนกับการกำหนดขาพอร์ต 0 นอกจากนั้นขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความจำภายนอก (A8-A15)



รูปที่ 2.3 รายละเอียดโครงสร้างหลักของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

- ขาพอร์ต 3 (P3.0-P3.7) มี 8 ขา ตั้งแต่ขา 10 - 17 โดยมีการกำหนดสถานะของขาพอร์ตเหมือนกับการกำหนดขาพอร์ต 0 นอกจากนั้นขาพอร์ต 3 ยังเป็นขาที่มีหน้าที่การใช้งานพิเศษ ดังมีรายละเอียดดังต่อไปนี้
  - P3.0 ใช้เป็นขาอินพุตสำหรับรับข้อมูลจากการสื่อสารอนุกรม หรือขา RxD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- P3.1 ใช้เป็นขาอินพุตสำหรับส่งข้อมูลจากการสื่อสารอนุกรม หรือขา TxD
- P3.2 ใช้เป็นขาอินพุตรับสัญญาณอินเตอร์รัพต์จากภายนอกช่อง 0 หรือ ขา INT0
- P3.3 ใช้เป็นขาอินพุตรับสัญญาณอินเตอร์รัพต์จากภายนอกช่อง 1 หรือ ขา INT0
- P3.4 ใช้เป็นขาอินพุตสำหรับรับสัญญาณไทเมอร์จากภายนอกช่อง 0 หรือขา T0
- P3.5 ใช้เป็นขาอินพุตสำหรับรับสัญญาณอินเตอร์รัพต์จากภายนอกช่อง 1 หรือขา T1
- P3.6 ใช้เป็นขาสัญญาณ WR ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก
- P3.7 ใช้เป็นขาสัญญาณ RD ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก

เบอร์ของไมโครคอนโทรลเลอร์	หน่วยความจำโปรแกรม	หน่วยความจำข้อมูล	จำนวนไทเมอร์/เคาน์เตอร์ 16 บิต
AT89SC1051	แบบแฟลช ขนาด 1 กิโลไบต์	แรม 64 ไบต์	1
AT89SC2051	แบบแฟลช ขนาด 2 กิโลไบต์	แรม 128 ไบต์	2
AT89SC51	แบบแฟลช ขนาด 4 กิโลไบต์	แรม 128 ไบต์	2
AT89SC52	แบบแฟลช ขนาด 8 กิโลไบต์	แรม 256 ไบต์	3
AT89SC55	แบบแฟลช ขนาด 20 กิโลไบต์	แรม 256 ไบต์	3
AT89S52	แบบแฟลช ขนาด 8 กิโลไบต์	แรม 256 ไบต์ อีอีพรอม 2 กิโลไบต์	3
AT89S53	แบบแฟลช ขนาด 12 กิโลไบต์	แรม 256 ไบต์	3

ตารางที่ 2.1 รายละเอียดบางส่วนของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

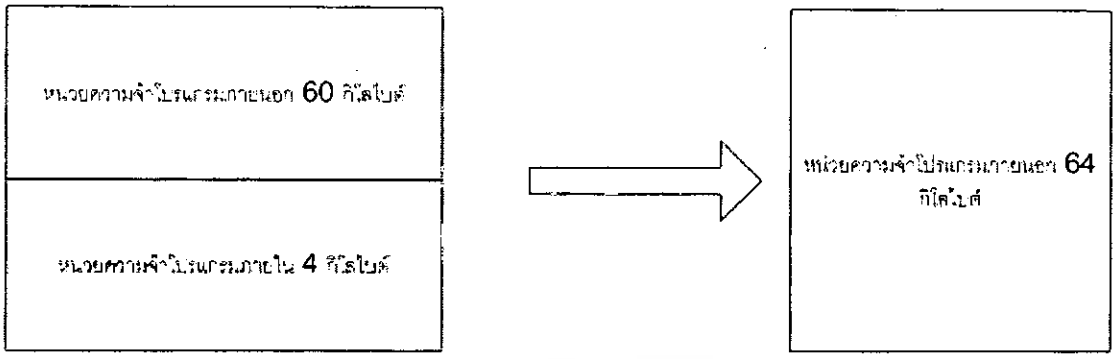
- ขารีเซต (Reset) ใช้ในการรีเซตการทำงานของไมโครคอนโทรลเลอร์ โดยนำการป้อนสัญญาณเพื่อรีเซตสถานะที่ขานี้ต้องอยู่ในระดับรีเซตอย่างน้อย 2 แมชชีนไซเคิล โดยที่วงจรกำหนดสัญญาณนาฬิกายังคงทำงานต่อเนื่องไปอย่างปกติ
- ขา ALE/PROG (Address Latch Enable/Program pulse input) เป็นขาที่ใช้ในการควบคุมการแลตช์ของขาพอร์ต 0 เมื่อมีการใช้งานหน่วยความจำภายนอก นอกจากนั้นขานี้ยังใช้เป็นขาสำหรับรับพัลส์ของการโปรแกรมสำหรับโปรแกรมข้อมูลลงในไมโครคอนโทรลเลอร์ MCS-51
- ขา PSEN (Program Store Enable) ขานี้ใช้ในการส่งสัญญาณเพื่อร้องขอติดต่อกับหน่วยความจำโปรแกรมภายนอก เมื่อไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก ตัวไมโครคอนโทรลเลอร์จะส่งสัญญาณออกมาที่ขานี้ 2 ครั้ง ในแต่ละ แมชชีนไซเกิล แต่ถ้าหากติดต่อกับหน่วยความจำข้อมูลภายนอก ขานี้จะไม่มีการส่งสัญญาณใดๆ ออกมา
- ขา EA/Vpp (External Access Enable/Programming Voltage input) ใช้สำหรับเลือกหน่วยความจำโปรแกรมจากภายนอกหรือภายในไมโครคอนโทรลเลอร์ ถ้าขานี้เป็น "0" จะเป็นการเลือกให้ไมโครคอนโทรลเลอร์ ติดต่อกับหน่วยความจำภายนอก แต่ถ้าหากขานี้เป็น "1" ก็จะเป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ นอกจากนี้ที่ขานี้ยังใช้เป็นขาอินพุตสำหรับรับแรงดันไฟฟ้าสูงสำหรับการโปรแกรมหน่วยความจำภายในของไมโครคอนโทรลเลอร์
- XTAL1 และ XTAL2 เป็นขาสำหรับติดต่อคริสตอลเพื่อสร้างสัญญาณนาฬิกาในการกำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์

2.2.4 การแบ่งประเภทของหน่วยความจำ

ในไมโครคอนโทรลเลอร์ MCS-51 มีหน่วยความจำภายในหลักๆ อยู่ 2 ส่วน คือ หน่วยความจำโปรแกรม และหน่วยความจำข้อมูล

- หน่วยความจำโปรแกรม (Program Memory) เป็นโปรแกรมที่ใช้เก็บโปรแกรมสั่งงาน โดยสามารถเลือกใช้หน่วยความจำภายในอย่างเดียวหรือรวมกับภายนอก หรือเลือกใช้หน่วยความจำภายนอกอย่างเดียวก็ได้ โดยภายใน AT89C51 จะมีหน่วยความจำโปรแกรมภายใน 4 กิโลไบต์ ถ้าเลือกใช้หน่วยความจำภายนอกทั้งหมด 64 กิโลไบต์ ดังแสดงในรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การเลือกใช้หน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์เบอร์ AT89C51

- หน่วยความจำข้อมูล (Data Memory) มีด้วยกัน 2 แบบ คือ หน่วยความจำข้อมูลภายนอกและภายใน โดยไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89 สามารถติดต่อกับหน่วยความจำภายนอกได้สูงสุด 64 กิโลไบต์ การติดต่อกับหน่วยความจำข้อมูลภายนอกของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช มีลักษณะคล้ายกับการติดต่อกับหน่วยความจำโปรแกรมภายนอก แตกต่างกันที่มีสัญญาณการอ่าน และเขียนข้อมูลภายนอก

สำหรับไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89 ทุกเบอร์จะมีหน่วยความจำข้อมูลภายในเป็นแบบแรม โดยแต่ละเบอร์จะมีขนาดแตกต่างกันไป สำหรับการจัดสรรหน่วยความจำข้อมูลภายในแบ่งเป็น 3 ส่วน คือ หน่วยความจำข้อมูลส่วนบน ส่วนล่าง และรีจิสเตอร์ฟังก์ชันพิเศษ

2.2.5 กระบวนการอินเตอร์รัพต์

การติดต่อกันระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอกสามารถทำได้ 2 ลักษณะคือ ใช้วิธีการโพลลิ่ง (Polling) โดยไมโครคอนโทรลเลอร์ต้องคอยตรวจสอบอุปกรณ์ภายนอกอยู่ตลอดเวลาว่ามีข้อมูลที่ต้องการติดต่อกับไมโครคอนโทรลเลอร์หรือไม่ ซึ่งวิธีนี้จะทำให้ไมโคร คอนโทรลเลอร์ต้องเสียเวลาไปกับการตรวจสอบนานมาก หากมีอุปกรณ์จำนวนมากติดต่ออยู่กับไมโครคอนโทรลเลอร์ การติดต่อแบบที่สองเป็นการใช้การอินเตอร์รัพต์ เป็นวิธีการที่ไมโคร คอนโทรลเลอร์ไม่ต้องคอยตรวจสอบอุปกรณ์ภายนอกอยู่ตลอด แต่จะให้อุปกรณ์ภายนอกส่งสัญญาณอินเตอร์รัพต์เข้ามาให้ไมโครคอนโทรลเลอร์ เมื่ออุปกรณ์ภายนอกต้องการติดต่อกับ ไมโครคอนโทรลเลอร์ เมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์ได้รับสัญญาณอินเตอร์รัพต์แล้ว จึงทำการรับส่งข้อมูลกับอุปกรณ์ภายนอกนั้นๆ ต่อไป

การอินเตอร์รัพต์ (Interrupt) เป็นชื่อเรียกกระบวนการที่เข้ามาขัดจังหวะการทำงานโดยปกติของไมโครคอนโทรลเลอร์ ในไมโครคอนโทรลเลอร์ MCS-51 สามารถตอบสนองการอินเตอร์รัพต์ที่เกิดขึ้นได้จาก 5 แหล่งกำหนดสำหรับเบอร์ AT89C51 ประกอบด้วยการอินเตอร์รัพต์จากภายนอกผ่านเข้าทางขา INT0 และ INT1 สัญญาณอินเตอร์รัพต์จากไทเมอร์/เคาน์เตอร์ T0 และ T1 ในขณะที่ไมโครคอนโทรลเลอร์ AT89C52 และในอนุกรม AT89Sxx สามารถตอบสนองการอินเตอร์รัพต์ได้จาก 6 แหล่งกำเนิด โดยเพิ่มการรับสัญญาณอินเตอร์รัพต์จากไทเมอร์/เคาน์เตอร์2 อีกหนึ่งแหล่งกำเนิด

2.2.6 การจัดการอินเตอร์รัพต์

เมื่อมีการอินเตอร์รัพต์ในไมโครคอนโทรลเลอร์ MCS-51 เกิดขึ้น และมีการ Enable การการตอบสนองการอินเตอร์รัพต์ไว้ กระบวนการหลังจากนั้นซีพียูจะกระโดดไปยังแอดเดรสในหน่วยความจำที่กำหนดไว้ เรียกตำแหน่งแอดเดรสนี้ว่า แอดเดรสอินเตอร์รัพต์ (Interrupt vector address) ดังนั้นจะต้องมีการเขียนโปรแกรมย่อยการบริการอินเตอร์รัพต์ไว้ที่แอดเดรสอินเตอร์รัพต์เวกเตอร์นี้โดยค่าของแอดเดรสอินเตอร์รัพต์เวกเตอร์จะแตกต่างกันไปในการอินเตอร์รัพต์ต่างๆ ดังรายละเอียดต่อไปนี้

- การอินเตอร์รัพต์ภายนอกที่ขา INT0 มีค่าแอดเดรสอินเตอร์รัพต์เวกเตอร์อยู่ที่ 0003H
- การอินเตอร์รัพต์จากไทเมอร์ 0 มีค่าแอดเดรสอินเตอร์รัพต์เวกเตอร์อยู่ที่ 0003H
- การอินเดอร์รัพต์ภายนอกที่ขา INT1 มีค่าแอดเดรสอินเตอร์รัพต์เวกเตอร์อยู่ที่ 0003H
- การอินเตอร์รัพต์จากไทเมอร์ 1 มีค่าแอดเดรสอินเตอร์รัพต์เวกเตอร์อยู่ที่ 001BH
- การอินเตอร์รัพต์จากพอร์ตอนุกรม มีค่าแอดเดรสอินเตอร์รัพต์เวกเตอร์อยู่ที่ 0023H
- การอินเตอร์รัพต์จากไทเมอร์ 2 มีค่าแอดเดรสอินเตอร์รัพต์เวกเตอร์อยู่ที่ 002BH

2.3 การสื่อสาร

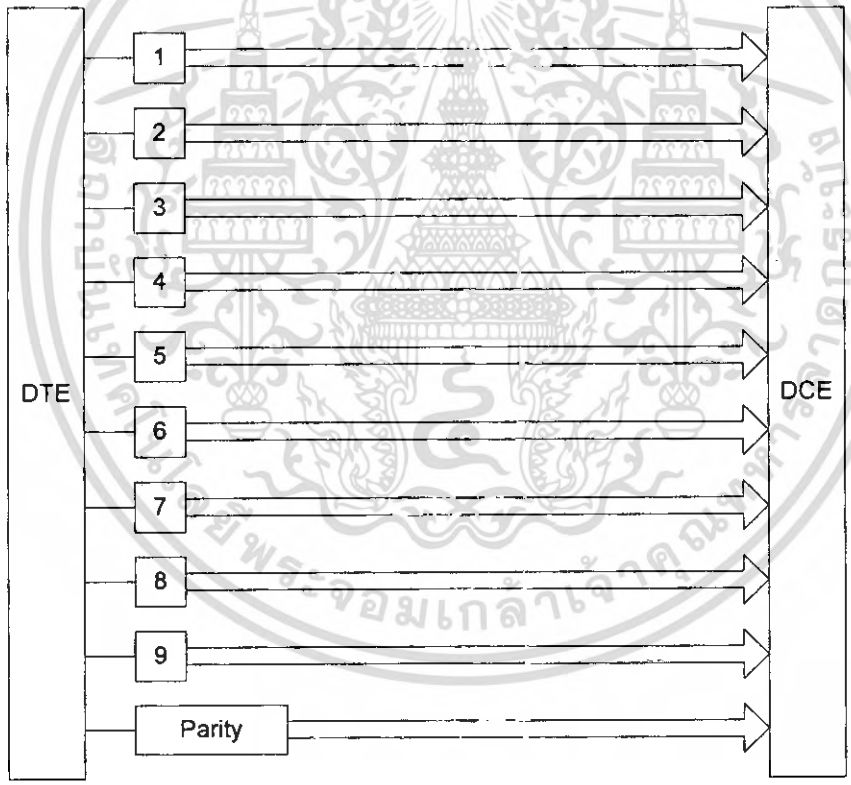
การสื่อสารข้อมูลมี 2 แบบคือ การสื่อสารแบบขนานและการสื่อสารแบบอนุกรม ข้อมูลในการสื่อสารแต่ละข้อมูลจะถูกแปลงให้อยู่ในรูปแบบของเลขฐานสอง แล้วนำมาประกอบกัน เช่น ถ้าข้อมูลที่ประกอบด้วย 4 บิต เราเรียกว่า 1 นิบเบิล หรือถ้าข้อมูลที่ประกอบด้วย 8 บิตเราจะเรียกว่า 1 ไบต์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 การสื่อสารแบบขนาน

การสื่อสารแบบขนานจะมีรูปแบบการส่งข้อมูลครั้งละ 1 ไบต์ก็คือจะทำการส่งข้อมูลครั้งละ 8 บิตนั่นเอง ซึ่งในการส่งต้องใช้สายไฟในการส่งข้อมูล 8 เส้น แล้วยังต้องใช้สายไฟอีก 1 เส้นในการควบคุม

ปัญหาที่สำคัญของการส่งข้อมูลแบบขนานคือ คุณสมบัติของบิตกับแรงดัน เวลาที่บิตหรือแรงดันไฟฟ้าที่ใช้ในการเปลี่ยนสถานะจากหนึ่งเป็นศูนย์นั้นสั้นมาก โดยเร็วถึงระดับนาโนวินาที การเปลี่ยนแปลงที่รวดเร็วนี้เป็นส่วนที่สำคัญมากต่อการส่งข้อมูล เพราะการเปลี่ยนแปลงระหว่างศูนย์และหนึ่งอย่างช้าๆ จะไม่ถูกอ่านเป็นข้อมูลเลย และเมื่อสายไฟที่ใช้ส่งข้อมูลยาวขึ้น คุณสมบัติทางไฟฟ้าของสายไฟเช่น ค่าความจุไฟฟ้าและค่าความเหนี่ยวนำจะจำกัดความเร็วในการเปลี่ยนแปลงระหว่างศูนย์และหนึ่งของบิต ซึ่งจะทำให้ข้อมูลอาจสูญหายหรือทำให้การส่งข้อมูลล้มเหลวได้ ดังนั้นการส่งข้อมูลบนสายยาวอาจจะเป็นปัญหาได้หากใช้วิธีการสื่อสารแบบขนาน



รูปที่ 2.5 การสื่อสารแบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 การสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรมนั้นแบ่งได้เป็น 2 แบบด้วยกันคือ การสื่อสารอนุกรมแบบซิงโครนัส และการสื่อสารอนุกรมแบบอะซิงโครนัส

2.3.3 การสื่อสารข้อมูลแบบซิงโครนัส

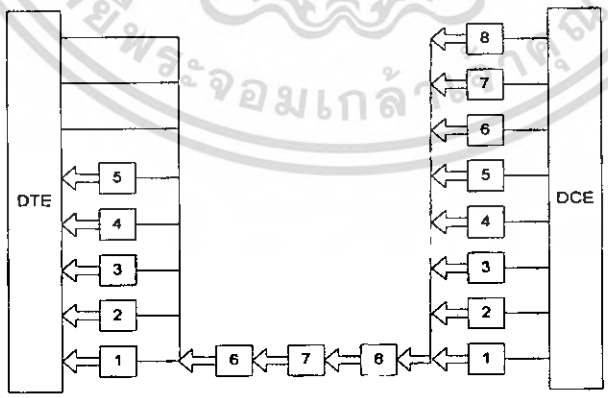
การสื่อสารแบบซิงโคนัสจะมีสัญญาณนาฬิการวมอยู่กับการรับและส่งสัญญาณด้วย ตัวอย่างการส่งข้อมูลแบบซิงโครนัสก็คือคีย์บอร์ดของคอมพิวเตอร์ ซึ่งสายหนึ่งจะเป็นสายสัญญาณนาฬิกา ส่วนอีกสายหนึ่งจะเป็นสายของข้อมูล ดังนั้นการติดต่อแบบซิงโครนัสนี้ จะต้องใช้สายในการเชื่อมต่ออย่างน้อยที่สุด 3 เส้นคือ สัญญาณนาฬิกา ข้อมูล และกราวด์

2.3.4 การสื่อสารข้อมูลแบบอะซิงโครนัส

การสื่อสารข้อมูลแบบอะซิงโครนัสคือ การรับและส่งข้อมูลไปในสายโดยไม่จำเป็น ต้องมีสัญญาณนาฬิการ่วมด้วย แต่จะมีการกำหนดค่าสัญญาณนาฬิกาทั้งภาครับและภาคส่งให้มีค่าเท่ากัน ซึ่งเรียกสัญญาณนาฬิกาที่ใช้ในการกำหนดค่าให้ภาครับและภาคส่งนี้ว่า อัตราการถ่ายทอดข้อมูล หรือ บอดเรต (Broad rate) มีหน่วยเป็นบิตต่อวินาที (Bit Per Second: bps)

รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบอะซิงโครนัสประกอบด้วย 4 ส่วนคือ

- บิตเริ่มต้น (Start Bit) ซึ่งจะมีขนาด 1 บิต
- บิดข้อมูลแบบอนุกรมจะมีขนาด 5,6,7 หรือ 9 บิต
- บิตตรวจสอบพาริตี้ (Parity Bit) จะมีขนาด 1 บิตหรือไม่มี
- บิตปิดท้าย (Stop Bit) จะมีขนาด 1 , 1.5 หรือ 2 บิต



รูปที่ 2.6 การสื่อสารข้อมูลแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน	แบบขนาน	แบบอนุกรม
1. ระยะทาง	จะใช้งานได้ในระยะไม่เกิน 100 ฟุต	จะสามารถใช้งานได้ตั้งแต่ในระยะใกล้ๆ ไปจนถึงระยะทางที่มากๆ จนถึงหลักไมล์
2. ความเร็ว	อัตราความเร็วสูงมากในระยะที่ไม่ไกลมากนักกำหนดได้เป็นจำนวนบิตต่อวินาที	อัตราความเร็วของข้อมูลที่ใช้กันอยู่ทั่วไปจะอยู่ในช่วง 0-2 ล้านบิตต่อวินาที
3. ระดับของสัญญาณ	ในการอินเตอร์เฟสจะใช้ระดับของสัญญาณที่ใช้กับอุปกรณ์ TTL คือสัญญาณลอจิก 1 และ 0 จะแทนด้วยระดับแรงดัน +5V และ 0V	ในมาตรฐานของ EIA-RS 232C ระบุว่ามีระดับสัญญาณไฟฟ้าขนาด 12V หรือใช้มาตรฐาน 20 mA current loop
4. ความผิดพลาดของสัญญาณ	ง่าย	การใช้งานจะเกิดการผิดพลาดของสัญญาณจะมีน้อยมาก
5. ค่าใช้จ่าย	ค่าใช้จ่ายจะสูงมากเพราะจะต้องใช้สายส่งสัญญาณหลายเส้น โดยเฉพาะการส่งในระยะทางไกลๆ	สิ้นเปลืองน้อยกว่ามาก ถึงแม้ว่าจะต้องใช้อุปกรณ์เปลี่ยนสัญญาณของข้อมูลจากขนานไปเป็นอนุกรมแล้วส่งผ่านสายส่งแล้วกลับสัญญาณมาเป็นขนานอีกครั้งก็ตาม

ตารางที่ 2.2 ตารางเปรียบเทียบการทำงานระหว่างการสื่อสารข้อมูลแบบขนานและแบบอนุกรม

2.4 การอินเตอร์เฟส

วัตถุประสงค์หลักของการอินเตอร์เฟสก็คือ การใช้อุปกรณ์อินเตอร์เฟสเป็นสื่อกลางของการส่งข้อมูล และง่ายต่อการใช้งานเมื่อเราสามารถทำการอินเตอร์เฟสได้สำเร็จ ก็สามารถที่จะส่งข้อมูลสู่ภายนอกได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

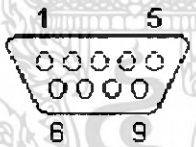
2.4.1 มาตรฐานพอร์ตอนุกรมแบบ RS-232

มาตรฐาน RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment:DTE) กับวงจรข้อมูลปลายทาง (Data Circuit Terminating:DCE) ไว้ว่าอุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวไมโครคอนโทรลเลอร์หรือไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างบิตข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้งสองจะกระทำผ่านมาตรฐาน RS-232

ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งที่เห็นได้ชัดคือ คอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ที่โมเด็มจะเป็นแบบ DCE

2.4.2 คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ

คอนเน็กเตอร์อนุกรม 25 ขาหรือแบบ DB-25



คอนเน็กเตอร์อนุกรมแบบ 9 ขาหรือแบบ DB-9

รูปที่ 2.7 คอนเน็กเตอร์ของพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอนเน็กเตอร์ DB-9	คอนเน็กเตอร์ DB-25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
1	8	Data Carrier Detect:DCD	อินพุต
2	3	Received Data:RxD	อินพุต
3	2	Transmitted Data:TxD	เอาต์พุต
4	20	Data Terminal Teady:DTR	เอาต์พุต
5	7	Signal Ground:GND	-
คอนเน็กเตอร์ DB-9	คอนเน็กเตอร์ DB-25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
6	6	Data Set Ready:DSR	อินพุต
7	4	Request To Send:RTS	เอาต์พุต
8	5	Clear To Send:CTS	อินพุต
9	22	Ring Indicator:RI	อินพุต

ตารางที่ 2.3 การจัดขา DB-9 และ DB-25

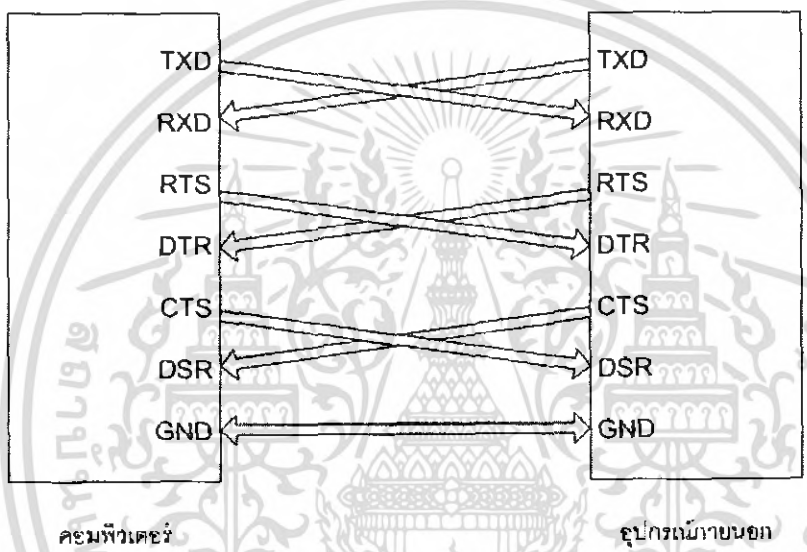
สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอก แสดงดังรูปที่ 2.8 ลูกศรในรูปแสดงถึงทิศทางของข้อมูล ดังรูปที่ 2.8 (ก) เป็นการเชื่อมต่อแบบ Null Modem หรือการเชื่อมต่อโดยตรงโดยไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบหรือแฮนด์เช็กเต็มรูปแบบ ส่วนในรูปที่ 2.8 (ข) เป็นการเชื่อมต่อแบบ Null Modem ในลักษณะที่ใช้สายสัญญาณ 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นสำหรับรับข้อมูล และเส้นสุดท้ายเป็นกราวน์ สำหรับรายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232 มีดังนี้

2.4.3 การแฮนด์เชคใน RS-232

การแฮนด์เชคหมายถึง กระบวนการที่อุปกรณ์หนึ่งใช้ตรวจสอบสถานะของอีกอุปกรณ์ที่ต่อเข้าด้วยกัน และตอบสนองสถานะนั้นอย่างเหมาะสมและถูกจังหวะเวลา ซึ่งก็คือวิธีการควบคุมการทำงานของอุปกรณ์สองตัวให้สัมพันธ์กันในการรับส่งข้อมูลนั่นเอง แบ่งออกเป็น 2 ประเภทคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

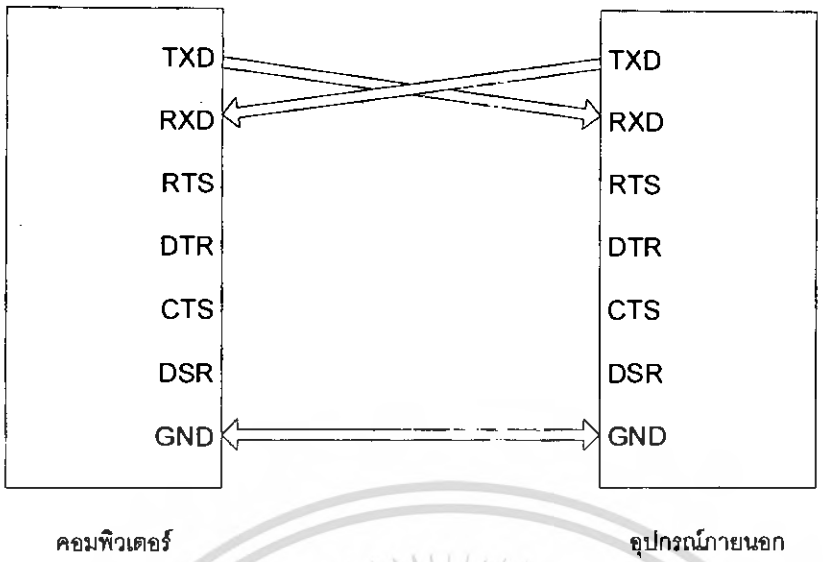
- การแฮนด์เชคทางซอฟต์แวร์ (Software Handshaking) เป็นวิธีการหนึ่งที่ควบคุมการทำงานของอุปกรณ์รับข้อมูลโดยส่งผ่านสัญญาณควบคุมไปพร้อมกับตัวข้อมูลที่ต้องการส่ง
- การแฮนด์เชคทางฮาร์ดแวร์ (Hardware Handshaking) สามารถควบคุมได้ทั้งระดับฮาร์ดแวร์โดยการเปลี่ยนระดับแรงดันในสายสัญญาณควบคุมเป็นตัวระงับไม่ให้คอมพิวเตอร์ส่งข้อมูลเพิ่มเข้ามาอีก ซึ่งเป็นการหลีกเลี่ยงการใช้รหัสหรือโปรแกรม แต่การแฮนด์เชคทางฮาร์ดแวร์นั้นมีข้อจำกัด คือจำเป็นต้องมีสายสัญญาณควบคุมต่างหากโดยเฉพาะ ทำให้วิธีนี้ไม่เหมาะที่จะนำมาใช้ในการอินเตอร์เฟสกันโมเด็ม



การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ Null Mode

72703

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



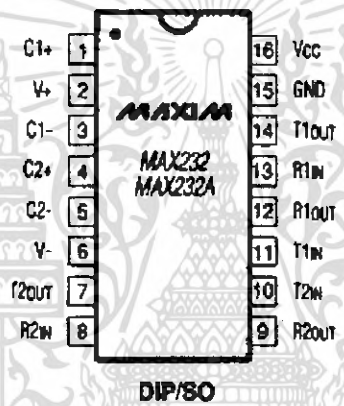
การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ RS-232 โดยใช้สัญญาณ 3 เส้น

รูปที่ 2.8 การเชื่อมต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์

- Data Carrier Detect:DCD หรืออาจเรียกว่า Carrier Detect:CD ขานี้จะแอกตีฟ เมื่อมีการส่งสัญญาณพาหะจากอุปกรณ์สื่อสารข้อมูลเช่น โมเด็ม สำหรับใช้งานปกติขานี้จะไม่ถูกใช้งานมากนัก
- Receive Data:RD หรือ RxD ขานี้ใช้เพื่อรับส่งสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์ โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์ บัฟเฟอร์
- Transmitted Data:TD หรือ TxD ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลออกไป
- Data Terminal Ready:DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่าต้องการติดต่อด้วย โดยขา DTR ของอุปกรณ์ปลายทาง และขา DTR ของอุปกรณ์ปลายทางจะต้องเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกันและต้องต่อขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาหะ
- Signal Ground:Gnd ขากราวด์ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Data Set Ready:DSR ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DTR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขา DTR
- Request To Send:RTS เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่การเชื่อมต่อแบบ Null Modem 3 สาย จะต้องเชื่อมต่อขา RTS และ CTS ของตัวมันเองเข้าด้วยกัน เพื่อจะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา
- Clear To Send:CTS ขานี้จะคอยรับสัญญาณเรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็มและโปรแกรมมีการตรวจสอบสัญญาณนี้เท่านั้น



รูปที่ 2.9 ลักษณะภายนอกของไอซี MAX 232

2.5 GSM AT Command กับมือถือ

AT-COMMAND คือ ชุดคำสั่งมาตรฐาน ที่สามารถใช้ติดต่อสื่อสารกับอุปกรณ์สื่อสารต่างๆ เช่น โมเด็ม หรือ อุปกรณ์ DTE (Data Terminal Equipment) เพื่อโต้ตอบตั้งค่าหรือสั่งงานอุปกรณ์เหล่านั้นให้ทำงานตามที่ต้องการ และสำหรับการติดต่อกับโทรศัพท์มือถือจะใช้ชุดคำสั่งที่เรียกว่า " GSM AT Command "

การสื่อสารกับอุปกรณ์สื่อสารต่างๆ เช่น โมเด็มหรืออุปกรณ์ DTE (Data Terminal Equipment) นั้นสามารถใช้ชุดคำสั่งที่เป็นมาตรฐานที่เรียกว่า AT Command ในการติดต่อเพื่อติดต่อตั้งค่าหรือสั่งงานอุปกรณ์เหล่านั้น ให้ทำงานตามที่ต้องการโดยชุดคำสั่งพื้นฐานจะถูกกำหนดไว้ใน Hayes AT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command ซึ่งบริษัท Hayes เป็นผู้คิดค้นชุดคำสั่งนี้เพื่อใช้กับโมเด็มของตนและต่อมาได้กลายเป็นมาตรฐานสำหรับผู้ผลิตโมเด็มรายอื่นๆ โดยอาจจะมีชุดคำสั่งขยาย (Extended AT Command) เพื่อใช้เป็นการเฉพาะสำหรับผู้ผลิตรายนั้นๆ ได้

การติดต่อมือถือก็เช่นกันเราสามารถใช้ชุดคำสั่งที่กำหนดไว้ใน GSM AT Command ซึ่งมีคำสั่งเพิ่มเติมที่เหมาะสมสำหรับการใช้งานและควบคุมมือถือและเนื่องจากมีรายละเอียดค่อนข้างมาก โดยในโครงงานนี้จะพูดถึงเฉพาะคำสั่งที่จำเป็นสำหรับบทความนี้เท่านั้น

การเชื่อมต่อคอมพิวเตอร์กับมือถือนั้นจะทำผ่านสาย Data Link ซึ่งเป็นการเชื่อมต่อแบบอนุกรมโดยใช้โปรแกรม Terminal ต่างๆ เช่น Hyper Terminal ของ Windows ส่วนความเร็วในการสื่อสารนั้นขึ้นอยู่กับอุปกรณ์ที่นำมาเชื่อมต่อแต่ส่วนใหญ่แล้วมักจะใช้ 9600 bps

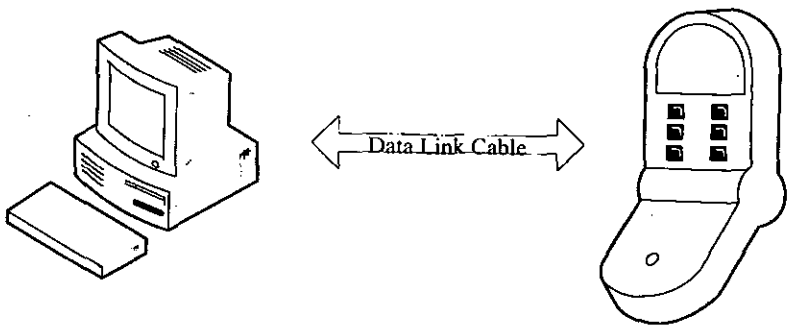
2.5.1 ทดสอบการเชื่อมต่อกับมือถือ

ก่อนอื่นเราต้องทดสอบการเชื่อมต่อกับมือถือด้วยสาย Data Link เพื่อให้มั่นใจว่า มือถือสามารถทำงานได้ถูกต้องและรองรับการส่ง SMS โดยทำการต่อตามรูปที่ 2.16 โดยนำสาย Data Link ที่ใช้ได้กับมือถือมาต่อปลายข้างหนึ่งที่มักจะเป็นหัวต่อแบบ DB 9 ต่อเข้ากับคอมพิวเตอร์ที่ Port Com1 หรือ Com2 ก็ได้ถ้ามีแล้วต่ออีกปลายหนึ่งของสายซึ่งจะเป็นหัวต่อที่เหมาะสมกับมือถือแต่ละรุ่นเสียบเข้ากับตัวมือถือ

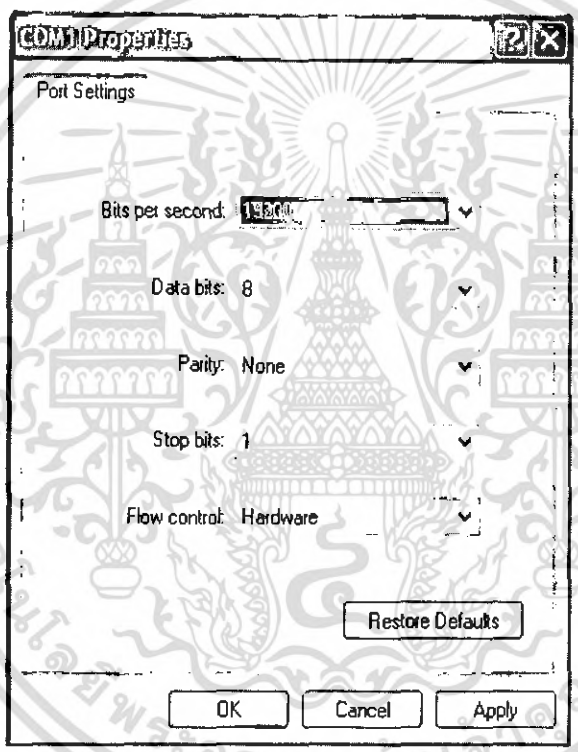
เมื่อต่อสายเรียบร้อยดีแล้วและตรวจให้แน่ใจว่าทั้งคอมพิวเตอร์และมือถือได้เปิดเครื่องไว้แล้ว ทางฝั่งคอมพิวเตอร์ให้เรียกใช้โปรแกรม Hyper Terminal บน Windows โดยเลือกการเชื่อมต่อเป็น Direct to COM1 หรือ COM2 ซึ่งก็แล้วแต่ว่าต่อสายเอาไว้ทาง Port ใด หลังจากนั้นให้เลือกความเร็วเป็น 19200 bps และเลือก Flow control เป็น Hardware ดังรูปที่ 2.17

เมื่อเข้าสู่โปรแกรมแล้วทดลองพิมพ์ at แล้วกด Enter ถ้าการเชื่อมต่อถูกต้องจะตอบ OK กลับมา หลังจากนั้น ทดลองพิมพ์คำสั่ง at+csms = 0 แล้ว กด Enter ถ้าหากตอบข้อมูลและ OK กลับมา ดังรูปที่ 2.18 แสดงว่ามือถือเครื่องนี้พร้อมจะใช้งานสำหรับโครงงานนี้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

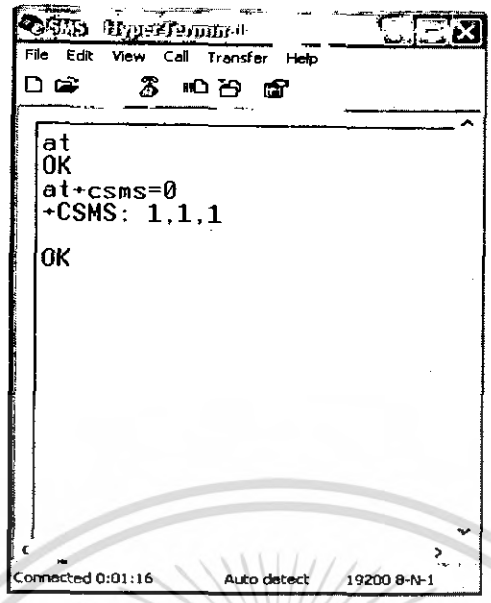


รูปที่ 2.10 การเชื่อมต่อมือถือเข้ากับคอมพิวเตอร์ด้วยสาย Data Link



รูปที่ 2.11 เลือกรูปแบบของการเชื่อมต่อจากโปรแกรม Hyper Terminal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 ทดสอบการเชื่อมต่อกับมือถือ

ตัวอย่างคำสั่ง GSM AT Command

- คำสั่งพื้นฐาน
- at // เช็คความพร้อมของมือถือ
- OK // พร้อม
- Atd018289492 // สั่งให้ต่อโทรศัพท์ไปยังหมายเลขนี้ (บางเครื่องอาจต้องใช้ ; ต่อท้าย)
- BUSY // สายไม่ว่าง (ถ้าอีกฝั่งรับสายจะตอบ CONNECT)
- ath // สั่งวางสาย
- OK // ตกลง

2.5.2 AT COMMAND ที่เกี่ยวข้องกับการรับ-ส่ง SMS

ชุดคำสั่ง AT COMMAND ที่ใช้กับโทรศัพท์มือถือได้มีอยู่มากมาย ทั้งการอ่านรุ่นโทรศัพท์มือถือ ตรวจสอบระดับแบตเตอรี่ ตรวจสอบระดับสัญญาณ แต่ในที่นี้จะกล่าวถึงเพียงคำสั่งที่เกี่ยวข้องกับการรับ-ส่ง SMS เท่านั้น

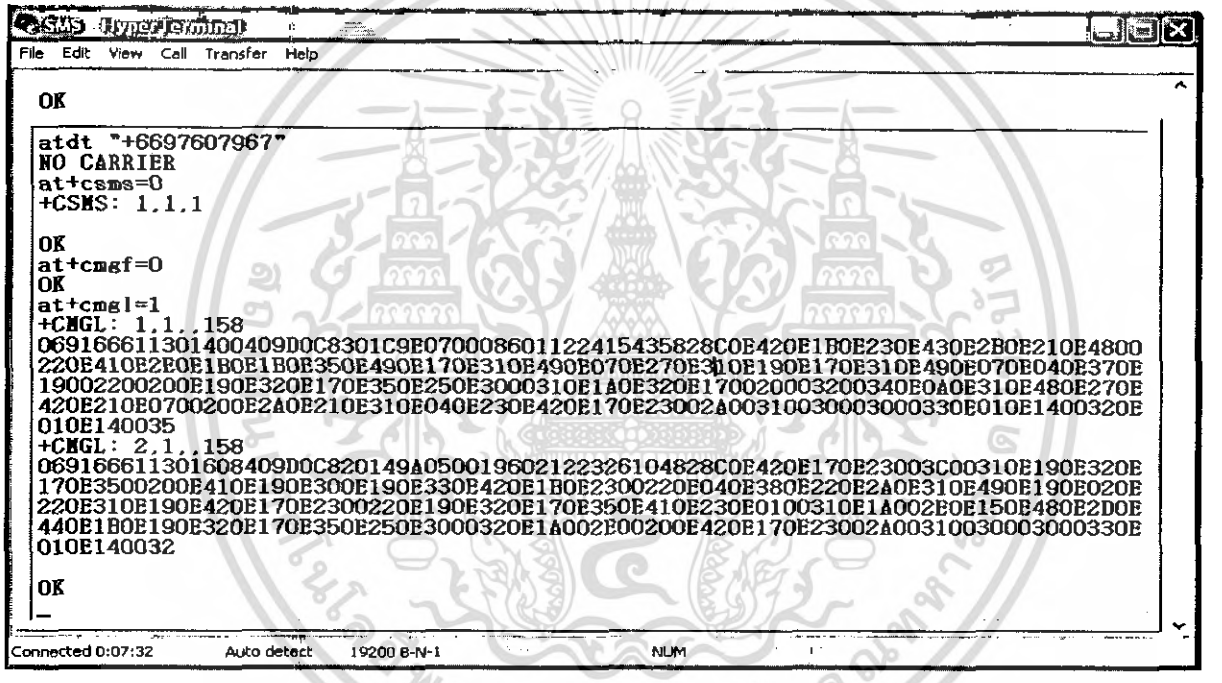
1) Message Format (AT+CMGF) เป็นคำสั่งกำหนดรูปแบบของข้อความที่จะให้แสดงออกมาโดย
   - AT+CSMS = 0 // เช็คว่ามือถือสนับสนุนการส่ง SMS หรือไม่
   - AT+CMGF = 1 คือ แสดงข้อความในรูปแบบ TEXT
   - AT+CMGF = 0 คือ แสดงข้อความในรูปแบบ PDU CODE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) List Message (AT+CMGL) เป็นคำสั่งที่ให้แสดงข้อความในสถานะต่างๆ โดยจะแสดงข้อความทั้งหมดนั้นสถานะนั้น ดังนี้

- AT+CMGL=0 คือ แสดงข้อความที่ได้รับแต่ยังไม่ได้อ่าน ("REC UNREAD")
- AT+CMGL=1 คือ แสดงข้อความที่ได้รับและอ่านแล้ว ("REC READ")
- AT+CMGL=2 คือ แสดงข้อความที่เก็บไว้และยังไม่ได้ส่ง ("STO UNSENT")
- AT+CMGL=3 คือ แสดงข้อความที่เก็บไว้และส่งออกไปแล้ว ("STO SENT")
- AT+CMGL=4 คือ แสดงข้อความทั้งหมด ("ALL")

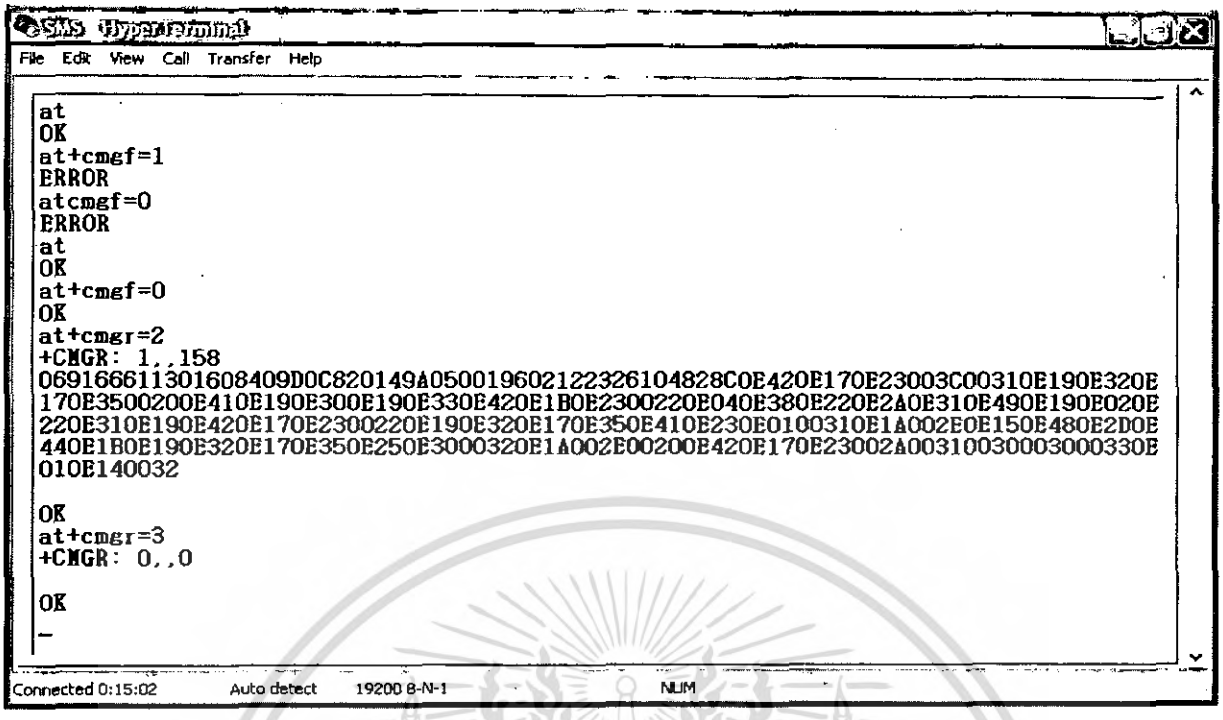
หมายเหตุ หากกำหนด Message Format เป็น PDU CODE จะต้องเลือกสถานะโดยใช้ตัวเลข 0 ถึง 4 แต่หากกำหนด Message Format เป็น Text จะต้องเลือกสถานะโดยใช้ตัวอักษรที่วงเล็บด้านหลัง



รูปที่ 2.13 ตัวอย่างคำสั่งพื้นฐานและ LIST ข้อความใน STO SENT

3) Read Message (AT+CMGR) เป็นคำสั่งที่ใช้อ่านข้อความที่เฉพาะเจาะจงได้โดยระบุตำแหน่งที่ข้อความนั้นถูกเก็บไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 ตัวอย่างการอ่านข้อความ

4) Set the Message center number (AT+CSCA="XXX") เป็นคำสั่งที่ใช้ระบุหมายเลขของ SMSC เพื่อจะทำการส่ง SMS ต่อไปได้

5) Send Message (AT+CMGS="YYY") เป็นคำสั่งที่ใช้ระบุเลขหมายปลายทางที่ต้องการส่ง SMS ไปถึง เช่น

AT+CMGS = 22 // ต้องการส่งทั้งหมด 22 byte (ไม่รวมตัวเลข 00 ที่อยู่ข้างหน้าสุด)

>0011000A916629502680000AA0AE8329BFD4697D9EC37 // เมื่อพิมพ์ข้อความครบแล้ว กด Ctrl+z ส่วนประกอบของข้อมูลที่ส่งอธิบายในตารางที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่มตัวเลข 8 บิต (Octets)	รายละเอียด
00	ความยาวของ SMSC information "00" หมายถึงให้ใช้ SMSC Information ที่เก็บอยู่ภายในเครื่อง (ปกติเครื่องที่สามารถส่ง SMS ได้ มีข้อมูล SMSC ภายในเครื่องอยู่แล้ว)
11	First octet of the SMS – SUBMIT message.
00	TP – Message – Reference "00" คือ ให้เครื่องตั้งหมายเลขอ้างอิงข้อความขึ้นเอง
0A	Address – Length. ความยาวของหมายเลขผู้รับ (10 ตัว)
91	Type – of – Address (91 indicates international format of the phone number).
66 29 50 26 80	หมายเลขโทรศัพท์ของผู้รับ (แบบ semi octets)
00	TP – PID. (Protocol identifier) เป็น 00
00	TP – DCS. (Data coding scheme) เป็น 00
AA	TP – Validity – Period "AA" หมายถึง ช่วงเวลาหมดอายุของข้อความ 4 วัน ถ้าภายในช่วงเวลานี้ ยังส่งไม่ถึงปลายทาง ข้อความจะถูกยกเลิกโดยอัตโนมัติ
0A	TP – User – Data – Length. จำนวนตัวอักษรของข้อความที่ส่ง (10ตัว)
E8329BFD4697D9EC37	TP – User – Data. เป็นข้อมูลในส่วนของข้อความตัวอักษรแบบ 7 บิต "hellohello" ที่ผ่านการแปลง (เข้ารหัส) เป็นข้อมูลแบบ 8 บิต แล้วโดยมีวิธีการตามหัวข้อการแปลงตัวอักษรชนิด 7 บิต เป็น ข้อมูล 8 บิต

ตารางที่ 2.4 อธิบายรายละเอียดของข้อความ 22 Byte ที่ส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 หลักการรับ – ส่ง SMS

องค์การ ETSI (European Telecommunications Standards Institute) เป็นองค์กรอิสระที่ไม่แสวงหาผลกำไรทำหน้าที่กำหนดมาตรฐานด้านโทรคมนาคมได้กำหนดมาตรฐานการส่ง SMS ไว้ในคู่มือ GSM 03.40 และ GSM 03.38 สามารถส่งได้สูงถึง 160 ตัวอักษร โดยแต่ละตัวอักษรใช้รหัสขนาด 7 บิต ที่กำหนดไว้ในตารางที่ 3 นอกจากนั้นยังมีการใช้ตัวอักษรชนิดอื่น ๆ เช่น ขนาด 8 บิต หรือ 16 บิต ซึ่งมีวัตถุประสงค์เพื่อการใช้งานที่แตกต่างกันออกไป ซึ่งในโครงงานนี้จะพูดถึงเฉพาะแบบ 7 บิต เท่านั้น

2.5.4 โหมดของการรับ – ส่ง ข้อมูล

การรับ – ส่ง ข้อมูล SMS มีอยู่ด้วยกัน 2 โหมด คือ เท็กซ์โหมด (Text Mode) และพีดียูโหมด (PDU : Protocol Description Unit Mode) การส่งข้อความในเท็กซ์โหมดนั้นจะเป็นการรับข้อความที่ต้องการส่งมาเข้ารหัสก่อน แล้วค่อยส่งข้อมูลในพีพียูโหมดอีกที อย่างไรก็ตามในมือถือบางรุ่นอาจไม่สนับสนุนการใช้งานในเท็กซ์โหมด ซึ่งการเข้ารหัส (ส่ง) และถอดรหัส (รับ) สำหรับเท็กซ์โหมดนี้มีหลายแบบด้วยกันเช่น “CCP437” “CDN” “8859 – 1” “IRA” และ “GSM” เมื่อเชื่อมต่อกับมือถือเพื่ออ่านข้อความ สามารถตั้งค่าการเข้ารหัส / ถอดรหัสได้โดยใช้คำสั่ง AT+CSCS แต่ถ้าต้องการอ่านข้อความจากจอของมือถือซึ่งตัวมือถือจะเลือกการถอดรหัสที่เหมาะสมให้เองโดยอัตโนมัติ

การเชื่อมต่อกับมือถือเพื่อรับ – ส่ง ข้อความสามารถเลือกใช้ได้ทั้ง 2 โหมดแต่จะเห็นได้ว่าการเลือกใช้เท็กซ์โหมดจะมีข้อจำกัดทั้งจากการที่มือถือบางรุ่นอาจไม่สนับสนุนและยังถูกจำกัดด้วยวิธีการเข้าและถอดรหัส ซึ่งมีเพียงไม่กี่แบบตามที่กล่าวมาข้างต้น ซึ่งบางกรณีอาจไม่สะดวกนักแต่ถ้าเลือกพีดียูโหมดจะสามารถเลือกหรือสร้างการเข้ารหัสและถอดรหัสได้ทุกรูปแบบตามต้องการโดยไม่มีข้อจำกัด ซึ่งในบทความนี้จะพูดถึงเฉพาะพีดียูโหมด

2.5.5 การรับข้อความ SMS ในพีดียูโหมด

หากทำการเชื่อมต่อกับมือถือแล้วทำการสั่งอ่านข้อความ SMS ที่อยู่ใน Inbox โดยใช้คำสั่ง AT+CMGR ข้อมูลที่ได้รับจะอยู่ในรูปของสตริงที่ประกอบไปด้วยข้อมูลของผู้ส่ง ข้อมูล SMS Service Center (SMSC) Time Stamp และอื่นๆ ที่จำเป็นและตามด้วยส่วนของข้อความซึ่งจะอยู่ท้ายสุดของสตริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างสตริงต่อไปนี้รับได้จาก Siemens รุ่น C45 ซึ่งข้อความที่ส่งมาคือ "hellohello" จากมือถืออีกเครื่องหนึ่งข้อมูลสตริงนี้จะอยู่ในรูปของตัวเลขฐาน 16 และฐาน 10 (ในบางส่วน) โดยจะเรียกตัวเลขแต่ละคู่ว่า Octet ซึ่งมีรายละเอียดดังตางรางที่ 2.5

0691668111808804 0A91662950268000

0040302121943482 0AE8329BFD4697

D9EC37

กลุ่มตัวเลข 8 บิต (Octet)	รายละเอียด
06	ความยาวของ SMSC Information 6 Octets (bytes)
91	รูปแบบของเลขหมาย SMSC 91 หมายถึง เลขหมายแบบสากล (international format)
66 81 11 80 88	เลขหมาย SMSC (แบบ decimal semi – octets) ซึ่งจะเป็นเลขฐาน 10 สลับ nibble ในกรณีนี้เลขหมายจริงของ Service Center คือ " +6618110888 "
04	First octets of this SMS – DELIVER message
0A	ความยาวของเลขหมายผู้ส่ง (0A hex = 10 ตัว)
91	รูปแบบของเลขหมายผู้ส่ง 91 หมายถึง เลขหมายแบบสากล (international format)
66 29 50 26 80	เลขหมายผู้ส่ง (แบบ decimal semi – octets) เป็นเลขฐาน 10 สลับ nibble หมายเลขผู้ส่งที่แท้จริงคือ " +6692056208 "
00	TP – PID. (Protocol identifier) ในกรณีนี้คือ 00
00	TP – DCS (Data coding scheme) 00 คือ เข้ารหัสข้อความแบบ 7 bits Default Alphabet
40 30 21 21 94 34 82	TP – SCTS. ข้อมูล Time – Stamp (แบบ decimal semi – octets) สลับ nibble
0A	TP – UDL, User data length จำนวนตัวอักษรของข้อความที่ส่งในที่นี้คือ 10 ตัว
E8329BFD4697D9EC37	TP – UD. ข้อความ " hellohello " ที่เข้ารหัสแล้วจากตัวอักษรแบบ 7 bits เป็นข้อมูล byte ขนาด 8 bits

ตารางที่ 2.5 ส่วนประกอบของสตริงการรับข้อความ SMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลทั้งหมดในตารางเป็นเลขฐาน 16 ขนาด 8 บิต ยกเว้นหมายเลข Service Center เลขหมายผู้ส่ง Time Stamp จะเป็นเลขฐาน 10 ขนาด 8 บิต สลับหลักเป็นคู่ๆ (สลับ nibble) ในส่วนของข้อมูลที่เป็นข้อความนั้นเป็นเลขฐาน 16 ขนาด 8 บิต เช่นกัน โดยข้อมูลนี้จะใช้แสดงข้อความที่ประกอบไปด้วยตัวอักษรขนาด 7 บิต ซึ่งผ่านการแปลง (เข้ารหัส) ข้อมูลจากตัวอักษรขนาด 7 บิต ให้เป็นเลขฐาน 16 ขนาด 8 บิต มาแล้ว ส่วนวิธีการแปลงจะกล่าวในภายหลัง

ในส่วนของข้อมูลที่เป็นเลขฐาน 10 เช่น หมายเลขผู้ส่งตัวเลขในแต่ละคู่ (1 byte) จะถูกสลับหลักกันเช่น เลขหมายจริง "+66 092056208" จะถูกสลับในแต่ละคู่เป็น "66 29 50 26 80"
(66 คือรหัสประเทศ ส่วนเลขหมวดของหมายเลขมือถือจะถูกตัดเลข 0 ออก เช่น 09 จะเหลือแต่ +669 เป็นต้น แล้วจึงนำตัวเลขทั้งหมดมาต่อกันแล้วสลับคู่) เช่นเดียวกันกับ Time Stamp ซึ่งมีรูปแบบเป็น "YY/MM/DD HH:MM:SS:ss" หมายถึงข้อความนี้ส่งเมื่อ "04/03/12 12:49:43:28"

2.5.6 การส่งข้อความ SMS ในพีดียูโหมด

ต่อไปนี้เป็นตัวอย่างการส่งข้อความ SMS "hellohello" โดยใช้มือถือ Siemens C45 ในโหมดพีดียูไปให้ผู้รับหมายเลข "+66 092056208"

AT+CMGF = 0 // เลือกโหมดพีดียู

AT+CSMS = 0 // เช็คว่ามือถือสนับสนุนการส่ง SMS หรือไม่

AT+CMGS = 22 // ต้องการส่งทั้งหมด 22 byte (ไม่รวมตัวเลข 00 ที่อยู่ข้างหน้าสุด)

>0011000A9166295026800000AA0AE8329BFD4697D9EC37 // เมื่อพิมพ์ข้อความครบแล้วกด Ctrl+z ส่วนประกอบของข้อมูลที่ส่งอธิบายในตารางที่ 2.6

กลุ่มตัวเลข 8 บิต (Octets)	รายละเอียด
00	ความยาวของ SMSC information "00" หมายถึงให้ใช้ SMSC Information ที่เก็บอยู่ภายในเครื่อง (ปกติเครื่องที่สามารถส่ง SMS ได้ มีข้อมูล SMSC ภายในเครื่องอยู่แล้ว)
11	First octet of the SMS – SUBMIT message.
00	TP – Message – Reference "00" คือ ให้เครื่องตั้งหมายเลขอ้างอิงข้อความขึ้นเอง
0A	Address – Length. ความยาวของหมายเลขผู้รับ (10 ตัว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

91	Type – of – Address (91 indicates international format of the phone number)
66 29 50 26 80	หมายเลขโทรศัพท์ของผู้รับ (แบบ semi octets)
00	TP – PID. (Protocol identifier) เป็น 00
00	TP – DCS. (Data coding scheme) เป็น 00
AA	TP – Validity – Period "AA" หมายถึง ช่วงเวลาหมดอายุของข้อความ 4 วัน ถ้าภายในช่วงเวลานี้ ยังส่งไม่ถึงปลายทาง ข้อความจะถูกยกเลิกโดยอัตโนมัติ
0A	TP – User – Data – Length. จำนวนตัวอักษรของข้อความที่ส่ง (10ตัว)
E8329BFD4697D9EC37	TP – User – Data. เป็นข้อมูลในส่วนของข้อความตัวอักษรแบบ 7 บิต "heliohello" ที่ผ่านการแปลง (เข้ารหัส) เป็นข้อมูลแบบ 8 บิต แล้วโดยมีวิธีการตามหัวข้อการแปลงตัวอักษรชนิด 7 บิต เป็น ข้อมูล 8 บิต

ตารางที่ 2.6 ส่วนประกอบของสตริงการส่งข้อความ SMS

2.5.7 การแปลงตัวอักษรชนิด 7 บิต เป็นข้อมูล 8 บิต (Octet)

จากตารางที่ 2.6 ในส่วนของ TP – User – Data จะเป็นส่วนที่สามารถใส่รหัสของข้อความที่ต้องการส่ง แต่เนื่องจากไม่สามารถนำรหัสของตัวอักษรแบบ 7 บิต ใส่ไปได้โดยตรงจึงจำเป็นต้องผ่านการแปลงให้เป็นรหัสข้อมูลแบบ 8 บิตก่อน โดยตัวอย่างต่อไปนี้เป็นการแปลงข้อความ "hellohello" ยาว 10 ตัวอักษร ซึ่งแต่ละตัวอักษรชนิด 7 บิต ให้เป็นข้อมูล 8 บิต สำหรับใช้ในการส่ง SMS

การแปลงเริ่มจากนำรหัส 7 บิตของตัวอักษรตัวแรก (h) มาเติมข้างหน้าด้วย 1 บิต ท้ายสุดของรหัส 7 บิต ของอักษรตัวที่ 2 (e) จะได้ผลลัพธ์ 8 บิต (1 byte) เป็น "E8"

ขั้นตอนต่อมาให้เอา 6 บิต ที่เหลือของตัวอักษรตัวที่ 2 มาเติมข้างหน้าด้วย 2 บิตท้ายของรหัส 7 บิต ของอักษรตัวที่ 3 (l) จะได้ผลลัพธ์ 8 บิต เป็น " 32 " และทำเช่นนี้เรื่อยไปโดยจำนวนบิตที่นำมากระทำจะเพิ่มขึ้นเป็น 3 บิต เป็น 4 บิตจนกระทั่งถึง 7 บิต แล้วเริ่มกระบวนการใหม่จนกระทั่งหมดชุดตัวอักษรดังตัวอย่างการแปลงตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากแปลงข้อความ "hellohello" จะได้ข้อมูลเป็นข้อมูลฐาน 16 จำนวน 9 ไบต์ E8 32 9B FD 46 97 D9 EC 37

2.5.8 รหัสตัวอักษรชนิด 7 บิต (7 bit default alphabet)

ตัวอักษรชนิด 7 บิต ถูกกำหนดโดยคู่มือ GSM 03.38 เป็นดังตารางที่ 2.7 ซึ่งเปรียบเทียบกับรหัสฐาน 10 ของ ISO – 8859 – 1 (Ascii) ซึ่งอยู่ทางด้านขวามือของตาราง

Hex	Dec	Character name	Character	ISO – 8859 – 1 DEC
0x00	0	COMMERCAIL AT	@	64
0x01	1	POUND SIGN		163
0x02	2	DOLLAR SIGN		36
0x03	3	YEN SIGN		165
0x04	4	LATIN SMALL LETTER E WITH GRAVE		232
0x05	5	LATIN SMALL LETTER E WITH ACUTE		233
0x06	6	LATIN SMALL LETTER U WITH GRAVE		249
0x07	7	LATIN SMALL LETTER I WITH GRAVE		236
0x08	8	LATIN SMALL LETTER O WITH GRAVE		242
0x09	9	LATIN CAPITAL LETTER C WITH CEDILLA		199
0x0A	10	LINE FEED		10
0x0B	11	LATIN CAPITAL LETTER O WITH STROKE		216
0x0C	12	LATIN SAMLL LETTER O WITH STROKE		248
0x0D	I3	CARRIAGE RETURN		13
0x0E	14	LATIN CAPITAL LETTER A WITH RING ABOVE		197
0x0F	15	LATIN SAMLL LETTER A WITH RING ABOVE		229
0x10	16	GREEK CAPITAL LETTER DELTA		
0x11	17	LOW LINE		95
0x12	18	GREEK CAPITAL LETTER PHI		
0x13	19	GREEK CAPITAL LETTER GAMMA		
0x14	20	GREEK CAPITAL LETTER LAMBDA		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0x15	21	GREEK CAPITAL LETTER OMEGA		
0x16	22	GREEK CAPITAL LETTER PI		
0x17	23	GREEK CAPITAL LETTER PSI		
0x18	24	GREEK CAPITAL LETTER SIGMA		
0x19	25	GREEK CAPITAL LETTER THETA		
0x1A	26	GREEK CAPITAL LETTER XI		
0x1B	27	ESCAPE TO EXTENSION TABLE		
0x1B0A	27 10	FORM FEED		12
0x1B14	27 20	CIRCUMFLEX ACCENT		94
0x1B28	27 40	LEFT CURLY BRACKET		123
0x1B29	27 41	RIGHT CURLY BRACKET		125
0x1B2F	27 47	REVERSE SOLIDUS (BACKSLASH)		92
0x1B3C	27 60	LEFT SQUARE BRACKET		91
0x1B3D	27 61	TILDE		126
0x1B3E	27 62	RIGHT SQUARE BRACKET		93
0x1B40	27 64	VERTICAL BAR		124
0x1B65	27 101	EURO SIGN		164 (ISO – 8859 -15)
0x1C	28	LATIN CAPITAL LETTER AE		198
0x1D	29	LATIN SMALL LETTER AE		230
0x1E	30	LATIN SMALL LETTER SHARP S (German)		233
0x1F	31	LATIN CAPITAL LETTER E WITH ACUTE		201
0x20	32	SPACE		32
0x21	33	EXCLAMATION MARK		33
0x22	34	QUOTATION MARK		34
0x23	35	NUMBER SIGN		35
0x24	36	CURRENCY SIGN		164 (ISO – 8859 – 1)
0x25	37	PERCENT SIGN		37
0x26	38	AMPERSAND		38
0x27	39	APOSTROPHE		39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0x28	40	LEFT PARENTHESIS		40
0x60	96	INVERTED QUESTION MARK		191
0x61	97	LATIN SMALL LETTER A		97
0x62	98	LATIN SMALL LETTER B		98
0x63	99	LATIN SMALL LETTER C		99
0x64	100	LATIN SMALL LETTER D		100
0x65	101	LATIN SMALL LETTER E		101
0x66	102	LATIN SMALL LETTER F		102
0x67	103	LATIN SMALL LETTER G		103
0x68	104	LATIN SMALL LETTER H		104
0x69	105	LATIN SMALL LETTER I		105
0x6A	106	LATIN SMALL LETTER J		106
0x6B	107	LATIN SMALL LETTER K		107
0x6C	108	LATIN SMALL LETTER L		108
0x6D	109	LATIN SMALL LETTER M		109
0x6E	110	LATIN SMALL LETTER N		110
0x6F	111	LATIN SMALL LETTER O		111
0x70	112	LATIN SMALL LETTER P		112
0x71	113	LATIN SMALL LETTER Q		113
0x72	114	LATIN SMALL LETTER R		114
0x73	115	LATIN SMALL LETTER S		115
0x74	116	LATIN SMALL LETTER T		116
0x75	117	LATIN SMALL LETTER U		117
0x76	118	LATIN SMALL LETTER V		118
0x77	119	LATIN SMALL LETTER W		119
0x78	120	LATIN SMALL LETTER X		120
0x79	121	LATIN SMALL LETTER Y		121
0x7A	122	LATIN SMALL LETTER Z		122
0x7B	123	LATIN SMALL LETTER A WITH DIAERESIS		228

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0x7C	124	LATIN SMALL LETTER O WITH DIAERESIS		246
0x7D	125	LATIN SMALL LETTER N WITH TILDE		241
0x7E	126	LATIN SMALL LETTER U WITH DIAERESIS		252
0x7F	127	LATIN SMALL LETTER A WITH GRAVE		224

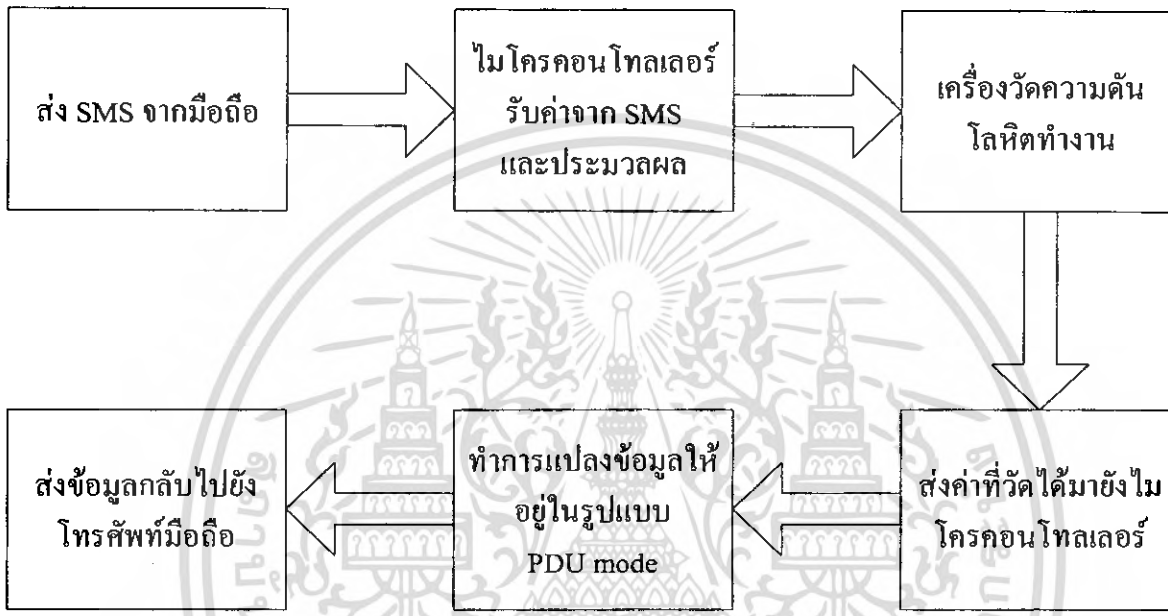
ตารางที่ 2.7 รหัสตัวอักษรชนิด 7 บิต (7 bit default alphabet)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการสร้าง

3.1 การออกแบบ

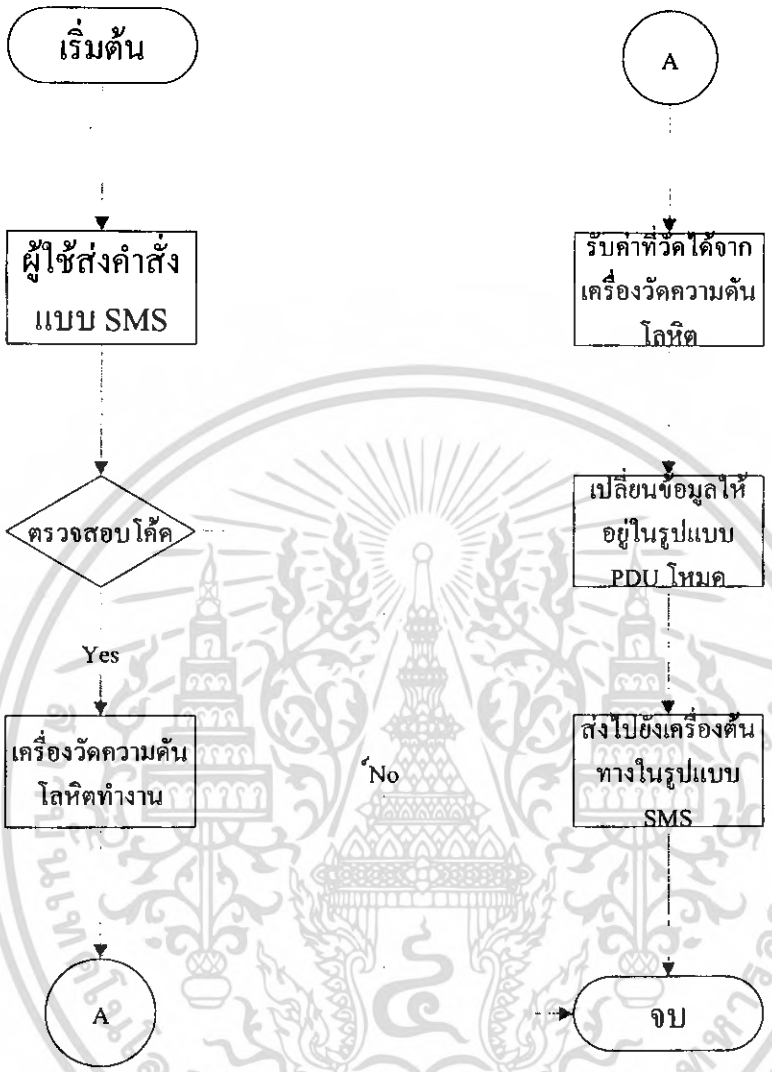
การทำงานของระบบสามารถอธิบายได้จาก Block Diagram ดังนี้



รูปที่ 3.1 Block Diagram อธิบายการทำงานโดยรวมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลว์ชาร์ตแสดงขั้นตอนการทำงานของระบบเป็นดังนี้

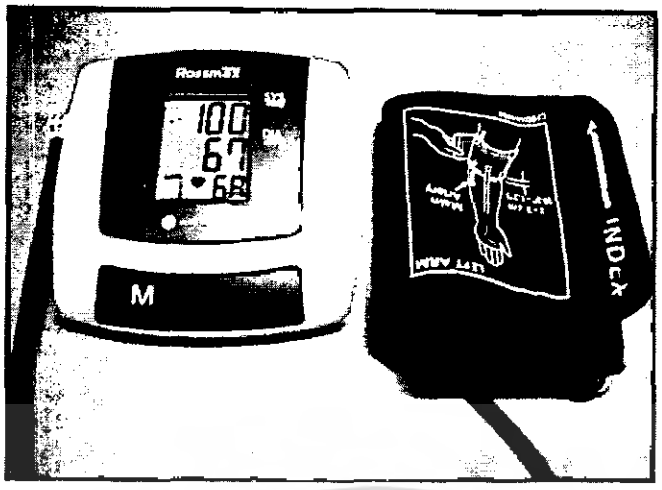


รูปที่ 3.2 โฟล์วชาร์ตแสดงขั้นตอนการทำงานของระบบ

3.2 เครื่องวัดความดันโลหิต

เครื่องวัดความดันโลหิตที่ใช้เป็นแบบดิจิตอล สามารถวัดค่าความดันขณะหัวใจบีบตัว (Systolic) ความดันขณะหัวใจคลายตัว (Diastolic) และอัตราการเต้นของชีพจร (Pulse Rate) ซึ่งเครื่องวัดความดันโลหิตชนิดนี้จะทำงานแบบ One Touch คือเพียงกดปุ่ม Start เพียงครั้งเดียวเครื่องก็สามารถทำงานได้จนเสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 เครื่องวัดความดันโลหิตแบบดิจิตอล

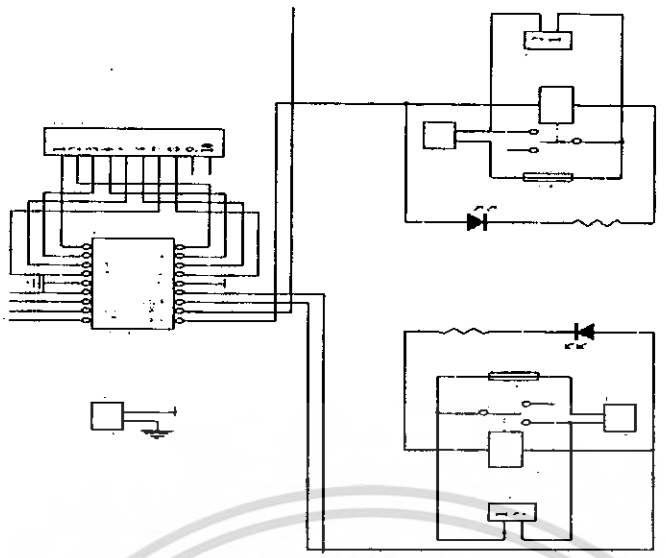
3.3 ส่วนควบคุม

สำหรับส่วนนี้จะทำหน้าที่ในการควบคุมการทำงานทั้งหมดของระแบบไม่ว่าจะเป็นการรับคำสั่งในรูปแบบข้อความ (SMS) การประมวลผล การควบคุมวงจรรีเลย์เพื่อทำการสวิตซ์เครื่องวัดความดันโลหิตให้ทำงาน การนำค่าที่วัดได้จากเครื่องวัดความดันส่งกลับไปยังโทรศัพท์มือถือในรูปแบบของ SMS

วงจรส่วนควบคุม ประกอบด้วย ไมโครคอนโทลเลอร์ โทรศัพท์มือถือ วงจรรีเลย์ ซึ่งโทรศัพท์มือถือจะเชื่อมต่อกับไมโครคอนโทลเลอร์ทางพอร์ทอนุกรม (Serial port) โดยเชื่อมต่อผ่านวงจร Max 232 และไมโครคอนโทลเลอร์จะเชื่อมต่อกับวงจรรีเลย์ผ่านทางพอร์ท 2.1 (ขา 22) ซึ่งหน้าที่ของอุปกรณ์แต่ละส่วนจะแตกต่างกันไป ดังนี้

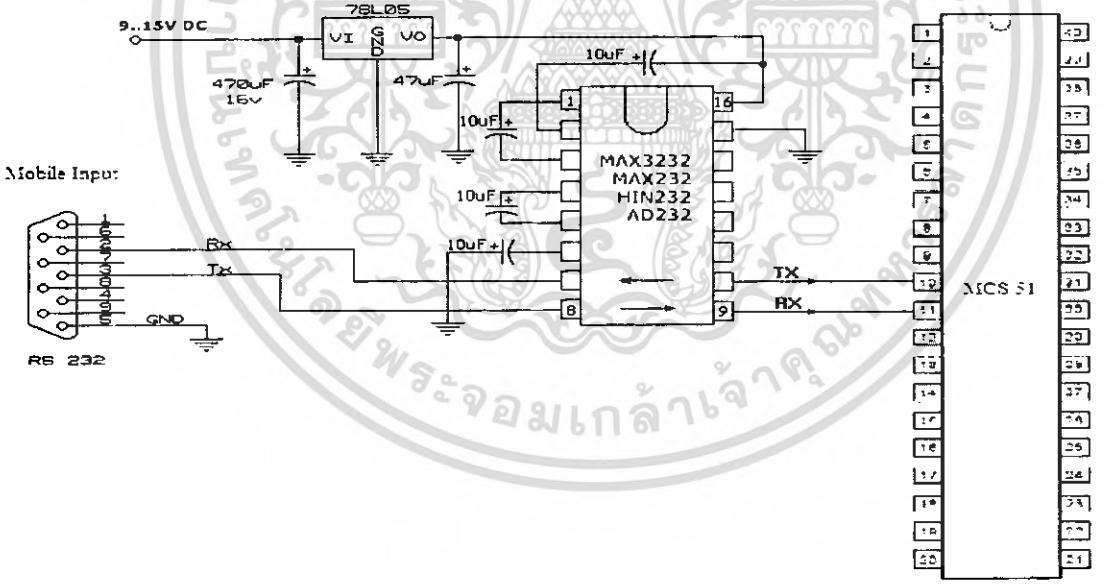
- โทรศัพท์มือถือ มีหน้าที่รอรับข้อความเพื่อส่งไปประมวลผลที่ไมโครคอนโทลเลอร์ ซึ่งโทรศัพท์มือถือจะถูกควบคุมด้วยชุดคำสั่ง AT command
- วงจรรีเลย์ มีหน้าที่สวิตซ์เครื่องวัดความดันโลหิตให้ทำงาน ซึ่งวงจรรีเลย์นี้จะเชื่อมต่อกับไมโครคอนโทลเลอร์ทางพอร์ท 2.1 และจะทำงานที่สัญญาณ ลอจิก "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



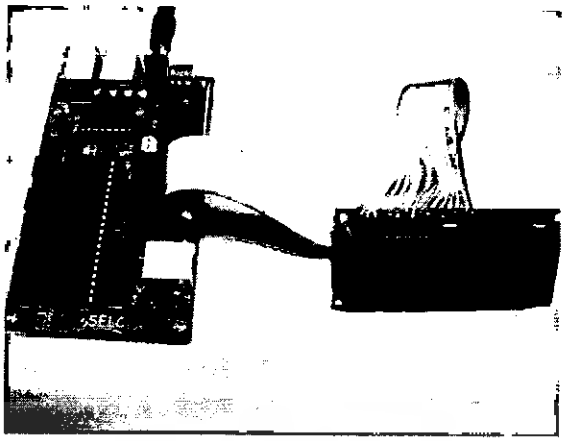
รูปที่ 3.4 วงจรรีเลย์

- ไมโครคอนโทลเลอร์ มีหน้าที่รับข้อมูลจากโทรศัพท์มือถือเพื่อนำไปประมวลผลและทำหน้าที่รับข้อมูลจากเครื่องวัดความดันโลหิตเพื่อส่งกลับไปยังโทรศัพท์มือถือที่อยู่ต้นทาง



รูปที่ 3.5 การเชื่อมต่อไมโครคอนโทลเลอร์กับโทรศัพท์มือถือผ่านทาง Serial port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



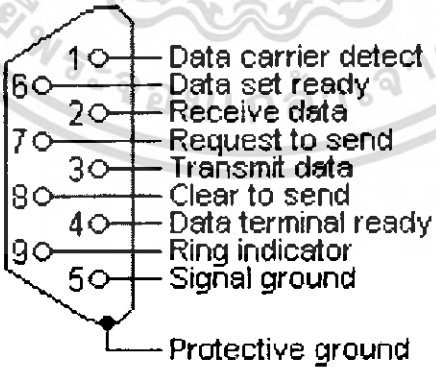
รูปที่ 3.6 วงจรส่วนควบคุม

3.4 การใช้งานพอร์ตอนุกรมมาตราฐาน RS-232

3.4.1 RS-232 serial connecter pin assignment

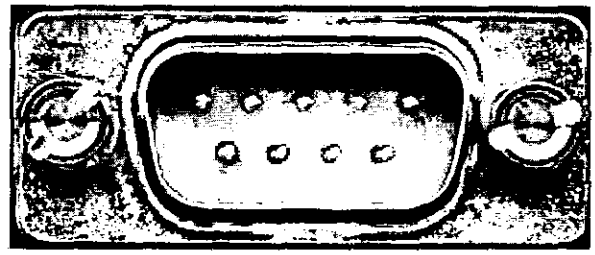
RS-232 คอนเน็คเตอร์ (Connecter) ออกแบบมาสำหรับการสื่อสารแบบอนุกรม RS-232 ถึง 2 ช่องการสื่อสาร (Channel) โดยช่องการสื่อสารช่องหนึ่งที่มีระบบตรวจสอบความถูกต้อง (Handshake) ติดมาด้วย และมีคอมพิวเตอร์บางรุ่นที่มีระบบตรวจสอบความถูกต้องทั้ง 2 ช่องการสื่อสาร ส่วนอีกช่องการสื่อสารเป็นการสื่อสารแบบโมเด็ม (Telebit Modem) ซึ่งสามารถเช็คสถานะของโมเด็มว่าพร้อมหรือไม่พร้อมในการสื่อสาร สำหรับเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วไป (Personal Computer) ในปัจจุบัน

RS-232 คอนเน็คเตอร์จะเป็นแบบเล็ก 9 ขาหรือเรียกว่า DB9 ดังแสดงในรูปที่ 26 โดยแต่ละขาเป็นการสื่อสารแบบโมเด็ม



รูปที่ 3.7 RS-232 คอนเน็คเตอร์แบบ DB9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 RS-232 คอนเน็คเตอร์แบบ DB9 ตัวผู้ที่อยู่ด้านหลังคอมพิวเตอร์

สำหรับเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วไป (Personal Computer) ในปัจจุบันจะเป็น RS-232 คอนเน็คเตอร์แบบเล็ก DB9 ที่หลังคอมพิวเตอร์จะเป็นตัวผู้ ซึ่งมีรายละเอียดของขาดังตารางที่ 11

DB25	DB9	Signal	Name	Direction (In/Out)
2	3	TD	Transmit Data	Output
3	2	RD	Receive Data	Input
4	7	RTS	Request to Send	Output
5	8	CTS	Clear to Send	Input
6	6	DSR	Data Set Ready	Input
7	5	SG	Signal Ground	Ground
8	1	CD	Carrier Detect	Input
20	4	DTR	Data Terminal Ready	Output
22	9	RI	Ring Indicator	Input

ตารางที่ 3.1 รายละเอียดขาสัญญาณของพอร์ตอนุกรม RS-232

โดยมีรายละเอียดหน้าที่การทำงานดังต่อไปนี้

- TD: Transmit Data ใช้สำหรับส่งข้อมูลออก
- RD: Receive Data ใช้สำหรับรับข้อมูลเข้า
- RTS: Request to Send บอกให้ปลายทางเตรียมรับข้อมูล
- CTS: Clear to Send สำหรับตรวจสอบความพร้อมของอุปกรณ์ปลายทางที่จะรับข้อมูล
- DSR: Data Set Ready ใช้ตรวจสอบการเชื่อมต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- SG: Signal Ground กราวน์
- CD: Carrier Detect จะ Active เมื่อมีสัญญาณ Carrier จากโมเด็ม
- DTR: Data Terminal Ready ใช้บอกอุปกรณ์ปลายทางว่าต้องการติดต่อ
- RI: Ring Indicator จะ Active เมื่อโมเด็มได้รับสัญญาณเรียกเข้าจากโทรศัพท์

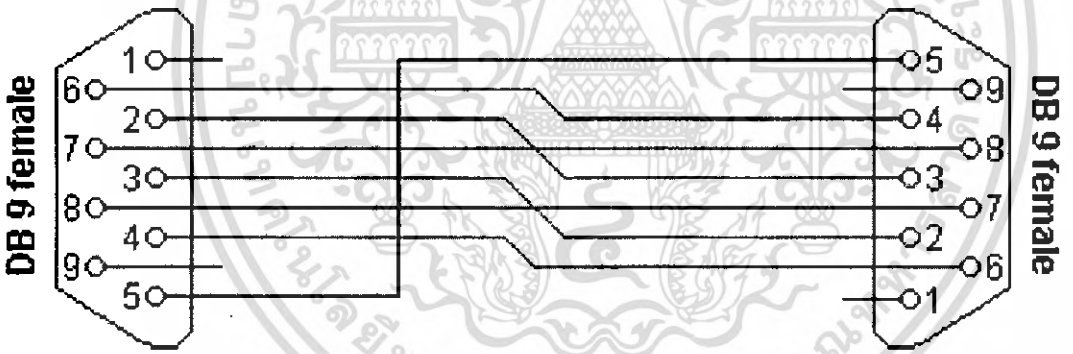
3.4.2 รูปแบบของการเชื่อมต่อ

สำหรับ RS-232 ที่เป็นการสื่อสารแบบโมเด็ม สามารถประยุกต์การเชื่อมต่อระหว่างอุปกรณ์ 2 ตัวโดยใช้สาย RS-232 ซึ่งสามารถรองรับรูปแบบการเชื่อมต่อมากมาย เช่น สามารถใช้สายสัญญาณเพียง 3 เส้น คือ RD, TD และ SG ก็เพียงพอสำหรับการเชื่อมต่อแล้ว โดยรูปแบบการเชื่อมต่อนั้นขึ้นอยู่กับ Software ที่ใช้ ซึ่งบาง Software อาจจะไม่จำเป็นต้องมีการตรวจสอบความผิดพลาดเลย หรือการสื่อสารระหว่างคอมพิวเตอร์ การเชื่อมต่อแบบ Loop Back Handshaking นั้นเป็นแบบที่เหมาะสมที่สุด

การเชื่อมต่อระหว่างอุปกรณ์ 2 ตัวนั้นมีหลากลายรูปแบบ เช่น Loop Back Handshaking เหมาะสาหรับการเชื่อมต่อคอมพิวเตอร์ หรือ แบบเต็มรูปแบบเหมาะสำหรับระบบใหญ่ 2 ระบบ เป็น

- Full Handshaking

เป็นการใช้งาน RS-232 แบบเต็มรูปแบบ



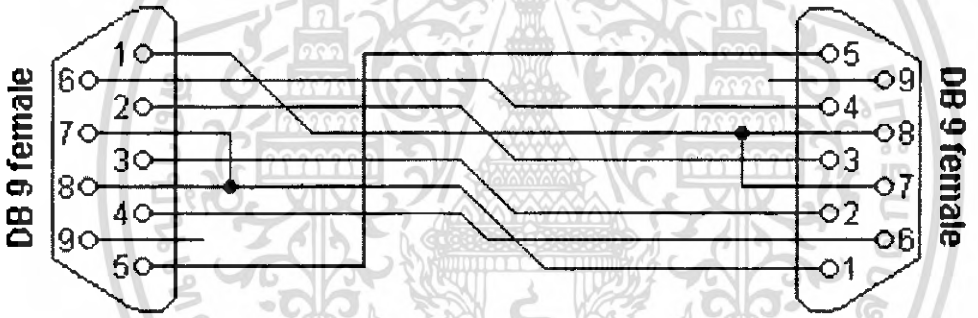
รูปที่ 3.9 ลักษณะการเชื่อมต่อแบบเต็มรูปแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Connector 1	Connector 2	Function
2	3	Rx & Tx
3	2	Tx & Rx
4	6	DTR & DSR
5	5	Signal ground
6	4	DSR & DTR
7	8	RTS & CTS
8	7	CTS & RTS

ตารางที่ 3.2 หน้าที่การทำงานของการเชื่อมต่อแบบเต็มรูปแบบ

- Partial Handshaking



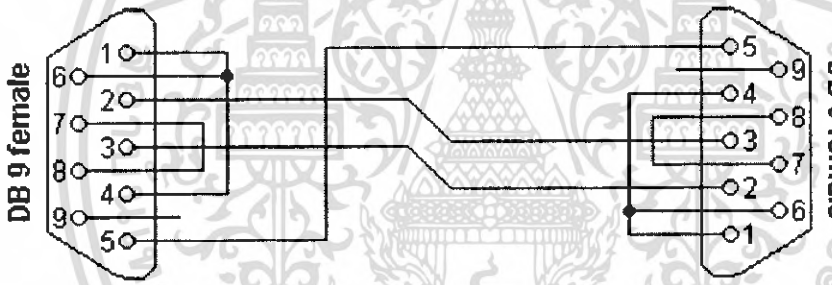
รูปที่ 3.10 ลักษณะการเชื่อมต่อแบบ Partial Handshaking

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Connector 1	Connector 2	Function
1	7 + 8	RTS2 &CTS2 + CD1
2	3	Rx &Tx
3	2	Tx &Rx
4	6	DTR & DSR
5	5	Signal ground
6	4	DSR &DTR
7 + 8	1	RTS1 &CTS1 + CD2

ตารางที่ 3.3 หน้าที่การทำงานของการเชื่อมต่อแบบ Partial Handshaking

- Loop Back Handshaking



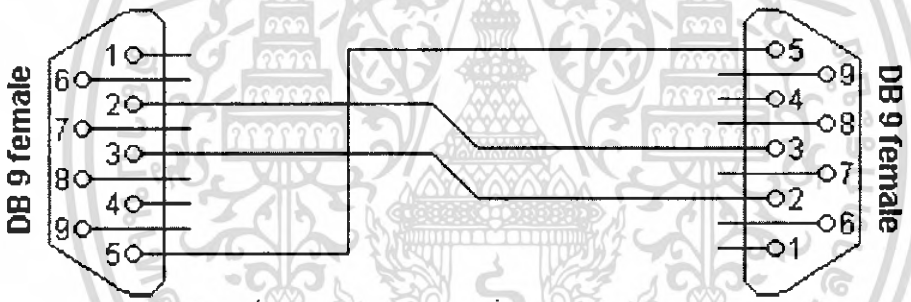
รูปที่ 3.11 ลักษณะการเชื่อมต่อแบบ Loop Back Handshaking

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Connector 1	Connector 2	Function
2	3	Rx &Tx
3	2	Tx &Rx
5	5	Signal ground
1 + 4 + 6	-	DTR & CD + DTR
-	1 + 4 + 6	DTR &CD + DTR
7 + 8	-	RTS &CTS
-	7 + 8	RTS &CTS

ตารางที่ 3.4 หน้าที่การทำงานของการเชื่อมต่อแบบ Loop Back Handshaking

• Simple



รูปที่ 3.12 ลักษณะการเชื่อมต่อแบบง่าย

Connector 1	Connector 2	Function
2	3	Rx & Tx
3	2	Tx & Rx
5	5	Signal ground

ตารางที่ 3.5 หน้าที่การทำงานของการเชื่อมต่อแบบง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การเชื่อมต่อโทรศัพท์มือถือ กับคอมพิวเตอร์

ในการเริ่มต้นขั้นแรกนี้ ทำการติดต่อกับโทรศัพท์มือถือจาก Computer โดยใช้โปรแกรม Hyper Terminal ส่งชุดคำสั่ง AT COMMAND ผ่านสาย Data Link แล้วดูผลตอบสนองจากโทรศัพท์มือถือ

อุปกรณ์ที่ใช้ : โทรศัพท์มือถือ สาย Data link Computer (PC)

รูปที่ 4.1 โทรศัพท์มือถือ และสาย Data Link

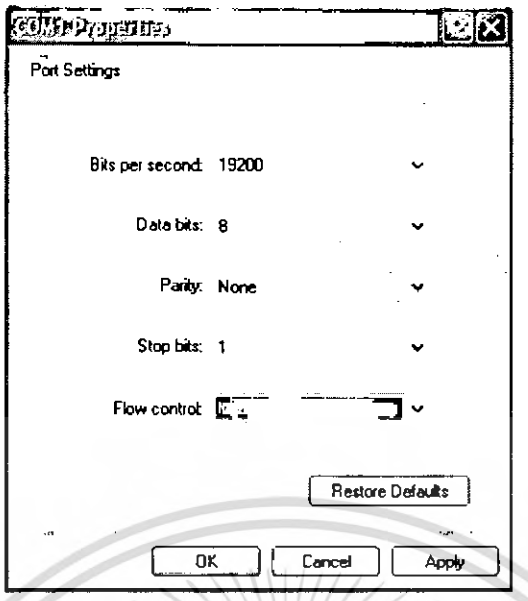
ซอฟต์แวร์ที่ใช้ : Hyper Terminal

หลักการทดสอบ : ใช้โปรแกรม Hyper Terminal ส่งผ่านชุดคำสั่ง AT COMMAND ไปยังโทรศัพท์มือถือ

การทดลองและผลที่ได้ :

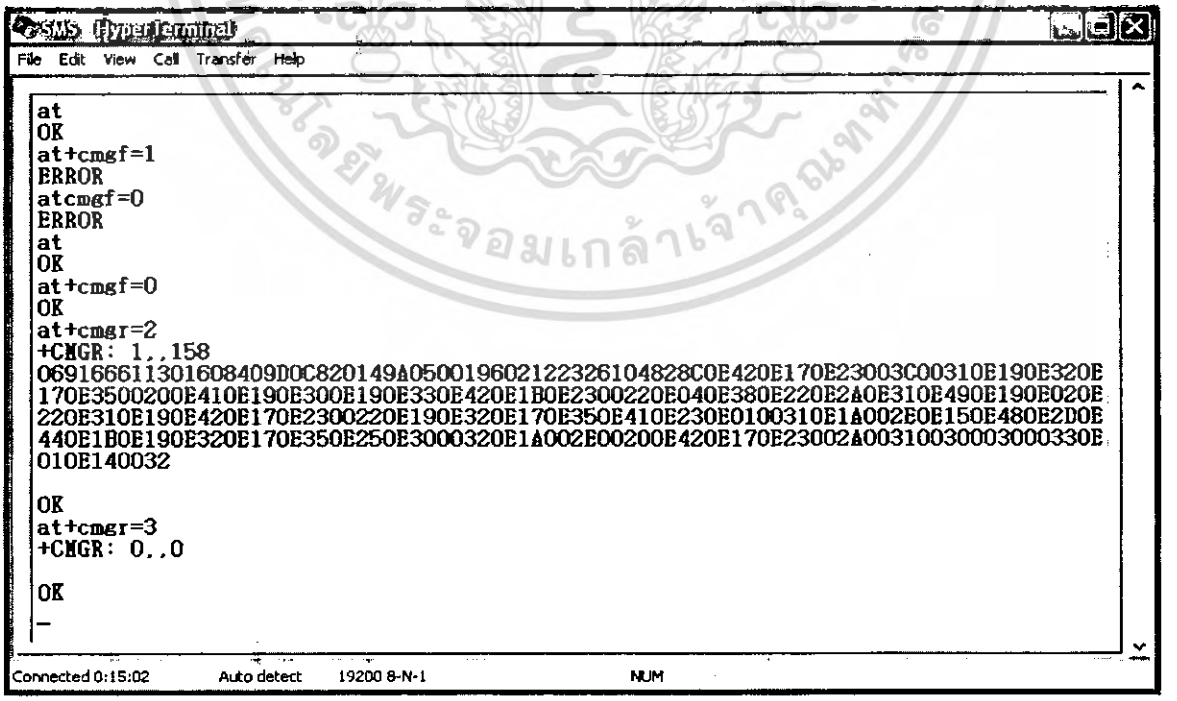
1. เซตอัตราการรับส่งข้อมูลระหว่างโทรศัพท์มือถือกับ PC ให้ตรงกัน สำหรับโทรศัพท์มือถืออัตราการรับ-ส่ง ข้อมูลสามารถเซตได้จากเมนูในตัวเครื่องโทรศัพท์มือถือเอง ซึ่งโทรศัพท์มือถือรุ่นที่ใช้ทดสอบคือ siemens C45 สามารถเซตค่าได้คือ 19200 bps ส่วนอัตรารับส่งข้อมูลของ PC เซตได้จาก Port setting ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

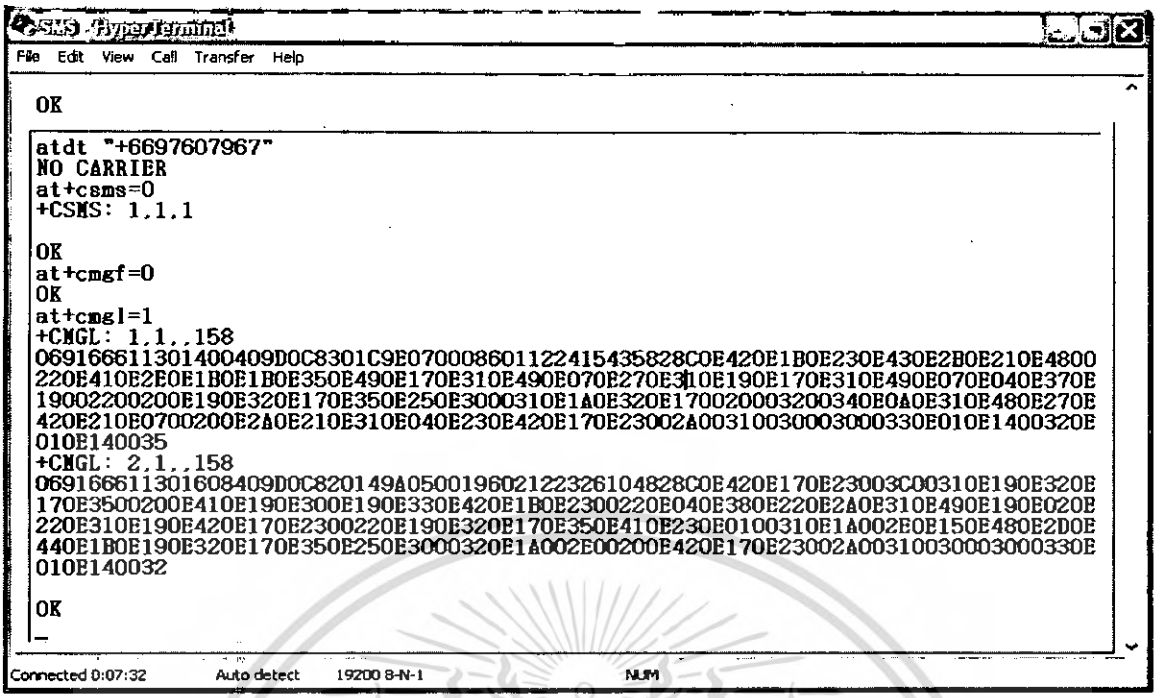


รูปที่ 4.2 Dialog Box Port setting จากโปรแกรม Hyper Terminal

2. เริ่มทำการทดสอบโดยส่งชุดคำสั่ง AT COMMAND ไปยังโทรศัพท์มือถือ เริ่มต้นจากคำสั่ง AT (Enter) หากระบบสามารถเชื่อมต่อได้ โทรศัพท์มือถือจะตอบกลับมาว่า OK จากนั้นจึงทดสอบคำสั่งพื้นฐานต่างๆ เพื่อสั่งการโทรศัพท์มือถือ เช่น โทรออก รับ-ส่งข้อความสั้น อ่านข้อความสั้น ซึ่งได้ผลดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 การทดลองส่ง AT command ไปยังโทรศัพท์มือถือ

4.2 การเชื่อมต่อเครื่องวัดความดันโลหิตกับคอมพิวเตอร์

เมื่อทำการทดลองส่งข้อมูลระหว่างโทรศัพท์มือถือกับคอมพิวเตอร์แล้วจากนั้นต้องทำการเชื่อมต่อเครื่องวัดความดันเข้ากับคอมพิวเตอร์เพื่อทำการดึงข้อมูลออกจากเครื่องวัดความดัน ดังนี้

อุปกรณ์ที่ใช้ : เครื่องวัดความดันโลหิต สาย Datalink สายสื่อสารพอร์ทอนุกรม RS 232 คอมพิวเตอร์

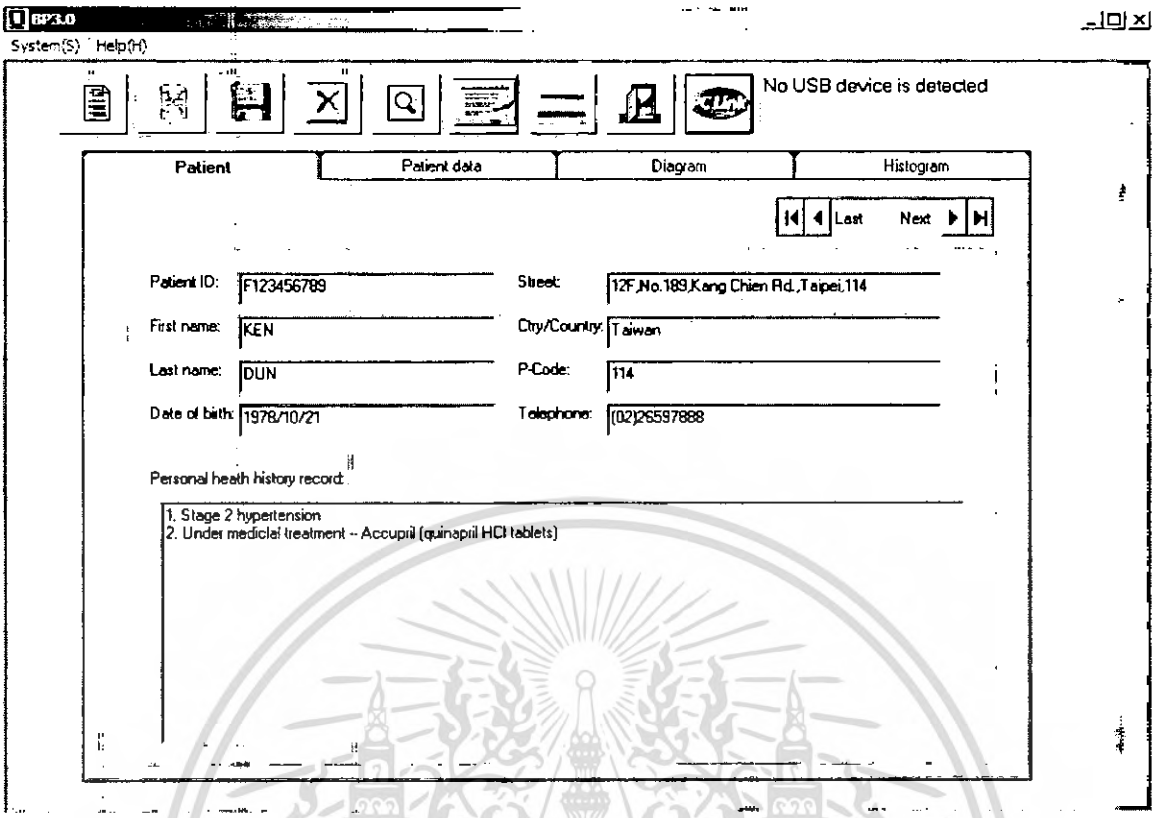
ซอฟต์แวร์ที่ใช้ : BP 3.0 และ Hyperterminal

หลักการทดสอบ : เชื่อมต่อเพื่อดึงข้อมูลจากเครื่องวัดความดันเข้ามายังคอมพิวเตอร์ ผ่านทางโปรแกรม BP3.0 และ Hyperterminal

การทดลองและผลที่ได้ :

เชื่อมต่อเครื่องวัดความดันโลหิตกับคอมพิวเตอร์ผ่านสาย Datalink โดยใช้พอร์ท USB และโปรแกรม BP 3.0 ผลที่ได้คือ คอมพิวเตอร์ไม่สามารถดึงข้อมูลจากเครื่องวัดความดันได้ เพราะคอมพิวเตอร์ไม่สามารถ Detect เครื่องวัดความดันโลหิตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงโปรแกรม BP 3.0 ขณะเชื่อมต่อเครื่องวัดความดันกับคอมพิวเตอร์

เชื่อมต่อเครื่องวัดความดันโลหิตกับคอมพิวเตอร์ผ่านทางพอร์ทอนุกรม RS 232 และใช้โปรแกรม HyperTerminal ผลที่ได้คือไม่สามารถดึงข้อมูลจากเครื่องวัดความดันได้ เพราะโปรแกรมเชื่อมต่อกับเครื่องวัดความดันไม่ได้

4.3 การเชื่อมต่อ Microcontroller โทรศัพท์มือถือ และเครื่องวัดความดันโลหิต

อุปกรณ์ที่ใช้ : เครื่องวัดความดันโลหิต Microcontroller เบอร์ P89V51RD2BN Board ทดลอง IC Max232 สาย Data Link โทรศัพท์มือถือ สายสื่อสารพอร์ทอนุกรม RS 232 DB9 จอ LCD วงจรรีเลย์

ซอฟต์แวร์ที่ใช้ : Flash Magic Keil uVision3

หลักการทดสอบ : ส่งโค้ดข้อความ (SMS) เพื่อสั่งให้เครื่องวัดความดันโลหิตทำงาน

การทดลองและผลที่ได้ :

1. ก่อนทำการส่งข้อความไปให้มือถือที่เชื่อมต่ออยู่กับไมโครคอนโทลเลอร์ ให้กดปุ่มรีเซทไมโครคอนโทลเลอร์หนึ่งครั้ง จอ LCD จะแสดงข้อความ “Standby” ดังรูป

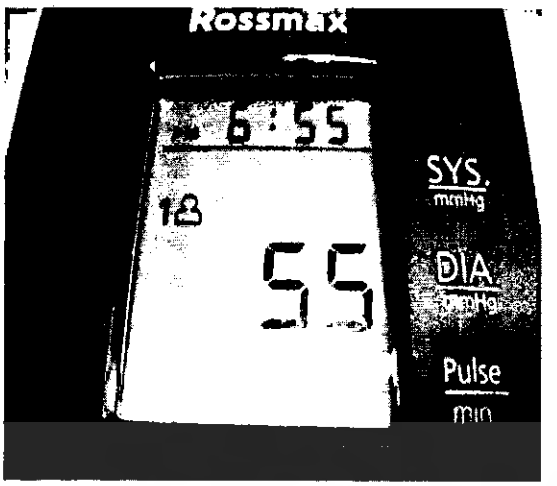
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.5 จอ LCD ในสถานะพร้อมทำงาน

2. เมื่อทำการส่งคำสั่งมายังเครื่องรับ จอ LCD จะแสดงเลขหมายโทรศัพท์ของผู้ที่ส่งคำสั่งมา และจะแสดงข้อความ "Robot Active" พร้อมกับส่งสัญญาณ ลอจิก "1" เพื่อให้ รีเลย์ไปทำการสวิทซ์เครื่องวัดความดันโลหิตให้ทำงาน

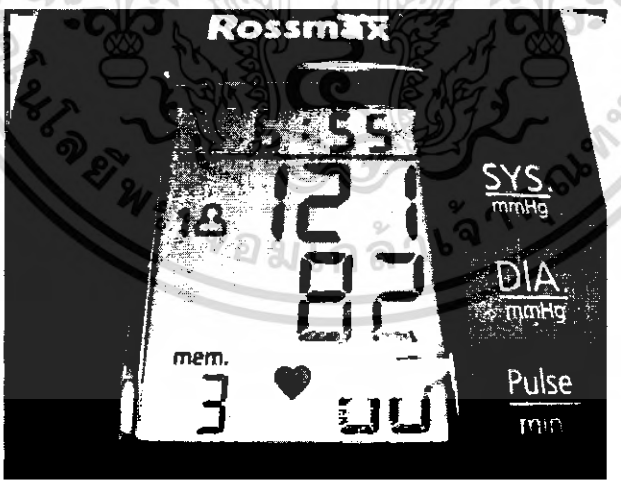
รูปที่ 4.6 ภาพจอ LCD ขณะได้รับ SMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 จอ LCD ของเครื่องวัดความดันโลหิตขณะกำลังทำงาน

เมื่อเครื่องวัดความดันโลหิตทำงานเสร็จ จะได้ค่ามา 3 ค่า ได้แก่ ความดันโลหิตขณะหัวใจบีบตัว ความดันโลหิตขณะหัวใจคลายตัว และอัตราการเต้นของหัวใจ ซึ่งจากจุดมุ่งหมายของโครงงานนี้ ต้องการที่จะทำการส่งค่าที่วัดได้ทั้ง 3 ค่านี้ กลับไปยังโทรศัพท์มือถือในรูปแบบ SMS แต่เนื่องจากไม่สามารถนำข้อมูลที่วัดได้นี้ออกมาจากเครื่องวัดความดันได้ จึงไม่สามารถส่งข้อมูลที่ได้จากการวัดจริง กลับไปยังโทรศัพท์มือถือได้ ซึ่งในการทดลองนี้ ได้ทำการส่งค่าที่ทำการสมมติขึ้นมากลับมายังโทรศัพท์มือถือเครื่องต้นทางแทน



รูปที่ 4.8 แสดงภาพหน้าจอ LCD ของเครื่องวัดความดันเมื่อทำงานเสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลอง

5.1 สรุปปัญหาจากการทดลอง

1. ไม่สามารถดึงข้อมูลออกมาจากเครื่องวัดความดันโลหิตได้ ทำให้ไม่ทราบว่า ข้อมูลมีลักษณะอย่างไร จึงไม่สามารถส่งผลที่ได้กลับมายังโทรศัพท์มือถือได้

2. เนื่องจากเครื่องวัดความดันโลหิตเป็นแบบสำเร็จรูป ซึ่งมีการทำงานที่ซับซ้อนพอสมควร ทำให้เมื่ออุปกรณ์เกิดการเสียหาย ซึ่งผู้ทดลองไม่สามารถตรวจสอบและแก้ไขได้

3. เมื่อส่งคำสั่งให้ เครื่องวัดความดันโลหิตทำงานบางครั้งอาจเกิดความล่าช้าในการทำงาน เนื่องจากความไม่เสถียรของวงจร

4. ถ้าโทรศัพท์ที่ใช้รับสัญญาณได้รับข้อความเต็มกล่องข้อความแล้ว เมื่อส่งคำสั่งมาให้เครื่องวัดความดันทำงาน ระบบจะไม่สามารถตอบสนองต่อคำสั่งได้เนื่องจาก ไม่สามารถอ่านข้อความใหม่ได้

5.2 สรุปผลการทดลอง

จากการทำโครงงานนี้ ได้ศึกษาในหลายๆด้าน ได้แก่ ความดันโลหิต ไมโครคอนโทรเลอร์ การรับส่งสัญญาณผ่าน Serial Port GSM AT command กับโทรศัพท์มือถือ ซึ่งจากการทำโครงงานในครั้งนี้ ได้นำเอาความรู้ทางด้าน การสื่อสาร อิเล็กทรอนิกส์ และคอมพิวเตอร์ มาประยุกต์ใช้งานให้เกิดผลตามเป้าหมายที่วางไว้ ทำให้ได้รับความรู้และมีความเข้าใจในการนำเทคโนโลยีการสื่อสารไร้สายมาพัฒนาและประยุกต์ใช้ให้เกิดประโยชน์ในด้านต่างๆ มากยิ่งขึ้น

โครงงานนี้ถือว่าได้บรรลุตามจุดประสงค์ แต่ไม่สามารถทำได้ตามขอบเขตที่ของงานที่ตั้งไว้ คือ สามารถส่งข้อความ สั่งให้เครื่องวัดความดันโลหิตทำงานได้ แต่ไม่สามารถส่งค่าที่วัดได้จริงกลับมายังโทรศัพท์มือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ธีรวัฒน์ ประกอบผล, **"การประยุกต์ใช้งานไมโครคอนโทรลเลอร์"**, สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), 2543

[2] สุธี พงศาสกุลชัย, หทัยชนก งามอินทร์, **"คัมภีร์ C"**, บริษัท เคทีพี คอมพ์ แอนด์ คอนซัลท์ จำกัด, 2542

[3] โอภาส ศิริครรชิตถาวร, **"เรียนรู้และพัฒนาไมโครคอนโทรเลอร์ ด้วยภาษาซี"**, โรงพิมพ์วชิรินทร์สาสน์ รัชดา, 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้