สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบตรวจสอบบุคคลบนรถยนต์ด้วยเทคโนโลยี GSM

PERSONAL MONITORING ON VEHICLE BY GSM TECHNOLOGY

นาย ชวลิต จิราพัชรนิธิสกุล

CHAVALIT JIRAPACHNITISAKUL

นาย ชัยพฤกษ์ ทองคำ

CHAIYAPRUKE YHONGKHAM

นางสาว ฐิตาภา ธินันทรักษ์

TITAPA TINANTARUG

เลขหมู่.....
เลขทะเบียน 117550
วัน,เดือน,ปี - 5 มี.ค. 2554

b 1234306Y
i

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PERSONAL MONITORING ON VEHICLE BY GSM TECHNOLOGY

CHAVALIT JIRAPACHNITISAKUL

CHAIYAPRUKE YHONGKHAM

TITAPA TINANTARUG

THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT

OFTHE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2010

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบตรวจสอบบุคคลบนรถยนต์ด้วยเทคโนโลยี GSM
รายชื่อนักศึกษา	นาย ชวลิต จิราพัชรนิธิสกุล รหัส 50010329
	นาย ชัยพฤกษ์ ทองคำ รหัส 50010342
	นางสาว ฐิตาภา ธินันทรักษ์ รหัส50010405
ปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมสารสนเทศ
พ.ศ.	2553
อาจารย์ที่ปรึกษาปริญญานิพนธ์	รศ.ดร.ปิติเขต สู้รักษา

ปริญญานิพนธ์ฉบับนี้ ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรวิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง

(รศ.ดร. ปิติเขต สู้รักษา)
อาจารย์ผู้ควบคุมปริญญานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบตรวจสอบบุคคลบนรถยนต์ด้วยเทคโนโลยี GSM
รายชื่อนักศึกษา	นาย ชวลิต จิราพัชรนิธิสกุล รหัส 50010329
	นาย ชัยพฤกษ์ ทองคำ รหัส 50010342
	นางสาว ฐิตาภา ธินันทรักษ์ รหัส 50010405
ปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมสารสนเทศ
พ.ศ.	2553
อาจารย์ที่ปรึกษาปริญญานิพนธ์	รศ.ดร.ปิติเขต สู้รักษา

บทคัดย่อ

ปริญญานิพนธ์นี้กล่าวถึงการนำ GSM Module ซึ่งก็คือ โมดูลสำหรับสร้างโทรศัพท์มือถือ มีคุณสมบัติมากมาย ตัวอย่างเช่น สามารถรองรับสัญญาณ GPRS และ รองรับ SIM Applications Toolkit ดังนั้น เราจึงนำ GSM Module มาช่วยเพิ่มประสิทธิภาพให้แก่ระบบกันขโมยบนรถยนต์ ซึ่งจะทำงานเมื่อระบบเตือนภัยที่อยู่ในรถนั้นเตือนขึ้นมา กล้องจะทำการถ่ายภาพเหตุการณ์ที่อยู่ในรถยนต์ จากนั้นจะส่งข้อความแจ้งให้เจ้าของได้ทราบบนโทรศัพท์มือถือ และส่งไฟล์ภาพที่ได้ไปเก็บที่เครื่องเซิร์ฟเวอร์(Server) เมื่อเจ้าของรถทราบว่ามีสิ่งผิดปกติเกิดขึ้นภายในรถยนต์ ก็จะสามารถดาวน์โหลดภาพเหตุการณ์ในรถมาตรวจสอบได้ว่าสิ่งที่เกิดขึ้นในรถนั้นโดนโจรกรรมจริงหรือไม่ หรือเป็นเหตุขัดข้อง นอกจากนี้ถ้าเกิดการโจรกรรมจริงๆ ภาพที่อยู่บนเซิร์ฟเวอร์สามารถเป็นหลักฐานในการหาผู้กระทำผิดได้ง่ายขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Personal Monitoring on Vehicle by GSM Technology		
Student	Mr. Chavalit	Jirapachnitisakul	Student ID. 50010329
	Mr. Chaiyapruke	Thongkham	Student ID. 50010342
	Mr.Titapa	Tinantarug	Student ID. 50010405
Degree	Bachelor of Engineering		
Program	Information Engineering		
Year	2010		
Thesis Advisor	Assoc.Prof.Dr.Pitikhate Sooraksa		

ABSTRACT

This project presents using GSM Module which it is module for create mobile phone which it has many qualification for example support signal GPRS, support SIM Applications Toolkit .Therefore, the project uses GSM Module to improve the efficiency of the car theft warning system. The warning system will be activated when there is the car theft. A camera will take pictures of the incident in the car and sends alert SMS to the car owner, and also sends the file of pictures in order to store it in the server. When a car owner knows that there is an abnormality in his car, he will download the pictures of an incident in his car in order to check that there is a car theft or some accident in the car. However, the recorded pictures from the server can be an evidence to find the criminal easily.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานระบบตรวจสอบบุคคลบนรถยนต์ด้วยเทคโนโลยีGSMนี้ จะไม่สำเร็จได้เลยถ้าไม่ได้รับคำแนะนำและคำปรึกษาจาก รศ.ดร.ปิติเขต สู้รักษา ซึ่งเป็นอาจารย์ที่ปรึกษาของโครงงานนี้ พวกเรารู้สึกซาบซึ้งในความอนุเคราะห์จากท่านอาจารย์ และขอขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณคณาจารย์สาขาวิชาวิศวกรรมสารสนเทศ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกๆท่านที่ได้ประสิทธิประสาทวิชาให้กับพวกเรา

ขอขอบคุณพี่ๆ เพื่อนๆที่คอยช่วยเหลือในเรื่องของการทำฮาร์ดแวร์และให้ความช่วยเหลือแนะนำและให้คำปรึกษาในเรื่องโครงงานเป็นอย่างดี

และนอกจากนี้พวกเราก็ยังมีอาจารย์ที่คอยให้คำปรึกษาในเรื่องของการหาข้อมูลความรู้ต่างๆ คือ อาจารย์กูเกิล แห่ง www.google.com

สุดท้ายนี้พวกเราขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของพวกเราที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่อง ทำให้พวกเราสามารถทำโครงงานนี้สำเร็จลุล่วงด้วยดี

ขอขอบคุณเพื่อนๆชาวคณะขอนลอยและห้อง 905 และ911 ที่คอยให้กำลังใจ เอื้อเฟื้อแก่กันเป็นอย่างดี คอยช่วยเหลืออยู่ตลอดเวลา เมื่อมีปัญหา

ขอให้คุณประโยชน์อันใดที่จะสามารถพึงเกิดขึ้นได้ของโครงงานนี้ ได้ส่งผลกลับไปถึงผู้มีพระคุณทุกๆท่าน

นาย ชวลิต จิราพัชรนิธิสกุล

นาย ชัยพฤกษ์ ทองคำ

นางสาว ฐิตาภา ธินันทรักษ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VIII

สารบัญรูป X

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหาของโครงงาน 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตของโครงงาน 2

1.4 ผลที่คาดว่าจะได้รับ 2

1.5 อุปกรณ์ที่ต้องใช้ 3

1.5.1 ฮาร์ดแวร์ 3

1.5.2 ซอฟต์แวร์ 3

1.6 ขั้นตอนการทำงาน 3

1.7 ส่วนประกอบของโครงงาน 5

บทที่ 2 ทฤษฎีและหลักการ 6

2.1 ภาษาและแอพพลิเคชั่นที่ใช้ในการทำโครงงาน 6

2.2 ฮาร์ดแวร์ที่ใช้ในการทำโครงงาน 7

2.2.1 ทางด้านผู้ใช้ 7

2.2.2 ทางด้านเซิร์ฟเวอร์ 7

2.2.3 ระบบภายในรถยนต์ 7

2.3 การตรวจจับสิ่งแปลกปลอม 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.3.1 การใช้งานเบื้องต้นของ PIR Motion Sensor 8

2.3.2 หลักการทำงานของ PIR Motion Sensor 10

2.3.3 หลักในการเลือกใช้เซนเซอร์ให้เหมาะกับงาน 10

2.4 การจับภาพจากกล้อง 11

2.4.1 คุณสมบัติของกล้องชนิด UCAM Serial JPEG Camera Module 12

2.4.2 การทำงานของกล้อง UCAM Serial JPEG Camera Module 12

2.4.3 หลักการถ่ายภาพของกล้อง UCAM Serial JPEG Camera Module 14

2.4.4 การเขียนข้อมูลภาพลงบน SD Card 15

2.5 การส่งการเตือนไปให้แก่ผู้ใช้โดยการส่ง SMS 16

2.5.1 ตัวอย่างคำสั่ง AT-Command 16

2.5.2 หลักการส่ง SMS ของ GSM Module 17

2.5.3 คุณสมบัติของ GSM Module 17

2.5.4 ข้อดีของการใช้ GSM Module 18

2.6 การส่งภาพขึ้นเซิร์ฟเวอร์ 19

2.7 GSM 21

2.7.1 โครงสร้างของระบบ GSM 23

2.7.1.1 เครื่องโทรศัพท์เคลื่อนที่ (Mobile Station, Mobile Equipment) 23

2.7.1.2 ระบบสถานีฐาน (BSS) 24

2.7.1.3 Switching System (SS) 24

2.7.1.4 Operation and Maintenance Center (OMC) 26

2.8 GPRS 26

2.8.1 ข้อดีของการใช้ GPRS 27

2.9 โปรแกรม Filezilla 28

2.10 การแสดงภาพบนมือถือ 29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.11 การรักษาความปลอดภัยของการสื่อสารข้อมูลและการเข้าถึงระบบ 30

2.11.1 ด้านผู้ใช้ 30

2.11.2 ด้านเซิร์ฟเวอร์ 31

2.11.3 ระบบภายในรถยนต์ 31

2.12 ส่วนของวงจรและอุปกรณ์ที่ใช้ติดตั้งกล้องในรถยนต์ 31

2.12.1 บอร์ดไมโครคอนโทรลเลอร์AVR ATmega128 31

2.12.2 ไมโครคอนโทรลเลอร์ AVR ATmega128 33

2.12.3 จอแสดงผล LCD (LCD Display) และ สวิตซ์ควบคุม 35

2.12.4 Real Time Clock ด้วย DS1307 35

2.12.5 การสื่อสารผ่านพอร์ทอนุกรม 41

2.12.5.1 มาตรฐานพอร์ท RS-232 41

2.125.2 สายสัญญาณผ่านพอร์ท RS-232 42

2.12.5.3 ชื่อและหน้าที่ของสัญญาณพอร์ทอนุกรม 43

2.12.5.4 มาตรฐานการรับส่งข้อมูลแบบอะซิงโครนัส(UART) 44

2.12.6 MODULE ET-MINI 232-TTL3 48

บทที่ 3 การออกแบบการทดลอง 51

3.1 ระบบตรวจสอบบุคคลภายในรถยนต์ 52

3.2 การตรวจจับสิ่งผิดปกติ 54

3.3 การจับภาพด้วย Camera Module 54

3.4 เซิร์ฟเวอร์ที่ใช้เก็บข้อมูล 56

3.5 การทำงานของ GSM Module 56

3.6 การส่ง SMS แจ้งเตือนไปให้กับเจ้าของรถยนต์ 59

3.7 การส่งภาพขึ้นเซิร์ฟเวอร์ 60

บทที่ 4 ผลการทดสอบ 63

4.1 ฮาร์ดแวร์ที่ใช้ในระบบ 63

4.2 การทดสอบโปรแกรม จะแสดงขั้นตอนผ่านทางหน้าจอ LCD 66

4.2.1 การทดสอบการเปิดระบบ 66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

4.2.2 การทดสอบการเชื่อมต่อ GPRS โดยใช้ GSM Module 67
4.2.3 การทดสอบการเชื่อมต่อกับ FTP Server พอร์ต 21 67
4.2.4 การทดสอบตรวจสิ่งผิดปกติ และเก็บลง SD Card 69
4.2.5 การทดสอบการส่งภาพขึ้น FTP Server 70
4.2.6 การทดสอบการส่ง SMS 71
4.3 ผลการทดลอง 72
4.3.1 แสดงข้อความ SMS บนมือถือ 72
4.3.2 แสดงไฟล์ภาพบนมือถือ 73
4.4 การนำฮาร์ดแวร์ของระบบตรวจสอบบุคคลติดตั้งบนรถยนต์ 74
บทที่ 5 สรุปผลการทดลอง 75
5.1 สรุปการพัฒนาโครงงาน 75
5.1.1 ด้านผู้ใช้ 75
5.1.2 ด้านเซิร์ฟเวอร์ 75
5.1.3 ระบบตรวจสอบบนรถยนต์ 75
5.2 ปัญหาและข้อจำกัดของโครงงาน 75
5.3 แนวทางแก้ไข 76
5.4 การประยุกต์ใช้ต่อไปในอนาคต 76
บรรณานุกรม 77

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

1.1 แผนการดำเนินโครงงาน 4

2.1 คำสั่งของกล้อง UCAM Serial JPEG Camera Module 12

2.2 การกำหนดขนาดภาพ 14

2.3 การควบคุมความถี่ออสซิลเลเตอร์ด้วยการเซตบิต RS1, RS0 40

2.4 รายละเอียดของขาสัญญาณพอร์ทอนุกรม 43

2.5 ความหมายของบิตต่างๆในรีจิสเตอร์สถานะของ UART 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

1.1 ขอบเขตของโครงงาน 2

2.1 ส่วนด้านหน้าของ PIR Sensor 9

2.2 วงจรของ PIR Sensor 9

2.3 การทำงานของ PIR Sensor 10

2.4 กล้องชนิด UCAM Serial JPEG Camera Module 11

2.5 เนื้อไฟล์ข้อมูลของภาพหนึ่งภาพ 15

2.6 GSM Module 19

2.7 FTP Model 20

2.8 การติดต่อระหว่าง เซิร์ฟเวอร์ 21

2.9 ระบบโทรศัพท์เคลื่อนที่ GSM 22

2.10 โครงสร้างของระบบ GSM 23

2.11 Switching System (SS) 25

2.12 การใช้งาน GPRS 27

2.13 โปรแกรม Filezilla 29

2.14 การแสดงภาพบนมือถือ 30

2.15 ส่วนต่างๆที่สำคัญของบอร์ด AVR ATmega128 32

2.16 ไมโครคอนโทรลเลอร์ ตระกูล AVR ATmega128 34

2.17 วงจรไมโครคอนโทรลเลอร์ ตระกูล AVR ATmega128 34

2.18 จอแสดงผล LCD (LCD Display) และสวิตซ์ควบคุม 35

2.19 ตำแหน่งขาไอซี RTC DS1307 36

2.20 การเชื่อมต่อ DS1307 เข้ากับไมโครคอนโทรลเลอร์ด้วยระบบบัสแบบ I2C 37

2.21 การรับส่งข้อมูลผ่านบัส I2C 38

2.22 การเขียนข้อมูลอุปกรณ์ Slave ผ่านบัส I2C 39

2.23 การอ่านข้อมูลจากอุปกรณ์ Slave ผ่านบัส I2C 39

2.24 รีจิสเตอร์ภายในไอซีฐานเวลา DS1307 40

2.25 อุปกรณ์สื่อสารในมาตรฐานพอร์ท RS-232 42

2.26 ตำแหน่งขาพอร์ท DB9 และ DB25 43

2.27 ตำแหน่งขาของสายเชื่อมต่อพอร์ทอนุกรม 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

2.28 การแปลงข้อมูลแบบขนานเป็นข้อมูลอนุกรม 44
2.29 เวิร์ดข้อมูลขนาด 8 บิตกับบิตสตาร์ท และบิตสต๊อปที่ใช้ในการโอนย้ายข้อมูลแบบอนุกรม 44
2.30 บล็อกไดอะแกรมของ UART 45
2.31 รีจิสเตอร์สถานะของ UART 46
2.32 พินของ 16550, 16450 & 8250 UARTs 47
2.33 Module ET-MINI 232-TTL3 48
2.34 โครงสร้าง ET-MINI 232-TTL3 และตำแหน่งขา Port ใช้งาน 49
2.35 วงจร ET-MINI 232-TTL3 50
3.1 ภาพรวมของระบบ 51
3.2 Flow Chart การทำงานของระบบตรวจสอบภายในรถยนต์ 53
3.3 Flow Chart การทำงานของ GSM Module 57
3.4 Flow Chart การทำการอัพโหลดไฟล์ภาพไปยังเซิร์ฟเวอร์ 61
4.1 ภาพรวมของโครงงาน (บอร์ด AVR ATmega128 ด้านหน้า) 63
4.2 ภาพรวมของโครงงาน (บอร์ด AVR ATmega128 ด้านบน) 64
4.3 ภาพรวมโครงงาน เมื่อประกอบเสร็จเรียบร้อยพร้อมใช้งานจริง 65
4.4 เริ่มต้นการทำงานของระบบ 66
4.5 แสดงสถานะเชื่อมต่อ GPRS 67
4.6 แสดงสถานะเชื่อมต่อกับ FTP Server 68
4.7 แสดงสถานะเชื่อมต่อกับ FTP Server เสร็จสมบูรณ์แล้ว 68
4.8 แสดงสถานะจับภาพ 69
4.9 แสดงสถานะเก็บภาพลง SD Card 69
4.10 แสดงสถานะไม่มี SD Card คงอยู่บนระบบ 70
4.11 แสดงสถานะอัพโหลดภาพขึ้น FTP Server 70
4.12 แสดงสถานะส่ง SMS 71
4.13 แสดง การเปิดข้อความ SMS 72
4.14 แสดง การเปิดไฟล์ภาพจากมือถือ 73
4.15ฮาร์ดแวร์ของระบบตรวจสอบบุคคล เมื่อนำมาใช้งานจริง 74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหาของโครงงาน

ปัจจุบันนี้เทคโนโลยีได้เข้ามามีความสำคัญมากขึ้นในการดำรงชีพของบุคคลทั่วไป ทั้งการสื่อสาร การคมนาคม เป็นต้น จึงอาจเป็นไปได้ว่า เมื่อเทคโนโลยีมีการพัฒนาไปมากเท่าไหร่ การนำเทคโนโลยีไปใช้ในทางที่ไม่ดีนั้นก็จะมีมากขึ้นไปด้วย

การโจรกรรมรถยนต์ส่วนบุคคลในปัจจุบันนั้นมีแนวโน้มเพิ่มสูงขึ้นอย่างสม่ำเสมอ และตามหาคืนได้ยาก ระบบสัญญาณกันขโมยในปัจจุบันส่วนใหญ่เป็นการส่งเสียงดังขึ้นขณะที่มีผู้บุกรุกนั้น อาจจะเกิดสัญญาณดังขึ้นเองโดยไม่ได้ตั้งใจจนบ่อยครั้งคนรอบข้างไม่ให้ความสนใจ รวมทั้งเจ้าของรถยนต์อาจไม่ได้ยินเสียงสัญญาณกันขโมย และไม่ทราบว่ารถของตนเกิดอะไรขึ้น ประจวบกับเทคโนโลยีในด้านการติดต่อสื่อสารนั้นแพร่หลายมากกว่าแต่ก่อน โดยเฉพาะอย่างยิ่งโทรศัพท์มือถือ จนเหมือนกับเป็นปัจจัยที่ 5 ในชีวิตประจำวันของคนทั่วไป จึงคิดหาแนวทางที่จะนำเอาโทรศัพท์มือถือมาใช้ให้เกิดประโยชน์

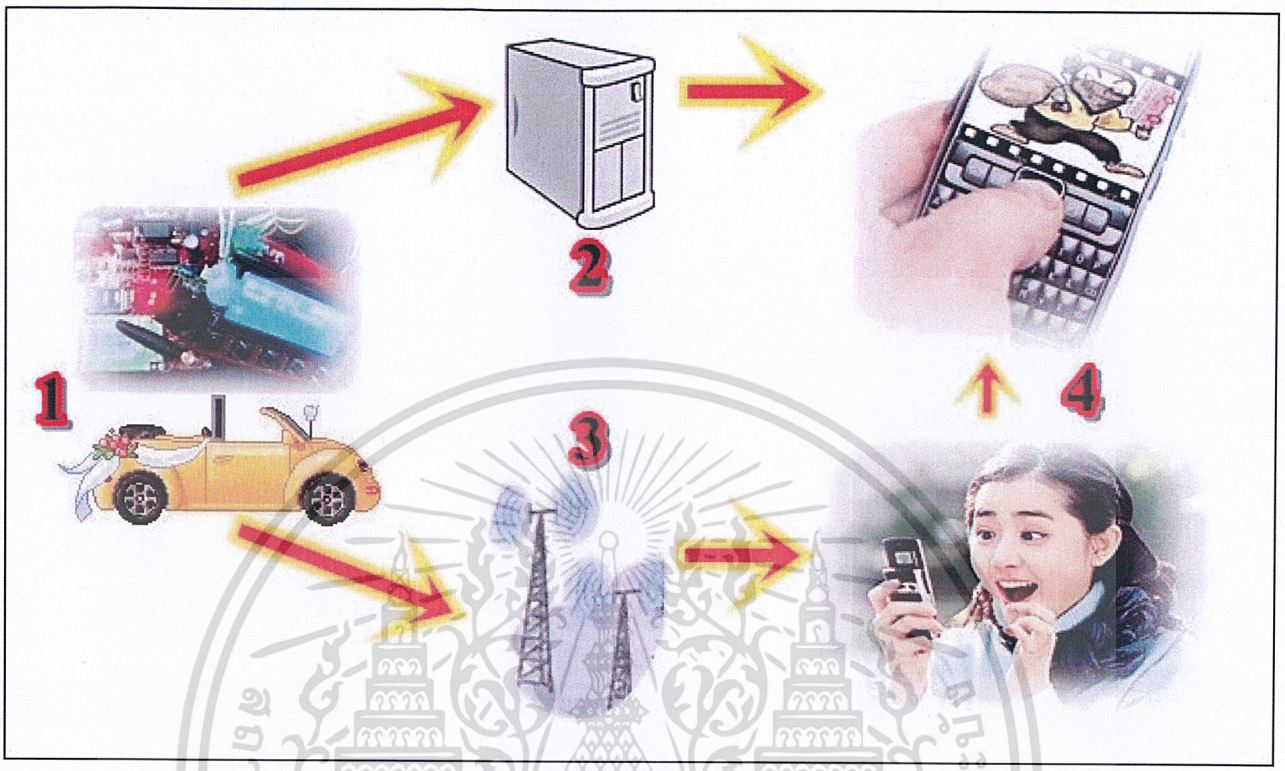
ดังนั้นจึงเห็นว่าหากนำระบบ GSM นี้มาใช้ร่วมกับระบบกันขโมย ทำให้ระบบกันขโมยบนรถยนต์ที่มีประสิทธิภาพ หากมีการโจรกรรมเกิดขึ้น ไม่ว่าเจ้าของรถยนต์จะอยู่ที่ไหนก็สามารถทราบถึงสิ่งผิดปกติที่เกิดขึ้นกับรถยนต์ได้ อีกทั้งโครงงานนี้ยังลงทุนน้อย และเป็นฝีมือที่เราทำขึ้นมาเอง เราจึงสามารถนำไปประยุกต์การใช้งานในแบบอื่นๆ ก็ได้

1.2 วัตถุประสงค์ของโครงงาน

- เพื่อสร้างระบบกันขโมยที่มีประสิทธิภาพดีกว่าในปัจจุบัน โดยนำเทคโนโลยีบนมือถือมาประยุกต์ใช้
- เพื่อเรียนรู้วิธีการติดต่อและควบคุมระหว่างฮาร์ดแวร์และซอฟต์แวร์ผ่านพอร์ตอนุกรม
- เพื่อเรียนรู้การอัพโหลดและดาวน์โหลดข้อมูลไปที่เซิร์ฟเวอร์
- เพื่อเรียนรู้การเขียนโปรแกรมภาษาซี ที่ใช้ในการควบคุมการทำงานของฮาร์ดแวร์ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน



รูปที่ 1.1 ขอบเขตของโครงงาน

ภาพรวมของโครงงานแสดงได้ดังรูปที่ 1.1 และมีขอบเขตของโครงงานดังนี้

1.3.1 ระบบสามารถถ่ายภาพอัตโนมัติเมื่อเซนเซอร์ตรวจจับสิ่งผิดปกติได้

1.3.2 ระบบสามารถอัพโหลดภาพไปเก็บไว้ที่เซิร์ฟเวอร์ได้

1.3.3 ระบบสามารถส่งข้อความเตือนไปยังมือถือได้โดยอัตโนมัติ

1.3.4 ระบบสามารถแสดงภาพที่เกิดขึ้นบนมือถือเมื่อเจ้าของรถต้องการดูได้และผู้ใช้ระบบสามารถดูภาพจาก SD Card ได้

1.4 ผลที่คาดว่าจะได้รับ

- เพื่อเรียนรู้การอัพโหลดภาพที่แสดงถึงความผิดปกติที่เกิดขึ้นไปที่เซิร์ฟเวอร์
- เพื่อเรียนรู้ และเข้าใจหลักการทำงานของการส่งถ่ายข้อมูลบนมือถือ
- ได้รับความรู้วิธีการติดต่อและควบคุมทางด้านซอฟต์แวร์ และฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ความปลอดภัยต่อรถยนต์ส่วนบุคคลมีมากขึ้น
- สามารถนำไปประยุกต์ใช้กับระบบความปลอดภัยอื่นๆได้อย่างกว้างขวาง

1.5 อุปกรณ์ที่ต้องใช้

1.5.1 ฮาร์ดแวร์

- ไมโครคอนโทรลเลอร์AVR ATmega128 จำนวน 1 ตัว
- Camera Module จำนวน 1 ตัว
- GSM Module จำนวน 1 ตัว
- PIR Sensor จำนวน 1 ตัว
- จอแสดงผล LCD และ สวิตซ์ควบคุม จำนวน 1 ตัว
- RTC DS1307 จำนวน 1 ตัว
- โทรศัพท์มือถือ จำนวน 1 เครื่อง
- SD/MMC Card จำนวน 1 ตัว
- คอมพิวเตอร์/โน้ตบุ๊ก จำนวน 1 เครื่อง

1.5.2 ซอฟต์แวร์

- โปรแกรมภาษาซี
- โปรแกรมเซิร์ฟเวอร์
- โปรแกรม AVR Studio 4 สำหรับไมโครคอนโทรลเลอร์ AVR
- โปรแกรม WinAVR C Compiler เป็นตัวคอมไพล์โปรแกรมภาษา C ให้เป็น Hex

1.6 ขั้นตอนการทำงาน

การทำงานของระบบเตือนภัยในรถผ่านมือถือนั้น สามารถแบ่งลำดับขั้นตอนคร่าวๆได้ดังนี้

- ศึกษา วางแผน และทำความเข้าใจเกี่ยวกับฮาร์ดแวร์ที่ใช้ในโครงงาน
- เลือกอุปกรณ์ให้เหมาะสมกับระบบ
- ศึกษาภาษาซี ที่ใช้ในการดำเนินการเขียนโปรแกรม
- ศึกษาติดต่อและควบคุมระหว่างฮาร์ดแวร์และซอฟต์แวร์ผ่านพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เขียนโปรแกรมให้กล้อง CAMERA MODULE สามารถถ่ายภาพอัตโนมัติเมื่อเซนเซอร์ตรวจจับสิ่งผิดปกติ
- เขียนโปรแกรมให้ระบบสามารถส่งข้อความเตือนไปยังมือถือได้โดยอัตโนมัติและอัพโหลดภาพไปเก็บไว้ที่เซิร์ฟเวอร์ได้
- ทดสอบส่วนต่างๆของระบบ แก้ไขข้อผิดพลาดของโปรแกรม และทดสอบในกรณีที่เกิดข้อผิดพลาด

ตารางที่ 1.1 แผนการดำเนินโครงงาน

หัวข้อ	ภาคเรียนที่ 1				ภาคเรียนที่ 2				
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1.กำหนดขอบเขต และปัญหาของโครงการ	↔								
2.ศึกษา วางแผน และทำความเข้าใจเกี่ยวกับฮาร์ดแวร์ที่ใช้ในโครงงาน		↔							
3.เลือกอุปกรณ์ให้เหมาะสมกับระบบ		↔	↔						
4.ศึกษาภาษาซี ที่ใช้ในการดำเนินการเขียนโปรแกรม		↔	↔	↔					
5.ศึกษาติดต่อและควบคุมระหว่างฮาร์ดแวร์และซอฟต์แวร์ผ่านพอร์ตอนุกรม				↔					
6.ศึกษาวิธีการส่ง SMS และการเปิดภาพบนมือถือ			↔	↔	↔				
7.เขียนโปรแกรมให้กล้อง Camera Module สามารถถ่ายภาพอัตโนมัติเมื่อเซนเซอร์ตรวจจับสิ่งผิดปกติ					↔	↔	↔		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1.1 แผนการดำเนินโครงงาน(ต่อ)

หัวข้อ	ภาคเรียนที่ 1				ภาคเรียนที่ 2				
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
8.เขียนโปรแกรมให้ระบบสามารถส่งข้อความเตือนไปยังมือถือได้โดยอัตโนมัติและส่งภาพไปเก็บไว้ที่เซิร์ฟเวอร์ได้			←	—	—	—	—	→	
9.ทดสอบการส่งต่างๆของระบบ							↔		
10.แก้ไขข้อผิดพลาดของโปรแกรมและทดสอบในกรณีที่เกิดข้อผิดพลาด								↔	
11.จัดทำรูปแบบโครงงาน		←	—	—	—	—	—	—	→

1.7 ส่วนประกอบของโครงงาน

เอกสารของโครงงานฉบับนี้ได้แบ่งเนื้อหาทั้งหมดออกเป็น 5 ส่วน โดยในบทที่ 1 จะเป็นการกล่าวแนะนำโครงสร้างของโครงงานทั้งหมดอย่างคร่าวๆ ทั้งในด้านหลักการเบื้องต้น วัตถุประสงค์ของโครงงาน ขอบเขตของโครงงาน รวมไปถึงอุปกรณ์ที่ใช้ในการดำเนินงานเบื้องต้น และขั้นตอนการทำงานในบทต่อๆไปจะเป็นการกล่าวถึงเนื้อหาดังต่อไปนี้

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการออกแบบ และพัฒนาระบบสำหรับโครงงาน

บทที่ 3 กล่าวถึงภาพรวมของระบบทั้งหมดได้แก่การออกแบบระบบ และส่วนต่างๆที่ได้ทำการพัฒนาขึ้นในโครงงาน

บทที่ 4 กล่าวถึงการทดลอง และผลการทดลอง จะทำการอธิบายถึงรายละเอียดการทำงานที่ได้จากการพัฒนาขึ้นมา

บทที่ 5 กล่าวถึงสรุป และวิจารณ์ผลการทดลอง ทั้งปัญหาที่พบ แนวทางการแก้ไข และแนวทางการพัฒนาต่อไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

ในการทำโครงงานนี้ ได้มีการแบ่งส่วนของการทำงานออกเป็น 3 ส่วน ได้แก่ การทำงานทางด้านผู้ใช้ การทำงานของเซิร์ฟเวอร์ และการทำงานในรถ ดังนั้น การอธิบายในบทนี้จึงขอกล่าวทีละหัวข้อ และแบ่งหัวข้อย่อยออกตามการทำงานในแต่ละส่วน ดังนี้

2.1 ภาษาและแอปพลิเคชั่นที่ใช้ในการทำโครงงาน

ภาษาหลักที่ใช้ในการเขียนแอปพลิเคชั่นในโครงงานนี้ใช้ภาษาซี ซึ่งถูกพัฒนาโดยเดนนิส ริดชี (Dennis Ritche) แห่งห้องทดลองเบลล์(Bell Laboratories) ที่เมอร์รีฮิล มลรัฐนิวเจอร์ซี่ โดยเดนนิสได้ใช้หลักการของภาษา บีซีพีแอล(BCPL : Basic Combine Programming Language) ซึ่งพัฒนาขึ้นโดยเคน ทอมสัน(Ken Tomson) การออกแบบและพัฒนาภาษาซีของเดนนิส ริดชี มีจุดมุ่งหมายให้เป็นภาษาสำหรับใช้เขียนโปรแกรมปฏิบัติการระบบยูนิกซ์ และได้ตั้งชื่อว่า ซี(C) เพราะเห็นว่า ซี(C) เป็นตัวอักษรต่อจากบี(B) ของภาษา BCPL ภาษาซีถือว่าเป็นภาษาระดับสูงและภาษาระดับต่ำ ทั้งนี้เพราะ ภาษาซีมีวิธีใช้ข้อมูลและมีโครงสร้างการควบคุมการทำงานของโปรแกรมเป็นอย่างเดียวกับภาษาของโปรแกรมระดับสูงอื่นๆ จึงถือว่าเป็นภาษาระดับสูง ในด้านที่ถือว่าภาษาซีเป็นภาษาระดับต่ำ เพราะภาษาซีมีวิธีการเข้าถึงในระดับต่ำที่สุดของฮาร์ดแวร์ ความสามารถทั้งสองด้านของภาษานี้เป็นสิ่งที่เกื้อหนุนซึ่งกันและกัน ความสามารถระดับต่ำทำให้ภาษาซีสามารถใช้เฉพาะเครื่องได้ และความสามารถระดับสูง ทำให้ภาษาซีเป็นอิสระจากฮาร์ดแวร์ ภาษาซีสามารถสร้างรหัสภาษาเครื่องซึ่งตรงกับชนิดของข้อมูลนั้นได้เอง ทำให้โปรแกรมที่เขียนด้วยภาษาซีที่เขียนบนเครื่องหนึ่ง สามารถนำไปใช้กับอีกเครื่องหนึ่งได้ ประกอบกับการใช้พอยท์เตอร์ในภาษาซี นับได้ว่าเป็นตัวอย่างที่ดีของการเป็นอิสระจากฮาร์ดแวร์ โดยสิ่งที่กล่าวมาข้างต้นทำให้สรุปได้ว่าภาษาซีเหมาะสมแก่การนำมาใช้ในโครงการนี้ ดังนี้

- เป็นภาษาคอมพิวเตอร์ที่มีการพัฒนาขึ้นใช้งาน เพื่อเป็นภาษามาตรฐานที่ไม่ขึ้นกับโปรแกรมจัดระบบงานและไม่ขึ้นกับฮาร์ดแวร์
- เป็นภาษาคอมพิวเตอร์ที่อาศัยหลักการที่เรียกว่า "โปรแกรมโครงสร้าง" จึงเป็นภาษาที่เหมาะกับการพัฒนาโปรแกรมระบบ
- เป็นคอมไพเลอร์ที่มีประสิทธิภาพสูง ให้รหัสออบเจ็กต์สั้น ทำงานได้รวดเร็ว เหมาะกับงานที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการ ความรวดเร็วเป็นสำคัญ

- มีความคล่องตัวคล้ายภาษาแอสเซมบลี ภาษาซีสามารถเขียนแทนภาษาแอสเซมบลีได้ดี ค้นหาที่ผิดหรือ แก้โปรแกรมได้ง่าย ภาษาซีจึงเป็นภาษาระดับสูงที่ทำงานเหมือนภาษาระดับต่ำ
- สามารถนำภาษาซีไปใช้ในการเขียนโปรแกรมประยุกต์ได้หลายระดับ เช่น เขียนโปรแกรมจัดระบบงาน(OS) คอมไพเลอร์ของภาษาอื่น โปรแกรมสื่อสารข้อมูลโปรแกรมจัดฐานข้อมูล โปรแกรมปัญญาประดิษฐ์(AI = Artificial Inteeligent) รวมทั้งโปรแกรมคำนวณงานทางด้านวิทยาศาสตร์และวิศวกรรมศาสตร์ เป็นต้น
- มีความคล่องตัวที่จะประยุกต์เข้ากับงานต่างๆได้เป็นอย่างดี การพัฒนาโปรแกรม เช่น Word Processing, Spreadsheet, Database ฯลฯ มักใช้ภาษาซีเป็นภาษาสำหรับการพัฒนา
- เป็นภาษาที่มีอยู่บนเกือบทุกโปรแกรมจัดระบบงาน มีในเครื่องไมโครคอมพิวเตอร์ตั้งแต่ 8 บิต ไปจนถึง 32 บิต เครื่องมินิคอมพิวเตอร์ และเมนเฟรม
- โปรแกรมที่เขียนขึ้นด้วยภาษาซีสามารถนำไปใช้กับเครื่องคอมพิวเตอร์ที่ใ ช้ซีพียูต่างเบอร์กันได้ หรือกล่าวได้ว่าโปรแกรมมีความยืดหยุ่น(probability) สูง
- เป็นภาษาที่รวมข้อดีเด่น ในเรื่องการพัฒนาจ นทำให้เป็นภาษาที่มีผู้สนใจมากมาย ที่จะเรียนรู้

หลักการ ของภาษา และวิธีการเขียนโปรแกรม ตลอดจนการพัฒนางานบนภาษานี้

2.2 ฮาร์ดแวร์ที่ใช้ในการทำโครงงาน

2.2.1 ทางด้านผู้ใช้

สามารถใช้เครื่องคอมพิวเตอร์ในการดูภาพจาก SD Card และใช้มือถือในการรับ SMS ที่ส่งมาเตือนว่ามีสิ่งผิดปกติเกิดขึ้น และสามารถใช้มือถือเข้าไปดูภาพ

2.2.2 ทางด้านเซิร์ฟเวอร์

ใช้ GSM Module เป็นตัวส่งภาพขึ้นเซิร์ฟเวอร์ โดยใช้ FTP Server เพื่อรองรับการทำงานจากในรถยนต์ที่ต้องมีการอัพโหลดภาพขึ้นไปเก็บที่เซิร์ฟเวอร์

2.2.3 ระบบภายในรถยนต์

ใช้ Embedded System ในการประมลผลโปรแกรมที่เขียนขึ้นมาเพื่อทำให้กล้อง Camera

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Module สามารถถ่ายภาพอัตโนมัติเมื่อเซนเซอร์ตรวจจับสิ่งผิดปกติ จากนั้นใช้ GSM Module ส่งข้อความเตือนไปยังมือถือได้โดยอัตโนมัติและส่งภาพไปเก็บไว้ที่เซิร์ฟเวอร์

Embedded System เป็นระบบอิเล็กทรอนิกส์ที่ใช้สำหรับงานควบคุมรวมถึงการแสดงผลการทำงานต่าง ๆ โดยใช้เป็นส่วนหนึ่งของระบบ และอุปกรณ์ควบคุม เรียกว่า ระบบแบบฝังตัว เนื่องจากระบบเหล่านี้เป็นส่วนหนึ่งของระบบใหญ่ โดยระบบแบบฝังตัว(Embedded System) นี้แม้ไม่ใช่เครื่องคอมพิวเตอร์ แต่ก็มีระบบคอมพิวเตอร์อยู่ภายใน อาจจะเป็นเพียงไมโครโปรเซสเซอร์(Microprocessor) หรือชิพ(chip) ธรรมดาหรือโปรเซสเซอร์(Processor) ที่ประกอบด้วยชิพที่มีวงจรซับซ้อน โดยจะมีหลักการทำงาน คือ มีสัญญาณข้อมูลเข้า(Input) จากอุปกรณ์เซนเซอร์(Sensor) เข้าสู่ระบบ และมีสัญญาณผลลัพธ์(Output) ของระบบไปควบคุมบังคับสวิตซ์เครื่องควบคุมต่าง ๆ โดยที่ประเภทของ Embedded System แบ่งออกเป็น 2 ประเภท คือ แบบไมโครโปรเซสเซอร์เดี่ยวเป็นระบบซึ่งใช้อยู่ในอุปกรณ์ขนาดเล็ก เช่น เครื่องตรวจจับต่าง ๆ และแบบไมโครโปรเซสเซอร์ หลายตัวรวมกันในวงจร ซึ่งเป็นระบบที่ใช้อยู่ในอุปกรณ์ควบคุมที่ซับซ้อน เช่น อุปกรณ์ควบคุมต่างๆ

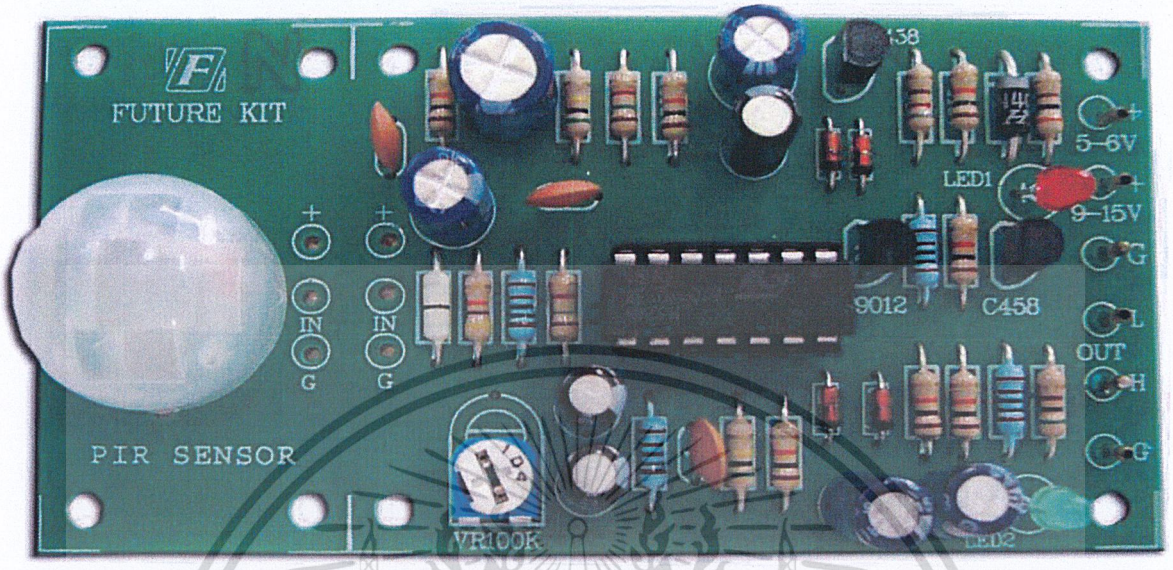
2.3 การตรวจจับสิ่งแปลกปลอม

ในส่วนนี้จะใช้ PIR Sensor เพื่อตรวจจับความเคลื่อนไหว

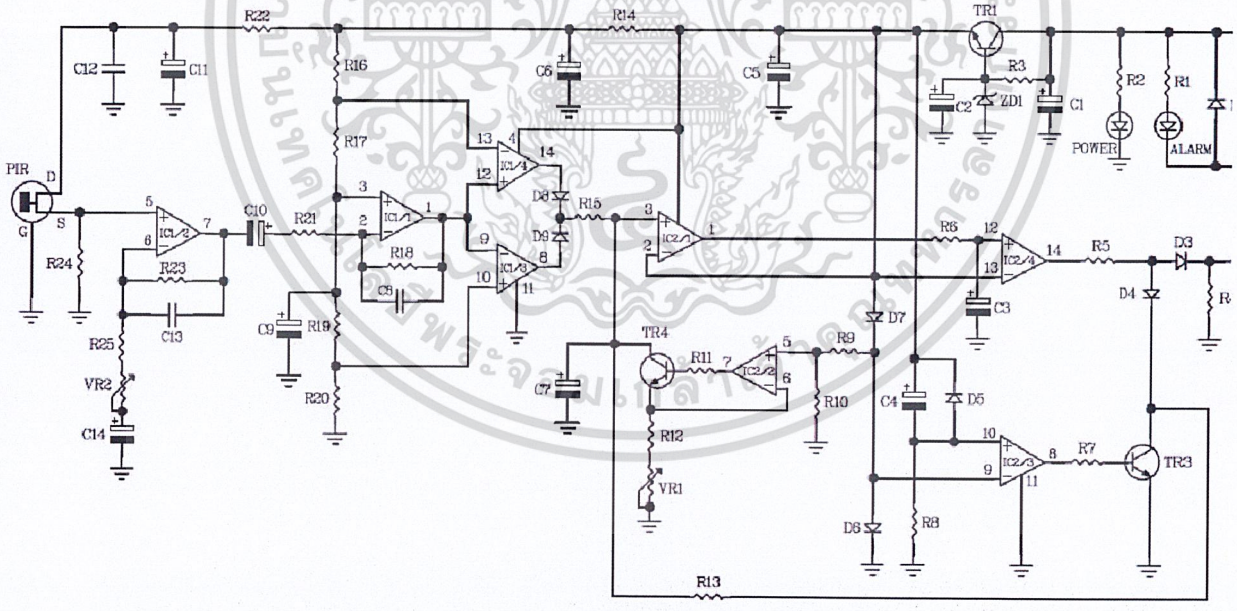
2.3.1 การใช้งานเบื้องต้นของ PIR Sensor

เป็นเซนเซอร์เพื่อตรวจจับความเคลื่อนไหว โดยมีการใช้งานเบื้องต้น คือใช้สายไฟแค่ 3 เส้น คือ เส้นสีแดง +9โวลต์, เส้นสีน้ำตาล คือ สายGround และเส้นสีดำคือ Alarm Output โดย PIR Sensor จะเป็น Open Collector คือ เมื่อมีการจับการเคลื่อนไหวก็จะปล่อยไฟที่ขา Alarm ลง Ground เพราะฉะนั้นการใช้งาน จะต้องต่อ R pullup ไว้ เมื่อมี Alarm ไฟก็จะลงGroundไฟ และเซนเซอร์ตัวนี้ ใช้งานที่ 9โวลต์ ถึง 12โวลต์ จะทำให้เซนเซอร์มีความเสถียรมากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 ส่วนด้านหน้าของ PIR Sensor



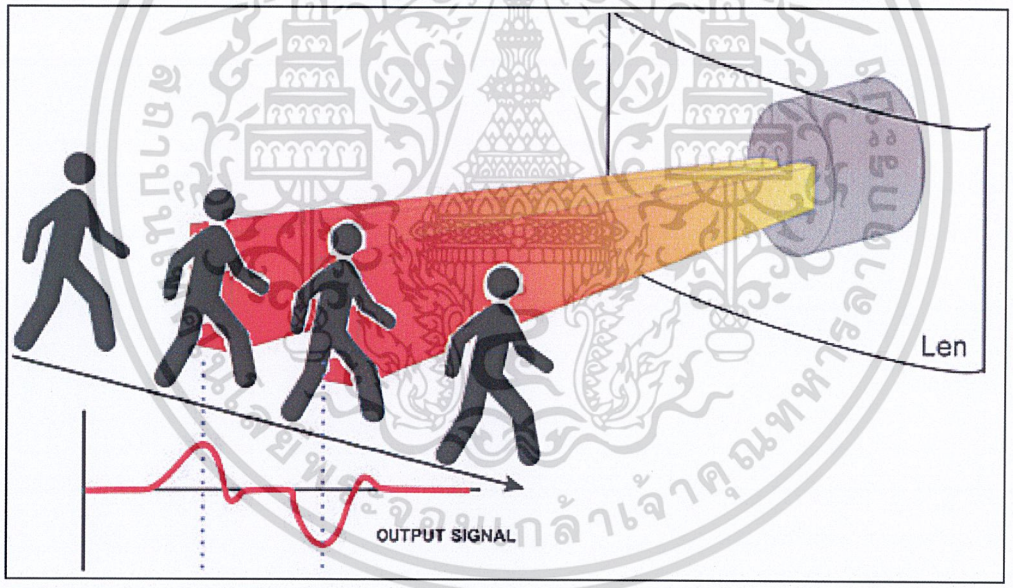
รูปที่ 2.2 วงจรของ PIR Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 หลักการทำงานของ PIR Sensor

PIR Sensor เป็น อุปกรณ์ Sensor ชนิดหนึ่งที่ใช้ตรวจจับคลื่นรังสี Infrared ที่แพร่จากมนุษย์ หรือ สัตว์ ที่มีการเคลื่อนไหว ทำให้มีการนำเอา PIR มาประยุกต์ใช้งานกันเป็นอย่างมากใช้เพื่อตรวจจับการเคลื่อนไหวของสิ่งมีชีวิต หรือ ตรวจจับการบุกรุกในงานรักษาความปลอดภัย

ภายใน PIR Sensor จะมีอุปกรณ์ตรวจจับรังสี Infrared อยู่ 2 ชุดด้วยกันดังรูปที่ 2.3 เมื่อมีคน หรือ สัตว์ที่มีความอบอุ่นในร่างกายเคลื่อนที่ผ่านเข้ามาในพื้นที่โซนที่ PIR สามารถตรวจจับคลื่นรังสี Infrared ที่แพร่ออกมาจากสิ่งมีชีวิตได้ PIR จะเปลี่ยนคลื่นรังสี Infrared ให้กลายเป็นกระแสไฟฟ้าดังรูปที่ 2.3 จะเห็นว่าเมื่อมีสิ่งมีชีวิตเคลื่อนที่ผ่านอุปกรณ์ตรวจจับรังสี Infrared ตัวที่ 1 จะได้สัญญาณ Output ออกมาสูงกว่าแรงดันปรกติ และ เมื่อสิ่งมีชีวิตเคลื่อนที่ผ่าน อุปกรณ์ตรวจจับรังสี Infrared ตัวที่ 2 จะได้แรงดัน Output ต่ำกว่าค่าแรงดันปรกติ



รูปที่ 2.3 การทำงานของ PIR Sensor [1]

2.3.3 หลักในการเลือกใช้เซนเซอร์ให้เหมาะกับงาน

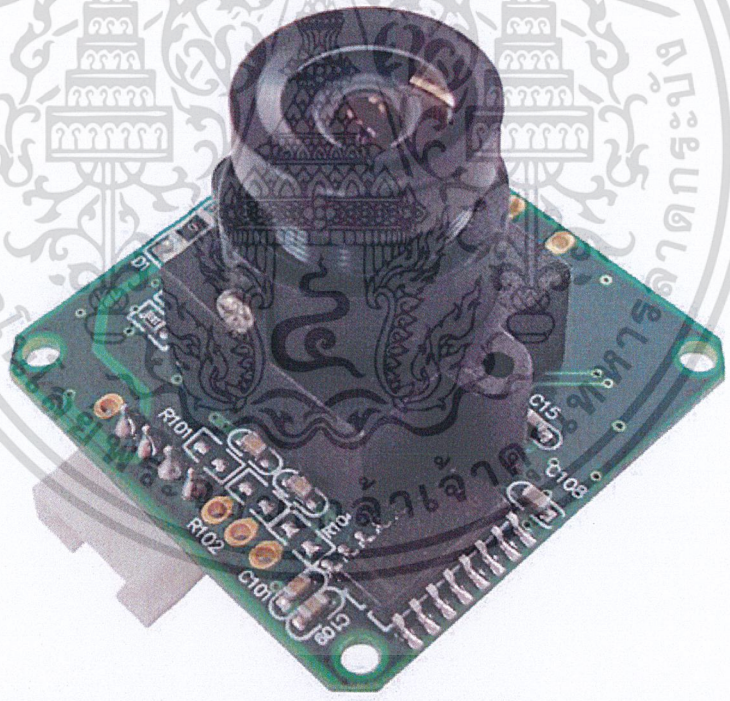
เนื่องจากเซนเซอร์นั้นมีหลายชนิดด้วยกัน ในการเลือกใช้จึงจำเป็นต้องเรียนรู้รายละเอียดคุณสมบัติบางอย่างของเซนเซอร์ตลอดลักษณะการใช้งาน เพื่อที่จะเลือกใช้งานได้อย่างถูกต้องเหมาะสมมากที่สุด ดังนั้นในการเลือกใช้จึงต้องคำนึงถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ขนาดของวัตถุที่ตรวจจับ
- ชนิดของวัตถุที่ตรวจจับ
- ความแม่นยำของตำแหน่งที่ตรวจจับ
- ความเร็วในการเคลื่อนที่ของวัตถุที่ตรวจจับ
- สภาพแวดล้อมที่ใช้เซนเซอร์

2.4 การจับภาพจากกล้อง

การจับภาพจากกล้องโดยการเขียนโปรแกรมภาษาซี ลงในไมโครคอนโทรลเลอร์ AVR ATmega128 เพื่อให้ไมโครคอนโทรลเลอร์ AVR ATmega128 ส่งคำสั่งให้กล้องทำงานเมื่อ PIR Sensor ตรวจจับถึงสิ่งผิดปกติ โดยในโครงงานนี้ได้ใช้กล้องชนิด UCAM Serial JPEG Camera Module



รูปที่ 2.4 กล้องชนิด UCAM Serial JPEG Camera Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 คุณสมบัติของกล้องชนิด UCAM Serial JPEG Camera Module

ในโครงงานนี้ได้เลือกใช้กล้องชนิด UCAM Serial JPEG Camera Module เนื่องมาจากว่ากล้องชนิดนี้มีข้อดีและคุณสมบัติหลายด้านที่เหมาะกับโครงงาน ดังนี้คือ

Serial JPEG Camera Module คือ โมดูลกล้องที่ใช้งานผ่านทาง Serial Interface โดยผู้ใช้งานสามารถกำหนดการทำงาน ต่างๆของโมดูลผ่านทาง Serial Protocol เช่นสั่งการให้กล้องจับภาพในแบบ low resolution(160x120, 80x60) สำหรับ quick viewing หรือ high resolution(640x480, 320x240) JPEG สำหรับ storage เป็นต้น ซึ่งการปรับขนาดภาพนี้เป็นสิ่งสำคัญมากในการทำงานของโครงงานนี้ เพราะเนื่องมากจากว่าโครงงานนี้มีการส่งรูปภาพขึ้นเซิร์ฟเวอร์ ดังนั้นการกำหนดขนาดรูปภาพในการส่งจึงเป็นส่วนหนึ่งที่สำคัญในการกำหนดความเร็วในการส่งภาพขึ้น เซิร์ฟเวอร์

ซึ่งกล้องชนิด UCAM Serial JPEG Camera Module ตัวโมดูลมีส่วนประกอบในแต่ละส่วน คือ Omni Vision CMOS VGA color sensor(OV7640) ทำงานร่วมกับ JPEG Compression chip(OV528) และอีกส่วนคือ On-board serial interface จากการทำงานของโมดูลซึ่งรับ/ส่งข้อมูลผ่านทาง serial interface ทำให้สามารถนำมาใช้งานในด้านต่างๆได้อย่างสะดวก เช่น การนำไปต่อใช้งานได้โดยตรงกับ Microcontroller UART(TTL Level) หรือใช้งานกับ PC ผ่านทาง COM port (ต่อใช้งานผ่าน MAX232 หรือ USB to Serial)

2.4.2 การทำงานของกล้อง UCAM Serial JPEG Camera Module

จะประกอบด้วยคำสั่งเพื่อการใช้งาน เช่น การใช้คำสั่ง สั่งงานเพื่อการติดต่อกับตัวกล้อง, การกำหนดค่าความละเอียดของภาพ, การอ่านเฟรมภาพ เป็นต้น คำสั่งต่างๆ ที่ใช้ควบคุมการทำงานของตัวกล้อง ดูได้จากตารางที่ 2.1 ดังนี้

ตารางที่ 2.1 คำสั่งของกล้อง UCAM Serial JPEG Camera Module

Command	Id Number	Parameter1	Parameter2	Parameter3	Parameter4
Initial	AA01h	00h	Code Type	Preview Resolution	JPEG Resolution
Get Picture	AA04h	Picture Type	00h	00h	00h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 คำสั่งของกล้อง UCAM Serial JPEG Camera Module(ต่อ)

Command	Id Number	Parameter1	Parameter2	Parameter3	Parameter4
Snapshot	AA05h	Snapshot Type	Skip Frame Low Byte	Skip Frame High Byte	00h
Set Package Size	AA06h	08h	Package Size Low Byte	Package Size High Byte	00h
Set Baudrate	AA07h	1st Divider	2nd Divider	00h	00h
Reset	AA08h	Reset Type	00h	00h	xxh*
Power Off	AA09h	00h	00h	00h	00h
Data	AA0Ah	Data Type	Length Byte 0	Length Byte 1	Length Byte 2
SYNC	AA0Dh	00h	00h	00h	00h
ACK	AA0Eh	Command ID	ACK counter	00h / Package ID Byte 0	00h / Package ID Byte 1
NAK	AA0Fh	00h	NAK counter	Error number	00h

รูปแบบ Package ข้อมูลจะประกอบด้วยข้อมูลจำนวน 6 ไบต์ ข้อมูลไบต์แรกเป็นคำสั่ง ไบต์ที่สองจะเป็น ID Number จะใช้คู่กับไบต์แรก(ดังรูปตารางที่ 2.1) และข้อมูลอีก 4 ไบต์เป็น Parameter ไบต์ จะเป็น Parameter เพื่อใช้เป็นตัวกำหนด เช่น หากใช้คำสั่งเพื่อการถ่ายภาพ คำสั่งจะต้อง set ขนาดของภาพว่าต้องการถ่ายภาพด้วยขนาดเท่าไหร่ ซึ่งสามารถกำหนดขนาดได้ทั้งหมด 4 ขนาดคือ 80x64, 160x128, 320x240, 640x480 สำหรับตัวกล้องรุ่นนี้มีการสื่อสารแบบ อนุกรม (Serial Interface) สามารถเลือกความเร็วของการสื่อสารได้ คือ อัตราที่ Baud Rate 7200 bps, 14400 bps, 28800 bps, 57600 bps, 115200 bps แต่ตัวกล้องจะถูก set ค่า Default ไว้ที่ Baud Rate 115200 bps ซึ่งอัตราความเร็วสามารถเปลี่ยนแปลงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 หลักการถ่ายภาพของกล้อง UCAM Serial JPEG Camera Module

เริ่มต้นการใช้คำสั่ง Sync(0xAA, 0x0D, 0x00, 0x00, 0x00, 0x00) การ Sync นี้จะวนลูปจนได้ข้อมูลตอบกลับมาจากตัวกล้อง เป็น 0xAA, 0x0E, 0x0D, 0xXX, 0x00, 0x00 ปกติแล้วการวนลูปนั้นจะอยู่ในช่วงไม่เกิน 25 ลูป จะได้ข้อมูลการตอบกลับจากตัวกล้อง โดยจะตั้งค่าลูปไว้ประมาณ 60 ลูป หากครบ 60 ลูปแล้วไม่มีการตอบกลับควรตรวจสอบการทำงานของโปรแกรมเพื่อติดต่อกับตัวกล้องว่าถูกต้องหรือไม่ อาจจะมีการตรวจสอบในเรื่องของการใช้อัตรา BaudRate ว่าใช้ค่าถูกต้องหรือไม่ หลังจาก Sync แล้วมีการตอบกลับจากตัวกล้องแล้ว คำสั่งต่อไปจะเป็นคำสั่งเพื่อต้องการบอกว่าต้องการขนาดภาพเท่าไหร่ (0xAA, 0x01 , 0x00 ,0xXX, 0xXX, 0xXX) สำหรับพารามิเตอร์ทั้งสามไบต์สามารถดูได้จากตารางที่ 2.2 ดังนี้

ตารางที่ 2.2 การกำหนดขนาดภาพ

Resolution Picture Size	Parameter
80 x 64	01
160x128	03
320x240	05
640x480	07

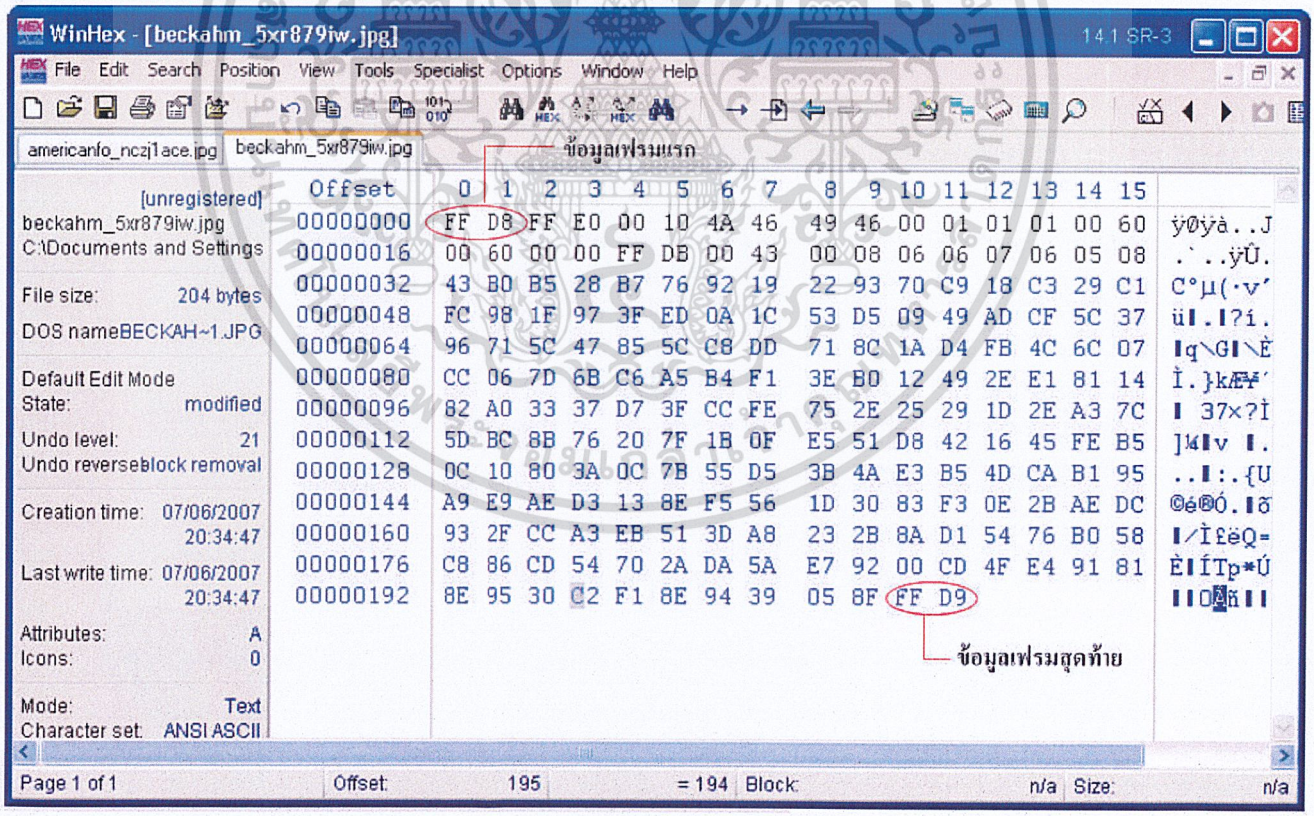
การส่งคำสั่งเพื่อติดต่อกล้อง จะต้องได้รับค่าตอบกลับ(Response) โดยที่ไบต์แรกจะเป็น 0xAA และไบต์ที่สองเป็น 0x0E เสมอสำหรับไบต์ที่สามจะเหมือนคำสั่งไบต์ที่สอง ซึ่งจะเป็นเช่นนี้ทุกๆ การใช้คำสั่ง หลังจากที่มีการบอกขนาดของภาพแล้ว ส่งที่จะต้องส่งเป็นคำสั่งต่อไปคือ คำสั่งเพื่อบอกว่าต้องการถ่ายภาพ(Snap Shot) หรือ การ Preview ภาพ รูปแบบของคำสั่งของการถ่ายภาพ (Snap Shot) คือ 0xAA, 0x05, 0x00, 0x00, 0x00, 0x00 ต่อด้วยคำสั่ง(Get Picture size) 0xAA, 0x04, 0x01, 0x00, 0x00, 0x00 เพื่อต้องการทราบขนาดของรูปว่าขนาดของไฟล์รูปที่ถ่ายนั้นมีขนาดกี่ไบต์ ข้อมูลที่ตอบกลับมาจะได้เป็นขนาดของรูป ซึ่งในการรับข้อมูลแต่ละเฟรมนั้นจะรับได้ครั้งละ 1 เฟรม โดยแต่ละเฟรมจะมีขนาด 512 ไบต์ แต่หากข้อมูลที่มีขนาดเศษที่ไม่ครบ 512 ไบต์ ก็จะทำการรับของจำนวนไบต์ที่เหลือ ยกตัวอย่างเช่น ข้อมูล 1 ภาพมีขนาด 2052 ไบต์ หากเรานำมาหารด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

512 ไบต์ จะได้ทั้งหมด 4 เฟรม และมีเศษ 4 ไบต์ คือจะต้องมีการรับทั้งหมด 5 ครั้ง เป็น 512 ไบต์หนึ่งครั้งและเศษข้อมูล 4 ไบต์หนึ่งครั้ง จึงจะได้ข้อมูลเป็นหนึ่งภาพ หลังจากที่รับเฟรมภาพทั้งหมดแล้วคำสั่งสุดท้ายคือ คำสั่ง ACK(0xAA, 0x0E, 0x00, 0x00, 0xF0, 0xF0) เพื่อบ่งบอกการจบขบวนการทำงานเพื่อให้ภาพหนึ่งภาพ

2.4.4 การเขียนข้อมูลภาพลงบน SD Card

ในการจัดเก็บไฟล์ข้อมูลจะต้องมีการตรวจสอบข้อมูลก่อน โดยข้อมูลในเฟรมภาพในไบต์แรกจะขึ้นต้นด้วย 0xFF 0xD8 และข้อมูลเฟรมสุดท้ายแม้จะเป็นเศษก็ตามจะต้องจบด้วยข้อมูลไบต์ที่เป็น 0xFF 0xD9 ซึ่งถ้าต้องการทดสอบไฟล์นามสกุล JPEG สามารถที่จะเปิดไฟล์นั้นด้วยโปรแกรม WinHex Editer จะสังเกตได้ว่าข้อมูลสองไบต์แรกของภาพจะเป็น 0xFF 0xD8 และสองไบต์สุดท้ายจะเป็น 0xFF 0xD9 เสมอ แต่อาจจะมีข้อมูลตามด้วยสองไบต์หลัง 0xD9 จะมีข้อมูลตามมาสองหรือสามไบต์อาจจะเป็น 0x0A, 0x0D



รูปที่ 2.5 เนื้อไฟล์ข้อมูลของภาพหนึ่งภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเขียนข้อมูลภาพลงบน SD Card นั้นจะเป็นการเขียนแบบ FAT System ซึ่งหากมีการเขียนไฟล์เสร็จ สามารถนำไปเปิดกับเครื่องคอมพิวเตอร์หรืออุปกรณ์อื่นๆ ที่รองรับมาตรฐาน FAT File เช่น PDA, Mobile เป็นต้น และต้องมีโปรแกรมแสดงภาพจากไฟล์เช่น Photo Shop, ADCSee Viewer ก็จะแสดงภาพที่ถ่ายได้จากตัวกล้อง

2.5 การส่งการเตือนไปให้แก่ผู้ใช้โดยการส่ง SMS

ในการส่งสัญญาณเตือนไปให้ผู้ใช้ทราบถึงสิ่งผิดปกตินั้น ใช้วิธีการส่ง SMS โดยจะเขียนคำสั่ง AT ไปที่ GSM Module ซึ่งเป็นตัวที่ทำการส่ง SMS ออกไปให้แก่สถานีฐานเข้าไปสู่ระบบของผู้ให้บริการ

AT Command คือ เป็นชุดคำสั่งมาตรฐานที่ใช้ติดต่อสื่อสารกับโมเด็ม โดยส่วนมากมักใช้ในการสื่อสารกับอุปกรณ์สื่อสารต่างๆ เช่น ชุดคำสั่ง AT ของโมเด็มในเครื่องคอมพิวเตอร์, ชุดคำสั่ง AT ของโมเด็มในมือถือ ซึ่งชุดคำสั่ง AT เหล่านี้ เป็นชุดของสตริงที่ต้องขึ้นต้นด้วยตัวอักษร AT เสมอ แล้วเว้นวรรค 1 ตัวอักษร แล้วตามด้วยคำสั่งที่เป็นสตริงอีกเช่นกัน โดยคำสั่งตามหลังนี้จะแตกต่างกันไปตามชนิดของโมเด็ม และตามยี่ห้อของโมเด็มนั้นๆ เพราะแต่ละผู้ผลิตจะใช้ชุดคำสั่งไม่เหมือนกัน เช่น ในมือถือยี่ห้อโนเกีย(NOKIA) กับซีเมนส์(SIEMENS) จะมีชุดคำสั่งที่ไม่เหมือนกัน(ไม่เหมือนในส่วนของคำสั่งที่พิเศษที่มีเฉพาะในแต่ละรุ่นแต่ละยี่ห้อเท่านั้น) แต่ว่าคำสั่งพื้นฐานของระบบ GSM จะเหมือนกัน เพราะในระบบโทรศัพท์มือถือระบบ GSM นั้นมีข้อกำหนดของการทำงานของโมเด็มที่เหมือนๆกัน ดังนั้นคำสั่งให้ส่ง SMS ออกไปนั้นจึงใช้ได้กับทุกเครื่องทุกยี่ห้อ ของเพียงแต่เป็นระบบ GSM เหมือนกัน

2.5.1 ตัวอย่างคำสั่ง AT-Command

คำสั่งที่ใช้ในการควบคุมสายโทรศัพท์ ดังตัวอย่าง

- ATD ใช้ในกรณีที่เป็น คำสั่งโทรออก
- ATA ใช้ในกรณีที่เป็น คำสั่งตอบรับการโทรเข้า
- ATH ใช้ในกรณีที่เป็น คำสั่งวางสาย

คำสั่งเกี่ยวกับการใช้ SMS ดังตัวอย่าง

- AT + CMGS ใช้ในกรณีที่เป็น ข้อความที่ส่ง
- AT + CMSS ใช้ในกรณีที่เป็น ข้อมูลจากแหล่งข้อมูล
- AT + CMGL ใช้ในกรณีที่เป็น ข้อความจากรายการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

- AT + CMGR ใช้ในกรณีที่เป็น การอ่านข้อความ

คำสั่งของโทรศัพท์

- AT + CGMI ใช้ในกรณีที่เป็น ชื่อของผู้ผลิต

- AT + CGMM ใช้ในกรณีที่เป็น หมายเลขรุ่น

- AT + CGSN ใช้ในกรณีที่เป็น หมายเลข IMEI(International Mobile Equipment Identity)

- AT + CGMR ใช้ในกรณีที่เป็น รุ่นซอฟต์แวร์(AT + CGMR)

2.5.2 หลักการส่ง SMS ของ GSM Module

ในการส่งข้อความ SMS นั้นจะใช้คำสั่ง “AT+CMGS” ในการสั่งงาน โดยในกรณีที่ใช้ Text Mode นั้นให้ใช้รูปแบบคำสั่งเป็น “AT+CMGS=”+เบอร์ผู้รับ” โดยเบอร์ของผู้รับต้องใส่รหัสประเทศนำหน้าแทนศูนย์ด้วยเสมอ ซึ่งในกรณีที่เป็นประเทศไทยจะใช้รหัสประเทศเป็น “66” ดังนั้นถ้าต้องการส่งข้อความ SMSให้กับเบอร์ที่ใช้งานอยู่ในประเทศไทย เช่น 081-1234567 ก็จะต้องกำหนดหมายเลขของเบอร์ผู้รับปลายทางเป็น 6681-1234567 แทน ซึ่งในกรณีนี้จะได้รหัสเบอร์ผู้รับข้อความเป็น “+66811234567” ซึ่งเมื่อโมดูล SIM300CZ ได้รับคำสั่ง “AT+CMGS” เรียบร้อยแล้วมันจะตอบรับด้วยการส่งเครื่องหมาย “>”กลับมาบอก ซึ่งหลังจากนี้เป็นต้นไปผู้ใช้ก็สามารถจะทำการพิมพ์ข้อความต่างๆที่ต้องการจะส่งให้กับโมดูลได้ทันที โดยให้ปิดท้ายข้อความด้วยรหัส “Ctrl+Z” ตามด้วย “Enter” เช่นถ้าต้องการส่งข้อความ SMS ให้กับหมายเลข 0811234567 ด้วยข้อความ “Hello Project Test SMS” จะเป็นดังนี้

AT+CMGS="+66811234567"<Ent>

> Hello Project Test SMS<Ctrl+Z><Ent>

+CMGS: 6

OK

2.5.3 คุณสมบัติของ GSM Module

- รองรับความถี่ GSM/GPRS 900/1800/1900 เมกะเฮิรตซ์
- รองรับ GPRS Multi-Slot Class10 และ GPRS Mobile Station Class B
- รองรับมาตรฐานคำสั่ง AT Command
- รองรับ SIM Applications Toolkit
- ทำงานที่ย่านแรงดัน 3.4โวลต์ ถึง 4.5โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

117550

- รองรับการเชื่อมต่อภายนอก

1) ใช้ได้กับ SIM 3โวลต์ และ 1.8โวลต์

2) มีวงจร Analog Audio(MIC & Speaker) จำนวน 2 ชุด

3) รองรับ 5x5 Keypad Interface & SPI LCD Interface

4) มีระบบ RTC พร้อมวงจร Backup

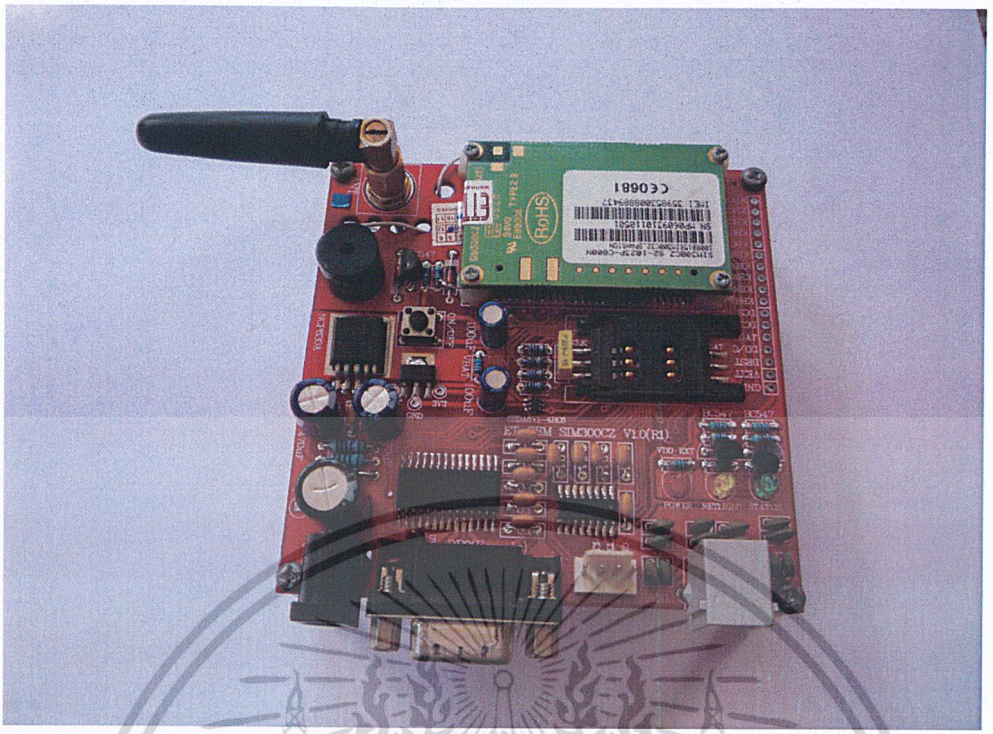
5) มีขั้วต่อเสาอากาศภายนอกแบบ Connector และจุดเชื่อมต่อแบบ PAD

6) มีระบบ Battery Charge ในตัว

2.5.4 ข้อดีของการใช้ GSM Module

เป็นโมดูลสื่อสารระบบ GSM/GPRS ขนาดเล็ก รองรับระบบสื่อสาร GSM ความถี่ 900/1800/1900เมกะเฮิรตซ์ โดยสั่งงานผ่านทางพอร์ตสื่อสารอนุกรมRS-232 ด้วยชุดคำสั่ง AT Command สามารถประยุกต์ใช้งานได้มากมายหลายรูปแบบ ไม่ว่าจะเป็นการรับส่งสัญญาณแบบ Voice, SMS, Data, FAX และยังรวมถึงการสื่อสารด้วย Protocol TCP/IP ด้วย ถึงแม้ว่าโมดูล SIM300CZ จะมีวงจร และ Firmware บรรจุไว้ภายในตัวเป็นที่เรียบร้อยแล้วก็ยังไม่สามารถนำไปใช้งานได้โดยตรงทันที เนื่องจากในการใช้งานจริงๆนั้น จำเป็นต้องออกแบบวงจรรอบนอกที่จำเป็นมาเชื่อมต่อกับขาสัญญาณของตัวโมดูลอีกในบางส่วน ไม่ว่าจะเป็นวงจรภาค PowerSupply,วงจรเชื่อมต่อกับ SIM Card รวมไปถึงวงจร Line Driver ของ RS-232 เป็นต้น ดังนั้น จึงมีการจัดสร้างบอร์ดสำหรับเป็นตัวกลางในการเชื่อมต่อระหว่างโมดูล SIM300CZ กับอุปกรณ์ภายนอกเพื่อให้สามารถนำโมดูล GSM ของ SIM300CZ ไปทำการทดลองและศึกษาเรียนรู้การสั่งงานต่างๆได้โดยสะดวก ก่อนที่จะนำเอาโมดูลตัวนี้ไปออกแบบดัดแปลงและประยุกต์ใช้งานในด้านต่างๆได้ต่อไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 GSM Module sim300cz

2.6 การส่งภาพขึ้นเซิร์ฟเวอร์

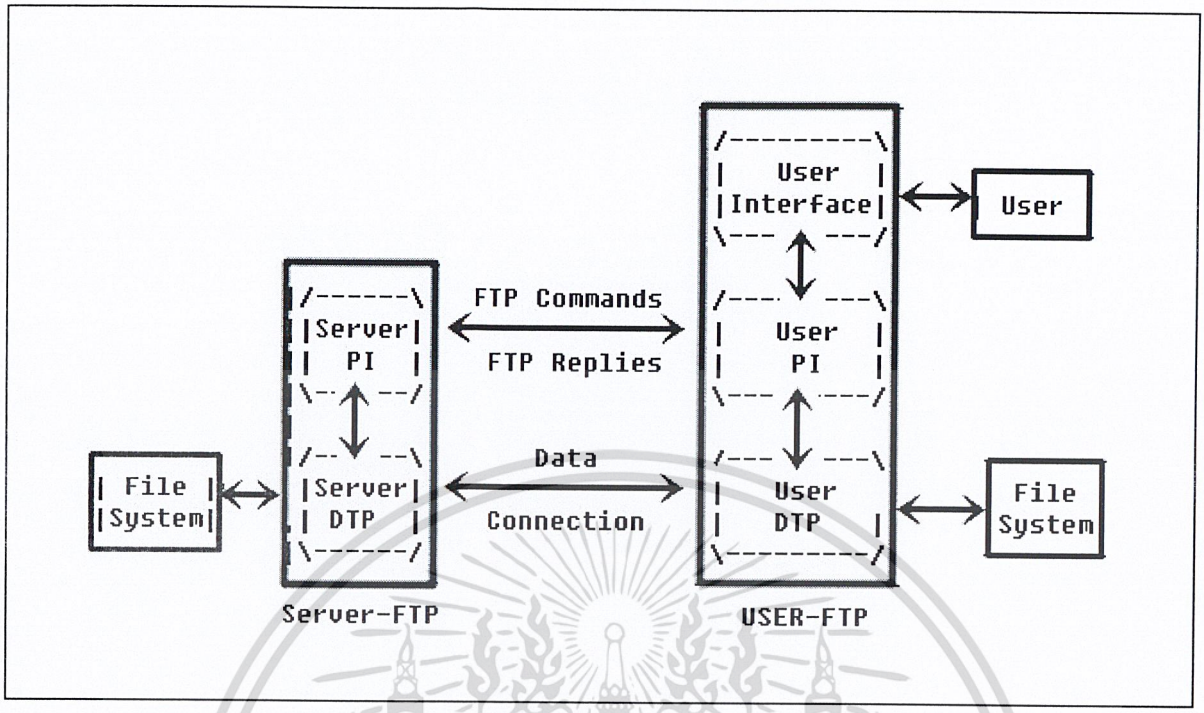
ในการส่งภาพไปที่มือถือนั้นไม่สามารถทำได้โดยตรง ต้องมีการส่งภาพไปเก็บที่เซิร์ฟเวอร์ก่อน แล้วจึงใช้มือถือในการเปิดดูภาพ เหตุผลที่ไม่สามารถส่งภาพไปที่มือถือได้โดยตรง เนื่องมาจากว่า ขีดขำกัดของระบบมือถือที่ใช้ไม่สามารถติดต่อโดยตรงผ่าน IP ได้ เพราะ IP ที่มือถือใช้ในการสื่อสารผ่านเครือข่ายนั้น เป็น IP เสมือน ไม่ใช่ IP จริง ดังนั้นในการทำงานส่วนนี้จึงจำเป็นต้องส่งไฟล์ภาพไปขึ้นวางที่เซิร์ฟเวอร์ โดยโปรโตคอลที่เลือกใช้คือ FTP หรือ File Transfer Protocol ซึ่ง FTP นั้นเป็นโปรโตคอลที่ทำหน้าที่รับส่งไฟล์โดยเฉพาะ

FTP เป็นโปรโตคอลมาตรฐานที่ใช้ในการแลกเปลี่ยนไฟล์ระหว่างคอมพิวเตอร์ บนเครือข่ายอินเตอร์เน็ต ซึ่งเป็นวิธีการที่ง่ายและปลอดภัยที่สุด ซึ่งจุดประสงค์ของ FTP คือ

- เพื่อส่งเสริมการแบ่งปันแฟ้มข้อมูล(โปรแกรมคอมพิวเตอร์ หรือ ข้อมูลต่างๆ)
- เพื่อสนับสนุนการใช้งานคอมพิวเตอร์จากระยะไกลโดยทางอ้อม(ผ่านโปรแกรมคอมพิวเตอร์)
- เพื่อป้องกันผู้ใช้งานจากความหลากหลายในกระบวนการเก็บแฟ้มข้อมูล ระหว่าง คอมพิวเตอร์แม่ข่าย
- เพื่อนถ่ายโอนข้อมูลอย่างมีประสิทธิภาพ และเชื่อถือได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FTP Model

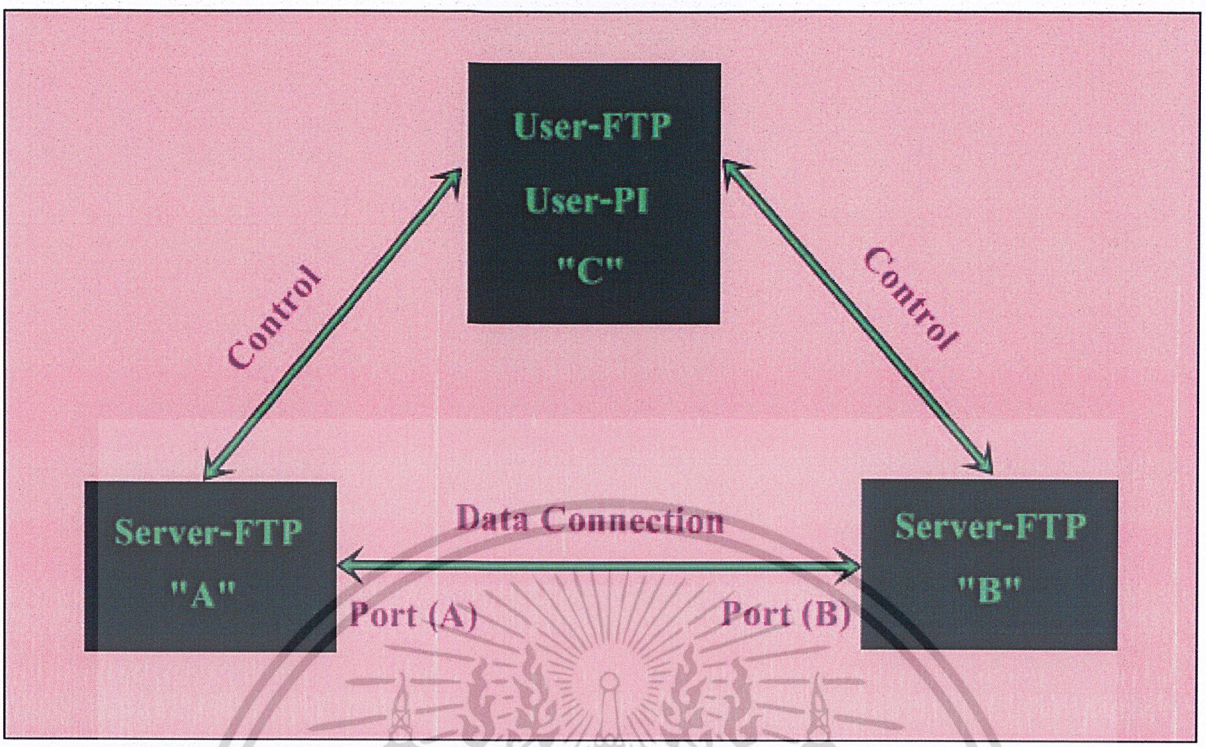


รูปที่ 2.7 FTP Model

การทำงานของ FTP เริ่มต้น จากโปรโตคอลฝั่งผู้ใช้ทำการติดต่อไปยัง Telnet โปรโตคอลในขั้นแรก คำสั่งมาตรฐาน FTP จะถูกส่งจาก USER-PI ไปยัง เซิร์ฟเวอร์ โดยผ่านการติดต่อควบคุม(ผู้ใช้จะทำการสร้างการติดต่อโดยตรงไปยัง Server-FTP และสร้างคำสั่งพื้นฐานด้วยตัวมันเอง โดยผ่านกระบวนการ USER-FTP) การตอบสนองมาตรฐานจะถูกส่งจาก SERVER-PI ไปยัง USER-PI ผ่านการติดต่อควบคุม เพื่อเป็นการตอบสนองคำสั่งที่ได้รับ คำสั่ง FTP เป็นตัวแปรในการส่งข้อมูล USER-DTP จะทำการรับสัญญาณข้อมูลที่ส่งมา และ เซิร์ฟเวอร์ จะเริ่มทำการติดต่อและส่งข้อมูลด้วยตัวแปรตามข้อกำหนด เป็นที่น่าสังเกตว่าพอร์ตข้อมูลไม่จำเป็นต้องเป็นพอร์ตเดียวกันเครื่องแม่ข่ายที่เริ่มทำการส่งคำสั่งผ่านการติดต่อควบคุม แต่ผู้ใช้หรือกระบวนการทำงานของ USER-FTP จะต้องทำงานผ่านพอร์ตข้อมูล และการติดต่อข้อมูลเป็นสิ่งที่จำเป็นสำหรับการส่งและรับข้อมูล

ในทางกลับกัน ผู้ใช้คาดหวังว่าจะทำการถ่ายโอนแฟ้มข้อมูลระหว่างเครื่องแม่ข่ายสองเครื่อง ผู้ใช้ทำการติดตั้งการติดต่อควบคุมระหว่างเซิร์ฟเวอร์ 2 เครื่อง หลังจากนั้นจึงทำการจัดเตรียมการส่งข้อมูลระหว่างกัน ในการทำเช่นนี้ การควบคุมข้อมูลจะถูกส่งผ่านไปยัง USER-PI แต่ข้อมูลจะถูกส่งระหว่าง เซิร์ฟเวอร์ ในกระบวนการถ่ายโอนข้อมูล ดังแสดงในรูปที่ 2.8 เป็นการแสดงการติดต่อระหว่าง เซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 การติดต่อระหว่าง เซิร์ฟเวอร์

ความต้องการโปรโตคอลจะเกิดขึ้นในกระบวนการถ่ายโอนข้อมูลในการติดต่อควบคุม มันเป็นการตอบสนองของผู้ใช้ในการตัดการติดต่อ หลังจากเสร็จสิ้นการใช้งาน FTP Server จะยกเลิกการถ่ายโอนข้อมูล ถ้าการติดต่อควบคุมถูกปิดโดยไม่มีคำสั่ง

ส่วนในเรื่องของการเขียนโปรแกรม ให้ทำการอัพโหลดภาพขึ้นไปไว้ที่เซิร์ฟเวอร์นั้นในโครงงานนี้จะใช้โปรแกรมภาษาซีในการอัพโหลดภาพขึ้น เซิร์ฟเวอร์ ซึ่งมี GSM Module เป็นสื่อกลางในการ อัพโหลดภาพขึ้น เซิร์ฟเวอร์ ซึ่ง GSM Module จะต้องมีการเชื่อมต่อ GPRS ตลอดเวลา

2.7 GSM

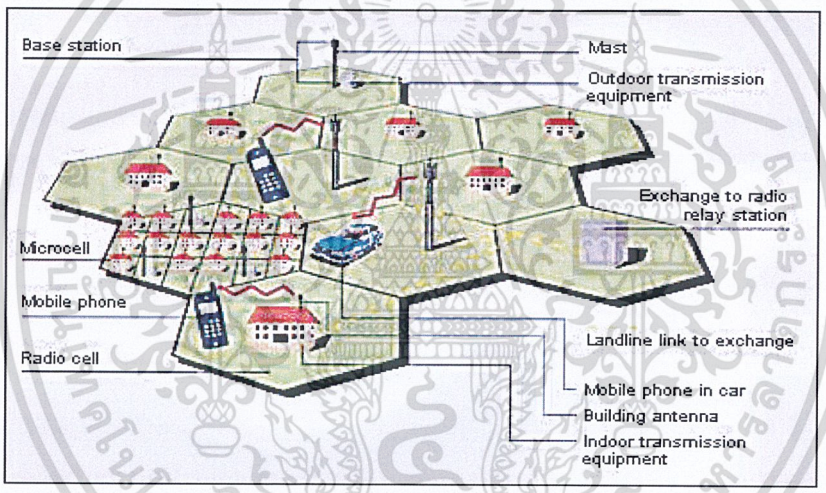
ในโครงการนี้เราใช้ระบบเครือข่าย GSM เพื่อใช้ในการอัพโหลดภาพขึ้นเซิร์ฟเวอร์ และใช้ในการส่ง SMS เตือนไปให้แก่ผู้ใช้ ดังนั้นเราจึงควรทราบถึงพื้นฐานของระบบเครือข่าย GSM พอสังเขป

GSM นั้นเป็นระบบเครือข่ายโทรศัพท์เคลื่อนที่แบบเซลลูล่าร์ชนิด ดิจิตอลเซลลูล่าร์(Digital Cellular) โดยย่อมาจาก Global System for Mobile ซึ่ง GSM ใช้เทคโนโลยีดิจิตอลสำหรับช่องสัญญาณควบคุมและสัญญาณเสียงแบบ TDMA ซึ่งแตกต่างจากเทคโนโลยีโทรศัพท์มือถือก่อนหน้านั้น จึงถือว่าเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โทรศัพท์มือถือในยุคที่สอง หรือ 2G มีพัฒนาการมาจากโทรศัพ์แบบเซลลูลาร์ จนกลายมาเป็น GSM ที่มีความเสถียรมากที่สุด ระบบ GSM ที่อยู่ในยุคที่สองเป็นช่องสัญญาณเสียงเป็นหลัก โดยใช้แถบขาเข้าเพียงประมาณ 9 กิโลบิตต่อหนึ่งช่องเสียง ความเร็วขนาด 9 กิโลบิตต่อ วินาทีคงไม่พอเพียงกับการเชื่อมต่อเครือข่ายของอุปกรณ์มือถือ ซึ่งกำลังเน้นการประยุกต์ที่ต้องการความเข้าใจในการรับค่ามากขึ้น

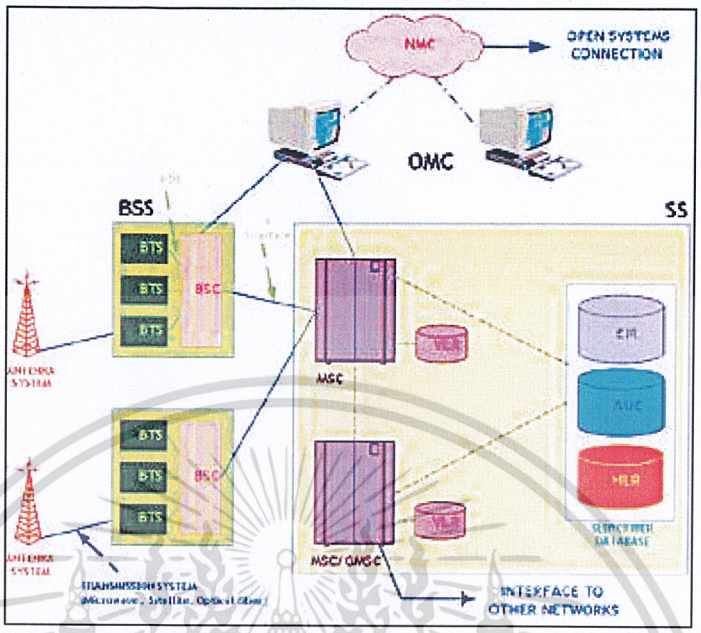
เมื่อเป็นเช่นนี้จึงต้องพัฒนาต่อโดยพัฒนาระบบ WCDMA(Wideband Code Division Multiple Access) ซึ่งเป็นเทคนิคที่ทำให้ขยายช่องสัญญาณได้ มากขึ้นและได้แถบกว้างขึ้น การพัฒนาจาก GSM ที่ใช้เทคนิค TDMA มาเป็น WCDMA เป็นมาตรฐานที่สำคัญของการพัฒนาระบบ GSM ขณะเดียวกัน ในสหรัฐอเมริกาก็พยายามพัฒนาระบบ 2G ซึ่งเป็น TDMA ขยายต่อโดยใช้ชื่อเทคโนโลยีที่ EDGE-Enhance Data Rate for GSM ซึ่งก็เป็นการพัฒนา เข้าสู่ 3G



รูปที่ 2.9 ระบบโทรศัพท์เคลื่อนที่ GSM [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1 โครงสร้างของระบบ GSM



รูปที่ 2.10 โครงสร้างของระบบ GSM [3]

โครงสร้างของระบบ GSM จะประกอบไปด้วย 4 ส่วนใหญ่ ๆ ดังนี้

2.7.1.1 เครื่องโทรศัพท์เคลื่อนที่(Mobile Station, Mobile Equipment)

- เครื่องโทรศัพท์เคลื่อนที่(MS, ME)

International Mobile Equipment Identity(IMEI) คือ หมายเลขประจำเครื่อง Mobile phone ซึ่งจะมีเลขเดียวในโลกยาว 15 หลัก

TAC คือ Type Approval Code 6 หลัก กำหนดโดยองค์กรกลาง GSM

FAC คือ Final Assembly Code 2 หลัก กำหนดโดยผู้ผลิต Mobile

SNR คือ Serial Number 6 หลัก กาหนดโดยผู้ผลิต Mobile

SPARE คือ ตัวเลขเพื่อเอาไว้ 1 หลัก

เราสามารถตรวจสอบรหัส IMEI ในเครื่องโทรศัพท์ได้โดยการกด *#06#

- ซิมการ์ด(SIM card)

International Mobile Subscriber Identity(IMSI) ย่อมาจาก Subscriber Identity Module เป็นอุปกรณ์ซึ่งใส่ในเครื่องโทรศัพท์เคลื่อนที่เพื่อให้เครื่องสามารถติดต่อกับเครือข่ายได้ ซิมการ์ดไม่เก็บหมายเลข IMSI(International

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mobile Subscriber Identity) ซึ่งเป็นหมายเลขที่ไม่ซ้ำกับ SIM อื่นๆ ทั่วโลก หมายเลขนี้ จะผูกกับหมายเลขโทรศัพท์ของเจ้าของ SIM โดยมีฐานข้อมูลเก็บที่ HLR

MCC คือ Mobile country code 3 หลัก

MNC คือ Mobile Network Code 1-2 หลัก

MSIN คือ Mobile Station Identification Number

MCC ของประเทศไทยคือ 520

2.7.1.2 ระบบสถานีฐาน(BSS)

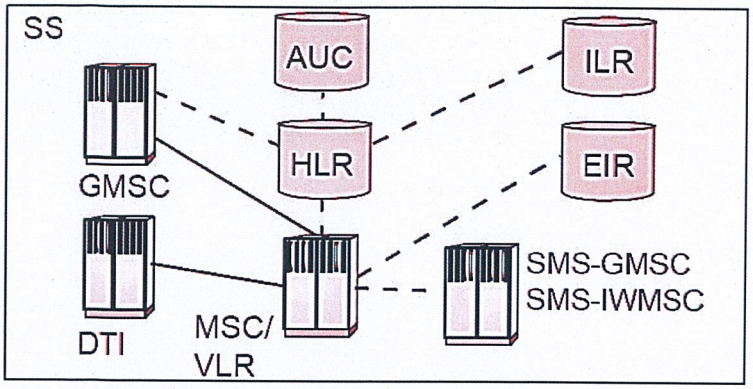
- สถานีฐาน(Base Transceiver Station-BTS) คือ เสาสัญญาณที่ติดตั้งอยู่ตามตึก หรือตามข้างถนน ซึ่งทำหน้าที่เป็นตัวกลางระหว่างผู้ใช้งานระบบโทรศัพท์เคลื่อนที่กับเครือข่ายของผู้ให้บริการโทรศัพท์เคลื่อนที่โดยอาศัยคลื่นวิทยุเป็นสื่อกลางในการติดต่อสื่อสารกับโทรศัพท์ที่ของผู้ใช้งาน

- สวนควบคุมสถานีฐาน(Base Station Controller-BSC) คือ ชุมสายหนึ่งที่ทำหน้าที่ควบคุมเกี่ยวกับคลื่นวิทยุในระบบเช่นควบคุมการ Handover จัดการเกี่ยวกับช่องสัญญาณวิทยุต่างๆ และเก็บข้อมูลเกี่ยวกับ Cell นอกจากนี้ ยังจะทำการควบคุมกำลังส่งสัญญาณของสถานีฐานและโทรศัพท์เคลื่อนที่อีกด้วย ในการทำงานนั้น MSC แต่ละ MSC จะควบคุม BSC 1 BSC หรือมากกว่านั้น และในแต่ละ BSC จะควบคุม BTS หลาย ๆ BTS

2.7.1.3 Switching System(SS)

จะทำหน้าที่สลับสัญญาณจากจุดหนึ่งไปยังอีกจุดหนึ่ง ซึ่งรวมถึงการเชื่อมโยงกับชุมสายอื่นด้วย ตลอดจนการเก็บข้อมูลตำแหน่งของเครื่องลูกข่ายใน Network ประกอบไปด้วย 5 ส่วน ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 Switching System(SS) [4]

- ชุมสายโทรศัพท์เคลื่อนที่(Mobile Services Switching Center -MSC) ทำหน้าที่ติดต่อจุดเชื่อมต่อสัญญาณเข้าออกชุมสาย, ควบคุมการสื่อสาร, ส่งข้อมูลเชื่อมต่อกับชุมสายอื่น, เก็บข้อมูลการใช้บริการและควบคุมการย้ายข้ามเซลล์

- หน่วยเก็บข้อมูลท้องถิ่นหรือฐานข้อมูลผู้ใช้บริการท้องถิ่น(Visiting Location Register -VLR) ทำหน้าที่ เป็นฐานข้อมูล เพื่อเก็บข้อมูลชั่วคราวของเครื่องโทรศัพท์เคลื่อนที่ที่ใช้บริการข้ามเขต ต่างชุมสาย รวมทั้งบอกตำแหน่งปัจจุบันของเครื่องโทรศัพท์เคลื่อนที่ด้วย

- หน่วยเก็บข้อมูลหลักของผู้ใช้บริการหรือฐานข้อมูลหลักของผู้ใช้บริการ (Home Location Register - HLR) ทำหน้าที่เป็น ฐานข้อมูลเก็บข้อมูลของผู้ใช้บริการ(ข้อมูลในซิมการ์ด) เช่น หมายเลขโทรศัพท์ในซิมการ์ดยอดเงินบริการที่ใช้, บริการเสริม ต่างๆ ในซิมการ์ด(ถ้าซิมการ์ดหาย เราสามารถแจ้งผู้ให้บริการเพื่อขอซิมใหม่เบอร์เดิมได้และข้อมูลต่างๆในซิมการ์ดเหมือนเดิมทุกอย่าง)

- ศูนย์ตรวจสอบการใช้งาน(Authentication Center -AUC) ใช้สาหรับเก็บ AUTHENTICATION เพื่อตรวจสอบว่าผู้ใช้โทรศัพท์ได้ลงทะเบียนอย่างถูกต้องหรือไม่และใช้สาหรับระบบการป้องกันการดักฟัง

- หน่วยเก็บข้อมูลเลขหมายประจำเครื่อง(Equipment Identity Register - EIR) เป็น DATABASE ที่เก็บข้อมูล IDENTITY ของเครื่องโทรศัพท์เคลื่อนที่เพื่อป้องกันไม่ให้โทรศัพท์เคลื่อนที่ที่ไม่ลงทะเบียนหรือได้มาอย่างไม่ถูกต้องตามกฎหมาย เข้ามาใช้งานในระบบได้ การติดตั้ง ELR ส่วนใหญ่จะอยู่ร่วมกับ AUC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1.4 Operation and Maintenance Center(OMC)

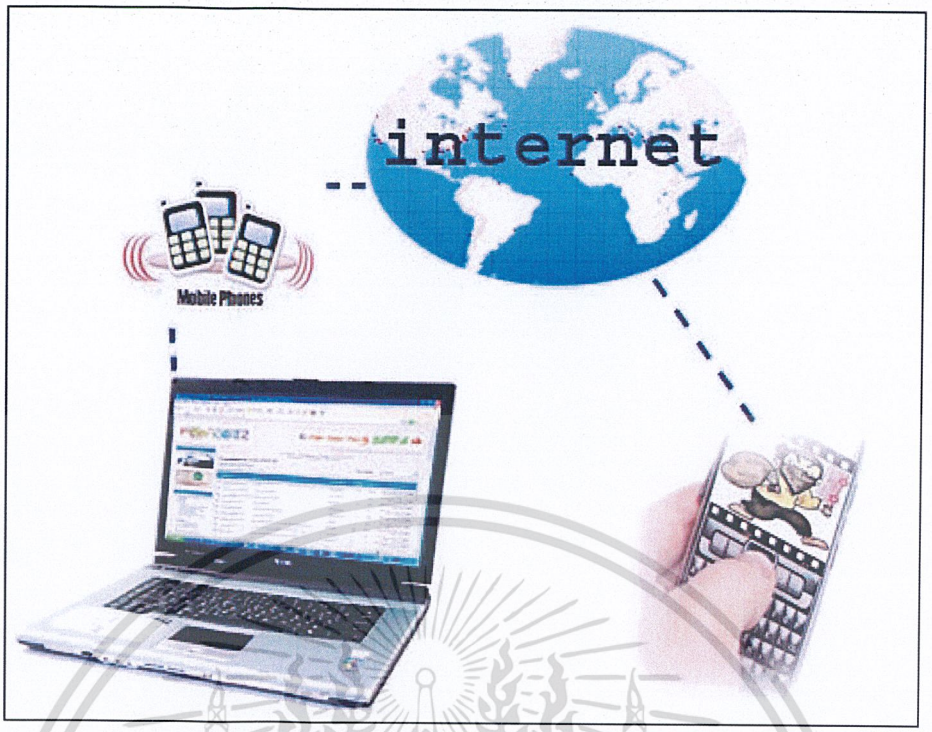
ทำหน้าที่ควบคุม และ บริหารการทำงานของระบบโครงข่ายโดยรวม จัดการกับปัญหาของอุปกรณ์บางส่วนที่อาจเกิดความเสียหายการปรับตั้งค่าต่างๆ ภายในระบบให้เหมาะสม การจัดการเรื่องสมาชิกผู้ใช้บริการของระบบซึ่งรวมไปถึงการคิดคาบริการและออกบิลเก็บค่าบริการการทำงานของ OMC ส่วนใหญ่แล้วจำต้องมีการติดต่อสื่อสารกับฐานข้อมูล HLR

2.8 GPRS

ในโครงงานนี้เราจะใช้ GPRS เพื่อใช้ในการอัพโหลดภาพขึ้นเซิร์ฟเวอร์ ดังนั้นเราจึงควรทราบพื้นฐานของ GPRS มาพอสังเขป

GPRS ย่อมาจากคำว่า General Packet Radio Service เป็นวิวัฒนาการของการสื่อสารข้อมูล ไร้สายแบบ Packet Switching คือ การแบ่งข้อมูลออกเป็นส่วนเล็ก ๆ ที่เรียกว่า Packet ซึ่งมีความสามารถในการส่งข้อมูลโครงข่ายได้ดีกว่าแบบเดิม ทำให้สามารถตรวจสอบ ความผิดพลาดในการส่งและยังช่วยเพิ่มอัตราการส่งข้อมูลสูงขึ้นอีกด้วยเทคโนโลยี GPRS นี้ สร้างขึ้นมาเพื่อให้สามารถทำธุรกรรมต่างๆ ผ่านโทรศัพท์เคลื่อนที่ได้ง่ายและสะดวกขึ้น จุดเด่นของระบบนี้คือ มีการเชื่อมต่อระบบอินเทอร์เน็ตตลอดเวลา (Always On) โดยไม่เสียค่าบริการและยังสามารถโทรออกและรับสายเข้าได้ในขณะที่เชื่อมต่ออินเทอร์เน็ตอยู่ การเสียค่าบริการจะคิดก็ต่อเมื่อมีการรับหรือส่งข้อมูล(Download หรือ Upload) เท่านั้น โดยคิดตามขนาดข้อมูลไม่คิดตามเวลาการเชื่อมต่ออินเทอร์เน็ต(Airtime) เหมือนเมื่อก่อน จึงทำให้ประหยัดค่าบริการได้มาก นอกจากนี้ยังสามารถใช้โทรศัพท์มือถือที่มีระบบGPRS เชื่อมต่อกับ PDA หรือ Computer Notebook ก็จะสามารถเข้าสู่โลกอินเทอรเน็ตได้อย่างง่ายดาย และยังสามารถรับข้อมูลข่าวสารในรูปแบบของ Video ไม่ว่าจะเป็นรายการข่าว ละคร กีฬา ข้อมูลการจราจร ซึ่งจะทำให้ผู้ใช้ทันต่อเหตุการณ์ในโลกปัจจุบันอยู่เสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 การใช้งาน GPRS

ข้อมูลที่รับส่งผ่านเครือข่าย GPRS จะถูกตัดแบ่งเป็น packet ย่อยๆ ก่อน ในแต่ละ packetจะมีข้อมูลระบุถึงที่มาที่สัมพันธ์กันเพื่อใช้ในการประกอบ กลับขึ้นมาเป็นข้อมูลเดิมอีกครั้ง ซึ่งการนำข้อมูลมาต่อเข้าด้วยกันจะใช้วิธีดูจากความสัมพันธ์ของแต่ละส่วน ซึ่งอาจจะมีวิธีที่แตกต่างกันไป GPRS จะต้องการโหนดใหม่เพิ่มเติมอีก 2 โหนดสำหรับจัดการกับทราฟฟิกแบบ Packet คือ

- **GGSN(Gateway GPRS Service Node)** ทำหน้าที่เป็น Gateway เชื่อมต่อระหว่างเครือข่าย GPRS กับ เครือข่ายข้อมูลทั่วไปเช่น IP และ X.25 ซึ่งรวมถึงการเชื่อมต่อกับเครือข่าย GPRS อื่นๆ เพื่อการ Roaming ด้วย
- **SGSN(Serving GPRS Service Node)** ทำหน้าที่เกี่ยวกับการเชื่อมต่อเส้นทาง(Routing) ระหว่าง SGSN ในแต่พื้นที่สำหรับผู้ใช้ทุกคนในพื้นที่ให้บริการ

2.8.1 ข้อดีของการใช้ GPRS

- GPRS เป็นระบบที่ทำให้สามารถรับส่งข้อมูลความเร็วสูง(non voice) ด้วยความเร็วสูงสุดถึง 40 Kbps ทำให้การใช้บริการ internet การใช้ WAP และ mms สามารถทำงานได้รวดเร็ว เป็นผลให้เกิดการพัฒนา เทคโนโลยีการบริการ non voice กันมากขึ้น เช่น การใช้บริการ WAP เพื่อการโหลดรูปภาพ เสียงเรียกเข้า การรับส่งข้อความแบบ mms และการใช้งานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินเตอร์เน็ต ผ่านโทรศัพท์เคลื่อนที่กับคอมพิวเตอร์โน้ตบุค และเครื่อง PDA เพื่อใช้งาน Web browsing และ E-mail

- ระบบ GPRS เป็นระบบที่มีความสามารถในการใช้งานแบบ always on คือในการเชื่อมต่อ สามารถเชื่อมต่อเข้ากับ Gateway ของระบบเครือข่ายได้โดยอัตโนมัติ ไม่จำเป็นต้องทำการโทรออกเหมือนกับระบบ CSD ทำให้การเชื่อมต่อสามารถทำได้อย่างรวดเร็ว

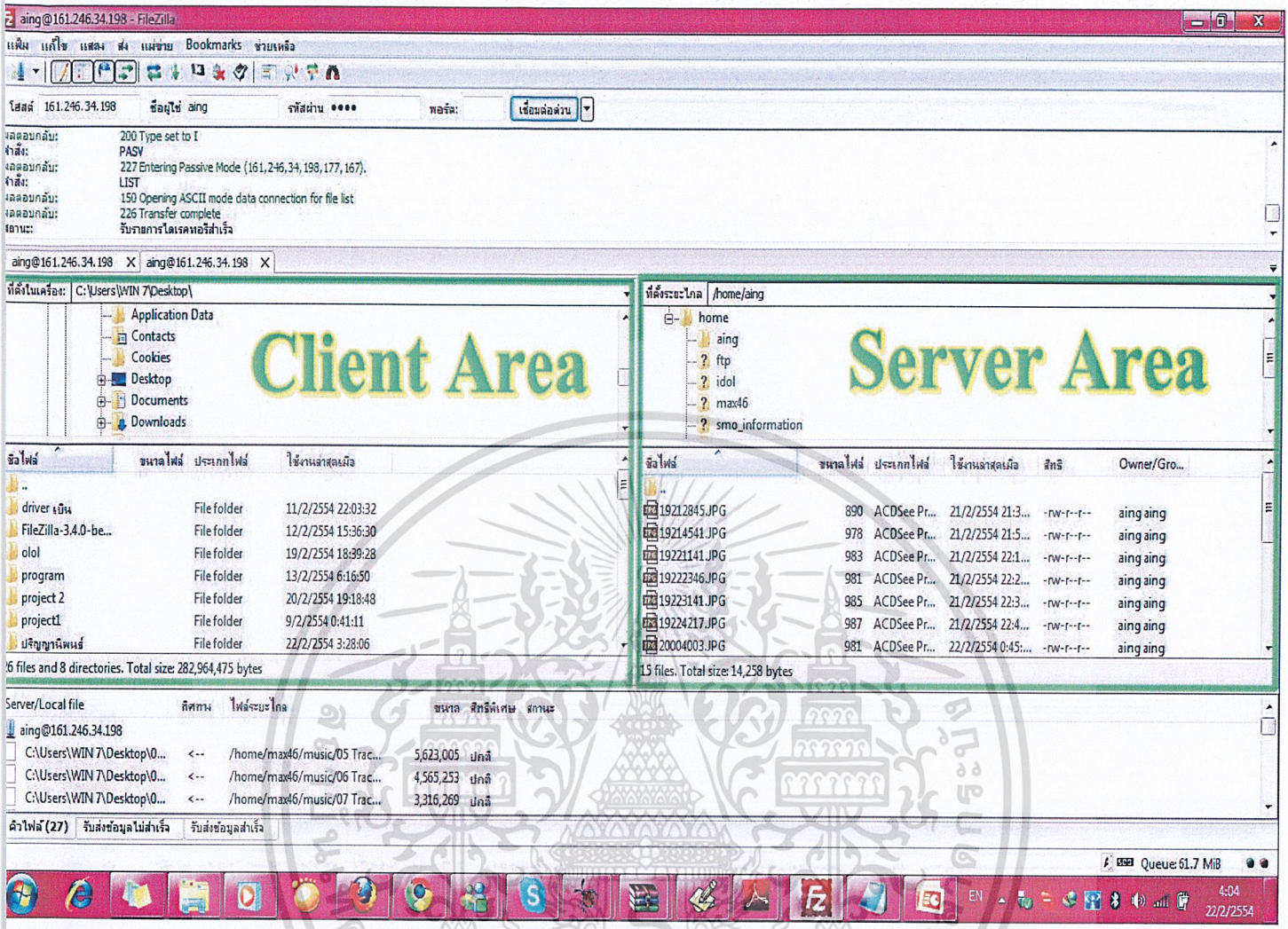
- ไม่สูญเสียการติดต่อขณะใช้งาน ระบบ GPRS อยู่เราจะไม่สูญเสียการติดต่อกับผู้อื่น เนื่องจาก GPRS ไม่มีการ Dial up ดังนั้นขณะใช้งานอยู่สามารถมีผู้โทรเข้า หาเราและรับสายสนทนาได้ทันที และขณะเดียวกันเราก็สามารถโทรออกเพื่อสนทนาได้ทันทีเช่นเดียวกัน

- ค่าใช้จ่ายในการใช้งานที่ถูกลง และมีความเป็นธรรม เพราะการคิดค่าใช้จ่าย จะคิดตามปริมาณของข้อมูลที่เรารับและส่งผ่านเครือข่าย เรียกว่าสามารถเชื่อมต่อ ทิ้งไว้ทั้งวันได้โดยไม่เสียเงินหากไม่มีการรับส่งข้อมูล

2.9 โปรแกรม Filezilla

FileZilla เป็นโปรแกรมที่ใช้ติดต่อกับ FTP server เพื่อดาวน์โหลดหรืออัปโหลดไฟล์ โดยเฉพาะกับเว็บไซต์ ทำให้เหมาะสำหรับนักออกแบบเว็บ นอกจากนี้ FileZilla ยังรองรับการถ่ายไฟล์อย่างปลอดภัยผ่าน SSH(SFTP) อีกด้วย FileZilla รองรับการกลับมาถ่ายไฟล์ต่อในกรณีที่อัปโหลดหรือดาวน์โหลดล้มเหลว และทำงานได้ดีผ่านไฟร์วอลและพรอกซี FileZilla มีวิธีใช้งานที่สะดวก แบ่งวินโดว์ออกเป็นสองส่วนคือแสดงฝั่ง local กับ remote จากนั้นคุณก็สามารถ drag and drop ไฟล์ระหว่างสองฝั่งเพื่ออัปโหลดหรือดาวน์โหลดได้ทันที คุณสามารถ login อย่างรวดเร็วโดยป้อนข้อมูลของไซต์คือที่อยู่ ผู้ใช้ และรหัสผ่านได้บนทูลบาร์เลยทันที นอกจากนี้ FileZilla ยังมี host manager ที่เก็บรายละเอียดของ เซิร์ฟเวอร์ที่คุณใช้เป็นประจำ เพื่อให้คุณเชื่อมต่อได้อย่างง่ายและรวดเร็ว FileZilla ทำงานได้เฉพาะระบบ Windows เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

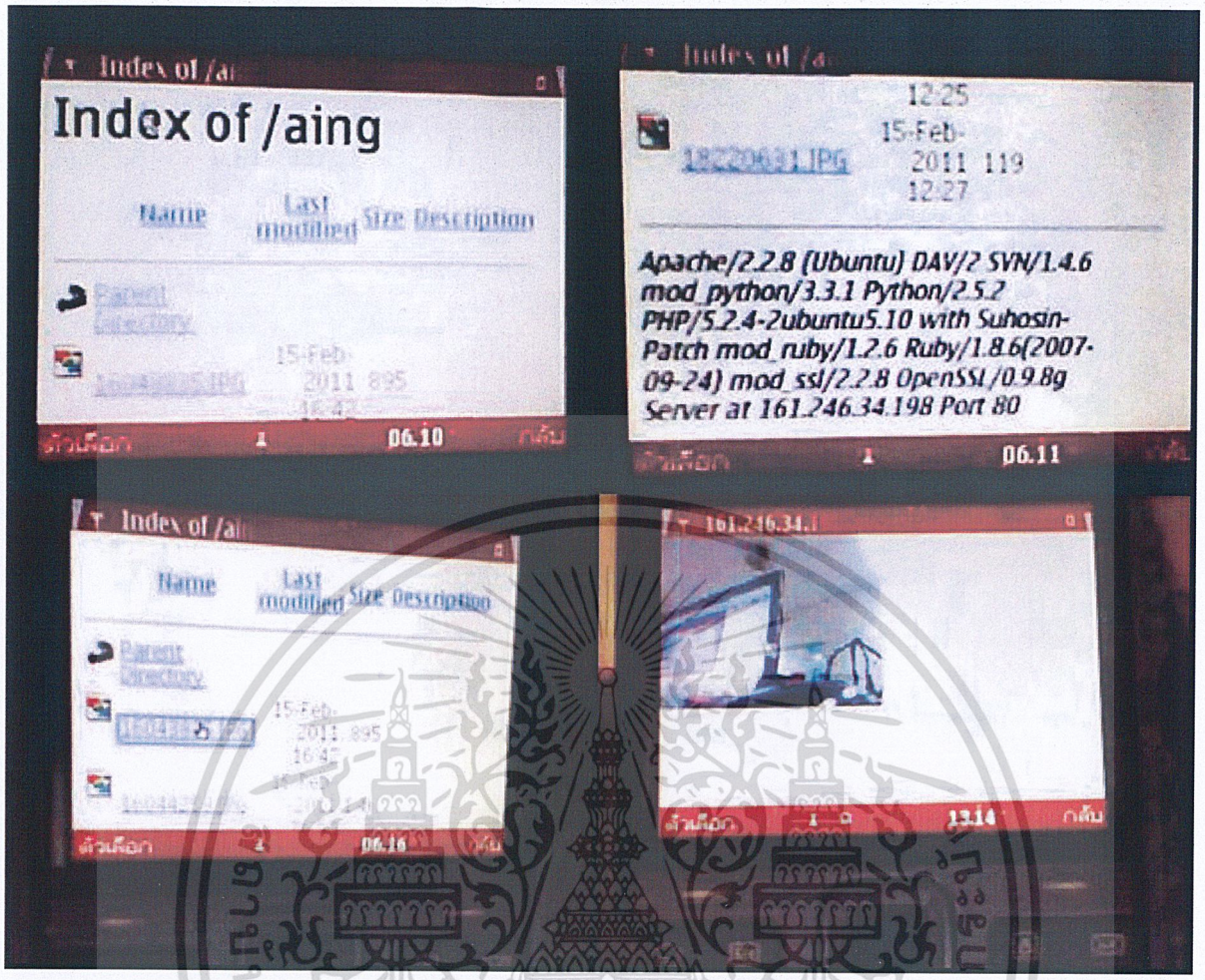


รูปที่ 2.13 โปรแกรม Filezilla

2.10 การแสดงภาพบนมือถือ

ในส่วนของการแสดงภาพบนมือถือ โครงงานนี้จะใช้มือถือในการเปิดดูภาพ ซึ่งจะใช้มือถือเครื่องนี้เข้าใช้งานอินเตอร์เน็ต โดยเข้าไปที่ URL: http://161.246.34.198/aing/ ซึ่งจะทราบถึง URL นี้ได้ผู้ใช้งานต้องเป็นสมาชิกของระบบนี้ก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 การแสดงภาพบนมือถือ

2.11 การรักษาความปลอดภัยของการสื่อสารข้อมูลและการเข้าถึงระบบ

เนื่องด้วยโครงงานนี้แบ่งการทำงานเป็น 3 ส่วน ได้แก่ การทำงานทางด้านผู้ใช้ การทำงานของเซิร์ฟเวอร์ และการทำงานในรถ จึงขอกล่าวทีละส่วน

2.11.1 ด้านผู้ใช้

การที่ผู้ใช้จะเข้ามาดูภาพในระบบได้นั้นจะต้องเข้าดูภาพผ่านเว็บไซต์ จะต้องเป็นสมาชิกของเซิร์ฟเวอร์นี้ก่อน จึงจะเข้าไปดูภาพได้ โดยการเปิดเว็บไซต์นั้น เราสามารถที่จะใช้ GPRS ในการเข้าสู่ เครือข่ายอินเตอร์เน็ต เพื่อเชื่อมต่อเว็บไซต์เพื่อดูข้อมูลบนเซิร์ฟเวอร์นั้นต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.2 ด้านเซิร์ฟเวอร์

ในส่วนของเซิร์ฟเวอร์นั้น ตัว Application ของเซิร์ฟเวอร์มักมีการดูแลเรื่องความปลอดภัยของตัวมันเองอยู่แล้ว คือมีการให้ใส่ชื่อผู้ใช้และรหัสผ่านก่อนที่จะเข้าไปที่เซิร์ฟเวอร์ได้

ไม่ใช่แค่ในส่วนของตัวโปรแกรม การรักษาความปลอดภัยของเครื่องคอมพิวเตอร์ที่เป็นเซิร์ฟเวอร์ก็สามารถทำได้ด้วยการตั้งให้มีการใส่ชื่อผู้ใช้และรหัสผ่านก่อนที่จะมีการใช้งานเครื่องคอมพิวเตอร์นั้นได้

2.11.3 ระบบภายในรถยนต์

การที่จะเข้าถึงโปรแกรมที่อยู่ในรถยนต์ก็มีหลักการทำงานเหมือนกันกับการเข้าใช้เครื่องเซิร์ฟเวอร์ คือ ต้องมีการเป็นสมาชิกของระบบนี้ก่อน

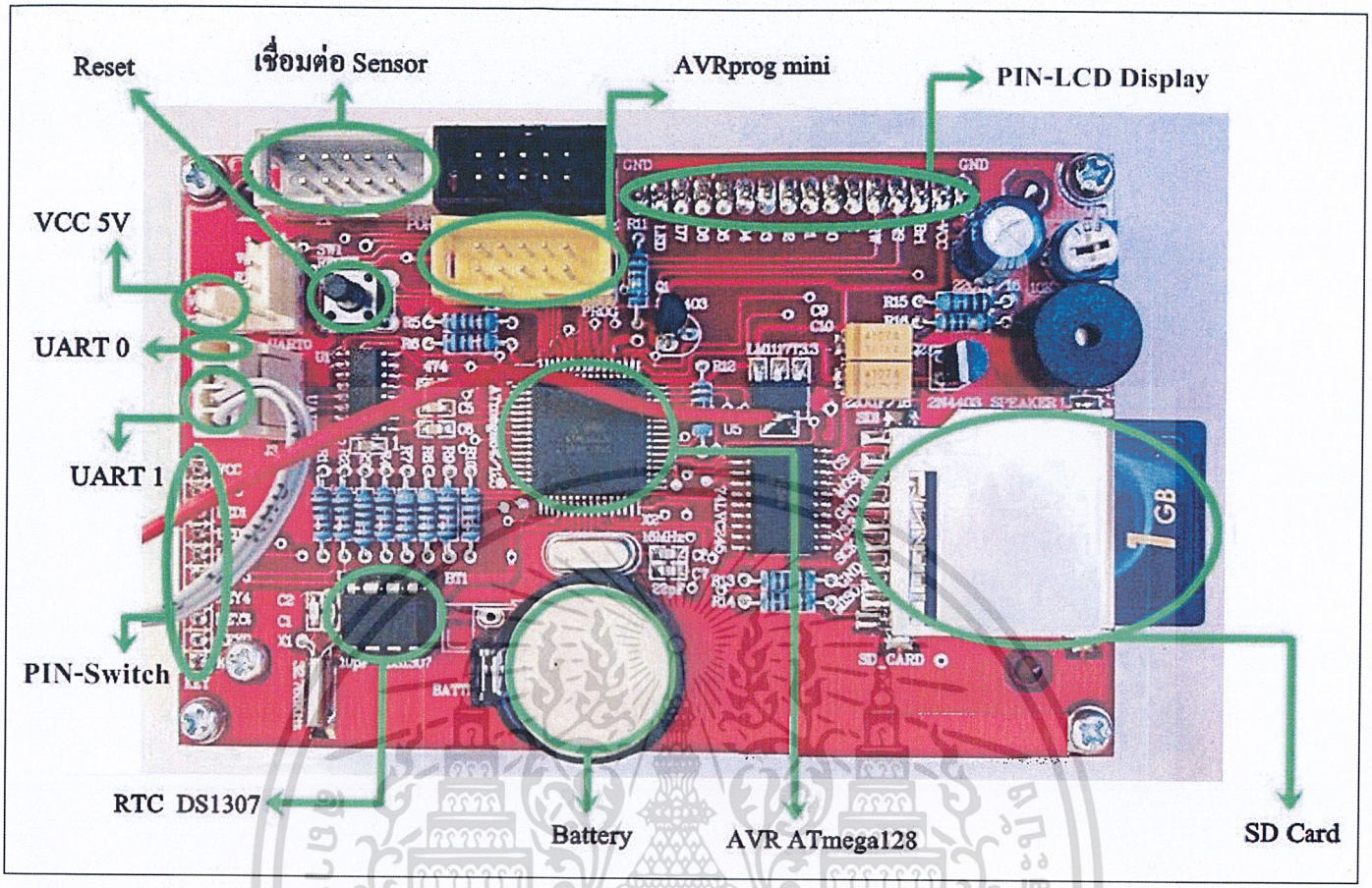
2.12 ส่วนของวงจรและอุปกรณ์ที่ใช้ติดตั้งกล้องในรถยนต์

2.12.1 บอร์ดไมโครคอนโทรลเลอร์AVR ATmega128

บอร์ดไมโครคอนโทรลเลอร์AVR ATmega128 ที่ใช้ในโครงงานนี้จะประกอบส่วนต่างๆที่สำคัญ ดังนี้

- ภาคแหล่งจ่ายไฟ(Power Supply) ชุดหน่วยประมวลผล(CPU)
- Input, Output(I/O) ของ CPU
- ซ็อกเก็ตสำหรับ SD Card เพื่อติดต่อกับ SD Card
- RTC DS1307 เพื่อเป็นระบบฐานเวลา
- ปุ่ม Reset เพื่อ Reset การทำงานของกล้อง และหน้าจอ LCD
- UART 0 เพื่อต่อกับ GSM Module
- UART 1 เพื่อต่อกับ กล้อง Camera Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 ส่วนต่างๆที่สำคัญของบอร์ด AVR ATmega128

การทำงานของวงจร จะเริ่มจากการป้อนไฟฟ้ากระแสตรงขนาด 9 ถึง 12 โวลต์ เข้าทางคอนเน็ตเตอร์ CON3 ซึ่งแรงดันที่เข้ามานี้จะถูก แบ่งออกเป็น 2 ค่า คือ 5 โวลต์ และ 3.3 โวลต์ เพื่อป้อนให้กับ ส่วนต่างๆของวงจร โดยที่ IC4(LM2575-5) ทำหน้าที่ลด ระดับแรงดันให้เหลือเท่ากับ 5 โวลต์ เพื่อจ่ายให้กับส่วนต่างๆของวงจรนอกจากอุปกรณ์ในส่วนของ SD Card และกล้อง Camera Module และแรงดัน 5 โวลต์ นี้จะถูกลดให้เหลือ 3.3 โวลต์ โดย IC2(LM1117T3.3) เพื่อจ่ายให้กับกับ ตัว หน่วย ความจำ SD Card และกล้อง Camera Module ส่วนตัว IC1(ATmega128) เป็นตัวประมวลผลกลาง จะมีหน้าที่ในการเชื่อมต่อกับอุปกรณ์ส่วนต่างๆ ทั้งหมดในวงจร ซึ่งอุปกรณ์แต่ละตัวก็จะมีรูปแบบการเชื่อมต่อที่แตกต่างกันไป ดังนี้

การเชื่อมต่อกับการ์ดหน่วยความจำ การเชื่อมต่อกับหน่วยความจำ SD Card จะใช้มาตรฐานแบบ SPI ซึ่งโดยปกติแล้วในขณะที่ IC1(ATmega128) มีสภาวะเป็นลอจิกสูง แรงดันจะมีค่าประมาณ 5 โวลต์ หากเชื่อมต่อเข้ากับการ์ดหน่วยความจำที่มีระดับการทำงานที่ 3.3 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยตรงจะทำให้การ์ดความจำเกิดความเสียหายได้ ดังนั้นจึงต้องใช้ IC3(74LVC245) ทำหน้าที่เปลี่ยนแปลงระดับแรงดันให้เหลือ 3.3 โวลต์

- ขา PB0 ต่อกับขา CS เพื่อเป็นขา Select ให้กับ SD Card
- ขา PB1 เป็นขาสัญญาณนาฬิกา ให้กับ ให้กับ SD Card
- ขา PB2 เป็นขาข้อมูลเพื่อส่งไปให้กับ SD Card
- ขา PB3 เป็นขาข้อมูลจาก SD Card
- ขา PB4 เป็นขาเพื่อตรวจสอบสถานะของ SD Card ว่ามีอยู่หรือไม่

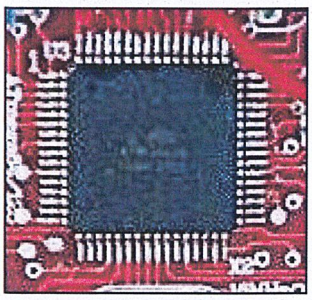
ในส่วนของการเชื่อมต่อกับกล้องจะใช้มาตรฐานแบบอนุกรม ซึ่งการเชื่อต่อในรูปแบบดังกล่าวจะใช้ขา PD2 และ PD3 ของ IC1(ATmega128) ที่ภายในมีคุณสมบัติเป็นพอร์ตสื่อสารอนุกรม (UART 1) อยู่แล้ว ในการทำงานของกล้องจะใช้ระดับแรงดัน 3.3 โวลต์ ซึ่งก็เช่นเดียวกับ SD Card เพราะจำเป็นต้องใช้ IC3(74LVC245) เพื่อทำการลดระดับแรงดันลงก่อนที่จะต่อเข้ากับพอร์ตสื่อสารของกล้อง สำหรับการจ่ายไปเลี้ยงให้กับกล้องจะไม่จ่ายค้างไว้ตลอดเหมือนกับอุปกรณ์ตัวอื่นๆ เนื่องจากเทคนิคของการเชื่อมต่อกับกล้องจะต้องมีการตัดและต่อไฟเลี้ยงเพื่อ Reset กล้องก่อนทำการเชื่อมต่อกล้องเสมอ

ในส่วนของการเชื่อมต่อกับ GSM Module จะเป็นการสื่อสารแบบอนุกรม(UART 0) ซึ่งเป็นหนึ่งในชุดสื่อสารที่บรรจุอยู่ภายใน IC1(ATmega128)

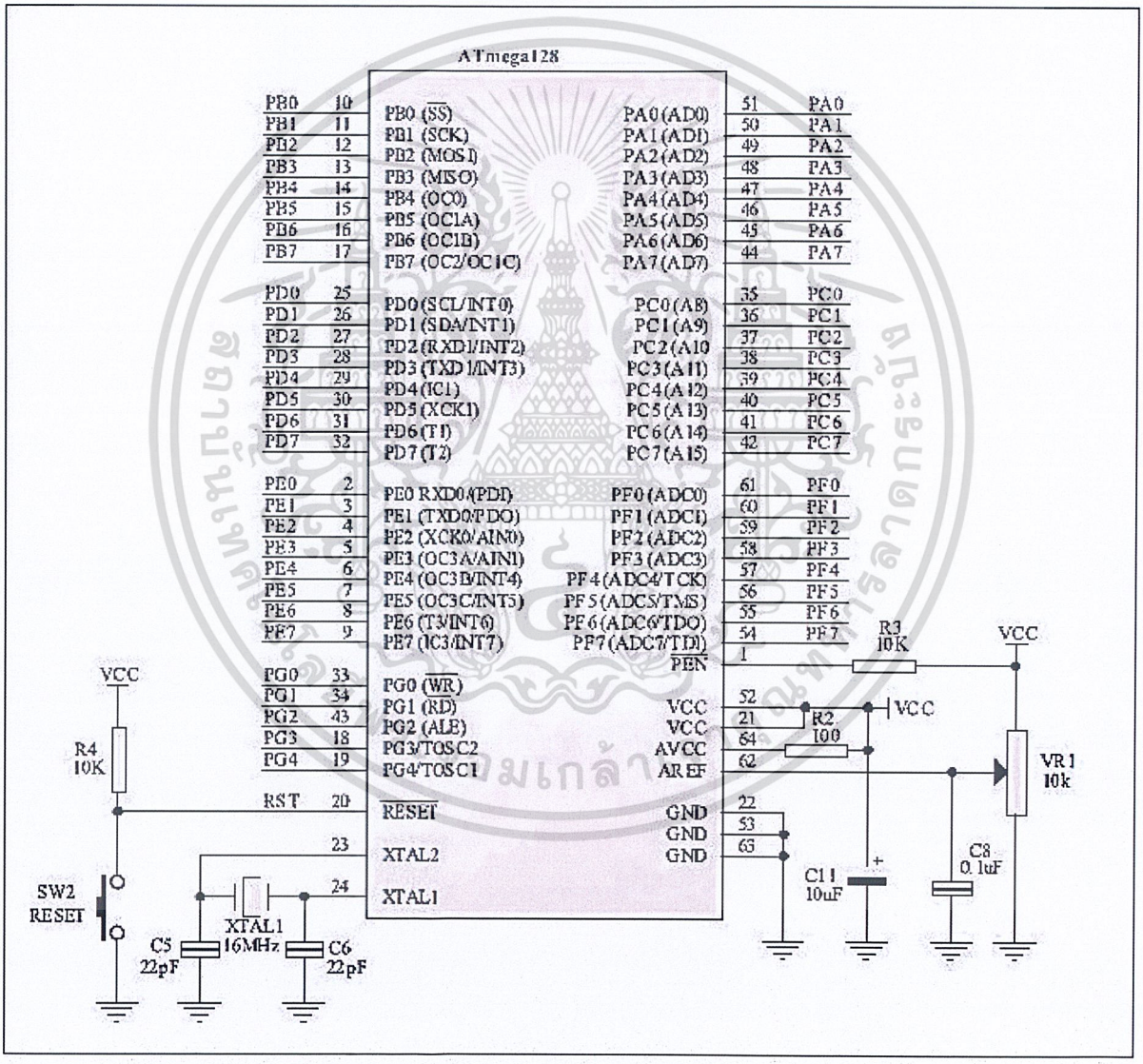
2.12.2 ไมโครคอนโทรลเลอร์ AVR ATmega128

ไมโครคอนโทรลเลอร์ ตระกูล AVR ATmega128(รูปที่2.10) มี address bus 16 บิต อ้างอิงหน่วยความจำได้สูงสุด 64 กิโลไบต์ เนื่องจาก instruction register มีความกว้างขนาด 16 บิต ดังนั้นจึงรองรับโปรแกรม 64 กิโลไบต์(128 กิโลไบต์)(ชุดคำสั่งของ AVR มีขนาดตั้งแต่ 1-2 word) PIC มีความกว้างของ instructions register ตั้งแต่ 10บิต, 12บิต, 14บิต, 16บิต , ds PIC มีความกว้างของ instruction register 24 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 ไมโครคอนโทรลเลอร์ ตระกูล AVR ATmega128



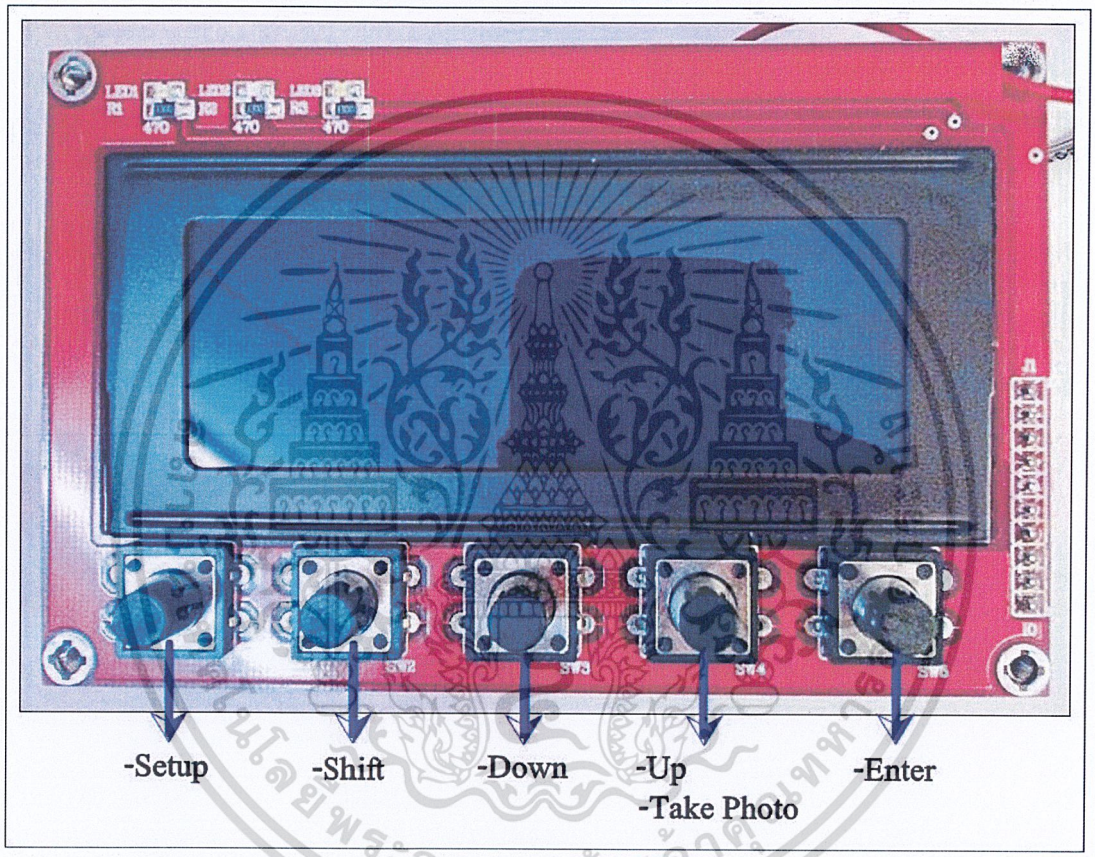
รูปที่ 2.17 วงจรไมโครคอนโทรลเลอร์ ตระกูล AVR ATmega128[5]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.3 จอแสดงผล LCD(LCD Display) และ สวิตซ์ควบคุม

จอแสดงผล LCD(LCD Display) และ สวิตซ์ควบคุม จะนำมาต่อกับบอร์ด เพื่อใช้แสดงเวลา ในการส่งภาพขึ้นเซิร์ฟเวอร์ และ สามารถตั้งเวลา วัน เดือน ปี ได้ โดยใช้ สวิตซ์ควบคุม

ซึ่งจอแสดงผล LCD(LCD Display) จะรับค่ามาจากการประมวลผลของไมโครคอนโทรลเลอร์ และส่งให้จอแสดงผล LCD(LCD Display)แสดงผลตามค่าที่ประมวลผลได้ ซึ่ง จอแสดงผล LCD(LCD Display) มีขนาด 4 แถว x 20 หลัก



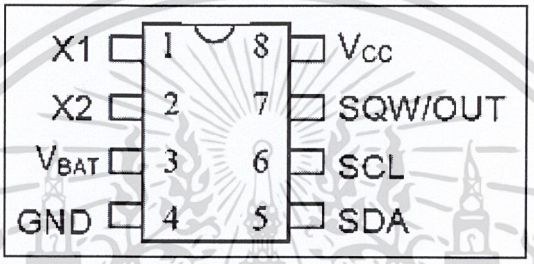
รูปที่ 2.18 จอแสดงผล LCD(LCD Display) และสวิตซ์ควบคุม

2.12.4 Real Time Clock ด้วย DS1307

ระบบฐานเวลา เป็นสิ่งสำคัญที่สามารถนำไปใช้ในอุปกรณ์อีเล็กโทรนิกส์ได้หลากหลาย ภายในไมโครคอนโทรเลอร์เองก็มีไทเมอร์เพื่อใช้ในการจับเวลา หรือนำไปใช้เป็นฐานเวลาจริงได้เช่นกัน แต่เนื่องจากไมโครคอนโทรเลอร์สามารถทำงานได้ต่อเมื่อมีไฟเลี้ยงเท่านั้น ดังนั้นการใช้ไทเมอร์ของไมโครคอนโทรลเลอร์ สร้างฐานเวลาจริงจึงไม่ เหมาะสมในบางแอปพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS1307 เป็น IC ฐานเวลาของDallas Semiconductor มีบัสรับส่งข้อมูลแบบ I²C ซึ่งเป็นแบบ 2 wire สามารถสื่อสารได้ 2 ทิศทาง(bi-direction bus) ฐานเวลาของ DS1307 นั้นสามารถเก็บข้อมูล วินาที, นาที, ชั่วโมง, วัน, วันที่, เดือน และปี ได้ ระบบเวลาสามารถทำงานโหมดรูปแบบ 24 ชั่วโมง หรือ 12 ชั่วโมง AM/PM ก็ได้ ภายมีระบบตรวจจับแหล่งจ่ายไฟ โดยถ้าแหล่งจ่ายไฟหลักถูกตัดไป DS1307 สามารถสวิตซ์ไปใช้ไฟจากแบตเตอรี่ และทำงานต่อไป โดยที่ยังสามารถรักษาข้อมูลไว้ได้ โครงสร้างมีขาทั้งหมด 8 ขาดังแสดงในรูปที่ 2.19 และมีรายละเอียดการทำงานของขาต่างๆ ดังนี้



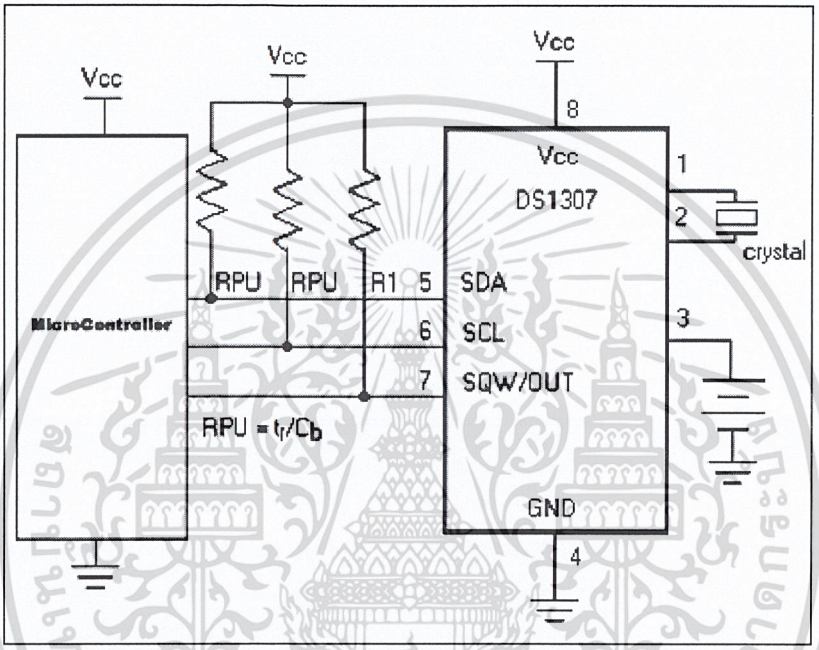
รูปที่ 2.19 ตำแหน่งขาไอซี RTC DS1307[6]

- VCC ใช้ต่อไฟเลี้ยง +5โวลต์
- GND ใช้ต่อGround
- VBAT ใช้ต่อกับแบตเตอรี่ 3โวลต์ เพื่อรักษาการทำงาน ในกรณีที่ไม่มีไฟเลี้ยงจ่าย
- SDA ขารับส่งข้อมูลด้วยระบบบัส I²C
- SCL ขาสัญญาณนาฬิกาสำหรับการรับส่งข้อมูลด้วยระบบบัส I²C
- SQW/OUT ขาเอาต์พุตสัญญาณ Square Wave สามารถเลือกความถี่ได้
- X1, X2 ใช้ต่อกับคริสตอลความถี่มาตรฐาน 32.768 กิโลเฮิรตซ์ เพื่อสร้างฐานเวลาจริงให้กับ IC

ระบบบัสข้อมูลแบบ I²C(Inter-IC Communication) ได้ถูกพัฒนาขึ้นโดยบริษัทฟิลิปส์ (Phillips) การรับส่งข้อมูลใช้สายสัญญาณเพียงแค่ 2 เส้น คือสายสัญญาณข้อมูล SDA(Serial Data line) และสายสัญญาณนาฬิกา SCL(Serial Clock line) มีการทำงานเป็นแบบ Master, Slave โดยอุปกรณ์ที่ทำหน้าที่เป็น Master(ไมโครคอนโทรลเลอร์) จะควบคุมการรับส่งข้อมูล และควบคุมสัญญาณนาฬิกาบน SCL ส่วนอุปกรณ์ Slave(DS1307) นั้นจะทำงานภายใต้การควบคุมของอุปกรณ์ Master

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อใช้งานร่วมกับไมโครคอนโทรลเลอร์ด้วยระบบบัส I²C นั้นสามารถทำได้โดยต่อตัวต้านทาน Pull up ดังแสดงในรูปที่ 2.20 ในกรณีที่ต้องการต่อร่วมกับอุปกรณ์ Slave หลายตัว ก็สามารถทำได้โดยต่ออุปกรณ์ Slaveขนานกันไป การติดต่อสื่อสารระหว่างอุปกรณ์ Master กับ Slave แต่ละตัวนั้น จะถูกแยกโดย Address ของอุปกรณ์ Slave ซึ่งจะถูกส่งจากอุปกรณ์ Master ไปยังอุปกรณ์ Slave ก่อนเริ่มการรับส่งข้อมูล



รูปที่ 2.20 การเชื่อมต่อ DS1307 เข้ากับไมโครคอนโทรลเลอร์ด้วยระบบบัสแบบ I²C [6]

การรับส่งข้อมูลแบบ I²C นั้นมีข้อกำหนดอยู่ 2 ประการด้วยกันคือ

- การรับส่งข้อมูลจะเริ่มขึ้นได้เมื่อบัสมีสถานะว่างเท่านั้น
- ในช่วงที่ทำการรับส่งข้อมูลอยู่ สายสัญญาณ SDA ต้องไม่เปลี่ยนสถานะในช่วงที่ SCL มี สถานะเป็นลอจิก “1” ถ้า SDA มีการเปลี่ยนสถานะในช่วงที่ SCL เป็นลอจิก “1” จะถือว่าเป็นสัญญาณควบคุมการรับส่งข้อมูล

สถานะของการรับส่งข้อมูลแบบ I²C สามารถแบ่งออกได้เป็น 5 สถานะด้วยกันดังแสดงในรูปที่ 2.21 และมีรายละเอียดดังนี้

- สถานะว่าง(Bus not busy): สัญญาณ SDA และ SCL มีระดับสัญญาณเป็น High
- เริ่มส่งข้อมูล(Start data transfer): มีการเปลี่ยนระดับสัญญาณของ SDA จาก High เป็น Low ในขณะที่ SCL มีระดับสัญญาณเป็น High ค้างไว้

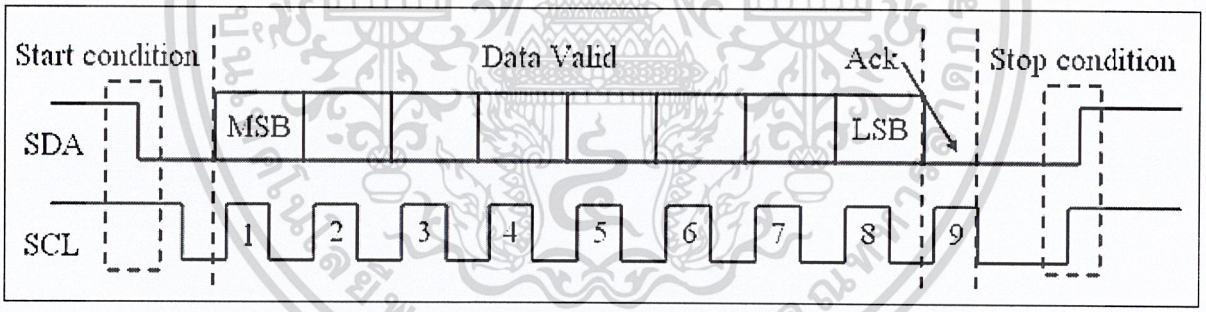
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หยุดส่งข้อมูล(Stop data transfer): มีการเปลี่ยนระดับสัญญาณของ SDA จาก Low เป็น High ในขณะที่ SCL มีระดับสัญญาณเป็น High ค้างไว้

- รับส่งข้อมูล(Data valid): มีการรับส่งข้อมูลผ่านสายสัญญาณ SDA โดยข้อมูลแต่ละบิตจะถูกส่งในช่วงที่ SCL มีระดับเป็น High โดยในช่วงที่ SCL มีสถานะเป็น High อยู่นั้น SDA จะต้องไม่เกิดการเปลี่ยนระดับสัญญาณ

SDA จะเปลี่ยนระดับของสัญญาณ ในช่วงที่ SCL มีระดับสัญญาณเป็น Low เท่านั้น ตามมาตรฐานการส่งข้อมูล แบบ I²C นี้สามารถส่งข้อมูลด้วยความถี่สัญญาณนาฬิกาสูงสุด 100 กิโลเฮิรตซ์ ที่โหมดการทำงานธรรมดา และ 400 กิโลเฮิรตซ์ ที่โหมดการทำงานแบบเร็ว แต่สำหรับ DS1307 สามารถทำงานได้ในโหมดธรรมดาเท่านั้น

ตอบรับ(Acknowledge): เกิดขึ้นหลังจากที่มีการรับส่งข้อมูลครบแล้วโดยอุปกรณ์ Master ต้องสร้างสัญญาณ Clock บน SCL เพิ่มอีกลูก อุปกรณ์ที่เป็นตัวรับข้อมูลจะดึงระดับสัญญาณบน SDA ให้เป็น Low เพื่อให้ตัวส่งรับรู้ว่าตัวรับได้รับข้อมูลครบแล้ว



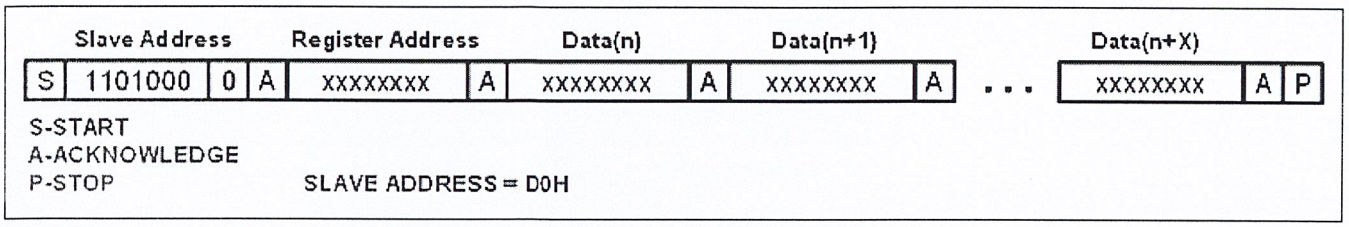
รูปที่ 2.21 การรับส่งข้อมูลผ่านบัส I²C [6]

ในการรับส่งข้อมูลผ่านบัส I²C อุปกรณ์ Master จะเป็นผู้สร้างสัญญาณ Clock บน SDA และเป็นตัวควบคุมสถานะ Start และ Stop เพื่อควบคุมการรับส่งข้อมูลทั้งหมด

การส่งข้อมูลไปยังอุปกรณ์ DS1307 ดังแสดงในรูปที่ 2.22 ไมโครคอนโทรลเลอร์ต้องสร้างสภาวะ Start ก่อน จากนั้นต้องส่ง Address ของDS1307 ขนาด 7 บิตซึ่งมีค่าเป็น 1101000 และตามด้วยบิตระบุทิศทางของข้อมูล ในกรณีที่เป็นการเขียนข้อมูลลง DS1307 จะต้องเป็น “0” จากนั้นไมโครคอนโทรเลอร์จะต้องส่งตำแหน่ง Address ภายในรีจิสเตอร์ของ DS1307 ที่ต้องการเขียน

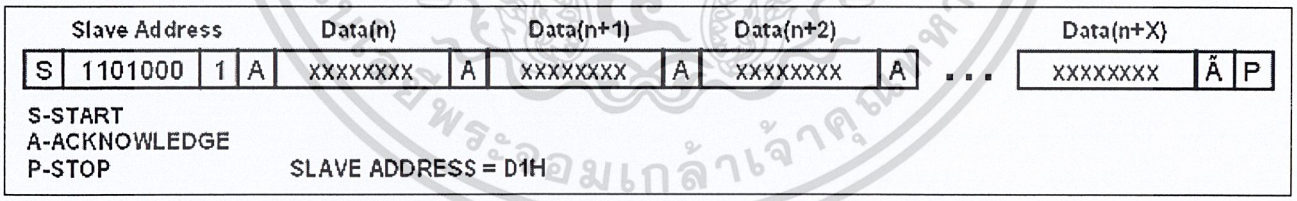
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลลง แล้วจึงค่อยเขียนข้อมูลลง โดยในการส่งข้อมูลแต่ละไบต์จะต้องรอบิต Ack จาก DS1307 ทุกไบต์ เมื่อส่งจนครบแล้ว ถึงจะสร้างสภาวะ Stop เพื่อกลับสู่สถานะว่าง



รูปที่ 2.22 การเขียนข้อมูลอุปกรณ์ Slave ผ่านบัส I²C [6]

การรับข้อมูลจากอุปกรณ์ Slave ดังแสดงในรูปที่ 2.23 เริ่มแรกไมโครคอนโทรลเลอร์ต้องสร้างสภาวะ Start ก่อน จากนั้นต้องส่ง Address ของ DS1307 ขนาด 7 บิตซึ่งมีค่าเป็น 1101000 และตามด้วยบิตระทิศทางของข้อมูล ในกรณีที่เป็นการอ่านข้อมูลจาก DS1307 จะต้องเป็น "1" จากนั้นจึงค่อยรับข้อมูลจากอุปกรณ์ Slave ทีละไบต์ โดยตำแหน่งที่อ่านเข้ามาจะขึ้นอยู่กับตำแหน่งรีจิสเตอร์พอยท์เตอร์ ซึ่งจะเป็นตำแหน่งท้ายสุดที่ได้ทำการเขียนข้อมูลไว้ เมื่ออ่านข้อมูลครบแต่ละไบต์อุปกรณ์ Master ต้องส่ง Acknowledge บิตกลับไปให้อุปกรณ์ Slave ด้วย ในกรณีที่เป็นไบต์สุดท้าย อุปกรณ์ Master ต้องส่ง "not acknowledge" กลับไป

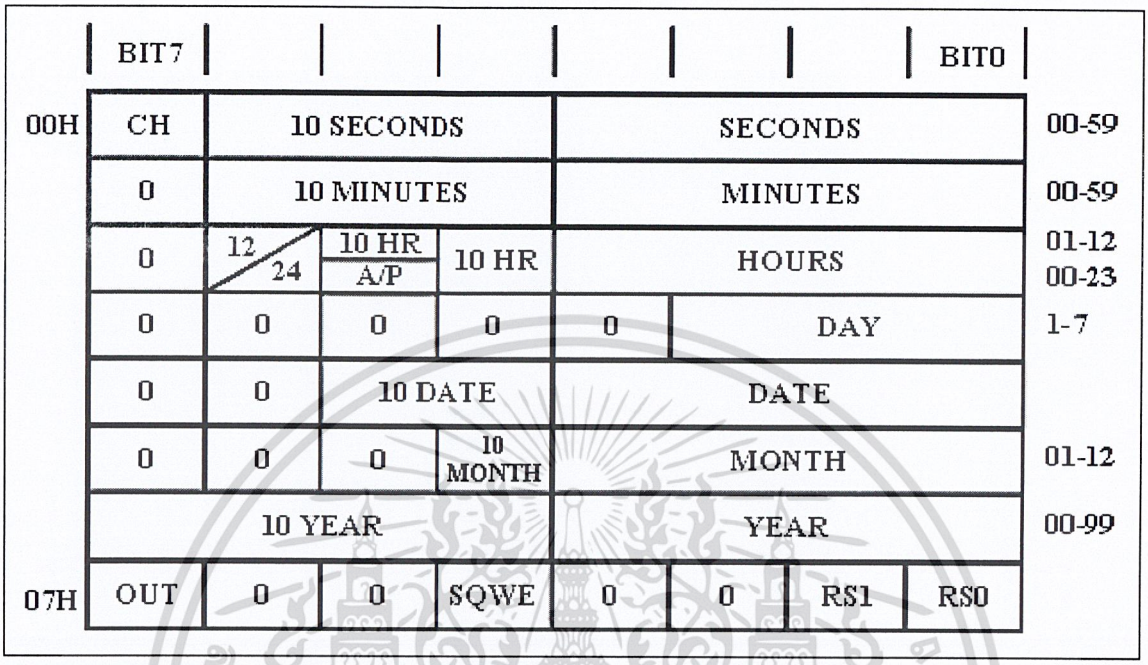


รูปที่ 2.23 การอ่านข้อมูลจากอุปกรณ์ Slave ผ่านบัส I²C [6]

ภายใน DS1307 มีรีจิสเตอร์ภายในใช้เก็บข้อมูลเวลาขนาด 7 ไบต์ 00H-06H ดังแสดงในรูปที่ 2.24 ข้อมูลค่าเวลา และวันที่จะถูกเก็บอยู่ในรูปของเลขฐาน 10 สามารถเลือกได้ว่าให้ทำงานแบบ 12 ชั่วโมง หรือ 24 ชั่วโมง โดยกำหนดที่บิตที่ 6 ที่แอดเดรส 02H โดยถ้าเป็น "1" จะเป็นการทำงานในโหมด 12 ชั่วโมง และเมื่อเลือกแบบ 12 ชั่วโมง ที่บิต 5 ในแอดเดรส 02H นั้นจะใช้แสดงค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AM/PM โดยถ้าบิตนี้เป็น “1” จะเป็น PM ในกรณีที่แสดงแบบ 24 ชั่วโมง บิตนี้จะใช้ในการแสดงค่าของหลักสิบในของหน่วยชั่วโมงด้วย



	BIT7						BIT0	
00H	CH	10 SECONDS			SECONDS			00-59
	0	10 MINUTES			MINUTES			00-59
	0	12/24	10 HR A/P	10 HR	HOURS			01-12 00-23
	0	0	0	0	0	DAY		1-7
	0	0	10 DATE		DATE			
	0	0	0	10 MONTH	MONTH			01-12
	10 YEAR				YEAR			00-99
07H	OUT	0	0	SQWE	0	0	RS1	RS0

รูปที่ 2.24 รีจิสเตอร์ภายในไอซีฐานเวลา DS1307 [6]

ที่แอดเดรส 07H เป็นรีจิสเตอร์ควบคุมการทำงานของ SQW/OUT โดยมีรายละเอียดดังนี้ OUT(Out control): ใช้ควบคุมเอาต์พุตSQWE(Square Wave Enable): ใช้ควบคุมออสซิลเลเตอร์ภายใน DS1307 โดยถ้าบิตนี้เป็น “1” จะเป็นการเปิดออสซิลเลเตอร์RS(Rate Select): ใช้ควบคุมความถี่ของ Square Wave เมื่อเปิดการทำงานของออสซิเลเตอร์ โดยสามารถปรับเปลี่ยนความถี่ได้ 4 ความถี่ด้วยกันดังแสดงในตารางที่ 2.3

ตารางที่ 2.3 การควบคุมความถี่ออสซิลเลเตอร์ด้วยการเซตบิต RS1, RS0

RS1	RS0	SQW OUTPUT FREQUENCY
0	0	1 Hz
0	1	4.096 กิโลเฮิรตซ์
1	0	8.192 กิโลเฮิรตซ์
1	1	32.768 กิโลเฮิรตซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.5 การสื่อสารผ่านพอร์ตอนุกรม

เป็นวิธีการอินเตอร์เฟสของคอมพิวเตอร์วิธีหนึ่งเพื่อทำการเชื่อมต่อกับอุปกรณ์อื่นๆ ซึ่งวิธีนี้จะเป็นการติดต่อแบบอนุกรมผ่านทางพอร์ตอนุกรมการส่งข้อมูลทางซีเรียลพอร์ตนั้นในเครื่องคอมพิวเตอร์จะใช้มาตรฐาน RS-232 ซึ่งเป็นมาตรฐานการส่งข้อมูลระยะทางไม่เกิน 15 เมตร ส่วนชิพที่ใช้ในการควบคุมการทำงานนั้นนิยมใช้ IC#8250 UART หรือตัวที่พัฒนามาใหม่อย่างเช่น IC#16550 UART ซึ่งมีการทำงานที่ดีขึ้น เราจึงต้องทำความเข้าใจมาตรฐานที่ใช้อยู่เพื่อนำไปประยุกต์ และใช้งานตามความต้องการได้

2.12.5.1 มาตรฐานพอร์ต RS-232

การเชื่อมต่อผ่านพอร์ตขนานจะทำได้เร็วกว่าพอร์ตอนุกรมทั้งนี้เป็นเพราะการส่งแบบขนานเป็นการพาสัญญาณข้อมูลพร้อมกันครั้งละ 8 บิต ดังนั้นจะต้องใช้สายสัญญาณมาก 8-10 เส้น และระยะทางในการส่งไม่เกิน 4.5 เมตร สำหรับสายนำสัญญาณจากพอร์ตอนุกรมจะใช้เพียงสายนำข้อมูล สายสัญญาณนาฬิกา และGround เพียง 2-3 เส้น เท่านั้น โดย ที่ประเภทของการสื่อสารแบบอนุกรมแบ่งตามลักษณะสัญญาณในการส่งแบ่งได้ 2 แบบ คือ

1. การสื่อสารแบบซิงโครนัส(Synchronous) เป็นการสื่อสารข้อมูลโดยใช้สัญญาณนาฬิกา ในการควบคุมจังหวะของการรับส่งสัญญาณ
2. การสื่อสารแบบอะซิงโครนัส(Asynchronous) เป็นการสื่อสารที่ใช้สายข้อมูลเพียงตัว เดียวจะใช้ รูปแบบของการส่งข้อมูล(บิต Pattern) เป็นตัวกำหนดว่าส่วนไหนเป็นส่วน เริ่มต้นข้อมูล ส่วนไหนเป็นตัวข้อมูล ส่วนไหนจะเป็นตัวตรวจสอบความถูกต้องของข้อมูล และส่วนไหนเป็นส่วนปิดท้ายของข้อมูล โดยต้องกำหนดให้สัญญาณนาฬิกาเท่ากันทั้งภาคส่ง และภาครับ

คุณสมบัติทางไฟฟ้าของซีเรียลพอร์ตตามมาตรฐาน RS-232C พอสรุปได้ดังนี้

1) ลอจิก ‘0’ หรือ “Space” มีค่า +3 โวลต์ ถึง +25 โวลต์

2) ลอจิก ‘1’ หรือ “Mark” มีค่า -3 โวลต์ ถึง -25 โวลต์

3) ช่วง +3 โวลต์ ถึง –3 โวลต์ เป็นช่วง Undefined

4) Open Circuit Voltage เมื่อเทียบกับGroundต้องไม่เกิน 25 โวลต์

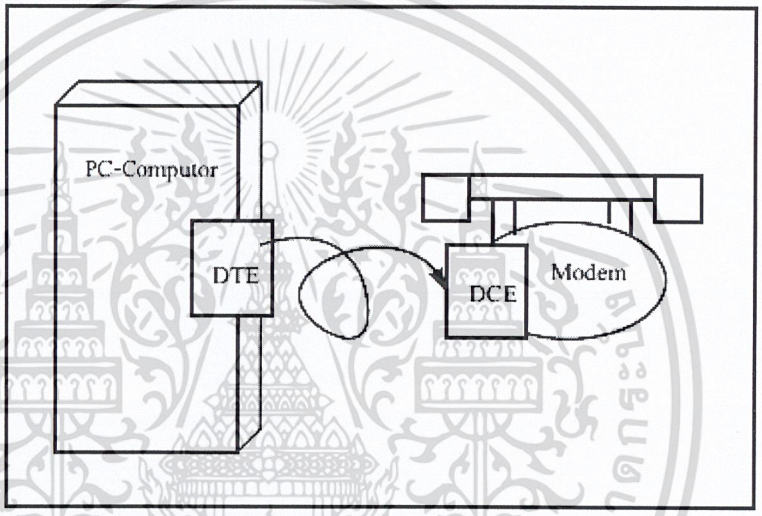
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) Short Circuit Current ต้องไม่เกิน 500 mA ซึ่ง Driver ต้องสามารถรองรับได้

เพื่อป้องกันไม่ให้อุปกรณ์ส่งข้อมูลภายในสายเส้นเดียวกัน อุปกรณ์สื่อสารในมาตรฐานพอร์ต RS-232 จึง จำแนกออกเป็น 2 ประเภทคือ

1) DTE(Data Terminal Equipment) คือ อุปกรณ์สำหรับส่งข้อมูล เช่น คอมพิวเตอร์

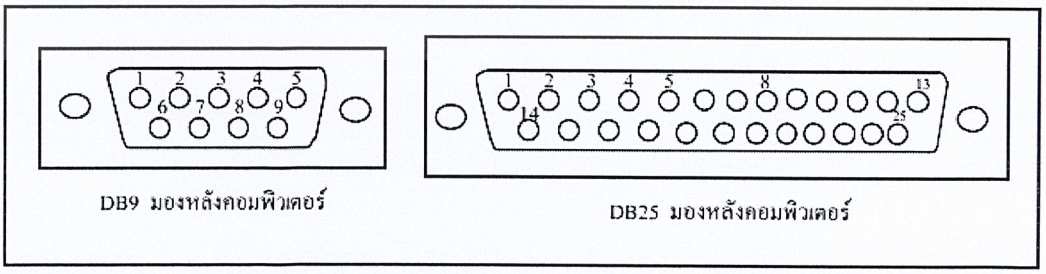
2) DCE(Data Communication Equipment) คือ อุปกรณ์สำหรับทำการติดต่อ เช่น โมเด็ม



รูปที่ 2.25 อุปกรณ์สื่อสารในมาตรฐานพอร์ต RS-232[7]

2.12.5.2 สายสัญญาณผ่านพอร์ต RS-232

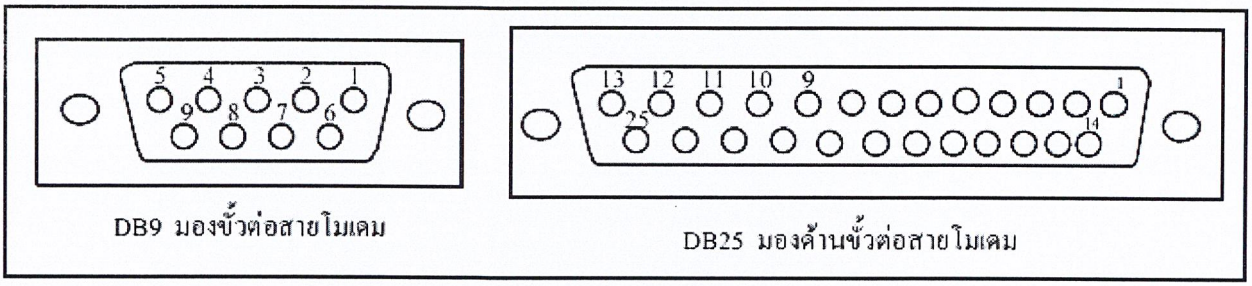
RS-232 ฝั่งคอมพิวเตอร์(DTE) จะเป็นขั้วต่อ DB9 หรือ DB25 ตัวผู้ ดังรูปที่ 2.26



รูปที่ 2.26 ตำแหน่งขาพอร์ต DB9 และ DB25[7]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้วต่อกับอุปกรณ์ภายนอก(DCE) จะเป็นขั้ว DB9 หรือ DB25 ตัวเมีย ดังรูปที่ 2.27



รูปที่ 2.27 ตำแหน่งขาของสายเชื่อมต่อพอร์ตอนุกรม[7]

2.12.5.3 ชื่อและหน้าที่ของสัญญาณพอร์ตอนุกรม

ตารางที่ 2.4 รายละเอียดของขาสัญญาณพอร์ตอนุกรม

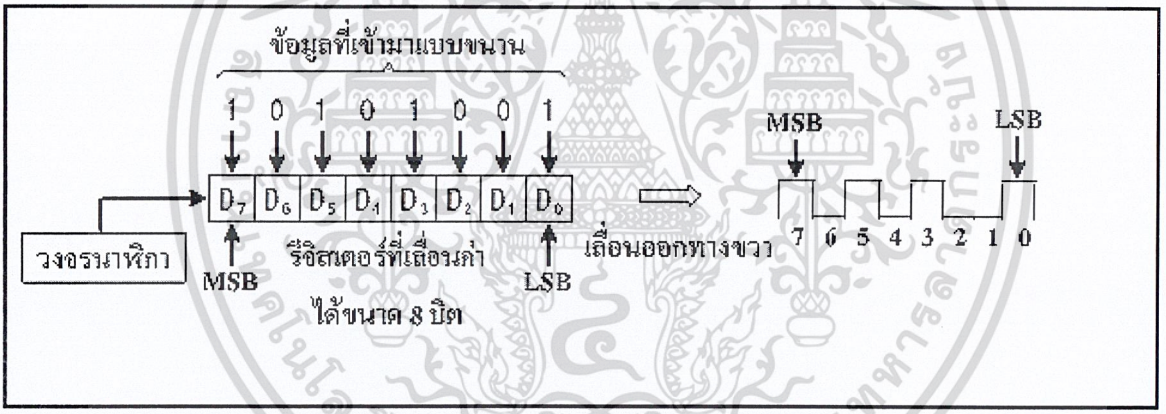
DB25	DB9	ชื่อสัญญาณ	ลักษณะ	หน้าที่
8	1	DCD : Data Carrier Detect	อินพุท	Active เมื่อโมเด็มส่งสัญญาณมา
3	2	RxD : Received Data	อินพุท	รับข้อมูลอนุกรมเข้าสู่คอมพิวเตอร์
2	3	TxD : Transmitted	เอ้าท์พุท	ส่งข้อมูลออกจากคอมพิวเตอร์
20	4	DRT : Data Terminal Ready	เอ้าท์พุท	แจ้งอุปกรณ์ปลายทางรู้ว่าจะติดต่อด้วย
7	5	GND : Ground	-	
6	6	DSR : Data Set Ready	อินพุท	ต่อกับ DTR ของอุปกรณ์ภายนอก เพื่อตรวจสอบการเชื่อมต่อ
4	7	RTS : Request To Send	เอ้าท์พุท	ขอให้อุปกรณ์ส่งข้อมูลเข้าที่ CTS ของคอมพิวเตอร์
5	8	CTS : Clear To Send	อินพุท	เมื่อรับสัญญาณ RTS แล้วจึงส่งข้อมูลจาก TxD ได้
22	9	RI : Ring Indicator	อินพุท	แสดงการเรียกเข้าของสัญญาณโทรศัพท์มายังโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

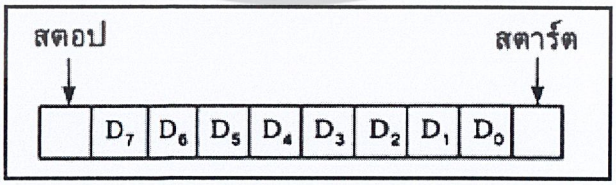
2.12.5.4 มาตรฐานการรับส่งข้อมูลแบบอะซิงโครนัส(UART)

UART ย่อมาจาก Universal Asynchronous Receiver-Transmitter หมายถึง อุปกรณ์ที่ทำหน้าที่รับ และส่งข้อมูลแบบอะซิงโครนัสนั่นเอง สำหรับการสื่อสารอนุกรมบนคอมพิวเตอร์แล้ว UART ถือว่าเป็นหัวใจของการสื่อสารอนุกรม

โดยที่หน้าที่หลักของ UART คือ แปลงข้อมูลแบบขนานเป็นข้อมูลแบบบอนุกรมแบบอะซิงโครนัส และทำการส่งข้อมูลออกไป นอกจาก UART จะมีหน้าที่ในการแปลงข้อมูลแล้ว มันยังมีหน่วยควบคุม และหน่วยตรวจสอบการทำงานด้วย ในการส่งข้อมูลขนาด 8 บิตแบบอนุกรมจะต้องมีบิตสตาร์ท(มีค่าเป็น 0) เพื่อบอก UART ว่ามีข้อมูลกำลังเข้ามา และบิตสต๊อป(มีค่าเป็น 1)เพื่อบอก UART ว่าข้อมูลส่งหมดแล้วซึ่งจะทำให้ข้อมูล ที่ส่งออกไปมีขนาด 10 บิต เราเรียกความเร็วในการส่งข้อมูลของ UART ว่า บอดเรท ซึ่งจะบอกจำนวนบิตที่ส่งออกไปในเวลา 1 วินาที

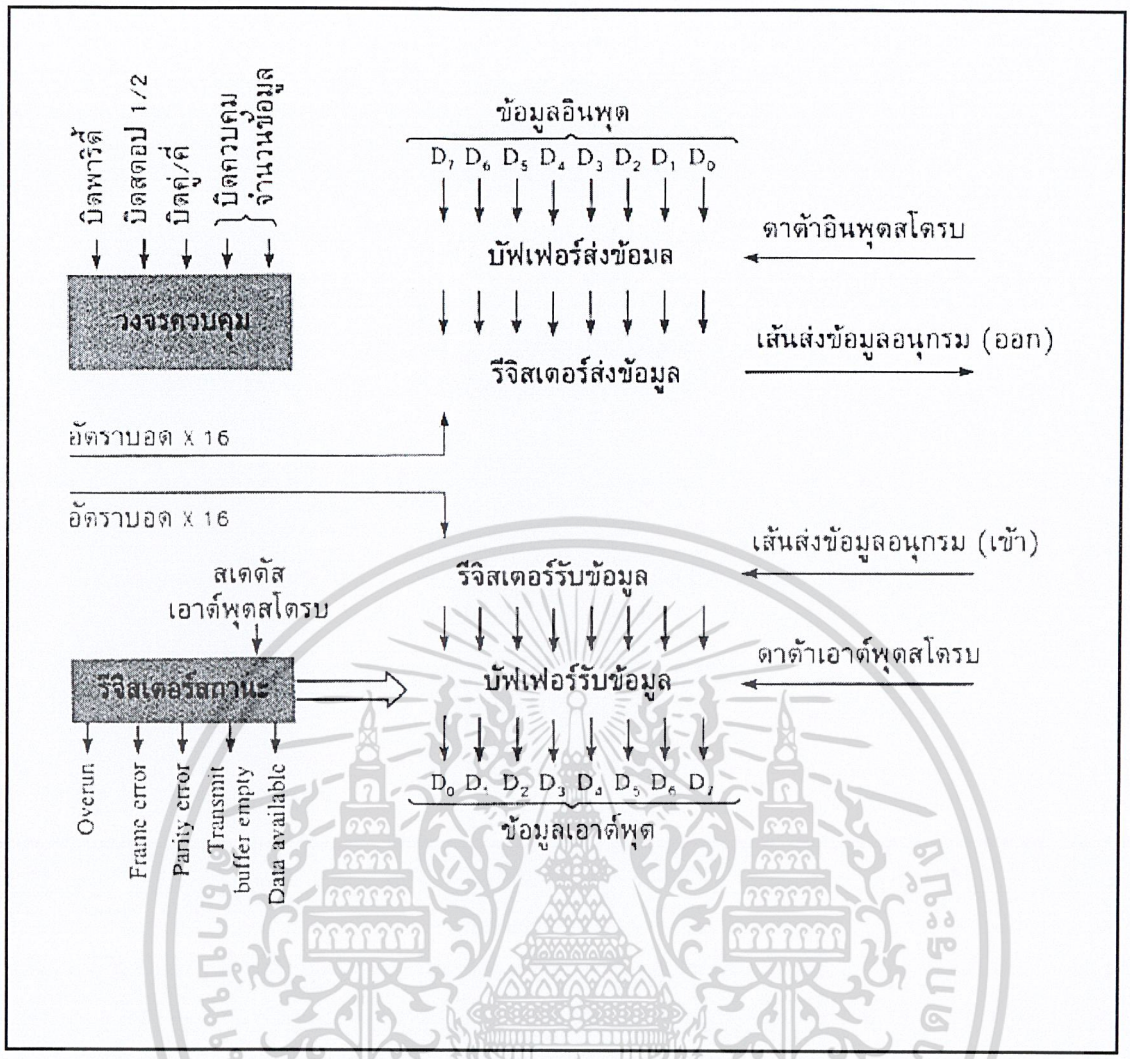


รูปที่ 2.28 การแปลงข้อมูลแบบขนานเป็นข้อมูลอนุกรม[7]



รูปที่ 2.29 เวิร์ดข้อมูลขนาด 8 บิตกับบิตสตาร์ทและบิตสต๊อปที่ใช้ในการโอนย้ายข้อมูลแบบอนุกรม[7]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.30 บล็อกไดอะแกรมของ UART[7]

จากรูปที่ 2.30 สามารถแบ่งได้เป็น 4 ส่วนใหญ่ ๆ คือ

- ส่วนที่ทำหน้าที่ส่งข้อมูลของ UART : แบ่งออกได้เป็น 2 ส่วน คือ transmitted data output buffer กับ รีจิสเตอร์ส่งข้อมูล(transmit register) โดยรีจิสเตอร์ส่งข้อมูลจะมี หน้าที่เลื่อนข้อมูลส่งออกไปยังเส้นส่งข้อมูลอนุกรม โดยเริ่มจากสตาร์ทบิต ตามด้วย D0-D7 และสต๊อปบิต ถ้าสัญญาณ data input strobe เปลี่ยนเปลี่ยน จาก 1 เป็น 0 ข้อมูลขนาด 8 บิตที่จะส่ง จะถูกนำไปเก็บใน transmitted-data output buffer และถ้าสัญญาณ data input strobe เปลี่ยนจาก 0 เป็น 1 การส่ง ข้อมูลจะเริ่มขึ้น
- ส่วนที่ทำหน้าที่รับข้อมูลของ UART จะเหมือนเป็นภาพสะท้อนของส่วนที่ทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่ส่งข้อมูล เมื่อมีสตาร์ทบิต เข้ามา ข้อมูลที่เข้ามาทางเส้นรับข้อมูลจะถูกเลื่อนเข้าไปเก็บในรีจิสเตอร์รับข้อมูล(receive register) จนครบ เมื่อมีสัญญาณ data output strobe ข้อมูลในรีจิสเตอร์นี้จะถูกนำไปเก็บในบัฟเฟอร์รับข้อมูล (received-data output buffer) ทั้งรีจิสเตอร์ส่งข้อมูล และรีจิสเตอร์รับข้อมูล จะได้รับสัญญาณนาฬิกาขนาด 16 หรือ 64 เท่า ของบอดเรท

0	0	Overrun	Frame error	Parity error	Transmit buffer empty	Data available	0

รูปที่ 2.31 รีจิสเตอร์สถานะของ UART[7]

ตารางที่ 2.5 ความหมายของบิตต่างๆในรีจิสเตอร์สถานะของ UART

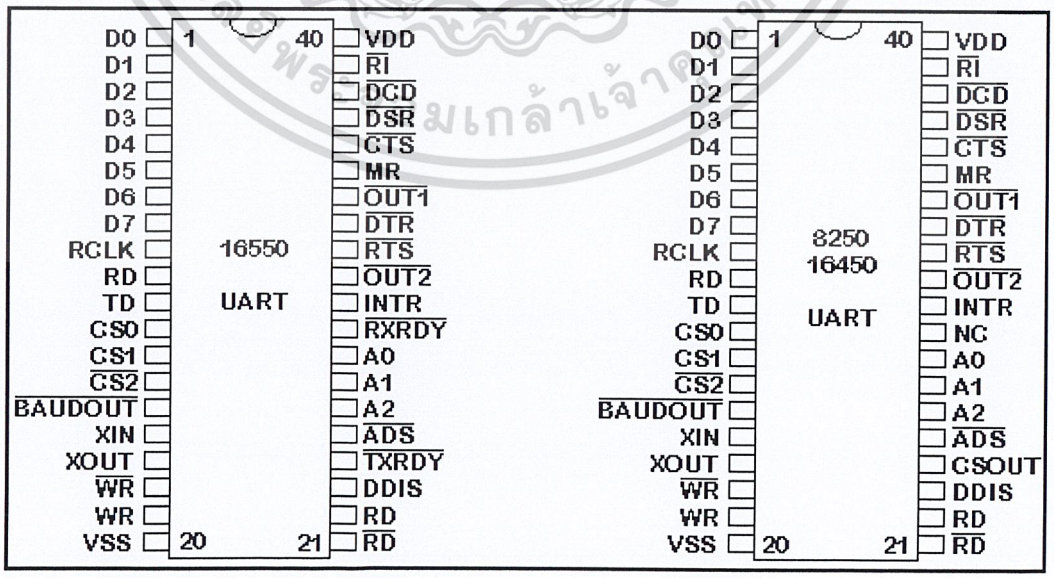
OR(Over Run)	บิตนี้จะมีค่าเป็น 1 เมื่อข้อมูลใหม่เข้ามาทับข้อมูลเก่าโดยที่ข้อมูลเก่ายังไม่ได้ถูกนำไปเก็บในบัฟเฟอร์รับข้อมูล
FE(Framing Error)	บิตนี้จะมีค่าเป็น 1 ถ้า UART ไม่พบสต๊อปบิต ซึ่งอาจเป็นเพราะว่า UART ไม่ได้อ่านสตาร์ทบิต ในตำแหน่งที่ถูกต้อง
PE(Parity Error)	บิตนี้จะมีค่าเป็น 1 เมื่อ UART พบว่าข้อมูลที่ได้รับมีพาริตี้ไม่ถูกต้อง
TBE(Transmit Buffer Empty)	บิตนี้จะมีค่าเป็น 1 เมื่อ UART ได้ทำการส่งข้อมูลออกไปแล้วซึ่งทำให้เราสามารถนำข้อมูลชิ้นต่อไปที่จะส่งเก็บลงในบัฟเฟอร์ส่งข้อมูลได้
DA(Data Available)	บิตนี้จะมีค่าเป็น 1 เมื่อ UART ได้รับข้อมูลใหม่เข้ามา และไมโครโปรเซสเซอร์สามารถทำการอ่านข้อมูลนี้ได้ โดยส่งสัญญาณ data output strobe

- ส่วนที่เป็นวงจร ภายในUARTจะมีส่วนของวงจรสร้างอัตราการถ่ายทอดข้อมูลแบบโปรแกรมได้ (Programmable Buadrate Generator) โดยการกำหนดค่าตัวหารให้กับ สัญญาณนาฬิกาของ UART โดยตัวหารนี้มีขนาด 16 บิตดังนั้นจึงกำหนดตัวหารให้อยู่ ในช่วง 10 – 65,535 บิต UART สามารถรับส่งข้อมูลได้ทั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบฮาล์ฟดูเพล็กซ์(Half Duplex) และฟูลดูเพล็ก(Full Doplex) โดยการส่งแบบฮาล์ฟดูเพล็กเป็นการส่งแบบทิศทางเดียว ส่วนการส่ง แบบฟูลดูเพล็กนั้น สามารถรับ และส่งข้อมูลได้ในคราวเดียวกัน ชนิดของ UART ในเครื่องคอมพิวเตอร์ทั่วไปมี UART ที่ใช้กันอยู่ 2 เบอร์ คือ

1. IC#8250 ซึ่งเป็น UART มาตรฐานที่มีใช้กันมายาวนาน UART เบอร์ นี้จะมีบัฟเฟอร์สำหรับรับและส่งข้อมูลตำแหน่งเดียวกัน ทำให้การรับ และส่งข้อมูลถูกจำกัดความเร็วอยู่ที่ 57.6 กิโลบิตต่อวินาทีเท่านั้น แต่ UART เบอร์นี้ก็ ถือว่าเป็นต้นแบบของ UART ที่ใช้ในคอมพิวเตอร์ โดยคอมพิวเตอร์ทุกๆรุ่นจะต้องสนับสนุนการทำงานตาม รูปแบบของ UART เบอร์นี้
2. IC#16450 ความสามารถรับส่งข้อมูลได้ที่ความเร็ว 115,200 บิตต่อวินาที และเพิ่มรีจิสเตอร์สำหรับพักข้อมูลสำหรับ UART นอกจากนั้นยังเพิ่มส่วนของชิพรีจิสเตอร์ แบบ FIFO (First In First Out) ขนาด 16 ไบต์เข้าไป ทำให้สามารถสนับสนุนความเร็วใน การรับส่งข้อมูลที่ 256 กิโลบิตต่อวินาทีได้ โดยคอมพิวเตอร์ในปัจจุบันใช้ UART เบอร์นี้ หรือใหม่กว่าเช่น เบอร์ TL 16C750 ซึ่งมีรีจิสเตอร์แบบ FIFO ขนาด 64 ไบต์ ทำงานได้ที่ระดับแรงดัน +5 โวลต์ และ +3 โวลต์ มีโหมดประหยัดพลังงาน สามารถรับส่งข้อมูลได้ที่ความเร็ว 1 เมกะบิตต่อวินาทีเมื่อใช้สัญญาณนาฬิกา 16 เมกะเฮิรตซ์

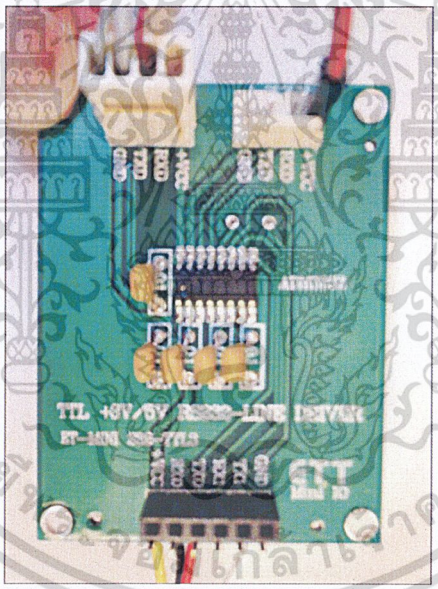


รูปที่ 2.32 พินของ 16550, 16450 & 8250 UARTs[7]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

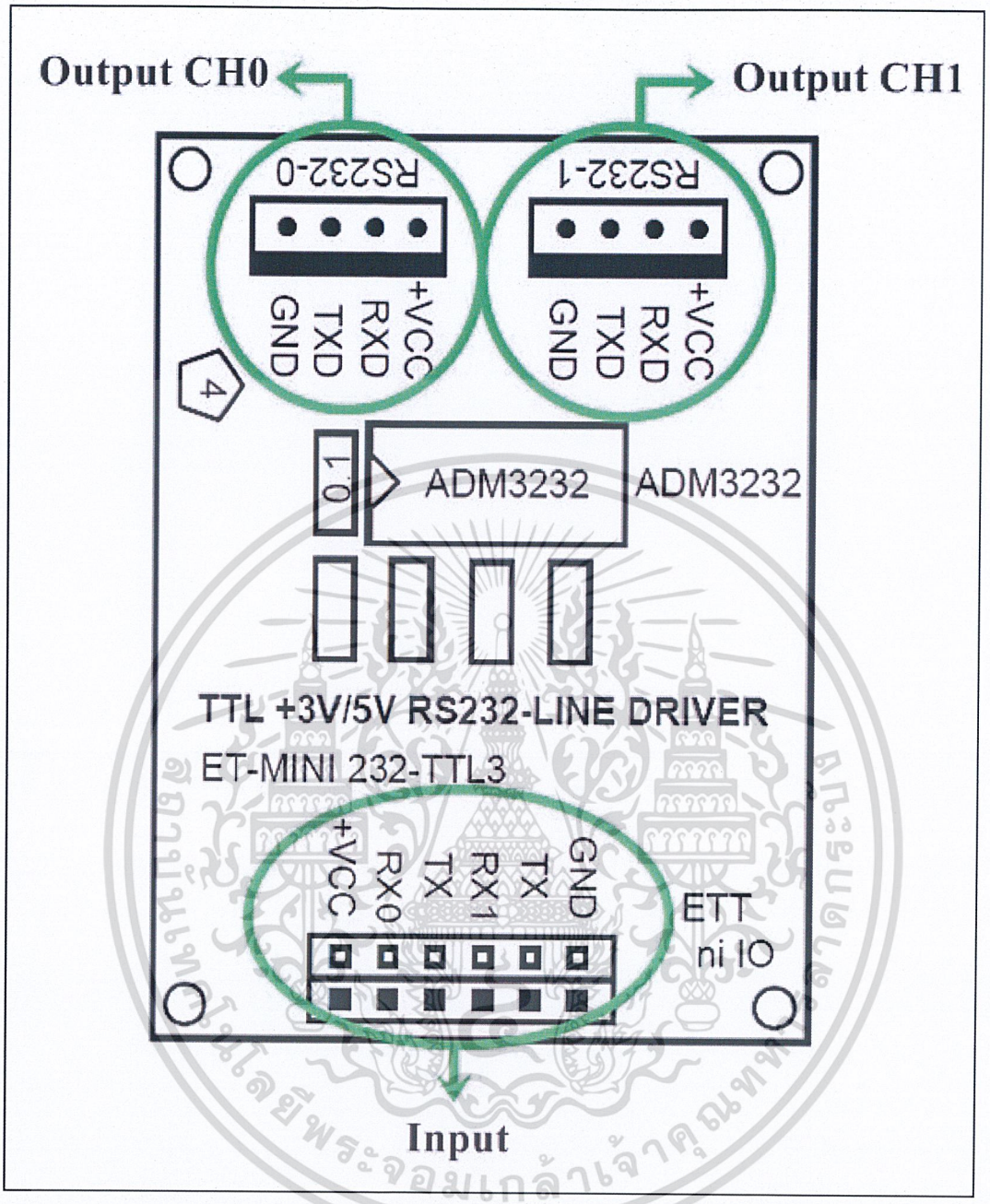
2.12.6 MODULE ET-MINI 232-TTL3

Module ET-MINI 232-TTL3 เป็นชุด Line Driver ในการสื่อสารแบบ RS-232 ทำงานที่แรงดัน 3โวลต์-5โวลต์ ซึ่งจะทำหน้าที่เปลี่ยนระดับสัญญาณทางไฟฟ้าของขาสัญญาณ รับ-ส่งข้อมูลแบบ TTL ของ MCU ให้เป็นระดับสัญญาณทางไฟฟ้าแบบ RS-232(±12โวลต์) ซึ่งใน Module นี้ จะสามารถ ต่อใช้งานได้ 2 แชนแนล โดยที่ขั้วต่อทางด้าน Input จะต้องต่อขา Tx และ Rx เข้ากับขา Tx และ Rx ของ MCU ตามลำดับ ขา Tx และ Rx จะต้องต่อในแชนแนลเดียวกัน (Tx0:Rx0 ,Tx1:Rx1) ส่วนขา +VCC และ GND ก็ต่อเข้ากับไฟเลี้ยง MCU ได้เลย ในส่วนขั้วต่อทางด้าน Output ถ้าผู้ใช้ต่อ Input เข้ามาทางแชนแนลไหนก็ให้ต่อ Output ของแชนแนลนั้นไปใช้งาน โดยจะต้องต่อขา Rx เข้ากับขา Tx และต่อขา Tx เข้ากับขา Rx ของอุปกรณ์ที่จะนำมาสื่อสารกัน และต่อขา GND เข้ากับ GND ของอุปกรณ์ที่นำมาสื่อสารด้วย ส่วน +VCC ไม่ต้องต่อ



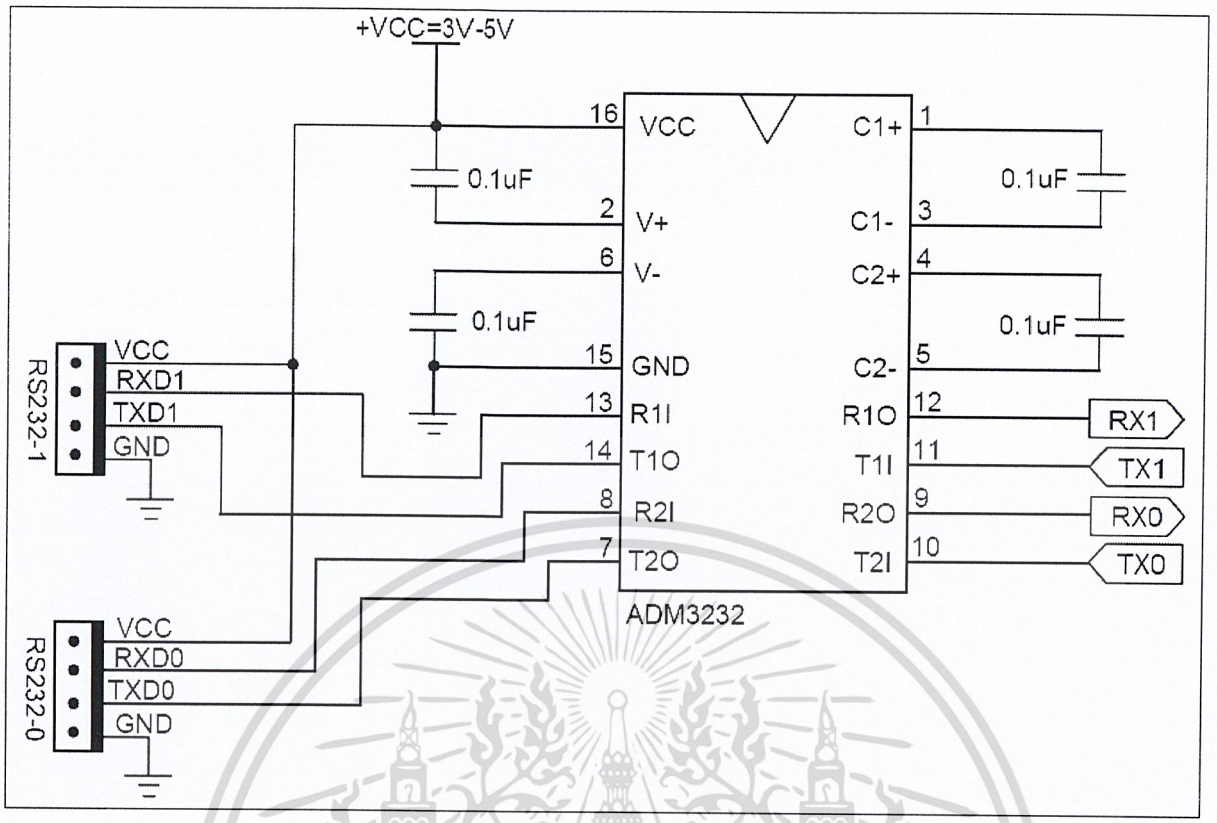
รูปที่ 2.33 Module ET-MINI 232-TTL3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.34 โครงสร้าง ET-MINI 232-TTL3 และตำแหน่งขา Port ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.35 วงจร ET-MINI 232-TTL3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

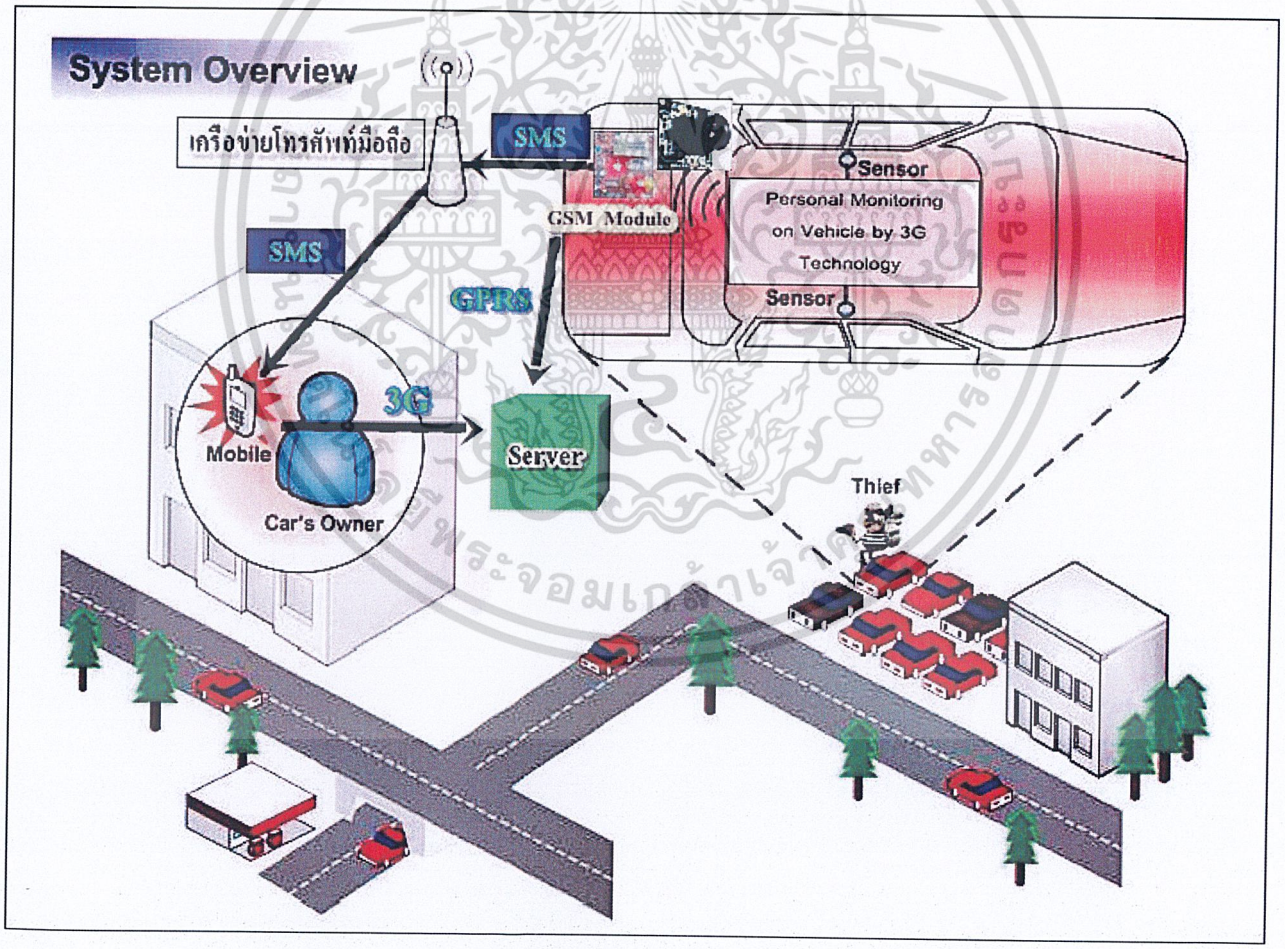
บทที่ 3

การออกแบบการทดลอง

ในการออกแบบระบบตรวจสอบบุคคลภายในรถนั้น จะประกอบไปด้วย

- ส่วนภายในรถยนต์
- ส่วนของเซิร์ฟเวอร์ที่ใช้ในการเก็บฐานข้อมูลรูปภาพ และอื่นๆ
- ส่วนของผู้ใช้จะมีโทรศัพท์มือถือที่สามารถรองรับการเชื่อมต่อในระบบ GPRS

โดยจะใช้สัญญาณ GPRS ในการเชื่อมต่อกับเซิร์ฟเวอร์และในการส่ง SMS เพื่อแจ้งเตือนให้เจ้าของรถยนต์ทราบโดยจะส่งผ่านเครือข่ายของผู้ให้บริการเครือข่ายโทรศัพท์



รูปที่ 3.1 ภาพรวมของระบบ

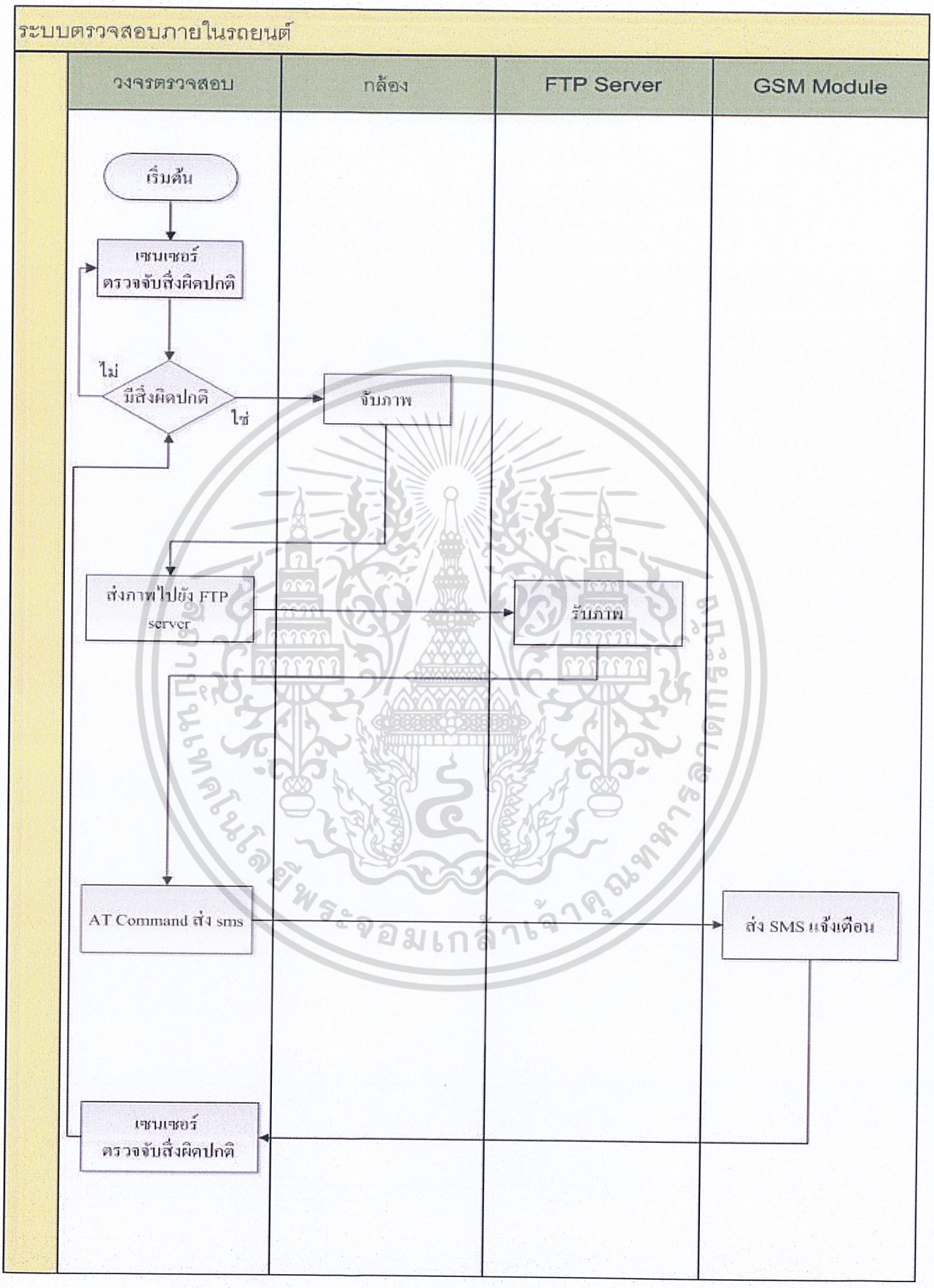
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 ระบบตรวจสอบบุคคลภายในรถยนต์

การทำงานของระบบตรวจสอบภายในรถยนต์ จะทำงานด้วยไมโครคอนโทรลเลอร์ ในการทำหน้าที่ควบคุมอุปกรณ์ต่างๆที่ติดตั้งอยู่ภายในรถยนต์ ระบบจะเริ่มตรวจสอบบุคคลที่จะขึ้นมายังรถด้วยเซนเซอร์ตรวจจับการเคลื่อนไหว(Motion Sensor) ที่ติดตั้งอยู่บนรถจะทำการตรวจสอบว่ามีการเคลื่อนไหวเกิดขึ้นบริเวณที่มีการติดตั้งไว้บนรถหรือไม่ ถ้ามีจะมีการจับภาพส่งขึ้นเซิร์ฟเวอร์พร้อมกันกับการจัดเก็บภาพลงบน SD Card เมื่อได้รับรูปจากกล้อง ไมโครคอนโทรลเลอร์จะทำการเชื่อมต่อกับทาง FTP Server โดยใช้พอร์ตเบอร์ 21 ในการเชื่อมต่อด้วยการนำไฟล์รูปภาพที่ได้รับขึ้นไปวางไว้บน FTP Server เมื่อทำการส่งไฟล์รูปขึ้นเซิร์ฟเวอร์เรียบร้อยแล้ว จะมีการส่ง SMS ไปเตือนยังผู้ใช้ว่า มีสิ่งผิดปกติเกิดขึ้นบนรถยนต์ของเจ้าของรถ เมื่อเจ้าของรถได้รับ SMS แจ้งเตือนแล้ว เราจะสามารถเข้าไปตรวจสอบรูปภาพบนเซิร์ฟเวอร์ได้ด้วยการใช้ Website ของเซิร์ฟเวอร์ ด้วยระบบ GPRS เพื่อเป็นการตรวจสอบรูปภาพที่ได้ว่าเป็น การโจรกรรม หรือเป็นอย่างอื่นก็ตาม

จากข้างต้นที่กล่าวมา สามารถที่จะนำมา อธิบาย Flow Chart การทำงานของระบบตรวจสอบภายในรถยนต์ ดังรูปที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 Flow Chart การทำงานของระบบตรวจสอบภายในรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การตรวจจับสิ่งผิดปกติ

การตรวจจับสิ่งผิดปกตินั้น จะใช้ PIR Sensor ในการตรวจสอบสิ่งผิดปกติภายในรถยนต์ เมื่อมีสิ่งผิดปกติเกิดขึ้นบนรถยนต์ เซนเซอร์จะทำการจับการเคลื่อนไหวที่ออกมายังขาสัญญาณของเซนเซอร์ เพื่อเป็นการบ่งบอกถึงสิ่งผิดปกติให้ไมโครคอนโทรลเลอร์ได้รับทราบ

3.3 การจับภาพด้วย Camera Module

ในส่วนของกล้องจะใช้เป็น Camera Module ในการส่งภาพผ่านทางพอร์ต UART ของไมโครคอนโทรลเลอร์ จะใช้ภาษาซีในการโปรแกรมให้กล้องจับภาพ โดยไฟล์รูปภาพที่ได้จะเป็นไฟล์ภาพตระกูล .jpeg

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง Code ที่เขียนด้วยภาษาซี ที่เขียนขึ้นเพื่อสั่งให้กล้อง Camera Module ถ่ายภาพโดยอัตโนมัติ

```
unsigned char reset [] =  { 0xAA, 0x08, 0x01, 0x00, 0x00, 0xFF };
unsigned char sync [] =  { 0xAA, 0x0D, 0x00, 0x00, 0x00, 0x00 };
unsigned char ack  [] =  { 0xAA, 0x0E, 0x0D, 0x02, 0x00, 0x00 };
unsigned char init [] =  { 0xAA, 0x01, 0x00, 0x07, 0x07, 0x07 };
// JPEG, 80*64
/*
0x01, 0x01  80 * 64
0x03, 0x03  160 * 128
0x05, 0x05  320 * 240
0x07, 0x07  640 * 480
*/
unsigned char pack [] =  { 0xAA, 0x06, 0x08, 0x00, 0x02, 0x00 };
unsigned char snap [] =  { 0xAA, 0x05, 0x00, 0x00, 0x00, 0x00 };
unsigned char getp [] =  { 0xAA, 0x04, 0x01, 0x00, 0x00, 0x00 };
unsigned char ack1 [] =  { 0xAA, 0x0E, 0x00, 0x00, 0x00, 0x00 };
unsigned char endp [] =  { 0xAA, 0x0E, 0x00, 0x00, 0xF0, 0xF0 };
unsigned char nak  [] =  { 0xAA, 0x0F, 0x00, 0x00, 0x00, 0x00 };
```

จากตัวอย่าง Code ข้างต้น สามารถที่จะอธิบายได้ว่า เป็นการตั้งค่าขนาดของไฟล์รูปภาพที่จะทำการถ่ายรูปออกมาผ่านทาง Camera Module โดยจะมีการกำหนดขนาดไว้ 4 ขนาด ได้แก่ ภาพขนาด 80x60 160x124 320x240 640x480 โดยในตัวอย่างโปรแกรมข้างต้นจะกำหนดขนาดไว้ที่ 640x480

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 เซิร์ฟเวอร์ที่ใช้เก็บข้อมูล

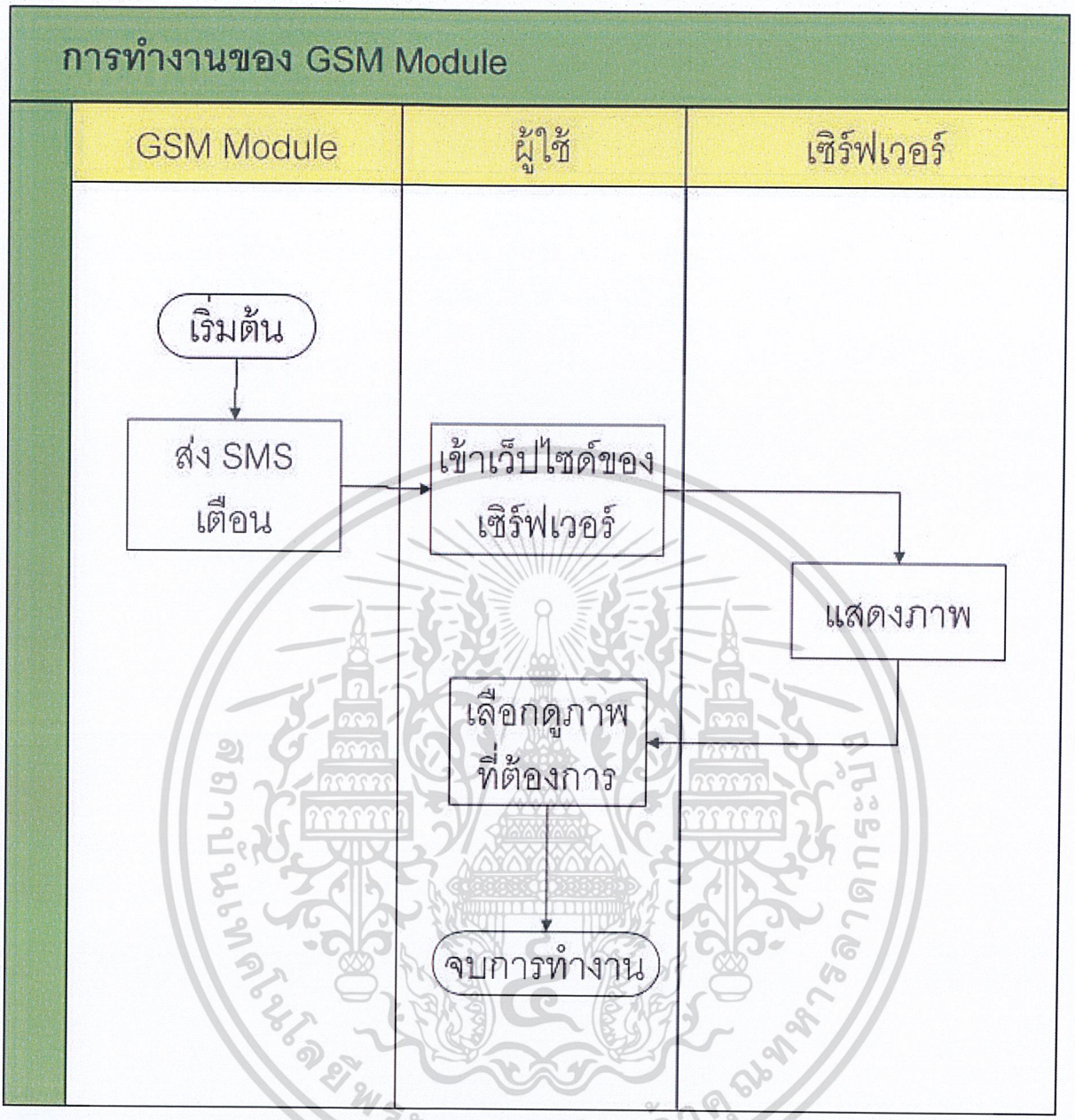
ในส่วนของเซิร์ฟเวอร์นั้น จะใช้ FTP Server ในการรับไฟล์ภาพที่ได้อัพโหลดขึ้นมาจากระบบตรวจสอบในรถยนต์ โดยเซิร์ฟเวอร์ที่ใช้นั้น จะทำการรับไฟล์รูปภาพจากระบบตรวจสอบในรถยนต์เพียงอย่างเดียว ซึ่งเซิร์ฟเวอร์ที่ใช้ จะใช้ระบบปฏิบัติการของ Ubuntu ด้วย Apache จะรับและส่งไฟล์ผ่านทางพอร์ต 21

3.5 การทำงานของ GSM Module

การทำงานของ GSM Module เจ้าของรถจะได้รับ SMS จากระบบตรวจสอบเมื่อมีสิ่งผิดปกติเกิดขึ้น เมื่อได้รับการเตือนภัยจากระบบแล้ว จะสามารถทำการตรวจสอบรูปภาพสิ่งผิดปกติที่เกิดขึ้นภายในรถได้ด้วยการเข้าสู่ Website ของเซิร์ฟเวอร์ด้วย GPRS ด้วยมือถือของตัวเองเพื่อใช้เป็นหลักฐาน และสามารถแก้ไขเหตุการณ์ได้อย่างทันท่วงที

จากข้างต้นที่กล่าวมา สามารถที่จะนำมา อธิบาย Flow Chart การทำงานของ GSM Module ดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 Flow Chart การทำงานของ GSM Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง Code ที่เขียนด้วยภาษาซี ที่เขียนขึ้นเพื่อให้ GSM Module เชื่อมต่อกับเซิร์ฟเวอร์

```
int8_t ftp_config(void)
                {
                uint8_t  nRet;

#ifdef   LCD
                        lcd_clear(2);
                        lcd_printf("at#fptserv");
#endif
                        nRet = at_Command(AT_FTPSERV);
                        if(nRet != OK)                  return 0;
                        _delay_ms(100);
#ifdef                  LCD
                        lcd_clear(2);
                        lcd_printf("at#fptuser");

#endif
                        nRet = at_Command(AT_FTPUN);
                        if(nRet != OK)                  return 0;
                        _delay_ms(100);
#ifdef   LCD
                        lcd_clear(2);
                        lcd_printf("at#fptpassword");
#endif
```

จากตัวอย่าง Code ข้างต้น สามารถที่จะอธิบายได้ว่า เป็นกระบวนการเข้าสู่ FTP Server โดยจะมีการใส่ Username กับ Password เพื่อที่จะสามารถเข้าถึงไฟล์ใน FTP Server ได้นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 การส่ง SMS แจ้งเตือนไปให้กับเจ้าของรถยนต์

การส่ง SMS นั้นจะเริ่มขึ้นเมื่อระบบตรวจสอบทำการตรวจสอบภาพว่ามีสิ่งผิดปกติเกิดขึ้นมาหรือไม่ด้วย Motion Sensor เมื่อตรวจพบว่ามีสิ่งผิดปกติ ระบบจะทำการอัพโหลดไฟล์รูปขึ้นเซิร์ฟเวอร์ก่อนถึงจะค่อยส่ง SMS แจ้งเตือนไปยังเจ้าของรถต่อไป

ตัวอย่าง Code ที่เขียนด้วยภาษาซี ที่เขียนขึ้นเพื่อส่งข้อความแจ้งเตือนไปยังเจ้าของรถยนต์

```
void sentsms(void)
                {
#ifdef   LCD
                        lcd_clear(3);
                        lcd_printf("Ready SMS");
#endif
                        printf("AT\r");
                        _delay_ms(2000);
#ifdef   LCD
                        lcd_clear(3);
                        lcd_printf("Stand By");
#endif
                        printf("AT+CMGF=1\r");
                        _delay_ms(2000);
/*#ifdef          LCD
                        lcd_clear(3);
                        lcd_printf("DTAC CENTER");
#endif
                        printf("AT+CSCA=\"+66816110400\"\r");
                        _delay_ms(2000);
#ifdef   LCD
                        lcd_clear(3);
                        lcd_printf("Dial Telephone");
#endif
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    printf("AT+CMGS=\"+66811700924\"\r");
                    _delay_ms(2000);
#ifdef   LCD
                    lcd_clear(3);
                    lcd_printf("Send SMS");
#endif
                    printf("Warning your Car");
                    _delay_ms(1000);
                    ser_putc(0x1A);
#ifdef   LCD
                    lcd_clear(3);
                    lcd_printf("Complete");
                    _delay_ms(1000);
#endif
                        }
```

จากตัวอย่าง Code ข้างต้น สามารถที่จะอธิบายได้ว่า การตั้งค่าการส่ง SMS ให้อยู่ในรูปโหมดตัวอักษร แล้วทำการระบุเบอร์โทรศัพท์ปลายทาง เพื่อที่จะส่งข้อความว่า "Warning your Car"

3.7 การส่งภาพขึ้นเซิร์ฟเวอร์

การส่งภาพขึ้นเซิร์ฟเวอร์นั้น จะใช้โปรโตคอล FTP ในการส่งภาพไปยังเซิร์ฟเวอร์จะทำการเชื่อมต่อโดยใช้ภาษาซี ผ่านทางพอร์ต 21 โดยจะมีการกำหนดค่าต่างๆ ก่อนที่จะสามารถอัพโหลดไฟล์ ขึ้นเซิร์ฟเวอร์ได้ เช่น ชื่อผู้ใช้ รหัสผู้ใช้โฮสต์ของเซิร์ฟเวอร์ พอร์ตที่ใช้ในการส่ง เป็นต้น เพื่อใช้ในการยืนยันเพื่อที่จะส่งไฟล์ขึ้นเซิร์ฟเวอร์นั่นเอง โดยการส่งภาพขึ้นเซิร์ฟเวอร์นั้น จะใช้เป็นระบบ GPRS ในการส่งไฟล์รูปภาพขึ้นเซิร์ฟเวอร์ผ่านทางอุปกรณ์ GSM Module

จากข้างต้นที่กล่าวมา สามารถที่จะนำมา อธิบาย Flow Chart การทำการอัพโหลดไฟล์ภาพไปยังเซิร์ฟเวอร์ดังรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.4 Flow Chart การทำการอัพโหลดไฟล์ภาพไปยังเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง Code ที่เขียนด้วยภาษาซี ที่เขียนขึ้นเพื่อส่งภาพขึ้นเซิร์ฟเวอร์

```
uint8_t ftp_Transfer(char *files)
        {
                uint8_t nRet,
                                text[100];
                        memset(text, 0, sizeof text);
                        sprintf(text,"%s\"%s\"%s",AT_FTPPUTFILE,files,AT_END);
#ifdef   LCD
                        lcd_clear(2);
                        lcd_printf("at#fptputname");
#endif
                        nRet = at_Command(text);
                //debug_response(nRet);
                        if(nRet != OK)                                          return 0;
                                        _delay_ms(100);
#ifdef   LCD
                        lcd_clear(2);
                        lcd_printf("at#fptputpath");
#endif
                        nRet = at_Command(AT_FTPPUTPATH);
                //debug_response(nRet);
                        if(nRet != OK)                                          return 0;
                                        _delay_ms(100);
```

จากตัวอย่าง Code ข้างต้น สามารถที่จะอธิบายได้ว่า เป็นการเรียกไฟล์ชื่อมาจากค่าต่าง ๆ ใน โปรแกรมเพื่อใช้เป็นการตั้งชื่อรูป

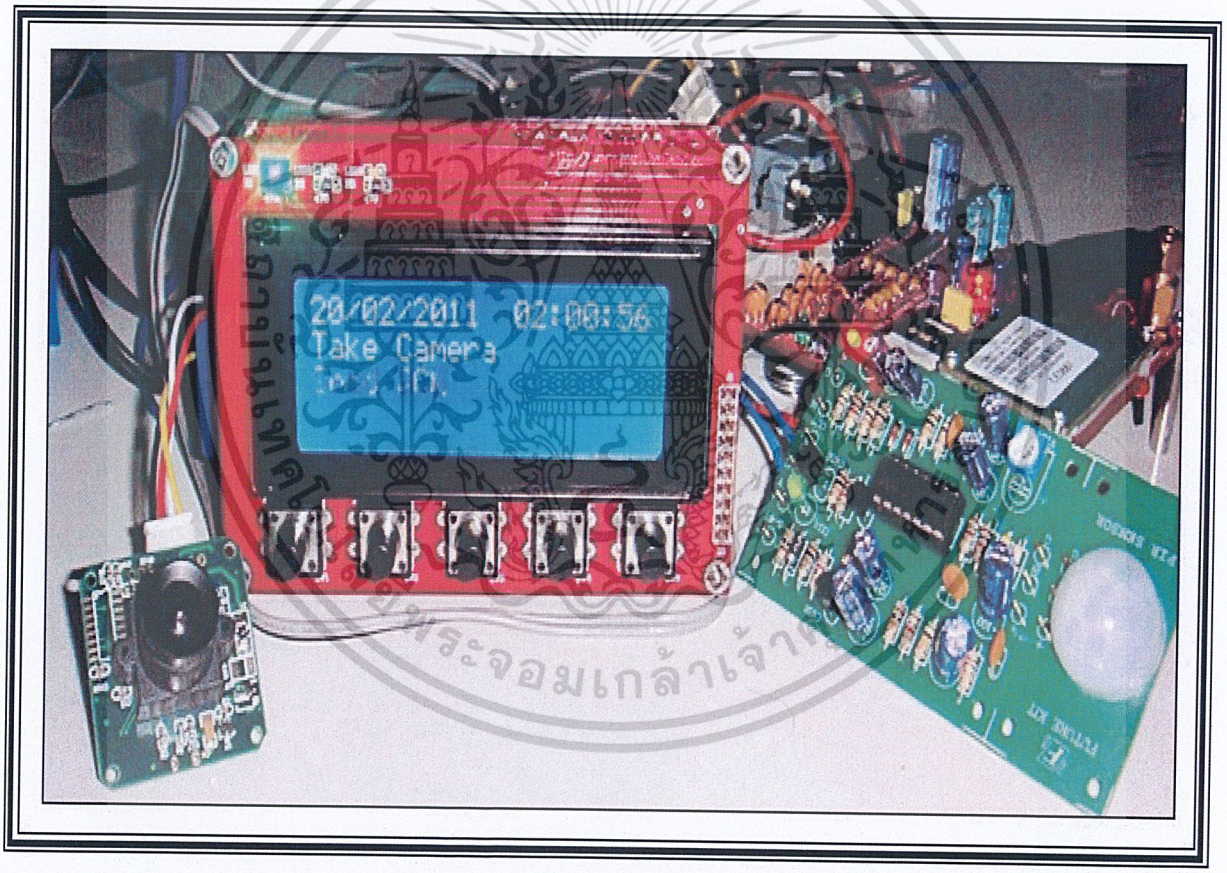
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดสอบ

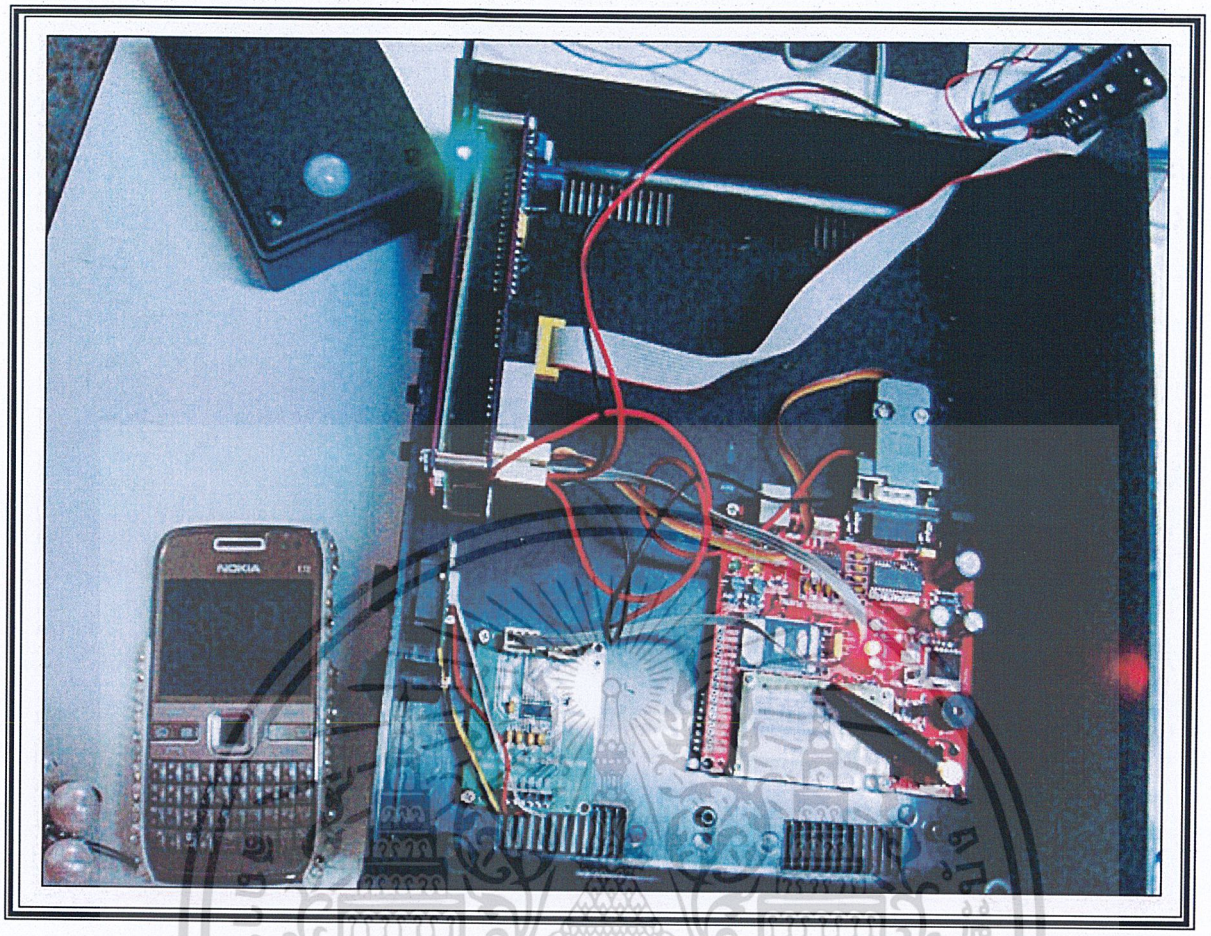
ในบทนี้เป็นการทดสอบการอัพโหลดภาพขึ้น FTP Server และส่ง SMS แจ้งเตือนเข้ามือถือ เนื่องจากโครงงานนี้ไม่มีผลการทดลองที่เป็นตัวเลข แต่ผลการทดลองนั้นได้มาจากการรันโปรแกรมจึงแสดงผลการทดสอบเป็นภาพ และมีขั้นตอนการทำงานแสดงผ่านทางจอLCD แบ่งเป็นส่วนต่างๆดังนี้

4.1 ฮาร์ดแวร์ที่ใช้ในระบบ



รูปที่4.1 ภาพรวมของโครงงาน(บอร์ด AVR ATmega128 ด้านหน้า)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.2 ภาพรวมของ โครงงาน(บอร์ด AVR ATmega128 ด้านบน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3 ภาพรวมโครงงาน เมื่อประกอบเสร็จเรียบร้อยพร้อมใช้งานจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดสอบโปรแกรม จะแสดงขั้นตอนผ่านทางหน้าจอ LCD

แบ่งการทดสอบโปรแกรมออกเป็น 6 ส่วนดังนี้

- การทดสอบการเปิดระบบ
- การทดสอบการเชื่อมต่อ GPRS โดยใช้ GSM Module
- การทดสอบการเชื่อมต่อกับ FTP Server พอร์ต 21
- การทดสอบตรวจสิ่งผิดปกติ และเก็บลง SD Card
- การทดสอบการส่งภาพขึ้น FTP Server
- การทดสอบการส่ง SMS

4.2.1 การทดสอบการเปิดระบบ

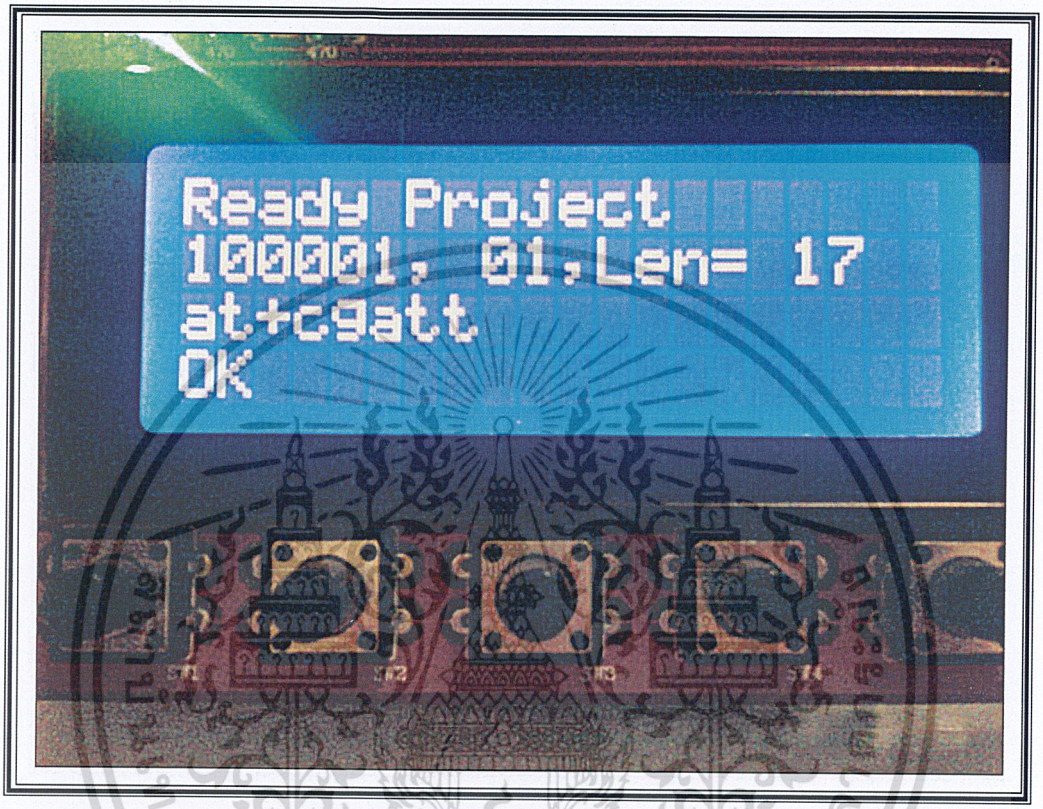
เมื่อทำการจ่ายไฟให้แก่วงจร จะเริ่มต้นการทำงานโดยแสดงสถานะผ่านทางจอLCD ดังรูปที่ 4.3

รูปที่4.4 เริ่มต้นการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 การทดสอบการเชื่อมต่อ GPRS โดยใช้ GSM Module

หลังจากเปิด GSM Module แล้ว ก็จะทำการเชื่อมต่อกับ GSM Module โดยใช้ AT Command

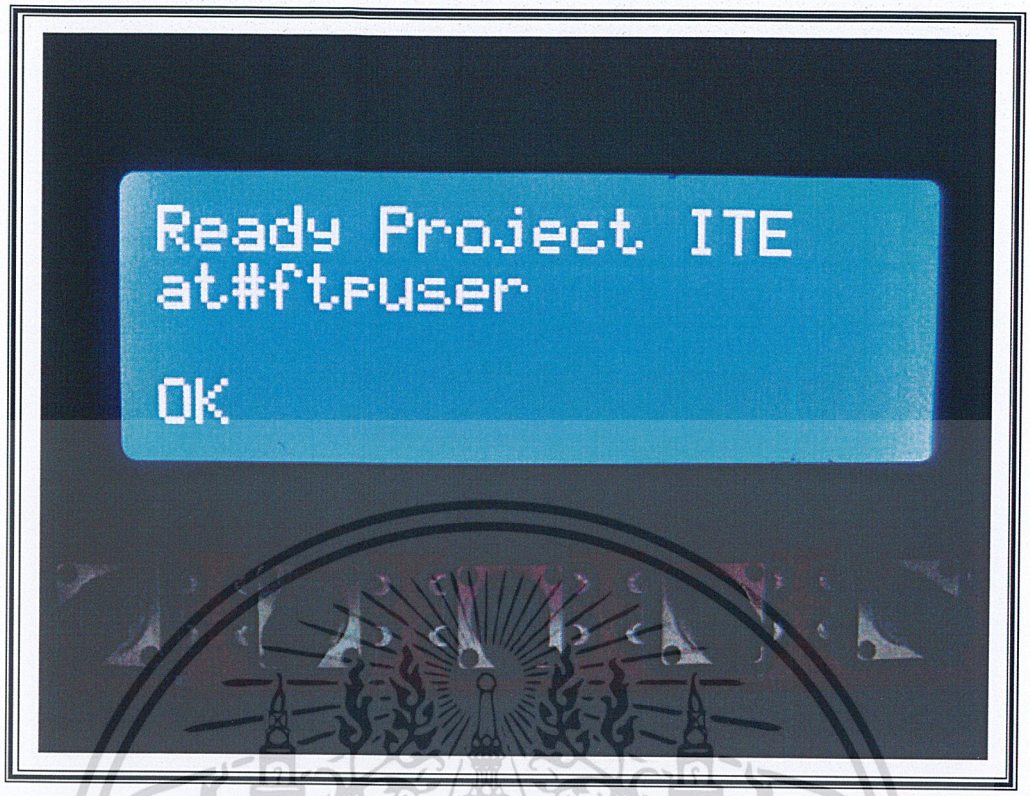


รูปที่4.5 แสดงสถานะเชื่อมต่อ GPRS

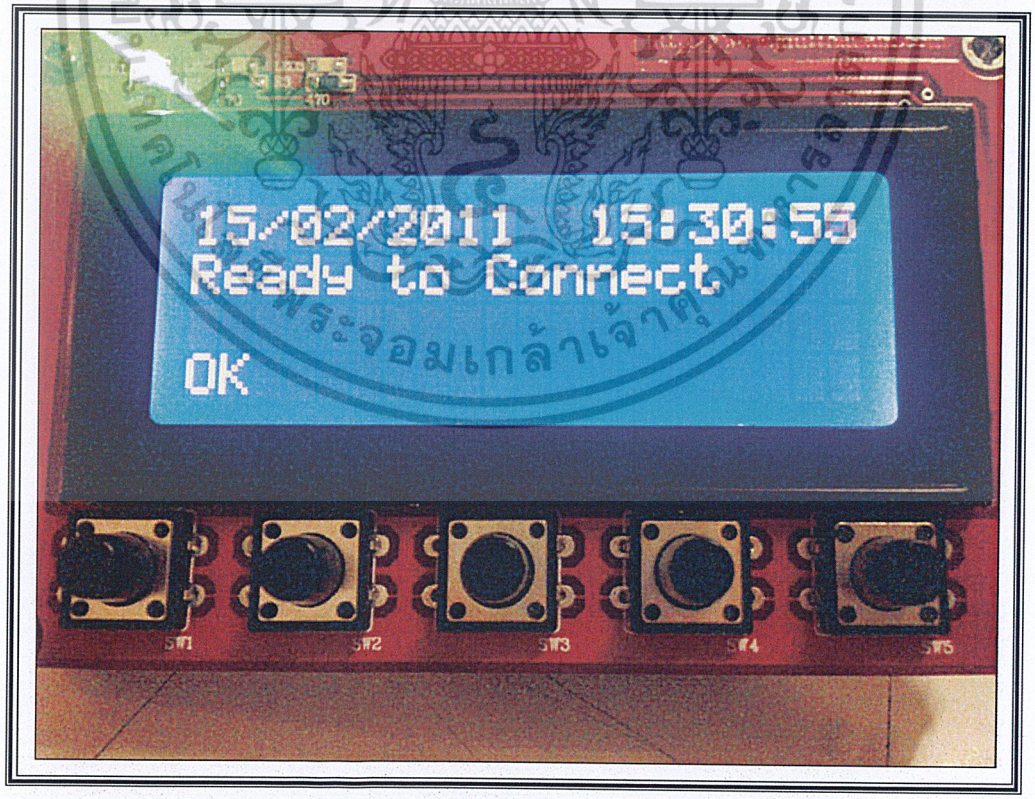
4.2.3 การทดสอบการเชื่อมต่อกับ FTP Server พอร์ต 21

เป็นการเข้าสู่ FTP Server ผ่านพอร์ต 21 ด้วย username และ password โดยใช้ AT Command

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.6 แสดงสถานะเชื่อมต่อกับ FTP Server



รูปที่4.7 แสดงสถานะเชื่อมต่อกับ FTP Server เสร็จสมบูรณ์แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 การทดสอบตรวจสิ่งผิดปกติ และเก็บลง SD Card

เมื่อเซนเซอร์ตรวจจับสิ่งผิดก็จะส่งพัลส์ไปยังระบบ กล้องก็จะทำการจับภาพ และเก็บลง SD Card ถ้าไม่มี SD Card จะไม่สามารถจับภาพได้ ดังรูป

รูปที่4.8 แสดงสถานะจับภาพ

รูปที่4.9 แสดงสถานะเก็บภาพลง SD Card

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่4.10 แสดงสถานะไม่มี SD Card คงอยู่บนระบบ

4.2.5 การทดสอบการส่งภาพขึ้น FTP Server

หลังจากเก็บภาพลง SD Card แล้วระบบจะทำการอัพโหลดภาพขึ้น FTP Server

รูปที่4.11 แสดงสถานะอัพโหลดภาพขึ้น FTP Server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.6 การทดสอบการส่ง SMS

เมื่อทำการอัพโหลดภาพขึ้น FTP Server แล้วจะทำการส่ง SMS แจ้งเตือนไปยังผู้ใช้รถยนต์

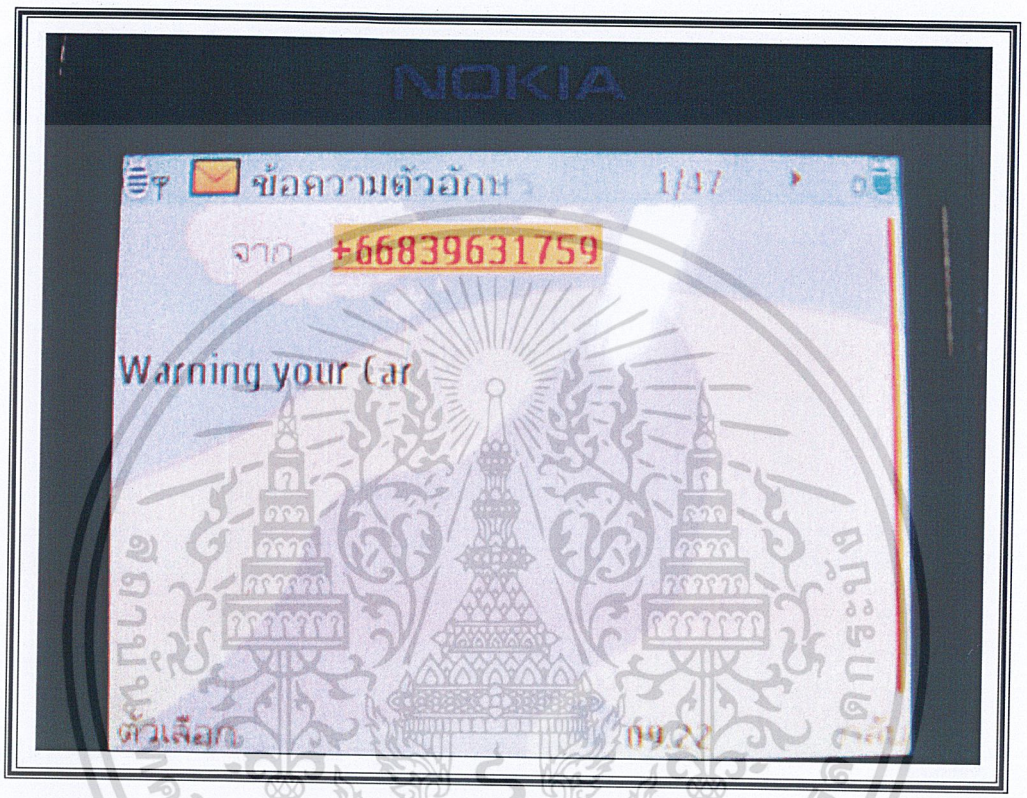
รูปที่4.12 แสดงสถานะส่ง SMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดลอง

ส่วนแสดงผลบนมือถือ

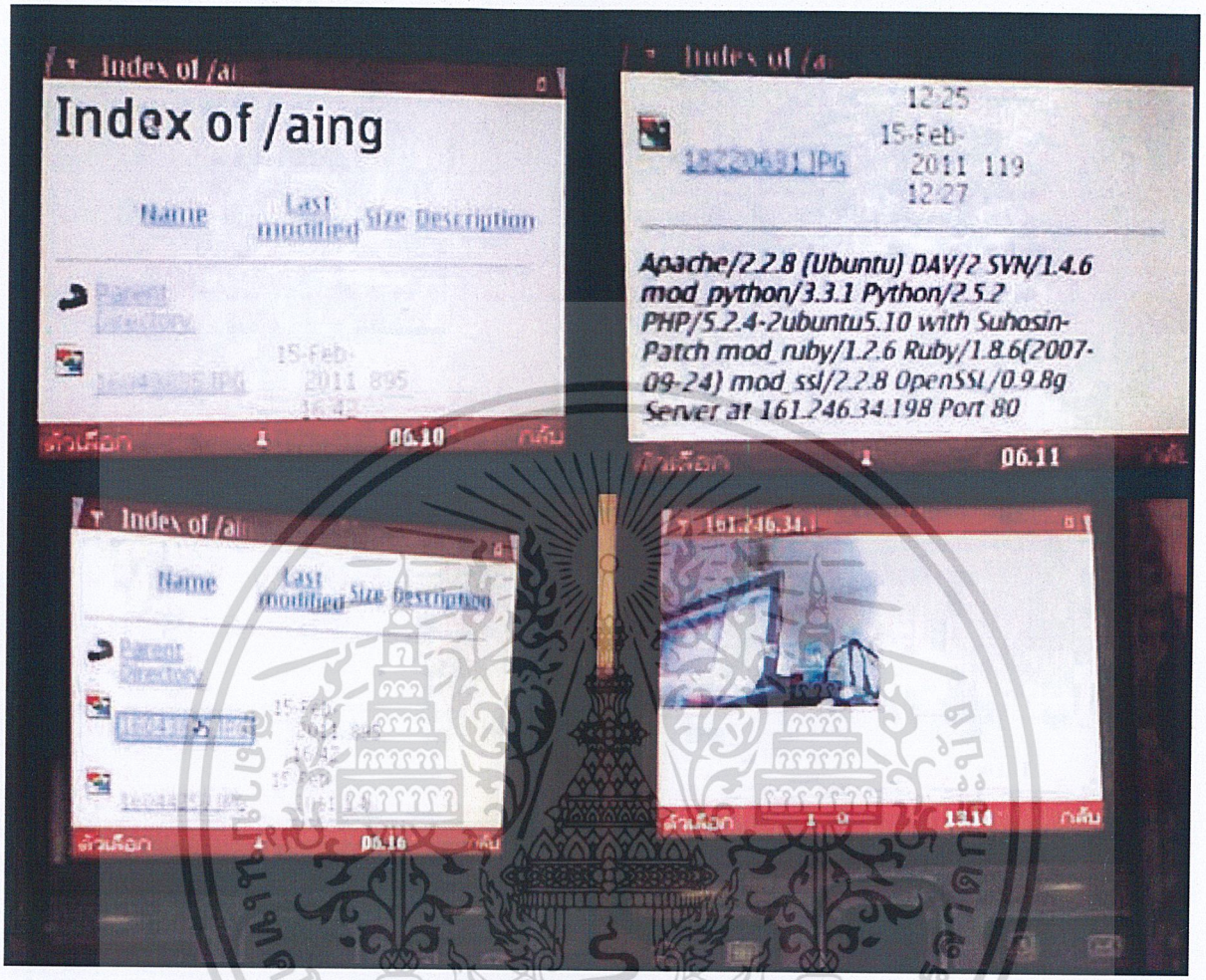
4.3.1 แสดงข้อความ SMS บนมือถือ



รูปที่4.13 แสดง การเปิดข้อความ SMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

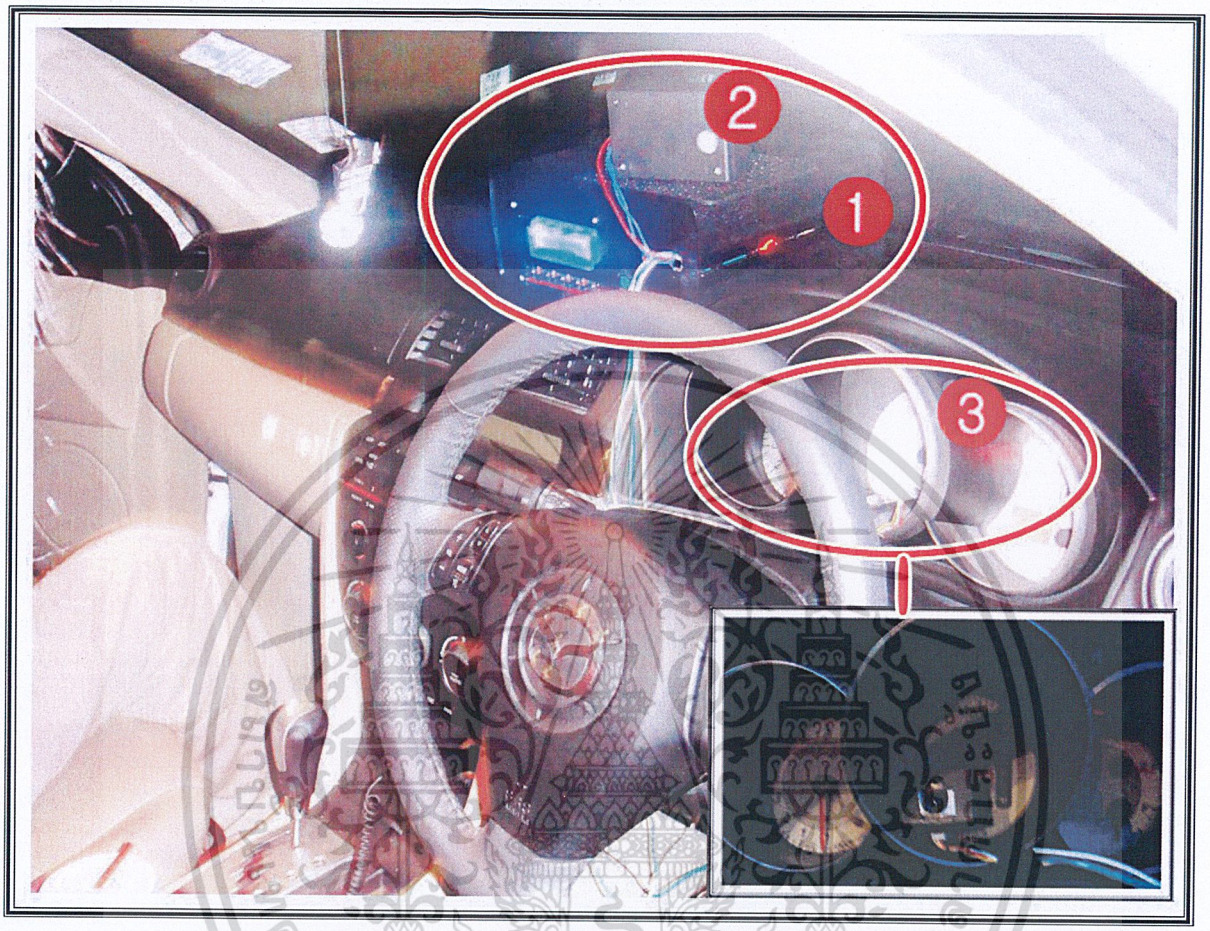
4.3.2 แสดงไฟล์ภาพบนมือถือ



รูปที่4.14 แสดงการเปิดไฟล์ภาพจากมือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การนำฮาร์ดแวร์ของระบบตรวจสอบบุคคลติดตั้งบนรถยนต์



รูปที่4.15 ฮาร์ดแวร์ของระบบตรวจสอบบุคคล เมื่อนำมาใช้งานจริง

จากรูปที่ 4.17 สามารถอธิบายได้ว่า

1. ส่วนที่ใส่บอร์ดไมโครคอนโทรลเลอร์AVR ATmega128 และ GSM Module
2. ส่วนที่ใส่ PIR Sensor
3. ส่วนที่ใส่ Camera Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง

5.1 สรุปการพัฒนาโครงงาน

ในโครงงานนี้ ระบบสามารถทำตามวัตถุประสงค์ของโครงงานได้บางประการ โดยจะแบ่งเป็นแต่ละส่วนคือ

5.1.1 ด้านผู้ใช้

- สามารถเปิดดูภาพจาก FTP Server ได้
- ได้รับ SMS เตือนจากระบบเตือนภัย

5.1.2 ด้านเซิร์ฟเวอร์

- สามารถรับไฟล์รูปภาพมาเก็บบนเซิร์ฟเวอร์ได้

5.1.3 ระบบตรวจสอบบนรถยนต์

- สามารถจับภาพได้
- สามารถตรวจสอบความเคลื่อนไหวได้
- สามารถส่งSMSเตือนภัยได้
- สามารถส่งรูปภาพขึ้นเซิร์ฟเวอร์ได้
- สามารถเก็บไฟล์รูปภาพสำรองใน SD Card ได้

5.2 ปัญหาและข้อจำกัดของโครงงาน

- ไฟล์รูปภาพบนเซิร์ฟเวอร์ที่อัพโหลดขึ้นไป มีไฟล์บางส่วนไม่เหมือนกับไฟล์ต้นฉบับที่ถ่ายได้ทำให้ภาพผิดเพี้ยน
- ความเร็วในการอัพโหลดไฟล์รูปภาพขึ้นเซิร์ฟเวอร์ เนื่องจาก GPRS อยู่ในจุดแหล่งอับสัญญาณทำให้รูปภาพที่ได้จะช้าไปบ้าง
- ระบบ 3G ที่จะนำมาใช้อัพโหลดไฟล์ภาพขึ้นเซิร์ฟเวอร์นั้น ไม่มีอุปกรณ์ที่รองรับและ ระบบ 3G นั้นในไทยยังไม่มีความเสถียร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางแก้ไข

- การแก้ไขรูปภาพที่ผิดเพี้ยนได้ จะทำการแก้ตัวเลขไบต์ของไฟล์รูปภาพของภาพที่ผิดเพี้ยน จากไบต์ OD OA เป็น OD อย่างเดียว ก็จะจะทำให้ไฟล์รูปภาพจะเหมือนไฟล์ต้นฉบับ
- เมื่อ GSM Module อยู่ในที่ที่มีสัญญาณ GPRS จะทำให้ภาพได้รับเร็วขึ้น
- ในอนาคตเมื่อ 3G มีอุปกรณ์รองรับและเสถียรมากขึ้น จะสามารถนำมาใช้ได้และทำให้การอัพโหลดไฟล์รูปภาพขึ้นเซิร์ฟเวอร์จะรวดเร็วขึ้นมาก

5.4 การประยุกต์ใช้ต่อไปในอนาคต

- สามารถนำไปประยุกต์ใช้ได้อย่างมากมาย ในระบบตรวจสอบบุคคลนั้น อาจจะใช้การส่งไฟล์รูปภาพ MMS ได้เลย ขึ้นอยู่กับอุปกรณ์และชุดคำสั่งที่ใช้ว่ารองรับหรือไม่
- สามารถนำระบบตรวจสอบนี้ไปใช้กับระบบความปลอดภัยที่ใช้ในชีวิตประจำวันได้อย่างหลากหลาย เช่น ระบบตรวจสอบภายในบ้าน ระบบตรวจสอบภายในอาคาร เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] http://www.thaieasyelec.com/Review-Product-Article/PIR-Motion-Sensor-Getting-Started.html

[2] http://skyway01.212cafe.com/archive/

[3] http://tang-mae.exteen.com/20061026/gsm-2-1

[4] http://pathloss40.wordpress.com/page/2/

[5] http://www.mechatronics-lab.com/index.php?name=News&file=print&sid=10

[6] http://www.mind-tek.net/ds1307.php

[7] http://pirun.ku.ac.th/~fsciwrs/articles/rs232/rs232c.html

[8] http://www.cpe.ku.ac.th/~yuen/204323/io/complete.htm

[9] http://www.micropik.com/PDF/MINIRS2323v.pdf

[10] "คู่มือเซมิคอนดักเตอร์ อิเล็กทรอนิกส์", บริษัท อีเล็กทรอนิคส์ ชอร์ซ จำกัด, 2553

[11] นายนิรุธ อำนวยศิลป์, "คู่มือการเขียนโปรแกรมด้วยภาษาซี", สำนักพิมพ์ โปรวิชั่น บจก., 2550

[12] "การประยุกต์ใช้งานภาษา C กับไมโครคอนโทรลเลอร์ AVR", บริษัท อีเล็กทรอนิคส์ ชอร์ซ จำกัด, 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

คู่มือการใช้งานและการติดตั้งโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ET-AVRProg mini

ET-AVRProg mini เป็นบอร์ดที่ออกแบบมาเพื่อใช้ในการดาวน์โหลด Hex File ให้กับไมโครคอนโทรลเลอร์ตระกูล AVR ของ Atmel โดยผ่านทางพอร์ต ISP โดยใช้งานร่วมกับโปรแกรม AvrProg, AvrOspII, CodeVision, avrdude หรือโปรแกรมอื่นๆ ที่รองรับโปรโตคอล AVR910

คุณสมบัติของทางเทคนิคของ ET-AVRProg mini

1. โปรแกรมผ่านทางพอร์ต ISP สามารถอ่าน เขียน ลบ และป้องกันการอ่านข้อมูลได้
2. สามารถใช้ได้กับบอร์ดเป้าหมายที่ใช้ไฟเลี้ยงตั้งแต่ 2.5V – 5.5V
3. การติดต่อสื่อสารผ่านพอร์ต USB
4. มีสัญญาณ Clock ไว้แก้ Fuse Bits ในกรณีที่เลือก Fuse Bits แหล่งของสัญญาณ Clock ผิด
5. มี LED แสดงการทำงาน USB,STATUS
6. ใช้ขั้วต่อ ISP แบบมาตรฐาน ISP 10 PIN สามารถใช้งานกับบอร์ด AVR ของทาง อีทีที หรือที่อื่นที่จัดขั้วต่อแบบมาตรฐาน 10 PIN ได้ เช่น ET-AVR STAMP ATMEGA64/128, ET-BASE AVR, ET-Easy168 STAMP ฯลฯ และนอกจากนี้ในชุดยังมีขั้วต่อ CONVERT จาก ISP 10 PIN ให้เป็นขั้วต่อแบบ ISP 6 PIN เพื่อให้สามารถใช้กับบอร์ด AVR อื่นๆ ได้อีก
7. สามารถใช้กับโปรแกรมต่างๆ ที่รองรับโปรโตคอล AVR910 เช่น AvrProg, AvrOspII, CodeVision, avrdude เป็นต้น
8. รองรับการใช้งานกับวินโดวน์ 98, me, 2000, XP, Vista

เบอร์ของ AVR ที่สามารถโปรแกรมได้เมื่อใช้ร่วมกับกับโปรแกรม AvrProg

AT90S1200, AT90S2313, AT90S2323, AT90S2333, AT90S2343, AT90S4414, AT90S4433, AT90S4434, AT90S8515, AT90S8535,

ATmega103, ATmega128, ATmega16, ATmega161, ATmega163, ATmega169, ATmega32, ATmega64, ATmega8, ATmega8515, ATmega8535,

ATtiny10, ATtiny12, ATtiny15, ATtiny26

เบอร์ของ AVR ที่สามารถโปรแกรมได้เมื่อใช้ร่วมกับกับโปรแกรม AvrOspII

AT90CAN128, AT90CAN32, AT90CAN64,

AT90PWM2, AT90PWM216, AT90PWM2B, AT90PWM3, AT90PWM316, AT90PWM3B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT90S1200, AT90S2313, AT90S2323, AT90S2343, AT90S4414, AT90S4433, AT90S4434, AT90S8515, AT90S8515, AT90S8535,

AT90SUSB1286, AT90SUSB1287, AT90SUSB162, AT90SUSB646, AT90SUSB647, AT90SUSB82,

ATmega103, ATmega128, ATmega1280, ATmega1281, ATmega16, ATmega161, ATmega162, ATmega163, ATmega164P, ATmega165, ATmega165P, ATmega168, ATmega168P, ATmega169, ATmega169P, ATmega16HVA, ATmega2560, ATmega2561, ATmega32, ATmega323, ATmega234P, ATmega235, ATmega3250, ATmega2350P, ATmega325P, ATmega328P, ATmega329, ATmega3290, ATmega3290P, ATmega329P, ATmega406, ATmega48, ATmega48P, ATmega64, ATmega640, ATmega644, ATmega644P, ATmega645, ATmega6450, ATmega649, ATmega6490, ATmega8, ATmega8515, ATmega8535, ATmega88, ATmega88P,

ATtiny11, ATtiny12, ATtiny13, ATtiny15, ATtiny22, ATtiny2313, ATtiny24, ATtiny25, ATtiny26, ATtiny261, ATtiny28, ATtiny43U, ATtiny44, ATtiny45, ATtiny461, ATtiny48, ATtiny84, ATtiny85, ATtiny861

เบอร์ของ AVR ที่สามารถโปรแกรมได้เมื่อใช้ร่วมกับกับโปรแกรม CodeVisionAVR

ATtiny13, ATtiny22, ATtiny22L, ATtiny2313, ATtiny2313V, ATtiny24, ATtiny25, ATtiny26, ATtiny26L, ATtiny261, ATtiny261V, ATtiny44, ATtiny45, ATtiny461, ATtiny461V, ATtiny48, ATtiny48V, ATtiny84, ATtiny85, ATtiny861, ATtiny861V, ATtiny88, ATtiny88V,

AT90S2343, AT90LS2343, AT90S4414, AT90S4433, AT90LS4433, AT90S4434, AT90LS4434, AT90S8515, AT90S8535, AT90LS8535,

AT90CAN128, AT90CAN32, AT90CAN64,

AT90PWM2, AT90PWM2B, AT90PWM216, AT90PWM3, AT90PWM3B, AT90PWM316,

AT90USB1286, AT90USB1287, AT90USB162, AT90USB646, AT90USB647, AT90USB82,

ATmega103, ATmega130L, ATmega128, ATmega128L, ATmega1280, ATmega1280V, ATmega1281, ATmega1281V, ATmega1284P, ATmega16, ATmega16L, ATmega161, ATmega161L, ATmega162, ATmega162L, ATmega162U, ATmega162V, ATmega163, ATmega163L, ATmega164, ATmega164V, ATmega164P, ATmega164PV,

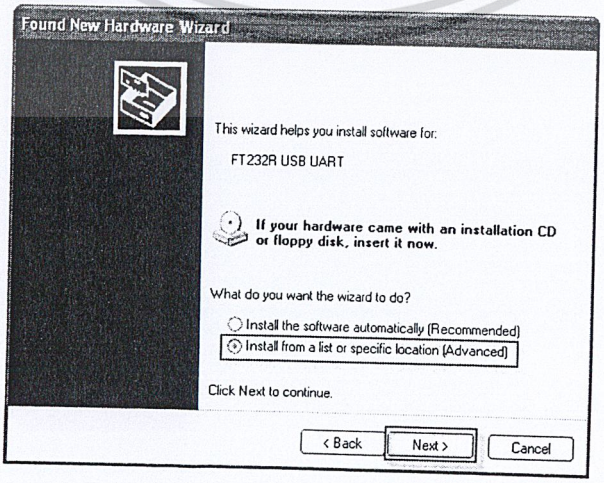
เอกสารนี้มีลิขสิทธิ์ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ATmega165, ATmega165V, ATmega168, ATmega168V, ATmega168P, ATmega168PV, ATmega169, ATmega169L, ATmega2560, ATmega2560V, ATmega2561, ATmega2561V, ATmega32, ATmega32L, ATmega232, ATmega323L, ATmega324, ATmega324V, ATmega324P, ATmega324PV, ATmega325, ATmega325V, ATmega3250, ATmega3250V, ATmega325P, ATmega325PV, ATmega3250P, ATmega3250PV, ATmega328P, ATmega328PV, ATmega329, ATmega329V, ATmega3290, ATmega3290V, ATmega329P, ATmega329PV, ATmega3290P, ATmega3290PV, ATmega48, ATmega48V, ATmega48P, ATmega48PV, ATmega603, ATmega603L, ATmega64, ATmega64L, ATmega640, ATmega640V, ATmega644, ATmega644V, ATmega644P, ATmega644PV, ATmega645, ATmega645V, ATmega6450, ATmega6450V, ATmega649, ATmega649V, ATmega6490, ATmega6490V, ATmega8, ATmega8L, ATmega8515, ATmega8515L, ATmega8535, ATmega8535L, ATmega88, ATmega88V, ATmega88P, ATmega88PV

การติดตั้งไดรเวอร์ ของ ET- AVRProg mini

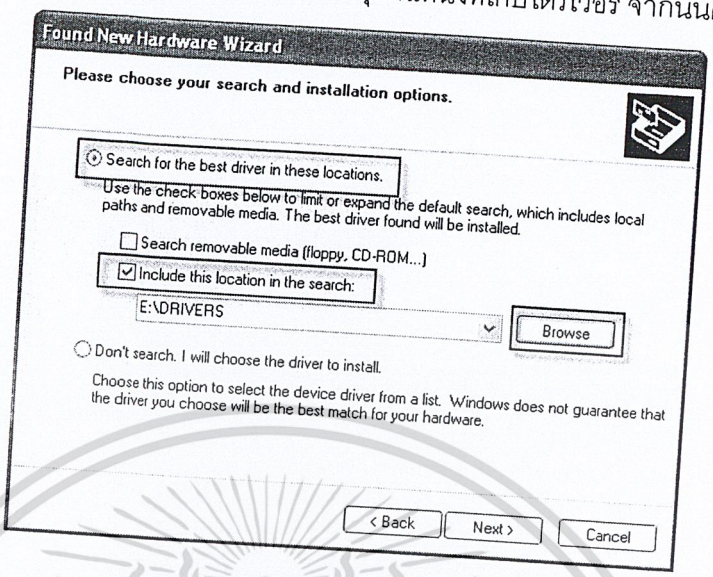
1. ทำการเชื่อมต่อ ET-AVRProg mini เข้ากับคอมพิวเตอร์ทางพอร์ต USB โดยยังไม่ต้องต่อ Target Board จากนั้นวินโดวส์จะตรวจพบฮาร์ดแวร์ใหม่ดังรูป

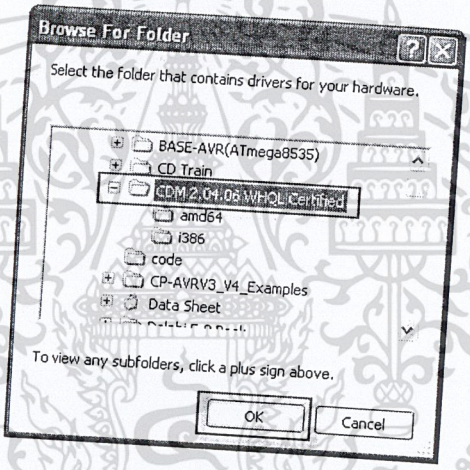
2. จากนั้นจะปรากฏหน้าต่าง Found New Hardware Wizard ให้เลือกที่ Install from a list or specific location (Advanced) และคลิก Next



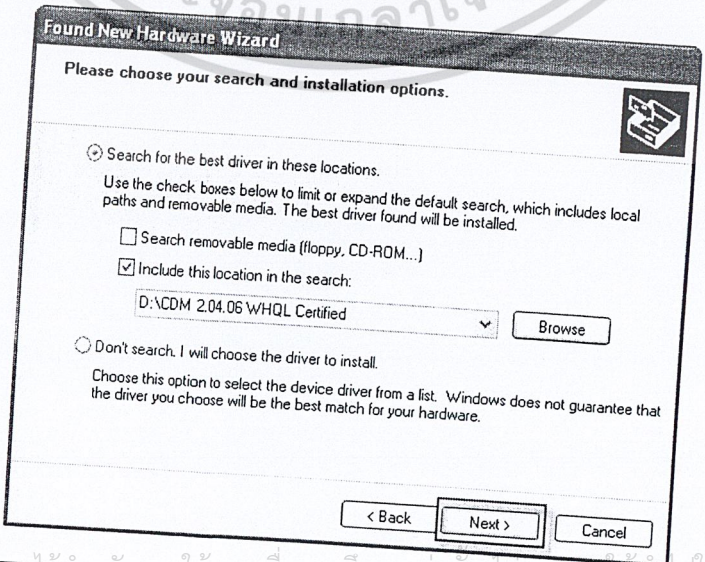
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการเลือกดังรูป และคลิกปุ่ม Browse เพื่อระบุตำแหน่งที่เก็บไดรเวอร์ จากนั้นคลิก OK

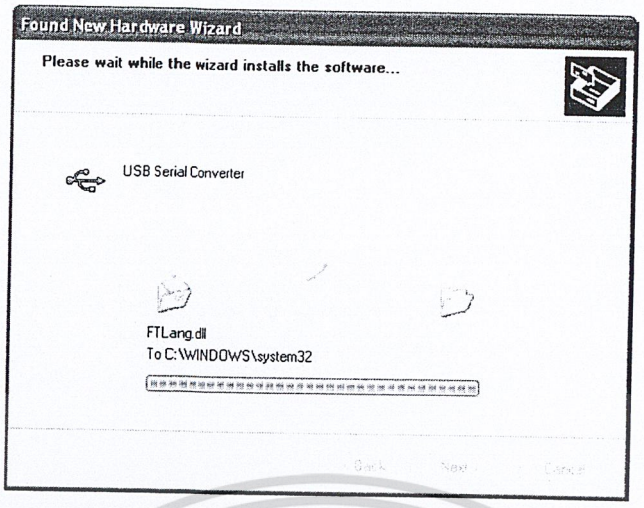




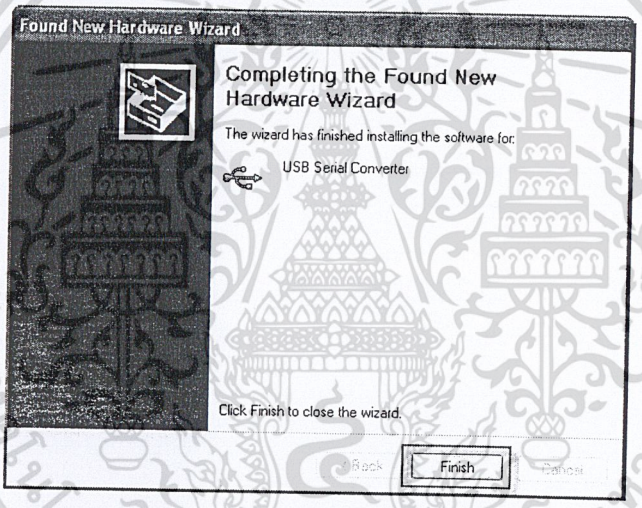
4. เมื่อทำการเลือกเรียบร้อยแล้วให้คลิกปุ่ม Next จากนั้นวินโดวส์จะทำการหาฮาร์ดแวร์เพื่อที่จะทำการติดตั้งไดรเวอร์



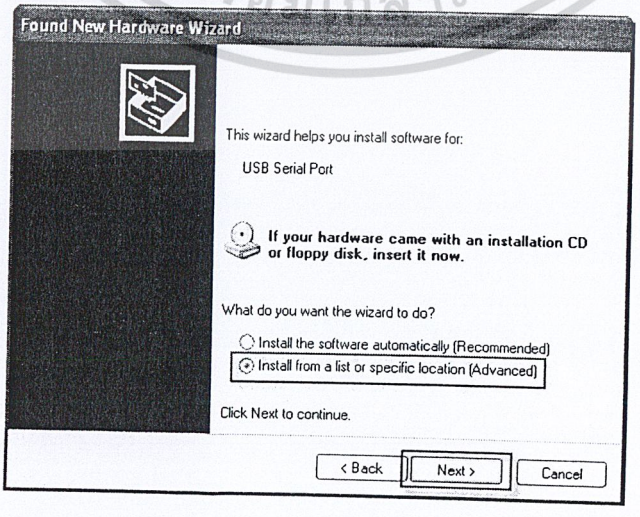
เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



5. เมื่อปรากฏหน้าต่างว่าได้ทำการติดตั้งเรียบร้อยแล้วให้คลิก Finish และจะเห็น LED แสดงสถานะ USB สว่างขึ้น แสดงว่าส่วนของ USB เริ่มทำงานแล้ว

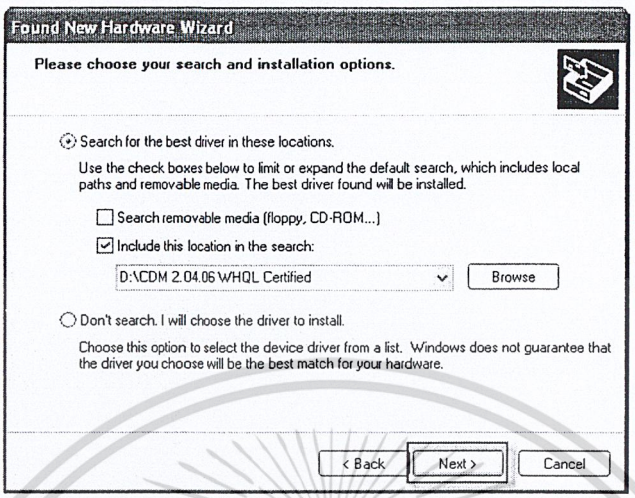


6. จากนั้นไม่นานตัวไดรเวอร์ จะมีการสร้างพอร์ตอนุกรมเสมือนขึ้นมาและมีหน้าต่างให้ติดตั้งไดรเวอร์ของ USB Serial Port ดังรูป ให้ทำการเลือกเหมือนข้อที่ผ่านมาจากนั้นคลิก Next

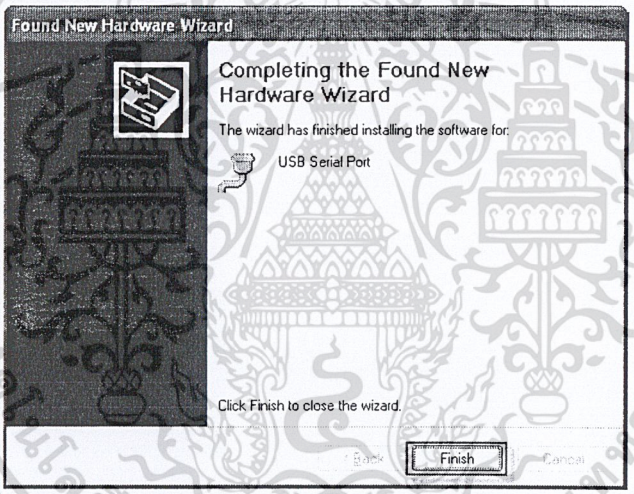


เอกสารนี้มีลิขสิทธิ์ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

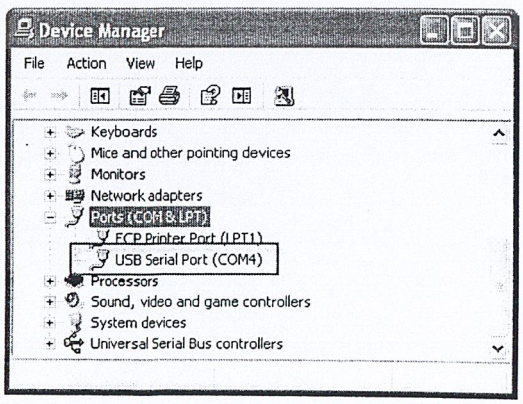
7. เลือกตำแหน่งที่ตั้งของไดรเวอร์ ซึ่งปกติจะจำค่าเดิมไว้อยู่แล้วให้คลิก Next ได้เลย จากนั้นวินโดวส์จะทำการหาฮาร์ดแวร์เพื่อที่จะทำการติดตั้งไดรเวอร์



8. เมื่อปรากฏหน้าต่างว่าได้ทำการติดตั้งเรียบร้อยแล้วให้คลิก Finish

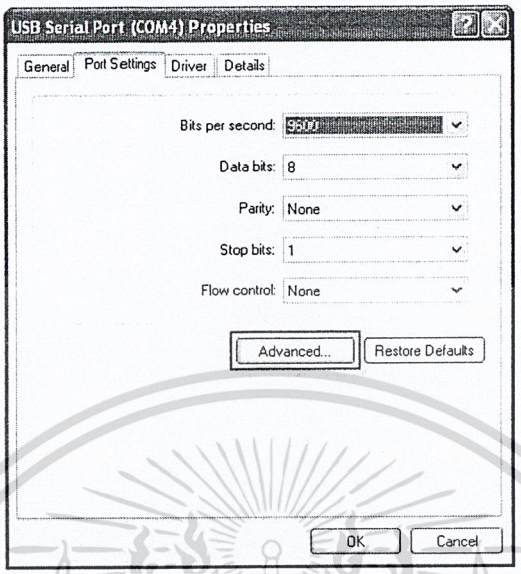


9. เราสามารถที่จะตรวจสอบว่าได้ติดตั้งไดรเวอร์ ของ ET-AVRProg mini เสร็จสมบูรณ์หรือไม่โดยดูที่ Control Panel → System เลือกแท็ป Hardware และเลือกที่ Device Manager ซึ่งจะเห็นรายการฮาร์ดแวร์ USB Serial Port เพิ่มขึ้นมาดังรูป

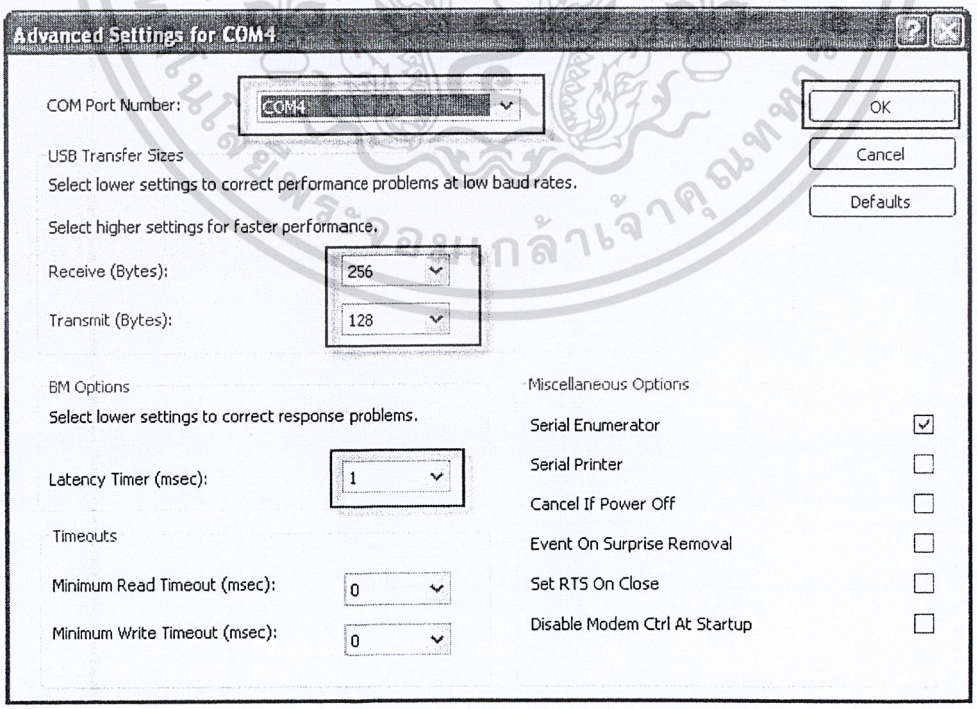


เอกสารนี้มีลิขสิทธิ์ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ดับเบิลคลิกที่ USB Serial Port ตรงส่วนของ Ports(COM&LPT) จากนั้นจะปรากฏหน้าต่าง Properties ดังรูป ให้เลือกมาที่ Port Setting และทำการคลิกที่ปุ่ม Advanced...



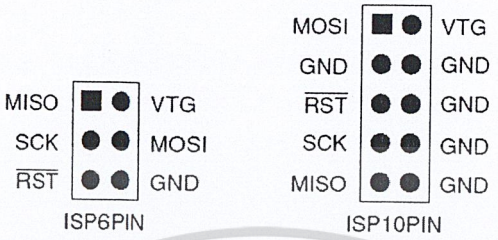
11. ทำการเปลี่ยนหมายเลขของ COM port ซึ่งในกรณีที่ใช้กับโปรแกรม AvrProg (AVR Studio) จะรองรับแค่ COM1-COM4 เท่านั้น แต่ถ้าใช้โปรแกรม AvrOspII หรือ CodeVision จะสามารถเลือก COM Port มากกว่า COM4 ได้ กำหนดค่า Receive(Bytes) เป็น 256 Transmit(Bytes) เป็น 128 และ Latency Timer (msec) เป็น 1 ดังรูป จากนั้นคลิกที่ OK เพื่อยืนยันการเปลี่ยนแปลง จากนั้นให้ทำการรีสตาร์ทเครื่องคอมพิวเตอร์หรือสแกนหาฮาร์ดแวร์ใหม่



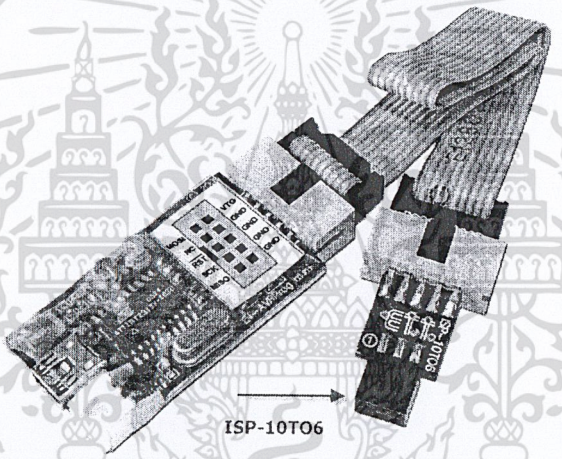
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อ ET- AVRProg mini เข้ากับไมโครคอนโทรลเลอร์ AVR

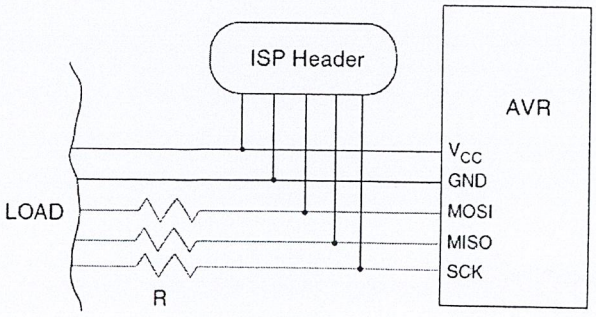
โดยมาตรฐานแล้วขั้วต่อ ISP สำหรับการโปรแกรมไมโครโทรลเลอร์ AVR จะมี 2 แบบ คือแบบ ISP6PIN และ ISP10PIN ดังรูป ซึ่งบอร์ดไมโครโทรลเลอร์ AVR ของทางอีทีทีจะใช้แบบ ISP10PIN



ในกรณีที่ผู้ใช้มีบอร์ดที่ใช้มาตรฐานแบบ ISP6PIN ก็สามารถใช้ร่วมกับ ET-AVRProg mini ได้ โดยใช้ร่วมกับ ADAPTER ISP-10TO6 ดังรูป

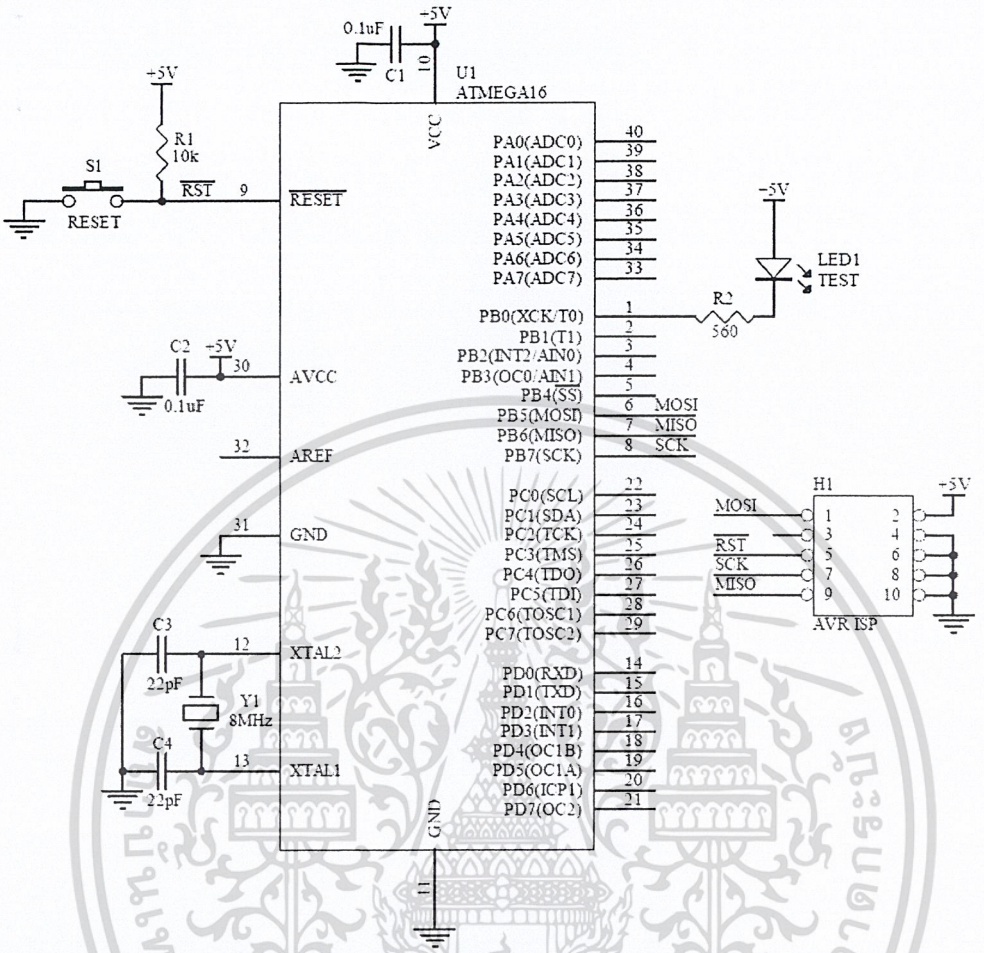


การต่อใช้งานจะใช้สายสัญญาณ MISO , MOSI , SCK , RST , VTG , GND ซึ่งจะต้องต่อกับขา ISP Interface ของ AVR ซึ่งจะต้องมีไฟเลี้ยงบอร์ดไมโครคอนโทรลเลอร์ AVR ต่างหาก ไม่สามารถใช้ไฟเลี้ยงจาก ET-AVRProg mini ได้ และในกรณีที่ผู้ใช้งานใช้ขา ISP เป็นขาพอร์ตใช้งานด้วยควรจะต่อตัวต้านทานอย่างน้อย 100 โอห์ม ก่อนที่จะไปใช้งานด้วยดังรูป เพราะถ้าไม่ต่อจะทำให้ไปโหลดสัญญาณในการโปรแกรมทำให้การโปรแกรมอาจมีปัญหาได้

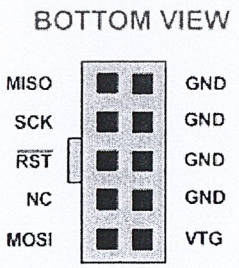


เอกสารนี้จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างวงจรไมโครโทรลเลอร์ AVR



จากรูปเป็นตัวอย่างวงจรใช้งานของ AVR เบอร์ ATMEGA16 ซึ่งจะใช้ประกอบในการอธิบายการใช้งาน ET-AVRProg mini โดยจะใช้ขั้ว ISP แบบ ISP10PIN ซึ่งผู้ใช้สามารถใช้สายแพ 10 PIN จาก ET-AVRProg mini ต่อเข้าที่ **คอนเน็คเตอร์** H1 เพื่อทำการโปรแกรมได้ทันที โดยที่ตำแหน่งขาสัญญาณของสายแพ 10 PIN เมื่อมองจากด้านล่างจะแสดงดังรูป

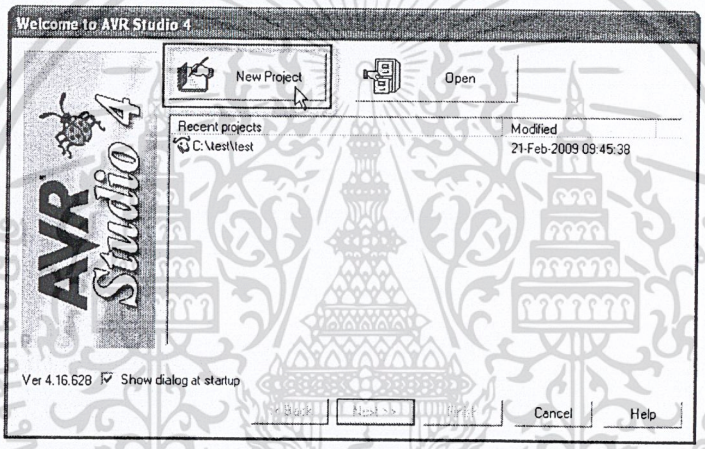


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

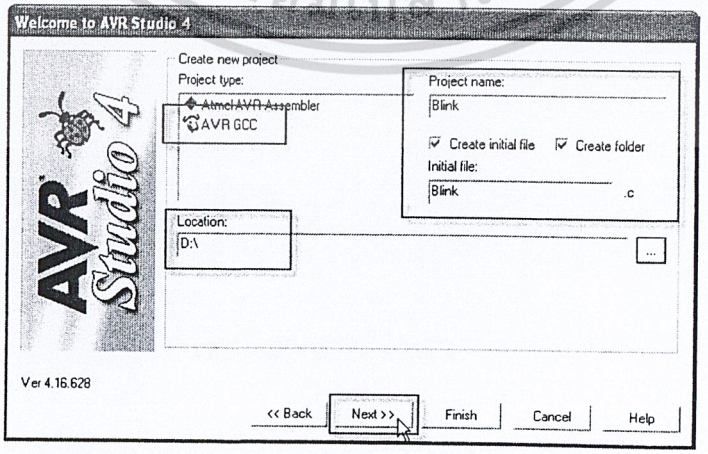
การใช้งาน ET-AVRProg mini ร่วมกับโปรแกรม AvrProg (AVR Studio)

ตัวอย่างนี้จะใช้ ET-AVRprog mini ร่วมกับโปรแกรม AVR Studio 4 +WinAVR ซึ่งเป็นคอมไพเลอร์ภาษาแอสแซมบลีและภาษาซีฟรี ผู้ใช้สามารถใช้งานได้โดยไม่มีปัญหาเรื่องลิขสิทธิ์ โดยโปรแกรม AVR Studio 4 และ WinAVR สามารถหาได้จาก CD คู่มือนี้ หรือดาวน์โหลดที่ www.atmel.com สำหรับ AVR Studio4 ส่วน WinAVR สามารถ ดาวน์โหลดที่ http://winavr.sourceforge.net/ ซึ่งขั้นตอนใช้งานจะมีดังนี้

1. ทำการเปิดโปรแกรม AVR Studio 4 จากนั้นจะปรากฏหน้าต่าง Welcome to AVR Studio 4 ให้เลือก New Project เพื่อสร้างโปรเจคใหม่ ดังรูป

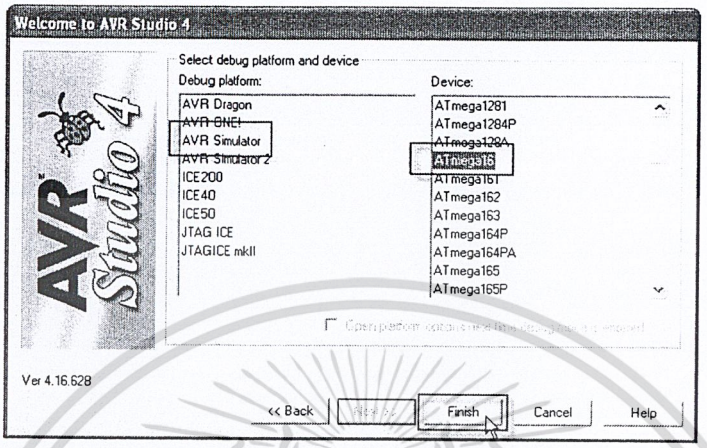


2. จากนั้นให้ทำการเลือก Project type เป็น AVR GCC ในกรณีที่ต้องการเขียนโปรแกรมเป็นภาษาซี หรือ Atmel AVR Assembler ในกรณีที่ต้องการเขียนโปรแกรมเป็นภาษาแอสแซมบลี จากนั้นทำการตั้งชื่อโปรเจค , ชื่อไฟล์ และตำแหน่งที่จะเก็บโปรเจคตามต้องการจากนั้นคลิกที่ปุ่ม Next>> ดังรูป

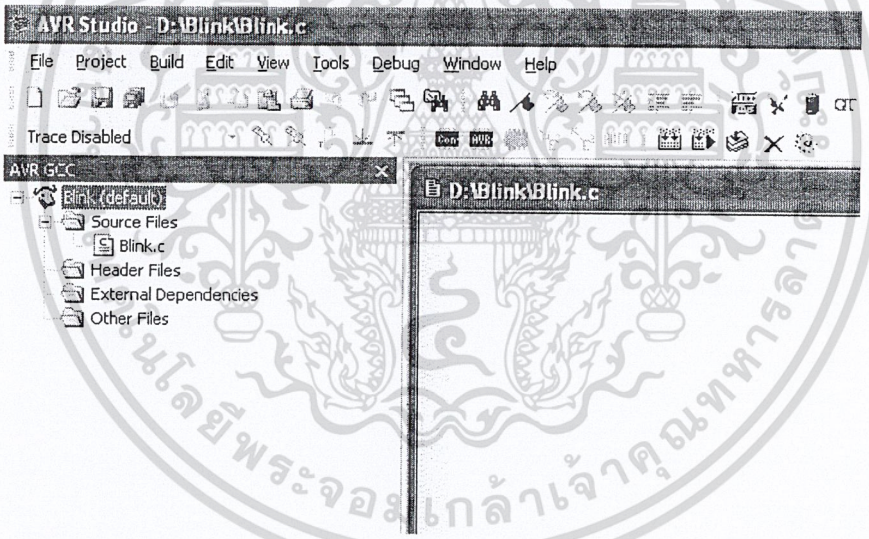


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เลือก Debug platform เป็น AVR Simulator ซึ่งจะใช้ในกรณีที่ต้องการจำลองการทำงานของโปรแกรมที่เขียนขึ้นด้วยซอฟแวร์ และเลือก Device เป็น Atmega16 ซึ่งเป็นเบอร์ที่ใช้อ้างอิงประกอบการใช้งาน จากนั้นคลิกที่ปุ่ม Finish ดังรูป

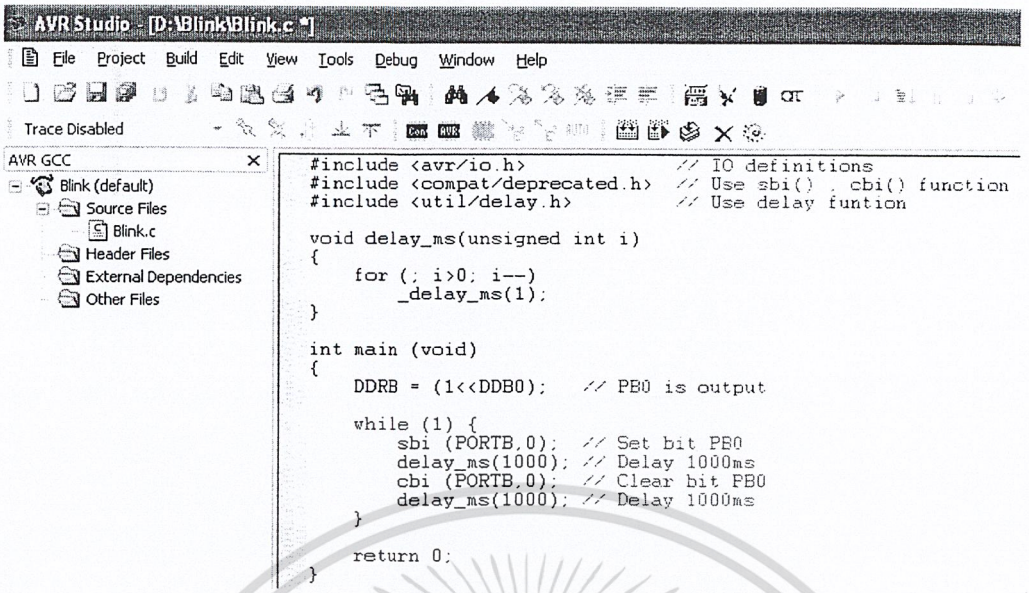


4. จากนั้นจะปรากฏหน้าต่างว่างๆ สำหรับการเขียนโปรแกรมดังรูป

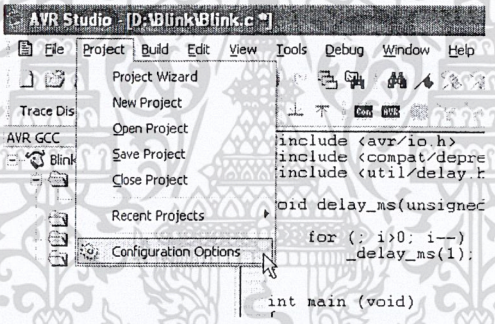


5. ทำการเขียนโปรแกรมภาษาซีตามต้องการ ในที่นี้จะยกตัวอย่างการเขียนโปรแกรมไฟกระพริบที่ขา PB0 ของ Atmega16 ด้วยอัตรา 1 วินาที ดังตัวอย่าง

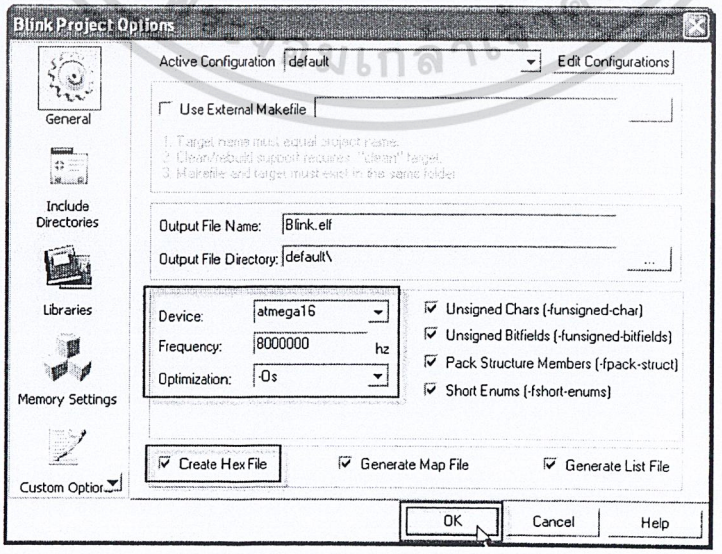
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



6. เมื่อทำการเขียนโปรแกรมเสร็จเรียบร้อยให้ทำการเลือกที่เมนู Project → Configuration Options ดังรูป

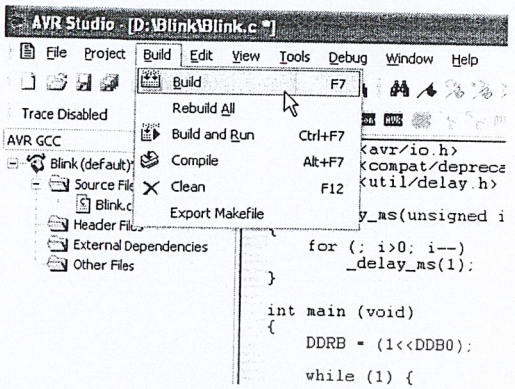


7. จากนั้นจะปรากฏหน้าต่าง Project Options ให้เลือก Device เป็น atmega16 , Frequency เป็น 8000000 hz ตามวงจรประกอบ ทำเลือกที่ช่อง Create Hex File จากนั้นคลิกที่ปุ่ม OK ดังรูป

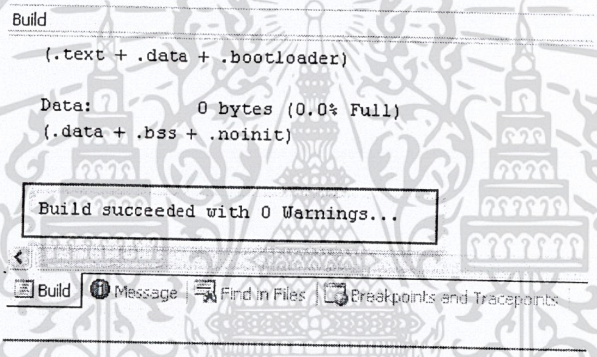


เอกสารนี้มีลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

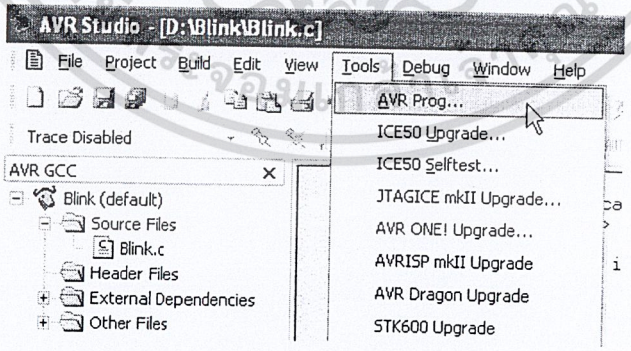
8. เมื่อตั้งค่าต่างๆ เสร็จเรียบร้อยแล้ว ให้ทำการเลือกที่เมนู Build เพื่อคอมไพล์โปรแกรม ดังรูป



9. ถ้าโปรแกรมที่เขียนขึ้นถูกต้องตามหลักไวยากรณ์จะปรากฏข้อความ Build succeeded ดังรูปแต่ถ้าข้อความแจ้งเตือน Error ให้ทำการแก้ไขและทำการ Build ใหม่

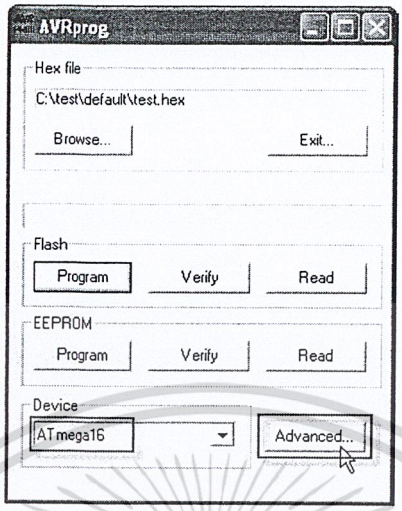


10. เมื่อทุกอย่างเรียบร้อยขั้นต่อไปก็เป็นการดาวน์โหลด Hex File เข้าสู่ไมโครคอนโทรลเลอร์ AVR โดยเลือกที่เมนู Tools → AVR Prog ดังรูป

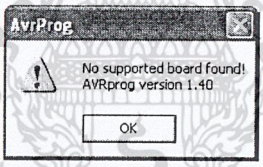


เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

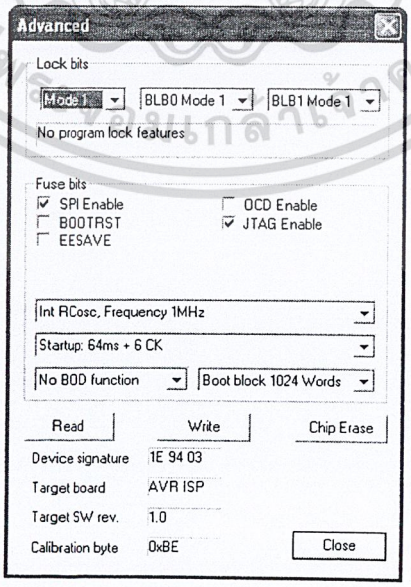
11. จากนั้นจะปรากฏหน้าต่างของโปรแกรม AVRprog ให้เลือก Device เป็น Atmega16 และคลิกที่ปุ่ม Advanced ดังรูป



12. ในกรณีที่โปรแกรมไม่สามารถติดต่อกับ ET-AVRprog mini ได้จะปรากฏข้อความดังรูป ให้ทำการตรวจสอบการเชื่อมต่อระหว่าง ET-AVRprog mini กับคอมพิวเตอร์อีกครั้ง สาเหตุอาจเกิดมาจากการลงไดร์เวอร์ไม่สมบูรณ์โดยสังเกตที่ LED สถานะ USB ที่ ET-AVRprog mini ว่าติดหรือไม่ถ้าไม่ติดแสดงว่ามีปัญหา อีกสาเหตุหนึ่งคือไม่ได้เลือกหมายของ COM Port ให้อยู่ในช่วง COM1-COM4 เนื่องจากโปรแกรม AvrProg รองรับแค่ COM1-COM4



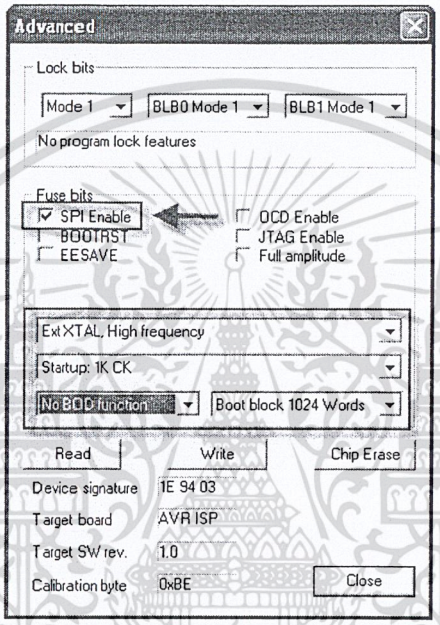
13. จากนั้นจะแสดงหน้าต่าง Advaced ดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปจะแสดงค่าพารามิเตอร์และค่า Fuse bits ของ Atmega16 ที่ต่ออยู่กับ ET-AVRprog mini ซึ่งจะเป็นค่าที่ตั้งมาจากโรงงาน ซึ่งจะเห็นว่าแหล่งของสัญญาณนาฬิกาที่ใช้เป็น IntRcosc, Frequency 1MHz ซึ่งเป็นการใช้ RC ออสซิลเลเตอร์ 1MHz ภายในไมโครคอลโทรงเลอร์เอง จะเห็นว่าจากวงจรประกอบของเราใช้คริสตอลความถี่ 8MHz ดังนั้นจำเป็นต้องแก้ค่า Fuse bits ใหม่

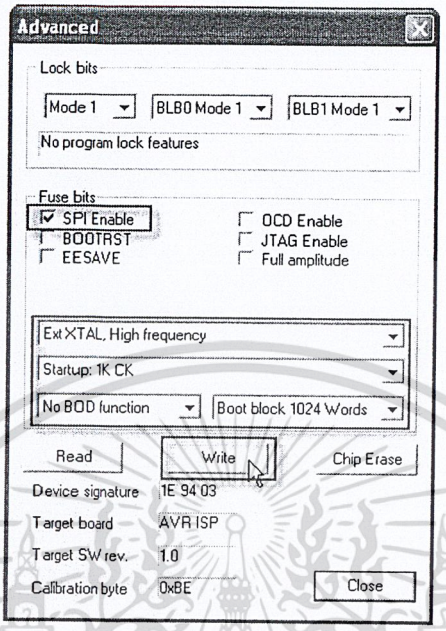
14. ทำการตั้ง Fuse bits ใหม่ให้ตรงกับวงจรที่ใช้งานจริงดังรูป ซึ่งจากรูปจะเลือกสัญญาณนาฬิกาเป็นคริสตอลจากภายนอก



<u>ข้อควรระวัง</u> Fuse bits ที่จะต้องเลือกไว้ตลอดคือ "SPI Enable" เพราะถ้าไม่เลือกไว้จะทำให้ไม่สามารถโปรแกรมผ่านทาง ISP ได้อีกต่อไป ซึ่งทางแก้ก็คือต้องนำไปแก้ด้วยเครื่องโปรแกรมแรงดันไฟสูง เช่น ALL11 , SUPERPRO เป็นต้น อีกส่วนหนึ่งที่สำคัญรองลงมาคือการเลือกแหล่งของสัญญาณนาฬิกาซึ่งจะต้องเลือกให้สอดคล้องกับวงจรที่ใช้ เช่น ถ้าวงจรใช้คริสตอลเป็นแหล่งกำเนิดสัญญาณนาฬิกา แต่เกิดไปเลือกสัญญาณนาฬิกาเป็นสัญญาณนาฬิกาจากภายนอก(ExtRCosc) ก็จะทำให้ครั้งต่อไปไม่สามารถโปรแกรมผ่านทาง ISP ได้อีก ซึ่งจะต้องแก้ด้วยเครื่องโปรแกรมแรงดันไฟสูงตามที่กล่าวมา หรือใช้ ET-AVRProg mini ในการแก้ไขแหล่งของสัญญาณนาฬิกาได้ได้ซึ่งจะกล่าวถึงในตอนท้ายของคู่มือ ดังนั้นการตั้งค่า Fuse bits นี้ผู้ใช้จำเป็นต้องใช้ความละเอียดรอบคอบเป็นอย่างมาก โดยรายละเอียดต่างๆ ของ Fuse bits สามารถดูได้จาก Data Sheet ของแต่ละเบอร์

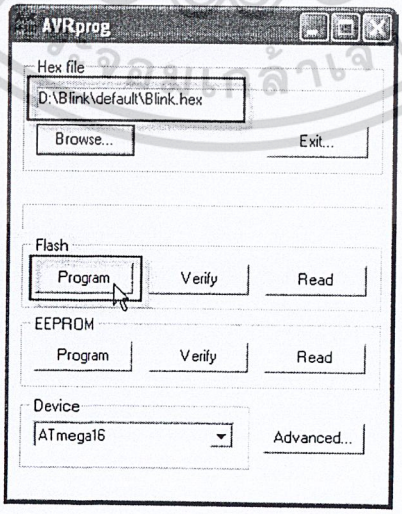
เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15. เมื่อทำการตั้งค่าต่างเสร็จแล้วก็ให้ทำการคลิกที่ปุ่ม Write เพื่อเขียนค่า Fuse bits เข้าสู่ไมโครคอนโทรลเลอร์ ดังรูป จากนั้นคลิกปุ่ม Close เพื่อปิดหน้าต่างนี้



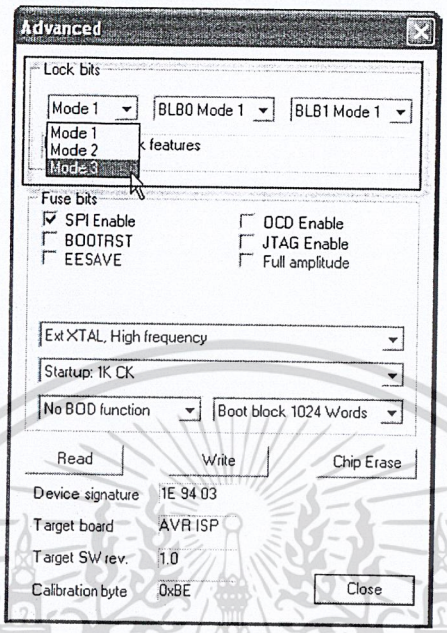
คำแนะนำ การเขียนค่า Fuse bits นี้จะเขียนเพียงครั้งแรกเท่านั้น ครั้งต่อไปก็ไม่จำเป็นต้องโปรแกรมแล้ว เพราะค่านี้จะอยู่ตลอดไม่ถูกลบไปพร้อมกับส่วนของ Program memory ดังนั้นค่า Fuse bits ควรโปรแกรมเมื่อจำเป็นเท่านั้น

16. จากนั้นก็ทำการเลือก Hex File ที่ได้จากการคอมไพล์ ซึ่งปกติจะอยู่ที่โฟลเดอร์ default ซึ่งจะอยู่ในโฟลเดอร์ของโปรเจคที่เราสร้างขึ้น ดังรูปนั้นคลิกปุ่ม Program เมื่อการโปรแกรมเสร็จเรียบร้อยจะเห็น LED ที่ขา PB0 กระพริบด้วยอัตรา 1 วินาที



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

17. ในการกรณีที่ต้องการป้องการอ่านข้อมูลกลับ (ป้องกันการ Copy) ก็สามารถทำได้โดยคลิกที่ปุ่ม Advance และเลือกที่ โหมดของ Lock bits ตามต้องการและคลิกปุ่ม Write

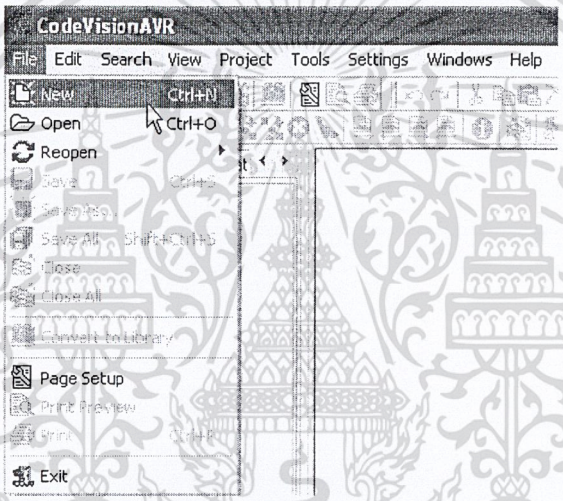


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

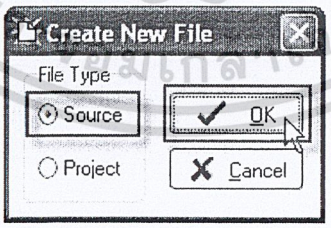
การใช้งาน ET-AVRProg mini ร่วมกับโปรแกรม CodeVisionAVR

โปรแกรม CodeVisionAVR เป็นคอมไพเลอร์ภาษาซีของไมโครคอนโทรลเลอร์ AVR ที่เป็นที่นิยมใช้กันมากเนื่องจากมีความง่ายต่อการใช้งาน แต่ข้อเสียคือ คอมไพเลอร์ตัวนี้ไม่ฟรี ซึ่งตัวอย่างนี้จะใช้ CodeVisionAVR เวอร์ชั่นทดลองใช้ซึ่งจำกัดขนาดโค๊ดของโปรแกรมไว้ที่ 2kbytes ซึ่งถ้าท่านต้องการเวอร์ชั่นเต็มหรือทดลองใช้ ก็สามารถไปที่ www.hpinfotech.com/ โดยที่โปรแกรม CodeVisionAVR นี้สามารถที่จะใช้งานร่วมกับ ET-AVRProg mini ได้ทันทีโดยไม่ต้องเปิดโปรแกรมอื่นร่วม ซึ่งขั้นตอนการใช้งานจะมีดังนี้

1. ทำการเปิดโปรแกรม CodeVisionAVR และคลิกเลือกที่เมนูคำสั่ง File → New ดังรูป

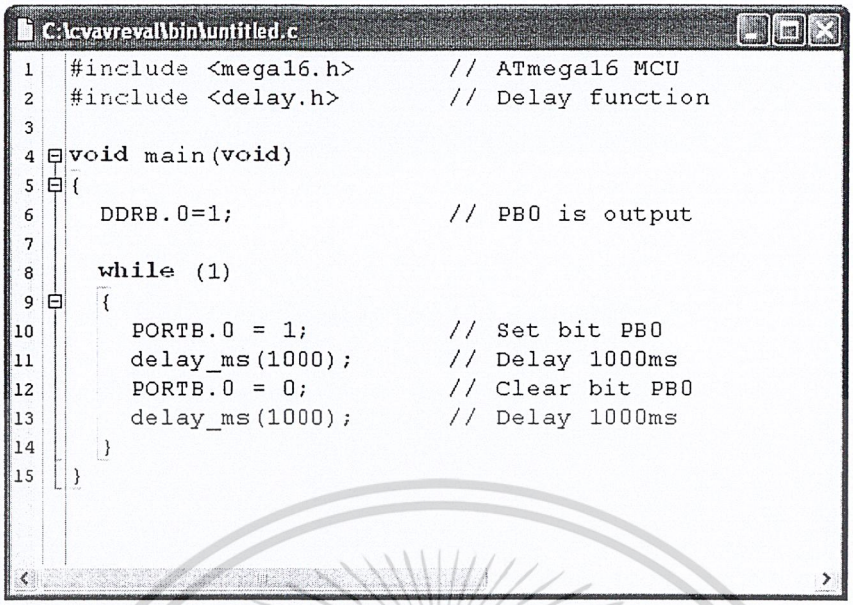


2. เลือก File Type เป็น Source เพื่อสร้างไฟล์ภาษาซีใหม่และคลิกปุ่ม OK ดังรูป

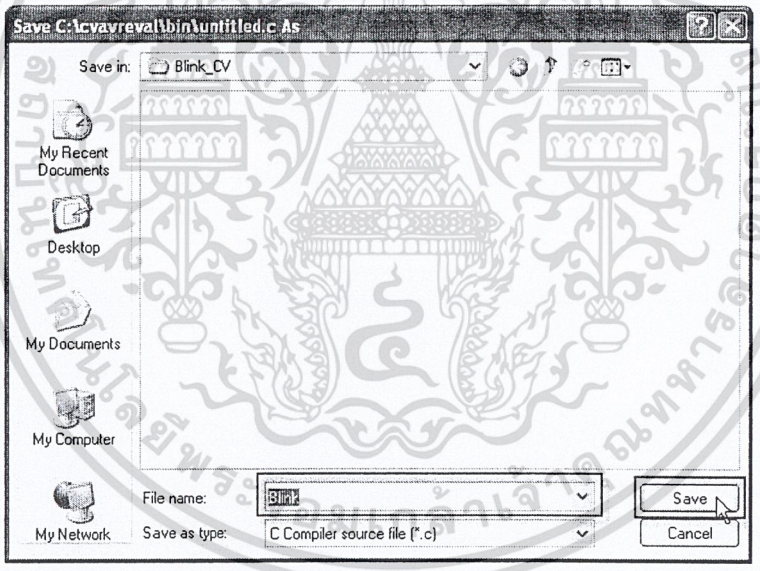


3. จากนั้นจะปรากฏหน้าต่าง Editor ให้ทำการเขียนโปรแกรมดังตัวอย่าง ซึ่งเป็นตัวอย่างโปรแกรมไฟกระพริบที่ PB0

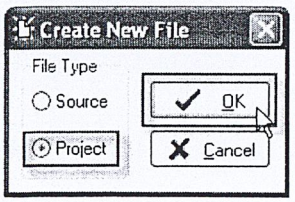
เอกสารนี้มีลิขสิทธิ์ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



4. ทำการบันทึกโปรแกรมภาษาซีที่เขียนโดยเลือกเมนู File → Save ทำการตั้งชื่อไฟล์และกดปุ่ม Save ดังรูป

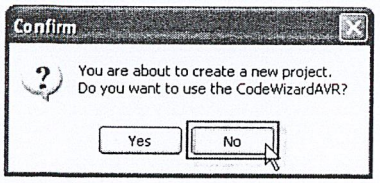


5. เลือกที่เมนู File → New และเลือก File Type เป็น Project เพื่อสร้างโปรเจกต์ใหม่และคลิกปุ่ม OK ดังรูป

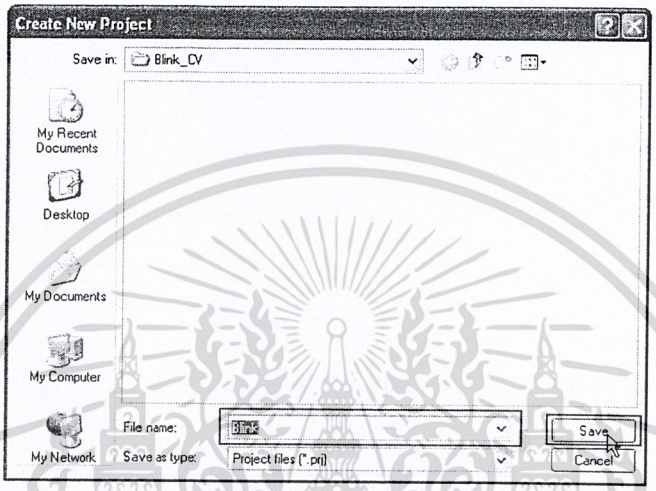


เอกสารนี้มีลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

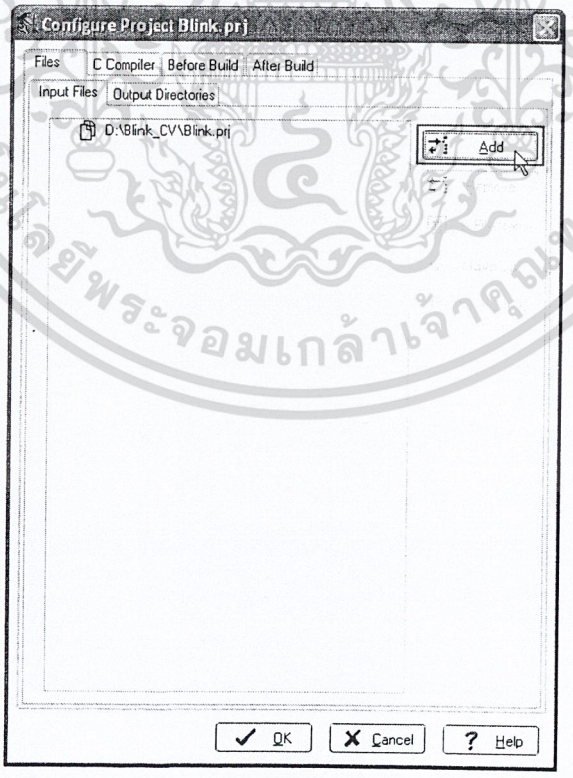
6. คลิกปุ่ม No ในกรณีที่ไม่ใช้ตัวช่วยในการสร้างโปรเจกต์ (CodeWizard) ดังรูป



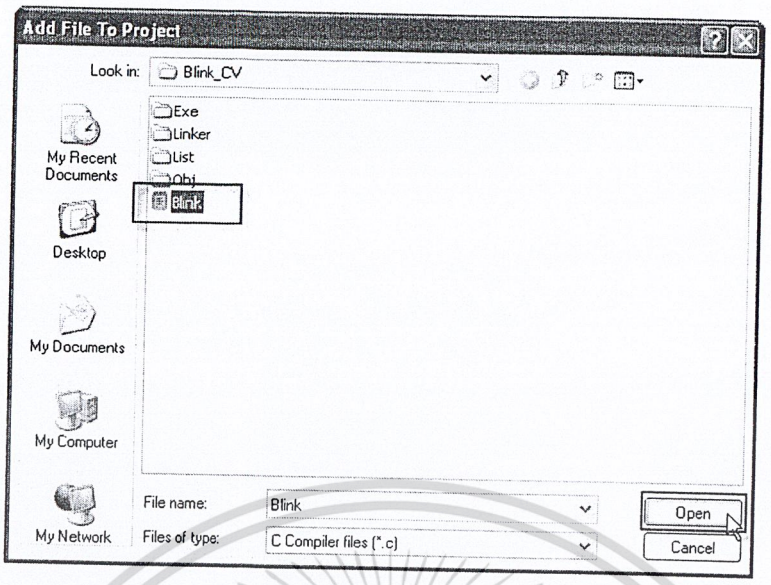
7. ทำการตั้งชื่อโปรเจกต์ตามต้องการและคลิกปุ่ม Save ดังรูป



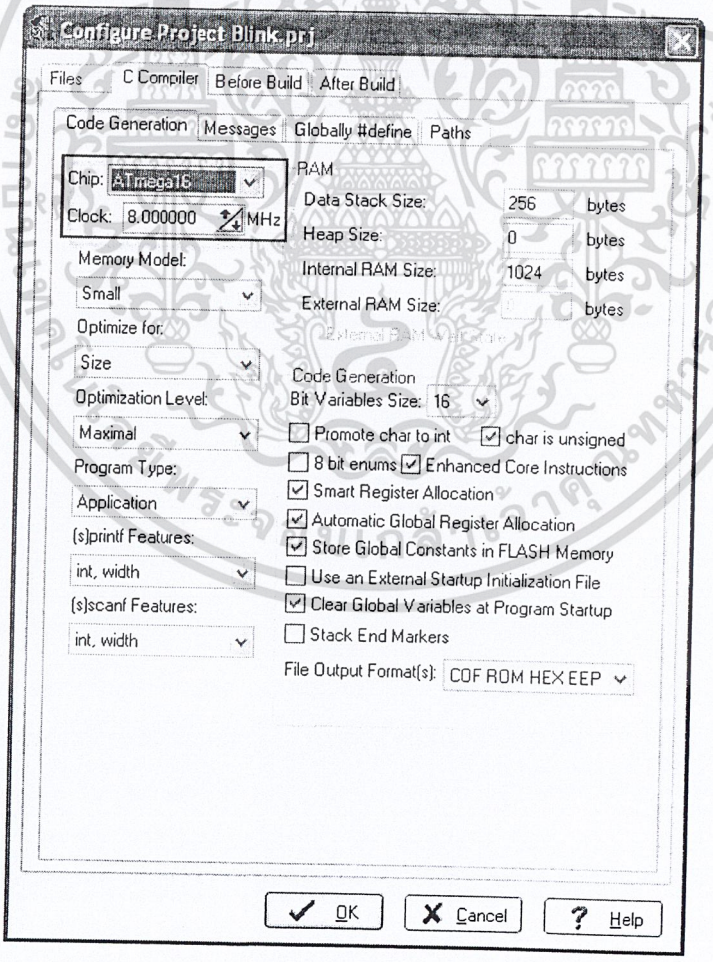
8. ทำการเพิ่มไฟล์ภาษาซีที่เขียนไว้ก่อนหน้านี้เข้ามาในโปรเจกต์โดยการคลิกปุ่ม Add ดังรูป



เอกสารนี้มีลิขสิทธิ์ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

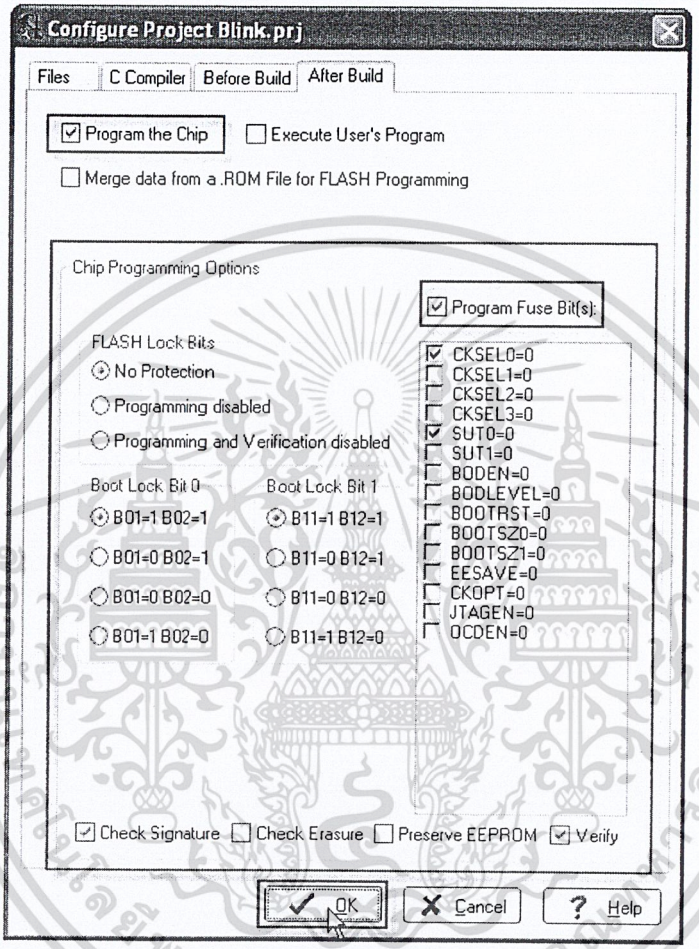


9. เลือกแท็ปของหน้าต่าง Configure Project มาที่ C Compiler เลือก Chip เป็น Atmega16 และ Clock เป็น 8.000000 MHz ดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

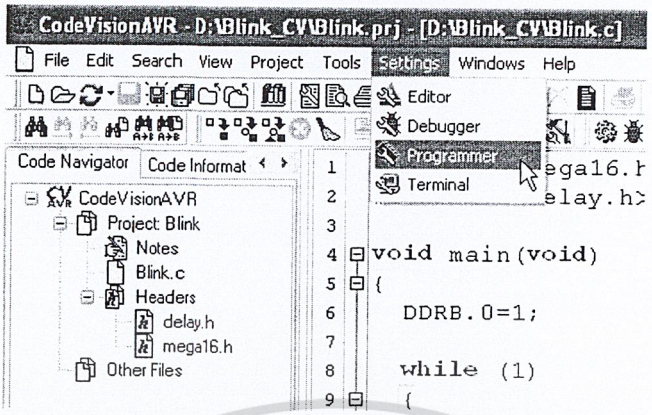
10. จากนั้นเลือกแท็ปมาที่ After Build เลือกที่ Program the Chip เพื่อให้ทำการโปรแกรมข้อมูลลงไมโครคอนโทรลเลอร์ หลังจากคอมไพล์โปรแกรมเสร็จ ส่วนค่าของ Fuse Bits และ Lock Bits สามารถตั้งได้ตามต้องการ ซึ่งค่า Fuse Bits จะตั้งดังรูป ซึ่งถ้าผู้ใช้ไม่ต้องการโปรแกรม Fuse Bits ก็ไม่ต้องเลือกออปชั่น Program Fuse Bits ได้ เมื่อตั้งค่าต่างๆ เสร็จเรียบให้คลิกปุ่ม OK



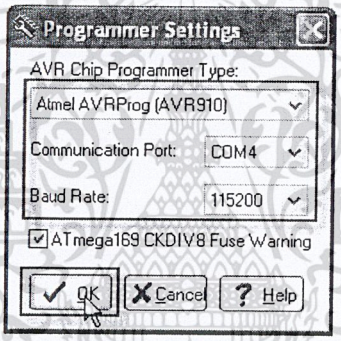
<u>ข้อควรระวัง</u> การตั้งค่า Fuse bits นี้ผู้ใช้จำเป็นต้องใช้ความละเอียดรอบคอบเป็นอย่างมาก โดยรายละเอียดต่างๆ ของ Fuse bits สามารถดูได้จาก Data Sheet ของแต่ละเบอร์

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

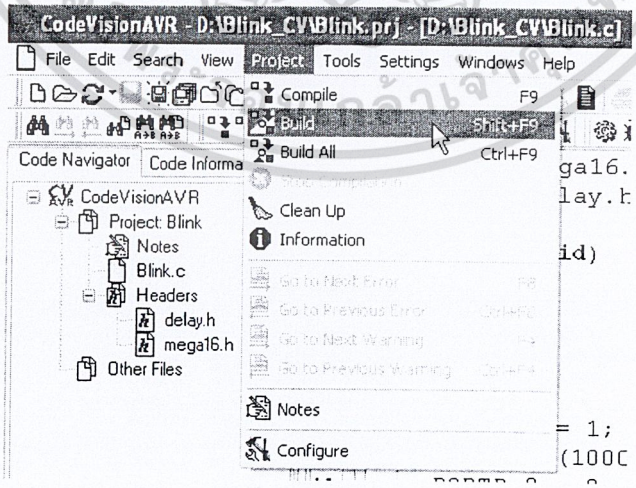
11. จากนั้นทำการเลือกชนิดของเครื่องโปรแกรมที่ใช้โดยเลือกที่เมนู Settings→Programmer ดังรูป



12. ทำการเลือก AVR Chip Programmer Type เป็น Atmel AVRProg(AVR910) , Communication Port เลือกตามที่ได้ลงได้ลง Driver ไว้ในที่นี้เลือก COM4 และ Baud Rate เป็น 115200 ดังรูป จากนั้นคลิก OK

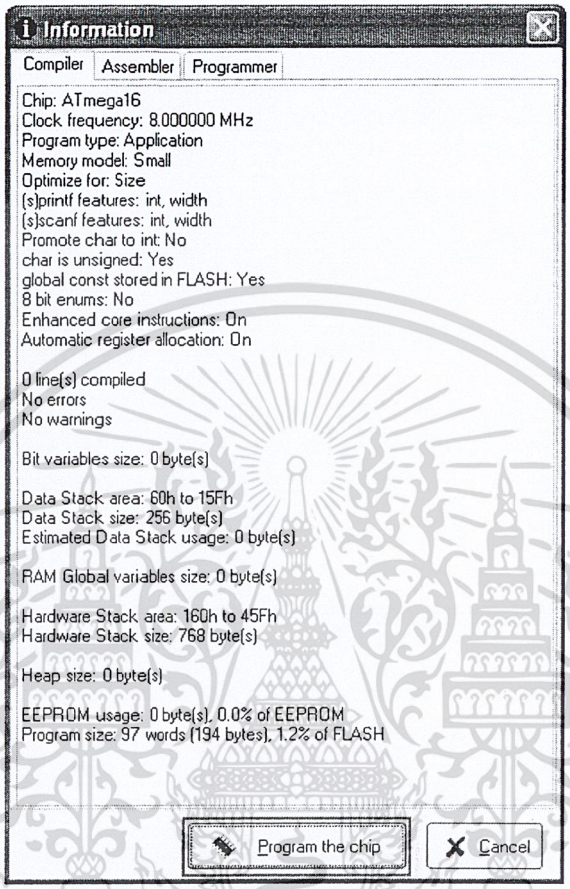


13. ทำการคอมไพล์โปรแกรมที่เขียนขึ้นโดยเลือกที่เมนู Project→Build ดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14. ถ้าโปรแกรมที่เขียนขึ้นถูกต้องตามหลักไวยากรณ์จะปรากฏข้อความ No errors ดังรูป แต่ถ้าข้อความแจ้งเตือน Error ให้ทำการแก้ไขและทำการ Build ใหม่ เมื่อไม่ข้อผิดพลาดใดๆ ให้คลิกที่ปุ่ม Program the chip เพื่อโปรแกรม Hex File เข้าสู่ไมโครคอนโทรลเลอร์ดังรูป

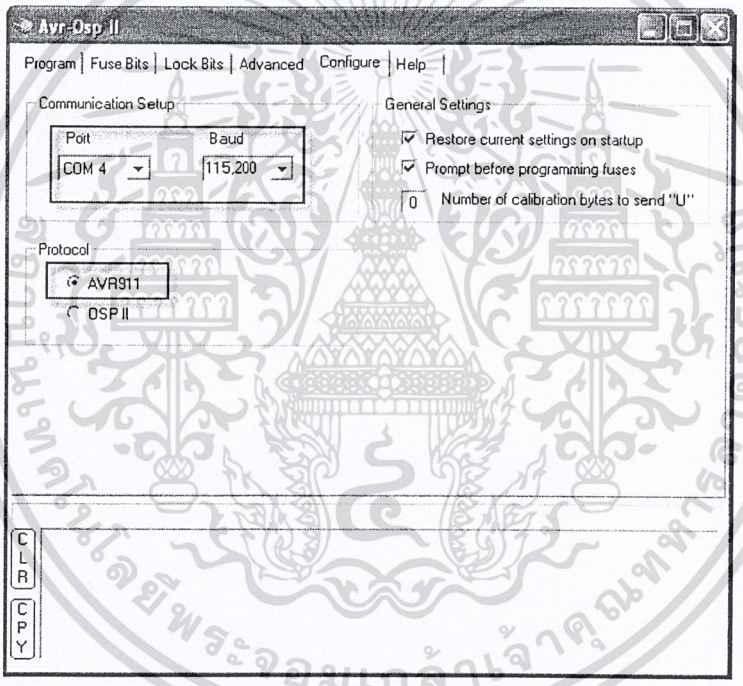


เอกสารนี้เป็นลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน ET-AVRProg mini ร่วมกับโปรแกรม AvrOspII

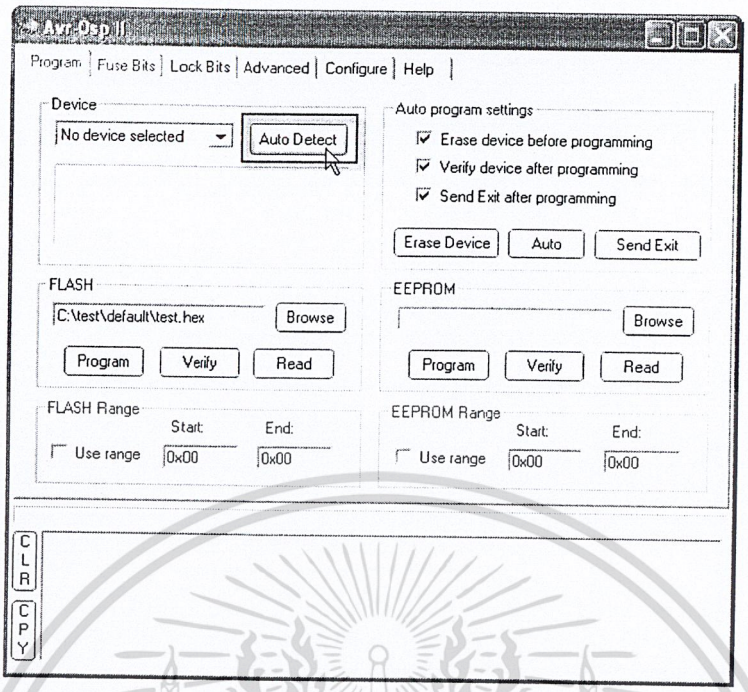
โปรแกรม AvrOspII เป็นโปรแกรมที่ใช้ในการโปรแกรม Hex File ให้กับไมโครคอนโทรลเลอร์ AVR ที่มีขนาดโปรแกรมที่เล็กใช้งานสะดวกและเบอร์ไมโครคอนโทรลเลอร์ AVR ที่สามารถโปรแกรมได้จะเยอะกว่าโปรแกรม AvrProg และสามารถเห็น COM Port ได้มากกว่า COM4 ทำให้สะดวกในการใช้งานมากขึ้น ซึ่งโปรแกรมนี้พัฒนาโดย Mike Henning ซึ่งผู้ใช้สามารถดาวน์โหลดเวอร์ชั่นใหม่ๆ ได้จาก http://www.esnips.com/web/AtmelAVR ซึ่งขั้นตอนใช้งานจะมีดังนี้

1. เปิดโปรแกรม AvrOspII จากนั้นเลือกที่แท็ป Configure เลือก Communication Setup หัวข้อ Port เป็น COM4 หรือตามที่ได้ลงไดรเวอร์ Baud เป็น 115,200 และ Protocal เป็น AVR911 ดังรูป

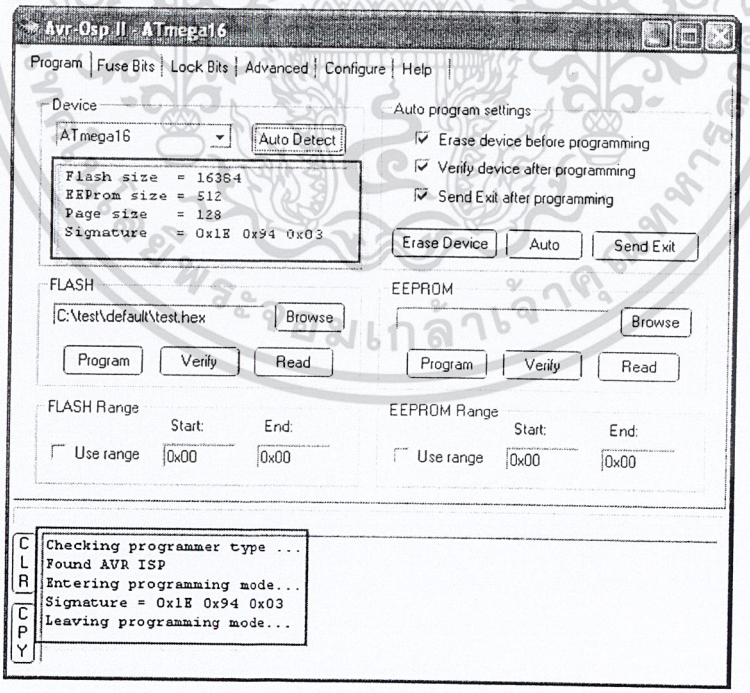


2. จากนั้นเลือกแท็ปมาที่ Program และคลิกที่ปุ่ม Auto Detect เพื่อค้นหาเบอร์ของไมโครคอนโทรลเลอร์ AVR อย่างอัตโมมัติ ดังรูป

เอกสารนี้มีลิขสิทธิ์ตามกฎหมาย ใช้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

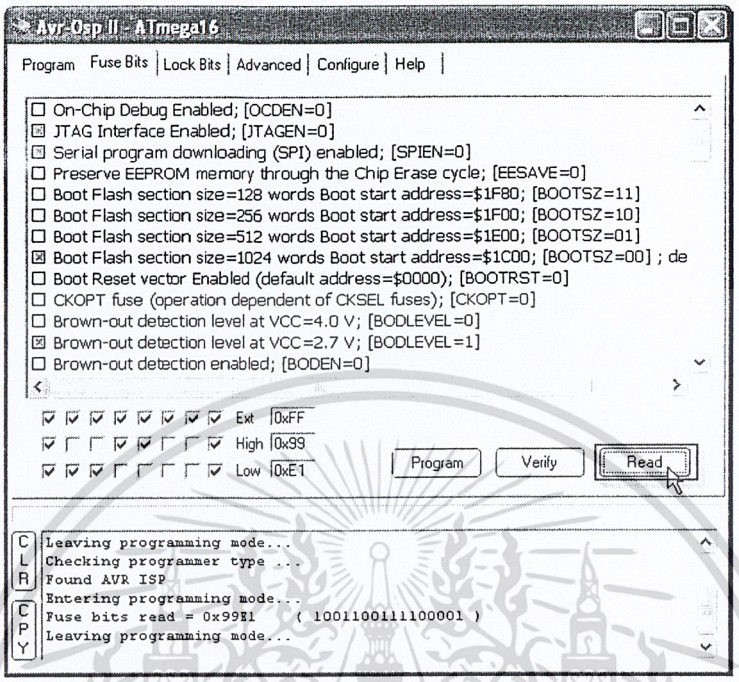


2. ถ้าทุกอย่างไม่มีข้อผิดพลาดโปรแกรมจะค้นหาเบอร์ของไมโครคอนโทรลเลอร์ AVR เจอ ดังรูป พร้อมทั้งแสดงข้อมูลจำเพาะต่างๆ ออกมาด้วย แต่ถ้ากดปุ่ม Auto Detect แล้วไม่เจอเบอร์ ก็ให้ลองเช็คบอร์ดที่นำมาต่อด้วยว่าต่อสาย ISP ถูกต้องหรือไม่ หรือบอร์ดที่นำมาต่อด้วยอาจมีปัญหา

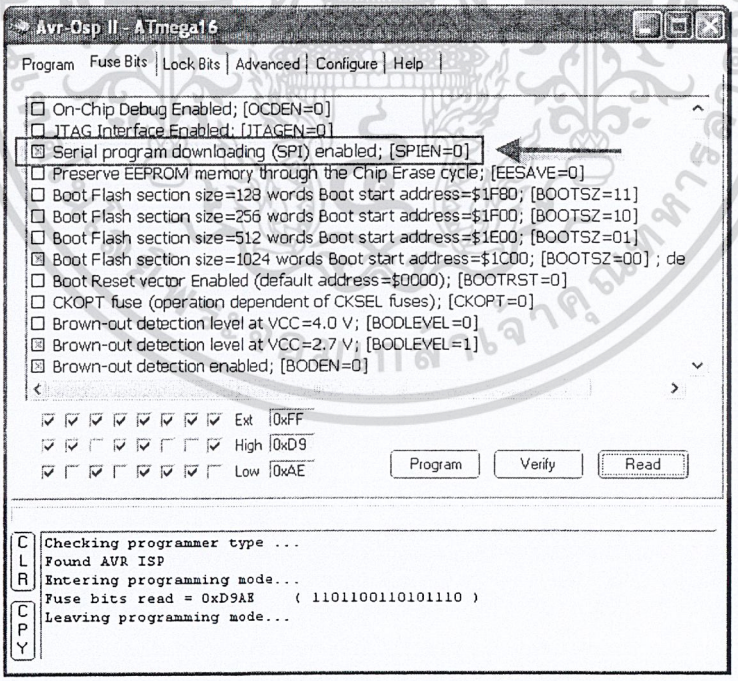


เอกสารนี้เป็นลิขสิทธิ์ของผู้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

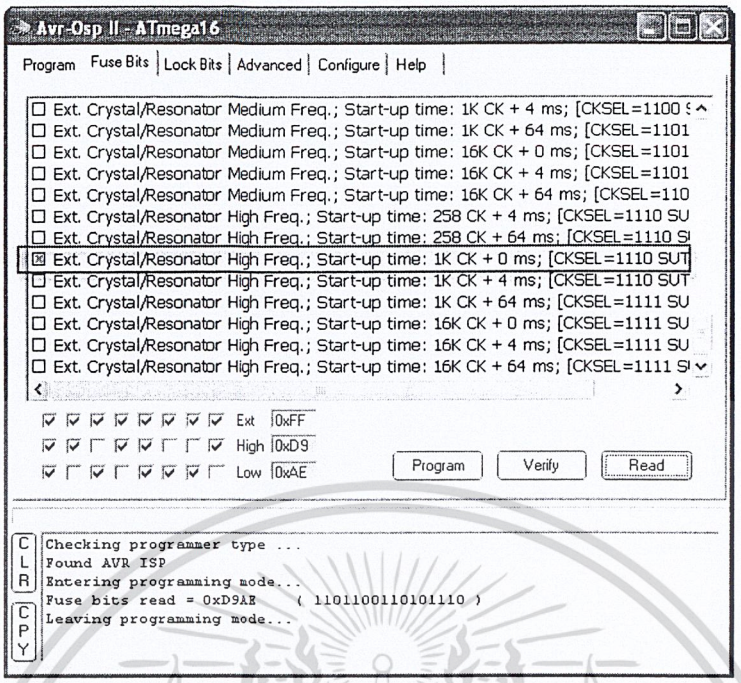
3. จากนั้นเลือกแท็ปมาที่ Fuse Bits และกดปุ่ม Read เพื่ออ่านค่า Fuse Bits มาตรฐานที่ตั้งมาจากโรงงาน ดังรูป



4. จากนั้นทำการตั้งค่า Fuse Bits ให้สอดคล้องกับวงจรที่เราใช้งาน ดังรูป

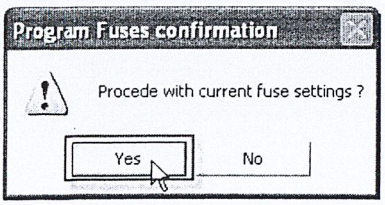


เอกสารนี้มีลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



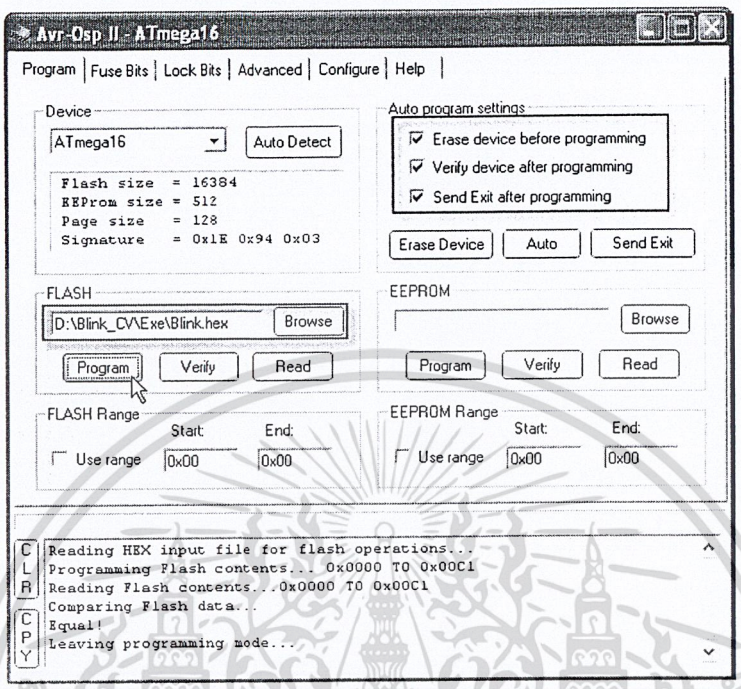
ข้อควรระวัง Fuse bits ที่จะต้องเลือกไว้ตลอดคือ "SPIEN=0" เพราะถ้าไม่เลือกไว้จะทำให้ไม่สามารถโปรแกรมผ่านทาง ISP ได้อีกต่อไป ซึ่งทางแก้ก็คือต้องนำไปแก้ด้วยเครื่องโปรแกรมแรงดันไฟสูง เช่น ALL11 , SUPERPRO เป็นต้น อีกส่วนหนึ่งที่สำคัญรองลงมาคือการเลือกแหล่งของสัญญาณนาฬิกาซึ่งจะต้องเลือกให้สอดคล้องกับวงจรที่ใช้ เช่น ถ้าวงจรใช้คริสตอลเป็นแหล่งกำเนิดสัญญาณนาฬิกา แต่เกิดไปเลือกสัญญาณนาฬิกาเป็นสัญญาณนาฬิกาจากภายนอก(Ext.RC Osc) ก็จะทำให้ครั้งต่อไปไม่สามารถโปรแกรมผ่านทาง ISP ได้อีก ซึ่งจะต้องแก้ด้วยเครื่องโปรแกรมแรงดันไฟสูงตามที่กล่าวมา หรือใช้ ET-AVRProg mini ในการแก้ไขแหล่งของสัญญาณนาฬิกาได้ได้ซึ่งจะกล่าวถึงในตอนท้ายของคู่มือ ดังนั้นการตั้งค่า Fuse bits นี้ผู้ใช้จำเป็นต้องใช้ความละเอียดรอบคอบเป็นอย่างมาก โดยรายละเอียดต่างๆ ของ Fuse bits สามารถดูได้จาก Data Sheet ของแต่ละเบอร์

5. เมื่อตั้งค่าต่างเสร็จเรียบร้อยแล้วก็ให้คลิกปุ่ม Program เพื่อโปรแกรมค่า Fuse Bits จากนั้นโปรแกรมจะถามว่าต้องการโปรแกรมค่า Fuse Bits ที่เลือกไว้ ให้คลิกปุ่ม Yes ดังรูป การโปรแกรมค่า Fuse Bits จะโปรแกรมเพียงครั้งแรกเท่านั้น ครั้งต่อไปก็ไม่จำเป็นต้องโปรแกรมแล้ว

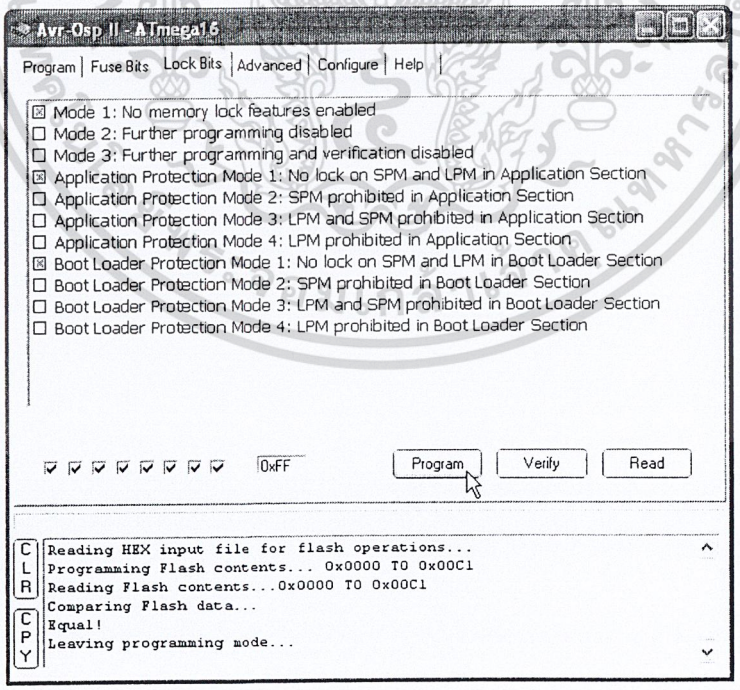


เอกสารนี้มีลิขสิทธิ์ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. จากนั้นทำการเลือก Auto program setting ดังรูป และเลือกตำแหน่งที่เก็บ Hex File ที่ต้องการโปรแกรมจากนั้นคลิกที่ปุ่ม Program



7. ในการกรณีที่ต้องการป้องการอ่านข้อมูลกลับ (ป้องกันการ Copy) ก็สามารถทำได้โดยเลือกมาที่แท็ป Lock Bits แล้วเลือกโหมดการป้องกันตามต้องการและคลิกที่ปุ่ม Program

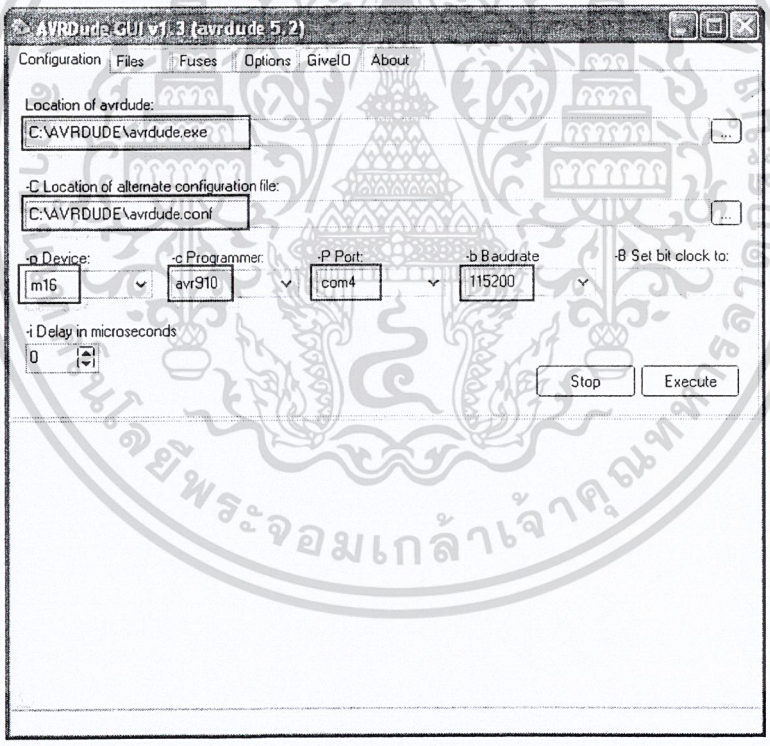


เอกสารนี้มีลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน ET-AVRProg mini ร่วมกับโปรแกรม avrdude

โปรแกรม avrdude นี้ปกติจะติดตั้งมากับโปรแกรม WinAVR มาแล้ว ซึ่งถ้าไม่ได้ติดตั้งโปรแกรม WinAVR ก็สามารถเลือกลงเฉพาะโปรแกรม avrdude ได้จาก CD แผ่นนี้ ข้อเสียของโปรแกรม avrdude คือการใช้งานค่อนข้างยุ่งยากเนื่องจากอินเตอร์เฟสเป็นแบบ Command line เพื่อให้ง่ายสำหรับมือใหม่ตัวอย่างนี้จึงแนะนำให้ใช้งานร่วมกับโปรแกรม AVRDudeGUI ซึ่งออกแบบมาใช้งานร่วมกับโปรแกรม avrdude โดยเฉพาะ ซึ่งการใช้งานมีดังนี้

1. ทำการคัดลอกโฟลเดอร์ชื่อ AVRDUDE ซึ่งอยู่ในโฟลเดอร์ Software ของ CD ROM ไว้ใน Drive C ("C:AVRDUDE\") ซึ่งในโฟลเดอร์นี้ จะประกอบด้วยไฟล์ที่สำคัญ 3 ไฟล์ด้วยกันคือ avrdude.exe, avrdude.conf และ AVRDudegui.exe
2. สั่ง Run Program AVRdudegui.exe แล้วเลือกที่แท็ป ของ Configuration ให้กำหนดค่าต่างๆ ดังต่อไปนี้



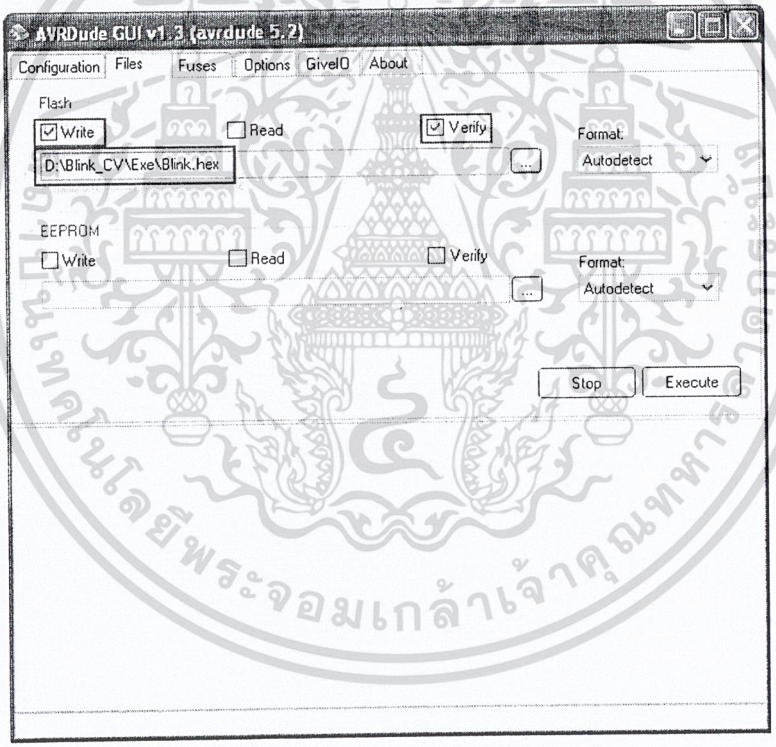
หมายเหตุ ถ้าหมายเลข Comport มากกว่า com8 ให้พิมพ์หมายเลข Comport ลงไปในช่องรับค่าของ -P Port ได้เอง เช่น com9 แต่ถ้าหมายเลข Comport มีค่าเป็น 2 หลักให้เพิ่มเครื่องหมาย "\\.\" นำหน้าชื่อหมายเลข Comport ลงไปด้วย เช่น ถ้าใช้กับ COM13 ก็ให้กำหนดเป็น "\\.\com13" เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ทำการกำหนดค่า Configuration ให้กับโปรแกรม AVRDudeGUI ดังนี้

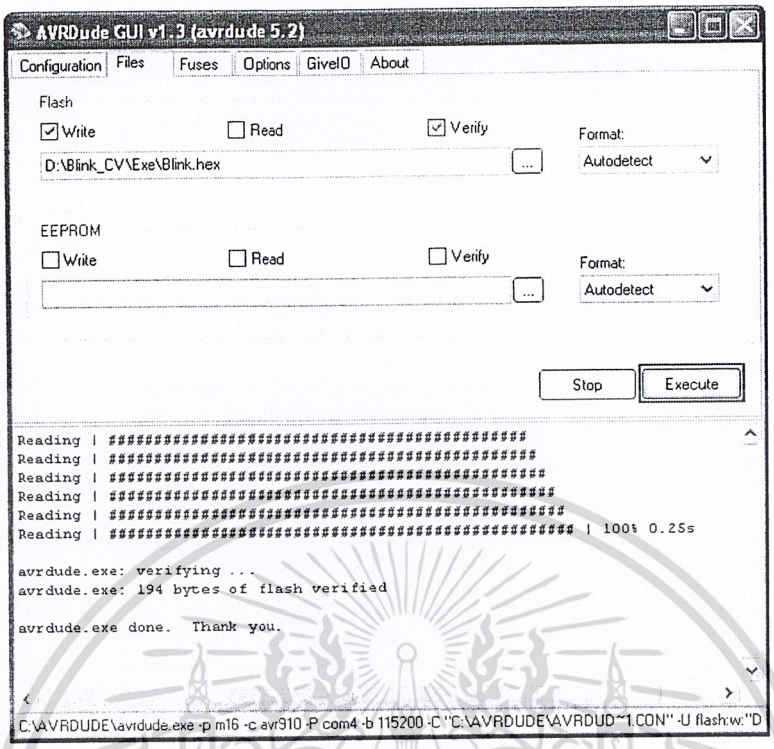
- Location of avrdude ให้เลือกไปที่ชื่อและที่อยู่ของไฟล์ avrdude.exe ที่ได้ติดตั้งไว้ตามขั้นตอนที่ 1 ซึ่งก็คือ "C:\AVRDUDE\avrdude.exe"
- -C Location of alternate configuration file ให้เลือกไปที่ชื่อและที่อยู่ของไฟล์ avrdude.conf ที่ได้ติดตั้งไว้ตามขั้นตอนที่ 1 ซึ่งก็คือ "C:\AVRDUDE\avrdude.conf"
- -p Device ให้เลือกเป็น m16
- -c Programmer ให้เลือกเป็น avr910
- -p Port ให้เลือกเป็นหมายเลข Comport ตามที่ติดตั้งไดร์เวอร์ ของ USB (FTDI) ไว้
- -b Baudrate ให้เลือกเป็น 115200

3. เปลี่ยนแท็ปไปที่ Files แล้วเลือกกำหนดส่วนของ Flash โดยให้เลือก Write และ Verify พร้อมกำหนดชื่อ Hex File ที่ต้องการจะโปรแกรม ดังรูป



4. เมื่อกำหนดค่าต่างๆ ให้กับโปรแกรมเรียบร้อยแล้ว ให้เลือก Execute ที่อยู่ในแท็ปของ Files ซึ่งโปรแกรม avrdude จะเริ่มทำการโปรแกรม Hex File ให้กับบอร์ดทันที ให้รอจนเสร็จดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

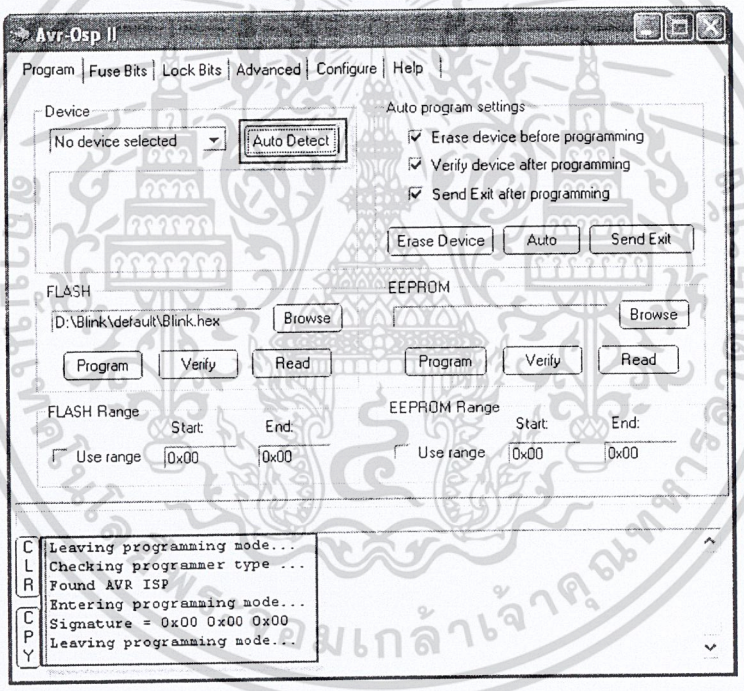


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแก้ไขในกรณีที่เลือก Fuse Bits แหล่งของสัญญาณนาฬิกาผิด

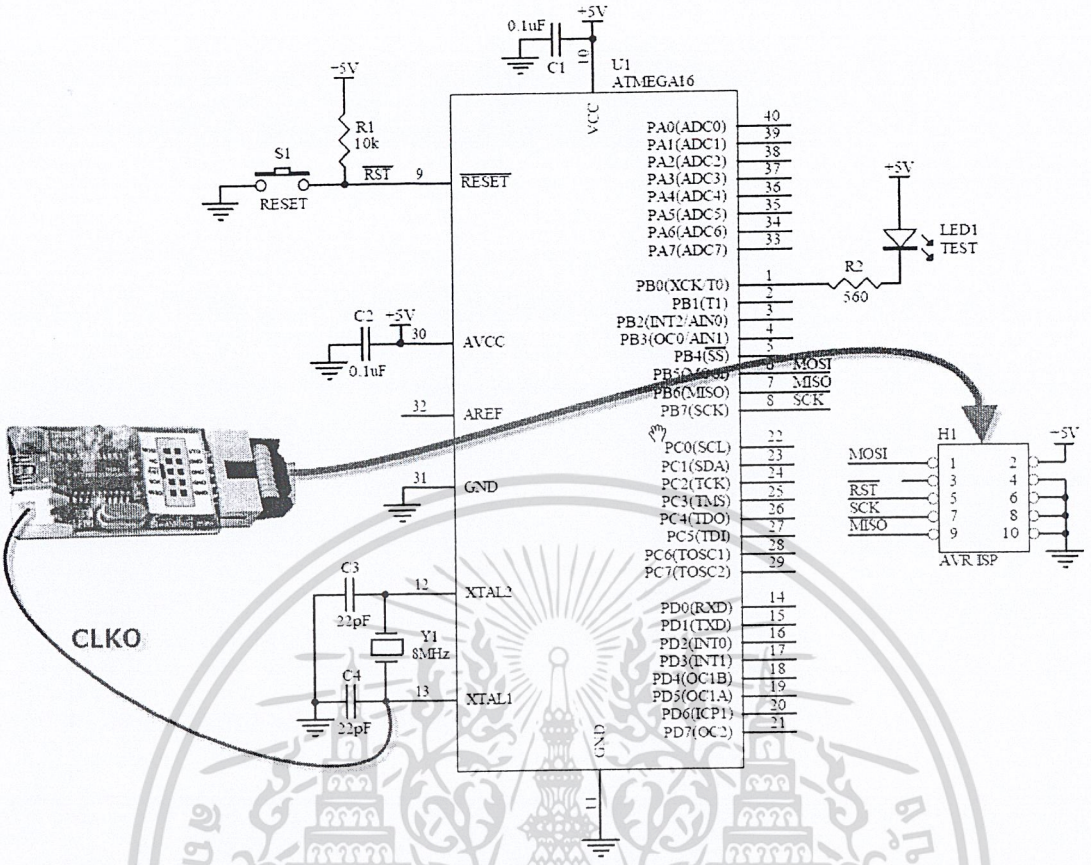
สิ่งที่ผู้ใช้ AVR มือใหม่มักจะประสบคือการโปรแกรม Fuse Bits เลือกสัญญาณนาฬิกาผิด เช่น เลือกเป็น External Clock ซึ่งเป็นการเลือกแหล่งสัญญาณนาฬิกาจากภายนอก ซึ่งถ้าต่อคริสตอลอยู่วงจรก็ไม่สามารถที่จะทำงานได้ เป็นผลให้ไม่สามารถที่จะโหลด Hex File ผ่านทาง ISP ได้ ซึ่งวิธีการแก้ไขก็คือ หาสัญญาณนาฬิกาจากภายนอกป้อนเข้าที่ขา XTAL1 ของ AVR และทำการแก้ไข Fuse Bits ใหม่ให้ถูกต้องและโปรแกรมเข้าไปใหม่ ซึ่งใน ET-AVRProg mini ได้เตรียมสัญญาณนาฬิกา (CLKO)ไว้สำหรับแก้ไข Fuse Bits ในกรณีนี้แล้ว ซึ่งมีวิธีดังนี้

1. อาการที่บ่งบอกว่าอาจจะโปรแกรม Fuse Bits คือ เมื่อเชื่อมต่อ ET-AVRProg mini เข้ากับบอร์ดแล้วกดปุ่ม Auto Detect แล้วไม่เจอ Device ดังรูป

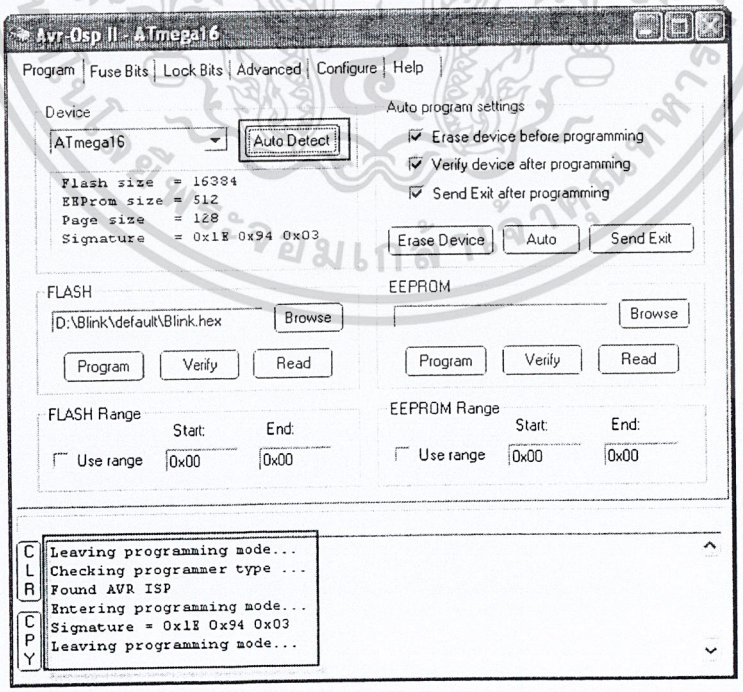


2. จากนั้นทำการต่อสายสัญญาณ CLKO จาก ET-AVRProg ให้กับขา XTAL1 ของไมโครคอนโทรลเลอร์ AVR เพื่อให้วงจรทำงานจากสัญญาณนาฬิกาภายนอก ซึ่งการต่อแสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

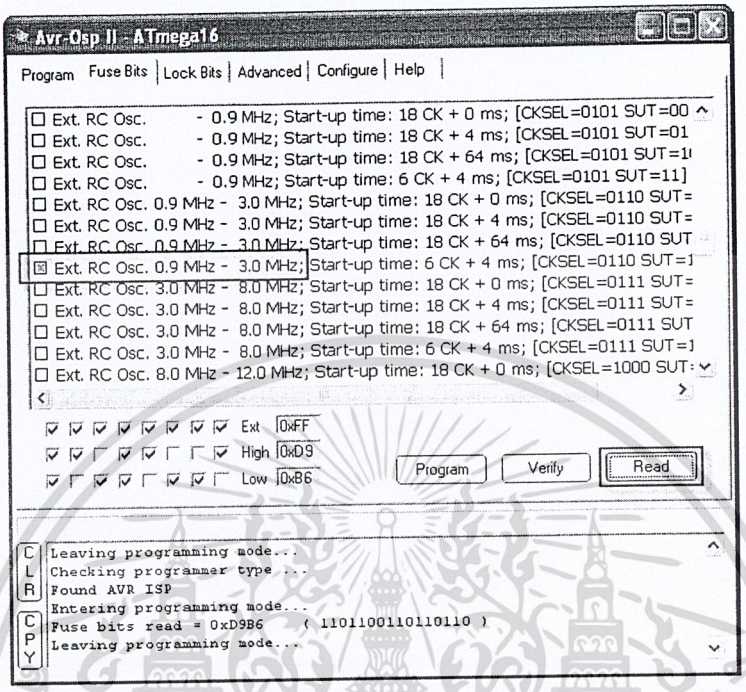


2. จากนั้นคลิกที่ปุ่ม Auto Detect อีกครั้ง ถ้าเกิดจากกรณีที่เลือก Fuse Bits เป็นแหล่งสัญญาณนาฬิกาจากภายนอกจริงก็จะเจอ Device ดังรูป

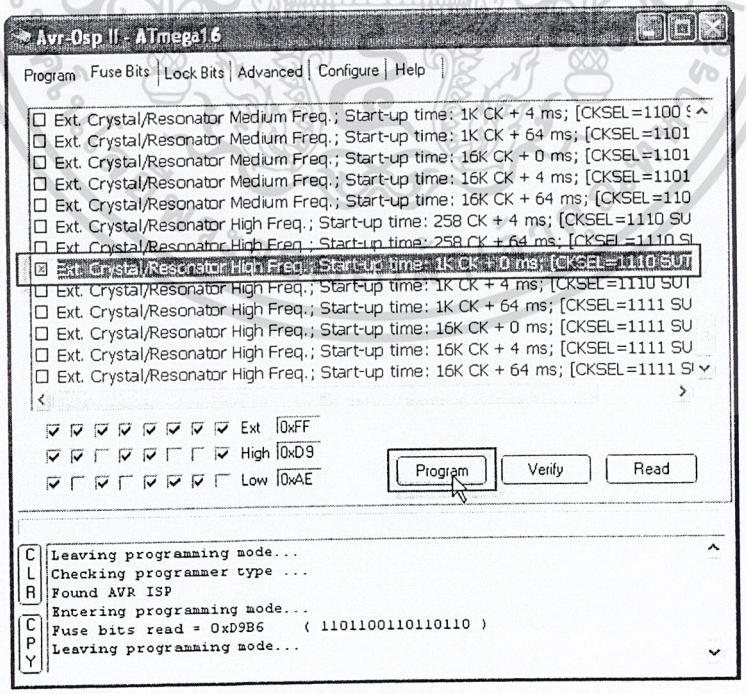


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. จากนั้นเลือกแท็ปมาที่ Fuse Bits และคลิกที่ปุ่ม Read เพื่ออ่านค่า Fuse Bits กลับมา ซึ่งจะเห็นว่าเลือกเป็น Ext.RC Osc. อยู่ ซึ่งก็คือสัญญาณนาฬิกาจะภายนอก



4. ทำการเลือก Fuse Bits ใหม่เป็น Ext. Crytal/Resonator ตามวงจรใช้งาน จากนั้นคลิก Program ดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปดัดแปลงเนื้อหา และนำไปใช้ในการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. จากนั้นเอาสัญญาณ CLKO ออกจากขา XTAL1 ของ AVR และลองกดปุ่ม Auto Detect จะเจอ Device ตามปกติ

หมายเหตุ วิธีไม่สามารถใช้แก้ไขในกรณีที่เลือก Fuse Bits ปิดโหมดของ ISP (SPIEN=0) และในการกรณีที่ ปิดการใช้งานของขา RESET เพื่อใช้เป็น IO ได้ ซึ่ง 2 กรณีนี้ต้องใช้เครื่องโปรแกรมแบบแรงดันไฟสูงในการแก้ไขเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

คู่มือการใช้งาน GSM Module

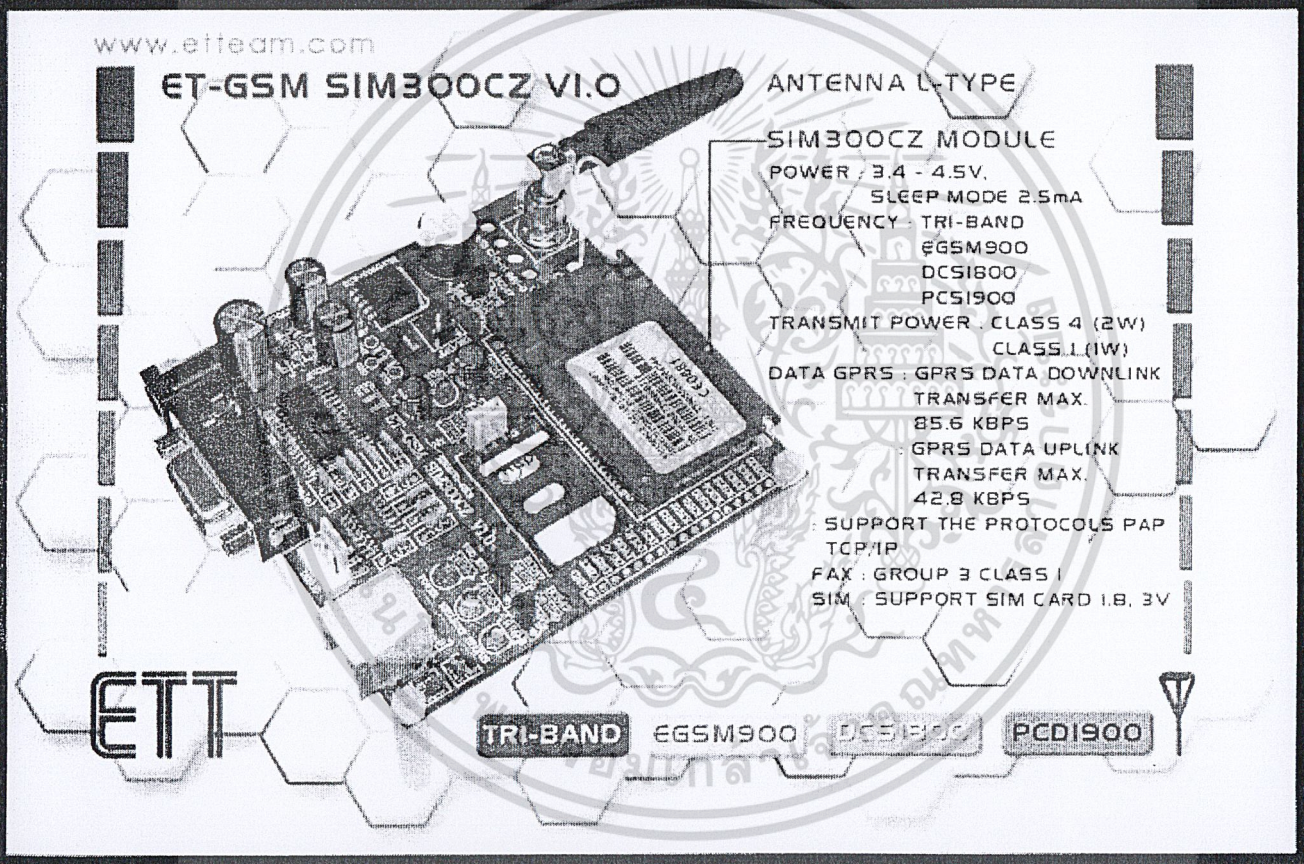
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GSM MODULE

คู่มือการใช้งาน

User's Manual

ET-GSM SIM300CZ V1.0



ETT

www.etteam.com

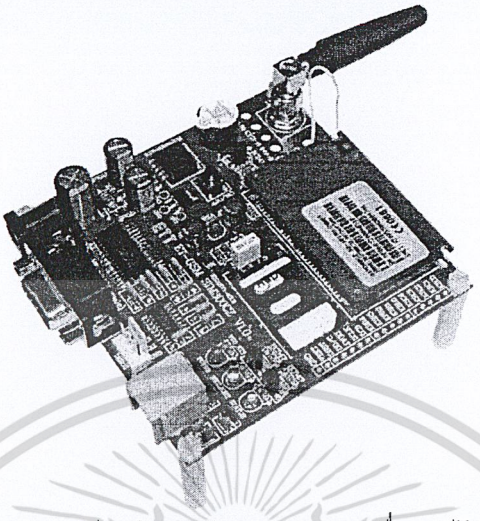
บริษัท อีทีที จำกัด ETT CO., LTD.

1112/96-98 ถนนสุขุมวิท แขวงพระโขนง เขตคลองเตย กรุงเทพฯ 10110 http:/www.etteam.com

1112/96-98 Sukhumvit Rd., Phrakanong Klongtoey Bangkok 10110 http:/www.ett.co.th

Tel : 02-7121120 Fax : 02-3917216 email : sale@etteam.com

ET-GSM SIM300CZ



ET-GSM SIM300CZ เป็นชุดเรียนรู้และพัฒนาระบบการสื่อสารไร้สาย โดยใช้โมดูล GSM/GPRS รุ่น SIM300CZ ของ "SIMCom Ltd." เป็นอุปกรณ์หลัก ซึ่ง SIM300CZ เป็นโมดูลสื่อสารระบบ GSM/GPRS ขนาดเล็ก รองรับระบบสื่อสาร GSM ความถี่ 900/1800/1900MHz โดยสั่งงานผ่านทางพอร์ตสื่อสารอนุกรม RS232 ด้วยชุดคำสั่ง AT Command สามารถประยุกต์ใช้งานได้มากมายหลายรูปแบบ ไม่ว่าจะเป็นการรับส่งสัญญาณแบบ Voice, SMS, Data, FAX และยังรวมถึงการสื่อสารด้วย Protocol TCP/IP ด้วย ซึ่งตามปรกติแล้ว ถึงแม้ว่าโมดูล SIM300CZ จะมีวงจร และ Firmware บรรจุไว้ภายในตัวเป็นที่เรียบร้อยแล้วก็ยังไม่สามารถนำไปใช้งานได้โดยตรงทันที เนื่องจากในการใช้งานจริงๆนั้น ผู้ใช้งานเองจำเป็นต้องออกแบบวงจรรอบนอกที่จำเป็นมาเชื่อมต่อกับขาสัญญาณของตัวโมดูลอีกในบางส่วน ไม่ว่าจะเป็นวงจรภาค Power Supply,วงจรเชื่อมต่อกับ SIM Card รวมไปถึงวงจร Line Driver ของ RS232 เป็นต้น ดังนั้นทางทีมงาน อีทีที จึงได้จัดสร้างบอร์ดสำหรับเป็นตัวกลางในการเชื่อมต่อระหว่างโมดูล SIM300CZ กับอุปกรณ์ภายนอก เพื่อให้ผู้ใช้งานสามารถนำโมดูล GSM ของ SIM300CZ ไปทำการทดลองและศึกษาเรียนรู้การสั่งงานต่างๆ ได้โดยสะดวก ก่อนที่จะนำเอาโมดูลตัวนี้ไปออกแบบดัดแปลงและประยุกต์ใช้งานในด้านต่างๆได้ต่อไปในอนาคต ซึ่งถึงแม้ว่าวงจรการเชื่อมต่อทั้งหมดที่ทาง อีทีที ได้จัดทำขึ้นมานี้จะยังไม่สามารถรองรับการใช้งานทรัพยากรต่างๆที่มีอยู่ภายในโมดูลได้ครบถ้วนทั้งหมดก็ตามที แต่ในส่วนของการใช้งานโมดูลในส่วนที่เป็นความสามารถหลักๆที่จำเป็นนั้นมีไว้รองรับอย่างครบถ้วนเพียงพอแล้ว

อย่างไรก็ตามถ้าผู้ใช้งาน ต้องการพัฒนา Application ที่สูงขึ้นไป ก็สามารถประยุกต์ดัดแปลงหรือทำการเชื่อมต่ออุปกรณ์เพิ่มเติมให้กับบอร์ดได้โดยง่าย ทั้งนี้ก็เพราะว่าขาสัญญาณต่างๆจากโมดูล ในส่วนที่ยังไม่ได้ทำการออกแบบวงจรเตรียมไว้ให้ภายในบอร์ด เช่น ขาสัญญาณสำหรับเชื่อมต่อกับ Keyboard , LCD Display และ GPIO ต่างๆนั้น ทางอีทีที เองก็ได้จัดทำเป็นจุดต่อ Connector เตรียมไว้ให้เป็นที่เรียบร้อยแล้วผู้ใช้เพียงแต่ทำการเชื่อมต่อสัญญาณต่างๆ จากจุดเชื่อมต่อที่เตรียมไว้ไปยังวงจรส่วนที่ได้ทำการออกแบบ ไว้ได้โดยสะดวกอยู่แล้ว

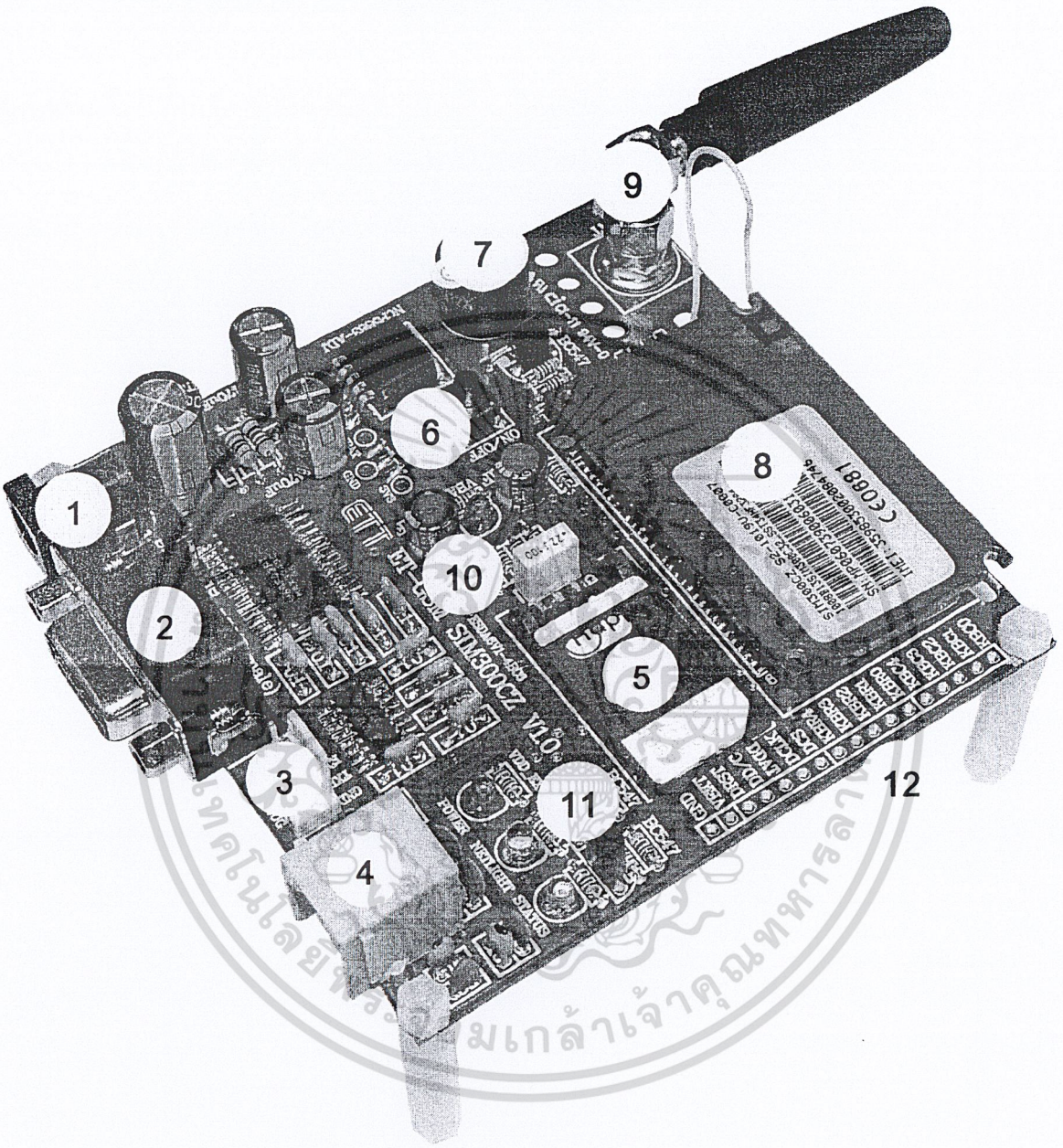
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติของบอร์ด ET-GSM SIM300CZ V1.0

- มีสวิตช์แบบ Push-Button สำหรับใช้สั่ง เปิด-ปิด การทำงานของโมดูลภายในบอร์ด
- มี Socket SIM รองรับ SIM Card พร้อมวงจร ESD ป้องกัน SIM เสียหาย
- มีวงจร Regulate แยกอิสระ จำนวน 2 ชุด สามารถใช้กับแหล่งจ่ายภาย Adapter ขนาดตั้งแต่ +5V ขึ้นไป สามารถจ่ายกระแสให้กับโมดูล SIM300CZ และอุปกรณ์เชื่อมต่อต่างๆได้อย่างเพียงพอ
    - มีวงจร Regulate ขนาด 4.2V / 3A สำหรับจ่ายให้กับโมดูล SIM300CZ ได้อย่างเพียงพอ สามารถใช้กับ SIM ของระบบ GSM900MHz แบบ 2-Watt ได้อย่างไม่เกิดปัญหา
    - มีวงจร Regulate ขนาด 3.3V / 1A สำหรับจ่ายให้กับวงจรเชื่อมต่อภายนอกโดยไม่ต้องไปดึงไฟจากตัวโมดูลมาใช้ ป้องกันปัญหาโมดูลเสียหายจากวงจรภายนอกดึงกระแสเกินพิกัด และสะดวกต่อการออกแบบวงจรเชื่อมต่อเพิ่มเติม โดยไม่ต้องกังวลว่ากระแสจะไม่พอจ่ายให้กับอุปกรณ์
- มีวงจร Line Driver สำหรับแปลงระดับสัญญาณโลจิกจากโมดูล SIM300CZ ให้เป็น RS232 ระดับมาตรฐานครบทุกเส้นสัญญาณ ทั้งพอร์ตที่ใช้ในการสื่อสารสำหรับสั่งงานโมดูล และ พอร์ตสำหรับใช้ในการพัฒนาโปรแกรม (Debug) สามารถเชื่อมต่อกับพอร์ต RS232 มาตรฐานได้ทันที
- มี LED แสดงสถานะพร้อมในบอร์ด สำหรับแสดงสถานะของแหล่งจ่ายไฟ สถานะพร้อมทำงานของโมดูล สถานะในการเชื่อมต่อกับ Network และ สถานะ Power-On/Power-OFF ของโมดูล
- มีขั้วสำหรับเชื่อมต่อกับ Handset (ชุดปากพูด และหูฟัง ของโทรศัพท์บ้าน) โดยใช้ขั้วต่อแบบ RJ11 มาตรฐาน พร้อมวงจร Voice Filter สามารถนำชุด Handset ของโทรศัพท์บ้าน ต่อเข้ากับบอร์ดทางขั้วต่อแบบ RJ11 สำหรับใช้พูดคุย โทรออก และ รับสายได้โดยสะดวก
- มี Buzzer พร้อมวงจรขับเพื่อสร้างสัญญาณเสียง ในกรณีมีการโทรเรียกเข้ามายังโมดูล
- มีจุดยึดเสาอากาศ สำหรับใช้เป็นจุดพักสำหรับเชื่อมต่อกับเสาอากาศแบบต่างๆได้โดยสะดวก
- มีขั้วต่อสำหรับติดตั้งโมดูล SIM300CZ พร้อมเสารองและสกรูยึดโมดูลกับตัวบอร์ด
- มีจุดต่อสัญญาณอื่นๆที่เหลือจากโมดูล เช่น Keyboard, Display ,GPIO ,Battery Charger ฯลฯ สำหรับให้ผู้ใช้ต่อขยายไปยังวงจรที่ออกแบบเพิ่มเติมได้โดยง่ายและสะดวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างของบอร์ด ET-GSM SIM300CZ V1.0



- **หมายเลข 1** เป็น JACK DC-IN แบบมีขั้ว โดยมีด้านนอกเป็นขั้วบวก และด้านในเป็น GND ใช้สำหรับรับแหล่งจ่ายไฟจากภายนอกโดยออกแบบให้ใช้กับ แหล่งจ่ายไฟขนาด 5V ขึ้นไปที่จ่ายกระแสได้ 1A ถึง 3A
- **หมายเลข 2** เป็น ขั้วต่อ RS232(DCE) แบบ DB9 ตัวเมีย สำหรับใช้เชื่อมต่อกับสัญญาณ RS232(DTE) แบบ DB9 ตัวผู้ จากคอมพิวเตอร์ PC หรืออุปกรณ์ภายนอกอื่นๆ โดยใช้สาย 9 Pin แบบต่อตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **หมายเลข 3** เป็น ขั้วต่อ DEBUG ใช้สำหรับพัฒนา และ DEBUG โปรแกรม สำหรับต่อกับ RS232 ในกรณีที่ต้องการพัฒนาโปรแกรมเพิ่มเติมให้กับโมดูล SIM300CZ เอง
- **หมายเลข 4** เป็น ขั้วต่อ RJ11 สำหรับใช้เชื่อมต่อกับชุด Handset ในกรณีที่ต้องการใช้งานโมดูล SIM300CZ เพื่อโทรออกและรับสาย โดยสามารถเชื่อมต่อกับ Handset มาตรฐานได้ทั่วไป
- **หมายเลข 5** เป็น Socket สำหรับติดตั้ง SIM Card ให้กับโมดูล
- **หมายเลข 6** เป็น Switch Push-Button สำหรับใช้ Power-On และ Power-OFF ตัวโมดูล
- **หมายเลข 7** เป็น Buzzer สำหรับสร้างเสียงเรียกเข้าในกรณีที่มีการโทรเข้ามายังโมดูล SIM300CZ
- **หมายเลข 8** เป็น จุดรองรับโมดูล SIM300CZ พร้อมเสาและสกรูสำหรับยึดโมดูลกับบอร์ด
- **หมายเลข 9** เป็น จุดยึด Connector เสาอากาศ GSM/GPRS ย่านความถี่ 900/1800/1900MHz
- **หมายเลข 10** เป็น LED แสดงแหล่งจ่าย VBAT โดยจะติดสว่างเมื่อมีการจ่ายไฟให้บอร์ดแล้ว
- **หมายเลข 11** เป็น LED แสดงสถานะของบอร์ด ซึ่งมีด้วยกัน 3 ดวงคือ
  - POWER สีแดง จะติดสว่าง เมื่อโมดูลอยู่ในสถานะ Power-ON
  - NETLIGHT สีเหลือง จะกระพริบเมื่อโมดูลอยู่ในสถานะ Power-ON
  - STATUS สีเขียว จะติดสว่างเมื่อโมดูลอยู่ในสถานะ Power-ON
- **หมายเลข 12** เป็น จุดต่อสัญญาณเพิ่มเติมในกรณีที่ต้องการประยุกต์ใช้งานโมดูลเพิ่มเติม

คุณสมบัติของโมดูล SIM300CZ

- รองรับความถี่ GSM/GPRS 900/1800/1900MHz
- รองรับ GPRS Multi-Slot Class10 และ GPRS Mobile Station Class B
- รองรับมาตรฐานคำสั่ง AT Command (GSM 07.07 / 07.05 และคำสั่งเพิ่มเติมจาก SIMCOM)
- รองรับ SIM Applications Toolkit
- ทำงานที่ย่านแรงดัน 3.4V ถึง 4.5V
- รองรับการเชื่อมต่อภายนอก
  - ใช้ได้กับ SIM 3V และ 1.8V
  - มีวงจร Analog Audio (MIC & Speaker) จำนวน 2 ชุด
  - รองรับ 5x5 Keypad Interface & SPI LCD Interface
  - มีระบบ RTC พร้อมวงจร Backup
  - มีขั้วต่อเสาอากาศภายนอกแบบ Connector และจุดเชื่อมต่อแบบ PAD
  - มีระบบ Battery Charge ในตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์แสดงการทำงานของโมดูล SIM300CZ

สำหรับบอร์ด ET-GSM SIM300CZ V1.0 นั้น ได้ออกแบบอุปกรณ์แสดงผลการทำงานของบอร์ดไว้ในบอร์ดเพื่อใช้แสดงสถานะของการทำงานต่างๆให้ผู้ใช้ทราบด้วย คือ

- Buzzer ใช้แสดงการทำงานของโมดูลเมื่อมีสายเรียกเข้า โดยการทำงานของ Buzzer นี้จะถูกควบคุมด้วยสัญญาณ BUZZER (Pin23) ของโมดูล SIM300CZ และสามารถปรับระดับความดังของเสียงได้จากคำสั่ง "AT+CRSL" ได้อีกด้วย
- **LED VBAT** ใช้ทำหน้าที่แสดงสถานะของแหล่งจ่ายไฟจากภายนอกที่ต่อมาให้กับบอร์ด โดย LED นี้จะติดสว่างก็ต่อเมื่อมีการจ่ายไฟให้กับบอร์ดเป็นที่เรียบร้อยแล้ว
- **LED POWER** ใช้แสดงสถานะความพร้อมของโมดูล SIM300CZ ว่าอยู่ในสภานะ Power ON หรือ Power OFF โดย LED ตัวนี้จะถูกควบคุมการทำงานด้วยสัญญาณ VDD_EXT(Pin15) ของโมดูล เมื่อทำงานจะมีสถานะทางโลจิกเป็นโลจิก "1" โดยถ้า LED Power ติดสว่าง แสดงว่า โมดูล SIM300CZ อยู่ในสถานะ Power ON และพร้อมทำงาน แต่ถ้า LED นี้ดับ แสดงว่าโมดูล อยู่ในสถานะ Power OFF อยู่
- **LED NETLIGHT** ใช้แสดงสถานะของโมดูล ในขณะทำการเชื่อมต่อกับเครือข่ายอยู่ โดย LED ตัวนี้จะถูกควบคุมด้วยสัญญาณ NETLIGHT(Pin16) ของโมดูล SIM300CZ เมื่อทำงานจะมีสถานะทางโลจิกเป็นโลจิก "1" โดยเมื่อโมดูลอยู่ในสถานะพร้อมทำงาน LED นี้จะติดกระพริบด้วยค่าความเร็วต่างๆ ซึ่งมีความหมายดังนี้
  - OFF แสดงว่าโมดูลอยู่ในสถานะของ Power OFF (ไม่ทำงาน)
  - 64mS ON / 800mS OFF แสดงว่า โมดูล SIM300CZ ทำงานปรกติและไม่ได้อยู่ระหว่างทำการค้นหาเครือข่ายอยู่
  - 64mS ON / 3000mS OFF แสดงว่าโมดูล SIM300CZ กำลังทำการค้นหาเครือข่าย เพื่อทำการเชื่อมต่อสัญญาณ
  - 64mS ON / 300mS OFF แสดงว่าโมดูล SIM300CZ อยู่ระหว่างการเชื่อมต่อกับเครือข่ายหรืออุปกรณ์อื่นๆด้วย GPRS อยู่
- **LED STATUS** ใช้แสดงสถานะของโมดูล SIM300CZ ว่าพร้อมทำงานหรือไม่ โดย LED ตัวนี้จะถูกควบคุมด้วยสัญญาณ STATUS(Pin19) ของโมดูล SIM300CZ เมื่อทำงานจะมีสถานะทางโลจิกเป็นโลจิก "1" ซึ่งเมื่อ LED นี้ติดสว่าง แสดงว่าโมดูลพร้อมรับคำสั่งต่างๆได้ แต่ถ้า LED ดับแสดงว่าโมดูลยังไม่พร้อมทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสั่ง เปิด และ ปิด การทำงานของโมดูล

ตามปรกติแล้ว โมดูล SIM300CZ จะมีโหมดการทำงานอยู่หลายโหมด เราสามารถทำงานสั่ง เปิด และ ปิดการทำงานของโมดูลได้ หลายวิธี

- Switch ON/OFF เป็นการสั่ง เปิด และ ปิด การทำงานของโมดูล SIM300CZ ด้วยการกดสวิตช์ โดยสวิตช์ตัวนี้ จะเป็นแบบ Push-Button Switch (สวิตช์ กดติด-ปล่อยดับ) โดยเป็นการกำหนดสถานะทางโลจิกให้กับขาสัญญาณ PWRKEY(Pin17) ของโมดูล โดยเมื่อกดสวิตช์จะเป็นโลจิก "0" เมื่อปล่อยสวิตช์จะเป็นโลจิก "1" โดยการทำงานของสวิตช์จะต้องทำการกดสวิตช์ต่อเนื่องกันเป็นเวลานานอย่างน้อย 2000mS (2 วินาที) จึงจะมีผลต่อการทำงานของโมดูล โดยลักษณะการทำงานของสวิตช์ จะเป็นแบบ Toggle กล่าวคือ ถ้าโมดูลอยู่ในสถานะของ Power OFF อยู่ แล้วทำการกดสวิตช์ เป็นเวลาอย่างน้อย 2000mS (2 วินาที) จะเป็นการสั่งให้โมดูลกลับเข้าสู่ Power On หรือพร้อมทำงาน แต่ถ้าหากว่าโมดูลอยู่ในสถานะของ Power ON อยู่ แล้วทำการกดสวิตช์ เป็นเวลาอย่างน้อย 2000mS (2 วินาที) แล้วปล่อยจะเป็นการสั่งให้โมดูลหยุดทำงานและกลับเข้าสู่สถานะของ Power OFF (หยุดทำงาน)

LED สถานะ	Power-ON	Power-OFF
VBAT (แดง)	ติดสว่าง	ติดสว่าง
POWER (แดง)	ติดสว่าง	ดับ
NETLIGHT (เหลือง)	กระพริบ	ดับ
STATUS (เขียว)	ติดสว่าง	ดับ

ตาราง แสดงสถานะของ LED ในโหมดต่างๆ

หลังจากทำการสั่ง Power-ON ในครั้งแรกนั้น ก่อนที่จะเริ่มต้นส่งคำสั่งใดๆให้กับโมดูล ควรรอให้ตัวโมดูลพร้อมเสียก่อน โดยจะมีข้อความ "Call Ready" ปรากฏให้เห็น ในกรณีที่กำหนด Baudrate เป็นแบบ Auto Baudrate ไว้ (AT+IPR=0") เมื่อทำการ Power-ON จะได้ผลดังตัวอย่าง

```
Call Ready
```

ในกรณีที่กำหนด Baudrate เป็นแบบ Fix Baudrate ไว้(AT+IPR=ค่า Baudrate) เมื่อทำการ สั่งให้โมดูล Power-ON แต่ละครั้งจะได้ผลดังตัวอย่าง

```
RDY
+CFUN: 1
+CPIN: READY
Call Ready
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การติดต่อสื่อสารกับโมดูล SIM300CZ

การติดต่อสื่อสารกับโมดูล SIM300CZ ของบอร์ด ET-GSM SIM300CZ นั้นจะเชื่อมต่อผ่านพอร์ตสื่อสารอนุกรม RS232 โดยใช้ขั้วต่อแบบ DB9 ตัวเมีย จัดเรียงสัญญาณตามมาตรฐาน RS232-DCE สามารถนำไปเชื่อมต่อกับสัญญาณ RS232-DTE มาตรฐาน โดยใช้สาย DB9 แบบต่อตรง ได้ทันที โดยสัญญาณทั้งหมดที่ DB9 นี้ได้ผ่านวงจร Line Driver เพื่อแปลงสัญญาณระดับ โลจิก จากโมดูล ให้เป็นสัญญาณระดับมาตรฐาน RS232 เป็นที่เรียบร้อยแล้ว ซึ่งถ้าต้องการนำไปเชื่อมต่อกับ RS232(Com Port) ของคอมพิวเตอร์ PC ก็สามารถทำการเชื่อมต่อกันโดยตรงได้ทันที โดยไม่ต้องทำการสลับสายสัญญาณใดๆ ทั้งสิ้น โดยสัญญาณเชื่อมต่อทางด้านโมดูล SIM300CZ นั้น จะมีทั้งหมด 8 เส้นสัญญาณ ซึ่งในการเชื่อมต่อใช้งานนั้น จะต่อให้ครบทั้ง 8 เส้น หรือ จะเลือกต่อเพียง 3 เส้น (RXD,TXD และ GND) ก็ได้เช่นเดียวกัน โดยสามารถกำหนดได้จากการ Setup ค่า Configuration และคำสั่งใช้งาน โดยสัญญาณการเชื่อมต่อ RS232 ด้านโมดูล SIM300CZ จะมีดังนี้

- Pin1 เป็นขา DCD (Data Carrier Detect) ของโมดูล SIM300CZ ซึ่งเป็น Output จาก SIM300CZ ที่ได้ผ่านการแปลงระดับสัญญาณเป็น RS232 แล้ว ซึ่งตามปรกติจะต่อเข้ากับ DCD Input ของอุปกรณ์ด้าน Host หรือคอมพิวเตอร์ PC
- Pin2 เป็นขา TXD(Transmit Data) ของโมดูล SIM300CZ ซึ่งเป็น Output จาก SIM300CZ ที่ได้ผ่านการแปลงระดับสัญญาณเป็น RS232 แล้ว ซึ่งตามปรกติจะต่อเข้ากับ RXD(Receive Data) ของอุปกรณ์ด้าน Host หรือคอมพิวเตอร์ PC
- Pin3 เป็นขา RXD (Receive Data) ของโมดูล SIM300CZ ซึ่งเป็น Input ของ SIM300CZ สามารถรับสัญญาณระดับ RS232 ได้โดยตรง ซึ่งตามปรกติจะต่อเข้ากับ TXD(Transmit Data) จากอุปกรณ์ด้าน Host หรือคอมพิวเตอร์ PC
- Pin4 เป็นขา DTR(Data Terminal Ready) ของโมดูล SIM300CZ ซึ่งเป็น Input ของ SIM300CZ ซึ่งตามปรกติจะต่อเข้ากับ DTR จากอุปกรณ์ด้าน Host หรือคอมพิวเตอร์ PC
- Pin5 เป็นสัญญาณ GND ของโมดูล SIM300CZ ต้องต่อเข้ากับ GND ของอุปกรณ์ด้าน Host หรือคอมพิวเตอร์ PC
- Pin6 ตามปรกติแล้วเป็นสัญญาณ DSR (Data Set Ready) แต่ในกรณีของ SIM300CZ จะไม่ได้ต่อใช้งาน แต่อย่างไรก็ตาม ในบอร์ดได้ทำการป้อนสัญญาณย้อนกลับหรือ Loop Back สัญญาณ DTR (Data Terminal Ready) ซึ่งเป็น Output ส่งมาจาก Host หรือ คอมพิวเตอร์ PC กลับไปแทน โดยจะถูกต่อไปเข้ากับสัญญาณ DSR Input ของอุปกรณ์ด้าน Host หรือคอมพิวเตอร์ PC
- Pin7 เป็นขาสัญญาณ RTS (Request To Send) ของโมดูล SIM300CZ ซึ่งเป็น Input ของ SIM300CZ ซึ่งตามปรกติจะต่อเข้ากับ RTS ของอุปกรณ์ด้าน Host หรือคอมพิวเตอร์ PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า แสวงหากำไร ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Pin8 เป็นขาสัญญาณ CTS (Clear To Send) ของโมดูล SIM300CZ ซึ่งเป็น Output จาก SIM300CZ ซึ่งตามปรกติจะต่อเข้ากับ CTS ของอุปกรณ์ด้าน Host หรือคอมพิวเตอร์ PC
- Pin9 เป็นขาสัญญาณ RI(Ring Indicator) ของโมดูล SIM300CZ ซึ่งเป็น Output จาก SIM300CZ ซึ่งตามปรกติจะต่อเข้ากับ RI ของอุปกรณ์ด้าน Host หรือ คอมพิวเตอร์ PC

DB9 Female(SIM300CZ)		Signal Direction	DB9 Male(Computer PC)	
Pin	Signal		Signal	Pin
1	DCD	→	DCD	1
2	TXD	→	RXD	2
3	RXD	←	TXD	3
4	DTR	←	DTR	4
5	GND	—	GND	5
6	(DSR)	→	DSR	6
7	RTS	←	RTS	7
8	CTS	→	CTS	8
9	RI	→	RI	9

แผนผัง แสดงการต่อสายสัญญาณระหว่าง ET-GSM SIM300CZ กับ คอมพิวเตอร์ PC

DB9 Female(SIM300CZ)		Signal Direction	ไมโครคอนโทรลเลอร์
Pin	Signal		Signal
2	TXD	→	RXD
3	RXD	←	TXD
5	GND	—	GND

แผนผัง แสดงการต่อสายสัญญาณระหว่าง ET-GSM SIM300CZ กับ ไมโครคอนโทรลเลอร์

<u>คำแนะนำ</u> ในกรณีที่ใช้การเชื่อมต่อสัญญาณแบบ 3 เส้น (RXD,TXD,GND) ต้องกำหนดเงื่อนไขของ Flow Control ให้กับโมดูล SIM300CZ เป็น XON/XOFF โดยใช้คำสั่ง "AT+IFC=1,1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติการทำงานของสัญญาณที่ควรรู้

- **RI(Ring Indicator)** เป็น Output จากโมดูล SIM300CZ ตามปรกติจะเป็น High แต่เมื่อมีสัญญาณเรียกเข้าจะ Active เป็น Low ตามเงื่อนไขต่อไปนี้
  - เมื่อมีสัญญาณเรียกเข้า Voice Calling สัญญาณ RI จะ Active เป็น LOW ค้างอยู่จนกว่าจะมีการตอบรับ(ATA) หรือ ได้รับคำสั่งยกเลิกการเชื่อมต่อ(ATH) หรือผู้เรียกสายทำการวางสายก่อนจะมีการตอบรับ
  - เมื่อมีสัญญาณเรียกเข้า Data Calling สัญญาณ RI จะ Active เป็น LOW ค้างอยู่จนกว่าจะมีการตอบรับ(ATA) หรือ ได้รับคำสั่งยกเลิกการเชื่อมต่อ (ATH)
  - เมื่อมีสัญญาณเรียกเข้า SMS สัญญาณ RI จะ Active เป็น LOW ประมาณ 120mS และกลับเป็น HIGH โดยอัตโนมัติ
- **DTR(Data Terminal Ready)** เป็น Input ของโมดูล SIM300CZ เมื่อต้องการให้โมดูลทำงานต้องให้ขาสัญญาณนี้ได้รับโลจิก LOW ถ้าขา DTR ได้รับโลจิก HIGH โมดูลจะหยุดทำงานและเข้าสู่ Sleep Mode โดยอัตโนมัติ (ถ้ามีการสั่ง Enable Sleep Mode ด้วยคำสั่ง "AT+CSCLK=1" ไว้) ดังนั้นถ้าต้องการให้โมดูลทำงานตลอดเวลาต้องควบคุมให้ขาสัญญาณ DTR ด้านโมดูลได้รับโลจิก LOW โดยการควบคุมจากสัญญาณ DTR ด้านคอมพิวเตอร์ PC หรืออุปกรณ์ที่ควบคุมโมดูลอยู่ให้ทำการ Active สัญญาณ DTR ไว้ตลอดการเชื่อมต่อ สำหรับกรณีที่นำโมดูล SIM300CZ ไปเชื่อมต่อกับ RS232 ระบบที่ไม่มีสัญญาณ DTR อยู่เช่น RS232 ของไมโครคอนโทรลเลอร์บางรุ่น ก็อาจเลือกกำหนด Jumper (DTR) ที่อยู่ใกล้กับขั้วต่อ DB9 บนบอร์ดไว้ทางด้าน GND เพื่อให้โมดูลทำงานตลอดเวลา หรือสั่งปิดการทำงานของ Sleep Mode โดยใช้คำสั่ง "AT+CSCLK=0" แล้วบันทึกค่า Configuration นี้ไว้ก็ได้เช่นเดียวกัน
- **ADC0(Analog to Digital)** เป็น Input แบบ ADC (ขา 12 ของโมดูล SIM300CZ) สามารถรับสัญญาณ Analog จากภายนอกได้ระหว่าง 0V ถึง 2.4V โดยสามารถสั่งอ่านค่าระดับแรงดันที่ขานี้ได้จากคำสั่ง "AT+CADC?" โดยจะได้ผลลัพธ์เป็นค่าระหว่าง 0 ถึง 2400
- **GPIO0:GPIO1** เป็นขาสัญญาณ I/O สามารถเชื่อมต่อกับสัญญาณโลจิกระดับ 3.3V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการใช้งาน AT Command เพื่อสั่งงานโมดูล SIM300CZ

โมดูล GSM/GPRS รุ่น SIM300CZ ถูกออกแบบให้ทำหน้าที่เหมือน Modem โดยจะใช้การติดต่อสั่งงานและสื่อสารกับโมดูล ผ่านทางพอร์ตสื่อสาร RS232 รองรับ Baudrate ตั้งแต่ 1200-115200 BPS โดยใช้ชุดคำสั่งแบบ AT Command ซึ่งจะมีรูปแบบการใช้งานเหมือนกับ Modem มาตรฐานทั่วๆไป เพียงแต่จะมีการเพิ่มเติม Option และคำสั่งพิเศษอื่นๆเพิ่มเติมขึ้นมาอีก เพื่อให้เหมาะสมและสอดคล้องกับความสามารถในการทำงานของโมดูลได้อย่างครบถ้วน

สำหรับรายละเอียดการใช้คำสั่ง AT Command ที่จะใช้ สำหรับติดต่อสั่งงานโมดูล SIM300CZ ไม่ว่าจะเป็น รูปแบบคำสั่ง และ หน้าที่การทำงานของแต่ละคำสั่ง ผู้ใช้สามารถศึกษารายละเอียดต่างๆได้จากคู่มือคำสั่ง AT Command (ไฟล์เอกสารชื่อ SIM300C_ATC_V1.06.PDF) ในแผ่น CD-ROM ซึ่งในที่นี้จะขอแนะนำถึงวิธีการและรูปแบบการใช้งานคำสั่งแบบย่อๆ แบบพอสังเขป เพื่อเป็นแนวทางให้ผู้เริ่มต้นได้ใช้เป็นแนวทางและประกอบความเข้าใจในการศึกษาการทำงานของคำสั่งต่างๆต่อไป โดยรูปแบบของคำสั่งต่างๆที่เป็น AT Command นั้น จะเริ่มต้นคำสั่งด้วยรหัส ASCII ของตัวอักษร 2 ตัว คือ "A" และ "T" ซึ่งจะใช้ตัวอักษรแบบพิมพ์เล็ก หรือ พิมพ์ใหญ่ก็ได้ มีความหมายเหมือนกัน จากนั้นก็จะตามด้วยรหัสคำสั่ง และ Option ต่างๆของคำสั่ง(ถ้ามี) โดยทุกๆคำสั่งจะต้องจบด้วยรหัส Enter หรือ 0DH (13) เสมอ เช่นคำสั่ง รีเซ็ต จะใช้รูปแบบคำสั่งเป็น "ATZ" หรือ "atz" ก็สามารถใช้งานได้ถูกต้องเหมือนกัน โดยรูปแบบคำสั่งทั้งหมดจะแบ่งออกเป็น 4 แบบด้วยกัน คือ

การใช้งาน	รูปแบบคำสั่ง	รายละเอียด
ทดสอบคำสั่ง	AT+<x>=?	รูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งอ่านค่ารูปแบบและพารามิเตอร์ต่างๆของคำสั่ง โดยถ้าคำสั่งนั้นมีอยู่จริง โมดูลจะตอบรับด้วยการพิมพ์ค่าของพารามิเตอร์ต่างๆของคำสั่งที่มีอยู่ทั้งหมดให้ทราบ
อ่านค่าพารามิเตอร์	AT+<x>?	รูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งอ่านค่าพารามิเตอร์ที่กำหนดไว้แล้วของคำสั่งนั้นๆ โดยโมดูลจะตอบรับด้วยการพิมพ์ค่าพารามิเตอร์ปัจจุบันที่กำหนดไว้แล้วให้ทราบ
กำหนดค่าการทำงาน	AT+<x>=<...>	รูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งเขียนหรือกำหนดค่าพารามิเตอร์ให้กับคำสั่ง เช่น การกำหนดค่า Baudrate
สั่งให้ทำงาน	AT+<x>	รูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งงานให้โมดูลปฏิบัติตามคำสั่งที่ต้องการ เช่น การสั่งรีเซ็ต (ATZ)

ตารางแสดง รูปแบบการใช้งาน AT Command (เมื่อ <x> คือ รหัสคำสั่ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า แสวงหากำไร ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบการสั่งงานโมดูล

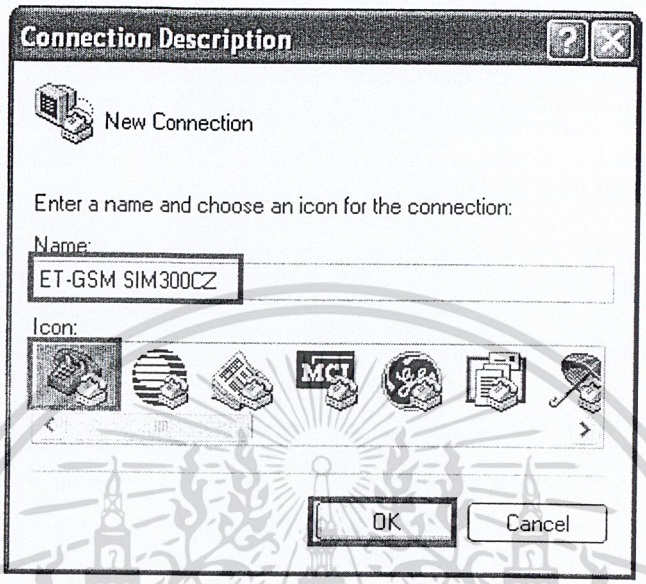
ดังได้ทราบแล้วว่าในการสั่งงานโมดูล SIM300CZ นั้น จะใช้วิธีการส่งค่ารหัสคำสั่งในรูปแบบของ AT Command ผ่านทางพอร์ตสื่อสารอนุกรมไปให้กับโมดูล ซึ่งตามปรกติจะต้องเขียนโปรแกรมเพื่อส่งรหัสคำสั่งต่างๆไปให้กับโมดูลเอง ขึ้นอยู่กับว่าจะใช้อุปกรณ์ใดเป็นตัวควบคุมการทำงานของโมดูล ซึ่งไม่ได้จำกัดว่าเป็นอุปกรณ์แบบใด อาจจะเป็นคอมพิวเตอร์ PC หรือ ไมโครคอนโทรลเลอร์ตระกูลใดๆก็ได้ ขอให้มีพอร์ตสื่อสารอนุกรม RS232 อยู่ก็สามารถนำมาเชื่อมต่อเพื่อสั่งงานโมดูล SIM300CZ ได้แล้ว ส่วนที่ว่าจะเขียนโปรแกรมอย่างไร และจะใช้ภาษาใดในการเขียนนั้น ขึ้นอยู่กับผู้พัฒนาโปรแกรมว่า มีความถนัดอย่างไรและมีพื้นฐานอะไรอยู่บ้าง ซึ่งหลักสำคัญก็คือ ผู้พัฒนาต้องหาคำตอบให้ได้ว่า การจะเขียนโปรแกรมสั่งงานอุปกรณ์ทำการ ส่ง และ รับ ข้อมูล ด้วยพอร์ตสื่อสารอนุกรม RS232 นั้นจะต้องทำอย่างไร ซึ่งจะไม่ขอกล่าวถึงในที่นี้ด้วย

สำหรับในการศึกษาเบื้องต้นนั้น ยังไม่จำเป็นต้องใช้วิธีการเขียนโปรแกรมก็ได้ แต่สามารถใช้โปรแกรมสำเร็จรูปจำพวก Serial Terminal ต่างๆของคอมพิวเตอร์เป็นตัวทดสอบการทำงานเพื่อทำความเข้าใจกับรูปแบบคำสั่งและผลของการทำงานต่างๆให้เข้าใจเสียก่อน ตัวอย่างเช่น ถ้าต้องการจะสั่งให้โมดูล SIM300CZ โทรออกไปยังโทรศัพท์มือถือหมายเลข 0811234567 นั้น ในอันดับแรกจะต้องศึกษารูปแบบการทำงานของคำสั่งให้เข้าใจเสียก่อน จนสามารถเข้าใจแล้วว่าจะต้องใช้คำสั่ง "ATD0811234567;" เพื่อสั่งให้โทรออก จากนั้นจึงค่อยปรับเปลี่ยนไปเป็นการเขียนโปรแกรมในภายหลัง ซึ่งผู้ใช้ก็จะต้องไปศึกษาหาคำตอบต่อไปอีกว่าการที่จะเขียนโปรแกรมเพื่อสั่งให้อุปกรณ์ส่งค่ารหัส "ATD0811234567;" ออกไปทางพอร์ตสื่อสารอนุกรมนั้นต้องทำอย่างไรบ้าง ซึ่งในที่นี้จะขอแนะนำให้ใช้โปรแกรม Hyper Terminal ของ Windows เป็นเครื่องมือในการทดลองในเบื้องต้นไปก่อน โดย Hyper Terminal เป็นโปรแกรม Terminal สำเร็จรูป ซึ่งแถมมาพร้อมกับระบบปฏิบัติการ Windows อยู่แล้ว โดยความสามารถของโปรแกรมตัวนี้จะมีอยู่มากมายหลายส่วน ซึ่งในที่นี้เราจะใช้ประโยชน์เฉพาะในส่วนของการทำหน้าที่เป็น Serial Terminal ใน Text Mode เท่านั้น โดยหลังจากสั่ง Run โปรแกรมแล้ว ข้อมูลใดๆที่รับได้จากสัญญาณด้านรับ (RXD) ของพอร์ตสื่อสารอนุกรม ในย่านที่เป็นรหัส ASCII Code (20H..FFH) จะถูกนำมาแปลงเป็นตัวอักษรและแสดงผลที่หน้าจอของโปรแกรมให้เห็นทันที ส่วนรหัสของข้อมูลที่มีค่าต่ำกว่า 20H (00H-1FH) จะไม่ถูกนำมาแสดงผล แต่จะถือว่าเป็นคำสั่ง เช่น เมื่อได้รับ รหัส 0DH โปรแกรม Hyper Terminal จะถือว่าเป็นคำสั่งให้เลื่อน Cursor ของการแสดงผลไว้ในตำแหน่งเริ่มต้นของบรรทัด หรือเมื่อได้รับรหัส 0AH ก็จะทำการเลื่อน Cursor ของการแสดงผลให้ขึ้นบรรทัดใหม่แทนดังนี้เป็นต้น และ ในทางตรงกันข้าม เมื่อเราทำการกดคีย์ใดๆ โปรแกรมก็จะแปลค่าการกดคีย์นั้นให้เป็นรหัส ASCII ของตัวอักษรของตำแหน่งคีย์นั้นๆส่งออกไปยังขา TXD ของพอร์ตสื่อสารอนุกรมโดยอัตโนมัติ โดยการใช้งานโปรแกรม สามารถทำได้ดังนี้

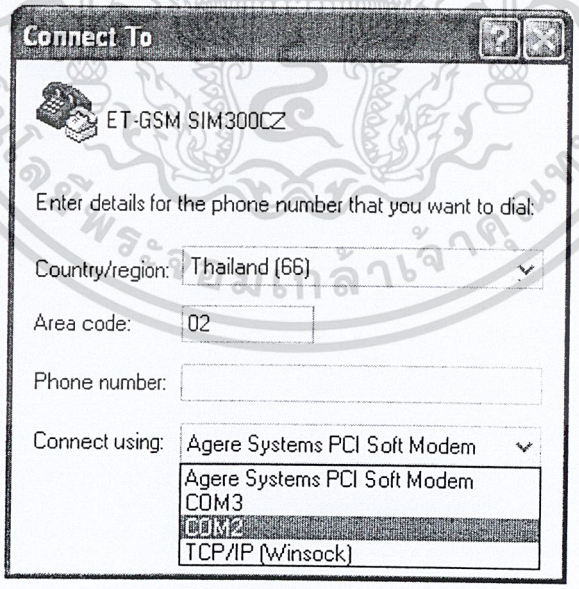
Start → Programs → Accessories → Communication → Hyper Terminal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขั้นตอนแรกจะปรากฏหน้าต่าง Connection Description ขึ้นมา ให้คลิกเมาส์เลือกรูปแบบของ ICON และกำหนดชื่อของการเชื่อมต่อตามต้องการ แล้วเลือก “OK” ดังตัวอย่าง



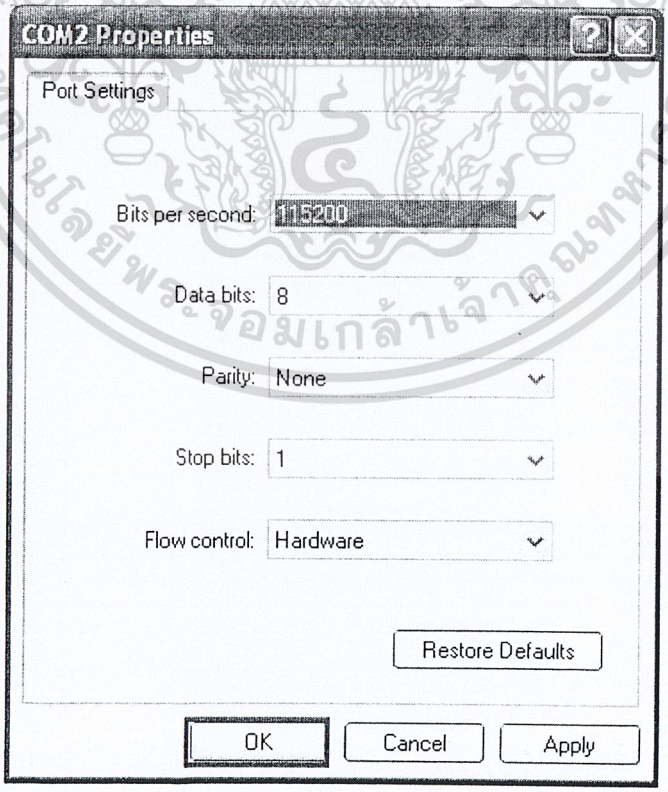
ในขั้นตอนนี้ให้คลิกเมาส์ที่ Connect using แล้วเลื่อนเมาส์ไปยังหมายเลข Comport ที่ใช้ในการเชื่อมต่อกับบอร์ดตามจริง แล้วเลือก “OK” ดังตัวอย่าง



↓

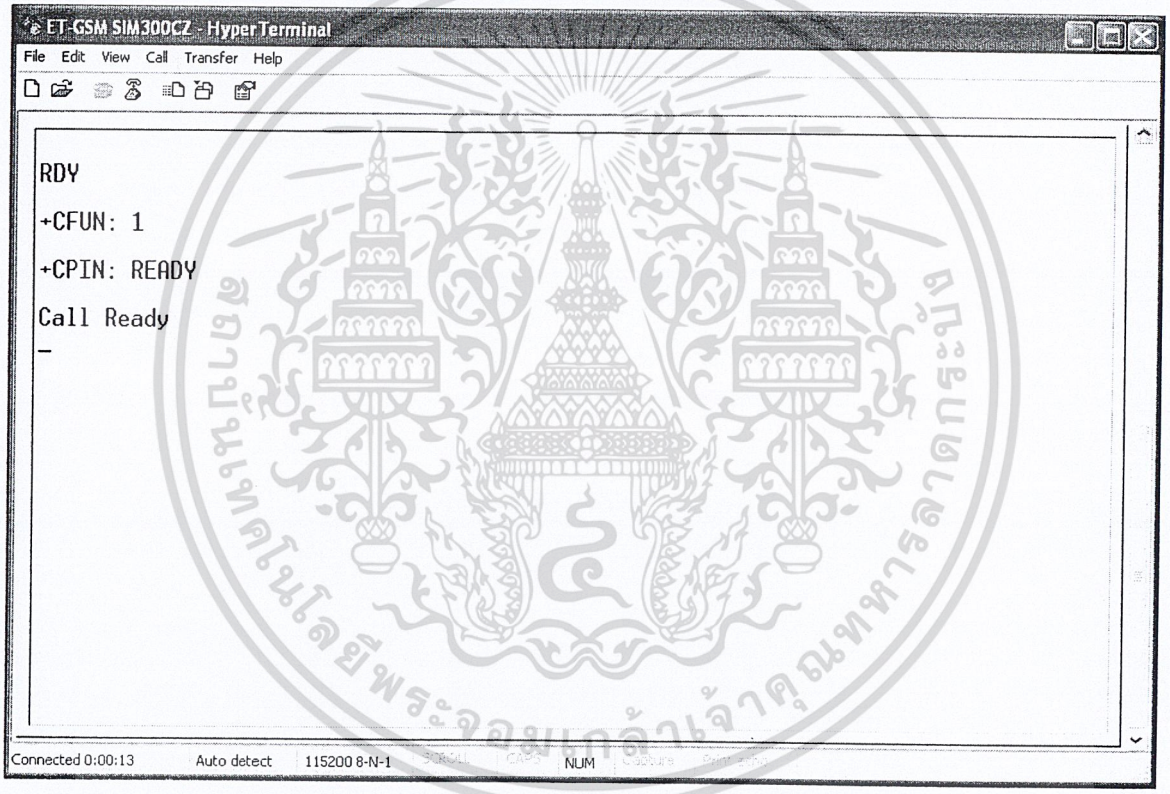
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขั้นตอนนี้ให้เลือกค่า Baud rate ให้ตรงและสอดคล้องกับที่กำหนดให้กับโมดูลไว้ หรือในกรณีที่กำหนดค่า Baud rate ของโมดูลเป็น Auto-Baud rate ไว้ก็สามารถกำหนดค่าใดๆ ที่โมดูลสามารถรองรับได้ระหว่าง 300,1200,2400,4800,9600,14400,19200,28800,38400,57600,115200 ส่วน Data ให้เลือกเป็น 8 Bit ,Parity =None, Stop bits=1, Flow Control = Hardware แล้วเลือก "OK" ดังตัวอย่าง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งหลังจากกำหนดการเชื่อมต่อต่างๆเรียบร้อยแล้ว ถ้าทุกอย่างถูกต้องให้ทดลองทำการต่อสายสัญญาณ RS232 ระหว่างบอร์ดกับ Comport ของคอมพิวเตอร์ PC แล้วจ่ายไฟเลี้ยงวงจรให้กับบอร์ด ซึ่งถ้าทุกอย่างถูกต้องจะเห็น LED VBAT สีแดงบนบอร์ดติดสว่างให้เห็น จากนั้นให้สั่ง Power-ON ตัวโมดูล โดยการกดสวิตช์ ON/OFF ค้างไว้ประมาณ 2 วินาที จะสังเกตุเห็น LED POWER และ STATUS ติดสว่างขึ้น จากนั้น LED NETLIGHT ก็จะเริ่มกระพริบเป็นจังหวะตลอดเวลา แสดงว่าโมดูลเริ่มต้นทำงานแล้ว ส่วนที่หน้าจอของ Hyper Terminal จะปรากฏข้อความการทำงานให้เห็น ให้รอจนพบคำว่า "Call Ready" ซึ่งหมายถึงโมดูลทำการค้นหาและเครือข่ายได้เรียบร้อยแล้ว จากนั้นก็จะสามารถสั่งงานโมดูลด้วยคำสั่งต่างๆ ได้ตามต้องการดังตัวอย่าง



รูปแสดงลักษณะของหน้าจอ Hyper Terminal เมื่อโมดูลพร้อมทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการใช้งานคำสั่ง

สมมุติว่าเราต้องการใช้คำสั่งสำหรับกำหนดค่าอัตราความเร็วของการสื่อสารของโมดูล ซึ่งจะต้องใช้คำสั่ง "AT+IPR" โดยเราสามารถสั่งงานคำสั่งนี้ได้หลายรูปแบบดังตัวอย่างต่อไปนี้ คือ

ถ้าเราจำไม่ได้ว่าค่าพารามิเตอร์ของคำสั่งหรืออัตรา Baudrate ที่สามารถกำหนดได้ มีค่าอะไรบ้างและกำหนดอย่างไร เราก็สามารถใช้รูปแบบการทดสอบคำสั่ง โดยใช้คำสั่ง "AT+IPR=?" และจบด้วย Enter เพื่อสอบถามได้ โดยโมดูลจะตอบรับด้วย "+IPR:" พร้อมกับพิมพ์ค่าพารามิเตอร์ของคำสั่งที่มีอยู่ทั้งหมดให้ทราบ คือ 0,300,1200,...,115200 ดังตัวอย่าง

```
AT+IPR=?<Ent>
+IPR: (),(0,300,1200,2400,4800,9600,14400,19200,28800,38400,57600,115200)
OK
```

ถ้าต้องการทราบว่าในขณะนี้ ค่า Baudrate ที่กำหนดไว้แล้ว มีค่าเป็นเช่นไร ก็สามารถใช้รูปแบบคำสั่งสำหรับสั่งอ่านค่าพารามิเตอร์ของคำสั่งนี้ คือ "AT+IPR?" ซึ่งโมดูลจะตอบรับด้วย "+IPR:" ตามด้วยค่าพารามิเตอร์ที่กำหนดไว้แล้วให้ทราบดังตัวอย่าง (0=Auto Baudrate)

```
AT+IPR?<Ent>
+IPR: 0
OK
```

จากตัวอย่างข้างต้น จะเห็นได้ว่าค่าพารามิเตอร์ของคำสั่ง IPR ที่กำหนดไว้แล้วคือ 0 ซึ่งหมายถึง Auto-Baudrate โดยโมดูลจะทำการปรับค่าความเร็ว Baudrate โดยอัตโนมัติ ในตอนเริ่มต้นการทำงานครั้งแรก แต่ถ้าเราต้องการกำหนดค่า Baudrate เป็นค่าคงที่ไปเลย เพื่อให้โมดูลใช้อัตราความเร็วนี้ตลอด ก็สามารถใช้รูปแบบคำสั่ง กำหนดค่าการทำงาน ได้ เช่น ถ้าต้องการกำหนด Baudrate เป็น 115200 ก็จะใช้รูปแบบคำสั่งเป็น "AT+IPR=115200" ดังตัวอย่าง

```
AT+IPR=115200<Ent>
OK
```

ซึ่งหลังจากสั่งเปลี่ยนค่า Baudrate เป็น 115200 แล้ว ต่อจากนี้ไปก็สามารถสื่อสารกับโมดูลด้วยค่าความเร็วนี้ได้ตลอด แต่ถ้าสั่ง รีเซ็ตโมดูลใหม่ ค่า Baudrate ก็จะกลับไปเป็นแบบ Auto-Baudrate อีก เนื่องจากเรายังไม่ได้สั่งบันทึกค่าให้เป็น Configuration ถาวรไว้ ซึ่งสามารถทำได้โดยใช้คำสั่ง "AT&W"

```
AT&W<Ent>
OK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนด Flow Control

โมดูล SIM300CZ สามารถกำหนด Flow Control หรือ รูปแบบการตรวจสอบความพร้อมในการสื่อสารและรับส่งข้อมูลได้ด้วย ซึ่ง Flow Control จะมีความสำคัญเป็นอย่างมาก เนื่องจากการประมวลผลของอุปกรณ์ต่างๆจะมีความช้าเร็วที่แตกต่างกัน เมื่อมีการรับส่งข้อมูลที่มีจำนวนข้อมูลมากๆแบบต่อเนื่องนั้น ถ้าฝ่ายรับไม่พร้อมรับข้อมูลแต่ฝ่ายส่งยังคงส่งข้อมูลออกไป ก็จะทำให้ข้อมูลสูญหายและเกิดความผิดพลาดขึ้นได้ โดย SIM300CZ เองรองรับการตรวจสอบความพร้อมหรือ Flow Control ได้ 2 แบบ คือ

- Software Flow Control (XON/XOFF Flow Control) เป็นการตรวจสอบความพร้อมด้วย Software โดยจะใช้รหัส XOF(13H) เป็นตัวสั่งหยุดการส่งข้อมูลจากฝ่ายส่ง และใช้รหัส XON(11H) เพื่อบอกหรืออนุญาตให้ฝ่ายส่งเริ่มต้นส่งข้อมูลลำดับต่อไปมายังโมดูลได้ โดยการใช้ Flow Control แบบนี้เหมาะกับการเชื่อมต่อกับอุปกรณ์ที่ไม่มีสัญญาณตรวจสอบความพร้อม เช่น ไมโครคอนโทรลเลอร์หรืออุปกรณ์ที่ใช้การต่อสายสัญญาณเพียง 3 เส้น (RXD,TXD และ GND)
- Hardware Flow Control (RTS/CTS Flow Control) เป็นการตรวจสอบความพร้อมด้วยสัญญาณทางฮาร์ดแวร์ โดยใช้การ Active("LOW") สัญญาณ CTS เพื่อบอกให้ฝ่ายส่งหยุดการส่งข้อมูลเมื่อโมดูลไม่พร้อมรับข้อมูล และในทางกลับกันก่อนการส่งข้อมูลกลับออกไปมันจะตรวจสอบสถานะของ RTS ว่า Active อยู่หรือไม่ ถ้า Active("LOW") แสดงว่าฝ่ายรับยังไม่พร้อมรับมันจะหยุดรอจนกว่า RTS จะเป็น "HIGH"

การกำหนด Flow Control เป็น XON/XOFF จะใช้คำสั่ง "AT+IFC=1,1"
การกำหนด Flow Control เป็น RTS/CTS จะใช้คำสั่ง "AT+IFC=2,2"

การกำหนด Format ข้อมูลของ RS232

เราสามารถกำหนด Format ของข้อมูล ได้ว่าจะใช้รูปแบบการส่งข้อมูลเป็นอย่างไร ใช้ขนาดข้อมูลเป็นกี่บิต ใช้บิตตรวจสอบความผิดพลาด Parity หรือไม่ และต้องการใช้ Stop Bit เป็นกี่บิต ซึ่งตามปรกติทั่วไปแล้วจะใช้ Data 8 Bit ,None Parity,1 Stop Bit ซึ่งสามารถเปลี่ยนแปลงได้โดยใช้คำสั่ง "AT+ICF"

การกำหนด Format ข้อมูลเป็น 8 Bit Data ,None Parity ,1 Stop Bit จะใช้คำสั่ง "AT+ICF=3,3"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การ Setup และตรวจสอบค่า Configuration

ตามปรกติแล้วการทำงานของโมดูล SIM300CZ นั้นจะสามารถกำหนดรูปแบบการทำงานได้มากมายหลายลักษณะ เช่น เงื่อนไขในการติดต่อสื่อสารกับโมดูล ผู้ใช้สามารถเปลี่ยนแปลงค่าต่างๆได้มากมาย ไม่ว่าจะเป็นค่า Baud rate หรือรูปแบบของการ Handshake ต่างๆ ที่จะใช้ในการสื่อสาร เป็นต้น ดังนั้นจึงจำเป็นต้องมีการกำหนดรูปแบบการทำงานของโมดูลให้ตรงกับความต้องการ ซึ่งตามปรกติแล้วเงื่อนไขต่างๆเหล่านี้จะมีค่าที่แน่นอนอยู่ค่าหนึ่งเสมอหลังการรีเซ็ต หรือ Power ON โดยโมดูลจะกำหนดค่าเงื่อนไขต่างๆให้กับตัวมันเองในตอนเริ่มต้นการทำงานด้วยค่าที่กำหนดไว้ใน Configuration ที่ถูกบันทึกไว้แล้ว แต่อย่างไรก็ตามผู้ใช้สามารถสั่งเปลี่ยนแปลงแก้ไขค่า Configuration ต่างๆได้เองตามต้องการ ซึ่งวิธีการกำหนดเงื่อนไขการทำงานให้กับโมดูลนั้นสามารถทำได้ 2 แบบ

- **การกำหนดค่าแบบถาวร** จะเป็นการสั่งบันทึกค่าเงื่อนไขการทำงานต่างๆของโมดูลตามรูปแบบที่เรากำหนดไว้ในหน่วยความจำถาวรภายในตัวโมดูล โดยใช้คำสั่ง **"AT&W"** ซึ่งหลังจากโมดูลเริ่มต้นทำงานใหม่ หรือ หลังการรีเซ็ตโมดูลแต่ละครั้ง ค่าการทำงานต่างๆของโมดูลจะถูกกำหนดเงื่อนไขตามที่เรากำหนดไว้แล้วเสมอ
- **การกำหนดค่าแบบชั่วคราว** เป็นการใช้คำสั่ง AT Command ต่างๆ เพื่อกำหนดเงื่อนไขการทำงาน ให้กับโมดูล แต่ไม่มีการสั่งบันทึกค่า Configuration ด้วยคำสั่ง **"AT&W"** ซึ่งการทำงานของโมดูลก็จะปรับเปลี่ยนไปตามการสั่งงานในขณะนั้นๆ แต่เมื่อสั่งรีเซ็ตการทำงานของโมดูล หรือ มีการ Power ON ใหม่คุณสมบัติการทำงานของโมดูลจะถูกเปลี่ยนกลับเป็นค่าเดิมอีก

โดยเราสามารถใช้คำสั่ง AT Command ในการสั่ง ตรวจสอบ และ บันทึกค่า Configuration ต่างๆ ให้กับโมดูล SIM300CZ ได้ดังนี้

- ใช้คำสั่ง **"AT&V"** เพื่อสั่งให้โมดูลแสดงค่า Configuration ปัจจุบันให้ทราบ
- ใช้คำสั่ง **"AT&F"** เพื่อสั่งกำหนดค่า Configuration ทั้งหมดให้กลับเป็นค่ามาตรฐาน
- ใช้คำสั่ง **"AT&W"** เพื่อสั่งบันทึกค่า Configuration ด้วยค่าที่เรากำหนดไว้ในขณะนั้นๆ

ค่า Configuration ที่แนะนำ

- AT+CMGF=1(SMS Message = Text Mode)
- ATE=1 (Echo Mode ON)
- AT+CSCLK=0(Disable Sleep Mode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบคุณภาพสัญญาณ

การตรวจสอบคุณภาพสัญญาณ จะใช้คำสั่ง "AT+CSQ" โดยคำสั่งนี้ใช้ตรวจสอบระดับความแรงของสัญญาณ โดยโมดูลจะตอบรับเป็นค่าตัวเลข 0..31 หรือ 00 โดยถ้าค่าระหว่าง 2..30 อยู่ในเกณฑ์ดี ถ้าค่า 31 ถือว่าระดับสัญญาณดีมาก ส่วน 99 หมายถึงตรวจสอบไม่ได้ รูปแบบดังตัวอย่าง

```
AT+CSQ<Ent>
+CSQ: 15,0
OK
```

การตรวจสอบ รหัสผลิตภัณฑ์

```
ATI<Ent>
SIMCOM_Ltd
SIMCOM_SIM300C
Revision:1008B10SIM300C32_SPANSION
OK
```

การตรวจสอบ รหัสผู้ผลิต

```
AT+GMI<Ent>
SIMCOM_Ltd
OK
```

การตรวจสอบ รหัสรุ่น

```
AT+GMM<Ent>
SIMCOM_SIM300C
OK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบ รหัส Version Firmware

```
AT+GMR<Ent>
Revision:1008B10SIM300C32_SPANSION
OK
```

การตรวจสอบ Serial Number ของโมดูล

```
AT+GSN<Ent>
359853000524859
OK
```

การตรวจสอบ รหัสเครือข่าย SIM ผู้ให้บริการ

```
AT+COPS?<Ent>
+COPS: 0,0,"AIS GSM"
OK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การโทรออก การรับสาย และ การยกเลิกการโทร

- ใช้คำสั่ง **"ATD"** เพื่อสั่งโทรออก โดยรูปแบบการใช้คำสั่งให้ตามด้วยเบอร์ปลายทาง
- ใช้คำสั่ง **"ATDL"** เพื่อสั่งโทรออกด้วยหมายเลขโทรออกครั้งสุดท้าย
- ใช้คำสั่ง **"ATA"** เพื่อรับสายเรียกเข้า โดยเมื่อมีสายเรียกเข้าจะมีเสียงเรียกเข้าที่ Buzzer ให้เราทราบ ถ้าต้องการรับสายให้ใช้คำสั่ง "ATA" เพื่อรับสายได้ทันที ซึ่งหลังจากสั่งรับสายแล้วผู้ใช้จะสามารถพูดคุยกับปลายสายได้ทันที โดยใช้ Handset หรือชุด ปากพูดหูฟังของโทรศัพท์บ้าน
- ใช้คำสั่ง **"ATH"** เพื่อสั่งวางสาย หรือยกเลิกการโทรออก

ตัวอย่างการโทรออก ซึ่งเป็นการสื่อสารด้วย Voice จะต้องปิดท้ายคำสั่งด้วยเครื่องหมายเซมิโคลอน (;) และจบคำสั่งด้วย Enter เช่นถ้าต้องการโทรออกไปยังเบอร์ 0811234567 จะเป็นดังนี้

```
ATD0811234567;<Ent>
OK
```

ในกรณีที่สั่งโทรออกแล้วไม่มีการรับสาย หรือ สายไม่ว่างโมดูลจะรายงานผลด้วยข้อความ "BUSY" ดังตัวอย่าง

```
ATD0812505187;<Ent>
OK
BUSY
```

ตัวอย่างการสั่งตรวจสอบยอดเงินของระบบ 1-2-CALL ซึ่งถ้าเป็นโทรศัพท์มือถือปรกติจะใช้วิธีการพิมพ์เครื่องหมาย "*121#" แล้วสั่งโทรออก แต่สำหรับโมดูล SIM300CZ จะต้องใช้คำสั่ง "ATD" สำหรับสั่งโทรออกแล้วตามด้วยเครื่องหมายแทน ดังตัวอย่าง

```
ATD*121#<Ent>
+CUSD: 0,"You have 67.00 baht, valid until 29/04/08",15
OK
```

ตัวอย่างการรับสายเรียกเข้า เมื่อมีสายเรียกเข้าโมดูล SIM300CZ จะมีข้อความ "RING" และสร้างเสียงเรียกเข้าเป็นจังหวะที่ Buzzer ให้ทราบ ถ้าผู้ใช้ต้องการรับสาย ให้ใช้คำสั่ง "ATA" เพื่อสั่งรับสาย หรือใช้คำสั่ง "ATH" เพื่อวางหูหรือยกเลิกไม่รับสาย ดังตัวอย่าง

```
RING
ATA<Ent>
OK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการรับข้อความ SMS

ตามปรกติแล้วโมดูล SIM300CZ จะสามารถกำหนดโหมดการทำงานของข้อความหรือ SMS ได้ 2 โหมด คือ PDU Mode และ Text Mode โดย PDU Mode การรับและแสดงผลการทำงานของคำสั่งจะเป็นรูปแบบของรหัสตัวเลขแบบ Binary Code ส่วน Text Mode การรับและแสดงผลการทำงานของคำสั่งจะเป็น ข้อความ ซึ่งจะง่ายต่อการแปลความหมายและทำความเข้าใจมากกว่า PDU Mode ซึ่งในการทดสอบจะขอแสดงให้เห็นด้วย Text Mode

- ใช้คำสั่ง "AT+CMGF=1" เพื่อกำหนดรูปแบบของข้อความเป็น Text Mode ซึ่งเมื่อมีการส่งข้อความ SMS มายังโมดูล จะมีข้อความแจ้งให้ทราบ เช่น +CMTI: "SM",3 ซึ่งหมายความว่า มีข้อความส่งเข้าและเก็บไว้ในหน่วยความจำลำดับที่ 3
- ใช้คำสั่ง "AT+CMGR" เพื่อสั่งอ่านข้อความ เช่นถ้าต้องการอ่านข้อความลำดับที่3 ก็ให้ใช้คำสั่งเป็น "AT+CMGR=3"
- ใช้คำสั่ง "AT+CMGL" เพื่อสั่งแสดงข้อความทั้งหมดที่เก็บไว้ในหน่วยความจำ โดยสามารถเลือกประเภทของข้อความได้ เช่น ข้อความใหม่ ข้อความทั้งหมด
- ใช้คำสั่ง "AT+CMGD" เพื่อสั่งลบข้อความออกจากหน่วยความจำ เช่น ถ้าต้องการสั่งลบข้อความลำดับที่3 ก็ให้ใช้คำสั่งเป็น "AT+CMGD=3"

ตัวอย่างการรับข้อความ SMS ในตัวอย่างจะทดสอบด้วยการส่งข้อความ "Hello 12345" ไปให้กับโมดูล SIM300CZ ซึ่งเมื่อรับข้อความได้จะมีข้อความ +CMTI: "SM",n โดย n หมายถึงลำดับที่ของข้อความ

```
+CMTI: "SM",3
AT+CMGR=3<Ent>
+CMGR: "REC UNREAD","+66812505187",,"07/11/19,13:29:25+28"
Hello 12345
OK
```

ถ้ามีการสั่งอ่านข้อความเดิมซ้ำใหม่สถานะของข้อความจะเปลี่ยนเป็น "REC READ" แทน เพื่อแสดงให้ทราบว่าข้อความนี้ถูกอ่านไปแล้วดังตัวอย่าง

```
AT+CMGR=3<Ent>
+CMGR: "REC READ","+66812505187",,"07/11/19,13:29:25+28"
Hello 12345
OK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการส่งข้อความ SMS

ในการส่งข้อความ SMS นั้นจะใช้คำสั่ง “AT+CMGS” ในการสั่งงาน โดยในกรณีที่ใช้ Text Mode นั้นให้ใช้รูปแบบคำสั่งเป็น “AT+CMGS=”+เบอร์ผู้รับ” โดยเบอร์ของผู้รับต้องใส่รหัสประเทศนำหน้าแทนศูนย์ด้วยเสมอ ซึ่งในกรณีที่เป็นประเทศไทยจะใช้รหัสประเทศเป็น “66” ดังนั้นถ้าต้องการส่งข้อความ SMS ให้กับเบอร์ที่ใช้งานอยู่ในประเทศไทย เช่น 081-1234567 ก็จะต้องกำหนดหมายเลขของเบอร์ผู้รับปลายทางเป็น 6681-1234567 แทน ซึ่งในกรณีนี้จะได้รหัสเบอร์ผู้รับข้อความเป็น “+66811234567” ซึ่งเมื่อโมดูล SIM300CZ ได้รับคำสั่ง “AT+CMGS” เรียบร้อยแล้วมันจะตอบรับด้วยการส่งเครื่องหมาย “>” กลับมาบอก ซึ่งหลังจากนี้เป็นต้นไปผู้ใช้ก็สามารถจะทำการพิมพ์ข้อความต่างๆที่ต้องการจะส่งให้กับโมดูลได้ทันที โดยให้ปิดท้ายข้อความด้วยรหัส “Ctrl+Z” ตามด้วย “Enter” เช่นถ้าต้องการส่งข้อความ SMS ให้กับหมายเลข 0811234567 ด้วยข้อความ “Hello Test SMS” จะเป็นดังนี้

```
AT+CMGS="+66811234567"<Ent>
> Hello Test SMS<Ctrl+Z><Ent>
+CMGS: 6
OK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รหัสข้อความ SMS ภาษาไทย

สำหรับข้อความ SMS ที่เป็นภาษาไทยนั้น จะไม่สามารถแสดงผลด้วยโปรแกรม Terminal ปรกติได้ ทั้งนี้ก็เนื่องมาจากว่าระบบตัวอักษรที่ใช้ในโปรแกรม Terminal นั้นจะใช้รหัส ASCII ปรกติที่มีขนาดเพียง 1 ไบท์ แต่สำหรับรหัสภาษาไทยที่ใช้ในระบบสื่อสารของโทรศัพท์มือถือต่างๆนั้น จะใช้รหัสพิเศษเฉพาะที่เรียกว่า "Unicode" ซึ่งตัวอักษร 1 ตัวจะประกอบไปด้วยข้อมูลจำนวน 2 ไบท์ โดยรหัส Unicode ของภาษาไทยนั้นจะมีค่าอยู่ระหว่าง 0E00H...0E7FH สำหรับภาษาอังกฤษนั้นถ้าเป็น Unicode จะใช้รหัสตัวอักษรขนาด 2 Byte เช่นเดียวกันกับภาษาไทย โดยจะมีค่าอยู่ระหว่าง 0000H..007FH โดยตามปรกติแล้วถ้าข้อความเป็นภาษาอังกฤษอย่างเดียวรหัสของตัวอักษรที่ใช้ใน SMS จะเป็นแบบ ASCII คือ ใช้รหัส ขนาด 1 ไบท์ โดยตัดรหัส 00H ไบท์แรกใน Unicode ทิ้งไป เช่น A แทนที่จะเป็นรหัส 0041H ก็จะเหลือเพียง 41H เป็นต้น

	┌	┐	└	┘	│	─	●	◘						♫	☼
0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	000B	000C	000D	000E	000F
┼	◄	↕	!!	¶	┴	┬	┤	↑	├	→	←				
0010	0011	0012	0013	0014	0015	0016	0017	0018	0019	001A	001B	001C	001D	001E	001F
	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	002A	002B	002C	002D	002E	002F
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
0030	0031	0032	0033	0034	0035	0036	0037	0038	0039	003A	003B	003C	003D	003E	003F
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
0040	0041	0042	0043	0044	0045	0046	0047	0048	0049	004A	004B	004C	004D	004E	004F
P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
0050	0051	0052	0053	0054	0055	0056	0057	0058	0059	005A	005B	005C	005D	005E	005F
`	a	B	c	d	e	f	g	h	i	j	k	l	m	n	o
0060	0061	0062	0063	0064	0065	0066	0067	0068	0069	006A	006B	006C	006D	006E	006F
p	q	R	s	t	u	v	w	x	y	z	{	|	}	~	
0070	0071	0072	0073	0074	0075	0076	0077	0078	0079	007A	007B	007C	007D	007E	007F

ตาราง แสดงรหัส Unicode ภาษาอังกฤษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F
`	a	B	c	d	e	f	g	h	i	j	k	l	m	n	o
60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F
p	q	R	s	t	u	v	w	x	y	z	{	|	}	~	
70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F

ตาราง แสดงรหัส ASCII ภาษาอังกฤษ

	ก	ข	ฃ	ค	ฅ	ฆ	ง	จ	ฉ	ช	ซ	ฌ	ญ	ฎ	ฏ
0E00	0E01	0E02	0E03	0E04	0E05	0E06	0E07	0E08	0E09	0E0A	0E0B	0E0C	0E0D	0E0E	0E0F
ฐ	ฑ	ฒ	ณ	ด	ต	ถ	ท	ธ	น	บ	ป	ผ	ฝ	พ	ฟ
0E10	0E11	0E12	0E13	0E14	0E15	0E16	0E17	0E18	0E19	0E1A	0E1B	0E1C	0E1D	0E1E	0E1F
ภ	ม	ย	ร	ฤ	ล	ฦ	ว	ศ	ษ	ส	ห	ฬ	อ	ฮ	ฯ
0E20	0E21	0E22	0E23	0E24	0E25	0E26	0E27	0E28	0E29	0E2A	0E2B	0E2C	0E2D	0E2E	0E2F
ะ	ั	า	ำ	ิ	ี	ึ	ื	ุ	ู	ฺ					฿
0E30	0E31	0E32	0E33	0E34	0E35	0E36	0E37	0E38	0E39	0E3A	0E3B	0E3C	0E3D	0E3E	0E3F
เ	แ	โ	ใ	ไ	ๅ	ๆ	็	่	้	๊	๋	์	๎	๏	๏
0E40	0E41	0E42	0E43	0E44	0E45	0E46	0E47	0E48	0E49	0E4A	0E4B	0E4C	0E4D	0E4E	0E4F
๐	๑	๒	๓	๔	๕	๖	๗	๘	๙	๚	๛				
0E50	0E51	0E52	0E53	0E54	0E55	0E56	0E57	0E58	0E59	0E5A	0E5B	0E5C	0E5D	0E5E	0E5F
0E60	0E61	0E62	0E63	0E64	0E65	0E66	0E67	0E68	0E69	0E6A	0E6B	0E6C	0E6D	0E6E	0E6F
0E70	0E71	0E72	0E73	0E74	0E75	0E76	0E77	0E78	0E79	0E7A	0E7B	0E7C	0E7D	0E7E	0E7F

ตาราง แสดงรหัส Unicode ภาษาไทย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการถอดรหัสตัวอักษร Unicode

สำหรับรหัสตัวอักษรที่เป็น Unicode นั้น จะเห็นได้ว่าแต่ละตัวอักษรจะประกอบไปด้วยรหัส Code จำนวน 2 ไบท์เสมอ โดยตัวแรกเป็นตัวบอกรหัส Table ว่าเป็น Unicode ของภาษาใด โดยถ้าเป็นรหัส Unicode ของภาษาอังกฤษ ไบท์แรกจะมีค่าเป็น 00H ส่วนไบท์ที่2 จะเป็นรหัสตัวอักษร ซึ่งมีค่าตรงกันกับรหัส ASCII ส่วนภาษาไทยนั้น ไบท์แรกจะมีค่ารหัสเป็น 0EH ส่วนไบท์ที่ 2 จะเป็นรหัสตัวอักษร ซึ่งจากการทดสอบรับข้อความรหัสตัวอักษรจาก SMS พบว่า ถ้าใช้ภาษาอังกฤษอย่างเดียว รหัสของตัวอักษรจะเป็นแบบรหัส ASCII คือ 1 ตัวอักษร จะมีรหัส 1 ไบท์ แต่เมื่อมีการใช้ข้อความที่มีทั้งภาษาไทยและภาษาอังกฤษรวมกันพบว่ามีการเข้ารหัสตัวอักษรภาษาอังกฤษเป็นแบบ Unicode ด้วย

ดังนั้นจึงน่าจะพอสรุปได้ว่า ถ้าใช้ข้อความที่เป็นภาษาไทย ในระบบ SMS จะใช้รหัสตัวอักษรที่เป็นแบบ Unicode เสมอ แต่สำหรับภาษาอังกฤษนั้น ในระบบโทรศัพท์จะสามารถเลือกใช้ได้ทั้งระบบ Unicode และ ASCII Code โดยถ้าเป็น Unicode จะใช้รหัสตัวอักษรขนาด 2 Byte เช่นเดียวกันกับภาษาไทย โดยจะมีค่าอยู่ระหว่าง 0000H..007FH โดยมีรหัส 00H เป็นข้อมูลไบท์แรก ซึ่งถ้าข้อความเป็นภาษาอังกฤษอย่างเดียวรหัสของตัวอักษรที่ใช้ใน SMS จะเป็นแบบ ASCII คือ ใช้รหัส ขนาด 1 ไบท์ โดยตัดรหัส 00H ไบท์แรกใน Unicode ทิ้งไป เช่น A แทนที่จะเป็นรหัส 0041H ก็จะเหลือเพียง 41H เป็นต้น แต่สำหรับข้อความที่มีทั้งภาษาไทยและภาษาอังกฤษรวมกันพบว่ามีการเข้ารหัส Code ตัวอักษรเป็นแบบ Unicode ด้วยเช่นเดียวกันกับภาษาไทย

ดังนั้นในการถอดรหัสตัวอักษรต้องพิจารณาถึงจุดนี้ด้วย โดยมีข้อสังเกตุว่า ถ้าพบรหัสตัวอักษรที่มีค่าระหว่าง 20H-7FH แสดงว่าเป็นรหัสแบบ ASCII สามารถนำไปแสดงผลได้เลย แต่ถ้าพบว่ารหัสเป็น 00H แสดงว่าเป็นรหัสแบบ Unicode ภาษาอังกฤษ ซึ่งรหัส Code ที่เป็นรหัสตัวอักษรจะอยู่ในรหัสข้อมูลไบท์ถัดไป และถ้าพบรหัสเป็น 0EH แสดงว่าเป็นรหัส Unicode ภาษาไทย ซึ่งรหัส Code ที่เป็นรหัสตัวอักษรจะอยู่ในข้อมูลไบท์ถัดไป เช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเช่น ถ้าเราทดลองส่งข้อความ SMS ด้วยคำว่า "สวัสดี Jack" ไปให้กับโมดูล SIM300CZ และสมมุติว่า SIM300CZ รับข้อความนั้นไว้ และจัดเก็บไว้เป็นข้อความที่1 ถ้าใช้โปรแกรม Hyper Terminal หรือ Terminal อื่นๆที่แสดงผลเป็น ASCII จะได้รับการรายงานผลดังรูป

```
+CMTI: "SM",1
```

แต่เมื่อแสดงผลของข้อมูลที่รับได้ในรูปแบบของ HEX String จะพบว่ามีข้อมูลที่รับได้มากกว่าที่มองเห็นจากหน้าจอของโปรแกรม Hyper-Terminal ทั้งนี้ก็เนื่องมาจากว่าโปรแกรม Terminal จะแสดงผลข้อมูลที่รับได้เฉพาะในส่วนของรหัส ASCII (20H..FFH) เท่านั้น ส่วนรหัสที่ต่ำกว่า 20H (00H-1FH) โปรแกรม Hyper-Terminal จะถือว่าเป็นคำสั่ง เช่น 0DH,0AH จะไม่ถูกนำมาแสดงผล แต่จะถือเป็นคำสั่งให้เลื่อน Cursor มาไว้ในตำแหน่งเริ่มต้นของบรรทัดและขึ้นบรรทัดใหม่เป็นต้น ซึ่งในที่นี้ผู้เขียนจะขอแสดงข้อมูลที่รับได้ในรูปแบบของ Hex String แทน เช่นเมื่อรับรหัส ASCII ของตัว "A" ได้จะแสดงค่าเป็น "41" แทน โดยจะแสดง HEX String ไว้ทางด้านซ้าย และ จะแสดงรหัส ASCII ไว้ทางด้านขวาเพื่อเปรียบเทียบให้เห็น เพื่อให้ผู้อ่านสามารถเข้าใจรูปแบบได้ดีขึ้น โดยจากข้อความ +CMTI: "SM",1 ที่เรามองเห็นจากหน้าจอของโปรแกรม Hyper-Terminal เมื่อนำมาแสดงให้เห็นในรูปแบบของ HEX String จะได้ผลดังนี้

```
 0D 0A                                                    ..
 2B 43 4D 54 49 3A 20 22 53 4D 22 2C 31 0D 0A      +CMTI: "SM",1..
```

จากผลข้อความ +CMTI: "SM",1 หมายถึง มีข้อความส่งเข้ามาใหม่และถูกเก็บไว้ในหน่วยความจำโดยเป็นข้อความลำดับที่1 ซึ่งเราสามารถสั่งอ่านข้อความได้โดยใช้คำสั่ง AT+CMGR=1 ดังตัวอย่าง

```
 41 54 2B 43 4D 47 52 3D 31 0D                        AT+CMGR=1.
```

โดยเมื่อได้รับคำสั่ง AT+CMGR=1 โมดูล SIM300CZ จะแสดงข้อความลำดับที่1 ให้ทราบโดยมีรูปแบบ

```
+CMGR: "REC UNREAD","+66811234567",,"07/11/22,10:21:37+28"
<...ข้อความที่รับได้...>
```

+CMGR: คือผลตอบรับการสั่งอ่านข้อความ

"REC UNREAD" คือสถานะของข้อความ โดย REC UNREAD หมายถึงข้อความที่ยังไม่เคยถูกสั่งอ่านมาก่อนแต่ถ้าเป็นข้อความที่เคยถูกสั่งอ่านมาแล้วจะมีสถานะเป็น REC READ

"+66811234567" คือหมายเลขโทรศัพท์ของผู้ส่ง ซึ่งเป็นหมายเลขในประเทศไทย คือ 0811234567

"07/11/22,10:21:37+28" คือ วันเวลาที่รับข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจากตัวอย่างการทดลองถ้าแสดงผลข้อมูลที่รับได้จากโมดูลในรูปแบบ HEX String ผลการสั่งอ่านข้อความจะได้ผลดังตัวอย่าง

```
0D 0A                                           ..
2B 43 4D 47 52 3A 20 22 52 45 43 20 55 4E 52 45 +CMGR: "REC UNRE
41 44 22 2C 22 2B 36 36 38 31 31 32 33 34 35 36 AD","+6681123456
37 22 2C 2C 22 30 37 2F 31 31 2F 32 32 2C 31 30 7",,"07/11/22,10
3A 32 31 3A 33 37 2B 32 38 22 0D 0A             :21:37+28"..
[redacted]                                      [redacted]
[redacted] 0D 0A                                [redacted]..
0D 0A                                           ..
4F 4B 0D 0A                                     OK..
```

เมื่อลองพิจารณาถึงรหัสส่วนที่เป็นข้อความจะเห็นได้ว่า รหัสของข้อความทั้งหมดจะเป็นรหัสแบบ Unicode โดยอักษรตัวแรกจะเป็น 0E 2A และตัวสุดท้ายจะเป็น 00 6B ซึ่งเมื่อถอดรหัสข้อความดูจะได้ว่า

0EH 2AH = รหัส Unicode ของตัวอักษร [redacted]

0EH 27H = รหัส Unicode ของตัวอักษร [redacted]

0EH 31H = รหัส Unicode ของตัวอักษร [redacted]

0EH 2AH = รหัส Unicode ของตัวอักษร [redacted]

0EH 14H = รหัส Unicode ของตัวอักษร [redacted]

0EH 35H = รหัส Unicode ของตัวอักษร [redacted]

00H 20H = รหัส Unicode ของตัวอักษรภาษาอังกฤษ Space

00H 4AH = รหัส Unicode ของตัวอักษรภาษาอังกฤษ [redacted]

00H 61H = รหัส Unicode ของตัวอักษรภาษาอังกฤษ [redacted]

00H 63H = รหัส Unicode ของตัวอักษรภาษาอังกฤษ [redacted]

00H 6BH = รหัส Unicode ของตัวอักษรภาษาอังกฤษ [redacted]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ถ้าเราทดลองส่งข้อความ SMS ที่เป็นภาษาอังกฤษเพียงอย่างเดียว เช่น "Hello Jack" ไปให้กับโมดูล SIM300CZ และสมมุติว่า SIM300CZ รับข้อความนั้นไว้ และจัดเก็บไว้เป็นข้อความที่2 ถ้าใช้โปรแกรม Hyper Terminal หรือ Terminal อื่นๆที่แสดงผลเป็น ASCII จะได้รับการรายงานผลดังรูป

```
+CMTI: "SM",2
```

โดยเมื่อแสดงผลด้วยรูปแบบการแสดงผลแบบ HEX String จะได้ผลดังรูป

```
0D 0A                                           ..
2B 43 4D 54 49 3A 20 22 53 4D 22 2C 32 0D 0A    +CMTI: "SM",2..
```

จากผลข้อความ +CMTI: "SM",2 หมายถึง มีข้อความส่งเข้ามาใหม่และถูกเก็บไว้ในหน่วยความจำ โดยเป็นข้อความลำดับที่2 ซึ่งเราสามารถสั่งอ่านข้อความได้โดยใช้คำสั่ง AT+CMGR=2 ดังตัวอย่าง

```
41 54 2B 43 4D 47 52 3D 32 0D                   AT+CMGR=2.
```

ซึ่งจากตัวอย่างการทดลองถ้าแสดงผลข้อมูลที่รับได้จากโมดูลในรูปแบบ HEX String ผลการสั่งอ่านข้อความจะได้ผลดังตัวอย่าง

```
0D 0A                                           ..
2B 43 4D 47 52 3A 20 22 52 45 43 20 55 4E 52 45 +CMGR: "REC UNRE
41 44 22 2C 22 2B 36 36 38 31 31 32 33 34 35 36 AD","+6681123456
37 22 2C 2C 22 30 37 2F 31 31 2F 32 32 2C 31 31 7",,"07/11/22,11
3A 33 34 3A 30 36 2B 32 38 22 0D 0A             :34:06+28"..
48 65 6C 6C 6F 20 4A 61 63 6B 0D 0A             Hello Jack..
0D 0A                                           ..
4F 4B 0D 0A                                     OK..
```

ซึ่งจะเห็นได้ว่ารหัสของข้อความใน SMS จะเป็นแบบ ASCII ปรกติ โดยแต่ละตัวอักษรจะใช้รหัสขนาด 1 Byte ดังนี้

48H = รหัส ASCII ของ H

65H = รหัส ASCII ของ e

6CH = รหัส ASCII ของ l

6FH = รหัส ASCII ของ o

20H = รหัส ASCII ของ Space

4AH = รหัส ASCII ของ J

61H = รหัส ASCII ของ a

63H = รหัส ASCII ของ c

6BH = รหัส ASCII ของ k

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการใช้ SIM Command (SIM Application Toolkit : STK)

SIM Application Tool Kit เป็นชุดคำสั่งพิเศษของโมดูล SIM300CZ ใช้สำหรับสั่งงาน SIM ผ่านทางชุดคำสั่งพิเศษใน SIM บางรุ่นที่รองรับสร้าง Application จากผู้ใช้ได้ ซึ่งตามปรกติแล้วผู้ที่จะทำการพัฒนา Application ต่างๆให้กับ SIM ได้นั้น จะต้องมีความรู้ความเข้าใจเกี่ยวกับ SIM เป็นอย่างดี โดยข้อกำหนดต่างๆสามารถศึกษาเพิ่มเติมได้จาก Technical Reference ของ "GSM11.14" ซึ่งสำหรับผู้ให้บริการ SIM ในประเทศไทยเอง ในปัจจุบันก็ได้มีการสร้าง Application บรรจุไว้ภายใน SIM บางรุ่นด้วย เช่น SIM รุ่นที่ใช้สำหรับบริการเติมเงินให้กับโทรศัพท์มือถือ (Mobile Top Up Service) ของค่ายต่างๆ ไม่ว่าจะเป็น AIS หรือ DTAC หรือ TRUE ต่างก็มีการพัฒนา Application บน SIM Card ไว้สนับสนุนและบริการแก่ผู้ใช้งานทั่วไปแล้วทั้งสิ้น

โดยในที่นี้จะไม่ขอกล่าวถึงวิธีการพัฒนา Application บน SIM แต่จะขออธิบายถึงการติดต่อและเข้าถึงคำสั่งใน Application ที่สร้างไว้แล้วใน SIM ตัวอย่างเช่น SIM ของระบบ DTAC ในส่วนที่มีการพัฒนา Application บน SIM ไว้เป็นที่เรียบร้อยแล้ว อันได้แก่ SIM สำหรับผู้ให้บริการเติมเงินมือถือผ่านมือถือ หรือ SIM Happy Online ซึ่งจะเห็นได้ว่าในส่วนของผู้ใช้งานนั้นไม่จำเป็นต้องทราบรายละเอียดของการทำงานใน Application บน SIM แต่อย่างใด ผู้ใช้งานเพียงแต่ศึกษาข้อกำหนดและวิธีการในการเลือกเมนูและป้อนข้อมูลต่างๆให้ถูกต้องเท่านั้น ตัวอย่างเช่น วิธีการสั่งเติมเงินให้กับมือถือจะมีขั้นตอนดังนี้

1. เข้าสู่เมนู Happy Refill
2. เลือกภาษาเพื่อทำการเติมเงิน
3. เลือกเมนูเติมเงิน
4. ระบุเบอร์มือถือที่ต้องการเติมเงิน
5. เลือกราคาหรือจำนวนเงินในการเติมเงิน
6. ใส่รหัสผ่าน
7. ยืนยันการเติมเงิน

ซึ่งหลังจากที่ผู้ใช้ทำรายการต่างๆครบทั้ง 7 ขั้นตอนแล้ว โปรแกรม Application ที่บรรจุไว้ภายใน SIM ก็จะสั่งงานให้โทรศัพท์ส่ง SMS ไปยัง Server ผู้ให้บริการเพื่อดำเนินการเติมเงินให้กับเบอร์ที่ระบุไว้ ซึ่งจะเห็นได้ว่าในส่วนของผู้ใช้งานเอง จะไม่มีโอกาสทราบได้เลยว่า ข้อความ SMS ที่ส่งออกไปเพื่อร้องขอการเติมเงินนั้นเป็นอย่างไร และส่งออกไปยังหมายเลขใด ข้อมูลต่างๆที่ผู้ใช้ทำการป้อนเข้าไปนั้น มีการนำไปแก้ไขดัดแปลง หรือ ผนวกรวมกับข้อมูลอื่นๆ หรือ มีการเข้ารหัสข้อมูลอย่างไรบ้าง ผู้ใช้เองจะไม่มีโอกาสได้ทราบเลย ทำให้ระบบการทำงาน ถูกปกปิดเป็นความลับ และ มีความปลอดภัย มากขึ้น เพราะไม่ต้องกังวลว่าจะมีใครพยายามส่งข้อความ SMS ปลอมเข้ามายัง Server ของผู้ให้บริการเติมเงิน เพื่อขอเติมเงินบ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการตรวจสอบยอดเงินจากกระเป๋าเงินสด True Money ของ SIM True

เราจะมาทดลองทำความเข้าใจกระบวนการทำงานของ Application ใน SIM กันสัก 1 ตัวอย่าง โดยในที่นี้จะขอยกตัวอย่างการใช้ Application ใน SIM ระบบเติมเงิน ของ True ซึ่งได้ทำการ Download Application ของ True Money ให้กับ SIM เรียบร้อยแล้ว โดยในอันดับแรกเราจะทดลองใช้โทรศัพท์มือถือในการทำรายการกันดูก่อน จากนั้นจึงจะทดลองให้โมดูล SIM300CZ สั่งทำรายการแบบเดียวกันเพื่อให้ผู้ใช้ได้เห็นการทำงานจะได้ทำความเข้าใจได้ดีขึ้น โดยในกรณีใช้โทรศัพท์มือถือทำรายการจะมีขั้นตอนดังนี้

1. เมื่อเลือกเมนูไปยัง True Money จะมีตัวเลือกรายการหัวข้อย่อยใน Menu ให้เลือก 6 ตัวเลือก คือ
   - 1 check balance
   - 2 transfer
   - 3 refill
   - 4 change password
   - 5 register
   - 6 about
2. สมมุติว่าเราต้องการตรวจสอบยอดเงิน ก็ต้องเลือกกดคีย์หมายเลข "1" เพื่อเลือกทำรายการใน Submenu ลำดับที่ 1 ซึ่งก็คือ "check balance"
3. เมื่อโปรแกรม Application ในโทรศัพท์มือถือรับรู้การกดคีย์ มันจะสั่งให้ Application ใน SIM ปฏิบัติงานตามหัวข้อที่ได้เลือกไว้ โดยในกรณีนี้จะปรากฏข้อความ password ที่หน้าจอแสดงผลของโทรศัพท์และรอรับการกดคีย์เพื่อป้อนรหัสผ่าน ในขั้นตอนนี้ผู้ใช้ต้องทำการป้อนรหัสผ่านเป็นตัวเลข 4 หลักตามที่ลงทะเบียนกับผู้ให้บริการไว้
4. เมื่อทำการป้อนรหัสผ่านเรียบร้อยแล้วโปรแกรม Application บนมือถือ ก็จะแปลรหัส Password ที่ได้รับจากผู้ใช้ส่งให้กับโปรแกรม Application ใน SIM ซึ่ง Application ใน SIM ก็จะทำการนำค่ารหัส Password ที่รับได้เข้ารหัสแปลงเป็นข้อความ SMS แล้วส่งไปยัง Server ของผู้ให้บริการ
5. ผู้ใช้จะได้รับข้อความเป็น SMS แจ้งยอดเงินคงเหลือให้ทราบดังตัวอย่าง

จากกระบวนการข้างต้นจะเห็นได้ว่าในกรณีที่ใช้ โทรศัพท์มือถือ เป็นตัวทำรายการนั้นจะทำได้ง่ายมาก เนื่องจากโทรศัพท์มือถือเองมี Application รองรับการทำงานเหล่านี้ไว้อยู่แล้ว และมีคีย์บอร์ดพร้อมหน้าจอสำหรับแสดงรายการต่างๆให้เห็นได้โดยสะดวก แต่สำหรับในกรณีที่จะใช้โมดูล SIM300CZ เป็นตัวทำรายการนั้น จะไม่สามารถทำได้โดยตรง เพราะเรายังไม่มี Application สำเร็จรูปรองรับการทำงานกับ SIM เหมือนกับโทรศัพท์มือถือ แต่สิ่งที่เราสามารถทำได้ก็คือ การใช้ชุดคำสั่งต่างๆของโมดูล SIM300CZ สำหรับติดต่อกับ SIM อันได้แก่ การเข้าถึง Application ใน SIM การขอดูรายการตัวเลือกในเมนูของ Application ใน SIM การสั่งให้โมดูลปฏิบัติงานตามหัวข้อ(เลือก Submenu) ต่างๆเหล่านี้เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

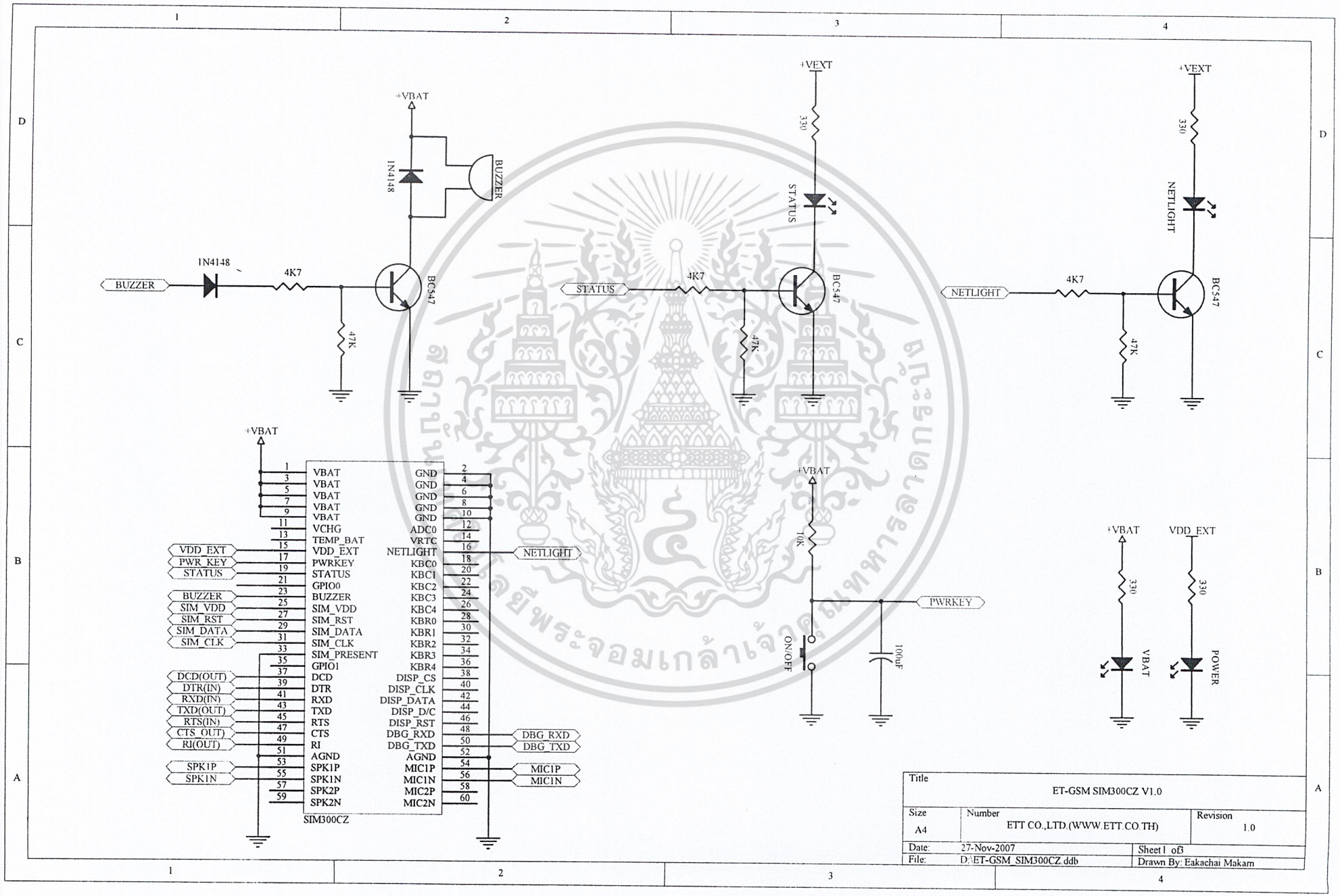
ตัวอย่าง ขั้นตอนการตรวจสอบยอดเงินของ "True Money" ด้วย SIM300CZ

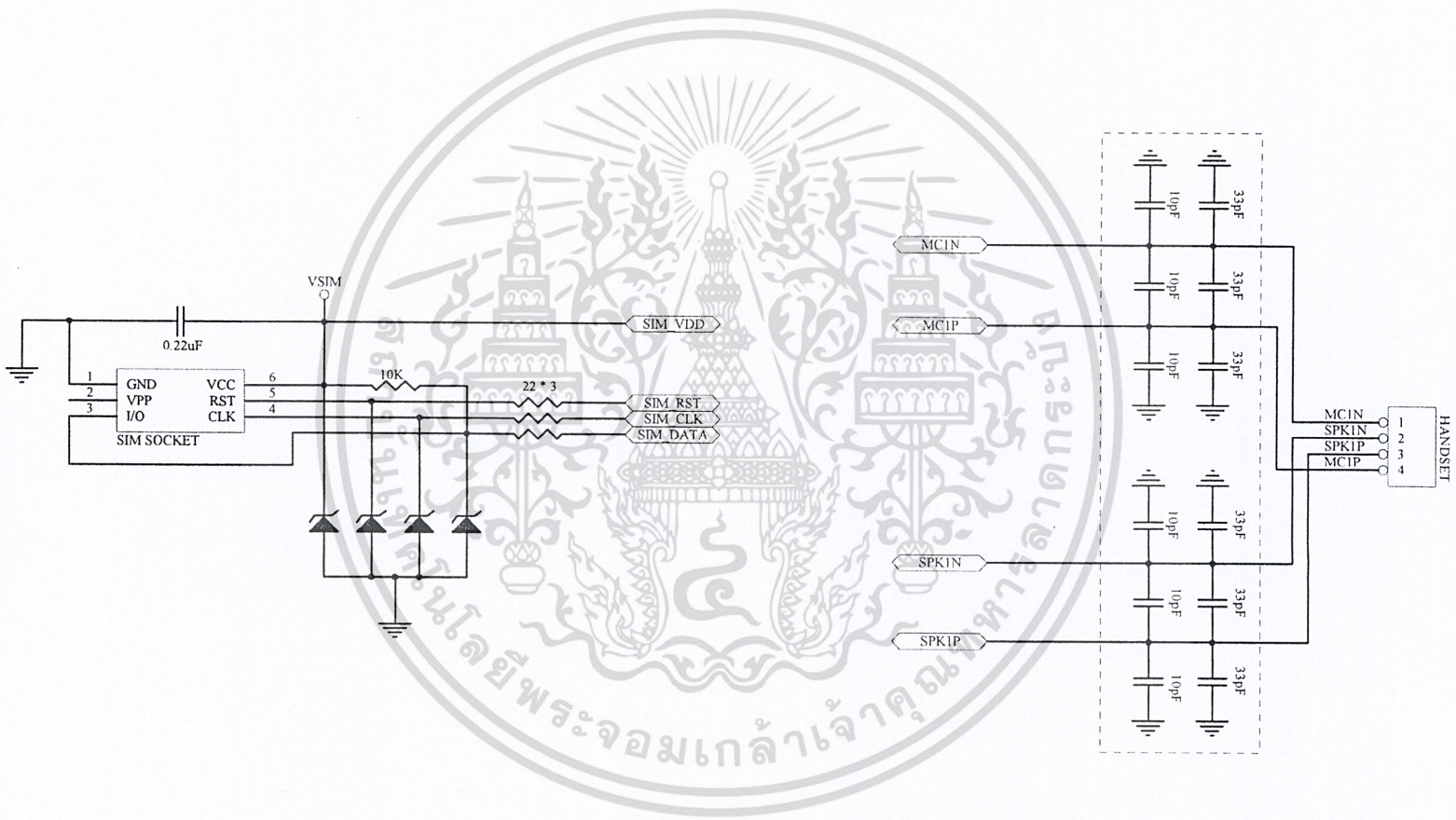
```
AT+CMGF=1<Enter>                          ;Setup SIM300C = Text Mode
OK
AT+CMEE=2<Enter>                          ;Enable result code ,use verbose values
OK
AT+STSR=240<Enter>                        ;Setup Response Time Out = 200 Second
OK
AT+STGI=0,1,0,0,0,0,0,0,0<Enter>          ;Send Setup SIM Tool Kit Command
OK
+STC: 25                                  ;Wait Setup Menu Command
AT+STGC=25<Enter>                         ;Read Parameter of Setup Menu Command
+STGC: 25,8,0,0,0,"Menu"                  ;Display SIM Main Menu
+STGC: 5,"Fun & Smart"
+STGC: 6,"Setting2Play"
+STGC: 8,"True Money"
+STGC: 9,"True Product"
+STGC: 10,"True Payment"
+STGC: 12,"True Transfer"
+STGC: 11,"Other Service"
+STGC: 13,"Mobile Top Up"
OK
ATH+STGR=25,0<Enter>                      ;Request Response of Setup Menu
OK
+STC: 81                                  ;End of proactive session
AT+STMS=8<Enter>                          ;Select Sub Menu = 8 (True Money)
OK
+STC: 24                                  ;Wait Select Item Command
AT+STGC=24<Enter>                         ;Read Parameter of Select Item Command
+STGC: 24,6,0,0                           ;Display Item
+STGC: 1,"check balance"
+STGC: 2,"transfer"
+STGC: 3,"refill"
+STGC: 4,"change password"
+STGC: 5,"register"
+STGC: 6,"about"
OK
ATH+STGR=24,0,1<Enter>                    ;Request Response of Select Item
OK
+STC: 23                                  ;Wait Get Input Command
AT+STGC=23<Enter>                         ;Read Parameter of Get Input Command
+STGC: 23,4,"password:",1,0,0,1,4         ;Display Input Choice
OK
AT+STGR=23,0,4,"1234"<Enter>                        ;Enter Password = 1234
OK
+STUD: 13                                 ;Send SMS unsolicited proactive command
+STC: 81                                  ;End of proactive session
+CMTI: "SM",5                             ;SMS Receive = Number 2
AT+CMGR=5<Enter>                          ;Read SMS Number 5
+CMGR: "REC UNREAD","True Money",,"07/11/22,15:29:45+07"
True Money A/C 0891465362 is 300.00 Baht.
OK
```

แสดง ตัวอย่างการเข้าถึง Application Menu ของ SIM ระบบ True

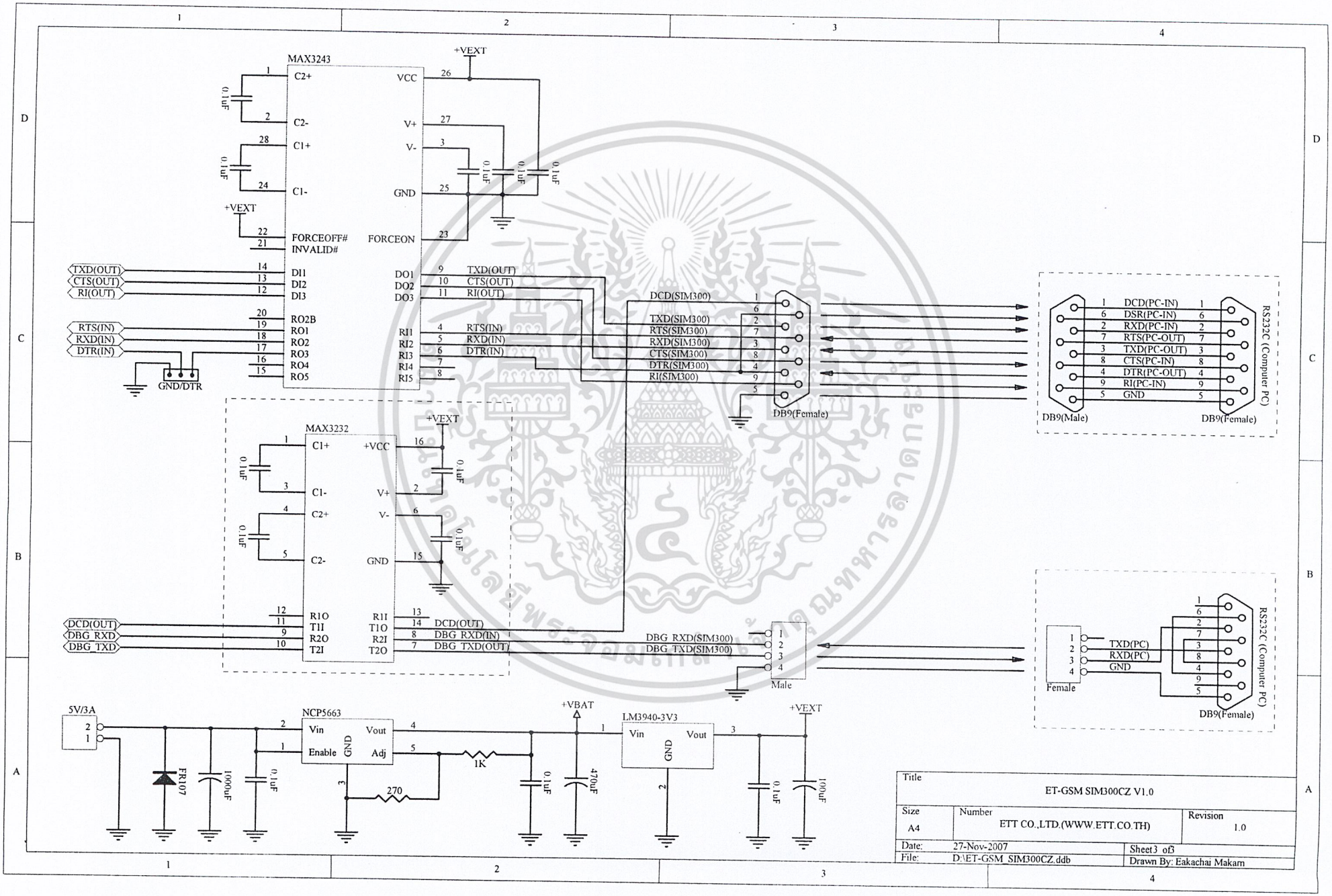
หมายเหตุ	ข้อความในกรอบสีแดง หมายถึง คำสั่งที่สั่งให้โมดูล SIM300CZ ข้อความในกรอบสีเหลือง หมายถึง ข้อความที่ตอบกลับมาจากโมดูล SIM300CZ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


+VBAT
1N4148
BUZZER
1N4148
4K7
BUZZER
BC547
47K
+VEXT
330
STATUS
4K7
STATUS
BC547
47K
+VEXT
330
NETLIGHT
4K7
NETLIGHT
BC547
47K
+VBAT
VBAT
GND
VCHG
ADC0
TEMP_BAT
VRTC
VDD_EXT
NETLIGHT
PWRKEY
KBC0
STATUS
KBC1
GPIO0
KBC2
BUZZER
KBC3
SIM_VDD
KBC4
SIM_RST
KBR0
SIM_DATA
KBR1
SIM_CLK
KBR2
SIM_PRESENT
KBR3
GPIO1
KBR4
DCD
DISP_CS
DTR
DISP_CLK
RXD
DISP_DATA
TXD
DISP_D/C
RTS
DISP_RST
CTS
DBG_RXD
RI
DBG_TXD
AGND
AGND
SPK1P
MIC1P
SPK1N
MIC1N
SPK2P
MIC2P
SPK2N
MIC2N
VDD_EXT
PWR_KEY
STATUS
BUZZER
SIM_VDD
SIM_RST
SIM_DATA
SIM_CLK
DCD(OUT)
DTR(IN)
RXD(IN)
TXD(OUT)
RTS(IN)
CTS_OUT)
RI(OUT)
SPK1P
SPK1N
NETLIGHT
DBG_RXD
DBG_TXD
MIC1P
MIC1N
SIM300CZ
+VBAT
10K
PWRKEY
ON/OFF
100uF
+VBAT
330
VBAT
VDD_EXT
330
POWER
Title
ET-GSM SIM300CZ V1.0
Size
A4
Number
ETT CO.,LTD.(WWW.ETT.CO.TH)
Revision
1.0
Date:
27-Nov-2007
Sheet1 of3
File:
D:\ET-GSM_SIM300CZ.ddb
Drawn By: Eakachai Makarn



Title	ET-GSM SIM300CZ V1.0	
Size: A4	Number: ETT CO.,LTD.(WWW.ETT.CO.TH)	Revision: 1.0
Date: 27-Nov-2007		Sheet 2 of 3
File: D:\ET-GSM_SIM300CZ.ddb		Drawn By: Eakachai Makarn

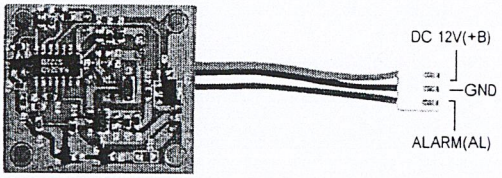

MAX3243
C2+
C2-
C1+
C1-
VCC
V+
V-
GND
FORCEOFF#
INVALID#
FORCEON
+VEXT
0.1uF
DI1
DI2
DI3
RO2B
RO1
RO2
RO3
RO4
RO5
DO1
DO2
DO3
RI1
RI2
RI3
RI4
RI5
TXD(OUT)
CTS(OUT)
RI(OUT)
RTS(IN)
RXD(IN)
DTR(IN)
GND/DTR
DCD(SIM300)
TXD(SIM300)
RTS(SIM300)
RXD(SIM300)
CTS(SIM300)
DTR(SIM300)
RI(SIM300)
DB9(Female)
DB9(Male)
DB9(Female)
RS232C (Computer PC)
DCD(PC-IN)
DSR(PC-IN)
RXD(PC-IN)
RTS(PC-OUT)
TXD(PC-OUT)
CTS(PC-IN)
DTR(PC-OUT)
RI(PC-IN)
GND
MAX3232
C1+
C2+
C2-
+VCC
V+
V-
GND
R1O
T1I
R2O
T2I
R1I
T1O
R2I
T2O
DCD(OUT)
DBG_RXD
DBG_TXD
DBG_RXD(IN)
DBG_TXD(OUT)
DBG_RXD(SIM300)
DBG_TXD(SIM300)
Male
Female
TXD(PC)
RXD(PC)
GND
DB9(Female)
RS232C (Computer PC)
5V/3A
FR107
1000uF
0.1uF
NCP5663
Vin
Enable
GND
Vout
Adj
270
1K
470uF
+VBAT
LM3940-3V3
Vin
GND
Vout
100uF
Title
ET-GSM SIM300CZ V1.0
Size
A4
Number
ETT CO.,LTD.(WWW.ETT.CO.TH)
Revision
1.0
Date:
27-Nov-2007
Sheet3 of3
File:
D:\ET-GSM_SIM300CZ.ddb
Drawn By: Eakachai Makarn

ภาคผนวก ค.

Datasheet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

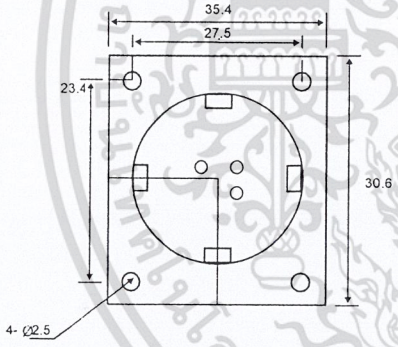
SE-10



• BACK SIDE

• FRONT SIDE

Dimensions



Specifications

ITEM		Specification	Unit	Condition
Sensor Type		Dual Element		
Housing		TO 5		
Element Size		2×1	mm	
Spacing		1	mm	
Responsivity	Min	3.2	xv/w	7...14mm, 1Hz, 100℃
	Typ	4.0		(One element cover)
Match	Max	<10	%	7...14mm, 1Hz, 100℃
				(Both element expose)
Noise	Typ	20	μVp-p	25℃, 0.4...10Hz
	Max	50	V	
Effect Voltage	Min	0.2		Re=47XΩ
	Max	1.5		
Window Material		Silicon, coated		
Spectral Range	Transmission	T>30 average	%	7...14mm
	Blocking	T<0.1		<5mm
Operating Voltage		12	V	
Operating Temperature		-10~40	℃	
Storage Temperature		-40~80	℃	

เอกสารนี้จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VISUAL

Intertec Components

DIGITAL PRODUCTS

ITM-C-328

Manual

Release Note:

1. Jan 28, 2004 – official released v1.0

2. Feb 24, 2004 – official released v1.1
   - Fix the definition of verify code
   - Fix the bug of unable jump to power save mode
   - Fix the incorrect connection speed after wake up from power save mode

3. Apr 24, 2004 – official released v2.0
   - Add auto baud-rate detection
   - Add support of 9600bps, 19200bps, 38400bps
   - Disable the 8-bit colour for uncompressed picture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

General Description

The C328 module is a highly integrated serial camera board that can be attached to a wireless or PDA host performing as a video camera or a JPEG compressed still camera. It provides a serial interface (RS-232) and JPEG compression engine to act as a low cost and low powered camera module for high-resolution serial bus security system or PDA accessory applications.

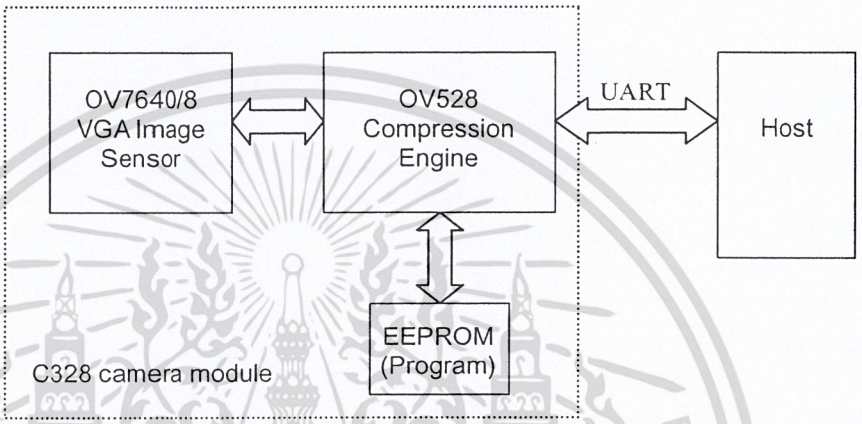


Figure 1 – System block diagram

Features

- Small in size, low cost and low powered (3.3V) camera module for high-resolution serial bus security system or PDA accessory applications.
- On-board EEPROM provides a command-based interface to external host via RS-232.
- UART: 115.2Kbps for transferring JPEG still pictures or 160x128 preview @8bpp with 0.75fps.
- On board OmniVision OV7640/8 VGA color sensor.
- Built-in JPEG CODEC for different resolutions.
- Built-in down sampling, clamping and windowing circuits for VGA, QVGA, 160x120 or 80x60 image resolutions.
- Built-in color conversion circuits for 2-bit gray, 4-bir gray, 8-bit gray, 12-bit RGB, 16-bit RGB or standard JPEG preview images.
- No external DRAM required.

System Configuration

1. Camera Sensor
   The C328-7640 module uses OmniVision OV7640/8 VGA color digital CameraChips with an 8-bit YCbCr interface.

2. OV528 Serial Bridge
   The OV528 Serial Bridge is a JPEG CODEC embedded controller chip that can compress and transfer image data from CameraChips to external device. The OV528 takes 8-bit YCbCr 422 progressive video data from an OV7640/8 CameraChip. The camera interface synchronizes with input video data and performs down sampling, clamping and windowing functions with desired resolution, as well as color conversion that is requested by the user through serial bus host commands.
   The JPEG CODEC can achieve higher compression ratio and better image quality for various image resolutions.

3. Program EEPROM
   A serial type program memory is built-in for C328-7640 to provide a set of user-friendly command interfacing to external host.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Board Layout

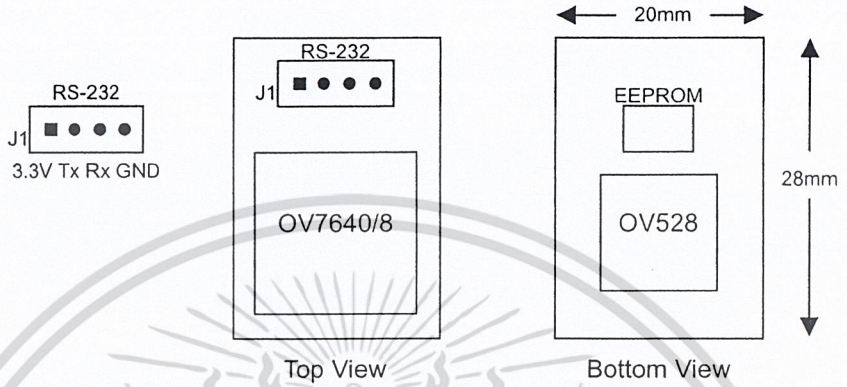


Figure 2 – C328-7640 board layout and serial interface pin

Serial Interface

1. Single Byte Timing Diagram
   A single byte RS-232 transmission consists of the start bit, 8-bit contents and the stop bit. A start bit is always 0, while a stop bit is always 1. LSB is sent out first and is right after the start bit.

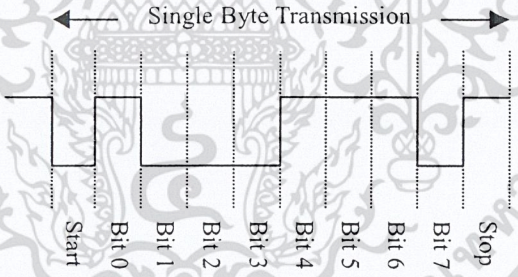


Figure 3 – RS-232 single byte timing diagram

2. Command Timing Diagram
   A single command consists of 6 continuous single byte RS-232 transmissions. The following is an example of SYNC (AA0D00000000h) command.

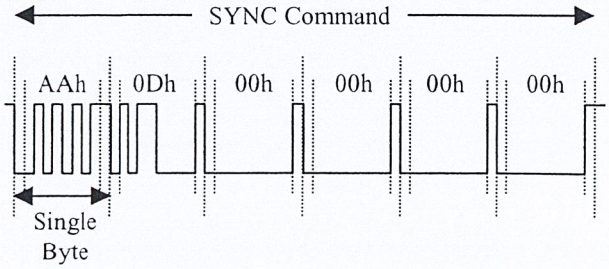


Figure 4 – RS-232 SYNC command timing diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Set

The C328-7640 module supports total 11 commands for interfacing to host as following:

Command	ID Number	Parameter1	Parameter2	Parameter3	Parameter4
Initial	AA01h	00h	Color Type	Preview Resolution	JPEG Resolution
Get Picture	AA04h	Picture Type	00h	00h	00h
Snapshot	AA05h	Snapshot Type	Skip Frame Low Byte	Skip Frame High Byte	00h
Set Package Size	AA06h	08h	Package Size Low Byte	Package Size High Byte	00h
Set Baudrate	AA07h	1st Divider	2nd Divider	00h	00h
Reset	AA08h	Reset Type	00h	00h	xxh*
Power Off	AA09h	00h	00h	00h	00h
Data	AA0Ah	Data Type	Length Byte 0	Length Byte 1	Length Byte 2
SYNC	AA0Dh	00h	00h	00h	00h
ACK	AA0Eh	Command ID	ACK counter	00h / Package ID Byte 0	00h / Package ID Byte 1
NAK	AA0Fh	00h	NAK counter	Error Number	00h

* If the parameter is 0xFF, the command is a special Reset command and the firmware responds to it immediately.

1. Initial (AA01h)

The host issues this command to configure the preview image size and color type. After receiving this command, the module will send out an ACK command to the host if the configuration success. Otherwise, an NACK command will be sent out.

1.1 Color Type

C328-7640 can support 7 different color types as follow:

2-bit Gray Scale	01h
4-bit Gray Scale	02h
8-bit Gray Scale	03h
12-bit Color	05h
16-bit Color	06h
JPEG	07h

1.2 Preview Resolution

80x60	01h
160x120	03h

1.3 JPEG Resolution

Since the Embedded JPEG Code can support only multiple of 16, the JPEG preview mode can support following image sizes. It is different from normal preview mode.

80x64	01h
160x128	03h
320x240	05h
640x480	07h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Set

The C328-7640 module supports total 11 commands for interfacing to host as following:

Command	ID Number	Parameter1	Parameter2	Parameter3	Parameter4
Initial	AA01h	00h	Color Type	Preview Resolution	JPEG Resolution
Get Picture	AA04h	Picture Type	00h	00h	00h
Snapshot	AA05h	Snapshot Type	Skip Frame Low Byte	Skip Frame High Byte	00h
Set Package Size	AA06h	08h	Package Size Low Byte	Package Size High Byte	00h
Set Baudrate	AA07h	1st Divider	2nd Divider	00h	00h
Reset	AA08h	Reset Type	00h	00h	xxh*
Power Off	AA09h	00h	00h	00h	00h
Data	AA0Ah	Data Type	Length Byte 0	Length Byte 1	Length Byte 2
SYNC	AA0Dh	00h	00h	00h	00h
ACK	AA0Eh	Command ID	ACK counter	00h / Package ID Byte 0	00h / Package ID Byte 1
NAK	AA0Fh	00h	NAK counter	Error Number	00h

* If the parameter is 0xFF, the command is a special Reset command and the firmware responds to it immediately.

1. Initial (AA01h)

The host issues this command to configure the preview image size and color type. After receiving this command, the module will send out an ACK command to the host if the configuration success. Otherwise, an NACK command will be sent out.

1.1 Color Type

C328-7640 can support 7 different color types as follow:

2-bit Gray Scale	01h
4-bit Gray Scale	02h
8-bit Gray Scale	03h
12-bit Color	05h
16-bit Color	06h
JPEG	07h

1.2 Preview Resolution

80x60	01h
160x120	03h

1.3 JPEG Resolution

Since the Embedded JPEG Code can support only multiple of 16, the JPEG preview mode can support following image sizes. It is different from normal preview mode.

80x64	01h
160x128	03h
320x240	05h
640x480	07h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Set

The C328-7640 module supports total 11 commands for interfacing to host as following:

Command	ID Number	Parameter1	Parameter2	Parameter3	Parameter4
Initial	AA01h	00h	Color Type	Preview Resolution	JPEG Resolution
Get Picture	AA04h	Picture Type	00h	00h	00h
Snapshot	AA05h	Snapshot Type	Skip Frame Low Byte	Skip Frame High Byte	00h
Set Package Size	AA06h	08h	Package Size Low Byte	Package Size High Byte	00h
Set Baudrate	AA07h	1st Divider	2nd Divider	00h	00h
Reset	AA08h	Reset Type	00h	00h	xxh*
Power Off	AA09h	00h	00h	00h	00h
Data	AA0Ah	Data Type	Length Byte 0	Length Byte 1	Length Byte 2
SYNC	AA0Dh	00h	00h	00h	00h
ACK	AA0Eh	Command ID	ACK counter	00h / Package ID Byte 0	00h / Package ID Byte 1
NAK	AA0Fh	00h	NAK counter	Error Number	00h

* If the parameter is 0xFF, the command is a special Reset command and the firmware responds to it immediately.

1. Initial (AA01h)

The host issues this command to configure the preview image size and color type. After receiving this command, the module will send out an ACK command to the host if the configuration success. Otherwise, an NACK command will be sent out.

1.1 Color Type

C328-7640 can support 7 different color types as follow:

2-bit Gray Scale	01h
4-bit Gray Scale	02h
8-bit Gray Scale	03h
12-bit Color	05h
16-bit Color	06h
JPEG	07h

1.2 Preview Resolution

80x60	01h
160x120	03h

1.3 JPEG Resolution

Since the Embedded JPEG Code can support only multiple of 16, the JPEG preview mode can support following image sizes. It is different from normal preview mode.

80x64	01h
160x128	03h
320x240	05h
640x480	07h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Get Picture (AA04h)

The host gets a picture from C328-7640 by sending this command.

2.1 Picture Type

Snapshot Picture	01h
Preview Picture	02h
JPEG Preview Picture	05h

3. Snapshot (AA05h)

C328-7640 keeps a single frame of JPEG still picture data in the buffer after receiving this command.

3.1 Snapshot Type

Compressed Picture	00h
Uncompressed Picture	01h

3.2 Skip Frame Counter

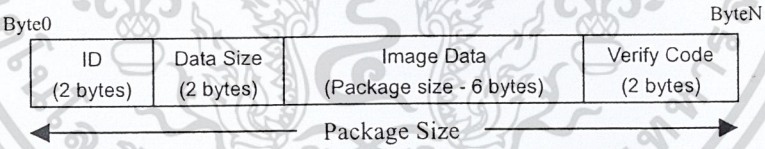
The number of dropped frames can be defined before compression occurs. "0" keeps the current frame, "1" captures the next frame, and so forth.

4. Set Package Size (AA06h)

The host issues this command to change the size of data package which is used to transmit JPEG image data from the C328-7640 to the host. This command should be issued before sending Snapshot command or Get Picture command to C328-7640. It is noted that the size of the last package varies for different image.

4.1 Package Size

The default size is 64 bytes and the maximum size is 512 bytes.



ID	-> Package ID, starts from zero for an image
Data Size	-> Size of image data in the package
Verify Code	-> Error detection code, equals to the lower byte of sum of the whole package data except the verify code field. The higher byte of this code is always zero. i.e. verify code = lowbyte(sum(byte[0] to byte[N-2]))

Note: As the transmission of uncompressed image is not in package mode, it is not necessary to set the package size for uncompressed image.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Set Baudrate (AA07h)

Set the C328-7640 baud rate by issuing this command. As the module can auto-detect the baud rate of the incoming command, host can make connection with one of the following baud rate in the table. The module will keep using the detected baud rate until physically power off

5.1 Baudrate Divider

Baudrate = 14.7456MHz / 2 x (2nd Divider + 1) / 2 x (1st Divider + 1)

Baudrate	1st Divider	2nd Divider	Baudrate	1st Divider	2nd Divider
7200 bps	ffh	01h	28800 bps	3fh	01h
9600 bps	bfh	01h	38400 bps	2fh	01h
14400 bps	7fh	01h	57600 bps	1fh	01h
19200 bps	5fh	01h	115200 bps	0fh	01h

6. Reset (AA08h)

The host reset C328-7640 by issuing this command.

6.1 Reset Type

"00h" resets the whole system. C328-7640 will reboot and reset all registers and state machines.
"01h" resets state machines only.

7. Power Off (AA09h)

C328-7640 will go into sleep mode after receiving this command. SYNC command (AA0Dh) must be sent to wake up C328-7640 for certain period until receiving ACK command from C328-7640.

8. Data (AA0Ah)

C328-7640 issues this command for telling the host the type and the size of the image data which is ready for transmitting out to the host.

8.1 Data Type

Snapshot Picture	01h
Preview Picture	02h
JPEG Preview Picture	05h

8.2 Length

These three bytes represent the length of data of the Snapshot Picture, Preview Picture or JPEG Preview Picture.

9. SYNC (AA0Dh)

Either the host or the C328-7640 can issue this command to make connection. An ACK command must be sent out after receiving this command.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Set Baudrate (AA07h)

Set the C328-7640 baud rate by issuing this command. As the module can auto-detect the baud rate of the incoming command, host can make connection with one of the following baud rate in the table. The module will keep using the detected baud rate until physically power off

5.1 Baudrate Divider

Baudrate = 14.7456MHz / 2 x (2nd Divider + 1) / 2 x (1st Divider + 1)

Baudrate	1st Divider	2nd Divider	Baudrate	1st Divider	2nd Divider
7200 bps	ffh	01h	28800 bps	3fh	01h
9600 bps	bfh	01h	38400 bps	2fh	01h
14400 bps	7fh	01h	57600 bps	1fh	01h
19200 bps	5fh	01h	115200 bps	0fh	01h

6. Reset (AA08h)

The host reset C328-7640 by issuing this command.

6.1 Reset Type

"00h" resets the whole system. C328-7640 will reboot and reset all registers and state machines.
"01h" resets state machines only.

7. Power Off (AA09h)

C328-7640 will go into sleep mode after receiving this command. SYNC command (AA0Dh) must be sent to wake up C328-7640 for certain period until receiving ACK command from C328-7640.

8. Data (AA0Ah)

C328-7640 issues this command for telling the host the type and the size of the image data which is ready for transmitting out to the host.

8.1 Data Type

Snapshot Picture	01h
Preview Picture	02h
JPEG Preview Picture	05h

8.2 Length

These three bytes represent the length of data of the Snapshot Picture, Preview Picture or JPEG Preview Picture.

9. SYNC (AA0Dh)

Either the host or the C328-7640 can issue this command to make connection. An ACK command must be sent out after receiving this command.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ACK (AA0Eh)

This command indicates the success of last operation. After receiving any valid command, ACK command must be sent out except when getting preview data. The host can issue this command to request image data package with desired package ID after receiving Data command from C328-7640. The host should send this command with package ID F0F0h after receiving a package to end the package transfer. Note that the field "command ID" should be 00h when request image data package.

10.1 Command ID

The command with that ID is acknowledged by this command.

10.2 ACK Counter

No use.

10.3 Package ID

For acknowledging Data command, these two bytes represent the requested package ID. While for acknowledging other commands, these two bytes are set to 00h.

11. NAK (AA0Fh)

This command indicates corrupted transmission or unsupported features.

11.1 NAK Counter

No use.

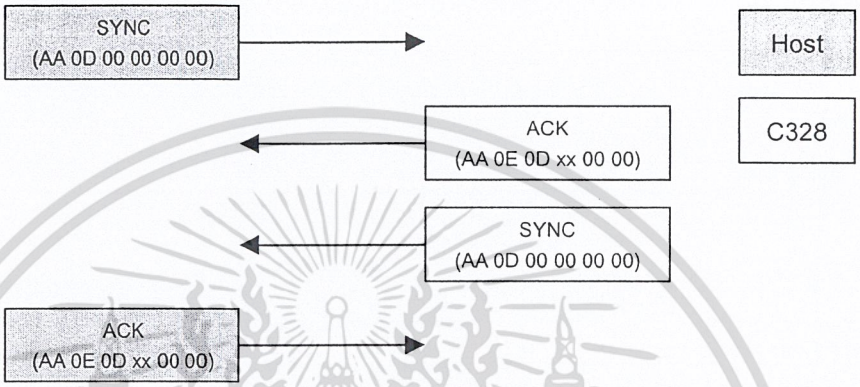
11.2 Error Number

Picture Type Error	01h	Parameter Error	0bh
Picture Up Scale	02h	Send Register Timeout	0ch
Picture Scale Error	03h	Command ID Error	0dh
Unexpected Reply	04h	Picture Not Ready	0fh
Send Picture Timeout	05h	Transfer Package Number Error	10h
Unexpected Command	06h	Set Transfer Package Size Wrong	11h
SRAM JPEG Type Error	07h	Command Header Error	F0h
SRAM JPEG Size Error	08h	Command Length Error	F1h
Picture Format Error	09h	Send Picture Error	F5h
Picture Size Error	0ah	Send Command Error	ffh

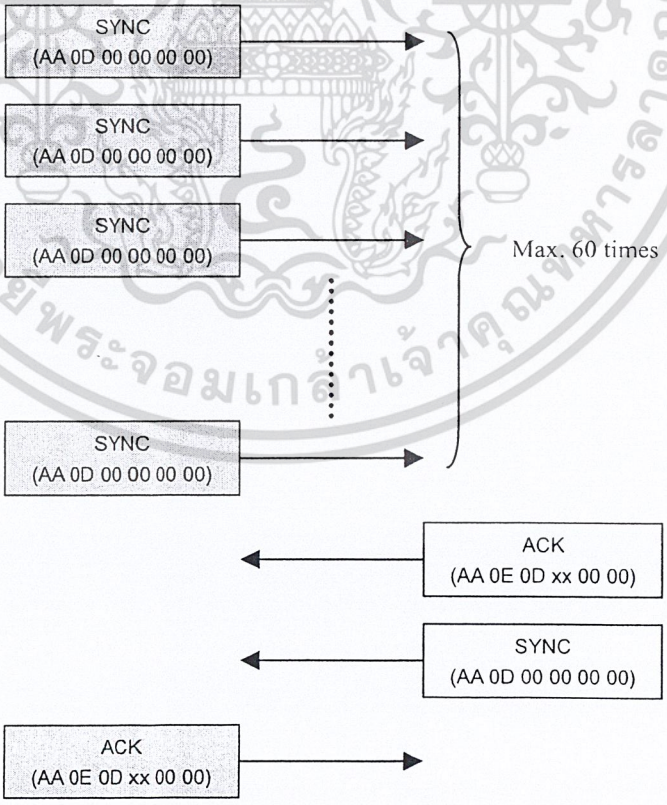
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Protocol

1. SYNC Command

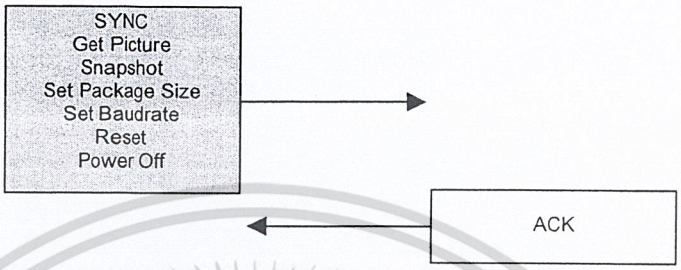


2. Make Connection with C328-7640
Send the SYNC command (at 14400bps) until receiving ACK command from C328-7640 (usually an ACK command is receive after sending 25 times of SYNC command). This must be done after power up.



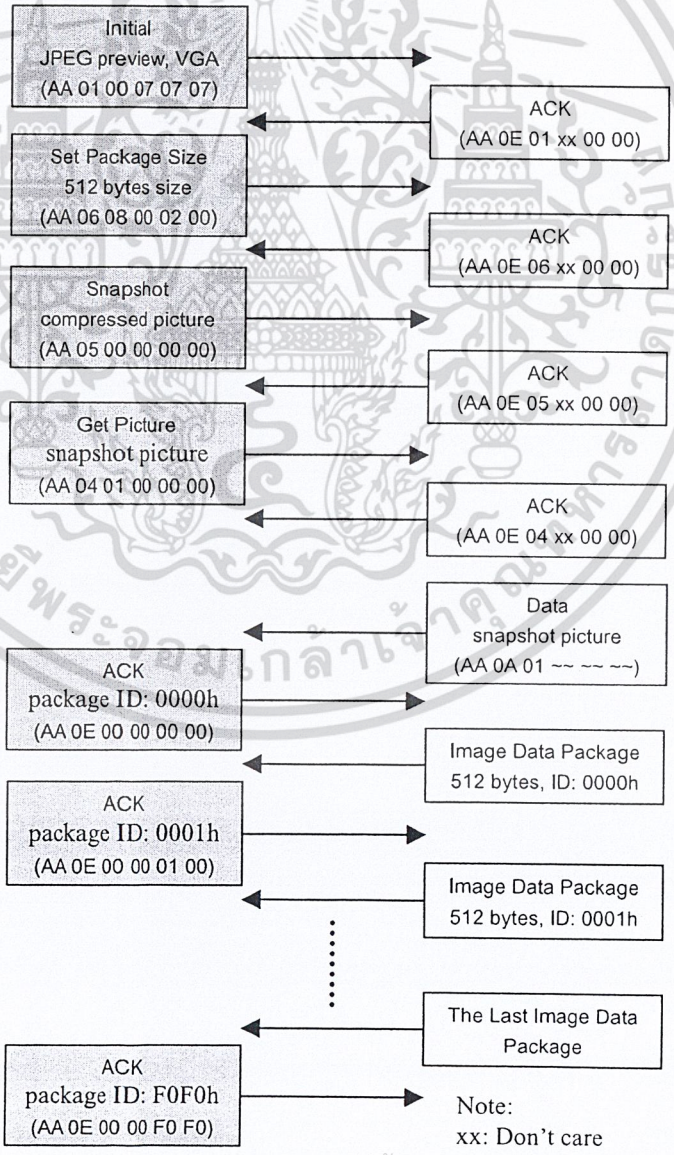
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Initial, Get Picture, Snapshot, Set Package Size, Set Baudrate, Reset and Power Off Command



4. Getting a Snapshot for RS232
Make sure connection is made before the following communication.

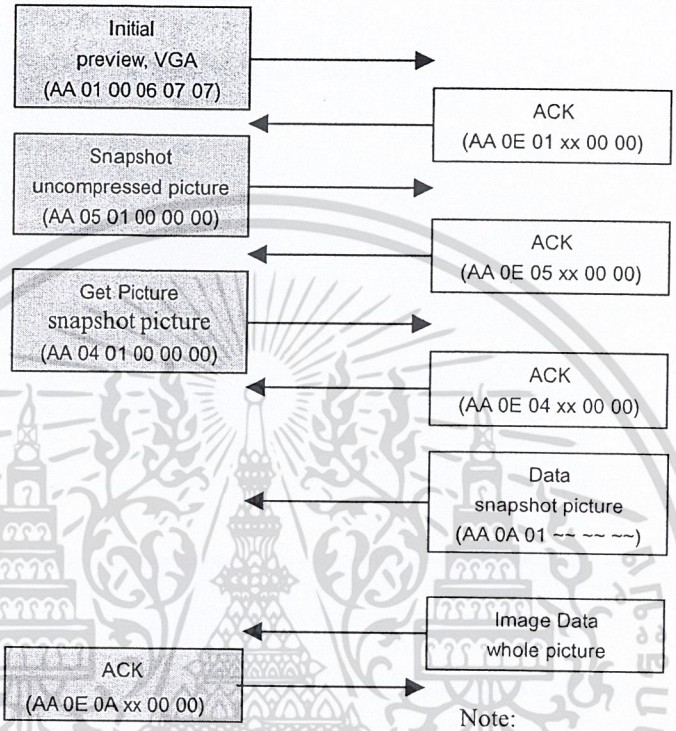
4.1 JPEG Snapshot Picture



Note:
xx: Don't care
~~: Image size returned by C328

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 Snapshot Picture (uncompressed snapshot picture)

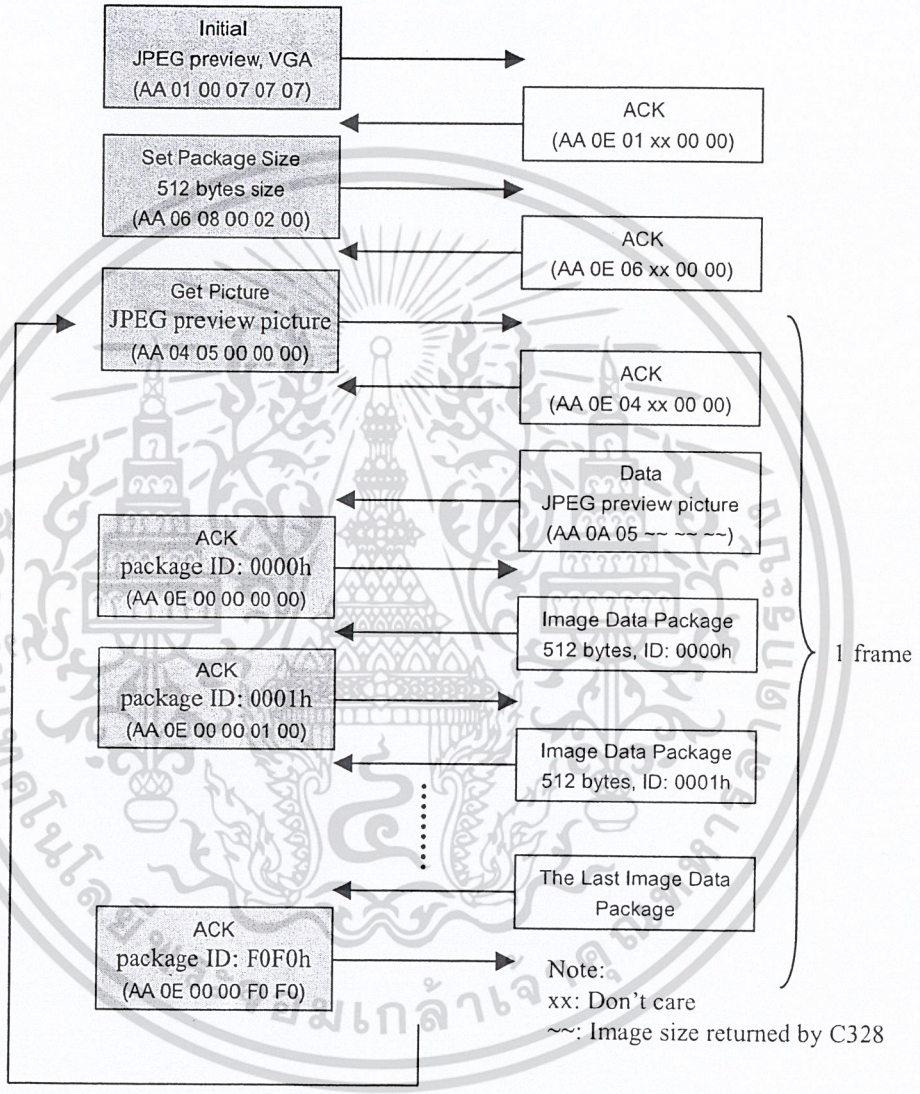


Note:
xx: Don't care
~~: Image size returned by C328

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

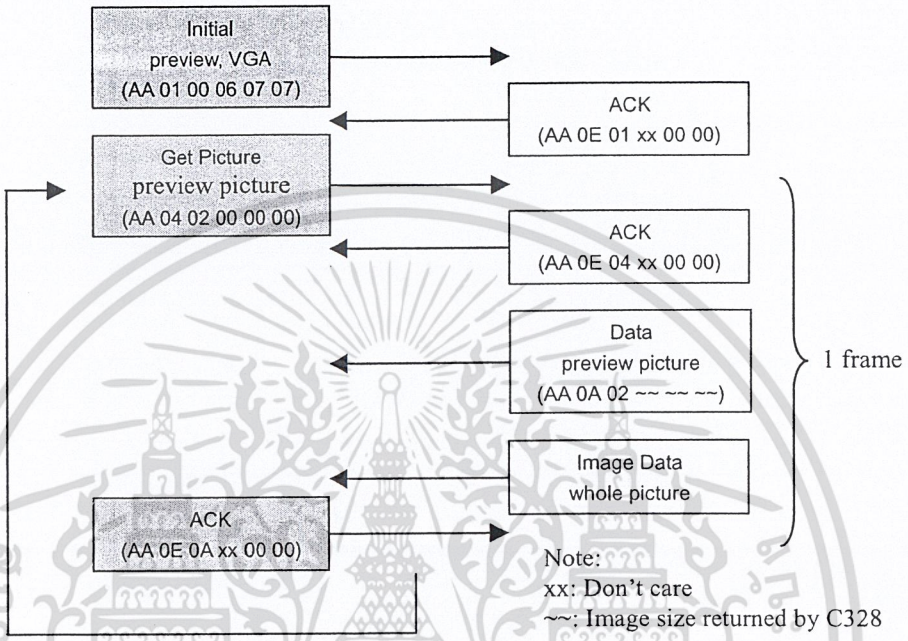
5. Getting JPEG preview pictures (video) for RS232
Make sure connection is made before the following communication.

5.1 JPEG Preview Picture



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 Preview Picture (uncompressed preview picture)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้