

โมบายอินเตอร์เฟส
MOBILE INTERFACE



นาย คชา สมทา รหัสประจำตัว 44015318
นาย สาทิต วิเชียรพงษ์เจริญ รหัสประจำตัว 44015354

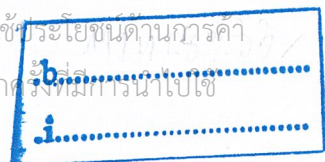


อาจารย์ที่ปรึกษา
ดร. วีชระ นัครวิริยะ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2546

21/7/46
21/46

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
เลขทะเบียน 55132
ไม่หวัดลิขสิทธิ์ในการดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
วัน,เดือน,ปี - 8 เม.ย. 2548



นโยบายอินเทอร์เน็ตเฟส

นาย กษา สมทา 44015318

นาย ศาธิต วิเชียรพงษ์เจริญ 44015354

ดร.วัชร ฉัตรวิริยะ อาจารย์ที่ปรึกษา

ปีการศึกษา 2546

บทคัดย่อ

โครงการนี้ เป็นการสร้างกล่องอินเทอร์เน็ตเฟส เพื่อนำมาใช้ในการเชื่อมต่อระหว่างคีย์บอร์ดของคอมพิวเตอร์และโทรศัพท์มือถือ โดยในกล่องอินเทอร์เน็ตเฟสนี้จะแบ่งออกเป็นสองส่วนใหญ่ๆคือ ส่วนที่ทำการรับข้อมูลเพื่อนำมาแปลงสแกนโค้ดและส่วนของการควบคุมบอร์ดทั้งหมด ในการออกแบบควบคุมจะใช้ไมโครคอนโทรลเลอร์ในการทำงานสองตัว นอกจากนั้นยังมีอุปกรณ์ต่อพ่วงอีกด้วย เช่น แรม แอลซีดี การต่อจอแอลซีดีก็เพื่อให้แสดงผลตอนเราพิมพ์ข้อความ ส่วนการเชื่อมต่อกับอุปกรณ์ภายนอกนั้น จะมีพอร์ตในการเชื่อมต่ออยู่ โดยทางอินพุตจะใช้พอร์ตพีเอสยูเพื่อต่อกับคีย์บอร์ดคอมพิวเตอร์ และทางเอาต์พุตจะใช้พอร์ตอนุกรมในการเชื่อมต่อกับสายดาต้าลิงค์ของโทรศัพท์มือถือ ทางด้านของตัวโปรแกรมจะแบ่งออกเป็นสองส่วนเช่นเดียวกัน ส่วนที่รับข้อมูลจะเขียนจากภาษาซีเพราะว่ากำหนดระดับบิตของข้อมูลได้ง่าย ส่วนตัวโปรแกรมตัวหลักที่ควบคุมการทำงานกล่องทั้งหมดจะเขียนจากภาษาเบสิกเพราะว่าสามารถเขียนควบคุมอุปกรณ์ภายนอกได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOBILE INTERFACE**Kacha Somtha****Satit Wicheanphongjarorn****Dr. Watchara Chadviriya Advisor****ABSTRACT**

This project is create interface box for use to interface between keyboard of computer and mobile. This interface box will divide two parts . The part of input for use to translate mobile. This interface box will divide two parts . The part of input for use two micro controller and equipment such as RAM and LCD. LCD with display when we type the message . The part of interface with external equipment. It has port for interface. Input will us use PS2 port for interface with keyboard and output will use serial port to interface with data link of mobile The part of main board will divide two parts. The part of input will write form C language because it able to fix level of bit easy. The part of main program which control will write from Basic language because it able to easy control external equipment.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการนี้คงจะไม่สำเร็จลงได้ด้วยดี หากไม่ได้รับการช่วยเหลือและความร่วมมือจากหลายฝ่ายด้วยกันบุคคลแรกที่มีส่วนสำคัญที่ทำให้โครงการนี้สำเร็จลงได้ด้วยดีคือ อาจารย์ วัชร นัทรวิริยะ ซึ่งเป็นอาจารย์ที่ดูแลนักศึกษาห้อง Network และเป็นอาจารย์ที่ปรึกษาของโครงการนี้ ที่คอยให้คำแนะนำช่วยเหลือเอาใจใส่ตลอดมาจะขอขอบคุณเพื่อนๆทุกคนที่ให้ความช่วยเหลือในทุกๆเรื่อง ขอขอบคุณด้วยใจจริง

และต้องขอขอบพระคุณบุคคลสำคัญที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูผู้เขียนมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอมาและ ขอกราบขอบพระคุณมา ณ ที่นี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	VII
บทที่ 1 บทนำ	I
1.1 วัตถุประสงค์	1
1.2 ขอบเขตของงานวิจัย	1
1.3 ความสำคัญและที่มา	1
1.4 วิธีการดำเนินงาน	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 อินเทอร์เน็ตของมือถือ	3
2.2 ลักษณะการทำงานของกล่องอินเทอร์เน็ตเฟส	3
2.3 องค์ประกอบของกล่องอินเทอร์เน็ตเฟส	4
2.3.1 ไมโครคอนโทรลเลอร์	4
2.3.2 หน่วยความจำ	4
2.3.3 ส่วนของอุปกรณ์เชื่อมต่อ	4
2.3.4 ซอฟต์แวร์	4
2.4 หลักการส่งข้อมูลของคีย์บอร์ด	5
2.5 การเชื่อมต่ออุปกรณ์แบบ I ² C	6
2.5.1 ระบบบัสแบบ I ² C	6
2.5.2 การเขียนโปรแกรมควบคุมบัส I ² C	8
2.6 พอร์ตอนุกรม	9
2.6.1 การสื่อสารแบบซิงโครนัส	9
2.6.2 การสื่อสารแบบอะซิงโครนัส	9
2.6.3 มาตรฐานพอร์ตอนุกรมแบบ RS-232	10
2.6.4 คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ	11
2.6.5 UART	14
2.6.6 ชนิดของ UART	14
2.7 เอสเอ็มเอสในรูปแบบของพีดียู	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.8 เอทีคอมมาน	15
2.8.1 ลักษณะของชุดคำสั่ง AT+CMGF	16
2.8.2 ลักษณะของชุดคำสั่ง AT+CMGS กรณีสั่งแบบ Text-Mode	16
2.8.3 ลักษณะของชุดคำสั่ง AT+CMGS กรณีสั่งแบบ PDU-Mode	16
2.8.4 ลักษณะของชุดคำสั่ง AT+CMGD	17
2.8.5 ลักษณะของชุดคำสั่ง AT+CMGW กรณีสั่งแบบ Text-Mode	17
2.8.6 ลักษณะของชุดคำสั่ง AT+CMSS	18
2.8.7 ลักษณะของชุดคำสั่ง AT+CMGW กรณีสั่งแบบ PDU-Mode	18
2.8.8 ลักษณะของชุดคำสั่ง AT+CPMS	19
บทที่ 3 การวิเคราะห์และการออกแบบระบบ	20
3.1 ภาพรวมของระบบ	20
3.2 การออกแบบส่วนของการรับข้อมูลและแปลงสแกนโค้ดเป็นรหัสแอสกี	21
3.3 การออกแบบส่วนการควบคุมการทำงานของกล่องอินเตอร์เฟสทั้งหมด	23
3.4 การทำงานในส่วนของเซิร์ฟเวอร์(Server)	25
บทที่ 4 การทดสอบและวิเคราะห์ผล	28
4.1 การทดสอบการทำงานของฮาร์ดแวร์	28
4.2 การทดสอบการทำงานของซอฟต์แวร์	31
บทที่ 5 บทวิจารณ์และผลสรุป	34
5.1 ผลที่ได้รับจากโครงการ	34
5.2 ปัญหาที่พบในกรทำโครงการ	34
5.2.1 ปัญหาทางด้านฮาร์ดแวร์	34
5.2.2 ปัญหาทางด้านซอฟต์แวร์	34
5.2.3 ปัญหาทางด้านเทคนิคต่างๆ	34
5.3 แนวทางในการพัฒนาต่อ	35
5.4 สรุปผลโครงการ	35
ภาคผนวก ก. โค้ดของโปรแกรมที่เขียนไว้บนไมโครคอนโทรลเลอร์	36
ภาคผนวก ข. โค้ดของโปรแกรมในส่วนของเซิร์ฟเวอร์	54
ภาคผนวก ค. PDU Format	63
ภาคผนวก ง. คำสั่ง AT COMMAND	65
บรรณานุกรม	121

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 แสดงรูปของพอร์ตมอด็มอีรุ่นต่างๆ	3
รูปที่ 2.2 สแกนโค้ดของคีย์บอร์ดของบริษัท IBM	5
รูปที่ 2.3 สแกนโค้ดของคีย์บอร์ดของบริษัท IBM	5
รูปที่ 2.4 เป็นพอร์ตของคีย์บอร์ดพีเอสทูแบบ 6 พิน	5
รูปที่ 2.5 การต่อระบบบัสแบบ I ² C	6
รูปที่ 2.6 แสดงสถานะของระบบบัสแบบ I ² C	7
รูปที่ 2.7 ไบต์แอดเดรส	8
รูปที่ 2.8 แสดงไดอะแกรมการส่งข้อมูลอนุกรมแบบซิงโครนัส	9
รูปที่ 2.9 แสดงไดอะแกรมการส่งข้อมูลอนุกรมแบบอะซิงโครนัส	10
รูปที่ 2.10 การจัดขาคอนเน็กเตอร์ของพอร์ตอนุกรมตามมาตรฐาน RS-232	11
รูปที่ 2.11 แสดงรูปของคอนเน็กเตอร์แบบ DB-9	11
รูปที่ 2.12 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ	13
รูปที่ 3.1 แสดงการทำงานโดยรวมของระบบ	26
รูปที่ 3.2 แสดงภาพจริงของระบบโดยรวมทั้งหมด	27
รูปที่ 3.3 Flow chart ในส่วนของการแปลงสแกนโค้ดเป็นรหัสแอสกีหรือส่วนรับข้อมูล	28
รูปที่ 3.4 แสดงภาพในส่วนของการรับข้อมูลและแปลงสแกนโค้ดเป็นรหัสแอสกี	29
รูปที่ 3.5 แสดงภาพในส่วนของการควบคุมระบบทั้งหมดของกล่องอินเตอร์เฟส	29
รูปที่ 3.6 Flow chart ในส่วนการเปลี่ยนเป็น PDU mode และควบคุมการทำงานของกล่อง	30
รูปที่ 3.7 แสดง Flow chart ในส่วนการทำงานของเซอฟเวอร์	31
รูปที่ 3.8 แสดงไดอะแกรมการทำงานในส่วนเซอฟเวอร์	32
รูปที่ 4.1 แสดงหน้าต่างโปรแกรม S3Term ซึ่งเป็นโปรแกรมที่ใช้ในการทดสอบฮาร์ดแวร์	33
รูปที่ 4.2 แสดงไดอะแกรมของการทดสอบฮาร์ดแวร์	34
รูปที่ 4.3 แสดงรูปของโปรแกรม S3Term ซึ่งทำการเช็คค่าพร้อมทำการทดสอบแล้ว	34
รูปที่ 4.4 แสดงผลของการทดสอบโดยใช้โปรแกรม S3Term ในการทดสอบฮาร์ดแวร์	35
รูปที่ 4.5 แสดงการส่งข้อมูลที่เป็นภาษาไทยและภาษาอังกฤษปนกัน	36
รูปที่ 4.6 แสดงหน้าต่างของโปรแกรม PDUspy	37
รูปที่ 4.7 แสดงหน้าต่างของโปรแกรม BASCOM 8051 IDE และ simulate ของโปรแกรม	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 วัตถุประสงค์

เพื่อให้โทรศัพท์มือถือใช้งานเกี่ยวกับการส่งและการแลกเปลี่ยนข้อมูลได้ง่ายขึ้น โดยเฉพาะเกี่ยวกับงานการพิมพ์บนโทรศัพท์มือถือเพราะว่าเป็นพิมพ์ของคีย์บอร์ดมีขนาดใหญ่กว่ามาก และถ้าคนที่ใช้คีย์บอร์ดคอมพิวเตอร์เป็นประจำอยู่แล้วก็จะมาสามารถใช้ได้อย่างคล่องแคล่ว โดยที่เรายังสามารถนำคีย์บอร์ดที่ต่อกับโทรศัพท์มือถือไปประยุกต์ใช้งานต่างๆ ได้มากมายเช่น การแลกเปลี่ยนข้อมูลทางธุรกิจโดยใช้โทรศัพท์มือถือที่ต้องมีการพิมพ์ข้อมูลจำนวนมาก จึงทำให้เสียเวลามาก แต่ถ้าเราใช้คีย์บอร์ดของคอมพิวเตอร์ซึ่งมีขนาดใหญ่กว่าของมือถือมาก ทำให้งานพิมพ์ทำได้เร็วขึ้น ซึ่งในปัจจุบันนี้คีย์บอร์ดคอมพิวเตอร์ก็มีขนาดเล็กลงและสามารถพับได้ ทำให้สามารถพกพาไปในสถานที่ต่างๆ ได้อย่างสะดวก

1.2 ขอบเขตของงานวิจัย

สร้างตัวอินเตอร์เฟซ(Interface Box) ระหว่างตัวคีย์บอร์ดของคอมพิวเตอร์และโทรศัพท์มือถือ โดยทำให้มือถือสามารถรับข้อมูลการพิมพ์ผ่านทางคีย์บอร์ดได้โดยไม่ต้องใช้ปุ่มของมือถือเพราะปุ่มของโทรศัพท์มือถือมีขนาดเล็กทำให้ใช้งานยาก ดังนั้นการพิมพ์ผ่านทางคีย์บอร์ดคอมพิวเตอร์จึงสะดวกกว่าการพิมพ์ผ่านทางตัวโทรศัพท์มือถือเอง และทำให้มือถือสามารถส่งข้อความ(Short Message) โดยการพิมพ์ผ่านทางคีย์บอร์ดคอมพิวเตอร์ได้

1.3 ความสำคัญและที่มา

ในปัจจุบันนี้โทรศัพท์มือถือมีขนาดเล็กและราคาถูกลงกว่าเมื่อก่อนมากและเทคโนโลยีทางด้านมือถือหรืออุปกรณ์ไร้สายขนาดเล็กพวกนี้ยังถูกพัฒนาไปอย่างรวดเร็วในปัจจุบันเพื่อให้มีขนาดเล็กลงและประโยชน์ด้านการใช้งานเพิ่มมากขึ้นในขณะเดียวกันที่โทรศัพท์ที่มีขนาดเล็กลงนั้นเป็นผลทำให้ปุ่มกดของโทรศัพท์มือถือมีขนาดเล็กลงตามไปด้วยเช่นกันทำให้ปุ่มกดเพื่อพิมพ์งานที่เป็นตัวอักษรบนโทรศัพท์มือถือทำได้ลำบากและช้ามาก โทรศัพท์มือถือในปัจจุบันไม่ได้มีไว้เพื่อการสนทนาเพียงอย่างเดียวเท่านั้น แต่ยังมีฟังก์ชันพิเศษต่างๆ มากมายที่เพิ่มเข้ามา เช่น การส่งข้อความ(Short Message), การส่งเมลล์(Mail), ดูทีวี, ถ่ายรูปได้ การส่งข้อความ(Short Message) และการใช้งานฟังก์ชันต่างๆ ของโทรศัพท์มือถือ โดยการกดปุ่มโทรศัพท์ที่มีขนาดเล็กอยู่แล้วถ้ายังเกี่ยวกับงานพิมพ์จะทำให้เกิดความลำบากในการใช้งาน เราจึงคิดที่จะใช้คีย์บอร์ดคอมพิวเตอร์ซึ่งปุ่มมีขนาดใหญ่กว่ามาทำงานแทนปุ่มของโทรศัพท์มือถือ เพราะปุ่มคีย์บอร์ดคอมพิวเตอร์มีขนาดใหญ่กว่าทำให้ใช้งานได้ง่ายกว่าปุ่มของโทรศัพท์มือถือมาก และคีย์บอร์ดสมัยนี้ยังมีขนาดเล็กและสามารถพับเก็บได้ทำให้นำไปไหนมาไหนทำได้สะดวกมากยิ่งขึ้นและไม่เป็นปัญหาหรือภาระในการพกพา

1.4 วิธีการดำเนินงาน

การดำเนินงานเริ่มต้นจากศึกษาทฤษฎีและแนวความคิดต่างๆ ที่เกี่ยวข้องที่จะนำมาสร้างเป็นตัว อินเตอร์เฟซ (Interface Box) ระหว่างคีย์บอร์ดคอมพิวเตอร์กับโทรศัพท์มือถือจากนั้นก็สร้างตัวฮาร์ดแวร์ (Hardware)

ขึ้นมาจริงๆ และนำไปทดลองใช้ในเรื่องของการพิมพ์ โดยการส่งข้อความผ่านโทรศัพท์มือถือ(Short Message) เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Message) ได้โดยการคีย์ผ่านคีย์บอร์ดของคอมพิวเตอร์และไม่ต้องส่งผ่านทางปุ่มของโทรศัพท์โดยตรง สุดท้ายจะทำการทดลองทั้งหมดเพื่อหาข้อผิดพลาดและจุดบกพร่องเพื่อทำการแก้ไขจุดบกพร่องต่างๆ เหล่านั้น เพื่อให้ระบบทำงานได้อย่างสมบูรณ์ที่สุด

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ช่วยให้การพิมพ์ข้อความ(Short Message) ของโทรศัพท์มือถือผ่านทางคีย์บอร์ดของคอมพิวเตอร์มีความสะดวกและรวดเร็วยิ่งขึ้น ทำให้ประหยัดเวลาในการพิมพ์ได้มาก
2. ทำให้การทำธุรกรรมที่จำเป็นต้องใช้ข้อความ(Short Message)ในการติดต่อมีความสะดวกมากกว่าเดิมเช่น การรับส่งข้อมูลทางการธุรกิจ ทำให้ได้ส่งข้อมูลเข้าระบบทำได้สะดวก และรวดเร็ว เพราะเราสามารถพิมพ์ข้อมูลได้เร็วขึ้นกว่าแบบที่ใช้ปุ่มบนโทรศัพท์มือถืออย่างเดียว
3. ประหยัดค่าใช้จ่ายในการติดต่อสื่อสาร โดยใช้ มือถือ ในการส่งข้อความ(Short Message) เมื่อเทียบกับการใช้บริการ GPRS หรือการติดต่อผ่านอินเทอร์เน็ต ซึ่งมีราคาที่สูงกว่า



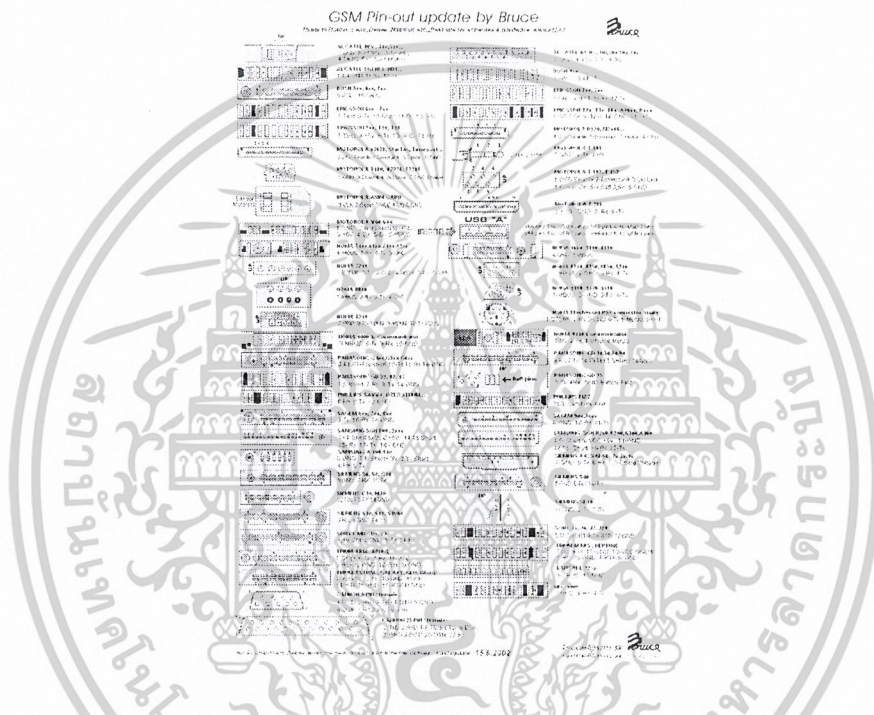
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 อินเทอร์เฟซของมือถือ(Mobile Interface)

กล่องอินเทอร์เฟซสำหรับมือถือ คือ กล่องที่เป็นตัวเชื่อมระหว่างอุปกรณ์ภายนอกกับตัวมือถือเอง โดยที่จะเชื่อมต่อเข้ากับมือถือตรงส่วนอยู่ทางด้านท้ายของมือถือที่เรียกกันว่าพอร์ตของมือถือซึ่งตรงส่วนนี้มือถือมีไว้เพื่อรับส่งข้อมูลเข้าและออกจากมือถือ การเชื่อมต่อเข้ากับพอร์ตของมือถือนี้จะต้องผ่านสายค่าาลังค์ของมือถือรุ่นนั้นเอง เพราะมือถือแต่ละรุ่นจะมีพอร์ตที่เอาไว้รับส่งข้อมูลต่างกัน ดังนั้นแต่ละรุ่นก็ต้องใช้สายค่าาลังค์ต่างกันด้วย จากรูปข้างล่างจะแสดงตัวอย่างพอร์ตของมือถือรุ่นต่างๆเอาไว้



รูปที่ 2.1 แสดงรูปของพอร์ตมือถือรุ่นต่างๆ

การสื่อสารของโทรศัพท์มือถือจะแบ่งออกเป็นสองแบบคือ แบบเอ็มบีเอส(MBUS) และแบบเอฟบีเอส(FBUS) ส่วนเครื่องโทรศัพท์มือถือของโนเกียจะใช้การสื่อสารแบบเอ็มบีเอสส่วนบริษัทโทรศัพท์มือถือค่ายอื่นๆจะใช้การสื่อสารแบบเอฟบีเอสทั้งหมด โดยที่การสื่อสารแบบเอ็มบีเอสนั้นจะใช้พินในการรับส่งข้อมูลเพียงพินเดียวเท่านั้น(ทั้งรับและส่งอยู่ในพินเดียวกัน) ส่วนการสื่อสารแบบเอฟบีเอสจะใช้สองพินคือใช้พินในการรับข้อมูลหนึ่งพินและพินในการส่งข้อมูลอีกหนึ่งพิน โดยพินทั้งแบบเอ็มบีเอสและเอฟบีเอสนี้จะอยู่ที่พอร์ตของมือถือทุกรุ่น

2.2 ลักษณะการทำงานของกล่องอินเทอร์เฟซ(Interface Box)

กล่องอินเทอร์เฟซที่ทำในโรงงานนี้จะมีลักษณะเป็นตัวเชื่อมระหว่างตัวคีย์บอร์ดของคอมพิวเตอร์และพอร์ตของโทรศัพท์มือถือโดยผ่านสายค่าาลังค์ การทำงานของระบบแบบคร่าวๆคือ เมื่อเราทำการกดคีย์บอร์ดคอมพิวเตอร์แล้วสแกนโค้ดก็จะถูกส่งจากตัวคีย์บอร์ดคอมพิวเตอร์มายังตัวกล่องอินเทอร์เฟซ จากนั้นตัวของกล่องอินเทอร์เฟซก็จะแปลงสแกนโค้ดที่ได้รับมาให้เป็นรหัสแอสกีแบบ 16 บิตเมื่อได้รับรหัสแอสกีมาแล้วก็จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำมาแปลงให้อยู่ในรูปแบบของเอสเอ็มเอส โดยการแปลงเป็นรูปแบบของเอสเอ็มเอสนี้จะอธิบายในหัวข้อถัดไป เมื่อได้รูปแบบของเอสเอ็มเอสมาแล้วก็จะนำไปเขียนกับเอทีคอมมานแล้วทำการส่งออกเพื่อให้โทรศัพท์มือถือโดยผ่านสายคาต้าลิงค์โดยสายคาต้าลิงค์นี้จะต่อกับ RS-232 ของกล่องอินเตอร์เฟสและพอร์ตของมือถือโดยมือถือแต่ละรุ่นก็จะมีสายคาต้าลิงค์แตกต่างกันออกไปเมื่อมือถือได้รับคำสั่งให้ส่งเอสเอ็มเอส โทรศัพท์มือถือก็จะส่งข้อความออกไปหาโทรศัพท์มือถือปลายทางโดยผ่านเครือข่ายของโทรศัพท์มือถือ

2.3 องค์ประกอบของกล่องอินเตอร์เฟส

2.3.1 ไมโครคอนโทรลเลอร์

เป็นหัวใจของกล่องอินเตอร์เฟส โดยในกล่องอินเตอร์เฟสนี้จะใช้ไมโครคอนโทรลเลอร์สองตัวโดยที่ตัวแรกจะทำหน้าที่รับข้อมูลการกดจากคีย์บอร์ดและแปลงข้อมูลที่เป็นสแกนโค้ดเป็นรหัสแอสกีที่ต้องการ ส่วนตัวที่สองจะทำหน้าที่ควบคุมการทำงานของกล่องทั้งหมดและแปลงข้อมูลที่ได้รับมาจากไมโครคอนโทรลเลอร์ตัวแรกมาแปลงเป็นรูปแบบของการส่งข้อความ(SMS Format)การที่ต้องแยกไมโครคอนโทรลเลอร์ออกเป็นสองตัวนี้เนื่องมาจากบอดเรต(Baud rate)ของอุปกรณ์ที่นำมาใช้คือคีย์บอร์ดของคอมพิวเตอร์และ โทรศัพท์มือถือ ไม่ตรงกันทำให้อุปกรณ์ทั้งสองสื่อสารกันไม่ได้

2.3.2 หน่วยความจำ

ต้องมีการต่อหน่วยความจำภายนอกด้วยเพราะว่าหน่วยความจำที่มีมาให้ในไมโครคอนโทรลเลอร์นั้นไม่เพียงพอต่อการใช้ เพราะว่าไมโครคอนโทรลเลอร์ตัวหลักใช้ภาษาเบสิกเขียน(เพราะสามารถควบคุมอุปกรณ์ภายนอกได้ง่าย)ทำให้ใช้หน่วยความจำค่อนข้างมากโดยหน่วยความจำที่ใช้ในกล่องอินเตอร์เฟสนี้ใช้หน่วยความจำขนาด 32 กิโลไบต์

2.3.3 ส่วนของอุปกรณ์เชื่อมต่อ

เป็นส่วนที่กล่องอินเตอร์เฟสเอาไว้เชื่อมต่อกับอุปกรณ์ภายนอก คือ พอร์ตเอาต์พุต RS-232 สองพอร์ต มีไว้เพื่อต่อกับปลายด้านหนึ่งของสายคาต้าลิงค์และนำไปต่อกับโทรศัพท์มือถือ โดยพอร์ต RS-232 นี้จะใช้ทั้งหมดสองพอร์ตมีทั้งตัวผู้และตัวเมีย โดยต่อเข้ากับ โทรศัพท์มือถือหนึ่งพอร์ตและต่อกับคอมพิวเตอร์อีกหนึ่งพอร์ต และพอร์ตอินพุตที่เอาไว้ต่อกับอุปกรณ์ภายนอกคือพอร์ตพีเอสยูซึ่งเอาไว้ต่อกับคีย์บอร์ดของคอมพิวเตอร์ ตัวสุดท้ายที่ต่อกับกล่องอินเตอร์เฟสคือ พอร์ตที่เอาไว้ต่อกับแหล่งจ่ายไฟเพื่อนำไปเลี้ยงอุปกรณ์ในกล่องและอุปกรณ์ที่ต่อพ่วงภายนอกด้วย

2.3.4 ซอฟต์แวร์

ซอฟต์แวร์เป็นส่วนประกอบหนึ่งของกล่องอินเตอร์เฟส ซึ่งเป็นฟังก์ชันเพื่อควบคุมการทำงานต่างๆ ของระบบ โดยซอฟต์แวร์ในกล่องอินเตอร์เฟสนี้จะแบ่งออกเป็นสองโปรแกรม โปรแกรมแรกเขียนไว้บนไมโครคอนโทรลเลอร์ตัวแรกซึ่งเขียนจากภาษาซีเพื่อแปลงสแกนโค้ดเป็นรหัสแอสกี ที่โปรแกรมนี้เขียนจากภาษาซีเพราะว่าภาษาซีสามารถกำหนดระดับบิตได้ง่ายกว่าภาษาอื่น ส่วนโปรแกรมอีกตัวหนึ่งเขียนไว้บนไมโครคอนโทรลเลอร์ตัวหลักโดยตัวนี้เขียนจากภาษาเบสิกเพราะสามารถควบคุมอุปกรณ์ภายนอกได้ง่าย เพราะฉะนั้นซอฟต์แวร์ถือเป็นหัวใจหลักของกล่องอินเตอร์เฟสนี้เลยก็ได้

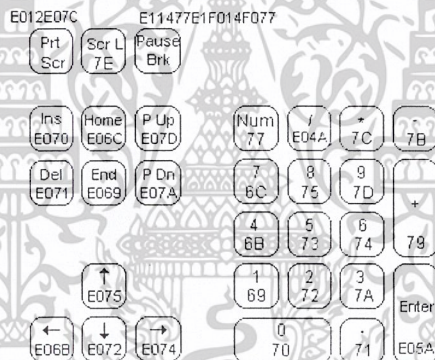
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 หลักการส่งข้อมูลของคีย์บอร์ด(Keyboard)

การส่งข้อมูลของคีย์บอร์ด(keyboard)นั้นจะทำการส่งข้อมูลแบบ I²C โดยหลักการส่งข้อมูลของระบบ บัสแบบ I²C สามารถดูได้จากหัวข้อถัดไป เมื่อทำการกดแป้นพิมพ์ของคีย์บอร์ดสแกนโค้ด(scan code table) ในไบออส(bios) ที่อยู่ภายในตัวของคีย์บอร์ดเองจะทำการแปลงตัวอักษรเป็นสแกนโค้ด(scan code) ตามรูปข้างล่างนี้สิ่งที่ส่งออกมาจากคีย์บอร์ดจะไม่ใช่ตัวอักษรแต่จะเป็นตัวสแกนโค้ด(scan code)เอง เช่น กด A คีย์บอร์ดจะไม่ส่ง A ออกไป แต่จะส่ง 1C (scan code) ออกไปทางพอร์ตอนุกรม(serial port) เป็นต้น



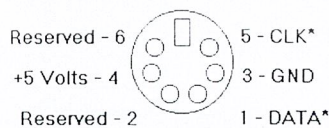
รูปที่ 2.2 สแกนโค้ดของคีย์บอร์ดของบริษัท IBM



รูปที่ 2.3 สแกนโค้ดของคีย์บอร์ดของบริษัท IBM

การส่งข้อมูลออกจากคีย์บอร์ดจะส่งออกทางพอร์ตที่ต่ออยู่กับตัวของคีย์บอร์ดเอง ข้างล่างนี้จะแสดงรูปพอร์ตของคีย์บอร์ดที่เอสทูแบบ 6 พิน เพราะปัจจุบันพอร์ตแบบนี้ถูกใช้งานมากที่สุด

Keyboard Port
6 pin female mini-DIN



รูปที่ 2.4 เป็นพอร์ตของคีย์บอร์ดที่เอสทูแบบ 6 พิน

จากรูปที่ 2.4 ด้านบน เป็นพอร์ตที่เอสทูแบบ 6 พิน โดยที่พินที่ 1 จะเป็นข้อมูล(data) เอาไว้ส่งข้อมูลออกมาจากตัวของคีย์บอร์ดเอง ส่วนขาที่ 3 จะเป็นขากาวของไฟ 5 โวลท์ ขาที่ 4 เป็นไฟบวก 5 โวลท์ ขาที่ 5 เป็นขาของสัญญาณนาฬิกา(clock) ส่วนขาที่ 2 และขาที่ 6 ไม่ได้ถูกใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

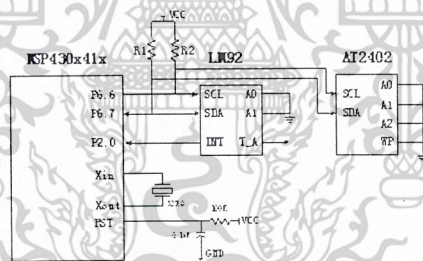
2.5 การเชื่อมต่ออุปกรณ์แบบ I2C

ในระบบไมโครคอนโทรลเลอร์บางครั้งต้องใช้พอร์ตจำนวนมากซึ่งพอร์ตในไมโครคอนโทรลเลอร์เองมีจำนวนจำกัด ถ้าหากต้องการขยายพอร์ตภายนอกหรือว่าใช้อุปกรณ์ต่างๆ ภายนอกเพิ่มเติมจะต้องนำชิปไอซีมาต่อกับไมโครคอนโทรลเลอร์ เช่น ใช้ 74LS244 เป็นพอร์ตอินพุต ใช้ 74LS373 เป็นพอร์ตเอาต์พุต หรือใช้ ชิป 8255 ซึ่งเป็นพอร์ตขนานที่สามารถโปรแกรมได้มาต่อ แต่การต่ออุปกรณ์ต่างๆจะต้องใช้สายสัญญาณของไมโครคอนโทรลเลอร์จำนวนมาก จึงมีการออกแบบระบบบัสแบบใหม่ที่ใช้สายสัญญาณในการเชื่อมต่ออย่างลง

2.5.1 ระบบบัสแบบ I2C

ระบบบัสแบบ I²C ย่อมาจาก Inter-IC Communication ซึ่งพัฒนาโดยห้องวิจัยเอง Philips ในประเทศ Netherlands เมื่อปี ค.ศ. 1980 สำหรับเชื่อมต่อชิปต่างๆกับระบบไมโครคอนโทรลเลอร์แบบอนุกรมโดยใช้สายสัญญาณเพียงสองเส้น เส้นหนึ่งเป็นสายข้อมูล อีกเส้นหนึ่งเป็นสายสัญญาณนาฬิกาสำหรับกำหนดจังหวะการสื่อสารข้อมูล ปัจจุบันได้มีชิปที่สนับสนุนการทำงานของไมโครคอนโทรลเลอร์หลายตัวที่ใช้การเชื่อมต่อระบบบัสแบบนี้

สายข้อมูลที่ใช้รับส่งข้อมูลแบบอนุกรมมีชื่อว่า Serial Data Line (SDA) ส่วนสายสัญญาณนาฬิกาควบคุมมีชื่อว่า Serial Clock Line (SCL) สายทั้งสองนี้จะรับส่งข้อมูลได้สองทิศทาง วงจรทางเอาต์พุตของอุปกรณ์ที่ใช้บัสแบบนี้จะเป็นแบบวงจรเปิด (Open-drain) หรือคอลเล็กเตอร์เปิด (Open-collector) ดังนั้นการต่อบัสแบบนี้จะต้องมีตัวต้านทานต่อพูลอัพกับแรงดันไฟ +5 โวลต์ ดังแสดงรูป



รูปที่ 2.5 การต่อระบบบัสแบบ I²C

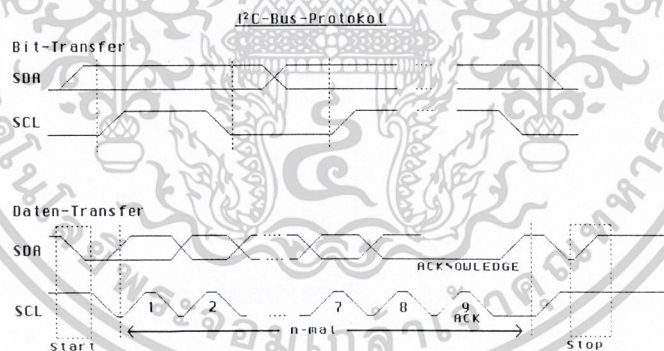
ระบบบัสแบบนี้สามารถรับส่งข้อมูลได้สองทิศทางด้วยความเร็วสูงถึง 100 กิโลบิตต่อวินาที และสามารถใช้กับไอซีที่มีแรงดันไฟฟ้าต่างกันได้ ระบบบัสแบบนี้สามารถเชื่อมต่อกับอุปกรณ์ได้หลายตัวโดยที่อุปกรณ์แต่ละตัวที่ต่ออยู่กับระบบบัสแบบนี้สามารถส่งข้อมูลถึงกันได้ โดยใช้รูปแบบการรับส่งข้อมูลหรือโปรโตคอล (Protocol) ที่อุปกรณ์ทุกตัวรู้จัก อุปกรณ์แต่ละชนิดจะมีค่าแอดเดรสประจำตัวมัน ถ้าหากต้องการให้อุปกรณ์ตัวใดรับข้อมูล ตัวส่งจะส่งแอดเดรสของอุปกรณ์ตัวนั้นออกไป ถ้าหากอุปกรณ์ตัวใดมีแอดเดรสตรงกันก็จะรับข้อมูลนั้นไป สำหรับอุปกรณ์ที่ต้องการส่งข้อมูลอุปกรณ์ตัวนั้นจะเรียกว่ามาสเตอร์ (Master) โดยจะเป็นตัวที่สร้างจังหวะสัญญาณต่างๆ บนระบบบัส ส่วนอุปกรณ์ที่ถูกควบคุมหรือเป็นตัวรับข้อมูลจะเรียกว่าสเลฟ (Slave)

ถ้าหากการเชื่อมต่อแบบไมโครคอนโทรลเลอร์กับอุปกรณ์ที่ใช้ระบบบัสแบบ I²C หลายตัว ถ้าไมโครคอนโทรลเลอร์ต้องการส่งข้อมูล ตัวมันจะทำหน้าที่เป็นมาสเตอร์ และจะส่งสัญญาณไปบนบัส โดยส่งค่าแอดเดรสออกไปก่อน ถ้าค่าแอดเดรสตรงกับแอดเดรสของชิปตัวใดชิปตัวนั้นก็ส่งสัญญาณตอบรับ แยกสารนี้เป็นเอกสารที่ส่งมอบให้สำหรับงานเพื่อการศึกษาเท่านั้น เมื่อนำมาใช้ในเชิงพาณิชย์ กรุณาติดต่อผู้จำหน่าย
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ACK) ออกมา และชิปตัวนั้นก็จะถูกไมโครคอนโทรลเลอร์ควบคุม แต่ถ้าแอดเดรสไม่ตรงกับชิปตัวใดก็ จะไม่เกิดอะไรขึ้น

การส่งข้อมูลต่างๆ จะเกิดขึ้นเมื่อระบบบัสว่างเท่านั้น โดยทั่วไปแล้วสถานะที่มีบนระบบบัสแบบนี้จะมีอยู่ 5 สถานะ ดังแสดงในรูป

1. บัสว่าง (bus not busy) สถานะนี้ค่าลอจิกบนสาย SDA และ SCL จะเป็นลอจิกสูงทั้งคู่
2. เริ่มส่งข้อมูล (start data transfer) สถานะนี้สาย SCL จะเป็นลอจิกสูง แต่สาย SDA เปลี่ยนจากลอจิกสูงเป็นลอจิกต่ำ เรียกว่าสถานะเริ่มต้น (start)
3. สถานะหยุด (stop) สถานะนี้สาย SCL จะเป็นลอจิกสูง แต่สาย SDA จะเปลี่ยนจากลอจิกต่ำไปเป็นลอจิกสูง
4. สถานะมีข้อมูล (data valid) สถานะนี้จะอยู่ระหว่างสถานะเริ่มต้นและสถานะหยุด โดยการรับส่งข้อมูลต่างๆ จะเกิดขึ้นในสถานะนี้ การรับส่งข้อมูลแต่ละบิตจะใช้สัญญาณนาฬิกาหนึ่งลูก โดยข้อมูลบน SDA จะต้องคงที่ ขณะที่ SCL เป็นลอจิกสูง และบิตข้อมูล SDA จะเปลี่ยนแปลงได้ขณะที่ SCL เป็นลอจิกต่ำ ถ้าหากบิตบน SDA มีการเปลี่ยนแปลงขณะที่ SCL เป็นลอจิกสูง ระบบจะตีความว่าเป็นสถานะเริ่มต้นส่งข้อมูล หรือสถานะหยุดแทน
5. สถานะตอบรับ (acknowledge) เมื่ออุปกรณ์มาสเตอร์ส่งข้อมูลออกมาครบหนึ่งไบต์แล้วในช่วงสัญญาณ SCL ลูกที่ 9 มาสเตอร์จะส่งข้อมูลลอจิกสูงออกมา และถ้าตัวรับได้รับข้อมูลครบแล้วมันจะส่งสัญญาณตอบรับ ACK โดยทำให้ระดับลอจิกสูงบนสัญญาณ SDA ให้กลับเป็นระดับลอจิกต่ำ แต่ถ้าตัวรับได้รับข้อมูลไม่ถูกต้องตัวรับจะบังคับให้ตัวส่งหยุดในสถานะรอ



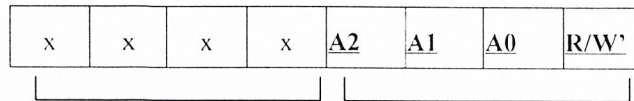
รูปที่ 2.6 แสดงสถานะของระบบบัสแบบ I²C

ในการถ่ายข้อมูลระหว่างสถานะเริ่มต้นและสถานะหยุดสามารถโอนถ่ายข้อมูลได้ไม่จำกัดแต่เมื่อสเลฟได้รับข้อมูลแต่ละไบต์แล้ว มันจะส่งสัญญาณกลับมาทุกครั้ง

การเชื่อมต่ออุปกรณ์ต่างๆ ที่ใช้ในระบบบัสแบบ I²C นั้นเราสามารถใช้อุปกรณ์หลายๆตัวมาต่อกับไมโครคอนโทรลเลอร์ได้ แต่ใช้สายทั้งหมดเพียงสองเส้นเท่านั้น ดังนั้นการติดต่อกับอุปกรณ์แต่ละตัวจะต้องระบุด้วยว่าจะต้องการติดต่อกับอุปกรณ์ตัวใด ขั้นตอนการเขียนข้อมูลที่ใช้กับระบบบัสแบบ I²C มีสามขั้นตอนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เขียนข้อมูลอ้างแอดเดรส (addressing) ข้อมูลไบต์แรกที่มาสเตอร์จะส่งออกไปคือข้อมูลที่ใช้อ้างแอดเดรสของอุปกรณ์ที่ต้องการติดต่อ โดยทั่วไปแล้วข้อมูลที่กำหนดแอดเดรสจะมีจำนวน 7 บิต ดังรูป



บิตแอดเดรสคงที่

บิตแอดเดรสโปรแกรมได้

รูปที่ 2.7 ไบต์แอดเดรส

จากรูปจะเห็นได้ว่าบิตต่ำสุดจะเป็นตัวระบุว่าอุปกรณ์ที่ต้องการติดต่อนั้นเราจะใช้อ่านหรือว่าเขียนข้อมูล ถ้าบิตนี้เป็นลอจิก “0” หมายความว่าเราจะใช้เขียนข้อมูลลงไป ถ้าเป็นลอจิก “1” หมายความว่าเราจะใช้อ่านข้อมูล ส่วน 7 บิตบนจะเป็นบิตแอดเดรส ซึ่งแบ่งออกเป็นบิตแอดเดรสคงที่ (fixed address bit) จำนวน 4 บิตบน ซึ่งจะถูกรหัสโปรแกรมมาจากโรงงานที่ผลิตชิปแต่ละประเภท ส่วนอีก 3 บิตต่อมาจะเป็นบิตแอดเดรสที่ผู้ใช้สามารถโปรแกรมได้ (programmable address bit)

2. การเขียนควบคุม (control byte) ข้อมูลไบต์นี้จะขึ้นอยู่กับอุปกรณ์แต่ละประเภท เพื่อกำหนดการทำงานต่างๆ ของตัวมัน อุปกรณ์บางประเภทอาจไม่ต้องมีการเขียนไบต์นี้ก็ได้

3. การเขียนไบต์ข้อมูล (data byte) เป็นข้อมูลที่ถ่ายโอนในระบบระหว่างสถานะเริ่มต้นและสถานะหยุดจะมีการเขียนข้อมูลจำนวนกี่ไบต์ก็ได้

2.5.2 การเขียนโปรแกรมควบคุมบัส I2C

จากข้างต้นจะเห็นได้ว่าการโอนถ่ายข้อมูลจะเกิดขึ้นได้เมื่อระบบบัสว่างเท่านั้น การอ่านและการเขียนข้อมูลจะต้องสร้างสถานะต่างๆ ขึ้นมาด้วยตั้งโต๊ะแอมป์ขึ้น โดยการใช้โปรแกรมสำหรับสถานะต่างๆ สามารถทำได้ดังนี้

1. การเริ่มส่งข้อมูล (start) โดยการเขียนโปรแกรมในสถานะเริ่มต้นสามารถทำได้ดังนี้

1.1 ทำให้บัสว่าง โดยให้ขา SCL และ SDA เป็นลอจิก “1”

1.2 ทำให้ SDA เป็นลอจิก “0” ขณะที่ SCL เป็นลอจิก “1”

1.3 ทำให้ SCL เป็นลอจิก “0” และเวลาต่อมา SCL และ SDA เป็นลอจิก “0” ทั้งคู่

1.4 ทำให้ SDA เป็นลอจิก “1” เพื่อเริ่มสถานะต่อไป

2. การสร้างสถานะหยุด ใช้ในการหยุดรับส่งข้อมูล หลังจากสถานะนี้จะทำให้บัสอยู่ในสถานะว่าง โดยการสร้างสถานะหยุดสามารถทำได้ด้วยขั้นตอนดังนี้

2.1 ให้ขา SCL และ SDA เป็นลอจิก “0” ทั้งคู่

2.2 ให้ขา SCL เป็นลอจิก “1”

2.3 ให้ขา SDA เป็นลอจิก “1”

3. การส่งข้อมูล ในการรับส่งข้อมูลนั้นขา SCL จะเป็นสัญญาณนาฬิกาควบคุม ระหว่างที่ SCL เป็นลอจิก “1” ข้อมูลบน SDA จะต้องไม่เปลี่ยนแปลง ในการส่งข้อมูลจะต้องทำการส่งครั้งละบิต โดยส่งบิตสูงสุดออกไปก่อน หลังจากส่งครบหนึ่งไบต์จะต้องรอสัญญาณตอบรับโดยส่งลอจิก “1” ออกไปทางขา SDA และทำให้ SCL เป็นลอจิก “0” ถ้าการส่งข้อมูลถูกต้อง ตัวสเลฟจะส่งสัญญาณตอบรับออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

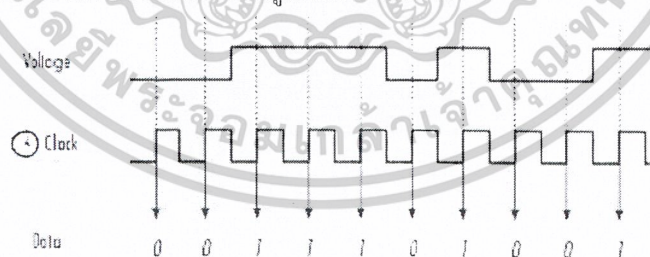
2.6 พอร์ตอนุกรม(Serial Port)

มีทางเลือกอยู่สองแนวทางในการที่จะเคลื่อนย้ายข้อมูลจากคอมพิวเตอร์ไปยังอุปกรณ์ต่อพ่วงอื่นๆ หรือคอมพิวเตอร์ด้วยกัน นั่นคือการรับส่งข้อมูลแบบขนานและการรับส่งข้อมูลแบบอนุกรม การรับส่งข้อมูลแบบขนาน จะเป็นการรับส่งข้อมูลคราวละสี่หรือแปดบิตในเวลาเดียวกัน ซึ่งจะทำให้การรับส่งข้อมูลทำได้ด้วยความเร็วสูง ซึ่งก็หมายความว่าจำนวนสายก็ต้องมีมากเท่ากับจำนวนบิตของข้อมูลที่ส่งด้วย นอกจากนี้ยังต้องรวมถึงสายที่ใช้สำหรับการควบคุมและการตรวจสอบการรับส่งข้อมูลด้วย ซึ่งอาจต้องใช้สายมากเป็นสองเท่าของบิตของข้อมูลที่ส่งก็ได้ ซึ่งเป็นปัญหาในเรื่องราคาของสายที่ใช้ในการเชื่อมต่อแบบขนานมักจะมีราคาแพง

ในขณะที่การรับส่งข้อมูลแบบอนุกรมเป็นการรับส่งข้อมูลครั้งละหนึ่งบิต "ดักก็สามารถรับส่งข้อมูลครั้งละหลายๆบิตได้ หากแต่จะต้องมีข้อดกคลงกันระหว่างตัวส่งและตัวรับว่า จะรับส่งข้อมูลคราวละกี่บิต ตัวรับจะต้องรอรับข้อมูลจนครบทุกบิตก่อนที่จะประมวลผล ส่งผลให้การสื่อสารข้อมูลแบบอนุกรมอาจมีความเร็วต่ำกว่าแบบขนาน ในเรื่องของสายสัญญาณการรับส่งข้อมูลแบบอนุกรมจะใช้จำนวนสายที่น้อยกว่ามาก อย่างน้อยสุดจะใช้เพียงสองถึงสามเส้นเท่านั้น แต่อัตราเร็วในการรับส่งข้อมูลอาจต่ำกว่าแบบขนาน อย่างไรก็ตามการรับส่งข้อมูลแบบอนุกรมสามารถใช้สายสัญญาณที่มีความยาวกว่าแบบขนานมาก ทำให้ระยะในการสื่อสารข้อมูลแบบอนุกรมสามารถทำได้ไกลกว่า

2.6.1 การสื่อสารแบบซิงโครนัส

การสื่อสารแบบอนุกรมนั้นจะแบ่งออกเป็นสองแบบ คือการสื่อสารอนุกรมแบบซิงโครนัสและการสื่อสารอนุกรมแบบอะซิงโครนัส การสื่อสารแบบซิงโครนัสจะมีสัญญาณนาฬิกาที่รวมอยู่กับการรับและส่งสัญญาณด้วย ตัวอย่างการส่งข้อมูลแบบซิงโครนัสคือเคียบอร์ดของคอมพิวเตอร์ ซึ่งสายเส้นหนึ่งเป็นสายของสัญญาณนาฬิกา ส่วนสายอีกเส้นหนึ่งจะเป็นสายของข้อมูล ดังนั้นการติดต่อสื่อสารแบบซิงโครนัสนี้จะต้องใช้สายในการเชื่อมต่ออย่างน้อยที่สุดสามเส้น คือ สัญญาณนาฬิกา ข้อมูลและกราวด์ จากรูปข้างล่างจะแสดงให้เห็นไทมิงไจอะแกรมของการส่งข้อมูลแบบซิงโครนัส



รูปที่ 2.8 แสดงไจอะแกรมการส่งข้อมูลอนุกรมแบบซิงโครนัส

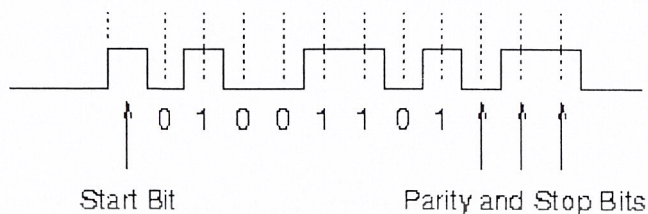
2.6.2 การสื่อสารแบบอะซิงโครนัส

การสื่อสารแบบอะซิงโครนัสคือการรับส่งข้อมูลโดยไม่จำเป็นต้องมีสัญญาณนาฬิกาที่รวมด้วยเหมือนการรับส่งข้อมูลแบบซิงโครนัส แต่จะใช้การกำหนดค่าสัญญาณนาฬิกาทั้งภาครับและภาคส่งให้มีค่าเท่ากัน ซึ่งเรียกเรียกสัญญาณนาฬิกาที่ใช้ในการกำหนดค่าให้ภาครับและภาคส่งนี้ว่า อัตราการถ่ายทอดข้อมูล หรือ บอดเรต(baud rate)มีหน่วยเป็นบิตต่อวินาที (bit per second : bps)

รูปแบบของข้อมูลที่ใช้ในการรับส่งข้อมูลแบบอะซิงโครนัสประกอบด้วยสี่ส่วน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. บิตเริ่มต้น (start bit) ซึ่งมีขนาดหนึ่งบิต
2. บิตข้อมูลแบบอนุกรมจะมีขนาด 5,6,7 หรือ 9 บิต
3. บิตตรวจสอบพาริตี (parity bit) จะมีขนาดหนึ่งบิต หรือจะไม่มีก็ได้
4. บิตปิดท้าย (stop bit) จะมีขนาด 1,1.5 หรือ 2 บิต



รูปที่ 2.9 แสดงไดอะแกรมการส่งข้อมูลอนุกรมแบบอะซิงโครนัส

จากรูปที่ 2.9 แสดงรูปของข้อมูลอนุกรมแบบอะซิงโครนัส ซึ่งเมื่อไม่มีข้อมูลที่จะส่ง ขาดำ (DATA) มีสถานะเป็น “1” ซึ่งเรียกสถานะนี้ว่าสถานะหยุดรอ (waiting stage) การเริ่มต้นของข้อมูลจะเริ่มจากการให้ขาดำ (DATA) มีลอจิกเป็น “0” ด้วยช่วงระยะเวลาหนึ่งบิต ซึ่งจะเรียกบิตนี้ว่าบิตเริ่มต้นจากนั้นบิตข้อมูลจะถูกส่งออกไป โดยเริ่มจากบิตที่มีนัยต่ำสุด (LSB) ก่อน ซึ่งข้อมูลในไบต์ที่จะส่งอาจมีจำนวนบิตเป็น 5,6,7 หรือ 8 บิตก็ได้ จากนั้นจะตามด้วยบิตพาริตี ซึ่งใช้เพื่อตรวจสอบความผิดพลาดที่เกิดขึ้นจากการส่งข้อมูล บิตสุดท้ายที่จะส่งคือบิตปิดท้าย ซึ่งจะทำให้ขาดำมีสถานะลอจิกเป็น “1” อีกครั้งด้วยระยะเวลาอย่างน้อย 1 บิต, 1.5 บิต หรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว

อุปกรณ์พิเศษที่ได้รับการออกแบบมาสำหรับการรับและส่งข้อมูลแบบอะซิงโครนัสเรียกว่า Universal Asynchronous Receiver / Transmitter หรือ UART อัตราความเร็วในการรับส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัสคือ ค่าบอดเรต ซึ่งก็คือค่าจำนวนบิตต่อวินาทีในการรับส่งข้อมูล บอดเรตมาตรฐานที่ใช้สำหรับพอร์ตอนุกรม RS-232 ได้แก่ 110,150,300,600,1200,2400,9600 และ 19200 บิตต่อวินาที และมีค่าเพิ่มมากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ ซึ่งการรับส่งข้อมูลโดยไม่ผ่านโมเด็มอาจจะกำหนดค่าบอดเรตได้สูงสุดถึง 115200 บิตต่อวินาที เนื่องจากบอดเรตคือจำนวนบิตข้อมูลที่สามารถถ่ายทอดได้ภายในระยะเวลาหนึ่งวินาที ยกตัวอย่างเช่น ข้อมูลอนุกรมถูกส่งในลักษณะของแปดบิต ไม่มีการตรวจสอบพาริตี มีบิตเริ่มต้นหนึ่งบิต และบิตปิดท้ายหนึ่งบิต ความยาวของข้อมูลที่ได้รับส่งนี้เท่ากับสิบบิต ถ้าใช้บอดเรตในการส่งเท่ากับ 9600 บิตต่อวินาที ก็จะสามารถรับส่งข้อมูลด้วยความเร็ว 960 ไบต์ต่อวินาที และถ้ามีการใช้พาริตีความเร็วในการรับส่งข้อมูลจะเหลือ 872 ไบต์ต่อวินาที

2.6.3 มาตรฐานพอร์ตอนุกรมแบบ RS-232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการรับส่งข้อมูลอนุกรมแบบอะซิงโครนัสสองทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มสื่อสารผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดหนึ่งซึ่งอยู่ห่างไกลกันออกไป โดยคณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB-25 โดยกำหนดความยาวสูงสุดของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สายสัญญาณไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3 V ถึง -12 V แสดงว่ามีข้อมูล(Mark) และ $+3$ V ถึง $+12$ V แสดงว่าเป็นช่องว่างหรือไม่มีข้อมูล(Space)

มาตรฐาน RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล(Data Terminal Equipment : DTE) กับปลายทาง(Data Circuit Terminating : DCE)ไว้ว่า อุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรลเลอร์หรือไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างบิตข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้งสองจะทำผ่านมาตรฐาน RS-232

ข้อแตกต่างของ DTE และ DCE อย่างหนึ่งที่ได้สังเกตเห็นคือคอนเน็กเตอร์(connector) ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ในส่วนของโมเด็มจะเป็นแบบ DCE

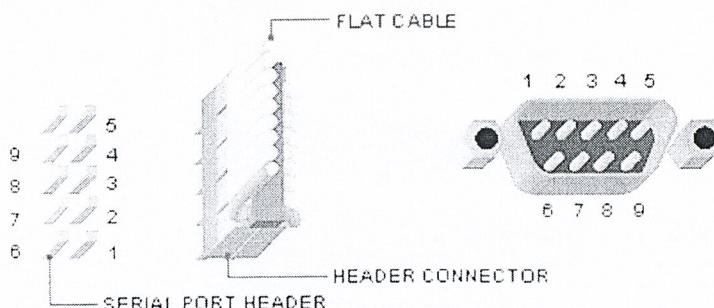
สำหรับการใช้งานบนคอมพิวเตอร์ พอร์ตอนุกรม RS-232 มักจะถูกใช้เชื่อมต่อกับโมเด็มหรือเมาส์ โดยสามารถรับส่งข้อมูลได้ความยาวสายสัญญาณสูงสุดถึง 20 เมตร

2.6.4 คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้คอนเน็กเตอร์แบบ DB-25 ตัวผู้ หรือ DB-9 ตัวผู้ซึ่งคอนเน็กเตอร์แบบ DB-25 จะมีขาต่อใช้งานเพียง 9 เส้นเช่นเดียวกับคอนเน็กเตอร์แบบ DB-9 เนื่องจากขาอื่นๆที่เคยใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนัก จึงถูกยกเลิกไปโดยแสดงรูปภาพและตำแหน่งขาของทั้งสองแบบดังรูปข้างล่างนี้

	PSION (DB9)		Modem (DB25)
DCD	1	—————	22 RI
Rx	2	—————	2 Tx
Tx	3	—————	3 Rx
DTR	4	—————	6 DSR
Gnd	5	—————	7 Gnd
DSR	6	—————	20 DTR
RTS	7	—————	5 CTS
CTS	8	—————	4 RTS
RI	9	—————	8 DCD

รูปที่ 2.10 การจัดขาของคอนเน็กเตอร์ของพอร์ตอนุกรมตามมาตรฐาน RS-232



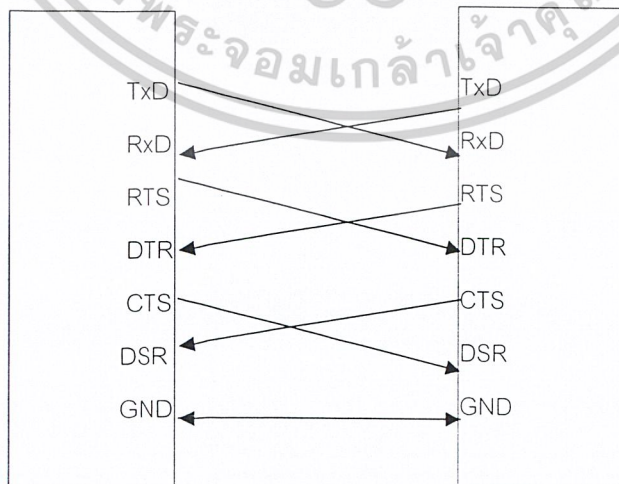
รูปที่ 2.11 แสดงรูปของคอนเน็กเตอร์แบบ DB-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 การจัดขาของ DB-9 และ DB-25

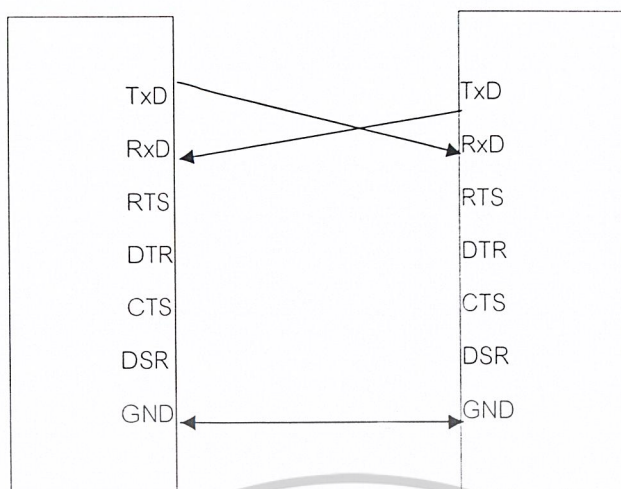
คอนเน็กเตอร์ DB-9	คอนเน็กเตอร์ DB-25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
1	8	DCD	อินพุต
2	3	RxD	อินพุต
3	2	TxD	เอาต์พุต
4	20	DTR	เอาต์พุต
5	7	GND	-
6	6	DSR	อินพุต
7	4	RTS	เอาต์พุต
8	5	CTS	อินพุต
9	22	RI	อินพุต

สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกดังแสดงในรูปข้างล่าง ถูกสรุปในรูปหมายถึงทิศทางของข้อมูล ในรูป (ก) เป็นการเชื่อมต่อแบบไม่มีโมเด็ม(Null Modem)หรือการเชื่อมต่อโดยตรงโดยไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบหรือแฮนด์เร็กเต็มรูปแบบ ส่วนในรูป (ข) เป็นการเชื่อมต่อแบบไม่มีโมเด็ม(Null Modem) เช่นเดียวกัน แต่จะใช้สายสัญญาณเพียง 3 เส้นเท่านั้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นหนึ่งสำหรับรับข้อมูล และสุดท้ายเป็นกราวด์ สำหรับรายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232 มีได้อธิบายไว้หลังจากรูปนี้



รูป (ก) การต่ออุปกรณ์ภายนอกกับคอมพิวเตอร์แบบไม่มีโมเด็ม (Null Modem)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป (ข) การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ RS-232 โดยใช้สัญญาณ 3 เส้น

รูปที่ 2.12 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ

สำหรับรายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232 มีดังนี้

-Data Carrier Detect : DCD หรืออาจเรียกว่า Carrier Detect : CD ขานี้จะแอกติฟเมื่อมีการส่งสัญญาณพหะกับอุปกรณ์สื่อสารข้อมูล เช่น โมเด็ม สำหรับการใช้งานแบบปกติ ขานี้จะไม่ถูกใช้งานมากนัก

-Receive Data : RD หรือ RxD ขานี้ใช้เพื่อรับส่งสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์ โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์ บัฟเฟอร์

-Transmitted Data : TD หรือ TxD ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์ โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป

-Data Terminal Ready : DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่าการติดต่อด้วย โดยขา DTR นี้จะต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง และขา DTR ของอุปกรณ์ปลายทางจะต้องเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกันและต้องต่อขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพหะ

-Signal Ground : GND ขากราวด์ของระบบ

-Data Set Ready : DSR ขานี้จะใช้คู่กับ DTR เพื่อตรวจสอบการเชื่อมโยงต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DSR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขา DTR

-Request To Send : RTS เป็นขาสำหรับส่งสัญญาณร้องขอให้อุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบไม่มีโมเด็ม (Null Modem) 3 สาย จะต้องเชื่อมต่อกับขา RTS และ CTS ของตัวมันเองเข้าด้วยกัน เพื่อให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา

-Clear To Send : CTS ขานี้จะคอยรับสัญญาณเรียกจากโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็มและโปรแกรมมีการตรวจสอบสัญญาณนี้เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.5 Universal Asynchronous Receiver Transmitter : UART

UART ย่อมาจากคำว่า Universal Asynchronous Receiver Transmitter ซึ่งหมายถึงอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสนั่นเอง สำหรับการสื่อสารอนุกรมบนคอมพิวเตอร์แล้ว UART ถือว่าเป็นหัวใจสำคัญของการสื่อสารแบบอนุกรม

หน้าที่หลักของ UART คือทำหน้าที่แปลงข้อมูลที่อยู่ในรูปแบบขนานจากคอมพิวเตอร์ให้อยู่ในรูปแบบอนุกรมแบบอะซิงโครนัสแล้วส่งออกไป และทำหน้าที่แปลงสัญญาณอนุกรมแบบอะซิงโครนัสที่ป้อนเข้ามายัง UART เป็นแบบขนานก่อนที่จะถูกส่งข้อมูลเข้าสู่คอมพิวเตอร์ ซึ่งนอกจาก UART จะส่งไปยังคอมพิวเตอร์แล้ว ยังแจ้งข้อมูลอื่นๆ ให้คอมพิวเตอร์รับทราบด้วย เช่น อัตราเร็วในการรับส่งข้อมูล (บอดเรต) , รูปแบบการส่งข้อมูล , ความผิดพลาดที่เกิดขึ้นระหว่างการถ่ายทอดข้อมูล(ผิดพลาดจากพริตตี, เฟรมข้อมูล , โอเวอร์รัน) เป็นต้น

ภายใน UART จะมีส่วนของวงจรสร้างบอดเรตโปรแกรมได้ (programmable baudrate generator) โดยการกำหนดค่าตัวหารให้กับสัญญาณนาฬิกาของ UART โดยตัวหารนี้มีขนาด 16 บิต ดังนั้นจึงสามารถกำหนดตัวหารให้อยู่ในช่วง 1-65,535 โดย UART สามารถรับส่งข้อมูลได้ทั้งแบบ ฮาล์ฟดูเพล็กซ์(half duplex) และฟูลดูเพล็กซ์(full duplex) โดยการส่งแบบฮาล์ฟดูเพล็กซ์เป็นการส่งแบบทิศทางเดียว ส่วนการส่งแบบฟูลดูเพล็กซ์นั้นสามารถรับและส่งข้อมูลได้ในคราวเดียวกัน

2.6.6 ชนิดของ UART

ในเครื่องคอมพิวเตอร์ทั่วไปจะมี UART ที่ใช้งานกันอยู่สองเบอร์คือ 8250 ซึ่งเป็น UART มาตรฐานที่ใช้งานกันมาช้านาน UART เบอร์นี้จะมีบัฟเฟอร์สำหรับรับส่งข้อมูลในตำแหน่งเดียวกัน ทำให้การรับและการส่งข้อมูลถูกจำกัดความเร็วอยู่ที่ 57.6 กิโลบิตต่อวินาทีเท่านั้น แต่ UART เบอร์นี้ถือว่าเป็นต้นแบบของ UART ที่ใช้ในคอมพิวเตอร์ โดยคอมพิวเตอร์ทุกรุ่นจะต้องสนับสนุนการทำงานตามรูปแบบของ UART เบอร์นี้

UART อีกเบอร์หนึ่งคือ 16450 มีความสามารถรับส่งข้อมูลได้ที่ความเร็ว 115,220 บิตต่อวินาที และเพิ่มรีจิสเตอร์สำหรับพักข้อมูลสำหรับ UART นอกจากนั้นยังเพิ่มส่วนของชิพรีจิสเตอร์แบบ FIFO(Fist In First Out) ขนาด 16 ไบต์เข้าไป ทำให้สามารถสนับสนุนความเร็วในการรับส่งข้อมูลที่ 256 กิโลบิตต่อวินาทีได้ โดยคอมพิวเตอร์ในปัจจุบันใช้ UART เบอร์นี้หรือใหม่กว่า เช่นเบอร์ TL15C750 ซึ่งมีรีจิสเตอร์แบบ FIFO ขนาด 64 ไบต์ ทำงานได้ที่ระดับแรงดัน +5V และ +3V มีโหมคประหยัดพลังงาน สามารถรับส่งข้อมูลที่ความเร็ว 1 เมกะบิตต่อวินาทีเมื่อใช้สัญญาณนาฬิกา 16 Mhz

อย่างไรก็ตาม ความเร็วในการส่งข้อมูลที่มากมายของ UART เบอร์ใหม่ๆก็ไม่ได้ช่วยให้การรับส่งข้อมูลของคอมพิวเตอร์เร็วขึ้น เนื่องจากว่าคอมพิวเตอร์ยังใช้ความถี่ของสัญญาณนาฬิกาแบบในการแปลงข้อมูลเพียง 1.8432 Mhz เท่านั้น

2.7 เอสเอ็มเอส(SMS) ในรูปแบบของพีดียู(PDU)

การส่งข้อมูลของ sms ในรูปแบบของ PDU สามารถส่งได้หลายแบบ ดังต่อไปนี้

1. การส่งแบบ 7 บิต การส่งแบบนี้จะเป็นการส่งแบบพื้นฐาน และการส่งแบบนี้จะสามารถส่งได้สูงสุด 160 ตัวอักษรต่อครั้ง
2. การส่งแบบ 8 บิต การส่งแบบนี้จะสามารถส่งได้สูงสุด 140 ตัวอักษรต่อครั้ง
3. การส่งแบบ 16 บิต การส่งแบบนี้จะสามารถส่งได้สูงสุด 70 ตัวอักษรต่อครั้ง ถ้าเราจะให้ส่งภาษาไทยได้ด้วย การส่งจะต้องเป็นแบบ 16 บิตเท่านั้น

มีสองทางให้เลือกในการส่งข้อความสั้น (short message) คือส่งในรูปแบบของเท็กโทมดและพีดียูโทมด(PDU:Protocol Description unit) โดยใน เท็กโทมดไม่สามารถใช้ได้กับโทรศัพท์มือถือบางเครื่อง การส่งเอสเอ็มเอสในพีดียูโทมด(Sending a message in the PDU mode)

ต่อไปจะเป็นการแสดงตัวอย่างการส่งข้อความ “hellohello” ในรูปแบบของพีดียูโทมดขนาด 7 บิต โดยการส่งนี้ต้องใช้งานร่วมกับเอทีคอมมานด์ (AT COMMAND) ซึ่งจะได้อธิบายต่อจากหัวข้อนี้

```
AT+CMGF = 0 //Set PDU Mode
```

```
AT+CSMS = 0 //Send an SMS
```

```
AT+CMGS = 23 //Send message, 23 octets
```

```
>0011000B916407281553F80000AA0AE8329BFD4697D9EC37<ctrl-z>
```

การที่มีขนาดของข้อมูลที่ถูกส่งออกไปเท่ากับ 23 อ็อกเตท นั้นก็เพราะว่ามีเลขฐานสิบหก 46 ตัว เพราะว่า “00”ของอ็อกเตทแรกนั้นจะไม่ถูกนับรวมอยู่ด้วย เพราะจะเป็นตัวนี้ไปที่ความยาวของข้อความ โดยแต่ละอ็อกเตทจะมีความหมายโดยแต่ละอ็อกเตทจะได้อธิบายไว้ในภาคผนวก

2.8 เอทีคอมมานด์(AT COMMAND)

เอทีคอมมานด์เป็นคำสั่งที่สั่งให้โทรศัพท์มือถือทำงานตามที่เราต้องการ โดยสั่งคำสั่งนี้ผ่านสายดาต้าลิงค์ที่ต่อเข้ากับพอร์ตของมือถือ ต่อไปเป็นการแสดงตัวอย่างของเอทีคอมมานด์ที่ใช้กันบ่อยๆ

AT-COMMAND

คำสั่ง AT-COMMAND ที่จะนำมาใช้งาน

AT+CMGF	เป็นคำสั่งในการเลือกโทมดของ Message ที่จะส่ง
AT+CMGL	เป็นคำสั่งสำหรับอ่าน Message ที่เลือกไว้ จาก SIM card
AT+CMGS	เป็นคำสั่งสำหรับส่ง Message ไปยัง Address ที่เลือกไว้
AT+CMSS	เป็นคำสั่งสำหรับส่ง Message จาก SIM card ไปยัง Address ที่เลือกไว้
AT+CMGW	เป็นคำสั่งสำหรับเขียน Message เก็บไว้ใน SIM card
AT+CMGD	เป็นคำสั่งสำหรับลบ Message ที่เก็บไว้ใน SIM card
AT+CPMS	เป็นคำสั่งสำหรับเลือก Memory ของการอ่านข้อมูล

ฯลฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.1 ลักษณะของชุดคำสั่ง AT+CMGF

Command	Possible responses
+CMGF=[<mode>]	OK
+CMGF?	-CMGF: <mode>
+CMGF=?	-CMGF: (list of supported <mode>s)

<mode>:0 PDU-Mode

1 Text-Mode

2.8.2 ลักษณะของชุดคำสั่ง AT+CMGS กรณีส่งแบบ Text-Mode

Command	Possible responses
+CMGS=<da>[,<todo>]<CR> text is entered<ctrl-Z/ESC>	-CMGS: <mr> +CMS ERROR: <err>
+CMGS=?	OK

<da> เป็นหมายเลขโทรศัพท์ที่เราต้องการส่ง SMS โดยจะอยู่ในรูป
“หมายเลข” ซึ่งอยู่ในรูปรหัส ASCII

<todo> เป็นหมายเลขโทรศัพท์ที่เราต้องการส่ง SMS โดยจะอยู่ในรูปของ“+
รหัสประเทศตามด้วยหมายเลข” ซึ่งอยู่ในรูปตัวเลข

<mr> จำนวนครั้งที่เราส่ง SMS

2.8.3 ลักษณะของชุดคำสั่ง AT+CMGS กรณีส่งแบบ PDU-Mode

Command	Possible responses
+CMGS=<length><CR> PDU is given <ctrl-Z/ESC>	-CMGS: <mr> +CMS ERROR: <err>
+CMGS=?	OK

<length> ความยาวของ TPDU โดยจะนับแบบ octets

<pdu> ข้อความที่เป็นส่วนของ SMSC รวมกับ TPDU

<mr> จำนวนครั้งที่เราส่ง SMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.4 ลักษณะของชุดคำสั่ง AT+CMGD

Command	Possible responses	
+CMGD=<index>	OK	+CMS ERROR: <err>
+CMGD=?	OK	

<index> ตัวบอกตัวตำแหน่งที่เราต้องการเลือกกว่าเป็นข้อความที่เท่าไรใน SIM Card

2.8.5 ลักษณะของชุดคำสั่ง AT+CMGW กรณีส่งแบบ Text-Mode

Command	Possible responses	
+CMGW=<oa/da>[,<tooa,toda>[,<stat>]] <CR>text is entered<ctrl-Z/ESC>	+CMGW: <index>	+CMS ERROR: <err>
+CMGW=?	OK	

<oa> เป็นหมายเลขโทรศัพท์ที่จะบอกว่าส่ง SMS มาจากเบอร์โทรศัพท์ใด โดยจะอยู่ในรูป “หมายเลข” ซึ่งอยู่ในรูปรหัส ASCII

<tooa> เป็นหมายเลขโทรศัพท์ที่จะบอกว่าส่ง SMS มาจากเบอร์โทรศัพท์ใด โดยจะอยู่ในรูปของ “รหัสประเทศตามด้วยหมายเลข” ซึ่งอยู่ในรูปตัวเลข

<da> เป็นหมายเลขโทรศัพท์ที่เราต้องการส่ง SMS โดยจะอยู่ในรูป “หมายเลข” ซึ่งอยู่ในรูปรหัส ASCII

<toda> เป็นหมายเลขโทรศัพท์ที่เราต้องการส่ง SMS โดยจะอยู่ในรูปของ “รหัสประเทศตามด้วยหมายเลข” ซึ่งอยู่ในรูปตัวเลข

<stat> ตัวบอกสถานะของข้อความที่อยู่ใน SIM Card
 "REC UNREAD" คือข้อความนี้ได้รับมาและยังไม่ได้อ่าน
 "REC READ" คือข้อความนี้ได้รับมาและได้อ่านแล้ว
 "STO UNSENT" คือเป็นข้อความที่เขียนขึ้นมาแต่ยังไม่ได้ส่ง
 "STO SENT" คือเป็นข้อความที่เขียนขึ้นมาแล้วได้ส่งไปแล้ว
 "ALL" คือข้อความทุกข้อความ

2.8.6 ลักษณะของชุดคำสั่ง AT+CMSS

Command	Possible responses	
+CMSS=<index>[,<da>[,< toda>]]	+CMSS: <mr>	+CMS ERROR: <err>
+CMSS=?	OK	

<index> ตัวบอกตัวตำแหน่งที่เราต้องการเลือกกว่าเป็นข้อความที่เท่าไรใน SIM Card

<da> เป็นหมายเลขโทรศัพท์ที่เราต้องการส่ง SMS โดยจะอยู่ในรูป “หมายเลข” ซึ่งอยู่ในรูปรหัส ASCII

< toda> เป็นหมายเลขโทรศัพท์ที่เราต้องการส่ง SMS โดยจะอยู่ในรูปของ “รหัสประเทศตามด้วยหมายเลข” ซึ่งอยู่ในรูปตัวเลข

<mr> จำนวนครั้งที่เราส่ง SMS

2.8.7 ลักษณะของชุดคำสั่ง AT+CMGW กรณีส่งแบบ PDU-Mode

Command	Possible responses	
CMGW=<length>[,<stat>]<CR> PDU is given <ctrl-Z/ESC>	+CMGW: <index>	+CMS ERROR: <err>
+CMGW=?	OK	

<length> ความยาวของ TPDU โดยจะนับแบบ octets

<stat> ตัวบอกสถานะของข้อความที่อยู่ใน SIM Card

0 Received unread message คือข้อความนี้ได้รับมาและยังไม่ได้อ่าน

1 Received read message คือข้อความนี้ได้รับมาและได้อ่านแล้ว

2 Stored unsent message คือเป็นข้อความที่เขียนขึ้นมาแต่ยังไม่ได้ส่ง

3 Stored sent message คือเป็นข้อความที่เขียนขึ้นมาแล้วได้ส่งไปแล้ว

5 All messages คือข้อความทุกข้อความ

<pdu> ข้อความที่เป็นส่วนของ SMSC รวมกับ TPDU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.8 ลักษณะของชุดคำสั่ง AT+CPMS

Command	Possible responses	
+CPMS=<mem1>[,<mem2>[,<mem3>]]	+CPMS: <used1>,<total1>,<used2>,<total2>,<used3>,<total3>	+CMS ERROR: <err>
+CPMS?	+CPMS: <mem1>,<used1>,<total1>,<mem2>,<used2>,<total2>,<mem3>,<used3>,<total3>	+CMS ERROR: <err>
+CPMS=?	+CPMS: (list of supported <mem1>s) ,(list of supported <mem2>s) ,(list of supported <mem3>s)	

<mem1> เป็นตัวเลือกว่าจะอ่าน Message และ ลบ Message จาก SIM card หรือ Memory ภายใน

"SM" อ่านและลบข้อความจาก SIM message storage

"ME" อ่านและลบข้อความจาก Memory message storage

<mem2> เป็นตัวเลือกว่าจะเก็บ Message ที่เขียนหรือส่งแล้วใน SIM card หรือ Memory ภายใน

"SM" อ่านและลบข้อความจาก SIM message storage

"ME" อ่านและลบข้อความจาก Memory message storage

<mem3> เป็นตัวเลือกว่าจะเก็บ Message ที่เขียนหรือส่งแล้วใน SIM card หรือ Memory ภายใน

"SM" อ่านและลบข้อความจาก SIM message storage

"ME" อ่านและลบข้อความจาก Memory message storage

<total1> จำนวนข้อความที่เก็บไว้ทั้งหมดใน <mem1>

<total2> จำนวนข้อความที่เก็บไว้ทั้งหมดใน <mem2>

<total3> จำนวนข้อความที่เก็บไว้ทั้งหมดใน <mem3>

<used1> จำนวนข้อความที่เก็บไว้ที่เปิดอ่านแล้วใน <mem1>

<used2> จำนวนข้อความที่เก็บไว้ที่เปิดอ่านแล้วใน <mem2>

<used3> จำนวนข้อความที่เก็บไว้ที่เปิดอ่านแล้วใน <mem3>

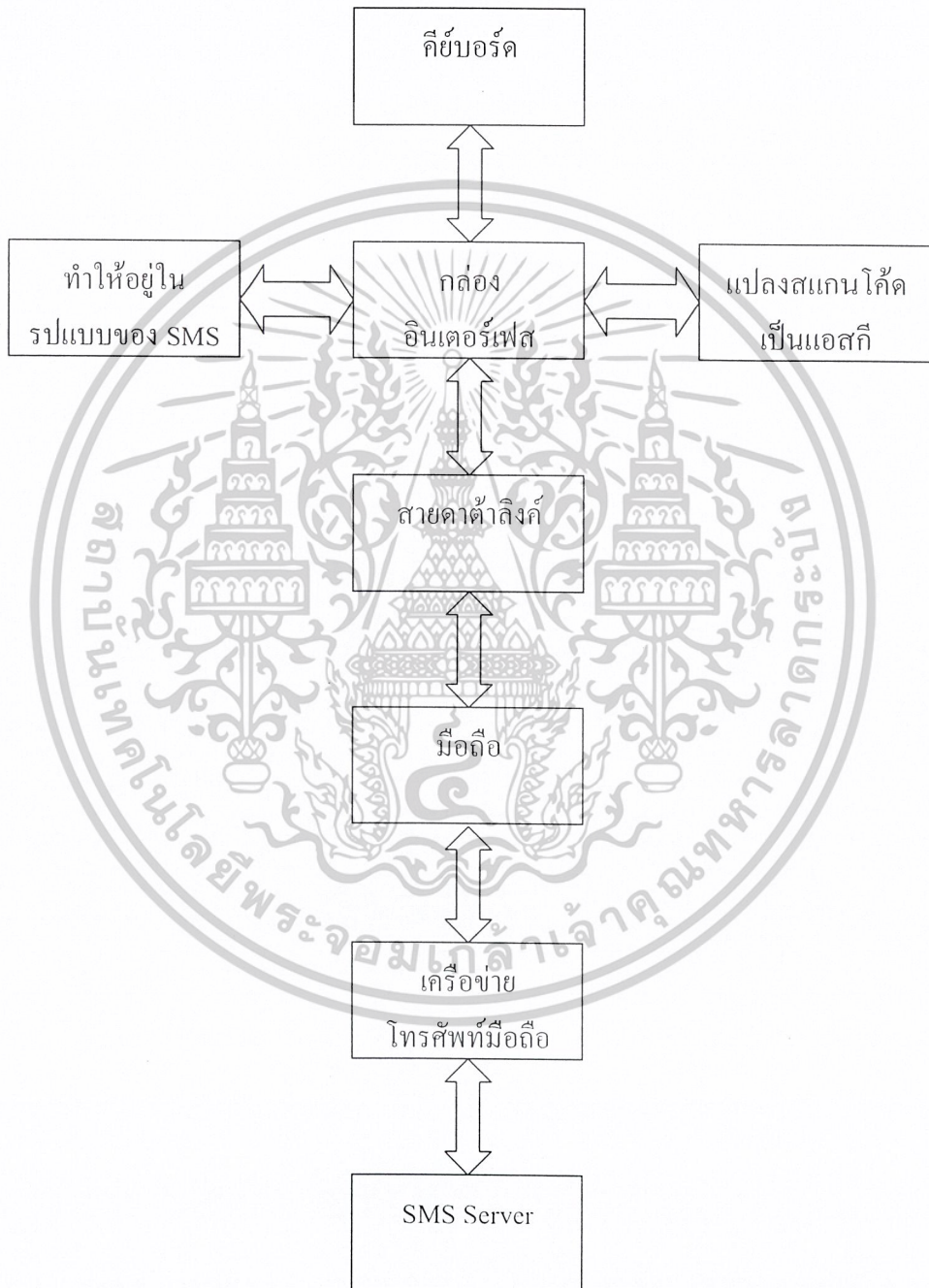
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การวิเคราะห์และการออกแบบระบบ

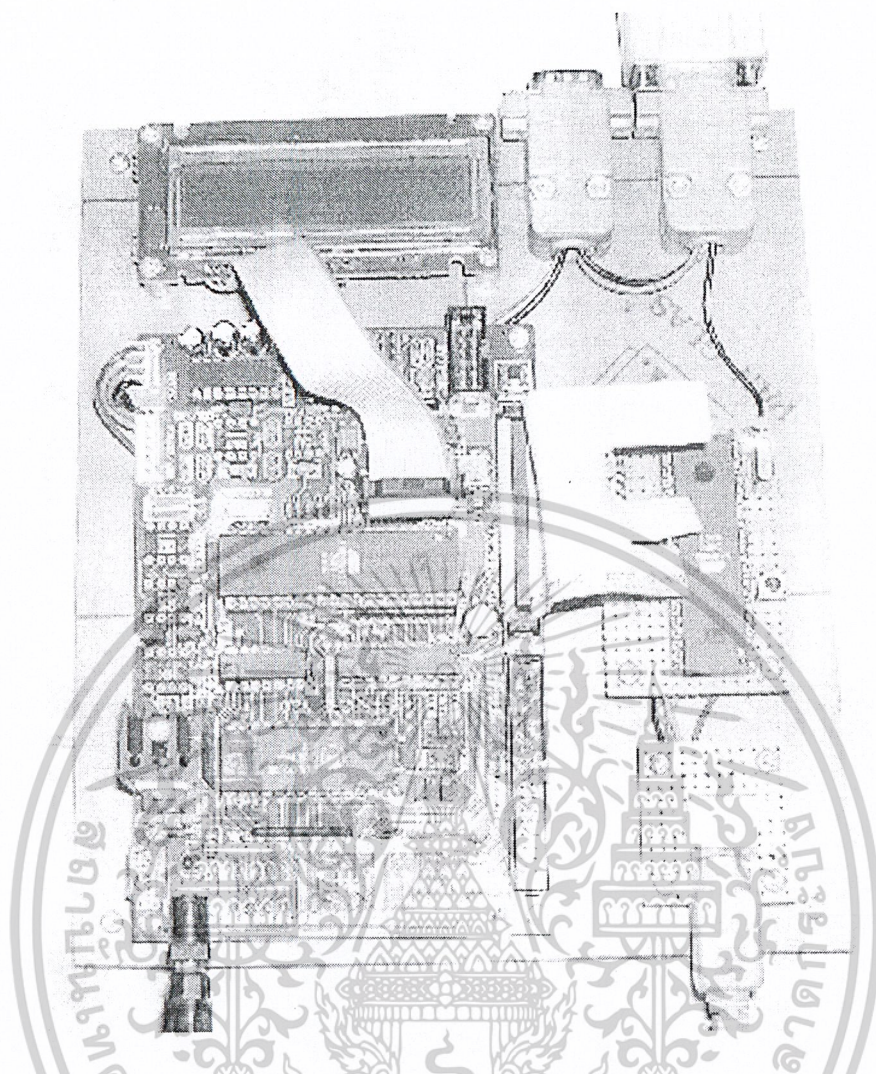
3.1 ภาพรวมของระบบ

ภาพรวมของระบบนี้ตัวกล่องอินเทอร์เน็ตเฟสจะเป็นตัวเชื่อมระหว่างตัวคีย์บอร์ดของคอมพิวเตอร์และโทรศัพท์มือถือ จากภาพที่ 3.1 แสดงให้เห็นถึงภาพโดยรวมของระบบ



รูปที่ 3.1 แสดงการทำงาน โดยรวมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

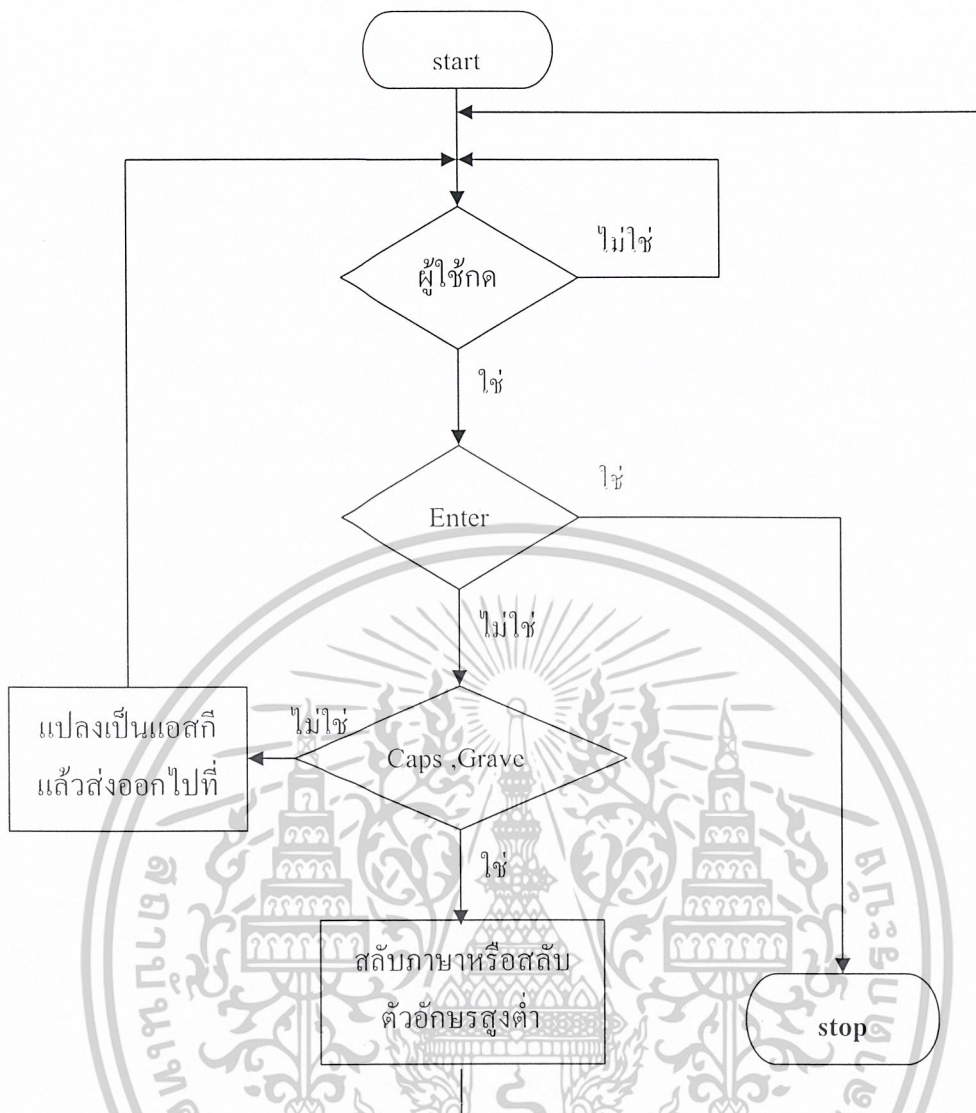


รูปที่ 3.2 แสดงภาพจริงของระบบ โดยรวมทั้งหมด

3.2 การออกแบบส่วนของการรับข้อมูล และแปลงสแกนโค้ดเป็นรหัสแอสกี

ในปฏิญญาฉบับนี้ เป็นการออกแบบกล่องอินเตอร์เฟส เพื่อใช้ในการเชื่อมต่อคีย์บอร์ดของคอมพิวเตอร์และโทรศัพท์มือถือ เพื่อให้ใช้งานได้การพิมพ์ได้ง่ายและรวดเร็วยิ่งขึ้น โดยการทำงานของระบบนี้แบ่งย่อยออกเป็นหลายส่วน ซึ่งสามารถดูได้จากไดอะแกรมด้านบน ซึ่งจากไดอะแกรมจะเห็นการทำงานโดยรวมของระบบดังต่อไปนี้ คือเมื่อผู้ใช้ทำการใช้คีย์บอร์ดของคอมพิวเตอร์พิมพ์ข้อความต่างๆที่ต้องการข้อมูลต่างๆที่เป็น สแกนโค้ดก็จะวิ่งเข้ามาที่กล่องอินเตอร์เฟส เมื่อกล่องอินเตอร์เฟสนี้ได้รับข้อมูลแล้วก็จะเข้าสู่ส่วนที่แปลงสแกน โค้ดให้เป็นรหัสแอสกีที่เป็นมาตรฐานซึ่งส่วนนี้เราจะใช้ไมโครคอนโทรลเลอร์หนึ่งตัว เมื่อได้รหัสแอสกีมาแล้วขั้นตอนนี้ไปคือทำให้ข้อความที่ผู้ใช้พิมพ์มานั้นให้อยู่ในรูปแบบของการส่งข้อมูล(Format) แล้วก็ทำการส่งออกไปที่โทรศัพท์มือถือที่ต่อกับกล่องอินเตอร์เฟสผ่านสายดาต้าลิงค์ เมื่อโทรศัพท์ได้รับรูปแบบการส่งข้อมูลที่ส่งมาจากกล่องอินเตอร์เฟสแล้วมันก็จะส่งไปยังเครือข่ายของโทรศัพท์มือถือ เพื่อติดต่อไปยังหมายเลขของโทรศัพท์มือถือปลายทาง

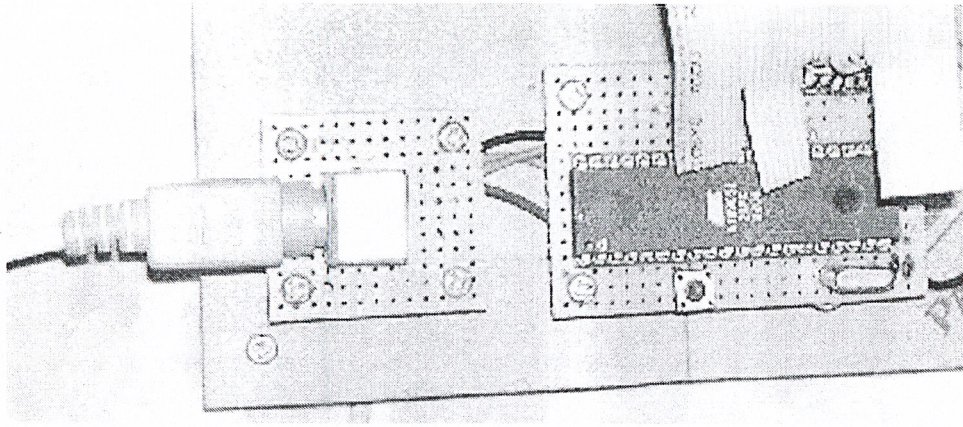
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 Flow chart ในส่วนของการแปลงสแกนโค้ดเป็นรหัสแอสกีหรือส่วนรับข้อมูล

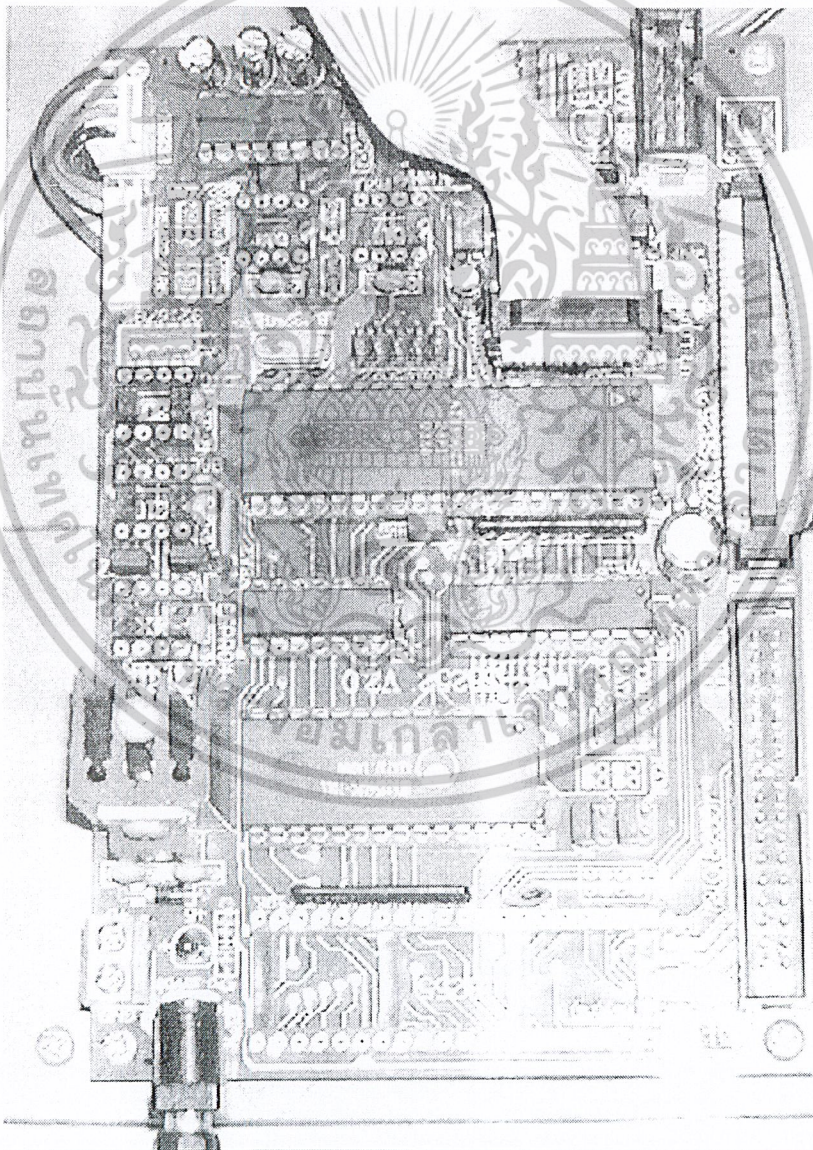
การทำงานของโคอะแกรมด้านบนนั้นเป็นส่วนของการรับข้อมูลมาจากคีย์บอร์ดของคอมพิวเตอร์และทำไปแปลงจากสแกนโค้ดที่ได้มาจากคีย์บอร์ดคอมพิวเตอร์ให้เป็นรหัสแอสกีที่เป็นมาตรฐานในการสื่อสารกัน โดยเราสามารถอธิบายการทำงานของโคอะแกรมได้ดังนี้ คือในภาวะเริ่มต้นจะไม่มีการทำงานอะไร เมื่อมีผู้ใช้คีย์บอร์ดของคอมพิวเตอร์เพื่อพิมพ์ข้อความนั้นคีย์บอร์ดก็จะส่งค่าให้กล่องอินเตอร์เฟซ และเมื่อกล่องอินเตอร์เฟซได้รับข้อความที่เป็นตัวอักษรแล้วก็จะดูสแกนโค้ดว่าตรงกับของปุ่มเอ็นเทอร์หรือไม่ถ้าตรงแสดงว่าสิ้นสุดข้อความแล้วก็ให้หยุดทำซึ่งเป็นการสิ้นสุดของส่วนรับข้อมูลแต่ถ้าไม่ใช่สแกนโค้ดของปุ่มเอ็นเทอร์ กล่องอินเตอร์เฟซก็จะตรวจสอบต่อว่าเป็นสแกนโค้ดของปุ่มเคปส์ล็อกและเกป(Grave) หรือไม่ ถ้าเป็นมันก็จะดูว่าถ้าเป็นเคปส์ล็อกมันก็จะสลับระหว่างตัวอักษรสูงและตัวอักษรต่ำ แต่ถ้าเป็นเกปมันก็จะสลับภาษาแล้วก็กลับไปรอรับคีย์ต่อไปจากผู้ใช้ ถ้าไม่ทั้งเคปส์ล็อกและเกปมันก็จะแปลงสแกนโค้ดนั้นเป็นรหัสแอสกีแล้วส่งออกไปทางพอร์ตสองเพื่อส่งให้ไมโครคอนโทรลเลอร์ตัวหลัก โดยส่วนของการรับข้อมูลนี้จะสิ้นสุดเมื่อมีการกดปุ่มเอ็นเทอร์เท่านั้น ถ้าไม่กดปุ่มเอ็นเทอร์มันโปรแกรมก็จะวนรอรับข้อมูลจากผู้ใช้เรื่อยๆ ไป โดยโค้ดในส่วนของารรับข้อมูลนี้จะแสดงไว้ในภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



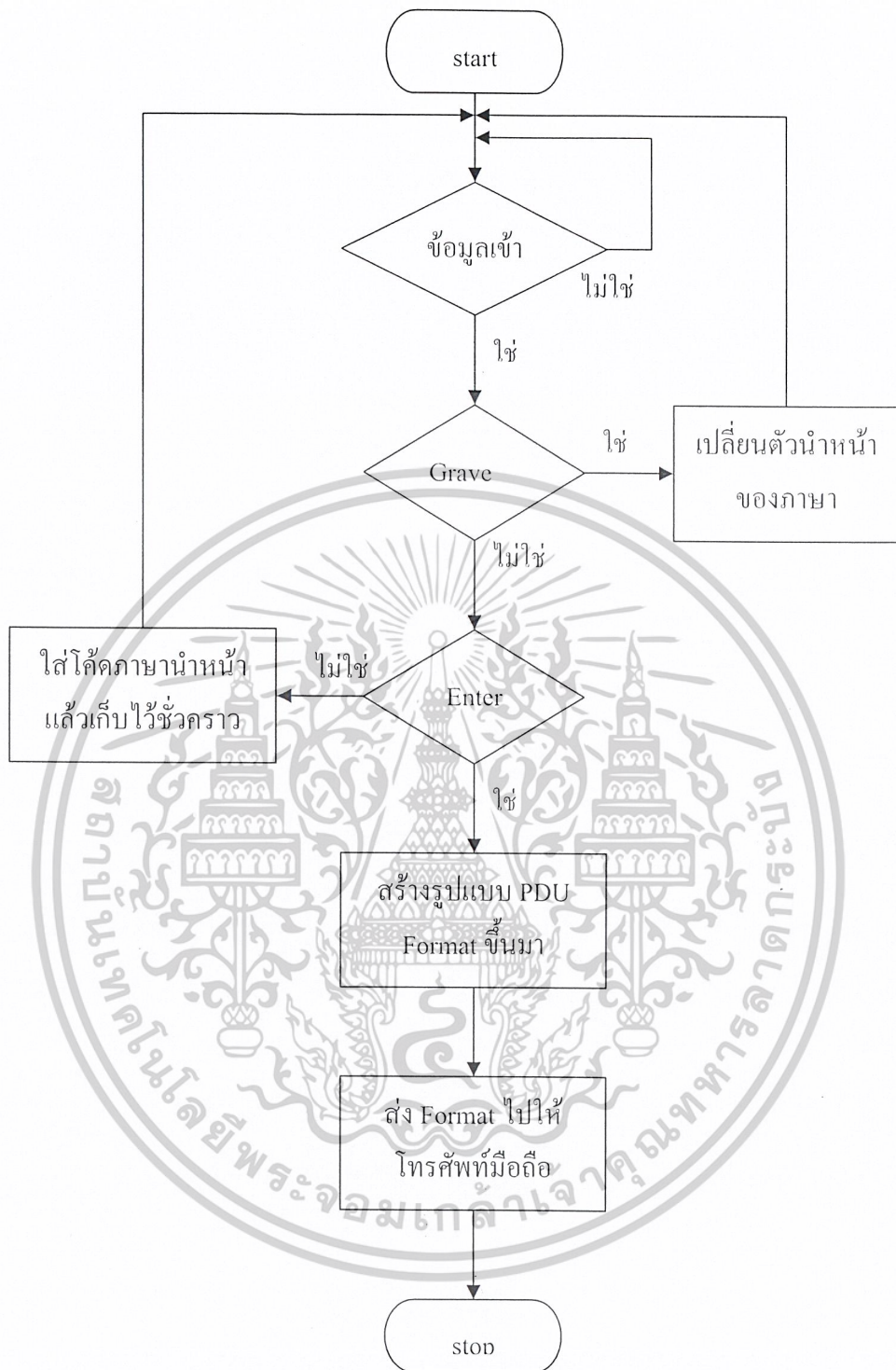
รูปที่ 3.4 แสดงภาพในส่วนของการรับข้อมูลและแปลงสแกน โค้ดเป็นรหัสแอสกี

3.3 การออกแบบส่วนการควบคุมการทำงานของกล้องอินเทอร์เฟสทั้งหมด



รูปที่ 3.5 แสดงภาพในส่วนของการควบคุมระบบทั้งหมดของกล้องอินเทอร์เฟส

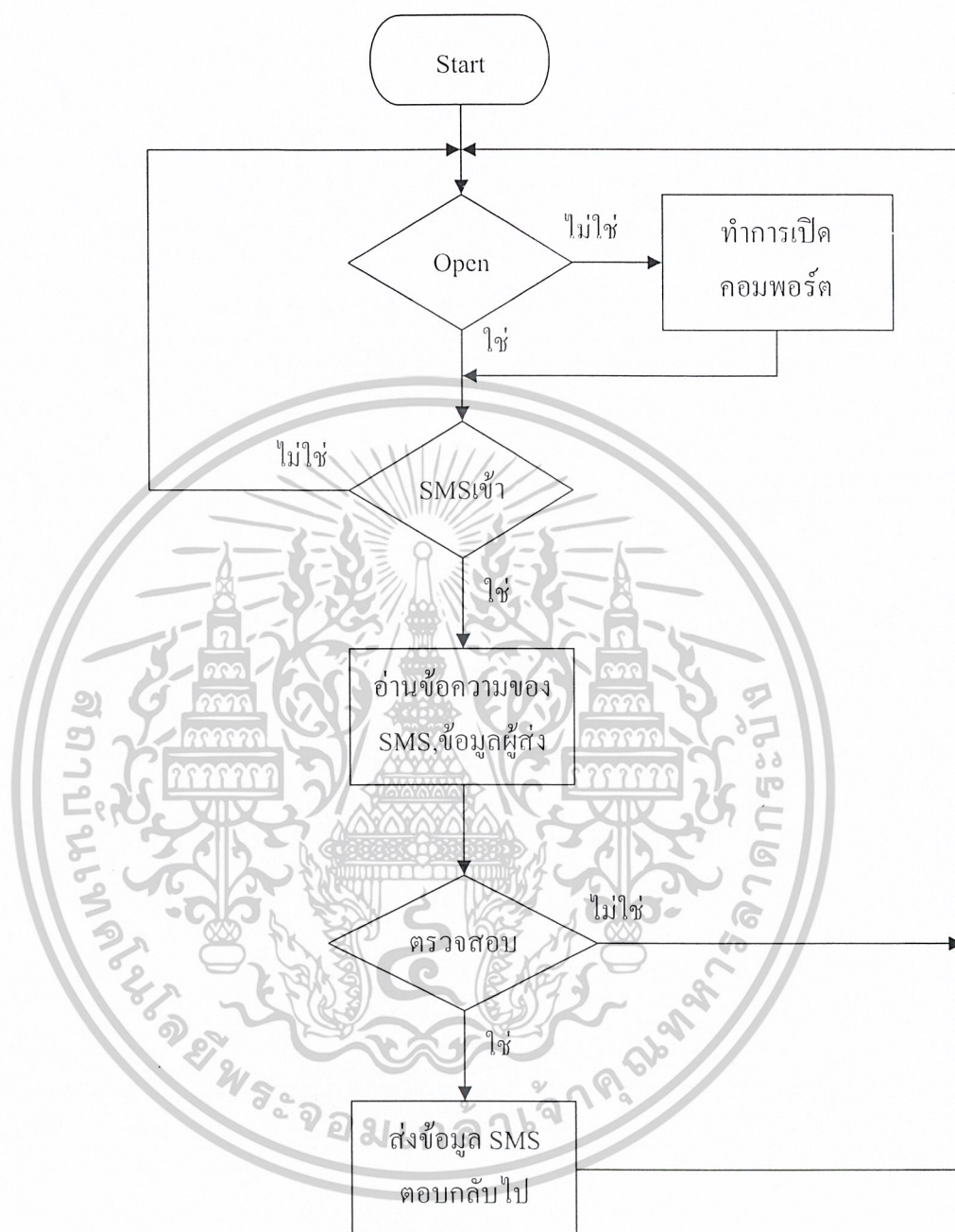
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 Flow chart ในส่วนของการแปลงเป็น PDU mode และควบคุมการทำงานของกล่องจากไออะแกรมด้านบนเป็นการทำงานของโปรแกรมในส่วนของการควบคุมกล่องอินเทอร์เน็ตทั้งหมด โดยการทำงานจะเริ่มจากการรับรหัสแอสกีเข้ามาแล้วทำการแปลงให้อยู่ในรูปของฟิสิกส์ฟอร์แมตแล้วทำการส่งออกไปที่โทรศัพท์มือถือโดยผ่านทางสายดาต้าลิงค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การทำงานในส่วนของเซิร์ฟเวอร์(Server)



รูปที่ 3.7 แสดง Flow chart ในส่วนการทำงานของเซิร์ฟเวอร์

ขั้นตอนการทำงาน

โปรแกรมจะทำการตรวจสอบพอร์ต RS-232 ของเครื่องคอมพิวเตอร์หรือ Comport ว่าเปิดการเชื่อมต่อแล้วหรือยังเพราะโปรแกรมจะทำการติดต่อกับโทรศัพท์มือถือผ่านสายดาต้าลิงค์ที่เชื่อมต่อกับ Port RS-232 ถ้าหากพอร์ตเปิดอยู่แล้ว ลำดับต่อไปจะเป็นการตรวจสอบว่ามีโทรศัพท์มือถือเชื่อมต่อกับสายดาต้าลิงค์หรือไม่โดยโปรแกรมจะส่งชุดคำสั่งเอทีคอมมาน(AT Command)ที่ใช้ในการเรียกดูรุ่นของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โทรศัพท์มือถือ ถ้ามีการเชื่อมต่ออยู่โปรแกรมจะแสดงชื่อรุ่นของโทรศัพท์มือถือทางช่องรุ่นของโทรศัพท์มือถือ พร้อมทั้งแสดงสถานะเชื่อมต่อทางช่องการเชื่อมต่อโทรศัพท์ และพร้อมที่จะทำการตรวจสอบการส่งข้อความ(SMS)เข้ามา

ในส่วนของตรวจสอบการส่งข้อความ(SMS)เข้ามา โปรแกรมจะคอยตรวจสอบว่ามี การส่งข้อความ(SMS)เข้ามาที่โทรศัพท์มือถือหรือไม่ โดยจะทำการตรวจทุกๆ 5 วินาทีซึ่งเมื่อครบ 5 วินาทีโปรแกรมจะทำการส่งชุดคำสั่งเอทีคอมมาน(AT Command) ที่ใช้ในการเรียกดูข้อความใหม่ออกไป เมื่อโทรศัพท์มือถือได้รับชุดคำสั่งแล้วก็จะทำการตอบกลับมา โดยโปรแกรมจะมีบัฟเฟอร์ที่ไว้รับค่าที่โทรศัพท์มือถือส่งกลับมา แล้วโปรแกรมจะตรวจสอบจากบัฟเฟอร์ว่าสิ่งที่โทรศัพท์มือถือตอบกลับมานั้น หมายถึงว่ามีข้อความใหม่เข้ามาหรือไม่ซึ่งถ้าหากว่ามีข้อความใหม่เข้ามา ก็จะแสดงข้อความว่า "SMS Incoming" ที่ช่องสถานะโทรศัพท์ แต่ถ้าหากว่าไม่มีข้อความใหม่เข้ามา ก็จะแสดงข้อความว่า "NO SMS Incoming" ที่ช่องสถานะโทรศัพท์เช่นกัน

ในกรณีที่มีข้อความใหม่เข้ามา โปรแกรมจะอ่านหมายเลขต้นทางของผู้ส่งจากบัฟเฟอร์ที่โทรศัพท์มือถือส่งเข้ามาแล้วจะแสดงหมายเลขของผู้ส่งที่ช่องหมายเลขของผู้ส่ง จากนั้นจะทำการแปลงหมายเลขของผู้ส่งที่ได้ให้อยู่ในรูปแบบ PDU Mode เพื่อใช้ในการส่งข้อความตอบกลับไป หลังจากนั้นโปรแกรมจะทำการอ่านในส่วนเนื้อหาของข้อความที่ผู้ส่งทำการส่งเข้ามา เพื่อที่จะนำไปใช้ในการตรวจสอบจำนวนสินค้าในสต็อก โดยข้อความที่ส่งเข้ามานั้นจะต้องอยู่ในรูปแบบที่กำหนดไว้ด้วย ในที่นี้ได้กำหนดรายการสินค้าที่สามารถตรวจสอบจำนวนสินค้าในสต็อกไว้ 4 รุ่นด้วยกันคือ nokia3310 ,Siemens C35i , Nokia 3650 และ Samsung E700 แต่เนื่องจากข้อความที่ตอบกลับมาจากโทรศัพท์มือถือยังอยู่ในรูปแบบ PDU Mode ไม่ใช่ข้อความธรรมดาที่จะสามารถอ่านแล้วเข้าใจได้เลยดังนั้นเมื่อโปรแกรมทำการอ่านข้อความจากบัฟเฟอร์ก็จะได้ข้อความที่อยู่ในรูปแบบ PDU Mode ซึ่งรายการสินค้าที่กำหนดไว้เมื่อแปลงให้อยู่ในรูปแบบ PDU Mode จะได้ดังนี้

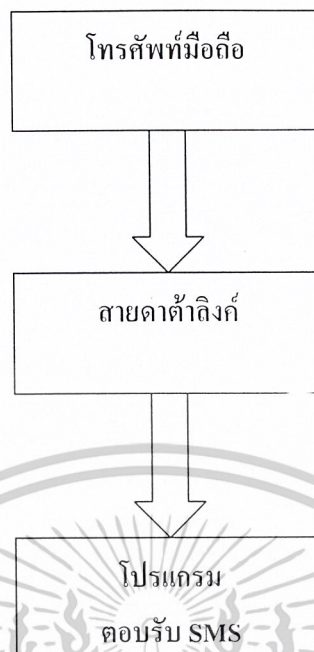
nokia3310 = "EEF73A1D9ECD6230"

Siemens C35i = "D374B95D76CF41C3592D0D"

Nokia 3650 = "CEF73A1D06CD6C3518"

Samsung E700 = "D3707B5E779F41C51B0C06"

เมื่อโปรแกรมอ่านข้อความเรียบร้อยแล้วก็จะนำไปตรวจสอบกับรายการสินค้าที่มีว่าตรงตามสินค้ารุ่นใดแล้วก็จะทำการส่งข้อความตอบกลับไปยังผู้ส่งว่าจำนวนสินค้าในสต็อกของรุ่นนั้นเหลืออยู่จำนวนเท่าไร โดยจะตอบกลับตามหมายเลขของผู้ส่งที่ได้อ่านไว้แล้วในตอนแรก เมื่อส่งข้อความตอบกลับไปแล้วโปรแกรมก็จะกลับมาสู่สถานะเตรียมพร้อมอีกครั้ง โดยจะเริ่มการตรวจสอบการส่งข้อความเข้ามาทุก 5 วินาที



รูปที่ 3.8 แสดงไดอะแกรมการทำงานในส่วนเซิร์ฟเวอร์

ปัญหานี้ในบทนี้จะกล่าวถึงการออกแบบทั้งหมดของระบบ โดยการออกแบบแบ่งออกเป็นสองส่วนใหญ่ๆคือ ส่วนแรกเป็นการทำกล่องอินเทอร์เน็ตระหว่างคีย์บอร์ดของคอมพิวเตอร์และโทรศัพท์มือถือ และอีกส่วนหนึ่งคือส่วนของเซิร์ฟเวอร์เอง แล้วในส่วนของกล่องอินเทอร์เน็ตยังแบ่งออกเป็นสองส่วนคือส่วนที่ทำหน้าที่รับข้อมูลเข้ามาทางคีย์บอร์ดแล้วแปลงเป็นสแกนโค้ด และส่วนที่ทำหน้าที่ควบคุมการทำงานทั้งหมดของกล่อง

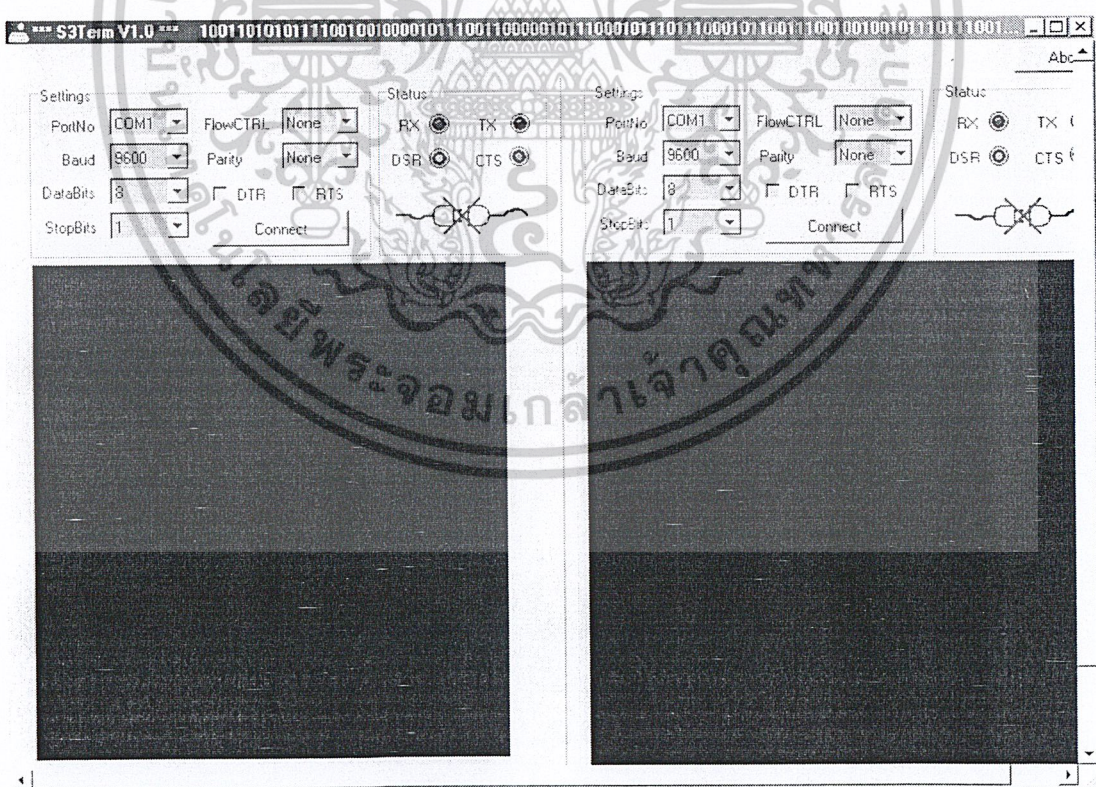
บทที่ 4

การทดสอบและวิเคราะห์ผล

ในแต่ละขั้นตอนของการพัฒนาโครงการนั้น มีการทดสอบและวิเคราะห์ผลว่าสิ่งที่ได้ทำไปแล้วนั้น เป็นไปตามเป้าหมายของแต่ละขั้นตอนหรือไม่ ซึ่งจำเป็นต้องให้เป็นไปตามเป้าหมายนั้น แต่ถ้าไม่สามารถทำให้เป็นไปตามเป้าหมายที่วางไว้ได้ จะต้องพิจารณาสิ่งที่เกิดขึ้นและหาทางแก้ไขในโครงสร้างของระบบหรือขั้นตอนและเป้าหมายต่างๆแต่การแก้ไขจะต้องไม่ให้กระทบถึงเป้าของโครงการ เมื่อได้ทำตามขั้นตอนต่างๆเป็นที่เรียบร้อยแล้วนั้น จะต้องมีการทดสอบและวิเคราะห์ระบบทั้งหมดอีกทีหนึ่ง ว่าระบบที่พัฒนาแก้ไขใหม่นั้นเป็นไปตามเป้าหมายของโครงการหรือไม่

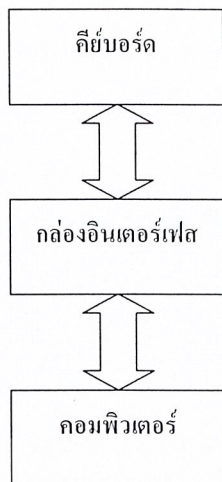
4.1 การทดสอบการทำงานของฮาร์ดแวร์

การทดสอบทางด้านฮาร์ดแวร์นั้น ก็คือการทดสอบตัวของกล่องอินเตอร์เฟสนั้นเอง โดยจะทดสอบผ่านทางพอร์ตอนุกรมก่อนแล้วจึงค่อยทดสอบผ่านตัวโทรศัพท์มือถือจริงๆ โดยการทดสอบผ่านพอร์ตอนุกรมของคอมพิวเตอร์จะทำให้เราทราบว่าของที่ส่งออกมาจากพอร์ตอนุกรมของกล่องอินเตอร์เฟสนั้น ตรงกันรูปแบบที่เราได้กำหนดไว้ในโปรแกรมหรือไม่ โดยการทดสอบจะทดสอบโดยการใช้โปรแกรมที่ชื่อว่าเอสทีเทอม(S3Term)ซึ่งโปรแกรมตัวนี้จะเป็นตัวที่ตรวจสอบว่ามีข้อมูลอะไรบ้างที่เข้ามาทางพอร์ตอนุกรมของคอมพิวเตอร์บ้าง



รูปที่ 4.1 แสดงหน้าต่างโปรแกรม S3Term ซึ่งเป็นโปรแกรมที่ใช้ในการทดสอบฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงไดอะแกรมของการทดสอบฮาร์ดแวร์

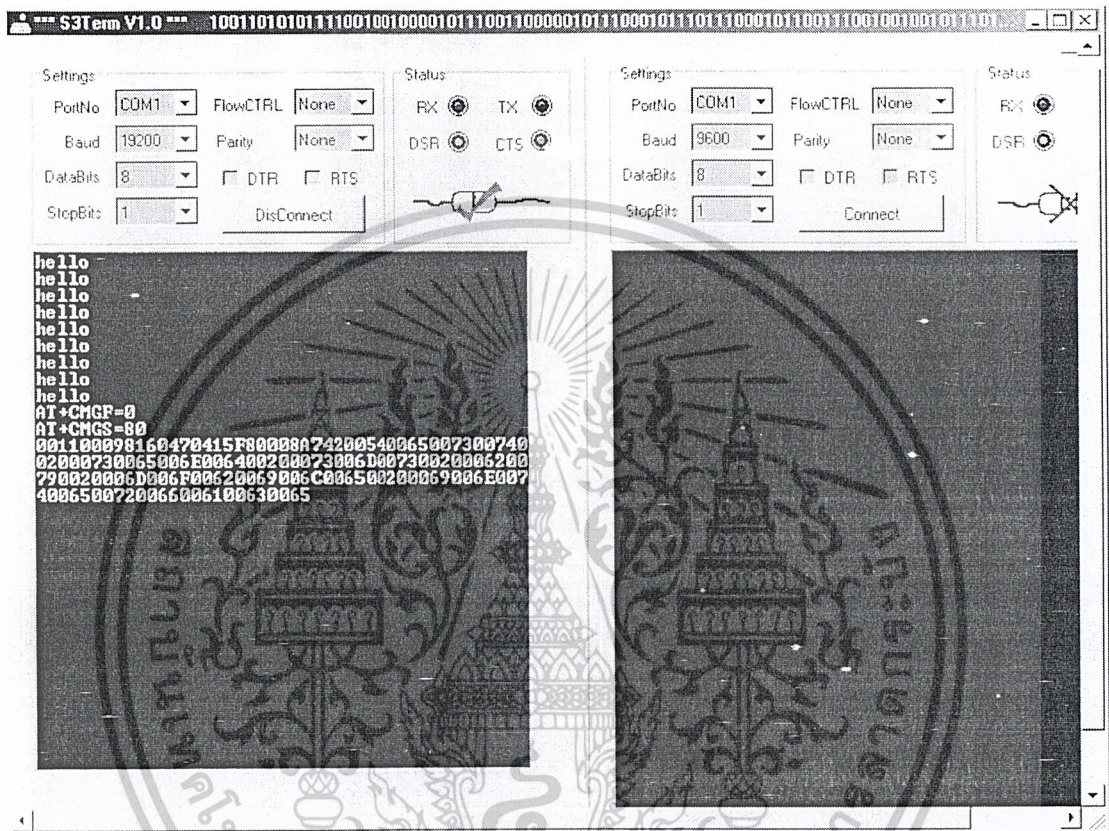
ขั้นตอนต่อไปคือการเชื่อมต่อค่าให้กับโปรแกรม S3Term โดยเชื่อมต่อของพอร์ตอนุกรมที่เราทำการติดต่อ เช่น ถ้าเราเสียบสายที่ต่อกล่องอินเทอร์เฟสไว้ที่คอม 1 ก็ให้เชื่อมต่อในโปรแกรมให้เป็นคอม 1 ตามที่เราต่อฮาร์ดแวร์ไว้ ต่อไปก็ทำการเชื่อมต่อของบอดเรตให้มีความเท่ากับ 19200 ซึ่งเป็นค่าบอดเรตของโทรศัพท์มือถือที่นำมาทำการทดสอบ ถ้าเชื่อมต่อเป็นอย่างอื่น โปรแกรมก็ยังคงแสดงค่าได้ แต่ค่าที่ออกมาจะอ่านไม่รู้เรื่องจากรูปข้างล่างจะแสดงให้เห็นค่าที่ทำการเชื่อมต่อแล้วจากตัวโปรแกรม



รูปที่ 4.3 แสดงรูปของโปรแกรม S3Term ซึ่งทำการเชื่อมต่อพร้อมทำการทดสอบแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบเราจะต่อคีย์บอร์ดเข้ากับกล่องอินเตอร์เฟซผ่านพอร์ตพีเอสทูของกล่องเตอร์เฟส ส่วนกล่องอินเตอร์เฟซก็ด้านเข้าที่พูดจะต่อเข้าทางพอร์ตอนุกรมของคอมพิวเตอร์ เมื่อเราจ่ายไฟให้กับกล่องอินเตอร์เฟซขั้นตอนต่อไปคือการใส่โปรแกรมเพื่อการทดสอบส่งข้อมูล โดยการเปิดโปรแกรมขึ้นมาแล้วก็ทำการพิมพ์ข้อความคีย์บอร์ดของคอมพิวเตอร์เมื่อพิมพ์เสร็จแล้วก็กดเอ็นเทอร์ต่อไปคือใส่หมายเลขโทรศัพท์มือถือปลายทางเมื่อเรากดปุ่มเอ็นเทอร์แล้วก็จะได้ผลลัพธ์ดังนี้



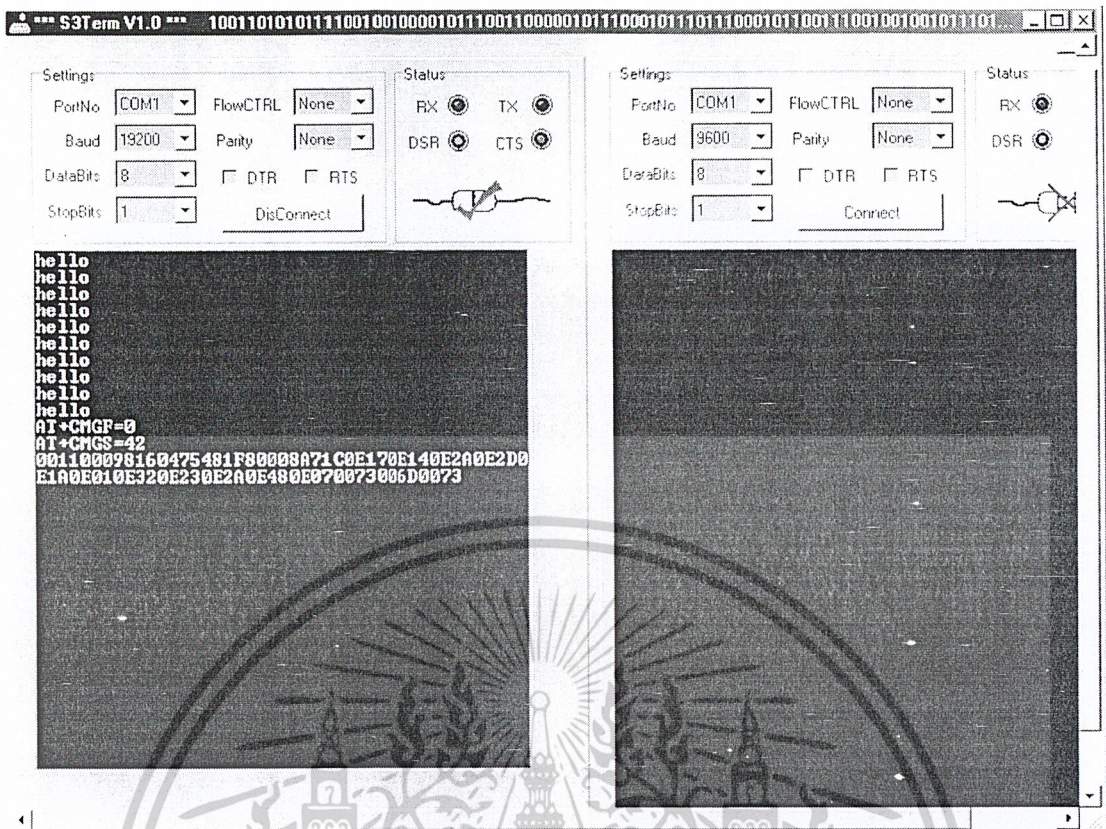
รูปที่ 4.4 แสดงผลของการทดสอบโดยใช้โปรแกรม S3Term ในการทดสอบฮาร์ดแวร์

จากรูปข้างบนนั้น เราได้ใช้โปรแกรม S3Term ในการทดสอบการส่งข้อความ(Short Message) จะได้ผลออกมาดังรูป โดยข้อความที่ใช้ในการทดสอบคือ Test send sms by mobile interface โดยข้อความนี้จะถูกส่งไปที่โทรศัพท์มือถือปลายทางหมายเลข 067440158 ซึ่งในส่วนของข้อความนั้นจะเป็นเลขฐานสิบหก ซึ่งข้อความที่แปลงเป็นเลขฐานสิบหกแล้วจะได้ดังนี้

0054006500730074002000730065006E006400200073006D00730020006200790020006D000F
00620069006C006500200069006E0074006500720066006100630065

ซึ่งเลขข้างบนนั้นเป็นเฉพาะส่วนของข้อความเท่านั้น ส่วนประกอบอื่นๆสามารถดูได้จากภาคผนวก ส่วนรูปต่อไปจะแสดงการส่งภาษาไทยและภาษาอังกฤษไปด้วยกัน โดยการทดสอบก็เหมือนกันข้างบน แต่จะส่งภาษาไทยออกไปด้วยเท่านั้น โดยข้อความที่เราจะนำมาทดสอบก็คือ ทดสอบการส่งsms เมื่อเราทำการส่งแล้วจะได้ข้อมูลออกมาทำโปรแกรมที่ทำการทดสอบดังรูปต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



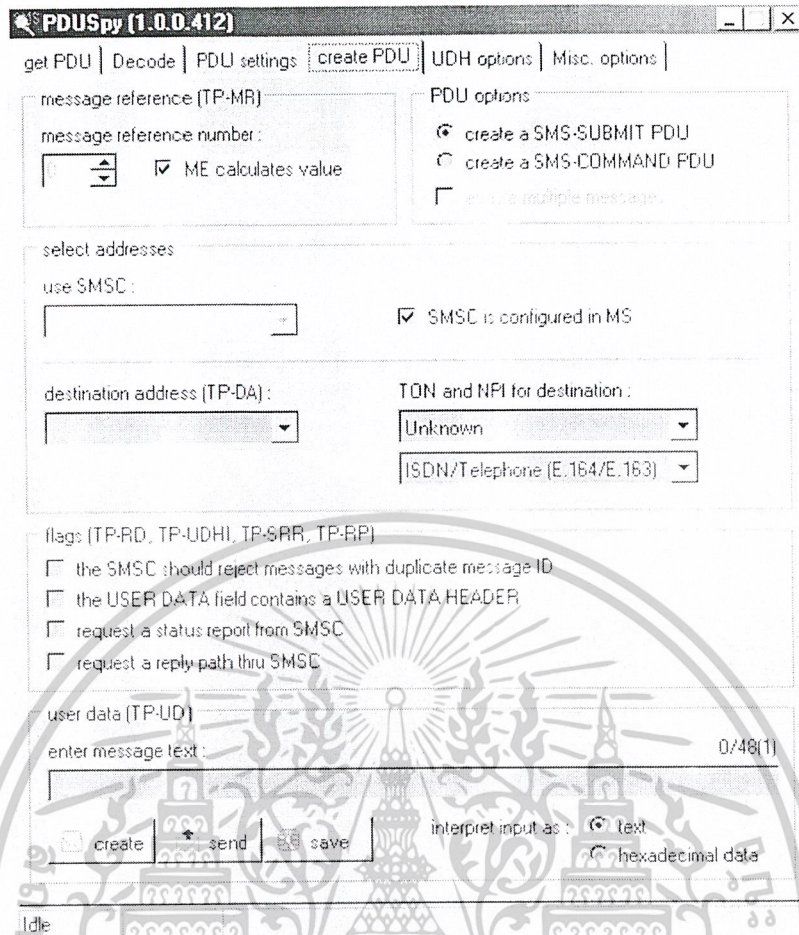
รูปที่ 4.5 แสดงการส่งข้อมูลที่เป็นภาษาไทยและภาษาอังกฤษบนกัน

เมื่อเราทำการส่งข้อมูลแล้วก็จะ ได้ข้อความเหมือนกับรูปข้างบนเพื่อที่จะ ส่งออกไปจากกล่อง อินเทอร์เน็ตแล้วเข้าสู่โทรศัพท์มือถือ เมื่อโทรศัพท์มือถือได้รับข้อความแล้ว มันก็จะส่งไปที่หมายเลข โทรศัพท์ปลายทาง หมายเลข โทรศัพท์ปลายทางก็จะได้รับข้อความ(Short Message)ที่มีข้อความว่า ทดสอบการส่งsms ถ้าเราดูที่โทรศัพท์หมายเลขปลายทางว่าข้อความนั้นถูกต้องตามที่เราส่งหรือไม่ถ้า ถูกต้องและเหมือนกันกับต้นทาง แสดงว่าฮาร์ดแวร์เราทำงานได้สมบูรณ์

4.2 การทดสอบการทำงานของซอฟต์แวร์

การเขียนโปรแกรมเข้าไปในไมโครคอนโทรลเลอร์นั้น การตรวจสอบที่ง่ายและรวดเร็วที่สุดคือการ ใช้ดีบั๊กเกอร์ของโปรแกรมที่เรานำมาเขียน โดยการควบคุมการทำงานของกล่องอินเทอร์เน็ตถูกเขียนมาจากโปรแกรม BASCOM 8051 IDE ซึ่งการเขียนจะคล้ายกับการเขียนภาษาเบสิกแต่เป็นการเขียนบน ไมโครคอนโทรลเลอร์ ซึ่งเมื่อทำการทดสอบโดยใช้ดีบั๊กเกอร์ของโปรแกรมแล้ว ถ้าทำการเบิร์นได้ใน ไมโครคอนโทรลเลอร์ แล้วก็ต้องมีการตรวจสอบอีกครั้งหนึ่งซึ่งต้องดูเอาที่พูดว่าตรงตามที่เราต้องการ หรือไม่โดยการโปรแกรมที่ชื่อว่า PDUspy ซึ่งโปรแกรมนี้จะใช้แปลงข้อความเป็นเลขฐาน 16 ให้เรา ซึ่ง เมื่อเราได้จากโปรแกรมนี้แล้วก็เอาไปเทียบกับเอาที่พูดในส่วนของข้อความว่าตรงกันหรือไม่ ถ้าเรา เขียน โปรแกรมให้ทำงานได้ถูกต้องมันจะต้องตรงกัน โดยโปรแกรมนี้จะแปลงได้ทั้งภาษาไทยและ ภาษาอังกฤษ ก่อนทำการแปลงเราก็ทำการเช็คค่าต่างๆก่อน เช่น ต้องการเป็นแบบที่บิตโดยรูปต่อไปจะ แสดงหน้าต่างของโปรแกรม PDUspy

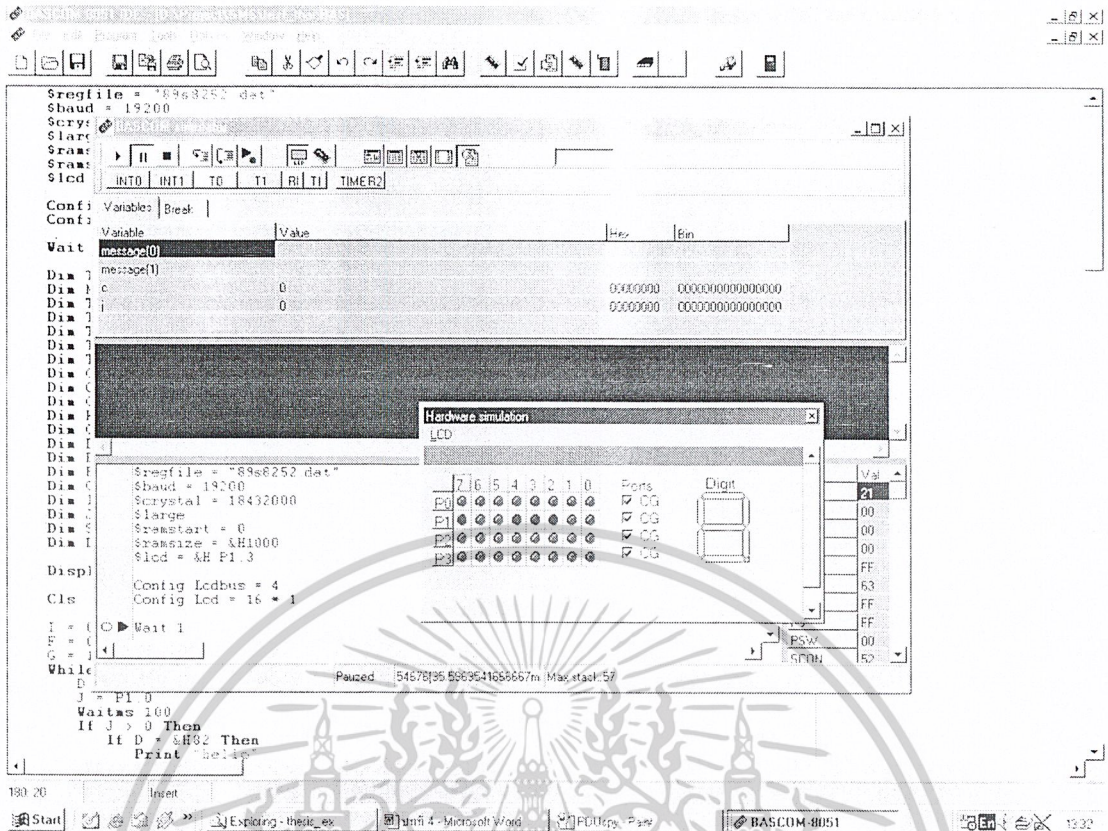
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงหน้าต่างของโปรแกรม PDUspy

การทำงานของดีบักเกอร์ของ BASCOM 8051 IDE นั้นจะแสดงให้เห็นดังรูปต่อไป เราก็เปิดหน้าต่างของโปรแกรมขึ้นมา เมื่อเปิดขึ้นมาแล้วก็ทำการ โหลดโปรแกรมที่เราต้องการตรวจสอบ มาไว้ในหน้าต่าง ที่ต้องเปิดโปรแกรมที่ต้องการดีบักไว้หน้าต่างหลักก็เพราะว่าโปรแกรม BASCOM นี้สามารถทำงานหลายโปรแกรมได้พร้อมกัน เมื่อโปรแกรมที่ต้องการตรวจสอบอยู่ในหน้าต่างหลักแล้วก็กดปุ่มรูปไอชี่สีแดงๆ ที่เขียนไว้ว่า simulate program เมื่อทำการกดแล้วจะได้โปรแกรมตรวจสอบการทำงานขึ้นมาอีกหนึ่งโปรแกรม โดยโปรแกรมตรวจสอบนี้จะทำงานตั้งแต่บรรทัดแรกไปเรื่อยๆถ้าจบแล้วก็จะวนกลับมาใหม่ โดยจะทำการตรวจสอบที่ละบรรทัดโดยการกดปุ่ม F8 แล้วเรายังสามารถใส่ตัวแปรที่เราต้องการทราบค่าลงไปได้ด้วยในช่อง variable ส่วนแถบสีน้ำเงินคือข้อมูลที่จะออกพอร์ต RS-232 ของไมโครคอนโทรลเลอร์ และโปรแกรมนี้เรายังสามารถเรียกให้การตรวจสอบผลที่จะส่งออกทางแอลซีดี และเซเวนเซกเมนได้ด้วย ด้วยการกดปุ่ม LCD ข้างบน โปรแกรมนี้ถือว่าเป็นโปรแกรมที่ดีในการตรวจสอบการทำงานของซอฟต์แวร์ เพราะทำให้เราได้เห็นภาพการทำงานจริงๆของโปรแกรม แต่ข้อเสียของโปรแกรม BASCOM 8051 IDE คือไฟล์ที่เขียนเสร็จแล้วจะมีขนาดใหญ่กว่าภาษาอื่น และการกำหนดระดับบิตของไมโครคอนโทรลเลอร์ทำได้ยากกว่าภาษาอื่นด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงหน้าต่างของโปรแกรม BASCOM 8051 IDE และ simulate ของโปรแกรม

การตรวจสอบทางด้านซอฟต์แวร์นี้ เมื่อทำการตรวจสอบเสร็จแล้วก็ต้องไปใส่ไมโครคอนโทรลเลอร์อีกทีหนึ่งแล้วทำการทดสอบฮาร์ดแวร์ว่าทำงานถูกต้องตามที่ได้กำหนดไว้หรือไม่ ที่ต้องตรวจสอบอีกครั้งหนึ่งเพราะว่า โปรแกรมที่เราเขียนอยู่นี้ทำงานบนไมโครคอนโทรลเลอร์อีกทีหนึ่งหรือจะกล่าวได้ว่าโปรแกรมนี้ทำงานในตัวฮาร์ดแวร์อีกทีหนึ่ง

เมื่อทำการตรวจสอบผลลัพธ์ทั้งหมดแล้วตรงกับเป้าหมายที่เราได้วางไว้ถือว่าฮาร์ดแวร์และซอฟต์แวร์ที่เราสร้างขึ้นมานั้นทำงานได้อย่างสมบูรณ์ตามที่เราต้องการ ถ้าเราเพิ่มฟังก์ชันการทำงานหรือเพิ่มส่วนประกอบอื่นๆเข้าไปเราก็ต้องทำการทดสอบระบบใหม่หมด เพื่อให้แน่ใจว่าระบบนั้นจะทำงานได้ถูกต้องตามเป้าหมายที่เราได้วางเอาไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และสรุปผล

5.1 ผลที่ได้รับจากโครงการ

5.1.1 กล้องอินเตอร์เฟสนี้ทำหน้าที่เป็นตัวเชื่อมให้กับโทรศัพท์มือถือและคีย์บอร์ดคอมพิวเตอร์ให้สามารถทำงานร่วมกันได้ โดยการพิมพ์ตามปกตินั้นจะต้องพิมพ์โดยใช้คีย์ของโทรศัพท์มือถือซึ่งมีขนาดเล็กแต่ถ้าใช้กล้องนี้สามารถพิมพ์ที่ตัวคีย์บอร์ดของคอมพิวเตอร์ได้เลย ทำให้สะดวกกว่าเพราะคีย์บอร์ดของคอมพิวเตอร์มีขนาดใหญ่กว่าของโทรศัพท์มือถือมาก

5.1.2 ทำให้การพิมพ์ตัวอักษร สะดวกและรวดเร็ว เพราะคีย์บอร์ดของคอมพิวเตอร์มีขนาดใหญ่ทำให้การพิมพ์ทำได้สะดวกและรวดเร็ว ไม่ต้องใช้เวลาในการพิมพ์นานเหมือนคีย์ของโทรศัพท์มือถือ

5.1.3 เป็นโครงการที่สามารถนำไปใช้พัฒนาต่อได้ เพื่อให้เกิดประโยชน์มากขึ้น

5.2 ปัญหาที่พบในการทำโครงการ

จากการที่ได้ทำโครงการนี้ต้องค้นคว้าหาข้อมูลอยู่มาก เนื่องจากเป็นโครงการที่ต้องพัฒนาทั้งด้านฮาร์ดแวร์และซอฟต์แวร์ให้ทำงานควบคู่กันไปได้ ซึ่งเป็นเรื่องที่ยากและใช้เวลาก่อนข้างมากอีกทั้งผู้พัฒนาไม่เคยมีประสบการณ์และความเชี่ยวชาญมาก่อน ปัญหาจึงมีมากพอสมควร ปัญหาที่พบในการทำโครงการแบ่งออกเป็น 3 ส่วนใหญ่ๆ

- ปัญหาทางด้านฮาร์ดแวร์
- ปัญหาทางด้านซอฟต์แวร์
- ปัญหาทางด้านเทคนิคต่างๆ

5.2.1 ปัญหาทางด้านฮาร์ดแวร์

5.2.1.1 เนื่องจากการพัฒนาบอร์ดในช่วงแรก ประสบปัญหาเกี่ยวกับการหาข้อมูล เพราะการหาข้อมูลที่ยากจึงทำให้การทำงานล่าช้าออกไปมาก

5.2.1.2 การขาดประสบการณ์ของผู้พัฒนา ในการทำฮาร์ดแวร์จึงต้องล่าช้าออกไปล่าช้ากว่าที่กำหนดไว้

5.2.1.3 ปัญหาในการออกแบบฮาร์ดแวร์ เพราะผู้พัฒนามีประสบการณ์น้อยทำให้ไม่สามารถมองระบบได้ออกทั้งหมด

5.2.2 ปัญหาทางด้านซอฟต์แวร์

5.2.2.1 ต้องใช้เวลานานพอสมควรในการศึกษาเครื่องมือที่จะนำมาใช้ในการพัฒนาโปรแกรมของบอร์ดในช่วงเริ่มต้นของการพัฒนา

5.2.2.2 ต้องใช้เวลาในการศึกษาและใช้งานภาษาที่เลือกมาในการพัฒนาโปรแกรม

5.2.3 ปัญหาทางด้านเทคนิคต่างๆ

5.2.3.1 ขาดเครื่องมือที่ทันสมัยในการพัฒนา

5.2.3.2 หาข้อมูลในเรื่องที่ใช้ทำโครงการมีน้อยมาก และเครื่องมือบางอย่างไม่เคยใช้มาก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางการพัฒนาต่อ

- พัฒนาในเรื่องของโปรแกรมเพื่อให้มีขนาดเล็กลงโดยไม่ต้องใช้หน่วยความจำภายนอก
- พัฒนาในเรื่องของขนาดของบอร์ดหรือกล่องอินเตอร์เฟสให้มีขนาดเล็กลงเพื่อให้สะดวกแก่การพกพามากขึ้น
- นำไปประยุกต์ใช้งานต่อ

5.4 สรุปผลโครงการ

การทำงานของกล่องอินเตอร์เฟสที่เป็นตัวเชื่อมระหว่างโทรศัพท์มือถือและคีย์บอร์ดของคอมพิวเตอร์ จะมีหลักการที่สำคัญ คือ เริ่มต้นจากการกดแป้นพิมพ์ของคีย์บอร์ดคอมพิวเตอร์ เมื่อมีการกดแป้นพิมพ์ คีย์บอร์ดของคอมพิวเตอร์ก็จะส่งค่าสแกนโค้ด จากแป้นพิมพ์เข้ามาที่กล่องอินเตอร์เฟส ซึ่งทำหน้าที่แปลง ข้อมูลที่เป็นสแกนโค้ดเป็นรหัสแอสกีแล้วทำให้ข้อมูลที่รับข้อมานั้นให้อยู่ในรูปแบบของการส่งข้อความ (SMS) การส่งข้อมูลจะส่งออกไปทางพอร์ตอนุกรมที่ต่อกับกล่องอินเตอร์เฟสผ่านเข้าทางพอร์ตของมือถือ เมื่อตัวมือถือได้รับรูปแบบ (Format) ในการส่งข้อความนี้แล้วมันก็จะส่งข้อความออกไปหา โทรศัพท์มือถือที่อยู่ปลายทาง โดยผ่านเครือข่ายของโทรศัพท์มือถือเอง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

โค้ดของโปรแกรม

โค้ดของโปรแกรมส่วนหลัก เขียนจากภาษา เบสิก

```
$regfile = "89s8252.dat"
```

```
$baud = 19200
```

```
$crystal = 18432000
```

```
$large
```

```
$ramstart = 0
```

```
$ramsize = &H1000
```

```
$lcd = &H P1.3
```

```
Config Lcdbus = 4
```

```
Config Lcd = 16 * 1
```

```
Wait 1
```

```
Dim Telno(10) As Xram String * 1
```

```
Dim Message(500) As Xram String * 4
```

```
Dim Temp As String * 4
```

```
Dim Temp1 As String * 2
```

```
Dim Temp2 As String * 1
```

```
Dim Temp3 As String * 1
```

```
Dim Temp4 As String * 2
```

```
Dim Count As Byte
```

```
Dim Count1 As Byte
```

```
Dim Count2 As Byte
```

```
Dim K1 As String * 1
```

```
Dim C As Byte
```

```
Dim D As Byte
```

```
Dim E As Byte
```

```
Dim F As Byte
```

```
Dim G As Byte
```

```
Dim I As Byte
```

```
Dim J As Byte
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim Sum As Byte
```

```
Dim Dif As Byte
```

```
Display On
```

```
Cls
```

```
I = 0
```

```
F = 0
```

```
G = 1
```

```
While P2 <> &H81
```

```
    D = P2
```

```
    J = P1.0
```

```
    Waitms 100
```

```
    If J > 0 Then
```

```
        If D = &H82 Then
```

```
            Print "hello"
```

```
            If E = 1 Then
```

```
                E = 0
```

```
            ElseIf E = 0 Then
```

```
                E = 1
```

```
            End If
```

```
            J = 0
```

```
        End If
```

```
    End If
```

```
    If J > 0 Then
```

```
        If D = &H83 Then
```

```
            I = I - 1
```

```
            G = G - 1
```

```
            'Waitms 100
```

```
            'Lcd Chr(p2)
```

```
            F = F - 1
```

```
            Shiftcursor Left
```

```
            J = 0
```

```
        End If
```

```
    End If
```

```
    If J > 0 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If E = 1 Then
```

```
    C = &H02
```

```
Elseif E = 0 Then
```

```
    C = &H03
```

```
End If
```

```
If C = 2 Then
```

```
    Temp1 = "0E"
```

```
Elseif C = 3 Then
```

```
    Temp1 = "00"
```

```
End If
```

```
C = D / 16
```

```
If C > 9 Then
```

```
    Select Case C
```

```
        Case 10 : Temp2 = "A"
```

```
        Case 11 : Temp2 = "B"
```

```
        Case 12 : Temp2 = "C"
```

```
        Case 13 : Temp2 = "D"
```

```
        Case 14 : Temp2 = "E"
```

```
        Case 15 : Temp2 = "F"
```

```
    End Select
```

```
Elseif C < 10 Then
```

```
    Temp2 = Str(c)
```

```
End If
```

```
C = D Mod 16
```

```
If C > 9 Then
```

```
    Select Case C
```

```
        Case 10 : Temp3 = "A"
```

```
        Case 11 : Temp3 = "B"
```

```
        Case 12 : Temp3 = "C"
```

```
        Case 13 : Temp3 = "D"
```

```
        Case 14 : Temp3 = "E"
```

```
        Case 15 : Temp3 = "F"
```

```
    End Select
```

```
Elseif C < 10 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Temp3 = Str(c)
End If
Temp4 = Temp2 + Temp3
Message(g) = Temp1 + Temp2 + Temp3
I = I + 1
G = G + 1
Waitms 50
Lcd Chr(p2)
F = F + 1
If F > 8 Then
    Shifted Left
End If
End If
Wend
*****
Count = 1
Count1 = Count * 2
Count2 = Count1 + 12
*****
Cls
I = 1
G = 0
F = 55
While F <> &H81
    J = P1.0
    If J > 0 Then
        Waitms 100
        F = P2
        E = P2 - 16
        'print "hello"
        If E > 10 Then
            E = E - 32
        End If
        If E < 10 Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Telno(i) = Str(c)
Waitms 50
Print "hello"
'Print Telno(i)
Lcd Telno(i)
I = I + 1
G = G + 1
If G > 8 Then
    Shiftlcd Left
End If
End If
End If
Wend
Cls
*****
Print "AT+CMGF=0"
Wait 2
Print "AT+CMGS=" ; Count2
Wait 2
Print "0011000981";
*****
For I = 1 To 9 Step 2
    J = I + 1
    If J >= 9 Then
        Print "F";
    Else
        K1 = Telno(j)
        Print Chr(k1);
    End If
    K1 = Telno(i)
    Print Chr(k1);
Next
*****
Print "0008A7";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Count = Count * 2
Count = Count - 2
Printex Count;
Count = Count + 2
Count = Count / 2
For I = 2 To Count
  Print Message(i);
Next
Print Chr(26);
*****
End

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โค้ดของโปรแกรมส่วนที่รับข้อมูลเข้ามาแล้วแปลงสแกนโค้ดให้กลายเป็นรหัสแอสกี

```
#pragma code symbols debug
#include <reg51.h>
#include <stdio.h>
#define bit2 0x04
#define bit7 0x80
unsigned char data data_key[8];
unsigned char data key_hi=0xff,key_lo=0xff,key_head=0,key_tcl=0;
bit key_rev = 0;
unsigned char bdata btest = 0;
sbit b00 = btest^0;
sbit key_data = P3^4;
sbit reset = P0^0;
sbit compare = P0^1;
sbit enter = P0^2;
sbit grave = P0^3;
void initial()
{
EA = 0;
IE = 0x00;
P0 = 0xff;
P1 = 0xff;
P2 = 0xff;
P3 = 0xff;
TMOD = 0x22;
SCON = 0x50;
PCON = 0x80;
TH1 = 255;
TR1 = 1;
ES = 0;
IT0 = 1;
IE0 = 0;
EX0 = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

EA = 1;
compare = 0;
}

void delay(int count)
{
    int i,j;
    for(i = 0;i < count;i++)
        for(j = 0;j < 500;j++);
}

void main(void){
    int flag1 = 0;
    int flag2 = 0;
    int flag3 = 0;
    reset = 1;
    initial();
    T1 = 1;
    reset = 0;
    grave = 0;
    while(1)
    {
        if(key_head != key_tel)
            {
                if(P1 == 0x1a) compare = 0;
                else compare = 1;
                delay(30);
                key_tel = (key_tel + 1)&0x07;
                T1 = 1;
                P1 = data_key[key_tel];
                if(P1 == 0x70) //grave
                    {
                        if(flag3 == 0) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        flag3 = 1;
        grave = 1;
    }
    else {
        flag3 = 0;
        grave = 0;
    }
}

if(P1 == 0x1a) //caplock
{
if(flag2 == 0) flag2 = 1;
    else flag2 = 0;
}
if(P1 == 0x1a) //caplock
{
if(flag1 == 0) flag1 = 1;
    else flag1 = 0;
}
/*if(P1 == 0x1a) //caplock
{
P2 = 0x80;
}*/
if(P1 == 0x5a) //enter
{
P2 = 0x81;
}
if(P1 == 0x70) //grave
{
P2 = 0x82;
}
if(P1 == 0x66) //delete
{
P2 = 0x83;
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

if(flag3 == 0);
if(flag2 == 0);           //English (little)
switch(P1)
{
case 0x38: P2 = 0x61;break; // "a"
        case 0x4c: P2 = 0x62;break;
case 0x84: P2 = 0x63;break;
        case 0xc4: P2 = 0x64;break;
case 0x24: P2 = 0x65;break;
        case 0xd4: P2 = 0x66;break;
case 0x2c: P2 = 0x67;break;
        case 0xcc: P2 = 0x68;break;
case 0xc2: P2 = 0x69;break;
        case 0xdc: P2 = 0x6a;break;
case 0x42: P2 = 0x6b;break;
        case 0xd2: P2 = 0x6c;break;
case 0x5c: P2 = 0x6d;break;
        case 0x8c: P2 = 0x6e;break;
case 0x22: P2 = 0x6f;break;
        case 0xb2: P2 = 0x70;break;
case 0xa8: P2 = 0x71;break;
        case 0xb4: P2 = 0x72;break;
case 0xd8: P2 = 0x73;break;
        case 0x34: P2 = 0x74;break;
case 0x3c: P2 = 0x75;break;
        case 0x54: P2 = 0x76;break;
case 0xb8: P2 = 0x77;break;
        case 0x44: P2 = 0x78;break;
case 0xac: P2 = 0x79;break;
        case 0x58: P2 = 0x7a;break; // "z"
case 0x68: P2 = 0x31;break; // "1"
        case 0x78: P2 = 0x32;break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 0x64: P2 = 0x33;break;
        case 0xa4: P2 = 0x34;break;
case 0x74: P2 = 0x35;break;
        case 0x6c: P2 = 0x36;break;
case 0xbc: P2 = 0x37;break;
        case 0x7c: P2 = 0x38;break;
case 0x62: P2 = 0x39;break; // "9"
        case 0xa2: P2 = 0x30;break; // "0"
//case 0x5a: P2 = 0x55;break; //enter
        case 0x94: P2 = 0x20;break; //space
        case 0x72: P2 = 0x2d;break; // "-"
case 0xaa: P2 = 0x3d;break; // "="
        case 0xba: P2 = 0x5c;break; // "\"
        case 0x82: P2 = 0x2c;break; // ","
case 0x92: P2 = 0x2e;break; // "."
        case 0x52: P2 = 0x2f;break; // "/"
case 0x32: P2 = 0x3b;break; // ";"
        case 0x4a: P2 = 0x27;break; // "'"
case 0x2a: P2 = 0x5b;break; // "["
        case 0xda: P2 = 0x5d;break; // "]"
        case 0x0c: P2 = 0x30;break; // "0" numlock
        case 0x96: P2 = 0x31;break; // "1"
        case 0x4e: P2 = 0x32;break;
case 0x5e: P2 = 0x33;break;
        case 0xd6: P2 = 0x34;break;
case 0xce: P2 = 0x35;break;
        case 0x2e: P2 = 0x36;break;
case 0x36: P2 = 0x37;break;
        case 0xae: P2 = 0x38;break;
case 0xbe: P2 = 0x39;break; // "9"
}
}

else if(flag2 == 1) //English (big)
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

switch(P1)
{
case 0x38: P2 = 0x41;break; // "A"
        case 0x4c: P2 = 0x42;break;
case 0x84: P2 = 0x43;break;
        case 0xc4: P2 = 0x44;break;
case 0x24: P2 = 0x45;break;
        case 0xd4: P2 = 0x46;break;
case 0x2c: P2 = 0x47;break;
        case 0xcc: P2 = 0x48;break;
case 0xc2: P2 = 0x49;break;
        case 0xdc: P2 = 0x4a;break;
case 0x42: P2 = 0x4b;break;
        case 0xd2: P2 = 0x4c;break;
case 0x5c: P2 = 0x4d;break;
        case 0x8c: P2 = 0x4e;break;
case 0x22: P2 = 0x4f;break;
        case 0xb2: P2 = 0x50;break;
case 0xa8: P2 = 0x51;break;
        case 0xb4: P2 = 0x52;break;
case 0xd8: P2 = 0x53;break;
        case 0x34: P2 = 0x54;break;
case 0x3c: P2 = 0x55;break;
        case 0x54: P2 = 0x56;break;
case 0xb8: P2 = 0x57;break;
        case 0x44: P2 = 0x58;break;
case 0xac: P2 = 0x59;break;
        case 0x58: P2 = 0x5a;break; //"Z"
case 0x68: P2 = 0x21;break; // "1"
        case 0x78: P2 = 0x40;break;
case 0x64: P2 = 0x23;break;
        case 0xa4: P2 = 0x24;break;
case 0x74: P2 = 0x25;break;
        case 0x6c: P2 = 0x5e;break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 0xbc: P2 = 0x26;break;
        case 0x7c: P2 = 0x2a;break;
case 0x62: P2 = 0x28;break; // "9"
        case 0xa2: P2 = 0x29;break; // "0"
//case 0x5a: P2 = 0x55;break; //enter
        case 0x94: P2 = 0x20;break; //space
        case 0x72: P2 = 0x5f;break; // "จ" // _
case 0xaa: P2 = 0x2b;break; // "ข" // +
        case 0xba: P2 = 0x7c;break; // "จ" // |
        case 0x82: P2 = 0x3c;break; // "ม" // <
case 0x92: P2 = 0x3e;break; // "ำ" // >
        case 0x52: P2 = 0x3f;break; // "ฟ" // ?
case 0x32: P2 = 0x3a;break; // "ว" // :
        case 0x4a: P2 = 0x22;break; // "ง" // "
case 0x2a: P2 = 0x7b;break; // "บ" // {
        case 0xda: P2 = 0x7d;break; // "ด" // }
        case 0x0c: P2 = 0x30;break; // "0" //numlock
        case 0x96: P2 = 0x31;break; // "1"
        case 0x4c: P2 = 0x32;break;
case 0x5e: P2 = 0x33;break;
        case 0xd6: P2 = 0x34;break;
case 0xce: P2 = 0x35;break;
        case 0x2c: P2 = 0x36;break;
case 0x36: P2 = 0x37;break;
        case 0xac: P2 = 0x38;break;
case 0xbe: P2 = 0x39;break; // "9"
}
}
}
if(flag3 == 1)
{
    if(flag1 == 0) //Thai (ฟหกค)
    {
        switch(P1)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
case 0x38: P2 = 0x1f;break; // "P" //A
        case 0x4c: P2 = 0x34;break; //B
case 0x84: P2 = 0x41;break; //C
        case 0xc4: P2 = 0x01;break; //D
case 0x24: P2 = 0x33;break; //E
        case 0xd4: P2 = 0x14;break; //F
case 0x2c: P2 = 0x40;break; //G
        case 0xcc: P2 = 0x49;break; //H
case 0xc2: P2 = 0x23;break; //I
        case 0xdc: P2 = 0x48;break; //J
case 0x42: P2 = 0x32;break; //K
        case 0xd2: P2 = 0x2a;break; //L
case 0x5c: P2 = 0x17;break; //M
        case 0x8c: P2 = 0x37;break; //N
case 0x22: P2 = 0x19;break; //O
        case 0xb2: P2 = 0x22;break; //P
case 0xa8: P2 = 0x46;break; //Q
        case 0xb4: P2 = 0x1e;break; //R
case 0xd8: P2 = 0x2b;break; //S
        case 0x34: P2 = 0x30;break; //T
case 0x3c: P2 = 0x35;break; //U
        case 0x54: P2 = 0x2d;break; //V
case 0xb8: P2 = 0x44;break; //W
        case 0x44: P2 = 0x1b;break; //X
case 0xac: P2 = 0x31;break; //Y
        case 0x58: P2 = 0x1c;break; // "P" //Z
case 0x68: P2 = 0x45;break; // "J" //1
        //case 0x78: P2 = 0x32;break; //2
//case 0x64: P2 = 0x33;break; //3
        case 0xa4: P2 = 0x20;break; //4
case 0x74: P2 = 0x16;break; //5
        case 0x6c: P2 = 0x38;break; //6
case 0xbc: P2 = 0x36;break; //7

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        case 0x7c: P2 = 0x04;break;      //8
case 0x62: P2 = 0x15;break;    // "ค"//9
        case 0xa2: P2 = 0x08;break;    // "จ"//0
        case 0x72: P2 = 0x02;break;    // "ช" // -
case 0xaa: P2 = 0x0b;break;    // "ง" // =
        case 0xba: P2 = 0x03;break;    // "ข" //\
        case 0x82: P2 = 0x21;break;    // "ม"// ,
case 0x92: P2 = 0x43;break;    // "ใ"// .
        case 0x52: P2 = 0x1d;break;    // "ฟ"// /
case 0x32: P2 = 0x27;break;    // "จ"// ;
        case 0x4a: P2 = 0x07;break;    // "ง"// '
case 0x2a: P2 = 0x1a;break;    // "ฎ"// [
        case 0xda: P2 = 0x25;break;    // "ค"// ]
//case 0x5a: P2 = 0x55;break;        //enter
        case 0x94: P2 = 0x20;break;    //space
        case 0x0e: P2 = 0x30;break;    // "0"
        case 0x96: P2 = 0x31;break;    // "1"
        case 0x4e: P2 = 0x32;break;
case 0x5e: P2 = 0x33;break;
        case 0xd6: P2 = 0x34;break;
case 0xce: P2 = 0x35;break;
        case 0x2e: P2 = 0x36;break;
case 0x36: P2 = 0x37;break;
        case 0xae: P2 = 0x38;break;
case 0xbe: P2 = 0x39;break;    // "9"
    }
}

    else if(flag1 == 1)          //Thai (ฤๅญ โ)
    {
        switch(P1)
        {
            case 0x38: P2 = 0x24;break;    // "ฤ"//A
            case 0x4c: P2 = 0x3a;break;    //B
            case 0x84: P2 = 0x09;break;    //C

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        case 0xc4: P2 = 0x0f;break;    //D
case 0x24: P2 = 0x0e;break;    //E
        case 0xd4: P2 = 0x42;break;    //F
case 0x2c: P2 = 0x0c;break;    //G
        case 0xcc: P2 = 0x47;break;    //H
case 0xc2: P2 = 0x13;break;    //I
        case 0xdc: P2 = 0x4b;break;    //J
case 0x42: P2 = 0x29;break;    //K
        case 0xd2: P2 = 0x28;break;    //L
//case 0x5c: P2 = 0x4d;break;    //M
        case 0x8c: P2 = 0x4c;break;    //N
case 0x22: P2 = 0x2f;break;    //O
        case 0xb2: P2 = 0x0d;break;    //P
case 0xa8: P2 = 0x50;break;    //Q
        case 0xb4: P2 = 0x11;break;    //R
case 0xd8: P2 = 0x06;break;    //S
        case 0x34: P2 = 0x18;break;    //T
case 0x3c: P2 = 0x4a;break;    //U
        case 0x54: P2 = 0x2e;break;    //V
//case 0xb8: P2 = 0x57;break;    //W
//case 0x44: P2 = 0x;break;    //X
case 0xac: P2 = 0x4d;break;    //Y
//case 0x58: P2 = 0x5a;break;    //"(" //Z
//case 0x68: P2 = 0x31;break;    //"+"//1
        case 0x78: P2 = 0x51;break;    //2
case 0x64: P2 = 0x52;break;    //3
        case 0xa4: P2 = 0x53;break;    //4
case 0x74: P2 = 0x54;break;    //5
        case 0x6c: P2 = 0x39;break;    //6
case 0xbc: P2 = 0x3f;break;    //7
        case 0x7c: P2 = 0x55;break;    //8
case 0x62: P2 = 0x56;break;    //"จ"//9
        case 0xa2: P2 = 0x57;break;    //"ฉ" //0
        case 0x72: P2 = 0x58;break;    //"ช" // -

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 0xaa: P2 = 0x59;break; // "จ" // =
        case 0xba: P2 = 0x05;break; // "ข" // \
        case 0x82: P2 = 0x12;break; // "ฃ" // ,
case 0x92: P2 = 0x2c;break; // "ค" // .
        case 0x52: P2 = 0x26;break; // "ฅ" // /
case 0x32: P2 = 0x0b;break; // "ง" // ;
        //case 0x4a: P2 = 0x07;break; // "ฉ" // '
case 0x2a: P2 = 0x10;break; // "บ" // [
        //case 0xda: P2 = 0x25;break; // "ด" // ]
//case 0x5a: P2 = 0x55;break; //enter
        case 0x94: P2 = 0x20;break; //space
        case 0x0e: P2 = 0x30;break; // "0"
        case 0x96: P2 = 0x31;break; // "1"
        case 0x4e: P2 = 0x32;break;
case 0x5e: P2 = 0x33;break;
        case 0xd6: P2 = 0x34;break;
case 0xce: P2 = 0x35;break;
        case 0x2e: P2 = 0x36;break;
case 0x36: P2 = 0x37;break;
        case 0xae: P2 = 0x38;break;
case 0xbe: P2 = 0x39;break; // "9"
;
;
;
compare = 0;
;
;

```

```
void ext0_service(void) interrupt 0 {
```

```

    b00 = key_lo&bit7;
    key_hi = (key_hi<<1) | b00;
    key_lo = (key_lo<<1) | key_data;
    if(!(key_hi&bit2))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
key_hi = 0xff;
    key_lo = 0xff;
}
if(key_hi == 0xfe)
{
if(key_lo == 15)
{
    key_rev = 1;
}
else
{
if(key_rev)
{
key_head = (key_head+1)&0x07;
    data_key[key_head] = key_lo;
    key_rev = 0;
}
}
}
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

โค้ดของโปรแกรมในส่วนของเซิร์ฟเวอร์(Server)

โค้ดของโปรแกรมส่วนของเซิร์ฟเวอร์เขียนจากภาษาเบสิก

Public bOK As Boolean

Public bError As Boolean

Private Sub Wait()

Dim Start

Start = Timer

Do While Timer < Start + 5

DoEvents

If bOK Then

Exit Sub

End If

If bError Then

Exit Sub

End If

Loop

End Sub

Private Sub MOBILE_Click()

Dim bufferOut As String

Dim bufferIn As String

Dim Buf_Len1 As String

Dim Buf_Len2 As String

Dim Buffer_OA As Variant

Dim Buffer_SMS As Variant

Dim Check_len As Integer

Dim data_sms1 As String

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Dim data_sms2 As String
Dim OA As String
Dim Len_sms As Integer
Dim Len_sms2 As Integer
Dim j, k As Integer

```

```

N = Time$
sec = Second(Time)
buf_sec = sec + 5

```

```
Do
```

```
sec = Second(Time)
```

```
Text5.Text = sec
```

```
If buf_sec >= 60 Then
```

```
buf_sec = buf_sec - 60
```

```
End If
```

```
If sec >= buf_sec Then
```

```
Text1.Text = ""
```

```
Text2.Text = ""
```

```
buf_sec = sec + 5
```

```
MSComm1.Output = "AT+CMGL=0" & Chr$(13)
```

```
Do
```

```
bufferIn = ""
```

```
DoEvents
```

```
Do
```

```
bufferIn = bufferIn & MSComm1.Input
```

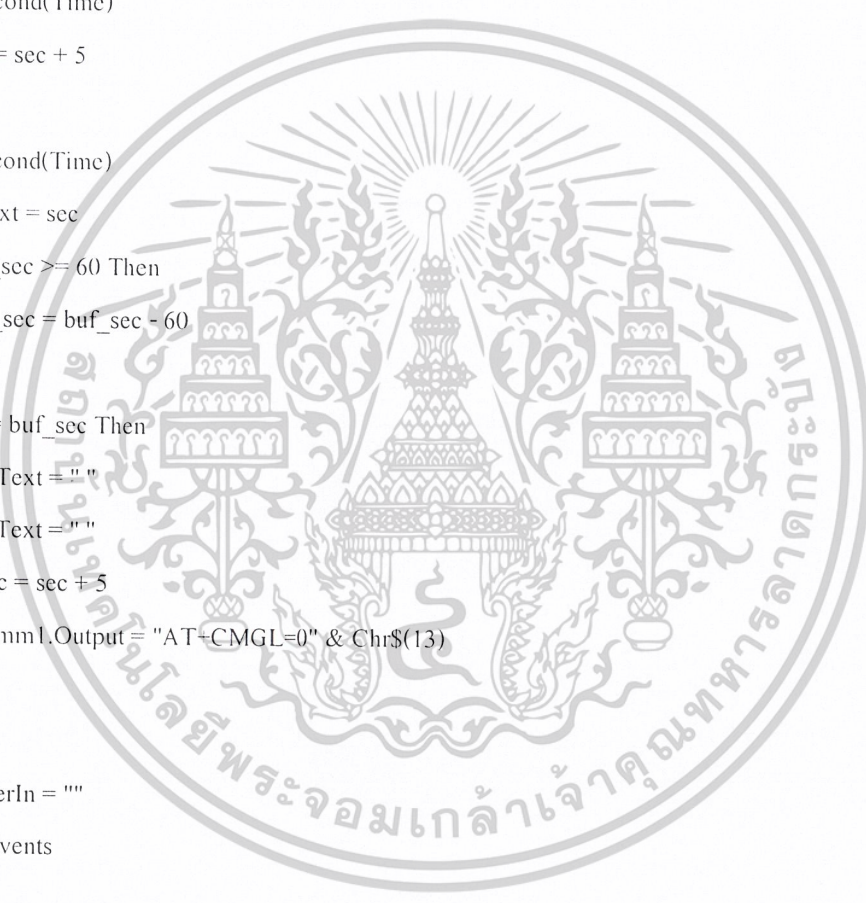
```
Loop Until InStr(bufferIn, "OK" & vbCrLf)
```

```
Loop Until bufferIn <> ""
```

```
Check_len = Len(bufferIn)
```

```
If Check_len < 17 Then
```

```
Wait
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        Text3.Text = "NO SMS Incoming"
    End If
End If

Loop Until Check_len > 17
Text3.Text = "SMS Incoming"
'----- OA Number -----
OA = "0"
For i = 51 To 58
    data_sms1 = ""
    data_sms1 = Mid(bufferIn, i + 1, 1)
    data_sms2 = ""
    data_sms2 = Mid(bufferIn, i, 1)
    OA = OA & data_sms1
    OA = OA & data_sms2
    i = i + 1
Next
Text4.Text = OA
Text2.Text = "[PDU TYPE]" & vbCrLf & bufferIn & vbCrLf
'---- SMS PDU Mode
OA = OA & "F"
'----- PDU OA Number -----
PDU_OA = ""
For i = 1 To 10
    data_sms1 = ""
    data_sms1 = Mid(OA, i + 1, 1)
    data_sms2 = ""
    data_sms2 = Mid(OA, i, 1)
    PDU_OA = PDU_OA & data_sms1
    PDU_OA = PDU_OA & data_sms2
    i = i + 1
Next

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

'----- Decode SMS -----'

Buf_Len1 = Mid(bufferIn, 77, 1)

Buf_Len2 = Mid(bufferIn, 78, 1)

Select Case Buf_Len1

Case "A"

j = 16 * 10

Case "9"

j = 16 * 9

Case "8"

j = 16 * 8

Case "7"

j = 16 * 7

Case "6"

j = 16 * 6

Case "5"

j = 16 * 5

Case "4"

j = 16 * 4

Case "3"

j = 16 * 3

Case "2"

j = 16 * 2

Case "1"

j = 16 * 1

Case "0"

j = 16 * 0

End Select

Select Case Buf_Len2

Case "F"



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

k = 15
 Case "E"
 k = 14
 Case "D"
 k = 13
 Case "C"
 k = 12
 Case "B"
 k = 11
 Case "A"
 k = 10
 Case "9"
 k = 9
 Case "8"
 k = 8
 Case "7"
 k = 7
 Case "6"
 k = 6
 Case "5"
 k = 5
 Case "4"
 k = 4
 Case "3"
 k = 3
 Case "2"
 k = 2
 Case "1"
 k = 1
 Case "0"
 k = 0



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End Select

Len_sms = j + k

Buffer_SMS = ""

Buffer_SMS = Mid(bufferIn, 79, (Len_sms * 2) - 2)

'----- Send SMS -----'

MSComm1.Output = "AT+CMGF=0" & Chr\$(13)

Do

bufferIn = ""

DoEvents

Do

bufferIn = bufferIn & MSComm1.Input

Loop Until InStr(bufferIn, "OK" & vbCrLf)

Loop Until bufferIn <> ""

Out_SMS = "0011000981" + PDU_OA

If Buffer_SMS = "EEF73A1D9ECD6230" Then

'----- nokia3310

Text8.Text = "Nokia 3310"

Out_SMS = Out_SMS + "0000A719CEF73A1D06CD663118A8E703D568A0831B34A7BFC76B"

ElseIf Buffer_SMS = "D374B95D76CF41C3592D0D" Then

'----- Siemens C35i

Text8.Text = "Siemens C35i"

Out_SMS = Out_SMS + "0000A716D374B95D76CF413D1F485603A5DDA039FD3D5E03"

ElseIf Buffer_SMS = "CEF73A1D06CD6C3518" Then

'----- Nokia 3650

Text8.Text = "Nokia 3650"

Out_SMS = Out_SMS + "0000A719CEF73A1D06CD6C3518A8E703C570A0831B34A7BFC76B"

ElseIf Buffer_SMS = "D3707B5E779F41C51B0C06" Then

'----- Samsung E700

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Text8.Text = "Samsung E700"
Out_SMS = Out_SMS +
"0000A71BD3707B5E779F41C51B0C06EAF940321A28ED06CDE9EFF11A"
End If
Len_sms2 = (Len(Out_SMS) / 2) + 2
Text1.Text = "๓๕ÒÁÀÒÇ =" & Len_sms2 & " octets" & vbCrLf
MSComm1.Output = "AT+CMGS=" & Len_sms2 & Chr$(13)
Do
    bufferIn = ""
    DoEvents
    bufferIn = bufferIn & MSComm1.Input
Loop Until bufferIn <> ""
MSComm1.Output = Out_SMS
Wait
MSComm1.Output = vbCrLf
Wait
MSComm1.Output = Chr$(26)
Text1.Text = Text1.Text & Out_SMS & vbCrLf
Wait
MOBILE_Click
End Sub

Private Sub TYPE_MOBILE_Click()
Dim buffer_Cgmm As Variant
Dim buffer_AT As Variant
    Wait
    MSComm1.Output = "AT" & Chr$(13)
    Wait
    buffer_AT = ""
    buffer_AT = buffer_AT & MSComm1.Input
    If InStr(buffer_AT, "OK" & vbCrLf) Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Text6.Text = "à"×èíÁµèí"
Else
Text6.Text = "äÁëä'èà"×èíÁµèí"
End If
If Len(buffer_AT) > 3 Then
MSComm1.Output = "AT+CGMM" & Chr$(13)
buffer_Cgmm = ""
DoEvents
Do
buffer_Cgmm = buffer_Cgmm & MSComm1.Input
Loop Until InStr(buffer_Cgmm, "OK" & vbCrLf)
Text7.Text = Mid(buffer_Cgmm, 8, Len(buffer_Cgmm) - 11)
MOBILE.Enabled = True
Else
Text7.Text = "ÿÄØ³Òµéíä·ÄËÑ¼¶"
End If
buffer_AT = ""
End Sub

Private Sub Form_Load()
MSComm1.CommPort = 1
MSComm1.Settings = "19200,N,8,1"
MSComm1.PortOpen = True
'----- Check ComPort -----
If MSComm1.PortOpen = True Then
MSComm1.PortOpen = False
OpenPort1.Caption = "Connect"
TYPE_MOBILE.Enabled = False
MOBILE.Enabled = False
Else
MSComm1.PortOpen = True

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
OpenPort1.Caption = "Disconnect"  
TYPE_MOBILE.Enabled = True  
End If  
End Sub
```

```
Private Sub OpenPort1_Click()  
If MSComm1.PortOpen = True Then  
MSComm1.PortOpen = False  
OpenPort1.Caption = "Connect"  
TYPE_MOBILE.Enabled = False  
MOBILE.Enabled = False  
Else  
MSComm1.PortOpen = True  
OpenPort1.Caption = "Disconnect"  
TYPE_MOBILE.Enabled = True  
End If  
End Sub
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

PDU FORMAT

Sending a message in the PDU mode

```
AT+CMGF=0 //Set PDU mode
```

```
AT+CSMS=0 //Check if modem supports SMS commands
```

```
AT+CMGS=23 //Send message, 23 octets(excluding the two initial zeros)
```

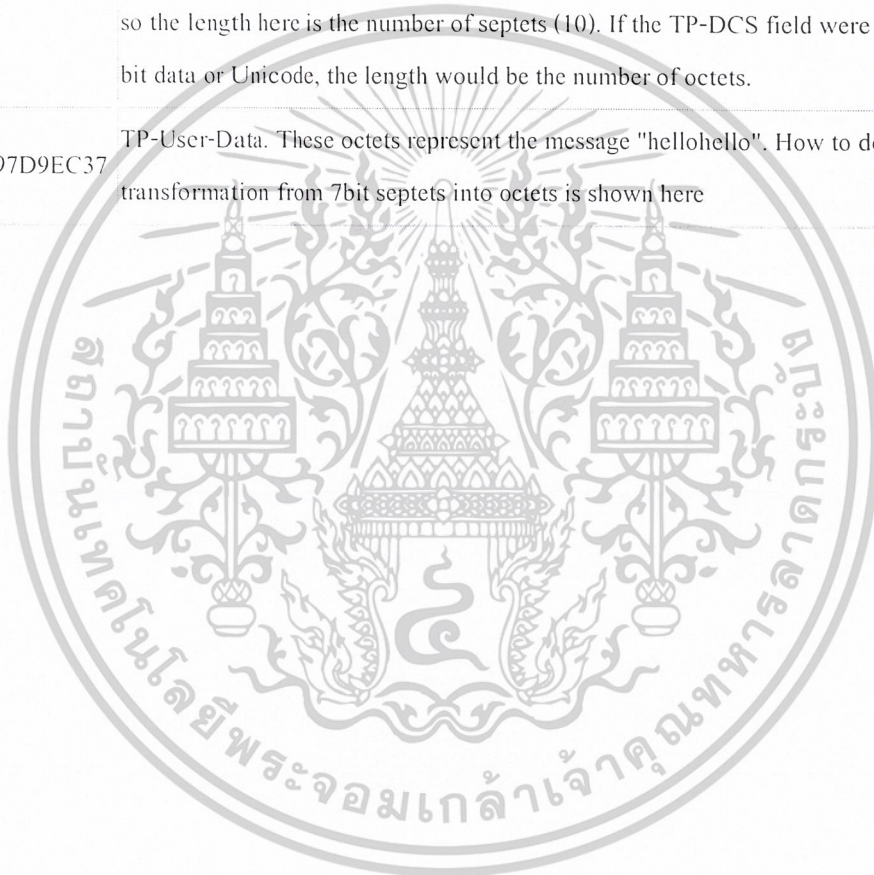
```
>0011000B916407281553F80000AA0AE8329BFD4697D9EC37
```

รูปแบบของ PDU Format นี้ประกอบไปด้วยส่วนต่างๆดังนี้

Octet(s)	Description
00	Length of SMSC information. Here the length is 0, which means that the SMSC stored in the phone should be used. <i>Note: This octet is optional. On some phones this octet should be omitted! (Using the SMSC stored in phone is thus implicit)</i>
11	First octet of the SMS-SUBMIT message.
00	TP-Message-Reference. The "00" value here lets the phone set the message reference number itself.
0B	Address-Length. Length of phone number (11)
91	Type-of-Address. (91 indicates international format of the phone number).
6407281553F8	The phone number in semi octets (46708251358). The length of the phone number is odd (11), therefore a trailing F has been added, as if the phone number were "46708251358F". Using the unknown format (i.e. the Type-of-Address 81 instead of 91) would yield the phone number octet sequence 7080523185 (0708251358). Note that this has the length 10 (A), which is even.
00	TP-PID. Protocol identifier

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

00	TP-DCS. Data coding scheme. This message is coded according to the 7bit default alphabet. Having "04" instead of "00" here, would indicate that the TP-User-Data field of this message should be interpreted as 8bit rather than 7bit (used in e.g. smart messaging, OTA provisioning etc).
AA	TP-Validity-Period. "AA" means 4 days. <i>Note: This octet is optional, see bits 4 and 3 of the first octet</i>
0A	TP-User-Data-Length. Length of message. The TP-DCS field indicated 7-bit data, so the length here is the number of septets (10). If the TP-DCS field were set to 8-bit data or Unicode, the length would be the number of octets.
E8329BFD4697D9EC37	TP-User-Data. These octets represent the message "hellohello". How to do the transformation from 7bit septets into octets is shown here



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.

คำสั่ง AT COMMAND

Manual Reference

AT Command Set

(GSM 07.07, GSM 07.05,
Siemens specific commands)

for the SIEMENS Mobile Phones

S35i, C35i, M35i

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Software Interface

1.1. Overview of the Supported AT Command Set

Commands	Function
07.07	
AT+CGMI	Issue manufacturer ID code
AT+CGMM	Issue model ID code
AT+CGMR	Output the GSM telephone version
AT+CGSN	Output the serial number (IMEI)
AT+GSN	Output the serial number (IMEI)
AT+CHUP	Terminate call
AT+CEER	Query the reason for disconnection of last call
AT+CREG	Network registration
AT+COPS	Commands concerning selection of network operator
AT+CLCK	Switch locks on and off
AT+CPWD	Change password to a lock
AT+CLIP	Display telephone number of calling party
AT+CCFC	Call forwarding
AT+CHLD	Call hold and multiparty
AT+CPAS	Query the telephone status
AT+CPIN	Enter PIN and query lock
AT+CBC	Battery charge
AT+CSQ	Output signal quality
AT+CPBS	Select a telephone book
AT+CPBW	Write a telephone-book entry
AT+CMEE	Expanded error messages according to GSM 07.07
AT+VTS	Send a DTMF tone
AT+VTD	Set duration of a DTMF tone
AT+WS46	Select wireless network
AT+CSCS	Select TE character set
AT+CAOC	Advice of charge
AT+CSSN	Supplementary service notifications
AT+CRSM	Restricted SIM access

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CIMI	Output of IMSI
AT+CACM	Accumulated call meter
AT+CAMM	Accumulated call meter maximum
AT+CLCC	List Current Calls
AT+CCLK	Clock
AT+COPN	Read operator names
AT+CPUC	Price per unit and currency table
AT+CALM	Alert sound mode
AT+CRSL	Ringer sound level
AT+CLVL	Loudspeaker volume level
AT+CMUT	Mute control
AT+CVIB	Vibrator mode
AT+CSMS	Selection of message service
AT+CPMS	Selection of SMS memory
AT+CMGF	SMS format
AT+CSCA	Address of the SMS service center
AT+CNMI	Display new incoming SMS
AT+CNMA	Acknowledgment of a short message directly output
AT+CMGL	List SMS
AT+CMGR	Read in an SMS
AT+CMGS	Send an SMS
AT+CMSS	Send an SMS from the SMS memory
AT+CMGW	Write an SMS to the SMS memory
AT+CMGD	Delete an SMS in the SMS memory
AT+CSCB	Select cell broadcast messages
AT+CMGC	Send an SMS command
Siemens-specific commands	Function
AT^SPBS	Select a telephone book (including Siemens-specific books)
AT^SDLD	Delete the "last number redial" memory"
AT^SPBC	Seek the first entry in the sorted telephone book which begins with

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	the selected (or next available) letter
AT^SPBG	Read entry from the sorted telephone book via the sorted index
AT^SLCK	Switch locks (including user-defined locks) on and off
AT^SPWD	Change password to a lock (including user-defined locks)
AT^SACM	Output ACM (accumulated call meter) and ACMmax
AT^SPLM	Read the PLMN
AT^SPLR	Read an entry from the preferred-operator list
AT^SPLW	Write an entry to the preferred-operator list
AT^SCNI	Output call number information
AT^SNFV	Set the volume
AT^SNFS	Select NF hardware
AT^SRTC	Set the ringing tone
AT^SCID	Output card ID
AT^SCKS	Output SIM card status
AT^SPIC	Output PIN counter
AT^SMGO	SMS overflow indicator
AT^SMGL	List SMS (without status change from unread to read)
AT^SMGR	Read SMS record without Changing unread->read
AT^SMSO	Switch device off
AT^SLNG	Language settings
AT^SSTK	SIM Toolkit
AT^SBNW	Binary Write
AT^SBNR	Binary Read

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2. AT Command Set

Remote control operation of the GSM mobile telephone runs via a serial interface (data cable or infrared connection), where AT+C commands according to ETSI GSM 07.07 and GSM 07.05 specification as well as several manufacturer specific AT commands are available. These commands are described in more detail later on.

The modem guideline V.25ter applies to the sequence of the interface commands.

According to this guideline, commands should begin with the character string "AT" and end with "<CR>" (= 0x0D). The input of a command is acknowledged by the display of "OK" or "ERROR". A command currently in process is interrupted by each additional character entered. This means that you should not enter the next command until you have received the acknowledgment; otherwise the current command is interrupted.

The commands supported are listed in the following tables:

1.2.1. Hayes-Standard Commands The Hayes-standard commands correspond to the commands of AT Hayes-compatible modems.

A/ Command Function	Repeat last command
AT...	Prefix for all other commands
ATA	Accept call
ATD<str>;	Dial the dialing string <str> with the voice utility Valid dial modifiers: "T" (tone dialing), "P" (pulse dialing) is ignored. The character ";" is important, for this tells the phone that the call should be set up with the voice utility. Otherwise an attempt is made to set up a data call, which the phone immediately acknowledges with "ERROR". The dial command responds with OK to the user right after starting a voice call. Other behavior like *# sequences in the dial command and also data calls remain unchanged.
ATD><n>;	Dial the telephone number from the current telephone book location number <n> The telephone book is selected with the command at+cpbs (or at^spbs).
ATD><mem><n>;	Dial the telephone number from the telephone book <mem> location number <n>
ATDL	Dial last telephone number

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ATE0	Deactivate command echo
ATE1	Activate command echo
ATH[0]	Separate connection
ATQ0	Display acknowledgments
ATQ1	Suppress acknowledgments
ATV0	Output acknowledgments as numbers
ATV1	Output acknowledgments as text
AT&F[0]	Reset to factory profile
ATZ	Set to default configuration
AT+GCAP	Output the capabilities list

1.2.2. Acknowledgments for Normal Data Communication

Response	Numeric	Meaning
OK	0	Command executed, no errors
RING	2	Ring detected
NO CARRIER	3	Link not established or disconnected
ERROR	4	Invalid command or command line too long
NO DIALTONE	6	No dial tone, dialing impossible, wrong mode
BUSY	7	Remote station busy

1.3. AT Commands and Responses According to GSM 07.07 and GSM 07.05

According to GSM, it is possible to execute an AT command in various forms.

Test command AT+CXXX=? The telephone responds by sending the list of

parameters and value ranges; these can be set using the affiliated Write command or by means of internal processes.

Read command AT+CXXX? This command tells you the current value setting of the parameter(s).

Write command AT+CXXX=<...> This command is used to set parameters that can be set.

Execute command AT+CXXX The Execute command reads non-settable parameters

which are influenced by internal processes in the telephone.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.1. AT Cellular Commands According to GSM 07.07

AT+CGMI	Issue manufacturer ID code
Test command AT+CGMI=?	Response OK
Execute command AT+CGMI	Response <manufacturer> Parameter <manufacturer> Name of manufacturer (SIEMENS) Important: There is a leading output prefix +CGMI in models before the S25.

AT+CGMM	Issue model ID code
Test command AT+CGMM=?	Response OK
Execute command AT+CGMM	Response <model> Parameter <model> Name of telephone (MOBILE) Important: There is a leading output prefix +CGMM in models before the S25.

AT+CGMR	Output the GSM telephone version
Test command AT+CGMR=?	Response OK
Execute command AT+CGMR	Response <revision> Parameter <revision> Version of the telephone software Important: There is a leading output prefix +CGMR in models before the S25.

AT+CGSN	Output the serial number (IMEI)
Test command AT+CGSN=?	Response OK
Execute command AT+CGSN	Response <sn> Parameter <sn> IMEI of the telephone Important: There is a leading output prefix +CGMI in models before the S25.

AT+GSN	Output the serial number (IMEI)
Test command AT+GSN=?	Response OK
Execute command AT+GSN	Response +GSN: <sn> Parameter <sn> IMEI of the telephone Important:The output prefix +GSN may be missing in future versions.

AT+CHUP	Terminate call
Test command AT+CHUP=?	Response OK
Execute command AT+CHUP	Response OK/ERROR
	Description: All active calls and all calls on hold are terminated.

AT+CEER	Query the reason for disconnection of last call
Test command AT+CEER=?	Response OK
Execute command AT+CEER	Response +CEER: <report> Parameter <report> Disconnection reason reported as number

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CREG	Network registration
Test command AT+CREG=?	Response +CREG: (list of supported <n>s) OK/ERROR/+CME ERROR
	Parameter <n> 0 Suppresses the unexpected network-status messages 1 Displays the unexpected network-status messages OK/ERROR/+CME ERROR
Read command AT+CREG?	Response +CREG: <n>,<stat>[,<lac>,<ci>] OK/ERROR/+CME ERROR
	Parameter <n> See Test command <stat> 0 Not checked in, not seeking 1 Checked in 2 Not checked in, but seeking a network 3 Check-in denied by network 4 Unknown 5 Registered, roaming <lac> Hexadecimal 2-byte string type of location area code <ci> Hexadecimal 2-byte string type of cell ID
Write command AT+CREG=<n>	Parameter <n> See Test command Response OK/ERROR/+CME ERROR
	Unexpected message +CREG: <stat>

AT+COPS	Commands concerning selection of network operator
Test command AT+COPS=?	Response +COPS: [[list of supported (<stat>, long alphanumeric <oper>, numeric <oper>)]s][, (list of supported <mode>s), (list of supported <format>s)] OK/ERROR/+CME ERROR Parameter <stat> 0 Unknown 1 Useful network operator 2 Used network operator 3 Prohibited network operator <oper> Operator in the format according to <mode> <mode> 0 Automatic mode 1 Manual selection of network operator 3 Setting of format 4 Automatic, manual selected <format> 0 Long alphanumeric 2 Numeric <oper>
Read command AT+COPS?	Response +COPS: <mode>[,<format>,<oper>] OK/ERROR/+CME ERROR Parameter <mode> See Test command <format> See Test command <oper> Network operator
Write command AT+COPS=<mode> [,<format>[,<oper>]]	Parameter <mode> See Test command <format> See Test command If <mode> = 1, <format> can only = 2 <oper> In numeric form only Response
	OK/ERROR/+CME ERROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CLCK	Switch locking on and off Revision to GSM 07.07 according to CR TDOC ETSI/SMG4 187/96
Test command AT+CLCK=?	Response +CLCK: (l ist of supported <fac>s) OK/ERROR/+CMEParameter ERROR <fac> "CS" Keyboard lock "PS" Phone locked to SIM (device code) "SC" SIM card (PIN) "FD" FDN lock "AO" BAOC (bar all outgoing calls) "OI" BOIC (bar outgoing international calls) "OX" BOIC-exHC (bar outgoing international calls except to home country) "AI" BAIC (bar all incoming calls) "IR" BIC-Roam (bar incoming calls when roaming outside the home country) "AB" All Barring services "AG" All outgoing barring services "AC" All incoming barring services
Write command AT+CLCK=<fac>,<mode>[,<passwd>],[,<class>]]	Parameter <fac> See Test command <mode> 0 Cancels lock 1 Activates lock 2 Queries lock status <passwd> Password <class> 1 Voice 2 Data 4 Fax 7 All classes (default value) Response If <mode>=2 and command is successful +CLCK: <status>[,<class1>][<CR><LF> +CLCK: <status>, class2....]] Parameter <status> 0 Off 1 On OK/ERROR/+CME ERROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CPWD	Change password to a lock
Test command AT+CPWD=?	Response +CPWD: list of supported (<fac>, <pwdlength>)s OK/ERROR/+CME ERROR Parameter <fac> "P2" PIN2 otherwise See Test command for AT+CLCK command, without "FD" <pwdlength> Password length
Write command AT+CPWD= <fac>, <oldpwd>, <newpwd>	Parameter <fac> <oldpwd>, <newpwd> Response See Test command for AT+CLCK command Old and new OK/ERROR/+CME password ERROR
AT+CLIP	Display telephone number of calling party
Test command AT+CLIP=?	Response +CLIP: (list of supported <n>s) OK/ERROR/+CME ERROR Parameter <n> 0 Suppresses the unexpected messages 1 Displays the unexpected messages
Read command AT+CLIP?	Response +CLIP: <n>, <m> OK/ERROR/+CME ERROR Parameter <n> See Test command <m> 0 CLIP not booked 1 CLIP booked 2 Unknown
Write command AT+CLIP=[<n>]	Parameter <n> See Read command Response OK/ERROR/+CME ERROR
	Unexpected message +CLIP: <num>,<type> Telephone number of caller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CCFC	Call forwarding
Test command AT+CCFC=?	Response +CCFC: (list of supported <reas>s) OK/ERROR/+CME ERROR Parameter <reas> 0 Always 1 If busy 2 If no answer 3 If not available 4 All reasons (0-3) 5 All conditional reasons (1-3)
Write command AT+CCFC=<reas>, <mode>[, <num> [,<type>[,<class> [,<time>]]]]	Parameter <reas> See Test command <mode> 0 Deactivate 1 Activate 2 Query 3 Install 4 Delete <num> Telephone number <type> Type of telephone number <class> 1 Voice 2 Data 4 Fax 7 All classes <time> 1-30 Time, rounded to a multiple of five seconds Response If <mode>=2 and command is successful +CCFC: <status>[, <class>[, <num>[, <type>[, <time>]]][<CR><LF>+CCFC: ...] OK/ERROR/+CME ERROR Parameter <status> 0 Not active 1 Active

AT+CHLD	Call hold and multiparty
Test command AT+CHLD=?	Response +CHLD: (list of supported <n>s) OK/ERROR/+CME ERROR
Write command AT+CHLD= [<n>]	Parameter <n> For 0 1 Terminates all held calls or sets UDUB (User terminNote: Response 1X 2 Determined User Busy) for a waiting call OK/ERROR/+CME 2X 3 Terminates all active calls (if there are any) and ating accepts the other call (waiting call or held call) Terminates call number X (X= 1-7) Puts all active calls on hold (if there are any) and accepts the other call (waiting call or held call) as active Puts all active calls except call X (X= 1-7) on hold Connects the call put on hold to the active call Terminating all calls except waiting calls is done with "AT+CHUP" Command scope depends on the SIM clearing and/or on the network support ERROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CPAS	Query the telephone status
Test command AT+CPAS=?	Response +CPAS: (list of supported <pas>s) OK/ERROR/+CMEParameter ERROR <pas> 0 Ready 3 Incoming call (phone is ringing) 4 Call is active
Execute command AT+CPAS	Response +CPAS: <pas> OK/ERROR/+CME ERROR Parameter <pas> See Test command OK/ERROR/+CME ERROR

AT+CPIN	Enter PIN and query lock
Test command AT+CPIN=?	Response OK
Read command AT+CPIN?	Response +CPIN: <code> OK/ERROR/+CME ERROR Parameter <code> READY No further input necessary SIM PIN SIM PIN input necessary SIM PUK SIM PUK input necessary PH-SIM PIN Device-code (theft protection) input necessary PH-SIM PUK Device-code PUK (theft protection) input necessary SIM PIN2 PIN2, e.g. for editing the FDN book; only possible if previous command was acknowledged with

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	SIM PUK2	+CME ERROR:17 Only possible if previous command was acknowledged with error +CME ERROR:18 message can (must) be provoked by an attempted Write command.
Write command AT+CPIN=<pin> [.<new pin>]	Parameter <pin> <new pin> Response OK/ERROR/+CME	Password for appropriate lock; if the lock is a PUK, then a <new pin> is necessary. New password for the lock ERROR

AT+CBC	Battery charge	
Test command AT+CBC=?	Response +CBC: (list of 0 1 2 3 0 1- sOK/ERROR/+CME 100 ERROR Parameter charupported <bc> <bcl> <bc>s	ME is supplied from battery ME has battery but is not supplied from there ME has no battery connected Error Battery is flat, but no more actions possible in per cent),(list of supported <bcl>s)
Execute command AT+CBC	Response +CBC: <bc>,<bcl>	

AT+CSQ	Output signal quality	
Test command AT+CSQ=?	Response +CSQ: (list of supported <rssi>s), list of supported <ber>) ERROR Parameter <rssi> <ber>	Reception level: 0 -113 dBm or less 1 -111 dBm 2-30 -109 to -53 dBm 31 -51 dBm or more 99 Unknown Bit error rate: 0-7 Like RXQUAL values from Table GSM 05.08 in Section 8.2.4 99 Unknown

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Execute command AT+CSQ	Response +CSQ: <rss>, <bOK/ERROR/+CME ERROR Parameter <rss> <ber> er>See Test command See Test command
---------------------------	---

AT+CPBS	Select a telephone book
Test command AT+CPBS=?	Response +CPBS: (list of supported <sto>s) OK/ERROR/+CME ERROR Parameter <sto> "FD" SIM fix-dialing phonebook "SM" SIM phonebook "ME" ME phonebook "DC" ME Dialed Calls List "ON" SIM (or ME) own numbers (MSISDNs) list "LD" SIM last-dialling phonebook "MC" ME missed (unanswered received) calls list "RC" ME received calls list *For description of telephone-book features, see Appendix A Note: "DC" and "LD" are never both available.
Read command AT+CPBS?	Response +CPBS: <sto> OK/ERROR/+CME ERROR Parameter <sto> See Test command
Write command AT+CPBS=<sto>	Parameter <sto> See Test command Response OK/ERROR/+CME ERROR

AT+CPBR	Read a telephone-book entry
Test command AT+CPBR=?	Response +CPBR: (list of supported <index>s), <nlength>, <tlength> OK/ERROR/+CME ERROR ERROR Parameter <index> Location number <nlength> Max. length of telephone number <tlength> Max. length of text corresponding to the number

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<p>Write command AT+CPBR= <index1> [,<index2>]</p>	<p>Response +CPBR: <index1>, <number>, <typ>, <text>[<CR><LF> +CPBR: +CPBR: <index2>, <number>, <typ>, <text>] OK/ERROR/+CME ERROR Parameter <index1> Location number where the read of the entry starts <index2> Location number where the read of the entry ends <number> Telephone number <typ> Type of number <text> Text corresponding to the telephone number NOTE: In the <text> field, there may appear special characters like ` ` (0x22), `@` (0x00), ` ` (0x08), ` ` (0x5c). (See also+CPBW and Appendix A: Special hints for using+CPBR/+CPBW command) In models before the S25, empty phonebook records are reported as follows: +CPBR: <index1>,empty In S25ff, those empty entries don't produce any output.</p>
--	---

AT+CPBW	Write a telephone-book entry
<p>Test command AT+CPBW=?</p>	<p>Response +CPBW: (list of supported <index>s), <nlength>,(list of supported <type>s), <tlength> OK/ERROR/+CME ERROR Parameter <index> Location number <nlength> Max. length of telephone number <tlength> Max. length of text corresponding to the number</p>
<p>Write command AT+CPBW= [<index>] [,<number> [,<typ>[,<text>]]]</p>	<p>Parameter <index> Location number at which the entry is written <number> Telephone number <typ> Type of number <text> Text corresponding to the telephone number Response OK/ERROR/+CME ERROR Note: The following characters in <text> must be entered via the escape sequence (see also Appendix A: Special hints for using +CPBR/+CPBW command) GSM Hex ASCII GSM Seq.(hex) Note Charchar. Esc Seq Ö 5C \ Ö5C 5C 35 43 Backslash " 22 " Ö22 5C 32 32 String delim ò 08 BSP Ö08 5C 30 38 Backspace @ 00 NULL Ö00 5C 30 30 GSM Null y cause problems on application level when using the function strlen() and should thus be represented by an escape sequence</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CMEE	Expanded error messages according to GSM 07.07
Test command AT+CMEE=?	Response +CMEE: (list of supported <n>s) Parameter <n> 0 Suppresses the expanded error format 1 Expanded error messages as number 2 Expanded error messages as text
Read command AT+CMEE?	Response +CMEE: <n> Parameter <n> See Read command
Write command AT+CMEE=<n>	Parameter <n> See Read command Response OK/ERROR/+CME ERROR
	Description: The following CME errors are possible: 0 PHONE FAILURE 1 NO CONNECTION TO PHONE 2 PH-TA LINK RESERVED 3 OPERATION NOT ALLOWED 4 OPERATION NOT SUPPORT 5 PH-SIM PIN REQUIRED 10 SIM NOT INSERTED 11 SIM PIN REQUIRED 12 SIM PUK REQUIRED 13 SIM FAILURE 14 SIM BUSY 15 SIM WRONG 15 INCORRECT PASSWORD 17 SIM PIN2 REQUIRED 18 SIM PUK2 REQUIRED 20 MEMORY FULL 21 INVALID INDEX 22 NOT FOUND

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

23	MEMORY FAILURE
24	TEXT TOO LONG
25	INV CHAR IN TEXT
26	DIAL STRING TOO LONG
27	INV CHAR IN DIAL
30	NO NETWORK SERVICE
31	NETWORK TIMEOUT
100	UNKNOWN
512	CALL BARRED BY BLACKLIST
513	PHONE LINK RESERVED
514	INVALID DIAL STRING
515	PHONE BUSY
550	PH-SIM PUK REQUIRED
551	NTF-SIM PIN REQUIRED
552	NTF-SIM PUK REQUIRED
553	PH-NET PIN REQUIRED
554	PH-NET PUK REQUIRED
555	PH-SP PIN REQUIRED
556	PH-SP PUK REQUIRED
557	PH-CP PIN REQUIRED
558	PH-CP PUK REQUIRED
559	FEATURE PIN REQUIRED
560	FEATURE PUK REQUIRED
	The following CMS errors have been defined for SMS:
300	ME failure
301	SMS service of ME reserved
302	operation not allowed
303	operation not supported
304	invalid PDU parameter
305	invalid TEXT mode
310	SIM not inserted
311	SIM PIN necessary
312	PH-SIM PIN necessary

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

313	SIM failure
314	SIM busy
315	SIM wrong
320	memory failure
321	invalid memory failure
322	memory full
330	SMSC address unknown
331	no network service
332	network timeout
340	NO +CNMA ACK EXPECTED
500	unknown error

AT+VTS	Send a DTMF tone
Test command AT+VTS=?	Response (list of supported <dtmf>s), (list of supported <duration>s) OK/ERROR+CME ERROR Parameter <dtmf> 0-9,#,*,A-D, exactly one character <duration> Duration of tone in (duration/10) seconds
Write command AT+VTS= <dtmf> [,<duration>] or AT+VTS= <dtmf-string>	Parameter <dtmf> One character from the list, see Test command<duration> See Test command <dtmf-string> max. 29 characters in quotation marks ("..."), then a duration cannot be specified Response OK/ERROR+CME ERROR Important: There is a leading output prefix +VTS in models before the S25.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+VTD	Set duration of a DTMF tone
Test command AT+VTD=?	Response +VTD: (list of supported <duration>s) OK/ERROR/+CME ERROR Parameter <duration> 1-255 Duration of tone in (duration/10) seconds
Read command AT+VTD?	Response +VTD: <duration> OK/ERROR/+CME ERROR
Write command AT+VTD= <duration>	Parameter <duration> See Test command Response OK/ERROR Important: There is a leading output prefix +VTD in models before the S25.

AT+WS46	Select wireless network
Test command AT+WS46=?	Response (list of supported <n>s) OK
Read command AT+WS46?	Response <n> OK/ERROR/+CME ERROR Parameter <n> Integer; WDS side stack 12 GSM digital cellular
Write command AT+WS46=[<n>]	Response OK/ERROR/+CME ERROR Important: There is a leading output prefix +WS46 in models before the S25.

AT+CSCS	Select TE character set
Test command AT+CSCS=?	Response +CSCS: (list of supported <chset>s) OK
Read command AT+CSCS?	Response +CSCS: <chset> OK/ERROR/+CME ERROR Parameter <chset> String; determines which TE character set is used
Write command AT+CSCS= [<chset>]	Response OK/ERROR/+CME ERROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CAOC	Advice of charge
Test command AT+CAOC=?	Response +CAOC: (list of supported <mode>s) Parameter <mode> 0 query CCM value
Read command AT+CAOC?	Response +CAOC: <mode> Parameter <mode> 0 See Test command
Write command AT+CAOC=<mode>	Response OK Parameter <mode> 0 See Test command
Execute command AT+CAOC	Response +CAOC: <ccm> OK/ERROR/+CME ERROR Parameter <ccm> Updated hexadecimal call meter, measured in home units, coding analogous to ACMmax on the SIM

AT+CSSN	Supplementary service notifications Revision according to GSM 07.07 Version 5.0.0
Test command AT+CSSN=?	Response +CSSN: (list of supported <n>s), (list of supported <m>s) Parameter <n> 0 Suppresses the +CSSI messages 1 Activates the +CSSI messages <m> 0 Suppresses the +CSSU messages 1 Activates the +CSSU messages For supported +CSSI/+CSSU messages, see also 1.3.4. Summary of All UnexpectedMessages
Read command	Response

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CSSN?	+CSSN: <n>,<m> Parameter <n> See Test command <m> See Test command
Write command AT+CSSN=<n>[,<m>]	Parameter <n> <m> See Read command See Read command
	Unexpected message +CSSI: <code1> +CSSU: <code2> Parameter <code1> Intermediate result code 3 Waiting call is pending <code2> Unsolicited result code 5 Held call was terminated

AT+CRSM	Restricted SIM access
Test command AT+CRSM=?	Response OK
Write command +CRSM=<command> [,<fileid> [,<P1>,<P2>,<P3> [,<data>]]]	Response +CRSM: <sw1>,<sw2>[,<response>] OK/ERROR/+CME ERROR Parameter <command>: 176 READ BINARY 178 READ RECORD 192 GET RESPONSE 214 UPDATE BINARY 220 UPDATE RECORD 242 STATUS <fileid>: Integer, identifier of the data file on the SIM, mandatory for every command except STATUS (see GSM 11.11) <P1>, <P2>, <P3>: Integer, transferal parameter from ME to SIM, mandatory for every command except GET RESPONSE,STATUS (see GSM 11.11) <data>: Hexadecimal string; information that is to be written to the SIM <sw1>, <sw2>: Integer; information from the SIM as to how/whether the command was executed <response>: Hexadecimal string; given when a command was successfully processed Note: The write access to CK boxes receives only limited support and differs from device to device.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CIMI	Output of IMSI
Test command AT+CIMI=?	Response OK
Execute command AT+CIMI	Response <imsi> Parameter <imsi> International Mobile Subscriber Identity (IMSI)

AT+CACM	Accumulated call meter
Test command AT+CACM=?	Response OK
Read command AT+CACM?	Response +CACM: <acm> OK/ERROR/+CME ERROR Parameter <acm> Accumulated call meter in hexadecimal format, measured in home units; coding analogous to ACMmax on the SIM
Write command AT+CACM=[<passwd>]	Response OK/ERROR/+CME ERROR Parameter <passwd> String type; usually PIN2

AT+CAMM	Accumulated call meter maximum
Test command AT+CAMM=?	Response OK
Read command AT+CAMM?	Response +CAMM: <acmmax> OK/ERROR/+CME ERROR Parameter <acmmax> Accumulated call meter maximum in hexadecimal format, measured in home units; coding analogous to ACMmax on the SIM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Write command AT+CMM=	Response OK/ERROR/+CME ERROR
[<acmmax>	Parameter <acmmax>
[,<passwd>]]	(see Read command) String type; usually PIN2 <passwd>

AT+CLCC	List Current Calls
Test command AT+CLCC=?	Response OK
Execute command AT+CLCC	Response [+CLCC: <id1>,<dir>,<stat>,<mode>,<empty>, <number>,<type>] [<CR><LF>+CLCC: <id2>,<dir>,<stat>,<mode>,<empty>, <number>,<type> [...]] OK/ERROR/+CME ERROR Parameter <idx>: integer type; call identification number as described in GSM 02.30 [19] subclause 4.5.5.1; this number can be used in +CHLD command operations <dir>: 0 mobile originated (MO) call 1 mobile terminated (MT) call <stat>(state of the call): 0 active 1 held 2 dialing (MO call) 3 alerting (MO call) 4 incoming (MT call) 5 waiting (MT call) <mode>(bearer/teleservice): 0 voice 1 data 2 fax 3 voice followed by data, voice mode 4 alternating voice/data, voice mode 5 alternating voice/fax, voice mode 6 voice followed by data, data mode 7 alternating voice/data, data mode 8 alternating voice/fax, fax mode 9 unknown <empty>: 0 call is not one of multiparty (conference) call parties 1 call is one of multiparty (conference) call parties <number>: string type phone number in format specified by <type> <type>: type of address octet in integer format

AT+CCLK	Clock
Test command AT+CCLK=?	Response OK
Read command AT^SCLK?	Response +CCLK: <time> OK/ERROR/+CME ERROR Parameter: <time>: string type value; format is "yy/MM/dd,hh:mm:ss", where characters indicate year (two last digits), month, day, hour, minutes; E.g. 6th of May 1994, 22:10:00 hours equals

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	to „94/05/06,22:10:00"
Write command AT+CCLK=<time>	Response OK/ERROR/+CME ERROR Parameter: <time> see Test commnd

AT+COPN	Read operator names
Test command AT+COPN=?	Response OK
Execute command AT+COPN	Response +COPN:numeric <+COPN:..... oper>,long alphanumeric <oper><CR><LF> ERROR OK/ERROR/+CMEParameter Network operator in numeric and alphanumeric <oper> notation see AT^SPLM

AT+CPUC	Price per unit and currency table
Test command AT+CPUC=?	Response OK
Read command AT+CPUC?	Response +CPUC: <currency>,<ppu> OK/ERROR/+CME ERROR Parameter <currency> three-character currency code (e.g. "FRA", "DEM") <ppu> price per unit; dot is used as a decimal separator (e.g. "1.33")
Write command AT+CPUC= <currency>,<ppu>[, <passwd>]	Response OK/ERROR/+CME ERROR Parameter <passwd> String type; usually PIN2

AT+CALM	Alert sound mode
Test command AT+CALM=?	Response +CALM: (list of supported <mode>s) OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Read command AT+CALM?	Response +CALM: <mode> OK/ERROR/+CME ERROR
Write command AT+CALM=<mode>	Response OK/ERR OR/+ ERRORCME Parameter <mode> : 0 normal mode 1 silent mode (all sounds are prevented) 2 beep (only a short beep indicates an incoming call)

AT+CRSL	Ringer sound level
Test command AT+CRSL=?	Response +CRSL: (list of supported <level>s) OK
Read command AT+CRSL?	Response +CRSL: <level> OK/ERROR/+CME ERROR
Write command AT+CRSL=<level>	Response OK/ERROR/+CME ERROR Parameter <level>: Ringer Sound Level

AT+CLVL	Loudspeaker volume level
Test command AT+CLVL=?	Response +CLVL: (list of supported <level>s) OK
Read command AT+CLVL?	Response +CLVL: <level> OK/ERROR/+CME ERROR
Write command AT+CLVL=<level>	Response OK/ERROR/+CME ERROR Parameter <level>: Loudspeaker Volume Level

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.2. AT Commands According to GSM 07.05 for SMS

The GSM 07.05 commands are used for operating the SMS functions of the GSM mobile phone. The GSM module MOBILE supports the SMS PDU mode.

AT+CSMS	Selection of message service Revision according to GSM 07.05 Version 5.0.0
Test command AT+CSMS=?	Response +CSMS: (list of supported <service>) Parameter <service> 0 GSM 3.40 and 3.41 1 GSM 3.40 and 3.41 and compatibility of the AT command syntax for phase 2+ NOTE: Deactivating the phase 2+ compatibility is only possible if the direct output of short messages +CNMI=1,2 or +CNMI=1,3 is not activated. If necessary, the latter should be deactivated first.
Read command AT+CSMS?	Response +CSMS: Parameter <service> 0 <mt> 1 GSM 3.40 and 3.41 Mobile terminated messages Type supported <mo> 1 <bm> 0 Mobile originated messages Type supported Broadcast type <service> messages Type not supported e>,<mt>,<mo>,<bm>
Write command AT+CSMS= <service>	Parameter <service> 0 Response +CSMS: GSM 3.40 and 3.41 mo>,<bm> OK/ERROR/+CMS ERROR <mt>,<

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CPMS	Selection of SMS memory Revision according to GSM 07.05 Version 4.7.0
Test command AT+CPMS=?	Response +CPMS: (list of supported <mem1>s),(list of supported <mem2>s) ,(list of supported <mem3>s) Parameter <mem1> Memory from which messages are read and deleted "SM" SIM-messages memory <mem2> Memory to which messages are written and sent "SM" SIM-messages memory <mem3> Memory in which received messages are stored, if forwarding to the ("+CNMI") "SM" SIM-messages memory PC is not set
Read command AT+CPMS?	Response +CPMS: <mem1>,<used1>,<total1>,<mem2>,<used2>,<total2> ,<mem3>,<used3>,<total3> Parameter <memx> Memory from which messages are read and deleted <usedx> Number of messages currently in <memx> <totalx> Number of storable messages in <memx>
Write command AT+CPMS= <mem1> [,<mem2> [,<mem3>]]	Parameter <mem1> See Test command <mem2> See Test command <mem3> See Test command Response +CPMS: <used1>,<total1>,<used2>,<total2>,<used3>,<total3> OK/ERROR/+CMS ERROR

AT+CMGF	SMS format
Test command AT+CMGF=?	Response +CMGF: (list of supported <mode>s) Parameter <mode>: 0 PDU mode
Read command AT+CMGF?	Response +CMGF: <mode> Parameter <mode>: 0 PDU mode
Write command AT+CMGF=[<mode>]	Parameter <mode>: 0 PDU mode Response OK/ERROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CSCA	Address of the SMS service center
Test command AT+CSCA=?	Response OK
Read command AT+CSCA?	Response +CSCA: <sca>,<tosca> Parameter <sca> Service-center address in string format <tosca> Service-center address format
Write command AT+CSCA= <sca>[,<tosca>]	Parameter <sca> Service-center address in string format Service-center address <tosca> format Response OK/ERROR



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CNMI	<p>Display new incoming SMS</p> <p>Revision according to GSM 07.05 Version 4.7.0</p>
<p>Test command</p> <p>AT+CNMI=?</p>	<p>Response +CNMI: (list of supported <mode>s),(list of supported <mt>s),(list of supported <bm>s),(list of supported <ds>s),(list of supported <bfr>s) Parameter <mode> 0 Buffers unexpected messages (but is equivalent to rejecting; see <bfr>) 1 Discard indication and reject new received message unsolicited result codes when TA-TE link is reserved. Otherwise forward them directly to the TE. (only with S25ff) 2 Buffers unexpected messages if serial interface is occupied, otherwise they are output (only models before S25) <mt> 0 Suppresses unexpected messages for incoming short messages 1 Unexpected messages of a received short message (SMS-DELIVER) that is stored on a chip card are output in the form +CMTI: <mem>,<index> 2 Unexpected messages of a received short message (SMS-DELIVER) (except class 2 and the message "Waiting Indication Group: store message") are output in the form +CMT: [<alpha>],<length><CR><LF><pdu> (<alpha> is not supported) Class 2 and the message "Waiting Indication Group: store message" are output as <mt>=1 3 Unexpected messages of a received short message (SMS-DELIVER) class 3 are output as <mt>=2. Messages with other data coding schemes are output as <mt>=1. NOTE:<mt>=2 and <mt>=3 are not possible unless the Phase 2+ compatibility has been activated by means of +CSMS=1 <bm> 0 Suppresses unexpected messages for incoming cell broadcast messages 2 Outputs unexpected messages for cell broadcast messages in the form +CBM: <length><CR><LF><pdu> <ds> 0 Suppresses unexpected messages for incoming SMS status reports 2 Outputs unexpected messages for SMS status reports in the form +CDS: <length><CR><LF><pdu></p>
	<p><bfr> 1 Buffered unexpected messages are rejected when switching from <mode> 0 to <mode> 2.</p> <p><mem> See +CPMS</p> <p><index> Index of the record on the chip card alphanumeric representation of the sender address Length of <pdu> See +CMGL</p> <p><length></p> <p><pdu></p>
<p>Read command</p> <p>AT+CNMI?</p>	<p>Response +CNMI: <mode>,<mt>,<bm>,<ds>,<bfr></p> <p>Parameter</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	<p><mode> See Test command</p> <p><mt> See Test command</p> <p><bm> See Test command</p> <p><ds> See Test command</p> <p><bfr> See Test command</p>
<p>Write command</p> <p>AT+CNMI=</p> <p>[<mode></p> <p>[,<mt>[,<bm></p> <p>[,<ds>[,<bfr>]]]]]</p>	<p>Parameter</p> <p><mode> See Test command <mt> See Test command <bm> See Test command <ds> See Test command <bfr> See Test command Response OK/ERROR/+CMS ERROR</p>
	<p>Unexpected message +CMTI:</p> <p><mem>,<index> Indication that new message has arrived</p> <p>+CMT:</p> <p>,<length><CR><LF><pdu> Direct output of the short message</p> <p>+CDS:</p> <p><length><CR><LF><pdu> Direct output of the status message</p> <p>+CBM:</p> <p><length><CR><LF><pdu> report Direct output of the cell broadcast message</p>
<p>AT+CNMA</p>	<p>Acknowledgment of a short message directly output (without storing on the chip card)</p> <p>Revision according to GSM 07.05 Version 5.0.0</p> <p>(NOTE: This command is not possible unless the Phase 2+ compatibility has been activated by means of +CSMS=1)</p>
<p>Test command</p> <p>AT+CNMA=?</p>	<p>Response +CNMA: (list of supported <n>s)</p> <p>Parameter</p> <p><n> 0 Mode of functioning analogous to GSM 07.05 text mode</p>
<p>Write command</p> <p>AT+CNMA[=<n>]</p>	<p>Parameter</p> <p><n> See Test command Response OK/ERROR/+CMS ERROR: <err></p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CMGL	List SMS Revision according to GSM 07.05 Version 4.7.0
Test command AT+CMGL=?	Response +CMGL: (list of supported <stat>s) Parameter <stat> 0 "REC UNREAD": received unread messages (default) 1 "REC READ": received read messages 2 "STO UNSENT": stored unsent messages 3 "STO SENT": stored sent messages 4 "ALL": all messages
Write command AT+CMGL [=<stat>]	Parameter <stat> See Test command Response If PDU mode (+CMGF=0) and command are successful: +CMGL:<index>,<stat>,[<alpha>],<length> <CR><LF><pdu><CR><LF> +CMGL: <index>,<stat>,[<alpha>],<length> <CR><LF><pdu><CR><LF> [...]
	Parameter <pdu> The PDU begins with the service-center address (according to GSM04.11), followed by the TPDU according to GSM03.40 hexadecimal format otherwise: +CMS ERROR: <err>

in

AT+CMGR	Read in an SMS Revision according to GSM 07.05 Version 4.7.0
Test command AT+CMGR=?	Response OK
Write command AT+CMGR= <index>	Parameter <index> Index of message in selected memory <mem I> Response If PDU mode (+CMGF=0) and command are successful: +CMGR: <stat>,[<alpha>],<length><CR><LF><pdu>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Parameter	<pdu> Siehe "AT+CMGL" otherwise: +CMS ERROR: <err>
-----------	---

AT+CMGS	Send an SMS
Test command AT+CMGS=?	Response OK
Write command If PDU mode (+CMGF=0) +CMGS=<length><CR>PDU is given <ctrl-Z/ESC>	Parameter <length> Length of PDU <pdu> See "AT+CMGL" <mr> Message reference Response If sending is successful: +CMGS: <mr> If sending is not successful: +CMS ERROR: <err>

AT+CMSS	Send an SMS from the SMS memory
Test command AT+CMSS=?	Response OK
Write command +CMSS=<index>[,<da>[,<toda>]]	Parameter <index> Index of message in selected memory <mem1> <da> Destination address in string format <toda> Format of destination address <mr> Message reference Response If sending is successful: +CMSS: <mr> If sending is not successful: +CMS ERROR: <err>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CMGW	Write an SMS to the SMS memory
Test command AT+CMGW=?	Response OK
Write command If PDU mode (+CMGF=0) AT+CMGW=<length>[,<stat>]<CR>PDU is given <ctrl-Z/ESC>	Parameter <length> Length of PDU <stat> See command +CMGL <pdu> See "AT+CMGL" <index> Index of message in selected memory <mem l> Response +CMGW: <index> +CMS ERROR: <err>

AT+CMGD	Delete an SMS in the SMS memory
Test command At+CMGD=?	Response OK
Write command AT+CMGD= <index>	Parameter <index> Index of message in the selected memory <mem l> Response OK/ERROR/+CMS ERROR

AT+CSCB	Select cell broadcast messages
Test command AT+CSCB=?	Response +CSCB: (list of supported <mode>s) Parameter <mode> 0 Accepts messages that are defined in <mids> and <dcss> 1 Does not accept messages that are defined in <mids> and <dcss>
Read command AT+CSCB? Write command AT+CSCB=[< mode>[,<mids >[,<dcss>]]]	Response +CSCB: <mode>,<mids>,<dcss> Parameter <mode> See Test command <mids> String type; combinations of CBM message IDs <dcss> String type; combinations of CBM data coding schemes

AT+CMGC	Send an SMS command
Test command AT+CMGC=?	Response OK
Write command If PDU mode (+CMGF=0) +CMGC=<length><CR>PDU is given <ctrl-Z/ESC>	Parameter <length> Length of PDU <pdu> See "AT+CMGL" <mr> Message reference Response If sending is successful: +CMGC: <mr> If sending is not successful: +CMS ERROR: <err>

1.3.3. User-Defined Commands for Controlling the GSM Mobile Phone

Since user-defined commands cannot be implemented according to official syntax, the character string "+C" is replaced by "^S" ("^" = 0x5E). For the future: if a user-defined command is accepted in the same syntax in GSM recommendations, the command can be addressed using both command strings.

AT^SPBS	Select a telephone book (including Siemens-specific books)
Test command AT^SPBS=?	Response ^SPBS: (list of supported <sto>s) OK/ERROR/+CME ERROR Parameter <sto> "FD" SIM fix-dialing telephone book "SM" SIM telephone book "ME" Telephone book in device "DC" ME Dialed Calls List "ON" Own telephone numbers "LD" SIM last dialing number "MC" ME Missed Calls List "RC" ME Received Calls List "MD" Last number redial memory in telephone device "OW" Own numbers "BD" Barred dialing numbers "SD" Service dialing numbers "MS" Missed dialing numbers (unanswered calls) "CD" Callback dialing numbers (answered calls) "BL" Blacklist dialing numbers (barred numbers from remote) "MB" Mailbox dialing numbers (network-operator mailbox) "CS" Common sortable telephone book (sorted combination of "SM", "ME", "FD"; access only via ^SPBC, ^SPBG) "RD" Red book (all entries in "CS" whose name portions have an exclamation point (!) as the final character) *For a description of the telephone-book features, see Appendix A
Read command	Response

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SPBS?	^SPBS: <sto> OK/ERROR/+CME ERROR Parameter <sto> See Test command
Write command AT^SPBS= <sto>	Parameter <sto> See Test command Response OK/ERROR/+CME ERROR

AT^SDLD	Delete the "last number redial" memory
Test command AT^SDLD=?	Response OK
Execute command AT^SDLD	Response OK/ERROR/+CME ERROR

AT^SPBC	Seek the first entry in the sorted telephone book which begins with the selected (or next available) letter
Test command AT^SPBC=?	Response ^SPBC: (list of sorted telephone books supported <mem>s) See AT+CPBS/AT^SPBS OK/ERROR/+CME ERROR
Write command AT^SPBC=<char>	Parameter <char> First letter of sought entry "A" to "Z" (with any other character, the index of the first entry that begins with a special character is sent back) <index> Response Index in the sorted telephone book (access via AT^SPBG) ^SPBC: <index> OK/ERROR/+CME ERROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ภายนอก
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ .

AT^SPBG	Read entry from the sorted telephone book via the sorted index
Test command AT^SPBG=?	Response ^SPBG: (list of supported <index>s), <nlength>, <tlength> OK/ERROR/+CME ERROR Parameter <index> Location number <nlength> Max. length of telephone number <tlength> Max. length of the text corresponding to the number
Write command AT^SPBG= <index1> [, <index2>]	Response ^SPBG: <index1>, <number>, <typ>, <text>[<CR><CL> ^SPBG: ^SPBG: <index2>, <number>, <typ>, <text>] OK/ERROR/+CME ERROR Parameter <index1> Location number where the read of the entry starts <index2> Location number where the read of the entry ends <number> Telephone number <typ> Type of number <text> Text corresponding to the telephone number

AT^SLCK	Switch locks (including user-defined locks) on and off
Test command AT^SLCK=?	Response ^SLCK: (list of supported <fac>s) OK/ERROR/+CME Parameter ERROR <fac> "PS" Phone locked to SIM (device code) "SC" SIM card (PIN) "FD" FDN lock "AO" BAOC (bar all outgoing calls) "OI" BOIC (bar outgoing international calls) "OX" BOIC-exHC (bar outgoing international calls except to home country) "AI" BAIC (bar all incoming calls) "IR" BIC-Roam (bar incoming calls when roaming outside the home country) "AB" All barring services "AG" All outgoing barring services

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	<p>"AC" All incoming barring services</p> <p>"PN" Network personalization (GSM 02.22)</p> <p>"PC" Corporate personalization (GSM 02.22)</p> <p>"PU" Network subset personalization (GSM 02.22)</p> <p>"PP" Service provider personalization (GSM 02.22)</p> <p>"PF" Phone locked to very first inserted SIM</p>
<p>Write command</p> <p>AT^SLCK =</p> <p><fac>, <mode></p> <p>[,<passwd></p> <p>[,<class>]]</p>	<p>Parameter <fac> See Test command <mode> 0 Cancels lock 1 Activates lock 2 Queries lock status <passwd> Password <class> 1 Voice 2 Data 4 Fax 7 All classes (default value)</p> <p>Response If <mode>=2 and command is successful ^SLCK:</p> <p><status>[,<class1>[<CR><LF> ^SLCK: <status>, class2....]] Parameter <status> 0 Off 1</p> <p>On OK/ERROR/+CME ERROR</p>

AT^SPWD	Change password to a lock (including user-defined locks)
<p>Test command</p> <p>AT^SPWD=?</p>	<p>Response</p> <p>^SPWD: list of supported (<fac>, <pwdlength>)s</p> <p>OK/ERROR/+CME ERROR</p> <p>Parameter</p> <p><fac> "P2" PIN2</p> <p>otherwise See Test command for the command AT^SLCK, without "FD"</p> <p><pwdlength> Length of password</p>
<p>Write command</p> <p>AT^SPWD =</p> <p><fac>,<oldpwd>,<newpwd></p>	<p>Parameter <fac> See Test command for the command AT^SLCK Old</p> <p><oldpwd>, <newpwd> and new password</p> <p>Response</p> <p>OK/ERROR/+CME ERROR</p>

AT^SACM	Output ACM (accumulated call meter) and ACMmax
<p>Test command</p> <p>AT^SACM=?</p>	Response ^SACM: (list of supported <n>s)
<p>Execute command</p> <p>AT^SACM</p>	<p>Response ^SACM: <n>,<acm>,<acm_max> OK/ERROR/+CME ERROR Paramter</p> <p><n> See Test command <acm> Accumulated call meter <acm_max> Maximum accumulated call meter</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Write command AT^SACM=<n>	Parameter <n> 0 Suppresses the unexpected message 1 Outputs the unexpected message
------------------------------	---

AT^SPLM	Read the PLMN list
Test command AT^SPLM=?	Response OK
Execute command AT^SPLM	Response ^SPLM:numeric <oper>,long alphanumeric <oper><CR><LF> ^SPLM:..... OK/ERROR/+CME ERROR Parameter <oper> Network operator in numeric and alphanumeric notation

AT^SPLR	Read an entry from the preferred-operator list
Test command AT^SPLR=?	Response ^SPLR: (list of supported <index>s) OK/ERROR/+CME ERROR Parameter <index> Location numbers
Write command AT^SPLR=<index1> [, <index2>]	Response ^SPLR: <index1>, numeric <oper> ^SPLR: ^SPLR: <index2>, numeric <oper> OK/ERROR/+CME ERROR Parameter <index1> Location number where the read of the entry starts <index2> Location number where the read of the entry ends <oper> Network operator in numeric form

AT^SPLW	Write an entry to the preferred-operator list
Test command AT^SPLW=?	Response ^SPLW: (list of supported <index>s) OK/ERROR/+CME ERROR Parameter <index> Location number

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Write command AT^SPLW=<index>[, <oper>]	Parameter <index> Location number at which the entry is written <oper> Network operator in numeric form Response OK/ERROR/+CME ERROR
---	---

AT^SCNI	Output call number information
Test command AT^SCNI=?	Response OK
Execute command AT^SCNI	Response ^SCNI: 1[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 2[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 3[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 4[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 5[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 6[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 7[,<cs>[,<number>,<type>]] OK/ERROR/+CME ERROR Parameter <cs> Call status of affiliated call number (first parameter) 0 Call on hold 1 Active call 2 Waiting call <number> Telephone number <type> Type of number

AT^SNFV	Set the volume
Test command AT^SNFV=?	Response ^SNFV: (list of supported <vol>s) Parameter <vol> Value range of volume (0 to 4) (0 low, ..., 4 max. volume; approx. 3 dB/level)
Read command AT^SNFV?	Response ^SNFV: <vol> Parameter <vol> See Test command
Write command AT^SNFV=<vol>	Parameter <vol> See Test command Response OK/ERROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SNFS	Select NF hardware
Test command AT^SNFS=?	Response ^SNFS: (list of supported <dev>s) Parameter <dev> 0 Cell phone mode 1 Handsfree
Read command AT^SNFS?	Response ^SNFS: <dev> Parameter <dev> See Test command Note: Volume should be temporarily set to „0“ before NF hardware is changed.
Write command AT^SNFS=<dev>	Parameter <dev> See Test command Response OK/ERROR

AT^SRTC	Set the ringing tone
Test command AT^SRTC=?	Response ^SRTC: (list of supported <type>s), (list of supported <vol>s) Parameter <type> I-X Number of ringing tone 0 Mutes the ringing tone; when MTC is set, the phone does not ring and the volume is ignored <vol> 0-Y Volume of ringing tone
Read command AT^SRTC?	Response ^SRTC: <type>, <vol>, <ringing> Parameter <type> See Test command <vol> See Test command <ringing> 0 Test-ring is switched off 1 Test-ring is switched on
Write command AT^SRTC=[<type>],[<vol>]	Parameter See Test command See Test command <type> <vol> Response

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	OK/ERROR
Execute command AT^SRTC	Response The ringing tone sounds on the current NF device; it is selected using "AT+CNFS" until AT^SRTC is called up again OK/ERROR/+CME ERROR Note:If an MTC arrives while the test-ring is active, the latter is switched off and the "normal" ring is switched on.

AT^SCID	Output card ID
Test command AT^SCID=?	Response OK/ERROR/+CME ERROR
Execute command AT^SCID	Response ^SCID: <cid> OK/ERROR/+CME ERROR Parameter <cid> Number of SIM card

AT^SCKS	Output SIM card status
Test command AT^SCKS=?	Response ^SCKS: (list of supported <n>s) Parameter <n> 0 Suppresses the unexpected messages 1 Outputs the unexpected messages
Read command AT^SCKS?	Response ^SCKS: <n>, <m> Parameter <m> 0 No card 1 Card in card reader
Write command AT^SCKS=<n>	Parameter <n> See Test command Response OK/ERROR
	Unexpected

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	message ^SCKS: <m> See Read command
--	--

AT^SPIC	Output PIN counter
Test command AT^SPIC=?	Response OK/ERROR/+CME ERROR
Execute command AT^SPIC	Response ^SPIC: <counter> OK/ERROR/+CME ERROR Parameter <counter> Number of tries still available to enter the <passwd. The command "AT+CPIN?" must be used to check which password is currently needed.

AT^SMGO	SMS overflow indicator
Test command AT^SMGO=?	Response ^SMGO: (list of supported <n>s) OK/ERROR/+CME ERROR Parameter <n> 0 Disable 1 Enable
Read command AT^SMGO?	Response ^SMGO: <n>,<mode> OK/ERROR/+CME ERROR
	Parameter <n> See Test command <mode> 0 Space still available 1 SMS buffer is full (chip card) 2 Buffer is full and new message that should be sent to the telephone is present in the SC
Write command AT^SMGO=<n>	Parameter <n> See Test command Response OK/ERROR/+CME ERROR
	Unexpected message ^SMGO: <mode> See Read command

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SMGL	List SMS (without status change from unread to read) Revision according to GSM 07.05 Version 4.7.0
Test command AT^SMGL=?	<p>Response</p> <p>^SMGL: (list of supported <stat>s)</p> <p>Parameter</p> <p><stat></p> <p>0 "REC UNREAD": received unread messages (default)</p> <p>1 "REC READ": received read messages</p> <p>2 "STO UNSENT": stored unsent messages</p> <p>3 "STO SENT": stored sent messages</p> <p>4 "ALL": all messages</p>
Write command AT^SMGL [=<stat>]	<p>Parameter</p> <p><stat> See Test command</p> <p>Response</p> <p>If PDU mode (+CMGF=0) and command is successful:</p> <p>^SMGL: <index>,<stat>,[<alpha>],<length></p> <p><CR><LF><pdu></p> <p>[<CR><LF>^SMGL: <index>,<stat>,[alpha],<length></p> <p><CR><LF><pdu></p> <p>[..]]</p>
	<p>Parameter</p> <p><pdu> The PDU begins with the service-center address (according to GSM04.11), followed by the TPDU according to GSM03.40 in hexadecimal format</p> <p>otherwise:</p> <p>+CMS ERROR: <err></p>

AT^SMGR	Read SMS record without Changing unread->read Syntax like AT+CMGR
Test command AT^SMGR=?	Response OK
Write command AT^SMGR= <index>	Parameter <index> Index of message in selected memory <mem1> Response If PDU mode (+CMGF=0) and command are successful: ^SMGR: <stat>,[<alpha>],<length><CR><LF><pdu> Parameter <pdu> Siehe "AT+CMGL" otherwise: +CMS ERROR: <err>

AT^SMSO	Switch device off
Test command AT^SMSO=?	Response OK
Execute command AT^SMSO	Response OK Device switches off

AT^SLNG	Language settings
Test command AT^SLNG=?	Response ^SLNG: (list of supported languages <lng>s) Parameter: <lng>: Integer; language coded according to GSM 03.38 or mobile-specific language (>100)
Read command AT^SLNG?	Response ^SLNG: <lng>
Write command AT^SLNG=<lng>	Response OK/ERROR/+CME ERROR

AT^SSTK	SIM Toolkit
Test command AT^SSTK=?	Response ^SSTK: <profile> Parameter: <profile> ME profile according to GSM 11.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<p>Write command AT^SSTK=<length>[,<mode>]<CR>PDU is given<ctrl-Z/ESC></p>	<p>Response: OK/ERROR/CME ERROR Parameter: <length>: Length of PDU in bytes <mode>: 0: Single command 1: Sequence of commands <pdu>: SIM Toolkit commands, see GSM 11.14 Limitation: The maximum PDU length is 176 bytes.</p>
--	--

<p>AT^SBNW</p>	<p>Binary Write</p>
<p>Test command AT^SBNW=?</p>	<p>Response ^SBNW: <types>s, list of supported ((list of supported <subtypes>s)) <OK/ERROR/+CME ERROR Parameter: <type> „bmp“ bitmap Windows bitmap format without compression; 2/16/256 colours, at least 97x26 pixels 0 shown permanently when registered in home network 1 shown temporarily, deleted by more important display contents ring tones in standard MIDI format 0, without polyphony specification: http://www.midi.org 0 first (and only) entry of type “mid” „vcs“ vcal format specification: http://www.imc.org/pdi</p>
	<p><subtypes> 0 first entry of type “vcs” 1 entry of type “vcs” ... <actNumber> 0 deletes entry of the act. subtype other actual packet number</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	<maxNumber> maximum number of packets
<p>Write command</p> <p>AT^SBNW=<type>,<subtype>, [<actNumber>[, <maxNumber>]]<CR>PDU is given<ctrl-Z/ESC></p>	<p>Response OK/ERROR/+CME ERROR Parameter: <type> see Test commnd <subtype> see Test commnd <actNumber> see Test commnd</p> <p><maxNumber> see Test commnd</p> <p>Note:</p> <ul style="list-style-type: none"> -It is not possible to upload data when a call is active or in progress. If a call is active the mobile responses with +CME ERROR: PHONE BUSY and the actual upload sequence is aborted and all data packets are discarded. -If uploaded data is not useable (e.g. wrong data format) the mobile responses with +CME ERROR: INV CHAR IN TEXT after the last packet is uploaded. -To get the extended +CME-ERROR-responses AT+CMEE=2 has to be sent before. Otherwise the mobile responses only with ERROR . (see GSM07.07) -If <actNumber> and <maxNumber> during the upload are omitted, the mobile aborts the whole input sequence for the current subtype. -If <actNumber> is 0 during the upload and <maxNumber> is omitted, the mobile deletes the actual record with index <subtype> -Packets have to be uploaded in the right order! -Limitation: The maximum pdu size is 176 bytes (or 352 characters) <p>See Appendix B for examples.</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SBNR	Binary Read
Test command AT^SBNR=?	Response ^SBNR: (list of supported <types>s, (list of supported subtypes)) OK/ERROR/+CME ERROR Parameter: <type> see AT^SBNW command <subtype> see AT^SBNW command
Write command AT^SBNR=<type>,<subtype>	Response ^SBNR: <type>,<subtype>,1,<maxNumber> <CR><LF><data><CR><LF> ^SBNR: <type>,<subtype>,2,<maxNumber> <CR><LF><data><CR><LF>[...] OK/ERROR/+CME ERROR Parameter: <type> see AT^SBNW command <subtype> see AT^SBNW command <data> data in hexadecimal form (PDU) <maxNumber> see AT^SBNW command See Appendix B for examples.

1.3.4. Summary of All Unexpected Messages

Message	Meaning
+CREG: <stat>	Network registration
+CLIP: <num>	Telephone number of caller
+CMTI: <mem>,<index>	Indication of a new short message
+CMT: ,<length><CR><LF> <pdu>	Short message
+CSSI: <code1> +CSSU: <code2>	Supplementary service intermediate/unsolicited result code
^SMGO: <mode>	SMS overflow indicator
^SCKS: <m>	Message indicating whether card has been removed or inserted
^SACM: <m>	Message indicating if ACM has reached the maximum value ACMmax

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix A

Factory settings made by AT&F

ATE1 (only in case of RCCP mode)

ATQ0

ATV1

AT+CREG=0

AT+CLIP=0

AT+CRC=0

AT+CAOC=0

AT+CMEE=0

AT+CPBS=SM (if available)

AT+COPS=0

AT+VTS=1

AT+CSCS="GSM"

AT+CSSN=0,0

AT^SCKS=0

Reset pending locks (Phone Pin/Puk, Pin2/Puk2 ...)

which are give as answer of AT+CPIN?

AT+CSMS=0

AT+CNMI=0,0,0,0,1

AT^SMGO=0

AT+CSCB=0

Features of the Telephone-Book Memory

Name	Description	Category / Access	Write	Delete completely
FD	Fix-dialing number (SIM fix-dialing telephone book)	GSM 07.07 / +CPBS or ^SPBS	Allowed (PIN2 required)	
SM	Abbreviate dialing number (SIM telephone book)	GSM 07.07 / +CPBS or ^SPBS	Allowed (device code required if FDN replacement is active)	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC (MD)	Mobile last dialing number (last number redial memory; only if "LD" is not available)	GSM 07.07 / +CPBS or ^SPBS	Not allowed	By means of AT^SDLD
ON (OW)	Own Numbers (SIM own telephone numbers)	GSM 07.07 (Siemens) / +CPBS (historical)	Allowed	
LD	SIM last dialing number (last number redial memory on SIM)	GSM 07.07 / +CPBS or ^SPBS	Not allowed	By means of AT^SDLD
ME	Mobile-equipment telephone book (ME dialing numbers)	GSM 07.07 / +CPBS or ^SPBS	Allowed (device code required if FDN replacement is active)	
BD	Barred dialing numbers (blocked numbers)	Siemens / ^SPBS	Not allowed	
SD	Service dialing numbers (Service numbers)	Siemens / ^SPBS	Not allowed	
MC (MS)	Missed dialing numbers (unanswered calls)	GSM 07.07 (Siemens) / +CPBS, ^SPBS	Not allowed	
RC (CD)	Callback dialing numbers (answered calls)	GSM 07.07 (Siemens) / +CPBS, ^SPBS	Not allowed	
BL	Blacklist dialing numbers (numbers that are blocked for a certain time in order to prevent continuous accesses from remote control)	Siemens / ^SPBS	Not allowed	
MB	Mailbox dialing numbers (network-operator mailbox)	Siemens / ^SPBS	Not allowed	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CS	Common sortable numbers (sorted combination of "SM", "ME", "FD")	Siemens / ^SPBS /^SPBC / ^SPBG	Not allowed	
RD	Red book numbers ("CS" entries with '!' at the end of the name portion)	Siemens / ^SPBS /^SPBC / ^SPBG	Not allowed	

Writing to the FDN Phonebook / FDN Replacement

Writing to the fix-dialing number phonebook is protected by PIN2.

A Write sequence (to e.g. record 5) runs as follows:

```

AT+CMEE=2 //Activate expanded error message
OK
AT+CPBS=? // Listing of available telephone books
+CPBS: ("FD","SM","LD")
OK
AT+CPBS="FD" // Selection of the FDN telephone book
OK
AT+CPBW=5,"1234","test" // A Write to record 5 is attempted...
+CME ERROR: SIM PIN2 REQUIRED // ... PIN2 is required for this purpose
AT+CPIN? // Query of the PIN status...
+CPIN: SIM PIN2 // ... PIN2 is to be entered
AT+CPIN="12345678" // Input of PIN2
OK
AT+CPBW=5,"1234","test" // A Write to record 5 is attempted...
OK // PIN2 remains active as long as you use the commands
// +CPIN, +CPBS, +CPBR, +CPBW, +CACM,
// +CMM, +CPUC
// ^SPIC, ^SPBS, ^SPBC, ^SPBG,;
// If you use other commands or if none of the
// above commands are executed within five
// minutes, the validity of PIN2 is voided.
AT+CPBW=6,"5678","new test" // A Write to record 6 is attempted...
OK

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

...

In addition, if there is no FDN phonebook available on the SIM, it is possible to activate a feature which activates FDN-like behavior for the "SM" and "ME" phonebooks (FDN replacement). (Currently this feature can only be activated via the MMI lock/device lock/excluding telephone book.)

In this case, the Write to the "SM" and "ME" phonebooks is ensured by the device code (PH-SIM PIN and PH-SIM PUK, respectively).

The sequence for entering the device code is analogous to the above example.

Special hints for using +CPBR/+CPBW command

String parameters, like the <text> in +CPBW command shall be entered using quotation marks "" (Ascii=Windows=GSM=0x22). e.g. "Doe Joe"

It is possible to enter string parameters without "" but not recommended, because following problems may occur:

If no "" are used:

-SPACES (Space, Blank, Ascii=Windows=GSM=0x20) are skipped.

E.g.

at+cpbw=1,"123",K. H. results in "K.H."

at+cpbw=1,"123","K. H." SPACES are kept

-No ` ` (Ascii=Windows=GSM=0x2C) and

` ` (Ascii=Windows=GSM=0x3B) in <text> is possible, because these characters are used as separator of parameters/commands.

E.g.

at+cpbw=1,"123",Kurz,Helmut result in ERROR

at+cpbw=1,"123","Kurz,Helmut"

But there are also some points to note when using quotation marks "":

There are some characters which cannot be entered in normal way:

e.g. quotation mark "" character itself, because this is interpreted as the end of the <text>

To make this (and some other special characters) possible to be entered,

the character with hex value 0x5c is used as escape character. In the ASCII character

set this is equal to the ` ` , like proposed in V.25ter. (Ascii=Windows=0x5C) Unfortunately there is no ` ` in GSM character set. The 0x5C equals to the ` `

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The escape sequence has the following structure:

-The sequence begins with the escape character 0x5C(ASCII=Windows='\', GSM='Ö')

-The special character follows and is entered 2 Byte representation of the GSM character set value .

e.g. the 2 Byte representation of the '@' (GSM=0x00) is '00'

Following special characters shall be entered by using the escape sequence:

GSM Char	Hex char.	ASCII	GSM Esc Seq	Seq.(hex)	Note
Ö	5C	\	Ö5C	5C 35 43	Backslash
"	22	"	Ö22	5C 32 32	String delimiter
ò	08	BSP	Ö08	5C 30 38	Backspace
@	00	NULL	Ö00	5C 30 30	GSM NULL

Examples:

GSM string wanted in Phonebook	String in AT+CPBW Command(GSM)	String in AT+CPBW Command (Hex)
Ölhändler	"Ö5C1hÖ7Bndler"	22 5C 35 43 6C 68 7B 6E 64 6C 65 72 22
"Eddi" Kurz	"Ö22EddiÖ22 Kurz"	22 5C 32 32 45 64 64 69 5C 32 32 20 4B 75 72 7A 22
Oòo	"OÖ08o"	22 4F 5C 30 38 6F 22
@Adr.	"Ö00Adr."	22 5C 30 30 41 64 72 2E 22
		[no problems with strlen()]
	"@Adr."	22 00 41 64 72 2E 22
		[may cause problems with strlen() in application]

Note:

When reading phonebook records, there is NO replacement. Every character will appear in normal GSM character set notation (like the left column in the example above).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix B

Example for creating/interrogation of an organizer entry

-vcs object which has to be uploaded:

```
BEGIN:VCALENDAR
VERSION:1.0
BEGIN:VEVENT
CATEGORIES:ANNIVERSARY
DTSTART:19991213T100000
DESCRIPTION:W. von Siemens
END:VEVENT
END:VCALENDAR
```

-hexadecimal representation of this object:

```
424547494E3A5643414C454E4441520D0A56455253494F4E3A312E300D0A4245
47494E3A564556454E540D0A43415445474F524945533A414E4E4956455253415
2590D0A445453544152543A3139393931323133543130303030300D0A44455343
52495054494F4E3A572E20766F6E205369656D656E730D0A454E443A56455645
4E540D0A454E443A5643414C454E4441520D0A
```

-upload of an entry on record 20

```
at^sbnw="vcs",20,1,3<CR>
<CR><LF> > <Space>
424547494E3A5643414C454E4441520D0A56455253494F4E3A312E300D0A4245
47494E3A564556454E540D0A43415445474F<Ctrl-Z>
<CR><LF>OK<CR><LF>
```

```
at^sbnw="vcs",20,2,3<CR>
```

```
<CR><LF> > <Space>
```

```
524945533A414E4E49564552534152590D0A445453544152543A3139393931323
133543130303030300D0A44455343524950<Ctrl-Z>
```

```
<CR><LF>OK<CR><LF>
```

```
at^sbnw="vcs",20,3,3<CR>
```

```
<CR><LF> > <Space>
```

```
54494F4E3A572E20766F6E205369656D656E730D0A454E443A564556454E540D
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0A454E443A5643414C454E4441520D0A<Ctrl-Z>

<CR><LF>OK<CR><LF>

All characters are answered with an echo.

Echoing can be switched off with „ATE0“. In this example the organizer entry is uploaded in 50 bytes packets (100 input characters in every pdu). The blue painted characters characterize the responses of the mobile.

-interrogation of the current <type>,<subtype>,<actNumber>,<maxNumber>

at^sbnw?<CR> <CR><LF>^SBNW: "vcs",20,2,3<CR><LF> <CR><LF>OK<CR><LF>

description: The actual object which is uploaded is an VCS object. It has to be stored on record 20. 2 of 3 packets are already uploaded.

-deleting of record 20

at^sbnw="vcs",20,0<CR> <CR><LF>OK<CR><LF>

-download entry from record 20

at^sbnr="vcs",20<CR> <CR><LF>^SBNR:<space>"vcs",20,1,1<CR><LF>
424547494E3A5643414C454E4441520D0A56455253494F4E3A312E300D0A4245
47494E3A564556454E540D0A43415445474F524945533A414E4E4956455253415
2590D0A445453544152543A313939393132313354313030303030D0A44455343
52495054494F4E3A572E20766F6E205369656D656E730D0A454E443A56455645
4E540D0A454E443A5643414C454E4441520D0A<CR><LF> <CR><LF>OK<CR><LF>

The mobile segments the record entry in 176 byte (=176*2 characters) packets.

-Download of an empty record 20

at^sbnr="vcs",20<CR> <CR><LF>OK<CR><LF>

-Test command of AT^SBNW

at^sbnw=?<CR> <CR><LF>^SBNW:<space>("bmp",(0)),(,"mid",(0)),(,"vcs",(1-30)) <CR><LF>
<CR><LF>OK<CR><LF>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] อภิชาติ ภู่วลัย. 2546. เริ่มต้นเขียนโปรแกรมติดต่อและควบคุมฮาร์ดแวร์ด้วย Visual Basic. กรุงเทพฯ Inforpress Developer Book.
- [2] ศิวะพงษ์ เพ็ชรสงค์. 2546. “ถอดรหัส SMS เรื่องลับที่ไม่ลับอีกต่อไป”. วารสาร PC Magazine Vol.9 No.9. P:P176-181



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้