สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบแจ้งเตือนผ่าน GPRS

WARNING SYSTEM VIA GPRS

นายไพฑูรย์ สว่างวงศ์
นายสุจินต์ กิจโสภากุล
นายสุรสิทธิ์ ณ นคร

เลขหมู่.....
เลขทะเบียน 62680
วัน,เดือน,ปี 21 ส.ค. 2549

b 11k29340
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2548

WARNING SYSTEM VIA GPRS

PAITHOON SAWANGWONG

SUJIN KITSOPAKUL

SURASIT NANAKORN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING

DEPARTMENT OF INSTRUMENTATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2005

ภาควิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

หัวข้อปริญญานิพนธ์	ระบบแจ้งเตือนผ่าน GPRS			
	WARNING SYSTEM VIA GPRS			
นักศึกษาผู้จัดทำ	นายไพฑูรย์	สว่างวงศ์	**รหัสประจำตัว**	46015451
	นายสุจินต์	กิจโสภากุล	**รหัสประจำตัว**	46015470
	นายสุรสิทธิ์	ณ นคร	**รหัสประจำตัว**	46015473
ปริญญา	วิศวกรรมศาสตรบัณฑิต			
สาขาวิชา	วิศกรรมการวัดคุม			
ปีการศึกษา	2548			

อาจารย์ผู้ควบคุมปริญญานิพนธ์	ลายมือชื่อ
ผศ.เชื้อ นกอยู่	

ภาควิชารับรองแล้ว

(รศ.ประสิทธิ์ จุลเสรีวงศ์)

หัวหน้าภาควิชาวิศวกรรมการวัดคุม

หัวข้อปริญญานิพนธ์	ระบบแจ้งเตือนผ่าน GPRS			
	WARNING SYSTEM VIA GPRS			
นักศึกษาผู้จัดทำ	นายไพฑูรย์	สว่างวงศ์	**รหัสประจำตัว**	46015451
	นายสุจินต์	กิจโสภากุล	**รหัสประจำตัว**	46015470
	นายสุรสิทธิ์	ณ นคร	**รหัสประจำตัว**	46015473
อาจารย์ที่ปรึกษา	ผศ.เชื้อ	นกอยู่		
ปีการศึกษา	2548			

บทคัดย่อ

ระบบควบคุมอัตโนมัติในอุตสาหกรรม แม้ว่าจะถูกออกแบบมาดีสักเพียงใด แต่ในความเป็นจริงอาจมีการรบกวนจากภายนอกทำให้เกิดความผิดพลาดต่อกระบวนการผลิต ดังนั้นถ้ากระบวนการมีระบบตรวจสอบและแจ้งเตือนความผิดพลาดที่เกิดขึ้น ก็จะทำให้เราสามารถตัดสินใจแก้ไขหรือปรับปรุงกระบวนการได้อย่างทันท่วงที ซึ่งการแจ้งเตือนนั้นได้แบ่งลำดับความสำคัญของการเกิดความผิดปกติคือ ตั้งแต่เริ่มผิดปกติซึ่งจะแสดงสัญญาณเตือนที่มอนิเตอร์ ไปจนถึงความผิดปกติระดับสุดท้ายซึ่งเป็นระดับที่กระบวนการไม่สามารถยอมรับได้ก็จะแสดงสัญญาณเตือนที่มอนิเตอร์ และทำการส่งข้อความมายังโทรศัพท์มือถือซึ่งอาจจะเป็นหัวหน้า หรือผู้บริหารระดับสูงเพื่อรับรู้และตัดสินใจต่อไป

Thesis Title	Warning System Via GPRS
Authors	Mr. Paithoon Sawangwong
	Mr. Sujin Kitsopakul
	Mr. Surasit Nanakorn
Thesis Advisor	Asst.Prof. Chuae Nok yoo
Year	2005

ABSTRACT

Any automatic controlled system in industry is always designed as good as we can do but in the fact, the system is sometimes disturbed from external that cause a mistake to production process. The mistake will occur less, if there are checking process and warning one about the mistake so we can decide, rectify or improve the process promptly. The warning is divided according to an order of importance of mistake which shows the mistake from beginning of mistake to final mistake. The beginning of mistake will be showed as warning on monitor. As for the final mistake is the mistake that process cannot control by itself; therefore the mistake will be showed on monitor and it will be send as message to chief or executive's cell phone for being informed and decision later.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีเพราะได้รับความเมตตาและสนับสนุนจาก ผู้ช่วยศาสตราจารย์เชื้อ นกอยู่ ที่ได้ให้คำแนะนำแก่ผู้วิจัยตลอดเวลา อีกทั้งยังเอื้อเฟื้ออุปกรณ์และเครื่องมือต่างๆ ในการทำปริญญานิพนธ์ในครั้งนี้ ผู้จัดทำรู้สึกซาบซึ้ง และขอขอบพระคุณเป็นอย่างสูง

ขอขอบคุณ อาจารย์ภาควิชาวิศวกรรมการวัดคุมทุกท่าน ที่ได้ให้คำแนะนำและสถานที่ภาควิชาอันเป็นประโยชน์ต่อการทำปริญญานิพนธ์ฉบับนี้

และที่ลืมเสียมิได้ ขอขอบพระคุณคุณพ่อ คุณแม่ และครอบครัวอันเป็นที่รัก ที่สนับสนุนและเป็นกำลังใจตลอดการทำการปริญญานิพนธ์ฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ คณะผู้จัดทำมอบแด่ผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VII

สารบัญภาพ.....VIII

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและเหตุจูงใจของการทำวิจัย.....1

1.2 วัตถุประสงค์ของปริญญานิพนธ์.....1

1.3 ขอบเขตของปริญญานิพนธ์.....1

1.4 ขั้นตอนการศึกษา.....2

บทที่ 2 ทฤษฎีและหลักการ.....3

2.1 ระบบโทรศัพท์เคลื่อนที่ GSM.....3

2.1.1 บทนำ.....3

2.1.2 โครงสร้างและองค์ประกอบของระบบ GSM.....4

2.2 ระบบระบบเคลื่อนที่ CDMA.....9

2.2.1 บทนำ.....9

2.2.2 ประเภทของรหัสที่ใช้ในระบบ CDMA (IS-95).....10

2.2.3 การส่งสัญญาณจากสถานีฐานไปหาโทรศัพท์ (Forward Link).....12

2.2.4 การส่งสัญญาณจากโทรศัพท์ไปหาสถานีฐาน (Reverse Link).....21

2.2.5 คุณลักษณะและข้อดีของระบบ CDMA.....26

2.3 การติดต่อผ่านแบบอนุกรม.....27

2.3.1 พื้นฐานการสื่อสารแบบอนุกรม.....27

2.3.2 มาตรฐาน RS-232C.....28

2.3.3 รายละเอียดของสายสัญญาณ.....29

2.4 เครื่องควบคุมแบบโปรแกรมได้.....30

สารบัญ (ต่อ)

หน้า

2.4.1 ประวัติความเป็นมาของ PLC....30
2.4.2 โครงสร้างฮาร์ดแวร์ของ PLC....31
2.4.3 ส่วนประกอบด้านหน้าของ PLC....32
2.4.4 การต่อกับแหล่งจ่ายไฟฟ้าภายนอก....34
2.4.5 การเชื่อมต่อกับเครื่องคอมพิวเตอร์ส่วนบุคคล....34
2.4.6 การติดต่อกับพอร์ตภายนอก....34
2.5 โปรแกรม Direct SOFT32....36
2.6 Analog Input Module....39
2.6.1 การติดตั้งการ์ด Analog กับ PLC KOYO รุ่น DL06....40
2.6.2 การสร้าง Scale ค่าใช้งาน....41
2.7 โปรแกรม DS Data server....42
2.7.1 การตั้งค่าการเชื่อมต่อกับ PLC....42
2.7.2 การสร้างหัวข้อในการการติดต่อ PLC....45
2.8 การติดต่อสื่อสารระหว่าง PLC กับ GPRS Module....48

บทที่ 3 หลักการและทฤษฎีที่ใช้ในการออกแบบ....50
3.1 บทนำ....50
3.1.1 ตัวควบคุม (Controller)....52
3.1.2 อุปกรณ์ควบคุมสุดท้าย (Final Control Element)....52
3.1.3 กระบวนการ (Plant or Process)....52
3.1.4 อุปกรณ์วัด (Measuring Instruments)....53
3.2 การตรวจวัดอุณหภูมิ....53
3.3 การตั้ง SPAN และ ZERO ของ TRANSMITTER....55
3.3.1 ย่านการวัด (RANGE)....56
3.3.2 สแปน (Span)....56
3.3.3 หลักการคำนวณ....56

สารบัญ (ต่อ)

หน้า

บทที่ 4 รายละเอียดในการสร้าง และประกอบปริญญานิพนธ์.....58

4.1 ส่วนของฮาร์ดแวร์ (Hardware).....59

4.1.1 Transmitter and RTD.....59

4.1.2 Solid State Relay.....60

4.1.3 Power Supply.....61

4.1.4 Heater.....61

4.1.3 PLC.....62

4.2 ส่วนของซอร์ฟแวร์ (Software).....63

4.2.1 โปรแกรมของ PLC.....63

4.2.2 โปรแกรมหน้าจอ SPC (Microsoft Visual Basic).....63

บทที่ 5 การทดลองและผลการทดลอง.....65

5.1 จุดประสงค์ในการทดลอง.....66

5.2 การออกแบบการทดลอง.....65

5.3 ขั้นตอนการทดลอง.....65

5.4 บันทึกผลการทดลอง.....66

5.5 สรุปผลการทดลอง.....72

บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ.....73

6.1 สรุปผลงาน.....73

6.2 ข้อเสนอแนะและแนวทางพัฒนา.....73

บรรณานุกรม.....74

ภาคผนวก.....75

สารบัญตาราง

ตารางที่ **หน้า**

2.1 กำลังส่งของเครื่องโทรศัพท์แต่ละประเภทตามมาตรฐานของระบบ GSM.....5
2.2 ระดับกำลังส่งของสถานีฐาน BTS ตามมาตรฐาน GSM 900 และ GSM 1800.....7
2.3 โครงสร้างของแพ็กเกต.....23
2.4 แสดงผลของไฟขณะ PLC ทำงาน.....33
2.5 แสดงการทำงานที่ Mode ต่างๆของสวิตช์.....33
2.6 แสดงค่า V-Memory ที่เกี่ยวข้องกับ Analog Input Module.....41
2.7 แสดงรายละเอียดของขา J1.....45
2.8 แสดงรายละเอียดของขา J3.....46
2.9 แสดงรายละเอียดของขา J5.....46
2.10 แสดงรายละเอียดของขา JP1.....46
2.11 แสดงรายละเอียดของขา J2.....46
3.1 แสดงค่าการทำ Scaling.....57
4.1 แสดง I/O Assignment ของ PLC.....63

สารบัญภาพ

ภาพที่ หน้า

2.1 โครงสร้างและองค์ประกอบของระบบ GSM....4

2.2 ประเภทของแผ่น SIM ทั้งสองแบบ....6

2.3 ภาพแสดงลักษณะและหน้าที่ของส่วน BTS และ BSC....6

2.4 มาตรฐานการอินเทอร์เฟสที่ใช้ระหว่างจุดต่อต่างๆ ภายในระบบ GSM....8

2.5 แสดงวงจร 15 stage LFSR สำหรับ....12

2.6 แสดงวงจร 15 stage LFSR สำหรับ Q....12

2.7 การแบ่งช่องสัญญาณที่ส่งออกโดยสถานีฐาน....13

2.8 วงจรสร้างสัญญาณไพลอต....14

2.9 วงจรสร้างสัญญาณซิงโครไนเซชัน....15

2.10 โครงสร้างเฟรมและรูปแบบของข้อมูลที่ใช้ในช่องสัญญาณซิงก์....16

2.11 โครงสร้างของโพลิโนเมียลที่ใช้ในการเข้ารหัส CRC ของสัญญาณซิงก์....17

2.12 โครงสร้างของวงจรเข้ารหัส CRC ของสัญญาณซิงก์....17

2.13 ขั้นตอนการสร้างสัญญาณเพจจิง....18

2.14 ขั้นตอนการสร้างช่องสัญญาณทราฟฟิก....20

2.15 รายละเอียดรูปแบบของ Forward traffic channel public long code mask....21

2.16 รายละเอียดรูปแบบของ ESN....21

2.17 การจัดช่องสัญญาณที่ส่งออกโดยเครื่องโทรศัพท์เคลื่อนที่....22

2.18 ขั้นตอนการสร้างการสร้างสัญญาณสำหรับส่งผ่านช่องสัญญาณแอกเซส
ในทิศทางที่ส่งจากโทรศัพท์เคลื่อนที่ไปยังสถานีฐาน....22

2.19 ขั้นตอนการสร้างสัญญาณทราฟฟิกในทิศทางจากโทรศัพท์เคลื่อนที่ไปยังสถานีฐาน....24

2.20 ตัวอย่างการเปิดปิดเกตสำหรับการส่งที่อัตราบิตค่าต่างๆ....25

2.21 รายละเอียดรูปแบบของ Public long code mask....25

2.22 แสดงลักษณะสัญญาณของการสื่อสารแบบซิงโครนัส....27

2.23 การเชื่อมต่อคอนเน็กเตอร์....29

2.24 แสดงการประมวลผลของ PLC....31

2.25 แสดงส่วนต่างๆด้านหน้าของ PLC KOYO DL06....32

2.26 แสดงการต่อกับแหล่งจ่ายไฟฟ้าภายนอก....34

2.27 การต่อสายเคเบิลระหว่าง คอมพิวเตอร์ส่วนบุคคลกับ PLC....34

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

2.28 แสดงการต่อสายเข้ากับพอร์ตเพื่อใช้ในการติดต่อสื่อสาร....35

2.29 แสดงการสร้าง Project ใหม่ขึ้นมาเพื่อทำการเขียนโปรแกรม....36

2.30 แสดงการเลือกชนิดของ PLC และรุ่นของ PLC หรือชนิดของ CPU แล้วคลิก OK....36

2.31 แสดงการต่อ PLC กับ computer และทำการ Link กับโปรแกรม DirectSOLFT32 37

2.32 แสดงหน้าจอที่ไว้สำหรับเขียนโปรแกรมให้กับ PLC....37

2.33 แสดงการเรียกใช้ข้อมูลจากฐานข้อมูลต่างๆ....37

2.34 แสดงบล็อกสัญลักษณ์การเขียน Ladder....38

2.35 แสดงไอคอนการติดต่อกับ PLC....38

2.36 แสดงการเลือก Mode การทำงานของ PLC....38

2.37 แสดงการทำงานของโปรแกรมที่อยู่ใน Mode RUN....39

2.38 แสดง Analog Input Module แบบ 4channel....39

2.39 แสดงโครงสร้างภายในของ Analog Input Module แบบ 4channel....40

2.40 แสดงตัวเครื่อง PLC KOYO รุ่น DL06 เมื่อทำการติดตั้งการ์ด Analog....40

2.41 แสดงสูตรการคำนวณสร้าง Scale ค่าใช้งาน....41

2.42 แสดงการเชื่อมโยงข้อมูลโดยการใช้โปรแกรม DS Data server....42

2.43 แสดงการเลือกการเชื่อมต่อกับ PLC....42

2.44 แสดงการเลือกพอร์ตของเครื่องคอมพิวเตอร์....43

2.45 แสดงการเลือกรุ่นต่างๆ ของ PLC....43

2.46 แสดงการเลือกชนิดของ Protocol....44

2.47 แสดงการการอ่านข้อมูลของ PLC....44

2.48 แสดงการตั้งชื่อหัวข้อโปรแกรมของเครื่อง PLC....45

2.49 แสดงด้านบนของตัวบอร์ด RS-EB-S1 และลายวงจร
ที่ใช้เชื่อมต่อเข้ากับ GPRS MODULE....47

2.50 แสดงด้านล่างของตัวบอร์ด RS-EB-S1 และลายวงจร
ที่ใช้เชื่อมต่อเข้ากับ GPRS MODULE....47

2.51 แสดงตัว GPRS MODULE....48

2.52 แสดงพอร์ตที่ใช้ในการเชื่อมต่อระหว่าง PLC กับ GPRS Module....49

3.1 แสดงการควบคุมอุณหภูมิในถังแบบ Manual Control....50

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

3.2 แสดง Block Diagram ของการควบคุมอุณหภูมิในถังแบบ Manual Control.....50
3.3 แสดงการควบคุมอุณหภูมิในถังด้วยการควบคุมแบบอัตโนมัติ.....51
3.4 แสดง Block Diagram ของระบบควบคุมแบบป้อนกลับโดยทั่วไป.....52
3.5 แสดงการเปลี่ยนแปลงค่าของอาร์ทีดี ที่ทำจากโลหะชนิดต่างๆ.....55
4.1 แสดงบล็อกไดอะแกรม.....58
4.2 แสดงภาพถ่ายของ Hardware.....59
4.3 แสดงอุปกรณ์ และตำแหน่งขาของ RTD และ Transmitter.....59
4.4 แสดงตัวอุปกรณ์ วงจรภายใน และตำแหน่งขาของ Solid State Relay.....60
4.5 แสดงอุปกรณ์ภาคจ่ายไฟที่ใช้.....60
4.6 แสดงสัญลักษณ์ของ Heater.....61
4.7 แสดงลักษณะของ PLC.....62
4.8 แสดงลักษณะการต่ออุปกรณ์อินพุตและเอาต์พุต.....62
4.9 แสดงหน้าจอ SPC (Microsoft Visual Basic).....64
5.1 แสดงกระบวนการการทำงานของระบบที่ใช้ทดลอง.....66
5.2 แสดงค่าตัวแปรต่างๆที่รับค่ามาจากกระบวนการที่ทำการควบคุมอยู่.....67
5.3 แสดงค่าสภาวะทำงานขณะ Alarm1.....68
5.4 แสดงค่าสภาวะทำงานขณะ Alarm2.....69
5.5 แสดงผลของการแจ้งเตือนของระบบในระดับสุดท้าย(Alarm3).....70
5.6 แสดงผลของข้อความเตือนที่ส่งมายังโทรศัพท์มือถือ.....71
5.7 แสดงค่าสภาวะทำงานขณะ RTD เกิดความเสียหาย.....71
5.8 แสดงผลของการแจ้งเตือนของระบบในส่วนของ Sensor Alarm.....72

บทที่ 1
บทนำ

1.1 ความเป็นมาและเหตุจูงใจของการทำวิจัย

ปัจจุบันการนำเอา PLC (Programmable Logic Controller) มาใช้ในการควบคุมกระบวนการต่างๆ ในงานอุตสาหกรรมเป็นที่นิยมอย่างมาก เนื่องจากราคาที่ไม่แพงมากนักอีกทั้งยังมีความสามารถหลากหลายในฟังก์ชันการทำงาน ทำให้สามารถนำมาประยุกต์ใช้งานร่วมกับอุปกรณ์อื่นได้ไม่ยากนัก ดังนั้นในโครงงานนี้เราจึงนำเอา PLC มาประยุกต์ใช้โดยการนำ Plant มาต่อ เพื่อจำลองสถานการณ์ของกระบวนการโดยที่ค่าสถานะของกระบวนการจะถูกนำไปแสดงที่หน้าจอคอมพิวเตอร์ นอกจากนั้นเมื่อกระบวนการเกิดความผิดพลาดหรือล้มเหลวขึ้นระบบจะทำการส่งข้อความเตือนมายังโทรศัพท์เคลื่อนที่ในรูปของ SMS

1.2 วัตถุประสงค์ของปริญญานิพนธ์

- ศึกษาและประยุกต์ใช้งาน PLC
- ศึกษาและประยุกต์ใช้งานโปรแกรม Visual Basic
- ศึกษาการเชื่อมต่อระหว่างโปรแกรม Visual Basic กับ PLC
- ศึกษาการเชื่อมต่อระหว่าง PLC กับ GPRS Module
- ศึกษาคำสั่งที่ใช้ในการเขียน GPRS Module เพื่อใช้ในการส่ง SMS
- ทำการนำ Plant จำลองมาต่อเพื่อให้เห็นภาพชัดเจนของการแจ้งเตือนของระบบ

1.3 ขอบเขตของปริญญานิพนธ์

- แสดงหน้าจอการทำงานของ Plant บนเครื่องคอมพิวเตอร์และสามารถควบคุม Plant ผ่านเครื่อง PLC

- สามารถเชื่อมต่อระหว่างโปรแกรมVisual Basic กับ PLC ได้

- สามารถส่งข้อความเตือนผ่านมายังโทรศัพท์เคลื่อนที่ในรูปของ SMS โดยใช้ PLC เป็นตัวแจ้งเตือน

- จอคอมพิวเตอร์สามารถแสดงผลการทำงานของกระบวนการได้

1.4 ขั้นตอนการศึกษา

- ศึกษาทฤษฎีเกี่ยวกับการสื่อสารข้อมูลในรูปแบบต่างๆ
- ศึกษาทฤษฎีการเขียนโปรแกรม Visual Basic 6.0
- ศึกษาและปฏิบัติการ การเขียนโปรแกรมเครื่อง PLC
- ศึกษาการการควบคุมอุณหภูมิจาก Plant จำลองโดยใช้ PLC
- ทำการเขียนโปรแกรม ทดสอบ และ แก้ปัญหา
- สรุปผลการทดลอง

บทที่ 2
ทฤษฎีและหลักการ

2.1 ระบบโทรศัพท์เคลื่อนที่ GSM

2.1.1 บทนำ

คำว่า GSM เดิมเป็นชื่อย่อมาจาก Groupe Speciale Mobile ซึ่งเป็นกลุ่มวิจัยที่เกิดขึ้นจากการรวมตัวกันของประเทศในยุโรปหลายประเทศเพื่อศึกษาและพัฒนาระบบโทรศัพท์เคลื่อนที่ ในราวปี ค.ศ. 1982 ภายใต้การดูแล และสนับสนุนจากหน่วยงาน CEPT (the European Conference of Postal and Telecommunications Administrations) โดยมีจุดประสงค์หลักเพื่อพัฒนาระบบโทรศัพท์เคลื่อนที่ระบบใหม่ที่สามารถรองรับความต้องการใช้งานของผู้ใช้บริการ ได้มากขึ้น และที่สำคัญคือต้องสามารถให้ผู้ใช้โทรศัพท์มีอิสระในการติดต่อสื่อสารจาก ณ สถานที่ใดๆ ก็ได้ภายในกลุ่มประเทศในยุโรป (roaming) ระบบโทรศัพท์เคลื่อนที่ GSM นั้นจัดว่าเป็นระบบโทรศัพท์เคลื่อนที่ในยุคแรก จุดแตกต่างที่สำคัญของระบบโทรศัพท์เคลื่อนที่ในยุคที่สองและยุคแรกคือ การนำเทคโนโลยีดิจิตอลมาแทนการใช้เทคโนโลยีอนาล็อก เหตุผลที่กลุ่มวิจัยของ GSM ได้เลือกใช้เทคโนโลยีดิจิตอลสำหรับการพัฒนาระบบ GSM ก็เพราะระบบดิจิตอลมีข้อดีกว่าระบบโทรศัพท์เคลื่อนที่แบบอนาล็อกหลายประการดังนี้

- ระบบดิจิตอลสามารถใช้ประโยชน์จากสเปกตรัมที่มีอยู่จำกัดได้ดีกว่าระบบอนาล็อก เพราะสามารถรับส่งสัญญาณได้จำนวนช่องมากกว่า และให้คุณภาพของสัญญาณที่ดีกว่า
- สัญญาณเคลื่อนที่จากระบบโทรศัพท์เคลื่อนที่แบบดิจิตอลยังสามารถนำมาติดต่อกับโทรศัพท์ที่มีใช้งานตามบ้านตามปกติทั่วไปได้โดยตรง เพราะในช่วงเวลานั้นชุมสายของระบบโทรศัพท์ PSTN เหล่านี้ได้มีการเปลี่ยนแปลงอย่างมากคือ จากเดิมที่ใช้สัญญาณอนาล็อกมาเป็นระบบดิจิตอล
- นอกจากนี้ในเวลานั้นได้มีการคาดคะเนกันว่าระบบ Integrated Services Digital Network หรือ ISDN นั้นมีการใช้งานกันแพร่หลายภายในยุโรปในอนาคตอันใกล้ ดังนั้นถ้าใช้ระบบดิจิตอลก็จะสะดวกต่อการสร้างวงจรเชื่อมต่อระหว่างกัน
- ในระบบดิจิตอลสามารถสร้างระบบป้องกันการดักฟัง และการแอบใช้การบริการจากผู้ที่ไม่ใช่สมาชิก ได้อย่างมีประสิทธิภาพกว่าระบบอนาล็อก

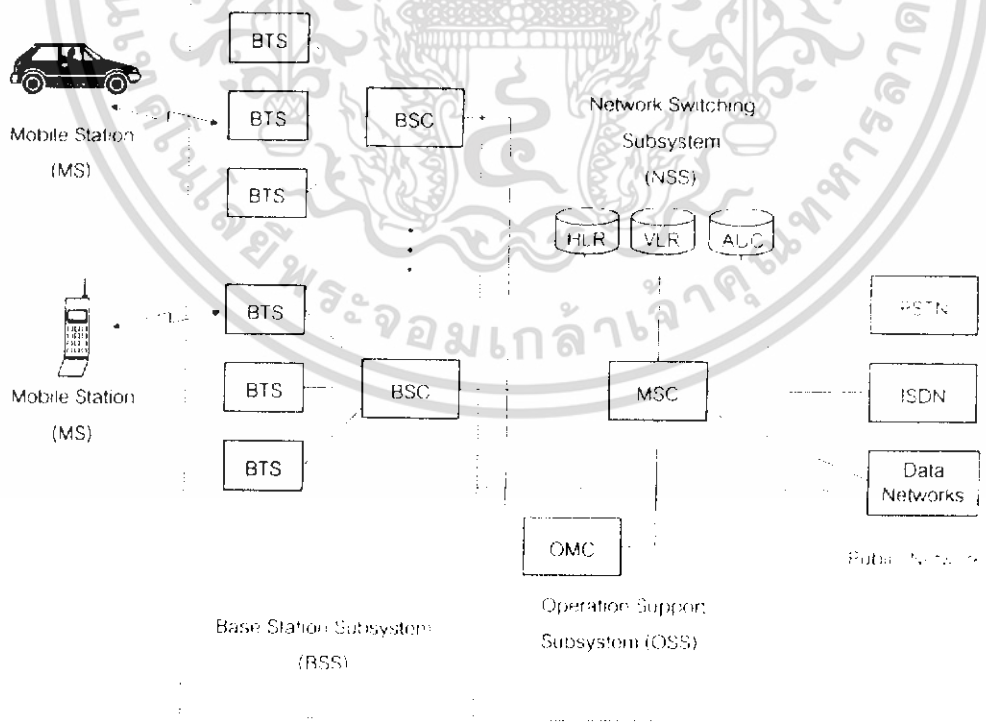
ด้วยเหตุนี้ระบบ GSM จึงออกแบบเป็นระบบดิจิตอล ในปัจจุบัน GSM เป็นที่รู้จักในชื่อเต็มว่า Global System for Mobile Communications ระบบโทรศัพท์เคลื่อนที่แบบ GSM มีการเริ่มนำมาใช้งานจริงครั้งแรกในราวปี ค.ศ. 1991 และในปัจจุบันระบบ GSM นี้มีการใช้งานกันอย่างแพร่หลายกว่า 120 ประเทศ รวมถึงประเทศไทยด้วย จากปริมาณความต้องการใช้งานที่มีมากทำให้มีการนำเอามาตรฐาน GSM ที่ใช้ในย่าน 900 MHz ไปใช้กับความถี่ย่าน 1800 MHz และมีการเรียกระบบดังกล่าวว่า Digital Cellular System 1800 (DCS 1800) หรือบางทีก็มีชื่อเรียกว่า Personal Communications Networks (PCN)

2.1.2 โครงสร้างและองค์ประกอบของระบบ GSM

โครงสร้างของระบบ GSM ประกอบด้วยองค์ประกอบต่างๆดังนี้

1. เครื่องโทรศัพท์เคลื่อนที่ (Mobile Station หรือ MS)
2. ส่วนสถานีฐาน (Bass Station Subsystem หรือ BSS)
3. ส่วนของระบบเนตเวิร์กและสวิตชิ่ง (Network and Switching Subsystem หรือ NSS)
4. ระบบปฏิบัติการ (Operation Support Subsystem หรือ OSS)

แต่ละส่วนมีลักษณะการต่อเชื่อมกันดังภาพที่ 2.1



ภาพที่ 2.1 โครงสร้างและองค์ประกอบของระบบ GSM

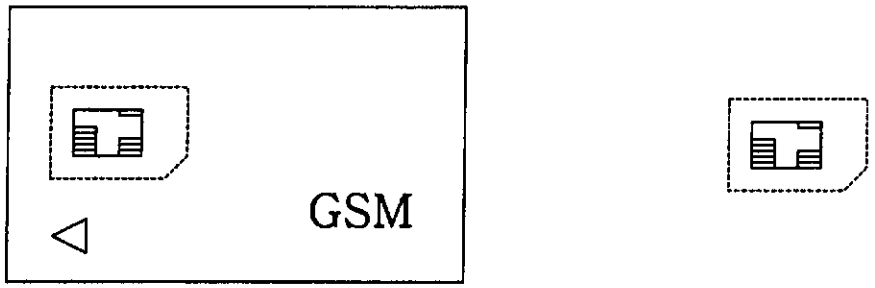
2.1.2.1 เครื่องโทรศัพท์เคลื่อนที่ (Mobile Station หรือ MS)

The Mobile Station (MS) คือ เครื่องโทรศัพท์เคลื่อนที่ที่ผู้ใช้บริการใช้ในการโทรออกหรือรับสายเรียกเข้านั้นเอง ในปัจจุบันเครื่องโทรศัพท์ที่มีใช้งานอยู่หลายรูปแบบ ตั้งแต่การติดตั้งอยู่กับยานพาหนะ (vehicle – mounted) แบบกระเป๋าหิ้ว (portable equipment) ไปจนถึงแบบมือถือ (handheld station) โดยทั่วไปแล้วเครื่องแต่ละประเภทจะแตกต่างกันก็ตรงลักษณะการใช้งาน และกำลังส่งของเครื่อง สำหรับระบบ GSM ได้แบ่งชนิดโทรศัพท์ออกเป็น 5 กลุ่ม ตามขนาดกำลังส่งสูงสุดที่ใช้ ดูรายละเอียดได้ในตารางที่ 2.1 ส่วนระบบ DCS 1800 มีวิธีการแบ่งประเภทของเครื่องโทรศัพท์ที่ต่างออกไปคือแบ่งเพียง 2 กลุ่ม ทั้งนี้เพราะย่านความถี่ดังกล่าว เซลล์มีขนาดเล็กกว่า ทำให้ไม่ต้องใช้กำลังส่งที่สูงนัก ดูรายละเอียดได้ในตารางที่ 2.1

ตารางที่ 2.1 กำลังส่งของเครื่องโทรศัพท์แต่ละประเภทตามมาตรฐานของระบบ GSM

Power class	Maximum power of a GSM 900 mobile station	Maximum power of a GSM 1800 mobile station
1	20 W (43 dBm)	1 W (30 dBm)
2	8 W (39 dBm)	0.25 W (24 dBm)
3	5 W (37 dBm)	
4	2 W (33 dBm)	
5	0.8 W (29 dBm)	

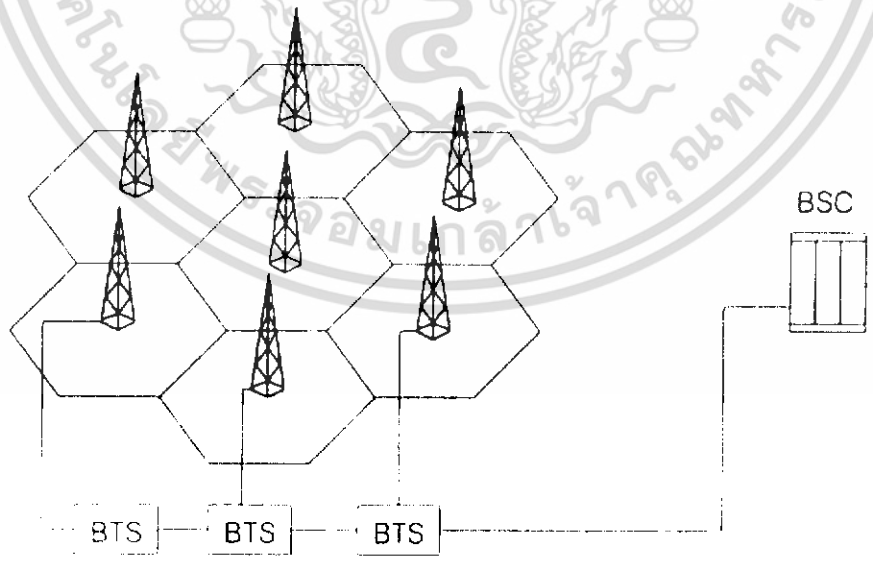
ภายในอุปกรณ์โทรศัพท์เครื่องหนึ่งประกอบด้วยส่วนย่อย 2 ส่วน คือ Mobile Equipment (ME) และ Subscriber Identity Module (SIM) ส่วนของ ME ทำหน้าที่จัดการกับการรับส่งคลื่นสัญญาณวิทยุระหว่างโทรศัพท์เคลื่อนที่กับส่วนสถานีฐาน และรวมไปถึงอุปกรณ์สำหรับใช้ในการติดต่อกับผู้ใช้ เช่น ไมโครโฟน ลำโพง จอภาพ และปุ่มต่างๆ ขณะที่ในส่วนของ SIM คือ สมาร์ตการ์ด (smart card) แผ่นบางๆ ที่ทำหน้าที่เก็บข้อมูลที่เกี่ยวกับผู้ใช้ รายการประเภทของบริการที่ผู้ใช้ได้ขอไว้ และข้อมูลที่เกี่ยวข้องกับกับโครงการ เช่น เลขประจำตัวของผู้ใช้ ตำแหน่งหรือบริเวณที่เครื่องโทรศัพท์มีการใช้งานอยู่ และรหัสลับที่ใช้ในการเข้ารหัสสัญญาณเพื่อป้องกันการแอบฟัง เป็นต้น นอกจากนี้ยังอาจจะเก็บหมายเลขโทรศัพท์ที่ผู้ใช้มีการติดต่อด้วยเป็นประจำเพื่อความสะดวกของผู้ใช้บริการ การจัดให้เครื่องโทรศัพท์ที่มีโครงสร้างในลักษณะนี้ช่วยให้เราสามารถเสียบแผ่น SIM เข้าไปในอุปกรณ์ ME ใดก็ได้ แผ่น SIM ที่ใช้ในปัจจุบันมี 2 รูปแบบคือ แผ่นที่มีขนาดเท่ากับบัตรเครดิตเรียกว่า ISO SIM และขนาดเล็กที่เรียกว่า plug – in SIM ดังแสดงในภาพที่ 2.2



ภาพที่ 2.2 ประเภทของแผ่น SIM ทั้งสองแบบ

2.1.2.2 ส่วนของสถานีฐาน (Base Station Subsystem)

Base Station Subsystem (BSS) ประกอบด้วย 2 ส่วนหลัก คือ Base Transceiver Station (BTS) และ Base Station Controller (BSC) ส่วนของ BTS ทำหน้าที่ติดต่อกับเครื่องโทรศัพท์เคลื่อนที่ (ME) ทั้งหลาย โดยที่ BTS หนึ่งตัวจะดูแลควบคุมบริเวณพื้นที่จำกัดบริเวณหนึ่งที่เรียกว่าเซลล์ โดยหลักๆ แล้ว BTS ประกอบด้วยอุปกรณ์ที่ทำหน้ารับส่งคลื่นวิทยุคล้ายๆ กับส่วน ME ของเครื่องโทรศัพท์เคลื่อนที่ โดยขนาดของสถานีส่งของสถานีฐานมีได้หลายระดับดูรายละเอียดได้ในตารางที่ 2.2 กลุ่มของ BTS ที่ครอบคลุมพื้นที่หลายๆ เซลล์จำนวนหนึ่งจะอยู่ภายใต้การดูแลการดูแลของ BSC หนึ่งตัว ซึ่งโดยปกติแล้ว BSC หนึ่งตัวจะสามารถดูแลและควบคุม BTS ได้จำนวนมากถึงหลายสิบหรือหลายร้อยชุด ดูภาพที่ 2.3



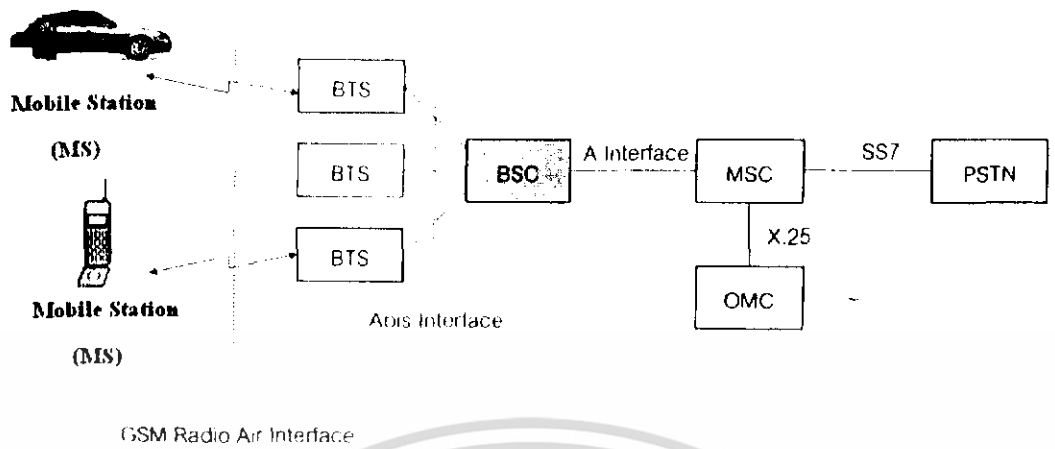
ภาพที่ 2.3 ภาพแสดงลักษณะและหน้าที่ของส่วน BTS และ BSC

ตารางที่ 2.2 ระดับกำลังส่งของสถานีฐาน BTS ตามมาตรฐาน GSM 900 และ GSM 1800

Power class	Maximum power of a GSM 900 mobile station	Maximum power of a GSM 1800 mobile station
1	320 W	20 W
2	160 W	10 W
3	80 W	5 W
4	40 W	2.5 W
5	20 W	-
6	10 W	-
7	5 W	-
8	2.5 W	-
Micro – BTS 1	0.25 W	1.6 W
Micro – BTS 2	0.08 W	0.5 W
Micro – BTS 3	0.03 W	0.16 W

ส่วนของ BSC มีหน้าที่หลักในการควบคุมการทำงานของ BTS ทุกตัวที่อยู่ภายใต้การดูแล เช่น การจักสรรช่องสัญญาณที่เหมาะสมสำหรับการติดต่อสื่อสาร การเริ่มต้น การเชื่อต่อและสิ้นสุดของการใช้ช่องสัญญาณแต่ละช่อง การควบคุมกำลังส่งของเครื่องโทรศัพท์ และรวมไปถึงเรื่องของการตัดสินใจ และการทำแฮนด์โอเวอร์ระหว่างเซลล์ ในกรณีที่โทรศัพท์เคลื่อนที่มีการย้ายจากเซลล์หนึ่งไปยังเซลล์ข้างเคียง สิ่งต่างๆ เหล่านี้เป็นหน้าที่ของ BSC ที่จะต้องจัดการทั้งหมด นอกจากนี้ BSC ยังมีความสามารถในการทำสวิตชิงได้ระดับหนึ่งด้วย โดย BSC สามารถเชื่อมต่อคู่สายระหว่างโทรศัพท์เคลื่อนที่สองเครื่องที่ใช้งานอยู่ภายในบริเวณที่ BSC ดูแลรับผิดชอบอยู่ได้

อีกด้านหนึ่งของ BTS ต่ออยู่กับ NSS (Network and Switching Subsystem) ซึ่งมี MSC (Mobile services Switching Center) เป็นองค์ประกอบสำคัญ การเชื่อมต่อของอุปกรณ์ทั้งสองนี้ จะมาตรฐานที่เรียกว่า A interface ส่วนการติดต่อระหว่าง BSC กับส่วน BTS ก็มีมาตรฐานที่แน่นอนเช่นกัน โดยมาตรฐานที่ใช้ชื่อเรียกว่า Abis interface ดังภาพที่ 2.4



ภาพที่ 2.4 มาตรฐานการอินเทอร์เฟสที่ใช้ระหว่างจุดต่อต่างๆ ภายในระบบ GSM

2.1.2.3 ระบบเนตเวิร์กและสวิตชิ่ง (Network and Switching Subsystem)

Network and Switching Subsystem (NSS) ประกอบด้วย 2 ส่วนหลักคือ Mobile services Switching Centre (MSC) และฐานข้อมูลสำหรับการจัดการกับการใช้งานของผู้ใช้บริการ ในส่วนของ MSC นั้นด้านหนึ่งเชื่อมต่อกับ BSC ซึ่งเป็นส่วนดูแลการรับส่งสัญญาณระหว่างสมาชิกผู้ใช้บริการโทรศัพท์เคลื่อนที่กับส่วนของโครงข่าย ส่วนอีกด้านหนึ่งต่อเชื่อมต่อกับระบบอื่นๆ ดังนั้น MSC จึงเป็นส่วนที่ทำหน้าที่สวิตช์และเชื่อมต่อคู่สายทั้งระหว่างผู้ใช้โทรศัพท์เคลื่อนที่ 2 เครื่องเข้าด้วยกัน และระหว่างผู้ใช้โทรศัพท์เคลื่อนที่กับผู้ใช้โทรศัพท์ธรรมดา หรือผู้ใช้บริการโครงข่ายประเภทอื่นๆ ด้วย ในการเชื่อมต่อระหว่าง MSC กับโครงข่ายภายนอกอาศัยมาตรฐานการเชื่อมต่อที่เรียกว่า CCITT Signaling no. 7 (SS7) โดยปกติแล้ว MSC หนึ่งชุดสามารถใช้ควบคุมดูแล BSC ได้หลายชุด MSC มีหน้าอื่นที่ต้องทำเพิ่มด้วยได้แก่ การจัดการกับการเคลื่อนที่ของผู้ใช้บริการ ซึ่งหมายถึง การทำแฮนด์โอเวอร์ และการรายงานตำแหน่งของเครื่องโทรศัพท์เคลื่อนที่

2.1.2.4 ระบบปฏิบัติการ (Operation Subsystem)

ในส่วนนี้ประกอบด้วย Operation and Maintenance Centre (OMC) ซึ่งมีหน้าที่หลักในการดูแลจัดการเรื่องการปฏิบัติการของระบบโดยรวม การจัดการกับปัญหาของอุปกรณ์บางส่วนที่อาจเกิดความเสียหาย การปรับค่าต่างๆ ภายในระบบอย่างเหมาะสม การจัดการเรื่องสมาชิกผู้ใช้บริการของระบบซึ่งรวมไปถึงการคิดค่าบริการและออกบิลเก็บค่าบริการการทำงานของ OMC ส่วนใหญ่แล้วจำเป็นต้องมีการติดต่อกับฐานข้อมูล HLR

2.2 ระบบระบบเคลื่อนที่ CDMA

2.2.1 บทนำ

การวิจัยและพัฒนาระบบโทรศัพท์เคลื่อนที่แบบ CDMA เริ่มขึ้นจริงจังเมื่อประมาณต้นปี ค.ศ. 1989 และเพียงไม่กี่ปีต่อมาคือในปลายปี ค.ศ. 1992 ผลงานวิจัยและพัฒนาเหล่านี้ก็ได้รับการเสนอและยอมรับให้เป็นมาตรฐานที่เรียกว่า IS-95 ในปัจจุบันระบบโทรศัพท์เคลื่อนที่แบบ CDMA นี้ได้มีการนำมาใช้งานจริงจังกันในหลายประเทศทั่วโลก สำหรับในประเทศไทยนั้น การสื่อสารแห่งประเทศไทยได้นำระบบนี้เข้ามาให้บริการโทรศัพท์เคลื่อนที่ในราวเดือนเมษายน ปี ค.ศ. 1998 หลักการเบื้องต้นของระบบ CDMA ได้จัดแบ่งความถี่ที่มีอยู่ทั้งหมดออกเป็นช่วง โดยแต่ละช่วงมีความกว้างเท่ากับ 1.25 MHz. โดยวิธี FDMA สังเกตว่าคลื่นพาห์หนึ่งความถี่มีขนาดแบนด์วิดท์ที่กว้างกว่าของระบบอื่นๆ เช่น GSM (200 kHz) มาก ด้วยขนาดแบนด์วิดท์ที่กว้างขนาดนี้ CDMA อนุญาตให้ผู้ใช้โทรศัพท์จำนวนมากจำนวนหนึ่งส่งสัญญาณเสียงลงบนคลื่นความถี่เดียวกัน ซึ่งหมายความว่าผู้ใช้ทุกๆคนที่ใช้คลื่นพาห์เดียวกันจะรบกวนกันในเชิงความถี่ตลอดเวลาการใช้งาน หากแต่ว่าโทรศัพท์แต่ละเครื่องจะสามารถแยกแยะว่าสัญญาณส่วนใดที่เป็นของตัวเองได้โดยอาศัยชุดรหัสที่แตกต่างกันในการรับและส่งข้อมูล ด้วยเหตุนี้จึงเรียกระบบนี้ว่า Code Division Multiple Access (CDMA) ชุดรหัสที่ใช้ในการทำสเปรดสเปกตรัมเหล่านี้มีชื่อเรียกว่า PN (Pseudorandom Noise) ซึ่งชุดรหัสแต่ละชุดที่ใช้จะต้องมีคุณสมบัติที่เรียกว่า orthogonal กับชุดรหัสอื่นๆ ทั้งหมด

ขั้นตอนการสร้างสัญญาณสำหรับการส่งออกเริ่มด้วยการนำสัญญาณเสียงที่อยู่ในรูปของสัญญาณดิจิตอลซึ่งมีอัตราของบิตข้อมูลค่อนข้างต่ำ (เช่น 9600 bps) มาคูณกับสัญญาณ PN ซึ่งมีอัตราของบิตข้อมูลที่สูงกว่า (1.228 Mcps) ผลลัพธ์ที่ได้ก็คือ ข้อมูลที่อัตราการส่งสูงกว่าสัญญาณเสียงที่ต้องการจะส่งมาก ด้วยเหตุนี้จึงเรียกว่ามีการสเปรดสเปกตรัม (spread spectrum) เกิดขึ้น และแบนด์วิดท์ของช่องสัญญาณที่ใช้ในการส่งข้อมูลนี้ก็ต้องมีขนาดใหญ่ขึ้นด้วยคือ 1.25 MHz (หมายเหตุ: สัญญาณที่ผ่านกระบวนการสเปรดสเปกตรัมแล้วมักจะใช้ Mcps แทน Mbps) ส่วนของภาครับสัญญาณที่ปลายทางก็จะทำการดีสเปรด (dispread) สัญญาณที่รับได้เพื่อดึงสัญญาณเสียงกลับคืนมา ในการดีสเปรดสัญญาณนั้นจะอาศัยวงจรคอรีเลเตอร์ (correlator) ซึ่งทำหน้าที่คูณสัญญาณที่ได้รับกับรหัส PN ชุดเดิมซึ่งเหมือนกันกับการคูณที่ภาคส่ง และจากนั้นก็หาค่าเฉลี่ยของสัญญาณทุกๆหนึ่งคาบของอัตราบิตข้อมูลของสัญญาณเสียง (หมายเหตุ: สัญญาณข้อมูลที่ต้องส่งออกมิจำเป็นต้องเป็นสัญญาณเสียงเท่านั้น อาจเป็นสัญญาณข้อมูลแบบอื่นเช่นข้อมูลประเภท data ก็ได้)

การที่เครื่องโทรศัพท์เคลื่อนที่แต่ละเครื่องที่ส่งผ่านสัญญาณเสียง หรือข้อมูลดาตาในรูปของสัญญาณดิจิตอลโดยใช้ชุดรหัส PN ที่แตกต่างกันและมีคุณสมบัติที่เป็น orthogonal ซึ่งกันและกัน ทำให้การรับ และดึงสัญญาณของเครื่องโทรศัพท์เครื่องๆหนึ่งจะไม่ไปดีสเปรดสัญญาณ

ที่ถูกสเปรดออกด้วยรหัส PN ชุดอื่น ด้วยเหตุนี้เครื่องโทรศัพท์แต่ละเครื่องจึงสามารถแยกแยะสัญญาณที่เป็นของตนเองได้ ค่าพารามิเตอร์ค่าหนึ่งที่สำคัญคือ Processing gain ซึ่งกำหนดให้มีค่าเท่ากับอัตราส่วนระหว่างอัตราบิตข้อมูลของรหัส PN ต่ออัตราการส่งของข้อมูลจริง ในกรณีที่ระบบส่งสัญญาณข้อมูลด้วยอัตรา 9600 bps จะได้ค่า Processing gain เท่ากับ 1.228 Mbps/9600 bps = 128 หรือเท่ากับ 21 dB

2.2.2 ประเภทของรหัสที่ใช้ในระบบ CDMA (IS-95)

ในระบบ CDMA ตามมาตรฐาน IS-95 มีการใช้รหัสทั้งหมด 3 ประเภท คือ

1. Walsh code
2. Long PN code
3. Short PN code

รหัสแต่ละประเภทได้ถูกนำมาใช้งานในระบบ IS-95 ในลักษณะที่แตกต่างกัน คุณสมบัติและรายละเอียดของรหัสทั้ง 3 ประเภทจะได้กล่าวถึงต่อไป

2.2.2.1 รหัส Walsh (Walsh Code)

รหัส Walsh มีชื่อเรียกอีกอย่างว่า Hadamard code ประกอบด้วยรหัสที่มีคุณสมบัติ orthogonal ระหว่างกันทั้งหมด 64 ชุด โดยที่รหัสแต่ละชุดมีความยาวเท่ากับ 64 บิต รหัสประเภทนี้ออกแบบมาเพื่อใช้สำหรับ สเปรดช่องสัญญาณที่มีอัตราการส่ง 1.228 Mbps (หมายเหตุ: รหัส 2 ชุดจะมีคุณสมบัติ orthogonal ระหว่างกันก็ต่อเมื่อค่าสหสัมพันธ์ข้าม (cross correlation) ระหว่างรหัสคู่นี้มีค่าเป็นศูนย์) การคำนวณหาค่าสหสัมพันธ์ข้ามสามารถหาได้จาก xiyi = 0 โดย xi และ yi คือค่าของบิตที่ i ของรหัสชุดแรกและชุดที่สองในรูปของไบโพลาร์ นั่นคือว่า xi, yi = -1 ถ้าบิตที่ i นั้นมีค่าเป็นศูนย์ และ xi, yi = 1 ถ้าบิตที่ i นั้นมีค่าเป็นหนึ่ง ตัวอย่างของรหัสที่มีคุณสมบัติ orthogonal กันเช่น X = 0011 และ Y = 0110

รหัส Walsh สามารถสร้างได้จาก Hadamard matrices H2m โดยที่ m คือ ค่า 2 ยกกำลังตัวเลขจำนวนเต็ม (m = 1,2,4,...) ค่าเริ่มต้นของ Hadamard matrices ที่ m = 1 กำหนดให้เป็น

$$H_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

และค่า H2m อื่นๆ สร้างจากความสัมพันธ์ต่อไปนี้

$$H_{2m} = \begin{bmatrix} Hm & Hm \\ Hm & \overline{Hm} \end{bmatrix}$$

ตัวอย่างเช่น กรณี m = 2 ได้

$$H_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

2.2.2.2 รหัส Long PN

รหัสประเภท Long PN สร้างจากชิฟต์รีจิสเตอร์ขนาด 42 บิต โดยต่อเชื่อมกันให้มีโครงสร้างเป็น maximum length LFSRs และใช้โพลิโนเมียลต่อไปนี้

$$P(x) = x^{42} + x^{35} + x^{33} + x^{31} + x^{27} + x^{26} + x^{25} + x^{22} + x^{21} + x^{19} + x^{18} + x^{17} + x^{16} + x^{10} + x^{7} + x^{6} + x^{5} + x^{3} + x^{2} + x + 1$$

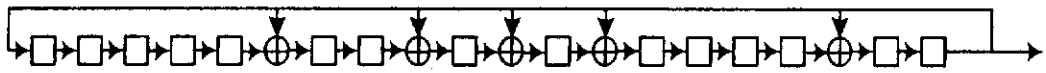
ผลที่ได้คือ รหัสที่มีความยาวถึง $2^{42} - 1 = 4.398 \times 10^{12}$ เนื่องจากอัตราการสร้างรหัสนี้มีค่าเท่ากับ 1.228 Mbps ดังนั้นรหัสนี้จะมีการซ้ำทุกๆ 41.425 วัน

2.2.2.3 รหัส Short PN

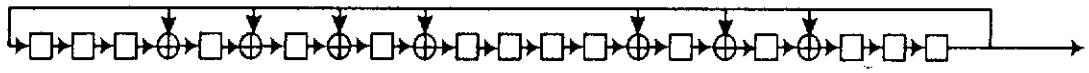
รหัส Short PN นี้มีอยู่ 2 ชุด คือ I และ Q โดยแต่ละชุดสร้างจากการใช้ชิฟต์รีจิสเตอร์ขนาด 15 บิต โดยต่อเชื่อมกันให้มีโครงสร้างเป็น maximum length LFSRs และใช้โพลิโนเมียล 2 ชุดต่อไปนี้

$$I(x) = x^{15} + x^{13} + x^{9} + x^{8} + x^{7} + x^{5} + 1$$

$$Q(x) = x^{15} + x^{12} + x^{11} + x^{10} + x^{6} + x^{5} + x^{4} + x^{3} + 1$$



ภาพที่ 2.5 แสดงวงจร 15 stage LFSR สำหรับ I



ภาพที่ 2.6 แสดงวงจร 15 stage LFSR สำหรับ Q

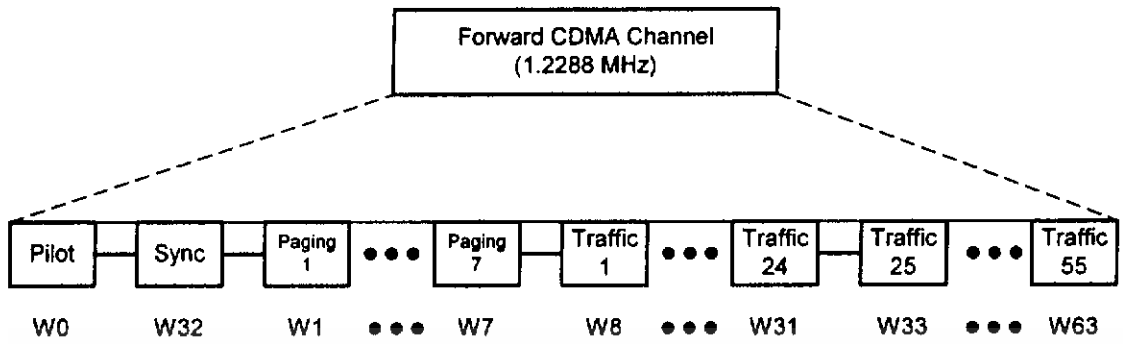
โครงสร้างของวงจรสร้างชุดรหัส Short PN มีลักษณะดังที่แสดงในภาพที่ 2.5 และภาพที่ 2.6 วงจรทั้งสองสามารถสร้างชุดรหัสที่มีความยาวเท่ากับ $2^{15} - 1 = 32,767$ จากนั้นจะมีเพิ่มบิต 0 อีกหนึ่งบิตเข้าไปเมื่อพบรหัสที่มีค่า 0 ติดต่อกัน 14 ตัว เพื่อให้ได้รหัสที่มีความยาวเท่ากับ 32,768 เนื่องจากอัตราการสร้างรหัสนี้มีค่าเท่ากับ 1.228 Mbps ดังนั้นรหัสนี้จะมีการซ้ำทุกๆ 26.66 ms

2.2.3 การส่งสัญญาณจากสถานีฐานไปหาโทรศัพท์ (Forward Link)

สัญญาณที่ส่งในทิศทางจากสถานีฐานไปยังเครื่องโทรศัพท์เคลื่อนที่หรือที่เรียกว่า Froward link นั้น ประกอบด้วยช่องสัญญาณทั้งหมด 4 ประเภท คือ

- ช่องสัญญาณไพลอต(Forward pilot channel)
- ช่องสัญญาณซิงโครไนเซชัน (Forward synchronization channel)
- ช่องสัญญาณเพจจิง (Forward paging channel)
- ช่องสัญญาณทราฟฟิก (Forward traffic channel)

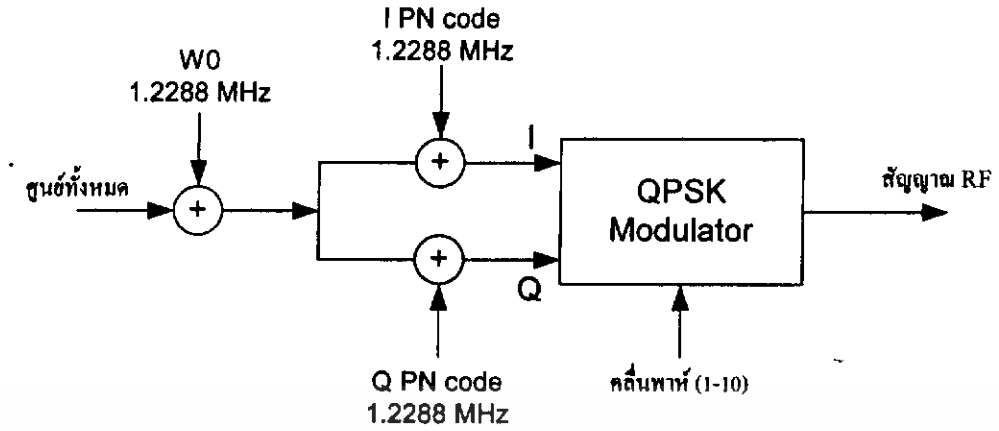
โดยคลื่นพาห์หนึ่งความถี่สามารถใช้ส่งสัญญาณทั้ง 4 ประเภทรวมกันได้ทั้งหมดไม่เกิน 64 ช่องสัญญาณ ทั้งนี้จะต้องมีช่องสัญญาณไพลอต (pilot channel) จำนวนหนึ่งช่องเสมอ มีช่องสัญญาณซิงโครไนเซชัน (synchronization channel) ไม่เกิน 1 ช่อง มีช่องสัญญาณเพจจิง (paging channel) ไม่เกิน 7 ช่อง และมีช่องสัญญาณทราฟฟิกอีกจำนวนหนึ่ง ช่องสัญญาณทั้ง 64 ช่องสามารถแยกออกจากกันได้โดย Walsh code 64 ชุด ดูตัวอย่างการแบ่งช่องสัญญาณที่มักจะมีการใช้งานตามปรกติในภาพที่ 2.7 สังเกตว่าช่องสัญญาณไพลอตจะใช้ Walsh code ชุดที่ 0 (W0)เสมอ ส่วนช่องสัญญาณซิงโครไนเซชัน ถ้ามีการใช้งานก็จะใช้ Walsh code ชุดที่ 32 (W32)เสมอ แต่ถ้าหากเป็นช่องสัญญาณเพจจิงก็จะมีการใช้ Walsh code ชุดที่ 1-7 (W1-W7) เรียงตามลำดับ และช่องสัญญาณทราฟฟิกจะใช้ W8-W31 และ W33-W64



ภาพที่ 2.7 การแบ่งช่องสัญญาณที่ส่งออกโดยสถานีฐาน

2.2.3.1 ช่องสัญญาณไพลอต(Forward Pilot Channel)

โดยปกติสถานีฐานของแต่ละเซลล์จะส่งสัญญาณไพลอต (pilot signal) ออกไปให้เครื่องโทรศัพท์ที่อยู่รอบ ๆ ข้างตลอดเวลา เพื่อใช้เป็นสัญญาณคลื่นพาห์อ้างอิงสำหรับการดีมอดูเลตแบบโคฮีเรนต์โดยที่ไม่มีส่วนของข้อมูลจากผู้ใช้มอดูเลตเข้าไปเลย การส่งสัญญาณนี้จะใช้กำลังส่งที่สูงกว่าสัญญาณประเภทอื่น ๆ ทั้งนี้เพื่อให้อุปกรณ์โทรศัพท์รอบข้างสามารถตรวจจับสัญญาณได้โดยง่าย ในการส่งสัญญาณไพลอตนั้นจะใช้ Walsh code ชุดที่ประกอบด้วยศูนย์ทั้งหมด 64 บิต (WO) เพราะฉะนั้นสัญญาณนี้ประกอบด้วยข้อมูลเพียงอย่างเดียวคือ Short PN code ของ I และ Q ซึ่งข้อมูลส่วนนี้ใช้เป็นตัวบ่งบอกได้ว่าสัญญาณไพลอตนี้ถูกส่งออกมาจากสถานีฐานใด ดังนั้นเครื่องโทรศัพท์เพียงแต่อ่านข้อมูลของ Short PN code จนครบหนึ่งคาบ (26.66 ms) ก็จะทราบว่าเป็นสัญญาณที่ได้รับนี้เป็นสถานีฐานใด ซึ่งโดยปกติแล้วทันทีที่มีการเปิดเครื่องโทรศัพท์เคลื่อนที่ตัวเครื่องโทรศัพท์จะทำการค้นหาและวัดกำลังของสัญญาณไพลอตที่ส่งออกจากสถานีฐานต่างๆ และเมื่อเครื่องโทรศัพท์พบสัญญาณที่มีกำลังสูงสุดก็จะทำการติดต่อกับสถานีฐานนั้นๆ ต่อไป ทุกสถานีฐานจะใช้ Short PN code ชุดเดียวกันหมดเพียงแต่จะมีการเลื่อนของชิปส์ออกไปด้วยจำนวนที่แตกต่างกัน ในการเลื่อนชิปส์นั้นจะเลื่อนออกไปทีละ 64 ชิปส์ เพราะฉะนั้นก็จะมีรหัสที่แตกต่างกันสำหรับการใช้งานได้ทั้งหมด 32,768/64 = 512 ชุด รหัสชุดหนึ่งอาจมีการนำมาใช้งานซ้ำในหลายๆ เซลล์ได้ หากแต่เซลล์เหล่านั้นจะต้องจัดให้อยู่ห่างกันมากเพียงพอที่จะไม่รบกวนซึ่งกันและกัน ภาพที่ 2.8 แสดงรายละเอียดขั้นตอนของการสร้างสัญญาณไพลอต

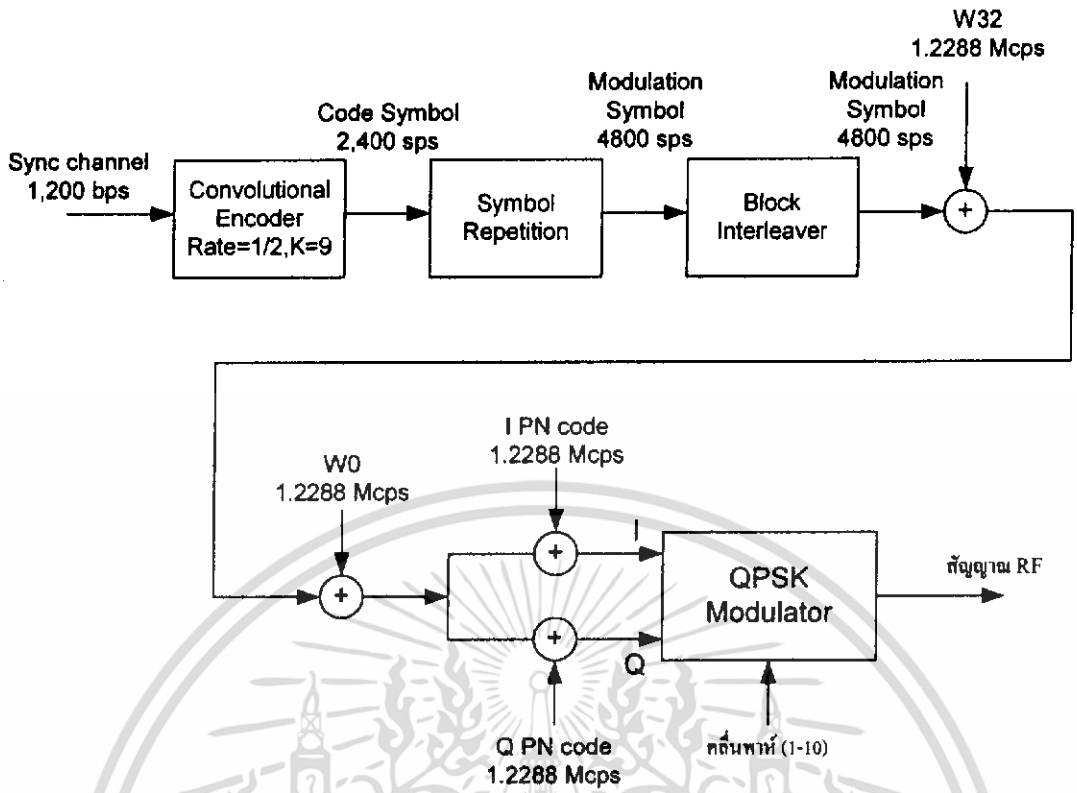


ภาพที่ 2.8 วงจรสร้างสัญญาณไพลอต

สัญญาณไพลอตยังมีประโยชน์สำหรับใช้ในการควบคุมกำลังส่งของสัญญาณ (power control) โดยทันทีที่เครื่องโทรศัพท์ได้รับสัญญาณไพลอตก็จะทำการคำนวณหาค่าเฉลี่ยของสัญญาณที่รับได้เพื่อใช้สำหรับประมาณค่ากำลังสูญเสียระหว่างเดินทางของสัญญาณ ซึ่งค่านี้จะนำมาใช้กำหนดขนาดกำลังที่เหมาะสมของสัญญาณที่โทรศัพท์จะใช้ในการส่งออก

2.2.3.2 ช่องสัญญาณซิงโครไนเซชัน (Forward Synchronization Channel)

ช่องสัญญาณซิงโครไนเซชัน (synchronization channel) มีไว้สำหรับส่งข้อมูลที่จำเป็นต่อการติดต่อในช่วงแรก เพื่อที่เครื่องโทรศัพท์จะได้สามารถซิงโครไนซ์กับสัญญาณที่ส่งออกจากสถานีฐานได้อย่างถูกต้อง ข้อมูลเหล่านี้ประกอบด้วย จังหวะเวลาในการส่งสัญญาณ หมายเลขประจำตัวของระบบโครงข่ายที่เครื่องโทรศัพท์ติดต่ออยู่ นอกจากนี้ก็ยังบอกถึงเวลาของนาฬิกาที่ระบบใช้อยู่ ดูโครงสร้างของวงจรที่ใช้ในการสร้างช่องสัญญาณซิงโครไนเซชันได้ในภาพที่ 2.9 ช่องสัญญาณนี้ใช้ Walsh code ชุดที่ W32 เป็นตัวบอกชนิดของช่องส่งสัญญาณ สังเกตว่าช่องสัญญาณซิงโครไนเซชันคล้ายกับช่องสัญญาณไพลอตตรงที่ไม่มีการใช้ long PN code เพราะเป็นสัญญาณที่แพร่ออกสำหรับโทรศัพท์เคลื่อนที่ทุกๆ เครื่อง ดูรายละเอียดของขั้นตอนการสร้างสัญญาณซิงโครไนเซชันได้ในภาพที่ 2.9



ภาพที่ 2.9 วงจรสร้างสัญญาณซิงโครไนเซชัน

จากภาพที่ 2.9 ในขั้นแรกบิตข้อมูลที่ส่งในช่องสัญญาณซิงก์จะถูกนำไปผ่านกระบวนการเข้ารหัสช่องสัญญาณคอนโวลูชันที่มีอัตราการเข้ารหัสเท่ากับ R = ½ และมี constraint span เท่ากับ 9 โดยชุดโพลิโนเมียลที่ใช้ประกอบด้วย

$$g_1(x) = 1 + x + x^2 + x^3 + x^5 + x^7 + x^8$$

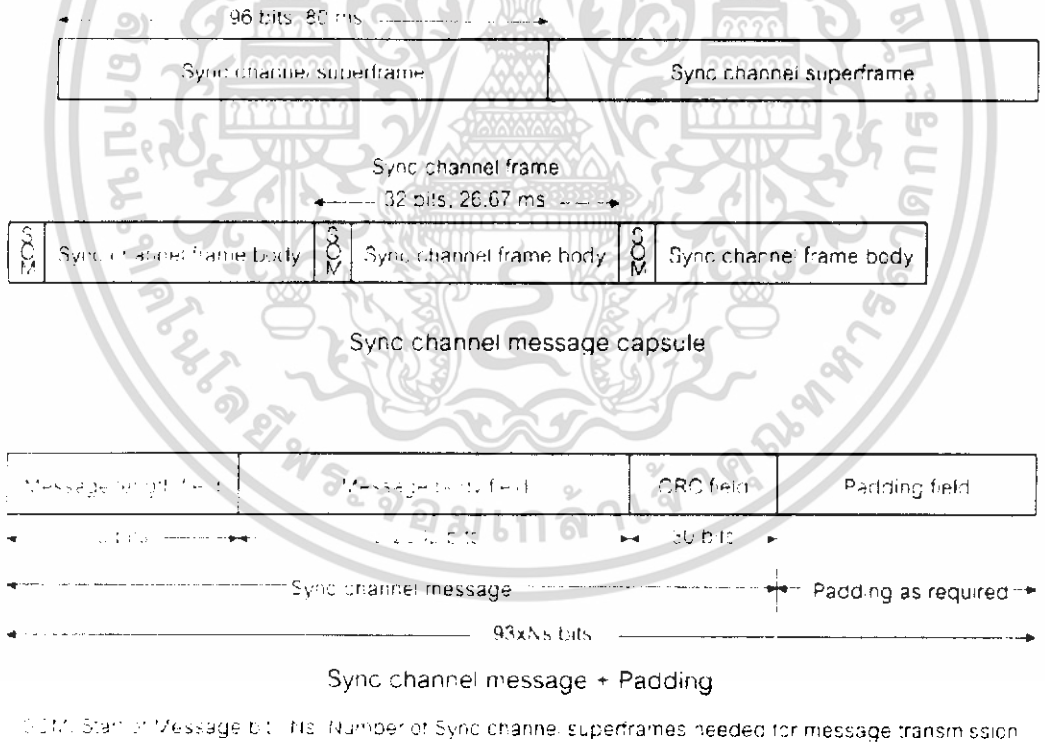
$$g_2(x) = 1 + x^2 + x^3 + x^4 + x^8$$

สำหรับโครงสร้างของวงจรเข้ารหัสคอนโวลูชันนี้ได้แสดงไว้ในภาพที่2.9หลังจากที่ผ่านการเข้ารหัสช่องสัญญาณแล้วอัตราบิตของข้อมูลเพิ่มขึ้นเท่าตัวเป็น 2.4 ksps ข้อมูลเหล่านี้จะถูกนำไปผ่านกระบวนการ symbol repetition เพื่อให้ได้ชุดข้อมูลที่มีอัตราเพิ่มสูงขึ้นเป็น 4.8 ksps จากนั้นนำไปเข้าวงจรอินเทอร์ลีฟ (block iinterleave) เพื่อลดผลกระทบของความผิดพลาดแบบเบิรสต์ ก่อนที่จะถูกสเปรดออกด้วย Walsh code ชุดที่ 32 ที่อัตราชิปส์ 1.2288 Mcps

อัตราบิตที่ใช้ในการส่งสัญญาณของช่องสัญญาณซิงโครไนเซชันมีค่าเท่ากับ 1,200 bps โดยรูปแบบโครงสร้างของข้อมูลที่จะส่งออกแบ่งเป็นช่วงเวลาที่เรียกว่าซูเปอร์เฟรม (superframe) ซึ่งมีขนาดเท่ากับ 80 ms และใช้ส่งทั้งหมด 96 บิต ดูภาพที่ 2.10 ประกอบ ในหนึ่ง

ซูเปอร์เฟรมจะบรรจุเฟรม (frame) ขนาด 32 บิตจำนวน 3 เฟรม ซึ่งแต่ละเฟรมจะมีความยาวเท่ากับ 26.66 ms ภายในแต่ละเฟรมประกอบด้วย 2 ส่วน คือ บิต SOM (Start of Message) และ Sync channel frame body ในส่วนของ SOM มีความยาว 1 บิต โดยมีหน้าที่เป็นตัวบ่งชี้ถึงว่าเฟรมดังกล่าวนี้บรรจุข้อความ (message) ชุดใหม่หรือไม่ นั่นถ้าคือ SOM=1 แสดงว่าเฟรมนี้บรรจุจุดเริ่มต้นของข้อความชุดใหม่ และถ้า SOM=0 แสดงว่าเป็นข้อความส่วนที่เหลือ ข้อความที่จะมาบรรจุลงในช่องสัญญาณซิงก์นี้มีรูปแบบดังที่แสดงในส่วนล่างของภาพที่ 2.10 ในส่วนของ 8 บิตแรก (message length field) มีไว้สำหรับบอกถึงความยาวของข้อความ จากนั้นตามด้วย message body field ซึ่งบรรจุข้อมูลต่างๆ ที่สถานีฐานต้องการจะส่งออกไปให้เครื่องโทรศัพท์ ในส่วนนี้มีขนาดความยาวที่ไม่แน่นอนโดยมีค่าระหว่าง 2-2002 บิต ส่วนถัดมาคือ CRC field ที่มีไว้สำหรับใช้ในการตรวจจับความผิดพลาดของการส่งข้อมูล ค่า CRC field คำนวณจาก message length field และ message body field โดยอาศัยโพลิโนเมียล

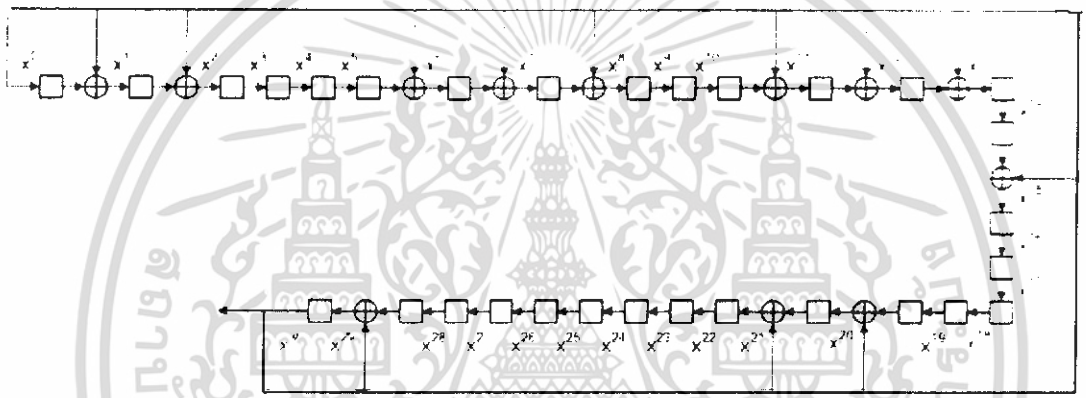
$$g(x) = 1 + x + x^2 + x^6 + x^7 + x^8 + x^{11} + x^{12} + x^{13} + x^{15} + x^{20} + x^{21} + x^{29} + x^{30}$$



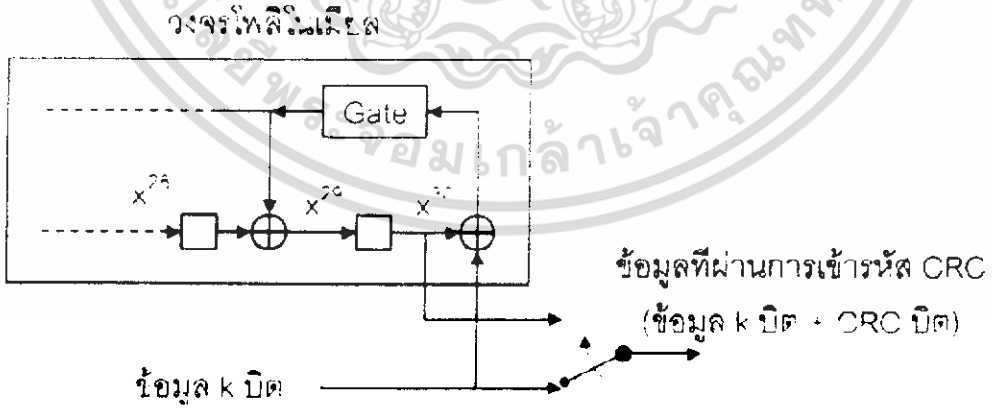
SOM: Start of Message bit; Ns: Number of Sync channel superframes needed for message transmission

ภาพที่ 2.10 โครงสร้างเฟรมและรูปแบบของข้อมูลที่ใช้ในช่องสัญญาณซิงก์

โครงสร้างของวงจรสำหรับโพลิโนเมียลข้างต้นมีลักษณะดังที่แสดงในภาพที่ 2.11 ส่วนวงจรเข้ารหัส CRC โดยรวมมีโครงสร้างดังในรูปที่ 2.12 ลำดับขั้นตอนการทำงานของวงจรเข้ารหัส CRC มีรายละเอียดดังนี้คือ ขั้นแรกระบบจะทำการตั้งค่าของชิฟต์รีจิสเตอร์แต่ละตัวให้มีค่าเป็น 1 ทั้งหมด จากนั้นปรับสวิตช์ที่ขาออกให้ตั้งอยู่ในตำแหน่งที่รับข้อมูลโดยตรงจากขาเข้าของวงจรและทำการเชื่อมวงจร gate เข้าด้วยกัน หลังจากที่ระบบได้ทำงานไปเป็นจำนวน k รอบ ให้ทำการปรับตำแหน่งของสวิตช์ที่ขาออกไปต่อเชื่อมกับส่วนของวงจรโพลิโนเมียลแทน พร้อมกันนั้นให้ทำการเปิด gate ออก และปล่อยให้วงจรทำงานในลักษณะนี้ไปอีก 30 รอบ ซึ่งในช่วงนี้เองที่ระบบจะได้ค่าของ CRC ออกมาจำนวนทั้งสิ้น 30 บิตที่ขาออก ทั้งนี้ข้อมูล k บิตที่กล่าวมานี้ประกอบด้วย message length field จำนวน 8 บิตและ message body field อีกจำนวน 2-2002 บิต



ภาพที่ 2.11 โครงสร้างของโพลิโนเมียลที่ใช้ในการเข้ารหัส CRC ของสัญญาณซิงก์

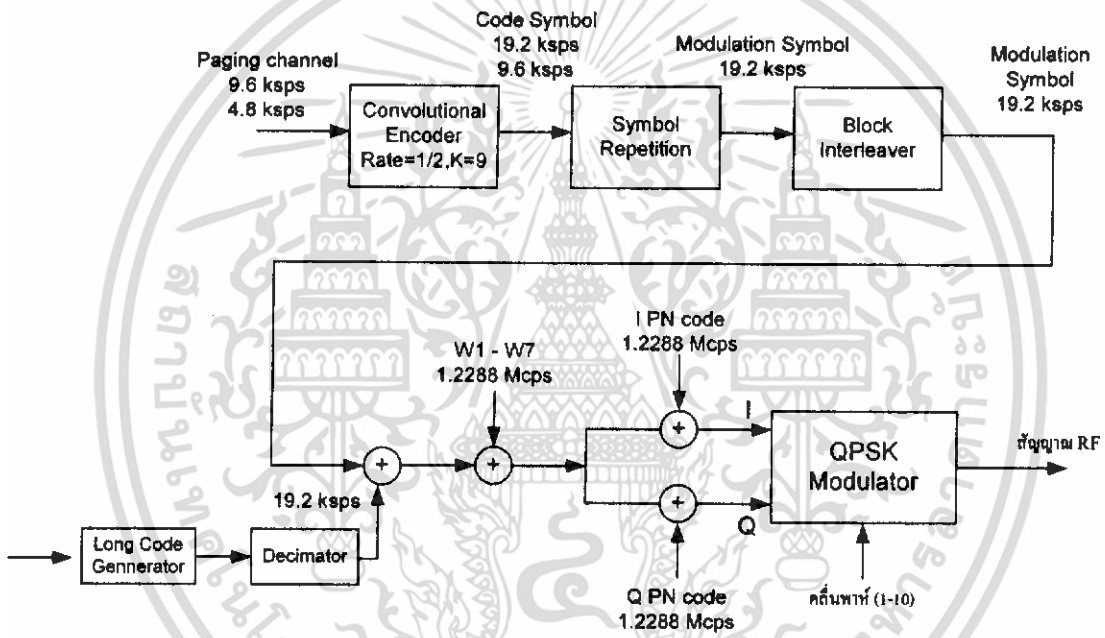


ภาพที่ 2.12 โครงสร้างของวงจรเข้ารหัส CRC ของสัญญาณซิงก์

62680

2.2.3.3 ช่องสัญญาณเพจจิง (Forward Paging Channel)

ช่องสัญญาณเพจจิง (paging channel) มีไว้สำหรับให้สถานีฐานใช้ในการเพจ (page) หรือค้นหาตำแหน่งของเครื่องโทรศัพท์ที่ต้องการจะติดต่อด้วย และหลังจากที่สถานีฐานสามารถติดต่อกับเครื่องโทรศัพท์ที่ต้องการได้แล้วก็จัดสรรช่องสัญญาณทราฟฟิก (traffic channel) สำหรับใช้ในการติดต่อรับส่งข้อมูลของผู้ใช้ต่อไป อัตราบิตข้อมูลที่ใช้ส่งมี 2 ขนาด คือ 9,600 หรือ 4,800 bps โดยมีขนาดความยาวของเฟรมเท่ากับ 20 ms ชุด Walsh code ที่ใช้คือ W1-W7 (หมายเหตุ: ในกรณีที่ชุดรหัส W1-W7 มิได้มีการใช้งานอยู่ระบบก็สามารถที่จะใช้รหัสนี้ในการส่งสัญญาณข้อมูลของผู้ใช้ได้คือ นำมาใช้เป็นช่องสัญญาณทราฟฟิก (traffic channel) แทน ดูขั้นตอนการสร้างสัญญาณเพจจิงได้ในภาพที่ 2.13



ภาพที่ 2.13 ขั้นตอนการสร้างสัญญาณเพจจิง

ในขั้นแรกสัญญาณเพจจิงจะถูกนำมาใช้เข้ารหัสช่องสัญญาณคอนโวลูชันที่มีอัตราการเข้ารหัสเท่ากับ R = ½ และมี constraint span เท่ากับ 9 โดยชุดโพลิโนเมียลที่ใช้ประกอบด้วย

$$g(x) = 1 + x + x^2 + x^3 + x^5 + x^7 + x^8$$

$$g(x) = 1 + x^2 + x^3 + x^4 + x^8$$

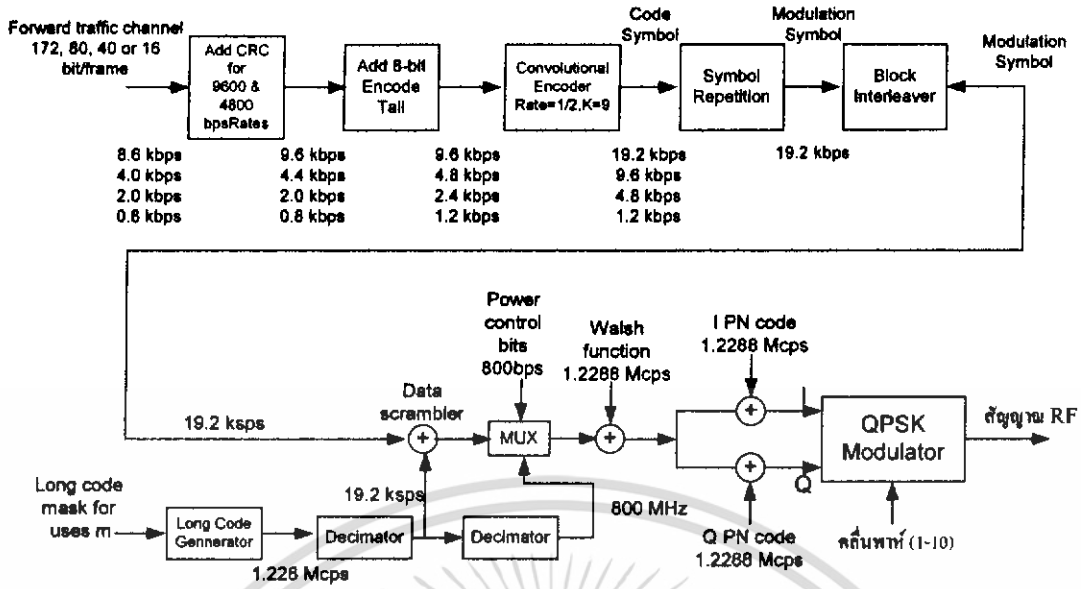
ผลที่ได้คือข้อมูลที่มีอัตราการบิตเพิ่มขึ้น 2 เท่า คือ เท่ากับ 9,600 sps และ 19.2 ksps จากนั้นนำไปผ่านวงจร Symbol Repetition เพื่อให้อัตราการส่งของทั้ง 2 ประเภทมีค่าเท่ากัน เป็น 19.6 ksps และทำการอินเทอร์ลีฟข้อมูลคราวละ 20 ms เพื่อลดผลกระทบของความผิดพลาดแบบเบิรสต์ที่อาจเกิดขึ้นจากช่องสัญญาณ

สัญญาณที่ออกจากวงจรอินเทอร์ลีฟจะถูกนำไปสแกรมเบิล (scramble) โดยการนำไปบวก (modulo-2) กับลำดับสัญญาณที่สร้างขึ้นจากวงจร long code generator และ decimator ส่วนของวงจร long code generator มีลักษณะโครงสร้างดังแสดงในภาพที่ 2.13 ภายในวงจรประกอบด้วยส่วนของ 42-bit Long Code Mask ที่ทำหน้าที่ในการกำหนดรูปแบบของการสแกรมเบิลข้อมูล รูปแบบของ *code mask* ชุดนี้มีโครงสร้างดังที่แสดงในภาพที่ 2.13 *long code generator* นี้ให้กำเนิดลำดับสัญญาณที่มีอัตราบิตอยู่ที่ 1.2288 kcps ดังนั้นเมื่อผ่านเข้าสู่วงจร decimator ซึ่งทำหน้าที่ในการดึงเฉพาะบิตแรกของลำดับสัญญาณที่มีอัตราบิตลดลงเหลือเท่าๆ กับสัญญาณที่ออกจากวงจรอินเทอร์ลีฟคือ 19.2 ksps เพื่อนำไปทำการสแกรมเบิลต่อไป

สำหรับขั้นตอนที่เหลือก็จะมีลักษณะที่คล้ายกับช่องสัญญาณ 2 ประเภทแรกที่ได้กล่าวไว้ก่อนหน้า โดยใช้ Walsh code ชุดที่ W1-W7

2.2.3.4 ช่องสัญญาณทราฟฟิก(Forward Traffic Channel)

ช่องสัญญาณทราฟฟิก (traffic channel) มีไว้สำหรับรับส่งสัญญาณเสียงและสัญญาณอื่น ๆ ตลอดช่วงเวลาที่มีการใช้งานของผู้ใช้ Walsh code ชุดที่สามารถนำมาใช้งานได้ก็คือ W8-W31, W33-W63 และ W1-W7 (ใช้เฉพาะเวลาที่มิได้นำไปใช้งานอย่างอื่น) อัตราบิตข้อมูลที่ใช้ในการส่งมีทั้งหมด 4 แบบ คือ 1,200 2,400 4,800 และ9,600bps โดยที่ขนาดของหนึ่งเฟรมมีความยาวเท่ากับ 20 ms



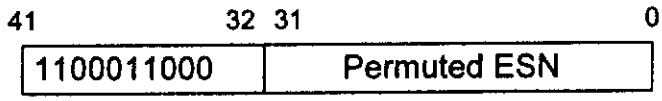
ภาพที่ 2.14 ขั้นตอนการสร้างช่องสัญญาณทราฟฟิก

ภาพที่ 2.14 แสดงขั้นตอนการสร้างช่องสัญญาณทราฟฟิก จะเห็นว่าในขั้นตอนแรก ๆ จะมีลักษณะคล้ายกับช่องสัญญาณประเภทอื่นที่ได้อธิบายไว้แล้วคือ มีการเข้ารหัสช่องสัญญาณแบบคอนโวลูชัน การทำ Symbol repetition และการทำอินเทอร์ลีฟ หลังจากนั้นสัญญาณที่ได้ก็จะนำมาทำการสแกรมเบิลเหมือนในกรณีของช่องสัญญาณเพจจิง หากแต่ในกรณีของช่องสัญญาณทราฟฟิกจะมีการใช้ long code mask ที่แตกต่างไป โดยโครงสร้างที่ใช้มีลักษณะดังแสดงในภาพที่ 2.15 ส่วนของ Permuted ESN นั้นได้มาจากการสลับตำแหน่งของบิตใน ESN แสดงในภาพที่ 2.16 โดยมีลำดับดังนี้คือ

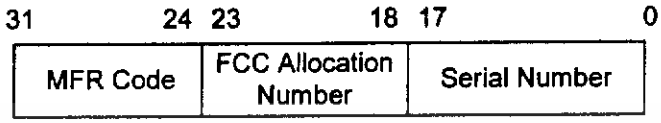
Permuted ESN = $\{E_0, E_{31}, E_{22}, E_{13}, E_4, E_{26}, E_{17}, E_8, E_{30}, E_{21}, E_{12}, E_3, E_{25}, E_{16}, E_7, E_{29}, E_{20}, E_{11}, E_2, E_{24}, E_{15}, E_6, E_{28}, E_{19}, E_{10}, E_1, E_{23}, E_{14}, E_5, E_{27}, E_{18}, E_9\}$

เหตุผลที่ต้องมีการสลับตำแหน่งก็เพื่อลดความสัมพันธ์ (correlation) ระหว่าง long code แต่ละชุด

ข้อมูลที่ส่งผ่านช่องสัญญาณทราฟฟิกนี้จะได้รับการปกป้องจากการดักฟังได้โดยการใช้ private long code mask แทน public long code mask ที่ได้อธิบายไว้ ซึ่งโดยปกติแล้วการที่ระบบจะเปลี่ยนไปใช้ private code mask ได้นั้นจะต้องมีการทำกระบวนการ authentication ก่อนเสมอ



ภาพที่ 2.15 รายละเอียดรูปแบบของ Forward traffic channel public long code mask



ภาพที่ 2.16 รายละเอียดรูปแบบของ ESN

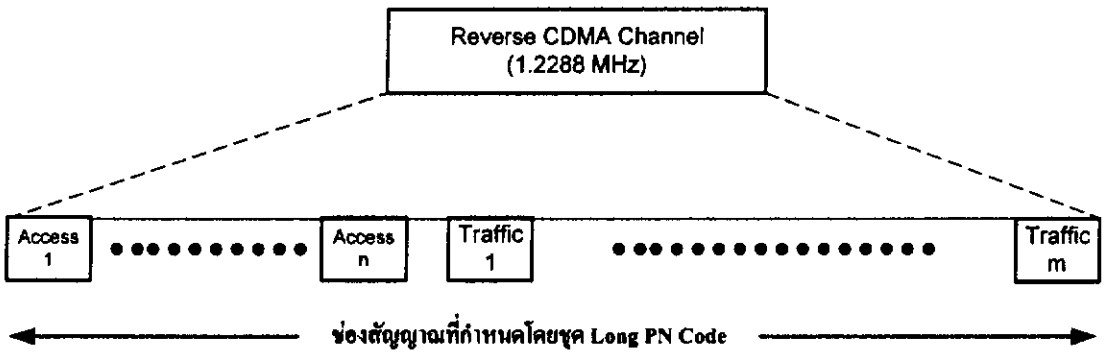
นอกจากนี้ระบบยังมีการแทรกบิตพิเศษลงในช่องสัญญาณทราฟฟิกสำหรับใช้ในการควบคุมกำลังส่งของเครื่องโทรศัพท์ให้อยู่ในระดับเหมาะสม โดยในการส่งนั้นจะส่งเพียง 1 บิต ทุกๆ ช่วงเวลา 1.25 ms (800 bps) หากบิตที่ส่งออกมีค่าเป็น 0 ก็หมายความว่าเครื่องโทรศัพท์จะต้องทำการปรับค่าเฉลี่ยกำลังของกำลังส่งขึ้นหนึ่งระดับ แต่หากบิตที่ส่งออกมีค่าเป็น 1 เครื่องโทรศัพท์ก็จะต้องปรับลดกำลังส่งลง

2.2.4 การส่งสัญญาณจากโทรศัพท์ไปหาสถานีฐาน (Reverse Link)

สำหรับการส่งสัญญาณในทิศทางจากโทรศัพท์เคลื่อนที่ไปยังสถานีฐานหรือที่เรียกว่า Reverse link นั้นมีความซับซ้อนกว่าในทิศทางตรงกันข้าม เพราะกำลังของสัญญาณที่เครื่องโทรศัพท์ส่งออกมีขนาดอ่อนกว่าเมื่อเทียบกับกำลังของสัญญาณที่ส่งออกจากสถานีฐาน ดังนั้นจึงต้องมีการเข้ารหัสช่องสัญญาณที่ซับซ้อนกว่า อีกทั้งที่สถานีฐานยังมีการใช้สายอากาศหลายชุดเพื่อช่วยเพิ่มสมรรถภาพในการรับสัญญาณ การส่งสัญญาณในทิศทางนี้มีเพียง 2 ประเภท คือ

- ช่องสัญญาณแอกเซส (Access channel)
- ช่องสัญญาณทราฟฟิกขากลับ (Reverse traffic channel)

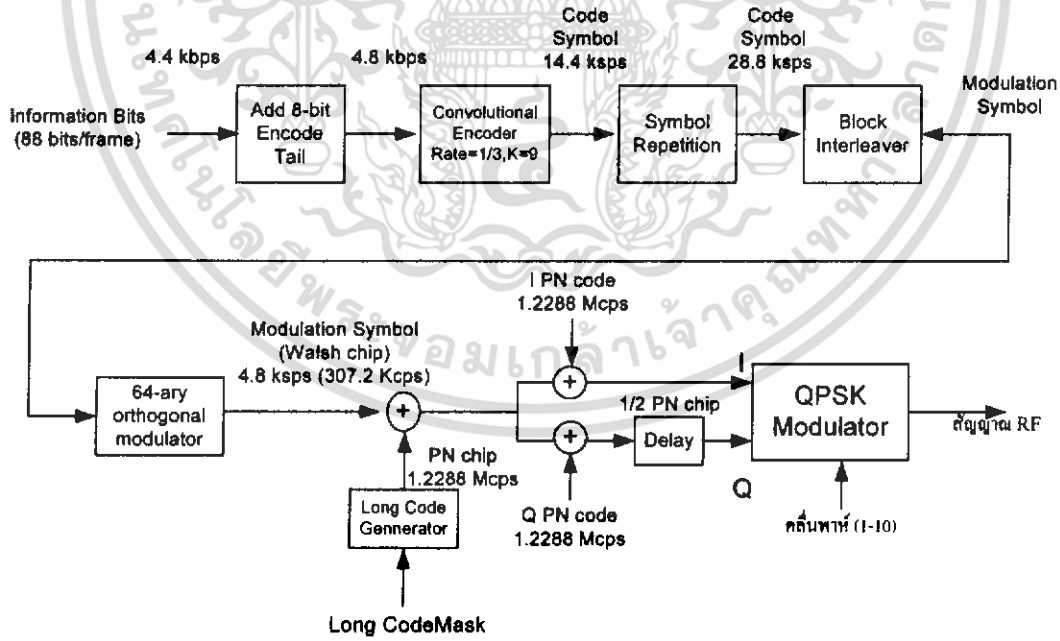
ดูลักษณะของการแบ่งช่องสัญญาณในทิศทาง Reverse link ได้ในภาพที่ 2.17 สังเกตว่าในทิศทาง reverse link นั้นอาศัยชุดรหัส long PN เป็นตัวระบุประเภทของช่องสัญญาณซึ่งแตกต่างจากในกรณีของ forward link ที่ใช้รหัส Walsh เป็นตัวบ่งบอกถึงประเภทของช่องสัญญาณ



ภาพที่ 2.17 การจัดช่องสัญญาณที่ส่งออกโดยเครื่องโทรศัพท์เคลื่อนที่

2.2.4.1 ช่องสัญญาณแอกเซส (Access channel)

ในส่วนของช่องสัญญาณแอกเซส (Access channel) มีหน้าที่หลักคือใช้ในการส่งสัญญาณซิกแนลลิงออกไปที่สถานีฐานเพื่อขอเริ่มการติดต่อในกรณีที่ผู้ใช้ต้องการจะโทรออก และใช้ช่องสัญญาณนี้ในการโต้ตอบกับสัญญาณซิกแนลลิงที่ส่งออกมาจากสถานีฐานผ่านทางช่องสัญญาณเพจจิง อัตราการส่งของช่องสัญญาณ access กำหนดเป็นค่าคงที่ไว้ที่ 4,800 bps โดยรูปแบบในการส่งนั้นใช้เฟรมขนาดเท่ากับ 20 ms ลงตัว สัญญาณที่จะส่งออกผ่านช่องสัญญาณ access จะต้องผ่านกระบวนการต่าง ดังที่แสดงในภาพที่ 2.18



ภาพที่ 2.18 ขั้นตอนการสร้างการสร้างสัญญาณสำหรับส่งผ่านช่องสัญญาณแอกเซสในทิศทางที่ส่งจากโทรศัพท์เคลื่อนที่ไปยังสถานีฐาน

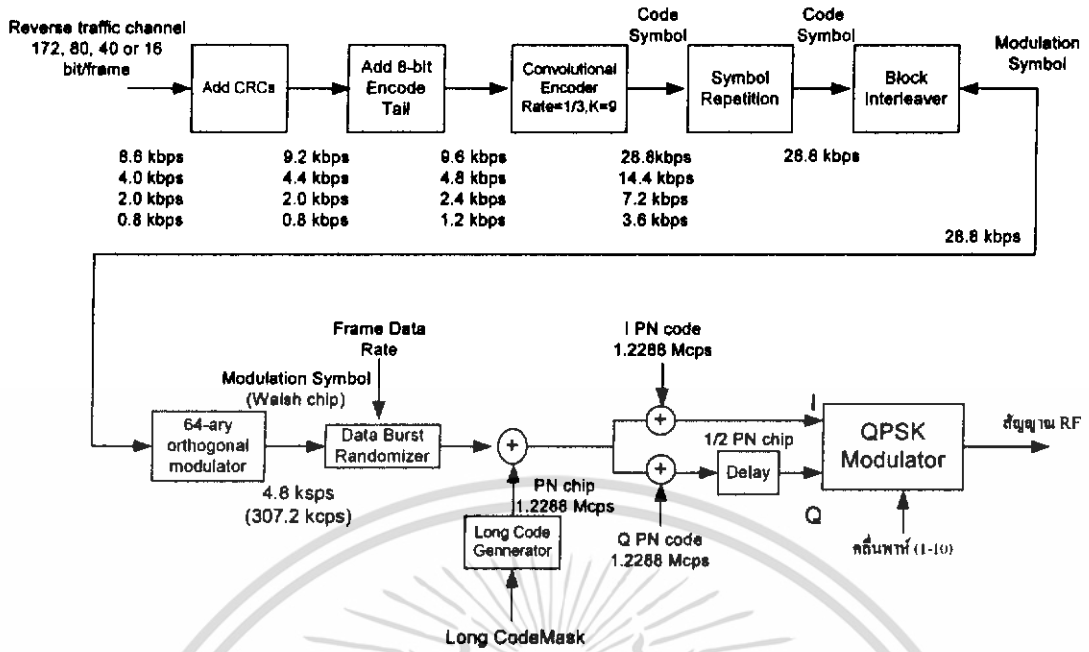
โครงสร้างของเฟรมแบ่งออกเป็น 2 ส่วน คือinformation bitsจำนวน 88 บิต และencoder tailอีก 8 บิต ดูรูปที่11.18 ประกอบ ดังนั้นอัตราบิตที่ส่งมีค่าเท่ากับ (88+8)/20 ms = 4,800 bps จากนั้นจะนำไปผ่านการเข้ารหัสแบบคอนโวลูชันทีละ 88 บิต โดยอาศัยวงจรในรูปที่11.19 ซึ่งเป็นวงจรเข้ารหัสที่มีอัตราการเข้ารหัสเท่ากับ 1/3 และมี constraint length เท่ากับ 9 ส่วนของencode tail จำนวน 8 บิตจะกำหนดให้มีค่าเป็นศูนย์ทั้งหมดก่อนที่จะทำการเข้ารหัสจากในรูปจะเห็นว่าวงจรเข้ารหัสที่ใช้ประกอบด้วยชิฟต์รีจิสเตอรทั้งหมด 8 ตัว วงจรบวกมอดูโล 2 อีก 3 ตัว และมีอุปกรณ์สวิตช์อีก 1 ตัวซึ่งทำหน้าที่วนรอบรับค่าจากผลลัพธ์ที่ได้จากการวงจรบวกมอดูเลต 2 ทีละตัว หลังจากที่ผ่านการเข้ารหัสแล้วอัตราบิตข้อมูลจะสูงขึ้น 3 เท่ากลายเป็น 14,400 bps

2.2.4.2 ช่องสัญญาณทราฟฟิกขากลับ (Reverse traffic channel)

ภาพที่ 2.19 แสดงโครงสร้างขั้นตอนของการสร้างช่องสัญญาณทราฟฟิกที่จะส่งจากโทรศัพท์เคลื่อนที่ไปยังสถานีฐาน วงจรเข้ารหัสสัญญาณเสียงที่ใช้ใน IS-95 จะกำหนดแพ็กเกต (packet) จำนวนหนึ่งแพ็กเกตทุกๆ 20 ms โดยที่ขนาดความยาวแพ็กเก็ตมีค่าไม่คงที่ทั้งนี้ขึ้นอยู่กับปริมาณการพูดของผู้ใช้โทรศัพท์ โดย IS-95 ได้แบ่งชนิดของแพ็กเกตออกเป็นทั้งหมด 4 แบบ ดังแสดงไว้ในตารางที่ 2.3 สัญญาณที่ได้จากที่แสดงในตารางที่ 2.3 จะนำไปเข้ารหัสแบบคอนโลลูชันที่มีอัตราการเข้ารหัสเท่ากับ 1/3 และมีค่า Constraint length เท่ากับ 9 ซึ่งทำให้จำเป็นเพื่อเพิ่ม Code tail อีก 8 บิตต่อท้ายดังแสดงในตารางเดียวกัน สังเกตว่ากรณีที่อัตราการส่งข้อมูล 4,800 และ 9,600 bps จะมีการใช้รหัส CRC เพิ่มเติมพิเศษเพื่อช่วยในการตรวจสอบว่ามีความผิดพลาดในข้อมูลเกิดขึ้นหรือไม่ วงจรเข้ารหัสคอนโวลูชันที่ใช้มีโครงสร้างเช่นเดียวกันกับในกรณีของช่องสัญญาณแอกเซสดังที่แสดงไว้ในภาพ 2.19

ตารางที่ 2.3 โครงสร้างของแพ็กเกต

Information (bits)	Signaling (bits)	CRC (bits)	Code tail (bits)	Total (bits)	Bit rate (bps)
171	1	12	8	192	9,600
80	0	8	8	96	4,800
40	0	0	8	48	2,400
16	0	0	8	24	1,200



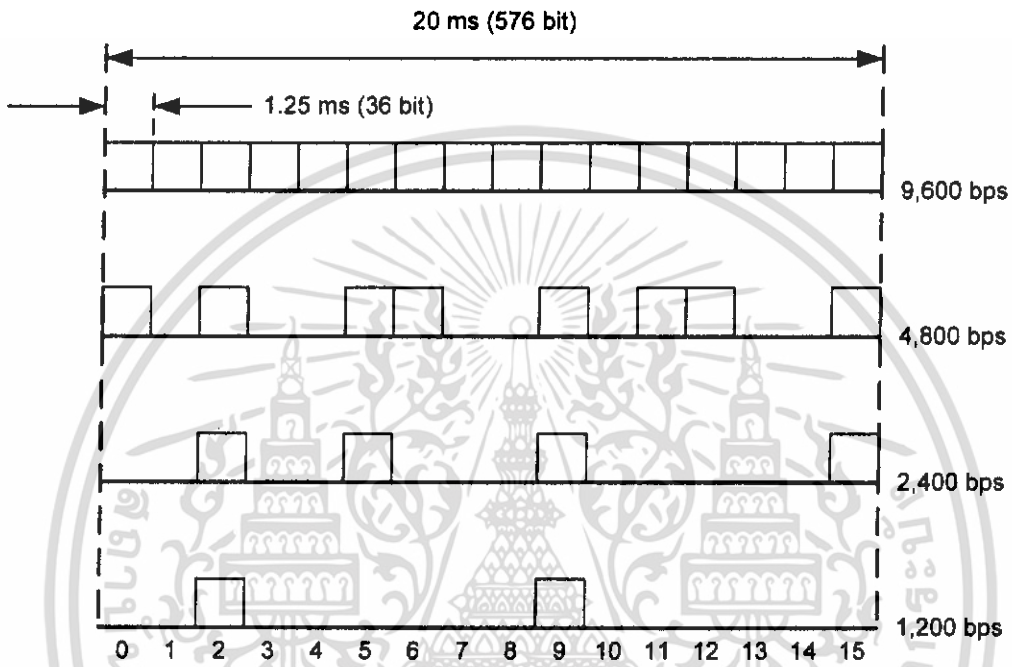
ภาพที่ 2.19 ขั้นตอนการสร้างสัญญาณทราฟฟิกในทิศทางจากโทรศัพท์เคลื่อนที่ไปยังสถานีฐาน

สัญญาณที่ได้จากวงจรเข้ารหัสช่องสัญญาณคอนโวลูชันนั้นจะนำไปเข้าวงจร symbol repetition ซึ่งทำหน้าที่ เพิ่มบิตข้อมูลซ้ำเพื่อให้อัตราบิตที่ขาออกมีค่าเท่ากับ 28.8 kbps เพราะฉะนั้นในกรณีที่อัตราบิตขาเข้ามีอัตราบิตเป็น R จะมีการส่งบิตนั้นซ้ำอีก 9,600/R-1 ครั้ง เช่น ถ้าอัตราบิตขาเข้ามีค่าเป็น 4,800 bps จะมีการส่งแต่ละบิตซ้ำอีกหนึ่งบิต จากนั้นสัญญาณที่ได้นี้จะนำไปทำการอินเทอร์ลีฟด้วยช่วงความลึกเท่ากับความยาวของหนึ่งเฟรมหรือ 20 ms เนื่องจากอัตราบิตก่อนการทำอินเทอร์ลีฟมีค่าเท่ากับ 28.8 kbps ดังนั้นในช่วงเวลา 20 ms จะมีจำนวนบิตทั้งสิ้น 576 บิต(32x18) ซึ่งจะนำมาทำการอินเทอร์ลีฟด้วยวิธีการเดียวกับในกรณีของช่องสัญญาณแอกเซสที่แสดงไว้ก่อนหน้านี้ในภาพที่ 2.19

ในลำดับถัดมาคือในส่วนของวงจร 64-ary orthogonal modulator จะแปลข้อมูลทีละ 6 บิตให้เป็น Walsh code ที่มีความยาวเท่ากับ 64 ชิปส์ การที่ใช้ 6 บิตก็เนื่องมาจากการที่ Walsh code ที่ใช้มีทั้งหมด $2^6 = 64$ ชุดนั่นเอง ผลที่ได้ก็คือข้อมูลที่มีอัตราชิปส์เท่ากับ 28,800/6x14 =307.2 k Walsh cps (kcps) หรือได้ 4.8 k Walsh sps สังเกตว่าการนำ Walsh code มาใช้งานในการส่งข้อมูลในทิศทางจากเครื่องโทรศัพท์ไปยังสถานีฐาน (reverse link) แตกต่างจากการใช้งานรหัสนี้ในการส่งข้อมูลในทิศตรงข้าม (forward link) อย่างชัดเจน ในกรณีของ forward link รหัสเหล่านี้ใช้สำหรับแบ่งแยกช่องสัญญาณออกเป็น 64 ช่อง

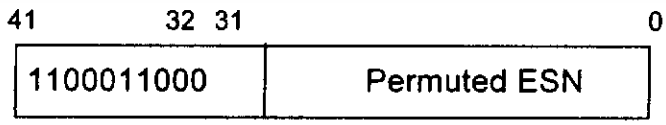
สัญญาณที่ได้จากวงจร 64-ary orthogonal modulator จะนำไปเข้าวงจร Data burst randomizer ซึ่งทำหน้าที่เปิดและปิดเกตเพื่อตัดที่สัญญาณที่ซ้ำซ้อนออกไป ซึ่งหมายความว่าวงจรส่วนนี้จะมีผลกระทบต่อสัญญาณก็เฉพาะกรณีที่อัตราบิตของข้อมูลมีค่าต่ำกว่า 9,600 bps เท่านั้น

โดยจะมีการเปิดและปิดแกตในทุก ๆ ช่วงเวลา 1.25 ms หรือทุก ๆ 36 บิต ช่วงเวลาการเปิดปิดจะมีลักษณะดังที่แสดงในรูปที่ 2.20 จะเห็นได้ว่า duty cycle ของการส่งที่อัตราบิต 9,600, 4,800, 2,400, และ 1,200bps จะมีค่าเท่ากับ100, 50, 25, และ12.5 % ตามลำดับ การที่ไม่มีการส่งสัญญาณออกไปในบางช่วงเวลานั้นมีประโยชน์ เพราะจะช่วยลดปริมาณสัญญาณรบกวนภายในระบบลงได้ ซึ่งจะส่งผลโดยตรงให้ระบบสามารถรองรับจำนวนผู้ใช้บริการได้มากขึ้น



ภาพที่ 2.20 ตัวอย่างการเปิดปิดเกตสำหรับการส่งที่อัตราบิตค่าต่างๆ

ในขั้นตอนถัดมาสัญญาณที่ได้จะนำไปสเปรดออกด้วยรหัส long PN ที่มีอัตราชิปส์เท่ากับ 1.2288 Mcps โดยอาศัยวงจรแบบเดียวกับในกรณีของช่องสัญญาณแอกเซส ซึ่งการสเปรดสัญญาณนั้นโทรศัพท์แต่ละเครื่องจะใช้ชุด public long code mask ที่แตกต่างกัน ดูโครงสร้างรายละเอียดของ public long code mask ที่ใช้ในช่องสัญญาณ forward traffic ได้ในภาพที่ 2.21



ภาพที่ 2.21 รายละเอียดรูปแบบของ Public long code mask

2.2.5 คุณลักษณะและข้อดีของระบบ CDMA

- โดยปกติในในระหว่างการสนทนาของมนุษย์จะมีเพียง 35 % ของเวลาทั้งหมดเท่านั้นที่มีสัญญาณเสียงของผู้พูด ส่วนเวลาที่เหลือจะเป็นการฟัง ในระบบ CDMA นั้นเนื่องจากผู้ใช้บริการทุกคนจะถูกจัดให้ใช้ช่องสัญญาณช่องเดียวกันหมด ดังนั้นถ้ามีผู้ใช้บางส่วนที่ไม่มีการพูดก็หมายถึงว่าปริมาณสัญญาณรบกวนระหว่างกันของผู้ใช้ ณ เวลานั้นๆ ก็ลดลงด้วย ซึ่งโดยหลักการแล้วอาจจะลดลงได้ถึง 65 % นั่นหมายถึงว่าเราสามารถเพิ่มขนาดความจุของระบบขึ้นถึง 3 เท่า

- ในการส่งข้อมูลที่อัตราการส่งสูงๆ เช่นมากกว่า 10 kbps มากๆ ในระบบ FDMA และ TDMA จำเป็นต้องอาศัยวงจรอีควอไลเซอร์ในการลดสัญญาณที่เรียกว่า intersymbol interference ซึ่งเกิดจากการแผ่ของสัญญาณเนื่องจากการสเปรดของรีเลย์ ในขณะวงจรภาครับของระบบ CDMA ไม่มีความจำเป็นต้องใช้วงจรอีควอไลเซอร์สำหรับการปรับแก้รูปสัญญาณ แต่ต้องอาศัยวงจรคอร์รีเลย์เตอร์ (correlator) เพื่อใช้ในการดีสเปรด (despread) สัญญาณกลับมาซึ่งคอร์รีเลย์เตอร์นั้นมีความซับซ้อนน้อยกว่าวงจรอีควอไลเซอร์

- ระบบ CDMA ใช้วงจรรับส่งวิทยุเพียงชุดเดียวสำหรับครอบคลุมบริเวณๆ หนึ่งนั้นจึงง่ายและสะดวกในการติดตั้ง อีกทั้งยังประหยัดเนื้อที่ในการจัดเก็บอุปกรณ์ด้วย

- เนื่องจากในแต่ละเซลล์ใช้คลื่นวิทยุความถี่เดียวกันจึงไม่จำเป็นต้องมีการแฮนด์ออฟ จากความถี่หนึ่งไปอีกความถี่หนึ่งในกรณีที่ตัวโทรศัพท์เคลื่อนที่จากเซลล์หนึ่งไปสู่เซลล์ข้างเคียง การแฮนด์ออฟแบบนี้เรียกว่า soft handoff ซึ่งต่างจาก hard handoff ตรงที่ไม่มีความจำเป็นต้องเปลี่ยนความถี่ของคลื่นพาห์

- ในระบบ TDMA มีความจำเป็นต้องเว้นช่องว่างที่เรียกว่า guard time ไว้ขนาดหนึ่งเพื่อป้องกันไม่ให้ข้อมูลที่อยู่คนละไทม์สล็อตเกิดการทับกัน ส่วนระบบ TDMA นั้นไม่จำเป็นต้องมี guard time เลย เพราะฉะนั้นช่วงเวลาตรงนี้สามารถนำมาใช้เพิ่มประสิทธิภาพการส่งข้อมูลให้สูงขึ้นได้

- ระบบ CDMA ไม่มีความจำเป็นต้องจัดแบ่งความถี่สำหรับการใช้งานที่ต่างกันสำหรับเซลล์ที่อยู่ใกล้กัน นั่นคือเซลล์ที่อยู่ติดกันก็สามารถใช้คลื่นความถี่เดียวกันได้ ซึ่งต่างจากระบบอื่น เช่น FDMA เพราะระบบเหล่านี้ต้องมีการจัดแบ่งความถี่ออกเป็นกลุ่มๆ และเซลล์ที่ใช้คลื่นความถี่เดียวกันจำเป็นต้องอยู่ห่างกันเพียงพอเพื่อลดผลกระทบของ Cp – Channel Interference ดังนั้นการใช้ประโยชน์จากสเปกตรัมความถี่ที่มีอยู่อย่างจำกัดของระบบ CDMA จึงมีประสิทธิภาพที่ดีกว่า ให้ขนาดความจุที่สูงกว่า และรองรับจำนวนผู้ใช้บริการได้มากกว่า

- ปัญหาและผลกระทบของเฟดดิงต่อการส่งสัญญาณแบบแถวความถี่กว้าง (wideband) มีน้อยกว่าเมื่อเทียบกับการส่งข้อมูลแบบแถบความถี่แคบ (narrowband) ดังนั้นการออกแบบระบบเพื่อจัดการปัญหาของเฟดดิงจึงมีความซับซ้อนน้อยกว่า

2.3 การติดต่อผ่านแบบอนุกรม

2.3.1 พื้นฐานการสื่อสารแบบอนุกรม

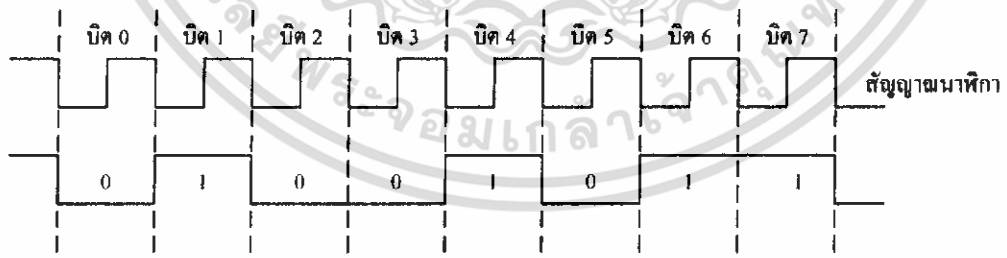
ถึงแม้ว่าการสื่อสารแบบอนุกรมในเครื่องคอมพิวเตอร์นั้นจะมีความเร็วในการสื่อสารช้ากว่าแบบขนาน ทั้งนี้ก็เพราะว่าการเคลื่อนย้ายข้อมูลแบบอนุกรมเป็นการส่งข้อมูลทีละบิตแต่พอร์ตขนานส่งข้อมูลได้ครั้งละหลายๆ บิตพร้อมกัน ส่งผลให้การสื่อสารข้อมูลแบบอนุกรมมีความเร็วต่ำกว่าแบบขนาน

แต่ว่าการส่งข้อมูลแบบอนุกรมนั้นมีข้อที่เหนือกว่าการส่งข้อมูลแบบขนานคือ การสามารถส่งข้อมูลได้ระยะทางที่ไกลกว่าแบบขนาน อีกทั้งสายสัญญาณที่ใช้ยังมีน้อยกว่าการส่งข้อมูลแบบขนานอีกด้วย การสื่อสารแบบอนุกรมสามารถแบ่งออกเป็น 3 รูปแบบดังนี้

- Simplex สามารถส่งข้อมูลได้อย่างเดียว เป็นการสื่อสารทางเดียว
- Half-Duplex สามารถส่งข้อมูลไปยังปลายทาง และสามารถรับข้อมูลจากปลายทางได้ แต่ไม่สามารถทำการส่งและรับข้อมูลในเวลาเดียวกัน
- Full-Duplex สามารถรับ-ส่งข้อมูลได้ในเวลาเดียวกัน

นอกจากนี้ยังสามารถแบ่งประเภทของการสื่อสารแบบอนุกรมตามลักษณะสัญญาณในการส่งได้อีก 2 แบบคือ

- การสื่อสารแบบซิงโครนัส (Synchronous) การสื่อสารแบบซิงโครนัสนี้จะใช้สัญญาณนาฬิกาควบคุมการรับส่งสัญญาณ เช่น สายคีย์บอร์ดคอมพิวเตอร์โดยจะมีสายสัญญาณเส้นหนึ่งเป็นสายสัญญาณนาฬิกา ส่วนอีกเส้นหนึ่งเป็นสายข้อมูล



ภาพที่ 2.22 แสดงลักษณะสัญญาณของการสื่อสารแบบซิงโครนัส

- การสื่อสารแบบอะซิงโครนัส (Asynchronous) การสื่อสารแบบอะซิงโครนัสนั้นจะใช้สายข้อมูลเพียงตัวเดียว แต่จะใช้รูปแบบการส่งข้อมูล หรือ Bit Pattern เป็นตัวกำหนดว่าส่วนไหนจะเป็นส่วนเริ่มต้นข้อมูล ส่วนไหนเป็นข้อมูล ส่วนไหนจะเป็นส่วนตรวจสอบความถูกต้องของข้อมูล และส่วนไหนเป็นส่วนปิดท้ายข้อมูล โดยต้องกำหนดให้สัญญาณนาฬิกาเท่ากันทั้งภาคส่ง และภาครับ

มาตรฐานของการส่งข้อมูลแบบอนุกรมอีกแบบที่ได้รับความนิยมอย่างสูงตั้งแต่อดีตถึงปัจจุบันโดยใช้งานกันอย่างแพร่หลายทั้งการสื่อสาร และการควบคุมทางอุตสาหกรรมนั่นคือ มาตรฐาน RS-232

2.3.2 มาตรฐาน RS-232C

มาตรฐาน RS-232C เป็นมาตรฐานที่ได้รับการออกแบบมาเพื่อที่จะทำให้อุปกรณ์ต่อพ่วงจากผู้ผลิตต่างกันสามารถทำงานร่วมกันได้ มาตรฐานหลายชนิดได้รับการออกแบบขึ้นมา แต่มาตรฐานที่ได้รับความนิยม และใช้กันอย่างกว้างขวางมากที่สุดคือ มาตรฐาน RS-232C ซึ่งถูกประกาศใช้ในปี 1969 โดยสมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ในยุคแรกๆ การอินเตอร์เฟสแบบ RS-232C ถูกออกแบบสำหรับเชื่อมต่อเทอร์มินอล (DTE : Data Terminal Equipment) กับโมเด็ม (DEC : Data Communication Equipment) ทั้งนี้ก็เพื่อป้องกันไม่ให้เกิดการส่งข้อมูลบนสายเส้นเดียวกัน

มาตรฐาน RS-232C ได้แบ่งอุปกรณ์ออกเป็น 2 ประเภท ซึ่งอุปกรณ์ทั้งสองประเภทนี้ก็คือ

1. อุปกรณ์ DTE (Data Terminal Equipment) เป็นอุปกรณ์สำหรับส่งข้อมูล (เอาต์พุต)
2. อุปกรณ์ DEC (DEC : Data Communication Equipment) เป็นอุปกรณ์สำหรับรับข้อมูล (อินพุต)

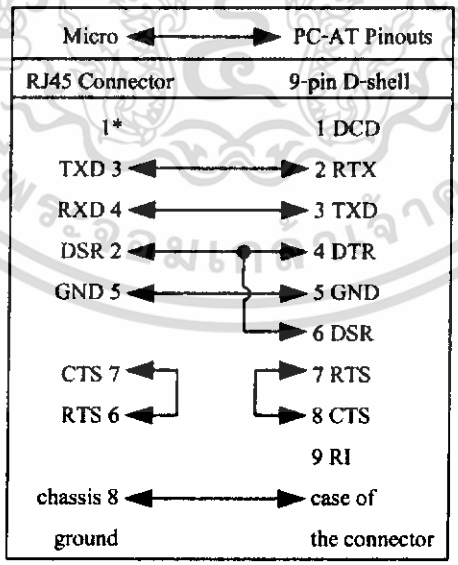
ตามมาตรฐาน RS-232C แล้วคอนเน็กเตอร์ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ DCE จะเป็นตัวเมีย ซึ่งคอนเน็กเตอร์ที่นิยมใช้กันอยู่จะเป็นชนิด D-Type แบบ 9 ขา และแบบ 25 ขา โดยจะติดตั้งอยู่หลังคอมพิวเตอร์ ระดับแรงดันจะมีค่าระหว่าง +3 V ถึง +15V

สำหรับลอจิก High และลอจิก Low จะมีระดับแรงดันระหว่าง +3 V ถึง +15V สามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุด 50 ฟุต หรือ 150 เมตร แต่ถ้าต้องการสื่อสารกับอุปกรณ์อื่นที่อยู่ห่างกันมากๆ เราจำเป็นต้องใช้อุปกรณ์อื่นๆ เข้าช่วย เช่น การใช้โมเด็ม เป็นต้น

2.3.3 รายละเอียดของสายสัญญาณ

สำหรับรายละเอียดของสายสัญญาณนั้นประกอบไปด้วย

- Transmit Data : TD ใช้สำหรับส่งข้อมูลอนุกรมออกจากคอมพิวเตอร์
- Receive Data : RD ใช้สำหรับรับข้อมูลอนุกรมเข้ามายังคอมพิวเตอร์
- Request To Send : RST ใช้สำหรับส่งข้อมูลไปยังอุปกรณ์ปลายทาง เพื่อร้องขอให้อุปกรณ์ปลายทางส่งข้อมูลกลับมา
- Clear To Send : CTS ใช้สำหรับตรวจสอบว่าอุปกรณ์ที่เชื่อมต่อด้วยพร้อมที่จะรับข้อมูลหรือไม่ โดยจะคอยรับสัญญาณ RTS เมื่อทุกอย่างพร้อมก็จะทำส่งข้อมูลออกทางขา TD
- Data Set Ready : DSR ใช้สำหรับตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง จะใช้คู่กับขา DTR
- Signal Ground : SG เป็นกราวด์ของระบบ
- Carrier Detect :CD ขานี้จะ Active เมื่อมีการส่งสัญญาณ Carrier จากโมเด็ม
- Data Terminal Ready : DTR ใช้สำหรับบอกให้ อุปกรณ์ปลายทางรับรู้ว่าต้องการติดต่อด้วยโดยขา DTR นี้ต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง
- Ring Indicator : RI ขานี้จะ Active เมื่อโมเด็มได้รับสัญญาณเรียกเข้าจากสายโทรศัพท์



ภาพที่ 2.23 การเชื่อมต่อคอนเน็กเตอร์

2. 4 เครื่องควบคุมแบบโปรแกรมได้

PLC เป็นอุปกรณ์โซลิดสเตจ (Solid State) ที่ทำงานแบบลอจิก (Logic Function) การออกแบบการทำงานของ PLC ใช้สำหรับควบคุมกระบวนการทำงานของเครื่องจักรและอุปกรณ์

การใช้ (PLC) สำหรับควบคุมเครื่องจักรหรืออุปกรณ์ต่างๆ ในโรงงานอุตสาหกรรมจะมีข้อได้เปรียบกว่าการใช้ระบบของรีเลย์ (Relay) ซึ่งจำเป็นจะต้องเดินสายไฟ ฉะนั้นเมื่อมีความจำเป็นจะต้องเปลี่ยนกระบวนการผลิตหรือลำดับการทำงานใหม่ ก็จะต้องเดินสายใหม่ ซึ่งเสียเวลาและค่าใช้จ่ายสูงแต่เมื่อเปลี่ยนมาใช้ PLC แล้วการเปลี่ยนกระบวนการผลิตก็ทำได้ง่ายโดยการเปลี่ยนโปรแกรมใหม่เท่านั้น นอกจากนี้แล้ว PLC ยังใช้ระบบโซลิด-สเตจ ซึ่งน่าเชื่อถือกว่าระบบเดิมกินกระแสไฟน้อยกว่า และสะดวกกว่าเมื่อต้องขยายการทำงาน

จากหลังการพื้นฐานแล้วอุปกรณ์ควบคุมตัวนี้ทำงานในลักษณะเลขฐานสอง คือ "เปิด" - "ปิด" หรือ "ON" - "OFF" หรือสัญญาณลอจิกเท่านั้น แต่ปัจจุบันนี้ไม่ได้เป็นเช่นนี้อีกต่อไปแล้ว คือสามารถรับและส่ง Input แบบต่อเนื่อง หรือสัญญาณอนาล็อก (Analog) ซึ่งในปริญญานิพนธ์ฉบับนี้ได้นำ PLC มาใช้ในการควบคุมการทำงานของกระบวนการ โดยการสั่งการและดูการสั่งการและดูสถานการณ์ทำงานจากคอมพิวเตอร์ ซึ่งเป็นการสื่อสารข้อมูลกันระหว่างเครื่อง PLC กับคอมพิวเตอร์ เป็นผลให้ระบบการควบคุมการทำงานประสิทธิภาพมากขึ้น

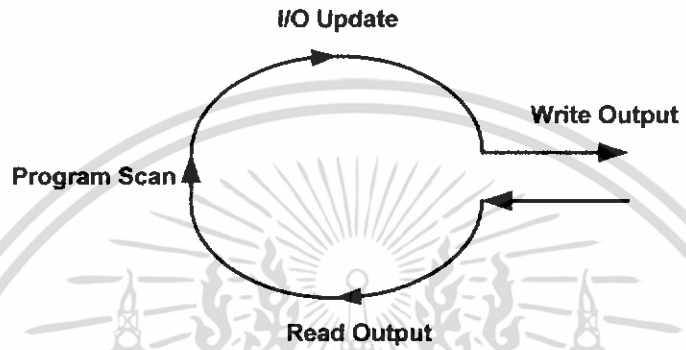
2.4.1 ประวัติความเป็นมาของ PLC

PLC ย่อมาจากคำว่า Programmable Logic Controller เป็นอุปกรณ์อิเล็กทรอนิกส์ที่มีหน่วยความจำ ในการเก็บโปรแกรมสำหรับควบคุมการทำงานของอุปกรณ์ต่างๆ ที่ต่อกับขั้วเข้าและขั้วออกของมัน PLC ยังมีชื่อเรียกอีกอย่างอื่นอีก เช่น PC ย่อมาจาก Programmable Controller และ SC ซึ่งย่อมาจาก Sequence Controller PLC ขนาดเล็กอาจเรียกว่าเป็นตัวซีเควนเซอร์ก็มี

PLC เริ่มพัฒนาขึ้นครั้งแรกในปี ค.ศ. 1968 โดย Hydromantic Division ของบริษัท General Motors Corporation เพื่อใช้แทนการควบคุมแบบเก่าๆ ซึ่งใช้รีเลย์ ติดตั้ง ดัดแปลง และแก้ไขได้ยากลำบาก มาเป็นการควบคุมแบบใหม่ ที่ใช้วงจรอิเล็กทรอนิกส์แทนรีเลย์ และใช้การเขียนโปรแกรมทำนองเดียวกันกับการเขียนโปรแกรมคอมพิวเตอร์ กำหนดเงื่อนไขในการควบคุมแทนการเดินสายเชื่อมต่อไฟฟ้าแบบเก่า เพื่อเพิ่มความสามารถ

2.4.2 โครงสร้างฮาร์ดแวร์ของ PLC

1. หน่วยประมวลผลกลาง (CPU : Central Processing Unit) ส่วนที่ทำหน้าที่ควบคุมการทำงานของ PC ซึ่งจะใช้ไมโครโปรเซสเซอร์ชนิด 8 บิต เป็นตัวประมวลผล ปกติหน้าที่ของ CPU คือรับข้อมูลอินพุตเข้ามาทำการประมวลผล แล้วส่งผลที่ได้ออกไปยังเอาท์พุต จากนั้นก็จะวนไปรับข้อมูลอินพุตเข้ามาอีก แล้วทำซ้ำๆ ลักษณะนี้ไปเรื่อยๆ ซึ่งเรียกว่า Scan การทำงานของ CPU จะอยู่ภายใต้การควบคุมของโปรแกรมที่ผู้ใช้ป้อนเข้าไปดังแสดงดังภาพที่ 2.24



ภาพที่ 2.24 แสดงการประมวลผลของ PLC

2. หน่วยความจำ (Memory Unit) ทำหน้าที่เก็บโปรแกรมและข้อมูล ขนาดของหน่วยความจำจะเป็นตัวกำหนดขีดความสามารถของแบบปกติ มักจะมีขนาดวัดเป็นจำนวน Step โดยมีหน่วยความจำชนิดต่างๆ ดังนี้

- ROM (Read Only Memory) เป็นหน่วยความจำที่ไม่อนุญาตให้ผู้ใช้เปลี่ยนแปลงแก้ไขข้อมูลภายใน แต่สามารถรักษาข้อมูลไว้ได้แม้ว่าจะไม่มีกระแสไฟฟ้า เหมาะสำหรับเก็บโปรแกรมบริหารระบบ หรือโปรแกรมที่เสร็จสมบูรณ์ไม่ต้องการแก้ไข

- RAM (Random Access Memory) เป็นหน่วยความจำที่ผู้ใช้สามารถเปลี่ยนแปลงแก้ไขได้ เหมาะสำหรับเก็บโปรแกรมที่อยู่ในช่วงพัฒนา หรือต้องการเปลี่ยนแปลงบ่อยครั้ง ในการใช้งานจริงๆ แล้วจะต้องมีแหล่งจ่ายไปสำรองต่อไว้เพื่อป้องกันไม่ให้ข้อมูลสูญหายเมื่อเกิดไฟดับ

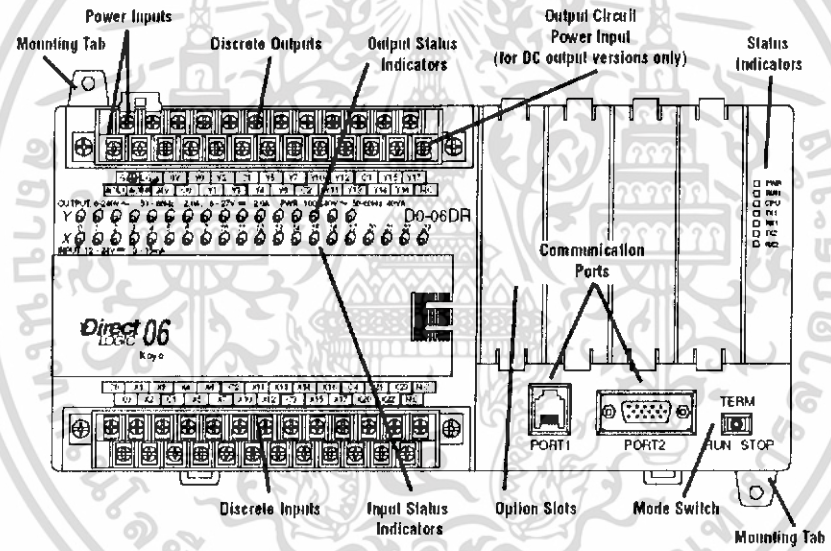
- EPROM (Erasable Access Read Only Memory) เป็นหน่วยความจำสำหรับเก็บโปรแกรมที่มีการพัฒนาจนใช้งานได้ดีให้เป็นการถาวร และมีความสามารถในการอัดโปรแกรม จะทำได้โดยการถ่ายข้อมูล จากหน่วยความจำ RAM ลงมาสู่หน่วยความจำ EPROM โดยอาศัยเครื่องอัดชนิดพิเศษเพื่อใช้ในการต่อรวมกับชุด PC หน่วยความจำชนิดนี้ หน่วยความจำประเภทนี้โปรแกรมจะไม่มีการสูญหาย เมื่อเกิดไฟดับ แต่ถ้ามีความจำเป็นที่จะลบโปแกรมภายใน ก็สามารถทำได้โดยการล้างโปรแกรมภายใน

- EEPROM (Electrically Erasable Access Read Only Memory) มีลักษณะคล้ายกับ EPROM คือหน่วยความจำที่เหมาะสมสำหรับโปรแกรมผู้ใช้ที่เสร็จสมบูรณ์แล้ว การอัดโปรแกรมจะอาศัยเครื่องอัดโปรแกรมชนิดพิเศษเช่นเดียวกัน แต่ตอนลบข้อมูลจะใช้วิธีป้อนสัญญาณพัลส์และการลบข้อมูลเก่าไม่จำเป็นต้องลบทั้งหมด ผู้ใช้สามารถแก้ไขเฉพาะตำแหน่งที่ต้องการ

PLC จะรับสัญญาณเข้ามาทางขั้วเข้า และให้สัญญาณออกทางขั้วออก การให้สัญญาณออกนี้จะเป็นไปตามโปรแกรมที่เก็บไว้ในเครื่อง PLC

2.4.3 ส่วนประกอบด้านหน้าของ PLC(Programmable Logic Controller KOYO DL06)

ประกอบด้วยส่วนของการติดต่อ ส่วนแสดงผล ไฟแสดงสถานะ และสัญลักษณ์คำอธิบายแสดงอยู่บนด้านหน้าของ PLC ขนาดเล็กรุ่น DL06 พอร์ตการติดต่อจะอยู่ด้านหน้าของ PLC ดังภาพที่ 2.25



ภาพที่ 2.25 แสดงส่วนต่างๆด้านหน้าของ PLC KOYO DL06

ส่วนของการติดต่อที่ Terminals ล่างจะเป็นในส่วนของอินพุทซึ่งมีให้เลือกใช้ตั้งแต่ X0 , X1 , X2 , X3 , X4 , X5 , X6 , X7 , X10 , X11 , X12 , X13 , X14 , X15 , X16 , X17 , X20 , X21 , X22 และ X23 โดยที่ X0 ถึง X3 จะใช้คอมมอนร่วมคือ C0 , X4 ถึง X7 จะใช้คอมมอนร่วมคือ C1, X10 ถึง X13 จะใช้คอมมอนร่วมคือ C2, X14 ถึง X17 จะใช้คอมมอนร่วมคือ C3 , X20 ถึง X23 จะใช้คอมมอนร่วม คือ C4

ส่วนของการติดต่อที่ Terminals บนจะเป็นในส่วนของเอาท์พุทและช่องต่อแหล่งจ่ายจากภายนอกซึ่งในส่วนของเอาท์พุทนั้นก็มีตั้งแต่Y0 , Y1 , Y2 , Y3 , Y4 , Y5 , Y6 , Y7 , Y10 , Y11 ,

Y12 , Y13 , Y14 , Y15 , Y16 และ Y17 โดยที่ Y0 ถึง Y3 จะใช้คอมมอนร่วมคือ C0 , Y4 ถึง Y7 จะใช้คอมมอนร่วมคือ C1 , Y10 ถึง Y13 จะใช้คอมมอนร่วมคือ C2 , Y14 ถึง Y17 จะใช้คอมมอนร่วมคือ C3

ในส่วนของ LED สองแถวใต้ Terminals ของเอาท์พุทคือไฟแสดงผลของอินพุทและเอาท์พุทของแต่ละ Terminals ส่วนไฟแสดงผลด้านซ้ายมือสุดของ PLC คือไฟแสดงสถานการณ์ทำงานของPLC ซึ่งมีรายระเอียดดังตารางที่ 2.4

ตารางที่ 2.4 แสดงผลของไฟขณะ PLC ทำงาน

Indicator	Status	Meaning
PWR	ON	Power good
	OFF	Power failure
RUN	ON	CPU is in Run Mode
	OFF	CPU is in Stop or program Mode
	Blinking	CPU is in firmware update mode
CPU	ON	CPU self diagnostics error
	OFF	CPU self diagnostics good
	Blinking	Low battery
TX1	ON	Data is being transmitted by the CPU - Port 1
	OFF	No data is being transmitted by the CPU - Port 1
RX1	ON	Data is being received by the CPU - Port 1
	OFF	No data is being received by the CPU - Port 1
TX2	ON	Data is being transmitted by the CPU - Port 2
	OFF	No data is being transmitted by the CPU - Port 2
RX2	ON	Data is being received by the CPU - Port 2
	OFF	No data is being received by the CPU - Port 2

ส่วน Option slots ทั้ง4ช่องมีไว้สำหรับนำการ์ดAnalog ต่างๆขึ้นอยู่กับการใช้งาน ส่วนด้านล่างของ Option slots จะเป็นพอร์ทที่ใช้ในการสื่อสารของPLC ซึ่งพอร์ทที่1 จะมีลักษณะเป็น RJ12 Phone plug 6-pin และพอร์ทที่2 เป็นแบบ DB15 pin VGA (female)

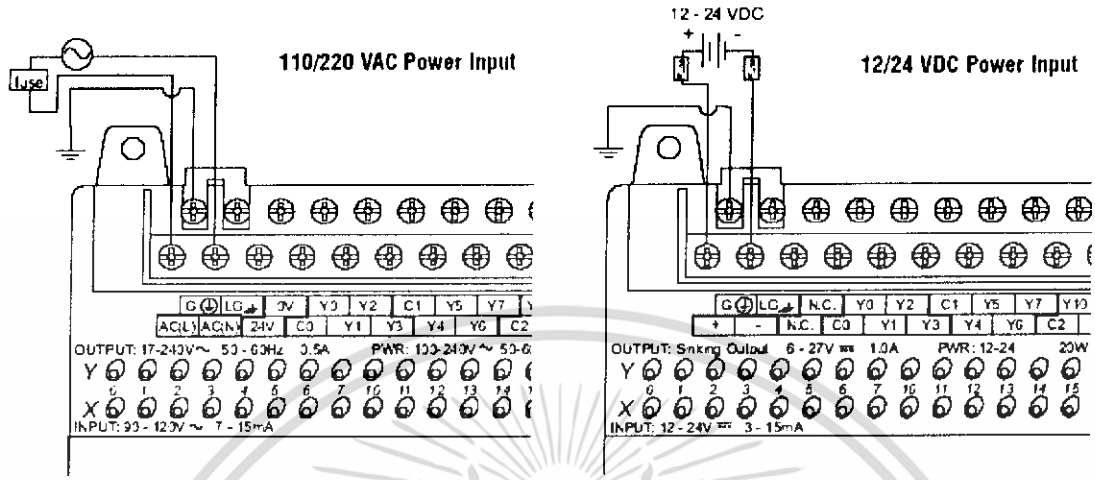
ตังสุดท้ายเป็น Mode switch ซึ่งมีรายระเอียดการทำงานดังตารางที่ 2.5

ตารางที่ 2.5 แสดงการทำงานที่ Mode ต่างๆของสวิตช์

Mode Switch Position	CPU Action
RUN (Run Program)	CPU is forced into the RUN mode if no errors are encountered. No changes are allowed by the attached programming/monitoring device.
TERM (Terminal) RUN	PROGRAM and the TEST modes are available. Mode and program changes are allowed by the programming/monitoring device.
STOP	CPU is forced into the STOP mode. No changes are allowed by the programming/monitoring device.

2.4.4 การต่อกับแหล่งจ่ายไฟฟ้าภายนอก

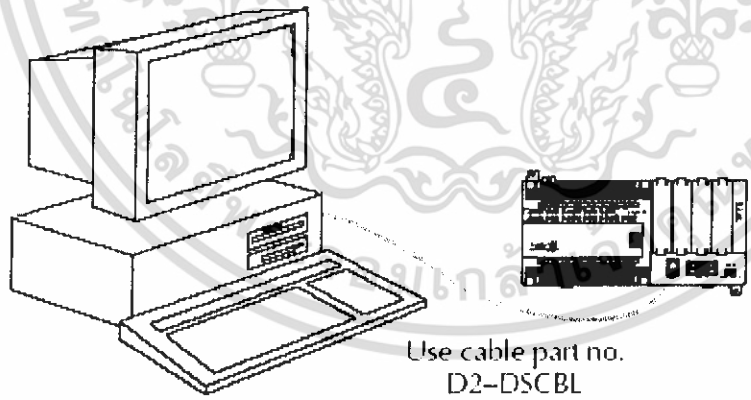
สำหรับ PLC รุ่น KOYO DL06 มีการต่อแหล่งจ่ายไฟฟ้าดังรูปข้างล่าง เมื่อทำการเชื่อมสายเสร็จแล้วให้เอาฉนวนคลุมส่วนของการติดต่อ ห้ามเปิดเครื่องขณะกำลังทำขั้นตอนดังกล่าวอยู่



ภาพที่ 2.26 แสดงการต่อกับแหล่งจ่ายไฟฟ้าภายนอก

2.4.5 การเชื่อมต่อกับเครื่องคอมพิวเตอร์ส่วนบุคคล

PLC KOYO DL06 สามารถทำการโปรแกรมได้จาก โปรแกรม Direct SOFT บนเครื่องคอมพิวเตอร์โดยทำการต่อสายเคเบิลดังภาพที่ 2.27

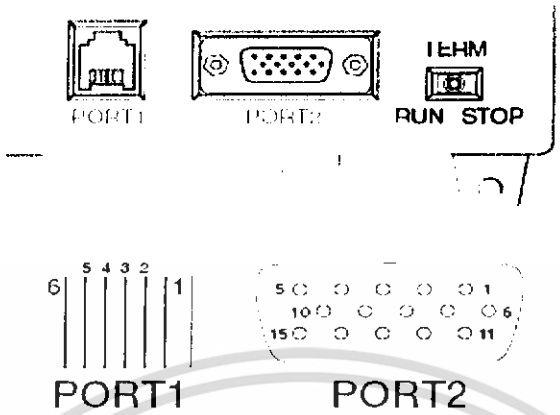


ภาพที่ 2.27 การต่อสายเคเบิลระหว่าง คอมพิวเตอร์ส่วนบุคคลกับ PLC

2.4.6 การติดต่อกับพอร์ตภายนอก

การติดต่อกับพอร์ตภายนอก เราจำเป็นต้องใช้สายที่มีความยาวเพียงพอเพื่อที่จะทำการเชื่อมต่อระหว่าง PLC กับคอมพิวเตอร์ส่วนบุคคลได้อย่างง่ายดายและรวดเร็ว โดยใช้สายสัญญาณ

มาตรฐาน RS-232C โดยใช้หัวปลั๊กโทรศัพท์ RJ-12 ในการทำหัวแจ๊คคู่กับพอร์ตอนุกรม ดังภาพที่ 2.28



Port 1 Pin Descriptions

1	0V	Power (-) connection (GND)
2	5V	Power (+) connection
3	RXD	Receive data (RS-232C)
4	TXD	Transmit data (RS-232C)
5	5V	Power (+) connection
6	0V	Power (-) connection (GND)

Port 2 Pin Descriptions

1	5V	Power (+) connection
2	TXD	Transmit data (RS-232C)
3	RXD	Receive data (RS-232C)
4	RTS	Ready to send
5	CTS	Clear to send
6	RXD-	Receive data (-) (RS-422/485)
7	0V	Power (-) connection (GND)
8	0V	Power (-) connection (GND)
9	TXD+	Transmit data (+) (RS-422/485)
10	TXD-	Transmit data (-) (RS-422/485)
11	RTS+	Ready to send (+) (RS-422/485)
12	RTS-	Ready to send (-) (RS-422/485)
13	RXD+	Receive data (+) (RS-422/485)
14	CTS+	Clear to send (+) (RS-422/485)
15	CTS-	Clear to send (-) (RS-422/485)

Communications Port 1

Port 1 Connects to HPP, DirectSOFT32, operator interfaces, etc.

6-pin, RS232C

Communication speed (baud): 9600 (fixed)

Parity: odd (fixed)

Station Address: 1 (fixed)

8 data bits

1 start, 1 stop bit

Asynchronous, half-duplex, DTE

Protocol (auto-select): K-sequence (slave only), DirectNET (slave only), MODBUS (slave only)

Communications Port 2

Port 2 Connects to HPP, DirectSOFT32, operator interfaces, etc.

15-pin, multifunction port, RS232C, RS422, RS485

Communication speed (baud): 300, 600, 1200, 2400, 4800, 9600, 19200, 38400

Parity: odd (default), even, none

Station Address: 1 (default)

8 data bits

1 start, 1 stop bit

Asynchronous, half-duplex, DTE

Protocol (auto-select): K-sequence (slave only), DirectNET (master/slave), MODBUS (master/slave), non-sequence/print/ASCII in/out

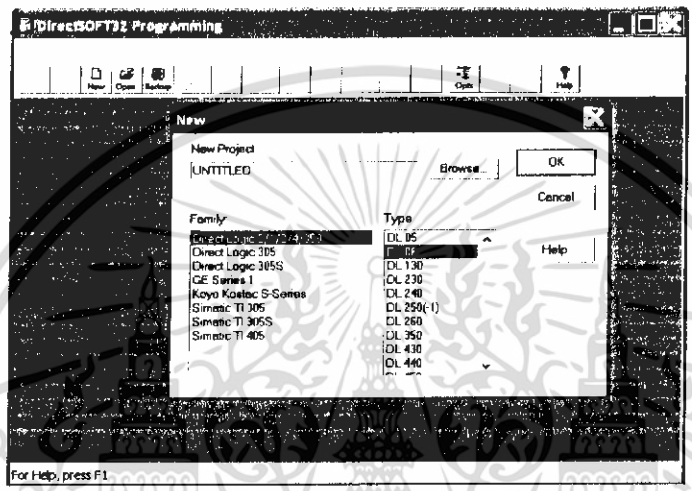
ภาพที่ 2.28 แสดงการต่อสายเข้ากับพอร์ตเพื่อใช้ในการติดต่อสื่อสาร

2.5 โปรแกรม Direct SOFT32

ในการเขียนโปรแกรมลงใน PLC KOYO รุ่น DL06 เราใช้โปรแกรม Direct SOFT32 ในการเขียน Ladder ซึ่งเป็นโปรแกรมการทำงานของ PLC แล้วทำการ Memory ลงใน CPU ของ PLC

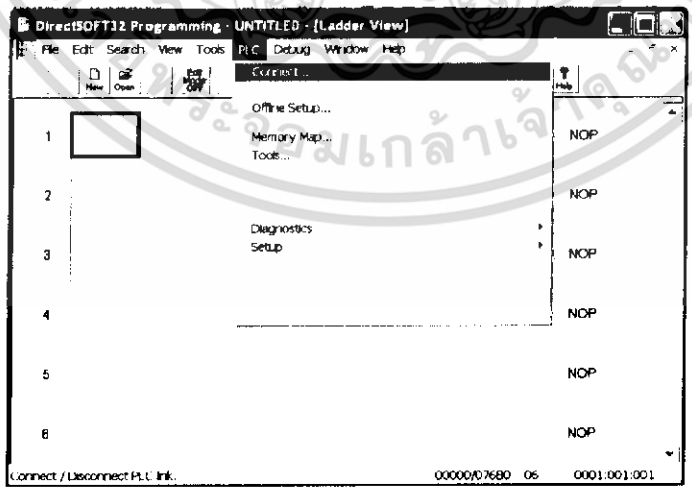
การใช้งานโปรแกรม DirectSOFT32

ขั้นตอนที่ 1 เมื่อทำการคลิกที่โปรแกรม DirectDOFT32 ก็จะปรากฏหน้าต่างดังภาพที่ 2.29

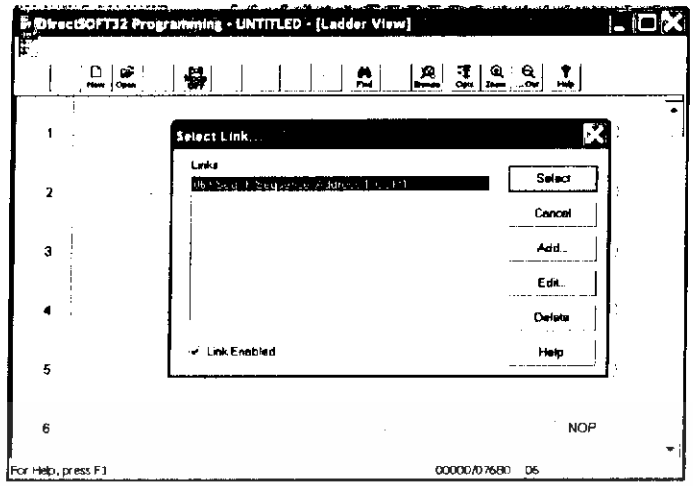


ภาพที่ 2.29 แสดงการสร้าง Project ใหม่ขึ้นมาเพื่อทำการเขียนโปรแกรม

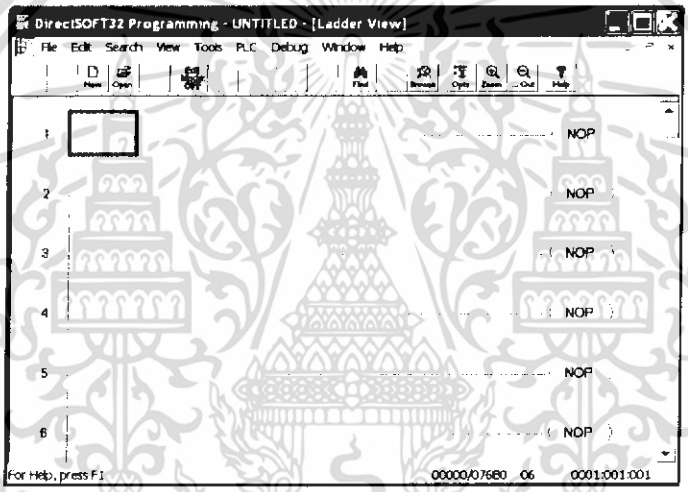
ขั้นตอนที่ 2 ทำการต่อ PLC เข้ากับ Computer จากนั้นทำการ Link กับโปรแกรม DirectSOFT32 โดยเลื่อนเมาท์ไปที่ menu bar>PLC>connect ก็จะปรากฏหน้าต่างดังภาพที่ 2.30



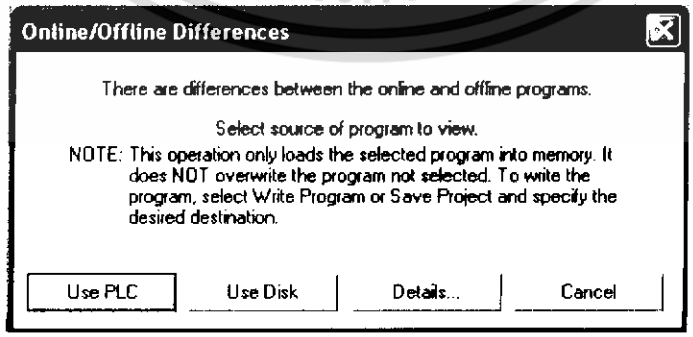
ภาพที่ 2.30 แสดงการเลือกชนิดของ PLC และรุ่นของ PLC หรือชนิดของ CPU แล้วคลิก OK



ภาพที่ 2.31 แสดงการต่อ PLC กับ computer และทำการ Link กับโปรแกรม DirectSOLFT32



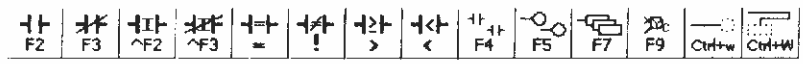
ภาพที่ 2.32 แสดงหน้าจอที่ไว้สำหรับเขียนโปรแกรมให้กับ PLC



ภาพที่ 2.33 แสดงการเรียกใช้ข้อมูลจากฐานข้อมูลต่างๆ

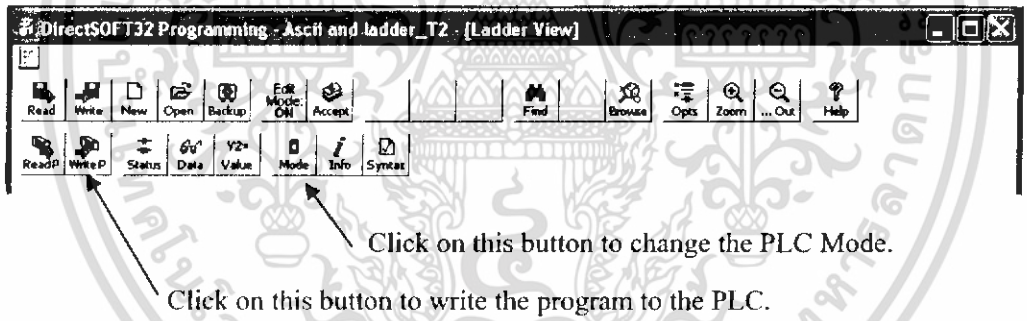
จากหน้าต่างข้างบนถ้าเราเลือก Use PLC โปรแกรมก็จะทำการเรียกข้อมูลที่อยู่ใน CPU ของ PLC ออกมาเพื่อทำการ RUN โปรแกรม แต่ถ้าเราเลือก Use Disk โปรแกรมก็จะทำการเตรียมพร้อมที่จะบันทึกโปรแกรมใหม่ลงใน CPU ของ PLC

ขั้นตอนที่3 การเขียนโปรแกรมทำได้โดยการคลิกที่ Edit Mode Off (ปุ่มจะมีสีเหลือง) จะปรากฏบล็อกสัญลักษณ์ในการเขียน Ladder ดังภาพที่ 2.34

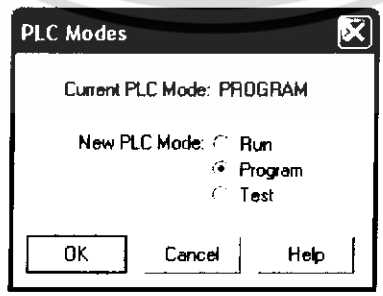


ภาพที่ 2.34 แสดงบล็อกสัญลักษณ์การเขียน Ladder

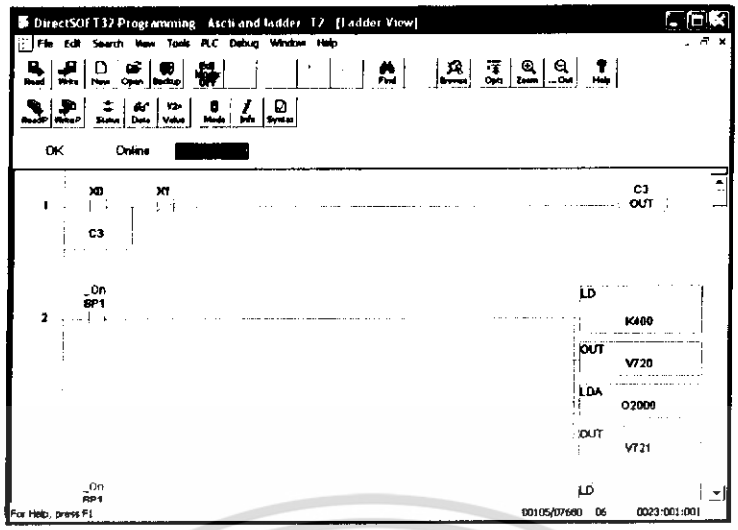
ขั้นตอนที่ 4 การเขียนโปรแกรมใน PLC เมื่อเครื่อง PC และ PLC ทำการติดต่อกันได้แล้ว ก็สามารถที่จะเขียนโปรแกรมลงบน PLC ได้ โดยผ่านโปรแกรม DirectSOFT32 ซึ่งทำได้โดยการคลิกที่ไอคอน ที่สองข้างล่างซ้ายซึ่งเป็นไอคอน Write โปรแกรมก็จะถูกเขียนลงใน PLC ดังภาพที่ 2.35



ภาพที่ 2.35 แสดงไอคอนการติดต่อกับ PLC



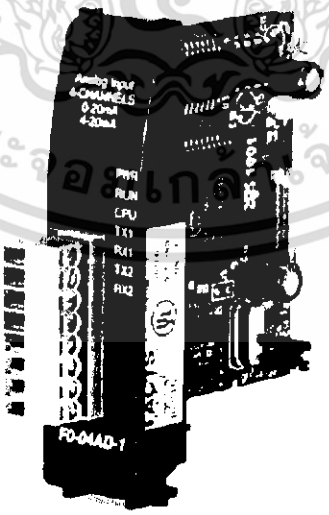
ภาพที่ 2.36 แสดงการเลือก Mode การทำงานของ PLC



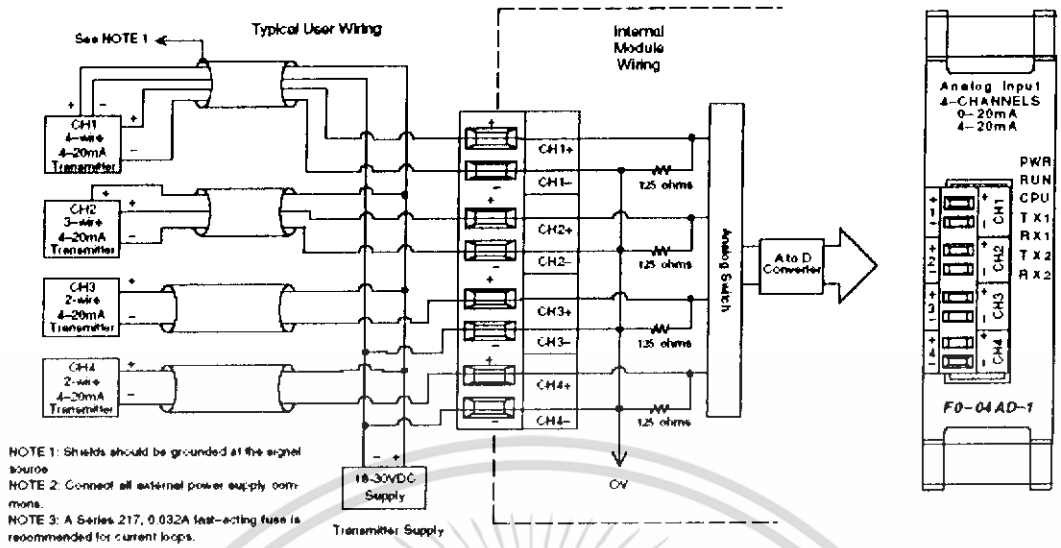
ภาพที่ 2.37 แสดงการทำงานของโปรแกรมที่อยู่ใน Mode RUN

2.6 Analog Input Module

Analog Input Module เป็นอุปกรณ์ที่ใช้รับสัญญาณอนาล็อกจากภายนอกแล้วทำการแปลงมาเป็นสัญญาณ Digital เพื่อให้ CPU ของเครื่อง PLC สามารถทำการประมวลผลได้ ดังนั้น Analog Input Module จึงมีความจำเป็นเมื่อต้องการที่จะควบคุมขบวนการที่มีการส่งสัญญาณควบคุมออกมาเป็นสัญญาณอนาล็อก เราสามารถเลือกใช้ Analog Input Module ได้ตาม ความเหมาะสมกับความต้องการได้ เนื่องจากในท้องตลาดจะมีให้เลือกจำนวน channel ได้เช่น 2channel, 4channel หรือจะมีทั้ง Input และ Output ใน Module ตัวเดียวกันได้ ซึ่งในโครงงานนี้จะเลือกใช้ Analog Input Module แบบ 4channel ดังภาพที่ 2.38



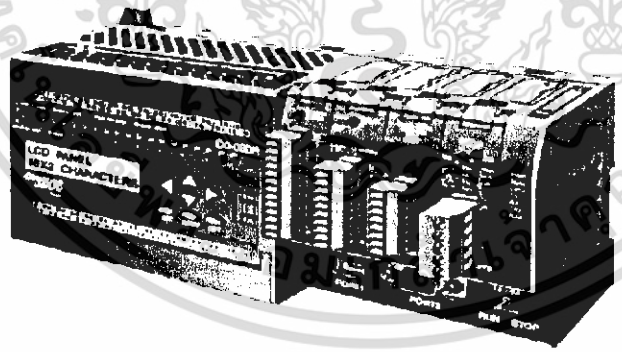
ภาพที่ 2.38 แสดง Analog Input Module แบบ 4channel



ภาพที่ 2.39 แสดงโครงสร้างภายในของ Analog Input Module แบบ 4channel

2.6.1 การติดตั้งการ์ด Analog กับ PLC KOYO รุ่น DL06

ในตัวเครื่อง PLC รุ่น DL06 ของ KOYO จะมีช่องสำหรับที่จะใช้ต่อกับ Module พิเศษอยู่แล้ว 1 ช่อง การที่จะติดตั้งจึงสามารถที่จะทำได้ง่ายมาก เพียงแค่ทำการเสียบลงในช่องที่มีอยู่แล้วในตัวเครื่อง PLC ก็จะสามารถทำการใช้งานได้ทันที



ภาพที่ 2.40 แสดงตัวเครื่อง PLC KOYO รุ่น DL06 เมื่อทำการติดตั้งการ์ด Analog

V-Memory ที่เกี่ยวข้องกับ Analog Input Module ก็จะมี Data Type เลขของ Channel อยู่ที่ V 700, V710, V720 และ V730 ขึ้นอยู่กับช่อง Slotที่ใช้ ส่วน Storage Pointer อยู่ที่ V701, V711, V721 และ V731 ขึ้นอยู่กับช่อง Slotที่ใช้ ซึ่งเห็นได้จากตารางที่ 2.6

ตารางที่ 2.6 แสดงค่า V-Memory ที่เกี่ยวข้องกับ Analog Input Module

Analog Input Module DL06 V-memory Locations				
Slot No.	1	2	3	4
Data Type and Number of Channels	V700	V710	V720	V730
Storage Pointer	V701	V711	V721	V731

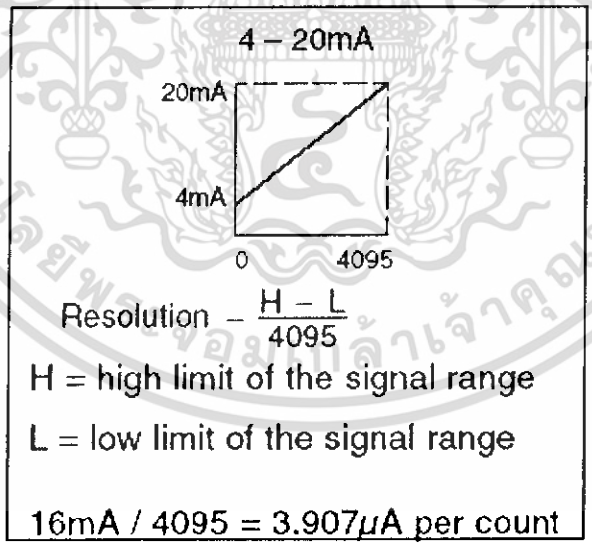
Data Type / Number of channels มีไว้สำหรับเลือกชนิดของข้อมูลที่เป็นค่ามาจากภายนอก และเลือกค่าว่าจะใช้ Channels ไหน เช่น เราเลือกใช้ Slot 3

V720 = 0400 (BCD) จะเป็น Analog Channel ที่ 4 และ ข้อมูลเป็นแบบ BCD

Storage Pointer (V721) ใช้สำหรับการ Set ให้นำข้อมูลไปเก็บไว้ที่ V-Memory ตำแหน่งต่างๆ

2.6.2 การสร้าง Scale ค่าใช้งาน

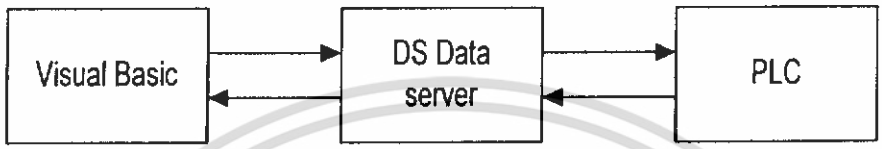
ความไวของการ Scale ค่าจะมีสูตรการคำนวณหาดังภาพข้างล่าง โดยค่าความไวจะมีผลต่อค่าประสิทธิภาพในการนำกลับมาแสดงผล ถ้าค่ามีความละเอียดต่อหน่วยมาก ก็จะทำให้กราฟฟิกแสดงผลมีความละเอียดมากขึ้นด้วย



ภาพที่ 2.41 แสดงสูตรการคำนวณสร้าง Scale ค่าใช้งาน

2.7 โปรแกรม DS Data server

โปรแกรม DS Data server เป็นโปรแกรมที่ช่วยในการที่จะให้ Software แลกเปลี่ยนข้อมูลกับ PLC ซึ่งเปรียบเสมือนว่าโปรแกรม DS Data server จะเป็นล่ามที่จะช่วยในการแลกเปลี่ยนข้อมูลจาก Software window เพื่อที่จะให้เครื่อง PLC เข้าใจ และทำงานได้ตามต้องการ หรือทำการแปลงข้อมูลที่ได้จาก PLC เพื่อให้ Software window เข้าใจ และสามารถนำไปแสดงผล การใช้โปรแกรม DS Data server จะใช้ได้เฉพาะเครื่อง PLC ของยี่ห้อ KOYO เท่านั้น

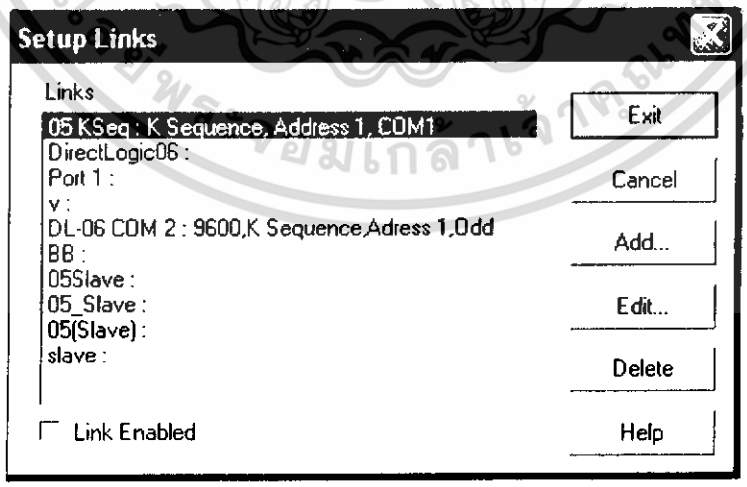


ภาพที่ 2.42 แสดงการเชื่อมโยงข้อมูลโดยการใช้โปรแกรม DS Data server

การติดต่อสื่อสารกันระหว่างโปรแกรม จะใช้การติดต่อได้ 2 แบบ คือ DDE หรือ OPC ของแต่ละโปรแกรม ซึ่ง DDE หรือ OPC จะเป็น Protocol ที่ใช้ในการสื่อสาร

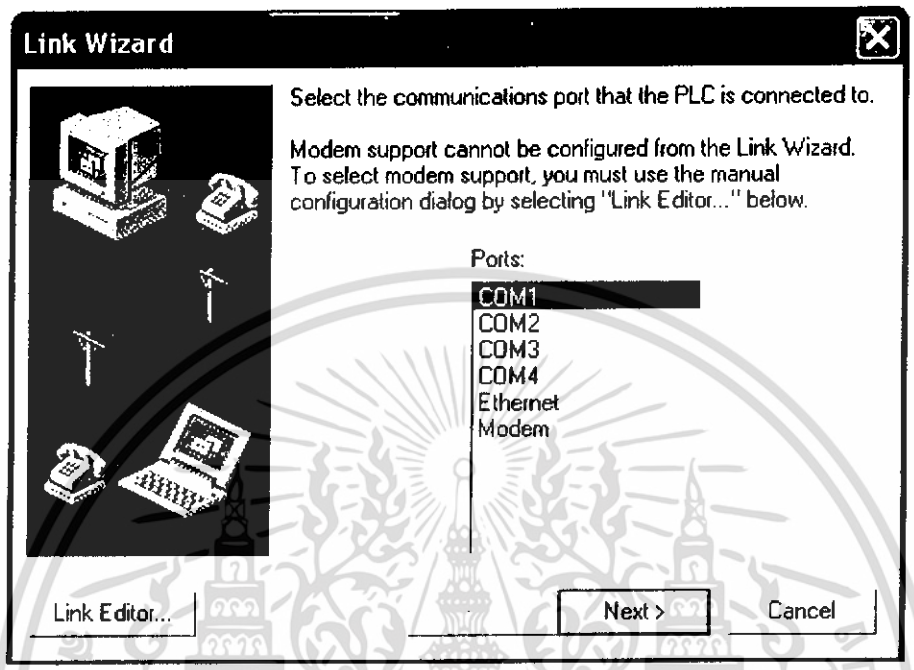
2.7.1 การตั้งค่าการเชื่อมต่อกับ PLC

การที่เราจะทำการใช้ DS Data server ทำงานได้ต้องทำการตั้งค่าเชื่อมต่อเสียก่อน โดยการคลิก จะปรากฏหน้าจอดังภาพที่ 2.43 ถ้าไม่การติดต่อสื่อสารมาก่อนจะปรากฏ เพียงแต่หน้าจอว่าง ถ้ามีการตั้งค่าไว้แล้วให้ทำการกด Select ได้ทันที

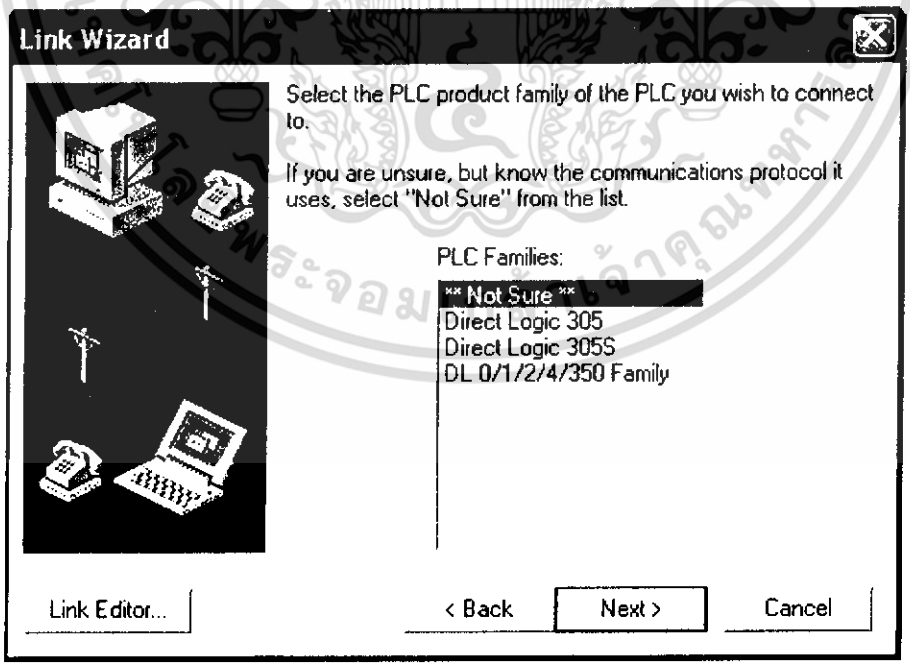


ภาพที่ 2.43 แสดงการเลือกการเชื่อมต่อกับ PLC

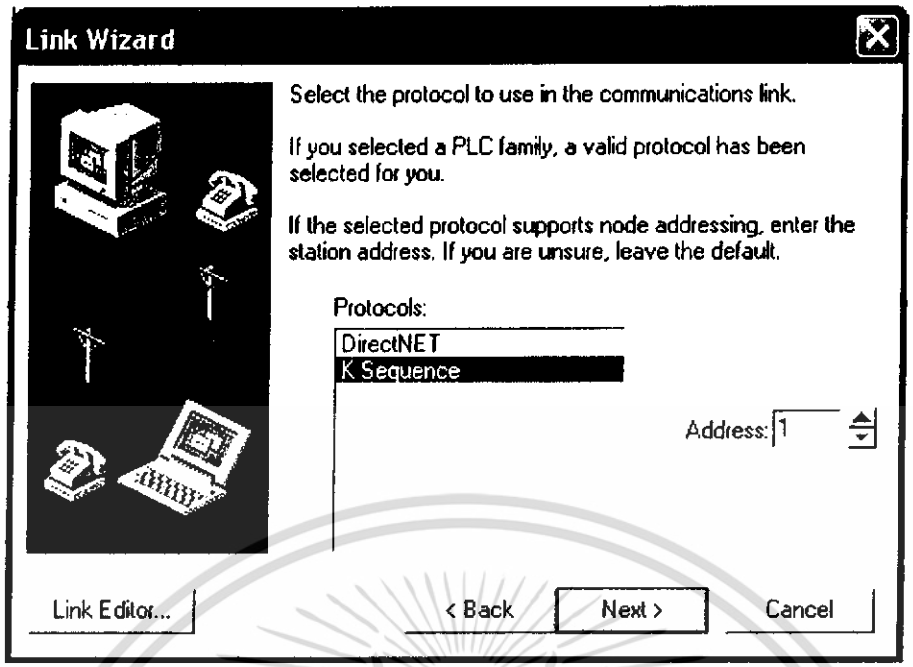
แต่ถ้าไม่มีการตั้งค่าไว้ จะต้องสร้างการเชื่อมต่อใหม่ โดยการคลิก Add เพื่อสร้างการเชื่อมต่อใหม่ จะปรากฏหน้าจอดังภาพที่ 2.44 ก็ให้ทำตามขั้นตอนไปเรื่อยๆ โดยสามารถเลือกที่จะติดต่อกับ PLC รุ่นไหน และทำการเลือก Protocol ว่าจะใช้ Protocol อะไรในการเชื่อมต่อ



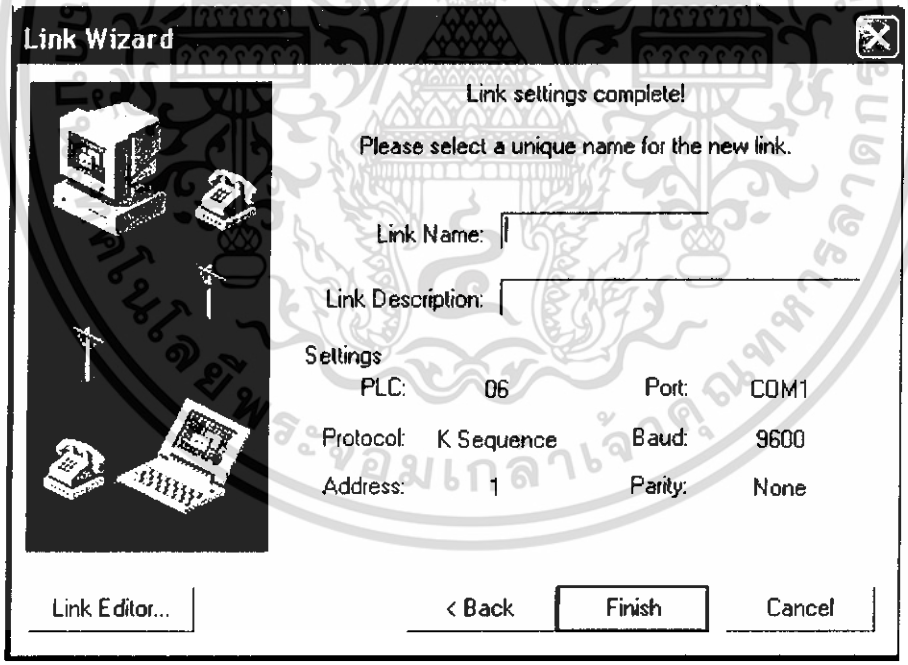
ภาพที่ 2.44 แสดงการเลือกพอร์ตของเครื่องคอมพิวเตอร์



ภาพที่ 2.45 แสดงการเลือกรุ่นต่างๆ ของ PLC



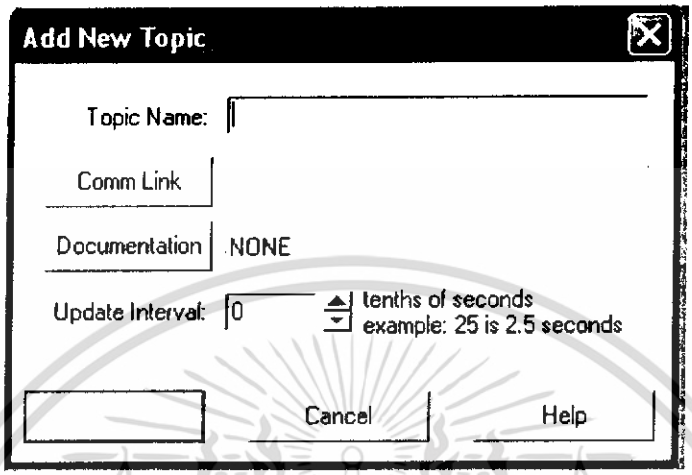
ภาพที่ 2.46 แสดงการเลือกชนิดของ Protocol



ภาพที่ 2.47 แสดงการการอ่านข้อมูลของ PLC

2.7.2 การสร้างหัวข้อในการการติดต่อ PLC

หัวข้อหรือ Topic ที่เราทำการสร้างไว้จะถูกนำไปในโปรแกรม Visual Basic ต่อไป ซึ่งการสร้าง Topic ทำได้โดยการคลิก ก็จะขึ้นหน้าจอดังภาพ 2.48



ภาพที่ 2.48 แสดงการตั้งชื่อหัวข้อโปรแกรมของเครื่อง PLC

GPRS MODULE

GPRS MODULE เป็นอุปกรณ์ที่เรานำมาใช้ทำการส่งข้อความเตือนต่างๆมายังโทรศัพท์เคลื่อนที่ โดย GPRS MODULE จะรับคำสั่งมาจาก PLC ซึ่งจะอยู่ในรูปของรหัสแอสกี ส่วนคำสั่งที่ใช้สั่งงาน GPRS MODULE จะอยู่ในรูปของ AT Commands โดยที่อุปกรณ์ทั้งสองตัวจะติดต่อสื่อสารกันผ่านทางพอร์ตอนุกรรมซึ่งต่อออกมาจากตัวบอร์ดที่เชื่อมต่อกับ GPRS MODULE ซึ่งขาที่ต่อใช้งานสามารถดูได้จากตารางข้างล่าง

ตารางที่ 2.7 แสดงรายละเอียดของขา J1

J1 30-pin Header

Pin#	Name	Pin Type	Pin#	Name	Pin Type
1	EAR_HF-	Analog Output	16	STAT_LED	Output
2	EAR_MT-	Analog Output	17	GND	Power
3	EAR_HF-	Analog Output	18	C103/TXD	Input
4	EAR_MT-	Analog Output	19	C106/CTS	Output
5	GND	Power	20	C125/RING	Output
6	MIC_HF-	Analog Input	21	C107/DSR	Output
7	MIC_MT-	Analog Input	22	C109/DCD	Output
8	MIC_HF-	Analog Input	23	C104/RXD	Output
9	MIC_MT-	Analog Input	24	C108/DTR	Input
10	ON/OFF	Input	25	C105/RTS	Input
11	PWRCTL	Output	26	GND	Power
12	RESET	Input / Output	27	GND	Power
13	GPIO2	Output	28	VBAT	Power
14	AXE	Input	29	GND	Power
15	GPIO1	Input	30	VBAT	Power

ตารางที่ 2.8 แสดงรายละเอียดของขา J3

J3 8-pin Header GPIO's					
Pin#	Name	Pin Type	Pin#	Name	Pin Type
1	GPIO1	Input	5	GPIO5	Input / Output
2	GPIO2	Output	6	GPIO6 / ALARM	Input / Output
3	GPIO3	Input / Output	7	GPIO7 / BUZZER	Input / Output
4	GPIO4	Input / Output	8	GND	Power

ตารางที่ 2.9 แสดงรายละเอียดของขา J5

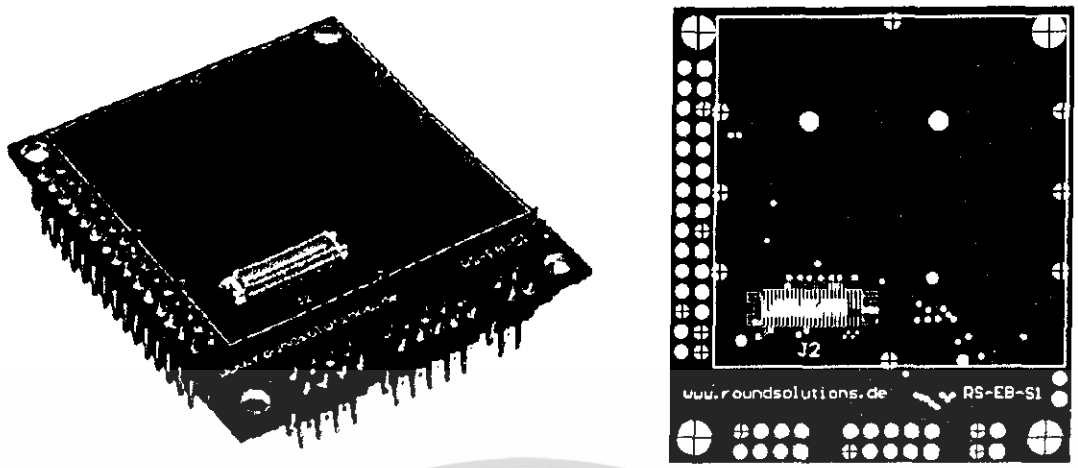
J5 4-pin External Power Connector			
Pin#	Name	Pin#	Name
1	External Power	3	GND
2	External Power	4	GND

ตารางที่ 2.10 แสดงรายละเอียดของขา JP1

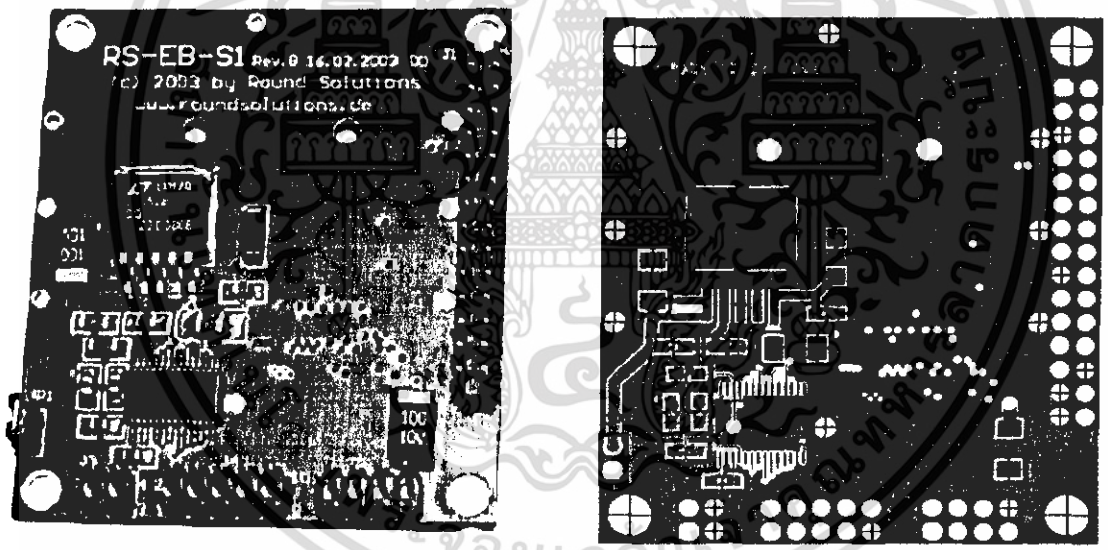
JP1 Power Select Jumper			
Pin#	Name	Pin#	Name
1	VBAT	2	Power Regulator Out

ตารางที่ 2.11 แสดงรายละเอียดของขา J2

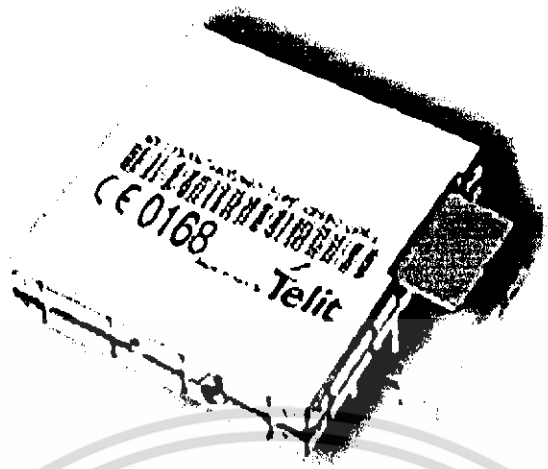
J2 Molex Connector					
Pin#	Name	Pin Type	Pin#	Name	Pin Type
1	VBAT	Power	26	SIMCLK	Output
2	Ground	Power	27	CCIN	Input / Output
3	VBAT	Power	28	GPIO2	Output
4	GND	Power	29	C106/CTS	Output
5	VBAT	Power	30	C125/RING	Output
6	GND	Power	31	GPIO1	Input
7	VBAT	Power	32	CLKSXM	For test purpose
8	GND	Power	33	C107/DSR	Output
9	EAR_PE-	Analog Output	34	TXDD	For test purpose
10	EAR_MT-	Analog Output	35	EMM_TX	Output
11	EAR_PE-	Analog Output	36	C109/DCD	Output
12	EAR_MT+	Analog Output	37	C104/RXD	Output
13	MIC_PE-	Analog Input	38	SCLL	For test purpose
14	MIC_MT+	Analog Input	39	STAT_LED	Output
15	MIC_PE+	Analog Input	40	RXDD	For test purpose
16	MIC_MT-	Analog Input	41	EMM_RX	
17	ON/OFF	Input	42	RFSD	For test purpose
18	AXE	Input	43	C108/DTR	Input
19	C_MIO	Input / Output	44	TFSD	For test purpose
20	C103/TXD	Input	45	C105/RTS	Input
21	PWRCTL	Output	46	GPIO3	Input / Output
22	SIMVCC	-	47	GPIO4	Input / Output
23	RESET	Input	48	GPIO5	Input / Output
24	C_MRST	Output	49	GPIO6 / ALARM	Input / Output
25	TEST32kHz	For test purpose	50	GPIO7 / BUZZER	Input / Output



ภาพที่ 2.49 แสดงด้านบนของตัวบอร์ด RS-EB-S1 และลายวงจร ที่ใช้เชื่อมต่อเข้ากับ GPRS MODULE



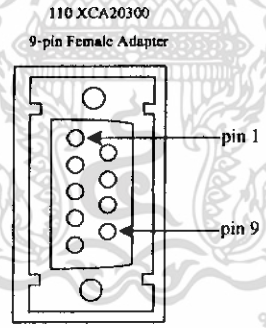
ภาพที่ 2.50 แสดงด้านล่างของตัวบอร์ด RS-EB-S1 และลายวงจร ที่ใช้เชื่อมต่อเข้ากับ GPRS MODULE



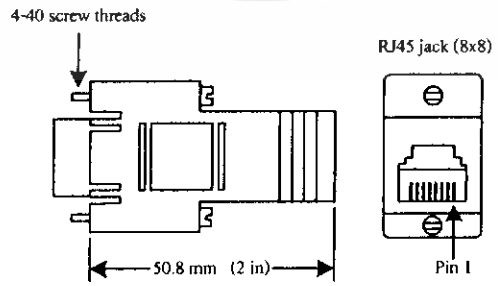
ภาพที่ 2.51 แสดงตัว GPRS MODULE

2.8 การติดต่อสื่อสารระหว่าง PLC กับ GPRS Module

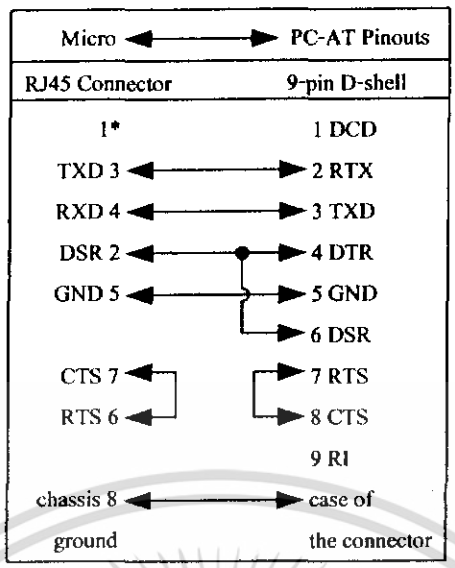
การติดต่อสื่อสารระหว่าง PLC กับ GPRS Module โดยใช้มาตรฐานของการรับส่งข้อมูลแบบอนุกรม (RS-232C)



(ก)



(ข)



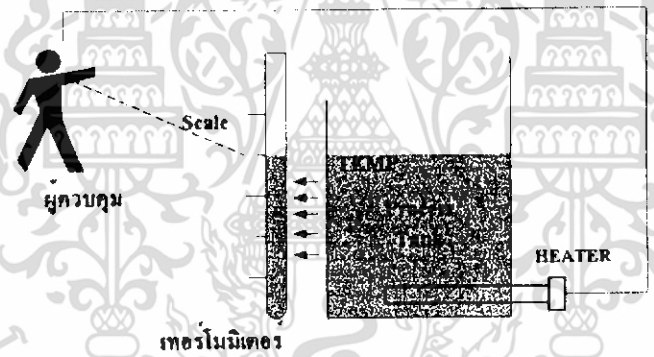
(ค)

ภาพที่ 2.52 แสดงพอร์ตที่ใช้ในการเชื่อมต่อระหว่าง PLC กับ GPRS Module

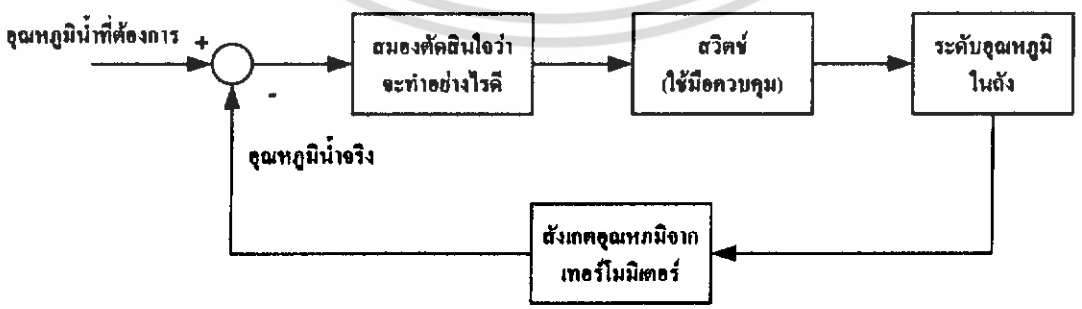
บทที่ 3
หลักการและทฤษฎีที่ใช้ในการออกแบบ

3.1 บทนำ

โดยทั่วไปเป้าหมายของระบบควบคุมกระบวนการต่างๆ ในอุตสาหกรรมนั้น คือการรักษาปริมาณทางฟิสิกส์อันได้แก่ อุณหภูมิ ความดัน อัตราการไหล ระดับ ค่าความเป็นกรดด่าง และอื่นๆ ให้มีค่าใกล้เคียงกับค่าเป้าหมาย หรือค่าที่ต้องการมากที่สุด แม้ว่าสภาวะการทำงานและสภาพแวดล้อมอาจเปลี่ยนแปลงตลอดเวลาก็ตาม ซึ่งการควบคุมที่ดีย่อมเริ่มจากการเลือกแบบการควบคุมที่เหมาะสม ในอดีตการควบคุมกระบวนการต่างๆ จะเลือกใช้การควบคุมแบบง่ายๆ ด้วยมือซึ่งอาศัยพนักงานหรือการควบคุม (Operator) คอยทำหน้าที่เกี่ยวกับการตรวจวัดและปรับแต่งการควบคุมให้ผลตอบสนองเป็นไปตามต้องการ ยกตัวอย่างเช่น การควบคุมระดับน้ำในถัง

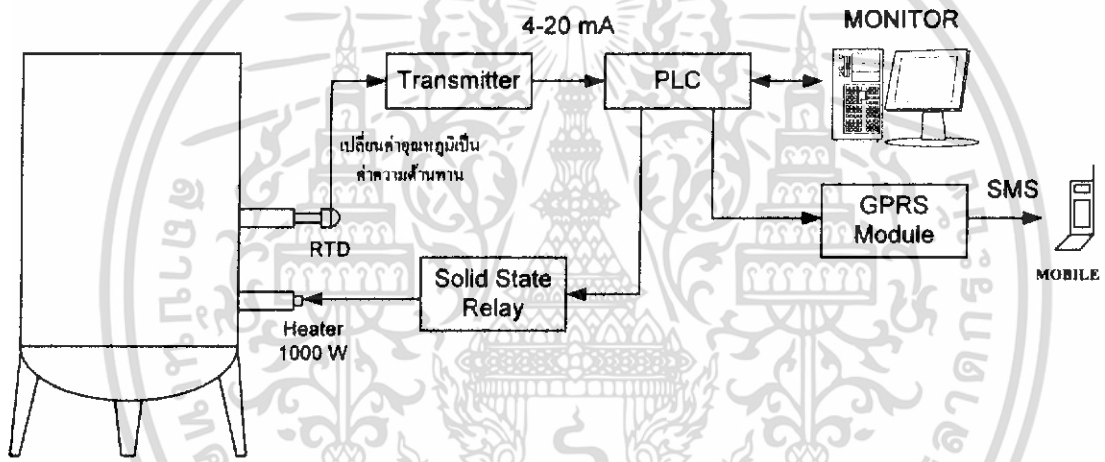


ภาพที่ 3.1 แสดงการควบคุมอุณหภูมิในถังแบบ Manual Control



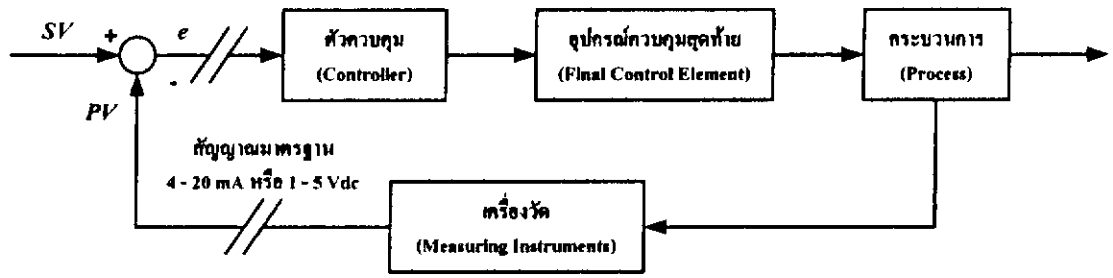
ภาพที่ 3.2 แสดง Block Diagram ของการควบคุมอุณหภูมิในถังแบบ Manual Control

จาก ภาพที่ 3.1 การควบคุมอุณหภูมิในถังจะอาศัยพนักงานคอยทำหน้าที่ในการตรวจวัดอุณหภูมิในถังว่าได้ระดับที่ต้องการหรือไม่ โดยใช้สายตาดูจากเทอร์โมมิเตอร์จากนั้นสมองจะตัดสินใจสั่งการให้มือทำหน้าที่เปิด-ปิดสวิตช์ควบคุม เพื่อให้อุณหภูมิเป็นไปตามต้องการ ซึ่งขั้นตอนของการควบคุมสามารถเขียนเป็นบล็อกไดอะแกรมได้ดัง ภาพที่ 3.2 จะเห็นว่าการควบคุมแบบง่ายๆ ด้วยมือวิธีนี้ จำเป็นต้องอาศัยประสบการณ์และความชำนาญของพนักงาน ดังนั้น การควบคุมจะดีหรือไม่อย่างไรจะขึ้นอยู่กับพนักงานผู้ควบคุมเป็นหลัก ซึ่งในการปฏิบัติงานจริงแล้วมนุษย์หรือพนักงานควบคุมไม่สามารถทำงานให้ดีได้เท่ากันตลอดเวลา จึงทำให้ประสิทธิภาพของการควบคุมลดลง ดังนั้นปัจจุบันโรงงานที่ต้องการการควบคุมที่มีความแม่นยำและประสิทธิภาพสูงจำเป็นต้องนำการควบคุมแบบอัตโนมัติ (Automatic Control) มาใช้งาน ถ้านำมาทำการควบคุมแบบอัตโนมัติ สามารถแสดงการควบคุมได้ดัง ภาพที่ 3.3



ภาพที่ 3.3 แสดงการควบคุมอุณหภูมิในถังด้วยการควบคุมแบบอัตโนมัติ

จากภาพที่ 3.3 ค่าอุณหภูมิในถังของกระบวนการจะถูกวัดโดยเครื่องวัดอุณหภูมิ และส่งสัญญาณในการวัดให้กับเครื่องควบคุม (Controller) ซึ่งเครื่องควบคุมจะนำไปใช้เป็นเกณฑ์ในการตัดสินใจสั่งให้ PLC ควบคุมการเปิด-ปิด เพื่อให้ระดับอุณหภูมิเป็นไปตามต้องการ โดยเครื่องควบคุมจะทำหน้าที่หลักในการคำนวณหาสัญญาณควบคุมที่เหมาะสมตามกฎเกณฑ์การควบคุม ที่พนักงานได้กำหนดไว้ล่วงหน้า สำหรับระบบควบคุมแบบอัตโนมัติที่เราพบเห็นกันในโรงงานอุตสาหกรรม คือ ระบบควบคุมแบบป้อนกลับ (Feedback Control System) ที่มีตัวควบคุมแบบ PID (PID Controller) เป็นเครื่องควบคุม โดยรูปแบบการควบคุมกระบวนการทางอุตสาหกรรมแบบป้อนกลับ



ภาพที่ 3.4 แสดง Block Diagram ของระบบควบคุมแบบป้อนกลับโดยทั่วไป

จากบล็อกไดอะแกรมดัง ภาพที่ 3.4 จะสังเกตเห็นว่าลักษณะคล้ายกับบล็อกไดอะแกรมใน ภาพที่ 3.2 ซึ่งถ้าจะกล่าวไปแล้วแนวความคิดของการควบคุมแบบอัตโนมัตินั้นก็มาจากรูปแบบการควบคุมในลักษณะตาม ภาพที่ 3.3 นั่นเอง โดยระบบการควบคุมทางอุตสาหกรรมแบบป้อนกลับทั่วไปประกอบด้วยอุปกรณ์ 4 ส่วน คือ

3.1.1 ตัวควบคุม (Controller)

เป็นเครื่องมือหรืออุปกรณ์ที่ใช้ในการสร้างสัญญาณควบคุม เพื่อทำหน้าที่ควบคุมให้ระบบหรือกระบวนการที่ต้องการควบคุมมีเอาท์พุท หรือผลตอบสนองเป็นไปตามต้องการ โดยเครื่องควบคุมที่ใช้กันอยู่ในปัจจุบันมีหลายแบบด้วยกัน แต่ที่นิยมใช้มากที่สุดคือ ตัวควบคุมแบบ PID

3.1.2 อุปกรณ์ควบคุมสุดท้าย (Final Control Element)

หมายถึง อุปกรณ์ที่ทำหน้าที่ปรับสภาวะของกระบวนการด้วยการเปลี่ยนแปลงค่าตัวแปรกระบวนการหรือสัญญาณควบคุมที่ได้รับจากตัวควบคุม อุปกรณ์ควบคุมสุดท้ายนั้นมีอยู่หลายอย่างด้วยกัน เช่น วาล์วควบคุม (Control Valve) อินเวอร์เตอร์ (Inverter) และตัวกระทำ (Actuator) เป็นต้น แต่ที่มักพบเห็นกันมากในกระบวนการทางอุตสาหกรรม ได้แก่ วาล์วควบคุม

3.1.3 กระบวนการ (Plant or Process)

หมายถึง ระบบหรือกระบวนการทางฟิสิกส์ที่ต้องการควบคุมให้มีสถานะเป็นไปตามต้องการ เช่น กระบวนการเกี่ยวกับการควบคุมระดับของของเหลว กระบวนการเกี่ยวกับการควบคุมอุณหภูมิ เป็นต้น ซึ่งสถานะของกระบวนการแสดงด้วยตัวแปรกระบวนการ (Process Variable: PV)

3.1.4 อุปกรณ์วัด (Measuring Instruments)

หมายถึง อุปกรณ์ซึ่งอาจจะได้แก่ เซนเซอร์ (Sensor) ทรานสดิวเซอร์ (Transducer) อุปกรณ์แปลง (Transmitter) หรือเครื่องวัดสัญญาณอื่นๆ ในกระบวนการเพื่อนำสัญญาณที่ได้ไปใช้ เป็นตัวแปรในการควบคุม โดยสัญญาณขาออกของอุปกรณ์วัดทั่วไปจะเป็นสัญญาณมาตรฐานทางอุตสาหกรรม เช่น สัญญาณกระแสไฟฟ้า 4-20 มิลลิแอมป์ (4-20 mA) สัญญาณแรงดันไฟฟ้ากระแสตรง 1-5 โวลท์ (1-5 VDC) หรือ สัญญาณลม (3-15 Psi หรือ 0.2-1.0 Kg/cm^2) เป็นต้น

การควบคุมแบบอัตโนมัติจาก ภาพที่ 3.3 มีขั้นตอนการทำงานคือ อุปกรณ์วัดหรือเครื่องวัดจะวัดค่าตัวแปรกระบวนการ (PV) เช่น อุณหภูมิ ความดัน อัตราการไหล และระดับของของเหลว เป็นต้น เพื่อเปรียบเทียบกับค่าอ้างอิงหรือค่าเป้าหมาย (Set Point: SV) จากนั้นตัวควบคุมจะนำค่าความคลาดเคลื่อน (Error: e) ในการควบคุมมาใช้ในการคำนวณเพื่อหาสัญญาณควบคุมที่เหมาะสมที่จะไปควบคุมกระบวนการให้เข้าสู่ค่าเป้าหมายที่ต้องการ นักศึกษาจะเห็นว่าก่อนที่จะทำการควบคุมกระบวนการใดๆ นั้น นักศึกษาจะต้องศึกษาถึงการทำงานของส่วนประกอบต่างๆ ในระบบควบคุมและลักษณะกระบวนการก่อน นั่นก็คือจะต้องศึกษาถึงชนิดและการทำงานของอุปกรณ์วัด ตัวควบคุม และอุปกรณ์ควบคุมสุดท้าย รวมถึงวิธีควบคุมและวิธีการปรับแต่งต่างๆ เสียก่อน เพื่อที่จะได้เลือกใช้อุปกรณ์และรูปแบบของการควบคุมได้อย่างถูกต้องในอันที่จะทำให้การควบคุมมีประสิทธิภาพสูงสุด

3.2 การตรวจวัดอุณหภูมิ

อุณหภูมิเป็นตัวแปรอีกประเภทหนึ่งที่มีความสำคัญมากในอุตสาหกรรมทั่วไปไม่เฉพาะแต่ในอุตสาหกรรมหนักเท่านั้น การเปลี่ยนแปลงของอุณหภูมิมีผลกระทบต่อตัวแปรอื่นๆ ที่อยู่ในกระบวนการทางอุตสาหกรรมด้วยเช่นกัน ไม่ทางตรงก็ทางอ้อม ไม่ว่าจะเป็นความดัน อัตราการไหล ระดับของของเหลว หรือค่าความต้านทาน นั่นหมายความว่าการตรวจวัดปริมาณของตัวแปรใดๆ ที่กล่าวมานี้ภายใต้สภาวะที่มีอุณหภูมิที่เปลี่ยนแปลงตลอดเวลา มักจะมีผลทำให้เกิดความผิดพลาดในการวัดและควบคุมปริมาณของตัวแปรนั้นๆ ดังนั้นจึงจำเป็นอย่างยิ่งที่จะต้องมีการตรวจวัดเพื่อควบคุมอุณหภูมิให้คงที่อยู่เสมอในระหว่างการทำงาน อุปกรณ์ที่ใช้สำหรับตรวจวัด อุณหภูมินั้นมีอยู่มากมายหลายชนิด ขึ้นอยู่กับย่านของอุณหภูมิที่ต้องการวัด ประเภทของตัวกลางที่ต้องการวัดอุณหภูมิและลักษณะการติดตั้งตัวอุปกรณ์วัด

โดยการตรวจวัดอุณหภูมิในปริญญานิพนธ์นี้จะใช้คุณสมบัติการเปลี่ยนแปลงทางไฟฟ้าโดยค่าความต้านทานของอาร์ทีดีจะเปลี่ยนแปลงไปตามการเปลี่ยนแปลงของอุณหภูมิที่จะตรวจวัด หลักการของอาร์ทีดีคือ ความต้านทานไฟฟ้าในเส้นลวดโลหะจะเปลี่ยนค่าไปตามสมการ

$$R_t = R_0(1+\alpha T) \quad \text{หรือ} \quad dR_t/dT = \alpha R_0$$

เมื่อ R_t คือ ค่าความต้านทานของลวดโลหะ ที่อุณหภูมิ t

R_0 คือ ค่าความต้านทานของลวดโลหะ ที่อุณหภูมิ 0

α คือ สัมประสิทธิ์ของการเปลี่ยนแปลงค่าความต้านทานไฟฟ้า ต่ออุณหภูมิ 1 ˚C ($\Omega/\Omega/^{\circ}C$) (Temperature Coefficient of Resistance)

ซึ่งค่า α มีค่าการเปลี่ยนแปลงไปตามชนิดของโลหะ เช่น แพลทินัม 0.00392 $\Omega/\Omega/^{\circ}C$ จากย่านอุณหภูมิ 0 ˚C ถึง 100˚C , นิเกิล 0.0063 $\Omega/\Omega/^{\circ}C$, ทองแดง 0.00425 $\Omega/\Omega/^{\circ}C$ ในทางปฏิบัติ ค่า α ของการเปลี่ยนแปลงอุณหภูมิแต่ละช่วงจะไม่แปรผันเป็นเส้นตรง (Non-linearity) ในห้องปฏิบัติการมาตรฐานที่ต้องการค่าแน่นอน สามารถทำได้โดยการใช้สมการต่อไปนี้

$$R_T = R_0(1 + \alpha T + \beta T^2 + \gamma T^4)$$

ค่า α , β และ γ ได้จากการทดลอง (Empirical Quantity) ซึ่งบริษัทผู้ผลิตเป็นผู้กำหนดมา เช่น แพลทินัม

$$\alpha = 3.985 \times 10^{-3}$$

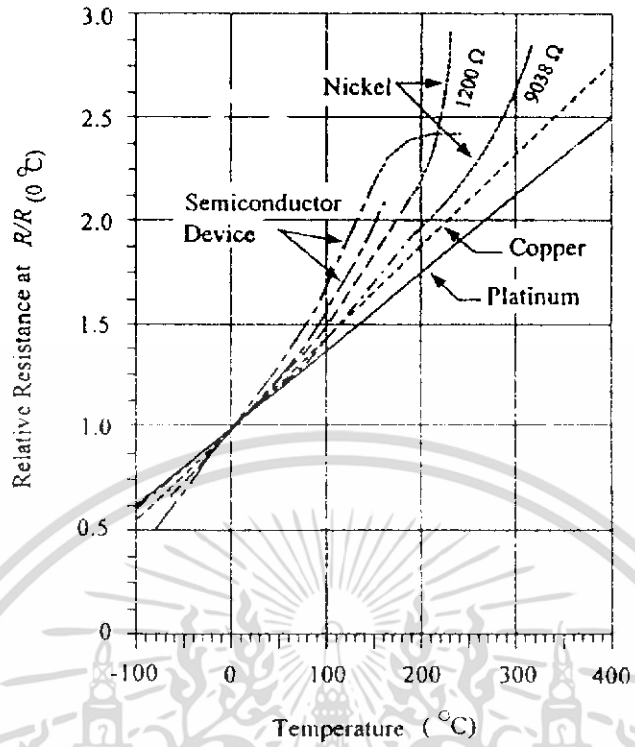
$$\beta = -5.856 \times 10^{-3}$$

$$\gamma = 4.330 \times 10^{-3}$$

สูตรนี้สามารถใช้ได้ทั้งย่านการใช้งานที่ต้องการ Accuracy สูง แต่โดยทั่วไป การคำนวณจะใช้สูตร

$$R_t = R_0(1 + \alpha T)$$

กลุ่มกำหนดมาตรฐาน IEC (IES Standard Groups) ได้กำหนด Calibration Curve ของตัวอาร์ทีดี แบบแพลทินัมใหม่ และเป็นที่ยอมรับของสมาคมอื่น ช่วงอุณหภูมิต่ำกว่า 400 ˚C ค่าความผิดพลาดไม่เกิน $\pm$ 0.2˚C และที่ช่วงอุณหภูมิ 400 ˚C ถึง 600 ˚C ค่าผิดพลาดไม่เกิน $\pm$ 0.2 ˚C กราฟใน ภาพที่ 3.5 แสดงการเปลี่ยนแปลงค่าของอาร์ทีดีที่ทำจากโลหะชนิดต่างๆ



ภาพที่ 3.5 แสดงการเปลี่ยนแปลงค่าของอาร์ทีดี ที่ทำจากโลหะชนิดต่างๆ

แกนในแนวตั้งจะบอกว่าความต้านทานที่อุณหภูมิ T(R) จะมีค่าเป็นกี่เท่าของความต้านทานที่ 0 ˚C ซึ่งอาจเปลี่ยนแปลงได้ถึง 5 เท่าหรือมากกว่า อาร์ทีดีแบบแพลทินัมเป็นที่ใช้กันมากที่สุดในงานอุตสาหกรรม มีค่ามาตรฐาน 100 โอห์ม ที่ 0˚C และเป็นแบบมาตรฐานที่ใช้ในห้องปฏิบัติการทั่วไปหรือเรียกว่าแบบ Standard Platinum Thermomrters (SPRTS) ในช่วงอุณหภูมิสูงถึง 600 ˚C ควรใช้อาร์ทีดีแพลททินัมแบบ 1000 โอห์ม เพราะจะให้เสถียรภาพที่ดีกว่าแบบ 100 โอห์ม

3.3 การตั้ง SPAN และ ZERO ของ TRANSMITTER

ในปริญญานิพนธ์นี้ เป็นการตั้งย่านอุณหภูมิการทำงานของระบบการสอบเทียบให้ทำงานในช่วงอุณหภูมิ 0 – 60 ˚C โดยเปลี่ยนเป็นสัญญาณมาตรฐานทางไฟฟ้า 4 – 20 mA

คำนิยามที่ใช้ในการวัดทางอุตสาหกรรม

คำนิยามที่ใช้ในการวัดทางอุตสาหกรรม คือ คำศัพท์ที่ใช้อธิบายค่าที่มีผลต่อการวัดด้วยวิธีการต่างๆ ที่เกิดขึ้นเนื่องจากโครงสร้างของเครื่องมือวัด เช่น

3.3.1 ย่านการวัด (RANGE)

ย่านการวัด หรือ พิสัยการวัด หมายถึง ขีดความสามารถหรือขีดจำกัดที่เครื่องมือวัดสามารถตรวจวัดได้ แบ่งเป็น 2 ระดับ คือ

ค่าต่ำสุดของย่านวัด (Lower Range Value : LRV) คือ ค่าต่ำสุดที่เครื่องมือวัดนั้นสามารถปรับลงมาอ่านหรือบันทึกค่าได้ ยกตัวอย่าง เช่น ในเครื่องมือวัดทางไฟฟ้าทั่วๆไป จะมีค่าต่ำสุดของย่านวัดเป็นศูนย์ แต่สำหรับเครื่องมือวัดทางอุตสาหกรรม เช่น เครื่องวัดความดัน (Pressure) อาจมีค่าต่ำสุดของย่านวัดเริ่มต้นที่ค่าใดๆ ซึ่งขึ้นอยู่กับขนาดของความดันที่ใช้ในระบบ ถ้าความดันที่ต้องการในระบบนั้นมีค่าสูงมากๆ ค่าต่ำสุดของย่านวัดก็อาจจะเริ่มต้นที่ค่าใดๆ ก็ได้ที่ไม่ใช่ศูนย์

ค่าสูงสุดของย่านวัด (Upper Range Value : URV) คือ ค่าสูงสุดที่เครื่องมือวัดนั้นสามารถปรับขึ้นไปอ่านหรือบันทึกค่าได้ โดยในเครื่องมือวัดตัวหนึ่งๆ อาจมีค่าสูงสุด (URV) หลายค่า (Multirange) ก็ได้

3.3.2 สแปน (Span)

คำว่า "สแปน" หมายถึง ผลต่างของค่าสูงสุดของย่านวัด (URV) กับค่าต่ำสุดของย่านวัด (LRV) ซึ่งในกรณีที่สแปนของเครื่องมือวัดเปลี่ยนแปลงย่านวัดจะต้องเปลี่ยนแปลงตามไปด้วย แต่ในกรณีที่ย่านวัดเปลี่ยนแปลงก็ไม่จำเป็นที่สแปนต้องเปลี่ยนตามถ้าค่าสูงสุดของย่านวัด และค่าต่ำสุดของย่านวัดมีการเปลี่ยนแปลง ในระดับที่เท่ากันทั้งเพิ่มขึ้น และลดลง แต่นอกเหนือจากกรณีดังกล่าวแล้วจะทำให้สแปนเปลี่ยนแปลงตาม ย่านวัดด้วย

$$\text{Span} = \text{URV} - \text{LRV}$$

3.3.3 หลักการคำนวณ

การวัดการเปลี่ยนแปลงของอุณหภูมิจะต้องแปลงสัญญาณให้เป็นสัญญาณมาตรฐานทางไฟฟ้า 4-20 mA เสียก่อน โดยการนำอาร์ทีดีที่ต่อไว้กับ Transmitter มาทำการตั้งค่า Span และ Zero ที่ Transmitter เริ่มต้นวัดอุณหภูมิที่ 0 องศาเซลเซียส โดยใช้น้ำแข็ง วัดอุณหภูมิที่ 0 องศาเซลเซียส โดยปรับ Span ให้ได้กระแส 4 mA จากนั้นนำน้ำผสมกับแอลกอฮอล์ทำให้อุณหภูมิที่จุดเดือดนิ่งอยู่ที่ประมาณ 60 องศาเซลเซียส แล้วทำการปรับ Span ให้ได้ 20 mA แล้วนำสัญญาณที่ได้นี้ไปเป็นสัญญาณควบคุมต่อไป

Rang การทำงานของระบบ คือ

อุณหภูมิที่ใช้งาน คือ 0 °C ถึง 60 °C

กระแสมาตรฐาน คือ 4 mA ถึง 20 mA

จะได้สูตรที่ใช้คำนวณหาค่าอุณหภูมิ

$$T = 3.75 (I - 4)$$

ที่ 4 - 20 mA และ 0 - 60 องศาเซลเซียส

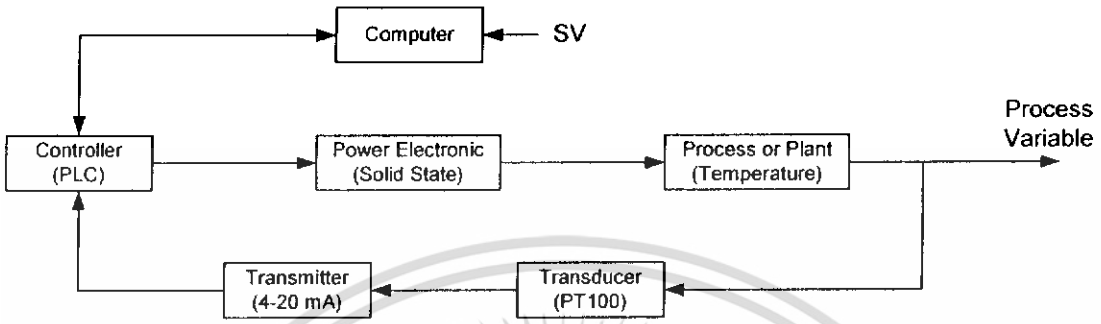
I มีหน่วยเป็น มิลลิแอมป์

T มีหน่วยเป็น องศาเซลเซียส

ตารางที่ 3.1 แสดงค่าการทำ Scaling

%	Temperature	กระแส (mA)
0	0	4.0
10	6	5.6
20	12	7.2
30	18	8.8
40	24	10.4
50	30	12.0
60	36	13.6
70	42	15.2
80	48	16.8
90	56	18.4
100	60	20

บทที่ 4
รายละเอียดในการสร้าง และประกอบปริญญานิพนธ์



ภาพที่ 4.1 แสดงบล็อกไดอะแกรม

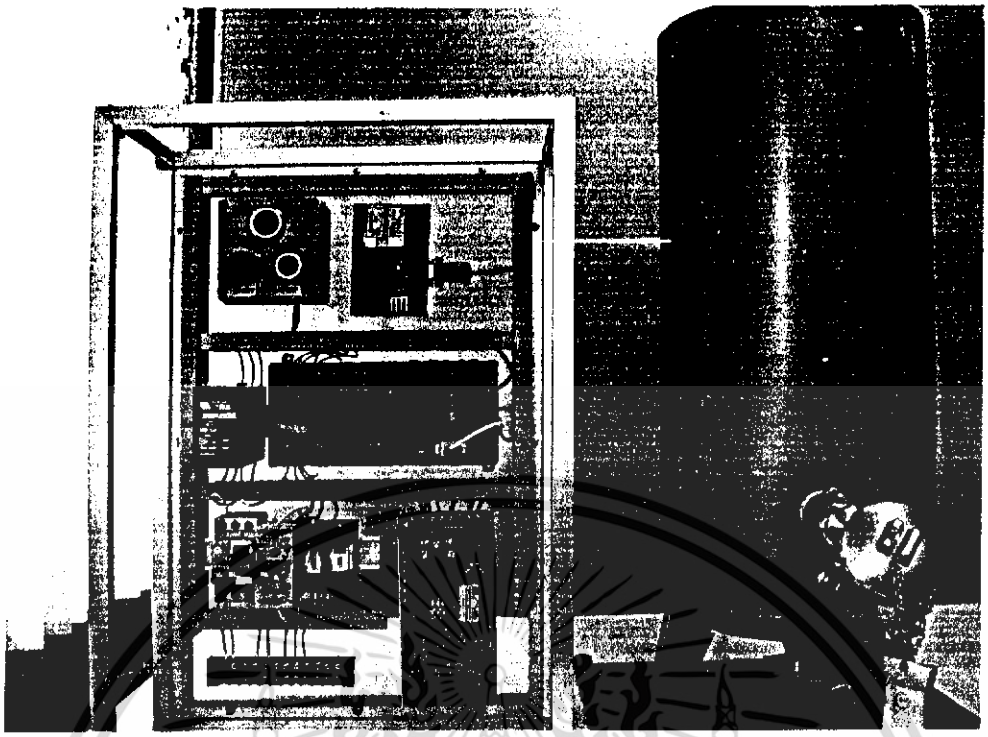
- Process (Temperature) หรือ Plant คือ Tank ที่ใช้ทำความร้อนประกอบด้วย Heater ทำความร้อน

- Transducer คือ ตัวตรวจจับอุณหภูมิ โดยใช้ RTD (PT100) ในการตรวจจับโดย โดย RTD จะให้ค่าความต้านทานเปลี่ยนแปลงไปตามอุณหภูมิ แล้วส่งค่าความต้านทานที่ได้ให้กับวงจรภาค Transmitter

- Transmitter ทำการแปลงค่าความต้านทานที่ได้รับมาจาก Transducer เป็นสัญญาณมาตรฐาน 4 – 20 mA และส่งค่ากระแสที่ได้ให้กับ Controller

- Controller ทำการแปลงค่ากระแส 4 – 20 mA ที่ได้มาจาก Transmitter เป็นค่าอุณหภูมิ 0 – 60 องศาเซลเซียส (ค่า PV) เพื่อทำการเปรียบเทียบค่าที่ได้ว่าตรงกับค่า Set Point (รับมาจากคอมพิวเตอร์) ซึ่งเมื่อค่าที่ได้รับมา (PV) น้อยกว่า ค่า Set Point Controller ก็จะทำการ on ค่า Y0 ส่งไปให้กับ Solid State เพื่อทำให้ Heater ที่ติดอยู่กับ Plant Process ทำงาน และเมื่อ ค่าที่ได้รับมา (PV) มีค่ามากกว่า ค่า Set Point Controller ก็จะทำการ off ค่า Y0 ส่งไปให้กับ Solid State เพื่อทำให้ Heater ที่ติดอยู่กับ Plant Process หยุดทำงาน

- Computer ทำการแสดงค่าการทำงานของ Process และเป็นตัวส่งค่า Set point ให้กับตัว Controller

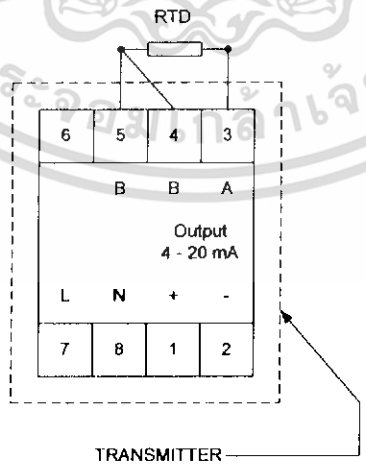


ภาพที่ 4.2 แสดงภาพถ่ายของ Hardware

ในปริญญานิพนธ์นี้แบ่งออกเป็น 2 ส่วน คือ

4.1 ส่วนของฮาร์ดแวร์ (Hardware)

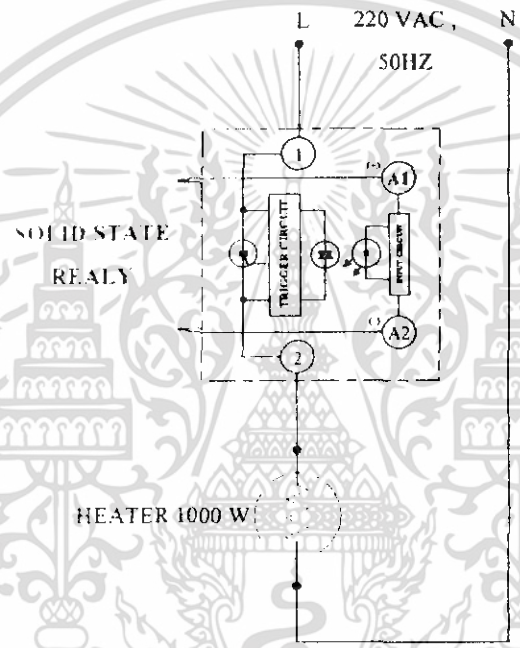
4.1.1 Transmitter and RTD เป็นอุปกรณ์วัดอุณหภูมิที่มีวงจรภายใน ดังภาพที่ 4.3



ภาพที่ 4.3 แสดงอุปกรณ์ และตำแหน่งขาของ RTD และ Transmitter

เมื่อตัวอุปกรณ์ RTD ที่ใช้คือ PT100 จะถูกบรรจุอยู่ใน Protectint Tube เพื่อป้องกันการสัมผัสกับของเหลวโดยตรง โดยวัดค่าอุณหภูมิจะทำการแปลงค่าอุณหภูมิเป็นค่าความต้านทานทางไฟฟ้าในรูปของงความต้านทานเปลี่ยนแปลงโดยตัว RTD จะใช้สาย 3 สาย เพื่อชดเชยค่าอุณหภูมิต่อเข้ากับตัว Transmitter โดยอุปกรณ์ Transmitter ที่ใช้จะใช้ตัวที่แปลงค่าความต้านทานให้เป็นสัญญาณมาตรฐานทางไฟฟ้า ที่ใช้คือ 4-20 mA ส่งค่าของการเปลี่ยนแปลงทางความต้านทานในรูปของสัญญาณกรระแสไฟฟ้าอยู่ในระหว่าง 4-20 mA ส่งต่อให้กับตัว PLC

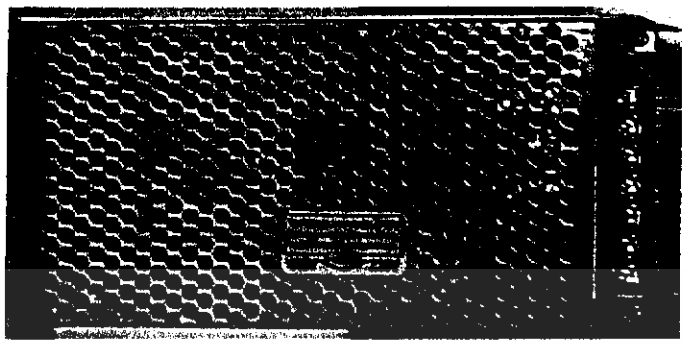
4.1.2 Solid State Relay



ภาพที่ 4.4 แสดงตัวอุปกรณ์ วงจรภายใน และตำแหน่งขาของ Solid State Relay

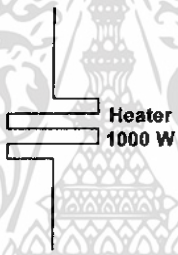
หลักการทำงานโดยทั่วไปทำงานเหมือน Relay แต่เนื่องจากความไวของ Relay มีความไวน้อยจึงนำ Solid State Relay มาทำการต่อ เพราะต้องการให้ความไวในการ NO – OFF เร็ว เนื่องจากอุปกรณ์ที่ใช้ทำมาจากสารกึ่งตัวนำ และยังสามารถตัดปัญหาเรื่องการสปาร์ค โดยได้รับการสั่งงานจาก PLC เพื่อควบคุม Heater

4.1.3 Power Supply



ภาพที่ 4.5 แสดงอุปกรณ์ภาคจ่ายไฟที่ใช้

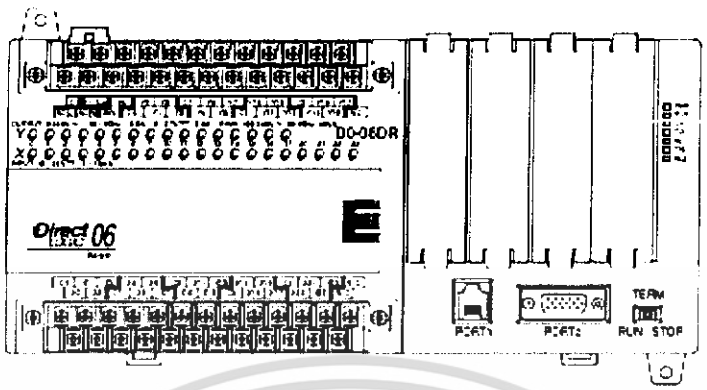
4.1.4 Heater



ภาพที่ 4.6 แสดงสัญลักษณ์ของ Heater

Heater 1000 W จะเป็นตัวสร้างความร้อนโดยรับพลังงานจาก Solid State Relay โดย Solid State Relay จะเป็นตัวตัดต่อไฟฟ้า 220 VAC เมื่อ Heater ร้อนจะถูกถ่ายเทความร้อนลงสู่ของเหลว ซึ่งอุณหภูมิของของเหลวนี้เองที่จะทำการควบคุม

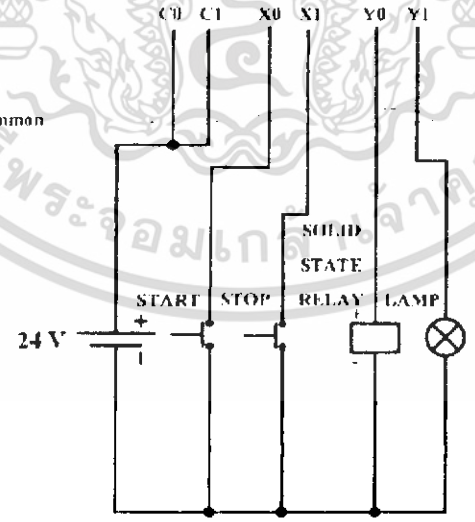
4.1.5 PLC



ภาพที่ 4.7 แสดงลักษณะของ PLC

PLC เป็นตัวควบคุมกระบวนการจัดค่าพารามิเตอร์ต่างๆ ทั้งตัวรับค่าอุณหภูมิจากทรานดิวเซอร์ และส่งสัญญาณควบคุมให้กับ Solid State Relay โดย PLC จะมีการเชื่อมต่อไปยังคอมพิวเตอร์ส่งค่าอุณหภูมิ(ในรูปของข้อมูล) และรับสัญญาณ Operate

การจัดอุปกรณ์อินพุต และเอาต์พุต ต่อเข้ากับขั้วอินพุตและเอาต์พุตของ PLC โดยลักษณะการต่อเป็นแบบ Souse ต่อไฟเลี้ยง บวก เข้ากับ Common ของ PLC กับขาด้านหนึ่งของอุปกรณ์อินพุต และเอาต์พุต



ภาพที่ 4.8 แสดงลักษณะการต่ออุปกรณ์อินพุตและเอาต์พุต

4.2 ส่วนของซอร์ฟแวร์ (Software)

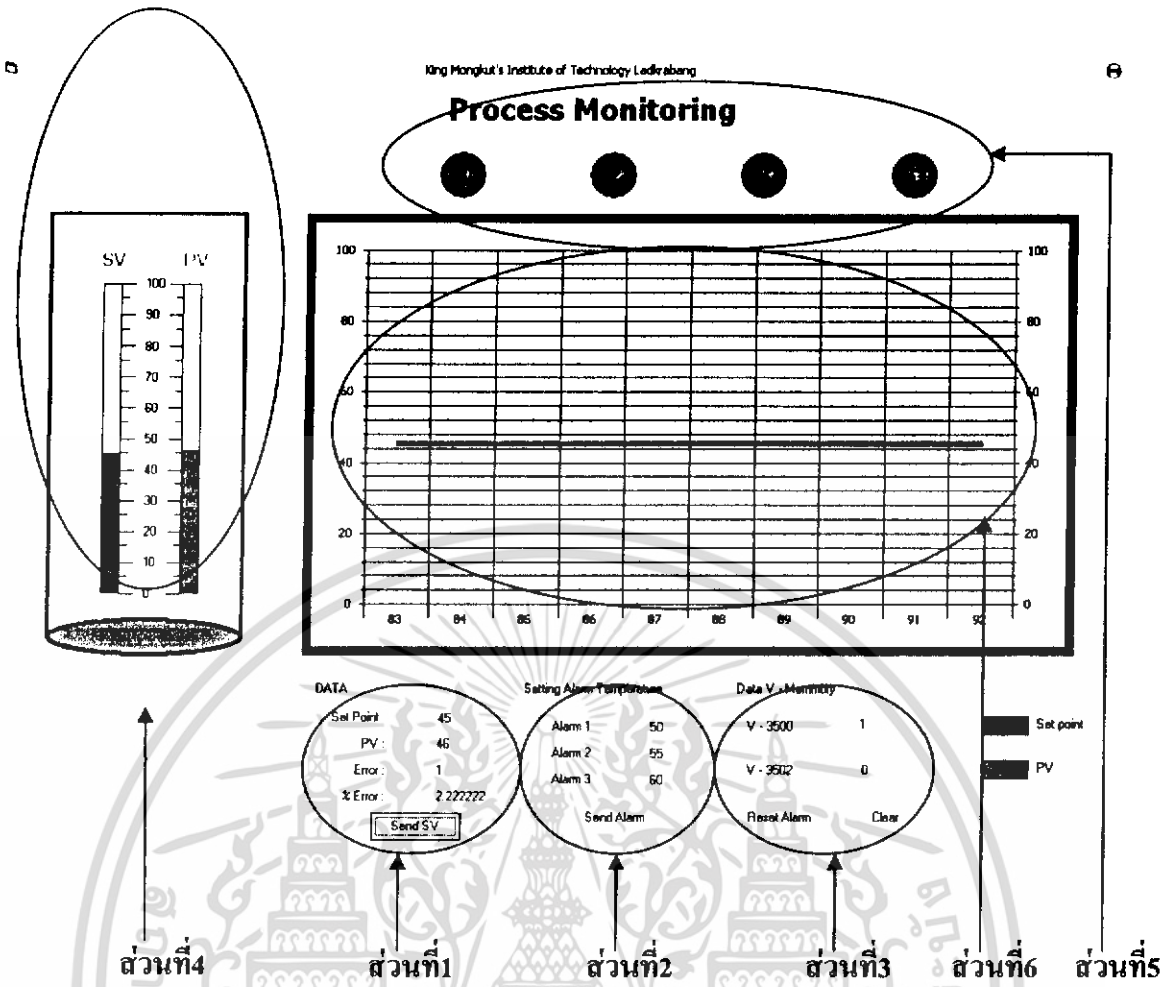
4.2.1 โปรแกรมของ PLC

ตารางที่ 4.1 แสดง I/O Assignment ของ PLC

ตำแหน่งหน่วยความจำ อินพุต - เอาท์พุต	รายการอุปกรณ์ และการใช้งาน	หมายเหตุ
X0	Start	เริ่มการทำงานของระบบ
X1	Stop	จบการทำงานของระบบ
X2	Reset	Reset Alarm
Y0	Heater 1000 W 220 VAC	ให้ความร้อนระบบ
V2000	รับสัญญาณ Analog จาก Transmitter 4-20 mA	
V2010	ส่งค่าอุณหภูมิ PV ไปยังคอมพิวเตอร์	
V4000	ส่งค่า Alarm 1 จากคอมพิวเตอร์ ไปยัง PLC	
V4002	ส่งค่า Alarm 2 จากคอมพิวเตอร์ ไปยัง PLC	
V4010	ส่งค่า Alarm 3 จากคอมพิวเตอร์ ไปยัง PLC	
V10002	รับค่าอุณหภูมิ SV จากคอมพิวเตอร์	

4.2.2 โปรแกรมหน้าจอแสดงผล (Microsoft Visual Basic)

เป็นส่วนที่ใช้ติดต่อผู้ใช้โดยตรง โปรแกรมจะติดต่อแลกเปลี่ยนข้อมูลกับ PLC โดยใช้โปรแกรม DS Data เป็นตัวช่วยจัดทำโปรโตคอลเพื่อให้โปรแกรมที่เขียนมาจาก Visual Basic และ PLC สามารถที่จะสื่อสารกันได้ โปรแกรมที่สร้างจะประกอบด้วยส่วนต่างๆ ดังนี้



ภาพที่ 4.9 แสดงหน้าจอแสดงผล (Microsoft Visual Basic)

การทำงานของส่วนต่างๆ ของหน้าจอ SPC (Microsoft Visual Basic)

ส่วนที่1 เป็นส่วนที่แสดงค่า Data ต่างๆได้แก่ Set Point , PV , Error และ %Error โดยค่า Set Point จะส่งไปยัง PLC ค่า PV จะรับมาจาก PLC มาแสดงผล

ส่วนที่ 2 เป็นส่วนที่กำหนดค่า Alarm ให้กับ PLC โดยมีลำดับความสำคัญของการเตือน ได้แก่ค่า Alarm 1, Alarm 2 และ Alarm3 โดยการกดปุ่ม Send Alarm

ส่วนที่ 3 แสดงค่าที่รับมาจาก V-Memory ของ PLC บางค่าที่ใช้ในการ Reset ค่าเมื่อมี Alarm แสดงผลแล้ว และถ้าต้องการ Reset ก็ทำการกดปุ่ม Reset Alarm และ กด ปุ่ม Clear

ส่วนที่ 4 เป็นส่วนที่ใช้แสดงผลค่า Set Point และค่า PV โดยแสดงผลเป็นบาร์กราฟ

ส่วนที่ 5 เป็นส่วนที่ใช้แสดง Status และลำดับความสำคัญของการเกิด Alarm โดยที่ 1, 2 และ3 เป็นลำดับการเกิด Alarm ของการควบคุมที่ผิดพลาด ส่วน S1 เป็น Alarm ในส่วนของ Sensor

ส่วนที่ 6 เป็นส่วนที่ใช้แสดงผลค่า Set Point และค่า PV โดยแสดงผลเป็นกราฟเส้น

บทที่ 5

การทดลองและผลการทดลอง

5.1 จุดประสงค์ในการทดลอง

1. เพื่อทดสอบว่าหน้าจอที่เขียนด้วยโปรแกรม Visual Basic สามารถควบคุมการทำงานของ Plant ได้จริง

2. เพื่อทดสอบว่าเมื่อระบบเกิดความล้มเหลว หรือ เกิดความผิดพลาดไม่ว่าจะเป็นในเรื่องของอุปกรณ์ที่ใช้ในการตรวจวัด หรือ อุณหภูมิที่ทำการวัด ให้ระบบทำการแจ้งเตือนส่งมายังโทรศัพท์เคลื่อนที่ในรูปของ SMS ได้จริง

5.2 การออกแบบการทดลอง

1. ระบบที่ใช้ในการทดลองนี้ เป็นการจำลองการตรวจจับความผิดพลาดที่เกิดจากอุปกรณ์ที่ใช้วัด หรือ ตัวอุณหภูมิของน้ำที่ทำการวัด

2. ระดับอุณหภูมิคงที่กำหนดโดยเงื่อนไขของโปรแกรมที่เขียนให้กับ PLC เพื่อสั่งให้ขดลวดความร้อนทำงาน

3. ปริมาณที่ใช้ในการทดลอง 15 ลิตร

4. ทำการทดลองสถานการณ์การเกิดความผิดพลาดของอุณหภูมิที่เกิดจากระบบโดยการกำหนดค่าผ่านทางหน้าจอ SPC (Visual Basic)

5. ทำการจำลองสถานการณ์การเกิดความผิดพลาดจากอุปกรณ์ที่ใช้ในการตรวจวัดอุณหภูมิ *(RTD)*

5.3 ขั้นตอนการทดลอง

1. *บันทึกค่าปริมาณน้ำ อุณหภูมิน้ำ อุณหภูมิห้อง ก่อนการทดลอง*

2. ทำการเริ่มกระบวนการเพื่อควบคุมอุณหภูมิ

3. กำหนดค่า Set Point ให้กับกระบวนการเพื่อควบคุมการทำงานของขดลวดความร้อน

4. เมื่อระบบอยู่ในสภาวะคงตัวแล้ว ให้จำลองสถานการณ์ความผิดดังนี้

4.1 เมื่อตัวตรวจวัดอุณหภูมิเสียหาย ให้ทำการปลดสัญญาณอนาลอกที่เข้ามาออกแล้วทำการบันทึกผลการทดลอง

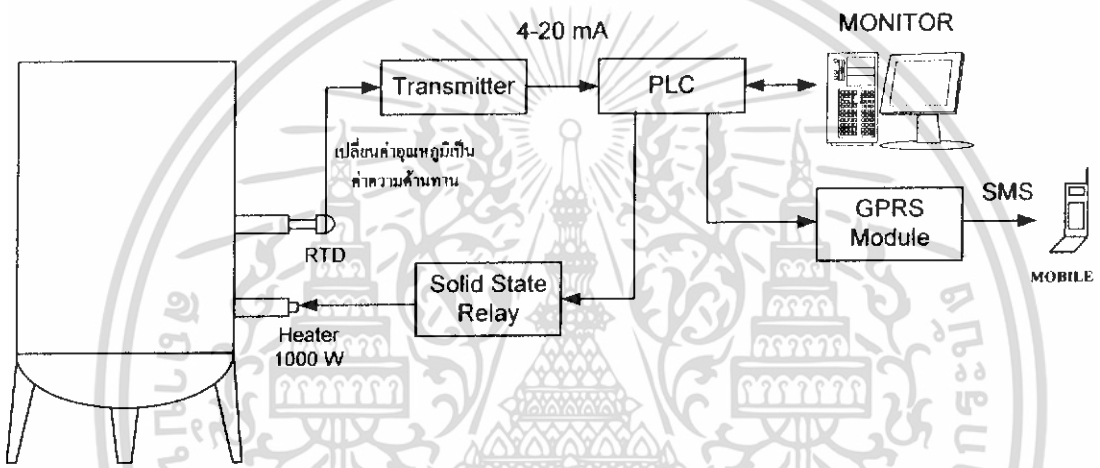
4.2 เมื่ออุณหภูมิมีค่าต่ำกว่าหรือสูงกว่าค่าปลอดภัยที่ตั้งไว้ โดยการเข้าไปกำหนดค่าที่หน้าจอ แล้วบันทึกผลการทดลอง

5. ทำการทดลองซ้ำอีกครั้ง ที่ระดับอุณหภูมิต่างๆกัน จากนั้นบันทึกผลการทดลอง

5.4 บันทึกผลการทดลอง

1. บันทึกค่าปริมาณน้ำ, อุณหภูมิน้ำและ อุณหภูมิห้อง ก่อนการทดลอง
   - ปริมาณน้ำที่ใช้ในการทดลองเท่ากับ 5 ลิตร
   - อุณหภูมิน้ำเท่ากับ 32.00°C
   - อุณหภูมิห้องเท่ากับ 34.00°C

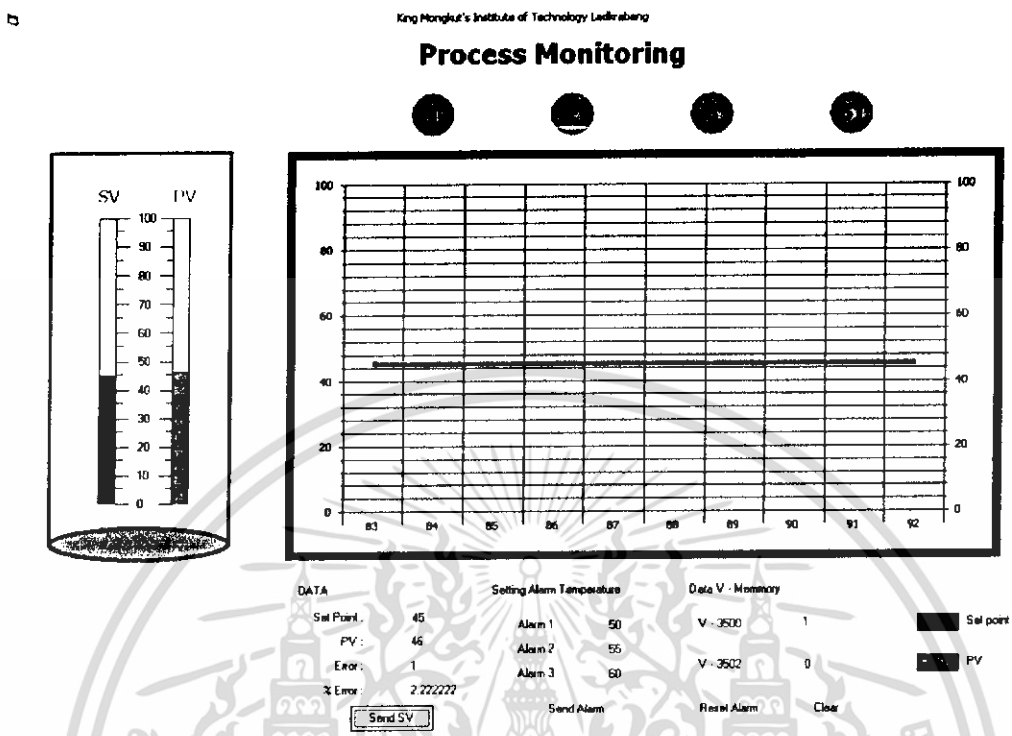
2. ทำการเริ่มกระบวนการเพื่อควบคุมอุณหภูมิ



ภาพที่ 5.1 แสดงกระบวนการการทำงานของระบบที่ใช้ทดลอง

จากภาพที่ 5.1 เป็นบล็อกแสดงการทำวานของระบบที่ใช้ในการทดลอง โดยลักษณะการทำงานของระบบคือใช้ PLC เป็นตัว Controller ควบคุมอุณหภูมิของน้ำให้เป็นไปตามค่า Set Point ที่ตั้งเอาไว้ โดยสามารถตั้งค่า Set Point ผ่านทางหน้าจอคอมพิวเตอร์ซึ่งเราใช้โปรแกรม Visual Basic ในการรับและส่งค่าต่างๆให้กับตัว PLC ซึ่งจากการทดลองได้ทำการตั้ง Set Point ไว้ที่ 40°C จากนั้นทำการตั้งค่าของ High Alarm และ Low Alarm ไว้ที่ 50°C และ 35°C ตามลำดับ ดังนั้นถ้าอุณหภูมิเกิดมีค่าสูงกว่า High Alarm หรือต่ำกว่า Low Alarm ที่ตั้งเอาไว้ระบบจะต้องมีการแจ้งเตือนออกมาที่ Monitor และมีการส่งข้อความแจ้งเตือนมายังโทรศัพท์มือถือ

3. เมื่อระบบอยู่ในสภาวะคงตัวแล้วทำการบันทึกผล



ภาพที่ 5.2 แสดงค่าตัวแปรต่างๆที่รับค่ามาจากกระบวนการที่ทำการควบคุมอยู่

จากภาพที่ 5.2 จะเห็นว่าในหน้าจอแสดงผลจะแบ่งออกเป็น5ส่วนด้วยกันคือ

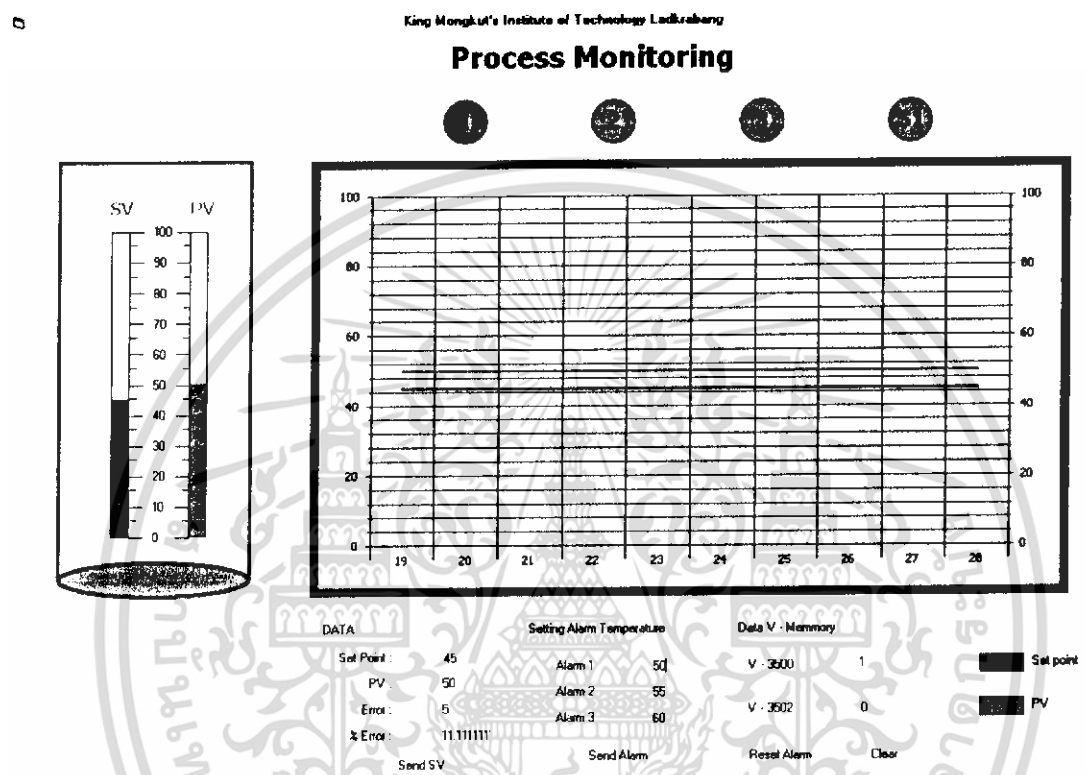
1. บาร์กราฟและกราฟแสดงสภาวะการทำงานระหว่างค่าของ set point กับค่าของ Process variable

2. สัญญาณไฟแสดงสภาวะการทำงานและแสดงสภาวะการแจ้งเตือนของ RTD และอุณหภูมิของกระบวนการ

3. DATA เป็นส่วนที่ไว้สำหรับรับส่งค่าที่ต้องการให้กับกระบวนการ เช่น การกำหนดค่าให้กับ set point, การดูค่าสภาวะการทำงานจาก PV (Process Variable), การดูค่าความผิดพลาด (Error) และเปอร์เซ็นต์ความผิดพลาดของกระบวนการ

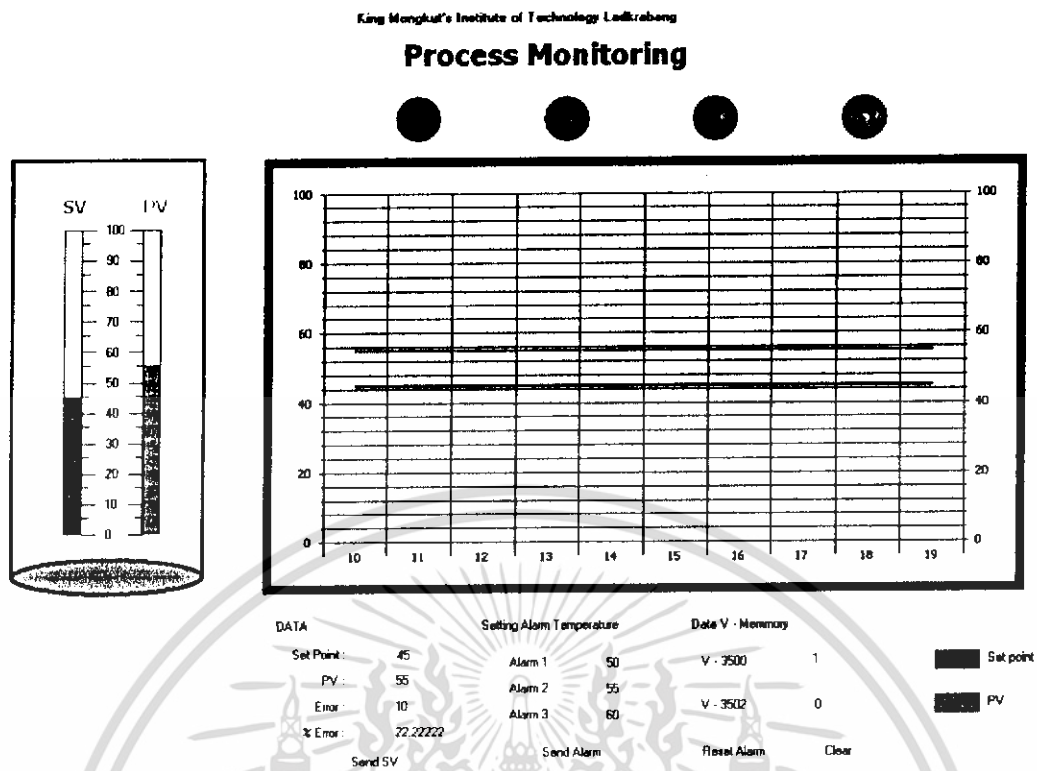
4. Setting Alarm Temperature เป็นส่วนที่ไว้สำหรับกำหนดค่าของอุณหภูมิที่ต้องการให้ทำการแจ้งเตือนโดยมีระดับการแจ้งเตือน 3 ระดับคือ Alarm1 เป็นการเตือนในระดับแรก คือกระบวนการผลิตเริ่มเกิดความผิดปกติ, Alarm2 เป็นการเตือนในระดับที่สองซึ่งจะมีความรุนแรงกว่าระดับแรกแต่ยังพอที่จะยอมรับและแก้ไขได้ และ Alarm3 เป็นการแจ้งเตือนในระดับสุดท้ายที่กระบวนการผลิตไม่สามารถยอมรับได้

5. DATA V-memory เป็นส่วนที่ไว้สำหรับทำการ Reset ค่าการทำงานของระบบในกรณีที่มีการแจ้งเตือนออกไปแล้ว นั่นหมายความว่าทุกครั้งที่ระบบทำการส่ง Message แจ้งเตือนออกมา หลังจากที่เราทำการแก้ไข้ข้อมูลให้กับระบบใหม่แล้วก่อนทำการเริ่มกระบวนการใหม่อีกครั้งเราต้องทำการ Reset ค่าในส่วนนี้ทุกครั้ง



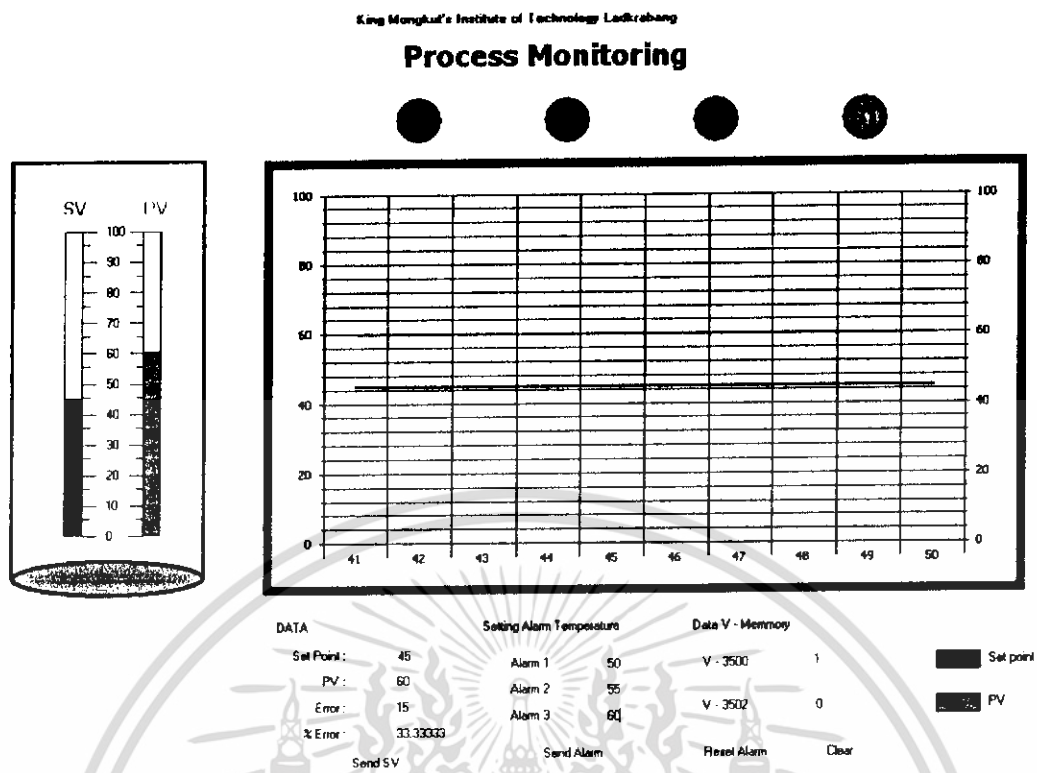
ภาพที่ 5.3 แสดงค่าสภาวะทำงานขณะ Alarm1

จากภาพที่5.3 จะพบว่าอุณหภูมิที่กำลังทำการวัดค่าอยู่นั้น (PV) มีค่าสูงกว่าค่าของอุณหภูมิที่เราต้องการให้แจ้งเตือน (Alarm1) ฉะนั้นที่หน้าจอคอมพิวเตอร์ก็จะแสดงสภาวะ "1" เป็นสีแดง โดยที่ Alam1 อาจจะยังไม่ร้ายแรงเท่าไรนักแต่ถ้าผู้ปฏิบัติงานยังไม่ทำการแก้ไขปรับปรุงกระบวนการให้ดีขึ้นกระบวนการก็จะทำงานต่อไปจนกว่าถึงระดับการแจ้งเตือนที่ 2 ดังภาพที่ 5.4



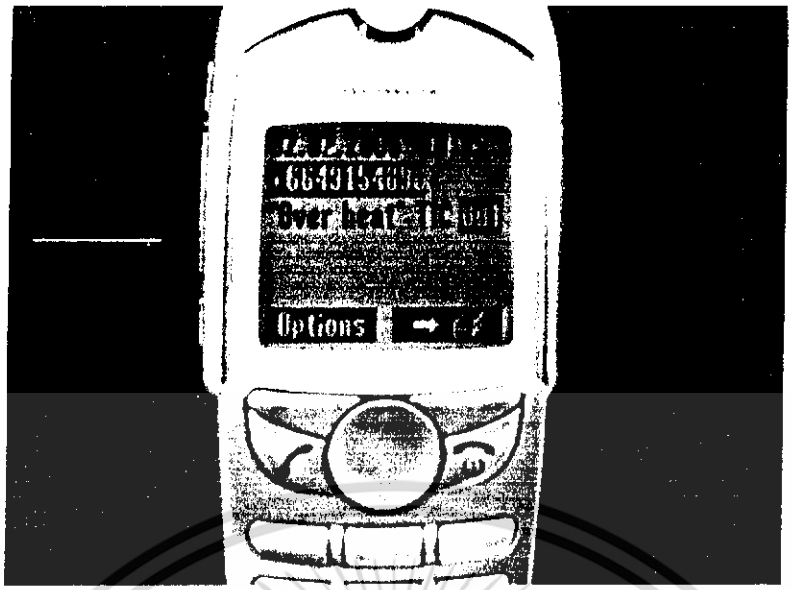
ภาพที่ 5.4 แสดงค่าสภาวะทำงานขณะ Alarm2

จากภาพที่ 5.3 จะพบว่าอุณหภูมิที่กำลังทำการวัดค่าอยู่นั้น (PV) มีค่าสูงกว่าค่าของอุณหภูมิที่เราต้องการให้แจ้งเตือนในระดับที่2 (Alarm2) ฉะนั้นที่หน้าจอคอมพิวเตอร์ก็จะแสดงสภาวะ "2" เป็นสีแดง แต่ถ้าระหว่างนี้ผู้ปฏิบัติยังไม่ทำการปรับปรุงแก้ไข กระบวนการก็จะทำงานไปจนถึงการเตือนระดับสุดท้ายที่ยอมรับได้ ดังภาพที่ 5.5

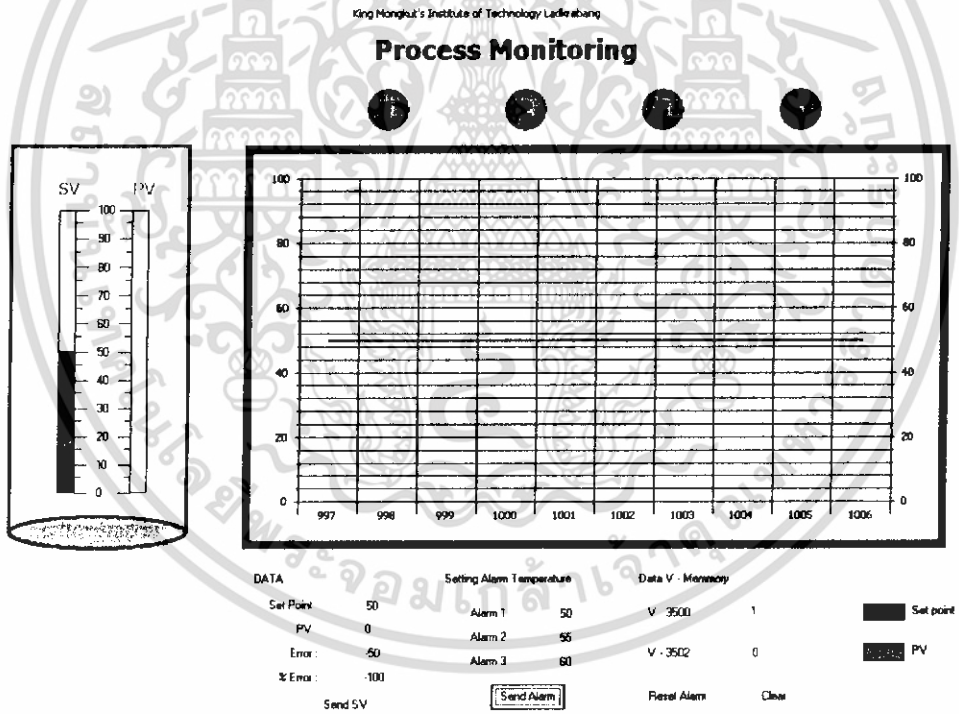


ภาพที่ 5.5 แสดงผลของการแจ้งเตือนของระบบในระดับสุดท้าย(Alarm3)

จะพบว่าอุณหภูมิที่กำลังทำการวัดค่าอยู่นั้น (PV) มีค่าสูงกว่าค่าของอุณหภูมิที่เราต้องการให้แจ้งเตือนในระดับที่3 (Alarm3) ฉะนั้นที่หน้าจอคอมพิวเตอร์ก็จะแสดงสภาวะ "3" เป็นสีแดง ซึ่งในระดับนี้เราถือว่าผู้ปฏิบัติงานเกิดความละเลยอย่างมาก อาจส่งผลเสียต่อกระบวนการผลิตทำให้ผลผลิตเกิดความเสียหาย ดังนั้นเราจึงทำการส่งข้อความแจ้งเตือนมายังโทรศัพท์มือถือเพื่อให้ผู้ควบคุมหรือผู้บริหารระดับสูงได้รับทราบถึงเหตุการณ์ที่เกิดเพื่อที่จะได้ทำการตัดสินใจต่อไป



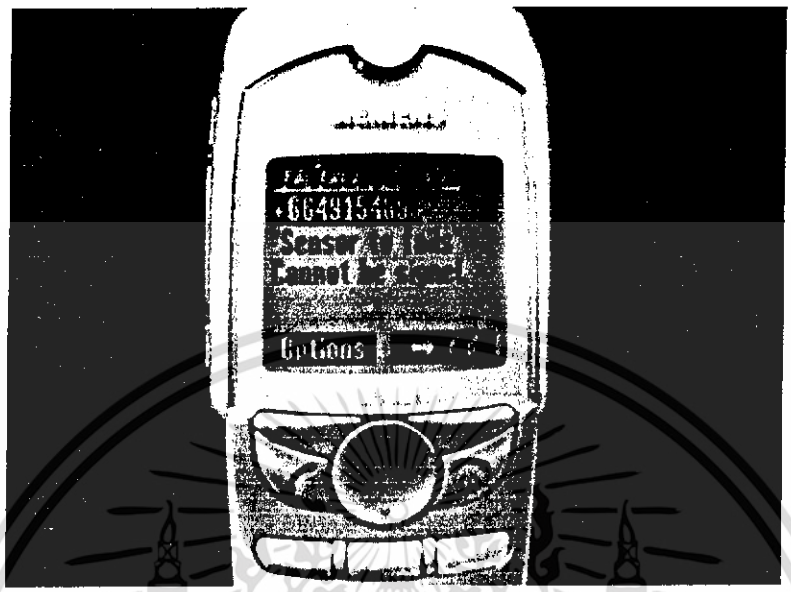
ภาพที่ 5.6 แสดงผลของข้อความเตือนที่ส่งมายังโทรศัพท์มือถือ



ภาพที่ 5.7 แสดงค่าสภาวะทำงานขณะ RTD เกิดความเสียหาย

จากภาพที่ 5.7 จะพบว่าถ้า RTD เกิดความเสียหายทำให้เราไม่สามารถรับรู้ค่าอุณหภูมิที่กำลังทำการวัดค่าอยู่นั้น (PV) ส่งผลให้ระบบไม่สามารถควบคุมอุณหภูมิของกระบวนการได้

ฉะนั้นเราจึงสั่งให้ระบบทำการตัดการทำงานของระบบและให้ทำการส่ง Message แจ้งเตือนมายังโทรศัพท์มือถือ ดังภาพที่ 5.8



ภาพที่ 5.8 แสดงผลของการแจ้งเตือนของระบบในส่วนของ Sensor Alarm

5.5 สรุปผลการทดลอง

1. เมื่อทำการทดสอบโดยการควบคุมอุณหภูมิของน้ำให้มีค่าเป็นไปตาม set point ที่กำหนด ปรากฏว่าระบบสามาควบคุมอุณหภูมิของน้ำได้จริง โดยในโครงงานนี้เราได้กำหนดค่าของ set point ไว้ไม่ให้เกิน 60°c

2. เมื่อทำการทดสอบระบบแจ้งเตือนอันเกิดมาจากความผิดปกติของกระบวนการปรากฏว่าระบบสามารถทำการแจ้งเตือนออกมาได้จริงอย่างทันทวงที ไม่ว่าจะเป็นในส่วนของ Alarm1, Alarm2, Alarm3 และ Sensor อีกทั้งยังส่งข้อความเตือนมายังโทรศัพท์มือถือได้จริง

บทที่ 6
สรุปผลการวิจัยและข้อเสนอแนะ

6.1 สรุปผลงาน

จากการทดลองเรานำเอา PLC ซึ่งมีลักษณะการทำงานแบบ Sequential มาทำการควบคุมอุณหภูมิของกระบวนการซึ่งมีลักษณะการควบคุมแบบ On-Off โดยที่ค่าสภาวะต่างๆของการทำงานของกระบวนการสามารถแสดงผลออกมาได้ทางหน้าจอคอมพิวเตอร์ โดยใช้โปรแกรม Visual Basic 6.0 เป็นตัวเชื่อมต่อกับ PLC นอกจากนั้นเรายังนำเอา GPRS Module มาทำการต่อเข้ากับระบบเพื่อทำการแจ้งเตือนโดยตัว GPRS Module จะเชื่อมต่อกับ PLC และสื่อสารกันโดยใช้รหัส ASCII ส่งสัญญาณแจ้งเตือนผ่านระบบเครือข่ายโทรศัพท์มือถือในรูปแบบของ SMS (Short Messaging Service) ซึ่งจากผลการทดลองพบว่า PLC สามารถควบคุมอุณหภูมิได้ตามที่ต้องการ อีกทั้งยังสามารถตรวจสอบความผิดพลาดของกระบวนการและส่งค่าไปสั่งงาน GPRS Module เพื่อส่งข้อความเตือนมายังโทรศัพท์มือถือ และยังแสดงผลออกหน้าจอคอมพิวเตอร์ได้จริง

6.2 ข้อเสนอแนะและแนวทางพัฒนา

การสร้างเครื่องมือในการควบคุมหรือแจ้งเตือนกระบวนการผลิตนั้นมีหลากหลายวิธีการในการปฏิบัติโดยการทดลองในปริญญานิพนธ์นี้ ถือเป็นเครื่องมือที่สำคัญชนิดหนึ่งในการวิเคราะห์การควบคุมและแจ้งเตือนของกระบวนการ

ซึ่งในกระบวนการควบคุมและแจ้งเตือนยังมีหลักการและวิธีคิดอื่นๆในการวิเคราะห์ และควบคุมเพื่อให้ระบบมีความหลากหลายและง่ายต่อการใช้งานในกระบวนการที่ต่างกันออกไป เพื่อเป็นการต่อยอดองค์ความรู้ดังกล่าวในอนาคต จึงควรมีการนำเอารูปแบบการส่งในรูปแบบอื่นๆที่นอกเหนือจาก SMS เช่น MMS หรือ GPRS เป็นต้นและทฤษฎีอื่นมาใช้ร่วมในการวิเคราะห์และส่งข้อความแจ้งเตือนที่เกิดจากกระบวนการ อีกทั้งยังควรทำการทดลองกับระบบอื่นๆที่มีคุณลักษณะทางกายภาพต่างกันออกไป อันจะเป็นการพัฒนาความรู้อย่างต่อเนื่องเช่นเดียวกับจุประสงค์ของการแจ้งเตือน

บรรณานุกรม

กิตติศักดิ์ พลอยพานิชเจริญ **การวิเคราะห์ระบบการวัด** , สมาคมส่งเสริมเทคโนโลยีไทย-ญี่ปุ่น.

ประสิทธิ์ จุลเสรีวงศ์ **เอกสารประกอบการสอน วิชาวิศวกรรมการวัดคุม** ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

สัจจะ จรัสรุ่งรวีวร **Visual Basic 6** พิมพ์ครั้งที่6 นนทบุรี: ไอ ดี ซี อินโฟ ดิสทริบิวเตอร์ ,2544.

อภิชาติ ภู่พลับ **เริ่มต้นเขียนโปรแกรมติดต่อและควบคุมฮาร์ดแวร์ด้วย Visual Basic** นนทบุรี: Infopress Developer Book ,2546.

PLC Direct Intercorporated , **DL 06 User Manual** , 1994 www.Indusoft.com

ภาคผนวก

Documentazione riservata. Questo documento non può essere riprodotto, anche parzialmente, né reso noto a terzi senza preventiva autorizzazione scritta di DAI Telecom S.p.A. Tutti i diritti sono tutelati a norma di legge.

Restricted document. No part of this document can be reproduced or copied in any format, in part or totally, nor its contents revealed in any manner to others without the express written permission of DAI Telecom S.p.A. All rights reserved

2.5.1.3 +CMGF - message format

+CMGF –message format	
Read command **AT+CMGF?**	Reports the current value of the parameter <mode> = 0.
Set command **AT+CMGF = <mode>**	Selects the SMS format to be used in reading and writing messages. Parameter: <mode> = 0 - PDU as defined in GSM 3.40 and GSM 3.41 <mode> = 1 – text
Test command **AT+CMGF=?**	Reports the supported value of <mode> parameter.
Example	
Reference	GSM 07.05
Note	

2.5.1.4 +CSMP – Set parameters in text mode

+CSMP –set parameters in text mode	
Read command **AT+CSMP?**	Reports the current setting in the format: +CSMP: < fo>,<vp>,<pid>,<dcs>
Set command **AT+CSMP =** **<fo>,<vp>,<pid>,<dcs>**	Set the additional parameters for storing and sending SMS when the text mode is used (+CMGF=1) Parameter: <fo>: message format, like defined for the first octet of message according to GSM 3.40 <vp>: Message validity period - numerical if in relative format or string if in absolute format according to GMS 3.40 <pid>: Protocol Identifier – defined by GSM 3.40 – in numerical format <dcs>: Data coding Scheme - defined by GSM 3.40 – in numerical format
Test command **AT+CSMP=?**	Reports the supported range of values for <fo>,<vp>,<pid>,<dcs> parameters.

Example	Set the parameters for an outgoing message with 24 hours of validity period and default properties:
	`AT+CSMP=17,167,0,0`
	`OK`

Reference	GSM 07.05
Note	

Documentazione riservata. Questo documento non può essere riprodotto anche parzialmente, né reso noto a terzi senza preventiva autorizzazione scritta di DAI Telecom S.p.A.
Tutti i diritti sono tutelati a norma di legge.

Restricted document. No part of this document can be reproduced or copied in any format, in part or totally, nor its contents revealed in any manner to others without the express written permission of DAI Telecom S.p.A.
All rights reserved.

2.5.4.3 +CMGW - write message to memory

+CMGW – write message to memory	
Set command *(PDU Mode)* **AT+CMGW = <length>**	**(PDU Mode)** Writes in the <memw> memory storage a new SMS message input as a PDU. Parameter: <length> = 7...164 - represents the length of the PDU to be written in bytes. The device responds to the command with the prompt '>' and awaits for the number of bytes of PDU specified. To complete the operation send Ctrl-Z char (0x1A hexadecimal) to exit without writing the message send ESC char (0x1B hexadecimal). If message is successfully written in the memory, then the result is sent in the format: +CMGW: <index> where <index> is the message location index in the memory <memw>("SM"). If message storing fails for some reason, an error code is reported
(Text Mode) **AT+CMGW = [<da>]**	**(Text Mode)** Writes in the <memw> memory storage a new SMS message input as Text. Parameter: <da> - destination address number The device responds to the command with the prompt '>' and awaits for the message text (max 160 characters). To complete the operation send Ctrl-Z char (0x1A hexadecimal) , to exit without writing the message send ESC char (0x1B hexadecimal). If message is successfully written in the memory, then the result is sent in the format: +CMGW: <index> where <index> is the message location index in the memory <memw>("SM"). If message storing fails for some reason, an error code is reported
Example	
Reference	GSM 07.05
Note	Care must be taken to ensure that during the command execution, no other SIM interacting commands are issued. To avoid malfunctions is suggested to wait for the +CMGS:<mr> or +CMS ERROR:<err> response before issuing further commands.

Documentazione riservata. Questo documento non può essere riprodotto, anche parzialmente, né reso noto a terzi senza preventiva autorizzazione scritta di DAI Telecom S.p.A.
Tutti i diritti sono tutelati a norma di legge

Restricted document. No part of this document can be reproduced or copied in any format, in part or totally, nor its contents revealed in any manner to others without the express written permission of DAI Telecom S.p.A.
All rights reserved

2.5.4.2 +CMSS - send message from storage

+CMSS – send message from storage	
Set command **AT+CMSS = <index>**	Sends to the network, the message which is already stored in the <memw> storage (see +CPMS) at the location <index>. Parameter: <index> If message is successfully sent to the network, Sends to the network, then the result is sent in the format: +CMSS: <mr> where <mr> is the message reference number. If message sending fails for some reason, an error code is reported: +CMS ERROR:<err> Note: to store a message in the <memw> storage see command +CMGW. Care must be taken to ensure that during the command execution, which might take several seconds, no other SIM interacting commands are issued. To avoid malfunctions is suggested to wait for the +CMGS:<mr> or +CMS ERROR:<err> response before issuing further commands.
Example	
Reference	GSM 07.05
Note	

[3] - CR stands for Carriage Return character

For example:

1- Let's assume you want to find the IRA code for the character '&':

From the table you find:

- most significant Nibble: 2
- least significant Nibble: 6

Hence the IRA code for the '&' character is the hexadecimal 0x26.

2- Let's assume you have the IRA code 0x6B and you want to find the corresponding character:

From the table you find at the position

- most significant Nibble: 6
- least significant Nibble: B

Hence, the character corresponding to the 0x6B IRA code is 'k'.

3.5.3 Writing a New SMS to storage

A new SMS can be written in the selected storage <memw> (in the current SW version only "SM" is supported) and then can be sent to the desired destination.

To write the new SMS:

- send command **AT+CMGW="<da>"<cr>**

where:

<da>: destination address

- wait for prompt ">"
- send SMS text (MAX 160 characters)
- end command with CTRL-Z character (0x1A hexadecimal) or abort command with ESC character (0x1B hexadecimal)
- wait for response:

Response	Reason	Action
+CMGW: <index> OK	Message has been successfully written in position number <index>	proceed ahead
ERROR	some error occurred	Enable the extended error codes report (see par.2.5.2.1) and retry.

Reproduction forbidden without DAI Telecom written authorization – All Right reserved – Right of modification reserved

+CMS ERROR: 330	SMSC address unknown	Insert SMSC address (see par. 3.5.1.3)
+CMS ERROR: 322	Memory Full	memory is full, hence delete some records and retry.

NOTE: if command is aborted with ESC character, then only the OK result code is returned.

For example:

1- Let's assume you want to write a new SMS to the storage and the destination address is the number +39338123456789. We suppose you already have set up the device for text SMS mode as described on the previous paragraphs:

command:

AT+CMGW="+39338123456789"

response:

>

now you can insert the message text in IRA format (note that the IRA format and ASCII format coincide for the alphabet characters but not for the others).

.... here will be inserted the SMS message text....

conclude text with the character CTRL-Z

response:

+CMGW: 3

OK

In this case, the new SMS was successfully written to the location index 3 of the selected write memory (always "SM" SIM Card memory).

3.5.4 Sending an SMS previously stored

An already written SMS can be sent from the selected storage <memw> (in the current SW version only "SM" is supported).

To send the written SMS its location index is needed:

Reproduction forbidden without DAI Telecom written authorization – All Right reserved – Right of modification reserved

- send command **AT+CMSS=<index><cr>**

where:

<index>: SMS location index

- wait for response:

Response	Reason	Action
+CMSS: <mr> OK	Message has been successfully sent. <mr> represents the message reference number.	proceed ahead
ERROR	some error occurred	Enable the extended error codes report (see par.2.5.2.1) and retry.
+CMS ERROR: 330	SMSC address unknown	Insert SMSC address (see par. 3.5.1.3)
+CMS ERROR: 41	"Temporary Failure", may be that the device is not registered on any network	Check for signal strength and network registration (see par. 2.5.3)
+CMS ERROR: 331	No network service	Check for signal strength and network registration (see par. 2.5.3)
+CMS ERROR: 1	Unassigned number	The destination address number does not exist. Check it and repeat command.
+CMS ERROR: 42	network congestion	Retry later
+CMS ERROR: 96	Mandatory information missing	Check for destination address in the SMS, overwrite it and retry.

For example:

1- Let's assume you want to send a SMS that was written to the storage index position number 3. We suppose you already have set up the device for text SMS mode as described on the previous paragraphs:

command

AT+CMSS=3

response

+CMSS: 1

OK

In this case, the SMS was successfully sent to the destination and its network message reference number is 1.

3.5.5 Sending a new SMS without storing it

A new SMS can be sent directly to the network without storing it.

- send command **AT+CMGS="<da>"<cr>**

where:

<da>: destination address

- wait for prompt ">"
- send SMS text (MAX 160 characters)
- end command with CTRL-Z character (0x1A hexadecimal) or abort command with ESC character (0x1B hexadecimal)
- wait for response:

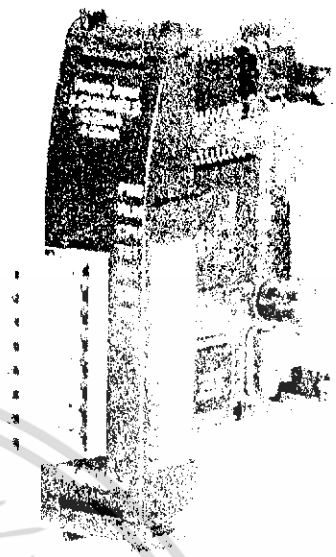
Response	Reason	Action
+CMGS: <mr> OK	Message has been successfully sent. <mr> represents the message reference number.	proceed ahead
ERROR	some error occurred	Enable the extended error codes report (see par.2.5.2.1) and retry.
+CMS ERROR: 330	SMSC address unknown	Insert SMSC address (see par. 3.5.1.3)
+CMS ERROR: 41	"Temporary Failure", may be that the device is not registered on any network	Check for signal strength and network registration (see par. 2.5.3)
+CMS ERROR: 331	No network service	Check for signal strength and network registration

Reproduction forbidden without DAI Telecom written authorization – All Right reserved – Right of modification reserved

Module Specifications

The F0-04AD-1 Analog Input module offers the following features:

- The DL05 and DL06 will read all four channels in one scan.
- The removable terminal block makes it possible to remove the module without disconnecting the field wiring.
- Analog inputs can be used as process variables for the four (4) PID loops in the DL05 and the eight (8) PID loops in the DL06 CPUs.
- Field device burn–out is detected on all four channels when 4–20mA range is selected.
- On-board active analog filtering and RISC-like microcontroller provide digital signal processing to maintain precise analog measurements in noisy environments.



NOTE: The DL05 CPU's analog feature requires ***Direct***SOFT32 Version 3.0c (or later) and firmware version 4.10 (or later).The DL06 requires ***Direct***SOFT32 version V4.0 (or later). See our website for more information: **www.automationdirect.com**.

The following tables provide the specifications for the F0–04AD–1 Analog Input Module. Review these specifications to make sure the module meets your application requirements.

Input Specifications

Number of Channels	4, single ended (one common)
Input Range	0 to 20 mA or 4 to 20 mA current (jumper selectable)
Resolution	12 bit (1 in 4096) for 0–20mA, scaled for 4–20mA
Step Response	25.0 mS (typ) to 95% of full step change
Crosstalk	–80 dB, 1/2 count maximum *
Active Low-pass Filtering	–3 dB at 40Hz (–12 dB per octave)
Input Impedance	125 Ohm ±0.1%, 1/8 W current input
Absolute Maximum Ratings	–30 mA to +30 mA, current input
Converter type	Successive approximation
Linearity Error (End to End)	±2 counts maximum *
Input Stability	±1 count *
Full Scale Calibration Error (Offset error not included)	±10 counts maximum, @ 20mA current input *
Offset Calibration Error	±5 counts maximum, @ 4mA current input *
Maximum Inaccuracy	±.4% @ 25°C (77°F) ±.85% 0 to 60°C (32 to 140°F)
Accuracy vs. Temperature	±100 ppm/°C maximum full scale calibration (including maximum offset change)
Recommended Fuse (external)	0.032 A, Series 217 fast-acting, current inputs

* One count in the specification table is equal to one least significant bit of the analog data value (1 in 4096).

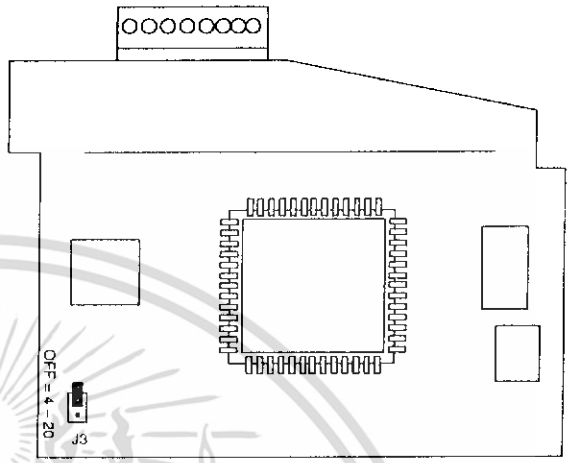
General Specifications

PLC Update Rate	4 channels per scan
16–bit Data Word	12 binary data bits, 2 channel ID bits, 2 diagnostic bits
Operating Temperature	0 to 60° C (32 to 140° F)
Storage Temperature	-20 to 70° C (-4 to 158° F)
Relative Humidity	5 to 95% (non-condensing)
Environmental air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3–304
Power Budget Requirement	50 mA @ 5VDC (supplied by base)
Connector	Phoenix Mecano, Inc. Part No. AK1550/8–3.5 – green
Connector Wire Size	28 – 16 AWG
Connector Screw Torque	0.4 Nm
Connector Screwdriver Size	DN–SS1 (recommended)

Setting the Module Jumper

The position of jumper J3 determines the input signal level. You can choose between 4–20mA and 0–20mA. The module ships with the jumper *not* connecting the two pins. In this position, the expected input signal is 4–20mA. To select 0–20mA signals, use the jumper to cover both pins.

The default jumper setting selects a 4–20mA signal source. The default jumper setting does *not* connect the two pins.



WARNING: Before removing the analog module or the terminal block on the face of the module, **disconnect power** to the **PLC** and **all field devices**. Failure to disconnect power can result in damage to the PLC and/or field devices.

Connecting and Disconnecting the Field Wiring

Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider:

- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter source. *Do not* ground the shield at both the module and the source.
- Do not run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- Route the wiring through an approved cable housing to minimize the risk of accidental damage. Check local and national codes to choose the correct method for your application.

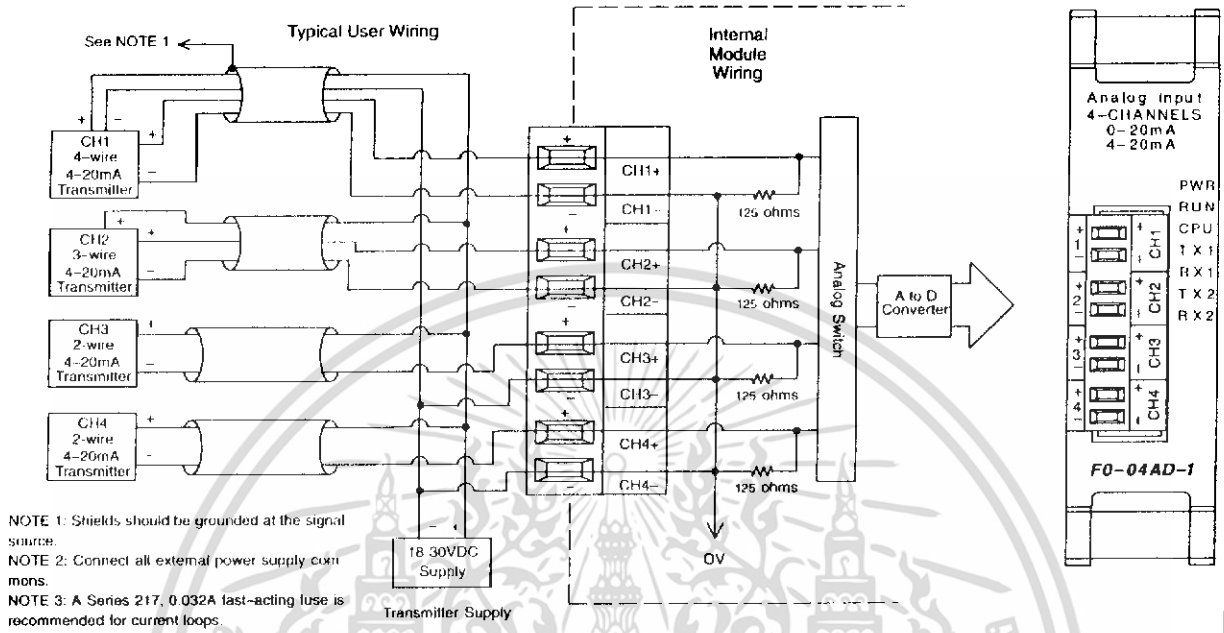
The F0–04AD–1 does not supply power to field devices. You will need to power transmitters separately from the PLC.

To remove the terminal block, disconnect power to the PLC and the field devices. Pull the terminal block firmly until the connector separates from the module.

You can remove the analog module from the PLC by folding out the retaining tabs at the top and bottom of the module. As the retaining tabs pivot upward and outward, the module's connector is lifted out of the PLC socket. Once the connector is free, you can lift the module out of its slot.

Wiring Diagram

Use the following diagram to connect the field wiring. If necessary, the F0–04AD–1 terminal block can be removed to make removal of the module possible without disturbing field wiring.



Current Loop Transmitter Impedance

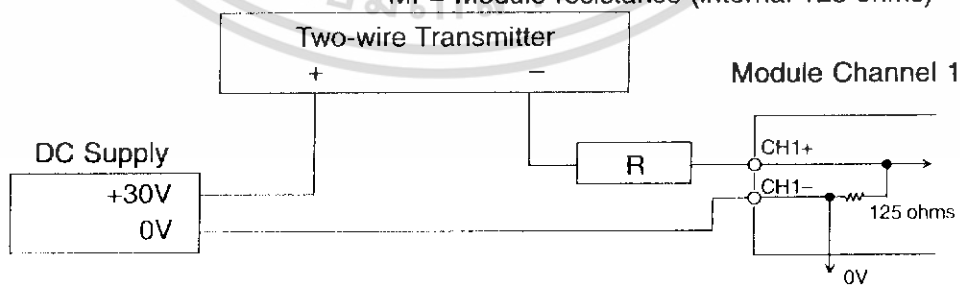
Manufacturers of transmitters and transducers specify a wide variety of power sources for their products. Follow the manufacturer's recommendations.

In some cases, manufacturers specify a minimum loop or load resistance that must be used with the transmitter. The F0-04AD-1 provides 125 ohm resistance for each channel. If your transmitter requires a load resistance below 125 ohms, you do not have to make any changes. However, if your transmitter requires a load resistance higher than 125 ohms, you need to add a resistor in series with the module.

Consider the following example for a transmitter being operated from a 30 VDC supply with a recommended load resistance of 750 ohms. Since the module has a 125 ohm resistor, you need to add an additional resistor.

$R = Tr - Mr$
$R = 750 - 125$
$R \geq 625$

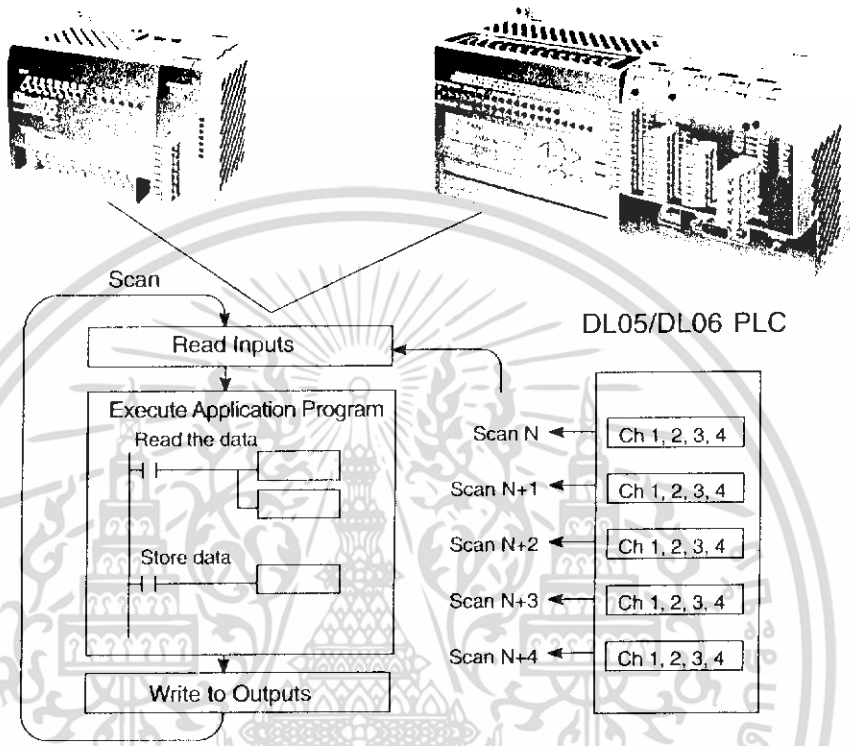
R = resistor to add
Tr = Transmitter Requirement
Mr = Module resistance (internal 125 ohms)



Module Operation

Channel Scanning Sequence

The DL05 and DL06 will read all four channels of input data during each scan. Each CPU supports special V-memory locations that are used to manage the data transfer. This is discussed in more detail beginning in the section on "Special V–memory Locations".



Analog Module Updates

Even though the channel updates to the CPUs are synchronous with the CPU scan, the module asynchronously monitors the analog transmitter signals and converts each signal into a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

The module takes approximately 25 milliseconds to sense 95% of the change in the analog signal. For the vast majority of applications, the process changes are much slower than these updates.

NOTE: If you are comparing other manufacturers' update times (step responses) with ours, please be aware that some manufacturers refer to the time it takes to convert the analog signal to a digital value. Our analog to digital conversion takes only a few microseconds. It is the settling time of the filter that is critical in determining the full update time. Our update time specification includes the filter settling time.

Special V-memory Locations

Formatting the Module Data

The DL05 and DL06 PLCs have special V-memory locations assigned to respective option slots. These V-memory locations allow you to:

- specify the data format (binary or BCD)
- specify the number of channels to scan (4 channels for the F0–04AD–1)
- specify the V-memory locations to store the input data

DL05 Data Formatting

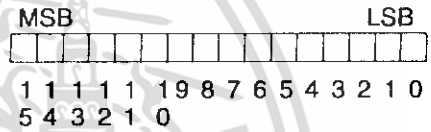
The table below shows the special V-memory locations used by the DL05 PLC for the F0–04AD–1.

Analog Input Module DL05 V-memory Locations	
Data Type and Number of Channels	V7700
Storage Pointer	V7701

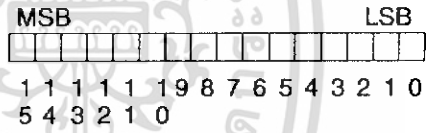
Structure of V7700

Special V–memory location 7700 identifies that a F0–04AD–1 module is installed in the DL05 option slot and the data type to be either binary or BCD.

Loading a constant of 400 into V7700 identifies a 4 channel analog input module is installed in the DL05 option slot, and reads the input data values as BCD numbers.



Loading a constant of 8400 into V7700 identifies a 4 channel analog input module is installed in the DL05 option slot, and reads the input data values as binary numbers.



Structure of V7701

V7701 is a system V–memory location used as a pointer to a user V-memory location where the analog input data is stored. The V–memory location loaded into V7701 is an octal number identifying the first user V-memory location for reading the analog input data. This V–memory location is user selectable. For example, loading O2000 causes the pointer to write Ch 1's data value to V2000, Ch 2's data value to V2001, Ch 3's data value to V2002, and Ch 4's data value to V2003.

You will find an example program that loads appropriate values to V7700 and V7701 on page 3–10.

DL06 Data Formatting

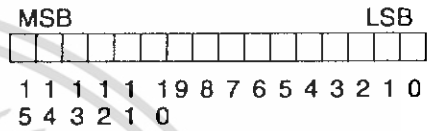
Special V–memory locations are assigned to the four option slots of the DL06 PLC. The table below shows these V-memory locations which can be used to setup the F0–04AD–1.

Analog Input Module DL06 V-memory Locations				
Slot No.	1	2	3	4
Data Type and Number of Channels	V700	V710	V720	V730
Storage Pointer	V701	V711	V721	V731

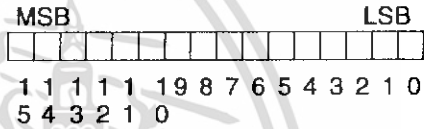
Setup Data Type and Number of Channels

V–memory locations 700, 710, 720 and 730 are used to set the data format to be read in either binary or BCD, and to set the number of channels that will be active.

For example, the F0–04AD–1 is installed in slot 1. Loading a constant of 400 into V700 sets 4 channels active, and the input data value is read as a BCD number



With the F0–4AD–1 in slot 1, loading a constant of 8400 into V700 sets 4 channels active, and the input data value is read as a binary number.



Storage Pointer Setup

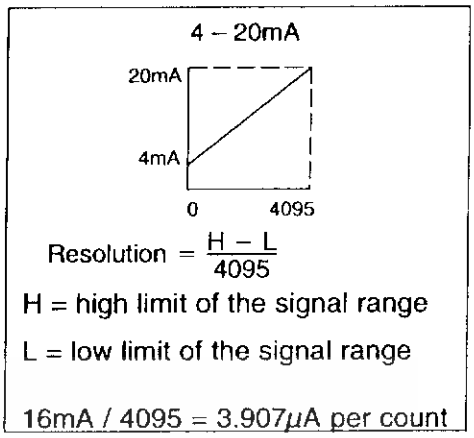
V–memory locations 701, 711, 721 and 731 are special locations used as storage pointers. A V–memory address is loaded into this location as an octal number identifying the first user V-memory location for the analog input data. This V–memory location is user selectable. For example, loading O2000 causes the pointer to write Ch 1's data value to V2000, Ch 2's data value to V2001, Ch 3's data value to V2002, and Ch 4's data value to V2003.

You will find an example program that loads appropriate values to V700 and V701 beginning on page 3–11.

Module Resolution

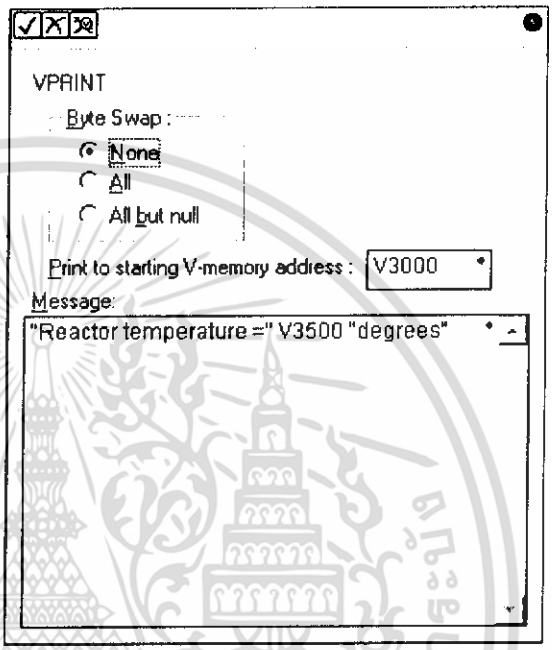
Since the module has 12-bit resolution, the analog signal is converted into 4096 counts ranging from 0 – 4095 (2^{12}). For example, a 4mA signal would be 0 and a 20mA signal would be 4095. This is equivalent to a binary value of 0000 0000 0000 to 1111 1111 1111, or 000 to FFF hexadecimal. The diagram to the right shows how this relates to the signal range.

Each count can also be expressed in terms of the signal level by using the equation shown.



ASCII Print to V–memory (VPRINT)

The ASCII Print to V–memory instruction will write a specified ASCII string into a series of V–memory registers. Other features include Byte Swap, options to suppress or convert leading zeros or spaces, and _Date and _Time options for U.S., European, and Asian date formats and 12 or 24 hour time formats.



- Byte Swap: swaps the high–byte and low–byte within each V–memory register the ASCII string is printed to. See the SWAPB instruction for details.
- Print to Starting V–memory Address: specifies the beginning of a series of V–memory addresses where the ASCII string will be placed by the VPRINT instruction.
- Starting V–memory Address: the first V–memory register of the series of registers specified will contain the ASCII string's length in bytes.
- Starting V–memory Address +1: the 2nd and subsequent registers will contain the ASCII string printed to V–memory.

Parameter	DL06 Range
Print to Starting V–memory Address	All V–memory

VPRINT Time / Date Stamping– the codes in the table below can be used in the VPRINT ASCII string message to "print to V–memory" the current time and/or date.

#	Character code	Date / Time Stamp Options
1	_Date:us	American standard (month/day/2 digit year)
2	_Date:e European standard	(day/month/2 digit year)
3	_Date:a Asian standard	(2 digit year/month/day)
4	_Time:12	standard 12 hour clock (0–12 hour:min am/pm)
5	_Time:24	standard 24 hour clock (0–12 hour:min am/pm)

VPRINT V-memory element – the following modifiers can be used in the VPRINT ASCII string message to "print to V-memory" register contents in integer format or real format. Use V-memory number or V-memory number with ":" and data type. The data types are shown in the table below. The Character code must be capital letters.

NOTE: There must be a space entered before and after the V-memory address to separate it from the text string. Failure to do this will result in an error code 499.

1	none	16-bit binary (decimal number)
2	:B	4 digit BCD
3	:D	32-bit binary (decimal number)
4	:D B	8 digit BCD
5	:R	Floating point number (real number)
6	:E	Floating point number (real number with exponent)

Examples:

V2000 Print binary data in V2000 for decimal number

V2000 : B Print BCD data in V2000

V2000 : D Print binary number in V2000 and V2001 for decimal number

V2000 : D B Print BCD data in V2000 and V2001

V2000 : R Print floating point number in V2000/V2001 as real number

V2000 : E Print floating point number in V2000/V2001 as real number with exponent

The following modifiers can be added to any of the modifies above to suppress or convert leading zeros or spaces. The character code must be capital letters.

1	S	Suppresses leading spaces
2	C0	Converts leading spaces to zeros
3	0	Suppresses leading zeros

Example with V2000 = 0018 (binary format)

	1	2	3	4
V2000	0	0	1	8
V2000:B	0	0	1	2
V2000:B0	1	2		

Example with V2000 = sp sp18 (binary format) where sp = space

	1	2	3	4
V2000	sp	sp	1	8
V2000:B	sp	sp	1	2
V2000:BS	1	2		
V2000:BC0	0	0	1	2

VPRINT V-memory text element – the following is used for "printing to V-memory" text stored in registers. Use the % followed by the number of characters after V-memory number for representing the text. If you assign "0" as the number of characters, the function will read the character count from the first location. Then it will start at the next V-memory location and read that number of ASCII codes for the text from memory.

Example:

V2000 %16 16 characters in V2000 to V2007 are printed.

V2000 %0 The characters in V2001 to Vxxxx (determined by the number in V2000) will be printed.

VPRINT Bit element – the following is used for "printing to V-memory" the state of the designated bit in V-memory or a control relay bit. The bit element can be assigned by the designating point (.) and bit number preceded by the V-memory number or relay number. The output type is described as shown in the table below.

1	none	Print 1 for an ON state, and 0 for an OFF state
2	BOOL	Print "TRUE" for an ON state, and "FALSE" for an OFF state
3	ONOFF	Print "ON" for an ON state, and "OFF" for an OFF state

Example:

V2000 . 15 Prints the status of bit 15 in V2000 in 1/0 format

C100 Prints the status of C100 in 1/0 format

C100 : BOOL Prints the status of C100 in TRUE/FALSE format

C100 : ON/OFF Prints the status of C100 in ON/OFF format

V2000.15 : BOOL Prints the status of bit 15 in V2000 in TRUE/FALSE format

The maximum numbers of characters you can VPRINT is 128. The number of characters required for each element, regardless of whether the :S, :C0 or :0 modifiers are used, is listed in the table below.

Text, 1 character	1
16 bit binary	6
32 bit binary	11
4 digit BCD	4
8 digit BCD	8
Floating point (real number)	3
Floating point (real with exponent)	13
V-memory/text	2
Bit (1/0 format)	1
Bit (TRUE/FALSE format)	5
Bit (ON/OFF format)	3

Text element – the following is used for "printing to V-memory" character strings. The character strings are defined as the character (more than 0) ranged by the double quotation marks. Two hex numbers preceded by the dollar sign means an 8-bit ASCII character code. Also, two characters preceded by the dollar sign is interpreted according to the following table:

1	$$	Dollar sign ($)
2	$"	Double quotation (")
3	$L or $l	Line feed (LF)
4	$N or $n	Carriage return line feed (CRLF)
5	$P or $p	Form feed
6	$R or $r	Carriage return (CR)
7	$T or $t	Tab

The following examples show various syntax conventions and the length of the output to the printer:

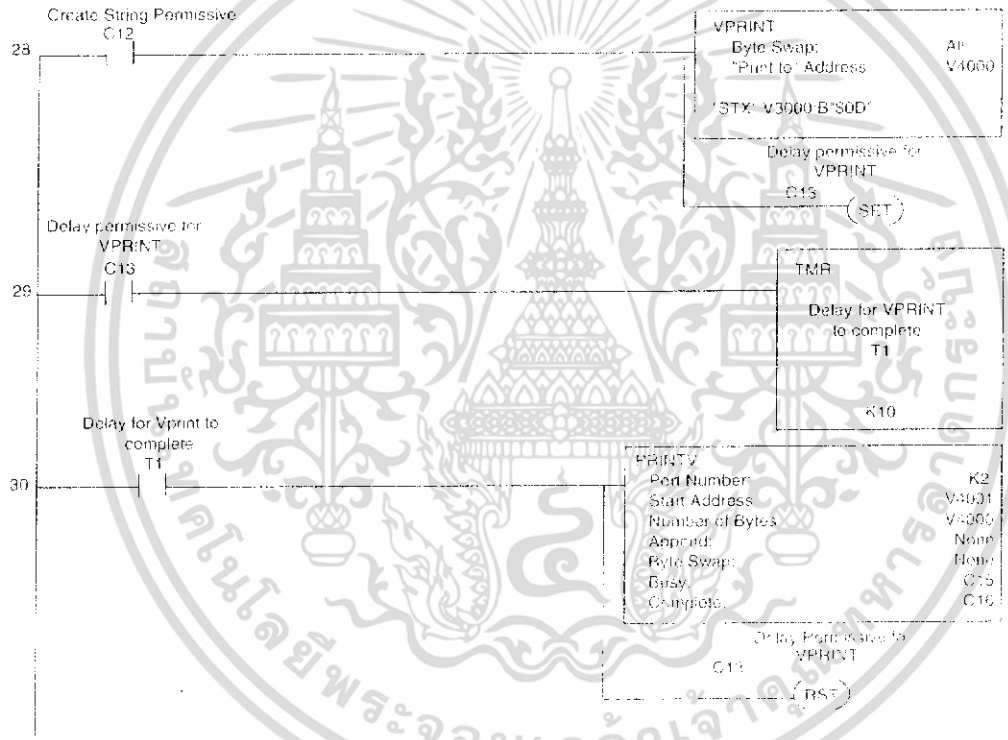
""	Length 0 without character
"A"	Length 1 with character A
" "	Length 1 with blank
"$""	Length 1 with double quotation mark
"RL"	Length 2 with one CR and one LF
"$0D$0A"	Length 2 with one CR and one LF
"$$"	Length 1 with one $ mark

In printing an ordinary line of text, you will need to include double quotation marks before and after the text string. Error code 499 will occur in the CPU when the print instruction contains invalid text or no quotations. It is important to test your VPRINT instruction data during the application development.

5

VPRINT Example Combined with PRINTV Instruction

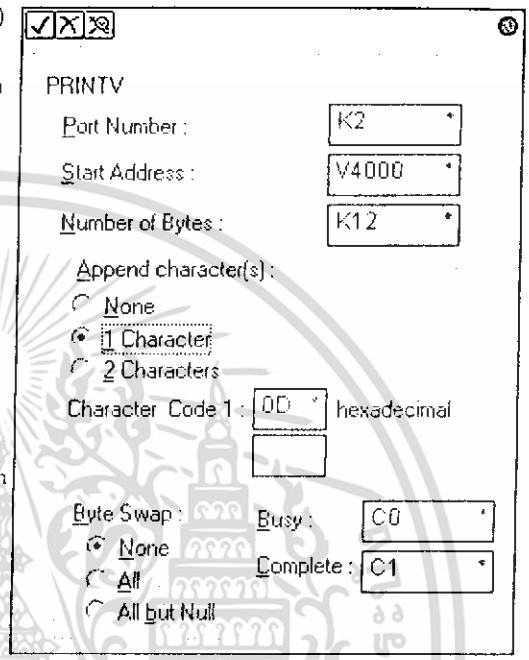
The VPRINT instruction is used to create a string in V–memory. The PRINTV is used to print the string out of port 2.



ASCII Print from V–memory (PRINTV)

The ASCII Print from V–memory instruction will send an ASCII string out of the designated communications port from a specified series of V–memory registers for a specified length in number of bytes. Other features include user specified Append Characters to be placed after the desired data string for devices that require specific termination character(s), Byte Swap options, and user specified flags for Busy and Complete.

- Port Number: must be DL06 port 2 (K2)
- Start Address: specifies the beginning of series of V–memory registers that contain the ASCII string to print
- Number of Bytes: specifies the length of the string to print
- Append Characters: specifies ASCII characters to be added to the end of the string for devices that require specific termination characters
- Byte Swap: swaps the high–byte and low–byte within each V–memory register of the string while printing. See the SWAPB instruction for details.
- Busy Bit: will be ON while the instruction is printing ASCII data
- Complete Bit: will be set once the ASCII data has been printed and reset when the PRINTV instruction permissive bits are disabled.



See the previous page for an example using the PRINTV instruction.

Port Number	port 2 (K2)
Start Address	All V–memory
Number of Bytes	All V- memory or k1–128
Bits: Busy, Complete	C0–3777