

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องวัดพลังงานไฟฟ้าแบบดิจิทัล

Digital Energy Meter



นายภาณุพงษ์ มุสิกโน
นางสาวสิโรชนี วรรณโทย

รพ.
๓๔๓๒๑
๒๕๔๙

เลขหมู่.....
เลขทะเบียน..... 72970
วัน,เดือน,ปี..... 26 ส.ย. 2550

b. 11๖๖๕๖๑๓
i.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา ๒๕๔๙

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องวัดพลังงานไฟฟ้าแบบดิจิทัล

Digital Energy Meter



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2549

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องวัดพลังงานไฟฟ้าแบบดิจิทัล

Digital Energy Meter

ผู้จัดทำ

1. นายกาญจน์ มุลจิน เลขประจำตัว 47015689
2. นางสาวสิโรจินี วรรณโทัย เลขประจำตัว 47015695



อาจารย์ที่ปรึกษา

(อ.คณัฐ ตั้งติสานนท์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ เครื่องวัดพลังงานไฟฟ้าแบบดิจิทัล

Digital Energy Meter

จัดทำโดย นายภาณุพงษ์ มุลจิโน รหัสประจำตัว 47015689

นางสาวสิโรชนี วรรณทัย รหัสประจำตัว 47015695

อาจารย์ที่ปรึกษา อ.คณัฐ ดังคิสานนท์

บทคัดย่อ

ปัจจุบันนี้การใช้เครื่องวัดพลังงานไฟฟ้ามีทั้งแบบที่ใช้หลักการเหนี่ยวนำไฟฟ้า และแบบที่ใช้ตัว Controller ในการควบคุม (Digital Energy Meter) โดย Digital Energy Meter นั้นยังไม่เป็นที่รู้จักกันมากนัก ทั้งที่การใช้งานสามารถอำนวยความสะดวกได้อย่างมากมาย เพราะสามารถที่จะคำนวณค่าพลังงานไฟฟ้าออกมาแล้วส่งข้อมูลไปยังศูนย์กลาง ซึ่งมีการทำงานแบบอัตโนมัติ

ด้วยเหตุนี้ทำให้ทางผู้จัดทำสนใจที่จะศึกษาเนื้อหาการทำงานและหลักการของตัว Digital Energy Meter เพื่อนำไปใช้งานจริงและทำการพัฒนาต่อไป โดยจะทำการศึกษาถึงพื้นฐานโครงสร้างของการทำงาน ซึ่งสามารถแบ่งออกเป็น 3 ส่วน คือ ส่วนแรกเป็นส่วนของการวัดค่าพลังงานไฟฟ้า (โดยในที่นี้จะใช้ ADE Chip ในการวัดค่าพลังงานไฟฟ้า) ส่วนที่สองคือ ส่วนของ Controller ทำหน้าที่ควบคุมการทำงานของ Energy Meter เช่น แสดงผล, เก็บข้อมูลต่างๆ ลง EEPROM และทำการส่งข้อมูลไปยัง Ethernet Module ส่วนสุดท้ายคือส่วนของ Ethernet Interface ซึ่งจะทำหน้าที่แปลงข้อมูลที่ได้รับจาก Controller ให้อยู่ในรูปแบบของ TCP Datagram เพื่อส่งไปยัง ศูนย์กลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title	Energy Meter Digital
Name	Mr.Panupong Moonjino Ms.Sirochinee Wannothai
Project Advisor	Mr.Kanut Tungtisanon

Abstract

A device use for measuring electrical energy; namely the digital energy meter has study and built. The circuit is based on the use of ADE7755 energy metering chip and P89V51RD2 microcontroller. The ADE7755 has two channel 16-bit delta-sigma ADC (Analog to Digital Circuit). The voltage and current inputs are supply to the ADE7755 by using voltage divider and shunt resistor respectively. The real power is integrated with time producing pulse output. The number of pulse , fed to P89V51RD2 , depends on the electrical energy drawn by a load. P89V51RD2 read of kWh are display on 20x1 line LCD and save at the specified duration. Digital Energy Meter can send data to server with Ethernet Module. Ethernet Module is function to change data from RS 232 to TCP Protocol. This data uses to calculate in application.

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้คงไม่อาจเสร็จลุล่วงได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และความ
ร่วมมือจากหลาย ๆ บุคคลด้วยกัน บุคคลแรกที่ต้องกล่าวถึงเพราะเป็นบุคคลสำคัญที่ทำให้ปริญญา
นิพนธ์นี้เสร็จลงได้ก็คือ อ.คณัฐ ดังติสานนท์ อาจารย์ที่ปรึกษาปริญญาานิพนธ์ที่ให้แนะนำต่อการทำ
โครงการนี้อยู่เสมอ ขอขอบคุณนายชัยศักดิ์ วัฒนวิเชียร ที่คอยให้คำแนะนำและช่วยเหลือในเรื่องๆ
ขอขอบคุณเพื่อนๆ ทุกคนที่ให้ขี้อุปกรณ์และคอยกำลังใจในการทำงาน

สุดท้ายนี้ ต้องขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา อัน
เป็นที่เคารพรักรยิ่งของข้าพเจ้า ซึ่งได้เลี้ยงดูผู้จัดทำมาเป็นอย่างดีพร้อมทั้งให้โอกาสและคอยสนับสนุน
ในเรื่องการศึกษาอย่างเต็มที่ และยังให้คอยกำลังใจ ในทุกๆ ด้าน

นายภาณุพงษ์ มุลจิโน
นางสาวสิโรชนี วรรณโทย

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญภาพ.....	IX

บทที่ 1 บทนำ

1.1 ความสำคัญและที่มาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของการดำเนินงาน	2
1.4 วิธีการดำเนินการ	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2

บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 พลังงานไฟฟ้า	3
2.2 ความหมายของกำลังไฟฟ้าและเพาเวอร์แฟกเตอร์	3
2.3 เครื่องวัดพลังงานไฟฟ้า	5
2.3.1 วัดด้วยอาร์มิตอร์เฟสเดียวแบบอาศัยการเหนี่ยวนำไฟฟ้า	6
2.4 ไมโครคอนโทรลเลอร์ P89V51RD2	9
2.4.1 คุณสมบัติทางเทคนิคที่สำคัญของ P89V51RD2	9
2.4.2 หน้าที่แต่ละขาของไมโครคอนโทรลเลอร์.....	9
2.5 แผงแสดงผลแบบ LCD	11
2.5.1 ชุดคำสั่งของ LCD Module	11
2.6 ระบบบัสแบบ I ² C และการเชื่อมต่อ	14
2.6.1 คุณสมบัติโดยทั่วไปของบัส I ² C	15
2.6.2 หลักการของบัส I ² C	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา แฉV ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.6.3	สถานะที่เกิดขึ้นบนบัส I ² C	16
2.6.4	การอ้างถึงแบบ 7 บิต (7 -bit Addressing)	17
2.7	ไอซีฐานเวลาจริง DS1307 (Real – Time Clock)	18
2.7.1	หน้าที่แต่ละขาของไอซี DS1307	18
2.7.2	หน่วยความจำ RAM ภายในไอซี DS1307	19
2.7.3	โหมดการทำงานของ DS1307	19
2.8	หลักการของ Serial EEPROM	20
2.8.1	คุณสมบัติ และลักษณะของ Serial EEPROM	20
2.8.2	ลักษณะการควบคุมการทำงาน 24C16	21
2.9	หลักการการทำงานของ ADE 7755	22
2.9.1	หลักการการทำงานของส่วนวัดค่าพลังงานไฟฟ้า (ADE 7755)	22
2.9.2	ขาหน้าที่ของแต่ละขาภายใน IC ADE 7755	23
2.10	Networking Concepts and Protocols	24
2.10.1	Physical Networks	24
2.10.2	Network Protocols	25
2.11	Sockets	25
2.11.1	Berkeley Sockets	26
2.11.2	Working with Sockets in .Net	27
2.11.3	Working with Sockets	27
2.12	Stream Sockets (Connection Oriented Socket)	28
2.12.1	ข้อควรระวังในการใช้งาน Stream Socket	30
2.13	Multithreading	35
2.13.1	Using Thread	35
2.13.2	Using Thread Pools	36
2.14	การออกแบบฐานข้อมูล	36
2.14.1	ระบบฐานข้อมูล (Database System)	36
2.14.2	การออกแบบฐานข้อมูลด้วยอี-อาร์โมเดล	37
2.14.3	โครงสร้างของภาษาเอสคิวแอล (SQL)	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.14.4 ลักษณะการใช้งานของภาษา SQL	38
2.14.5 การบันทึกข้อมูล ,การปรับปรุงข้อมูล และการลบข้อมูล	39
2.14.6 การเรียกค้นข้อมูล (SELECT)	41

บทที่ 3 การดำเนินงานวิจัย

3.1 โครงสร้างโดยรวมของระบบ	42
3.2 โครงสร้าง Digital Energy Meter.....	42
3.3 วงจรส่วนวัดพลังงานของ Digital Energy Meter.....	43
3.4 วงจรส่วน Controller ของ Digital Energy Meter.....	44
3.5 Flowchart ส่วน Digital Energy Meter.....	47
3.5.1 Flowchart โปรแกรมหลักทำหน้าที่ควบคุมการทำงานของ Digital Energy Meter....	47
3.5.2 Flowchart โปรแกรมย่อย LCD_Data_unit()	48
3.5.3 Flowchart โปรแกรมย่อย LCD_init()	49
3.5.4 Flowchart โปรแกรมย่อย i2c init ()	49
3.5.5 Flowchart โปรแกรมย่อย int0 init()	49
3.5.6 Flowchart โปรแกรมย่อย init_serial()	50
3.5.7 Flowchart โปรแกรมย่อยอ่าน EEPROM	50
3.5.8 Flowchart โปรแกรมย่อยอ่าน EEPROM	51
3.6 Flowchart ส่วน Application	52
3.7 ER – Model	53
3.8 Relationship Model	54

บทที่ 4 การทดลองและผลการทดลอง

4.1 ส่วนวงจร Digital Energy Meter.....	55
4.1.1 Digital Energy Meter	55
4.1.2 ผลการทดลองที่วัดใน 1 วัน ที่โหลดสม่ำเสมอ	56
4.1.3 ผลการทดลองที่วัดใน 1 วัน ที่โหลดไม่สม่ำเสมอ	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

4.2 ส่วน Application	59
4.2.1 หน้าต่างหลักของตัว Application	59
4.2.2 หน้าต่างของส่วน Detail	60
4.2.3 หน้าต่างของ Config	63
บทที่ 5 บทวิจารณ์และสรุป	
5.1 บทสรุป	64
5.2 วิจารณ์สิ่งที่ได้จากโครงการ	64
5.3 ปัญหาอุปสรรคและแนวทางแก้ไข	64
5.4 แนวทางการพัฒนาต่อ	65
ภาคผนวก	
บรรณานุกรม	

สารบัญตาราง

หน้า

ตารางที่ 2.1 The Thread Class Method	36
ตารางที่ 2.2 The ThreadPool Static Method	37
ตารางที่ 4.1 ผลการทดลองที่โหลดสม่ำเสมอโดยใช้หลอดไฟ 100 W	56
ตารางที่ 4.2 ผลการทดลองที่โหลดสม่ำเสมอโดยใช้โทรทัศน์ 50 วัตต์	57
ตารางที่ 4.3 ผลการทดลองที่โหลดไม่สม่ำเสมอ	58



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา VIII ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1	กราฟความสัมพันธ์ของกำลังไฟฟ้าและเวลา	3
รูปที่ 2.2	กำลังไฟฟ้ากระแสสลับในโพลชนิดความต้านทาน	5
รูปที่ 2.3	กำลังไฟฟ้ากระแสสลับในโพลต่างๆ ไป	5
รูปที่ 2.4	รูปแสดงส่วนประกอบของวัตต์อาร์มิเตอร์	6
รูปที่ 2.5	แสดงส่วนประกอบต่างๆ และชุดบอกรปริมาณพลังงานไฟฟ้า	7
รูปที่ 2.6	แสดงการต่อกิโลวัตต์อาร์มิเตอร์ร่วมกับอุปกรณ์อื่น ๆ	8
รูปที่ 2.7	แสดงวงจรเอาต์พุตของอุปกรณ์ในระบบบัส I ² C	16
รูปที่ 2.8	ไดอะแกรมเวลาแสดงสถานะต่างๆ ในบัส I ² C	18
รูปที่ 2.9	แบบของข้อมูลกำหนดแอดเดรสที่ใช้ในการอ้างถึงแบบ 7 บิต	18
รูปที่ 2.10	ไดอะแกรมโครงสร้างภายในชิพ DS1307	19
รูปที่ 2.11	แผนผังรีจิสเตอร์ตำแหน่งต่างๆ ในการจัดการด้านฐานข้อมูลเวลา	20
รูปที่ 2.12	รูปแบบของข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการเขียนข้อมูล	21
รูปที่ 2.13	รูปแบบของข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการอ่านข้อมูล	21
รูปที่ 2.14	โครงสร้างภายในและการจัดขาของชิพ 24C16	22
รูปที่ 2.15	แสดงถึงการใช้อนุบิตเลือกตำแหน่ง	22
รูปที่ 2.16	Functional Block Diagram ของ ADE7755	23
รูปที่ 2.17	การเชื่อมต่อส่วนรับสัญญาณอินพุต V1 กับ ADE 7755	23
รูปที่ 2.18	แสดงการเชื่อมต่อส่วนรับสัญญาณอินพุต V2 ADE 7755	24
รูปที่ 2.19	Test Circuit for Perfomance Curve	24
รูปที่ 2.20	OSI-TCP/IP Protocol Stack	26
รูปที่ 2.21	Descriptor Table and Data structure for a socket	27
รูปที่ 2.22	Diagram ขั้นตอนการสร้าง Active Sockets	28
รูปที่ 2.23	Diagram ขั้นตอนการสร้าง Passive Sockets	29
รูปที่ 2.24	แบบจำลอง TCP Buffer	32
รูปที่ 3.1	โครงสร้างโดยรวมของระบบ	43
รูปที่ 3.2	โครงสร้างของ Digital Energy Meter.....	44
รูปที่ 3.3	วงจรส่วนวัดพลังงานของ Digital Energy Meter.....	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา IIX ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 3.4 วงจรส่วน Controller ของ Digital Energy Meter.....	46
รูปที่ 3.5 วงจร Digital Energy Meter.....	47
รูปที่ 4.1 Digital Energy Meter.....	56
รูปที่ 4.2 หน้าต่างหลักของตัว Application	57
รูปที่ 4.3 หน้าต่างของส่วน Detail	58
รูปที่ 4.4 หน้าต่างของส่วน Print	59
รูปที่ 4.5 หน้าต่างของส่วน Print Dialog	60
รูปที่ 4.6 หน้าต่างของส่วน Print Preview Dialog	60
รูปที่ 4.7 หน้าต่างของส่วน Config	61



บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของโครงการ

ปัจจุบันนี้ไฟฟ้าเป็นสิ่งที่สำคัญ แทบทุกครัวเรือนที่การไฟฟ้าสามารถให้บริการได้ต่างก็ต้องการใช้พลังงานไฟฟ้า แต่พลังงานไฟฟ้ามีกระบวนการผลิตที่สำคัญมาจากทรัพยากรธรรมชาติ เช่น ถ่านหิน ซึ่งใช้แล้วหมดสิ้นไป จึงต้องมีการประหยัดการใช้พลังงานไฟฟ้า โดยการประหยัดพลังงานไฟฟ้านี้เราจำเป็นต้องทราบค่าพลังงานที่ใช้ไป แต่การวัดค่าพลังงานไฟฟ้านี้เราไม่สามารถทราบถึงพลังงานไฟฟ้าได้จากประสาทสัมผัสโดยตรง จึงต้องอาศัยปรากฏการณ์ทางกายภาพ เพื่อเปลี่ยนปริมาณพลังงานไฟฟ้าเหล่านั้นให้เป็นปริมาณที่สามารถรับรู้ได้จากประสาทสัมผัส โดยมีวิธีวัดพลังงานไฟฟ้าเหล่านั้นด้วยวิธีต่างๆ เช่น กิโลวัตต์อวาร์มิเตอร์ และเครื่องวัดพลังงานไฟฟ้าแบบดิจิตอล

ปัจจุบันนี้การใช้เครื่องวัดพลังงานไฟฟ้ามีทั้งแบบที่ใช้หลักการเหนี่ยวนำไฟฟ้า และแบบที่ใช้ตัว Controller ในการควบคุม (Digital Energy Meter) โดย Digital Energy Meter นั้นยังไม่เป็นที่รู้จักกันมากนัก ทั้งที่การใช้งานสามารถอำนวยความสะดวกได้อย่างมากมาย เพราะสามารถที่จะคำนวณค่าพลังงานไฟฟ้าออกมาแล้วส่งข้อมูลไปยังศูนย์กลาง ซึ่งมีการทำงานแบบอัตโนมัติ ด้วยเหตุนี้ทำให้ทางผู้จัดทำสนใจที่จะศึกษาเนื้อหาการทำงานและหลักการของตัว Digital Energy Meter เพื่อนำไปใช้งานจริงและทำการพัฒนาต่อไป

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อเรียนรู้หลักการในการออกแบบให้ใช้งานได้ง่ายและใช้งานได้จริง
- 1.2.2 เพื่อเรียนรู้ลักษณะการทำงานที่เป็นระบบและรู้จักวิเคราะห์ปัญหา
- 1.2.3 เพื่อศึกษาลักษณะการทำงานของเครื่องวัดพลังงานไฟฟ้าแบบดิจิตอล
- 1.2.4 เพื่อศึกษาการใช้ระบบไมโครคอนโทรลเลอร์
- 1.2.5 เพื่อนำหลักการที่ศึกษาได้มาออกแบบวงจร

1.3 ขอบเขตของการดำเนินงาน

- 1.3.1 ศึกษาลักษณะการทำงานของเครื่องวัดพลังงานไฟฟ้าแบบดิจิทัล
- 1.3.2 ศึกษาการใช้ระบบไมโครคอนโทรลเลอร์
- 1.3.3 ออกแบบวงจร Digital Energy Meter
- 1.3.4 Digital Energy Meter สามารถส่งข้อมูลไปยังคอมพิวเตอร์ได้ด้วยโปรโตคอล TCP/IP
- 1.3.5 ศึกษาและจัดหาอุปกรณ์เพื่อนำมาใช้งานจริง

1.4 วิธีการดำเนินการ

- 1.4.1 ศึกษาโครงสร้างทั่วไปและหน้าที่การทำงานของแต่ละส่วนของ Energy Meter
- 1.4.2 ศึกษาการทำงานและจัดหาอุปกรณ์ตัว IC Energy Meter
- 1.4.3 ศึกษาการทำงานและจัดหาอุปกรณ์ตัว Controller
- 1.4.4 ศึกษาการทำงานและจัดหาอุปกรณ์ Display และตัว Memory
- 1.4.5 วิเคราะห์และออกแบบวงจร โครงสร้างทั้งหมดของวงจร Digital Energy Meter
- 1.4.6 ทดลองประกอบวงจรที่ได้ออกแบบไว้ และเขียนโปรแกรมในการควบคุมการทำงาน
- 1.4.7 ศึกษาการทำงานและจัดหาอุปกรณ์ส่วนส่งผ่านข้อมูล
- 1.4.8 เขียนโปรแกรมในการเชื่อมต่อ Digital Energy Meter กับคอมพิวเตอร์ได้ด้วยโปรโตคอล TCP/IP
- 1.4.9 วิเคราะห์ผลที่ได้ และแก้ไขส่วนที่ผิดพลาดเพื่อให้สามารถใช้งานได้ถูกต้อง

1.5 ประโยชน์ที่คาดว่าจะได้รับ

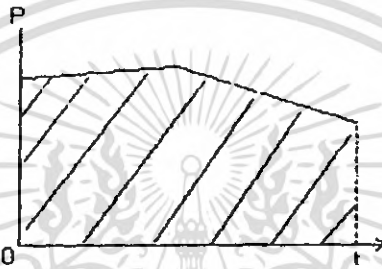
- 1.5.1 เข้าใจระบบการทำงานของตัวอุปกรณ์แต่ละส่วน
- 1.5.2 รู้จักประยุกต์ใช้งานระบบไมโครคอนโทรลเลอร์

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 พลังงานไฟฟ้า

พลังงานไฟฟ้านั้นจะเป็นผลคูณของกำลังไฟฟ้ากับเวลาจึงทำให้สามารถหาพลังงานไฟฟ้าออกมาได้โดยจากกราฟความสัมพันธ์ของกำลังไฟฟ้าและเวลาดังนี้



รูปที่ 2.1 กราฟความสัมพันธ์ของกำลังไฟฟ้าและเวลา

พลังงานไฟฟ้านั้นสามารถหาได้จากสมการ

$$w = \int_0^T P dt \quad (2.1)$$

พื้นที่ใต้กราฟ = (ค่าเฉลี่ย) × (ความหนาของกราฟ)

จะได้ว่า $w = V \times I \times T$; $P = V \times I$ (2.2)

เพราะฉะนั้น $W = P \times T$ (2.3)

Energy Meter (Kwh) = Power (P) × Time (T) (2.4)

2.2 ความหมายของกำลังไฟฟ้าและเพาเวอร์แฟกเตอร์

ในวงจรไฟฟ้ากระแสตรงกำลังไฟฟ้า P ที่ใช้ไปในความต้านทานที่โหลด R จะมีค่าเป็น

$$P = VI = I^2 R = \frac{V^2}{R} \quad (\text{วัตต์}) \quad (2.5)$$

โดยที่ V เป็นแรงดันตกคร่อมโหลดและ I เป็นกระแสที่ไหลผ่านโหลด ในวงจรไฟฟ้ากระแสสลับ กำลังไฟฟ้าที่ใช้ไปในโหลด ขณะที่แรงดันตกคร่อมโหลดเท่ากับ V(t) และกระแสในโหลดเท่ากับ i(t) เป็นกำลังงานไฟฟ้าแต่ละขณะซึ่งค่าเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$p(t) = v(t)i(t) \quad (\text{วัตต์}) \quad (2.6)$$

สมมติว่าแรงดันตกคร่อมโหลดเป็นแรงดันไฟฟ้าสลับรูปไซน์ที่มีสามารถเป็น

$$v = V_m \sin \omega t \quad (\text{โวลต์}) \quad (2.7)$$

ถ้าโหลดที่มีความต้านทานเป็น R กระแสที่ไหลผ่านโหลด I จะเขียนได้เป็น

$$\frac{V_m}{R} \sin \omega t = I_m \sin \omega t \quad (\text{แอมแปร์}) \quad (2.8)$$

เพราะฉะนั้น p จะเขียนได้เป็น

$$p = V_m I_m \sin^2 \omega t \quad (\text{วัตต์}) \quad (2.9)$$

สมการข้างบนนี้ได้เขียนรูปแสดงไว้ในรูปที่ 2 เช่นเดียวกับค่าจำกัดความของค่าประสิทธิภาพ ค่าเฉลี่ยของ p ในหนึ่งช่วงเวลาคาบรอบซึ่งก็คือ กำลังไฟฟ้าโดยเฉลี่ย P จะมีค่าเป็น

$$P = \left(\frac{V_m}{\omega L} \right) = VI = I^2 R = \frac{V^2}{R} \quad (\text{วัตต์}) \quad (2.10)$$

โดยให้ V และ I เป็นค่าประสิทธิภาพของ V และ i ตามลำดับ สมการนี้รูปแบบเดียวกับ สมการกระแสตรง P ในสมการนี้เรียกว่ากำลังไฟฟ้ากระแสสลับ ถ้าโหลดเป็นตัวรีแอคทีฟ (Reactive element) เช่น อินดักเตนซ์ L เป็นต้น

$$i = \left(\frac{V_m}{\omega L} \right) \sin \left(\omega t \left(-\frac{\pi}{2} \right) \right) = I_m \sin \left(\omega t \left(-\frac{\pi}{2} \right) \right) \quad (\text{วัตต์}) \quad (2.11)$$

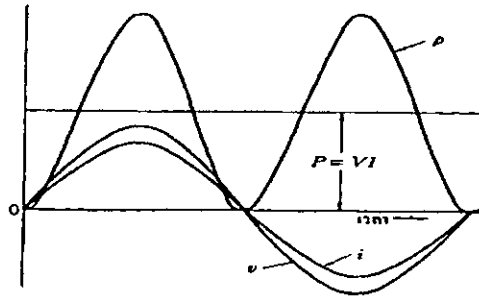
ความสัมพันธ์แสดงไว้ในรูปที่ 2.3 และกรณีนี้ กำลังเฉลี่ย P จะมีค่าเป็นดังนี้

$$P = VI \cos \varphi \quad (2.12)$$

โดยอาศัยความสัมพันธ์ที่ว่า

$$\cos \varphi = \frac{R}{\sqrt{R^2 + X^2}} \quad (2.13)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

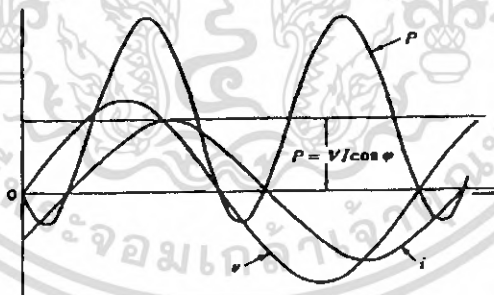


รูปที่ 2.2 กำลังไฟฟ้ากระแสสลับในโหลดชนิดความต้านทาน

เราจะหาต่อไปได้ว่า

$$P = I^2 R = \frac{V^2}{R} \quad (2.14)$$

ที่กล่าวมาข้างต้น จะเห็นได้ว่า โดยทั่วไปแล้วกำลังเฉลี่ยในกรณีของไฟฟ้ากระแสสลับแสดงโดยผลคูณแรงดันประสิทธิผลกร่อม โหลดกับกระแสประสิทธิผลที่ไหลผ่านโหลดกับค่า \cos ของมุมระหว่างเฟสของแรงดันกับกระแส กำลังเฉลี่ยนี้จะมีค่าเท่ากับกำลังที่ใช้ไปในความต้านทานของโหลด กำลังนี้จึงเรียกว่า กำลังประสิทธิผล ส่วนกำลังแต่ละขณะซึ่งมีค่าเท่ากับ $P = VI \cos \phi$ ซึ่งเป็นกำลังถ่ายเทไปมาระหว่าง ต้นกำเนิดกำลังกับรีแอกแตนซ์ของโหลดไม่มีการสิ้นเปลืองไป เรียกว่า กำลังรีแอกตีฟค่า VI เรียกว่า กำลังที่ปรากฏและค่าของ $\cos \phi$ เรียกว่า เพาเวอร์แฟคเตอร์



รูปที่ 2.3 กำลังไฟฟ้ากระแสสลับในโหลดทั่วๆ ไป

2.3 เครื่องวัดพลังงานไฟฟ้า

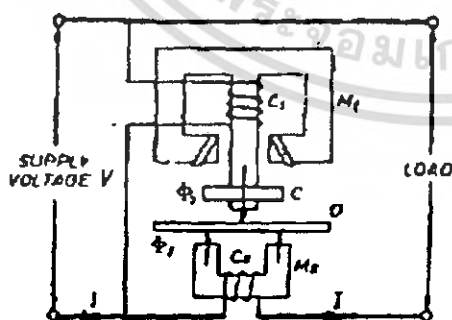
เครื่องวัดพลังงานไฟฟ้า คือ เครื่องวัดแบบรวมยอด (บวทบ) ใช้สำหรับวัดปริมาณของพลังงานไฟฟ้าที่ป้อนให้กับวงจรหนึ่งในเวลาที่กำหนดให้เวลาหนึ่ง เครื่องวัดแบบนี้ไม่สามารถที่บอกค่ากำลังของไฟฟ้าโดยตรงได้ เช่นไม่สามารถบอกได้ว่าขณะนี้กำลังไฟฟ้าหรือพลังงานไฟฟ้าที่ป้อนให้กับวงจรมีค่าเท่าไร เพราะว่าการบวทบหรือรวมยอดของกำลังไฟฟ้านี้ เป็นการรวมยอดของปริมาณที่ได้ใช้มาทั้งหมด เครื่องวัดพลังงานไฟฟ้าหลายแบบด้วยกัน เช่น วัดคาวร์มเตอร์ เป็นต้น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 วัดค่าอวอร์มิเตอร์เฟสเดียวแบบอาศัยการเหนี่ยวนำไฟฟ้า

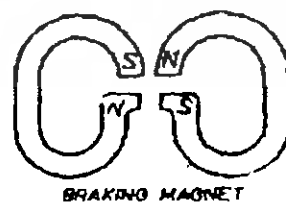
เครื่องวัดที่ทำงานด้วยการเหนี่ยวนำไฟฟ้าเป็นแบบที่ใช้กันทั่วไป ในเครื่องวัดไฟฟ้า กระแสสลับ และมักจะได้พบเห็นเสมอทั้งในบ้านในโรงงานอุตสาหกรรม มิเตอร์แบบนี้จะวัดพลังงานไฟฟ้าในหน่วยกิโลวัตต์อวอร์มิเตอร์ หรือกิโลวัตต์ชั่วโมง หลักการทำงานเหมือนกันกับเครื่องวัดกำลังไฟฟ้าที่ทำงานด้วยการเหนี่ยวนำไฟฟ้า และส่วนประกอบที่เหมือนกันคือเคอร์เรนคอยล์ และ โวลท์เดจคอยล์ ส่วนที่แตกต่างกันคือในเครื่องวัดกำลังไฟฟ้าหรือวัดคีมิตอร์จะใช้สปริง และเข็มชี้ ส่วนวัดค่าอวอร์มิเตอร์จะใช้แม่เหล็กหน่วงและชุดของเฟือง

แม่เหล็กหน่วงจะทำให้เกิดกระแสไหลวนขึ้นในงาน อลูมิเนียมที่หมุนตลอดเวลาแทนที่จะให้ งานเคลื่อนที่ไปเป็นมุมหนึ่งมุมใดและค่าที่ได้ออกมาจะเป็นค่าของปริมาณไฟฟ้า หรือปริมาณของ พลังงานไฟฟ้าที่ป้อนให้กับวงจรหนึ่งในเวลาที่กำหนด โดยการรวมยอดด้วยชุดของตัวเลขหมุนหรือชุด ของเข็มชี้

ส่วนประกอบ เครื่องวัดแบบนี้ประกอบด้วยแม่เหล็กไฟฟ้า 2 ชุด คือ แม่เหล็กไฟฟ้า M_1 และ M_2 แม่เหล็ก M_1 เรียกว่าแม่เหล็กขนาน ขดลวดที่ทำให้เกิดแม่เหล็กไฟฟ้าชุดนี้จะต่อขนานกับแรงดัน ของโหลด กระแสไฟฟ้าที่ไหลผ่านขดลวดชุดนี้จะแปรผันโดยตรงกับแรงดันของโหลด V แม่เหล็กอีก ชุดหนึ่งคือ M_2 เรียกว่าแม่เหล็กอันดับ ขดลวดที่ทำให้เกิดแม่เหล็กชุดนี้ต่ออันดับกับโหลด กระแสไฟฟ้าที่ไหลผ่านขดลวดชุดนี้จะแปรผันโดยตรงกับกระแสของโหลด แม่เหล็ก M_2 นี้จะสร้าง เส้นแรงแม่เหล็ก ϕ_2 ขึ้นมา และแม่เหล็ก M_1 จะสร้างเส้นแรงแม่เหล็ก ϕ_1 ขึ้นมา ดังนั้นจึงกล่าวได้ว่า ϕ_2 จะแปรผันโดยตรงกับแรงดันของโหลด V และมีมุมต่างเฟส 90° ส้าหลังกับแรงดันของโหลด มุม ต่างเฟสอันนี้สามารถที่จะปรับแต่งได้โดยการปรับตำแหน่งของแหวนทองแดง C ที่สวมอยู่ที่แกนกลาง ของแม่เหล็กขนาน M_1 ดังรูป



ก. แสดงการต่อวัดค่าอวอร์มิเตอร์



ข. แม่เหล็กหน่วง

รูปที่ 2.4 รูปแสดงส่วนประกอบของวัดค่าอวอร์มิเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เส้นแรงแม่เหล็ก ϕ ส่วนใหญ่จะเคลื่อนที่ผ่านช่องว่างไปยังแกนด้านข้างของแม่เหล็ก M_1 แต่มีเส้นแรงแม่เหล็กอีกจำนวนหนึ่งที่มีจำนวนน้อยที่เคลื่อนที่ผ่านแกน D และเป็นเส้นแรงแม่เหล็กส่วนที่ทำให้เกิดแรงบิดภายในขั้วแม่เหล็ก D ส่วนแรงหน่วงหรือแรงบิดควบคุมเกิดขึ้นได้จากแม่เหล็กถาวรหนึ่งคู่ดังรูปที่ 5 ข และรูปที่ 6 ซึ่งติดตั้งเป็นแนวเส้นตรงตามแนวเส้นผ่านศูนย์กลางของแกนและให้ทิศทางของอำนาจแม่เหล็กต้านกันกับอำนาจแม่เหล็กของ M_1 และ M_2 ทั้งนี้ก็เพื่อที่จะลดปฏิกิริยาระหว่างอำนาจแม่เหล็กของ M_1 และ M_2

เมื่อขอบของแกน D เคลื่อนที่ผ่านช่องว่างระหว่างขั้ว N และ S ของแม่เหล็กหน่วงก็จะทำให้เกิดกระแสไฟไหลวนเหนี่ยวนำขึ้นในแกน กระแสไฟจำนวนนี้จะทำให้เกิดแรงหน่วงขึ้น ซึ่งแรงหน่วง T_B จะมีค่าดังนี้

$$T_B \propto \frac{\phi^2 N}{r} \quad (2.15)$$

ϕ = เส้นแรงแม่เหล็กของแม่เหล็กหน่วง

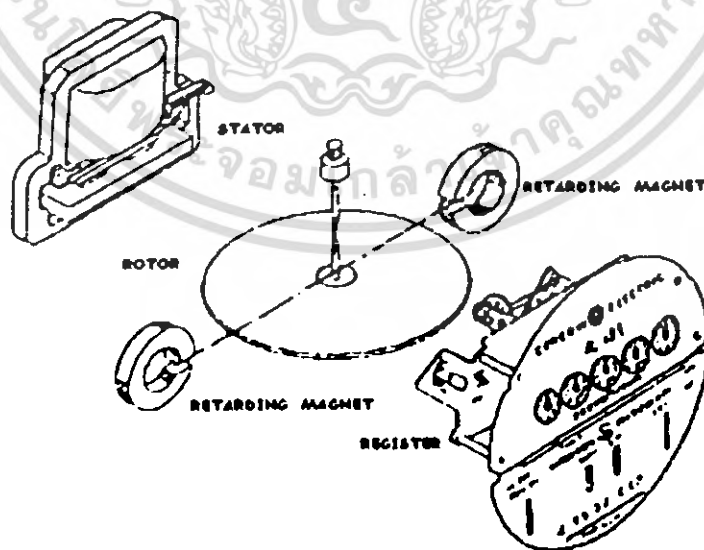
N = ความเร็วของแกนหมุน

r = ความต้านทานของวงจรกระแสไหลวนในแกน

จะได้ว่า $T_B \propto N$ (2.16)

เพราะว่า ϕ และ r จะมีค่าคงที่เสมอ

จากรูปที่ 2.5 เป็นส่วนประกอบของกิโลวัตต์อวาร์มิเตอร์แบบหนึ่งเฟสหน้าปิดเป็นแบบเข็มชี้ ส่วนรูปที่ 2.6 เป็นวงจรการต่อวัตต์อวาร์หรือกิโลวัตต์อวาร์มิเตอร์เพื่อใช้งาน



รูปที่ 2.5 แสดงส่วนประกอบต่างๆ และชุดบอกปริมาณพลังงานไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดบอกริมาณของพลังงานไฟฟ้าแบบเก่าจะประกอบด้วยชุดของเฟืองทด (เฟืองทดตัวเล็กขับเฟืองตัวใหญ่) 4 ถึง 6 ตัวนี้มีเข็มชี้ติดอยู่ทุกตัว ให้เข็มชี้เคลื่อนที่ไปบนหน้าปัทม์ที่มีสเกลบอกไว้ สเกลบนหน้าปัทม์ของเฟืองหรือเข็มชี้แต่ละตัวจะแบ่งออกไว้เป็น 10 ช่องเท่ากัน สเกลบนหน้าปัดของเฟืองตัวใหญ่จะมีค่ามากกว่าสเกลหน้าปัดของเฟืองตัวเล็ก 10 เท่า คือ เมื่อเฟืองตัวเล็กหมุนไปได้ 10 รอบ เฟืองตัวใหญ่อยู่ถัดไปจะหมุนได้ 10 รอบ โดยเพลลาของเฟืองตัวเล็กจะเป็นเพลลาเดียวกันกับเพลลาของระบบเคลื่อนของเครื่องวัดพลังงาน

ได้อธิบายมาแล้วว่า แรงบิดบ้ายเบนหรือแรงขับของเครื่องมือวัดแบบนี้จะเป็นสัดส่วนกับกำลังไฟฟ้าในวงจรไฟฟ้านั้น ๆ

$$T_d \propto VI \cos \phi \quad (2.17)$$

เมื่อความเร็วของเครื่องวัดคงที่

$$T_B = T_d \quad (2.18)$$

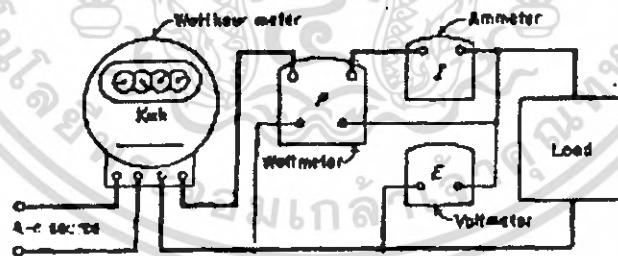
จะได้ว่า

$$T_B \propto N \quad (2.19)$$

$$N \propto VI \cos \phi \quad (2.20)$$

$$\propto \text{กำลังไฟฟ้า (W)}$$

หรือ
$$N \propto \frac{VI \cos \phi t}{3600 \times 1000} \quad (2.21)$$



รูปที่ 2.6 แสดงการต่อกิโลวัตต์อวาร์มิเตอร์ร่วมกับอุปกรณ์อื่น ๆ

$$N = K \frac{VI \cos \phi t}{3600 \times 1000} \quad (2.22)$$

$$= K \frac{W \cdot t}{3600 \times 1000} \quad \text{รอบ}$$

K = ค่าคงที่ของเครื่องวัด (จำนวนรอบ / กิโลวัตต์ชั่วโมง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น จำนวนรอบทั้งหมดของเครื่องวัดจะเป็นสัดส่วนกับพลังงานไฟฟ้าของวงจรไฟฟ้านั้นเอง วงจรสำหรับการปรับแต่งสเกลของวัดค่าอวาร์มิเตอร์จะต่อดังรูปที่ 2.6

2.4 ไมโครคอนโทรลเลอร์ P89V51RD2

2.4.1 คุณสมบัติทางเทคนิคที่สำคัญของ P89V51RD2

เป็นไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งมีหน่วยความจำภายในเป็นแบบแฟลช มีคุณสมบัติทางเทคนิคที่โดดเด่นดังนี้

- เป็นไมโครคอนโทรลเลอร์ 8 บิต ที่เข้ากันได้กับไมโครคอนโทรลเลอร์ MCS-51 พื้นฐาน
- หน่วยความจำโปรแกรมภายในตัวไมโครคอนโทรลเลอร์เป็นแบบแฟลช ลบและเขียนใหม่ได้ถึงหมื่นครั้ง ขนาดของหน่วยความจำโปรแกรมสูงถึง 64 กิโลไบต์
- หน่วยความจำข้อมูลแรมภายในมีขนาด 1 กิโลไบต์
- ความถี่สัญญาณนาฬิกาสูงสุด 40MHz
- ขาพอร์ต 8 บิต 4 พอร์ต แบบกึ่งสองทิศทาง เป็นได้ทั้งอินพุตและเอาต์พุต
- มีวงจรสื่อสารอนุกรมแบบ Full Duplex
- ไทเมอร์/คาน์เตอร์ขนาด 16 บิต 3 ตัว
- มีรีจิสเตอร์ตัวชี้ตำแหน่งข้อมูลหรือ DPTR 2 ตัว
- สามารถรองรับแหล่งกำเนิดอินเตอร์รัปต์ได้ 8 ประเภท
- กำหนดนัยสำคัญของการตอบสนองอินเตอร์รัปต์ได้ 4 ระดับ

2.4.2 หน้าที่แต่ละขาของไมโครคอนโทรลเลอร์

หน้าที่แต่ละขาของไมโครคอนโทรลเลอร์	
V _{CC}	ต่อไฟเลี้ยง +5V
GND	ต่อกราวด์
P0.0 – P0.7	- ใช้งานเป็นขาพอร์ตอินพุตเอาต์พุต ถ้าต้องการกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุต สามารถทำได้โดยการเขียนข้อมูล “1” ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ทำให้มีสถานะลอย (float) ค่าอินพุตอิมพีแดนซ์สูงสามารถใช้งานเป็นขาพอร์ตอินพุตได้ - ใช้ในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำภายนอก(A0-A7) และขาข้อมูล (D0-D7) โดยจะใช้การมัลติเพล็กซ์เข้าช่วย เพื่อสลับการทำงานให้เป็นที่ทั้งขาติดต่อกับแอดเดรสและขาข้อมูลในการติดต่อกับหน่วยความจำภายนอก
P1.0 – P1.7	- ใช้งานเป็นขาพอร์ตอินพุตเอาต์พุตสำหรับใช้งานทั่วไป เฉพาะขา P1.5 ถึง P1.7 สามารถขับกระแสได้สูง 16mA ต่อขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่แต่ละขาของไมโครคอนโทรลเลอร์	
	T2 (P1.0) ขาอินพุตสำหรับนับค่าของไทเมอร์ 2 ขาเอาต์พุตสัญญาณนาฬิกาโปรแกรมได้
	T2EX(P1.1) ขาอินพุตสำหรับควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 2
	ECI (P1.2) ขาอินพุตสัญญาณนาฬิกาจากภายนอกสำหรับโมดูล PCA
	CEX0(P1.3) ขาอินพุตเอาต์พุตภายนอกของวงจรตรวจจับ และเปรียบเทียบสัญญาณสำหรับ PCA โมดูล 0
	CEX1(P1.4) ขาอินพุตเอาต์พุตภายนอกของวงจรตรวจจับ และเปรียบเทียบสัญญาณสำหรับ PCA โมดูล 1
	CEX2(P1.5) ขาอินพุตเอาต์พุตภายนอกของวงจรตรวจจับ และเปรียบเทียบสัญญาณสำหรับ PCA โมดูล 2
	CEX3(P1.6) ขาอินพุตเอาต์พุตภายนอกของวงจรตรวจจับ และเปรียบเทียบสัญญาณสำหรับ PCA โมดูล 3
	CEX4(P1.7) ขาอินพุตเอาต์พุตภายนอกของวงจรตรวจจับ และเปรียบเทียบสัญญาณสำหรับ PCA โมดูล 4
P2.0 – P2.7	- ใช้งานเป็นขาพอร์ตอินพุตเอาต์พุตสำหรับใช้งานทั่วไป - ใช้ต่อกับขาแอสแตโรสไบต์สูงของหน่วยความจำภายนอก(A8-A15)
P3.0 – P3.7	- ใช้งานเป็นขาพอร์ตอินพุตเอาต์พุตสำหรับใช้งานทั่วไป - ใช้งานเป็นขาพอร์ตที่พิเศษ ดังต่อไปนี้
	RxD (P3.0) ใช้เป็นขาอินพุตสำหรับรับข้อมูลจากการสื่อสารแบบอนุกรม
	TxD (P3.1) ใช้เป็นขาอินพุตสำหรับส่งข้อมูลจากการสื่อสารแบบอนุกรม
	INT0(P3.2) ใช้เป็นขาอินพุตรับสัญญาณอินเทอร์รัพท์จากภายนอกช่อง 0
	INT1(P3.3) ใช้เป็นขาอินพุตรับสัญญาณอินเทอร์รัพท์จากภายนอกช่อง 1
	T0 (P3.4) ใช้เป็นขาอินพุตสำหรับรับสัญญาณไทเมอร์จากภายนอกช่อง 0
	T1 (P3.5) ใช้เป็นขาอินพุตสำหรับรับสัญญาณไทเมอร์จากภายนอกช่อง 1
	\overline{WR} (P3.6) ใช้เป็นสัญญาณในกรณีใช้เชื่อมต่อหน่วยความจำข้อมูลภายนอก
	\overline{RD} (P3.7) ใช้เป็นสัญญาณในกรณีใช้เชื่อมต่อหน่วยความจำข้อมูลภายนอก
RESET	ใช้ในการรีเซ็ตการทำงานของไมโครคอนโทรลเลอร์ โดยในการป้อนสัญญาณลอจิก “1” อย่างน้อยเป็นเวลา 2 แมกซ์ไซเคิล โดยที่วงจรกำเนิดสัญญาณนาฬิกายังคงทำงานต่อเนื่องไปอย่างเป็นปกติ
ALE	Address Latch Enable : ออกมาทุกๆ แมกซ์ไซเคิล อย่งไรก็ตาม สามารถดิสเอเบิลสัญญาณพัลส์นี้ได้ โดยการเซตบิต 0 ของรีจิสเตอร์ AUXR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่แต่ละขาของไมโครคอนโทรลเลอร์	
\overline{PSEN}	Program Store Enable : ใช้ในการส่งสัญญาณเพื่อร้องขอติดต่อกับหน่วยความจำโปรแกรมภายนอกในการอ่าน - เขียนข้อมูลในหน่วยความจำด้วยกระบวนการ ISP
\overline{EA} / V_{pp}	External Access enable/Programming voltage input : ใช้สำหรับเลือกการติดต่อหน่วยความจำโปรแกรมจากภายนอก หรือภายในตัวไมโครคอนโทรลเลอร์ “0” เลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำโปรแกรมภายนอก “1” เลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำภายใน
XTAL1	ขาอินพุตรับสัญญาณจากวงจรขยายออสซิลเลเตอร์ (ขา XTAL2) และจากภายนอก
XTAL2	ขาเอาต์พุตรับสัญญาณจากวงจรขยายออสซิลเลเตอร์ภายในไมโครคอนโทรลเลอร์

2.5 แผงแสดงผลแบบ LCD

อุปกรณ์ที่ใช้ติดต่อกับไมโครคอนโทรลเลอร์ในด้านแสดงผลมีมากมาย ซึ่งข้อดีของ LCD ก็คือว่า สามารถแสดงตัวอักษรที่มีความละเอียดได้มากกว่า และกินไฟต่ำ พร้อมทั้งมีให้เลือกใช้งานได้หลายแบบ โดย LCD แบบตัวอักษรที่มีจำหน่ายและใช้กันค่อนข้างมากเป็นโมดูลสำเร็จรูป ภายในมี LSI ชิพเบอร์ HD44780 ทำหน้าที่ควบคุม เนื่องจากโมดูล LCD ดังกล่าวมีชิพ HD44780 ทำหน้าที่รับข้อมูลที่ต้องการจากไมโครคอนโทรลเลอร์และจะกำเนิดสัญญาณที่ใช้ควบคุม LCD ข้อมูลที่ไมโครคอนโทรลเลอร์ส่งไปกับ HD44780 จะเก็บในหน่วยความจำบนชิพ HD44780

2.5.1 ชุดคำสั่งของ LCD Module

INSTRUCTION	RS	R/ \overline{W}	DATA BIT								EXECUTE TIME (nS)	
			7	6	5	4	3	2	1	0		
CLEAR DISPLAY	0	0	0	0	0	0	0	0	0	0	1	1640
CURSOR AT HOME	0	0	0	0	0	0	0	0	0	1	*	1640
ENTRY MODE SET	0	0	0	0	0	0	0	1	I/D	S		40
DISPLAY ON/OFF	0	0	0	0	0	0	1	D	C	B		40
DISPLAY SHIFT	0	0	0	0	0	1	S/C	R/L	*	*		40
FUNCTION SET	0	0	0	0	1	DL	N	F	*	*		40
SET CGRAM ADDR.	0	0	0	1	CGRAM ADDRESS						40	
SET DDRAM ADDR.	0	0	1	DDRAM ADDRESS						40		
BUSY,ADDR,READ	0	1	BF	ADDRESS						0		
CGRAM,DDRAM WR	1	0	WRITE DATA						40			
CGRAM,DDRAM RD	1	1	READ DATA						40			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1.1 คำสั่งเคลียร์ตัวแสดงผล (CLEAR DISPLAY)

RS	R/ \overline{W}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0	1

คำสั่ง CLEAR DISPLAY ใช้เขียนข้อมูลหรืออักขรว่างลงใน DDRAM ทั้งหมดและกำหนด DDRAM Address เป็น 0 และเคอร์เซอร์จะกลับไปอยู่ที่ตำแหน่งบนซ้ายสุดของจอแสดงผล

2.5.1.2 คำสั่ง CURSOR AT HOME

RS	R/ \overline{W}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0	*

คำสั่ง CURSOR AT HOME หรือ RETURN HOME เป็นคำสั่งที่ใช้ในการเลื่อนตำแหน่งของเคอร์เซอร์ไปอยู่ที่ตำแหน่งบนซ้ายสุดของจอแสดงผล โดยข้อมูลที่อยู่ใน DDRAM หรือที่หน้าจอแสดงผลจะไม่เปลี่ยนแปลง

2.5.1.3 คำสั่งโหมดในการป้อนข้อมูล (ENTRY MODE SET)

RS	R/ \overline{W}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	1	I/D	S

เป็นโหมดใช้สำหรับกำหนดการเลื่อนของเคอร์เซอร์และตำแหน่งแอดเดรสของ DDRAM

-I/D เป็นบิตที่ใช้ในการกำหนดการเลื่อนของเคอร์เซอร์และตำแหน่งแอดเดรสของ DDRAM ว่าจะให้เพิ่มหรือลดลงเมื่อเขียนหรืออ่านข้อมูลแล้ว

บิต I/D = 0 แอดเดรสของ DDRAM จะลดลง

บิต I/D = 1 แอดเดรสของ DDRAM เพิ่มขึ้นส่วนเคอร์เซอร์จะเลื่อนตามตำแหน่งของแอดเดรส

-S เป็นบิตที่ใช้กำหนดลักษณะของการแสดงผลเมื่อมีการเขียนข้อมูล

บิต S = 1 เมื่อเขียนข้อมูลใหม่ลงไปแล้วตัวเคอร์เซอร์จะอยู่กับที่แต่ตัวอักษรข้อมูลเดิมจะถูกผลักไปทางซ้าย

บิต S = 0 เมื่อเขียนข้อมูลใหม่ลงไปแล้วตัวเคอร์เซอร์จะเลื่อนไปทางขวามือ

2.5.1.4 คำสั่งควบคุมการแสดงผล (DISPLAY ON/OFF)

RS	R/ \overline{W}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	1	D	C	B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งควบคุมการแสดงผลเป็นคำสั่งที่ใช้ในการเปิดปิดจอแสดงผลและเคอร์เซอร์มีลักษณะดังนี้

D = 0 กำหนดให้ปิดจอแสดงผล (Display OFF)

D = 1 กำหนดให้เปิดจอแสดงผล (Display ON)

C = 0 กำหนดให้ปิดเคอร์เซอร์ (Cursor OFF)

C = 1 กำหนดให้เปิดเคอร์เซอร์ (Cursor ON)

B = 0 กำหนดให้ไม่มีการกระพริบที่เคอร์เซอร์

B = 1 กำหนดให้มีการกระพริบที่เคอร์เซอร์ (กระพริบเป็นรูปสี่เหลี่ยมทึบ)

2.5.1.5 คำสั่งควบคุมการเลื่อนเคอร์เซอร์และตัวอักษร (DISPLAY SHIFT)

RS	R/ \bar{H}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	1	S/C	R/L	*	*

คำสั่งควบคุมการเลือกเคอร์เซอร์และตัวอักษร เป็นการควบคุมการเลื่อนของเคอร์เซอร์และตัวอักษรบนจอแสดงผล โดยขึ้นอยู่กับกำหนัดบิต S/C และ R/L โดยมีลักษณะดังนี้

S/C	R/L	ลักษณะการเลื่อน
0	0	เลื่อนเคอร์เซอร์ไปทางซ้าย
0	1	เลื่อนเคอร์เซอร์ไปทางขวา
1	0	เลื่อนเคอร์เซอร์ตัวใหม่ไปทางซ้าย
1	1	เลื่อนเคอร์เซอร์ตัวใหม่ไปทางขวา

2.5.1.6 คำสั่งกำหนดฟังก์ชันการทำงาน (FUNCTION SET)

RS	R/ \bar{H}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	DL	N	F	*	*

DL = 0 กำหนดให้ติดต่อกับ LCD Module เป็นแบบ 4 บิต

DL = 1 กำหนดให้ติดต่อกับ LCD Module เป็นแบบ 8 บิต

N = 0 กำหนดการแสดงผลแบบ 1 บรรทัด

N = 1 กำหนดการแสดงผลแบบ 2 บรรทัดขึ้นไป

F = 0 กำหนดความละเอียดของการแสดงผลเป็น 5 x 8 Dot.

F = 1 กำหนดความละเอียดของการแสดงผลเป็น 5 x 10 Dot.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1.7 คำสั่งเลือกแอดเดรสของ CGRAM (SET CGRAM ADDRESS)

RS	R/ \overline{W}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	1	CGRAM ADDRESS					

คำสั่งนี้ใช้สำหรับกำหนดตำแหน่ง Address ของ Character Generator หรือ CGRAM โดยจะต้องกำหนดค่านี้ทุกครั้งที่ในการเขียนหรืออ่านข้อมูลกับ CGRAM ซึ่งกำหนดที่ (DB0 – DB5) ส่วน DB6 ต้องเป็น “1” และ DB7 ต้องเป็น “0” (01XX XXX B) ซึ่งก็คือ (40H – 7FH)

2.5.1.8 คำสั่งเลือกแอดเดรสของ DDRAM (SET DDRAM ADDRESS)

RS	R/ \overline{W}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	DDRAM ADDRESS						

การกำหนด Address ของ DDRAM หรือตำแหน่งของเคอร์เซอร์สำหรับการแสดงผลทางหน้าจอ LCD ซึ่งเมื่อมีการอ่านหรือเขียนค่าตัวอักษรให้กับ LCD ในแต่ละครั้งนั้น ค่าตำแหน่งของ DDRAM Address จะขึ้นหรือลดลง 1 ตำแหน่งโดยอัตโนมัติเสมอ ซึ่งจะเพิ่มหรือลดนั้นกำหนดได้จากบิต I/D ใน ENTRY MODE SET แต่เราก็สามารถที่จะกำหนดตำแหน่งแอดเดรสของ DDRAM ณ ตำแหน่งใดๆ ก็ได้บนจอ LCD ที่เราต้องการ โดยกำหนดแอดเดรสก่อนที่จะทำการอ่านหรือเขียนตัวอักษรให้กับ DDRAM Address เสมอ

2.5.1.9 คำสั่งอ่านแฟล็ก BUSY และ Address (Read Busy Flag & Address)

RS	R/ \overline{W}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	BF	CGRAM/DDRAM ADDRESS						

คำสั่งนี้ใช้สำหรับการอ่านค่าของ Busy Flag (BF) ซึ่งบอกถึงความพร้อมของ LCD ในการรับข้อมูล ถ้า BF = 1 หมายถึง LCD พร้อมที่จะรับข้อมูลต่อไปได้
ถ้า BF = 0 หมายถึงว่ายังไม่พร้อมที่จะรับข้อมูล

2.6 ระบบบัสแบบ I²C และการเชื่อมต่อ

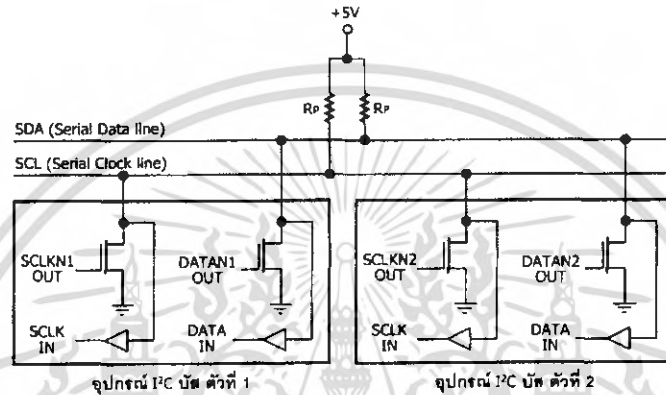
I²C ซึ่งย่อมาจากคำว่า Inter – IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซี โดยให้ไอซีหรือโมดูลสามารถติดต่อทำงาน และควบคุมภายใต้สายสัญญาณเพียง 2 เส้น คือ สายข้อมูล (SDL : Serial Data Line) และสายสัญญาณนาฬิกา (SCL : Serial Clock Line) ที่ใช้ในการกำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์บนบัส I²C ทำได้โดยการต่อสายข้อมูลและสายสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นาฬิกาของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไป ส่วนการกำหนดแอดเดรสหรือติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสถานะลอจิกที่ขาแอดเดรสของอุปกรณ์แต่ละตัว

2.6.1 คุณสมบัติโดยทั่วไปของบัส I²C

สาย SDA และ SCL เป็นสายสัญญาณ 2 ทิศทาง ต้องมีการต่อตัวต้านทานพูลอัพกันแรงดัน +5V ไว้ตลอดเวลา เพื่อให้สายมีสถานะลอจิกสูงในกรณีที่ไม่มีการทำงานทั้งยังช่วยในการป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสอง



รูปที่ 2.7 แสดงวงจรเอาต์พุตของอุปกรณ์ในระบบบัส I²C

อัตราการถ่ายเทข้อมูลบนบัส I²C สูงถึง 100 กิโลบิตต่อวินาทีในโหมดปกติ และสูงถึง 400 กิโลบิตต่อวินาทีในโหมดความเร็วสูง อุปกรณ์ที่ต่อจะต้องมีค่าความจุไฟฟ้าที่รวมที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400 pF

ข้อเด่นอีกประการหนึ่งของบัส I²C สามารถเชื่อมต่ออุปกรณ์ที่ใช้ไฟเลี้ยงไม่เท่ากันให้สามารถติดต่อสื่อสารกันได้ โดยอุปกรณ์บนบัส I²C ตัวหนึ่งอาจใช้ไฟเลี้ยง +5V ในขณะที่อีกตัวหนึ่งใช้ไฟเลี้ยง +12 V การต่อร่วมกันบนบัส I²C สามารถกระทำได้ในลักษณะเดียวกับกรณีที่อุปกรณ์ทั้งสองใช้ไฟเลี้ยงเท่ากัน

2.6.2 หลักการของบัส I²C

บัส I²C สามารถต่ออุปกรณ์ได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูปแบบของการติดต่อบนบัส เพื่อให้ทราบว่าขณะนี้อุปกรณ์ใดติดต่อกันอยู่ ตัวใดเป็นตัวรับหรือตัวส่ง หลักการการทำงานดังนี้ อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่า ตัวส่ง (transmitter)

อุปกรณ์ที่เป็นผู้รับข้อมูล เรียกว่า ตัวรับ (receiver) อุปกรณ์บนบัส I²C สามารถได้ทั้งตัวรับและตัวส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัส I²C ที่ทำหน้าที่เป็นตัวส่งเพียงอย่างเดียว

อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส I²C เรียกว่า มาสเตอร์ (master)
 อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส I²C เรียกว่า สเลฟ (slave)

ข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัส I²C คือ

1. การถ่ายทอดข้อมูลขณะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น
2. ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ อย่าให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้น สัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาณควบคุมแทน

2.6.3 สภาวะที่เกิดขึ้นบนบัส I²C

มีด้วยกัน 5 สภาวะ ดังนี้

2.6.3.1 บัสว่าง (bus not busy)

สภาวะนี้เกิดขึ้นเมื่อสถานะลอจิกบนสาย SDA และ SCL เป็นเป็นลอจิกสูงทั้งคู่ นั้นหมายความว่า การถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้

2.6.3.2 เริ่มต้นการถ่ายทอดข้อมูล (start data transfer)

เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะเริ่มต้น (START)

2.6.3.3 หยุดการถ่ายทอดข้อมูล (stop data transfer)

เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูง ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกว่าสภาวะที่เกิดขึ้นว่า สภาวะหยุด (STOP)

2.6.3.4 ข้อมูลดำรงอยู่บนบัส (data valid)

สภาวะนี้เกิดขึ้นถัดจากสภาวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA ก็คือข้อมูลที่ทำการถ่ายทอด เมื่อสาย SCL เป็นลอจิกสูง สถานะที่สาย SDA ต้องคงที่ เพื่อให้อุปกรณ์รับรู้ข้อมูลในจังหวะนั้นว่าเป็น “0” หรือ “1” ข้อมูลอาจเกิดการเปลี่ยนแปลงได้ในขณะที่สาย SCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์สถานะลอจิกที่ขา SDA ต้องคงที่ตลอด ช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะลอจิกในขณะที่สาย SCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ทำการควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุดหรือสภาวะเริ่มต้นก็ได้ ทำให้ข้อมูลที่ทำการถ่ายตอดนั้นเกิดความผิดพลาดขึ้น

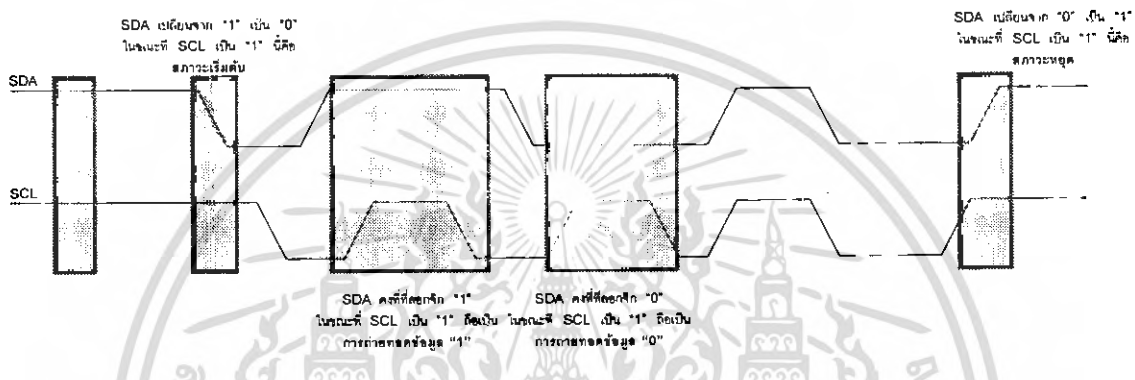
2.6.3.5 รับรู้ข้อมูล (acknowledge)

เกิดขึ้นหลังจากที่การถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิต เรียกว่า บิตรับรู้ (acknowledge bit) มีสถานะเป็นลอจิกสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากส่งข้อมูลมาครบ ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกา เพื่อตอบสนองบิตรับรู้ว่ามาจากตัวส่ง ทางด้านรับจะส่งบิตรับรู้ที่มีสถานะลอจิกต่ำลงบนบัส อุปกรณ์สเลฟที่ถูกอ้างถึงในการติดต่อหรือกำลังติดต่ออยู่ในขณะนั้นก็จะทำกำเนิดบิตรับรู้เพื่อตอบสนองให้ทราบว่าได้รับข้อมูลในแต่ละไบต์เรียบร้อยแล้ว

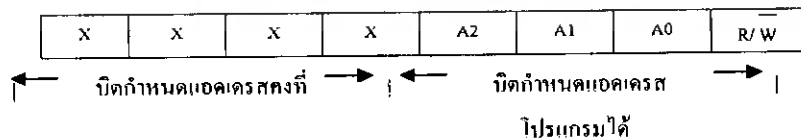
รูปที่ 2.8 เป็นไดอะแกรมเวลาที่แสดงถึงการเกิดสถานะต่างๆ บนบัส I²C ไม่ว่าจะเป็นสภาวะบัสว่าง เริ่มต้น ถ่ายทอดข้อมูล รับรู้ และหยุดการถ่ายทอดข้อมูล



รูปที่ 2.8 ไดอะแกรมเวลาแสดงสถานะต่างๆ ในบัส I²C

2.6.4 การอ้างถึงแบบ 7 บิต (7-bit Addressing)

ข้อมูลไบต์แรกที่เกิดขึ้นหลังจากสภาวะเริ่มต้น เป็นข้อมูลที่ใช้ในการอ้างถึงอุปกรณ์ที่ต้องติดต่อ โดยมีรูปแบบแสดงในรูปที่ 2.9 ในบิต 7 บิตบนจะเป็นข้อมูลแอดเดรสของอุปกรณ์สเลฟที่ต้องการติดต่อ โดยแบ่งเป็น บิตกำหนดแอดเดรสคงที่ จำนวน 4 บิต ซึ่งข้อมูลนี้ อุปกรณ์แต่ละตัวจะถูกกำหนดจากผู้ผลิต ไม่สามารถเปลี่ยนแปลงได้ ถัดมาอีก 3 บิตเป็นตัวกำหนดแอดเดรสที่สามารถโปรแกรมได้ โดยผู้ใช้งานต้องกำหนดสถานะลอจิกให้แก่ขา A0-A2 ของอุปกรณ์ที่มีการเชื่อมต่อแบบบัส I²C ส่วน LSB เป็นบิตที่ใช้กำหนดการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวนั้นๆ



รูปที่ 2.9 แบบของข้อมูลกำหนดแอดเดรสที่ใช้ในการอ้างถึงแบบ 7 บิต

ข้อมูลในไบต์ต่อมาคือ ข้อมูลควบคุม อุปกรณ์แต่ละตัวมีการกำหนดไว้ที่แตกต่างกันไป

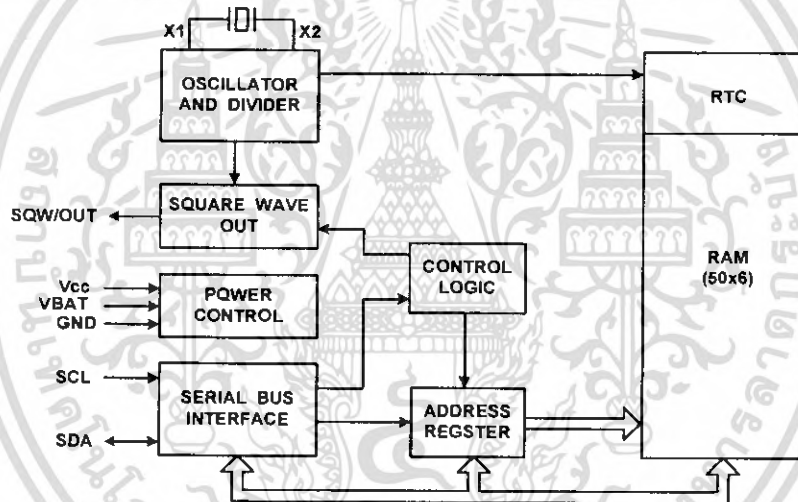
ข้อมูลในไบต์ต่อมาคือ ข้อมูลที่ทำการถ่ายทอดจริง (Data)

หลังจากที่มีการถ่ายถอดข้อมูลในแต่ละไบต์ อุปกรณ์สเลฟที่ได้รับการติดต่อต้องส่งสัญญาณรับรู้ว่าตอบกลับมาด้วยทุกครั้ง เพื่อให้กระบวนการถ่ายถอดข้อมูลสามารถดำเนินต่อไปได้

2.7 ไอซีฐานเวลาจริง DS1307 (Real – Time Clock)

DS1307 มีหน้าที่สร้างฐานเวลาจริงให้แก่ระบบไมโครคอนโทรลเลอร์โดย DS1307 จะให้ข้อมูลเกี่ยวกับเวลาทั้งหมด คุณสมบัติทางเทคนิคมีดังนี้

- เป็นไอซีรีลไทม์คล็อกให้ข้อมูลตั้งแต่วันที่จนถึงปี รวมถึงการปรับวันที่ในปีอธิกสุรทินด้วย สามารถให้ข้อมูลเวลาได้อย่างเที่ยงตรงถึงปีคริสตศักราช 2100
- มีหน่วยความจำอนโวลทาไทล์แรม 56 ไบต์อยู่ภายใน สามารถเก็บข้อมูลทั่วไปได้
- ใช้การเชื่อมต่อแบบระบบบัส I²C
- มีวงจรตรวจจับไฟเลี้ยงต่ำ และสามารถรักษาข้อมูลเวลาไว้ได้แม้ไม่มีไฟเลี้ยงไอซี



รูปที่ 2.10 โค้ดแอมโมแกรมโครงสร้างภายในชิพ DS1307

2.7.1 หน้าที่แต่ละขาของไอซี DS1307

ขา	หน้าที่
SDA	(Serial Data Input/Output) เป็นขารับส่งข้อมูลแบบอนุกรม ในการอินเตอร์เฟสจะต้องมีตัวต้านทานต่อพูล้อัพภายนอกด้วย
SCL	(Serial Clock Input) เป็นขาอินพุตสัญญาณนาฬิกาอนุกรมเพื่อให้เกิดการซิงโครไนซ์ในการรับส่งข้อมูลแบบอนุกรม
V _{CC} , GND	เป็นขาที่ใช้ต่อกับไฟเลี้ยงต่อกับแรงดันไฟ 5 โวลต์ ถ้าแรงดันไฟต่ำกว่า $1.25 \times V_{BAT}$ การรักษาสารองภายในจะทำงานต่อ โดยจะรับพลังงานจากแบตเตอรี่สำรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา	หน้าที่
SQW/OUT	เป็นขาส่งสัญญาณคลื่นสี่เหลี่ยมออกมาทางเอาต์พุต เมื่อเริ่มทำงานบิตนี้จะถูกเซตเป็นลอจิก "1" สัญญาณเอาต์พุตได้จะมีค่าอยู่ 4 ค่า คือ 1 Hz , 4 Hz , 8 Hz และ 32 Hz โดยเราสามารถเลือกได้ในการใช้งานจะต้องมีตัวต้านทานต่อพูลท์ภายนอกด้วย ในการใช้งานบางประเภทไม่จำเป็นต้องใช้ขานี้
X1 . X2	เป็นขาที่ใช้ต่อกับคริสตอลภายนอก โดยใช้ความถี่ 32.768 kHz
V _{BAT}	เป็นขาต่อกับแรงดันไฟเลี้ยงสำรอง โดยทั่วไปแล้วจะใช้แบตเตอรี่แรงดัน 3 โวลต์

2.7.2 หน่วยความจำ RAM ภายในไอซี DS1307

หน่วยความจำ RAM ภายในไอซี DS1307 โดยตำแหน่ง 00H ถึง 07H จะใช้เป็น RTC ส่วนตำแหน่ง 08H ถึง 3FH จะเป็นหน่วยความจำ RAM โดยตำแหน่ง 00H จะเก็บเวลาเป็นวินาที ตำแหน่ง 01 จะเก็บเวลาเป็นนาฬิกา โดยข้อมูลที่เก็บจะอยู่ในรูปแบบรหัส BCD

สำหรับการอ่านและเขียนค่าเวลากับไอซีตัวนี้ทำได้โดยการอ่านและเขียนข้อมูลในลักษณะตัวเลข BCD กับรีจิสเตอร์ที่เก็บค่าต่างๆ ตามที่กำหนดเอาไว้ แสดงในรูปที่ 2.11

BIT7										BIT0	
00H	CH	10 SECONDS				SECONDS				00-59	
	0	10 MINUTES				MINUTES				00-59	
	0	12	24	10 HR A/P	10 HR	HOURS				01-12 00-23	
	0	0	0	0	0	DAY				1-7	
	0	0	10 DATE			DATE				01-28/29 01-30 01-31	
	0	0	0	10 MONTH		MONTH				01-12	
					YEAR				00-99		
07H	OUT	0	0	SQWE	0	0	RS1	RS0			

รูปที่ 2.11 แผนผังรีจิสเตอร์ตำแหน่งต่างๆ ในการจัดการด้านฐานข้อมูลเวลา

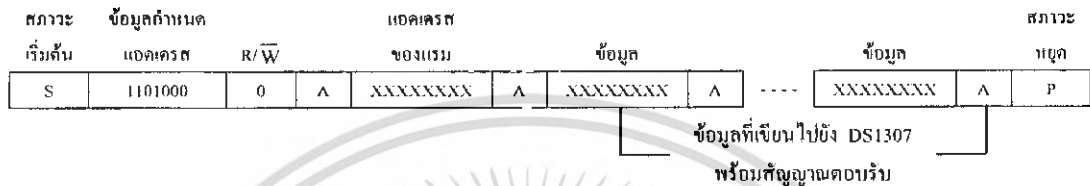
2.7.3 โหมดการทำงานของ DS1307

DS1307 มี 2 โหมด คือโหมดเขียนข้อมูลและโหมดอ่านข้อมูล ตามปกติจะใช้เฉพาะโหมดอ่านข้อมูลเท่านั้น เนื่องจากไมโครคอนโทรลเลอร์จะติดต่อกับ DS1307 เพื่ออ่านข้อมูลเวลาไปใช้งาน โหมดการเขียนข้อมูลจะถูกใช้งานก็ต่อเมื่อการตั้งเวลาใหม่ และต้องการเขียนข้อมูลลงในหน่วยความจำใช้งานทั่วไป เมื่อเริ่มต้นติดต่อกับ DS1307 จะต้องเข้าสู่โหมดการเขียนข้อมูลก่อนเพื่อที่จะทำการกำหนดค่าแอดเดรสที่ต้องการอ่านข้อมูล จากนั้นจึงเปลี่ยนโหมดทำงานมาเป็นโหมดการอ่านข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.3.1 โหมคการเขียนข้อมูล

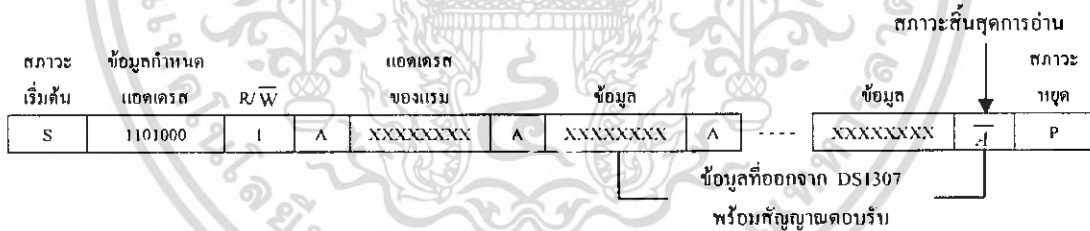
รูปที่ 2.12 เป็นรูปแบบในการเขียนข้อมูล เริ่มเมื่อไมโครคอนโทรลเลอร์ทำการกำหนดสถานะเริ่มต้น จากนั้นส่งข้อมูลกำหนดแอดเดรส 1101000 ตามด้วยข้อมูลเลือกการเขียน นั้นคือค่า 0 แล้วรอการตอบรับจาก DS1307 ต่อมาส่งข้อมูลเพื่อเลือกแอดเดรสที่จะเขียน รอการตอบรับจาก DS1307 เมื่อมีการตอบรับมาเรียบร้อยแล้ว ก็เริ่มทยอยเขียนข้อมูลลงไปครั้งละแอดเดรส เมื่อเขียนข้อมูลเสร็จแล้วให้ส่งสถานะหยุด เป็นอันสิ้นสุดกระบวนการเขียนข้อมูล



รูปที่ 2.12 รูปแบบของข้อมูลสำหรับติดต่อกับ DS1307 ในโหมคการเขียนข้อมูล

2.7.3.2 โหมคการอ่านข้อมูล

มีรูปแบบดังรูปที่ 2.13 เริ่มต้นด้วยไมโครคอนโทรลเลอร์กำหนดสถานะเริ่มต้นแล้วส่งข้อมูลกำหนดแอดเดรสตามด้วยข้อมูลเลือกการอ่าน รอการตอบรับจาก DS1307 เมื่อตอบรับเรียบร้อยแล้ว DS1307 จะทยอยส่งข้อมูลออกมาให้ไมโครคอนโทรลเลอร์คราวละ 1 แอดเดรส โดยแอดเดรสที่เลือกอ่านข้อมูลจะต้องมีการกำหนดมาก่อนด้วยโหมคการเขียนข้อมูล



รูปที่ 2.13 รูปแบบของข้อมูลสำหรับติดต่อกับ DS1307 ในโหมคการอ่านข้อมูล

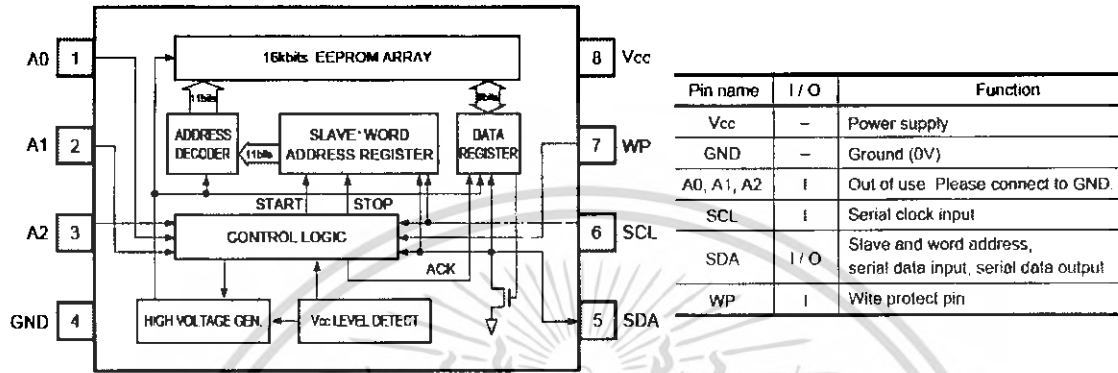
2.8 หลักการของ Serial EEPROM

Serial EEPROM เป็นหน่วยความจำสำหรับระบบไมโครคอนโทรลเลอร์ และสำหรับเก็บรักษาข้อมูลเพราะคุณสมบัติของ EEPROM นั้นบันทึกข้อมูลและคงข้อมูลได้โดยไม่ต้องใช้ไฟเลี้ยงสามารถโปรแกรมและลบได้ด้วยไฟฟ้า

2.8.1 คุณสมบัติ และลักษณะของ Serial EEPROM

- มีขนาด 8 ขา ในลักษณะบรรจุภัณฑ์แบบ DIP และ PLCC
- ความจุหน่วยความจำตั้งแต่ 16 Kbit ถึง 256 Kbit

- ลบได้ 1,000,000 ครั้ง ช่วงอายุการเก็บข้อมูล 100 ปี ถึง 400 ปี (ตามแต่คุณสมบัติของชิพ) โดยไม่ต้องใช้ไฟเลี้ยง
- ใช้ไฟเลี้ยง + 5.0 โวลท์
- ใช้มาตรฐานการสื่อสารแบบ Microwire หรือ I²C หรือ SPI

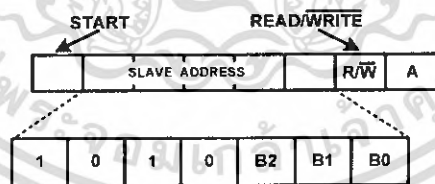


รูปที่ 2.14 โครงสร้างภายในและการจัดขาของชิพ 24C16

2.8.2 ลักษณะการควบคุมการทำงาน 24C16

Slave address ของชิพ 24C16 จะอยู่ในไบต์ควบคุมไบต์แรกที่ตัวส่งให้กับ 24C16 เป็นดังนี้ จะสังเกตว่ามีบิตที่ใช้เลือกตำแหน่งของหน่วยความจำอยู่สามบิตด้วยกัน คือ B2 B1 และ B0

Operation	Control code	Block Select	R/W
Read	1010	Block Address	1
Write	1010	Block Address	0



รูปที่ 2.15 แสดงถึงการใช้บิตเลือกตำแหน่ง

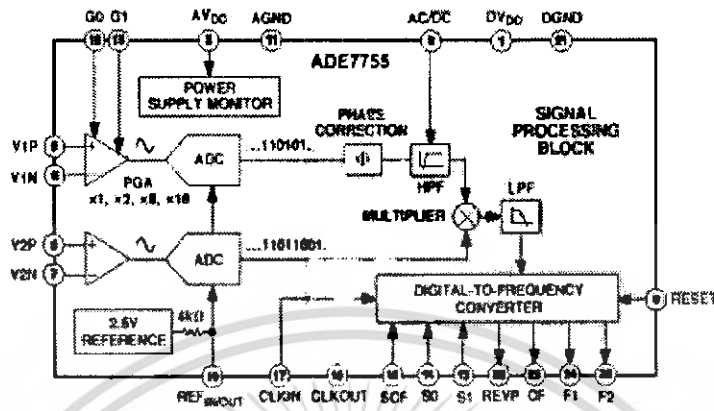
ฟอร์แมตของลำดับไบต์ที่ใช้ในการควบคุมการเขียนข้อมูลลงในชิพ 24C16

เริ่มต้นด้วยไบต์ควบคุมพร้อมทั้งบิตทั้งสามที่ใช้เลือกบิตเลือกตามด้วย word address และข้อมูลที่จะเขียนลงในหน่วยความจำ

กระบวนการการเขียนตำแหน่งที่ต้องการอ่านเช่นเดียวกับการเขียนข้อมูล โดยระบุตำแหน่งในสองไบต์แรก ตามด้วยไบต์ควบคุมแต่เป็นการอ่านแทน ไบต์ที่สี่จะเป็นไบต์ที่อ่านได้ โดยเราเลือกใช้การอินเตอร์เฟสกับไมโครคอนโทรลเลอร์แบบ I²C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 หลักการทำงานของ ADE 7755



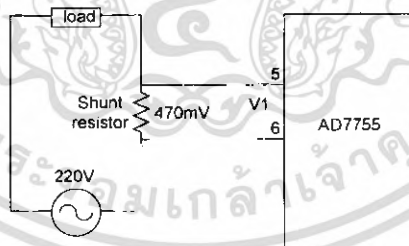
รูปที่ 2.16 Functional Block Diagram ของ ADE7755

2.9.1 หลักการทำงานของส่วนวัดค่าพลังงานไฟฟ้า (ADE 7755)

ส่วนวัดค่าพลังงานไฟฟ้าจะแบ่งออกเป็น 3 ส่วนคือ

2.9.1.1 ส่วนรับค่าสัญญาณอนาล็อกอินพุตที่เป็นกระแสไฟฟ้า

แชลแนล วี 1 ของ ADE 7755 (ขา 5 และขา 6) สัญญาณอินพุตที่เป็นกระแสนี้จะถูกวัดในรูปของความต่างศักย์ โดยใช้ชันดรีซิสเตอร์ค่าความต่างศักย์ที่ได้นำมาต่อขาแชลแนล วี 1 ค่าความต่างศักย์สูงสุด ของสัญญาณที่แชลแนล วี 1 ควรจะไม่เกิน 470 มิลลิโวลต์

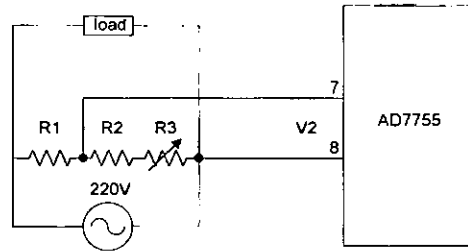


รูปที่ 2.17 การเชื่อมต่อส่วนรับสัญญาณอินพุต V1 กับ ADE 7755

2.9.1.2 ส่วนรับค่าสัญญาณอนาล็อกอินพุตที่เป็นแรงดันไฟฟ้า

แชลแนล วี 2 ของ ADE 7755 (ขา 7 และขา 8) เนื่องจากชิพ ADE 7755 แชลแนล วี 2 รับค่าแรงดันไฟฟ้าสูงสุดได้ 660 มิลลิโวลต์ แต่ค่าแรงดันไฟฟ้าที่จะวัดมีค่าสูงถึง 220 โวลต์ จึงต้องทำการลดค่าของแรงดันไฟฟ้าลงโดยจะทำการใช้วงจรโวลต์เดจิติไวเคอร์ จะนำวงจรโวลต์เดจิติไวเคอร์มาต่อกับขาแชลแนล วี 2 ซึ่งเป็นแรงดันไฟฟ้า

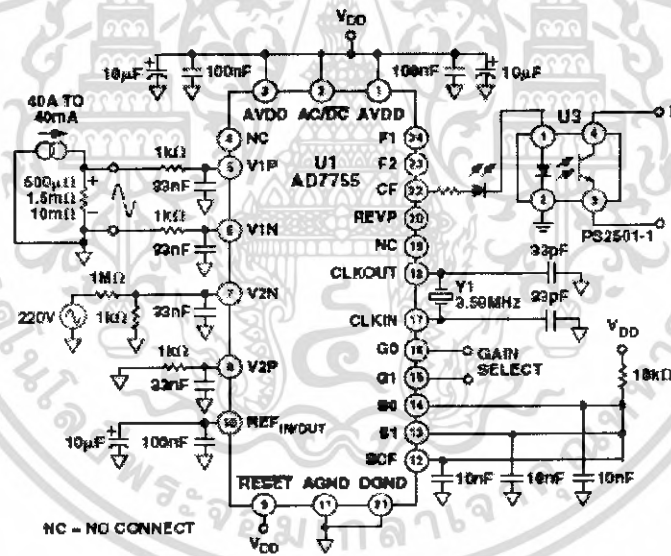
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 แสดงการเชื่อมต่อส่วนรับสัญญาณอินพุต V2 ADE 7755

2.9.1.3 ส่วนส่งค่าพลังงานไฟฟ้าที่วัดได้ไปยังไมโครคอนโทรลเลอร์

ADE 7755 จะแปลงสัญญาณอนาล็อกที่รับเข้ามาทางอินพุตให้เป็นดิจิทัล หลังจากนั้นนำค่าของกระแสไฟฟ้า และแรงดันไฟฟ้าที่เป็นดิจิทัลมาคูณกันด้วยวงจรคูณซึ่งจะทำให้ได้ค่าของพลังงานไฟฟ้า ก่อนที่จะทำการส่งค่าพลังงานไฟฟ้าออกมาจะแปลงค่าพลังงานไฟฟ้าที่เป็นดิจิทัลนี้เป็นความถี่ เอาท์พุทที่ได้จึงอยู่ในรูปของพัลส์ที่มีความถี่ขึ้นอยู่กับการพลังงานไฟฟ้าที่ใช้



รูป 2.19 Test Circuit for Perfomance Curve

2.9.2 ขาหน้าที่ของแต่ละขาภายใน IC ADE 7755

หน้าที่ของแต่ละขาภายในชิพ ADE 7755	
DV _{DD}	Digital Power Supply ใช้สำหรับจ่าย Digital Circuitry ภายในตัว ADE 7755 จ่ายไฟเลี้ยง +5 V ต่อกับ C ค่า 10uF ขนาดกับ C ค่า 100nF
AV _{DD}	Analog Power Supply ใช้สำหรับจ่าย Analog Circuitry ภายในตัว ADE 7755 จ่ายไฟเลี้ยง +5 V ต่อกับ C ค่า 10uF ขนาดกับ C ค่า 100nF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่ของแต่ละขาภายในชิพ ADE 7755	
DGND	ต่อกราวด์สำหรับ Digital Circuitry ภายในตัว ADE 7755
AGND	ต่อกราวด์สำหรับ Analog Circuitry ภายในตัว ADE 7755
AC/ \overline{DC}	High Pass Filter Select เป็น input ง่ายให้ HPF in Channel 1 ตั้งแต่ 45Hz – 1kHz ค่าเป็น high ใช้วัดกระแสสลับ ค่าเป็น low วัดกระแสตรง
VIP,VIN	Analog Inputs for Channel 1 (Current Channel) แรงดันที่จ่ายให้ไม่เกิน 470mV
V2N,V2P	Negative and Positive Input for Channel 2 (Voltage Channel) แรงดันที่จ่ายให้ไม่เกิน 660mV
SCF	Select Calibration Frequency ใช้เลือกความถี่ที่เป็นมาตรฐานของ Output CF
CF	Calibration Frequency Logic Output เป็นการส่งค่าเป็น real power
S1,S0	ใช้เลือกความถี่ที่ใช้สำหรับ digital of frequency conversion ซึ่งใช้สำหรับ Energy Meter Application
G1,G0	ใช้ในการเลือก Gain สำหรับ Channel 1
CLKIN	รับสัญญาณ Clock จากภายนอกเข้าตัว ADE 7755
CLKOUT	เป็นสัญญาณ Clock ที่ส่งออกไปใช้งาน
\overline{RESET}	ขา Reset ใช้ในการ Clear ค่า register ภายใน IC
REF _{IN/OUT}	ขานี้ใช้ในการจัดการการเข้าถึง on - chip voltage reference ทำการต่อกับ C ค่า 1uF ขนาดกับ C ค่า 100nF ลงกราวด์
REVP	ค่า Output ที่ได้มาจะมีค่าเป็น Logic high เมื่อตรวจพบค่า Negative power
F2,F1	Low Frequency Logic Outputs เป็น Output ใช้ขับ electromechanical counters

2.10 Networking Concepts and Protocols

2.10.1 Physical Networks

หมายถึงกลุ่มของคอมพิวเตอร์ หรืออุปกรณ์ซึ่งเชื่อมโยงและสื่อสารกันได้ผ่านการเชื่อมต่อในรูปแบบใดรูปแบบหนึ่ง โดยที่อุปกรณ์ใด ๆ ซึ่งเชื่อมต่ออยู่กับโครงข่าย(Network) จะถูกเรียกว่า โหนด (node) โดยที่โหนดแต่ละโหนดจะเชื่อมต่อกันโดยลิงค์ (Links) แต่ละโหนดในโครงข่ายนั้นสามารถสื่อสารได้โดยการส่งข้อมูลถึงกันได้ผ่านทางลิงค์เหล่านี้ สำหรับตัวโครงข่ายเองแล้วสามารถจำแนกออกเป็นประเภทต่าง ๆ โดยใช้ขนาดของโครงข่ายเป็นเกณฑ์ในการจำแนกได้ดังนี้

- LAN (Local Area Network)
- WAN (Wide Area Network)
- MAN (Metropolitan Area Network)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการที่จะให้แต่ละโหนดในโครงข่ายสามารถสื่อสารระหว่างกันได้ นอกจากการเชื่อมต่อแต่ละโหนดเข้าด้วยกันทางกายภาพด้วยลิงค์แล้วนั้น ยังมีความจำเป็นที่จะต้องกำหนดรูปแบบและขั้นตอนของการส่งสัญญาณ โดยรูปแบบและขั้นตอนดังกล่าวจะถูกเรียกว่า Protocol ในปัจจุบัน Protocol ซึ่งเป็นที่นิยมเชื่อมต่อ LAN ได้แก่ Ethernet โดย Ethernet Protocol ยังได้ถูกพัฒนาต่อไป ทำให้เกิด Ethernet Protocol อีกหลายมาตรฐานมีข้อแตกต่างกันที่ความเร็วในการรับส่งข้อมูล หรือสายสัญญาณที่ใช้

2.10.2 Network Protocols

OSI จะเป็นการกำหนดรูปแบบของ Protocol Layers ในรูป 2.20 จะเป็นการนำเอาชุดของ Protocol ที่ใช้งานจริง ในที่นี้คือ TCP/IP Protocol Stack มาเปรียบเทียบกับ OSI Model

Application Layer	HTTP	FTP	SMTP	RIP	DNS
Presentation Layer					
Session Layer					
Transport Layer	TCP		UDP		
Network Layer	ICMP	IP	IGMP		
Data Link Layer					
Physical Layer	ETHERNET , ATM , Etc				

รูปที่ 2.20 OSI-TCP/IP Protocol Stack

อาจกล่าวได้ว่า TCP/IP Protocol Stack เป็นรูปแบบอย่างง่ายของ OSI Model กล่าวคือสามารถมองเป็น 4 Layers ได้ โดยในชุดของ TCP/IP Protocol จะพบว่า Protocol ในชั้นบนสุดนั้นจะไม่สามารถจัดอยู่ใน Layer เพียง Layer เดียวได้ ด้วยเหตุผลที่ว่า Functions การทำงานของ Protocol เหล่านั้นครอบคลุมทั้ง 3 Layers ของ OSI Model

2.11 Sockets

โดยทั่วไปในการติดต่อสื่อสารที่เป็นที่นิยมใช้กันในปัจจุบันจะกระทำบนพื้นฐานของ Protocol TCP/IP และชุดของ Software ซึ่งทำงานกับ Packets ตามมาตรฐานของ TCP/IP นั้นจะถูกสร้างขึ้นมาเป็นส่วนหนึ่งของตัวระบบปฏิบัติการ ดังนั้น เมื่อ Application Software ใด ๆ ที่ต้องการสื่อสารผ่านระบบเครือข่ายตามมาตรฐาน TCP/IP Protocol จะต้องติดต่อกับระบบปฏิบัติการเพื่อขอรับบริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีของการออกแบบ TCP/IP คือ ไม่กำหนดรายละเอียดปลีกย่อยทำให้นักออกแบบระบบสามารถที่จะใส่ TCP/IP ลงในระบบปฏิบัติการได้โดยง่าย นอกจากนั้นด้วยการที่ไม่กำหนดวิธีการเชื่อมต่อกับ Application Software ทำให้นักออกแบบระบบสามารถเลือกใช้รูปแบบในการเชื่อมต่อได้อย่างอิสระ ข้อดีของ TCP/IP คือการที่นักออกแบบระบบมีอิสระในการใส่รายละเอียดในการเชื่อมต่อจึงทำให้เกิดความหลากหลายของรูปแบบการเชื่อมต่อ การเขียน Application Program เพื่อการใช้งาน TCP/IP มีความยุ่งยากและซับซ้อนมากยิ่งขึ้น

2.11.1 Berkeley Sockets

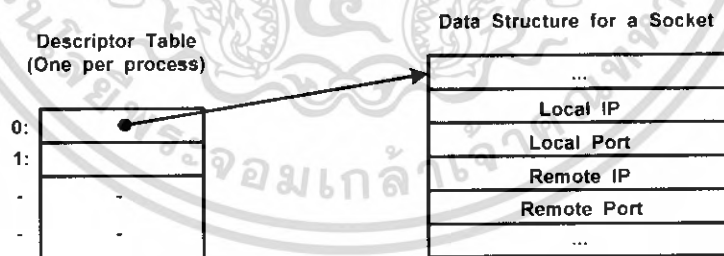
Berkeley Sockets Interface นั้นคือการสนับสนุนการสื่อสารข้อมูลผ่านเครือข่ายด้วย Protocols ที่หลากหลายที่สุดเท่าที่จะเป็นไปได้

2.11.1.1 System Data Structure for Sockets

เมื่อสร้าง Socket ใหม่ระบบปฏิบัติการจะ allocate พื้นที่ในหน่วยความจำส่วนหนึ่งเพื่อสร้าง Data Structure สำหรับเก็บข้อมูลซึ่งจำเป็นสำหรับการสื่อสารข้อมูลผ่านเครือข่าย ซึ่ง Data Structure นี้จะประกอบไปด้วย field หลาย field ด้วยกัน

2.11.1.2 Sockets Descriptors

ในระบบปฏิบัติการมีการสร้าง Descriptor Table สำหรับเก็บ Data Structure ซึ่งเกี่ยวข้องกับ I/O โดย Descriptor Table สำหรับแต่ละ Process จะถูกเก็บแยกกัน ซึ่งมีลักษณะเป็น Array of Pointer ที่ชี้ไปยัง Data Structure ที่เกี่ยวข้อง ใน Sockets ก็จะเป็นเช่นเดียวกัน ทั้งนี้เพราะ Sockets นับเป็น I/O ด้วยเช่นกัน



รูปที่ 2.21 Descriptor Table and Data structure for a socket

2.11.1.3 Using Sockets

Sockets ที่ถูกสร้างขึ้นนั้นจะมีวัตถุประสงค์ในการใช้งานอยู่เพียง 2 ลักษณะได้แก่ การรอรับการเชื่อมต่อและการเชื่อมต่อไปยังเครื่องปลายทาง โดย Sockets ที่รอรับการเชื่อมต่อนั้นจะเป็น Sockets ที่ใช้ใน Server Application เรียกว่า Passive Socket ในขณะที่ Sockets ที่พยายามเชื่อมต่อไปยังเครื่องปลายทางซึ่งใช้ใน Client Application เรียกว่า Active Socket

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.2 Working with Sockets in .Net

ใน .Net การทำงานที่เกี่ยวข้องกับ Sockets จะต้องเรียกใช้ System.Net.Sockets เป็น Namespace ที่บรรจุ Classes ต่าง ๆ ที่สนับสนุนการทำงานกับ Sockets Classes ที่เกี่ยวข้องกับ Sockets หลักๆ มีดังนี้

- MulticastOption
- TcpClient
- UdpClient
- Socket
- NetworkStream
- TcpListener
- SocketException

โดย Class Socket จะมี function พื้นฐานต่าง ๆ ที่จำเป็นสำหรับการสร้าง Application ซึ่งใช้งาน Sockets สำหรับ Properties ที่สำคัญสำหรับ System.Net.Sockets.Socket ได้แก่

- o AddressFamily
- o Blocking
- o LocalEndPoint
- o RemoteEndPoint
- o Available
- o Connected
- o ProtocolType
- o SocketType

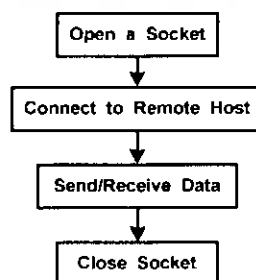
และ Method ที่สำคัญของ System.Net.Sockets.Socket มีดังนี้

- o Accept()
- o Bind()
- o Close()
- o Connect()
- o GetSocketOption()
- o IOControl()
- o Listen()
- o Receive()
- o Poll()
- o Select()
- o Send()
- o SetSocketOption()
- o Shutdown()

2.11.3 Working with Sockets

2.11.3.1 Active Sockets

Active Sockets หรือ Socket ใช้งานใน Client Application มีขั้นตอนการสร้างดังนี้

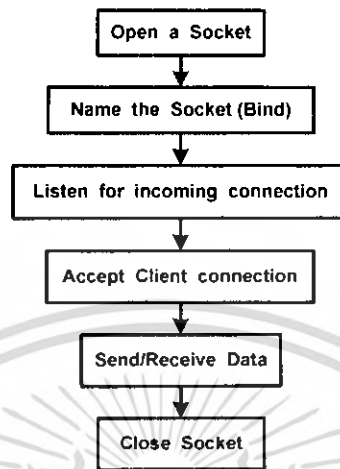


รูปที่ 2.22 Diagram ขั้นตอนการสร้าง Active Sockets

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.3.2 Passive Sockets

Passive Sockets หรือ Sockets ที่ใช้ใน Server Application นั้นมีขั้นตอนการสร้างดังนี้



รูปที่ 2.23 Diagram ขั้นตอนการสร้าง Passive Sockets

2.12 Stream Sockets (Connection Oriented Socket)

Stream Socket เป็น Socket ที่ใช้ Protocol TCP ในการสื่อสารข้อมูล โดยลักษณะการทำงานจะแยกออกเป็น 2 ฝั่งคือ Server และ Client ขั้นตอนการเปิด Socket ใน Application ของทั้ง 2 ฝั่งนั้นจะมีความแตกต่างกัน โปรแกรม 2.1 เป็นโปรแกรมเพื่อเปิด Socket ในฝั่งของ Server Application และโปรแกรม 2.2 เป็นตัวอย่างของโปรแกรมสำหรับใช้เชื่อมต่อกับ Socket ใน Client Application

โปรแกรม 2.1

```

using System;
using System.Net;
using System.Net.Sockets;
using System.Text;

class TcpSrvr
{
    public static void Main()
    {
        int recv;
        byte[] data = new byte[1024];
        IPEndPoint ipep = new IPEndPoint(IPAddress.Any, 9050);

        Socket newsock = new Socket(AddressFamily.InterNetwork,
            SocketType.Stream, ProtocolType.Tcp);
  
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

newsock.Bind(ipcp);
newsock.Listen(10);
Console.WriteLine("Waiting for a client...");

Socket client = newsock.Accept();
string welcome = "Welcome to my test server";
data = Encoding.ASCII.GetBytes(welcome);
client.Send(data, data.Length, SocketFlags.None);
IPEndPoint newclient = (IPEndPoint)client.RemoteEndPoint;
Console.WriteLine("Connected with {0} at port {1}",
    newclient.Address, newclient.Port);

```

A

```

for (int i = 0; i < 5; i++)
{
    recv = client.Receive(data);
    Console.WriteLine(Encoding.ASCII.GetString(data, 0, recv));
}
Console.WriteLine("Disconnecting from {0}", newclient.Address);
client.Close();
newsock.Close();
}
}

```

โปรแกรม 2. 2

```

using System;
using System.Net;
using System.Net.Sockets;
using System.Text;
class TcpClient
{
    public static void Main()
    {
        byte[] data = new byte[1024];

```

1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

string stringData;
IPEndPoint ipep = new IPEndPoint(IPAddress.Parse("127.0.0.1"), 9050);

Socket server = new Socket(AddressFamily.InterNetwork,
    SocketType.Stream, ProtocolType.Tcp);

try
{
    server.Connect(ipep);
} catch (SocketException c)
{
    Console.WriteLine("Unable to connect to server.");
    Console.WriteLine(e.ToString());
    return;
}

int recv = server.Receive(data);
stringData = Encoding.ASCII.GetString(data, 0, recv);
Console.WriteLine(stringData);

```

B

```

    Console.WriteLine("Disconnecting from server...");
    server.Shutdown(SocketShutdown.Both);
    server.Close();
}
}

```

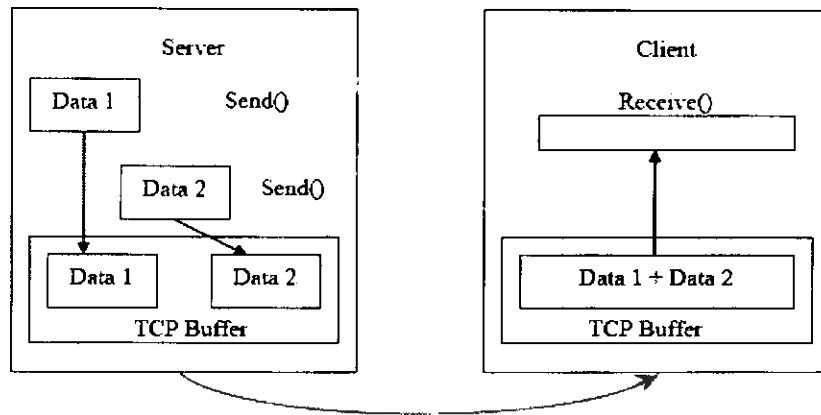
2.12.1 ข้อควรระวังในการใช้งาน Stream Socket

ในการใช้งาน Stream Socket หรือ Connection Oriented Socket นั้น มีข้อควรระวังอยู่ 2 ประการ ได้แก่

2.12.1.1 ปัญหาด้านการจัดการ Buffer

สำหรับ Buffer ในการทำงานกับ Stream Socket นั้นจะแบ่งเป็น 2 ส่วน ส่วนแรกเป็น Buffer ของระบบ ซึ่งเป็น Buffer ที่จะรับเอาข้อมูลซึ่งส่งผ่านโครงข่ายไว้เพื่อรอโปรแกรมใช้คำสั่ง Receive() เพื่ออ่านออกไป สำหรับ Buffer ในส่วนที่ 2 นั้นจะเป็น Buffer ซึ่งเกิดจากการที่นักเขียน โปรแกรมสร้างตัวแปรขึ้นมาเพื่อให้รองรับข้อมูลที่อ่านจาก TCP Buffer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 แบบจำลอง TCP Buffer

สำหรับฝั่งของ Server TCP Buffer มีไว้รองรับการ re-transmit ในกรณีที่ Packet ที่ส่งไปเกิดสูญหาย ข้อมูลใน Buffer จะถูกปล่อยออกเมื่อฝั่ง Client ตอบกลับมามีได้รับข้อมูลนั้นแล้ว ส่วนฝั่งของ Client มีไว้รับข้อมูลที่ส่งมาไว้รอการนำออกไปใช้งานโดยคำสั่ง Receive() ในการอ่านข้อมูลจาก TCP Buffer นั้น ปริมาณของข้อมูลซึ่งอ่านในแต่ละครั้งจะขึ้นอยู่กับ 2 ปัจจัย ได้แก่

- ขนาดของ data buffer ซึ่งใส่เป็น parameter ให้กับคำสั่ง Receive()
- ขนาดของตัวแปรซึ่งใช้เป็น data buffer ซึ่งใส่เป็น parameter ให้กับคำสั่ง Receive()

ตัวอย่างโปรแกรม 2.1 ส่วนของโปรแกรมในบล็อก A ของ server

```
while(true)
{
    data = new byte[1024];
    recv = client.Receive(data);
    if (recv == 0)
        break;
    Console.WriteLine( Encoding.ASCII.GetString(data, 0, recv));
    client.Send(data, recv, SocketFlags.None);
}
```

ตัวอย่างโปรแกรม 2.2 ส่วนของโปรแกรมในบล็อก B ฝั่ง client เป็น

```
while(true)
{
    input = Console.ReadLine();
    if (input == "exit")
        break;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

server.Send(Encoding.ASCII.GetBytes(input));

data = new byte[1024];

recv = server.Receive(data);

stringData = Encoding.ASCII.GetString(data, 0, recv);

Console.WriteLine(stringData);

}

```

ผลลัพธ์ของ โปรแกรมฝั่ง Client ที่ต้องการจะเป็นดังผลลัพธ์ 2.1

ผลลัพธ์ 2.1 ผลลัพธ์ที่ฝั่ง Client

```

Welcome to my test server

Test

Test

ABCdef

ABCdef

exit

Disconnecting from server

```

ถ้ามีการกำหนด data buffer ให้มีขนาดใหญ่ ถ้าการจัดการไม่ดี กล่าวคือในการนำเอาข้อมูลมาใช้งานนั้น ไม่ได้นำเอาค่าในตัวแปร recv ซึ่งจะบอกปริมาณของข้อมูลที่อ่านจาก TCP Buffer มาใช้ประกอบ ผลลัพธ์ที่ได้ก็อาจเปลี่ยนแปลงไปจากที่ต้องการได้

สำหรับกรณีที่กำหนด data buffer ให้มีขนาดเล็กเกินไปนั้นผลลัพธ์ที่ได้ก็จะออกมาในอีกลักษณะหนึ่งตัวอย่างเช่นถ้ากำหนดให้ data buffer มีขนาดเล็กลงจาก 1024 ไบต์ เหลือเพียง 20 ไบต์ ในโปรแกรมฝั่ง client (โปรแกรม 2) โดยการเปลี่ยนแปลงคำสั่งในส่วนที่

ผลที่เกิดขึ้นคือ Data buffer มีขนาดเล็กเกินไปจนไม่เพียงพอที่จะบรรจุข้อมูลใน TCP Buffer ได้ ดังนั้น Receive() จึงนำข้อมูลออกจาก TCP Buffer เท่าที่จะสามารถบรรจุลงใน data buffer แล้วทั้งส่วนที่ไม่สามารถนำออกมาได้ไว้ใน TCP Buffer ตามเดิม เพื่อรอการเรียกใช้งานคำสั่ง Receive() ในรอบถัดไป

2.12.1.2 ปัญหาด้านการจัดการข้อมูลที่ส่งผ่านระหว่างเครือข่าย

การสื่อสารระหว่างเครือข่ายนั้น สิ่งที่จะเกิดขึ้นได้คือการสูญหายหรือความล่าช้าของข้อมูลที่ส่งจากต้นทางไปยังปลายทาง แม้ว่า TCP Protocol นั้นจะรับประกันในเรื่องของการส่งข้อมูลว่าจะไม่มีเรื่องของข้อมูลสูญหายระหว่างการรับส่ง หรือหากเกิดการสูญหายของข้อมูลขึ้น ข้อมูลชุดที่สูญหายนั้นก็จะถูกส่งใหม่ (retransmit) แต่ทั้งนี้แม้ข้อมูลจะไม่สูญหายแต่ TCP Protocol นั้นเมื่อรับข้อมูลเข้าสู่ TCP Buffer แล้วนั้นจะทำการเรียงลำดับข้อมูล แล้วรวมข้อมูลที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รับมานั้นเข้าด้วยกัน หรือกรณีที่ข้อมูลมีขนาดเล็กก็จะนำข้อมูลนั้นมาเก็บเรียงต่อกันไปภายใน TCPBuffer โดยไม่มีการทำเครื่องหมายเพื่อระบุถึงขอบเขตของข้อมูลแต่ละชุด

ตัวอย่างโปรแกรม 2.3 โปรแกรมในบล็อก A

```
for (int i = 0; i < 5; i++)
{
    recv = client.Receive(data);
    Console.WriteLine(Encoding.ASCII.GetString(data, 0, recv));
}
```

ตัวอย่างโปรแกรม 2.4 โปรแกรมในบล็อก B

```
server.Send(Encoding.ASCII.GetBytes("message 1"));
server.Send(Encoding.ASCII.GetBytes("message 2"));
server.Send(Encoding.ASCII.GetBytes("message 3"));
server.Send(Encoding.ASCII.GetBytes("message 4"));
server.Send(Encoding.ASCII.GetBytes("message 5"));
```

จากโปรแกรม 2.1 และ 2.2 ที่ทำการเติมบล็อก A และ B ด้วยส่วนของโปรแกรมจาก ตัวอย่างโปรแกรม 2.3 และ 2.4 โปรแกรม 2.1 ทำหน้าที่เป็น Server โดยจะเปิด port 9050 เพื่อรอรับข้อมูลส่วน โปรแกรม 2.2 จะทำหน้าที่เป็น client ของโปรแกรม 2.1 โดยโปรแกรม 2.1 จะรอรับข้อมูลจากโปรแกรม 2.2 ทั้งหมด 5 ครั้ง และโปรแกรม 2.2 เมื่อสามารถ connect กับโปรแกรม 2.1 ได้แล้วก็จะส่งข้อมูลให้กับโปรแกรม 2.1 ทั้งหมด 5 messages แล้ว disconnect

ผลลัพธ์ 2.2 ผลจากการทำงานของโปรแกรมในฝั่ง Client เมื่อเติมตัว บล็อก B

```
Welcome to my test server
Disconnecting from server...
```

ผลลัพธ์ 2.3 ผลจากการทำงานของโปรแกรมในฝั่ง Server ที่คาดหวังเมื่อเติมบล็อก A

```
Waiting for a client...
Connected with 161.246.xx.xx at port xxxx
message 1
message 2
message 3
message 4
message 5
Disconnecting from 161.246.xx.xx
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากนำโปรแกรมไป execute บนเครื่องอื่น/เครือข่ายอื่น หรือเมื่อสภาวะของระบบเครือข่ายเปลี่ยนไป เช่น มี traffic load มากขึ้น ผลลัพธ์ที่ได้ก็อาจเปลี่ยนไป ซึ่งผลลัพธ์ที่นำมาแสดงให้เห็นข้างต้นนี้เป็นตัวอย่างที่ดีในเรื่องที่ว่า TCP Protocol นั้นไม่มีการป้องกันขอบเขตของแต่ละชุดข้อมูล (Unprotected Message Boundary) ดังนั้นในการเขียนโปรแกรมเพื่อสื่อสารข้อมูลด้วย TCP Protocol จึงต้องออกแบบโปรแกรมให้สามารถรองรับปัญหานี้ด้วย

สำหรับวิธีการแก้ปัญหาดังกล่าวนั้น โดยทั่วไปนิยมใช้กันอยู่ 3 วิธี ได้แก่

- การจำกัดขนาดของแต่ละชุดข้อมูลที่จะส่งให้มีขนาดเท่ากัน

เป็นวิธีการแก้ปัญหาที่ง่ายที่สุด เพราะเมื่อขนาดของข้อมูลที่ส่งแต่ละชุดมีขนาดเท่ากัน โปรแกรมจะทราบได้ทันทีว่า ข้อมูลมาครบแล้วหรือยัง โดยการตรวจสอบขนาดของชุดข้อมูลนั้นนั่นเอง นอกจากนี้แม้ชุดข้อมูลมากกว่า 1 ชุดมาถึงพร้อม ๆ กันก็ยังสามารถแยกชุดข้อมูลออกจากกันได้ โดยการนับจำนวนไบต์ในข้อมูลที่รับเข้ามานั้นเทียบกับขนาดมาตรฐานของชุดข้อมูลปกติ โดยทั่วไปถ้ามีส่วนของโปรแกรมดังนี้

```
byte[] data = new byte[1024];
```

```
...
```

```
...
```

```
int sent = socket.Send(data);
```

จะทำให้คิดได้ว่าชุดของข้อมูลขนาด 1024 ไบต์จะถูกส่งไปยังปลายทางทั้งหมด แต่ในทางปฏิบัติข้อมูลอาจถูกส่งไปไม่พร้อมกันในคราวเดียว อันเนื่องมาจากสาเหตุเช่น ขนาดของ TCP Buffer และปริมาณของข้อมูลโดยรวม ซึ่งส่งออกจากเครื่องในเวลานั้น ดังนั้นมีโอกาสที่ข้อมูลอาจไม่สามารถส่งออกไปได้ทั้งหมดในคราวเดียว จึงต้องมีการตรวจสอบการทำงานของโปรแกรมว่าทำการส่งข้อมูลออกไปหมดแล้วหรือไม่ ตรวจสอบได้โดยการใช้ค่าที่ Send() return ให้กลับตัวแปร ค่าที่ return ดังกล่าวนี้อาจบอกได้ว่าได้ทำการส่งข้อมูลออกไปเท่าใด ถ้าจำนวนข้อมูลที่ส่งออกนั้นยังน้อยกว่าข้อมูลที่ต้องการส่ง ก็ให้วน loop เพื่อส่งไปเรื่อย ๆ จนกว่าจะส่งข้อมูลออกหมด

- การส่งขนาดข้อมูลไปพร้อมกันกับชุดข้อมูลนั้น

หากให้มีขนาดข้อมูลเท่ากันทุก packet จะสามารถแก้ปัญหาในเรื่องของ message boundaries ได้ แต่ข้อเสียคือ ถ้าข้อมูลที่จะส่งนั้นมีขนาดที่ยาวมากก็จะต้องแบ่งข้อมูลนั้นออกเป็นส่วน ๆ ที่เท่า ๆ ทำให้สิ้นเปลือง bandwidth ของระบบสื่อสาร เพื่อแก้ปัญหานี้จึงได้มีการคิดค้นวิธีการที่จะให้สามารถส่งชุดข้อมูลที่มีขนาดต่างกันก็ได้ แต่การจะส่งชุดข้อมูลซึ่งมี

ขนาดแตกต่างกันไปได้ นั่น ส่วนของทางฝั่งรับจะต้องทราบขนาดของชุดข้อมูลที่จะส่งไปด้วย ซึ่งก็สามารถทำได้โดยการส่งขนาดของชุดข้อมูลไปรวมไปกับชุดข้อมูลที่จะส่งนั้น

- การใส่สัญลักษณ์เพื่อบอกขอบเขตของข้อมูลแต่ละชุด

เป็นวิธีการแก้ไขปัญหเกี่ยวกับ Message boundaries วิธีสุดท้าย เป็นวิธีการแก้ปัญหาโดยการใส่สัญลักษณ์ ที่ได้กำหนดไว้ล่วงหน้าเพื่อบอกจุดสิ้นสุดของข้อมูลแต่ละชุด เมื่อรับข้อมูลเข้ามา โปรแกรมจะทำการตรวจสอบข้อมูลนั้นทีละตัวอักษรเพื่อหาสัญลักษณ์บอกจุดสิ้นสุด สำหรับวิธีการนี้จะมีข้อด้อยอยู่ 2 ประการคือ ต้องกำหนดสัญลักษณ์ที่ใช้ระบุขอบเขตของชุดข้อมูลซึ่งจะต้องไม่ซ้ำกันกับข้อมูลที่จะส่ง และการที่จะต้องตรวจสอบข้อมูลที่ได้รับเข้ามาครั้งละ 1 ตัวอักษรก็จะทำให้เกิด Overhead ขึ้น

2.13 Multithreading

2.13.1 Using Thread

การสร้าง Thread สามารถทำได้โดยการใช้คำสั่งดังนี้

```
Thread newThread = new Thread(new ThreadStart(.....));
```

โดยที่ class ThreadStart นั้นจะเป็นตัวชี้ไปยังส่วนของโปรแกรมซึ่งจะทำงานบน Thread ใหม่ที่ถูกสร้างขึ้นสำหรับ method ต่าง ๆ ที่สำคัญใน Thread Class นั้นจะแสดงไว้ในตาราง 2.1

ตารางที่ 2.1 The Thread Class Method

Method	Description
Abort()	Terminates the thread
Equals()	Determines whether two Thread objects are the same
GetHashCode()	Gets a unique representation for the thread
GetType()	Gets the type for the thread
Join()	Blocks the calling thread until the thread terminate
Resume()	Resumes a thread that has been suspended
Interrupt()	Interrupts thread that is in the Wait thread state
Start()	Causes the OS to change the thread state to Running
Suspend()	Suspends the execution of the thread
ToString()	Get a string representation of the thread

2.13.2 Using Thread Pools

Method ที่สำคัญของ Class ThreadPool จะแสดงไว้ในตาราง 2.2

ตารางที่ 2.2 The ThreadPool Static Method

Method	Description
BindHandle()	Binds an OS handle to Thread pool
GetAvailableThreads()	Gets the number of worker threads available for use in the thread pool
GetMaxThreads()	Gets the maximum number of worker threads available in the thread pool
QueueUserWorkItem()	Queues a user delegate to the thread pool
RegisterWaitForSingleObject()	Register a delegate waiting for a <i>WaitHandle</i> Object
UnsafeQueueUserWorkItem()	Queues an unsafe user delegate to the thread pool
Unsafe RegisterWaitForSingleObject()	Register an unsafe delegate waiting for a <i>WaitHandle</i> Object

การใช้งาน Thread Pool นั้นไม่จำเป็นต้องสร้าง instance ของ Class ThreadPool เนื่องจาก ThreadPool Class นั้นเป็น Static Class สามารถทำงานกับ ThreadPool ได้ดังนี้

```
ThreadPool.QueueUserWorkItem(new WaitCallback(... ..)):
```

ซึ่งจะสังเกตว่าเมื่อใส่ส่วนของโปรแกรมที่ต้องการเข้าไปใน Pool โดย method QueueUserWorkItem จะเริ่มต้นทำงานทันที

2.14 การออกแบบฐานข้อมูล

2.14.1 ระบบฐานข้อมูล (Database System)

ฐานข้อมูลถือได้ว่าเป็นแอปพลิเคชัน (Application) หรือ โปรแกรมตัวหนึ่งซึ่งทำหน้าที่ในการเก็บข้อมูลต่าง ๆ ด้วยวิธีการและรูปแบบที่เหมาะสมเพื่อให้ผู้ใช้งานสามารถเก็บข้อมูล ดูแลรักษาข้อมูล และนำข้อมูลมาใช้งานได้ง่ายกว่าการเก็บข้อมูลในรูปแบบไฟล์

ในความเป็นจริงคำว่า “ระบบฐานข้อมูล” มีความหมายแตกต่างกับคำว่า “ฐานข้อมูล” โดยระบบฐานข้อมูลจะประกอบไปด้วย 4 ส่วนหลักคือ ฐานข้อมูล (Database) , ซอร์ฟแวร์จัดการระบบฐานข้อมูล (DBMS) , โปรแกรมใช้งานฐานข้อมูล (Application Programs) และผู้ใช้งาน (USER)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.14.2 การออกแบบฐานข้อมูลด้วยอี-อาร์โมเดล

อี - อาร์โมเดลเป็นแบบจำลองข้อมูลที่ได้รับความนิยมมากในการใช้เป็นเครื่องมือสำหรับงานออกแบบฐานข้อมูล โดยอี - อาร์โมเดลจะเสนอโครงสร้างของฐานข้อมูลในระดับแนวคิดออกมาในรูปแบบของแผนภาพที่มีโครงสร้างง่ายต่อการทำความเข้าใจ ทำให้เห็นภาพรวมของเอนทิตีทั้งหมดและเห็นถึงภาพความสัมพันธ์ระหว่างเอนทิตีในระบบฐานข้อมูล

ขั้นตอนการออกแบบฐานข้อมูลด้วยอี - อาร์โมเดล ประกอบด้วยขั้นตอนต่าง ๆ ดังนี้คือ

- การศึกษารายละเอียดและลักษณะหน้าที่งานของระบบ
- การกำหนดเอนทิตีที่ควรมีในระบบฐานข้อมูล
- การกำหนดความสัมพันธ์ระหว่างเอนทิตี
- การกำหนดคุณลักษณะของเอนทิตี
- การกำหนดคีย์หลักของแต่ละเอนทิตี
- การนำสัญลักษณ์ที่ใช้ในอี-อาร์โมเดล

2.14.3 โครงสร้างของภาษาเอสคิวแอล (SQL)

ภาษา SQL (สามารถอ่านออกเสียงได้ 2 แบบ คือ “เอสคิวแอล : SQL” หรือ “ซีเควล” (Sequel)) ย่อมาจาก Structured Query Language เป็นภาษาทางด้านฐานข้อมูล ที่สามารถสร้างและ ปฏิบัติการกับฐานข้อมูลแบบสัมพันธ์โดยเฉพาะ และเป็นภาษาที่มีลักษณะคล้ายกับภาษาอังกฤษ SQL ได้ถูกนำมาพัฒนาโดยผู้ผลิตซอฟต์แวร์ด้านระบบจัดการฐานข้อมูลเชิงสัมพันธ์จนเป็นที่นิยม โดยผู้ผลิตแต่ละรายก็พัฒนาระบบจัดการฐานข้อมูลของตนให้มีลักษณะเด่นเฉพาะขึ้นมา ทำให้รูปแบบการใช้คำสั่ง SQL มีรูปแบบที่แตกต่างกันไปบ้าง ดังนั้นในปี ค.ศ. 1986 ทางด้าน American National Standards Institute (ANSI) จึงได้กำหนดมาตรฐานของ SQL ขึ้น อย่างไรก็ตาม โปรแกรมฐานข้อมูลที่ขายในท้องตลาด ได้ขยาย SQL ออกไปจนเกินข้อกำหนดของ ANSI โดยเพิ่มคุณสมบัติอื่นๆ ที่คิดว่าเป็นประโยชน์เข้าไปอีก

2.14.3.1 ประเภทของคำสั่งของภาษา SQL

ประเภทของคำสั่งในภาษา SQL แบ่งออกเป็น 3 ประเภท คือ

- ภาษาสำหรับการนิยามข้อมูล (Data Definition Language: DDL)
- ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language: DML)
- ภาษาควบคุม (Data Control Language: DCL)

2.14.3.2 ชนิดของข้อมูลที่ใช้ในภาษา SQL

ในภาษา SQL การบรรจุข้อมูลลงในคอลัมน์ต่าง ๆ ของตารางจะต้องกำหนดชนิดของข้อมูล (data type) ให้แต่ละคอลัมน์ ชนิดของข้อมูลนี้จะแสดงชนิดของค่าที่อยู่ในคอลัมน์ ค่าทุกค่าในคอลัมน์ที่กำหนดจะต้องเป็นชนิดเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิดของข้อมูลของแต่ละคอลัมน์จะขึ้นกับลักษณะของข้อมูลแต่ละคอลัมน์ ซึ่งแบ่งได้ดังนี้ชนิดข้อมูลพื้นฐานในภาษา SQL ดังนี้

ตัวหนังสือ (character) ในภาษา SQL จะใช้

- ตัวหนังสือแบบความยาวคงที่ (fixed-length character) จะใช้ char (n) หรือ character (n) ชนิดของข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุดได้ 255 ตัวอักษร

- ตัวหนังสือแบบความยาวไม่คงที่ (variable-length character) จะใช้ varchar (n) แทนประเภทของข้อมูลที่เป็นตัวหนังสือใดๆที่มีความยาวของข้อมูลไม่คงที่ ข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุดได้ 4000 ตัวอักษร

จำนวนเลข (numeric)

- จำนวนเลขที่มีจุดทศนิยม (decimal) ในภาษา SQL จะใช้ dec (m, n) หรือ decimal (m, n) เป็นประเภทข้อมูลที่เป็นจำนวนเลขที่มีจุดทศนิยม

- จำนวนเลขที่ไม่มีจุดทศนิยมในภาษา SQL จะใช้ int หรือ integer เป็นเลขจำนวนเต็มบวกหรือลบขนาดใหญ่ เป็นตัวเลข 10 หลัก และในภาษา SQL จะใช้ smallint มีค่าตั้งแต่ - 32,768 ถึง + 32,767 ตัวเลขจำนวนเต็มประเภทนี้จะมีการจองเนื้อที่น้อยกว่าแบบ integer

- เลขจำนวนจริง ในภาษา SQL อาจใช้ number (n) แทนจำนวนเลขที่ไม่มีจุดทศนิยมและจำนวนเลขที่มีจุดทศนิยม

ข้อมูลในลักษณะอื่นๆ

- วันที่และเวลา (Date/Time) ซึ่งจะมีหลายรูปแบบให้เลือกใช้

2.14.4 ลักษณะการใช้งานของภาษา SQL

ภาษา SQL เป็นส่วนประกอบหนึ่งของ DBMS มักพบใน DBMS เริงสัมพันธ์หลายตัวและเป็นที่ยอมรับในปัจจุบัน ภาษา SQL ง่ายต่อการเรียนรู้ การใช้งานในภาษา SQL แบ่งเป็น 2 ลักษณะ คือ ภาษา SQL ที่โต้ตอบได้ (interactive SQL) และภาษา SQL ที่ฝังในโปรแกรม (embedded SQL)

2.14.4.1 ภาษา SQL ที่โต้ตอบได้ ใช้เพื่อปฏิบัติงานกับฐานข้อมูลโดยตรง เป็นการใช้คำสั่งภาษา SQL สั่งงานบนจอภาพ โดยเรียกดูข้อมูลได้โดยตรงในขณะที่ทำงาน เพื่อให้ได้ผลลัพธ์ที่นำไปใช้ได้ ตัวอย่างเช่น ต้องการเรียกดูข้อมูลในคอลัมน์ SALENAME และ SALECOM จากตาราง SALESTAB จะใช้คำสั่งของภาษา SQL ดังนี้

```
SELECT SALENAME, SALECOM
FROM SALESTAB;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.14.4.2 ภาษา SQL ที่ฝังในโปรแกรม เป็นภาษา SQL ที่ประกอบด้วยคำสั่งต่าง ๆ ของภาษา SQL ที่ใส่ไว้ในโปรแกรมที่ส่วนมากแล้วเขียนด้วยภาษาอื่น ลักษณะของคำสั่ง SQL จะแตกต่างจากภาษาอื่นๆ ในแง่ที่ว่า SQL ไม่มีคำสั่งที่เกี่ยวข้องกับการควบคุม (control statement) เหมือนภาษาอื่น เช่น if..then...else for...do หรือ loop หรือ while ทำให้มีข้อจำกัดในการเขียนชุดคำสั่งงาน การใช้ภาษา SQL ฝังในโปรแกรมอื่นจะทำให้ภาษา SQL มีความสามารถ และมีประสิทธิภาพมากยิ่งขึ้น

2.14.5 การบันทึกข้อมูล ,การปรับปรุงข้อมูล และการลบข้อมูล

ในระบบฐานข้อมูล การบันทึกข้อมูล การปรับปรุงข้อมูลและการลบข้อมูลถือเป็นสิ่งสำคัญ ในภาษา SQL มีภาษาสำหรับการจัดการข้อมูล (Data manipulation Language : DML) ซึ่งเป็นภาษาที่ใช้ในการบันทึกข้อมูล การปรับปรุงข้อมูลและการลบข้อมูล คำสั่งในภาษาสำหรับการจัดการข้อมูลจะเป็นคำสั่งที่ช่วยในการจัดการข้อมูลภายในโครงสร้างตารางที่สร้างขึ้น

คำสั่งที่ใช้ในการปรับปรุงข้อมูลของภาษา SQL คือ การเพิ่มข้อมูล (INSERT) การปรับปรุงข้อมูล (UPDATE) และ การลบข้อมูล (DELETE) เป็นคำสั่งในภาษาการจัดการข้อมูล เมื่อโครงสร้างหลักของตารางได้ถูกกำหนดขึ้นเรียบร้อยแล้ว ก็จะทำกรบันทึกข้อมูลลงในตารางหลักหรืออาจทำการปรับปรุง หรือลบข้อมูลในภายหลัง คำสั่งทั้ง 3 นี้ เมื่อดำเนินการในภาษา SQL จะไม่แสดงผลลัพธ์ออกมาทางหน้าจอ แต่ผลของคำสั่งจะมีผลต่อข้อมูล ผู้ใช้สามารถดูผลของการใช้คำสั่งในการเพิ่มข้อมูล การปรับปรุงและการลบข้อมูล โดยใช้คำสั่งการเรียกค้นข้อมูล (SELECT)

2.14.5.1 คำสั่งการเพิ่มข้อมูล (INSERT)

คำสั่งการเพิ่มข้อมูลในตารางจะใช้คำสั่ง INSERT จะมีอยู่ 2 รูปแบบคือ การเพิ่มข้อมูลเข้าไปทีละแถว และ การเพิ่มข้อมูลโดยการดึงกลุ่มข้อมูลด้วยคำสั่งค้นหาข้อมูล

- คำสั่งการเพิ่มข้อมูลที่ละแถวโดยระบุข้อมูลที่จะ INSERT เข้าไปโดยตรงรูปแบบของคำสั่งเป็นดังนี้

```
INSERT INTO <Table name> [(column 1, column 2,...)]
VALUE (<valu1, value2 ,...>);
```

INSERT INTO เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล
Table_name ชื่อตารางที่จะเพิ่มข้อมูล
column 1, column 2,... คอลัมน์ที่ต้องการเพิ่มข้อมูล
value1,value2, ค่าข้อมูลของแต่ละคอลัมน์ที่ต้องการเพิ่ม

- คำสั่งการเพิ่มข้อมูลโดยการดึงกลุ่มข้อมูลด้วยคำสั่งค้นหาข้อมูล สามารถใช้คำสั่ง INSERT ในการนำค่าหรือหาค่าจากตารางหนึ่งแล้วไปใส่ไว้ในอีกตารางหนึ่งได้ โดยได้ค่านั้นมาจากการสอบถามข้อมูล รูปแบบเป็นดังนี้

```
INSERT INTO <Table_name>[(column 1, column 2,...)]
SELECT statement;
```

INSERT INTO เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล

Table_name ชื่อตารางที่จะเพิ่มข้อมูล

SELECT statement เป็นคำสั่ง SELECT ที่ต้องการข้อมูลอีกตารางหนึ่ง

2.14.5.2 คำสั่งปรับปรุงแถวข้อมูล (UPDATE)

หลังจากที่ป้อนข้อมูลเข้าไปเก็บไว้ในตารางแล้ว กรณีที่ต้องการปรับปรุงแก้ไขข้อมูลสามารถทำได้ด้วยภาษา SQL การปรับปรุงแถวข้อมูลเป็นการปรับปรุงหรือแก้ไขค่าคอลัมน์ ซึ่งในคำสั่งปรับปรุงข้อมูลอาจมีมากกว่า 1 คอลัมน์ในแถวทุกแถวที่มีเงื่อนไขสอดคล้องกับที่ระบุไว้หลังคำว่า WHERE รูปแบบของคำสั่งปรับปรุงแถวข้อมูลมีดังนี้

```
UPDATE <table name> SET <column 1>[, column 2,...] = <expression
|sunquery>
[WHERE<condition>];
```

UPDATE เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการปรับปรุงข้อมูล

Table_name ชื่อตารางที่ต้องการปรับปรุง

SET <column > ชื่อคอลัมน์ที่ต้องการปรับปรุง

expression ค่าข้อมูลที่ต้องการปรับปรุง

WHERE<condition> เงื่อนไขในการปรับปรุง

2.14.5.3 คำสั่งการลบข้อมูลทั้งแถว (DELETE)

คำสั่งในการลบแถวข้อมูล เป็นคำสั่งที่ใช้ในการลบแถวข้อมูลทุกแถวที่มีเงื่อนไขสอดคล้องกับที่ระบุไว้หลัง WHERE คำสั่งการลบข้อมูลมีรูปแบบทั่วไปดังนี้

```
DELETE FROM <Table_name>
[WHERE<condition>];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DELETE FROM เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการลบข้อมูล
 Table_name ชื่อตารางที่ต้องการลบข้อมูล
 WHERE<condition> เงื่อนไขในการลบข้อมูล

2.14.6 การเรียกค้นข้อมูล (SELECT)

การจัดทำฐานข้อมูลให้อยู่ในรูปแบบตารางนั้น เกิดจากการที่ข้อมูลได้ออกแบบมาเพื่อลดความซ้ำซ้อน (Normalization) ดังนั้นข้อมูลที่มีรายละเอียดของข้อมูลมากอาจจะถูกเก็บไว้ในหลายๆตารางแยกออกมาต่างหาก การแยกออกเป็นตารางข้อมูลย่อยๆนั้นนอกจากลดความซ้ำซ้อนแล้วยังช่วยในการประหยัดเนื้อที่ และยังเพิ่มประสิทธิภาพของฐานข้อมูล

- การเรียกค้นข้อมูลจากหลายตารางของภาษา SQL เป็นการกำหนดความสัมพันธ์ระหว่างตารางทั้งหลาย โดยสามารถเอาข้อมูลในตารางที่ตารางก็ได้ให้มาสัมพันธ์กัน ดังนั้นจึงสามารถเชื่อมต่อข้อมูลที่แตกต่างกันได้โดยการใช้คำสั่ง WHERE คำสั่ง WHERE เป็นคำสั่งในการกำหนดเงื่อนไขในการเรียกดูข้อมูลใช้คู่กับคำสั่ง SELECT และ FROM

```
SELECT * FROM TABLE1, TABLE2
```

- การเรียกดูข้อมูลแบบซ้อนกัน (subqueries) เป็นการสร้างคำสั่ง SELECT ซ้อนกัน การเรียกดูข้อมูลแบบซ้อนกันนั้นทำเพื่อ จุดประสงค์เพื่อลดภาระในการเชื่อมตารางที่ต้องใช้ในหน่วยความจำเป็นจำนวนมาก คำสั่งย่อยนี้สามารถสร้างหลังคำสั่ง WHERE มีรูปแบบดังนี้

```
SELECT [*] <column 1, column 2,...>
```

```
FROM <table name>
```

```
[WHERE<column list = <Select Statement>]
```

SELECT คำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล

column 1, column 2,... คอลัมน์ที่ต้องการเรียกค้น

FROM การกำหนดว่าให้เรียกดูข้อมูลได้จากตารางใดบ้าง

table name ชื่อตารางที่ต้องการเรียกค้นข้อมูล

WHERE<condition> ส่วนของคำสั่งที่บอกเงื่อนไขที่จะใช้ในการค้นหาข้อมูล

Select Statement ส่วนของคำสั่งที่เรียกค้นข้อมูลตามเงื่อนไข

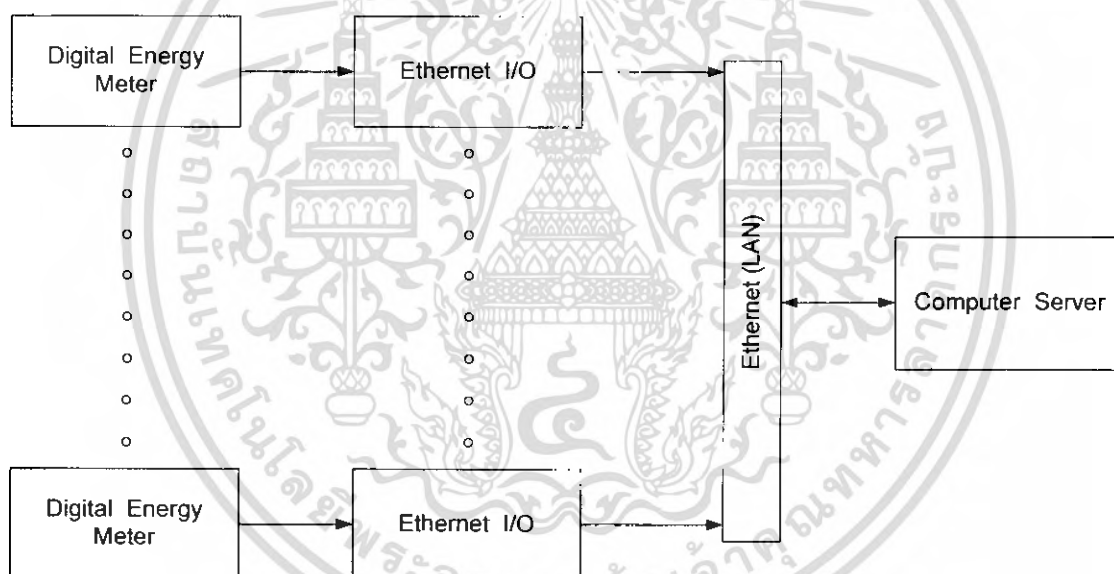
การทำงานของคำสั่งย่อยที่ใช้ในการระบุเงื่อนไข หรือเรียกข้อมูลจะทำจากคำถามย่อยด้านในสุดผลที่ได้จะเป็นค่ากลับมาให้กับค่าที่อยู่หน้าเครื่องหมายเท่ากับ เพื่อเรียกค้นข้อมูล (SELECT) ตามต้องการ

บทที่ 3

การดำเนินงานวิจัย

3.1 โครงสร้างโดยรวมของระบบ

โครงสร้างโดยรวมของระบบจะประกอบไปด้วย 3 ส่วนคือ Digital Energy Meter ซึ่งเป็นส่วนวัดค่าพลังงานที่ใช้ไป จากนั้นนำค่าพลังงานที่ได้มาส่งผ่านไปยังส่วน Ethernet Module โดยตัว Ethernet Module ทำหน้าที่รับสัญญาณจาก Digital Energy Meter โดยผ่าน RS-232 แล้วทำการแปลงสัญญาณให้อยู่ในรูปแบบของ Ethernet Protocol ส่งผ่านออกไปทางพอร์ต JR-45 ไปยัง Computer ซึ่งทำหน้าที่เป็นตัว Server เก็บค่าพลังงานไฟฟ้าและข้อมูลต่างๆ

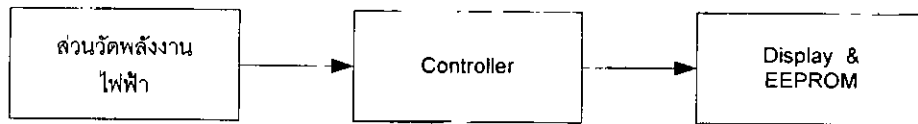


รูปที่ 3.1 โครงสร้างโดยรวมของระบบ

3.2 โครงสร้าง Digital Energy Meter

โครงสร้างของ Digital Energy Meter แบ่งออกเป็น 3 ส่วนคือ ส่วนวัดค่าพลังงานไฟฟ้าจะใช้ ADE Chip เป็นตัววัดค่าพลังงานไฟฟ้าที่ใช้ จากนั้นทำการส่งค่าที่ได้ไปยังส่วนที่สองซึ่งเป็นส่วนของ Controller ส่วนนี้จะรับค่าพลังงานจากส่วนแรกและทำหน้าที่ในการควบคุมการทำงานว่าตัว Digital Energy Meter ทั้งหมด โดยจะส่งข้อมูลที่ได้ออกไปแสดงที่หน้าจอ LCD จะทำการเก็บข้อมูลไปยัง EEPROM เมื่อเกิดไฟฟ้าดับ และจะทำการส่งข้อมูลออก RS-232

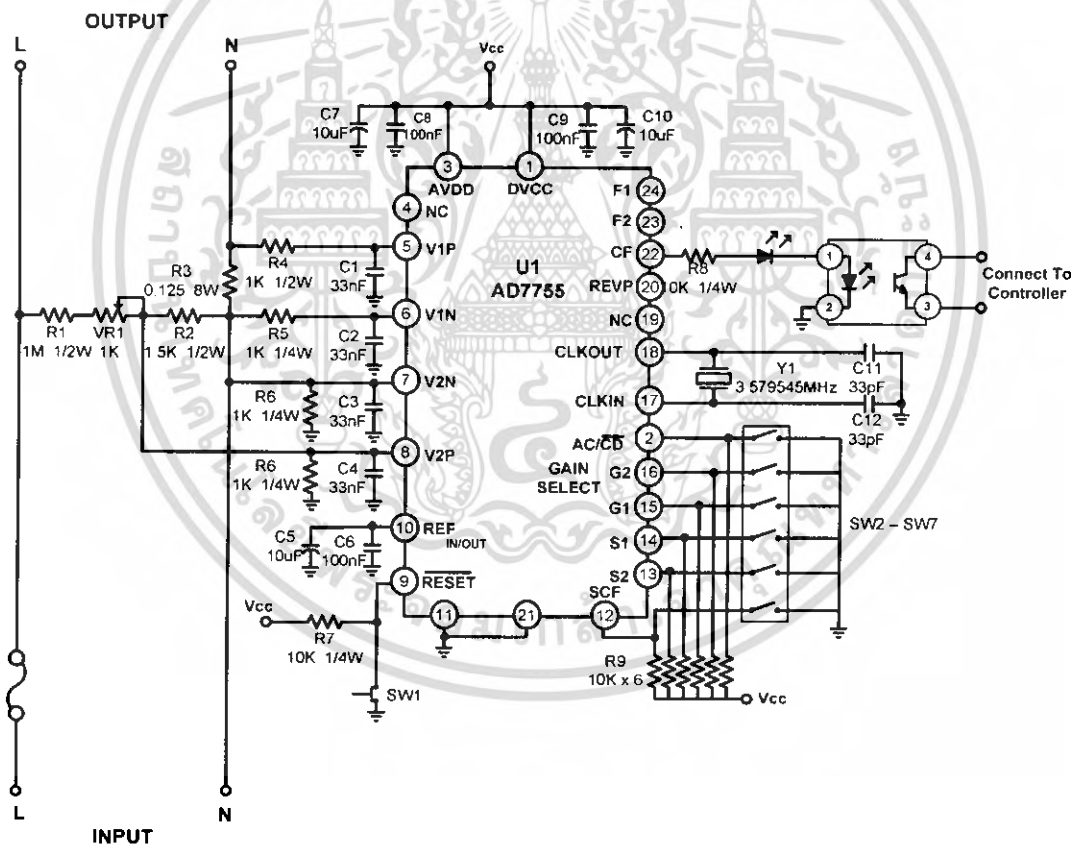
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 โครงสร้างของ Digital Energy Meter

3.3 วงจรส่วนวัดพลังงานของ Digital Energy Meter

ส่วนวัดพลังงานของ Digital Energy Meter จะใช้ IC ADE 7755 ในการวัดค่าพลังงาน โดยจะป้อนสัญญาณอินพุตที่เป็นกระแสโดยใช้เซ็นเซอร์เข้าที่ขาเชลแนล วี 1 (ขา 5 และขา 6) โดยจะมีค่าความต่างศักย์ของสัญญาณอนาล็อกอินพุตไม่เกิน 470 มิลลิโวลต์ ค่าแรงดันนั้นจะมากขึ้นอยู่กับการใช้พลังงานไฟฟ้า ในส่วนของขาเชลแนล วี 2 ของ ADE 7755 (ขา 7 และขา 8) จะใช้ วงจร โวลต์เตจดีไวเดอร์ในการลดระดับแรงดันลงจาก 220 โวลต์ ให้ไม่เกิน 660 มิลลิโวลต์



รูปที่ 3.3 วงจรส่วนวัดพลังงานของ Digital Energy Meter

IC ADE 7755 ทำการรับค่ากระแสไฟฟ้าและแรงดันไฟฟ้าจากเชลแนล วี 1 (ซึ่งสามารถปรับค่า Gain ของเชลแนล วี 1 ได้ว่าจะให้รับค่ากระแสไฟฟ้าได้เท่าไรด้วยสวิตช์ SW3 และ SW4) และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แชลแนล วี2 สัญญาณที่ป้อนมาจะถูกแปลงเป็นดิจิตอลและทำการจะนำค่าทั้ง 2 ค่าที่ได้มาคูณกัน ซึ่งจะ ทำให้ได้ค่าของพลังงานไฟฟ้าออกมา จากนั้น ADE 7755 ได้ทำการแปลงค่าพลังงานไฟฟ้าที่เป็น ดิจิตอลให้อยู่ในรูปของความถี่ ด้วยวงจรแปลงสัญญาณดิจิตอลเป็นความถี่ ซึ่งจะ ทำให้ได้ค่าพลังงาน ไฟฟ้าที่ออกมาจาก ADE 7755 ออกมาในรูปของพัลส์ที่มีความถี่ขึ้นกับพลังงานไฟฟ้า ความถี่ที่ออกมา สามารถปรับให้มีความถี่ที่ละเอียดหรือหยาบได้โดยสวิตช์ SW4 และ SW5

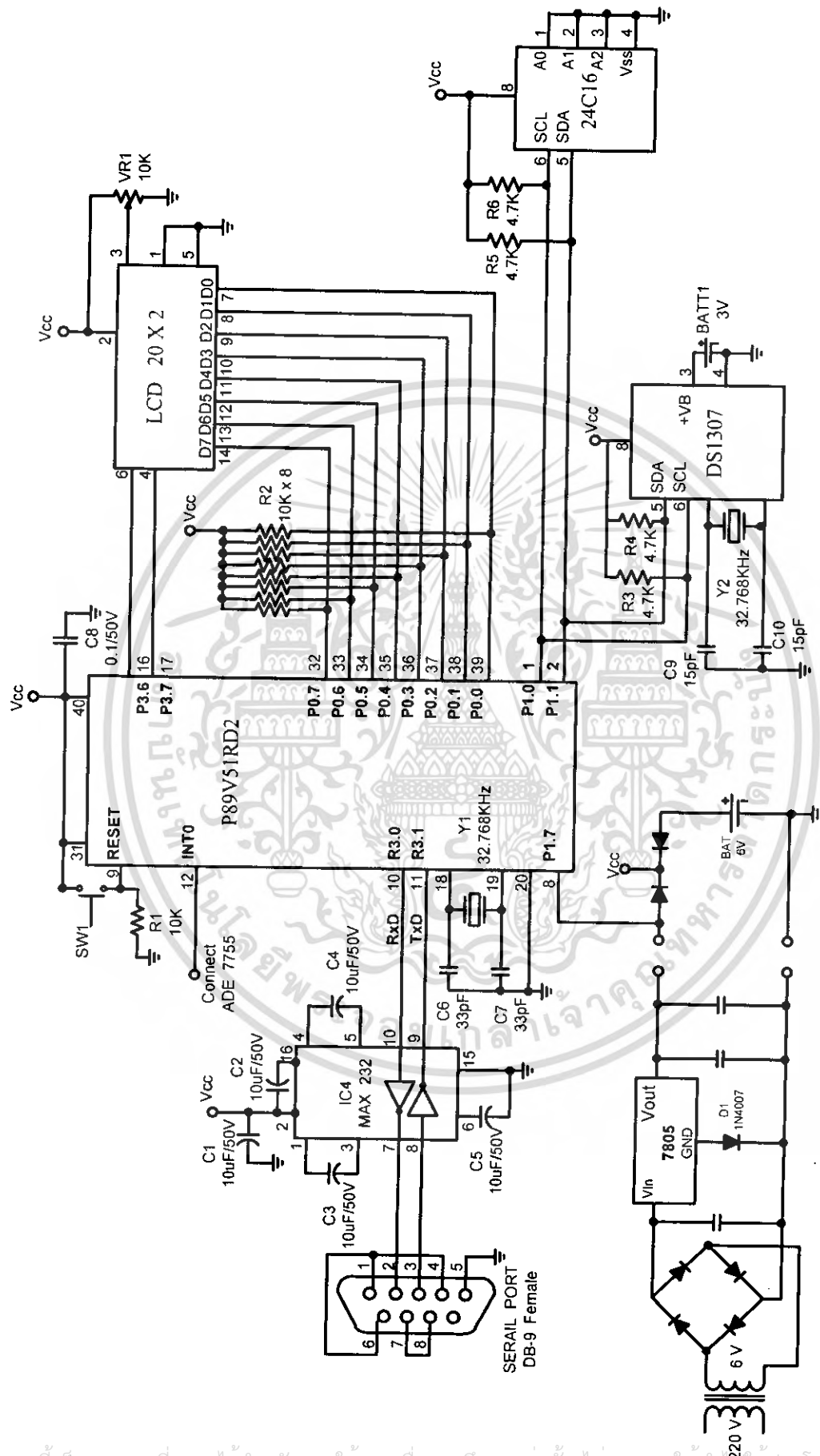
3.4 วงจรส่วน Controller ของ Digital Energy Meter

ส่วน Controller ของ Digital Energy Meter โดยในที่นี้ใช้ P89V51RD2 เป็นตัว Controller ที่ใช้ในการควบคุมการทำงานของ Digital Energy Meter ซึ่งจะทำการต่อกับสัญญาณ Output ของ IC ADE 7755 เข้ากับตัว P89V51RD2 ผ่านทางขา INTO เพื่อนำค่าพัลส์ที่ได้จากตัว IC ADE 7755 มาทำการคำนวณและแสดงผลออกเป็นค่าพลังงานที่ได้ใช้ไป และใช้ P89V51RD2 ในการควบคุมการ ทำงานของตัว Digital Energy Meter

ซึ่งตัว P89V51RD2 จะทำการต่อกับ IC DS1307 ซึ่งเป็น IC Real – Time Clock ใช้ในการ ตั้งเวลาให้กับตัว Digital Energy Meter ซึ่งในการเชื่อมต่อกับ IC DS1307 จะเชื่อมต่อแบบ I²C เข้า กับพอร์ท P1.0 และ P1.1 ของ P89V51RD2 เข้ากับส่วนของ SCL และ SDA ของตัว IC DS1307 Digital Energy Meter ที่สร้างขึ้นจะแสดงผลค่าพลังงานผ่านทางจอ LCD ขนาด 20 ตัวอักษร 2 แถว โดยควบคุมผ่านทางพอร์ท P0.0 – P0.7 และมีการเชื่อมต่อเข้ากับตัว EEPROM ในที่นี้ใช้ 24C16 ในการเชื่อมต่อกับ EEPROM เชื่อมต่อแบบ I²C เช่นเดียวกับตัว IC Real – Time Clock ซึ่งตัว EEPROM นั้นมีเพื่อไว้เก็บข้อมูลต่างๆ เมื่อเกิดเหตุการณ์ไฟดับขึ้น โดยตัว P89V51RD2 จะทำการ ตรวจสอบว่าไฟดับหรือไม่ด้วยการต่อสัญญาณของ Vcc ผ่านทาง P1.7 หากเกิดไฟดับขึ้นสัญญาณที่ เข้ามายัง P1.7 จะเป็น 0 ทำให้ตัว Controller ทำการเก็บข้อมูลพลังงานไฟฟ้าที่เข้าไปยัง EEPROM ที่ทำการเชื่อมต่อไว้

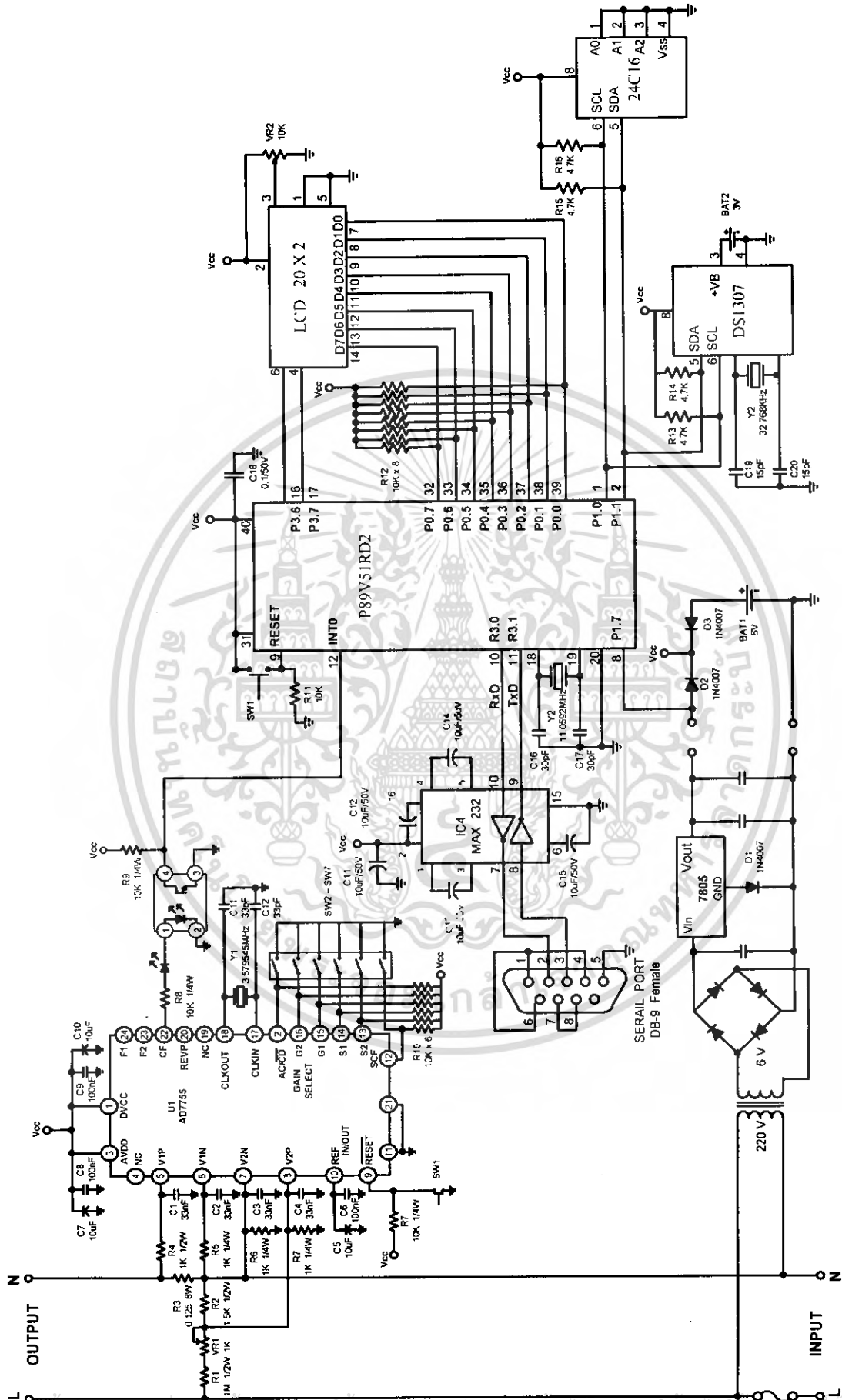
ในการใช้งานตัว Digital Energy Meter นั้น หากมีเกิดเหตุการณ์ไฟดับขึ้นตัว Digital Energy Meter จะทำการบันทึกค่าพลังงานที่ตรวจจับได้ลงที่ EEPROM โดยมีตัวแหล่งจ่ายสำรองไฟ ที่ต่อกับ Battery 6 โวลต์ เอาไว้ และเมื่อไฟฟ้ามามีตัว Controller ก็จะทำการดึงค่าจากตัว EEPROM ที่เก็บไว้จากนั้นทำการนับค่าพลังงานต่อจากค่าเดิม

โดยในการทำงานของตัว Digital Energy Meter นั้น จะมีการส่งผ่านข้อมูลไปยังเครื่อง คอมพิวเตอร์ตลอดเวลา โดยจะทำการส่งข้อมูลผ่านทางขา Tx และ Rx จะส่งข้อมูลออกไปทาง Serial Port (RS 232) โดยสัญญาณที่ได้จะทำการผ่านตัว Ethernet Module ซึ่งทำหน้าที่ในการเปลี่ยน สัญญาณที่รับมาจาก RS 232 ให้อยู่ในรูปของ Protocol ที่ใช้กับระบบ LAN ออกมาเป็น RJ-45



รูปที่ 3.4 วงจรส่วน Controller ของ Digital Energy Meter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

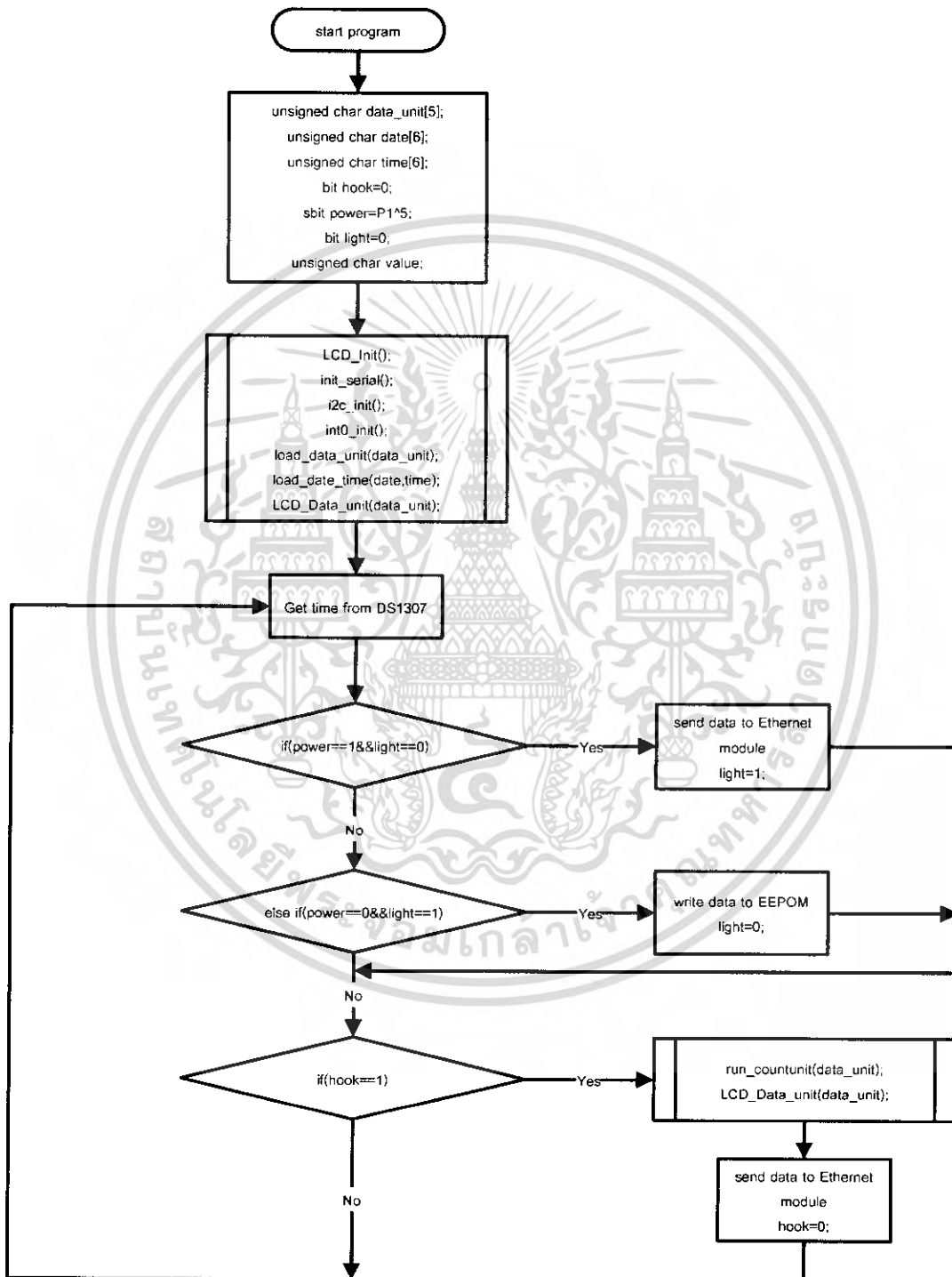


รูปที่ 3.5 วงจร Digital Energy Meter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

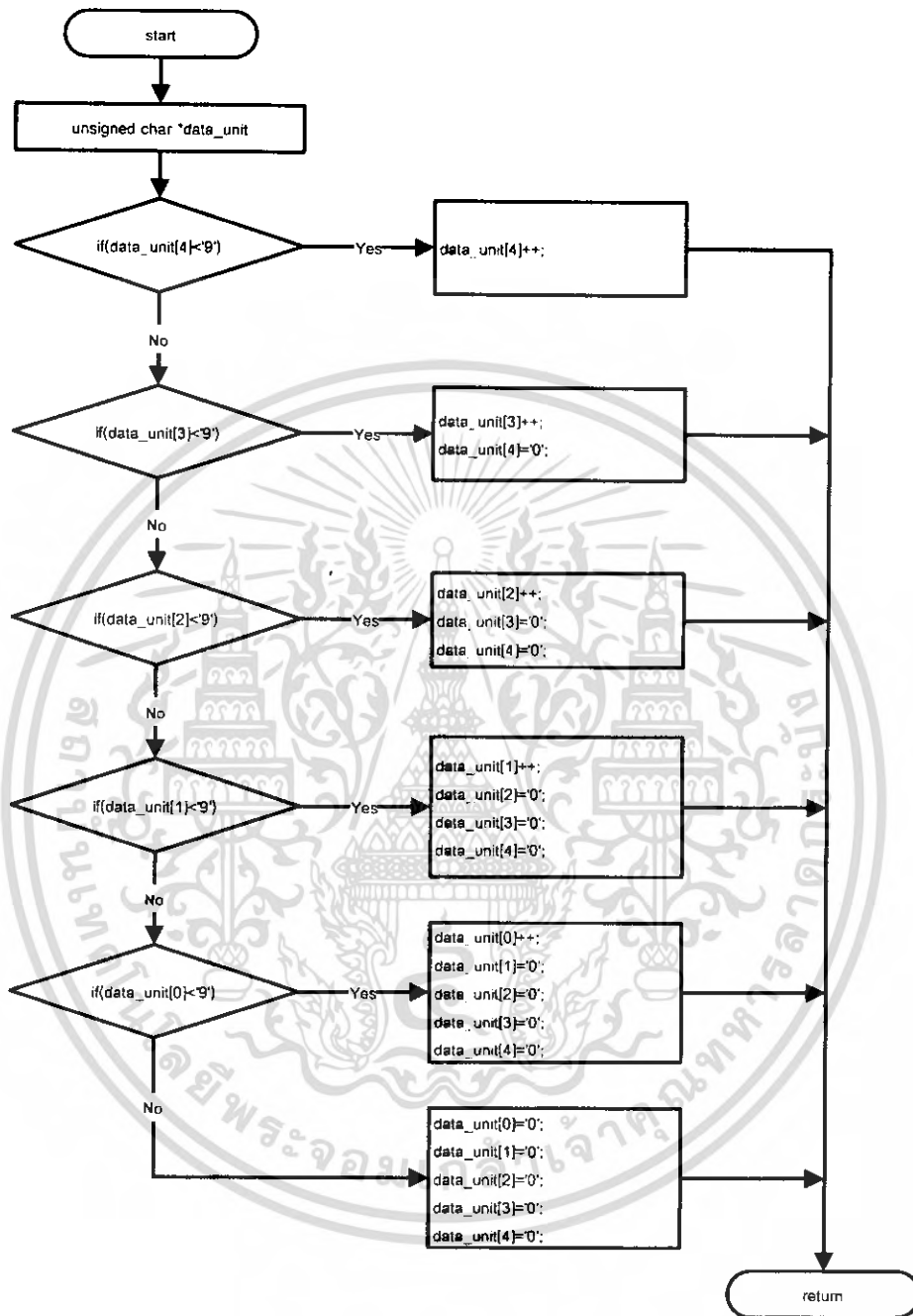
3.5 Flowchart ส่วน Digital Energy Meter

3.5.1 Flowchart โปรแกรมหลักทำหน้าที่ควบคุมการทำงานของ Digital Energy Meter



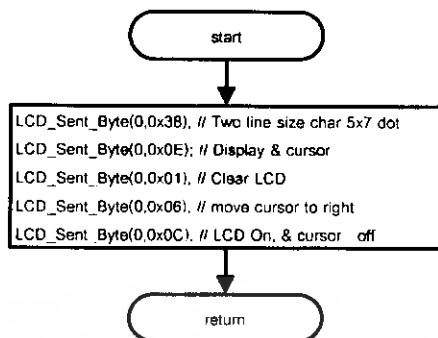
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.2 Flowchart โปรแกรมย่อย LCD Data_unit()

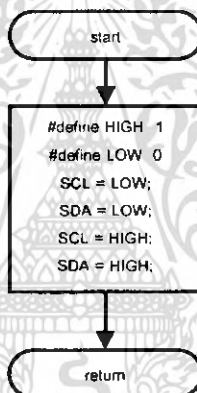


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

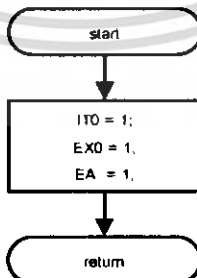
3.5.3 Flowchart โปรแกรมย่อย LCD init()



3.5.4 Flowchart โปรแกรมย่อย i2c_init()

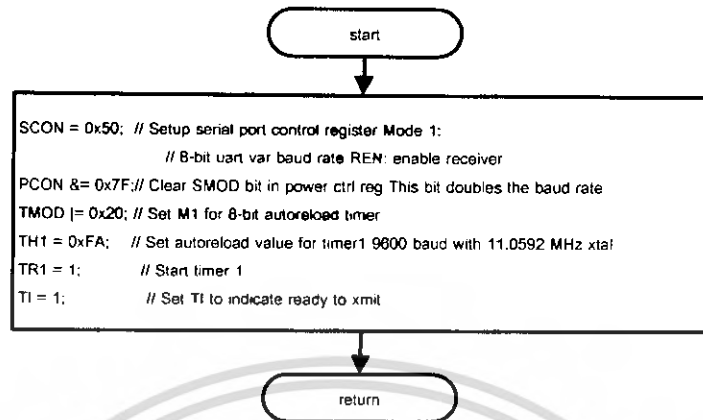


3.5.5 Flowchart โปรแกรมย่อย int0_init()

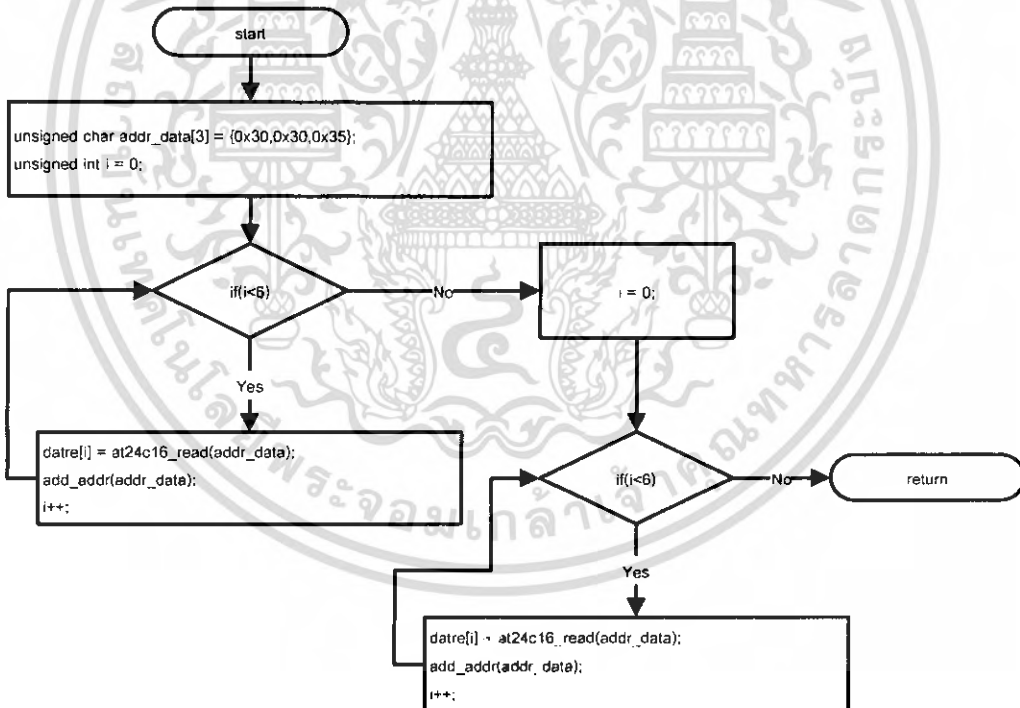


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

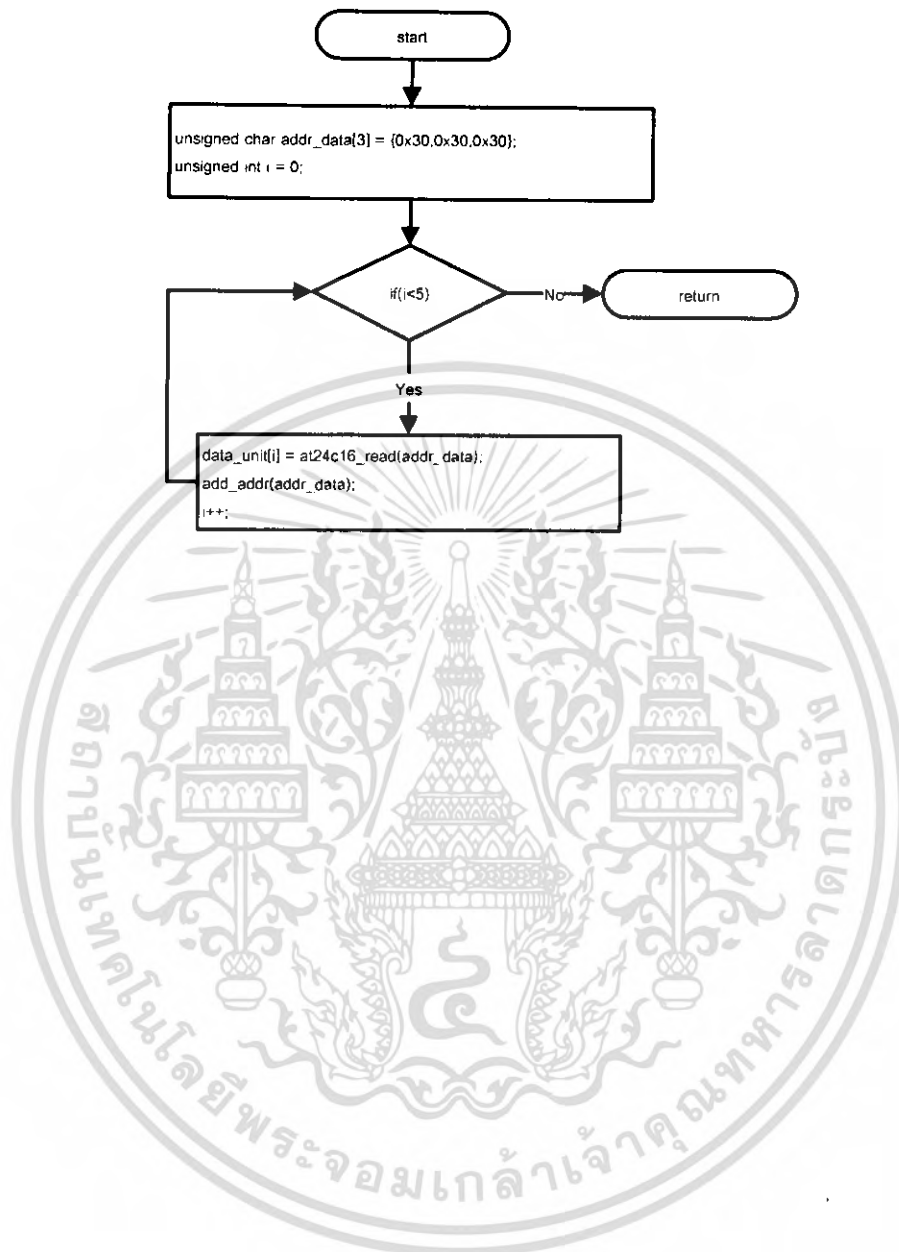
3.5.6 Flowchart โปรแกรมย่อย init_serial()



3.5.7 Flowchart โปรแกรมย่อยอ่าน EEPROM

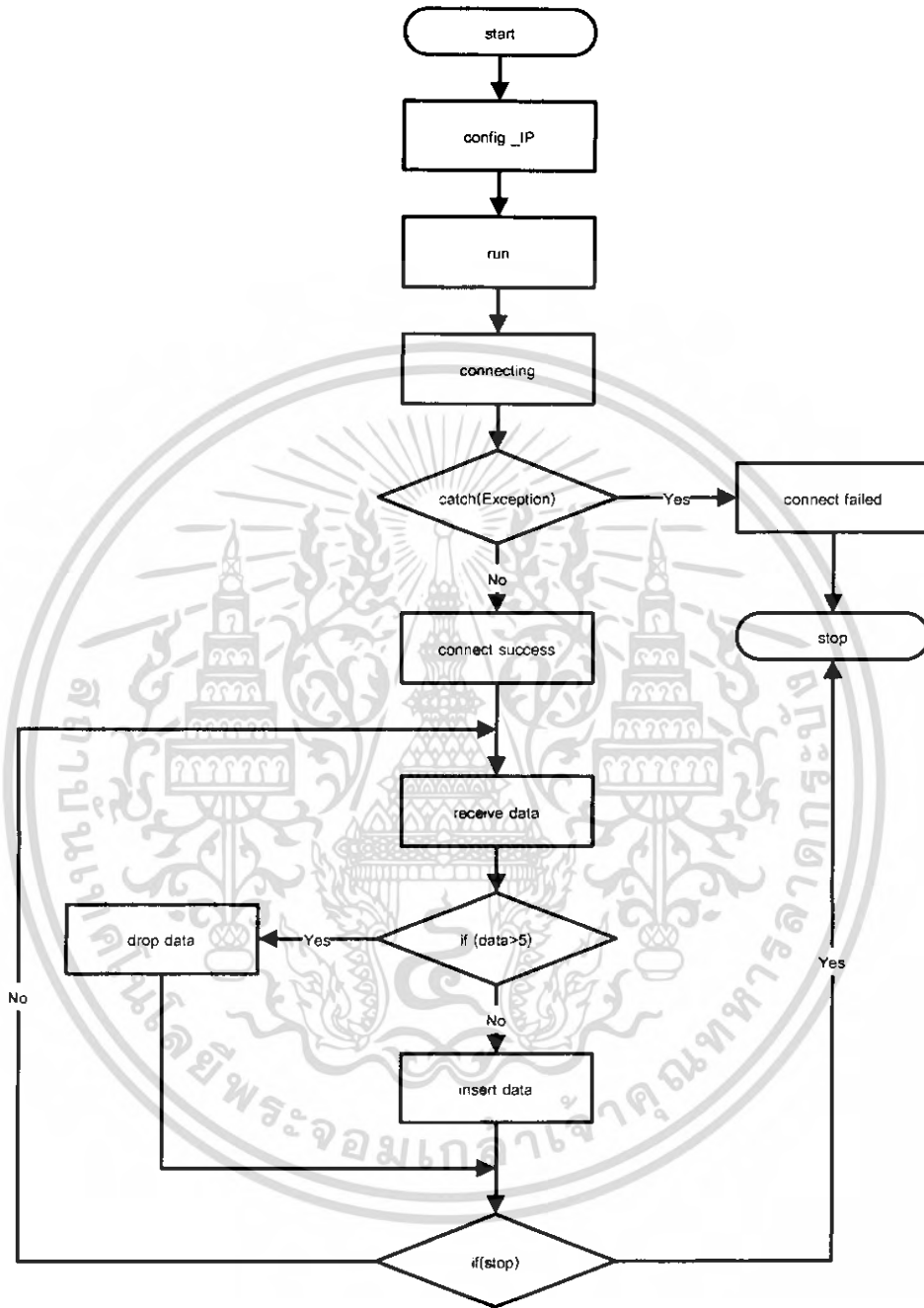


3.5.8 Flowchart โปรแกรมข่อยอ่าน EEPROM



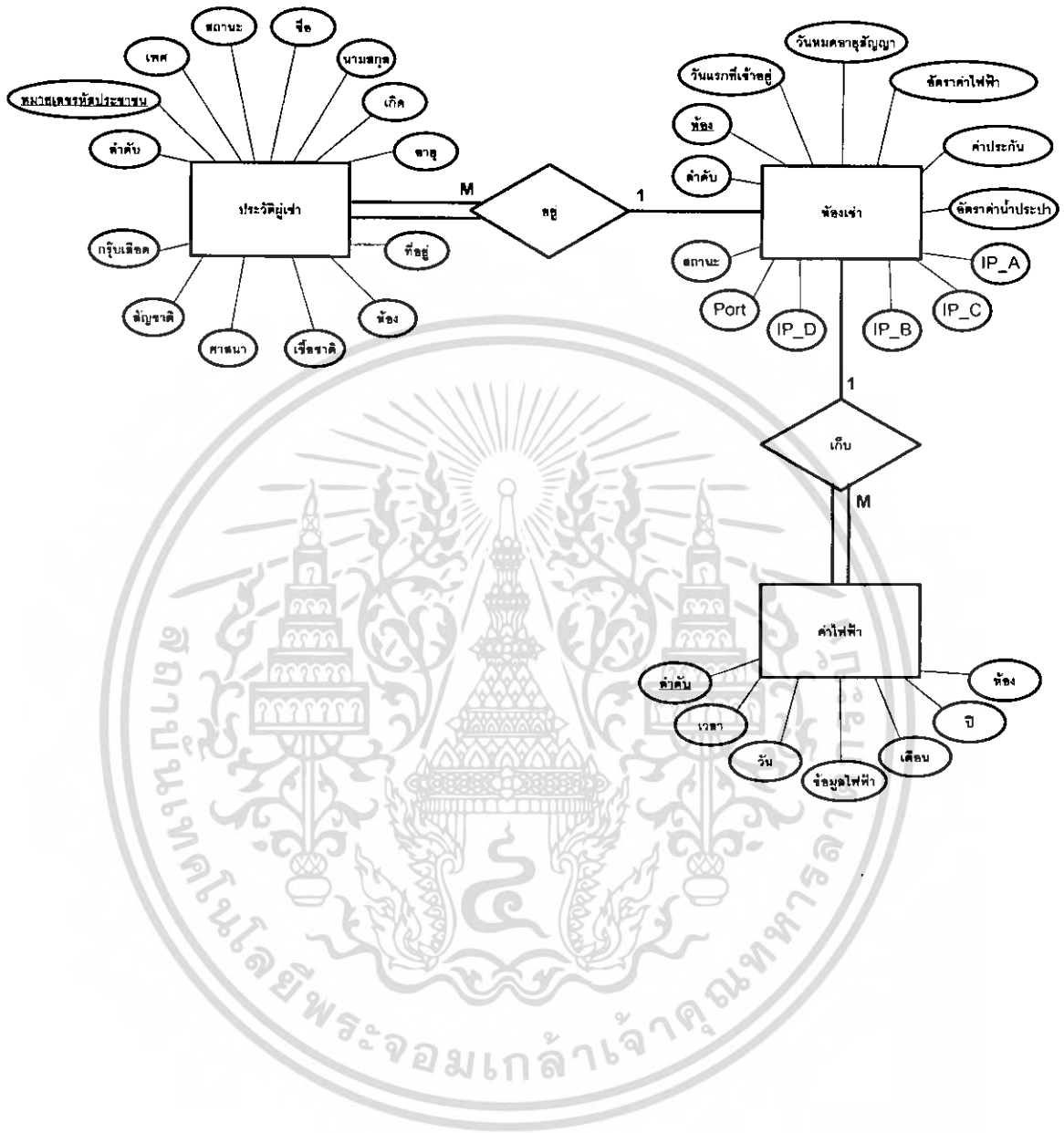
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 Flowchart ส่วน Application



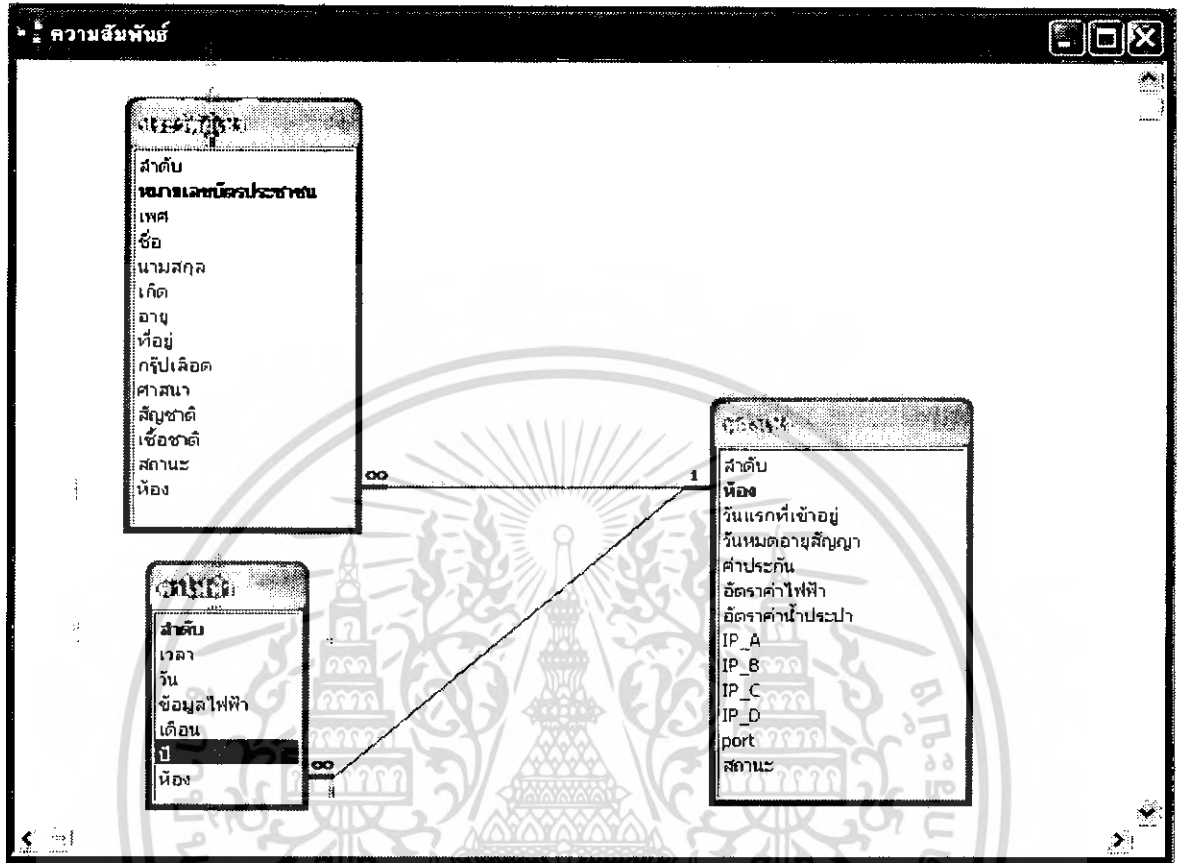
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 ER – Model



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8 Relationship Model



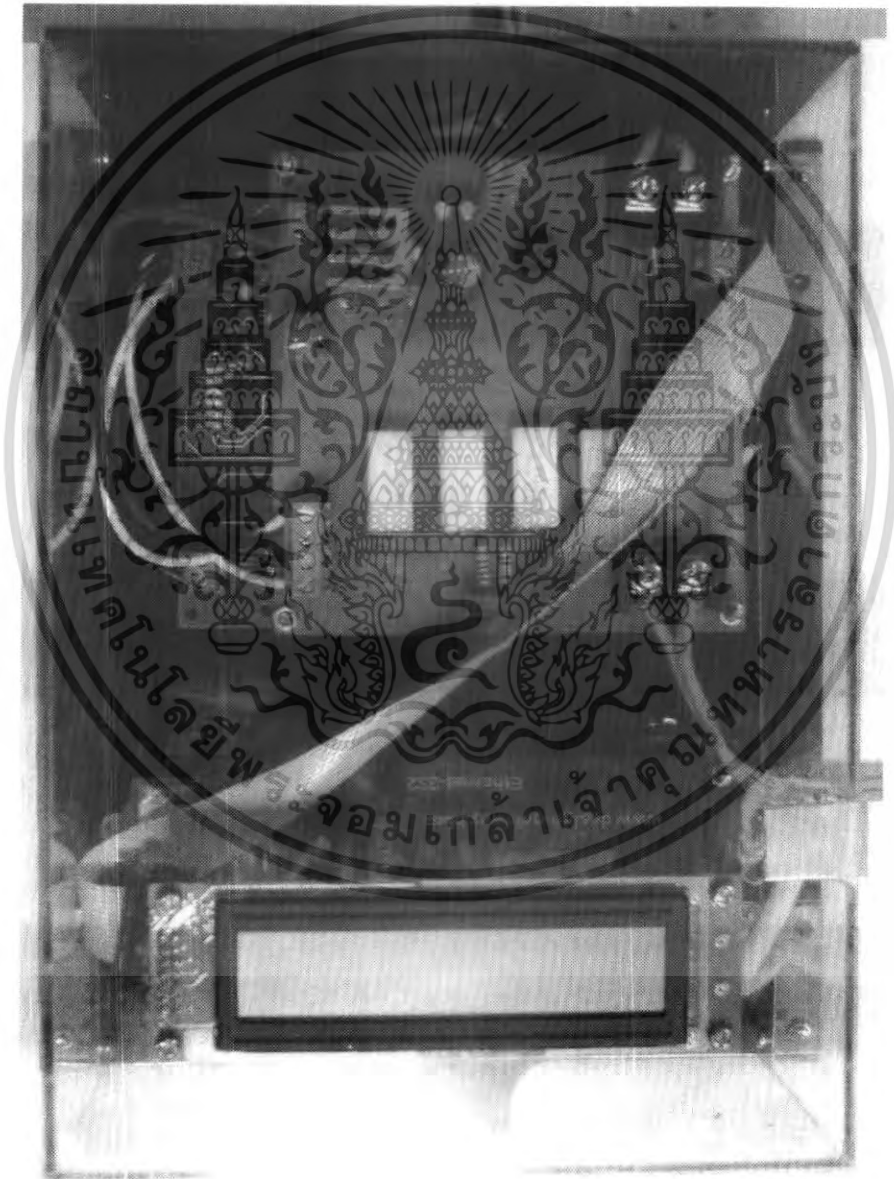
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 ส่วนของวงจร Digital Energy Meter

4.1.1 Digital Energy Meter



รูปที่ 4.1 Digital Energy Meter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ผลการทดลองที่วัดใน 1 วัน ที่โหลดสม่ำเสมอ

ตารางที่ 4.1 ผลการทดลองที่โหลดสม่ำเสมอโดยใช้หลอดไฟ 100 W

ชั่วโมง	Digital Energy Meter (kWh)	Entergy Meter Digital (kWh)
2	0.20	0.20
4	0.40	0.40
6	0.60	0.60
8	0.80	0.80
10	1.00	1.00
12	1.21	1.20
14	1.41	1.40
16	1.61	1.60
18	1.91	1.80
20	2.11	2.10
22	2.31	2.30
24	2.52	2.50

ค่าไฟฟ้าของหลอดไฟขนาด 100 วัตต์

$$\begin{aligned} \text{จำนวนหน่วยไฟฟ้าที่ใช้} &= \frac{(\text{จำนวนวัตต์ของเครื่องใช้ไฟฟ้า} \times \text{จำนวนชั่วโมงที่ใช้งาน})}{1,000 \text{ วัตต์}} \\ &= \frac{100 \times 1}{1000} = 0.1 \end{aligned}$$

หลอดไฟ 100 วัตต์ 1 ชั่วโมงใช้ไฟไป 0.1 kW

ค่าความผิดพลาด

$$\frac{|2.5 - 2.52|}{2.5} = 0.008$$

ความผิดพลาดคิดเป็น 0.8%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 ผลการทดลองที่โหลดสม่ำเสมอโดยใช้โทรศัพท์ 50 วัตต์

ชั่วโมง	Digital Energy Meter (kWh)	Energy Meter Digital (kWh)
2	0.10	0.10
4	0.20	0.20
6	0.30	0.30
8	0.40	0.40
10	0.50	0.50
12	0.60	0.60
14	0.70	0.70
16	0.80	0.80
18	0.90	0.90
20	1.10	1.00
22	1.21	1.10
24	1.31	1.20

ค่าไฟฟ้าของหลอดไฟขนาด 100 วัตต์

$$\text{จำนวนหน่วยไฟฟ้าที่ใช้} = \frac{(\text{จำนวนวัตต์ของเครื่องใช้ไฟฟ้า} \times \text{จำนวนชั่วโมงที่ใช้งาน})}{1,000 \text{ วัตต์}}$$

$$= \frac{50 \times 1}{1000} = 0.05$$

หลอดไฟ 100 วัตต์ 1 ชั่วโมงใช้ไฟไป 0.05 kW

ค่าความผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 | 1.2 - 1.31 | = 0.008
 ไม่ว่าจะกรณีใดก็ตาม อีกหนึ่งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2

ความผิดพลาดคิดเป็น 0.8%

4.1.3 ผลการทดลองที่วัดใน 1 วัน ที่โหลดไม่สม่ำเสมอ

ตารางที่ 4.3 ผลการทดลองที่โหลดไม่สม่ำเสมอ

ชั่วโมง	พลังงาน (kWh)	Energy Digital Meter (kWh)	Energy Analog Meter (kWh)
1	0.10	0.10	0.10
2	0.20	0.20	0.20
3	0.30	0.30	0.30
4	0.40	0.40	0.40
5	0.45	0.45	0.45
6	0.50	0.50	0.50
7	0.55	0.55	0.55
8	0.60	0.60	0.60
9	0.65	0.65	0.65
10	0.70	0.70	0.70
11	1.70	1.71	1.70
12	2.70	2.72	2.70
13	3.70	3.73	3.70
14	4.70	4.73	4.70
15	5.20	5.24	5.20
16	5.70	5.74	5.70
17	6.20	6.25	6.20
18	6.70	6.75	6.70
19	6.72	6.78	6.72
20	6.74	6.80	6.74
21	6.76	6.82	6.76
22	6.78	6.84	6.78
23	6.80	6.86	6.80
24	6.82	6.88	6.82

ค่าความผิดพลาด

$$\frac{|6.82 - 6.88|}{6.82} = 0.009$$

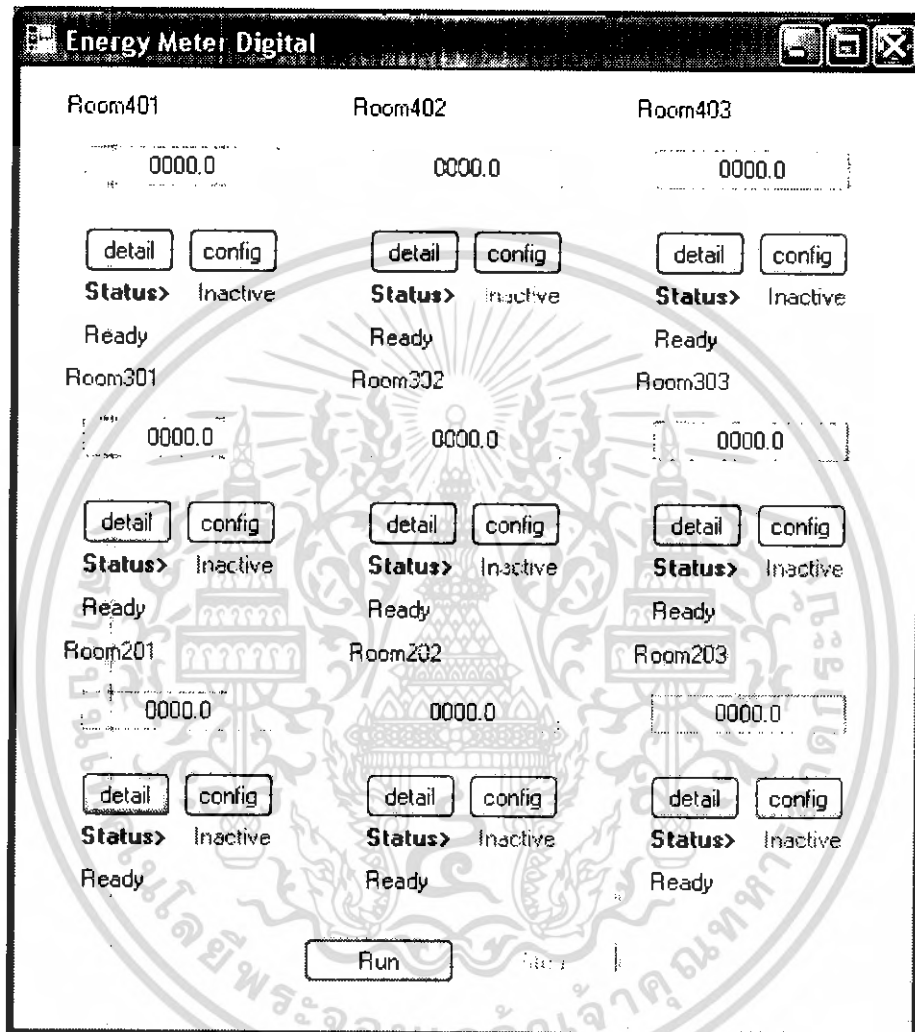
ความผิดพลาดคิดเป็น 0.9%

6.82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ส่วน Application

4.2.1 หน้าต่างหลักของตัว Application



รูปที่ 4.2 หน้าต่างหลักของตัว Application

จากรูปที่ 4.2 แสดงถึงหน้าต่างหลักที่ใช้ในการแสดงค่าพลังงานไฟฟ้าที่ใช้ไป โดยจะมีส่วนต่างๆ ดังนี้

- Detail ใช้แสดงข้อมูลของผู้อยู่อาศัย
- Config เป็นการตั้งค่า IP Address และ Port เพื่อรับข้อมูลจากตัว Energy Meter Digital ของแต่ละห้อง
- Run/Stop ทำการเริ่ม/หยุด โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Status มี 2 สถานะ
 - Active แสดงถึงว่าตัวโปรแกรมได้เปิดรอรับข้อมูลจากตัว Digital Energy Meter
 - Inactive แสดงถึงว่าตัวโปรแกรมได้ปิดการรอรับข้อมูลจากตัว Digital Energy Meter

4.2.2 หน้าต่างของส่วน Detail

ภายในหน้าต่าง Detail จะแสดงข้อมูลของผู้อยู่อาศัย โดยสามารถที่จะทำการแก้ไข เพิ่ม และ ลบข้อมูลต่างๆ ได้ หน้าต่างนี้จะแบ่งออกเป็น 4 ส่วน คือ

- ผู้เช่า
- ผู้เช่าร่วม
- รายละเอียดห้อง
- พิมพ์ใบเสร็จ

The screenshot shows a software window titled "Room" with a background watermark of a university seal. It is divided into four main sections:

- ผู้เช่า (Tenant):** Displays details for two tenants.

หมายเลขบัตรประชาชน	1-2584-00253-23-3	หมายเลขบัตรประชาชน	3-2105-00195-36-9
ชื่อ	เดชา	นามสกุล	สนเด็ก
เกิด	31/4/2530	อายุ	19 เพศ ชาย
สัญชาติ	ไทย	เชื้อชาติ	อังกฤม
หมู่เลือด	บี	ศาสนา	คริส
ที่อยู่	456 ม.5 ต.ระยะ อ.บ่อวิน จ.ระยอง 00000123		
- ผู้เช่าร่วม (Co-tenant):** Displays details for a co-tenant.

ชื่อ	ดวงใจ	นามสกุล	เสียมงาม
เกิด	14/2/2525	อายุ	24 เพศ หญิง
สัญชาติ	ไทย	เชื้อชาติ	ไทย
หมู่เลือด	เอ	ศาสนา	พุทธ
ที่อยู่	45/10 หมู่.6 ต.หนองบัว อ.บ้านค่าย จ.ระยอง 21120		
- รายละเอียดห้อง (Room Details):** Displays room information.

ห้องเลขที่	401	ค่าไฟฟ้าล่าสุด	5625.5
วันที่แรกที่เช่าอยู่	1/5/2006	วันที่	8 กุมภาพันธ์ 2550
วันที่ต่ออายุสัญญา	1/5/2007	เวลา	1:31:34
ค่าเช่าห้อง	2500 บาท	ค่าไฟฟ้าเดือนก่อน	5625.5
อัตราค่าไฟฟ้า	6 บาทต่อหน่วย	วันที่	23 มกราคม 2550
อัตราค่าน้ำประปา	17 บาทต่อหน่วย	เวลา	4:05:56
- Buttons:** At the bottom of each section are buttons for "แก้ไข" (Edit), "เพิ่ม" (Add), and "ลบ" (Delete).

รูปที่ 4.3 หน้าต่างของส่วน Detail

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือกพิมพ์ใบเสร็จจะแสดงหน้าต่างดังรูปที่ 4.4 โดยภายในหน้าต่างนี้จะแสดงรายละเอียดใบแจ้งหนี้ โดยจะมีส่วนของ

- พิมพ์ จะปรากฏ Print Dialog ขึ้นมา ดังรูปที่ 4.5
- ดูตัวอย่าง จะปรากฏ Print Preview Dialog ขึ้นมา ดังรูปที่ 4.6

The screenshot shows a 'Print' dialog box with a large watermark of a university seal in the background. The text inside the dialog is as follows:

ใบแจ้งหนี้ ห้อง 401

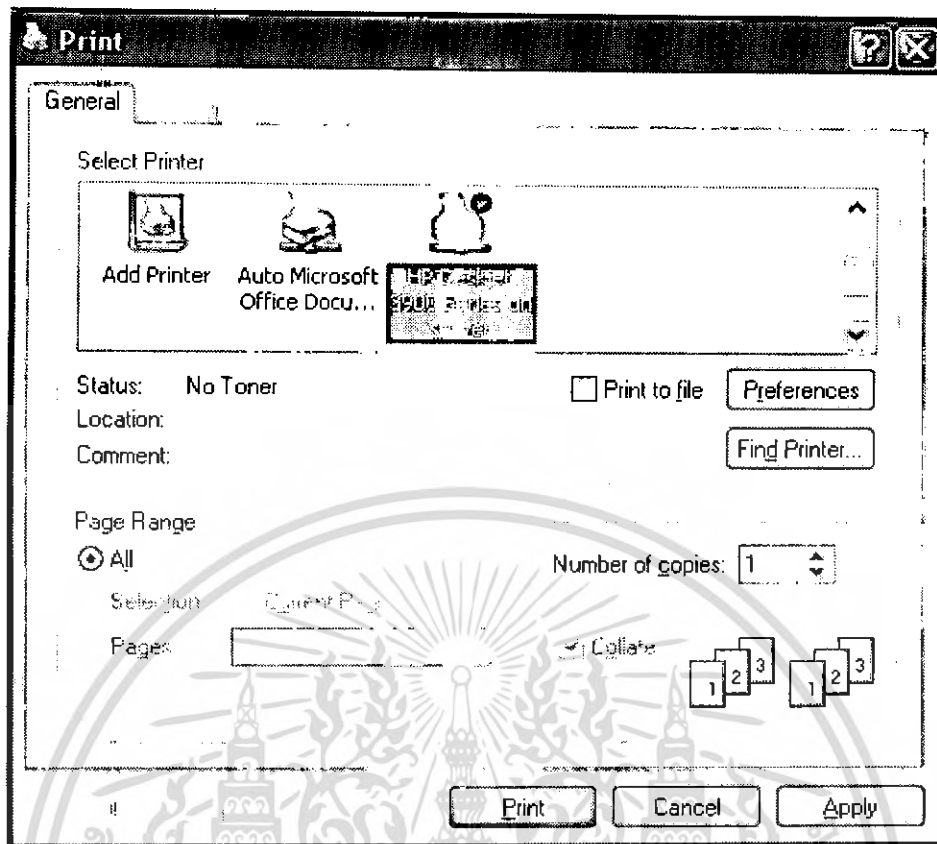
ประจำวันที่ 8 กุมภาพันธ์ 2550

หมายเลขมิเตอร์ไฟฟ้าวันที่ 23 มกราคม 2550	เท่ากับ	5625.5
หมายเลขมิเตอร์ไฟฟ้าวันที่ 31 ธันวาคม 2549	เท่ากับ	5255.4
ผู้เช่า เดต้า	ส่วนตัว	
ผู้เช่าร่วม ดวงใจ	เตรียมงาม	
ค่าเช่าห้อง	2500	บาท
ค่าไฟฟ้าหน่วยละ 6 บาท	ใช้ทั้งหมด	370 หน่วย
	เป็นเงินจำนวน	2220 บาท
	รวมเป็นเงินจำนวน	4720 บาท

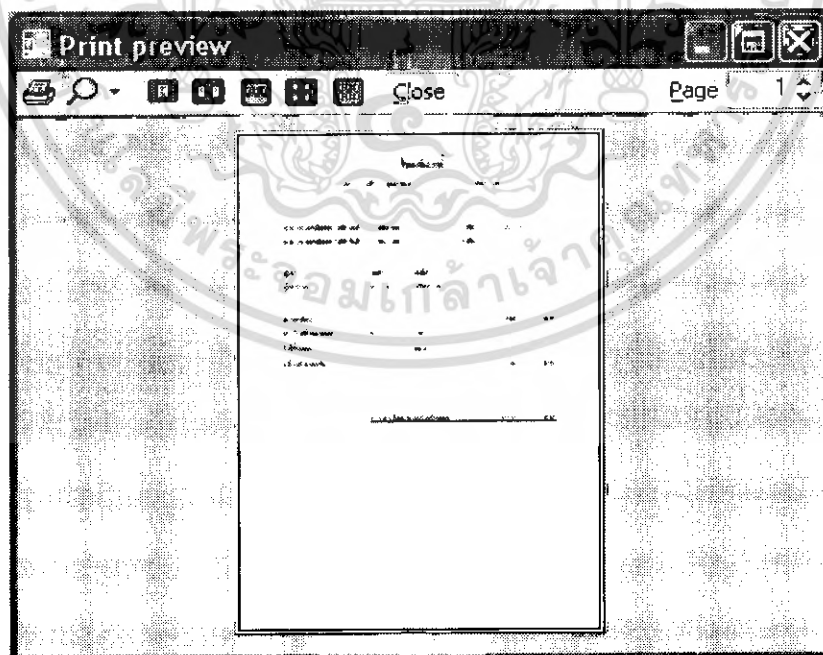
At the bottom of the dialog, there are two buttons: 'พิมพ์' (Print) and 'ดูตัวอย่าง' (Preview).

รูปที่ 4.4 หน้าต่างของส่วน Print

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



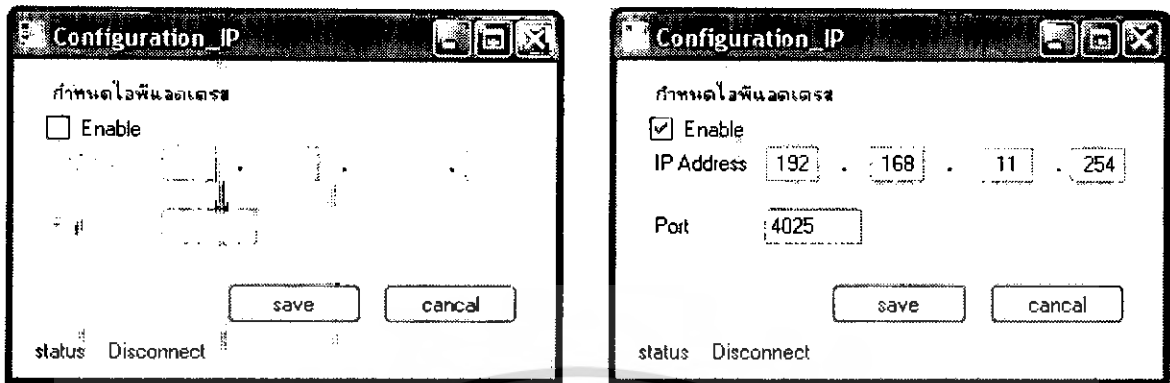
รูปที่ 4.5 หน้าต่างของส่วน Print Dialog



รูปที่ 4.6 หน้าต่างของส่วน Print Preview Dialog

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 หน้าต่างของ Config



ก. เมื่อไม่มีการ Enable

ข. เมื่อมีการ Enable

รูปที่ 4.7 หน้าต่างของส่วน Config

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และสรุป

5.1 บทสรุป

ในการทดลองโครงการนี้ ได้ทำการศึกษาถึงเนื้อหาและพื้นฐานโครงสร้างการทำงานโดยทั่วไปของตัว Digital Energy Meter ซึ่งสามารถแบ่งออกเป็น 3 ส่วน คือ ส่วนแรกเป็นส่วนของการวัดค่าพลังงานไฟฟ้า โดยจะใช้ ADE 7755 Chip ซึ่งในการใช้งานตัว ADE 7755 Chip นั้นจำเป็นต้องทำการลดค่าแรงดันลงด้วยวงจรแบ่งแรงดัน และแบ่งกระแสเพื่อให้ได้ค่าแรงดันที่ต้องการ ส่วนที่สองคือส่วนของ Controller (ในที่นี้ใช้ P89V51RD2) จะรับค่าพลังงานที่ใช้จากตัว ADE Chip ซึ่งตัว Controller จะทำหน้าที่ในการควบคุมการทำงานว่าตัว Energy Meter เช่น แสดงผล , เก็บข้อมูลต่างๆ ลง EEPROM และทำการส่งข้อมูลไปยัง Ethernet Module ส่วนสุดท้ายคือส่วนของ Ethernet Interface ซึ่งจะทำหน้าที่แปลงข้อมูลที่ได้รับจาก Controller ให้อยู่ในรูปของ TCP Datagram เพื่อส่งไปยังศูนย์กลาง

5.2 วิจัยถึงสิ่งที่ได้จากโครงการ

- 5.2.1 ทราบหลักการและโครงสร้างทั่วไปของตัว Energy Meter ทั้งแบบ Digital และ Analog
- 5.2.2 สร้าง Digital Energy Meter ขึ้นมาใช้งานได้ โดยมีคุณสมบัติ คือ แสดงผลผ่านหน้าจอ LCD และสามารถส่งผ่านข้อมูลไปยังเครื่องคอมพิวเตอร์ได้โดยผ่าน Ethernet Module
- 5.2.3 การทำงานสามารถต่อเข้ากับคอมพิวเตอร์ได้โดยจะมีตัว Application รองรับการทำงานของตัว Digital Energy Meter ทำให้มีการทำงานเป็นแบบอัตโนมัติ

5.3 ปัญหาอุปสรรคและแนวทางแก้ไข

- 5.3.1 การใช้งานตัว Digital Energy Meter จริงยังไม่เป็นที่รู้จักกันอย่างแพร่หลาย
- 5.3.2 ตัว IC ที่ใช้ในการวัดค่าพลังงานที่ใช้ในการทำโครงการนี้หาได้ยาก
- 5.3.3 ในการสร้างมาตรฐานให้กับตัว Digital Energy Meter อาจทำได้ลำบาก เนื่องจากต้องนำผลที่ได้มาเปรียบเทียบกับ Energy Meter แบบ Analog ที่ใช้กันทั่วไป ซึ่งตัวที่นำมาเปรียบเทียบกับนั้นจะมีความเที่ยงตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 แนวทางการพัฒนาต่อ

- 5.4.1 พัฒนา Digital Energy Meter ให้มีมาตรฐานที่น่าเชื่อถือมากขึ้น โดยอาจจะนำเครื่อง Digital Energy Meter ที่สร้างขึ้นไปเปรียบเทียบกับตัวเครื่องวัดมาตรฐานที่การไฟฟ้าใช้งานจริง เมื่อสร้างความมาตรฐานให้แก่ Energy Meter ที่สร้างขึ้น
- 5.4.2 พัฒนา Digital Energy Meter ให้สามารถแจ้งอัตราค่าไฟฟ้าให้เร็วขึ้น เช่น การส่งใบแจ้งหนี้ผ่าน ทาง sms เป็นต้น
- 5.4.3 พัฒนาให้สามารถแจ้งเตือนเมื่อมีการใช้ปริมาณไฟฟ้าเกินกำหนด จนถึงขั้นอันตราย อาจเกิดไฟฟ้าลัดวงจรได้ หรือตรวจเช็คผ่านระบบ Network
- 5.4.4 พัฒนาทางด้านความปลอดภัยต่าง ๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Energy Metering IC with Pulse Output

ADE7755*

FEATURES

- High Accuracy, Surpasses 50 Hz/60 Hz IEC 687/1036
- Less than 0.1% Error over a Dynamic Range of 500 to 1
- The ADE7755 Supplies Average Real Power on the Frequency Outputs F1 and F2
- The High-Frequency Output CF Is Intended for Calibration and Supplies Instantaneous Real Power
- Pin Compatible with AD7755 with Synchronous CF and F1/F2 Outputs
- The Logic Output REVP Can Be Used to Indicate a Potential Miswiring or Negative Power
- Direct Drive for Electromechanical Counters and Two Phase Stepper Motors (F1 and F2)
- A PGA in the Current Channel Allows the Use of Small Values of Shunt and Burden Resistance
- Proprietary ADCs and DSP Provide High Accuracy over Large Variations in Environmental Conditions and Time
- On-Chip Power Supply Monitoring
- On-Chip Creep Protection (No Load Threshold)
- On-Chip Reference 2.5 V \pm 8% (30 ppm/ $^{\circ}$ C Typical) with External Overdrive Capability
- Single 5 V Supply, Low Power (15 mW Typical)
- Low Cost CMOS Process

GENERAL DESCRIPTION

The ADE7755 is pin compatible with the AD7755. The only difference between the ADE7755 and the AD7755 is that the ADE7755 features a synchronous CF and F1/F2 outputs under all load conditions.

The ADE7755 is a high accuracy electrical energy measurement IC. The part specifications surpass the accuracy requirements as quoted in the IEC1036 standard. See Analog Devices' Application Note AN-559 for a description of an IEC1036 watt-hour meter reference design based on the AD7755.

The only analog circuitry used in the ADE7755 is in the ADCs and reference circuit. All other signal processing (e.g., multiplication and filtering) is carried out in the digital domain. This approach provides superior stability and accuracy over extremes in environmental conditions and over time.

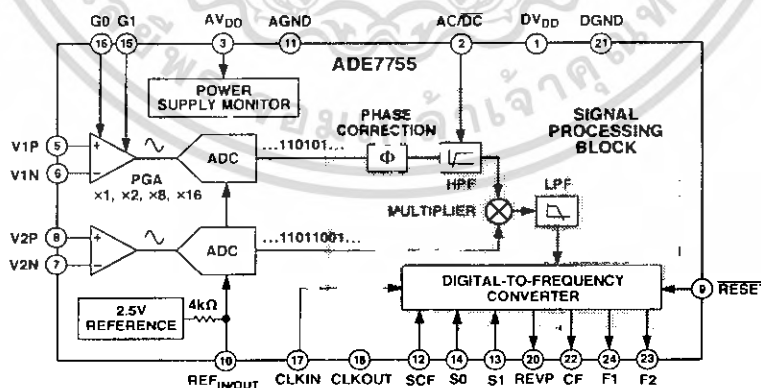
The ADE7755 supplies average real power information on the low-frequency outputs F1 and F2. These logic outputs may be used to directly drive an electromechanical counter or interface to an MCU. The CF logic output gives instantaneous real power information. This output is intended to be used for calibration purposes or for interfacing to an MCU.

The ADE7755 includes a power supply monitoring circuit on the AV_{DD} supply pin. The ADE7755 will remain in a reset condition until the supply voltage on AV_{DD} reaches 4 V. If the supply falls below 4 V, the ADE7755 will also be reset and no pulses will be issued on F1, F2, and CF.

Internal phase matching circuitry ensures that the voltage and current channels are phase matched whether the HPF in Channel 1 is on or off. An internal no-load threshold ensures that the ADE7755 does not exhibit any creep when there is no load.

The ADE7755 is available in a 24-lead SSOP package.

FUNCTIONAL BLOCK DIAGRAM



*U.S. Patents 5,745,323, 5,760,617, 5,862,069, and 5,872,469

REV. 0

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
 Tel: 781/329-4700 www.analog.com
 Fax: 781/326-8703 © Analog Devices, Inc., 2002

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADE7755—SPECIFICATIONS ($AV_{DD} = DV_{DD} = 5\text{ V} \pm 5\%$, $AGND = DGND = 0\text{ V}$, On-Chip Reference, $CLKIN = 3.58\text{ MHz}$, T_{MIN} to $T_{MAX} = -40^{\circ}\text{C}$ to $+85^{\circ}\text{C}$.)

Parameter	Specifications	Unit	Test Conditions/Comments
ACCURACY^{1, 2}			
Measurement Error ¹ on Channel 1			Channel 2 with Full-Scale Signal ($\pm 660\text{ mV}$), 25°C
Gain = 1	0.1	% Reading typ	Over a Dynamic Range 500 to 1
Gain = 2	0.1	% Reading typ	Over a Dynamic Range 500 to 1
Gain = 8	0.1	% Reading typ	Over a Dynamic Range 500 to 1
Gain = 16	0.1	% Reading typ	Over a Dynamic Range 500 to 1
Phase Error ¹ Between Channels			Line Frequency = 45 Hz to 65 Hz
V1 Phase Lead 37° (PF = 0.8 Capacitive)	± 0.1	Degrees($^{\circ}$) max	$AC/\overline{DC} = 0$ and $AC/\overline{DC} = 1$
V1 Phase Lag 60° (PF = 0.5 Inductive)	± 0.1	Degrees($^{\circ}$) max	$AC/\overline{DC} = 0$ and $AC/\overline{DC} = 1$
AC Power Supply Rejection ¹			$AC/\overline{DC} = 1$, $S0 = S1 = 1$, $G0 = G1 = 0$
Output Frequency Variation (CF)	0.2	% Reading typ	V1 = 100 mV rms, V2 = 100 mV rms, @ 50 Hz Ripple on AV_{DD} of 200 mV rms @ 100 Hz
DC Power Supply Rejection ¹			$AC/\overline{DC} = 1$, $S0 = S1 = 1$, $G0 = G1 = 0$
Output Frequency Variation (CF)	± 0.3	% Reading typ	V1 = 100 mV rms, V2 = 100 mV rms, $AV_{DD} = DV_{DD} = 5\text{ V} \pm 250\text{ mV}$
ANALOG INPUTS			See Analog Inputs section
Maximum Signal Levels	± 1	V max	V1P, V1N, V2N, and V2P to AGND
Input Impedance (DC)	390	k Ω min	CLKIN = 3.58 MHz
Bandwidth (-3 dB)	14	kHz typ	CLKIN/256, CLKIN = 3.58 MHz
ADC Offset Error ^{1, 2}	± 25	mV max	Gain = 1, See Terminology and Performance Graphs
Gain Error ¹	± 7	% Ideal typ	External 2.5 V Reference, Gain = 1
Gain Error Match ¹	± 0.2	% Ideal typ	V1 = 470 mV dc, V2 = 660 mV dc External 2.5 V Reference
REFERENCE INPUT			
REF _{IN,OUT} Input Voltage Range	2.7 2.3	V max V min	2.5 V + 8% 2.5 V - 8%
Input Impedance	3.2	k Ω min	
Input Capacitance	10	pF max	
ON-CHIP REFERENCE			Nominal 2.5 V
Reference Error	± 200	mV max	
Temperature Coefficient	± 30	ppm/ $^{\circ}\text{C}$ typ	
CLKIN			Note All Specifications for CLKIN of 3.58 MHz
Input Clock Frequency	4 1	MHz max MHz min	
LOGIC INPUTS³			
SCF, S0, S1, AC/ \overline{DC} , RESET, G0, and G1			
Input High Voltage, V_{INH}	2.4	V min	$DV_{DD} = 5\text{ V} \pm 5\%$
Input Low Voltage, V_{INL}	0.8	V max	$DV_{DD} = 5\text{ V} \pm 5\%$
Input Current, I_{IN}	± 3	μA max	Typically 10 nA, $V_{IN} = 0\text{ V}$ to DV_{DD}
Input Capacitance, C_{IN}	10	pF max	
LOGIC OUTPUTS³			
F1 and F2			
Output High Voltage, V_{OH}	4.5	V min	$I_{SOURCE} = 10\text{ mA}$ $DV_{DD} = 5\text{ V}$
Output Low Voltage, V_{OL}	0.5	V max	$I_{SINK} = 10\text{ mA}$ $DV_{DD} = 5\text{ V}$
CF and REVP			
Output High Voltage, V_{OH}	4	V min	$I_{SOURCE} = 5\text{ mA}$ $DV_{DD} = 5\text{ V}$
Output Low Voltage, V_{OL}	0.5	V max	$I_{SINK} = 5\text{ mA}$ $DV_{DD} = 5\text{ V}$

Parameter	Specifications	Unit	Test Conditions/Comments
POWER SUPPLY			
AV _{DD}	4.75	V min	5 V - 5%
	5.25	V max	5 V + 5%
DV _{DD}	4.75	V min	5 V - 5%
	5.25	V max	5 V + 5%
AI _{DD}	3	mA max	Typically 2 mA
DI _{DD}	2.5	mA max	Typically 1.5 mA

NOTES

- ¹See Terminology section for explanation of specifications.
 - ²See Plots in Typical Performance Graphs.
 - ³Sample tested during initial release and after any redesign or process change that may affect this parameter.
- Specifications subject to change without notice.

TIMING CHARACTERISTICS^{1, 2} (AV_{DD} = DV_{DD} = 5 V ± 5%, AGND = DGND = 0 V, On-Chip Reference, CLKIN = 3.58 MHz, T_{MIN} to T_{MAX} = -40°C to +85°C.)

Parameter	Specifications	Unit	Test Conditions/Comments
t ₁ ³	275	ms	F1 and F2 Pulsewidth (Logic Low)
t ₂	See Table III	sec	Output Pulse Period. See Transfer Function section.
t ₃	1/2 t ₂	sec	Time between F1 Falling Edge and F2 Falling Edge
t ₁ ^{3, 1}	90	ms	CF Pulsewidth (Logic High)
t ₃	See Table IV	sec	CF Pulse Period. See Transfer Function section.
t ₀	CLKIN ⁴	sec	Minimum Time between F1 and F2 Pulse

NOTES

- ¹Sample tested during initial release and after any redesign or process change that may affect this parameter.
 - ²See Figure 1.
 - ³The pulsewidths of F1, F2, and CF are not fixed for higher output frequencies. See Frequency Outputs section.
 - ⁴The CF pulse is always 18 μs in the high-frequency mode. See Frequency Outputs section and Table IV.
- Specifications subject to change without notice.

ABSOLUTE MAXIMUM RATINGS*

(T_A = 25°C unless otherwise noted.)

AV _{DD} to AGND	-0.3 V to +7 V
DV _{DD} to DGND	0.3 V to +7 V
DV _{DD} to AV _{DD}	-0.3 V to +0.3 V
Analog Input Voltage to AGND	
V1P, V1N, V2P, and V2N	-6 V to +6 V
Reference Input Voltage to AGND	-0.3 V to AV _{DD} + 0.3 V
Digital Input Voltage to DGND	-0.3 V to DV _{DD} + 0.3 V
Digital Output Voltage to DGND	-0.3 V to DV _{DD} + 0.3 V
Operating Temperature Range	
Industrial	-40°C to +85°C
Storage Temperature Range	-65°C to +150°C

Junction Temperature	150°C
24-Lead SSOP, Power Dissipation	450 mW
θ _{JA} Thermal Impedance	112°C/W
Lead Temperature, Soldering	
Vapor Phase (60 sec)	215°C
Infrared (15 sec)	220°C

*Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those listed in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ORDERING GUIDE

Model	Package Description	Package Options
ADE7755ARS	Shrink Small Outline Package	RS-24
ADE7755ARSRL	Shrink Small Outline Package in Reel	RSRL-24
ADE7755AN-REF	ADE7755 Reference Design PCB (See AN-559)	
EVAL-ADE7755EB	ADE7755 Evaluation Board	

CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although the ADE7755 features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high-energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.



ADE7755

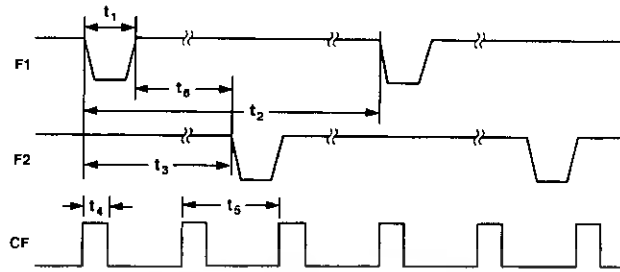


Figure 1. Timing Diagram for Frequency Outputs



PIN FUNCTION DESCRIPTIONS

Pin No.	Mnemonic	Description
1	DV _{DD}	Digital Power Supply. This pin provides the supply voltage for the digital circuitry in the ADE7755. The supply voltage should be maintained at $5\text{ V} \pm 5\%$ for specified operation. This pin should be decoupled with a 10 μF capacitor in parallel with a ceramic 100 nF capacitor.
2	AC/ $\overline{\text{DC}}$	High-Pass Filter Select. This logic input is used to enable the HPF in Channel 1 (Current Channel). A logic one on this pin enables the HPF. The associated phase response of this filter has been internally compensated over a frequency range of 45 Hz to 1 kHz. The HPF filter should be enabled in power metering applications.
3	AV _{DD}	Analog Power Supply. This pin provides the supply voltage for the analog circuitry in the ADE7755. The supply should be maintained at $5\text{ V} \pm 5\%$ for specified operation. Every effort should be made to minimize power supply ripple and noise at this pin by the use of proper decoupling. This pin should be decoupled to AGND with a 10 μF capacitor in parallel with a ceramic 100 nF capacitor.
4, 19	NC	No Connect
5, 6	V1P, V1N	Analog Inputs for Channel 1 (Current Channel). These inputs are fully differential voltage inputs with a maximum differential signal level of $\pm 470\text{ mV}$ for specified operation. Channel 1 also has a PGA, and the gain selections are outlined in Table I. The maximum signal level at these pins is $\pm 1\text{ V}$ with respect to AGND. Both inputs have internal ESD protection circuitry. An overvoltage of $\pm 6\text{ V}$ can be sustained on these inputs without risk of permanent damage.
7, 8	V2N, V2P	Negative and Positive Inputs for Channel 2 (Voltage Channel). These inputs provide a fully differential input pair. The maximum differential input voltage is $\pm 660\text{ mV}$ for specified operation. The maximum signal level at these pins is $\pm 1\text{ V}$ with respect to AGND. Both inputs have internal ESD protection circuitry, and an overvoltage of $\pm 6\text{ V}$ can also be sustained on these inputs without risk of permanent damage.
9	$\overline{\text{RESET}}$	Reset Pin for the ADE7755. A logic low on this pin will hold the ADCs and digital circuitry in a reset condition. Bringing this pin logic low will clear the ADE7755 internal registers.
10	REF _{IN OUT}	This pin provides access to the on-chip voltage reference. The on-chip reference has a nominal value of $2.5\text{ V} \pm 8\%$ and a typical temperature coefficient of $30\text{ ppm}^\circ\text{C}$. An external reference source may also be connected at this pin. In either case, this pin should be decoupled to AGND with a 1 μF ceramic capacitor and 100 nF ceramic capacitor.
11	AGND	This provides the ground reference for the analog circuitry in the ADE7755, i.e., ADCs and reference. This pin should be tied to the analog ground plane of the PCB. The analog ground plane is the ground reference for all analog circuitry, e.g., antialiasing filters and current and voltage transducers. For good noise suppression, the analog ground plane should only connect to the digital ground plane at one point. A star ground configuration will help to keep noisy digital currents away from the analog circuits.
12	SCF	Select Calibration Frequency. This logic input is used to select the frequency on the calibration output CF. Table IV shows how the calibration frequencies are selected.
13, 14	S1, S0	These logic inputs are used to select one of four possible frequencies for the digital-to-frequency conversion. This offers the designer greater flexibility when designing the energy meter. See Selecting a Frequency for an Energy Meter Application section.
15, 16	G1, G0	These logic inputs are used to select one of four possible gains for Channel 1, i.e., V1. The possible gains are 1, 2, 8, and 16. See Analog Input section.
17	CLKIN	An external clock can be provided at this logic input. Alternatively, a parallel resonant AT crystal can be connected across CLKIN and CLKOUT to provide a clock source for the ADE7755. The clock frequency for specified operation is 3.579545 MHz. Crystal load capacitance of between 22 pF and 33 pF (ceramic) should be used with the gate oscillator circuit.
18	CLKOUT	A crystal can be connected across this pin and CLKIN as described above to provide a clock source for the ADE7755. The CLKOUT Pin can drive one CMOS load when an external clock is supplied at CLKIN or by the gate oscillator circuit.
20	REVP	This logic output will go logic high when negative power is detected, i.e., when the phase angle between the voltage and current signals is greater than 90° . This output is not latched and will be reset when positive power is once again detected. The output will go high or low at the same time as a pulse is issued on CF.

ADE7755

Pin No.	Mnemonic	Description
21	DGND	This provides the ground reference for the digital circuitry in the ADE7755, i.e., multiplier, filters, and digital-to-frequency converter. This pin should be tied to the digital ground plane of the PCB. The digital ground plane is the ground reference for all digital circuitry, e.g., counters (mechanical and digital), MCUs, and indicator LEDs. For good noise suppression, the analog ground plane should only be connected to the digital ground plane at one point only, e.g., a star ground.
22	CF	Calibration Frequency Logic Output. The CF logic output gives instantaneous real power information. This output is intended to be used for calibration purposes. Also see SCF Pin description.
23, 24	F2, F1	Low Frequency Logic Outputs. F1 and F2 supply <i>average real power</i> information. The logic outputs can be used to directly drive electromechanical counters and two phase stepper motors. See Transfer Function section.

TERMINOLOGY

MEASUREMENT ERROR

The error associated with the energy measurement made by the ADE7755 is defined by the following formula:

$$\text{Percentage Error} = \frac{\text{Energy Registered by the ADE7755} - \text{True Energy}}{\text{True Energy}} \times 100\%$$

PHASE ERROR BETWEEN CHANNELS

The HPF (High-Pass Filter) in Channel 1 has a phase lead response. To offset this phase response and equalize the phase response between channels, a phase correction network is also placed in Channel 1. The phase correction network matches the phase to within $\pm 0.1^\circ$ over a range of 45 Hz to 65 Hz and $\pm 0.2^\circ$ over a range 40 Hz to 1 kHz. See Figures 4 and 5.

POWER SUPPLY REJECTION

This quantifies the ADE7755 measurement error as a percentage of the reading when the power supplies are varied.

For the ac PSR measurement, a reading at nominal supplies (5 V) is taken. A 200 mV rms/100 Hz signal is then introduced onto the supplies and a second reading obtained under the same input signal levels. Any error introduced is expressed as a percentage of the reading (see Measurement Error definition).

For the dc PSR measurement, a reading at nominal supplies (5 V) is taken. The supplies are then varied $\pm 5\%$ and a second reading is obtained with the same input signal levels. Any error introduced is again expressed as a percentage of the reading.

ADC OFFSET ERROR

This refers to the dc offset associated with the analog inputs to the ADCs. It means that with the analog inputs connected to AGND, the ADCs still see a small dc signal (offset). The offset decreases with increasing gain in Channel V1. This specification is measured at a gain of 1. At a gain of 16, the dc offset is typically less than 1 mV. However, when the HPF is switched on, the offset is removed from the current channel and the power calculation is not affected by this offset.

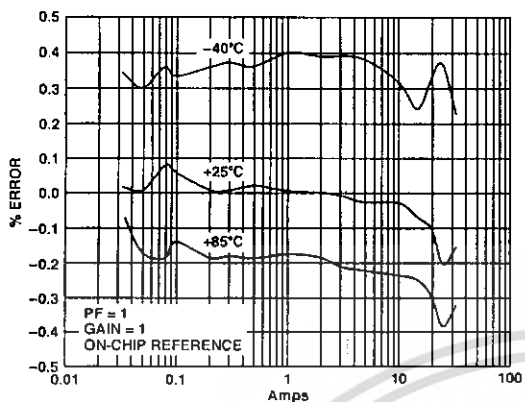
GAIN ERROR

The gain error of the ADE7755 is defined as the difference between the measured output frequency (minus the offset) and the ideal output frequency. It is measured with a gain of 1 in Channel V1. The difference is expressed as a percentage of the ideal frequency. The ideal frequency is obtained from the ADE7755 transfer function (see Transfer Function section).

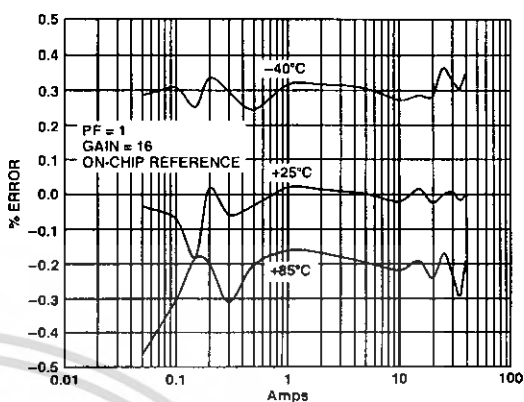
GAIN ERROR MATCH

The gain error match is defined as the gain error (minus the offset) obtained when switching between a gain of 1 and a gain of 2, 8, or 16. It is expressed as a percentage of the output frequency obtained under a gain of 1. This gives the gain error observed when the gain selection is changed from 1 to 2, 8, or 16.

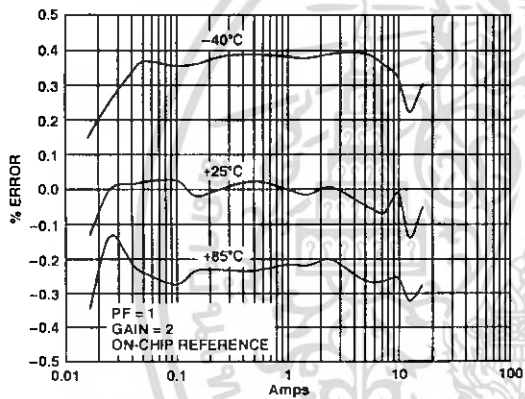
Typical Performance Characteristics—ADE7755



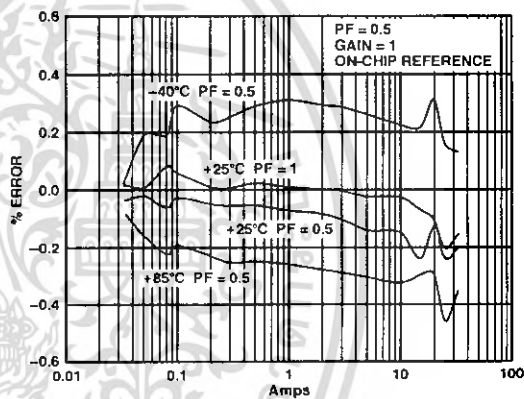
TPC 1. Error as a % of Reading (Gain = 1)



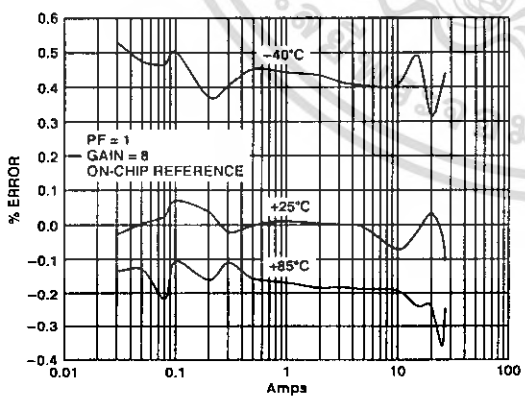
TPC 4. Error as a % of Reading (Gain = 16)



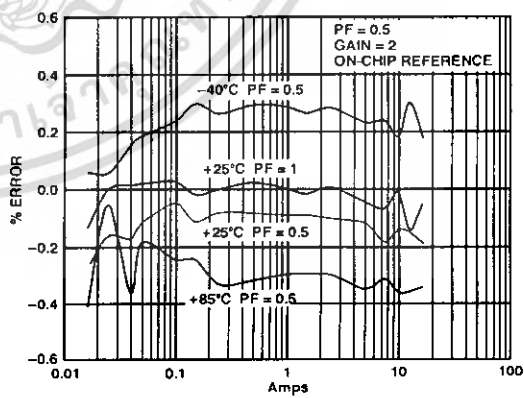
TPC 2. Error as a % of Reading (Gain = 2)



TPC 5. Error as a % of Reading (Gain = 1)

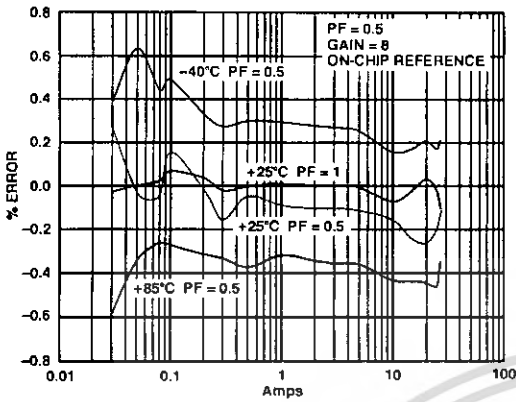


TPC 3. Error as a % of Reading (Gain = 8)

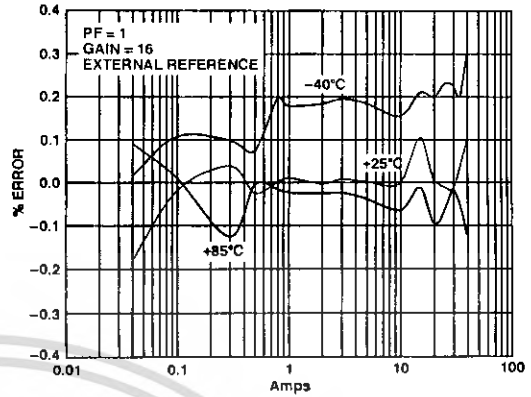


TPC 6. Error as a % of Reading (Gain = 2)

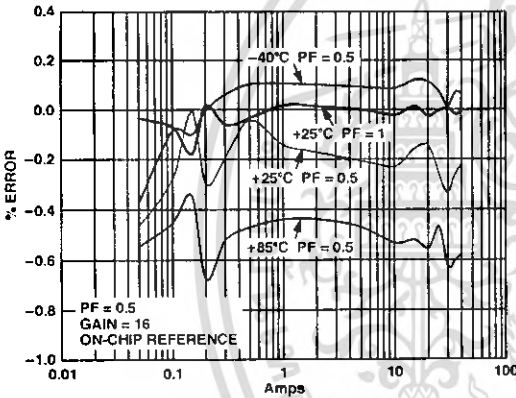
ADE7755



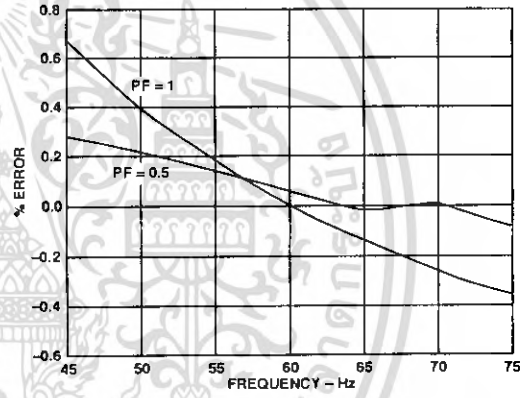
TPC 7. Error as a % of Reading (Gain = 8)



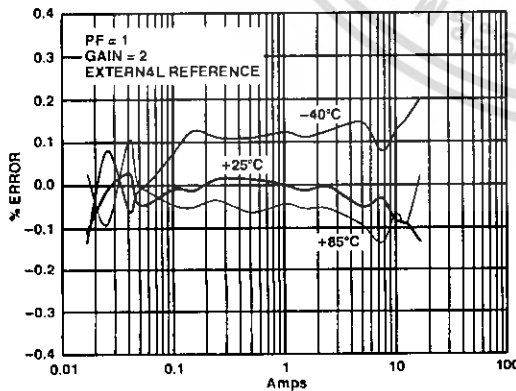
TPC 10. Error as a % of Reading over Temperature with an External Reference (Gain = 16)



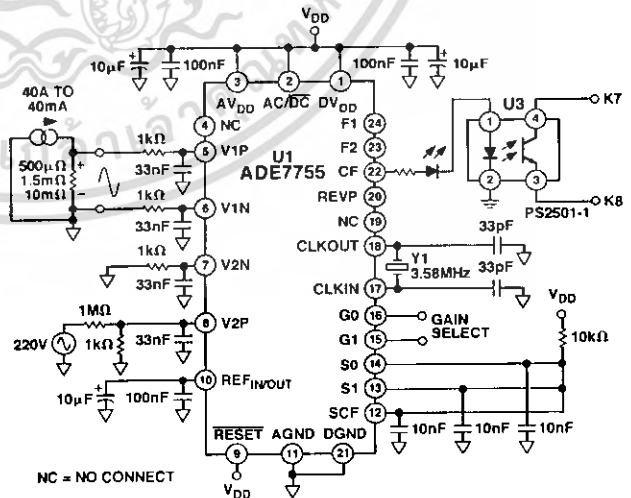
TPC 8. Error as a % of Reading (Gain = 16)



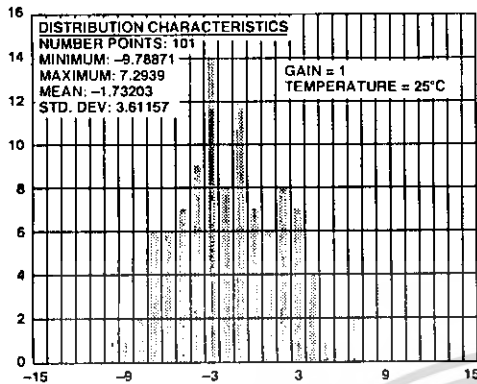
TPC 11. Error as a % of Reading over Frequency



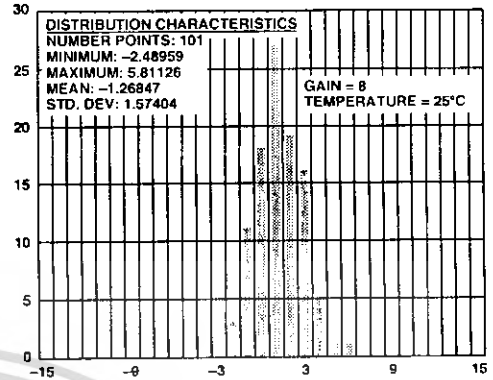
TPC 9. Error as a % of Reading over Temperature with an External Reference (Gain = 2)



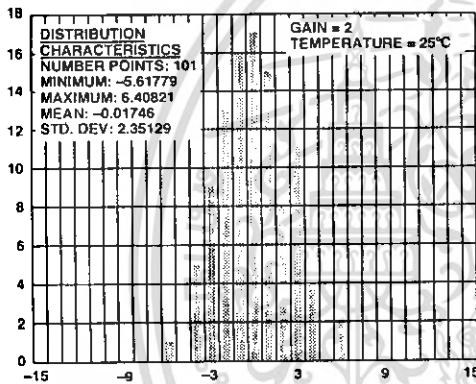
TPC 12. Test Circuit for Performance Curves



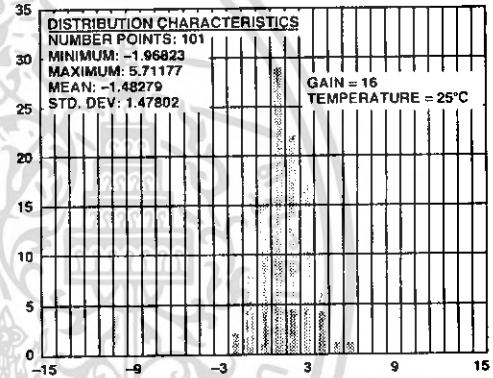
TPC 13. Channel 1 Offset Distribution (Gain = 1)



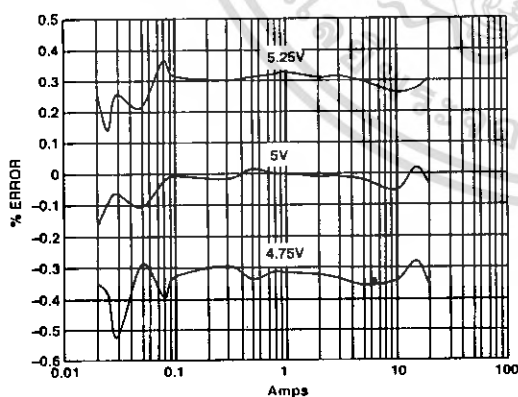
TPC 16. Channel 1 Offset Distribution (Gain = 8)



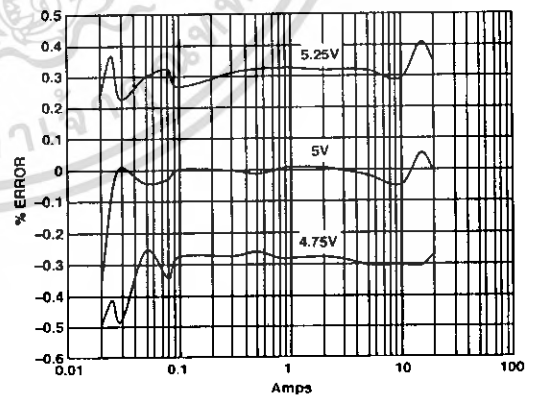
TPC 14. Channel 1 Offset Distribution (Gain = 2)



TPC 17. Channel 1 Offset Distribution (Gain = 16)



TPC 15. PSR with Internal Reference (Gain = 16)



TPC 18. PSR with External Reference (Gain = 16)

ADE7755

THEORY OF OPERATION

The two ADCs digitize the voltage signals from the current and voltage transducers. These ADCs are 16-bit second order sigma-delta with an oversampling rate of 900 kHz. This analog input structure greatly simplifies transducer interfacing by providing a wide dynamic range for direct connection to the transducer and also by simplifying the antialiasing filter design. A programmable gain stage in the current channel further facilitates easy transducer interfacing. A high-pass filter in the current channel removes any dc component from the current signal. This eliminates any inaccuracies in the real power calculation due to offsets in the voltage or current signals (see HPF and Offset Effects section).

The real power calculation is derived from the instantaneous power signal. The instantaneous power signal is generated by a direct multiplication of the current and voltage signals. In order to extract the real power component (i.e., the dc component), the instantaneous power signal is low-pass filtered. Figure 2 illustrates the instantaneous real power signal and shows how the real power information can be extracted by low-pass filtering the instantaneous power signal. This scheme correctly calculates real power for nonsinusoidal current and voltage waveforms at all power factors. All signal processing is carried out in the digital domain for superior stability over temperature and time.

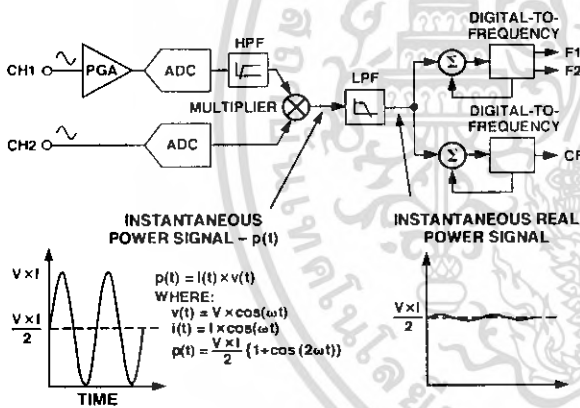


Figure 2. Signal Processing Block Diagram

The low-frequency output of the ADE7755 is generated by accumulating this real power information. This low frequency inherently means a long accumulation time between output pulses. The output frequency is therefore proportional to the average real power. This average real power information can, in turn, be accumulated (e.g., by a counter) to generate real energy information. Because of its high output frequency and shorter integration time, the CF output is proportional to the instantaneous real power. This is useful for system calibration purposes that would take place under steady load conditions.

Power Factor Considerations

The method used to extract the real power information from the instantaneous power signal (i.e., by low-pass filtering) is still valid even when the voltage and current signals are not in phase. Figure 3 displays the unity power factor condition and a DPF (Displacement Power Factor) = 0.5, i.e., current signal lagging

the voltage by 60°. If we assume the voltage and current waveforms are sinusoidal, the real power component of the instantaneous power signal (i.e., the dc term) is given by:

$$\left(\frac{V \times I}{2}\right) \times \cos(60^\circ)$$

This is the correct real power calculation.

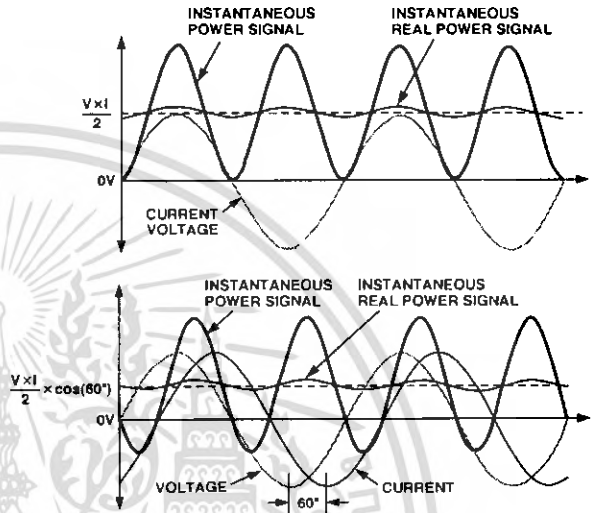


Figure 3. DC Component of Instantaneous Power Signal Conveys Real Power Information PF < 1

Nonsinusoidal Voltage and Current

The real power calculation method also holds true for nonsinusoidal current and voltage waveforms. All voltage and current waveforms in practical applications will have some harmonic content. Using the Fourier Transform, instantaneous voltage and current waveforms can be expressed in terms of their harmonic content.

$$v(t) = V_0 + \sqrt{2} \times \sum_{h=0}^{\infty} V_h \times \sin(h\omega t + \alpha h) \quad (1)$$

where:

- $v(t)$ is the instantaneous voltage
- V_0 is the average value
- V_h is the rms value of voltage harmonic h
- and
- αh is the phase angle of the voltage harmonic

$$i(t) = I_0 + \sqrt{2} \times \sum_{h=0}^{\infty} I_h \times \sin(h\omega t + \beta h) \quad (2)$$

where:

- $i(t)$ is the instantaneous current
- I_0 is the dc component
- I_h is the rms value of current harmonic h
- and
- βh is the phase angle of the current harmonic

Using Equations 1 and 2, the real power P can be expressed in terms of its fundamental real power (P_1) and harmonic real power (P_H).

$$P = P_1 + P_H$$

where:

$$P_1 = I_1^2 \times I_1 \cos \phi_1$$

$$\phi_1 = \alpha_1 - \beta_1$$

and:

$$P_H = \sum_{h=2}^{\infty} V_h \times I_h \cos \phi_h$$

$$\phi_h = \alpha_h - \beta_h$$

As can be seen from Equation 4 above, a harmonic real power component is generated for every harmonic, provided that harmonic is present in both the voltage and current waveforms. The power factor calculation has previously been shown to be accurate in the case of a pure sinusoid; therefore the harmonic real power must also correctly account for the power factor since it is made up of a series of pure sinusoids.

Note that the input bandwidth of the analog inputs is 14 kHz with a master clock frequency of 3.5795 MHz.

ANALOG INPUTS

Channel V1 (Current Channel)

The voltage output from the current transducer is connected to the ADE7755 here. Channel V1 is a fully differential voltage input. V1P is the positive input with respect to V1N.

The maximum peak differential signal on Channel 1 should be less than ± 470 mV (330 mV rms for a pure sinusoidal signal) for specified operation. Note that Channel 1 has a programmable gain amplifier (PGA) with user selectable gain of 1, 2, 8, or 16 (see Table I). These gains facilitate easy transducer interfacing.

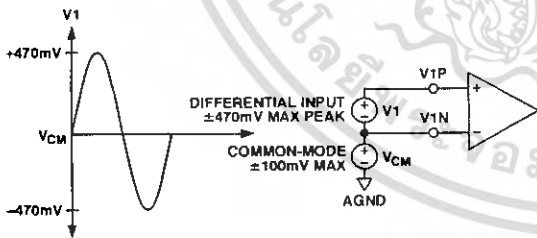


Figure 4. Maximum Signal Levels, Channel 1, Gain = 1

The diagram in Figure 4 illustrates the maximum signal levels on V1P and V1N. The maximum differential voltage is ± 470 mV divided by the gain selection. The differential voltage signal on the inputs must be referenced to a common mode, e.g., AGND. The maximum common-mode signal is ± 100 mV as shown in Figure 4.

Table I. Gain Selection for Channel 1

G1	G0	Gain	Maximum Differential Signal
0	0	1	± 470 mV
0	1	2	± 235 mV
1	0	8	± 60 mV
1	1	16	± 30 mV

Channel V2 (Voltage Channel)

The output of the line voltage transducer is connected to the ADE7755 at this analog input. Channel V2 is a fully differential voltage input. The maximum peak differential signal on Channel 2 is ± 660 mV. Figure 5 illustrates the maximum signal levels that can be connected to the ADE7755 Channel 2.

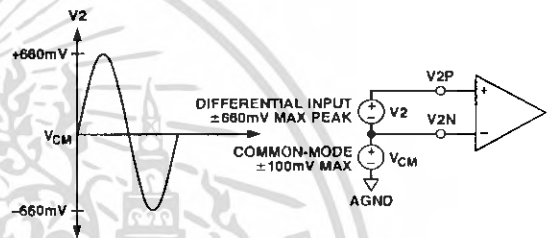


Figure 5. Maximum Signal Levels, Channel 2

Channel 2 must be driven from a common-mode voltage, i.e., the differential voltage signal on the input must be referenced to a common mode (usually AGND). The analog inputs of the ADE7755 can be driven with common-mode voltages of up to 100 mV with respect to AGND. However, best results are achieved using a common mode equal to AGND.

Typical Connection Diagrams

Figure 6 shows a typical connection diagram for Channel V1. A CT (current transformer) is the current transducer selected for this example. Notice the common-mode voltage for Channel 1 is AGND and is derived by center tapping the burden resistor to AGND. This provides the complementary analog input signals for V1P and V1N. The CT turns ratio and burden resistor Rb are selected to give a peak differential voltage of ± 470 mV/Gain at maximum load.

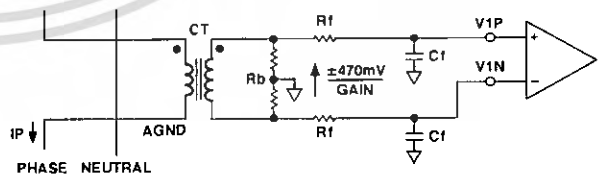


Figure 6. Typical Connection for Channel 1

ADE7755

Figure 7 shows two typical connections for Channel V2. The first option uses a PT (potential transformer) to provide complete isolation from the power line. In the second option, the ADE7755 is biased around the neutral wire, and a resistor divider provides a voltage signal that is proportional to the line voltage. Adjusting the ratio of Ra, Rb, and VR is also a convenient way of carrying out a gain calibration on the meter.

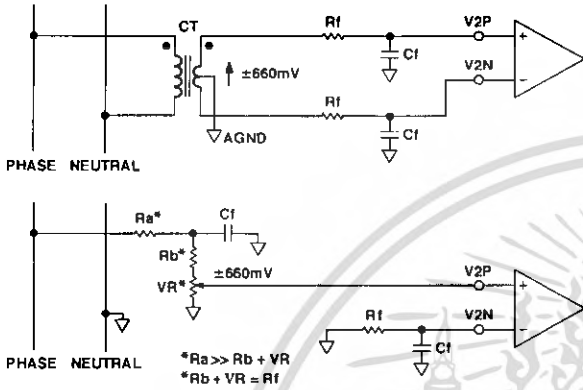


Figure 7. Typical Connections for Channel 2

POWER SUPPLY MONITOR

The ADE7755 contains an on-chip power supply monitor. The Analog Supply (AVDD) is continuously monitored by the ADE7755. If the supply is less than $4\text{ V} \pm 5\%$, the ADE7755 will be reset. This is useful to ensure correct device startup at power-up and power-down. The power supply monitor has built in hysteresis and filtering. This gives a high degree of immunity to false triggering due to noisy supplies.

In Figure 8, the trigger level is nominally set at 4 V. The tolerance on this trigger level is about $\pm 5\%$. The power supply and decoupling for the part should be such that the ripple at AVDD does not exceed $5\text{ V} \pm 5\%$ as specified for normal operation.

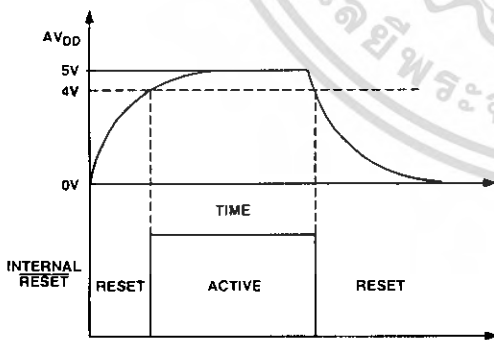


Figure 8. On-Chip Power Supply Monitor

HPF and Offset Effects

Figure 9 shows the effect of offsets on the real power calculation. An offset on Channel 1 and Channel 2 will contribute a dc component after multiplication. Since the dc component is extracted by the LPF, it will accumulate as real power. If not properly filtered, dc offsets will introduce error to the energy accumulation. This problem is easily avoided by enabling the HPF (i.e., Pin AC/DC is set logic high) in Channel 1. By removing the offset from at least one channel, no error component can be generated at dc by the multiplication. Error terms at $\cos(\omega t)$ are removed by the LPF and the digital-to-frequency conversion (see Digital-to-Frequency Conversion section).

$$\{V \cos(\omega t) + V_{OS}\} \times \{I \cos(\omega t) + I_{OS}\} = \frac{V \times I}{2} + V_{OS} \times I_{OS} + V_{OS} \times I \cos(\omega t) + I_{OS} \times V \cos(\omega t) + \frac{V \times I}{2} \times \cos(2\omega t)$$

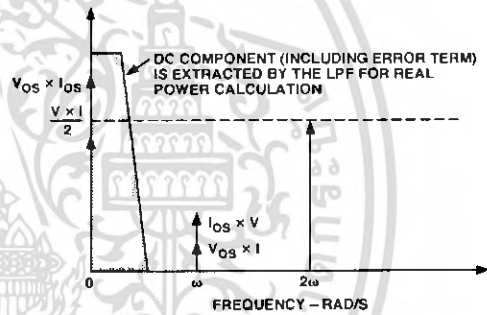


Figure 9. Effect of Channel Offset on the Real Power Calculation

The HPF in Channel 1 has an associated phase response that is compensated for on-chip. The phase compensation is activated when the HPF is enabled and is disabled when the HPF is not activated. Figures 10 and 11 show the phase error between channels with the compensation network activated. The ADE7755 is phase compensated up to 1 kHz as shown. This will ensure correct active harmonic power calculation even at low power factors.

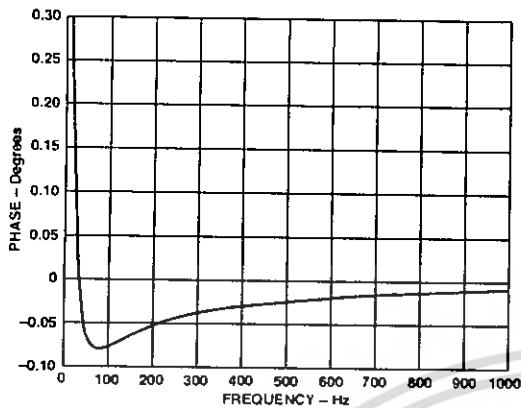


Figure 10. Phase Error between Channels (0 Hz to 1 kHz)

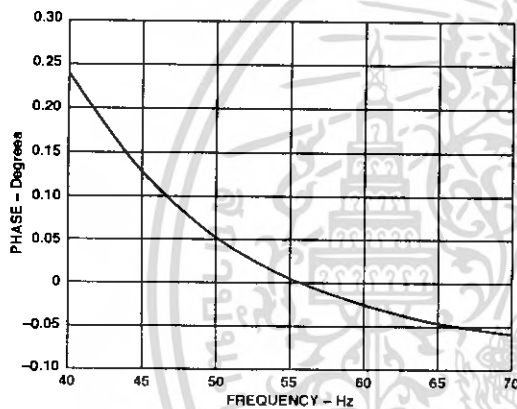


Figure 11. Phase Error between Channels (40 Hz to 70 Hz)

DIGITAL-TO-FREQUENCY CONVERSION

As previously described, the digital output of the low-pass filter after multiplication contains the real power information. However, since this LPF is not an ideal “brick wall” filter implementation, the output signal also contains attenuated components at the line frequency and its harmonics, i.e., $\cos(h\omega t)$ where $h = 1, 2, 3,$ and so on.

The magnitude response of the filter is given by:

$$|H(f)| = \frac{1}{1 + (f/8.9 \text{ Hz})^2} \tag{5}$$

For a line frequency of 50 Hz this would give an attenuation of the 2ω (100 Hz) component of approximately -22 dBs. The dominating harmonic will be at twice the line frequency, i.e., $\cos(2\omega t)$, and this is due to the instantaneous power signal.

Figure 12 shows the instantaneous real power signal at the output of the CPF, which still contains a significant amount of instantaneous power information, i.e., $\cos(2\omega t)$. This signal is then passed to the digital-to-frequency converter where it is integrated (accumulated) over time to produce an output frequency. This accumulation of the signal will suppress or average out any non-dc components in the instantaneous real power signal. The average value of a sinusoidal signal is zero. Hence, the frequency generated by the ADE7755 is proportional to the average real power. Figure 12 shows the digital-to-frequency conversion for steady load conditions, i.e., constant voltage and current.

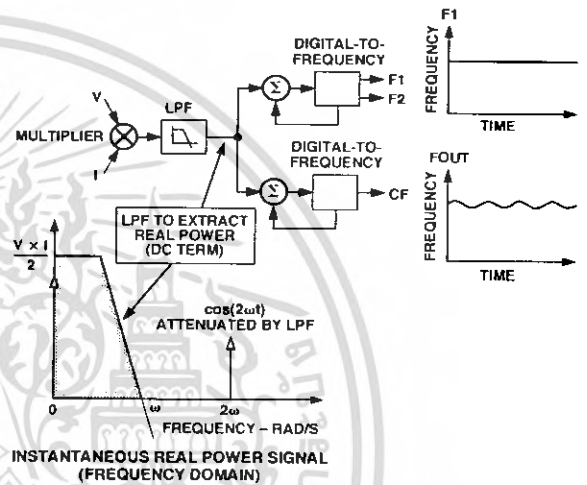


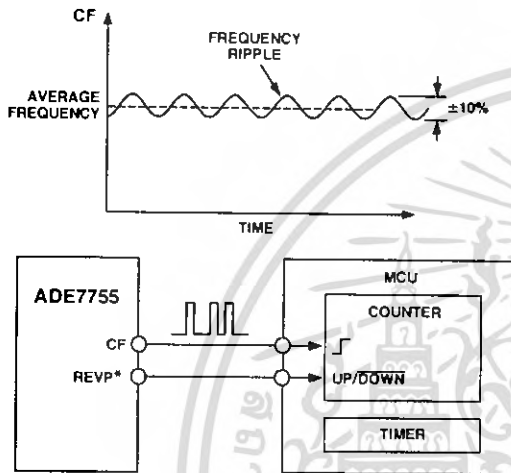
Figure 12. Real Power-to-Frequency Conversion

As can be seen in the diagram, the frequency output CF is seen to vary over time, even under steady load conditions. This frequency variation is primarily due to the $\cos(2\omega t)$ component in the instantaneous real power signal. The output frequency on CF can be up to 2048 times higher than the frequency on F1 and F2. This higher output frequency is generated by accumulating the instantaneous real power signal over a much shorter time while converting it to a frequency. This shorter accumulation period means less averaging of the $\cos(2\omega t)$ component. As a consequence, some of this instantaneous power signal passes through the digital-to-frequency conversion. This will not be a problem in the application. When CF is used for calibration purposes, the frequency should be averaged by the frequency counter. This will remove any ripple. If CF is measuring energy, e.g., in a microprocessor-based application, the CF output should also be averaged to calculate power. Because the outputs F1 and F2 operate at a much lower frequency, more averaging of the instantaneous real power signal is carried out. The result is a greatly attenuated sinusoidal content and a virtually ripple-free frequency output.

ADE7755

Interfacing the ADE7755 to a Microcontroller for Energy Measurement

The easiest way to interface the ADE7755 to a microcontroller is to use the CF high-frequency output with the output frequency scaling set to $2048 \times F1, F2$. This is done by setting $SCF = 0$ and $S0 = S1 = 1$ (see Table IV). With full-scale ac signals on the analog inputs, the output frequency on CF will be approximately 5.5 kHz. Figure 13 illustrates one scheme that could be used to digitize the output frequency and carry out the necessary averaging mentioned in the previous section.



*REVP MUST BE USED IF THE METER IS BIDIRECTIONAL OR DIRECTION OF ENERGY FLOW IS NEEDED

Figure 13. Interfacing the ADE7755 to an MCU

As shown, the frequency output CF is connected to an MCU counter or port. This will count the number of pulses in a given integration time that is determined by an MCU internal timer. The average power proportional to the average frequency is given by:

$$\text{Average Frequency} = \text{Average Real Power} = \frac{\text{Counter}}{\text{Timer}}$$

The energy consumed during an integration period is given by:

$$\text{Energy} = \text{Average Power} \times \text{Time} = \frac{\text{Counter}}{\text{Time}} \times \text{Time} = \text{Counter}$$

For the purpose of calibration, this integration time can be 10 to 20 seconds to accumulate enough pulses to ensure correct averaging of the frequency. In normal operation, the integration time can be reduced to one or two seconds depending, for example, on the required update rate of a display. With shorter integration times on the MCU, the amount of energy in each update may still have some small amount of ripple, even under steady load conditions. However, over a minute or more, the measured energy will have no ripple.

Power Measurement Considerations

Calculating and displaying power information will always have some associated ripple that will depend on the integration period used in the MCU to determine average power and also the load. For example, at light loads, the output frequency may be 10 Hz. With an integration period of two seconds, only about 20 pulses will be counted. The possibility of missing one pulse always exists, since the ADE7755 output frequency is running asynchronously to the MCU timer. This would result in a one-in-twenty (or 5%) error in the power measurement.

TRANSFER FUNCTION

Frequency Outputs F1 and F2

The ADE7755 calculates the product of two voltage signals (on Channel 1 and Channel 2) and then low-pass filters this product to extract real power information. This real power information is then converted to a frequency. The frequency information is output on F1 and F2 in the form of active low pulses. The pulse rate at these outputs is relatively low, e.g., 0.34 Hz maximum for ac signals with $S0 = S1 = 0$ (see Table III). This means that the frequency at these outputs is generated from real power information accumulated over a relatively long period of time. The result is an output frequency that is proportional to the average real power. The averaging of the real power signal is implicit to the digital-to-frequency conversion. The output frequency or pulse rate is related to the input voltage signals by the following equation.

$$\text{Freq} = \frac{8.06 \times V1 \times V2 \times \text{Gain} \times F_{1..4}}{V_{REF}^2}$$

where:

Freq = Output frequency on F1 and F2 (Hz)

V1 = Differential rms voltage signal on Channel 1 (Volts)

V2 = Differential rms voltage signal on Channel 2 (Volts)

Gain = 1, 2, 8, or 16, depending on the PGA gain selection made using logic inputs G0 and G1

V_{REF} = The reference voltage ($2.5 \text{ V} \pm 8\%$) (Volts)

$F_{1..4}$ = One of four possible frequencies selected by using the logic inputs S0 and S1—see Table II

Table II. $F_{1..4}$ Frequency Selection

S1	S0	$F_{1..4}$ (Hz)	XTAL/CLKIN*
0	0	1.7	$3.579 \text{ MHz}/2^{21}$
0	1	3.4	$3.579 \text{ MHz}/2^{20}$
1	0	6.8	$3.579 \text{ MHz}/2^{19}$
1	1	13.6	$3.579 \text{ MHz}/2^{18}$

NOTE

* $F_{1..4}$ is a binary fraction of the master clock and therefore will vary if the specified CLKIN frequency is altered

Example 1

Thus if full-scale differential dc voltages of +470 mV and -660 mV are applied to V1 and V2 respectively (470 mV is the maximum differential voltage that can be connected to Channel 1, and 660 mV is the maximum differential voltage that can be connected to Channel 2), the expected output frequency is calculated as follows:

Gain = 1, G0 = G1 = 0

F₁₋₄ = 1.7 Hz, S0 = S1 = 0

V1 = +470 mV dc = 0.47 V (rms of dc = dc)

V2 = -660 mV dc = 0.66 V (rms of dc = |dc)

V_{REF} = 2.5 V (nominal reference value)

NOTE: If the on-chip reference is used, actual output frequencies may vary from device to device due to reference tolerance of ±8%.

$$Freq = \frac{8.06 \times 0.47 \times 0.66 \times 1 \times 1.7}{2.5^2} = 0.68$$

Example 2

In this example, with ac voltages of ±470 mV peak applied to V1 and ±660 mV peak applied to V2, the expected output frequency is calculated as follows:

Gain = 1, G0 = G1 = 0

F₁₋₄ = 1.7 Hz, S0 = S1 = 0

V1 = rms of 470 mV peak ac = 0.47√2 volts

V2 = rms of 660 mV peak ac = 0.66√2 volts

V_{REF} = 2.5 V (nominal reference value)

NOTE: If the on-chip reference is used, actual output frequencies may vary from device to device due to reference tolerance of ±8%.

$$Freq = \frac{8.06 \times 0.47 \times 0.66 \times 1 \times 1.7}{\sqrt{2} \times \sqrt{2} \times 2.5^2} = 0.34$$

As can be seen from these two example calculations, the maximum output frequency for ac inputs is always half of that for dc input signals. Table III shows a complete listing of all maximum output frequencies.

Table III. Maximum Output Frequency on F1 and F2

S1	S0	Max Frequency for DC Inputs (Hz)	Max Frequency for AC Inputs (Hz)
0	0	0.68	0.34
0	1	1.36	0.68
1	0	2.72	1.36
1	1	5.44	2.72

Frequency Output CF

The pulse output CF (Calibration Frequency) is intended for use during calibration. The output pulse rate on CF can be up to 2048 times the pulse rate on F1 and F2. The lower the F₁₋₄ frequency selected, the higher the CF scaling (except for the high-frequency mode SCF = 0, S1 = S0 = 1). Table IV shows how the two frequencies are related, depending on the states of the logic inputs S0, S1, and SCF. Because of its relatively high

pulse rate, the frequency at this logic output is proportional to the instantaneous real power. As is the case with F1 and F2, the frequency is derived from the output of the low-pass filter after multiplication. However, because the output frequency is high, this real power information is accumulated over a much shorter time. Hence, less averaging is carried out in the digital-to-frequency conversion. With much less averaging of the real power signal, the CF output is much more responsive to power fluctuations (see Figure 2, signal processing block diagram).

Table IV. Maximum Output Frequency on CF

SCF	S1	S0	F ₁₋₄ (Hz)	CF Max for AC Signals (Hz)
1	0	0	1.7	128 × F1, F2 = 43.52
0	0	0	1.7	64 × F1, F2 = 21.76
1	0	1	3.4	64 × F1, F2 = 43.52
0	0	1	3.4	32 × F1, F2 = 21.76
1	1	0	6.8	32 × F1, F2 = 43.52
0	1	0	6.8	16 × F1, F2 = 21.76
1	1	1	13.6	16 × F1, F2 = 43.52
0	1	1	13.6	2048 × F1, F2 = 5.57 kHz

SELECTING A FREQUENCY FOR AN ENERGY METER APPLICATION

As shown in Table II, the user can select one of four frequencies. This frequency selection determines the maximum frequency on F1 and F2. These outputs are intended to be used to drive the energy register (electromechanical or other). Since only four different output frequencies can be selected, the available frequency selection has been optimized for a meter constant of 100 imp/kWhr with a maximum current of between 10 A and 120 A. Table V shows the output frequency for several maximum currents (I_{MAX}) with a line voltage of 220 V. In all cases the meter constant is 100 imp/kWhr.

Table V. F1 and F2 Frequency at 100 imp/kWhr

I _{MAX}	F1 and F2 (Hz)
12.5 A	0.076
25 A	0.153
40 A	0.244
60 A	0.367
80 A	0.489
120 A	0.733

The F₁₋₄ frequencies allow complete coverage of this range of output frequencies on F1 and F2. When designing an energy meter, the nominal design voltage on Channel 2 (voltage) should be set to half scale to allow for calibration of the meter constant. The current channel should also be no more than half scale when the meter sees maximum load. This will allow over current signals and signals with high crest factors to be accommodated. Table VI shows the output frequency on F1 and F2 when both analog inputs are half scale. The frequencies listed in Table VI align very well with those listed in Table V for maximum load.

ADE7755

Table VI. F1 and F2 Frequency with Half-Scale AC Inputs

S1	S0	F ₁₋₄	Frequency on F1 and F2 CH1 and CH2 Half-Scale AC Inputs
0	0	1.7	0.085 Hz
0	1	3.4	0.17 Hz
1	0	6.8	0.34 Hz
1	1	13.6	0.68 Hz

When selecting a suitable F₁ frequency for a meter design, the frequency output at I_{MAX} (maximum load) with a meter constant of 100 imp/kWhr should be compared with Column 4 of Table VI. The frequency that is closest in Table VI will determine the best choice of frequency (F₁₋₄). For example, if a meter with a maximum current of 25 A is being designed, the output frequency on F1 and F2 with a meter constant of 100 imp/kWhr is 0.153 Hz at 25 A and 220 V (from Table V). Looking at Table VI, the closest frequency to 0.153 Hz in column four is 0.17 Hz. Therefore, F₂ (3.4 Hz—see Table II) is selected for this design.

Frequency Outputs

Figure 1 shows a timing diagram for the various frequency outputs. The outputs F1 and F2 are the low-frequency outputs that can be used to directly drive a stepper motor or electromechanical impulse counter. The F1 and F2 outputs provide two alternating low going pulses. The pulsewidth (t₁) is set at 275 ms and the time between the falling edges of F1 and F2 (t₂) is approximately half the period of F1 (t₂). If, however, the period of F1 and F2 falls below 550 ms (1.81 Hz), the pulsewidth of F1 and F2 is set to half of their period. The maximum output frequencies for F1 and F2 are shown in Table III.

The high-frequency CF output is intended to be used for communications and calibration purposes. CF produces a 90 ms-wide active high pulse (t₄) at a frequency proportional to active power. The CF output frequencies are given in Table IV. As in the case of F1 and F2, if the period of CF (t₃) falls below 180 ms, the CF pulsewidth is set to half the period. For example, if the CF frequency is 20 Hz, the CF pulsewidth is 25 ms.

NOTE: When the high-frequency mode is selected, (i.e., SCF = 0, S1 = S0 = 1), the CF pulsewidth is fixed at 18 μs. Therefore, t₄ will always be 18 μs, regardless of the output frequency on CF.

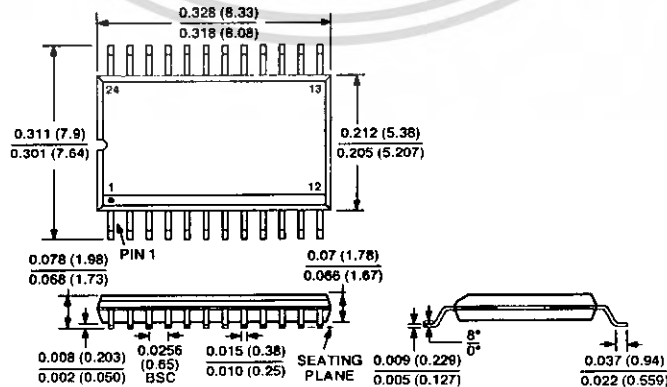
NO LOAD THRESHOLD

The ADE7755 also includes a “no load threshold” and “start-up current” feature that will eliminate any creep effects in the meter. The ADE7755 is designed to issue a minimum output frequency on all modes except when SCF = 0 and S1 = S0 = 1. The no-load detection threshold is disabled on this output mode to accommodate specialized application of the ADE7755. Any load generating a frequency lower than this minimum frequency will not cause a pulse to be issued on F1, F2, or CF. The minimum output frequency is given as 0.0014% of the full-scale output frequency for each of the F₁ frequency selections (see Table II). For example, an energy meter with a meter constant of 100 imp/kWhr on F1 and F2 using F₂ (3.4 Hz), the maximum output frequency at F1 or F2 would be 0.0014% of 3.4 Hz or 4.76 × 10⁻⁵ Hz. This would be 3.05 × 10⁻³ Hz at CF (64 × F1 Hz). In this example, the no-load threshold is equivalent to 1.7 W of load or a start-up current of 8 mA at 220 V. IEC 1036 states that the meter must start up with a load current equal to or less than 0.4% I_b. For a 5A (I_b) meter, 0.4% I_b is equivalent to 20mA. The start-up current of this design therefore satisfies the IEC requirement. As illustrated from this example, the choice of F1-F4 and the ratio of the stepper motor display will determine the start-up current.

OUTLINE DIMENSIONS

Dimensions shown in inches and (mm)

24-Lead Shrink Small Outline Package (RS-24)



1. Introduction

Thank you very much for purchasing Ethernet-232. Please check that all the following items are in the box. If anything is missing or damaged, contact your distributor or Design Gateway Co.,Ltd.

- Ethernet-232 Board
- +5V Power Adapter
- CD (contents : Configuration software, User's Manual)
- User's Manual

Ethernet-232 is an industrial-grade Ethernet-serial device server for connecting one RS-232 port from the equipment like POS, Kiosk, Printer, PLC and other devices directly to Network. The Ethernet-232 adds networking capability to existing equipment by enabling serial communication over longer distances and low addition cost. Ethernet-232 can be configured over Web Browser without platform and distance limitation.

1.1. Summary Feature

1. RS – 232 Serial Interface

- Data Rate : 300 – 115200 bps
- Character : 7 or 8 bits
- Parity : Odd / Even / None
- Stop Bit : 1 or 2 bits
- Control Signal : DTR, DCD, CTS, RTS
- Flow Control : NONE

2. 10 BaseT Ethernet interface

- RJ45 Connector with LED indicator.
- Protocol : ARP, ICMP, TCP/IP, DHCP, HTTP, SNMP

3. Device configuration

- Web Browser.
- Windows Utility.

4. SecurityDevice driver
Optional 128 bit AES Encryption
5. Device driver
Virtual COM Port of Windows OS

1.2. System Requirement

1. Windows 2000 or Windows XP
2. 10 MB of available disk space for Windows 2000 / Windows XP
3. 64 MB of available RAM for Windows 2000
4. 128 MB of available RAM for Windows XP
5. 10/100 Mbps LAN card

1.3. Warranty Policy

1. Product warranty is valid for 1 year from purchasing date.
2. Warranty is void if any modification has been made to this product and any incorrect operation from this manual or warranty sticker is torn or damaged.
3. In order to claim for product exchange or technical support within warranty period, official receipt is required for unregistered customer as an evidence of purchasing whereas official receipt is unnecessary for registered customer (please fill up registration card attached herewith the product and send back to Design Gateway Co.,Ltd).

2. Ethernet-232 Detail

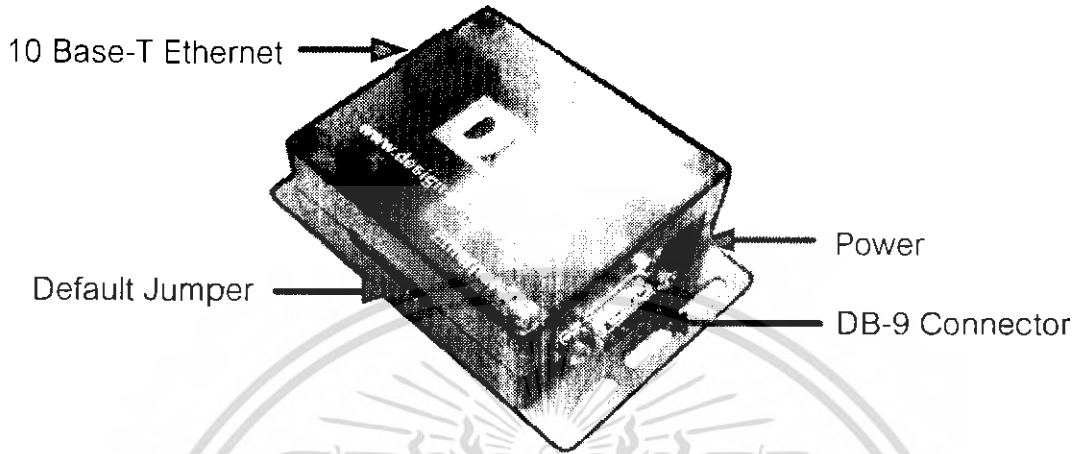


Figure 2-1 Ethernet-232

2.1. Ethernet-232 Description

A high-level block diagram of Ethernet-232 is shown in Figure 2-2 followed by a brief description of connector module.

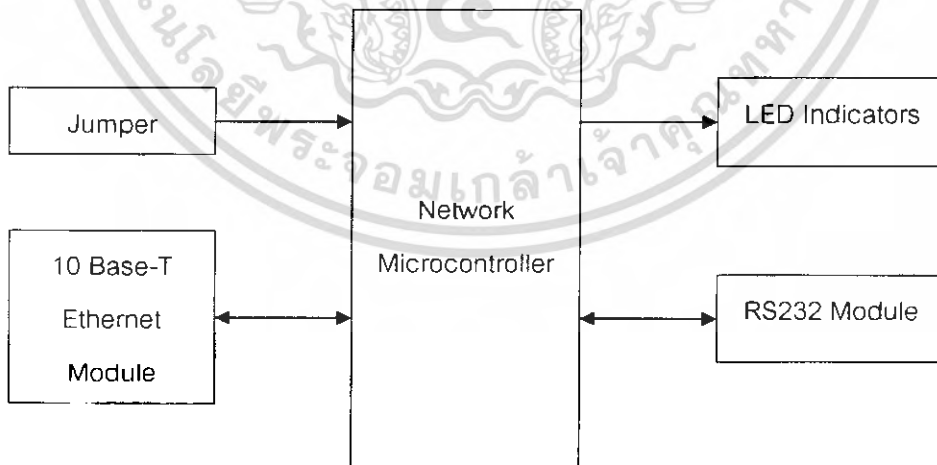


Figure 2-2 Block Diagram of Ethernet-232

2.2. 10 Base-T Ethernet Module

Ethernet-232 provides 10 Base-T Ethernet Module to connect Ethernet-232 with network. Both LAN cable and cross cable can be used with Ethernet-232.

2.3. RS232 Module

Ethernet-232 module provides RS232 module for connection from Ethernet-232 to PC or user's board. The RS232 interface signal is RS232 standard.

2.4. Jumper

Default value or user setting value can be selected by using jumper. If jumper is close, Ethernet-232 use default both IP address and MAC address. Otherwise, if jumper is open, Ethernet-232 use user setting both IP address and Mac address.

2.5. LED Indicator

LED indicator indicates status of power.

3. Quick Start

1. Unplug LAN cable from user's PC.
2. Remove jumper from Ethernet-232.
3. Use cross cable to connect Ethernet-232 board to PC.
4. Set IP Address of user's PC to 192.168.11.xx (Ex. 192.168.11.20 ; except 192.168.11.242 is default IP Address of Ethernet-232)
5. Open Ethernet-232 Configuration Application and then click Connect button.
6. After configuration application establishes connection successfully, click Preference button to change IP Address for IP Address, Subnet Mask and Gateway user's group and then click Save button. (IP Address must not to repeat with other user's IP Address in group)
7. Set IP Address of user's PC to old previous IP Address.

8. Unplug cross cable from user's PC and plug in LAN cable to make connection between user's PC and Ethernet-232.
9. Change IP Address in Ethernet-232 Configuration Application to new IP Address of Ethernet-232 and then click Connect button. User can connect Ethernet-232 board via Ethernet.

4. Configuration Application

Configuration application is software that shows how to use Ethernet-232. This configuration application can help user to configure data to Ethernet-232.

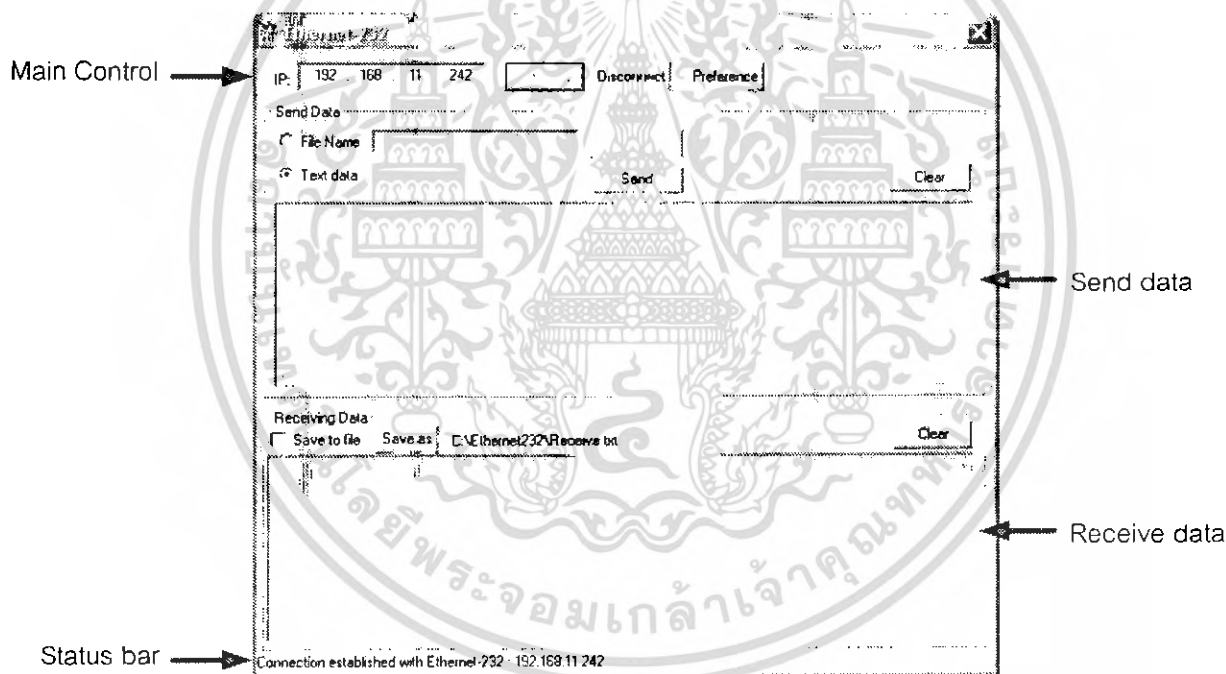


Figure 4-1 Application GUI

4.1. Main Control

Main control is for communication setting between PC and Ethernet-232 and it controls data input / output from Ethernet-232 to user's board.

4.1.1 Communication setting : This section is for network setting and UART setting. Figure 4-2 shows communication setting section.

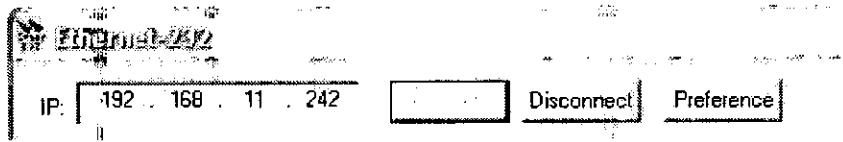


Figure 4-2 Communication setting

- 1). "IP" is IP Address of Ethernet-232 board that user would like to connect.
- 2). "Connect Button" is to start connection between PC and Ethernet-232. After press this button, PC will try to connect to Ethernet-232 board with IP Address as same as IP : box. If PC is successfully connected to Ethernet-232 board, status bar should display "Connection established with Ethernet-232 : (IP Address)" as shown in Figure 4-3. If ,for some reasons, PC fails to connect to Ethernet-232 board, status bar should display "Connection failed" as shown in Figure 4-4.

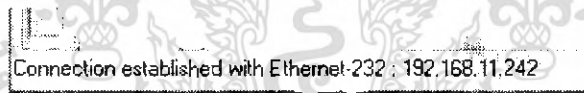


Figure 4-3 Connection from PC to Ethernet-232 completed



Figure 4-4 Connection from PC to Ethernet-232 failed

- 3). "Disconnect Button" is to stop connection between PC and Ethernet-232. After press this button, communication between PC and Ethernet-232 is disconnected.
- 4). "Preference Button" is for setting UART and network. This button will be active after connection between PC and Ethernet-232 is established. After press this button, configuration window will appear. In configuration window, user can set UART properties and network properties as shown in Figure 4-5.

- Baud rate: Can be set from 300 to 115200 bits per second
- Data bits : Can be set 7 or 8 bits data only
- Stop bits : Can be set 1 or 2 stop bits
- Parity bit : Can be none, odd or even parity bit
- Flow control : Can be None
- IP : New IP address that user would like to set to Ethernet-232
- Subnet mask : New subnet mask that user would like to set to Ethernet-232
- Gateway : New gateway that user would like to set to Ethernet-232
- MAC : New MAC address that user would like to set to Ethernet-232

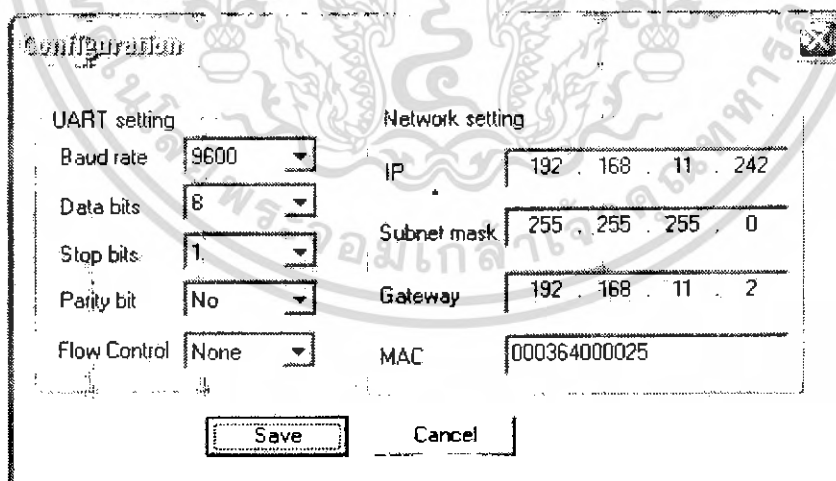


Figure 4-5 UART and Network properties

- Default IP Address is 192.168.11.242
- Default Subnet Mask is 255.255.255.0
- Default Gateway is 192.168.11.2
- Default Mac Address is 0x00, 0x03, 0x64, 0x00, 0x00, 0x25

4.2. Sending Data

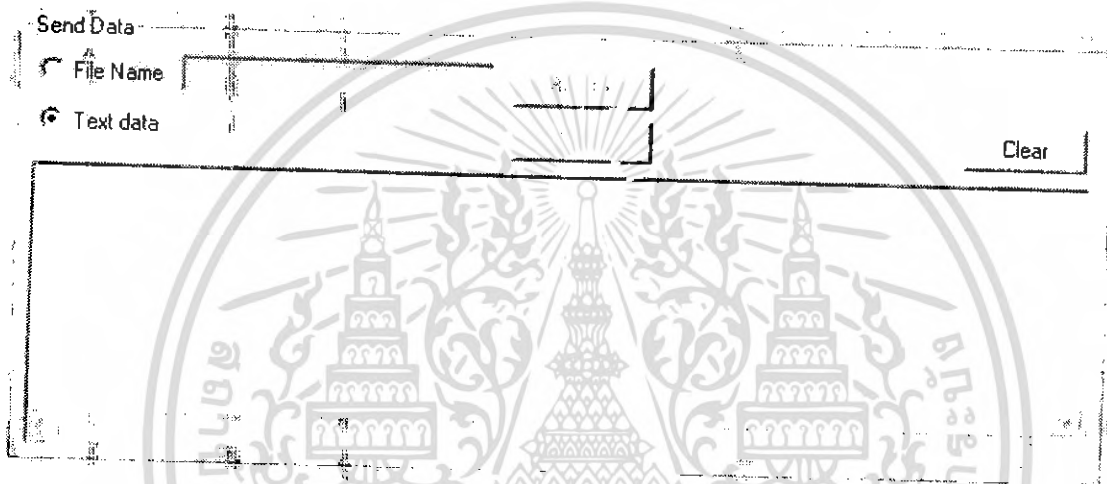


Figure 4-7 Sending Data Section

This section is for sending binary data in the form of text file to UART port of Ethernet-232. Figure 4-7 shows sending data section

User can send both text data and text file. For text data, user can insert text into text box and press "Send button" when user wants to send. For text file, user can browse file and press send button to send. The size of file is limited at 128 Kbytes because serial port is too slow communication. "Clear button" is for deleting a message in text box. Sending Data Section will become active when PC is connected to Ethernet-232. For adds a virtual serial com port for communication, user can use The Hardware Virtual Serial Port (VSP). (Detail in CD or www.design-gateway.com)

4.3. Receiving Data

This section displays text data from UART port. There is a limit of input buffer. User must save and analyze input data before new incoming data. In this example application, user can save the current input data to file and also can use "Clear button" to delete a message from text display. Figure 4-8 shows receiving data section.

The text display has limited size of 64 Kbytes. If data in text display is full and there is new incoming data then it will shift out the old data from text display (deleted) and append a new data.



Figure 4-8 Receiving Data Section

4.4. Status Bar

Status bar will display these messages when application is executing.

- Connection Established : Connect to Ethernet-232 successful
- Connection Failed : Connect to Ethernet-232 fail
- Connection Abandoned : Application loss connection
- Zero Length Message : Sending zero size message
- Unknown Socket Errors : Socket error can not keep connection
- Disconnected : Disconnection of Ethernet-232
- Sending Fail... : Sending message fail
- Open Send File Failed... : Can not open file
- Connecting... : Try to connect to Ethernet-232

4.5. Configuration Application Utility

4.5.1 How to change IP Address :

- 1). Connect configuration application to Ethernet-232.
- 2). After configuration application establishes connection successfully, press "Preference" then new dialog window will appear.
- 3). Change current IP address to new IP address then press "Save".
- 4). Disconnect Ethernet-232, press "Disconnect".
- 5). Remove Jumper and connect to Ethernet-232 again by using new IP address.

4.5.2 How to send text data from Ethernet-232:

- 1). Open HyperTerminal then set and connect HyperTerminal to serial port.
- 2). Connect configuration application to Ethernet-232 and wait until the connection is successfully established.
- 3). Type text data into text box and press "Send" then Ethernet-232 will send text data from UART module.
- 4). Text data will display in HyperTerminal

4.5.3 How to receive text data :

- 1). Connect configuration application to Ethernet-232 and wait until the connection is successfully established.
- 2). Open HyperTerminal and connect them to serial port.
- 3). Type text data on HyperTerminal.
- 4). Text data will be automatically displayed in text display.
- 5). User can save text data to file by press "save as" and browse to directory that would like to save.

5. Web Configuration

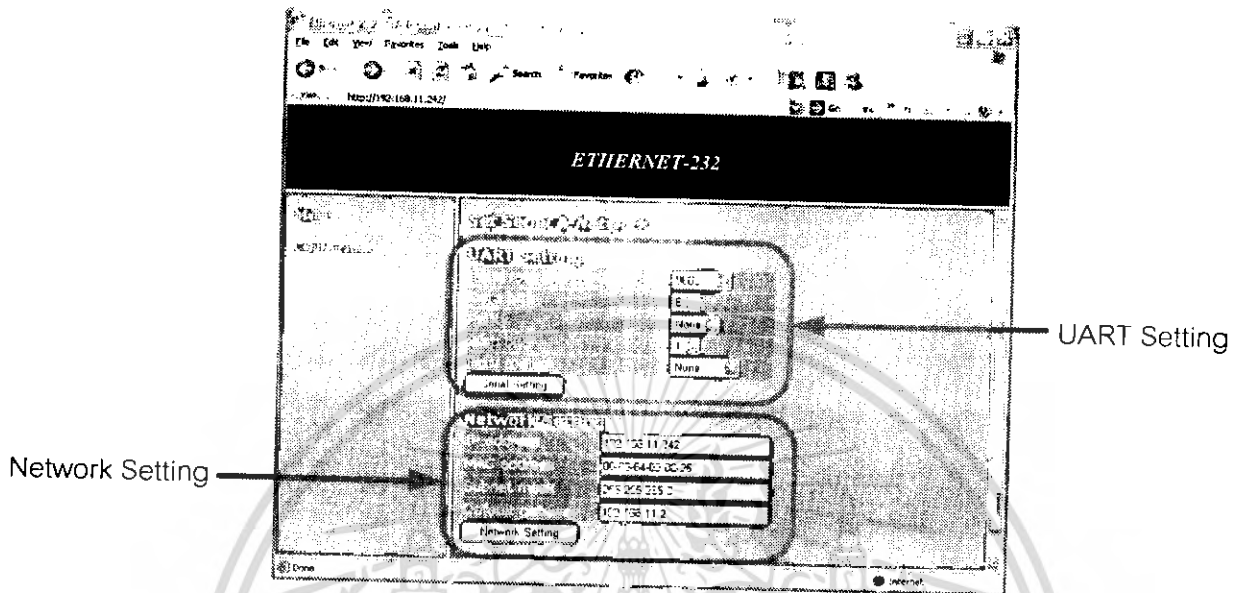


Figure 5-1 Web Configuration.

Web configuration is for setting UART and network. User can use web configuration after connection between PC and Ethernet-232 is established. User can type IP Address of Ethernet-232 in address bar of web browser. After web configuration appears, user can set UART properties and network properties as shown in Figure 5-1.

- Baud rate: Can be set from 300 to 115200 bits per second
- Data bits : Can be set 7 or 8 bits data only
- Stop bits : Can be set 1 or 2 stop bits
- Parity bit : Can be none, odd or even parity bit
- Flow control : Can be None
- IP : New IP address that user would like to set to Ethernet-232
- Subnet mask : New subnet mask that user would like to set to Ethernet-232
- Gateway : New gateway that user would like to set to Ethernet-232
- MAC : New MAC address that user would like to set to Ethernet-232

- Default IP Address is 192.168.11.242
- Default Subnet Mask is 255.255.255.0
- Default Gateway is 192.168.11.2
- Default Mac Address is 0x00, 0x03, 0x64, 0x00, 0x00, 0x25



บรรณานุกรม

- [1] ปิติ เลิศมั่งมี , วีระชัย ถึกสถิต “เครื่องวัดพลังงานไฟฟ้า” วิทยานิพนธ์วิทยาศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2544
- [2] ชัยวัฒน์ ลิ้มพรจิตรวิไล, นคร ภักดีชาติ , 2521 , “ไมโครคอนโทรลเลอร์ MSC – 51 ด้วยโปรแกรมภาษา C ฉบับ P89V51RD2” , จัดพิมพ์และจำหน่ายโดย บริษัท อินโนเวทีฟ เอ็กเพอริเมนต์ จำกัด
- [3] ณรงค์ ขอนตะวัน “เครื่องวัดไฟฟ้า” จัดพิมพ์โดย อมรินทร์การพิมพ์
- [4] วัจนพงศ์ เกษมศิริ , เอกสารประกอบการสอนวิชา 01074201 Network Programming ภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
- [5] ศุภชัย สุรินทร์วงศ์ “เทคโนโลยีไฟฟ้ากระแสสลับ” จัดพิมพ์โดย สมาคมส่งเสริมเทคโนโลยี (ไทย - ญี่ปุ่น)
- [6] สมยศ จุณณะปิยะ , 2546 , “การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ ตระกูล MSC – 51” คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง