การพัฒนาระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์

The Development of a Remote Data Acquisition System Using Microprocessor

วัฒนพงศ์ เกิดทองมี

Wattanapong Kurdthongmee

วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาวิชาฟิสิกส์

มหาวิทยาลัยสงขลานครินทร์

Master of Science Thesis in Physics

Prince of Songkla University

2537

เลขหมู่ Q 385.9 S55 W3 2537 ฉ.2

Bib Key 44346

ชื่อวิทยานิพนธ์ การพัฒนาระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์

ผู้เขียน นายวัฒนพงศ์ เกิดทองมี

สาขาวิชา พิสิกส์

คณะกรรมการที่ปรึกษา

.....ประธานกรรมการ
(รองศาสตราจารย์ ดร.พูลพงษ์ บุญพราหมณ์)

.....กรรมการ
(รองศาสตราจารย์บุญเหลือ พงศ์ดารา)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ชวลิต รอฮาพิริยกุล)

คณะกรรมการสอบ

.....ประธานกรรมการ
(รองศาสตราจารย์ ดร.พูลพงษ์ บุญพราหมณ์)

.....กรรมการ
(รองศาสตราจารย์บุญเหลือ พงศ์ดารา)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ชวลิต รอฮาพิริยกุล)

.....กรรมการ
(อาจารย์สมศักดิ์ เตียวสุรินทร์)

.....กรรมการ
(ดร.ชูศักดิ์ ลิ่มสกุล)

บัณฑิตวิทยาลัยมหาวิทยาลัยสงขลานครินทร์ อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาพิสิกส์

.....

(ดร.ไพรัตน์ สงวนไทร)

คณบดีบัณฑิตวิทยาลัย

ชื่อวิทยานิพนธ์ การพัฒนาระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์

ผู้เขียน นายวัฒนพงศ์ เกิดทองมี

สาขาวิชา ฟิสิกส์

ปีการศึกษา 2536

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้ได้กล่าวถึงการออกแบบ และสร้างระบบเก็บข้อมูลระยะไกลโดยเลือกใช้ไมโครคอนโทรลเลอร์เบอร์ 8031 เป็นตัวควบคุม เพื่อช่วยในงานการเก็บข้อมูลสำหรับการทดลองทางวิทยาศาสตร์ ประกอบด้วยระบบย่อยที่สามารถทำงานได้โดยอิสระ 3 ระบบ คือ ระบบแสดงผล ระบบควบคุมการทำงานหลัก และระบบเครือข่ายย่อยแบบ 1 ช่องรับสัญญาณอนาลอก การเชื่อมโยงของระบบทั้งหมดอยู่ในรูปแบบเครือข่ายของไมโครโปรเซสเซอร์สามารถรับคำสั่งงานและข้อมูลเข้าสู่ไมโครคอมพิวเตอร์เพื่อการวิเคราะห์ผลได้ ระบบเครือข่ายที่รับหรือแปลงข้อมูลจากจุดทดลองสามารถอยู่ห่างจากระบบควบคุมหลักไม่เกิน 1000 เมตร แต่สามารถขยายระยะทางได้โดยใช้ระบบขยายสัญญาณการสื่อสารแบบอนุกรม อัตราเร็วในการส่งผ่านข้อมูลระหว่างระบบควบคุมหลักและระบบเครือข่ายย่อย 9600 บิทต่อวินาที จำนวนชุดของระบบเครือข่ายย่อยที่สามารถต่อกับเครื่องต้นแบบได้มากที่สุด 8 ชุด ความสามารถในการบันทึกข้อมูลแบบ 8 บิทได้ 32 กิโลไบท์ต่อชุดข้อมูลของระบบเครือข่ายย่อย โดยกำหนดอัตราการสุ่มอ่านข้อมูลระหว่าง 200 ไมโครวินาทีถึง 10 นาที

Thesis Title	The Development of a Remote Data Acquisition System Using Microprocessor
Author	Mr.Wattanapong Kurdthongmee
Major Program	Physics
Academic Year	1993

Abstract

This thesis describes a design and implementation of a remote data acquisition system for scientific experriment using 8031 microcontroller. The developed system consists of 3 independent sub-systems:intelligent display systems,master controller system and slave system with one analog signal input channel.The master controller system can be programmed for parameters setting and read from microcomputer for data transfering. The prototype can be connected with 8 remote slave systems with the maximum lenght of network lines less than 1 kilometer and the maximum data communication rate is 9600 bits per second but the lenght of network line can be expanded by using the suitable serial data communication repeater. This develope system can be stored data up to 32 kilobytes per slave channel and the range of sampling rate is between 200 microsecond-10 minute.

กิตติกรรมประกาศ

ผู้เขียนขอแสดงความขอบพระคุณต่อรองศาสตราจารย์ ดร.ชูลพงษ์ บุญพราหมณ์ ประธานกรรมการที่ปรึกษา รองศาสตราจารย์ บุญเหลือ พงศ์คารา และ ผู้ช่วย-ศาสตราจารย์ ชวลิต โอฬาพิริยกุล ซึ่งเป็นกรรมการร่วมที่ได้ช่วยให้คำปรึกษาและตรวจสอบแก้ไขวิทยานิพนธ์ฉบับนี้ จนทำให้วิทยานิพนธ์สำเร็จไปด้วยดี

ขอขอบคุณ ดร.ชูศักดิ์ ลิ่มสกุล และอาจารย์สมศักดิ์ เดียวสุรินทร์ ที่กรุณาตรวจสอบและแก้ไขวิทยานิพนธ์ฉบับนี้ จนบรรลุวัตถุประสงค์

ขอขอบคุณ คุณสนกิบหลี อาหลี ที่ได้ช่วยทำต้นแบบ ของระบบควบคุมในวิทยานิพนธ์ และเจ้าหน้าที่ภาควิชาฟิสิกส์ ที่มีส่วนในการช่วยเหลือให้ยืมอุปกรณ์และเครื่องมือบางอย่าง

ขอขอบคุณ คุณปิยะธิดา ทองใหญ่ ที่ได้ให้กำลังใจในการจัดทำวิทยานิพนธ์

ขอขอบคุณ คุณสวัสดิ์ เอิบโชคชัย อดีตวิศวกรบริษัท อินเตอร์ ฟาร์อิสต์ วิศวการ จำกัด ซึ่งทำให้ผู้เขียนได้รู้จักและสัมผัสกับโลกของไมโครคอนโทรลเลอร์เป็นครั้งแรก

สุดท้ายนี้ผู้เขียนขอน้อมระลึกถึงพระคุณบิดา มารดา และครู อาจารย์ที่ได้ประสิทธิประสาทวิชาความรู้ตลอดจนอุปถัมภ์ในทุกด้านด้วยดีตลอดมา

วัฒนพงศ์ เกิดทองมี

สารบาญ

	หน้า
บทคัดย่อ	(3)
Abstract	(4)
กิตติกรรมประกาศ	(5)
สารบาญ	(6)
รายการตาราง	(9)
รายการภาพประกอบ	(10)
ตัวย่อและสัญลักษณ์	(15)
บทที่	
1 บทนำ	1
วัตถุประสงค์	2
ขอบเขตและวิธีการวิจัย	2
ขั้นตอนและระยะเวลาการดำเนินการวิจัย	3
ประโยชน์ที่คาดว่าจะได้รับ	5
สถานที่ทำการวิจัยและเครื่องมือที่ใช้	6
2 หลักการของระบบเก็บข้อมูลระยะไกล	8
โครงสร้างของระบบเก็บข้อมูลระยะไกล	8
ระบบไมโครโปรเซสเซอร์กับการทดลอง	12
การสื่อสารข้อมูลระหว่างระบบไมโครโปรเซสเซอร์	17
รูปแบบการสื่อสารข้อมูลแบบอนุกรมสำหรับระบบเครือข่าย	20
ข้อแนะนำในการเลือกใช้รูปแบบของระบบเครือข่ายในการสื่อสารข้อมูล	24
รูปแบบการสื่อสารข้อมูลแบบอนุกรมมาตรฐาน RS-232C และ RS-422	25
รูปแบบการทำงานของระบบเก็บข้อมูลระยะไกลในงานวิทยานิพนธ์	27

		หน้า
3	สถาปัตยกรรมด้านฮาร์ดแวร์	31
	ไมโครโปรเซสเซอร์และการออกแบบระบบ ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ของบริษัท INTEL CORP.	31
		35
	คำแนะนำในการออกแบบระบบไมโครคอนโทรลเลอร์	36
	รูปแบบการแสดงผลของระบบไมโครคอนโทรลเลอร์	37
	ระบบแสดงผล	40
	ส่วนขยายหน่วยความจำ	54
	ระบบควบคุมการทำงานหลัก	60
	ระบบเครือข่ายย่อย	77
4	โปรแกรมควบคุมระบบงาน	84
	ภาษาคอมพิวเตอร์ที่เลือกใช้และรูปแบบของโปรแกรม ควบคุมไมโครคอนโทรลเลอร์ 8031 โดยทั่วไป	84
	การบริหารหน่วยความจำส่วนข้อมูลแบบ RING หรือ CIRCULAR BUFFER	92
	โครงสร้างของโปรแกรมควบคุมระบบแสดงผลแบบอิสระ	95
	โครงสร้างของโปรแกรมควบคุมระบบเครือข่ายย่อย	100
	โครงสร้างของโปรแกรมควบคุมระบบควบคุมการ ทำงานหลัก	109
5	บทสรุป	125
	สรุปผลการวิจัย	127
	ข้อเสนอแนะ	156
	คำแนะนำในการดำเนินงานต่อไป	156
6	บรรณานุกรม	158
7	ภาคผนวก	159
	ระบบพัฒนาไมโครโปรเซสเซอร์ตระกูล INTEL MCS-51	159

หน้า

ระบบแสดงผลแบบแยกอิสระ 185

รูปแบบคำสั่งสำหรับระบบเก็บข้อมูลระยะไกลโดยใช้
ไมโครโปรเซสเซอร์ 196

การใช้งานระบบเก็บข้อมูลระยะไกลโดยใช้
ไมโครโปรเซสเซอร์ 202

8 ประวัติผู้เขียน 208

รายการตาราง

ตาราง		หน้า
2.1	ปริมาณพื้นฐานและสิ่งที่สามารถวัดค่าได้	13
2.2	ตารางเปรียบเทียบข้อดีและข้อเสียของมาตรฐานการสื่อสารข้อมูลที่มีใช้ในเครื่องมือวิจัย	19
2.3	ข้อเปรียบเทียบการเลือกรูปแบบเครือข่าย	25
2.4	สายสัญญาณมาตรฐาน RS-232C ที่จำเป็นในการเชื่อมโยงระบบ	26
3.1	ตารางแสดงรีจิสเตอร์ของชิพนาฬิกาจริงและรายละเอียด	70
4.1	ตารางแสดงแหล่งกำเนิดอินเตอร์รัพท์และชื่อตำแหน่งคำสั่งของไมโครคอนโทรลเลอร์ MCS-51	89
4.2	ตารางแสดงตัวอย่างค่าเริ่มต้นของวงจรนับฐานเวลา 60 เฮิร์ต และอัตราการสุ่มอ่านข้อมูลในรูปแบบการทำงานของระบบเก็บข้อมูลเวลาจริง	112

รายการภาพประกอบ

ภาพประกอบ		หน้า
2.1	โครงสร้างสมบูรณ์ของระบบเก็บข้อมูลระยะไกล	10
2.2	ไมโครคอนโทรลเลอร์กับการทดลอง	12
2.3	ระบบไมโครคอนโทรลเลอร์ในรูปแบบของระบบเวลาจริง	17
2.4	รูปแบบของระบบเครือข่ายแบบ STAR	21
2.5	รูปแบบของระบบเครือข่ายแบบ LOOP	22
2.6	รูปแบบของระบบเครือข่ายแบบ LOOP-STAR	23
2.7	รูปแบบของระบบเครือข่ายแบบ STAR-LOOP	24
2.8	แผนภูมิสายงานการทำงานโดยสังเขปของระบบเก็บข้อมูลโดยทั่วไป	28
3.1	องค์ประกอบของไมโครโปรเซสเซอร์	32
3.2	องค์ประกอบของไมโครคอนโทรลเลอร์	34
3.3	วงจรควบคุมการแสดงผลของ LED แบบตัวเลขโดยวิธี TIME-MULTIPLEX	39
3.4	ภาคแสดงผล LED แบบดอทเมทริกซ์ขนาด 5x7	40
3.5	ตัวอย่างวงจรควบคุมการแสดงผล LED แบบดอทเมทริกซ์ 1 ตัว	42
3.6	แผนผังเวลาเปรียบเทียบของการเกิดสัญญาณควบคุมสวิทชิ่งเพาเวอร์ทรานซิสเตอร์	43
3.7	รูปแบบการถอดระหัสบัสแอดเดรสเพื่ออ้างอิงพอร์ทโดยตรง	45
3.8	รูปแบบของวงจรชิฟรีจิสเตอร์เอาท์พุทแบบขนานขนาด 24 บิท	46

ภาพประกอบ หน้า

3.9 วงจรชิฟรีจิสเตอร์แบบขนานขนาด 48 บิทที่ใช้ในการส่งข้อมูลให้กับภาคแสดงผลแบบคอทเมทริกซ์ 47

3.10 วงจรส่วนรับข้อมูลเพื่อแสดงผลของระบบแสดงผล 50

3.11 วงจรสมบูรณ์ของระบบแสดงผล 51

3.12 วงจรสมบูรณ์ของระบบแสดงผล(ต่อ) 52

3.13 วงจรสมบูรณ์ของระบบแสดงผล(ต่อ) 53

3.14 วงจรของการบริหารหน่วยความจำส่วนขยายขนาด 128 กิโลไบท์ของ 8031 56

3.15 การถอดรหัสพอร์ท P1 เพื่อเพิ่มหน่วยความจำส่วนขยายของรูปที่ 3.14 57

3.16 สภาวะของสัญญาณในการเชื่อมโยงระหว่างระบบภายนอกกับระบบเก็บข้อมูลส่วนขยายแบบอิสระ 58

3.17 วงจรควบคุมของระบบเก็บข้อมูลส่วยขยายโดยใช้ไมโครคอนโทรลเลอร์อิสระ 59

3.18 แผนผังการทำงานของระบบควบคุมการทำงานหลัก 60

3.19 วงจรคัดเลือกช่องสัญญาณส่งข้อมูลของการสื่อสารข้อมูลแบบอนุกรมแบบเลือกส่งข้อมูลออกทุกช่องสัญญาณได้ 63

3.20 วงจรจัดการเครือข่าย 65

3.21 วงจรส่วนจัดการแสดงผลของระบบควบคุมหลัก 66

3.22 รูปแบบเมทริกซ์ของสวิทช์ชนิดกลไกสัมผัสชนิด 16 แป้นพิมพ์ 68

3.23 วงจรส่วนจัดการนาฬิกาเวลาจริง 72

3.24 วงจรกำเนิดสัญญาณความถี่ 60 เฮิร์ต 73

3.25 วงจรสมบูรณ์ของระบบควบคุมการทำงานหลัก 74

3.26 วงจรสมบูรณ์ของระบบควบคุมการทำงานหลัก 75

ภาพประกอบ	หน้า
3.27 วงจรสมบูรณ์ของระบบควบคุมการทำงานหลัก	76
3.28 แผนผังขอบเขตการทำงานของระบบเครือข่ายย่อย	78
3.29 วงจรเชื่อมต่อระหว่างชิพแปลงสัญญาณอนาลอกเป็นดิจิตอลกับไมโครคอนโทรลเลอร์	80
3.30 วงจรเปลี่ยนมาตรฐานการสื่อสารข้อมูลระหว่าง RC-232C และ RS-422	83
4.1 แผนภูมิขั้นตอนการพัฒนาระบบควบคุมโดยใช้ภาษาแอสเซมบลี	87
4.2 การเกิดอินเตอร์รัพท์และการดำเนินไปของโปรแกรมควบคุมการทำงานของไมโครคอนโทรลเลอร์หรือไมโครโปรเซสเซอร์	91
4.3 แผนภูมิสายงานของการบริหารหน่วยความจำส่วนข้อมูลโดยใช้ RING หรือ CIRCULAR BUFFER	94
4.4 แผนภูมิสายงานการบริการอินเตอร์รัพท์ส่วนแสดงผลข้อมูล	98
4.5 แผนภูมิสายงานส่วนการถอดระหัสเพรมข้อมูลเพื่อเลือกบรรทัดและรูปแบบการแสดงผลของระบบแสดงผลอิสระ	99
4.6 แผนภูมิสายงานของการใช้บริการอินเตอร์รัพท์จากภายนอกหมายเลข 0 ของระบบเครือข่ายย่อย	102
4.7 แผนภูมิสายงานของโปรแกรมส่วนบริการอินเตอร์รัพท์แบบอนุกรมของระบบเครือข่ายย่อย	104
4.8 สายงานโปรแกรมบริการอินเตอร์รัพท์จากฐานเวลาสำหรับระบบเครือข่ายย่อย	105
4.9 แผนภูมิสายงานส่วนขยายสำหรับโปรแกรมควบคุมหลักของระบบเครือข่ายย่อย	106

ภาพประกอบ หน้า

4.10 แผนภูมิสายงานสำหรับโปรแกรมหลักของระบบเครือข่ายย่อยในการส่งข้อมูลปัจจุบัน 107

4.11 แผนภูมิสายงานสำหรับโปรแกรมควบคุมหลักของระบบเครือข่ายย่อยในการส่งข้อมูลในหน่วยความจำจากจุดเริ่มต้นจนถึงตำแหน่งปัจจุบัน 108

4.12 แผนภูมิสายงานของโปรแกรมให้บริการอินเตอร์รัพท์แก่ระบบนาฬิกาเวลาจริงของระบบควบคุมหลัก 111

4.13 แผนภูมิสายงานของโปรแกรมให้บริการอินเตอร์รัพท์จากฐานเวลา ซึ่งกำหนดอัตราการสุ่มอ่านมาตรฐานสำหรับระบบควบคุมการทำงานหลัก 113

4.14 แผนภูมิสายงานส่วนเริ่มต้นและการเรียกโปรแกรมย่อยของสายงานโปรแกรมหลักของระบบควบคุมการทำงานหลัก 115

4.15 แผนภูมิสายงานโปรแกรมย่อยในการสั่งงาน และติดต่อเพื่ออ่านข้อมูลโดยพื้นฐานของระบบควบคุมการทำงานหลัก 120

4.16 แผนภูมิสายงานโปรแกรมย่อยในการสั่งงาน และติดต่อเพื่ออ่านข้อมูลของระบบควบคุมการทำงานหลักในรูปแบบเวลาจริง 121

4.17 แผนภูมิสายงานโปรแกรมย่อยในการสั่งงาน และติดต่อเพื่ออ่านข้อมูลของระบบควบคุมการทำงานหลักในรูปแบบไม่เป็นเวลาจริง 122

4.18 แผนภูมิสายงานโปรแกรมย่อยสำหรับการจัดกระทำเมนูในระบบควบคุมการทำงานหลัก 123

4.19 โปรแกรมย่อยในการตรวจสอบสถานะแป้นพิมพ์ 124

ภาพประกอบ หน้า

5.1 รูปแบบจัดระบบเก็บข้อมูลระยะไกลเพื่อการทดสอบการทำงาน 128

5.2-5.24 กราฟแสดงผลการทดสอบการสุ่มอ่านสัญญาณในรูปแบบและความถี่ต่าง ๆ 129-155

ตัวย่อและสัญลักษณ์

การเก็บข้อมูลในรูปแบบเวลาจริง	REAL-TIME DATA ACQUISITION
การเก็บข้อมูลในรูปแบบไม่เป็นเวลาจริง	NON REAL-TIME DATA ACQUISITION
การติดต่อกับหน่วยความจำโดยตรง	DIRECT MEMORY ACCESS
กระแสสูงสุด	PEAK CURRENT
ข้อมูลดิจิตอล	DIGITAL DATA
ข้อมูลแบบขนาน	PARALLEL DATA
ข้อมูลแบบอนุกรม	SERIAL DATA
ขั้นตอนการทำงาน	ALGORITHMIC PROCESSES
ความแม่นยำของการแปลงค่า	ACCURACY
ความเร็วในการแปลงสัญญาณ	CONVERSION TIME
ความเร็วในการสุ่มอ่านสัญญาณ	SAMPLING RATE TRIGGERING RATE
ความละเอียดของการแปลงค่า	RESOLUTION
ความแตกต่างของเหตุการณ์	ASYNCRONOUS EVENT
ความต้องการกำลังไฟฟ้า	POWER CONSUMPTION
ความเป็นเชิงเส้นของการทำงาน	LINEARITY
ความสูงของสัญญาณ	SIGNAL AMPLITUDE
คำสั่งในโปรแกรมงาน	SYNTAX
ชิพแปลงสัญญาณอนาลอกเป็นดิจิตอล	ANALOG TO DIGITAL CONVERTER
ไดโอดเปล่งแสง	LIGH EMITTING DIODE (LED)
ตัวชี้ตำแหน่งคำสั่ง	PROGRAM COUNTER

แผนภูมิสายงาน	FLOW CHART
เฟรมข้อมูล	DATA FRAME
ภาษาเครื่อง	MACHINE CODE
ไมโครคอมพิวเตอร์	MICROCOMPUTER
ไมโครโปรเซสเซอร์	MICROPROCESSOR
ระบบเครือข่าย	NETWORK
ระบบเครือข่ายย่อย	NODE
ระบบจัดการสัญญาณ	SIGNAL CONDITIONER
ระบบพัฒนาไมโครโปรเซสเซอร์แบบฮาร์ดแวร์	EMULATOR
ระหัสแอสกี	ASCII
รูปแบบการอ้างอิงตำแหน่งข้อมูล	ADDRESSING MODE
วงจรขยายแบบกลับขั้วสัญญาณ	INVERTING AMPLITUDE
วงจรขยายแบบไม่กลับขั้วสัญญาณ	NON-INVERTING AMPLITUDE
วงจรรวม	CHIP,IC(INTEGRATED CIRCUIT)
สัญญาณตอบรับ	ACKNOWLEDGE
สัญญาณแบบอนาลอก	ANALOG SIGNAL
หน่วยความจำ	MEMORY
หน่วยความจำส่วนเก็บข้อมูล	RAM(RANDOM ACCESS MEMORY)
หน่วยความจำส่วนเก็บโปรแกรมงาน	EPROM(ERASABLE-PROGRAMMABLE READ ONLY MEMORY)
หน่วยประมวลผลกลาง	CENTRAL PROCESSING UNIT
แหล่งจ่ายกำลังไฟฟ้า	POWER SUPPLY

อันดับของการให้บริการอินเตอร์รัพท์	INTERRUPT PRIORITY
อินเตอร์รัพท์	INTERRUPT

บทที่ 1

บทนำ

การที่นักวิจัยจะได้มาซึ่งข้อมูลที่มีความจำเป็นในงานวิจัย ในบางครั้งจำเป็นต้องเสี่ยงกับอันตรายอันเกิดจากสารเคมี สารรังสี ทั้งที่สามารถรับรู้ได้ด้วยประสาทสัมผัส และอยู่นอกเหนือจากการรับรู้ นอกจากนั้นในบางครั้งนักวิจัยจำเป็นต้องมีการสำรวจเพื่อเก็บข้อมูลในระยะไกลหรือครอบคลุมบริเวณกว้าง ไม่สะดวกในการใช้วิธีการเก็บบันทึกค่าตัวแปรตามรูปแบบปกติในห้องปฏิบัติการ และไม่คุ้มค่าในการใช้คอมพิวเตอร์เป็นตัวควบคุมและเก็บข้อมูล ดังนั้นจึงมีแนวคิดที่จะพัฒนาระบบเก็บข้อมูลที่ให้ความยืดหยุ่นในการพัฒนาไปใช้งานเพื่อเก็บข้อมูลในรูปแบบต่างๆ เช่น ในงานการสำรวจโครงสร้าง และการวางตัวของชั้นหินธรณีวิทยาที่จำเป็นต้องเก็บข้อมูลคลื่นสั่นสะเทือน (SHOCK WAVE) ที่กระจายออกไปในบริเวณกว้าง สามารถใช้ระบบเก็บข้อมูลอิสระที่เชื่อมต่อกับระบบวัดสัญญาณ วางในบริเวณที่ต้องการสำรวจเพื่อเก็บค่าสัญญาณความสูงของคลื่นภายใต้การควบคุมของระบบควบคุมหลักก็สามารถทำให้ได้ข้อมูลโดยใช้แรงงานน้อยกว่า สามารถกำหนดความละเอียดของข้อมูลได้ นอกจากนั้นงานทางนิวเคลียร์ฟิสิกส์ที่จำเป็นต้องเสี่ยงกับความเข้มของรังสีที่สูง ไม่สามารถให้ผู้ปฏิบัติการวิเคราะห์ได้โดยตรง ก็สามารถพัฒนาระบบเพื่อช่วยเหลือในการวัดระดับความเข้มของสารรังสีที่จุดต่างๆ ในเวลาต่างๆ กันได้

ในปัจจุบันนี้ได้มีการนำเอาเทคโนโลยีทางด้านคอมพิวเตอร์ เช่น ไมโครโปรเซสเซอร์มาใช้ในการควบคุมระบบต่าง ๆ ที่เกี่ยวข้องกับการทดลอง ซึ่งช่วยเสริมประสิทธิภาพในการทำงานของระบบการทดลอง และยังลดขนาดของอุปกรณ์หรือเครื่องมือที่ใช้ในการทดลอง ดังนั้น จึงเป็นการดีที่จะได้ทดลองพัฒนาระบบภายใต้การควบคุมของไมโครโปรเซสเซอร์เพื่อใช้ในงานทางวิทยาศาสตร์

การวิจัยครั้งนี้ ผู้วิจัยพยายามออกแบบระบบให้เป็นระบบเปิด ภายใต้มาตรฐานของการเชื่อมโยงระบบ เช่น การเชื่อมต่อผ่านพอร์ทแบบขนาน และแบบอนุกรม เพื่อให้ผู้วิจัยท่านอื่นสามารถนำไปเชื่อมโยงกับระบบที่พัฒนาขึ้นเองได้ เป็น

การประหยัดเวลาที่จะต้องออกแบบระบบขึ้นมาใหม่ ดังนั้น นอกจากผลสำเร็จของงานวิจัยนี้จะได้ระบบเก็บข้อมูลระยะไกลแล้วยังได้ระบบอิสระอื่น ๆ ที่คาดว่าจะเป็นประโยชน์ต่อการพัฒนาระบบอื่น ๆ ต่อไป

วัตถุประสงค์

1. เพื่อสร้างและพัฒนาเครื่องต้นแบบสำหรับควบคุมการเก็บข้อมูลระยะไกล ที่ใช้งานได้โดยอิสระภายใต้การควบคุมไมโครโปรเซสเซอร์โดยสามารถเชื่อมโยงระบบไมโครคอมพิวเตอร์เพื่อส่งข้อมูลเข้าวิเคราะห์
2. เพื่อพัฒนาระบบย่อยเครือข่าย ที่ทำหน้าที่ในการเปลี่ยนรูปแบบของข้อมูลจากปริมาณทางฟิสิกส์ ให้อยู่ในรูปแบบทางดิจิตอลที่มีความยืดหยุ่นในการดัดแปลง และเปลี่ยนแปลงโปรแกรมได้ ทำการเชื่อมโยงกับระบบควบคุมการเก็บข้อมูลหลัก
3. เพื่อทดสอบเครื่องต้นแบบที่ได้กับการใช้งาน เพื่อวิเคราะห์ผลสำหรับงานวิจัยจริง
4. เพื่อพัฒนาระบบอิสระภายใต้การควบคุมโดยไมโครโปรเซสเซอร์ เพื่อที่นักวิจัยท่านอื่นสามารถนำไปดัดแปลงใช้งานกับงานของตนได้โดยง่าย
5. เพื่อออกแบบและสร้างระบบพัฒนาไมโครโปรเซสเซอร์ตระกูล MCS-51 ของบริษัท INTEL(MCS-51 EMULATOR) ทั้งทางด้านซอร์ฟแวร์และฮาร์ดแวร์เพื่อนำไปใช้งานพัฒนาระบบอื่น ๆ ที่มีประโยชน์ต่อไป

ขอบเขตและวิธีการวิจัย

เนื่องจากการที่จะสร้างระบบใด ๆ โดยใช้ไมโครโปรเซสเซอร์เป็นตัวควบคุมนั้น ผู้วิจัยจำเป็นต้องศึกษาการทำงานของระบบไมโครโปรเซสเซอร์นั้น ๆ ก่อน ดังนั้นขั้นตอนการดำเนินการวิจัยจึงสามารถสรุปเป็นตอน ๆ ได้ดังนี้

1. ศึกษารายละเอียดของระบบไมโครโปรเซสเซอร์ตระกูล MCS-51 ของบริษัท INTEL CORP. โดยเน้นที่ไมโครคอนโทรลเลอร์เบอร์ 8031 ทั้งรายละเอียดทาง

ด้านซอร์ฟแวร์และฮาร์ดแวร์ เพื่อพัฒนาและสร้างระบบพัฒนาไมโครโปรเซสเซอร์ MCS-51(MCS-51 EMULATOR) ซึ่งจะนำไปพัฒนาระบบอื่น ๆ ที่ควบคุมโดยไมโครโปรเซสเซอร์ตระกูลนี้ทั้งหมดที่เกี่ยวข้องกับงานวิจัยต่อไป

2. ศึกษารายละเอียดของตัวแปรต่าง ๆ ที่จำเป็นต้องมีการบันทึกในการประมวลผลของงานวิจัยโดยทั่ว ๆ ไปและรูปแบบของการอ่านค่าข้อมูล ตลอดจนความเร็วในการอ่านค่าตัวแปร เพื่อที่จะได้ออกแบบระบบให้สามารถเลือกรูปแบบการทำงานครอบคลุมตามความต้องการมากที่สุด

3. ออกแบบสร้างและทดสอบระบบควบคุมหลักตลอดจนอุปกรณ์เชื่อมโยงที่เกี่ยวข้องเพื่อให้สามารถรับค่าตัวแปรสั่งงานระบบเครือข่ายย่อยต่าง ๆ ได้

4. ศึกษารายละเอียดของการเปลี่ยนรูปแบบสัญญาณจากสัญญาณที่ได้ จากการวัดค่าทางฟิสิกส์ให้อยู่ในรูปแบบที่สามารถจัดเก็บโดยไมโครโปรเซสเซอร์ได้ และออกแบบระบบเครือข่ายย่อยที่ทำหน้าที่ควบคุม และการส่งข้อมูลระหว่างระบบควบคุมหลักและระบบเครือข่ายย่อย

5. ศึกษามาตรฐานการติดต่อรับส่งข้อมูลระยะไกล และเลือกใช้รูปแบบของการสื่อสารที่เหมาะสมเพื่อใช้เป็นมาตรฐานสำหรับการควบคุม และการส่งข้อมูลระหว่างระบบควบคุมหลักและระบบเครือข่ายย่อย

6. ทดสอบการเชื่อมต่อของระบบต่างๆ เข้าเป็นระบบเครือข่ายทดสอบการทำงานรวมของระบบและแก้ไขข้อผิดพลาดของการทำงาน

ขั้นตอนและระยะเวลาการดำเนินการวิจัย

1. **ขั้นตอนการวิจัย**

1.1 ศึกษาสถาปัตยกรรมของไมโครโปรเซสเซอร์ตระกูล INTEL MCS-51 โดยเน้นหนักที่ไมโครคอนโทรลเลอร์เบอร์ 8031

1.2 ศึกษาการเขียนโปรแกรมควบคุมการทำงาน ของไมโครคอนโทรลเลอร์เบอร์ 8031(8031 MICROCONTROLLER)

1.3 ออกแบบ และสร้างระบบพัฒนาไมโครคอนโทรลเลอร์ตระกูล MCS-51 (MCS-51 EMULATOR) ทั้งทางด้านซอร์ฟแวร์ช่วยเหลือในการพัฒนา และฮาร์แวร์ของระบบพัฒนา

1.4 หาประสบการณ์ในการพัฒนาระบบอิสระ ที่ควบคุมโดยไมโครโปรเซสเซอร์ทั้งทางด้านซอร์ฟแวร์ และฮาร์ดแวร์รวมทั้งวิธีการแก้ไขปัญหาของระบบไมโครโปรเซสเซอร์โดยเน้นหนักที่ไมโครคอนโทรลเลอร์เบอร์ 8031

1.5 ศึกษารูปแบบเปลี่ยนแปลงข้อมูลจากสัญญาณอนาลอก(ANALOG SIGNAL) เป็นข้อมูลแบบดิจิตอล (DIGITAL DATA) แนวการออกแบบ การขยายสัญญาณ การลดสัญญาณและทดลองออกแบบจริง

1.6 พัฒนาโปรแกรมจัดเก็บข้อมูล บนไมโครคอนโทรลเลอร์เบอร์ 8031

1.7 ศึกษาและพัฒนาระบบเชื่อมต่อแบบหลายโปรเซสเซอร์โดยใช้ไมโครคอนโทรลเลอร์(8031 MULTIPROCESSOR SYSTEM)

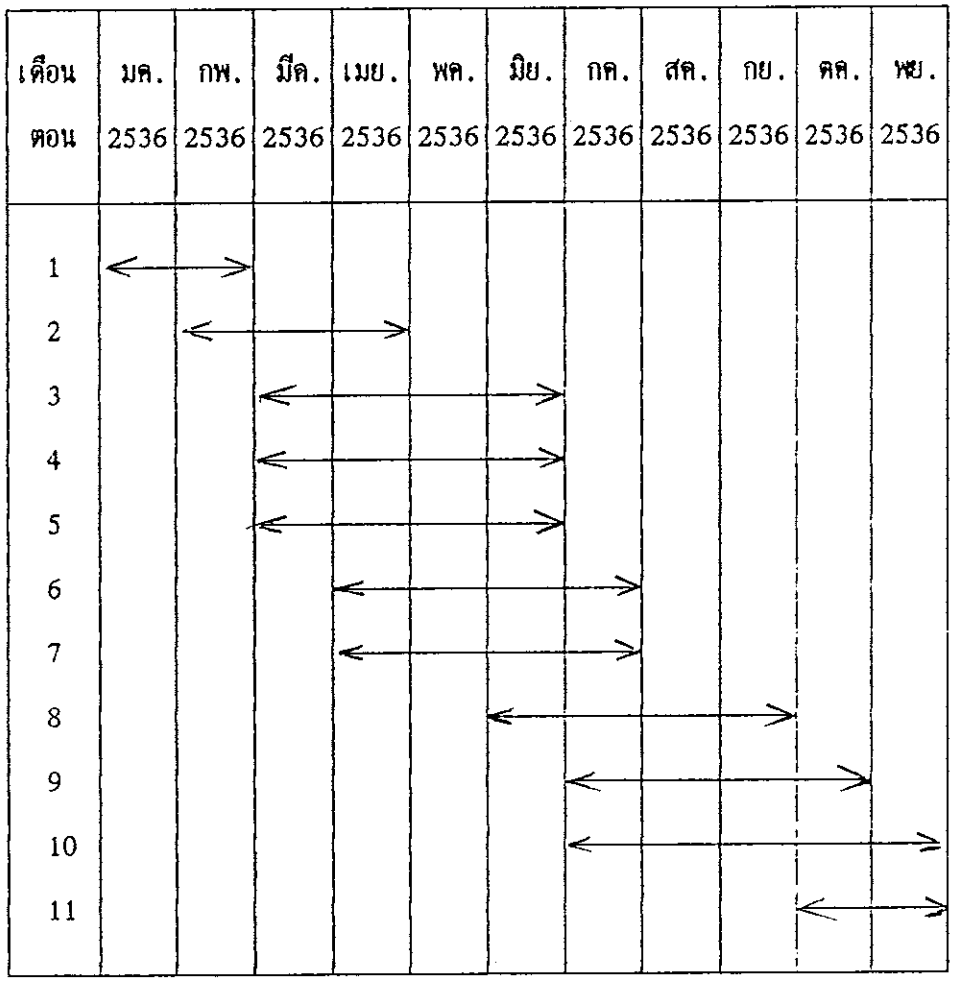
1.8 ศึกษาและพัฒนาการเชื่อมต่อแบบเครือข่ายระยะไกลของระบบอิสระย่อยโดยทิศทางของการส่งและรับข้อมูลเป็นแบบ 2 ทิศทาง ภายใต้การควบคุมของระบบควบคุมหลัก

1.9 เชื่อมต่อระบบจากขั้นตอนที่ 1.5 และ 1.6 ให้เป็นระบบเก็บข้อมูลจากการทดลองย่อย

1.10 เชื่อมต่อหลาย ๆ ระบบจากขั้นตอนที่ 9 กับขั้นตอนที่ 8 และทดสอบการทำงานตลอดจนปรับปรุงแก้ไขความผิดพลาด

1.11 ทำรายงานวิทยานิพนธ์

2.ระยะเวลาดำเนินงานวิจัย



ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถสร้างระบบเก็บข้อมูลระยะไกล ที่สามารถทำงานได้โดยอิสระภายใต้การควบคุมของไมโครโปรเซสเซอร์ ซึ่งมีขีดความสามารถในการควบคุมระบบเครือข่ายย่อยได้มากถึง 8 ตัว

2. สามารถสร้างระบบเก็บข้อมูลย่อยที่สามารถสั่งงาน และส่งข้อมูลเข้าสู่ระบบควบคุมหลักหรือเข้าสู่ไมโครคอมพิวเตอร์โดยตรง

3. ผู้วิจัยอื่น ๆ สามารถนำระบบย่อยที่เป็นอุปกรณ์ประกอบ มาพัฒนาเชื่อมโยงกับระบบที่พัฒนาขึ้นเอง และใช้งานได้เนื่องจากในงานวิจัยจะพยายามออกแบบระบบย่อยแบบเปิดที่สามารถเชื่อมโยงโดยอาศัยมาตรฐานต่าง ๆ ได้เช่น ระบบการแสดงผล ระบบเก็บข้อมูลชั่วคราว

สถานที่ทำการวิจัยและเครื่องมือที่ใช้

1. **สถานที่ทำการวิจัย**

   ภาควิชาฟิสิกส์ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่

   บริษัท ไบร์ท ซิสเท็ม จำกัด อำเภอหาดใหญ่ สงขลา

2. **เครื่องมือที่ใช้**

   2.1 เครื่องคอมพิวเตอร์ที่ใช้ IBM PC/AT COMPATIBLE หน่วยความจำอย่างน้อย 640 KB อย่างน้อย 1 DISK DRIVE พร้อม HARD DISK 40 MBYTE

   2.2 การ์ดแสดงผล ระดับ EGA CARD ขึ้นไป

   2.3 ออสซิลโลสโคป 20 MHz ขึ้นไป

   2.4 เครื่องจ่ายแรงดันไฟตรง 5 VOLT และ 12 VOLT

   2.5 ดิจิตอลมัลติมิเตอร์

3. **ภาษาคอมพิวเตอร์และซอร์ฟแวร์ที่ใช้**

   3.1 ภาษาซี โดยใช้คอมไพเลอร์ของ BORLAND INTERNATIONAL (BORLANDC++ VERSION 2.0)

   3.2 MCS-51 ASSEMBLER ของบริษัท BINARY TECHNOLOGY INC.

   3.3 8031 C COMPILER AND ASSEMBLER ของบริษัท AVOCET INC.

   3.4 Q-EDITOR VERSION 2.1 ของบริษัท SEMWARE

3.5 โปรแกรมสำหรับเขียนวงจรและออกแบบลายวงจร TAGNO ของบริษัท ACCEL TECHNOLOGIES INC.

บทที่ 2

หลักการของระบบเก็บข้อมูลระยะไกล

งานจัดเก็บบันทึกข้อมูลเป็นงานที่มีความละเอียดอ่อนจนในบางครั้งนักวิจัยอาจมองข้ามความสำคัญในส่วนนี้ไป หากพิจารณาโดยถี่ถ้วนแล้วจะพบว่ามีส่วนสำคัญยิ่งต่องานวิจัย เพราะถึงแม้ว่าระบบการทดลองมีความสมบูรณ์ เครื่องที่ใช้ในการทดลองเป็นเครื่องมือที่มีมาตรฐานแต่หากมีความผิดพลาดหรือความไม่สะดวกในระบบการเก็บบันทึกข้อมูลก็อาจทำให้งานวิจัยล่าช้าไปกว่าความจำเป็น ตัวอย่างงานวิจัยที่สามารถเห็นความสำคัญของการเก็บบันทึกข้อมูลที่ต้องใช้เวลาและแรงงาน เพื่อให้ได้มาซึ่งข้อมูลสำหรับการวิเคราะห์ผลมี อาทิเช่น งานสำรวจทางธรณีวิทยา ซึ่งจุดสำรวจข้อมูลจำเป็นต้องครอบคลุมอาณาบริเวณที่กว้างขวาง และหากเป็นงานที่ต้องติดตามความเปลี่ยนแปลงของตัวแปร จำเป็นต้องใช้แรงงานของผู้วิจัยไปในการสำรวจซึ่งเป็นวิธีการที่ไม่สะดวกเป็นอย่างยิ่งในงานสำรวจ หรืองานวิจัยบางอย่างต้องการความรวดเร็วในการอ่านค่า และบันทึกข้อมูลทำให้มีการนำไมโครคอมพิวเตอร์มาใช้ในการควบคุมระบบการทดลองและจัดเก็บข้อมูล

ในบทนี้ จะนำเสนอเนื้อหาเกี่ยวกับระบบไมโครโปรเซสเซอร์ กับการควบคุมการทดลองการสื่อสารข้อมูลระหว่างระบบที่ใช้ไมโครโปรเซสเซอร์ เป็นตัวควบคุมและข้อมูลที่จำเป็นต้องใช้ในการออกแบบระบบเก็บข้อมูล โดยรายละเอียดในการออกแบบทั้งทางฮาร์ดแวร์และซอร์ฟแวร์ของระบบจะนำเสนอในบทต่อไป

โครงสร้างของระบบเก็บข้อมูลระยะไกล

1. โครงสร้างของระบบเก็บข้อมูลระยะไกล

จากจุดประสงค์หนึ่งของงานวิทยานิพนธ์ชิ้นนี้ ก็เพื่อที่จะพัฒนาระบบย่อยที่สามารถทำงานได้โดยอิสระ เพื่อที่นักวิจัยท่านอื่นสามารถนำไปประยุกต์ใช้ได้โดยง่าย ดังนั้น ผู้วิจัยจึงได้แบ่งงานเป็นส่วนย่อย ๆ โดยแต่ละส่วนอิสระต่อกันและ

เชื่อมโยงต่อกันได้โดยผ่านทางรหัสกำหนดของโปรแกรมซอร์ฟแวร์ โดยระบบย่อยเป็นดังนี้

1.1 ส่วนขยายหน่วยความจำ เป็นส่วนที่ทำหน้าที่ในการเก็บข้อมูลจากระบบควบคุมหลัก ตามรูปแบบที่กำหนด ในการสั่งงานไว้ในหน่วยความจำเก็บข้อมูลประเภท RAM ภายใต้การจัดการของไมโครคอนโทรลเลอร์ ซึ่งทำหน้าที่เป็นตัวบริหารหน่วยความจำส่วนขยาย(MEMORY MANAGEMENT CONTROLLER)โดยเฉพาะการส่งข้อมูลเพื่อจัดเก็บจะกระทำโดยผ่านทางพอร์ทเชื่อมโยงแบบขนานและการถ่ายข้อมูลออกจะกระทำโดยผ่านทางพอร์ทเชื่อมโยงแบบอนุกรม

1.2 ระบบแสดงผล ซึ่งจะรับข้อมูลในรูปแบบรหัสแอสกีจากระบบควบคุมหลักเพื่อแสดงผลในรูปแบบของ PULL-UP DOWN LEFT RIGHT MENU ทำให้สามารถสั่งงานได้ง่าย การติดต่อส่งข้อมูลจากระบบควบคุมหลักจะกระทำโดยผ่านทางพอร์ทเชื่อมโยงแบบขนาน

1.3 ระบบควบคุมหลักทำหน้าที่ในการจัดส่งข้อมูล เพื่อแสดงผลในกรณีที่ต้องการให้ผู้ใช้เลือกค่าการทำงาน จัดเก็บข้อมูลที่ได้จากระบบเครือข่ายย่อยตามระยะเวลาและรูปแบบที่ผู้ใช้กำหนดจัดการรับค่าคีย์บอร์ดสั่งงาน โปรแกรมการทำงานของระบบเครือข่ายย่อย และบริการการติดต่อกับไมโครคอมพิวเตอร์ สำหรับระบบแสดงผลและการจัดการคีย์บอร์ดรวมเป็นส่วนการติดต่อกับผู้ใช้งาน

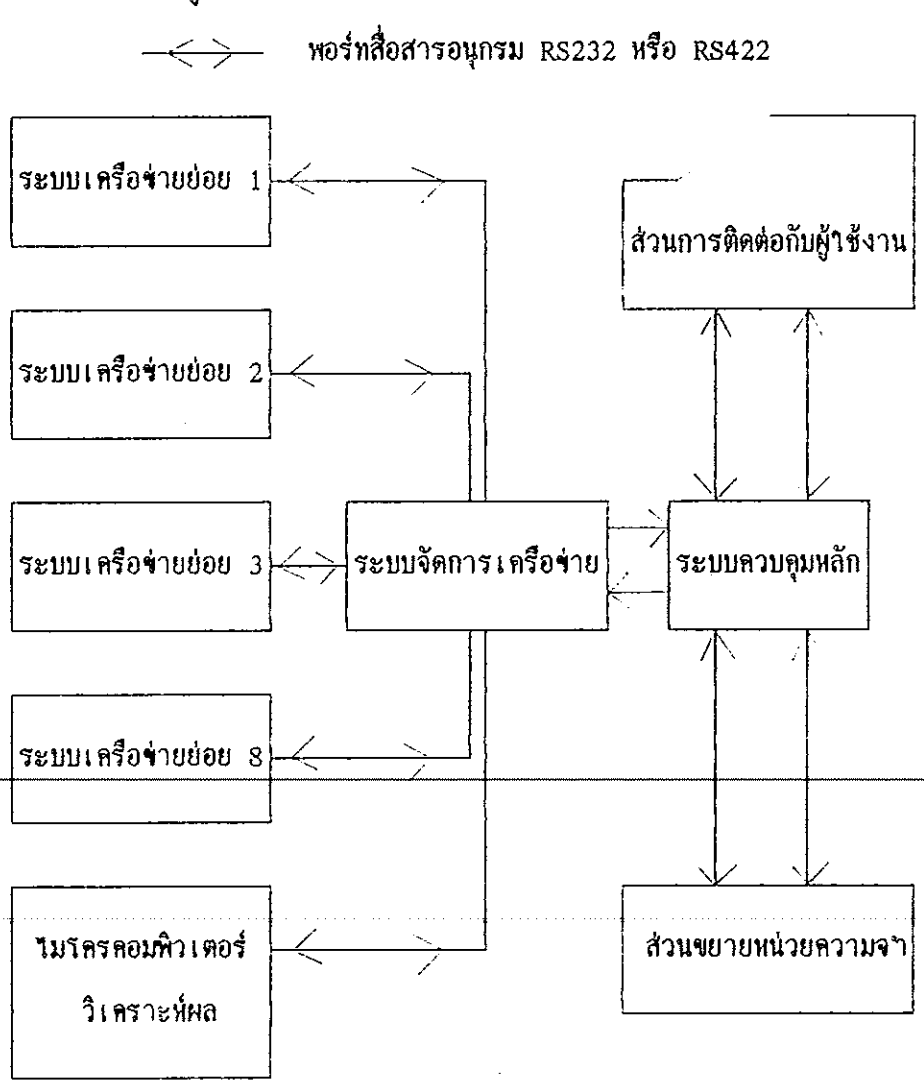
1.4 ระบบเครือข่ายย่อย เป็นส่วนของการจัดการแปลงค่าข้อมูลจากการทดลองจัดเก็บ หรือส่งเข้าสู่ระบบควบคุมหลัก ขึ้นอยู่กับโปรแกรมสั่งงานจากระบบควบคุมหลัก

1.5 ระบบจัดการเครือข่าย เนื่องจากการเชื่อมโยงของระบบจะอยู่ในรูปแบบของระบบเครือข่ายย่อยมากกว่า 1 ตัว และระบบควบคุมหลัก ดังนั้นจึงจำเป็นต้องมีระบบจัดการทางฮาร์ดแวร์ เพื่อตอบสนองต่อการติดต่อระหว่างระบบควบคุมหลักกับระบบเครือข่ายย่อยนั้น ๆ (ใช้พอร์ทอนุกรมมาตรฐาน RS-232 และ RS-422) สำหรับระบบเครือข่ายที่เลือกใช้จะเป็นแบบ STAR ซึ่งรายละเอียดของรูปแบบการเชื่อมต่อระบบไมโครโปรเซสเซอร์จะได้กล่าวถึงต่อไป

1.6 ไมโครคอมพิวเตอร์ โดยปกติแล้วระบบเครือข่าย จะใช้ไมโคร-

โปรเซสเซอร์เป็นระบบควบคุมหลัก เนื่องจากสามารถพัฒนาโปรแกรมสั่งงานได้ง่ายและมีความเร็วในการทำงานสูงเมื่อเปรียบเทียบกับไมโครโปรเซสเซอร์ที่ใช้เป็นระบบควบคุมหลักในกรณี แต่เนื่องจากงานการเก็บข้อมูลระยะไกลนั้น บางครั้งไม่สะดวกในการเคลื่อนย้ายไมโครคอมพิวเตอร์ ดังนั้นในระบบนี้ไมโครคอมพิวเตอร์จะทำหน้าที่เป็น NODE ให้กับระบบเครือข่ายโดยที่ระบบควบคุมหลักสามารถรับคำสั่งงานเงื่อนไขการทำงานจากไมโครคอมพิวเตอร์ และไมโครคอมพิวเตอร์จะทำหน้าที่ในการอ่านข้อมูลเพื่อใช้ในการประมวลผล

รูปที่ 2.1 แสดงการเชื่อมโยงระหว่างระบบที่เป็นองค์ประกอบ ของระบบเก็บข้อมูลระยะไกล



รูปที่ 2.1 โครงสร้างสมบูรณ์ของระบบเก็บข้อมูลระยะไกล

2. สายงานการควบคุม (CONTROL FLOW)

เนื่องจากระบบที่พัฒนาขึ้น มีความเป็นอิสระของไมโครโปรเซสเซอร์ที่เป็นตัวควบคุม ดังนั้นสายงานการควบคุมระบบจะสามารถแบ่งย่อยเป็นส่วน ๆ ดังนี้

2.1 ส่วนขยายหน่วยความจำหลังจากที่ระบบเริ่มทำงานไมโครโปรเซสเซอร์จะทำการตรวจสอบความผิดพลาดของหน่วยความจำ สำหรับเก็บข้อมูลจากระบบเครือข่ายย่อยแต่ละตัวและรายงานผลให้กับระบบควบคุมหลัก

2.2 ระบบแสดงผล จะอยู่ในสภาวะพร้อมที่จะรับข้อมูลจากพอร์ทขยาย โดยการอินเตอร์รัพท์สภาวะความถูกต้องการของการเริ่มทำงานจะแสดงข้อมูลของระบบแสดงผล คือหมายเลขเวอร์ชั่นของระบบแสดงผล

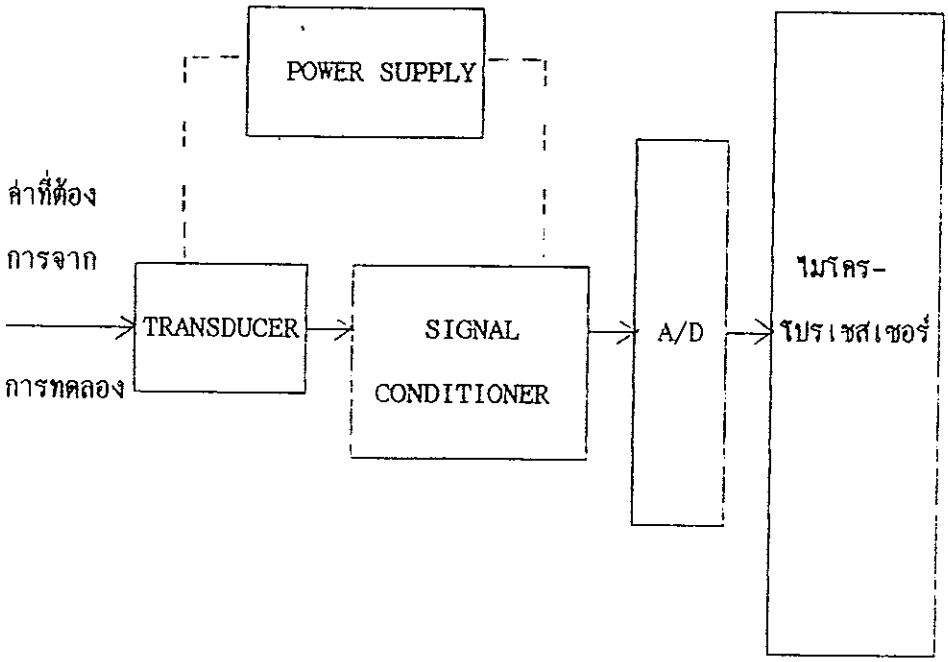
2.3 ระบบควบคุมหลัก เริ่มการทำงานจากการส่งข้อความให้กับระบบแสดงผลและจะอ่านค่าจำนวนหน่วยความจำที่ต่อกับระบบ เพื่อแสดงผลตรวจสอบความผิดพลาดของหน่วยความจำที่ติดตั้งในระบบควบคุมหลักเอง ตรวจสอบและรายงานผลการตอบสนองของระบบเครือข่ายย่อยต่อรหัสตรวจสอบที่ส่งไป และตรวจสอบการทำงานของระบบนาฬิกาเวลาจริง (REAL-TIME CLOCK) หลังจากนั้นจะเข้าสู่โปรแกรมบริการผู้ใช้งานเพื่อรับรูปแบบการทำงาน โดยพร้อมที่จะนำตัวแปรต่าง ๆ มาจากไมโครคอมพิวเตอร์โดยตรง หรือจากการป้อนค่าผ่านส่วนรับข้อมูลและเข้าสู่รูปแบบการทำงาน การหยุดการทำงานของระบบควบคุมหลัก จะเกิดขึ้นเมื่อเงื่อนไขเป็นไปตามที่กำหนด เช่น ครบกำหนดตามเวลาที่ต้องการข้อมูลเต็มหน่วยความจำ หรือเกิดความผิดพลาดในการสื่อสารข้อมูลระหว่างระบบควบคุมหลักและระบบเครือข่ายย่อย

2.4 ระบบจัดการเครือข่าย เป็นส่วนของระบบที่ไม่มีไมโครโปรเซสเซอร์เป็นตัวจัดการแต่การสั่งงาน เพื่อเลือกระบบเครือข่ายย่อยที่ต้องการติดต่อด้วยจะอยู่ภายใต้การควบคุมของไมโครโปรเซสเซอร์ ของระบบควบคุมหลัก

2.5 ไมโครคอมพิวเตอร์ หากอยู่ในรูปแบบที่ผู้ใช้ต้องการส่งตัวแปรควบคุมการทำงานจากไมโครคอมพิวเตอร์ ก็สามารถเข้าสู่โปรแกรมระบบงานเพื่อกำหนดค่าตัวแปรและส่งเข้าสู่ระบบเก็บข้อมูล และการอ่านค่าข้อมูล จะถูกควบคุมโดยผู้ใช้งาน

ระบบไมโครโปรเซสเซอร์กับการทดลอง

เพื่อให้เกิดความเข้าใจในการประยุกต์ใช้งานไมโครโปรเซสเซอร์ กับงานการทดลองก่อนที่จะนำไปสู่ระบบเก็บข้อมูลระยะไกล จะกล่าวถึงระบบไมโครโปรเซสเซอร์กับการทดลองก่อน โดยทั่วไปแล้วสิ่งที่ได้จากการทดลองจะอยู่ในรูปแบบที่ไม่สามารถส่งเข้าสู่ระบบไมโครโปรเซสเซอร์ได้โดยตรง เนื่องจากปริมาณดังกล่าวจะอยู่ในรูปแบบสัญญาณแบบอนาลอก (ANALOG SIGNAL) ดังนั้น การนำไมโครโปรเซสเซอร์มาใช้ในงานการทดลอง



รูปที่ 2.2 ระบบไมโครโปรเซสเซอร์กับการทดลอง

ดังนั้น การนำไมโครโปรเซสเซอร์มาใช้ในงานการทดลอง จำเป็นต้องศึกษากระบวนการในการเปลี่ยนรูปแบบของสัญญาณ ที่ได้จากการทดลองให้อยู่ในรูปแบบที่ไมโครโปรเซสเซอร์สามารถประมวลผลได้ สำหรับขั้นตอนของการอ่านค่าปริมาณที่ได้จากการทดลอง จนกระทั่งเข้าสู่ระบบไมโครโปรเซสเซอร์ สามารถประมวลผลได้ สำหรับขั้นตอนของการอ่านค่าปริมาณที่ได้จากการทดลองจนกระทั่ง

เข้าสู่ระบบไมโครโปรเซสเซอร์ สามารถสรุปได้ตามรูปที่ 2.2 และมีรายละเอียดของแต่ละส่วนดังนี้

1. TRANSDUCER

สิ่งที่ต้องการอ่านค่า และประมวลผลจากการทดลองจะถูกแปลงให้เป็นสัญญาณทางไฟฟ้าโดยใช้ TRANSDUCER ที่เหมาะสมซึ่งโดยทั่วไปแล้วสิ่งที่ได้จากการทดลอง และค่าที่ต้องการอ่านและประมวลผล พอที่จะสรุปได้ตามตาราง 2.1 ผู้วิจัยทราบว่าสิ่งที่ต้องการอ่านค่า และประมวลผลในการทดลองนั้น ๆ ก็สามารถเลือกใช้ TRANSDUCER ที่เหมาะสมกับการวัด โดยอาจใช้วิธีทางตรงหรือทางอ้อม เช่น ในการทดลองใด ๆ ต้องการอ่านค่าความสูงของคลื่นเสียงที่สะท้อนจากการระเบิด เพื่อหาโครงสร้างการวางตัวของชั้นหิน ก็สามารถเลือกใช้ไมโครโฟนซึ่งสามารถเปลี่ยนความสูงของคลื่นเสียงให้เป็นปริมาณทางไฟฟ้า เพื่อป้อนเข้าสู่ระบบจัดการสัญญาณต่อไป

ตาราง 2.1 ปริมาณพื้นฐานและสิ่งที่สามารถวัดค่าได้

ปริมาณพื้นฐาน	ตัวอย่างค่าที่วัด
การขจัดเชิงเส้น(LINEAR DISPLACEMENT)	ความยาว ความกว้าง ความหนา ความเร็ว
การขจัดเชิงมุม(ANGULAR DISPLACEMENT)	มุมของการไหล มุมของการแกว่ง
ความเร็วเชิงเส้น(LINEAR VELOCITY)	ความเร็ว อัตราการไหล
ความเร็วเชิงมุม(ANGULAR VELOCITY)	ความเร็วเชิงมุม อัตราการหมุน

ตาราง 2.1 (ต่อ)

ความเร่งเชิงเส้น(LINEAR ACCLERATOR)	การสั่น การเคลื่อนที่
ความเร่งเชิงมุม(ANGULAR DISPLACEMENT)	ทอร์ก โมเมนต์เฉื่อย
แรง (FORCE)	น้ำหนัก ความหนาแน่น
เวลา (TIME)	ความถี่
การแผ่ของคลื่นแม่เหล็กไฟฟ้า (ELECTROMAGNETIC RADIATION)	ความคลื่น กำลัง ความเข้มของสนามแม่เหล็ก

2. ระบบจัดการสัญญาณ (SIGNAL CONDITIONER)

หลังจากที่ได้สัญญาณทางไฟฟ้าจาก TRANSDUCER ซึ่งจะเป็นสัดส่วนกับปริมาณที่ต้องการวัดค่าจากการทดลองแล้ว ในรูปแบบอนาลอก (ANALOG SIGNAL) อันเป็นสัญญาณที่มีความเปลี่ยนแปลงต่อเนื่องเป็นฟังก์ชั่นของเวลา ไม่สามารถป้อนเข้าสู่ตัวแปลงสัญญาณจากอนาลอกเป็นดิจิตอล (ANALOG TO DITAL CONVERTER : ADC) ได้โดยตรง จำเป็นต้องมีการปรับแต่งสัญญาณอนาลอกให้อยู่ในรูปแบบ หรือเงื่อนไขที่เหมาะสมกับตัวแปลงสัญญาณ เช่น การกำจัดสัญญาณรบกวนที่ไม่พึงประสงค์ซึ่งอาจรบกวนต่อค่าที่ได้จากการทดลอง การปรับระดับ ขยาย หรือลดทอนสัญญาณให้มีความแรงของสัญญาณพอเหมาะ นอกจากนี้รูปแบบของสัญญาณที่ได้จาก TRANSDUCER ในบางครั้งอาจไม่เป็นเชิงเส้นกับปริมาณที่ต้องการวัด ดังนั้นจำเป็นต้องมีการชดเชยความผิดพลาดในส่วนนี้ก่อน

โดยทั่วไปแล้วการจัดการของระบบจัดการสัญญาณ จะเกี่ยวข้องกับวงจร

อนาลอกประเภทออปเปอร์เรชันนอลแอมปลิไฟเออร์(OPERATIONAL AMPLIFILER : OP-AMP) ในรูปแบบต่าง ๆ เช่น วงจรเปรียบเทียบแรงดัน (COMPARATOR) วงจรขยายแบบไม่กลับขั้ว (NON-INVERTING AMPLIFIER)วงจรขยายแบบกลับขั้ว (INVERTING AMPLIFIER) วงจรพรีแอมปลิไฟร์ (PRE-AMPLIFIER) ซึ่งรูปแบบของวงจรส่วนนี้สามารถหาได้จากเอกสารทั่วไป

3. การแปลงรูปแบบสัญญาณจากอนาลอกเป็นดิจิตอล (ANALOG TO DIGITAL CONVERTER)

เพื่อให้ไมโครโปรเซสเซอร์ สามารถประมวลผลสัญญาณที่ได้จากการทดลอง ส่วนจัดการส่วนนี้เป็นสิ่งที่ขาดไม่ได้ โดยหลักจากที่สัญญาณที่ได้จากการทดลองถูกปรับให้มีความเหมาะสมแล้วก็พร้อมสำหรับการเปลี่ยนรูปแบบ ให้อยู่ในรูปแบบดิจิตอล คือ มีการแทนค่าสัญญาณด้วยเลขฐาน 2(0 และ 1) ที่สอดคล้องกับน้ำหนักของค่าสัญญาณ

หน้าที่ของส่วนจัดการแปลงรูปแบบสัญญาณอนาลอกเป็นดิจิตอล คือจะสร้างรูปแบบของเลขฐานสองเพื่อแทนค่าน้ำหนัก ของสัญญาณอนาลอกซึ่งอาจอยู่ในรูปแบบของแรงเคลื่อนไฟฟ้าหรือกระแสไฟฟ้า (ขึ้นอยู่กับการจัดการของระบบจัดการสัญญาณ) โดยสิ่งที่จำเป็นต้องทราบเกี่ยวกับหลักการแปลงสัญญาณอนาลอกเป็นดิจิตอล คือ

3.1 ความละเอียดของการแปลงค่า (RESOLUTION) ขึ้นอยู่กับจำนวนบิทเอาท์พุท (OUTPUT BIT) ของวงจรส่วนแปลงสัญญาณอนาลอกเป็นดิจิตอล เช่น ถ้าเลือกใช้วงจรส่วนแปลงค่าสัญญาณอนาลอกเป็นดิจิตอลที่มีจำนวนบิทเอาท์พุท เป็น 8 บิท ดังนั้น จะมีค่าความละเอียดในการแปลงค่าเป็น 1 ใน 2^8 หรือ 1/256 และหากเลือกใช้จำนวนบิทเอาท์พุทเป็น 12 จะมีค่าความละเอียดในการแปลงค่าเป็น 1 ใน 2^{12} หรือ 1/4096 ซึ่งมีค่าความละเอียดในการแปลงค่ามากขึ้น

3.2 ความแม่นยำของการแปลงค่า (ACCURACY) เป็นค่าเปรียบเทียบระหว่างค่าจริง ของการแทนค่าสัญญาณที่ได้จากเอาท์พุทของวงจรแปลงค่าสัญญาณ (TRUE VALUE) กับค่าที่คาดหวัง (EXPECTED VALUE) จะแสดงออกมาใน

รูปของค่าเปอร์เซ็นต์ความผิดพลาด

3.3 ความเป็นเชิงเส้นของการทำงาน (LINEARITY) เป็นค่าเปรียบเทียบระหว่างค่าเชิงเส้นของสัญญาณอนาลอกจริง กับผลที่ได้จากการแทนค่าจากค่าเริ่มต้นจนถึงค่าสูงสุดที่จะเป็นได้สัญญาณอินพุท

3.4 ความเร็วในการแปลงค่าสัญญาณ (CONVERSION TIME) เป็นตัวบ่งบอกความเร็วในการแปลงค่าสัญญาณอนาลอกเป็นรูปแบบข้อมูลดิจิตอล ซึ่งหากปัจจัยกำหนดนี้มีค่าน้อยจะบ่งบอกความเร็วในการแปลงข้อมูลได้เร็วกว่าค่ามาก

ข้อมูลที่ได้จากวงจรส่วนแปลงค่าสัญญาณอนาลอกเป็นดิจิตอล สามารถป้อนให้กับระบบไมโครโปรเซสเซอร์ได้โดยตรง

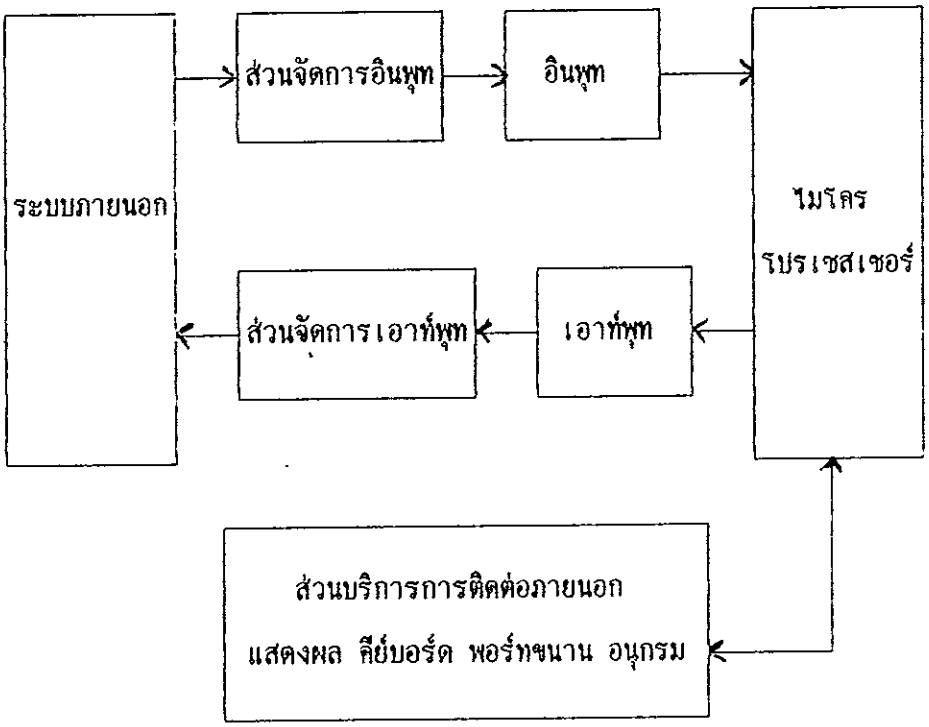
4. ระบบไมโครโปรเซสเซอร์ (MICROPROCESSOR SYSTEM)

ในงานการเก็บบันทึกข้อมูลที่ได้จากการทดลอง จำเป็นที่จะต้องติดตามความเป็นไปของตัวแปรในการทดลอง และบันทึกค่าเหล่านั้นไว้เพื่อใช้ในการวิเคราะห์ผลภายในระยะเวลาที่กำหนด จำเป็นต้องจัดรูปแบบให้ระบบไมโครโปรเซสเซอร์เก็บบันทึกข้อมูลจากการทดลองตามช่วงเวลา (TIME STEP) ที่กำหนด รูปแบบการทำงานของไมโครโปรเซสเซอร์ในกรณีนี้ เรียกว่าเป็นระบบเวลาจริง (REAL-TIME SYSTEM)

ระบบเวลาจริง หมายถึง ระบบซึ่งตอบสนองหรือมีปฏิกิริยาโต้ตอบกับสิ่งกระตุ้นโดยทันที แม้ว่าในขณะนั้นระบบจะมีงานอื่นที่ต้องกระทำก็ตาม สมรรถนะในการทำงานของระบบเวลาจริงเป็นสิ่งที่มีความสำคัญมาก เช่น ในกรณีที่จัดรูปแบบการทำงานของไมโครโปรเซสเซอร์เป็นระบบเวลาจริง เพื่อรับข้อมูลจากกระบวนการทดลองภายใต้ข้อกำหนดคือ ให้ใช้งานวงจรแปลงค่าสัญญาณอนาลอก เป็นดิจิตอลเต็มรูปแบบซึ่งหมายถึง เมื่อวงจรแปลงค่าทำงานเสร็จแล้วจะต้องกระตุ้นไมโครโปรเซสเซอร์รับข้อมูลและวงจรแปลงค่าพร้อมที่จะรับค่าใหม่ (สถานการณ์นี้มีความสำคัญมาก ในกรณีการทดลองที่ตัวแปรที่ต้องการวัดเกิดขึ้นเพียงช่วงสั้น ๆ) ดังนั้นไมโครโปรเซสเซอร์ต้องอยู่ในสภาวะที่พร้อมจะรับค่าข้อมูลใหม่เสมอ อาจมีข้อโต้แย้งว่าการที่ข้อมูลสูญหายไปเพียง 1 ชุด ในทางการวิเคราะห์ข้อมูลแล้วไม่มีความ

แตกต่างแต่ถ้าพิจารณาในกรณีที่ต้องการเปรียบเทียบข้อมูลระหว่างจุดใด ๆ ที่มีระบบไมโครโปรเซสเซอร์เป็นอิสระต่อกันแล้ว การเปรียบเทียบจะไม่สามารถทำได้ เนื่องจากความแตกต่างของเหตุการณ์ (ASYNCRONOUS EVENT)

องค์ประกอบของระบบไมโครโปรเซสเซอร์เวลาจริงแสดงดังรูปที่ 2.3



รูปที่ 2.3 ระบบไมโครโปรเซสเซอร์ในรูปแบบของระบบเวลาจริง

โดยทั่วไปไมโครโปรเซสเซอร์จะได้รับสัญญาณจากระบบภายนอกโดยผ่านการแปลงจาก TRANSDUCER ระบบจัดการสัญญาณและระบบแปลงรูปแบบสัญญาณสำหรับการส่งสัญญาณเข้าสู่ระบบภายนอก จะกระทำโดยโปรแกรมสั่งงานตามขั้นตอนผ่านระบบแปลงสัญญาณและเข้าสู่ระบบจัดการทางด้านเอาท์พุทเพื่อสั่งงานต่อไป

การสื่อสารข้อมูลระหว่างระบบไมโครโปรเซสเซอร์

ปัจจุบันเครื่องมือวัดที่ใช้ในงานวิจัยส่วนใหญ่ จะใช้ไมโครโปรเซสเซอร์

เป็นตัวจัดการ ทำให้นักวิจัยสามารถเชื่อมโยงระบบเครื่องมือวัดต่าง ๆ เข้ากับคอมพิวเตอร์เพื่อเก็บบันทึกข้อมูลสั่งงานตามรูปแบบที่ต้องการ และอ่านข้อมูลเพื่อวิเคราะห์ผลการทดลองได้ทันที โดยทั่วไปแล้วการสื่อสารข้อมูลระหว่างระบบไมโครโปรเซสเซอร์หรือไมโครคอมพิวเตอร์สามารถกระทำได้ในสองรูปแบบคือ แบบขนาน (PARALLEL DATA COMMUNICATION)โดยใช้สายสัญญาณหลายเส้นและแบบอนุกรม (SERIAL DATA COMMUNICATION) โดยใช้สายสัญญาณที่น้อยกว่า ซึ่งขึ้นอยู่กับมาตรฐานที่เลือกใช้(RS232 หรือ RS422 เป็นต้น)

ในรูปแบบการสื่อสารข้อมูลแบบอนุกรม ข้อมูลที่ต้องการส่งหรือรับจะผ่านสายสัญญาณครั้งละ 1 บิท ดังนั้นในรูปแบบของรหัสข้อมูลแบบแอสกีซึ่งมีความยาวของรหัส 8 บิท จำเป็นต้องรอถึง 8 ครั้งก่อนที่จะสามารถถอดรหัสได้

สำหรับรูปแบบการสื่อสารข้อมูลแบบขนาน แต่ละบิทของรหัส จะมีสายสัญญาณเป็นของตัวเอง ดังนั้น ในการส่งข้อมูลเพียงครั้งเดียวก็สามารถถอดรหัสได้ ดังนั้น ถ้าพิจารณาในแง่ของความเร็วในการสื่อสารข้อมูล รูปแบบการสื่อสารข้อมูลแบบขนานให้ความเร็วในการรับหรือส่งข้อมูลเร็วกว่า แต่อย่างไรก็ตามรูปแบบการสื่อสารข้อมูลแบบขนาน มีใช้เฉพาะการสื่อสารข้อมูลในระบบไมโครโปรเซสเซอร์ (รวมทั้งไมโครคอมพิวเตอร์) เช่น การติดต่อระหว่างไมโครโปรเซสเซอร์ ซึ่งเป็นหน่วยประมวลผลกลางหรือซีพียู(CPU) กับหน่วยความจำ และอุปกรณ์เชื่อมโยงของระบบ เนื่องจากสายส่งจำนวนมาก ทำให้ราคาของการติดตั้งระบบเพิ่มขึ้นต่อระยะทางระหว่างระบบ นอกจากนี้คุณสมบัติทางไฟฟ้าของสายส่งสัญญาณ ซึ่งจะทำให้ระดับของสัญญาณต่ำลงเมื่อระยะทางเพิ่มขึ้นยังเป็นข้อจำกัดของการสื่อสารข้อมูลรูปแบบนี้

สำหรับระบบเครื่องมือวัดในงานวิจัยได้มีการพัฒนาระบบการสื่อสารข้อมูลแบบขนานมาตรฐาน IEEE-488 ทำให้สามารถเชื่อมโยงระบบเครื่องมือวัดหลายตัวเข้ากับไมโครคอมพิวเตอร์ที่ทำหน้าที่ในการควบคุมได้

แต่ละรูปแบบของการสื่อสารข้อมูลมีข้อดีข้อเสียสำหรับงานวิจัย พอที่จะสรุปตามตารางที่ 2.2 ซึ่งแสดงข้อเปรียบเทียบเท่าที่จำเป็นสำหรับงานวิจัย จะเห็นได้ว่าการเชื่อมต่อผ่านทางพอร์ทเชื่อมโยงแบบขนาน(โดยปกติอุปกรณ์มาตรฐานที่ติดตั้งให้กับไมโครคอมพิวเตอร์ พอร์ทขนานจะเป็นรูปแบบของ PRINTER PORT

ซึ่งเป็นมาตรฐานของ CENTRONIC ADAPTER เป็นพอร์ที่ตอบสนองต่อการทำงานในรูปแบบส่งข้อมูลอย่างเดียว)ไม่เหมาะต่อการนำมาใช้เป็นพอร์ทสำหรับงานเก็บข้อมูล

พอร์ทมาตรฐาน IEEE-488 (GPIB-488) เมื่อเปรียบเทียบกับพอร์ทอนุกรม ความเหมาะสมต่อการนำไปใช้งานจะขึ้นอยู่กับรูปแบบการนำไปใช้งาน เช่น

- ถ้าใช้พอร์ทมาตรฐาน IEEE-488อุปกรณ์เครื่องมือวัดทุกตัวที่มาต่อกับระบบจำเป็นต้องติดตั้งพอร์ท(เป็นอุปกรณ์เพิ่มเติม ซึ่งมีราคาค่อนข้างแพง และไม่สามารถดัดแปลงเพื่อติดตั้งในอุปกรณ์เครื่องมือวัดที่มีใช้อยู่แล้วได้) และจำเป็นต้องเพิ่มเติมการ์ดควบคุมมาตรฐาน IEEE-488 บนระบบไมโครคอมพิวเตอร์ นอกจากนี้สายเชื่อมต่อของระบบ IEEE-488 ยังเป็นสายที่ใช้เฉพาะงาน

ข้อเปรียบเทียบ	พอร์ทอนุกรม SERIAL:RS232C	พอร์ทขนาน RARALLEL	พอร์ท IEEE-488
ความยาวของสายสื่อสาร	ไม่เกิน 300 ฟุต	ไม่เกิน2ม.	ไม่เกิน20ม.
จำนวนเส้นของสายสัญญาณที่จำเป็น	3 เส้น	12 เส้น	12 เส้น
ทิศทางการสื่อสารจากคอมพิวเตอร์	ส่ง/รับ	ส่ง	ส่ง/รับ
ความเร็วในการส่งผ่านข้อมูล	ช้า	เร็ว	เร็ว
การตอบสนองต่อระบบหลากหลาย อุปกรณ์เชื่อมต่อ MULTI-USER SYSTEMS)	ตอบสนอง	ไม่ตอบสนอง	ตอบสนอง
อุปกรณ์มาตรฐานบนคอมพิวเตอร์	เป็น	เป็น	ไม่เป็น

ตาราง 2.2 ตารางเปรียบเทียบข้อดี และข้อเสียของมาตรฐานการสื่อสารข้อมูลที่มีใช้ในเครื่องมือวิจัย

ฝ่ายหอสมุด
คุณหญิงหลง อรรถกระวีสุนทร

- พอร์ทอนุกรมไม่จำเป็นต้องมีอุปกรณ์เพิ่มเติม เนื่องจากปกติเครื่องมือวัดในวิจัยจะตอบสนองต่อการเชื่อมโยงในรูปแบบนี้อยู่แล้ว แต่มีข้อเสียเมื่อเทียบกับระบบพอร์ท IEEE-488 ในแง่ของความเร็วในการรับ/ส่งข้อมูล

ในงานวิจัยที่จำเป็นต้องเก็บข้อมูลระยะไกล พอร์ทอนุกรมจะมีความเหมาะสมในการใช้งานมากกว่า โดยเฉพาะอย่างยิ่งหากจุดสำรวจย่อยแต่ละจุดอยู่ห่างจากระบบประมวลผล หรือระบบเก็บข้อมูลหลักมากกว่าขีดความสามารถในการสื่อสารข้อมูลตามมาตรฐาน RS-232C แล้วก็สามารถใช้พอร์ทอนุกรมมาตรฐาน RS-422 เข้ามาใช้แทนได้

จุดเด่นอีกประการที่ผู้วิจัยได้ให้ความสำคัญกับพอร์ทอนุกรมก็ คือ สามารถดัดแปลงอุปกรณ์เครื่องมือวัดราคาถูกที่มีใช้ในห้องปฏิบัติการ ให้สามารถสื่อสารข้อมูลในรูปแบบนี้ได้โดยง่าย

จากข้อดีของพอร์ทอนุกรมที่ได้กล่าวมาแล้วนั้น ทำให้ผู้วิจัยเลือกใช้การสื่อสารข้อมูลระหว่างระบบเครือข่ายย่อย และระบบเก็บข้อมูลหลักเป็นแบบอนุกรม โดยสามารถเลือกใช้มาตรฐานได้ทั้งสองรูปแบบ คือ RS-232C และ RS-422

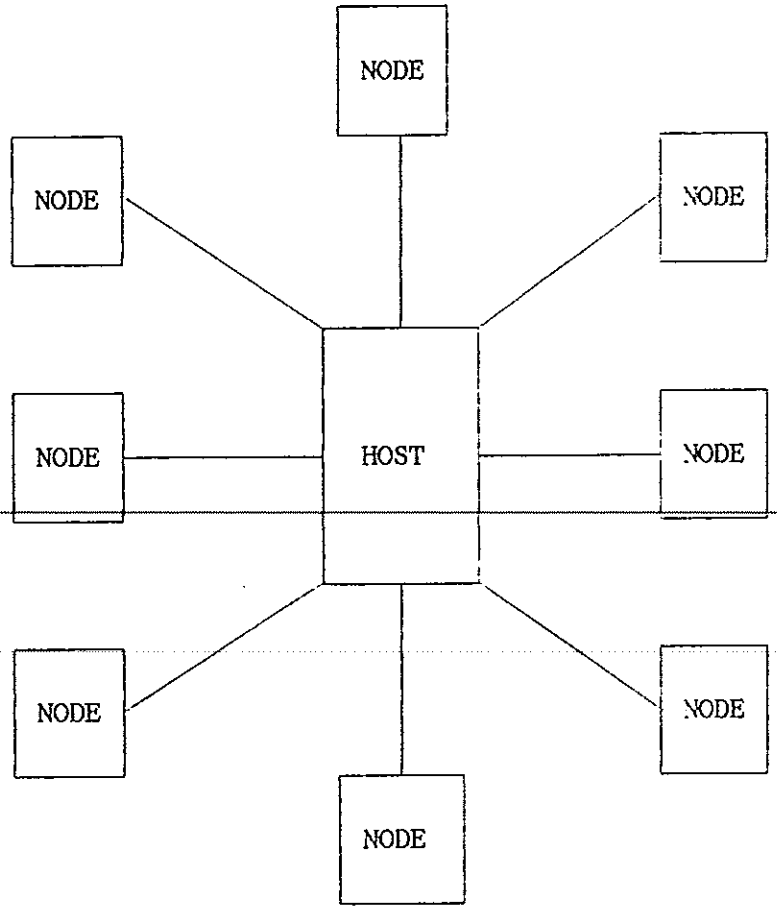
รูปแบบการสื่อสารข้อมูลแบบอนุกรมสำหรับระบบเครือข่าย

ในงานวิจัย ได้พัฒนาระบบเก็บข้อมูลระยะไกลโดยการจัดการของไมโครโปรเซสเซอร์ ทั้งในระบบเครือข่ายย่อยและระบบเก็บข้อมูลหลัก โดยภาพรวมแล้วระบบทั้งหมดนั้นจะเชื่อมโยงในรูปแบบของระบบเครือข่าย (NETWORK) โดยระบบเครือข่ายย่อยและระบบเก็บข้อมูลหลักจะถูกเรียกว่า NODE และ HOST ตามลำดับ ซึ่งรูปแบบของระบบเครือข่าย(NETWORK) เท่าที่มีใช้กันอยู่โดยทั่วไปจะมี 3 รูปแบบ คือ

1. **ระบบเครือข่ายแบบ** STAR

เป็นระบบเครือข่าย ซึ่งการเชื่อมโยงระหว่างระบบควบคุมหลัก(HOST) และระบบเครือข่ายย่อย (NODE) จะเป็นแบบ 1:1 คือจะมีสายเชื่อมโยงระหว่างกันเป็นอิสระตามรูปที่ 2.4

รูปแบบระบบเครือข่ายแบบนี้จะมีใช้ในงานประเภท TIME-SHARING คือ เป็นงานที่ระบบควบคุมหลัก (HOST) สามารถตอบสนองต่อการเรียกของระบบเครือข่ายย่อย (NODE) เพื่อเก็บ/ดึงข้อมูลได้พร้อมกัน โดยอาศัยการแบ่งเวลาเป็นช่วงเพื่อตอบสนองต่อระบบเครือข่ายย่อยใด ๆ โดยไม่จำเป็นต้องตอบสนองระบบเครือข่ายย่อยให้เสร็จเป็นตัว ๆ การที่ระบบเครือข่ายแบบนี้สามารถตอบสนองต่อการเรียกและสามารถส่งข้อมูลได้พร้อม ๆ กัน เนื่องจากลักษณะของการต่อสายเชื่อมโยงระหว่างระบบควบคุมหลัก และระบบเครือข่ายย่อยเป็นแบบเจาะจง ทำให้ระบบเครือข่ายย่อยมองเห็นเฉพาะข้อมูลในสายส่งเป็นของตัวเองเท่านั้น ไม่จำเป็นต้องมีการกำหนดหมายเลขประจำตัวของระบบเครือข่ายย่อย ที่สำคัญคือระบบควบคุมหลักจำเป็นต้องมีความสามารถในการแยกแยะชุดของข้อมูลที่จะส่งไปยังระบบเครือข่ายย่อย และต้องมีความเร็วในการทำงานมากพอที่จะบริการเรียกจากระบบเครือข่ายย่อยได้ทัน



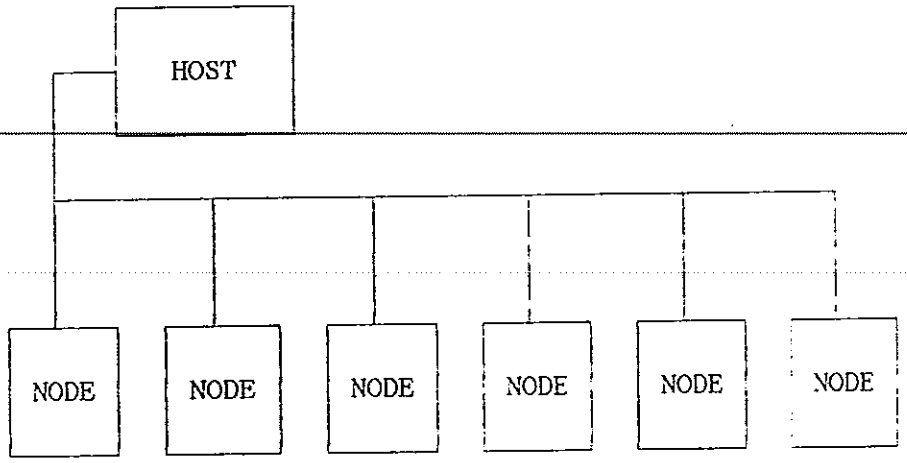
รูปที่ 2.4 รูปแบบของระบบเครือข่ายแบบ STAR

2. **ระบบเครือข่ายแบบ LOOP**

การเชื่อมโยงของระบบเครือข่ายแบบนี้ จะช่วยลดความยาวของสายสัญญาณที่เป็นสื่อกลางในการรับและส่งข้อมูลจากระบบควบคุมหลักและระบบเครือข่ายย่อย เนื่องจากการเชื่อมต่อระบบเครือข่ายในรูปแบบนี้ ใช้สายสัญญาณจากระบบควบคุมหลักไปยังระบบเครือข่ายย่อยเพียงเส้นเดียว ระบบเครือข่ายทุกตัวสามารถดึงข้อมูลที่อยู่ในระบบสายส่งได้ ดังนั้น โดยรูปแบบการสื่อสารข้อมูลแล้ว ในกรณีของเครือข่ายแบบ LOOP ระบบเครือข่ายย่อยใด ๆ สามารถกระทำตัวเป็นระบบควบคุมได้ ขึ้นอยู่กับการเลือกของผู้ใช้งาน ลักษณะทางฮาร์ดแวร์โดยทั่วไปแล้ว ระบบเครือข่ายย่อยจะถูกเจาะจงให้รับข้อมูลโดยจำเป็นต้องถอดรหัสหมายเลขแทนตัวเองก่อน และในกรณีที่มีความจำเป็นต้องส่งข้อมูลก็อาจจำเป็นต้องเข้ารหัสหมายเลขของระบบเครือข่ายย่อยตัวนั้น ๆ ในส่วนหัวของเฟรมข้อมูลที่จะส่งทุกครั้ง ดังนั้น ข้อเสียพอที่จะเห็นได้จากกรณีนี้ คือ การทำงานของระบบเครือข่ายในรูปแบบนี้ค่อนข้างช้ากว่าระบบเครือข่ายแบบ STAR

สำหรับระบบเครือข่ายแบบนี้มีใช้ในงานการเก็บและแสดงข้อมูล โดยที่ระบบควบคุมหลักจะทำหน้าที่ในการอ่านค่าจากระบบเครือข่ายย่อย หรือส่งค่าให้กับระบบเครือข่ายย่อยเป็นช่วง ๆ และสิ่งที่แสดงหรือจัดเก็บจะเป็นค่าปัจจุบันของข้อมูล เช่น ระบบการแสดงผลราคาซื้อขายหุ้นของตลาดหลักทรัพย์

ระบบเครือข่ายจะมีรูปแบบดังรูปที่ 2.5

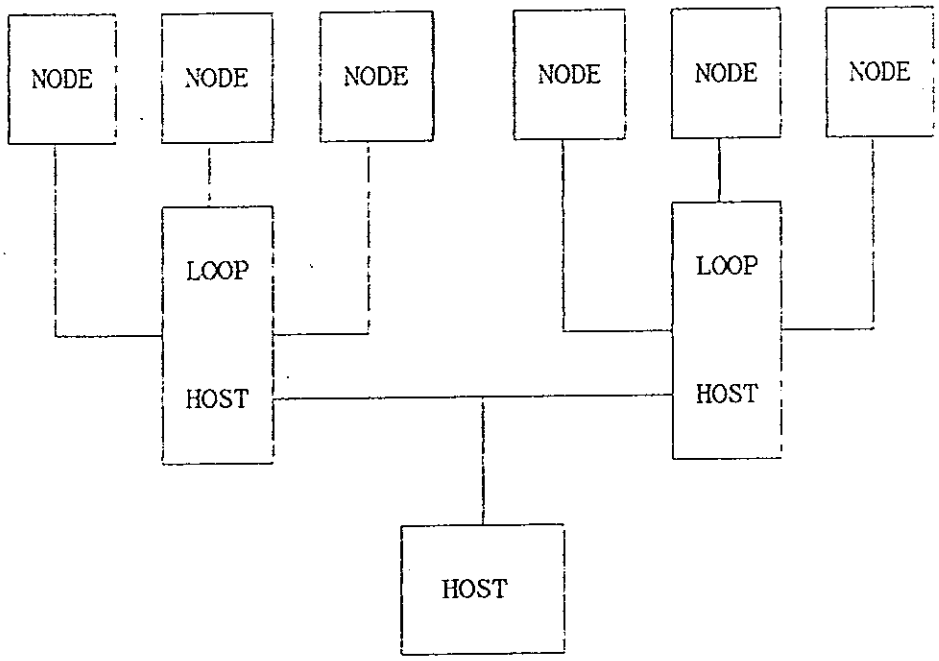


รูปที่ 2.5 รูปแบบของระบบเครือข่ายแบบ LOOP

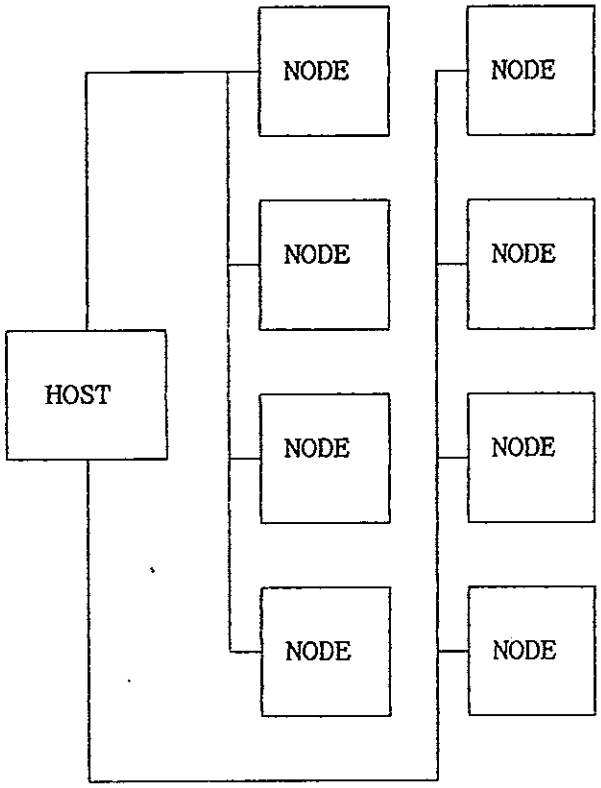
3. ระบบเครือข่ายแบบผสม LOOP-STAR หรือ STAR-LOOP

เป็นระบบเครือข่ายแบบลูกผสมโดยการรวมข้อดี และข้อเสียของระบบเครือข่ายแบบ LOOP หรือ STAR โดยหลีกเลี่ยงในบางจุดของแต่ละระบบ

รูปแบบของระบบเครือข่ายแบบผสมจะเป็นดังนี้



รูปที่ 2.6 รูปแบบของระบบเครือข่ายแบบ LOOP-STAR



รูปที่ 2.7 รูปแบบของระบบเครือข่ายแบบ STAR-LOOP

ข้อแนะนำในการเลือกใช้รูปแบบของระบบเครือข่ายในการสื่อสารข้อมูล

การเลือกรูปแบบของการต่อเครือข่ายการสื่อสารข้อมูลขึ้นอยู่กับความต้องการของผู้ออกแบบระบบและมีหลายปัจจัยที่มีผลต่อการตัดสินใจเลือก โดยมีคำแนะนำในการเลือกรูปแบบของระบบเครือข่ายโดยย่อ ๆ ดังนี้

ตาราง 2.3 ข้อเปรียบเทียบการเลือกรูปแบบระบบเครือข่าย

ความต้องการ	รูปแบบ	เหตุผล
ต้องการความถูกต้องของการรับ/ส่งข้อมูลสูง	STAR	หากเกิดความผิดพลาดของสายรับ/ส่งข้อมูลเส้นใดเส้นหนึ่งระบบเครือข่ายย่อยได้รับผลกระทบเพียงระบบเดียว
ความเร็วในการทำงานของระบบ	STAR	แต่ละระบบเครือข่ายย่อยติดต่อกับระบบควบคุมโดยตรง ไม่มีความจำเป็นที่จะต้องกำหนดรหัสการรับ/ส่ง
ราคาการติดตั้งระบบ	LOOP	ใช้สายส่งเพียงเส้นเดียวเพื่อเชื่อมโยงระบบ
ความสามารถในการแยกแยะความผิดพลาด	STAR	ใช้ระบบสายส่งแบบแยกอิสระต่อระบบเครือข่ายย่อย

รูปแบบการสื่อสารข้อมูลแบบอนุกรมมาตรฐาน RS-232C **และ** RS-422

รูปแบบการสื่อสารข้อมูลแบบอนุกรม ที่รู้จักกันมานานนั้นเป็นมาตรฐาน RS-232 ซึ่งใช้สายสัญญาณที่จำเป็นในการสื่อสารข้อมูลเพียง 3 เส้น คือสายสัญญาณส่งข้อมูล(TRANSMITTED DATA:TSD) และสายสัญญาณรับข้อมูล(RECEIVED DATA : RXD) โดยสัญญาณจะเปรียบเทียบกับระดับอ้างอิงที่ 0(GROUND) และในบางงานที่จำเป็นต้องมีการตรวจสอบสถานะของระบบ อาจเพิ่มเติมสายสัญญาณที่จำเป็นได้ รูปแบบของสายสัญญาณตามมาตรฐานจะเป็นดังตาราง 2.4

ตาราง 2.4 สายสัญญาณมาตรฐาน RS-232 ที่จำเป็นในการเชื่อมโยงระบบ

หมายเลข	ชื่อย่อ	รายละเอียด
1		PROTECTIVE GROUND
2	TXD	TRANSMITTED DATA
3	RXD	RECEIVED DATA
4	RTS	REQUEST TO SEND
5	CTS	CLEAR TO SEND
6	DSR	DATA SET READY
7	GND	SIGNAL GROUND
20	DTR	DATA TERMINAL READY

แม้ว่ารูปแบบการสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-232 จะเป็นที่นิยมใช้ในการเชื่อมโยงระบบไมโครโปรเซสเซอร์มาเป็นเวลานาน แต่จากข้อจำกัดในด้านความเร็วในการส่งข้อมูลผ่านระยะทางไกล ๆ (RS-232 จำกัดอัตราการส่งผ่านข้อมูลสูงสุดไว้ที่ 20,000 บิทวินาที ที่ความยาวของสายส่งสัญญาณสูงสุด 50 ฟุต) ซึ่งเป็นที่ต้องการสำหรับการสื่อสารข้อมูลในปัจจุบัน ทำให้มีการพัฒนามาตรฐานในการส่งข้อมูลแบบใหม่ที่มีขีดความสามารถสูงกว่า RS-232 จึงเป็นจุดกำเนิดของมาตรฐาน RS-423 RS-422 และ RS-499

มาตรฐาน RS-423 ถูกออกแบบมาให้มีรูปแบบการใช้งานที่ใกล้เคียงกับ RS-232 คือ เป็นการสื่อสารข้อมูลระหว่างระบบโดยใช้สายสัญญาณเดี่ยว (SINGLE ENDED TRANSMISION)แต่ได้ขยายขีดความสามารถของการส่งผ่านข้อมูลเป็น 100 กิโลบิทต่อวินาที (KILOBIT PER SECOND:KBPS) ที่ความยาวของสายส่งสัญญาณ

สูงสุด 300 ฟุต และสามารถส่งผ่านข้อมูลได้ไกลกว่านั้น คือ ใช้อัตราการส่งผ่านข้อมูล 1 กิโลบิทต่อวินาทีที่ความยาวสายส่งสัญญาณไกลถึง 4000 ฟุต(1200 เมตร) เนื่องจากมาตรฐานของ RS-423 ได้แก้ไขจุดบกพร่องของ CROSS TALK ซึ่งเป็นการรบกวนของสัญญาณระหว่างคู่สายส่งอันเกิดจากการเหนี่ยวนำ และยังได้ปรับปรุงรูปแบบของสัญญาณ (WAVESHAPE) ในทางคุณสมบัติทางไฟฟ้านั้น RS-423 มีค่าอิมพิแดนซ์ (IMPEDANCE) ขณะที่ไม่มีแรงเคลื่อนไฟฟ้าสูงกว่า RS-232 ทำให้คุณลักษณะของโหลดต่ำกว่าปกติ

มาตรฐาน RS-422 ได้ใช้หลักความแตกต่างของการส่งสัญญาณ (DIFFERENTIAL TRANSMISSION) ในการขยายขีดความสามารถในการส่งผ่านสัญญาณให้ได้มากกว่า 100 กิโลบิทต่อวินาที ที่ความยาวของสายส่งข้อมูล 4000 ฟุต ซึ่งโดยหลักการแล้วที่ความยาวของสายส่งต่ำกว่าที่กำหนด สามารถเพิ่มอัตราเร็วในการส่งผ่านข้อมูลได้สูงถึง 10 เมกะบิทต่อวินาที

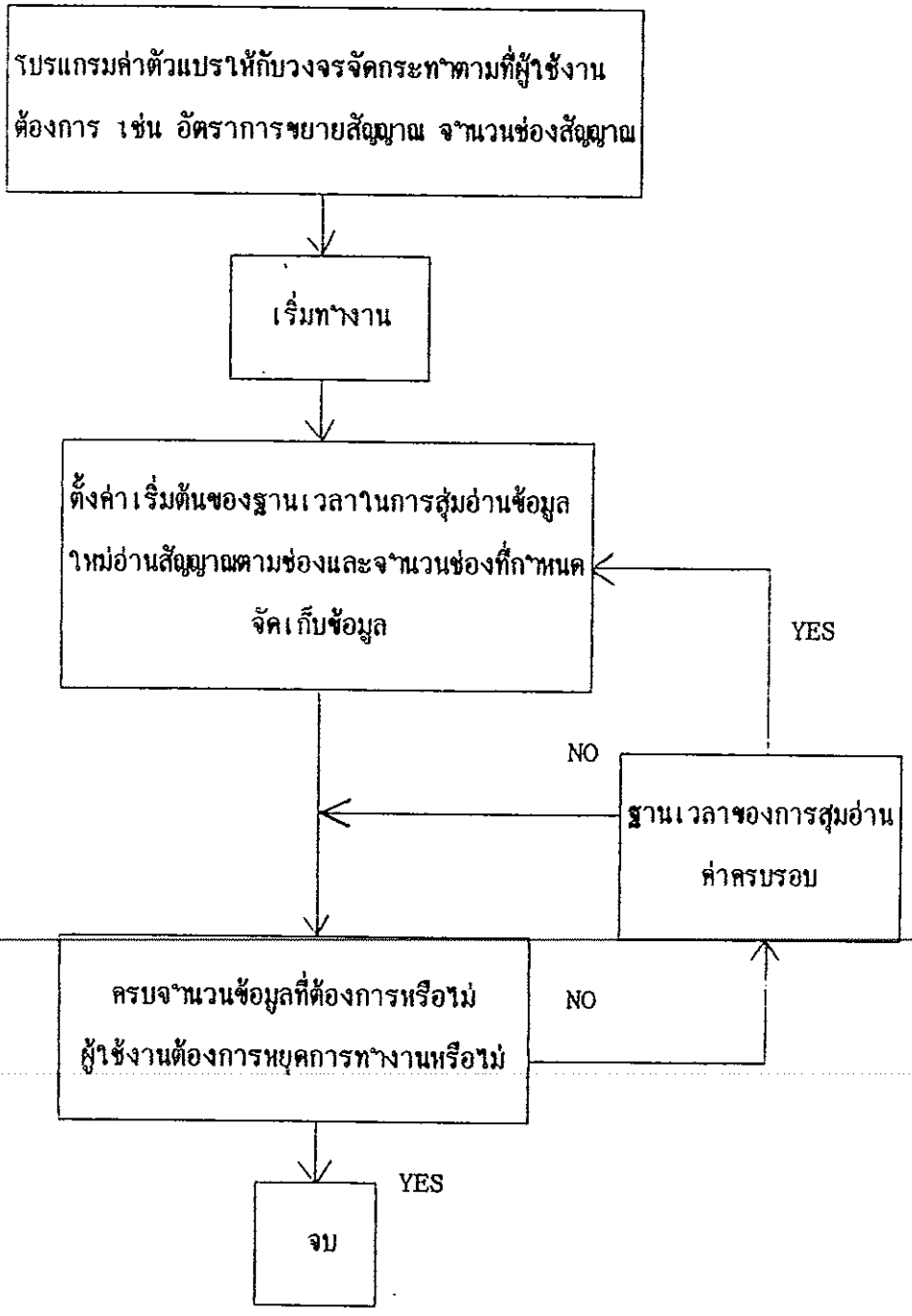
รูปแบบการทำงานของระบบเก็บข้อมูลระยะไกลในงานวิทยานิพนธ์

ก่อนที่จะบรรยายการทำงานของระบบเก็บข้อมูลระยะไกล ที่เป็นหัวข้อวิทยานิพนธ์ จะขอกล่าวถึงรูปแบบการทำงานของระบบเก็บข้อมูลสำหรับงานวิจัยโดยทั่วไป

การทำงานของระบบเก็บข้อมูลมาตรฐานที่มีขาย จะเป็นระบบที่ติดตั้งบนไมโครคอมพิวเตอร์ในของสลอดส่วนขยายระบบ (EXPANDSION SLOT) สำหรับข้อมูลที่ต้องการวัด ซึ่งผ่านการจัดกระทำจนได้สัญญาณเป็นไปตามข้อกำหนดของระบบแล้วจะถูกป้อนเข้าสู่ภาครับสัญญาณ ซึ่งโดยทั่วไปแล้วระบบเก็บข้อมูลเหล่านี้จะไม่สามารถวัดและประมวลผลสัญญาณที่อยู่ไกลจากตัวระบบได้ เนื่องจากสัญญาณที่ต้องการวัดซึ่งส่วนใหญ่จะเป็นสัญญาณแบบอนาลอก จะมีปัญหาของระดับสัญญาณที่ลดลงเป็นตัวแปรตามกับระยะทางการสั่งงานของระบบเก็บข้อมูล จะกระทำโดยผ่านทางโปรแกรมควบคุมการทำงานในไมโครคอมพิวเตอร์

การทำงานระบบเก็บข้อมูลจะอ่านค่าตัวแปรต่าง ๆ เช่นความเร็วในการ

สุ่มอ่านค่าสัญญาณ (SAMPLING RATE หรือ TRIGGERING TIME) จำนวนช่องสัญญาณ และอัตราการขยายสัญญาณ และกำหนดรูปแบบการทำงานที่เหมาะสม ซึ่งตัวแปรที่มีความสำคัญมาก คือ ความเร็วในการสุ่มสัญญาณ ซึ่งจำเป็นต้องใช้ตลอดเวลาที่ระบบทำงาน ผลของสัญญาณที่ต้องการวัดค่าจะมีความถูกต้องเพียงใดขึ้นอยู่กับการเลือกค่าตัวแปรนี้ โดยมีขั้นตอนการทำงานแสดงในรูปที่ 2.5



รูปที่ 2.8 แผนภูมิสายงานการทำงานโดยสังเขปของระบบเก็บข้อมูลโดยทั่วไป

สำหรับแนวคิดของการออกแบบระบบเก็บข้อมูลระยะไกลนั้น ได้ใช้รูปแบบของการจัดกระทำทางเครือข่ายดังที่ได้กล่าวไปแล้ว โดยหลีกเลี่ยงการประมวลผลสัญญาณข้อมูลแบบอนาลอก และขยายขีดความสามารถของระบบเก็บข้อมูลให้สามารถนำข้อมูลจากจุดทดลอง ซึ่งอยู่ห่างไกลมาประมวลผลได้โดยการแปลงสัญญาณข้อมูลที่ต้องการวัดเป็นข้อมูลแบบดิจิตอล ซึ่งเป็นหน้าที่ของระบบเครือข่ายย่อย เพื่อส่งให้กับระบบควบคุมหลัก ตามโครงสร้างของระบบที่ได้บรรยายไปแล้วในหัวข้อ 2.1

รูปแบบการสื่อสารข้อมูลภายในระบบเก็บข้อมูลระยะไกล ระหว่างเครือข่ายย่อยและระบบควบคุมหลักอยู่ในรูปแบบอนุกรม ซึ่งมีข้อดี คือ ระยะห่างระหว่างระบบสามารถขยายได้ตามความต้องการ แต่ข้อเสียของการสื่อสารข้อมูลแบบอนุกรม คือ ความเร็วในการสื่อสารซึ่งมีผลต่อความเร็วในการทำงานของระบบทั้งหมด ข้อจำกัดเกี่ยวกับความเร็วในการทำงานนี้ มีผลกระทบต่อการเก็บข้อมูลของการทดลองในบางรูปแบบ ซึ่งทำให้สามารถแบ่งรูปแบบการทำงานของระบบเก็บข้อมูลระยะไกลเป็นสองรูปแบบ คือ

1. **การเก็บข้อมูลในรูปแบบไม่เป็นเวลาจริง** (NON REAL-TIME DATA AQUISITION)

การเก็บข้อมูลในรูปแบบไม่เป็นเวลาจริง ใช้ในงานการทดลองประเภทที่ต้องการความเร็วในการอ่านค่าข้อมูลจากจุดทดลองต่าง ๆ สูง เช่น การบันทึกข้อมูลประเภทเกิดในช่วงเวลาสั้น ๆ ซึ่งต้องการความเร็วในการสุ่มอ่านค่าข้อมูล (SAMPLING RATE) จากจุดทดลองทุกจุดพร้อม ๆ กันด้วยความเร็วสูง ซึ่งงานการบันทึกในรูปแบบเช่นนี้ ไม่สามารถอ่านค่าจากระบบเครือข่ายย่อย เข้าสู่ระบบควบคุมหลัก และระบบคอมพิวเตอร์เพื่อแสดงผลได้ทันที เนื่องจากระบบการสื่อสารข้อมูลไม่อำนวยต่อการจัดการ ดังนั้น จึงจำเป็นต้องมีการจัดเก็บข้อมูลที่ได้ไว้ในหน่วยความจำของระบบเครือข่ายย่อยเพื่อรอคำสั่งในการเรียกอ่านข้อมูล

2. **การเก็บข้อมูลในรูปแบบเวลาจริง**(REAL-TIME DATA ACQUISITION)

การเก็บข้อมูลในรูปแบบเวลาจริง ใช้ในงานการทดลองที่ต้องการติดตามความเปลี่ยนแปลงของตัวแปรการทดลองเป็นฟังก์ชั่นของเวลา โดยมีค่าของช่วงเวลาที่แน่นอน และไม่เกินขอบเขตความสามารถในการประมวลผลระบบควบคุม

การทำงานหลัก (การอ่านค่าข้อมูลจากระบบเครือข่ายย่อยทุกตัวที่กำหนด ซึ่งหากจำนวนระบบเครือข่ายย่อยมากจำเป็นต้องใช้เวลานาน) รูปแบบเช่นนี้ระบบควบคุมการทำงานหลัก สามารถรับข้อมูลในแต่ละช่วงเวลา และจัดเก็บข้อมูลในหน่วยความจำและ/หรือส่งต่อให้กับคอมพิวเตอร์เพื่อแสดงผลได้ทันที

การทำงานของระบบทั้งสองรูปแบบระบบเครือข่ายย่อย มีตัวแปรร่วมที่จำเป็นต้องได้รับจากระบบควบคุมหลัก คือ คำสั่งเริ่มสุ่มอ่านค่าสัญญาณในรูปแบบของรหัสทางการสื่อสารแบบอนุกรม ซึ่งระบบเครือข่ายย่อยทุกตัวต้องได้รับสัญญาณนี้พร้อม ๆ กัน เพื่อความถูกต้องในการจัดเก็บข้อมูลของระบบเก็บข้อมูลระยะไกล

บทที่ 3

สถาปัตยกรรมด้านฮาร์ดแวร์

จากโครงสร้างของระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์ที่ได้กล่าวมาแล้วในบทที่ 2 ซึ่งเป็นแบบย่อ ๆ จะได้นำมากล่าวถึงรายละเอียดแยกเป็นส่วน ๆ ในแง่ของหน้าที่ การออกแบบระบบ และการทำงานขององค์ประกอบทั้งหมด ซึ่งคาดว่าจะเป็นประโยชน์ในการนำมาประยุกต์ใช้ หรือ การดัดแปลงเพื่อใช้งานที่เหมาะสมต่อไป

ไมโครโปรเซสเซอร์และการออกแบบ

เนื่องจากระบบย่อยต่าง ๆ ที่เป็นองค์ประกอบของระบบเก็บข้อมูลระยะไกลในงานวิทยานิพนธ์นี้ต่างก็มีไมโครโปรเซสเซอร์เป็นตัวควบคุมการทำงาน ดังนั้นจึงจำเป็นต้องกล่าวถึงไมโครโปรเซสเซอร์ และการออกแบบระบบที่ควบคุมโดยไมโครโปรเซสเซอร์ก่อนที่จะกล่าวในรายละเอียดของระบบย่อยในงานวิทยานิพนธ์

1. **ไมโครโปรเซสเซอร์** โดยทั่วไปแล้วมีความหมายถึง ระบบเอนกประสงค์ที่มีหน่วยประมวลผลกลาง (CENTRAL PROCESSING UNIT : CPU) จนในบางครั้งอาจเข้าใจว่าไมโครโปรเซสเซอร์ เป็นระบบคอมพิวเตอร์ขนาดเล็กในชิ้นสารกึ่งตัวนำ ซึ่งความจริงแล้วคอมพิวเตอร์ ไม่สามารถทำงานได้โดยอาศัยเพียงไมโครโปรเซสเซอร์ โดยปกติแล้วไมโครโปรเซสเซอร์มีองค์ประกอบดังรูปที่ 3.1

ระบบของคอมพิวเตอร์จำเป็นต้องมีส่วนประกอบที่สำคัญเพิ่มเติมนอกเหนือจากไมโครโปรเซสเซอร์ คือ

1.1 หน่วยความจำส่วนโปรแกรมงาน ซึ่งเป็นตัวเก็บโปรแกรมการทำงานของคอมพิวเตอร์ประเภทถาวร จัดเป็นหน่วยความจำประเภท ROM (READ ONLY MEMORY)

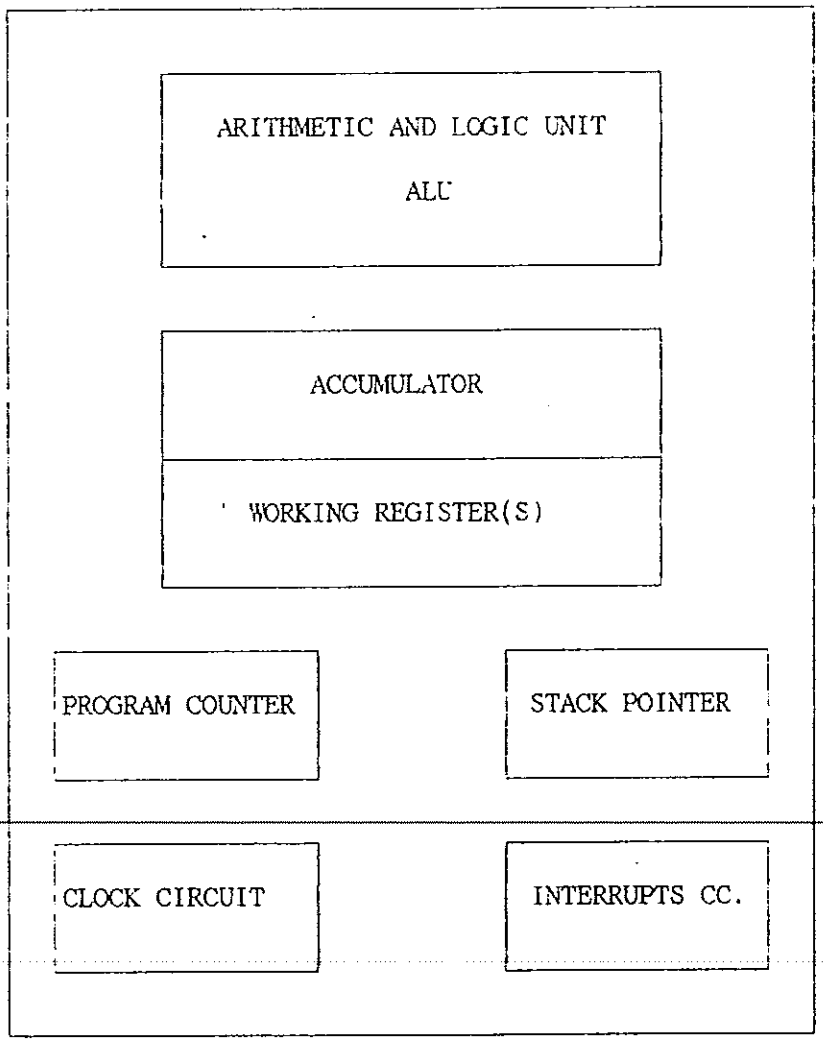
1.2 หน่วยความจำสำหรับเก็บค่าตัวแปร หรือข้อมูลที่ได้จากการทำงาน

รดยที่คอมพิวเตอร์สามารถอ่านค่าและเขียนค่าเข้าไปเก็บได้ จัดเป็นหน่วยความจำประเภท RAM (RANDOM ACCESS MEMORY)

1.3 วงจรถอดรหัสสำหรับเลือกติดต่อกับหน่วยความจำ(MEMORY DECODER)

1.4 วงจรกำเนิดสัญญาณนาฬิกาซึ่งเป็นฐานเวลาของระบบ

1.5 พอร์ทอินพุทและเอาท์พุท (INPUT AND OUTPUT PORT) ทั้งแบบขนานและอนุกรม



รูปที่ 3.1 องค์ประกอบของไมโครโปรเซสเซอร์

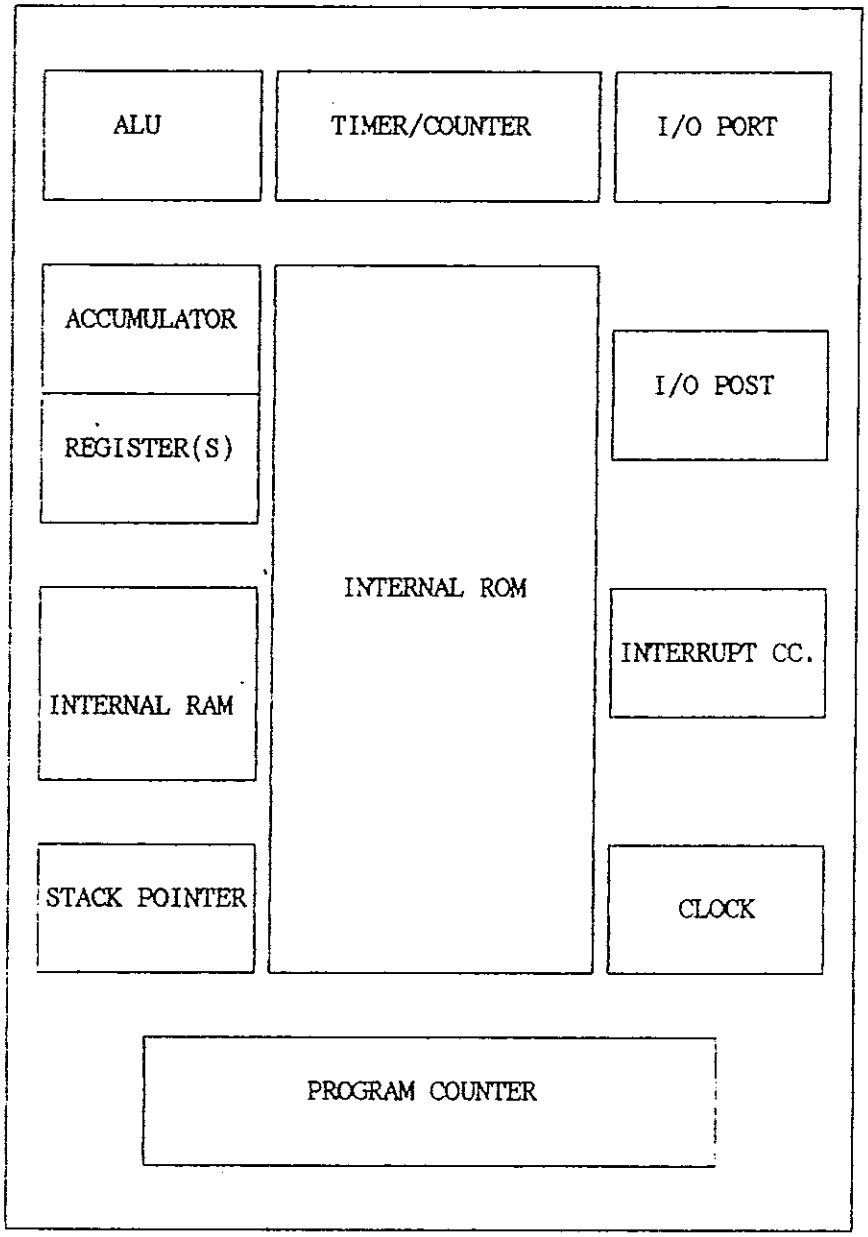
นอกจากนี้ ระบบคอมพิวเตอร์ยังอาจจำเป็นต้องมีระบบอื่น ๆ เพิ่มเติม ได้แก่ ส่วนบริการอินเตอร์รัพท์ ส่วนบริการสำหรับติดต่อกับหน่วยความจำโดยตรง (DIRECT MEMORY ACCESS:DMA) ส่วนบริการอ่านหรือเขียนข้อมูลลงในสื่อกลางข้อมูลแบบต่าง ๆ เช่น FLOPPY DISK DRIVE,HARD DISK DRIVE

ดังนั้น จึงค่อนข้างมีเหตุผลที่เรียกไมโครโปรเซสเซอร์ ว่าเป็นหน่วยประมวลผลแบบอเนกประสงค์ เนื่องจากผู้ใช้งานสามารถเพิ่มเติมอุปกรณ์ประกอบได้ตามความต้องการภายใต้การควบคุมโดยโปรแกรมจัดการ และสถาปัตยกรรมของไมโครโปรเซสเซอร์

2. **ไมโครคอนโทรลเลอร์** พิจารณาระบบของไมโครคอนโทรลเลอร์จากรูปที่ 3.2

ระบบไมโครคอนโทรลเลอร์ จัดเป็นรูปแบบของระบบคอมพิวเตอร์ในชิ้นสารกึ่งตัวนำ โดยมีองค์ประกอบของไมโครโปรเซสเซอร์ที่ครบถ้วนพอที่จะใช้งานได้ คือ มี ARITHMETIC LOGIC UNIT,PROGRAM COUNTER. STACK POINTER และ REGISTER และได้เพิ่มองค์ประกอบที่ทำให้เป็นคอมพิวเตอร์สมบูรณ์แบบคือ RAM ROM พอร์ทอินพุทและเอาท์พุทแบบขนาน พอร์ทอินพุทและเอาท์พุทแบบอนุกรม COUNTER และวงจรกำเนิดสัญญาณนาฬิกา

ดังนั้น ไมโครคอนโทรลเลอร์ ก็คือไมโครโปรเซอร์ที่มีองค์ประกอบครบถ้วนอยู่ภายในตัวเดียวสำหรับวงจรรวม(CHIP) ที่จัดเป็นไมโครโปรเซสเซอร์ได้แก่ Z80 (ZILOG) 80X80 (INTEL) ส่วนวงจรรวมที่จัดเป็นไมโครคอนโทรลเลอร์ได้แก่ ตระกูล MCS-51 ของ INTEL MC68HC11 ของ MOTOROLA เป็นต้น



รูปที่ 3.2 องค์ประกอบของไมโครคอนโทรลเลอร์

สำหรับในงานวิทยานิพนธ์ ได้เลือกใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ของ INTEL โดยเจาะจงที่ไมโครคอนโทรลเลอร์เบอร์ 8031

สาเหตุที่เลือกใช้ไมโครคอนโทรลเลอร์ สำหรับงานวิทยานิพนธ์ ซึ่งเป็นการเก็บข้อมูลในลักษณะเวลาจริง เนื่องจากระบบควบคุมโดยไมโครคอนโทรลเลอร์

มีขนาดเล็ก ราคาต้นทุนต่ำ สามารถใช้งานได้โดยไม่จำเป็นต้องเพิ่มเติมอุปกรณ์ภายนอกมากนัก ประสิทธิภาพในการทำงานประเภทระบบควบคุมค่อนข้างสูง เพราะคำสั่งงานที่ใช้ควบคุมการทำงานเป็นคำสั่งที่ใช้เวลาในการทำงานสั้น และเด่นชัดมีกลุ่มคำสั่งที่เกี่ยวข้องกับการกระทำทางอินพุทและเอาท์พุทมาก และนอกจากนั้นโครงสร้างของการตอบสนองทางอินเตอร์รัพท์ค่อนข้างรวดเร็ว ทันต่อสถานการ์การเปลี่ยนแปลงของระบบที่ต้องการติดต่อด้วย

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ของบริษัท INTEL CORP.

ไมโครคอนโทรลเลอร์ของบริษัท INTEL CORP ในตระกูล MCS-51 ได้รับการพัฒนามาจากตระกูล MCS-48 โดยเพิ่มประสิทธิภาพการทำงานทั้งชุดคำสั่งและสถาปัตยกรรมทางด้านฮาร์ดแวร์สมาชิกในตระกูล MCS-51 จะเริ่มจาก 8031 ไปจนถึง 8751 ผู้ใช้งานสามารถเลือกประเภทของโครงสร้างภายในระบบไมโครคอนโทรลเลอร์ได้ในสองรูปแบบ คือ NMOS (N-CHANNEL METAL OXIDE SILICON) และ CMOS (COMPLEMENTARY METAL OXIDE SILICON) โดยโครงสร้างแบบ CMOS จะมีข้อได้เปรียบที่เด่นชัด คือ มีความต้องการกำลังไฟฟ้า (POWER CONSUMPTION) ต่ำ แต่มีราคาสูงกว่า NMOS

นอกจากสมาชิกดังที่กล่าวแล้ว INTEL ยังได้ออกไมโครคอนโทรลเลอร์ในตระกูลเดียวกันนี้ที่สามารถโปรแกรมสั่งงานโดยใช้ภาษา BASIC ทำให้การพัฒนาโปรแกรมสามารถกระทำได้ง่ายกว่าการใช้ภาษาแอสเซมบลี ในกรณีที่ผู้ใช้งานต้องการระบบไมโครคอนโทรลเลอร์ขนาดเล็ก INTEL ได้ติดตั้งหน่วยความจำประเภท EPROM ความจุ 4 กิโลไบท์ ซึ่งทำหน้าที่ในการเก็บโปรแกรมการทำงานของระบบไว้ในชิพเบอร์ 8751 ส่วนชิพเบอร์ 8051 นั้น ติดตั้งหน่วยความจำประเภท ROM ความจุ 4 กิโลไบท์ซึ่งเก็บโปรแกรมควบคุมเฉพาะอย่าง ไม่สามารถเปลี่ยนแปลงได้ และผู้ใช้งานจำเป็นต้องติดต่อกับบริษัทผู้ผลิตโดยตรง

สำหรับสถาปัตยกรรมหลักของไมโครคอนโทรลเลอร์ตระกูลนี้ จะประกอบด้วย

1. หน่วยประมวลผลกลางซึ่งมีขนาดของบัสข้อมูล 8 บิท ประกอบด้วย รีจิสเตอร์ A และ B

2. PROGRAM COUNTER และ DATA POINTER ขนาด 16 บิท ทำหน้าที่ในการอ้างอิงตำแหน่งข้อมูลหรือควบคุมบัสแอดเดรส (ADDRESS BUS)

3. PROGRAM STATUS WORD (PSW) ขนาด 16 บิท

4. STACK POINTER ขนาด 8 บิท เก็บค่าในหน่วยความจำภายใน

5. หน่วยความจำส่วนโปรแกรม:EPROM (8751), ROM(8051) และไม่ติดตั้งสำหรับ 8031

6. หน่วยความจำประเภท RAM 128 KBYTE

7. สามารถอ้างอิงหน่วยความจำส่วนโปรแกรมภายนอก 64 กิโลไบท์ และหน่วยความจำส่วนข้อมูลภายนอก 64 กิโลไบท์

8. อินพุท/เอาท์พุทพอร์ท ขนาด 32 บิท สามารถควบคุมในรูปแบบของบูลีน

9. TIMER/COUNTER ขนาด 16 บิท จำนวน 2 ชุดที่เป็นอิสระจากกัน

10. พอร์ทอนุกรมแบบ FULL DUPLEX

11. สามารถตอบสนองต่อการเกิดอินเทอร์รัพท์ภายนอก 2 แหล่ง และภายใน 3 แหล่ง

12. มีวงจรกำเนิดสัญญาณนาฬิกาของระบบ

คำแนะนำในการออกแบบระบบไมโครคอนโทรลเลอร์

ในการที่จะสามารถใช้ประโยชน์จากไมโครคอนโทรลเลอร์ได้นั้นผู้ใช้งานจำเป็นต้องมีความรู้ในเรื่องต่าง ๆ พอที่จะสรุปได้ดังนี้

1. ผู้ใช้งานจำเป็นต้องศึกษาทำความเข้าใจในพื้นฐานของไมโครคอนโทรลเลอร์ที่ได้เลือกใช้โดยเฉพาะอย่างยิ่งในโครงสร้างของรีจิสเตอร์ คำสั่งในการใช้งาน รูปแบบการอ้างอิงตำแหน่งข้อมูล(ADDRESSING MODE) การใช้งานอุปกรณ์ที่ติดตั้งมาในไมโครคอนโทรลเลอร์ การขยายระบบไมโครคอนโทรลเลอร์

เพื่อติดต่อกับอุปกรณ์เพิ่มเติมภายนอก เช่น อินพุทและเอาท์พุทพอร์ทเพิ่มเติม การติดต่อกับคีย์บอร์ดและภาพแสดงผล นอกเหนือจากความรู้ความเข้าใจในรูปแบบการทำงานทั่วไปแล้ว การทำงานในรูปแบบของอินเตอร์รัพท์ ก็ยังเป็นสิ่งที่มีความสำคัญมากในระบบไมโครคอนโทรลเลอร์ ที่จำเป็นต้องติดต่อกับระบบสิ่งแวดล้อมภายนอก

2. ผู้ใช้งานต้องมีความเข้าใจในขั้นตอนการทำงาน (ALGORITHMIC PROCESSES) ของระบบที่กำลังพิจารณา และสามารถเปลี่ยนขั้นตอนการทำงานให้เป็นโปรแกรมการทำงานจริงของไมโครคอนโทรลเลอร์ เช่น ในการออกแบบระบบควบคุมความหนาของการเคลือบผิวชิ้นสารตัวอย่าง โดยใช้พลาสมา (PLASMA COATING) ผู้ใช้งานจำเป็นต้องมีความเข้าใจในระบบ PLASMA COATING ว่าในกระบวนการนั้นตัวแปรที่จำเป็นต้องติดตาม และสามารถแปลงขั้นตอนดังกล่าวให้อยู่ในรูปของคำสั่งงานของไมโครคอนโทรลเลอร์ได้

3. ผู้ใช้งานควรมีความสามารถในการแยก ระบบสายงานการควบคุมเป็นส่วนย่อย และแทนที่ขั้นตอนงานในขั้นต่าง ๆ ด้วยวิธีการที่เหมาะสมในบางขั้นตอนของสายงานการควบคุมอาจมีทางเลือกหลายรูปแบบ ซึ่งการตัดสินใจในเลือกรูปแบบจะขึ้นอยู่กับประสพการณ์ เช่น ในขั้นตอนสายงานของระบบควบคุมอาจมีทางเลือกให้สามารถแก้ปัญหาโดยการใช้ซอร์ฟแวร์หรือฮาร์ดแวร์ การเลือกรูปแบบขึ้นอยู่กับข้อดีและข้อเสีย

รูปแบบการแสดงผลของระบบไมโครคอนโทรลเลอร์

ส่วนประกอบแรกของระบบเก็บข้อมูลระยะไกล ที่ทำหน้าที่ในการติดต่อกับผู้ใช้งาน(USER INTERFACE) เพื่อรายงานคำสั่งงาน สถานะของการทำงานและค่าของตัวแปรตามที่ต้องการโดยปกติแล้ว ระบบควบคุมขนาดเล็กที่ใช้ไมโครคอนโทรลเลอร์ เป็นตัวจัดการจะมีรูปแบบการแสดงผลเพื่อติดต่อกับผู้ใช้งานเป็น 2 รูปแบบ คือ

1. **การแสดงผลโดยการใช้ภาคแสดงผลแบบ 7 ส่วน (7 SEVEN SEGMENT DISPLAY)**

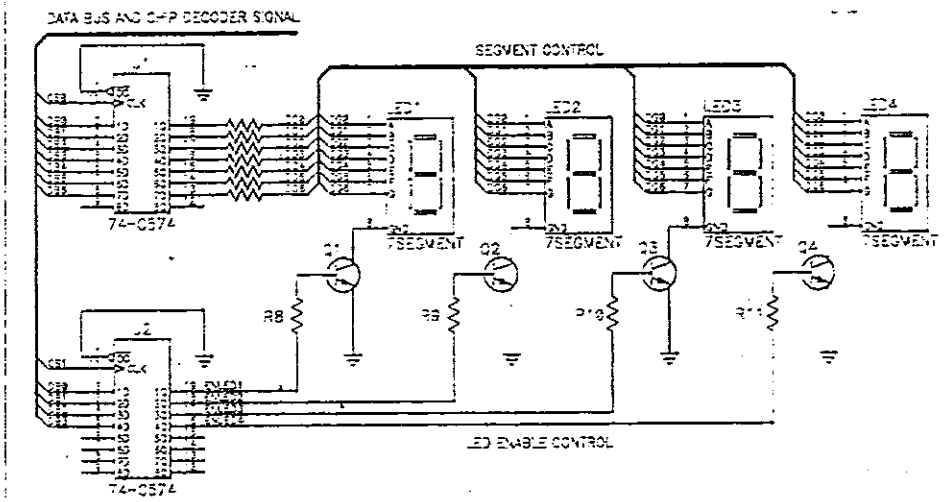
การแสดงผลในรูปแบบนี้ มักพบในการรายงานผลทางตัวเลขเพราะสามารถสื่อความหมายให้กับผู้ใช้งานได้ครบถ้วน และสามารถเพิ่มจำนวนหลักของข้อมูลได้ตามต้องการ อุปกรณ์ที่ใช้เป็นตัวแสดงผลอาจเป็น

1.1 LED (LIGHT EMITTING DIODE) เป็นอุปกรณ์ประเภทสารกึ่งตัวนำที่นำมาวางรูปแบบเป็นตัวแสดงผล 7 ส่วน ซึ่งมีข้อดี คือ สามารถมองเห็นได้ในระยะไกล เนื่องจากการแสดงผลเป็นแบบเปล่งแสง ปัจจุบันสามารถเลือกสีได้เป็น สีแดง สีเขียว และสีส้ม(เกิดจากการผสมของสีแดงและสีเขียว) นอกจากนี้ยังมีรายละเอียดของความยาวคลื่นที่แสดงให้เลือกเป็นสีต่าง ๆ ได้อีกมากมาย ข้อเสียของอุปกรณ์แสดงผลแบบนี้คือต้องการกระแสค่อนข้างมากในการขับ LED สำหรับรูปแบบการใช้งาน LED เพื่อแสดงโดยการควบคุมของไมโครคอนโทรลเลอร์ จะใช้รูปแบบที่เรียกว่า TIME MULTIPLEXING โดยมีวงจรในการควบคุมเป็นไปตามรูปที่ 3.3

ในรูปแบบการแสดงผลแบบ TIME MULTIPLEXING นั้น จะทำให้สามารถลดจำนวนของเอาท์พุทพอร์ ที่ทำหน้าที่ในการควบคุมการทำงานของ LED ในแต่ละตัว โดยแทนที่เอาท์พุทพอร์ทจะถูกต่อเข้ากับ LED ทุกตัว ที่ต้องการผ่านตัวต้านทานจำกัดกระแสให้เหมาะสมกับการทำงาน สำหรับขารวมของ LED(COMMON) จะถูกควบคุมโดยสวิทชิ่งทรานซิสเตอร์ (SWITCHING TRANSISTOR) ซึ่งต้องสามารถรับกระแสที่ไหลผ่านได้มากกว่า 8 เท่าของกระแสที่ไหลผ่านแต่ละเซกเมนต์ (SEGMENT) เอาท์พุทพอร์ท จะทำหน้าที่ในการควบคุมการทำงานของสวิทชิ่งทรานซิสเตอร์ ภายใต้โปรแกรมควบคุมการทำงาน

โปรแกรมควบคุม จะทำหน้าที่ในส่งข้อมูลออกทางเอาท์พุทพอร์ที่ควบคุม LED ตามค่าที่สอดคล้องกับหลักของ LED ที่ต้องการแสดงผลจากนั้นก็จะเปิดการทำงานของสวิทชิ่งทรานซิสเตอร์เพื่อให้กระแสไหลผ่าน LED หลักนั้น ๆ ได้อย่างครบวงจร ซึ่งเท่ากับ LED หลักที่ต้องการแสดงผลตามค่าที่ส่งออก หลังจากนั้นก็จะเลือกแสดงผลในหลักต่อไปด้วยวิธีเดียวกัน

เมื่อความเร็วในการเลือกแสดงผลหลักต่าง ๆ สูงพอ (สูงกว่าความสามารถในแยกการติดดับของสายตามนุษย์)ก็จะทำให้ผู้ใช้งานมองเห็น LED ทุกหลักติดพร้อมกัน



รูปที่ 3.3 วงจรควบคุมการแสดงผลของ LED แบบตัวเลขโดยวิธี TIME MULTIPLEXING

1.2 LCD (LIQUID CRYSTAL DISPLAY) ในรูปแบบตัวแสดงผลแบบ 7 ส่วน การควบคุมการแสดงผลทำได้ยาก เนื่องจากการตอบสนองต่อการเปลี่ยนแปลงของ LCD ค่อนข้างช้าไม่สามารถใช้ TIME MULTIPLEXING ได้และวงจรในการควบคุมค่อนข้างยุ่งยาก โดยทั่วไปแล้วผู้ผลิตจะสร้างวงจรแสดงผลแบบนี้เป็นชุด ซึ่งสามารถส่งข้อมูล เพื่อแสดงผลได้ทันทีทั้งแบบตัวอักษรและตัวเลขเรียกว่า INTELLIGENT ALPHANUMERIC DISPLAY ข้อเสียของภาคแสดงผลที่ใช้ LCD คือไม่มีแสงสว่างในตัวเองจำเป็นต้องใช้แหล่งกำเนิดแสงสว่างในการใช้งานในที่มืด

2. การแสดงผลโดยใช้ภาคแสดงผลแบบดอทเมทริกซ์ (DOT MATRIX DISPLAY)

เนื่องจากข้อจำกัดของการใช้ภาคแสดงผลแบบ 7 ส่วน คือ ไม่สามารถแสดงตัวหนังสือได้ การติดต่อต่อกับผู้ใช้งานบางประเภทจึงไม่สะดวก ดังนั้นจึงมีการพัฒนาระบบแสดงผลเป็นแบบ ALPHANUMERIC DISPLAY สำหรับในงาน

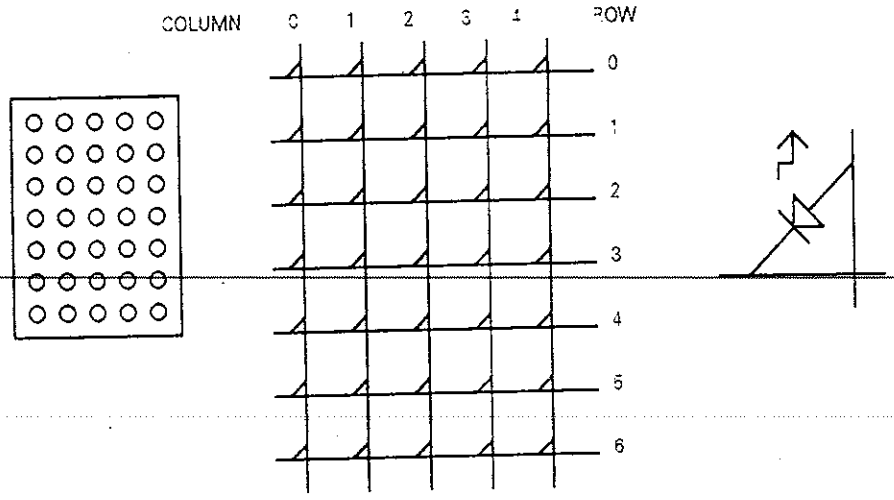
วิทยานิพนธ์ได้เลือกใช้รูปแบบการแสดงผลนี้ โดยออกแบบระบบจัดการแสดงผลขึ้นมาเองภายใต้การควบคุมโดยไมโครคอนโทรลเลอร์ และภาคแสดงผล เป็นแบบดอทเมทริกซ์ (DOT MATRIX DISPLAY)

ระบบแสดงผล

ได้เลือกใช้รูปแบบการแสดงผลแบบ ALPHANUMERIC คือ สามารถแสดงได้ทั้งตัวอักษรภาษาอังกฤษทั้ง UPPERCASE และ LOWERCASE และตัวเลข การควบคุมการทำงาน และการจัดการต่าง ๆ กระทำโดยไมโครคอนโทรลเลอร์ 8031 สำหรับข้อมูลที่ต้องการแสดงจะถูกส่งเข้าสู่ระบบแสดงผล โดยผ่านพอร์ทเชื่อมโยงมาตรฐานแบบขนาน

1. **การควบคุมการแสดงผลของภาคแสดงผลแบบดอทเมทริกซ์**

รูปร่างและวงจรการเชื่อมต่อภายในของภาคแสดงผล LED แบบดอท-เมทริกซ์ขนาด 5X7 เป็นดังรูปที่ 3.4



รูปที่ 3.4 ภาคแสดงผล LED แบบดอทเมทริกซ์ขนาด 5X7

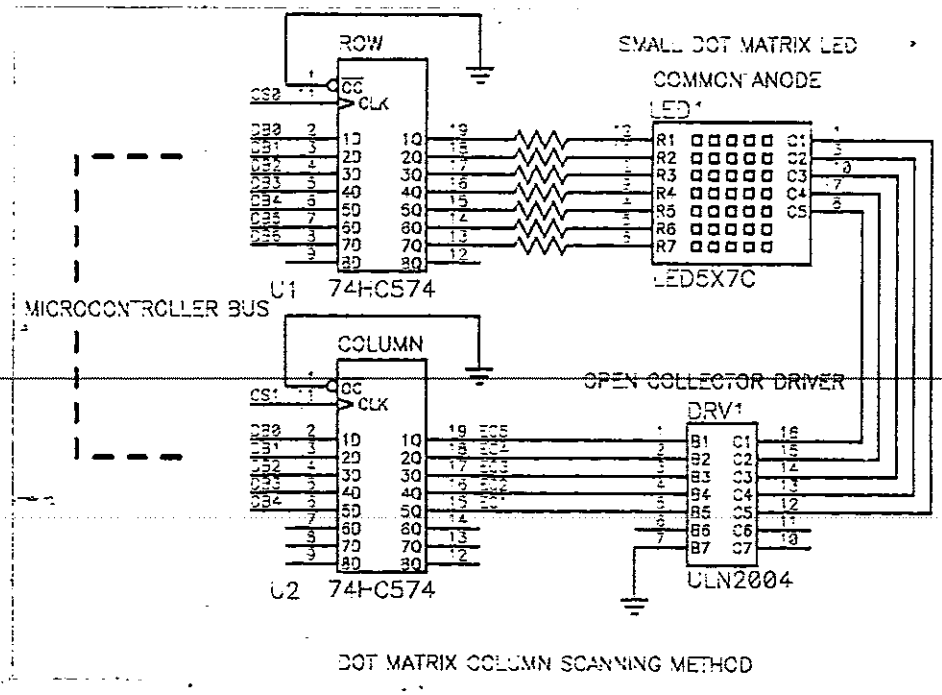
ประเภทของภาคแสดงผล LED แบบดอทเมทริกซ์ สามารถแบ่งย่อยได้เป็นอีกสองรูปแบบขึ้นอยู่กับวงจรการเชื่อมต่อของ LED ภายในตัวดอทเมทริกซ์ คือ ถ้าขาที่เชื่อมต่อในแถว (ROW) เดียวกันของ LED ที่เป็นองค์ประกอบของดอท-เมทริกซ์เป็นขาแอโนด (ANODE) เรียกดอทเมทริกซ์แบบนี้ว่า ดอทเมทริกซ์แบบคอมมอนแอโนด (COMMON ANODE DBT MATRIX) และในทางตรงกันข้ามสำหรับกรณีของดอทเมทริกซ์แบบคอมมอนแคโทด (COMMON CATHODE DOT MATRIX) ในการนำไปใช้งานจริงจะมีความแตกต่างของลอจิกที่ใช้ควบคุมการทำงาน ในกรณีที่ระบบการแสดงผลไม่จำเป็นต้องใช้วงจรขยายกระแสทั้งสองแบบ คือ

กรณีที่ใช้ดอทเมทริกซ์แบบคอมมอนแอโนดและใช้วงจรขยายกระแส ภาคเอาท์พุทของวงจรขยายกระแสที่ควบคุมทางด้านคอลัมน์จะต้องเป็นทรานซิสเตอร์แบบ NPN เปิดคอลเลกเตอร์(OPEN COLLECTOR) ซึ่งปัจจุบันได้มีการรวมวงจรขยายกระแสเป็นชิพและชิพที่มีรูปแบบการทำงานแบบนี้ได้แก่ ULN2003, ULN2803 และ UCN5833 ของบริษัท SPRAQUE จำกัด โดยชิพเหล่านี้ สามารถจ่ายกระแสเอาท์พุทได้สูงถึง 500 มิลลิแอมป์ต่อเอาท์พุท ดังนั้น ผู้ใช้งานจำเป็นต้องต่อตัวต้านทานจำกัดกระแสที่จะจ่ายให้กับ LED ให้เป็นไปตามเงื่อนไขข้อกำหนดของ LED ที่เลือกใช้งาน(ตามปกติแล้วกระแสที่ใช้ในการจ่าย LED ให้สามารถทำงานได้จะไม่เกิน 15 มิลลิแอมป์ ในการใช้งานแบบต่อเนื่อง แต่เมื่อนำมาใช้งานแบบ TIME MULTI-PLEXING จะสามารถป้อนกระแสให้กับ LED ได้สูงกว่าปกติขึ้นอยู่กับเวลาในการนำกระแสของเพาว์เวอร์ทรานซิสเตอร์) ที่เป็นองค์ประกอบของดอทเมทริกซ์ส่วนสวิท-ชิ่งทรานซิสเตอร์ที่ทำหน้าที่ในการควบคุมดอทเมทริกซ์ทางด้านแถวจะต้องเป็น PNP

ส่วนการเลือกใช้ดอทเมทริกซ์แบบคอมมอนแคโทดนั้น ภาคเอาท์พุทของวงจรขยายที่ควบคุมด้านคอลัมน์ จะต้องประกอบด้วยทรานซิสเตอร์แบบ PNP เปิดคอลเลคเตอร์ เช่น ชิพเบอร์ 75491 ของบริษัท TEXAS INSTRUMENT INC. และสวิทชิ่งทรานซิเตอร์ที่ควบคุมด้านแถวจะต้องเป็นแบบ NPN

รูปที่ 3.5 จะแสดงตัวอย่างการควบคุมภาคแสดงผลแบบดอทเมทริกซ์โดยใช้ LED แบบสแกน COMMON ANODE โดยใช้เทคนิคในการแสดงผลแบบ TIME MULTIPLEXING

เทคนิคการแสดงผลแบบ TIME MULTIPLEXING สำหรับดอทเมทริกซ์ที่นิยมาใช้นั้นจะเป็นแบบสแกน (SCAN) ด้านแถว โดยที่แถวเดียวกันของดอทเมทริกซ์ทุกตัวในระบบแสดงผลจะถูกเชื่อมต่อเข้าด้วยกัน และถูกควบคุมการไหลของกระแสด้วยสวิทชิ่งเพาเวอร์ทรานซิสเตอร์(SWITCHING POWER TRANSISTOR) 1 แถวต่อ 1 ตัว(ขึ้นอยู่กับความสามารถในการรับกระแสของทรานซิสเตอร์หากไม่สามารถหาทรานซิสเตอร์ที่สามารถรับกระแสสูงมากได้ ก็อาจใช้ทรานซิสเตอร์ควบคุมเป็นชุด ๆ โดยใช้สัญญาณควบคุมทางด้านขา BASE ชุดเดียวกัน) ข้อมูลที่ต้องการแสดงในแต่ละแถว จะถูกส่งผ่านเอาท์พุทพอร์ทที่ควบคุมด้านคอลัมน์ (อาจผ่านวงจรขยายกระแสและตัวต้านทานจำกัดกระแส) สอดคล้องกับสัญญาณควบคุมการทำงานของสวิทชิ่งเพาเวอร์ทรานซิสเตอร์ทางด้านแถว การสแกนแบบนี้สามารถใช้ได้กับระบบแสดงผลที่ต้องการแสดงแบบหลายหลัก เพียงแค่เพิ่มเอาท์พุทพอร์ทที่ควบคุมด้านคอมลัมน์และเปลี่ยนสวิทชิ่งเพาเวอร์ทรานซิสเตอร์



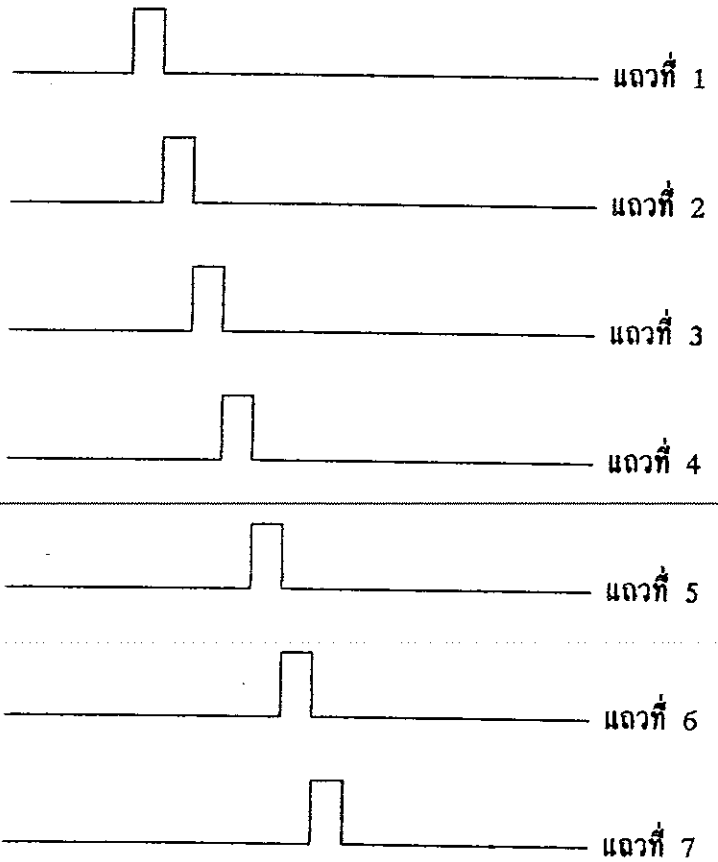
รูปที่ 3.5 ตัวอย่างวงจรควบคุมการแสดงผล LED แบบดอทเมทริกซ์ 1 ตัว

การเลือกชนิดของสวิทชิ่งเพาเวอร์ทรานซิสเตอร์ จะขึ้นอยู่กับจำนวนคอลัมน์โดยรวมของระบบแสดงผล (N) และกระแสที่ต้องการใน 1 จุดของ LED ที่เป็นองค์ประกอบของดอทเมทริกซ์ (I)

จากตัวแปรที่กำหนด จะได้ว่าสวิทชิ่งทรานซิสเตอร์ จะต้องสามารถรับกระแสสูงสุด I_{max} (เกิดในกรณีที่ในแถวใด ๆ ของระบบแสดงผลติดพร้อมกันหมด)

$$I_{max} = NI$$

ดังที่กล่าวแล้วว่า เทคนิคในการแสดงผลแบบ TIME MULTIPLEXING แบบสแกน จำเป็นต้องมีสัญญาณควบคุมการทำงานของสวิทชิ่งเพาเวอร์ทรานซิสเตอร์ โดยแผนผังเวลาเปรียบเทียบของสัญญาณจะเป็นดังรูป 3.6 สัญญาณควบคุมเหล่านี้จะทำงานหลังจากที่ข้อมูลที่ต้องการแสดงในแถวนั้น ๆ ถูกส่งเข้าสู่พอร์ททางด้านคอลัมน์เรียบร้อยแล้ว



รูปที่ 3.6 แสดงแผนผังเวลาเปรียบเทียบของการเกิดสัญญาณควบคุมสวิทชิ่งเพาเวอร์ทรานซิสเตอร์

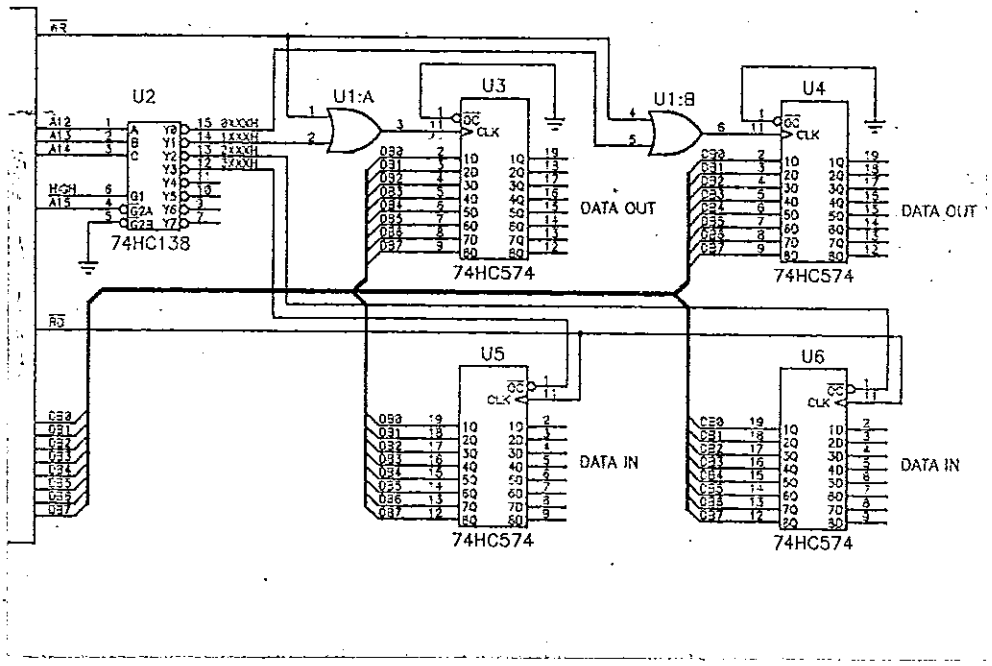
2. เอาท์พุทพอร์ทควบคุมการแสดงผล

ในระบบควบคุมด้วยไมโครคอนโทรลเลอร์ เอาท์พุทพอร์ทเป็นสิ่งที่มีความสำคัญถึงแม้ว่าไมโครคอนโทรลเลอร์ 8031 จะมีเอาท์พุทพอร์ทอยู่ในระบบแล้วก็ตาม แต่เนื่องจากจำนวนพอร์ทที่สามารถต่อใช้งานได้จริงมีไม่มากนัก อันเนื่องมาจากระบบควบคุมโดยไมโครคอนโทรลเลอร์ 8031 จำเป็นต้องอ้างอิงหน่วยความจำส่วนโปรแกรมและหน่วยความจำส่วนข้อมูลภายนอก ซึ่งจำเป็นต้องใช้พอร์ทถึง 2 พอร์ท(เพื่อกำเนิดบัสข้อมูลและบัสแอดเดรส) ดังนั้นพอร์ทของไมโครคอนโทรลเลอร์ที่สามารถใช้งานจริงเหลือเพียง 2 พอร์ท คือ พอร์ท P1 และ P2

จากข้อจำกัดของระบบควบคุมด้วยไมโครคอนโทรลเลอร์ 8031 ที่ต้องการใช้เอาท์พุทพอร์ทจำนวนมาก ดังนั้นจึงจำเป็นต้องออกแบบเอาท์พุทพอร์ทเพิ่มเติมให้กับระบบ โดยมีทางเลือกสำหรับการออกแบบ ดังรายละเอียด

2.1 ถอดรหัสแอดเดรสโดยตรง เพื่ออ้างอิงอินพุทหรือเอาท์พุทเป็นตัว ๆ ไป แต่ละตัวของพอร์ทจะมีแอดเดรสในการอ้างอิงเฉพาะ ดังนั้นจึงลดความยุ่งยากของไมโครคอนโทรลเลอร์ในการติดต่อกับพอร์ท แต่มีข้อเสียอันเกิดจากการอ้างอิงแอดเดรส เนื่องจากจำเป็นต้องอ้างหมายเลขแอดเดรสทุกครั้งในการติดต่อ รูปแบบการถอดรหัสโดยตรงเป็นรูปแบบที่ใช้งานในระบบไมโครคอนโทรลเลอร์/ไมโครโปรเซสเซอร์โดยทั่วไป

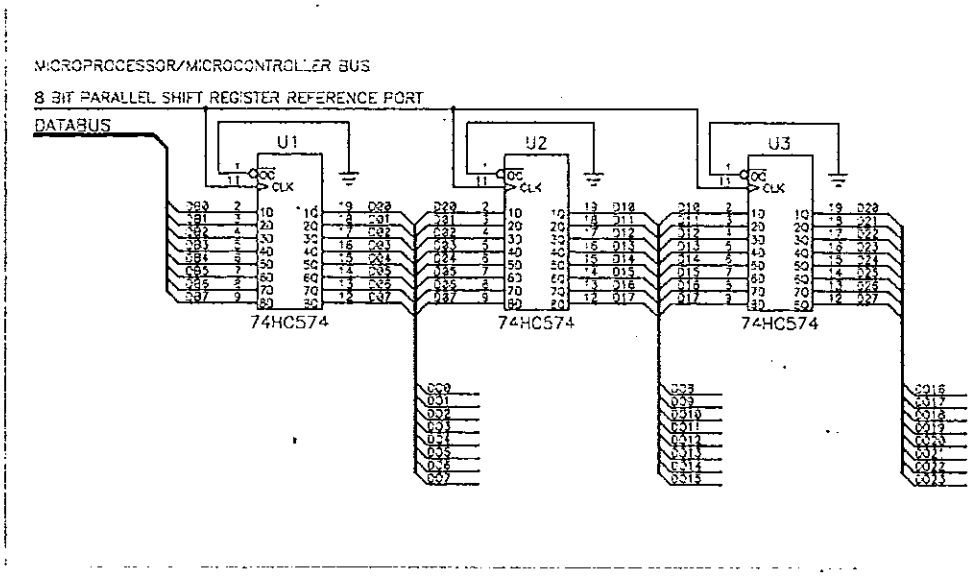
2.2 การถอดรหัสบัสแอดเดรสในการอ้างอิงอินพุท หรือเอาท์พุทที่มีรูปแบบเป็นชิฟรีจิสเตอร์แบบขนาน (PARALLEL SHIFT REGISTER) รูปแบบของวงจรชิฟรีจิสเตอร์แบบขนานเป็นดังรูปที่ 3.7



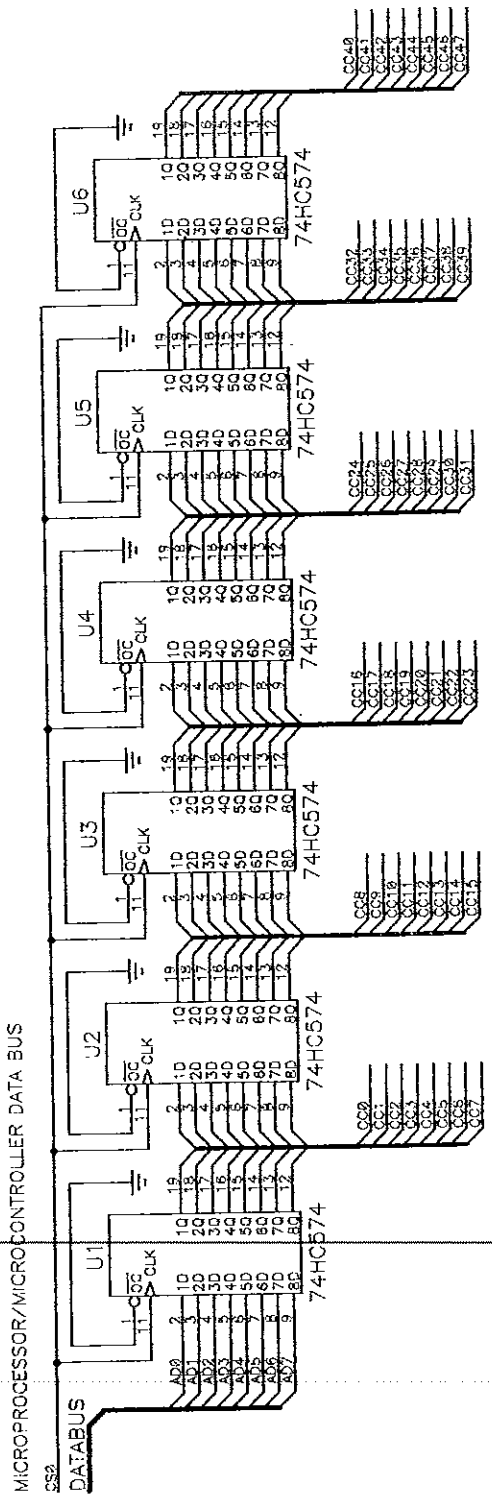
รูปที่ 3.7 รูปแบบการถอดรหัสบัสแอดเดรสเพื่ออ้างอิงพอร์ทโดยตรง

ในระบบแสดงผล ได้ใช้วงจรชิพรีจิสเตอร์เอาท์พุทแบบขนาน เพื่อส่งสัญญาณควบคุมภาคแสดงผลแบบดอทเมทริกซ์ทางด้านคอลัมน์ โดยขยายระบบพอร์ทเป็น 48 บิท (6 เอาท์พุทพอร์ท) ในการอ้างอิงแอดเดรส เพื่อทำการติดต่อกับกลุ่มของเอาท์พอร์ทนั้น ใช้แอดเดรสเพียงค่าเดียวจำนวนครั้งของการส่งข้อมูลออกพอร์ท (จำนวนครั้งของการเรียกใช้เอาท์พุทตำแหน่งนั้น) จะเป็นตัวกำหนดตำแหน่งของข้อมูลเช่น ต้องการให้พอร์ทตัวที่ 3 ของวงจรชิพรีจิสเตอร์แบบขนาน จากนั้นจะเป็นการเคลียร์สภาวะเอาท์พุทของพอร์ททุกตัวโดยการส่งค่า 00H ออกทางพอร์ทที่กำหนดไว้แล้ว 8 ครั้ง และตามด้วยการส่งค่า 7FH และส่งค่า 00H (หรือค่าข้อมูลใด ๆ) อีก 2 ครั้ง ค่า 7FH ก็จะปรากฏที่เอาท์พุทของพอร์ทตัวที่ 3 ตามต้องการในรูปแบบการทำงานเช่นนี้ สัญญาณที่เป็นตัวกำหนดจำนวนครั้งของการเลื่อน

(SHIFT) ข้อมูลคือ สัญญาณที่ได้จากการถอดรหัสแอดเดรสบัส เพื่ออ้างอิงพอร์ทของชิพรีจิสเตอร์แบบขนานนั่นเอง



รูปที่ 3.8 รูปแบบของวงจรชิพรีจิสเตอร์เอาท์พุทแบบขนานขนาด 24 บิท



รูปที่ 3.9 วงจรชิฟรีจิสเตอร์แบบขนานขนาด 48 บิทที่ใช้ในการส่งข้อมูล
ให้กับภาคแสดงผลแบบดอทเมทริกซ์

จะเห็นได้ว่ารูปแบบของวงจรชิฟรีจิสเตอร์แบบขนาน ค่อนข้างง่ายต่อการทำความเข้าใจคือ ถ้าพิจารณาในบิทข้อมูลใด ๆ จะมีรูปแบบการต่อแบบเอาท์พุทของพอร์ทตัวหน้าจะเป็นอินพุทของพอร์ทตัวถัดไป และจะเป็นสัญญาณที่ควบคุมภาพแสดงผลแบบดอทเมทริกซ์ทางด้านคอลัมน์ด้วย(ใช้ชื่อบัสข้อมูลว่า COLUMNCONTROL) พอร์ทตัวแรกเท่านั้นที่เชื่อมโยงกับบัสข้อมูลของระบบไมโครคอนโทรลเลอร์สัญญาณที่ควบคุมการเลื่อนของข้อมูลคือสัญญาณ CSO ตามรูป ซึ่งเป็นสัญญาณที่ได้จากการถอดรหัสบัสข้อมูลเพื่อเป็นตัวแทนในการอ้างอิงตำแหน่งของวงจรชิฟรีจิสเตอร์แบบขนานได้ก็ คือ 74LS374 74HC374 และ 74HC574 ซึ่งเป็นชิพเอาท์พุทพอร์ทขนาน 8 บิท

2.3 เอาท์พุทพอร์ทที่ควบคุมแบบอนุกรม เป็นอีกรูปแบบของการขยายพอร์ทเพื่อเพิ่มจำนวนบิทของข้อมูล หลักการของการขยายพอร์ทแบบนี้ คือ แทนที่จะใช้บัสข้อมูลทั้ง 8 เส้นเพื่อต่อกับเอาท์พุทพอร์ทแบบขนาน ซึ่งเป็นวิธีการแบบตรงก็ใช้กลุ่มสัญญาณในการควบคุมเพียง 4 เส้น คือ CLOCK, DATA IN,STROBE และ OUTPUT ENABLE โครงสร้างภายในของวงจร จะอยู่ในรูปแบบของวงจรชิฟรีจิส-เตอร์แบบอนุกรม ซึ่งมีจำนวนชุดเท่ากับจำนวนบิทเอาท์พุทพอร์ทสัญญาณ CLOCK จะเป็นสัญญาณที่ควบคุมการเลื่อนข้อมูล DATA IN คือ ข้อมูลเข้าสู่วงจรชิฟรีจิสเตอร์ STROBE และ OUTPUT ENABLE เป็นสัญญาณที่ควบคุมการ LATCH ข้อมูลเมื่อชุดของข้อมูลที่ต้องการส่งออกครบตามที่ต้องการ และแสดงผลข้อมูลออกทางเอาท์พุทตามลำดับ

ข้อจำกัดของจำนวนบิทเอาท์พุทพอร์ทที่สามารถขยายได้ขึ้นอยู่กับโปรแกรมที่ใช้ในการควบคุม เนื่องจากโปรแกรมต้องทำหน้าที่ในการสร้างสัญญาณ CLOCK และเปลี่ยนรูปแบบของข้อมูลที่ต้องการส่งจากแบบขนาน 8 บิท ให้เป็นข้อมูลแบบอนุกรม ดังนั้นหากระบบต้องการพอร์ทจำนวนมากโปรแกรมควบคุมก็ต้องใช้เวลามากในการทำขั้นตอนดังกล่าว ชิพที่ออกมาเฉพาะเพื่อจัดการเอาท์พุทรูปแบบนี้ คือ ULN5832 ULN5833 ของบริษัท SPRAQUE โดยมีเอาท์พุทพอร์ท 32 บิท ที่สามารถทนต่อกระแสสูงสุด (PEAK) 150 มิลลิแอมป์ สามารถขยายจำนวนเอาท์พุทโดยต่อชิพเพิ่มเติมแบบขนาน ในกรณีที่งานต้องการออกแบบเอาท์พุทแบบนี้เองก็สามารถทำได้โดยการใช้ชิพมาตรฐาน 74595 (TTL) หรือ 4094 (CMOS)

3. วงจรควบคุมการแสดงผล

ตามหลักการของระบบแสดงผล และรูปแบบของเอาท์พุทพอร์ทที่ควบคุมการแสดงผลที่ได้กล่าวมาแล้วนั้น ทำให้สามารถออกแบบวงจรควบคุมการแสดงผลทั้งหมดภายใต้การควบคุมของไมโครคอนโทรลเลอร์ 8031

ระบบแสดงผลได้ใช้ LED แบบดอทเมทริกซ์ขนาด 5 คอลัมน์ 7 แถว (5X7) จำนวน 3 บรรทัด แต่ละบรรทัดจะมี LED แสดงผลจะมีจำนวน 50 คอลัมน์และ 21 แถว ใช้หลักการแสดงผลแบบ TIME MULTIPLEXING สแกนข้อมูลทางด้านแถว และส่งข้อมูลเพื่อแสดงผลทางด้านคอลัมน์ เพื่อให้จำนวน LED ทางด้านคอลัมน์ลงตัวกับจำนวนบิทของเอาท์พุทพอร์ทที่ใช้ (6 พอร์ท=48 บิท) จึงจำเป็นต้องปล่อยให้ 2 คอลัมน์ของแผง LED เป็นส่วนที่ไม่ใช้งาน ดังนั้นโดยรูปแบบของการสแกนจะเหลือเป็น 48 คอลัมน์ X 21 แถว สวิทชิ่งเพาเวอร์ทรานซิสเตอร์ที่ควบคุมทางด้านแถวจำเป็นต้องจ่ายกระแสได้มากกว่า I_{max} ซึ่งเท่ากับ

$$I_{max} = NI_{led}$$

N : จำนวน LED ในแต่ละแถวกรณีนี้เท่ากับ 48 ตัว

I_{led} : กระแสที่ไหลผ่าน LED ต่อ 1 ดวง

เมื่อกำหนดให้กระแสที่ไหลผ่าน LED แต่ละตัว 40 มิลลิแอมป์ ซึ่งมากกว่าเงื่อนไขกำหนด(15 มิลลิแอมป์) สามารถทำได้ในกรณีนี้เนื่องจากเทคนิคการแสดงผลแบบ TIME MULTIPLEXING การติดดับของ LED จะเป็นช่วง ๆ ช่วงเวลาขึ้นอยู่กับสัญญาณที่ใช้ควบคุมการสแกน ดังนั้นเวลาในการติดของ LED สามารถคำนวณได้จากค่า DUTY CYCLE ของสัญญาณควบคุมทางด้านแถวและหาค่าเฉลี่ย ซึ่งยังคงน้อยกว่าเงื่อนไขที่กำหนด

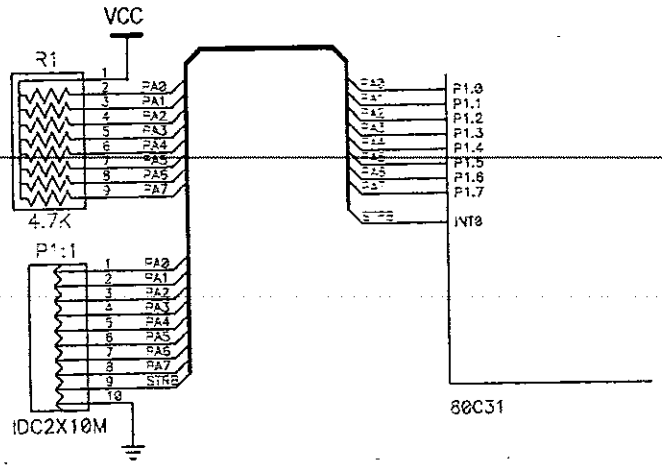
จากค่ากระแสสูงสุดที่ได้จากการคำนวณ ทำให้สามารถเลือกเบอร์ของสวิทชิ่งเพาเวอร์ทรานซิสเตอร์ที่ได้ใช้ในงานวิทยานิพนธ์ได้เลือกใช้สวิทชิ่งเพาเวอร์ทรานซิสเตอร์แบบ PNP(เนื่องจากใช้ LED แบบคอมมอนอาโนด) เบอร์ 2SA715 โครงสร้างภายในเป็นวงจรทรานซิสเตอร์แบบดาร์ลิงตัน(DARLINGTON) สามารถทนกระแสทางด้านคอนเลคเตอร์ต่อเนื่องได้ถึง 6 แอมแปร์ซึ่งมากกว่าความต้องการของวงจร

การควบคุมการทำงานของสวิทชิ่งเพาเวอร์ทรานซิสเตอร์ (สวิทชิ่งเพาเวอร์ทรานซิสเตอร์ที่ควบคุมด้านแถวคือ กลุ่ม ENABLE CONTROL POWER TRANSISTOR ตามรูปที่ 3.10) กระทำโดยพอร์ท 74HC574 จำนวน 3 ตัว ผ่านตัวต้านทานจำกัดกระแส (ตามรูปที่ 3.10 ชิพที่ควบคุมสวิทชิ่งเพาเวอร์ทรานซิสเตอร์คือ U6,U7 และ U8) ภายใต้ระบบควบคุมหลักของไมโครคอนโทรลเลอร์ 8031

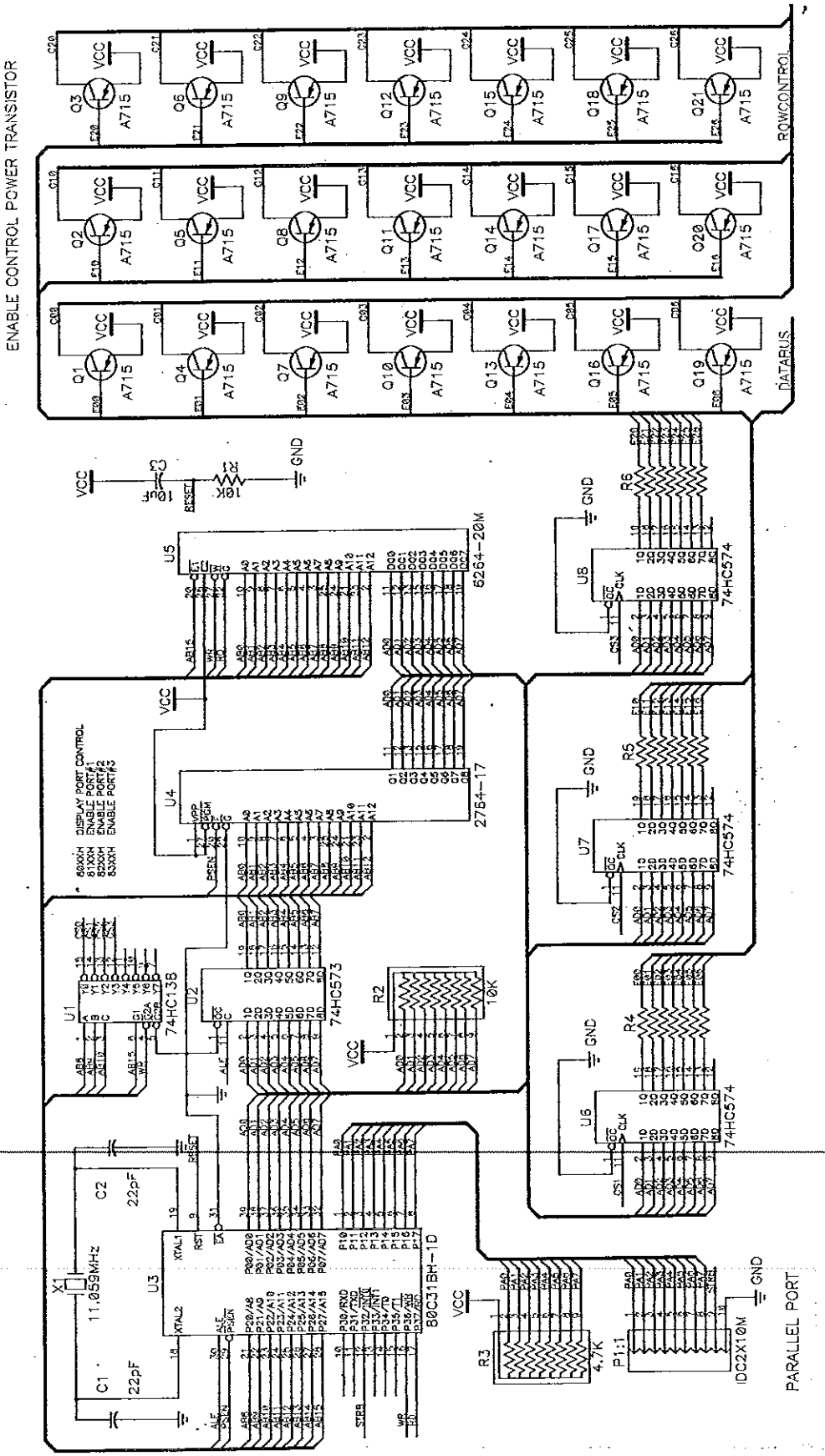
จากการที่ LED แบบดอทเมทริกซ์ที่เลือกใช้มีขนาดเล็ก ความสูงโดยรวมเป็น 0.8 นิ้ว และกระแสที่ใช้ในการขับให้สว่างเป็น 40 มิลลิแอมป์ เมื่อหาค่าเฉลี่ยโดยใช้ค่า DUTY CYCLE แล้ว ไม่จำเป็นต้องใช้วงจรขยายกระแส สามารถต่อเอาท์พุทจากพอร์ทผ่านตัวต้านทานจำกัดกระแสเข้าสู่ LEDได้เลย

4. วงจรส่วนรับข้อมูลเพื่อแสดงผล

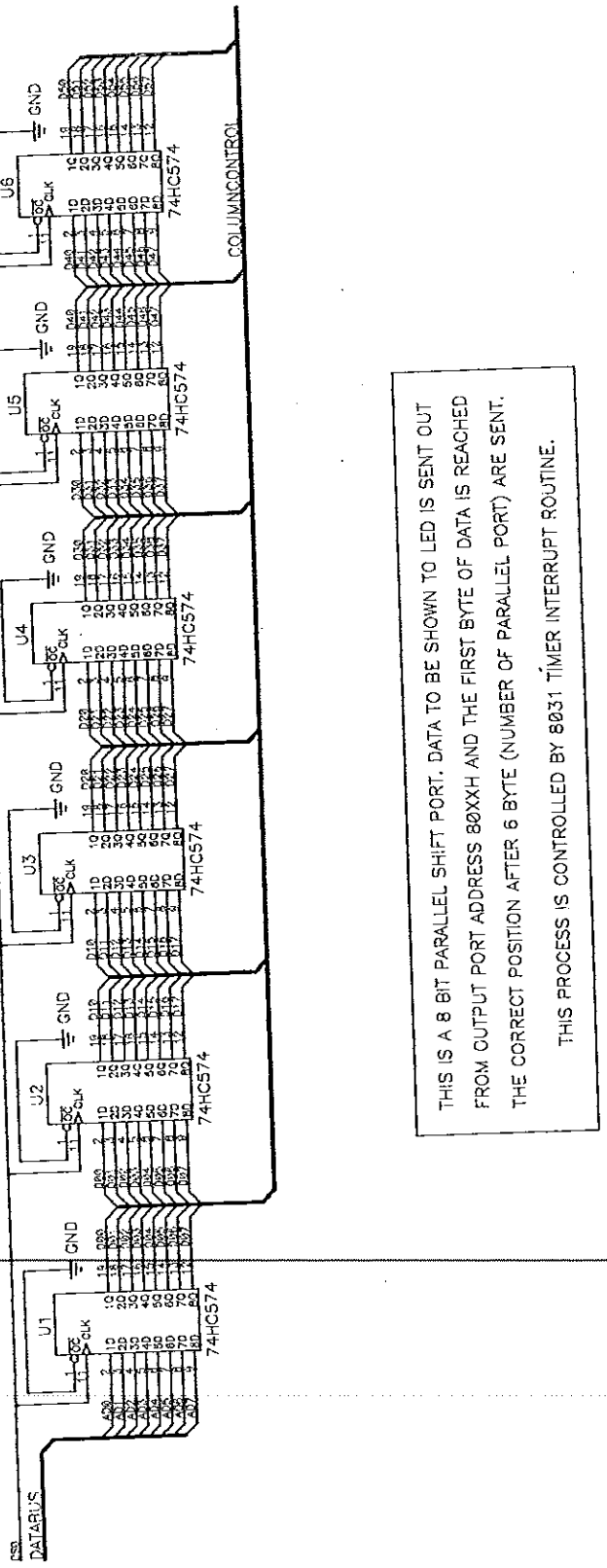
การรับข้อมูลของระบบแสดงผล จะกระทำโดยผ่านพอร์ทเชื่อมโยงแบบขนานขนาด 8 บิทมาตรฐานเช่นเดียวกับของ PRINTER PORT โดยมีบัสข้อมูลขนาดความกว้าง 8 บิท รูปแบบของอินพุทพอร์ท เป็นบัสของไมโครคอนโทรลเลอร์ 8031 หมายเลข P1 ซึ่งถูก PULL-UP ด้วยตัวต้านทาน 4.7 กิโลโอห์ม สัญญาณบ่งบอกการส่งข้อมูล(STROBE) เป็นสัญญาณกระตุ้นที่ระดับลอจิกต่ำที่ระบบภายนอกแจ้งให้ระบบแสดงผลทราบเมื่อต้องการส่งข้อมูล เนื่องจากโครงสร้างโปรแกรมส่วนรับข้อมูลที่ต้องการแสดงผลเป็นแบบอินเตอร์รัพท์ ดังนั้น สัญญาณนี้จึงต่อเข้ากับขารับอินเตอร์รัพท์ภายนอกหมายเลข 0 (INT0) ของไมโครคอนโทรลเลอร์ 8031



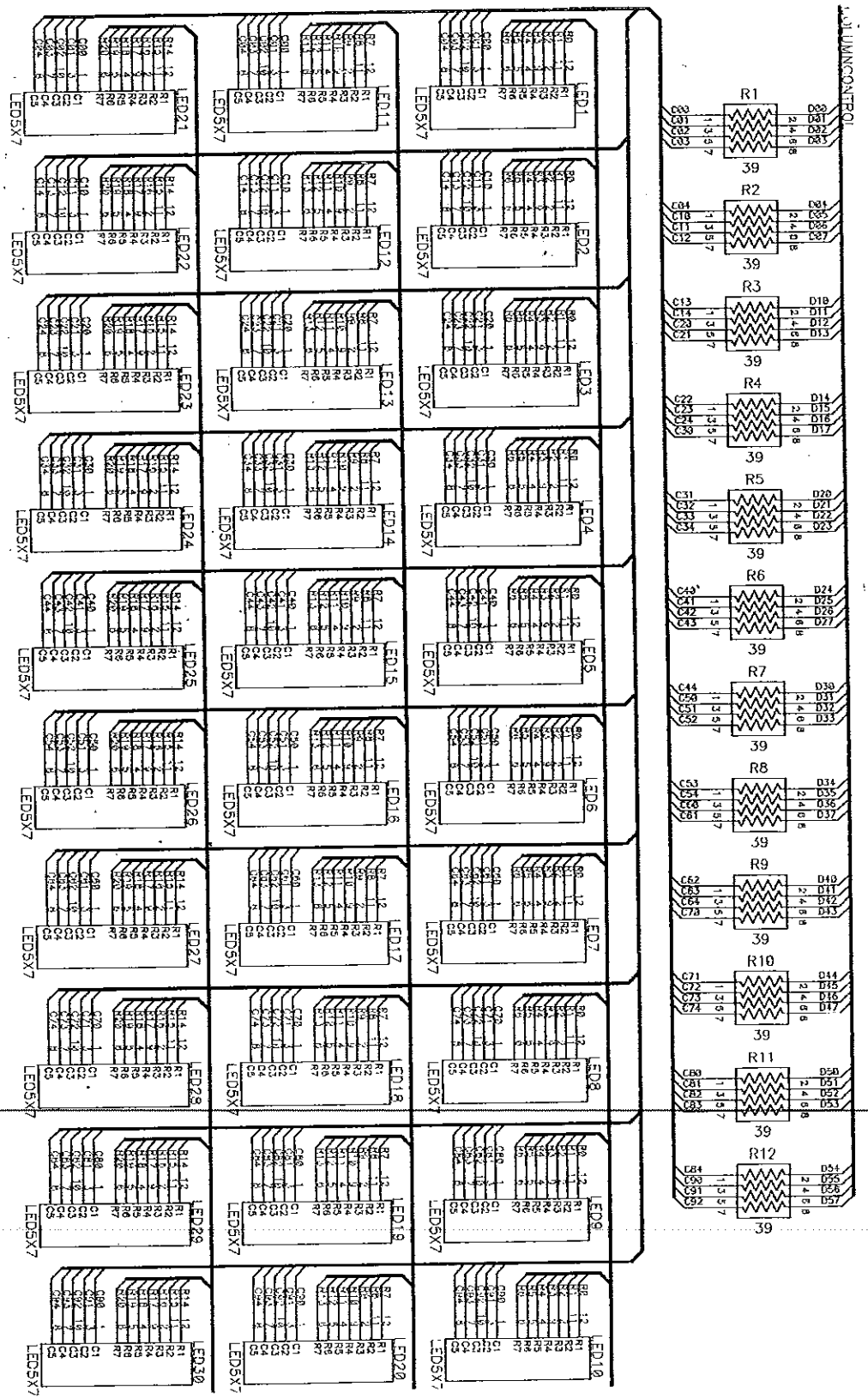
รูปที่ 3.10 วงจรส่วนรับข้อมูลเพื่อแสดงผลของระบบแสดงผล



รูปที่ 3.11 วงจรควบคุมโดยสมบูรณ์ของระบบแสดงผล



รูปที่ 3.12 วงจรควบคุมโดยสมบูรณ์ของระบบแสดงผล



รูปที่ 3.13 วงจรควบคุมโดยสมบูรณ์ของระบบแสดงผล

ส่วนขยายหน่วยความจำ

ในการเก็บข้อมูลที่จำเป็นต้องใช้ความละเอียดของชุดข้อมูลมาก หรือในงานวิจัยที่จำเป็นใช้เวลาในการติดตามความเปลี่ยนแปลงของข้อมูล จำเป็นต้องใช้หน่วยความจำของระบบจำนวนมากซึ่งความสามารถสูงสุดของไมโครคอนโทรลเลอร์ 8031 ที่สามารถอ้างอิงหน่วยความจำส่วนข้อมูลได้ก็ คือ 64 กิโลไบท์ ซึ่งในบางกรณีไม่เพียงพอต่อความต้องการในการใช้งาน ดังนั้นจึงจำเป็นต้องมีการขยายหน่วยความจำส่วนเก็บข้อมูลโดยรูปแบบของการขยายที่สามารถเป็นไปได้มี 2 รูปแบบ คือ

1. **การบริหารหน่วยความจำส่วนเพิ่มเติมสำหรับระบบไมโครคอนโทรลเลอร์ 8031**

ในการอ้างอิง หน่วยความจำส่วนข้อมูลของไมโครคอนโทรลเลอร์ 8031 มีแนวทางที่สามารถเพิ่มหน่วยความจำส่วนข้อมูลได้ โดยการถอดรหัสพอร์ทหมายเลข 1 (หรือพอร์ทที่ว่างจากการใช้งาน ซึ่งเป็นพอร์ทของไมโครคอนโทรลเลอร์เอง) เพื่อกำหนดสัญญาณเลือก BANK ของหน่วยความจำส่วนที่เกินไปจาก 64 กิโลไบท์ เช่น ในกรณีที่ต้องการเพิ่มหน่วยความจำเป็น 256 กิโลไบท์ โดยมีข้อกำหนดให้เลือกใช้หน่วยความจำมาตรฐานขนาด 32 กิโลไบท์ จำนวน 8 ตัว สามารถใช้พอร์ทหมายเลข 1 ในการถอดรหัสเพื่อเป็นตัวเลือก BANK 0 ขนาด 32 กิโลไบท์ จำนวน 8 ชุด และใช้สัญญาณในการจัดการเพื่อเลือกบัสข้อมูล และบัสอ้างอิงตำแหน่งข้อมูลเพื่อสลับการเรียกใช้หน่วยความจำส่วนขยายและหน่วยความจำหลักของระบบ ภายใต้การควบคุมของโปรแกรม โดยส่วนโปรแกรมควบคุมการทำงาน จะต้องมีการตรวจสอบการเพิ่มของตัวแปรที่ใช้อ้างอิงตำแหน่งของข้อมูล (DATA POINTER) การเพิ่มเติมหน่วยความจำแบบนี้ในการเลือกติดตั้งให้กับระบบควบคุมการทำงาน โดยวงจรที่ได้ออกแบบเพื่อเชื่อมโยงระบบไมโครคอนโทรลเลอร์ 8031 แสดงในรูปที่ 3.14

พิจารณาจากวงจรที่ได้ออกแบบขึ้น จะเห็นได้ว่าข้อมูลและบัสอ้างอิงตำแหน่งข้อมูลจะถูกต่อเข้ากับหน่วยความจำส่วนขยายขนาด 32 กิโลไบท์ (RAM เบอร์ 62256) ทุกตัว สำหรับบัสข้อมูลส่วน 8 บิทล่าง จะถูกแยกสัญญาณเป็นส่วนสัญญาณของบัสข้อมูลเองและส่วนบัสอ้างอิงตำแหน่งข้อมูล โดยการควบคุมของสัญญาณ

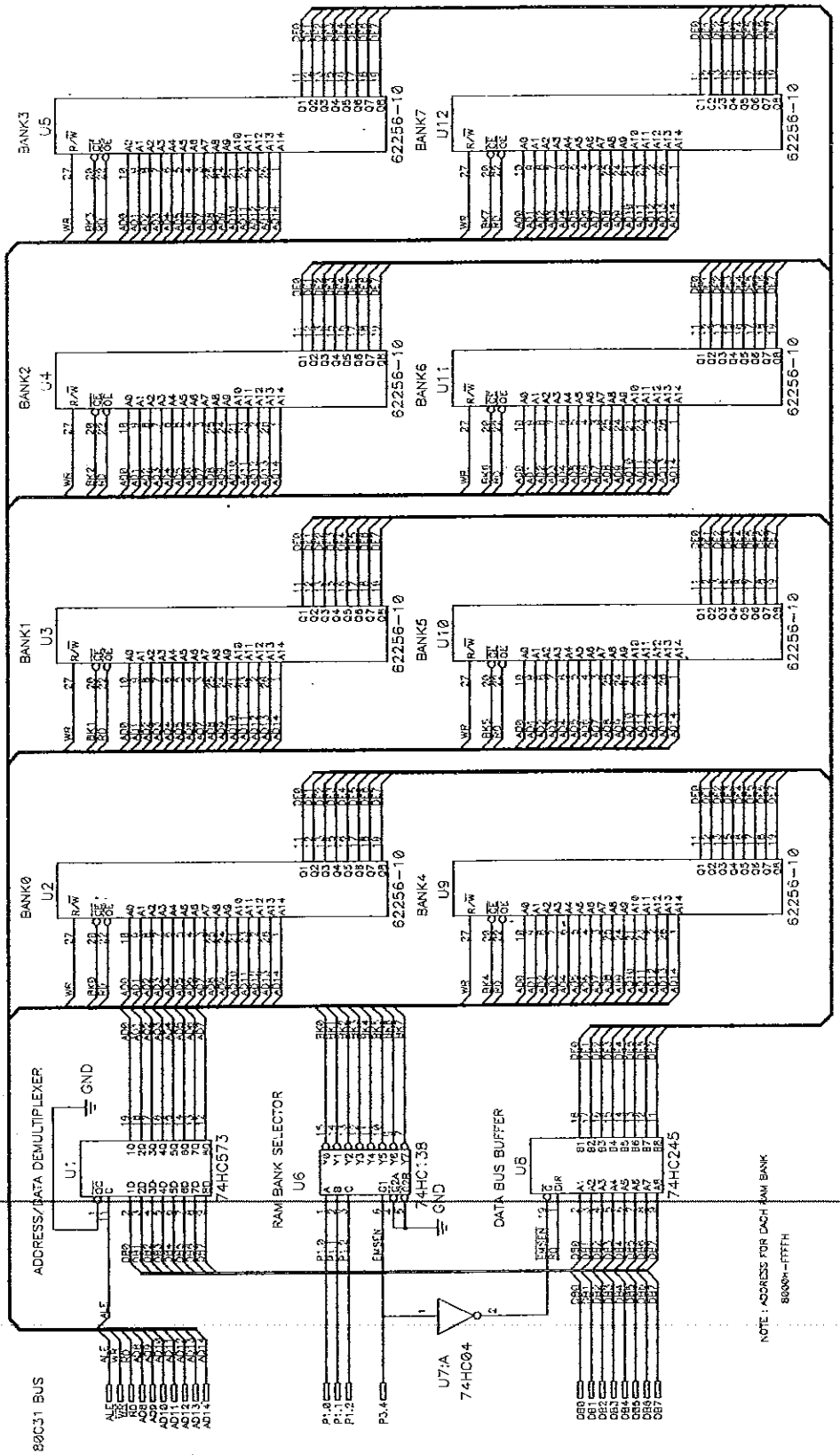
ALE(ADDRESS LATCH ENABLE) ของไมโครคอนโทรลเลอร์เองและเพื่อไม่ให้บัสข้อมูลของระบบบริหารหน่วยความจำในส่วนทั้งสองถูกเรียกใช้พร้อมกันโดยบังเอิญจึงใช้บัฟเฟอร์บัสข้อมูลเบอร์ 74HC245 ภายใต้การควบคุมของขา P3.4

ระบบบริหารหน่วยความจำส่วนขยายจะเชื่อมต่อกับบัสข้อมูล ของไมโครคอนโทรลเลอร์ก็ต่อเมื่อขา P3.4 ซึ่งต่อไปจะเรียกว่าขาสัญญาณ EMSEN (EXTENDED MEMORY SYSTEM ENABLE) มีสภาวะลอจิกสูง ขาสัญญาณ EMSEN นอกจากจะเชื่อมต่อระบบหน่วยความจำส่วนขยายเข้ากับบัส ของไมโครคอนโทรลเลอร์ ยังตัดการเชื่อมโยงระหว่างบัสของไมโครคอนโทรลเลอร์กับหน่วยความจำส่วนปกติ และยังเปิดโอกาสให้ชิพ 74HC138 ของระบบหน่วยความจำส่วนขยายทำการเลือก BANK ของหน่วยความจำตามค่าของพอร์ท P1 บิทที่ 1-3

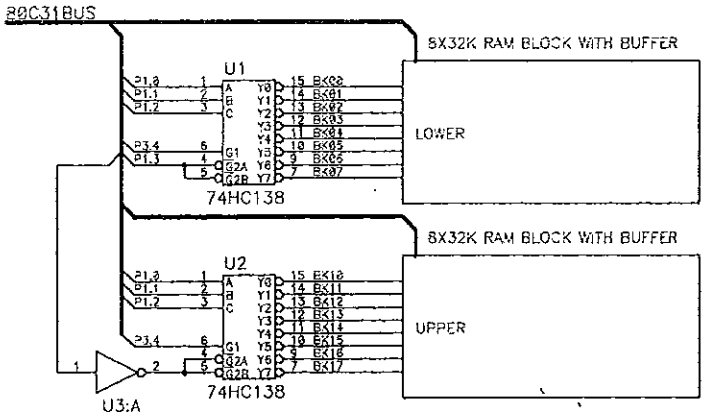
ดังนั้น โดยสรุปแล้วในการเขียน หรืออ่านหน่วยความจำส่วนขยาย โปรแกรมควบคุมการทำงานจะต้องทำให้สภาวะของบิทที่ขาสัญญาณ EMSEN เป็นลอจิกสูงก่อนที่จะทำการส่งค่าตัวเลือกหมายเลข BANK ไปยังพอร์ทหมายเลข 1 และกำหนดตำแหน่งบัสข้อมูลที่ต้องการอ่านหรือเขียนแก่บัสอ้างอิงตำแหน่งข้อมูล จากนั้นก็จะเข้าสู่ขั้นตอนอ่าน/เขียนตามที่ต้องการและในการกลับเข้าสู่การติดต่อกับหน่วยความจำส่วนปกติ โปรแกรมควบคุมก็เพียงแต่ทำให้สภาวะของ EMSEN เป็นลอจิกต่ำ

โดยการเลียนแบบรูปแบบการบริหารหน่วยความจำส่วนขยายที่ได้ออกแบบขึ้นนี้ ทำให้สามารถขยายหน่วยความจำส่วนข้อมูลออกได้อีกมากมายเพียงแต่ถอดรหัสพอร์ท P1 ส่วนที่เหลือมาใช้งาน

รูปที่ 3.15 จะเป็นรูปแบบการถอดรหัสการบริหารหน่วยความจำขนาด 512 กิโลไบท์ โดยแสดงเฉพาะชิพ 74HC138 ที่เชื่อมต่อกับพอร์ท P1 และกลุ่มของหน่วยความจำขนาด 32 กิโลไบท์จำนวน 8 BANK 2 ชุด



รูปที่ 3.14 วงจรของการบริหารหน่วยความจำส่วนขยายขนาด 128 กิโลไบท์ ของไมโครคอนโทรลเลอร์ 8031 ที่ใช้ในงานวิทยานิพนธ์



รูปที่ 3.15 แสดงการถอดรหัสพอร์ท P1 เพื่อเพิ่มหน่วยความจำส่วนขยาย ของรูปที่ 3.14 เป็น 2 เท่า

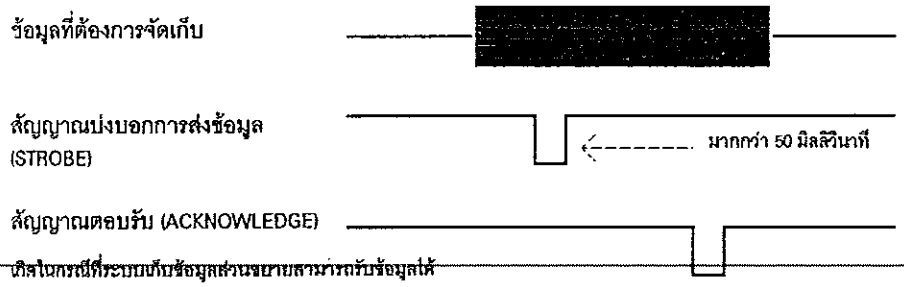
2. **การบริหารหน่วยความจำส่วนเพิ่มเติมกรณีหลายไมโครคอนโทรลเลอร์**

ในกรณีที่ 6.1 เป็นการบริหารหน่วยความจำสำหรับระบบที่ออกแบบเพื่อทำงานโดยเฉพาะแต่ในกรณีที่ต้องการทำให้ระบบเป็นมาตรฐาน เพื่อเชื่อมโยงกับระบบภายนอกใด ๆ ได้ จำเป็นต้องใช้รูปแบบการจัดการแบบหลายไมโครคอน-โทรลเลอร์ เพื่อให้ขีดความสามารถในการขยายไม่จำกัดและสามารถเลือกเพิ่มเติมหน่วยความจำให้เหมาะสมกับการใช้งานได้

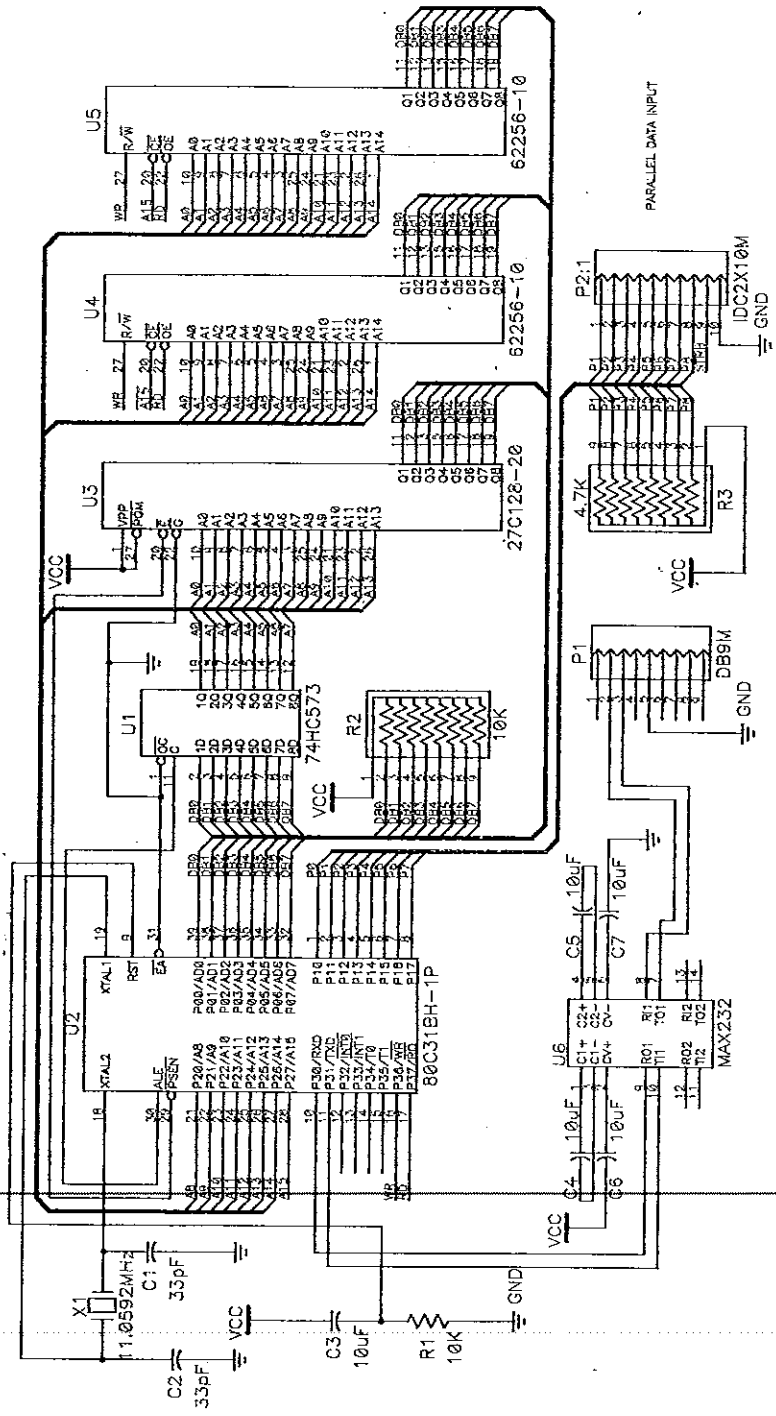
ในกรณีของการบริหารหน่วยความจำในรูปแบบนี้ จำเป็นต้องมีส่วนฮาร์ดแวร์เป็นระบบอิสระภายใต้การควบคุมโดยไมโครคอนโทรลเลอร์ 8031 ติดต่อกับหน่วยความจำขนาด 64 กิโลไบท์ ซึ่งเป็นขนาดสูงสุดของหน่วยความจำภายนอกที่

ไมโครคอนโทรลเลอร์ 8031 สามารถบริหารได้โดยวิธีปกติการติดต่อระหว่างระบบภายนอกที่เป็นตัวจัดการส่งข้อมูลเข้าสู่ระบบเก็บข้อมูล ผ่านทางพอร์ทเชื่อมรียงแบบขนานมาตรฐานของ PRINTER PORT ขนาดความกว้างของบัสข้อมูล 8 บิท โดยใช้ PORT P1 ของไมโครคอนโทรลเลอร์เป็นตัวจัดการภาครับข้อมูล สัญญาณบ่งบอกการส่งข้อมูล (STROBE) กระตุ้นที่ระดับลอจิกต่ำเพื่อบ่งบอกสภาวะการส่งข้อมูล เพื่อจัดการเก็บในรูปแบบของการอินเทอร์รัพท์จากภายนอกผ่านทางขา INTO ของไมโครคอนโทรลเลอร์นอกจากนี้ยังมีสัญญาณตอบรับ (ACKNOWLEDGE) จะทำงานที่ระดับลอจิกต่ำ เพื่อบ่งบอกในกรณีที่ระบบเก็บข้อมูลส่วนขยายรับข้อมูลแล้ว หรือยังคงอยู่ในสภาวะลอจิกสูงในกรณีที่ข้อมูลเต็มหน่วยความจำ และในกรณีที่ผู้ใช้งานต้องการอ่านข้อมูลจากระบบเก็บข้อมูลส่วนขยายก็สามารถกระทำได้โดยการส่งรหัสคำสั่งผ่านทางพอร์ทเชื่อมรียงแบบอนุกรมและสัญญาณส่งออกก็จะกระทำผ่านทางพอร์ทนี้เช่นเดียวกัน

สำหรับรูปแบบของสัญญาณในการเชื่อมรียง และวงจรของระบบบริหารหน่วยความจำแบบอิสระแสดงในรูปที่ 3.16 และ 3.17 ตามลำดับ



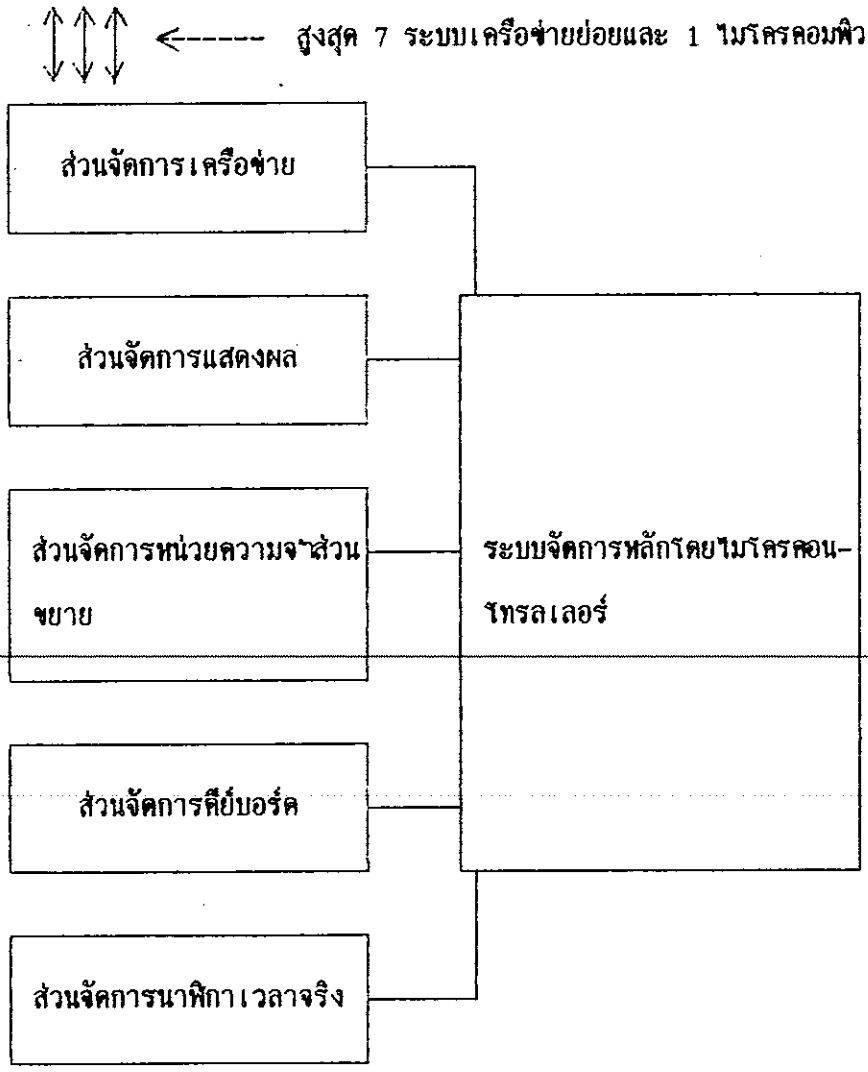
รูปที่ 3.16 สภาวะสัญญาณในการเชื่อมรียงระหว่างระบบภายนอกกับระบบเก็บข้อมูลส่วนขยายแบบอิสระ



รูปที่ 3.17 วงจรควบคุมของระบบเก็บข้อมูลส่วนขยายโดยใช้ไมโครคอนโทรลเลอร์อิสระ

ระบบควบคุมการทำงานหลัก

ระบบควบคุมการทำงานหลัก จัดเป็นหัวใจของระบบเก็บข้อมูลระยะไกลที่พัฒนาขึ้น จุดประสงค์ของการออกแบบระบบควบคุมการทำงานหลัก ก็เพื่อให้ระบบเก็บข้อมูลระยะไกลสามารถทำงานได้โดยอิสระ เพื่ออำนวยความสะดวกในการนำระบบไปใช้ในงานภาคสนามโดยมีส่วนรับคำสั่งงานและตัวแปรที่ใช้ในการทำงานผ่านทางคีย์บอร์ด และนอกจากนั้นระบบควบคุมการทำงานหลักยังคงสามารถรับคำสั่งงานและตัวแปรจากไมโครคอมพิวเตอร์ผ่านทางพอร์ทเชื่อมโยงแบบอนุกรม ภายใต้ข้อกำหนดของโปรโตคอลซึ่งจะกล่าวในรายละเอียดต่อไป สำหรับแผนผังของข่ายงานของระบบควบคุมการทำงานหลักแสดงในรูปที่ 3.18



รูปที่ 3.18 แผนผังการทำงานของระบบควบคุมการทำงานหลัก

ในระบบการทำงานหลักจะประกอบด้วยส่วนเพิ่มเติมย่อยที่มีความสำคัญต่อการทำงานของระบบเก็บข้อมูลระยะไกล 5 ส่วน ภายใต้การควบคุมด้วยไมโครคอนโทรลเลอร์ 8031 1 ตัว คือ

1. **ส่วนจัดการเครือข่าย**

จากรูปแบบของระบบเก็บข้อมูลระยะไกล โดยใช้ไมโครโปรเซสเซอร์ซึ่งมีโครงสร้างประกอบด้วย ระบบเครือข่ายที่ประจำในจุดสำรวจ หรือจุดที่ต้องการอ่านค่าข้อมูลการทำงานเป็นแบบอิสระที่ควบคุม โดยไมโครคอนโทรลเลอร์ของตัวเอง ดังนั้นการติดต่อเพื่อรับหรือส่งข้อมูลหรือคำสั่งควบคุมการทำงานระหว่างระบบเครือข่ายย่อยกับระบบควบคุมการทำงานหลัก จะต้องกระทำผ่านทางเครือข่าย (NETWORK) เครือข่ายที่เหมาะสมและเลือกใช้ คือ ระบบเครือข่ายแบบ STAR ซึ่งรายละเอียดได้กล่าวไปแล้วในบทที่ 2

จากการที่ระบบเครือข่ายที่ได้เลือกใช้ จำเป็นต้องออกแบบวงจรบริการเครือข่าย เนื่องจากรูปแบบการรับหรือส่งข้อมูลกระทำผ่านตัวกลางของการสื่อสารข้อมูลแบบอนุกรมมาตรฐาน RS-232 หรือ RS-422 ซึ่งมีข้อดีคือมีสายสัญญาณควบคุมที่จำเป็นมีเพียง 2 เส้น คือ สัญญาณส่งข้อมูล (TXD) และสัญญาณรับข้อมูล (RXD) โดยมีรายละเอียดย่อยตามมาตรฐานที่เลือกใช้คือ

- RS-232 ใช้การเปรียบเทียบระดับความสูงของสัญญาณรับ และส่งกับ GROUND ของระบบ

- RS-422 ใช้รูปแบบการแปลงสัญญาณทั้งสองให้เป็นรูปแบบของสัญญาณเอาท์พุทแบบกลับขั้วสัญญาณ(INVERTING OUTPUT) และสัญญาณเอาท์พุทแบบไม่กลับขั้วสัญญาณ (NON-INVERTING OUTPUT) โดยเมื่อสัญญาณทั้งสองเข้าสู่ภาครับจะมีการเปรียบเทียบ และจัดกระทำเพื่อให้ได้ข้อมูลจริงที่ต้องการสื่อสาร โดยมีอุปกรณ์ช่วยเหลือทั้งทางในการเปลี่ยนรูปแบบทางด้านอินพุทและเปลี่ยนรูปแบบกลับเป็นข้อมูลแบบอนุกรมทางด้านเอาท์พุท โดยไม่จำเป็นต้องอ้างอิงกับ GROUND ของระบบ

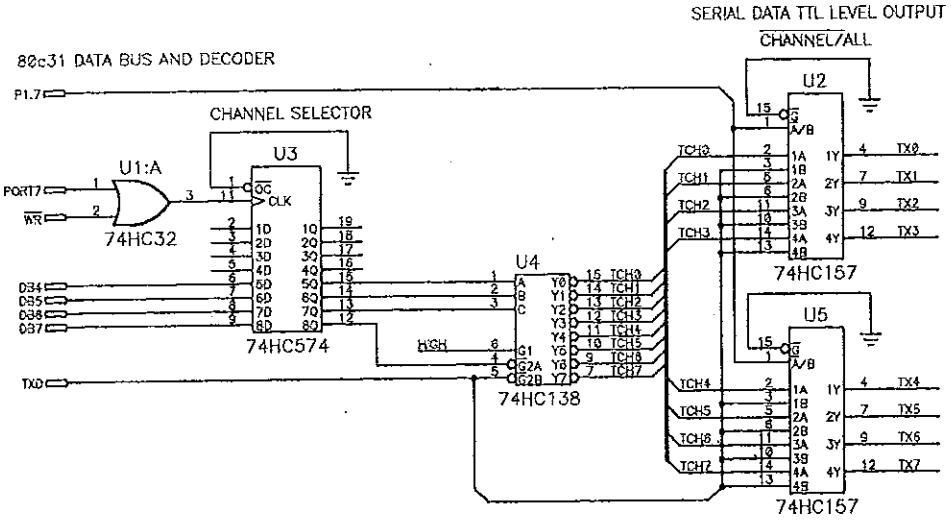
การออกแบบระบบบริการเครือข่ายส่วนฮาร์ดแวร์ เพื่อขยายจำนวนช่องการสื่อสารได้ใช้รูปแบบ MULTIPLEXER หรือ DEMULTIPLEXER สัญญาณการสื่อสารทั้ง 2 โดยใช้ชิพ DATA MULTIPLEXER หรือ DATA DEMULTIPLEXER

เป็นตัวช่วยจัดการ โดยที่

1.1 สัญญาณส่งข้อมูล (TXD)

พิจารณาจากรูปที่ 3.19 ซึ่งแสดงวงจรในภาคการส่งข้อมูลสัญญาณส่งข้อมูล (TXD) จากไมโครคอนโทรลเลอร์ ของระบบควบคุมการทำงานหลัก จะถูกป้อนเข้าสู่อินพุทของชิพ 74HC138 DATA DEMULTIPLEXER AND DECODER ทางขา ENABLE ในการเลือกติดต่อกับระบบเครือข่ายใด ๆ สามารถทำได้โดยการส่งค่าแทนหมายเลขของระบบเครือข่ายย่อย ในรูปแบบของรหัส BCD (BINARY CODED DECIMAL) ให้กับเอาท์พุทพอร์ท (ซึ่งในโครงสร้างทางฮาร์ดแวร์ตามรูปที่ 3.19 เอาท์พุทพอร์ทต่อเชื่อมโยงกับขาเลือกช่องเอาท์พุทของ 74HC138) ที่กำหนดการเลือกช่องเอาท์พุทของสัญญาณส่งข้อมูลก่อน ความเปลี่ยนแปลงของข้อมูลทางเอาท์พุทของชิพ 74HC138 ที่ช่องเอาท์พุทใด ๆ จะสอดคล้องกับสัญญาณที่ควบคุมการ ENABLE ชิพ โดยที่สถานะของเอาท์พุทของชิพ 74HC138 ในขณะที่ขา ENABLE อยู่ในสภาวะลอจิกสูงจะอยู่ในสภาวะลอจิกสูงทุกเอาท์พุท การใช้ชิพ 74HC138 1 ตัว สามารถบริการระบบเครือข่ายย่อยได้ 8 ตัว และสามารถขยายได้ตามต้องการ (ในต้นแบบของระบบเก็บข้อมูลระยะไกลสามารถใช้ระบบเครือข่ายย่อยได้ 8 ตัว)

เนื่องจากในการกระตุ้นการทำงานของระบบเครือข่ายย่อยซึ่งจำเป็นต้องส่งรหัสแอสกี (เพราะการกระตุ้นการทำงานของระบบเครือข่ายย่อย เพื่อให้มีการสุ่มอ่านข้อมูลกระทำโดยผ่านทางระบบเครือข่าย จึงไม่สามารถใช้การกระตุ้นโดยการเปลี่ยนสภาวะลอจิก ดังเช่นที่ใช้ในวงจรดิจิตอลโดยทั่วไปได้ จึงใช้การกระตุ้นโดยการส่งรหัสผ่านทางเครือข่ายแบบอนุกรม) ไปยังระบบเครือข่ายย่อยทุกตัวที่ต่อกับระบบเครือข่ายพร้อม ๆ กัน ดังนั้น สัญญาณส่งข้อมูลที่ไปยังทุก ๆ ระบบเครือข่ายย่อยจึงจำเป็นต้องเปิดทางเลือกให้กับสัญญาณส่งข้อมูลจากไมโครคอนโทรลเลอร์โดยตรงด้วย ซึ่งได้ออกแบบวงจรให้ไมโครคอนโทรลเลอร์หลักสามารถเลือกการส่งสัญญาณแบบแยกช่อง หรือส่งสัญญาณไปยังทุกช่องของระบบเครือข่าย โดยใช้ชิพ 74HC157 (4 CHANNELS 2 TO 1 DATA SELECTOR) คอยสลับรูปแบบตามคำสั่งงานของไมโครคอนโทรลเลอร์ผ่านบิทควบคุม P1.7



รูปที่ 3.19 วงจรจัดเลือกช่องสัญญาณส่งข้อมูล (TXD) ของการสื่อสารข้อมูลแบบอนุกรมแบบเลือกส่งข้อมูลออกทุกช่องสัญญาณ

1.2 **สัญญาณรับข้อมูล** (RXD)

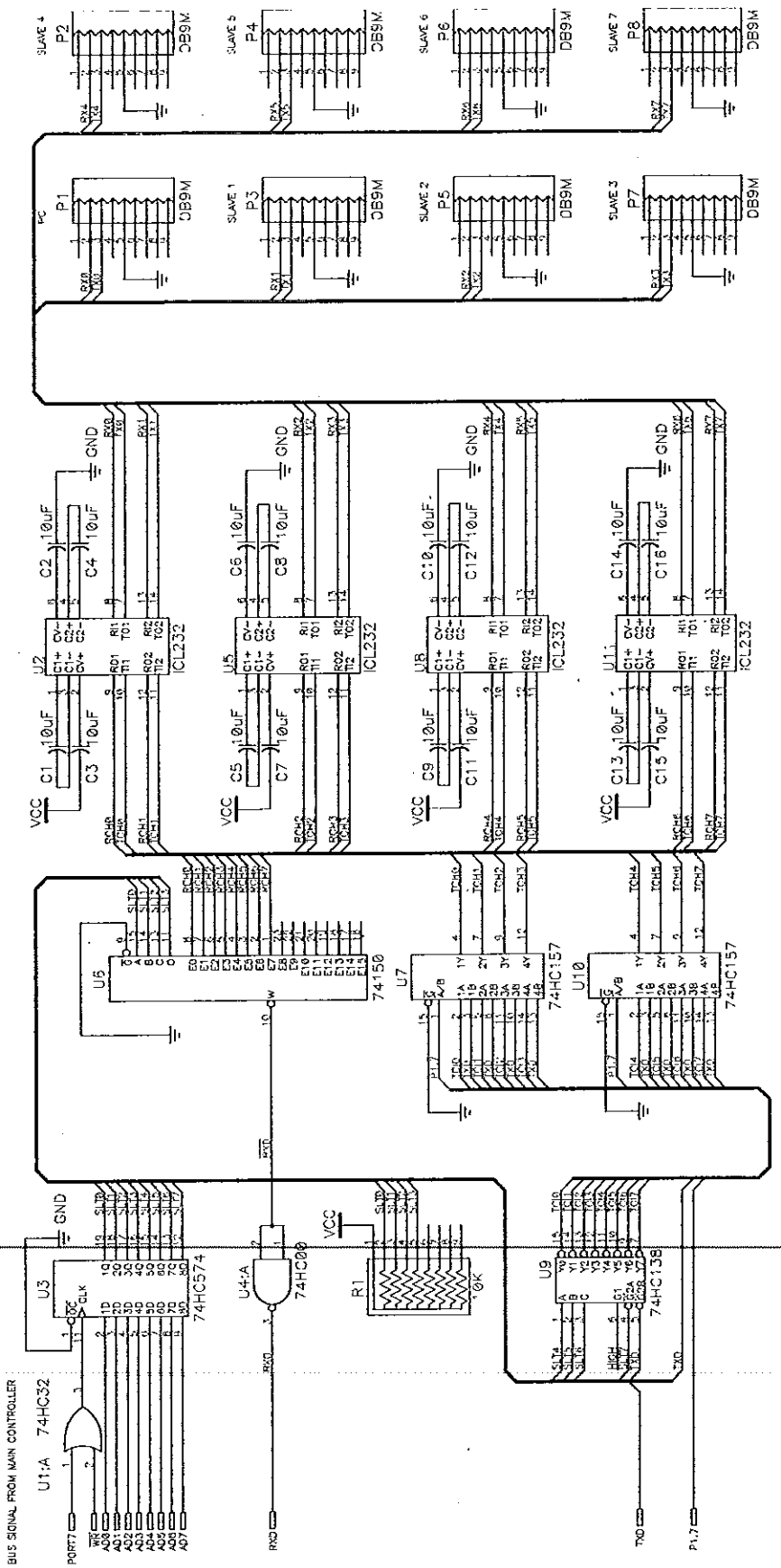
สัญญาณรับข้อมูลเข้าสู่ไมโครคอนโทรลเลอร์ ของระบบควบคุมการทำงานหลักจะรับสัญญาณส่งข้อมูลที่มาจากระบบเครือข่ายย่อย ที่ผ่านการแปลงรูปแบบสัญญาณจากมาตรฐาน RS-232 เป็นระดับลอจิกของ TTL เข้าสู่ชิพ 74150 (16 LINE-TO-1-LINE DATA SELECTOR/MULTIPLEXER)ซึ่งเป็นตัวเลือกช่องสัญญาณจากระบบเครือข่ายย่อยเข้าสู่ขารับสัญญาณของไมโครคอนโทรลเลอร์ ในการเลือกช่องสัญญาณจะอยู่ภายใต้การควบคุมของโปรแกรมงานผ่านเอาท์พุทพอร์ทเช่นเดียวกับสัญญาณส่งข้อมูลวงจรส่วนนี้ค่อนข้างง่ายในการออกแบบ จึงแสดงรวมเป็นวงจรส่วนบริหารเครือข่ายในรูปที่ 3.20

2. **ส่วนจัดการแสดงผล**

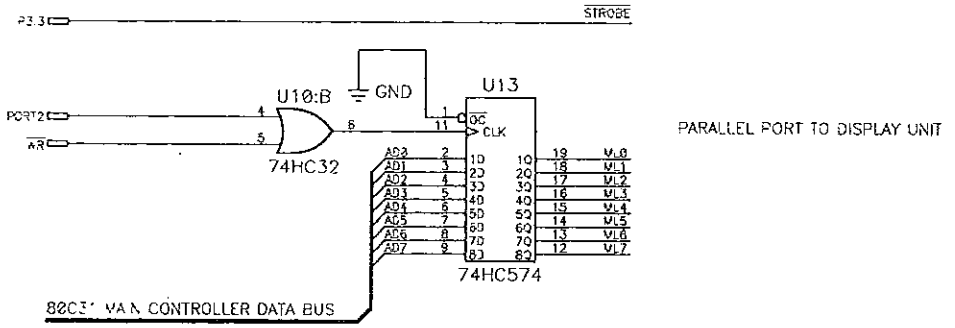
โครงสร้างของส่วนจัดการในส่วนของพอร์ท มีลักษณะเป็นเอาท์พุทพอร์ทแบบขนาน ขนาด 8 บิท มาตรฐานเอาท์พุทแบบ TTL (ระดับลอจิกสูงเป็น 5 โวลท์และระดับลอจิกต่ำเป็น 0 โวลท์) และมีความสามารถในการคงค่าข้อมูล (LATCH) ที่ต้องการส่งออกทางด้านเอาท์พุทได้ โดยได้เลือกใช้ชิพมาตรฐานแบบซีมอสความเร็วสูง(HIGH SPEED CMOS : HC)เบอร์ 74HC574(8 BIT LATCH) ซึ่งมีความเร็วในการทำงานเทียบเท่ากับชิพมาตรฐานแบบ TTL (74LS574 หรือ 74574) แต่มีข้อได้เปรียบ คือต้องการกระแสในการทำงานต่ำกว่ามากและสามารถเชื่อมโยงกับวงจรที่มีโครงสร้างแบบ TTL ได้

สำหรับกลุ่มสัญญาณ ที่เป็นตัวบ่งบอกการสภาวะการส่งข้อมูล (STROBE) และบิทตรวจสอบสัญญาณการตอบรับได้เลือกใช้บิทว่างจากพอร์ทของไมโครคอนโทรลเลอร์ ภายใต้การควบคุมของโปรแกรมควบคุมการทำงาน

วงจรในส่วนจัดการแสดงผลแสดงในรูปที่ 3.21



รูปที่ 3.20 วงจรส่วนจัดการเครือข่าย



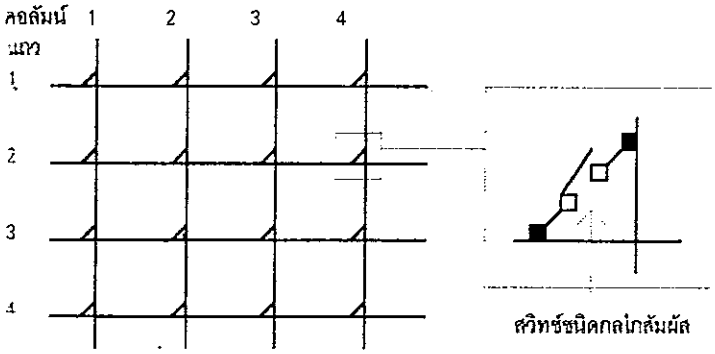
รูปที่ 3.21 วงจรส่วนจัดการแสดงผล

3. ส่วนจัดการคีย์บอร์ด

ส่วนจัดการนี้ทำหน้าที่ในการรับข้อมูล ตัวแปร หรือคำสั่งงานเข้าสู่ระบบควบคุมการทำงานหลัก โดยทั่วไปแล้ววงจรคีย์บอร์ดสำหรับระบบไมโครโปรเซสเซอร์จะอยู่ในรูปแบบของสวิทช์ชนิดกลไกสัมผัส ที่เชื่อมต่อกันในลักษณะเมทริกซ์ทางด้านแถวและคอลัมน์ เนื่องจากการขยายจำนวนคีย์บอร์ด เพื่อใช้งานในบางกรณีนั้นสามารถทำได้ง่าย และใช้อุปกรณ์เพิ่มเติมน้อยกว่าการใช้วงจรคีย์บอร์ดแบบอินพุทพอร์ทเชื่อมรียงกับสวิทช์กดแบบ 1 บิทต่อ 1 แป้นพิมพ์ เช่น ในกรณีที่ต้องการใช้งานคีย์บอร์ดจำนวน 64 ตัววงจรคีย์บอร์ดแบบอินพุทพอร์ทต้องการพอร์ทมากถึง 8 พอร์ท ในขณะที่วงจรคีย์บอร์ดแบบเมทริกซ์ต้องการพอร์ทขนาด 8 บิทเพียง 2 พอร์ทเท่านั้น

รูปแบบของวงจรคีย์บอร์ดแบบเมทริกซ์นั้น การกดสวิทช์ที่ตำแหน่งแถวและคอลัมน์มีผลต่อการแปรความหมายของตำแหน่งสวิทช์ที่แตกต่างกันโดยรูปแบบเมทริกซ์ของสวิทช์ชนิดกลไกสัมผัส ขนาด 16 ตัว แสดงในรูปที่ 3.22

ในการระบุความแตกต่างของตำแหน่งการกดสวิทช์ จะเชื่อมรียงแถวและคอลัมน์เข้ากับพอร์ทโดยที่กำหนดให้ด้านแถวเป็นส่วนรับข้อมูล และด้านคอลัมน์เป็นส่วนส่งสัญญาณกำหนดรูปแบบสแกนค่า ดังนั้น ด้านแถวจะเชื่อมรียงเข้ากับอินพุทพอร์ท 1 แถวต่อ 1 บิทข้อมูล ตำแหน่งการถอดรหัสพอร์ทในระบบควบคุมการทำงาน คือ E4XXH และด้านคอลัมน์จะถูกเชื่อมต่อเข้ากับเอาท์พุทพอร์ท 1 แถวต่อ 1 บิท ตำแหน่งการถอดรหัสพอร์ทในระบบควบคุมการทำงานคือ E5XXH และการแสกนจะส่งข้อมูลออกทางเอาท์พุทพอร์ทในระดับลอจิกสูงครั้งละ 1 บิท(รูปแบบของสัญญาณเอาท์พุทที่ใช้ในการสแกนทางด้านคอลัมน์ มีลักษณะเช่นเดียวกับเอาท์พุทของ RING COUNTER คือ จะมีบิทของเอาท์พุทใด ๆ ที่เป็น 0 ในช่วงเวลาใด ๆ เมื่อลอจิก 0 เป็นตัวกระตุ้นหรือเป็นลอจิก 1 เพียง 1 บิท ในกรณีที่ลอจิก 1 เป็นตัวกระตุ้น) ไมโครคอนโทรลเลอร์จะตรวจสอบทางด้านอินพุทพอร์ทว่ามีความเปลี่ยนแปลงสถานะจากกรณีที่ไม่มีการกดสวิทช์ใด ๆ หรือไม่(ในกรณีที่ไม่มีการกดสวิทช์ใด ๆ อินพุทพอร์ทจะอยู่ในสภาวะเปิดหรือระดับลอจิกสูงทั้งหมด) เมื่อมีความเปลี่ยนแปลงเกิดขึ้นก็จะนำค่าที่ส่งออกทางเอาท์พุทพอร์ทซึ่งกำหนดคอลัมน์ขณะที่ถูกกด และค่าที่อ่านได้จากทางอินพุทพอร์ทซึ่งกำหนดแถว เพื่อใช้ในการเปิดตารางแทนความหมายของการกดสวิทช์นั้น ๆ รูปแบบของการรับข้อมูลจากคีย์บอร์ด ในส่วนโปรแกรมควบคุมการทำงานของระบบควบคุมหลักที่ได้สร้างขึ้นนั้นจะเป็นแบบ POLLING คือ โปรแกรมควบคุมการทำงานจะเรียกส่งค่าสแกน และอ่านค่าก็ต่อเมื่อถึงเวลาที่ระบบต้องการติดต่อเท่านั้น ซึ่งในงานบางงานการรับข้อมูลต้องการอินเตอร์รัพท์จากผู้ใช้งาน ซึ่งจำเป็นต้องเพิ่มเติมฮาร์ดแวร์ในส่วนตรวจสอบการกดคีย์บอร์ด เพื่อสร้างสัญญาณบ่งบอกการขอบริการอินเตอร์รัพท์ซึ่งจะไม่ขอกล่าวถึงในที่นี้



รูปที่ 3.22 รูปแบบเมทริกซ์ของสวิทช์ชนิดกลไกสัมผัสชนิด 16 แป้นพิมพ์

4. ส่วนจัดการนาฬิกาเวลาจริงและสัญญาณในการสุ่มอ่านข้อมูล

ในบางรูปแบบการทำงานระบบควบคุมการทำงานหลัก จำเป็นต้องสร้างสัญญาณมาตรฐานกำหนดช่วงเวลาในการสุ่มอ่านข้อมูล (SAMPLING RATE) จากระบบเครือข่ายย่อยที่อยู่ในบริเวณที่มีการทดลองหรือจุดสำรวจเพื่อให้ข้อมูลที่ต้องการอ่านนั้นเป็นข้อมูลที่เกิดขึ้นในเวลาเดียวกัน ดังนั้น ฐานเวลาในระบบควบคุมการทำงานหลักจึงมีความสำคัญค่อนข้างมาก จึงจำเป็นต้องมีการเพิ่มเติมส่วนจัดการนาฬิกาเวลาจริง (REAL TIME CLOCK) และวงจรสร้างสัญญาณพื้นฐานในการสุ่มอ่านข้อมูล

จุดประสงค์ของการเพิ่มเติมส่วนจัดการนาฬิกาเวลาจริง โดยสามารถกำหนดเวลาเริ่มต้นและเวลาสิ้นสุดของกระบวนการ ก็สามารถใช้โปรแกรมจัดการร่วมกับส่วนจัดการนาฬิกาเวลาจริงได้

4.1 ส่วนจัดการนาฬิกาเวลาจริง

ส่วนจัดการนาฬิกาเวลาจริงเลือกใช้ชิพเบอร์ MSM6242B ของบริษัท FUJITSU ซึ่งมีความสามารถในการรายงานวัน เดือน ปี วัน ในรอบสัปดาห์ ชั่วโมง นาที และวินาที การติดต่อกับระบบควบคุมการทำงานหลักสามารถกระทำโดยผ่านบัสข้อมูลบัสแอดเดรสและใช้สัญญาณควบคุมของไมโครคอนโทรลเลอร์ 8031 โดยใช้บัสข้อมูลเพียง 4 เส้น การอ้างอิงตำแหน่งของชิพใช้การถอดรหัสโดยใช้ส่วน 8 บิทบนของบัสแอดเดรส ในระบบควบคุมหลักกำหนดให้ส่วนจัดการนาฬิกาเวลาจริงครอบครองตำแหน่งแอดเดรส E3-XH โดยที่ -XH จะแทนความหมายในการติดต่อกับรีจิสเตอร์ภายในชิพนาฬิกาจริง (ซึ่งจะถูกถอดรหัสภายในตัวชิพเอง) มีค่าอยู่ในช่วง 0 ถึง FH ทำให้ค่าไบท์บนของบัสแอดเดรสในส่วนต่ำไม่มีความหมายในการถอดรหัสค่าที่อ่านได้จากชิพนาฬิกาเวลาจริงจะมีความหมายเฉพาะส่วน 4 บิทล่างเท่านั้น ซึ่งสอดคล้องกับการต่อสัญญาณของบัสข้อมูลเฉพาะ 4 บิทล่างสำหรับรีจิสเตอร์ภายในของชิพนาฬิกาเวลาจริงและรายละเอียดแสดงในตารางที่ 3.1

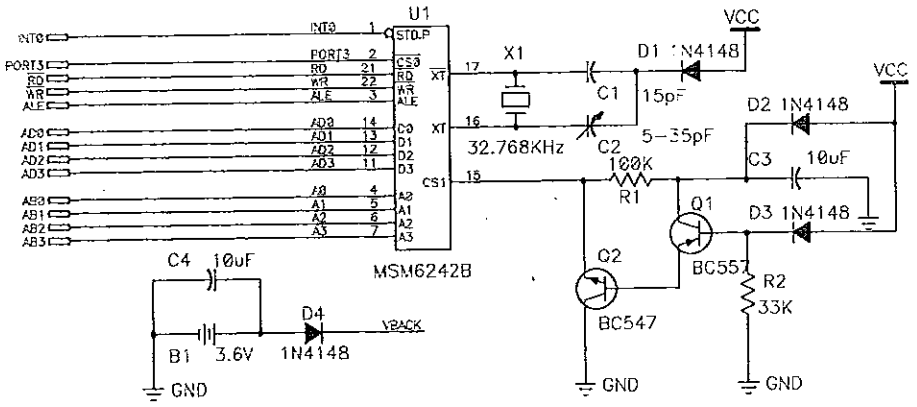
สำหรับรีจิสเตอร์ที่ใช้ในการโปรแกรมการทำงานของชิพนาฬิกาเวลาจริงนั้นมี 3 ตัวด้วยกัน โดยแต่ละตัวจะมีรูปแบบการโปรแกรมของบิทต่าง ๆ ซึ่งรายละเอียดจะไม่กล่าวถึง สำหรับในงานวิทยานิพนธ์ได้เลือกโปรแกรมการทำงานในรูปแบบแสดงเวลา 24 ชั่วโมง และช่วงเวลาของการอ่านข้อมูลจากรีจิสเตอร์เวลาในแต่ละครั้งจะห่างกันเท่ากับ 1 วินาที โดยใช้หลักการของการอินเตอร์รัพท์ไมโครคอนโทรลเลอร์ 8031 ที่เป็นตัวจัดการของระบบควบคุมการทำงานหลัก ขาสัญญาณที่ใช้ในการขอบริการอินเตอร์รัพท์ของชิพนาฬิกาเวลาจริงคือขา STD.P ซึ่งถูกโปรแกรมรูปแบบให้กำเนิดสัญญาณกระตุ้นทุก ๆ 1 วินาทีสำหรับวงจรที่ใช้งานจริงในการเชื่อมโยงส่วนจัดการนาฬิกาเวลาจริงแสดงในรูปที่ 3.23

ตาราง 3.1 แสดงรีจิสเตอร์ของชิพนาฬิกาเวลาจริงและรายละเอียด

หมายเลขรีจิสเตอร์	ค่าที่เป็นไปได้	รายละเอียด
0000	0-9	รีจิสเตอร์หลักหน่วยของวินาที
0001	0-5	รีจิสเตอร์หลักสิบของวินาที
0010	0-9	รีจิสเตอร์หลักหน่วยของนาที
0011	0-5	รีจิสเตอร์หลักสิบของนาที
0100	0-9	รีจิสเตอร์หลักหน่วยของชั่วโมง
0101	0-2 หรือ 0-1	รีจิสเตอร์หลักสิบของชั่วโมงหรือแสดง AM/PM ในกรณีที่ผู้ใช้งานเลือกแสดงแบบ 12 ชั่วโมง
0110	0-9	รีจิสเตอร์หลักหน่วยของวันในรอบเดือน
0111	0-3	รีจิสเตอร์หลักสิบของวันในรอบเดือน
1000	0-9	รีจิสเตอร์หลักหน่วยของเดือน
1001	0-1	รีจิสเตอร์หลักสิบของเดือน

ตารางที่ 3.1 (ต่อ)

1010	0-9	รีจิสเตอร์หลักหน่วยของปี
1011	0-9	รีจิสเตอร์หลักสิบของปี
1100	0-6	รีจิสเตอร์แสดงวันในสัปดาห์
1101		รีจิสเตอร์ในการควบคุมการทำงานของชิพ
1110		รีจิสเตอร์ในการควบคุมการทำงานของชิพ E
1111		รีจิสเตอร์ในการควบคุมการทำงานของชิพ F

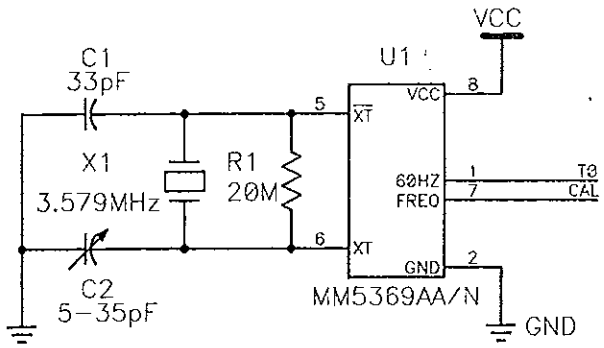


รูปที่ 3.23 วงจรส่วนจัดการนาฬิกาเวลาจริง

4.2 ส่วนสร้างสัญญาณอ้างอิงมาตรฐานในการสุ่มอ่านข้อมูล

ในการสุ่มอ่านค่าข้อมูลจากระบบเครือข่าย ในรูปแบบเวลาจริง จำเป็นต้องมีสัญญาณอ้างอิงมาตรฐาน เพื่อสั่งให้ระบบเครือข่ายประมวลผลสัญญาณพร้อม ๆ กัน เพื่อให้ผลของข้อมูลที่อ่านได้ เป็นผลที่เกิดจากการอ่านค่าข้อมูลที่เวลาเดียวกัน สัญญาณนี้จึงมีความจำเป็นอย่างยิ่งในการกำหนดความถูกต้องของช่วงเวลาของการสุ่มอ่าน และเนื่องจากรูปแบบการทำงานของระบบควบคุมพลักแบบเวลาจริงมีขั้นตอนโดยการส่งรหัสแบบอนุกรมให้กับระบบเครือข่ายทุกตัวและรออ่านข้อมูล ซึ่งจำเป็นต้องใช้เวลาในส่วนหนึ่ง จึงไม่สามารถใช้สัญญาณความถี่สูงในการกำหนดช่วงเวลาในการส่งรหัสได้จึงกำหนดให้สัญญาณกำหนดการสุ่มอ่านข้อมูลมีความถี่สูงสุด 60 HERTZ (Hz) และได้เลือกใช้ชิพกำเนิดสัญญาณ 60 Hz เบอร์ MM5369AA/N ซึ่งความเที่ยงตรงของความถี่สัญญาณ ถูกกำหนดโดยคริสตอลความถี่ 3.57954 MHz และตัวเก็บประจุโดยวงจรในส่วนกำเนิดสัญญาณความถี่ 60 Hz แสดงในรูปที่ 3.24

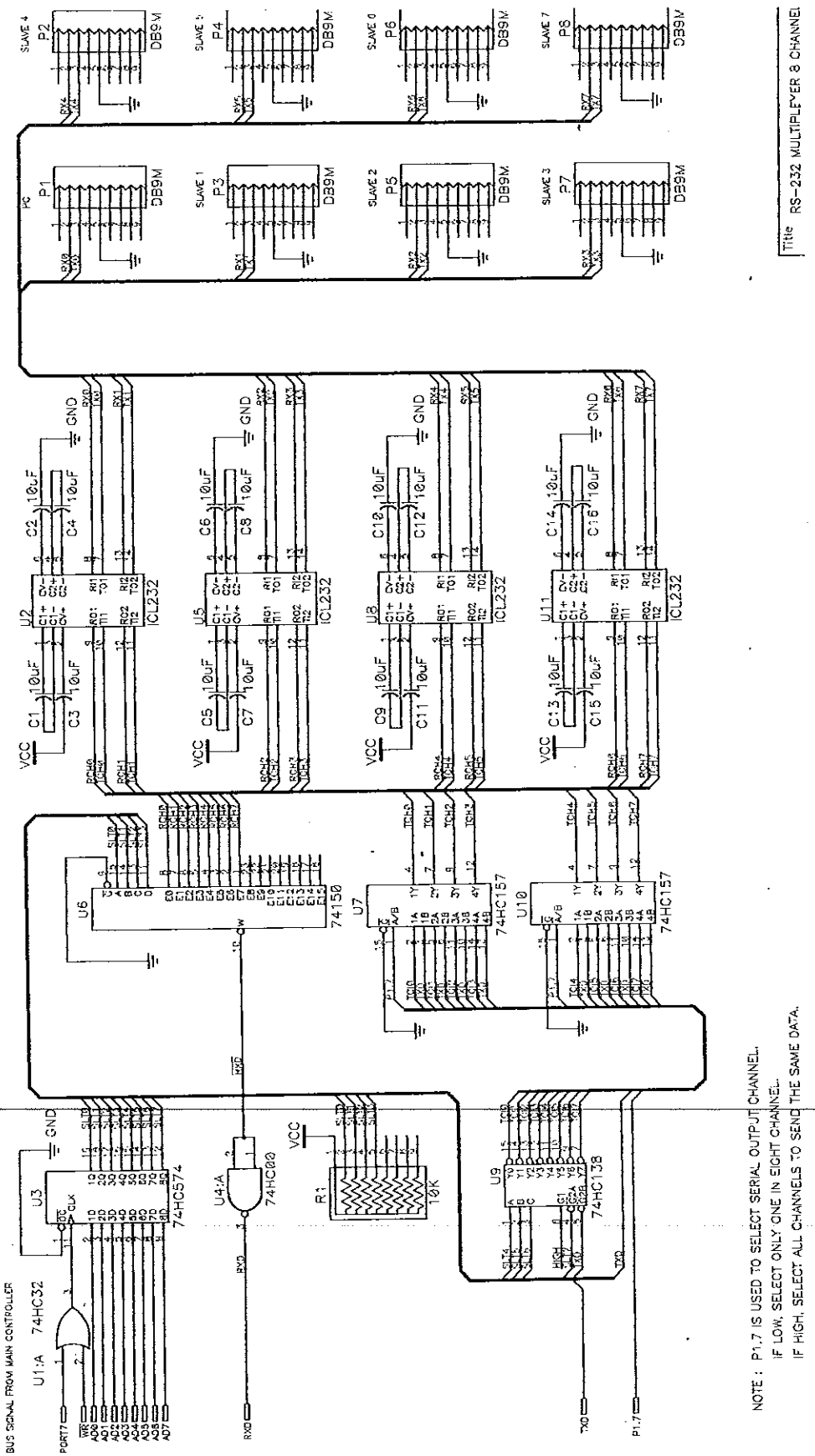
สัญญาณที่ได้จากเอาท์พุทของชิพ MM5369AA/N จะถูกป้อนเข้าสู่ขารับสัญญาณอ้างอิงในการนับ TO (ขาหมายเลข 14) ของไมโครคอนโทรลเลอร์ซึ่งโปรแกรมควบคุมการทำงานกำหนดค่าในการหารความถี่ เพื่อให้ได้ช่วงเวลาการสุ่มอ่านข้อมูลที่ผู้ใช้งานต้องการ



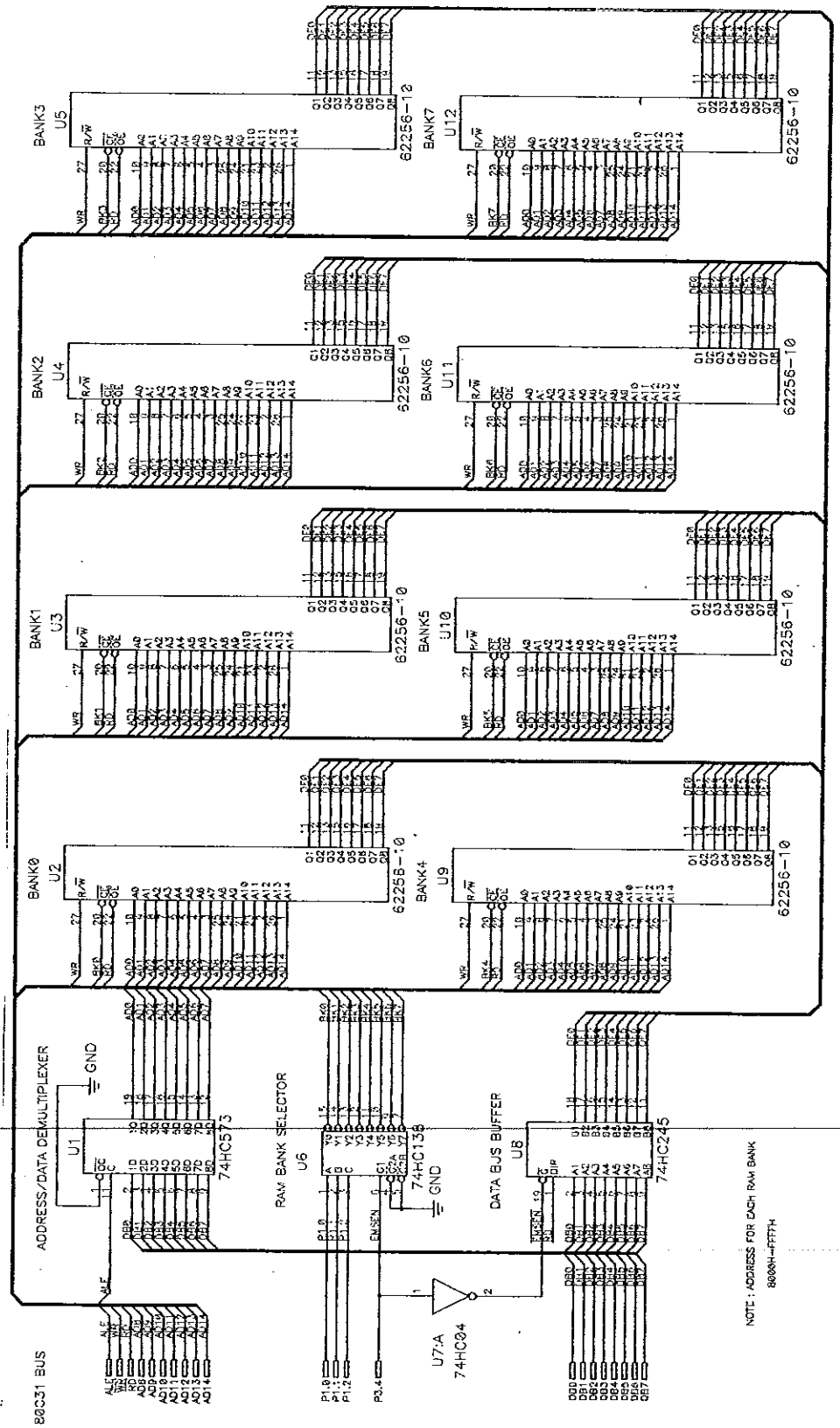
รูปที่ 3.24 วงจรกำเนิดสัญญาณความถี่ 60 Hz

จากส่วนจัดการย่อยของระบบควบคุมการทำงานหลักที่ได้แยกกล่าวมาแล้วนั้นสามารถนำมาเชื่อมโยงเป็นระบบควบคุมการทำงานหลัก โดยมีวงจรใช้งานจริงที่ครบถ้วนแสดงในรูปที่ 3.25 3.26 และ 3.27

รูปที่ 3.25 วงจรสมบูรณ์ของระบบควบคุมการทำงาน



รูปที่ 3.26 วงจรสมบูรณ์ของระบบควบคุมการทำงานหลัก (ต่อ)



รูปที่ 3.27 วงจรสมบูรณ์ของระบบควบคุมการทำงานหลัก (ต่อ)

ระบบเครือข่ายย่อย

จากข้อกำหนดของระบบเก็บข้อมูลระยะไกลที่ได้พัฒนาขึ้นนั้นจะต้องมีความสามารถในการอ่านข้อมูลจากจุดสำรวจหรือจุดทดลองต่าง ๆ ได้ จึงมีความจำเป็นอย่างยิ่งที่ระบบเก็บข้อมูลจะต้องประกอบด้วยระบบเครือข่ายย่อย ที่มีความสามารถในการแปลงรูปแบบสัญญาณที่ได้จากตัวแปรของการทดลองซึ่งส่วนใหญ่จะอยู่ในรูปของสัญญาณอนาลอกดังรายละเอียดที่ได้กล่าวไปแล้วในบทที่ 2 ให้อยู่ในรูปแบบของข้อมูลแบบดิจิตอล และระบบเครือข่ายย่อยยังจำเป็นต้องมีความสามารถในการจัดการสื่อสารเพื่อรับค่าตัวแปรกำหนดคำสั่งงานในการเริ่มสุ่มอ่านค่าข้อมูล จากระบบควบคุมการทำงานหลัก และส่งค่าตัวแปรที่ได้จากการทดลองเข้าสู่ระบบควบคุมการทำงานหลัก

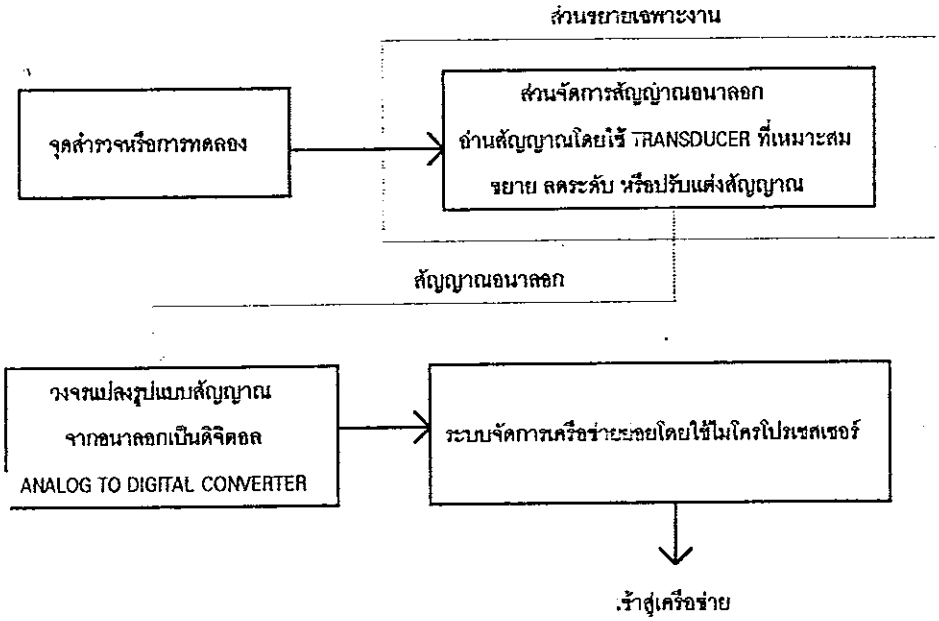
จากขีดความสามารถของระบบเครือข่ายย่อยตามข้อกำหนดเป็นการยากที่จะออกแบบระบบโดยการใช้วงจรแบบแยกส่วน จึงได้เลือกใช้ไมโครคอนโทรลเลอร์ 8031 เป็นตัวจัดการตามขอบเขตการทำงานของระบบแสดงในรูป 3.28

จากจุดสำรวจที่ต้องการอ่าน และบันทึกค่าตัวแปรซึ่งมีระบบเครือข่ายตั้งอยู่ข้อมูลที่ต้องการอ่านในรูปแบบของปริมาณทางฟิสิกส์ จะถูกเปลี่ยนเป็นสัญญาณทางไฟฟ้า โดยการเลือกใช้ทรานสดิวเซอร์(TRANSDUCER) ที่เหมาะสมกับการใช้งาน สัญญาณที่ได้จะมีคุณสมบัติเป็นไปตามเงื่อนไขของทรานสดิวเซอร์ และอยู่ในรูปแบบอนาลอก(ANALOG SIGNAL) จำเป็นต้องผ่านการจัดการโดยใช้วงจรรวมประเภทออปแอมป์(OP-AMP) ที่ผู้ใช้งานต้องออกแบบเองเพื่อให้ได้ค่าสัญญาณตามข้อกำหนด โดยการจัดการสัญญาณในส่วนนี้อาจเป็น

- การลดขนาดความสูงของสัญญาณ (SIGNAL AMPLITUDE)
- การขยายขนาดความสูงของสัญญาณ
- การกรองสัญญาณรบกวนออกจากสัญญาณจริงที่ได้จากทรานสดิวเซอร์

การที่จำเป็นต้องมีการจัดการสัญญาณ ก่อนที่จะเข้าสู่วงจรแปลงสัญญาณจากอนาลอกเป็นดิจิตอลได้ เนื่องจากว่าสิ่งที่ได้จากทรานสดิวเซอร์ในการทดลองนั้นผู้ใช้งานไม่สามารถกำหนดค่าให้เป็นไปตามต้องการได้โดยตรง และจำเป็นต้อง

อ้างอิงตามคุณสมบัติของการใช้งานมาตรฐานในการออกแบบระบบส่วนนี้จึงค่อนข้างมีปัญหามาก และจำเป็นต้องให้เป็นส่วนขยายของการนำไปใช้งาน



รูปที่ 3.28 แผนผังชอบเขตการทำงานของระบบเครือข่ายย่อย

1. วงจรแปลงรูปแบบสัญญาณจากอนาลอกเป็นดิจิตอล

เมื่อสัญญาณอนาลอกผ่านการจัดการจนได้เงื่อนไข เป็นไปตามที่ต้องการ คือ ต้องมีระดับความสูงของสัญญาณในช่วง 0-5 โวลท์แล้ว ก็สามารถป้อนเข้าสู่อินพุทของระบบเครือข่ายย่อยได้

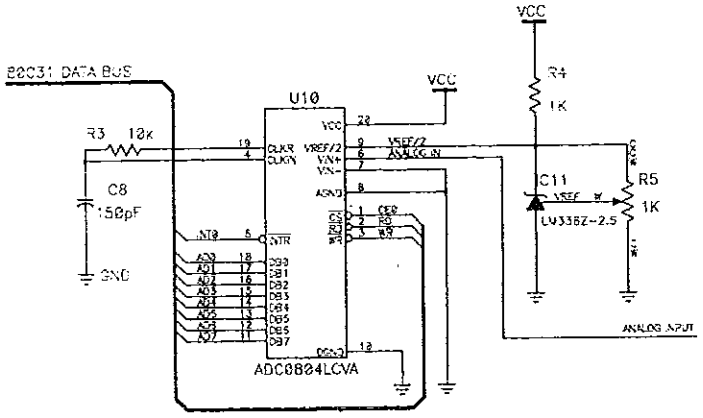
สัญญาณที่ผ่านการจัดการโดยแล้วระดับสัญญาณอยู่ในช่วง 0-5 โวลท์ ซึ่ง

เป็นสัญญาณที่สามารถประมวลผล โดยวงจรแปลงรูปแบบสัญญาณจากอนาลอกเป็นดิจิตอล ได้เลือกใช้ชิพแปลงสัญญาณเบอร์ ADC0804 ของบริษัท NATIONAL SEMICONDUCTOR เป็นชิพที่มีความสามารถในการแปลงสัญญาณ ขนาดความกว้างของบัสเอาท์พุท ซึ่งแทนค่าสัญญาณอนาลอกอินพุท 8 บิท ดังนั้นความละเอียด (RESOLUTION) ในการแปลงค่าสัญญาณจะเท่ากับ 1 ใน 256 แม้ว่าความเร็วในการแปลงสัญญาณ (CONVERSION TIME) ของชิพเบอร์นี้ไม่สูงมากนักและความละเอียดในการแทนค่าสัญญาณอนาลอกจะเท่ากับ 8 บิทซึ่งจัดว่าค่อนข้างหยาบในการใช้งานของงานวิจัย แต่เนื่องจากสามารถหาได้ง่ายในท้องตลาดในราคาที่ไม่แพงมากนัก จึงได้เลือกใช้ในการแปลงสัญญาณสำหรับงานวิทยานิพนธ์นี้

ในการต่อวงจรเพื่อใช้งานสำหรับส่วนแปลงสัญญาณโดยใช้ชิพ ADC0804 สามารถทำได้ โดยการเพิ่มส่วนประกอบภายนอกเพื่อกำหนดค่าครึ่งแรงดันอ้างอิง (Vref/2) ให้กับวงจรแปลงสัญญาณทางขาสัญญาณ(Vref/2) และส่วนกำเนิดสัญญาณนาฬิกาเพื่อเป็นฐานเวลาในการแปลงค่าสัญญาณ ซึ่งอยู่ในรูปแบบของวงจร RC คือ ต่อตัวต้านทานและตัวเก็บประจุให้กับขา CLKR และ CLKIN สำหรับการเชื่อมต่อกับไมโครคอนโทรลเลอร์ได้ต่อบัสเอาท์พุทของชิพแปลงสัญญาณ ADC0804 เข้ากับบัสข้อมูลของไมโครคอนโทรลเลอร์โดยตรงและมีกลุ่มสัญญาณ RD WR CS ซึ่งเป็นสัญญาณในการควบคุมการอ่าน เขียน และการเรียกใช้ชิพตามลำดับ

ในการสั่งงานให้ชิพทำการเปลี่ยนสัญญาณของข้อมูลที่ต้องการอ่านค่า เมื่อได้รับคำสั่งงานจากระบบควบคุมหลักในรูปแบบการทำงานแบบเวลาจริง หรือเมื่อค่าของเวลาในการสุ่มอ่านข้อมูลครบ ตามกำหนดในรูปแบบการทำงานแบบไม่เป็นเวลาจริง ไมโครคอนโทรลเลอร์จะส่งระหัสเพื่ออ้างอิงตำแหน่งของชิพ ADC0804 ซึ่งเทียบเท่ากับการสั่งงานให้ชิพเริ่มทำการแปลงสัญญาณโดยในระบบเครือข่ายย่อยที่ได้สร้างขึ้นนั้นได้กำหนดระหัสอ้างอิงเป็น 8XXXH หลังจากการเปลี่ยนค่าสัญญาณเสร็จสิ้น ชิพจะส่งสัญญาณ INTR ซึ่งถูกต่อเข้ากับขาสัญญาณขอบริการอินเตอร์รัพท์จากระบบภายนอกหมายเลข 0 (INT0) มีผลทำให้ไมโครคอนโทรลเลอร์อ่านค่าจากชิพ

สำหรับรูปแบบการเชื่อมต่อ ระหว่างชิพแปลงสัญญาณอนาลอกเป็นดิจิตอลและไมโครคอนโทรลเลอร์รวมทั้งวงจรส่วนขยายภายนอกของชิพแสดงในรูปที่ 3.29



รูปที่ 3.29 วงจรเชื่อมต่อระหว่างชิพแปลงสัญญาณอนาลอกเป็นดิจิตอล กับไมโครคอนโทรลเลอร์

2. ส่วนเชื่อมโยงระหว่างระบบไมโครคอนโทรลเลอร์กับระบบเครือข่าย

เนื่องจากรูปแบบเครือข่ายสำหรับการรับส่งข้อมูล เป็นแบบอนุกรมซึ่งไมโครคอนโทรลเลอร์ 8031 สามารถจัดกระทำได้โดยไม่จำเป็นต้องเพิ่มเติมอุปกรณ์ช่วยเหลือภายนอก ซึ่งต่างจากกรณีที่ใช้ไมโครโปรเซสเซอร์เป็นตัวจัดการระบบการแปลงข้อมูลแบบขนาน 8 บิทในรูปแบบของข้อมูลแบบอนุกรมจะเป็นกระบวนการภายในโดยสัญญาณหลังจากที่ผ่านทางขาสัญญาณ RXD โดยที่สัญญาณ ทั้งสองนี้จะมีหรือต้องมีระดับสัญญาณที่ขา TXD และ RXD ของไมโครคอนโทรลเลอร์จะอยู่ในระดับ TTL(ระดับลอจิกต่ำ 0 โวลท์ สูง 5 โวลท์) แต่ในการเชื่อมโยงเพื่อสื่อสารข้อมูลกับระบบอื่น ๆ ตามมาตรฐานแล้วจำเป็นต้องมีการเปลี่ยนระดับ หรือเปลี่ยนรูปแบบสัญญาณ สำหรับมาตรฐานของการเชื่อมโยงระบบภายใต้เครือข่ายแบบ STAR ที่เลือกใช้ จะมีรายละเอียดในการแปลงรูปแบบสัญญาณดังนี้

2.1 การเชื่อมโยงระบบไมโครโปรเซสเซอร์แบบอนุกรมมาตรฐาน RS-232C

ในการเชื่อมโยงระหว่างระบบ ที่มีไมโครโปรเซสเซอร์ หรือไมโครคอนโทรลเลอร์ควบคุมแบบอนุกรม รูปแบบสัญญาณจะต้องถูกเปลี่ยนจากระดับ TTL ให้อยู่ในระดับมาตรฐาน RS-232 คือ

2.1.1 ระดับลอจิกสูงกรณีที่มี LOAD ซึ่งหมายถึง ในสายสื่อสารมีระบบที่ต้องการติดต่อด้วยเชื่อมต่ออยู่ในระดับ 3 ถึง -15 โวลท์ และจะต้องเป็น -25 โวลท์ในกรณีที่ไม่มี LOAD

2.1.2 ระดับลอจิกต่ำในกรณีที่มี LOAD จะต้องอยู่ในระดับ +3 ถึง +15 โวลท์ และในกรณีที่ไม่มี LOAD จะต้องเป็น +25 โวลท์

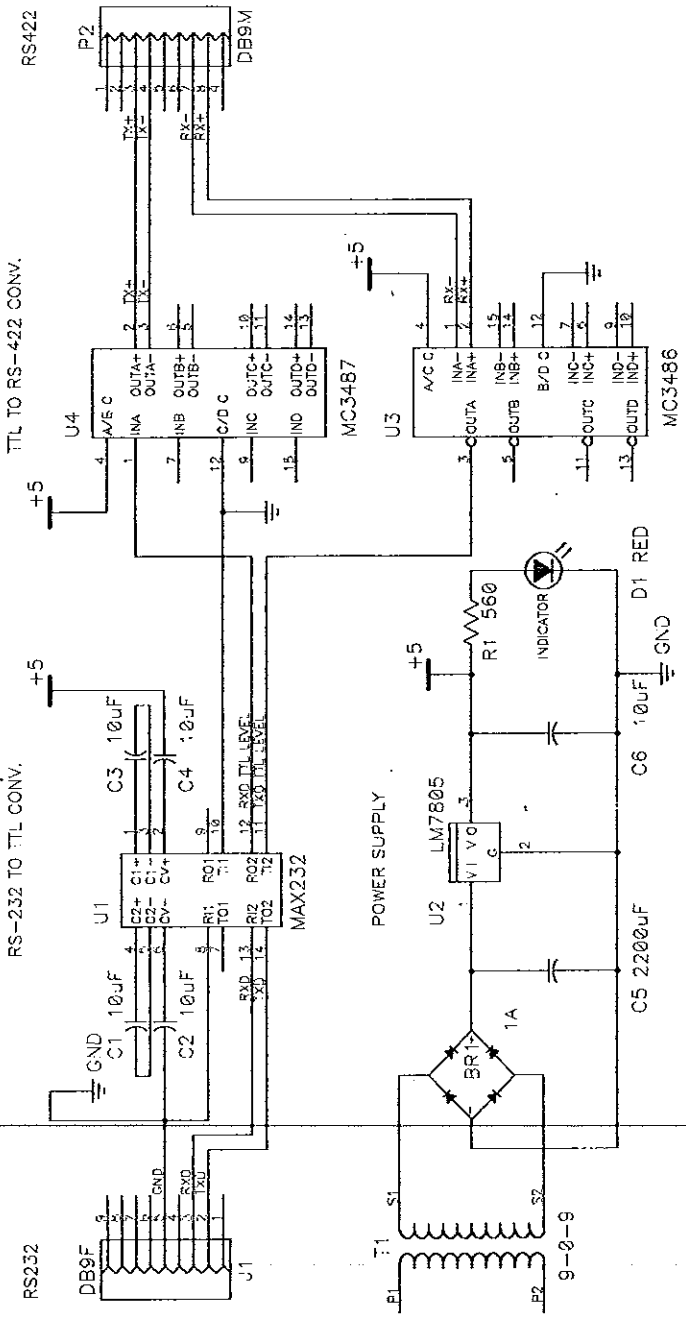
สำหรับระดับที่เลือกใช้งานโดยทั่วไปนั้นจะอยู่ในช่วง +/- 12 โวลท์ผู้ใช้งานสามารถเลือกใช้ชิพมาตรฐานเฉพาะงาน ที่ทำหน้าที่ในการเปลี่ยนระดับสัญญาณให้เป็นไปตามมาตรฐานการเชื่อมต่อได้เช่นในภาคส่งข้อมูล(TRANSMITTED DATA) ใช้ชิพมาตรฐานเบอร์ MC1488 และในภาครับข้อมูล(RECEIVED DATA) ใช้ชิพมาตรฐานเบอร์ MC1489 ของบริษัท MOTOROLA ซึ่งมีข้อเสียคือจำเป็นต้องใช้แหล่งจ่ายกำลังไฟฟ้าระดับ +12 และ -12 โวลท์ ในกรณีที่เลือกใช้ชิพเหล่านี้จึงได้มีการออกแบบชิพมาตรฐานเบอร์ MAX232 เพื่อทำหน้าที่ในการจัดการทั้งการส่งและการรับข้อมูลโดยใช้แหล่งจ่ายไฟฟ้าเพียงชุดเดียวคือระดับ 5 โวลท์ ซึ่งเป็นระดับปกติที่ใช้ในระบบไมโครคอนโทรลเลอร์อยู่แล้ว โดยในชิพตัวเดียวจะมีส่วนจัดการรับและส่งข้อมูล 2 ชุด เป็นอิสระจากกัน จากข้อดีของชิพ MAX232 ทำให้ผู้วิจัยเลือกติดตั้งชิพเบอร์นี้สำหรับการเชื่อมต่อระบบภายใต้มาตรฐาน RS-232 ในเครื่องต้นแบบ

2.2 การเชื่อมโยงระบบไมโครโปรเซสเซอร์และอนุกรมมาตรฐาน RS-422

มาตรฐานการเชื่อมโยงระบบแบบอนุกรมแบบนี้ ใช้หลักการการเปลี่ยนข้อมูลที่ต้องการส่งให้เป็นสัญญาณเอาท์พุทแบบกลับขั้วสัญญาณ (INVERTING OUTPUT) และสัญญาณเอาท์พุทแบบไม่กลับขั้วสัญญาณ(NON-INVERTING OUTPUT)

ในระดับ 5 รีวลท์และส่งสัญญาณทั้งสองนี้ไปกับสายส่งสัญญาณเมื่อสัญญาณไปถึงอุปกรณ์รับข้อมูล จะมีการเปลี่ยนกับสัญญาณเป็นข้อมูลจริง ดังนั้นในการจัดการกับข้อมูลแต่ละอย่างจะใช้สายสัญญาณ 2 เส้น ซึ่งในกรณีที่เป็นระบบการสื่อสารระยะไกลกว่าที่ความเร็วในการสื่อสารข้อมูลเท่าๆ กัน(รายละเอียดเปรียบเทียบระหว่างมาตรฐาน RS-232 และ RS-422 ได้กล่าวไปแล้วในบทที่ 2) ในปัจจุบันนี้สามารถหาชิพที่ทำหน้าที่ในการเปลี่ยนข้อมูลให้อยู่ในมาตรฐาน RS-422 ได้ คือ ชิพเบอร์ MC3486 ในภาครับข้อมูล และ MC3487 ในภาคส่งข้อมูล ซึ่งชิพทั้งสองนี้ได้เลือกติดตั้งในเครื่องต้นแบบ

วงจรส่วนเปลี่ยนมาตรฐานการจัดการสื่อสารข้อมูล ระหว่างระบบเครือข่ายย่อยและเครือข่ายระบบจากระบบ RS-232C เป็น RS-442 แสดงในรูปที่ 3.30



รูปที่ 3.30 วงจรเปลี่ยนมาตรฐานการสื่อสารข้อมูลระหว่าง RS-232C และ RS-422

บทที่ 4

โปรแกรมควบคุมระบบงาน

จากสถาปัตยกรรมทางฮาร์ดแวร์ ของระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์ประเภทไมโครคอนโทรลเลอร์ที่ได้กล่าวไปแล้วในบทที่ 3 เป็นตัวกำหนดรูปแบบการทำงานของระบบในส่วนหนึ่ง ระบบของไมโครโปรเซสเซอร์และไมโครคอนโทรลเลอร์จะไม่สามารถทำงานได้โดยอาศัยเพียงตัวไมโครโปรเซสเซอร์และไมโครคอนโทรลเลอร์และวงจรเชื่อมต่อภายนอกเท่านั้น แต่ระบบควบคุมจำเป็นต้องมีโปรแกรมในการควบคุมการทำงาน ซึ่งต่างจากระบบควบคุมที่ใช้อุปกรณ์ลอจิกแบบแยกส่วน ซึ่งสามารถทำงานตามขั้นตอนโดยอาศัยเพียงส่วนประกอบทางฮาร์ดแวร์

ดังนั้น รายละเอียดในส่วนของโปรแกรมควบคุมระบบทำงาน หรือส่วนซอร์ฟแวร์ ซึ่งทำหน้าที่ในการคำนวณผลและการควบคุมทุกองค์ประกอบของระบบโดยรวม จึงได้แยกนำเสนอในบทนี้โดยกล่าวถึงขั้นตอนการทำงานแบบใช้แผนภูมิสายงาน (FLOWCHART) และรายละเอียดของแผนภูมิสายงานแยกเป็นส่วน ๆ เช่นเดียวกับที่ได้กล่าวไปในรายละเอียดส่วนสถาปัตยกรรมทางฮาร์ดแวร์

ภาษาคอมพิวเตอร์ที่เลือกใช้ และรูปแบบของโปรแกรมควบคุมไมโครคอนโทรลเลอร์ 8031 โดยทั่วไป

1. ภาษาคอมพิวเตอร์ที่เลือกใช้

ในการพัฒนาระบบควบคุมโดยใช้ไมโครโปรเซสเซอร์ หรือไมโครคอน-โทรลเลอร์ ปัจจุบันมีทางเลือกในการใช้ภาษาอื่น ๆ นอกเหนือจากการใช้ภาษาแอสเซมบลีพัฒนาเพียงอย่างเดียว เนื่องจากมีบริษัทผู้ผลิตซอร์ฟแวร์บางรายได้สร้างโปรแกรมคอมไพเลอร์ที่มีความสามารถจะแปลงภาษาสูงในรูปแบบปาสคาล(PASCAL) หรือ C ให้ไฟล์ข้อมูลเอาท์พุทเป็นภาษาแอสเซมบลีของไมโครโปรเซสเซอร์

ใด ๆ นอกเหนือจากไมโครโปรเซสเซอร์ตระกูล 80X86 เช่น โปรแกรม C-COMPILER ของบริษัท AVOCET INC. สามารถคอมไพล์ (COMPILE) เพื่อให้เป็นผลที่ได้เป็นภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล MCS-51

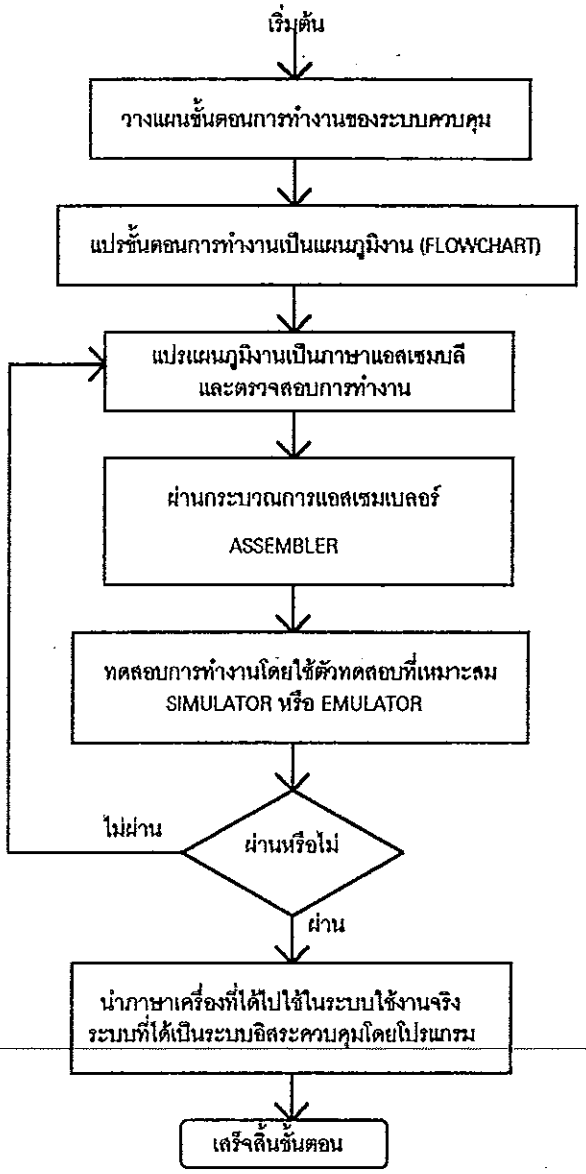
สำหรับภาษาแอสเซมบลีที่ได้ ก็สามารถเรียกใช้โปรแกรมแอสเซมเบลอร์เพื่อสร้างระหัสภาษาเครื่อง เพื่อนำไปใช้ในการควบคุมจริงโดยปกติแล้วแอสเซม-เบลอร์สำหรับโปรแกรมภาษาแอสเซมบลีที่ได้จากภาษาสูงนั้นจะมีรูปแบบเฉพาะ ทำให้ผู้ใช้งานจำเป็นต้องเจาะจงการใช้โปรแกรมแอสเซมเบลอร์ที่มาคู่กับคอมไพเลอร์นั้น ๆ และการที่จะเขียนโปรแกรมภาษาแอสเซมบลี โดยในหลักการที่กำหนดมากับแอสเซมเบลอร์นั้น ๆ เป็นสิ่งที่ทำได้ยาก เนื่องจากมีข้อจำกัดเกี่ยวกับการใช้ DIRECTIVE และมาโคร (MACRO) บางอย่างซึ่งเป็นประสบการณ์ของผู้วิจัยในการใช้งานโปรแกรม C-COMPILER ของบริษัท AVOCET INC. นอกจากนั้นยังมีรายละเอียดอื่น ๆ ที่เป็นข้อจำกัดของการใช้ภาษาสูงในการพัฒนาระบบควบคุม

นอกเหนือจากรายละเอียดของการใช้งานโปรแกรมคอมไพเลอร์ภาษาสูงแล้ว โดยรูปแบบแล้ว การใช้ภาษาแอสเซมบลีในการพัฒนาระบบยังมีข้อได้เปรียบภาษาสูง คือ โปรแกรมหรือฟังก์ชั่นที่ได้จากการพัฒนาด้วยภาษาแอสเซมบลีนั้น มีความเฉพาะเจาะจงในการทำงาน การเลือกใช้หรือเก็บค่าตัวแปรมีความรัดกุมมากกว่า เนื่องจากในการสร้างคอมไพเลอร์สำหรับภาษาใด ๆ นั้น ผู้สร้างจำเป็นต้องเลือกการแปรความหมายของการสั่งงาน (SYNTAX) ให้อยู่ในแบบทั่วไป เพราะจะทำให้การคอมไพล์หรือแปรความหมายทำได้ง่ายขึ้น ดังนั้นการใช้คำสั่งหนึ่ง ๆ อาจจะโยงไปเรียกฟังก์ชั่นการทำงานย่อย ๆ ที่มีอยู่อีกมากมาย ซึ่งแต่ละขั้นจำเป็นต้องมีการเก็บค่าตัวแปรประเภทรีจิสเตอร์(REGISTER) ทั้งหมด(เนื่องจากคอมไพเลอร์ไม่สามารถรู้ได้ว่าในการประมวลผลโดยฟังก์ชั่นก่อนหน้านี้ ตัวแปรใดบ้างที่ยอมให้มีการเปลี่ยนค่าและตัวแปรใดที่ไม่ยอมให้เปลี่ยนค่า จึงใช้วิธีการเก็บค่าทั้งหมดก่อนที่จะดึงค่าเก่ากลับมาหลังจากที่ประมวลผลเสร็จในฟังก์ชั่นปัจจุบัน) ในขณะที่การใช้ภาษาแอสเซมบลีในการพัฒนาโปรแกรมนั้น ผู้ใช้งานสามารถเลือกเก็บค่ารีจิสเตอร์เฉพาะที่มีการเปลี่ยนแปลงในการเรียกใช้ฟังก์ชั่นต่อไปเท่านั้น ดังนั้นจึงเป็นปกติที่โปรแกรมภาษาเครื่องที่ได้จากพัฒนาด้วยภาษาสูงมีขนาดโตกว่าโปรแกรมที่พัฒนาด้วย

ภาษาแอสเซมบลีทั้ง ๆ ที่มีรูปแบบการทำงานอย่างเดียวกัน และผลที่ตามมาเมื่อเปรียบเทียบความเร็วในการทำงานระหว่างระบบที่พัฒนาด้วยภาษาสูง และระบบที่พัฒนาด้วยภาษาแอสเซมบลี ก็คือระบบที่พัฒนาด้วยภาษาแอสเซมบลีมีความเร็วในการทำงานสูงกว่าซึ่งสามารถเห็นได้อย่างชัดเจน

นอกจากนั้น ถึงแม้ว่าผู้พัฒนาระบบควบคุมจะสามารถเลือกใช้ภาษาสูงในการพัฒนาระบบแต่ในบางขั้นตอนของโปรแกรม เช่น ขั้นตอนของโปรแกรมบริการอินเตอร์รัพท์ ซึ่งจำเป็นต้องเขียนขั้นตอนในการบริการให้สั้นที่สุดเท่าที่จะเป็นไปได้ ก็ยังจำเป็นต้องใช้โปรแกรมภาษาแอสเซมบลี

จากข้อเปรียบเทียบดังที่ได้กล่าวมาแล้วนั้น ทำให้ผู้วิจัยได้ตัดสินใจเลือกพัฒนาระบบโดยใช้ภาษาแอสเซมบลีทั้งหมด เนื่องจากประเภทของงานเป็นระบบไมโครคอนโทรลเลอร์แบบเวลาจริง และในบางขั้นตอนของงานจำเป็นต้องมีการบริหารอินเตอร์รัพท์ ซึ่งโปรแกรมในการควบคุมในส่วนนี้ต้องสั้น สำหรับแผนภูมิรูปแบบของการพัฒนาระบบควบคุมโดยใช้ภาษาแอสเซมบลีในรูปที่ 4.1



รูปที่ 4.1 แผนภูมิขั้นตอนการพัฒนาระบบควบคุมโดยใช้ภาษาแอสเซมบลี

สำหรับโปรแกรมแอสเซมเบลอร์ที่เลือกใช้ เป็นของบริษัท BINARY TECHNOLOGY INC. เนื่องจากเป็นแอสเซมเบลอร์เฉพาะเจาะจงสำหรับไมโครคอนโทรลเลอร์ตระกูล MCS-51 จึงมีความสามารถในการทำกระบวนการแอสเซมเบลอร์ หรือแปลงระหัสจากภาษาแอสเซมบลีเป็นเลขฐาน 16 ของภาษาเครื่อง (MACHINE CODE) ได้รวดเร็ว (ในกรณีที่ใช้แอสเซมเบลอร์ที่ไม่เจาะจงไมโครโปรเซสเซอร์)จะใช้หลักการเปิดตารางซึ่งอยู่ในรูปแบบของไฟล์ข้อมูล เพื่อแปลงความหมายของภาษาแอสเซมบลีให้เป็นภาษาเครื่อง ดังนั้นความเร็วในการทำงานจึงต่ำกว่า) นอกจากรูปแบบของ DIRECTIVE เช่น การกำหนดตัวแปร และการกำหนดตารางข้อมูล มีความคล้ายคลึงกับ DIRECTIVE ของภาษาแอสเซมบลีที่กำหนดโดยบริษัท INTEL จึงมีความสะดวกในการพัฒนาระบบมากขึ้น

2. โครงสร้างส่วนโปรแกรมควบคุมของไมโครคอนโทรลเลอร์ MCS-51 โดยทั่วไป

ในส่วนของโปรแกรมควบคุมระบบไมโครคอนโทรลเลอร์ตระกูล MCS-51 ของบริษัท INTEL โดยทั่ว ๆ ไป สามารถจัดโครงสร้างของสายงานโปรแกรมได้เป็นสองส่วน คือ สายงานโปรแกรมควบคุมหลัก (MAIN PROGRAM) และสายงานโปรแกรมบริการอินเตอร์รัพท์

โดยขั้นตอนการทำงานของระบบไมโครคอนโทรลเลอร์ MCS-51 แล้วหลังจากที่ระบบถูกรีเซ็ต (RESET) ด้วยระบบลอจิกสูงที่ขาหมายเลข 9 (RESET) ไมโครคอนโทรลเลอร์จะเริ่มอ่านคำสั่งควบคุมทำงาน โดยที่สถานะของตัวชี้ตำแหน่งคำสั่ง (PROGRAM COUNTER) จะชี้ไปยังตำแหน่งแอดเดรสหมายเลข 0000 และในกรณีที่ระบบลวบคุม ต้องการใช้สายงานโปรแกรมบริการอินเตอร์รัพท์ด้วย โครงสร้างของไมโครคอนโทรลเลอร์ MCS-51 ใช้หลักการให้บริการอินเตอร์รัพท์แบบเวคเตอร์ คือ เมื่อเกิดอินเตอร์รัพท์ (รวมทั้ง RESET) จากแหน่งกำเนิดใด ๆ จะมีผลทำให้ ตัวชี้ตำแหน่งคำสั่งของไมโครคอนโทรลเลอร์ชี้ไปยังตำแหน่งแอดเดรสที่เฉพาะเจาะจงต่อแหล่งกำเนิดนั้น ๆ โดยรายละเอียดของตัวชี้ตำแหน่งคำสั่งและแหล่งกำเนิดอินเตอร์รัพท์ หลังจากที่มีการตรวจจับการขอบริการแสดงในตารางที่ 4.1

ในรูปแบบของการอ้างอิงตำแหน่งโปรแกรมการ ให้บริการอินเตอร์รัพท์

แบบนี้ ผู้ใช้งานจำเป็นต้องระบุตำแหน่งของโปรแกรมบริการที่แท้จริงไว้ที่ตำแหน่งของตัวชี้ เช่น ในกรณีต้องการเขียนสายงานโปรแกรมควบคุมหลัก ซึ่งเรียกชื่อว่า MAIN และสายงานโปรแกรมบริการอินเตอร์รัพท์ 2 แหล่ง คือ EXT0 (EXTERNAL INTERUPT 0) และ SERIAL สามารถกำหนดโปรแกรมควบคุมได้ดังนี้

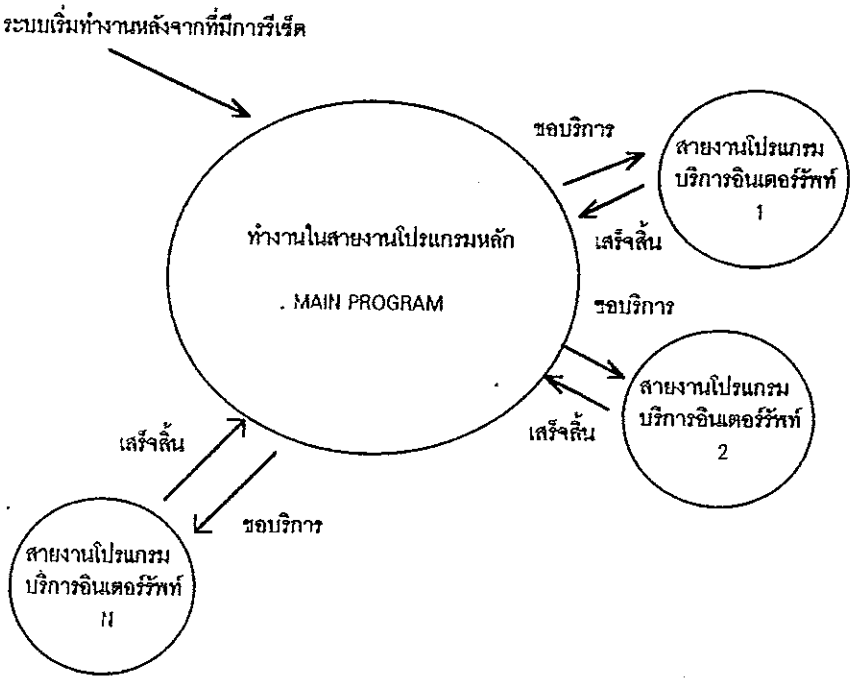
แหล่งกำเนิดอินเตอร์รัพท์	ตัวชี้ตำแหน่งคำสั่ง (VECTOR)	รายละเอียด
RESET	0000H	RESET
IE0	0003H	EXTERNAL INTERRUPT #0
TF0	000BH	TIMER INTERRUPT #0
IE1	0013H	EXTERNAL INTERRUPT #1
TF1	001BH	TIMER INTERRUPT #1
SERIAL	0023H	SERIAL INTERRUPT

ตารางที่ 4.1 แหล่งกำเนิดอินเตอร์รัพท์และตัวชี้ตำแหน่งคำสั่งของไมโครคอนโทรลเลอร์ MCS-51

```
ORG 0000H
AJMP MAIN
ORG 0003H
AJMP EXT0
ORG 0023H
AJMP SERIAL
```

และโปรแกรมบริการต่าง ๆ สามารถเขียนไว้ในตำแหน่งใด ๆ ในไฟล์ข้อมูล โดยที่โปรแกรมแอสเซมเบลอร์จะทำหน้าที่ในการแปลงค่าให้โดยอัตโนมัติการสั่งงานในรูปแบบนี้มีความหมายว่าในกรณีที่ระบบเกิดการรีเซ็ตให้กระโดดไปทำงานที่สายงานโปรแกรมควบคุมหลักและหากเกิดการอินเตอร์รัพท์ EXTERNAL INTERRUPT หมายเลข 0 ก็ให้กระโดดไปทำงานที่สายงานโปรแกรมบริการอินเตอร์รัพท์ EXT0 และเช่นเดียวกันสำหรับกรณีที่เกิดการอินเตอร์รัพท์ของ SERIAL รูปที่ 4.2 แสดงให้เห็นความชัดเจนของขั้นตอนการเกิดอินเตอร์รัพท์และการให้บริการ

ในส่วนของสายงานโปรแกรมบริการอินเตอร์รัพท์ จำเป็นต้องมีการเก็บค่าตัวแปรและรีจิสเตอร์ที่ใช้ร่วมกับสายงานโปรแกรมบริการหลัก มิฉะนั้นค่าของตัวแปรหรือรีจิสเตอร์ของสายงานโปรแกรมหลักจะถูกเปลี่ยนแปลงค่า ซึ่งในบางครั้งก่อให้เกิดปัญหามาก โดยการเก็บค่าจะกระทำก่อนที่จะเข้าสู่รูปแบบโปรแกรมบริการอินเตอร์รัพท์จริง โดยใช้คำสั่ง PUSH ซึ่งมีผลในการนำค่าตัวแปรหรือรีจิสเตอร์ออกมาจาก STACK POINTER ซึ่งมีข้อชี้ คือ ค่าตัวแปรหรือรีจิสเตอร์ที่ถูกเก็บก่อนจะต้องถูกนำออกมาหลัง ดังตัวอย่างโปรแกรมให้บริการอินเตอร์รัพท์



รูปที่ 4.2 การเกิดอินเตอร์รัพท์และการดำเนินไปของโปรแกรมควบคุมการทำงานของไมโครคอนโทรลเลอร์หรือไมโครโปรเซสเซอร์

```
EXTO:   PUSH ACC
        PUSH PSW
        PUSH DPH
        PUSH DPL
        ตัวโปรแกรมบริการ
        POP DPL
        POP DPH
        POP PSW
        POP ACC
        RETI
```

เงื่อนไขการออกจากโปรแกรมบริการอินเตอร์รัพท์ จะเกิดขึ้นเมื่อโปรแกรมทำงานถึงคำสั่ง RETI ในกรณีที่ระบบควบคุมต้องการใช้อินเตอร์รัพท์จากแหล่ง และมีบางแหล่งที่ต้องการการตอบสนองและบริการโดยทันทีไมโครคอนโทรลเลอร์ MCS-51 เปิดโอกาสให้ผู้ใช้สามารถกำหนดระดับของการตอบสนอง และบริการได้ ซึ่งเรียกว่าการจัดอันดับของการให้บริการอินเตอร์รัพท์ (INTERRUPT PRIORITY) โดยกระทำกับรีจิสเตอร์ IP (INTERRUPT PRIORITY CONTROL) และในการสั่งงานให้มีการเปิดรับหรือปิดรับการขออินเตอร์รัพท์ สามารถกระทำโดยผ่านทางบิทควบคุมของอินเตอร์รัพท์นั้น ๆ ซึ่งรายละเอียดสามารถหาอ่านได้จากคู่มือการใช้งานไมโครคอนโทรลเลอร์ MCS-51

การบริหารหน่วยความจำส่วนข้อมูลแบบ RING หรือ CIRCULAR BUFFER

จากที่ได้กล่าวไปแล้วนั้นเป็นรูปแบบโดยทั่วไปของการพัฒนาด้านโปรแกรมควบคุมการทำงานของระบบไมโครคอนโทรลเลอร์ สำหรับในรายละเอียดนั้นผู้พัฒนาจำเป็นต้องใช้ประสบการณ์ควบคู่กับเครื่องมือในการช่วยเหลือ เช่น ระบบพัฒนา

ในหัวข้อนี้จะได้กล่าวถึงองค์ประกอบของโปรแกรมควบคุมการทำงานที่ใช้ร่วมในระบบที่ได้พัฒนาขึ้นสำหรับงานวิทยานิพนธ์

เนื่องจากการรับ และถอดระหัสข้อมูล เป็นองค์ประกอบร่วมของระบบไมโครคอนโทรลเลอร์ที่ได้ออกแบบใช้ในงานวิทยานิพนธ์ ดังนั้น จึงได้แยกกล่าวถึงหลักการของการบริหารหน่วยความจำส่วนข้อมูลแบบ RING หรือ CIRCULAR BUFFER เพื่อใช้ในการอ้างอิงภายหลัง

ในส่วนการรับข้อมูลของระบบที่ควบคุมโดยไมโครโปรเซสเซอร์ โดยปกติจะสามารถแบ่งได้เป็น 2 แบบตามรูปแบบการใช้งาน คือ

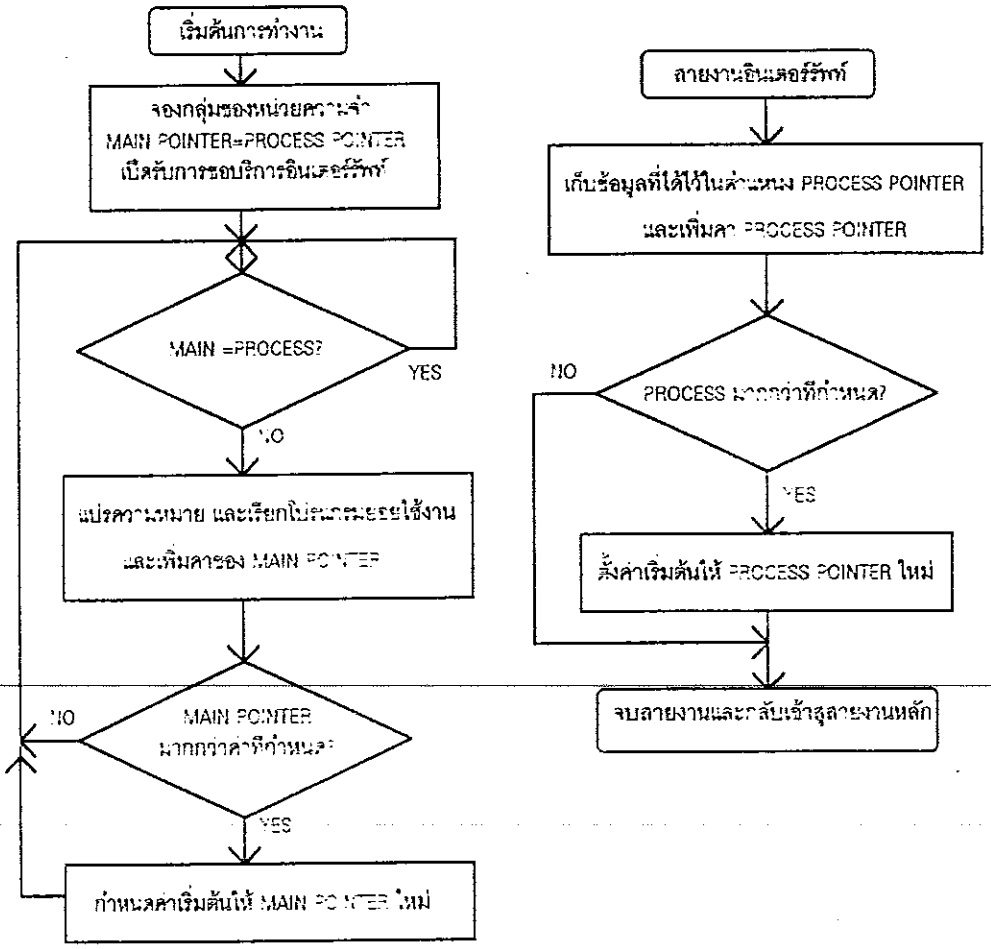
1. การรับข้อมูลโดยใช้หลักการ POLLING หลักการแบบนี้กำหนดให้ไมโครโปรเซสเซอร์รอรับข้อมูลแบบทำซ้ำใน LOOP ของการรับข้อมูลโดยมีเงื่อนไขของการออกจาก LOOP ก็ต่อเมื่อข้อมูลเข้าสู่ระบบแล้ว หรือครบตามเวลาที่กำหนดแล้วยังไม่มีข้อมูลเข้า (TIME OUT) หลักการนี้มีข้อเสีย คือ ไมโครคอนโทรลเลอร์

จำเป็นต้องรอข้อมูล โดยยังไม่สามารถข้ามไปดำเนินการในสายงานส่วนอื่น ๆ ของโปรแกรมหลักได้

2. การรับข้อมูลโดยการใช้หลักอินเตอร์รัพท์ (INTERRUPT) หลักการนี้จะเริ่มจากการจองหน่วยความจำของระบบ เพื่อเป็นส่วนรับข้อมูลโดยเฉพาะรู้ตำแหน่งเริ่มต้นและตำแหน่งสิ้นสุด(หรือขนาดของกลุ่มข้อมูล) ที่แน่นอนและมีการใช้สายงานของโปรแกรมให้บริการอินเตอร์รัพท์ เพื่อทำหน้าที่ในการรับข้อมูลเข้าหน่วยความจำส่วนรับข้อมูลเพิ่มค่าตัวชี้ตำแหน่งหน่วยความจำ(ในกรณีของไมโครคอนโทรลเลอร์ MCS-51 ใช้ DATA POINTER เป็นตัวชี้ตำแหน่งหน่วยความจำ) และตรวจสอบหาตำแหน่งสิ้นสุดของหน่วยความจำ ขั้นตอนที่กล่าวมานั้น กำหนดให้เกิดใน 1 ครั้งต่อครั้งของการเรียกขอบริการอินเตอร์รัพท์ ในส่วนของสายงานโปรแกรมหลักจะทำหน้าที่ในการแปรความหมายหรือทำงานอื่น ๆ ก่อนได้ตามต้องการ และย้อนกลับมาอ่านข้อมูลเพื่อแปรความหมาย โดยไม่จำเป็นต้องคอยรับข้อมูลเหมือนหลักการแบบ POLLING หลักการนี้เหมาะสำหรับงานการรับข้อมูลที่มีความสำคัญสูง ไม่ยอมให้เกิดความผิดพลาด และไมโครคอมพิวเตอร์จำเป็นต้องจัดการในส่วนโปรแกรมอื่นก่อน

โปรแกรมควบคุมส่วนรับข้อมูลสำหรับงานวิทยานิพนธ์ ได้เลือกใช้รูปแบบการรับข้อมูลแบบอินเตอร์รัพท์และใช้รูปแบบการบริหารแบบ RING หรือ CIRCULAR BUFFER ซึ่งมีรายละเอียดดังแผนภูมิสายงานในรูปที่ 4.3 โดยจะเริ่มจากการกำหนดค่าตัวแปรที่เป็นตัวชี้ตำแหน่งหน่วยความจำในสายงานอินเตอร์รัพท์ เรียกว่า PROCESS POINTER และตัวแปรที่เป็นตัวชี้ตำแหน่งหน่วยความจำในสายงานโปรแกรมหลักเรียกว่า MAIN POINTER และจะเริ่มเปิดรับข้อมูลหรือเปิดให้มีการขอบริการอินเตอร์รัพท์(รูปแบบของการเปิดให้มีการบริการขึ้นอยู่กับประเภทของการส่งข้อมูล เช่น หากเป็นแบบขนาน ใช้การเปิดให้มีการขอบริการอินเตอร์รัพท์โดยสัญญาณ STROBE เป็นสัญญาณในการขอบริการ) โดยหลักการอินเตอร์รัพท์หลังจากนี้จะเป็นหน้าที่ของโปรแกรมส่วนบริการอินเตอร์รัพท์ และโปรแกรมหลักตัวอย่าง เช่น ในกรณีที่โปรแกรมหลักทำหน้าที่ในการแปรความหมายของระหัสข้อมูล และทำหน้าที่ตามระหัสที่แปรความหมายได้นั้น

โปรแกรมหลักจะควบคุม MAIN POINTER โดยจะอ่านค่าข้อมูล แปรความหมาย ประมวลผลตามระหัสและเพิ่มค่า MAIN POINTER ในกรณีที่ MAIN POINTER มีค่าเท่ากับ PROCESS POINTER จะมีความหมายว่า โปรแกรมหลักสามารถอ่านค่ารับข้อมูลได้ครบแล้ว จำเป็นต้องรอเพื่อให้ PROCESS POINTER มากกว่า MAIN POINTER จึงเริ่มกระบวนการใหม่ สายงานของโปรแกรมทั้งสองส่วนจะต้องทำหน้าที่เพิ่มเติม คือ คอยตรวจสอบตัวชี้ตำแหน่งหน่วยความจำส่วน RING BUFFER ว่าเกินจากค่าที่กำหนดหรือไม่ หากเกินกว่าก็จะกลับไปตั้งต้นทำงานที่จุดเริ่มต้นของหน่วยความจำที่จองไว้



รูปที่ 4.3 แผนภูมิสายงานของการบริหารหน่วยความจำส่วนข้อมูลโดยใช้ RING หรือ CIRCULAR BUFFER

โครงสร้างของโปรแกรมควบคุมระบบแสดงผลแบบอิสระ

ดังที่ได้กล่าวไปแล้วเกี่ยวกับโครงสร้างของโปรแกรม ซึ่งจะประกอบด้วยสายงานโปรแกรมควบคุมหลัก (MAIN OR MAINLINE PROGRAM) และสายงานโปรแกรมบริการอินเตอร์รัพท์(INTERRUPT SERVICE ROUTINE) สำหรับในระบบแสดงผลที่ได้ออกแบบ เพื่อให้เป็นระบบแสดงผลมาตรฐานแบบ ALPHANUMERIC INTELLIGENT DISPLAY MODULE โปรแกรมควบคุมการทำงาน ก็ประกอบด้วยสายงานย่อยทั้งสองเช่นเดียวกัน โดยมีรายละเอียดของโปรแกรมในสายงานทั้งสองดังนี้

1. สายงานการบริการอินเตอร์รัพท์

1.1 สายงานการให้บริการอินเตอร์รัพท์จากภายนอกหมายเลข 0

จากเงื่อนไขของการรับข้อมูล เพื่อแสดงผลแบบอิสระที่กระทำผ่านพอร์ทแบบขนาน P1 และสัญญาณ STROBE ของระบบที่ต้องการติดต่อด้วยจะเชื่อมต่อกับขาสัญญาณอินพุท INTO และจะใช้ในการขอบริการอินเตอร์รัพท์ เมื่อระบบแสดงผลถูกขอบริการอินเตอร์รัพท์เพื่อส่งข้อมูลก็จะเข้าสู่โปรแกรมบริการ ซึ่งรูปแบบการทำงานของโปรแกรมให้บริการ จะเป็นแบบรับข้อมูลทุกตัวเข้าสู่หน่วยความจำส่วนข้อมูลที่กำหนดรูปแบบการทำงานแบบ RING BUFFER ดังแผนภูมิสายงานในรูปที่ 4.3

1.2 สายงานการให้บริการอินเตอร์รัพท์อันเกิดจากฐานเวลาของระบบ

จากส่วนของสถาปัตยกรรมทางด้านฮาร์ดแวร์ ที่ได้กล่าวไปแล้วของระบบแสดงผลแบบอิสระ ซึ่งในการแสดงผลแบบดอทเมทริกซ์นั้น จำเป็นต้องใช้การสแกนทางด้านแถวด้วยคาบเวลาในการเลือกแสดงแต่ละแถวคงที่ และมากกว่าความสามารถในการแยกการติดดับของสายตามนุษย์ได้ และฐานเวลาที่สามารถจัดการได้ง่ายโดยไม่จำเป็นต้องเพิ่มเติมวงจรภายนอก คือ ฐานเวลาภายในของไมโครคอนโทรลเลอร์ 8031 โดยฐานเวลาดังกล่าวนี้วงจรนับ หรือหารความถี่

(COUNTER/TIMER) ภายในไมโครคอนโทรลเลอร์ 8031 จะทำการสุ่มอ่านฐานเวลาของระบบทุก ๆ 1 MACHINE CYCLE ระบบแสดงผลได้เลือกใช้ความถี่ฐานเวลา 11.0592 เมกะเฮิร์ต และ 1 MACHINE CYCLE จะเท่ากับ 11.0592/12 ไมโครวินาที การทำให้เกิดการอินเตอร์รัพท์ของฐานเวลา ทำได้โดยการกำหนดค่าเริ่มต้นให้กับวงจรนับหรือหารความถี่ภายใน และสั่งเริ่มการสุ่มอ่านหรือนับค่าเมื่อค่าของการนับเพิ่มจนล้นรีจิสเตอร์ (OVERFLOW) ก็จะเกิดการอินเตอร์รัพท์ เนื่องจากในระบบแสดงผลได้เลือกใช้รูปแบบการทำงานของวงจรนับ หรือหารความถี่แบบ 16 บิท หากตั้งค่าเริ่มต้นเป็น FFFFH ระบบแสดงผลจะเกิดการอินเตอร์รัพท์ทุก 1 MACHINE CYCLE

ส่วนของสายงานบริการอินเตอร์รัพท์ จะทำหน้าที่ในการป้อนค่าเริ่มต้นของการนับให้กับวงจรนับใหม่ ก่อนที่จะทำขั้นตอนอื่นต่อไป (เพื่อให้เกิด TIMER INTERRUPT เป็นช่วง ๆ โดยตลอดด้วยคาบเวลาที่คงที่) หลังจากนั้นสายงานการบริการอินเตอร์รัพท์ จะคำนวณหาจุดเริ่มต้นในตำแหน่งของหน่วยความจำส่วน DISPLAY BUFFER ขณะนั้นซึ่งขึ้นอยู่กับแถวที่ต้องการแสดงและจำนวนหน่วยความจำที่ต้องใช้ใน 1 แถว เช่น

ในกรณีที่ต้องการแสดงผลในแถวที่ 3 ของภาคแสดงผลที่ใช้เอาท์พุทพอร์ทควบคุมทางด้านคอลัมน์ (จะเท่ากับจำนวนตำแหน่งหน่วยความจำที่ต้องใช้ต่อ 1 แถวของ DISPLAY BUFFER) จำนวน 6 ตัวก็สามารถคำนวณหาตำแหน่งหน่วยความจำ DISPLAY BUFFER เริ่มต้นของแถวนี้ โดย

ตำแหน่งหน่วยความจำ = (หมายเลขแถวที่ต้องการแสดงผล)X(จำนวนเอาท์พุทพอร์ท)

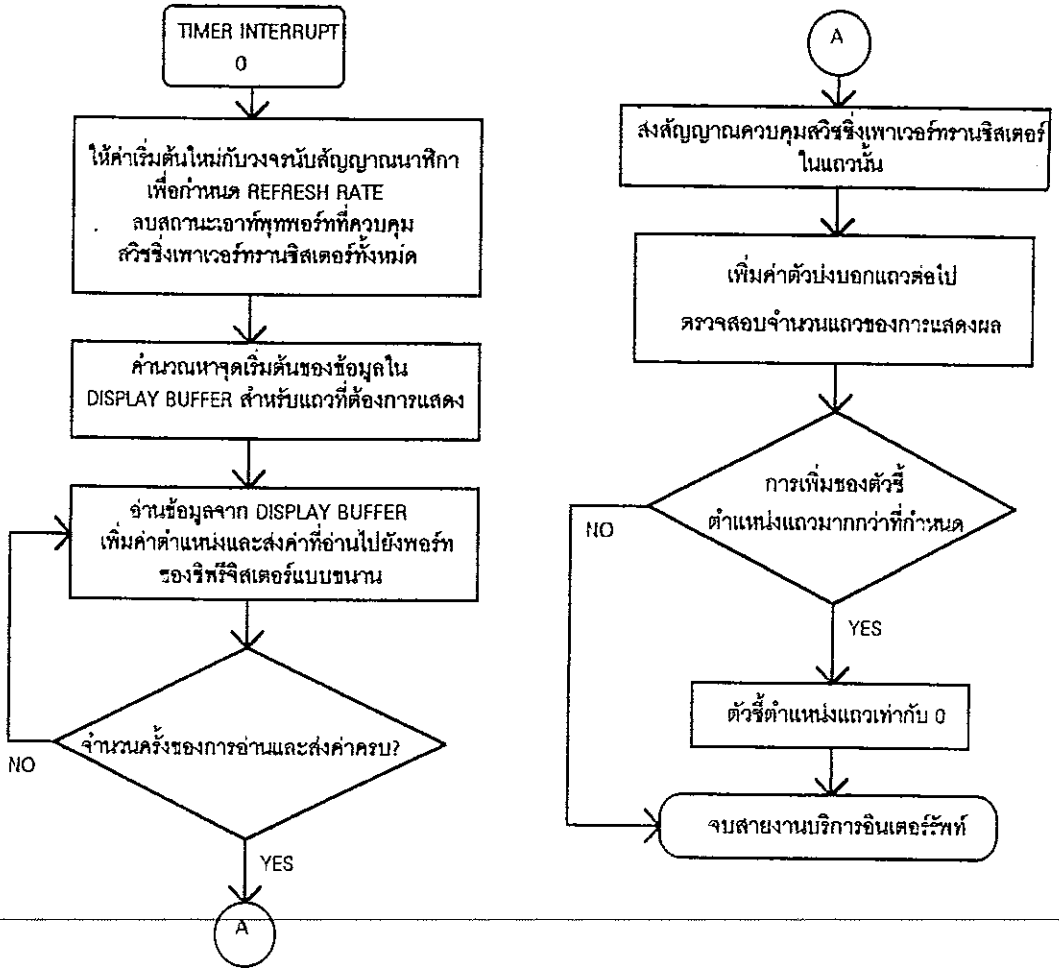
โปรแกรมส่วนบริการจะอ่านค่าจากตำแหน่งเริ่มต้นนี้ส่งออกเอาท์พุทพอร์ทที่มีโครงสร้างของฮาร์ดแวร์ในรูปแบบของชิฟรีจิสเตอร์แบบขนาน จากนั้นจะเพิ่มค่าของตำแหน่งหน่วยความจำของข้อมูลในส่วน DISPLAY BUFFER อ่านค่าข้อมูลและกระทำขั้นตอนซ้ำจนครบ 6 ครั้ง (เท่ากับจำนวนพอร์ทที่ใช้)

เมื่อข้อมูลในส่วน COLUMN ได้ส่งออกไปยังเอาท์พุทพอร์ทแล้ว จะเข้าสู่ขั้นตอนของการส่งสัญญาณเพื่อควบคุมสวิทชิ่งเพาเวอร์ทรานซิสเตอร์ทางด้านแถว ซึ่งรายละเอียดของสัญญาณได้กล่าวไปแล้วในบทที่ 3 สำหรับขั้นตอนที่ได้กล่าวมาแล้วนั้น สามารถสรุปเป็นแผนภูมิสายงานได้ตามรูปที่ 4.4

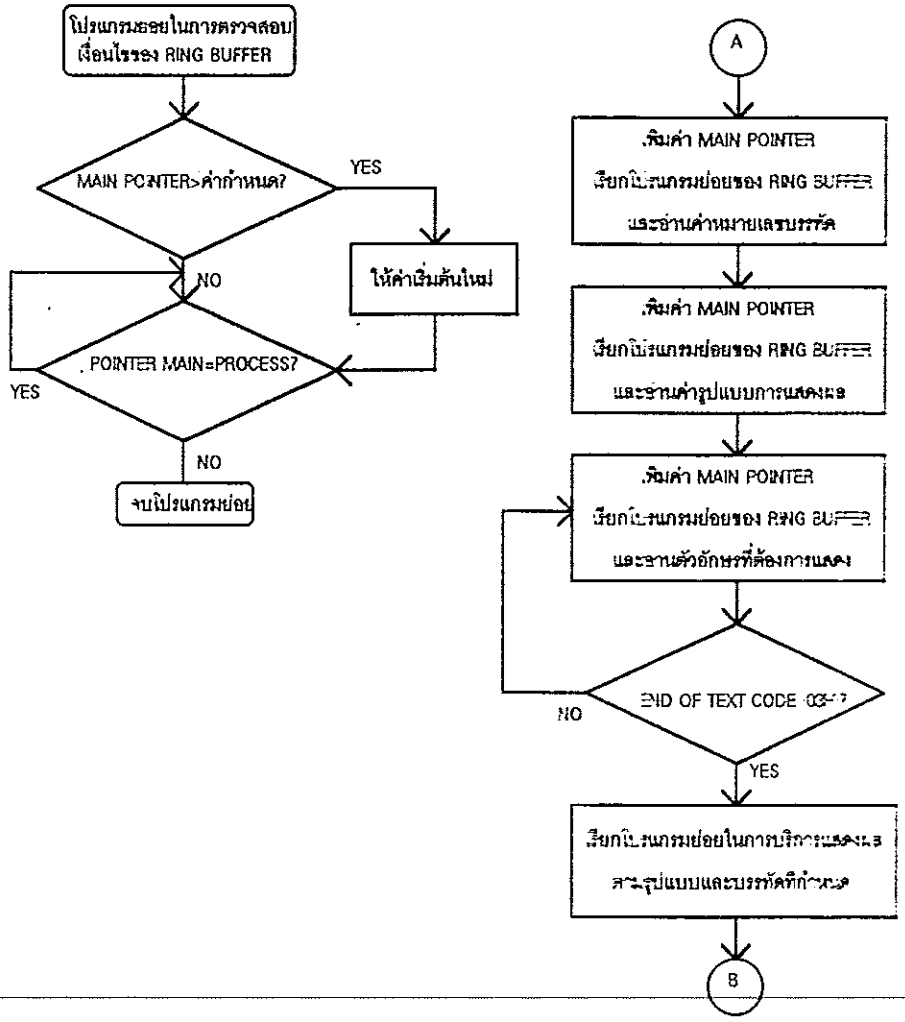
2. สายงานโปรแกรมควบคุมหลัก

ในสายงานของโปรแกรมควบคุม จะทำการกำหนดค่าเริ่มต้นให้ตัวแปรและรีจิสเตอร์ต่าง ๆ รวมทั้งค่าเริ่มต้น เลือกรูปแบบการทำงานของการกำเนิดอินเตอร์รัพท์ของฐานเวลาและเปิดให้มีการนับสัญญาณ เพื่อทำให้เกิดอินเตอร์รัพท์ครั้งแรกและหลังจากนั้น การอินเตอร์รัพท์จะเกิดเรื่อย ๆ ด้วยคาบเวลาที่คงที่

หลังจากนั้นโปรแกรมควบคุม จะทำการกำหนดจุดเริ่มต้นของ RING BUFFER และให้ค่าเริ่มต้นของ MAIN POINTER และ PROCESS POINTER ก่อนที่จะเปิดให้มีการขอบริการอินเตอร์รัพท์จากภายนอก เพื่อส่งข้อมูลในการแสดงผล โปรแกรมควบคุมการทำงานหลักจะรอเพื่อถอดระหัส และนำข้อมูลไปใช้โดยการเปรียบเทียบค่า MAIN POINTER และ PROCESS POINTER หากค่าทั้งสองยังคงเท่ากันก็แสดงว่ายังไม่มีข้อมูลถูกส่งเข้าสู่หน่วยความจำส่วน RING BUFFER และจะทำการถอดระหัสชุดคำสั่งหลังจากที่มีข้อมูลเข้าสู่หน่วยความจำแล้ว โดยรูปแบบการทำงานในสายงานโปรแกรมหลักแสดงในรูปที่ 4.5



รูปที่ 4.4 แผนภูมิสายงานการบริการอินเตอร์รัพท์ส่วนแสดงผลข้อมูล



รูปที่ 4.5 แผนภูมิสายงานการถอดระหัสเพรมข้อมูลเพื่อเลือกบรรทัด
และรูปแบบการแสดงผล

โครงสร้างของโปรแกรมควบคุมระบบเครือข่ายย่อย

จากโครงสร้างและขีดความสามารถของระบบเบื้องต้น กำหนดให้ระบบเครือข่ายย่อยถูกกำหนดรูปแบบการทำงานจากระบบควบคุมหลักในสองรูปแบบคือ

- รูปแบบเวลาจริง สัญญาณในรูปของระหัสแอสกีในการกระตุ้นการสุ่มอ่านค่าสัญญาณถูกส่งจากระบบควบคุมหลักไปยังระบบเครือข่ายย่อยทุกตัวที่ต่อในเครือข่าย จากนั้นระบบควบคุมหลักจะทำการอ่านค่าข้อมูล (ที่เวลาเดียวกันของทุกระบบเครือข่าย) จากระบบเครือข่ายย่อยแต่ละตัวเพื่อจัดเก็บ

- ในรูปแบบไม่เป็นเวลาจริง ระบบเครือข่ายย่อยจะใช้ฐานเวลาในการสุ่มอ่านสัญญาณหรือตัวแปรที่ต้องการเก็บค่าเป็นของตัวเอง และเป็นอิสระจากกันภายใต้ระหัสการสั่งเริ่มทำงานและการเลือกฐานเวลาในการสุ่มอ่าน ที่มาจากระบบควบคุมหลัก เมื่อครบตามช่วงเวลาที่ต้องการแล้วระบบควบคุมหลักจะสั่งหยุดการสุ่มอ่านสัญญาณ และจะขอบริการอ่านข้อมูลจากหน่วยความจำของระบบเครือข่ายย่อยแต่ละตัว

จากเงื่อนไขคร่าว ๆ ของการทำงานของระบบเครือข่ายย่อย ทำให้สามารถกำหนดหน้าที่หลักในการทำงานของระบบเครือข่ายย่อยได้ 4 ประการ คือ

- การรับคำสั่งงานจากระบบควบคุมการทำงานหลัก

- การอ่านค่าข้อมูลจากวงจร เปลี่ยนรูปแบบสัญญาณจากอนาลอกเป็นดิจิตอล ตามรูปแบบที่กำหนด

- การจัดการส่งข้อมูลเข้าสู่เครือข่ายหรือเก็บข้อมูลไว้ในหน่วยความจำก่อนที่จะมีการเรียกอ่านข้อมูล

- การกำหนดค่าของการสุ่มอ่านข้อมูล จากวงจรแปลงสัญญาณจากอนาลอกเป็นดิจิตอล ให้กับวงจรนับฐานเวลา

โครงสร้างของโปรแกรมควบคุมในส่วนนี้ ประกอบด้วยสายงานโปรแกรมทั้งในส่วนโปรแกรมหลักและโปรแกรมบริการอินเตอร์รัพท์ โดยสายงานโปรแกรมบริการอินเตอร์รัพท์เกิดจาก 3 แหล่งกำเนิดคือ

- อินเตอร์รัพท์จากภายนอกหมายเลข 0 เพื่อรับข้อมูลจากวงจรแปลง

สัญญาณอนาลอกเป็นดิจิตอลเมื่อการแปลงค่าข้อมูลเสร็จสิ้นแล้ว

- อินเตอร์รัพท์จากการรับหรือส่งข้อมูลแบบอนุกรมเพื่อรับคำสั่งงานและส่งข้อมูล

- อินเตอร์รัพท์จากฐานเวลา เพื่อใช้ในการสุ่มอ่านค่าสัญญาณโดยใช้ TIMER 0 ในรูปแบบการการทำงานแบบไม่เป็นเวลาจริง

โดยกำหนดให้มีความแตกต่างของลำดับความสำคัญ ในการให้บริการอินเตอร์รัพท์จากแหล่งกำเนิดต่าง ๆ (INTERRUPT PRIORITY) ในส่วนของการบริหารหน่วยความจำส่วนข้อมูลของระบบได้จัดแบ่งหน่วยความจำออกเป็น 2 ส่วนคือ

- หน่วยความจำรูปแบบ RING BUFFER ขนาด 256 ไบท์ ทำหน้าที่ในการรับกลุ่มคำสั่งงานในรูปแบบของเฟรมข้อมูลที่มีระหัสนำหน้าและระหัสปิดท้าย เป็นหน่วยความจำส่วนของหน่วยความจำที่รับค่าข้อมูล จากโปรแกรมบริการอินเตอร์รัพท์แบบอนุกรม

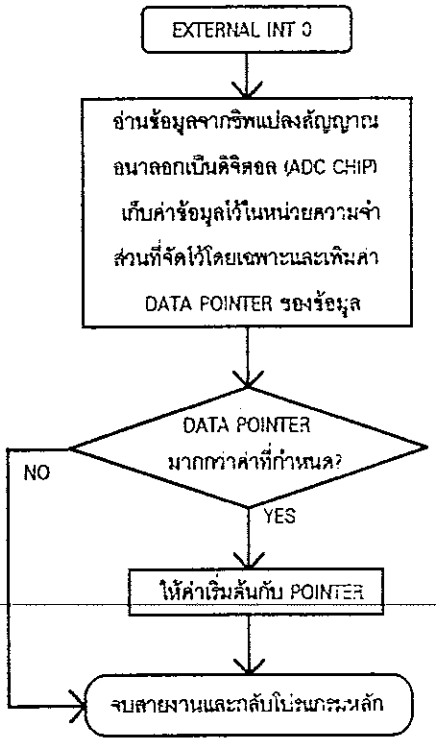
- หน่วยความจำส่วนที่เหลือจัดรูปแบบเป็น RING BUFFER ทำหน้าที่ในการเก็บข้อมูลที่อ่านค่าได้ เพื่อรอการเรียกอ่านค่าจากระบบควบคุมการทำงานหลัก ซึ่งเป็นส่วนของหน่วยความจำที่รับค่าข้อมูลจากโปรแกรมบริการอินเตอร์รัพท์จากภายนอกหมายเลข 0 รายละเอียดของสายงานโปรแกรมเป็นดังนี้

1. สายงานบริการอินเตอร์รัพท์ แบ่งย่อยเป็น 3 สายงานโปรแกรมเป็นดังนี้

   1.1 สายงานให้บริการอินเตอร์รัพท์จากภายนอกหมายเลข 0 (EXTERNAL INTERRUPT 0 SERVICE ROUTINE)

จากรูปแบบของฮาร์ดแวร์ส่วนเชื่อมโยงเติมของระบบเครือข่ายย่อยซึ่งมีวงจรในการแปลงสัญญาณจากอนาลอกเป็นดิจิตอล ในการแปลงสัญญาณของวงจรนี้เมื่อระบบควบคุมสั่งให้มีการเริ่มแปลงสัญญาณโดยการกระตุ้นขาสัญญาณ START OF CONVERSION (หรือส่งสัญญาณเลือกชิพแปลงสัญญาณอนาลอกเป็นดิจิตอลซึ่งในตารางการแปลงสรรหน่วยความจำและพอร์ทของระบบเครือข่ายย่อยกำหนดให้เป็น 8XXXH) วงจรแปลงสัญญาณจะเริ่มทำงานโดยการประมวลผลสัญญาณอินพุท และแปลงสัญญาณเป็นข้อมูลแบบดิจิตอล และเมื่อกระบวนการเสร็จสิ้นก็จะส่งสัญญาณที่ขาสัญญาณ END OF CONVERSION (INTR ของ ADC0804) ซึ่งเป็นสัญญาณที่ใช้ในการขอบริการ

อินเตอร์รัพท์จากภายนอกหมายเลข 0 โปรแกรมในสายงานอินเตอร์รัพท์ จะส่งสัญญาณเพื่อเลือกชิพไปยังตำแหน่งของวงจรแปลงสัญญาณอีกครั้งและอ่านค่าข้อมูลเพื่อเก็บไว้ในหน่วยความจำส่วนข้อมูลที่จัดรูปแบบการทำงานเป็น RING BUFFER สำหรับการเก็บข้อมูลจากโปรแกรมบริการนี้โดยเฉพาะเพื่อรอการส่งข้อมูลเมื่อระบบควบคุมหลักต้องการอ่านค่าข้อมูล หลังจากนั้นโปรแกรมบริการจะเพิ่มค่าของตัวชี้ตำแหน่งหน่วยความจำตำแหน่งต่อไป และหากค่านี้เกินไปจากส่วนเก็บข้อมูลการอ่านค่า ก็จะตั้งค่าเริ่มต้นใหม่ แผนภูมิสายงานของการให้บริการอินเตอร์รัพท์จากภายนอกแสดงในรูป 4.6



รูปที่ 4.6 แผนภูมิสายงานของการให้บริการอินเตอร์จากภายนอกหมายเลข 0 ของระบบเครือข่ายย่อย

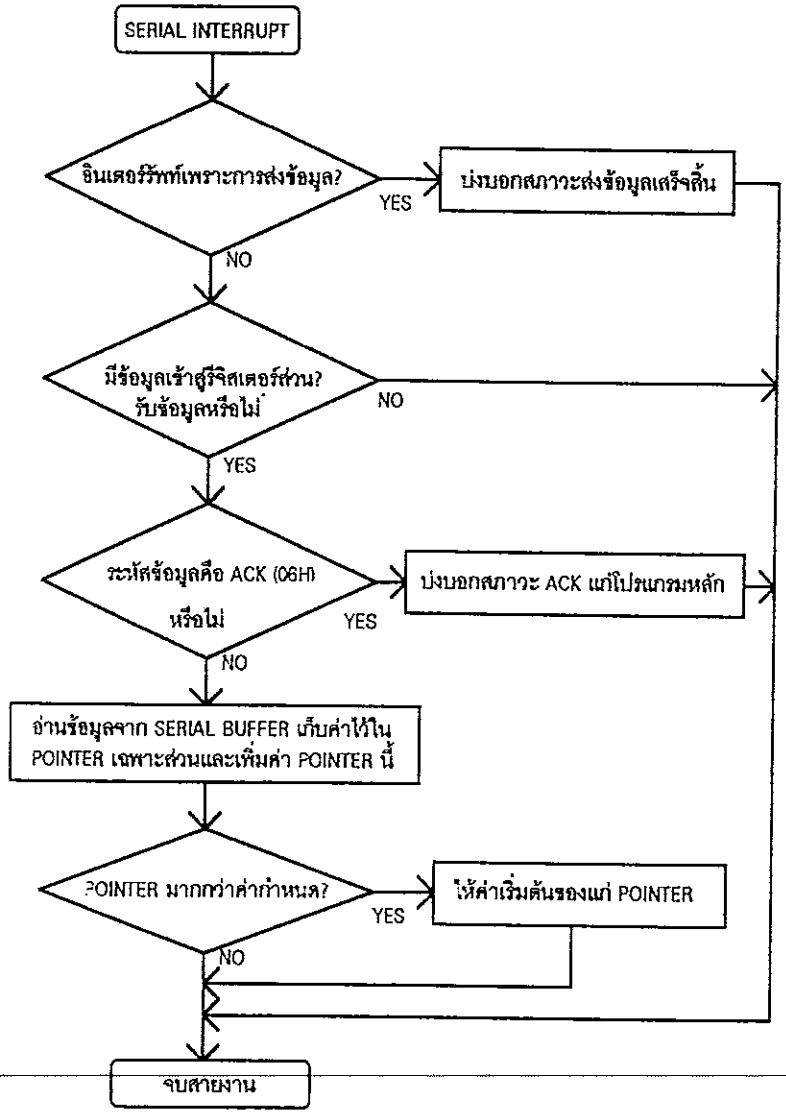
1.2 สายงานให้บริการอินเตอร์รัพท์แบบอนุกรม (SERIAL INTERRUPT SERVICE ROUTINE)

สายงานให้บริการอินเตอร์รัพท์แบบอนุกรมจะทำหน้าที่ในการรับข้อมูลซึ่งอยู่ในรูปของเฟรมข้อมูล (DATA FRAME) หรือตรวจสอบว่าระหัสที่ถูกส่งมานั้นเป็นระหัสตอบสนองการส่งข้อมูล (ACK : ระหัส 06H) และในกรณีนี้จะไม่มีการจัดเก็บค่าลงในหน่วยความจำ สำหรับเฟรมข้อมูลจะถูกถอดระหัสในการสั่งงานโดยส่วนของสายงานโปรแกรมควบคุมหลัก หน่วยความจำส่วนเก็บค่าจะมีขนาด 256 ไบท์ ในส่วนสูงสุดของหน่วยความจำที่มีอยู่ในระบบเครือข่ายย่อย (256 ไบท์บนสุดของหน่วยความจำขนาด 32 กิโลไบท์) นอกจากหน้าที่ในการรับข้อมูลแล้วในสายงานบริการนี้ยังตรวจสอบสถานภาพการส่งข้อมูลออกจากระบบเครือข่ายย่อยทางพอร์ทอนุกรมและแจ้งสถานภาพให้กับสายงานโปรแกรมควบคุมหลักทราบ แผนภูมิสายงานในส่วนนี้แสดงในรูปที่ 4.7

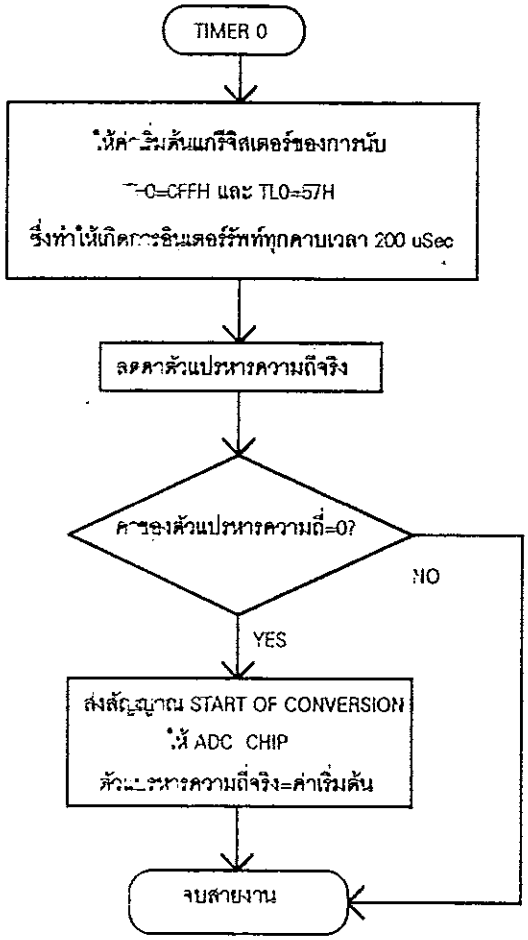
1.3 สายงานการให้บริการอินเตอร์รัพท์จากฐานเวลาของระบบ

สายงานส่วนนี้ถูกเรียกใช้ ในกรณีที่ระบบควบคุมหลักเลือกรูปแบบทำงานแบบไม่เป็นเวลาจริง โดยเมื่อสายงานโปรแกรมหลักถอดระหัสการเลือกรูปแบบการทำงาน และ ระหัสส่งค่าตัวหารความถี่มูลฐานได้แล้ว จะส่งค่าตัวหารความถี่มูลฐานซึ่งมีคาบเวลาคงที่ 200 ไมโครวินาที (ค่าคาบเวลามาตรฐานนี้ได้จากการตั้งค่าตัวนับเริ่มต้นให้กับรีจิสเตอร์ด้วยค่า FF5AH ที่ความถี่สัญญาณของระบบ 11.059 MHz เลือกรูปแบบการรับแบบ 16 บิท) ในรูปแบบของตัวแปรให้กับตัวแปรสำรองของการหารความถี่ ซึ่งจะส่งค่าให้กับตัวแปรในการหารความถี่จริงทุกครั้งเมื่อค่าของตัวแปรที่ใช้หารความถี่จริงถูกลดลงจนเป็น 0 และก่อให้เกิดการสั่งงานชิพแปลงสัญญาณอนาลอกเป็นดิจิตอลแล้ว

การสั่งงานให้เกิดกระบวนการของสายงานบริการอินเตอร์รัพท์ จากฐานเวลาจะเกิดขึ้นเมื่อโปรแกรมในสายงานหลักถอดระหัสเฟรมข้อมูลที่สั่งให้เริ่มทำงานและการสิ้นสุดของการยอมให้เกิดการบริการส่วนนี้ ก็เกิดจากการถอดระหัสเฟรมข้อมูลเช่นเดียวกัน สำหรับสายงานแผนภูมิสายงานในส่วนนี้แสดงในรูปที่ 4.8



รูปที่ 4.7 แผนภูมิสายงานของโปรแกรมบริการอินเตอร์รัพท์แบบอนุกรม ของระบบเครือข่ายย่อย

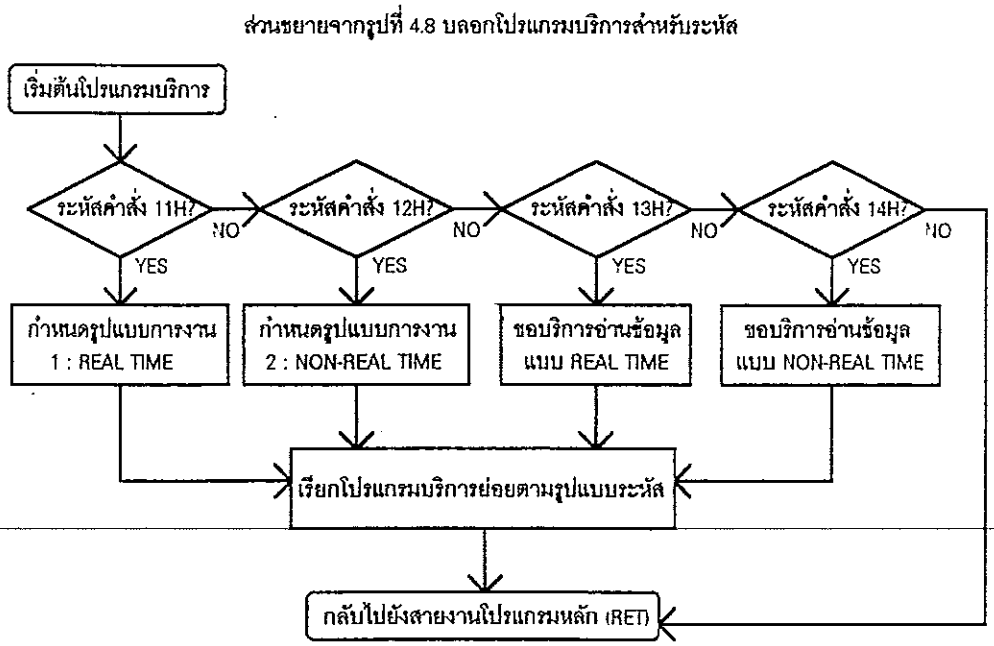


รูปที่ 4.8 สายงานโปรแกรมบริการอินเตอร์รัพท์จากฐานเวลา
สำหรับระบบเครือข่ายย่อย

2. สายงานโปรแกรมหลัก

กลุ่มคำสั่งส่วนต้นของสายงานโปรแกรมหลัก จะกำหนดจุดเริ่มต้นของหน่วยความจำในแต่ละส่วน คือ หน่วยความจำส่วน RING BUFFER ของการรับ

คำสั่ง และหน่วยความจำส่วน RING BUFFER ของการรับข้อมูลจากกระบวนการให้กับตัวแปรที่เก็บค่าตัวชี้ตำแหน่งหน่วยความจำทั้งสอง จากนั้นจะตรวจสอบรูปแบบการทำงานที่ได้ส่งมาจากระบบควบคุมการทำงานหลัก และจัดรูปแบบการทำงานหลังจากนั้นจะเปิดให้มีการขอบริการอินเตอร์รัพท์ ภาระงานหลักของสายงานนี้จะอยู่ที่การถอดระหัสข้อมูลที่ได้รับจากหน่วยความจำส่วนอนุกรม และทำหน้าที่ตามรูปแบบคำสั่งที่ได้ รูปแบบของแผนภูมิสายงานจึงคล้ายคลึงกับส่วนถอดระหัสเลือกรูปแบบการแสดงผลของระบบแสดงผลแบบอิสระที่ได้แสดงไปแล้วในรูปแบบที่ 4.4 และมีส่วนขยายของการเรียกโปรแกรมบริการโดยได้แสดงตัวอย่างการถอดระหัสในบางรูปแบบส่วนนี้ในรูปที่ 4.9



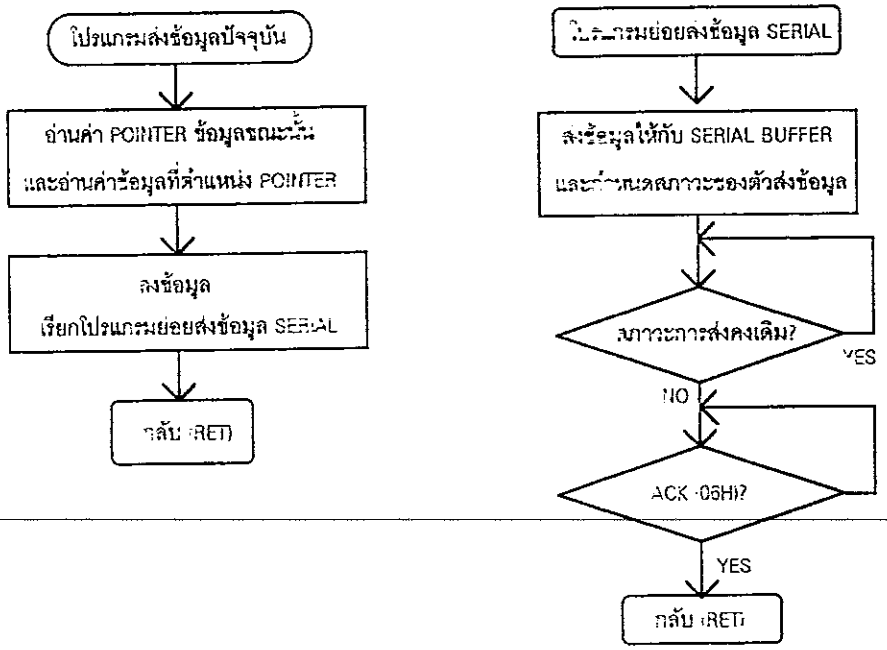
รูปที่ 4.9 แผนภูมิสายงานส่วนขยายสำหรับโปรแกรมควบคุมหลักของระบบเครือข่ายย่อย

นอกจากนั้นในการถอดระหัสยังมีบางระหัสที่เป็นคำสั่งตรง และไม่จำเป็นต้องอ่านค่าอื่นเพิ่มเติม เช่น

- คำสั่งสุ่มอ่านสัญญาณใช้ระหัสคำสั่งเป็น OEH ซึ่งมีผลในการสั่งให้ชิพแปลงสัญญาณอนาลอกเป็นดิจิตอลเริ่มแปลงสัญญาณทันที

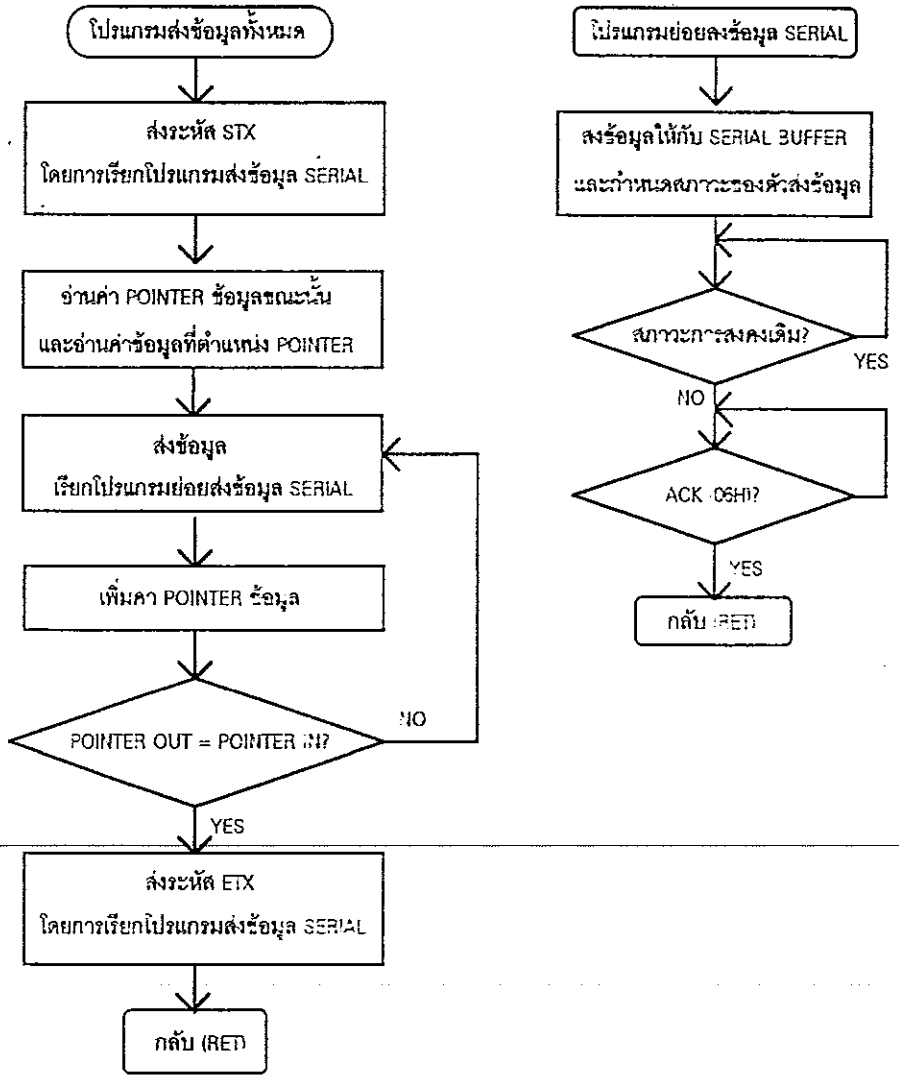
- คำสั่งขออ่านข้อมูลปัจจุบันของการแปลงสัญญาณ ใช้ระหัสคำสั่ง OFH ซึ่งจะมีผลในการอ่านค่าข้อมูลที่ตำแหน่งตัวชี้ข้อมูลปัจจุบัน เพื่อส่งให้กับระบบควบคุมหลัก

สำหรับแผนภูมิสายงานของการส่งค่าข้อมูลในปัจจุบัน ให้กับระบบควบคุมการทำงานหลักแสดงในรูปที่ 4.9 โดยมีข้อบ่งชี้การเรียกใช้ก็ คือ ระบบควบคุมหลักจำเป็นต้องส่งคำสั่งขออ่านข้อมูลปัจจุบัน (ระหัส OFH) ก่อนหน้านี้แล้ว



รูปที่ 4.10 แผนภูมิสายงานสำหรับโปรแกรมควบคุมหลักของระบบเครือข่ายย่อยในการส่งข้อมูลปัจจุบัน

สำหรับในกรณีที่มีการร้องขออ่านข้อมูลในหน่วยความจำส่วนข้อมูล ของระบบเครือข่ายย่อยจากจุดเริ่มต้นจนถึงค่าปัจจุบัน ในรูปแบบของการเก็บข้อมูลไม่เป็นเวลาจริงนั้น สายงานโปรแกรมหลักมีโปรแกรมย่อยเพื่อจัดการดังกล่าว โดยมีแผนภูมิสายงานแสดงในรูปที่ 4.11 โดยการเรียกใช้โปรแกรมย่อยในส่วนนี้ โปรแกรมหลัก จำเป็นต้องได้รับระหัสในรูปของเฟรมข้อมูลที่เป็นคำสั่งร้องขอบริการจากระบบควบคุมหลักก่อน



รูปที่ 4.11 แผนภูมิสายงานสำหรับโปรแกรมควบคุมหลักของระบบเครือข่ายย่อยในการส่งข้อมูลในหน่วยความจำจากจุดเริ่มต้นจนถึงปัจจุบัน

โครงสร้างของโปรแกรมควบคุมระบบควบคุมการทำงานหลัก

เนื่องจากโปรแกรมในส่วนนี้ เป็นส่วนที่ควบคุมกระบวนการทุกอย่างของระบบเก็บข้อมูลระยะไกลจึงมีความซับซ้อนของโครงสร้างโปรแกรมและรายละเอียดการทำงานมาก สายงานของโปรแกรมควบคุม จะมีสายงานของการให้บริการอินเตอร์รัพท์จากสองแหล่งมาเกี่ยวข้อง โดยอ้างอิงหน้าที่การทำงานโดยทั่วไปของระบบควบคุมการทำงานหลักจากที่ได้กล่าวไปแล้วในบทที่ 3 คือ

- ทำหน้าที่ในการบริการเครือข่าย
- ทำหน้าที่ในการจัดการแสดงผล
- ทำหน้าที่ในการติดต่อกับระบบเก็บข้อมูลส่วนขยาย
- ทำหน้าที่ในการจัดการคีย์บอร์ด
- ทำหน้าที่ในการจัดการนาฬิกาเวลาจริง

สายงานการบริการให้อินเตอร์รัพท์เข้ามาเกี่ยวข้อง ในส่วนบริการเครือข่าย ส่วนจัดการนาฬิกาเวลาจริง ส่วนกำเนิดสัญญาณในการสุ่มอ่านข้อมูลในรูปแบบการทำงานแบบเวลาจริง และส่วนของหน้าที่การทำงานที่เหลือเป็นการจัดการของสายงานโปรแกรมหลัก โดยสายงานในส่วนบริการเครือข่ายจะครอบครองหน่วยความจำหลักแทบทั้งหมดของระบบจะเหลือส่วนของ หน่วยความจำที่มีรูปแบบเป็น RING BUFFER ไว้ให้กับสายงานโปรแกรมหลักขนาด 256 ไบท์ สำหรับรายละเอียดของแต่ละสายงานเป็นดังนี้

1. **สายงานบริการอินเตอร์รัพท์**

สายงานการให้บริการอินเตอร์รัพท์แบ่งเป็น 3 สายงานจากแหล่งเรียกขอบริการต่างกันดังที่ได้กล่าวมาแล้ว

1.1 สายงานการให้บริการอินเตอร์รัพท์จากภายนอกหมายเลข 0 (EXTERNAL INTERRUPT 0 SERVICE ROUTINE)

สายงานการบริการอินเตอร์รัพท์ในส่วนนี้ เกิดจากการเรียกขอบริการของนาฬิกาเวลาจริงดังที่กล่าวไปแล้วในบทที่ 3 หัวข้อ 3.8.4 เกี่ยวกับรูปแบบของการเรียกบริการของชิพนาฬิกาเวลาจริง โดยได้จัดรูปแบบการทำงานให้การเรียก

ขอบริการเกิดขึ้นทุก ๆ ช่วงเวลาที่ 1 วินาทีซึ่งเป็นช่วงเวลาที่เกิดจากการประมวลผลของชิพนาฬิกาเอง ในการให้บริการของสายงานจะเริ่มจากการอ่านค่ารีจิสเตอร์เวลาต่าง ๆ ภายในชิพนาฬิกาจริง (โดยการกำหนดระหัสของบัสแอดเดรส) ผ่านทางบัสข้อมูลของระบบ และส่งเข้าสู่ระบบแสดงผลอิสระที่เชื่อมโยงอยู่กับพอร์ทข้อมูลแบบขนานทันทีไม่ผ่านการจัดกระทำของสายงานโปรแกรมหลัก ซึ่งจะลดขั้นตอนในการประมวลผลของสายงานโปรแกรมหลักลงได้มาก แต่จำเป็นต้องกำหนดเงื่อนไขของการยอมตอบรับให้บริการ ให้อยู่ภายใต้การควบคุมของสายงานของโปรแกรมหลักเพื่อขจัดความสับสนของการส่งข้อมูลเพื่อแสดงผล เช่น ในขณะที่สายงานโปรแกรมหลักกำลังทำการส่งค่าบางอย่างให้กับระบบแสดงผล เพื่อสื่อความหมายกับผู้ใช้งาน จำเป็นต้องมีการยกเลิกการให้บริการในสายงานบริการอินเตอร์รัพท์จากชิพนาฬิกาเวลาจริงหลังจากที่ขั้นตอนในสายงานโปรแกรมหลักเสร็จสิ้น ก็พร้อมที่จะยอมให้มีการตอบรับการขอบริการได้ สำหรับส่วนของแผนภูมิสายงานของบริการให้บริการอินเตอร์รัพท์จากภายนอกหมายเลข 0 แสดงในรูป ที่ 4.12

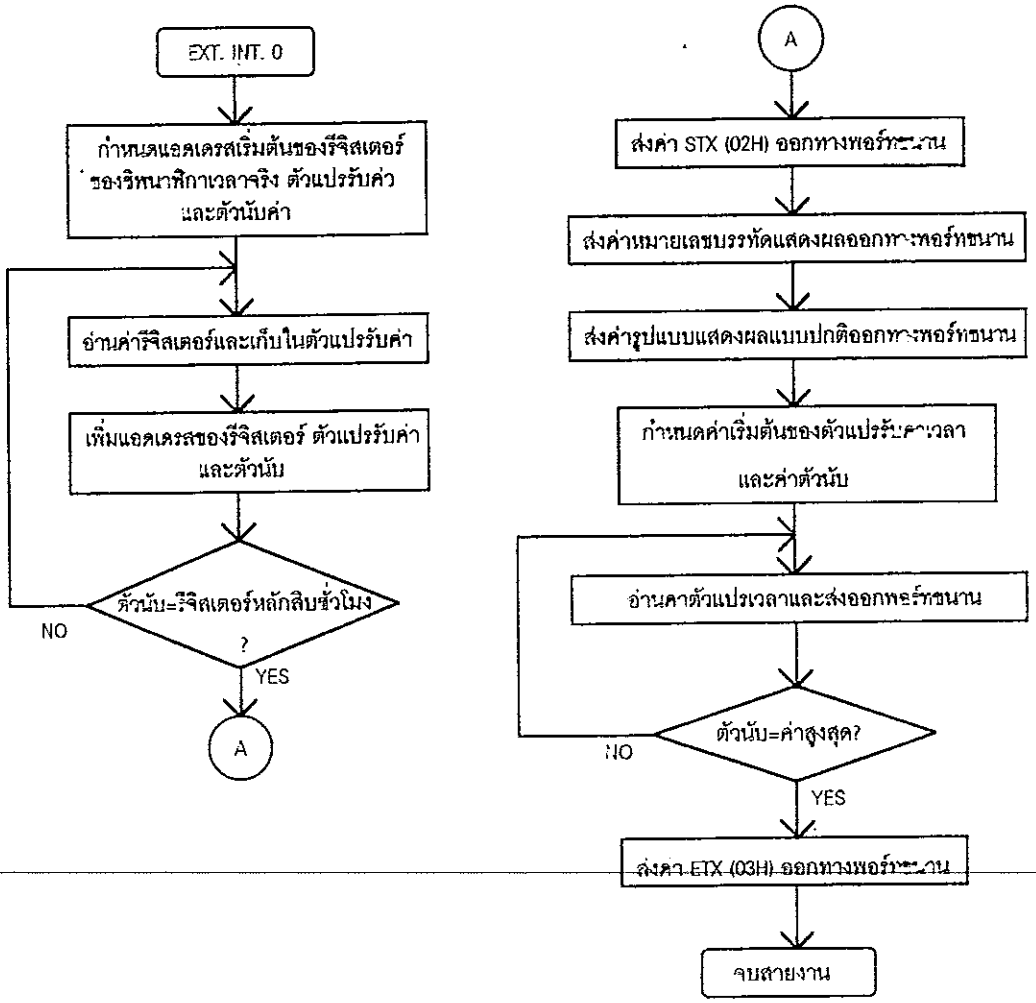
1.2 สายงานการให้บริการอินเตอร์รัพท์แบบอนุกรม (SERIAL INTERRUPT SERVICE ROUTINE)

สายงานการให้บริการอินเตอร์รัพท์แบบออนุกรมจะทำหน้าที่ในการรับค่าข้อมูล ที่ส่งมาจากระบบเครือข่ายย่อยหรือจากไมโครคอมพิวเตอร์ เพื่อถอดระหัสสำหรับเลือกรูปแบบการทำงานที่เหมาะสม ข้อมูลที่ถูกส่งเข้าสู่ระบบควบคุมการทำงานหลักซึ่งเป็นคำสั่งงานจากไมโครคอมพิวเตอร์จะถูกจัดเก็บไว้ในหน่วยความจำแบบ RING BUFFER ขนาด 256 ไบท์เพื่อรอการถอดระหัสโดยสายงานโปรแกรมหลัก การบริหารหน่วยความจำโดยสายงานโปรแกรมบริการอินเตอร์รัพท์ จะใช้รูปแบบของ RING BUFFER ซึ่งรายละเอียดของแผนผังสายงานได้แสดงไปแล้วในตอนต้น

1.3 สายงานการให้บริการอินเตอร์รัพท์จากฐานเวลา(TIMER INTERRUPT SERVICE ROUTINE)

ในระบบเก็บข้อมูลระยะไกลที่ได้ออกแบบและสร้างขึ้นในรูปแบบของการอ่านข้อมูลแบบเวลาจริงจำเป็นต้องมีการกำหนดช่วงเวลาในการอ่านข้อมูลจากระบบ

เครือข่ายย่อยซึ่งฐานเวลาในการกำหนดไม่สามารถใช้ฐานเวลาจากระบบเครือข่ายย่อยแต่ละตัวได้ เนื่องจากจะทำให้เกิดปัญหาของความแตกต่างของเวลาในการสุ่มอ่านข้อมูลแต่ละครั้ง ผลของข้อมูลที่อ่านได้จะมีความผิดพลาด



รูปที่ 4.12 แผนภูมิสายงานการบริการอินเตอร์รัพท์แก่ระบบนาฬิกาเวลาจริงของระบบควบคุมการทำงานหลัก

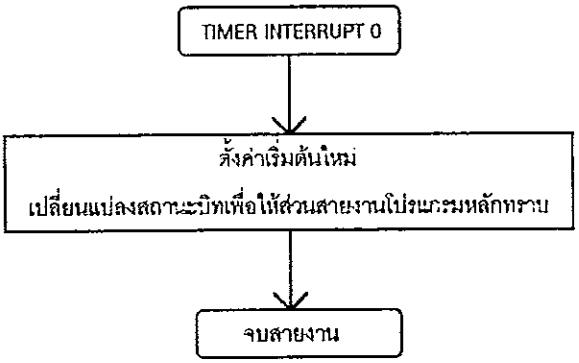
ดังนั้น จึงจำเป็นต้องมีฐานเวลาหลักของระบบทั้งหมด ซึ่งเป็นอิสระจากระบบเครือข่ายย่อย ซึ่งในที่นี้ได้เลือกการสร้างฐานเวลาในระบบควบคุมการทำงานหลัก เพื่อใช้ในการควบคุมการสุ่มอ่านข้อมูลจากระบบเครือข่าย ในรูปแบบการทำงานเวลาจริง โดยในระบบที่ใช้งานจริงได้เลือกโปรแกรมการทำงานของวงจรนับและหารความถี่ของไมโครคอนโทรลเลอร์ 8031 (TIMER/COUNTER) จากฐานเวลามาตรฐานจากภายนอกความถี่ 60 เฮิร์ต (ซึ่งเป็นสองเท่าอัตราสูงสุดในการสุ่มอ่านค่าข้อมูลการทดลองในรูปแบบการทำงานแบบเวลาจริง) โดยแหล่งกำเนิดฐานเวลาได้จากผลึกคริสตอลและชิพจัดการ MM5369AA/N ซึ่งความถี่ที่ได้มีความเที่ยงตรงสูง สัญญาณจากแหล่งกำเนิดฐานเวลาจะถูกป้อนเข้าสู่ขารับสัญญาณความถี่อินพุทอ้างอิง (ขา TO) ของไมโครคอนโทรลเลอร์ 8031 ค่าเริ่มต้นที่ผู้ใช้งานกำหนดซึ่งจะเป็นจำนวนเท่าของ 60 เฮิร์ต เช่น สุ่มข้อมูลด้วยคาบเวลาเป็น 1/30 วินาที 1/20 วินาที เป็นต้น ตัวอย่างของการตั้งค่าเริ่มต้นให้กับรีจิสเตอร์ตัวนับฐานเวลา และอัตราของการสุ่มอ่านข้อมูลแสดงในตารางที่ 4.2

ตาราง 4.2 ตารางแสดงตัวอย่างค่าเริ่มต้นของวงจรนับฐานเวลา 60 เฮิร์ต และอัตราการสุ่มอ่านข้อมูลในรูปแบบการทำงานของระบบเก็บข้อมูลแบบเวลาจริง

ค่าเริ่มต้น	อัตราการสุ่มอ่าน	ค่าเริ่มต้น	อัตราการสุ่มอ่าน
FFFFH	60 Hz	FFC4H	1 Hz
FFFEH	30 Hz	FF88H	1/2 Hz
FFFDH	20 Hz	FED4H	1/5 Hz
FFFCH	10 Hz		

ค่าเริ่มต้นที่สอดคล้องกับอัตราการสุ่มอ่านข้อมูลเหล่านี้จะถูกเลือกโดยผู้ใช้งาน และโปรแกรมหลักจะอ่านตัวหารและป้อนให้กับรีจิสเตอร์ตัวนับ (ในกรณีนี้เป็น THO และ TLO)

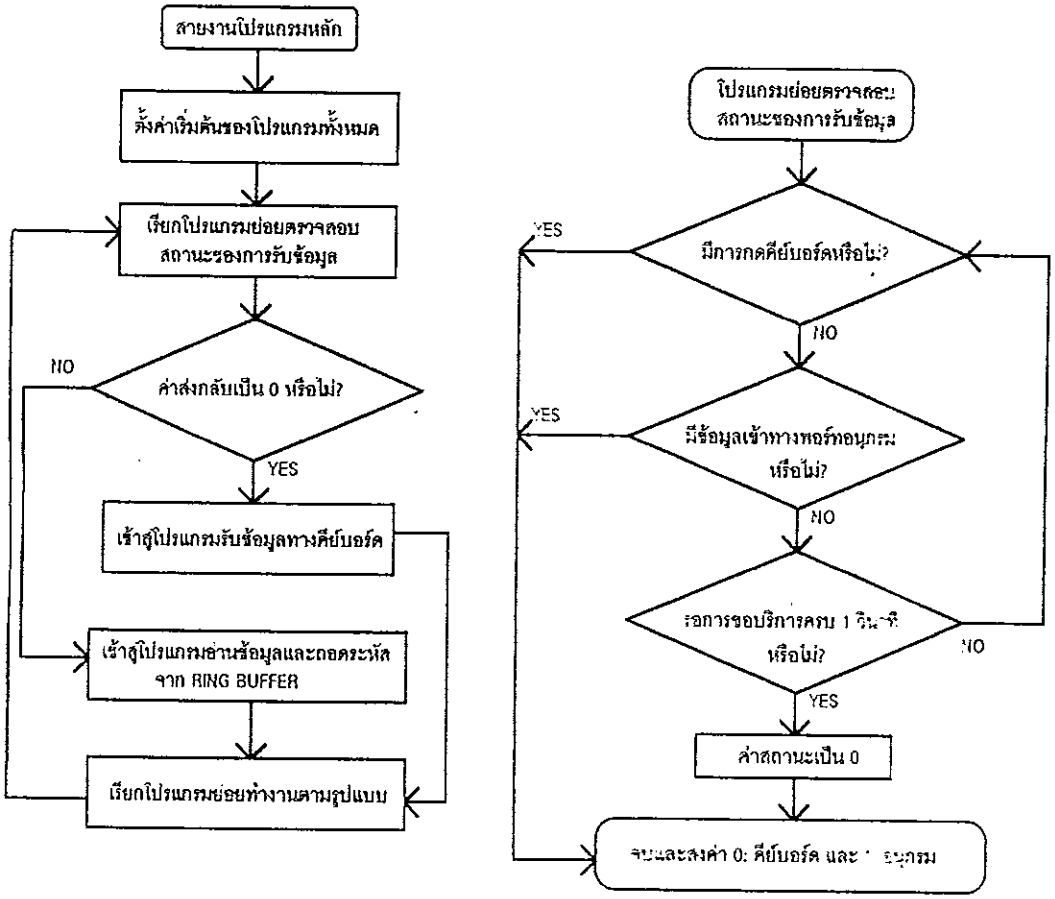
หลังจากกำหนดค่าเริ่มต้นให้กับวงจรนับแล้ว วงจรนับจะทำการเพิ่มค่าของการนับสัญญาณความถี่มูลฐาน 60 เฮิร์ต เมื่อถึงค่าสูงสุดที่เป็นไปได้(FFFFH) จะทำให้เกิดการอินเตอร์รัพท์ภายในระบบจากฐานเวลา และเข้าสู่โปรแกรมบริการอินเตอร์รัพท์ ซึ่งจะแสดงภาวะของบิทแจ้งให้สายงานโปรแกรมหลักทราบ เพื่อทำการส่งระหัสสั่งสุ่มอ่านข้อมูล (OEH) ไปยังระบบเครือข่ายย่อยต่อไป สำหรับแผนภูมิสายงานของการบริการอินเตอร์รัพท์จากฐานเวลาแสดงในรูปที่ 4.13



รูปที่ 4.13 แผนภูมิสายงานของโปรแกรมให้บริการอินเตอร์รัพท์จากฐานเวลาซึ่งกำหนดอัตราการสุ่มอ่านมาตรฐานสำหรับระบบควบคุมการทำงานหลัก

สายงานโปรแกรมหลัก

เนื่องจากในสายงานโปรแกรมหลักมีส่วนจัดการ และโปรแกรมย่อยมากมาย เพราะระบบควบคุมหลักจำเป็นต้องมีหน้าที่ในการจัดการแทบทุกอย่างเพื่อให้สามารถใช้งานแทนไมโครคอมพิวเตอร์ในการสั่งงานได้ การจัดการของสายงานโปรแกรมหลักของระบบควบคุมหลัก มีรายละเอียดของการเรียกโปรแกรมย่อยแสดงในรูปที่ 4.14 และในรายละเอียดการทำงานของโปรแกรมย่อย จะเลือกนำเสนอเฉพาะส่วนที่มีความสำคัญเท่านั้น คือ การจัดการติดต่อกับระบบเครือข่ายย่อย การนำข้อมูลเข้าจัดเก็บในหน่วยความจำ เก็บข้อมูลส่วนขยาย การรับค่าตัวแปรของการใช้งานจากระบบรับข้อมูลจากคีย์บอร์ดในรูปแบบของ PULL-DOWN MENU



รูปที่ 4.14 แผนภูมิสายงานส่วนเริ่มต้นและการเรียกโปรแกรมย่อยของสายงานโปรแกรมหลักของระบบควบคุมการทำงานหลัก

หลังจากที่ระบบควบคุมหลักเข้าสู่โปรแกรมการทำงานจะทำการตรวจสอบสถานะของการสั่งงานซึ่งแบ่งเป็นสองรูปแบบคือ การสั่งงานจากไมโครคอมพิวเตอร์ผ่านทางพอร์ทแบบอนุกรมและการสั่งงาน โดยผ่านทางคีย์บอร์ดของระบบในขั้นตอน

แรกจะรออ่านความเปลี่ยนแปลงของสถานะของพอร์ทอนุกรม และตรวจสอบว่ามีคีย์บอร์ดถูกกดหรือไม่

กรณีที่มีความเปลี่ยนแปลงของสถานะของพอร์ทอนุกรม โปรแกรมหลักจะเข้าสู่โปรแกรมบริการถอดระหัสเฟรมข้อมูลจากหน่วยความจำส่วน RING BUFFER ซึ่งรูปแบบการถอดระหัสในส่วนนี้จะประกอบด้วยคำสั่งงานย่อยมากกว่า 10 คำสั่ง ที่มีระหัสเริ่มต้นเป็น STX(02H) และมีระหัสปิดท้ายเป็น 03H ได้สรุปคำสั่งงานทั้งหมดไว้ในภาคผนวก การถอดระหัสและแปรความหมายเพื่อเรียกโปรแกรมบริการย่อยของแต่ละคำสั่ง มีรูปแบบเช่นเดียวกับกรณีของการถอดระหัสของระบบแสดงผลแบบอิสระที่ได้กล่าวไปแล้ว

กรณีที่มีความเปลี่ยนแปลงของสถานะคีย์บอร์ด โปรแกรมหลักจะเข้าสู่โปรแกรมย่อยรับข้อมูลจากคีย์บอร์ดซึ่งกำหนดรูปแบบการรับค่าทั้งหมดผ่านเมนู เช่น การเลือกรูปแบบการทำงานในสองรูปแบบ คือ แบบเวลาจริงและแบบไม่เป็นเวลาจริง การตั้งเวลาเริ่มต้นและเวลาสิ้นสุดของกระบวนการเก็บข้อมูล จำนวนระบบเครือข่ายย่อยที่ใช้ และการเลือกใช้ค่าอัตราการสุ่มอ่านข้อมูล รูปแบบและตัวแปรที่ผู้ใช้งานเลือกโดยใช้คีย์บอร์ด จะถูกเก็บเป็นตัวแปรไว้ในตำแหน่งเดียวกับรูปแบบและตัวแปรที่ส่งมาจากไมโครคอมพิวเตอร์ เพื่อใช้ในการอ้างอิงเมื่อเข้าสู่โปรแกรมสั่งงานและส่งเข้าสู่ไมโครคอมพิวเตอร์เพื่อใช้ในการอ้างอิง

1. **รูปแบบการจัดการเครือข่ายของระบบควบคุมการทำงานหลัก**

เนื่องจากการเชื่อมโยงระหว่างระบบเครือข่ายย่อย และระบบควบคุมการทำงานหลักจะอยู่ในรูปแบบของโครงสร้างทางฮาร์ดแวร์เป็นแบบระบบเครือข่ายของไมโครคอนโทรลเลอร์ ซึ่งในเครื่องต้นแบบสามารถบริการระบบเครือข่ายย่อยได้ 8 ระบบ ที่เป็นอิสระจากกันและสามารถขยายได้ตามความต้องการโดยการเปลี่ยนแปลงตัวแปรในส่วนของโปรแกรม และเพิ่มฮาร์ดแวร์ให้สอดคล้องกับจำนวนระบบเครือข่ายย่อยที่ใช้ ดังนั้นโครงสร้างของโปรแกรมควบคุมในระบบควบคุมการทำงานหลักจำเป็นต้องมีการจัดรูปแบบ เพื่อให้สามารถบริการต่อระบบเครือข่ายย่อยได้

โดยโครงสร้างทางฮาร์ดแวร์แล้ว ในการเลือกติดต่อกับระบบเครือข่าย

ย่อยใดๆ ระบบควบคุมการทำงานหลักจำเป็นต้องกำหนดทิศทางการติดต่อกับเอาท์พุท ในลักษณะเจาะจงตัวรับ ซึ่งข้อมูลจะถูกส่งออกไปยังตัวรับที่ต้องการเท่านั้นและเช่นเดียวกันกรณีที่ต้องการรับข้อมูล ระบบควบคุมการทำงานหลักจำเป็นต้องเปิดรับอินพุทให้กับระบบเครือข่ายในขณะนั้น โดยการกำหนดการติดต่อกับระบบเครือข่ายย่อยใด ๆ จะถูกควบคุมโดยสายงานโปรแกรมหลัก การรับหรือส่งข้อมูลแบบอนุกรมจะอยู่ภายใต้การควบคุมของสายงานโปรแกรมบริการอินเตอร์รัพท์แบบอนุกรม

สายงานโปรแกรมหลักจะทำการแบ่งหน่วยความจำข้อมูลเป็นส่วน ๆ ที่สอดคล้องกับหมายเลขระหัสเรียกระบบเครือข่ายย่อย ดังนี้

ระบบเครือข่ายย่อยที่เชื่อมต่อกับช่องสัญญาณอินพุทแบบอนุกรมหมายเลข 1 จะมีหน่วยความจำเก็บข้อมูลส่วนขยายในช่วง 32 กิโลไบท์ล่างสุด และสำหรับระบบเครือข่ายย่อยตัวอื่นๆ ก็จะมีหน่วยความจำเก็บข้อมูลส่วนขยายใน 32 กิโลไบท์ที่สูงกว่าขึ้นมาและในการติดต่อกับระบบเครือข่ายย่อยใด ๆ ระบบควบคุมการทำงานหลักจะทำการเปิดช่องเอาท์พุทสำหรับการติดต่อแบบอนุกรม และส่งระหัสของอ่านข้อมูลจากระบบเครือข่ายย่อยตัวนั้น ๆ ก็จะทำการกำหนด BANK ของหน่วยความจำเก็บข้อมูลส่วนขยายที่สอดคล้องกับระบบเครือข่ายย่อยตัวนั้น ๆ หลังจากนั้นจะทำการกำหนดตำแหน่งของตัวชี้หน่วยความจำที่จะเก็บค่าข้อมูลในขณะนั้น และอ่านข้อมูลจากรีจิสเตอร์ของการติดต่อแบบอนุกรม (SERIAL BUFFER : หรือรีจิสเตอร์ SBUF) โดยตรง รูปแบบการทำงานของโปรแกรมย่อยที่ทำหน้าที่ในการติดต่อกับระบบเครือข่ายเพื่อสั่งงาน และอ่านข้อมูลได้แสดงในแผนภูมิสายงานรูปที่ 4.15

2. โปรแกรมย่อยสำหรับการสั่งงานรูปแบบการอ่าน และเก็บข้อมูลจริงแบบเวลาจริง

ขอบข่ายการทำงานของโปรแกรมย่อย เริ่มจากการอ่านค่าจำนวนของระบบเครือข่ายที่ต้องการติดต่อจากตำแหน่งเก็บค่าตัวแปรดังกล่าว และให้ค่ากับตัวนับรอบ จากนั้นจะอ่านเวลาเริ่มต้นของการอ่านและเก็บข้อมูล ตั้งค่าของการสุ่มอ่านข้อมูลให้กับรีจิสเตอร์นับฐานเวลามาตรฐานเปรียบเทียบกับเวลาปัจจุบัน หากผลการเปรียบเทียบเป็นจริงก็จะสั่งเริ่มการทำงาน และกระทำซ้ำอยู่ในวงรอบ

2.1 รอรับการกระตุ้นอันเกิดจากการตั้งค่าของการสุ่มอ่าน ของสายงานโปรแกรมบริการอินเตอร์รัพท์จากฐานเวลาซึ่งจะก่อให้เกิดความเปลี่ยนแปลงของบิทสถานะ

2.2 ส่งระหัสเพื่อสั่งสุ่มอ่านข้อมูลให้กับระบบเครือข่ายทุกตัวพร้อม ๆ กัน

2.3 ส่งระหัส เพื่อขออ่านข้อมูลในปัจจุบันจากระบบเครือข่ายย่อยแต่ละตัว และเก็บในหน่วยความจำข้อมูลส่วนขยายที่สอดคล้องกับหมายเลขของระบบเครือข่ายและส่งระหัสตอบรับข้อมูล (ACKNOWLEDGE)

2.4 เปรียบเทียบเวลาปัจจุบัน กับเวลาสิ้นสุดของกระบวนการเก็บข้อมูลที่ถูกกำหนด

จำนวนรอบของการทำซ้ำในขั้นตอน 2.2.3 จะเท่ากับจำนวนระบบเครือข่ายย่อยหรือจำนวนจุดทดลองที่ผู้ใช้งานต้องการซึ่งได้ป้อนค่าเข้ามาเก็บไว้ในตัวแปรนับรอบแล้ว

รูปแบบของแผนภูมิสายงานของโปรแกรมบริการย่อย สำหรับการสั่งงานในรูปแบบการอ่านและเก็บข้อมูลแบบเวลาจริง แสดงในรูปแบบที่ 4.16

3. โปรแกรมย่อยสำหรับการสั่งงานรูปแบบการอ่าน และเก็บข้อมูลแบบไม่เป็นเวลาจริง

การทำงานของโปรแกรมย่อยจะเริ่มจากการอ่านเวลาเริ่มต้น ของกระบวนการเก็บข้อมูลอ่านช่วงเวลาในการอ่านข้อมูล อ่านอัตราการสุ่มอ่านข้อมูลและส่งให้กับระบบเครือข่ายย่อยทุกตัวเพื่อกำหนดค่าของการหารความถี่มูลฐานในการสุ่มอ่านข้อมูล จากนั้นจะรอจนกระทั่งเวลาปัจจุบันเท่ากับเวลาที่ต้องการเริ่มอ่านข้อมูลและจะส่งคำสั่งในการเริ่มทำงานรูปแบบไม่เป็นเวลาจริง ให้กับระบบเครือข่ายย่อยทุกตัว และจะทำงานในวงรอบของการลดตัวแปรกำหนดช่วงเวลาในการอ่านข้อมูลจนเป็น 0 ก่อนที่จะส่งระหัสสั่งหยุดการทำงาน และอ่านค่าบลอกของข้อมูลจากระบบเครือข่ายย่อยที่ละตัว

รูปแบบการทำงานของโปรแกรมย่อยในส่วนนี้ แสดงในรูปที่ 4.17

4. โปรแกรมย่อยสำหรับการจัดกระทำเมนูตัวเลือกแบบเลื่อน

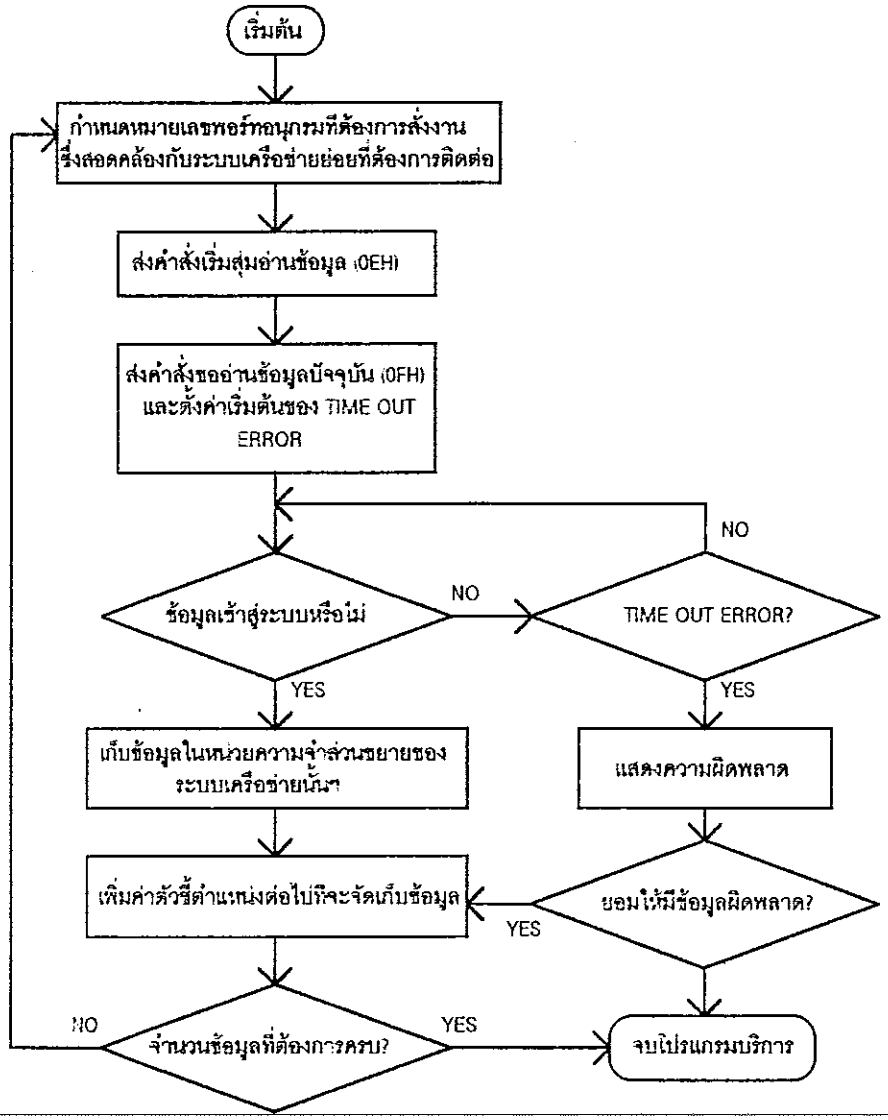
เพื่อให้การตั้งค่าเริ่มต้น และการรับคำสั่งงานตลอดจนตัวแปรต่าง ๆ

สามารถกระทำได้โดยง่ายจึงได้พัฒนาส่วนรับคำสั่งงานเป็นแบบเมนูตัวเลือก ในการเลือก ผู้ใช้งานเพียงแค่เลื่อนแถบเมนูตัวเลือกไปยังตำแหน่งของตัวเลือกที่ต้องการและเลือกป้อนค่าอื่น ๆ ที่จำเป็น โดยการจัดกระทำเมนูในระบบไมโครคอนโทรลเลอร์ได้ใช้แผนภูมิสายงานในรูปแบบที่ 4.18

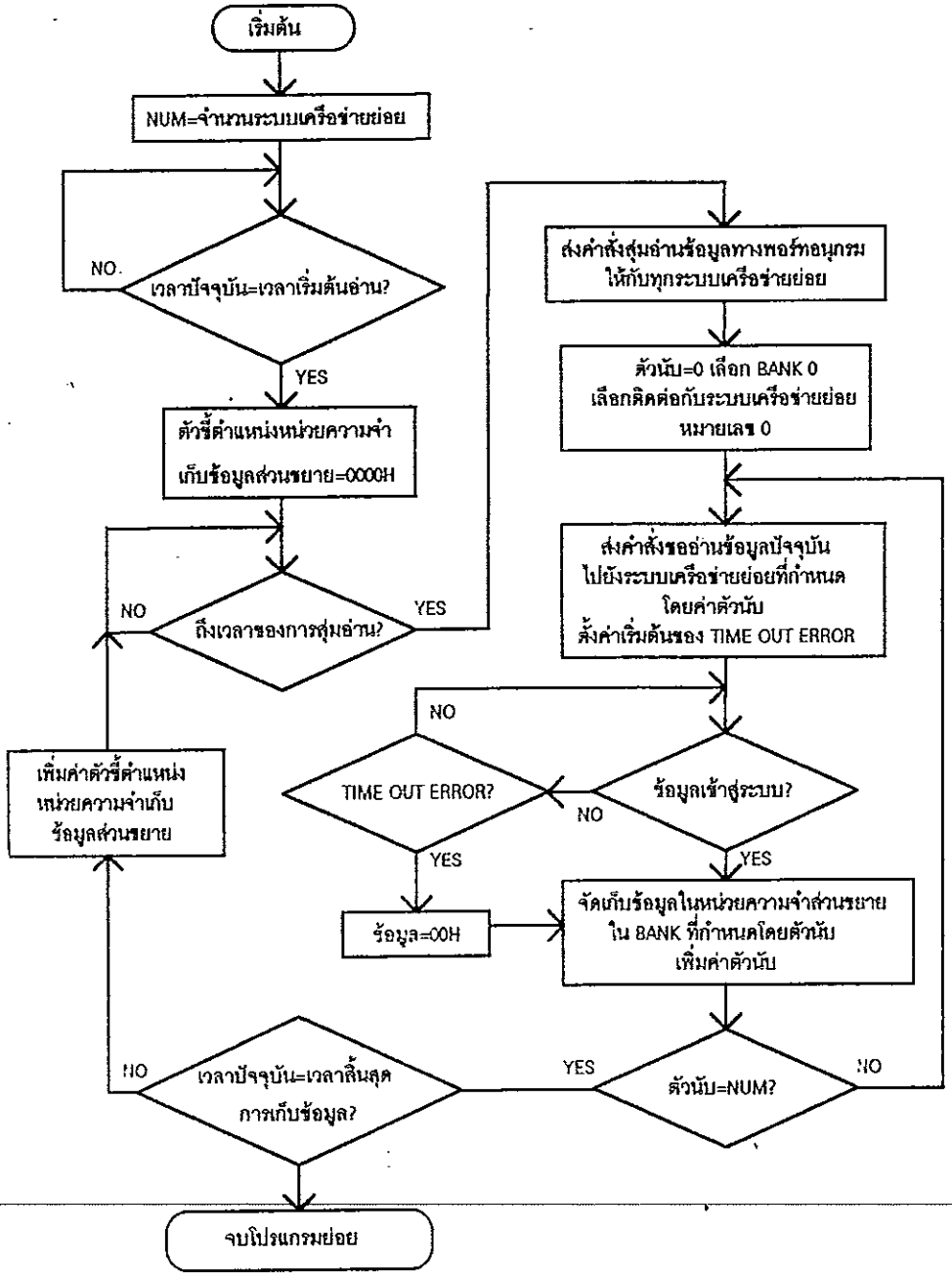
5. โปรแกรมย่อยสำหรับการอ่านสถานะของแป้นพิมพ์

ใช้หลักการในการอ่านค่าสถานะแป้นพิมพ์ ซึ่งเป็นส่วนฮาร์ดแวร์ที่เพิ่มเติมให้กับระบบควบคุมหลัก ในรูปแบบการอ่านค่าแบบ POLLING โดยที่โปรแกรมย่อยในส่วนนี้ได้เพิ่มความสามารถในการตั้งเวลาเพื่อตรวจสอบการกดแป้นพิมพ์ คือ ในกรณีที่มีการเรียกโปรแกรมย่อยเพื่อรอรับค่าการกดแป้นพิมพ์แต่ไม่มีการกดแป้นพิมพ์ในช่วงเวลา 30 วินาที โปรแกรมย่อยจะกลับไปสู่โปรแกรมที่เรียกใช้ทันที

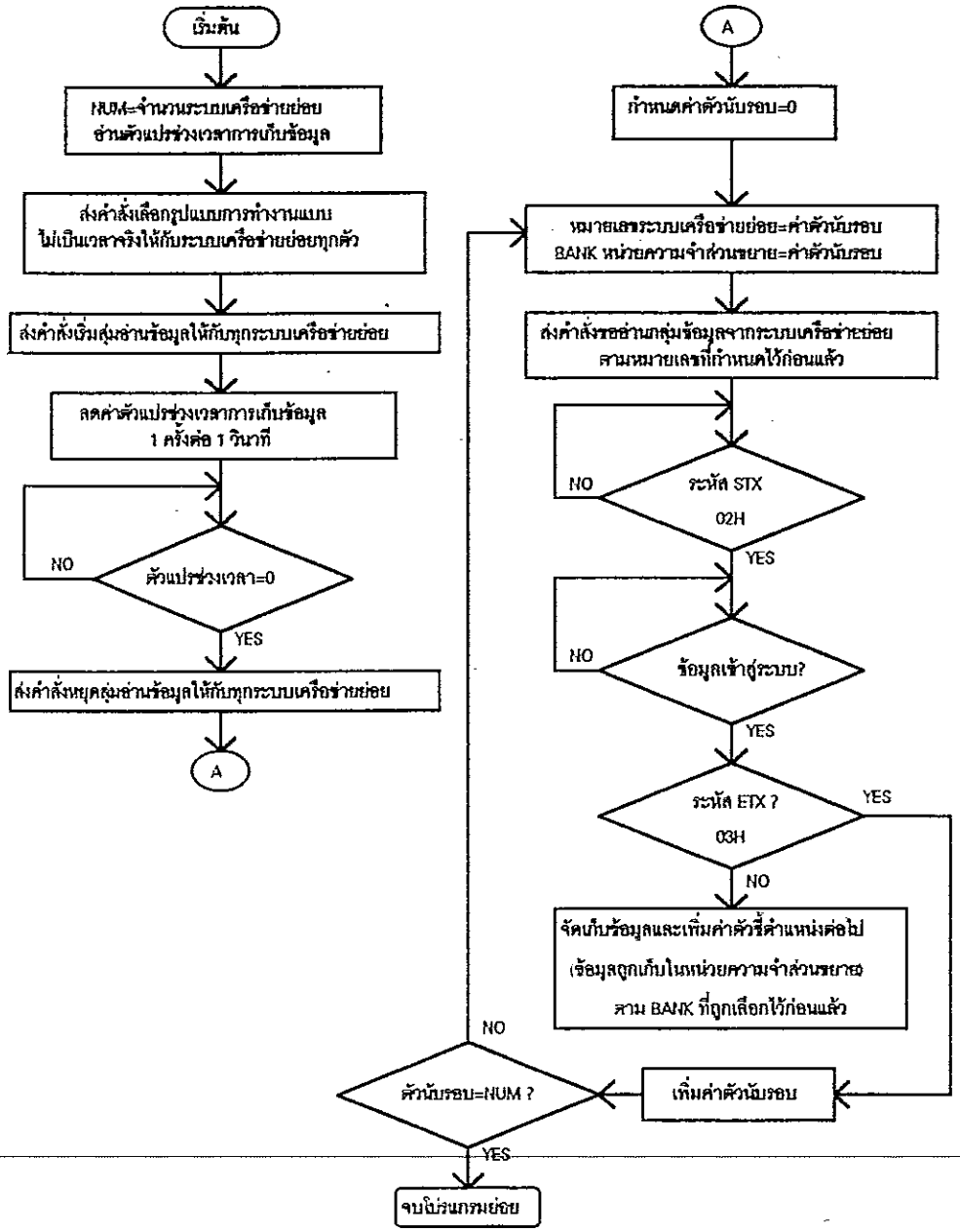
รูปแบบของแผนภูมิสายงานที่ใช้ในโปรแกรมย่อยส่วนนี้แสดงในรูปที่ 4.19



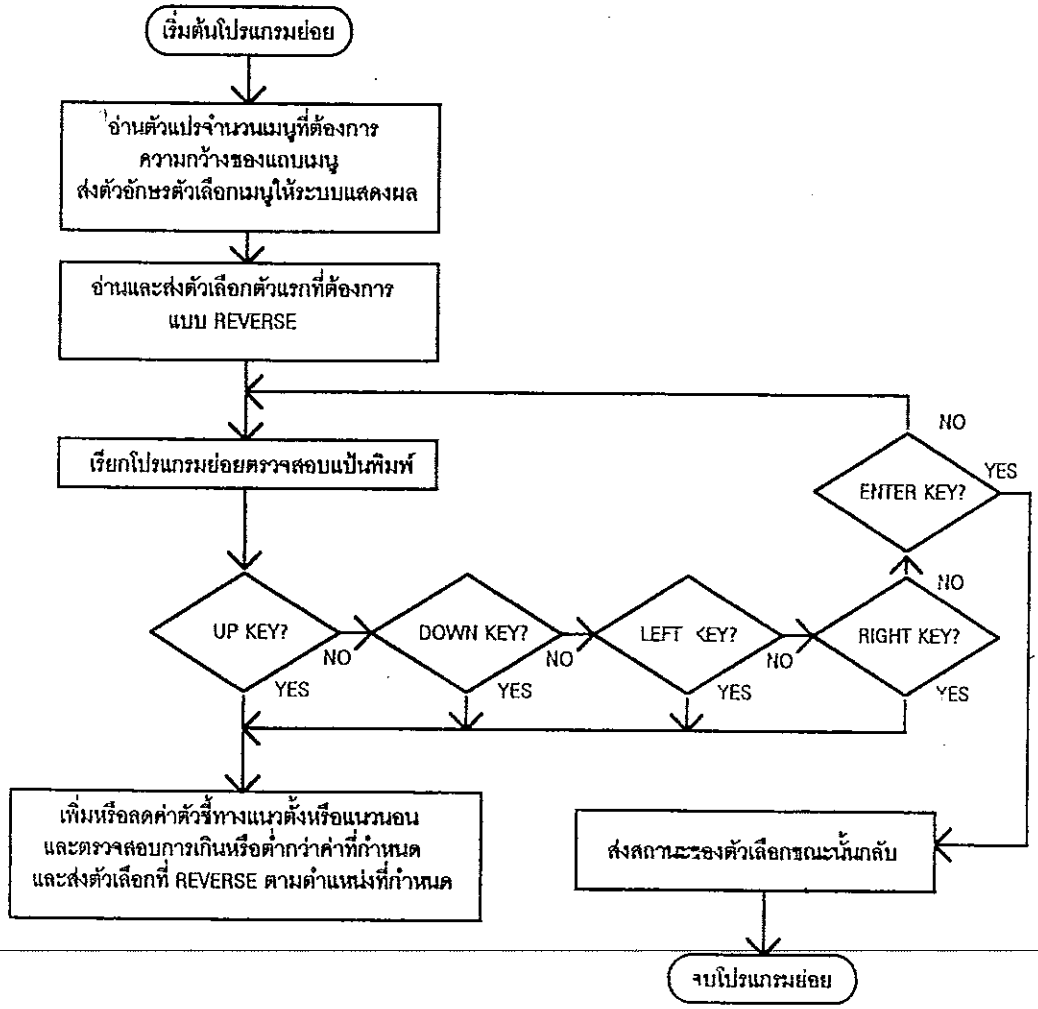
รูปที่ 4.15 แผนภูมิสายงานโปรแกรมย่อยในการสั่งงานและติดต่อเพื่ออ่านข้อมูล
โดยพื้นฐานของระบบควบคุมการทำงานหลัก



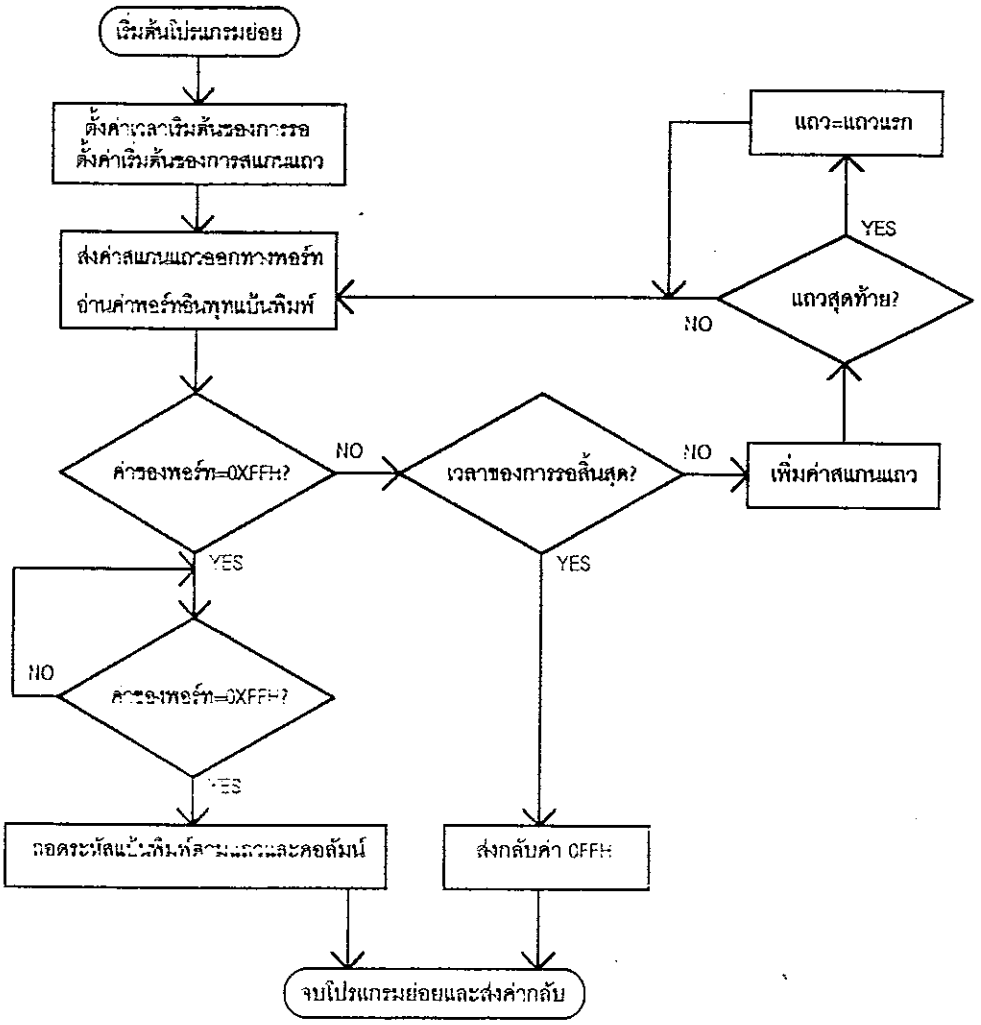
รูปที่ 4.16 แผนภูมิสายงานโปรแกรมย่อยในการสั่งงานและติดต่อเพื่ออ่านข้อมูลของระบบควบคุมการทำงานหลัก



รูปที่ 4.17 แผนภูมิสายงานโปรแกรมในการสั่งงานและติดต่อเพื่ออ่านข้อมูลของระบบควบคุมการทำงานหลักในรูปแบบของการเก็บข้อมูลแบบไม่เป็นเวลาจริง



รูปที่ 4.18 แผนภูมิสายงานโปรแกรมย่อยสำหรับการจัดกระทำเมนู
ในระบบควบคุมหลัก



รูปที่ 4.19 โปรแกรมย่อยในการตรวจสอบสถานะของแป้นพิมพ์

บทที่ 5

บทสรุป

จากวัตถุประสงค์ของวิทยานิพนธ์ การออกแบบและพัฒนาระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์ คือ

1. เพื่อสร้างและพัฒนาเครื่องต้นแบบสำหรับควบคุมการเก็บข้อมูลระยะไกล ที่ใช้งานได้โดยอิสระภายใต้การควบคุมของไมโครโปรเซสเซอร์ โดยสามารถเชื่อมโยงระบบกับไมโครคอมพิวเตอร์เพื่อส่งข้อมูลเข้าวิเคราะห์ผลได้

2. เพื่อพัฒนาระบบย่อยเครือข่าย ที่ทำหน้าที่ในการเปลี่ยนรูปแบบของข้อมูลจากปริมาณทางฟิสิกส์ให้อยู่ในรูปแบบทางดิจิตอลที่มีความยืดหยุ่นในการดัดแปลงและเปลี่ยนแปลงโปรแกรมได้ทำการเชื่อมโยงกับระบบควบคุมการเก็บข้อมูลหลัก

3. เพื่อทดสอบเครื่องต้นแบบที่ได้กับการใช้งานเพื่อวิเคราะห์ผลกับงานวิจัยจริง

4. เพื่อพัฒนาระบบอิสระ ภายใต้การควบคุมโดยไมโครโปรเซสเซอร์ เพื่อที่นักวิจัยท่านอื่น ๆ สามารถนำไปดัดแปลงใช้งานกับงานของตนได้โดยง่าย

5. เพื่อออกแบบ และสร้างระบบพัฒนาไมโครโปรเซสเซอร์ ตระกูล MCS-51 ของบริษัท INTEL CORP.(MCS-51 EMULATOR) ทั้งทางด้านซอร์ฟแวร์และฮาร์ดแวร์ เพื่อนำไปใช้งานพัฒนาระบบอื่น ๆ ที่มีประโยชน์ต่อไป

สำหรับขั้นตอนการดำเนินการจริงนั้น จำเป็นต้องเริ่มจากการศึกษาและทำความเข้าใจการใช้งานไมโครโปรเซสเซอร์และไมโครคอนโทรลเลอร์ รูปแบบการทำงานระบบเก็บข้อมูลมาตรฐานที่มีขายในท้องตลาด และข้อจำกัดในการใช้งานเครื่องมือเหล่านั้น โดยที่ผู้วิจัยได้ศึกษารูปแบบการทำงานของระบบเก็บข้อมูลภายในห้องปฏิบัติการของบริษัท NATIONAL INSTRUMENT จำกัด ซึ่งพบว่ามีข้อจำกัดของการใช้งานในรูปแบบการเก็บข้อมูลระยะไกล จึงได้ทำการออกแบบระบบภายใต้หัวข้อของงานวิทยานิพนธ์เพื่อให้มีความสามารถในส่วนดังกล่าว โดยการใช้ทฤษฎีของการสื่อสารข้อมูลระยะไกลของระบบที่ควบคุมโดยไมโครคอนโทรลเลอร์ มาเป็น

สื่อกลางในการส่งคำสั่งควบคุม และรับข้อมูลและใช้การทำงานของไมโครคอน-โทรลเลอร์แบบเวลาจริง การเปลี่ยนรูปแบบของสัญญาณจากการทดลองให้เป็นข้อมูลทางดิจิตอลและจัดการสื่อสารข้อมูล เพื่อให้งานวิทยานิพนธ์สามาถบรรลุเป้าหมายที่ได้วางไว้

สำหรับวัตถุประสงค์ของการทำวิทยานิพนธ์ในข้อ 5 ซึ่งจำเป็นต้องทำในขั้นตอนแรกก่อน ได้ประสบความสำเร็จในการออกแบบระบบพัฒนาไมโครคอนโทรลเลอร์ตระกูล MCS-51 ของบริษัท INTEL ทั้งทางด้านฮาร์ดแวร์ของระบบพัฒนาและโปรแกรมช่วยในการพัฒนา ซึ่งสามารถนำไปใช้งานในส่วนอื่น ๆ ของงานวิทยานิพนธ์ได้ และคาดว่าจะเป็นประโยชน์ต่อการนำไปใช้งานพัฒนาระบบอิสระภายใต้การควบคุมของไมโครคอนโทรลเลอร์ 8031 ต่อไป

วัตถุประสงค์ในข้อ 1,2 และ 3 นั้นได้ใช้ระบบพัฒนาไมโครคอนโทรล-เลอร์ 8031 เป็นตัวช่วยในการพัฒนาโปรแกรม และตรวจสอบการทำงานของระบบโดยได้สร้างระบบ ภายใต้การควบคุมการทำงานของไมโครคอนโทรลเลอร์ 8031 เป็นอิสระจากกันจำนวน 3 ระบบภายใต้โปรแกรมคุมงานของไมโครคอนโทรลเลอร์ 8031 ที่ต้องออกแบบ 3 โปรแกรมใน 3 ระบบควบคุมที่มีโครงสร้างทางฮาร์ดแวร์และรูปแบบการทำงานที่แตกต่างกันคือระบบแสดงผลแบบอิสระแบบ ALPHANUMERIC DISPLAY ระบบเครือข่ายย่อยแปลงข้อมูลการทดลอง เป็นระหัสการสื่อสารแบบโปรแกรมได้และระบบควบคุมการทำงานหลัก โดยองค์ประกอบของโปรแกรมควบคุมการทำงานแบ่งเป็นสองสายคือ สายงานโปรแกรมหลักและสายงานโปรแกรมบริการอินเตอร์รัพท์แบบหลายแหล่งกำเนิด ระบบต่าง ๆ สามารถเชื่อมโยงกับระบบควบคุมอื่น ๆ ภายใต้มาตรฐานการเชื่อมโยงแบบขนานหรือแบบอนุกรมซึ่งสามารถบรรลุผลตามวัตถุประสงค์ข้อที่ 4

สรุปผลการวิจัย

จากการที่ระบบทั้งหมดที่ได้ออกแบบขึ้นแบ่งเป็น 3 ระบบหลัก คือ

- ระบบเครือข่ายย่อย ซึ่งประกอบด้วยฮาร์ดแวร์และโปรแกรมควบคุมงานที่ซ้ำซ้อนกัน 8 ชุด
- ระบบควบคุมการทำงานหลัก
- ระบบแสดงผลแบบอิสระ

ดังนั้น จึงจำเป็นต้องมีการทดสอบแบบแยกส่วนก่อนที่จะประกอบเป็นระบบเครือข่ายที่มีระบบเครือข่ายย่อย 8 ตัวได้

1. การทดสอบ

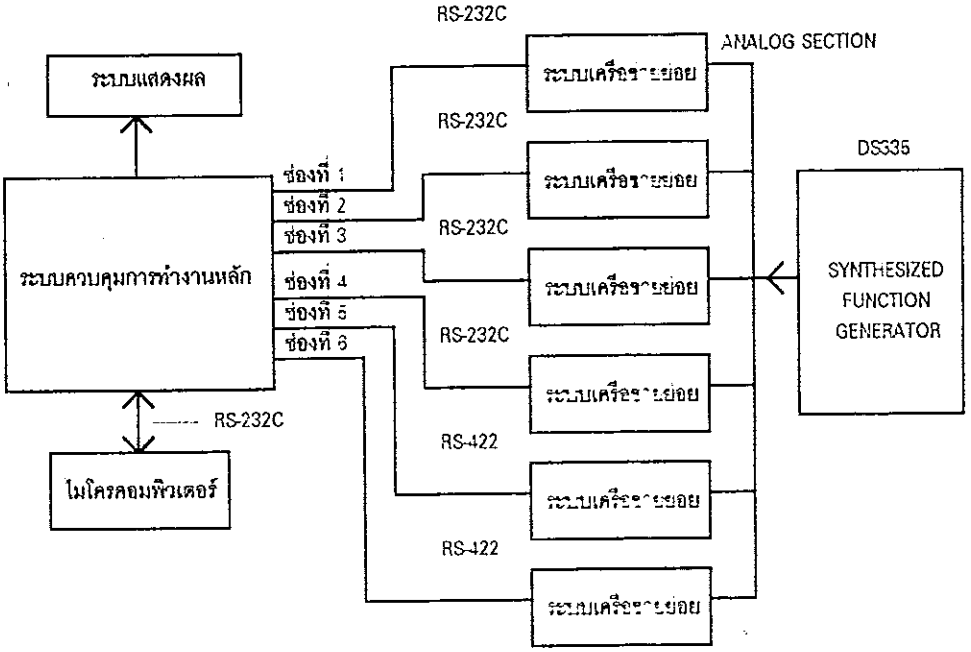
จากรูปแบบการทำงานขั้นต้นของระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครคอนโทรลเลอร์ที่แบ่งรูปแบบการทำงานเป็น 2 รูปแบบ คือ

- รูปแบบการเก็บข้อมูลแบบเวลาจริง ซึ่งใช้สัญญาณในการสุ่มอ่านข้อมูลจากระบบควบคุมการทำงานหลัก
- รูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง ซึ่งระบบเครือข่ายย่อยมีสัญญาณในการสุ่มอ่านข้อมูลที่เป็นอิสระจากกันภายใต้คำสั่ง เริ่มต้นจากระบบควบคุมการทำงานหลัก

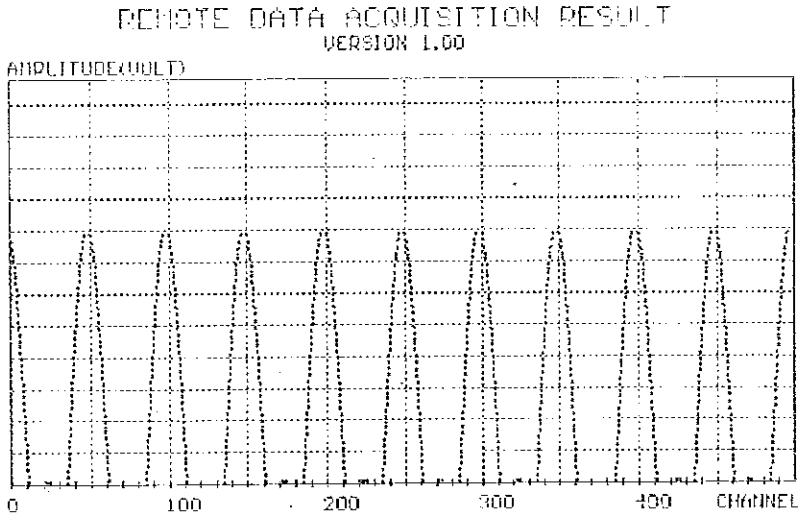
เนื่องจากการที่จะกำหนดความเที่ยงตรงในการทำงาน ของระบบเก็บข้อมูลระยะไกลได้นั้นจำเป็นต้องทดสอบระบบกับแหล่งกำเนิดรูปแบบสัญญาณ ความถี่และความสูงของระดับสัญญาณที่มีความน่าเชื่อถือได้สูง จึงเลือกต่ออินพุทสัญญาณอนาลอกของระบบเครือข่ายย่อย เข้ากับเครื่องกำเนิดสัญญาณแบบโปรแกรมรูปแบบการทำงานได้(SYNTHESIZED FUNCTION GENERATOR) ของบริษัท STANFORD RESEARCH SYSTEMS จำกัด รุ่นของเครื่องที่ใช้ คือ DS335 สามารถกำหนดความสูงของสัญญาณที่ใช้ และความถี่ของสัญญาณสามารถกำหนดรายละเอียดความถูกต้องได้ถึงทศนิยมตำแหน่งที่ 4

1.1 การทดสอบการทำงานในรูปแบบการทำงานไม่เป็นเวลาจริง

รูปแบบของการจัดระบบเพื่อการทดสอบในขั้นตอนที่ 1 แสดงในรูปที่ 5.1 ผลของการทดสอบการทำงานของระบบที่ความถี่สัญญาณ และอัตราการสุ่มอ่านข้อมูลอนาลอกต่าง ๆ และกราฟแสดงของการอ่านข้อมูลที่ได้จากหน่วยความจำระบบควบคุมหลัก โดยการเรียกอ่านข้อมูลกระทำโดยโปรแกรมเรียกอ่านข้อมูลที่ได้พัฒนาบนไมโครคอมพิวเตอร์โดยใช้ภาษา C แสดงในรูปที่ 5.2-5.8



รูปที่ 5.1 รูปการจัดระบบเก็บข้อมูลระยะไกลเพื่อการทดสอบการทำงาน



รูปที่ 5.2 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จากเครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 100 เฮิร์ต ความสูง 2.5 โวลท์

เลือกรูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง

ใช้คาบของการสุ่มอ่านข้อมูล 200 ไมโครวินาที

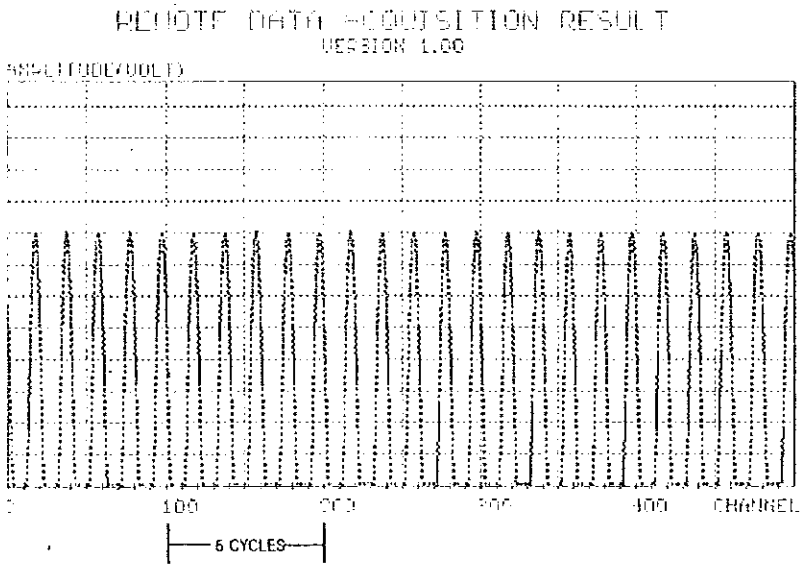
จากรูปที่ 5.2 ซึ่งเป็นผลของการอ่านสัญญาณรูป SINE มาตรฐานความถี่สัญญาณ 100 เฮิร์ตและความสูง(AMPLITUDE)ของสัญญาณเป็น 2.5 โวลท์เงื่อนไขของการอ่านค่าสัญญาณ ได้เลือกใช้รูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง และตัวแปรของการสุ่มอ่านเป็นดังนี้

อัตราการสุ่มอ่านข้อมูลเป็น 200 ไมโครวินาที

จำนวนระบบเครือข่ายย่อยที่ใช้ 1 ตัว

การทดสอบความถูกต้องของการอ่านความถี่สัญญาณ ทำได้โดยพิจารณาจากกราฟที่ได้ซึ่งใน 1 ช่องทางแกน X (CHANNEL)จะแทนช่วงเวลา 200 ไมโครวินาที คาบของสัญญาณ (PEAK TO PAEK) ที่อ่านได้มีความกว้าง 5 ช่องบนสเกลแสดงผล(1 สเกลแสดงผลของกราฟเทียบเท่ากับจำนวนช่วงสัญญาณ 10 ช่อง)ดังนั้นคาบของสัญญาณรูป SINE แทนค่าด้วยตัวแปร T

T = 5X10X200 = 10000 ไมโครวินาทีหรือเท่ากับความถี่ 100 เฮิร์ต



รูปที่ 5.3 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จากเครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 250 เฮิร์ต ความสูง 2.5 โวลท์

เลือกรูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง

ใช้คาบของการสุ่มอ่าน 200 ไมโครวินาที

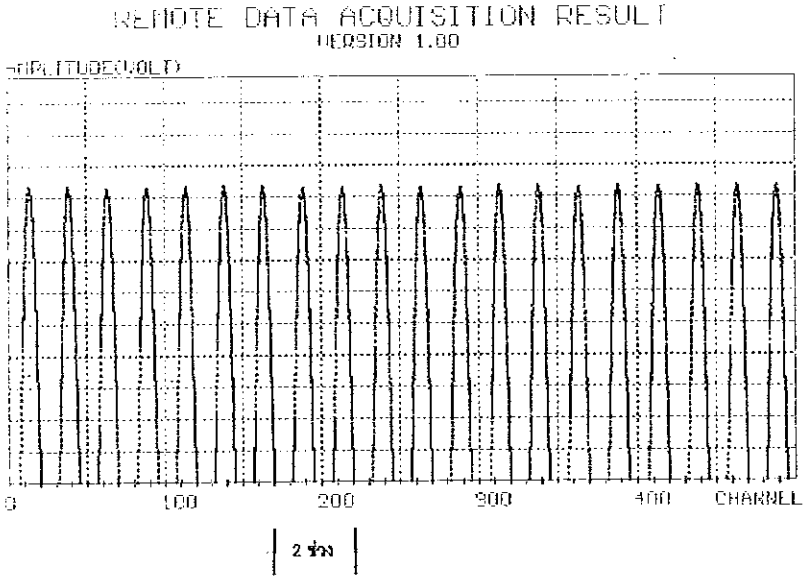
จากรูปที่ 5.3 เป็นผลที่ได้จากการสุ่มอ่านข้อมูล ในรูปแบบไม่เป็นเวลาจริง ตามเงื่อนไขของระบบเก็บข้อมูล ดังนี้

อัตราการสุ่มอ่านข้อมูลเป็น 200 ไมโครวินาที

จำนวนระบบเครือข่ายย่อยที่ใช้ 2 ตัว

ผลของการคำนวณความถี่ที่ได้โดยเลือกช่วง 5 คาบของสัญญาณจะเท่ากับจำนวนสเกลบนแกน X 100 ช่องสัญญาณ ดังนั้นความถี่สัญญาณที่ได้จะเป็น 5/(200 ไมโครวินาที X 100) คือ 250 เฮิร์ต

สำหรับจุดประสงค์ในการแสดงกราฟรูปนี้ เพื่อแสดงให้เห็นความสามารถในการสุ่มอ่านข้อมูลแบบไม่เป็นเวลาจริง ที่อัตราการสุ่มอ่านสูงจะได้เส้นกราฟจากระบบเครือข่ายย่อยแต่ละตัวที่ทับสนิทกัน ซึ่งเทียบได้กับการที่ระบบเครือข่ายย่อยเริ่มสุ่มอ่านข้อมูลที่เวลาเดียวกัน



รูปที่ 5.4 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จาก
เครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 200 เฮิร์ต ความสูง 3.00 โวลท์

เลือกรูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง

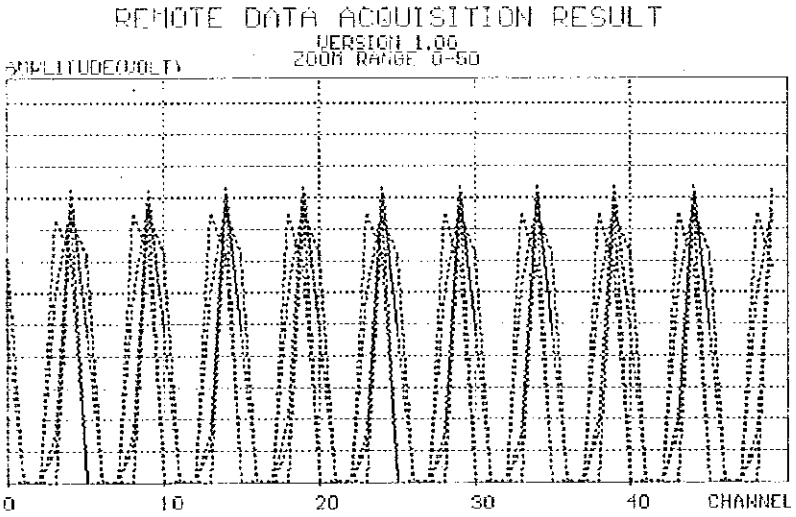
เลือกใช้จำนวนเครือข่ายย่อย 7 ตัว

ใช้คาบของการสุ่มอ่านข้อมูล 200 ไมโครวินาที

การคำนวณหาค่าความถี่ของสัญญาณที่สุ่มอ่านได้จากรูปที่ 5.4 สามารถทำได้ดังนี้

คาบเวลา = (50 ช่องสัญญาณX200 ไมโครวินาที)/2 = 5 มิลลิวินาที

ดังนั้น ความถี่ของสัญญาณที่สุ่มอ่าน คือ 200 เฮิร์ต ซึ่งเท่ากับความถี่สัญญาณเข้า



รูปที่ 5.5 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จาก
เครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 1 กิโลเฮิร์ต ความสูง 3.00 โวลท์

เลือกรูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง

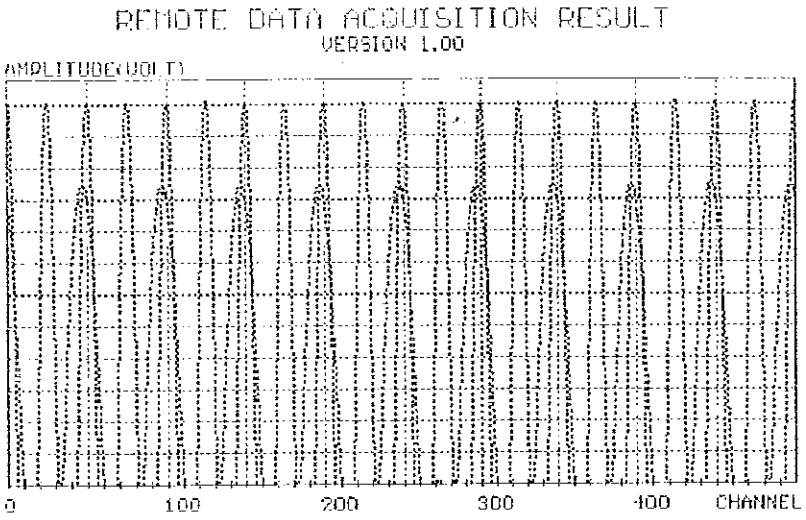
เลือกใช้จำนวนเครือข่ายย่อย 5 ตัว

ใช้คาบของการสุ่มอ่านข้อมูล 200 ไมโครวินาที

กราฟที่ได้จากรูปที่ 5.5 ความถี่ที่สามารถคำนวณได้คือ

คาบของสัญญาณ = 5 X 200 = 1000 ไมโครวินาที

ดังนั้นความถี่สัญญาณที่อ่านได้เป็น 1 กิโลเฮิร์ต



รูปที่ 5.6 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จาก
เครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 100 เฮิร์ต ความสูง 3.00 โวลท์และ 4 โวลท์

เลือกรูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง

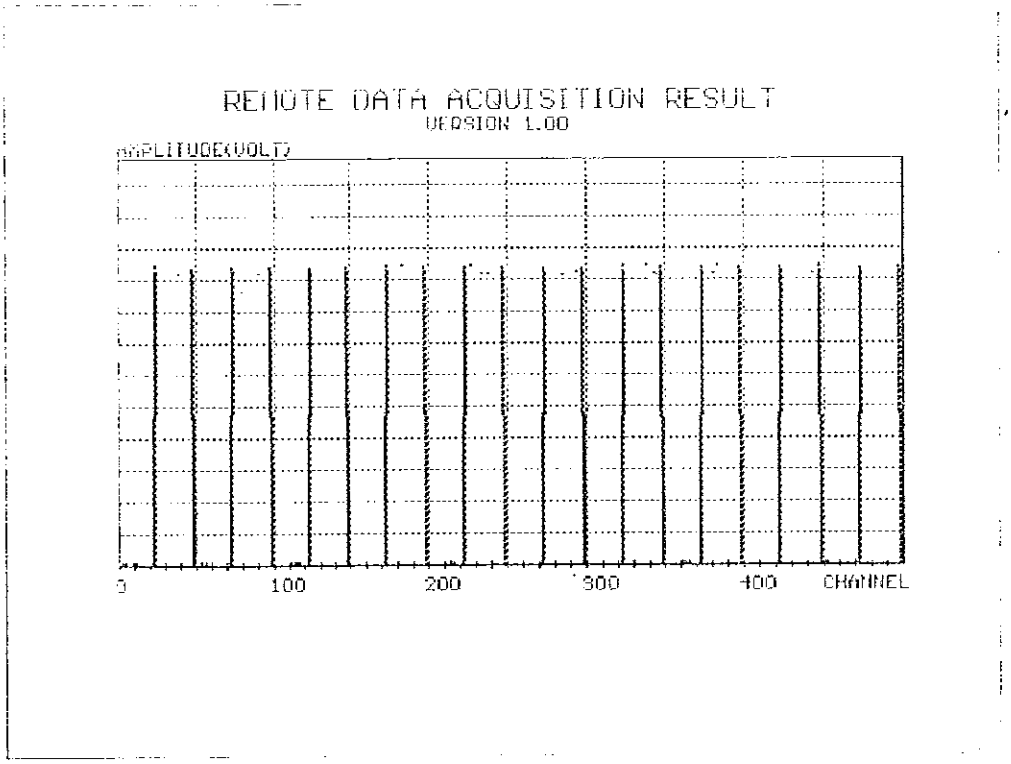
เลือกใช้จำนวนเครือข่ายย่อย 2 ตัว

โดยใช้คาบของการสุ่มอ่านข้อมูล 400 และ 200 ไมโครวินาที

สำหรับเส้นกราฟที่มีความสูงสัญญาณ 4 และ 3 โวลท์ ตามลำดับ

กราฟแสดงผลในรูปที่ 5.6 แสดงการเปรียบเทียบผลที่ได้ของการสุ่มอ่านข้อมูลที่อัตราการสุ่มอ่านต่างกันเป็นจำนวนเท่า (คาบ 200 ไมโครวินาที และ 400 ไมโครวินาที) ผลคือสามารถเปรียบเทียบข้อมูลกันได้ โดยหากเลือกการขยายสเกล

แสดงผลของกราฟที่มีคาบการสุ่มอ่านข้อมูลเป็น 400 ไมโครวินาทีเป็น 2 เท่า จะได้รายละเอียดของข้อมูลเท่ากับกรณีคาบการสุ่มอ่านข้อมูลเป็น 200 ไมโครวินาที เมื่อความถี่ของสัญญาณที่ต้องการเก็บข้อมูล ไม่เกินความสามารถในการสุ่มอ่านด้วยคาบเวลาที่มากกว่า



รูปที่ 5.7 กราฟผลของการสุ่มอ่านสัญญาณรูป SQUARE ที่ได้จาก
เครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 50 เฮิร์ต ความสูง 3.00 โวลท์

เลือกรูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง

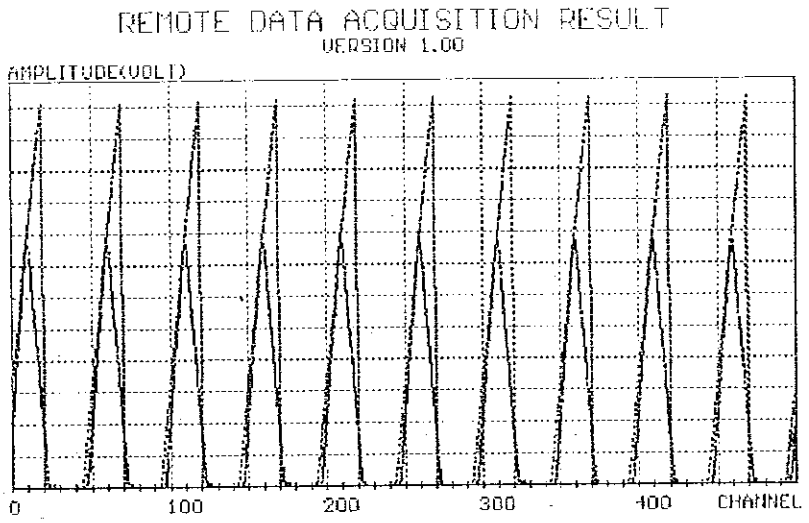
เลือกใช้จำนวนเครือข่ายย่อย 7 ตัว

ใช้คาบของการสุ่มอ่านข้อมูล 400 ไมโครวินาที

ค่าคาบเวลาของสัญญาณ และความถี่ที่ได้จากการคำนวณของกราฟใน รูปที่ 5.7

คาบเวลาสัญญาณ = (400 ไมโครวินาที) X 100/2 = 0.02 วินาที

ดังนั้น ความถี่สัญญาณที่อ่านได้เป็น 50 เฮิร์ต



รูปที่ 5.8 กราฟผลของการสุ่มอ่านสัญญาณรูปอื่น ๆ ที่ได้จาก
เครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 25 เฮิร์ต ความสูง 2.50 และ 4.00 โวลท์

เลือกรูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง

เลือกใช้จำนวนเครือข่ายย่อย 5 และ 2 ตัว สำหรับความสูงของสัญญาณตามลำดับ

ใช้คาบของการสุ่มอ่านข้อมูล 800 ไมโครวินาที

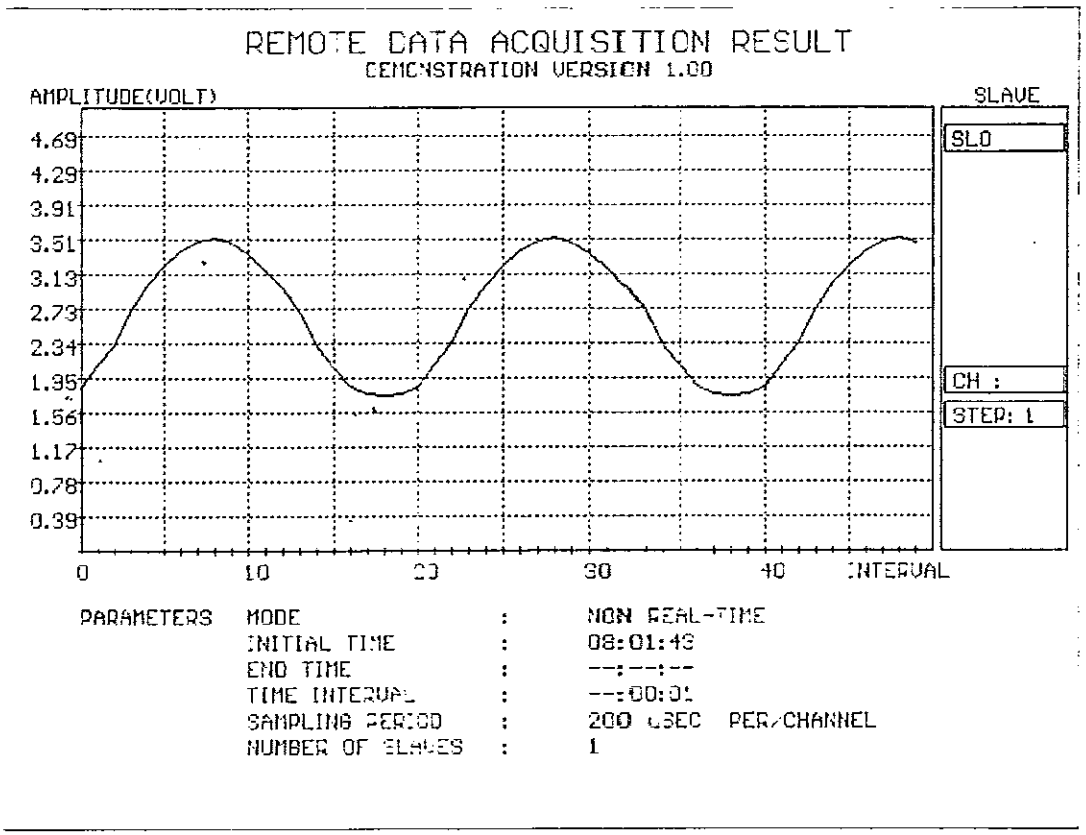
ค่าคาบเวลาของสัญญาณ และความถี่ที่ได้จากการคำนวณของกราฟในรูปที่ 5.8

คาบเวลาสัญญาณ = (800 ไมโครวินาที) X 50 = 0.04 วินาที

ดังนั้น ความถี่สัญญาณที่อ่านได้เป็น 25 เฮิร์ต

- การทดสอบหาสัญญาณความถี่สูงสุดที่ระบบเก็บข้อมูลระยะไกลสามารถบันทึกได้ โดยรูปแบบของสัญญาณนั้นไม่เปลี่ยนแปลงไปจากรูปแบบปกติ

ในการทดสอบนี้ ได้ใช้แหล่งกำเนิดความถี่ที่ได้จากเครื่องกำเนิดความถี่มาตรฐานและปรับค่าเพิ่มขึ้นเรื่อย ๆ จนกระทั่งกราฟผลของการสุ่มอ่านข้อมูลสัญญาณมีความผิดเพี้ยน ซึ่งผลของกราฟที่ได้แสดงในรูปที่ 5.9 เป็นต้นไป



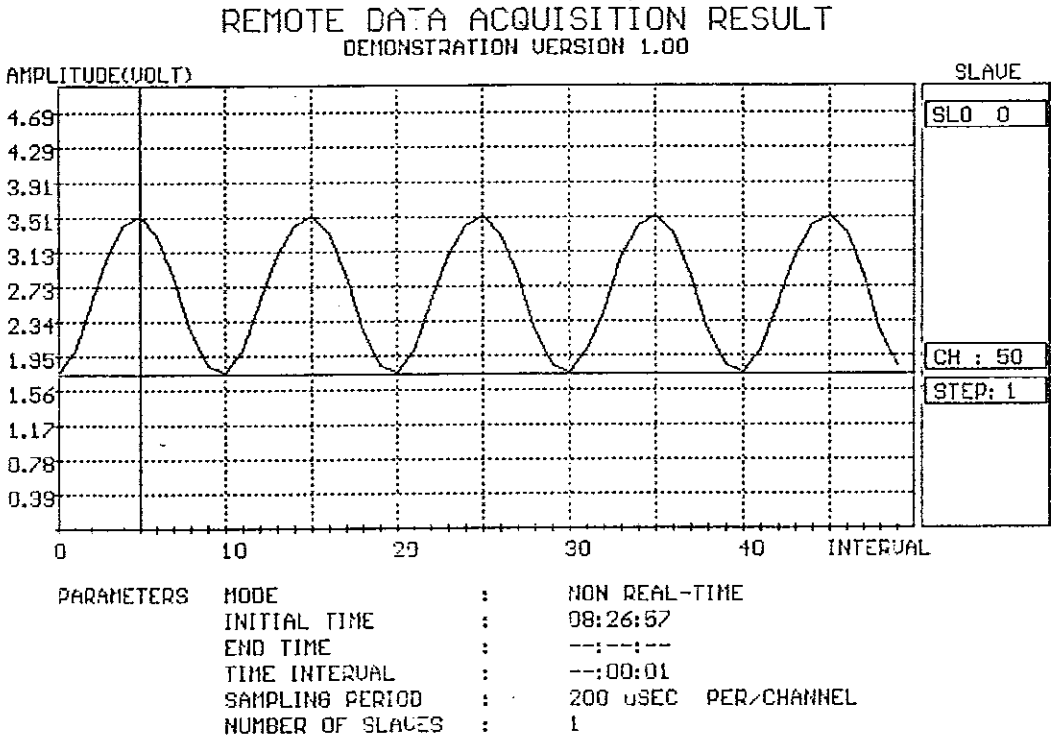
รูปที่ 5.9 กราฟแสดงผลของการสุ่มอ่านสัญญาณความถี่ 250 เฮิร์ต ความสูงสัญญาณ 1.5 โวลท์

ความถี่ที่คำนวณได้จากกราฟผลการอ่านค่า

สัญญาณตามรูปมี AMPLITUDE ที่ช่องสัญญาณ 8 และ 28 ซึ่งมีระยะห่าง 20 ช่อง หรือ 20 X 200 ไมโครวินาที

ดังนั้น คาบเวลาของสัญญาณจากกราฟผลการสุ่มอ่าน เป็น 4 มิลลิวินาที

หรือเทียบเท่ากับความถี่ 250 เฮิร์ต ซึ่งยังเป็นสัญญาณที่ไม่ผิดเพี้ยนจากการสุ่มอ่าน



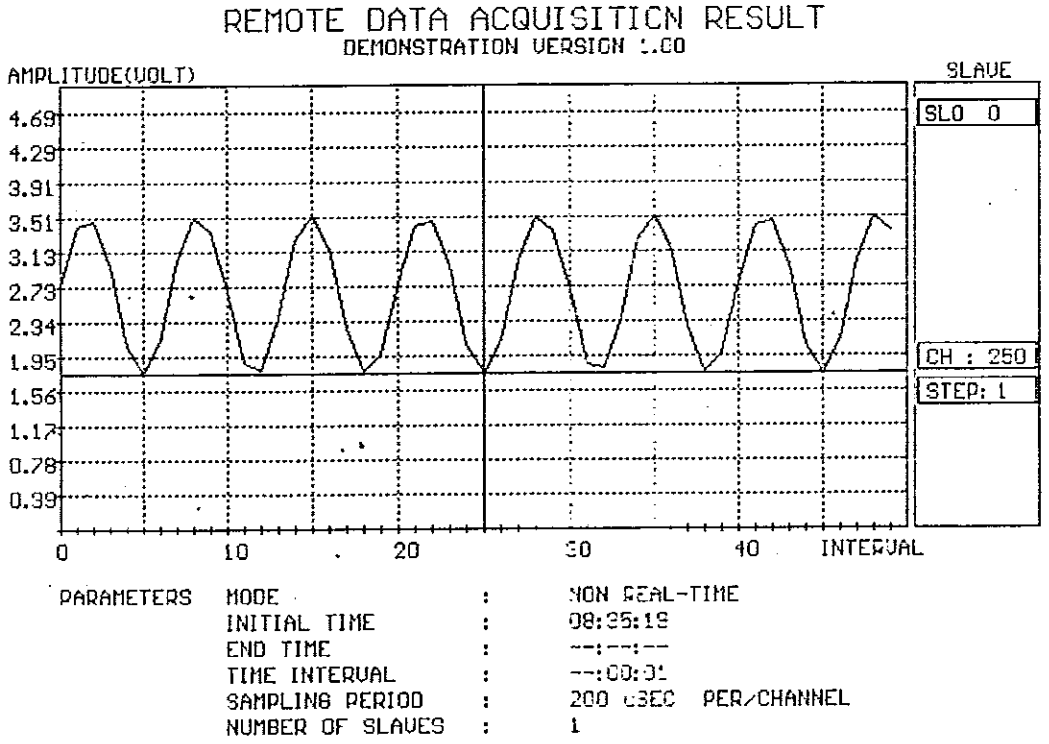
รูปที่ 5.10 กราฟแสดงผลของการสุ่มอ่านสัญญาณความถี่ 500 เฮิร์ต
ความสูงสัญญาณ 1.5 โวลท์

ความถี่ที่คำนวณได้จากกราฟผลการอ่านค่า

สัญญาณตามรูปมี AMPLITUDE ที่ช่องสัญญาณ 5 และ 15 ซึ่งมีระยะห่าง 10 ช่อง หรือ 10 X 200 ไมโครวินาที

ดังนั้น คาบเวลาของสัญญาณจากกราฟผลการสุ่มอ่าน เป็น 2 มิลลิวินาที

หรือเทียบเท่ากับความถี่ 500 เฮิร์ต ซึ่งยังเป็นสัญญาณที่ไม่ผิดเพี้ยนจากการสุ่มอ่าน



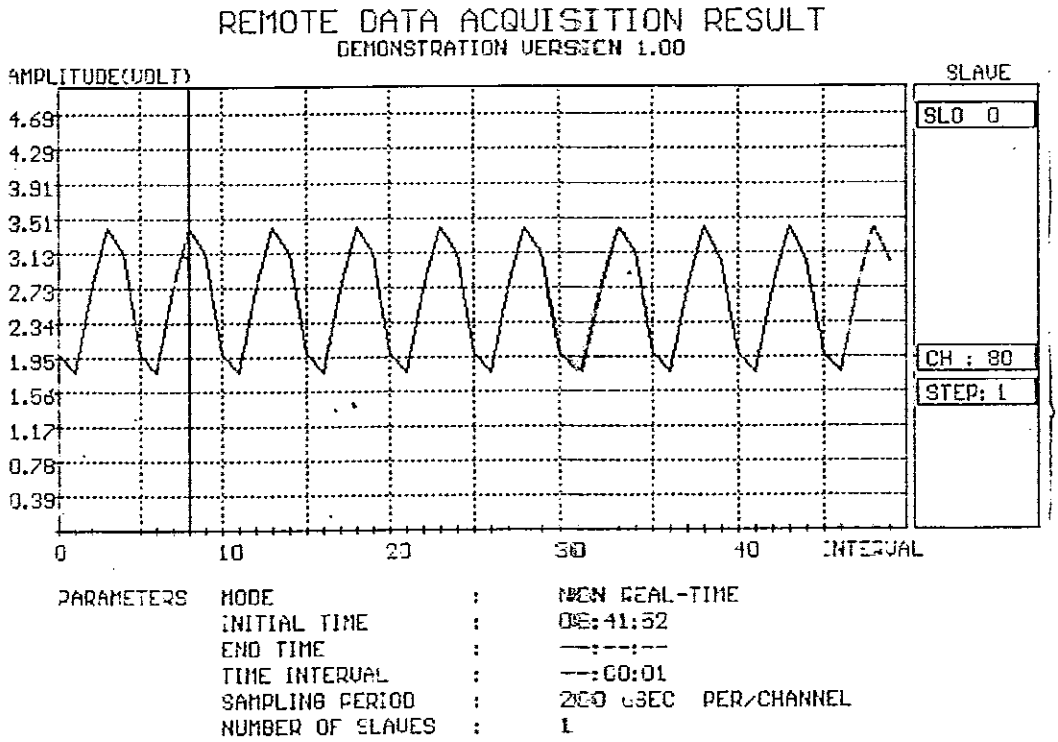
รูปที่ 5.11 กราฟแสดงผลของการสุ่มอ่านสัญญาณความถี่ 750 เฮิร์ต ความสูงสัญญาณ 1.5 โวลท์

ความถี่ที่คำนวณได้จากกราฟผลการอ่านค่า

สัญญาณตามรูปมี AMPLITUDE ที่บรรจุ 3 ลูกคลื่น ที่ช่องสัญญาณ 5 และ 25 ซึ่งมีระยะห่าง 20 ช่อง หรือ 20 X 200/3 ไมโครวินาที

ดังนั้น คาบเวลาของสัญญาณจากกราฟผลการสุ่มอ่าน เป็น 1.333 มิลลิวินาที

หรือเทียบเท่ากับความถี่ 750 เฮิร์ต ซึ่งยังเป็นสัญญาณที่ไม่ผิดเพี้ยนจากการสุ่มอ่าน



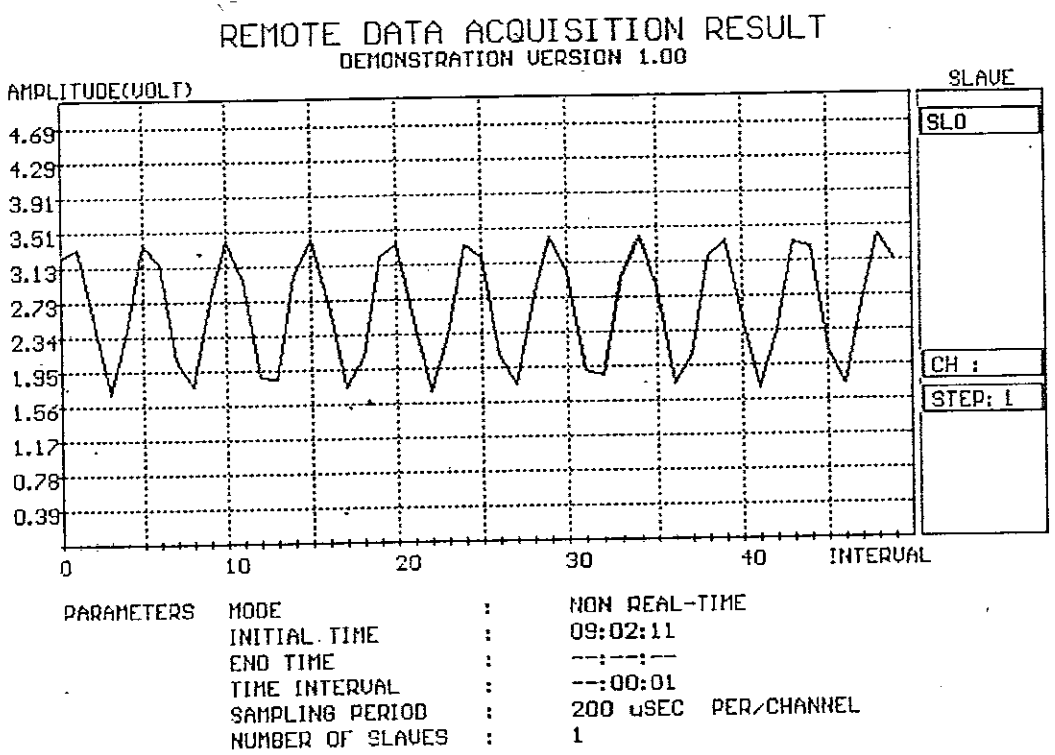
รูปที่ 5.12 กราฟแสดงผลของการสุ่มอ่านสัญญาณความถี่ 1000 เฮิร์ต ความสูงสัญญาณ 1.5 โวลท์

ความถี่ที่คำนวณได้จากกราฟผลการอ่านค่า

สัญญาณตามรูปมี AMPLITUDE ที่ช่องสัญญาณ 3 และ 8 ซึ่งมีระยะห่าง 5 ช่อง หรือ 5 X 200 ไมโครวินาที

ดังนั้น คาบเวลาของสัญญาณจากกราฟผลการสุ่มอ่าน เป็น 10 มิลลิวินาที

หรือเทียบเท่ากับความถี่ 1000 เฮิร์ต ซึ่งยังเป็นสัญญาณที่ไม่ผิดเพี้ยนจากการสุ่มอ่าน



รูปที่ 5.13 กราฟแสดงผลของการสุ่มอ่านสัญญาณความถี่ 1050 เฮิร์ต
ความสูงสัญญาณ 1.5 โวลท์

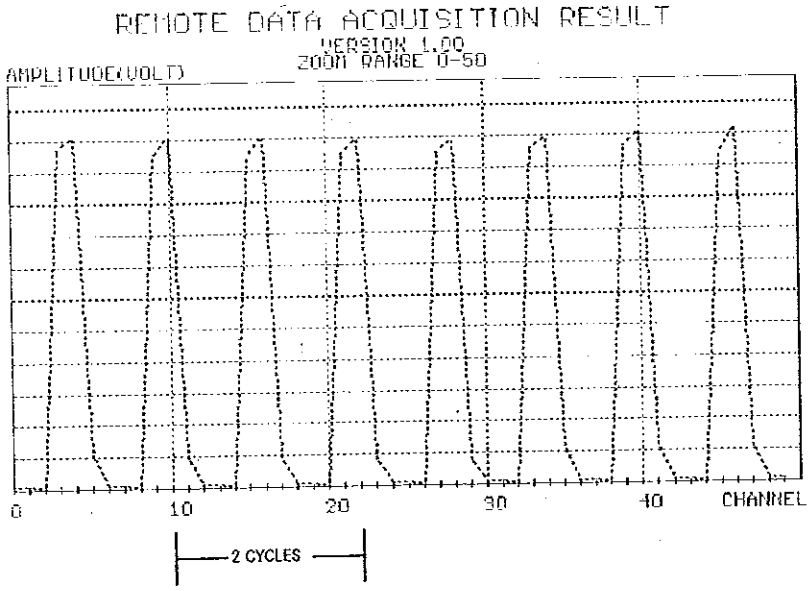
ไม่สามารถระบุ AMPLITUDE ของสัญญาณที่แน่นอนได้เนื่องจากมีความผิดเพี้ยนของสัญญาณจนทำให้ไม่มีจุดซ้ำซ้อนของรูปแบบสัญญาณที่แน่นอน

จากผลของการสุ่มอ่านสัญญาณทดสอบในรูปที่ 5.9-5.13 ทำให้สามารถระบุได้ว่าในรูปแบบการทำงานแบบไม่เป็นเวลาจริงนั้น สัญญาณหรือข้อมูลที่ต้องการจัดเก็บจากการทดลองจะต้องมีความเปลี่ยนแปลงของคาบเวลาไม่เกิน 1 กิโลเฮิร์ต โดยใช้อัตราการสุ่มอ่านข้อมูลสูงสุดที่ 200 ไมโครวินาที

1.2 การทดสอบการทำงานของระบบเก็บข้อมูล ในรูปแบบการทำงานแบบเวลาจริง

การทดสอบในขั้นตอนนี้อาศัยรูปแบบการจัดระบบ เพื่อการทดสอบเช่นเดียวกับการทดสอบการทำงานในรูปแบบไม่เป็นเวลาจริงที่แสดงไปแล้วในรูปที่ 5.1

เนื่องจากการทำงานในรูปแบบเวลาจริง คาบของการสุ่มอ่านค่าสัญญาณจะเริ่มจาก 1/30 วินาที ไปจนถึงค่าสูงสุดของการสุ่มอ่านเป็น 10 นาที ในการทดสอบได้เลือกทดสอบระบบในคาบของการทดลองสูงสุด ที่สามารถใช้งานในรูปแบบการทำงานนี้ ผลของการทดสอบแสดงในรูปที่ 5.14 เป็นต้นไป



รูปที่ 5.14 กราฟผลของการสุ่มอ่านสัญญาณรูปอื่น ๆ ที่ได้จากเครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 10 เฮิร์ต

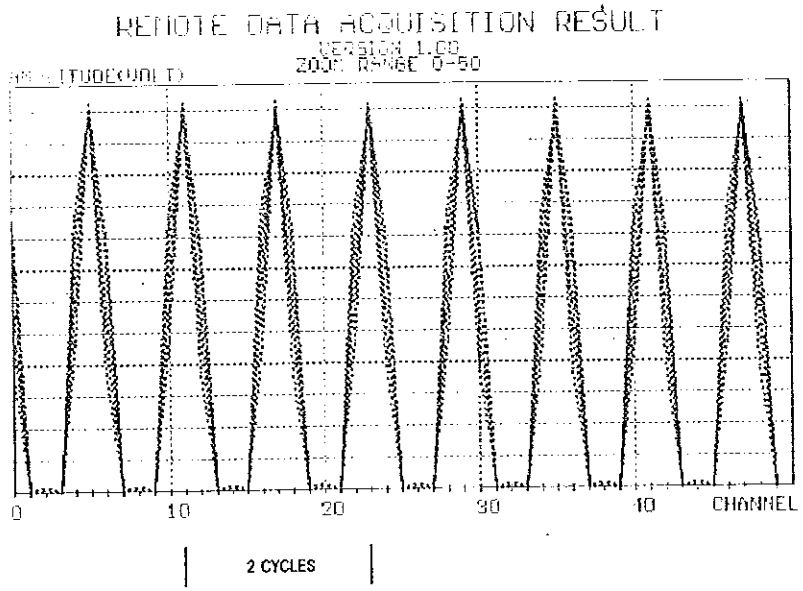
รูปแบบการเก็บข้อมูลแบบเป็นเวลาจริง

ใช้จำนวนระบบเครือข่ายย่อย 1 ตัว คาบเวลาของการสุ่มอ่านข้อมูล 1/60 วินาที

ค่าคาบเวลาของสัญญาณ และความถี่ที่ได้จากการคำนวณของกราฟในรูปที่ 5.14

คาบเวลาสัญญาณ = (1/60 วินาที x 12)/2 = 0.1 วินาที

ดังนั้น ความถี่สัญญาณที่อ่านได้เป็น 10 เฮิร์ต



รูปที่ 5.15 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จาก
เครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 10 เฮิร์ต

รูปแบบการเก็บข้อมูลแบบเป็นเวลาจริง

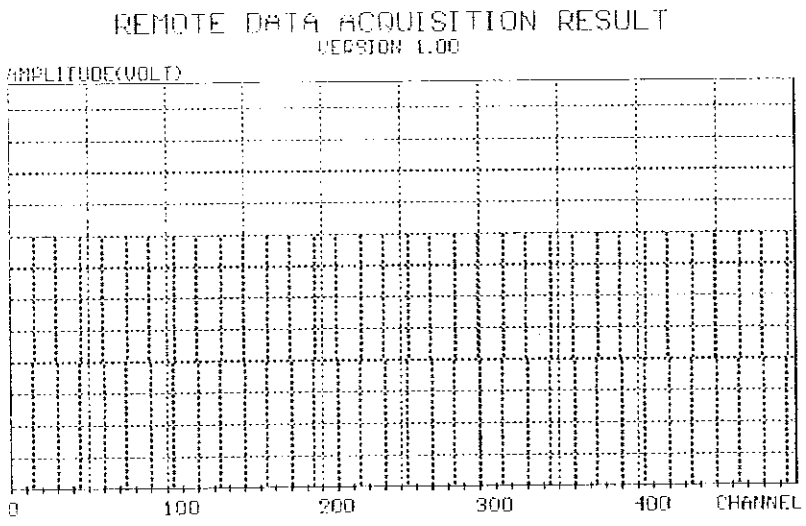
ใช้จำนวนระบบเครือข่ายย่อย 7 ตัว คาบเวลาของการสุ่มอ่านข้อมูล 1/60 วินาที

ค่าคาบเวลาของสัญญาณ และความถี่ที่ได้จากการคำนวณของกราฟใน รูปที่ 5.15

คาบเวลาสัญญาณ = (1/60 วินาที x 12)/2 = 0.1 วินาที

ดังนั้น ความถี่สัญญาณที่อ่านได้เป็น 10 เฮิร์ต

สำหรับความผิดเพี้ยนของรูปแบบสัญญาณที่ได้จากสัญญาณอินพุท ที่มีรูปแบบเป็น SINE WAVE เกิดจากการเลือกขยายช่องสัญญาณของโปรแกรมอ่านข้อมูล



รูปที่ 5.16 กราฟผลของการสุ่มอ่านสัญญาณรูป SQUARE ที่ได้จาก เครื่องกำหนดสัญญาณความถี่

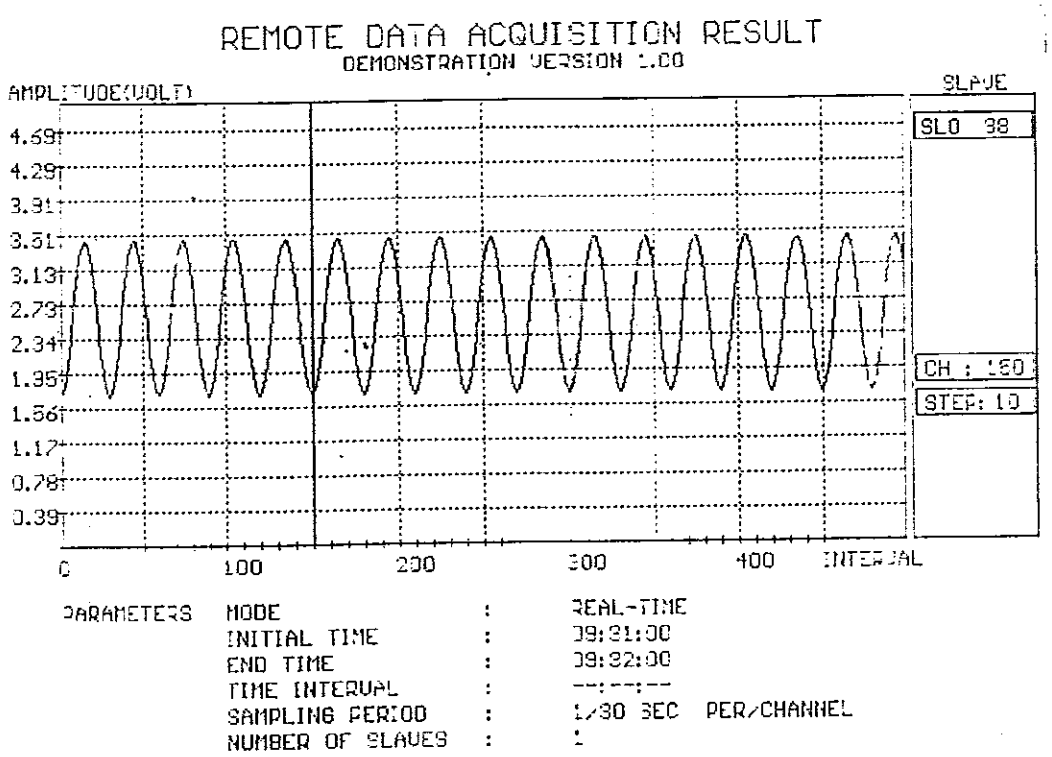
กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 1 เฮิร์ต

รูปแบบการเก็บข้อมูลแบบเป็นเวลาจริง

ใช้จำนวนระบบเครือข่ายย่อย 1 ตัว คาบเวลาของการสุ่มอ่านข้อมูล 1/30 วินาที

การคำนวณคาบเวลาของสัญญาณที่อ่านค่าได้จากรูปที่ 5.16

คาบเวลาของสัญญาณ = (1/30 วินาที x 30) = 1 วินาที

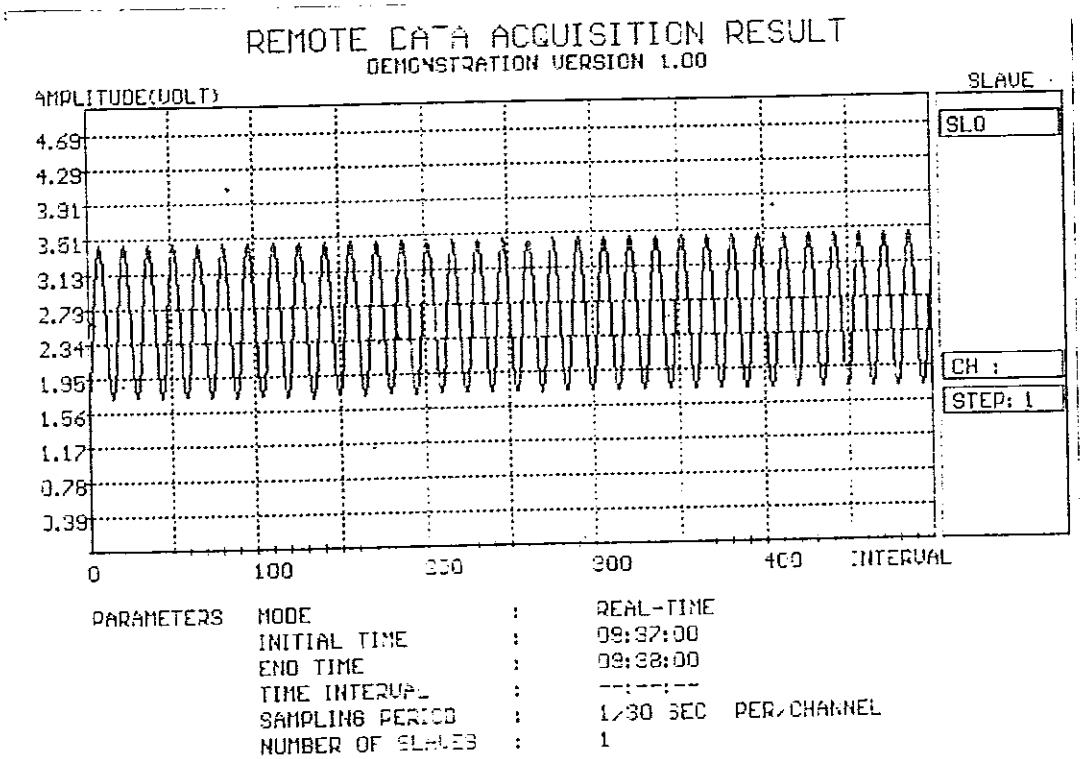


รูปที่ 5.17 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จาก
เครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 1 เฮิร์ต

การคำนวณคาบเวลาของสัญญาณที่อ่านค่าได้จากรูปที่ 5.17

คาบเวลาของสัญญาณ = (1/30 วินาที x 30) = 1 วินาที

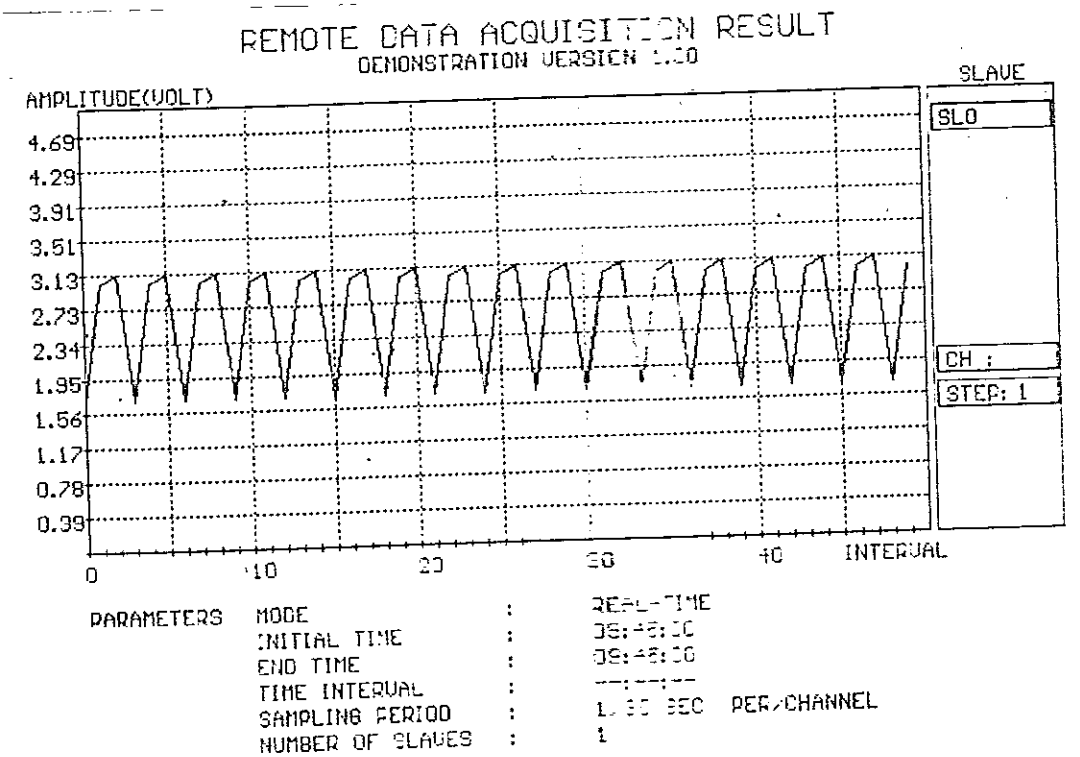


รูปที่ 5.18 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จากเครื่องกำเนิดสัญญาณความถี่

กำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 2 เฮิร์ต

การคำนวณคาบเวลาของสัญญาณที่อ่านค่าได้จากรูปที่ 5.18

คาบเวลาของสัญญาณ = (1/30 วินาที x 15) = 0.5 วินาที

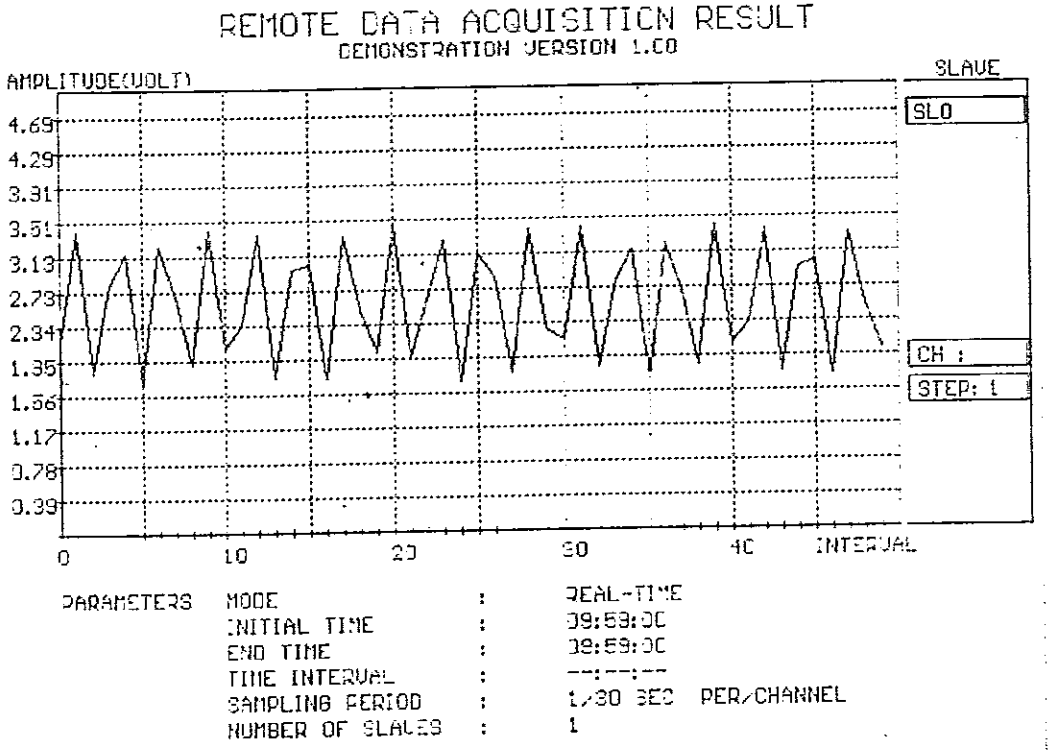


รูปที่ 5.19 กราฟผลความผิดเพี้ยนของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จากเครื่องกำเนิดสัญญาณความถี่เมื่อกำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 10 เฮิร์ต

การคำนวณคาบเวลาของสัญญาณที่อ่านค่าได้จากรูปที่ 5.19

คาบเวลาของสัญญาณ = (1/30 วินาที x 3) = 0.3 วินาที

จากรูปที่ 5.19 จะเห็นได้ว่าผลของการสุ่มอ่านเมื่อนำมาสร้างกราฟจะได้รูปแบบที่มีความผิดเพี้ยน แต่ยังสามารถระบุคาบเวลาที่ถูกต้องได้



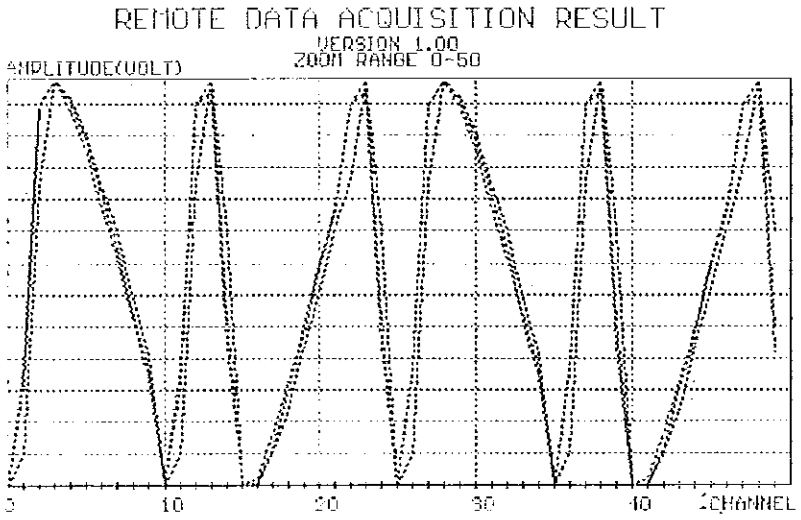
รูปที่ 5.20 กราฟผลความผิดเพี้ยนของการสุ่มอ่านสัญญาณรูป SINE ที่ได้จากเครื่องกำเนิดสัญญาณความถี่เมื่อกำหนดเงื่อนไขของสัญญาณป้อนเข้าความถี่ 11 เฮิร์ต

จากกราฟในรูป 5.20 ซึ่งแสดงให้เห็นความผิดเพี้ยนของการสุ่มอ่านที่ไม่สามารถระบุความถี่สัญญาณที่แท้จริงได้

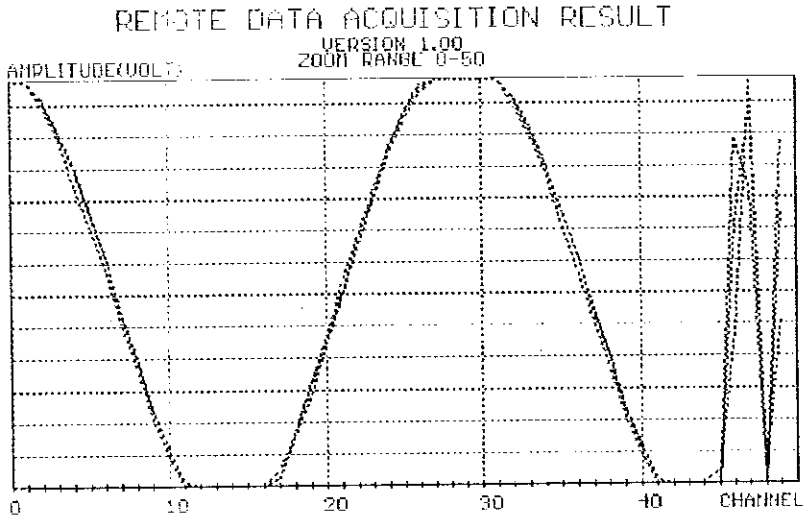
ดังนั้น ในรูปแบบการทำงานแบบเวลาจริง ที่อัตราการสุ่มอ่านข้อมูลสูงสุด 1/30 วินาที ข้อมูลที่ต้องการบันทึกจากการทดลองในกรณีที่มีความเปลี่ยนแปลงเป็นคาบเวลา จะต้องมีความถี่ของความเปลี่ยนแปลงไม่เกิน 10 เฮิร์ต

1.3 การทดสอบระบบเก็บข้อมูลระยะไกลกับความถี่ที่เปลี่ยนแปลงค่า

ในการทดสอบรูปแบบนี้ใช้จำนวนระบบเครือข่ายย่อย 5 ตัว เพื่อทำการสุ่มอ่านสัญญาณที่ได้จากเครื่องกำเนิดสัญญาณความถี่ของบริษัทSTANFORD RESEARCH SYSTEMS จำกัด เช่นเดียวกับที่ทดสอบในหัวข้อ 1.1 และ 1.2 โดยปรับการทำงานเป็นรูปแบบของการกำเนิดความถี่แบบ FSK(FREQUENCY SHIFT KEYING) ซึ่งเป็นสัญญาณ รูปแบบของสัญญาณที่ได้จากการจัดเก็บเปรียบเทียบกับสัญญาณจริงที่ได้โปรแกรมค่าไว้

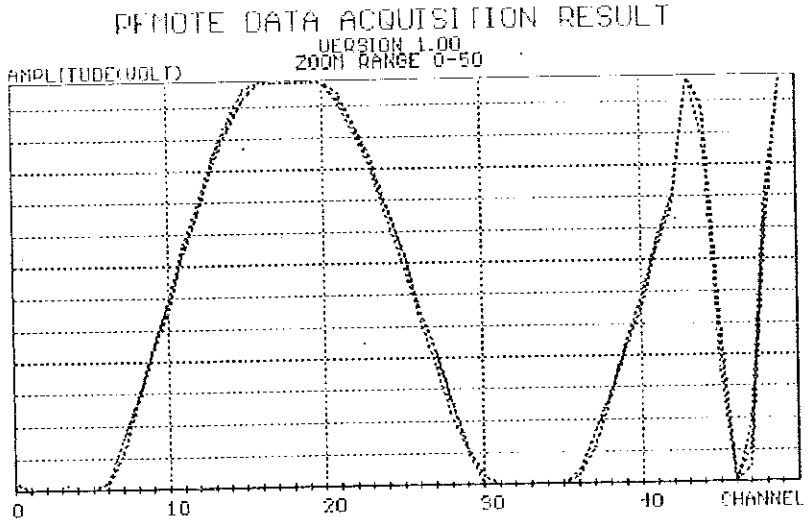


รูปที่ 5.21 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE แบบ FSK
ความถี่ 1 และ 5 กิโลเฮิร์ต
กำหนดอัตราการเปลี่ยนแปลงความถี่ (SWEEP RATE) 200 เฮิร์ต
ระดับความสูงของสัญญาณเท่ากับ 5 โวลท์
รูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง
โดยใช้คาบเวลาในการสุ่มอ่านข้อมูล 400 ไมโครวินาที



รูปที่ 5.23 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE แบบ FSK ความถี่ 1 และ 20 เฮิร์ต แสดงการขยายในบางส่วนของสัญญาณ 50 ช่องสัญญาณแรก

จากรูปที่ 5.23 ซึ่งหากคำนวณค่าความถี่ของสัญญาณที่ได้ จะเท่ากับ 1 เฮิร์ต และ 10 เฮิร์ตตามลำดับ ซึ่งผิดไปจากค่าสัญญาณป้อนเข้า สาเหตุที่มีความผิดพลาดก็เพราะคาบของการสุ่มอ่านข้อมูลที่ใช้นั้น ไม่สามารถใช้ในการสุ่มอ่านข้อมูลที่มีความเปลี่ยนแปลงเร็วมากกว่า 10 เท่าได้



รูปที่ 5.24 กราฟผลของการสุ่มอ่านสัญญาณรูป SINE แบบ FSK
ความถี่ 1 และ 5 เฮิร์ต
กำหนดอัตราการเปลี่ยนแปลงความถี่ (SWEEP RATE) 0.1 เฮิร์ต
ระดับความสูงของสัญญาณเท่ากับ 5 โวลท์
รูปแบบการเก็บข้อมูลแบบเวลาจริง
โดยใช้คาบเวลาในการสุ่มอ่านข้อมูล 1/30 วินาที

ข้อเสนอแนะ

จากการทดสอบการทำงานของระบบเก็บข้อมูลระยะไกล โดยใช้ไมโครโปรเซสเซอร์ ซึ่งผลของการตอบสนองต่อความถี่ และระดับความสูงของสัญญาณป้อนเข้าได้ผลเป็นที่น่าพอใจดังรายละเอียดในหัวข้อ 1.1 ถึง 1.3 แต่ยังมีรายละเอียดบางประการ ที่ผู้วิจัยคาดว่าคงเป็นประโยชน์ในการพัฒนาระบบหรือแก้ไขเพื่อใช้งาน ดังนี้

1. โปรแกรมควบคุมการทำงาน ซึ่งเชื่อมต่อกับระบบควบคุมหลักเก็บข้อมูลระยะไกล ซึ่งทำงานบนไมโครคอมพิวเตอร์ ผู้วิจัยได้ทำส่วนเริ่มต้นเพื่อใช้ในการอ่านข้อมูลจากหน่วยความจำส่วนเก็บข้อมูลของระบบควบคุมการทำงานหลัก ซึ่งแทนหน่วยความจำของระบบเครือข่ายย่อยแต่ละตัว เพื่อใช้ในการพิมพ์รูปแบบของสัญญาณที่ได้จากการสุ่มอ่านลงในรายงานวิทยานิพนธ์ยังไม่มีความสามารถสูงพอทั้ง ๆ ที่ตามความเป็นจริงแล้วระบบควบคุมหลักของระบบเก็บข้อมูลระยะไกล สามารถรับค่าตัวแปรสั่งงานจากไมโครคอมพิวเตอร์ได้ ซึ่งคาดว่าคงได้มีการพัฒนาโปรแกรมในส่วนนี้ สอดคล้องกับรหัสเพรมคำสั่งที่กล่าวถึงในภาคผนวก

2. วงจรแปลงรูปแบบสัญญาณจากอนาลอกเป็นดิจิตอลที่ใช้ ซึ่งมีขนาด 8 บิท น่าจะได้มีการพัฒนาโดยการเปลี่ยนเป็นขนาด 12 บิท เพื่อผลของระดับความสูงของสัญญาณที่ได้จะมีรายละเอียดมากขึ้น

3. ในระบบเครือข่ายย่อยควรมีรูปแบบโปรโตคอลที่แจ้งให้ระบบควบคุมหลักทราบถึงสภาวะที่หน่วยความจำเต็ม เพื่อขอบริการโอนถ่ายข้อมูลในบล็อกหน่วยความจำก่อนที่จะมีการเก็บข้อมูลต่อไป

คำแนะนำในการดำเนินงานต่อไป

สำหรับผู้ที่ต้องการพัฒนาระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์ เพื่อเพิ่มประสิทธิภาพในการทำงาน โดยอาศัยพื้นฐานจากงานวิทยานิพนธ์นี้ ผู้วิจัยขอเสนอแนะขอบเขตของงานวิจัยในลำดับขั้นต่อไป ดังนี้

1. เปลี่ยนวงจรแปลงรูปแบบสัญญาณจากอนาลอกเป็นดิจิตอล เป็นขนาดบัสข้อมูล 12 บิท และปรับปรุงระบบเครือข่ายย่อยให้สามารถอ่าน และจัดเก็บข้อมูลของการแปลงสัญญาณจากอนาลอกเป็นดิจิตอลแบบ 12 บิทได้

2. พัฒนาโปรแกรมสั่งงานบนไมโครคอมพิวเตอร์และส่วนของโปรแกรมที่ทำหน้าที่ในการรายงานผลข้อมูลแบบเวลาจริง ภายใต้รหัสโปรโตคอลที่กำหนด

3. ออกแบบวงจรส่วนข้อมูลอนาลอกให้เป็นมาตรฐาน ซึ่งประกอบด้วยวงจรบัฟเฟอร์สัญญาณ วงจรกรองความถี่สัญญาณเฉพาะช่วง และวงจรในการปรับอัตราการขยายสัญญาณที่ใช้ข้อมูลดิจิตอลควบคุม โดยผ่านชิพแปลงสัญญาณดิจิตอลเป็นอนาลอก

บรรณานุกรม

มณเทพ เกียรติวีระสกุล. 2533. "การออกแบบเครื่องควบคุมการให้สารน้ำทาง-หลอดเลือดดำ", วิทยานิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต สาขาวิชาวิศกรรมไฟฟ้าบัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์.

Ayala, K.J. 1992. The 8051 Microcontroller:Architecture, Programming,and Application. Minnesota:West Publiching.

Driscoll, F.F. 1992. Data Communication. Fort Worth:Saunders College Publishing.

Hall, D.V. 1987. Microprocessor and Interfacing. New York: McGraw-Hill.

Hintz, K. and Tabak D. 1992. Microcontroller:Architecture, Implementation,and Programming. New York:McGraw-Hill.

Hyes, J.P. 1988. Computer Architecture and Organization. New York:McGraw-Hill.

Intel Corporation. 1985. Microcontroller Handbook. Santa Clara.

Peatman, J.B. 1988. Design with Microcontroller. New York: McGraw-Hill.

Rangan, C.S., Sarma, G.R. and Mani, V.S. 1983. Instrumentation :Device and Systems. New Delhi:Tata McGraw-Hill.

ภาคผนวก ก

ระบบพัฒนาไมโครโปรเซสเซอร์ตระกูล INTEL MCS-51
(INTEL MCS-51 EMULATOR)

ในการพัฒนาระบบควบคุมอิสระภายใต้การควบคุมโดยใช้ไมโครโปรเซส-เซอร์ หรือไมโครคอนโทรลเลอร์ (MICROPROCESSOR BASED STAND-ALONE CONTROL SYSTEM) หลังจากที่พัฒนาโปรแกรมควบคุมการทำงานแล้ว สิ่งที่จำเป็นต้องทำก่อนที่จะใช้ระบบนั้น ๆ เป็นระบบอิสระก็คือ การทดสอบการทำงานโดยใช้ระบบที่มีฮาร์ดแวร์ครบถ้วนตามที่ออกแบบไว้ อุปกรณ์ที่เข้ามาช่วยในงานการทดสอบนั้นก็คือ ระบบพัฒนาไมโครโปรเซสเซอร์ (EMULATOR) หรือระบบสังเคราะห์การทำงานของไมโครโปรเซสเซอร์ (SIMULATOR)

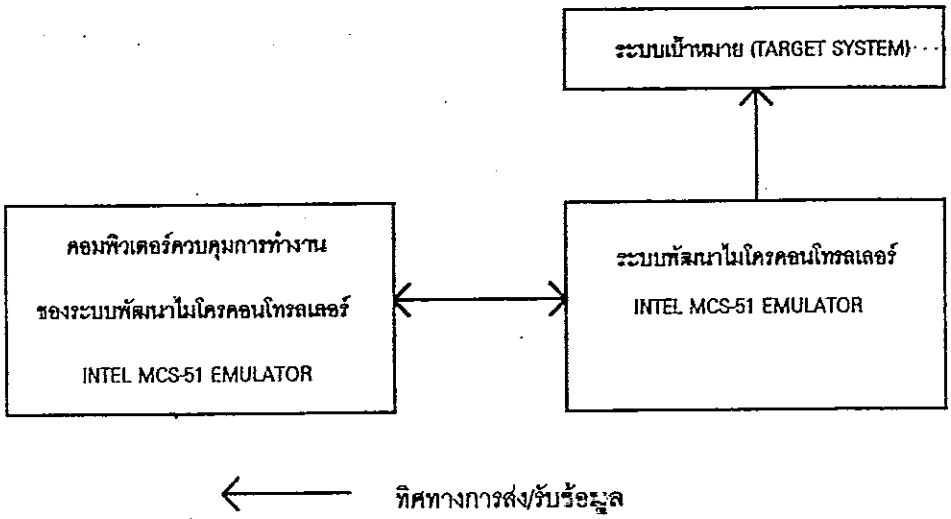
ระบบสังเคราะห์การทำงานของไมโครโปรเซสเซอร์ เป็นซอร์ฟแวร์สำเร็จรูปที่ทำงานบนไมโครคอมพิวเตอร์ โดยทั่วไปแล้วผู้ใช้งานสามารถเห็นความเปลี่ยนแปลงระดับรีจิสเตอร์ หน่วยความจำของระบบ และสถานะของพอร์ทต่าง ๆ หลังจากที่ไมโครโปรเซสเซอร์ทำงานตามคำสั่งใด ๆ มีรูปแบบการทำงานให้เลือกหลายรูปแบบ เช่น การสั่งทำงานทีละคำสั่ง (SINGLE STEP) ระบบสังเคราะห์การทำงานมีข้อจำกัดในหลายด้านด้วยกันซึ่งแตกต่างกันไปตามบริษัทที่พัฒนา

ระบบพัฒนาไมโครโปรเซสเซอร์(EMULATOR)มีข้อดีกว่าระบบสังเคราะห์การทำงานในแง่ที่ว่ากระบวนการเปลี่ยนแปลงทุกอย่าง เป็นสิ่งที่เกิดขึ้นจริงจากการทำงานของไมโครโปรเซสเซอร์นั้น ทำให้สามารถพัฒนาระบบได้ง่ายกว่าแต่มีข้อเสียคือ ราคาของชุดพัฒนาที่สมบูรณ์แบบนั้นค่อนข้างแพง ส่วนชุดพัฒนาที่มีราคาถูกกว่านั้นเท่าที่ผู้วิจัยได้ทดลองใช้ NICE-51 ของบริษัท SUNSHINE ELECTRONICS (TAIWAN) พบความไม่คล่องตัวหลายประการ เช่น ไม่สามารถทดสอบการสื่อสารข้อมูลผ่านพอร์ทสื่อสารข้อมูลแบบอนุกรมระหว่างระบบที่กำลังพัฒนา กับโปรแกรมทดสอบการสื่อสารได้ เนื่องจากการทดสอบดังกล่าวจำเป็นต้องใช้โปรแกรมภายนอกส่งค่าข้อมูลจึงต้องออกจากโปรแกรมควบคุมระบบพัฒนา ทำให้ไม่สามารถควบคุมการทำงานได้อีกต่อไป

จากข้อดีและข้อจำกัดของการใช้งานในระดับหนึ่ง ทำให้ผู้วิจัยจำเป็นต้องออกแบบระบบพัฒนาไมโครโปรเซสเซอร์ ประเภทไมโครคอนโทรลเลอร์ตระกูล MCS-51 ของบริษัท INTEL ขึ้นมาเอง เพื่อให้สามารถทดสอบการทำงานตามที่ต้องการได้

1. แนวคิดในการออกแบบระบบพัฒนาไมโครโปรเซสเซอร์ (EMULATOR)

โดยการใช้หลักการง่าย ๆ คือ ไมโครโปรเซสเซอร์ จะทำงานตามโปรแกรมซึ่งปกติจะถูกเก็บไว้ในหน่วยความจำแบบอ่านได้อย่างเดียว (READ ONLY MEMORY=ROM) ผู้วิจัยได้ออกแบบ และสร้างระบบที่ทำให้ไมโครโปรเซสเซอร์ที่ถูกพัฒนา ซึ่งปกติจะถูกเรียกว่าเป็นระบบเป้าหมาย (TARGET-SYSTEM) มองเห็นหน่วยความจำของระบบพัฒนาเป็น ROM ในขณะที่ไมโครคอมพิวเตอร์ที่เป็นตัวควบคุมระบบพัฒนา (โดยผ่านทางโปรแกรมควบคุมการทำงาน) นี้จะมองเห็นหน่วยความจำส่วนนั้นเป็นหน่วยความจำแบบอ่านเขียนได้ (READ/WRITE MEMORY : RANDOM ACCESS MEMORY = RAM) ดังนั้นโปรแกรมสั่งงานไมโครโปรเซสเซอร์ของระบบเป้าหมาย (MONITOR PROGRAM) จะถูกส่งออกจากไมโครคอมพิวเตอร์เข้าสู่หน่วยความจำของระบบเป้าหมาย ซึ่งจะถูกสั่งให้ทำงานต่อไป รูปแบบของการสั่งงานและทิศทางของการส่งและอ่านข้อมูลเป็นไปตามรูปที่ 1



รูปที่ 1 ทิศทางการส่งและอ่านข้อมูลของระบบพัฒนาไมโครคอนโทรลเลอร์ 8031

สำหรับวงจรควบคุมระบบพัฒนาไมโครโปรเซสเซอร์ INTEL MCS-51 EMULATOR ที่ออกแบบขึ้นนั้นได้ใช้อุปกรณ์ที่สามารถหาได้ง่าย โดยได้เลือกใช้พอร์ทอเนกประสงค์ 8255 ทำหน้าที่ในการสร้างสัญญาณอ้างอิงตำแหน่งข้อมูล (ADDRESS BUS) เทียม ซึ่งจะถูกสั่งงานจากไมโครคอมพิวเตอร์ที่เป็นตัวควบคุมการทำงานของระบบพัฒนาภายใต้การควบคุมโดยโปรแกรมสั่งงานข้อมูล (DATA) ที่จะถูกส่งไปยังตำแหน่งที่อ้างอิงจะถูกส่งผ่านพอร์ทที่ถูกกำหนดให้เป็นพอร์ทข้อมูล โดยระบบพัฒนาที่ออกแบบขึ้นสามารถเก็บข้อมูล ซึ่งเป็นโปรแกรมที่กำลังพัฒนาได้ 64 กิโลไบท์ (KBYTE) ซึ่งจะเท่ากับขนาดสูงสุดที่ไมโครคอนโทรลเลอร์ MCS-51 ของ INTEL สามารถอ้างอิงโปรแกรมได้ การอ้างอิงตำแหน่งข้อมูลจะถูกเลือกระหว่างสัญญาณที่มาจากไมโครคอลโทรลเลอร์เอง และที่มาจากการสร้างโดยพอร์ท 8255 ทิศทางการเลือกถูกควบคุมโดยคอมพิวเตอร์ หลังจากที่ข้อมูลซึ่งเป็นโปรแกรมสั่งงานไมโครคอนโทรลเลอร์ถูกส่งเข้าสู่หน่วยความจำของระบบพัฒนาเรียบร้อยแล้ว ก็สามารถสั่งให้ระบบทำงานเพื่อทดสอบได้

2. โปรแกรมควบคุมการทำงานของระบบพัฒนาไมโครคอนโทรลเลอร์ INTEL MCS-51

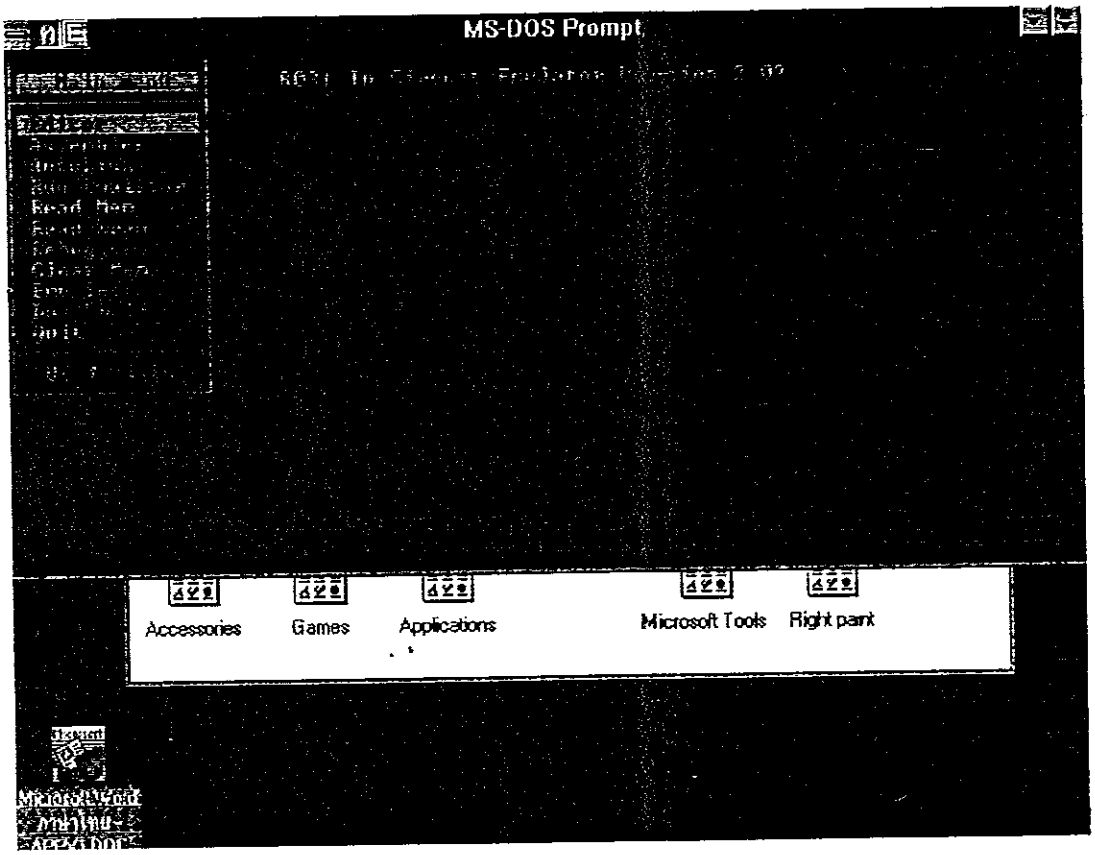
ชื่อโปรแกรม	NEW31.EXE
วัตถุประสงค์	เพื่อควบคุมการทำงานของระบบพัฒนาไมโครคอนโทรลเลอร์ INTEL MCS-51

อุปกรณ์ที่เกี่ยวข้องกับโปรแกรม

คอมพิวเตอร์ที่ใช้	IBM PC AT หรือ COMPATIBLE
ระบบปฏิบัติการ	DOS 3.3 ขึ้นไป
อุปกรณ์แสดงผล	EGA ขึ้นไป
หน่วยความจำ	640 กิโลไบท์ ขึ้นไป
อุปกรณ์ที่จำเป็นเพิ่มเติม	การ์ดถอดรหัสตำแหน่ง 0204H-0300H และแผงวงจรควบคุมระบบพัฒนาไมโครคอนโทรลเลอร์ INTEL MCS-51

ขั้นตอนการทำงานของโปรแกรม

โปรแกรม NEW31.EXE เป็นโปรแกรมที่รวบรวมการเรียกโปรแกรมที่เกี่ยวข้องกับการพัฒนาระบบไมโครคอนโทรลเลอร์ INTEL MCS-51ไว้ในโปรแกรมเดียวกัน เพื่อให้เกิดความสะดวกในการพัฒนา หลังจากที่ผู้ใช้งานพัฒนาโปรแกรมเสร็จสิ้นแล้ว สามารถส่งโปรแกรมที่กำลังพัฒนา เข้าสู่ระบบพัฒนาไมโครคอนโทรลเลอร์ได้โดยตรง และนอกจากนี้ยังเพิ่มเติมการอ่านข้อมูลในหน่วยความจำของระบบเป้าหมาย เพื่อตรวจสอบความถูกต้องหรือแก้ไขข้อมูลในบางส่วนที่เป็นตัวแปร เพื่อให้เป็นโปรแกรมที่ใช้งานได้ง่าย ในบรรทัดสุดท้ายของการแสดงผลของโปรแกรมจะมีคำอธิบายเมนู (MENU) ที่กำลังแสดงอยู่ในขณะนั้น สำหรับรูปแบบของการแสดงผลของโปรแกรมในเมนูหลักจะเป็นดังรูปที่ 2



รูปที่ 2 รูปแสดงส่วนหน้าจอของโปรแกรมควบคุมระบบพัฒนาไมโครคอนโทรลเลอร์ 8031

คำสั่งในเมนูหลัก

EDITOR : เมื่อต้องการสร้าง หรือแก้ไขโปรแกรมภาษาแอสเซมบลี (ASSEMBLY)

ASSEMBLER : คำสั่งสำหรับเรียก โปรแกรมแอสเซมเบลอร์ (ASSEMBLER PROGRAM) เพื่อสร้างภาษาเครื่อง(MACHINE CODE) จากโปรแกรมภาษาแอสเซมบลี

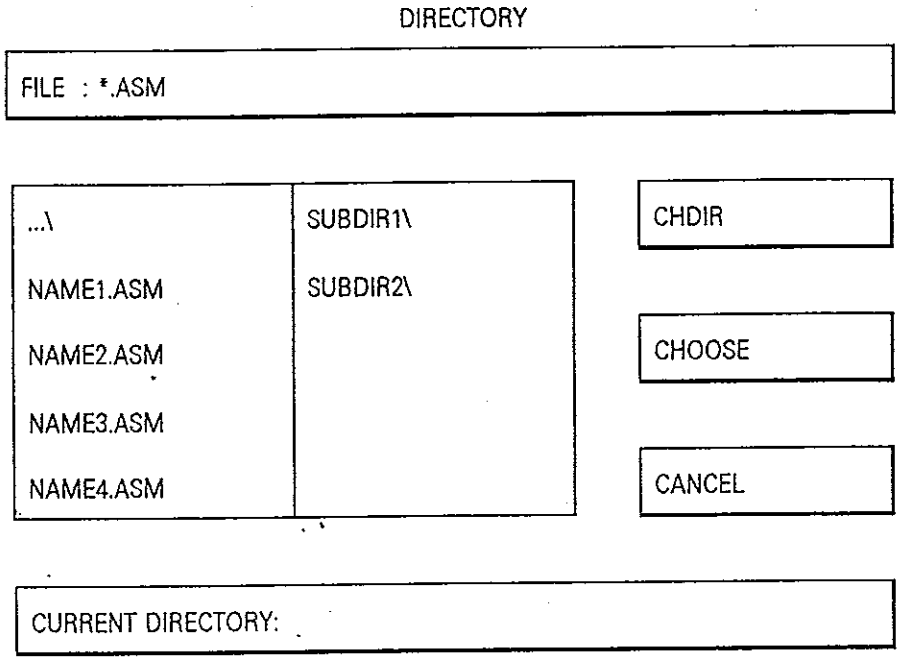
AUTOLINK : คำสั่งสำหรับส่งโปรแกรมภาษาเครื่อง เข้าสู่ระบบพัฒนา (EMULATOR) ซึ่งเป็นอุปกรณ์ฮาร์ดแวร์ภายนอก

RUN EMULATOR	:	คำสั่งให้ระบบพัฒนาทำงานในรูปแบบใช้งานจริง (FREE RUNNING MODE)
READ MEM	:	ต้องการอ่าน หรือเปลี่ยนแปลงข้อมูลในส่วนของโปรแกรมที่กำลังพัฒนา
READ XRAM	:	อ่านข้อมูลในหน่วยความจำ ของระบบพัฒนาที่ได้จากการทำงานของระบบพัฒนา
DEBUGGER	:	ทำการ DEBUG การทำงานของระบบพัฒนา
CLEAR MEM	:	ลบโปรแกรมทั้งหมดที่ได้ส่งเข้าสู่ระบบพัฒนาแล้ว
EMU TEST	:	ทดสอบการทำงานบางประการของระบบพัฒนา
DOS SHELL	:	กลับสู่ระบบปฏิบัติการ (DOS) ชั่วคราว
QUIT	:	ออกจากโปรแกรมควบคุมระบบพัฒนาเข้าสู่ระบบปฏิบัติการ โดยที่ระบบพัฒนายังคงทำงานในรูปแบบใช้งานจริง (FREE RUNNING MODE)

พิจารณาการทำงานของคำสั่ง

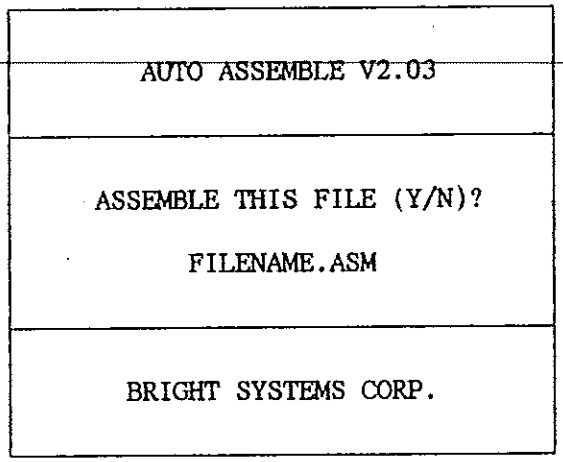
1. คำสั่ง EDITOR

หลังจากที่ผู้ใช้งานเรียกคำสั่งนี้โปรแกรมจะแสดงรายชื่อแฟ้มข้อมูลทั้งหมดที่อยู่ในไดเรคทอรีที่กำลังใช้งาน (WORKING DIRECTORY) ที่มีนามสกุลเป็น .ASM เพื่อให้ผู้ใช้งานเลือกแฟ้มข้อมูลที่ต้องการแก้ไขหรือปรับปรุงโดยการใช้ระบบเมนูเป็นตัวเลือก รูปแบบของการแสดงรายชื่อแฟ้มข้อมูลจะเป็นดังนี้



หลังจากที่มีการเลือกชื่อแฟ้มข้อมูล ที่ต้องการแก้ไขหรือเปลี่ยนแปลงแล้ว โปรแกรมจัดการจะทำการเรียกเอดิเตอร์ ชื่อ Q EDITOR และดึงชื่อแฟ้มข้อมูลเข้าสู่ระบบโดยอัตโนมัติ

เมื่อผู้ใช้งานได้แก้ไขแฟ้มข้อมูลเสร็จเรียบร้อย ก็สามารถออกจากโปรแกรมเอดิเตอร์ โดยการใช้ฟังก์ชันคีย์ ALT-X เมื่อกลับเข้าสู่โปรแกรมควบคุม จะแสดงข้อความ



หมายเหตุ เนื่องจากโปรแกรมควบคุมชุดนี้ได้พัฒนาให้บริษัทไบร์ท ซิสเท็ม จำกัดด้วย

เมื่อผู้ใช้งานกด Y โปรแกรมควบคุมจะจัดการแอสเซมเบิล(ASSEMBLE) แฟ้มข้อมูลที่ถูกเรียกใช้ก่อนหน้านี้ เพื่อให้ได้เป็นแฟ้มข้อมูลภาษาเครื่อง (MACHINE CODE) ของ INTEL MCS-51 ที่มีนามสกุลเป็น .HEX การที่จะข้ามขั้นตอนของการทำ AUTO ASSEMBLE ทำได้โดยการกดแป้นพิมพ์ใด ๆ จะทำให้โปรแกรมควบคุมกลับเข้าสู่เมนูหลัก

หลังจากที่ผู้ใช้งานเลือกทำ AUTO ASSEMBLE และได้ .HEX FILE แล้วจะปรากฏข้อความ

AUTO LINKER V2.03
DO YOU WANT TO LINK THIS FILE (Y/N)? FILENAME.HEX
BRIGHT SYSTEMS CORP.

ถ้าผู้ใช้งานเลือกกด Y โปรแกรมจะทำการส่งข้อมูลให้กับระบบพัฒนาไมโครคอนโทรลเลอร์และหากกดแป้นพิมพ์อื่น ๆ จะกลับเข้าสู่เมนูหลัก

2. คำสั่ง ASSEMBLER

ในกรณีที่ผู้ใช้งานต้องการแฟ้มข้อมูลภาษาเครื่อง จากโปรแกรมภาษาแอสเซมบลีที่ได้พัฒนามาก่อนแล้ว โปรแกรมควบคุมสามารถจัดการทำให้โดยตรงได้โดยการเรียกใช้เมนูนี้ หลังจากที่ผู้ใช้งานเรียกใช้โปรแกรมจะแสดงรายชื่อแฟ้มข้อมูลทั้งหมดที่อยู่ในไดเรคทอรีที่กำลังใช้งาน(WORKING DIRECTORY) ที่มีนามสกุลเป็น .ASM เพื่อให้ผู้ใช้งานเลือกแฟ้มข้อมูลที่ต้องการแก้ไขหรือปรับปรุงโดยการใช้ระบบเมนูเป็นตัวเลือกรูปร่างของหน้าต่าง(WINDOW) แสดงรายชื่อแฟ้มข้อมูล .ASM จะเหมือนกับหน้าต่างแรกของการเรียกใช้คำสั่ง EDITOR ที่กล่าวมาแล้ว

กรณีที่ผู้ใช้งานไม่ต้องการเลือกแฟ้มข้อมูลใด ๆ และกลับเข้าสู่เมนูหลัก สามารถทำได้โดยการกดแป้นพิมพ์ ESC ถ้าผู้ใช้งานเลือกแฟ้มข้อมูล .ASM (โดยการเลื่อนแถบเมนูไปยังชื่อโปรแกรมที่ต้องการ)ที่ต้องการแอสเซมเบิลแล้วโปรแกรมควบคุมจะทำการเรียกโปรแกรมแอสเซมเบลอร์ (ASSEMBLER PROGRAM) ของบริษัท BINARY TECHNOLOGY INC ชื่อ SXA51 โดยการใส่ชื่อแฟ้มข้อมูลที่ต้องการแอสเซมเบิลและรูปแบบการทำแอสเซมเบิลให้โดยอัตโนมัติ และจะแสดงข้อความ

8051 CROSS-ASSEMBLER (1.3) COPYRIGHT (C)1987,1989 09:00:00 BINARY TECHNOLOGY,INC,MERIDEN,NH

เป็นการเข้าสู่กระบวนการของแอสเซมเบลอร์ (ASSEMBLER) สิ่งที่ได้จากการทำแอสเซมเบิลโปรแกรมก็คือ ได้แฟ้มข้อมูลภาษาเครื่องที่สอดคล้องกับภาษาแอสเซมบลีตั้งต้นและแฟ้มข้อมูลรายละเอียด (LISTING PROGRAM) ซึ่งบ่งบอกความผิดพลาดในการพัฒนาโปรแกรมส่วนของซอร์ฟแวร์

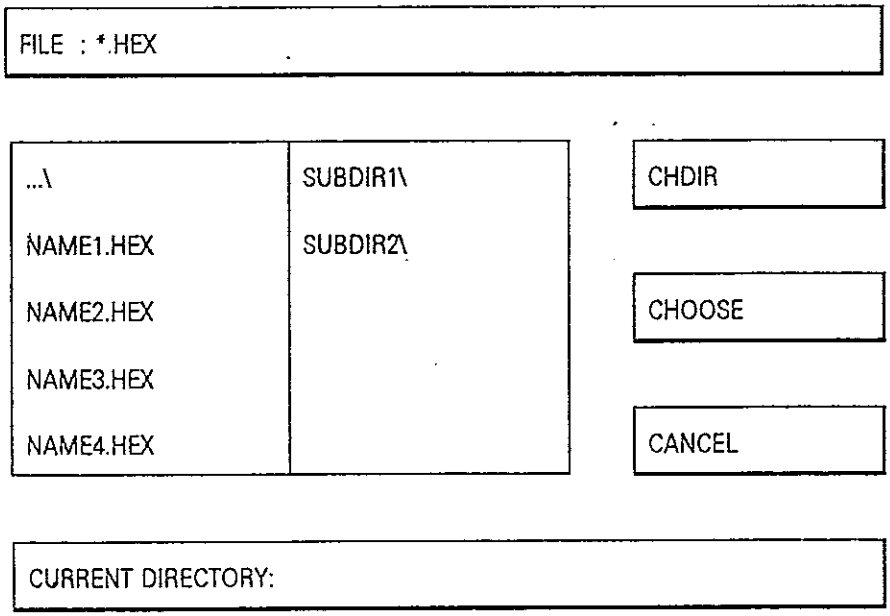
หลังจากที่การทำแอสเซมเบิลโปรแกรมเสร็จสิ้นแล้ว ให้ผู้ใช้งานกดแป้นพิมพ์ใด ๆ เพื่อกลับเข้าสู่ระบบของโปรแกรมควบคุม สิ่งที่ผู้ใช้งานจะได้พบต่อไปก็คือหน้าต่างของ AUTO LINK (เช่นเดียวกับ AUTO LINK ของคำสั่ง EDITOR) ผู้ใช้งานก็สามารถ

AUTO LINKER V2.03
DO YOU WANT TO LINK THIS FILN (Y/N)? FILENAME.HEX
BRIGHT SYSTEMS CORP.

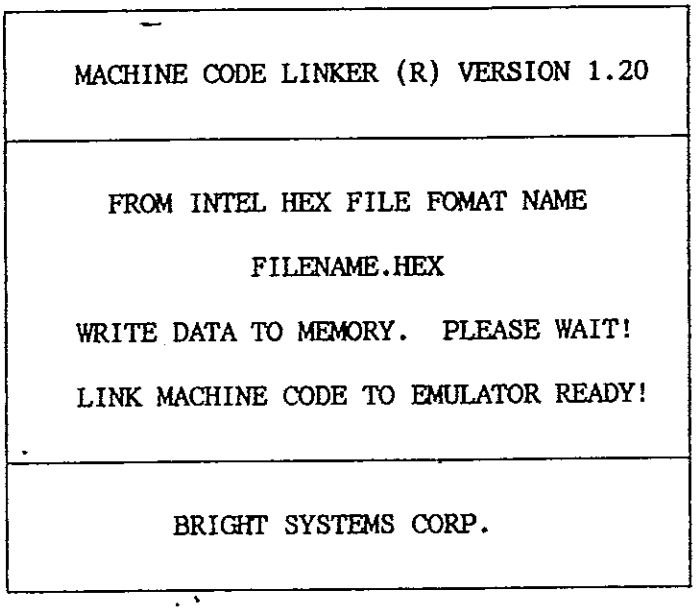
เลือกตัดสินใจได้ว่าจะส่งโปรแกรมภาษาเครื่องที่ได้เข้าสู่ระบบพัฒนาหรือไม่ (เลือก Y หรือแป้นพิมพ์ใด ๆ)

3. คำสั่ง AUTO LINK

เป็นคำสั่งที่เลือกเมื่อผู้ใช้งานมีแฟ้มข้อมูล .HEX ของโปรแกรมที่พัฒนาเสร็จแล้วโดยมีความถูกต้องในทางซอร์ฟแวร์และต้องการส่งให้กับระบบพัฒนา เพื่อทดสอบรูปแบบการทำงานจริง หลังจากที่เลือกใช้คำสั่งดังกล่าว โปรแกรมควบคุมจะแสดงรายชื่อแฟ้มข้อมูล .HEX ทั้งหมดที่มีอยู่



เพื่อให้ผู้ใช้งานเลือกแฟ้มข้อมูลที่ต้องการส่งให้กับระบบพัฒนา จากนั้นโปรแกรมควบคุมจะแสดง หน้าต่าง



หลังจากที่เสร็จสิ้นการส่งโปรแกรมภาษาเครื่องให้กับระบบพัฒนา และจะรอให้ผู้ใช้งานกดแป้นพิมพ์ใด เพื่อกลับเข้าสู่เมนูหลักกต่อไป

4. RUN EMULATOR

เป็นคำสั่งที่ผู้ใช้งานต้องการให้ระบบพัฒนาไมโครคอนโทรลเลอร์ INTEL MCS-51 ทำงานในรูปแบบใดรูปแบบหนึ่งดังนี้

4.1 ต้องการให้ระบบพัฒนา เริ่มทำงานจากจุดเริ่มต้น ของโปรแกรม (ORIGIN) หลังจากที่ได้ส่งโปรแกรมภาษาเครื่องให้กับระบบพัฒนาแล้ว (โดยการเลือกใช้คำสั่ง 2.1-2.3)

4.2 ต้องการให้ระบบพัฒนากลับเข้าสู่จุดเริ่มต้นของโปรแกรมการทำงานใหม่หลังจากที่มีการสิ้นสุดการทำงานแล้ว

4.3 ต้องการให้ระบบพัฒนากลับเข้าสู่จุดเริ่มต้นของการทำงานโดยปราศจากเงื่อนไขใด ๆ

ข้อสังเกต การเรียกใช้คำสั่งนี้จะมีผลทำให้ตัวแปรต่าง ๆ ที่เปลี่ยนแปลงไปตามโปรแกรมการทำงานของระบบพัฒนา ถูกเปลี่ยนแปลงค่าเป็นค่าเริ่มต้นใหม่

5. READ MEM

ในขณะพัฒนาโปรแกรม ผู้ใช้งานอาจจำเป็นต้องเปลี่ยนค่าตัวแปรใด ๆ หรือเปลี่ยนแปลงคำสั่งในบางส่วนของโปรแกรม ที่เป็นตัวแปรในส่วนของโปรแกรมควบคุมการทำงาน (โปรแกรมภาษาแอสเซมบลีของ INTEL MCS-51) ซึ่งมีทางเลือกในการเปลี่ยนแปลงค่าหรือคำสั่งดังกล่าว คือ

5.1 กลับเข้าสู่ EDITOR โดยเรียกแฟ้มข้อมูลที่ต้องการเปลี่ยนแปลง และเปลี่ยนแปลงแก้ไขโปรแกรมในบางส่วนที่ต้องการ หลังจากนั้นจึงทำการแปลงเป็นภาษาเครื่องและส่งเข้าสู่ระบบพัฒนาใหม่โดยอาจเลือกใช้ AUTO ASSEMBLE และ AUTO LINK

5.2 เรียกใช้คำสั่ง READ MEM โดยหลังจากที่ผู้ใช้งาน เลือกใช้คำสั่งนี้ โปรแกรมควบคุมจะแสดงหน้าต่าง

GET PAGE OF MEMORY
PAGE 0 (OOOOO-OFFFF)
PAGE 1 (10000-1FFFF)
PAGE 2 (20000-2FFFF)
PAGE 3 (30000-2FFFF)

ผู้ใช้งานสามารถเลือกหน้าของหน่วยความจำ (MEMORY PAGE) ของโปรแกรมที่ต้องการแก้ไขโดยการเลื่อนแถบเมนูขึ้น-ลงตามต้องการ (ตำแหน่งของตัวแปรหรือคำสั่งที่ต้องการเปลี่ยนแปลง สามารถดูได้จากแฟ้มข้อมูลรายละเอียด

(LISTING PROGRAM) ที่อาจจำเป็นต้องพิมพ์ออกมาโดยหลังจากที่ผู้ใช้งานได้เลือกหน้าของหน่วยความจำแล้ว โปรแกรมควบคุมจะแสดงหน้าต่างดังนี้

8031 EMULATOR MEMORY WINDOW	PAGE NUMBER:PAGE

	00	01	02	..	0F	ASCLL CODE
00	02	01	34	00	00	รหัส ASCLL ที่สอดคล้อง
01	90	40	FF	00	00	กับเลขฐาน 16 ทางซ้าย
02	23	67	00	00	00	มือตามลำดับจากซ้าย
	34	AB	00	00	00	ไปขวา
0E	56	CD	00	00	00	
0F	FF	FF	00	00	00	

MEMORY UTILITY:PgUp,PgDn, ESC,F1-MEMEDIT

ซึ่งเป็นหน้าต่างแสดงหน่วยความจำในส่วนของโปรแกรมภาษาเครื่อง ในตำแหน่งที่สอดคล้องกับโปรแกรมรายละเอียดโดยที่คอลัมน์แรก (ทางซ้ายมือสุด) จะแสดงตำแหน่งข้อมูลไบท์ต่ำ(LOW BYTE ADDRESS) และแถวบนสุดจะแสดงตำแหน่งข้อมูลไบท์สูง (HIGH BYTE ADDRESS) รูปแบบจะคล้ายคลึงกับโปรแกรม DEBUG ของ DOS และโปรแกรม PCTOOLS ในส่วนแถวและคอลัมน์ที่เหลือก็เป็นข้อมูล (DATA) ที่สอดคล้องกับตำแหน่งข้อมูล (ADDRESS) ส่วนคอลัมน์ทางขวามือสุดนั้นเป็นรหัส ASCLL ที่สอดคล้องกับข้อมูลเลขฐาน 16 ในตาราง โดยเรียงจากซ้ายไปขวา

สำหรับในหน้าต่างหน่วยความจำนี้ จะมีโปรแกรมบริการที่คอยช่วยเหลือ (UTILTRY PROGRAM)ในการเปลี่ยนแปลงตำแหน่งตัวแปรหรือคำสั่ง 5 รูปแบบดังนี้

5.2.1 PgUp เพื่อต้องการให้แสดงหน่วยความจำในหน้าต่อไป โดยรูปแบบการแสดงจะได้ไม่เกิน 16 หน้า (256 ตำแหน่งข้อมูล ต่อ 1 หน้า) และจะกลับเข้าสู่หน้าแรกเมื่อเกินหน้าสุดท้าย

5.2.2 PgDn เพื่อต้องการแสดงข้อมูลหน่วยความจำ ในหน้าที่ผ่านมา โดยจะกลับเข้าสู่หน้าสุดท้าย เมื่อถูกเรียกใช้ขณะที่กำลังแสดงหน้าแรก

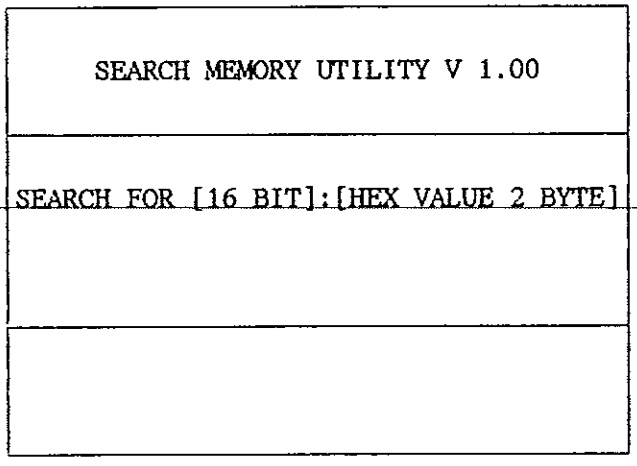
5.2.3 F1 เรียกโปรแกรมบริการ MEMORY EDITOR เมื่อผู้ใช้งานต้องการเปลี่ยนแปลงตัวแปรหรือคำสั่งที่ตำแหน่งใด ๆ ของหน่วยความจำ หลังจากที่ผู้ใช้งานเรียกโปรแกรมบริการนี้แล้วบนหน้าต่างแสดงหน่วยความจำ ของระบบพัฒนาจะแสดงแถบเมนูครอบคลุมตำแหน่งข้อมูลตำแหน่งแรกของหน้าที่มุมขวาบนของตาราง (ตำแหน่งหน่วยความจำ 0X00) และบรรทัดสุดท้ายของหน้าต่างแสดงหน่วยความจำจะเปลี่ยนข้อความเป็น

MOMORY EDITOR IS INSTALLED,ESC-EXIT,UP,DN,LF,RT

ผู้ใช้งานสามารถใช้การสั่งงานจากแป้นพิมพ์ ← หรือ → หรือ ↑ หรือ ↓ เพื่อเลื่อนแถบเมนูดังกล่าวไปยังตำแหน่งที่ต้องการ เมื่อถึงตำแหน่งที่ต้องการ

แล้วก็สามารถใส่ค่าตัวแปรหรือคำสั่งใหม่ ไปยังตำแหน่งนั้นได้เลย โดยลักษณะการรับข้อมูลจากแป้นพิมพ์ที่ตำแหน่งที่ต้องการเปลี่ยนแปลงจะรับค่าเฉพาะตัวเลข 0-9 และตัวอักษร A-F หรือรับค่าเฉพาะตัวเลขฐาน 16 เท่านั้น และจะรับค่าเพื่อนำไปใช้งานจริงเฉพาะ 2 ตัวสุดท้ายของตัวเลขที่ป้อนเข้าไปหลังสุดเท่านั้น ดังนั้นหากเกิดความผิดพลาดในการป้อนค่าตัวแปรหรือคำสั่ง ผู้ใช้งานก็สามารถป้อนค่าตัวเลขฐาน 16 วนไปเรื่อย ๆ จนกว่าจะได้ค่าที่ถูกต้อง หลังจากที่เปลี่ยนแปลงค่าได้ตามที่ต้องการแล้วและยอมรับค่าดังกล่าว ทำได้โดยการเลื่อนแถบเมนูไปยังตำแหน่งอื่นได้เลย หรือกด ESC เพื่อออกจาก MEMORY EDITOR

5.2.4 F2 เรียกโปรแกรมบริการ SEARCH MEMORY ซึ่งเป็นโปรแกรมบริการในการค้นหาข้อมูลที่คาดว่าจะมีอยู่ในหน่วยความจำ โดยผลที่ได้หลังจากที่การค้นหาสิ้นสุดคือ ตำแหน่งเริ่มต้นที่โปรแกรมบริการมีการตรวจพบข้อมูลที่มีรูปแบบเหมือนกับที่ผู้ใช้งานป้อนค่าและแสดงข้อความ"SUCESSFULLY SEARCH ADDRESS" ในกรณีที่ไม่พบชุดข้อมูลจะแสดงตำแหน่งหลังจากการค้นหาเป็น OFFF ขนาดความยาวของข้อมูลที่ต้องป้อนให้กับโปรแกรมบริการนี้จะต้องเป็น 2 ไบท์ ที่ต่อเนื่องกัน เช่น ต้องการค้นหาตำแหน่งที่มีข้อมูลเป็น 0200 รูปแบบของหน้าต่างโปรแกรมบริการเป็นดังนี้



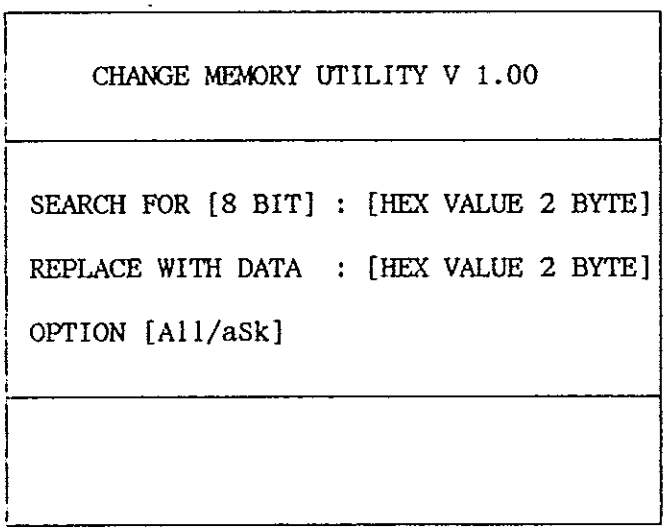
5.2.5 F3 เรียกโปรแกรมบริการ CHANGE MEMORY UTILITY เพื่อต้องการเปลี่ยนตัวแปรในหน่วยความจำส่วนโปรแกรมของระบบพัฒนา โดยที่

โปรแกรมบริการจะทำการตรวจสอบค่าตัวแปรหรือข้อมูลตั้งต้นที่ผู้ใช้งานป้อนค่าขนาด 8 บิท (1 ไบท์) และจะทำการแทนที่ด้วยตัวแปรหรือข้อมูลใหม่ซึ่งได้จากการป้อนค่าของผู้ใช้งานเช่นเดียวกัน โดยรูปแบบการใช้งานจะแบ่งย่อยเป็นสองรูปแบบตามความต้องการของผู้ใช้งาน คือ

- เปลี่ยนแปลงตัวแปร หรือข้อมูลในหน่วยความจำทุกตำแหน่งที่มีการค้นพบด้วยค่าใหม่ที่ผู้ใช้งานต้องการ

- เปลี่ยนแปลงเฉพาะตัวแปรหรือข้อมูล โดยให้โปรแกรมบริการสอบถามผู้ใช้งานก่อนที่จะทำการเปลี่ยนแปลง

รูปแบบของหน้าต่างโปรแกรมบริการ CHANGE MEMORY UTILITY เป็นดังนี้



หลังจากที่ผู้ใช้งานเลือกค่าข้อมูลที่ต้องการเปลี่ยนแปลง และค่าที่ต้องการให้เป็นแล้วการเลือกรูปแบบย่อยของการทำงาน ทำได้โดยการเลือกกด A หรือ S ซึ่งแทนรูปแบบการทำงานแบบเปลี่ยนทั้งหมด หรือเปลี่ยนโดยสอบถามผู้ใช้งานก่อน หากผู้ใช้งานเลือก A (เปลี่ยนแปลงทั้งหมดที่ค้นพบ) โปรแกรมบริการจะเปลี่ยนแปลงทุกค่าตัวแปรหรือข้อมูลที่ค้นพบ และรายงานผลการเปลี่ยนแปลงในรูปแบบจำนวนข้อมูลที่ได้ถูกเปลี่ยนแปลงไปแล้ว

หากเลือกกด S (เปลี่ยนแปลงโดยสอบถามก่อน) โปรแกรมบริการจะเข้าสู่หน้าต่างแสดงหน่วยความจำ และแสดงแถบเมนูในตำแหน่งที่มีการค้นพบข้อมูลหรือตัวแปร และจะแสดงข้อความที่บรรทัดล่างสุดของหน้าต่างเป็น

CHARACTER TO BE CHANGED IS SEARCHED. CHANGE [Y/N]?

กด Y หากต้องการเปลี่ยนแปลง หลังจากที่ผู้ใช้งานตอบรับ หรือต้องการข้ามตำแหน่งที่แสดงแถบเมนู โปรแกรมบริการจะหาตำแหน่งที่มีข้อมูลตรงกับที่ต้องการเปลี่ยนแปลงตำแหน่งต่อไปจนกว่าจะจบหน้าของหน่วยความจำ และเมื่อเสร็จสิ้นการเปลี่ยนแปลงข้อมูลโปรแกรมบริการจะรายงานผลจำนวนข้อมูลที่ถูกเปลี่ยนแปลง

6. READ XRAM

ในระบบเป้าหมาย (TARGET SYSTEM) ที่ใช้ไมโครคอนโทรลเลอร์ INTEL MCS-51 เป็นตัวจัดการมีการแบ่งหน่วยความจำเป็นสองประเภท คือ

6.1 หน่วยความจำส่วนที่เก็บค่าโปรแกรมการทำงาน (เป็นหน่วยความจำในระบบพัฒนาไมโครคอนโทรลเลอร์ INTEL MCS-51 ซึ่งโปรแกรมควบคุมสามารถส่งค่าและอ่านค่าได้โดยระบบเป้าหมายสามารถอ่านค่าได้อย่างเดียว) โดยปกติแล้วหน่วยความจำในส่วนนี้คือ EPROM ในระบบเป้าหมายนำไปใช้งานจริงการดัดแปลง แก้ไข อ่านค่า ถูกจัดการผ่านเมนู READ RAM ในหัวข้อที่ผ่านมา

6.2 หน่วยความจำภายนอก ทำหน้าที่เป็นตัวเก็บข้อมูลการทำงานของระบบเป้าหมายภายใต้การควบคุมโดยโปรแกรม หน่วยความจำในส่วนนี้ไมโครคอนโทรลเลอร์ของระบบเป้าหมายสามารถอ่านและเขียนได้ ดังนั้นจึงเป็นความสะดวกในการพัฒนาระบบถ้าสามารถอ่านความเป็นไปของข้อมูลในหน่วยความจำส่วนนี้ได้

โปรแกรมควบคุมระบบพัฒนาไมโครคอลโทรลเลอร์ INTEL MCS-51 ได้บริการอ่านค่าหน่วยความจำภายนอกระบบพัฒนา โดยผู้ใช้งานสามารถเลือกใช้

เมนูนี้ โดยมีเงื่อนไขว่าระบบเป้าหมายต้องมีวงจรจัดการสำหรับการติดต่อผ่านทางพอร์ทเชื่อมต่อแบบอนุกรม (SERIAL PORT) มาตรฐาน RS232 COM1 และจำเป็นต้องต่อสายสัญญาณระหว่างระบบเป้าหมายกับไมโครคอมพิวเตอร์ที่ควบคุมการทำงานก่อนที่เมนูนี้จะถูกเรียกใช้

ผู้ใช้งานจำเป็นต้องสร้างแฟ้มข้อมูลชื่อ MCS.CFG เพื่อกำหนดรหัสตำแหน่งหน่วยความจำภายนอกในส่วนของแอสเดรสสูง (HIGH ADDRESS) หรือหน้าของหน่วยความจำให้สอดคล้องกับการให้รหัส ในการติดต่อกับหน่วยความจำภายนอกในระบบเป้าหมาย สำหรับรูปแบบฟอร์แมทของ MCS.CFG เป็นดังนี้

HEADER:จำนวนหน้าของหน่วยความจำภายนอกระบบเป้าหมายที่ต้องการอ่านค่า\N
หน้าที่ 1\N
หน้าที่ 2\N
หน้าที่ 3\N
หน้าที่ N\N

ตัวอย่างเช่น ระบบเป้าหมายมีการถอดรหัส สำหรับส่วนที่เป็นหน่วยความจำภายนอกตั้งแต่ 0000H-7FFFH และผู้ใช้งานต้องการอ่านหน่วยความจำในหน้าที่ถูกกำหนดด้วยแอสเดรสส่วนบน 00H-04H สามารถเขียน MCS.CFG ได้เป็น

5

00

01

02

03

04

โปรแกรมจัดการ จะแสดงหน้าต่างของหน่วยความจำภายนอกในรูปแบบและวิธีการอ่านค่าเหมือนกับหน้าต่างของหน่วยความจำระบบที่กล่าวมาแล้ว ในหัวข้อ 5 READ MEM โดยจะมีชื่อหน้าต่างเป็น

EXTERNAL MEMORY WINDOW UTILILT 1.00 PAGE NUMBER: PAGE

และเนื่องจากหน่วยความจำในส่วนนี้เป็นของระบบเป้าหมายโดยตรงการเปลี่ยนแปลงแก้ไขทำได้โดยยาก จึงไม่มีโปรแกรมบริการต่าง ๆ

ในกรณีที่ระบบการติดต่อทางพอร์ทเชื่อมต่อแบบอนุกรมมีปัญหาใด ๆ จะมีหน้าต่างรายงานปัญหาแสดงข้อความเป็น

EMU31 REPORT
ERROR 1:SERIAL PORT COM1 ERROR
SEND ERROR DETECTED IN SERIAL PORT

และโปรแกรมควบคุมอาจค้างการทำงาน (HANG) ให้ผู้ใช้งานกด CTRL-BREAK และตรวจสอบการสื่อสารข้อมูลก่อนที่จะเรียกโปรแกรมใช้งานอีกครั้ง

7. DEBUGGER

เป็นตัวเลือกของโปรแกรมควบคุม ที่ทำหน้าที่ในการให้ข้อมูลเกี่ยวกับการเปลี่ยนแปลงของส่วนต่าง ๆ ของไมโครคอลโทรลเลอร์ เช่น รีจิสเตอร์ (REGISTER) หน่วยความจำภายใน(INTERNAL RAM) ของไมโครคอลโทรลเลอร์ของระบบเป้าหมาย INTEL MCS-51 ซึ่งเป็นข้อมูลที่เป็นประโยชน์ในการพัฒนา

ระบบ เนื่องจากในออกแบบระบบควบคุมโดยการใช้ไมโครคอนโทรลเลอร์ หากต้องการตรวจสอบการทำงานของโปรแกรมควบคุมการทำงานที่พัฒนาขึ้นอาจทำได้ใน 2 รูปแบบ คือ

7.1 ใช้โปรแกรม SIMULATOR ซึ่งเป็นซอร์ฟแวร์ที่พัฒนาขึ้นเพื่อทำงานบนไมโครคอมพิวเตอร์ในการเลียนแบบการทำงานของไมโครโปรเซสเซอร์ หรือไมโครคอนโทรลเลอร์แต่ละตัวโดยผู้ใช้งานสามารถสั่งงานให้โปรแกรมทำงานแบบทีละคำสั่ง (SINGLE STEP) ทำงานแบบต่อเนื่อง หรือทำงานจนกว่าจะพบจุดจบที่กำหนดขึ้น (BREAK POINT) โดยโปรแกรมจะรายงาน ค่าความเปลี่ยนแปลงของรีจิสเตอร์ หน่วยความจำภายใน (หรือภายนอกขึ้นอยู่กับขีดความสามารถของโปรแกรม) เมื่อไมโครคอนโทรลเลอร์ทำงานถึงคำสั่งใด ๆ การตรวจสอบการทำงานแบบนี้ไม่จำเป็นต้องสร้างวงจรควบคุมและมีข้อเสียคือไม่สามารถคาดหวังการทำงานของระบบจริงที่มีฮาร์ดแวร์เป็นองค์ประกอบได้

7.2 ตรวจสอบการทำงานของระบบจริง โดยใช้การสั่งงานให้ระบบควบคุมทำงานในรูปแบบที่ต้องการและรายงานผลความเปลี่ยนแปลงของตัวแปร การทดสอบแบบนี้ต้องให้เวลาในการสร้างวงจรควบคุม(ระบบเป้าหมาย)ที่มีองค์ประกอบของฮาร์ดแวร์ครบทุกอย่างตามที่ต้องการนำไปใช้งานจริง

DEBUGGER ที่ผู้วิจัยได้ออกแบบและสร้างขึ้นนี้เป็นการทดสอบในรูปแบบที่ 7.2 คือ ผู้ใช้งานต้องต่อระบบเป้าหมายที่ใช้งานจริงเชื่อมโยงกับระบบพัฒนาไมโครคอนโทรลเลอร์ มีความสามารถดังนี้

- รายงานผลความเปลี่ยนแปลงของตัวแปรระดับรีจิสเตอร์
- แสดงคำสั่ง ที่อยู่ในรูปแบบที่แสดงอยู่ใน แฟ้มข้อมูลรายละเอียด (LISTING PROGRAM) ของโปรแกรมที่กำลังพัฒนาอยู่ในขณะนั้น
- รายงานผลความเปลี่ยนแปลง ของหน่วยความจำภายในไมโครคอนโทรลเลอร์
- ตอบสนองต่อการทำงานทีละคำสั่ง การทำงานแบบต่อเนื่องจนจบโปรแกรมและการทำงานแบบต่อเนื่องจนกระทั่งถึงจุดจบที่กำหนดขึ้น(BREAK POINT)

การรายงานผลของ DEBUGGER จะอยู่ในหน้าต่างดังนี้

8031 IN-CIRCUIT DEBUGGER VERSION 1.1		
PC	P0 P1 P2 P3 IP IE	
ACC	ค่าของตัวแปรหรือรีจิสเตอร์	
B	TM TC TH TL Th Tl	
PSW	ที่เปลี่ยนแปลงตามโปรแกรม	แสดงคำสั่งขณะนั้นและก่อนหน้า
SP	SC SB PCON SUMB	4 คำสั่ง
DPTR		0000 00 NOP

00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F	ASCII VALUE
00	
01	
02	
03	ค่าของหน่วยความจำภายในไมโครคอลโทรลเลอร์
04	INTEL MCS-51 ในฐาน 16 และระหัส ASCII ที่สอดคล้อง
05	
06	
07	

COMMAND : F1 : RUN TILL KEY F2 : SINGLE STEP F4 : RESET ESCape

โดยมีคำสั่งย่อยที่สามารถเรียกใช้ ดังนี้

7.2.1 RUN TILL KEY เป็นคำสั่งที่ทำให้ระบบเป้าหมายทำงานไปจนกว่าผู้ใช้งานจะกดแป้นพิมพ์ใด ๆ

7.2.2 SINGLE STEP เป็นการสั่งงานให้ระบบเป้าหมายทำงานทีละคำสั่งและหยุดเมื่อจบการทำงานคำสั่งนั้น ๆ

7.2.3 RESET เพื่อทำให้ระบบเป้าหมายรีเซ็ตและกลับสู่จุดเริ่มต้นของโปรแกรมควบคุม

8. CLEAR MEM

เป็นตัวเลือกของโปรแกรมควบคุม ที่ทำให้หน่วยความจำส่วนโปรแกรมในระบบเป้าหมายทุกตำแหน่ง(ADDESSS) เป็น 00 ซึ่งเป็นคำสั่ง NOP (NO OPERATION) สำหรับไมโครคอนโทรลเลอร์ตระกูล INTEL MCS-51

9. EMU TEST

เป็นตัวเลือกเพื่อทดสอบการทำงานของระบบพัฒนาไมโครคอนโทรลเลอร์ INTEL MCS-51 โดยจะรายงานผลตำแหน่งการถอดรหัสของระบบพัฒนาบนไมโครคอมพิวเตอร์ขนาด และตำแหน่งของหน่วยความจำส่วนโปรแกรมที่มีความผิดพลาดในการเขียนหรืออ่าน และรายงานผลการตรวจสอบการเชื่อมต่อพอร์ทอนุกรมระหว่างไมโครคอมพิวเตอร์ที่ควบคุมการทำงาน กับระบบเป้าหมาย

10. DOS SHELL

เป็นตัวเลือกเพื่อออกจากโปรแกรมควบคุมการทำงาน ของระบบพัฒนาไมโครคอนโทรลเลอร์ INTEL MCS-51 ชั่วคราว และพร้อมที่จะกลับเข้าสู่การทำงานใหม่ เมื่อผู้ใช้งานพิมพ์ EXIT ในระบบปฏิบัติการ

11. EXIT

เป็นตัวเลือก เมื่อผู้ใช้งานต้องการออกจากโปรแกรมควบคุมการทำงานไมโครคอนโทรลเลอร์ INTEL MCS-51 และกลับเข้าสู่ระบบปฏิบัติการ

3. ขั้นตอนการพัฒนาโปรแกรมควบคุมการทำงานของระบบไมโครคอนโทรลเลอร์

ในการออกแบบระบบควบคุมโดยใช้ไมโครคอนโทรลเลอร์นั้น ตามประสบการณ์ของผู้วิจัยนั้นพอจะสรุปเป็นขั้นตอนได้ดังนี้

1. วิเคราะห์ระบบควบคุมที่จะทำการออกแบบสร้างขึ้นว่า มีความจำเป็นหรือไม่ที่จะต้องใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมการทำงาน เนื่องจากโดยปกติแล้วระบบควบคุมโดยการใช้ไมโครคอนโทรลเลอร์ มีความซับซ้อนมากกว่าระบบควบคุมแบบอนาลอก (ANALOG CONTROL SYSTEM) แต่มีความซับซ้อนน้อยกว่าการควบคุมแบบดิจิตอลโดยใช้วงจรแบบแยกส่วน แต่ระบบควบคุมโดยใช้ไมโครคอนโทรลเลอร์สามารถเพิ่มเติมส่วนขยายได้ง่ายกว่า รวมทั้งสามารถออกแบบให้มีความซับซ้อนทางการควบคุมได้มาก เช่น

เปรียบเทียบการออกแบบระบบควบคุมการเปิดปิดอัตโนมัติ ตามอุณหภูมิที่กำหนดซึ่งสามารถใช้การควบคุมแบบอนาลอกง่ายกว่า โดยใช้วงจรรวมประเภทออปแอมป์(OP-AMP:OPERATIONAL AMPLIFIER) ทำหน้าที่เป็นวงจรเปรียบเทียบ (COMPARATOR) (หรืออาจเลือกใช้การควบคุมแบบอื่น ซึ่งสามารถควบคุมได้ดีกว่า เช่น แบบ PID:PROPORTIONAL INTEGRAL AND DERIVATIVE CONTROL) อุณหภูมิที่ตั้งค่าไว้และเปรียบเทียบกับค่าของอุณหภูมิที่วัดในขณะนั้น

เมื่อเรานำระบบควบคุมดังกล่าวมาสร้างโดยการควบคุมของไมโครคอนโทรลเลอร์ ในส่วนหลักของระบบควบคุมนอกเหนือจากระบบไมโครคอนโทรลเลอร์แล้ว สิ่งที่จำเป็นต้องเพิ่มเติมมีเพียงวงจรแปลงสัญญาณจากรูปแบบอนาลอกของอุณหภูมิเป็นแบบดิจิตอล สำหรับการควบคุมนั้นสามารถเลือกใช้โปรแกรมควบคุมเป็นแบบ PID (ในทางซอร์ฟแวร์) ได้เลยนอกจากนี้ยังสามารถเพิ่มเติมส่วนขยายอื่น ๆ เช่น ส่วนรับข้อมูลอุณหภูมิที่กำหนด ช่วงกว้างของความแตกต่างของอุณหภูมิที่ยอมรับได้ (DELTA T:$\triangle$ T)ระบบการตั้งเวลาเพื่อเปิดปิดระบบแสดงผล

ความสำคัญของขั้นตอนนี้ บางครั้งอาจทำให้ไม่จำเป็นต้องออกแบบระบบลวบคุมต่อไปเลย เนื่องจากอาจแก้ปัญหาโดยการออกแบบวงจรอื่นแทนได้

2. วิเคราะห์รายละเอียด ของปัญหาที่ต้องการใช้ระบบควบคุมไปช่วยแก้ไข ว่าจำเป็นต้องมีส่วนอื่นเพิ่มเติมหรือไม่ มีอะไรบ้าง และวางระบบของโปรแกรมควบคุมการทำงาน

3. เลือกรูปแบบการควบคุม การถอดรหัสอ้างอิงอุปกรณ์ส่วนขยาย รูปแบบของการรับหรือส่งข้อมูล รูปแบบของระบบโปรแกรมควบคุมการทำงาน

4. ออกแบบวงจร และสร้างต้นแบบระบบควบคุมที่มีความสมบูรณ์ของอุปกรณ์ส่วนขยาย ที่คาดว่าจะใช้งานได้

5. พัฒนาส่วนของโปรแกรมควบคุมการทำงาน และทดสอบการทำงานกับระบบจริงในบางครั้งโปรแกรมควบคุมการทำงานอาจจำเป็นต้องทดสอบอัลกอริทึม (ALGORETHM) โดยใช้ภาษาสูง เช่น ภาษา C และเมื่อมีความถูกต้องก็เปลี่ยนจากอัลกอริทึมเป็นภาษาแอสเซมบลี เพื่อทดสอบกับระบบจริงอีกครั้งหากขั้นตอนนี้มีความผิดพลาด หรือต้องการเพิ่มเติมในรายละเอียดก็สามารถย้อนกลับไปในขั้นตอนที่ 4 เพื่อเพิ่มเติมส่วนของต้นแบบระบบควบคุม

6. พัฒนาส่วนเชื่อมต่อและโปรแกรมจัดการของการติดต่อ ของระบบควบคุมที่ต้องการกับระบบควบคุมอื่น ๆ (ถ้ามี)

7. ทดสอบการทำงานรวมของระบบทั้งหมดหาข้อผิดพลาดและแก้ไข หาจุดจำกัดในการทำงานและแก้ไข

4. ตัวอย่างการสร้างระบบแสดงผลแบบอิสระโดยใช้ไมโครคอนโทรลเลอร์

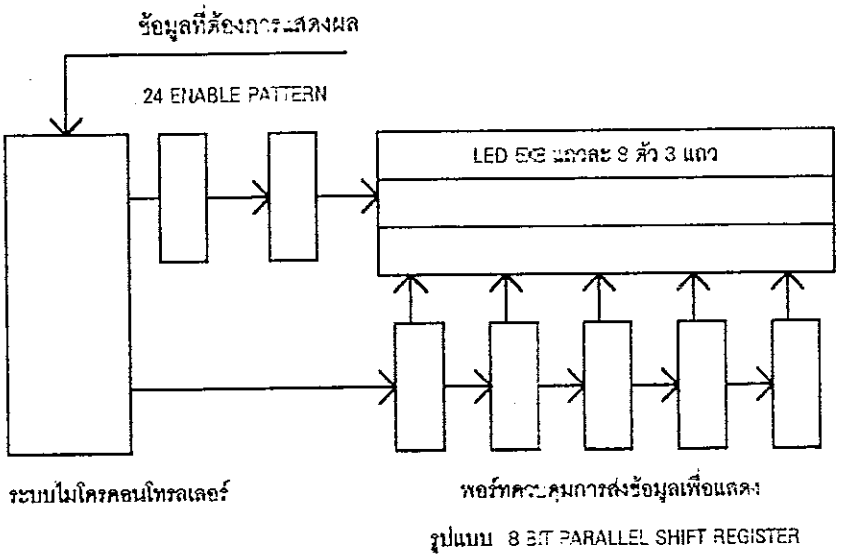
ปัญหาเริ่มต้นต้องการสร้างระบบแสดงผลแบบอิสระที่สามารถนำไปเชื่อม-โยงกับระบบควบคุมใด ๆ ผ่านทางพอร์ทแบบขนาน โดยมีความสามารถในการแสดงผลได้มากกว่า 1 บรรทัด บรรทัดละ 8 ตัวอักษรสามารถแสดงผลในรูปแบบปกติและรูปแบบ REVERSE FONT

วิเคราะห์ปัญหา ในทางการค้าได้มีการพัฒนาระบบแสดงผลแบบอิสระโดยใช้หน่วยแสดงผลเป็นแบบ LCD (LIQUID CRYSTAL DISPLAY) สามารถแสดงผลแบบกราฟิกส์ (GRAPHIC) ได้ ทำให้สามารถแสดงตัวอักษรทั้งสองรูปแบบได้ตามที่ต้องการแต่ราคาค่อนข้างแพง ดังนั้นจึงจำเป็นต้องออกแบบและสร้างโดยใช้ไมโครคอนโทรลเลอร์

รายละเอียดที่ต้องเพิ่มเติมเลือกใช้หน่วยแสดงผลเป็นแบบ LED (LIGHT EMITTING DIODE) แบบ DOT-MATRIX บรรทัดละ 8 ตัว ให้แสดงได้มากถึง 3 บรรทัด ดังนั้นจำเป็นต้องมีพอร์ทของระบบควบคุมสำหรับส่วนส่งข้อมูลแสดงแบบขนาด 8 บิท จำนวน 5 ตัว และพอร์ทควบคุมการส่งสัญญาณ ENABLE แบบขนาด 8 บิท จำนวน 3 ตัว เลือกรูปแบบของตัวอักษรที่แสดงเป็นแบบ 5X7

เลือกรูปแบบการควบคุม เลือกการส่งข้อมูลให้กับระบบควบคุมสำหรับส่วนส่งข้อมูลแสดงแบบขนาด 8 บิท เป็นแบบขนาด 8 บิทชิฟรีจิสเตอร์ (8 BIT PARALLEL SHIFT REGISTER รายละเอียดวงจรอยู่ในบทที่ 3) ทำให้ถอดรหัสพอร์ทเพียง 1 พอร์ท การส่งข้อมูลที่ต้องการแสดงผลให้กับระบบแสดงผลแบบอิสระนี้ให้กระทำผ่านพอร์ทเชื่อมโยงแบบขนาน ภายใต้รหัสควบคุมการส่งข้อมูลในรูปแบบของเฟรมแบบมาตรฐาน

ออกแบบวงจรต้นแบบ รูปแบบร่างของวงจรที่ได้เป็นดังนี้ (วงจรจริงสามารถดูได้จากบทที่ 3)



พัฒนาส่วนของโปรแกรมควบคุมการทำงาน โดยการเลือกใช้การส่งข้อมูลที่ต้องการแสดงส่งออกสู่พอร์ทแสดงผลและส่งสัญญาณควบคุมการติดของ LIED ในแถวนั้น ก็สามารถเขียนโปรแกรมในส่วน TIMER INTERRUPT เพื่อทำการ SCAN LED ในแต่ละแถว ด้วยเวลาที่เท่า ๆ กัน และในส่วนการแสดงผลก็สามารถเขียน

โปรแกรมจัดการเพื่อสร้างรูปแบบการแสดงผลได้มากเท่าที่ต้องการ สำหรับในส่วนการรับข้อมูลที่ต้องการแสดงผลผ่านทางพอร์ทเชื่อมต่อมาตรฐานแบบขนาด ในทางโปรแกรมเลือกใช้การรับข้อมูลเป็นแบบวงแหวน (CIRCULAR OR RING BUFFER) ดังนั้นจำเป็นต้องเขียนโปรแกรมตอบสนองในส่วนนี้ หลังจากที่มีการเพิ่มเติมในส่วนใดของโปรแกรม ก็สามารถทดสอบการทำงานโดยใช้ระบบพัฒนาไมโครคอนโทรล-เลอร์ได้

ทดสอบการเชื่อมโยงกับระบบควบคุมที่ต้องการ

หมายเหตุ การออกแบบระบบแสดงผลแบบอิสระที่เป็นตัวอย่างนี้ ผู้วิจัยได้สร้างเป็นที่เรียบร้อยแล้วละเอียดของการใช้งานอยู่ในภาคผนวก ข

ภาคผนวก ข

ระบบแสดงผลแบบแยกอิสระ

เพื่อให้นักวิจัยท่านอื่น ๆ มีความสะดวกในการพัฒนาระบบจึงได้พยายามออกแบบระบบต่าง ๆ ที่ใช้ในงานวิทยานิพนธ์นี้ให้เป็นระบบอิสระ ซึ่งประกอบด้วยไมโครโปรเซสเซอร์ 8031 เป็นตัวจัดการ และสามารถเชื่อมต่อสั่งงานจากระบบควบคุมผ่านพอร์ทเชื่อมต่อมาตรฐานแบบขนาน (PARALLEL PORT) ระบบหนึ่งที่ได้ออกแบบและสร้างขึ้นมาคือ ระบบแสดงผลแบบแยกอิสระ (INTELLIGENT DISPLAY UNIT) จากการที่ผู้วิจัยพยายามทำให้การสั่งงานระบบเก็บข้อมูลระยะไกล ซึ่งเป็นงานวิทยานิพนธ์ทำได้โดยง่าย จึงได้มีแนวคิดที่จะให้การสั่งงานผ่านการเลือกเมนู (MENU) ดังนั้น จึงได้เลือกออกแบบระบบแสดงผลเป็นแบบดอทแมทริกซ์ (DOT MATRIX) ให้สามารถแสดงผลได้ 3 บรรทัด 8 ตัวอักษรต่อบรรทัด ซึ่งเพียงพอต่อการออกแบบระบบเมนู รูปแบบการทำงานมีลักษณะเป็นแบบ INTELLIGENT DISPLAY UNIT คือ ผู้ใช้งานไม่มีความจำเป็นในการเขียนโปรแกรมควบคุมการแสดงผลขึ้นมาเอง และไม่จำเป็นต้องออกแบบฮาร์ดแวร์ (HARDWARE) เพิ่มเติมการสั่งงานและส่งข้อมูลเพื่อแสดงผล สามารถกระทำโดยผ่านทางพอร์ทเชื่อมต่อแบบขนาน (PARALLEL PORT)

1. รูปแบบการแสดงผลของระบบแสดงผลแบบอิสระ

ผู้วิจัยได้สร้างรูปแบบการแสดงผลให้กับระบบแสดงผลนี้ 7 รูปแบบด้วยกัน ดังนี้

1. การแสดงผลในรูปแบบปกติ (รหัสควบคุมการแสดงผล 10H) เป็นรูปแบบการแสดงผลที่ผู้ใช้งานต้องการให้ข้อความที่ส่งให้กับระบบแสดงผลแสดงออกมา ตามหมายเลขบรรทัดที่กำหนดโดยปราศจากการกระพริบหรือการเลื่อนตัวอักษร

2. การแสดงผลแบบตัวอักษรเลื่อน (รหัสควบคุมการแสดงผล 11H) ตัวอักษรที่ส่งออกมาจะแสดงในรูปแบบตัวอักษรเลื่อน โดยเริ่มจากขอบด้านขวาของ

บรรทัดที่กำหนด ไม่จำกัดความยาวของข้อความเป็นรูปแบบการแสดงผลที่สร้างขึ้นเพื่อใช้ในกรณีที่ต้องการบรรยายรายละเอียด

3. การแสดงผลแบบกระพริบข้อความ (รหัสควบคุมการแสดงผล 12H) ใช้ในกรณีที่ต้องการดึงดูดความสนใจของการแสดงข้อความเช่น กรณีที่มีความผิดพลาดของระบบเกิดขึ้นโดยค่าของการกระพริบจะถูกกำหนดเป็น 5 ครั้งต่อ 1 บรรทัด

4. การแสดงผลแบบเลื่อนตัวอักษรขึ้นจากขอบล่างของบรรทัด (รหัสควบคุมการแสดงผล 13H)

5. การแสดงผลทุกบรรทัดพร้อมกัน (รหัสควบคุมการแสดงผล 14) เพื่อต้องการแสดงข้อความที่มีความเปลี่ยนแปลงทุกบรรทัดพร้อมกัน

6. การเปลี่ยนรูปแบบตัวอักษรเป็นแบบ REVERSE ทั้งบรรทัดที่กำหนด (รหัสควบคุมการแสดงผล 15H)

7. การเปลี่ยนรูปแบบตัวอักษรเป็นแบบ REVERSE เฉพาะตำแหน่งตัวอักษรและจำนวนตัวอักษรที่ต้องการเปลี่ยนแปลงตามที่กำหนด (รหัสควบคุมการแสดงผล 16H)

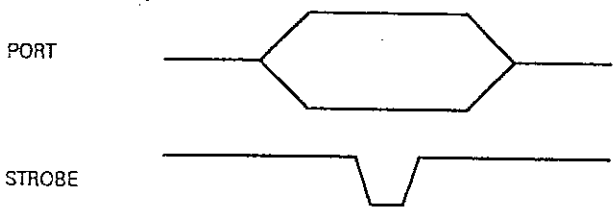
สำหรับรูปแบบของการแสดงผล 6 และ 7(รหัสควบคุมการแสดงผล 15H และ 16H) นั้นเป็นการเพิ่มเติม เพื่อให้สามารถสร้างระบบเมนู (MENU) ได้โดยง่ายดาย

2. การเชื่อมต่อระบบควบคุมโดยทั่วไปกับระบบแสดงผล

ดังที่ได้กล่าวไว้ตอนต้นแล้วว่า รูปแบบการติดต่อระหว่างระบบที่ต้องการพัฒนากับระบบแสดงผล(INTELLIGENT DISPLAY UNIT) สามารถทำได้โดยการติดต่อผ่านทางพอร์ทมาตรฐานแบบขนาน (PARALLEL PORT) โดยมีกลุ่มของสัญญาณที่จำเป็นคือ

- พอร์ทข้อมูลขนาด 8 บิท (8 BIT OUTPUT PORT)
- สายสัญญาณตอบรับ(ACKNOWLEDGE)ทำงานที่ระดับลอจิกต่ำ(ACTIVE LOW)
- สายสัญญาณควบคุมการส่งข้อมูล (STROBE) ทำงานที่ระดับลอจิกต่ำ

สำหรับระบบแสดงผลนี้ไม่มีการส่งสัญญาณตอบรับกลับสู่ระบบควบคุม ซึ่งเป็นรูปแบบการควบคุมแบบเปิด โดยที่ซอร์ฟแวร์ของระบบแสดงผลจะทำหน้าที่ในการเก็บข้อมูลทุกค่าที่ได้รับเข้าสู่บัฟเฟอร์ซึ่งอยู่ในรูปแบบของบัฟเฟอร์วงกลม(CIRCULAR BUFFER หรือ RING BUFFER) โดยหลักการอินเทอร์รัพท์ (INTERRUPT) จึงสามารถรับข้อมูลที่ส่งมาได้ทุกค่าอย่างแน่นอน สำหรับสัญญาณที่ควบคุมการส่งข้อมูล (STROBE) จำเป็นต้องมีเงื่อนไขควบคุมสภาวะของสัญญาณ คือ สัญญาณนี้จะต้องมีระดับลอจิกต่ำ (LOGIC 0) ค้างไว้ไม่น้อยกว่า 10 มิลลิวินาที หลังจากที่มีการส่งข้อมูลออกทางพอร์ทข้อมูลแล้วเพื่อให้ระบบแสดงผลสามารถตอบรับสัญญาณได้ รูปแบบของการติดต่อระหว่างระบบทั้งสองจะเป็นไปตามแผนผังเวลาดังนี้



เวลาที่สัญญาณควบคุมการส่งข้อมูล (STROBE) ค้างอยู่ที่ระดับลอจิกต่ำไม่น้อยกว่า 10 มิลลิวินาที

3. รูปแบบการส่งข้อมูลเพื่อแสดงผล

การส่งข้อมูลให้กับระบบแสดงผลจะอยู่ในรูปของเฟรม (FRAME) ของข้อมูลตามรูปแบบมาตราฐานของการส่งข้อมูลในรูปแบบนี้โดยทั่วไป คือ ประกอบด้วยรหัสเริ่มต้นของเฟรม (STX) และปิดท้ายด้วยรหัสสิ้นสุด (ETX) ของเฟรมระหว่างรหัสทั้งสองในเฟรมข้อมูลจะเป็นข้อมูลที่ต้องการส่งสำหรับการติดต่อกับระบบแสดงผล ไว้วางเฟรมข้อมูลมาตรฐานในรูปแบบดังนี้

STX	หมายเลขบรรทัด LINE NUMBER	รหัสควบคุม การแสดงผล DISPLAY TYPE	ข้อความที่ต้องการแสดงผล ASCII CODE OF EXPRESSION TO DISPLAY	ETX

โดยที่ รหัสเริ่มต้นของเฟรม STX:START OF TEXT = 02H

และ รหัสสิ้นสุดของเฟรม ETX:END OF TEXT = 03H

สำหรับหมายเลขบรรทัดที่ต้องการแสดงข้อความดังที่ได้กล่าวไปในตอนต้นแล้วว่าระบบแสดงผลนี้ สามารถแสดงได้มากที่สุด 3 บรรทัด 8 ตัวอักษรต่อบรรทัด โดยที่รหัสควบคุมบรรทัดนั้นสามารถเลือกได้ตามตาราง

ตารางแสดงรหัสที่ส่งให้กับระบบแสดงผลเปรียบเทียบกับหมายเลขบรรทัด

ตารางที่ 1 แสดงรหัสควบคุมและรูปแบบการแสดงผลที่สอดคล้องกับรหัสควบคุม

รหัส (CODE)	หมายเลขบรรทัด (LINE NUMBMER)
10H	1
11H	2
12H	3

ตารางที่ 2 แสดงรูปแบบของรหัสสั่งงานและรูปแบบการแสดงผล

รหัส (CODE)	รูปแบบการแสดงผล (DISPLAY TYPE)
10H	แสดงผลข้อความแบบธรรมดา
11H	แสดงผลแบบตัวอักษรเลื่อน โดยไม่จำกัดความยาวของข้อความที่ต้องการแสดง
12H	แสดงข้อความที่ต้องการแบบกระพริบ
13H	แสดงข้อความแบบเลื่อนข้อความในบรรทัดนั้น ๆ จากขอบล่างของบรรทัด
14H	แสดงข้อความทุกบรรทัดพร้อมกัน
15H	เปลี่ยนรูปแบบตัวอักษรของบรรทัดที่ต้องการ
16H	เปลี่ยนรูปแบบตัวอักษรเฉพาะตัวอักษรใด ๆ หรือจากตัวอักษรที่กำหนด+ความยาวของบริเวณที่ต้องการเปลี่ยนแปลงรูปแบบตัวอักษร

สำหรับรูปแบบการแสดงผลตามรหัสควบคุมการแสดงผล 15H และ 16H จะมีความแตกต่างจากกรณีรหัสควบคุมการแสดงผลอื่น ๆ คือ

STX (02H)	หมายเลขบรรทัด (LINE NUMBER)	ต้องการเปลี่ยนรูปแบบตัวอักษรทั้งบรรทัดส่งรหัสควบคุมการแสดงผล 15H	ETX (03H)

STX(02H)	หมายเลขบรรทัด (LINE NUMBER)	รหัสควบคุมการแสดงผล 16H	ตัวอักษรเริ่มต้น	จำนวนตัวอักษรที่ต้องการเปลี่ยน	ETX (03H)

ตารางตัวอักษรที่สอดคล้องกับบรรทัดตามที่แสดงจริง

00	01	02	03	04	05	06	07	บรรทัดที่ 1 (10H)
00	01	02	03	04	05	06	07	บรรทัดที่ 2 (11H)
00	01	02	03	04	05	06	07	บรรทัดที่ 3 (12H)

4. ตัวอย่างการส่งข้อความมาให้กับระบบแสดงผลเพื่อแสดงผลในรูปแบบต่าง ๆ

1. ต้องการส่งข้อความ "ABCDEFGH" เพื่อแสดงในบรรทัดที่ 1 ตามรูปแบบการแสดงผลแบบธรรมดา โดยที่ข้อความที่ต้องการแสดงนั้นมีความยาว 8 ตัวอักษร ซึ่งสามารถแสดงได้ครบใน 1 บรรทัด

02H	10H	10H	ABCDEFGH	03H

2. ต้องการแสดงข้อความบรรยายรูปแบบการทำงานของ F1 ในบรรทัดที่ 2 โดยที่ประโยคที่ต้องการแสดงคือ " F1 IS A HELP MENU. ACTIVATE WHEN USER WANT TO KNOW THE DESCRIPTION OF ANY FUNCTION KEY." ซึ่งเป็นประโยคที่มีความยาวมาก จำเป็นต้องแสดงผลโดย

ใช้รูปแบบ 11H โดยใช้รูปแบบการส่งข้อมูลดังนี้

02H	11H	11H	F1 IS HELP MENU, ACTIVATE WHEN USER WANT TO KNOW THE DESCRIPTION OF ANY FUNCTION KEY.	03H

3. ต้องการเปลี่ยนรูปแบบของตัวอักษรที่บรรทัดหมายเลข 2 เป็น REVERSE

02H	11H	15H	03H

4. ต้องการเปลี่ยนรูปแบบตัวอักษรจากปกติเป็น REVERSE จำนวน 2 ตัว อักษรเริ่มจากตัวอักษรที่ 1 ในบรรทัดหมายเลข 1

02H	11H หมายเลขบรรทัด	16H รูปแบบการแสดงผลต้องการเปลี่ยนรูปแบบตัวอักษร	00H ตัวอักษรตัวแรกที่ต้องการเปลี่ยนรูปแบบ	02H จำนวนตัวอักษรที่ต้องการเปลี่ยนรูปแบบนับรวมตัวอักษรเริ่ม	03H

5. ตัวอย่างโปรแกรมภาษาแอสเซมบลี MCS-51 สำหรับการส่งข้อมูลให้ระบบแสดงผล

```
        ;*****************************************************
        ;ตัวอย่างโปรแกรมเพื่อส่งข้อมูลที่ถูกกำหนดในรูปแบบของ STRING ที่ถูก
         ปิดท้ายด้วย 03H
        ;R0 : หมายเลขบรรทัดที่ต้องการส่งข้อความไปแสดง
        ;R1 : รูปแบบการแสดงผล
        ;DPTR : เก็บจุดเริ่มต้นของ STRING ที่ต้องการอ่านค่าและส่ง
        ;PPLPORT : หมายเลข ADDRESS ของพอร์ทแบบขนานของตัวควบคุมที่
         ต่อกับระบบแสดงผล
        ;*****************************************************
SEND:   PUSH DPH                    ;เก็บค่าตัวแปรที่เปลี่ยนค่าในโปรแกรม
        PUSH DPL
        PUSH 02
        MOV TEMP,DPH                ;เก็บค่าจุดเริ่มต้นของข้อความไว้ใช้ใน
                                     โปรแกรม
        MOV TEMP+1,DPL
```

```
        MOV DPTR,#PPLPORT       ;อ้างอิงหมายเลข ADDRESS ของพอร์ท
                                 แบบขนาน
                                ;ที่ต่อกับระบบ
        MOV A,#02H              ;ส่งรหัสแสดงจุดเริ่มต้นของเฟรม(STX)
        ACALL STROBE            ;ส่งค่าออกพอร์ทขนาด  และส่งสัญญาณ
                                 ควบคุม
                                ;การส่งข้อมูล
        MOV A,R0                ;หมายเลขบรรทัดที่ต้องการแสดง
        ORL A,#10H              ;จะถูกเปลี่ยนเป็นรหัสหมายเลขบรรทัด
        ACALL STROBE            ;ส่งข้อมูลออกทางพอร์ทขนาน
        MOV A,R1                ;รูปแบบการแสดงผล
        ORL A,#10H              ;จะถูกเปลี่ยนเป็นรหัสควบคุมการแสดงผล
        ACALL STROBE            ;ส่งค่าออกพอร์ทขนาน  และส่งสัญญาณ
                                 ควบคุม
                                ;การส่งข้อมูล
        MOV R2,#00H             ;จุดเริ่มต้นของข้อความ  ที่อยู่ในตาราง
                                 ข้อความ
SEND_L: PUSH DPH
        PUSH DPL
        MOV A,R2
        MOVC A,@A+DPTR          ;ดึงข้อความในตาราง ที่ต้องการส่งมา
                                 ใช้ครั้งละ 1
                                ;ตัว
        CJNE A,#03H,SEND_S      ;ถ้าตัวอักษรในตารางที่เรียกมาใช้เป็น
                                 ตัวอักษร
                                ;บอกจุดจบของ
        POP DPL                 ;ข้อความ จบการส่งข้อความ
        POP DPH
```

```
        SJMP SEND_E
SEND_S: MOV DPTR,#PPLPORT       ;ถ้าไม่ใช่
        ACALL STROBE            ;ถ้าค่าออกพอร์ทขนาด และส่งสัญญาณ
                                 ควบคุม
                                ;การส่งข้อมูล
        POP DPL                 ;เรียกจุดเริ่มต้นของตารางข้อความกลับ
                                 มา
        POP DPH
        INC R2                  ;ตัวอักษรตัวต่อไปที่ต้องอ่านจากตาราง
        SJMP SEND_L             ;ทำซ้ำจนกว่าจะพบจุดจบของข้อความ
SEND_E: MOV DPTR,#PPLPORT       ;พบจุดจบแล้ว ส่งรหัสบ่งบอกจุดจบของ
                                 การส่ง
                                ;ข้อมูลทาง
        MOV A,#03               ;พอร์ทขนาน
        ACALL STROBE            ;ส่งค่าออกพอร์ทขนาน และส่งสัญญาณ
                                 ควบคุม
                                ;การส่งข้อมูล
        POP 02
        POP DPL
        POP DPH
        RET
        ;*****************************************************
        ;ฟังก์ชั่นควบคุมสัญญาณควบคุมการส่งข้อมูล (STROBE)
        ;*****************************************************
STROBE: MOVX @DPTR,A            ;ส่งค่าข้อมูลออกทางพอร์ทขนาน
        SETB P1.1               ;บิท P1.1เป็นตัวควบคุมสัญญาณควบคุม
                                 การส่งข้อมูล
```

```
        CLR P1.1                ;(STROBE) เมื่อข้อมูลถูกส่งออกทาง
                                 พอร์ทขนานแล้ว
        MOV R7,#OFFH            ;บิทนี้จะถูกดึงให้อยู่ในระดับลอจิก 0
                                 เป็นเวลามาก
        DJNZ R7,$               ;10 มิลลิวินาที
        SETB P1.1               ;แล้วจึงคืนระดับปกติที่ลอจิก 1
        SET
        ;*************************************************
        ;การเรียกใช้โปรแกรมส่งข้อความ
        ;*************************************************
MAIN:   MOV RO,#00H             ;เลือกบรรทัดที่ 1
        MOV R1,#00H             ;รูปแบบการแสดงผลแบบธรรมดา 00H
        MOV DPTR,#TEXT          ;จุดเริ่มต้นของข้อความที่ต้องการส่ง
        ACALL SEND              ;เรียกโปรแกรมส่งข้อความ
        SJMP $

TEXT:   "LINE#1"+03H
```

ภาคผนวก ค

รูปแบบคำสั่งสำหรับระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์

เนื่องจากการสื่อสารของระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์ ระหว่างระบบเครือข่ายย่อยและระบบควบคุมหลักอยู่ในรูปแบบของการสื่อสารข้อมูลแบบอนุกรม ดังนั้นในการติดต่อเพื่อส่งคำสั่งและอ่านข้อมูลระหว่างระบบจำเป็นต้องอาศัยโปรโตคอล ซึ่งเป็นกลุ่มของรหัสแอสกีที่มีรหัสเริ่มต้น และรหัสปิดท้ายที่แน่นอนเป็นตัวบ่งบอกความต้องการในการติดต่อ

ดังนั้น ในภาคผนวก ค นี้จะสรุปโปรโตคอลคำสั่งในการติดต่อระหว่างไมโครคอมพิวเตอร์และระบบควบคุมการทำงานหลัก และระหว่างระบบควบคุมการทำงานหลักหรือไมโครคอมพิวเตอร์กับระบบเครือข่ายย่อย

1. รูปแบบโปรโตคอลการติดต่อระหว่างไมโครคอมพิวเตอร์กับระบบควบคุมหลัก

1.1 กลุ่มคำสั่งสำหรับเลือกรูปแบบการทำงาน

1.1.1 เลือกรูปแบบการเก็บข้อมูลแบบเวลาจริง

02H	10H	12H	10H	03H

1.1.2 เลือกรูปแบบการเก็บข้อมูลแบบไม่เป็นเวลาจริง

02H	10H	12H	11H	03H

1.2 กลุ่มคำสั่งตั้งค่าเริ่มต้น

1.2.1 กำหนดจำนวนระบบเครือข่ายย่อยให้กับระบบควบคุมหลัก

02H	10H	11H	14H	จำนวนเครือข่าย	03H

โดยที่ NUMBER คือจำนวนระบบเครือข่ายมีค่าอยู่ในช่วง 0-7

1.2.2 กำหนดช่วงเวลาการเก็บข้อมูล สำหรับการเก็บข้อมูลแบบไม่เป็นเวลาจริง

02H	10H	11H	16H	นาที	วินาที	03H

1.2.3 ส่งเวลาปัจจุบันจากไมโครคอมพิวเตอร์ให้กับระบบควบคุมหลัก

02H	10H	11H	10H	ชั่วโมง	นาที	วินาที	03H

1.2.4 กำหนดเวลาเริ่มต้นการเก็บข้อมูลแบบเวลาจริง

02H	10H	11H	11H	ชั่วโมง	นาที	วินาที	03H

1.2.5 กำหนดเวลาสิ้นสุดการเก็บข้อมูลแบบเวลาจริง

02H	10H	11H	11H	ชั่วโมง	นาที	วินาที	03H

1.2.6 กำหนดอัตราการสุ่มการข้อมูล (TRIGGERING RATE)

02H	10H	11H	13H	อัตรา	00H	03H

1.2.7 คำสั่งขอให้ระบบควบคุมหลักแสดงค่าเริ่มต้น

02H	10H	11H	17H	03H

1.3 กลุ่มคำสั่งควบคุมการทำงานของระบบควบคุมหลัก

1.3.1 คำสั่งลบข้อมูลในหน่วยความจำส่วนเก็บข้อมูล จากระบบเครือข่ายย่อย

02H	10H	13H	BANK	จุดเริ่มต้น	จุดสิ้นสุด	03H

หมายเหตุ จุดเริ่มต้น : ประกอบด้วยข้อมูลหลายเลขแอดเดรสเริ่มต้น 2 ไบท์

จุดสิ้นสุด : ประกอบด้วยข้อมูลหลายเลขแอดเดรสสิ้นสุด 2 ไบท์

1.3.2 คำสั่งให้ระบบควบคุมหลักกลับเข้าสู่รูปแบบการรับค่าข้อมูลจากระบบเมนู

02H	10H	14H	03H

1.3.3 คำสั่งให้ระบบควบคุมการทำงาน เริ่มการทำงานหลังจากที่กำหนดค่าเริ่มต้นแล้ว

02H	10H	16H	03H

1.3.4 คำสั่งขออ่านข้อมูลจากหน่วยความจำส่วนเก็บข้อมูล จากระบบเครือข่ายย่อย

02H	10H	10H	BANK	จุดเริ่มต้น	จุดสิ้นสุด	03H

หมายเหตุ จุดเริ่มต้น : ประกอบด้วยข้อมูลหลายเลขแอดเดรสเริ่มต้น 2 ไบท์

จุดสิ้นสุด : ประกอบด้วยข้อมูลหลายเลขแอดเดรสสิ้นสุด 2 ไบท์

2. **รูปแบบโปรโตคอลการติดต่อระหว่างไมโครคอมพิวเตอร์ หรือระบบควบคุมหลักกับระบบเครือข่ายย่อย**

2.1 คำสั่งกระตุ้นการสุ่มอ่านข้อมูลอนาลอก

02H	10H	03H

หรืออาจใช้รหัสตัวเดียวเป็น OEH

2.2 คำสั่งเลือกการทำงานแบบเวลาจริง

02H	11H	03H

2.3 คำสั่งเลือกการทำงานแบบไม่เป็นเวลาจริง

02H	12H	03H

2.4 คำสั่งขออ่านข้อมูลในกรณีที่ระบบเครือข่ายย่อย ถูกกำหนดให้ทำงานแบบเวลาจริง

02H	13H	03H

หรืออาจใช้รหัสตัวเดียวเป็น OFH

2.5 คำสั่งขออ่านกลุ่มข้อมูลในกรณีที่ระบบเครือข่ายย่อย ถูกกำหนดให้ทำงานแบบไม่เป็นเวลาจริง

02H	14H	03H

2.6 คำสั่งกำหนดรูปแบบของการส่งข้อมูลกลับสู่ระบบควบคุมหลัก ให้มีรหัสนำหน้าและปิดท้าย

02H	15H	03H

2.7 คำสั่งกำหนดรูปแบบของการส่งข้อมูลเป็นแบบข้อมูลเดี่ยว ๆ

02H	16H	03H

2.8 คำสั่งลบข้อมูลที่มีอยู่ในหน่วยความจำของระบบเครือข่ายย่อย

02H	17H	03H

2.9 คำสั่งให้ระบบเครือข่ายย่อยเริ่มทำงานหลังจากกำหนดตัวแปรแล้ว

02H	18H	03H

2.10 คำสั่งกำหนดตัวแปรอัตราการขยายสัญญาณ

02H	19H	อัตราการขยาย 00-FFH	03H

2.11 คำสั่งหยุดการทำงานของระบบเครือข่ายย่อย

02H	1AH	03H

2.12 คำสั่งเพื่ออ่านข้อมูลจากช่องอ่านข้อมูลแบบดิจิตอลในขณะนั้น

02H	1BH	03H

2.13 คำสั่งเพื่อส่งตัวแปรกำหนดอัตราการขยายสัญญาณ

02H	1CH	03H

2.14 คำสั่งเพื่อส่งตัวแปรกำหนดอัตราการสุ่มอ่านข้อมูลในรูปแบบไม่เป็นเวลาจริง

02H	1DH	ตัวแปร	03H

โดยที่ค่าของตัวแปรจะเป็นค่าของตัวคูณ กับค่าคงที่ของอัตราการสุ่มอ่านพื้นฐาน 200 ไมโครวินาที

ภาคผนวก ง

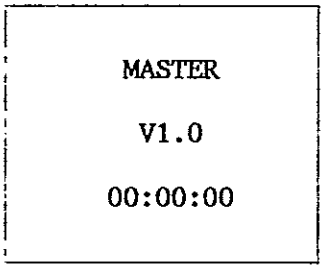
การใช้งานระบบเก็บข้อมูลระยะไกลโดยใช้ไมโครโปรเซสเซอร์

วิธีใช้งาน

หลังจากที่ได้ติดตั้งระบบเครือข่ายย่อยจากจุดทดลองต่าง ๆ ตามจำนวนที่ต้องการเข้ากับช่องรับข้อมูลแบบอนุกรมของระบบควบคุมหลักแล้วก็สามารถเปิดระบบเก็บข้อมูลระยะไกลเพื่อใช้งานได้ทันที

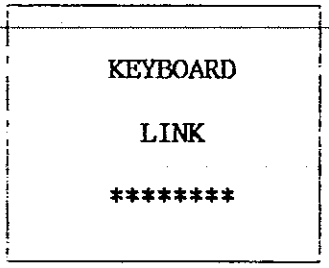
1. การกำหนดรูปแบบการทำงานของระบบเก็บข้อมูลระยะไกล

หลังจากที่ผู้ใช้งานได้เปิดเครื่องเพื่อทำงานแล้วส่วนแสดงผลของระบบจะแสดงหมายเลข VERSION ของระบบเก็บข้อมูลตามรูป

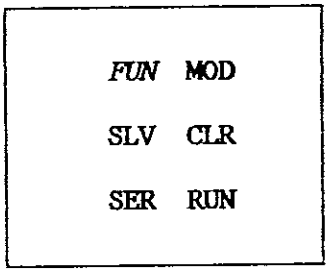


โดยที่บรรทัดสุดท้ายของส่วนแสดงผลจะแสดงเวลาปัจจุบัน

หลังจากที่ผู้ใช้งานมีการกดแป้นพิมพ์ใด ๆ ของระบบเก็บข้อมูล ส่วนแสดงผลจะเปลี่ยนเป็น



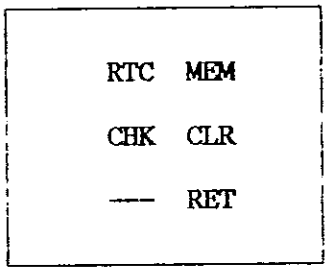
และจะค้างข้อความไว้ประมาณ 10 วินาที ก่อนที่จะเข้าสู่การแสดงเมนูของระบบซึ่งมีการจัดเรียงรูปแบบดังนี้



หมายเหตุ ตัวอักษรเอนจะเป็นส่วนแสดงตัวอักษรแบบ REVERSE ในส่วนแสดงผลจริง

ความหมายของเมนูตัวเลือกจะเป็นดังนี้

1.1 FUNCTION เป็นตัวเลือกเพื่อใช้ฟังก์ชั่นช่วยเหลือการทำงาน ซึ่งในกรณีที่ผู้ใช้งานต้องการเรียกฟังก์ชั่นเพื่อใช้งาน ทำได้โดยการเลือกแถบเมนูมายังตำแหน่ง FUN และกดแป้นพิมพ์ ENTER ในส่วนแสดงผลจะแสดงเมนูย่อยของเมนูหลัก FUNCTION ดังนี้



ซึ่งในส่วนของเมนูย่อยจะมีความหมายดังนี้

1.1.1 RTC (REAL TIME CLOCK) เป็นตัวเลือกเมื่อผู้ใช้งานต้องการตั้งเวลาปัจจุบันให้กับระบบเก็บข้อมูลใหม่ ในการเข้าสู่ฟังก์ชั่นบริการย่อยสามารถกระทำได้โดยการกดแป้นพิมพ์ ENTER เมื่อแถบเมนูเลื่อนมายังตำแหน่ง RTC และเมื่อเข้าสู่ฟังก์ชั่นนี้แล้วการเปลี่ยนแปลงค่าเวลาใช้รูปแบบการเลือกแถบเมนูไปยังค่าที่ต้องการเปลี่ยนแปลง กดแป้นพิมพ์ ENTER และใช้แป้นพิมพ์ UP เพื่อเพิ่มค่าขึ้นและ DOWN เพื่อกดค่าลง และเมื่อต้องการรับค่านั้น ๆ ทำได้โดยการกดแป้นพิมพ์ ENTER

1.1.2 MEM (MEMORY) เป็นตัวเลือกเมื่อผู้ใช้งานต้องการ-ทดสอบสภาวะการทำงานของหน่วยความจำเก็บข้อมูล ของระบบควบคุมหลักโดยฟังก์ชั่นนี้จะรายงานสภาวะของการอ่านข้อมูลและผลของการตรวจสอบทุก ๆ BANK ของหน่วยความจำส่วนขยาย

1.1.3 CHK (CHECK) เป็นตัวเลือกเพื่อการตรวจสอบหาความผิดพลาดของการเชื่อมโยงและการทำงานของระบบเครือข่ายย่อย เพื่อที่จะได้ทำการแก้ไขก่อนที่จะเริ่มกระบวนการเก็บข้อมูล

1.1.4 CLR (CLEAR) เป็นตัวเลือกเมื่อผู้ใช้งานต้องการลบค่าของหน่วยความจำส่วนเก็บข้อมูลส่วนขยาย โดยสามารถเลือกลบเฉพาะบาง BANK หรือลบทั้งหมด

1.1.5 RET (RETURN) กลับไปสู่เมนูหลัก

1.2 MODE เป็นตัวเลือกเพื่อเข้าสู่ฟังก์ชั่นกำหนดรูปแบบการทำงานของระบบเก็บข้อมูล เมื่อผู้ใช้งานเลือกฟังก์ชั่นนี้แล้ว ระบบควบคุมจะแสดงเมนูย่อยของรูปแบบการทำงานเป็นสองรูปแบบย่อยคือ

1.2.1 รูปแบบการทำงานแบบเวลาจริง (เมนูย่อย REAL-T) ในรูปแบบการทำงานนี้ผู้ใช้งานจำเป็นต้องป้อนตัวแปรต่าง ๆ โดยที่ระบบจะจัดลำดับการรับค่าตัวแปรโดยอัตโนมัติ ดังนี้

ก. เวลาเริ่มต้นของการเก็บข้อมูล โดยเลื่อนแถบเมนูไปยังตำแหน่งที่ต้องการเปลี่ยนแปลงค่าและใช้แป้นพิมพ์ UP และ DOWN เพื่อเปลี่ยนแปลงค่าในการรับค่าใช้การกดแป้นพิมพ์ ENTER

ข. เวลาสิ้นสุดของการเก็บข้อมูล สามารถกระทำการเปลี่ยนแปลงค่าได้โดยใช้ขั้นตอนเช่นเดียวกับการตั้งค่าเวลาเริ่มต้น

ค. อัตราการสุ่มอ่านข้อมูล ในการเปลี่ยนแปลงค่าใช้แป้นพิมพ์ UP และ DOWN เพื่อเลือกค่าที่กำหนด เช่น 1/30,1/20,1/10,1,2 วินาที และเมื่อได้ค่าที่ต้องการแล้วใช้การกดแป้นพิมพ์ ENTER

หลังจากที่มีการป้อนค่าตัวแปรตามลำดับแล้ว ระบบจะกลับเข้าสู่เมนูหลัก

1.2.2 รูปแบบการทำงานแบบไม่เป็นเวลาจริง(N-REAL) ผู้ใช้จำเป็นต้องป้อนค่าตัวแปร ตามลำดับดังต่อไปนี้

ก. ช่วงเวลาการเก็บข้อมูล ใช้การป้อนค่าเฉพาะนาทีและวินาที โดยรูปแบบเช่นเดียวกับการตั้งค่าเวลาเริ่มต้นของการเก็บข้อมูลที่กล่าวไปแล้ว

ข. อัตราการสุ่มอ่านข้อมูลซึ่งมีอัตราตัวเลือกที่แตกต่างจากกรณีเลือกรูปแบบการทำงานแบบเวลาจริง แต่วิธีใช้เหมือนกัน

หมายเหตุ ในการออกจากขั้นตอนการเปลี่ยนแปลงค่าตัวแปรของเวลาเริ่มต้น สิ้นสุดให้เลื่อนแถบเมนูไปยังหลักวินาที ส่วนการออกจากขั้นตอนการเปลี่ยนแปลงตัวแปรช่วงเวลาการเก็บข้อมูลให้เลือกแถบเมนูไปยังหลักชั่วโมง และกดแป้นพิมพ์ ENTER

สำหรับในกรณีอื่น ๆ การกดแป้นพิมพ์ ENTER มีผลต่อการเสร็จสิ้นและรับค่าตัวแปรเหล่านั้น

1.3 SLV (SLAVE) เป็นเมนูสำหรับค่าจำนวนระบบเครือข่ายที่ต้องการใช้งาน การป้อนค่าใช้วิธีกดแป้นพิมพ์ UP DOWN และ ENTER

1.4 LST (LIST) เพื่อต้องการดูตัวแปรที่ได้ป้อนค่าเข้าสู่ระบบแล้ว

1.5 SERIAL เพื่อต้องการเลือกการรับการตั้งค่าตัวแปรต่าง ๆ จากไมโครคอมพิวเตอร์

1.6 RUN เป็นตัวเลือกในการสั่งระบบให้ทำงานตามรูปแบบการทำงานและตัวแปรที่กำหนดไว้ก่อนแล้ว

2. สถานะของระบบเก็บข้อมูลระยะไกลขณะทำงาน

สถานะการทำงานของระบบเก็บข้อมูลระยะไกลแบ่งเป็น 2 รูปแบบ

2.1 ในรูปแบบการทำงานแบบเวลาจริง ในกรณีที่เวลาปัจจุบันยังไม่ถึงเวลาเริ่มต้นการเก็บข้อมูล ในส่วนแสดงผลจะแสดง

เวลาปัจจุบัน
เวลาเริ่มต้น

และจะรอจนกว่าจะถึงเวลาเริ่มต้น เมื่อถึงเวลาเริ่มต้นการทำงานแล้วก็จะแสดงเวลาเริ่มต้นและสิ้นสุด

2.2 ในรูปแบบการทำงานแบบไม่เป็นเวลาจริง ส่วนแสดงผลจะแสดงเวลาปัจจุบันในบรรทัดแรกของภาคแสดงผล และช่วงเวลาที่เหลือในบรรทัดสุดท้าย เมื่อช่วงเวลาสิ้นสุดก็จะเริ่มอ่านข้อมูลจากระบบเครือข่ายย่อยแต่ละตัว และกลับเข้าสู่เมนูหลักเมื่อเสร็จสิ้นกระบวนการ

3. การอ่านข้อมูลเพื่อทำการวิเคราะห์โดยไมโครคอมพิวเตอร์

ไมโครคอมพิวเตอร์ สามารถอ่านข้อมูลจากหน่วยความจำส่วนขยายของระบบเก็บข้อมูลได้ตลอดเวลาที่ระบบเก็บข้อมูลแสดงผลอยู่ในเมนูหลักโดยที่โปรแกรมควบคุมระบบเก็บข้อมูลหลัก พร้อมที่จะเชื่อมโยงการสื่อสารข้อมูลเข้ากับไมโครคอมพิวเตอร์ทันทีที่มีคำสั่งออกมาจากระบบการสื่อสารข้อมูล

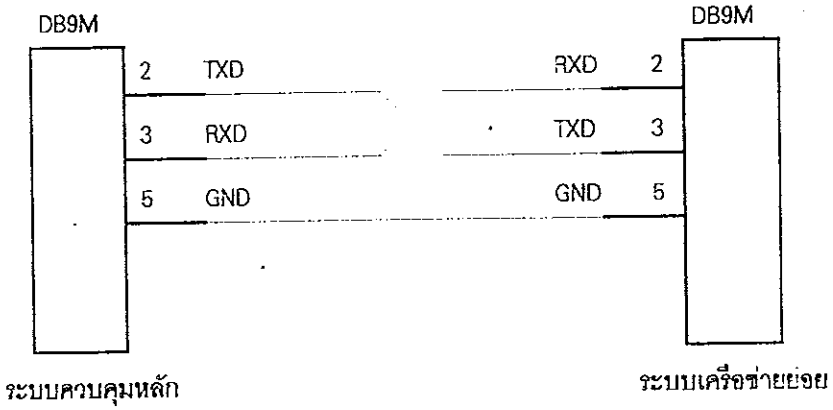
ปัญหาและวิธีการแก้ไข

โดยปกติแล้วระบบเก็บข้อมูลระยะไกลได้ถูกแก้ไขให้ยืดหยุ่นต่อการใช้งานและความผิดพลาดต่าง ๆ แล้ว เช่น ในขณะที่ระบบควบคุมหลักกำลังอ่านข้อมูลจากระบบเครือข่ายย่อยและเกิดการขาดของสายสื่อสารข้อมูล ซึ่งทำให้การสื่อสารข้อมูลผิดพลาด ระบบจะข้ามการอ่านข้อมูลจากระบบเครือข่ายย่อยนั้น ๆ ทันที ทำให้ระบบยังสามารถอ่านข้อมูลจากระบบเครือข่ายย่อยตัวอื่นได้

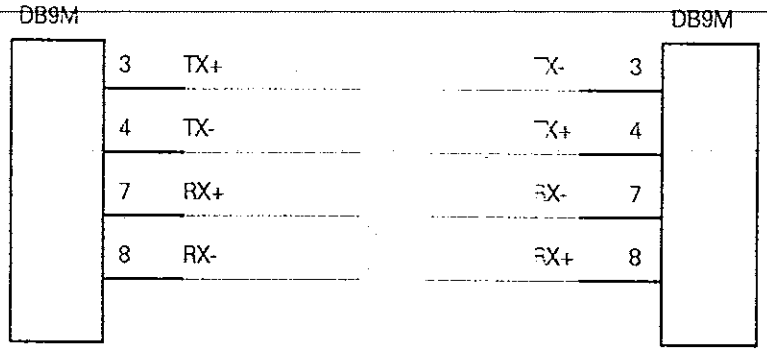
สำหรับปัญหาเท่าที่ได้ทดสอบและพบความผิดพลาด เช่น ในกรณีที่เวลาเริ่มต้นของกระบวนการเก็บข้อมูล มีค่าน้อยกว่าเวลาปัจจุบัน ทำให้ระบบจำเป็นต้องรอนานกว่าที่จะเริ่มทำงาน และผู้ใช้งานได้สั่งทำงานไปแล้ว สามารถแก้ไขโดยการกดแป้นพิมพ์ในแนวตั้งพร้อมกันทั้งหมด ซึ่งมีความหมายเท่ากับการ BREAK ระบบ

3. รูปแบบของการต่อสายเชื่อมโยงของการสื่อสารข้อมูลแบบอนุกรม

1. กรณีของสายสื่อสารข้อมูลแบบอนุกรมมาตรฐาน RS-232C



2. กรณีของสายสื่อสารข้อมูลแบบอนุกรมมาตรฐาน RS-422



ประวัติผู้เขียน

ชื่อ นายวัฒนพงศ์ เกิดทองมี

วัน เดือน ปีเกิด 16 มิถุนายน 2512

วุฒิการศึกษา

วุฒิ	ชื่อสถาบัน	ปีที่สำเร็จการศึกษา
วิทยาศาสตรบัณฑิต(ฟิสิกส์)	มหาวิทยาลัยสงขลานครินทร์	2533

ทุนการศึกษาที่ได้รับระหว่างการศึกษา

ได้รับทุนการศึกษาจากโครงการพัฒนาและส่งเสริมผู้มีความสามารถพิเศษทางวิทยาศาสตร์และเทคโนโลยี (โครงการ พสวท.) ตั้งแต่การศึกษาระดับปริญญาตรีจนถึงปริญญาเอก

VITAE

Name Mr.Wattanapong Kurdthongmee

Birthdaay 16 June 1969

Educational Attainment

Degree	Name of Institution	Year of Graduation
Bachelor of Science	Prince of Songkhla University	1990

Scholarship Awards during Enrolment

A scholarship from the Institute for the Promotion of Teaching and Technology under the IPDT project from bachelor degree to doctorate degree.