ปีการศึกษา 2535

การควบคุมโดยใช้ IEEE-488 (GPIB)

โดย

นายชัยยุทธ	สุพัฒน์โสภณ	32.1074
นายณุวัฒน์	จันทร์ถาวร	32.1092

อาจารย์ที่ปรึกษา

รศ.ดร.มนัส สังวรศิลป์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032558

ปริญญานิพนธ์ปีการศึกษา 2535

ภาควิชาอิเลคทรอนิคส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การควบคุมโดยใช้ IEEE-488 (GPIB)

ผู้จัดทำ

1. นายชัยยุทธ สุพัฒน์โสภณ 32.1074
2. นายณุวัฒน์ จันทร์ถาวร 32.1092

_____อาจารย์ที่ปรึกษา

(รศ.ดร.มนัส สังวรศิลป์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032558

The controling method by using IEEE488 (GPIB)

Chaiyut Supatsopol 32.1074

Nuwat Chanthaworn 32.1092

Advisor

Professor

Manas Sungwornsilpa Ph.D.

Academic year 2535

ABSTRACT

This thesis is about controling with IEEE488 (GPIB) , that is sending information standard system IEEE. It use in the industrial factory. Now it is sending between controller (PC) and Talker or listener example oscilloscope, spectrum analyzer, DMM,printer etc.,or sending information between Talker and listener. It sends parallel information to data bus size 8 bit and control information system with control line 8 bit . In this thesis,it is operating with spectrum analyzer. In addition to do simulate circuit. That is the our measuring instrument.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมโดยใช้ IEEE488 (GPIB)

ชัยยุทธ สุพัฒน์โสภณ 32.1074
ณุวัฒน์ จันทร์ถาวร 32.1092
อาจารย์ที่ปรึกษา
รศ.ดร.มนัส สังวรศิลป์
ปีการศึกษา 2535

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้กล่าวถึง การควบคุมโดยใช้ IEEE 488 (GPIB) ซึ่งเป็นระบบมาตราฐานการส่งข้อมูลของIEEE ซึ่งได้ใช้ในโรงงานอุตสาหกรรมในปัจจุบัน เป็นการส่งข้อมูลระหว่างตัว CONTROLLER(PC) กับตัว TALKERหรือLISTENER เช่น OSCILLOSCOPE, SPECTRUM ANALYSER, DMM, PRINTER เป็นต้น หรือเป็นการส่งข้อมูลระหว่างตัว TALKER กับ LISTENER มีการส่งข้อมูลแบบขนานเข้า data bus ขนาด 8 บิต และควบคุมการส่งข้อมูลโดย control line 8 บิต โดยในปริญญานิพนธ์ฉบับนี้ได้ทำการติดต่อสื่อสารกับเครื่อง SPECTRUM ANALYSER รวมทั้งได้ทำวงจรที่ simulate เป็นอุปกรณ์เครื่องมือวัดขึ้นอีกหนึ่งตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทนำ	1
ประวัติการพัฒนา IEEE-488	1
บทที่ 1	2
โครงสร้างของ IEEE-488	2
ขีดจำกัดของ IEEE-488	2
รายละเอียดเกี่ยวกับ IEEE-488	3
ความหมายของสัญญาณต่างๆภายใน IEEE-488	4
การเชื่อมต่ออุปกรณ์ต่างๆในระบบ IEEE-488	6
ขบวนการแฮนด์เชค (Hand shake procedure)	9
ขบวนการแฮนด์เชคของตัวควบคุมซึ่งทำหน้าที่เป็นตัวส่ง	11
ขบวนการแฮนด์เชคของตัวควบคุมซึ่งทำหน้าที่เป็นตัวรับ	12
คำสั่งใช้งานของ GPIB	14
การขอบริการและการตรวจสอบ	18
รูปแบบของข้อมูล	18
บทที่ 2	21
การออกแบบวงจรที่ simulate เป็นเครื่องมือวัด	21
การเขียนโปรแกรมควบคุม microprocesser	26
โปรแกรมควบคุม microprocesser	27
บทที่ 3	38
การออกแบบแผนผังการทำงานโปรแกรมควบคุม Spectrum Analyzer	38
โปรแกรมควบคุม Spectrum Analyzer	40
ส่วนของโปรแกรมแสดงภาพหน้าจอ	53
ส่วนของโปรแกรมควบคุมเมาส์	72
สรุปและวิจารณ์ผลการทดลอง	83

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ
การควบคุมด้วย IEEE-488

ระบบอุตสาหกรรมในอดีตมักใช้อุปกรณ์ไม่มากชิ้น หากมีการเชื่อมต่อก็สามารถทำได้โดยง่าย ต่อมาการอุตสาหกรรมได้เจริญรุดหน้าไปอย่างรวดเร็ว ดังนั้นระบบอุตสาหกรรมจึงมีขนาดใหญ่ และ ซับซ้อนขึ้นด้วย การเชื่อมต่อทางอุตสาหกรรมจึงมีความซับซ้อนขึ้นตามไปด้วย หากจะมีการเชื่อมต่อระหว่างอุปกรณ์หลายๆชิ้นจะต้องเสียค่าใช้จ่ายจำนวนมาก ดังนั้นจึงมีการคิดค้นระบบเชื่อมต่อซึ่งเป็นมาตรฐานขึ้นมา นั่นคือIEEE-488นั่นเอง

ประวัติการพัฒนาIEEE-488

ดังที่ได้กล่าวมาแล้วข้างต้นว่าการเชื่อมต่อทางอุตสาหกรรมในอดีตเป็นไปด้วยความยากลำบากและเสียค่าใช้จ่ายมาก ดังนั้นบริษัทผู้ผลิตเครื่องมือวัดต่างๆในประเทศสหรัฐอเมริกาจึงร่วมกันจัดหาระบบเชื่อมต่อมาตรฐานขึ้นมา ซึ่งในประเทศเยอรมันก็มีการพัฒนาระบบเชื่อมต่อมาตรฐานเช่นกันโดยความร่วมมือของ IEC (International Electro-technical Commission) จนกระทั่งในปี 1972 สหรัฐอเมริกาโดยการนำของ IEEE (Institute of electrical and Electronics Engineers : IEEE) จึงได้มีการประชุมเพื่อวางแผนพิจารณาระบบเชื่อมต่อมาตรฐานร่วมกัน

บริษัทฮิวเล็ตแพกการ์ด ผู้ผลิตเครื่องมือวัดรายใหญ่ในอเมริกา ได้ทำการพัฒนาระบบเชื่อมต่อมาตรฐานอยู่ก่อนแล้วชื่อว่า HPIB (Hewwlett Packard Interface Bus)จึงได้เสนอโครงการให้ IEEE เพื่อพิจารณา และ ได้รับการยอมรับในปี 1975 โดย IEEE จัดให้เป็นมาตรฐานลำดับที่ 488 ดังนั้นจึงได้ชื่อว่า IEEE-Std 488-1975 ซึ่งต่อมาได้มีการปรับปรุงเป็น IEEE-Std 488-1978 หรือที่นิยมเรียกกันว่า GPIB (General Purpose Interface Bus)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

IEEE-488 (GPIB)

โครงสร้างของ IEEE-488

ในระบบพื้นฐานของ GPIB จะประกอบด้วยอุปกรณ์ คือ ผู้ส่ง(Talker) , ผู้รับ (Listener) และ ผู้ควบคุม (Controller)

- Talker ทำหน้าที่ส่งข้อมูล โดยในระบบสามารถมี Talker ได้หลายตัว แต่จะมีเพียงตัวเดียวเท่านั้นที่กำลังทำงานอยู่

- Listener ทำหน้าที่เป็นตัวรับข้อมูล โดยในระบบเดียวกันสามารถมีListener ได้หลายตัวเช่นเดียวกัน แต่ Listener สามารถทำงานได้ครั้งละหลายๆตัว นั่นคือ ในระบบที่กำลังทำงานจะมี Talker 1 ตัว แต่มี Listener หลายตัวได้

- Controller ทำหน้าที่ควบคุมอุปกร์ต่างๆในระบบ โดยจะกำหนดให้ Talker ทำการส่งข้อมูล หรือ กำหนดให้ Listener ทำการรับข้อมูล

อุปกรณ์ที่มี GPIB นั้นสามารถแบ่งตามหน้าที่ได้ดังนี้

1. ทำหน้าที่เป็น Talker เท่านั้น เช่น เครื่องมือวัด เป็นต้น

2. ทำหน้าที่เป็น Listener เท่านั้น เช่น เครื่องพิมพ์(Printer),เครื่องบันทึก (Recorder) เป็นต้น

3. ทำหน้าที่เป็นทั้ง Talker และ Listener เช่น คอมพิวเตอร์,เครื่องมือวัดที่สามารถควบคุมได้จากภายนอก เป็นต้น

4. ทำหน้าที่เป็น Talker Listener และ Controller ในตัวเดียวกัน เช่น คอมพิวเตอร์ที่ทำหน้าที่ควบคุมระบบ

ขีดจำกัดของ IEEE-488

1. จำนวนอุปกรณ์ในระบบ(Talker,Listener,Controller)ที่ต่อกับสายสัญญาณ 1 เส้น จะต้องมีไม่เกิน 15 เครื่อง

2. สายเคเบิ้ลที่ต่อระหว่างอุปกรณ์ จะต้องยาวไม่เกิน 4 เมตร และ ความยาวรวมของสายเคเบิ้ลในระบบจะต้องไม่เกิน 20 เมตร

3. ความเร็วในการส่งข้อมูลจะต้องไม่เกิน 1Mb/Sec(1ล้านไบต์ต่อวินาที)

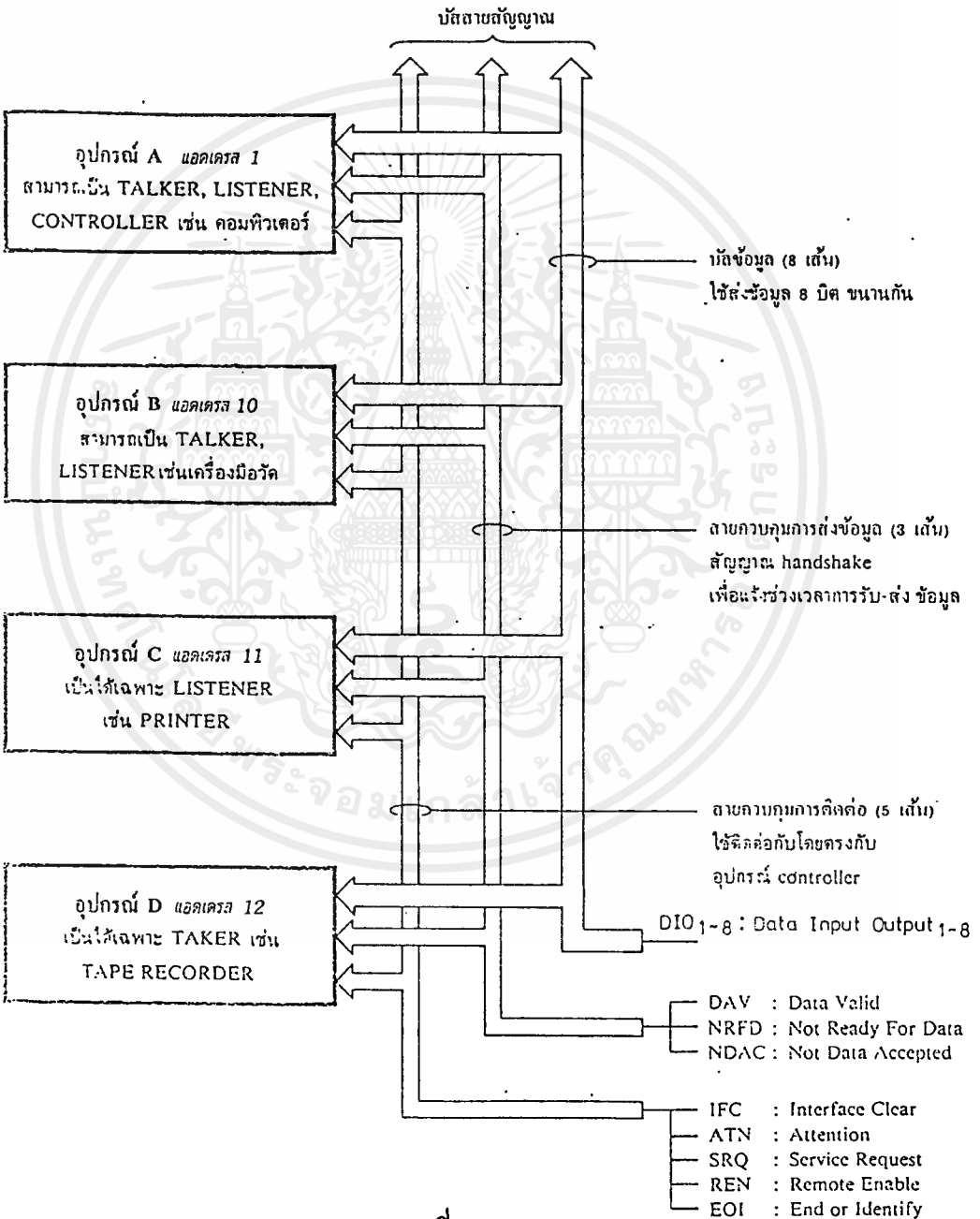
4. ต้องมีการจ่ายไฟให้กับอุปกรณ์มากกว่าครึ่งหนึ่งของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดเกี่ยวกับ IEEE-488

ลักษณะทางกายภาพของ IEEE-488 นั้น คือ เป็นสายสัญญาณแบบ 24 เส้นขนานกัน และมีขั้วต่ออยู่ทางปลายทั้งสองของสาย เพื่อต่อกับอุปกรณ์ หรือ ต่อกันเพื่อให้สายสัญญาณมีความยาวเพิ่มขึ้น ในจำนวนสายสัญญาณ 24 เส้นนี้มีเพียง 16 เส้นเท่านั้น ที่ทำหน้าที่นำสัญญาณ ส่วนที่เหลืออีก 8 เส้น ทำหน้าที่กราวนด์(ground) และ ชีลด์(shield)

โดยจำนวนสายที่ใช้นำสัญญาณ 16 เส้นนั้นยังแบ่งได้เป็น 3 ประเภท ตามรูปที่ 1 คือ



รูปที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.บัสข้อมูล(Data Bus) จำนวน 8 สาย คือ

DIO1-DIO8

2.สายสัญญาณควบคุม(Control Line) จำนวน 5 สาย คือ

IFC (Interface Clear)

ATN (Attention)

SRQ (Service request)

REN (remote Enable)

EOI (End or Identify)

3.สายแฮนด์เชค(Hand Shake) 3 สาย คือ

DAV (Data Valid)

NRFD (Not Ready For Data)

NDAC (Not Data Accepted)

ความหมายของสัญญาณต่างๆภายใน IEEE-488

ดังที่ได้กล่าวมาแล้วว่าสายสัญญาณต่างๆใน GPIB ได้แบ่งออกเป็น 3กลุ่ม ในหัวข้อนี้จะอธิบายความหมายของสัญญาณต่างๆดังนี้

กลุ่มสัญญาณข้อมูล

1.DIO1-DIO8 (Data Input/Output) สายสัญญาณทั้ง 8 เส้นนี้ทำหน้าที่เป็นทางผ่านของข้อมูลในระบบ

กลุ่มสัญญาณควบคุมการเชื่อมต่อ(Interface)

1.IFC (Interface Clear) เป็นสัญญาณรีเซต หรือ เคลียร์ระบบ กำเนิดได้โดยตัวควบคุม(Controller)เท่านั้น เมื่ออุปกรณ์ในบัสได้รับสัญญาณเคลียร์นี้ จะกลับคืนสู่สภาวะเริ่มต้นใหม่ ซึ่งเป็นสภาวะแรกเริ่มก่อนการกำหนดฟังก์ชันเหมือนแรกเปิดสวิตช์

2.ATN (Attention) เป็นสัญญาณที่ถูกส่งโดยอุปกรณ์ที่เป็นตัวควบคุมเช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้ในการสั่งให้อุปกรณ์ทุกตัวในระบบเตรียมพร้อมเพื่อรอรับคำสั่งต่อไป

3.SRQ (Service Request) เป็นสัญญาณที่ถูกส่งจากอุปกรณ์ต่างๆ เพื่อเป็นการบอกแก่ระบบว่าขณะนี้อุปกรณ์ดังกล่าวต้องการติดต่อจากตัวควบคุม

4.REN (Remote Enable) สัญญาณนี้เป็นสัญญาณที่ถูกส่งมาจากตัวควบคุมเพียงตัวเดียวเท่านั้น เพื่อใช้สั่งให้อุปกรณ์ต่างๆเปลี่ยนจากโหมดที่ใช้งานปกติมาเป็นการควบคุมโดยตัวควบคุมแทน

5.EOI (End or Identify) เป็นสัญญาณที่ถูกส่งได้โดยอุปกรณ์ที่เป็นตัวควบคุม (Controller) หรืออุปกรณ์ที่เป็นตัวส่ง(Talker)ก็ได้ ใช้สำหรับแสดงว่าข่าวสารที่ส่งเป็นชุดนั้นได้เสร็จสิ้นลงแล้ว

กลุ่มสัญญาณควบคุมการรับ-ส่งข้อมูล

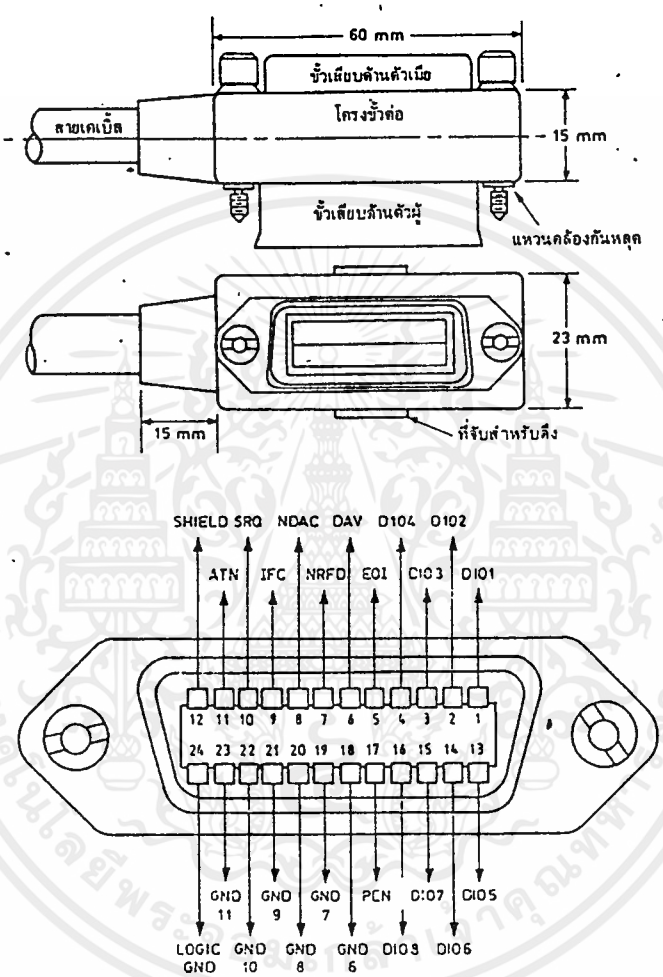
1.DAV (Data Valid) เมื่อสัญญาณนี้ถูกดึงเป็นลอจิก "Low" โดยอุปกรณ์ที่เป็นตัวส่ง(Talker) เป็นการแจ้งแก่ระบบบัสว่า ขณะนี้ตัวส่งได้ทำการส่งข้อมูลลงไปที่สายสัญญาณข้อมูลเรียบร้อยแล้ว

2.NRFD (Not Ready For Data) เมื่อสัญญาณนี้มีลอจิกเป็น "Low" จะเป็นการแสดงว่าในขณะนี้ระบบบัสยังไม่พร้อมที่จะรับข้อมูล เนื่องจากอุปกรณ์ในระบบยังพร้อมไม่หมดทุกตัว ซึ่งสัญญาณเส้นนี้จะไม่เป็น "Hi" จนกว่าอุปกรณ์ทุกตัวให้ลอจิกที่เป็น "Hi" ครบถ้วนแล้ว สัญญาณนี้มีประโยชน์ในกรณีที่อุปกรณ์ในระบบมีความเร็วแตกต่างกัน

3.NDAC (Not Data Accepted) สัญญาณเส้นนี้เป็นสัญญาณที่ถูกควบคุมโดยอุปกรณ์ที่เป็นตัวรับ(Listener) โดยสัญญาณนี้จะมีลอจิกเป็น "Low" ในขณะที่อุปกรณ์ที่เป็นตัวรับกำลังเก็บข้อมูลจากสายข้อมูล(Data Bus) และจะเป็น "Hi" เมื่ออุปกรณ์นั้นได้ทำการอ่านข้อมูลเสร็จเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยสัญญาณลอจิกที่ใช้ใน DATA BUS(D1-D8) ของ IEEE-488 Bus นี้มีลักษณะเป็นคอมพลีเมนต์ทั้งหมด คือ "1" เท่ากับ "Low" และ "0" เท่ากับ "Hi" ซึ่งตรงข้ามกับในวงจรลอจิกที่เราคุ้นเคย



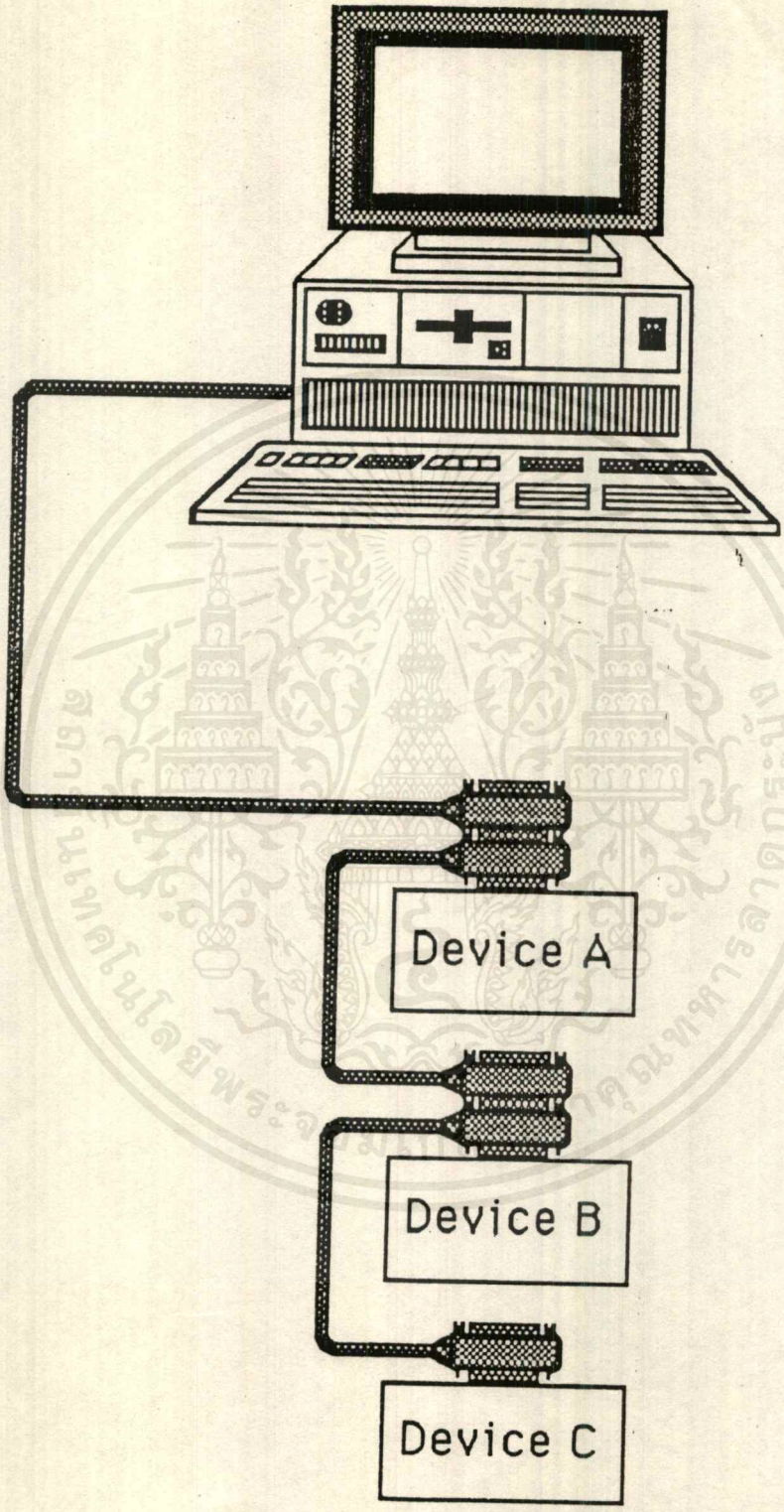
รูปที่ 2 ขั้วต่อของ GPIB และการจัดขาของสัญญาณต่างๆ

การเชื่อมต่ออุปกรณ์ต่างๆในระบบ IEEE-488 BUS

สำหรับการเชื่อมต่อระหว่างอุปกรณ์ต่างๆในระบบ IEEE-488 Bus นั้น มีอยู่ 2 วิธี คือ

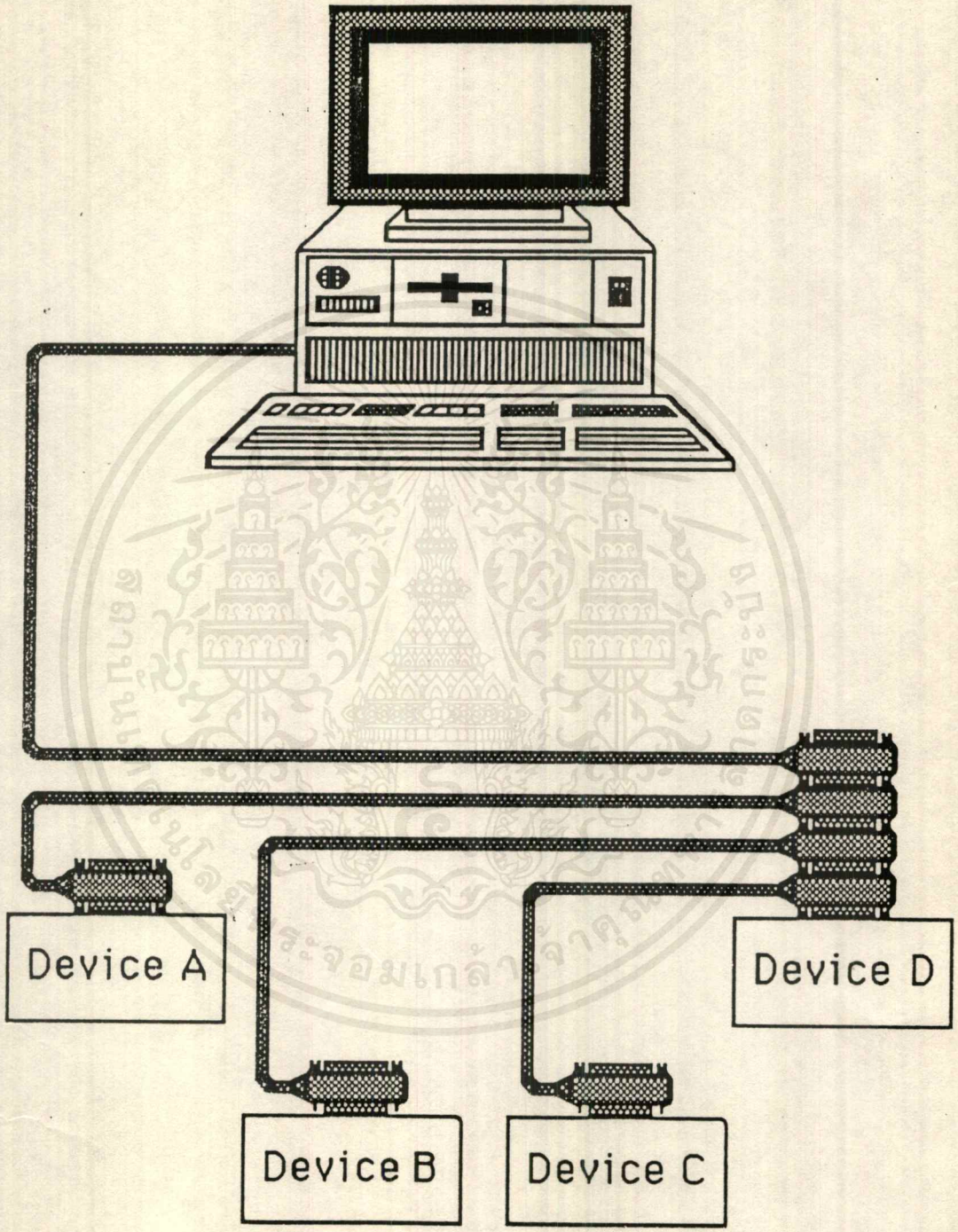
1. การเชื่อมต่อแบบเรียงต่อเนื่องกัน (Daisy Chain Configuration)
2. การเชื่อมต่อแบบกระจาย (Star Configuration)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3 การเชื่อมต่อแบบเรียงต่อเนื่องกัน (Daisy Chain Configuration)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



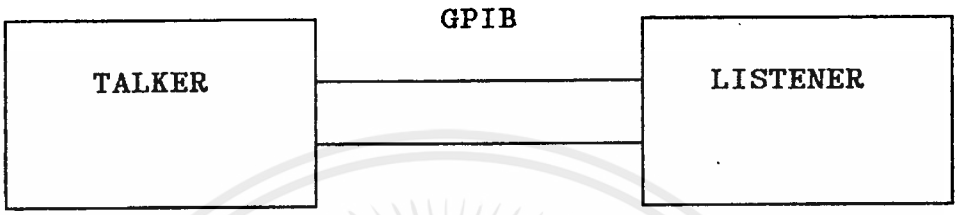
รูปที่ 4 การเชื่อมต่อแบบกระจาย (Star Configuration)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

ขบวนการแฮนด์เชค(Handshake Procedure)

เมื่อมีการเชื่อมต่อระหว่างอุปกรณ์ต่างๆในระบบ ดังนั้นเมื่อระบบจะทำงานจึงต้องมีการตรวจสอบเสียก่อน ซึ่งการตรวจสอบนี้เราเรียกว่าขบวนการแฮนด์เชค (Handshake Procedure)



รูปที่ 5

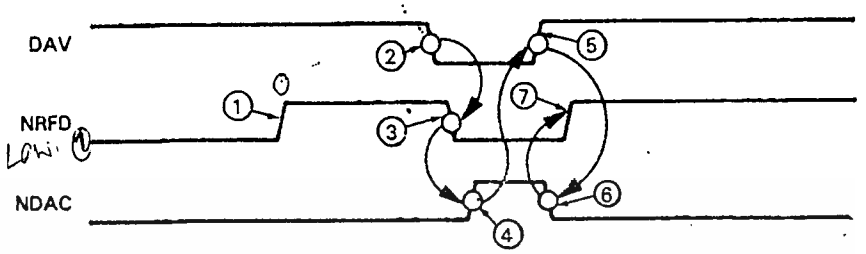
พิจารณาระบบที่ไม่ซับซ้อนนักโดยกำหนดให้มีตัวส่ง(Talker)และตัวรับ(Listener) อย่างละ 1 ตัว

การที่จะมีการสื่อสารระหว่างตัวส่งและตัวรับจะกระทำได้โดยการที่ตัวส่งจะทำหน้าที่ส่งสัญญาณออกมาเพื่อให้ตัวรับทราบว่ามีข้อมูลในสายสัญญาณ และตัวรับจะทำการเก็บข้อมูลจากสายสัญญาณ เมื่อเก็บข้อมูลเสร็จเรียบร้อยแล้วตัวรับก็จะส่งสัญญาณเพื่อให้ตัวส่งทราบว่าได้ทำการรับข้อมูลเป็นที่เรียบร้อยแล้ว

การส่งสัญญาณในการสื่อสารระหว่างตัวส่งและตัวรับนั้น จะส่งมาทางสายสัญญาณ 3 สาย ซึ่งสายสัญญาณดังกล่าวนั้นเป็นสายสัญญาณในกลุ่มควบคุมการรับ-ส่งข้อมูล คือ NRFD (Not Ready For Data),DAV (Data Valid) และNDAC (Not Data Accepted)

DAV เป็นสัญญาณที่ถูกส่งโดยตัวส่ง NRFD และ NDAC เป็นสัญญาณที่ถูกส่งโดยตัวรับ โดยตัวรับจะใช้สัญญาณNDACเพื่อชี้ให้เห็นว่า ในขณะนั้นพร้อม หรือไม่พร้อมในการเก็บข้อมูล และ จะใช้สัญญาณ NRFD เพื่อชี้ให้เห็นว่าพร้อม หรือไม่พร้อมที่จะรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
032558



NOTE: The timing shown is relative

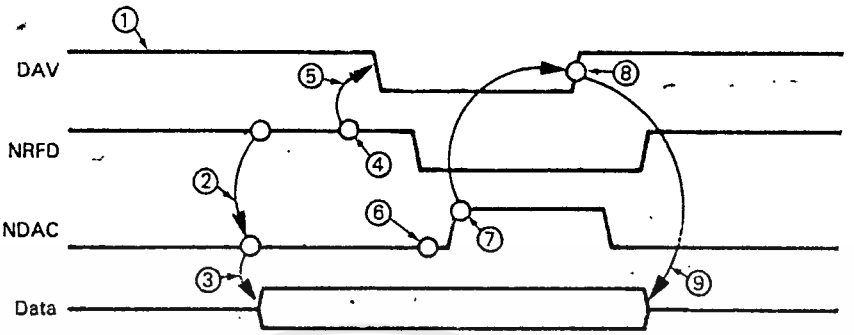
รูปที่ 6

ขบวนการแฮนด์เชคเริ่มขึ้นเมื่อ ตัวส่งมีข้อมูลที่จะถ่ายทอดลงสายสัญญาณไปยังตัวรับ เริ่มด้วยการที่ตัวรับจะทำให้สัญญาณNRFDเป็น "Hi" (มีค่าเป็น 0) นั่นคือเป็นการบอกให้ทราบว่าตัวรับพร้อมที่จะให้ตัวส่งทำการถ่ายข้อมูลลงในสายสัญญาณแล้ว (ในจุดที่ 1). ตัวส่งจึงทำการถ่ายข้อมูลลงในสายสัญญาณDIO1-DIO8 หลังจากทำการรอเวลาเพื่อให้ตัวส่งถ่ายข้อมูลแล้ว ตัวส่งจะทำให้สัญญาณDAVมีลอจิกเป็น "Low" (มีค่าเป็น 1) เพื่อที่จะบอกให้ทราบว่าขณะนี้ได้มีข้อมูลในสายสัญญาณทั้ง 8 เส้นแล้ว (ในจุดที่ 2) ต่อมาเมื่อผู้รับทราบว่ามีข้อมูลอยู่ในบัสก็จะทำให้สัญญาณ NRFD ให้มีค่าเป็น "Low"เพื่อบอกให้ตัวส่งทราบว่ายังไม่พร้อมที่จะให้ตัวส่งทำการส่งข้อมูลชุดต่อไป และ ตัวรับพร้อมที่จะเก็บข้อมูลจากสายสัญญาณ (ในจุดที่ 3) หลังจากตัวรับทำการเก็บข้อมูลจากสายสัญญาณเสร็จแล้วตัวรับจะทำให้สัญญาณ NDAC ให้มีสถานะเป็น "Hi" เพื่อบอกให้ทราบว่าได้ทำการเก็บข้อมูลเสร็จแล้ว (ในจุดที่-4) เมื่อตัวส่งตรวจพบว่าสัญญาณNDAC มีลอจิกเป็น "Hi" ก็จะทำให้สัญญาณDAV ให้มีลอจิกเป็น "Hi" ด้วยเพื่อไม่ให้ตัวรับทำการรับข้อมูลในบัสอีก (ในจุดที่ 5) เมื่อตัวรับทราบว่าสัญญาณDAV มีลอจิกเป็น "Hi" ก็จะทำให้สัญญาณ NDAC ให้เป็น "Low" ทำให้ข้อมูลในบัสถูกกำจัดออกไป หลังจากนั้นก็จะทำให้สัญญาณ NRFD ให้เป็น "Hi" เพื่อบอกให้ทราบว่าพร้อมที่จะรับข้อมูลชุดต่อไปแล้ว

ขบวนการแฮนด์เชคจึงเสร็จสิ้นลง และ จะเริ่มขึ้นใหม่เมื่อมีสัญญาณชุดใหม่เข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขบวนการแฮนด์เชคของตัวควบคุมซึ่งทำหน้าที่เป็นตัวส่ง



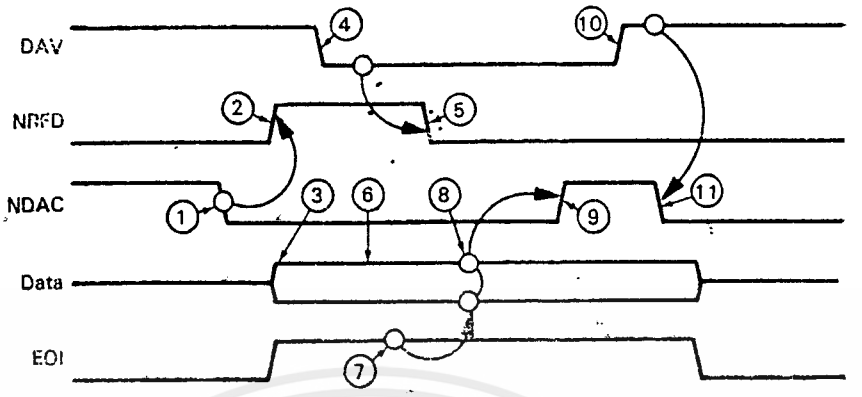
รูปที่ 7

ขบวนการแฮนด์เชค,เริ่มขึ้นโดยการที่ตัวควบคุมทำการเซตค่าให้สัญญาณDAV มีค่าเป็น "Hi" (ในจุดที่ 1) ต่อมาเป็นการตรวจสอบว่าสัญญาณNDAC และ NRFD มีลอจิกเป็น"Hi" ทั้งคู่หรือไม่ (ในจุดที่ 2) ถ้าทั้งคู่เป็น "Hi" นั่นคือเกิดการผิดปกติ คืออุปกรณ์นั้นไม่พร้อมที่จะทำงาน จึงทำการออกจากขบวนการแฮนด์เชค

กลับมาดูในจุดที่ 2 หากสัญญาณใดสัญญาณหนึ่งมีลอจิกเป็น "Low" ตัวควบคุมจะทำการส่งข้อมูลลงใน GPIB Bus (ในจุดที่ 3) หลังจากนั้นตัวควบคุมจะทำการตรวจสอบว่าสัญญาณ NRFD มีลอจิกเป็น "Hi" ใช่หรือไม่ หากไม่ใช่ก็จะทำการตรวจสอบซ้ำอีกครั้งจนกระทั่งสัญญาณ NRFD มีลอจิกเป็น "Hi" (ในจุดที่ 4)หลังจากนั้นตัวควบคุมจะทำให้สัญญาณ DAV ให้มีลอจิกเป็น "Low" เพื่อให้ตัวรับทราบว่ามีข้อมูลอยู่ในบัสข้อมูล (ในจุดที่ 5) ตัวควบคุมจะรอเวลาเพื่อให้ตัวรับทำการเก็บข้อมูลในบัสข้อมูล เมื่อครบกำหนดเวลาตัวควบคุมจะทำการเชคว่าตัวรับทำงานเกินเวลาที่กำหนดหรือไม่หากเกินเวลาก็จะออกจากขบวนการแฮนด์เชคหากไม่เกินเวลาก็จะทำการตรวจสอบต่อไปว่าสัญญาณ NDAC มีลอจิกเป็น "Hi" ใช่หรือไม่ (ในจุดที่ 6) หากไม่ก็จะทำการตรวจสอบอีกครั้งว่าเกินเวลาหรือไม่ กลับไปดูที่จุดที่ 6 หากสัญญาณ NDAC เป็น "Hi" (ในจุดที่ 7)ตัวควบคุมจะทำการตอบสนองโดยการทำสัญญาณDAV ให้มีลอจิกเป็น "Hi" เพื่อบอกให้ตัวรับทราบว่าให้หยุดการเก็บข้อมูลจากบัส (ในจุดที่ 8) หลังจากนั้นข้อมูลในบัสข้อมูลจะถูกนำออกไป (จุดที่ 9) จึงเป็นการเสร็จสิ้นขบวนการแฮนด์เชค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขบวนการแฮนด์เชคของตัวควบคุมซึ่งทำหน้าที่เป็นตัวรับ



รูปที่ 8

ขบวนการแฮนด์เชคเริ่มขึ้น หลังจากการที่ตัวควบคุมรับรู้ว่าตัวส่งจะทำการส่งข้อมูล เมื่อตัวควบคุมทราบว่าตัวส่งจะทำการส่งข้อมูล ตัวควบคุมจะทำให้สัญญาณNDAC มีลอจิกเป็น "Low" เพื่อบอกให้ทราบว่าตัวควบคุมยังไม่ได้ทำการเก็บข้อมูล(ข้อมูลในที่นี้คือ แอดเดรส หรือ ข้อมูลทั่วไปซึ่งตัวส่งทำการส่งมาในบัสข้อมูล) นั่นคือในจุดที่ 1

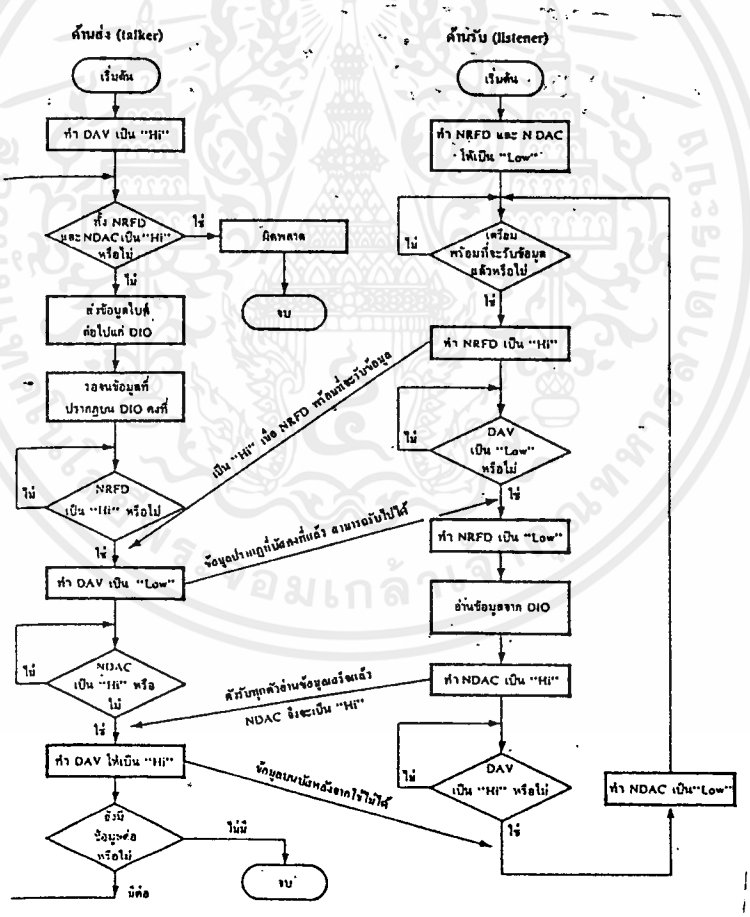
ต่อมาตัวควบคุมจะทำการเซตค่าสัญญาณNRFD ให้เป็น "Hi" เพื่อที่จะให้ตัวส่งทราบว่าขณะนี้ตัวควบคุมพร้อมที่จะรับข้อมูลแล้ว (ในจุดที่ 2) เมื่อสัญญาณ NRFD มีค่าเป็น "Hi" และสัญญาณ NDAC มีค่าเป็น "Low" แล้ว ตัวส่งจะทราบทันทีว่าทำการส่งข้อมูลลงมาในบัสข้อมูลได้แล้ว (ในจุดที่ 3) หลังจากนั้นตัวควบคุมจะทำการรอเวลาให้ตัวส่งทำการส่งข้อมูลลงในบัสข้อมูลให้เสร็จ โดยจะทำการตรวจสอบไปด้วยว่าเกินเวลาที่กำหนดหรือยังหากเกินเวลาที่กำหนดก็จะออกจากขบวนการแฮนด์เชค หากไม่เกินกำหนดเวลาจะทำการตรวจสอบต่อไปว่าตัวส่งทำการเซตค่าให้สัญญาณ DAV เป็น "Low"หรือไม่หากไม่ก็จะตรวจสอบเวลาว่าเกินเวลาที่ต้องรอแล้วหรือยัง หากยังไม่เกินเวลาก็จะทำการเชคต่อไปจนกระทั่งหมดเวลาหรือจนกว่าสัญญาณ DAV จะมีค่าเป็น "Low"

เมื่อสัญญาณ DAV เป็น "Low" แล้ว ตัวควบคุมจะตอบรับโดยการทำให้สัญญาณNRFD มีค่าเป็น "Low" (ในจุดที่ 5) นั่นเป็นการบอกว่าตัวควบคุมพร้อมที่จะเริ่มเก็บข้อมูล (ในจุดที่ 6) หลังจากนั้นตัวควบคุมจะทำการตรวจสอบสัญญาณ EOI เพื่อตรวจสอบว่าข้อมูลที่ตัวส่งทำการส่งลงมานั้นหมดแล้วหรือยัง (ถ้าตัวส่งทำการส่งข้อมูลลงในบัสหมดแล้วจะทำการตั้งค่าของสัญญาณ EOI ให้เป็น "Low") (ในจุดที่ 7) หากสัญญาณ EOI เป็น "Low" ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควบคุมจะเซตสถานะว่าขณะนี้ตัวส่งได้ทำการส่งข้อมูลลงมาในบัสหมดแล้ว หากสัญญาณ EOI เป็น "Hi" ก็ไม่ทำการเซตสถานะ หลังจากนั้นตัวควบคุมจะทำการเก็บข้อมูลในบัส (ในจุดที่ 8) แล้วทำให้สัญญาณ NDAC มีค่าเป็น "Hi"เพื่อบอกให้ทราบว่าตัวควบคุมได้ทำการอ่านข้อมูลเสร็จเรียบร้อยแล้ว (ในจุดที่ 9) เมื่อตัวควบคุมทำให้สัญญาณ NDAC เป็น "Hi"แล้ว ตัวส่งจะลบข้อมูลในบัสข้อมูลโดยการทำให้สัญญาณ DAV มีค่าเป็น "Hi" (ในจุดที่ 10) ซึ่งก่อนหน้านี้ตัวควบคุมจะทำการตรวจสอบอยู่แล้วว่าสัญญาณ DAV เป็น "Hi" หรือยัง

เมื่อสัญญาณ DAV เป็น "Hi"แล้ว ตัวควบคุมจะทำให้สัญญาณ NDAC มีค่าเป็น"Low" (ในจุดที่ 11) แล้วจึงออกจากขบวนการแฮนด์เชค



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งใช้งานของ GPIB

การสั่งการต่างๆเพื่อกำหนดหน้าที่การทำงาน และกำหนดฟังก์ชั่น เช่น กำหนดช่วงการวัด โหมดการวัด หรืออื่นๆแก่เครื่องวัดที่ต่ออยู่ เหล่านี้นั้นตัวควบคุมจะเป็นตัวกำหนดโดยการส่งรหัสคำสั่งไปที่อุปกรณ์ โดยผ่าน DI_1-DI_8 รหัสคำสั่งนี้จะถูกส่งไปในช่วงที่สายสัญญาณ ATN เป็น LOW

คำสั่งสำหรับกำหนดหน้าที่การทำงานต่างๆตามมาตราฐานของ GPIB มีอยู่ด้วยกัน 128 คำสั่งดังแสดงในตารางที่ 1 ต่อไปนี้ โดยแบ่งเป็น 5 กลุ่มคำสั่ง

รหัสที่ใช้ในระบบ GPIB บัสนั้นใช้ร่วมกันทั้งรหัสข้อมูล และรหัสคำสั่ง นั่นคือรหัสเดียวกันมีความหมายได้ 2 อย่าง คือเมื่อ ATN เป็น LOWจะหมายถึงรหัสคำสั่ง แต่ถ้า ATN เป็น HIGHรหัสนี้จะแทนข้อมูลที่เป็น ASCII แทน ซึ่งในตาราง ก็ได้แบ่งความหมายออกเป็น 2 คอลัมน์

ASCII — IEEE 488 BUS MESSAGES (COMMANDS AND ADDRESSES) HEX CODES

MSD / LSD	0		1		2		3		4		5		6		7	
	ASCII	MSG	ASCII	MSG	ASCII	MSG¹	ASCII	MSG¹	ASCII	MSG¹	ASCII	MSG¹	ASCII	MSG	ASCII	MSG
0	NUL		DLE		SP	00	0	16	@	00	P	16	`		p	
1	SOH	GTL	DC1	LLO	!	01	1	17	A	01	Q	17	a		q	
2	STX		DC2		"	02	2	18	B	02	R	18	b		r	
3	ETX		DC3		#	03	3	19	C	03	S	19	c		s	
4	EOT	SDC	DC4	DCL	$	04	4	20	D	04	T	20	d		t	
5	ENQ	PPC	NAK	PPU	%	05	5	21	E	05	U	21	e		u	
6	ACK		SYN		&	06	6	22	F	06	V	22	f		v	
7	BEL		ETB		'	07	7	23	G	07	W	23	g		w	
8	BS	GET	CAN	SPE	(	08	8	24	H	08	X	24	h		x	
9	HT	TCT	EM	SPD	)	09	9	25	I	09	Y	25	i		y	
A	LF		SUB		*	10	:	26	J	10	Z	26	j		z	
B	VT		ESC		+	11	;	27	K	11	[	27	k		{	
C	FF		FS		,	12	<	28	L	12	\	28	l		|	
D	CR		GS		-	13	=	29	M	13	]	29	m		}	
E	SO		RS		.	14	>	30	N	14	^	30	n		~	
F	SI		US		/	15	?	UNL	O	15	_	UNT	o		DEL	

MEANING DEFINED BY PCG CODE

MEANING DEFINED BY PCG CODE

ADDRESSED COMMAND GROUP

UNIVERSAL COMMAND GROUP

LISTEN ADDRESS GROUP

TALK ADDRESS GROUP

SECONDARY COMMAND GROUP

PRIMARY COMMAND GROUP (PCG)

ตารางที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. กลุ่มคำสั่งเจาะจงจุดหมาย (addressed command group) เป็นคำสั่งที่ส่งไปยังอุปกรณ์ที่เป็นตัวส่งหรือตัวรับที่กำหนดไว้ล่วงหน้าแล้ว คำสั่งนี้ประกอบด้วย

GTL (go to local) สั่งให้อุปกรณ์กลับสู่สภาพการควบคุมปกติด้วยมือ

SDC (seleted divice clear) สั่งให้อุปกรณ์เคลียร์ตัวเองสู่สภาพเริ่มต้นใหม่

PPC (paralled poll configure) เป็นคำสั่งสำหรับทำการจัดสายสัญญาณ-ของการทำการจัดสรรสายสัญญาณของกราทำกระบวนการตรวจสอบสภาพอุปกรณ์โดยวิธี-ขนาน โดยใช้กับกลุ่มคำสั่งรอง

GET (group excute trigger)ใช้สั่งเริ่มต้นการทำงานของอุปกรณ์ทีละหลายตัว

TCT (take control) เป็นการกำหนดให้อุปกรณ์ตัวส่งทำหน้าที่เป็นตัวควบคุม

2. กลุ่มคำสั่งครอบคลุม (universal command grop) เป็นคำสั่งที่ส่งไปยังอุป-กรณ์ทุกตัวที่ต่ออยู่ในบัส ประกอบด้วย

LLO (local lockout) เป็นการสั่งให้อุปกรณ์ล็อกอยู่ที่สภาวะควบคุมโดยปุ่มปรับที่หน้าปัดตามปกติ

DCL (device clear) สั่งให้อุปกรณ์ทุกตัวกลับไปสู่สภาวะเริ่มต้น

PPU (parallel poll unconfigure) ใช้ยกเลิกกระบวนการตรวจสอบสภาพแบบขนานทั้งหมด

SPE (serial poll enable) เปลี่ยนโหมดของการตรวจสอบสภาพเป็นแบบอนุ-กรม ในโหมดนี้จะเป็นการส่งสถานะของเครื่องแทนการส่งข้อมูล

SPD (serial poll disable) ยกเลิกโหมดการตรวจสอบแบบอนุกรม

3.กลุ่มคำสั่งกำหนดอุปกรณ์ตัวรับ(listener address group) เป็นคำสั่งสำหรับกำหนดให้อุปกรณ์เป็นตัวรับ ตามรหัสหมายเลขจาก 0-30 และมีคำสั่ง UNL(unlisten) สำหรับยกเลิก

4.กลุ่มคำสั่งกำหนดอุปกรณ์ตัวส่ง(talker address group) สำหรับกำหนดให้อุปกรณ์เป็นตัวส่ง ตามรหัสหมายเลขจาก 0-30 และมีคำสั่ง UNT(untalker)สำหรับ-ยกเลิกเช่นกัน

คำสั่งในกลุ่มที่ 1 ถึง 4 นั้น จัดเป็นกลุ่มคำสั่งหลัก ที่มีความหมายตายตัว ยังมี-คำสั่งอีกกลุ่มที่ขึ้นอยู่กับการกำหนดภายหลังนั่นคือกลุ่มคำสั่งรอง

5.กลุ่มคำสั่งรอง(secondary command group) เป็นคำสั่งที่กำหนดรายละเอียดย่อยของอุปกรณ์แต่ละตัวที่ต่ออยู่ในระบบ ให้มีการทำงานอย่างไร ตามจุดประสงค์ใช้งาน-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของเครื่องมือนั้น เช่นเดียวกับการปรับปุ่มต่างๆด้วยมือนั่นเอง คำสั่งรองนี้จะตามหลังคำสั่งหลัก คือจะใช้หลังจากอุปกรณ์ต่างๆถูกกำหนดวางตัวในระบบเรียบร้อยแล้ว

คำสั่งต่างๆที่กล่าวไป ซึ่งใช้ในการกำหนดสภาวะการทำงานของอุปกรณ์ แต่ละ-สภาวะที่กำหนดไปนั้นเป็นอย่างไร และมีจุดประสงค์เพื่ออะไร ดังต่อไปนี้

Device clear/ Interface Clear

Device clear ใช้ในการทำให้อุปกรณ์ที่ต่ออยู่ในบัสกลับไปสู่สภาวะเริ่มต้น ยังไม่-มีการกำหนดฟังก์ชั่นใดๆ สภาวะเริ่มต้นนี้จะแตกต่างกันไปแล้วแต่ว่าอุปกรณ์นั้นออกแบบไว้-อย่างไร Device clear มีอยู่ 2 ลักษณะ คือ เคลียร์หมดทุกตัวที่ต่ออยู่ (DCL) กับเคลียร์เจาะจงอุปกรณ์ตัวใดตัวหนึ่ง(SDC)

แต่ว่าในการเคลียร์อุปกรณ์ให้อยู่ในสภาวะเริ่มต้นนั้น ไม่ได้หมายความว่าinterface functionของ GPIB จะถูกเคลียร์ให้ไปอยู่ในสภาวะเริ่มต้นด้วยแต่อย่างใด interface function คือสภาพการinterfaceที่ได้กำหนดไว้ในระบบประกอบด้วยฟังก์ชันต่างๆ ดัง-แสดงในตารางที่ 2

ฟังก์ชั่น	สัญลักษณ์	การกลับสู่สภาวะเริ่มต้นโดยIFC
source handshake	SH	ได้
acceptor handshake	AH	ได้
talker or enlarge talker	T or TE	ได้
listener or enlarge listener	L หรือ LT	ได้
service request	SR	ไม่ได้
remote/local	RL	ไม่ได้
parallel poll	pp	ไม่ได้
device clear	DC	ได้
device trigger	DT	ได้
controller	C	ได้

ตารางที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Remote / Local

remote เป็นการกำหนดให้อุปกรณ์ที่อยู่ในระบบเช่น เครื่องมือวัด ให้อยู่ในการควบคุมของอุปกรณ์ของอุปกรณ์ตัวอื่นแทน ซึ่งปุ่มปรับต่างๆบนหน้าปัดเครื่องจะไม่มีผลต่อการทำงาน ส่วนlocal เป็นการควบคุมการทำงานของเครื่องมือวัดด้วยปุ่มปรับบนหน้าปัดตามปกติ

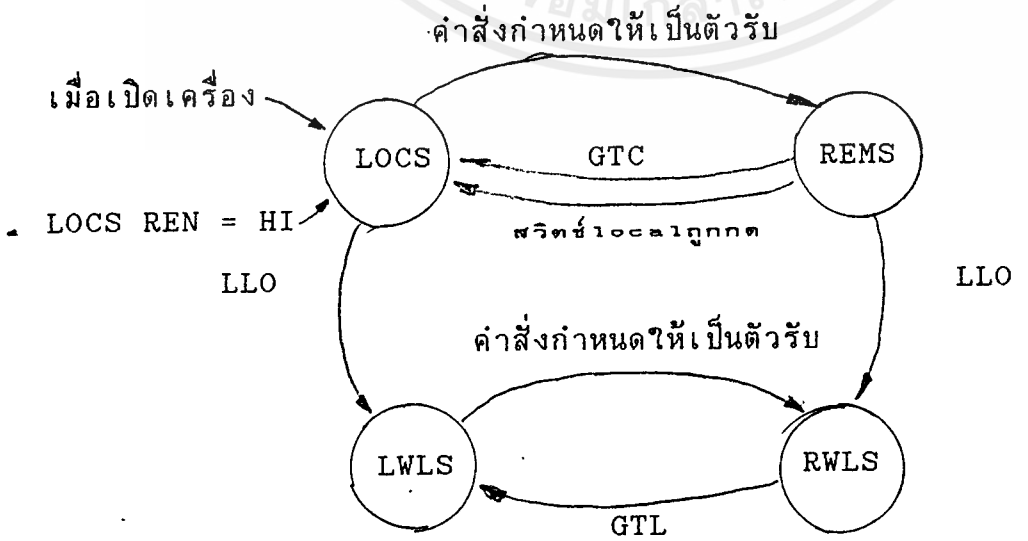
การใช้remoteมีประโยชน์ในแง่ที่ขณะที่ตัวการควบคุมเช่นคอมพิวเตอร์กำลังติดต่ออุปกรณ์ตัวนั้นอยู่ หากไม่มีการตัดการควบคุมโดยปุ่มปรับบนหน้าปัดออก ถ้ามีใครมาปรับแต่งก็จะทำให้การทำงานผิดพลาดไปได้ การทำงานของ GPIB ในremote and local มี 4. ลักษณะดังนี้

1.LOCS ก็คือlocalนั่นเอง เป็นสภาพการควบคุมที่ปุ่มปรับตามปกติ จะอยู่ในสภาพนี้ตอนเปิดเครื่อง หรือ REN เป็น HIGH หรือเมื่อได้รับคำสั่ง GTL

2.REMS คือremote หมายถึงการตัดการควบคุมโดยปุ่มหน้าปัดออก จะเกิดขึ้นเมื่อ REN เป็น LOW และจะถูกล็อกไว้ เว้นแต่ว่าสวิตช์localที่ตัวอุปกรณ์จะถูกเปลี่ยนไปตำแหน่ง local

3.RWLS เป็นสภาพ remote ที่ถูกล็อกเอาไว้เช่นกัน แต่ว่าจะตัดการควบคุมตรงสวิตช์localที่ตัวอุปกรณ์ออกไป สภาพremoteโดย RWLS จึงมีความสำคัญสูงกว่า REMS อย่างไรก็ตามยังถูกยกเลิกได้ด้วยคำสั่ง LLO

4.LWLS มีสภาพเช่นเดียวกับlocal แต่จะแตกต่างกันตรงที่สภาพlocalโดย LWLS นี้ เมื่อได้รับคำสั่งกำหนดอุปกรณ์ตัวรับ จะเปลี่ยนไปอยู่ในสภาพแบบล็อกหรือ RWLS ทันที ในการที่จะมาที่สภาพ LWLS นี้ได้ก็มี 2 กรณีคือ เมื่ออยู่ในสภาพlocalธรรมดา (LOCS) แล้วได้รับคำสั่ง LLO หรืออยู่ใน RWLS แล้ว ได้รับคำสั่ง GTL



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การขอบริการและการตรวจสอบ (Service Request and Polling)

เมื่อตัวควบคุมได้รับ SRQ เป็น LOW จะให้อุปกรณ์ส่งข้อมูลแสดงสถานะการทำงาน ซึ่งมีอยู่ 2 วิธีคือ

1.การตรวจสอบแบบอนุกรม ซึ่งมีขั้นตอนดังนี้

1.1 ATN ถูกดึงเป็น LOW หลังจากได้รับ LOW จากสายสัญญาณ SRQ

1.2 คำสั่ง UNL ถูกสั่งไปยังอุปกรณ์

1.3 ตัวควบคุมจะแจ้งรหัสตัวรับของตน และกำหนดรหัสตัวส่งของอุปกรณ์ที่จะตรวจสอบไปที่บัส

1.4 ตามด้วยคำสั่ง SPE และสาย ATN กลายเป็น HI ซึ่งอุปกรณ์ที่ถูกเรียกจะส่งข้อมูลแสดงสถานะออกมา 1 ไบต์โดยบิตที่ 7 จะเป็นตัวชี้ว่าอุปกรณ์นี้เป็นตัวขอบริการ ถ้าใช่จะเป็น LOW ส่วนบิตอื่นๆก็ใช้บอกข้อมูลอื่นๆซึ่งมิได้กำหนดเฉพาะ

1.5 สาย ATN ถูกดึงเป็นLOWอีกที เพื่อส่งคำสั่งยกเลิกการตรวจสอบคือSPD

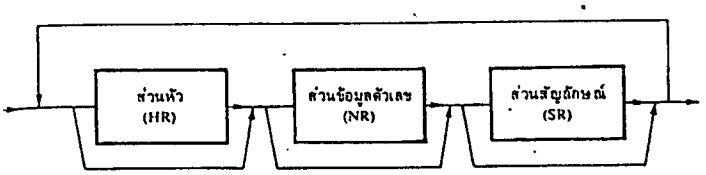
1.6 จากนั้นคำสั่ง UNT ก็ถูกส่งไปยังอุปกรณ์เพื่อยกเลิกการเป็นตัวส่ง

ซึ่งถ้าหาก SQR ยังคงเป็น LOW อยู่ ก็จะมีการตรวจสอบไปยังอุปกรณ์ตัวอื่นๆต่อไปตามขั้นตอนเดิม

2.การตรวจสอบแบบขนาน สามารถทำได้เร็วกว่าแบบอนุกรม ทั้งนี้เพราะสามารถอ่านข้อมูลเพียงไบต์เดียวก็สามารถรู้ได้ทันทีว่าอุปกรณ์ตัวใดเป็นผู้ขอบริการ

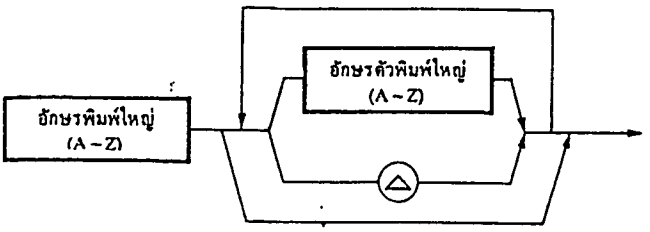
รูปแบบของข้อมูล

โดยทั่วไปข้อมูลจากอุปกรณ์(device message) แบ่งได้ออกเป็น 3 ส่วน ดังแสดงในรูปที่ 9 อันได้แก่ส่วนหัว(HR) ซึ่งจะอยู่ส่วนหน้าสุด เป็นตัวบอกชนิดข้อมูล ส่วนประกอบ HR แสดงในรูปที่ 10 จะเห็นว่าประกอบด้วยตัวอักษรภาษาอังกฤษตัวพิมพ์ใหญ่กับช่องว่างที่เป็นเว้นวรรค(△)ปกติจะมีอักษรประมาณ 1-3 ตัว



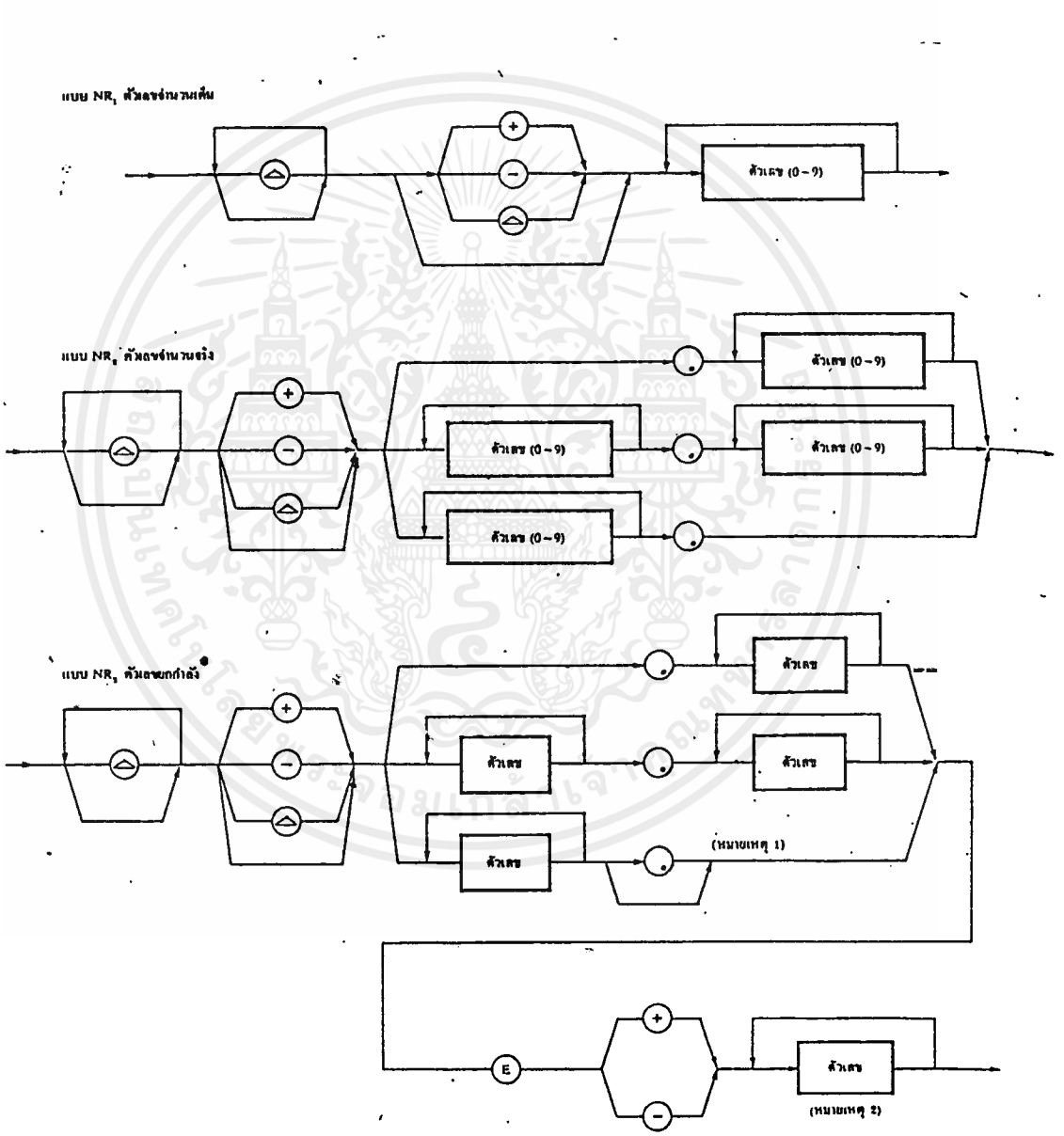
รูปที่ 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 10

ส่วนที่สองคือเนื้อหาข้อมูล (NR) ซึ่งใช้แสดงค่าตัวเลข มีอยู่ 3 แบบ คือ NR_1, NR_2 และ NR_3 ดังแสดงในรูปที่ 11 ส่วนท้ายของ NR ยังอาจมีตัวอักษรแสดงหน่วยตามมา.



หมายเหตุ 1 : จุดทศนิยมนี้สำคัญต้องส่งเสมอไปด้วยทุกครั้ง และตัวโปรต้องถึงจำนวนหน่วยนี้ด้วย

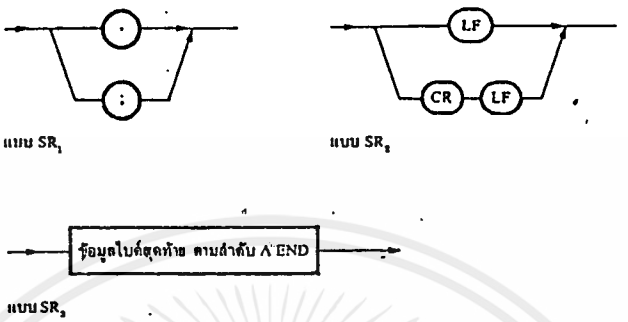
หมายเหตุ 2 : เอนจิ้นจำนวนขนาด 2 หลักเป็นมาตรฐาน แต่สามารถใช้ 1 หลักหรือ 3 หลักก็ได้

รูปที่ 11

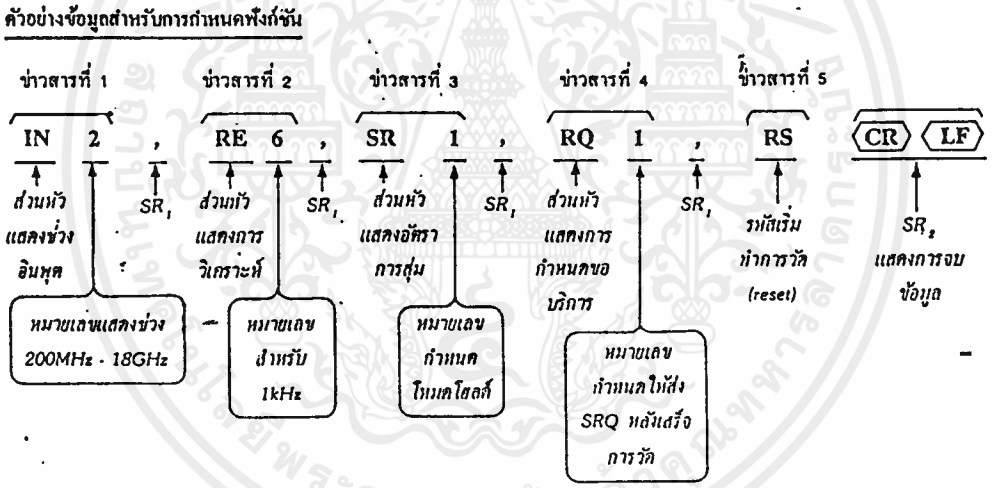
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

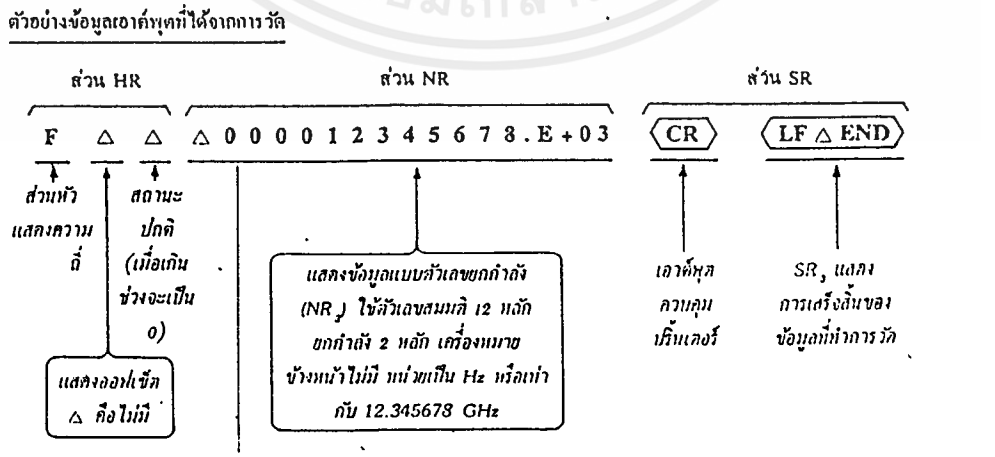
ส่วนที่ 3 คือสัญญาณแบ่งข้อมูลแต่ละชุด(SR) ดังแสดงในรูปที่ 12 โดน SR_1 ใช้-แสดงการต่อเนื่องของข้อมูล (ข้อมูลยังมีต่อ) SR_2 และ SR_3 แสดงการสิ้นสุดของข้อมูล แต่ SR_3 เป็นการบอกการเสร็จสิ้นข้อมูลทั้งหมดจากการวัด

รูปที่ 13 เป็นตัวอย่างข้อมูลสำหรับการกำหนดฟังก์ชั่นให้เครื่องวัดความถี่และข้อมูลความถี่ที่วัดได้



รูปที่ 12





รูปที่ 13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

การ simulate เป็นเครื่องมือวัดใดๆ

ผู้จัดทำได้ทำการออกแบบวงจรซึ่ง simulate เป็นเครื่องมือวัดใดๆอีกตัวหนึ่ง เพื่อทำการสื่อสารส่งข้อมูลกับตัว controller (PC) รวมทั้งเขียนprogramควบคุมส่วนของ MICROPROCESSOR ของตัว simulate ซึ่งเป็นภาษาแอสแซมบลี้ของ Z80 ซึ่งจะอธิบายเป็นขั้นตอนต่างๆ ดังต่อไปนี้

2.1 การออกแบบวงจรที่ simulate เป็นเครื่องมือวัด

จากรูปที่ 2.1 จะสามารถอธิบายการทำงานในส่วนต่างๆ ซึ่งแบ่งออกได้เป็น 2 ส่วนใหญ่ๆ ดังต่อไปนี้

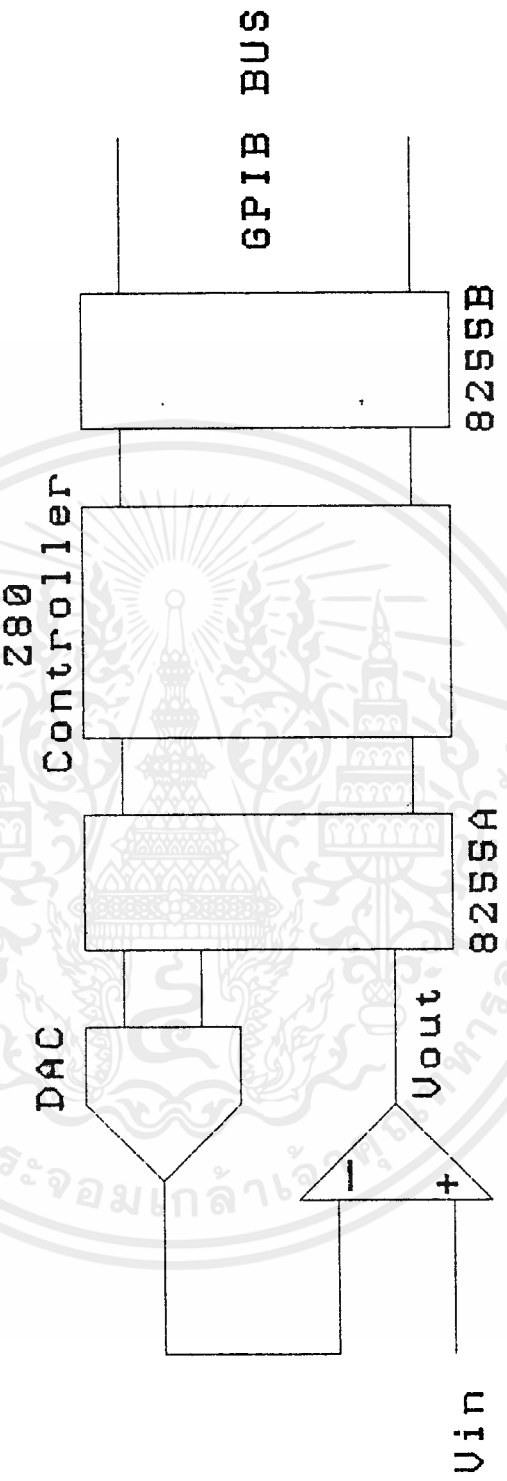
2.1 ส่วนของวงจร MICROPROCESSOR ซึ่งเป็นส่วนที่ควบคุมการรับส่งข้อมูลบน GPIB BUS ,วิเคราะห์ความหมายของข้อมูลที่รับจาก CONTROLLER รวมทั้งการรับข้อมูลจากส่วนของวงจรที่ simulate เป็นตัวเซ็นเซอร์

2.2 ส่วนของวงจรที่ simulate เป็นตัวเซ็นเซอร์ เป็นวงจรที่ส่งข้อมูลที่ได้จากการวัด ให้กับส่วนของวงจร MICROPROCESSOR ซึ่งแบ่งได้ออกเป็น 2 ส่วน

2.2.1 วงจร digital to analog converter เป็นวงจรที่แปลงค่าจาก digital ไปเป็น analog เพื่อนำไปเปรียบเทียบกับค่าที่วัดได้ ซึ่งค่า digital ที่ได้จะเป็นค่าที่ส่งไปยังตัว CONTROLLER

2.2.2 วงจร comparater เป็นวงจรที่เปรียบเทียบค่าระหว่าง ค่าที่แปลงจาก digital เป็น analog กับ ค่าที่วัดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

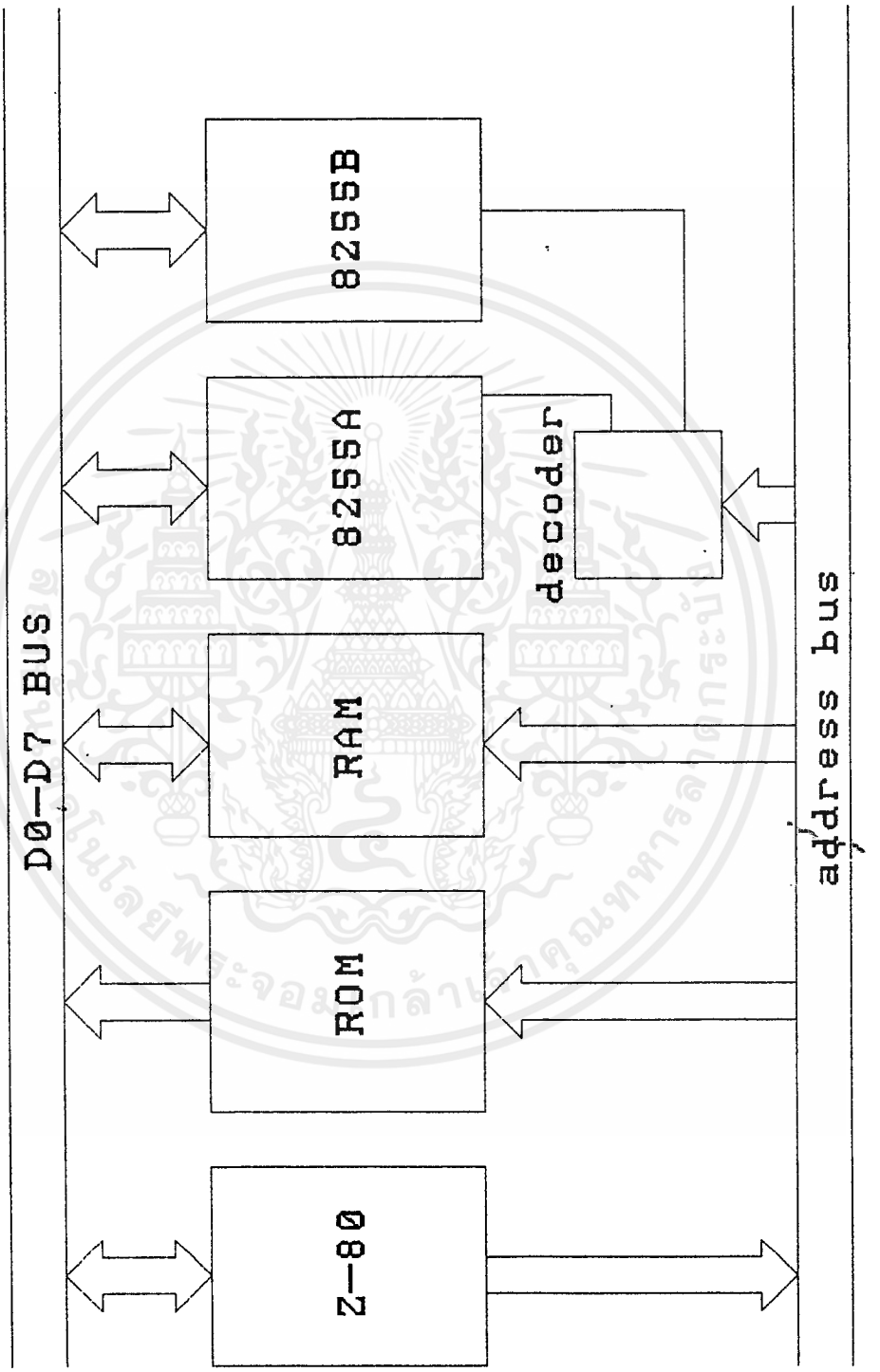


รูปที่ 2.1 บล็อกไดอะแกรมของวงจร simulate เป็นเครื่องมือวัดใดๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 ส่วนของวงจร MICROPROCESSOR

แบ่งได้เป็น 3 ส่วนใหญ่ๆ ตามรูปที่ 2.2 คือ



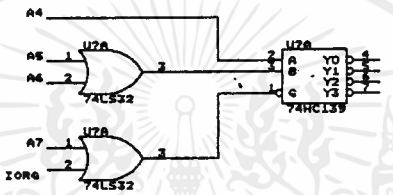
รูปที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1 CPU ซึ่งใช้เบอร์ Z-80

2.1.2 MEMORY ประกอบไปด้วย ROM และ RAM ซึ่ง ROM จะมี adress เริ่มที่ 0000H และ RAM จะมี adress เริ่มที่ 8000H

2.1.3 I/O PORT ได้ใช้ 8255 สองตัวเป็น I/O PORT โดยมีส่วนของวงจร decoder ตามรูปที่ 2.3 ซึ่งในวงจรนี้จะได้ adress ตามตารางที่ 2.1



รูปที่ 2.3

Y0	00 - 03H
Y1	10 - 13H
Y2	20 - 23H
Y3	30 - 33H

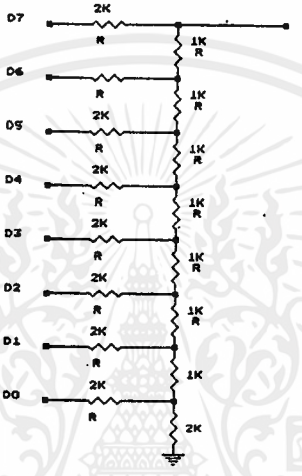
ตารางที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ส่วนของวงจรที่ simulate เป็นตัวเซ็นเซอร์

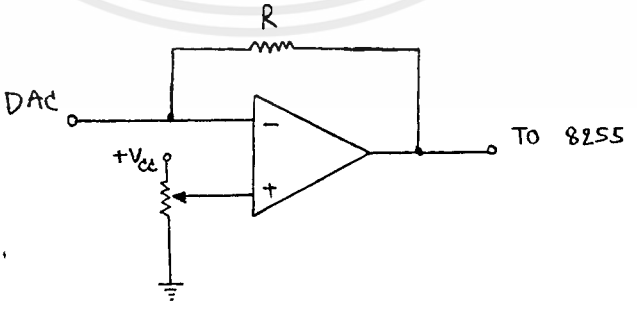
แบ่งเป็น 2 ส่วน ดังต่อไปนี้

2.2.1 วงจร digital to analog converter เป็นวงจรที่แปลงค่า-digital ไปเป็น analog ซึ่งประกอบไปด้วย resister 2 ค่า คือ 2 k-ohm และ 1 k-ohm ต่อตามรูปที่ 2.4 โดยค่า digital ได้มาจาก port-B ของ 8255A สำหรับตัววัดที่ 1 และมาจาก port-C ของ 8255A สำหรับตัววัดที่ 2



รูปที่ 2.4

2.2.2 วงจร comparater เป็นวงจรที่เปรียบเทียบค่าระหว่าง ค่าที่แปลงจาก digital เป็น analog กับ ค่าที่วัดได้ ดังรูปที่ 2.5

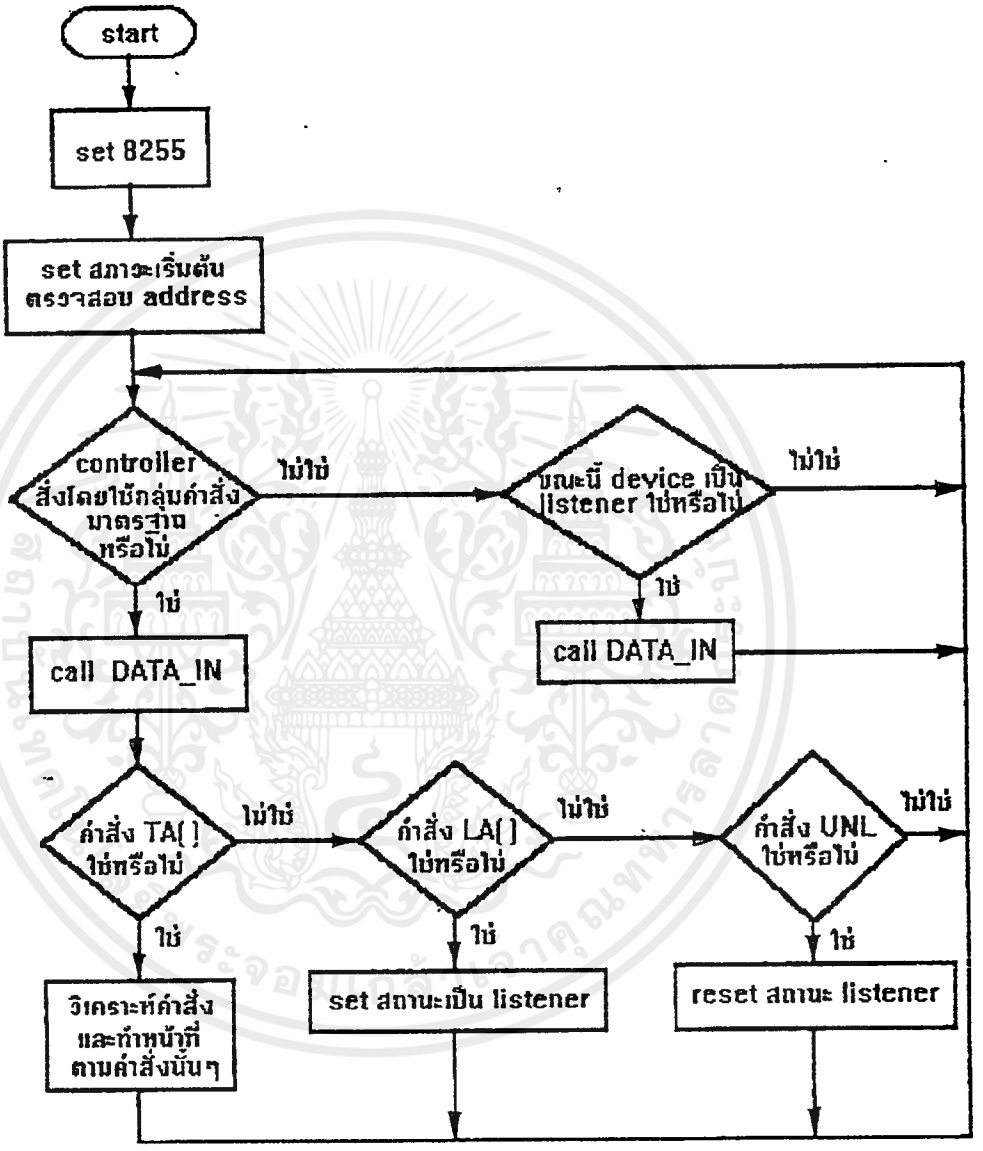


รูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การเขียนโปรแกรมควบคุม MICROPROCESSOR

สามารถเขียนเป็น FLOW CHART ได้ตามรูปที่ 2.6



รูปที่ 2.6 FLOW CHART ของโปรแกรมควบคุม MICROPROCESSOR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมควบคุม MICROCESSOR

```
                CPU     "Z80.TBL"
                HOF     "INT8"
                ORG     0000H
PA1:            EQU     00H
PB1:            EQU     01H
PC1:            EQU     02H
PD1:            EQU     03H
PA2:            EQU     10H
PB2:            EQU     11H
PC2:            EQU     12H
PD2:            EQU     13H
T1:             EQU     41H
T2:             EQU     42H
UNL:            EQU     3FH
                LD      A,10010000B         ; SET 8255A
                OUT     (PD2),A
                LD      A,10010010B         ; SET 8255B
                OUT     (PD1),A
                LD      A,00110111B
                OUT     (PC1),A
                LD      A,00H
                LD      (STATUS),A
                LD      SP,STACK            ; SET STACK
                LD      A,0000B             ;NRFD = 0 NDAC = 0
                OUT     (PD1),A
                ;
                CALL    FINEADD
                ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ATEN:   LD      A,0010B
        OUT     (PD1),A
        IN      A,(PB1)
        BIT     6,A
        JR      Z,COMM              ; ATN = 0 ?
        LD      IX,STATUS
        BIT     0,(IX+0)
        JR      Z,ELO
        CALL    DIN
        LD      IX,STATUS
        RES     0,(IX+0)            ; RESET STATUS
ELO:    JP      ATEN
        ;
COMM:   CALL    DIN
        LD      A,(TA)
        LD      HL,DATA
        CP      (HL)                ; TALKER COMMAND
        JR      NZ,NEXT
        CALL    DOUT
        JP      ATEN
        ;
NEXT:   LD      A,(LA)
        LD      HL,DATA
        CP      (HL)                ; LISTENER COMMAND
        JR      NZ,NEXT1
        LD      A,(STATUS)
        OR      01H                 ;SET STATUS
        LD      (STATUS),A
        JP      ATEN
        ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
NEXT1:      LD      A,UNL
            LD      HL,DATA
            CP      (HL)                ;COMMAND UNLISTENER
            JR      NZ,ATEN
            LD      IX,STATUS
            RES     0,(IX+0)            ; RESET STATUS
            JP      ATEN
            ;
            ;************************
            ; ROUTINE FINED ADDRESS
            ;************************
            ;
FINEADD:    IN      A,(PA2)
            AND     0FH
            ADD     A,20H
            LD      (LA),A
            ADD     A,20H
            LD      (TA),A
            RET
            ;
            ;*********************
            ; ROUTINE DATA IN
            ;*********************
            ;
DIN:        LD      A,1101B             ; SET NRFD = HI
            OUT     (PD1),A
WAIT1:      IN      A,(PB1)
            BIT     1,A                 ; DAV = 0 ?
            JR      NZ,WAIT1
            LD      A,1100B             ; RESET NRFD = LOW
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                OUT         (PD1),A
                LD          A,0100B
                OUT         (PD1),A
                LD          IX,STATUS
                BIT         0,(IX+0)
                JR          NZ,DIN1
                IN          A,(PA1)             ; D1-D8
                XOR         0FFH
                LD          (DATA),A
                JR          DIN2
DIN1:           IN          A,(PA1)
                XOR         0FFH
                LD          (DATA1),A
DIN2:           LD          A,1111B             ; NDAC =HI
                OUT         (PD1),A
WAIT2:          IN          A,(PB1)
                BIT         1,A
                JR          Z,WAIT2             ; DAV = HI ?
                LD          A,1110B             ; NDAC = LOW
                OUT         (PD1),A
                RET
                ;
                ;*********************
                ;ROUTINE DATA OUT
                ;*********************
                ;
DOUT:           LD          A,1111B
                OUT         (PD1),A
                IN          A,(PB1)
                BIT         2,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JR      Z,DOUT1             ; NRFD = 0 ?
                BIT     3,A
                JR      NZ,EXDOUT           ; NDAC = 1 ?
DOUT1:          LD      A,T1
                LD      HL,DATA1
                CP      (HL)
                JR      NZ,DOUT2
                LD      B,00H
DOUT1_1:        LD      A,B
                OUT     (PB2),A
                INC     B
                IN      A,(PA2)             ; THERMO1
                BIT     6,A
                JR      NZ,DOUT1_1
                JP      DOUT4
DOUT2:          LD      A,T2
                LD      HL,DATA1
                CP      (HL)
                JR      NZ,EXDOUT
                LD      B,00H
DOUT2_1:        LD      A,B
                OUT     (PC2),A
                INC     B
                IN      A,(PA2)             ; THERMO2
                BIT     7,A
                JR      NZ,DOUT2_1
DOUT4:          LD      A,0100B
                OUT     (PD1),A
                LD      A,0111B
                OUT     (PD1),A
```

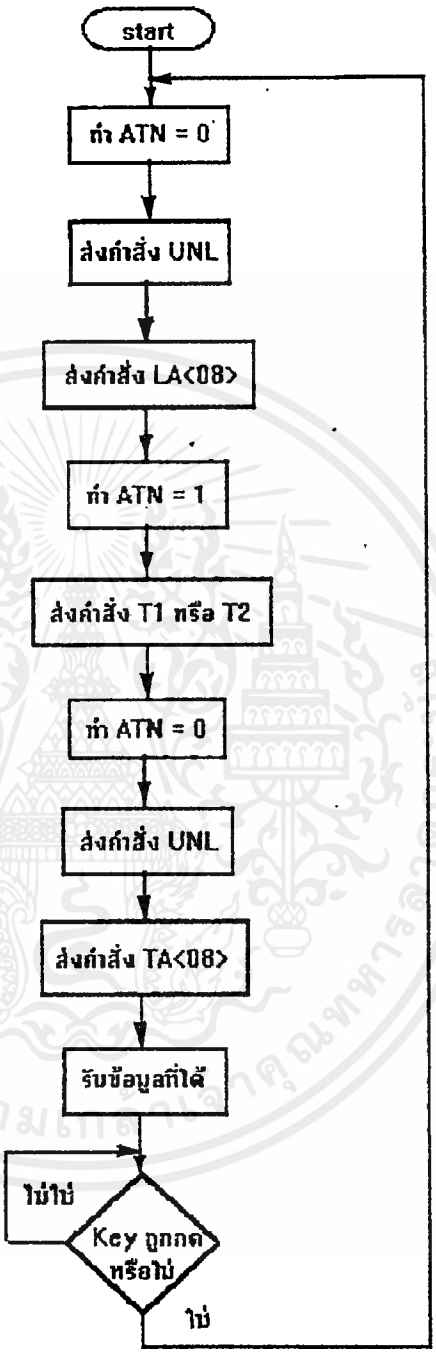
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LD      A,B
                OUT     (PA1),A
                LD      A,1101B
                OUT     (PD1),A
NRFD:           IN      A,(PB1)
                BIT     2,A
                JR      Z,NRFD          ; NRFD = 0 ?
                LD      A,1111B
                OUT     (PD1),A
                LD      A,1010B         ; SET DAV = 0
                OUT     (PD1),A
                LD      A,0000B
                OUT     (PD1),A
NDAC:           IN      A,(PB1)
                BIT     3,A
                JR      Z,NDAC          ; NDAC = 0 ?
                LD      A,1011B         ; RESET DAV =1
                OUT     (PD1),A
EXDOUT:         LD      A,00H
                LD      (DATA),A
                LD      A,1110B
                OUT     (PD1),A
                LD      A,1100B
                OUT     (PD1),A
                RET
                ;
                ORG     8000H
TA:             DFS     1
LA:             DFS     1
DATA:           DFS     1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DATA1:          DFS         1
STATUS:         DFS         1
USTACK:         DFS         20H
STACK:
                END
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 FLOW CHART ของโปรแกรมที่ส่งจาก CONTROLLER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include<graphics.h>
#include<stdio.h>
#include<dos.h>
#define PA 0x300
#define PB 0x301
#define PC 0x302
#define PD 0x303

unsigned char datain,dat,ch;
char cal[4];
main()
{
  while ( (ch != 'q') || (ch!='Q'))
  {
    recdat(190);          /*T1*/
    trans(dat);
    recdat(189);          /*T2*/
    trans(dat);
    ch = getchar();
  }

}
recdat(tt)
int tt;
{
  outportb(PD,04);              /* ATN =0 */
  send(192);                    /* unlistener */
  outportb(PD,04);              /* ATN =0 */
  send(215);                    /* LA<08> */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  outportb(PD,05);              /* ATN =1 */
  send(tt);                     /* T */
  outportb(PD,04);              /* ATN =0 */
  send(192);                    /* unlistener */
  outportb(PD,04);              /* ATN =0 */
  send(183);                    /* TA<08> */
  reci();

}

send(comm)
int comm;
{
   delay(100);
   outportb(PD,139);
   datain = inportb(PC);
   if((datain%4)== '\003')          /* NRFD & NDAC = HI ? */
      exit();
   outportb(PA,comm);
   datain = inportb(PC);
   while (((datain%4) != '\001') || ((datain%4) != '\003'))
      datain = inportb(PC);         /* NRFD = HI ? */
   outportb(PD,137);
   outportb(PB,64);                 /* DAV = LOW */
   datain = inportb(PC);
   while (((datain%4) != '\002') || ((datain%4) != '\003'))
       datain = inportb(PC);        /* NDAC = HI */
   outportb(PB,65);                 /* DAV = LOW */
   outportb(PD,139);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
reci()

{
   outportb(PD,146);
   outportb(PC,64);
   outportb(PD,01);                    /* NRFD = HI */
   datain = inportb(PB);
   while ((datain%2) != '\000')        /* DAV = LOW ? */
      datain = inportb(PB);
   outportb(PD,00);                    /* NRFD = LOW */
   dat = inportb(PA);
   outportb(PD,03);                    /* NDAC = HI */
   datain = inportb(PB);
   while ((datain%2) != '\001')        /* DAV = HI ? */
      datain = inportb(PB);
   outportb(PD,02);                    /* NDAC = LOW */
}
unsigned char bbb;

trans(aaa)
char aaa;
{
   bbb = aaa/100;
   cal[1] = (aaa-(bbb*100))/10+'0';
   cal[2] = aaa%10+'0';
   cal[0] = bbb+'0';
   cal[3] = '\0';
   }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่3
โปรแกรมควบคุมเครื่อง Spectrum Analyzer

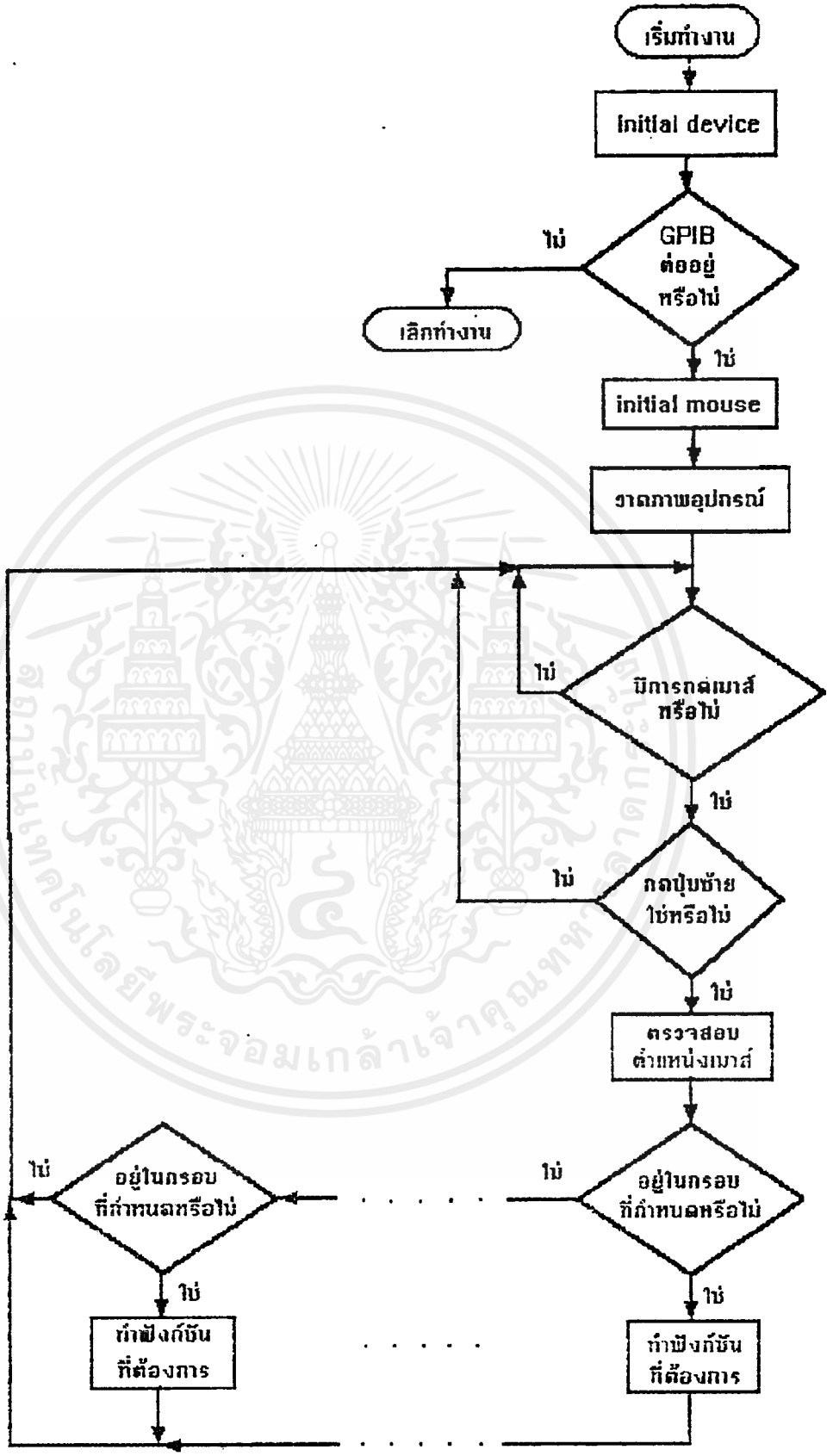
ผู้จัดทำได้ทำการออกแบบโปรแกรมเพื่อสั่งให้เครื่องมือวัดทำงาน โดยทำการส่งคำสั่งผ่าน CARD IEEE-488 ไปยังเครื่องมือวัด ในที่นี้คือเครื่อง SPECTRUM ANALYZER ของ ADVANTEST รุ่น R4131C ซึ่งแผนผังการทำงานเป็นไปตามรูปที่ 3.1 และสามารถอธิบายแผนผังการทำงานได้ดังนี้

3.1 การออกแบบแผนผังการทำงาน

จากรูปที่ 3.1แสดงแผนผังการทำงานของโปรแกรม เราสามารถอธิบายการทำงานได้ดังนี้

เมื่อโปรแกรมเริ่มทำงาน โปรแกรมจะทำการส่งคำสั่งผ่านCARD IEEE-488 โดยในช่วงนี้จะกำหนดตัวแปร และส่งคำสั่งให้ค้นหาอุปกรณ์ที่นำมาเชื่อมต่อ และค้นหาตำแหน่งของ CARD IEEE-488 หากไม่มีอุปกรณ์ หรือ GPIB ต่ออยู่จะแสดงข้อผิดพลาดออกมาทางจอภาพ และออกจากโปรแกรม หากมีอุปกรณ์ และ GPIB ต่ออยู่ จะทำงานขั้นต่อไปโดยการเชื่อมต่อเมาส์เข้ากับเครื่องCOMPUTER เนื่องจากโปรแกรมนี้เขียนมาเพื่อให้ใช้กับเมาส์เท่านั้น เมื่อทำการเชื่อมต่อเมาส์กับเครื่อง COMPUTER เรียบร้อยแล้ว จะทำการวาดภาพของอุปกรณ์ที่นำมาเชื่อมต่อ ในที่นี้ คือ เครื่อง SPECTRUM ANALYZER ของบริษัท ADVANTEST รุ่น R4131C หลังจากนั้นจะทำการตรวจสอบว่ามีการกดสวิตช์ของเมาส์หรือไม่ หากไม่มีการกดสวิตช์ จะกลับไปทำการตรวจสอบการกดสวิตช์ต่อไป แต่หากมีการกดสวิตช์ของเมาส์ จะทำการตรวจสอบต่อไป ว่าสวิตช์ที่กดนั้นเป็นสวิตช์ทางซ้าย ใช่หรือไม่ หากสวิตช์ที่ถูกกดไม่ใช่สวิตช์ทางซ้าย (กดสวิตช์ทางขวา) ก็จะกลับไปทำการตรวจสอบใหม่ว่ามีการกดสวิตช์เมาส์หรือไม่ แต่ถ้าหากสวิตช์ที่ถูกกดนั้นเป็นสวิตช์ทางซ้าย จะทำการตรวจสอบตำแหน่งของเมาส์ หากไม่อยู่ในกรอบที่กำหนด ก็จะทำการตรวจสอบตำแหน่งต่อไปจนครบ หากไม่อยู่ในกรอบที่กำหนดเลย จะกลับไปตรวจสอบการกดสวิตช์ของเมาส์อีก แต่หากอยู่ในกรอบที่กำหนดก็จะทำขั้นตอนต่อไป โดยจะทำงานตามฟังก์ชั่นที่ต้องการ ตัวอย่างเช่น ปรับตำแหน่งจอภาพ หรือหากอยู่ในกรอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดงแผนผังการทำงานของโปรแกรมควบคุม SPECTRUM ANALYZER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ถูกกำหนดให้โปรแกรมเลิกทำงาน จะทำการลบจอภาพและออกจากโปรแกรม

3.2 โปรแกรมควบคุม SPECTRUM ANALYZER

```
#include <graphics.h>
#include "mouse.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "decl.h"
  void gpiberr(char *mes);
  void deverr(char *mes, char *dat);

  int bd,dv;


main() {
  int gd =DETECT, gm = 0 ;
  initgraph(&gd,&gm,"");
  init_dev();
  initialize();
  init_mouse(MOUSE_NEEDED, gd, gm);
  do {
      if(button_press(1)) break;
      if(button_press(0)){
   if(mouse_in_box(1,532,190,560,246)){
     ibwrt (dv,"FD",2L);
     if (ibsta & ERR) gpiberr("ibwrt error");
   }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(mouse_in_box(1,504,190,531,246)){
  ibwrt (dv,"FU",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
if(mouse_in_box(1,364,151,382,167)){
  bp(364,151,382,167);
  ibwrt (dv,"LC",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
if(mouse_in_box(1,364,182,382,198)){
  bp(364,182,382,198);
  ibwrt (dv,"WR",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
  if(mouse_in_box(1,364,216,382,232)){
  bp(364,216,382,232);
  ibwrt (dv,"SE",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
  if(mouse_in_box(1,364,251,382,267)){
  bp(364,251,382,267);
  ibwrt (dv,"VW",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
if(mouse_in_box(1,364,280,382,296)){
  bp(364,280,382,296);
  ibwrt (dv,"RC",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(mouse_in_box(1,364,309,382,325)){
  bp(364,309,382,325);
  ibwrt (dv,"SH",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
if(mouse_in_box(1,406,309,424,325)){
  bp(406,309,424,325);
  ibwrt (dv,"FC",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
if(mouse_in_box(1,454,309,472,325)){
  bp(454,309,472,325);
  ibwrt (dv,"UN",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
if(mouse_in_box(1,498,161,516,177)){
  bp(498,161,516,177);
  ibwrt (dv,"CF",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
if(mouse_in_box(1,522,161,540,177)){
  bp(522,161,540,177);
  ibwrt (dv,"M1",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
}
if(mouse_in_box(1,546,161,564,177)){
  bp(546,161,564,177);
  ibwrt (dv,"M0",2L);
  if (ibsta & ERR) gpiberr("ibwrt error");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    if(mouse_in_box(1,498,262,516,278)){
      bp(498,262,516,278);
      ibwrt (dv,"FL",2L);
      if (ibsta & ERR) gpiberr("ibwrt error");
    }
    if(mouse_in_box(1,522,262,540,278)){
      bp(522,262,540,278);
      ibwrt (dv,"M4",2L);
      if (ibsta & ERR) gpiberr("ibwrt error");
    }
    if(mouse_in_box(1,546,262,564,278)){
      bp(546,262,564,278);
      ibwrt (dv,"M3",2L);
      if (ibsta & ERR) gpiberr("ibwrt error");
    }
    if(mouse_in_box(1,347,375,365,391)){
      bp(347,375,365,391);
      ibwrt (dv,"SR",2L);
      if (ibsta & ERR) gpiberr("ibwrt error");
    }
    if(mouse_in_box(1,405,385,423,401)){
      bp(405,385,423,401);
      ibwrt (dv,"TR",2L);
      if (ibsta & ERR) gpiberr("ibwrt error");
    }
    if(mouse_in_box(1,435,362,453,378)){
      bp1(435,362,453,378,
  8,11,7,9,7,9,10,7,13,9,12,9,12,11,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
7,10,6,8,6,8,9,6,12,8,11,8,11,10);
    ibwrt (dv,"TU",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,435,385,453,401)){
    bp1(435,385,453,401,
8,7,8,9,7,9,10,11,13,9,12,9,12,7,
7,6,7,8,6,8,9,10,12,8,11,8,11,6);
    ibwrt (dv,"TD",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,466,362,484,378)){
    bp1(466,362,484,378,
8,11,7,9,7,9,10,7,13,9,12,9,12,11,
7,10,6,8,6,8,9,6,12,8,11,8,11,10);
    ibwrt (dv,"VU",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,466,385,484,401)){
    bp1(466,385,484,401,
8,7,8,9,7,9,10,11,13,9,12,9,12,7,
7,6,7,8,6,8,9,10,12,8,11,8,11,6);
    ibwrt (dv,"VD",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,504,306,522,322)){
    bp1(504,306,522,322,
8,7,8,9,7,9,10,11,13,9,12,9,12,7,
7,6,7,8,6,8,9,10,12,8,11,8,11,6);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    ibwrt (dv,"AD",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,542,306,560,322)){
    bp1(542,306,560,322,
8,11,7,9,7,9,10,7,13,9,12,9,12,11,
7,10,6,8,6,8,9,6,12,8,11,8,11,10);
    ibwrt (dv,"AU",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,394,257,436,277)){
    bp1(394,257,436,277,
19,8,19,12,16,12,22,15,28,12,25,12,25,8,
18,7,18,11,15,11,21,14,27,11,24,11,24,7);
    ibwrt (dv,"LD",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,442,257,484,277)){
    bp1(442,257,484,277,
19,15,19,11,16,11,22,8,28,11,25,11,25,15,
18,14,18,10,15,10,21,7,27,10,24,10,24,14);
    ibwrt (dv,"LU",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,394,202,436,222)){
    bp3(394,202,436,222,
13,8,17,8,17,5,20,11,17,17,17,14,13,14,
12,7,16,7,16,4,19,10,16,16,16,13,12,13,
31,8,27,8,27,5,24,11,27,17,27,14,31,14,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
30,7,26,7,26,4,23,10,26,16,26,13,30,13);
    ibwrt (dv,"NR",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,442,202,484,222)){
    bp3(442,202,484,222,
20,8,16,8,16,5,13,11,16,17,16,14,20,14,
19,7,15,7,15,4,12,10,15,16,15,13,19,13,
24,8,28,8,28,5,31,11,28,17,28,14,24,14,
23,7,27,7,27,4,30,10,27,16,27,13,23,13);
    ibwrt (dv,"WD",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,397,161,421,177)){
    bp2(397,161,421,177,7,4,8,5,"RBW");
    ibwrt (dv,"RB",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,428,161,452,177)){
    bp2(428,161,452,177,5,4,6,5,"AUTO");
    ibwrt (dv,"BA",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
  if(mouse_in_box(1,458,161,482,177)){
    bp2(458,161,482,177,5,4,6,5,"SPAN");
    ibwrt (dv,"SP",2L);
    if (ibsta & ERR) gpiberr("ibwrt error");
  }
   }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      }while(1);
   mouse_reset();
   closegraph();
   ibonl (bd,0);
   clrscr();
}
button_p()
{
   struct viewporttype vp;
   getviewsettings(&vp);

   setcolor(DARKGRAY);
   line(2,2,(vp.right-vp.left)-2,2);
   line(2,(vp.bottom-vp.top)-2,2,2);

   setcolor(LIGHTGRAY);
   line((vp.right-vp.left)-2,2,(vp.right-vp.left)-2,
(vp.bottom-vp.top)-2);
   line((vp.right-vp.left)-2,(vp.bottom-vp.top)-2,2,
(vp.bottom-vp.top)-2);
}
bp(int q, int w, int e, int r)
{
mouse_off(1);
   setviewport(q,w,e,r,1);
delay(50);
button_p();
delay(50);
button();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mouse_on(1);
}
bp1(int q, int w, int e, int r,
    int a, int b, int c, int d, int F, int f, int g,
 int h, int i, int j, int k
   ,int l, int m, int n, int o, int p, int s, int t,
 int u, int v, int x, int y
   , int z, int A, int B, int C, int D, int E)
{
mouse_off(1);
setviewport(q,w,e,r,1);
delay(50);
button_p();
setcolor(LIGHTGRAY);
arrow(o,p,s,t,u,v,x,y,z,A,B,C,D,E);
setcolor(WHITE);
arrow(a,b,c,d,F,f,g,h,i,j,k,l,m,n);
delay(50);
button();
setcolor(LIGHTGRAY);
arrow(a,b,c,d,F,f,g,h,i,j,k,l,m,n);
setcolor(WHITE);
arrow(o,p,s,t,u,v,x,y,z,A,B,C,D,E);
mouse_on(1);
}
bp2(int q1, int w1, int e1, int r1, int a1, int b1,
 int c1, int d1,char *f1)

{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mouse_off(1);
setviewport(q1,w1,e1,r1,1);
delay(50);
button_p();
setusercharsize(3,4,3,4);
settextstyle(SMALL_FONT,HORIZ_DIR,USER_CHAR_SIZE);
setcolor(LIGHTGRAY);
outtextxy(a1,b1,f1);
setcolor(BROWN);
outtextxy(c1,d1,f1);
delay(50);
button();
setcolor(LIGHTGRAY);
outtextxy(c1,d1,f1);
setcolor(BROWN);
outtextxy(a1,b1,f1);
mouse_on(1);
}
bp3(int Q, int W, int E, int R,
    int a2, int b2, int c2, int d2, int e2,
int f2, int g2, int h2, int i2, int j2,
int k2, int l2, int m2, int n2,
    int a3, int b3, int c3, int d3, int e3,
int f3, int g3, int h3, int i3,
int j3, int k3, int l3, int m3, int n3,
    int a4, int b4, int c4, int d4, int e4,
int f4, int g4, int h4, int i4,
int j4, int k4, int l4, int m4, int n4,
    int a5, int b5, int c5, int d5, int e5,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int f5, int g5, int h5, int i5,
int j5, int k5, int l5, int m5, int n5)
{
mouse_off(1);
setviewport(Q,W,E,R,1);
delay(50);
button_p();
setcolor(LIGHTGRAY);
arrow(a3,b3,c3,d3,e3,f3,g3,h3,i3,j3,k3,l3,m3,n3);
arrow(a5,b5,c5,d5,e5,f5,g5,h5,i5,j5,k5,l5,m5,n5);
setcolor(WHITE);
arrow(a2,b2,c2,d2,e2,f2,g2,h2,i2,j2,k2,l2,m2,n2);
arrow(a4,b4,c4,d4,e4,f4,g4,h4,i4,j4,k4,l4,m4,n4);
delay(50);
button();
setcolor(LIGHTGRAY);
arrow(a2,b2,c2,d2,e2,f2,g2,h2,i2,j2,k2,l2,m2,n2);
arrow(a4,b4,c4,d4,e4,f4,g4,h4,i4,j4,k4,l4,m4,n4);
setcolor(WHITE);
arrow(a3,b3,c3,d3,e3,f3,g3,h3,i3,j3,k3,l3,m3,n3);
arrow(a5,b5,c5,d5,e5,f5,g5,h5,i5,j5,k5,l5,m5,n5);
mouse_on(1);
}
init_dev()
{
  clrscr();
  system("cls");

  dv = ibfind ("DEVA");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if (dv < 1) gpiberr("ibfind error");

  bd= ibfind ("GPIB0");
  if (bd < 0) gpiberr("ibfind error");

  ibsic (bd);
  if (ibsta & ERR) gpiberr("ibsic error");

  ibsre (bd,dv);
  if (ibsta & ERR) gpiberr("ibsre error");

  ibcmd (bd,"\021!\024@",4L);
  if (ibsta & ERR) gpiberr("ibcmd error");
 }

void gpiberr(char *mes)
{
  mouse_reset();
  closegraph();
  printf ("%s\n",mes);

  printf ("ibsta = &H%x  <",ibsta);
  if (ibsta & ERR)    printf ("ERR");
  if (ibsta & TIMO)   printf ("TIMO");
  if (ibsta & END)    printf ("END");
  if (ibsta & SRQI)   printf ("SRQI");
  if (ibsta & RQS)    printf ("RQS");
  if (ibsta & SPOLL)  printf ("SPOLL");
  if (ibsta & EVENT)  printf ("EVENT");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (ibsta & CMPL)   printf ("CMPL");
if (ibsta & LOK)    printf ("LOK");
if (ibsta & REM)    printf ("REM");
if (ibsta & CIC)    printf ("CIC");
if (ibsta & ATN)    printf ("ATN");
if (ibsta & TACS)   printf ("TACS");
if (ibsta & LACS)   printf ("LACS");
if (ibsta & DTAS)   printf ("DTAS");
if (ibsta & DCAS)   printf ("DCAS");
printf (" >\n");

printf ("iberr = %d", iberr);
if (iberr == EDVR) printf (" EDVR <dos error>\n");
if (iberr == ECIC) printf (" ECIC <not cic>\n");
if (iberr == ENOL) printf (" ENOL <no listener>\n");
if (iberr == EADR) printf (" EADR <address error>\n");
if (iberr == EARG) printf (" EARG <invalid argument>\n");
if (iberr == ESAC) printf (" ESAC <not sys ctrlr>\n");
if (iberr == EABO) printf (" EABO <op. aborted>\n");
if (iberr == ENEB) printf (" ENEB <no gpib board>\n");
if (iberr == EOIP) printf (" EOIP <async i/o in prg>\n");
if (iberr == ECAP) printf (" ECAP <no capability>\n");
if (iberr == EFSO) printf (" EFSO <file sys. error>\n");
if (iberr == EBUS) printf (" EBUS <command error>\n");
if (iberr == ESTB) printf (" ESTB <status byte lost>\n");
if (iberr == ESRQ) printf (" ESRQ <srq stuck on>\n");
if (iberr == ETAB) printf (" ETAB <table overflow>\n");

printf ("ibcnt = %d\n", ibcnt);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (ibsta & CMPL)   printf ("CMPL");
if (ibsta & LOK)    printf ("LOK");
if (ibsta & REM)    printf ("REM");
if (ibsta & CIC)    printf ("CIC");
if (ibsta & ATN)    printf ("ATN");
if (ibsta & TACS)   printf ("TACS");
if (ibsta & LACS)   printf ("LACS");
if (ibsta & DTAS)   printf ("DTAS");
if (ibsta & DCAS)   printf ("DCAS");
printf (" >\n");

printf ("iberr = %d", iberr);
if (iberr == EDVR) printf (" EDVR <dos error>\n");
if (iberr == ECIC) printf (" ECIC <not cic>\n");
if (iberr == ENOL) printf (" ENOL <no listener>\n");
if (iberr == EADR) printf (" EADR <address error>\n");
if (iberr == EARG) printf (" EARG <invalid argument>\n");
if (iberr == ESAC) printf (" ESAC <not sys ctrlr>\n");
if (iberr == EABO) printf (" EABO <op. aborted>\n");
if (iberr == ENEB) printf (" ENEB <no gpib board>\n");
if (iberr == EOIP) printf (" EOIP <async i/o in prg>\n");
if (iberr == ECAP) printf (" ECAP <no capability>\n");
if (iberr == EFSO) printf (" EFSO <file sys. error>\n");
if (iberr == EBUS) printf (" EBUS <command error>\n");
if (iberr == ESTB) printf (" ESTB <status byte lost>\n");
if (iberr == ESRQ) printf (" ESRQ <srq stuck on>\n");
if (iberr == ETAB) printf (" ETAB <table overflow>\n");

printf ("ibcnt = %d\n", ibcnt);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
printf("\n");
ibonl(bd,0);
exit(1);
}
```

3.3 ส่วนของโปรแกรมแสดงภาพหน้าจอ

```
#include <graphics.h>
#include <conio.h>
int graphdriver;
int graphmode;
int maxX,maxY,minX,minY;


initialize()
{
struct viewporttype vp;
 graphdriver = DETECT;
 graphmode = VGA;
 initgraph(&graphdriver,&graphmode,"");
 minX = 0;
 minY = 0;
 maxX = getmaxx();
 maxY = getmaxy();

 setcolor(WHITE);
 line(minX+50,minY+115,maxX-50,minY+115);
 line(minX+35,minY+130,minX+35,maxY-50);
 line(maxX-35,minY+130,maxX-35,maxY-50);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
line(minX+50,maxY-35,maxX-50,maxY-35);     /* MAKE BORDER */
arc(maxX-50,minY+130,0,90,15);
arc(minX+50,minY+130,90,180,15);
arc(minX+50,maxY-50,180,270,15);
arc(maxX-50,maxY-50,270,360,15);

setfillstyle(SOLID_FILL,LIGHTBLUE);    /* FILL BACKGROUND */
floodfill(1,1,4);

setcolor(WHITE);
line(minX+50,minY+115,maxX-50,minY+115);
line(minX+35,minY+130,minX+35,maxY-50);
line(maxX-35,minY+130,maxX-35,maxY-50);
line(minX+50,maxY-35,maxX-50,maxY-35); /* MAKE BORDER    */
arc(maxX-50,minY+130,0,90,15);
arc(minX+50,minY+130,90,180,15);
arc(minX+50,maxY-50,180,270,15);
arc(maxX-50,maxY-50,270,360,15);

setfillstyle(SOLID_FILL,LIGHTGRAY);
bar(minX+51,minY+116,maxX-51,maxY-36);
bar(minX+36,minY+130,minX+50,maxY-50);   /*      FILL     */
bar(maxX-51,minY+130,maxX-36,maxY-50);   /*  BACKGROUND */
setcolor(LIGHTGRAY);                     /*      PANEL    */
pieslice(maxX-50,minY+130,0,90,14);
pieslice(minX+50,minY+130,90,180,14);
pieslice(minX+50,maxY-50,180,270,14);
pieslice(maxX-50,maxY-50,270,360,14);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
setcolor(WHITE);
setfillstyle(SOLID_FILL,WHITE);
bar(minX+35,minY+140,minX+43,maxY-60);
bar(maxX-43,minY+140,maxX-35,maxY-60);
setfillstyle(SOLID_FILL,LIGHTGRAY);
bar(minX+38,minY+150,minX+43,maxY-70);
bar(maxX-43,minY+150,maxX-38,maxY-70);    /* MAKE HANDLE */
line(maxX-43,minY+140,maxX-43,maxY-60);   /*     LEFT    */
setcolor(DARKGRAY);                       /*      &      */
line(minX+43,minY+140,minX+43,maxY-60);   /*     RIGHT    */
line(minX+42,minY+150,minX+42,maxY-60);
line(maxX-36,minY+144,maxX-36,maxY-60);
line(maxX-37,minY+147,maxX-37,maxY-60);
line(maxX-38,minY+150,maxX-38,maxY-60);
line(maxX-39,maxY-70,maxX-39,maxY-60);
line(maxX-40,maxY-70,maxX-40,maxY-60);
line(maxX-41,maxY-70,maxX-41,maxY-60);
line(maxX-42,maxY-70,maxX-42,maxY-60);
line(maxX-43,maxY-70,maxX-43,maxY-60);

line(minX+41,maxY-70,minX+41,maxY-60);
line(minX+40,maxY-70,minX+40,maxY-60);
line(minX+39,maxY-70,minX+39,maxY-60);
line(minX+38,maxY-70,minX+38,maxY-60);
line(minX+37,maxY-70,minX+37,maxY-60);
line(minX+36,maxY-65,minX+36,maxY-60);
line(minX+35,maxY-60,minX+35,maxY-60);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
setcolor(WHITE);
line(minX+55,minY+115,minX+55,maxY-35);
line(minX+56,minY+116,minX+56,maxY-36);
line(minX+57,minY+117,minX+57,maxY-37);
line(minX+55,minY+115,maxX-55,minY+115);
line(minX+56,minY+116,maxX-56,minY+116);  /* MAKE BORDER */
line(minX+57,minY+117,maxX-57,minY+117);  /*    OF PANEL  */
setcolor(DARKGRAY);
line(maxX-55,minY+116,maxX-55,maxY-36);
line(maxX-56,minY+116,maxX-56,maxY-36);
line(maxX-57,minY+117,maxX-57,maxY-37);
line(minX+55,maxY-35,maxX-55,maxY-35);
line(minX+56,maxY-36,maxX-56,maxY-36);
line(minX+57,maxY-37,maxX-57,maxY-37);
setcolor(WHITE);              /*  SET MONITOR'S BORDER */
setviewport(minX+67,minY+127,(maxX/2)+10,minY+347,1);
getviewsettings(&vp);
line(3,0,(vp.right-vp.left)-3,0);
line(0,3,0,((vp.bottom-vp.top)-3));
arc(3,3,90,180,3);
arc(3,((vp.bottom-vp.top)-3),180,270,3);
setcolor(DARKGRAY);
line(3,vp.bottom-vp.top,(vp.right-vp.left)-3,
vp.bottom-vp.top);
line((vp.right-vp.left),3,(vp.right-vp.left),
((vp.bottom-vp.top)-3));
arc(((vp.right-vp.left)-3),3,0,90,3);
arc(((vp.right-vp.left)-3),((vp.bottom-vp.top)-3),
270,360,3);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
setcolor(DARKGRAY);
line(15,12,((vp.right-vp.left)-15),12);
line(12,15,12,((vp.bottom-vp.top)-15));
arc(((vp.right-vp.left)-15),15,0,90,3);
arc(15,15,90,180,3);

line(((vp.right-vp.left)-14),14,((vp.right-vp.left)-22),26);
line(14,((vp.bottom-vp.top)-14),22,((vp.bottom-vp.top)-26));
line(14,14,22,26);
setcolor(WHITE);
line(((vp.right-vp.left)-14),((vp.bottom-vp.top)-14),
((vp.right-vp.left)-22),((vp.bottom-vp.top)-26));

setcolor(WHITE);
line(((vp.right-vp.left)-12),15,((vp.right-vp.left)-12),
((vp.bottom-vp.top)-15));
line(15,((vp.bottom-vp.top)-12),((vp.right-vp.left)-15),
((vp.bottom-vp.top)-12));
arc(((vp.right-vp.left)-15),((vp.bottom-vp.top)-15),
270,360,3);
arc(15,((vp.bottom-vp.top)-15),180,270,3);
/* setfillstyle(SOLID_FILL,DARKGRAY);
floodfill(18,18,8);*/
setviewport(minX+89,minY+154,(maxX/2)-12,minY+320,1);
clearviewport();

setviewport(minX,minY,maxX,maxY,1);
setcolor(DARKGRAY);
arc((maxX/2)+25,minY+145,90,180,3);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
line((maxX/2)+22,minY+170,maxX-253,minY+170);
line(maxX-253,minY+170,maxX-253,minY+142);       /*TOP LEFT*/
line(maxX-253,minY+142,(maxX/2)+25,minY+142);
line((maxX/2)+22,minY+145,(maxX/2)+22,minY+170);

line((maxX/2)+22,minY+270,maxX-253,minY+270);
line(maxX-253,minY+270,maxX-253,minY+173);
line(maxX-253,minY+173,(maxX/2)+22,minY+173);  /*MED LEFT*/
line((maxX/2)+22,minY+173,(maxX/2)+22,minY+270);

arc((maxX/2)+25,minY+329,180,270,3);
line((maxX/2)+25,minY+332,maxX-253,minY+332);
line(maxX-253,minY+332,maxX-253,minY+273);  /*BOTTOM LEFT*/
line(maxX-253,minY+273,(maxX/2)+22,minY+273);
line((maxX/2)+22,minY+273,(maxX/2)+22,minY+332);

line(maxX-151,minY+142,maxX-250,minY+142);
line(maxX-250,minY+142,maxX-250,minY+236);  /*TOP CENTER*/
line(maxX-250,minY+236,maxX-151,minY+236);
line(maxX-151,minY+236,maxX-151,minY+142);

line(maxX-151,minY+332,maxX-151,minY+239);
line(maxX-151,minY+239,maxX-250,minY+239);
line(maxX-250,minY+239,maxX-250,minY+332);/*BOTTOM CENTER*/
line(maxX-250,minY+332,maxX-151,minY+332);

arc(maxX-71,minY+145,0,90,3);
line(maxX-148,minY+289,maxX-68,minY+289);
line(maxX-68,minY+289,maxX-68,minY+145);    /*TOP RIGHT*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
line(maxX-71,minY+142,maxX-148,minY+142);
line(maxX-148,minY+142,maxX-148,minY+289);

arc(maxX-71,minY+329,270,360,3);
line(maxX-71,minY+332,maxX-148,minY+332);
line(maxX-148,minY+332,maxX-148,minY+292);/*BOTTOM RIGHT*/
line(maxX-148,minY+292,maxX-68,minY+292);
line(maxX-68,minY+292,maxX-68,minY+329);

arc(maxX-71,minY+349,0,90,3);
line(maxX-68,minY+349,maxX-68,minY+415);
line(maxX-71,minY+346,maxX-145,minY+346);
arc(maxX-145,minY+349,90,180,3);
line(maxX-148,minY+349,maxX-148,minY+415);      /*RIGHT*/
arc(maxX-145,minY+415,180,270,3);
line(maxX-145,minY+418,maxX-71,minY+418);
arc(maxX-71,minY+415,270,360,3);

setfillstyle(SOLID_FILL,WHITE);
fillellipse(maxX-115,minY+390,15,15);
circle(maxX-115,minY+390,13);
circle(maxX-115,minY+390,10);
circle(maxX-115,minY+390,4);
circle(maxX-115,minY+390,2);
settextstyle(SMALL_FONT,HORIZ_DIR,3);
setfillstyle(SOLID_FILL,LIGHTGRAY);      /*INPUT 50OHM*/
bar(maxX-136,minY+342,maxX-97,minY+350);
setcolor(BLACK);
settextstyle(SMALL_FONT,HORIZ_DIR,2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
outtextxy(maxX-133,minY+342,"INPUT 50O");
outtextxy(maxX-136,minY+348,"1GHz-3.5GHz");
outtextxy(maxX-135,minY+356,"+20dbm MAX");
outtextxy(maxX-138,minY+362,"+/-25VDC MAX");

setcolor(DARKGRAY);

arc(maxX-154,minY+349,0,90,3);
line(maxX-151,minY+349,maxX-151,minY+415);
arc(maxX-154,minY+415,270,360,3);
line(maxX-154,minY+346,maxX-177,minY+346);      /*CENTER*/
line(maxX-177,minY+346,maxX-177,minY+418);
line(maxX-177,minY+418,maxX-154,minY+418);

arc((maxX/2)+25,minY+349,90,180,3);
line((maxX/2)+22,minY+349,(maxX/2)+22,minY+415);
line((maxX/2)+25,minY+346,maxX-180,minY+346);      /*LEFT*/
line(maxX-180,minY+346,maxX-180,minY+418);
line(maxX-180,minY+418,(maxX/2)+25,minY+418);
arc((maxX/2)+25,minY+415,180,270,3);


setfillstyle(SOLID_FILL,DARKGRAY);
bar(minX+90,maxY-90,minX+130,maxY-75);
setfillstyle(SOLID_FILL,BLACK);
bar(minX+125,maxY-90,minX+130,maxY-75);
setfillstyle(SOLID_FILL,LIGHTGRAY);
bar(minX+80,maxY-90,minX+90,maxY-75);
setfillstyle(SOLID_FILL,WHITE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
bar(minX+120,maxY-80,minX+121,maxY-85);    /*POWER BUTTON*/
setcolor(LIGHTGRAY);
line(minX+125,maxY-90,minX+125,maxY-75);
setcolor(BLACK);
line(minX+80,maxY-90,minX+130,maxY-90);
line(minX+130,maxY-90,minX+130,maxY-75);
line(minX+130,maxY-75,minX+80,maxY-75);
line(minX+80,maxY-75,minX+80,maxY-90);
settextstyle(SMALL_FONT,HORIZ_DIR,2);
outtextxy(minX+82,maxY-97,"OFF");
outtextxy(minX+123,maxY-97,"ON");
setusercharsize(1,0,1,0);
settextstyle(SMALL_FONT,HORIZ_DIR,3);
outtextxy(minX+97,maxY-105,"POWER");

setfillstyle(SOLID_FILL,BLACK);
bar(minX+68,maxY-126,minX+141,maxY-114);
settextstyle(DEFAULT_FONT,HORIZ_DIR,1);
setcolor(WHITE);
line(minX+67,maxY-126,minX+67,maxY-114);
line(minX+67,maxY-126,minX+141,maxY-126);
setlinestyle(SOLID_LINE,1,3);                /*ADVANTEST*/
line(minX+74,maxY-123,minX+74,maxY-117);
line(minX+71,maxY-117,minX+74,maxY-123);
outtextxy(minX+78,maxY-123,"DVANTEST");
setusercharsize(3,4,1,1);
settextstyle(SMALL_FONT,HORIZ_DIR,USER_CHAR_SIZE);
setcolor(BLACK);
outtextxy(minX+150,maxY-124,"R4131B  SPECTRUM ANALYZER");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
setfillstyle(SOLID_FILL,BLACK);
fillellipse(minX+170,maxY-82,8,8);
setfillstyle(SOLID_FILL,LIGHTGRAY);
fillellipse(minX+170,maxY-82,6,6);          /*PHONE*/
setfillstyle(SOLID_FILL,BLACK);
fillellipse(minX+170,maxY-82,4,4);
settextstyle(SMALL_FONT,HORIZ_DIR,3);
outtextxy(minX+161,maxY-102,"PHONE");

fillellipse(minX+206,maxY-82,5,5);
setfillstyle(SOLID_FILL,LIGHTGRAY);
fillellipse(minX+206,maxY-82,4,4);
setfillstyle(SOLID_FILL,BLACK);          /*AMPTD CAL.*/
fillellipse(minX+206,maxY-82,2,2);
settextstyle(SMALL_FONT,HORIZ_DIR,2);
outtextxy(minX+197,maxY-101,"AMPTD");
outtextxy(minX+200,maxY-95,"CAL.");

fillellipse(minX+250,maxY-82,11,11);
setcolor(LIGHTGRAY);
circle(minX+250,maxY-82,9);
setcolor(BLACK);
circle(minX+250,maxY-82,8);                  /*INTENSITY*/
setfillstyle(SOLID_FILL,LIGHTGRAY);
fillellipse(minX+250,maxY-82,7,7);
outtextxy(minX+235,maxY-102,"INTENSITY");

fillellipse(minX+300,maxY-82,12,12);
line(minX+299,maxY-94,minX+299,maxY-70);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
line(minX+301,maxY-94,minX+301,maxY-70);
setfillstyle(SOLID_FILL,BLACK);
fillellipse(minX+300,maxY-82,10,10);
setfillstyle(SOLID_FILL,LIGHTGRAY);
fillellipse(minX+300,maxY-82,7,7);          /*CAL. OUT*/
setcolor(BLACK);
circle(minX+300,maxY-82,5);
circle(minX+300,maxY-82,3);
outtextxy(minX+286,maxY-102,"CAL. OUT");
settextstyle(SMALL_FONT,HORIZ_DIR,2);
outtextxy(minX+276,maxY-66,"200MHz");
outtextxy(minX+276,maxY-60,"-30dbm");


setviewport(minX+466,minY+362,minX+484,minY+378,1);
button();                          /*VIDEO FILTER*/
setcolor(WHITE);
arrow(7,10,6,8,6,8,9,6,12,8,11,8,11,10);
setviewport(minX+466,minY+385,minX+484,minY+401,1);
button();                          /*NORMAL DET*/
setcolor(WHITE);
arrow(7,6,7,8,6,8,9,10,12,8,11,8,11,6);
setviewport(minX+435,minY+362,minX+453,minY+378,1);
button();                          /*TIME/DIV*/
setcolor(WHITE);
arrow(7,10,6,8,6,8,9,6,12,8,11,8,11,10);
setviewport(minX+435,minY+385,minX+453,minY+401,1);
button();                          /*POS PK DET*/
setcolor(WHITE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
arrow(7,6,7,8,6,8,9,10,12,8,11,8,11,6);
setviewport(minX+405,minY+385,minX+423,minY+401,1);
button();                          /*SAMPLE DET*/
setviewport(minX+364,minY+151,minX+382,minY+167,1);
button();                           /*LCL*/
setviewport(minX+364,minY+182,minX+382,minY+198,1);
button();                          /*WRITE*/
setviewport(minX+364,minY+216,minX+382,minY+232,1);
button();                          /*STORE*/
setviewport(minX+364,minY+251,minX+382,minY+267,1);
button();                          /*VIEW*/
setviewport(minX+364,minY+280,minX+382,minY+296,1);
button();                          /*RECALL*/
setviewport(minX+364,minY+309,minX+382,minY+325,1);
button();                         /*SHIEFT*/
setviewport(minX+397,minY+161,minX+421,minY+177,1);
button();                          /*NORM*/
setusercharsize(3,4,3,4);
settextstyle(SMALL_FONT,HORIZ_DIR,USER_CHAR_SIZE);
outtextxy(7,4,"RBW");
setviewport(minX+428,minY+161,minX+452,minY+177,1);
button();                        /*NOISE/Hz*/
settextstyle(SMALL_FONT,HORIZ_DIR,USER_CHAR_SIZE);
outtextxy(5,4,"AUTO");
setviewport(minX+458,minY+161,minX+482,minY+177,1);
button();                          /*ZERO*/
settextstyle(SMALL_FONT,HORIZ_DIR,USER_CHAR_SIZE);
outtextxy(5,4,"SPAN");
setviewport(minX+394,minY+202,minX+436,minY+222,1); /*PLOT*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
button();
setcolor(WHITE);              /*DSPL LINE*/
arrow(12,7,16,7,16,4,19,10,16,16,16,13,12,13);
arrow(30,7,26,7,26,4,23,10,26,16,26,13,30,13);
setviewport(minX+442,minY+202,minX+484,minY+222,1);
button();
setcolor(WHITE);              /*10DB/DIV*/
arrow(19,7,15,7,15,4,12,10,15,16,15,13,19,13);
arrow(23,7,27,7,27,4,30,10,27,16,27,13,23,13);
setviewport(minX+394,minY+257,minX+436,minY+277,1);
button();
setcolor(WHITE);              /*2DB/DIV*/
arrow(18,7,18,11,15,11,21,14,27,11,24,11,24,7);
setviewport(minX+442,minY+257,minX+484,minY+277,1);
button();
setcolor(WHITE);            /*COARSE/FINE*/
arrow(18,14,18,10,15,10,21,7,27,10,24,10,24,14);
setviewport(minX+406,minY+309,minX+424,minY+325,1);
button();
setviewport(minX+454,minY+309,minX+472,minY+325,1);
button();                     /*LIMITS*/
setviewport(minX+498,minY+161,minX+516,minY+177,1);
button();                   /*CTR FREQ*/
setviewport(minX+522,minY+161,minX+540,minY+177,1);
button();                     /*MARKER*/
setviewport(minX+546,minY+161,minX+564,minY+177,1);
button();                       /*OFF*/
setviewport(minX+498,minY+262,minX+516,minY+278,1);
button();                     /*CF CAL*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
setviewport(minX+522,minY+262,minX+540,minY+278,1); /*PEAK*/
button();
setviewport(minX+546,minY+262,minX+564,minY+278,1);/*MCR-CF*/
button();
setviewport(minX+504,minY+306,minX+522,minY+322,1); /*INPUT*/
button();
setcolor(WHITE);
arrow(7,6,7,8,6,8,9,10,12,8,11,8,11,6);
setviewport(minX+542,minY+306,minX+560,minY+322,1);
button();                    /*ATTENUATER*/
setcolor(WHITE);
arrow(7,10,6,8,6,8,9,6,12,8,11,8,11,10);
setviewport(minX+347,minY+375,minX+365,minY+391,1);
button();                  /*START/RESET*/

setviewport(minX,minY,maxX,maxY,1);
setfillstyle(SOLID_FILL,DARKGRAY);
sector(maxX-108,minY+218,0,57,28,28);
sector(maxX-108,minY+218,237,360,28,28);  /*DATA KNOB*/
setfillstyle(SOLID_FILL,WHITE);
sector(maxX-108,minY+218,58,238,28,28);
setfillstyle(SOLID_FILL,LIGHTGRAY);
fillellipse(maxX-108,minY+218,25,25);

settextstyle(SMALL_FONT,HORIZ_DIR,2);
setcolor(BLACK);
outtextxy((maxX/2)+49,minY+145,"LCL");
outtextxy((maxX/2)+27,minY+149,"RNT");
light((maxX/2)+32,minY+159);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
line((maxX/2)+32,minY+180,(maxX/2)+41,minY+180);
outtextxy((maxX/2)+45,minY+176,"WRITE");
light((maxX/2)+32,minY+180);

line((maxX/2)+32,minY+202,(maxX/2)+45,minY+202);
outtextxy((maxX/2)+49,minY+199,"MAX");
outtextxy((maxX/2)+45,minY+210,"STORE");
light((maxX/2)+32,minY+202);

line((maxX/2)+32,minY+246,(maxX/2)+44,minY+246);
outtextxy((maxX/2)+47,minY+243,"VIEW");
outtextxy((maxX/2)+43,minY+274,"RECALL");
light((maxX/2)+32,minY+246);

line((maxX/2)+32,minY+305,(maxX/2)+42,minY+305);
outtextxy((maxX/2)+47,minY+296,"SAVE");
outtextxy((maxX/2)+45,minY+303,"SHIFT");
light((maxX/2)+32,minY+305);

outtextxy(minX+402,minY+178,"NORM");
trian(406,159,412,159,409,156);
light(minX+409,minY+150);

trian(437,159,443,159,440,156);
outtextxy(minX+426,minY+178,"NOISE/Hz");
light(minX+440,minY+150);

trian(467,159,473,159,470,156);
outtextxy(minX+463,minY+178,"ZERO");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
light(minX+470,minY+150);

outtextxy(minX+496,minY+178,"CF ADJ");
setusercharsize(1,2,3,4);
settextstyle(SMALL_FONT,HORIZ_DIR,USER_CHAR_SIZE);
outtextxy(minX+495,minY+154,"CTR FREQ");
light(minX+507,minY+150);

outtextxy(minX+522,minY+154,"MARKER");
settextstyle(SMALL_FONT,HORIZ_DIR,2);
outtextxy(minX+550,minY+155,"OFF");
outtextxy(minX+526,minY+178,"OBW");
outtextxy(minX+547,minY+178,"INSTR");
outtextxy(minX+544,minY+184,"PRESET");
light(minX+531,minY+150);

setlinestyle(SOLID_LINE,NORM_WIDTH,1);
setcolor(DARKGRAY);
line(540,169,546,169);
line(minX+436,minY+267,minX+442,minY+267);
line(minX+439,minY+267,minX+439,minY+317);
line(minX+439,minY+317,minX+424,minY+317);
line(minX+356,minY+391,minX+356,minY+396);
line(minX+356,minY+396,minX+361,minY+396);
line(minX+401,minY+389,minX+401,minY+396);
trian(400,393,402,393,401,396);
light(minX+439,minY+301);

outtextxy(minX+509,minY+323,"0db");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
line(minX+499,minY+326,minX+506,minY+326);
light(minX+499,minY+326);

outtextxy(minX+405,minY+349,"SWEEP");
light(minX+414,minY+360);

setlinestyle(SOLID_LINE,THICK_WIDTH,3);
line(minX+390,minY+355,minX+390,minY+401);
setlinestyle(SOLID_LINE,NORM_WIDTH,1);
outtextxy(minX+355,minY+357,"FREE RUN");
light(minX+390,minY+360);

outtextxy(minX+372,minY+369,"LINE");
light(minX+390,minY+372);

outtextxy(minX+368,minY+381,"VIDEO");
light(minX+390,minY+384);

outtextxy(minX+364,minY+393,"SINGLE");
light(minX+390,minY+396);

outtextxy(minX+347,minY+363,"START");
outtextxy(minX+344,minY+369,"/RESET");
outtextxy(minX+401,minY+379,"TRIGGER");
outtextxy(minX+403,minY+402,"SAMPLE");
outtextxy(minX+409,minY+408,"DET");
outtextxy(minX+434,minY+402,"POS PK");
outtextxy(minX+440,minY+408,"DET");
outtextxy(minX+465,minY+402,"NORMAL");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
outtextxy(minX+470,minY+408,"DET");
outtextxy(minX+466,minY+350,"VIDEO");
outtextxy(minX+468,minY+356,"FLTR");
outtextxy(minX+429,minY+356,"TIME/DIV");
outtextxy(minX+496,minY+256,"CF CAL");
outtextxy(minX+524,minY+256,"PEAK");
outtextxy(minX+545,minY+256,"MKR-CF");
outtextxy(minX+408,minY+223,"PLOT");
outtextxy(minX+447,minY+223,"DSPL LINE");
outtextxy(minX+401,minY+278,"10db/DIV");
outtextxy(minX+451,minY+278,"2db/DIV");
outtextxy(minX+404,minY+297,"COARSE");
outtextxy(minX+396,minY+303,"XXFINE");
setfillstyle(SOLID_FILL,BLACK);
fillellipse(minX+399,minY+300,1,1);
outtextxy(minX+454,minY+303,"UNITS");
outtextxy(minX+412,minY+325,"QP");
outtextxy(minX+453,minY+325,"LINEAR");
setusercharsize(1,2,3,4);
settextstyle(SMALL_FONT,HORIZ_DIR,USER_CHAR_SIZE);
outtextxy(minX+521,minY+277,"SIG TRK");
outtextxy(minX+494,minY+277,"ZERO CAL");
outtextxy(minX+508,minY+295,"INPUT ATTENUATOR");
settextstyle(SMALL_FONT,HORIZ_DIR,3);
outtextxy(minX+411,minY+245,"REFERENCE LEVEL");

}
button()
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
struct viewporttype vp;
 getviewsettings(&vp);
 setcolor(BLACK);
 line(1,1,(vp.right-vp.left)-1,1);
 line((vp.right-vp.left)-1,1,(vp.right-vp.left)-1,
 (vp.bottom-vp.top)-1);
 line((vp.right-vp.left)-1,(vp.bottom-vp.top)-1,1,
 (vp.bottom-vp.top)-1);
 line(1,(vp.bottom-vp.top)-1,1,1);
 setcolor(WHITE);
 line(2,2,(vp.right-vp.left)-2,2);
 line(2,(vp.bottom-vp.top)-2,2,2);
 setcolor(DARKGRAY);
 line((vp.right-vp.left)-2,2,(vp.right-vp.left)-2,
 (vp.bottom-vp.top)-2);
 line((vp.right-vp.left)-2,(vp.bottom-vp.top)-2,2,
 (vp.bottom-vp.top)-2);

}
light( int x, int y)
{
  setfillstyle(SOLID_FILL,GREEN);
  fillellipse(x,y,3,3);
}
trian( int x1, int y1, int x2, int y2, int x3, int y3)
{
 setcolor(BLACK);
 line(x1,y1,x2,y2);
 line(x2,y2,x3,y3);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 line(x3,y3,x1,y1);
 setfillstyle(SOLID_FILL,BLACK);
 floodfill((x1+x2)/2,(y1+y3)/2,0);
}
arrow( int x1, int y1, int x2, int y2, int x3, int y3, int x4,
       int y4, int x5, int y5, int x6, int y6, int x7, int y7)
{
 line(x1,y1,x2,y2);
 line(x2,y2,x3,y3);
 line(x3,y3,x4,y4);
 line(x4,y4,x5,y5);
 line(x5,y5,x6,y6);
 line(x6,y6,x7,y7);
 line(x7,y7,x1,y1);
}
```

3.4 ส่วนของโปรแกรมควบคุมเมาส์

```
#include <bios.h>
#include <dos.h>
#include <conio.h>
#include <process.h>
#include <stdio.h>
#include "mouse.h"

int mouse_text_x;
int mouse_text_y;
int mouse_grph_x;
int mouse_grph_y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 }
 return 1;
}
int init_mouse(int need_mouse, int gd, int gm)
{
  int m1;
  mouse_initialized = 0;
  if (check_mouse_driver(need_mouse)) {
     if (gd == 7) *bios_video_area = 6;
     if (low_res_mode(gd,gm))
low_resolution = 1;
     m1 = mouse_reset();
     if (m1) {
mouse_initialized = 1;
move_mouse(0,0);
mouse_on(0);
     }
     else {
       if (need_mouse) {
  printf("ERROR activating mouse ... \n");
  exit(1);
        }
     }
  }
  return mouse_initialized;
}
static int low_res_mode(int gd, int gm)
{
  if (
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
void mouse_txt_posn(int *x, int *y)
{
  mouse_grph_posn(x,y);
  *x = mouse_text_x;
  *y = mouse_text_y;
  return;
}
int mouse_in_box(int graphflag,int left,int top,
                 int right,int bottom){
  int x,y;
  if (mouse_initialized) {
     if (graphflag) {
       x = mouse_grph_x;
       y = mouse_grph_y;
     }
     else {
x = mouse_text_x;
y = mouse_text_y;
     }
     if ((y >= top) && (y <= bottom) &&
(x >= left) && (x <= right)) return 1;
  }
  return 0;
}
int button_release(int b)
{
  int m1, m2, m3, m4;
  if (mouse_initialized) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 ส่วนโปรแกรม MOUSE.H

```
/* Mouse toolkit header file */
/* Macros to retrieve low and high byte of an integer */
#define lo(f)   ((f) & 0xff)
#define hi(f)   (lo(f >> 8))
/* Common scan-ascii codes */
#define CTRLC          0x2e03
#define CTRLH          0x2308
#define CTRLI          0x1709
#define CTRLL          0x260c
#define CTRLK          0x250b
#define CTRLJ          0x240a
#define CTRLU          0x1615
#define CTRLR          0x1312
#define CRKEY          0x1c0d
#define CTRLCRKEY      0x1c0a
#define UPKEY          0x4800
#define DOWNKEY        0x5000
#define LEFTKEY        0x4b00
#define RIGHTKEY       0x4d00
#define SHFTLEFT       0x4b34
#define SHFTRIGHT      0x4d36
#define DELKEY         0x5300
#define INSKEY         0x5200
#define BSKEY          0x0e08
#define SPACEBAR       0x3920
#define PGUPKEY        0x4900
#define PGDNKEY        0x5100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define SHFTUPKEY     0x4838
#define SHFTDNKEY     0x5032
#define SHFTPGUPKEY   0x4939
#define SHFTPGDNKEY   0x5133
#define HOMEKEY       0x4700
#define ENDKEY        0x4f00
#define ESCKEY        0x011b
#define ALT_D         0x2000
#define ALT_E         0x1200
#define ALT_I         0x1700
#define ALT_R         0x1300
#define ALT_S         0x1f00
#define ALT_T         0x1400
#define ALT_X         0x2d00
#define F10KEY        0x4400
/* Mouse pseudo "key" codes */
#define LEFT_MOUSE_PRESS   0xff01
#define RIGHT_MOUSE_PRESS  0xff02
#define LEFT_MOUSE_REL     0xff11
#define RIGHT_MOUSE_REL    0xff12
/* Mouse driver function codes */
#define M_RESET              0
#define M_SHOW_CURS          1
#define M_HIDE_CURS          2
#define M_GET_STATUS         3
#define M_SET_CURS           4
#define M_GET_PRESS          5
#define M_GET_REL            7
#define M_SET_X_BOUNDS       8
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define M_SET_Y_BOUNDS      8
#define M_SET-G_CURS        9
#define M_SET_T_CURS        10
/* define other constants */
#define MOUSE_NEEDED     1
#define MOUSE_OPTIONAL   0
#define MOUSE_TEXT_MODE  0
/* Mouse external variable */
extern int mouse_text_x;
extern int mouse_text_y;
extern int mouse_grph_x;
extern int mouse_grph_y;
extern int mouse_intialized;
/* Mouse function prototypes */
extern void mouse(int *m1, int *m2, int *m3, int *m4);
extern int check_mouse_driver(int need_mouse);
extern int init_mouse(int need_mouse, int gd, int gm);
extern int mouse_reset(void);
extern void move_mouse(int x, int y);
extern void mouse_on(int code);
extern void mouse_off(int code);
extern void mouse_grph_posn(int *x, int *y);
extern void mouse_txt_posn(int *x, int *y);
extern int mouse_in_box(int graphflag, int left, int right,
int top, int bottom);
extern int button_release(int b);
extern int button_press(int b);
extern int button_state(void);
extern int mouse_trigger(int button_dir);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปและวิจารณ์ผลการทดลอง

ปริญญานิพนธ์ฉบับนี้สามารถแบ่งได้เป็น 2 ส่วน คือ

1. ส่วนของการทำ simulate เป็นเครื่องมือวัด
2. การเขียนโปรแกรมควบคุม Spectrum Analyzer

ในส่วนของตัว simulate เป็นเครื่องมือวัด ได้ออกแบบวงจร และ เขียนโปรแกรมการรับส่งข้อมูลซึ่งสามารถทำกระบวนการ Hand shakeได้ทำให้สามารถรับข้อมูลได้แต่ที่ต้องปรับปรุงคือในส่วนของDACซึ่งยังได้ผลที่ไม่แน่นอน

ในส่วนของการเขียนโปรแกรมควบคุม Spectrum Analyzer ผู้จัดทำได้เขียนโปรแกรมควบคุมได้แล้วทุกฟังก์ชันของตัว Spectrum Analyzer และ เขียนโปรแกรมเพื่อให้ใช้งานได้สะดวก ซึ่งได้ออกแบบเป็นรูปกราฟฟิก เหมือนกับตัว Spectrum Analyzer โดยใช้เมาส์ทำหน้าที่ควบคุมฟังก์ชันต่างๆ

แนวทางและการพัฒนา

อาจจะทำ card 8255 ไว้เป็นตัว controller โดยศึกษาได้จากในส่วนของ simulate และอาจทำการต่อเครื่องมือวัดหลายๆเครื่อง เพื่อทำให้เป็นระบบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้