

การควบคุมโดยใช้มาตรฐาน
IEEE-488 (GPIB)
The controlling method by using
IEEE-488 (GPIB)



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมอิเล็กทรอนิกส์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2536

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มี 033221

ปริญญาโทปีการศึกษา 2536

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การควบคุมโดยใช้ IEEE-488 (GPIB)

ผู้จัดทำ

นาย จรุณ นิงขวัญ 33100047

นาย ศักดาคม เลาจเวชกุล 33100369

นาย อาทิตย์ ศรีแก้ว 33100521



(รศ.ดร.มนัส สังวรศิลป์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ทางคณะผู้จัดทำปริญญานิพนธ์ขอกราบขอบพระคุณเป็นอย่างสูงต่อ รองศาสตราจารย์
ดร.มนัส สัจวารศิลป์ อาจารย์ที่ปรึกษา ซึ่งให้คำแนะนำในการจัดทำโครงการและ ปริญญา
นิพนธ์ฉบับนี้จนเสร็จสมบูรณ์ และขอขอบคุณที่ ๆ เพื่อน ๆ ทุกคน ที่กรุณาได้ให้คำแนะนำในด้าน
ต่าง ๆ และคอยเป็นกำลังใจ ซึ่งทำให้ปริญญานิพนธ์ฉบับนี้เสร็จสมบูรณ์ยิ่งขึ้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมโดยใช้ IEEE-488 (GPIB)

Controlling method by using IEEE-488 (GPIB)

โดย นาย จรุณ มิ่งขวัญ 33100047

นาย ศักคาคม เลาจเวชกุล 33100369

นาย อาทิกย์ ศรีแก้ว 33100521

อาจารย์ที่ปรึกษา รศ.ดร. มนัส สังวรศิลป์

บทคัดย่อ

ปริญญาานิพนธ์ฉบับนี้กล่าวถึง การควบคุมอุปกรณ์ต่าง ๆ โดยใช้มาตรฐาน IEEE 488 (GPIB) อันเป็นระบบมาตรฐานการส่งข้อมูลของ IEEE ซึ่งใช้อย่างแพร่หลายในโรงงานอุตสาหกรรมในปัจจุบัน มาตรฐานการส่งข้อมูลแบบนี้ เป็นการรับหรือส่งข้อมูลระหว่าง Controller(PC) กับตัว Talker หรือ Listener โดยใช้การส่งข้อมูล แบบขนาน 8-bit และควบคุมโดย Control line 8-bit โดยในปริญญาานิพนธ์ฉบับนี้ ได้ทำการติดต่อสื่อสารกับเครื่อง Spectrum Analyzer, Oscilloscope และ Digital multimeter (DVM) รวมทั้งยังได้นำเสนอการประยุกต์ใช้งานโดยนำเอา DVM 2 ชุดมาวัดกระแสและแรงดันไฟฟ้าของ Zener diode และ plot กราฟ เพื่อหาค่า Breakdown voltage ของ Zener diode นั้น

ABSTRACT

This thesis is about controlling method by using IEEE 488 (GPIB). That sending standard information system of IEEE. It use widespreadly in industrial factory now. By sending or receiving information between Controller (PC) and Talker or Listener in parallel 8-bit of databus and 8-bit control information. In this thesis IEEE 488 is used to control Spectrum analyzer, Oscilloscope and Digital multimeter (DVM). In addition to present the application with IEEE-488, by use two DVM to measure Voltage and Current of Zener diode and plot graph for finding breakdown voltage of zener diode.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทนำ : การควบคุมด้วย IEEE-488	1
ประวัติการพัฒนา IEEE-488	1
บทที่ 1 : IEEE-488 (GPIB)	2
โครงสร้างของ IEEE-488	2
ขีดจำกัดของ IEEE-488	2
รายละเอียดเกี่ยวกับ IEEE-488	3
ความหมายของสัญญาณต่างๆ ภายใน IEEE-488	4
การเชื่อมต่ออุปกรณ์ต่างๆ ในระบบ IEEE-488 BUS	7
ขบวนการแฮนด์เชค (Handshake Procedure)	9
ขบวนการแฮนด์เชคของตัวควบคุมซึ่งทำหน้าที่เป็นตัวส่ง	11
ขบวนการแฮนด์เชคของตัวควบคุมซึ่งทำหน้าที่เป็นตัวรับ	12
คำสั่งใช้งานของ GPIB	14
การขอบริการและการตรวจสอบ	19
รูปแบบของข้อมูล	19
บทที่ 2 : การออกแบบวงจร	22
IC TMS9914A	22
ลักษณะภายนอกและการจัดขาสัญญาณต่างๆ ของ TMS9914A	23
ฟังก์ชันการทำงานของ REGISTER ภายใน IC TMS9914A	26
วงจรส่วนการ DECODE ADDRESS จาก PC	30
วงจรส่วน INTERFACE กับเครื่องมือวัด	31

	หน้า
บทที่ 3 : ผลการทดลอง	32
CODE การทำงานของอุปกรณ์ในระบบ	
- DIGITAL MULTIMETER model TR6845.	32
- SPECTRUM ANALYZER model R4131	37
- INTRON OSCILLOSCOPE model DSO-2022	42
การเขียนโปรแกรมควบคุม IEEE-488 GPIB	46
บทที่ 4 : Principle of Programming	54
Initial controller card	54
IFC (Interface Clear)	55
การส่งข้อมูลจาก controller	57
การรับข้อมูลเข้า controller	59
SDC/DCL (Select Device Clear/Device Clear)	61
REN/GL (Remote Enable/Go to Local)	64
รายละเอียด flow chart ย่อย	67
How about this project?	72
บทที่ 5 : สรุปผลและวิจารณ์	75

บทนำ

การควบคุมด้วย IEEE-488

ระบบอุตสาหกรรมในอดีตมักใช้อุปกรณ์ไม่มากนัก . หากมีการเชื่อมต่อก็สามารถทำได้โดยง่าย ต่อมาการอุตสาหกรรมได้เจริญรุดหน้าไปอย่างรวดเร็วดังนั้นระบบอุตสาหกรรมจึงมีขนาดใหญ่และ ซับซ้อนขึ้นด้วยการเชื่อมต่อระหว่างอุปกรณ์หลายๆชิ้น จะต้องเสียค่าใช้จ่ายจำนวนมากไปด้วย หากจะมีการเชื่อมต่อระหว่างอุปกรณ์หลายๆชิ้นจะต้องเสียค่าใช้จ่ายจำนวนมาก ดังนั้นจึงมีการคิดค้นระบบเชื่อมต่อ ซึ่งเป็นมาตรฐานขึ้นมา นั่นคือ IEEE-488 นั่นเอง

ประวัติการพัฒนา IEEE-488

ดังที่ได้กล่าวมาแล้วข้างต้นว่า การเชื่อมต่อทางอุตสาหกรรมในอดีต เป็นไปด้วยความยากลำบากและ เสียค่าใช้จ่ายมาก ดังนั้นบริษัทผู้ผลิตเครื่องมือวัดต่างๆในประเทศสหรัฐอเมริกาจึงร่วมกันจัดหาระบบเชื่อมต่อมาตรฐานขึ้นมา ซึ่งในประเทศเยอรมันก็มีการพัฒนาระบบเชื่อมต่อมาตรฐานเช่นกันโดยความร่วมมือของ IEC (International Electrotechnical Commission) จนกระทั่งในปี 1972 สหรัฐอเมริกาโดยการนำของ IEEE (Institute of electrical and Electronics Engineers : IEEE) จึงได้มีการประชุมเพื่อวางแผนพิจารณาระบบเชื่อมต่อมาตรฐานร่วมกัน

บริษัทฮิวเลตต์แพกการ์ด ผู้ผลิตเครื่องมือวัดรายใหญ่ในอเมริกา ได้ทำการพัฒนาระบบเชื่อมต่อมาตรฐานอยู่ก่อนแล้วชื่อว่า HPIB (Hewlett Packard Interface Bus) จึงได้เสนอโครงการให้ IEEE เพื่อพิจารณา และได้รับการยอมรับในปี 1975 โดย IEEE จัดให้เป็นมาตรฐานลำดับที่ 488 ดังนั้น จึงได้ชื่อว่า IEEE-Std 488-1975 ซึ่งต่อมาได้มีการปรับปรุงเป็น IEEE-Std 488-1978 หรือที่นิยมเรียกกันว่า GPIB (General Purpose Interface Bus)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

IEEE-488 (GPIB)

โครงสร้างของ IEEE-488

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

- Talker ทำหน้าที่ส่งข้อมูล โดยในระบบสามารถมี Talker ได้หลายตัวแต่จะมีเพียงตัวเดียวเท่านั้นที่กำลังทำงานอยู่
- Listener ทำหน้าที่เป็นตัวรับข้อมูล โดยในระบบเดียวกันสามารถมี Listener ได้หลายตัวเช่นเดียวกัน แต่ Listener สามารถทำงานได้ครั้งละหลายๆตัวได้
- Controller ทำหน้าที่ควบคุมอุปกรณ์ต่างๆในระบบ 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

ข้อกำหนดของ IEEE-488

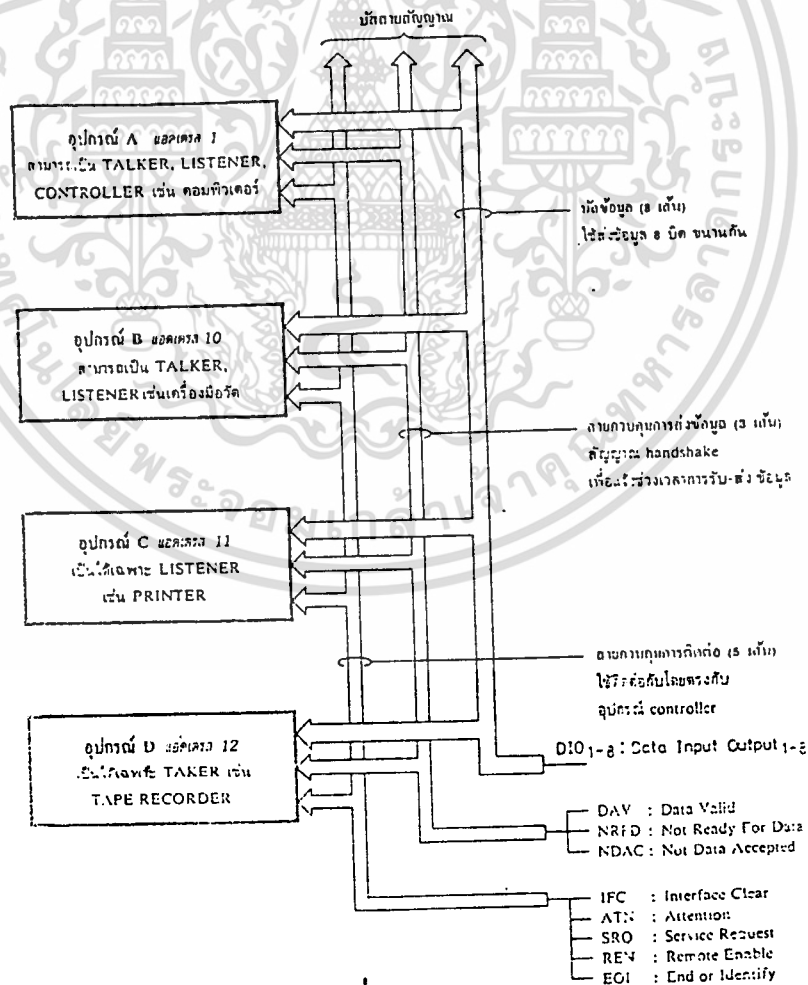
1. จำนวนอุปกรณ์ในระบบ(Talker, Listener, Controller)ที่ต่อกับสายสัญญาณ 1 เส้น จะต้องไม่เกิน 15 เครื่อง
 2. สายเคเบิ้ลที่ต่อระหว่างอุปกรณ์ จะต้องยาวไม่เกิน 4 เมตร และความยาวรวม
- เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของสายเคเบิ้ลในระบบจะต้องไม่เกิน 20 เมตร

3. ความเร็วในการส่งข้อมูลจะต้องไม่เกิน 1Mb/Sec (1ล้านบิตต่อวินาที)
4. ต้องมีการจ่ายไฟให้กับอุปกรณ์มากกว่าครึ่งหนึ่งของระบบ

รายละเอียดเกี่ยวกับ IEEE-488

ลักษณะทางกายภาพของ IEEE-488 นั้นคือ เป็นสายสัญญาณแบบ 24 เส้นขนานกัน และมีขั้วต่ออยู่ทางปลายทั้งสองของสาย เพื่อต่อกับอุปกรณ์ หรือ ต่อกันเพื่อให้สายสัญญาณมีความยาวเพิ่มขึ้น ในจำนวนสายสัญญาณ 24 เส้นนี้ มีเพียง 16 เส้นเท่านั้น ที่ทำหน้าที่นำสัญญาณ ส่วนที่เหลืออีก 8 เส้น ทำหน้าที่กราวด์ (ground) และ ชีลด์ (shield) โดยจำนวนสายที่ใช้นำสัญญาณ 16 เส้นนั้นยังแบ่งได้เป็น 3 ประเภท ตามรูปที่ 1 คือ



รูปที่ 1.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 3 กลุ่ม ในหัวข้อนี้ จะอธิบายความหมายของสัญญาณต่าง ๆ ดังนี้

กลุ่มสัญญาณข้อมูล

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

กลุ่มสัญญาณควบคุมการเชื่อมต่อ (Interface)

1. IFC (Interface Clear) เป็นสัญญาณรีเซ็ต หรือ เคลียร์ระบบ กำเนิดได้โดยตัวควบคุม(Controller)เท่านั้น เมื่ออุปกรณ์ในบัสได้รับสัญญาณเคลียร์นี้ จะกลับคืนสู่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สภาวะเริ่มต้นใหม่ ซึ่งเป็นสภาวะแรกเริ่มก่อนการกำหนดฟังก์ชันเหมือนแรกเปิดสวิตช์

2. ATN (Attention) เป็นสัญญาณที่ถูกส่งโดยอุปกรณ์ที่เป็นตัวควบคุมเช่นเดียวกันใช้ในการสั่งให้อุปกรณ์ทุกตัวในระบบเตรียมพร้อมเพื่อรอรับคำสั่งต่อไป

3. SRQ (Service Request) เป็นสัญญาณที่ถูกส่งจากอุปกรณ์ต่าง ๆ เพื่อเป็นการบอกแก่ระบบว่าขณะนี้อุปกรณ์ดังกล่าวต้องการติดต่อจากตัวควบคุม

4. REN (Remote Enable) สัญญาณนี้ เป็นสัญญาณที่ถูกส่งมาจากตัวควบคุมเพียงตัวเดียวเท่านั้น เพื่อใช้สั่งให้อุปกรณ์ต่างๆเปลี่ยนจากโหมดที่ใช้งานปกติมาเป็นการควบคุมโดยตัวควบคุมแทน

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

กลุ่มสัญญาณควบคุมการรับ-ส่งข้อมูล

1. DAV (Data Valid) เมื่อสัญญาณนี้ถูกตั้งเป็นลอจิก "Low" โดยอุปกรณ์ที่เป็นตัวส่ง (Talker) เป็นการแจ้งแก่ระบบบัสว่า ขณะนี้ตัวส่ง ได้ทำการส่งข้อมูล ลงไปที่สายสัญญาณข้อมูลเรียบร้อยแล้ว

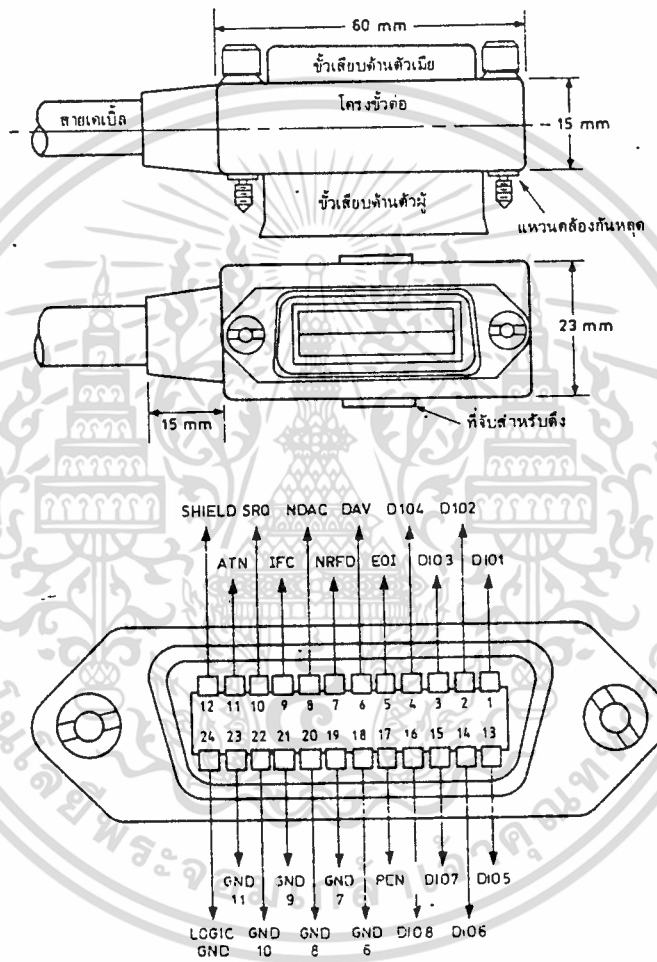
2. NRFD (Not Ready For Data) เมื่อสัญญาณนี้มีลอจิกเป็น "Low" จะเป็นการแสดงว่าในขณะที่ระบบบัสยังไม่พร้อมที่จะรับข้อมูล เนื่องจากอุปกรณ์ในระบบยังพร้อมไม่หมดทุกตัว ซึ่งสัญญาณเส้นนี้จะไม่เป็น "Hi" จนกว่าอุปกรณ์ทุกตัวให้ลอจิกที่เป็น "Hi" ครบถ้วนแล้ว สัญญาณนี้มีประโยชน์ในกรณีที่อุปกรณ์ในระบบมีความเร็วแตกต่างกัน

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราคุ้นเคย



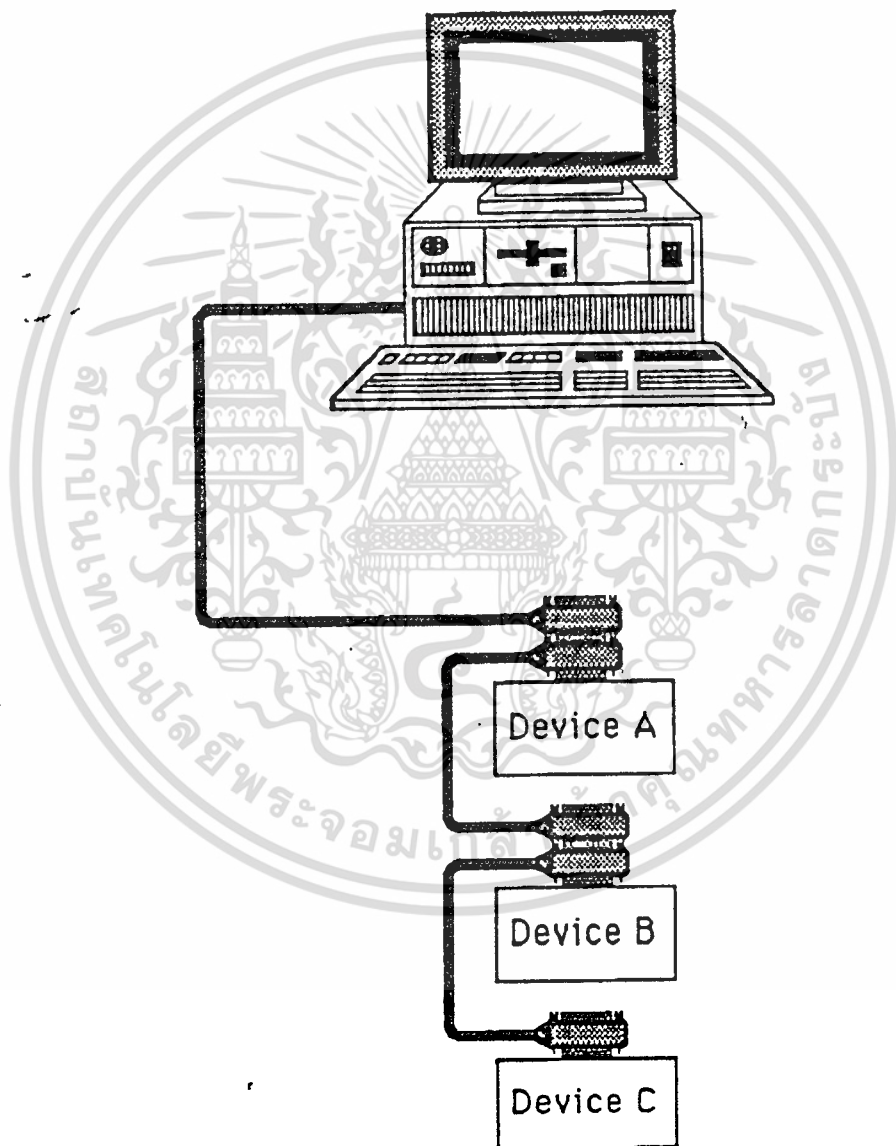
รูปที่ 1.2 ขั้วต่อของ GPIB และการจัดขาของสัญญาณต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่ออุปกรณ์ต่างๆในระบบ IEEE-488 BUS

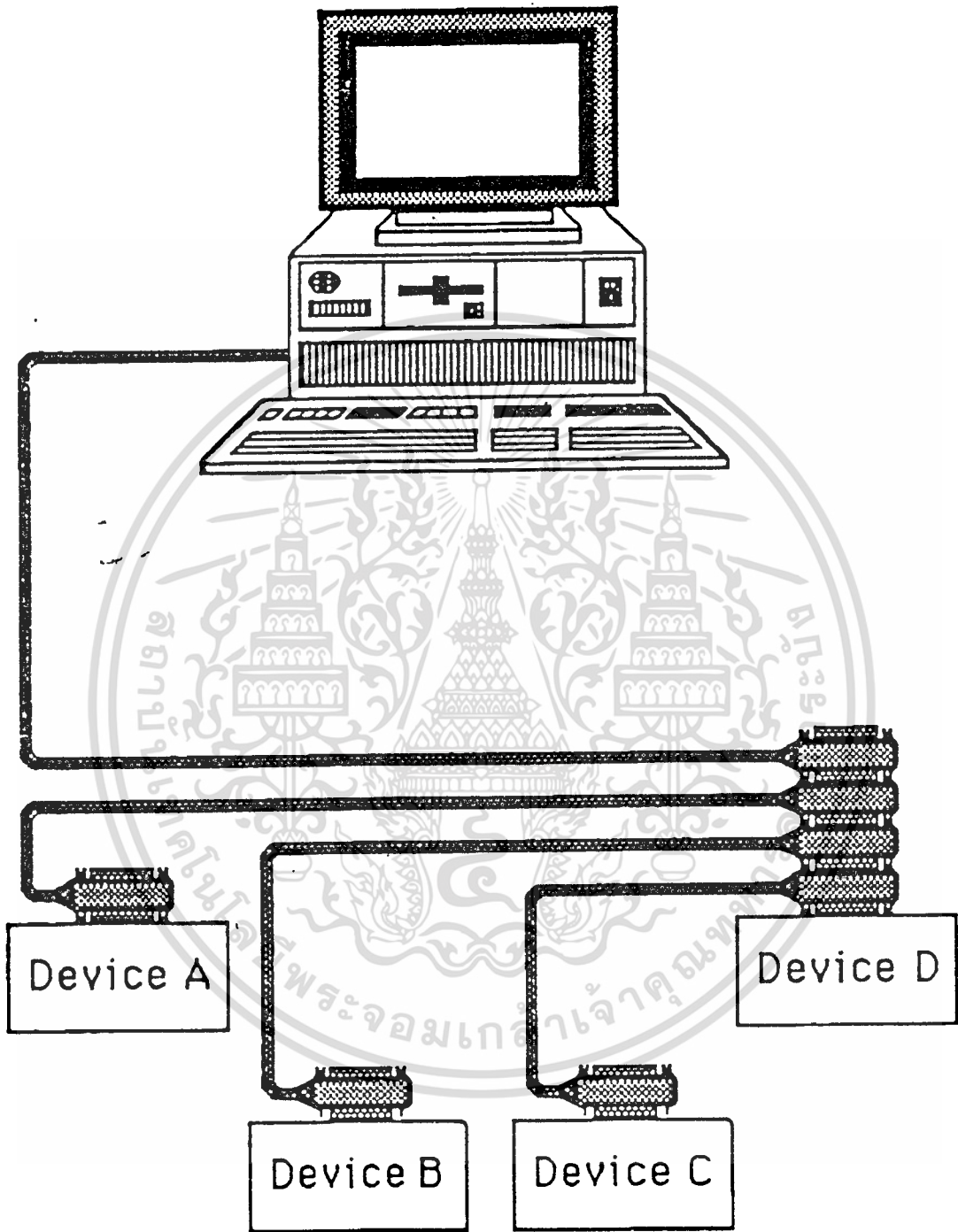
สำหรับการเชื่อมต่อระหว่างอุปกรณ์ต่างๆในระบบ IEEE-488 Bus นั้น มีอยู่ 2 วิธีคือ

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



รูปที่ 1.3 การเชื่อมต่อแบบเรียงต่อเนื่องกัน (Daisy Chain Configuration)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.4 การเชื่อมต่อแบบกระจาย (Star Configuration)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ขบวนการแฮนด์เชค (Handshake Procedure)

เมื่อมีการเชื่อมต่อระหว่างอุปกรณ์ต่าง ๆ ในระบบ ดังนั้นเมื่อระบบจะทำงานจึงต้องมีการตรวจสอบเสียก่อน ซึ่งการตรวจสอบนี้เราเรียกว่าขบวนการแฮนด์เชค (Handshake Procedure)



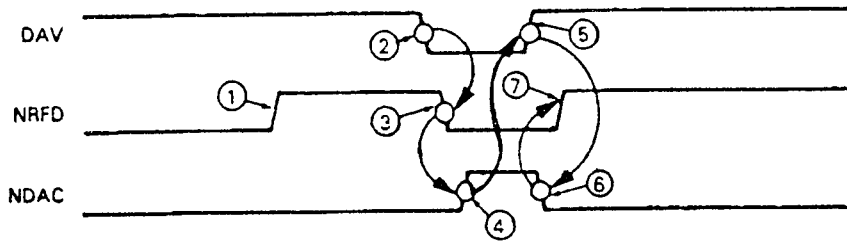
พิจารณาระบบที่ไม่ซับซ้อนนักโดยกำหนดให้มีตัวส่ง (Talker) และตัวรับ (Listener) อย่างละ 1 ตัว

การที่จะมีการสื่อสารระหว่างตัวส่งและตัวรับจะกระทำได้โดยการที่ตัวส่งจะทำหน้าที่ส่งสัญญาณออกมาเพื่อให้ตัวรับทราบว่า มีข้อมูลเสร็จในสายสัญญาณ และตัวรับก็จะส่งสัญญาณเพื่อให้ตัวส่งทราบว่า ได้ทำการรับข้อมูลเป็นที่เรียบร้อยแล้ว

การส่งสัญญาณในการสื่อสารระหว่างตัวส่งและตัวรับนั้น จะส่งมาทางสายสัญญาณ 3 สาย ซึ่งสายสัญญาณดังกล่าวนี้เป็นสายสัญญาณในกลุ่มควบคุมการรับ-ส่งข้อมูล คือ NRFD (Not Ready For Data), DAV (Data Valid) และ NDAC (Not Data Accepted)

DAV เป็นสัญญาณที่ถูกส่งโดยตัวส่ง NRFD และ NDAC เป็นสัญญาณที่ถูกส่งโดยตัวรับ โดยตัวรับจะใช้สัญญาณ NDAC เพื่อชี้ให้เห็นว่าพร้อม หรือไม่พร้อมที่จะรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการน **033221**



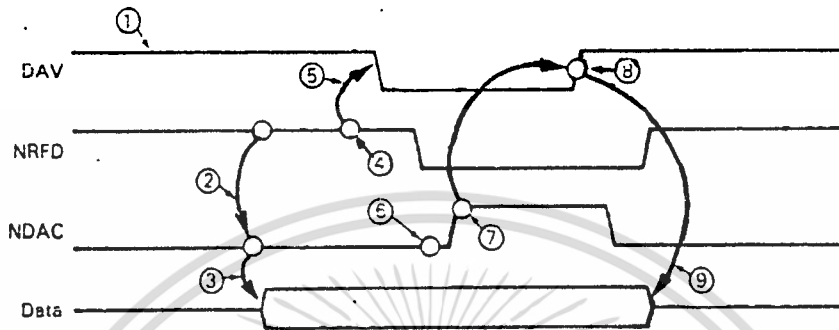
NOTE: The timing shown is relative

รูปที่ 1.6

ขบวนการแฮนด์เชคเริ่มขึ้นเมื่อ ตัวส่งมีข้อมูลที่จะถ่ายทอดลงสายสัญญาณไปยังตัวรับ เริ่มด้วยการที่ตัวรับจะทำให้สัญญาณ NRFD เป็น "Hi" (มีค่าเป็น 0) นั่นคือเป็นการบอกให้ทราบว่าตัวรับพร้อมที่จะให้ตัวส่งทำการถ่ายข้อมูลลงในสายสัญญาณแล้ว (ในจุดที่ 1) ตัวส่ง จะทำการถ่ายข้อมูลลงในสายสัญญาณ ^{*p-3X} DIO1-DIO8 หลังจากทำการรอเวลา เพื่อให้ตัวส่ง ถ่ายข้อมูลแล้ว ตัวส่งจะทำให้สัญญาณ DAV มีลอจิกเป็น "Low" (มีค่าเป็น 1) เพื่อที่จะบอก ให้ทราบว่าขณะนี้ได้มีข้อมูลในสายสัญญาณทั้ง 8 เส้นแล้ว (ในจุดที่ 2) ต่อมาเมื่อผู้รับทราบว่า มีข้อมูลอยู่ในบัสก็จะทำให้สัญญาณ NRFD ให้มีค่าเป็น "Low" เพื่อบอกให้ตัวส่งทราบว่า ยังไม่พร้อมที่จะให้ตัวส่งทำการส่งข้อมูลชุดต่อไป และตัวรับพร้อมที่จะเก็บข้อมูลจากสายสัญญาณ (ในจุดที่ 3) หลังจากตัวรับทำการเก็บข้อมูลจากสายสัญญาณเสร็จแล้ว ตัวรับจะทำให้สัญญาณ NDAC ให้มีสถานะเป็น "Hi" เพื่อบอกให้ทราบว่าได้ทำการเก็บข้อมูลเสร็จแล้ว (ในจุดที่ 4) เมื่อตัวส่งตรวจพบว่าการสัญญาณ NDAC มีลอจิกเป็น "Hi" ก็จะทำให้สัญญาณ DAV ให้มีลอจิกเป็น "Hi" ด้วยเพื่อไม่ให้ตัวรับทำการรับข้อมูลในบัสอีก (ในจุดที่ 5) เมื่อตัวรับทราบว่า สัญญาณ DAV มีลอจิกเป็น "Hi" ก็จะทำให้สัญญาณ NDAC ให้เป็น "Low" ทำให้ข้อมูลในบัสถูกกำจัด ออกไป หลังจากนั้นก็จะทำให้สัญญาณ NRFD ให้เป็น "Hi" เพื่อบอกให้ทราบว่าพร้อมที่จะรับ ข้อมูลชุดต่อไปแล้วขบวนการแฮนด์เชคจึงเสร็จสิ้นลง และจะ เริ่มต้นใหม่เมื่อมีสัญญาณชุดใหม่ เข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขบวนการแฮนด์เชคของตัวควบคุมซึ่งทำหน้าที่เป็นตัวส่ง



รูปที่ 1.7

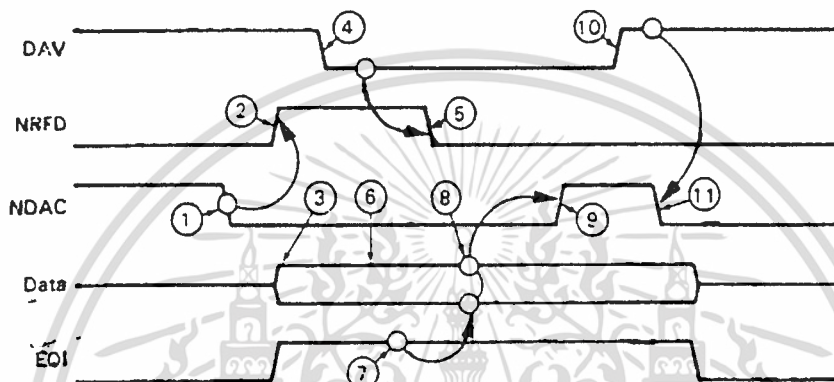
ขบวนการแฮนด์เชคเริ่มขึ้นโดยการที่ตัวควบคุม ทำการเซตค่าให้สัญญาณ DAV มีค่าเป็น "Hi" (ในจุดที่ 1) ต่อมาเป็นการตรวจสอบว่าสัญญาณ NDAC และ NRFD มีลอจิกเป็น "Hi" ทั้งคู่หรือไม่ (ในจุดที่ 2) ถ้าทั้งคู่เป็น "Hi" นั่นคือเกิดการผิดพลาด คืออุปกรณ์นั้นไม่พร้อมที่จะทำงาน จึงทำการออกจากขบวนการแฮนด์เชค

กลับมาดูในจุดที่ 2 หากสัญญาณใดสัญญาณหนึ่งมีลอจิกเป็น "Low" ตัวควบคุมจะทำการส่งข้อมูลลงใน GPIB Bus (ในจุดที่ 3) หลังจากนั้นตัวควบคุมจะทำการตรวจสอบว่าสัญญาณ NRFD มีลอจิกเป็น "Hi" (ในจุดที่ 4) หลังจากนั้นตัวควบคุมจะทำให้สัญญาณ DAV ให้มีลอจิกเป็น "Low" เพื่อให้ตัวรับทราบว่ามีข้อมูลอยู่ในบัสข้อมูล (ในจุดที่ 5) ตัวควบคุมจะรอเวลาเพื่อให้ตัวรับทำการเก็บข้อมูล เมื่อครบกำหนดเวลาตัวควบคุมจะทำการเซตว่าตัวรับทำงานเกินเวลาที่กำหนดหรือไม่หากเกินเวลาก็จะทำการตรวจสอบต่อไปว่าสัญญาณ NDAC ทำลอจิกเป็น "Hi" ไข่หรือไม่ก็จะทำการตรวจสอบอีกครั้งว่าเกินเวลาหรือไม่ กลับไปดูที่จุดที่ 6 หากสัญญาณ NDAC เป็น "Hi" (ในจุดที่ 7) ตัวควบคุมจะทำการตอบสนองโดยการทำให้สัญญาณ DAV ให้มีลอจิกเป็น "Hi" เพื่อบอกให้ตัวรับทราบว่าให้หยุดการเก็บข้อมูลจากบัส(ในจุดที่ 8) หลังจากนั้นข้อมูลในบัสข้อมูลจะถูกนำออกไป (จุดที่ 9) จึงเป็นการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เสร็จสิ้นขบวนการแฮนด์เชค

ขบวนการแฮนด์เชคของตัวควบคุมซึ่งทำหน้าที่เป็นตัวรับ



รูปที่ 1.8

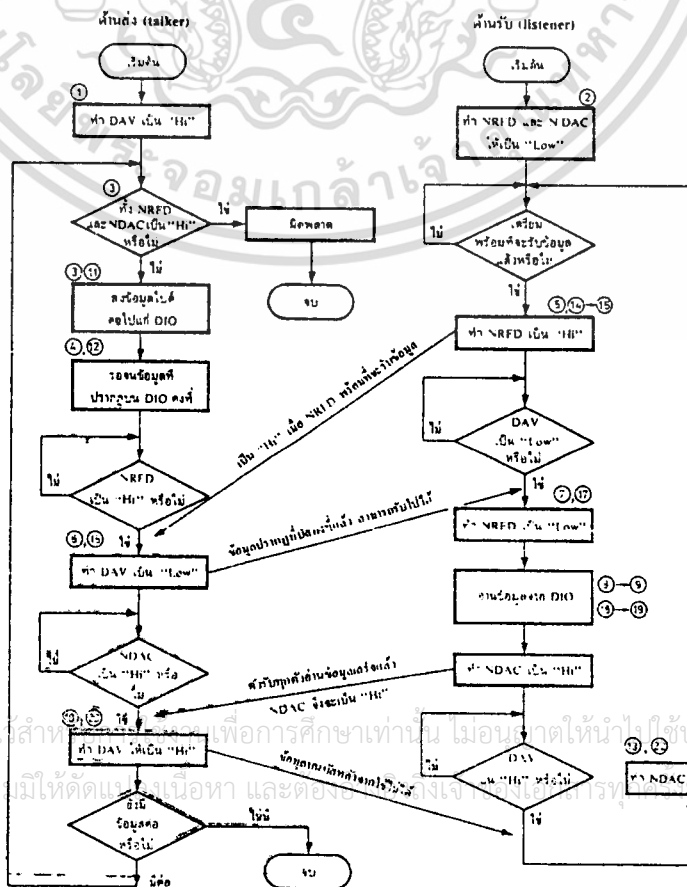
ขบวนการแฮนด์เชคเริ่มขึ้น หลังจากการที่ตัวควบคุมรับรู้ว่าตัวส่งจะทำการส่งข้อมูล เมื่อตัวควบคุมทราบว่าตัวส่งจะทำการส่งจะทำการส่งข้อมูล ตัวควบคุมรับรู้ว่าตัวส่งจะทำให้สัญญาณ NDAC มีลอจิกเป็น "Low" เพื่อบอกให้ทราบว่าตัวควบคุมยังไม่ได้ทำการเก็บข้อมูล (ข้อมูลในที่นี้คือ แอดเดรสหรือ ข้อมูลทั่วไปซึ่งตัวส่งทำการส่งมาในบัสข้อมูล) นั่นคือในจุดที่ 1

ต่อมาตัวควบคุมจะทำการเซตค่าสัญญาณ NRFD ให้เป็น "Hi" เพื่อที่จะให้ตัวส่งทราบว่าขณะนี้ตัวควบคุมพร้อมที่จะรับข้อมูลแล้ว (ในจุดที่ 2) เมื่อสัญญาณ NRFD มีค่าเป็น "Hi" และสัญญาณ NDAC มีค่าเป็น "Low" แล้ว ตัวส่งจะทราบทันทีว่าทำการส่งข้อมูลลงมาในบัสข้อมูลได้แล้ว (ในจุดที่ 3) หลังจากนั้นตัวควบคุมจะทำการรอเวลาให้ตัวส่งทำการส่งข้อมูลลงในบัสข้อมูลให้เสร็จ โดยจะทำการตรวจสอบไปด้วยว่าเกินเวลาที่กำหนดหรือยังหากเกินเวลาที่กำหนดก็จะออกจากขบวนการแฮนด์เชค หากไม่เกินกำหนดเวลาจะทำการตรวจสอบเวลาว่าเกินเวลาที่ต้องคอยแล้วหรือยัง หากยังไม่เกินเวลาที่จะทำการเซคต่อไปจนกระทั่งเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมดเวลาหรือจนกว่าสัญญาณ DAV เป็น "Low" หรือไม่หากไม่ก็จะตรวจสอบเวลาว่าเกินเวลาที่ต้องรอแล้วหรือยัง หากยังไม่เกินเวลาก็จะทำการเช็คต่อไปจนกระทั่งหมดเวลาหรือจนกว่าสัญญาณ DAV จะมีค่าเป็น "Low"

เมื่อสัญญาณ DAV เป็น "Low" แล้วตัวควบคุมจะตอบรับ โดยการทำให้สัญญาณ NRFD มีค่าเป็น "Low" (ในจุดที่ 5) นั่นเป็นการบอกว่าตัวควบคุมพร้อมที่จะเริ่มเก็บข้อมูล (ในจุดที่ 6) หลังจากนั้นตัวควบคุมจะทำการตรวจสอบสัญญาณ EOI เพื่อตรวจสอบว่าข้อมูลที่ตัวส่งทำการส่งลงมานั้นหมดแล้วหรือยัง (ถ้าตัวส่งทำการส่งข้อมูลลงในบัสหมดแล้วจะทำการตั้งค่าสัญญาณ EOI ให้เป็น "Low") (ในจุดที่ 7) หากสัญญาณ EOI เป็น "Low" ตัวควบคุมจะเชตสถานะว่า ขณะนี้ตัวส่ง ได้ทำการส่งข้อมูลลงในบัสหมดแล้ว หากสัญญาณ EOI เป็น "Hi" ก็ไม่ทำการเชตสถานะ หลังจากนั้นตัวควบคุมจะทำการเก็บข้อมูลในบัส (ในจุดที่ 8) แล้วทำให้สัญญาณ NDAC มีค่าเป็น "Hi" เพื่อบอกให้ทราบว่าตัวควบคุมได้ทำการอ่านข้อมูลเสร็จเรียบร้อยแล้ว (ในจุดที่ 9) เมื่อตัวควบคุมทำให้สัญญาณ NDAC เป็น "Hi" แล้วตัวส่งจะลบข้อมูลในบัสข้อมูล โดยการทำให้สัญญาณ DAV เป็น "Hi" (ในจุดที่ 10) ซึ่งก่อนหน้านี้ตัวควบคุมจะทำการตรวจสอบอยู่แล้วว่าสัญญาณ DAV เป็น "Hi" หรือยัง

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



คำสั่งใช้งานของ GPIB

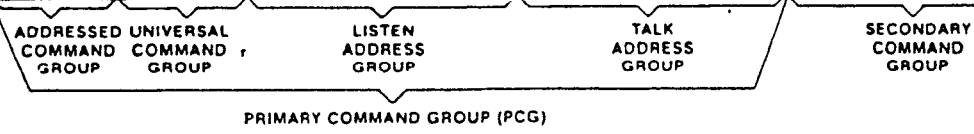
คำสั่งการต่างๆเพื่อกำหนดหน้าที่การทำงานและกำหนดฟังก์ชัน เช่น กำหนดช่วงการวัด โหมดการวัด หรืออื่น ๆ แก่เครื่องวัดที่ต่ออยู่ เหล่านี้ในตัวควบคุมจะเป็นตัวกำหนดโดยการส่งรหัสคำสั่งไปที่อุปกรณ์โดยผ่าน DI1-DI6 รหัสคำสั่งนี้จะถูกส่งไปในช่วงที่สายสัญญาณ ATN เป็น LOW

คำสั่งสำหรับกำหนดหน้าที่การทำงานต่าง ๆ ตามมาตรฐานของ GPIB มีอยู่ด้วยกัน 128 คำสั่งดังแสดงในตารางที่ 1 ต่อไปนี้ โดยแบ่งเป็น 5 กลุ่มคำสั่ง

รหัสที่ใช้ในระบบ GPIB บัสนั้นใช้ร่วมกัน ทั้งรหัสข้อมูล และรหัสคำสั่ง นั่นคือรหัสเดียวกันมีความหมายได้ 2 อย่าง คือ เมื่อ ATN เป็น LOW จะหมายถึงรหัสคำสั่ง แต่ถ้า ATN เป็น HIGH รหัสนี้จะแทนข้อมูลที่เป็น 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	MSG	ASCII	MSG	ASCII	MSG	ASCII	MSG	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	ENO	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	



ตารางที่ 1.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. กลุ่มคำสั่งเจาะจงจุดหมาย (addressed command group) เป็นคำสั่งที่ส่งไปยังอุปกรณ์ที่เป็นตัวส่งหรือตัวรับที่กำหนดไว้ล่วงหน้าแล้ว คำสั่งนี้ประกอบด้วย

GTL (got to local) สั่งให้อุปกรณ์กลับสู่สภาพการควบคุมปกติด้วยมือ

SDC (selected device clear) สั่งให้อุปกรณ์เคลียร์ตัวเองสู่สภาพเริ่มต้นใหม่

PPC (parallel poll configure) เป็นคำสั่งสำหรับทำการจัดสายสัญญาณของการทำการจัดสรรสายสัญญาณของการทำกระบวนการตรวจสอบสภาพอุปกรณ์โดยวิธีขนานโดยใช้กับกลุ่มคำสั่งรอง

GET (group excute trigger) ใช้สั่งเริ่มต้นการทำงานของอุปกรณ์ที่หลายตัว

TCT (take control) เป็นการกำหนดให้อุปกรณ์ตัวส่งทำหน้าที่เป็นตัวควบคุม

2. กลุ่มคำสั่งครอบคลุม (universal command group) เป็นคำสั่งที่ส่งไปยังอุปกรณ์ทุกตัวที่ต่ออยู่ในบัส ประกอบด้วย

LLO (local lockout) เป็นการสั่งให้อุปกรณ์ล็อกอยู่ที่สภาวะควบคุมโดยปุ่มปรับที่หน้าปัดตามปกติ

DCL (devide clear) สั่งให้อุปกรณ์ทุกตัวกลับไปสู่สภาวะเริ่มต้น

PPU (parallel poll unconfigure) ใช้ยกเลิกกระบวนการตรวจสอบสภาพแบบขนานทั้งหมด

SPE (serial poll enable) เปลี่ยนโหมดของการตรวจสอบสภาพเป็นแบบอนุกรม ในโหมดนี้จะเป็นการส่งสถานะของเครื่องแทนการส่งข้อมูล

SPD (serial poll disable) ยกเลิกโหมดการตรวจสอบแบบอนุกรม

3. กลุ่มคำสั่งกำหนดอุปกรณ์ตัวรับ (listener address group) เป็นคำสั่งสำหรับกำหนดให้อุปกรณ์เป็นตัวรับ ตามรหัสหมายเลขจาก 0-30 และมีคำสั่ง UNT(untalker) สำหรับยกเลิก

4. กลุ่มคำสั่งกำหนดอุปกรณ์ตัวส่ง (talker address group) สำหรับกำหนดให้อุปกรณ์เป็นตัวส่ง ตามรหัสหมายเลขจาก 0-30 และมีคำสั่ง UNL (unlisten) สำหรับเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ยกเลิกเช่นกัน

คำสั่งกลุ่มที่ 1 ถึง 4 นั้น จัดเป็นกลุ่มคำสั่งหลัก ที่มีความหมายตายตัวยังมีคำสั่งอีกกลุ่มที่ขึ้นอยู่กับข้อกำหนดภายหลังนั่นคือกลุ่มคำสั่งรอง

5. กลุ่มคำสั่งรอง (secondary command group) เป็นคำสั่งที่กำหนดรายละเอียดย่อยของอุปกรณ์แต่ละตัวที่ต่ออยู่ในระบบ ให้มีการทำงานอย่างไร ตามจุดประสงค์ใช้งานของเครื่องมือชิ้นนั้น เช่นเกี่ยวกับการปรับปุ่มต่าง ๆ ด้วยมือตนเองคำสั่งรองนี้จะตามหลังคำสั่งหลักคือจะใช้หลังจากอุปกรณ์ต่างๆ ถูกกำหนดวางตัวในระบบเรียบร้อยแล้ว

คำสั่งต่างๆที่กล่าวไป ซึ่งใช้ในการกำหนดสภาวะการทำงานของอุปกรณ์ แต่ละสภาวะที่กำหนดไปนั้นเป็นอย่างไร และมีจุดประสงค์เพื่ออะไร ดังต่อไปนี้

Device clear/ Interface Clear

Device clear ใช้ในการทำให้อุปกรณ์ที่ต่ออยู่ในบัสกลับไปสู่สภาวะเริ่มต้น ยังไม่มีการกำหนดฟังก์ชันใดๆ สภาวะเริ่มต้นนี้จะแตกต่างกันไปแล้วแต่ว่าอุปกรณ์นั้นออกแบบไว้อย่างไร Device clear มี อยู่ 2 ลักษณะ คือ เคลียร์หมดทุกตัวที่ต่ออยู่ (DCL) กับเคลียร์เจาะจงอุปกรณ์ตัวใดตัวหนึ่ง (SDC)

แต่ว่าในการเคลียร์อุปกรณ์ให้อยู่ในสภาวะเริ่มต้นนั้นไม่ได้หมายความว่า Interface function ของ GPIB จะถูกเคลียร์อุปกรณ์ให้ไปอยู่ในสภาวะเริ่มต้นด้วยแต่อย่างใด interface function คือสภาพการ interface ที่ได้กำหนดไว้ในระบบประกอบด้วยฟังก์ชันต่างๆ ดังแสดงในตารางที่ 2

ฟังก์ชัน	สัญลักษณ์	การกลับสู่จุดเริ่มต้นโดยIFC
source handshake	SH	ได้
acceptor handshake	AH	ได้
talker or enlarge talker	T or TE	ได้
listener or enlarge listener	L or 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 1.2

Remote / Local

remote เป็นการกำหนดให้อุปกรณ์ที่อยู่ในระบบเช่น เครื่องมือวัดให้อยู่ในการควบคุมของอุปกรณ์ของอุปกรณ์ตัวอื่นแทน ซึ่งปุ่มปรับต่างๆบนหน้าปัดเครื่องจะไม่มีผลต่อการทำงาน ส่วน local เป็นการควบคุมการทำงานของเครื่องมือวัดด้วยปุ่มปรับบนหน้าปัดตามปกติ

การใช้ remote มีประโยชน์ในแง่ที่ขณะที่ตัวการควบคุมเช่น คอมพิวเตอร์กำลังติดต่ออุปกรณ์ตัวนั้นอยู่ หากไม่มีการตัดการควบคุมโดยปุ่มปรับบนหน้าปัดออก ถ้ามีใครมาปรับแต่งก็จะทำให้การทำงานผิดพลาดไปได้ การทำงานของ GPIB ใน remote and local มี 4 ลักษณะดังนี้

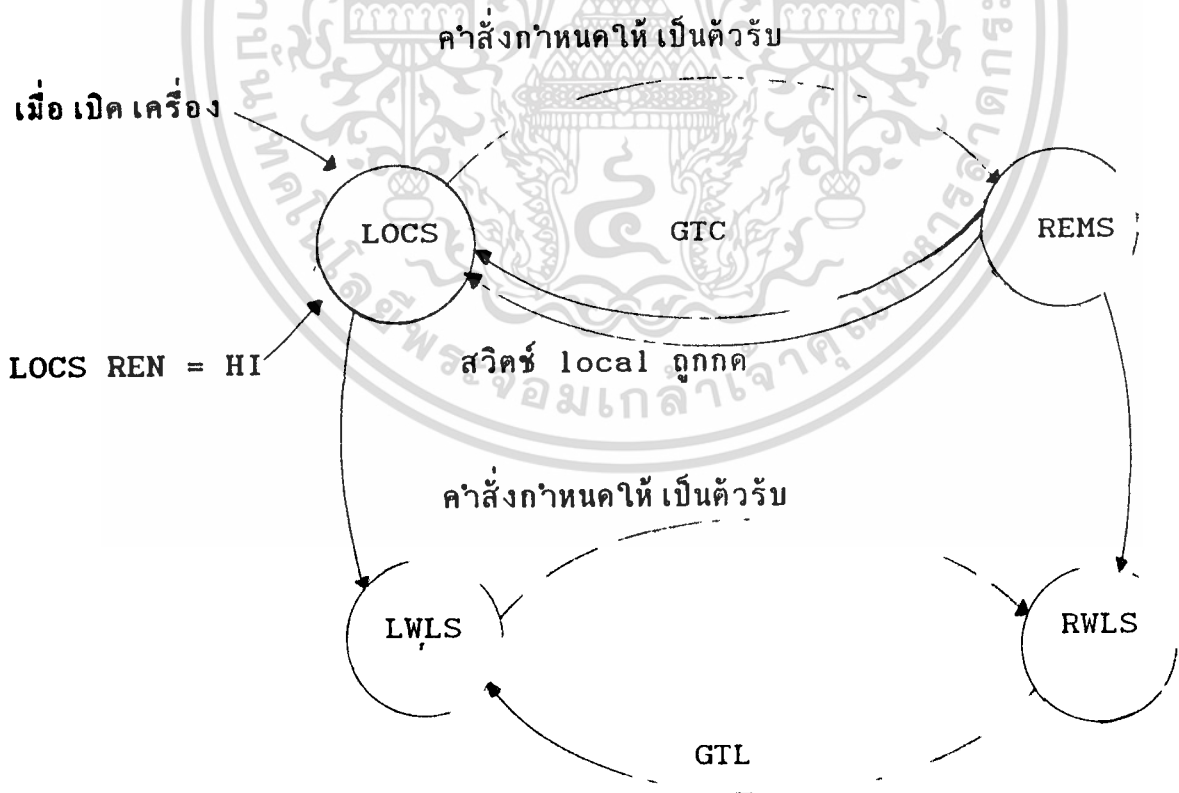
1. LOCS ก็คือ local นั่นเอง เป็นสภาพการควบคุมที่ปุ่มตามปกติ จะอยู่ในสภาพนี้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตอนเปิดเครื่อง หรือ REN เป็น HIGH หรือเมื่อได้รับคำสั่ง GTL

2. REMS คือ remote หมายถึงการตัดการควบคุมโดยปุ่มหน้าปัดออก จะเกิดขึ้นเมื่อ REN เป็น LOW และจะถูกล็อกไว้ เว้นแต่ว่าสวิทช์ local ที่ตัวอุปกรณ์จะถูกเปลี่ยนไปตำแหน่ง local

3. RWLS เป็นสภาพ remote ที่ถูกล็อกเอาไว้เช่นกัน แต่ว่าจะตัดการควบคุมตรงสวิทช์ local ที่ตัวอุปกรณ์ออกไป สภาพremoteโดย RWLS จึงมีความสำคัญสูงกว่า REMS อย่างไรก็ตามยังถูกยกเลิกได้ด้วยคำสั่ง LLO

4. LWLS มีสภาพเช่นเดียวกับ local แต่จะแตกต่างกันตรงที่สภาพlocal โดย LWLSนี้ เมื่อได้รับคำสั่งกำหนดอุปกรณ์ตัวรับจะเปลี่ยนไปอยู่ในสภาพแบบล็อกหรือ RWLS ทันทีในการที่จะมาที่สภาพ LWLS นี้ได้ก็มี 2 กรณีคือ เมื่ออยู่ในสภาพlocalธรรมดา (LOCS) แล้วได้รับคำสั่ง LLO หรืออยู่ใน RWLS แล้วได้รับคำสั่ง GTL



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การขอบริการและการตรวจสอบ (Service Request and Polling)

เมื่อตัวควบคุมได้รับ SRQ เป็น LOW จะให้อุปกรณ์ส่งข้อมูลแสดงสถานะการทำงานซึ่งมีอยู่ 2 วิธีคือ

1. การตรวจสอบแบบอนุกรม ซึ่งมีขั้นตอนดังนี้

1.1 ATN ถูกดึงเป็น LOW หลังจากได้รับ LOW จากสายสัญญาณ SRQ

1.2 คำสั่ง UNL ถูกส่งไปยังอุปกรณ์

1.3 ตัวควบคุมจะแจ้งรหัสตัวรับของตน และกำหนดรหัสตัวส่งของอุปกรณ์

ที่จะตรวจสอบไปที่บัส

1.4 ตามด้วยคำสั่ง SPE และสาย ATN กลายเป็น HI ซึ่งอุปกรณ์ที่ถูกเรียกจะส่งข้อมูลสถานะออกมา 1 ไบต์โดยบิตที่ 7 จะเป็นตัวชี้ว่าอุปกรณ์เส้นเป็นตัวขอบริการ ถ้าใช่จะเป็น LOW ส่วนบิตอื่นๆ ก็ใช้บอกข้อมูลอื่นๆซึ่งมิได้กำหนดเฉพาะ

1.5 สาย ATN ถูกดึงเป็น LOW อีกที เพื่อส่งคำสั่งยกเลิกการตรวจสอบคือ SPD

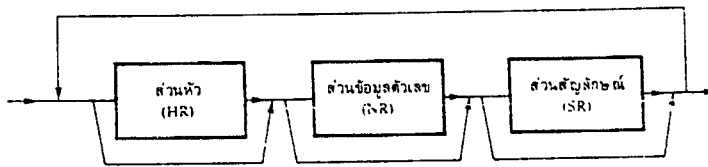
1.6 จากนั้นคำสั่ง UNT ก็ถูกส่งไปยังอุปกรณ์เพื่อยกเลิกการเป็นตัวส่งซึ่งถ้าหาก SQR ยังคงเป็น LOW อยู่ ก็จะมีการตรวจสอบไปยังอุปกรณ์ตัวอื่นๆต่อไปตามขั้นตอนเดิม

2. การตรวจสอบแบบขนาน สามารถทำได้เร็วกว่าแบบอนุกรมทั้งนี้เพราะสามารถอ่านข้อมูล เพราะสามารถอ่านข้อมูลเพียงไบต์เดียวก็สามารถรู้ได้ทันที ว่าอุปกรณ์ตัวใดเป็นผู้ขอบริการ

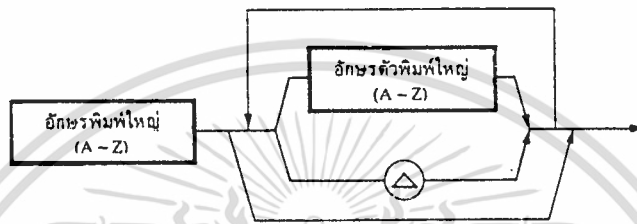
รูปแบบของข้อมูล

โดยทั่วไปข้อมูลจากอุปกรณ์ (device message) แบ่งได้ออกเป็น 3 ส่วนดังแสดงในรูปที่ 9 อันได้แก่ส่วนหัว (HR) ซึ่งจะอยู่ส่วนหน้าสุด เป็นตัวบอกชนิดข้อมูล ส่วนประกอบ HR แสดงในรูปที่ 10 จะเห็นว่าประกอบด้วยตัวอักษรภาษาอังกฤษตัวพิมพ์ใหญ่ช่องว่าที่เป็นเว้นวรรค () ปกติจะมีอักษรประมาณ 1-3 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

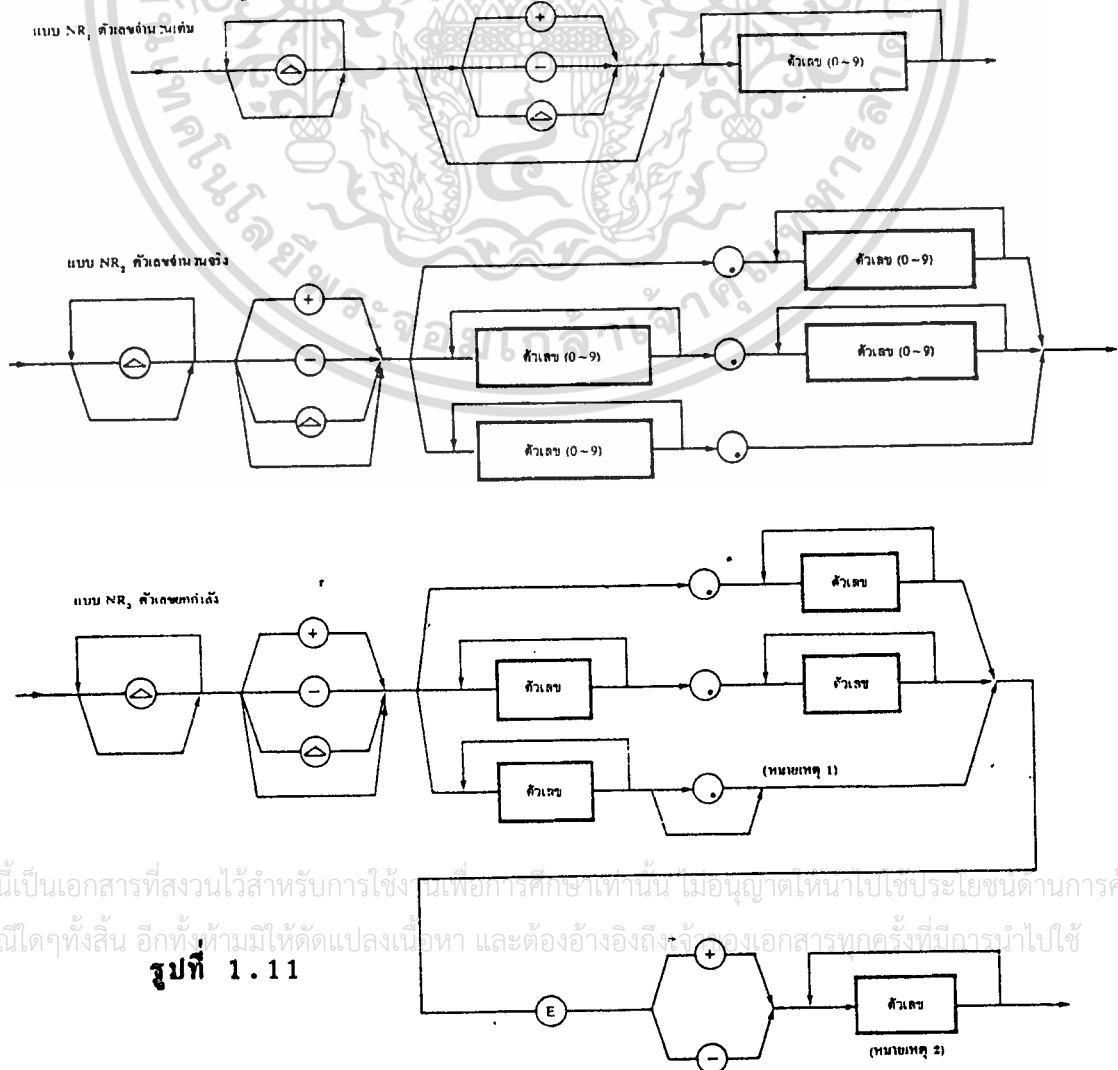


รูปที่ 1.9



รูปที่ 1.10

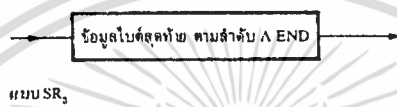
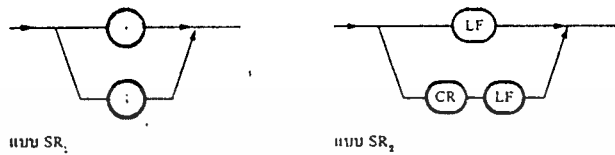
ส่วนที่สองคือเนื้อหาข้อมูล (NR) ซึ่งใช้แสดงค่าตัวเลข มีอยู่ 3 แบบ คือ NR1, NR2 และ NR3 ดังแสดงในรูปที่ 11 ส่วนท้ายของ NR ยังอาจมีตัวอักษรแสดงหน่วยตามมา



รูปที่ 1.11

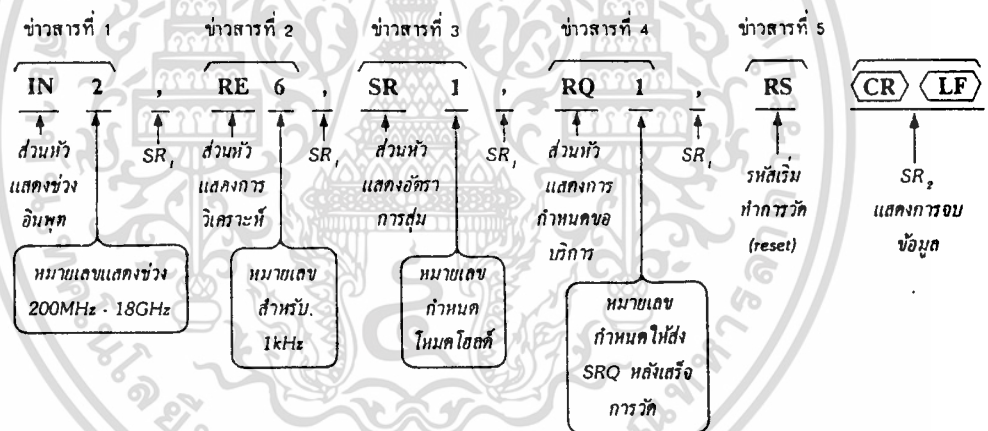
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตเห็นไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนที่ 3 คือสัญญาณแบ่งข้อมูลแต่ละชุด (SR) ดังแสดงในรูปที่ 12 โคน SR1 ใช้แสดงการต่อเนื่องของข้อมูล (ข้อมูลยังมีต่อ) SR2 และ SR3 แสดงการสิ้นสุดของข้อมูล แต่ SR3 เป็นการบอกการเสร็จสิ้นข้อมูลทั้งหมดจากการวัด และรูปที่ 13 เป็นตัวอย่างข้อมูลสำหรับการกำหนดฟังก์ชันให้เครื่องวัดความถี่และข้อมูลความถี่ที่วัดได้



รูปที่ 1.12

ตัวอย่างข้อมูลสำหรับการกำหนดฟังก์ชัน



ตัวอย่างข้อมูลเอาต์พุตที่ได้จากการวัด



รูปที่ 1.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 การออกแบบวงจร

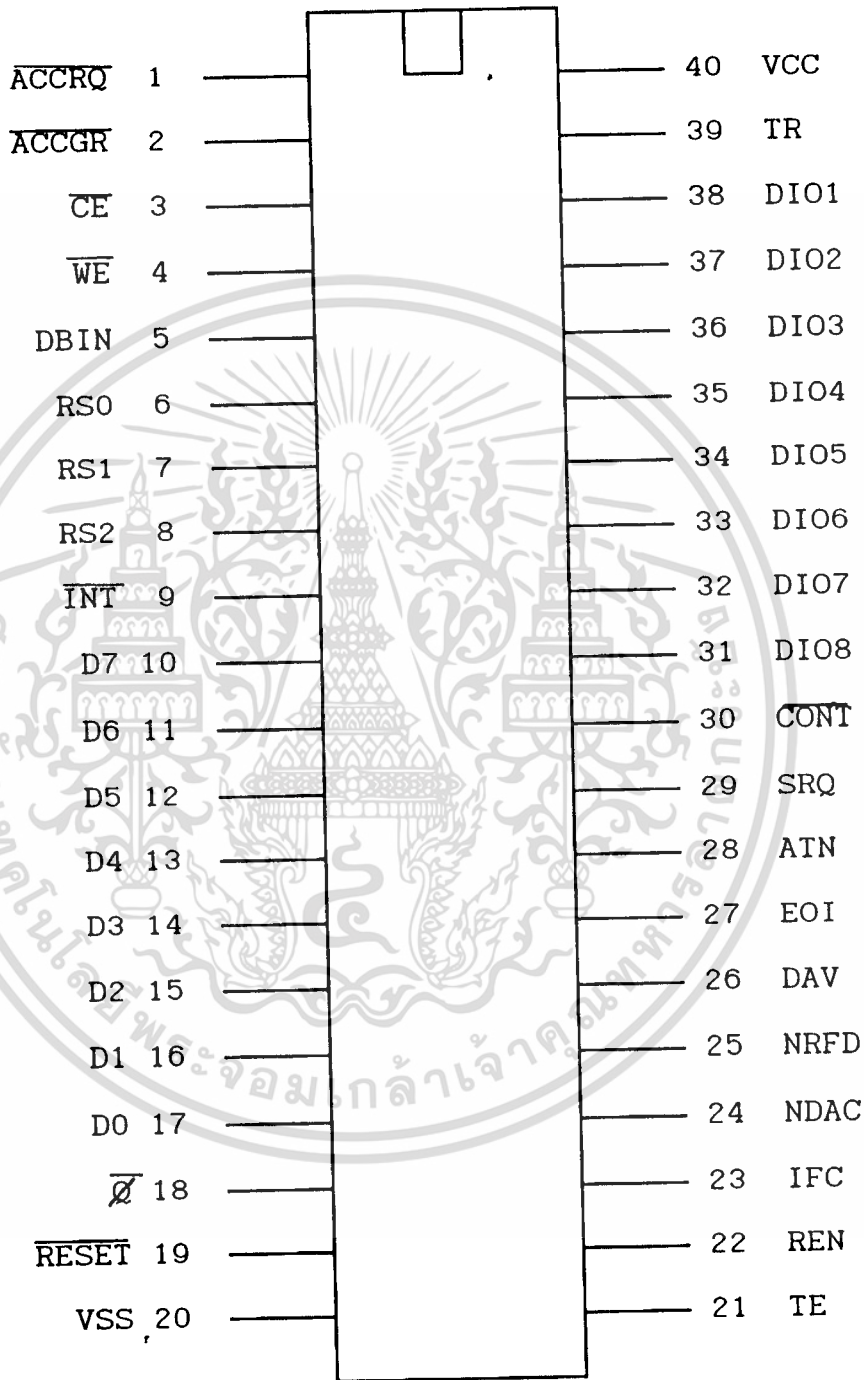
IC TMS9914A

IC TMS9914A เป็น IC ที่ทางบริษัท TEXAS INSTRUMENT จำกัด ออกแบบมาเพื่อการใช้ทำหน้าที่เป็น CPU ของ CARD GPIB INTERFACE ซึ่งวงจรภายในของประกอบด้วยกลุ่ม REGISTER ซึ่งออกแบบมาให้ใช้ฟังก์ชันมาตรฐานของ IEEE-488 ทำให้ TMS9914A เป็น IC ที่มีความเหมาะสมสำหรับการประยุกต์ใช้งาน IEEE-488 ได้เป็นอย่างดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะภายนอกและการจัดขาสัญญาณต่าง ๆ ของ TMS9914A



รูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 รายละเอียดของขาสัญญาณต่าง ๆ ของ IC TMS9914A

SIGNAL NAME	PIN NO.	I/O TYPE	SIGNAL DEFINITION
DIO8 (MSB)	31	I/O (p/p)	INPUT/OUTPUT DATA 1-8
DIO7	32	I/O (p/p)	
DIO6	33	I/O (p/p)	
DIO5	34	I/O (p/p)	
DIO4	35	I/O (p/p)	
DIO3	36	I/O (p/p)	
DIO2	37	I/O (p/p)	
DIO1 (LSB)	38	I/O (p/p)	
DAV	26	I/O (p/p)	DATA VALID
NRFD	25	I/O (p/p)	NOT READY FOR DATA
NDAC	24	I/O (p/p)	NOT DATA EXCEPTED
IFC	23	I/O (o/d)	INTERFACE CLEAR
REN	22	I/O (o/d)	REMOTE ENABLE
TE	21	O (p/p)	TALK ENABLE
CONT	30	O (p/p)	
SRQ	29	I/O (p/p)	SERVICE REQUEST
ATN	28	I/O (p/d)	ATTENTION
EOI	27	I/O (p/p)	END OR IDENTIFY
\overline{CE}	3	I	CHIP ENABLE
\overline{WE}	4	I	WRITE ENABLE
DBIN	5	I	DATA BUS IN
INT	9	O(o/d)	INTERRUPT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 รายละเอียดของขาสัญญาณต่าง ๆ ของ IC TMS9914A (ต่อ)

SIGNAL NAME	PIN NO.	I/O TYPE	SIGNAL DEFINITION
D0 (MSB)	17	I/O (p/p)	DATA 0-7
D1	16	I/O (p/p)	
D2	15	I/O (p/p)	
D3	14	I/O (p/p)	
D4	13	I/O (p/p)	
D5	12	I/O (p/p)	
D6	11	I/O (p/p)	
D7 (LDB)	10	I/O (p/p)	
RS0	6	I	REGISTER SELECT LINE
RS1	7	I	
RS2	8	I	
$\overline{\text{ACCRO}}$	1	O(p/p)	ACCESS REQUEST : DMA
$\overline{\text{ACCGR}}$	2	I	ACCESS GRANT
$\overline{\text{RESET}}$	19	I	RESET
TR	39	O(p/p)	TRIGGER
$\overline{\text{Q}}$	18	I	CLOCK : 500KHz ~ 5MHz
VSS	20		0 V
VCC	40		+5 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันการทำงานของ REGISTER ภายใน IC TMS9914A

ตารางที่ 2.3 สภาวะการอ่านหรือรับข้อมูลจาก IC TMS9914A

REGISTER SELECT LINE			STATUS OF DATA PIN							
RS2	RS1	RS0	D0	D1	D2	D3	D4	D5	D6	D7
0	0	0	INT0	INT1	BI	BO	END	SPAS	RLC	MAC
0	0	1	GET	ERR	UNC	APT	DCAS	MA	SRQ	IFC
0	1	0	REM	LLO	ATN	LPAS	TPAS	LADS	TADS	ulpa
0	1	1	ATN	DAV	NDAC	NRFD	EOI	SRQ	IFC	REN
1	0	0	NOT USE							
1	0	1	NOT USE							
1	1	0	DIO8	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1
1	1	1	DIO8	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 สภาวะการเขียนหรือส่งข้อมูลไปยัง IC TMS9914A

REGISTER SELECT LINE			STATUS OF DATA PIN							
RS2	RS1	RS0	D0	D1	D2	D3	D4	D5	D6	D7
0	0	0	X	X	BI	BO	END	SPAS	RLC	MAC
0	0	1	GET	ERR	UNC	APT	DCAS	MA	SRQ	IFC
0	1	0	NOT USE							
0	1	1	\bar{C}/S	X	X	F4	F3	F2	F1	F0
1	0	0	edpa	dal	dat	A5	A4	A3	A2	A1
1	0	1	S8	RSV	S6	S5	S4	S3	S2	S1
1	1	0	PP8	PP7	PP6	PP5	PP4	PP3	PP2	PP1
1	1	1	DIO8	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1

จากตารางพบว่าในสถานะการเขียนข้อมูลไปยัง IC TMS9914A ค่าของ REGISTER ตัวที่ 3 (RS2=0,RS1=1,RS0=1) เป็นการส่งคำสั่งมาตรฐานของ IEEE-488 นั้นเอง โดยที่คำสั่งต่าง ๆ มีความสัมพันธ์กับค่า \bar{C}/S , F4, F3, F2, F1, F0 ตามตาราง

ตารางที่ 2.5 ฟังก์ชันมาตรฐานของ IC TMS9914A

C/S	F4	F3	F2	F1	F0	MNEMONIC	FUNCTION
0/1	0	0	0	0	0	swrst	Chip reset
0/1	0	0	0	0	1	dacr	Release ACDS holdoff
na	0	0	0	1	0	rhdf	Release RFD holdoff
0/1	0	0	0	1	1	hdfa	Holdoff on all data
0/1	0	0	1	0	0	hdre	Holdoff on EOI ONLY
na	0	0	1	0	1	nbafe	Set new byte available false
0/1	0	0	1	1	0	fget	Force group execute trigger
0/1	0	0	1	1	1	rtl	Return to local
na	0	1	0	0	0	feoi	Send EOI with next byte
0/1	0	1	0	0	1	lon	Listen only
0/1	0	1	0	1	0	ton	Talk only
na	0	1	0	1	1	gts	Go to standby
na	0	1	1	0	0	tca	Take control asynchronously
na	0	1	1	0	1	tcs	Take control synchronous
0/1	0	1	1	1	0	rpp	Request parallel poll
0/1	0	1	1	1	1	sic	Send interface clear
0/1	1	0	0	0	0	sre	Sent remote enable
na	1	0	0	0	1	rqc	Request control
na	1	0	0	1	0	rlc	Release control
0/1	1	0	0	1	1	dai	Disable all interrupts
na	1	0	1	0	0	pts	Pass through next secondary

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.6 ฟังก์ชันมาตรฐานของ IC TMS9914A

C/S	F4	F3	F2	F1	F0	MNEMONIC	FUNCTION
0/1	1	0	1	0	1	std1	Set T1 delay
0/1	1	0	1	1	0	shdw	Shadow handshake
0/1	1	0	1	1	1	vstd1	Very short T1 delay
0/1	1	1	0	0	0	rsv2	Request service Bit2

จากคุณสมบัติของ IC TMS9914A สามารถนำมาออกแบบส่วนอินเทอร์เฟซระบบของ GPIB IEEE-488 เข้ากับตัวคอนโทรลเลอร์ซึ่งใช้ PC เป็นตัวคอนโทรลเลอร์ ได้ดังวงจรในรูปที่ 2.2 ซึ่งแยกอธิบายเป็นส่วน ๆ ได้ดังนี้

วงจรส่วน INTERFACE กับเครื่องมือวัด

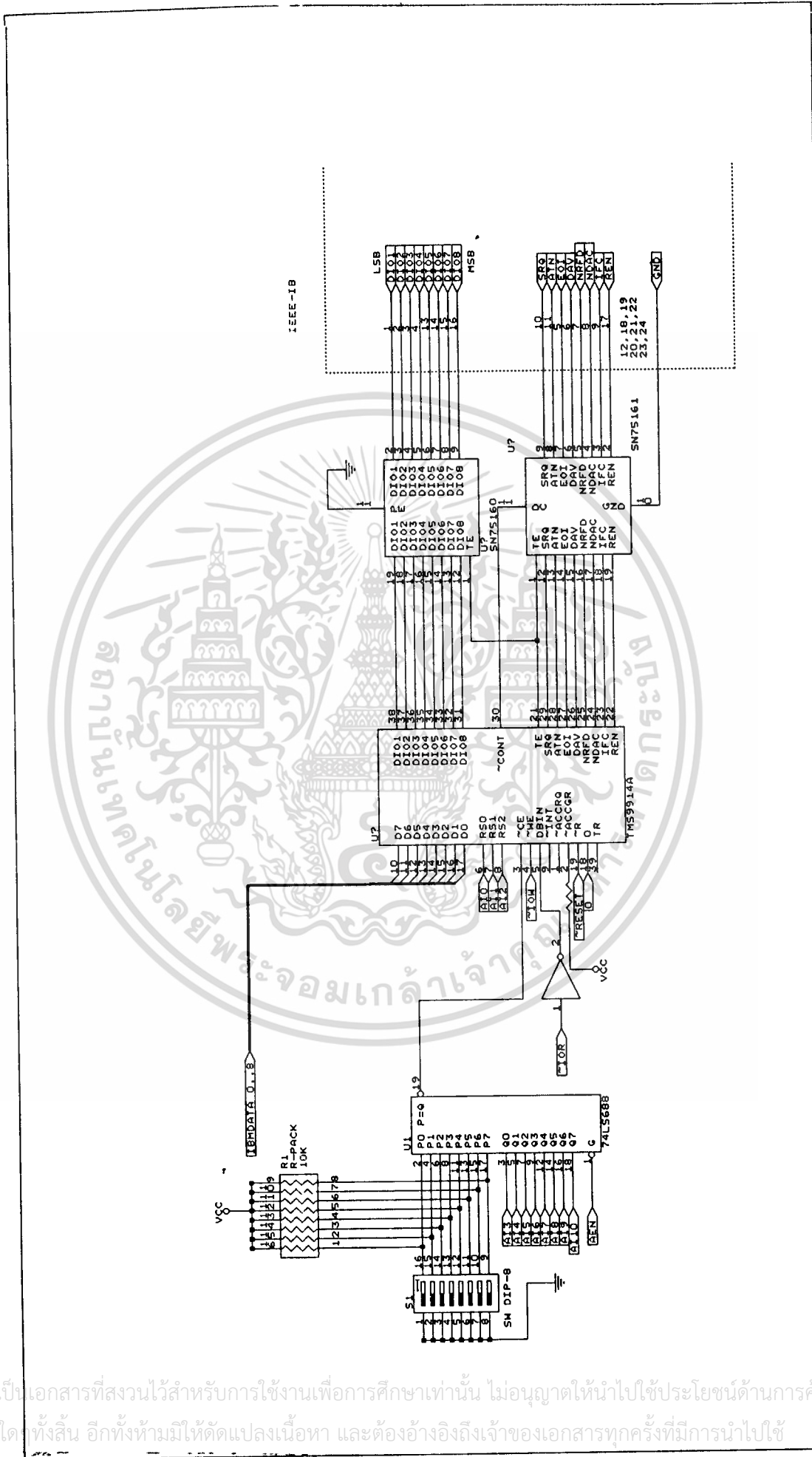
จากวงจรในรูปที่ 2.2 เป็นวงจรส่วนที่ INTERFACE กับเครื่องมือวัด หรือตัวอุปกรณ์อื่นที่ทำหน้าที่เป็นตัว LISTENER ในระบบ IEEE-488 โดยในวงจร IC TMS9914 ทำหน้าที่เป็น CPU ของส่วน INTERFACE นี้และ IC SN75160 , SN75161 ทำหน้าที่เป็นบัฟเฟอร์ของระบบ ซึ่งทำให้ CARD สามารถต่อพ่วงกับตัว LISTENER ได้ทั้งหมด 32 ตัว เอาท์พุทของบัฟเฟอร์จะต่อเข้ากับ CONNECTOR ซึ่งจะต่อออกสู่ระบบ IEEE-488

วงจรส่วนการ DECODE ADDRESS จาก PC

วงจรส่วนแรกประกอบด้วย IC 74LS688, 8 Bit DIP Switch และ RESISTOR ขนาด 10 K เป็นวงจรที่ทำการดีโค๊ดกลุ่มแอดเดรสขนาด 8 แอดเดรส ซึ่งการเลือกกลุ่มแอดเดรสที่จะทำการดีโค๊ดนี้ จะทำได้โดยการเซ็ท DIP Switch ที่ขา Q0-Q7 ของ IC 74LS688

สำหรับหน้าที่ของ 74LS688 นี้จะทำการเปรียบเทียบค่าของอินพุต 2 ชุด ที่ถูกส่งเข้ามาทางขา P0-P7 และขา Q0-Q7 ถ้าอินพุตทั้ง 2 ชุดนี้เท่ากันแล้ว เอาท์พุทที่ขา P=Q จะให้เอาท์พุทเป็นลอจิก "0" จากในวงจรขา P0-P7 ของ 74LS688 ต่อกับแอดเดรสบิต A3-A10 ในขณะที่ขา Q0-Q7 ต่อกับความต้านทานที่ทำหน้าที่เป็น Pull Up (รักษาระดับแรงดันให้เป็นลอจิก "1" ไว้ในกรณีที่ไม่มีอินพุตใดๆ เข้ามา) และ ขา Q0-Q7 นี้จะต่อกับปลายข้างหนึ่งของ DIP Switch ด้วย ส่วนปลายอีกข้างหนึ่งของ DIP Switch ที่ต่อกับขาใดขานั้นก็จะได้รับลอจิก "0" ในขณะที่ถ้า DIP Switch ที่ต่อกับขาใดถูก "OFF" ขานั้นก็จะได้รับลอจิก "1" และ เนื่องจากอินพุทที่ขา P0-P7 (แอดเดรส A3-A10) ต้องเท่ากับอินพุทที่ขา Q0-Q7 ดังนั้นถ้าเราเปลี่ยนแปลงการเซ็ท DIP Switch เหล่านี้ก็จะทำให้แอดเดรสบิต A3-A5 ซึ่งต่อกับขา P0-P7 นั้นต้องเปลี่ยนแปลงค่าแอดเดรสที่ต้องการจะดีโค๊ดได้ง่ายกว่าวิธีการดีโค๊ดแบบ Fixed ค่าของแอดเดรส

วงจรในลักษณะนี้เราสามารถจะนำไปใช้เป็นการดีโค๊ดในแบบ Fixed ได้โดยการนำเอา DIP Switch ออก จากนั้นถ้าอินพุทใดต้องการลอจิก "0" จึงจะใช้ตัวนำเชื่อมต่อระหว่างขั้วทั้งสองแทนการเซ็ท DIP Switch ให้ "ON" แต่ถ้าอินพุทใดต้องการลอจิก "1" ก็ปล่อยขั้วทั้งสองนั้นไว้



รูปที่ 2.2

ELECTRONIC DEPARTMENT, FACULTY OF ENGINEER	
Title	IEEE-488 CARD
Size Document Number	33100047
REV	1.0
Date	January 30, 1994 Sheet 1 of 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 ผลการทดลอง

ในโครงการนี้ใช้ CARD IEEE-488 ในการควบคุมเครื่องมือวัดอันได้แก่ DIGITAL MULTIMETER model TR6845 ของบริษัท ADVANCETEST , SPECTRUM ANALYZER model R4131 SERIES ของบริษัท ADVANCETEST , INTRON DIGITAL OSCILLOSCOPE model DSO-2022 ของบริษัท INTRON INSTRUMENTS ซึ่งในการควบคุมเครื่องมือเหล่านี้จำเป็นต้องทราบ CODE คำสั่งในการควบคุม การ SET PARAMETER ต่าง ๆ ของเครื่องมือวัด ตลอดจนการอ่านข้อมูลจากอุปกรณ์เหล่านี้ ซึ่งตารางแสดง CODE คำสั่งต่าง ๆ เหล่านี้ได้รวบรวมไว้ตามตารางเหล่านี้

ตารางแสดง CODE ต่าง ๆ ของ DIGITAL MULTIMETER model TR6845

ตารางที่ 3.1 Measuring Function Selecting Command Codes

Codes	Function	Initial value
F1	DC voltage measurement (VDC)	0
F2	AC voltage measurement (VAC)	
F3	Resistance measurement (OHM)	
F4	Low power ohm measurement (L.P.OHM)	
F5	DC current measurement (ADC)	
F6	AC current measurement (AAC)	
F7*1	Temperature measurement (TMP, C) Continuity test	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 Measuring Function Selecting Command Codes (ต่อ)

Codes	Function	Initial value
F8	AC+DC voltage measurement (VAC+VDC) P-P voltage measurement	
F9	AC+DC current measurement (AAC+ADC) P-P current measurement	

ตารางที่ 3.2 Measuring Range Selecting Command Codes

Funt Code	VDC	VAC VAC(AC+DC)	OHM	L.P.OHM	ADC,AAC AAC(AC+DC)	Initial value
R0	AUTO	AUTO	AUTO	AUTO	AUTO	0
R1	- *2	- *2	- *2	- *2	3uA *3	
R2	30 mV	- *2	30 OHM	- *2	30uA *3	
R3	300 mV	300 mV	300 OHM	300 OHM	300 uA	
R4	3000 mV	3000 mV	3000 OHM	3000 OHM	3000 uA	
R5	30 mV	30 V	30 KOHM	30 KOHM	30 mA	
R6	300 V	300 V	300 KOHM	300 KOHM	300 mA	
R7	1000 V	750 V	3000 KOHM	3000 KOHM	3000 mA *4	
R8	- *2	- *2	30 MOHM	30 MOHM	10 A *4	
R9	- *2	- *2	300 MOHM	- *2	- *2	

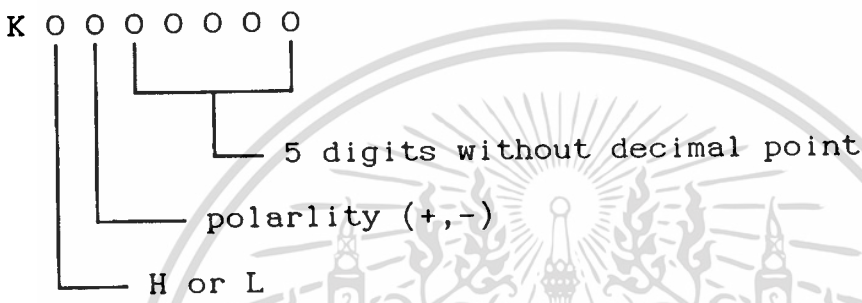
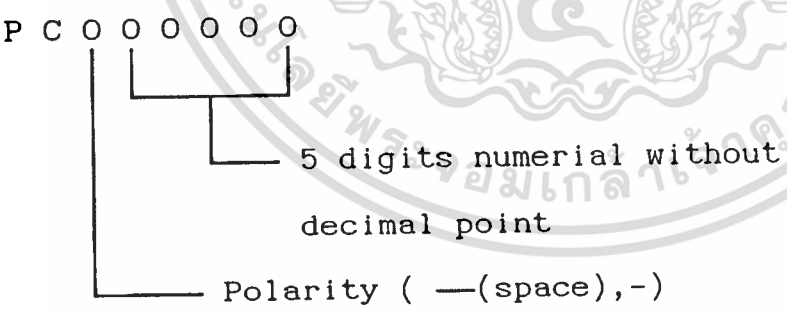
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 Function Selecting Command Code

Function	Code	Initial value	Remarks
Null Calculation ON OFF	NL1 NL0	0	Even if "NL1" is received while executing NULL calculation the multimeter continues measurement without starting calculation over again
Sampling mode FREE RUN HOLD	M0 M1	0	
Comparator Calculation ON OFF	CO1 CO0	0	
Buzer ON OFF	BZ1 BZ0	0	
Sampling rate FAST MID LOW	PR1 PR2 PR3	0	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 Function selecting command codes (Cont'd)

Function	Code	Initial value	Remarks
H and L constant setting K 0 0 0 0 0 0 0 		Initial value 0	
Header	ON OFF PH1 PH0	0	
Calibration P C 0 0 0 0 0 0 			Valid only when CAL ON key is set ON A syntax error will result if calibra- tion data is out of range
Delimeter mode CR LF and EOI is output LF alone is output EOI alone is output	DL0 DL1 DL2	0	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 Function selecting command codes (Cont'd)

Function	Code	Initial value	Remarks
Service request To send out Not to send out	S0 S1	0	
Start of measurement	E		Valid is set to HOLD GET command is equivalent to this
Execution of equivalent routine when POWER is turn on	C		Execution is performed from the beginning of program. (Same as turning on of POWER) Equivalent to "DCL" and "SDC"
Parameter initialization	Z		Parameter is initialized("o" mark), and execution is performed from the beginning of program.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 Function selecting command codes (Cont'd)

Function		Code	Initial value	Remarks
Display	YES	DS0		7-segment display
	NO	DS1	0	ON/OFF

ตารางแสดง CODE ต่าง ๆ ของ SPECTRUM ANALYZER model R4131 SERIES

ตารางที่ 3.4 Frequency Span Set Value Codes

Code	SPAN	Code	SPAN	Code	SPAN
SP50KZ	50 kHz				
SP100KZ	100 kHz	SP10MZ	10 MHz	SP1GZ	1 GHz
SP200KZ	200 kHz	SP20MZ	20 MHz	SP2GZ	2 GHz
SP500KZ	500 kHz	SP50MZ	50 MHz	SP4GZ	4 GHz
SP1MZ	1 MHz	SP100MZ	100 MHz	ZS	ZEROSPAN
SP2MZ	2 MHz	SP200MZ	200 MHz		
SP5MZ	5 MHz	SP500MZ	500 MHz		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 Resolution Band Width Set Value Code

Code	Resolution band width	Code	Resolution band width
RB1KZ	1 KHz	RB100KZ	100 KHz
RB3KZ	3 KHz	RB300KZ	300 KHz
RB10KZ	10 KHz	RB1MZ	1 MHz
RB30KZ	30 KHz		

ตารางที่ 3.6 VIDEO Filter Band Width Set Value Code

Code	Value of VIDEO Filter Band Width
VF10Hz	10 Hz
VF100Hz	100 Hz
VF1KZ	1 kHz
VF10KZ	10 kHz
VF100KZ	100 kHz
VF300KZ	300 kHz
VF1MZ	1 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 Sweep Time Set Value Code

Code	Sweep time	Code	Sweep time
ST5MS	5 ms/	ST1S	1 s/
ST10MS	10 ms/	ST2S	2 s/
ST20MS	20 ms/	ST5S	5 s/
ST50MS	50 ms/	ST10S	10 s/
ST100MS	100 ms/	ST20S	20 s/
ST200MS	200 ms/	ST50S	50 s/

ตารางที่ 3.8 OUTPUT Parameter Codes

Code	Parameter output
AT	ATTENUATOR
CF	CENTER FREQUENCY
MF	MARKER FREQUENCY
ML	MARKER LEVEL
RB	RESOLUTION BAND WIDTH
RL	RESOLUTION LEVEL
SP	FREQ SPAN
ST	SWEEP TIME
VF	VIDEO FILTER BAND WIDTH
PL	DISPLAY LINE
OB	OCCUPIED BAND WIDTH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.9 Relation Between Output Data and Header

Type of output data .		Header
CENTER FREQUENCY		CF
SPAN		SP
REFERENCE LEVEL	dBm	DM
	dBu	DU
	dBu/m	VM
	LINEAR dBmV	LV DQ
SWEEP TIME/DIV		ST
RESOLUTION BAND WIDTH		RB
VIDEO FILTER		VF
ATT		AT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.9 Relation Between Output Data and Header (Cont'd)

Type of output data.		Header	
MARKER	FREQUENCY	MF	
	LEVEL	dBm	MM
		dBu	MU
		dBu/m	ME
		LINEAR	ML
		dBmV	MQ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางแสดง CODE ต่าง ๆ ของ INTRON OSCILLOSCOPE MODEL DSO-2022

ตารางที่ 3.10 INTRON OSCILLOSCOPE MODEL DSO-2022 DATA-FRAME FORMAT

Byte	Bit	7	6	5	4	3	2	1	0
0		— INV	— ADD	— XY	— MAG	D3	D2	D1	D0
1		X	X	X	X	A3	A2	A1	A0
2-13		NO DATA							
14-2061		CRT-DATA SELECTED CHANNEL							
2062-2079		NO DATA							
REMARK ***		— — — INV,ADD,XY and MAG are set according to the switch setting							

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.11 TIMEBASE SETTING FOR DSO-2022

A3	A2	A1	A0	Timebase setting
0	0	0	0	0.5 Sec/Div.
0	0	0	1	0.2 Sec/Div.
0	0	1	0	0.1 Sec/Div.
0	0	1	1	50 mSec/Div.
0	1	0	0	20 mSec/Div.
0	1	0	1	10 mSec/Div.
0	1	1	0	5 mSec/Div.
0	1	1	1	2 mSec/Div.
1	0	0	0	1 mSec/Div.
1	0	0	1	0.5 mSec/Div.
1	0	1	0	0.2 mSec/Div.
1	0	1	1	0.1 mSec/Div.
1	1	0	0	50 uSec/Div.
1	1	0	1	20 uSec/Div.
1	1	1	0	10 uSec/Div.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.12 VOLTS/DIV SETTING FOR DSO-2022

D3	D2	D1	D0	Volts/Div. selected channel
0	0	0	0	20 V/Div.
0	0	0	1	10 V/Div.
0	0	1	0	5 V/Div.
0	0	1	1	2 V/Div.
0	1	0	0	1 V/Div.
0	1	0	1	0.5 V/Div.
0	1	1	0	0.2 V/Div.
0	1	1	1	0.1 V/Div.
1	0	0	0	50 mV/Div.
1	0	0	1	20 mV/Div.
1	0	1	0	10 mV/Div.
1	0	1	1	5 mV/Div.
1	1	1	1	Channel off.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.13 Transferring data table

Bit	7	6	5	4	3	2	1	0
Mode								
slow data transfer (<12.5 Kbyte/sec)	1	x	x	x	x	x	x	x
fast data transfer (>12.5 Kbyte/sec)	0	x	x	x	x	x	x	x
Data channel 1	x	x	x	x	x	x	1	0
Data channel 2	x	x	x	x	x	x	0	1
<p>EXAMPLE *** If you want to read the data of channel 2 with a transferspeed > 12.5 Kbyte/sec. you must send the byte 0xxxxxx1 to the DSO-2022</p>								

***** คำสั่งในการอ่านข้อมูลที่วัดได้จาก OSCILLOSCOPE คือคำสั่ง A0 *****

การเขียนโปรแกรมควบคุม IEEE - 488 GPIB

ในการเขียนโปรแกรมควบคุม GPIB ในด้านการใช้งานจริงนั้น มีส่วนที่เกี่ยวข้อง 2 ส่วน คือ ส่วนสัญญาณควบคุม (control) และส่วนแฮนด์เชค (hand shake) ซึ่งรายละเอียดต่างๆ มีดังที่กล่าวมาแล้วก่อนหน้านี้ จะขอกล่าวในส่วนใช้งานจริงเท่านั้น

การควบคุมผ่าน GPIB นั้น ในโปรแกรมตัวอย่างมีการใช้ควบคุมอุปกรณ์ดังนี้

1. คอนโทรลเลอร์ (controller) เป็นส่วน interface card ที่ควบคุมผ่าน PC ซึ่งเป็นส่วนที่คอยประมวลผลการทำงานทุกอย่างของระบบนี้ โดยประกอบไปด้วยการควบคุมอุปกรณ์ผ่านทาง PC , การรับ-ส่งข้อมูลผ่านอุปกรณ์กับคอนโทรลเลอร์, การแสดงผล รวมไปถึง utility ที่ใช้ในการอำนวยความสะดวกในการใช้โปรแกรมกับการทำงานร่วมกับระบบ IEEE - 488 GPIB card ที่ใช้ สร้างขึ้นโดยใช้ IC TMS9914 เป็นตัวประมวลผลตามมาตรฐาน IEEE - 488 ซึ่งได้กล่าวถึงหน้าที่ของ IC TMS9914 มาแล้ว คอนโทรลเลอร์นี้เป็นได้ทั้งตัวรับ (listener) และ/หรือ ตัวส่ง (talker) ขึ้นอยู่กับการโปรแกรมควบคุม

2. อุปกรณ์ (device) เป็นส่วนที่ถูกควบคุมผ่านคอนโทรลเลอร์ โดยอุปกรณ์จะสามารถเป็นได้ทั้งตัวรับ (listener) และ/หรือ ตัวส่ง (talker) ขึ้นอยู่กับ function ของอุปกรณ์ตัวนั้นและการควบคุมผ่านทางคอนโทรลเลอร์

ขั้นตอนการเขียนโปรแกรมควบคุม IEEE - 488 GPIB

ดังที่ได้กล่าวไปแล้วว่า ในการเขียนโปรแกรมควบคุมตามมาตรฐาน IEEE - 488 นั้น จะมีขบวนการเฉพาะในการติดต่อกันระหว่างอุปกรณ์ทั้งหมดบนระบบ bus นั้น ซึ่งจะสามารถพิจารณาขบวนการดังกล่าวเป็น 2 ช่วงดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ช่วงที่สัญญาณ ATN เป็น Low

ช่วงนี้เป็นช่วงที่แสดงว่าข้อมูลที่ส่งลงบน bus เป็นคำสั่งซึ่งหมายถึงคำสั่งที่เป็นมาตรฐานของ IEEE-488 ซึ่งพิจารณาจากตารางที่ 1.1 code คำสั่ง ใน column แรก คำสั่งดังกล่าวส่วนมากเป็นคำสั่งที่ใช้กับระบบ เช่น DCL (14h) เป็นคำสั่งที่ใช้ในการ clear อุปกรณ์ที่ต่ออยู่บน bus ทุกตัว เป็นต้น ภายในช่วงที่สัญญาณ ATN เป็น Low นี้ จะสามารถใช้คำสั่งมาตรฐานเหล่านี้เท่าไรก็ได้ ความหมายของคำสั่งต่าง ๆ ได้กล่าวมาแล้ว จะไม่ขอกล่าวอีกครั้ง

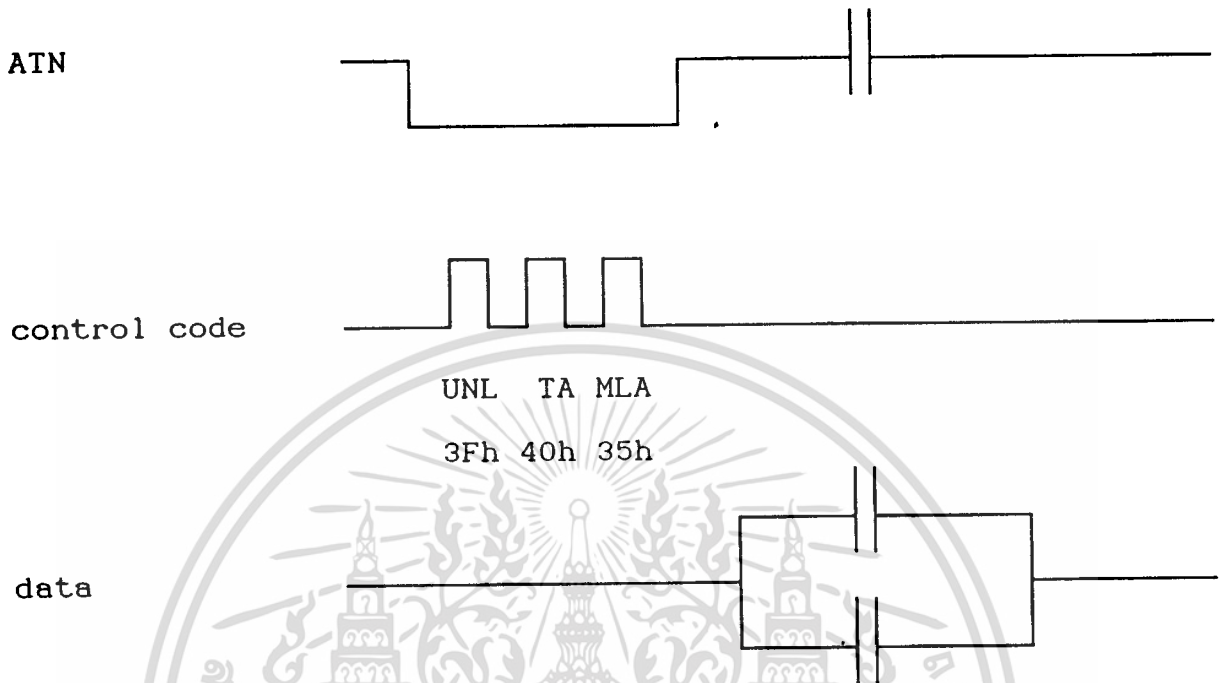
2. ช่วงที่สัญญาณ ATN เป็น High

ช่วงนี้เป็นช่วงที่แสดงว่าข้อมูลที่ส่งลงบน bus เป็นข้อมูลจริงๆ ซึ่งข้อมูลจะใช้มาตรฐาน ASCII ข้อมูลในช่วงนี้ส่วนมากจะใช้เป็น code คำสั่งในแต่ละอุปกรณ์ โดยที่แต่ละอุปกรณ์จะมีการออกแบบ code คำสั่งดังกล่าวแตกต่างกันออกไป ซึ่งจะทำให้อุปกรณ์สามารถมีการควบคุมได้หลาย ๆ อย่างด้วย code คำสั่งเหล่านี้ ยกตัวอย่างเช่น ใน digital multimeter ของบริษัท ADVANTEST รุ่น TR6845 มี code คำสั่ง F1 ถึง F8 ใช้ในการสั่งให้ตัวอุปกรณ์ทำการเปลี่ยน mode การวัดไปเป็นแบบต่าง ๆ ดังตารางที่ 3.1

ในการส่งข้อมูลดังกล่าวมาแล้ว ไม่ว่าจะ เป็นข้อมูลที่เป็นคำสั่งหรือเป็นข้อมูลจริงๆ ในกรณีที่อุปกรณ์สามารถส่งข้อมูลที่เป็นรายละเอียดต่างๆ เช่น ข้อมูลการวัด เป็นต้น การส่งข้อมูลนั้นจะต้องผ่านขบวนการในการตรวจสอบว่า ได้ส่งข้อมูลถูกต้องเรียบร้อยแล้ว หรือยัง ซึ่งขบวนการดังกล่าวก็คือ hand checking ในขบวนการ hand checking มีสัญญาณที่เกี่ยวข้องอยู่ 3 สัญญาณ คือ DAV (DATA VALID), NRFD (NOT READY FOR DATA) และ NDAC (NOT DATA ACCEPT) ตามที่ได้มีการอธิบายไปแล้ว

จากรูปจะแสดงถึง diagram ของการทำงานของระบบ ในการติดต่อกับ อุปกรณ์ทั้งตัวรับและตัวส่ง

- กรณีติดต่อกับตัวส่ง (talker)



เมื่อคอนโทรลเลอร์จะทำการติดต่อกับอุปกรณ์ที่เป็นตัวส่ง เช่น DVM ฯลฯ คอนโทรลเลอร์จะทำหน้าที่เป็นตัวรับ โดยที่เมื่อมีการทำงานดังนี้ คอนโทรลเลอร์จะทำการดึงสัญญาณ ATN เป็น Low เมื่อเป็น Low แล้ว จะทำการส่ง code คำสั่ง ซึ่งมีความหมายดังนี้

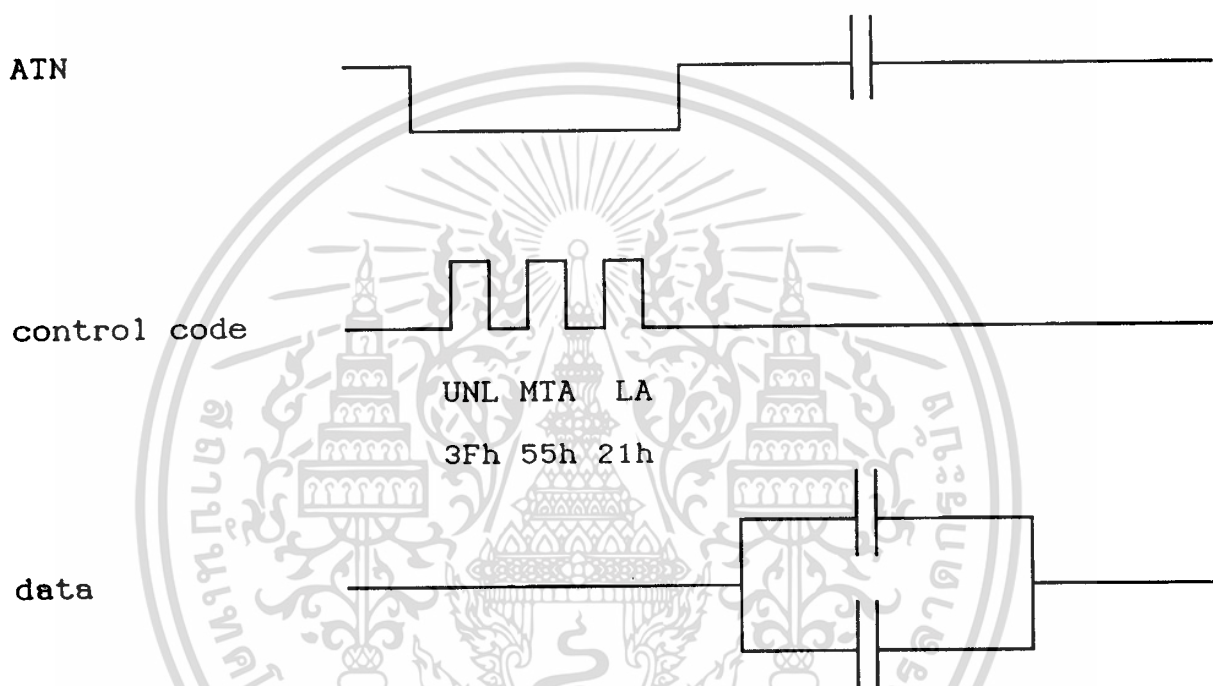
- UNL (3Fh) ระบบจะทำการ clear อุปกรณ์ทุกตัวออกจากการเป็นตัวรับ listener บนระบบ bus ดังนั้นตอนนี้จะไม่มีอุปกรณ์ใดทำหน้าที่เป็นตัวรับอยู่บน bus เลย

- TA (40h) คอนโทรลเลอร์ขอติดต่อกับอุปกรณ์ที่ทำหน้าที่เป็นตัวส่ง โดยที่อุปกรณ์ตัวนั้นมี address ที่ 0 (ในการติดต่อกับอุปกรณ์ที่เป็นตัวส่งจะทำการบวก address ของตัวส่งด้วย 40h ดังนั้น $0h + 40h = 40h$) อุปกรณ์ที่เป็นตัวส่ง address ที่ 0 ก็จะรับรู้ว่คอนโทรลเลอร์ติดต่อกำลังต้องการให้ส่งข้อมูล

- MLA (35h) คอนโทรลเลอร์บอก address ของตัวเองในการติดต่อกับอุปกรณ์ โดยจะทำหน้าที่เป็นตัวรับที่ address ที่ 21 (ในการติดต่อกับอุปกรณ์ที่เป็นตัวรับจะทำการบวก address ของตัวรับด้วย 20h ดังนั้น $21 + 20h = 35h$)

เมื่อสิ้นสุดคำสั่งต่างๆ คอนโทรลเลอร์จะทำการดึงสัญญาณ ATN ขึ้นเป็น High เป็นการสิ้นสุด code คำสั่ง ข้อมูลที่ตามมาในช่วงที่ ATN เป็น High จะเป็นข้อมูลจริงๆ ในรูปแบบของ ASCII CODE เช่น เป็นกลุ่มของข้อมูลที่ได้จากการวัดด้วย DVM เป็นต้น

- กรณีติดต่อกับตัวรับ (listener)



เมื่อคอนโทรลเลอร์จะทำการติดต่อกับอุปกรณ์ที่เป็นตัวรับ เช่น DVM ฯลฯ คอนโทรลเลอร์จะทำหน้าที่เป็นตัวส่ง โดยที่มมีการทำงานดังนี้ คอนโทรลเลอร์จะทำการดึงสัญญาณ ATN เป็น Low เมื่อเป็น Low แล้ว จะทำการส่ง code คำสั่ง ซึ่งมีความหมายดังนี้

- UNL (3Fh) ระบบจะทำการ clear อุปกรณ์ทุกตัวออกจากการเป็นตัวรับ listener บนระบบ bus ดังนั้นตอนนี้จะไม่มีอุปกรณ์ใดทำหน้าที่เป็นตัวรับอยู่บน bus เลย

- MTA (55h) คอนโทรลเลอร์บอก address ของตัวเองในการติดต่อกับอุปกรณ์ โดยจะทำหน้าที่เป็นตัวส่งที่ address ที่ 21 (ในการติดต่อกับอุปกรณ์ที่เป็นตัวส่งจะทำการบวก address ของตัวรับด้วย 40h ดังนั้น $21 + 40h = 55h$)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- LA (21h) คอนโทรลเลอร์ขอติดต่อกับอุปกรณ์ที่ทำหน้าที่เป็นตัวรับ โดยที่อุปกรณ์ตัวนั้นมี address ที่ 1 (ในการติดต่อกับอุปกรณ์ที่เป็นตัวรับจะทำการบวก address ของตัวรับด้วย 20h ดังนั้น $1 + 20h = 21h$) อุปกรณ์ที่เป็นตัวรับ address ที่ 1 ก็จะรับรู้ว่าคอนโทรลเลอร์ติดต่อกมาต้องการให้รับข้อมูล

เมื่อสิ้นสุดคำสั่งต่างๆ คอนโทรลเลอร์จะทำการดึงสัญญาณ ATN ขึ้นเป็น High เป็นการสิ้นสุด code คำสั่ง ข้อมูลที่ตามมาในช่วงที่ ATN เป็น High จะเป็นข้อมูลจริงๆ ในรูปแบบของ ASCII CODE เช่น เป็นกลุ่มของข้อมูลที่ใช้สั่ง DVM ให้ทำงานตามคำสั่ง เป็นต้น

คำสั่งมาตรฐานในช่วงที่สัญญาณ ATN เป็น Low นี้ อาจจะมีมากหรือน้อยกว่านี้ และอาจจะมีการเรียงลำดับการใช้งานแตกต่างกันไปแล้วแต่การออกแบบ

ในการส่งข้อมูลแต่ละครั้งไม่ว่าจะเป็น code คำสั่ง หรือข้อมูลจริงๆ จะต้องมีการ hand checking ทุกครั้ง ตาม flow chart และที่ได้อธิบายมาแล้ว

โดยทั่วไปแล้ว ข้อมูลที่ใช้ส่งไปยังอุปกรณ์ในช่วงที่อุปกรณ์ทำหน้าที่เป็นตัวรับนั้น ส่วนมากจะเป็น code คำสั่งที่ไปทำการสั่งงานอุปกรณ์โดยเฉพาะอีกครั้ง ไม่ได้เป็นคำสั่งมาตรฐาน แต่ในบางครั้งคำสั่งดังกล่าวก็อาจมีหน้าที่ในการทำงานเหมือนกับเช่น เป็นการ trig อุปกรณ์ ซึ่งในคำสั่งมาตรฐานก็มีใช้เช่นกัน

การเขียนโปรแกรมควบคุม IEEE - 488 GPIB โดย IC TMS9914

การเขียนโปรแกรมควบคุม GPIB โดยใช้ IC TMS9914 นั้นเป็นการเขียนโปรแกรมที่ใช้คุณสมบัติของ IC TMS9914 ซึ่งปกติแล้วการเขียนโปรแกรมควบคุมตามมาตรฐาน IEEE 488 GPIB นั้นจะมีขบวนการติดต่อกับอุปกรณ์ ทั้งที่เป็น listener และ talker รวมไปถึงขบวนการ hand checking ในการรับส่งข้อมูลกับอุปกรณ์ต่างๆ บนระบบ bus ตามที่ได้กล่าวมาแล้วข้างต้นเรื่อง IC TMS9914 มีความสามารถในการจัดการขบวนการดังกล่าว โดยอาศัยวงจรที่ออกแบบมาในตัว IC ดังตารางที่ 2.3, ตารางที่ 2.4 , ตารางที่ 2.5 , และตารางที่ 2.6

สภาวะการอ่าน/เขียนข้อมูลจาก register ในตัว IC TMS9914 ทำให้เราสามารถตรวจสอบสภาวะของระบบได้ตลอดเวลา ทำให้ขบวนการต่างๆ ในการติดต่อสื่อสารผ่านระบบ IEEE 488 มีความสะดวกขึ้น

ในการใช้งานปกตินั้น register ที่จำเป็นในการใช้งานมีดังนี้

1. register อ่านข้อมูลตัวที่ 1 (RS2 RS1 RS0 = 0 0 0) มีขนาด 8 bit (1 byte) ประกอบไปด้วย bit ต่างๆ ดังนี้

D0 - INT 0

D1 - INT 1

D2 - BI

D3 - BO

D4 - END

D5 - SPAS

D6 - RLC

D7 - MAC

ซึ่ง bit ที่จำเป็นต้องใช้ในการ hand checking ได้แก่ bit D2 และ D3 ซึ่งมีความหมาย คือ bit D2 - BI นั้นเป็น bit ที่ใช้ในการอ่านข้อมูลจากอุปกรณ์ที่เป็น talker เมื่อสามารถอ่านข้อมูลจากอุปกรณ์ได้เรียบร้อยแล้ว จะมีการ set ค่าที่ bit D2 - BI นี้ เช่นเดียวกันกับ bit D3 - BO ก็มีความหมายเช่นเดียวกัน แต่เป็นกระบวนการในการเขียนข้อมูลไปยังอุปกรณ์ที่เป็น listener เมื่อมีการเขียนข้อมูลเสร็จเรียบร้อยแล้วก็จะมีการ set ค่าที่ bit D3 - BO เช่นกัน

2. register อ่านข้อมูลตัวที่ 7 (RS2 RS1 RS0 = 1 1 1) มีขนาด 8 bit (1 byte) ประกอบไปด้วย bit ข้อมูลทั้ง 8 bit ใช้ในการอ่านข้อมูลที่ต้องการจาก bus ข้อมูลตามที่ต้องการ

3. register เขียนข้อมูลตัวที่ 3 (RS2 RS1 RS0 = 0 1 1) มีขนาด 6 bit ประกอบไปด้วย bit ต่างๆ ดังนี้

D0 - C/S

D3 - f4

D4 - f3

D5 - f2

D6 - f1

D7 - f0

register นี้มีความจำเป็นในการเขียนโปรแกรมควบคุมเป็นอย่างมาก เป็น register ที่ทำหน้าที่เป็น function ดังตาราง function มาตรฐาน วิธีเขียนควบคุม function นี้ก็โดยการเขียนไปที่ register ตามค่า f0 - f4 ตามตาราง IC TMS9914 ก็จะทำการประมวลผลตาม function ที่ตั้งไว้เช่น function 'sre' จะทำการส่งสัญญาณ remote ไปยังอุปกรณ์ที่กำหนด เป็นต้น

4. register เขียนข้อมูลตัวที่ 7 (RS2 RS1 RS0 = 1 1 1) มีขนาด 8 bit (1 byte) ประกอบไปด้วย bit ข้อมูลทั้ง 8 bit ใช้ในการเขียนข้อมูลที่ต้องการลงบน bus ข้อมูลตามที่ต้องการ

function ของ IC TMS9914 ที่จำเป็นต้องใช้มีดังนี้

1. swrst (ChiP reset)
2. rhdf (Release RFD holdoff)
3. hdfa (Holdoff on all data)
4. rtl (Return to local)
5. lon (Listen only)
6. ton (Talk only)
7. gts (Go to standby)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. sic (Send interface clear)
9. sre (Send remote enable)
10. rqc (Request control)
11. rlc (Release control)

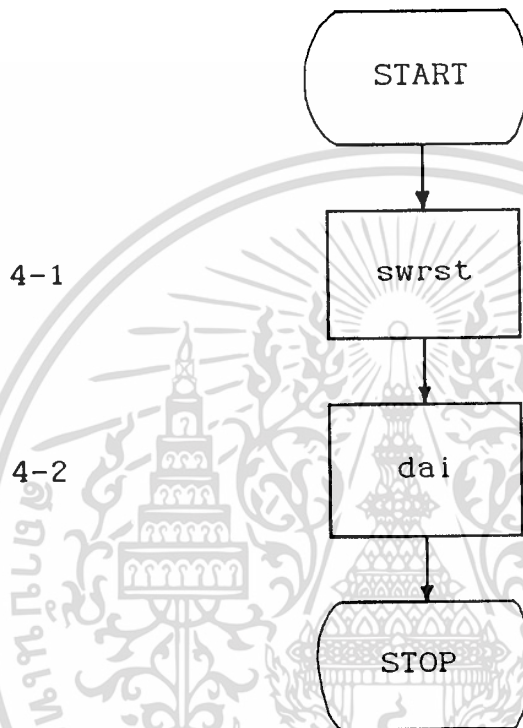


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

Principle of programming

Initial controller card

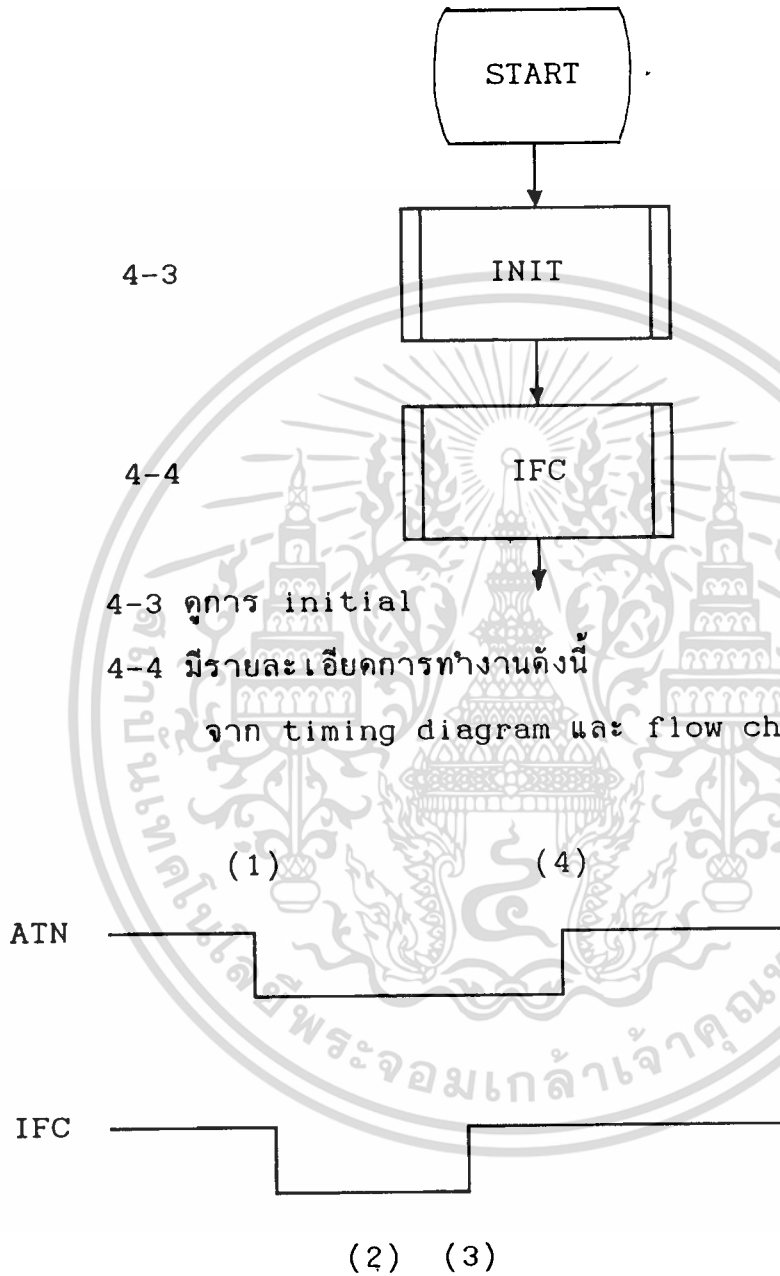


4-1 Chip reset

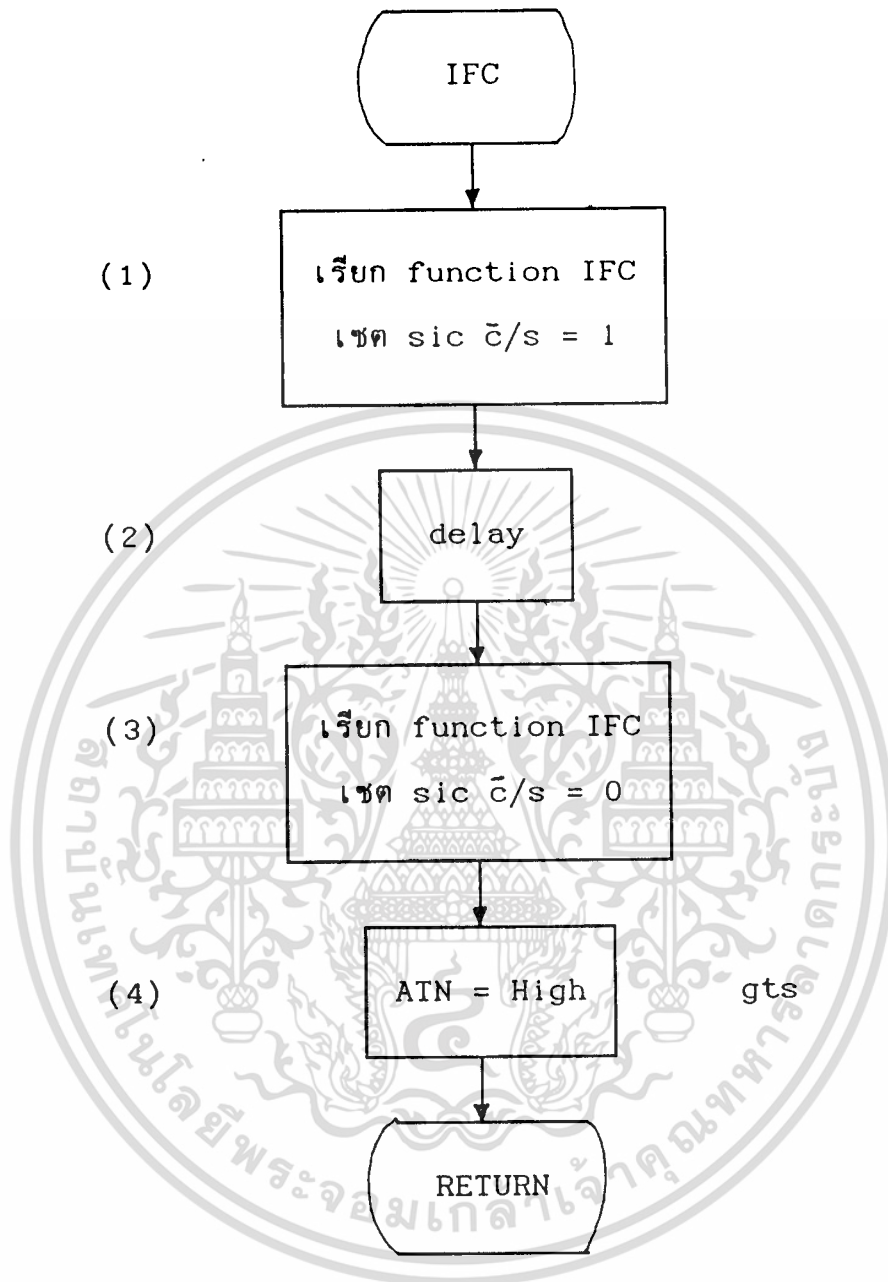
รีเซ็ต TMS 9914 เรียกใช้ function F4 F3 F2 F1 F0 = 0 0 0 0 0 0 (Chip reset) โดยเปลี่ยนบิต c/s จาก 1 ไป 0

4-2 Disable all interrupts เรียกใช้ function F4 F3 F2 F1 F0 = 1 0 0 1 1 (Disable all interrupts)

นอกจากนั้นยังสามารถเพิ่มค่าเริ่มต้นต่างๆ ตามความต้องการใช้งานได้

IFC (Interface Clear)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(1) Send interface clear

เรียกใช้ function F4 F3 F2 F1 F0 = 0 1 1 1 1 โดยเซต
บิต c/s เป็น 1 สำหรับการทำสัญญาณ IFC เป็น Low

(2) หน่วงเวลา

(3) Send interface clear

เรียกใช้ function F4 F3 F2 F1 F0 = 0 1 1 1 1 โดยเซต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต \bar{c}/s เป็น 0 สำหรับการทำสัญญา IFC เป็น High

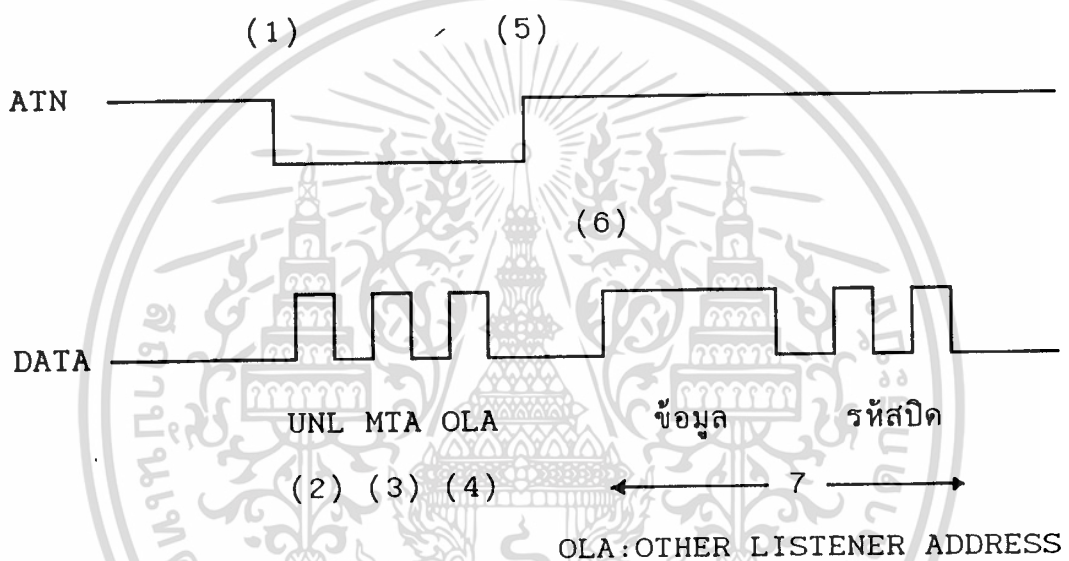
(4) Go to standby

เรียกใช้ function F4 F3 F2 F1 F0 = 0 1 0 1 1 เป็น

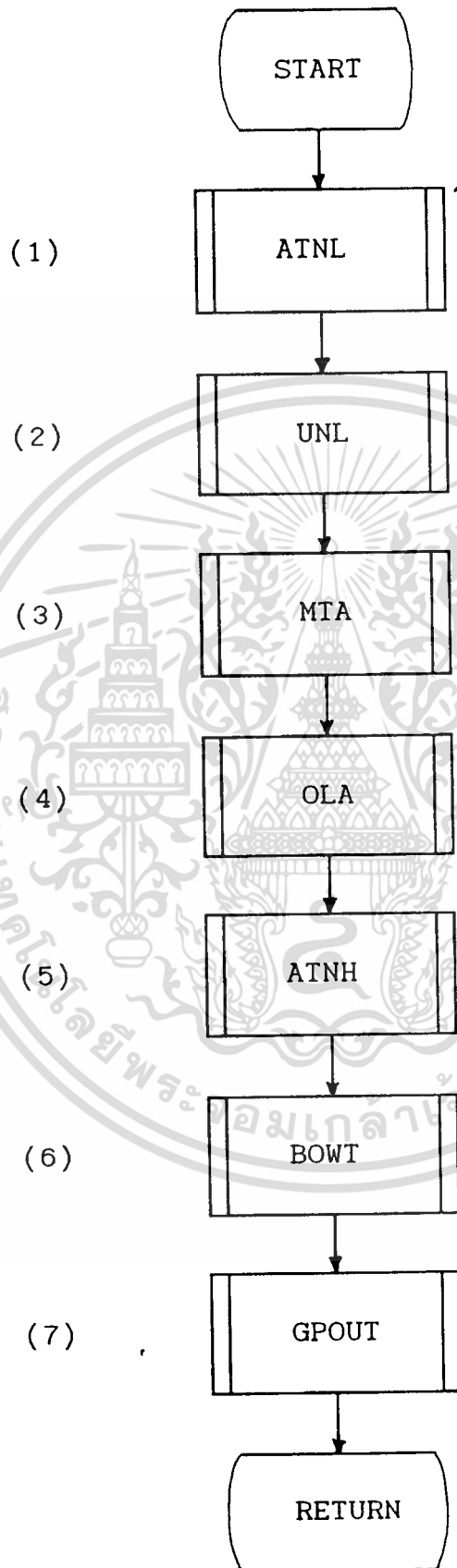
การกำหนดขาสัญญา ATN ให้มีสถานะเป็น High

การส่งข้อมูลจาก controller

จาก timing diagram และ flow chart



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

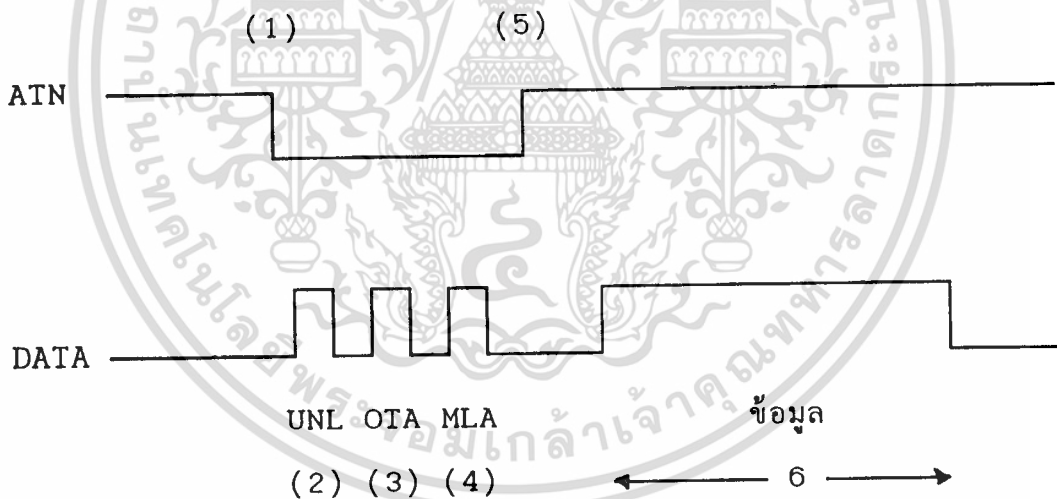


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

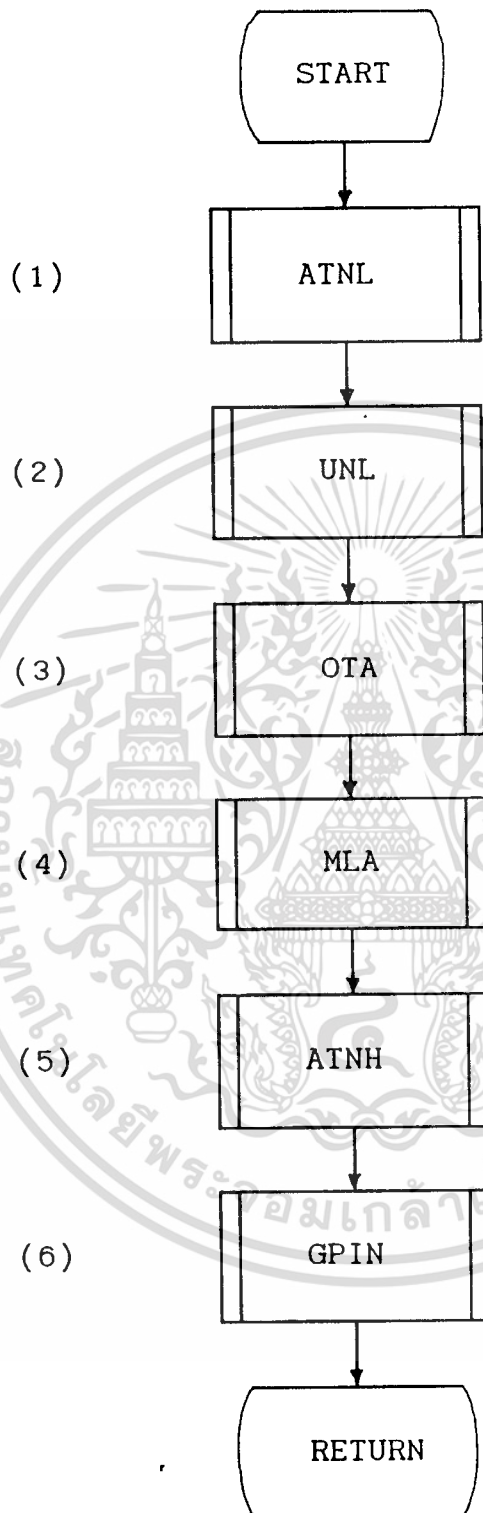
- (1) ทำ ATN เป็น Low
- (2) Unlistener เป็นการ clear address ของอุปกรณ์ที่เป็น listener ทุกตัวออกจากระบบบัส
- (3) My talker address กำหนด controller เป็น talker (ผู้ส่ง)
- (4) Other listener address กำหนด address ของ listener
- (5) ทำ ATN เป็น High
- (6) รอสถานะ (Hand shake)
- (7) ส่งข้อมูลออก

การรับข้อมูลเข้า controller

จาก timing diagram และ flow chart



OTA: OTHER TALKER ADDRESS



(1) ทำ ATN เป็น Low

(2) Unlistener เป็นการ clear address ของอุปกรณ์ที่เป็น listener เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ner ทุกตัวออกจากระบบบัส

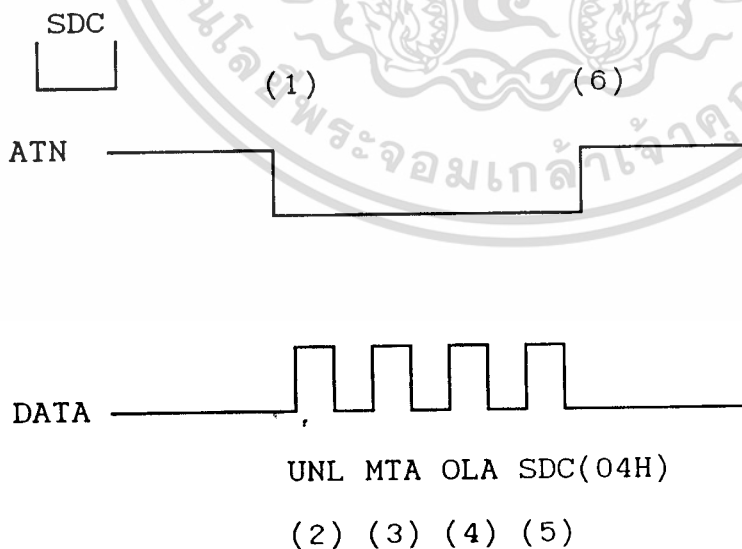
- (3) Other talker address กำหนด controller เป็น listener (ผู้รับ)
- (4) My listener address กำหนด address ของอุปกรณ์ที่เป็น talker (ผู้ส่ง)
- (5) ทำ ATN เป็น High
- (6) รับข้อมูลเข้า controller

SDC/DCL (Select Device Clear/Device Clear)

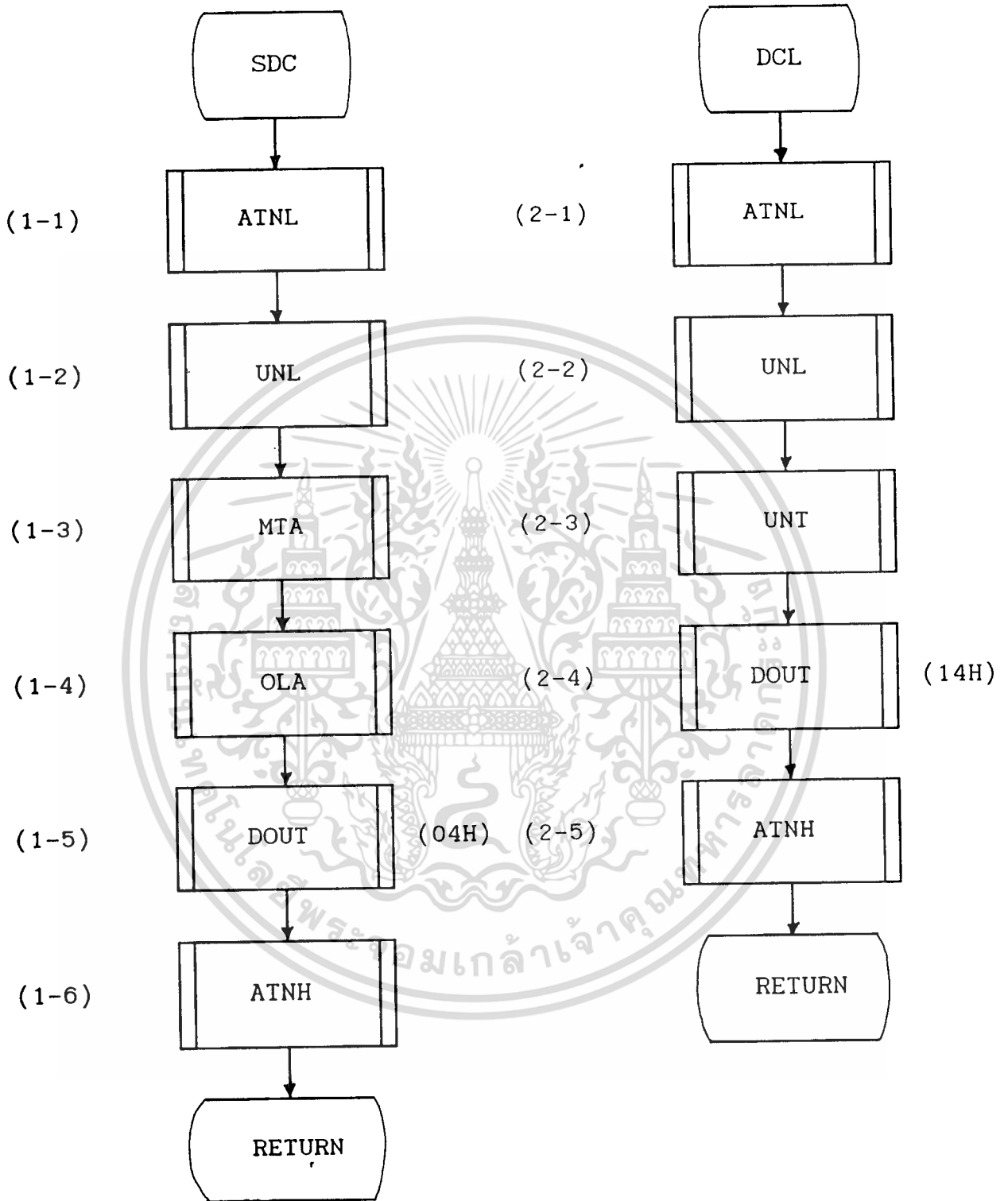
เป็นคำสั่งที่มีในกลุ่มคำสั่งมาตรฐานของ IEEE 488 แต่ละอุปกรณ์อาจจะมีคำสั่งนี้เป็นของตนเองได้ รายละเอียดและการใช้งานมีดังนี้

- SDC เป็นการเลือกการ clear อุปกรณ์ตาม address ที่กำหนด
- DCL เป็นการ clear อุปกรณ์ทุกตัวที่ต่ออยู่บนบัส

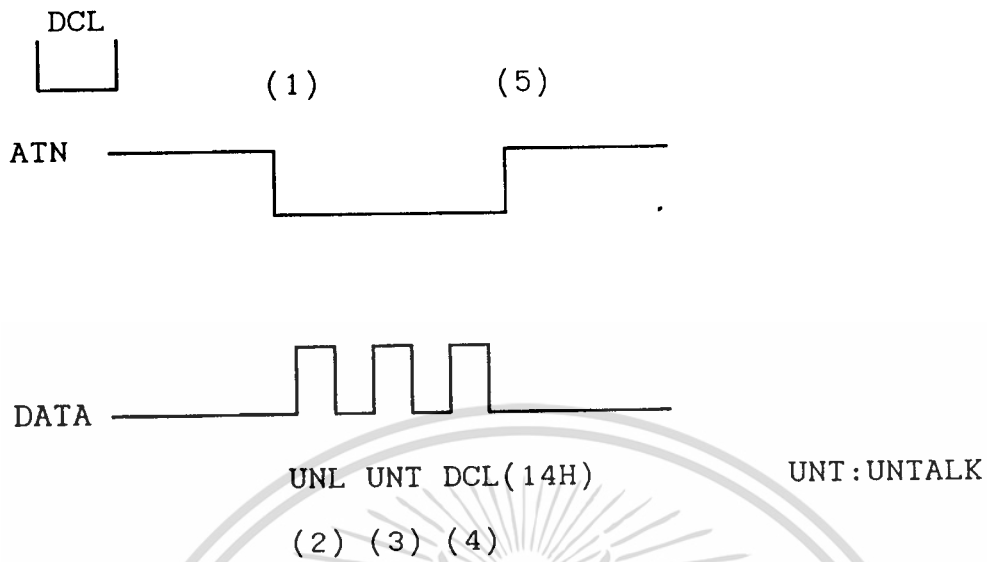
timing diagram และ flow chart



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



SDC

- (1-1) ทำ ATN เป็น Low
- (1-2) Unlistener เป็นการ clear address ของอุปกรณ์ที่เป็น listener ทุกตัวออกจากระบบบัส
- (1-3) My talker address กำหนด controller เป็น talker (ผู้ส่ง)
- (1-4) Other listener address กำหนด address ของอุปกรณ์ที่ต้องการ clear
- (1-5) ส่งโค้ดคำสั่ง SDC ซึ่งได้แก่ 04H ไปยังอุปกรณ์ที่กำหนด
- (1-6) ทำ ATN เป็น High

DCL

- (2-1) ทำ ATN เป็น Low
- (2-2) Unlistener เป็นการ clear address ของอุปกรณ์ที่เป็น listener ทุกตัวออกจากระบบบัส
- (2-3) Untalker เป็นการ clear address ของอุปกรณ์ที่เป็น talker ทุกตัวออกจากระบบบัส
- (2-4) ส่งโค้ดคำสั่ง DCL ซึ่งได้แก่ 14H ไปยังอุปกรณ์ทุกตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

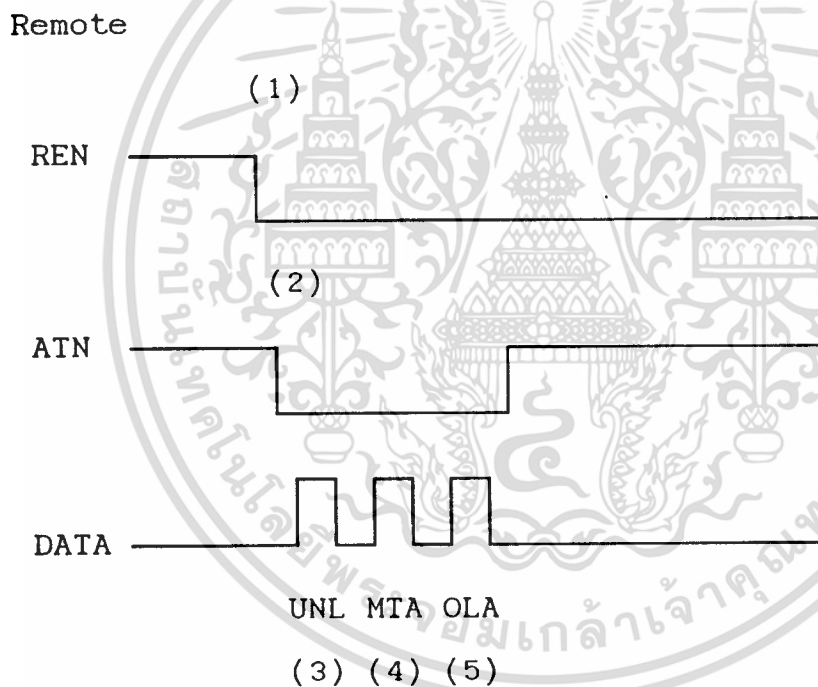
(2-5) ทำ ATN เป็น High

REN/GL(Remote Enable/Go to Local)

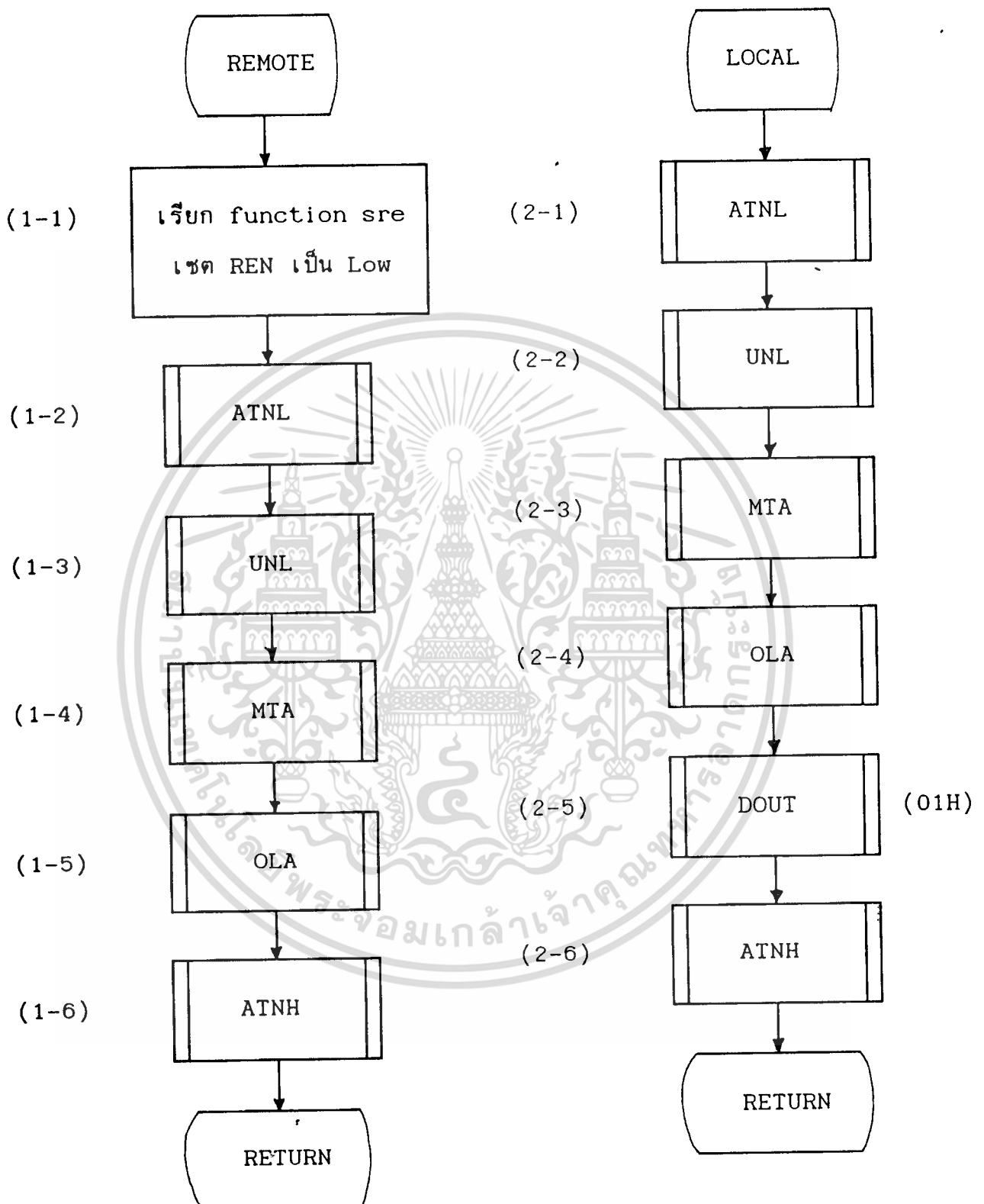
เป็นการกำหนดโหมดการทำงานของอุปกรณ์

- Remote เป็นการสั่งอุปกรณ์ให้ถูกควบคุมจาก controller
- Go to Local เป็นการสั่งอุปกรณ์ให้ควบคุมจากหน้าปัดปกติ

timing diagram และ flowchart

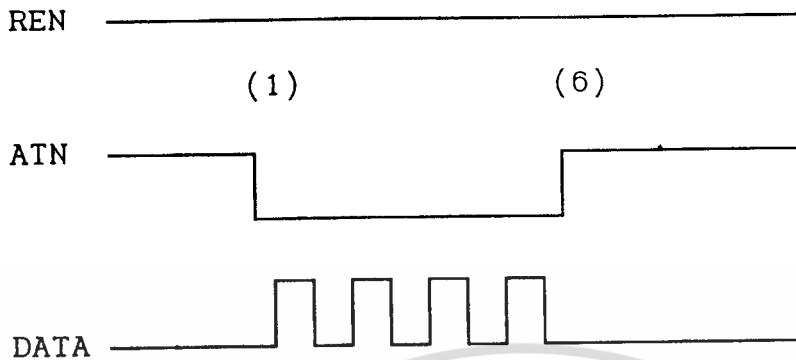


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Go to Local



UNL MTA OLA GL(01H)

(2) (3) (4) (5)

Remote

- (1-1) เรียกใช้ function sre (Send remote enable) เพื่อตั้งขาสัญญาณ remote ให้เป็น Low
- (1-2) ทำ ATN เป็น Low
- (1-3) Unlistener เป็นการ clear address ของอุปกรณ์ที่เป็น listener ทุกตัวออกจากระบบบัส
- (1-4) My talker address กำหนด controller เป็น talker (ผู้ส่ง)
- (1-5) Other listener address กำหนด address ของอุปกรณ์ที่ต้องการติดต่อในระบบ remote
- (1-6) ทำ ATN เป็น High

Go to Local

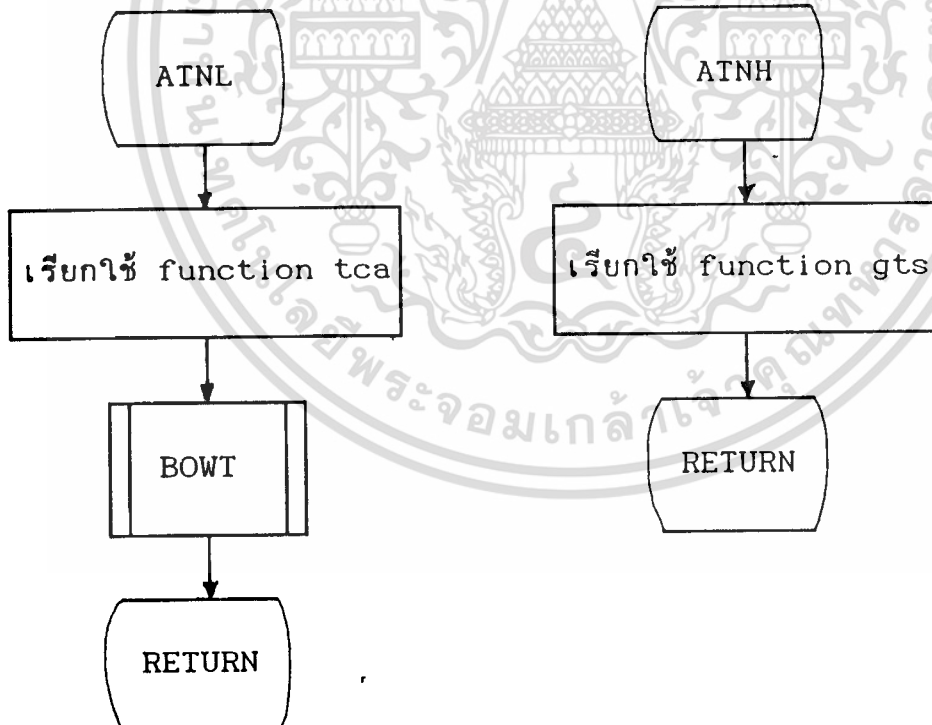
- (2-1) ทำ ATN เป็น Low
- (2-2) Unlistener เป็นการ clear address ของอุปกรณ์ที่เป็น listener ทุกตัวออกจากระบบบัส
- (2-3) My talker address กำหนด controller เป็น talker (ผู้ส่ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

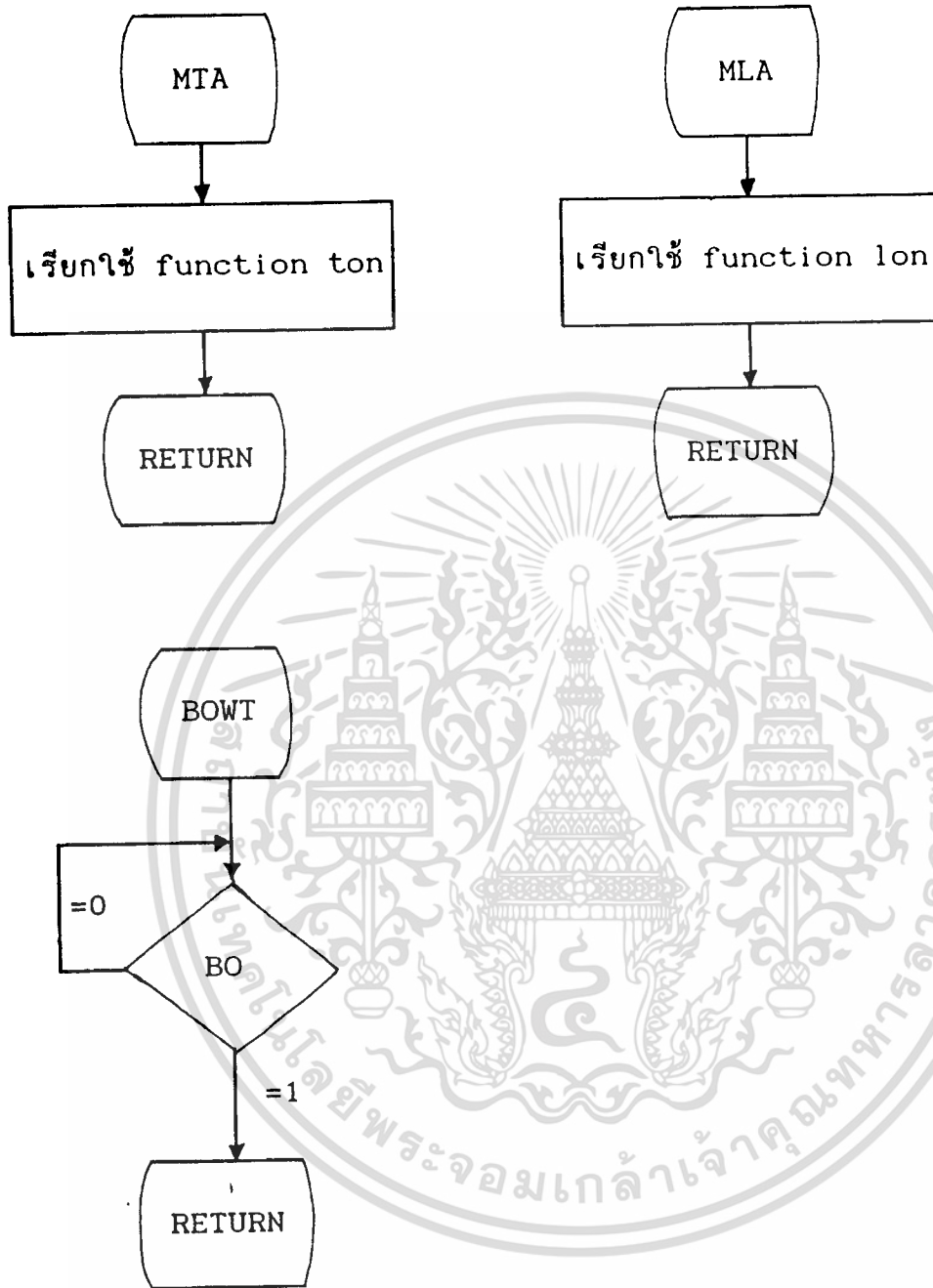
- (2-4) Other listener address กำหนด address ของอุปกรณ์ที่ต้องการติดต่อในระบบ Local
- (2-5) ส่งโค้ดคำสั่ง GL ซึ่งได้แก่ 01H ไปยังอุปกรณ์ทุกตัว
- (2-6) ทำ ATN เป็น High

เช่นเดียวกัน ในการปฏิบัติโค้ดคำสั่งที่เป็นคำสั่งมาตรฐานนั้นก็สามารถใช้หลักการปฏิบัติเดียวกันตามที่ยกตัวอย่างมา ในกรณีที่โค้ดคำสั่งเป็นโค้ดคำสั่งเฉพาะของแต่ละอุปกรณ์ก็ใช้การส่งข้อมูลออกไปยังอุปกรณ์ตัวนั้นตามโค้ดคำสั่งดังกล่าว นอกจากนั้นแล้วคำสั่งหลายๆ คำสั่งสามารถใช้จาก function ที่มีมาให้ใน IC IMS 9914 ได้ด้วย

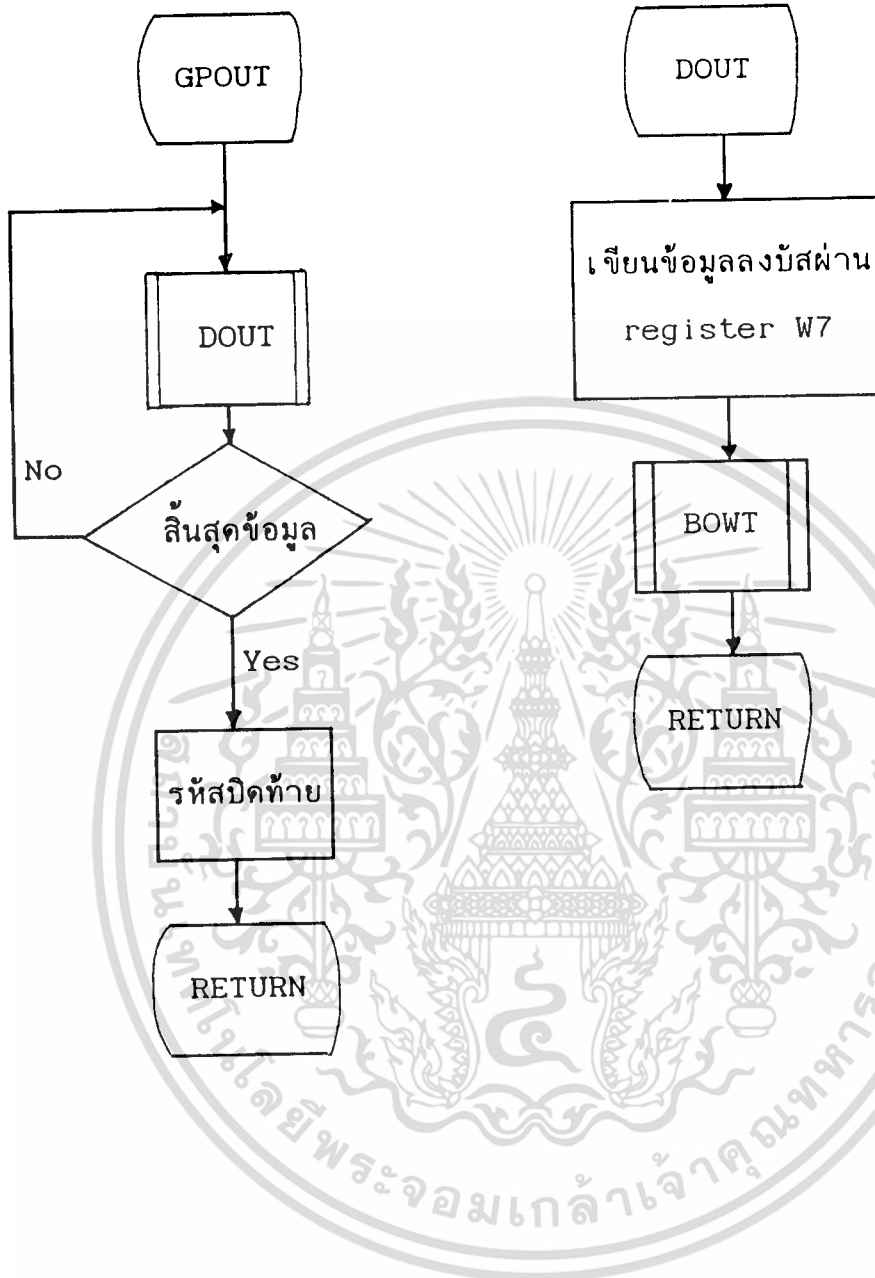
รายละเอียด flow chart ย่อย



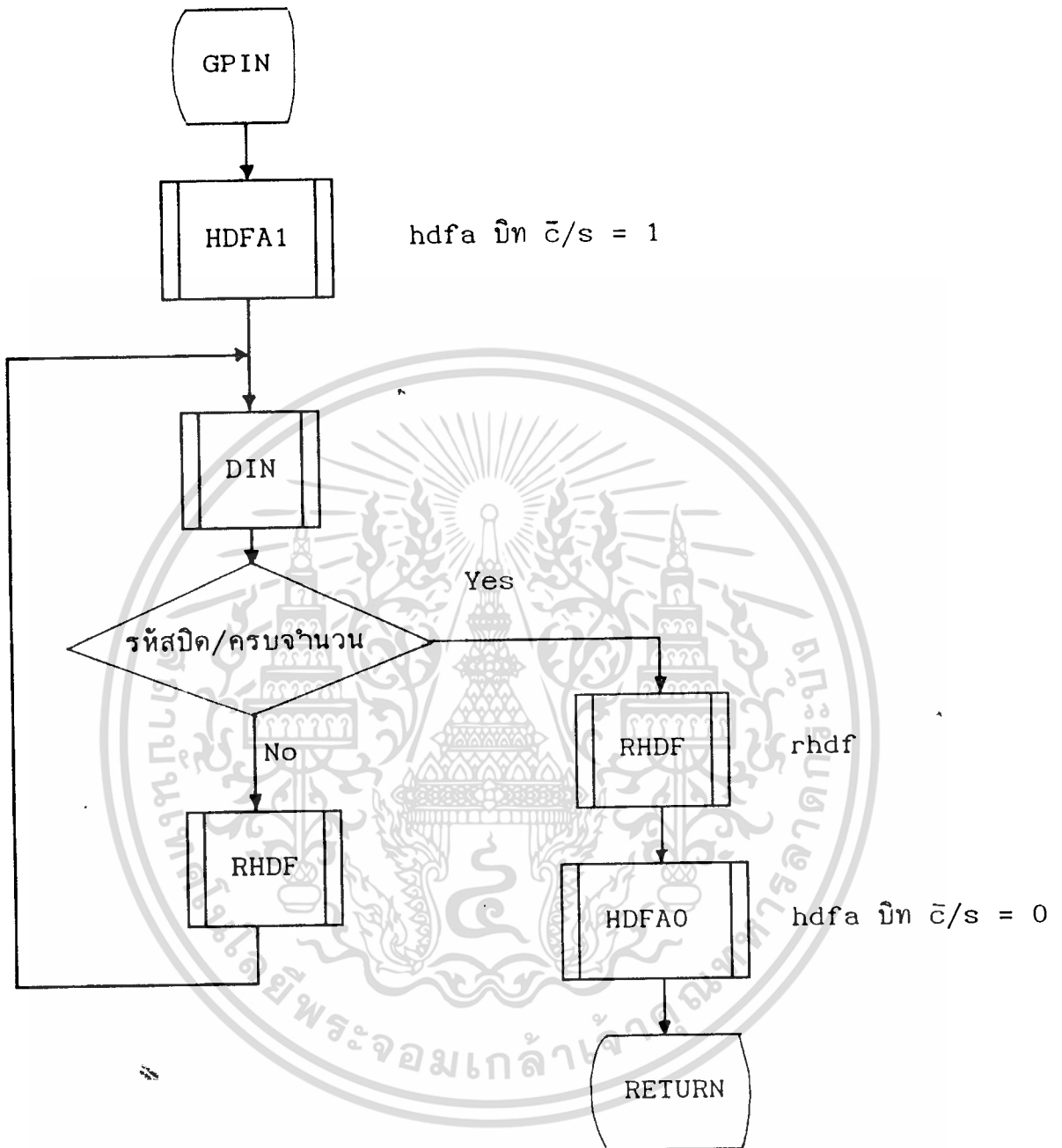
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



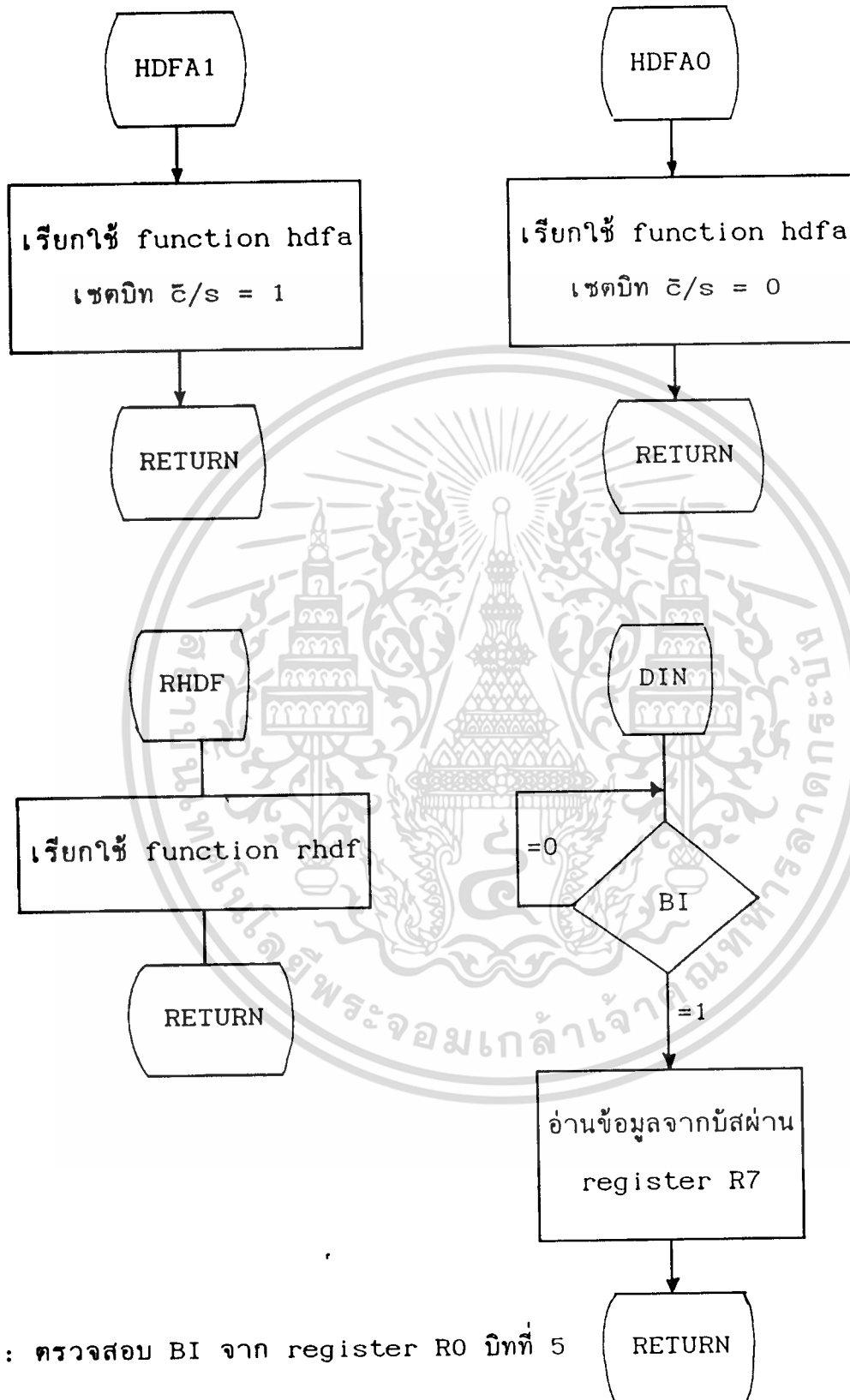
: ตรวจสอบ BO จาก register R0 บิตที่ 4



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



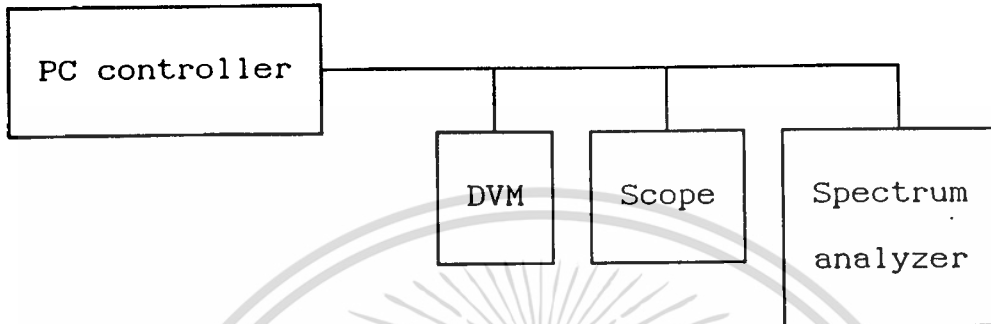
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

How about this project?

โครงการเครื่องต้นแบบมีระบบ IEEE - 488 ดังนี้



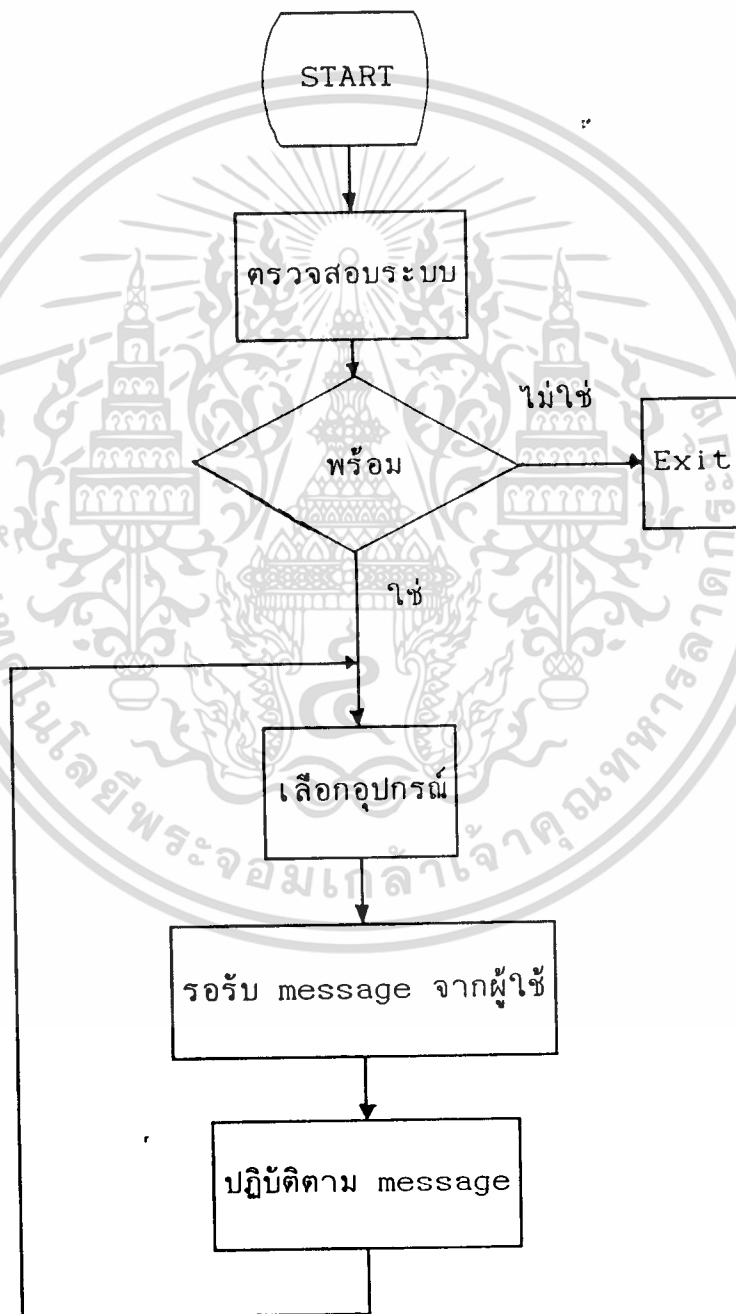
PC controller เป็นตัวควบคุมระบบทำงานบนวินโดวส์ การใช้งานเป็นไปตามระบบของวินโดวส์ประกอบไปด้วย menu, keyboard และ mouse แสดงผลในระบบกราฟฟิก อุปกรณ์ที่อยู่ในระบบมีรายละเอียดดังนี้

- DVM (Digital Multimeter) เป็นได้ทั้ง talker และ listener สามารถควบคุมการปรับ function การทำงานต่างๆ ของตัว DVM ผ่านทาง PC มีทั้งระบบ menu และกราฟฟิกจำลองการทำงานของแผงหน้าเครื่อง รวมไปถึงระบบการวัดที่สามารถวัดค่าตาม function ที่ตั้งไว้ โดยสามารถวัดได้สูงสุด 100 ครั้ง ปรับค่าได้ตามต้องการ
- Scope เป็นได้เพียง talker กล่าวคือสามารถอ่านค่าการวัดจาก monitor ของ Scope รวมไปถึงรายละเอียดทางด้าน function เช่น Time /Div , Volt/Div ฯลฯ ทำให้สามารถแสดงผลทางกราฟฟิกผ่าน PC ได้ รวมไปถึงการประมวลผลข้อมูลที่วัดได้จาก Scope
- Spectrum analyzer เป็นได้ทั้ง talker และ listener สามารถควบคุมการปรับ function การทำงานต่างๆ ของตัว spectrum analyzer ผ่านทาง PC มีทั้งระบบ menu และกราฟฟิกจำลองการทำงานของแผงหน้าเครื่อง รวมไปถึงระบบการวัดที่สามารถวัดค่าจาก monitor ของ spectrum analyzer รวมไปถึงรายละเอียดทางด้าน function เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

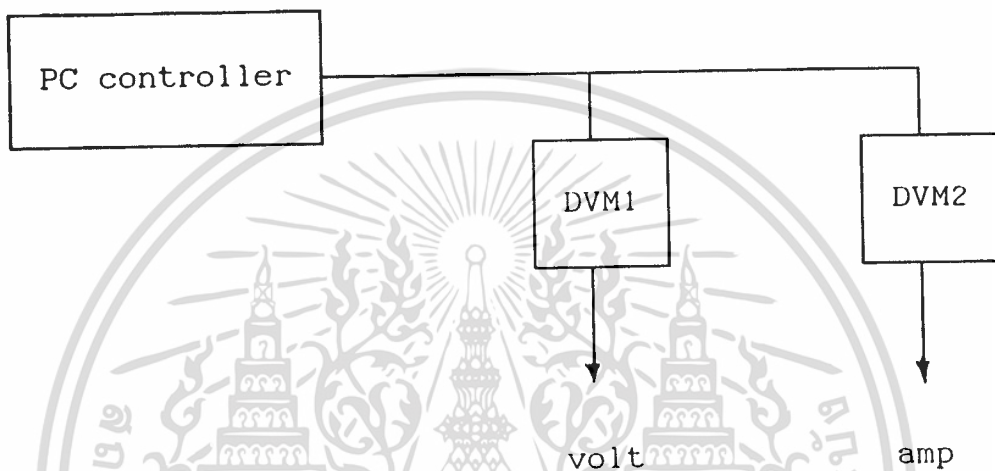
frequency span ,center frequency ฯลฯ ทำให้สามารถแสดงผลทางกราฟฟิกผ่าน PC ได้ รวมไปถึงการประมวลผลข้อมูลที่วัดได้จาก spectrum analyzer

flow chart การทำงานมีดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ในระบบทำงานได้โดยลำพังไม่ขึ้นกับอุปกรณ์ใดๆ และสามารถเรียกทำงานได้พร้อมๆ กัน นอกจากนั้นแล้วยังสามารถออกแบบการประยุกต์ใช้ได้หลายอย่าง ยกตัวอย่างเช่น



เราสามารถโปรแกรมการวัดตั้งรูปเพื่อวัดค่าใดๆ จาก DVM สองตัวโดยควบคุมการวัดผ่าน PC แล้วนำข้อมูลการวัดมาประมวลผล การควบคุมโดย PC สามารถทำให้ ออกแบบการวัดได้หลากหลาย เช่น ตั้งเวลาวัดเองอัตโนมัติ , แสดงผลในระบบกราฟฟิก ฯลฯ การเขียนโปรแกรมก็ใช้ตามมาตรฐาน IEEE 488 ดังที่ได้กล่าวมาแล้ว

บทที่ 5

สรุปผลและวิจารณ์

ปริญญานิพนธ์นี้ประกอบไปด้วย

1. ศึกษา IEEE 488 GPIB (General Purpose Interface Bus)
2. ออกแบบสร้าง hardware ในส่วนของการ์ดควบคุมโดย PC ตามมาตรฐาน IEEE 488 โดย IC ประมวลผล IEEE 488 คือ TMS 9914 ซึ่งสามารถทำงานได้ผลเป็นอย่างดี
3. ออกแบบสร้าง software ในระบบการควบคุมอุปกรณ์โดย IEEE 488 ซึ่งประกอบไปด้วย
 - DVM
 - Spectrum Analyzer
 - Scope
 ปฏิบัติงานบนระบบวินโดวส์ สามารถควบคุมอุปกรณ์ได้พร้อมๆ กัน และรวมไปถึงระบบการวัดของอุปกรณ์แต่ละชนิดด้วย การใช้งานและการแสดงผลในเป็นระบบกราฟฟิก เพื่อความสะดวกและความน่าใช้งาน รวมไปถึงเป็นมาตรฐานสำหรับการใช้งานด้วยระบบวินโดวส์ ที่กำลังเป็นที่นิยม
4. ออกแบบสร้าง Application การวัดด้วย DVM โดย IEEE 488 ด้วยการวัดจาก DVM จำนวน 2 ตัวแบบอัตโนมัติ และเก็บข้อมูลเพื่อความสะดวกในการประมวลผลข้อมูล ด้วยโปรแกรมที่มีความสามารถในการคำนวณสูงคือ โปรแกรม mathematica

แนวทางในการพัฒนาต่อ

- พัฒนาโปรแกรมควบคุมด้วย IC TMS 9914 ที่ยังมีความสามารถอีกมากมาย
- พัฒนาโปรแกรมให้มีความสมบูรณ์ยิ่งขึ้น
- ออกแบบสร้าง hardware ตามมาตรฐาน IEEE 488 สำหรับงานเฉพาะ
- ออกแบบให้เป็นระบบที่มีความสมบูรณ์เพื่อ Application ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. Eugene Fisher and C.W.Jenson, "PET/CBM and the IEEE 488 Bus (GPIB)," Osborne/McGraw-Hill Berkeley, California
2. Advantest Corporation, "TR6845/46/47/48 DIGITAL MULTIMETER INSTRUCTION MANUAL,"
3. Advantest Corporation, "R4131 SERIES SPECTRUM ANALYZER INSTRUCTION MANUAL,"
4. จิรพัฒน์ จันทรเจดศักดิ์ และ วีระ นพนิราพาธม, "เขียนโปรแกรมบน Microsoft Windows," ซีเอ็ดยูเคชั่น จำกัด, กรุงเทพมหานคร
5. ธานินทร์ ถาวรศาสนวงศ์ และ ทินกร ดูก, "การอิเทอร์เฟส IBM PC," ซีเอ็ดยูเคชั่น จำกัด, กรุงเทพมหานคร

