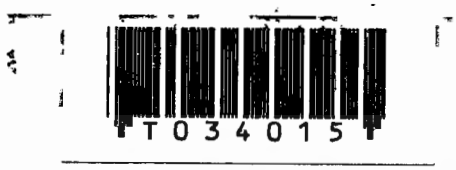
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์ควบคุมผ่าน GPIB

(Curve Tracer Control via GPIB)



โดย

นาย สุรศักดิ์ ตันติตระการวัฒนา

นาย เอกวิทย์ อินเขียวสาย

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2541

เลขหมู่.....
เลขทะเบียน 34015
วัน, เดือน, ปี 1 ต.ค. 2542

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์ควบคุมผ่าน GPIB

(Curve Tracer Control via GPIB)

โดย

นาย สุรศักดิ์ ตันติตระการวัฒนา รหัส 39013183

นาย เอกวิทย์ อินเขียวสาย รหัส 39013187

อาจารย์ที่ปรึกษา

รศ.ดร. มนัส สังวรศิลป์

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2541

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์ควบคุมผ่าน GPIB

ผู้จัดทำ

1. นาย สุรศักดิ์ ตันติตระการวัฒนา รหัส 39013183
2. นาย เอกวิทย์ อินเขียวสาย รหัส 39013187

..... อาจารย์ที่ปรึกษา

(รศ.ดร. มนัส สังวรศิลป์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์ควบคุมผ่าน GPIB

(Curve Tracer Control via GPIB)

นาย สุรศักดิ์ ตันติตระการวัฒนา รหัส 39013183

นาย เอกวิทย์ อินเขียวสาย รหัส 39013187

โครงงานได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

.....

(รศ.ดร. มนัส สังวรศิลป์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์ควบคุมผ่าน GPIB

นาย สุรศักดิ์ ตันติตระการวัฒนา รหัส 39013183
นาย เอกวิทย์ อินเขียวสาย รหัส 39013187
รศ.ดร. มนัส สังวรศิลป์ (อาจารย์ที่ปรึกษา)
ปีการศึกษา 2541

บทคัดย่อ

ในวิทยานิพนธ์ฉบับนี้ได้ทำการสร้างเครื่องวัดคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์โดยควบคุมผ่าน IEEE488 (GPIB) ซึ่งเป็นระบบบัสมาตรฐานที่ใช้เครื่องคอมพิวเตอร์ในการควบคุมเครื่องมือวัดได้แก่ แหล่งจ่ายไฟ และ มัลติมิเตอร์ โดยมีคำสั่งมาตรฐานของ IEEE-488 ใช้ในการควบคุมการวัดอุปกรณ์แต่ละตัว ซึ่งสามารถที่จะวัดอุปกรณ์ได้ คือ ไดโอด ทรานซิสเตอร์ เฟท และมอสเฟทได้ ค่าที่ได้จากการวัดก็ถูกต้อง 80% แต่จะมีปัญหาในเรื่องของความเร็วในการวัด ซึ่งเกิดจากมัลติมิเตอร์ที่มีความช้าในการวัดเนื่องจากต้องการค่าที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Curve Tracer Control via GPIB

Mr. Surasak Tantitrakranwattana

Mr. Ekawit Inkhieosai

Assoc.Prof.Dr.Manus Sangworasilp (Advisor)

Educational Year 1998

Abstract

In this project the equipment for measuring the electronic device characteristic is presented. The measurement is carried out by using IEEE488(GPIB) equipment such as power supply and multimeter. Various standard command available in IEEE488 can be employed to obtain diode, BJT, MOSFET characteristic curve. It exists, however, the problem of regarding the measurement speed due to the speed limitation of the equipment being used to measure the device characteristic.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	I
Abstract	II
สารบัญ	III
สารบัญรูป	V
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีพื้นฐานเกี่ยวกับระบบบัสมาตรฐาน IEEE-488(GPIB)	3
2.1 โครงสร้างของ GPIB	3
2.2 ข้อจำกัดของระบบบัส IEEE-488	3
2.3 มาตราฐานระบบบัส IEEE-488	5
2.3.1 การเชื่อมต่ออุปกรณ์ต่างๆในระบบ	5
2.3.2 ขั้วต่อและความหมายของสัญญาณต่างๆภายในบัส	6
2.3.3 ความหมายของคำสัญญาณต่างๆในบัส GPIB	7
2.4 การกำหนด GPIB Addressing	8
2.5 คำสั่งใช้งานของ IEEE-488 (GPIB)	9
2.6 ขบวนการแฮนด์เช็ค (Handshake Procedure)	12
2.6.1 ขบวนการแฮนด์เช็คเมื่อใช้ไมโครคอมพิวเตอร์เป็นตัวควบคุมและตัวส่ง	13
2.6.2 ขบวนการแฮนด์เช็คเมื่อใช้ไมโครคอมพิวเตอร์เป็นตัวควบคุมและตัวรับ	14
บทที่ 3 การใช้โปรแกรม NI-488.2	
3.1 การใช้โปรแกรม NI-488.2M กับ Windows 95	16
3.2 ฟังก์ชันในการติดต่อกับ NI-488.2	17
3.3 การเขียนโปรแกรมบน Ni-488 หรือ NI-488.2	20
3.4 คำสั่งที่ใช้ควบคุมเครื่องมือวัดของอุปกรณ์	22
บทที่ 4 ทฤษฎีการทำงานพื้นฐานของอุปกรณ์	31
4.1 ไดโอดชนิดหัวต่อ P-N (P- N Junction diode)	31
4.1.1 คุณลักษณะสมบัติของไดโอด	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ทรานซิสเตอร์ 32
4.2.1 กราฟคุณสมบัติของทรานซิสเตอร์ 36
4.3 ทรานซิสเตอร์สนามไฟฟ้า 36
4.3.1โครงสร้างและคุณลักษณะของเฟท 36
4.3.2 ขณะ V_{GS} มีค่าเท่ากับศูนย์ 37
4.3.3 ขณะ V_{GS} มีค่าเป็นลบ 39
บทที่ 5 การออกแบบและการเขียนโปรแกรมควบคุม 41
5.1 การเขียนโปรแกรมควบคุมในส่วนของไมโครคอนโทรลเลอร์ 41
5.2 ส่วนควบคุมการวัดอุปกรณ์ 52
5.2.1 การวัดไดโอด 52
5.2.2 การวัดทรานซิสเตอร์ 54
5.2.3 การวัดเฟท 56
บทที่ 6 การทดลองและการใช้งานโปรแกรม 61
6.1 การทดลอง 61
6.1.1 การวัดคุณสมบัติของไดโอด 61
6.1.2 การวัดทรานซิสเตอร์ 62
6.1.3 การวัดคุณสมบัติของเฟท 65
6.2 ผลการทดลองในส่วนของไมโครคอนโทรลเลอร์ 68
6.3 การใช้งานโปรแกรม 71
บทที่ 7 สรุปและวิจารณ์ผลการทดลอง 77
ภาคผนวก
บรรณานุกรม
กิตติกรรมประกาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 1.1 แสดงบล็อกไดอะแกรมของเครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์ 1
รูปที่ 2.1 แสดงแผนผังแสดงอุปกรณ์และสายสัญญาณต่างๆในระบบ GPIB 4
รูปที่ 2.2 แสดงการเชื่อมต่อแบบเรียงต่อเนื่องกัน 5
รูปที่ 2.3 แสดงการเชื่อมต่อแบบกระจาย 6
รูปที่ 2.4 แสดงตำแหน่งขาขั้วสัญญาณ GPIB 6
รูปที่ 2.5 แสดงการกำหนดตำแหน่งของGPIB 8
รูปที่ 2.6 แสดงโฟว์ชาร์ทของการแฮนด์เช็คทั้งด้านส่งและด้านรับ 11
รูปที่ 2.7 แสดงขบวนการแฮนด์เช็ค 12
รูปที่ 2.8 แสดงแผนผังเวลาของการแฮนด์เช็คเมื่อใช้คอมพิวเตอร์เป็นตัวควบคุมและตัวส่ง 13
รูปที่ 2.9 แสดงแผนผังเวลาของการแฮนด์เช็คเมื่อใช้คอมพิวเตอร์เป็นตัวควบคุมและตัวส่ง 14
รูปที่ 3.1 แสดงการใช้ NI-488.2M กับ Windows95 17
รูปที่ 4.1 ลักษณะโครงสร้างของไดโอดชนิดหัวต่อและสัญลักษณ์ 31
รูปที่ 4.2 กราฟแสดงความสัมพันธ์ระหว่างกระแสกับแรงดันของไดโอด 31
รูปที่ 4.3 โครงสร้างพื้นฐานของทรานซิสเตอร์ 32
รูปที่ 4.4 วงจร 4 ขั้วมีขาอิมิเตอร์เป็นขาร่วมระหว่างอินพุทและเอาท์พุต 33
รูปที่ 4.5 ลักษณะการจัดทรานซิสเตอร์พื้นฐาน 3 แบบ 34
รูปที่ 4.6วงจรใช้งานทรานซิสเตอร์พื้นฐาน 35
รูปที 4.7 กราฟคุณสมบัติทางด้านเอาท์พุตของทรานซิสเตอร์เบอร์ BC107 36
รูปที่ 4.8 แสดงโครงสร้างของเฟท 37
รูปที่ 4.9 เมื่อป้อนแรงดัน V_{DD} ที่ $V_{GS} = 0$ 38
รูปที่ 4.10 (ก)แสดงกราฟคุณสมบัติของเฟท 38
(ข) แสดงโครงสร้างของเฟทขณะเกิดจุดพินช์ออฟ (Pinch-off) 38
รูปที่ 4.11 (ก) แสดงโครงสร้างของเฟท เมื่อ V_{GS} เป็นลบ 39
(ข) แสดงกราฟความสัมพันธ์ระหว่างกระแสกับแรงดัน 39
รูปที่ 5.1 แสดงบล็อกไดอะแกรมในส่วนของไมโครคอนโทรลเลอร์ติดต่อกับคอมพิวเตอร์ 41
รูปที่ 5.2แสดงแผนผังเวลาเมื่อตัวควบคุมทำการติดต่อกับตัวส่ง 43
รูปที่ 5.3 แสดงแผนผังเวลาเมื่อตัวควบคุมทำการติดต่อกับตัวรับ 43
รูปที่ 5.4 แสดงโฟว์ชาร์ท ในาการรับส่งข้อมูลกับตัวควบคุม 45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.5 แสดงโฟว์ชาร์ทในการแฮนด์เช็คในการรับข้อมูลจากตัวควบคุม 47
รูปที่ 5.6 แสดงโฟว์ชาร์ทในการแฮนด์เช็คในการส่งข้อมูลให้กับตัวควบคุม 48
รูปที่ 5.7 แสดงวงจรที่ใช้ทดสอบคุณสมบัติของไดโอด 52
รูปที่ 5.8 แสดงโฟว์ชาร์ทของการวัดคุณสมบัติของไดโอด 53
รูปที่ 5.9 แสดงวงจรที่ใช้ในการวัดคุณสมบัติของทรานซิสเตอร์ 54
รูปที่ 5.10 แสดงโฟว์ชาร์ทของการวัดคุณสมบัติของทรานซิสเตอร์ 55
รูปที่ 5.11 แสดงวงจรที่ใช้วัดคุณสมบัติของเฟท 56
รูปที่ 5.12 แสดงวงจรในส่วนของไมโครคอนโทรลเลอร์ 58
รูปที่ 5.13 แสดงวงจรในส่วนของการหาตำแหน่งขาของทรานซิสเตอร์ 59
รูปที่ 5.14 แสดงวงจรจ่ายกระแสสำหรับทรานซิสเตอร์ 60
รูปที่ 6.1 ผลการวัดคุณสมบัติของไดโอดเบอร์ 1N4001 61
รูปที่ 6.2 แสดงผลการวัดซีเนอร์ไดโอด 62
รูปที่ 6.3 แสดงผลการวัดคุณสมบัติของทรานซิสเตอร์เบอร์ D786 ชนิด N 63
รูปที่ 6.4 แสดงผลการวัดของเบอร์ 2N4037 ชนิด P 63
รูปที่ 6.5 แสดงผลการวัดทรานซิสเตอร์เบอร์ 2SC1815 ในการเลือกค่ากระแสเบส 9 ค่า 64
รูปที่ 6.6 แสดงผลการวัดทรานซิสเตอร์เบอร์ 2SC1815 ในการเลือกค่ากระแสเบส 5 ค่า 64
รูปที่ 6.7 ผลการวัดคุณสมบัติของเฟทเบอร์ BF245C 65
รูปที่ 6.8 ผลการทดลองวัดคุณสมบัติของเฟทเบอร์ BF245C โดยใช้ Pspice 65
รูปที่ 6.9 แสดงผลการวัดคุณสมบัติของเฟทชนิด N เบอร์ BF245
เมื่อเปลี่ยนค่าแรงดันที่เกทใหม่ 66
รูปที่ 6.10 แสดงผลการวัดคุณสมบัติของเฟทชนิด P เบอร์ 2N5462 66
รูปที่ 6.11แสดงผลการวัดคุณสมบัติของมอสเฟท ชนิด N เบอร์ BS170 67
รูปที่ 6.12 แสดงค่าแรงดันขีดเริ่มของมอสเฟท เบอร์ BS170 67
รูปที่6.13 แสดงหน้าจอเมื่อใช้คำสั่ง lbclr 68
รูปที่ 6.14 แสดงหน้าจอเมื่อใช้คำสั่ง lbloc 68
รูปที่ 6.15 แสดงหน้าจอเมื่อใช้คำสั่ง lbwrt"SG" 68
รูปที่ 6.16 แสดงหน้าจอเมื่อใช้คำสั่ง lbwrt"STN" 69
รูปที่ 6.17 แสดงหน้าจอเมื่อใช้คำสั่ง lbwrt"STP" 69
รูปที่ 6.18 แสดงหน้าจอเมื่อใช้คำสั่ง lbwrt"SFN" 69
รูปที่ 6.19 แสดงหน้าจอเมื่อใช้คำสั่ง lbwrt"SFP" 70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.20 แสดงหน้าจอเมื่อใช้คำสั่ง Ibwrt"STEST" 70
รูปที่ 6.21 แสดงหน้าจอเมื่อใช้คำสั่ง Ibwrt"ST:1" 70
รูปที่ 6.22 แสดงหน้าจอเมื่อเปิดโปรแกรมครั้งแรก 71
รูปที่ 6.23 แสดงสวิทช์ที่ตั้งค่าตำแหน่งของเครื่องซึ่งจะอยู่ด้านหลังของเครื่อง 72
รูปที่ 6.24 แสดงตำแหน่งของเครื่องหลังจากเริ่มเปิดเครื่องใช้งาน 72
รูปที่ 6.25 แสดงหน้าจอในการตั้งค่าตำแหน่งของเครื่องที่โปรแกรม 72
รูปที่ 6.26 แสดงการตั้งค่าก่อนทำการวัดทรานซิสเตอร์ 73
รูปที่ 6.27 จุดทดสอบการหาตำแหน่งของทรานซิสเตอร์ 73
รูปที่ 6.28 จุดทดสอบคุณสมบัติของอุปกรณ์ซึ่งจะต้องทราบตำแหน่งขาก่อนที่จะวัด 74
รูปที่ 6.29 แสดงหน้าจอเมื่อทำการวัดทรานซิสเตอร์เสร็จแล้ว 74
รูปที่ 6.30 แสดงหน้าจอเมื่อทำการSave ภาพที่ได้จากการวัด 75
รูปที่ 6.31 การต่อสายระหว่างเครื่องฯ กับอุปกรณ์ต่อร่วม 75
รูปที่ 6.32 ตำแหน่งด้านหลังของเครื่องที่จะต้องต่อ 76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

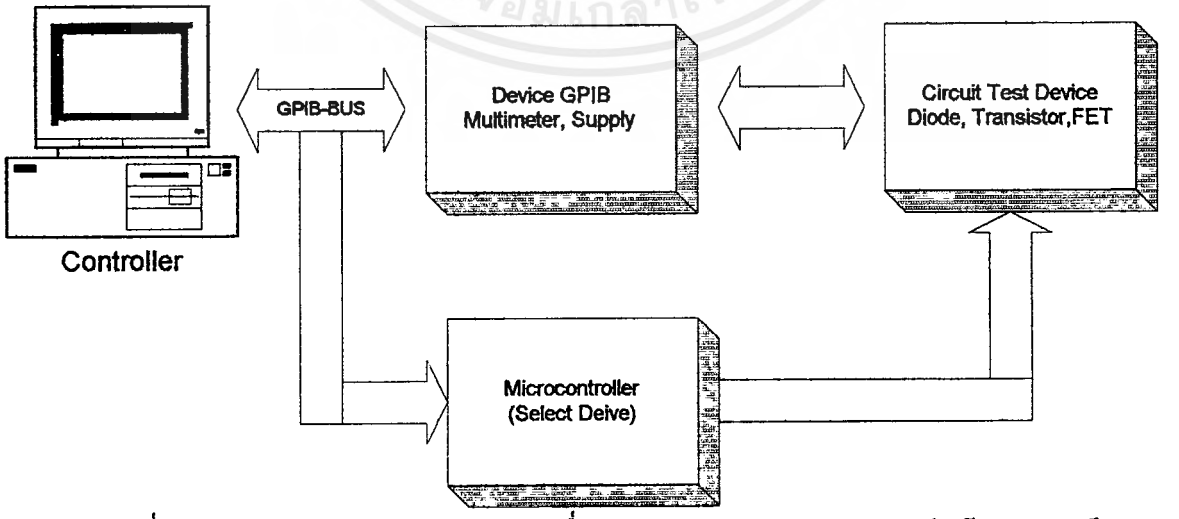
ในการเชื่อมต่ออุปกรณ์ภายนอกกับคอมพิวเตอร์ได้มีการนำมาตราฐานในการติดต่อ IEEE 488.1-1987 หรือที่นิยมเรียกกันว่า GPIB (General Purpose Interface Bus) ซึ่งมีลักษณะในการติดต่อระหว่างเครื่องมือวัดและอุปกรณ์ควบคุมจากผู้ใช้ (User) ลักษณะในการใช้งานจะเกี่ยวกับทางด้านอิเล็กทรอนิกส์ และทางด้านเครื่องกล

GPIB จะมีลักษณะเป็นสัญญาณดิจิตอลโดยพื้นฐานที่สำคัญจะประกอบด้วย

- การส่งข้อมูลแบบขนาน มีขนาด 8 บิท มีความเร็วในการส่ง 1 เมกกะไบท์ต่อวินาที (Mbytes/S)
- การใช้สายสัญญาณ 3 เส้น ในการแฮนด์เช็ค (Handshake) และใช้สายสัญญาณ 5 เส้น ในการควบคุม (Controller)

และได้มีการพัฒนามาเป็น IEEE 488.2-1992 โดยใช้กติกาการติดต่อแบบบัส มีการเซ็ทชนิดของข้อมูลและรูปแบบของข้อมูล

ในการใช้งานอุปกรณ์ GPIB เราสามารถที่จะใช้โปรแกรม Delphi มาควบคุมการทำงานของเครื่องมือวัดแต่ละตัวได้ ซึ่งในโครงงานนี้ได้ทำการควบคุมแหล่งจ่ายไฟและ มิเตอร์ เพื่อมาใช้ในการวัดหาคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์ ซึ่งจะนำค่าที่อ่านได้จากมิเตอร์มาพล็อตเป็นเส้นกราฟแสดงผลที่หน้าจอคอมพิวเตอร์ โดยจะมีบล็อกไดอะแกรม ดังรูปที่ 1



รูปที่ 1.1 แสดง Block diagram ของเครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 1.1 การควบคุมจะถูกสั่งงานมาจากคอมพิวเตอร์ ซึ่งจะเป็นตัวควบคุม (Controller) คอยสั่งให้อุปกรณ์แต่ละตัวทำงาน อุปกรณ์ GPIB แต่ละตัวจะมีตำแหน่งของเครื่อง ไม่เหมือนกัน ซึ่งเราสามารถที่จะกำหนดตำแหน่งได้จากตัวอุปกรณ์เอง ในส่วนของไมโครคอนโทรลเลอร์นั้นจะเปรียบเสมือนตัวอุปกรณ์ GPIB ตัวหนึ่ง ซึ่งสามารถที่จะตั้งค่าตำแหน่งได้ โดยจะเป็นตัวรับและตัวส่ง ซึ่งจะรับคำสั่งจากตัวควบคุม เช่นกัน แล้วมาควบคุมการเลือกอุปกรณ์ที่ต้องการจะทดสอบ

1.1 คุณสมบัติของเครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์

ความสามารถในการตรวจสอบคุณสมบัติของอุปกรณ์อิเล็กทรอนิกส์ที่สามารถวัดได้ ซึ่งอธิบายได้เป็นข้อๆ ดังนี้

1. สามารถวัดกราฟคุณสมบัติของไดโอดทางด้านไบอัสตรง ได้
2. สามารถวัดกราฟคุณสมบัติของทรานซิสเตอร์ได้ทั้งชนิด NPN และ PNP ได้ ซึ่งแรงดัน Vcc ที่จ่ายให้ทรานซิสเตอร์สูงสุดได้ 30V
3. ในการวัดทรานซิสเตอร์นั้นสามารถที่จะเลือกย่านการจ่ายกระแสเบสได้ 4 ย่านคือ 10uA,100uA 1mA และ10mA ซึ่งในการวัดนั้นจะต้องเลือกย่านให้เหมาะสมกับทรานซิสเตอร์ด้วย
4. สามารถวัดกราฟคุณสมบัติของเฟทได้ ทั้งชนิดเอ็น และ ชนิด พี
5. ในส่วนของทรานซิสเตอร์จะมีส่วนหาขาให้อัตโนมัติในกรณีที่ไม่ทราบขามาก่อน
6. การส่งข้อมูลติดต่อกับคอมพิวเตอร์จะใช้การส่งข้อมูลผ่านบัสมาตรฐาน IEEE488(GPIB)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีพื้นฐานเกี่ยวกับระบบบัสมาตราฐาน IEEE- 488 (GPIB)

2.1 โครงสร้างของ GPIB

ส่วนประกอบพื้นฐานของระบบ GPIB แบ่งออกได้เป็น 3 ส่วน คือ

1. **ตัวส่ง** (Talker) ทำหน้าที่ในการส่งข้อมูล ในระบบสามารถมีตัวส่งได้หลาย ๆ ตัว แต่จะมีตัวส่งเพียงตัวเดียวเท่านั้นที่กำลังทำงานอยู่ในระบบ เพราะไม่สามารถที่จะทำงานพร้อมกันได้

2. **ตัวรับ** (Listener) ทำหน้าที่ในการรับข้อมูล ตัวรับในระบบนั้น สามารถมีได้หลายตัวเช่นกันและสามารถที่จะทำงานพร้อมๆ กันได้ด้วย ถ้าเป็นอุปกรณ์ที่เหมือนกัน

3. **ตัวควบคุม** (Controller) ทำหน้าที่ควบคุมสัญญาณต่าง ๆ บนบัส โดยตอบสนองความต้องการของอุปกรณ์ที่จะส่งข้อมูล โดยการกำหนดตัวส่งที่จะทำการส่งข้อมูล หรือกำหนดตัวรับให้ทำการรับข้อมูลที่ส่งลงบนบัส

การทำหน้าที่ของอุปกรณ์ในระบบ อุปกรณ์หรือเครื่องมือในระบบ GPIB นั้น สามารถแบ่งตามหน้าที่การทำงานได้ดังนี้

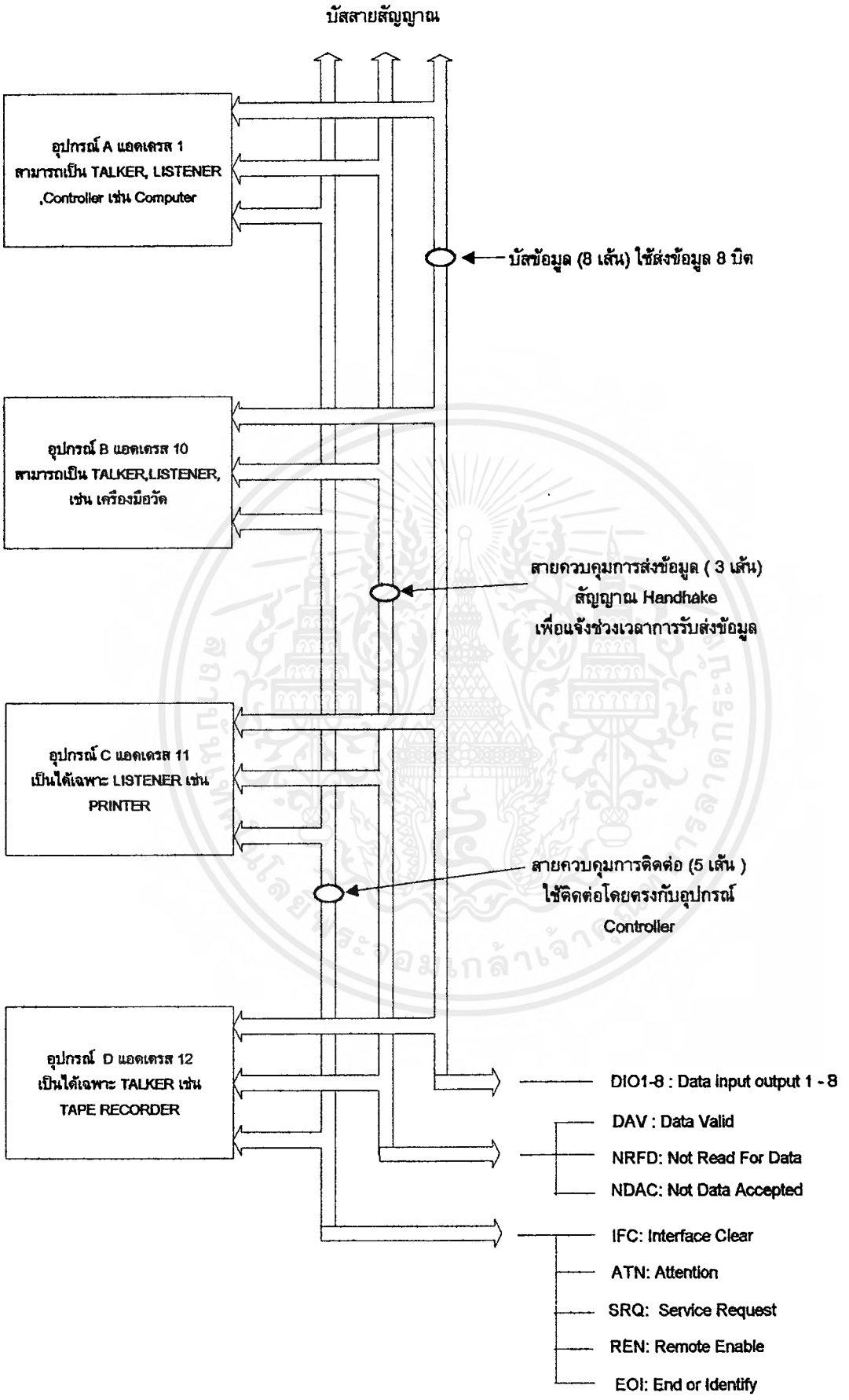
1. ทำหน้าที่เป็นตัวส่งเท่านั้น เช่น เครื่องมือวัด เป็นต้น
2. ทำหน้าที่เป็นตัวรับเท่านั้น เช่น เครื่องพิมพ์ เป็นต้น
3. ทำหน้าที่เป็นทั้งตัวส่งและตัวรับ เช่น ไมโครคอมพิวเตอร์ , เครื่องมือวัดที่สามารถควบคุมได้จากภายนอก เป็นต้น
4. ทำหน้าที่เป็นตัวส่ง ตัวรับและตัวควบคุม เช่น ไมโครคอมพิวเตอร์ เป็นต้น

สำหรับแผนผังการแสดงอุปกรณ์และสายสัญญาณต่าง ๆ ในระบบ GPIB นั้น แสดงดังรูปที่ 2.1

2.2 ข้อจำกัดของระบบบัส IEEE - 488

1. จำนวนอุปกรณ์ (ตัวรับ , ตัวส่ง , ตัวควบคุม) ที่ต่ออยู่กับบัสของระบบ 1 เส้น จะต้องไม่เกิน 15 เครื่อง
2. สายเคเบิ้ลที่ใช้ต่อระหว่างอุปกรณ์แต่ละตัวต้องยาวไม่เกิน 4 เมตร และความยาวรวมของสายเคเบิ้ลต้องไม่เกิน 20 เมตร
3. ความเร็วในการส่งข้อมูลต้องไม่เกิน 1 Mb/s
4. จำนวนของอุปกรณ์หรือเครื่องมือมากกว่าครึ่งหนึ่งต้องเปิดให้ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 แสดงแผนผังแสดงอุปกรณ์และสายสัญญาณต่างๆ ในระบบ GPIB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

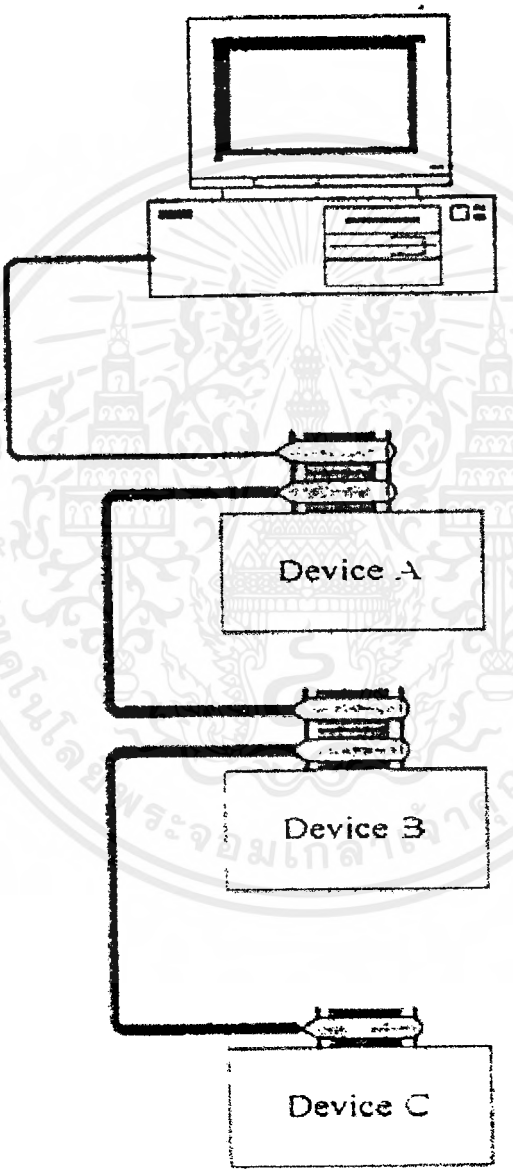
2.3 มาตราฐานระบบบัส IEEE-488

มาตราฐานของระบบบัส IEEE -488 สามารถแบ่งออกได้ดังนี้

2.3.1 การเชื่อมต่ออุปกรณ์ต่าง ๆ ในระบบ

การเชื่อมต่ออุปกรณ์หรือเครื่องมือเข้าในระบบ GPIB นั้นมี 2 แบบ คือ

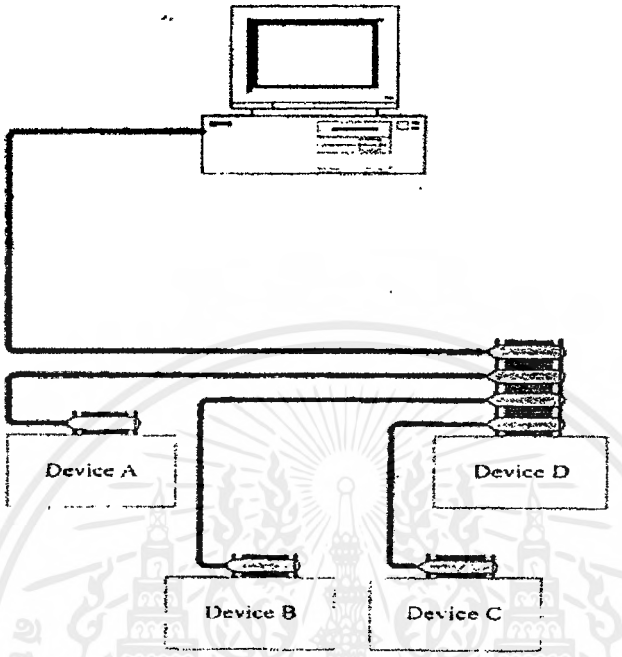
1. การเชื่อมต่อแบบเรียงต่อเนื่องกัน (Daisy chain configuration)ดังแสดงในรูปที่ 2.2



รูปที่ 2.2 แสดงการเชื่อมต่อแบบเรียงต่อเนื่องกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

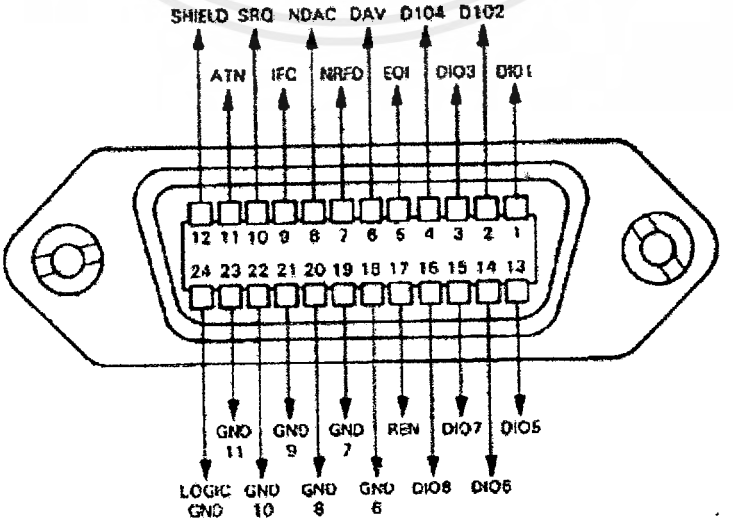
2. การเชื่อมต่อแบบกระจาย (Star configuration) ดังแสดงในรูปที่ 2.3



รูปที่ 2.3 แสดงการเชื่อมต่อแบบกระจาย

2.3.2 ขั้วต่อและความหมายของสัญญาณต่าง ๆภายในบัส

ขั้วต่อมาตราฐานของ GPIB มีทั้งหมด 24 ขา มีรูปแบบและการจัดตำแหน่งของขาสัญญาณต่าง ๆดังแสดงในรูปที่ 2.4



รูปที่ 2.4 แสดงตำแหน่งขาขั้วต่อสัญญาณ GPIB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 ความหมายของคำสัญญาณต่าง ๆ ในบัส GPIB

ระบบบัส GPIB เป็นระบบบัสแบบขนาน ดังนั้นอุปกรณ์ทุกตัวที่ต่อรวมบัสกันอยู่จึงต่อขนานกันหมด ดังนั้นสายสัญญาณทั้ง 24 เส้นในระบบจึงเชื่อมต่อถึงอุปกรณ์ในระบบทุกตัว สายสัญญาณทั้ง 24 เส้นสามารถแบ่งออกได้เป็น 3 กลุ่ม สัญญาณ คือ

1. กลุ่มสัญญาณควบคุมการรับส่งข้อมูล มี 3 เส้น ใช้สำหรับการส่งข้อมูลแบบแฮนด์เช็ค ประกอบด้วย

DAV (Data Vaild) เมื่อสัญญาณถูกดึงเป็นลอจิก Low โดยอุปกรณ์ตัวส่ง เป็นการแจ้งให้ทราบว่าขณะนี้ตัวส่งได้ทำการส่งข้อมูลลงไปที่บัสข้อมูลเรียบร้อยแล้ว

NRFD (Not Ready For Data) เมื่อสายสัญญาณนี้มีลอจิกเป็น Low เป็นการแจ้งให้ทราบว่า ขณะนี้ตัวรับในระบบยังไม่พร้อมที่จะรับข้อมูล เนื่องจากอุปกรณ์ในระบบยังไม่พร้อมที่จะทำงานหมดทุกตัว จนกว่าอุปกรณ์ทุกตัวจะส่งสัญญาณที่มีลอจิกเป็น High ครบทุกตัว เพื่อแสดงว่าพร้อมที่จะรับข้อมูล

NDAC (Not Data Accept) สัญญาณนี้ถูกควบคุมโดยอุปกรณ์ตัวรับ โดยจะให้ลอจิกเป็น Low ขณะที่ตัวรับกำลังเก็บข้อมูลจากบัสข้อมูลอยู่ และจะมีลอจิกเป็น High เมื่อทำการเก็บข้อมูลเรียบร้อยแล้ว

2. กลุ่มสัญญาณควบคุมการเชื่อมต่อ มี 5 เส้น ประกอบด้วย

ATN (Attention) เป็นสัญญาณที่ถูกส่งจากตัวควบคุม ใช้ในการสั่งให้อุปกรณ์ในระบบทุกตัวเตรียมพร้อมเพื่อรอรับคำสั่งต่อไป

IFC (Interface Clear) เป็นสัญญาณที่ใช้ในการเคลียร์ระบบ ถูกควบคุมโดยตัวควบคุมเท่านั้น เมื่ออุปกรณ์ในระบบได้รับสัญญาณนี้ก็จะกลับคืนสู่สภาวะเริ่มต้นใหม่ เป็นสภาวะแรกเริ่มก่อนการกำหนดฟังก์ชั่นการทำงานเหมือนตอนเริ่มเปิดสวิทช์

REN (Remote Enable) เป็นสัญญาณที่ถูกควบคุมโดยตัวควบคุมอีกเช่นกัน เพื่อสั่งให้อุปกรณ์ในระบบเปลี่ยนสถานะการทำงานจากโหมดการใช้งานปกติมาเป็นการทำงานแบบควบคุมโดยตัวควบคุมแทนหรือโหมดรีโมทนั่นเอง

SRQ (Service request) เป็นสัญญาณอินเทอร์รัพท์ เพื่อบอกแก่ระบบว่าขณะนี้อุปกรณ์ต้องการการติดต่อจากตัวควบคุม

EOI (End Or Identify) เป็นสัญญาณที่ถูกควบคุมโดยตัวควบคุมหรือตัวส่งก็ได้ ใช้แสดงว่าข้อมูลชุดที่ส่งนั้นสิ้นสุดลงแล้ว

3. กลุ่มสัญญาณข้อมูล ประกอบด้วยสายสัญญาณข้อมูลทั้ง 8 เส้น ได้แก่ DIO1 – DIO8 สำหรับเป็นทางผ่านของข้อมูลของระบบ ส่วนสายสัญญาณที่เหลืออีก 7 เส้นจะเป็นกราว์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และมีเส้นหนึ่งเป็นสาย Shield สัญญาณควบคุมที่มีอยู่อาจไม่ได้ใช้พร้อมกันทั้งหมด บางเส้นอาจไม่ได้ใช้งานในอุปกรณ์บางเครื่องก็ได้ แล้วแต่ความจำเป็นในการติดต่อ

สำหรับลอจิกที่ใช้ในระบบบัสดังกล่าวนี้จะมีลักษณะเป็นคอมพลีเมนต์ทั้งหมด คือ ลอจิก Low จะมีค่าเป็น 1 และลอจิก High จะมีค่าเป็น 0

2.4 การกำหนด GPIB Addressing

อุปกรณ์ GPIB และบอร์ดจะต้องมีการกำหนดตำแหน่ง (Address) ซึ่งในการกำหนดตำแหน่งนั้นสามารถทำได้ 2 ส่วนคือ ตำแหน่งหลัก (Primary Address) และตำแหน่งสำรอง (Secondary) ดังแสดงดังรูปที่ 2.5

Bit Position	7	6	5	4	3	2	1	0
Meaning	0	TA	LA	GPIB Primary Address (range 0 – 30)				

รูปที่ 2.5 แสดงการกำหนดตำแหน่งของ GPIB

- การกำหนดตำแหน่งหลัก คือจะแสดงค่าในช่วง 0 – 30 และในการควบคุมอุปกรณ์ GPIB จะใช้การกำหนดตำแหน่งในช่องของ TA และ LA

-การกำหนดผู้ส่ง (TA) โดยการเซ็ทที่บิท 6

-การกำหนดผู้รับ (LA) โดยการเซ็ทที่บิท 5

ตัวอย่าง ถ้าอุปกรณ์อยู่ที่ตำแหน่ง 1 แล้วตัวควบคุมส่งข้อมูล 41H (จะทำให้ที่ตำแหน่ง TA เป็น "1 ") ทำให้อุปกรณ์ทำหน้าที่เป็นผู้ส่ง และโดยทั่วไปตัวควบคุมจะใช้ตำแหน่งหลัก (Primary Address) ที่ตำแหน่ง "0" และจะส่งค่า 20H (จะทำการเซ็ท LA เป็น "1") จะทำให้ตัวเองทำหน้าที่เป็นผู้รับข้อมูลหรือคำสั่งควบคุม

- การกำหนดตำแหน่งสำรอง คือจะแสดงค่าในช่วง 60H – 7EH เมื่อตำแหน่งสำรองถูกใช้ตัวควบคุมจะถูกกำหนดให้เป็นผู้ส่งหรือผู้รับเพื่อส่งค่าให้กับอุปกรณ์ ตามค่าในการกำหนดตำแหน่งสำรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 คำสั่งใช้งานของ IEEE 488 (GPIB)

การควบคุมและการกำหนดฟังก์ชั่นการทำงานให้แก่อุปกรณ์เครื่องมือวัดในระบบ GPIB นั้นตัวควบคุมจะเป็นตัวกำหนด โดยการส่งรหัสคำสั่งไปยังตัวอุปกรณ์โดยผ่านทางบัสของระบบคำสั่งสำหรับการกำหนดการทำงานต่าง ๆ ตามมาตราฐานของ IEEE 488 มีอยู่ 128 คำสั่ง แบ่งได้เป็น 2 กลุ่มคำสั่งใหญ่ ๆ คือ กลุ่มคำสั่งหลัก (Primary Command Group) และกลุ่มคำสั่งรอง (Secondary Command Group) โดยกลุ่มคำสั่งหลักประกอบด้วย 4 กลุ่มคำสั่ง คือ กลุ่มคำสั่งเจาะจงจุดหมาย (Addressed Command Group) ,กลุ่มคำสั่งครอบคลุม (Universal Command Group), กลุ่มคำสั่งกำหนดอุปกรณ์ตัวรับ (Listen Address Group) และกลุ่มคำสั่งกำหนดอุปกรณ์ตัวส่ง (Talk Address Group) รหัสที่ใช้ในระบบ IEEE 488 นั้นสามารถที่จะใช้ร่วมกันได้ทั้งรหัสข้อมูลและรหัสคำสั่ง นั่นคือ ข้อมูลที่เหมือนกันมีความหมายได้ 2 อย่าง คือ เมื่อสัญญาณ ATN เป็น Low ข้อมูลที่อยู่ในบัสข้อมูลจะหมายถึงรหัสคำสั่ง แต่ถ้าสัญญาณ ATN เป็น High ข้อมูลที่อยู่ในบัสข้อมูลจะหมายถึงข้อมูลที่เป็นรหัส SACII

1. กลุ่มคำสั่งเจาะจงจุดหมาย (Address Command Group)เป็นคำสั่งที่ส่งไปยังอุปกรณ์ที่เป็นตัวส่งหรือตัวรับที่กำหนดไว้ล่วงหน้าแล้ว ประกอบด้วย

GTL (GO To Local) สั่งให้อุปกรณ์กลับไปสู่สถานะควบคุมด้วยปุ่มปรับที่หน้าปัทม์ตามปกติ

SDC (Selected Device Clear) สั่งให้อุปกรณ์กลับไปสู่สถานะเริ่มต้น

PPC (Parallel Poll Configure)เป็นคำสั่งสำหรับทำการจัดสรรสายสัญญาณของการกระทำกระบวนการตรวจสอบสภาพอุปกรณ์โดยวิธีขนาน โดยใช้งานร่วมกับกลุ่มคำสั่งรอง

GET (Group Execute Trigger) ใช้สั่งเริ่มต้นการทำงานของอุปกรณ์ทีละหลาย ๆ ตัว

TCT (Take Control) กำหนดให้อุปกรณ์ตัวส่งทำหน้าที่เป็นตัวควบคุม

2.กลุ่มคำสั่งครอบคลุม (Universal Command Group) เป็นคำสั่งที่ส่งไปยังอุปกรณ์ทุกตัวที่ต่ออยู่ในระบบ ประกอบด้วย

LLO (Local Lockout) สั่งให้อุปกรณ์ล็อคอยู่ในสถานะควบคุมด้วยปุ่มหน้าปัทม์ตามปกติ

DCL (Device Clear) สั่งให้อุปกรณ์ทุกตัวกลับไปสู่สถานะเริ่มต้น

PPU (Parallel Poll Configure) ใช้ยกเลิกกระบวนการตรวจสอบสภาพแบบขนานทั้งหมด

SPE (Serail Poll Enable) เป็นการเปลี่ยนโหมดการตรวจสอบสภาพแบบอนุกรม โดยในโหมดนี้จะเป็นการส่งสถานะของเครื่องแทนการส่งของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPD (Serail Poll Disable) ยกเลิกกระบวนการตรวจสอบสภาพแบบอนุกรม

3. กลุ่มคำสั่งกำหนดอุปกรณ์ตัวรับ (Listen Address Group) เป็นคำสั่งสำหรับกำหนดอุปกรณ์เป็นตัวรับตามรหัสหมายเลข 0 ถึง 30 และมีคำสั่ง UNL (Unlistener) สำหรับใช้ยกเลิก

4.กลุ่มคำสั่งกำหนดอุปกรณ์ตัวส่ง (Talk Address Group) ใช้สำหรับกำหนดอุปกรณ์เป็นตัวส่งตามรหัสหมายเลข 0 ถึง 30 และมีคำสั่ง UNT (Untalker) ใช้สำหรับยกเลิกเช่นกัน

5.กลุ่มคำสั่งรอง (Secondary Command Group) เป็นคำสั่งที่กำหนดรายละเอียดย่อยของอุปกรณ์แต่ละตัวต่ออยู่ในระบบ ให้มีการทำงานตามจุดประสงค์การใช้งานของอุปกรณ์นั้น คำสั่งรองนี้จะใช้ตามคำสั่งหลักคือ จะใช้หลังจากที่อุปกรณ์ต่าง ๆ ถูกกำหนดไว้ในระบบเรียบร้อยแล้ว

ตารางที่ 2.1

แสดงคำสั่งมาตรฐาน IEEE488 และรหัส ASCII

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

MSD / LSD	0 ASCII	0 MSG	1 ASCII	1 MSG	2 ASCII	2 MSG¹	3 ASCII	3 MSG¹	4 ASCII	4 MSG¹	5 ASCII	5 MSG¹	6 ASCII	6 MSG	7 ASCII	7 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

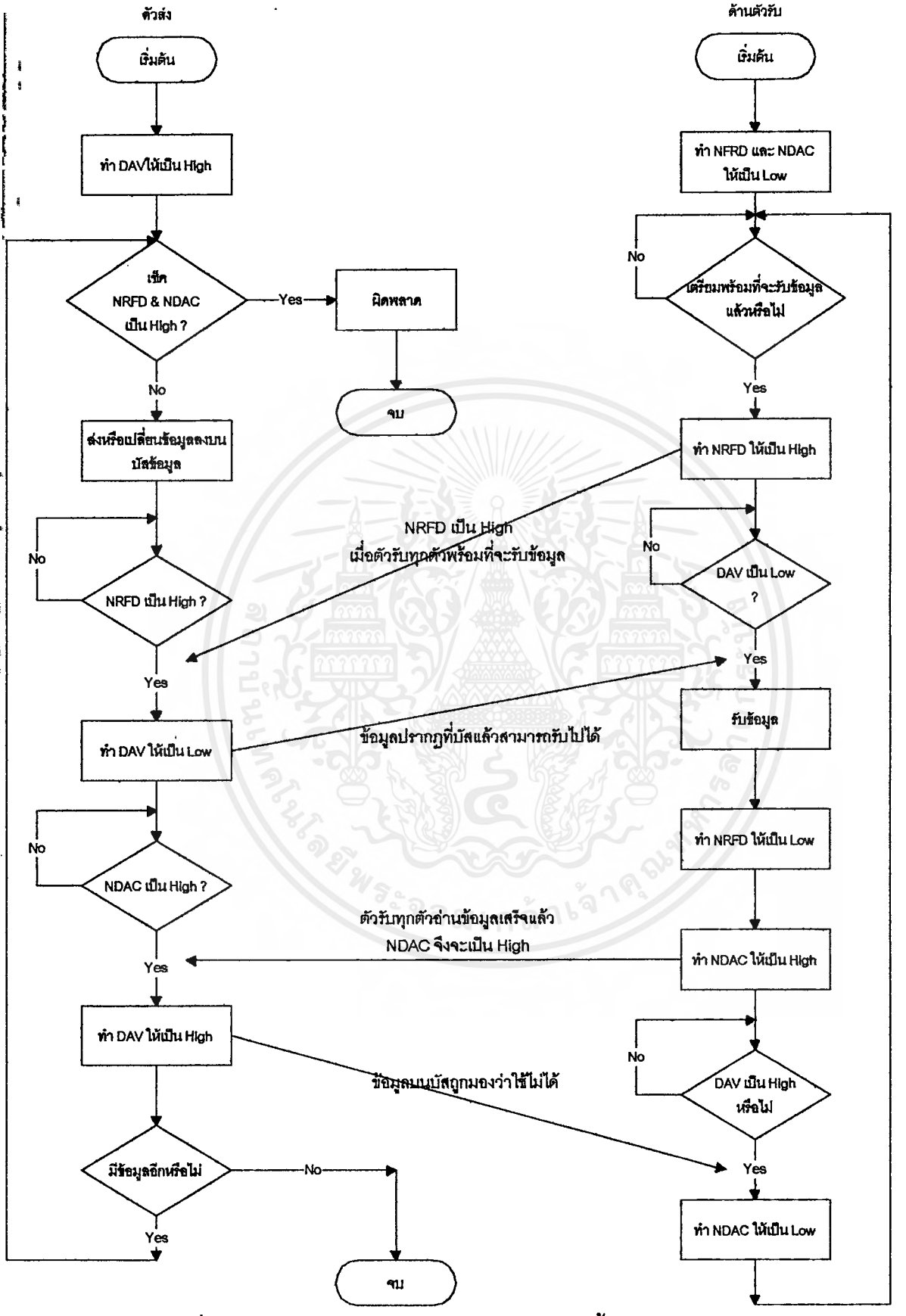
ADDRESSED COMMAND GROUP | UNIVERSAL COMMAND GROUP | LISTEN ADDRESS GROUP | TALK ADDRESS GROUP | SECONDARY COMMAND GROUP

PRIMARY COMMAND GROUP (PCG)

Notes:
1. Device Address messages shown in decimal
2. Message codes are:
   DCL — Devices Clear; LLO — Local Lockout; SDC — Selected Device Clear
   GET — Device Trigger; PPC — Parallel Poll Configure; SPD — Serial Poll Disable
   GTL — Go to Local; PPU — Parallel Poll Unconfigure; SPE — Serial Poll Enable
3. ATN off, Bus data is ASCII; ATN on, Bus data is an IEEE MSG.

This table reproduced with permission:
ICS ELECTRONICS CORPORATION
1450 Koll Circle • Suite 105
San Jose, CA. 95112 • (408) 298-4844

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



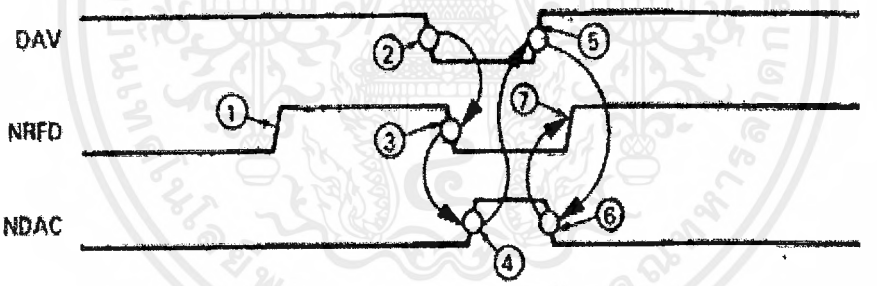
รูปที่ 2.6 แสดง Flow Chat ของการ Handshake ทั้งด้านส่งและด้านรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 ขบวนการแฮนด์เช็ค (Handshake Procedure)

ในการสื่อสารระหว่างภายในระบบ GPIB นั้นจะเป็นการสื่อสารแบบอะซิงโครนัส คือเมื่อมีการรับส่งข้อมูลระหว่างตัวส่งและตัวรับ ตัวส่งจะต้องแจ้งให้ตัวรับทราบว่า ตัวส่งได้ส่งข้อมูลลงไปบนบัสแล้ว และให้ตัวรับทำการเก็บข้อมูลได้ เมื่อตัวรับทำการเก็บข้อมูลเสร็จแล้ว ก็จะต้องแจ้งแก่ตัวส่งให้ทราบว่า ได้รับข้อมูลที่ส่งมาเรียบร้อยแล้ว เพื่อที่ตัวส่งจะได้ทำการหยุดส่งข้อมูลหรือทำการส่งข้อมูลชุดใหม่ลงไปบนบัส กระบวนการเหล่านี้จะเกิดทุกครั้งที่มีการรับส่งข้อมูลในระบบ ซึ่งกระบวนการเหล่านี้ถูกเรียกว่า ขบวนการแฮนด์เช็ค (Handshake Procedure)

ในการพิจารณาถึงขบวนการแฮนด์เช็คนั้น จะทำการพิจารณาถึงระบบที่ไม่ซับซ้อนนัก เพื่อที่จะทำความเข้าใจได้โดยง่าย โดยกำหนดให้ในระบบมีตัวส่งและตัวรับอย่างละหนึ่งตัว ในการสื่อสารระหว่างตัวส่ง และตัวรับนั้น จะมีสายสัญญาณควบคุมการรับส่งข้อมูลอยู่ 3 สัญญาณ คือ NRFD,NDAC, DAV โดยสัญญาณ DAV จะเป็นสัญญาณที่จะชี้ให้เห็นว่าตัรับพร้อมที่จะรับข้อมูลที่ส่งลงมาบนบัสของระบบหรือไม่ สำหรับขั้นตอนของขบวนการแฮนด์เช็คสามารถแสดงได้ดังในรูปที่ 2.7



รูปที่ 2.7 แสดงขบวนการแฮนด์เช็ค

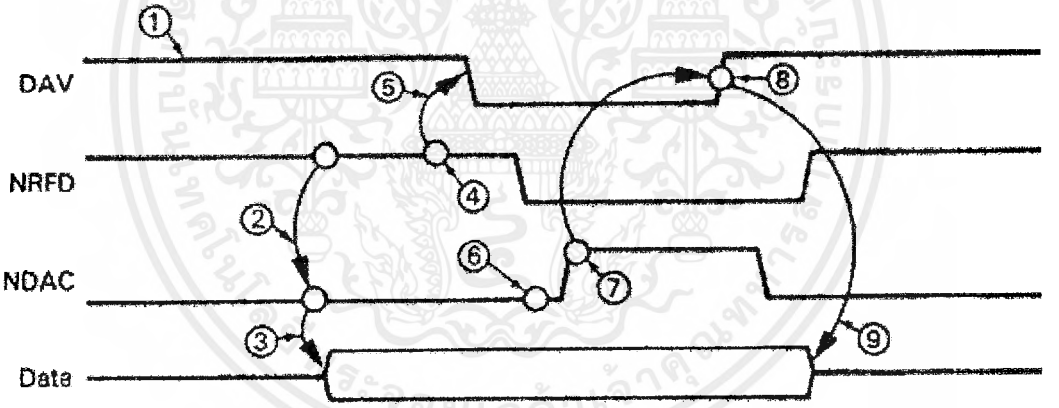
ขบวนการแฮนด์เช็คจะเริ่มขึ้นหลังจากที่ตัวควบคุมทำการบอกให้ระบบทราบว่าอุปกรณ์ตัวไหนที่เป็นตัวรับหรือตัวส่ง เมื่อตัวรับทราบแล้วก็จะส่งสัญญาณ NRFD ให้เป็น High (สถานะที่ 1) เพื่อบอกให้ตัวส่งทราบว่าตัวรับพร้อมที่จะรับข้อมูลแล้ว และตัวส่งก็จะทำการส่งข้อมูลไปบนบัสข้อมูล DIO1 – DIO8 และจะทำการรออยู่ชั่วขณะหนึ่งแล้วตัวส่งจะส่งสัญญาณ DAV ให้เป็น LOW เพื่อแจ้งให้ทราบว่าขณะนี้ตัวส่งได้ส่งข้อมูลลงบนบัสข้อมูลแล้ว (สถานะที่ 2) เมื่อตัวรับทราบว่ามีข้อมูลอยู่บนบัสข้อมูลก็จะส่งสัญญาณ NRFD ให้มีค่าเป็น Low เมื่อตัวรับพร้อมที่จะรับข้อมูล(สถานะที่ 3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากตัวรับได้รับข้อมูลไปเก็บไว้ในบัฟเฟอร์เรียบร้อยแล้วก็จะส่งสัญญาณ NDAC ให้มีค่าเป็น High เพื่อแจ้งให้ทราบว่าตัวรับได้รับข้อมูลเรียบร้อยแล้ว (สถานะที่ 4) เมื่อตัวส่งได้รับสัญญาณ NDAC ที่เป็น High ตัวส่งก็จะทำการส่งสัญญาณ DAV ให้เป็น High เพื่อแจ้งให้ตัวรับไม่ต้องทำการเก็บข้อมูลนั้นอีก (สถานะที่ 5) เมื่อตัวรับได้รับสัญญาณ DAV ที่มีค่าเป็น High ก็จะส่งสัญญาณ NDAC ให้เป็น Low (สถานะที่ 6)ทำให้ข้อมูลในบัสถูกกำจัดออกไป หลังจากนั้นตัวรับก็จะส่งสัญญาณ NRFD ให้เป็น High (สถานะที่ 7) เพื่อบอกให้ทราบว่าตัวรับพร้อมที่จะรับข้อมูลชุดต่อไปที่จะถูกส่งเข้ามาในบัส เป็นอันเสร็จสิ้นขบวนการแฮนด์เช็ค

2.6.1 ขบวนการแฮนด์เช็คเมื่อใช้ไมโครคอมพิวเตอร์เป็นตัวควบคุมและตัวส่ง

ขบวนการแฮนด์เช็คเมื่อใช้ไมโครคอมพิวเตอร์เป็นตัวควบคุมและตัวส่งสามารถแสดงด้วยแผนผังเวลา ดังรูปที่ 2.8 ซึ่งจะทำให้ง่ายต่อการเข้าใจในขั้นตอนการส่งข้อมูล โดยขบวนการดังกล่าวจะเกิดขึ้นหลังจากการกำหนดอุปกรณ์ในระบบแล้ว



รูปที่ 2.8 แสดงแผนผังเวลาของการแฮนด์เช็คเมื่อใช้คอมพิวเตอร์เป็นตัวควบคุมและตัวส่ง

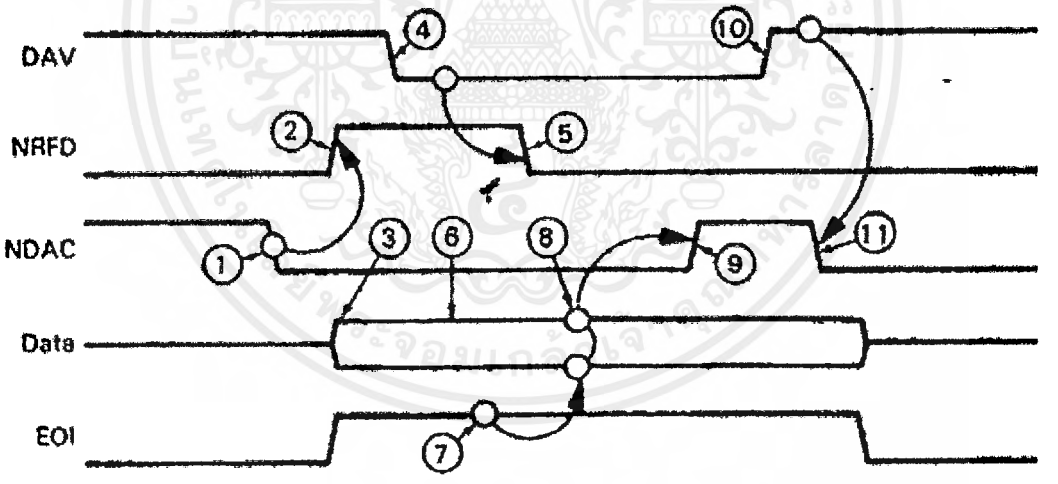
ขบวนการแฮนด์เช็คจะเริ่มขึ้น เมื่อตัวควบคุมส่งสัญญาณ DAV ให้เป็น High (สถานะที่ 1) ซึ่งตัวควบคุมได้เซ็ทให้สัญญาณ DAV ให้มีค่าเป็น High อยู่ก่อนแล้ว หลังจากนั้นตัวควบคุมจะทำการตรวจสอบสัญญาณ NRFD และ NDAC ว่ามีค่าเป็น High ทั้งคู่หรือไม่ (สถานะที่ 2) ถ้าสัญญาณทั้งสองเป็น High ทั้งคู่ แสดงว่าอุปกรณ์ไม่พร้อมที่จะทำงาน ขบวนการแฮนด์เช็คก็จะถูกยกเลิกไป แต่ถ้าที่สถานะที่ 2 หากสัญญาณใดสัญญาณหนึ่งเป็น Low ตัวควบคุมจะทำการส่งข้อมูลลงในบัสข้อมูล (สถานะที่ 3) หลังจากนั้นตัวควบคุมจะทำการตรวจสอบสัญญาณ NRFD ว่าเป็น High

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือไม่ (สถานะที่ 4) ถ้าสัญญาณ NRFD เป็น High ตัวควบคุมก็จะส่งสัญญาณ DAV ให้มีค่าเป็น Low เพื่อบอกให้ตัวรับทราบว่ามีข้อมูลอยู่ในบัสข้อมูล (สถานะที่ 5) แต่ถ้าที่สถานะที่ 4 สัญญาณ NRFD มีค่าเป็น Low แสดงว่าขบวนการแฮนด์เช็ค เกิดความผิดพลาดขึ้น จะต้องทำการเริ่มต้นใหม่จากสถานะที่ 5 ตัวควบคุมจะรอเวลาให้ตัวรับทำการเก็บข้อมูล เมื่อถึงเวลาที่กำหนดตัวควบคุมจะทำการตรวจสอบสัญญาณ ว่าสัญญาณ NDAC ถูกเปลี่ยนให้เป็น High ในเวลาที่กำหนดหรือไม่ (สถานะที่ 6) ถ้าตัวรับไม่ได้รับข้อมูลในเวลาที่กำหนดขบวนการแฮนด์เช็คจะถูกยกเลิกไป แต่ถ้าตัวรับได้รับข้อมูลภายในเวลาที่กำหนดตัวรับจะทำการเปลี่ยนสัญญาณ NDAC ให้เป็น High (สถานะที่ 7) ตัวควบคุมก็จะทำการตอบสนองโดยการเปลี่ยนสัญญาณ DACให้เป็น High (สถานะที่ 8) และตัวควบคุมก็จะทำการลบข้อมูลที่อยู่ในบัสข้อมูลออกไป (สถานะที่ 9) เป็นการเสร็จสิ้นขบวนการแฮนด์เช็คดังกล่าว

2.6.2 ขบวนการแฮนด์เช็คเมื่อใช้ไมโครคอมพิวเตอร์เป็นตัวควบคุมและเป็นตัวรับ

ขบวนการแฮนด์เช็คแบบนี้สามารถเขียนแทนด้วยแผนผังเวลา ดังรูปที่ 2.9



รูปที่ 2.9 แสดงแผนผังเวลาของขบวนการแฮนด์เช็คเมื่อใช้ไมโครคอมพิวเตอร์เป็นตัวควบคุมและตัวรับ

ขบวนการแฮนด์เช็คเริ่มขึ้นโดยตัวควบคุมรับรู้ว่าตัวส่งจะทำการส่งข้อมูล ตัวควบคุมจะส่งสัญญาณ NDAC ให้มีค่าเป็น Low เพื่อบอกให้ทราบว่าตัวควบคุมยังไม่ได้รับข้อมูล (สถานะที่ 1) ต่อจากนั้นตัวควบคุมจะส่งสัญญาณ NRFD ให้เป็น High เพื่อบอกให้ตัวส่งทราบว่าตัวควบคุมพร้อมที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะรับข้อมูลแล้ว (สถานะที่ 2) ตัวส่งจะทราบได้ทันทีว่า ขณะนี้สามารถที่จะส่งข้อมูลลงบนบัสข้อมูลได้แล้ว (สถานะที่ 3)จากการที่สัญญาณ NRFD มีลอจิกเป็น High และสัญญาณ NDAC มีลอจิกเป็น Low ขั้นตอนต่อไปตัวควบคุมจะทำการรอเวลาให้ตัวส่งทำการส่งข้อมูลลงบนบัสข้อมูลให้เสร็จ และจะทำการตรวจสอบด้วยว่าเกินเวลาที่กำหนดหรือไม่ หากเกินเวลาที่กำหนดก็จะทำการออกจากขบวนการแฮนด์เช็ค หากยังไม่เกินก็จะรอจนหมดเวลาหรือจนกว่าสัญญาณ DAV จะเป็น Low (สถานะที่ 4) เมื่อสัญญาณ DAV มีลอจิกเป็น Low ตัวควบคุมก็จะตอบรับโดยการทำให้สัญญาณ NRFD มีค่าเป็น Low (สถานะที่ 5)เพื่อบอกให้ตัวส่งทราบว่าตัวควบคุมพร้อมที่จะเริ่มทำการเก็บข้อมูลแล้ว (สถานะที่ 6) หลังจากนั้นตัวควบคุมจะทำการตรวจสอบสัญญาณ EOI ว่า ข้อมูลที่ตัวส่งทำการส่งมานั้นหมดหรือยัง โดยตัวส่งจะทำการเปลี่ยนสัญญาณ EOI ให้มีลอจิกเป็น Low เมื่อข้อมูลไบต์สุดท้ายถูกส่งลงไปบนบัสข้อมูล (สถานะที่ 7)หากสัญญาณ EOI ยังคงเป็น High ตัวควบคุมจะทำการเก็บข้อมูล ในบัสข้อมูลต่อไป (สถานะที่ 8)และเมื่อตัวควบคุมได้ทำการเก็บข้อมูลเรียบร้อยแล้ว ตัวควบคุมจะเปลี่ยนลอจิกของสัญญาณ NDAC ให้เป็น High (สถานะที่ 9) เมื่อตัวควบคุมทำให้สัญญาณ NDAC มีลอจิกเป็น High แล้ว ตัวส่งจะทำการลบข้อมูลบนบัสข้อมูลออกโดยการเปลี่ยนสัญญาณ DAV ให้มีค่าเป็น High (สถานะที่ 10) ตัวควบคุมก็จะทำการนำข้อมูลที่ได้ไปใช้งานและเปลี่ยนสัญญาณ NDAC ให้มีค่าเป็น Low (สถานะที่ 11)เป็นการสิ้นสุดขบวนการแฮนด์เช็ค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การใช้โปรแกรม NI – 488.2

3.1 การใช้โปรแกรม NI - 488.2M กับ Windows 95

การใช้โปรแกรม NI -488.2M กับ Windows 95 ดังแสดงในรูปที่ 3.1 โดยแสดงการทำงานเป็นขั้นตอน โดยจะขึ้นอยู่กับการติดตั้งของผู้ใช้ที่จะเลือกใช้ ในที่นี้สามารถแบ่ง ได้ 2 โหมดการทำงาน

- User Mode คือโหมดผู้ใช้ ซึ่งสามารถใช้โปรแกรมที่เขียนขึ้นแล้วทำการติดต่อกับในส่วนของ Kernal Mode โดยการเลือกใช้ gpib-32.dll

- Kernal Mode คือโหมดการติดต่อกับอุปกรณ์ GPIB โดยใช้โปรแกรมที่มีมากับ Driver ของการ์ดซึ่งเป็นโปรแกรมที่ใช้สำหรับทดสอบอุปกรณ์ว่าต่ออยู่กับระบบหรือไม่ VxD จะหมายถึง Virtual Device Driver ทั่ว ๆไป โดย X จะแสดงถึงชนิดของ Device Driver นั้น ตัวอย่างเช่น VDD จะหมายถึง Virtual Device Driver สำหรับอุปกรณ์แสดงผล VTD จะหมายถึง Virtual Device Driver สำหรับอุปกรณ์ Timer และ VPD จะหมายถึง Virtual Device Driver สำหรับเครื่องพิมพ์ Windows จะใช้ Virtual Device Driver เพื่อสนับสนุนการทำงานแบบมัลติทาส์กของ Application แบบ MSDOS based โดยการจำลองส่วนประกอบทางฮาร์ดแวร์บนระบบให้ Application MSDOS เห็น Application แบบ MSDOS based แต่ละตัวเมื่อทำงานภายใต้ Windows 95 ดังนั้นจะเรียกแทนด้วยคำว่า MD-DOS Virtual Machine หรือ VM จะทำงานเพื่อเชื่อมต่อกับ Windows ในการ Interrupt และการทำงานของ I /O ให้แก่ Application โดยไม่ทำลายการทำงานของ Application อื่นที่ทำงานอยู่

Windows 95 ได้รับประโยชน์จากการจัดเตรียมการสนับสนุน Device Driver จำนวนมาก ซึ่งอยู่ในรูปแบบของ VxD (Virtual Device Driver) ในสภาวะแวดล้อมของ Windows แทนที่จะใช้ Device Driver ของ MSDOS ที่ทำงานใน real mode การทำงานที่เคยถูกสนับสนุนแบบเดิมจึงถูกเปลี่ยนมาใช้แบบ VxD ใน Windows 95 ซึ่งได้แก่องค์ประกอบดังต่อไปนี้

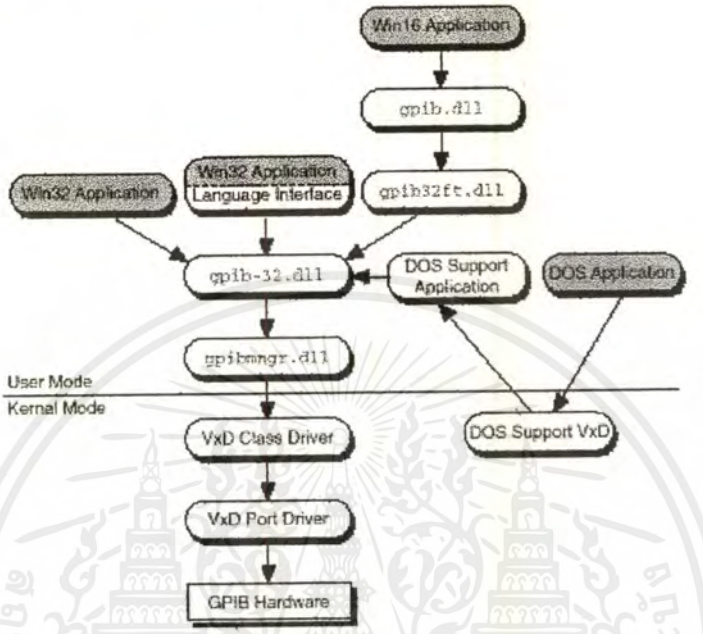
- ปรับปรุงสมรรถนะการทำงานซึ่งเป็นผลมาจากการใช้โค้ดคำสั่งแบบ 32 บิตและการลดความจำเป็นในการสับเปลี่ยนไปมาระหว่างการทำงานใน real mode และ protect mode

- ลดการใช้หน่วยความจำปกติโดยการจัดเตรียมการทำงานของ Device Driver และโปรแกรม TSR ใน protect mode

- ปรับปรุงเสถียรภาพและความน่าเชื่อถือในการทำงานให้ดีขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Virtual Device Driver ใน Windows 95 จะถูกบ่งชี้โดยนามสกุลไฟล์ . VXD ส่วนของ Windows3.1 จะถูกบ่งชี้โดยนามสกุลไฟล์ .386



รูปที่ 3.1 แสดงการใช้ NI-488.2M กับ Windows 95

3.2 ฟังก์ชั่นในการติดต่อกับ NI - 488.2

Tibclr = function (ud : integer) : integer ; stdcall ;

- Tibclr คือ การเคลียร์สถานะของอุปกรณ์
- ud คือการบอกตำแหน่งชนิดของอุปกรณ์

Tibdev = function (ud : integer ;
pad : integer ;
sad : integer ;
tmo : integer ;
eot : integer ;
eos : integer) : integer ; stdcall

- Tibdev คือ การติดต่อและกำหนดค่าเริ่มต้นให้กับอุปกรณ์
- ud คือ การบอกตำแหน่งชนิดของอุปกรณ์
- pad คือ การบอกตำแหน่งหลัก GPIB ของอุปกรณ์
- sad คือ การบอกตำแหน่งสำรอง GPIB ของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- eot คือ การบอกตำแหน่งสิ้นสุดในการติดต่อ

- eos คือ การบอกตำแหน่งข้อความสุดท้าย

Tibrd = function (ud : integer ;

Var rdbuf ;

Cnt : longint) : integer ; stdcall ;

- Tibrd คือ การอ่านข้อมูลจากอุปกรณ์เข้ามาเก็บไว้ในหน่วยความจำ

- ud คือ การบอกตำแหน่งชนิดของอุปกรณ์

- rdbuf คือ ชื่อของหน่วยความจำที่ใช้เก็บข้อมูล

- cnt คือ การกำหนดขนาดของการอ่านข้อมูล

Tibtrg = function (ud : integer) : integer; stdcall ;

- Tibtrg คือ การเลือกชนิดของอุปกรณ์

- ud คือ การบอกตำแหน่งชนิดของอุปกรณ์

Tibwrt = function (ud : integer ;

var wrtbuf ;

cnt : longint) : integer ; stdcall ;

- Tibwrt คือ การส่งข้อมูลจากผู้ใช้ไปให้อุปกรณ์

- ud คือ การบอกตำแหน่งชนิดของอุปกรณ์

- wrtbuf คือ ชื่อของหน่วยความจำที่ใช้ในการส่งข้อมูล

- cnt คือ การกำหนดขนาดการส่งข้อมูล

Tibwait = funtion (ud : integer ;

mask : integer) : integer ; stdcall ;

- Tibwait คือ การรอเวลาในการติดต่อหรือการดำเนินงานของ GPIB

- ud คือ การบอกตำแหน่งชนิดของอุปกรณ์

- mask คือ บิตแสดงสถานะของ GPIB ที่บอกถึงเหตุการณ์ในการรอคอยใน

การ ติดต่อกับ GPIB

Tdev Clear = procedure (boardID : integer ;

padlist : AddrList ;

resultlist : AddrList ;

limit : integer) ; stdcall ;

- Tdev Clear คือ การเคลียร์สถานะของอุปกรณ์

- boardID คือ การบอกตำแหน่งของบอร์ด GPIB ที่จะทำการติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- address คือการบอกตำแหน่งของอุปกรณ์
- buffer คือชื่อของตำแหน่งที่ใช้ในการเก็บข้อมูล
- cnt คือความยาวในการอ่านข้อมูล

```
TSend = procedure ( boardID ; integer ;
          address : word ;
          var buffer ;
           datacnt : longint ;
           eotmode : integer ) ; stdcall ;
```

- Tsend คือการส่งข้อมูลไปให้อุปกรณ์
- boardID คือการบอกตำแหน่งการติดต่อ GPIB
- address คือการบอกตำแหน่งของอุปกรณ์
- buffer คือชื่อของตำแหน่งที่ใช้ในการส่งข้อมูล
- datacnt คือความยาวในการส่งข้อมูล
- eotmode คือการบอกตำแหน่งสุดท้ายในการส่งข้อมูล

```
TSendIFC = produre ( boardID : integer ) ; stdcall ;
```

- TSendIFC คือการรีเซ็ท GPIB โดยการส่งสัญญาณ interface เคลียร์
- boardID คือการบอกตำแหน่งการติดต่อ GPIB

```
TWaitSRQ = procedure ( boardID : integer ;
             var result : integer ) ; stdcall ;
```

- TWaitSRQ คือการคอยจนกระทั่งอุปกรณ์ยืนยันพร้อมที่จะขอบริการจาก GPIB
- boardID คือการบอกตำแหน่งการติดต่อ GPIB
- result คือสถานะของผลลัพธ์จะเป็น '1' เมื่อมีการยืนยันและจะเป็น '0' เมื่อไม่มีการยืนยัน

3.3 การเขียนโปรแกรมบน NI-488 หรือ NI-488.2

NI-488 และ NI-488.2 เป็น Software Driver ของการ์ด GPIB-PCII / IIA ซึ่งสามารถนำมาใช้งานกับ Delphi ได้ทั้ง 2 โปรแกรมแตกต่างกันเพียงแต่รูปแบบของ function และ procedure ที่มีให้เรียกใช้งานอยู่ในไฟล์. DLL ดังจะได้แสดงตัวอย่างการใช้งานของโปรแกรมทั้ง 2 ในหัวข้อต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 คำสั่งที่ใช้ควบคุมเครื่องมือวัดของอุปกรณ์

ระบบเครื่องมือวัดตามมาตราฐาน IEEE-488 ที่ทำการศึกษานั้น นอกจากจะมีตัวควบคุมระบบแล้ว ยังมีอุปกรณ์ที่ต่ออยู่ในระบบซึ่งได้แก่อุปกรณ์ที่เป็นเครื่องมือพื้นฐานทางอิเลกทรอนิกส์ที่ใช้สำหรับการทดลองทางอิเลกทรอนิกส์ โดยอุปกรณ์เหล่านี้นอกจากจะสามารถควบคุมได้ผ่านหน้าปัทม์แล้ว ยังสามารถควบคุมได้โดยผ่านระบบบัสตามมาตราฐาน IEEE-488 หรือ GPIB ได้ด้วย ซึ่งในระบบจะประกอบไปด้วย

1. Programmable Power Supply ของบริษัท HAMEG รุ่น HM 8142 ทำหน้าที่เป็นแหล่งจ่ายไฟฟ้ากระแสตรง เพื่อใช้เป็นไฟเลี้ยงให้แก่วงจรที่จะทำการทดลอง โดยมีแหล่งจ่ายไฟอยู่ 2 ตัว และมีคำสั่งควบคุมฟังก์ชั่นในการทำงานผ่านระบบบัสแบบ IEEE-488 คือ

คำสั่ง : RM1 / RM0

รูปแบบ : RM1

ฟังก์ชัน : สั่งให้ Power Supply ทำงานในโหมดรีโมท โดยไม่สามารถควบคุมผ่านหน้าปัทม์ได้ แต่สามารถควบคุมการทำงานได้โดยผ่านระบบบัสแบบ IEEE-488 เท่านั้น และสามารถออกจากโหมดรีโมทได้โดยการส่งคำสั่ง RM0 หรือการกดปุ่ม LOCAL ที่หน้าปัทม์ของเครื่อง

รูปแบบ : RM0

ฟังก์ชัน : ทำการยกเลิกโหมดรีโมทให้เครื่องกลับสู่สถานะ LOCAL (สามารถควบคุมการทำงานของเครื่องได้โดยผ่านปุ่มหน้าปัทม์)

ข้อสังเกต : คำสั่ง RM0 จะมีผลต่อคำสั่ง LK1

คำสั่ง : MX1 / MX0

รูปแบบ : MX1

ฟังก์ชัน : เปลี่ยนโหมดรีโมทไปเป็นโหมดแบบผสม ในโหมดนี้สามารถที่จะควบคุมการทำงานของเครื่องได้โดยการส่งคำสั่งผ่านระบบบัสแบบ IEEE-488 และการควบคุมผ่านทางหน้าปัทม์ของเครื่อง

รูปแบบ : MX0

ฟังก์ชัน : ยกเลิกโหมดแบบผสมและกลับเข้าสู่สถานะปกติที่ควบคุม โดยผ่านระบบบัสแบบ IEEE-488

คำสั่ง : LK1 / LK0

รูปแบบ : LK1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน : สั่งให้เครื่องเข้าสู่โหมดรีโมท โดยไม่สามารถกดปุ่ม LOCAL เพื่อที่จะให้เครื่องกลับเข้าสู่โหมด LOCAL ได้

รูปแบบ : LK0

ฟังก์ชัน : สั่งให้เครื่องออกจากโหมดรีโมทที่ไม่สามารถกดปุ่ม LOCAL ทำให้สามารถกดปุ่ม LOCAL เพื่อที่จะสามารถควบคุมการทำงานผ่านหน้าปัทม์ของเครื่องได้

ข้อสังเกต : โหมดรีโมทที่ไม่สามารถกดปุ่ม LOCAL นี้สามารถถูกยกเลิกได้โดยคำสั่ง RM0

คำสั่ง : SU1 และ SU2

รูปแบบ : SU1 W.mV.mV หรือ SU2 01.34

ฟังก์ชัน : ตั้งค่าศักดาไฟฟ้าของแหล่งจ่ายที่ 1 หรือ 2

คำสั่ง : SI1 และ SI2

รูปแบบ : SI1 A.mAmAmA หรือ SI 0.123

ฟังก์ชัน : ตั้งค่ากระแสไฟฟ้าของแหล่งจ่ายตัวที่ 1 หรือ 2 เพื่อจำกัดค่ากระแสไฟฟ้า

คำสั่ง : RU1 และ RU2

รูปแบบ : RU1 หรือ RU2

ฟังก์ชัน : ค่าศักดาไฟฟ้าที่ส่งกลับโดยเครื่อง จะเป็นค่าศักดาไฟฟ้าที่ถูกกำหนดค่าโดยคำสั่งกำหนดค่า (SET)

ข้อสังเกต : ใช้คำสั่ง MUX เพื่อที่จะถามค่ากระแสไฟฟ้าจริงที่วัดได้จากเอาท์พุท

คำสั่ง : RI1 และ RI2

รูปแบบ : RI1 หรือ RI2

ค่าที่ส่งกลับ : I1_1.000A หรือ I2_0.012A

ฟังก์ชัน : ค่ากระแสไฟฟ้าที่ส่งกลับโดยเครื่อง จะเป็นค่ากระแสไฟฟ้าที่ถูกกำหนดค่าโดยคำสั่งจำกัดค่ากระแสไฟฟ้า (SI1 หรือ SI2)

ข้อสังเกต : ใช้คำสั่ง MIX เพื่อถามค่ากระแสไฟฟ้าที่วัดได้จากเอาท์พุทจริง

คำสั่ง : MU1 และ MU 2

รูปแบบ : MU1 หรือ MU 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าที่ส่งกลับ : U1:12.34V หรือ U2:12.24V

ฟังก์ชัน : ค่าศักดาไฟฟ้าที่ส่งกลับโดยเครื่อง ซึ่งเป็นค่าที่วัดได้จากเอาท์พุทจริง

ข้อสังเกต : คำสั่ง RUX ใช้เพื่อถามถึงค่าศักดาไฟฟ้าที่ทำการกำหนด

คำสั่ง : MI1 และ MI2

รูปแบบ : MI1 หรือ MI2

ค่าที่ส่งกลับ : I1=+1.000A หรือ I2=-0.123A

ฟังก์ชัน : ค่ากระแสไฟฟ้าที่ส่งกลับโดยเครื่องเป็นค่ากระแสไฟฟ้าที่วัดได้

ข้อสังเกต : คำสั่ง RIX ใช้เพื่อถามถึงค่ากระแสไฟฟ้าที่จำกัดเอาไว้ ถ้าเอาท์พุทถูกปิดไว้ ค่าที่ส่งกลับจะเป็น I1=_1.000A

คำสั่ง : TRU

รูปแบบ : TRU

ฟังก์ชัน : กำหนดค่าศักดาไฟฟ้าของแหล่งจ่ายตัวที่ 1 และ 2 ให้มีค่าเท่ากันตามความต้องการ

คำสั่ง : TRI

รูปแบบ : TRI

ฟังก์ชัน : กำหนดค่ากระแสไฟฟ้าของแหล่งจ่ายตัวที่ 1 และ 2 ให้มีค่าเท่ากันตามต้องการ

คำสั่ง : SR1 และ SR0

รูปแบบ : SR1

ฟังก์ชัน : ทำการ enable โหมด service request โดยไม่มีการเปลี่ยนแปลงใด ๆ ในสถานะของอุปกรณ์ที่จะมีผลต่อสัญญาณ SRQ (service request)

รูปแบบ : SRO

ฟังก์ชัน : ทำการยกเลิกโหมด service request

คำสั่ง : STA

รูปแบบ : STA

ค่าที่ส่งกลับ : OP1/OSR1/OER0/1CV1/CC1/CV2/CC2/RM0/1

OP0 สวิทช์เอาท์พุทปิด

OP1 สวิทช์เอาท์พุทเปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SQ1 แสดงสถานะของอุปกรณ์ที่เปลี่ยนแปลง (CV ไปเป็น CC หรือ OP1 ไปเป็น OP0 เป็นต้น) (จะทำงานเฉพาะเมื่อ SRQ นั้น enable หรือมีค่าเป็น 1 ,ดูคำสั่ง SR1)

SQ0 เมื่อสัญญาณ service request (SRQ) ถูกทำให้ enable แสดงว่าไม่มีการเปลี่ยนแปลงสถานะของเครื่องมือ

ER0 ไม่มีความผิดพลาด

ER1 ความร้อนสูงเกิน

CV1 แหล่งจ่ายตัวที่ 1 ทำการจ่ายศักดาไฟฟ้าคงที่

CC1 แหล่งจ่ายตัวที่ 1 ทำการจ่ายกระแสไฟฟ้าคงที่

CV2 แหล่งจ่ายตัวที่ 2 ทำการจ่ายศักดาไฟฟ้าคงที่

CC2 แหล่งจ่ายตัวที่ 2 ทำการจ่ายกระแสไฟฟ้าคงที่

RM1 อุปกรณ์อยู่ในโหมดรีโมท

Rm0 อุปกรณ์ไม่อยู่ในโหมดรีโมท

ฟังก์ชัน : สั่งให้เครื่องส่งค่าสตริงที่เก็บสถานะปัจจุบันของเครื่องมาให้

คำสั่ง : OP1 และ OP0

รูปแบบ : OP1

ฟังก์ชัน : สวิทช์เอาท์พุทเปิด

รูปแบบ : OP0

ฟังก์ชัน : สวิทช์เอาท์พุทปิด

คำสั่ง : CLR (clear)

รูปแบบ : CLR

ฟังก์ชัน : ยกเลิกฟังก์ชั่นทั้งหมดของเครื่อง และทำการเริ่มต้นใหม่ที่สถานะศูนย์ (zero status) ในโหมดรีโมท คีย์บอร์ดไม่สามารถใช้งานได้และสวิตช์เอาท์พุทปิด รวมทั้งศักดาไฟฟ้าและกระแสไฟฟ้าจะถูกตั้งค่าให้เป็นศูนย์

คำสั่ง : VER

รูปแบบ : VER

ค่าที่ส่งกลับ : sw Vx.xhw Vx.xxxxxxx HAMEG /Paris KRP&VM

ฟังก์ชัน : แสดงเวอร์ชั่นของซอฟต์แวร์และฮาร์ดแวร์ของเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง : ID?

รูปแบบ : ID?

ค่าที่ส่งกลับ : HM8142-1

ฟังก์ชัน : แสดงถึงชื่อรุ่นของเครื่อง

2. Programmable Multimeter ของบริษัท HAMEG รุ่น HM 8112-2 ใช้สำหรับวัดค่าพารามิเตอร์ต่าง ๆ ทางไฟฟ้า โดยมีคำสั่งควบคุมฟังก์ชั่นการทำงานผ่านระบบบัสแบบ IEEE-488 คือ

คำสั่ง : VD

รูปแบบ : VD

ฟังก์ชัน : การวัดค่าศักดาไฟฟ้ากระแสตรง

คำสั่ง : VA

รูปแบบ : VA

ฟังก์ชัน : การวัดศักดาไฟฟ้ากระแสสลับ

คำสั่ง : O2

รูปแบบ : O2

ฟังก์ชัน : การวัดค่าความต้านทาน

คำสั่ง : ID

รูปแบบ : ID

ฟังก์ชัน : การวัดไฟฟ้ากระแสตรง

คำสั่ง : IA

รูปแบบ : IA

ฟังก์ชัน : การวัดไฟฟ้ากระแสสลับ

คำสั่ง : TC

รูปแบบ : TC

ฟังก์ชั่น : การวัดอุณหภูมิแบบเซลเซียส

คำสั่ง : TK

รูปแบบ : TK

ฟังก์ชัน : การวัดอุณหภูมิแบบเคลวิน

คำสั่ง : TF

รูปแบบ : TF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน : การวัดอุณหภูมิแบบฟาเรนไฮต์

คำสั่ง : RX

รูปแบบ : RX

ฟังก์ชัน : เลือกช่วงการวัดค่าต่าง ๆ โดย " X " คือช่วงการวัดที่ต้องการ คือ

R1 ช่วงเวลา 0.2 Vdc. Vac, K

R2 ช่วงเวลา 2 Vdc, Vac , K

R3 ช่วงเวลา 20 Vdc , Vac, K

R4 ช่วงเวลา 200Vdc, Vac, K

R5 ช่วงเวลา 1000 Vdc,Vac, K

R6 ช่วงเวลา 10000Vdc,Vac,K

คำสั่ง : A0 / A1

รูปแบบ : A0 /A1

ฟังก์ชัน : ไม่เลือกฟังก์ชั่นการปรรับช่วงการวัดอัตโนมัติ เลือกฟังก์ชั่นการปรับช่วงการวัดอัตโนมัติ

คำสั่ง : TX

รูปแบบ : TX

ฟังก์ชัน : เลือกช่วงเวลาและจำนวนหลักในการแสดงผลที่ได้จากการวัด

T1 ช่วงเวลา 100 ms การแสดวงผล 5 1/2 หลัก

T2 ช่วงเวลา 1 ms การแสดงผล 5 1/2 หลัก

T3 ช่วงเวลา 1 ms การแสดงผล 6 1/2 หลัก

T4 ช่วงเวลา 10 ms การแสดงผล 6 1/2 หลัก

คำสั่ง : Z0

รูปแบบ : Z0

ฟังก์ชัน : เลือกการปรับค่าออฟเซ็ท

คำสั่ง : S0

รูปแบบ : S0

ฟังก์ชัน : เริ่มการวัดแบบต่อเนื่อง

คำสั่ง : S1

รูปแบบ : S1

ฟังก์ชัน : หยุดการวัดแบบต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง : MO

รูปแบบ : MO

ฟังก์ชัน : เลือกช่องมัลติเพลกเซอร์

คำสั่ง : M0 – M9

รูปแบบ : MO – M9

ฟังก์ชัน : เลือกช่อง 0 – 9

คำสั่ง : L0

รูปแบบ : L0

ฟังก์ชัน : ส่งเฉพาะข้อมูลบล็อกแรก

คำสั่ง : L1

รูปแบบ : L1

ฟังก์ชัน : ส่งข้อมูลทั้งสองบล็อกข้อมูล (ข้อมูลที่วัดและตัวอักษรจะอยู่ในข้อมูลบล็อกแรก ส่วนข้อมูลที่เป็นการโปรแกรมจะอยู่ในข้อมูลบล็อกที่ 2)

รูปแบบข้อมูลที่ส่งนั้น จะประกอบด้วยข้อมูล 2 บล็อกข้อมูล โดยข้อมูลบล็อกแรกจะประกอบด้วยตัวอักษร 12 ตัว และข้อมูลบล็อกที่ 2 จะประกอบด้วย ตัวอักษร 20 ตัวอักษรบวกกับตัวอักษรที่ปิดท้ายข้อมูล โดยรูปแบบข้อมูลจะมีลักษณะดังนี้

+X.XXXXXXXE+XVDR1A0T1S0QM0X0P0B0

โดยข้อมูลบล็อกแรกจะเป็นข้อมูลที่ได้จากการวัด (+X.XXXXXXXE+X) และข้อมูลบล็อกที่ 2 จะเป็นฟังก์ชั่นในการวัดของเครื่องหรือสถานะของเครื่องนั่นเอง

คำสั่ง : Q0

รูปแบบ : Q0

ฟังก์ชั่น : ไม่ส่งสัญญาณ SRQ (service request)

คำสั่ง : Q1

รูปแบบ : Q1

ฟังก์ชั่น : ส่งสัญญาณ SRQ (service request)

คำสั่ง :NVXXXXXXXX

รูปแบบ : NVXXXXXXXX

ฟังก์ชั่น : ทำการปรับแต่งเครื่องผ่านระบบบัสแบบ IEEE-488 โดยหลักคำสั่ง NV นั้น เครื่องจะถือว่าเลขจำนวนเต็ม 6 หลักเป็นค่าที่ใช้สำหรับการปรับแต่ง

คำสั่ง : P1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบ : P1

ฟังก์ชั่น : แสดงค่าออฟเซ็ท คำนวนจาก R = X - C

คำสั่ง : P2

รูปแบบ : P2

ฟังก์ชั่น : แสดงเปอร์เซนต์ความเบี่ยงเบน คำนวนจาก R = 100 (X-C) / C

คำสั่ง : P3

รูปแบบ : P3

ฟังก์ชั่น : แสดง dB คำนวนจาก R = 20 LOG (X/C)

คำสั่ง : P4

รูปแบบ : P4

ฟังก์ชั่น : แสดง dBm คำนวนจาก R = 20 LOG (X-C)

เมื่อ C = 0.775 V สำหรับค่าศักดาไฟฟ้า และ C = 1.29 mA สำหรับค่ากระแสไฟฟ้า

คำสั่ง : PXEN

รูปแบบ : PXEN

ฟังก์ชั่น : เลือกค่าการวัดที่ X เป็นค่าคงที่ ถ้า P2-4;X มีค่าเท่ากับ 1, 2 , 3 ตามลำดับ

คำสั่ง : ID?

รูปแบบ : ID?

ฟังก์ชั่น : ทำการส่งหมายเลขรุ่นของเครื่องลงบนหน่วยความจำของเครื่อง

คำสั่ง : STA?

รูปแบบ :STA?

ฟังก์ชั่น : ทำการส่งสถานะของเครื่อง (ข้อมูลบล็อกที่ 2)

คำสั่ง : D0/D1

รูปแบบ : D0/D1

ฟังก์ชั่น : ปิด / เปิด หน้าจอ

คำสั่ง : BX

รูปแบบ : BX

ฟังก์ชั่น : ส่งค่าของการสวิทซ์ที่เลือกครั้งสุดท้าย โดยที่ X = 1,....,9,A,....,F

คำสั่ง : EOI

รูปแบบ : EOI

ฟังก์ชั่น : เลือกให้มีการส่งสัญญาณ EOI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง : EOS1 / EOS2

รูปแบบ : EOS1/ EOS2

ฟังก์ชั่น : เลือกให้ข้อมูลที่ส่งมีการส่งค่า EOS1 หรือ EOS2 ปิดท้ายข้อมูล

คำสั่ง : END

รูปแบบ : END

ฟังก์ชั่น : ยกเลิกตัวอักษรที่ปิดท้ายสตริงข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

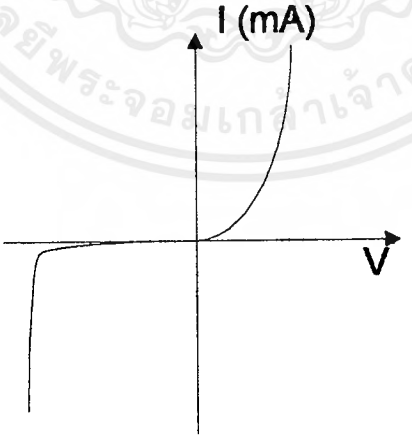
บทที่ 4
ทฤษฎีการทำงานพื้นฐานของอุปกรณ์

4.1 ไดโอดชนิดหัวต่อ P -N (P -N junction diode)

เป็นไดโอดที่สร้างขึ้นจากการนำเอาสารกึ่งตัวนำชนิด N มาแล้วแพร่อนุภาคอะตอมของสารบางชนิดเข้าไปในเนื้อสารเกิดเป็นสาร P ขึ้นบางส่วน แล้วจึงต่อขั้วออกใช้งาน ไดโอดชนิดนี้มีบทบาทในวงจรอิเล็กทรอนิกส์และมีการใช้งานกันอย่างแพร่หลาย

รูปที่ 4.1 ลักษณะโครงสร้างของไดโอดชนิดหัวต่อและสัญลักษณ์

4.1.1 คุณลักษณะสมบัติของไดโอด



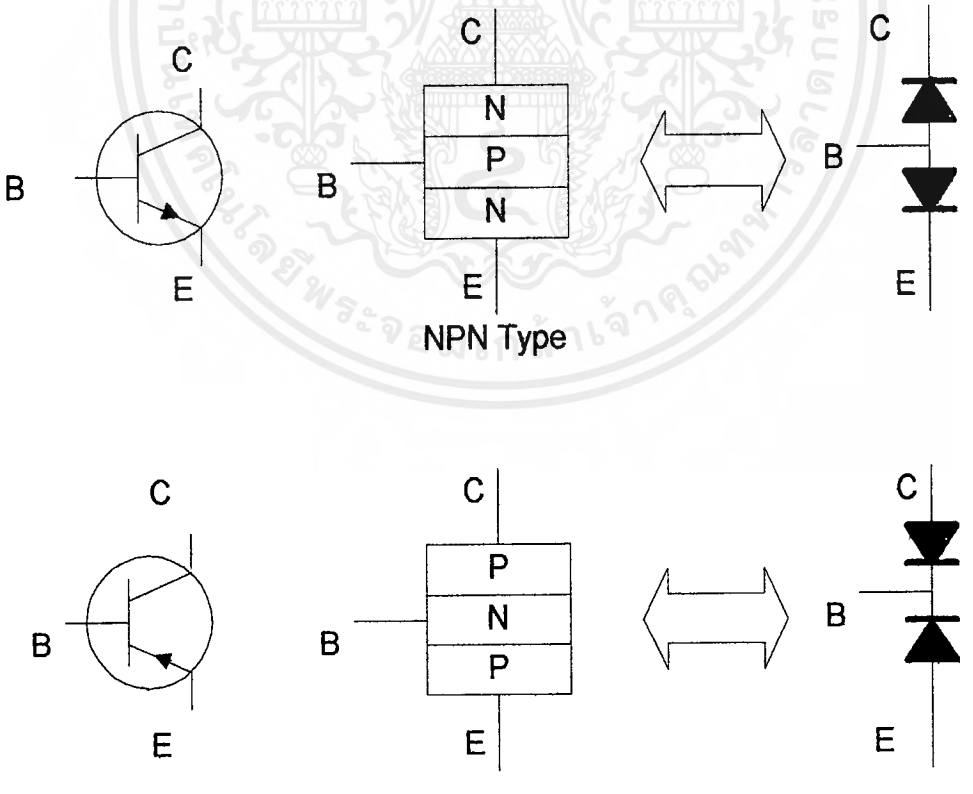
รูปที่ 4.2 กราฟแสดงความสัมพันธ์ระหว่างกระแสกับแรงดันของไดโอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวลูกศรเป็นสัญลักษณ์แทนสารกึ่งตัวนำชนิด P ซึ่งเป็นขั้วอาโนด(ขั้วบวก) ของไดโอด ลูกศรจะชี้ในทิศทางที่โฮลเคลื่อนที่ ส่วนขีดคั่นเป็นสสารกึ่งตัวนำชนิด N ซึ่งเป็นขั้วคาโถด(ขั้วลบ) ดังนั้นเราจะสามารถพิจารณาว่าไดโอดถูกไบอัสตรงหรือไบอัสกลับได้ โดยทั่วไปจะถือว่าขณะที่ไดโอดถูกไบอัสตรงเปรียบเสมือนว่าเกิดลัดวงจรตรงส่วนที่เป็นไดโอด ดังนั้นกระแสที่ไหลที่ผ่านไดโอดจึงมีค่าสูงและถือว่าแรงดันตกคร่อมไดโอดเป็นศูนย์ แต่ขณะที่ไดโอดถูกไบอัสกลับจะเปรียบเสมือนว่าเป็นวงจรเป็นจะไม่มีกระแสไหลผ่านเลย และแรงดันตกคร่อมไดโอดจะมีค่าเท่ากับแรงดันไบอัสกลับที่ป้อนให้ไดโอด ไดโอดที่มีคุณสมบัติเช่นนี้เราเรียกว่า ไดโอดอุดมคติ (ideal diode) แต่สำหรับไดโอดที่มีใช้อยู่จริงๆ ขณะที่ถูกไบอัสตรงจะมีกระแสไหลผ่านได้สูง และมีแรงดันตกคร่อมไดโอดเพียงเล็กน้อยอยู่ในช่วงประมาณ 0.5 - 1.5 โวลท์ ส่วนขณะถูกไบอัสกลับจะมีกระแสไหลผ่านน้อยมากไม่กี่ไมโครแอมป์

4.2 ทรานซิสเตอร์

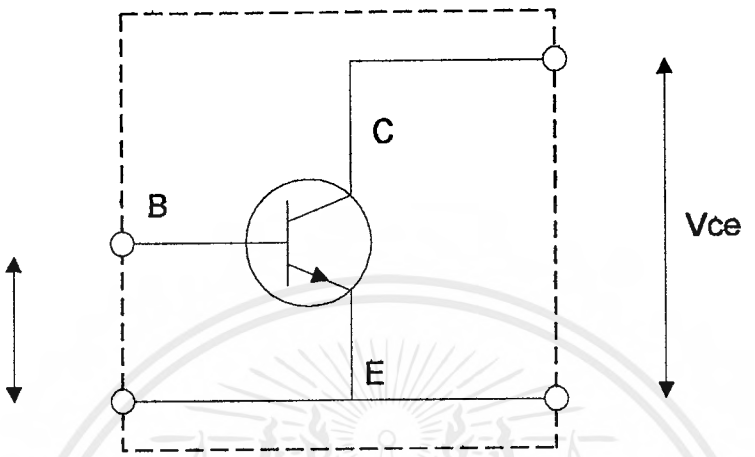
ซึ่งองค์ประกอบภายในของทรานซิสเตอร์จะเสมือนมีไดโอดสองตัวต่อกลับหัวกันอยู่ดังรูปที่ 4.3



รูปที่ 4.3 โครงสร้างพื้นฐานของทรานซิสเตอร์

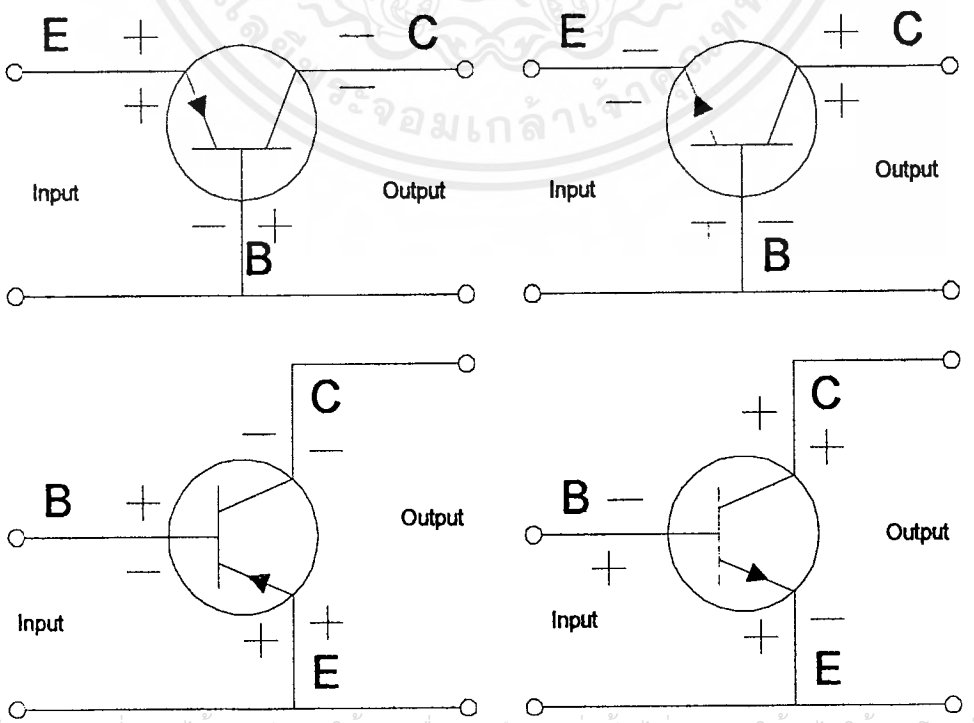
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยปรกติทรานซิสเตอร์จะมี 3 ขา ประกอบด้วยขาคอลเลคเตอร์ (Collector) ใช้สัญลักษณ์ C ขาเบส (Base) ใช้สัญลักษณ์ B และขา อิมิเตอร์ (Emitter)ใช้สัญลักษณ์ E แต่ในการอธิบายคุณสมบัติต่างๆ นั้นจะใช้วงจร 4 ขั้วในการอธิบายดังในรูปที่ 4.4

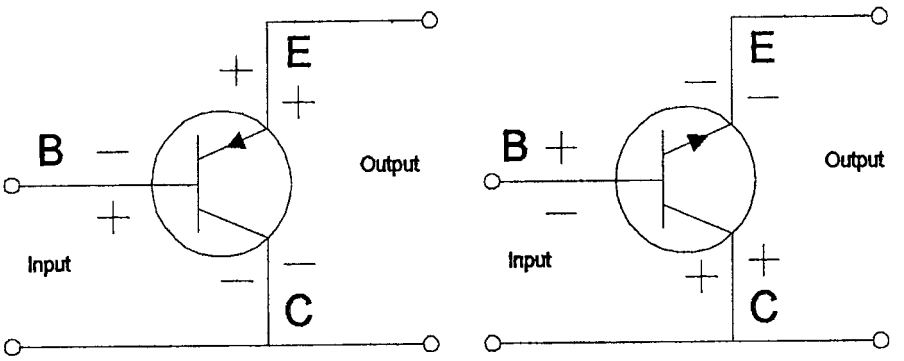


รูปที่ 4.4 เป็นวงจร 4 ขั้วมีขาอิมิเตอร์เป็นขาร่วมระหว่างอินพุทและเอาท์พุต

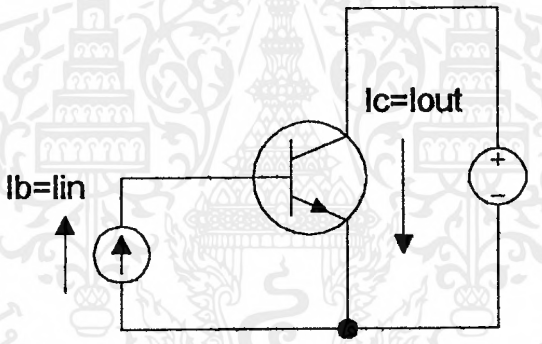
ลักษณะการจัดวงจรของทรานซิสเตอร์พื้นฐานจะจัดได้ 3 แบบ คือ วงจรเบสร่วม วงจรอิมิเตอร์ร่วม และวงจรคอลเลคเตอร์ร่วมและเพื่อที่จะทำให้ทรานซิสเตอร์ทำงานได้ในสภาวะแอกตีฟ (Active) แรงดันระหว่างขาต่างๆจะต้องมีขั้วตามรูปที่ 4.5



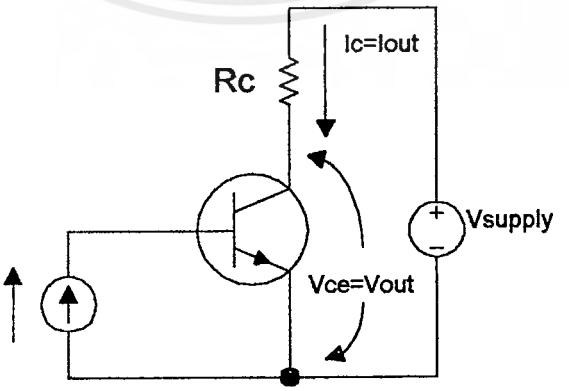
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 ลักษณะการจัดทรานซิสเตอร์พื้นฐาน 3 แบบ

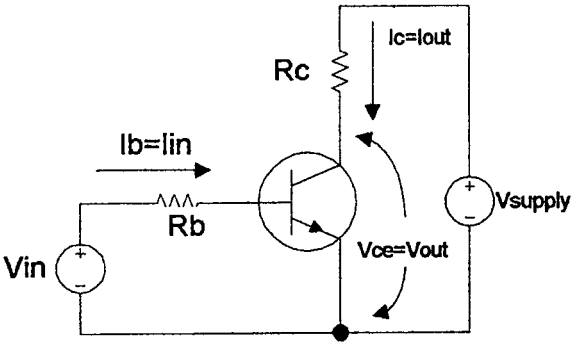


(ก) วงจรขยายกระแสเป็นกระแส



(ข) วงจรขยายกระแสเป็นแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ค) วงจรขยายแรงดันเป็นแรงดัน

รูปที่ 4.6 วงจรใช้งานทรานซิสเตอร์พื้นฐาน

ในรูปที่ 4.6 (ก) แสดงถึงกระแสคอลเลคเตอร์ที่มีค่ามากกว่ากระแสเบสตามค่าของ HFE ซึ่งแรงดันอินพุตที่ป้อนมาจะต้องมากกว่า 0.6 – 0.7 V (แรงดันตกคร่อมไดโอดเบส – อิมิตเตอร์) ทรานซิสเตอร์จึงจะเริ่มทำงาน กระแสคอลเลคเตอร์ที่เกิดขึ้นจะไม่ขึ้นแรงดันไฟเลี้ยวงจรแต่จะขึ้นอยู่กับกระแสเบสเพียงอย่างเดียว เราเรียกลักษณะนี้ว่าเป็นวงจรขยายกระแส

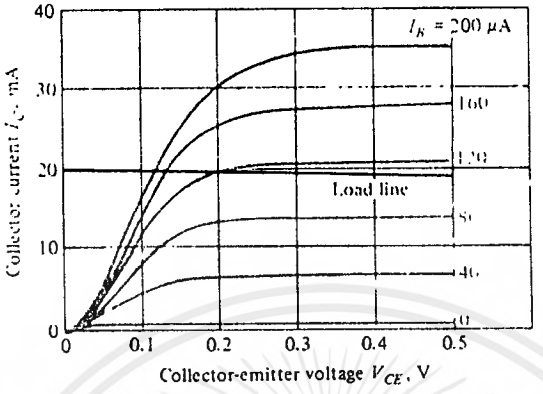
เมื่อพิจารณาดูแรงดันเอาท์พุตคอลเลคเตอร์ในรูปที่ 4.6 (ข) จะพบว่าแรงดันที่ได้จากแรงดันตกคร่อมของตัวต้านทานที่คอลเลคเตอร์เมื่อมีกระแสคอลเลคเตอร์ผ่านรูปแบบวงจรนี้เป็นลักษณะวงจรขยายแบบหนึ่ง ซึ่งเปลี่ยนจากกระแสอินพุตให้เป็นแรงดันทางเอาท์พุต เมื่อวัดแรงดันที่คอลเลคเตอร์เทียบกับอิมิตเตอร์ จะพบว่า เมื่อกระแสเบสเพิ่มขึ้น แรงดันที่คอลเลคเตอร์จะลดลงจึงเรียกการทำงานแบบนี้ว่าเป็นวงจรขยายแบบกลับเฟส (Inverting Amplifier)

ลักษณะการใช้ทรานซิสเตอร์อีกแบบหนึ่งคือ ใช้เป็นวงจรขยายในลักษณะที่เปลี่ยนแรงดันอินพุตเป็นแรงดันเอาท์พุตที่มากกว่าเดิม ดังรูปที่ 4.6 (ค) ปรกติจะไม่ป้อนแรงดันที่เกิน 0.7 V ให้กับขาเบสโดยตรงเพราะอาจทำให้เกิดความเสียหายขึ้นได้ ดังนั้นในการใช้งานส่วนใหญ่จะต้องเปลี่ยนจากแรงดันให้เป็นกระแสก่อน แล้วจึงป้อนให้แก่ทรานซิสเตอร์โดยการใช้ตัวต้านทานต่ออนุกรมกับขาเบส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 กราฟคุณสมบัติของทรานซิสเตอร์

ซึ่งสามารถที่จะแสดงได้ดังรูปที่ 4.7



รูปที่ 4.7 กราฟคุณสมบัติทางด้านเอาท์พุตของทรานซิสเตอร์เบอร์ BC107

จากกราฟคุณสมบัติของทรานซิสเตอร์เบอร์ BC107 เป็นค่าที่พล็อตขึ้นระหว่างกระแสคอลเลคเตอร์กับแรงดันคร่อมคอลเลคเตอร์กับอิมิเตอร์ โดยมีกระแสเบสเป็นค่าพารามิเตอร์กำหนด ในทางอุดมคติทรานซิสเตอร์จะต้องให้เส้นของกระแสคอลเลคเตอร์ที่ขนานไปกับแกนนอนเมื่อ Vbe มีค่ามากกว่า 0.7V ซึ่งแสดงว่ากระแสคอลเลคเตอร์จะคงที่เมื่อไบอัสด้วยกระแสเบสค่าหนึ่งและจะเปลี่ยนเมื่อกระแสเปลี่ยนไปเท่านั้น ไม่เปลี่ยนไปตามแรงดันตกคร่อมคอลเลคเตอร์กับอิมิตเตอร์

4.3 ทรานซิสเตอร์สนามไฟฟ้า หรือเฟท

ทรานซิสเตอร์สนามไฟฟ้า คือ ทรานซิสเตอร์ที่ใช้สนามไฟฟ้าในการเปลี่ยนแปลงสภาพของสารกึ่งตัวนำเพื่อให้เกิดการนำกระแสเมื่อได้รับแรงดันไฟฟ้าที่เหมาะสม โดยทั่วไปมักเรียกชื่อว่า " เฟท "

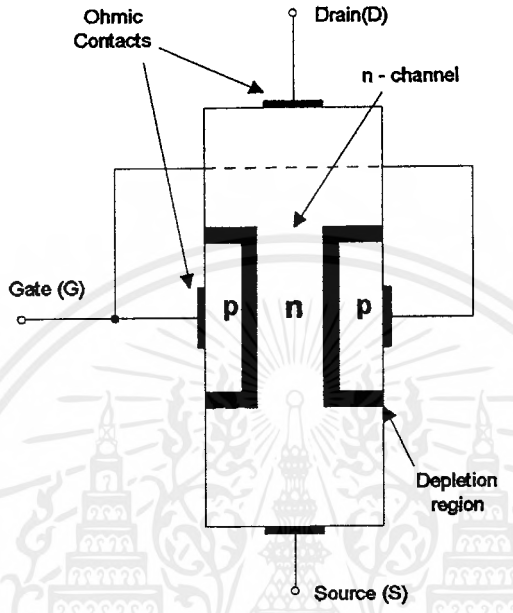
เฟทแบ่งออกเป็น 2กลุ่ม คือ ทรานซิสเตอร์สนามไฟฟ้าแบบรอยต่อ (Junction Field Effect Transistor)หรือ เจเฟท (JFET) และทรานซิสเตอร์สนามไฟฟ้าแบบโลหะ- ออกไซด์- สารกึ่งตัวนำ (Metal-Oxide-Semiconductor Field Effect Transistor) หรือ มอสเฟท (MOSFET)

4.3.1 โครงสร้างและคุณลักษณะของเจเฟท

เจเฟทเป็นอุปกรณ์สารกึ่งตัวนำที่มี 3 ขั้ว คือ ขั้วซอร์ส (Source: S) ขั้วเดรน (Drain: D) และ ขั้วเกท (Gate: G) เจเฟทแบ่งออกเป็น 2ประเภท คือแบบแชนแนล N (N- channel) และแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แชลแนลP (P - Channel) การทำงานของเจเฟทคล้ายกับการทำงานของก๊อกน้ำ กล่าวคือ เราสามารถควบคุมการไหลของน้ำซึ่งเปรียบเสมือนการไหลของอิเล็กตรอน จากขั้ว S มายังขั้ว D ด้วยการควบคุมแรงดันไฟฟ้าที่ขั้ว G ดังรูปที่ 4.8



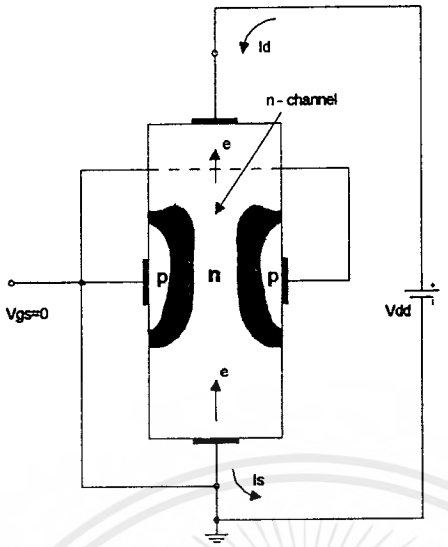
รูปที่ 4.8 แสดงโครงสร้างของเฟท

จากรูปที่ 4.8 เป็นโครงสร้างของเจเฟทแบบ N- Channel ขั้ว S และขั้ว D ต่ออยู่กับสารกึ่งตัวนำชนิด N โดยผ่านหน้าสัมผัสโอห์มมิค ส่วนขั้ว G ต่ออยู่กับสารกึ่งตัวนำชนิด P หลักการทำงานของเจเฟทประเภทนี้แบ่งการอธิบายออกเป็น 2 กรณีคือ ขณะแรงดันระหว่างขั้ว G และขั้ว S (V_{GS}) มีค่าเท่ากับศูนย์และขณะ V_{GS} มีค่าเป็นลบ

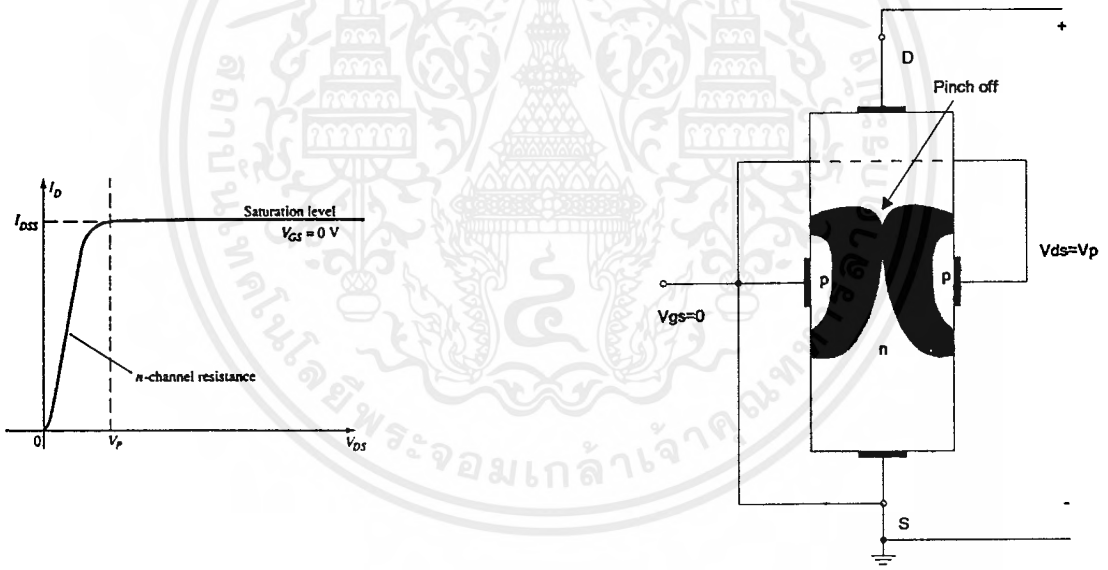
4.3.2 ขณะ V_{GS} มีค่าเท่ากับศูนย์

จากรูปที่ 4.9 เมื่อจ่ายแรงดัน V_{DD} จากแหล่งจ่ายให้มีค่าเท่ากับ V_{DS} โดยกำหนด V_{GS} =0 อิเล็กตรอนจะถูกดึงดูดออกมาทางขั้ว D ทำให้กระแสที่ขั้วเดรนหรือ กระแสเดรน (I_D)ไหลผ่านแชนแนล n กลายเป็นกระแสของขั้วซอสหรือ กระแสซอส (I_S) ดังนั้นขนาดของ I_D จึงเท่ากับ I_S และมีความต้านทานของแชนแนล n เป็นตัวจำกัดกระแสดังกล่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 เมื่อป้อนแรงดัน V_{DD} ที่ $V_{GS} = 0$



(ก) (ข)

รูปที่ 4.10 (ก) แสดงกราฟคุณสมบัติของเฟท

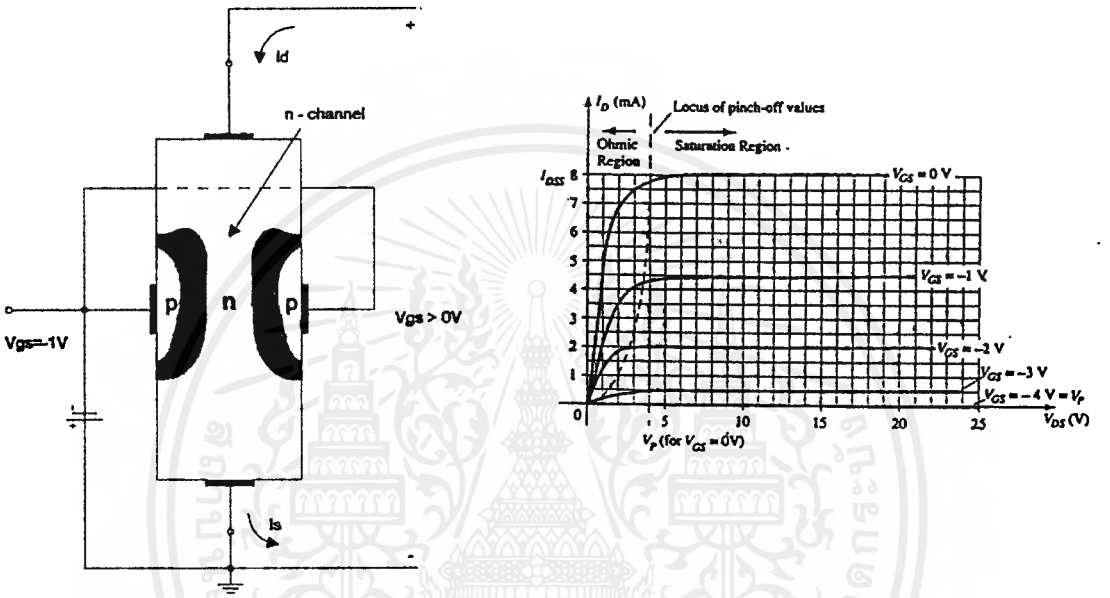
(ข) แสดงโครงสร้างของเฟทขณะเกิดจุดพินช์ออฟ (Pinch-off)

จากรูปที่ 4.10(ก) แสดงความสัมพันธ์ระหว่าง I_Dกับ V_{DS} คือ เมื่อ V_{DS} มีค่าเพิ่มขึ้นเพียงเล็กน้อย ($V_{DS} < V_P$) I_D ก็จะเพิ่มขึ้นในลักษณะเป็นเชิงเส้น(ตามกฎของโอห์ม) ขณะ V_{DS} มีค่าต่ำ ความต้านทานของเจเฟทจะมีค่าคงที่ (เมื่อ V_{DS} เพิ่มขึ้น บริเวณปลอดพาหะจะขยายตัวมากขึ้น ทำให้ความกว้างของแชนแนลลดลง ความต้านทานจึงเพิ่มขึ้น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขณะV_{DS} เพิ่มขึ้น ดังรูปที่ 4.10 (ข) จนทำให้บริเวณปลอดพาหะทั้ง 2 บริเวณใมผัสกัน ซึ่งเราเรียกว่า จุดพิสช์ออฟ (Pinch-off) ค่า V_{DS} ที่จุดนี้เราเรียกว่า แรงดันพินช์ออฟ (Pich-off Voltage : V_P) I_D จะเกิดการอิ่มตัว (I_{DSS})หรือสภาพที่เจเฟทมี I_D สูงสุดของเจเฟท ค่า I_{DSS} นี้เกิดขึ้นได้เมื่อ V_{GS} = 0V และ V_{DS} >|V_P|

4.3.3 ขณะ V_{GS} มีค่าเป็นลบ



(ก) (ข)

รูปที่ 4.11 (ก) แสดงโครงสร้างของเฟท เมื่อ V_{GS}เป็นลบ

(ข) แสดงกราฟความสัมพันธ์ระหว่างกระแสกับแรงดัน

V_{GS} ซึ่งเป็นแรงดันที่ใช้ควบคุมเจเฟทมีความสัมพันธ์กับ I_D และ V_{DS} เช่นเดียวกับ I_B มีความสัมพันธ์กับ I_C และ V_{CE}

สำหรับเจเฟทแบบแชนแนล n แรงดัน V_{GS}ที่ใช้ควบคุมมีค่าเป็นลบ จากรูปที่ 4.11(ก) เมื่อจ่าย V_{GS} =-1V จะทำให้ขั้ว G ได้รับการไบอัสกลับ บริเวณปลอดพาหะจึงขยายตัว ทำให้ระดับการอิ่มตัวของ I_D(I_{DESS}) ลดลง ส่งผลให้ V_{DS} ที่ระดับอิ่มตัวมีค่าลดลงด้วย(ตามกฎของโอห์ฒ) ดังรูปที่ 4.11(ข) และลดลงเรื่อยๆ เมื่อ V_{GS} เป็นลบมากขึ้นตามลำดับ ดังนั้นค่า V_P จึงลดลงตามเส้นทางโลกัสของค่าพินช์ออฟ (Locus of Pinch-off Value) ซึ่งมีลักษณะเป็นพาราโบลา ถ้า V_{GS} มีค่าลบเพิ่มขึ้นเรื่อยๆ จนกระทั่ง V_{GS} =V_P จะได้ I_P = 0 mA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

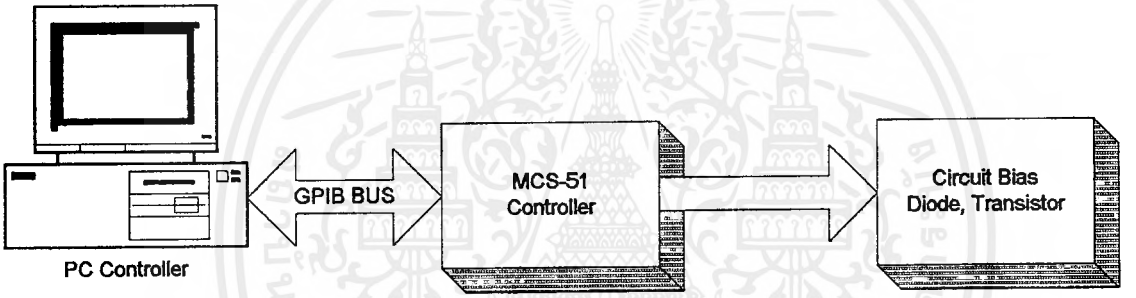
บริเวณด้านขวามือของโลกัสของค่าพินช์ออฟคือบริเวณอิ่มตัว (Saturation Region ถ้านำไปใช้งานในวงจรขยายจะได้สัญญาณออกมาเป็นเชิงเส้น (วงจรขยายจ่ายสัญญาณเอาท์พุตไม่เพี้ยน) ต่างกับบริเวณอิ่มตัวของทรานซิสเตอร์ BJT ซึ่งไม่สามารถนำไปใช้ในการขยายสัญญาณได้ เพราะจะทำให้สัญญาณเอาท์พุตเพี้ยน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การออกแบบและการเขียนโปรแกรมควบคุม

ในการเขียนโปรแกรมควบคุมเครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์นั้นจะแบ่งออกเป็น 2 ส่วนคือ การเขียนโปรแกรม Delphi เพื่อควบคุมอุปกรณ์ GPIB ซึ่งได้แก่ Power Supply และ Multimeter และอีกส่วนหนึ่งจะเป็นการเขียนโปรแกรมไมโครคอนโทรลเลอร์ (Mcs-51) เพื่อทำการรับค่าและส่งค่ากับตัวควบคุม (Controller) ซึ่งจะสามารถอธิบายในแต่ละส่วนดังนี้

5.1 การเขียนโปรแกรมควบคุมในส่วนของไมโครคอนโทรลเลอร์



รูปที่ 5.1 แสดง Block Diagram ในส่วนของไมโครคอนโทรลเลอร์ติดต่อกับคอมพิวเตอร์

จากรูปที่ 5.1 จะแสดงให้เห็นว่า ในส่วนของไมโครคอนโทรลเลอร์ นั้นจะเปรียบเสมือนเป็นตัวรับ (Listener) และตัวส่ง (Talker) ตัวหนึ่งของระบบโดยจะมีตำแหน่งแอดเดรส ตามที่ตั้งค่าไว้ ซึ่งจะคอยรับคำสั่งที่สั่งมาจากตัวควบคุม คำสั่งที่ส่งมาจะเป็นคำสั่งมาตราฐานของระบบ GPIB ที่ใช้กับการ์ดของบริษัทเนชั่นแนลอินทรูเมนท์

ในการเขียนโปรแกรมควบคุมอุปกรณ์บนระบบบัสมาตรฐาน IEEE-488 (GPIB) จะมีขั้นตอนในการติดต่อระหว่างอุปกรณ์ทั้งหมดบนระบบบัสโดยเฉพาะ โดยการอาศัยสัญญาณ ATN (Attention) ซึ่งเป็นสัญญาณที่ใช้ในการควบคุม ขั้นตอนดังกล่าวจะแบ่งออกได้เป็น 2 ช่วง คือ

1. ช่วงสัญญาณ ATN มีลอจิกเป็น Low

ข้อมูลที่ถูกส่งลงบนบัสในช่วงที่สัญญาณ ATN มีลอจิกเป็น Low จะหมายถึงคำสั่งที่เป็นรหัสคำสั่งมาตรฐานของ IEEE –488 ตามตารางที่ 2.1 โดยรหัสคำสั่งในคอลัมน์ที่ 0 และ 1 จะเป็นคำสั่งที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UNL (3Fh) (Unlistenner) ตัวควบคุมจะทำการเคลียร์อุปกรณ์ทุกตัวออกจากการเป็นตัวรับในระบบ

MTA (55h) (My Talker Address) ตัวควบคุมจะแจ้งแอดเดรสของตัวเองในการติดต่อกับอุปกรณ์ที่เป็นตัวรับ โดยตัวควบคุมจะทำหน้าที่เป็นตัวส่งที่มีแอดเดรส 21 (21+40h=55h)

OLA (Our Listener Address) จัวควบคุมทำการติดต่อกับอุปกรณ์ที่ทำหน้าที่เป็นตัวรับที่มีแอดเดรสเป็น 1 (1+20h=21h) ตัวรับที่มีแอดเดรสดังกล่าวจะรับรู้ว่าตัวควบคุมต้องการติดต่อ เพื่อให้ทำการรับข้อมูลที่ส่งลงมาบนบัส

จากรูปที่ 5.2 จะแสดงถึงโฟลว์ชาร์ท ที่เขียนโปรแกรมควบคุมในส่วนของไมโครคอนโทรลเลอร์ ซึ่งในการติดต่อในมาตรฐาน IEEE488.2 นั้นจะต้องทำการ แฮนด์เช็ค(Handshake) กันทุกครั้ง ถ้าเกิดมีการผิดพลาดเกิดขึ้นระบบจะยกเลิกการแฮนด์เช็ค นั้น ในการส่งข้อมูลจากตัวควบคุม นั้นจะส่งมาเป็นรูปแบบคำสั่งของ GPIB ซึ่งคำสั่งที่ใช้ในการควบคุมจะมีดังนี้

ibclr ➔ เป็นคำสั่งที่ใช้ในการควบคุมในอุปกรณ์ที่ต่อในระบบอยู่ในสภาวะ Remote

ibloc➔ เป็นคำสั่งที่ทำให้อุปกรณ์กลับมาอยู่ในสภาวะเริ่มต้น

ibwrt "SG" ➔ เป็นคำสั่งที่ใช้วัดคุณสมบัติของDiode ซึ่งจะทำให้วงจรไบอัสจัดวงจรสำหรับไดโอดไว้

ibwrt "STN" ➔ เป็นคำสั่งที่ใช้ในการวัดทรานซิสเตอร์ชนิด N

ibwrt "STP" ➔ เป็นคำสั่งที่ใช้ในการวัดทรานซิสเตอรชนิด P

ibwrt "SFN" ➔ เป็นคำสั่งที่ใช้ในการวัดFET ชนิด N

ibwrt "SFP" ➔ เป็นคำสั่งที่ใช้ในการวัดFET ชนิด P

ibwrt "STEST" ➔ เป็นคำสั่งที่ใช้ในการวัดหาตำแหน่งขาของอุปกรณ์ที่ต้องการทราบตำแหน่งขาสำหรับทรานซิสเตอร์

ibwrt "ST:1" ➔ เป็นคำสั่งที่ใช้ในการเลือกย่านกระแส IB มีค่าเท่ากับ 10 uA

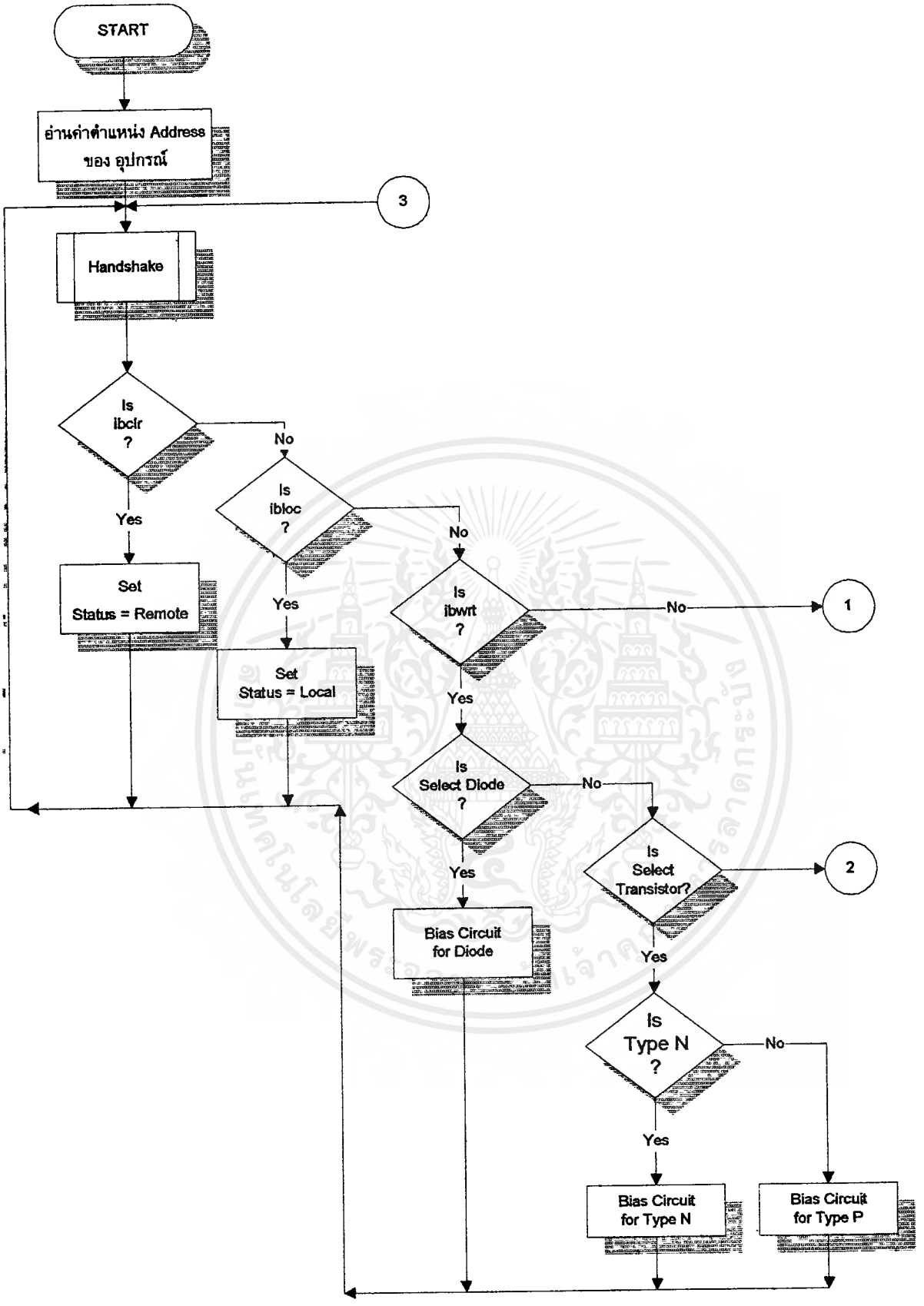
ibwrt "ST:2" ➔ เป็นคำสั่งที่ใช้ในการเลือกย่านกระแส IB มีค่าเท่ากับ 100 uA

ibwrt "ST:3" ➔ เป็นคำสั่งที่ใช้ในการเลือกย่านกระแส IB มีค่าเท่ากับ 1 mA

ibwrt "ST:4" ➔ เป็นคำสั่งที่ใช้ในการเลือกย่านกระแส IB มีค่าเท่ากับ 10 mA

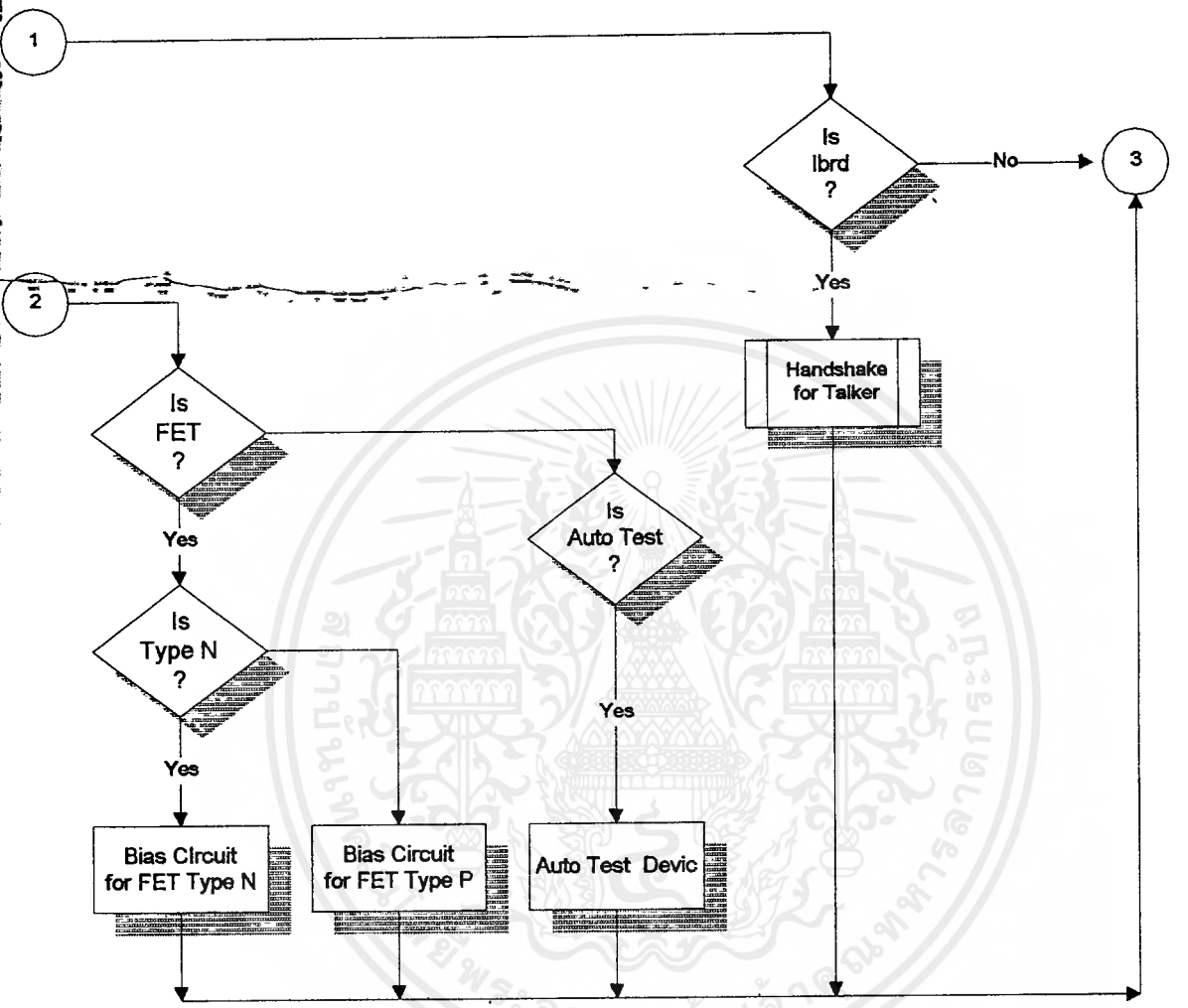
ibrd ➔ เป็นคำสั่งที่ใช้ในการอ่านค่าจากเครื่องตรวจสอบฯ ซึ่งจะส่งข้อมูล 1 byte กลับไปยัง Computer เพื่อบอกขาของอุปกรณ์ที่วัด (ในกรณีของทรานซิสเตอร์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



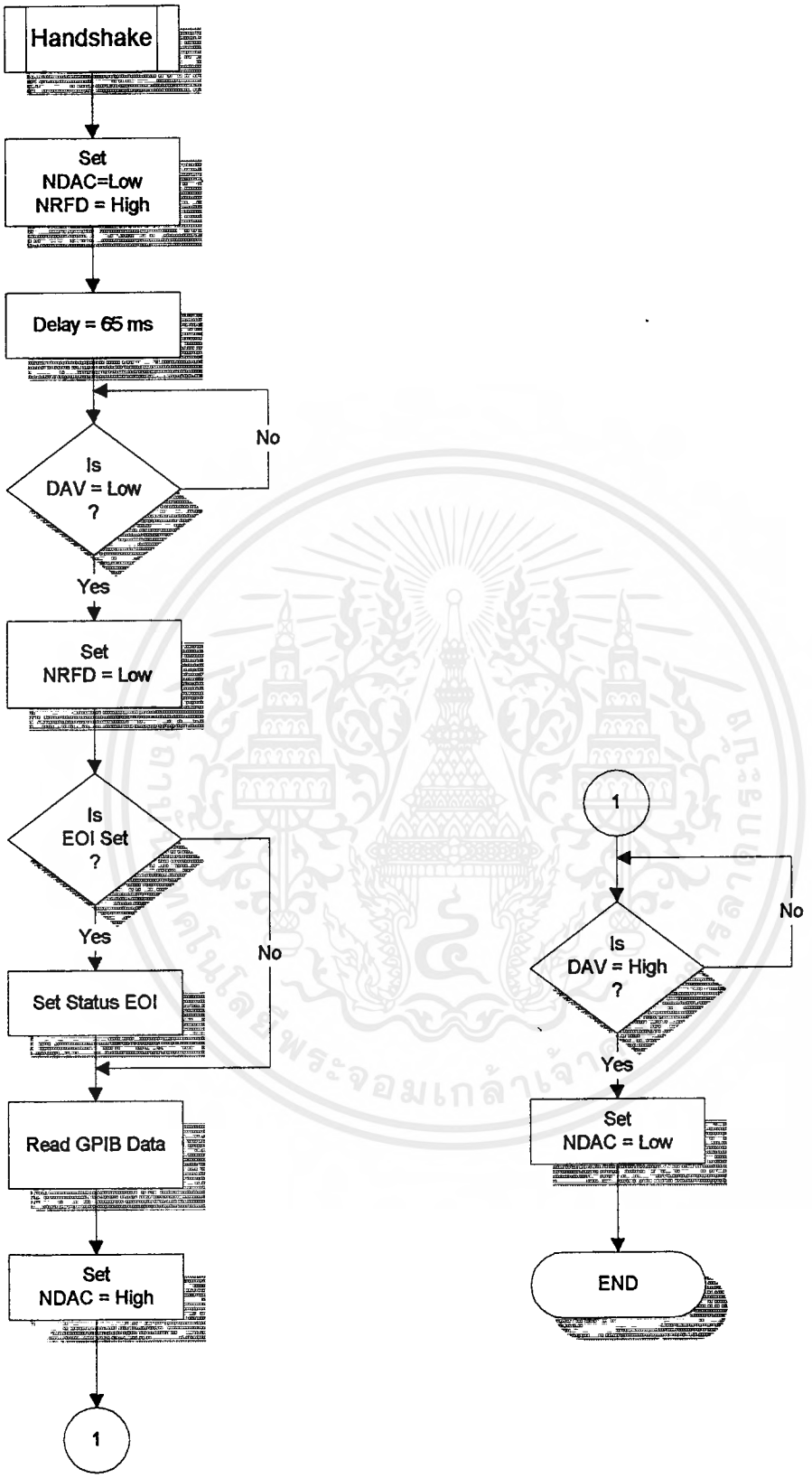
รูปที่ 5.4 แสดง Flowchart ในการรับส่งข้อมูลกับ Controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



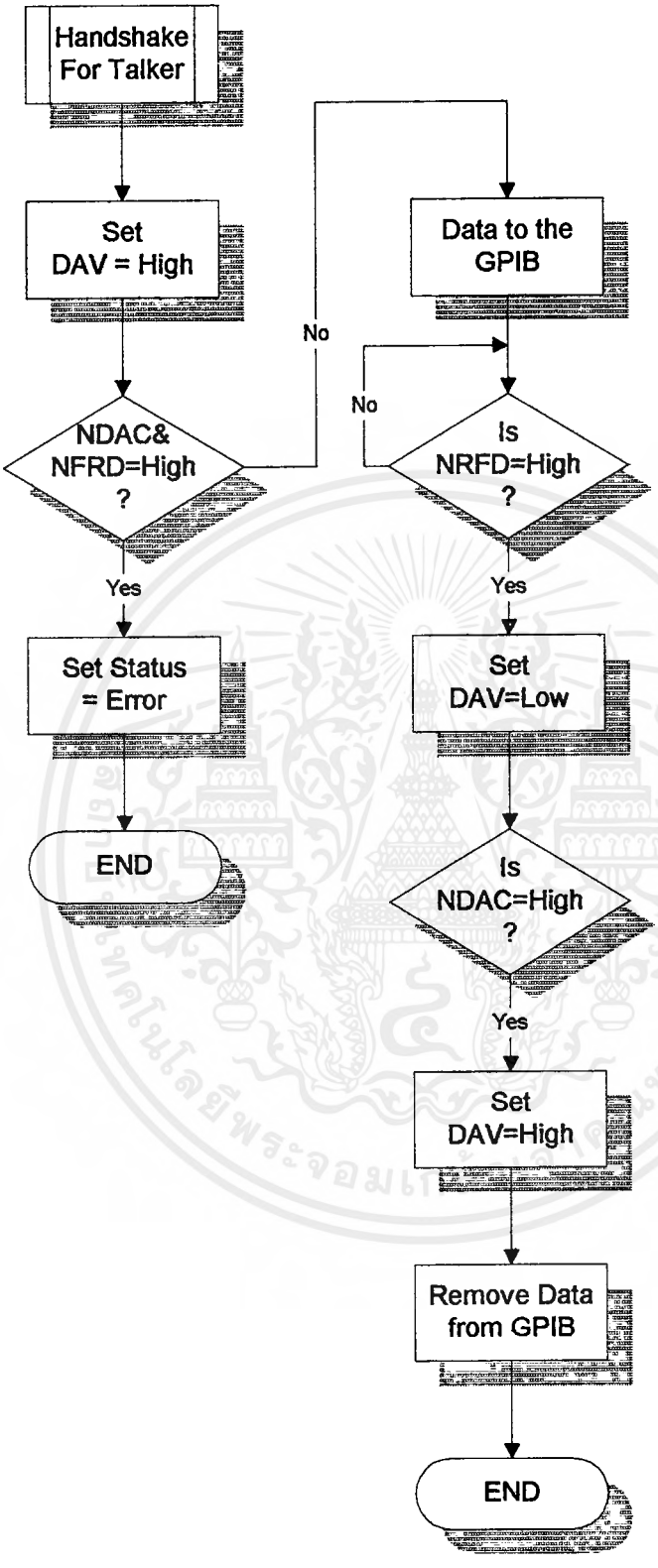
รูปที่ 5.4 (ต่อ) แสดง Flowchart ในการรับส่งข้อมูลกับ Controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.5 แสดงโฟว์ชาร์ทในการแฮนด์เช็คในการรับข้อมูลจากตัวควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 แสดงโฟว์ชาร์ทในการแฮนด์เช็คในการส่งข้อมูลให้กับตัวควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างในการเขียนโปรแกรมในส่วนของการแฮนด์เช็คในการรับข้อมูล

```
HAND_CHECK:   PUSH DPH
              PUSH DPL
              PUSH R7
              PUSH R2
              MOV DPTR,#PORTC
              MOV A,#08H
              MOVX @DPTR,A    ; set NRFD=High , NDAC=Low

              MOV R7,#30h
DAV0:         MOV R2,#00h
DAV:          DJNZ R2,DAV_1
              DJNZ R7,DAV0
              MOV R5,#00H
              POP R2
              POP R7
              POP DPL
              POP DPH
              RET

DAV_1:        MOVX A,@DPTR
              ANL A,#40H
              CJNE A,#00H,DAV  ; DAV=Low ?
              MOV DPTR,#PORTCL
              MOV A,#06H
              MOVX @DPTR,A    ; SET NRFD=LOW
              MOV DPTR,#PORTCL
              MOV A,#05H
              MOVX @DPTR,A ; SET NDAC=HIGH
              MOV DPTR,#PORTA
              MOVX A,@DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CPL A
        MOV R5,A
        MOV DPTR,#PORTC
DAV1:   MOVX A,@DPTR
        ANL A,#40H
        CJNE A,#40H,DAV1  ; check DAV=Hi?
        MOV DPTR,#PORTCL
        MOV A,#04H
        MOVX @DPTR,A    ; set NDAV=Low

        mov a,#07h
        movx @dptr,a
        POP R2
        POP R7
        POP DPL
        POP DPH
        RET
```

ตัวอย่างการเขียนโปรแกรมในส่วนของการแฮนด์เช็คในการส่งข้อมูล

```
CHECK_SEND:     MOV A,#83H
                MOV DPTR,#PORTCL
                MOVX @DPTR,A
                MOV DPTR,#PORTCL
                MOV A,#0DH
                MOVX @DPTR,A    ; SET DAV=HIGH

cc:             MOVX A,@DPTR        ; CHECK NRFD , NDAC
                ANL A,#0CH
                CJNE A,#0CH,SEND
                lcall delay
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ljmp cc

SEND:   MOV DPTR,#PORTA
        MOV A,R5
        CPL A
        MOVX @DPTR,A

        MOV DPTR,#PORTC
CHECK_NRFD:     MOVX A,@DPTR
        ANL A,#08H
        CJNE A,#08H,CHECK_NRFD   ; check NRFD=High ?
        MOV DPTR,#PORTCL
        MOV A,#0CH
        MOVX @DPTR,A        ; SET DAV= LOW

        MOV R7,#30H
NDAC1:  MOV R2,#00H
NDAC2:  DJNZ R2,CHECK_NDAC
        DJNZ R7,NDAC1
        RET
CHECK_NDAC:     MOV DPTR,#PORTC
        MOVX A,@DPTR
        ANL A,#04H
        CJNE A,#04H,CHECK_NDAC   ; check NDAC=High?
        MOV DPTR,#PORTCL
        MOV A,#0DH
        MOVX @DPTR,A        ; SET DAC=High
        RET
```

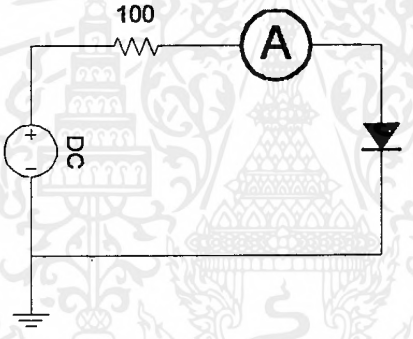
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ส่วนควบคุมการวัดอุปกรณ์

เป็นส่วนที่เขียนโปรแกรมโดยใช้ Delphi เขียนควบคุมเครื่องมือวัดแต่ละตัว ในการควบคุมก็จะควบคุมผ่าน GPIB เครื่องมือวัดที่ใช้งานในการวัดได้แก่ มัลติมิเตอร์ และ แหล่งจ่ายไฟ ซึ่งเราจะเขียนโปรแกรมให้ควบคุมเครื่องมือวัดเหล่านี้ให้ทำการทดสอบอุปกรณ์ได้ซึ่งอุปกรณ์ที่สามารถวัดได้คือ ไดโอด ทรานซิสเตอร์ และ เฟท ซึ่งจะอธิบายหลักการในการวัดดังนี้

5.2.1 การวัดไดโอด

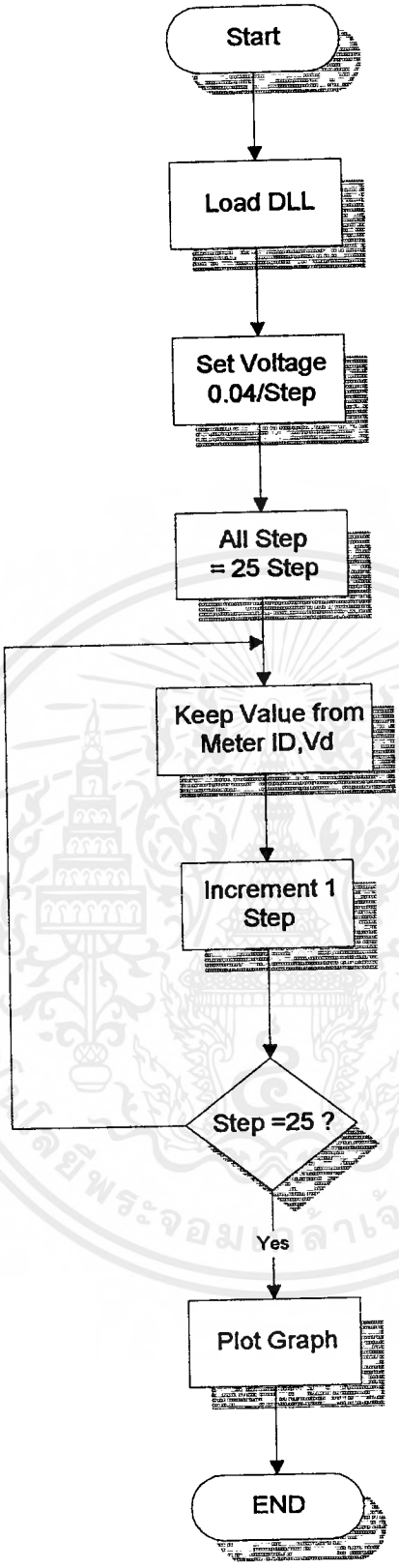
ในการวัดคุณสมบัติของไดโอดนั้น จะเป็นการวัดของ กระแสและแรงดันที่ผ่านไดโอดแล้วนำมาแสดงกราฟ โดยแกนตั้งเป็นค่าของกระแส และแกนนอนเป็นค่าของแรงดันตกคร่อมไดโอด ซึ่งตรงนี้เป็นหลักการเบื้องต้นของการวัดคุณสมบัติของไดโอด โดยจะใช้วงจรดังรูปที่ 5.7 ซึ่งเป็นวงจรพื้นฐานในการทดสอบ แหล่งจ่ายไฟนั้นสามารถที่จะปรับแหล่งดันค่าต่างๆได้ โดยควบคุมจากคอมพิวเตอร์ และ มิเตอร์จะอ่านค่ากระแสที่ได้แล้ส่งค่ากลับมายังคอมพิวเตอร์เพื่อแสดงกราฟ



รูปที่ 5.7 แสดงวงจรที่ใช้ทดสอบคุณสมบัติของไดโอด

จากรูปที่ 5.8 เป็นโฟลว์ชาร์ท แสดงการวัดคุณสมบัติของไดโอด ซึ่งจะทำการเปลี่ยนค่าของแหล่งจ่ายไฟตรงไปที่ค่าต่างๆ แล้วทำการวัดค่ากระแสที่ไหลผ่านไดโอด แล้วนำมาคำนวณหาค่าแรงดันตกคร่อมไดโอด แล้วนำมาแสดงเส้นกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

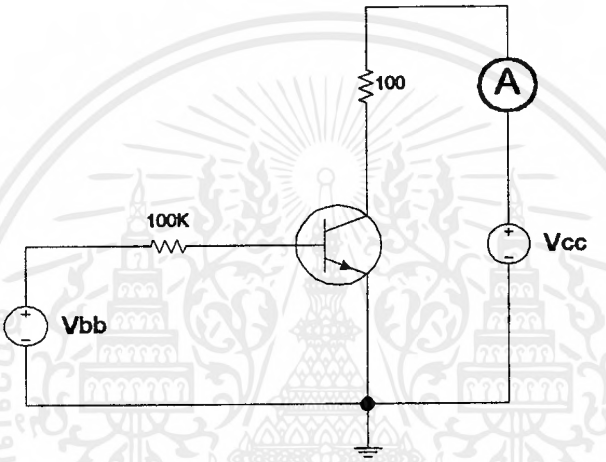


รูปที่ 5.8 แสดง Flowchart ของการวัดคุณสมบัติของไดโอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2 การวัดทรานซิสเตอร์

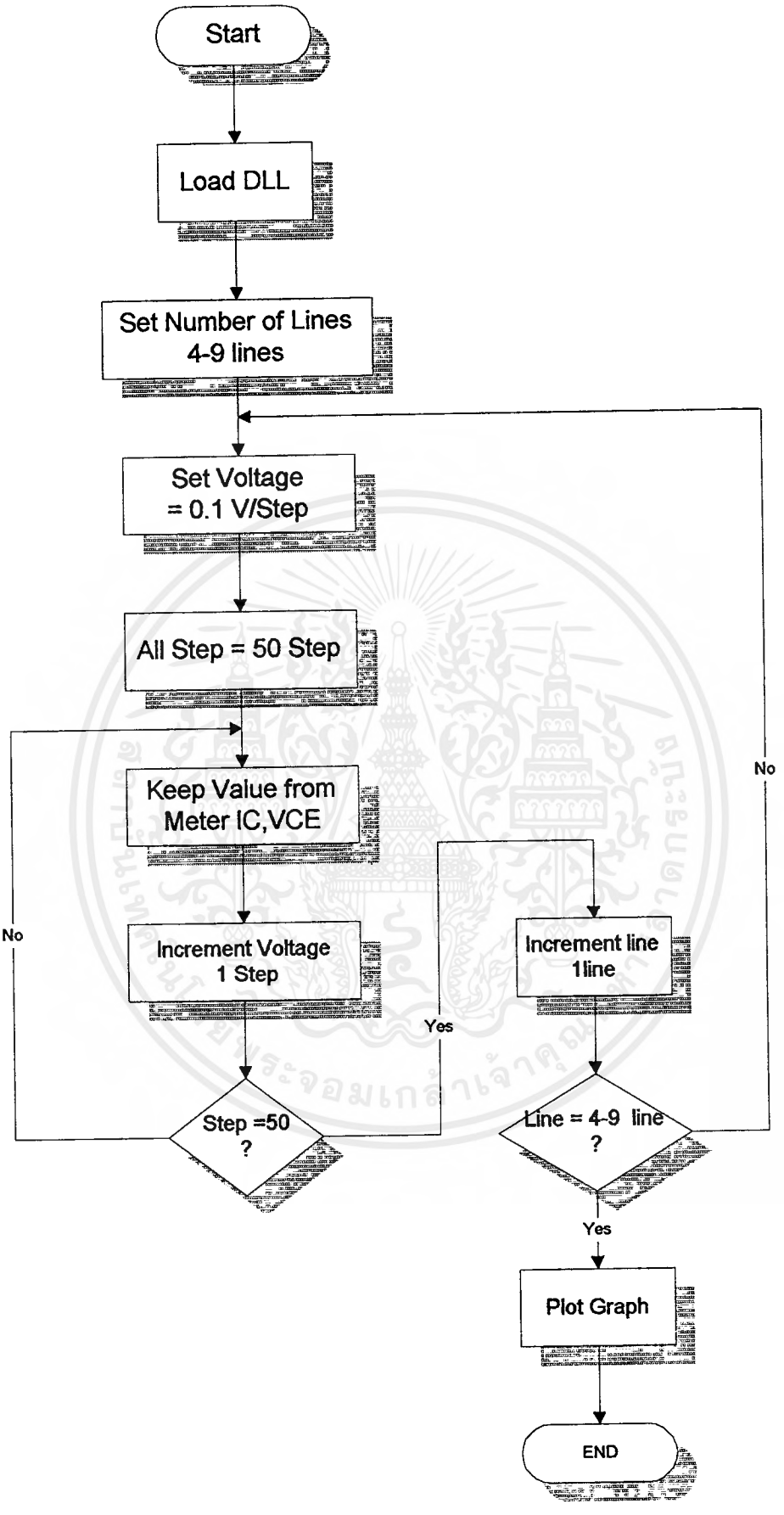
ในการวัดทรานซิสเตอร์นั้นจะใช้แหล่งจ่ายไฟอยู่สองแหล่ง คือ ส่วนที่ทำการจ่ายกระแสเบสให้กับทรานซิสเตอร์ กับส่วนของแหล่งจ่ายไฟที่ป้อนให้กับขาคอลเลคเตอร์ ซึ่งกระแสคอลเลคเตอร์ที่จะไหลผ่านทรานซิสเตอร์นั้นจะขึ้นอยู่กับค่ากระแสเบส โดยมีตัวแปรคือค่าอัตราการขยายกระแสอีกตัวหนึ่ง ซึ่งค่าอัตราการขยายกระแสนั้นไม่ได้เป็นค่าคงที่ๆค่าหนึ่งแต่ค่าอัตราการขยายกระแสนั้น จะเปลี่ยนตามค่ากระแสคอลเลคเตอร์ วงจรที่ใช้ทำการวัดคุณสมบัติของทรานซิสเตอร์นั้นแสดงได้ดังรูปที่ 5.9



รูปที่ 5.9 แสดงวงจรที่ใช้ในการวัดคุณสมบัติทรานซิสเตอร์

ซึ่งหลักการในการวัดคุณสมบัติทรานซิสเตอร์นั้น เราจะทำการเปลี่ยนค่ากระแสเบสเป็นค่าต่างๆ โดยการปรับเปลี่ยนค่าแรงดัน Vbb ซึ่งจะต้องให้ค่ากระแสเบสคงที่ค่าๆหนึ่ง แล้วทำการปรับค่าแรงดันที่ Vcc ตั้งแต่ 0- 20 โวลท์ ซึ่งจะได้ค่ากระแสของกระแสคอลเลคเตอร์ จากนั้นนำค่ากระแสที่วัดได้มาคำนวณหาค่าแรงดันตกคร่อมระหว่างคอลเลคเตอร์กับอิมิตเตอร์ เพื่อนำมาแสดงเส้นกราฟ ซึ่งจุดนี้จะได้เส้นกราฟเพียงเส้นเดียว ในกรณีที่ต้องการเส้นกราฟหลายเส้นก็ต้องป้อนค่ากระแสเบส หลายๆค่า เส้นกราฟที่ได้ก็จะมีหลายเส้น ซึ่งจะเป็นกราฟที่สามารถบอกคุณสมบัติของทรานซิสเตอร์ได้ หลักการในการวัดคุณสมบัติแสดงได้ดังโฟลว์ชาร์ท ดังรูปที่ 5.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

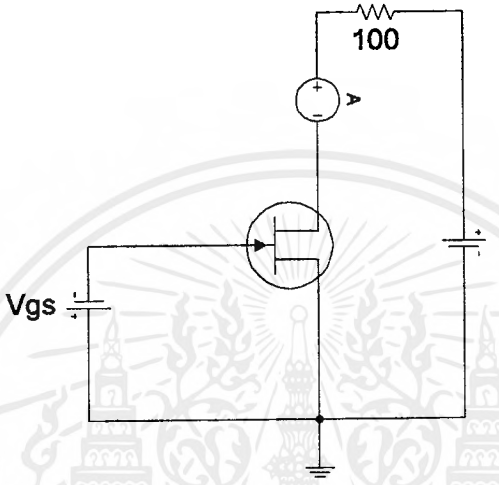


รูปที่ 5.10 แสดง Flowchart ของการวัดคุณสมบัติของทรานซิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.3 การวัดเฟท

ในขั้นตอนของการวัดเฟทนั้นจะมีขั้นตอนในการวัดเหมือนกับทรานซิสเตอร์ทุกประการ เพียงแต่ในกรณีของการวัดเฟทนั้น จะใช้แรงดันทางอินพุทเป็นตัวควบคุมการไหลของกระแสเดรน ซึ่งจะทำการเปลี่ยนค่าแรงดันเกท-ซอส แล้วทำการเพิ่มค่าแรงดันเดรน-ซอส แล้วทำการอ่านค่ากระแสที่ได้ นำมาพล็อตกราฟแสดงคุณสมบัติของเฟท ซึ่งวงจรที่ใช้ในการวัดแสดงดังรูปที่ 5.11



รูปที่ 5.11 แสดงวงจรที่ใช้วัดคุณสมบัติของเฟท

จากรูปที่ 5.11 แรงดันที่ใช้ป้อนที่ขาเกทของเฟทจะเป็นลบสำหรับในกรณีของชนิด N ส่วนในกรณีชนิด P แรงดันที่ขาเกทจะเป็นบวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 การออกแบบวงจรในส่วนต่างๆ

ในการออกแบบวงจรในส่วนต่างๆ นั้น สามารถที่จะแบ่งได้เป็น 2ส่วน คือในส่วนของไมโครคอนโทรลเลอร์ และส่วนของการไบอัสอุปกรณ์ที่ต้องการวัด ซึ่งจะอธิบายในส่วนต่างๆ ได้ดังนี้

5.3.1วงจรในส่วนของไมโครคอนโทรลเลอร์

ในส่วนของวงจรไมโครคอนโทรลเลอร์ นั้นจะใช้เบอร์ 8031 ซึ่งเป็นตระกูลของ MCS-51วงจรที่ใช้งานแสดงดังรูปที่ 5.12 สามารถอธิบายหน้าที่ของอุปกรณ์แต่ละตัวได้ดังนี้

U1 8031 เป็นตัวควบคุมการทำงานทั้งหมด

U2 74S373 เป็นตัวแลทช์ข้อมูล

U3 74S138 ทำหน้าทีถอดรหัสในการติดต่ออุปกรณ์แต่ละตัว

U4 2764 เป็น EPROM ที่เก็บโปรแกรม

U5 8255 เป็นตัวควบคุมการรับส่งข้อมูลผ่านบัส IEEE488

U6 74S02 ทำหน้าที่ควบคุมการแสดงผลของLCD

U7 8255 เป็นตัวควบคุมการจัดวงจรไบอัส

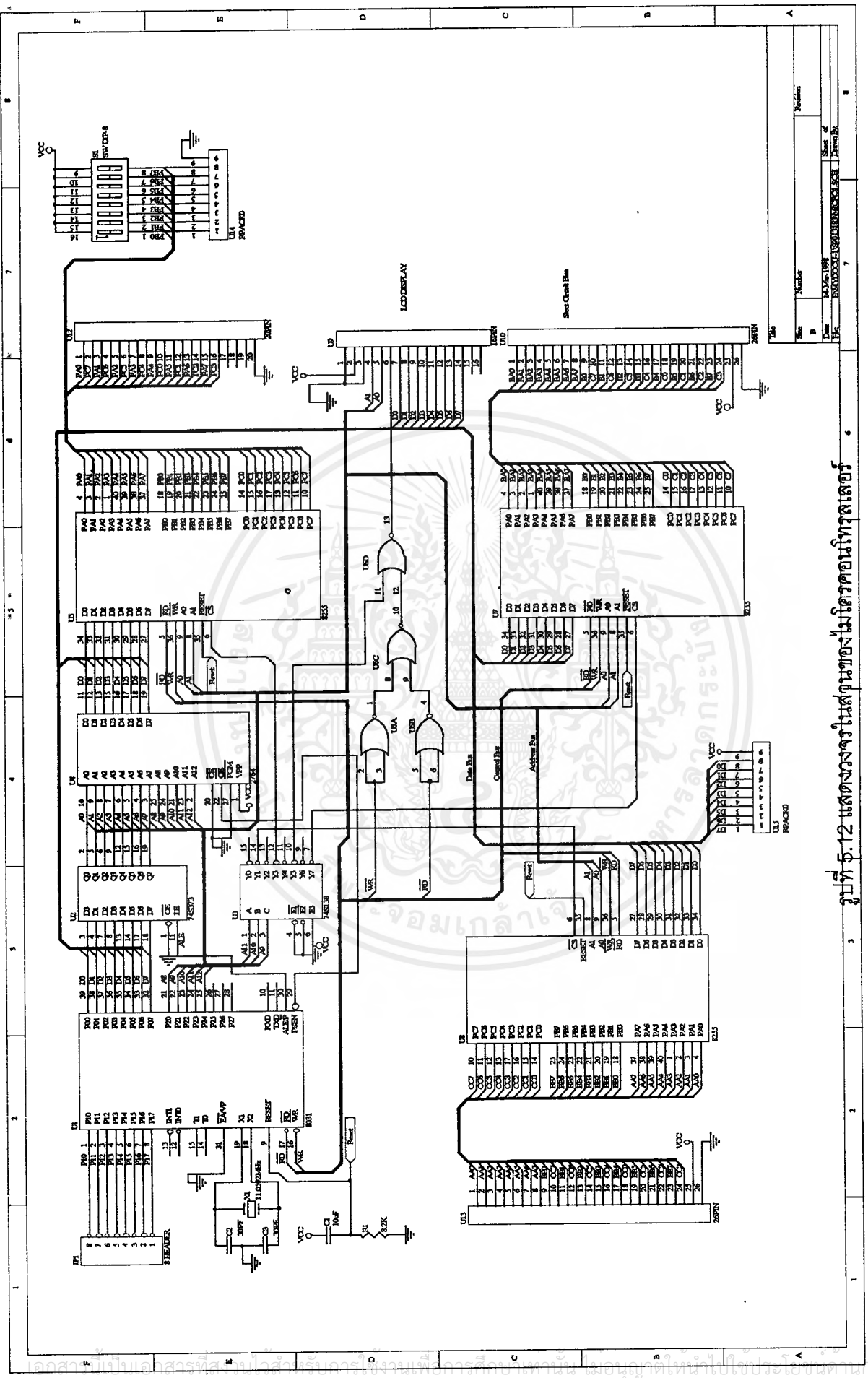
U8 8255 เป็นตัวติดต่อในส่วนของการวัดหาขาของอุปกรณ์

S1 DIP-Switch เป็นการกำหนดตำแหน่งของเครื่อง

ตำแหน่งการถอดรหัสที่ใช้ในวงจร

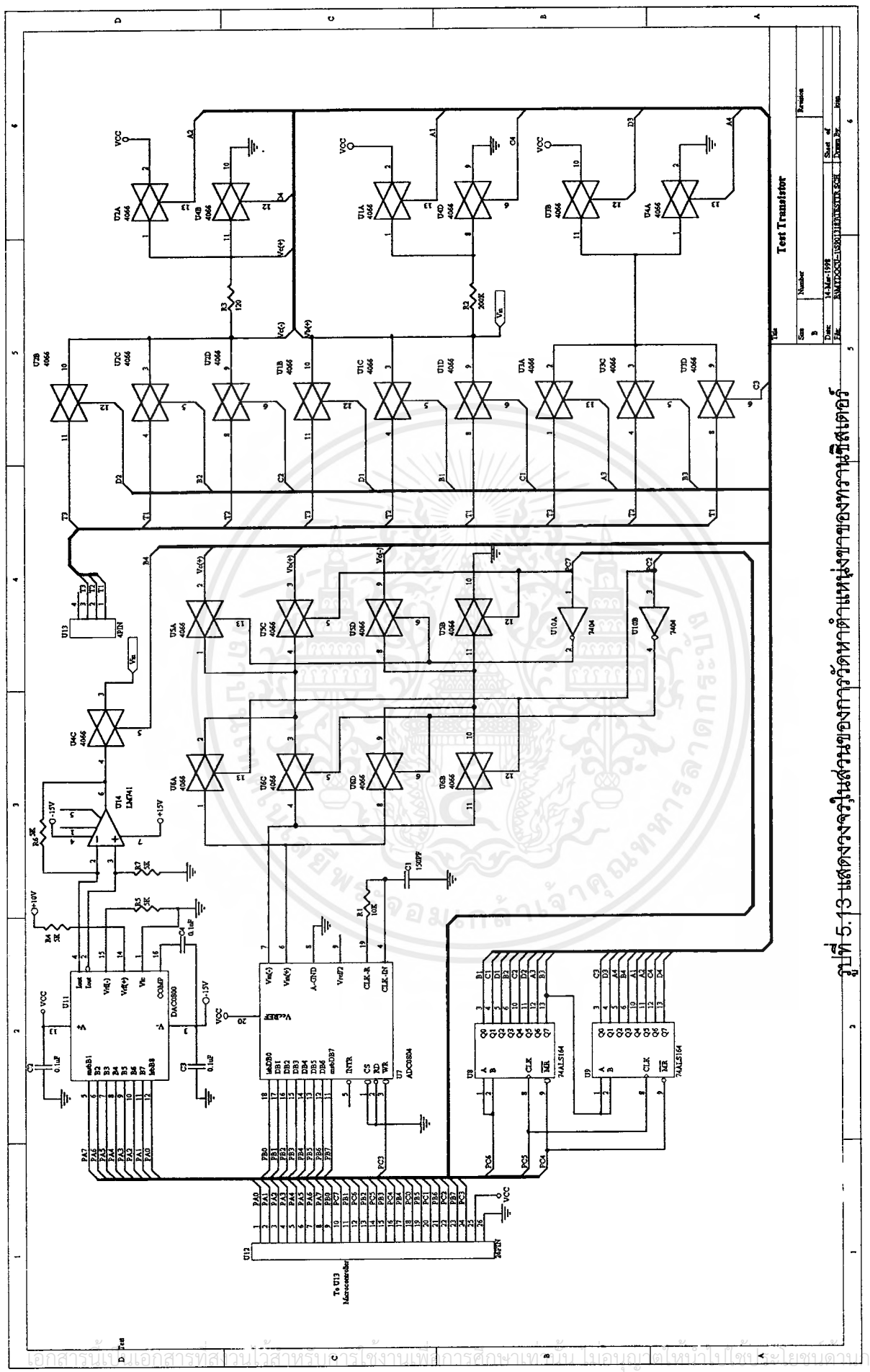
0F800	ควบคุม U8 ในการวัดหาขาอุปกรณ์
0FA00	ควบคุมการเขียนอ่านกับLCD
0FC00	ควบคุม U5 ในการรับส่งข้อมูล IEEE488
0FE00	ควบคุมU7 ในการจัดวงจรไบอัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.12 แสดงวงจรในส่วนของการติดต่อกับคอมพิวเตอร์

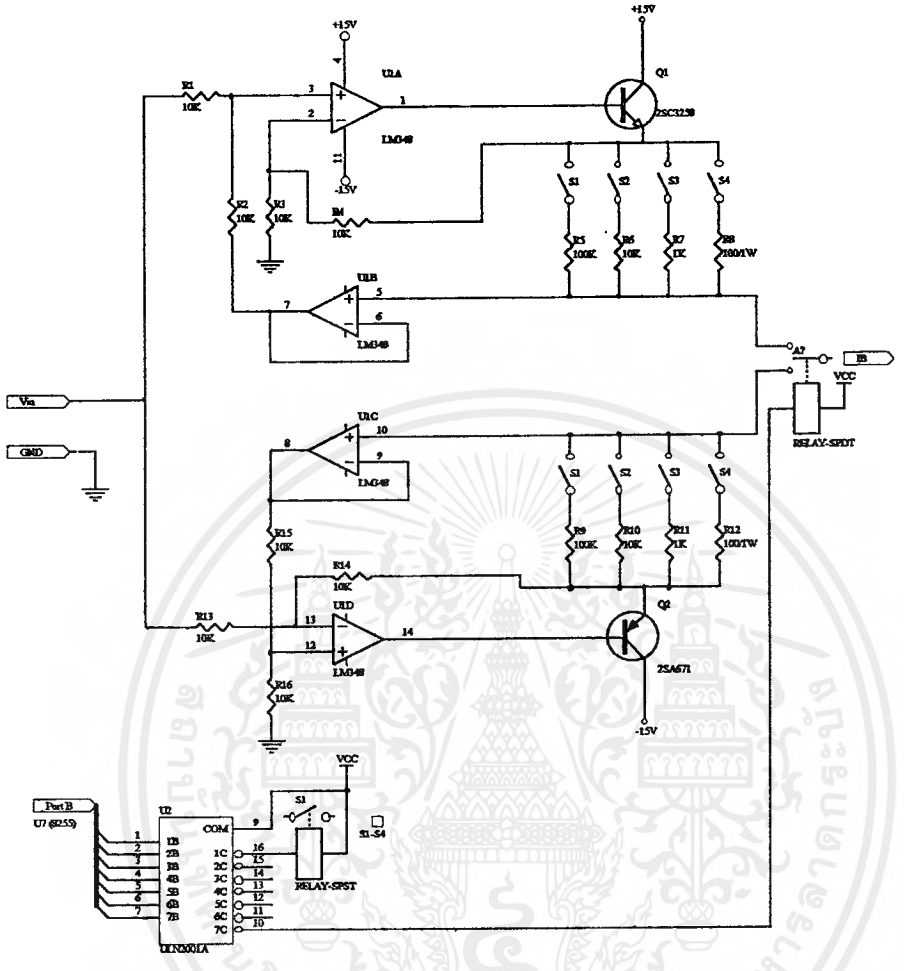
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.13 แสดงวงจรในส่วนของการวัดหาค่าแหน่งขาของทรานซิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.2 ส่วนของวงจรจ่ายกระแส



รูปที่ 5.14 แสดงวงจรจ่ายกระแสสำหรับทรานซิสเตอร์

ในการวัดทรานซิสเตอร์นั้น จะต้องมีการจัดวงจรไบอัส ดังรูปที่ 5.9 โดยการเพิ่มค่าแรงดันทางอินพุท ซึ่งในการทดลองจะพบว่าเมื่อเรากำหนดให้แรงดันทางอินพุทค่าหนึ่งแล้วซึ่งจะได้ค่ากระแสที่ค่าหนึ่ง แล้วเราทำการเพิ่มแรงดันที่คอลเลคเตอร์ ซึ่งจะพบว่ากระแสเบสมีการเปลี่ยนแปลง จะมีผลทำให้การวัดมีความผิดพลาดเนื่องมาจากกระแสเบสไม่คงที่ตลอดย่านการวัด ดังนั้นในาการแก้ปัญหานี้จึ่งได้ทำการสร้างวงจรจ่ายกระแสสำหรับในการวัดทรานซิสเตอร์ วงจรแสดงดังรูปที่ 5.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

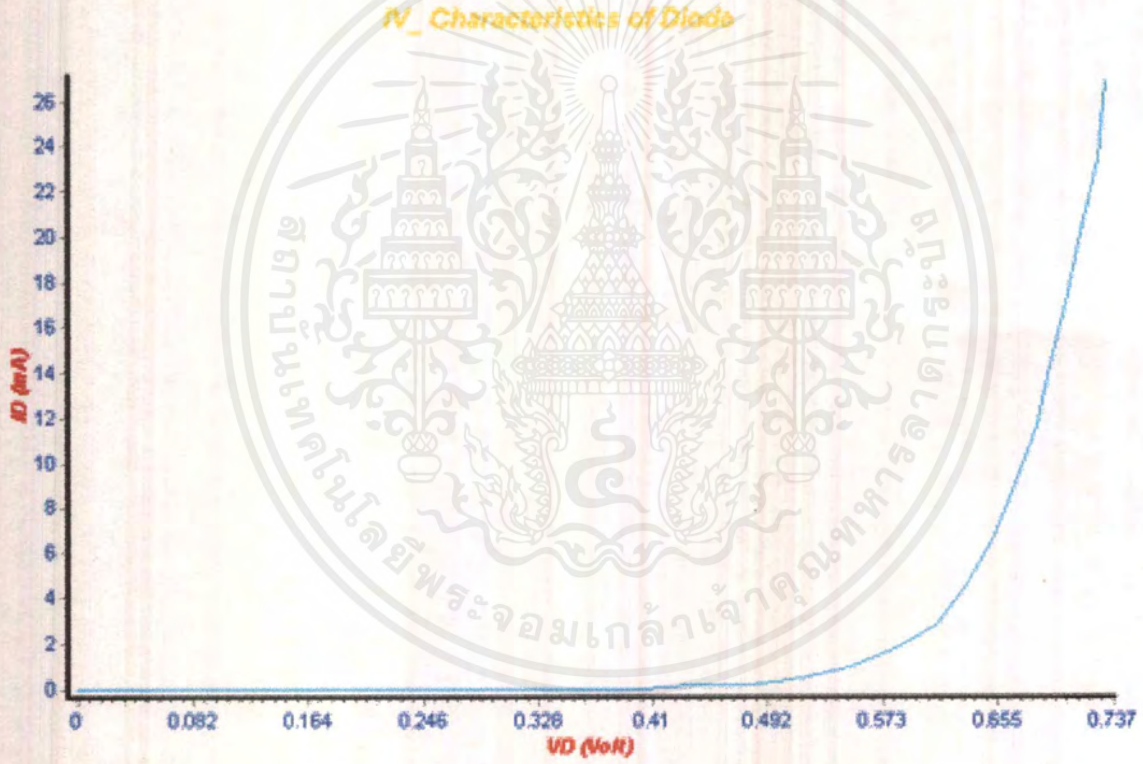
บทที่ 6
การทดลองและการใช้งานโปรแกรม

6.1 การทดลอง

ในการทดลองวัดคุณสมบัติของอุปกรณ์นั้น ซึ่งอุปกรณ์อิเล็กทรอนิกส์ที่สามารถวัดได้จะมี ไดโอด, ทรานซิสเตอร์ ,เฟท ซึ่งสามารถที่จะแสดงได้ดังต่อไปนี้

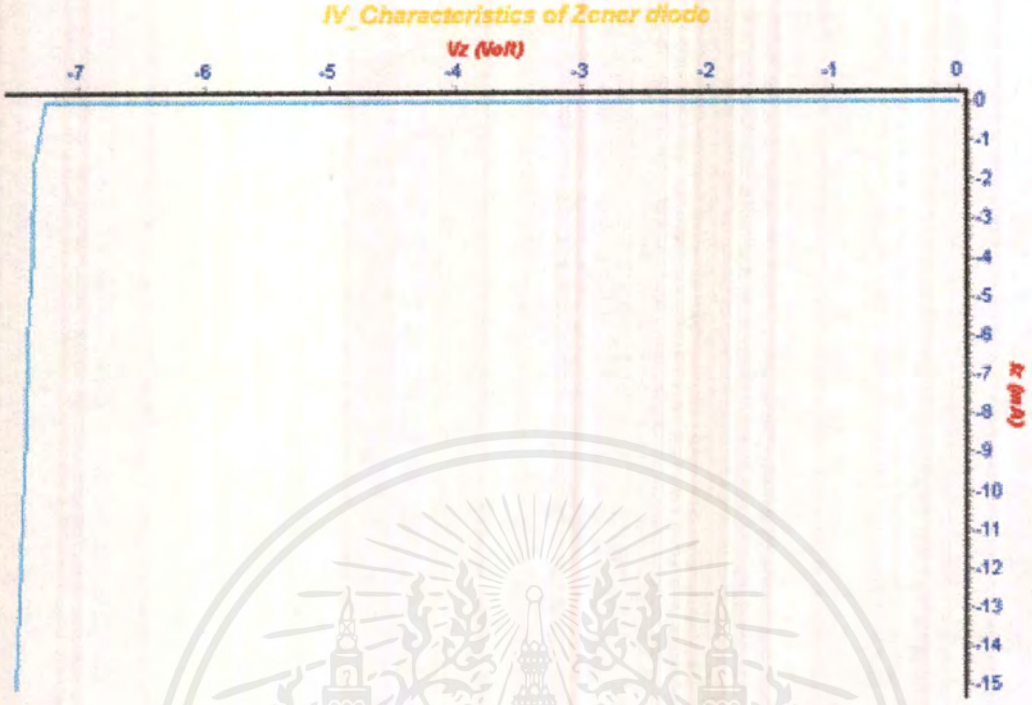
6.1.1 การวัดคุณสมบัติของไดโอด

ในการทดลองนั้นได้ใช้ไดโอดเบอร์ 1N4001 ทำการวัดซึ่งได้ผลการวัดดังรูปที่ 6.1



รูปที่ 6.1 ผลการวัดคุณสมบัติของไดโอดเบอร์ 1N4001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.2 แสดงผลการวัด ซีเนอร์ไดโอด

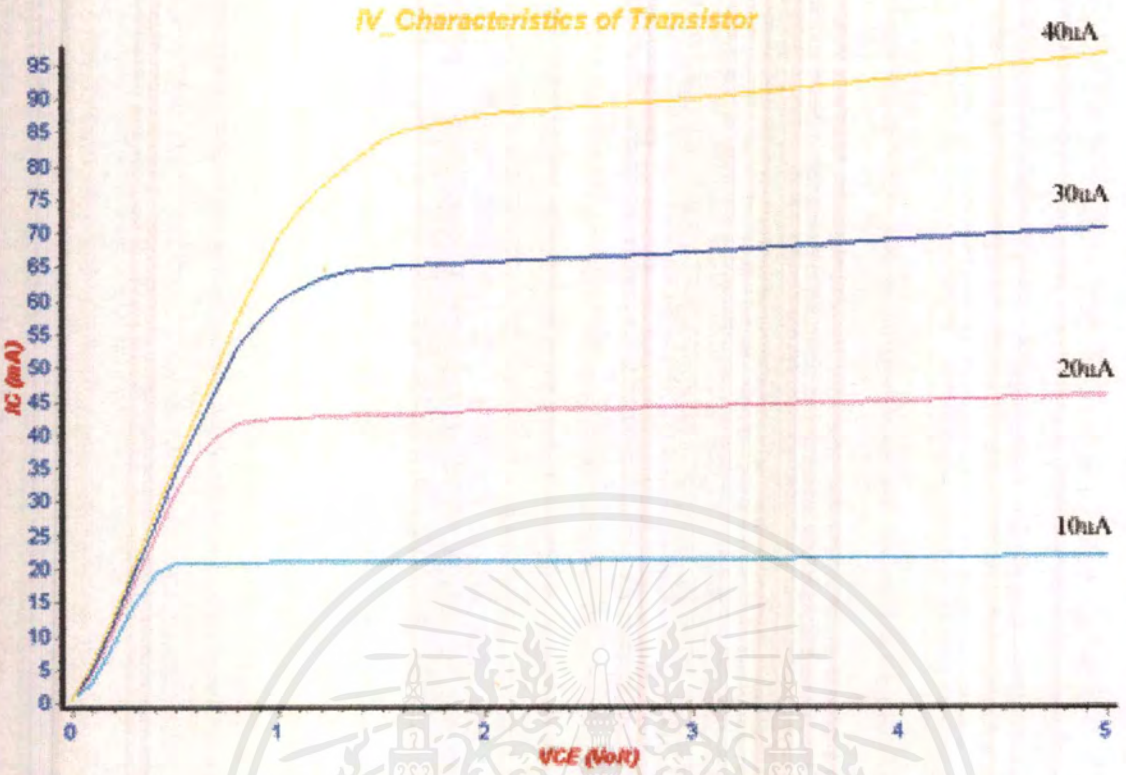
จากรูปที่ 6.2 เป็นผลการวัดของซีเนอร์ไดโอด ซึ่งจะมีค่าแรงดันเบรคดาว์นที่ค่าแรงดันประมาณ 7.3 โวล์ท

6.1.2 การวัดทรานซิสเตอร์

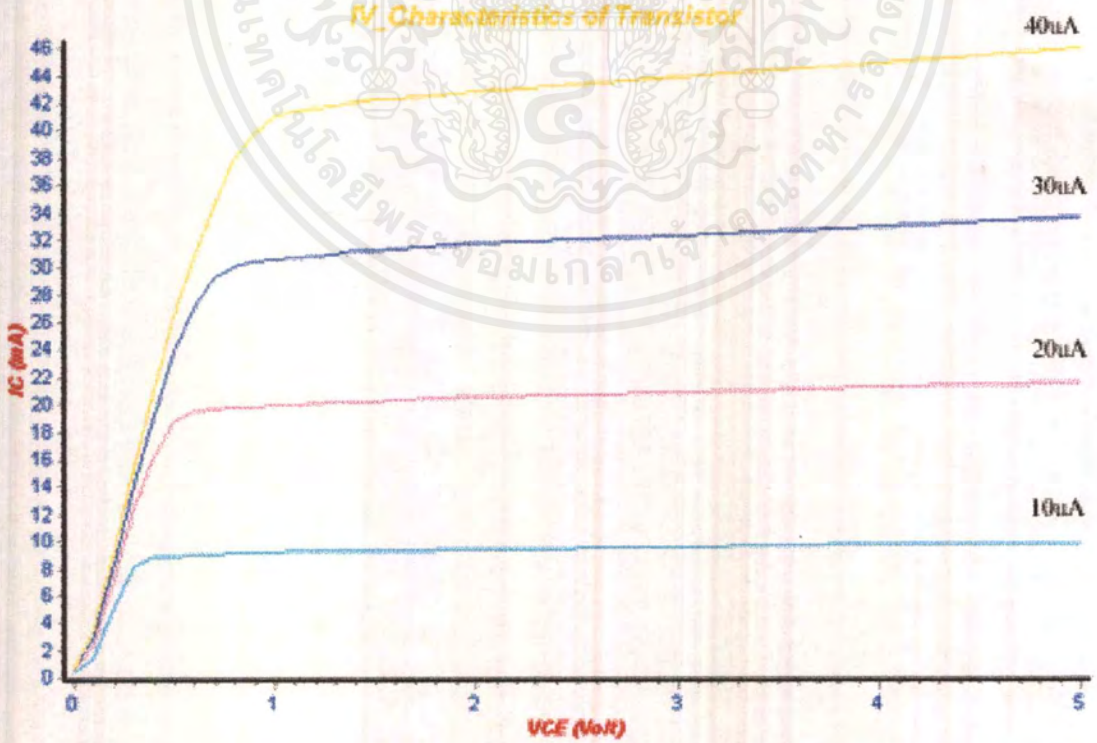
ในการวัดทรานซิสเตอร์นั้นเราจะแบ่งการวัดได้ 2 กรณี คือในกรณีของชนิด N และชนิด P ในการวัดทรานซิสเตอร์นั้นจะต้องมีการตั้งย่านค่ากระแสเบสก่อนที่จะทำการวัดในการเลือกย่านกระแสเบสจะมีผลต่อกราฟคุณสมบัติของอุปกรณ์ด้วย ดังนั้นในการวัดจึงต้องค่าให้เหมาะสมเพื่อให้ได้ค่าที่ถูกต้อง ซึ่งผลการทดลองสามารถแสดงได้ดังรูปที่ 6.3-6.6

ในการวัดทรานซิสเตอร์นั้นการตั้งค่าย่านกระแสเบสจะมีผลต่อกราฟคุณสมบัติ ดังนั้นในการวัดจึงต้องมีการเลือกย่านกระแสเบสที่เหมาะสมกับทรานซิสเตอร์ เช่นทรานซิสเตอร์ที่ใช้ขยายสัญญาณเล็ก ก็ต้องเลือกย่านที่กระแสเบสต่ำๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

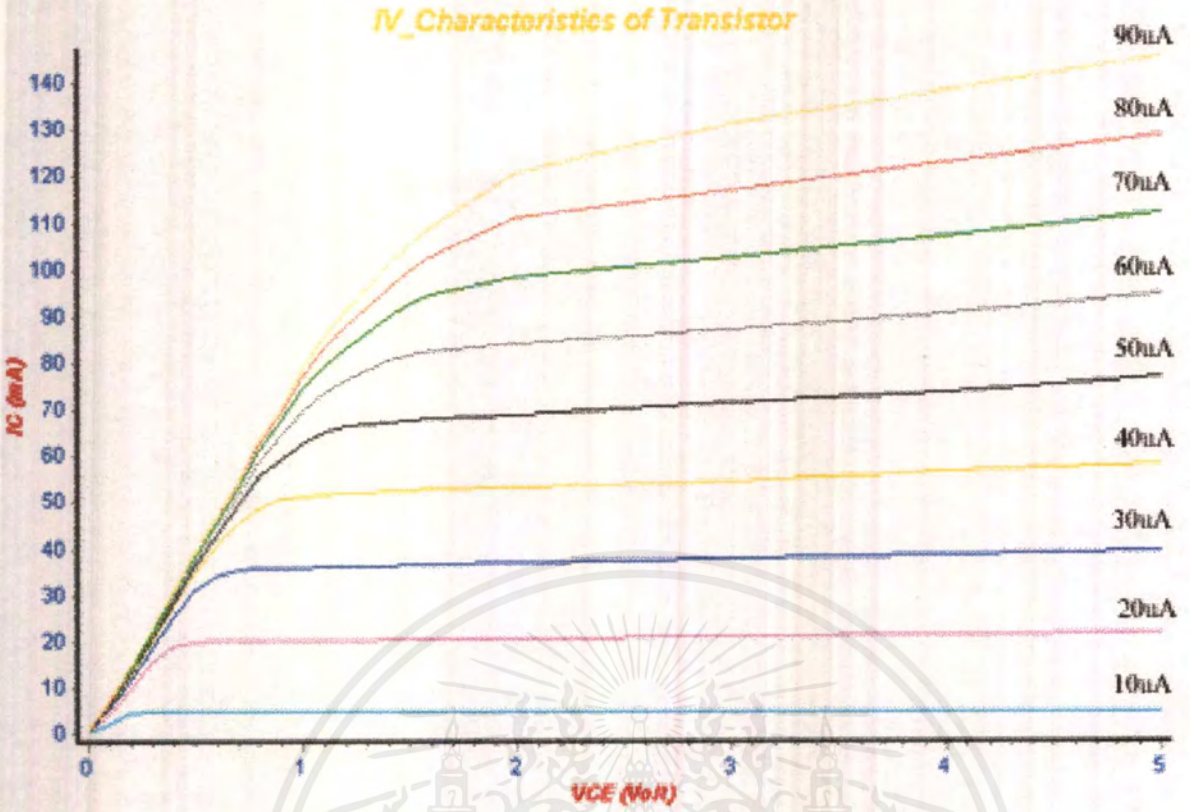


รูปที่ 6.3 แสดงผลการวัดคุณสมบัติของทรานซิสเตอร์เบอร์ D786 ชนิด N

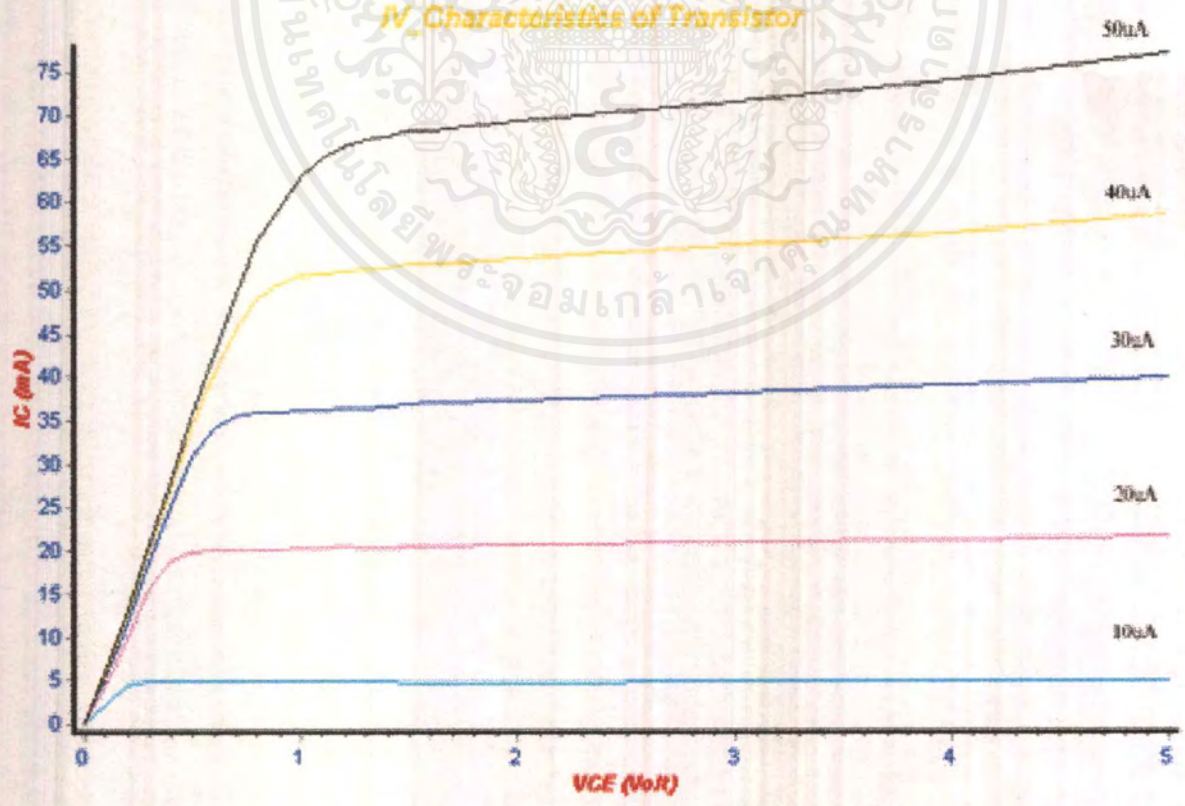


รูปที่ 6.4 แสดงผลการวัดของเบอร์ 2N4037 ชนิด P

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.5 แสดงผลการวัดทรานซิสเตอร์เบอร์ 2SC1815 ในการเลือกค่ากระแสเบส 9 ค่า

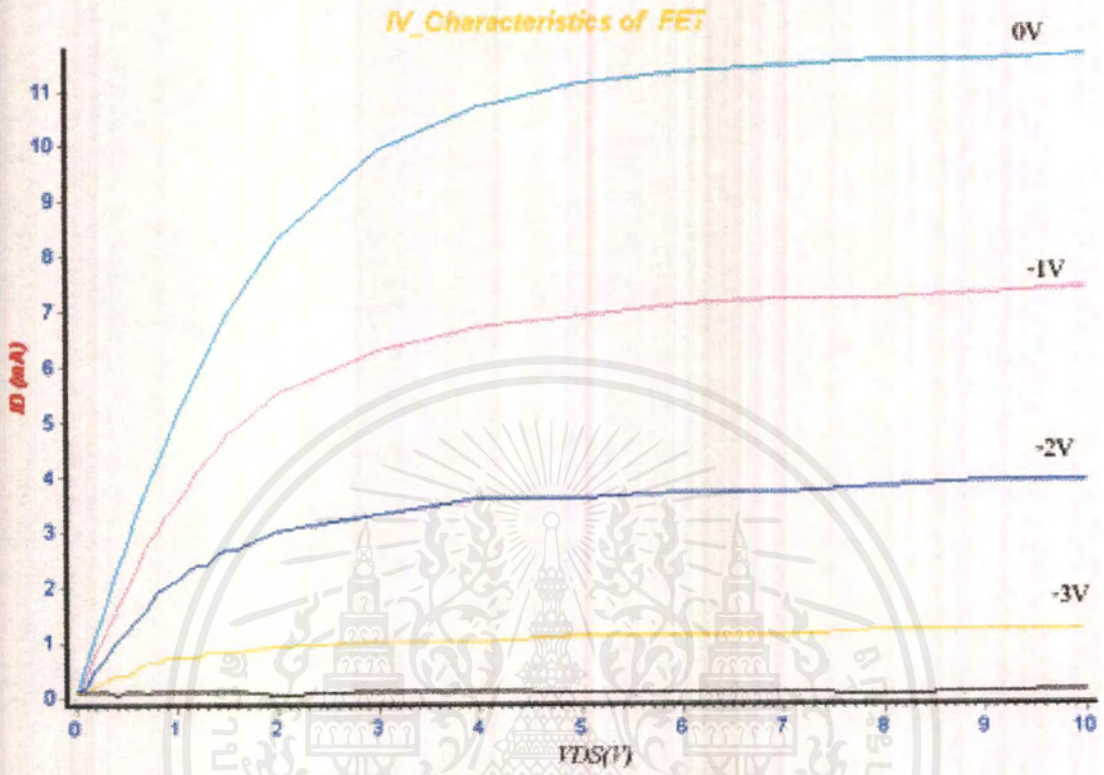


รูปที่ 6.6 แสดงผลการวัดทรานซิสเตอร์เบอร์ 2SC1815 ในการเลือกค่ากระแสเบส 5 ค่า

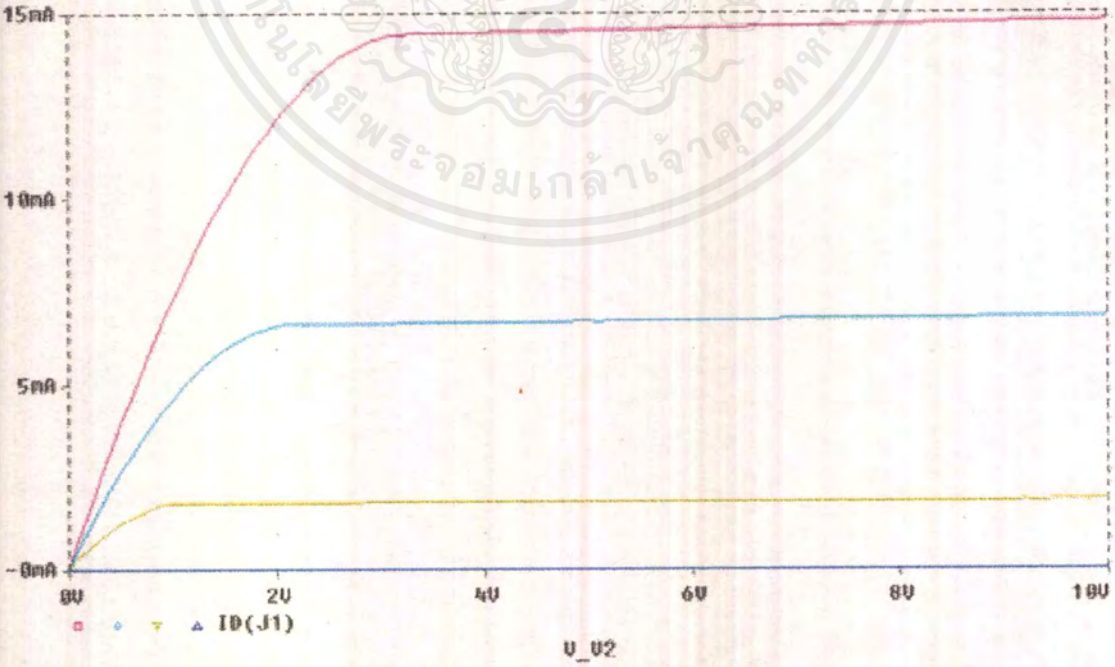
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.3 การวัดคุณสมบัติของเฟท

ในการวัดคุณสมบัติของเฟทสามารถแสดงได้ดังรูปที่ 6.7 ซึ่งเป็นการวัดเฟทชนิด N

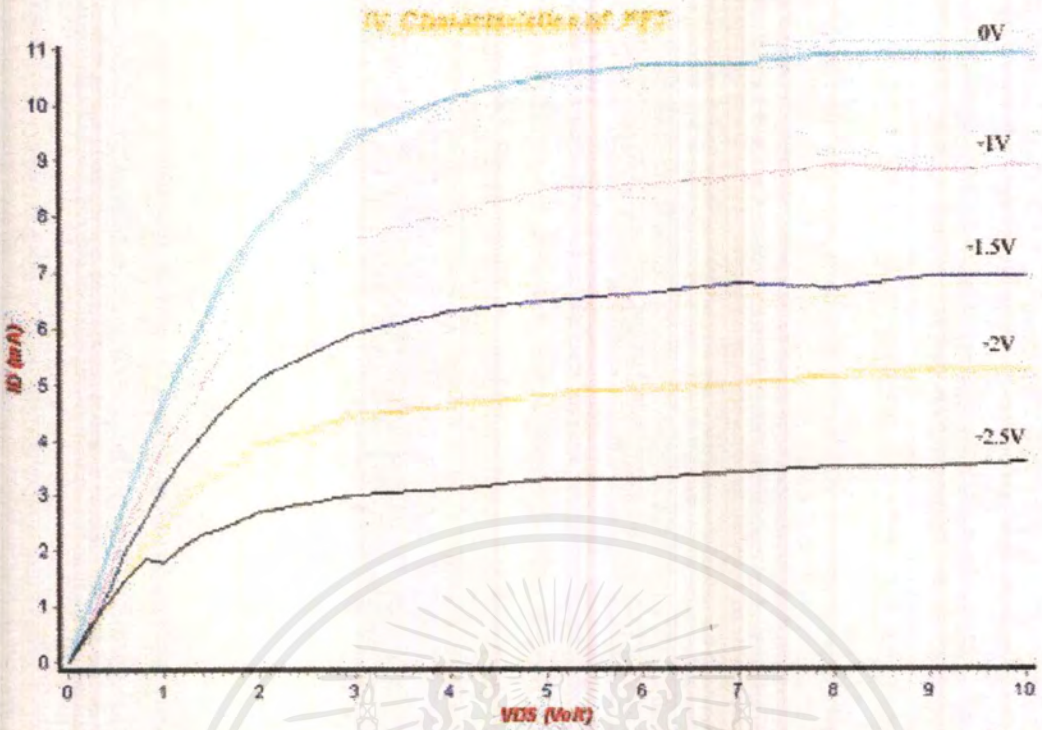


รูปที่ 6.7 ผลการวัดคุณสมบัติของเฟทเบอร์ BF245C

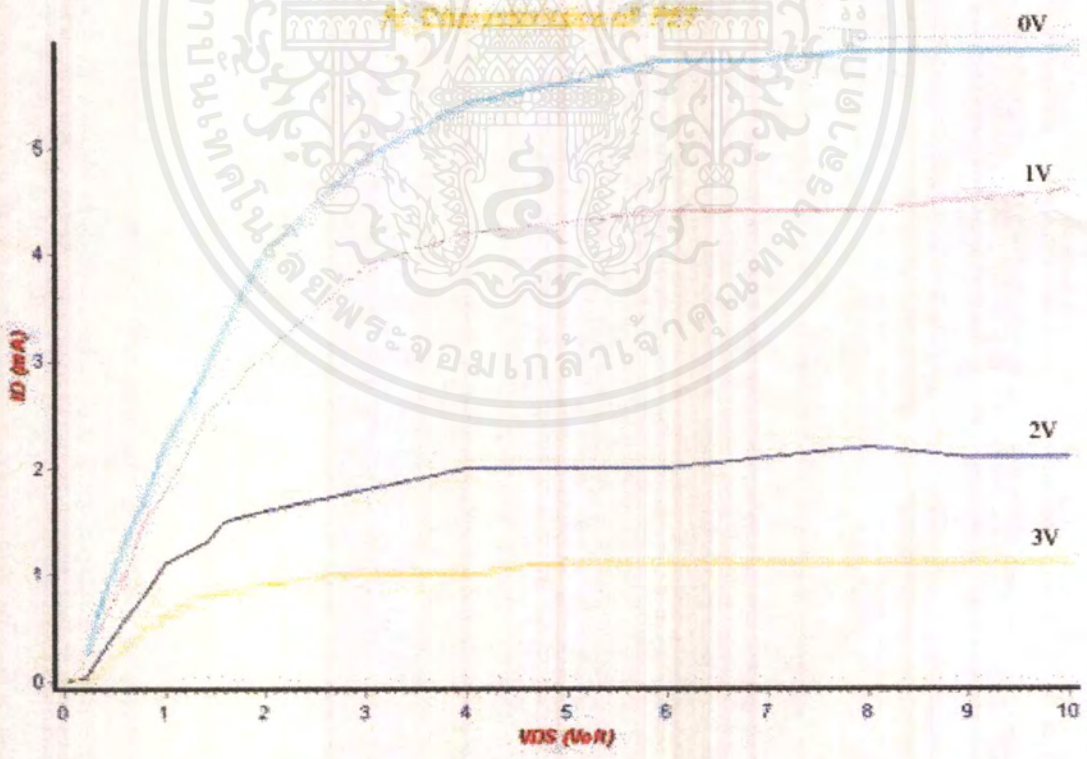


รูปที่ 6.8 ผลการทดลองวัดคุณสมบัติของเฟท เบอร์ BF245C โดยใช้ Pspice

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

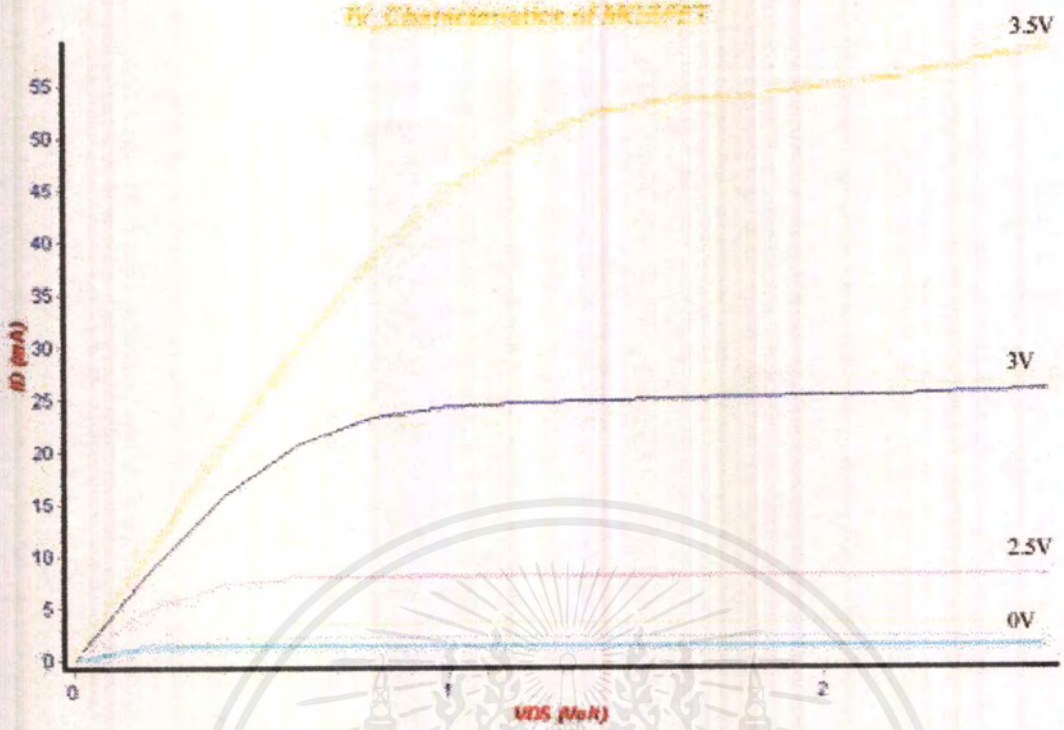


รูปที่ 6.9 แสดงผลการวัดคุณสมบัติของเฟทชนิด N เบอร์ BF245 เมื่อเปลี่ยนค่าแรงดันที่เกทใหม่

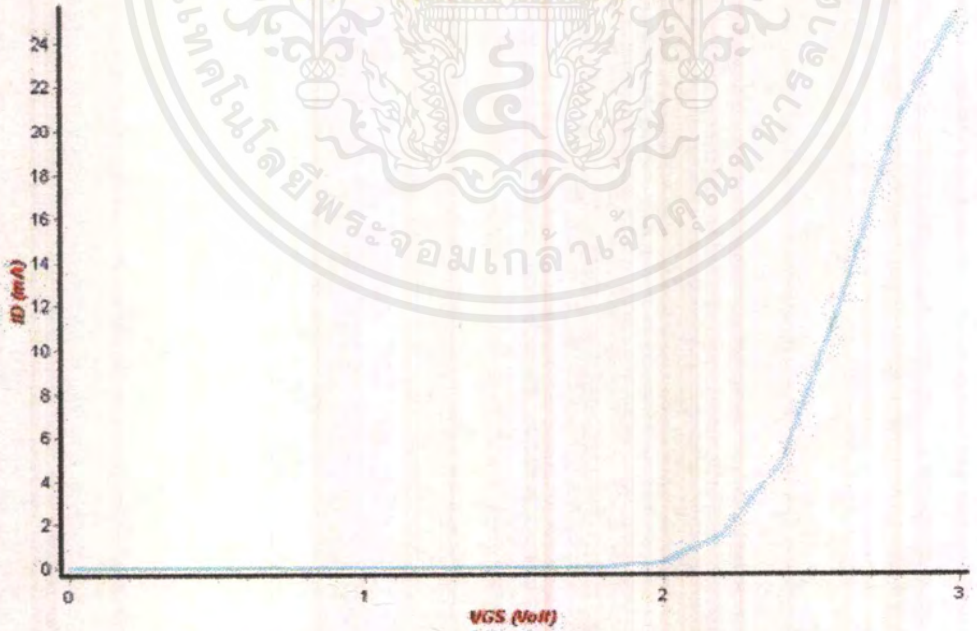


6.10 แสดงผลการวัดคุณสมบัติของเฟทชนิด P เบอร์ 2N5462

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.11 แสดงผลการวัดคุณสมบัติของมอสเฟท ชนิด N เบอร์ BS170



รูปที่ 6.12 แสดงค่าแรงดันขีดเริ่มของมอสเฟท เบอร์ BS170

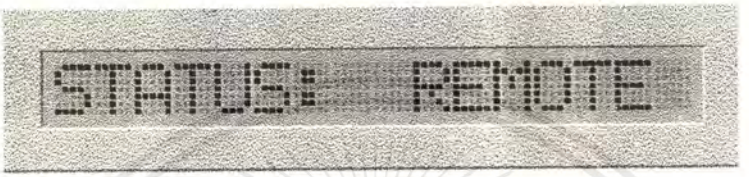
จากรูปที่ 6.12 เป็นการแสดงค่าขีดเริ่ม ซึ่งจากกราฟสามารถที่จะอ่านค่าแรงดันได้ประมาณ 2.1 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 ผลการทดลองในส่วนของไมโครคอนโทรลเลอร์

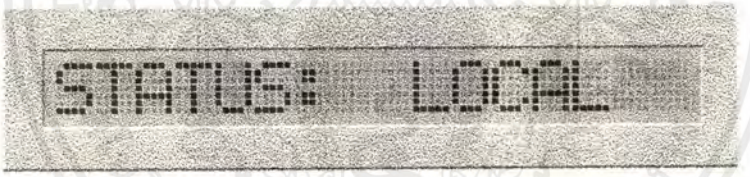
ในการทดลองการสั่งงานการควบคุมจากตัวควบคุม โดยใช้คำสั่งมาตรฐาน IEEE-488 จะใช้โปรแกรมทดสอบของบริษัทเนชั่นแนลอิทรูเมนท์ ซึ่งคำสั่งที่สามารถใช้งานในการควบคุมได้ อธิบายไว้ในส่วนของการออกแบบในบทที่ 5 ผลการทดลองที่ได้สามารถอธิบายได้ดังนี้

การใช้คำสั่ง ibclr เป็นการสั่งให้เครื่องอยู่ในสภาวะเริ่มต้นพร้อมที่จำรับคำสั่งของ GPIB ซึ่งเมื่อใช้คำสั่งนี้แล้ว หน้าจอแสดงผลที่เครื่องจะแสดงดังรูปที่ 6.13



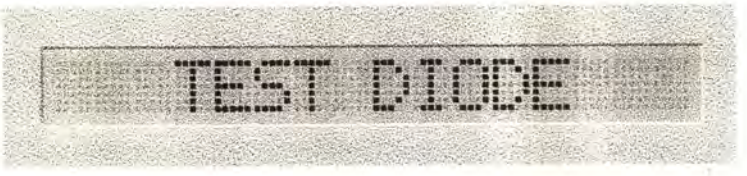
รูปที่ 6.13 แสดงหน้าจอ เมื่อใช้คำสั่ง ibclr

คำสั่ง ibloc เป็นคำสั่งที่ทำให้เครื่องกลับมาอยู่ในสภาวะเหมือนเปิดเครื่องครั้งแรก สามารถใช้ปุ่ม ต่างๆบนเครื่องได้ ซึ่งหน้าจอจะแสดงดังรูปที่ 6.14



รูปที่ 6.14 แสดงหน้าจอเมื่อใช้คำสั่ง ibloc

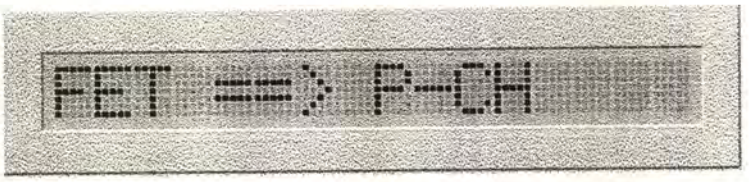
คำสั่ง ibwrt "SG" เป็นคำสั่งที่ใช้ในการทดสอบไดโอด ซึ่งหน้าจอจะแสดงดังรูปที่ 6.15



รูปที่ 6.15 แสดงหน้าจอเมื่อใช้คำสั่ง ibwrt "SG"

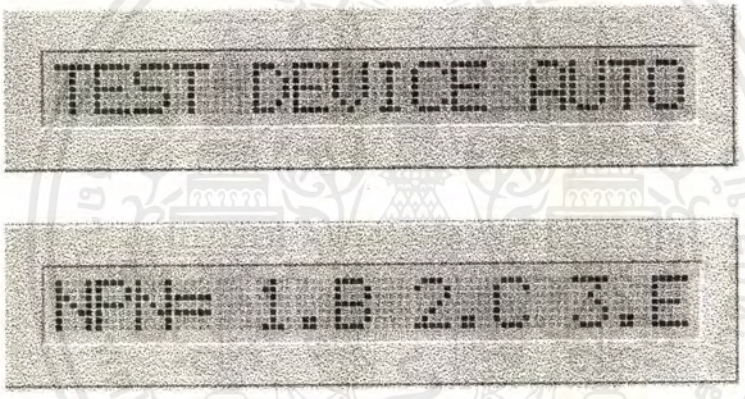
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง ibwr "SFP" เป็นการสั่งให้ทำการวัดคุณสมบัติของเฟทชนิด P



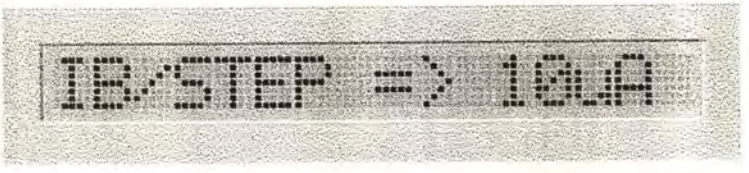
รูปที่ 6.19 แสดงหน้าจอเมื่อใช้คำสั่งlibwrt"SFP"

คำสั่ง ibwrt "STEST" เป็นคำสั่งที่ใช้ในการทดสอบหาตำแหน่งขาของอุปกรณ์ คือ ทรานซิสเตอร์ ซึ่งผลการวัดหาตำแหน่งขาจะแสดงดังหน้าจอตามรูปที่ 6.20



รูปที่ 6.20 แสดงผลการใช้คำสั่ง ibwrt "STEST"

คำสั่ง ibwrt "ST:1" เป็นคำสั่งที่ใช้เลือกย่านกระแสเบส สำหรับทรานซิสเตอร์



รูปที่ 6.21 แสดงหน้าจอเมื่อใช้คำสั่ง ibwrt "ST:1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 การใช้งานโปรแกรม

เมื่อเริ่มทำการเปิดโปรแกรม ซึ่งจะแสดงหน้าจอดังรูปที่ 6.22

รูปที่ 6.22 แสดงหน้าจอเมื่อเปิดโปรแกรมครั้งแรก

ซึ่งจะอธิบายหน้าที่ของปุ่ม ต่างๆได้ดังนี้

Save Graph จะทำหน้าที่เก็บบันทึกรูปกราฟที่ได้จากการวัด

Load Graph จะทำการเปิดไฟล์ข้อมูลของกราฟที่ได้ทำการบันทึกเก็บไว้

Clear GPIB จะเคลียร์สถานะของอุปกรณ์ GPIB ให้อยู่ในสภาวะเริ่มต้น พร้อมที่จะใช้งาน

Delete Data จะทำการลบเส้นกราฟที่วัดได้

Run Graph เริ่มทำการวัดกราฟ

Setup จะตั้งค่าเริ่มต้นต่างๆ เช่น การตั้งค่าตำแหน่งของเครื่อง หรือการเลือกย่านค่ากระแสเบส

Auto Tester จะทำการวัดหาขาของทรานซิสเตอร์อัตโนมัติ

Diode เป็นการเลือกวัดกราฟคุณสมบัติของไดโอด

Zener เป็นการเลือกวัดกราฟคุณสมบัติของซีเนอร์

Transistor เป็นการเลือกวัดกราฟคุณสมบัติของทรานซิสเตอร์

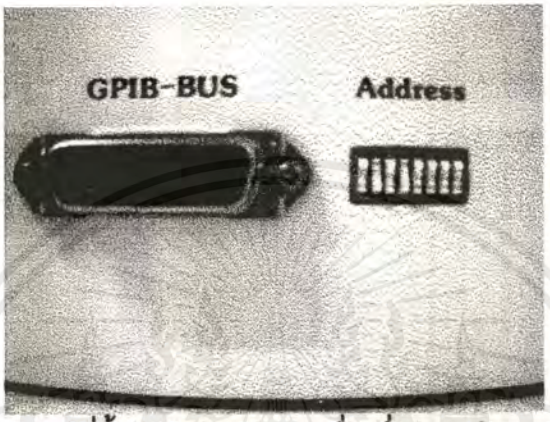
FET เป็นการเลือกวัดกราฟคุณสมบัติของเฟท

Close ปิดโปรแกรม

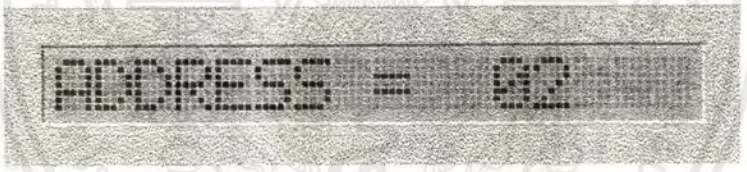
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.1 ขั้นตอนในการใช้โปรแกรม

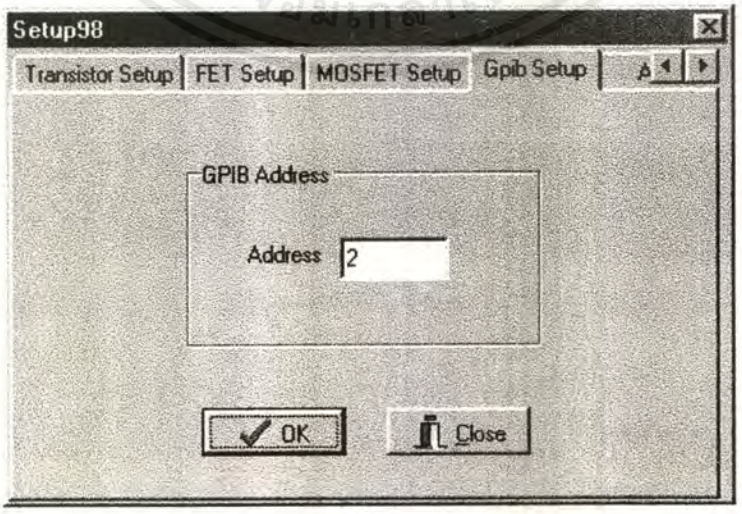
1. เมื่อเปิดโปรแกรมเรียบร้อยแล้ว จะต้องทำการตั้งค่าตำแหน่งของตัวเครื่องฯ ก่อน ซึ่งค่าตำแหน่งนั้นสามารถที่เปลี่ยนได้จากการตั้งค่าสวิทช์ด้านหลังเครื่อง ดังรูปที่ 6.23 เมื่อเปิดเครื่องที่จอแสดงผล (LCD) ก็จะแสดงค่าตำแหน่งของเครื่องให้ทราบ จากนั้นก็จะทำการตั้งค่าตำแหน่งของเครื่องที่ตัวโปรแกรม ด้วยจะเข้าเมนู Setup → GPIB Setup ซึ่งจะแสดงหน้าจอดังรูปที่ 6.25



รูปที่ 6.23 แสดงสวิทช์ที่ตั้งค่าตำแหน่งของเครื่องซึ่งจะอยู่ด้านหลังของเครื่อง



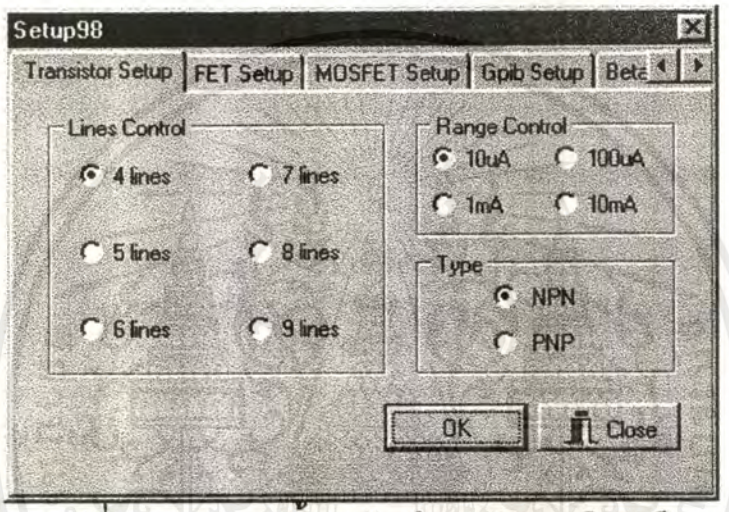
รูปที่ 6.24 แสดงตำแหน่งของเครื่องหลังจากเริ่มเปิดเครื่องใช้งาน



รูปที่ 6.25 แสดงหน้าจอในการตั้งค่าตำแหน่งของเครื่องที่โปรแกรม

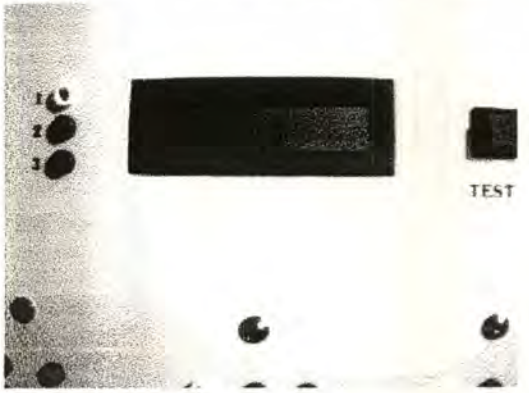
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หลังจากตั้งค่าตำแหน่งเรียบร้อยแล้ว ก็จะกดปุ่ม Clear GPIBที่ ToolBar เพื่อทำการเคลียร์สถานะของอุปกรณ์ทั้งหมดที่ต่ออยู่ในระบบให้อยู่ในสภาวะเริ่มต้น
3. ในกรณีที่ต้องการวัดไดโอด ก็ให้กดปุ่มDiode จากนั้นกดปุ่ม Run Graph
4. ในกรณีที่ต้องการวัดทรานซิสเตอร์ นั้นเมื่อกดปุ่ม Transistor แล้ว ต้องมาตั้งค่าที่เมนู Setup อีกครั้งหนึ่ง โดยจะเป็นการเลือกว่าต้องการที่จะแสดงผลของกระแสเบส กี่เส้น แล้วเลือกค่ากระแสเบสในแต่ละเส้น และเลือกชนิดของทรานซิสเตอร์ที่ต้องการวัดคุณสมบัติซึ่งหน้าจอในการตั้งค่าแสดงดังรูป 6.26



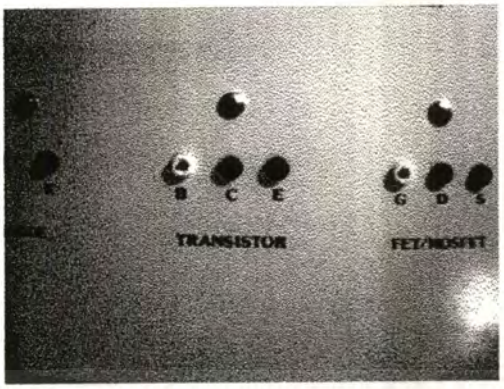
รูปที่ 6.26 แสดงการตั้งค่าก่อนทำการวัดทรานซิสตอร์

5. ในกรณีที่วัดทรานซิสเตอร์แล้วไม่ทราบขาของอุปกรณ์ที่นำมาทดสอบ สามารถที่จะทดสอบหาขาได้ก่อน โดยการนำอุปกรณ์ไปต่อยังจุดทดสอบขา แล้วกดปุ่ม AutoTester หลังจากนั้นตำแหน่งขาของอุปกรณ์ก็แสดงที่หน้าจอ monitor ทางด้านขวามือ และที่หน้าจอแสดงผล(LCD)



รูปที่ 6.27 จุดทดสอบการหาตำแหน่งขาของทรานซิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



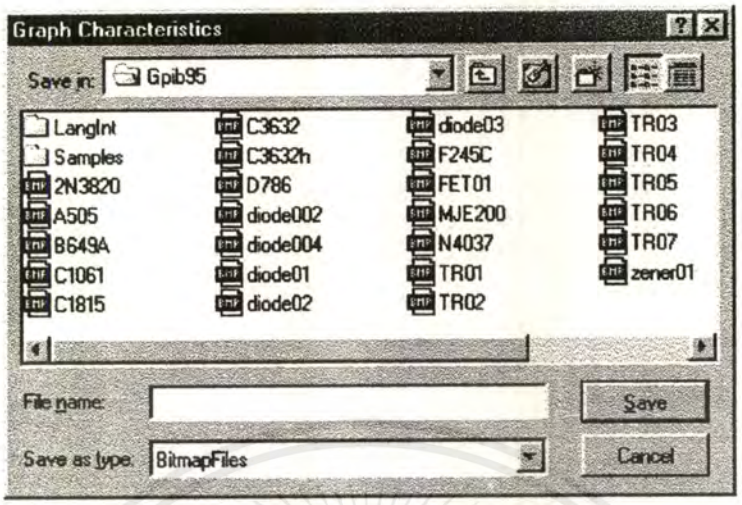
รูปที่ 6.28 จุดทดสอบคุณสมบัติของอุปกรณ์ซึ่งจะต้องทราบตำแหน่งขาก่อนที่จะวัด

6. เมื่อตั้งค่าเริ่มต้นของทรานซิสเตอร์เสร็จแล้วก็จะทำการกดปุ่ม Run Graph ซึ่งจะแสดงหน้าจอดังรูปที่ 6.29

รูปที่ 6.29 แสดงหน้าจอเมื่อทำการวัดทรานซิสเตอร์เสร็จแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เมื่อต้องการบันทึกกราฟที่วัดได้ ก็กดปุ่ม Save ซึ่งจะทำการบันทึกกราฟที่ได้

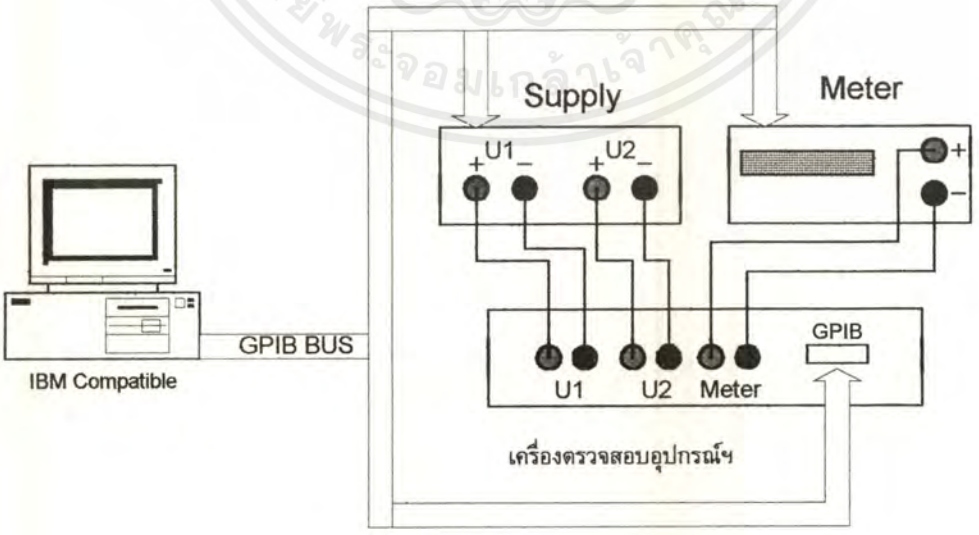


รูปที่ 6.30 แสดงหน้าจอเมื่อทำการ Save ภาพที่ได้จากการวัด

8. ในการวัดอุปกรณ์ตัวอื่น ก็เช่นเดียวกัน อย่างเช่นในกรณีของการวัดเฟทก็จะต้องไปทำการตั้งค่าก่อนว่าต้องการวัดเส้นกราฟกี่เส้น วิธีการอย่างอื่นก็เช่นเดียวกับการวัดทรานซิสเตอร์

6.4 การต่อใช้งานกับอุปกรณ์ภายนอก

ในการต่อใช้งานนั้นอุปกรณ์ภายนอกที่จะต่อร่วมด้วยจะได้แก่ Power Supply และ Multimeter ซึ่งอุปกรณ์ทั้งสองก็จะต่อกับระบบบัส IEEE-488 ด้วย ซึ่งการต่อก็จะแสดงดังรูปที่ 6.31



รูปที่ 6.31 การต่อสายระหว่างเครื่องฯ กับอุปกรณ์ต่อร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.32 ตำแหน่งด้านหลังของเครื่องที่จะต้องต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
สรุปและวิจารณ์ผลการทดลอง

ในการทดลองวัดคุณสมบัติของอุปกรณ์ที่ได้ทำการวัดผ่านการควบคุมแบบ IEEE488.2(GPIB) นั้น จะพบว่าในการวัดนั้นจะมีปัญหาในเรื่องความเร็วในการวัด เนื่องจากในงานนี้ได้ใช้ มัลติมิเตอร์ (Multimeter) มาทำการวัดค่ากระแสเพื่อนำไปแสดงเส้นกราฟ ซึ่ง มัลติมิเตอร์นั้นเป็นเครื่องมือวัดที่มีความเร็วช้าที่สุด จึงทำให้ในการวัดเป็นไปได้ช้า ซึ่งถ้าเราทำการตั้ง ค่าเวลาในการอ่านของมัลติมิเตอร์ให้เร็วขึ้นก็จะพบปัญหาของการอ่านค่าผิดพลาดซึ่งจะทำให้กราฟที่ได้ในการวัด ก็จะไม่ถูกต้อง ในทดลองครั้งนี้ได้ตั้งค่าของเวลาในการอ่านมัลติมิเตอร์ไว้ที่ 1 วินาที ซึ่งเป็นค่าที่ให้ความถูกต้องมาก

ในการทดลองการรับส่งข้อมูลผ่านระบบบัส IEEE488 นั้นได้ทำการจำลองเครื่องที่สร้างขึ้นมาให้ทำการส่งข้อมูลของการวัดขาทรานซิสเตอร์ ส่งกลับมายังตัวควบคุม (คอมพิวเตอร์) ในการทดลองก็สามารถที่จะรับส่งข้อมูลได้ และตัวเครื่องทีสร้างขึ้นมาจะใช้คำสั่งมาตรฐานของ IEEE488 ทุกประการ

ส่วนในการวัดคุณสมบัติของตัวอุปกรณ์นั้น ในการวัดของไดโอดนั้น จะไม่พบปัญหาอะไรเกิดขึ้น ผลการทดลองที่ได้ก็เป็นไปตามทฤษฎีทุกประการ เมื่อแรงดันที่ป้อนให้มากกว่า 0.6 โวลท์ ไดโอดก็จะเริ่มนำกระแส ส่วนในการวัดของทรานซิสเตอร์นั้นจะต้องมีการเลือกค่าย่านกระแสเบสก่อนที่จะทำการวัด สามารถเลือกย่านกระแสเบสได้ 4 ย่าน และต้องกำหนดว่าต้องการให้แสดงค่ากระแสเบสกี่เส้น ในการเลือกค่าย่านกระแสเบสนั้นจะต้องเลือกให้เหมาะสมกับตัวอุปกรณ์ เพื่อที่จะได้กราฟที่มีความถูกต้อง ซึ่งในการวัดทรานซิสเตอร์นั้นสามารถที่จะวัดได้ทั้งชนิด N และ P ส่วนการวัดเฟทนั้นก็สามารถที่จะวัดได้ ซึ่งจะต้องตั้งค่าแรงดันที่เกทให้เหมาะสม ในการวัดมอสเฟทนั้นวัดเฉพาะชนิด N เท่านั้น ซึ่งผลการวัดที่ได้ก็ถูกต้อง และยังสามารถที่จะวัดค่าแรงดันขีดเริ่มของมอสเฟทได้ ในการวัดเฟท และมอสเฟท นั้นยังมีข้อผิดพลาดบ้าง จากการตั้งค่าแรงดันที่ไม่เหมาะสม จึงทำให้กราฟการวัดที่ได้ไม่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวทางการพัฒนาปรับปรุง

ในการพัฒนาปรับปรุงความสามารถของเครื่องตรวจสอบคุณสมบัติอุปกรณ์อิเล็กทรอนิกส์นั้นก็ทำได้โดยการเพิ่มความเร็วในการวัดให้สูงขึ้น และพัฒนาให้สามารถวัดคุณสมบัติต่างๆได้มากขึ้น เช่น การวัดทรานซิสเตอร์ ก็สามารถวัดคุณสมบัติความสัมพันธ์อัตราการขยายกระแสกับค่ากระแสคอลเลคเตอร์ ซึ่งกราฟนี้จะมีประโยชน์ในการนำทรานซิสเตอร์ไปใช้งาน และคุณสมบัติอีกอย่างคือ การตอบสนองทางด้านความถี่ของทรานซิสเตอร์ จะมีประโยชน์ในการนำเอาทรานซิสเตอร์ไปใช้ขยายสัญญาณที่ความถี่ต่างๆ จะทำให้เราทราบว่า ณ ที่ความถี่หนึ่งทรานซิสเตอร์นั้นจะมีค่าอัตราการขยายกระแสเป็นเท่าไร ซึ่งจะต้องมีการเพิ่มเครื่องมือที่ใช้ในการวัดเพิ่มเข้ามา ได้แก่ เครื่องกำเนิดสัญญาณ และ ออสซิโลสโครบ ซึ่งทั้งสองเครื่องก็สามารถที่จะสั่งงานทาง IEEE488 ได้ทั้งคู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Win32 Interactive Control Utility

Chapter 6

This chapter introduces you to Win32 Interactive Control, the interactive control utility you can use to communicate with GPIB devices interactively.

Overview

With the Win32 Interactive Control utility, you communicate with the GPIB devices through functions you enter at the keyboard. For specific information about communicating with your particular device, refer to the manual that came with the device. You can use the Win32 Interactive Control utility to practice communication with the instrument, troubleshoot problems, and develop your application.

The Win32 Interactive Control utility helps you to learn about your instrument and to troubleshoot problems by displaying the following information on your screen after you enter a command:

- Results of the status word (`ibsta`) in hexadecimal notation
- Mnemonic constant of each bit set in `ibsta`
- Mnemonic value of the error variable (`iberr`) if an error exists (the ERR bit is set in `ibsta`)
- Count value for each read, write, or command function
- Data received from your instrument

Getting Started with Win32 Interactive Control

This section shows you how to use the Win32 Interactive Control utility to test a sequence of GPIB calls.

To run the Win32 Interactive Control utility, select the **Win32 Interactive Control** item under **Start»Programs»GPIB Software**.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

When the Win32 Interactive Control utility starts, it displays the following banner message:

```
Win32 Interactive Control
Copyright 1996 National Instruments Corporation
All rights reserved

Type `help' for help or `q' to quit

:
```

First, you must open either a board handle or device handle to use for further GPIB calls. Use `ibdev` to open a device handle, `ibfind` to open a board handle, or the `set 488.2` command to switch to a 488.2 prompt. For help on any Win32 Interactive Control command, type in `help` followed by the command, for example `help ibdev` or `help set`.

If you want to use device-level calls, open a device handle using `ibdev`. The following example shows you how to use `ibdev` to open a device, assign it to access board `gpib0`, choose a primary address of 6 with no secondary address, set a timeout of 10 seconds, enable the END message, and disable the EOS mode:

```
:ibdev
   enter board index: 0
   enter primary address: 6
   enter secondary address: 0
   enter timeout: T10s
   enter 'EOI on last byte' flag: 1
   enter end-of-string mode/byte: 0
ud0:
```

If you enter a command and no parameters, you are prompted for the necessary arguments. If you already know the required arguments, you can enter them from the command line, as follows:

```
:ibdev 0 6 0 T10s 1 0

ud0:
```

The new prompt, `ud0`, represents a device-level handle that you can use for further GPIB calls. To clear the device, use `ibclr`, as follows:

```
ud0: ibclr
[0100] (cmpl)
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

To write data to the device, use `ibwrt`, as follows. Make sure that you refer to the instrument user manual that came with your GPIB instrument for specific command messages.

```
ud0: ibwrt
   enter string: "*RST; VAC; AUTO; TRIGGER 2; *SRE 16"
[0100] (cmpl)
count: 35
```

Or, equivalently:

```
ud0: ibwrt "*RST; VAC; AUTO; TRIGGER 2; *SRE 16"
[0100] (cmpl)
count: 35
```

To send a trigger, use `ibtrg`, as follows:

```
ud0: ibtrg
[0100] (cmpl)
```

To read data from your device, use `ibrd`. The data that is read from the instrument is displayed. For example, to read 18 bytes, enter the following:

```
ud0: ibrd
   enter byte count: 18
[0100] (cmpl)
count: 18
4e 44 43 56 20 30 30 30      N D C V  0 0 0
2e 30 30 34 37 45 2b 30      . 0 0 4 7 E + 0
0a 0a                        . .
```

Or, equivalently:

```
ud0: ibrd 18
[0100] (cmpl)
count: 18
4e 44 43 56 20 30 30 30      N D C V  0 0 0
2e 30 30 34 37 45 2b 30      . 0 0 4 7 E + 0
0a 0a                        . .
```

When you are finished communicating with the device, make sure you put it offline using the `ibonl` command, as follows:

```
ud0: ibonl 0
[0100] (cmpl)

:
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The `ibonl` command properly closes the device handle and the `ud0` prompt is no longer present.

Win32 Interactive Control Syntax

The following special rules apply to making calls from theWin32 Interactive Control utility:

- The `ud` or `BoardId` parameter is implied by the Win32 Interactive Control prompt, therefore it is never included in the call.
- The `count` parameter to functions is unnecessary because buffer lengths are automatically determined by Win32 Interactive Control.
- Function return values are handled automatically by Win32 Interactive Control. In addition to printing out the return `ibsta` value for the function, it also prints other return values.
- If you do not know what parameters are appropriate to pass to a given function call, type in the function name and press <Enter>. The Win32 Interactive Control utility then prompts you for each required parameter.

Number Syntax

You can enter numbers in either hexadecimal or decimal format.

Hexadecimal numbers—You must prefix hexadecimal numbers with 0x. For example, `ibpad 0x16` sets the primary address to 16 hexadecimal (22 decimal).

Decimal numbers—Enter the number only. For example, `ibpad 22` sets the primary address to 22 decimal.

String Syntax

You can enter strings as an ASCII character sequence, hex bytes, or special symbols.

ASCII character sequence—You must enclose the entire sequence in quotation marks.

Hex byte—You must use a backslash character and an x followed by the hex value. For example, hex 40 is represented by `\x40`.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special symbols—Some instruments require special termination or end-of-string (EOS) characters that indicate to the device that a transmission has ended. The two most common EOS characters are `\r` and `\n`. `\r` represents a carriage return character and `\n` represents a linefeed character. You can use these special characters to insert the carriage return and linefeed characters into a string, as in `"F3R5T1\r\n"`.

Address Syntax

Many of the NI-488.2 routines have an address or address list parameter. An address is a 16-bit representation of the GPIB device address. The primary address is stored in the low byte and the secondary address, if any, is stored in the high byte. For example, a device at primary address 6 and secondary address 0x67 has an address of 0x6706. A NULL address is represented as 0xffff. An address list is represented by a comma-separated list of addresses, such as `1,2,3`.

Win32 Interactive Control Commands

Tables 6-1 and 6-2 summarize the syntax of NI-488 functions in the Win32 Interactive Control utility. Table 6-3 summarizes the syntax of NI-488.2 routines in the Win32 Interactive Control utility. Table 6-4 summarizes the auxiliary functions that you can use in the Win32 Interactive Control utility. For more information about the function parameters, use the online help. If you enter only the function name, the Win32 Interactive Control utility prompts you for parameters.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 6-1. Syntax for Device-Level NI-488 Functions in Win32 Interactive Control

Syntax	Description
`ibask option`	Return configuration information where `option` is a mnemonic for a configuration parameter
`ibbna bname`	Change access board of device where `bname` is symbolic name of new board
`ibclr`	Clear specified device
`ibconfig option value`	Alter configurable parameters where `option` is mnemonic for a configuration parameter
`ibdev BdIndx pad sad tmo eot eos`	Open an unused device; `ibdev` parameters are `BdIndx pad sad tmo eot eos`
`ibeos v`	Change/disable EOS message
`ibeot v`	Enable/disable END message
`ibln pad sad`	Check for presence of device on the GPIB at `pad, sad`
`ibloc`	Go to local
`ibonl v`	Place device online or offline
`ibpad v`	Change primary address
`ibpct`	Pass control
`ibppc v`	Parallel poll configure
`ibrd count`	Read data where `count` is the bytes to read
`ibrda count`	Read data asynchronously where `count` is the bytes to read
`ibrdf flname`	Read data to file where `flname` is pathname of file to read
`ibrpp`	Conduct a parallel poll
`ibrsp`	Return serial poll byte
`ibsad v`	Change secondary address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 6-1. Syntax for Device-Level NI-488 Functions in Win32 Interactive Control (Continued)

Syntax	Description
`ibstop`	Abort asynchronous operation
`ibtmo v`	Change/disable time limit
`ibtrg`	Trigger selected device
`ibwait mask`	Wait for selected event where `mask` is a hex or decimal integer or a list of mask bit mnemonics, such as `ibwait TIMO CMPL`
`ibwrt wrtbuf`	Write data
`ibwrta wrtbuf`	Write data asynchronously
`ibwrtf flname`	Write data from a file where `flname` is pathname of file to write

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 6-2. Syntax for Board-Level NI-488 Functions in Win32 Interactive Control

Syntax	Description
`ibask option`	Return configuration information where `option` is a mnemonic for a configuration parameter
`ibcac v`	Become active Controller
`ibcmd cmdbuf`	Send commands
`ibcmda cmdbuf`	Send commands asynchronously
`ibconfig option value`	Alter configurable parameters where `option` is mnemonic for a configuration parameter
`ibdma v`	Enable/disable DMA
`ibeos v`	Change/disable EOS message
`ibeot v`	Enable/disable END message
`ibfind udname`	Return unit descriptor where `udname` is the symbolic name of board (for example, `gpib0`)
`ibgts v`	Go from Active Controller to standby
`ibist v`	Set/clear `ist`
`iblines`	Read the state of all GPIB control lines
`ibln pad sad`	Check for presence of device on the GPIB at `pad`, `sad`
`ibloc`	Go to local
`ibonl v`	Place device online or offline
`ibpad v`	Change primary address
`ibppc v`	Parallel poll configure
`ibrd count`	Read data where `count` is the bytes to read
`ibrda count`	Read data asynchronously where `count` is the bytes to read
`ibrdf flname`	Read data to file where `flname` is pathname of file to read

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 6-2. Syntax for Board-Level NI-488 Functions in Win32 Interactive Control (Continued)

Syntax	Description
`ibrpp`	Conduct a parallel poll
`ibrsc v`	Request/release system control
`ibrsv v`	Request service
`ibsad v`	Change secondary address
`ibsic`	Send interface clear
`ibsre v`	Set/clear remote enable line
`ibstop`	Abort asynchronous operation
`ibtmo v`	Change/disable time limit
`ibwait mask`	Wait for selected event where `mask` is a hex or decimal integer or a list of mask bit mnemonics, such as `ibwait TIMO CMPL`
`ibwrt wrtbuf`	Write data
`ibwrta wrtbuf`	Write data asynchronously
`ibwrtf flname`	Write data from a file where `flname` is pathname of file to write

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 6-3. Syntax for NI-488.2 Routines in Win32 Interactive Control

Routine Syntax	Description
`AllSpoll addrlist`	Serial poll multiple devices
`DevClear address`	Clear a device
`DevClearList addrlist`	Clear multiple devices
`EnableLocal addrlist`	Enable local control
`EnableRemote addrlist`	Enable remote control
`FindLstn padlist limit`	Find all Listeners
`FindRQS addrlist`	Find device asserting SRQ
`PassControl address`	Pass control to a device
`PPoll`	Parallel poll devices
`PPollConfig address dataline lineSense`	Configure device for parallel poll
`PPollUnconfig addrlist`	Unconfigure device for parallel poll
`RcvRespMsg count termination`	Receive response message
`ReadStatusByte address`	Serial poll a device
`Receive address count termination`	Receive data from a device
`ReceiveSetup address`	Receive setup
`ResetSys addrlist`	Reset multiple devices
`Send address buffer eotmode`	Send data to a device
`SendCmds buffer`	Send command bytes
`SendDataBytes buffer eotmode`	Send data bytes
`SendIFC`	Send interface clear
`SendList addrlist buffer eotmode`	Send data to multiple devices

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 6-3. Syntax for NI-488.2 Routines in Win32 Interactive Control (Continued)

Routine Syntax	Description
`SendLLO`	Put devices in local lockout
`SendSetup addrlist`	Send setup
`SetRWLS addrlist`	Put devices in remote with lockout state
`TestSRQ`	Test for service request
`TestSys addrlist`	Cause multiple devices to perform self-tests
`Trigger address`	Trigger a device
`TriggerList addrlist`	Trigger multiple devices
`WaitSRQ`	Wait for service request

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 6-4. Auxiliary Functions in Win32 Interactive Control

Function	Description
`set udname`	Select active device or board where `udname` is the symbolic name of the new device or board (for example, `dev1` or `gpib0`). Call `ibfind` or `ibdev` initially to open each device or board.
`set 488.2 v`	Enter 488.2 mode for board `v`.
`help`	Display the Win32 Interactive Control utility online help.
`help option`	Display help information about `option`, where `option` is any NI-488, NI-488.2, or auxiliary call (for example, `help ibwrt` or `help set`).
`!`	Repeat previous function.
`-`	Turn OFF display.
`+`	Turn ON display.
`n * function`	Execute function `n` times where `function` represents the correct Win32 Interactive Control function syntax.
`n * !`	Execute previous function `n` times.
`$ filename`	Execute indirect file where `filename` is the pathname of a file that contains Win32 Interactive Control functions to be executed.
`buffer option`	Set type of display used for buffers. Valid options are `full`, `brief`, `ascii`, and `off`. Default is `full`.
`q`	Exit or quit.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Status Word

In the Win32 Interactive Control utility, all NI-488 functions (except `ibfind` and `ibdev`) and NI-488.2 routines return the status word `ibsta` in two forms: a hex value in square brackets and a list of mnemonics in parentheses. In the following example, the status word is on the second line, showing that the write operation completed successfully:

```
ud0: ibwrt "f2t3x"
[0100] (cmpl)
count: 5

ud0:
```

For more information about the status word, refer to Chapter 3, *Developing Your Application.*

Error Information

If an NI-488 function or NI-488.2 routine completes with an error, the Win32 Interactive Control utility displays the relevant error mnemonic. In the following example, an error condition EBUS has occurred during a data transfer:

```
w

ud0: ibwrt "f2t3x"
[8100] (err cmpl)
error: EBUS
count: 1

ud0:
```

In this example, the addressing command bytes could not be transmitted to the device. This indicates that either the device that `ud0` represents is powered off, or the GPIB cable is disconnected.

For a detailed list of the error codes and their meanings, refer to Chapter 4, *Debugging Your Application.*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Count Information

When an I/O function completes, the Win32 Interactive Control utility displays the actual number of bytes sent or received, regardless of the existence of an error condition.

If one of the addresses in an address list of an NI-488.2 routine is invalid, then the error is EARG and the Win32 Interactive Control utility displays the index of the invalid address as the count.

The count has a different meaning depending on which NI-488 function or NI-488.2 routine is called. For the correct interpretation of the count return, refer to the function descriptions in the *NI-488.2M Function Reference Manual for Win32* or the online help.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ในการทำโครงงานชิ้นนี้สำเร็จลุล่วงไปด้วยดีต้องขอขอบพระคุณ รศ.ดร. มนัส สังวรศิลป์ อาจารย์ที่ปรึกษา ที่ให้คำแนะนำในการทำโครงงาน ตลอดจนการแก้ปัญหาต่างๆที่เกิดขึ้น และขอขอบคุณเพื่อนๆที่เป็นกำลังใจและคำแนะนำต่างๆในการทำโครงงานนี้ ขอบคุณเจ้าหน้าที่ห้องคอมพิวเตอร์ภาคอิเล็กทรอนิกส์ที่ให้ยืมอุปกรณ์ในการทดลอง และพี่สาวที่ช่วยในการพิมพ์รายงานฉบับนี้ให้สำเร็จด้วยดี

สุดท้ายขอขอบคุณ คุณพ่อ คุณแม่ ที่ได้ให้การสนับสนุนการศึกษามาด้วยดีโดยตลอดมา

.....

(นาย สุรศักดิ์ ตันติตระการวัฒนา)

.....

(นาย เอกวิทย์ อินเขียวสาย)

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. กนก เจนจิระพงศ์เวช. GPIB (IEEE488) บัสอินเตอร์เฟสมาตรฐานและการใช้ วารสารเซมิ คอนดักเตอร์ (พฤษภาคม, สิงหาคม, กันยายน) หน้า 214 – 218,201-207 ,196-201.
2. นพดล มณีรัตน์. การออกแบบและสร้างระบบการส่งข้อมูลแบบขนานสำหรับการควบคุมเครื่องมือวัดผ่านระบบบัสมาตรฐาน วิทยานิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต, คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2540.
3. Eugene Fisher, C.W. Jensen PET/CBM and the IEEE488 Bus (GPIB), Berkley, Osborne McGrew-Hill, Inc. 1982

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้