



ชุดสาธิตการทำงานโรงไฟฟ้าพลังความร้อนร่วม  
(COMBINED CYCLE POWER PLANT SIMULATOR)



นายรณชัย	ทวานนท์	37013340
นายปกรณ์	สิทธิวงศ์	37013344
นายพรชัย	ทองวิสัย	37013347

วัน เดือน ปี.....	-1 ค.ค. 2531
เลขทะเบียน.....	038344
เลขเรียกหนังสือ.....	T. 99364 ๖1๐1 ๕

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์

COMBINED CYCLE POWER PLANT SIMULATOR

โดย

นายธนชัย ทวานนท์

นายปกรณ์ ลิทธิวงศ์

นายพรชัย ทองวิสัย

ภาควิชา

เทคนิคอุตสาหกรรม

อาจารย์ที่ปรึกษา

อาจารย์ ชวลิต เบญจางคประเสริฐ

อาจารย์ อรลาภ แสงอรุณ

คณะวิศวกรรมศาสตร์  
อนุมัติให้นับปริญญานิพนธ์  
ศาสตรบัณฑิต

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรม

คณะกรรมการสอบปริญญานิพนธ์

..... อาจารย์ที่ปรึกษา

( )

..... กรรมการ

( )

..... กรรมการ

( )

..... กรรมการ

( )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ ชุดสาริตการงานโรงไฟฟ้าพลังความร้อนร่วม

โดย นายธนชัย ทวานนท์

นายปรกรณ์ ลิทวิวงศ์

นายพรชัย ทองวิลัย

อาจารย์ที่ปรึกษา อาจารย์ชวลิต เบนญางคประเสริฐ

อาจารย์อรลาภ แสงอรุณ

ภาควิชา เทคนิคอุตสาหกรรม

สาขาวิชา เทคโนโลยีอิเล็กทรอนิกส์

ปีการศึกษา 2539

### บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ จัดทำขึ้นโดยมีวัตถุประสงค์เพื่อทำการทดลองชุดสาริตการงานโรงไฟฟ้าพลังความร้อนร่วม และศึกษาการเขียนโปรแกรมควบคุมการทำงานของชุดสาริตให้สามารถแสดงสภาวะการทำงานของโรงไฟฟ้าพลังความร้อนร่วมในช่วงสภาวะปกติ และช่วงสภาวะในการเกิดความผิดปกติของโรงไฟฟ้าพลังความร้อนร่วมได้ โดยการใช้โปรแกรม Turbo C ในการควบคุมการทำงานของชุดสาริตโรงไฟฟ้าพลังความร้อนร่วม รวมทั้งสามารถนำไปประยุกต์ใช้กับงานอื่นได้

เนื้อหาภายในจะกล่าวถึงการทำงานของชุดสาริตโรงไฟฟ้าพลังความร้อนร่วม ซึ่งประกอบด้วยแผงวงจรอินเตอร์เฟสแบบดิจิทัลและอนาล็อก โดยใช้ IC เบอร์ 8255 , PAL16L8 , AD7528 และ LMS403 เป็นตัวส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังชุดสาริตการงานโรงไฟฟ้าพลังความร้อนร่วม

**Project Report**                    **COMBINED CYCLE POWER PLANT SIMULATOR**  
**By**                                    **Mr.THANONTCHAI THAWANONT**  
   **Mr.PAKRON SITIWONG**  
   **Mr.PHONCHAI THONGWILAI**  
**Project Report Advisor**   **Mr.CHAWALIT BANJANGKAPRASERT**  
   **Miss ORNLARP SAINGAROON**  
**Department of**                    **Industrial Technology**  
**Academic Year**                    **1996**

---

**Abstract**

**The purpose of this project is the first experiment which control the Combined Cycle Power Plant Simulator and setted program software to control normal and fault operation of Combined Cycle Power Plant by used to program Turbo C language for work control and can be applied used to work another.**

**This paper is described about interface card by used IC # 8255 , PAL16L8 , AD7528 and LM3403 which to communicate for software and the Combined Cycle Power Plant Simulator.**

## กิติกรรมประกาศ

ปริญญาโทระดับนี้ สำเร็จลุล่วงไปได้ด้วยดี โดยได้รับคำแนะนำและคำปรึกษา ทางด้านข้อมูล จากแหล่งความรู้ ตลอดจนการให้ความช่วยเหลือเป็นอย่างดี จากอาจารย์ที่ ปรึกษาโครงการ รวมทั้งเพื่อนนักศึกษาทุกท่านที่ได้ให้ความช่วยเหลือทางด้านการจัดทำโครง งานและให้ข้อเสนอแนะ จนทำให้ปริญญาโทระดับนี้เสร็จสมบูรณ์ ผู้จัดทำขอขอบพระคุณ ทุกท่านเป็นอย่างสูง



นายธนาชัย  
นายปกรณ์  
นายพรชัย

ทวาทน์  
สิทธิวงศ์  
ทองวิสัย

# สารบัญ

หน้า

บทคัดย่อภาษาไทย .....	จ
บทคัดย่อภาษาอังกฤษ .....	ฉ
กิตติกรรมประกาศ .....	ช
บทที่ 1 ความมุ่งหมายของปริยฐานิพนธ์ .....	1
1.1 บทนำ .....	1
1.2 จุดประสงค์ของปริยฐานิพนธ์ .....	1
บทที่ 2 ภาษา C เบื้องต้น .....	2
2.1 จุดเริ่มต้นของภาษา C .....	2
2.2 ภาษา C ในไมโครคอมพิวเตอร์ .....	3
2.3 โครงสร้างของโปรแกรมภาษา C .....	4
2.4 การกำหนดตัวแปร .....	5
2.5 การกำหนดรูปแบบการพิมพ์ .....	6
2.6 ฟังก์ชัน print( ) และ scanf( ) .....	6
2.7 ชนิดของตัวแปร .....	7
2.8 ตัวโอเปอเรเตอร์ .....	7
2.9 การประมวลผลเบื้องต้น ( Preprocessor ) .....	8
บทที่ 3 หลักการ INTERFACE .....	10
3.1 IBM PC กับการ INTERFACE .....	10
3.2 AT Slot .....	10
3.3 I/O Port ใน IBM PC .....	11
3.4 อินเทอร์รัพท์ .....	13
3.5 DMA .....	14
3.6 รายละเอียดของสัญญาณต่างๆบน Slot .....	16
3.7 Power Supply .....	16
3.8 บัสและสัญญาณต่างๆ ที่เกี่ยวข้อง .....	16
3.9 สัญญาณในการ Decode Port .....	23
3.10 เทคนิคในการ Decode Port .....	23
3.11 วงจร Decode .....	23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.12 Input/Output Port .....	28
3.13 การทำงานของวงจร Decode .....	29
3.14 รายละเอียดเกี่ยวกับ 8255 .....	33
3.15 รายละเอียดการจัดเรียงขาของ .....	34
3.16 โหมด 0 : การทำงานเป็นพอร์ตอินพุท/เอาต์พุทพื้นฐาน .....	35
3.17 โหมด 1 : การทำงานเป็นพอร์ตอินพุท/เอาต์พุทแบบสไตรบ.....	38
3.18 โหมด 2 : การทำงานแบบบััสสองทิศทาง.....	42
บทที่ 4 รายละเอียดของชุดสาธิตโรงไฟฟ้าพลังความร้อนร่วม .....	47
4.1 หลักการทำงานของชุดสาธิตโรงไฟฟ้าพลังความร้อนร่วม .....	67
4.2 ลำดับขั้นตอนการทำงานของโรงไฟฟ้าพลังความร้อนร่วม .....	71
บทที่ 5 หลักการทำงานของวงจร INVERTER .....	72
5.1 วงจร VOLTAGE TO CURRENT INVERTER .....	72
5.2 วงจร CURRENT TO VOLTAGE INVERTER.....	74
บทที่ 6 รายละเอียดเกี่ยวกับ INTERFACE CAED.....	77
6.1 DIGITAL CARD .....	77
6.2 ANALOG CARD .....	79
6.3 ส่วนของวงจรที่ต่อกับชุดสาธิต .....	79
บทที่ 7 สรุปการวิจัยและข้อเสนอแนะ .....	83
โปรแกรมควบคุมการทำงานของชุดสาธิต .....	85
เอกสารอ้างอิง .....	129
ภาคผนวก ก Data Sheet of 8255.....	130
ภาคผนวก ข Data Sheet of AD7582 .....	149
ภาคผนวก ค Data Sheet of LM3403 .....	154
ภาคผนวก ง การวางอุปกรณ์และลายปริ้นท์ .....	163

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

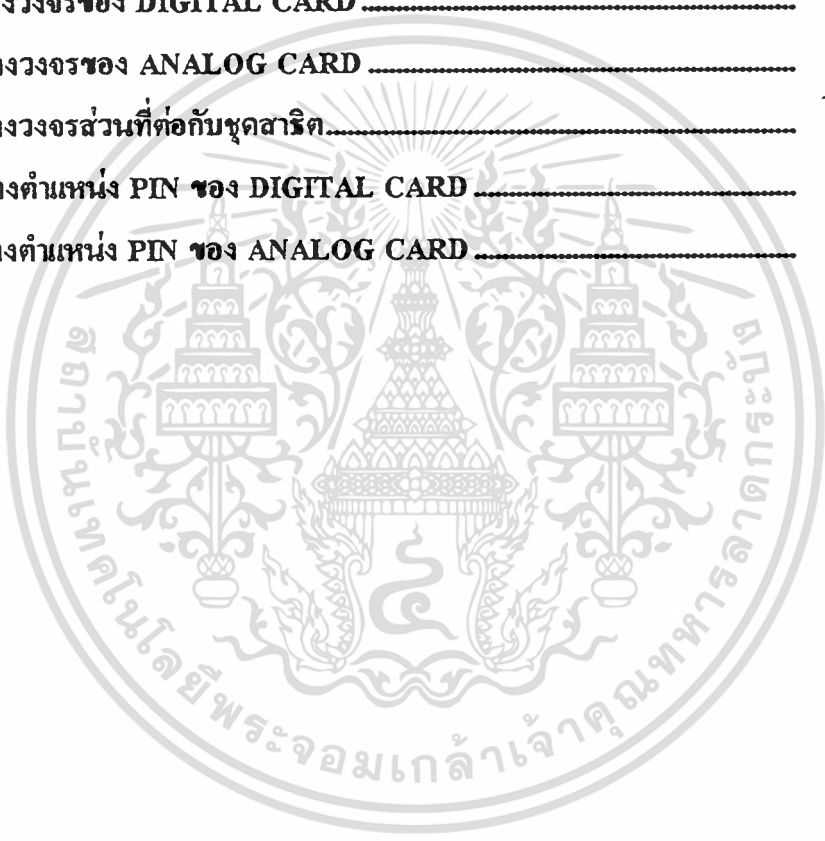
## สารบัญรูป

	หน้า
รูปที่ 3.1 โครงสร้าง Single Output Single Address.....	24
รูปที่ 3.2 วงจร Single Output Single Address .....	24
รูปที่ 3.3 โครงสร้าง Single Output Multiple Address .....	25
รูปที่ 3.4 วงจร Single Output Multiple Address.....	25
รูปที่ 3.5 โครงสร้าง Multiple Output Single Address .....	26
รูปที่ 3.6 วงจร Multiple Output Single Address .....	27
รูปที่ 3.7 โครงสร้าง Multiple Output Multiple Address .....	27
รูปที่ 3.8 วงจร Multiple Output Multiple Address .....	28
รูปที่ 3.9 วงจร Enable Circuit.....	30
รูปที่ 3.10 วงจร Output Latches Circuit.....	30
รูปที่ 3.11 การแสดงผลส่งข้อมูลออก .....	31
รูปที่ 3.12 วงจร Input Buffer Circuit .....	32
รูปที่ 3.13 การนำข้อมูลเข้าจากคิปสวิทช์ .....	32
รูปที่ 3.14 แสดงบล็อกไดอะแกรมของ 8255 .....	33
รูปที่ 3.15 แสดงรหัสควบคุมการทำงานของ 8255 .....	35
รูปที่ 3.16 ไดอะแกรมแสดงการทำงานของ 8255 ในโหมด 0.....	37
รูปที่ 3.17 แสดงลักษณะต่างๆของการใช้งาน 8255 ในโหมด 0 .....	38
รูปที่ 3.18 แสดงรหัสควบคุมและการทำงานในโหมด 1.....	40
รูปที่ 3.19 แสดงไทมิงสัญญาณต่างๆตามการทำงานในโหมด 1 (A) .....	40
รูปที่ 3.20 แสดงไทมิงสัญญาณต่างๆตามการทำงานในโหมด 1 (B) .....	41
รูปที่ 3.21 แสดงรูปแบบการทำงานในโหมด 1.....	42
รูปที่ 3.22 แสดงรหัสควบคุมและการทำงานในโหมด 1 .....	42
รูปที่ 3.23 แสดงรหัสควบคุมและไทมิงสัญญาณต่างๆ.....	44
รูปที่ 3.24 แสดงการทำงานโหมด 2 (พอร์ท A) ร่วมกับโหมด 0 (พอร์ท B).....	45
รูปที่ 3.25 แสดงการใช้งานไอซีชิป 8255 อินเตอร์เฟสกับอุปกรณ์ภายนอก.....	46
รูปที่ 3.26 แสดงรูป CONDENSERS .....	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.27 แสดงรูป CLOSED CIRCUIT .....	62
รูปที่ 3.28 แสดงรูป Water Flow an Open Recirculating Cooling Water System .....	63
รูปที่ 4.1 แสดงบล็อกไดอะแกรมของโรงไฟฟ้าพลังความร้อนร่วม .....	65
รูปที่ 4.2 แสดงโครงสร้างของโรงไฟฟ้าพลังความร้อนร่วม .....	66
รูปที่ 4.3 แสดงหลักการควบคุมชุดสาริตด้วยคอมพิวเตอร์ .....	67
รูปที่ 4.4 แสดง Flow Chart ในการควบคุมการทำงานของชุดสาริต.....	68
รูปที่ 5.1 แสดงวงจร VOLTAGE TO CURRENT INVERTER .....	72
รูปที่ 5.2 แสดงวงจร CURRENT TO VOLTAGE INVERTER.....	74
รูปที่ 6.1 แสดงวงจรของ DIGITAL CARD .....	78
รูปที่ 6.2 แสดงวงจรของ ANALOG CARD .....	80
รูปที่ 6.3 แสดงวงจรส่วนที่ต่อกับชุดสาริต.....	81
รูปที่ 6.4 แสดงตำแหน่ง PIN ของ DIGITAL CARD .....	82
รูปที่ 6.5 แสดงตำแหน่ง PIN ของ ANALOG CARD .....	82



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

	หน้า
ตารางที่ 3.1 แสดงการจัดแอดเดรสของหน่วยความจำบน IBM .....	11
ตารางที่ 3.2 แสดงการใช้งานของ Port บน IBM PC .....	12
ตารางที่ 3.3 แสดงลำดับการ Interrupt.....	13
ตารางที่ 3.4 แสดงการจัดช่อง DMA IBM.....	15



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 1

### ความมุ่งหมายของปริญญาโท

#### 1.1 บทนำ

โครงการปริญญาโทเรื่องชุดสาธิตการทำงานโรงไฟฟ้าพลังความร้อนร่วม เป็นการศึกษาหลักการควบคุมการทำงานของชุดสาธิต โดยการใช้โปรแกรมภาษาซีเป็นเครื่องมือในการเขียนโปรแกรม โดยเพื่อให้เป็นไปตามความมุ่งหมายของปริญญาโทจะเริ่มจากการศึกษา ถึงหลักการทำงานของโรงไฟฟ้าพลังความร้อนร่วม จากนั้นจึงทำการดำเนินการจัดทำชุดสาธิตและการเขียนโปรแกรมควบคุมการทำงาน โดยการนำความรู้จากการศึกษาระบบคอมพิวเตอร์ตลอดจนลักษณะการเขียนโปรแกรมมาประยุกต์ใช้ในการทำการทดลอง แก้ไข ปรับปรุง เพื่อให้สามารถควบคุมการทำงานของชุดสาธิตได้

#### 1.2 จุดประสงค์ของปริญญาโท

ปริญญาโทเรื่องชุดสาธิตการทำงานโรงไฟฟ้าพลังความร้อนร่วม เป็นการนำความรู้และข้อมูลหลาย ๆ ด้านมาประยุกต์ใช้เพื่อให้บรรลุผลตามจุดมุ่งหมาย ซึ่งมีจุดประสงค์ดังต่อไปนี้

- 1.2.1 เพื่อเป็นการศึกษาหลักการทำงาน ของโรงไฟฟ้าพลังความร้อนร่วม เนื่องจากในปัจจุบันโรงไฟฟ้าพลังความร้อนร่วมเป็นแหล่งผลิตพลังงานไฟฟ้าที่มีบทบาทสำคัญต่อระบบพลังงานไฟฟ้าในประเทศ การศึกษาหลักการทำงาน จากชุดสาธิต จึงเป็นพื้นฐานที่นำไปใช้ในการศึกษาหลักการการทำงานของโรงไฟฟ้าพลังความร้อนร่วมต่อไป
- 1.2.2 เพื่อเป็นการศึกษาโครงสร้าง และหลักการเขียนโปรแกรมภาษาซี
- 1.2.3 เพื่อเป็นการประยุกต์ใช้คอมพิวเตอร์ในการควบคุมชุดสาธิตการทำงานโรงไฟฟ้าพลังความร้อนร่วม โดยการนำความรู้จากการศึกษาระบบการเขียนโปรแกรมภาษาซี มาใช้ในการควบคุมการทำงานของชุดสาธิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ภาษาซีเบื้องต้น

#### 2.1 จุดเริ่มต้นของภาษาซี

ในปี พ.ศ. 2515 เดนนิส ริทช์ (Dennis Ritchie) ก็ได้พัฒนาภาษาซีขึ้นที่ห้องปฏิบัติการเบล ที่เมอร์ริลล์ มลรัฐนิวเจอร์ซีย์ โดยที่เดนนิส ริทช์ ได้นำเอาหลักการของภาษา B ที่พัฒนาโดย เคน ทอมสัน (Ken Thomson) โดยที่ทอมสันให้ชื่อเป็น BCPL (Basic Combined Programming Language) การพัฒนาของริทช์ จึงให้ชื่อภาษาว่า C เห็นจะมาจาก C ซึ่งเป็นตัวอักษรต่อจาก B และ C ก็เป็นตัวที่สองต่อจาก B ในคำ BCPL เช่นกัน

ภาษาซีและโปรแกรมจัดระบบงานยูนิกซ์ เป็นเสมือนของที่เกิดขึ้นร่วมกันในตอนแรกจาก การพัฒนาของทอมสัน และริทช์ ผู้ร่วมกันบุกเบิกยูนิกซ์จนในปัจจุบันยูนิกซ์ได้แพร่หลาย และพัฒนาไปอย่างมากมาย

ภาษาซี เป็นคอมไพเลอร์ภาษาที่มีอยู่บนเกือบทุกโปรแกรมจัดระบบงาน และมีบนเครื่อง ไมโครคอมพิวเตอร์ขนาดเล็ก 8 บิต ไปจนถึงมินิเมนเฟรม หรือแม้กระทั่งเครื่องคอมพิวเตอร์ซูเปอร์เมนเฟรมเช่น เคลย์วัน (Cray-1) ก็ยังมีภาษาซี

ภาษาซี เป็นภาษาคอมพิวเตอร์ที่มีการพัฒนาขึ้นใช้งาน เพื่อให้เป็นภาษามาตรฐานที่ไม่ขึ้นกับโปรแกรมจัดระบบงานหรือขึ้นกับฮาร์ดแวร์ จึงทำให้ภาษาซีเป็นภาษาคอมพิวเตอร์ตามอุดมการณ์ของนักคอมพิวเตอร์ ภาษาคอมพิวเตอร์อื่น ๆ เช่น ภาษาเบสิก มีมากมายหลายรุ่นหลายแบบ เช่น แอปเปิ้ลซอฟท์ ไมโครซอฟท์เบสิก BASICA และอื่น ๆ อีกมาก ภาษาเบสิกดังกล่าวนี้มีข้อแตกต่างกันและมีรายละเอียดไม่เหมือนกัน จึงทำให้โปรแกรมที่เขียนไม่สามารถใช้งานร่วมกันได้ ซึ่งเป็นข้อเสียของนักพัฒนาโปรแกรมอย่างยิ่ง ทำให้ต้องเสียเวลาทำการแก้ไขดัดแปลง เมื่อมีการนำมาใช้กับเครื่องคอมพิวเตอร์ต่างเครื่องกัน ภาษาซีเป็นภาษาที่พยายามเอาชนะในเรื่องเล็กน้อยเหล่านี้ ภาษาซีจึงเป็นภาษาที่ไม่ขึ้นกับเครื่องและฮาร์ดแวร์ ภาษาซีสามารถเขียนขึ้นมาใช้กับเครื่องคอมพิวเตอร์อะไรก็ได้ ที่แตกต่างกันจากไมโครคอมพิวเตอร์ จนถึงเมนเฟรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษาซี เป็นภาษาคอมพิวเตอร์ที่อาศัยหลักการทางวิธีการ โปรแกรมแบบสมัยใหม่ที่เรียกว่าโปรแกรมโครงสร้าง การออกแบบระบบซอฟต์แวร์จึงมีรูปแบบการออกแบบที่เป็นโมดูล และสามารถนำไปใช้ได้ง่าย ผู้พัฒนาโปรแกรมจึงรู้สึกว่าการใช้ภาษาซีเป็นเรื่องที่เหมาะสมในงานพัฒนา

ในการพัฒนาโปรแกรมจัดระบบงานหรือตัวแปรภาษา เรามักใช้ภาษาระดับต่ำ เช่น แอสเซมบลี ภาษาแอสเซมบลีให้ข้อดีในแง่ความเร็วของการทำงาน แต่ก็มีคามยุ่งยากในการเขียนโปรแกรม การแก้ไขดัดแปลงและการค้นหาที่ผิด แต่หากจะใช้ภาษาระดับสูงอื่นก็จะมี ความล่าช้าในการทำงาน ทำให้ยากต่อการใช้เป็นโปรแกรมจัดระบบงานได้ แต่เนื่องจากภาษาซี เป็นภาษาที่มีโครงสร้างแบบภาษาระดับสูง และสามารถใช้งานหรือทำงานได้รวดเร็วเหมือนภาษา ระดับต่ำ ภาษาซีจึงเป็นโปรแกรมที่น่าสนใจในการพัฒนา

ในการคอมไพล์ด้วยภาษาซี คอมไพเลอร์ของภาษาซีจะให้ประสิทธิภาพที่เหนือกว่าภาษาชั้นสูงอื่น ๆ คือภาษาซีให้ออบเจ็กต์ได้สั้นกว่า ภาษาซีให้ความเร็วในการทำงานเร็วกว่า ภาษาซีจึงเป็นภาษาที่มีประสิทธิภาพในการทำงานได้ดีกว่าหลายภาษาในระดับสูง

ภาษาซีมีความคล่องตัวที่จะได้รับการประยุกต์เข้ากับงานต่าง ๆ ได้อย่างดี เราสามารถนำภาษาซีมาใช้ในงานพัฒนาโปรแกรมต่าง ๆ ได้ เช่น โปรแกรมเวิร์ดโปรเซสซิ่ง โปรแกรมสเปรดชีต โปรแกรมสำเร็จรูปอื่นๆ ปัจจุบันบริษัทผู้พัฒนาโปรแกรมสำเร็จรูปมักใช้ ภาษาซีเป็นเครื่องมือในการพัฒนาโปรแกรม

จากรูปโครงที่ทันสมัยของภาษาซีจึงเป็นภาษาที่เหมาะสมสำหรับผู้สนใจที่จะศึกษา หลักการทางด้านวิธีการ โปรแกรมเป็นอย่างดี การศึกษาภาษาซีจะทำให้ผู้ศึกษาได้เรียนรู้รูปแบบ ของวิธีการเขียนโปรแกรมที่ถูกต้อง เป็นการฝึกนิสัยในเรื่องการเขียนโปรแกรมและก้าวหน้าต่อไปในอนาคตได้อย่างดี

## 2.2 ภาษาซีในไมโครคอมพิวเตอร์

เป็นที่คาดหวังกันว่าภาษาซีจะเป็นภาษาที่ได้รับความนิยมกันต่อไป ในไมโครคอมพิวเตอร์ปัจจุบันทั้งเครื่องไมโครคอมพิวเตอร์ 8 บิต และ 16 บิตส่วนแล้วแต่มีคอมไพเลอร์ ภาษาซีให้ใช้ทั้งสิ้น คอมไพเลอร์ที่มีชื่อเสียงในเครื่อง 8 บิต ได้แก่ Aztec CBDS C, C/80, Telecom C และ Whitesmiths C บิดความสามารถของภาษาซีดังกล่าวแตกต่างกันไปแล้วแต่ผู้พัฒนาได้สร้างขึ้นสำหรับในเครื่อง 6 บิต มีคอมไพเลอร์ที่มีชื่อเสียงหลายตัว เช่น Lattice หรือ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครซอฟท์ซี และซีคอมไพเลอร์ของบริษัทดิจิตอลรีเสิร์ช เป็นต้น ภาษาซีเหล่านี้อาจมีวิธีการใช้ที่แตกต่างกันบ้าง โดยเฉพาะวิธีการคอมไพล์ แต่หลักการโดยทั่วไปจะคล้ายคลึงกัน

### 2.3 โครงสร้างของโปรแกรมภาษาซี

ก่อนที่จะทำการเขียนโปรแกรมภาษาซี มาลองดูโปรแกรมภาษาซีกันก่อนว่า โครงสร้างหน้าตาของภาษาซีเป็นอย่างไร โปรแกรมภาษาซีเป็นโปรแกรมโครงสร้างเหมือนโปรแกรมภาษาอื่นที่เป็นโครงสร้าง เช่น ภาษาปาสคาล อัลกอท ทีแอลวัน

รูปแบบของโปรแกรมภาษาซี มีลักษณะเฉพาะตัวหลายอย่าง เช่น การใช้เครื่องหมายพิเศษ ดังตัวอย่างข้างล่างนี้

```
#include <stdio.h>

main ( ) /* a simple program */
{
    int num;
    num = 1;
    printf("I think ");
    printf("C is number %d of all the computer programmings.
\n"),num;
    printf("Do you think so?\n");
}
```

หากอธิบายตามโปรแกรมที่ยกตัวอย่าง ตามลำดับจะได้รายละเอียดดังนี้

#include <stdio.h> มีความหมายถึง การบอกให้ซีคอมไพเลอร์นำไฟล์อินพุทเอาต์พุทมาตรฐาน (standard I-O) ซึ่งกำหนดฟังก์ชันของการติดต่ออินพุทเอาต์พุทมารวมกับไฟล์นี้

main ( ) เป็นฟังก์ชันที่จะต้องมีในทุกโปรแกรม โดยเครื่องหมายวงเล็บจะไว้บอกค่าตัวแปรที่จะผ่านค่าเข้าออก ในที่นี้ไม่มี

เครื่องหมาย { } และ ( ) มีความหมายในเรื่องรูปโครงสร้างที่สำคัญ ภาษาซีใช้ { และ }

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดบล็อกของฟังก์ชัน โดยหลักการทั่วไป ภาษาซีจะมีฟังก์ชันภายในให้เรียกใช้จำกัด แต่สามารถเรียกจากไฟล์อื่นที่ `#include` ไปได้

ฟังก์ชันของภาษาซีจะมี ( ) ตามมา เช่น `main ( )` , `printf ( )` จะเห็นว่า ( ) คือ การแสดงว่ามีอาร์กิวเมนต์ หรือตัวผ่านค่าของฟังก์ชันหรือไม่ เช่น `main ( )` ซึ่งในวงเล็บไม่มีสัญลักษณ์ หมายถึงไม่มีตัวผ่านค่า แต่ `printf ( " I think " )` หมายความว่าข้อความ " I think " เป็นอาร์กิวเมนต์ที่จะผ่านค่าไปยังฟังก์ชัน `printf ( )`

```

/* ข้อความ */ หมายถึง การคอมเมนต์เหมือน REM ในภาษาเบสิก
{ หมายถึง การเริ่มต้นบล็อกของฟังก์ชัน
int num ; หมายถึง การกำหนดชนิดของตัวแปร (declaration) โดย
              กำหนด ตัวแปร num เป็นตัวแปรชนิดเลขจำนวนเต็ม
printf หมายถึง เป็นฟังก์ชันภายในให้พิมพ์ข้อความในวงเล็บ
\n หมายถึง รหัสควบคุมที่บอกภาษาซีให้ขึ้นบรรทัดใหม่
%d หมายถึง รูปแบบในการพิมพ์ตัวเลข
} หมายถึง การสิ้นสุดของบล็อก

```

บล็อกของโปรแกรมภาษาซี เริ่มต้นด้วยการกำหนดชื่อเป็น `main ( )` และใช้ ( ) เป็น การกำหนดบล็อกเหมือน BEGIN END ในภาษาปาสคาล บล็อกในภาษาซีจะใหญ่โตเพียงใดก็ได้ และสามารถกำหนดบล็อกซ้อนได้ โดยจะต้องมีเครื่องหมาย { ..... { ..... } ..... } และเครื่องหมาย { จะต้องเท่ากับ }

อนึ่ง ในที่นี้ไม่ได้กำหนดส่วนนำ เพราะโปรแกรมนี้อยังไม่มีการเรียกฟังก์ชันอื่นเข้ามาประกอบด้วย จึงไม่จำเป็น

จากโปรแกรมหกข้อกล่าวเมื่อทำการคอมไพล์ และลิงค์แล้วจะได้โปรแกรม `EXE` ซึ่งเป็นเอกซ์คิวทีฟไฟล์ ซึ่งเราสามารถเรียกให้ทำงานได้โดยตรง

## 2.4 การกำหนดตัวแปร

ชื่อของตัวแปรที่ใช้ในโปรแกรมคอมพิวเตอร์นั้น ควรจะได้สื่อความหมายในชื่อของมันเอง เช่น `age` , `account` , `asset` อย่างไรก็ตามการใช้ชื่อตัวแปรนี้อาจจะยาวเท่าไรก็ได้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ก็ต้องขึ้นด้วยตัวอักษรก่อน จะใช้เครื่องหมายทางคณิตศาสตร์เข้ามาในชื่อตัวแปรไม่ได้ เครื่องหมายที่จะใช้ได้แก่ \_ ( ชิดล่าง ) ซึ่งมีตัวอย่างของตัวแปรที่ใช้ได้และไม่ได้ ดังนี้

ชื่อตัวแปรที่ใช้ได้	ชื่อตัวแปรที่ใช้ไม่ได้
account	SZ**
Cat3	3cat
My_Data	My-Data
_Ok	don't

## 2.5 การกำหนดรูปแบบการพิมพ์

คำสั่ง `printf` ซึ่งมีรูปฟอร์มเมตเพื่อกำหนดการพิมพ์ การกำหนดฟอร์มเมตใน `printf` ดังนี้

<code>%d</code>	พิมพ์ด้วยตัวเลขฐานสิบ
<code>%o</code>	พิมพ์ด้วยตัวเลขฐานแปด
<code>%x</code>	พิมพ์ด้วยตัวเลขฐานสิบหก
<code>%u</code>	พิมพ์ด้วยตัวเลขที่ไม่คิดเครื่องหมาย
<code>%e</code>	พิมพ์ด้วยตัวเลขแบบวิทยาศาสตร์
<code>%f</code>	พิมพ์ด้วยตัวเลขมีจุดทศนิยม
<code>%g</code>	พิมพ์ด้วยตัวเลขแบบ <code>%e</code> หรือ <code>%f</code> โดยเลือกแบบที่สั้นที่สุด

## 2.6 ฟังก์ชัน `printf ( )` และ `scanf ( )`

โดยปกติส่วนของคำสั่งที่ติดต่อกับอินพุทของคอมพิวเตอร์ในภาษาซี ผู้เขียนคอมพิวเตอร์จะเว้นไว้ เพื่อว่าเมื่อมีการนำคอมพิวเตอร์ตัวนี้ไปใช้กับเครื่องอื่น ๆ ก็จะมีการเขียนฟังก์ชันทางอินพุทเอาท์พุทเพิ่มเติมได้ ทั้งนี้เพราะรูปแบบของอินพุทเอาท์พุทของคอมพิวเตอร์แต่ละเครื่องไม่เหมือนกัน จึงเป็นการสะดวกและทำให้ภาษาซีมีความคล่องตัวสูง

แต่อย่างไรก็ดี คอมพิวเตอร์ภาษาซีทุกตัวก็มีฟังก์ชันที่ใช้ติดต่อกับอินพุทเอาท์พุทที่สร้างไว้ภายในให้ใช้ได้เหมือนกัน ซึ่งก็ได้แก่ ฟังก์ชัน `printf ( )` และ `scanf ( )`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

`printf ( )` เป็นฟังก์ชันที่นำค่าจากโปรแกรมส่งมาให้ผู้ใช้ โดยผ่านทาง  
เอาต์พุตมาตรฐาน

`scanf ( )` เป็นฟังก์ชันที่รับค่าจากคีย์บอร์ด แล้วมาใส่ให้กับตัวแปรใน  
โปรแกรม

## 2.7 ชนิดของตัวแปร

ภาษาซีกำหนดตัวแปรออกเป็นชนิดต่าง ๆ โดยมีการกำหนดให้ตัวแปรเป็นชนิด  
อะไรเสียก่อน และในการใช้งานจะต้องเป็นไปตามชนิดของตัวแปรนั้น จะใช้เปลี่ยนไปเปลี่ยนมา  
ไม่ได้ ชนิดของตัวแปรที่ซีกำหนดไว้มีดังนี้

<code>int</code>	ชนิดเลขจำนวนเต็มปกติ
<code>short</code>	ชนิดเลขจำนวนปกติ
<code>long</code>	ชนิดเลขจำนวนเต็มที่มีความยาว 2 เท่า
<code>unsigned</code>	ชนิดของเลขที่ไม่คิดเครื่องหมาย
<code>char</code>	ชนิดตัวอักษร
<code>float</code>	ชนิดเลขมีจุดทศนิยม
<code>double</code>	ชนิดเลขที่มีจุดทศนิยมความยาวเป็น 2 เท่า

## 2.8 ตัวโอเปอเรเตอร์

ภาษาซีมีตัวโอเปอเรเตอร์มาก เมื่อเทียบกับภาษาคอมพิวเตอร์ภาษาอื่น ในที่นี้จะ  
ขอสรุปเป็นตาราง โดยเรียงลำดับความสำคัญของลำดับการทำงานจากบรรทัดบนลงล่างของ  
ตาราง และในแต่ละกลุ่มจะให้ระดับการทำงานจากขวามาซ้าย หรือจากซ้ายมาขวา ส่วนตัว  
โอเปอเรเตอร์ที่เกี่ยวข้องกับบิตจะทำให้รายละเอียดแยกออกไป

ตัวโอเปอเรเตอร์	การจัดการทำงาน
( ) { } ->	L - R
! ++ - -(TYPE) , * & size of	R - L
* / %	L - R
+ -	L - R
<< >>	L - R
< <= >=	L - R
= ! =	L - R
&	L - R
^	L - R
/ %	L - R
+ -	L - R
&&	L - R
	L - R
? :	L - R
= += -= *= /= %=	R - L
	L - R

## 2.9 การประมวลผลเบื้องต้น ( Preprocessor )

บางครั้งเราอาจกำหนดค่าให้กับตัวแปรก่อนการคอมไพล์ และเมื่อจะทำการคอมไพล์ก็  
จะมีการแทนค่าต่าง ๆ เข้าไป เช่น

เรากำหนดค่า PI ไว้ก่อน ให้มีค่าเป็น 3.1415926

เมื่อต้องการหาพื้นที่วงกลมจากสูตร  $PI * radius * radius$  ก็จะมีการนำเอาค่า PI  
เข้าไปแทนที่ ดังตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง

```
#define PI 3.1514

main( )
{
    :
    area = PI * radius * radius
    :
}
```

การกระทำการประมวลผลเบื้องต้นนี้เราจะใช้เครื่องหมาย # นำหน้าเพื่อบอกให้รู้ว่าสิ่งต่อไปนี่คือการกระทำแบบ preprocessor ซึ่งมีคำสั่งที่จะทำได้ในขั้นตอนนี้ ดังนี้

#### 2.9.1 #define

เป็นคำสั่งของการประมวลผลเบื้องต้นให้ทำการกำหนดค่าตัวอักษร เช่น

```
#DEFINE BEEP '\007'
#DEFINE NULL '\0'
#DEFINE SAY "Hello How are you! "
#DEFINE PI 3.1415926
#DEFINE BEGIN (
```

จะเห็นว่าภาษาซี #define นี้จะใช้อยู่ในเครื่องหมาย ‘ หรือ “ หรือไม่มีก็ได้

#### 2.9.2 #include

ในบางครั้งต้องมีการ #define เป็นจำนวนมาก เพื่อเรียกมาใช้ในหลายๆ โปรแกรม ภาษาซีได้เปิดโอกาสสำหรับผู้ที่ไม่อยากเขียน #define บ่อย ๆ ในหลาย ๆ โปรแกรม โดยนำเอา #define มารวมกันไว้เป็นไฟล์ ๆ หนึ่งแล้วตั้งชื่อไว้โดยมี .h เช่น stdio.h , conio.h เป็นต้น การรวมกันในลักษณะนี้ทำให้เราเรียกมาได้ด้วย #include

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 8

### หลักการ INTERFACE

#### 3.1 IBM PC กับ การ Interface

ในปัจจุบันนี้มีการนำ Computer ไปใช้งานกันอย่างกว้างขวาง ทั้งทางด้านเอกสาร , งานการทำเสียง Effect ใช้ในประกอบภาพยนตร์ หรือในด้านการทำโฆษณา ฯลฯ ในการทำงานเหล่านี้ ส่วนมากจะมีอุปกรณ์เป็นตัวต่อเชื่อมกับผู้ใช้งาน เช่น Joystick , Mouse , Light Penning ซึ่งอุปกรณ์เหล่านี้จะมีการใช้ Bus ติดต่อกับ RS 232 , IEEE 488 (GPIB) ทำให้สะดวกในการใช้งาน

แต่อย่างไรก็ตาม มาตรฐานเหล่านี้ยังมีข้อจำกัด ดังนั้นเมื่อต้องการจะนำคอมพิวเตอร์ไป Control งานบางอย่างจึงต้องทำการอินเทอร์รัพท์ I/O ที่ต้องการติดต่อโดยตรงกับระบบ Bus ของ IBM PC เพื่อเป็นการเพิ่มขีดความสามารถในการติดต่อ และเพิ่มประสิทธิภาพของอุปกรณ์ I/O ได้ดียิ่งขึ้น

#### 3.2 AT Slot

เนื่องจาก AT ใช้ CPU 80286 (หรือ 80386 , 80486 หรือสูงกว่านี้) ซึ่งเป็น CPU ขนาด 16 บิตแท้ มีการประมวลผลภายใน 10 บิต (CPU 80386 ประมวลผล 32 บิต) และมี Data Bus 10 บิต ในขณะที่ XT มีการประมวลผล 16 บิต แต่มี Data Bus 8 บิต จึงต้องทำการ Read ครั้งละ 8 บิต จำนวนถึง 2 ครั้ง ซึ่งจะเห็นว่า AT สามารถทำงานได้เร็วกว่า XT

บน Mainboard ของ AT จะมี Slot อยู่ 2 ส่วนคือ Slot สั้น และ Slot ยาว ซึ่ง Slot บน จะมีจำนวนสัญญาณ และตำแหน่งเหมือนกับ XT ส่วนสัญญาณที่ Slot ส่วนล่างที่เพิ่มขึ้น เป็นสัญญาณเฉพาะ AT เท่านั้น ซึ่งประกอบด้วย Data Bus ครึ่งบน จำนวน 8 บิต Add ที่เพิ่มขึ้นอีก 4 บิต และสัญญาณควบคุมที่เพิ่มขึ้น โดย Card XT สามารถนำมาใช้ได้กับ Slot บนของ AT แต่ที่ต้องคำนึงถึงคืออัตราการรับส่งข้อมูลบน Card นั้นด้วย เพราะความเร็ว AT จะเร็วกว่า XT เมื่อนำ Card XT มาใช้กับ AT จะทำให้การทำงานช้าลงได้ แต่ถ้า Card นั้นไม่ทำงานในส่วน of หน่วยความจำหรือการทำ DMA ก็ไม่ต้องไปสนใจกับอัตราการรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 แสดงการจัดแอดเดรสของหน่วยความจำ

3.3 I/O Port ใน IBM PC

Block 0	00000-0FFFF	RAM to 64K
Block 1	10000-1FFFF	RAM to 128K
Block 2	20000-2FFFF	RAM to 192K
Block 3	30000-3FFFF	RAM to 256K
Block 4	40000-4FFFF	RAM to 320K
Block 5	50000-5FFFF	RAM to 384K
Block 6	60000-6FFFF	RAM to 448K
Block 7	70000-7FFFF	RAM to 512K
Block 8	80000-8FFFF	RAM to 576K
Block 9	90000-9FFFF	RAM to 640K
Block A	A0000-AFFFF	Extended video memory
Block C	C0000-CFFFF	BIOS extension (eg EGA)
Block D	D0000-DFFFF	Other use
Block E	E0000-EFFFF	Other use
Block F	F0000-FFFFF	BIOSEPROM

การอ้างแอดเดรสของหน่วยความจำ จะใส่แอดเดรสทั้งหมด 20 เส้น คือ A0-A19 ในรุ่น XT (8088) แต่ในรุ่น AT (80286) ใช้แอดเดรส 24 เส้น คือ A0-A19 และ A20-A23 การตั้งแอดเดรสของ Port สำหรับ CPU เบอร์ 8088 และ 80286 สามารถอ้างได้ถึง 64k Port แต่ในรุ่นทั้ง IBM PC AT , XT ออกแบบให้ใช้แอดเดรสเพียง 10 เส้นเท่านั้นคือ A0-A9 ดังนั้นจำนวน Port ที่อ้างได้สูงสุดก็คือ 1024 Port และในจำนวนทั้งหมดนี้ยังแบ่งออกเป็น 2 กลุ่ม คือ กลุ่ม Port ที่มีแอดเดรสอยู่ในช่วง 00H-0FFH จะใช้งานบนเมนบอร์ดสำหรับ ชิพพอร์ทเท่านั้น เช่น 8259(Interrupt-Controller), 8237(DMA Cotrol), 8253(Time & Counter)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 แสดงการใช้งานของ Port บน IBM PC

Address	Description	Note	Remark
1F0-1F8	Fixed disk	1	Devices on main board
200-20F	Games adapter		not included
210-217	Expansion unit	2	1. AT only
278-27F	2 <sup>nd</sup> parallel printer port	1	2. PC only
2B0-2DF	Alternate EGA		3. 2 <sup>nd</sup> Bisynchronous on
2F8-2FF	2 <sup>nd</sup> serial port		At only
2EI	GPIB (o)	4	4. These devices decode
2E2-2E3	Data Acquisition (o)	4	the full 16 address bits
30031F	Prototype card		thereby allowing
320-32F	Fixed disk	2	further devices in the
360-36F	PC Network		same category above
378-37F	1 <sup>st</sup> parallel printer port		3FF (egg GPIB (1) =
380-38F	SDLC/2nd Bisynchronous	3	22e1 etc)
390-393	Cluster (o)	4	
3A0-3AF	1 <sup>st</sup> Bisynchronous	1	
3B0-3BF	Monochrome display / printer		
3C0-3CF	EGA		
3D0-3DF	CGA		
3F0-3F7	Floppy Disk		
3F8-3FF	1 <sup>st</sup> Serial Port		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งแอดเดรสจะถูกแบ่งออกเป็นช่วงย่อย ๆ ไว้ให้ใช้กับอุปกรณ์ I/O เฉพาะอย่าง แต่ถ้าในระบบไม่ได้ใช้งาน สามารถที่จะนำมาใช้งานทางด้านการขยาย Card ก็ได้ เช่น ถ้าไม่ใช้ Joystick เราสามารถใช้งาน Port ในช่วง 200 - 20FH ได้ แต่อย่างไรก็ตาม การใช้งานใน Port ที่ไม่ได้กำหนดอุปกรณ์ใช้จะดีกว่า

### 3.4 อินเทอร์รัพต์ (Interrupts)

การอินเทอร์รัพต์ใน CPU เบอร์ 80286 แบ่งเป็น 2 ชนิด คือ

#### 3.4.1 NMI (Non-Maskable Interrupts)

#### 3.4.2 INT (Maskable Interrupts)

แต่ใน IBM PC NMI ถูกใช้ในการเช็คข้อผิดพลาดของการรับส่งข้อมูล โดยการเช็คพาริตี (พาริตี คือ ข้อมูลบิตที่ 9 ใช้ในการทดสอบว่า ข้อมูลอ่านออกมาจากหน่วยความจำนั้น ถูกต้องหรือไม่) ถ้าพบความผิดพลาด CPU จะถูกอินเทอร์รัพต์แบบ NMI แล้วแสดงข้อความว่า "Parity Check Error" แล้วจะหยุดการทำงานทันที (Halt) ในอีกกรณีหนึ่งที่ทำให้เกิด NMI ได้ก็คือ เมื่อ CPU ได้รับสัญญาณ I/O CHECK จาก Slot การอินเทอร์รัพต์แบบ Maskable มีจำนวน 256 เวกเตอร์ ซึ่งในจำนวนทั้งหมด 256 เวกเตอร์นี้ จะรวมถึง Software อินเทอร์รัพต์ (BIOS Call) สัญญาณที่เกิดขึ้นจาก CPU เองด้วย

สำหรับทางด้าน Hardware แบ่งออกเป็น ดังนี้

- รุ่น XT แบ่งได้ 8 ระดับ
- รุ่น AT แบ่งได้ 16 ระดับ

ใน IBM PC การขอทำ Hardware อินเทอร์รัพต์ แบบ Maskable ไม่จำเป็นต้องใช้ค่าเวกเตอร์เพราะ 8259 (Interrupt Controller) จะเป็นตัวจัดการเองทั้งหมด เพียงแต่ใช้สัญญาณอินเทอร์รัพต์ผ่านทาง Slot เท่านั้น

ตารางที่ 3.3 แสดงลำดับการ Interrupt

ระดับความสำคัญ	หน้าที่
NMI	รับสัญญาณการตรวจสอบ Parity และการตรวจสอบ Channel I/O
IRQ 0	Timer Output 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับความสำคัญ	หน้าที่
IRQ 1	Keyboard (เมื่อ Buffer เต็ม)
IRQ 2	Interrupt มาจาก CTRL 2
IRQ 8	สัญญาณกำหนดเวลาเรียกมาจาก
IRQ 9	Software INT CAH หรือ IRQ 2 เดิม
IRQ 10	สงวนไว้
IRQ 11	
IRQ 12	
IRQ 13	อินเตอร์รัพต์แมตซ์โคร โปรเซสเซอร์
IRQ 14	อินเตอร์รัพต์ฮาร์ดดิสค์
IRQ 15	สงวนไว้
IRQ 3	อินเตอร์รัพต์พอร์ตสื่อสาร 2
IRQ 4	อินเตอร์รัพต์พอร์ตสื่อสาร 1
IRQ 5	อินเตอร์รัพต์พอร์ตขนาน 2
IRQ 6	ฟลอปปีดิสก์
IRQ 7	อินเตอร์รัพต์พอร์ตขนาน 1

### 3.5 DMA (Direct Memory Access)

DMA เป็นการส่งข้อมูลระหว่างหน่วยความจำกับอุปกรณ์อื่น ๆ โดยตรง โดยไม่ผ่าน CPU เพื่อลดเวลาในการรับส่งข้อมูลที่มีจำนวนมาก ๆ ด้วยวิธีนี้การรับส่งข้อมูลจากหน่วยความจำไปยังอุปกรณ์ I/O CPU จะอ่านข้อมูลจากหน่วยความจำมาเก็บไว้ในรีจิสเตอร์ก่อน แล้วจึงส่งข้อมูลจากรีจิสเตอร์ไปยังอุปกรณ์ I/O จะเห็นได้ว่า การรับส่งข้อมูลแบบนี้ มีการรับส่งถึง 2 ครั้ง แต่ก็เป็นการรับส่งแบบ DMA จะรับส่งข้อมูลระหว่างหน่วยความจำไปยังอุปกรณ์

I/O ได้โดยตรง โดยไม่ผ่าน CPU จะทำให้ลดเวลาลงได้มาก อุปกรณ์ที่ใช้ DMA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

♦ การวัดคุมฯ ดิสก์ (Floppy & Harddisk Controller)

การจัดลำดับ DMA

- ♦ IBM XT มี 4 ลำดับ คือ 0, 1, 2, 3
- ♦ IBM AT มี 7 ลำดับ คือ 0, 1, 2, 3, 4, 5, 6, 7

โดยลำดับ "0" มีความสำคัญมากที่สุด และใน AT ลำดับ 5, 6, 7 สามารถทำการ DMA ได้ครั้งละ 15 บิต โดย DMA Controller ใช้ IC#8237 ใน XT 1 ตัว และใน AT 2 ตัว ดังนั้นเราสามารถนำเอา DMA นี้ มาจัดลำดับความสำคัญของอุปกรณ์ I/O ได้

ตารางที่ 3.4 แสดงการจัดช่อง DMA IBM

DMA Channel		Function
DMA CONTROL 1	DMA CONTROL 2 (AT only)	
	4	Cascade for Control 1
0	}	Memory Refresh (AT-spare)
1		SDLC
2		Floppy Disk
3		Spare
	5	Spare
	6	Spare
	7	Spare

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.6 รายละเอียดของสัญญาณต่าง ๆ บน Slot

(VI) , (O) และ (I/O) หมายถึงทิศทางของขาสัญญาณเมื่อเทียบกับเมนบอร์ด โดยที่

(VII) หมายถึง ขาสัญญาณอินพุต

(O) หมายถึง ขาสัญญาณเอาต์พุต

(I/O) หมายถึง ขาสัญญาณที่เป็นได้ทั้งอินพุต และเอาต์พุต

(\* I/O) หมายถึง ในช่วงการทำงานปกติจะเป็นขาสัญญาณเอาต์พุต แต่จะเป็นอินพุตในช่วงที่เกิดขบวนการ DMA

สำหรับขาสัญญาณที่มีเครื่องหมายเลข (-) นำหน้า จะหมายถึง ขาสัญญาณที่แอกทีฟที่ลอจิก “0” และขาสัญญาณที่ไม่มีหรือมีเครื่องหมายบวก (+) นำอยู่หน้าจะหมายถึง ขาสัญญาณที่แอกทีฟที่ลอจิก “1” สัญญาณที่ต่ออยู่บน Slot สามารถขับ IC TTL ชนิด Low Power ได้สองตัว โดยไม่ทำให้เกิดการโหลดหรือการเพี้ยนของสัญญาณ ขาสัญญาณต่าง ๆ บน Slot ของ XT และ AT สามารถแบ่งออกเป็นกลุ่ม ๆ ได้ดังนี้

### 3.7 Power Supply

Ground	ขาสัญญาณนี้ต่ออยู่กับ Ground ของระบบเรกูเลเตอร์
+ 5 V	ขาสัญญาณนี้ต่ออยู่กับ ไฟ DC เรกูเลเตอร์ + 5 โวลต์
- 5 V	ขาสัญญาณนี้ต่ออยู่กับ ไฟ DC เรกูเลเตอร์ - 5 โวลต์
+ 12 V	ขาสัญญาณนี้ต่ออยู่กับ ไฟ DC เรกูเลเตอร์ + 12 โวลต์
- 12 V	ขาสัญญาณนี้ต่ออยู่กับ ไฟ DC เรกูเลเตอร์ - 12 โวลต์

### 3.8 บัส และสัญญาณต่าง ๆ ที่เกี่ยวข้อง

#### 3.8.1 แอดเดรสบัส

SA0-SA19 เป็นแอดเดรสบิตที่ 0 ถึง 19 โดยที่ SA0 มีนัยสำคัญต่ำที่สุด ขาสัญญาณนี้จะแอกทีฟ เมื่อขาสัญญาณ BALE มีสถานะเป็น “1” และจะถูกแลตช์ไว้ตอนขอบขาของสัญญาณ BALE แอดเดรสทั้ง 20 บิตนี้สามารถอ้างหน่วยความจำได้ถึง 20 เมกกะไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SBHE  (I/O)	(เฉพาะรุ่น AT) (Bus High Enable) เป็นขาสัญญาณที่ใช้แสดงว่า มีการรับส่งข้อมูลในบิตที่ SD8-SD15
LA17-LA23  (* I/O)	(เฉพาะรุ่น AT) ขาสัญญาณนี้จะแอกทีฟเมื่อขาสัญญาณ BALE มีสถานะเป็นลอจิก “1” แต่จะไม่มีกระแสไหลย้อนกลับของขาสัญญาณ BALE ดังนั้น ถ้าอุปกรณ์ I/O ไม่มีการอ้างแอดเดรสเกิน 1 เมกกะไบต์ ขาสัญญาณนี้ก็ไม่ใช่ ต้องใช้ แต่ถ้ามีการอ้างแอดเดรสเป็นอุปกรณ์ I/O จะต้องทำการแลตช์สัญญาณนี้ โดยใช้ขอบขาลงของขาสัญญาณ BALE ร่วมกับขาสัญญาณ MEMW และ MEMR
ABN  (0)	(Address Enable) ขาสัญญาณนี้จะแอกทีฟ เมื่อตัวควบคุม DMA ได้ทำการควบคุมบัสต่าง ๆ ของระบบแล้ว ดังนั้นการอ้างพอร์ตของอุปกรณ์ I/O จะต้องใช้สัญญาณนี้ ในการดีโค้ดด้วย เพื่อที่จะไม่ทำให้เกิดการติดต่อกันระหว่างระบบกับอุปกรณ์ I/O ตัวอื่น ยกเว้น ตัวที่กำลังทำขบวนการ DMA
BALE  (0)	(Address Latch Enable) ขาสัญญาณนี้ใช้ในการแสดงการเริ่มต้นของขบวนการต่าง ๆ ที่มีการติดต่อกับหน่วยความจำ โดยจะแอกทีฟเมื่อค่าแอดเดรสที่ CPU ต้องการติดต่อด้วยอยู่บนแอดเดรสบัสเรียบร้อยแล้ว ตามปกติขอบขาลงของสัญญาณนี้ จะทำให้เกิดการแลตช์สัญญาณ SA0-SA19 และถ้ามีการอ้างแอดเดรสเกิน 1 เมกกะไบต์ ใน AT จะใช้ขอบขาลงสัญญาณนี้ในการแลตช์สัญญาณ LA17-LA23 ด้วยเช่นกัน แต่สำหรับ ในขบวนการ DMA สัญญาณนี้ จะมีสถานะเป็น “1” ตลอด

### 3.8.2 คาต้าบัส

SD0-SD7  (I/O)	สำหรับรุ่น AT จะมี SD0-SD15 เพิ่มขึ้นมาด้วย คือ Data บิต 0 ถึง 7 สำหรับรุ่น XT และสำหรับรุ่น AT คือ Data บิต 0 ถึง 15 โดยที่ SD0 นัยสำคัญต่ำสุดสำหรับ AT ถ้ามีการติดต่อกับบิตที่ SD8-SD15 สามารถตรวจสอบได้จากขาสัญญาณ SBHE
----------------------	--

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.8.3 สัญญาณอินเทอร์รัพต์

**IRQ2-IRQ7** (Interrupt Request) (สำหรับรุ่น AT จะเป็น IRQ3-7 , 9-12 , 14 , 15) เป็นขาสัญญาณอินเทอร์รัพต์ CPU สำหรับ AT ลำดับความสำคัญของสัญญาณ IRQ เป็นดังนี้คือ 9 , 10 , 11 , 12 , 14 , 15 , 3 , 4 , 5 , 6 และ 7 โดย IRQ 7 มีลำดับความสำคัญน้อยที่สุดสำหรับ XT IRQ 2 จะมีลำดับความสำคัญมากที่สุดรองลงไป คือ IRQ 3 , 4 , 5 , 6 , 7 สำหรับรายละเอียดในการใช้งานของแต่ละอินเทอร์รัพต์ในหัวข้อการอินเทอร์รัพต์โดยปกติสัญญาณนี้จะมีสถานะเป็น “0” เสมอ ถ้าต้องการอินเทอร์รัพต์ CPU ให้ส่งพัลส์ที่เป็นลอจิก “1” ให้กับมันโดยไม่จำเป็นต้องคำนึงถึงคาบเวลาของพัลส์ ทั้งนี้เพราะระบบของ IBM ตัวอินเทอร์รัพต์ Controller (8259 Interrupt Controller) จะถูกโปรแกรมให้ทำการตรวจสอบสัญญาณอินเทอร์รัพต์โดยใช้ขอบขาลงของสัญญาณนี้

**I/O Check (I)** (I/O Channel Check) เป็นขาสัญญาณที่บอกถึงความผิดพลาดในการรับส่งข้อมูล ซึ่งตรวจสอบพาริตีบิต ถ้าพาริตีบิตที่อ่านจากหน่วยความจำกับพาริตีบิตที่สร้างขึ้นจากขบวนการรับส่งข้อมูล มีค่าไม่เท่ากัน แสดงว่า เกิดความผิดพลาดในการรับส่งข้อมูล สัญญาณนี้จะทำให้เกิดการอินเทอร์รัพต์ CPU แบบ NMI เพื่อบอกให้ CPU ทราบว่า เกิด Parity Error ขึ้น CPU จะแสดงข้อความบอกความผิดพลาดขึ้น และจะหยุดการทำงาน (Halt) เพื่อให้ผู้ใช้ตรวจสอบหาสาเหตุของการผิดพลาด

### 3.8.4 สัญญาณที่ใช้ในขบวนการ DMA

**DACK 0-3** (DMA Acknowledge) (สำหรับรุ่น AT จะเป็น DACK 0, 5-7) เป็นสัญญาณตอบสนองการขอทำ DMA ของอุปกรณ์ I/O เพื่อให้อุปกรณ์ I/O ทราบว่าการขอทำการ DMA นั้นได้รับการตอบสนองแล้ว เช่น ถ้ามีการขอทำ DMA ผ่านทาง DRQ 2 และเมื่อ CPU รับรู้แล้ว ทำให้สัญญาณ DACK 2 แอคทีฟไดนามิกแรมก็ตาม แต่สัญญาณ DACK 0 ก็จะถูกต่อออกมาที่ Slot ด้วย เพื่อแสดงถึงขบวนการรีเฟรชไดนามิกแรม และอุปกรณ์ I/O สามารถนำสัญญาณนี้ไปใช้ในการรีเฟรชหน่วยความจำแบบไดนามิกที่อยู่ในตัวมัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



DRQ 1 - DRQ 3 (DMA Request) (สำหรับรุ่น AT จะเห็น DRQ 0-3 , 5-4) เป็นขาสัญญาณใช้ในการขอทำขบวนการ DMA โดยที่ DRQ 0 มีลำดับความสำคัญมากที่สุด และ DRQ 3 มีลำดับความสำคัญน้อยที่สุด สำหรับรุ่น XT และสำหรับรุ่น AT ขา DRQ 7 จะมีลำดับความสำคัญน้อยที่สุด บน XT DRQ 0 ใช้สำหรับการรีเฟรชหน่วยความจำแบบไดนามิก จึงไม่มีขาสัญญาณ DRQ 0 ค่อยออกมาที่ Slot แต่สำหรับ AT แล้วจะมีวงจรโดยเฉพาะสำหรับใช้ในการรีเฟรชหน่วยความจำแบบไดนามิกอยู่แล้ว ดังนั้น ขา DRQ 0 เป็น DRQ 4 ก็ได้ เพื่อป้องกันความสับสนกับส่วนที่ใช้รีเฟรชไดนามิกแรม บน XT (เช่นเดียวกัน ก็จะเรียก DACK 0 เป็น DACK 4)

การขอทำ DMA ทำได้โดยทำให้ขาสัญญาณนี้มีสถานะเป็น "1" แล้วรอนกระทั่งได้รับตอบสนองการทำ DMA จาก CPU โดยการตรวจสอบสัญญาณ DACK ที่ส่งออกมา

Master

(1)

(เฉพาะรุ่น AT) (Master) ขาสัญญาณนี้จะใช้ร่วมกับ DMA Request ในการเข้าควบคุมระบบ Bus ในขบวนการ DMA โดยที่ตัว DMA คอนโทรลเลอร์จะส่งสัญญาณ DMA Request แล้วรอนกระทั่งได้รับการตอบสนอง โดยสัญญาณ DACK เกิดการแอกทีฟขึ้นแล้ว จึงจะส่งสัญญาณนี้ให้กับ CPU จะทำให้ Address Bus , Data Bus และ High Impedance หลังจากตัว DMA คอนโทรลเลอร์ จะต้องรออีกหนึ่งคาบสัญญาณคล็อก ก่อนที่จะเข้าควบคุม Bus ต่าง ๆ และจะต้องรออีก 2 ไชเคิล ก่อนที่จะทำการอ่านหรือเขียนข้อมูล ช่วงเวลาที่สัญญาณนี้แอกทีฟไม่ควรเกิน 15 ไมโครวินาที มิฉะนั้นข้อมูลภายในหน่วยความจำจะสูญหายไป เนื่องจากขาดสัญญาณรีเฟรชหน่วยความจำ

Refresh (๑)	(เฉพาะรุ่น AT) (Memory Refresh) หน้าที่เหมือนกับขาสัญญาณ DACK 0 ในรุ่น XT คือใช้แสดงขบวนการรีเฟรชหน่วยความจำ เพราะว่าในรุ่น AT จะมีวงจรที่ใช้ในการรีเฟรชหน่วยความจำโดยตรงอยู่แล้ว ดังนั้นจึงไม่จำเป็นต้องใช้ขาสัญญาณ DRQ0 และ DACK
T/C (๐)	(Terminal Count) เป็นขาสัญญาณที่บอกอุปกรณ์ I/O ที่ทำ DMA ให้ทราบว่า จำนวนข้อมูลที่รับส่งในขบวนการ DMA นี้ครบจำนวนแล้ว โดยจะส่งสัญญาณนี้เป็นพัลส์ให้กับอุปกรณ์

### 3.8.5 สัญญาณควบคุมต่าง ๆ

MEMR (* I/O)	(Memory Read) (สำหรับรุ่น AT คือ ขาสัญญาณ SMBMR (System Memory Read) ขาสัญญาณนี้จะเป็นตัวบอกให้หน่วยความจำส่งข้อมูลออกมาที่ Data Bus แต่สำหรับ AT สัญญาณ AMEMR จะแอกทีฟเมื่อเกิดการอ่านข้อมูล จากหน่วยความจำที่อยู่ภายใน 1 เมกกะไบต์แรกเท่านั้น
MEMR (๐)	(เฉพาะรุ่น AT) (Memory Read) สัญญาณนี้ไม่ใช่สัญญาณเดียวกันกับสัญญาณ MEMR ใน XT มันจะแอกทีฟในทุก ๆ ขบวนการอ่านข้อมูลที่เกิดขึ้น ไม่ว่าจะอยู่ในช่วงหน่วยความจำ 1 เมกกะไบต์แรกหรือไม่
MEMW (* I/O)	(Memory Write) สำหรับรุ่น AT คือ ขาสัญญาณ SMBMW (System Memory Write) ขาสัญญาณนี้จะเป็นตัวบอกให้หน่วยความจำเก็บข้อมูลจาก Data Bus แต่สำหรับ AT สัญญาณ SMBMW จะแอกทีฟเมื่อเกิดการเก็บข้อมูล จากหน่วยความจำอยู่ภายใน 1 เมกกะไบต์แรกเท่านั้น
MEMW (๐)	(เฉพาะรุ่น AT) Memory Write ขาสัญญาณนี้ไม่ใช่สัญญาณเดียวกันกับสัญญาณ MEMW ใน XT มันจะแอกทีฟในทุก ๆ ขบวนการเก็บข้อมูลที่เกิดขึ้น ไม่ว่าจะอยู่ในช่วงหน่วยความจำ 1 เมกกะไบต์แรกหรือไม่
IOR	(I/O Read) เป็นสัญญาณที่บอกให้อุปกรณ์ I/O ที่ต่ออยู่ทำการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(* I/O)	ลงมาที่ Data Bus
IOW (* I/O)	(I/O) Write) เป็นขาสัญญาณที่บอกให้อุปกรณ์ I/O ที่ต่ออยู่ ทำการเก็บข้อมูลจาก Data Bus เข้าไป
RESET DRV	(Reset Driver) เป็นขาสัญญาณที่แอกทีฟตอนช่วงที่เราเริ่มจ่ายไฟให้กับระบบ เพื่อใช้ในการรีเซต CPU และอุปกรณ์ต่าง ๆ ในระบบคอมพิวเตอร์ รวมทั้งอุปกรณ์ I/O ที่ต่ออยู่ด้วย
MEM CS16 (I)	(เฉพาะรุ่น AT) Memory 16 Chip Select เป็นขาสัญญาณที่ใช้บอกระบบให้ทราบว่า ต้องการรับส่งข้อมูลกับหน่วยความจำที่ละ 16 บิต ถ้าไม่ป้อนสัญญาณนี้ การรับส่งข้อมูลจะทำเหมือนกับ XT คือ ทำการรับส่งข้อมูลที่ละ 8 บิต สองครั้งเพื่อให้ได้ข้อมูลขนาด 16 บิต
I/O CS16	(เฉพาะรุ่น AT) Memory 16 Chip Select เป็นขาสัญญาณที่ใช้บอกระบบให้ทราบว่า ต้องการรับส่งข้อมูลกับอุปกรณ์ I/O ละ 16 บิต ถ้าไม่ป้อนสัญญาณนี้ การรับส่งข้อมูลจะทำเหมือนกับ XT คือ ทำการรับส่งข้อมูลที่ละ 8 บิต สองครั้ง เพื่อให้ได้ข้อมูลขนาด 16 บิต

### 3.8.6 สัญญาณนาฬิกา

CLK (O)	(System Clock) สำหรับ XT ขาสัญญาณนี้ จะมีความถี่ประมาณ 4.77 MHz หรืออาจจะสูงกว่านี้ก็ได้สำหรับรุ่นใหม่ ๆ และ สำหรับ AT จะมีความถี่ประมาณ 6 MHz หรือในรุ่นใหม่ ๆ อาจจะมีความถี่สูงถึง 15 MHz
------------	---

โดยปกติ ขาสัญญาณนี้มี Duty Cycle 50 % สำหรับ CPU เบอร์ 80286 ตัวกำหนดสัญญาณนาฬิกาที่ป้อนให้ จะมีความถี่เป็น 2 เท่าของความถี่ที่ CPU งาน แต่ขาสัญญาณนี้ก็ยังคงมีความถี่เท่ากับ ความถี่ที่ CPU ทำงานอยู่เสมอ

- OSC (Oscillator) เป็นขาสัญญาณที่มีความถี่สูงคือ 14.31818 MHz ความถี่ของ  
(O) สัญญาณนี้จะคงที่เสมอ และจะไม่ซิงโครนัสกับสัญญาณอื่น ๆ ในระบบ  
ดังนั้นจึงไม่ควรนำสัญญาณนี้ไปใช้เป็นสัญญาณคล็อกของอุปกรณ์ I/O  
ที่ต่ออยู่กับระบบ

### 3.8.7 สัญญาณที่ใช้สร้าง Wait States

- I/O CH RDY (I/O Channel Ready) ขาสัญญาณนี้จะถูกทำให้แอกทีฟโดยอุปกรณ์ I/O  
(I) หรือหน่วยความจำที่ไม่สามารถทำงานได้ทันทีที่กับระบบ ดังนั้นจะต้องทำ  
การหน่วยระบบ ให้ทำงานช้าลง ด้วยการเพิ่ม Wait States โดยการทำให้  
สัญญาณนี้ แอกทีฟในช่วงเวลาที่ I/O ได้รับสัญญาณจากการตีโค้ด แอก  
เดรส สัญญาณ MEMR, สัญญาณ MRMW, สัญญาณ IOR, สัญญาณ  
IOW
- OWS (เฉพาะรุ่น AT) Zero wait states การแอกทีฟของขาสัญญาณนี้จะบังคับ  
(I) ไม่ให้เกิดการสร้าง Wait States โดยอัตโนมัติ นั่นคือ การที่จะเกิด Wait  
States ขึ้นใน จะต้องขึ้นอยู่กับสัญญาณนี้ เช่น การทำงานในขบวนการ  
อ่านเขียนข้อมูล ขนาด 16 บิต โดยไม่ใช้ Wait States ทำได้โดยการ  
สร้างสัญญาณ OWS จากสัญญาณการตีโค้ดแอกเดรส และสัญญาณที่ใช้  
ในการอ่าน, เขียน หรือการลด Wait States ในขบวนการอ่านเขียนข้อมูล  
8 บิต ให้เหลือเพียง 2 Wait States ทำได้โดยให้สัญญาณ OWS แอกทีฟ  
หลังจากสัญญาณอ่าน หรือเขียนไปแล้ว 1 คล็อก โดยปกติการขับสัญญาณ  
นี้ ควรใช้เกตที่มีเข้าที่ทุกเป็นแบบ Open Collector ที่ทนกระแสได้ 20 mA  
(Sinking Current)

### 3.9 สัญญาณในการ Decode Port

- สัญญาณ Address
- สัญญาณ AEN เป็น “0” CPU จะติดต่อกับ I/O ถ้าเป็น “1” เกิดขบวนการ DMA
- สัญญาณ IOR ปกติจะเป็น “1” ซึ่งจะ Active ที่ Logic “0” เป็นการบอกว่า ขณะนี้ เป็นอ่านข้อมูลจาก Port Input
- สัญญาณ IOW ปกติจะเป็น “1” ซึ่งจะ Active ที่ Logic “0” เป็นการบอกว่า ขณะนี้ เป็นการเขียนข้อมูลไปยัง Port Output

### 3.10 เทคนิคในการ Decode Port

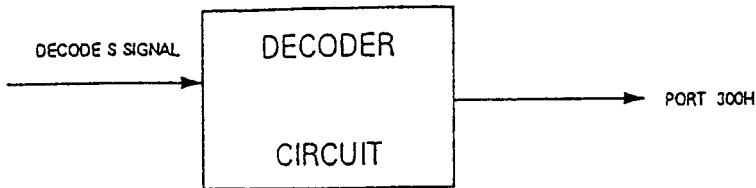
- ใช้ IC Logic ธรรมดา เช่น AND, OR, NOR, NAND และ INVERTER ในการ Decode ซึ่งจะใช้เมื่อต้องการ Output เพียง Output เดียว
- ใช้ IC Decode เบอร์ต่าง ๆ เช่น 74ALS138 , 74ALS139 มักใช้เมื่อต้องการ OUTPUT ขึ้นไป
- ใช้ IC เปรียบเทียบระหว่าง 2 Input เช่น Exclusive OR หรือ 74ALS688 โดยใช้สวิทช์เลือกเปรียบเทียบกับ Address ว่ามี Logic เหมือนกันหรือไม่ มักใช้เมื่อต้องการให้มีการแปรเปลี่ยน Address ได้
- ใช้ PROM (Programable Read Only Memory) มักใช้เมื่อ Output ที่เราต้องการให้มีการแยกกันอย่างอิสระ และต้องการ Program ก่อนนำมาใช้งาน

#### 3.1.1 วงจร Decoder

การจะพิจารณาเลือกเทคนิคใดมาใช้ในการ Decode จึงขึ้นอยู่กับ Output ที่เราต้องการมีลักษณะอย่างไร บางครั้งก็จำเป็นต้องนำแต่ละเทคนิคมารวมกัน ซึ่งส่วนที่ใช้งานทั่วไป จะเป็น 2 แบบแรก ส่วนการให้ PROM นั้นมีความยุ่งยากและใช้ในกรณีพิเศษเท่านั้น ดังนั้นการออกแบบวงจร Decode แบบต่าง ๆ เพื่อสร้าง Output ที่ต้องการ ซึ่งพอจะแบ่งลักษณะของ Output ได้ 4 ลักษณะดังนี้

3.1.1.1 Single Output Single Address เป็นลักษณะวงจร Decode ที่สร้าง Output ขึ้นมาเพียง Output เดียวและจะมี Address ติดต่อกับ Output นี้เพียง Address เดียวมีโครงสร้างดังรูปที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

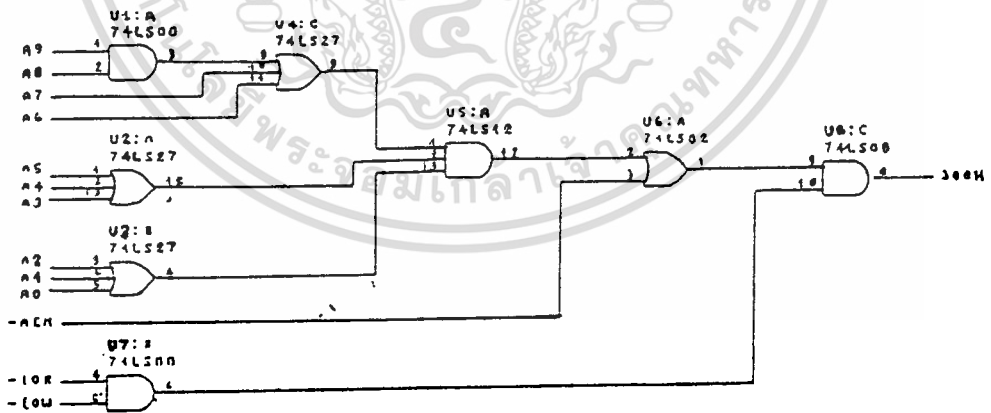


รูปที่ 3.1 โครงสร้าง Single Output Single Address

วงจรนี้ของเอาสัญญาณตั้งแต่ A0 จนถึง A9 มาเป็น Decode ทั้งหมดโดยกำหนดให้ Decode ออกที่ Port 300H ดังนั้น สัญญาณ A0 ถึง A9 ซึ่งมี Logic ดังนี้

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
1	1	0	0	0	0	0	0	0	0

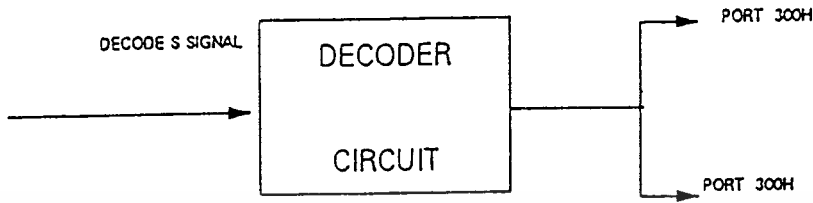
ซึ่งสามารถออกแบบวงจรได้ดังรูปที่ 3.2



รูปที่ 3.2 วงจร Single Output Single Address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11.2 Single Output Multiple Address เป็นลักษณะวงจร Decode ที่สร้าง Output ขึ้นมาเพียง Output เดียว แต่มี Address ที่จะติดต่อกับ Output ได้หลาย Address โดยมีโครงสร้างดังรูปที่ 3.3

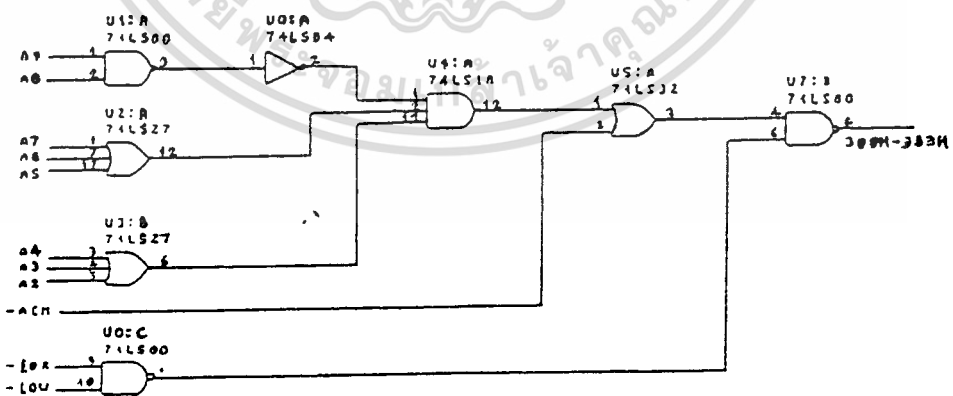


รูปที่ 3.3 โครงสร้าง Single Output Multiple Address

วงจรนี้เราไม่นำสัญญาณ Address มา Decode ทั้งหมด โดยจะปล่อย Address บางเส้นลอยไว้ จากโครงสร้างนี้กำหนดให้ Decode ออกที่ Output ในช่วง Port 300 H ถึง 303H ดังนั้นสัญญาณ Logic ของเส้น Address จะมีลักษณะดังนี้

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
1	1	0	0	0	0	0	0	X	X

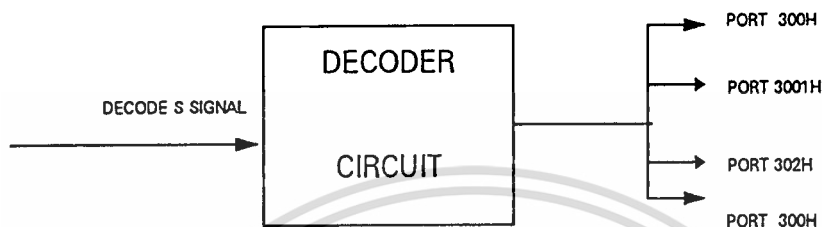
ซึ่งเราสามารถออกแบบวงจรได้ดังนี้



รูปที่ 3.4 วงจร Single Output Multiple Address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11.3 Multiple Output Single Address เป็นลักษณะวงจร Decode ที่สร้าง Output ขึ้นมาหลาย Output และมี Address ที่จะติดต่อกับ Output เพียง Output ละ 1 Address เท่านั้นมีโครงสร้างดังรูปที่ 3.5



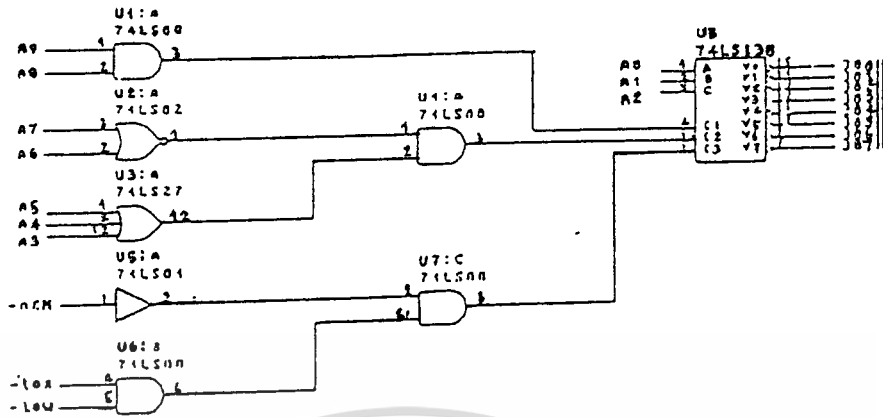
รูปที่ 3.5 โครงสร้าง Multiple Output Single Address

วงจรใน 2 แบบแรกซึ่งเป็นแบบ Output เดียว มักใช้ IC Logic Gate ขรรมดา ในการ Decode แต่ถ้าต้องการใช้งานมากกว่า 1 OUTPUT เช่น 8 Output จะใช้ IC Decoder เช่น เบอร์ 74ALS138 ช่วยในการ Decode โดยกำหนดไว้ 8 Port คือ Port 300H, 301H, .....,307H เมื่อนำสัญญาณ Address มาเขียน Logic ดู จะมีลักษณะดังนี้

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	
1	1	0	0	0	0	0	0	0	0	Port 300H
1	1	0	0	0	0	0	0	0	1	Port 301H
1	1	0	0	0	0	0	0	1	0	Port 302H
1	1	0	0	0	0	0	0	1	1	Port 303H
1	1	0	0	0	0	0	1	0	0	Port 304H
1	1	0	0	0	0	0	1	0	1	Port 305H
1	1	0	0	0	0	0	1	1	0	Port 306H
1	1	0	0	0	0	0	1	1	1	Port 307H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

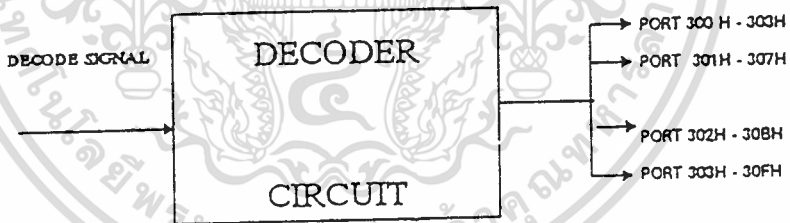
ซึ่งสามารถออกแบบวงจรได้ดังนี้



รูปที่ 3.6 วงจร Multiple Output Single Address

สำหรับ A2, A1, A0 จะปล่อยลอยไว้เพื่อ Variable Address ที่ 300H-307H

3.11.4 Multiple Output Multiple Address เป็นลักษณะของวงจร Decode ที่สร้าง Output ขึ้นมาหลาย Output แต่ละ Output มีหลาย Address ดังรูปที่ 3.7

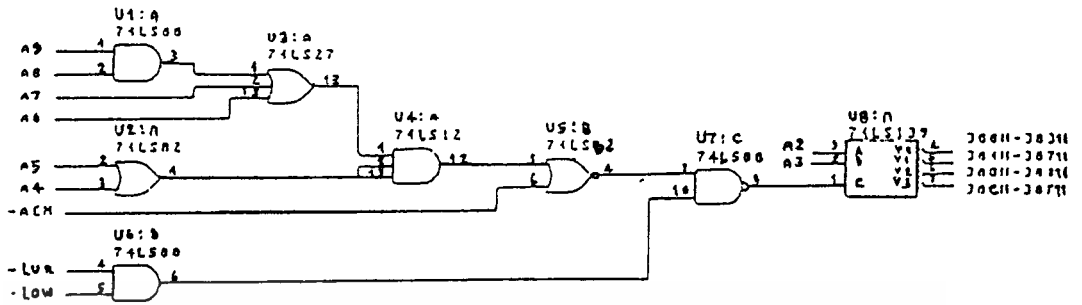


รูปที่ 3.7 โครงสร้าง Multiple Output Multiple Address

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	
1	1	0	0	0	0	0	0	X	X	Port 300H - 303H
1	1	0	0	0	0	0	1	X	X	Port 304H - 307H
1	1	0	0	0	0	1	0	X	X	Port 308H - 30BH
1	1	0	0	0	0	1	1	X	X	Port 30CH - 30FH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถออกแบบวงจรได้ดังรูปที่ 3.8



รูปที่ 3.8 วงจร Multiple Output Multiple Address

จากการ Decode ด้วย Gate ต่าง ๆ หอจะสรุปได้ว่า

1. ถ้าสัญญาณที่เข้ามาเป็น 0 จะใช้ OR หรือ NOR เป็นตัวรวม
2. ถ้าสัญญาณที่เข้ามาเป็น 1 จะใช้ AND หรือ NAND เป็นตัวรวม
3. Fixed Address ให้นำ Address ทุกเส้นเข้า Decode
4. Variable Address ให้นำ Address บางเส้นยกลอยไว้

### 3.12 Input/Output PORT

อุปกรณ์ที่ทำหน้าที่ในการรับและส่งข้อมูลนี้เรียกว่า พอร์ตอินพุท เอาท์พุท โดยความจริงแล้วการติดต่อกับ พอร์ตอินพุท เอาท์พุท ก็เหมือนกับการติดต่อกับหน่วยความจำ RAM แต่เหมือนเป็นการเขียนหรืออ่านในแอดเดรสหนึ่ง ซึ่งถ้าเป็นการเขียนก็คือ การส่งข้อมูลไปยัง เอาท์พุท แต่ถ้าเป็นการอ่านก็คือการรับข้อมูลเข้ามา การรับส่งข้อมูลจะเกิดขึ้นจากการทำงานร่วมกัน ระหว่าง ADDRESS ของอุปกรณ์ , สัญญาณ AEN , READ WRITE และ DATA BUS ซึ่ง จะทำการรับส่งข้อมูลก็ต่อเมื่อมีสัญญาณ CHIP SELECT ในช่วงเวลาดังกล่าว IC ที่เป็นเอาท์พุท พอร์ต จะมีคุณสมบัติในการตรวจจับ DATA จากบัส พร้อมทั้งเก็บค่า (LATCHE) ไว้ เป็นเวลานานตามต้องการ หรือจนกระทั่งมีข้อมูลใหม่ส่งเข้ามา ส่วน IC ที่เป็นอินพุทพอร์ตจะทำงานได้ 3 สถานะ ( 3 STATE ) ขณะที่ไม่มีสัญญาณออกที่เอาท์พุทคือไม่ถูกเลือกใช้งาน มันจะมีสัถยภาพ เป็น HIGH IMPEDANCE หรือสถานะของการเปิดวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

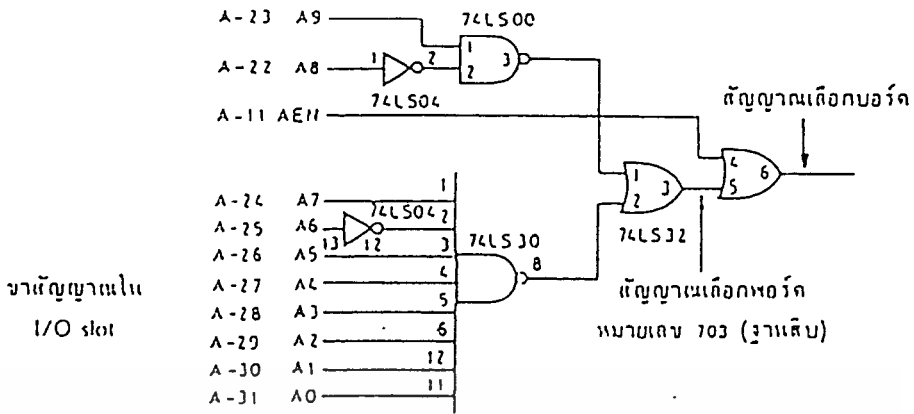
พอร์ตอินพุท เอาท์พุท ที่ใช้เกทลอจิก 3 สภาวะและแลทช์ฟลิปฟลอปมีข้อเสียในเรื่องจำนวน IC ซึ่งอาจต้องใช้หลายชิป (ถ้าต้องการหลายพอร์ต) และยากที่จะกำหนดลักษณะการทำงานให้แตกต่างกันไปจากวงจรเดิมที่ออกแบบไว้ ดังนั้นจึงได้มีการออกแบบ IC ที่สามารถโปรแกรมได้โดยการส่งรหัสควบคุม (CONTROL WORD) เข้าไปในรีจิสเตอร์ควบคุม ทำให้สามารถกำหนดพอร์ตเป็นอินพุทหรือเอาท์พุท มีการตรวจสอบ HAND SHAKING และสัญญาณอินเตอร์รัพท์เพียงแต่เปลี่ยนแปลงรหัสควบคุมเท่านั้น โดยไม่ต้องมีการเปลี่ยนแปลงทางด้านฮาร์ดแวร์ ดังนั้นจึงสรุปได้ว่า I/O Port จึงมีได้ 2 ชนิด คือ Simple I/O และ Programmable I/O

### 3.13 การทำงานของวงจร Decoder

สามารถแบ่งเป็นส่วน ๆ ได้ดังนี้

3.13.1 วงจร Enable Circuit การสื่อสารภายในคอมพิวเตอร์มีสายข้อมูล (Data Bus) เพื่อส่งและรับข้อมูลถึงกัน การส่งและรับข้อมูลถ้าไม่มีตำแหน่งในการส่งและรับก็ไม่ว่างส่งไปที่ไหนและให้ที่ใครรับ จึงจำเป็นต้องมีสายแอดเดรส (Address Bus) ในการกำหนดตำแหน่งภายในคอมพิวเตอร์ เช่นเดียวกับเลขที่ห้องเพื่อจะรับรู้กันภายในคอมพิวเตอร์ว่าต้องส่งข้อมูลดังกล่าวจากเลขที่ห้องไหนไปสู่ห้องไหน

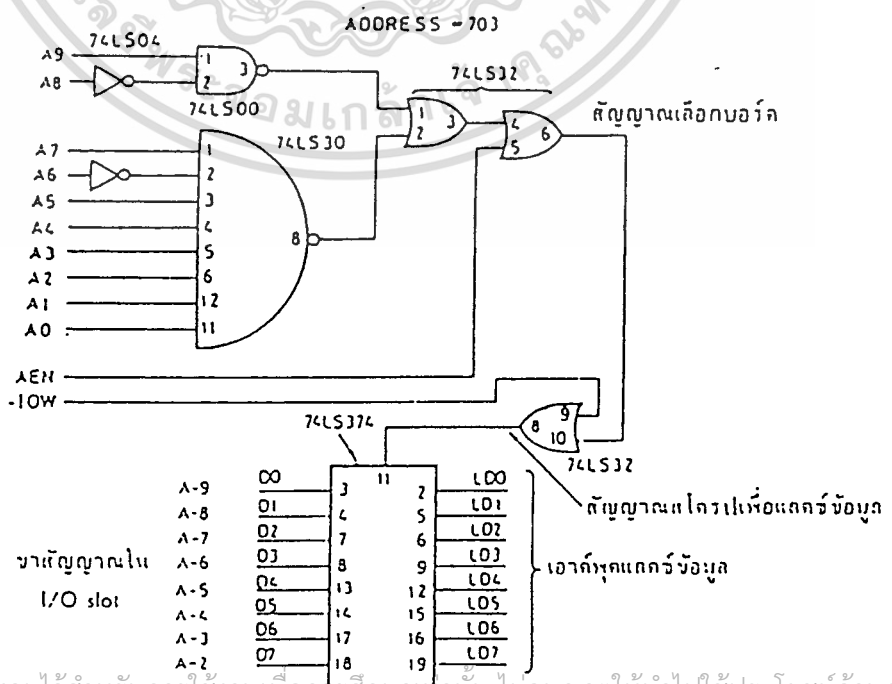
ในกรณีที่มีการสื่อสารกับอุปกรณ์ ก็ต้องมีการถอดรหัสแอดเดรสให้หมายเลขแก่อุปกรณ์ภายนอกนั้น การถอดรหัสแอดเดรสต้องไม่มีการซ้ำซ้อนกับแอดเดรสที่ใช้สื่อสารกับอุปกรณ์ภายนอกอื่น ๆ การถอดรหัสแอดเดรสเพื่อการเลือกพอร์ต (Port Select) ได้ใช้สาย แอดเดรส A0-A9 มีค่าเท่ากับ 1010111111 (พอร์ตหมายเลข 703 เมื่อเป็นเลขฐานสิบ) จึงทำให้สัญญาณการเลือกพอร์ตมีลอจิก 0 แต่การที่จะส่งออกหรือรับข้อมูลจากสายข้อมูลได้ต้องมีการต่อกับสาย AEN (Address Enable) ของ IBM PC ด้วย เพราะไม่เช่นนั้น การส่งหรือรับข้อมูลจากสายข้อมูลจะเกิดการผิดพลาดได้ สัญญาณที่ได้จากการเลือกพอร์ตนำมาออร์ (OR) กับ AEN จะใช้ในการอินเวิลวงจรส่งหรือรับข้อมูล จึงทำให้มีการส่งข้อมูลหรือรับข้อมูลเกิดขึ้น รายละเอียดของวงจรแสดงดังรูปที่ 3.9



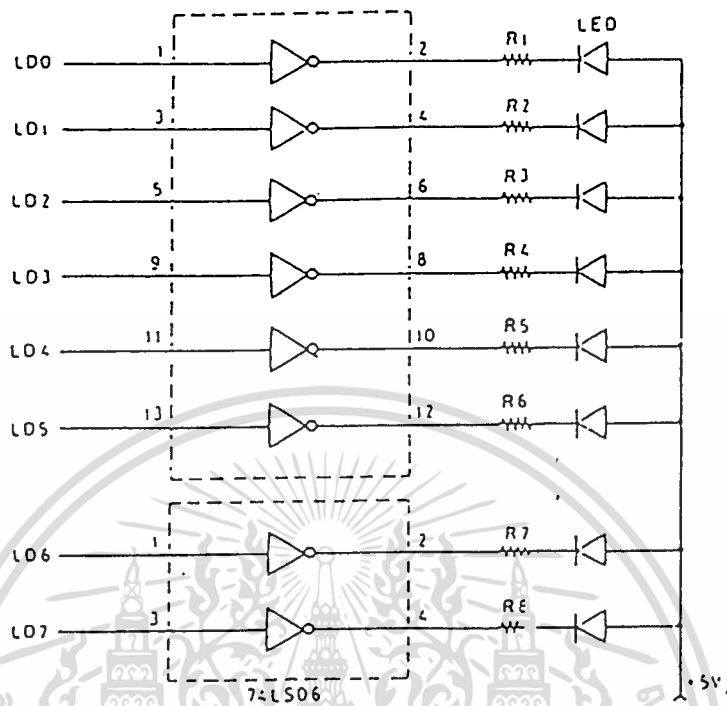
รูปที่ 3.9 วงจร Enable Circuit

3.13.2 วงจร Output Latches Circuit การส่งข้อมูลออกใช้ IC เบอร์ 74LS374 เป็นตัวแลทช์ข้อมูล การส่งข้อมูลออกต้องผ่านทางสายข้อมูล D0 - D7 และต้องทำการอินาเบิลเพื่อทำการแลทช์ข้อมูล ซึ่งยังไม่เพียงพอที่วงจรจะทำงานได้ ต้องมีสายสัญญาณอีกสายหนึ่งเข้าเกี่ยวข้องกับคือ -IOW (Input/Output Write) สายสัญญาณนี้มีลอจิก 0 เมื่อมีการใช้คำสั่ง OUT address , data ในภาษาเบสิก เพื่อทำการส่งข้อมูลออก ฉะนั้นจำเป็นต้องต่อเชื่อมสายสัญญาณดังกล่าวไว้ในวงจรอินาเบิลเพื่อการแลทช์ข้อมูลในวงจรส่งข้อมูลออก รายละเอียดของวงจรดังแสดงในรูปที่ 3.10

การแสดงผลส่งข้อมูลออกสามารถใช้ LED 8 ดวง เป็นตัวแสดงผลของข้อมูล 8 บิต แสดงการต่อดังรูปที่ 3.11



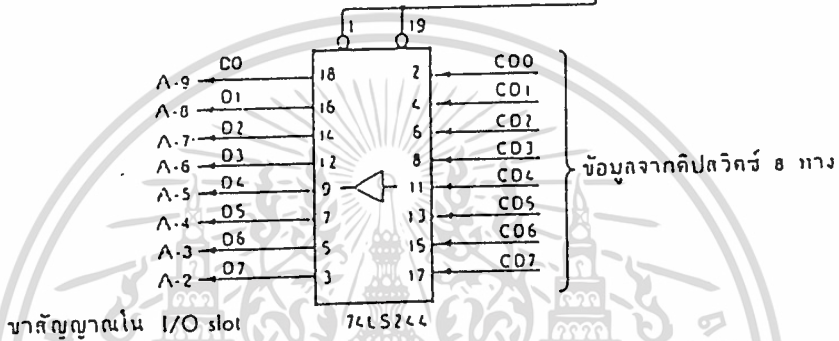
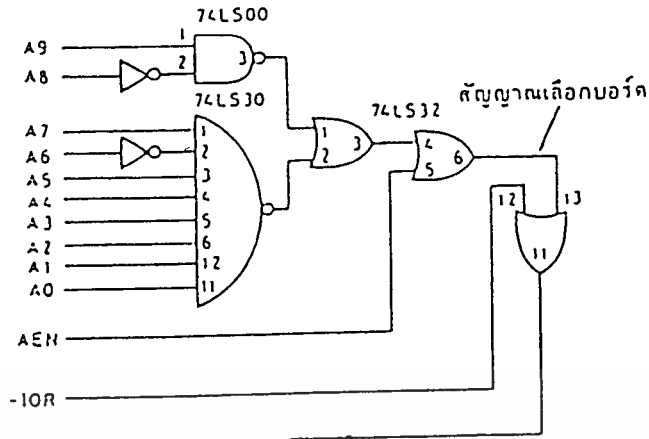
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงรูปที่ 3.10 วงจร Output Latches Circuit สารทุกครั้งที่มีการนำไปใช้



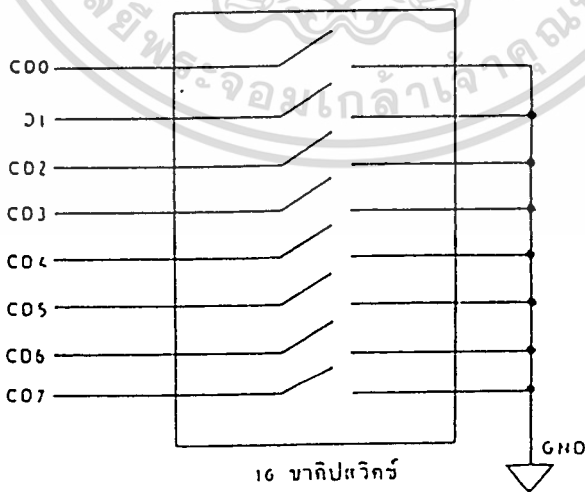
รูปที่ 3.11 การแสดงผลส่งข้อมูลออก

3.13.3 วงจร Input Buffer Circuit การนำข้อมูลเข้าใช้ IC 74LS244 เป็นบัฟเฟอร์ของข้อมูลการนำข้อมูลเข้าก็ต้องการอินเอาเบิล เพื่อนำข้อมูลเข้าทางสายข้อมูล อย่งไรก็ตามต้องมีสายสัญญาณอีกสายหนึ่งเข้ามาเกี่ยวข้องกับสายดังกล่าวคือ -IOR (Input/Output Read) สายสัญญาณนี้มีลอจิก 0 เมื่อมีการใช้คำสั่ง  $A = INP$  (address) ในภาษาเบสิก ฉะนั้นจำเป็นต้องเชื่อมต่อสายสัญญาณนี้เข้ากับวงจรอินเอาเบิล รายละเอียดของวงจรแสดงดังรูปที่ 3.12

จากรูปที่ 3.12 ที่ตำแหน่ง CD0 - CD7 ทำการเชื่อมต่อกับคิปลิวท์ 8 ทาง (16 ขา) เพื่อทำการสับสวิทช์ ON-OFF เป็นข้อมูลเข้าของแต่ละบิต (D0-D7) แสดงการต่อดังรูปที่ 3.13 คือเมื่อสวิทช์เปิด (ON) ข้อมูลที่เข้าสู่ IC 74LS244 มีลอจิกเป็น 1 (เพราะเป็น IC TTL : Transister Transister Logic) แต่เมื่อสวิทช์ปิด (OFF) ข้อมูลที่เข้าจะเป็น 0 ตามลำดับ



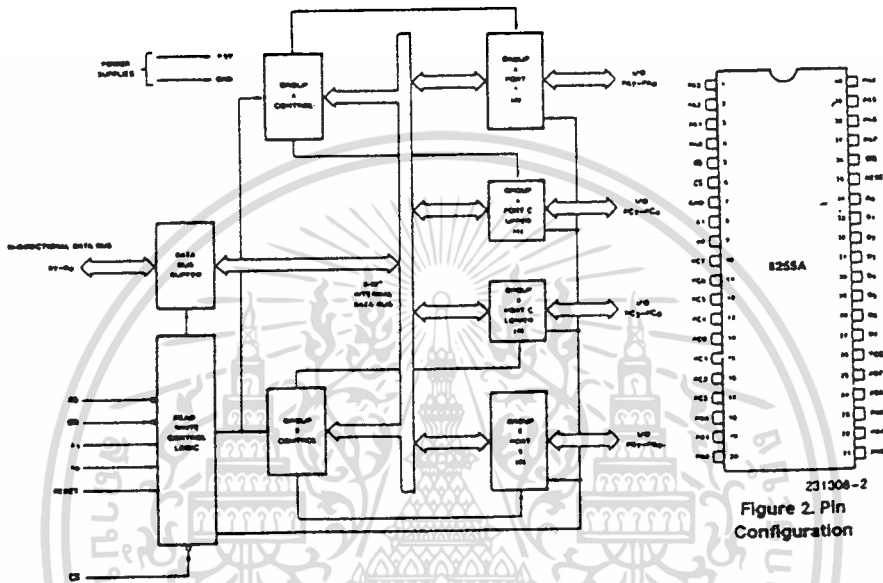
รูปที่ 3.12 วงจร Input Buffer Circuit



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเฉพาะเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.14 รายละเอียดเกี่ยวกับ 8255

8255 เป็นอุปกรณ์ LSI (Large Scale Intergrated Circuit) บรรจุอยู่ใน Package 40 ขาแบบ DIP (Dual In Line Package) เริ่มผลิตโดยบริษัท INTEL Corporation ผู้ผลิตไมโครโปรเซสเซอร์เบอร์ 8080 จุดประสงค์เพื่อใช้งานร่วมกับ 8080 โดยเฉพาะ แต่ในภายหลังได้มีการนำ 8255 ไปประยุกต์ใช้งานร่วมกับไมโครโปรเซสเซอร์เบอร์อื่น ๆ



รูปที่ 3.14 แสดงบล็อกไดอะแกรมของ 8255

รูปที่ 3.14 นี้แสดงบล็อกไดอะแกรมของ 8255 ซึ่งหน้าที่ของแต่ละบล็อกมีดังต่อไปนี้ คือ บล็อกกลุ่มแรกที่เราจะพูดถึงนี้ ได้แก่ บล็อกจำนวน 4 บล็อก ที่อยู่ทางด้านขวาของรูป ซึ่งจะเป็นส่วนที่เชื่อมต่อกับอุปกรณ์ภายนอกอื่น ๆ โดยมีสาย PA0-PA7, PB0-PB7 และ PC0-PC7 เป็นทางผ่านของข้อมูลระหว่างอุปกรณ์ภายนอกกับ 8255 สายสัญญาณเหล่านี้จะถูกแบ่งออกเป็น 0 I/O พอร์ต ได้แก่ พอร์ต A (PA), พอร์ต B (PB) และพอร์ต C (PC) พอร์ตเหล่านี้แต่ละพอร์ตสามารถเป็นได้ทั้งพอร์ตอินพุตและเอาต์พุต แต่ละบล็อกจะมีสายสัญญาณเชื่อมเข้ากับบัสข้อมูลภายในของ 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บล็อกรหัสควบคุมได้แก่ Group A Control และ Group B Control ซึ่งจะเป็นตัวกำหนดลักษณะการทำงานของทั้ง 3 I/O พอร์ต (8255 มีลักษณะการทำงานที่แตกต่างกันอยู่ 3 โหมดสามารถกำหนดได้โดยการโปรแกรมส่ง Control Word ให้กับ 8255 ซึ่งจะกล่าวถึงในภายหลัง) จากรูปที่ 4.1 จะเห็นว่า พอร์ต C นี้จะประกอบด้วยพอร์ตขนาด 4 บิต 2 พอร์ต กลุ่มหนึ่งจะถูกควบคุมโดย Group A Control และอีกกลุ่มหนึ่งจะถูกควบคุมโดย Group B Control สำหรับเหตุผลนั้นจะกล่าวถึงในภายหลัง

บล็อกรหัสสุดท้ายที่ได้แก่ Data Bus Buffer และ Read/Write Control Logic ซึ่งบล็อกเหล่านี้จะเป็นส่วนที่ติดต่อกับ CPU Data Bus Buffer นี้จะเป็นบัฟเฟอร์ให้กับบัสข้อมูลของ CPU ส่วน Read/Write Control Logic จะเป็นส่วนที่ควบคุมให้ข้อมูลเข้าหรือออกจากรีจิสเตอร์ภายในตัวที่ถูกต้อง และในเวลาที่เหมาะสม

### 3.15 รายละเอียดการจัดเรียงขาของ 8255

ในส่วนนี้เราจะพิจารณาหน้าที่ของขาแต่ละขา 8255 ซึ่งข้อมูลเหล่านี้จะมีประโยชน์ในการเชื่อมต่อเข้ากับระบบบัสของ CPU สำหรับการจัดขาแสดงไว้ใน รูปที่ 4.1 รายละเอียดของขาแต่ละขามีดังนี้คือ

**D0-D7 :** เป็นสายข้อมูลอินพุทเอาต์พุทแบบสองทิศทาง (Bi-Directional Bus) จะเป็นทางผ่านของข้อมูลระหว่างพอร์ตต่าง ๆ ของ 8255 กับบัสข้อมูลของ Z-80

**$\overline{CS}$  (Chip Select Input) :** เมื่อขานี้มีสถานะลอจิกเป็น "0" CPU จะสามารถที่จะอ่านหรือเขียนข้อมูลกับ 8255 ได้

**$\overline{RD}$  (Read Input) :** เมื่อขานี้มีสถานะลอจิกเป็น "0" สัญญาณ  $\overline{CS}$  ในลอจิกเป็น "0" และสัญญาณ  $\overline{CS}$  มีลอจิกเป็น "0" ข้อมูลจาก 8255 จะปรากฏส่งระบบบัสข้อมูล CPU ก็จะสามารถอ่านข้อมูลออกไปได้ (ในการตั้งชื่อของขาสัญญาณนี้จะถือเอา CPU เป็นหลัก)

**$\overline{WR}$  (Write Input) :** เมื่อขานี้มีสถานะลอจิกเป็น "0" และสัญญาณ  $\overline{CS}$  มีลอจิกเป็น "0" ข้อมูลจากระบบบัสข้อมูลจะถูกเขียนเข้าไปยัง 8255 ได้

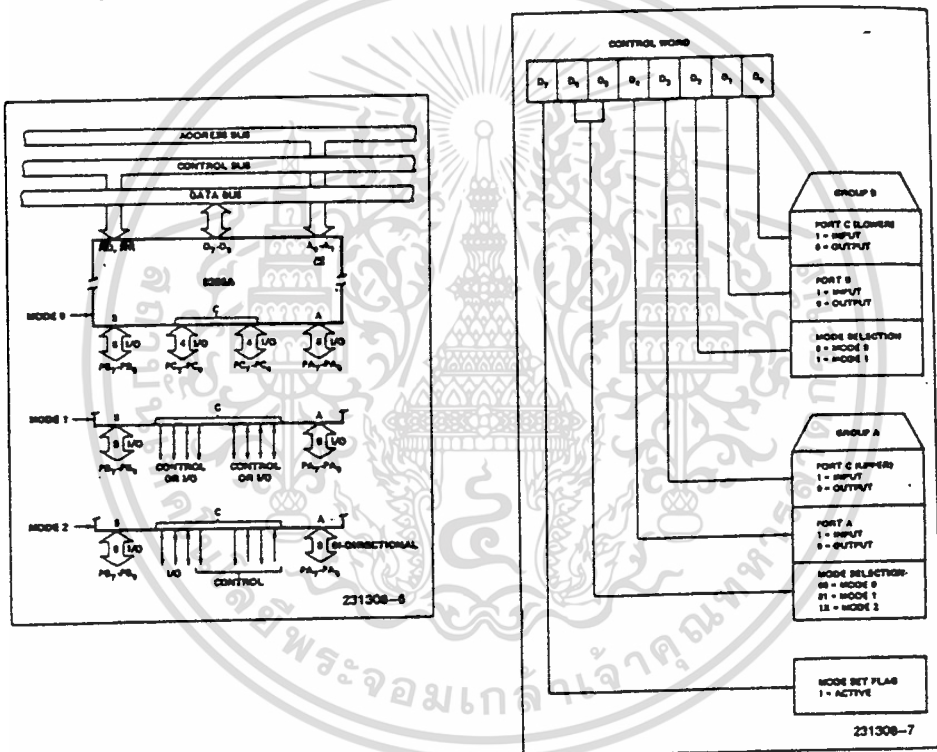
**A0-A1 (Address Input) :** จะเป็นตัวกำหนดการเลือกใช้รีจิสเตอร์ภายในของ 8255 ซึ่งจะกล่าวรายละเอียดในภายหลัง

**RESET :** เมื่อขานี้มีสถานะเป็น "1" 8255 จะอยู่ในสภาวะรีเซ็ต ทุก ๆ พอร์ตของ 8255 จะถูกเซ็ตให้อยู่ในโหมดอินพุท

### 3.16 โหมด 0 : การทำงานเป็นพอร์ตอินพุต / เอาต์พุตพื้นฐาน

ในการเซ็ท 8255 ให้อยู่ในโหมด 0 นั้น เราจะต้องส่งคำสั่งควบคุม (Control Word) ให้แก่วีธีเตอร์ควบคุมก่อน คำสั่งควบคุมนี้จะกำหนดลักษณะการทำงานให้แก่แต่ละพอร์ต 8255 ตัวอย่างหนึ่งของคำสั่งควบคุมที่จะสั่งให้ 8255 ทำงานอยู่ในโหมด 0 นี้ได้แก่

D7 D6 D5 D4 D3 D2 D1 D0  
 . 1 0 0 0 0 0 0 0



รูปที่ 3.15 แสดงรหัสควบคุมการทำงานของ 8255

จากรูปที่ 3.15 จะเห็นว่า

บิต D7 เป็นตัวกำหนดว่า เป็นคำสั่งควบคุม (CONTROL WORD)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต D6 และ D5 กำหนดโหมดการทำงานของ พอร์ต A D6, D5 มีค่าเป็น "0"

แสดงว่าอยู่ในโหมด 0

บิต D4 = "0" กำหนดให้พอร์ต A เป็นพอร์ตเอาต์พุต

บิต D3 = "0" เช็ทพอร์ต C 4 บิตบนเป็นพอร์ตเอาต์พุต

บิต D2 = "0" เช็ทโหมดของพอร์ต B ให้พอร์ต B อยู่ในโหมด 0

บิต D1 = "0" เช็ทพอร์ต B เป็นพอร์ตเอาต์พุต

บิต D0 = "0" เช็ทพอร์ต C ให้ 4 บิตล่างเป็นพอร์ตเอาต์พุต

การทำงานในโหมด 0 ของ 8255 นี้จะสั่งให้พอร์ตของ 8255 เป็นอินพุตหรือเอาต์พุตก็ได้ เช่น ให้พอร์ต A และพอร์ต C เป็นพอร์ตเอาต์พุตและพอร์ต B เป็นพอร์ตอินพุต เราจะต้องส่งคำสั่งควบคุมให้แกรีจิสเตอร์ควบคุมในลักษณะนี้คือ

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	1	0

หลังจากที่ส่งคำสั่งควบคุมให้แกรีจิสเตอร์ควบคุมแล้ว 8255 จะถูกเช็ทให้มีลักษณะการทำงานดังรูปที่ 3.15 เราจะใช้คำสั่ง IN อ่านข้อมูลมาจากพอร์ต B ได้

IN A, (00) : อ่านข้อมูลจากพอร์ต B

A		B		GROUP A			GROUP B	
D4	D3	D1	D0	PORT A	PORT C		PORT B	PORT C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ขออนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

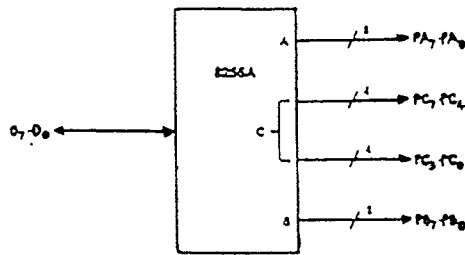
0	0	0	0	OUTPUT	OUTPUT	0	LOUTPUT	OUTPUT
0	0	0	1	OUTPUT	OUTPUT	1	LOUTPUT	INPUT
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	OUTPUT	INPUT	4	LOUTPUT	OUTPUT
0	1	0	1	OUTPUT	INPUT	5	LOUTPUT	INPUT
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUTPUT
1	1	1	1	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	INPUT	OUTPUT	8	LOUTPUT	OUTPUT
1	0	0	1	INPUT	OUTPUT	9	INPUT	INPUT
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	INPUT	OUTPUT	11	LOUTPUT	INPUT
1	1	0	0	INPUT	INPUT	12	LOUTPUT	OUTPUT
1	1	0	1	INPUT	INPUT	13	LOUTPUT	INPUT
1	1	1	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	INPUT	INPUT	15	INPUT	INPUT

รูปที่ 3.16 ไดอะแกรมแสดงการทำงาน 8255 ในโหมด 0

หลังจากส่งคำสั่งควบคุมให้ 8255 แล้ว ลักษณะการทำงานของพอร์ตต่าง ๆ ที่สามารถกำหนดได้ในโหมด 0 แสดงไว้ใน รูปที่ 3.17

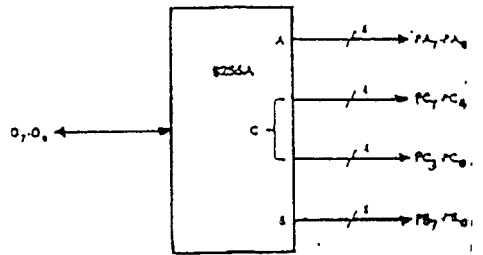
CONTROL WORD #0

$D_7$	$D_6$	$D_5$	$D_4$	$D_3$	$D_2$	$D_1$	$D_0$
1	0	0	0	0	0	0	0



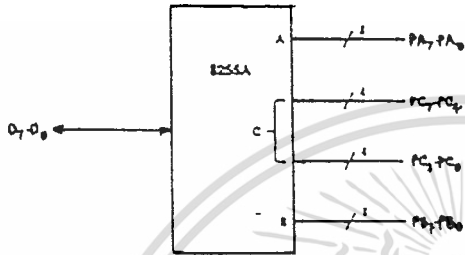
CONTROL WORD #2

$D_7$	$D_6$	$D_5$	$D_4$	$D_3$	$D_2$	$D_1$	$D_0$
1	0	0	0	0	0	1	0



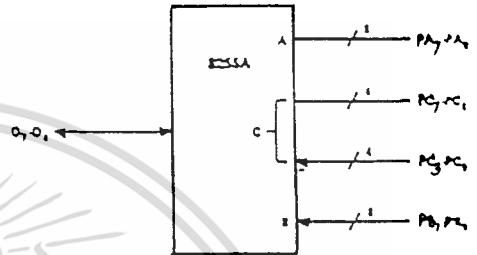
CONTROL WORD #1

$D_7$	$D_6$	$D_5$	$D_4$	$D_3$	$D_2$	$D_1$	$D_0$
1	0	0	0	0	0	0	1



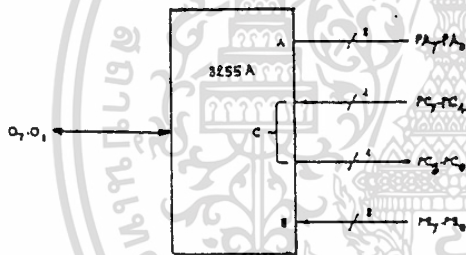
CONTROL WORD #3

$D_7$	$D_6$	$D_5$	$D_4$	$D_3$	$D_2$	$D_1$	$D_0$
1	0	0	0	0	0	1	1



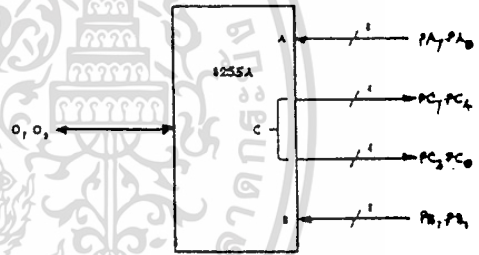
CONTROL WORD #6

$D_7$	$D_6$	$D_5$	$D_4$	$D_3$	$D_2$	$D_1$	$D_0$
1	0	0	0	1	0	1	0



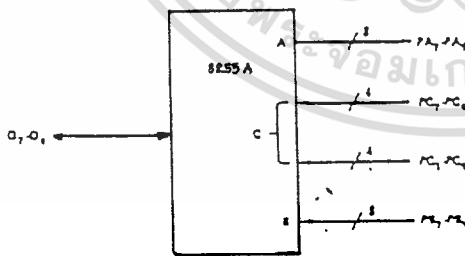
CONTROL WORD #10

$D_7$	$D_6$	$D_5$	$D_4$	$D_3$	$D_2$	$D_1$	$D_0$
1	0	0	1	0	0	1	0



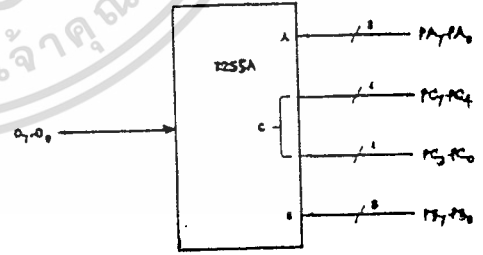
CONTROL WORD #7

$D_7$	$D_6$	$D_5$	$D_4$	$D_3$	$D_2$	$D_1$	$D_0$
1	0	0	0	1	0	1	1



CONTROL WORD #11

$D_7$	$D_6$	$D_5$	$D_4$	$D_3$	$D_2$	$D_1$	$D_0$
1	0	0	1	0	0	1	1



รูปที่ 3.17 แสดงลักษณะต่างๆ ของการใช้งาน 8255 ในโหมด 0

3.17 โหมด 1 : การทำงานเป็นพอร์ทอินพุท/เอาต์พุทแบบสโตริ่ง

ในโหมดนี้ 8255 จะทำหน้าที่เป็นพอร์ทอินพุท/เอาต์พุท แบบมีสัญญาณตรวจสอบความพร้อม (Hand Shaking) ด้วย โดยพอร์ท A และ พอร์ท B ทำหน้าที่เป็นพอร์ทข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาด 8 บิต ส่วนพอร์ท C บิตที่ 4-7 ทำหน้าที่เป็นสายสัญญาณตรวจสอบความพร้อมสำหรับพอร์ท B การทำงานในโหมด 1 มีรายละเอียดต่าง ๆ ดังนี้

การควบคุมแบ่งออกเป็น 2 กลุ่ม คือ กลุ่ม A และ B

แต่ละกลุ่มประกอบด้วยพอร์ทข้อมูล 8 บิต (พอร์ท A หรือ B) และบิตควบคุมอีก 8 บิต (พอร์ท C แบ่งครึ่ง) ใช้เป็นสัญญาณตรวจสอบความพร้อม

สำหรับพอร์ทข้อมูล 8 บิต (พอร์ท A หรือ B) จะเป็นได้ทั้งอินพุท/เอาต์พุท แบบค้างข้อมูล (Latch) เลือกได้ตามต้องการ สำหรับแต่ละ 4 บิต ของพอร์ท C จะใช้เป็นสัญญาณควบคุม และใช้แสดงสถานะของข้อมูลของพอร์ทข้อมูล 8 บิตแต่ละกลุ่ม

สัญญาณควบคุมในกรณีเลือกเป็นอินพุทแบบสโตรบ (Input Signal Control Definition)

STB (Strobe Input) เป็นขาสัญญาณอินพุทเมื่อได้รับลอจิก “LOW” จะแสดงให้เห็น 8255 รู้ว่า มีข้อมูลส่งเข้ามา และข้อมูลจะถูกค้างเก็บไว้ใน 8255 ซึ่งเป็นสัญญาณส่งมาจากอุปกรณ์ภายนอก

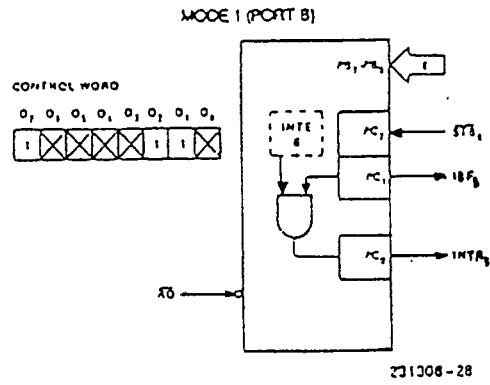
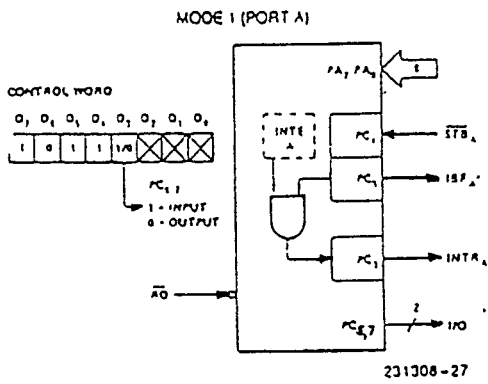
IBF (Input Buffer Full FF) เป็นขาเอาต์พุทเมื่อเป็นลอจิก “HIGH” จะแสดงว่า 8255 ได้รับข้อมูลเข้าไปเก็บในบัฟเฟอร์อินพุทแล้ว โดยที่ IBF จะส่งลอจิกเป็น “HIGH” หลังจาก 8255 ได้รับสัญญาณ STB ที่ขอบขาลง (Falling Edge of Strobe Input) และจะถูกรีเซ็ตเป็น “LOW” โดยสัญญาณ RD ที่ขอบขาลงซึ่งแสดงว่า ซีพียู ยอมรับการอินเตอร์รัพท์ และทำการรับข้อมูลจาก 8255 เสร็จแล้ว รูปที่ 3.20

INTB A เป็นฟลิปฟล็อปภายใน 8255 จะถูกควบคุมให้เป็น “1” หรือ “0” ด้วยรหัสควบคุม บิตเซ็ท / รีเซ็ต ของพอร์ท C ที่บิต PC4 (กลุ่ม A รูปที่ 31.9)

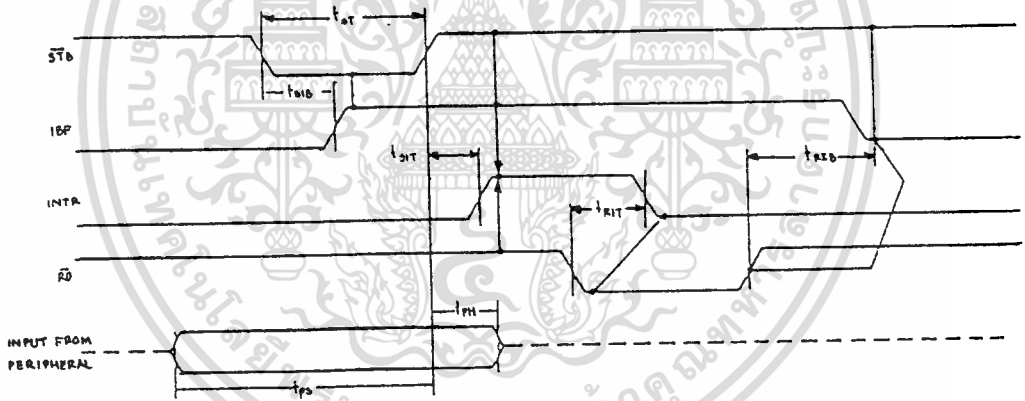
INTB B เป็นฟลิปฟล็อปภายใน 8255 จะถูกควบคุมให้เป็น “1” หรือ “0” ด้วยรหัสควบคุม บิตเซ็ท / รีเซ็ต ของพอร์ท C ที่บิต PC2 (กลุ่ม B รูปที่ 31.9)

Note : สัญญาณขออินเตอร์รัพท์ INTR สัญญาณสโตรบ STB สัญญาณ IBF เป็นสัญญาณควบคุมและแสดงสถานะข้อมูล กรณีกำหนดให้ 8255 ทำงานโหมด 1 แบบสโตรบอินพุท ซึ่งมีทั้งของกลุ่ม A และกลุ่ม B ได้จากขาของพอร์ท C รูปที่ 3.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 แสดงรหัสควบคุมและการทำงานในโหมด 1



รูปที่ 3.19 แสดงไทมิงสัญญาณต่าง ๆ ตามการทำงานในโหมด 1 ก

สัญญาณควบคุมในกรณีเลือกเป็นเอาต์พุตแบบโสตริบ (Output Control Signal Definition)

OBF (Output Buffer Full ff) เป็นขาสัญญาณเอาต์พุต จะเป็น "LOW" เพื่อแสดงว่า ได้มีการส่งข้อมูลออกมาที่พอร์ตที่กำหนดจาก ซีพียู มายัง 8255 แล้ว โดยที่ OBF จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

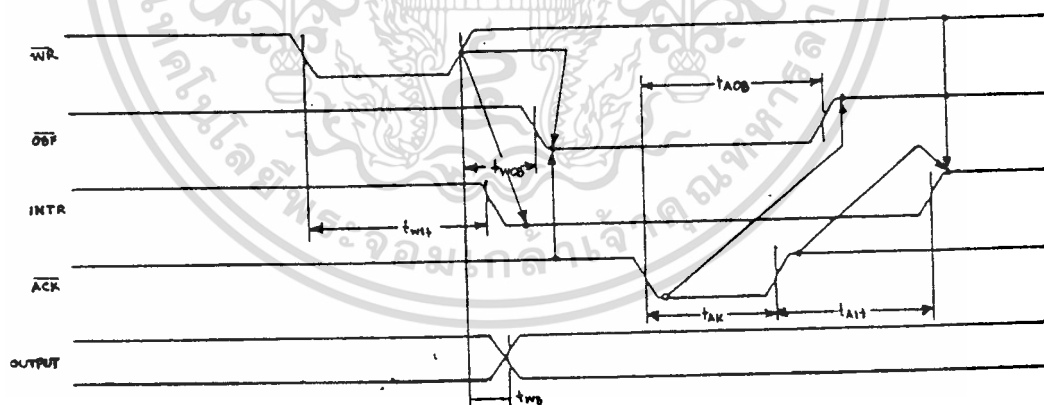
ส่งลอจิกเป็น “LOW” เมื่อได้รับสัญญาณ WR ที่ขอบขาขึ้น และจะถูกทำให้เป็น “HIGH” เมื่อได้รับสัญญาณ ACK ที่ขอบขาลง รูปที่ 3.20

ACK (Acknowledge Input) เป็นขาสัญญาณอินพุต เมื่อได้รับลอจิก “LOW” จะแสดงให้ 8255 รู้ว่าข้อมูลที่พอร์ท A หรือ B ของ 8255 ได้ถูกรับไปแล้ว ซึ่งเป็นสัญญาณตอบรับจากอุปกรณ์ภายนอกส่งมาที่ 8255 เพื่อขอรับข้อมูลจากพอร์ทของ 8255

INTR (Interrupt Request) เป็นสัญญาณเอาต์พุตซึ่งจะเป็น “HIGH” เพื่อขออินเตอร์รัพท์ ซึ่พียู เมื่ออุปกรณ์ภายนอกรับข้อมูลจากพอร์ทของ 8255 แล้ว โดยส่งสัญญาณ ACK เป็น “LOW” โดยสัญญาณขอบขาลงของ WR ซึ่งแสดงว่า ซึ่พียู รับรู้การขออินเตอร์รัพท์ และทำการส่งข้อมูลใหม่ให้กับ 8255 แล้ว

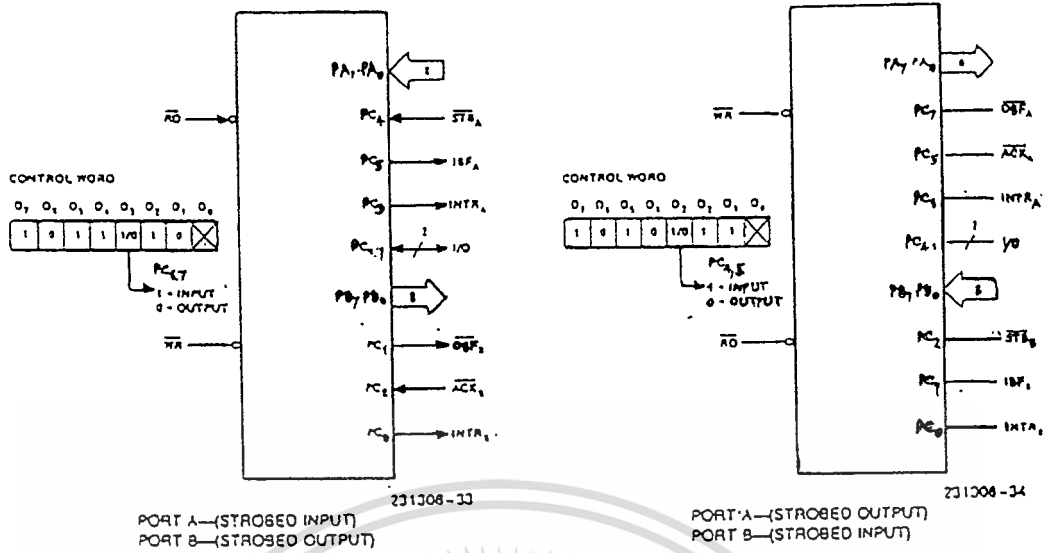
INTE A เป็นฟลิปฟลอปภายใน 8255 จะถูกควบคุมให้เป็น “1” หรือ “0” ด้วยรหัสควบคุมบิตเจ็ด / รีเซ็ท ของพอร์ท C ที่มีบิต PC 6 (กลุ่ม A รูปที่ 3.22)

Note : สัญญาณ INTR , ACK และ OBF เป็นสัญญาณควบคุม และแสดงสถานะการทำงานของ 8255 ในโหมด 1 แบบ ไสตรับเอาต์พุต ซึ่งมีทั้งของกลุ่ม A และกลุ่ม B ได้จากขาของพอร์ท C รูปที่ 3.22

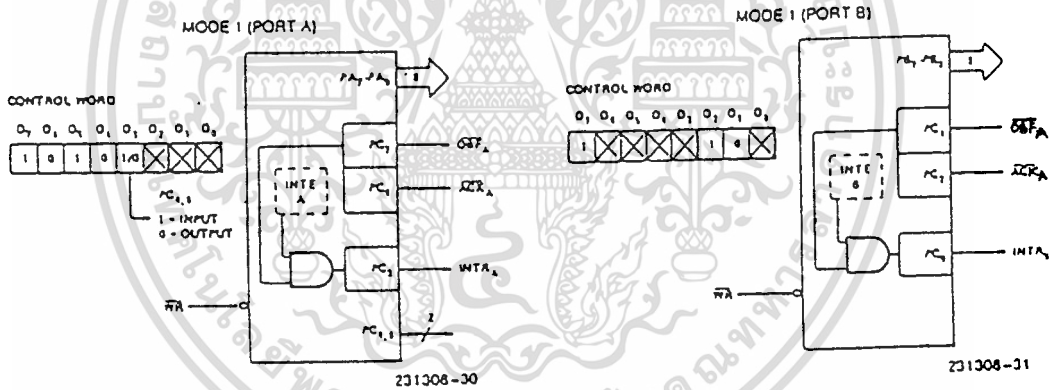


รูปที่ 3.20 แสดงไทมิ่งสัญญาณต่าง ๆ ตามการทำงานในโหมด 1 ข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.21 แสดงรูปแบบการทำงานในโหมด 1



รูปที่ 3.22 แสดงรหัสควบคุมและการทำงานในโหมด 1

3.18 โหมด 2 : การทำงานแบบบัสสองทิศทาง

ในโหมดนี้จะใช้ได้กับกลุ่ม A เท่านั้น โดยพอร์ท A ทั้ง 8 บิตจะถูกจัดให้เป็นทั้งพอร์ทสำหรับการส่งข้อมูลและการรับข้อมูล (Bi-directional Bus) โดยมีพอร์ท C ทำหน้าที่เป็นสายสัญญาณตรวจสอบความพร้อม (Hand Shaking) การทำงานโหมด 2 มีรายละเอียด ดังนี้

ใช้ได้กับกลุ่ม A เพียงกลุ่มเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีพอร์ท A ขนาด 8 บิต เพียงพอร์ทเดียว ที่ทำงานแบบบัสดสองทิศทางได้โดยมี สัญญาณสำหรับควบคุมอีก 5 บิต ได้จากพอร์ท C

การทำงานทั้งอินพุทและเอาต์พุท เป็นแบบค้างข้อมูลได้ (Latch Signal)

สัญญาณสำหรับควบคุม 5 บิต (พอร์ท C) ใช้สำหรับควบคุมการทำงานและแสดงสถานะ (Status) ของข้อมูลที่พอร์ท

สัญญาณควบคุมในกรณีโหมด 2 ทำงานแบบบัสดสองทิศทาง

INTR A (Interrupt Request) เป็นขาสัญญาณเอาต์พุทจะเป็น "HIGH" เมื่อต้องการขออินเตอร์รัพท์ ซึ่พียู ใช้ทั้งการทำงานเป็นอินพุทและเอาต์พุท

กรณีทำงานเป็นเอาต์พุทพอร์ท

(1) OBF A (Output Buffer Full) เป็นขาสัญญาณเอาต์พุทเมื่อขานี้เป็น "LOW" แสดงว่า ขณะนี้ 8255 ได้รับข้อมูลจากซีพียู มายังบัฟเฟอร์ภายในของพอร์ท A แล้ว

(2) ACK A (Acknowledge) เป็นขาอินพุทเมื่อได้รับลอจิก "LOW" จะเป็นสัญญาณบอกให้ 8255 ส่งข้อมูลออกมาที่พอร์ท A ให้กับอุปกรณ์ภายนอกกลับไป และเมื่อ ACK เป็น "HIGH" จะทำให้เอาต์พุทพอร์ท A อยู่ในสถานะอิมพีแดนซ์สูง

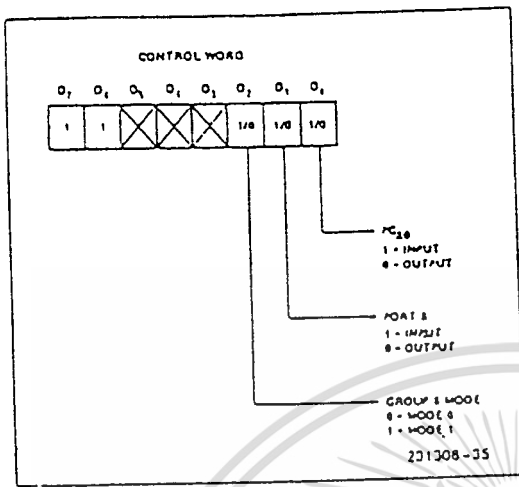
(3) Inte 1 (The Interup FF Associated with OBF) เป็นฟลิปฟลอปภายใน 8255 ซึ่งจะสร้างสัญญาณขออินเตอร์รัพท์ INTR A โดยทำงานร่วมกับขา OBF โดยสามารถเจ็ทหรือรีเซ็ทด้วยรหัสควบคุม บิตเจ็ท/รีเซ็ท ของพอร์ท C ที่บิต PC6 รูปที่ 3.23

กรณีที่ทำงานเป็นอินพุทพอร์ท

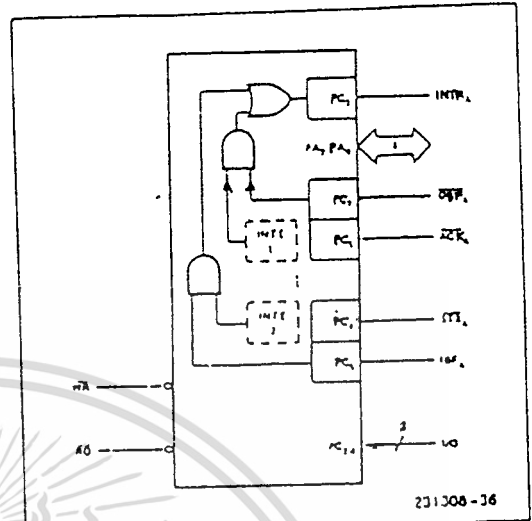
(1) STB A (Strobe Input) เป็นสัญญาณอินพุท เมื่อได้รับสัญญาณลอจิก "LOW" จะทำให้ 8255 รับข้อมูลเข้าทางพอร์ท A และเก็บค้างไว้ในอินพุทบัฟเฟอร์ของ 8255

(2) IBF A (Input Buffer Full FF) เป็นขาสัญญาณเอาต์พุทจะเป็น "HIGH" เพื่อแสดงให้อุปกรณ์นอกัรู้ว่า ได้ทำการรับข้อมูลเข้ามาเก็บค้างไว้ใน 8255 เรียบร้อยแล้ว

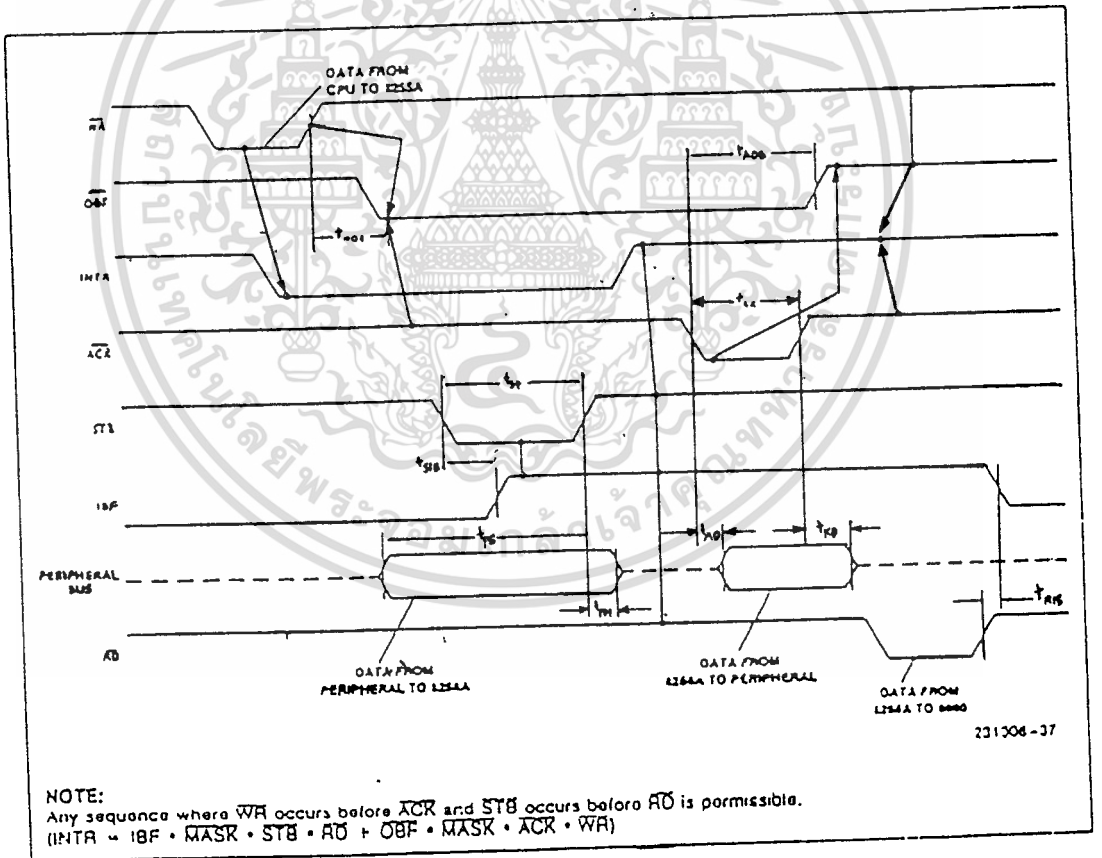
(3) INTB 2 (The Interrupt FF Associated with IBF) เป็นฟลิปฟลอปภายใน 8255 ซึ่งจะสร้างสัญญาณขออินเตอร์รัพท์ INTR A โดยทำงานร่วมกับขา IBF โดยสามารถเจ็ทหรือรีเซ็ทด้วยรหัสควบคุม บิตเจ็ท/รีเซ็ท ของพอร์ท C ที่บิต PC รูปที่ 3.23



MOOE Control Word



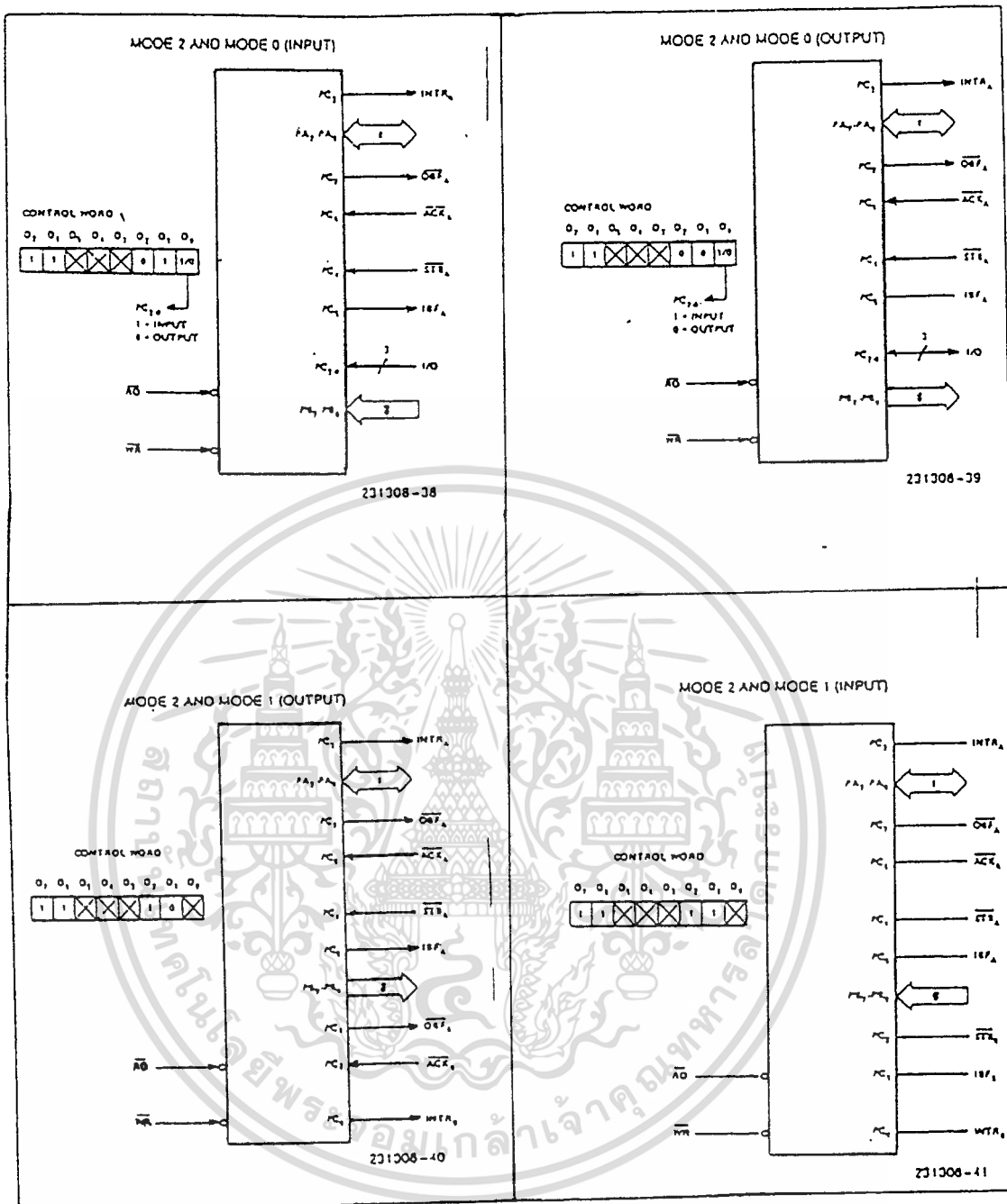
MODE 2



NOTE:  
Any sequence where WA occurs before ACR and STB occurs before RD is permissible.  
(INTR = IBF • MASK • STB • RD + OBF • MASK • ACR • WA)

รูปที่ 3.23 แสดงรหัสควบคุมและไทมิ่งสัญญาณต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.24 แสดงการทำงานโหมด 2 (พอร์ต A) ร่วมกับโหมด 0 (พอร์ต B) และร่วมกับโหมด 1 (พอร์ต B)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

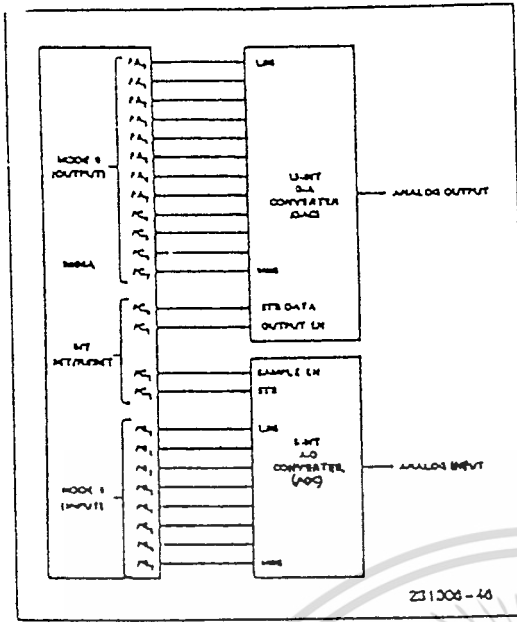


Figure 22. Digital to Analog, Analog to Digital

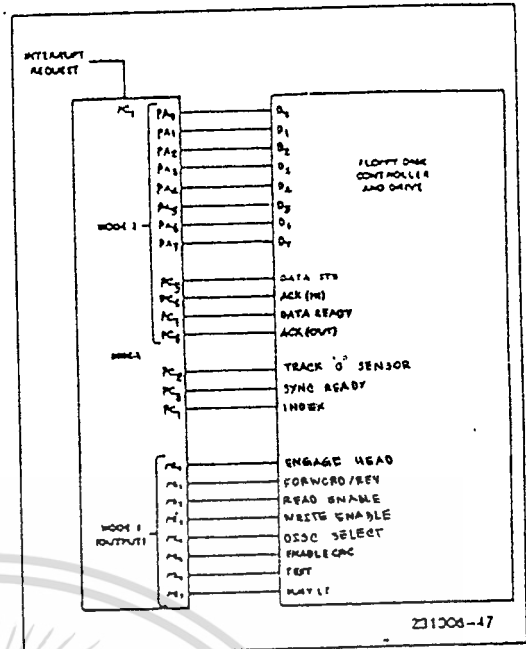
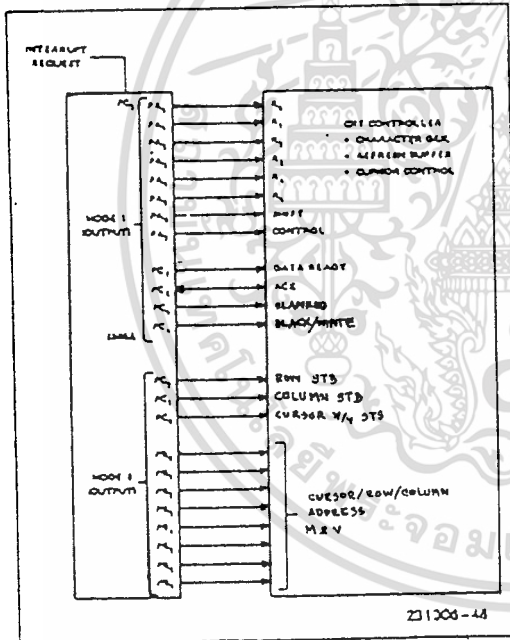


Figure 23. Basic Floppy Disk Interface



Basic CRT Controller Interface

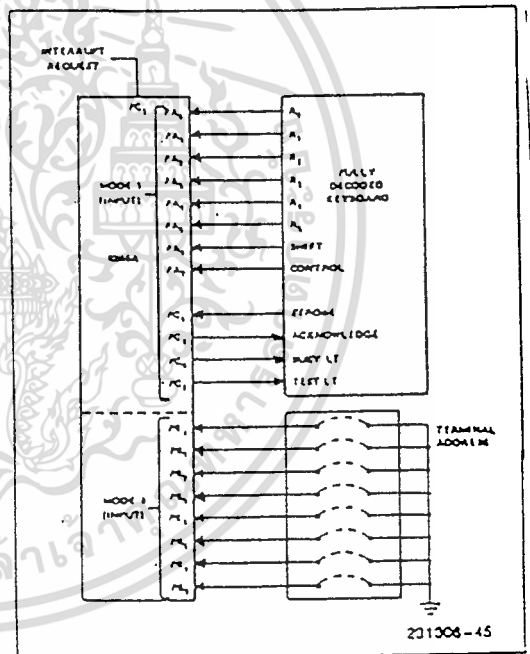


Figure 21. Keyboard and Terminal Address Interface

รูปที่ 3.25 แสดงการใช้งานไอซีชิป 8255 อินเทอร์เฟซกับอุปกรณ์ภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

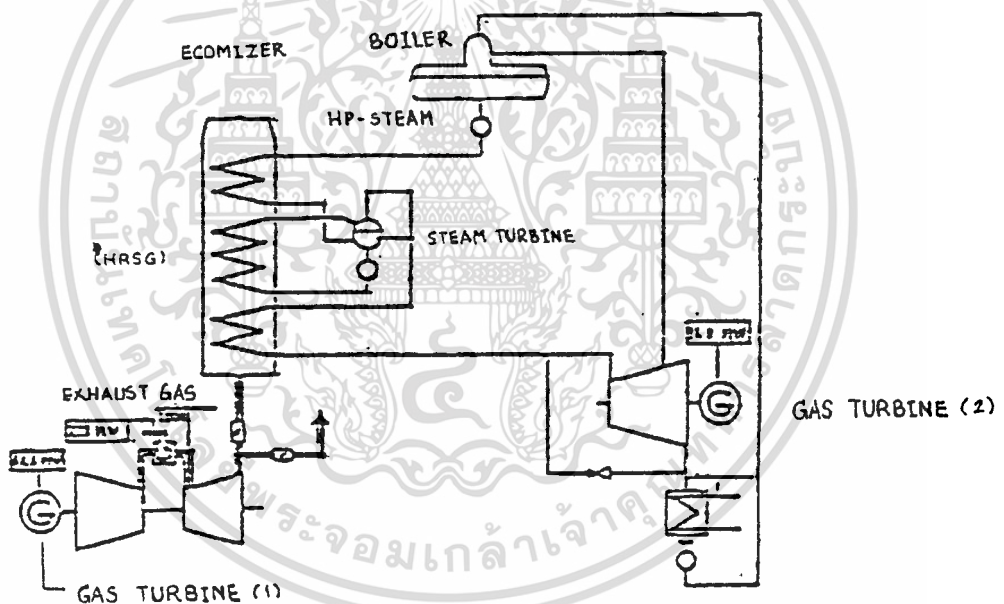
## บทที่ 4

### รายละเอียดของชุดสาธิต โรงไฟฟ้าพลังความร้อนร่วม

ความรู้พื้นฐานเกี่ยวกับ โรงงาน ไฟฟ้าพลังความร้อนร่วม

โรงงานไฟฟ้าความร้อนร่วมแบ่งออกเป็นชนิดต่าง ๆ 6 แบบ คือ

#### 1. SIMPLE SINGLE PRESSURE SYSTEM (STEAM PRESSURE ระดับเดียว)



#### รูป SIMPLE SINGLE PRESSURE SYSTEM

การวางอุปกรณ์ของโรงไฟฟ้าพลังความร้อนร่วม ชนิดที่ธรรมดาที่สุดคือ แบบ SINGLE PRESSURE ซึ่งโดยทั่วไปจะประกอบด้วย GAS TURBINE หนึ่งเครื่องหรือมากกว่าหนึ่งเครื่อง พร้อม WASTE HEAT BOILER แบบ SINGLE, STEAM TURBINE CONDENSER และ FEEDWATER PREHEATER ใน DEAERATOR ใช้น้ำที่ใช้สำหรับ DEAERATOR จึงมาจาก STEAM TURBINE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โรงงานไฟฟ้าแบบนี้เหมือนกับโรงไฟฟ้าระบบความร้อน BLOCK 1,2 ที่บางปะกง ซึ่งประกอบด้วยตัวกังหันชุดเดียวหรือหลายชุด แต่มีกังหันไอน้ำชุดเดียว

HEAT RECOVERY STEAM GENERATOR ประกอบด้วย 3 ส่วน คือ

A) ECOMIZER เป็นส่วนที่เพิ่มความร้อนให้กับน้ำ โดย SENSIBLE HEAT ของ EXHAUST GAS

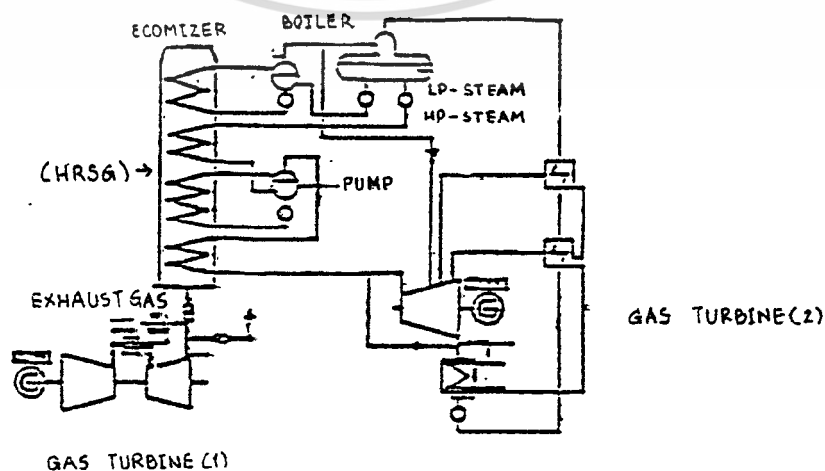
B) EVAPORATOR เป็นส่วนที่น้ำกลายเป็น WET STEAM โดย LATENT HEAT และส่วนนี้จะมีระบบ CIRCULATION ด้วย NATURAL CIRCULATION หรือ FORCED CIRCULATION ก็ได้

C) SUPERHEAT เป็นส่วนที่ทำให้ WET STEAM กลายเป็น SUPERHEAT STEAM เพื่อนำไปหมุน STEAM TURBINE

โรงไฟฟ้าแบบนี้จะมีประสิทธิภาพปานกลาง 45 % และง่ายต่อการเดินเครื่องและใช้ STEAM จาก EXTRACTION LOW PRESSURE TURBINE ไปอุ่นน้ำใน DEAERATOR HEATER

ส่วนข้อเสียแบบ SIMPLE SINGLE PRESSURE นั้น คือการได้ใช้ HEAT จาก EXHAUST GAS หรือได้พลังงานจาก HEAT RECOVERY STEAM GENERATOR นั้น น้อยเกินไปจึงทำให้ประสิทธิภาพของ PLANT ไม่สูงนัก

## 2. SINGLE PRESSURE STEAM WITH A PREHEATING LOOP



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับอาจารย์และบุคลากรที่สอนวิชานี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจัดวางโรงไฟฟ้าแบบนี้จะสามารถเพิ่ม CYCLE EFFICIENCY จากแบบ SINGLE PRESSURE เพราะมีการเพิ่ม PREHEATING LOOP เพื่อดึงความร้อนจาก FLUE GAS มาใช้งานให้มากขึ้นก่อนที่ออกปล่อง แต่ PREHEATING LOOP จะต้องได้รับการออกแบบให้เหมาะสมเพื่อไม่ให้อุณหภูมิของ FLUE GAS มีค่าต่ำกว่า ACID DEW POINT หาก SULPHUR ปนอยู่ในเชื้อเพลิงกับ LOW TEMP CORROSION ใน HRSG

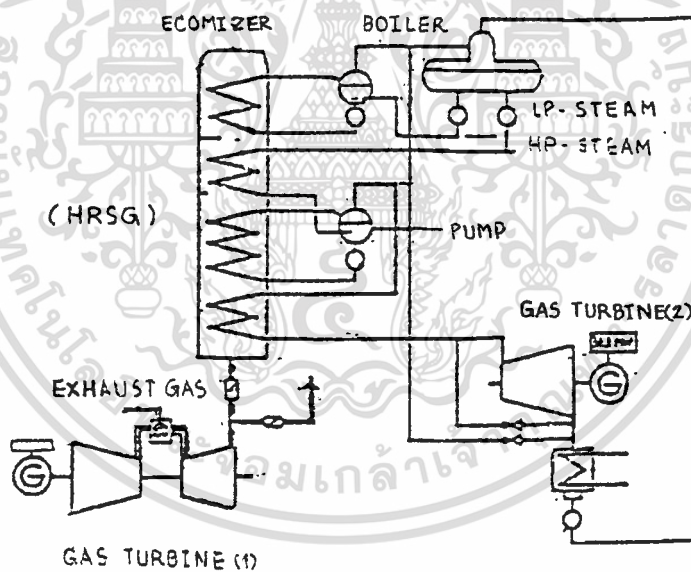
PREHEATER LOOP

ในที่นี้ทำหน้าที่ผลิตไอน้ำอิมตัวป้อนเข้า  
DEAERATOR

GROSS EFFICIENCY

ของโรงไฟฟ้าพลังความร้อนประเภทนี้มีค่าประมาณ  
45.13 %

### 3. TWO PRESSURE SYSTEM (ใช้เชื้อเพลิงที่มีกำมะถันปนอยู่)



รูป TWO PRESSURE SYSTEM

พิจารณาจากรูปด้านบนจะเป็นว่าภายใน WASTE HEAT BOILER นั้น มีวงจรไอน้ำอยู่ 2 แบบ คือ LOW PRESSURE STEAM และ HIGH PRESSURE STEAM PLANT แบบนี้ การใช้ HEAT จาก EXHAUST GAS ตามสภาพ TEMPERATURE INGREDIENT มากขึ้น จึงให้ STEAM FLOW มาก ประสิทธิภาพของ PLANT 47.9% และ OUT PUT ของ

เอกสารนี้เป็นเอกสารตัวอย่างไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

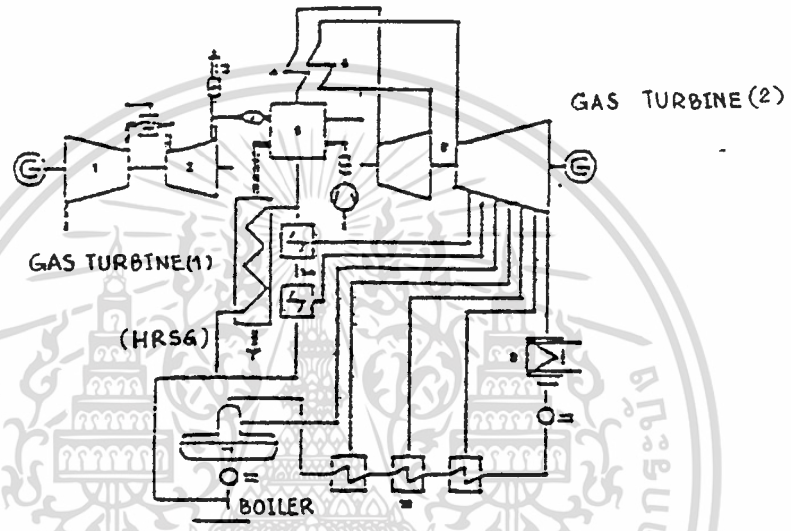
STEAM ก็จะดีขึ้นด้วย

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ข้อควรระวังในการ OPERATE PLANT แบบนี้

ACID DEW POINT TEMPRATURE ที่จะเกิดใน COIL ของ WASTE HEAT BOILER เสมอ เริ่มต้นที่ต้องรักษา FEED FINAL WATER TEMP. ให้เหนือกว่า (จึงต้องการ HEAT น้ำถึง 3 STAGE)

#### 4. COMBINED CYCLE PLANTS WITH MAXIMUM SUPPLEMENTARY FIRING



รูป COMBINED CYCLE PLANTS WITH MAXIMUM SUPPLEMENTARY FIRING

โรงไฟฟ้าแบบนี้จะมี FORCE DRAFT FAN STANBY อยู่ด้วย เพื่อที่ว่า PLANT จะยังสามารถเดินเครื่องได้เมื่อ GAS TURBINE ทั้งหมด TRIP ไปแล้ว เชื้อเพลิงที่ใช้ใน HRSG อาจเป็นคณละชนิดกับที่ใช้ใน GAS TURBINE ได้ ตัวอย่างของโรงไฟฟ้าประเภทนี้คือ EMILE HUCHAT POWER STATION ขนาดกำลังผลิต 343 MW.

การเดินเครื่องโรงไฟฟ้าพลังความร้อนร่วมโดยใช้น้ำมันเตาเป็นเชื้อเพลิง

โดยหลักการทั่ว ๆ ไปนั้นเหมือนกับโรงไฟฟ้าพลังความร้อนทั่ว ๆ ไป เพียงแต่พนักงานเดินเครื่องนั้นจะต้องศึกษา หรือมีความรู้เพิ่มเติมดังนี้

1. ความรู้พื้นฐานทางเคมี และหลักการของ Gas Turbine แต่ละ Type เป็นอย่างไร เพราะคุณลักษณะของน้ำมันเตา แต่ละชนิดนั้น มีผลกระทบต่ออายุ และประสิทธิภาพของ Gas Turbine และ Waste Heat Boiler
2. ความรู้เรื่อง Heavy Oil Fuel AND Corros เครื่องปรับปรุงสภาพน้ำมันเตา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า (Washing Plant) ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ต้องทราบจุดอ่อนของ Hot Gas Part จึงประกอบด้วย Alloys แต่ละชนิดที่ต่างกัน และจะมีความทนทานต่อสารประกอบของน้ำมันเตาที่เกิดจากการเผาไหม้ได้ไม่เหมือนกัน (บางครั้งพนักงานเดินเครื่องต้องลด Inlet Temperature ลง เพื่อรักษาอายุของเครื่องจักรให้ยาวขึ้น)

4. การตรวจสอบคุณภาพของน้ำมันเตาควรทำทุก ๆ 3 - 4 ชั่วโมง หรือมากกว่านั้นถ้าสภาพน้ำมันไม่แน่นอน

5. พนักงานเดินเครื่องต้องรักษา Feed Final Water Temperature ที่จะป้อนเข้า Waste Heat Boiler เพื่อหลีกเลี่ยงการสึกกร่อนของโลหะผิวนอกเนื่องจาก Acid Dew Point

6. ต้องเอาใจใส่ Fouling Effect ที่เกิดกับ Blades อันเนื่องจากการเผาไหม้ของน้ำมันเตา และวางแผนการ Cleaning Blades ตามความเหมาะสม

7. เมื่อ Blades เกิดรอย Cracked ควรศึกษาว่าอะไรบ้างที่เป็นตัวที่ทำให้เกิดรอย Crack เพื่อป้องกันมิให้ Crack นั้นขยายใหญ่เพิ่มขึ้นอย่างรวดเร็วจนเครื่องจักรเสียหายได้ ส่วนมากเกิดจาก Temperature เปลี่ยนแปลงมากในระยะเวลาอันสั้นจากการ Startup หรือเพื่อลด Load พวก Unburnt Carbon ต่าง ๆ ที่เกิดจากการเผาไหม้ที่ไม่สมบูรณ์จะพ่นชน Blades สึกกร่อนมากกว่าปกติด้วย

8. ถ้า Steam Turbine มีความดันของไอน้ำเป็น 2 ระบบ เช่น Low Pressure กับ High Pressure พนักงานเดินเครื่องต้องระวังอย่าให้เกิดความเปลี่ยนแปลงของอุณหภูมิในช่วงกว้างบ่อยครั้ง (Frequency of High Differential Temperature) ของทั้งสองระบบดังกล่าวมีมากนัก จะทำให้เกิด Thermal Stress Fatigue ทำให้อุปกรณ์แตกร้าวก่อนเวลาอันสมควร

การเลือกสถานที่ติดตั้งโรงไฟฟ้าพลังความร้อนร่วม ควรมีหลักเกณฑ์ ดังนี้

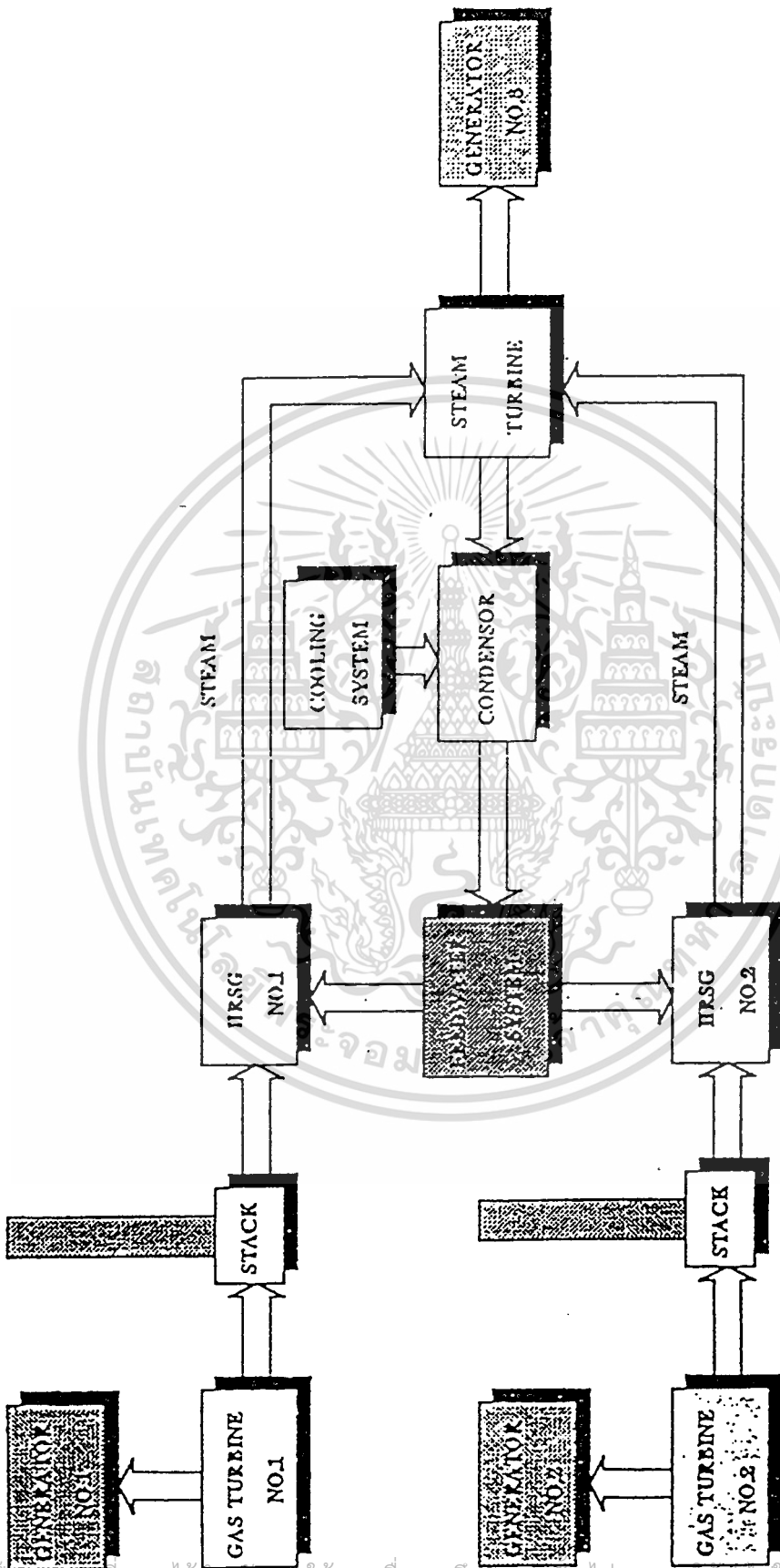
- ควรห่างไกลทะเลอย่างน้อย 60 กิโลเมตร เนื่องจากไอน้ำร้อนนั้นทำให้อายุของ Blade และระยะเวลาการทำความสะอาด Blade สั้นลง

- ควรห่างจากเมืองอุตสาหกรรม สารจำพวกเคมีต่าง ๆ Unburnt Hydro Carbon และ Sulphur ที่ออกจากปล่องค่าใช้จ่ายค่าอุปกรณ์กรองฝุ่นและอากาศมากขึ้นไปอีก

- การสร้าง Plant แบบนี้ต้องพิจารณาทิศทางลมพัดด้วยเนื่องจากควันและพวก Unburnt Hydro Carbon Sulphur ที่ออกจากปล่อง อาจถูกลมพัดกลับเข้า Compressor ได้อีก ซึ่งมีผลต่อประสิทธิภาพ และอายุใช้งานของ Plant หรือบางที่ต้อง Shut Down Plant เพื่อเปลี่ยน Filter บ่อย ๆ ทำให้ Availability ของ Plant ลดลง และยังทำให้ Maintenance Cost

สูงขึ้นด้วย เอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 BLOCK DIAGRAM OF COMBINED CYCLE POWER PLANT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Condensate System (FWC)

### 1. หน้าที่ของระบบ

Turbine LP Exhaust Steam และ LP & HP By Pass Steam หลังจากถูกควบแน่น (Condense) ภายใน Condenser เปลี่ยนสถานะจากไอน้ำเป็นน้ำ ระบบ Condensate ทำหน้าที่

1.1 รักษาระดับน้ำใน Deaerator Storage Tank ให้มีระดับน้ำคงที่ และเพียงพอที่จะจ่ายให้กับ HRSG ทั้ง 2 ตัว

- 1.2 จ่ายน้ำให้กับ
- HP. Bypass Desuper Heat Spray Sys
  - LP. Bypass Desuper Heat Spray Sys
  - Condenser Desuper Heat Spray
  - Warm Up Stem Desuper Heat Spray
  - Steam Turbine Seal Sys. Desuper Heat Spray
  - Steam Turbine Exhaust Hood Spray

1.3 ทำหน้าที่ระบายความร้อนจาก Steam ใน Gland Condenser Condenser Hot Well

Condenser Hot Well อุปกรณ์ล่างสุดของ Condenser รองรับ Condensate ซึ่งควบแน่น จาก Steam Drain ที่ลง Condenser และ Water Drain ที่ลง Condenser

Steam Drain และ Water Drain ที่ Drain ลง Condenser มี

- Steam Turbine Exhaust
- HP. และ LP. Steam Bypass
- Main Steam Warm Up Line
- Vent จาก Condensate Dump Stuffing Box และ Strainer
- HP. และ LP. Steam Line Drain Pots.
- Gland Steam Condenser
- Gland Steam Separator
- Gland Steam Seal System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Condenser Hot Well เป็น Suction ของ Condensate Pumps สูบน้ำไปเก็บที่ Deaerator ระดับน้ำใน Con. Hot Well ถูกควบคุมให้คงที่ตลอดเวลา เพื่อ Supply น้ำให้ระบบ และไม่เกิดความเสียหายกับ Condensate Pumps

\* Condensate Pumps

Pump มี 3 ตัว ใช้งาน 2 ตัว Stand By 1 ตัว

Pump เป็นแบบ Multistage Centrifugal Vertical Pump

- Suction มี Manual Isolating Value
- Discharge มี Check Value, Motor Operated isolating Value
- Suction Strainer เป็นแบบ Simplex Basket มี Diff. Press วัดคล่อม Strainer เพื่อคุณภาพการดูดตันของ Strainer
- Pump Seal Water ที่ควบคุมความดันเหมาะสมกับการ Seal ที่ Pump Stuffing

\* Gland Condenser

\* Deaerator Heater

ทำหน้าที่เป็น Heater และ แยก Dissolved Non-Condensable Gases.

น้ำจาก Condensate Pumps ไหลผ่าน Gland Condenser ถูก Heat ด้วย LP. Steam Turbine Extraction ที่ Deaerator Heater โดยการถ่ายเทความร้อน จากการผสมกันระหว่าง Steam กับน้ำเพิ่มอุณหภูมิของน้ำให้สูงขึ้นก่อน Feed น้ำเข้า Boiler

ที่ Deaerator Heater Dissolved Non-Condensable Gases จะถูกแยกตัวออกจากน้ำ และ ถูกกำจัดออกทิ้งทาง Deaerator Heater Vents

\* Deaerator Storage Tank

เป็นถังเก็บน้ำร้อนจาก Condensate Pumps -> Gland Condenser -> Deaerator Heater ปริมาณน้ำที่เก็บไว้ใน Deaerator ต้องมีเพียงพอ เพื่อเป็น Suction ของ HP. และ LP. Boiler Feed Pumps Supply น้ำให้กับ Boiler

ระดับน้ำใน Deaerator ถูกควบคุมให้ระดับคงที่โดย Condensate Control Values.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2. Normal Operation

เดินเครื่องปกติ HRSG 2 ตัว ใช้ Condensate Pump 2 ตัว Standby 1 ตัว

### DEAERATOR AND FEEDWATER TANK

\*\*\*\*\*

#### 1. DESCRIPTION AND OPERATING INSTRUCTIONS

The D

ELAS deaerators for power plants have :

- horizontal storage tank
- horizontal dome deaerator

The dome is located above the storage tank as shown on figure 1 here below.

The deaerator dome is attached to the storage tank with 3 large pipes (one down comer and two steam balancing pipes).

The drawings RAY OO MF-070 FM 001 and FM 006 are necessary for fully understand this notice.

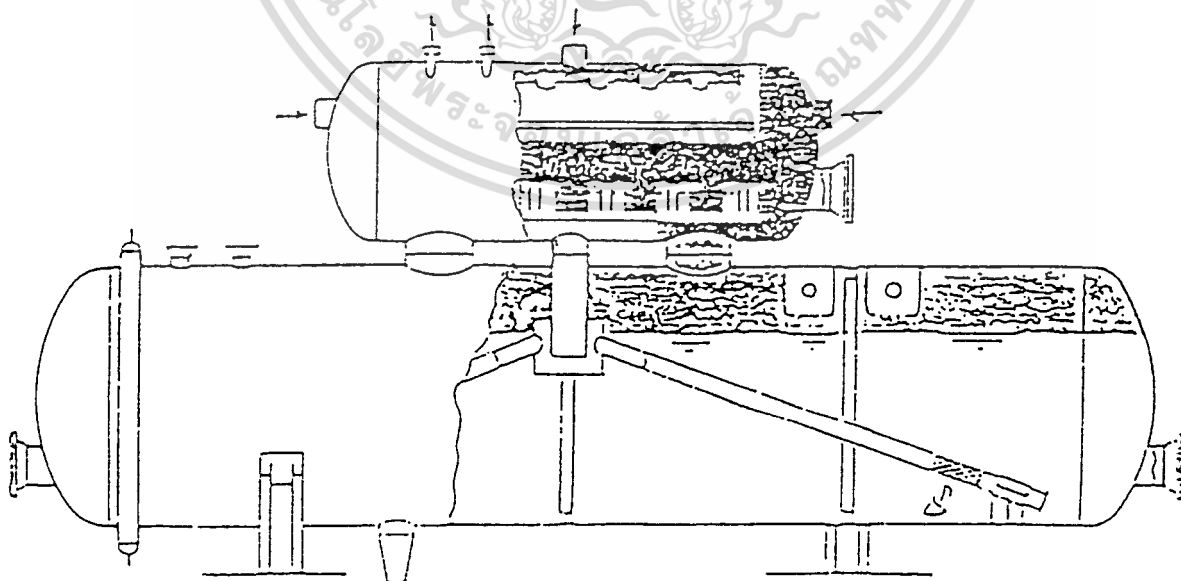


Figure 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Deaerator

### หน้าที่ (Function)

Deaerating Heater มีหน้าที่ กำจัดพวก Dissolved Non-Condensable Gases และทำหน้าที่อุ่นน้ำให้ร้อนขึ้นก่อนที่จะเข้า Boiler Deaerating Heater ซึ่งต่อไปจะเรียก Deaerator นี้ ประกอบด้วยเปลือกนอก (Pressure Vessel) ซึ่งเป็นที่ให้น้ำกับ Steam ผสมกันเพื่อเพิ่มความร้อนให้กับน้ำ เมื่อน้ำมีอุณหภูมิสูงขึ้น Non-Condensable Dissolves Gases จะแยกตัวออกมาและถูกปล่อยทิ้งไป นั่นหมายถึงน้ำที่กำจัด Non-Condensable Dissolves Gases นี้จะไม่ทำให้เกิด Corrosion

Deaerator จะเปรียบเสมือนเป็นยามเฝ้า Boiler Plant ป้องกัน Feed Pumps, Piping, Boiler และอุปกรณ์อื่น ๆ ของ Boiler ไม่ให้เกิดปัญหา อันเนื่องมาจาก Corrosion

### หลักการทำงาน

ในช่วงแรก (First Stage) น้ำจะถูก Spray พร้อมกับ การผ่าน Steam เข้ามาเพื่อเป็นการอุ่นน้ำให้ร้อนขึ้นมากกว่า Steam Saturation Temperature ประมาณ 2 องศา ทำให้ Oxygen และ Free Carbon Dioxide ถูกกำจัดออกจากน้ำ น้ำจะถูกฉีดเป็นฝอย โดย Self-Adjusting Spray Valves ซึ่งถูกออกแบบมาให้มีรูปแบบในการ Spray เป็นไปในแบบเดียวกันตลอดในทุกสถานะของ Load และเนื่องจากสถานะของอุณหภูมิดังที่ Gas จึงถูกกำจัดออกที่จุดนี้เอง

น้ำจาก First Stage ซึ่งจะเหลือ Dissolved Gas อยู่ น้อยมาก จะผ่านเข้าสู่ช่วงที่ 2 (Second Stage) ในส่วนนี้ประกอบไปด้วย Distributor หรือถาดซึ่งประกอบกันหลายชุด ตรงจุดนี้น้ำจะสัมผัสกันโดยตรงกับ Steam ซึ่งไม่มี Gas ละลายหรือเจือปนอยู่ด้วย Steam ที่ผ่านเข้าไปใน Stage นี้ จะถูกสัมผัสกับ Preheat Water จาก Stage ที่ 1 Steam จะเกิดการกลั่นตัว (Condense) เพียงเล็กน้อย และส่วนที่เหลือจะลอยขึ้นไปยัง Stage ที่ 1 และรวมกับ Gases ที่แยกตัวออกมาจาก Stage ที่ 1 ผ่านออกสู่บรรยากาศทาง Vent Steam ที่สูญเสียชีวิตกับการ Vent นี้ น้อยมาก และการ Vent นี้ จะดำเนินต่อไปทุกขณะ

น้ำร้อนที่ออกจาก Stage ที่ 2 นี้ จะไหลลงสู่ Storage Tank ที่อยู่ด้านล่างของ Deaerator เพื่อนำไปเก็บสำรองเพื่อป้อนให้กับ Boiler ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การปฏิบัติเพื่อให้อายุการใช้งานของ Deaerator ยาวนาน

1. อัตราการเปลี่ยนแปลงอุณหภูมิของเปลือก หรือใน Water Box จะต้องไม่เกิน  $400^{\circ}\text{F}$  ต่อ ชม.

2. การเปลี่ยนแปลงของ Pressure ใน Heater ต้องเป็นไปทีละน้อย การเปลี่ยนแปลงของ Pressure จะสัมพันธ์กับการเปลี่ยนแปลงของอุณหภูมิ

3. Cold Start Up จะเกิด Stress ขึ้นกับ Deaerator ควรหลีกเลี่ยงการปล่อย Steam ที่อุณหภูมิ  $500^{\circ}\text{F}$  เข้าไปใน Deaerator เพราะจะทำให้เกิด Thermal Shock ควรมีช่วงเวลา Warm Up โดยการเปิดให้ Steam เข้าไปอย่างทีละน้อย และเปิด Vent ไม่ให้มี Flow เข้าไปใน Water Box ความร้อนของ Steam ที่จะเป็นตัว Warm Up Shell ของ Deaerator ให้ร้อนขึ้น จะต้องอยู่ในช่วง  $50^{\circ}\text{F}$  ต่อนาที และจะอุ่นให้ร้อนขึ้นมาถึง  $200^{\circ}\text{F}$  น้ำใน Storage Tank ก็จะถูกอุ่นให้ร้อนขึ้นมาใน Rate เดียวกันนี้

เมื่ออุณหภูมิได้แล้ว ก็ให้เปิด Steam ที่เข้ามาพร้อมคัรบรอ จนกระทั่งไอของ Steam ถูก Vent ออกหมด ถึงขั้นนี้ก็สามารถเริ่มการ Start Up ตามขั้นตอนได้

4. การ Cool Down สำหรับการ Maintenance ถ้าใช้น้ำในการ Cool Down น้ำที่จะใช้ต้องมีอุณหภูมิต่ำกว่า Metal Temp  $100 - 150^{\circ}\text{F}$  จนกระทั่ง Metal Temp ลดลงมาถึง  $250^{\circ}\text{F}$  อัตราการเย็นตัวของ Metal Temp ควรอยู่ในช่วง  $100^{\circ}\text{F}$  ต่อชม. หลังจาก Metal Temp ลดลงต่ำกว่า  $250^{\circ}\text{F}$  แล้ว น้ำที่อุณหภูมิ  $60 - 70^{\circ}\text{F}$  สามารถนำมาใช้ได้

## Condenser Air Extraction

### หน้าที่ของระบบ

Condenser Air Extraction System ทำหน้าที่ดึงเอาอากาศ และรักษาการเป็น Vacuum ภายใน Condenser

Vacuum ภายใน Condenser มีไว้เพื่อให้ Turbine LP. Exhaust Steam, LP. และ HP. Bypass Steam, Steam Drain และ Water Drain ต่าง ๆ ที่ไหลลง Condenser แล้ว ถ่ายเทความร้อนให้กับ Condenser Tube Heat Exchanger ทำให้ Steam เกิดการควบแน่น (Condense) กลับตัวเป็นน้ำอย่างรวดเร็ว และค่อเนื่อง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Condenser

### Condenser

คือ Heat Exchanger ขนาดใหญ่ ซึ่งจะถ่ายเทความร้อน จาก Fluid ชนิดหนึ่งไปยัง Fluid อีกชนิดหนึ่ง ซึ่งในที่นี้ เราจะนำน้ำ Circulating Water มารับความร้อนจาก Steam โดยที่ Steam เป็นตัวถ่ายเทความร้อน แล้วกลั่นตัวเป็นหยดน้ำ

### ส่วนประกอบหลัก

- Inlet & Outlet Water Box
- Reverse Water Box
- Condenser Tube
- Condenser Hot-Well

### หน้าที่

- รักษา Press ของ Turbine Exhaust ให้ต่ำกว่าความดันบรรยากาศ เพื่อใช้ประโยชน์จากพลังงานให้มากที่สุด
- นำน้ำที่ Condense จาก Steam มาใช้งาน <Close Loop>
- แยก Non-Condensable Gas ออกจากระบบ

### การทำงาน

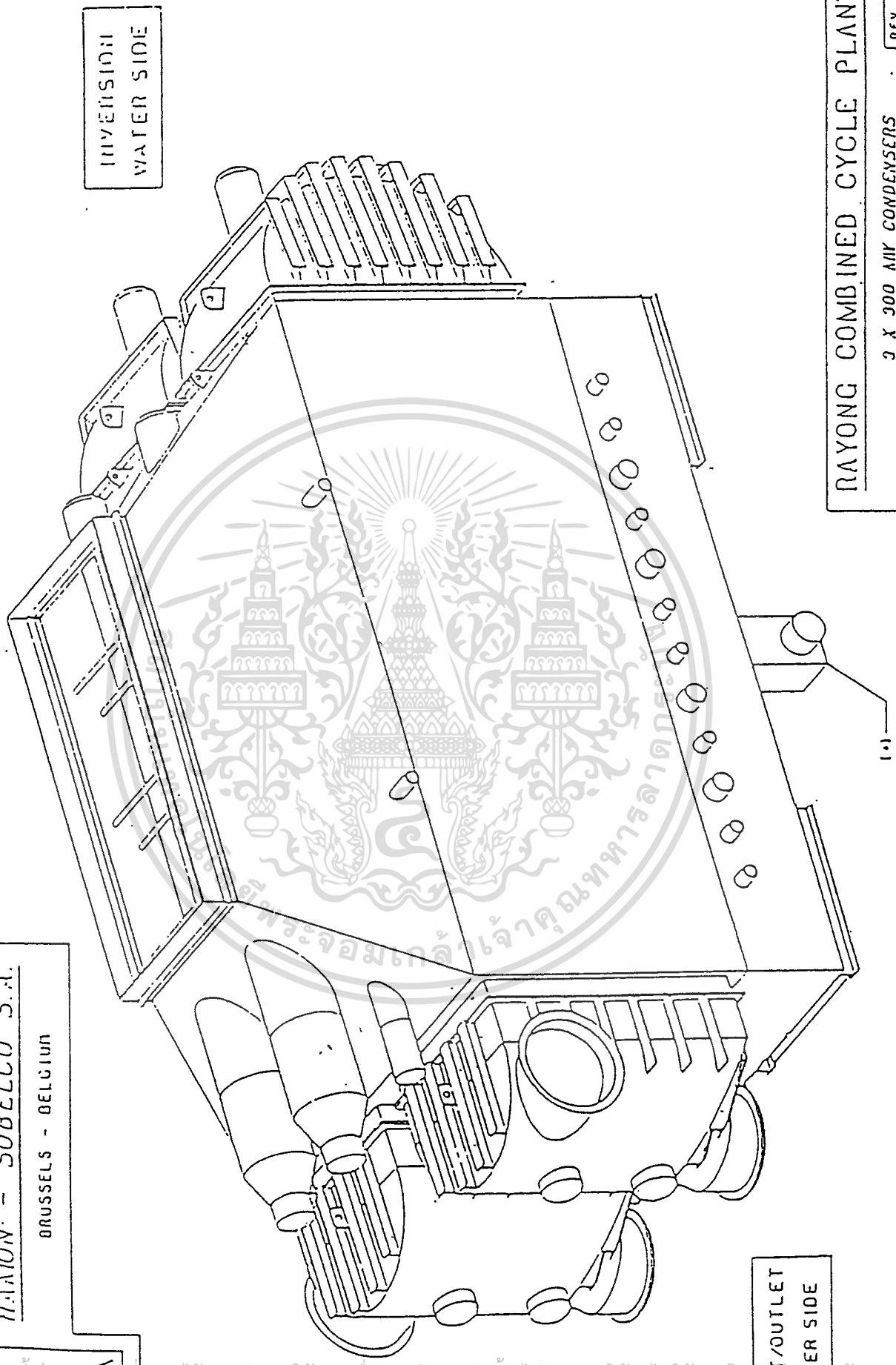
น้ำจาก CWP จะเข้ามาระบายความร้อนให้กับ Condenser โดยเข้าทางด้าน Inlet Water Box ผ่าน Condenser Tube ผ่าน Reverse Water Box แล้วผ่าน Outlet Water Box กลับสู่ Condenser Tube

ส่วน Steam ที่ออกจาก Turbine ก็จะไปไหลผ่าน Condenser Tube แล้วกลั่นตัวเป็นหยดน้ำ ที่เรียกว่า Condensate ซึ่งจะหยดลงไปอยู่ที่ส่วนล่างของ Condenser ที่เรียกว่า Hot-Well เพื่อเป็น Suction ให้กับ Condensate Pump และการที่ Steam กลั่นตัวเป็นหยดน้ำใน Condenser ก็เนื่องจาก Circulating Water Circulating Water จึงเป็นหลักใหญ่ที่ทำให้เกิด Vacuum ใน Condenser

RAYONG COMBINED CYCLE PLANT  
3 X 300 MW CONDENSERS  
REV. 1

รูปทว. 26

MAISON - SOBELCO S.A.  
BRUSSELS - BELGIUM



INLET/OUTLET  
WATER SIDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีงานนำไปใช้

## Circulating Water System

### หน้าที่ของระบบ

Circulating Water System เป็นระบบ Closed Cycle Cooling Water Heat Exchangers เพื่อทำหน้าที่ระบายความร้อนให้กับระบบดังกล่าวแล้ว ก็จะนำความร้อนมาถ่ายเทออก หรือถูกทำให้เย็นลงที่ตัว Cooling Tower

### Cooling Towers and Basin

ที่ตัว Cooling Tower จะมีชุด Cooling Tower Riser อยู่ 5 ชุด ซึ่งแต่ละชุดสามารถแยกการทำงานออกจากกันได้ โดยผ่าน Butterfly Valve เป็นการนำเอาน้ำที่ใช้งานแล้ว มาทำให้เย็นลง และเมื่อมีการรั่วไหลของ Circulating Water ออกนอกระบบ เราสามารถเติมเข้าระบบได้ที่ตัว Cooling Tower Basin โดยผ่านทาง Dia Phragm Operated Valve

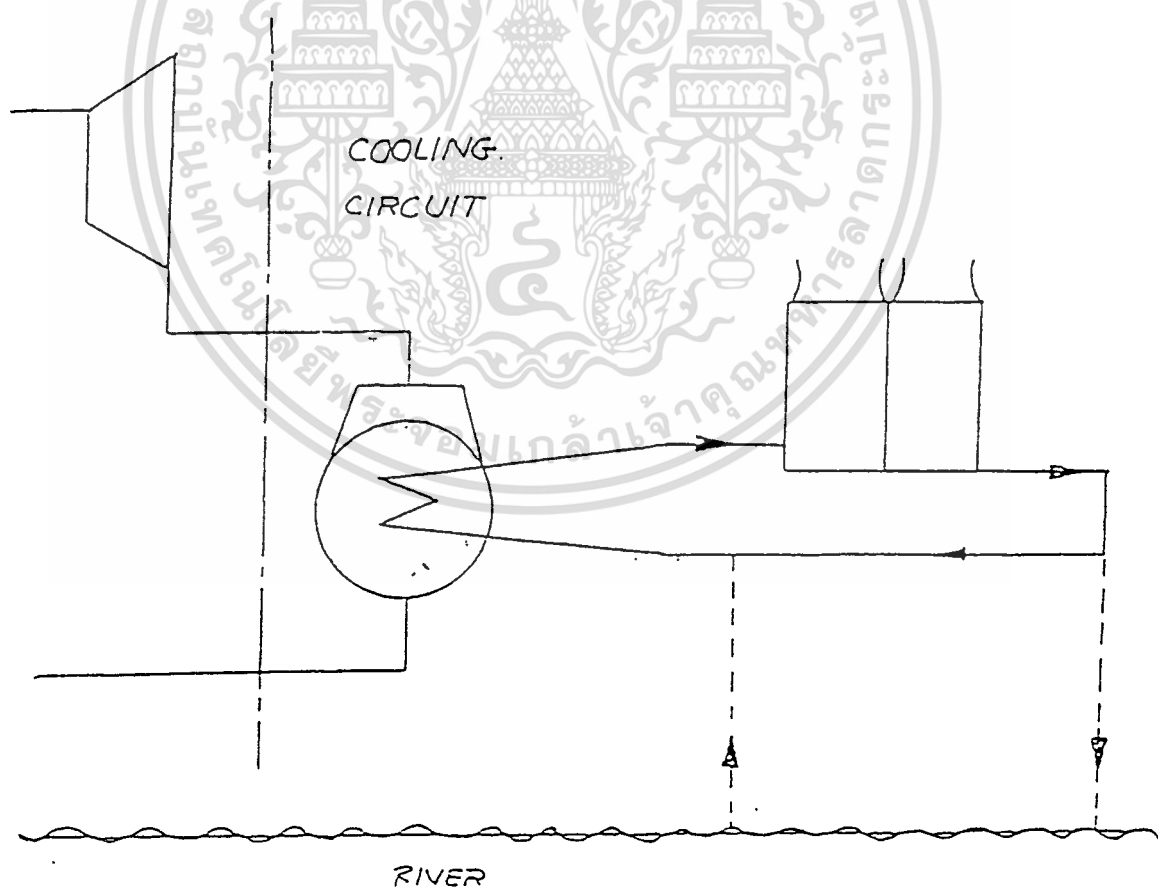
ทางด้าน Discharge ของ Pump จะมี Blow Down Control Valve เป็นตัวควบคุมการปล่อยน้ำเสีย ซึ่งผ่านขบวนการแยกสารเคมีแล้ว และส่งผ่านไปยัง Waste Water Collection ผ่านขบวนการแปรสภาพน้ำ (Water Treatment) ก่อนที่จะปล่อยทิ้งไป

## Cooling Tower

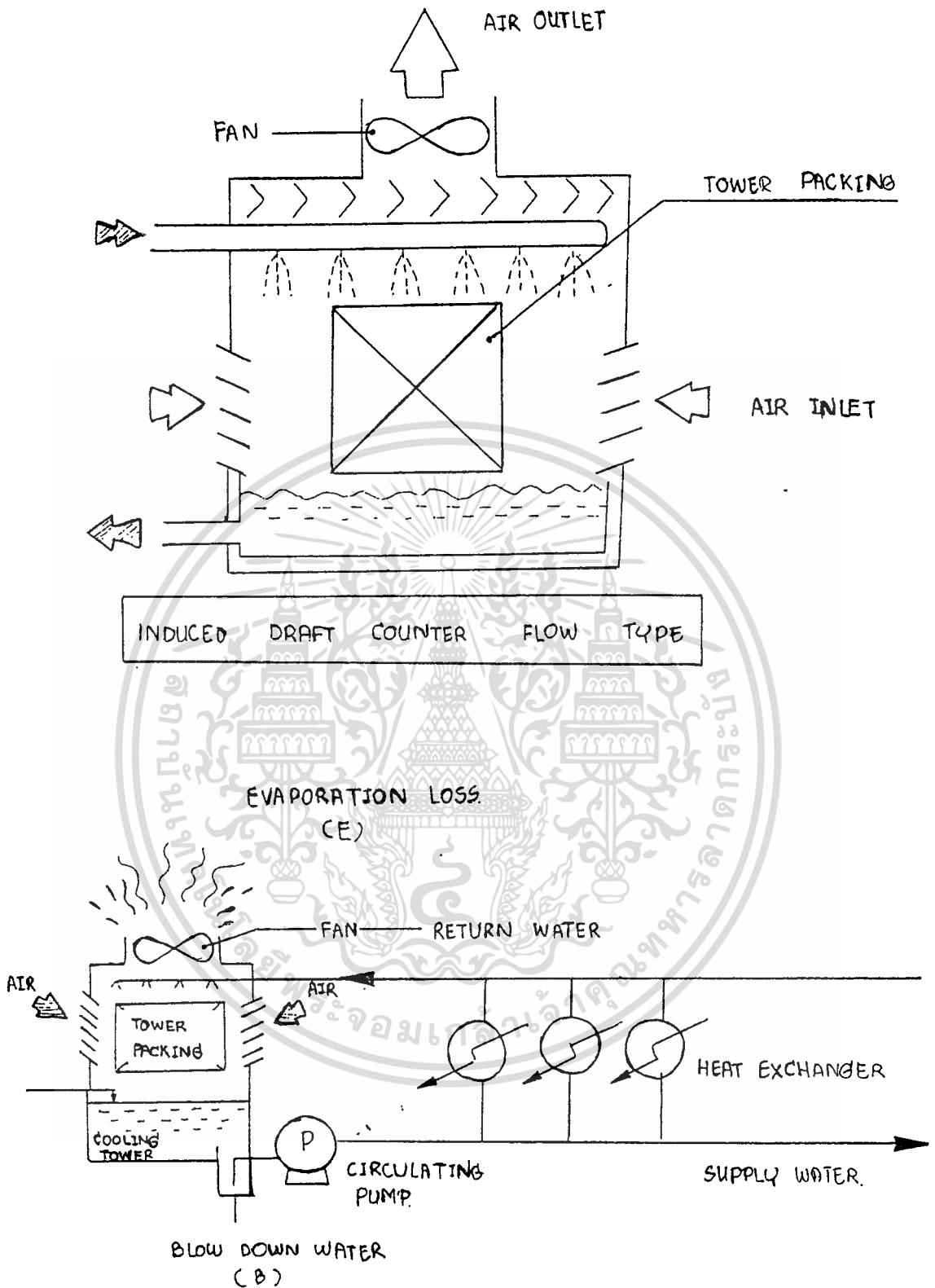
หน้าที่

ทำหน้าที่ถ่ายเทความร้อน (Heat Exchange) ระหว่าง Circulating Water กับ อากาศ

Circulating Water หลังจากนำไปใช้งานใน Condenser เพื่อใช้ในการควบแน่น (Condense) Steam จาก Steam Turbine และ Steam Drain ต่าง ๆ Circulating Water จะดึงความร้อน จาก Steam อุณหภูมิของน้ำจะสูงขึ้นไหลกลับมายัง Cooling Tower เพื่อถ่ายเทความร้อนให้กับ อากาศ แล้วไหลกลับไปยัง Condenser เป็น Closed Cycle



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ข้อมูลนี้โดยไม่ได้รับอนุญาตจากศูนย์ฯ ของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.28 WATER FLOW AND OPEN RECIRCULATING COOLING WATER SYSTEM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Circulating Water Pump Seal

### หน้าที่ของระบบ

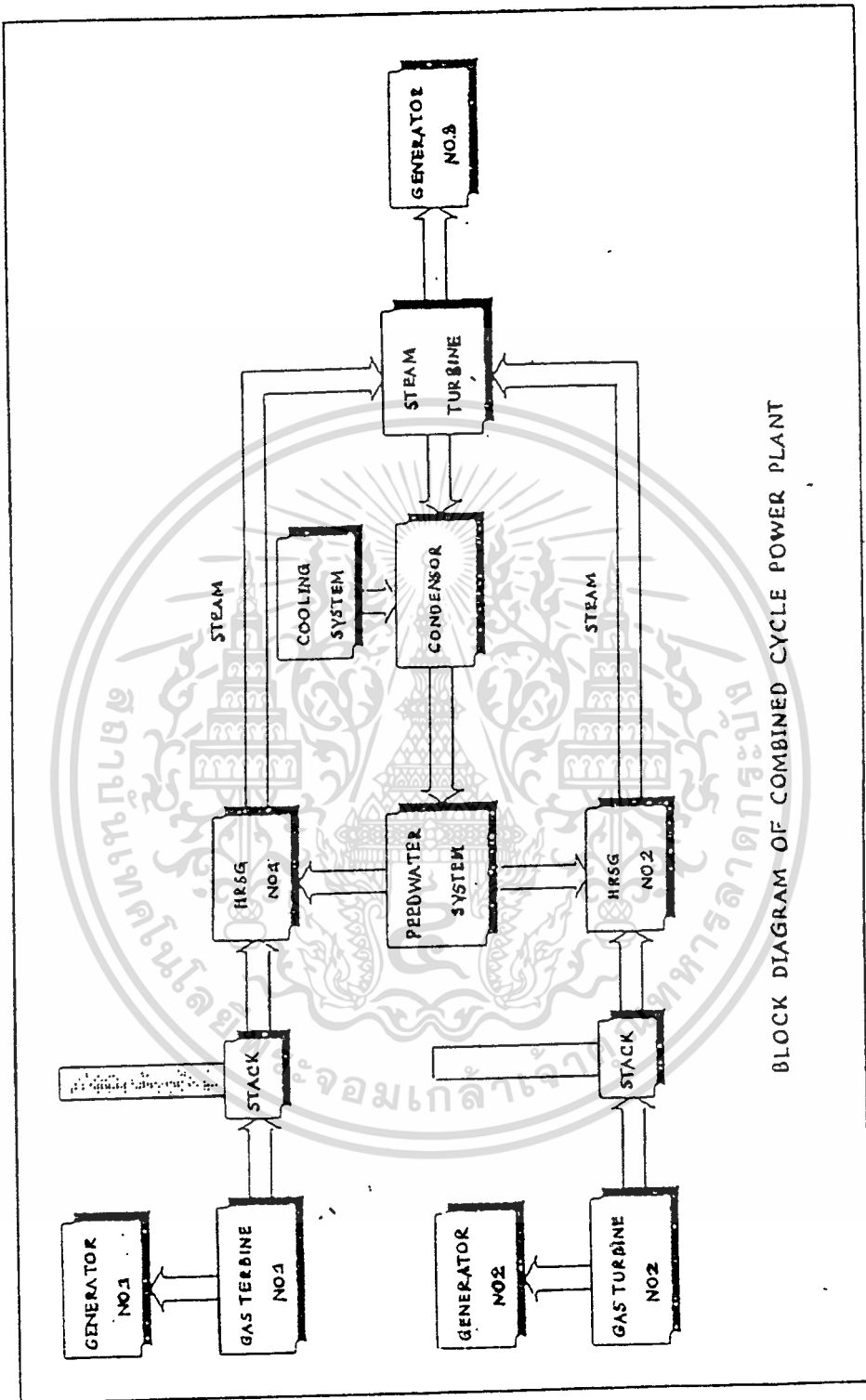
ทำหน้าที่ Supply น้ำให้กับ Circulating Water Pump เพื่อใช้งานกับ

- Circulating Water Pumps Seal
- Circulating Water Pumps Motor Cooling

Seal and Cooling Water ต้องมีเพียงพอเพื่อ Seal คอเพลลาของ Pump และถ่ายเทความร้อนจากการเสียดสีของ Mechanical Seal กับเพลลาของ Pump และมีเพียงพอสำหรับหล่อเย็นให้กับระบบน้ำมันของ Motor Bearing



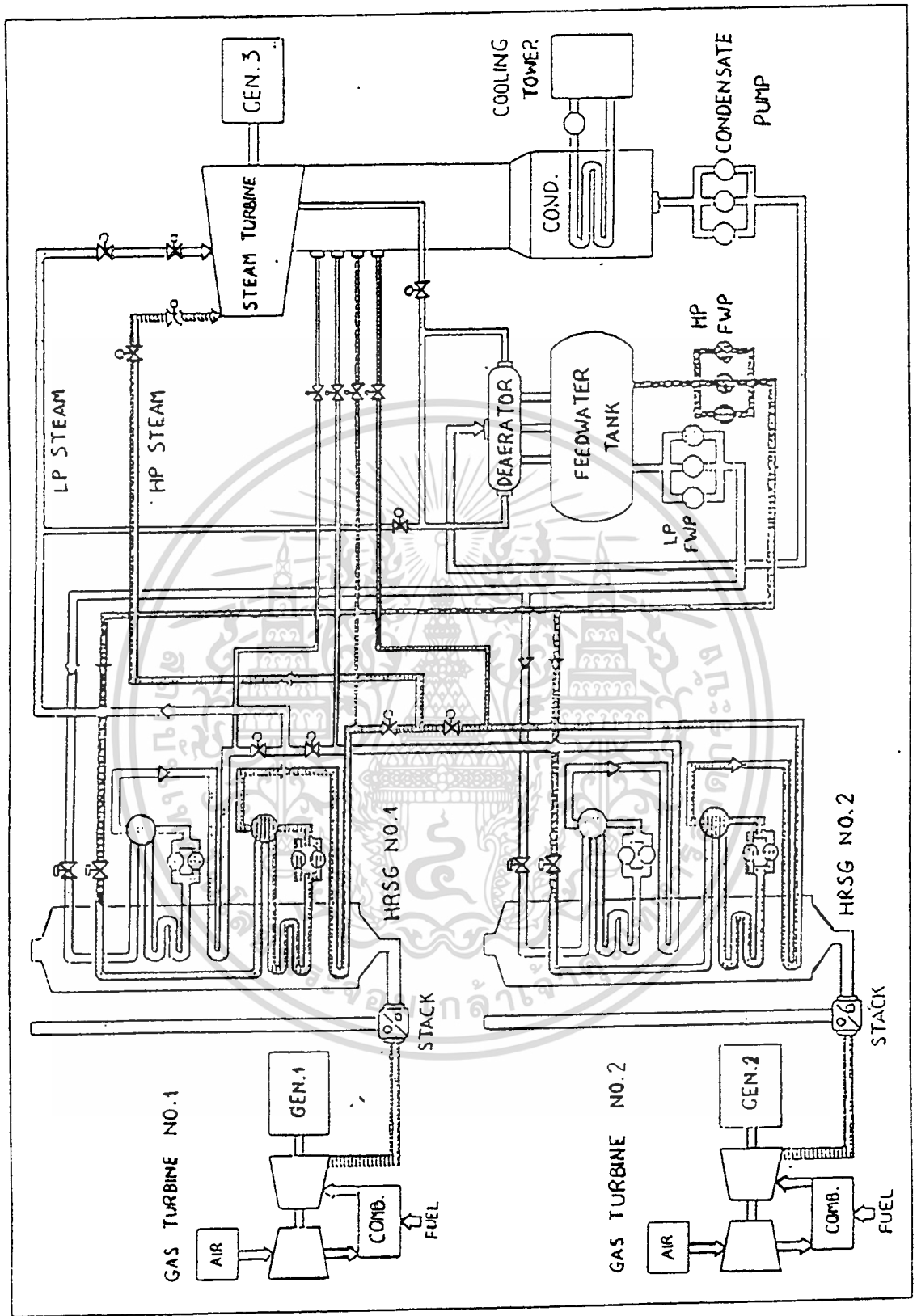
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



BLOCK DIAGRAM OF COMBINED CYCLE POWER PLANT

รูปที่ 4.1 แสดงบล็อกไดอะแกรมของโรงไฟฟ้าพลังความร้อนร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

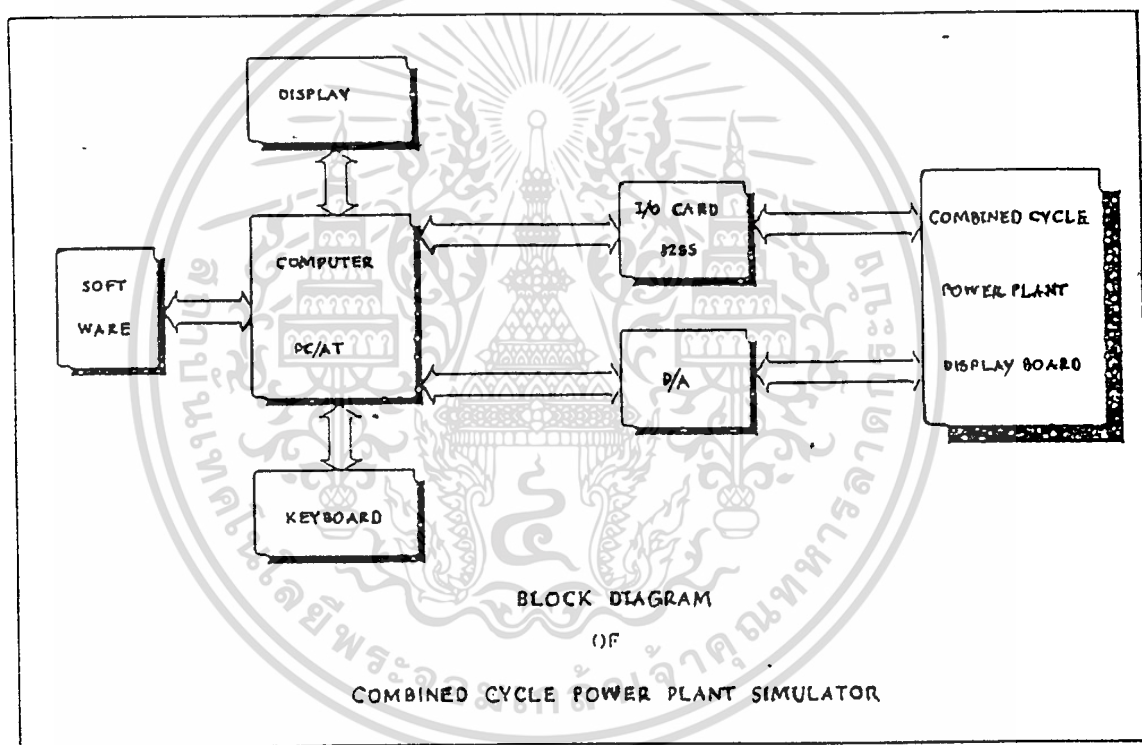


รูปที่ 4.2 แสดงโครงสร้างของโรงไฟฟ้าพลังความร้อนร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานและเผยแพร่โดยไม่หวังผลตอบแทนในนโยบายด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1 หลักการทำงานของชุดสาธิตโรงไฟฟ้าพลังความร้อนร่วม

จากหลักการทำงานของโรงไฟฟ้าพลังความร้อนร่วม สามารถที่จะแสดงหลักการทำงานอย่างง่าย ๆ และแสดงถึงการเกิดความผิดพลาดของโรงไฟฟ้าพลังความร้อนร่วมได้ โดยผ่านทางชุดสาธิต ซึ่งจะถูกรับควบคุมการทำงานด้วยคอมพิวเตอร์ ซึ่งมีหลักการในการควบคุมชุดสาธิต ดังแสดงในรูปที่ 4.3

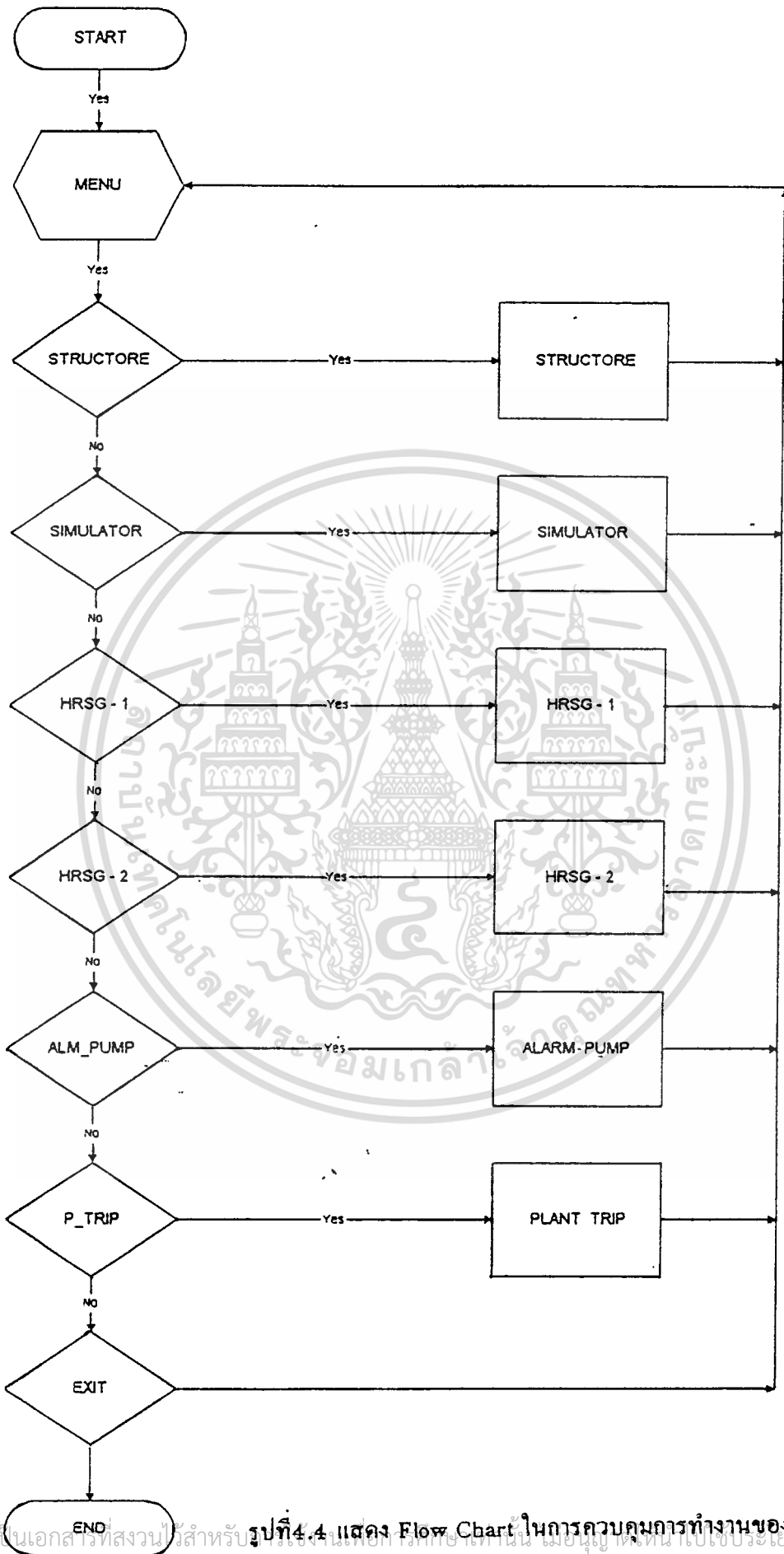


รูปที่ 4.3 แสดงหลักการควบคุมชุดสาธิตด้วยคอมพิวเตอร์

หลักการทำงานของชุดสาธิตโรงไฟฟ้าพลังความร้อนร่วม ทำการเขียนโปรแกรมเทอร์โบซีในการควบคุมการทำงานทั้งหมด ซึ่งจะมีเมนูให้เลือกดูการทำงานแต่ละส่วนของโรงไฟฟ้าพลังความร้อนร่วม พร้อมทั้งจำลองสภาวะการทำงานผิดปกติของโรงไฟฟ้า ดังแสดงในรูปที่

4.4 ซึ่งการทำงานสามารถแบ่งได้ 7 ส่วน จากเมนูที่แสดงหน้าจอคอมพิวเตอร์ ซึ่งมีรายละเอียดดังนี้

ไม่ปรากฏมีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบุคลากรในโรงงานเท่านั้น เมื่ออนุญาตให้พิมพ์ไปใช้ประโยชน์ในการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1. STRUCTOR สามารถเลือกดูการแสดงรูปโดยการกด STRUCTOR ซึ่งโปรแกรมจะแสดงรูปแบบการติดตั้งระหว่าง COMPUCTOR , SOFEWARE และ HARDWARE เข้าด้วยกันเพื่อทำการติดต่อสื่อสารกัน

4.1.2. SIMULATOR สามารถเลือกดูการทำงานโดยการกด SIMULATOR ซึ่งโปรแกรมจะแสดงการทำงานครบทั้งระบบ โดยที่ GAS-1 และ GAS-2 ผลิตพลังงานไฟฟ้าได้ 100 MW , GAS-2 ผลิตพลังงานได้ 100 MW ดังนั้น STEAM TURBINE จะผลิตได้ 100 MW ค่าย ซึ่งรวมทั้งระบบได้เท่ากับ 300 MW เทียบเท่ากับ 100 % เต็ม

4.1.3. HRSG-1 สามารถเลือกดูการทำงานโดยการกด HRSG-1 ซึ่งโปรแกรมจะแสดงการทำงานของชุดสาริตพร้อมกันกับที่หน้าจอกอมพิวเตอร์ ตามขั้นตอนการทำงานของโรงไฟฟ้าพลังความร้อนร่วม ซึ่ง EXHAUST GAS สามารถนำไปขับ STEAM TURBINE จ่าย LOAD ได้ 50 %

4.1.4. HRSG-2 สามารถเลือกดูการทำงานโดยการกด HRSG-2 ซึ่งโปรแกรมจะแสดงการทำงานของชุดสาริตพร้อมกันกับที่หน้าจอกอมพิวเตอร์ ตามขั้นตอนการทำงานของโรงไฟฟ้าพลังความร้อนร่วม ซึ่ง EXHAUST GAS สามารถนำไปขับ STEAM TURBINE จ่าย LOAD ได้ 50 %

4.1.5. ALARM ALL สามารถเลือกดูการทำงานโดยการกด ALARM ALL ซึ่งจะแสดงในสภาวะ ที่ FEED WATER PUMP (FWP) ของ LP และ HP เกิดการทำงานผิดปกติ ดังนั้น FWP ที่ทำหน้าที่สำรองอยู่ จะต้องทำงานทันทีเมื่อมี ALARM เกิดขึ้น ALARM DRUM ซึ่งจะแสดงการทำงานในสภาวะ ที่ CIRCULATING PUMP ทำงานไม่ได้ ดังนั้น CIRCULATING PUMP ที่ทำหน้าที่สำรองจะต้องทำงานแทนทันที ALARM CONDEN ซึ่งจะแสดงการทำงานในสภาวะที่ CONDENSATE PUMP ทำงานไม่ได้ 1 ตัว ดังนั้น CONDENSATE PUMP ที่ทำหน้าที่สำรองจะต้องทำงานแทนทันที

4.1.6. PLANT TRIP สามารถเลือกดูการทำงานโดยการกด P-TRIP ซึ่งโปรแกรมจะแสดงการ TRIP ของโรงไฟฟ้า ในกรณีที่ CIRCULATING PUMP ใช้งานไม่ได้ทั้งหมด ซึ่งจะทำให้ไม่มี STEAM ไปขับชุด STEAM TURBINE ดังนั้น เราจึงต้องหยุดเดินเครื่องในส่วนของ STEAM TURBINE แต่ในส่วนของ GAS-1 และ GAS-2 ยังสามารถผลิตพลังงานไฟฟ้าต่อไปได้

4.1.7. EXIT เป็นการออกจากการทำงานของโปรแกรม โดยการกด EXIT หรือ กด Esc.

## 4.2 ลำดับชั้นการทำงานของโรงไฟฟ้าพลังความร้อนร่วม

1. เดินเครื่อง Gas Turbine เครื่องที่หนึ่งหรือที่สอง หรือเดินทั้งสองเครื่องพร้อมกัน โดยการดูดอากาศผ่าน Air Inlet และทำการอัดอากาศที่ Compressor เพื่อนำมาผสมกับเชื้อเพลิงแล้วทำการเผาไหม้ที่ Combustion ทำให้ได้แก๊สร้อนนำไปขับ Turbine ที่ต่อเข้าโดยตรงกับ Generator ซึ่งจะเป็นเครื่องผลิตกระแสไฟฟ้า ส่วนแก๊สเสียหรือ Exhaust Gas ซึ่งยังมีความร้อนสูงจะถูกปล่อยออกไปทาง Stack

2. เมื่อต้องการเดินเครื่อง Steam Turbine จะต้องทำการ Pump น้ำจาก Feed Water Tank ซึ่งจะทำหน้าที่เก็บน้ำที่มีอุณหภูมิพอเหมาะ เพื่อส่งผ่านไปสู่อุปกรณ์ Boiler สาเหตุที่ต้องอุ่นน้ำ เพราะ น้ำที่เย็น (ไม่มีคาร์บอน) จะทำให้เกิดการเสียหายของ Boiler โดยผ่านทาง HP FWP และ LP FWP (เป็นมอเตอร์ที่ทำการปัมน้ำจาก Feed Water Tank) นำน้ำมาเก็บไว้ใน HP Drum และ LP Drum ของ HRSG ทั้งสองเครื่อง ให้ได้ระดับน้ำใน Drum ประมาณ 50 %

3. ขั้นตอนนี้จะต้องทำการปฏิบัติการพร้อมกัน 3 ส่วน คือ Damper, Circulating Pump, Cooling Pump ซึ่งสามารถอธิบายการทำงานในแต่ละส่วนได้ดังนี้

3.1 ส่วนของ Damper จะทำการเปิดเพื่อ นำแก๊สร้อนหรือ Exhaust Gas ผ่านเข้ามาทาง HRSG โดยจะเปิดให้มีความสัมพันธ์กับการทำงานของ Circulating Pump

3.2 ส่วนของ Circulating Pump จะทำการ Pump น้ำจาก Drum ไปตามท่อของ Water Wall เมื่อน้ำได้รับความร้อนจาก Exhaust Gas ก็จะกลายเป็นน้ำร้อนไหลไปยัง Drum อีกครั้งโดยน้ำร้อนจะอยู่ด้านล่าง และไอน้ำจะอยู่ด้านบนของ Drum

3.3 ส่วนของ Cooling Pump จะทำการ Pump น้ำเย็นจาก Cooling Tank ผ่านเข้าไปในส่วนของ Condenser เพื่อทำให้อไอน้ำกลั่นตัวเป็นน้ำ ซึ่งในการเดินเครื่องช่วงแรกนี้ไอน้ำที่ได้ยังไม่มีแรงดัน และอุณหภูมิไม่เพียงพอ ดังนั้นจึงต้องทำการปิดวาล์วที่ Steam Turbine และเปิด By-Pass Value เพื่อให้ไอน้ำที่ได้จาก Drum ผ่านมายัง Condenser โดยตรงเมื่อไอน้ำผ่านความเย็น ก็จะกลั่นตัวเป็นน้ำสะสมอยู่ที่ Hot Well

4. Condensate Pump จะทำการ Pump น้ำจาก Hot Well (ทำหน้าที่เก็บน้ำที่เกิดจากไอน้ำแล้วกลั่นตัวเป็นน้ำ) เพื่อนำไปเก็บ Water Tank แต่น้ำที่ได้จาก Hot Well ยังมีอุณหภูมิ

ต่ำ จึงต้องมีการเพิ่มอุณหภูมิ โดยการนำเอา LP Steam มาผสมกับน้ำที่ Deaerator ซึ่ง Deaerator จะทำหน้าที่เป็นตัวผสมน้ำให้มีอุณหภูมิที่อุ่นพอเหมาะ น้ำที่ได้จะมาจาก 3 ส่วน ดังนี้ คือ

- Hot well ที่ Condensate ป้อนมาให้ (สถานะเป็นน้ำ)
- LP Steam เป็นไอน้ำที่มีความร้อน
- Steam Turbine เป็นน้ำที่ได้จากกังหันปั่นไอน้ำที่มีความร้อนภายใน Steam LP Steam ผสมน้ำที่ Deaerator ทำให้น้ำที่ได้อุณหภูมิสูงขึ้นถูกนำมาเก็บไว้ใน Feedwater Tank เพื่อที่จะทำการ Pump น้ำไปยัง Drum ต่อไป

5. เมื่อไอน้ำที่ได้จาก Drum มีแรงดันและอุณหภูมิที่เพียงพอแล้ว By-Pass Valve ถูกเปิด และวาล์วที่ Steam Turbine จะถูกเปิดเพื่อนำไอน้ำที่ได้ไปขับ Steam Turbine ต่อเข้าโดยตรงกับ Generator ทำให้เกิดการผลิตกระแสไฟฟ้า ส่วนของไอน้ำที่ผ่าน Turbine ก็จะไปผ่านมายัง Condenser เพื่อทำให้น้ำกลั่นตัวกลายเป็นน้ำ และสะสม Well และน้ำที่ได้นี้จะถูก Pump กลับไปยัง Feedwater Tank อีกต่อไป

6. ขั้นตอนนี้ ระบบการทำงานของโรงไฟฟ้าพลังความร้อนร่วม จะทำงานพร้อมคือน้ำที่อยู่ใน Feedwater Tank จะถูก Pump ไปยัง Drum และน้ำใน Drum จะกลับเข้าไปในท่อของ Water Wall เมื่อน้ำในท่อได้รับความร้อนจาก Exhaust Gas กลับไปยัง Drum อีกครั้ง ส่วนของไอน้ำที่ได้จาก Drum จะถูกนำมาผ่านความร้อนที่ Heater อีกครั้ง เพื่อทำให้ไอน้ำปราศจากความชื้นและอุณหภูมิสูงขึ้น ไอน้ำที่ได้นี้จะถูก Steam Turbine ทำให้ Generator หมุนและผลิตกระแสไฟฟ้าขึ้นมา หลังจากไอน้ำ Steam Turbine ทำให้ Generator หมุนและผลิตกระแสไฟฟ้าขึ้นมา หลังจาก Well น้ำที่เก็บอยู่ใน Hot Well จะถูก Pump มาผสมกับ LP Steam ที่ Deaerator ทำให้อุณหภูมิของน้ำสูงขึ้น และน้ำที่ได้จะถูกเก็บไว้ใน Feedwater Tank และน้ำจะถูก Drum เป็นลักษณะวนรอบอยู่เช่นนี้โดยตลอดไป

7. เมื่อโรงไฟฟ้าเกิดการ Trip หรือ Shut Down จะทำให้วาล์วที่ Steam Turbine ถูกปิด และ By-Pass Valve จะถูกเปิดออกเพื่อให้ไอน้ำค้างอยู่ในท่อผ่านออกมาที่ SOR โดยตรง แล้วกลั่นตัวเป็นน้ำสะสมอยู่ใน Hot Well

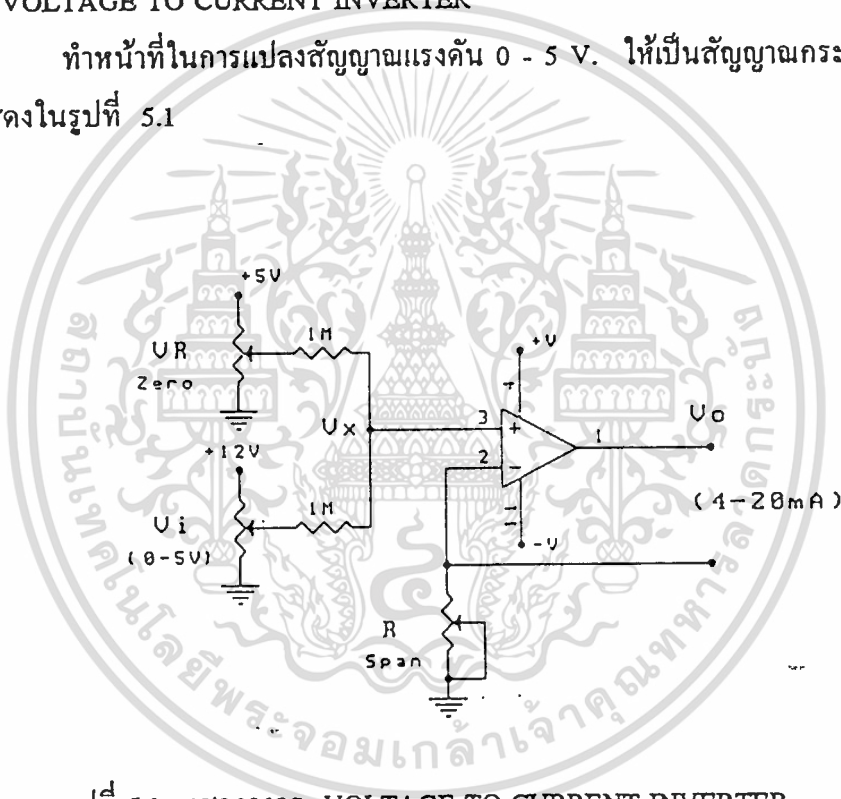
## บทที่ 5

### หลักการทํางานของวงจร INVERTER

วงจร INVERTER ที่ใช้ในการแปลงสัญญาณแรงดันและกระแส ประกอบด้วย วงจร 2 ส่วน คือ วงจร VOLTAGE TO CURRENT INVERTER และ CURRENT TO VOLTAGE INVERTER ซึ่งวงจรแต่ละส่วนมีรายละเอียดดังต่อไปนี้

#### 5.1 วงจร VOLTAGE TO CURRENT INVERTER

ทำหน้าที่ในการแปลงสัญญาณแรงดัน 0 - 5 V. ให้เป็นสัญญาณกระแส 4 - 20 mA. ดังแสดงในรูปที่ 5.1



รูปที่ 5.1 แสดงวงจร VOLTAGE TO CURRENT INVERTER

จากวงจรในรูปที่ 5.1 สามารถคำนวณหาค่ากระแส 4 - 20 mA. ได้ดังนี้

$$\begin{aligned} \text{จาก} \quad V_x &= \frac{V_R + V_I}{2} \\ \text{และ} \quad I &= \frac{V_x}{R} \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{ดังนั้น} \quad I = \frac{V_R + V_I}{2R} \quad \dots\dots (1)$$

จาก (1) จะได้ว่า

$$2R \cdot I_A = V_R + V_A \quad \dots\dots (2)$$

$$2R \cdot I_B = V_R + V_B \quad \dots\dots (3)$$

(3) - (2) จะได้ว่า

$$2R (I_B - I_A) = V_B - V_A$$

ดังนั้น

$$R = \frac{V_B - V_A}{2(I_B - I_A)}$$

แทนค่า

$$\begin{aligned} V_A &= 0 \text{ V.} \\ V_B &= 5 \text{ V.} \\ I_A &= 4 \text{ mA.} \\ I_B &= 20 \text{ mA.} \\ &= \frac{(5 - 0) \text{ V.}}{2(20 - 4) \text{ mA.}} \\ R &= 156.25 \ \Omega \end{aligned}$$

แทนค่า R ในสมการที่ (2) จะได้ว่า

$$2(156.25 \ \Omega)(4 \text{ mA.}) = V_R + 0$$

$$\text{ดังนั้น} \quad V_R = 1.25 \text{ V.}$$

นำมาเขียนสมการเส้นตรง จะได้

จากสมการเส้นตรง

$$Y = MX + C$$

เมื่อ

$$M = \text{Slope}$$

$$C = \text{Offset}$$

จากสมการที่ ( 2 ) และ ( 3 )

$$I_A = \frac{V_A + V_R}{2R}$$

$$I_B = \frac{V_B + V_R}{2R}$$

จะได้ว่า

$$\frac{1}{2R} = \text{Slope}$$

$$\frac{V_R}{2R} = \text{Offset}$$

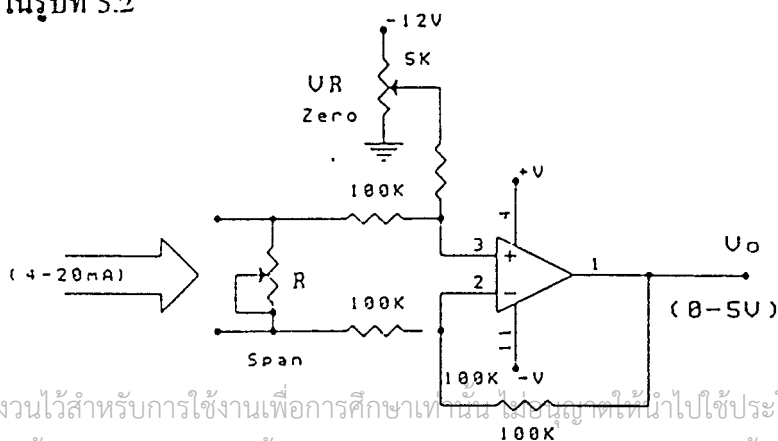
จากการคำนวณ ทำการป้อนแรงดัน 0 V. ให้กับวงจร และทำการปรับ  $V_R$

ให้ได้ค่าของ Offset มีค่ากระแส = 4 mA.

ทำการป้อนแรงดัน 5 V. ให้กับวงจร และทำการปรับ R ให้ได้ค่าของ Slope มีค่ากระแส = 20 mA.

## 5.2 วงจร CURRENT TO VOLTAGE INVERTER

ทำหน้าที่ในการแปลงสัญญาณกระแส 4 - 20 mA. ให้เป็นสัญญาณแรงดัน 0 - 5 V. ดังแสดงในรูปที่ 5.2



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
รูปที่ 5.2 แสดงวงจร CURRENT TO VOLTAGE INVERTER

จากวงจรในรูปที่ 5.2 สามารถคำนวณหาค่าแรงดัน 0 - 5 V. ได้ดังนี้

จาก 
$$V_o = \frac{R_F}{R_I} (V_I) + V_R$$

เมื่อ 
$$R_F = R_I$$

ดังนั้น 
$$V_o = V_I + V_R$$

และ 
$$V_I = I \cdot R$$

ดังนั้น 
$$V_o = (I \cdot R) + V_R \dots\dots\dots (1)$$

จาก (1) จะได้ว่า

$$V_A = (I_A \cdot R) + V_R \dots\dots\dots (2)$$

$$V_B = (I_B \cdot R) + V_R \dots\dots\dots (3)$$

(3) - (2) จะได้ว่า

$$V_B - V_A = R (I_B - I_A)$$

ดังนั้น

$$R = \frac{V_B - V_A}{I_B - I_A}$$

แทนค่า

$$V_A = 0 \text{ V.}$$

$$V_B = 5 \text{ V.}$$

$$I_A = 4 \text{ mA.}$$

$$I_B = 20 \text{ mA.}$$

จะได้

$$R = \frac{(5 - 0) \text{ V.}}{(20 - 4) \text{ mA.}}$$

$$R = 312.5 \Omega$$

แทนค่า R ในสมการที่ (2) จะได้ว่า

$$0 = (4 \text{ mA.}) (312.5 \Omega) + V_R$$

ดังนั้น 
$$V_R = -1.25 \text{ V.}$$
  
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำมาเขียนสมการเส้นตรง จะได้

จากสมการเส้นตรง

$$Y = MX + C$$

เมื่อ

$$M = \text{Slope}$$

$$C = \text{Offset}$$

จากสมการที่ ( 2 ) และ ( 3 )

$$V_A = (I_A \cdot R) + V_R$$

$$V_B = (I_B \cdot R) + V_R$$

จะได้ว่า

$$R = \text{Slope}$$

$$V_R = \text{Offset}$$

จากการคำนวณ ทำการป้อนกระแส 4 mA. ให้กับวงจร และทำการปรับ  $V_R$  ให้  
ได้ค่าของ Offset มีค่าแรงดัน = 0 V.

ทำการป้อนกระแส 20 mA. ให้กับวงจร และทำการปรับ R ให้ได้ค่าของ Slope  
มีค่าแรงดัน = 5 V.

## บทที่ 6

### รายละเอียดเกี่ยวกับ INTERFACE CARD

INTERFACE CARD ที่ต่อระหว่าง COMPUTER และชุด SIMULATOR แบ่งออกเป็น 2 ส่วน คือ

1. DIGITAL CARD
2. ANALOG CARD

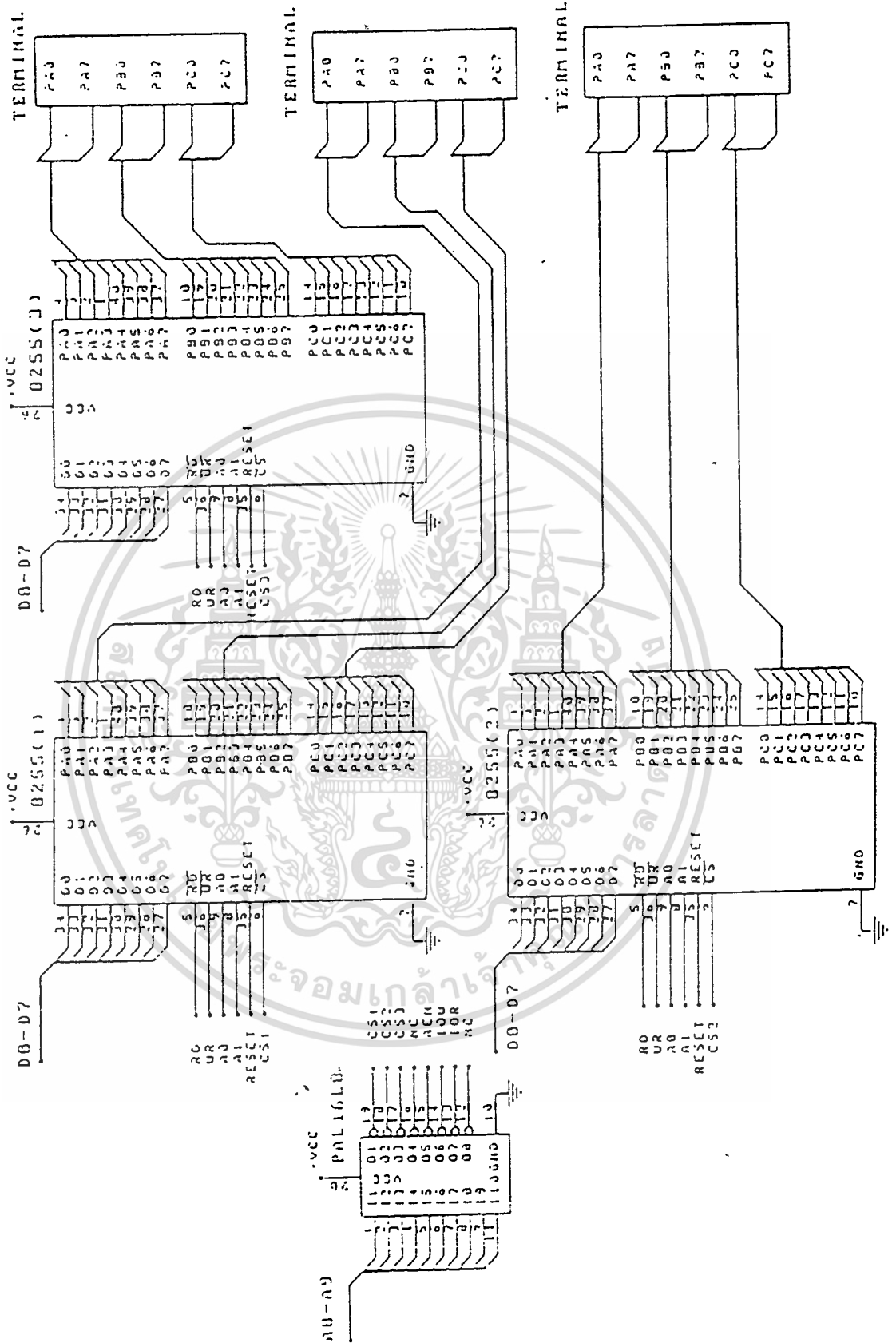
#### 6.1 DIGITAL CARD

วงจรของ DIGITAL CARD ดังแสดงในรูปที่ 6.1 ทำหน้าที่เชื่อมต่อระหว่าง COMPUTER กับชุด SIMULATOR เพื่อใช้ในการควบคุม LED จำนวน 33 ดวง และชุดของ 7-SEGMENT ซึ่งประกอบด้วย IC 8255 จำนวน 3 ตัว และ IC PAL16L8 จำนวน 1 ตัว โดยมีรายละเอียด ดังนี้

- > IC PAL16L8 ทำหน้าที่ในการ DECODE PORT
- > IC 8255 ทำหน้าที่ในการกำหนด PORT ที่ต่อใช้งานกับชุด SIMULATOR โดยแบ่งออกเป็นสามย่อยได้คือ

- IC 8255 ตัวที่ 1 ทำหน้าที่ควบคุมการใช้งาน PORT หมายเลข 300 , 301 , 302
- IC 8255 ตัวที่ 2 ทำหน้าที่ควบคุมการใช้งาน PORT หมายเลข 304 , 305 , 306
- IC 8255 ตัวที่ 3 ทำหน้าที่ควบคุมการใช้งาน PORT หมายเลข 308 , 309 , 30A

to display led.



## 6.2 ANALOG CARD

วงจรของ ANALOG CARD ดังแสดงในรูปที่ 6.2 ทำหน้าที่เชื่อมต่อระหว่าง COMPUTER กับชุด SIMULATOR เพื่อใช้ในการ CONTROL LED BARGRAPH ให้ทำหน้าที่แสดงระดับ PRESSURE ของน้ำ, ระดับ PRESSURE ของ STEAM และระดับของ LOAD ซึ่งมีรายละเอียด ดังนี้

- IC PAL16L8 ทำหน้าที่ในการ DECODE PORT หมายเลข 30C , 30D , 30E , 30F , 310 , 311 , 312
- IC AD7528 ทำหน้าที่ในการแปลงสัญญาณ DIGITAL เป็น ANALOG ระดับแรงดัน 0 - 5 Volt
- IC LM3403 (OP-AMP) ทำหน้าที่ในการแปลงแรงดัน 0-5 Volt เป็นกระแส 4 - 20 mA ซึ่งการทำงานชุดนี้เราเรียกว่า VOLTAGE TO CURRENT INVERTER
- IC AD581 ทำหน้าที่ในการสร้าง VOLTAGE REFERENCE -5 Volt ให้กับ IC AD7528

## 6.3 ส่วนของวงจรที่ต่อกับชุดสาริต

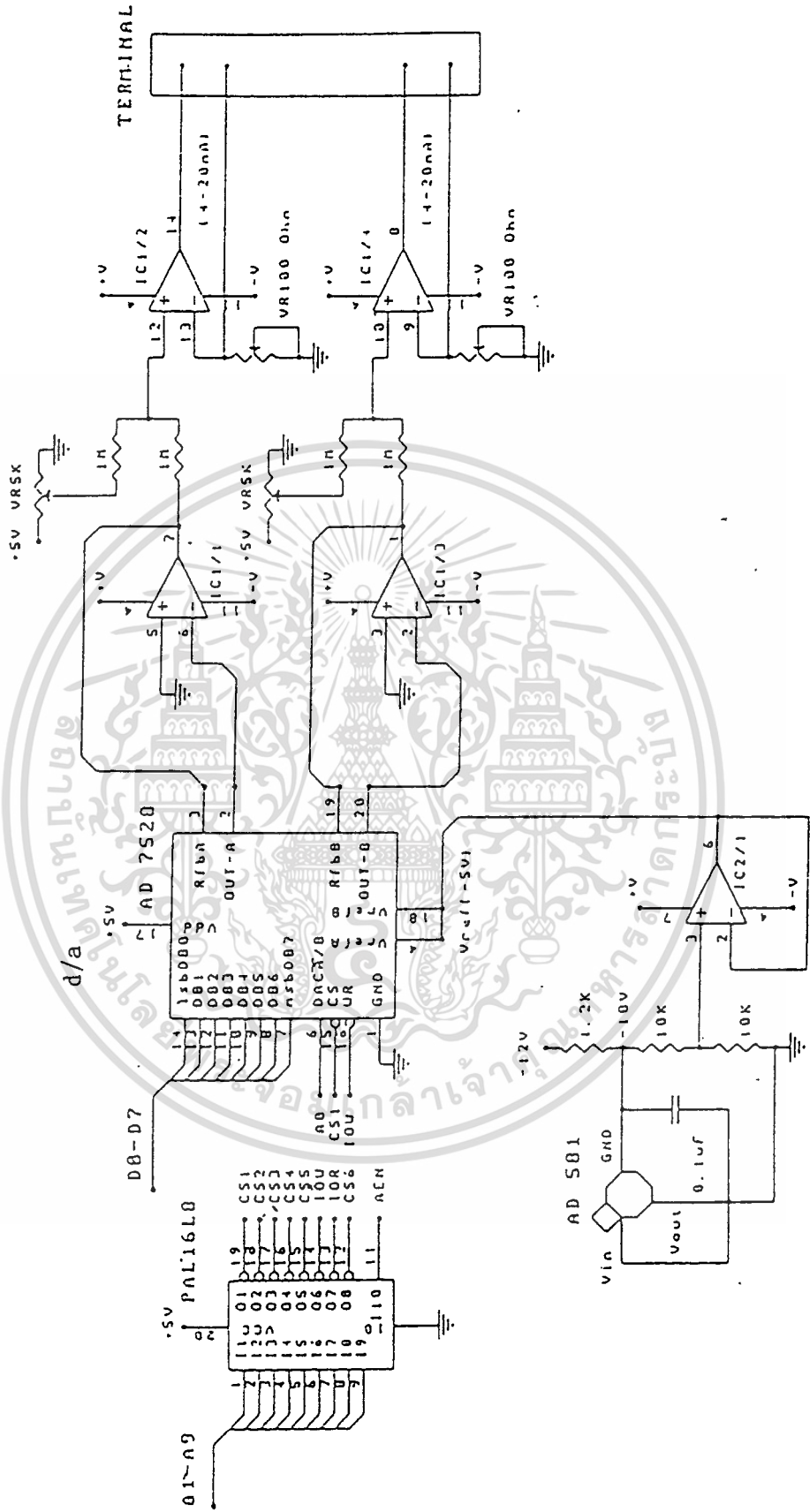
วงจรส่วนที่ต่อกับชุดสาริต ดังแสดงในรูปที่ 6.3 ประกอบด้วย 2 ส่วน คือ

### 6.3.1 CURRENT TO VOLTAGE INVERTER ทำหน้าที่ในการแปลง

กระแส 4 - 20 mA ที่ส่งออกมาจาก ANALOG CARD เป็นแรงดัน 0 - 5 Volt

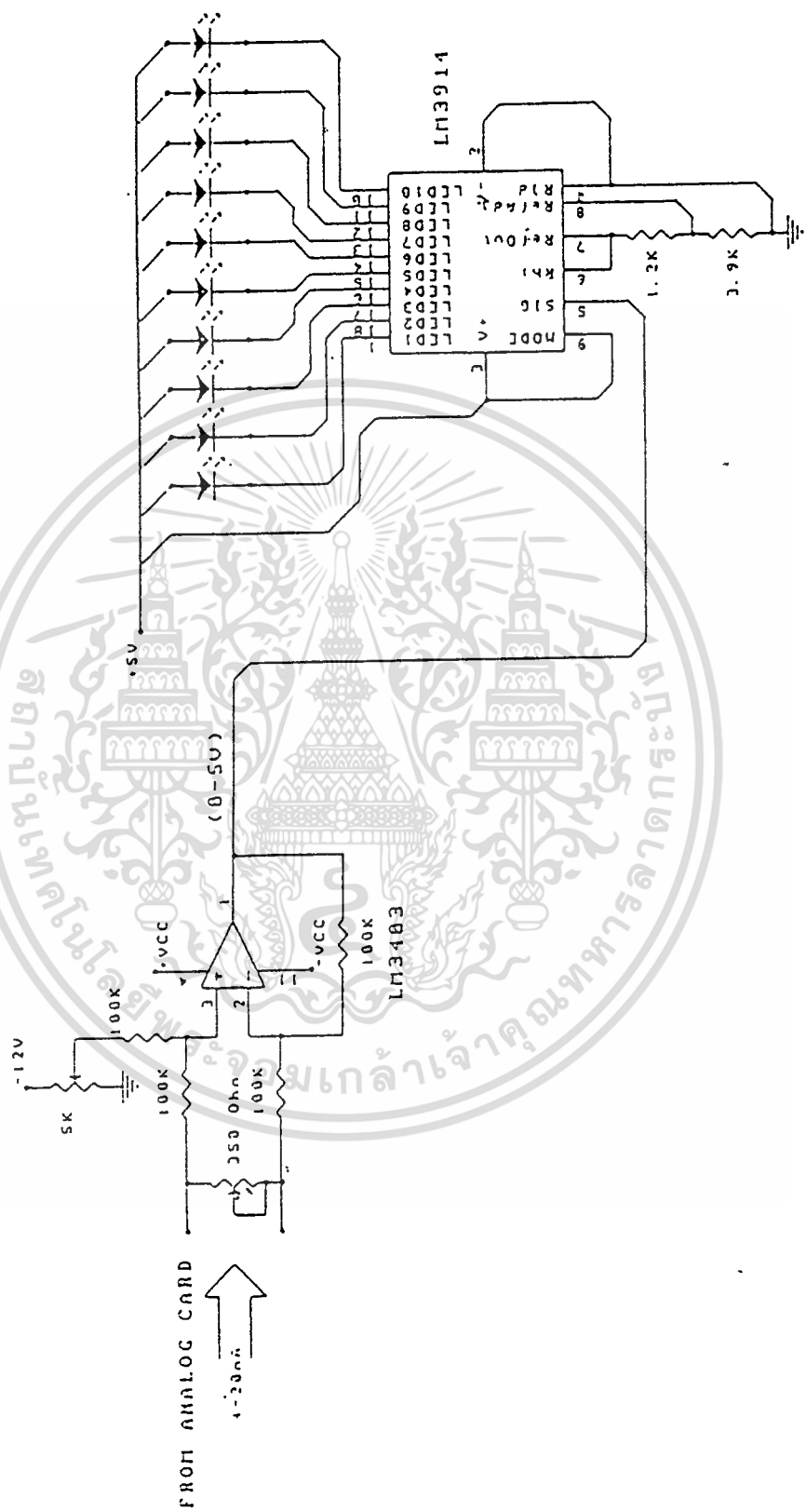
6.3.2 LED BARGRAPH DISPLAY DRIVER ทำหน้าที่ในการรับแรงดัน 0 - 5 Volt จาก CURRENT TO VOLTAGE INVERTER เพื่อนำ OUTPUT ไปขับ LED BARGRAPH โดยใช้ IC LM3914

CONVERSION OF CURRENT

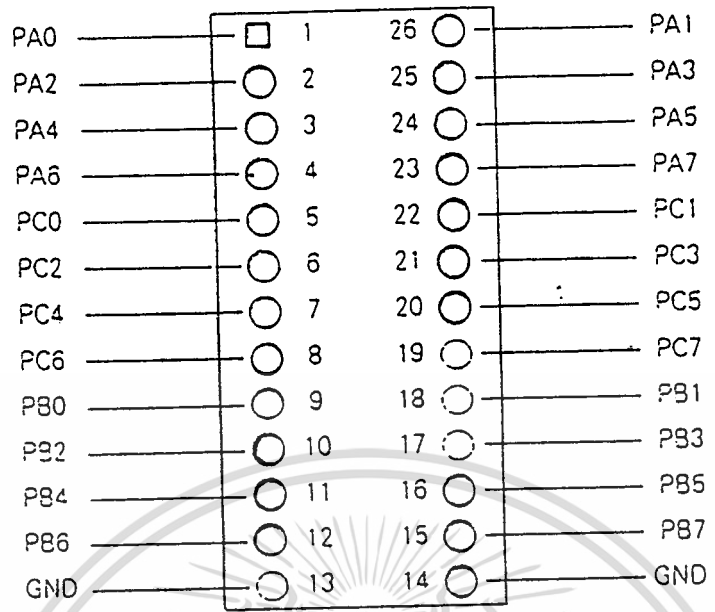


รูปที่ 6.2 แสดงวงจรของ ANALOG CARD

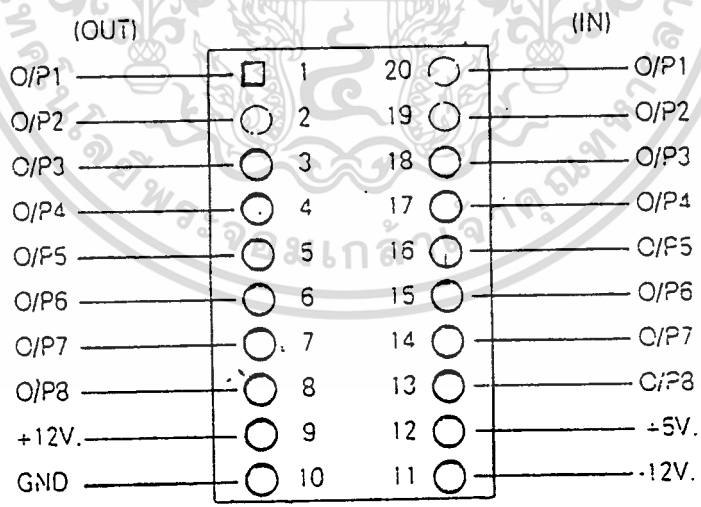
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 6.3 แสดงวงจรส่วนที่ต่อกับชุดสาริตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 แสดงตำแหน่ง PIN ของ DIGITAL CARD



รูปที่ 6.5 แสดงตำแหน่ง PIN ของ ANALOG CARD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 7

### สรุปผลและข้อเสนอแนะ

จากการดำเนินการจัดทำโครงการปริญญาโทฉบับนี้ ผลการจัดทำได้เป็นไปตามวัตถุประสงค์ที่ตั้งไว้ ซึ่งสามารถสรุปการวิจัยได้ดังนี้ คือ

#### 7.1 ได้รับประโยชน์ทางด้านระบบการทำงานของโรงไฟฟ้าพลังความร้อนร่วม

จากการศึกษาหลักการการทำงานของโรงไฟฟ้าพลังความร้อนร่วม ผู้จัดทำ ได้มีความรู้เกี่ยวกับระบบการควบคุมของโรงไฟฟ้าพลังความร้อนร่วมเป็นอย่างมาก ทราบถึงวิธีการเดินเครื่องของโรงไฟฟ้าพลังความร้อนร่วม รวมทั้งวิธีการปฏิบัติเมื่อเกิดเหตุการณ์ผิดปกติของโรงไฟฟ้าพลังความร้อนร่วมด้วย

#### 7.2 ประโยชน์ทางด้านการศึกษาการทำงานของโรงไฟฟ้าพลังความร้อนร่วม

การศึกษาระบบการทำงานของโรงไฟฟ้าพลังความร้อนร่วมนี้ สามารถศึกษาได้จากการทำงานของชุดสาธิต ซึ่งเป็นขั้นตอนตั้งแต่การเดินเครื่องในสภาวะปกติ และในสภาวะที่โรงไฟฟ้าเกิดความผิดพลาด ซึ่งมีประโยชน์คือ สามารถประหยัดค่าใช้จ่ายในการฝึกอบรม และมีความปลอดภัยกว่าการศึกษาจากโรงไฟฟ้าจริง

#### 7.3 ประโยชน์ทางด้านการศึกษาการเขียนโปรแกรมภาษาซี

โครงการนี้ ใช้โปรแกรมเทอร์โบซีในการควบคุมการทำงานของชุดสาธิต ผู้จัดทำจึงได้ศึกษาโครงสร้างและหลักการในการเขียนโปรแกรม ทำให้ทราบถึงวิธีการเขียนโปรแกรม และวิธีในการแก้ปัญหาเมื่อเกิดข้อผิดพลาดในการทำงาน ซึ่งสามารถนำความรู้ที่ได้รับนี้ไปประยุกต์ใช้ในงานอื่น ๆ ได้

## 7.4 ประโยชน์ทางการได้ความรู้จากการออกแบบวงจรอินเทอร์เฟซ

การออกแบบวงจรอินเทอร์เฟซ มีทั้งแบบดิจิทัล และอนาลอก รวมทั้งวงจรแปลงสัญญาณแรงดันและกระแส ผู้จัดทำได้นำความรู้จากการศึกษา รวมทั้งการค้นคว้า และการปรึกษาอาจารย์ที่ปรึกษาโครงการ จนสามารถสร้างวงจรอินเทอร์เฟซให้สามารถเชื่อมต่อระหว่างคอมพิวเตอร์และชุดสาธิตได้

## 7.5 ประโยชน์ทางการร่วมมือในการทำงานเป็นกลุ่ม

เป็นการประสานงาน และการแบ่งงานกันทำในกลุ่ม มีการจัดแบ่งหน้าที่ ความรับผิดชอบของแต่ละคน ทำให้มีการประสานงานในการทำงาน ซึ่งสามารถนำไปประยุกต์ใช้ในการทำงานได้

นอกจากนี้ ผู้จัดทำยังได้ประสบปัญหาในการทำงาน แต่ก็สามารถแก้ไขปัญหาดังกล่าวได้โดยลุล่วง โดยแยกประเภทของปัญหาออกเป็นดังนี้ คือ

### 1. ปัญหาทางด้าน HARDWARE

จากผลการทดลองโครงการในช่วงแรก ๆ จะพบปัญหาทางการทำงานของวงจร เนื่องจากการออกแบบลายปริ้นซ์ผิดพลาด เมื่อทำการประกอบอุปกรณ์ลงบนแผ่นปริ้นซ์แล้ว จะทำให้วงจรทำงานผิดพลาด เมื่อได้ตรวจสอบดูแล้วก็พบว่าเกิดจากการลัดวงจรระหว่างจุดบัดกรี ซึ่งก็สามารถแก้ไขปัญหานั้นได้

### 2. ปัญหาทางด้าน SOFTWARE

ภายหลังจากการประกอบอุปกรณ์ลงบน INTERFACE CARD เรียบร้อยแล้ว ได้ทำการทดลองร่วมกับ SOFTWARE ซึ่งเขียนขึ้นโดยใช้โปรแกรม Turbo C ผลปรากฏว่าใช้งานในช่วงแรกไม่ได้ เนื่องจากข้อมูลที่ส่งออกจากคอมพิวเตอร์ไม่เป็นไปตาม Step การทำงานที่เขียนไว้ในโปรแกรม ซึ่งจากการตรวจสอบดูพบว่าเกิดขึ้นเนื่องจากทางผู้จัดทำได้เขียนโปรแกรมไว้ล่วงหน้าไว้ก่อนที่จะทำการประกอบอุปกรณ์ลงบน INTERFACE CARD ผู้จัดทำจึงได้ทำการแก้ไขโปรแกรมใหม่ โดยการศึกษาจากหนังสือคู่มือภาษาซี และขอคำแนะนำจากอาจารย์ที่ปรึกษาโครงการ จึงสามารถแก้ไขปัญหานี้ได้ โครงการนี้จึงเสร็จสมบูรณ์ตามจุดประสงค์ที่ตั้งไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include <dos.h>
#include <graphics.h>
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <conio.h>

#define UP    0x4800
#define DOWN  0x5000
#define LEFT  0x4B00
#define RIGHT 0x4D00
#define ENTER 0x1C0D
#define ESC   0x011B
#define SPBAR 0x3920
#define DELAY 200

void Initial_graph(void);
void PrintBg(int x,int y,int color,char *ch,int size,int fon);
void Window(int x,int y,int x1,int y1,int h1,int wh,int bk);
void Init_Menu(void);
void Menu(void);
void Show_Combined(void);
void Combined_Cycle(void);
void Title(void);
void Structure(void);
void oom(int x,int y);
void arrow_1(int x,int y,int color,int t);
void arrow_2(int x,int y,int color);
void reotan(int x1,int y1,int x2,int y2,int color);
void boiler(int x,int y,int color);
void turbine1(int x,int y,int color);
void turbine2(int x,int y,int color);
void Reotan(int x,int y,int x1,int y1,int color);
void Iri_Vale(int value);
void Iri_Pump(int value);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void Vale_Hor(int x,int y,int color);
void Vale_Ver(int x,int y,int color);
void Line_Hor(int x,int y,int x1,int o1,int o2,int t,int l);
void Line_Ver(int x,int y,int y1,int o1,int o2,int t,int l);
void follow(void);
void Close(int r,int o);
void sh_text(void);
void combustion(int y);
void Condition(void);
void Dump_W(int w);
void Heat(int h);
void Alarm(void);
void Sound(int freq);
void CHECK_OUTPORT(void);

char *Sh_menu[7]={"Structure","Simulate ","HRSG No.1","HRSG No.2",
                "Alm_Pump","P_Trip","Exit"};
char *stat[5] = {"SENER","A to D","","8255","MODEL"};
char *Sh_C[5] = {"Show Simulate","Show HRSG No.1",
                "Show HRSG No.2","Pump Falute","Plant Trip"};
char *Active[2] = {"Start","Close"};
char *TEMP[6] = {"0","1","2","3","4","5"};
char *GEN;
int pos,Start,Al,Pr0,Pr1,Pr2;
int Lin[130][2],P_STOP;
int gen1,gen2,gen3,level,level1,level2,level_d,Temp,l_old;
int Temp1,Temp2;

union REGS inregs,outregs;

main()
{
  Initial_graph();
  Menu();
  closegraph();
  printf("Good Bye !");
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* Initial Graph Mode */
void Initial_graph()
{ int GraphDriver = DETECT;
  int ErrorCode =0;
  int GraphMode;
  initgraph(&GraphDriver,&GraphMode,"");
  ErrorCode = graphresult();
  if (ErrorCode != grOk)
    { printf("\n\tGraphics error \n %s",grapherrormsg(ErrorCode));
      exit(1); }
  outportb(0x303,0x80);outportb(0x307,0x80); outportb(0x30B,0x80);
}

/* Prints English Character */
void PrintEg(int x,int y,int color,char *oh,int size,int fon)
{ int a,b,c;
  if(size>4)size=4;
  for(b=1;b<=3;b++)
    { if(b==1){a=0;c=1;}
      if(b==2){a=15;c=3;}
      if(b==3){a=color;c=2;}
      setcolor(a);
      settextjustify(1,1);
      settextstyle(fon,0,size);
      outtextxy(x-o,y-o,oh);
    }
}

void Window(int x,int y,int x1,int y1,int hin,int wh,int bk)
{ int l,o1,o2,o3;
  if(bk==0){o1=1;o2=11;o3=9;}
  if(bk==1){o1=8;o2=15;o3=7;}
  setlinestyle(0,0,1);
  for(l=0;l<=hin;l++)
    { if(wh==1) setcolor(o1); else setcolor(o2);
      line(x+l,y+l,x+l,y1-l);
      line(x1-l,y+l,x1-l,y1);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    if(wh==0) setcolor(c1); else setcolor(c2);
line(x1-1,y1-1,x1-1,y1+1);
line(x+1,y1-1,x1-1,y1-1);
}    setfillstyle(1,c3);
    bar(x+hin,y+hin,x1-hin,y1-hin);
    setcolor(0);
    reotangle(x,y,x1,y1);
    if(bk==1&&wh==1)
    { setcolor(15);
    line(x,y,x,y1-1); line(x,y,x1-1,y);
    }
}
/* Initail Menu */
void Irit_Menu()
{ int i;
  setfillstyle(1,0);
  bar(0,0,getmaxx(),getmaxy());
  Window(5,100,getmaxx()/4-5,getmaxy()-30,2,0,1);
  setfillstyle(1,4);
  bar(15,152,getmaxx()/4-15,getmaxy()-37);
  Window(getmaxx()/4,5,getmaxx()-5,getmaxy()-5,2,0,1);
  Window(getmaxx()/4+5,10,getmaxx()-10,getmaxy()-10,3,1,1);
  setfillstyle(1,9);
  bar(getmaxx()/4+9,14,getmaxx()-14,getmaxy()-14);
  PrintEg(80,125,13,"MENU" ,5,1);
  for(i=0;i<7;i++)
  { Window(20,160+i*40,140,195+i*40,2,0,1);
    PrintEg(85,i*40+176,1,Sh_menu[i],1,8);
  }
  Clean_Buffer();
  setcolor(10);
  reotangle(19,159+pos*40,141,196+pos*40);
  PrintEg(getmaxx()/2+80,70,4," KING MONGKUT 'S INSTITUTE " ,5,1);
  PrintEg(getmaxx()/2+80,120,4," OF TECHNOLOGY LADKRABANG " ,3,1);
  PrintEg(getmaxx()/2+85,180,1,"COMBINED CYCLE POWER PLANT SIMULATOR",2,1);
  PrintEg(getmaxx()/2+80,250,9,"BY." ,6,1);
  PrintEg(getmaxx()/2+80,320,5,"MR THANONCHAI THAWANON ID.37013340",1,1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PrintEg(getmaxx()/2+80,360,5,"MR PAKORN SITTHIWONG ID.37013344",1,1);
PrintEg(getmaxx()/2+80,400,5,"MR PHONCHAI THONGWILAI ID.37013347",1,1);
}
/* Show Main Menu for select function */
void Menu()
{ int i,key; pos=0;
  Init_Menu();
  do
  { key=bioskey(0);
    switch(key)
    { case UP: setcolor(4);
      rectangle(19,159+pos*40,141,196+pos*40);
      pos = (pos==0) ? 6 : -pos; delay(100);
      setcolor(10);
      rectangle(19,159+pos*40,141,196+pos*40);
      break;
    case DOWN: setcolor(4);
      rectangle(19,159+pos*40,141,196+pos*40);
      pos = (pos==6) ? 0 : ++pos; delay(100);
      setcolor(10);
      rectangle(19,159+pos*40,141,196+pos*40);
      break;
    case ENTER:
      Window(20,160+pos*40,140,195+pos*40,2,1,1);
      PrintEg(85,pos*40+176,1,Sh_menu[pos],1,8);
      delay(300);
      Window(20,160+pos*40,140,195+pos*40,2,0,1);
      PrintEg(85,pos*40+176,1,Sh_menu[pos],1,8);
      delay(300);
      if(pos==0){ Close(0,0); Structure(); }
      if(pos==1){ Combined_Cycle(); }
      if(pos==2){ Combined_Cycle(); }
      if(pos==3){ Combined_Cycle(); }
      if(pos==4){ Set_Bu(); Combined_Cycle(); }
      if(pos==5){ Set_Bu(); Combined_Cycle(); }
      if(pos==6){ key = ESC; goto Exit; }
    }
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Irit_Menu(); Temp=Pr1=Pr2=0; Clean_Buffer(); Exit:
```

```
    break;
    default: break;
}
```

```
} while(key!=ESC);
```

```
Close(0,0);
```

```
}
```

```
void Close(int r,int op)
```

```
{ int i;
```

```
  setcolor(op); setlinestyle(0,0,1);
```

```
  if(r==1)
```

```
  for(i=320;i>0;i--)
```

```
    { rectangle(i,i/1.3,getmaxx()-i,getmaxy()-i/1.3); }
```

```
  if(r==0)
```

```
  for(i=0;i<=320;i++)
```

```
    { rectangle(i,i/1.3,getmaxx()-i,getmaxy()-i/1.3); }
```

```
}
```

```
/* Show Title Introduction */
```

```
void Title()
```

```
{ setfillstyle(1,9);
```

```
  bar(0,0,getmaxx(),getmaxy());
```

```
  Window(3,3,getmaxx()-3,getmaxy()-3,3,0,1);
```

```
  Window(8,8,getmaxx()-8,getmaxy()-8,2,1,0);
```

```
  Window(540,425,620,455,4,0,1);
```

```
  Window(540,425,620,455,4,1,1);
```

```
  PrintEg(580,440,9,"Close",1,1); getch();
```

```
  delay(300);
```

```
  Window(540,425,620,455,4,0,1);
```

```
  PrintEg(580,440,9,"Close",1,1);
```

```
  Close(0,5);
```

```
}
```

```
void Structure()
```

```
{ int i,key0;
```

```
  setfillstyle(1,1);
```

```
  bar(0,0,getmaxx(),getmaxy());
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Window(3,3,getmaxx()-3,getmaxy()-3,3,0,1);
Window(8,8,getmaxx()-8,getmaxy()-8,2,1,0);
    Window(270,420,370,460,3,0,1);
    PrintEg(getmaxx()/2,440,1,"Close",1,1);
    PrintEg(getmaxx()/2,70,9,"STRUCTURE",10,1);
for(i=0;i<=4;i++)
{ if(i!=2){
    Window(30+120*(i),250,120+120*(i),310,2,1,1);
    setcolor(14);
    settextstyle(1,0,1);
    settextjustify(1,1);
    outtextxy(75+120*(i),280,stut[i]); }
if(i!=4)
arrow_1(148+120*(i),280,13,1,5);
}
Window(270,145,365,190,2,0,0);
setcolor(14);
setfillstyle(1,10);
settextstyle(1,0,2);
outtextxy(315,158,"C");
settextstyle(0,0,1);
outtextxy(315,178,"COMPLIER");
outtextxy(315,255,"COMPUTER");
arrow_2(315,200,12);
com(290,310);
    do
    { key0=bioskey(0);
    } while(key0!=ESC);
Window(270,420,370,460,3,1,1);
PrintEg(getmaxx()/2,440,1,"Close",1,1);
    delay(300);
Window(270,420,370,460,3,0,1);
PrintEg(getmaxx()/2,440,1,"Close",1,1);
Close(1,5);
}

```

```
void arrow_1(int x,int y,int color,int t)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{ struct PTS {int x,y; };
  struct PTS poly[7];
  setfillstyle(SOLID_FILL,color);
  setcolor(15);
  setlinestyle(0,1,1);
  poly[0].x = x;
  poly[0].y = y;
  poly[1].x = x-12*t;
  poly[1].y = y+6*t;
  poly[2].x = x-9*t;
  poly[2].y = y+2*t;
  poly[3].x = x-20*t;
  poly[3].y = y+2*t;
  poly[4].x = x-20*t;
  poly[4].y = y-2*t;
  poly[5].x = x-9*t;
  poly[5].y = y-2*t;
  poly[6].x = x-12*t;
  poly[6].y = y-6*t;
  fillpoly(7,(int far*)poly);
}
void arrow_2(int x,int y,int color)
{ struct PTS {int x,y; };
  struct PTS poly[10];
  setfillstyle(SOLID_FILL,color);
  setcolor(7);
  setlinestyle(0,1,2);
  poly[0].x = x;
  poly[0].y = y;
  poly[1].x = x+10;
  poly[1].y = y+12;
  poly[2].x = x+4;
  poly[2].y = y+9;
  poly[3].x = x+4;
  poly[3].y = y+31;
  poly[4].x = x+10;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

poly[4].y = y+28;
poly[5].x = x;
poly[5].y = y+40;
poly[6].x = x-10;
poly[6].y = y+28;
poly[7].x = x-4;
poly[7].y = y+31;
poly[8].x = x-4;
poly[8].y = y+9;
poly[9].x = x-10;
poly[9].y = y+12;
fillpoly(10,(int far*)poly);
}
void com(int x,int y)
{
    reotan(x-5,y+5,70,50,7);
    reotan(x,y,60,40,14);
    reotan(x+15,y+8,30,3,13);
    reotan(x+25,y+13,10,5,12);
    reotan(x+10,y+15,40,2,11);
    reotan(x-15,y+35,90,20,7);
    reotan(x+45,y+24,25,2,15);
    line(x+65,y+22,x+66,y+26);
}
void reotan(int x1,int y1,int x2,int y2,int color)
{
    struct PTS {int x,y; };
    struct PTS poly[4];
    setfillstyle(1,color);
    setcolor(1);
    setlinestyle(0,1,1);
    poly[0].x = x1;
    poly[0].y = y1;
    poly[1].x = x1+x2;
    poly[1].y = y1;
    poly[2].x = x1+x2;
    poly[2].y = y1-y2;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

poly[3].x = x1;
poly[3].y = y1-y2;
fillpoly(4,(int far*)poly);
}

void Combined_Cycle()
{ int key1,pos1;
  Show_Combined();   Start=0;
  for(pos1=0;pos1<2;pos1++)
  {
Window(460+pos1*90,435,530+pos1*90,465,3,0,1);
PrintEg(495+pos1*90,450,3,Active[pos1],1,1);
  setcolor(7);   reotangle(459+pos1*90,434,531+pos1*90,466);
  } pos1=0; setcolor(10);
  reotangle(459+pos1*90,434,531+pos1*90,466);
do
{ while(!kbhit()){
if(Start==1) follow(); } key1=bioskey(0);
  setlinestyle(0,1,1);
  switch(key1)
  { case RIGHT: setcolor(7);
    reotangle(459+pos1*90,434,531+pos1*90,466);
    pos1 = (pos1==0) ? 1 :-pos1; delay(300);
    setcolor(10);
    reotangle(459+pos1*90,434,531+pos1*90,466);
    break;
    case LEFT: setcolor(7);
    reotangle(459+pos1*90,434,531+pos1*90,466);
    pos1 = (pos1==1) ? 0 :++pos1; delay(300);
    setcolor(10);
    reotangle(459+pos1*90,434,531+pos1*90,466);
    break;
    case ENTER: setcolor(7);
    reotangle(459,434,531,466);
    reotangle(459+90,434,531+90,466);
    setcolor(10);
    reotangle(459+pos1*90,434,531+pos1*90,466);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if(pos1==0){ Start = 1; }
        if(pos1==1){ key1= ESC; Start = 0; goto mex; }
Window(460+pos1*90,435,530+pos1*90,465,3,1,1);
    PrintEg(495+pos1*90,450,3,Active[pos1],1,1);
        delay(300);
Window(460+pos1*90,435,530+pos1*90,465,3,0,1);
    PrintEg(495+pos1*90,450,3,Active[pos1],1,1);
        mex: break;
        default: break;
    }
} while(key1!=ESC); pos1=1; setcolor(7);
    reotangle(459,434,531,466);
    reotangle(459+90,434,531+90,466);
    setcolor(10);
    reotangle(459+pos1*90,434,531+pos1*90,466);
Window(460+pos1*90,435,530+pos1*90,465,3,1,1);
    PrintEg(495+pos1*90,450,3,Active[pos1],1,1);
        delay(300);
Window(460+pos1*90,435,530+pos1*90,465,3,0,1);
    PrintEg(495+pos1*90,450,3,Active[pos1],1,1);
        delay(300);
}
void boiler(int x,int y,int color)
{ struct PTS {int x,y; };
  struct PTS poly[16];
  setfillstyle(SOLID_FILL,color);
  setcolor(14);
  setlinestyle(0,1,2);
  poly[0].x = x+152;
  poly[0].y = y+214;
  poly[1].x = x+177;
  poly[1].y = y+214;
  poly[2].x = x+177;
  poly[2].y = y+210;
  poly[3].x = x+165;
  poly[3].y = y+200;

```

เอกสารนี้เป็น poly[3].y = y+200; สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

poly[4].x = x+160;
poly[4].y = y+200;
poly[5].x = x+160;
poly[5].y = y+50;
poly[6].x = x+175;
poly[6].y = y+45;
poly[7].x = x+175;
poly[7].y = y+40;
poly[8].x = x+185;
poly[8].y = y+40;
poly[9].x = x+185;
poly[9].y = y+45;
poly[10].x = x+200;
poly[10].y = y+50;
poly[11].x = x+200;
poly[11].y = y+200;
poly[12].x = x+195;
poly[12].y = y+200;
poly[13].x = x+183;
poly[13].y = y+210;
poly[14].x = x+183;
poly[14].y = y+220;
poly[15].x = x+152;
poly[15].y = y+220;
fillpoly(16,(int far*)poly);
}

```

```

void turbine1(int x,int y,int oolor)
{
  struct PTS {int x,y; };
  struct PTS poly[16];
  setfillstyle(SOLID_FILL,color);
  setcolor(7);
  setlinestyle(0,1,2);
  poly[0].x = x+20;
  poly[0].y = y+155;
  poly[1].x = x+20;

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    poly[1].y = y+185;
    poly[2].x = x+45;
    poly[2].y = y+180;
    poly[3].x = x+45;
    poly[3].y = y+160;
    fillpoly(4,(int far*)poly);
}

void turbine2(int x,int y,int color)
{
    struct PTS {int x,y; };
    struct PTS poly[16];
    setfillstyle(SOLID_FILL,color);
    setcolor(7);
    setlinestyle(0,1,2);

    poly[0].x = x+55;
    poly[0].y = y+160;
    poly[1].x = x+55;
    poly[1].y = y+180;
    poly[2].x = x+80;
    poly[2].y = y+185;
    poly[3].x = x+80;
    poly[3].y = y+155;
    fillpoly(4,(int far*)poly);
}

void Reotan(int x,int y,int x1,int y1,int color)
{
    struct PTS {int x,y; };
    struct PTS poly[16];
    setfillstyle(SOLID_FILL,color);
    setcolor(11);
    setlinestyle(0,1,2);

    poly[0].x = x;
    poly[0].y = y;
    poly[1].x = x1;
    poly[1].y = y;
    poly[2].x = x1;
    poly[2].y = y1;
    poly[3].x = x;
    poly[3].y = y1;
    fillpoly(4,(int far*)poly);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    poly[3].y = y1;
    fillpoly(4,(int far*)poly);
}

```

```

void Steam(int color)

```

```

{  struct PTS {int x,y; };

```

```

    struct PTS poly[16];

```

```

    setfillstyle(SOLID_FILL,color);

```

```

    setcolor(11);

```

```

    setlinestyle(0,1,2);

```

```

    poly[0].x = 480;

```

```

    poly[0].y = 130;

```

```

    poly[1].x = 550;

```

```

    poly[1].y = 120;

```

```

    poly[2].x = 550;

```

```

    poly[2].y = 170;

```

```

    poly[3].x = 480;

```

```

    poly[3].y = 160;

```

```

    fillpoly(4,(int far*)poly);

```

```

}

```

```

void Vale_Hor(int x,int y,int color)

```

```

{  struct PTS {int x,y; };

```

```

    struct PTS poly[16];

```

```

    setfillstyle(SOLID_FILL,color);

```

```

    setcolor(15);

```

```

    setlinestyle(0,1,2);

```

```

    poly[0].x = x-4;

```

```

    poly[0].y = y-3;

```

```

    poly[1].x = x+4;

```

```

    poly[1].y = y+3;

```

```

    poly[2].x = x+4;

```

```

    poly[2].y = y-3;

```

```

    poly[3].x = x-4;

```

```

    poly[3].y = y+3;

```

```

    fillpoly(4,(int far*)poly);

```

```

    if(color==4)setfillstyle(1,12);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    fillellipse(x,y-5,2,2);
}
void Vale_Ver(int x,int y,int color)
{
    struct PTS {int x,y; };
    struct PTS poly[16];
    setfillstyle(SOLID_FILL,color);
    setcolor(15);
    setlinestyle(0,1,2);
    poly[0].x = x-3;
    poly[0].y = y-4;
    poly[1].x = x+3;
    poly[1].y = y+4;
    poly[2].x = x-3;
    poly[2].y = y+4;
    poly[3].x = x+3;
    poly[3].y = y-4;
    fillpoly(4,(int far*)poly);
    if(color==4)setfillstyle(1,12);
    fillellipse(x+5,y,2,2);
}

/* COMBUSTION TURBINE */
void oombustion(int y)
{
    setcolor(15);
    moveto(30,140+y);lineto(30,153+y);lineto(28,153+y);lineto(32,157+y);
    lineto(36,153+y);lineto(34,153+y);lineto(34,140+y);

    moveto(30,183+y);lineto(30,195+y);lineto(28,195+y);lineto(32,200+y);
    lineto(36,195+y);lineto(34,195+y);lineto(34,182+y);

    moveto(47,220+y);lineto(43,225+y);lineto(45,225+y);lineto(45,230+y);
    lineto(50,230+y);lineto(50,225+y);lineto(52,225+y);lineto(47,220+y);

    moveto(65,200+y);lineto(65,187+y);lineto(67,187+y);lineto(63,182+y);
    lineto(59,187+y);lineto(61,187+y);lineto(61,200+y);

    line(74,220+y,134,220+y);line(134,215+y,80,215+y);

    line(80,215+y,80,185+y);

    line(74,184+y,74,220+y);

```

เอกสารนี้ใช้เผยแพร่ฟรีโดยไม่หวังผลตอบแทนการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Reotan(20,120+y,45,140+y,9);
Reotan(134,212+y,136,222+y,14);
Reotan(90,155+y,130,185+y,6);
Reotan(30,200+y,65,220+y,4);
Reotan(136,210+y,150,224+y,1);
Reotan(150,212+y,152,222+y,14);
Reotan(140,40+y,146,210+y,9);
Reotan(505,280,555,340,9);
Reotan(570,130,610,160,4);
Reotan(580,290,610,330,10);
Reotan(408,275,412,290,14);
Reotan(423,275,427,290,12);
Reotan(438,275,442,290,14);
Reotan(525,340,535,343,12);
Reotan(550,160,510,270,7);
Reotan(550,143,570,147,4);
Steam(1);
line(550,270,555,280);line(510,270,505,280);
setcolor(14);line(136,224+y,150,210+y); setfillstyle(1,9);
setcolor(15);fillellipse(140,214+y,2,2);
setcolor(15);fillellipse(146,219+y,2,2);
boiler(0,y,8);
turbine1(0,y,14);
turbine2(0,y,12);
)

```

```

void sh_text()
{
  setcolor(11);
  setttextstyle(2,0,4);
  outtextxy(60,100,"GAS TURBINE NO.1");
  outtextxy(60,320,"GAS TURBINE NO.2");
  outtextxy(230,210,"HRSG. NO.1");
  outtextxy(230,430,"HRSG. NO.2");
  outtextxy(110,170,"GEN.1");
  outtextxy(110,390,"GEN.2");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

outtextxy(33,350,"AIR");
outtextxy(50,210,"COMB.");
outtextxy(50,430,"COMB.");
outtextxy(48,240,"FUEL");
outtextxy(48,460,"FUEL");
outtextxy(145,235,"STACK");
outtextxy(145,455,"STACK");
outtextxy(425,266,"DEAERATOR");
outtextxy(425,300,"FEEDWATER");
outtextxy(425,312,"TANK");
outtextxy(588,145,"GEN.3");
outtextxy(515,145,"STEAM TURB.");
outtextxy(445,160,"BYPASS");
outtextxy(532,288,"COND.");
outtextxy(590,260,"COOLING");
outtextxy(590,275,"TOWER");
outtextxy(583,370,"CONDENSATE");
outtextxy(583,385,"PUMPS");
outtextxy(425,50,"LP STEAM");
outtextxy(425,95,"HP STEAM");
outtextxy(480,365,"HP");
outtextxy(480,380,"FWP");
outtextxy(370,350,"LP");
outtextxy(370,365,"FWP");
}
void Show_Combined()
{
    setfillstyle(8,5); bar(0,0,getmaxx(),getmaxy());
    Window(4,4,getmaxx()-4,getmaxy()-4,2,1,1);
    setfillstyle(11,pos);
    bar(7,7,getmaxx()-9,getmaxy()-9);
    PrintBg(getmaxx()/2,20,7,"COMBINED CYCLE POWER PLANT",1,1);
    PrintBg(getmaxx()/2,460,9,Sh_C[pos-1],3,1);
    oombustion(0); oombustion(220);

```

**/\* DEAERATOR AND FEEDWATER TANK \*/**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Reotan(400,260,450,275,1);fillellipse(450,268,5,7);
fillellipse(400,268,5,7); bar(401,261,449,274);
Reotan(390,290,460,330,3);fillellipse(390,310,10,20);
fillellipse(460,310,10,20);bar(391,291,459,329);
moveto(454,264);lineto(457,264);lineto(457,271);lineto(454,271);
moveto(396,264);lineto(393,264);lineto(393,271);lineto(396,271);
Reotan(420,258,430,260,13);

```

```
/* LINE OF HP DRUM1 */
```

```

fillellipse(235,153,8,8);
line(228,154,170,154);line(172,158,230,158);line(170,164,188,164);
line(172,160,190,160);
ellipse(172,159,90,270,1,1);ellipse(170,159,90,270,3,5);
line(170,166,188,166);line(172,170,190,170);
ellipse(188,165,270,90,1,1);ellipse(190,165,270,90,3,5);
line(172,172,210,172);line(170,176,210,176);
ellipse(172,171,90,270,1,1);ellipse(170,171,90,270,3,5);
moveto(210,172);lineto(210,167);lineto(216,167);
moveto(210,176);lineto(210,181);lineto(216,181);
moveto(216,171);lineto(214,171);lineto(214,177);lineto(216,177);
moveto(223,171);lineto(226,171);lineto(226,177);lineto(223,177);
moveto(223,167);lineto(230,167);lineto(230,172);lineto(233,172);
lineto(233,160);
moveto(223,181);lineto(230,181);lineto(230,176);lineto(237,176);
lineto(237,160);

```

```
/* LINE OF HP DRUM2 */
```

```

fillellipse(235,373,8,8);
line(228,374,170,374);line(172,378,230,378);line(170,384,188,384);
line(172,380,190,380);
ellipse(172,379,90,270,1,1);ellipse(170,379,90,270,3,5);
line(170,386,188,386);line(172,390,190,390);
ellipse(188,385,270,90,1,1);ellipse(190,385,270,90,3,5);
line(172,392,210,392);line(170,396,210,396);
ellipse(172,391,90,270,1,1);ellipse(170,391,90,270,3,5);
moveto(210,392);lineto(210,387);lineto(216,387);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

moveto(210,396);lineto(210,401);lineto(216,401);
moveto(216,391);lineto(214,391);lineto(214,397);lineto(216,397);
moveto(223,391);lineto(226,391);lineto(226,397);lineto(223,397);
moveto(223,387);lineto(230,387);lineto(230,392);lineto(233,392);
lineto(233,380);
moveto(223,401);lineto(230,401);lineto(230,396);lineto(237,396);
lineto(237,380);

```

*/\* LINE OF HP DRUM1 TO HP DRUM2 \*/*

```

moveto(233,145);lineto(233,138);lineto(264,138);lineto(264,160);
lineto(266,160);lineto(262,165);lineto(258,160);lineto(260,160);
lineto(260,142);lineto(237,142);lineto(237,145);
moveto(260,162);lineto(260,185);lineto(168,185);
ellipse(170,190,90,270,1,1);ellipse(168,190,90,270,3,5);
moveto(168,195);lineto(280,195);lineto(280,210);lineto(284,210);
lineto(284,198);lineto(305,198);lineto(305,196);lineto(284,196);
lineto(284,191);lineto(170,191);
moveto(264,162);lineto(264,189);lineto(170,189);
line(280,217,280,240);line(284,217,284,226);line(284,230,284,240);
line(284,247,284,259);
moveto(284,261);lineto(284,415);lineto(168,415);
moveto(170,411);lineto(280,411);lineto(280,247);
moveto(233,365);lineto(233,358);lineto(264,358);lineto(264,380);
lineto(266,380);lineto(262,385);lineto(258,380);lineto(260,380);
lineto(260,362);lineto(237,362);lineto(237,365);
moveto(264,383);lineto(264,409);lineto(170,409);
moveto(260,383);lineto(260,405);lineto(168,405);
ellipse(170,410,90,270,1,1);ellipse(168,410,90,270,3,5);

```

*/\* LINE OF LP DRUM1 \*/*

```

fillellipse(235,93,8,8);
line(177,94,228,94);line(179,98,230,98);
line(179,100,192,100);line(177,104,190,104);
ellipse(179,99,90,270,1,1);ellipse(177,99,90,270,3,5);
line(177,106,190,106);line(179,110,192,110);
ellipse(190,105,270,90,1,1);ellipse(192,105,270,90,3,5);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

line(179,112,210,112);line(177,116,210,116);
ellipse(179,111,90,270,1,1);ellipse(177,111,90,270,3,5);
moveto(210,112);lineto(210,107);lineto(216,107);
moveto(210,116);lineto(210,121);lineto(216,121);
moveto(216,111);lineto(214,111);lineto(214,117);lineto(216,117);
moveto(223,111);lineto(226,111);lineto(226,117);lineto(223,117);
moveto(223,107);lineto(230,107);lineto(230,112);lineto(233,112);
lineto(233,100);
moveto(223,121);lineto(230,121);lineto(230,116);lineto(237,116);
lineto(237,100);

```

```

/* LINE OF LP DRUM2 */

```

```

fillellipse(235,313,8,8);
line(177,314,228,314);line(179,318,230,318);
line(177,320,192,320);line(177,324,190,324);
ellipse(179,319,90,270,1,1);ellipse(177,319,90,270,3,5);
line(177,326,190,326);line(179,330,192,330);
ellipse(190,325,270,90,1,1);ellipse(192,325,270,90,3,5);
line(179,332,210,332);line(177,336,210,336);
ellipse(179,331,90,270,1,1);ellipse(177,331,90,270,3,5);
moveto(210,332);lineto(210,327);lineto(216,327);
moveto(210,336);lineto(210,341);lineto(216,341);
moveto(216,331);lineto(214,331);lineto(214,337);lineto(216,337);
moveto(223,331);lineto(226,331);lineto(226,337);lineto(223,337);
moveto(223,327);lineto(230,327);lineto(230,332);lineto(233,332);
lineto(233,320);
moveto(223,342);lineto(230,342);lineto(230,336);lineto(237,336);
lineto(237,320);

```

```

/* LINE OF LP DRUM1 TO LP DRUM2 */

```

```

moveto(233,85);lineto(233,78);lineto(264,78);lineto(264,100);
lineto(266,100);lineto(262,105);lineto(258,100);lineto(260,100);
lineto(260,82);lineto(237,82);lineto(237,85);
moveto(260,102);lineto(260,125);lineto(177,125);
moveto(264,102);lineto(264,129);lineto(179,129);
ellipse(179,130,90,270,1,1);ellipse(177,130,90,270,3,5);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ภายใต้การเป็นเจ้าของของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

moveto(179,131);lineto(274,131);lineto(274,136);lineto(290,136);  
 lineto(290,138);lineto(274,138);lineto(274,145);lineto(270,145);  
 lineto(270,135);lineto(177,135);  
 line(270,152,270,172);line(274,152,274,160);line(274,164,274,172);  
 moveto(270,190);lineto(270,179);lineto(274,179);lineto(274,186);  
 lineto(305,186);lineto(305,188);lineto(274,188);  
 lineto(274,191);lineto(270,191);  
 moveto(179,351);lineto(270,351);lineto(270,195);lineto(274,195);  
 lineto(274,355);lineto(177,355);  
 moveto(233,305);lineto(233,298);lineto(264,298);lineto(264,320);  
 lineto(266,320);lineto(262,325);lineto(258,320);lineto(260,320);  
 lineto(260,302);lineto(237,302);lineto(237,305);  
 moveto(260,323);lineto(260,345);lineto(177,345);  
 moveto(264,323);lineto(264,349);lineto(179,349);  
 ellipse(179,350,90,270,1,1);ellipse(177,350,90,270,3,5);

/\* LINE OF HP DRUM TO STEAM TURBINE \*/

moveto(284,226);lineto(305,226);lineto(305,165);lineto(303,165);  
 lineto(307,160);lineto(311,165);lineto(309,165);lineto(309,230);  
 lineto(284,230);  
 moveto(305,162);lineto(305,85);lineto(482,85);lineto(482,105);  
 lineto(478,105);lineto(478,89);lineto(309,89);lineto(309,162);  
 moveto(478,110);lineto(478,125);lineto(476,125);lineto(480,130);  
 lineto(484,125);lineto(482,125);lineto(482,110);lineto(478,110);

/\* LINE OF LP DRUM TO STEAM TURBINE \*/

moveto(274,160);lineto(290,160);lineto(290,105);lineto(288,105);  
 lineto(292,100);lineto(296,105);lineto(294,105);lineto(294,164);  
 lineto(274,164);  
 moveto(290,103);lineto(290,40);lineto(512,40);lineto(512,105);  
 lineto(508,105);lineto(508,44);lineto(380,44);lineto(380,85);  
 lineto(376,85);lineto(376,44);lineto(294,44);lineto(294,103);  
 moveto(508,110);lineto(508,120);lineto(506,120);lineto(510,126);  
 lineto(514,120);lineto(512,120);lineto(512,110);lineto(508,110);

/\* LINE OF LP STEAM PASS DEAERATOR \*/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

reotangle(376,89,380,176);reotangle(376,178,380,186);
reotangle(376,188,380,196);reotangle(376,198,380,206);
reotangle(376,208,380,214);
moveto(380,221);lineto(380,226);lineto(488,226);lineto(488,230);
lineto(473,230);lineto(473,269);lineto(457,269);lineto(457,265);
lineto(469,265);lineto(469,230);lineto(380,230);lineto(380,265);
lineto(393,265);lineto(393,269);lineto(376,269);lineto(376,221);
moveto(528,167);lineto(528,226);lineto(494,226);lineto(494,230);
lineto(532,230);lineto(532,168);

```

```

/* LINE OF LP BY-PASS */

```

```

reotangle(294,136,305,138);
moveto(309,136);lineto(322,136);lineto(322,176);lineto(440,176);
lineto(440,178);lineto(320,178);lineto(320,138);lineto(309,138);
reotangle(447,176,507,178);reotangle(507,174,510,180);
moveto(274,186);lineto(305,186);lineto(305,188);lineto(274,188);
reotangle(309,186,440,188);reotangle(447,186,507,188);
reotangle(507,184,510,190);

```

```

/* LINE OF HP BY-PASS */

```

```

moveto(284,196);lineto(305,196);lineto(305,198);lineto(284,198);
reotangle(309,196,440,198);reotangle(447,196,507,198);
reotangle(507,194,510,200);
moveto(284,259);lineto(320,259);lineto(320,206);lineto(440,206);
lineto(440,208);lineto(322,208);lineto(322,261);lineto(284,261);
reotangle(447,206,507,208);reotangle(507,204,510,210);

```

```

/* LINE OF CONDENSOR */

```

```

line(573,299,580,299);line(573,303,580,303);line(513,299,563,299);
line(515,303,563,303);line(513,309,545,309);line(515,305,547,305);
ellipse(515,304,90,270,1,1);ellipse(513,304,90,270,3,5);
line(513,311,545,311);line(515,315,547,315);
ellipse(545,310,270,90,1,1);ellipse(547,310,270,90,3,5);
line(515,317,580,317);line(513,321,580,321);
ellipse(515,316,90,270,1,1);ellipse(513,316,90,270,3,5);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/\* LINE OF CONDENSATE PUMPS TO DEAERATOR \*/

moveto(528,343);lineto(528,360);lineto(513,360);lineto(513,370);  
 moveto(532,343);lineto(532,360);lineto(547,360);lineto(547,370);  
 moveto(517,370);lineto(517,364);lineto(528,364);lineto(528,370);  
 moveto(532,370);lineto(532,364);lineto(543,364);lineto(543,370);  
 moveto(517,381);lineto(517,386);lineto(528,386);lineto(528,381);  
 moveto(532,381);lineto(532,386);lineto(543,386);lineto(543,381);  
 moveto(547,381);lineto(547,390);lineto(532,390);lineto(532,415);  
 lineto(356,415);lineto(356,400);lineto(360,400);lineto(360,411);  
 lineto(528,411);lineto(528,390);lineto(513,390);lineto(513,381);  
 reotangle(356,396,360,385);  
 moveto(356,381);lineto(356,240);lineto(376,240);lineto(376,244);  
 lineto(360,244);lineto(360,381);  
 moveto(380,240);lineto(427,240);lineto(427,253);lineto(429,253);  
 lineto(425,258);lineto(421,253);lineto(423,253);lineto(423,244);  
 lineto(380,244);

/\* LINE OF HP FWP TO HP DRUM \*/

moveto(443,330);lineto(447,330);lineto(447,360);lineto(462,360);  
 lineto(462,370);  
 moveto(443,330);lineto(443,360);lineto(428,360);lineto(428,370);  
 moveto(432,370);lineto(432,364);lineto(443,364);lineto(443,370);  
 moveto(447,370);lineto(447,364);lineto(458,364);lineto(458,370);  
 moveto(462,380);lineto(462,390);lineto(447,390);lineto(447,400);  
 lineto(336,400);lineto(336,295);lineto(315,295);lineto(315,297);  
 lineto(310,293);lineto(315,289);lineto(315,291);lineto(336,291);  
 lineto(336,208);lineto(340,208);lineto(340,396);lineto(443,396);  
 lineto(443,390);lineto(428,390);lineto(428,380);  
 reotangle(274,291,280,295);  
 line(284,291,312,291);line(284,295,312,295);  
 moveto(432,380);lineto(432,386);lineto(443,386);lineto(443,380);  
 moveto(447,380);lineto(447,386);lineto(458,386);lineto(458,380);  
 reotangle(220,291,270,295);  
 line(213,291,168,291);line(213,295,170,295);line(165,294,165,368);  
 line(169,296,169,366);line(170,367,230,367);line(168,371,228,371);

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของวิศวกรรมโยธา ภาควิชาโยธา มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ellipse(168,294,90,180,3,3);ellipse(168,368,180,270,3,3);
line(168,71,213,71);line(170,75,213,75);line(165,74,165,148);
line(169,76,169,146);line(170,147,230,147);line(168,151,228,151);
ellipse(170,76,90,180,1,1);ellipse(170,146,180,270,1,1);
ellipse(168,74,90,180,3,3);ellipse(168,148,180,270,3,3);
rectangle(336,198,340,206);rectangle(336,188,340,196);
rectangle(336,178,340,186);rectangle(336,89,340,176);
moveto(336,85);lineto(340,85);lineto(340,71);lineto(315,71);
lineto(315,69);lineto(310,73);lineto(315,77);lineto(315,75);
lineto(336,75);lineto(336,85);
line(294,71,312,71);line(294,75,312,75);
rectangle(220,71,290,75);

/* LINE OF LP FWP TO LP DRUM */
moveto(407,330);lineto(407,345);lineto(422,345);lineto(422,355);
moveto(403,330);lineto(403,345);lineto(388,345);lineto(388,355);
moveto(392,355);lineto(392,349);lineto(403,349);lineto(403,355);
moveto(407,355);lineto(407,349);lineto(418,349);lineto(418,355);
moveto(392,365);lineto(392,371);lineto(403,371);lineto(403,365);
moveto(407,365);lineto(407,371);lineto(418,371);lineto(418,365);
moveto(422,365);lineto(422,375);lineto(407,375);lineto(407,385);
lineto(346,385);lineto(346,280);lineto(340,280);lineto(340,276);
lineto(346,276);lineto(346,208);lineto(350,208);lineto(350,381);
lineto(403,381);lineto(403,375);lineto(388,375);lineto(388,365);
rectangle(346,89,350,176);rectangle(346,178,350,186);
rectangle(346,188,350,196);rectangle(346,198,350,206);
rectangle(220,276,270,280);rectangle(274,276,280,280);
line(284,276,313,276);line(284,280,313,280);
moveto(336,276);lineto(315,276);lineto(315,274);lineto(310,278);
lineto(315,282);lineto(315,280);lineto(336,280);
line(177,276,213,276);line(179,280,213,280);line(174,279,174,291);
line(178,281,178,291);line(174,295,174,308);line(178,295,178,306);
line(179,307,230,307);line(177,311,228,311);
ellipse(179,281,90,180,1,1);ellipse(179,306,180,270,1,1);
ellipse(177,279,90,180,3,3);ellipse(177,308,180,270,3,3);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ภายใต้การดำเนินงานของสำนักงาน กสทช. ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lineto(310,58);lineto(315,54);lineto(315,56);lineto(350,56);

lineto(350,85);

line(294,56,312,56);line(294,60,312,60);

rectangle(220,56,290,60);

line(177,56,213,56);line(179,60,213,60);line(174,59,174,71);

line(178,61,178,71);line(174,75,174,88);line(178,75,178,86);

line(179,87,230,87);line(177,91,228,91);

ellipse(179,61,90,180,1,1);ellipse(179,86,180,270,1,1);

ellipse(177,59,90,180,3,3);ellipse(177,88,180,270,3,3);

Iri_Vale(0);

Iri_Pump(0);

sh_text(); if(pos==4||pos==5) Pr1=3300; P_STOP=1;
}

void Iri_Vale(int value)
{ int i; i=1; if(pos==4&&value==1||pos==5&&value==1)i=4;
  Vale_Hor(217,58,i); Vale_Hor(217,73,i);
  Vale_Hor(217,278,i); Vale_Hor(217,293,i);
  Vale_Hor(492,228,i); Vale_Hor(442,207,1);
  Vale_Hor(442,197,1); Vale_Hor(442,187,1);
  Vale_Hor(442,177,1); Vale_Ver(480,107,i);
  Vale_Ver(510,107,i); Vale_Ver(272,148,i);
  Vale_Ver(272,176,i); Vale_Ver(282,213,i);
  Vale_Ver(282,243,i); Vale_Ver(378,217,i);
}

void Iri_Pump(int value)
{ setcolor(14); setfillstyle(1,1);
  fillellipse(220,179,4,4); fillellipse(220,399,4,4);
  fillellipse(220,119,4,4); fillellipse(220,339,4,4);
  fillellipse(390,360,5,5); fillellipse(515,375,5,5);
  fillellipse(430,375,5,5);
  if(pos==4&&value==1||pos==5&&value==1)
  { setcolor(15); setfillstyle(1,12);}
  fillellipse(220,169,4,4); fillellipse(220,389,4,4);
  fillellipse(220,109,4,4); fillellipse(220,329,4,4);
  fillellipse(405,360,5,5); fillellipse(420,360,5,5);
  fillellipse(568,301,5,5); fillellipse(460,375,5,5);

```

```

filellipse(530,375,5,5); filellipse(545,375,5,5);
filellipse(445,375,5,5);
    setlinestyle(0,0,1); setcolor(15);
reotangle(243,108,255,120); reotangle(243,168,255,180);
reotangle(243,328,255,340); reotangle(243,388,255,400);
    setfillstyle(1,9);
bar(245,109,254,119); bar(245,169,254,179);
bar(245,329,254,339); bar(245,389,254,399);
settextjustify(1,1); setfillstyle(1,9); settextstyle(0,0,1);
outtextxy(249,115,"0"); outtextxy(249,175,"0");
outtextxy(249,335,"0"); outtextxy(249,395,"0");
    setfillstyle(1,9); filellipse(110,133,2,2);
filellipse(110,353,2,2); setcolor(7); reotangle(491,111,498,89);
reotangle(521,111,528,89); reotangle(584,111,591,89);
bar(492,110,497,90); bar(522,110,527,90); bar(585,110,590,90);
bar(230,50,235,30); bar(245,50,250,30);
bar(230,270,235,250); bar(245,270,250,250);
}
void Line_Ver(int x,int y,int y1,int o1,int o2,int t,int l)
{ int Sol;
    if(y<y1)Sol= y1-y; if(y>y1)Sol= y-y1;
    if(A1==0){ setcolor(o1);setlinestyle(2,3,t); }
    if(A1==1){ setcolor(o2);setlinestyle(1,2,t); }
    if(Lin[1][1]==1)line(x,y,x,y1);
    if(l==0) line(x,y,x,y1);
        if(Lin[1][1]==2)
        { Pr0 = Sol;
            if(Pr0==Lin[1][0]) Pr0=Lin[1][1]=1;
            if(Pr0!=Lin[1][0]&&P_STOP==0) Lin[1][0]++;
            if(y1-y>1)line(x,y,x,y+Lin[1][0]);
            if(y-y1>1)line(x,y,x,y-Lin[1][0]); }
    setlinestyle(1,1,0);
}
void Line_Hor(int x,int y,int x1,int o1,int o2,int t,int l)
{ int Sol;
    if(x<x1)Sol= x1-x; if(x>x1)Sol= x-x1;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของบริษัทฯ เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(A1==0){ setcolor(c1); setlinestyle(2,3,t); }
    if(A1==1){ setcolor(c2); setlinestyle(1,2,t); }
    if(Lin[1][1]==1)line(x,y,x1,y);
    if(l==0) line(x,y,x1,y);
        if(Lin[1][1]==2)
        { Pr0 = Sol;
          if(Pr0==Lin[1][0]) Pr0=Lin[1][1]=1;
          if(Pr0!=Lin[1][0]&&&P_STOP==0) Lin[1][0]++;
          if(x1-x>1)line(x,y,x+Lin[1][0],y);
          if(x-x1>1)line(x,y,x-Lin[1][0],y); }
    setlinestyle(1,1,0);
}
void Condition()
{
    if(Pr1<2)Lin[1][1]=Pr0=2;
    if(Pr1>18&&Pr1<20)Lin[2][1]=Pr0=2;
    if(Pr1>55&&Pr1<57)Lin[3][1]=Pr0=2;
    if(Pr1>64&&Pr1<66){ Lin[4][1]=Pr0=2;
    setcolor(15); setfillstyle(1,12); fillellipse(568,301,5,5);
    Sound(300);}
    if(Pr1>114&&Pr1<116)Lin[5][1]=Pr0=2;
    if(Pr1>150&&Pr1<152)Lin[6][1]=Pr0=2;
    if(Pr1>190&&Pr1<192)Lin[7][1]=Pr0=2;
    if(Pr1>270&&Pr1<272)Lin[8][1]=Pr0=2;
    if(Pr1>260&&Pr1<262)Lin[9][1]=Pr0=2;
    if(Pr1>290&&Pr1<292)Lin[10][1]=Pr0=2;
    if(Pr1>300&&Pr1<302)Lin[11][1]=Pr0=2;
    if(Pr1>340&&Pr1<342){ Lin[12][1]=Pr0=2; ;
    setcolor(15); setfillstyle(1,12);
    fillellipse(445,375,5,5); fillellipse(460,375,5,5);
    fillellipse(405,360,5,5); fillellipse(420,360,5,5);
    Sound(300);}
    if(Pr1>343&&Pr1<345)Lin[13][1]=Pr0=2;
    if(Pr1>400&&Pr1<402)Lin[14][1]=Pr0=2;
    if(Pr1>343&&Pr1<345)Lin[15][1]=Pr0=2;
    if(Pr1>450&&Pr1<452)Lin[16][1]=Pr0=2;
    if(Pr1>575&&Pr1<577)Lin[17][1]=Pr0=2;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(Pr1>580&&Pr1<582)Lin[18][1]=Pr0=2;
if(Pr1>585&&Pr1<587)Lin[19][1]=Pr0=2;
if(Pr1>590&&Pr1<592)Lin[20][1]=Pr0=2;
if(Pr1>670&&Pr1<672)Lin[21][1]=Pr0=2;
if(Pr1>635&&Pr1<637)Lin[22][1]=Pr0=2;
if(Pr1>640&&Pr1<642)Lin[23][1]=Pr0=2;
if(Pr1>645&&Pr1<647)Lin[24][1]=Pr0=2;
if(Pr1>650&&Pr1<652)Lin[25][1]=Pr0=2;
if(Pr1>730&&Pr1<732)Lin[26][1]=Pr0=2;
if(Pr1>700&&Pr1<702)Lin[27][1]=Pr0=2;
if(Pr1>730&&Pr1<732)Lin[28][1]=Pr0=2;
if(Pr1>750&&Pr1<752)Lin[29][1]=Pr0=2;
if(Pr1>746&&Pr1<748)Lin[30][1]=Pr0=2;
if(Pr1>776&&Pr1<778)Lin[31][1]=Pr0=2;
if(Pr1>796&&Pr1<798)Lin[32][1]=Pr0=2;
if(Pr1>508&&Pr1<510)Lin[33][1]=Pr0=2;
if(Pr1>515&&Pr1<517)Lin[34][1]=Pr0=2;
if(Pr1>535&&Pr1<537)Lin[35][1]=Pr0=2;
if(Pr1>540&&Pr1<542)Lin[36][1]=Pr0=2;
if(Pr1>560&&Pr1<562)Lin[37][1]=Pr0=2;
if(Pr1>565&&Pr1<567)Lin[38][1]=Pr0=2;
if(Pr1>558&&Pr1<560)Lin[39][1]=Pr0=2;
if(Pr1>565&&Pr1<567)Lin[40][1]=Pr0=2;
if(Pr1>585&&Pr1<587)Lin[41][1]=Pr0=2;
if(Pr1>600&&Pr1<602)Lin[42][1]=Pr0=2;
if(Pr1>900&&Pr1<902){ Lin[43][1]=Pr0=2;
if(pos!=3){ Vale_Hor(217,58,4); Vale_Hor(217,73,4);}
if(pos!=2){ Vale_Hor(217,278,4); Vale_Hor(217,293,4);}
Sound(600);}
if(Pr1>940&&Pr1<942)Lin[44][1]=Pr0=2;
if(Pr1>955&&Pr1<957)Lin[45][1]=Pr0=2;
if(Pr1>970&&Pr1<972)Lin[46][1]=Pr0=2;
if(Pr1>1210&&Pr1<1212)Lin[47][1]=Pr0=2;
if(Pr1>1215&&Pr1<1217)Lin[48][1]=Pr0=2;
if(Pr1>1230&&Pr1<1232)Lin[49][1]=Pr0=2;
if(Pr1>1250&&Pr1<1252)Lin[50][1]=Pr0=2;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(Pr1>1270&&Pr1<1272)Lin[51][1]=Pr0=2;
if(Pr1>1340&&Pr1<1342)Lin[52][1]=Pr0=2;
if(Pr1>1400&&Pr1<1402)Lin[53][1]=Pr0=2;
if(Pr1>1030&&Pr1<1032)Lin[54][1]=Pr0=2;
if(Pr1>1045&&Pr1<1047)Lin[55][1]=Pr0=2;
if(Pr1>1055&&Pr1<1057){ Lin[57][1]=Pr0=2;
    setcolor(15); setfillstyle(1,12);
if(pos!=2){ fillellipse(220,329,4,4); fillellipse(220,389,4,4);}
if(pos!=3){ fillellipse(220,169,4,4); fillellipse(220,109,4,4);}
    Sound(300);}
if(Pr1>1090&&Pr1<1092)Lin[56][1]=Pr0=2;
if(Pr1>1105&&Pr1<1107)Lin[58][1]=Pr0=2;
if(Pr1>1125&&Pr1<1127)Lin[59][1]=Pr0=2;
if(Pr1>1410&&Pr1<1412)Lin[60][1]=Pr0=2;
if(Pr1>1570&&Pr1<1572)Lin[61][1]=Pr0=2;
if(Pr1>900&&Pr1<902)Lin[62][1]=Pr0=2;
if(Pr1>930&&Pr1<932)Lin[63][1]=Pr0=2;
if(Pr1>970&&Pr1<972)Lin[64][1]=Pr0=2;
if(Pr1>1400&&Pr1<1402)Lin[65][1]=Pr0=2;
if(Pr1>1420&&Pr1<1422)Lin[66][1]=Pr0=2;
if(Pr1>1445&&Pr1<1447)Lin[67][1]=Pr0=2;
if(Pr1>1460&&Pr1<1462)Lin[68][1]=Pr0=2;
if(Pr1>1260&&Pr1<1262)Lin[69][1]=Pr0=2;
if(Pr1>1320&&Pr1<1322)Lin[70][1]=Pr0=2;
if(Pr1>1495&&Pr1<1497)Lin[71][1]=Pr0=2;
if(Pr1>1800&&Pr1<1802){ Lin[72][1]=Pr0=2;
if(pos!=2){ Vale_Hor(442,207,4); Vale_Hor(442,187,4);}
if(pos!=3){ Vale_Hor(442,197,4); Vale_Hor(442,177,4);}
    Sound(600); }
if(Pr1>1060&&Pr1<1062)Lin[73][1]=Pr0=2;
if(Pr1>1090&&Pr1<1092)Lin[74][1]=Pr0=2;
if(Pr1>1110&&Pr1<1112)Lin[75][1]=Pr0=2;
if(Pr1>1130&&Pr1<1132)Lin[76][1]=Pr0=2;
if(Pr1>1400&&Pr1<1402)Lin[77][1]=Pr0=2;
if(Pr1>1400&&Pr1<1402)Lin[78][1]=Pr0=2;
if(Pr1>1570&&Pr1<1572)Lin[79][1]=Pr0=2;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(Pr1>1600&&Pr1<1602)Lin[80][1]=Pr0=2;
if(Pr1>1390&&Pr1<1392)Lin[81][1]=Pr0=2;
if(Pr1>1415&&Pr1<1417)Lin[82][1]=Pr0=2;
if(Pr1>1555&&Pr1<1557)Lin[83][1]=Pr0=2;
if(Pr1>1570&&Pr1<1572)Lin[84][1]=Pr0=2;
if(Pr1>1630&&Pr1<1632)Lin[85][1]=Pr0=2;
if(Pr1>1900&&Pr1<1902)Lin[86][1]=Pr0=2;
if(Pr1>1930&&Pr1<1932)Lin[87][1]=Pr0=2;
if(Pr1>1940&&Pr1<1942)Lin[88][1]=Pr0=2;
if(Pr1>1980&&Pr1<1982){ Lin[89][1]=Pr0=2;
setcolor(15); setfillstyle(1,12);
fillellipse(530,375,5,5);fillellipse(545,375,5,5);
Sound(500); }
if(Pr1>1990&&Pr1<1992)Lin[90][1]=Pr0=2;
if(Pr1>1995&&Pr1<1997)Lin[91][1]=Pr0=2;
if(Pr1>2035&&Pr1<2037)Lin[92][1]=Pr0=2;
if(Pr1>2210&&Pr1<2212)Lin[93][1]=Pr0=2;
if(Pr1>2235&&Pr1<2237)Lin[94][1]=Pr0=2;
if(Pr1>2245&&Pr1<2247)Lin[95][1]=Pr0=2;
if(Pr1>2390&&Pr1<2392)Lin[96][1]=Pr0=2;
if(Pr1>2405&&Pr1<2407)Lin[97][1]=Pr0=2;
if(Pr1>2470&&Pr1<2472)Lin[98][1]=Pr0=2;
if(Pr1>2525&&Pr1<2527)Lin[99][1]=Pr0=2;
if(Pr1>2500&&Pr1<2502){ Lin[100][1]=Pr0=2;
if(pos!=2){ Vale_Ver(282,243,4); Vale_Ver(272,176,4);}
if(pos!=3){ Vale_Ver(282,213,4); Vale_Ver(272,148,4); }
Vale_Hor(442,207,1); Vale_Hor(442,187,1);
Vale_Hor(442,197,1); Vale_Hor(442,177,1);
Sound(400); }
if(Pr1>2525&&Pr1<2527)Lin[101][1]=Pr0=2;
if(Pr1>2540&&Pr1<2542)Lin[102][1]=Pr0=2;
if(Pr1>2610&&Pr1<2612)Lin[103][1]=Pr0=2;
if(Pr1>2690&&Pr1<2692)Lin[104][1]=Pr0=2;
if(Pr1>2870&&Pr1<2872)Lin[105][1]=Pr0=2;
if(Pr1>2970&&Pr1<2972){ Lin[106][1]=Pr0=2;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    Sound(300); }

    if(Pr1>2550&&Pr1<2552)Lin[107][1]=Pr0=2;
    if(Pr1>2610&&Pr1<2612)Lin[108][1]=Pr0=2;
    if(Pr1>2670&&Pr1<2672)Lin[109][1]=Pr0=2;
    if(Pr1>2895&&Pr1<2897)Lin[110][1]=Pr0=2;
    if(Pr1>2970&&Pr1<2972)Lin[111][1]=Pr0=2;
    if(Pr1>2760&&Pr1<2762)Lin[112][1]=Pr0=2;
    if(Pr1>2815&&Pr1<2817)Lin[113][1]=Pr0=2;
    if(Pr1>2915&&Pr1<2917)Lin[114][1]=Pr0=2;
    if(Pr1>2925&&Pr1<2927)Lin[115][1]=Pr0=2;
    if(Pr1>2935&&Pr1<2937)Lin[116][1]=Pr0=2;
    if(Pr1>2945&&Pr1<2947)Lin[117][1]=Pr0=2;
    if(Pr1>3150&&Pr1<3152){ Lin[118][1]=Pr0=2;
        Vale_Ver(378,217,4); Vale_Hor(492,228,4);
        Sound(300); }
    if(Pr1>3210&&Pr1<3212)Lin[119][1]=Pr0=2;
    if(Pr1>3030&&Pr1<3032)Lin[120][1]=Pr0=2;
    if(Pr1>3100&&Pr1<3102)Lin[121][1]=Pr0=2;
    if(Pr1>3150&&Pr1<3152)Lin[122][1]=Pr0=2;
    if(Pr1>3160&&Pr1<3162)Lin[123][1]=Pr0=2;
    if(Pr1>3195&&Pr1<3197)Lin[124][1]=Pr0=2;
}

void follow()
{ int h;
  /* Combustion & flow Stack */
  if(pos!=3){
    Line_Ver(32,140,151,14,1,3,0); Line_Ver(32,183,194,14,1,3,0);
    Line_Ver(63,186,198,12,1,3,0); setlinestyle(0,0,1); setcolor(15);
    if(Pr2==140){ setfillstyle(1,12);filellipse(140,214,2,2); Sound(700);}
        if(Pr2>145&&Pr2<315){ h=Pr2; setcolor(14);
            line(141,355-h,145,355-h); }
    if(Pr2>1500&&Pr2<1670){ h=Pr2; setcolor(9);
        line(141,40+h-1500,145,40+h-1500); }
    if(Pr2==1500){ setcolor(15);setfillstyle(1,12); filellipse(146,219,2,2);
        setfillstyle(1,9); filellipse(140,214,2,2); Sound(700); }
    Line_Ver(77,185,217,12,4,3,1); Line_Hor(77,218,132,12,4,3,2);

```

```

}
if(Pr2>1500&&Pr2<1700&&pos!=3)Heat(0);
if(Pr2>1500&&Pr2<1700&&pos!=2)Heat(220);
if(Pr2>1080&&Pr2<1300&&pos!=3)Dump_W(0);
if(Pr2>1080&&Pr2<1300&&pos!=2)Dump_W(220);
if(pos!=2){
Line_Ver(32,360,371,14,1,3,0); Line_Ver(32,403,414,14,1,3,0);
Line_Ver(63,406,418,12,1,3,0); setlinestyle(0,0,1); setcolor(15);
if(Pr2==140){ setfillstyle(1,12); fillellipse(140,434,2,2);Sound(700);}
if(Pr2>145&&Pr2<315){ h=Pr2; setcolor(14);line(141,575-h,145,575-h); }
if(Pr2>1500&&Pr2<1670){ h=Pr2; setcolor(9);
line(141,260+h-1500,145,260+h-1500); }
if(Pr2==1500){ setcolor(15); setfillstyle(1,12); fillellipse(146,439,2,2);
setfillstyle(1,9); fillellipse(140,434,2,2); Sound(700);}
Line_Ver(77,405,437,12,4,3,1); Line_Hor(77,438,132,12,4,3,2); }
if(pos!=5||Pr1<3345){
/* Cooling Tower */
Line_Hor(580,301,575,3,2,3,3); Line_Hor(561,301,512,3,2,3,4);
Line_Hor(512,307,548,3,2,3,5); Line_Hor(548,313,512,3,2,3,6);
Line_Hor(512,319,579,3,2,3,7);
/* flow Fwp Hp */
Line_Ver(445,331,367,12,4,3,9); Line_Hor(448,362,461,12,4,3,10);
Line_Hor(445,362,432,12,4,3,10); Line_Ver(460,362,367,12,4,3,11);
Line_Ver(430,362,367,12,4,3,11); Line_Ver(460,380,386,12,4,3,12);
Line_Hor(461,388,446,12,4,3,12); Line_Ver(445,381,398,12,4,3,12);
Line_Hor(445,398,338,12,4,3,13); Line_Ver(338,397,208,12,4,3,16);
Line_Ver(338,205,198,12,4,3,22); Line_Ver(338,195,188,12,4,3,23);
Line_Ver(338,185,178,12,4,3,24); Line_Ver(338,173,90,12,4,3,25);
Line_Ver(338,84,72,12,4,3,26); Line_Hor(336,73,315,12,4,3,30);
Line_Hor(308,73,296,12,4,3,31); Line_Hor(290,73,224,12,4,3,32);
Line_Hor(335,293,314,12,4,3,39); Line_Hor(308,293,284,12,4,3,40);
Line_Hor(280,293,276,12,4,3,41); Line_Hor(270,293,224,12,4,3,42);
/* flow Fwp Lp */
Line_Ver(405,331,352,12,4,3,8); Line_Hor(406,347,419,12,4,3,10);
Line_Hor(402,347,389,12,4,3,10); Line_Ver(420,347,352,12,4,3,11);
Line_Ver(390,347,352,12,4,3,11); Line_Ver(420,365,370,12,4,3,12);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Line\_Hor(422,373,408,12,4,3,12); Line\_Ver(405,368,381,12,4,3,12);  
 Line\_Hor(405,383,348,12,4,3,15); Line\_Ver(348,383,209,12,4,3,14);  
 Line\_Ver(348,205,198,12,4,3,17); Line\_Ver(348,195,188,12,4,3,18);  
 Line\_Ver(348,185,178,12,4,3,19); Line\_Ver(348,173,90,12,4,3,20);  
 Line\_Ver(348,84,57,12,4,3,21); Line\_Hor(346,58,315,12,4,3,27);  
 Line\_Hor(308,58,296,12,4,3,28); Line\_Hor(290,58,224,12,4,3,29);  
 Line\_Hor(346,278,342,12,4,3,33); Line\_Hor(335,278,314,12,4,3,34);  
 Line\_Hor(308,278,284,12,4,3,35); Line\_Hor(280,278,276,12,4,3,37);  
 Line\_Hor(270,278,224,12,4,3,38);

/\* flow Hp Steam \*/

Line\_Ver(282,217,237,12,4,3,100); Line\_Ver(282,237,217,12,4,3,100);  
 Line\_Hor(282,228,306,12,4,3,101); Line\_Ver(307,228,165,12,4,3,102);  
 Line\_Ver(307,158,85,12,4,3,103); Line\_Hor(307,87,480,12,4,3,104);  
 Line\_Ver(480,88,101,12,4,3,105); Line\_Ver(480,112,125,12,4,3,106);

/\* flow Lp Steam \*/

Line\_Ver(272,150,170,12,4,3,100); Line\_Ver(272,170,150,12,4,3,100);  
 Line\_Hor(272,162,292,12,4,3,99); Line\_Ver(292,160,104,12,4,3,107);  
 Line\_Ver(292,96,40,12,4,3,108); Line\_Hor(292,42,510,12,4,3,109);  
 Line\_Ver(510,42,100,12,4,3,110); Line\_Ver(510,110,120,12,4,3,111);

/\* flow Condensate to Deaerator \*/

Line\_Ver(530,342,368,12,4,3,86); Line\_Hor(530,362,542,12,4,3,87);  
 Line\_Hor(527,362,515,12,4,3,87); Line\_Ver(545,362,368,12,4,3,88);  
 Line\_Ver(515,362,368,12,4,3,88); Line\_Ver(545,380,386,12,4,3,89);  
 Line\_Hor(547,388,532,12,4,3,90);  
 Line\_Ver(530,380,413,12,4,3,91); Line\_Hor(530,413,358,12,4,3,92);  
 Line\_Ver(358,413,400,12,4,3,93); Line\_Ver(358,395,385,12,4,3,94);  
 Line\_Ver(358,380,242,12,4,3,95); Line\_Hor(356,242,375,12,4,3,96);  
 Line\_Hor(380,242,425,12,4,3,97); Line\_Ver(425,242,253,12,4,3,98);

/\* flow Steam to Deaerator \*/

Line\_Ver(530,168,228,12,4,3,120); Line\_Hor(528,228,497,12,4,3,121);  
 Line\_Hor(488,228,378,12,4,3,122); Line\_Ver(471,229,266,12,4,3,123);  
 Line\_Hor(471,267,458,12,4,3,124);

/\* flow Lp Steam to Deaerator \*/

Line\_Ver(378,44,82,12,4,3,112); Line\_Ver(378,88,173,12,4,3,113);  
 Line\_Ver(378,178,185,12,4,3,114); Line\_Ver(378,188,195,12,4,3,115);  
 Line\_Ver(378,198,205,12,4,3,116); Line\_Ver(378,207,211,12,4,3,117);

Line\_Ver(378,221,266,12,4,3,118); Line\_Hor(378,267,392,12,4,3,119);

if(pos!=3){ /\*flow Drump 11\*/

Line\_Hor(212,73,168,12,4,3,62); Line\_Ver(167,73,147,12,4,3,63);

Line\_Hor(165,149,226,12,4,3,64);

Line\_Ver(235,144,139,12,4,3,47); Line\_Hor(235,140,262,12,4,3,48);

Line\_Ver(262,140,160,12,4,3,49); Line\_Ver(262,166,186,12,4,3,50);

Line\_Hor(262,187,166,12,4,3,69); Line\_Hor(166,193,282,12,4,3,70);

Line\_Ver(235,160,172,12,4,3,54); Line\_Hor(235,174,227,12,4,3,55);

Line\_Hor(214,174,170,12,4,3,73); Line\_Hor(168,168,191,12,4,3,74);

Line\_Hor(193,162,170,12,4,3,75); Line\_Hor(168,156,226,12,4,3,76);

Line\_Ver(282,194,206,12,4,3,77);

/\*flow Drump 12\*/

Line\_Hor(212,58,176,12,4,3,43); Line\_Ver(176,58,69,12,4,3,44);

Line\_Ver(176,75,87,12,4,3,45); Line\_Hor(177,89,226,12,4,3,46);

Line\_Ver(235,82,77,12,4,3,47); Line\_Hor(235,80,262,12,4,3,48);

Line\_Ver(262,80,100,12,4,3,49); Line\_Ver(262,106,126,12,4,3,50);

Line\_Hor(262,127,177,12,4,3,51); Line\_Hor(175,133,272,12,4,3,52);

Line\_Ver(272,131,142,12,4,3,53);

Line\_Ver(235,100,112,12,4,3,54); Line\_Hor(235,114,227,12,4,3,55);

Line\_Hor(214,114,178,12,4,3,57); Line\_Hor(175,108,191,12,4,3,56);

Line\_Hor(194,102,178,12,4,3,58); Line\_Hor(175,96,226,12,4,3,59);

/\* flow Condensate \*/

Line\_Hor(273,137,290,12,4,1,65); Line\_Hor(294,137,303,12,4,1,66);

Line\_Hor(308,137,319,12,4,1,67); Line\_Ver(321,136,176,12,4,1,68);

Line\_Hor(320,177,435,12,4,1,71); Line\_Hor(446,177,505,12,4,1,72);

Line\_Hor(283,197,302,12,4,1,81); Line\_Hor(309,197,435,12,4,1,82);

Line\_Hor(446,197,505,12,4,1,72);

}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(pos!=2){ /*flow Drump 21*/
Line_Hor(212,293,168,12,4,3,62);
Line_Ver(167,293,367,12,4,3,63); Line_Hor(165,369,226,12,4,3,64);

Line_Ver(235,364,359,12,4,3,47); Line_Hor(234,360,262,12,4,3,48);
Line_Ver(262,360,380,12,4,3,49); Line_Ver(262,386,406,12,4,3,50);
Line_Hor(262,407,166,12,4,3,69); Line_Hor(166,413,280,12,4,3,70);

Line_Ver(235,380,392,12,4,3,54); Line_Hor(235,394,227,12,4,3,55);
Line_Hor(214,394,170,12,4,3,73); Line_Hor(168,388,191,12,4,3,74);
Line_Hor(193,382,170,12,4,3,75); Line_Hor(168,376,226,12,4,3,76);
Line_Ver(282,412,248,12,4,3,78);

/*flow Drump 22*/
Line_Hor(212,278,176,12,4,3,43);
Line_Ver(176,278,289,12,4,3,44); Line_Ver(176,295,307,12,4,3,45);
Line_Hor(177,309,226,12,4,3,46);

Line_Ver(235,303,298,12,4,3,47); Line_Hor(235,300,262,12,4,3,48);
Line_Ver(262,300,320,12,4,3,49); Line_Ver(262,326,346,12,4,3,50);
Line_Hor(262,347,177,12,4,3,51); Line_Hor(175,353,272,12,4,3,52);

Line_Ver(235,320,332,12,4,3,54); Line_Hor(235,334,227,12,4,3,55);
Line_Hor(214,334,176,12,4,3,57); Line_Hor(175,328,191,12,4,3,56);
Line_Hor(194,322,176,12,4,3,58); Line_Hor(175,316,226,12,4,3,59);
Line_Ver(272,352,196,12,4,3,60); Line_Ver(272,187,180,12,4,3,61);

/* flow Condensate */
Line_Hor(274,187,302,12,4,1,79); Line_Hor(309,187,435,12,4,1,80);
Line_Hor(446,187,505,12,4,1,72);

Line_Hor(283,260,319,12,4,1,83); Line_Ver(321,260,208,12,4,1,84);
Line_Hor(320,207,435,12,4,1,85); Line_Hor(446,207,505,12,4,1,72);
}
}

```

```
if(Pr2>0&&Pr2%2==0&&Pr2<300)Num_Gen(1);
```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (Pr2>0&&Pr2%2==0&&Pr2<300)Num\_Gen(2); เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(Pr1>2970&&Pr1<3300&&Pr1%5==0)Num_Gen(3);
if(Pr1>2970&&Pr1<3300&&Pr1%5==0)Num_Gen(4);
if(Pr1==2500)
{ Line_Hor(446,177,505,0,0,1,72);
Line_Hor(446,197,505,0,0,1,72);
Line_Hor(446,187,505,0,0,1,72);
Line_Hor(446,207,505,0,0,1,72);
Lin[72][1]=0; } P_STOP=0; if(Pr2<50)P_STOP=1;
if(Pr2>120&&Pr2<140)P_STOP=1;
if(Pr2>1095&&Pr2<1300)P_STOP=1;
if(Pr2>1480&&Pr2<1700)P_STOP=1;
if(Pr2<3000&&pos!=4&&pos!=5)Pr2++;
Al++; if(Al==2)Al=0; if(Pr1<5000&&P_STOP==0||pos==4||pos==5)Pr1++;
if(pos!=4||pos!=5){ Condition();/* delay(DELAY);*/ }
if(pos==4||pos==5) Alarm();
}
void Alarm()
{ delay(300);
if(Pr1==3305){ Num_Gen(1); Num_Gen(2); Num_Gen(3);
Iri_Vale(1); Iri_Pump(1); setcolor(15); setfillstyle(1,12);
fillellipse(110,353,2,2); fillellipse(110,133,2,2); Heat(1);
setfillstyle(1,9); bar(245,109,254,119); bar(245,169,254,179);
bar(245,329,254,339); bar(245,389,254,399); settextrjust(1,1);
setfillstyle(1,10); outportb(0x30C,0xC2);
outportb(0x30D,0xC2); outportb(0x30E,0xC2);
outportb(0x30F,0xC2); outportb(0x310,0xC2);
outportb(0x311,0xC2); outportb(0x312,0xC2);
outportb(0x306,0xCC); outportb(0x308,0xDA);
outportb(0x309,0xCC); outportb(0x30A,0xDA);
bar(492,110,497,90); bar(522,110,527,90); bar(585,110,590,90);
bar(230,50,235,30); bar(245,50,250,30); bar(245,270,250,250);
bar(230,270,235,250); setcolor(15); setfillstyle(1,12);
fillellipse(146,219,2,2); fillellipse(146,439,2,2);
setfillstyle(1,9); settextrstyle(0,0,1);
outtextxy(249,115,"4"); outtextxy(249,175,"5");
outtextxy(249,335,"4"); outtextxy(249,395,"5"); Sound(300);}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(pos==4){ setcolor(15); setfillstyle(1,12);
    if(Pr1==3320){ fillellipse(430,375,5,5);Sound(800);
Line_Ver(430,380,386,12,4,3,12); Line_Hor(429,388,444,12,4,3,12);}
    if(Pr1==3350){ fillellipse(390,360,5,5);Sound(800);
Line_Ver(390,365,370,12,4,3,12); Line_Hor(389,373,406,12,4,3,12);}
    if(Pr1==3380){ fillellipse(220,179,4,4);Sound(800); }
    if(Pr1==3410){ fillellipse(220,119,4,4);Sound(800); }
    if(Pr1==3440){ fillellipse(220,399,4,4);Sound(800); }
    if(Pr1==3470){ fillellipse(220,339,4,4);Sound(800); }
    if(Pr1==3500){ fillellipse(515,375,5,5);Sound(800);
Line_Ver(515,380,386,12,4,3,89); Line_Hor(514,388,532,12,4,3,90); }
    setcolor(12); setfillstyle(1,11);
    if(Pr1==3315){ fillellipse(445,375,5,5);Sound(500); }
    if(Pr1==3345){ fillellipse(405,360,5,5);Sound(500); }
    if(Pr1==3375){ fillellipse(220,169,4,4);Sound(500); }
    if(Pr1==3405){ fillellipse(220,109,4,4);Sound(500); }
    if(Pr1==3435){ fillellipse(220,389,4,4);Sound(500); }
    if(Pr1==3465){ fillellipse(220,329,4,4);Sound(500); }
    if(Pr1==3490){ fillellipse(530,375,5,5);Sound(500); }
    }
if(pos==5){
    if(Pr1==3330){ setcolor(12); setfillstyle(1,11);
        fillellipse(220,169,4,4);Sound(500);
        fillellipse(220,109,4,4);Sound(500);
        fillellipse(220,389,4,4);Sound(500);
        fillellipse(220,329,4,4);Sound(500); }
        if(Pr1>3340) Down();
    if(Pr1==3340){ Vale_Hor(442,207,4);Sound(800);
        Vale_Hor(442,197,4);Sound(800); Vale_Hor(442,187,4);Sound(800);
        Vale_Hor(442,177,4);Sound(800); }
    if(Pr1==3350){ Vale_Ver(272,148,1); Vale_Ver(272,176,1);
Sound(600); Vale_Ver(282,213,1); Vale_Ver(282,243,1);Sound(600); }
    if(Pr1==3370){ setfillstyle(1,1); setcolor(14);
        fillellipse(405,360,5,5); fillellipse(420,360,5,5); Sound(500);
        fillellipse(445,375,5,5); fillellipse(460,375,5,5); Sound(500); }
    if(Pr1==3460){ setfillstyle(1,1); setcolor(14);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    fillellipse(568,301,5,5); Sound(700);
    fillellipse(530,375,5,5); fillellipse(545,375,5,5); Sound(700); }
    if(Pr1==3465){ Vale_Hor(442,207,1);Sound(800);
Vale_Hor(442,197,1);Sound(800); Vale_Hor(442,187,1);Sound(800);
Vale_Ver(480,107,1);Sound(700); Vale_Ver(510,107,1);Sound(700);
    Vale_Hor(442,177,1);Sound(800); }
    if(Pr1==3470){ Vale_Hor(217,58,1); Sound(800);
Vale_Hor(217,73,1); Sound(700); Vale_Hor(217,278,1); Sound(700);
Vale_Hor(217,293,1); Sound(700); Vale_Hor(492,228,1); Sound(700);
Vale_Ver(378,217,1); Sound(700); }
}
}
Clean_Buffer()
{ int i,j;
for(i=0;i<130;i++)
for(j=0;j<2;j++)
Lin[i][j]=0;
gen1=gen2=gen3=level=level1=level2=0;level_d=10;
outportb(0x300,0x00); outportb(0x301,0x00); outportb(0x302,0x00);
outportb(0x304,0x00); outportb(0x305,0x00); outportb(0x306,0x7E);
outportb(0x308,0x7E); outportb(0x309,0x7E); outportb(0x30A,0x7E);
outportb(0x30C,0x00); outportb(0x30D,0x00); outportb(0x30E,0x00);
outportb(0x30F,0x00); outportb(0x310,0x00); outportb(0x311,0x00);
    outportb(0x312,0x00);
}
Set_Bu()
{ int i;
for(i=0;i<130;i++)
    Lin[i][1]=1; level_d=10; l_old=0;
}
void Sound(int freq)
{ sound(freq);
delay(300); nosound(); CHECK_OUTPORT();
}
Num_Gen(int i)
{ int level[10] = {0x00,0x60,0x6F,0x75,0x80,0x90,0x99,0xA5,0xA9,0xB5};

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int leve2[10] = {0x00,0x45,0x50,0x60,0x65,0x70,0x75,0x80,0x8A,0x90};
int leve3[10] = {0x00,0x45,0x50,0x58,0x65,0x6C,0x72,0x7C,0x82,0x95};
int i;
settextjustify(1,1); settextstyle(2,0,4);
setfillstyle(1,1);
if(i==1){ if(gen1<100&&pos==1||pos==2&&gen1<100)gen1++;
if(pos==4||pos==5)gen1=100; govt(gen1,3,GEN);
bar(95,140,130,152); setcolor(15);
outtextxy(104,146,GEN);outtextxy(122,146,"MW");
if(gen1==5){ setfillstyle(1,12);
fillellipse(110,133,2,2);Sound(500); } }
if(i==2){ if(gen2<100&&pos==1||pos==3&&gen2<100)gen2++;
if(pos==4||pos==5)gen2=100; govt(gen2,3,GEN);
bar(95,360,130,372); setcolor(15);
outtextxy(104,366,GEN);outtextxy(122,366,"MW");
if(gen2==5){ setfillstyle(1,12);
fillellipse(110,353,2,2);Sound(500); } }
if(i==3){ if(gen3<50&&pos==2||pos==3&&gen3<50)gen3++;
if(gen3<100&&pos==1)gen3+=2;if(pos==4||pos==5)gen3=100;
govt(gen3,3,GEN); bar(573,115,608,127); setcolor(14);
outtextxy(582,120,GEN);outtextxy(600,120,"MW");
l=gen3/10; setfillstyle(1,10); bar(585,110,590,110-l*2);
if(gen3%10==0)outportb(0x312,leve3[l]);}
if(i==4&&level<10){ level++; setfillstyle(1,10);
if(pos==2&&level>4||pos==3&&level>4)level=4;
bar(492,110,497,110-level*2); bar(522,110,527,110-level*2);
outportb(0x310,leve1[level-1]); outportb(0x311,leve2[level-1]); }
}
void Heat(int h)
{ int x,l,Color;
int segm[6] = { 0x7E,0x0C,0xB6,0x9E,0xCC,0xDA };
if(h!=1)
{
for(x=154;x<200;x++)
{ Color = getpixel(x,235+h-Pr2+1500);
if(Color==8)putpixel(x,235+h-Pr2+1500,6); }
}

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้กับโรงเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
if(h==1)
{
for(l=30;l<440;l++)
for(x=154;x<200;x++)
{
Color = getpixel(x,470-l);
if(Color==8)putpixel(x,470-l,6);
}
if(Pr2%10==0&&Temp<5){ setcolor(15);
settextjustify(1,1); setfillstyle(1,9); settextstyle(0,0,1);
if(pos!=3){bar(244,109,254,119); outtextxy(249,115,TEMP[Temp]);
outportb(0x306,segm[Temp]); outportb(0x308,segm[Temp+1]);
bar(244,169,254,179); outtextxy(249,175,TEMP[Temp+1]);}
if(pos!=2){bar(244,329,254,339); outtextxy(249,335,TEMP[Temp]);
outportb(0x309,segm[Temp]); outportb(0x30A,segm[Temp+1]);
bar(244,389,254,399); outtextxy(249,395,TEMP[Temp+1]);}
if(Pr2%40==0)Temp++;
}
}
void Dump_W(int w)
{
int ll;
int x,y,Color;
int lev1[10] = {0x00,0x4A,0x55,0x60,0x63,0x70,0x80,0x8A,0x92,0xCF};
int lev2[10] = {0x00,0x4F,0x5F,0x6F,0x7F,0x8F,0x92,0xAF,0xBF,0xCF};
int lev3[10] = {0x00,0x4A,0x55,0x60,0x65,0x6F,0x75,0x83,0x88,0xCF};
int lev4[10] = {0x00,0x45,0x50,0x55,0x5C,0x6B,0x72,0x85,0x9F,0xCF};
y=(Pr2-1080)/9;
for(x=225;x<245;x++)
{
Color = getpixel(x,105+w-y);
if(Color==13)putpixel(x,105+w-y,14);
Color = getpixel(x,165+w-y);
if(Color==13)putpixel(x,165+w-y,14);
if(Pr1%40==0||Pr2%20==0){
if(level1<10&&w==0) { setfillstyle(1,10); level1++;
bar(230,50,235,50-level1*2); bar(245,50,250,50-level1*2);
outportb(0x30C,lev1[level1-1]); outportb(0x30D,lev2[level1-1]);}
if(level2<10&&w==220) { setfillstyle(1,10); level2++;
bar(230,270,235,270-level2*2); bar(245,270,250,270-level2*2);
outportb(0x30E,lev3[level2-1]); outportb(0x30F,lev4[level2-1]);}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
}
void CHECK_OUTPORT()
{ int PPT[5] = { 0x300,0x301,0x305,0x304,0x302 };
  int CoordinateX[40] = { 110,140,146,110,140,146,220,220,
                          220,220,220,220,220,277,277,
                          287,287,383,442,442,442,442,492,
                          390,405,420,430,445,460,515,530,
                          0,545,568,217,217,217,217,0 };

  int CoordinateY[40] = { 133,214,219,353,434,439,109,119,
                          169,179,329,339,389,399,148,176,
                          213,243,217,172,182,192,202,223,
                          360,360,360,375,375,375,375,375,
                          0,375,301,53 ,68 ,273,288,0 };

  int CD[8] = { 1,2,4,8,16,32,64,128};
  int i,j,G_Color,DP;
  for(i=0;i<5;i++)
  { DP=0;
    for(j=0;j<8;j++)
    { G_Color = getpixel(CoordinateX[j+8*i],CoordinateY[j+8*i]);
      if(G_Color==12) DP=DP+CD[j];
    }  outputb(PPT[i],DP); }
}
Down()
{ int l;
  int x,y,Color;
  int le1[10] = {0x00,0x4A,0x55,0x60,0x63,0x70,0x80,0x8A,0x92,0xCF};
  int le2[10] = {0x00,0x4F,0x5F,0x6F,0x7F,0x8F,0x92,0xAF,0xBF,0xCF};
  int le3[10] = {0x00,0x4A,0x55,0x60,0x65,0x6F,0x75,0x83,0x88,0xCF};
  int le4[10] = {0x00,0x45,0x50,0x55,0x5C,0x6B,0x72,0x85,0x9F,0xCF};
  int le5[10] = {0x00,0x60,0x6F,0x75,0x80,0x90,0x99,0xA5,0xA9,0xB5};
  int le6[10] = {0x00,0x45,0x50,0x60,0x65,0x70,0x75,0x80,0x8A,0x90};
  int le7[10] = {0x00,0x45,0x50,0x58,0x65,0x6C,0x72,0x7C,0x82,0x95};
  int segm[6] = { 0x7E,0x0C,0xB6,0x9E,0xCC,0xDA };
  if(gen3!=0){ if(gen3>0)gen3--; setfillstyle(1,1); settextstyle(2,0,4);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

govt(gen3,3,GEN); bar(573,115,608,127); setcolor(14);
outtextxy(582,120,GEN); outtextxy(600,120,"MW");
if(gen3%10==0&&level_d>0)level_d--;
    setfillstyle(1,12); bar(585,90,590,110-level_d*2);
    bar(492,90,497,110-level_d*2); bar(522,90,527,110-level_d*2);
    bar(230,30,235,50-level_d*2); bar(245,30,250,50-level_d*2);
    bar(230,250,235,270-level_d*2); bar(245,250,250,270-level_d*2);
if(gen3%10==0){
    outportb(0x312,le7[level_d]); outportb(0x310,le5[level_d]);
    outportb(0x311,le6[level_d]); outportb(0x30E,le3[level_d]);
    outportb(0x30F,le4[level_d]); outportb(0x30C,le1[level_d]);
    outportb(0x30D,le2[level_d]); }
}
if(Pr1==3345){ setfillstyle(1,12); setcolor(15); Temp1=4;
    fillellipse(140,214,2,2); fillellipse(140,434,2,2); Sound(900);
    setfillstyle(1,9); setcolor(15); Temp2=5;
    fillellipse(146,219,2,2); fillellipse(146,439,2,2); Sound(900); }
if(l_old<200)
{ l_old = Pr1-3340;
for(x=154;x<200;x++)
{ Color = getpixel(x,35+l_old);
    if(Color==6)putpixel(x,35+l_old,8);
    Color = getpixel(x,260+l_old);
    if(Color==6)putpixel(x,260+l_old,8); }
for(x=135;x<147;x++)
{ Color = getpixel(x,230-l_old);
    if(Color==9)putpixel(x,230-l_old,14);
    Color = getpixel(x,450-l_old);
    if(Color==9)putpixel(x,450-l_old,14); }
}
if(Pr1%20==0&&Temp2==0)
{ setcolor(15); settextjustify(1,1);
    setfillstyle(1,9); settextstyle(0,0,1);
    if(Temp1<0) Temp1=0;
    bar(244,109,254,119); outtextxy(249,115,TEMP[Temp1]);
    bar(244,169,254,179); outtextxy(249,175,TEMP[Temp2]);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

outportb(0x306,segm[Temp1]); outportb(0x308,segm[Temp2]);
bar(244,329,254,339); outtextxy(249,335,TEMP[Temp1]);
bar(244,389,254,399); outtextxy(249,395,TEMP[Temp2]);
outportb(0x309,segm[Temp1]); outportb(0x30A,segm[Temp2]);
Temp2--; Temp1--;}
}
□

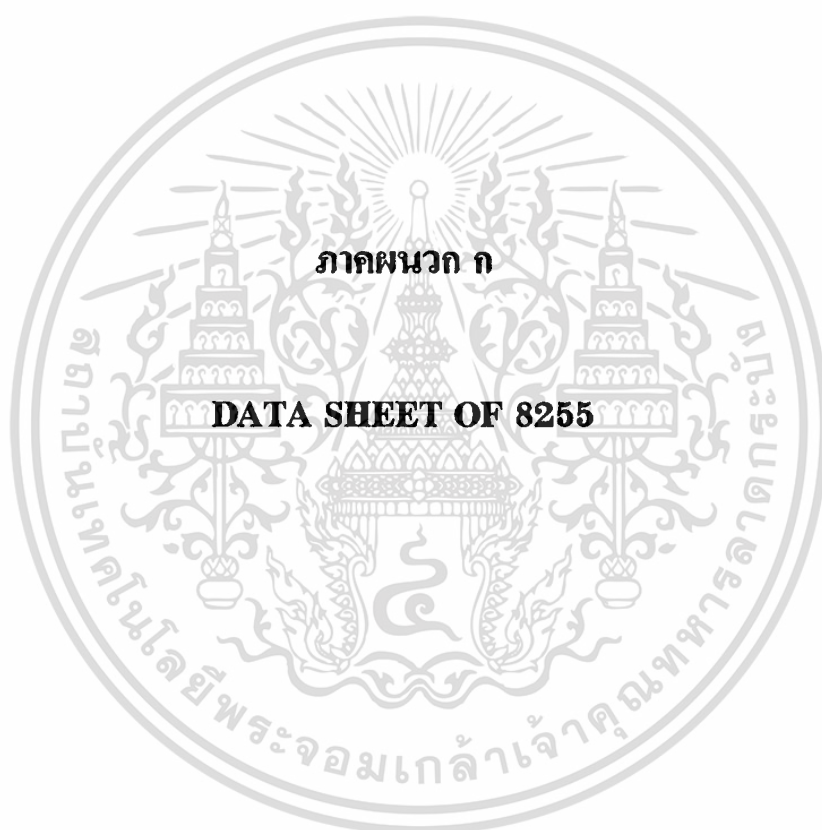
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### เอกสารอ้างอิง

- [ 1 ] หลักสูตรวิชาการทางพลังงาน ภาคความรู้เบื้องต้นของการเดินเครื่องโรงไฟฟ้า, ศูนย์ฝึกอบรมทางวิชาการ, กฟผ.
- [ 2 ] KHANOM COMBINED CYCLE POWER PLANT OPERATION, แผนกวิศวกรรมบำรุงรักษาไฟฟ้า, ฝ่ายบำรุงรักษาไฟฟ้า, กฟผ.
- [ 3 ] GENERAL SYSTEM DESCRIPTION RAYONG COMBINED CYCLE POWER PLANT, ฝ่ายฝึกอบรม, กฟผ.
- [ 4 ] มนตรี พจนารถสาลีย์, “ การเขียนโปรแกรมคอมพิวเตอร์ด้วยเทอร์โบซี ”, บริษัท ซีเอ็ด ยูเคชั่น จำกัด, พ.ศ. 2521
- [ 5 ] จิต หนูแก้ว, “ ไมโครโปรเซสเซอร์และการออกแบบเบื้องต้น ”, บริษัท ซีเอ็ด ยูเคชั่น จำกัด, พ.ศ. 2521



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## 8255A/8255A-5 PROGRAMMABLE PERIPHERAL INTERFACE

- MCS-85™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel Microprocessor Families
- Improved Timing Characteristics
- Direct Bit Set/Reset Capability Easing Control Application Interface
- Reduces System Package Count
- Improved DC Driving Capability
- Available in EXPRESS
  - Standard Temperature Range
  - Extended Temperature Range
- 40 Pin DIP Package or 44 Lead PLCC  
(See Intel Packaging Order Number: 231369)

The Intel 8255A is a general purpose programmable I/O device designed for use with Intel microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.

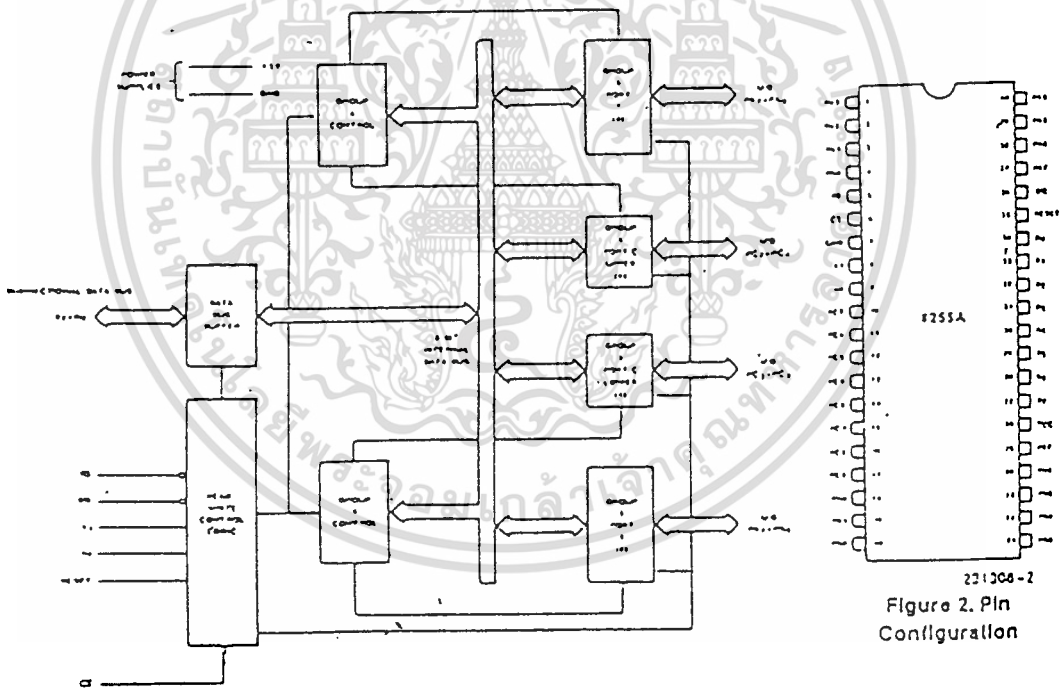


Figure 1. 8255A Block Diagram

Figure 2. Pin Configuration

September 1977

Order Number: 231304-002

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



8255A/8255A-5

## 8255A FUNCTIONAL DESCRIPTION

### General

The 8255A is a programmable peripheral interface (PPI) device designed for use in Intel microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 8255A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

### Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 8255A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

### Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the

CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

( $\overline{CS}$ )

Chip Select. A "low" on this input pin enables the communication between the 8255A and the CPU.

( $\overline{RD}$ )

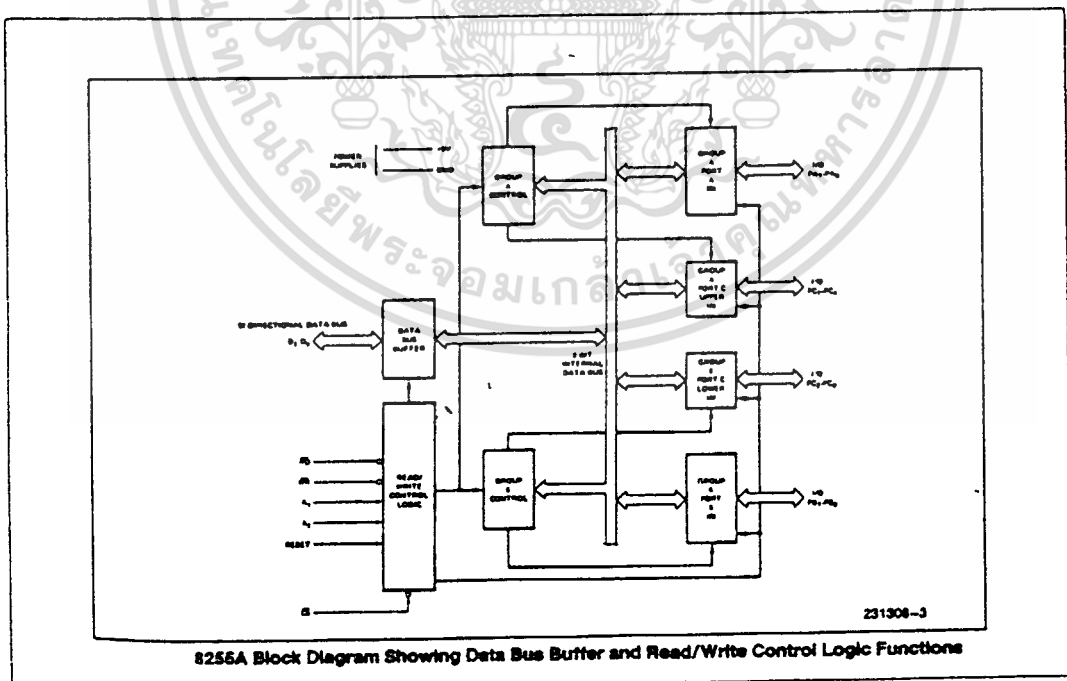
Read. A "low" on this Input pin enables the 8255A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 8255A.

( $\overline{WR}$ )

Write. A "low" on this input pin enables the CPU to write data or control words into the 8255A.

( $A_0$  and  $A_1$ )

Port Select 0 and Port Select 1. These input signals, in conjunction with the  $\overline{RD}$  and  $\overline{WR}$  inputs, control the selection of one of the three ports or the control word registers. They are normally connected to the least significant bits of the address bus ( $A_0$  and  $A_1$ ).



8255A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 8255A BASIC OPERATION

A <sub>1</sub>	A <sub>0</sub>	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	Input Operation (READ)
0	0	0	1	0	Port A → Data Bus
0	1	0	1	0	Port B → Data Bus
1	0	0	1	0	Port C → Data Bus
					Output Operation (WRITE)
0	0	1	0	0	Data Bus → Port A
0	1	1	0	0	Data Bus → Port B
1	0	1	0	0	Data Bus → Port C
1	1	1	0	0	Data Bus → Control
					Disable Function
X	X	X	X	1	Data Bus → 3-State
1	1	0	1	0	Illegal Condition
X	X	1	1	0	Data Bus → 3-State

## (RESET)

Reset. A "high" on this input clears the control register and all ports (A, B, C) are set to the input mode.

## Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 8255A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 8255A.

Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control Logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Control Group A—Port A and Port C upper (C7–C4)  
Control Group B—Port B and Port C lower (C3–C0)

The Control Word Register can Only be written into. No Read operation of the Control Word Register is allowed.

## Ports A, B, and C

The 8255A contains three 8-bit ports (A, B, and C). All can be configured in a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 8255A.

Port A. One 8-bit data output latch/buffer and one 8-bit data input latch.

Port B. One 8-bit data input/output latch/buffer and one 8-bit data input buffer.

Port C. One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B.

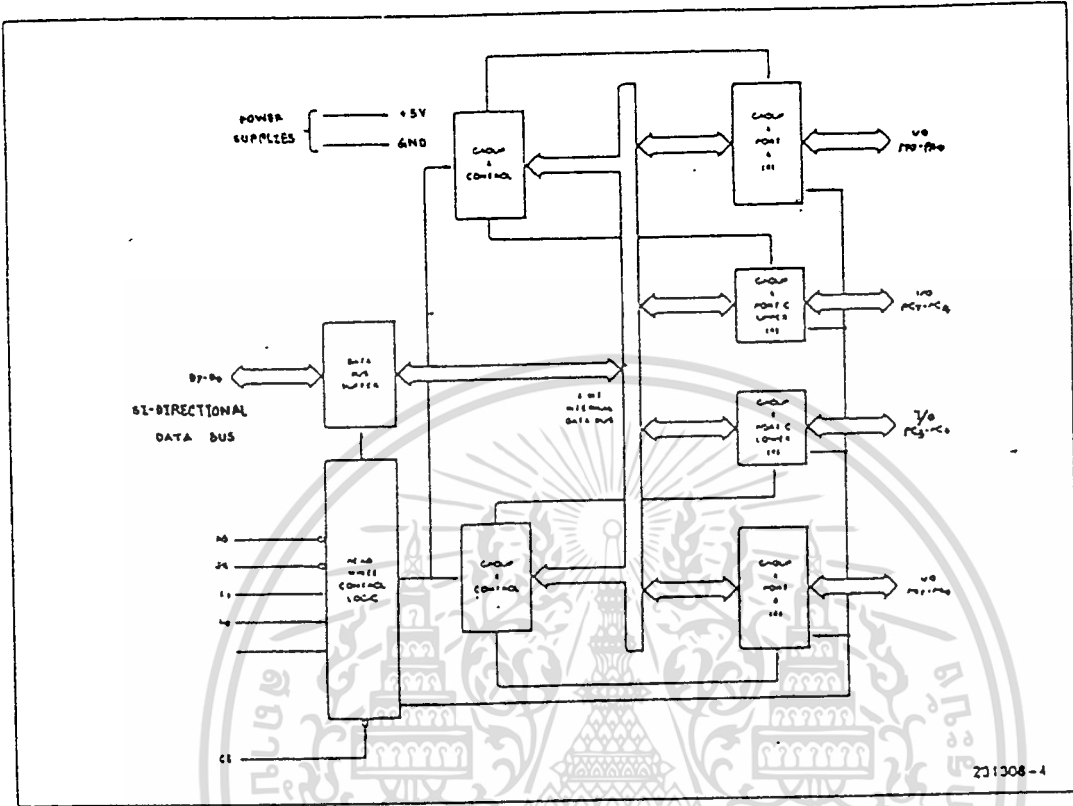
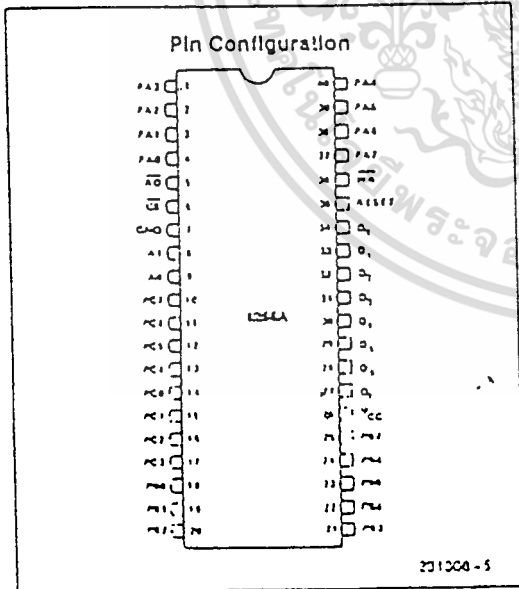


Figure 4. 8255A Block Diagram Showing Group A and Group B Control Functions



Pin Names	
D7-D0	Data Bus (Bi-Directional)
RESET	Reset Input
CS	Chip Select
RD	Read Input
WR	Write Input
A0, A1	Port Address
PA7-PA0	Port A (BIT)
PB7-PB0	Port B (BIT)
PC7-PC0	Port C (BIT)
Vcc	+ 5 Volts
GND	0 Volts

8255A OPERATIONAL DESCRIPTION

Mode Selection

There are three basic modes of operation that can be selected by the system software:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Mode 0—Basic Input/Output

Mode 1—Strobed Input/Output

Mode 2—Bi-Directional Bus

When the reset input goes "high" all ports will be set to the input mode (i.e., all 24 lines will be in the high impedance state). After the reset is removed the 8255A can remain in the input mode with no additional initialization required. During the execution of the system program any of the other modes may be selected using a single output instruction. This allows a single 8255A to service a variety of peripheral devices with a simple software maintenance routine.

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance; Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

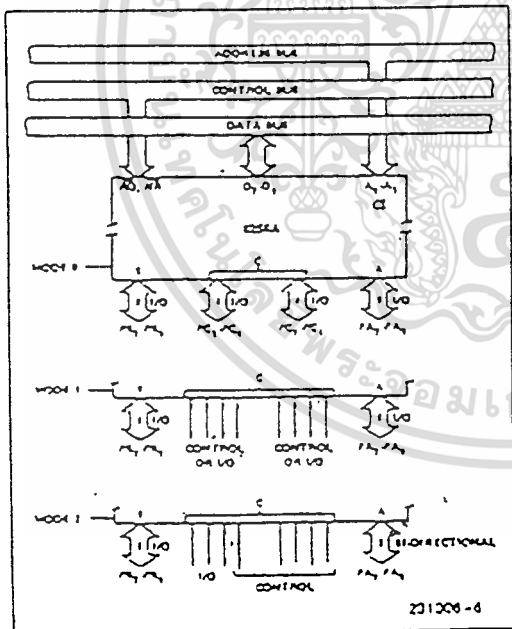


Figure 5. Basic Mode Definitions and Bus Interface

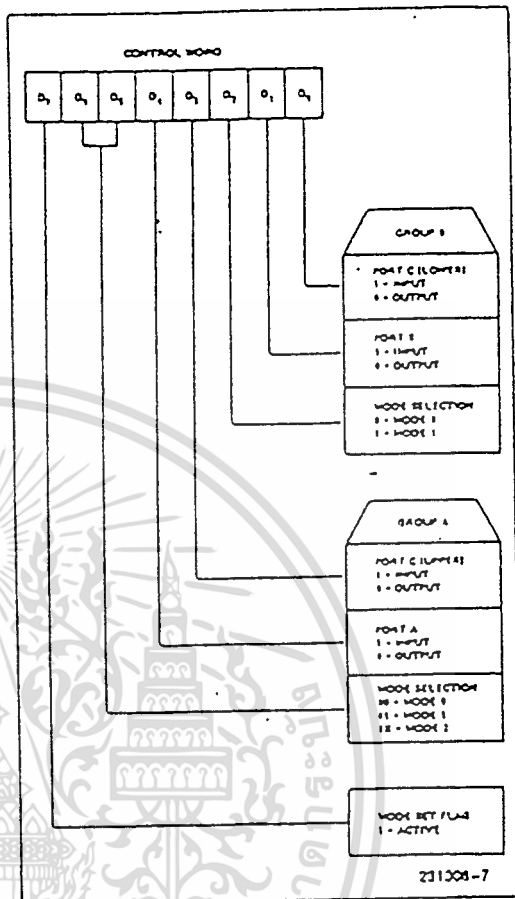


Figure 6. Mode Definition Format

The mode definitions and possible mode combinations may seem confusing at first but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 8255A has taken into account things such as efficient PC board layout, control signal definition vs PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

### Single Bit Set/Reset Feature

Any of the eight bits of Port C can be Set or Reset using a single OUTPUT Instruction. This feature reduces software requirements in Control-based applications.

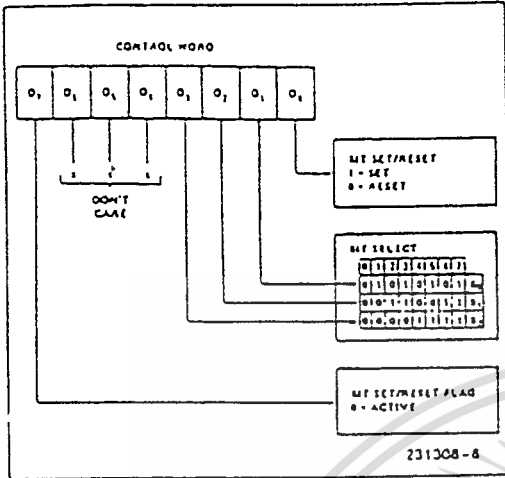


Figure 7. Bit Set/Reset Format

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were data output ports.

Interrupt Control Functions

When the 8255A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the Programmer to disallow or allow a specific I/O device to interrupt the CPU without affecting any other device in the interrupt structure.

INTE flip-flop definition:

(BIT-SET)—INTE is set—Interrupt enable

(BIT-RESET)—INTE is RESET—Interrupt disable

NOTE:

All Mask flip-flops are automatically reset during mode selection and device Reset.

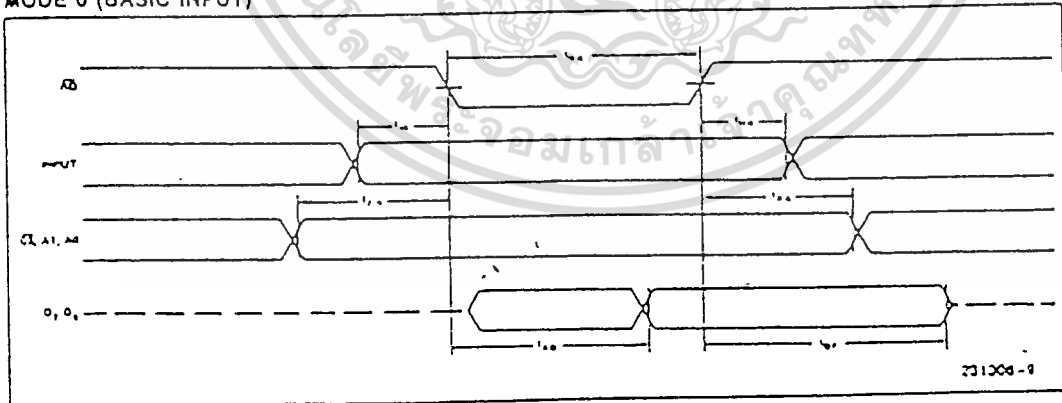
Operating Modes

MODE 0 (Basic Input/Output). This functional configuration provides simple input and output operations for each of the three ports. No "handshaking" is required, data is simply written to or read from a specified port.

Mode 0 Basic Functional Definitions:

- Two 8-bit ports and two 4-bit ports.
- Any port can be input or output.
- Outputs are latched.
- Inputs are not latched.
- 16 different Input/Output configurations are possible in this Mode.

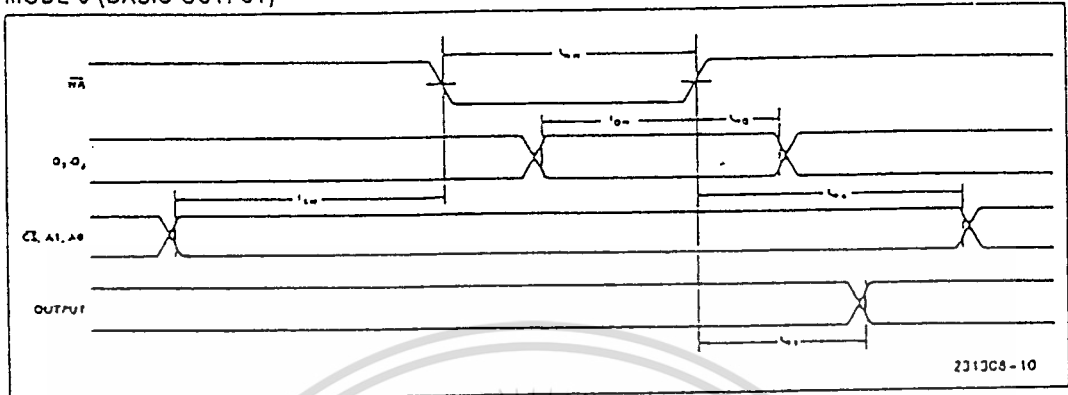
MODE 0 (BASIC INPUT)



intel®

8255A/8255A-5

## MODE 0 (BASIC OUTPUT)

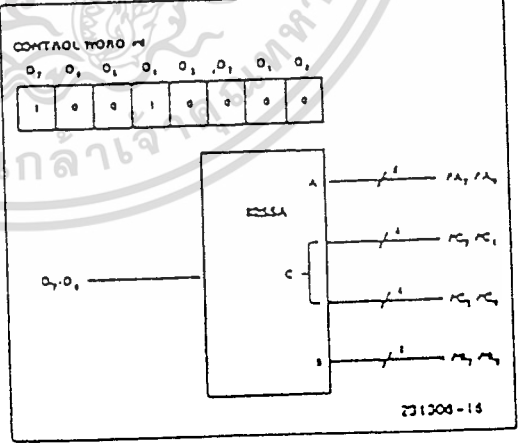
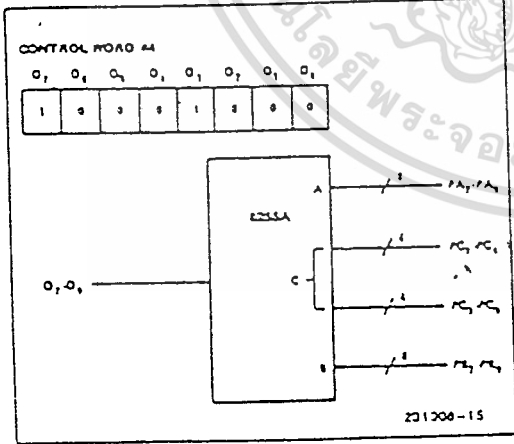
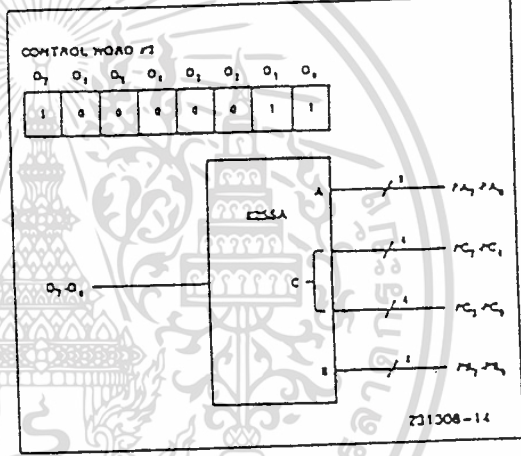
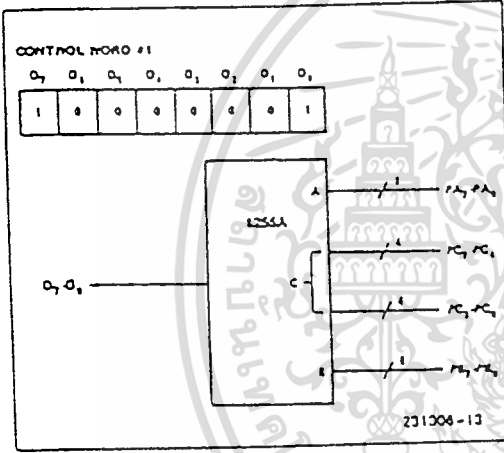
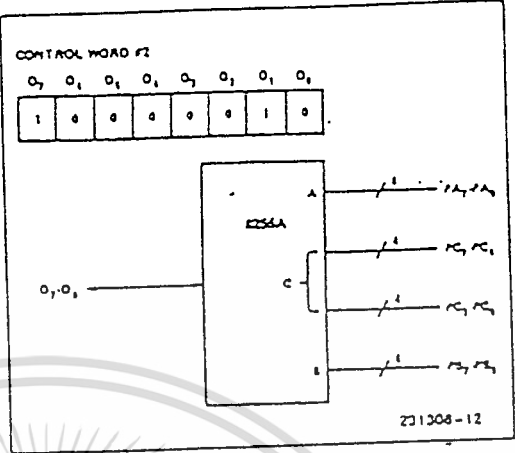
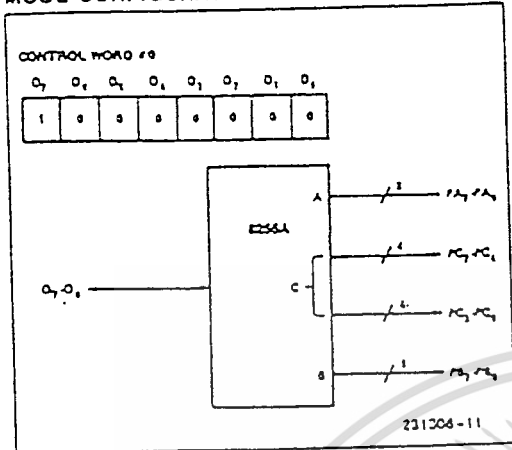


## MODE 0 PORT DEFINITION

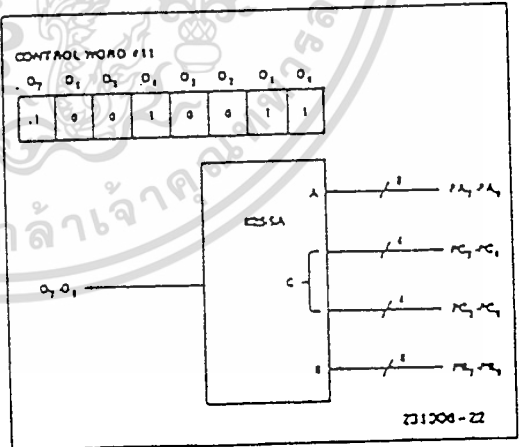
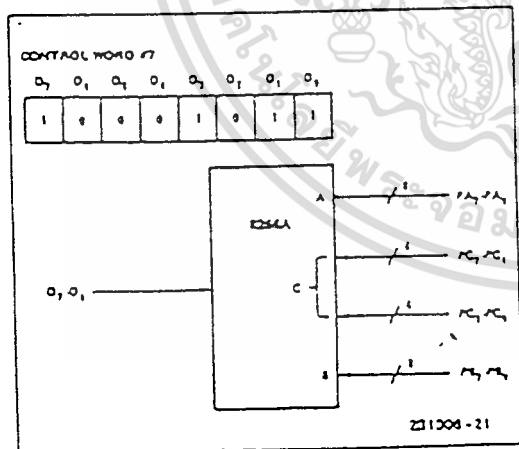
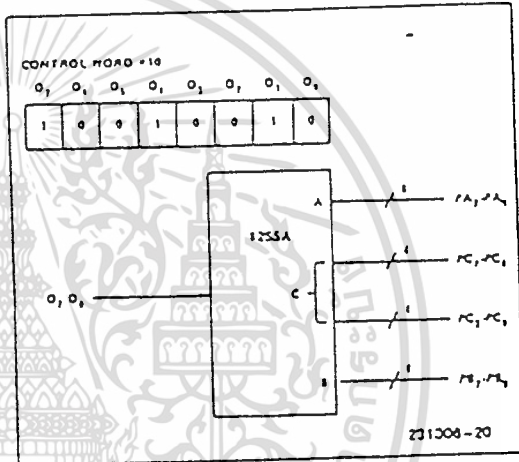
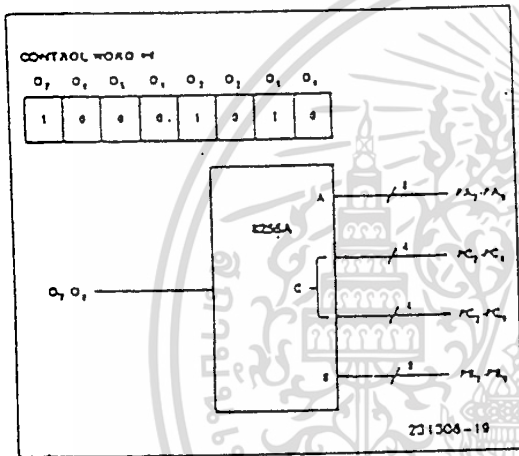
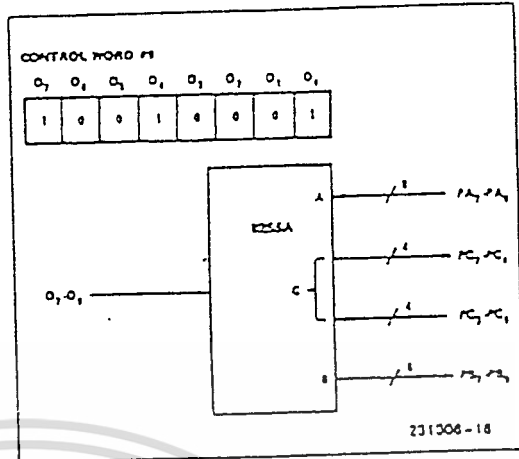
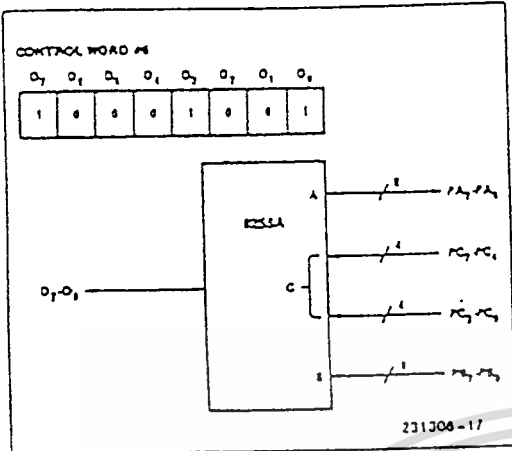
A		B		Group A			Group B	
D <sub>4</sub>	D <sub>3</sub>	D <sub>1</sub>	D <sub>0</sub>	Port A	Port C (Upper)	#	Port B	Port C (Lower)
0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUTPUT
0	0	0	1	OUTPUT	OUTPUT	1	OUTPUT	INPUT
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	OUTPUT	INPUT	4	OUTPUT	OUTPUT
0	1	0	1	OUTPUT	INPUT	5	OUTPUT	INPUT
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUTPUT
0	1	1	1	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUTPUT
1	0	0	1	INPUT	OUTPUT	9	OUTPUT	INPUT
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	INPUT	OUTPUT	11	INPUT	INPUT
1	1	0	0	INPUT	INPUT	12	OUTPUT	OUTPUT
1	1	0	1	INPUT	INPUT	13	OUTPUT	INPUT
1	1	1	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	INPUT	INPUT	15	INPUT	INPUT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MODE CONFIGURATIONS



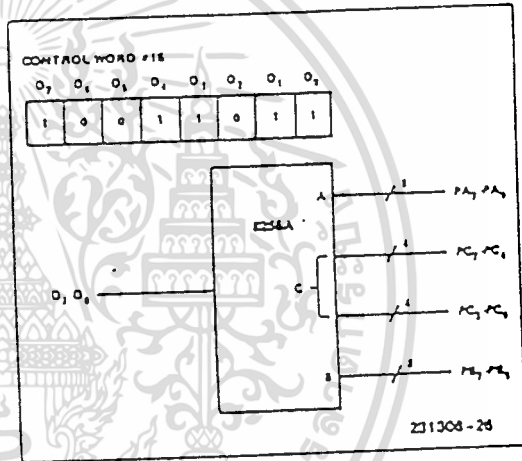
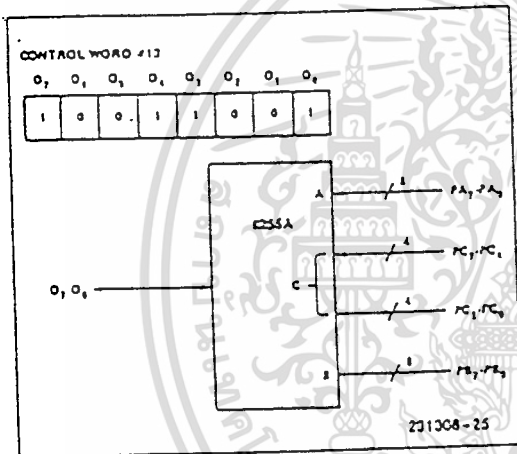
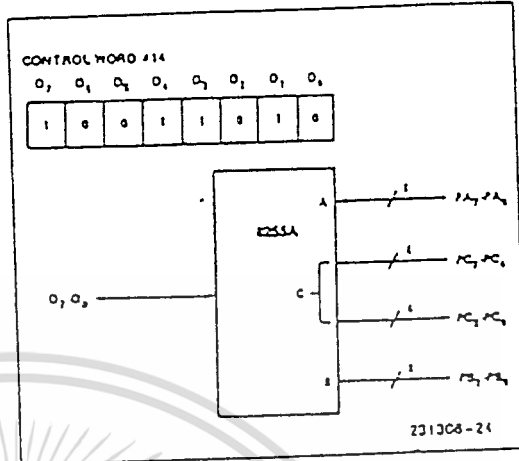
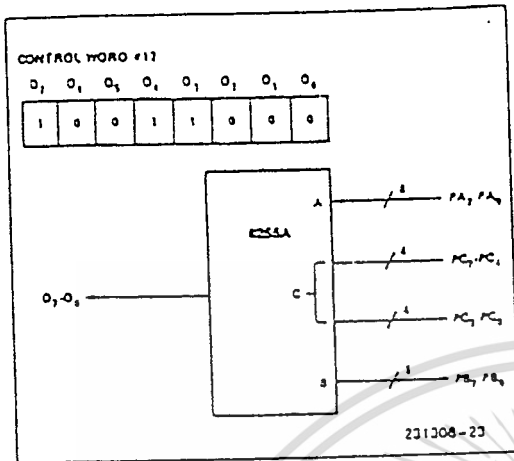
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



8255A/8255A-5



Operating Modes

MODE 1 (Strobed Input/Output). This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or "handshaking" signals. In mode 1, port A and port B use the lines on port C to generate or accept these "handshaking" signals.

Mode 1 Basic Functional Definitions:

- Two Groups (Group A and Group B)
- Each group contains one 8-bit data port and one 4-bit control/data port.
- The 8-bit data port can be either input or output. Both inputs and outputs are latched.
- The 4-bit port is used for control and status of the 8-bit data port.

Input Control Signal Definition

STB (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has been loaded into the input latch; in essence, an acknowledgement. IBF is set by STB input being low and is reset by the rising edge of the RD input.

INTR (Interrupt Request)

A "high" on this output can be used to interrupt the CPU when an input device is requesting service. INTR is set by the STB is a "one", IBF is a "one" and INTE is a "one". It is reset by the falling edge of RD. This procedure allows an input device to request service from the CPU by simply strobing its data into the port.



INTE A

Controlled by bit set/reset of PC<sub>4</sub>.

INTE B

Controlled by bit set/reset of PC<sub>2</sub>.

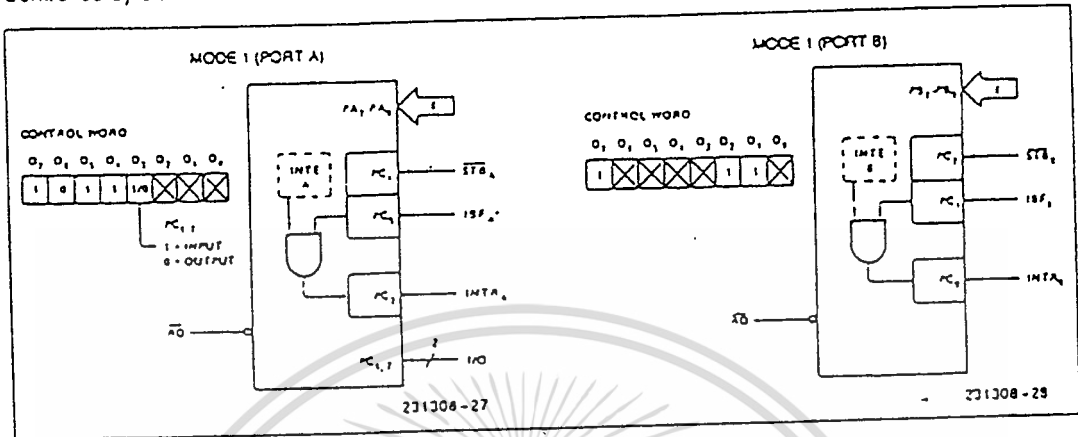


Figure 8. MODE 1 Input

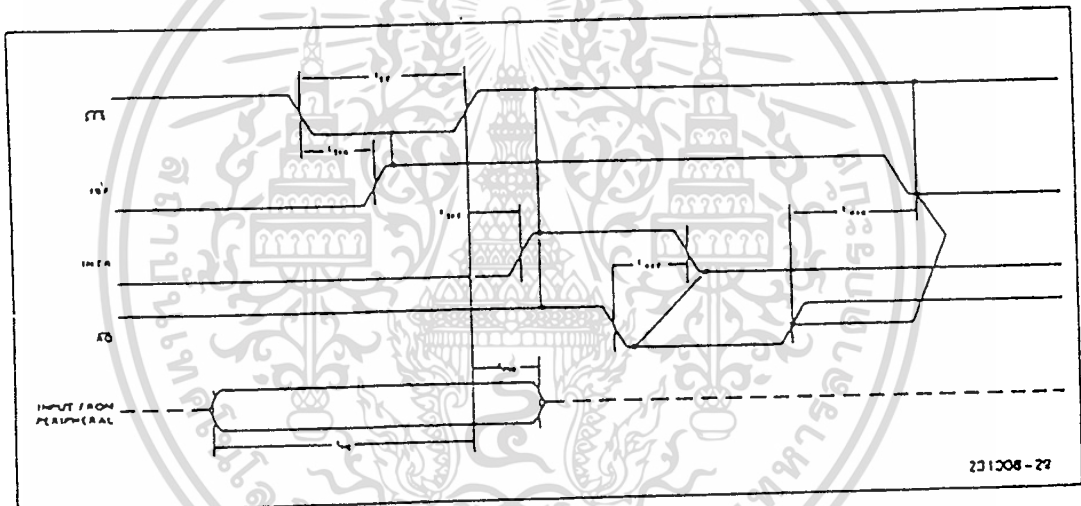


Figure 9. MODE 1 (Strobed Input)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Output Control Signal Definition

**OB $\bar{F}$**  (Output Buffer Full F/F). The **OB $\bar{F}$**  output will go "low" to indicate that the CPU has written data out to the specified port. The **OB $\bar{F}$**  F/F will be set by the rising edge of the **WR** input and reset by **ACK** input being low.

**ACK** (Acknowledge Input). A "low" on this input informs the 8255A that the data from port A or port B has been accepted. In essence, a response from the peripheral device indicating that it has received the data output by the CPU.

**INTR** (Interrupt Request). A "high" on this output can be used to interrupt the CPU when an output

device has accepted data transmitted by the CPU. **INTR** is set when **ACK** is a "one", **OB $\bar{F}$**  is a "one", and **INTE** is a "one". It is reset by the falling edge of **WR**.

**INTE A**

Controlled by bit set/reset of **PC<sub>6</sub>**.

**INTE B**

Controlled by bit set/reset of **PC<sub>2</sub>**.

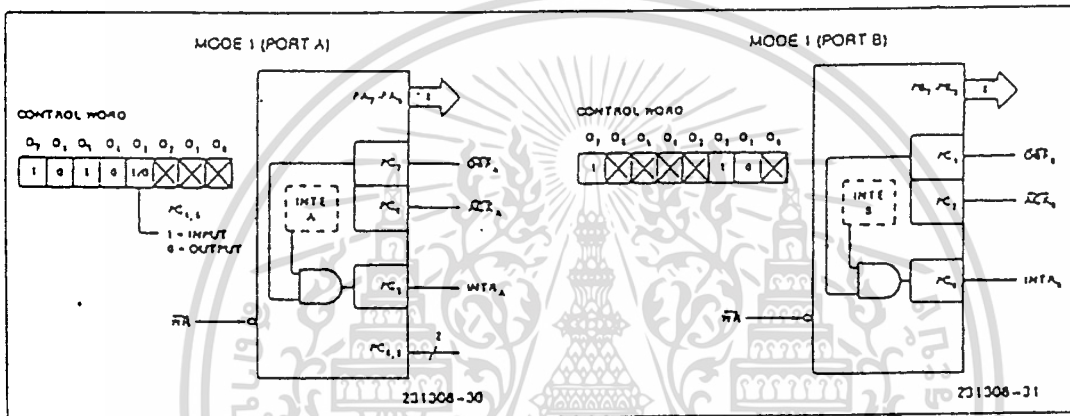


Figure 10. MODE 1 Output

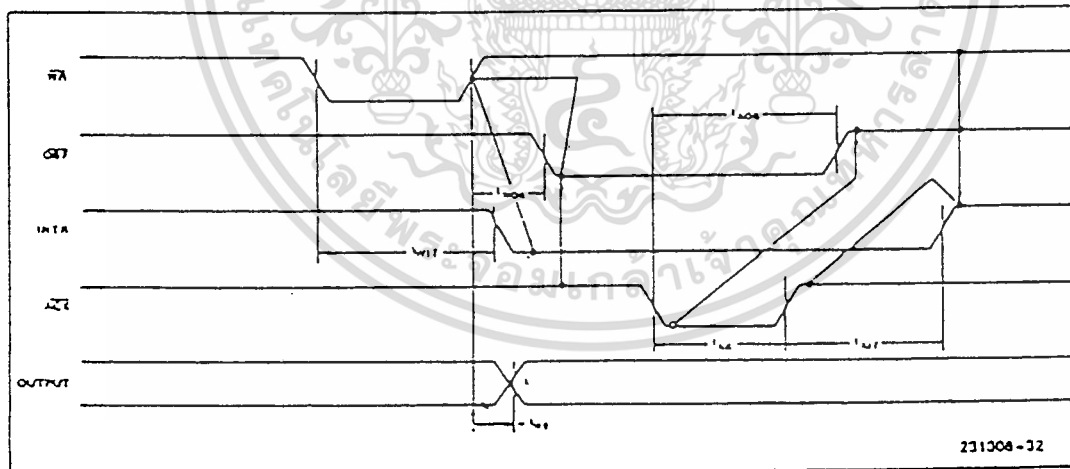


Figure 11. MODE 1 (Strobed Output)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

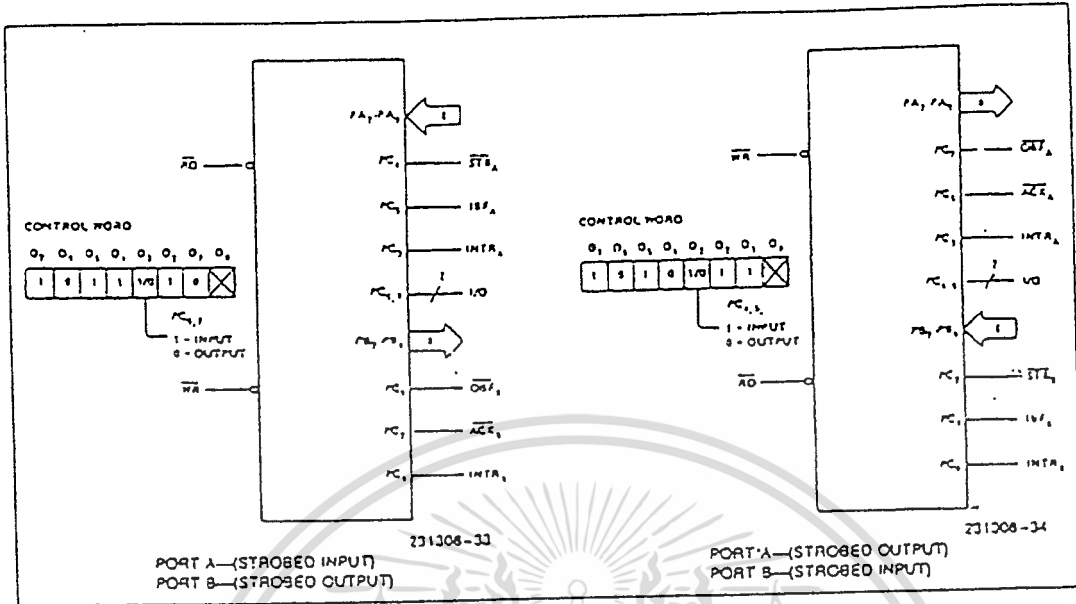


Figure 12. Combinations of MODE 1

Combinations of MODE 1

Port A and Port B can be individually defined as input or output in MODE 1 to support a wide variety of strobed I/O applications.

Bidirectional Bus I/O Control Signal Definition

INTR (Interrupt Request). A high on this output can be used to interrupt the CPU for both input or output operations.

Operating Modes

MODE 2 (Strobed Bidirectional Bus I/O). This functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bidirectional bus I/O). "Handshaking" signals are provided to maintain proper bus flow discipline in a similar manner to MODE 1. Interrupt generation and enable/disable functions are also available.

Output Operations

OBFF (Output Buffer Full). The OBFF output will go "low" to indicate that the CPU has written data out to port A.

ACK (Acknowledge). A "low" on this input enables the tri-state output buffer of port A to send out the data. Otherwise, the output buffer will be in the high impedance state.

MODE 2 Basic Functional Definitions:

- Used in Group A only.
- One 8-bit, bi-directional bus Port (Port A) and a 5-bit control Port (Port C).
- Both inputs and outputs are latched.
- The 5-bit control port, (Port C) is used for control and status for the 8-bit, bi-directional bus port (Port A).

INTE 1 (The INTE Flip-Flop Associated with OBFF). Controlled by bit set/reset of PC6.

Input Operations

STB (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F). A "high" on this output indicates that data has been loaded into the input latch.

INTE 2 (The INTE Flip-Flop Associated with IBF). Controlled by bit set/reset of PC<sub>4</sub>.

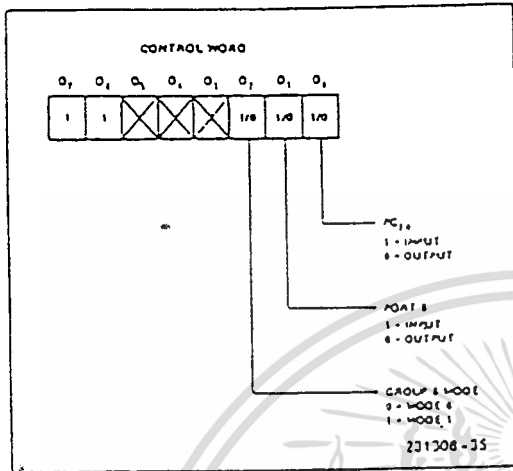


Figure 13. MODE Control Word

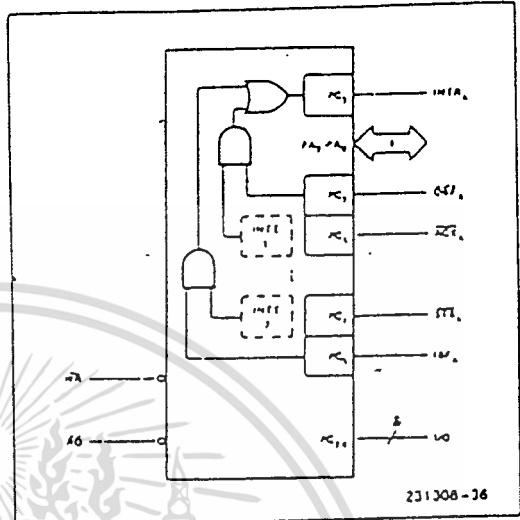


Figure 14. MODE 2

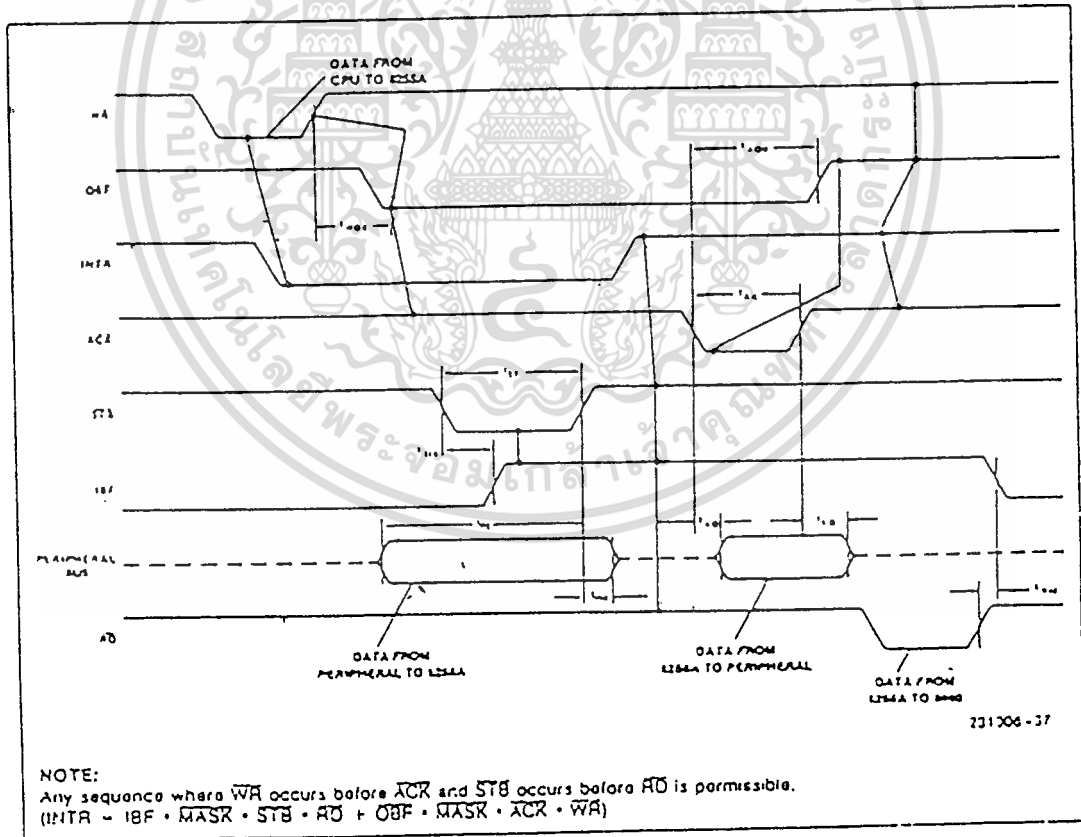


Figure 15. MODE 2 (Bidirectional)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

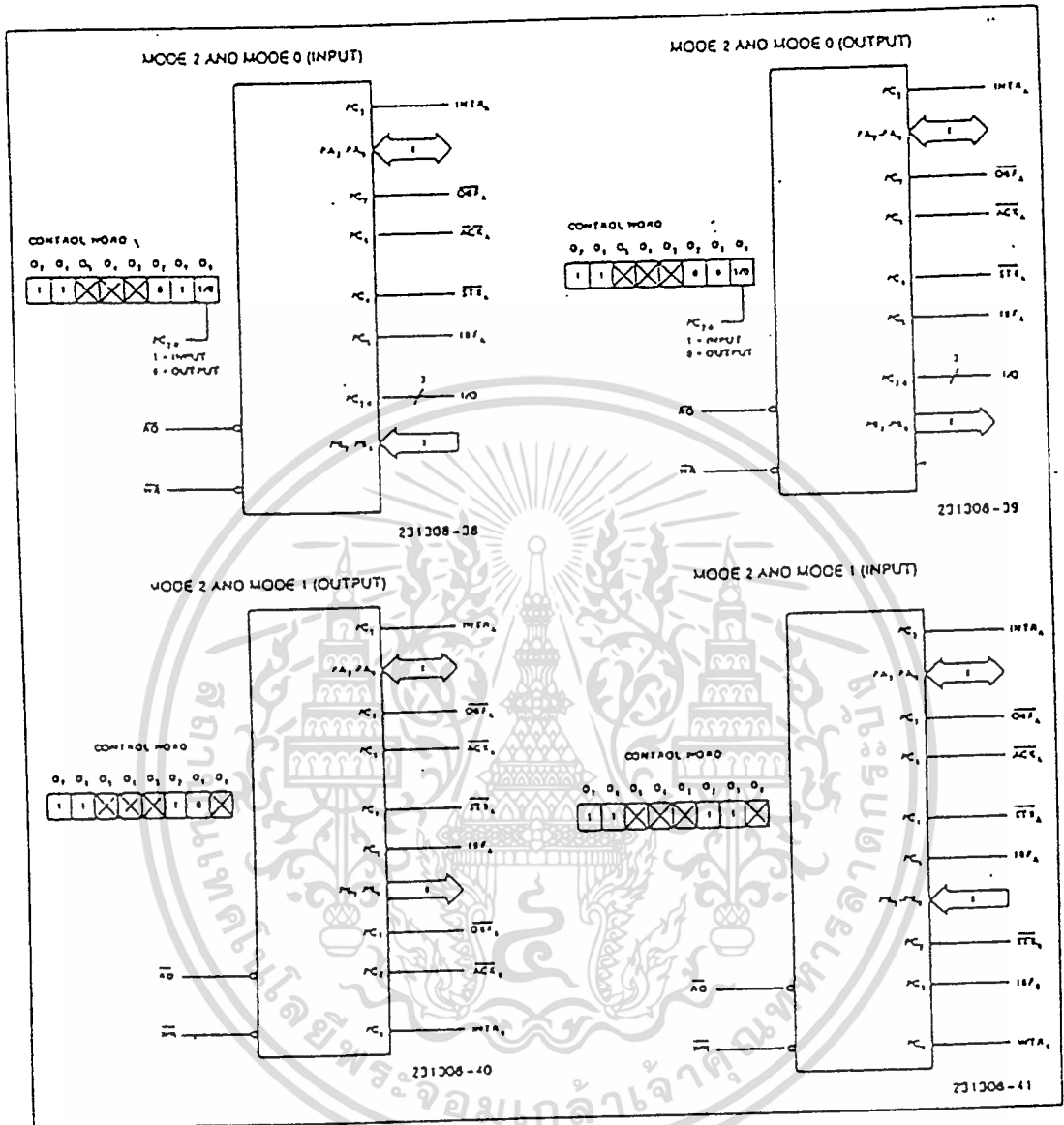


Figure 16. MODE 0 Combinations

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Mode Definition Summary

	MODE 0		MODE 1		MODE 2
	IN	OUT	IN	OUT	GROUP A ONLY
PA <sub>0</sub>	IN	OUT	IN	OUT	↔
PA <sub>1</sub>	IN	OUT	IN	OUT	↔
PA <sub>2</sub>	IN	OUT	IN	OUT	↔
PA <sub>3</sub>	IN	OUT	IN	OUT	↔
PA <sub>4</sub>	IN	OUT	IN	OUT	↔
PA <sub>5</sub>	IN	OUT	IN	OUT	↔
PA <sub>6</sub>	IN	OUT	IN	OUT	↔
PA <sub>7</sub>	IN	OUT	IN	OUT	↔
PB <sub>0</sub>	IN	OUT	IN	OUT	—
PB <sub>1</sub>	IN	OUT	IN	OUT	—
PB <sub>2</sub>	IN	OUT	IN	OUT	—
PB <sub>3</sub>	IN	OUT	IN	OUT	—
PB <sub>4</sub>	IN	OUT	IN	OUT	—
PB <sub>5</sub>	IN	OUT	IN	OUT	—
PB <sub>6</sub>	IN	OUT	IN	OUT	—
PB <sub>7</sub>	IN	OUT	IN	OUT	—
PC <sub>0</sub>	IN	OUT	INTR <sub>B</sub>	INTR <sub>B</sub>	I/O
PC <sub>1</sub>	IN	OUT	IBF <sub>B</sub>	ÖBF <sub>B</sub>	I/O
PC <sub>2</sub>	IN	OUT	STB <sub>B</sub>	ACK <sub>B</sub>	I/O
PC <sub>3</sub>	IN	OUT	INTR <sub>A</sub>	INTR <sub>A</sub>	INTR <sub>A</sub>
PC <sub>4</sub>	IN	OUT	STB <sub>A</sub>	I/O	STB <sub>A</sub>
PC <sub>5</sub>	IN	OUT	IBF <sub>A</sub>	I/O	IBF <sub>A</sub>
PC <sub>6</sub>	IN	OUT	I/O	ACK <sub>A</sub>	ACK <sub>A</sub>
PC <sub>7</sub>	IN	OUT	I/O	ÖBF <sub>A</sub>	ÖBF <sub>A</sub>

MODE 0  
OR MODE 1  
ONLY

## Special Mode Combination Considerations

There are several combinations of modes when not all of the bits in Port C are used for control or status. The remaining bits can be used as follows:

## If Programmed as Inputs—

All input lines can be accessed during a normal Port C read.

## If Programmed as Outputs—

Bits in C upper (PC<sub>7</sub>–PC<sub>4</sub>) must be individually accessed using the bit set/reset function.

Bits in C lower (PC<sub>3</sub>–PC<sub>0</sub>) can be accessed using the bit set/reset function or accessed as a threesome by writing into Port C.

## Source Current Capability on Port B and Port C

Any set of eight output buffers, selected randomly from Ports B and C can source 1 mA at 1.5 volts.

This feature allows the 8255 to directly drive Darlington type drivers and high-voltage displays that require such source current.

## Reading Port C Status

In Mode 0, Port C transfers data to or from the peripheral device. When the 8255 is programmed to function in Modes 1 or 2, Port C generates or accepts "hand-shaking" signals with the peripheral device. Reading the contents of Port C allows the programmer to test or verify the "status" of each peripheral device and change the program flow accordingly.

There is no special instruction to read the status information from Port C. A normal read operation of Port C is executed to perform this function.

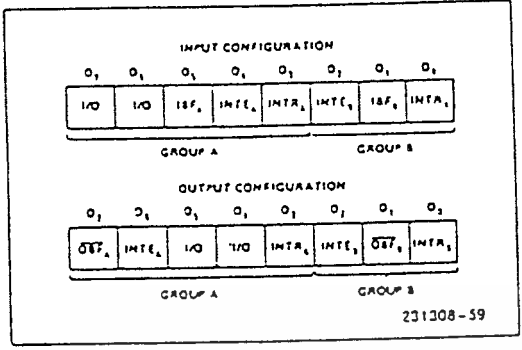


Figure 17. MODE 1 Status Word Format

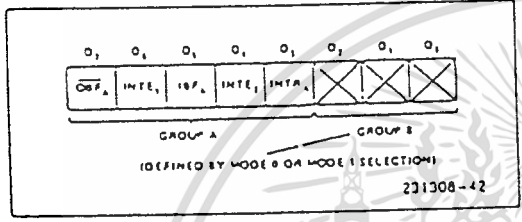


Figure 18. MODE 2 Status Word Format

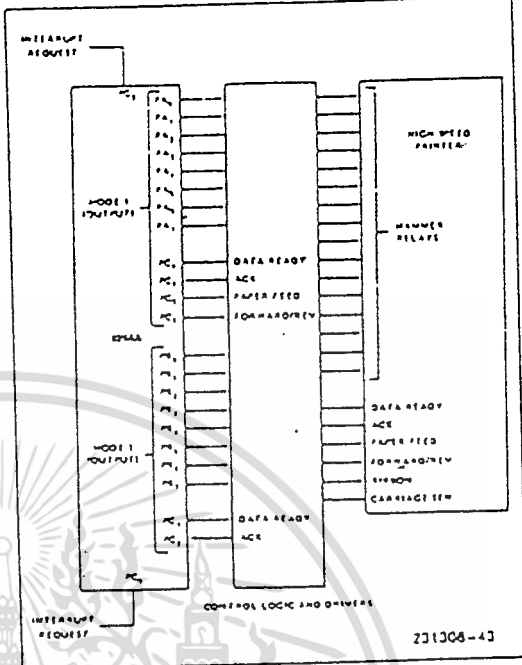


Figure 19. Printer Interface

### APPLICATIONS OF THE 8255A

The 8255A is a very powerful tool for interfacing peripheral equipment to the microcomputer system. It represents the optimum use of available pins and is flexible enough to interface almost any I/O device without the need for additional external logic.

Each peripheral device in a microcomputer system usually has a "service routine" associated with it. The routine manages the software interface between the device and the CPU. The functional definition of the 8255A is programmed by the I/O service routine and becomes an extension of the system software. By examining the I/O devices interface characteristics for both data transfer and timing, and matching this information to the examples and tables in the detailed operational description, a control word can easily be developed to initialize the 8255A to exactly "fit" the application. Figures 19 through 25 represent a few examples of typical applications of the 8255A.

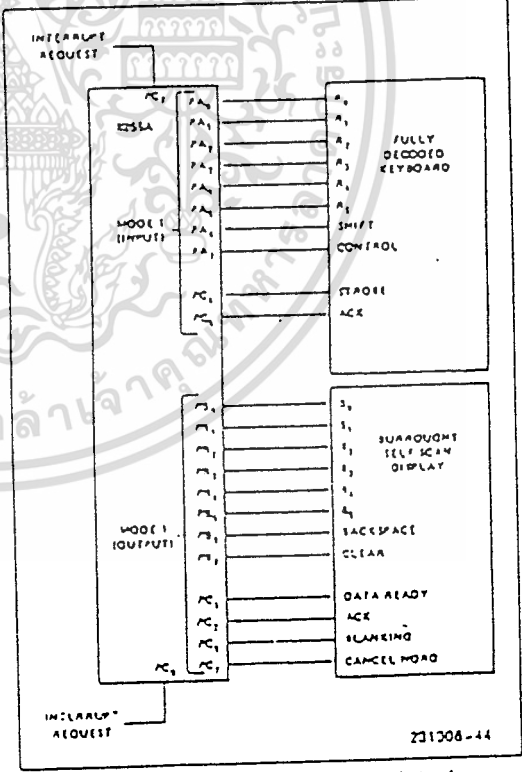


Figure 20. Keyboard and Display Interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

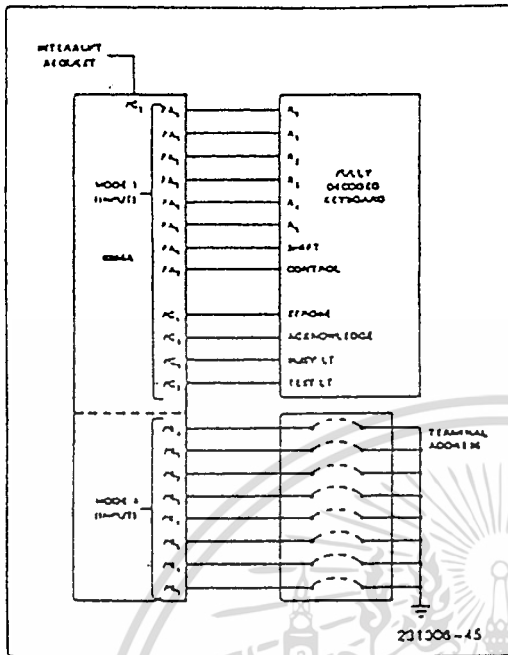


Figure 21. Keyboard and Terminal Address Interface

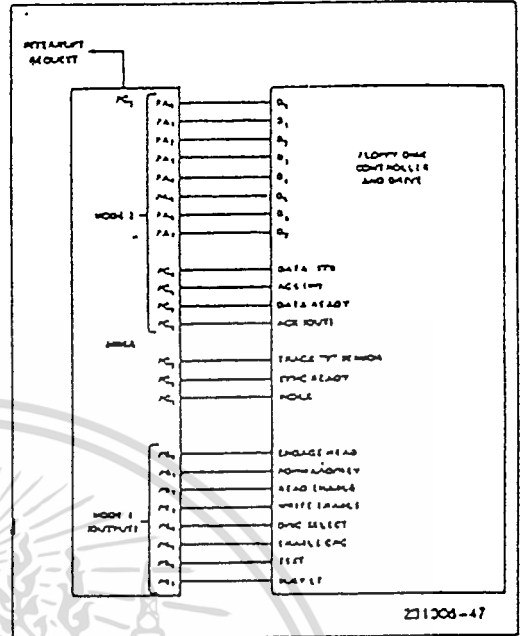


Figure 23. Basic Floppy Disk Interface

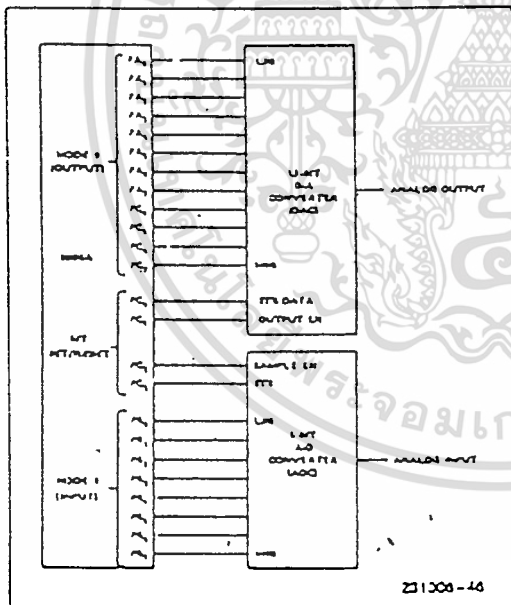


Figure 22. Digital to Analog, Analog to Digital

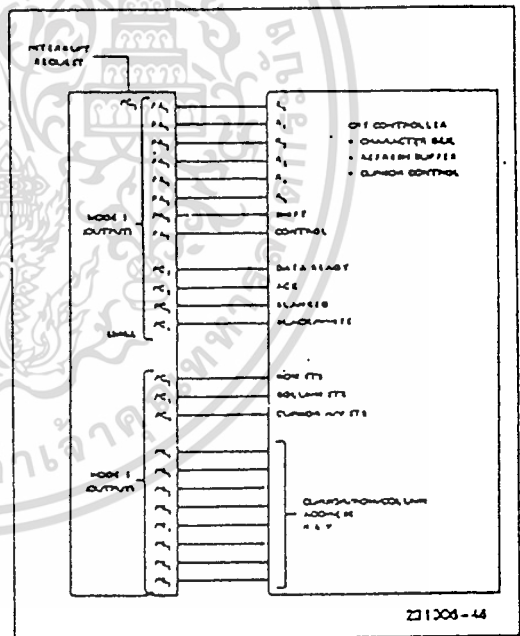


Figure 24. Basic CRT Controller Interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



# CMOS Dual 8-Bit Buffered Multiplying DAC

AD7528

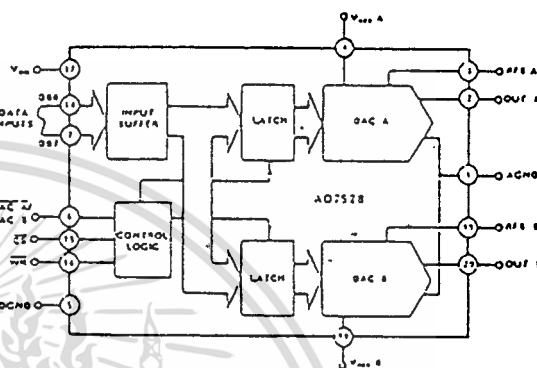
**FEATURES**

- On-Chip Latches for Both DACs
- +5V to +15V Operation
- DACs Matched to 1%
- Four Quadrant Multiplication
- TTL/CMOS Compatible
- Latch Free (Protection Schottkys not Required)

**APPLICATIONS**

- Digital Control of:
  - Gain/Attenuation
  - Filter Parameters
  - Stereo Audio Circuits
  - X-Y Graphics

AD7528 FUNCTIONAL BLOCK DIAGRAM



**GENERAL DESCRIPTION**

The AD7528 is a monolithic dual 3-bit digital-to-analog converter featuring excellent DAC-to-DAC matching. It is available in skinny 0.3" wide 20-pin DIPs and in 20-terminal surface mount packages.

Separate on-chip latches are provided for each DAC to allow easy microprocessor interface.

Data is transferred into either of the two DAC data latches via a common 3-bit TTL/CMOS compatible input port. Control input DAC A/DAC B determines which DAC is to be loaded. The AD7528's load cycle is similar to the write cycle of a random access memory and the device is bus compatible with most 8-bit microprocessors, including 6800, 8030, 8085, Z80.

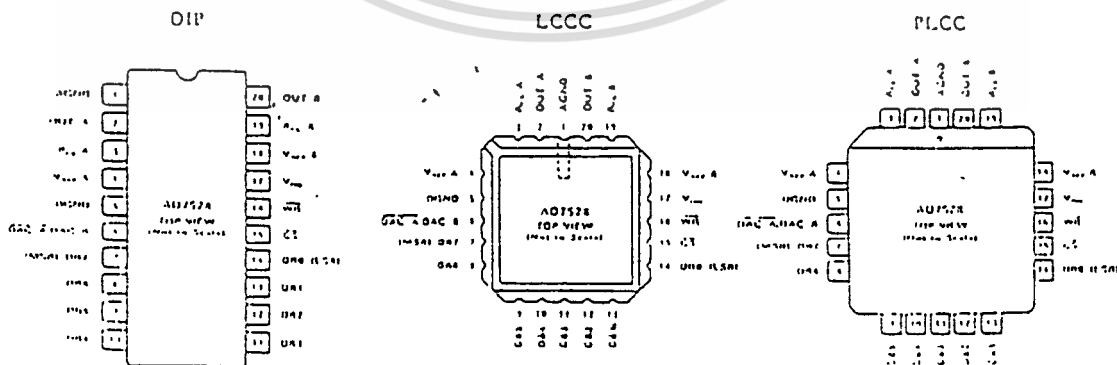
The device operates from a +5V to +15V power supply, dissipating only 20mW of power.

Both DACs offer excellent four quadrant multiplication characteristics with a separate reference input and feedback resistor for each DAC.

**PRODUCT HIGHLIGHTS**

1. DAC to DAC matching: since both of the AD7528 DACs are fabricated at the same time on the same chip, precise matching and tracking between DAC A and DAC B is inherent. The AD7528's matched CMOS DACs make a whole new range of applications circuits possible, particularly in the audio, graphics and process control areas.
2. Small package size: combining the inputs to the on-chip DAC latches into a common data bus and adding a DAC A/DAC B select line has allowed the AD7528 to be packaged in either a small 20-pin 0.3" wide DIP or in 20-terminal surface mount packages.

**PIN CONFIGURATIONS**



**DIGITAL-TO-ANALOG CONVERTERS**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SPECIFICATIONS ( $V_{REF A} = V_{REF B} = +10V$ ; $OUT A = OUT B = 0V$ unless otherwise specified)

Parameter	Version <sup>1</sup>	$V_{DD} = +5V$		$V_{DD} = +15V$		Units	Test Conditions/Comments
		$T_a = +15^\circ C$	$T_{min}, T_{max}$	$T_a = +15^\circ C$	$T_{min}, T_{max}$		
<b>STATIC PERFORMANCE<sup>1</sup></b>							
Resolution	All	1	1	1	1	Bits	This is an Endpoints Linearity Specification
Relative Accuracy	J, A, S	$\pm 1$	$\pm 1$	$\pm 1$	$\pm 1$	LSB max	
	K, B, T	$\pm 1/2$	$\pm 1/2$	$\pm 1/2$	$\pm 1/2$	LSB max	
	L, C, U	$\pm 1/2$	$\pm 1/2$	$\pm 1/2$	$\pm 1/2$	LSB max	
Differential Nonlinearity	All	$\pm 1$	$\pm 1$	$\pm 1$	$\pm 1$	LSB max	All Grades Guaranteed Monotonic Over Full Operating Temperature Range
Gain Error	J, A, S	$\pm 4$	$\pm 4$	$\pm 4$	$\pm 3$	LSB max	
	K, B, T	$\pm 2$	$\pm 4$	$\pm 2$	$\pm 1$	LSB max	Measured Using Internal RFB A and RFB B, Both DAC Latches Loaded with 11111111. Gain Error is Adjustable Using Circuits of Figures 1 and 2...
	L, C, U	$\pm 1$	$\pm 3$	$\pm 1$	$\pm 1$	LSB max	
Gain Temperature Coefficient <sup>1</sup>	All	$\pm 0.007$	$\pm 0.007$	$\pm 0.0035$	$\pm 0.0035$	%/°C max	DAC Latches Loaded with 00000000
Normal Leakage Current	All	$\pm 50$	$\pm 400$	$\pm 50$	$\pm 200$	nA max	
OUT A (Pin 2)	All	$\pm 50$	$\pm 400$	$\pm 50$	$\pm 200$	nA max	Input Resistance TC = $\pm 100ppm/^\circ C$ , Typical Input Resistance is 11k $\Omega$
OUT B (Pin 10)	All	$\pm 50$	$\pm 400$	$\pm 50$	$\pm 200$	nA max	
Input Resistance ( $V_{REF A}, V_{REF B}$ )	All	1	1	1	1	k $\Omega$ min	Input Resistance TC = $\pm 100ppm/^\circ C$ , Typical Input Resistance is 11k $\Omega$
	All	15	15	15	15	k $\Omega$ max	
$V_{REF A}, V_{REF B}$ Input Resistance Match	All	$\pm 1$	$\pm 1$	$\pm 1$	$\pm 1$	% max	
<b>DIGITAL INPUTS<sup>1</sup></b>							
Input High Voltage	All	2.4	2.4	11.5	11.5	V min	$V_{DD} = 0$ or $V_{DD}$
Input Low Voltage	All	0.8	0.8	1.5	1.5	V max	
Input Current	All	$\pm 1$	$\pm 10$	$\pm 1$	$\pm 10$	$\mu A$ max	$V_{DD} = 0$ or $V_{DD}$
Input Capacitance	All	10	10	10	10	pF max	
DR, DR2	All	15	15	15	15	pF max	
DR, CS, DAC/DAC2	All	15	15	15	15	pF max	
<b>SWITCHING CHARACTERISTICS<sup>1</sup></b>							
Chip Select to Tri-state Set Up Time	All	200	130	60	60	ns min	See Timing Diagram
Chip Select to Tri-state Hold Time	All	20	30	10	15	ns min	
DAC Select to Tri-state Set Up Time	All	200	130	60	60	ns min	See Timing Diagram
DAC Select to Tri-state Hold Time	All	20	30	10	15	ns min	
DRs Valid to Tri-state Set Up Time	All	110	130	30	40	ns min	See Timing Diagram
DRs Valid to Tri-state Hold Time	All	0	0	0	0	ns min	
Tri-state Pulse Width	All	120	200	60	10	ns min	
<b>POWER SUPPLY</b>							
$I_{DD}$	All	1	1	2	2	mA max	All Digital Inputs $V_{DD}$ or $V_{DD}$
	All	100	100	100	100	$\mu A$ max	

## AC PERFORMANCE CHARACTERISTICS<sup>5</sup> (Measured Using Recommended P.C. Board Layout and AD644 as Output Amplifiers)

Parameter	Version <sup>1</sup>	$V_{DD} = +5V$		$V_{DD} = +15V$		Units	Test Conditions/Comments
		$T_a = +15^\circ C$	$T_{min}, T_{max}$	$T_a = +15^\circ C$	$T_{min}, T_{max}$		
DC SUPPLY REJECTION (25dB $V_{DD}$ )	All	0.01	0.04	0.01	0.02	% per % max	$\Delta V_{DD} = \pm 1\%$
CURRENT SETTLING TIME <sup>1</sup>	All	130	400	110	200	ns max	To 1/10 LSB, One An/Dac Channel = 10k $\Omega$ , $\overline{CS} = 0V$ , DRs/DAC2 = 0V to $V_{DD}$ or $V_{DD}$ 0V
PROPAGATION DELAY (From Digital Inputs to 1/10 Final Analog Output Current)	All	120	120	10	100	ns max	$V_{REF A} = V_{REF B} = +10V$ OUT A, OUT B Load = 100k $\Omega$ , $C_{LOAD} = 15pF$ $\overline{CS}, \overline{CS} = 0V$ , DRs/DAC2 = 0V to $V_{DD}$ or $V_{DD}$ 0V
DIGITAL CHANNEL GLITCH IMPULSE	All	160	-	140	-	nV ns typ	For Cable Transition 0.200000 to 11111111
<b>OUTPUT CAPACITANCE</b>							
$C_{OUT A}$	All	10	50	10	50	pF max	DAC Latches Loaded with 00000000
$C_{OUT B}$	All	10	50	10	50	pF max	
$C_{REF A}$	All	120	120	120	120	pF max	DAC Latches Loaded with 11111111
$C_{REF B}$	All	120	120	120	120	pF max	
<b>AC FEED THROUGH</b>							
$V_{REF A}$ to 10k $\Omega$	All	-75	-45	-70	-45	dB max	$V_{REF A}, V_{REF B} = 20V$ pp sine wave At 100kHz
$V_{REF B}$ to 10k $\Omega$	All	-70	-45	-70	-45	dB max	
<b>CHANNEL TO CHANNEL ISOLATION</b>							
$V_{REF A}$ to 10k $\Omega$	All	-77	-	-77	-	dB typ	Both DAC Latches Loaded with 11111111 $V_{REF A} = 20V$ pp sine wave 100kHz $V_{REF B} = 0V$ $V_{REF A} = 20V$ pp sine wave 100kHz $V_{REF B} = 0V$
$V_{REF B}$ to 10k $\Omega$	All	-77	-	-77	-	dB typ	
<b>DIGITAL INPUTS FAULT</b>							
All	All	40	-	40	-	nV ns typ	Measured with Data Transition 00000000 to 11111111
<b>HARMONIC DISTORTION</b>							
All	All	-15	-	-15	-	dB typ	$V_{DD} = 0V$ and 0V

<sup>1</sup>Temperature Range is: J, L, Systems,  $\pm 0^\circ C$  to  $+110^\circ C$ ; S, R, Systems,  $\pm 0^\circ C$  to  $+110^\circ C$ ; K, T, Systems,  $\pm 15^\circ C$  to  $+110^\circ C$ ; L, C, U, Systems,  $\pm 15^\circ C$  to  $+110^\circ C$ .  
<sup>2</sup>Typical values are shown in parentheses (in parentheses) in parentheses.  
<sup>3</sup>Typical values are shown in parentheses (in parentheses) in parentheses.  
<sup>4</sup>These characteristics are for single channel only and are not guaranteed for two channel operation.

### 2-112 DIGITAL-TO-ANALOG CONVERTERS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AD7528

**INTERFACE LOGIC INFORMATION**

**DAC Selection:**

Both DAC latches share a common 3-bit input port. The control input  $\overline{DAC\ A}$  /  $\overline{DAC\ B}$  selects which DAC can accept data from the input port.

**Mode Selection:**

Inputs  $\overline{CS}$  and  $\overline{WR}$  control the operating mode of the selected DAC. See Mode Selection Table below.

**Write Mode:**

When  $\overline{CS}$  and  $\overline{WR}$  are both low the selected DAC is in the write mode. The input data latches of the selected DAC are transparent and its analog output responds to activity on DB0-DB7.

**Hold Mode:**

The selected DAC latch retains the data which was present on DB0-DB7 just prior to  $\overline{CS}$  or  $\overline{WR}$  assuming a high state. Both analog outputs remain at the values corresponding to the data in their respective latches.

DAC A / DAC B	$\overline{CS}$	$\overline{WR}$	DACA	DACB
L	L	L	WRITE	HOLD
H	L	L	HOLD	WRITE
X	H	X	HOLD	HOLD
X	X	H	HOLD	HOLD

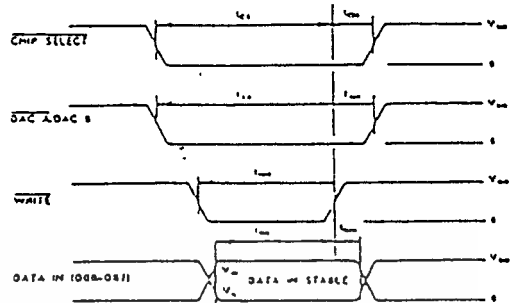
L = Low State H = High State X = Don't Care

*Mode Selection Table*

**CAUTION:**

- ESD sensitive device. The digital control inputs are diode protected; however, permanent damage may occur on unconnected devices subjected to high energy electrostatic fields. Unused devices must be stored in conductive foam or shunts.
- Do not insert this device into powered sockets. Remove power before insertion or removal.

**WRITE CYCLE TIMING DIAGRAM**



**NOTES:**  
 1. ALL INPUT SIGNAL RISE AND FALL TIMES MEASURED FROM 10% TO 90% OF  $V_{DD}$ .  
 $V_{DD} = +5V, t_r = t_f = 10ns$   
 $V_{DD} = +15V, t_r = t_f = 10ns$   
 2. TIMING MEASUREMENT REFERENCE LEVEL IS  $V_{DD}/2$

**ABSOLUTE MAXIMUM RATINGS**

( $T_A = +25^\circ C$  unless otherwise noted)

- $V_{DD}$  to AGND . . . . . 0V, +17V
- $V_{DD}$  to DGND . . . . . 0V, +17V
- AGND to DGND . . . . .  $V_{DD} + 0.3V$
- DGND to AGND . . . . .  $V_{DD} + 0.3V$
- Digital Input Voltage to DGND . . . . . -0.3V,  $V_{DD} + 0.3V$
- $V_{PIN2}, V_{PIN3}$  to AGND . . . . . -0.3V,  $V_{DD} + 0.3V$
- $V_{REF\ A}, V_{REF\ B}$  to AGND . . . . . = 25V
- $V_{REF\ A}, V_{REF\ B}$  to AGND . . . . . = 25V
- Power Dissipation (Any Package) to  $+75^\circ C$  . . . . . 450mW
- Derates above  $+75^\circ C$  by . . . . . 6mW/ $^\circ C$
- Operating Temperature Range
- Commercial (J, K, L) Grades . . . . .  $-40^\circ C$  to  $+85^\circ C$
- Industrial (A, B, C) Grades . . . . .  $-40^\circ C$  to  $+85^\circ C$
- Extended (S, T, U) Grades . . . . .  $-55^\circ C$  to  $+125^\circ C$
- Storage Temperature . . . . .  $-65^\circ C$  to  $+150^\circ C$
- Lead Temperature (Soldering, 10 secs.) . . . . .  $+300^\circ C$

**ORDERING INFORMATION<sup>1</sup>**

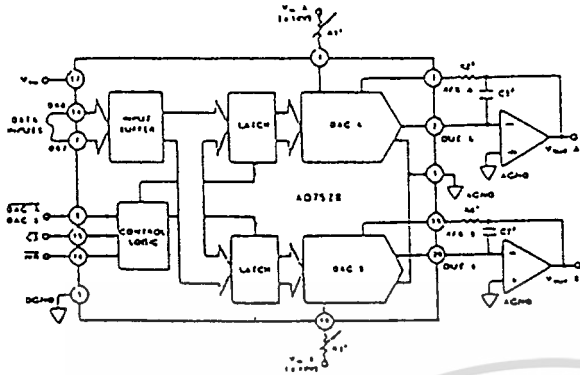
Relative Accuracy	Gain Error $T_A = +25^\circ C$	Temperature Range and Package Options <sup>1, 2</sup>		
		$-40^\circ C$ to $+85^\circ C$	$-40^\circ C$ to $+85^\circ C$	$-55^\circ C$ to $+125^\circ C$
$\pm 1LSB$	$\pm 4LSB$	Plastic DIP (N-20)	Hermetic (Q-20)	Hermetic (Q-20)
$\pm 1/2LSB$	$\pm 2LSB$	AD7528JN	AD7528AQ	AD7528SQ
$\pm 1/2LSB$	$\pm 1LSB$	AD7528KN	AD7528HQ	AD7528TQ
$\pm 1/2LSB$	$\pm 1LSB$	AD7528LN	AD7528CQ	AD7528UQ
$\pm 1LSB$	$\pm 4LSB$	Leadless Ceramic (P-20A)		Leadless Ceramic (E-20A)
$\pm 1/2LSB$	$\pm 2LSB$	AD7528JP		AD7528SE
$\pm 1/2LSB$	$\pm 1LSB$	AD7528KP		AD7528TE
$\pm 1/2LSB$	$\pm 1LSB$	AD7528LP		AD7528UE

**NOTES**

- To order MIL-STD-883, Class B processed parts, add 883B to part number. Contact your local sales office for military data sheet. For U.S. Standard Military Drawing (SMD), see DASC drawing #3762-47701.
- See Section 14 for package outline information.
- Also available in SOIC package (AD7528KR, AD7528LU).
- PLCC: Plastic Leaded Chip Carrier.
- LCCC: Leadless Ceramic Chip Carrier.

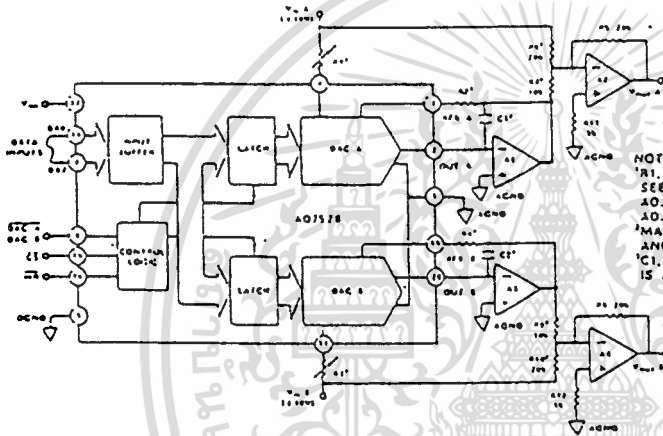
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Applying the AD7528



NOTES:  
 \*R1, R2 AND R3, R4 USED ONLY IF GAIN ADJUSTMENT IS REQUIRED. SEE TABLE 3 FOR RECOMMENDED VALUES.  
 \*C1, C2 PHASE COMPENSATION (100pF-150pF) IS REQUIRED WHEN USING HIGH SPEED AMPLIFIERS TO PREVENT RINGING OR OSCILLATION.

Figure 1. Dual DAC Unipolar Binary Operation (2 Quadrant Multiplication). See Table I.



NOTES:  
 \*R1, R2 AND R3, R4 USED ONLY IF GAIN ADJUSTMENT IS REQUIRED. SEE TABLE 3 FOR RECOMMENDED VALUES.  
 \*ADJUST R1 FOR Vout A = 0V WITH CODE 10000000 IN DAC A LATCH. ADJUST R3 FOR Vout B = 0V WITH CODE 10000000 IN DAC B LATCH. MATCHING AND TRACKING IS ESSENTIAL FOR RESISTOR PAIRS R1, R3 AND R2, R4.  
 \*C1, C2 PHASE COMPENSATION (100pF-150pF) MAY BE REQUIRED IF AMPLIFIER IS A HIGH-SPEED AMPLIFIER.

Figure 2. Dual DAC Bipolar Operation (4 Quadrant Multiplication). See Table II.

DAC Latch Contents	Analog Output (D.A.C.A or D.A.C.B)
MSB	LSB
11111111	$-V_{IN} \left( \frac{255}{256} \right)$
10000001	$-V_{IN} \left( \frac{129}{256} \right)$
10000000	$-V_{IN} \left( \frac{128}{256} \right) = -\frac{V_{IN}}{2}$
01111111	$-V_{IN} \left( \frac{127}{256} \right)$
01000001	$-V_{IN} \left( \frac{1}{256} \right)$
00000000	$-V_{IN} \left( \frac{0}{256} \right) = 0$

Note:  $V_{OUT} = (2^7 - 1) V_{IN} - \frac{1}{256} (V_{IN})$

Table I. Unipolar Binary Code Table

DAC Latch Contents	Analog Output (D.A.C.A or D.A.C.B)
MSB	LSB
11111111	$+V_{IN} \left( \frac{127}{128} \right)$
10000001	$+V_{IN} \left( \frac{1}{128} \right)$
10000000	0
01111111	$-V_{IN} \left( \frac{1}{128} \right)$
00000001	$-V_{IN} \left( \frac{127}{128} \right)$
00000000	$-V_{IN} \left( \frac{128}{128} \right)$

Note:  $V_{OUT} = (2^7 V_{IN}) - \frac{1}{128} (V_{IN})$

Table II. Bipolar (Offset Binary) Code Table

Trim Resistor	J/MS	K/MT	UCU
R1;R3	1k	500	200
R2;R4	330	150	R2

Table III. Recommended Trim Resistor Values vs. Gain

## 2-2-11 DIGITAL-TO-ANALOG CONVERTERS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



# LM3303/LM3403/LM3503 Quad Operational Amplifiers

## General Description

The LM3303, LM3403, and LM3503 are monolithic quad operational amplifiers consisting of four independent high gain, internally frequency compensated, operational amplifiers designed to operate from a single power supply or dual power supplies over a wide range of voltages. The common mode input range includes the negative supply, thereby eliminating the necessity for external biasing components in many applications.

- Four internally compensated operational amplifiers in a single package
- Wide power supply range single supply of 3.0V to 36V dual supply of  $\pm 1.5V$  to  $\pm 18V$
- Class AB output stage for minimal crossover distortion
- Short circuit protected outputs
- High open loop gain 200k
- LM741 operational amplifier type performance

## Features

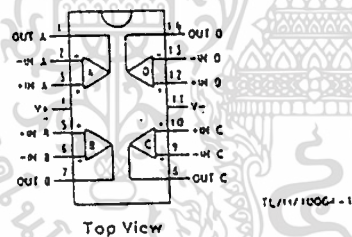
- Input common mode voltage range includes ground or negative supply
- Output voltage can swing to ground or negative supply

## Applications

- Filters
- Voltage controlled oscillators

## Connection Diagram

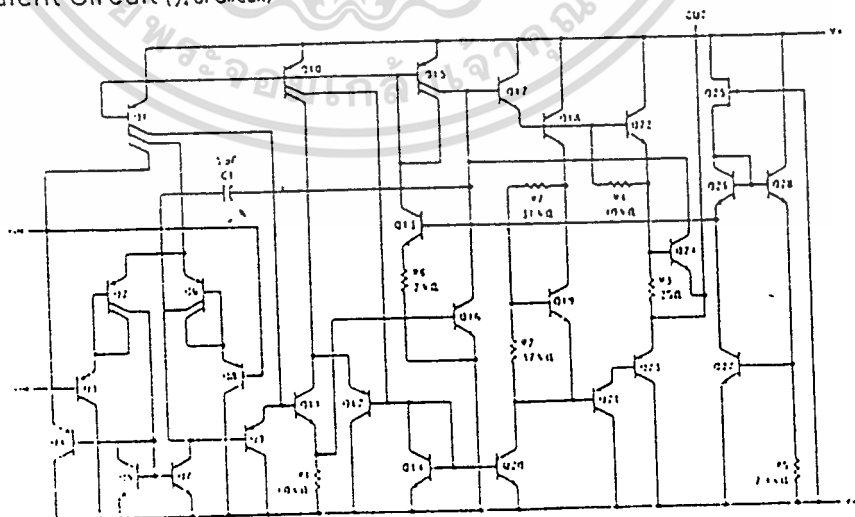
14-Lead DIP and SO-14 Package



## Order Information

Device Code	Package Code	Package Description
LM3303J	J14A	Ceramic DIP
LM3303N	N14A	Molded DIP
LM3403J	J14A	Ceramic DIP
LM3403N	N14A	Molded DIP
LM3403M	M14A	Molded Surface Mount
LM3503J	J14A	Ceramic DIP

## Equivalent Circuit (1/4 of Circuit)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LM3303/LM3403/LM3573

### Absolute Maximum Ratings

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Storage Temperature Range		
Ceramic DIP	-65°C to +175°C	
Molded DIP and SO-14	-65°C to +150°C	
Operating Temperature Range		
Military (LM3503)	-55°C to +125°C	
Industrial (LM3303)	-40°C to +85°C	
Commercial (LM3403)	0°C to +70°C	
Lead Temperature		
Ceramic DIP (Soldering, 60 sec.)	300°C	
Molded DIP and SO-14 (Soldering, 10 sec.)	265°C	

### Internal Power Dissipation (Notes 1, 2)

14L-Ceramic DIP	1.36W
14L-Molded DIP	1.04W
SO-14	0.93W
Supply Voltage between V+ and V-	36V
Differential Input Voltage (Note 3)	±30V
Input Voltage	(V-) - 0.3V to V+
ESD Tolerance	(To Be Determined)

### LM3303 and LM3403

Electrical Characteristics  $T_A = 25^\circ\text{C}$ ,  $V_{CC} = \pm 15\text{V}$ , unless otherwise specified

Symbol	Parameter	Conditions	LM3303			LM3403			Units
			Min	Typ	Max	Min	Typ	Max	
$V_{IO}$	Input Offset Voltage			2.0	8.0		2.0	8.0	mV
$I_{IO}$	Input Offset Current			30	75		30	50	nA
$I_{IQ}$	Input Bias Current			200	500		200	500	nA
$Z_i$	Input Impedance		0.3	1.0		0.3	1.0		M $\Omega$
$I_{CC}$	Supply Current	$V_O = 0\text{V}$ , $R_L = \infty$		2.8	7.0		2.8	7.0	mA
CMR	Common Mode Rejection	$R_S \leq 10\text{ k}\Omega$	70	90		70	50		dB
$V_{IR}$	Input Voltage Range		+12V to V-	+12.5V to V-		+13V to V-	+13.5V to V-		V
PSRR	Power Supply Rejection Ratio			30	150		30	150	$\mu\text{V/V}$
$I_{OS}$	Output Short Circuit Current (Per Amplifier) (Note 4)		±10	±30	±45	±10	±30	±45	mA
$A_{VS}$	Large Signal Voltage Gain	$V_O = \pm 10\text{V}$ , $R_L \geq 2.0\text{ k}\Omega$	20	200		20	200		V/mV
$V_{OP}$	Output Voltage Swing	$R_L = 10\text{ k}\Omega$	±12	12.5		±12	+13.5		V
		$R_L = 2.0\text{ k}\Omega$	±10	12		±10	±13		
TR	Transient Response	Rise Time/ Fall Time	$V_O = 50\text{ mV}$ , $A_V = 1.0$ , $R_L = 10\text{ k}\Omega$		0.3		0.3		$\mu\text{s}$
		Overshoot	$V_O = 50\text{ mV}$ , $A_V = 1.0$ , $R_L = 10\text{ k}\Omega$		5.0		5.0		%
BW	Bandwidth	$V_O = 50\text{ mV}$ , $A_V = 1.0$ , $R_L = 10\text{ k}\Omega$		1.0		1.0		MHz	
SN	Slew Rate	$V_i = -10\text{V}$ to $+10\text{V}$ , $A_V = 1.0$		0.5		0.5		V/ $\mu\text{s}$	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## LM3303 and LM3403 (Continued)

Electrical Characteristics  $T_A = 25^\circ\text{C}$ ,  $V_{CC} = \pm 15\text{V}$ , unless otherwise specifiedThe following specifications apply for  $-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$  for the LM3303, and  $0^\circ\text{C} \leq T_A \leq +70^\circ\text{C}$  for the LM3403

Symbol	Parameter	Conditions	LM3303			LM3403			Units
			Min	Typ	Max	Min	Typ	Max	
$V_{IO}$	Input Offset Voltage				10			10	mV
$\Delta V_{IO}/\Delta T$	Input Offset Voltage Temperature Sensitivity			10			10		$\mu\text{V}/^\circ\text{C}$
$I_{IO}$	Input Offset Current				250			200	nA
$\Delta I_{IO}/\Delta T$	Input Offset Current Temperature Sensitivity			50			50		$\text{pA}/^\circ\text{C}$
$I_{IB}$	Input Bias Current				1000			800	nA
$A_{VS}$	Large Signal Voltage Gain	$V_O = \pm 10\text{V}$ , $R_L \geq 2.0\text{ k}\Omega$	15			15			V/mV
$V_{OP}$	Output Voltage Swing	$R_L = 2.0\text{ k}\Omega$	$\geq 10$			$\geq 10$			V

## LM3303 and LM3403

Electrical Characteristics  $T_A = 25^\circ\text{C}$ ,  $V_+ = 5.0\text{V}$ ,  $V_- = \text{GND}$ , unless otherwise specified

Symbol	Parameter	Conditions	LM3303			LM3403			Units
			Min	Typ	Max	Min	Typ	Max	
$V_{IO}$	Input Offset Voltage				8.0		2.0	8.0	mV
$I_{IO}$	Input Offset Current				75		30	50	nA
$I_{IB}$	Input Bias Current				500		200	500	nA
$I_{CC}$	Supply Current			2.5	7.0		2.5	7.0	mA
PSRR	Power Supply Rejection Ratio				150			150	$\mu\text{V}/\text{V}$
$A_{VS}$	Large Signal Voltage Gain	$R_L \geq 2.0\text{ k}\Omega$	20	200		20	200		V/mV
$V_{OP}$	Output Voltage Swing (Note 5)	$R_L = 10\text{ k}\Omega$ $5.0\text{V} \leq V_+ \leq 30\text{V}$ , $R_L = 10\text{ k}\Omega$	3.3			3.3			V
CS	Channel Separation	1.0 Hz $\leq f \leq 20\text{ kHz}$ (input referenced)		-120			-120		dB

## LM3503

Electrical Characteristics  $T_A = 25^\circ\text{C}$ ,  $V_{CC} = \pm 15\text{V}$ , unless otherwise specified

Symbol	Parameter	Conditions	LM3503			Units
			Min	Typ	Max	
$V_{IO}$	Input Offset Voltage			2.0	5.0	mV
$I_{IO}$	Input Offset Current			30	50	nA
$I_{IB}$	Input Bias Current			200	500	nA
$Z_i$	Input Impedance		0.3	1.0		M $\Omega$
$I_{SC}$	Supply Current	$V_O = 0\text{V}$ , $R_L$		2.0	4.0	mA
CMR	Common Mode Rejection	$R_L = 10\text{ k}\Omega$	70	90		dB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LM3303/LM3403/LM3503

LM3503

Electrical Characteristics  $T_A = 25^\circ\text{C}$ ,  $V_{CC} = \pm 15\text{V}$ , unless otherwise specified (Continued)

Symbol	Parameter	Conditions	LM3503			Units
			Min	Typ	Max	
$V_{IR}$	Input Voltage Range		+13V to $V^-$	+13.5V to $V^-$		V
PSRR	Power Supply Rejection Ratio			30	150	$\mu\text{V/V}$
$I_{OS}$	Output Short Circuit Current (For Amplifier) (Note 4)		$\pm 10$	$\pm 30$	$\pm 45$	mA
$A_{VS}$	Large Signal Voltage Gain	$V_O = \pm 10\text{V}$ , $R_L \geq 2.0\text{ k}\Omega$	50	200		V/mV
$V_{OP}$	Output Voltage Swing	$R_L = 10\text{ k}\Omega$	$\pm 12$	$\pm 13.5$		V
		$R_L = 2.0\text{ k}\Omega$	$\pm 10$	$\pm 13$		
$T_R$	Transient Response	Rise Time	$V_O = 50\text{ mV}$ , $A_V = 1.0$ , $R_L = 10\text{ k}\Omega$	0.3		$\mu\text{s}$
		Overshoot	$V_O = 50\text{ mV}$ , $A_V = 1.0$ , $R_L = 10\text{ k}\Omega$	5.0		%
BW	Bandwidth	$V_O = 50\text{ mV}$ , $A_V = 1.0$ , $R_L = 10\text{ k}\Omega$		1.0		MHz
SR	Slew Rate	$V_I = -10\text{V}$ to $+10\text{V}$ , $A_V = 1.0$		0.6		V/ $\mu\text{s}$

LM3503

Electrical Characteristics  $-55^\circ\text{C} \leq T_A \leq +125^\circ\text{C}$ ,  $V_{CC} = \pm 15\text{V}$ , unless otherwise specified

Symbol	Parameter	Conditions	LM3503			Units
			Min	Typ	Max	
$V_{IO}$	Input Offset Voltage				6.0	mV
$\Delta V_{IO}/\Delta T$	Input Offset Voltage Temperature Sensitivity			10		$\mu\text{V}/^\circ\text{C}$
$I_{IO}$	Input Offset Current				200	nA
$\Delta I_{IO}/\Delta T$	Input Offset Current Temperature Sensitivity			50		$\text{pA}/^\circ\text{C}$
$I_{IB}$	Input Bias Current				1200	nA
$A_{VS}$	Large Signal Voltage Gain	$V_O = \pm 10\text{V}$ , $R_L \geq 2.0\text{ k}\Omega$	25			V/mV
$V_{OP}$	Output Voltage Swing	$R_L = 2.0\text{ k}\Omega$	$\pm 10$			V

The following specifications apply for  $T_A = 25^\circ\text{C}$ ,  $V^+ = +5.0\text{V}$ ,  $V^- = \text{GND}$

$V_{IO}$	Input Offset Voltage			2.0	5.0	mV
$I_{IO}$	Input Offset Current			30	50	nA
$I_{IB}$	Input Bias Current			200	500	nA
$I_{CC}$	Supply Current			2.5	4.0	mA
PSRR	Power Supply Rejection Ratio				150	$\mu\text{V/V}$
$A_{VS}$	Large Signal Voltage Gain	$R_L = 2.0\text{ k}\Omega$	30	300		V/mV
$V_{OP}$	Output Voltage Swing (Note 5)	$R_L = 10\text{ k}\Omega$	3.3			V
		$5.0\text{V} \leq V_I \leq 30\text{V}$ , $R_L = 10\text{ k}\Omega$	( $V_I - 2.0$ )			
CS	Channel Separation	10 Hz to 100 kHz (Input Referenced)		120		nV

Note 1:  $T_A = 25^\circ\text{C}$  for the Models 1303 and 3403, and  $175^\circ\text{C}$  for the Ceramic DIP.

Note 2: Values apply to ambient temperature of  $25^\circ\text{C}$ . At other temperatures, consult the 141, Ceramic DIP at 9.1 mW/ $^\circ\text{C}$ , the 141, Plastic DIP at 0.3 mW/ $^\circ\text{C}$ , and the 3403 at 1.5 mW/ $^\circ\text{C}$ .

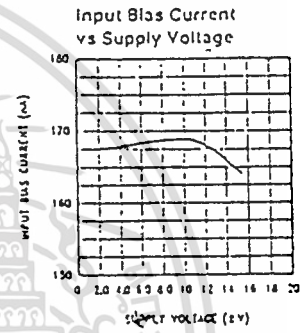
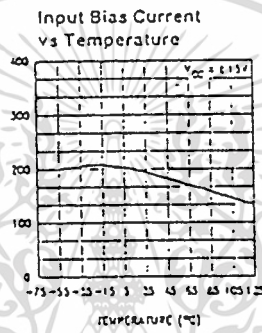
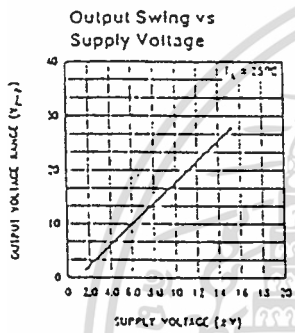
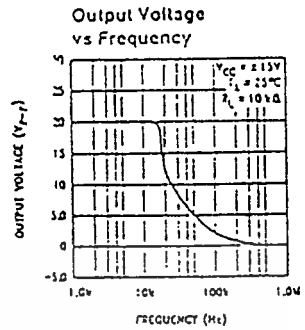
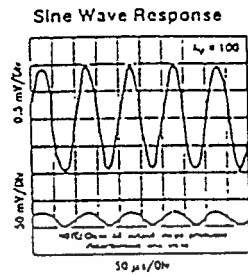
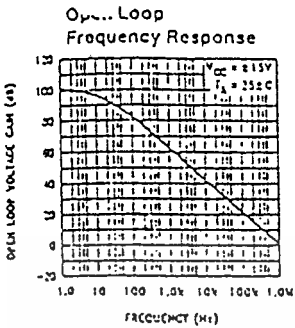
Note 3: For supply voltage less than 30V between  $V^+$  and  $V^-$ , the absolute maximum input voltage is equal to the supply voltage.

Note 4: Not to exceed maximum pin package dissipation.

Note 5: Output will swing to ground.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### Typical Performance Characteristics



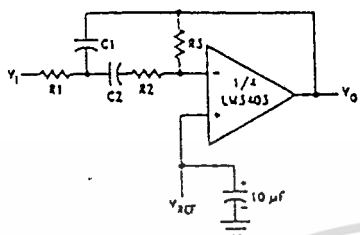
TL/M/10064-3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LM3303/LM3403/LM3503

### Typical Applications

Multiple Feedback Bandpass Filter



TL/M/10064-4

$f_0$  = center frequency  
 BW = Bandwidth  
 R in k $\Omega$   
 C in  $\mu$ F

$$Q = \frac{f_0}{BW} < 10$$

$$C1 = C2 = \frac{10}{Q}$$

$$R1 = R2 = 1 R3 = 9Q^2 - 1 \quad \left\{ \text{Using scaling factors in these expressions.} \right.$$

If source impedance is high or varies, filter may be preceded with voltage follower or buffer to stabilize filter parameters.

Design example:

given:  $Q = 5, f_0 = 1 \text{ kHz}$

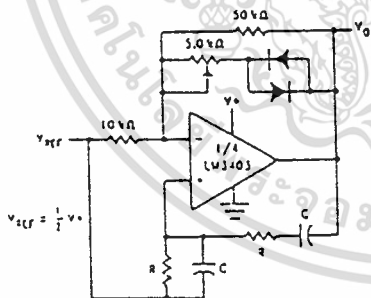
Let  $R1 = R2 = 10 \text{ k}\Omega$

then  $R3 = 9(5)^2 - 10$

$R3 = 215 \text{ k}\Omega$

$$C = \frac{10}{Q} = 1.8 \text{ nF}$$

Wien Bridge Oscillator



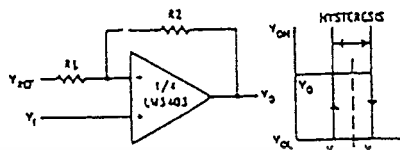
TL/M/10064-5

$$f_0 = \frac{1}{2\pi RC} \text{ for } f_0 = 1 \text{ kHz}$$

$$R = 10 \text{ k}\Omega$$

$$C = 0.01 \text{ }\mu\text{F}$$

Comparator with Hysteresis



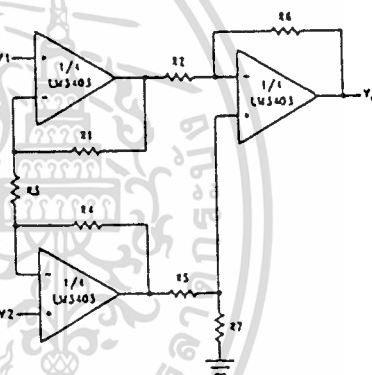
TL/M/10064-6

$$V_{IL} = \frac{R1}{R1 + R2} (V_{OL} - V_{REF}) + V_{REF}$$

$$V_{IH} = \frac{R1}{R1 - R2} (V_{OH} - V_{REF}) + V_{REF}$$

$$H = \frac{R1}{R1 - R2} (V_{OH} - V_{OL})$$

High Impedance Differential Amplifier



TL/M/10064-7

$$V_{OUT} = C(1 + a + b)(V2 - V1)$$

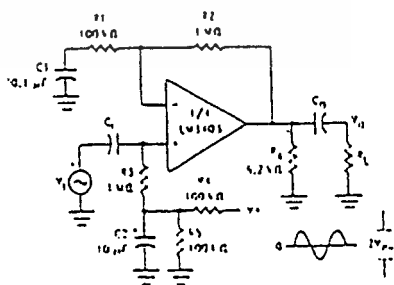
$$\frac{R2}{R3} = \frac{R6}{R7} \text{ for best CMRR}$$

$$R1 = R4$$

$$R2 = R5$$

$$Gain = \frac{R6}{R3} \left( 1 + \frac{2R1}{R2} \right) = C(1 + a + b)$$

AC Coupled Non-Inverting Amplifier



TL/M/10064-8

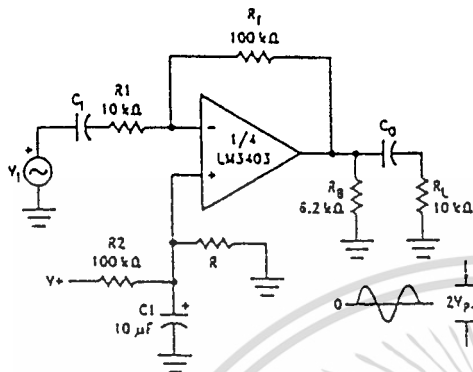
$$A_v = 1 + \frac{R2}{R3}$$

$$A_v = 11 \text{ (for } R2 = 10 \text{ k}\Omega, R3 = 1 \text{ k}\Omega)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Applications (Continued)

AC Coupled Inverting Amplifier



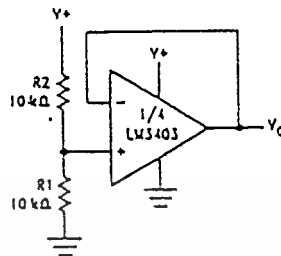
$$V_o = \frac{R_f}{R_1 + R_2} \left( -\frac{V_i}{2} \text{ as shown} \right)$$

$$V_o = -\frac{1}{2} V_i$$

$A_v = \frac{R_f}{R_1}$   
 $A_v = 10$  (as shown)

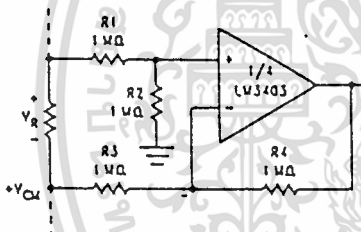
TL/M/10064-8

Voltage Reference



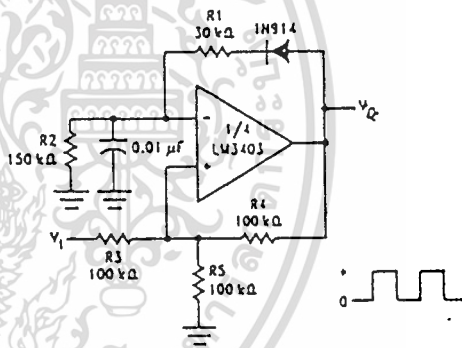
TL/M/10064-10

Ground Referencing a Differential Input Signal



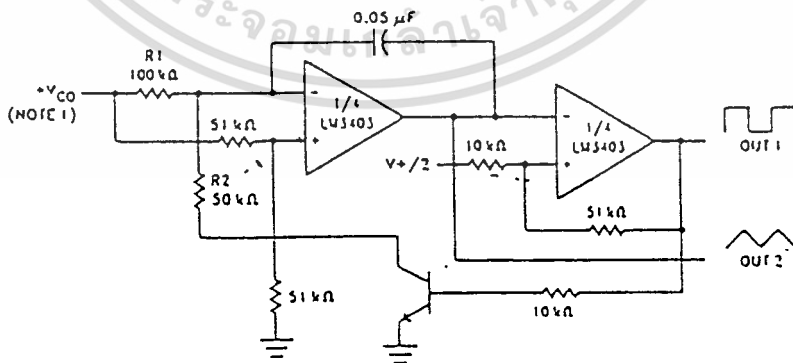
TL/M/10064-11

Pulse Generator



TL/M/10064-14

Voltage Controlled Oscillator



TL/M/10064-17

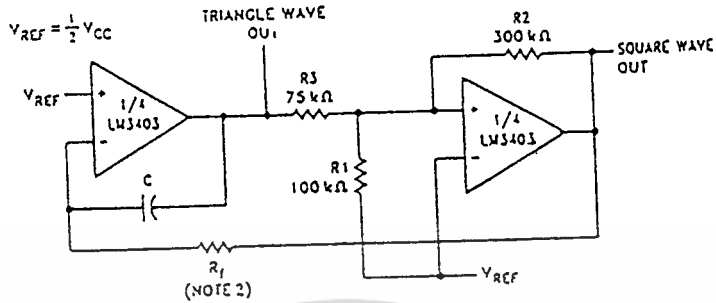
Note 1: Wave Control Voltage Range  
 $0V < V_{CO} < 2(V_{+}/2)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LM3303/LM3403/LM3503

Typical Applications (Continued)

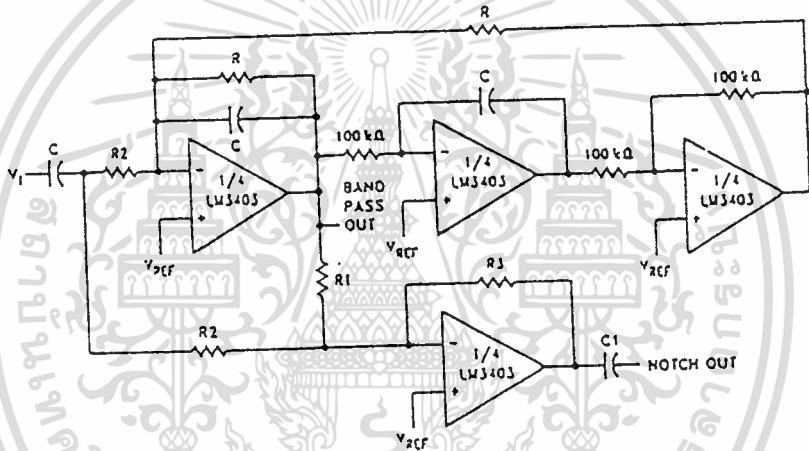
Function Generator



TL/H/10064-13

Note 2:  $f = \frac{R1 + R2}{4CR1R1} \text{ or } f = \frac{R2R1}{R2 + R1}$

Bi-Quad Filter



TL/H/10064-15

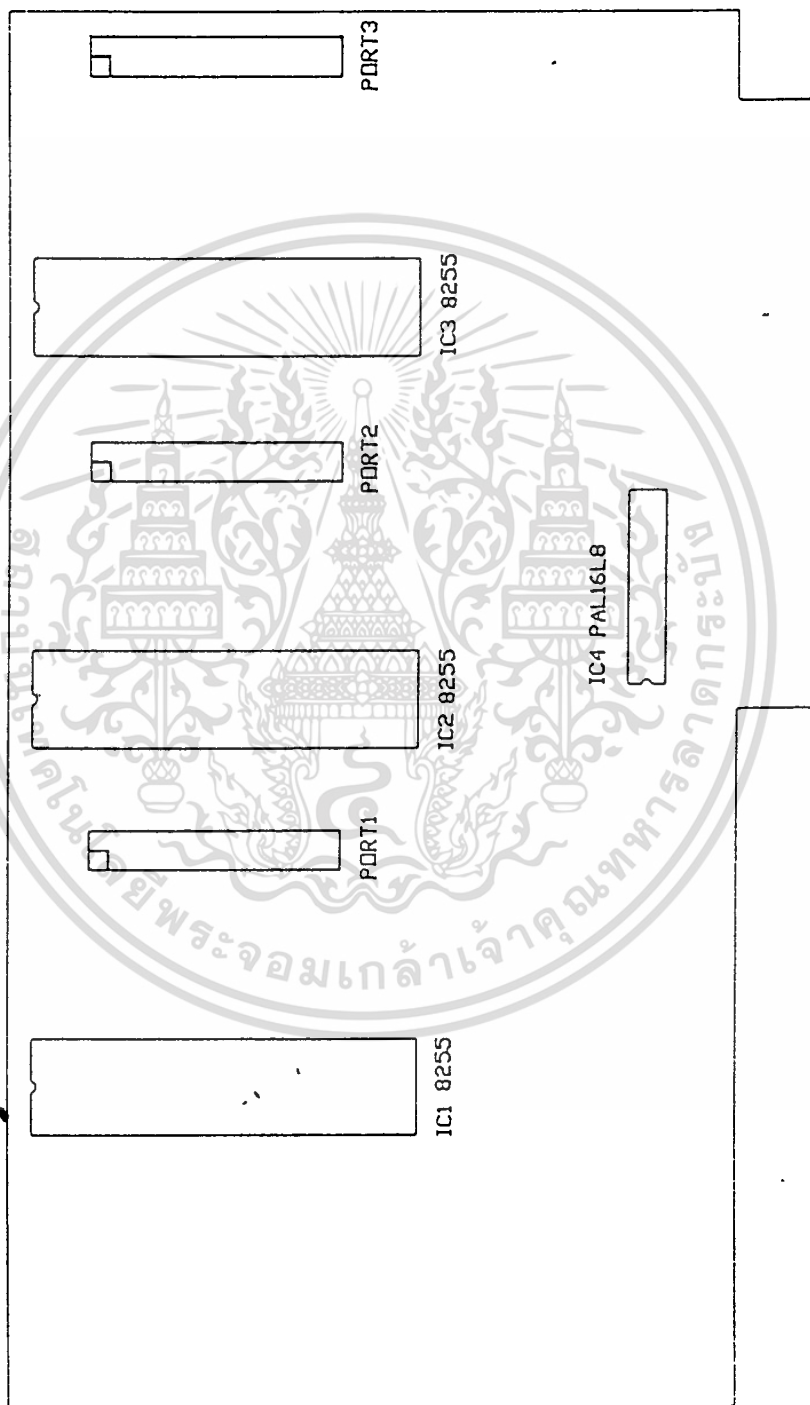
$Q = \frac{BW}{f_0}$   
 where:  
 $f_0$  = Center Frequency Gain  
 $f_{BW}$  = Bandpass Notch Gain  
 $f_0 = \frac{1}{2\pi RC} \cdot V_{REF} = \frac{1}{2} V_{CC}$   
 $R1 = 0.1M$   
 $R2 = \frac{R1}{f_{BW}}$   
 $R3 = f_{BW}^2$   
 $C1 = 10 \mu F$

Example:  
 $f_0 = 1000 \text{ Hz}$   
 $BW = 100 \text{ Hz}$   
 $f_{BW} = 1$   
 $f_M = 1$   
 $R = 100 \text{ k}\Omega$   
 $R1 = 1.6 \text{ M}\Omega$   
 $R2 = 1.6 \text{ M}\Omega$   
 $R3 = 1.6 \text{ M}\Omega$   
 $C = 100 \mu F$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

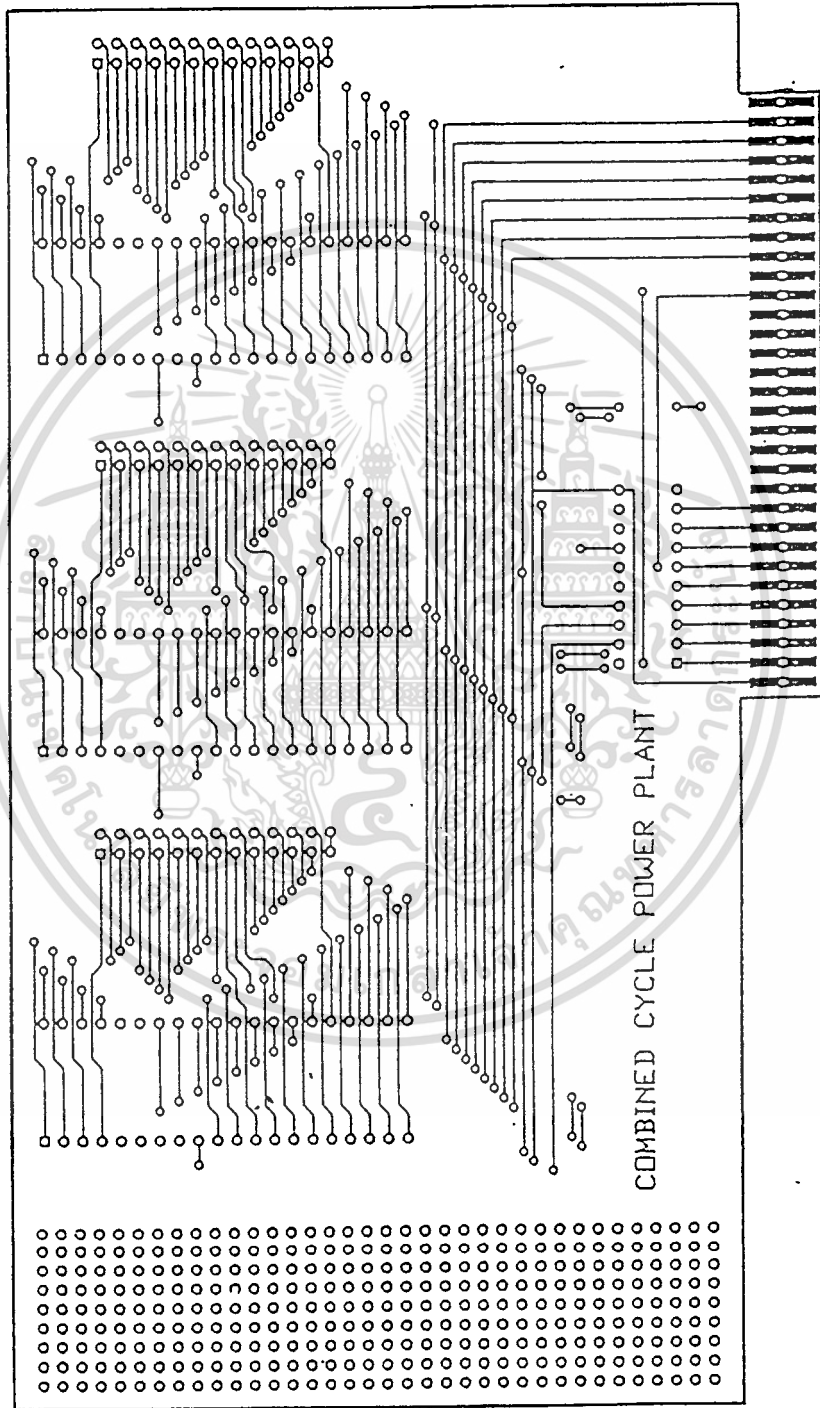


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



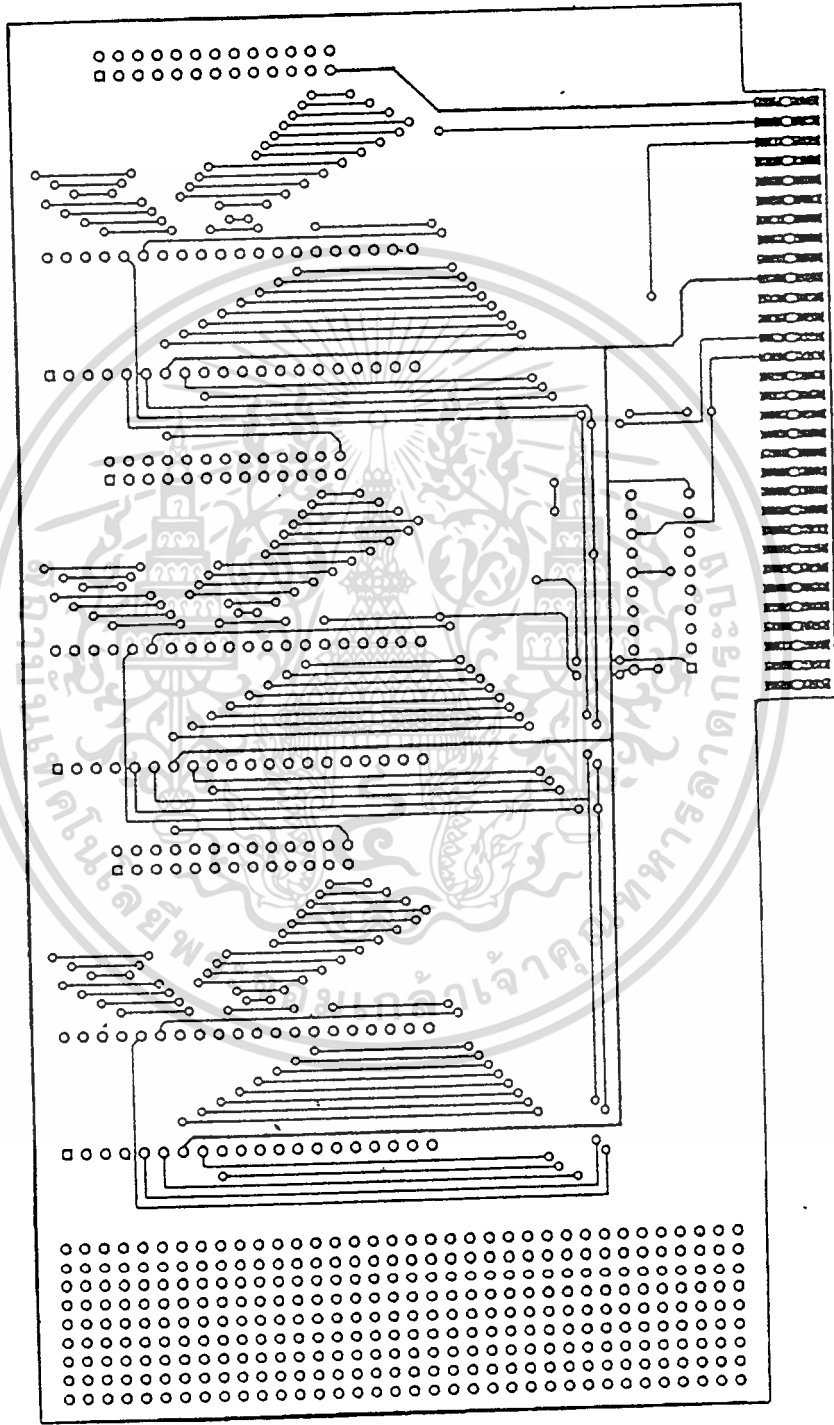
รูป การวางอุปกรณ์ของ DIGITAL CARD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



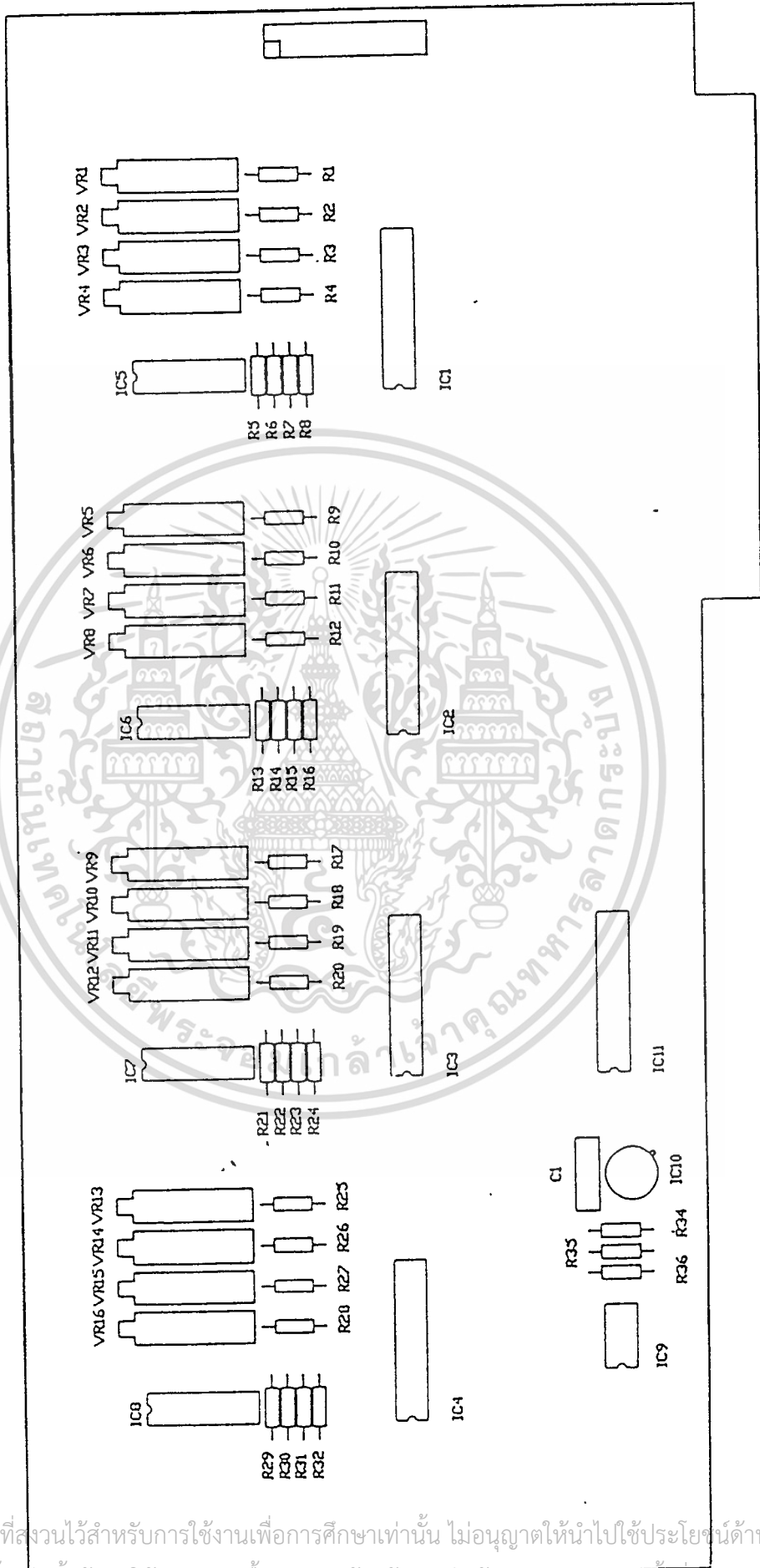
รูป ลายปริ้นท์ของ DIGITAL CARD (I)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



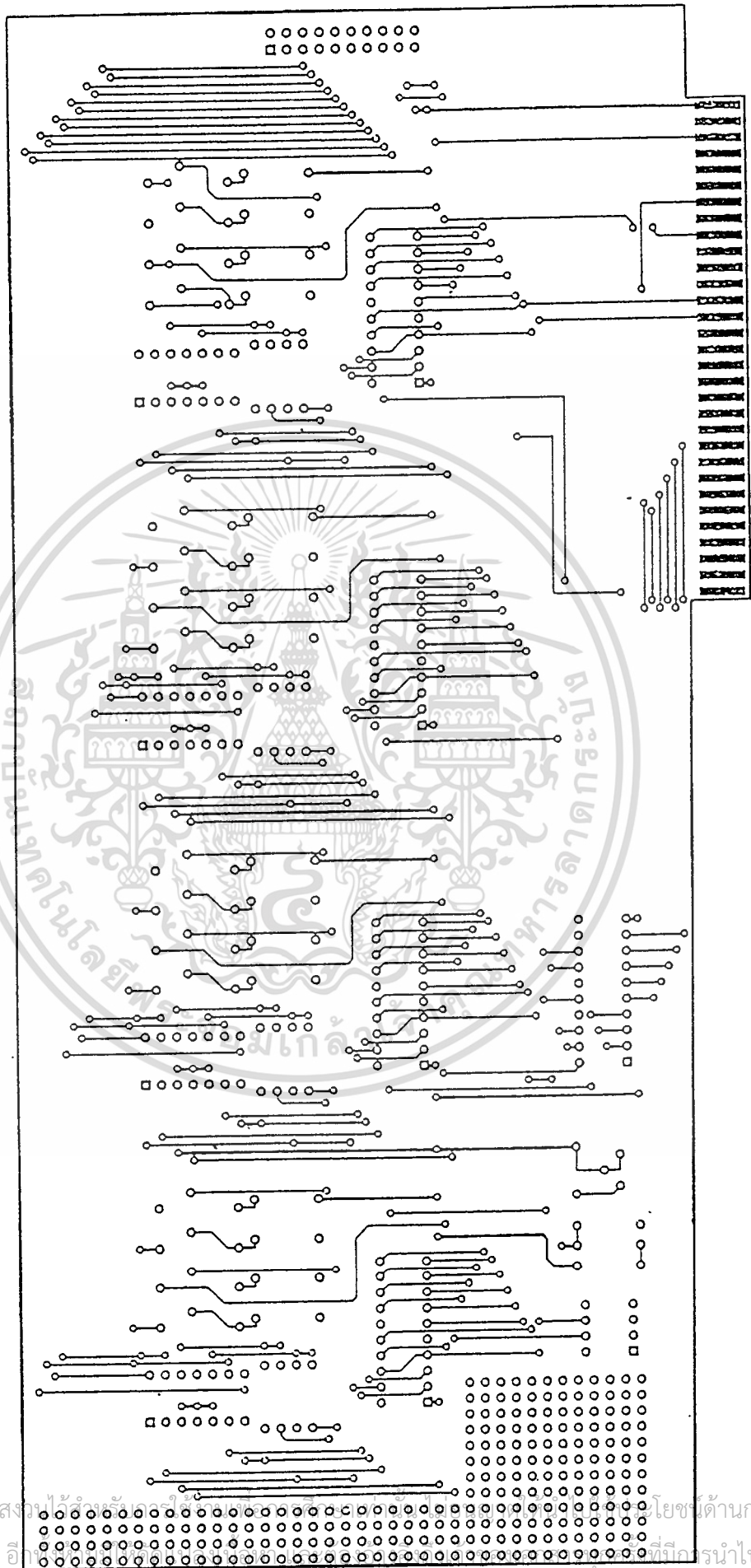
รูป ลายปริ้นท์ของ DIGITAL CARD (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



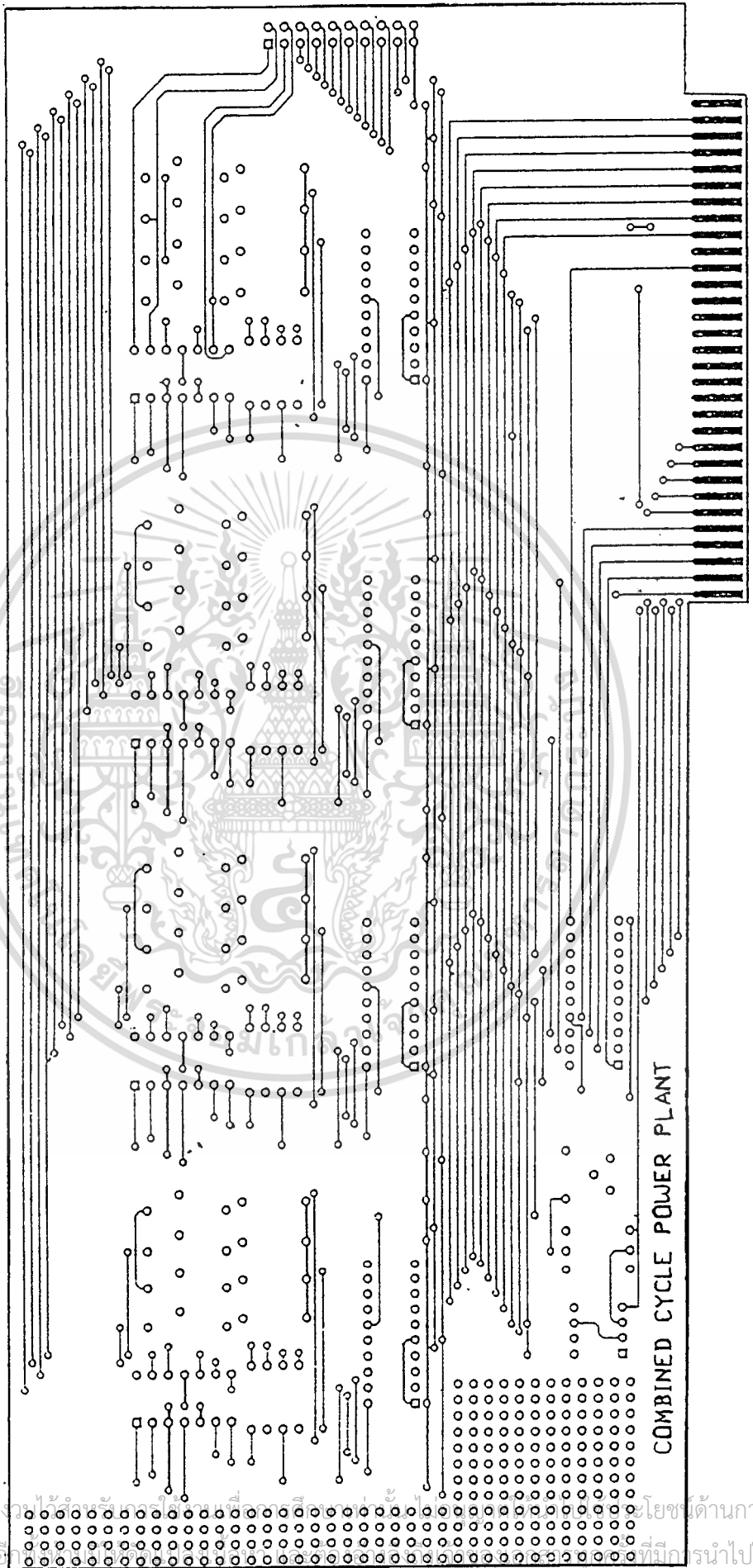
รูป การวางอุปกรณ์ของ ANALOG CARD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นที่ให้มีเหตุที่เปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป ลายปริ้นซ์ของ ANALOG CA(1)

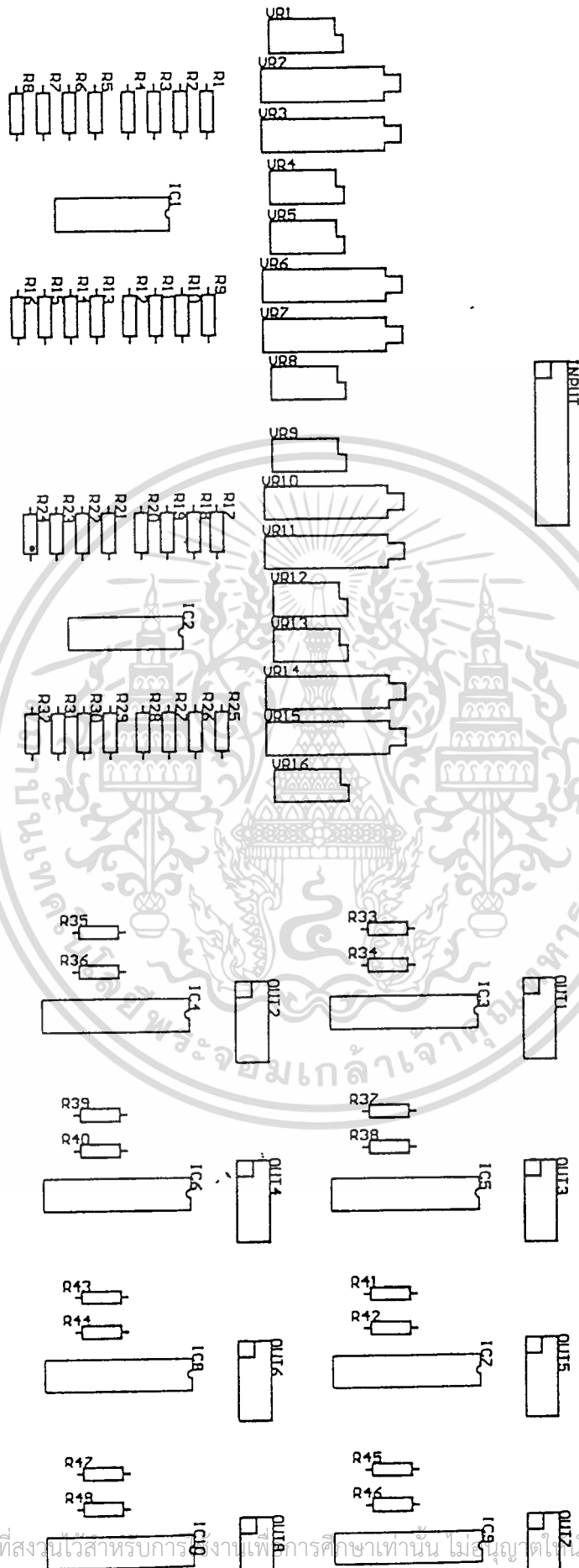
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะในโครงการวิจัยของคณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อื่นๆ ที่เกี่ยวข้องกับการนำเอกสารนี้ไปใช้



COMBINED CYCLE POWER PLANT

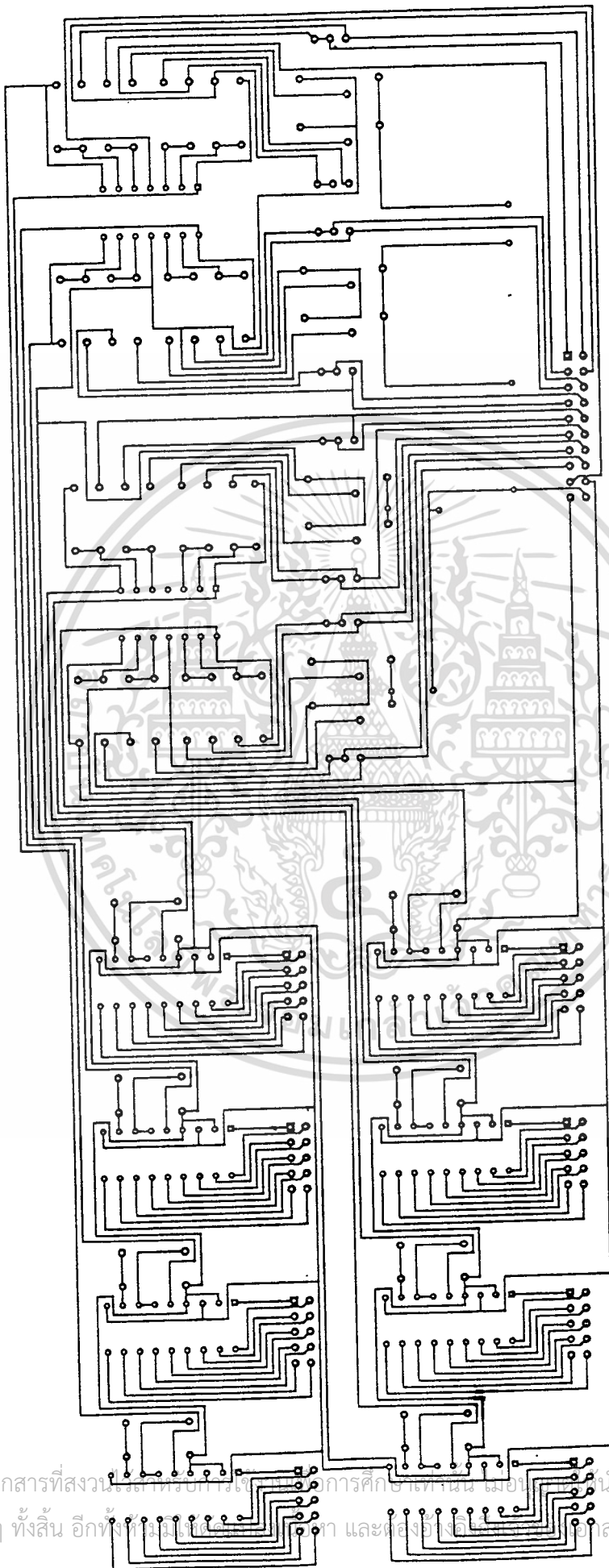
รูป ลายปริ้นต์ของ ANALOG CA(2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรณีฉุกเฉินหรือกรณีศึกษาเท่านั้น ไม่ควรนำเอกสารนี้ไปใช้ในการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังขอสงวนสิทธิ์ในข้อมูลและเนื้อหาที่มีปรากฏในเอกสารนี้



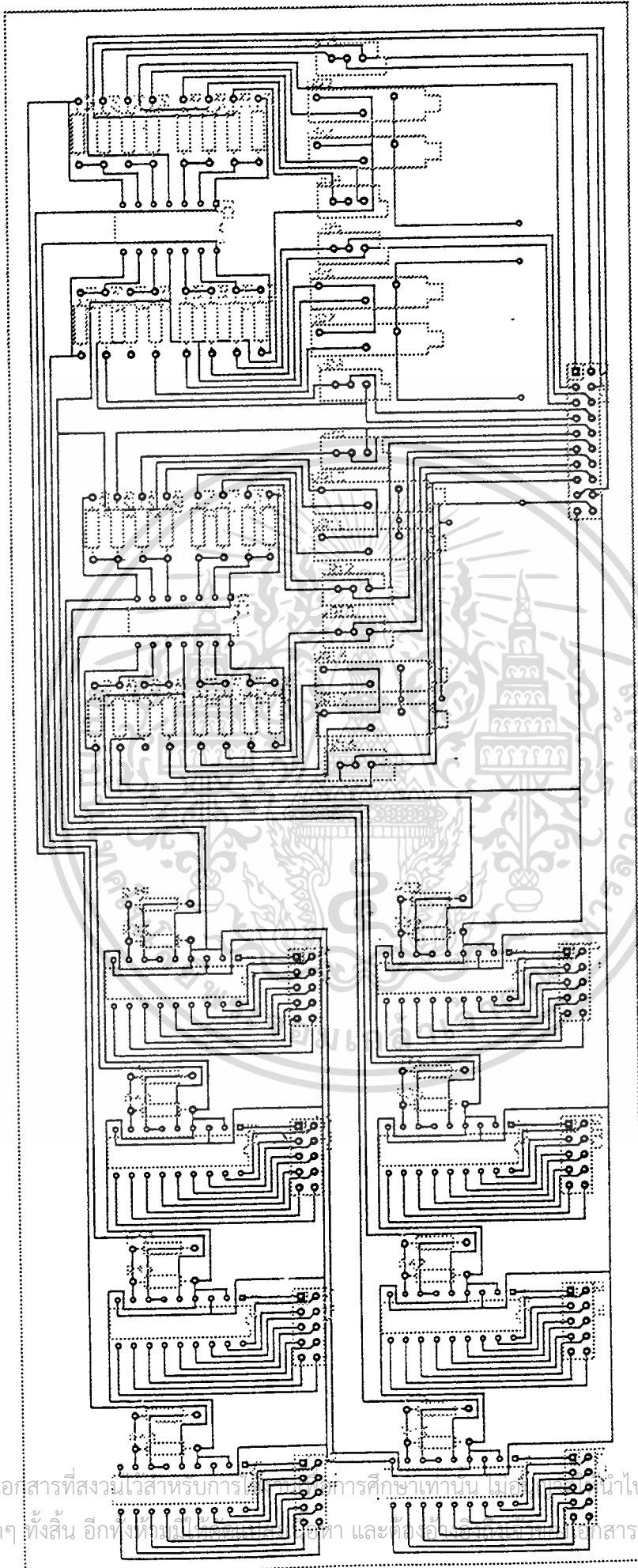
รูป การวางอุปกรณ์ของ BAR DISPLAY DRIVER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

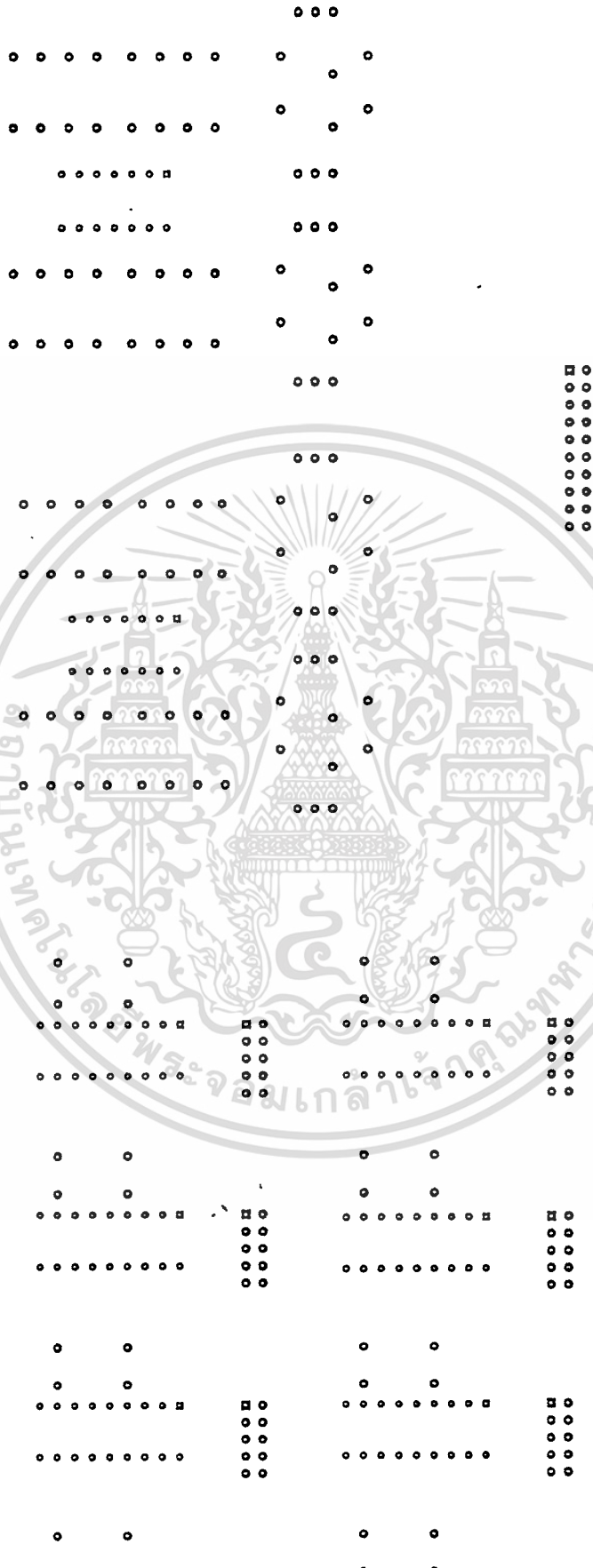


รูป ลายปริ้นซ์ของ BAR DISPLAY DRIVER

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการศึกษาด้านอื่น เมื่อมีการนำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่หรือทำซ้ำ และต้องอ้างอิงถึงเอกสารทุกครั้งที่มีการนำไปใช้



รูป การวางอุปกรณ์และสายปรีนซ์ของ BAR DISPLAY DRIVER



รูป ตำแหน่งการวางอุปกรณ์ของ BAR DISPLAY DRIVER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้