



การควบคุมแบบ PID ด้วย ไมโครคอนโทรลเลอร์ 68HC11

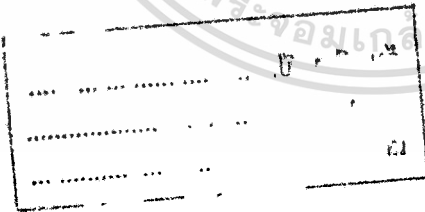
PID CONTROLLER 8 LOOP

WITH MICROCONTROLLER 68HC11



โดย

|               |             |           |
|---------------|-------------|-----------|
| นาย บัญชา     | เสาชงไชย    | 35.102101 |
| นาย ประสิทธิ์ | เยี่ยมทอง   | 35.102103 |
| นาย สมวัฒน์   | โสภณธรรมกิจ | 35.102117 |



ปฏิญญาฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต  
สาขา วิศวกรรมการวัดคุม  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2537

สารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการ

ภาควิชา

เทคโนโลยีการวัดคุมทางอุตสาหกรรม

สาขา

วิศวกรรมการวัดคุมทางอุตสาหกรรม

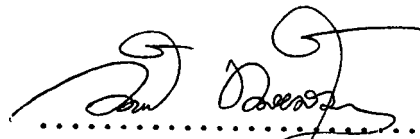
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง

การควบคุมแบบ PID 8 LOOP ด้วย ไมโครคอนโทรลเลอร์ 68HC11  
PID CONTROLLER 8 LOOP WITH MICROCONTROLLER 68HC11

ผู้จัดทำ

|               |             |           |
|---------------|-------------|-----------|
| นาย บัญชา     | เสาชงไชย    | 35.102101 |
| นาย ประสิทธิ์ | เอี่ยมทอง   | 35.102103 |
| นาย สมวัจน์   | โสภณธรรมกิจ | 35.102117 |



.....อาจารย์ที่ปรึกษา

( ผศ. วิทยา ทิพย์สุวรรณพร )

## การควบคุมแบบ PID 8 LOOP ด้วย ไมโครคอนโทรลเลอร์ 68HC11

๙

บัญชา เสาธงไชย  
ประสิทธิ์ เอี่ยมทอง  
ศมวจน์ ไสกณธรรมกิจ  
ผศ. วิทยา ทิพย์สุวรรณพร อาจารย์ที่ปรึกษา

ปีการศึกษา 2537

### บทคัดย่อ

ปฏิญานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของผลงาน ที่เกี่ยวกับการวิจัยชุดควบคุมขบวนการแบบ PID 8 ชุด (PID CONTROLLER 8 LOOP) โดยใช้ไมโครคอนโทรลเลอร์เบอร์ 68HC11 เป็นหน่วยควบคุมหลักและหน่วยรับส่งข้อมูลที่แสดงผลค่าพารามิเตอร์ต่างๆ ด้วย LCD โดยระบบนี้สามารถควบคุมแบบ PID ได้ 8 ลูป ภายใน Sampling Time 62.5 ms โดยแต่ละลูปจะมีโหมดของการควบคุมที่สามารถปรับให้เป็นแบบ PI, PD, หรือ PID ได้อย่างอิสระต่อกันและสามารถ แสดงสภาวะการต่างๆที่ติดตั้งไว้ โดยจะส่งค่าควบคุมจาก Controller เป็นกระแสและรับค่าที่วัดจากกระบวนการเป็นกระแส ในช่วง 4 - 20 mA นอกจากนี้ยังแสดงผลเป็นกราฟแท่งทั้ง 8 Loop พร้อมทั้งจะสามารถกำหนด ค่าพารามิเตอร์ โดยผ่านทาง PORT RS 232 โดยมีอัตราการรับส่งข้อมูล 9600 bit/sec

## PID CONTROLLER 8 LOOP WITH MICROCONTROLLER 68HC11

BUNCHAR SAOPSONGCHAI  
PRASIT AJUMTONG  
SAMAWAT SOPONTHUMMAKIT  
VITTAYA TIPSUVANAPORN ADVISOR

1994

### ABSTRACT

THIS THESIS IS A FRACTION OF 8 LOOP PROCESS CONTROLLER RESEARCH. BY USING MICROCONTROLLER 68HC11 IS A MAIN CONTROL UNIT & COMMUNICATE DATA WITH LCD DISPLAY. THIS CONTROLLER CAN CONTROL PID 8 LOOP WITH IN 62.5 ms SAMPLING TIME. EACH LOOP CONTROL BY INDEPENDENT MODE PI, PD OR PID AND DISPLAY CONDITION OF SETTING CONTROLLER. THE CONTROLLER SENT AND RECEIVE CURRENT SIGNAL 4 - 20 mA. MICROCOMPUTER CAN DISPLAY PID 8 LOOP IN BAR GRAPH WITH 9600 BAUD RATE

## บทนำ

จากอุตสาหกรรมในปัจจุบันจะเห็นว่าต้องการความเร็ว แม่นยำ การประมวลผลที่ถูกต้อง และความสะดวกรวดสบายในการใช้งาน ตลอดจนค่าใช้จ่ายที่จะช่วยลดต้นทุน ในขบวนการผลิต ดังนั้น จึงเป็นเหตุให้เกิดความต้องการระบบอัตโนมัติมา ใช้ทำหน้าที่ดังกล่าวมา ซึ่งในระบบปัจจุบันนี้ระบบการควบคุมจะแบ่งออกเป็น หลากแบบหลายชนิดโดยขึ้นอยู่กับจุดประสงค์ของ การใช้งานโดยส่วนมากแล้วในอุตสาหกรรมต่างๆ จะนิยมใช้ตัวควบคุมแบบ PID เพราะสามารถตอบสนองต่อระบบที่ถูกควบคุมได้ดี และลดค่าผิดพลาดต่างๆตงันั้น ปรินญญาณิพนธ์ฉบับนี้ จะศึกษาค้นคว้าตัวควบคุมแบบ PID



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## วัตถุประสงค์ของปริญญาโท

เนื่องจากระบบควบคุมแบบ PROPORTIONAL INTEGRAL AND DERIVATIVE CONTROL หรือ PID CONTROL เป็นระบบควบคุมที่มีประสิทธิภาพมากพอสมควร จึงเป็นระบบที่มีผู้สนใจให้ความสนใจเป็นจำนวนมากที่จะนำมาเป็นระบบควบคุม แต่มีขีดจำกัดในด้านประสิทธิภาพของตัว CPU ที่จะนำมาใช้

ในปัจจุบันได้มี CPU เบอร์หนึ่งซึ่งที่น่าสนใจมาก ทั้งในด้านความเร็ว ประสิทธิภาพและราคา เพราะใน CPU ตัวนี้มี A/D อยู่ภายในถึง 8 ช่อง ซึ่ง CPU เบอร์นี้คือ 68HC11 ของ MOTOROLA จึงได้นำมาประยุกต์ใช้งานให้ได้เต็มประสิทธิภาพ โดยนำมาใช้ทั้ง 8 ช่อง และพัฒนาให้มีประสิทธิภาพทั้งในด้านการคำนวณ ความแม่นยำและรวดเร็วในการควบคุม จากเดิมที่มีผู้เคยทำมาแล้ว

## ขอบเขตของปริญญาโท

เป็นการนำ Microcontroller มาทำการพัฒนา Soft ware เพื่อให้มีขีดความสามารถสูงสุดที่จะนำมาใช้ ในการทำเครื่องควบคุมระบบ PID 8 Loop

ใบปริญญาโทฉบับนี้ประกอบด้วยส่วนหลักอยู่ 2 ส่วนด้วยกันคือ

- HARD WARE
- SOFT WARE

โดยที่ส่วนของปริญญาโทฉบับนี้เราได้ทำการศึกษาและพัฒนาในเรื่องของ Soft ware และการศึกษาความสามารถของ CPU เบอร์ 68HC11 เป็นหลัก เนื่องจากเป็นส่วนที่สำคัญที่สุดในการประมวลผล และ การควบคุมส่วน HARD WARE นั้นไม่ค่อยมีอะไรซับซ้อนมากนัก ดังนั้น ข้อมูลทาง HARD WARE สามารถศึกษาได้จาก Flow chat และวงจรที่อยู่ใน APPENDIX C

ส่วน Soft Ware เราได้มุ่งเน้นในเรื่องของการแก้สมการในลักษณะการควบคุมแบบ PID ให้อยู่ในรูปที่มีประสิทธิภาพสูงสุด เพื่อที่จะได้ใช้เวลาในการประมวลผลได้อย่างรวดเร็วและจัดสรรเวลาการควบคุมอุปกรณ์ต่าง ๆ เพื่อการติดต่อกับอุปกรณ์ภายนอกในรูปแบบต่าง ๆ

# สารบัญ

## เรื่อง

- บทคัดย่อ
- ABSTRACT
- บทนำ
- วัตถุประสงค์ของปริญญาานิพนธ์
- ขอบเขตของปริญญาานิพนธ์

## เนื้อเรื่อง

### บทที่ 1

- หลักการเบื้องต้นของการควบคุมกระบวนการ..... 1
  - 1. ระบบควบคุมแบบลูปปิด
  - 2. ชนิดของการควบคุม
    - ระบบการควบคุมแบบสัดส่วน
    - ระบบการควบคุมแบบร่วมกับ
- ตัวควบคุมและแบบการควบคุม..... 5
  - Proportional Action
  - Integral Action
  - Proportional Integral Action
  - Derivative Action
  - Proportional Derivative Action
  - Proportional Integral Derivative Action
- PID Controller..... 15
- การเขียนโปรแกรมการคำนวณ..... 16
- การปรับค่า PID..... 20
  - เทคนิคการปรับค่า PID
- การรบกวนการกันและกันของค่า PID..... 27

### บทที่ 2

- ไมโครคอนโทรลเลอร์ 68HC11..... 29
  - คุณสมบัติ 68HC11
  - ตระกูลของ 68HC11
  - โหมดการทำงาน
  - ลักษณะของสัญญาณและการจัดขา
  - การจัดหน่วยความจำ
  - หน่วยความจำแรมภายในซีพียู
  - วงจรกำเนิดสัญญาณนาฬิกา
  - รีจิสเตอร์
  - วงจรเอทิวติ
  - รีลไทม์
  - พอสต์แอกทีวูเลเตอร์

|                                    |           |
|------------------------------------|-----------|
| - รีเซต                            |           |
| - อินเทอร์รัพต์                    |           |
| - พอร์ตอนุกรม                      |           |
| - มัลติโปรเซสเซอร์                 |           |
| - V TO I Converter Circuit.....    | <b>43</b> |
| - I TO V Converter Circuit.....    | <b>47</b> |
| - แสดงผังการทำงานของ DAC 0808..... | <b>49</b> |
| - ลักษณะคุณสมบัติของ DAC 0808..... | <b>50</b> |

### บทที่ 3

|                              |           |
|------------------------------|-----------|
| - การทดลองและผลการทดลอง..... | <b>51</b> |
|------------------------------|-----------|

บทสรุป  
 กติกรรมประกาศ  
 เอกสารอ้างอิง

|            |                    |
|------------|--------------------|
| APPENDIX A | - GRAPHICS PROGRAM |
| APPENDIX B | - MONITOR PROGRAM  |
| APPENDIX C | - HARD WARE        |

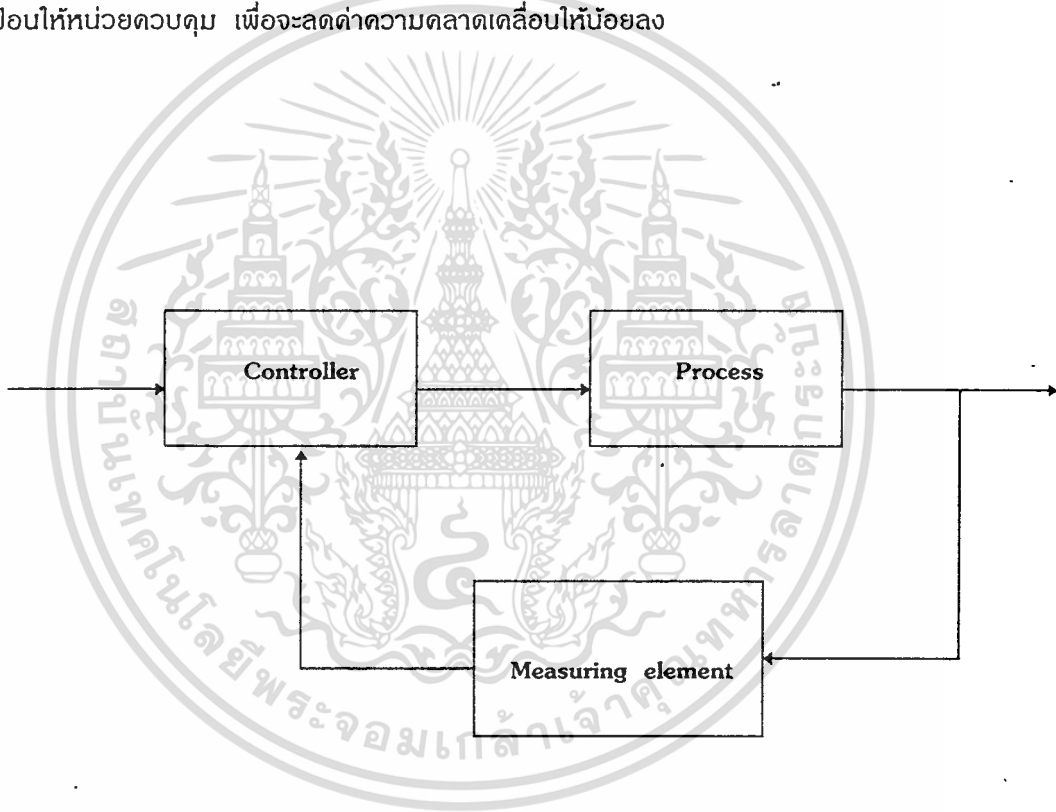


# บทที่ 1

## หลักการเบื้องต้นของการควบคุมกระบวนการและการคำนวณ

### 1. ระบบควบคุมแบบลูปปิด [ Close Loop Control System ]

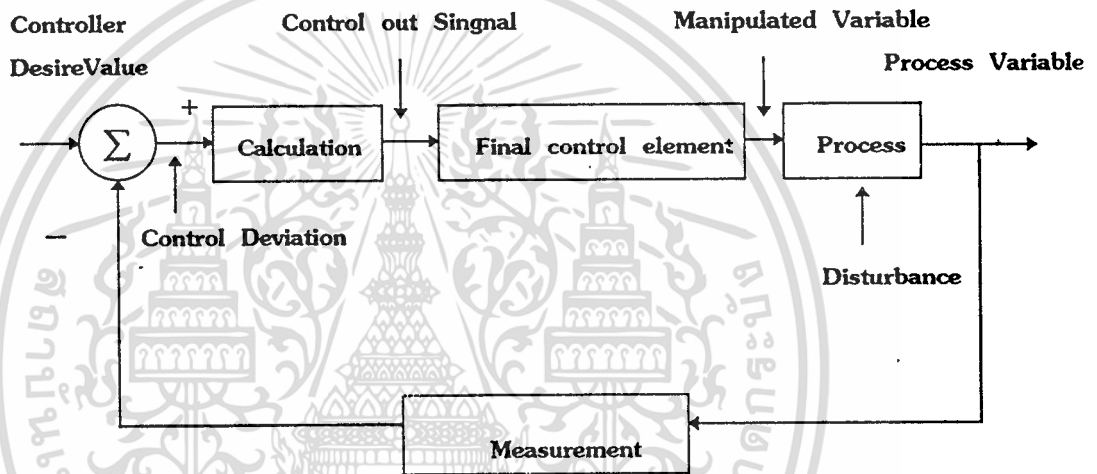
ระบบควบคุมแบบนี้ เป็นระบบควบคุมที่นำเอาสัญญาณ OUTPUT ป้อนกลับ ( Feedback Out Put ) มาเปรียบเทียบกับสัญญาณ INPUT ที่ต้องการ สัญญาณความคลาดเคลื่อน ( Actuating Error Signal ) ซึ่งเป็นสัญญาณ ระหว่าง สัญญาณ INPUT กับสัญญาณป้อนกลับ และจะถูกป้อนให้หน่วยควบคุม เพื่อจะลดค่าความคลาดเคลื่อนให้น้อยลง



รูปที่ 1.1 ระบบการควบคุมแบบลูปปิด

## 2. ชนิดของระบบควบคุม

ระบบควบคุมอัตโนมัติ คือ การใช้ตัวควบคุมซึ่งทำหน้าที่เปรียบเทียบบัดสินใจและปรับแทนมนุษย์นั่นเองการทำงานของระบบอัตโนมัติเป็นการนำค่าเป้าหมายกับค่าของการตรวจวัดใน PROCESS เข้ามาทำการคำนวณหาค่าที่เหมาะสมเพื่อส่งเป็นสัญญาณควบคุมออกไปควบคุม PROCESS ให้มีค่าเท่ากับค่าเป้าหมายอยู่ตลอดเวลา แบบของการควบคุมมีอยู่หลายแบบเช่น ON - OFF

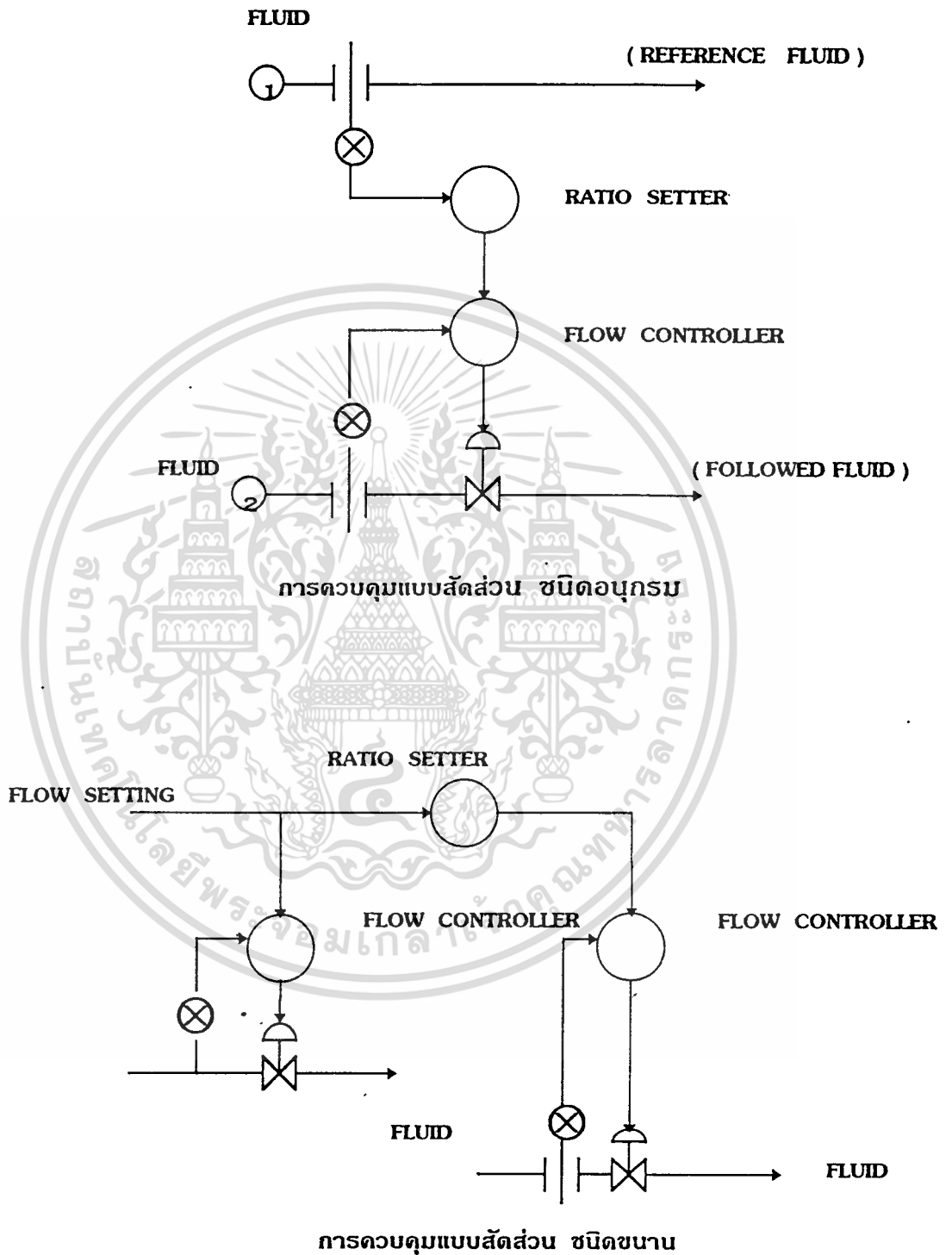


รูปที่ 1.2 บล็อกไดอะแกรมของระบบควบคุมอัตโนมัติ

### ระบบการควบคุมแบบสัดส่วน ( Ratio control )

เป็นการควบคุมให้ตัวแปร PROCESS สองตัวรักษา อัตราส่วนของมันและกันให้คงที่อยู่เสมอ มักพบในการควบคุมอัตราการไหลของอากาศและน้ำมันเชื้อเพลิง ในระบบเผาไหม้ต่างๆ และการควบคุมการผสมของเหลวสองชนิด ตัวอย่างของ Ratio control แสดงดังรูป 1.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

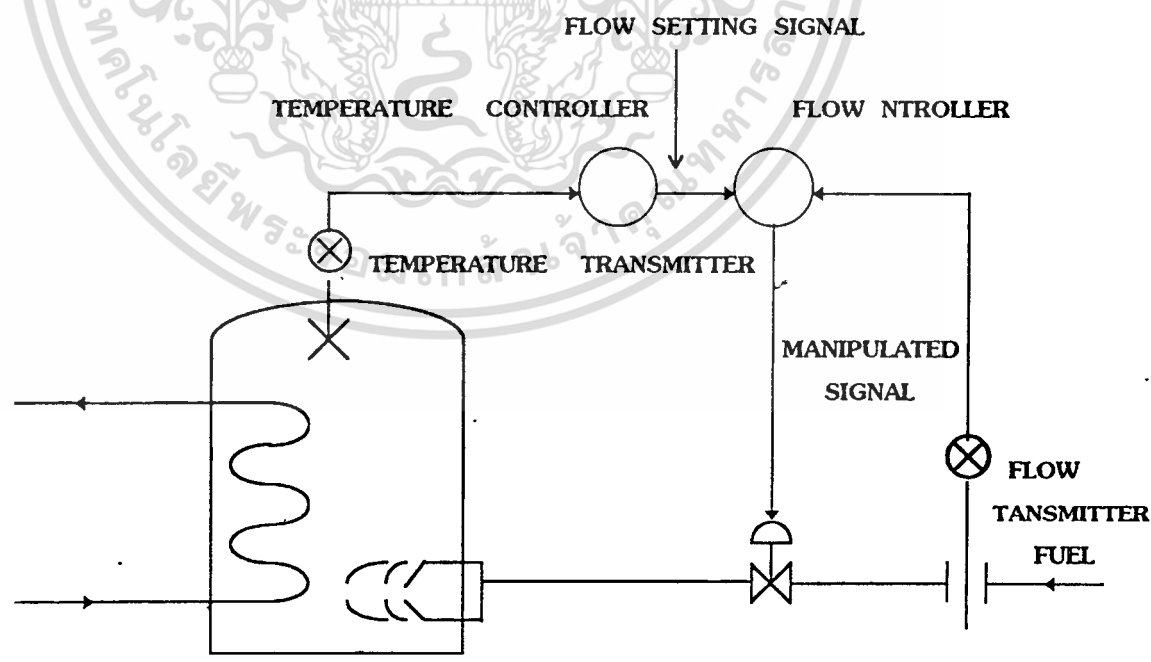


**รูปที่ 1.3 ระบบควบคุมแบบ Ratio Control**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ระบบการควบคุมแบบร่วมกัน ( Cascade Control )

Cascade Control เป็นการควบคุมชนิด Combine Control ชนิดหนึ่ง นิยมใช้กันมาก ในการควบคุมโปรเซส การควบคุมแบบ Cascade Control นี้จะมีรูป การควบคุมอยู่ 2 รูป ใน 2 รูปนี้จะมีตัวควบคุมคือ Master Controller และ Slave Controller สัญญาณออกของ Master Controller จะเป็นค่าเป้าหมายของ Slave Controller cascade Controller นี้ จะใช้ได้ผลดีในโปรเซสที่มี Disturbance มี Dead Time มาก นอกจากนี้ ในกรณีที่ต้องการรักษาความสัมพันธ์ของตัวโปรเซสสองตัวให้คงที่ จะเหมาะสมที่สุด ดังในรูป 1.4 เป็นตัวอย่าง Cascade Control ของการควบคุมอุณหภูมิในเตา รูปการควบคุม อุณหภูมิ จะเป็น Master และรูปการควบคุมการไหลจะเป็น Slave สัญญาณออกของตัวควบคุม อุณหภูมิจะส่งไปเป็นค่าเป้าหมายของตัวควบคุมการไหล รูปการควบคุมการไหลจะควบคุมให้การไหล ของ วัสดุเป็นไปตามสัญญาณจากตัวควบคุมอุณหภูมิเสมอ แม้ว่าความดันในท่อก๊าซจะเปลี่ยนแปลงไป อย่างไรก็ตาม จะไม่มีผลต่ออุณหภูมิในเตาเลย การลดลงหรือเพิ่มขึ้นของความดันก๊าซในท่อนับเป็น Disturbance อย่างหนึ่ง ถ้าเราใช้ระบบควบคุมแบบธรรมดาซึ่งสัญญาณออกของตัวควบคุมอุณหภูมิ ส่งมาควบคุมการเปิดของ Valve โดยตรงแล้ว Disturbance จะทำให้อุณหภูมิในเตาไม่คงที่หรือเท่ากับ ค่าเป้าหมายที่ตั้งไว้

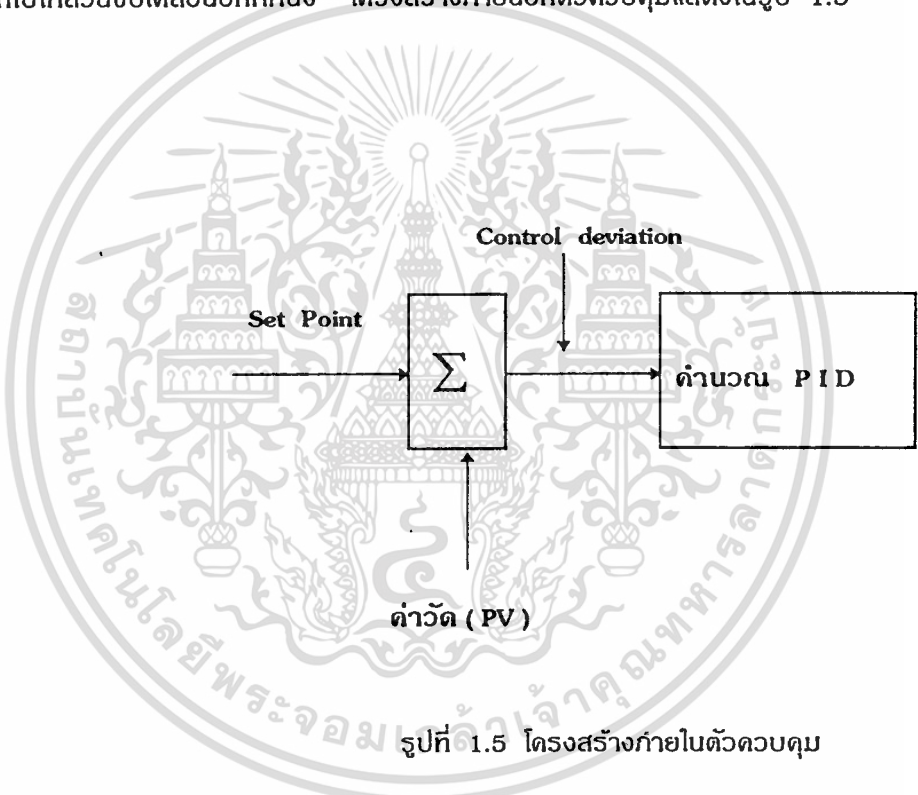


รูปที่ 1.4 ระบบควบคุมร่วมกัน ( Cascade Control )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตัวควบคุมและแบบการควบคุม ( Controller and Control Action )

ตัวควบคุมจะรับสัญญาณเข้าจากการวัด เพื่อทำการเปรียบเทียบกับค่าเป้าหมาย หรือ Set point ผลต่างของค่าทั้งสองจะถูกส่งให้ภาคคำนวณ เพื่อผลิตสัญญาณควบคุมขาออกซึ่งจะถูกส่งออกไปให้ส่วนขับเคลื่อนอีกทีหนึ่ง โครงสร้างภายนอกตัวควบคุมแสดงในรูป 1.5



รูปที่ 1.5 โครงสร้างภายนอกตัวควบคุม

ความสัมพันธ์ของผลต่างกับสัญญาณควบคุมขาออก กำหนดโดย ภาคคำนวณ ความสัมพันธ์นี้เรียกว่า Control Action แบ่งออกเป็น 4 ชนิดคือ

ON-OFF ACTION

PROPORTIONAL ACTION ( P - Control )

INTEGRAL OR RESET ACTION ( I - Action )

DERIVATIVE ACTION ( D - Action )

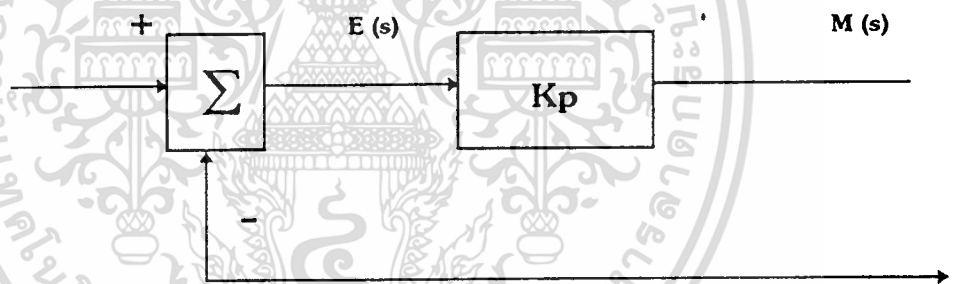
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมจะใช้ Action แต่ละชนิด หรือ หลายชนิดผสมกันเช่น On - off , P , PI , PID , การควบคุมด้วยมือ นั้นถ้าคนงานมีความชำนาญมาก การปรับจะมีลักษณะคล้ายคลึงกับ Control Action เหล่านี้ PROPORTIONAL ACTION คือ ความสัมพันธ์ระหว่างสัญญาณควบคุม ( หรือ OUT PUT ของหน่วยควบคุม ) กับสัญญาณคลาดเคลื่อน  $e(t)$  คือ

$$m(t) = K_p * e(t)$$

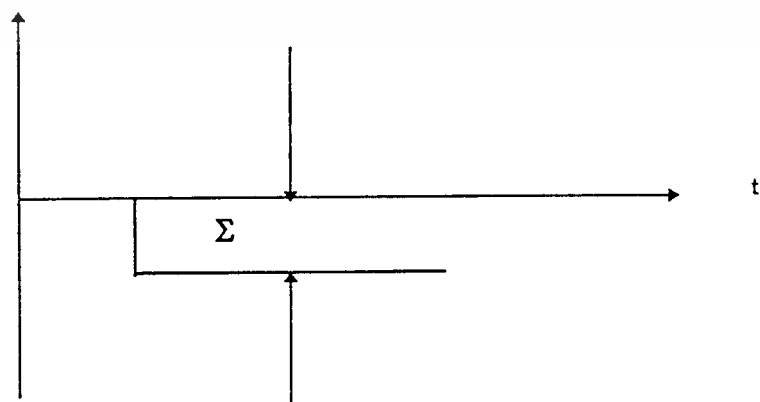
$$\text{หรือ } M(s)/E(s) = K_p$$

โดย  $K_p$  จะอยู่ในเทอมของ Proportional Sensitivity หรือ Gain หน่วยควบคุมแบบ Proportional จึงเป็น Amplifier ที่สามารถปรับค่าของ Gain ได้ดังรูป 1.6 แสดงถึง Block Diagram ของหน่วยควบคุมแบบ proportional ที่กล่าวมานี้

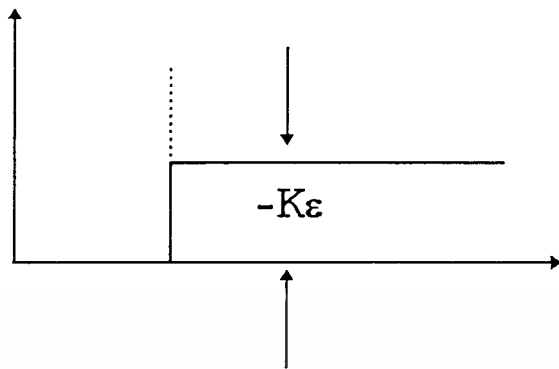


รูป 1.6 Block Diagram ของหน่วยควบคุมแบบ P

คำวัต



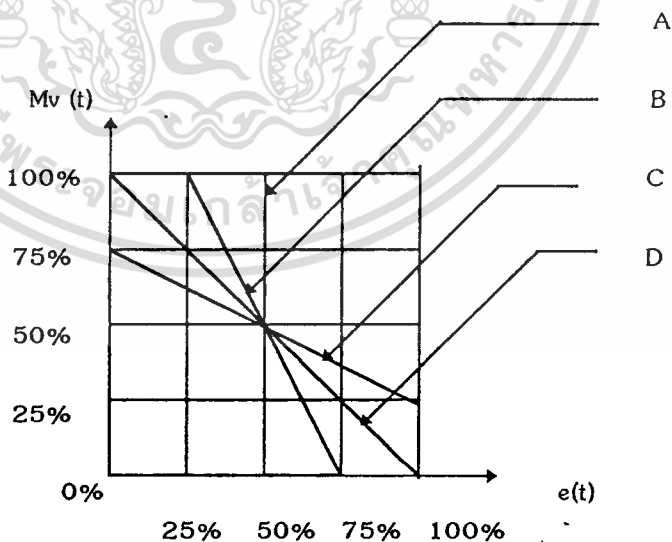
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.7 ผลตอบสนองของ Proportion Action

การควบคุมแบบนี้มักแสดงในรูป Proportional Band (%PB) ค่า PB คือ ค่าสัญญาณ Error ที่ทำให้สัญญาณควบคุมเปลี่ยนไป 100% เมื่อ PB = 50% หมายถึง ถ้าสัญญาณ Error เกิดขึ้น 50% สัญญาณควบคุมเปลี่ยนไป 100% ดังในรูป 1.8

$$PB = \frac{1}{K_p} * 100$$



รูปที่ 1.8 Proportional Band

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|    |           |   |                |             |
|----|-----------|---|----------------|-------------|
| A; | PB = 0%   | , | $K_p = \infty$ | ( 100/0 )   |
| B; | PB = 50%  | , | $K_p = 2$      | ( 100/50 )  |
| C; | PB = 100% | , | $K_p = 1$      | ( 100/100 ) |
| D; | PB = 200% | , | $K_p = 0.5$    | ( 100/200 ) |

จากรูป 1.8 แสดงว่าค่า PB เป็นตัวบอกความไวของการแก้ความผิดพลาดจึงอาจเรียกได้ว่า  $K_p$  เป็น ความไวของ Proportional Control

### คุณสมบัติของ Proportional Action

- เมื่อลด PB ลงไปจะทำให้อัตราขยายสูงขึ้น การควบคุมจะไวขึ้นทำให้โปรเซสเกิดการแกว่งขึ้นได้
- เมื่อเพิ่มค่า PB อัตราขยายจะลดลง จะทำให้ค่าวัดกับค่าเป้าหมายแตกต่างกันมากขึ้น เราว่าเกิด Off Set
- เมื่อโหลดของโปรเซส สภาพแวดล้อมเปลี่ยนไป ซึ่งเราเรียกว่ามี Disturbance ก็เกิด Off Set ได้ง่าย

### OFF SET

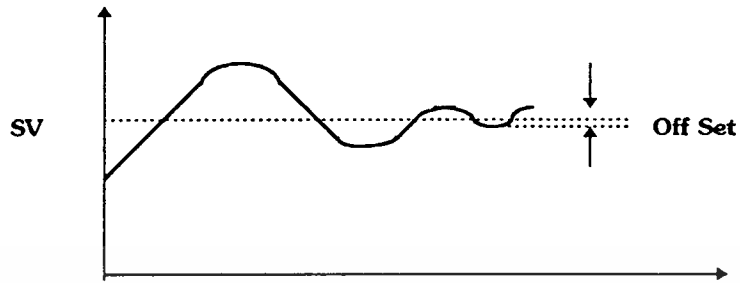
Off Set เป็นชื่อใช้เรียกปรากฏการณ์ที่ตัวแปรโปรเซสหรือวัดมีค่าไม่เท่ากับค่าเป้าหมาย ทำให้การควบคุมไม่เป็นไปตามต้องการ รูป 1.9 แสดงความหมายของ Off Set ในระบบการควบคุมแบบป้อนกลับนี้เมื่อดูอย่างผิวเผิน Off Set ไม่น่าจะเกิดขึ้นได้เพราะตัวแปรโปรเซสจะถูกป้อนกลับมาเปรียบเทียบกับค่าเป้าหมายตลอดเวลา อย่างไรก็ตาม Off Set มักจะเกิดขึ้นกับระบบควบคุมที่ใช้ Proportional Control ที่มีค่า PB ใหญ่และขณะที่เกิด Disturbance ในโปรเซสซึ่งได้แก่การเปลี่ยนแปลงของโหลดสภาพแวดล้อม เป็นต้น

### วิธีการแก้ Off Set ที่เกิดขึ้นใน Process

1. ให้ลด PB ให้เล็กน้อย
2. เปลี่ยน BIAS ของการควบคุมด้วยมือ
3. เปลี่ยนค่าเป้าหมายใหม่
4. ใช้การควบคุมที่มี Reset Action



Process Variable



รูปที่ 1.9 Show Off Set

Integral Action (Reset Action)

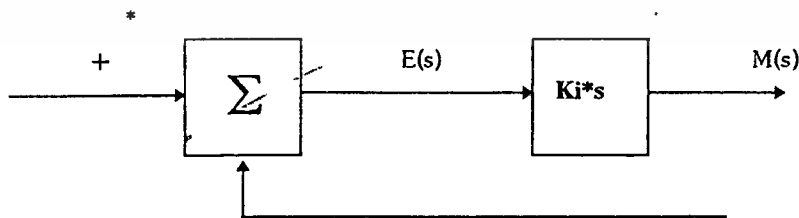
Integral Action คือ ค่าของเอาต์พุทของหน่วยควบคุม  $m(t)$  จะมีความสัมพันธ์กับความคลาดเคลื่อน  $e(t)$  ดังนี้

$$\begin{aligned} \frac{d m(t)}{dt} &= (1/T_i) * e(t) \\ \int d m(t) &= (1/T_i) * \int e(t) dt \\ m(t) &= (1/T_i) * \int e(t) dt \end{aligned}$$

โดยที่  $T_i$  เป็นค่า Integral Time ที่สามารถปรับค่าได้ Transfer Function ของหน่วยควบคุมแบบ Integral คือ

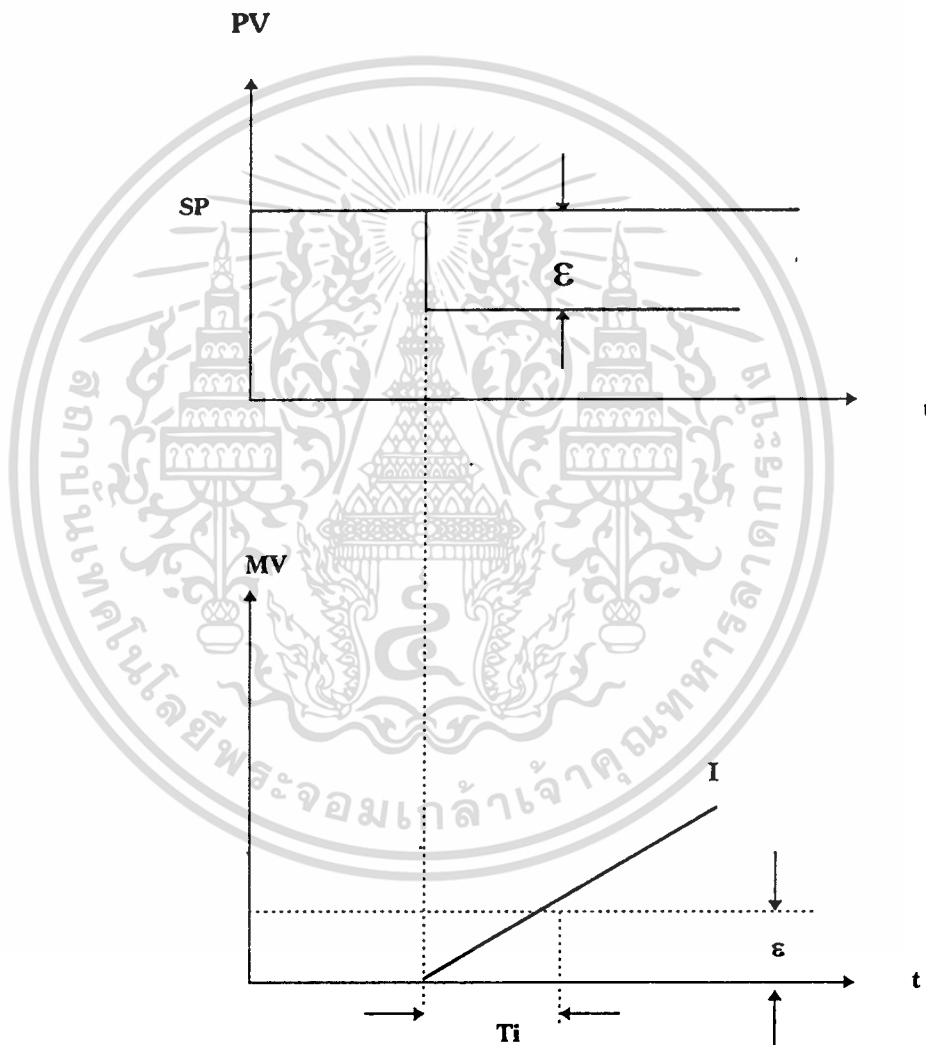
$$\begin{aligned} M(s)/E(s) &= (1/T_i)*s \\ &= K_i * s \end{aligned}$$

โดยที่  $K_i = 1/T_i$



รูปที่ 1.10 บล็อกไดอะแกรมของระบบควบคุมแบบ Integral

กริยาการควบคุมแบบ Integral นี้บางครั้งเราเรียกว่า Reset Control การควบคุมแบบนี้สัญญาณควบคุมจะแปรค่าตามอินทิเกรตต่อเวลาของผลต่าง เมื่อผลต่างเปลี่ยนเป็น Step สัญญาณควบคุมจะค่อยๆเพิ่มขึ้นตามค่าเวลา ดังรูป 1.11 ตามรูปความหมายของ Integral time ( $T_i$ ) คือเวลาที่ใช้ในการเพิ่มค่าของสัญญาณควบคุมจนมีขนาดเท่ากับผลต่าง  $e$  ที่เปลี่ยนไป ดังนั้น เมื่อ  $T_i$  มีค่าน้อยผลของ Integral Action จะมากกว่าเมื่อ  $T_i$  มีค่ามาก



รูป 1.11 ผลตอบสนองของ Reset Action

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### Proportional - Integral (PI)

การควบคุมของหน่วยควบคุมแบบ PI นั้นสามารถจะแสดงได้ด้วยสมการดังต่อไปนี้

$$m(t) = K_p * [ e(t) + (1/T_i) * \int e(t) dt ]$$

หรือ

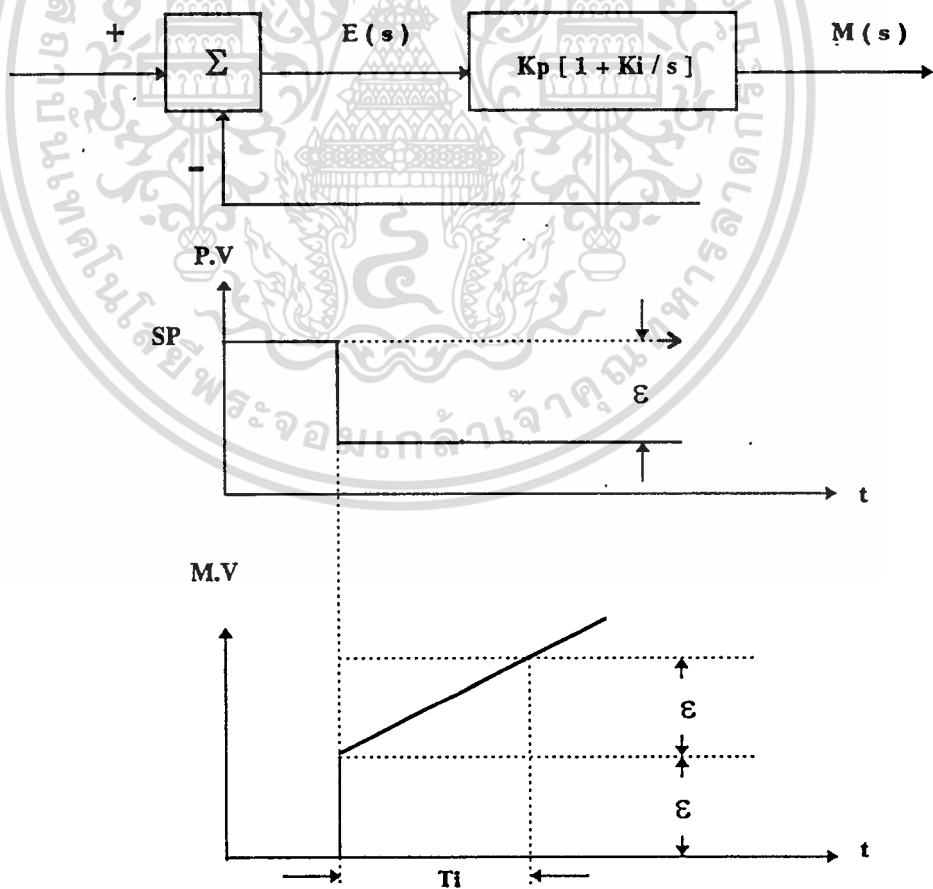
$$\begin{aligned} M(s)/E(s) &= K_p * [ 1 + 1/(T_i * s) ] \\ &= K_p * [ 1 + K_i/s ] \end{aligned}$$

โดยที่

$K_p$  เป็นค่าของ Gain

$T_i$  เป็นค่าของ Integral Time

ทั้งนี้  $K_p$  และ  $T_i$  เป็นค่าที่ปรับ  $T_i$  นั้นจะเป็นการปรับระยะเวลาการควบคุมแบบ Integral นั้นจะมีผลต่อส่วน Proportional รูป 1.12 แสดงถึง Block Diagram ของหน่วยควบคุม PI และผลตอบสนองของ PI Action



รูป 1.12 Block Diagram & Response of PI Action

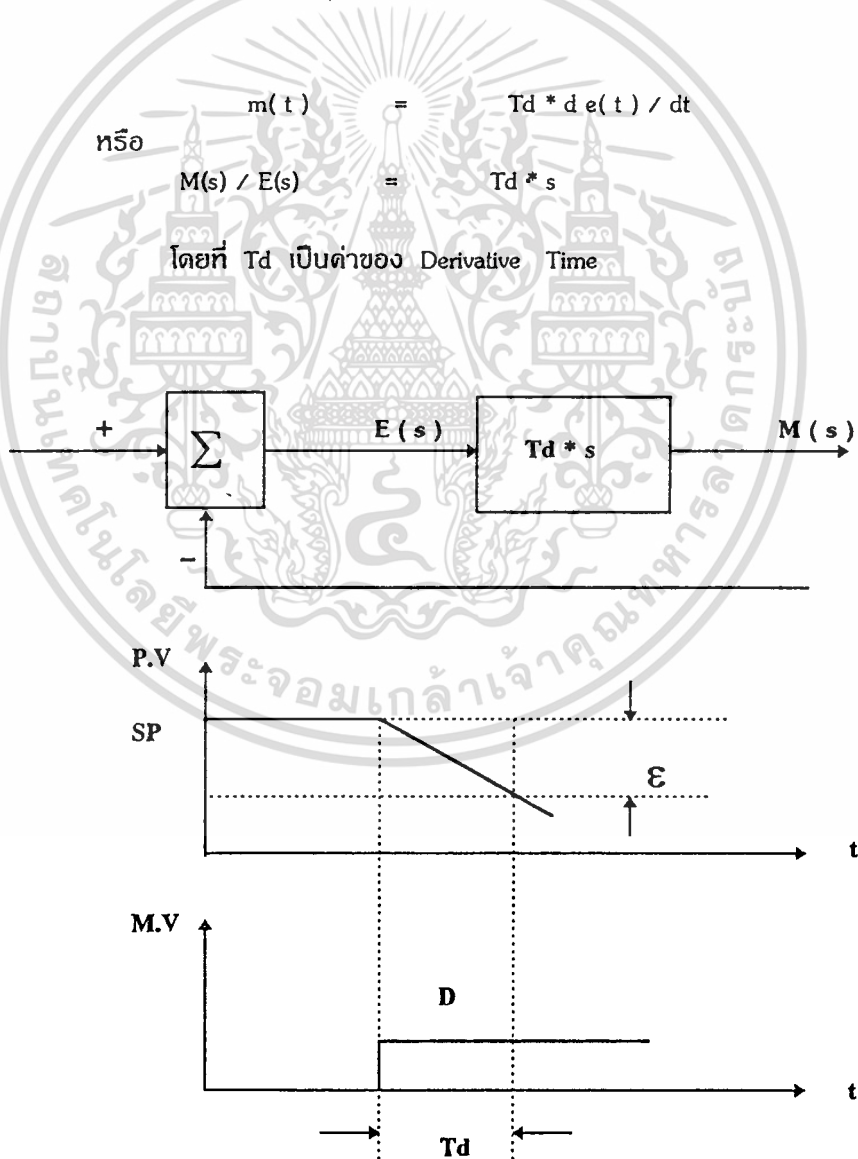
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### คุณสมบัติของ Integral Action

1. Off set ที่เกิดจาก P Action จะถูก Integral Action แก้จนหมดไป
2. เมื่อ  $T_i$  มีค่าน้อยลง ผลของ Integral Action จะมาก ทำให้เกิดการแกว่งขึ้นได้ ซึ่งจะก่อให้เกิดเสถียรภาพ

### Derivative Action (Rate Action)

การควบคุมแบบเหมาะกับขบวนการที่มี Time Lag มากๆ เพราะสามารถแก้ข้อผิดพลาดโดยการกระทำล่วงหน้าก่อนที่จะมีความผิดพลาดเกิดขึ้น ทั้งนี้เพราะสัญญาณควบคุมจะแปรตาม อัตราการเปลี่ยนแปลงของสัญญาณ ERROR และในบางครั้งจะเรียกกริยาการควบคุมแบบนี้ว่า Rate Action การควบคุมแบบ Derivative Action นี้ไม่สามารถนำไปใช้งานตามลำพังได้ เพราะว่า ERROR เป็น 0 หรือมีค่าคงที่ ค่าควบคุมจะไม่ให้ค่าเอาท์พุทเลย ดังนั้น Derivative Action จึงใช้รวมกับการควบคุม Proportional



รูปที่ 1.13 โดอะแกรมของ Derivative Action

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปจะเห็นว่า  $T_d$  จะเท่ากับช่วงเวลาที่ผลต่าง  $e$  แปรเปลี่ยนไปจนมีค่าเท่ากับค่าความชันของการเปลี่ยนนี้ ถ้า  $T_d$  มากจะทำให้ผลของ D Action มาก

### คุณสมบัติของ D Action

1. เหมาะสำหรับโปรเซสที่มี Time Lag มากๆ จะช่วยให้การควบคุมถึง จุดที่ต้องการเร็วขึ้น
2. ถ้า  $T_d$  มากไปผลของ D จะมากทำให้ระบบทั้งระบบไวขึ้น ชาดเสถียรภาพ
3. ไม่เหมาะสำหรับโปรเซสที่มี Time Lag น้อยๆ และ ตัวแปรโปรเซสเปลี่ยนแปลงได้ง่าย เช่น ระบบควบคุมการไหล ความดัน เป็นต้น

### Proportional - Derivative Action (PD)

การควบคุมของหน่วยควบคุมแบบ PD นี้จะอยู่ในรูปของสมการ

$$m(t) = K_c [e(t) + T_d \cdot de(t)/dt]$$

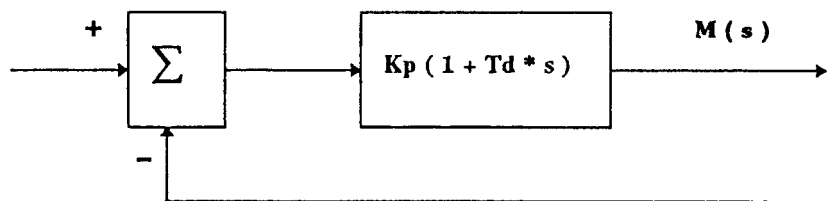
หรือ

$$M(s)/E(s) = K_c [1 + T_d \cdot s]$$

โดยที่  $K_c$  เป็นค่าของ Gain

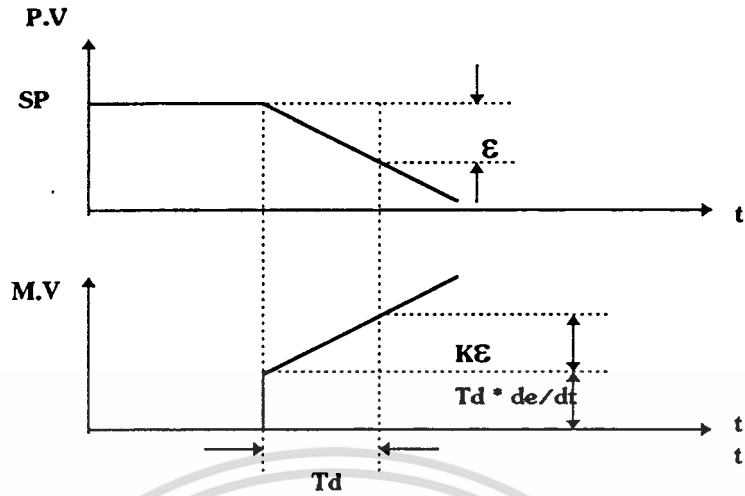
$T_d$  เป็นค่าของ Derivative Time

ค่าของ  $K_c$  และ  $T_d$  นั้นจะเป็นค่าที่สามารถปรับได้



รูปที่ 1.14 แสดงบล็อกไดอะแกรมของการควบคุมแบบ PD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.15 การควบคุมแบบ PD

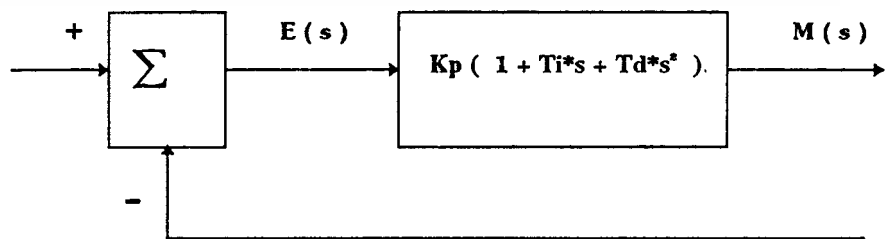
**Proportional - Integral - Derivative (PID)**

การควบคุมแบบ PID นั้นได้จากการรวมเอาการควบคุมแบบ Proportional ซึ่งอัตราขยาย การควบคุมแบบ Integral สามารถแก้ค่า Off Set และ Overshoot ที่จะทำให้เกิดการ Oscillate และกริยาการควบคุมแบบ Derivative จะทำให้เกิดผลตอบสนองได้อย่างรวดเร็วสมการของ PID มีรูปแบบดังต่อไปนี้

$$m(t) = K_c * [ e(t) + (1/T_i) * \int e(t) dt + T_d * de(t)/dt ]$$

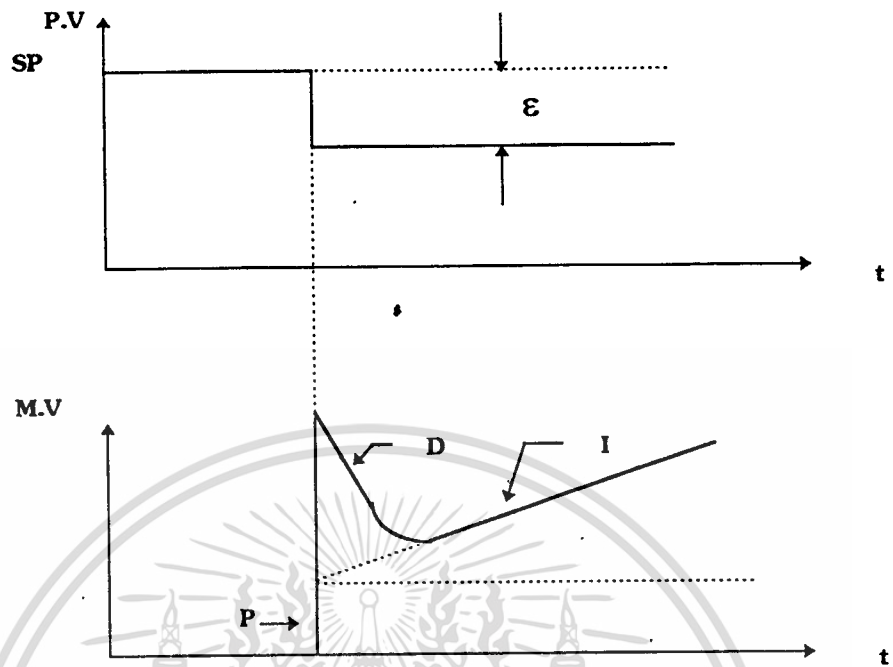
Transfer function คือ

$$\begin{aligned}
 m(s) &= K_p * [ 1 + I/(T_i * s) + T_d * s ] \\
 &= K_p * [ 1 + K_i/s + K_d * s ]
 \end{aligned}$$



**บล็อกไดอะแกรมของการควบคุมแบบ PID**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.16 การควบคุมแบบ PID

**PID CONTROLLER**

โดยการใช้สัญญาณเอาต์พุตที่คำนวณได้เป็นช่วงเวลาสั้น ๆ แทนที่จะเป็นเวลาต่อเนื่อง สมการ PID รูปทั่วไปคือ

$$m(t) = Kc [ e(t) + (1/Ti) \int e(t) dt + Td de(t)/dt ] + p(0)$$

$$e(t) = sv(t) - mv(t)$$

เราสามารถจะเขียนอยู่ในรูปของสมการความแตกต่างได้ดังต่อไปนี้

$$\int e(t) dt \cong \sum_{k=1}^n e_k \Delta t$$

$$de(t)/dt = [ e(n) - e(n-1) ] / \Delta t$$

$$m(n) = Kc [ e(n) + (1/Ti) \sum_{k=1}^n e_k \Delta t + Td \{ e(n) - e(n-1) \} / \Delta t ] + p(0)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### Velocity Algorithm

#### การเขียนโปรแกรมการคำนวณ

การควบคุมเนื่องจากสัญญาณความเบี่ยงเบน ( error ) การควบคุมที่เกิดขึ้นด้วยความแตกต่างของสัญญาณที่วัดได้จาก Process ( DV digital input variable ) กับค่าที่ต้องการ ( DSP digital set point ) เนื่องจากค่าที่วัดเข้ามาเป็นลักษณะของ digital to analog โดยที่ Microcontroller เป็น 8 bit จึงมีค่าตั้งแต่ 0 ถึง FF ค่า 0 จะมีความหมายถึงที่ 0% ของสัญญาณที่วัดได้จาก Process และ ที่ FF จะหมายถึงที่ 100% ของสัญญาณที่วัดได้จาก Process และที่ Set point ก็เช่นเดียวกัน 0 - FF หรือ 0 - 100%

ตัวอย่างที่ 40% ก็จะมีค่าในการควบคุมเท่ากับ  $[40/100]*255$

การควบคุมเนื่องจากสัญญาณความเบี่ยงเบน ( error ) เราสามารถเขียนสมการได้ดังนี้

$$DE = DV - DSP$$

DE = CONTROLLER VARIABLE ERROR

DV = DIGITAL INPUT FROM THE CONTROLLER VARIABLE

DSP = DIGITAL SET POINT

#### PROPORTIONAL MODE

เราสามารถเขียนเป็นสมการที่จะนำไปใช้ได้ดังนี้

$$DCP = K_p * DE + DCO$$

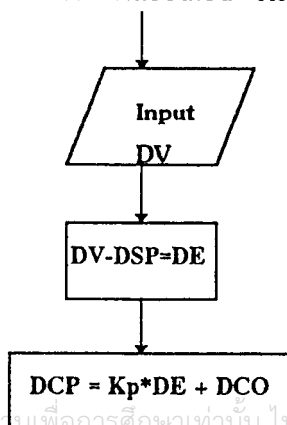
DCP = DIGITAL CONTROLLER OUTPUT

$K_p$  = DIGITAL PROPORTIONAL GAIN

DE = CONTROLLER VARIABLE ERROR

DCO = ZERO ERROR CONTROLLER OUT PUT

จากสมการเราสามารถเขียนเป็น Flow Chart ได้ดังต่อไปนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## INTEGRAL MODE

สมการในการคำนวณของ Integral Mode

$$DCI = [K_p/TI] * AREA$$

เมื่อ  $DCI = \text{integral - mode output}$   
 $K_p = \text{proportional mode gain}$   
 $TI = \text{integral time}$   
 $AREA = \text{error - time curve area approximation}$

$$DE = DV - DSP$$

$$SUM = SUM + DE$$

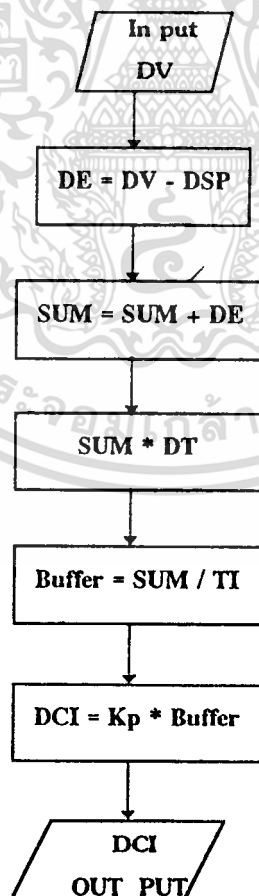
พื้นที่ที่หาได้

$$AREA = SUM * DT$$

จากสมการที่ผ่านทำให้เราได้

$$DCI = [K_p/TI] * SUM * DT$$

จากสมการข้างต้นเราสามารถอธิบายได้ด้วย Flow Chart ดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### DEVITIVE MODE

สมการในการคำนวณของ Devitive Mode

เมื่อ  
เนื่องจาก  
ดังนั้น  
เราจึงได้

$$DCD = K_p * TD * ( DE - DEO ) / DT$$

$$DDE = DE - DEO$$

$$DE = DV - DSP$$

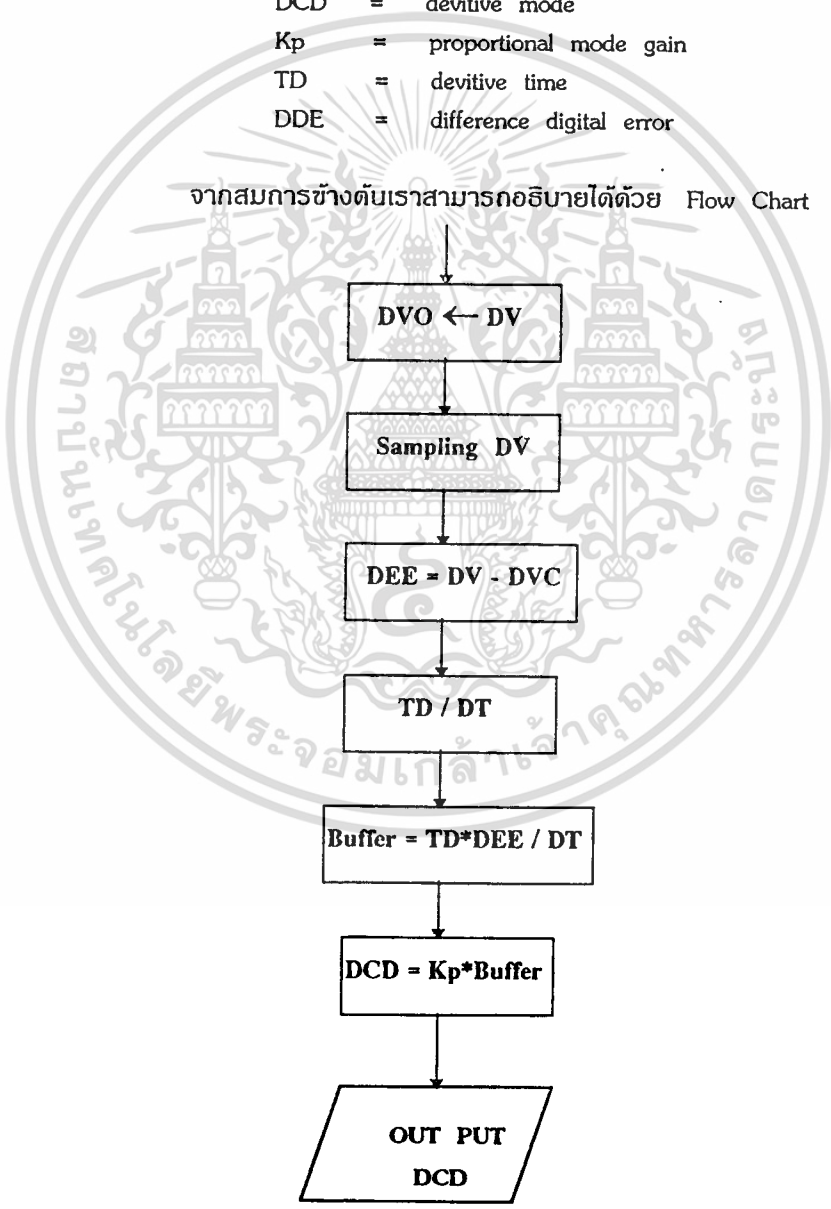
$$DDE = DV - DVO$$

$$DCD = K_p * TD * DDE / DT$$

เมื่อ

- DCD = devitive mode
- Kp = proportional mode gain
- TD = devitive time
- DDE = difference digital error

จากสมการข้างต้นเราสามารถอธิบายได้ด้วย Flow Chart ดังนี้



### การคำนวณ MODE

#### MODE\_PI

$$MV = DCD + DCI + BIAS$$

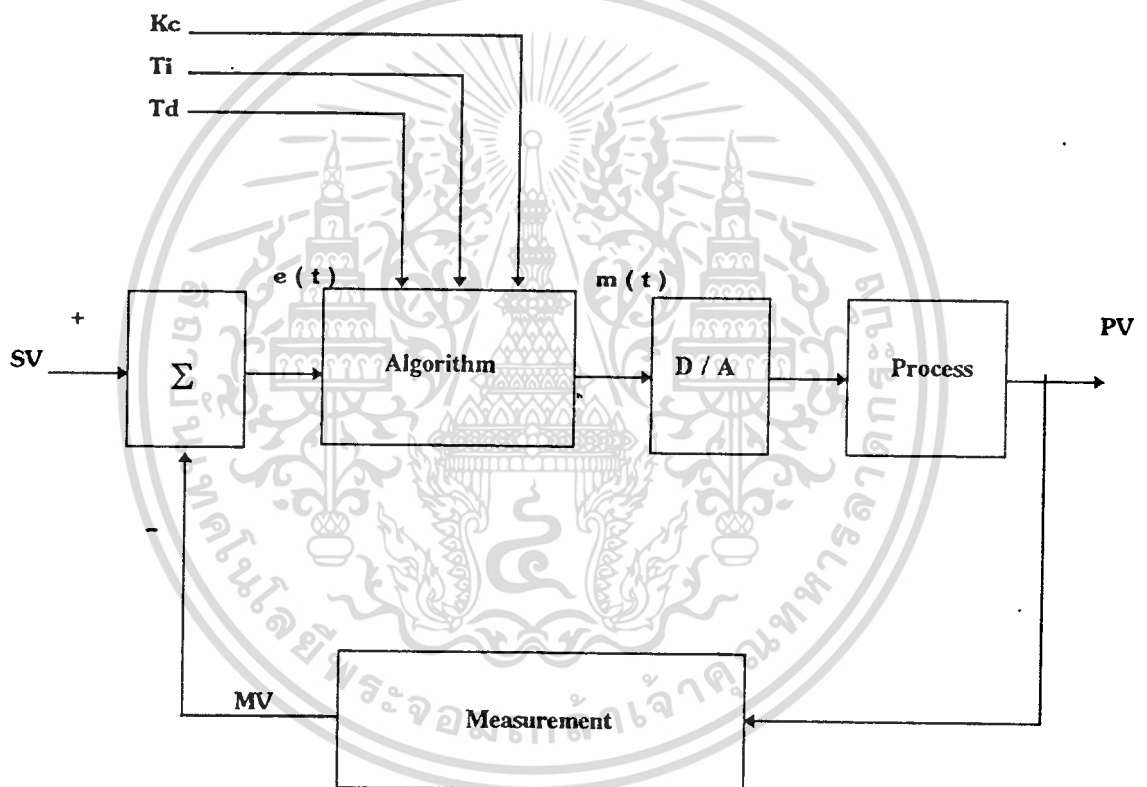
#### MODE\_PD

$$MV = DCP + DCD + BIAS$$

#### MODE\_PID

$$MV = DCP + DCI + DCD + BIAS$$

\* เมื่อ BIAS กำหนดให้มีค่าเท่ากับ 50 % \*



Block Diagram การคำนวณ PID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การปรับค่า PID

การปรับค่าพารามิเตอร์ของการควบคุมแบบ PID เป็นสิ่งจำเป็นสำหรับการควบคุมที่ดีที่สุด ในหัวข้อนี้จะได้นำวิธีการปรับค่า parameter ของการควบคุม PID เพื่อให้ได้ผลตอบสนองที่ดีที่สุด วิธีการปรับค่ามีหลายวิธี แต่ไม่มีวิธีไหนที่ยอมรับว่าเป็นวิธีที่ดีที่สุด กล่าวคือบางวิธีจะต้องพิจารณาผลตอบสนองของระบบและอาศัยประสบการณ์ในการปรับค่า ในขณะที่บางวิธีอาศัยคณิตศาสตร์ในการพิจารณา โดยทั่วไปแล้วการปรับค่า parameter ของการควบคุมแบบ PID จะพิจารณาจากเงื่อนไขใน Frequency Domain ซึ่งจะอยู่ในรูปของ Gain Margin & Phase Margin; เงื่อนไขใน Time Domain ที่นิยมใช้เพื่อการปรับมีดังต่อไปนี้

### - อัตราการเสื่อม 1/4 (Quarter Decay)

กล่าวคือ  $\text{Prak Overshoot}$  อันที่ 2 ต่อ  $\text{Peak Overshoot}$  อันแรก (Maximum Overshoot) ของผลตอบสนองของระบบที่มีการแกว่งจะต้องมีค่าเท่ากับ 1/4 รูปที่ 1.18 แสดงถึงเงื่อนไข

### - ดัชนีแสดงสถานะแบบ Integral of Square Error (ISE) ที่มีค่าน้อยที่สุด

ดัชนีแสดงสมรรถนะแบบนี้แสดงได้ด้วยสมการ

$$\text{ISE} = \int [e(t)]^2 dt = \text{น้อยที่สุด}$$

โดยที่  $e(t) =$  ค่าความเคลื่อน =  $sv(t) - mv(t)$

### - ดัชนีแสดงสมรรถนะแบบ Integral of Absolute Error (IAE) ที่มีค่าน้อยที่สุด

สมการของดัชนีแสดงสมรรถนะแบบนี้คือ

$$\text{IAE} = \int e(t) dt = \text{น้อยที่สุด}$$

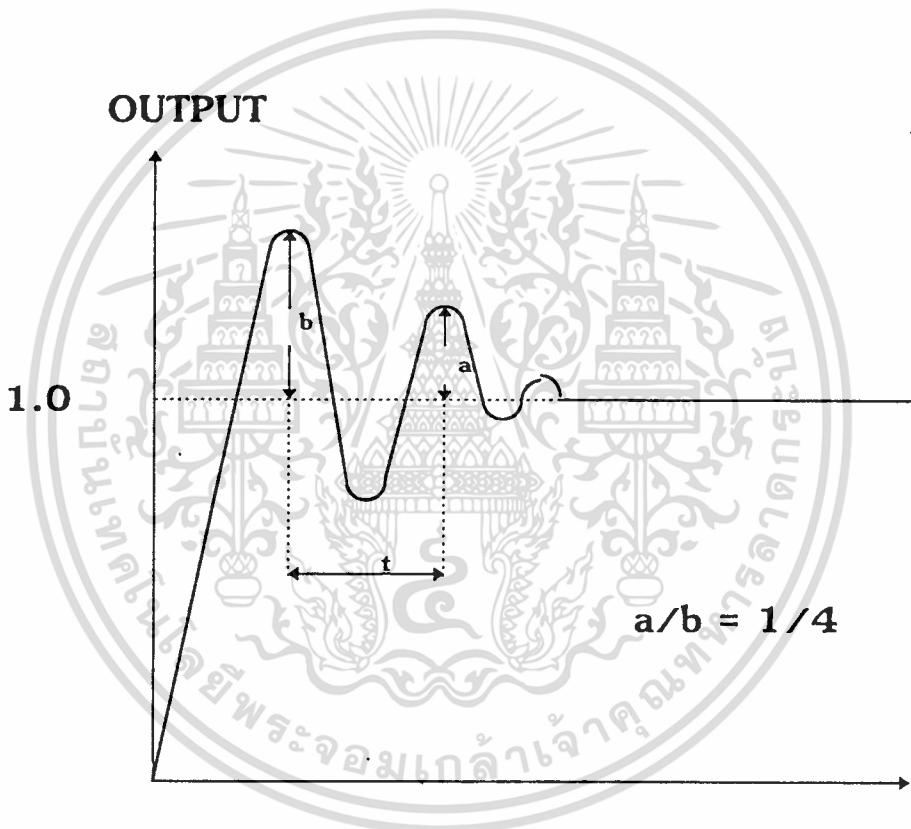
### - ดัชนีแสดงสมรรถนะแบบ Integral of Time Multiplied Absolute Error (ITAE)

ที่มีค่าน้อยที่สุด นั่นคือ

$$\text{ITAE} = \int t * e(t) dt = \text{น้อยที่สุด}$$

เงื่อนไขทั้ง 4 ที่กล่าวมานี้ แบบแรกเป็นแบบที่สะดวกเพราะสามารถวัดได้จากผลตอบสนองของระบบโดยตรง ส่วนแบบที่ 2 ถึง แบบที่ 4 มีข้อดีคือ จะมีความแม่นยำสูงกว่าแบบแรก กล่าวคือ parameter ต่างๆ ของการควบคุมแบบ PID อาจมีหลายค่าที่ทำให้ผลตอบสนองมีอัตราการเสื่อม 1/4 แต่จะมีเพียงค่าเดียวที่ทำให้เงื่อนไขมีค่าน้อยที่สุด

เงื่อนไขแบบอัตราเสื่อม 1/4 นั้นจะเป็นเงื่อนไขที่ดี ในการเข้ากันได้ระหว่าง Rise time ที่เร็วและ Seting Time ที่น้อย สำหรับเงื่อนไขที่ 2 ถึงแบบที่ 4 นั้น เนื่องจากว่าผลตอบสนองของระบบจะไม่เหมือนกัน ดังนั้นค่าของดัชนีแสดงสมรรถนะแต่ละแบบก็เปลี่ยนแปลงตามระบบด้วย แต่คุณลักษณะทั่วไปของดัชนีแต่ละระบบนั้นสามารถจะอธิบายได้ดังลักษณะต่อไปนี้คือ เงื่อนไขที่ 2 ซึ่งเป็นดัชนีแสดงสมรรถนะแบบ ISE นั้นถ้าความคลาดเคลื่อนมาก ค่าดัชนีแสดงสมรรถนะก็จะมาก ดังนั้นการทำให้ดัชนีแสดงสมรรถนะแบบนี้มีค่าน้อยที่สุดคือ ผลตอบสนองของระบบจะต้องมี Rise Time ที่น้อยมาก ซึ่งทำให้มีข้อเสียคือมี Overshoot ที่สูง สำหรับเงื่อนไขแบบ ITAE ซึ่งเป็นค่าของเวลาคูณกับค่าของความคลาดเคลื่อนนั้น ถ้าทำให้มีค่าน้อยที่สุดแล้ว ส่วนเงื่อนไขแบบอัตราเสื่อม 1/4



รูปที่ 1.18 ผลตอบสนองของระบบที่มีอัตราเสื่อม 1/4

## เทคนิคการปรับค่า PID

ได้มีนักคณิตศาสตร์คิดวิธีที่จะหาทางตั้งค่า PID เพื่อได้การควบคุมคุณภาพที่ดีที่สุดหลายวิธี ดังตัวอย่างที่แสดง ในตารางที่ 1.1 ด้านล่างอย่างไรก็ตามในทางปฏิบัติก็ยังคงไม่มีวิธีใดที่ดีพอที่จะสามารถจะนำไปใช้ในทุกระบบได้

| ชื่อผู้เสนอ               | แบบ | แบบการควบคุม   | Control Action PB (%)                         | Control Action Ti (min)            | Control Action Td (min)  | เกณฑ์             |
|---------------------------|-----|----------------|---|------------------------------------|--------------------------|-------------------|
| Zieg-ler                  | A.B | P<br>PI<br>PID | $2PB_{\mu}$<br>$2.2PB_{\mu}$<br>$1.7PB_{\mu}$ | -<br>$0.83P_{\mu}$<br>$0.5P_{\mu}$ | -<br>-<br>$0.125P_{\mu}$ | 25%<br>damp-ratio |
| Zieg-ler                  | A.B | P<br>PI<br>PID | $100kpL/T$<br>$110kpL/T$<br>$83kpL/T$         | -<br>33L<br>2L                     | -<br>-<br>0.5L           | 25%<br>damp-ratio |
| Taka-hashii               | A   | P<br>PI<br>PID | $110kpL/T$<br>$110kpL/T$<br>$77kpL/T$         | -<br>3.3L<br>2.2L                  | -<br>-<br>0.45L          | mininum           |
| Chien<br>Hrones<br>Rewick | A   | P<br>PI<br>PID | $333kpL/T$<br>$286kpL/T$<br>$167kpL/T$        | -<br>1.2L<br>T                     | -<br>-<br>0.5L           | no over<br>shoot  |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| ชื่อผู้เสนอ               | แบบ | แบบการควบคุม  | Control Action PB (%)                                    | Control Action Ti (min)                | Control Action Td (min)               | เกณฑ์                      |
|---------------------------|-----|---|--|--|---------------------------------------|----------------------------|
| Chien<br>Hrones<br>Rewick | B   | P<br>PI<br>PID                                      | 143kpL/T<br>167kpL/T<br>105kpL/T                         | -<br>T<br>0.5P $\mu$                   | -<br>-<br>0.125P $\mu$                | 20%<br>Over<br>shoot       |
| Chien<br>Hrones<br>Rewick | B   | P<br>PI<br>PID                                      | 333kpL/T<br>167kpL/T<br>105kpL/T                         | -<br>4L<br>2.4L                        | -<br>-<br>0.4L                        | no<br>over<br>shoot        |
| Chien<br>Hrones           | B   | P<br>PI<br>PID                                      | 143kpL/T<br>143kpL/T<br>83kpL/T                          | -<br>2.3L<br>2L                        | -<br>-<br>0.42L                       | 20%<br>over<br>shoot       |
| Fuji<br>Yoshi-<br>Kawa    | A   | P<br>PI L/T =1<br>L/T =1<br>PI $\mu$ L/T=1<br>L/T=1 | 100kpL/T<br>167kpL/T<br>250kpL/T<br>133kpL/T<br>200kpL/T | -<br>T+L<br>2L<br>0.5*( T+L )<br>0.25L | -<br>-<br>-<br>0.125*( T+L )<br>0.25L | mininum<br>control<br>area |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในทางปฏิบัติวิธีที่นิยมใช้ในการปรับค่า PID สำหรับการควบคุมเพื่อให้ได้  
1/4 Damping Ratio มี 3 วิธี คือ

1. Reaction Curve Method
2. Ultimate Sensitivity Method
3. Trial and error

### 1. Reaction Curve Method (Transient Response Method)

วิธีการ

1. ให้เปลี่ยนระบบควบคุมเป็นแบบลูปเปิด (Open Loop)
2. ให้ทำการหาคุณสมบัติของขบวนการ (Process Characteristic) โดยการเปลี่ยนค่าสัญญาณควบคุมไป  $\Delta PV$  แล้วบันทึกรูปคลื่นของตัวแปรกระบวนการ (Process Variable)
3. หา Process Gain ( $K_p$ ), Dead Time ( $LE$ ) และค่าคงเวลา ( $TE$ ) จาก Process Characteristic จากรูปที่ 1.19

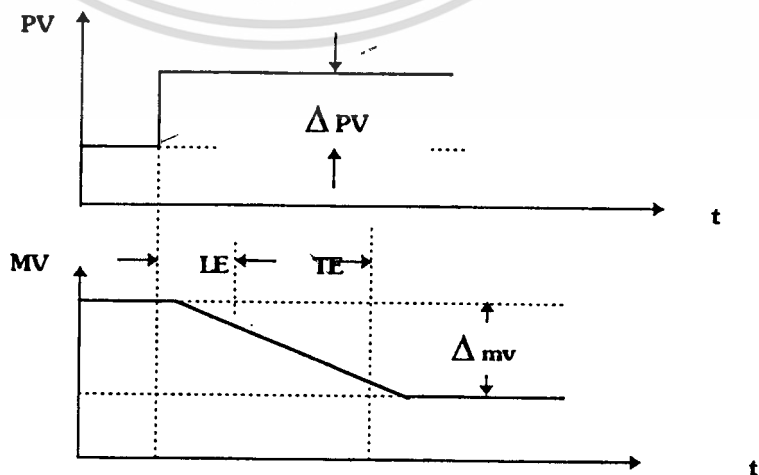
$$K_p = \Delta MV / \Delta PV$$

4. นำค่า  $K_p$ ,  $LE$ ,  $TE$  ที่หามาได้ไปคำนวณหา  $PB$ ,  $T_i$ ,  $T_d$  จากตาราง

1.2

ตาราง 1.2

| ชนิดของการควบคุม | PB %                    | $T_i$ (min) | $T_d$ (min) |
|------------------|-------------------------|-------------|-------------|
| P                | $100 K_p \cdot LE / TE$ | $\infty$    | 0           |
| PI               | $110 K_p \cdot LE / TE$ | $3.3 LE$    | 0           |
| PID              | $83 K_p \cdot LE / TE$  | $2.0 LE$    | $0.5 LE$    |



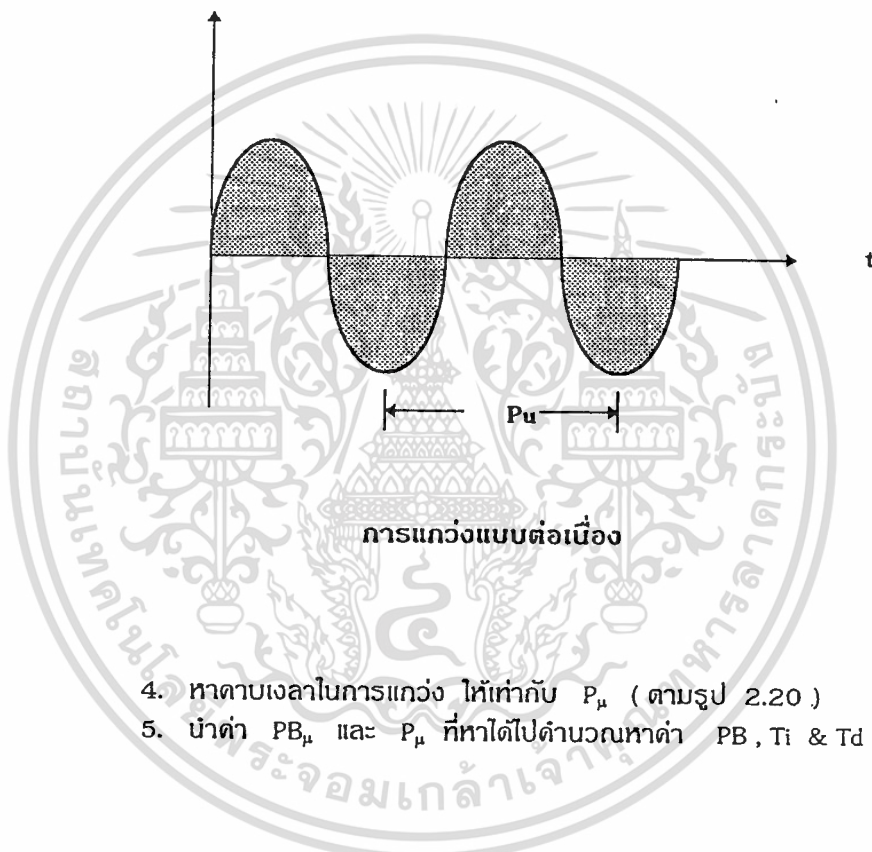
### Reaction Curve Method

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2. Ultimate Sensitivity Method

### วิธีการ

1. ให้ระบบควบคุมเป็นแบบลูปปิด
2. ตั้ง  $T_i$  สูงสุด และ  $T_d$  ต่ำสุด ใช้ P action ในการควบคุมอย่างเดียว
3. ครั้งแรกตั้งค่า PB ไว้ที่ค่าสูงสุด แล้วลดค่า PB ลงมา ลองเปลี่ยนค่าเป้าหมายเพื่อดูผลตอบสนองของลดค่า PB ให้ต่ำลงมาเรื่อยๆ จนถึงค่าที่เมื่อเปลี่ยนค่าเป้าหมายไปเล็กน้อย จะทำให้ขอบการเกิดการแกว่งต่อเนื่องไปตลอดค่า PB ในขณะนั้นเรียกว่า  $PB_\mu$  (Ultimate Proportional Band)



4. หาคาบเวลาในการแกว่ง ให้เท่ากับ  $P_\mu$  (ตามรูป 2.20)
5. นำค่า  $PB_\mu$  และ  $P_\mu$  ที่หาได้ไปคำนวณหาค่า  $PB$ ,  $T_i$  &  $T_d$  จาก

ตารางที่ 1.3

ตารางที่ 3.1

| ชนิดการควบคุม | PB (%)       | $T_i$ (min)  | $T_d$ (min)   |
|---------------|--------------|--------------|---------------|
| P             | $2 PB_\mu$   | $\infty$     | 0             |
| PI            | $2.2 PB_\mu$ | $0.83 P_\mu$ | 0             |
| PID           | $1.7 PB_\mu$ | $0.5 P_\mu$  | $0.125 P_\mu$ |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. Trial and error

เป็นวิธีการที่ใช้หลักการของ Ultimate Sensitivity Method เพียงแต่ว่าไม่ต้องคำนวณค่าและใช้การทดลองปรับค่าต่างๆเพื่อหาผลตอบแทนที่เร็วที่สุด

#### 1. P Control

- 1.1 ปรับตัวควบคุมไปที่ Manual Mode
- 1.2 ปรับ PB ไปสูงสุด  $T_i$  สูงสุด และ  $T_d$  ต่ำสุด
- 1.3 ปรับค่าเป้าหมาย (Set Point) ไปสู่ค่าที่ต้องการ
- 1.4 ปรับ Manual Control จนตัวแปรกระบวนการหรือค่าที่วัดได้เท่ากับค่าเป้าหมาย

กับค่าเป้าหมาย

- 1.5 ปรับตัวควบคุมไปที่ Automatic Mode
- 1.6 เปลี่ยนค่าเป้าหมายไปเล็กน้อย เมื่อค่าวัดเริ่มเปลี่ยนจึงลดค่าเป้าหมายกับมาอยู่ที่เดิม

หมายกับมาอยู่ที่เดิม

- 1.7 ลดค่า PB ลงมาและทำขั้น 6 ใหม่ โดยสังเกตผลตอบของค่า

วัด

- 1.8 ทำขั้น 6 และ 7 หลายๆครั้ง จนได้อัตราส่วนช่วงกว้างของ

การแกว่งของผลตอบเป็น  $1/4$  Damping Ratio

#### 2. PI Control

- 2.1 ทำเหมือน P Control จากขั้น 1 ถึง 8 เพื่อหาค่า PB ที่ดีที่สุด
- 2.2 ลดค่า  $T_i$  จน Offset หายไป
- 2.3 เพิ่มค่า  $T_i$  ถ้าเกิดการแกว่งขึ้น
- 2.4 ทำขั้น 2 และ 3 ซจนกว่าจะให้ผลตอบแทนเป็น  $1/4$  Damping Ratio

Damping Ratio

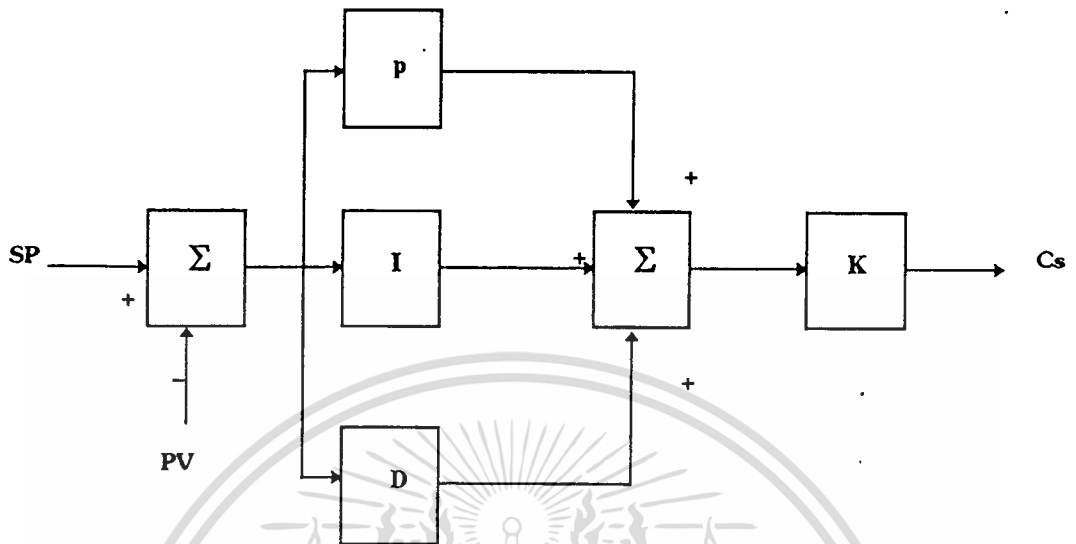
#### 3. PID Control

- 3.1 ทำเหมือน P Control จากขั้น 1 ถึง 2
- 3.2 ลดค่า PB ลงมาจนเกิดการแกว่ง
- 3.3 เพิ่ม  $T_d$  จนการแกว่งหยุด
- 3.4 ลดค่า PB จนเกิดการแกว่งใหม่อีก
- 3.5 ทำขั้น 2 ถึง 4 หลายๆครั้งจน  $T_d$  ไม่สามารถหยุดการแกว่งได้
- 3.6 เพิ่ม PB จนหยุดการแกว่ง
- 3.7 ตั้งค่า  $T_i$  ให้เท่ากับ  $T_d$  ค่าสุทธัย ( $T_i = T_d$  กรณีที่ PID Interfere Coefficient เท่ากับ 2)

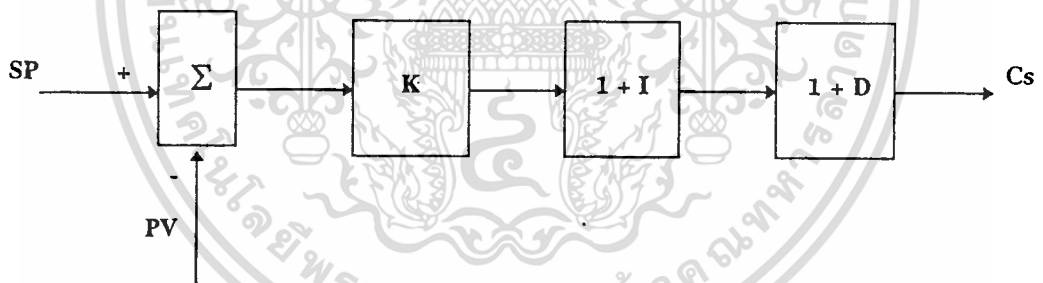
Interfere Coefficient เท่ากับ 2 )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การรบกวนกันและกันของค่า PID (PID Mutual Interference)



รูป A ตัวควบคุมแบบที่ดีที่สุด



รูป B ตัวควบคุมทั่วไป

แสดง Block Diagram ของตัวควบคุมแบบอุดมคติ ซึ่งค่า PID จะสามารถตั้งค่าได้อิสระไม่มีการรบกวนซึ่งกันและกัน ตัวควบคุมแบบนี้ก็มีราคาแพง โดยทั่วไปตัวควบคุมที่มีขายในท้องตลาดมักจะตัดแปลงวงจรให้ง่ายลง โดยมี Block Diagram ดังรูป B จะเห็นได้ว่า I & D Control ไม่ใช่ออย่างอุดมคติ และจะมีค่าการรบกวนซึ่งกันและกัน (Mutual Interference) ซึ่งได้แก่ การเปลี่ยนค่า  $T_i$  จะมีผลทำให้  $PB$  &  $T_d$  เปลี่ยนแปลงไปหรือถ้าเปลี่ยนค่า  $T_d$  ก็จะมีผลทำให้  $PB$  &  $T_i$  เปลี่ยนไปเช่นกัน

|                       | TEMPERATURE                | FLOW         | PRESSURE          | LEVEL             |
|-----------------------|----------------------------|--------------|-------------------|-------------------|
| <b>Characteristic</b> | 2 nd order ขึ้น<br>ไป      | แปรผันโดยตรง | 1 st order        | 1 st order        |
| <b>Response</b>       | ช้า , ปานกลาง              | เร็วมาก      | ปานกลาง , เร็ว    | ปานกลาง           |
| <b>Time Constant</b>  | ใหญ่ , ปานกลาง             | เล็ก         | เล็กมาก           | เล็ก , ปานกลาง    |
| <b>Dead Time</b>      | เล็ก , ปานกลาง             | เล็ก         | เล็ก              | เล็ก              |
| <b>Controler</b>      | on - off , P , PI ,<br>PID | P , PI       | on - off , P , PI | on - off , P , PI |

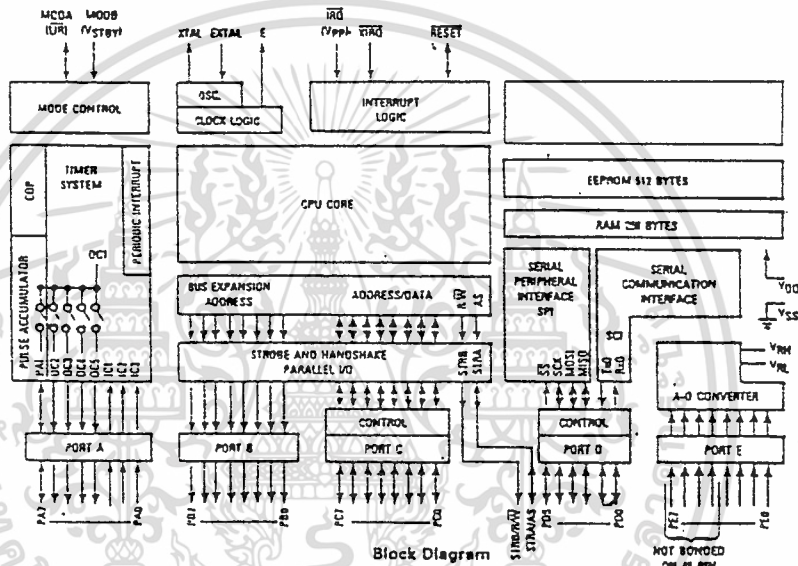
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไปว่ากรณีใดที่ขึ้น ลือจะขึ้นห่วยเป็นใช้คัดแปลงมีค่า และต้องระวังถึงถึงว่าของเอกสารหรือสิ่งที่มีกรบไปก็ได้

ไมโครคอนโทรลเลอร์ 68HC11

ในการออกแบบไมโครคอมพิวเตอร์ เพื่อใช้ในงานควบคุมหรืองานเฉพาะอย่าง โดยส่วนใหญ่ จะใช้ไอซีประเภท ไมโครคอนโทรลเลอร์ เพราะมีการทำงานที่รวดเร็วและเที่ยงตรงแม่นยำ นอกจากนี้ยังสามารถเปลี่ยนเงื่อนไขของงาน โดยเพียงแต่เปลี่ยนซอฟต์แวร์เท่านั้น เหตุผลอีกประการ หนึ่งที่นิยมใช้ไมโครคอนโทรลเลอร์คือ มีการรวมอุปกรณ์สนับสนุน ทางด้านอินพุทเอาต์พุทและอินเตอร์ เฟสต่างๆ ไว้อย่างพร้อมมูล

68HC11 เป็นไมโครคอนโทรลเลอร์อีกตัวหนึ่ง ของบริษัทโมโตโรล่า ที่ได้ถูกออกแบบมาโดยรวมอุปกรณ์ที่จำเป็นในการใช้งานไว้ภายในตัวมันเอง ได้แก่ อินพุท เอาต์พุท วงจรตั้งเวลา วงจรนับ วงจรอ่านสัญญาณแบบอนาลอก ระบบการอินเตอร์รัพแบบเวกเตอร์อินเตอร์รัพ และระบบป้องกัน ความผิดพลาดที่เกิดจากโปรแกรม รูปที่ 1 เป็นบล็อกไดอะแกรมการทำงานของ 68HC11



รูปที่ 1 บล็อกไดอะแกรมแสดงโครงสร้างของ 68HC11

คุณสมบัติของ 68HC11

- เป็นซีพียูขนาด 8 บิต
- มีหน่วยความจำภายในสำหรับเก็บโปรแกรมชนิด รอมขนาด 4,8,12 กิโลไบต์
- มีหน่วยความจำภายในสำหรับเก็บข้อมูลชนิดอีอีพรอม 512ไบต์หรือ 2 กิโลไบต์
- มีหน่วยความจำภายในสำหรับเก็บข้อมูลชนิดแรม 192,256 หรือ 512ไบต์
- มีวงจรตั้งเวลา/วงจรนับขนาด 8 บิต
- มีวงจรนับพัลส์ขนาด 8 บิต
- มีวงจรรับส่งข้อมูลได้สองทิศทาง ( UNIVERSAL SYNCHRONOUS RECEIVER TRANSMITTER :

USRT)

- ขนาดที่ใช้ในการติดต่อสำหรับวงจรการส่งข้อมูลอนุกรมเป็นแบบมัลติโปรเซสเซอร์
- มีวงจรเปลี่ยนสัญญาณจากอนาลอกเป็นดิจิตอล 8 ช่อง

- มีวงจรรีลไทม์อินเตอร์รัพต์
- มีระบบวอตช์ด็อกที่ตั้งเวลาได้
- มีวงจรตรวจสอบสัญญาณนาฬิกาให้กับซีพียู
- มีระบบอินเตอร์รัพ 2 ระดับจาก 21 แหล่ง
- สามารถติดต่อกับหน่วยความจำภายนอกได้ 64 กิโลไบต์

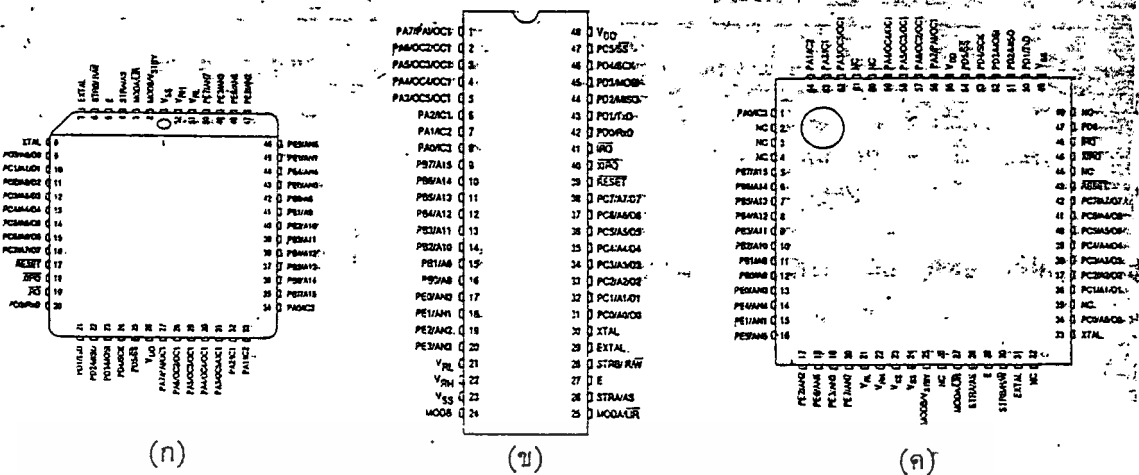
#### ตระกูลของ 68HC11

68HC11 เป็นไมโครคอนโทรลเลอร์ที่แบ่งออกเป็นหลายรุ่น โดยแบ่งตามลักษณะของหน่วยความจำที่อยู่ภายในซีพียูซึ่งแสดงตาม ตารางที่ 1 รูปแบบตัวถังของชิพตระกูลนี้ มีด้วยกัน 3 แบบ คือ PLCC (PLASTIC LEADED CHIP CARRIER) เป็นตัวถังรูปสี่เหลี่ยม 52 ขา , แบบ DIP ( DUAL IN LINE PACKAGE ) มีขนาด 18 ขา ในตัวถังแบบนี้จะไม่มีขาสัญญาณของพอร์ต PE4-PE7 และ แบบสุดท้ายคือ QFP (QUAT FIAT PACK) เป็นตัวถังรูปสี่เหลี่ยมขนาด 64 ขาตาม รูปที่ 2

| เบอร์ชิพ   | รอม | อีพรอม | แรม | รีจิสเตอร์ CONFIG | หมายเหตุ   |
|------------|-----|--------|-----|-------------------|--|
| MC68HC11A8 | 8K  | 512    | 256 | \$OF              | เป็นตัวมาตรฐานของชิพตระกูลนี้                                  |
| MC68HC11A1 | 0   | 512    | 256 | \$OD              | เหมือน MC68HC11A8 แต่ไม่มีรอม                                  |
| MC68HC11A0 | 0   | 0      | 256 | \$OC              | เหมือน MC68HC11A8 แต่ไม่มีรอมและอีพรอม                         |
| XC68HC11B8 | 8k  | 512    | 256 | \$OF              | เป็นรุ่นทดลองรุ่นแรก   |
| XC68HC11B1 | 0   | 512    | 256 | \$CD              | เหมือน XC68HC11B8 แต่ไม่มีรอม                                  |
| XC68HC11B0 | 0   | 0      | 256 | \$OC              | เหมือน XC68HC11B8 แต่ไม่มีทั้ง ROM และ RAM                     |
| MC68HC11E9 | 12k | 512    | 512 | \$OF              | มี 4 แดปเจอร์อินพุต แรมขนาดใหญ่ และรอม 12 กิโลไบต์             |
| MC68HC11E1 | 0   | 512    | 512 | \$OD              | เหมือน MC68HC11E9 แต่ไม่มีรอม                                  |
| MC68HC11E0 | 0   | 0      | 512 | \$OC              | เหมือน MC68HC11E9 แต่ไม่มีทั้งรอมและอีพรอม                     |
| MC68HC11E2 | 0   | 2k     | 256 | \$FF              | ไม่สามารถเพิ่มรอมได้   |
| MC68HC11D3 | 4k  | 0      | 192 | N/A               | เป็นรุ่นประหยัดไม่มีเทกูติดคอนเวอร์เตอร์และหน่วยความจำขนาดเล็ก |

ตารางที่ 1 แสดงตระกูลของ 68HC11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 เป็นตัวถังและการจัดขา (ก)ตัวถังแบบ PLCC (ข)ตัวถังแบบ DIP (ค)ตัวถังแบบ QFP

### โหมดการทำงาน

68HC11 มีโหมดการทำงาน 4 โหมด สามารถกำหนดที่ขาสัญญาณ MODA และ MODB โดยมีลักษณะการทำงานดังนี้

1. **SINGLE CHIP OPERATING MODE** โหมดนี้โปรแกรมควบคุมการทำงานจะอยู่ภายในตัว ซีพียู และขาสัญญาณต่าง ๆ สามารถใช้เป็นที่อินพุตและเอาต์พุต
2. **EXPANDED MULTIPLEXED OPERATING MODE** การทำงานในโหมดนี้โปรแกรมควบคุม การทำงานจะอยู่ภายนอก โดยใช้พอร์ทอินพุตเอาต์พุตเป็นแอดเดรสและข้อมูล ในการติดต่อกับหน่วยความจำภายนอกและสัญญาณควบคุมต่าง ๆ
3. **SPECIAL BOOTSTRAP OPERATING MODE** การทำงานในโหมดนี้จะมีลักษณะคล้ายกับSINGLE CHIP OPERATING MODE คือมีโปรแกรมควบคุมการทำงานอยู่ภายในซีพียู แต่ตำแหน่งในการรับและเวกเตอร์อินเตอร์รัพ จะต่างกันและมีลักษณะการทำงานที่พิเศษกว่า SINGLE CHIP MODE โดยสามารถป้องกันการตัดลอกหรือเลียนแบบได้
4. **SPECIAL TEST OPERATING MODE** โหมดนี้จะใช้ในการทดสอบการทำงานต่าง ๆ ตาม โรงงานที่ผลิตเป็นจำนวนมาก โดยมีการทำงานคล้ายกับ EXPANDED MULTIPLEXED OPERATING MODE

### ลักษณะของสัญญาณและการจัดขา

**RESET** เป็นขาที่ทำงานแบบสองทิศทางคือเป็นอินพุตสำหรับการทำให้ซีพียูเริ่มต้นการทำงาน และเป็นเอาต์พุตแสดงถึงการทำงานภายในที่ผิดพลาด ซึ่งเกิดขึ้นได้ 3 กรณีคือ ความผิดพลาดจาก สัญญาณนาฬิกา ความผิดพลาดจากโปรแกรมในการใช้ระบบวอตซ์ ดีคกและการเอ็ทคิวต์ต่อไปได้ผิดพลาด  
**XTAL,EXTAL** เป็นขาอินพุตสำหรับต่อคริสตอล เพื่อผลิตสัญญาณนาฬิกาให้กับซีพียู หรือจะใช้ สัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นาฬิกาจากภายนอก ป้อนเข้าที่ขา EXTAL โดยตรงก็ได้ โดยมีตัวต้านทาน 10-100 กิโลโอห์มต่อลงกราวด์ เพื่อป้องกันสัญญาณรบกวน

**E** เป็นเอาต์พุต โดยจะมีความถี่สัญญาณนาฬิกา น้อยกว่าความถี่ของสัญญาณนาฬิกาที่ขา XTAL และ EXTAL 4 เท่า และยังเป็นขาที่แสดงการทำงานของซีพียูด้วย คือ ถ้ามีสัญญาณที่ ลอจิก "0" แสดงว่าซีพียูกำลังแสดงผลภายใน และหากเป็นลอจิก "1" หมายถึงขณะนี้ซีพียูกำลังกับข้อมูล เมื่ออยู่ใน STOP โหมด จะไม่มีสัญญาณ E ออกมา

**IRQ** เป็นขาอินพุตทำงานที่ลอจิก "0" หรือขอบขาลง เป็นการขออินเตอร์รัพต์ซีพียู โดยที่ขา นี้ จะต้องต่อตัวต้านทานภายนอก ค่า 4.7 กิโลโอห์มต่อกับไฟเลี้ยง +5 โวลต์เพื่อให้สภาวะปกติ เป็นลอจิก "1"

**XIRQ** เป็นขาอินพุตทำงานที่ ลอจิก "0" เป็นการขออินเตอร์รัพต์แบบนอนมาสเดเบิ้ลอินเตอร์รัพต์ MODA, MODB เป็นขาที่เลือกใช้โหมดการทำงานของซีพียู ซึ่งแสดงการเลือกโหมดดัง ตารางที่ 2 หลังจาก ที่ ซีพียูเลือกโหมดการทำงานแล้ว ขา MODA จะเป็นขาเอาต์พุตลอจิก "0" เพื่อ แสดงถึงการเฟตช์ของซีพียู ที่จะใช้ในการทำให้ซีพียูทำงานที่ละคำสั่ง และขา MODB จะเป็นขา อินพุตสำหรับ ป้อนไฟเพื่อไปเลี้ยงหน่วยความจำภายใน เพื่อไม่ให้ข้อมูลสูญหาย หลังจากที่ไม่มีไฟ เลี้ยงซีพียู

| MODB | MODA | Mode Selected     |
|------|------|-------------------|
| 1    | 0    | Single Chip       |
| 1    | 1    | Expanded          |
| 0    | 0    | Special Bootstrap |
| 0    | 1    | Special Test      |

ตารางที่ 2 แสดงลักษณะการเลือกโหมดการทำงาน

**VRL, VRH** เป็นขาอินพุตสำหรับป้อนแรงดันอ้างอิงในวงจรแปลงอะนาลอก เป็นดิจิตอล

**STRB/R/W** เป็นขาเอาต์พุตแสดงผล การทำงานในโหมด SINGLE CHIP ขา นี้จะเป็นเอาต์พุตแสดง การแอนด์เซตข้อมูล และเมื่อทำงานในโหมด EXPANDED MULTIPLEXED ขา นี้จะเป็นเอาต์พุตแสดงการเขียน ของซีพียู โดยถ้าเป็นลอจิก "0" หมายถึงการเขียนข้อมูล

**STRA/AS** เป็นทั้งขาอินพุตและเอาต์พุต ถ้าทำงานในโหมด SINGLE CHIP ขา นี้จะเป็นเอาต์พุตในการ แอนด์เซตข้อมูล และถ้าทำงานในโหมด EXPANDED MULTIPLEXED จะเป็นเอาต์ ในการแยกสัญญาณระหว่าง แอดเดรส 8 บิตล่างกับสัญญาณข้อมูล

**PORT A** เป็นพอร์ตอินพุตเอาต์พุต โดยบิตที่ 0,1,2 จะถูกกำหนดเป็นอินพุต และบิตที่ 4,5,6 เป็น เอาต์พุต ส่วนบิตที่ 7 สามารถกำหนดเป็นอินพุตหรือเอาต์พุตก็ได้

**PORT B** เป็นพอร์ตเอาต์พุตเมื่อทำงานในโหมด SINGLE SHIP และจะเป็นแอดเดรส เมื่อทำงานในโหมด EXPANDED MULTIPLEXED

**PORT C** เป็นพอร์ตอินพุตเอาต์พุต เมื่อทำงานในโหมด SINGLE SHIP และเมื่อทำงานในโหมด EXPANDED MULTIPLEXED พอร์ตนี้จะเป็นทั้งสายสัญญาณแอดเดรสและข้อมูล โดยมีสัญญาณ AS เป็นตัวแยกแยะระหว่าง แอดเดรสและข้อมูล ซึ่งต้องใช้งานร่วมกับสัญญาณ R/W

**PORT D** เป็นพอร์ตอินพุตเอาต์พุต ที่สามารถกำหนดได้ว่าจะให้ปิดไต่เป็นอินพุตหรือเอาต์พุต และยังใช้เป็นพอร์ตในการส่งและรับข้อมูลแบบอนุกรมได้อีกด้วย

**PORT E** เป็นพอร์ตอินพุตสำหรับใช้งานทั่วไป และยังใช้เป็นพอร์ตอินพุตสำหรับวงจรเปลี่ยนสัญญาณอนาลอกเป็นดิจิทัลด้วย

| Port-Bit | Single-Chip and Bootstrapped Mode | Expanded Multiplexed and Special Test Mode |
|----------|-----------------------------------|--|
| A-0      | P0 IC3                            | PA0 IC3                                    |
| A-1      | P1 IC2                            | PA1 IC2                                    |
| A-2      | P2 IC1                            | PA2 IC1                                    |
| A-3      | P3 OC5 and-or OC1                 | PA3 OC5 and-or OC1                         |
| A-4      | P4 OC4 and-or OC1                 | PA4 OC4 and-or OC1                         |
| A-5      | P5 OC3 and-or OC1                 | PA5 OC3 and-or OC1                         |
| A-6      | P6 OC2 and-or OC1                 | PA6 OC2 and-or OC1                         |
| A-7      | P7 PA1 and-or OC1                 | PA7 PA1 and-or OC1                         |
| B-0      | P80                               | A8   |
| B-1      | P81                               | A9   |
| B-2      | P82                               | A10  |
| B-3      | P83                               | A11  |
| B-4      | P84                               | A12  |
| B-5      | P85                               | A13  |
| B-6      | P86                               | A14  |
| B-7      | P87                               | A15  |
| C-0      | P00                               | A0 D0                                      |
| C-1      | P01                               | A1 D1                                      |
| C-2      | P02                               | A2 D2                                      |
| C-3      | P03                               | A3 D3                                      |
| C-4      | P04                               | A4 D4                                      |
| C-5      | P05                               | A5 D5                                      |
| C-6      | P06                               | A6 D6                                      |
| C-7      | P07                               | A7 D7                                      |
| D-0      | P00 RxD                           | PD0 RxD                                    |
| D-1      | P01 TxD                           | PD1 TxD                                    |
| D-2      | P02 MISO                          | PD2 MISO                                   |
| D-3      | P03 MOSI                          | PD3 MOSI                                   |
| D-4      | P04 SCK                           | PD4 SCK                                    |
| D-5      | P05 SS                            | PD5 SS                                     |
|          | STPA                              | AS   |
|          | STRB                              | R/W  |
| E-0      | PE0 AN0                           | PE0 AN0                                    |
| E-1      | PE1 AN1                           | PE1 AN1                                    |
| E-2      | PE3 AN2                           | PE2 AN2                                    |
| E-3      | PE3 AN3                           | PE3 AN3                                    |
| E-4      | PE4 AN4                           | PE4 AN4                                    |
| E-5      | PE5 AN5                           | PE5 AN5                                    |
| E-6      | PE6 AN6                           | PE6 AN6                                    |
| E-7      | PE7 AN7                           | PE7 AN7                                    |

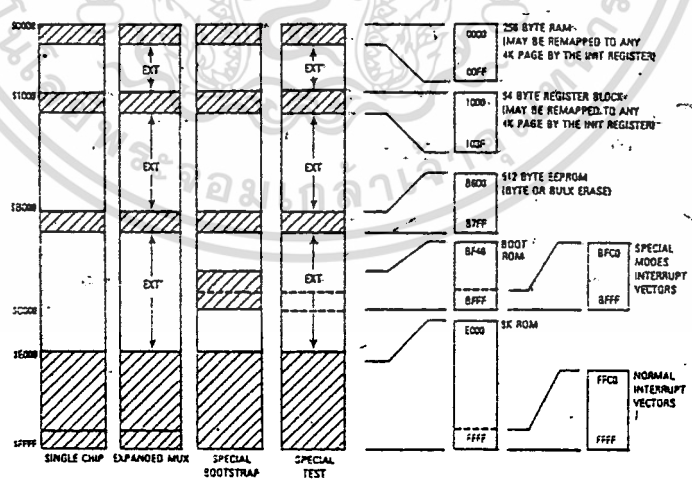
ตารางที่ 3 สรุปการใช้ขาของ 68HC11 ในโหมดการทำงานต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### การจัดหน่วยความจำ

68HC11 เป็นชิพที่มีการจัดหน่วยความจำอย่างมีประสิทธิภาพ โดยสามารถติดต่อกับหน่วยความจำได้ 64 กิโลไบต์ รวมทั้งหน่วยความจำที่อยู่ภายในชิพด้วย การจัดหน่วยความจำแบ่งออกเป็น 4 แบบ ตามโหมดการทำงานดังรูปที่ 3 พื้นที่หน่วยความจำ

- แอดเดรส 0000H - 00FFH เป็นพื้นที่ ของหน่วยความจำแรมที่อยู่ภายในชิพของทุกโหมดการทำงาน
- แอดเดรส 0100H - 0FFFH เป็นพื้นที่ หน่วยความจำภายนอกในโหมดการทำงาน EXPANDED MUX กับ SPECIAL TEST
- แอดเดรส 1000H - 103FH เป็นพื้นที่ของรีจิสเตอร์ภายในชิพของทุกโหมดการทำงาน
- แอดเดรส 1040H - B5FFH เป็นพื้นที่หน่วยความจำภายนอกใน EXPANDED MUX กับ SPECIAL TEST ถัดมาที่
- แอดเดรส B600H - B7FFH เป็นพื้นที่หน่วยความจำอีพรอมภายในชิพ
- แอดเดรส BF40H - BFFFH เป็นพื้นที่บูตรอมใน โหมด SPECIAL BOOT STRAP
- แอดเดรส B800H - DFFFH เป็นพื้นที่สำหรับหน่วยความจำภายนอก และที่
- แอดเดรส FFC0H - FFFFH เป็นพื้นที่สำหรับเก็บเวกเตอร์อินเตอร์รัพต์ ของอินเตอร์รัพท์ทุกประเภท



รูปที่ 3 แสดงการจัดหน่วยความจำในโหมดต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## หน่วยความจำแรมภายในซีพียู

หน่วยความจำแรมภายในซีพียู แบ่งออกเป็น 2 ประเภทคือ แรมที่ใช้ในการอ่านเขียนข้อมูล และ รีจิสเตอร์โดยตั้งแต่แอดเดรส 0000H - 00FFH (256ไบต์) เป็นหน่วยความจำที่ใช้ในการอ่านเขียนข้อมูลทั่วไป ซึ่งหน่วยความจำบริเวณนี้จะสามารถเก็บข้อมูลได้ แม้ไฟเลี้ยงของวงจรจะหายไป ข้อมูลแรมส่วนนี้ในสภาวะปกติจะได้ไฟเลี้ยงจาก VDD แต่เมื่อ VDD หายไปจะรับไฟเลี้ยงจากอีกทางหนึ่ง โดยก่อนที่ VDD จะหายไปจะต้องมีสัญญาณรีเซ็ตเพื่อทำให้ซีพียูหยุดการทำงาน เพื่อป้องกันไม่ให้มีการเขียนข้อมูลลงในหน่วยความจำ และเมื่อมี VDD เข้ามาก็จะไปจ่ายให้แรมในส่วนในส่วนนี้แทนไฟเลี้ยงอีกชุดหนึ่งได้ทันที โดยที่ข้อมูลที่อยู่ในแรมจะไม่เปลี่ยนแปลง กระแสที่ใช้ในการเลี้ยงแรมนี้ ใช้กระแสไฟน้อยมาก จึงสามารถใช้แบตเตอรี่เล็ก ๆ เป็นไฟเลี้ยงให้กับแรมนี้ได้ ส่วนแรมอีกส่วนหนึ่งที่เป็นรีจิสเตอร์ จะถูกใช้ในการควบคุม การทำงานต่างๆของซีพียู โดยมีหน้าที่ที่ แตกต่างกันไปดังนี้

|      |              |   |
|------|--------------|---|
|      | <b>DDRD</b>  | เป็นรีจิสเตอร์ ควบคุมทิศทางการติดต่อของ พอร์ต D   |
|      | <b>PORTE</b> | เป็นรีจิสเตอร์พอร์ต E ใช้สำหรับอ่านข้อมูลของ พอร์ต E  |
|      | <b>CFORC</b> | เป็นรีจิสเตอร์บอกผลของการเปรียบเทียบ ของไทเมอร์ต่าง ๆ   |
| OC1  | <b>OC1M</b>  | เป็นรีจิสเตอร์ที่ใช้สำหรับกำหนดบิตของ ข้อมูลใน พอร์ต A ที่มีการเปรียบเทียบกับ                                       |
|      | <b>OC1D</b>  | เป็นรีจิสเตอร์สำหรับเก็บข้อมูลของการเปรียบเทียบระหว่าง พอร์ต A กับ OC1  |
| TOC1 | <b>OC1M</b>  | เป็นรีจิสเตอร์สำหรับการแสดงผลการเปรียบเทียบบิตต่อบิตระหว่าง พอร์ต A กับรีจิสเตอร์                                   |
|      | <b>TCNT</b>  | เป็นรีจิสเตอร์ 16 บิตใช้เป็นตัวฟรีเด้าน์เตอร์ของระบบไทเมอร์ โดยจะทำการนับตั้งแต่ 0000 - FFFF แล้วกลับมาเริ่มต้นใหม่ |
|      | <b>TIC1</b>  | เป็นรีจิสเตอร์ 16 บิต เป็นอินพุตไทเมอร์ 1   |
|      | <b>TIC2</b>  | เป็นรีจิสเตอร์ 16 บิต เป็นอินพุตไทเมอร์ 2   |
|      | <b>TIC3</b>  | เป็นรีจิสเตอร์ 16 บิต เป็นอินพุตไทเมอร์ 3   |
|      | <b>TOC1</b>  | เป็นรีจิสเตอร์ 16 บิต ใช้ในการเปรียบเทียบกับไทเมอร์เด้าน์เตอร์  |
|      | <b>TOC2</b>  | เป็นรีจิสเตอร์ 16 บิต ใช้ในการเปรียบเทียบกับไทเมอร์เด้าน์เตอร์  |

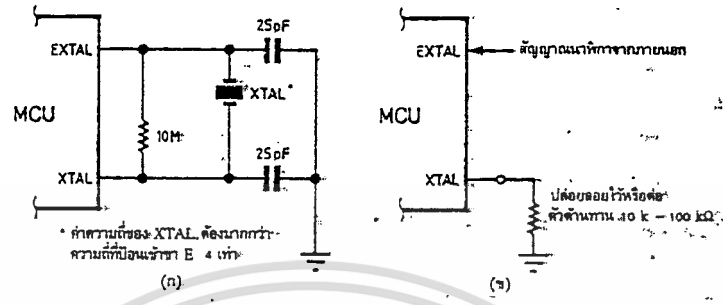
- TOC3** เป็นรีจิสเตอร์ 16 บิต ใช้ในการเปรียบเทียบกับไทมเมอร์เดาน์เตอร์
- TOC4** รีจิสเตอร์ 16 บิต ใช้ในการเปรียบเทียบกับไทมเมอร์เดาน์เตอร์
- TOC5** รีจิสเตอร์ 16 บิต ใช้ในการเปรียบเทียบกับไทมเมอร์เดาน์เตอร์
- TCTL1** เป็นรีจิสเตอร์ที่ใช้ในการเลือกลักษณะเอาต์พุตจากการใช้ฟังก์ชันการเปรียบเทียบ เอาต์พุต
- TCTL2** เป็นรีจิสเตอร์ที่ใช้ในการเลือกลักษณะของอินพุตเพื่อใช้ฟังก์ชันอินพุตแคปเจอร์ (INPUT CAPTURE) CAPTURE ฟังก์ชัน
- TMSK1** เป็นรีจิสเตอร์ที่ใช้สำหรับอินาบิลหรือดิสเอเบิล อินเตอร์รัพต์ของตัวเปรียบเทียบเอาต์พุตกับตัวเปรียบเทียบอินพุต
- TFLG1** เป็นรีจิสเตอร์ที่แสดงถึงการเท่ากับของไทมเมอร์เดาน์เตอร์กับตัวเปรียบเทียบเอาต์พุตกับตัวเปรียบเทียบอินพุต
- TMSK2** เป็นรีจิสเตอร์ที่ใช้ในการอินาบิลหรือดิสเอเบิล อินเตอร์รัพต์ของริลไทม์ และเป็นตัวกำหนดตัวหารของเวลาริลไทม์
- TFLG2** เป็นรีจิสเตอร์ที่ใช้แสดงการทำงานต่างๆ เป็นริลไทม์อินเตอร์รัพต์โอเวอร์โฟลว์
- PACTL** เป็นรีจิสเตอร์ที่ใช้ในการควบคุมพัลส์แอกติวเมเตอร์
- PACNT** เป็นรีจิสเตอร์เดาน์เตอร์ของพัลส์แอกติวเมเตอร์
- APCR** เป็นรีจิสเตอร์ที่ใช้ในการควบคุมการทำงานแบบ SPI (SERIAL PERIPHERAL INTERFACE)
- SPSR** เป็นรีจิสเตอร์ที่ใช้ในการบอกลักษณะการติดต่อแบบ SPI
- SPDR** เป็นรีจิสเตอร์ที่ใช้ในการรับข้อมูลและส่งข้อมูล
- BAC** เป็นรีจิสเตอร์ที่ใช้ในการกำหนดความเร็วในการส่งข้อมูลแบบ SCI (SERIAL COMMUNICATION INTERFACE)
- SCCR1** เป็นรีจิสเตอร์ที่ใช้ในการกำหนดรูปแบบในการส่งข้อมูลแบบ SCI SCCR2 เป็น รีจิสเตอร์ที่ใช้ในการควบคุมการอินเตอร์รัพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|               |   |
|---------------|---|
| <b>SCCR</b>   | เป็นรีจิสเตอร์ที่ใช้ในการบอกสถานะการรับส่งข้อมูลแบบ SCI   |
| <b>SCDR</b>   | เป็นรีจิสเตอร์ที่ใช้ในการเก็บข้อมูลจากการรับหรือส่งข้อมูลแบบ SCI  |
| <b>ADCTL</b>  | เป็นรีจิสเตอร์ที่ใช้เลือกการอ่านข้อมูลแบบอะนาล็อกและแสดงสถานะการอ่านข้อมูล  |
| <b>ADR1</b>   | เป็นรีจิสเตอร์ที่ใช้เก็บข้อมูลที่ได้จากวงจร A/D   |
| <b>ADR2</b>   | เป็นรีจิสเตอร์ที่ใช้เก็บข้อมูลที่ได้จากวงจร A/D   |
| <b>ADR3</b>   | เป็นรีจิสเตอร์ที่ใช้เก็บข้อมูลที่ได้จากวงจร A/D   |
| <b>ADR4</b>   | เป็นรีจิสเตอร์ที่ใช้เก็บข้อมูลที่ได้จากวงจร A/D   |
| <b>OPTION</b> | เป็นรีจิสเตอร์ที่ใช้ในการกำหนดเวลาของการใช้ระบบ COP (COMPUTER OPERATING PROPERTY) การเปิดสัญญาณสำหรับการแชมปลิงในวงจร A/D |
| <b>COPRST</b> | เปิดสัญญาณนาฬิกาเลือกกรีเซตค่าเวลาในระบบ COP  |
| <b>PPROG</b>  | เป็นรีจิสเตอร์ที่ใช้กำหนดรูปแบบของการติดต่อกับอีพ롬  |
| <b>HPIRO</b>  | เป็นรีจิสเตอร์ที่ใช้ในการจัดลำดับการอินเตอร์รัพ   |
| <b>INIT</b>   | เป็นรีจิสเตอร์ที่ใช้ในการกำหนดตำแหน่งของแรมและรีจิสเตอร์  |
| <b>TEST1</b>  | เป็นรีจิสเตอร์ที่ใช้ในการทดสอบของโรงงานผู้ผลิต  |
| <b>CONFIG</b> | เป็นรีจิสเตอร์ที่ใช้ในการบ่งบอกอุปกรณ์ภายในตัวชิพ   |

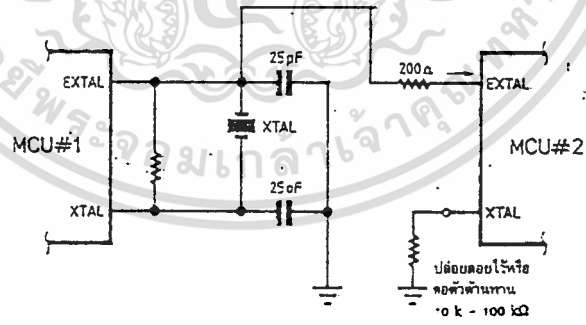
### วงจรกำเนิดสัญญาณนาฬิกา

68HC11 สามารถต่อสัญญาณนาฬิกาได้สองรูปแบบคือ ใช้วงจรกำเนิดสัญญาณนาฬิกาภายในโดยต่อคริสตอลภายนอกกับตัวต้านทานและตัวเก็บประจุอีกสองตัว ดังรูป 4 ก. และการต่อสัญญาณนาฬิกาจากภายนอกเข้าที่ขา EXTAL และต่อตัวต้านทานเข้าที่ขา XTAL ลงกราวด์ ดังรูปที่ 4 ข. นอกจากนั้น สัญญาณนาฬิกาจากการต่อ จาก EXTAL จากไมโครคอนโทรลเลอร์ตัวที่หนึ่งยังสามารถใช้เป็น สัญญาณนาฬิกา ให้กับไมโครคอนโทรลเลอร์ตัวต่อไปได้อีก ดังรูปที่ 5 โดยสัญญาณนาฬิกาที่ให้กับชิพ ต้องมีความถี่สูงกว่าสัญญาณที่ขา E ประมาณ 4 เท่า



รูปที่ 4 การต่อสัญญาณนาฬิกาให้กับ 68HC11

(ก)สัญญาณนาฬิกาภายใน  
(ข)สัญญาณนาฬิกาจากภายนอก



รูปที่ 5 การต่อสัญญาณนาฬิกาจากไมโครคอนโทรลเลอร์ตัวหนึ่งไปยังอีกตัว

หนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## รีจิสเตอร์

68HC11 มีรีจิสเตอร์ที่ใช้ในการเขียนโปรแกรมต่างๆ ทั้ง 8 บิตและ 16 บิตอยู่ด้วยกัน 7 ตัว ได้แก่ แอดเดรสเดอเรเตอร์ A และ B เป็นรีจิสเตอร์ขนาด 8 บิตที่ใช้ในการคำนวณทางด้านคณิตศาสตร์ และตรรกศาสตร์ นอกจากนั้นยังรวมกันเป็นรีจิสเตอร์ D ขนาด 8 บิต

โดย แอดเดรสเดอเรเตอร์ A เป็น 8 บิตสูง และ แอดเดรสเดอเรเตอร์ B เป็น 8 บิตต่ำ

อินเด็กซ์รีจิสเตอร์ IX เป็น รีจิสเตอร์ขนาด 16 บิต ใช้เป็นตัวชี้ตำแหน่งของข้อมูลและเป็นตัวตั้งในการบวกแบบ 16 บิตกับ แอดเดรสเดอเรเตอร์ B

อินเด็กซ์รีจิสเตอร์ IY เป็น รีจิสเตอร์ขนาด 16 บิต มีลักษณะคล้ายกับรีจิสเตอร์ IX แต่คำสั่งที่ใช้จะมีออปโค้ดขนาด 2 ไบต์

แอสดีดพอยเตอร์ SP เป็น รีจิสเตอร์ขนาด 16 บิต ชี้ ตำแหน่งของแอสดีดในการเก็บข้อมูลต่าง ๆ จาก การเรียกโปรแกรมย่อยหรือการอินเตอร์รัพ การเก็บข้อมูลลงสแต็คมีลักษณะเป็นแบบเข้าก่อนออกทีหลัง (FILO)

โปรแกรมเคาท์เตอร์ PC เป็น รีจิสเตอร์ขนาด 16 บิตใช้เป็นตัวชี้ตำแหน่งของคำสั่งต่อไปที่จะประมวลผล

คอนดิชันโค้ดรีจิสเตอร์ CCR เป็น รีจิสเตอร์ขนาด 8 บิตเป็นตัวบ่งบอกลักษณะการทำงานต่างๆของชิพ ีพียู เช่นการกำหนดให้อินาเบิลหรือดิสเอเบิลอินเตอร์รัพและการทำงานใน STOP โหมด

## วงจรถูกตี

เป็นวงจรถูกตีสัญญาณอะนาลอกให้เป็นดิจิตอลขนาด 8 บิต โดยสามารถรับอินพุตได้ 8 ช่อง จะทำการรับครั้งละช่อง โดยการโปรแกรมเลือกว่าจะทำการแปลงสัญญาณที่ช่องใด ผลลัพธ์ที่ได้จาก การแปลงจะเป็นอัตราส่วนระหว่างสัญญาณอินพุตกับแหล่งจ่ายที่เป็นไฟอ้างอิงของวงจรถูกตี A/D โดย แรงดันที่ใช้จะอยู่ในช่วง 2.5-5 โวลต์ เวลาที่ใช้ในการแปลงสัญญาณนั้นจะใช้จำนวนสัญญาณนาฬิกา 32 ลูกต่อหนึ่งครั้ง หรือที่เวลา 16 ไมโครวินาทีที่ความถี่ 8 เมกะเฮิรตซ์ ผลลัพธ์ที่อ่านเข้ามาได้ แล้วจะถูกเก็บอยู่ในรีจิสเตอร์ ADR1-ADR4 ดังในตารางที่ 5

| CD | CC | CB | CA | Channel Signal ---- | Result in ADRx if MULT=1 |
|----|----|----|----|---------------------|--------------------------|
| 0  | 0  | 0  | 0  | AN0                 | ADR1                     |
| 0  | 0  | 0  | 1  | AN1                 | ADR2                     |
| 0  | 0  | 1  | 0  | AN2                 | ADR3                     |
| 0  | 0  | 1  | 1  | AN3                 | ADR4                     |
| 0  | 1  | 0  | 0  | AN4                 | ADR1                     |
| 0  | 1  | 0  | 1  | AN5                 | ADR2                     |
| 0  | 1  | 1  | 0  | AN6                 | ADR3                     |
| 0  | 1  | 1  | 1  | AN7                 | ADR4                     |
| 1  | 0  | 0  | 0  | Reserved            | ADR1                     |
| 1  | 0  | 0  | 1  | Reserved            | ADR2                     |
| 1  | 0  | 1  | 0  | Reserved            | ADR3                     |
| 1  | 0  | 1  | 1  | Reserved            | ADR4                     |
| 1  | 1  | 0  | 0  | VRH Pin             | ADR1                     |
| 1  | 1  | 0  | 1  | VRL Pin             | ADR2                     |
| 1  | 1  | 1  | 0  | (VRH)/2             | ADR3                     |
| 1  | 1  | 1  | 1  | Reserved            | ADR4                     |

## ตารางที่ 5 การกำหนดช่องการทำงาน และการเก็บข้อมูลของวงจร A/D ใน 68HC11 ไทเมอร์

68HC11 มีไทเมอร์ขนาด 16 บิตที่ทำงานในลักษณะฟรีรันนิ่งคือจะทำการนับตั้งแต่ 0000-FFFF และกลับมาเริ่มต้นนับใหม่โดยจะใช้สัญญาณนาฬิกาจากภายในตัวชิพยู ไทเมอร์นี้จะนำไปใช้เป็นตัวเปรียบเทียบกับรีจิสเตอร์ไทเมอร์อินพุตและไทเมอร์เอาต์พุต

รีจิสเตอร์ไทเมอร์อินพุต(INPUT CAPTURE) เป็นรีจิสเตอร์ที่อ่านได้อย่างเดียวมีทั้งหมด 3 ตัว ถ้ามีสัญญาณลอจิก "0" ที่ขาอินพุตของแต่ละตัวหลังจากที่ได้ทำการโหลดค่าจากฟรีรันนิ่งไทเมอร์ แล้ว สามารถที่จะทำการอินเทอร์รัพท์ชิพยูได้ และสามารถนำค่าที่โหลดมา ใช้ในการวิเคราะห์คาบเวลาได้ว่า เป็นความถี่เท่าไร

รีจิสเตอร์ไทเมอร์เอาต์พุต(OUTPUT COMPARE) เป็นรีจิสเตอร์ที่อ่านและเขียนได้มีทั้งหมด 5 ตัว อยู่ในตำแหน่งตั้งแต่แอดเดรส 1016H-101FH เราสามารถเขียนค่าลงไปยังรีจิสเตอร์ไทเมอร์เอาต์พุต เพื่อจะนำไปเปรียบเทียบกับฟรีรันนิ่งไทเมอร์ เมื่อค่าทั้ง 2 เท่ากันก็จะทำการอินเทอร์รัพท์ชิพยู หรือจะทำการหารความถี่จากสัญญาณนาฬิกา ไปยังเอาต์พุตได้ทั้ง 5 ตัวซึ่งสามารถควบคุมโดยการกำหนดค่าใน รีจิสเตอร์ TMSK1 และ TCTL1

## รีลไมน์

เป็นวงจรถูกส่วนหนึ่งที่ทำหน้าที่ เป็นฐานเวลาให้กับโปรแกรมโดยสามารถโปรแกรมเลือกเวลาไปยังรีจิสเตอร์ PACTLที่บิต RTR0 และ RTR1 บิต RTIF ในรีจิสเตอร์ TFLG2 จะเป็นตัวกำหนดว่าจะให้ทำการอินเทอร์รัพท์หรือไม่ถ้าต้องการก็ควบคุมให้บิต RTIF มีค่าเท่ากับ "1"

## พัลส์แอกติวูเลเตอร์

68HC11 มีรีจิสเตอร์สำหรับนับพัลส์จากภายนอกขนาด 8 บิตโดยความถี่ที่สามารถนับได้สูงสุดจะเท่ากับความถี่ของสัญญาณนาฬิกาภายในซีพียูหารด้วย 64 ลักษณะของการนับจะเป็นแบบนับขึ้นและก้านับจน

เกินค่าของรีจิสเตอร์แล้วจะทำการอินเตอร์รัฟซีพียูได้ โดยการเซตที่บิต PAOVI ของรีจิสเตอร์TMSK2

## รีเซต

การรีเซตของ 68HC11 จะแบ่งออกเป็น 3 ชนิดคือ

- (1) การรีเซตจากภายนอก ซึ่งเกิดจากการเปิดเครื่องครั้งแรก หรือสัญญาณรีเซตจากภายนอก
- (2) การรีเซตจากวงจร วอตช์ดีอก
- (3) การรีเซตเนื่องจากการผิดพลาดของสัญญาณนาฬิกา

การรีเซตจากการเปิดเครื่องจะใช้จำนวนพัลส์สัญญาณนาฬิกา 4064 ลูก หลังจากสัญญาณนาฬิกา ลูกแรกทำงาน ซึ่งถ้าใช้คริสตอลภายนอกที่ 8 เมกะเฮิรตซ์ จะใช้เวลา 2 มิลลิวินาที ถ้าซีพียูทำงานในโหมด BOOT STRAP หรือ SPECIAL TEST หลังจากซีพียูรีเซต จะไปเฟตซ์ ข้อมูลที่เวกเตอร์แอดเดรส FFFCH-FFFFH หรือ BFFEh-BFFFFH ในกรณีการรีเซตจากวงจรวอตซ์ ดีอกจะทำให้สัญญาณที่รีเซตเป็นลอจิก"0" ด้วย และจะไปเฟตซ์ข้อมูลที่เวกเตอร์แอดเดรส FFFAH-FFFBH และในกรณีการรีเซตจากการผิดพลาดของสัญญาณนาฬิกาจะไปเฟตซ์ข้อมูลที่เวกเตอร์แอดเดรส FFFCH-FFFDH

## อินเตอร์รัฟต์

ลักษณะการอินเตอร์รัฟต์แบ่งออกเป็น 2 แบบคือฮาร์แวร์และซอฟต์แวร์ อินเตอร์รัฟต์ ฮาร์แวร์ อินเตอร์รัฟต์ยังแบ่งออกได้เป็นการอินเตอร์รัฟต์จากภายในและจากภายนอก สามารถจัดลำดับความสำคัญของการขออินเตอร์รัฟต์ ได้ด้วยการควบคุมรีจิสเตอร์ HPRIO การอินเตอร์รัฟต์จะทำงานเป็นแบบ เวกเตอร์อินเตอร์รัฟต์

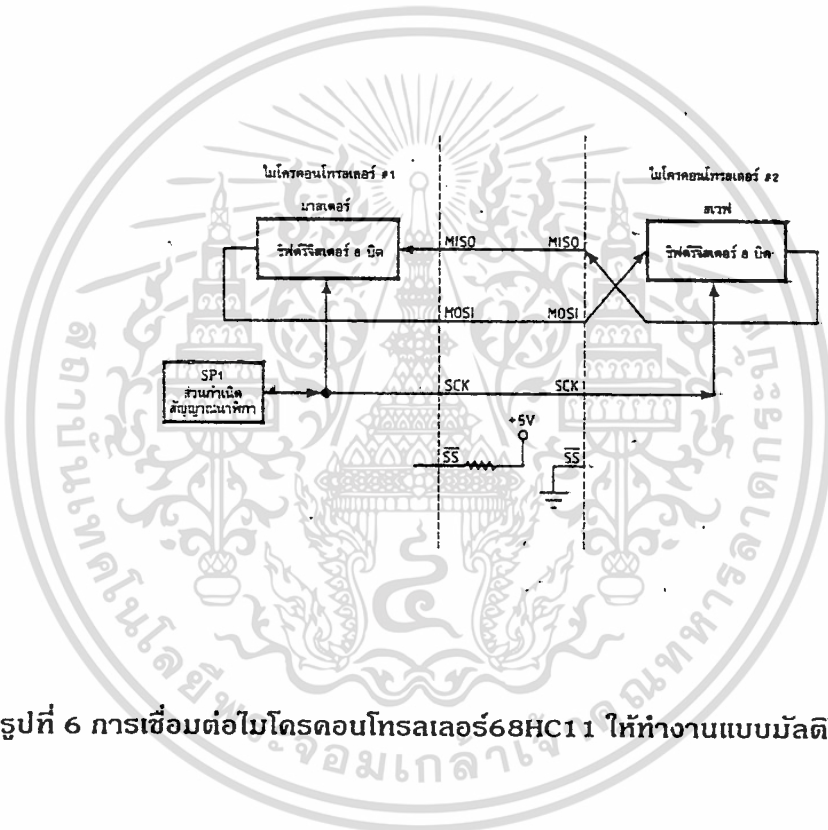
## พอร์ตอนุกรม

พอร์ตอนุกรมของ 68HC11 มีลักษณะเหมือนกับพอร์ตอนุกรมไมโครคอนโทรลเลอร์โดยทั่วไป การรับส่งข้อมูลเป็นแบบ ฟูลดูเพล็กซ์คือ สามารถรับส่งข้อมูลได้พร้อมกันในเวลาเดียวกัน ก่อนจะรับข้อมูลมายังรีจิสเตอร์จะต้อง อ่านข้อมูลเดิมออกไปก่อน มิฉะนั้นข้อมูลที่ส่งมาใหม่จะทับข้อมูลเดิม แอดเดรส ของรีจิสเตอร์อยู่ที่ตำแหน่ง 102FH การรับส่งข้อมูลสามารถที่จะกำหนดการรับส่งข้อมูลได้ว่าจะ เป็นแบบ 10 บิต หรือ 11 บิต แต่ลักษณะที่พิเศษของพอร์ตอนุกรมสำหรับ 68HC11 นี้คือ สามารถที่จะทำการตรวจ สอบ

สัญญาณรบกวนได้ โดยความกว้างของสัญญาณรบกวนจะต้องไม่เกิน 1/16 ของอัตราความเร็วในกําร รับ  
ส่ง

### มัลติโปรเซสเซอร์

นอกจากการติดต่อข้อมูลแบบพอร์ตอับพุต เอาต์พุต และอนุกรมแล้ว 68HC11 ยังมีการติดต่อแบบ  
มัลติโปรเซสเซอร์ ที่สามารถนำซีพียูมาต่อหลาย ๆ ตัวได้ โดยมีซีพียูตัวหนึ่งเป็นมาสเตอร์ และมีซีพียูตัว  
อื่น ๆ เป็นสเลฟ จะใช้สัญญาณที่ขา SS,SCK,MISO,MOSI เป็นตัวติดต่อดังแสดงในรูปที่ 6



รูปที่ 6 การเชื่อมต่อไมโครคอนโทรลเลอร์ 68HC11 ให้ทำงานแบบมัลติโปรเซสเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## V TO I CONVERTER CIRCUIT

เนื่องจากการส่งสัญญาณเป็นแรงดันมีปัญหาเกิดขึ้นมากมาย ซึ่งทำให้เกิด % ERROR ขึ้นได้กระแสนั้นมีบทบาทสำคัญมากในการส่งสัญญาณเป็นแบบ LOOP การเปลี่ยนสัญญาณเป็นกระแส และการส่งสัญญาณไปยังโหลด นั้น จะไม่มีการสูญเสียสัญญาณ เนื่องจากค่าความต้านทานของสายไฟ หรือจุดต่อที่ไม่ดีในการส่งสัญญาณให้กับกระบวนการมาตรฐานที่ใช้ในการส่งสัญญาณคือถ้าเป็นความดัน จะมีค่า 3 - 15 PSI , แรงดันไฟฟ้าที่ใช้ 0 - 5 VOLT และถ้าเป็นกระแสจะใช้ 4 - 20 mA ในการส่งสัญญาณ 0 - 100 % สาเหตุที่ 0% สัญญาณจะมีค่าเท่ากับ 0 V นั้นเพราะว่าจะได้ทำให้ทราบว่ามีสัญญาณควบคุมอยู่หรือไม่ โดยแสดงค่า 3 PSI , 0 VOLT และ 4 mA ให้เราได้ทราบ การเลือกวงจร I TO V จะขึ้นอยู่กับความต้านทานของโหลดและการต่อกราวด์ของโหลดแบบ FLOTATION LOAD จะช่วยลดสัญญาณรบกวนและอีกแบบหนึ่งคือ GROUND LOAD ซึ่งก็เหมาะกับการใช้ในการขับสัญญาณเพราะ ส่วนแสดงผลและอุปกรณ์อิเล็กทรอนิกส์ที่ใช้ในการควบคุมส่วนใหญ่จะออกมาในลักษณะเดียวกันแบบ GROUND LOAD

วงจรอย่างง่ายสำหรับ V/I ดังในรูปซึ่งเป็น NON INVERTING AMPLIFIER CIRCUIT การส่งสัญญาณและการควบคุม LOAD นี้ได้จากสัญญาณที่ขับจากส่วนป้อนกลับแบบลบซึ่งเราสามารถวิเคราะห์วงจรง่าย ๆ คือ OP-AMP เป็นตัวทำหน้าที่ต่อควบคุม LOOP และแรงดันที่ขา NON INVERTING จะปรากฏที่ขา NON INVERTING ด้วยแต่แรงดันนี้จะลดคร่อมความต้านทานอยู่ดังนั้นกระแสที่ไหลผ่าน R คือ I

$$I = e_{th} / R$$

วงจรถ่ายเปลี่ยนกระแสเป็นแรงดันอย่างง่าย

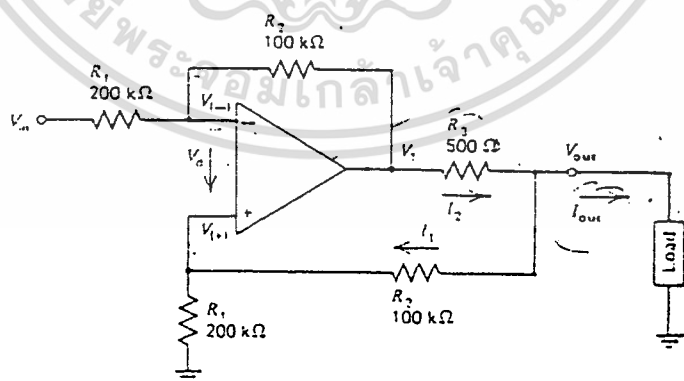


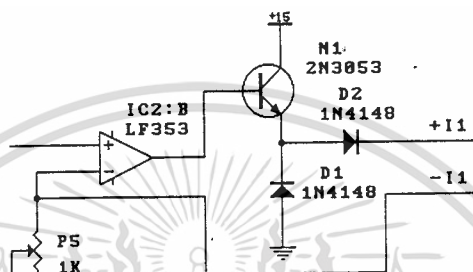
Figure 10.21 Voltage-controlled current source.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีข้อพิจารณาหลายข้อที่ต้องพิจารณาในการใช้วงจร V/I แบบโหลดลอย ดังวงจร ในรูป ค่าความต้านทานในรูปที่ใช้ในการส่ง  $R_{loop} = R_{wire} + R_{load}$  จะไม่ผลต่อกระแสในการส่ง แต่ แรงดันเอาต์พุทจะขึ้นอยู่กับ  $R_{loop}$

$$V_{out} = [1 + R_{loop}/R] * E_{in} < V_{sat}$$

โดยต้องพยายามทำให้  $R_{loop}$  มีค่ามากเกินไป จนทำให้เกิดการ Saturation ได้ OP-AMP ต้องสามารถจ่ายกระแสได้เพียงพอความต้านทาน เพราะมีการส่งสัญญาณอยู่หลายมาตรฐานที่ต้องใช้กระแส 20 - 60 mA ซึ่ง OP-AMP ทั่วไปไม่สามารถจ่ายได้จึงต้องใช้ TRANSISTOR ช่วยดังรูป



วงจร V/I แบบมีตัวขับกระแส

เนื่องจาก TRANSISTOR อยู่ใน LOOP การป้อนกลับแบบลบ OP-AMP จึงสามารถทำการชดเชย TRANSISTOR ในเรื่องการ BIAS, การ OFFSET & NON - LINEARITY ถ้าสัญญาณต้องเปลี่ยนไปถึงช่วงล่าง ซึ่งกระแสไหลกลับทิศทางเราต้องใส่ Q2 (PNP) ซึ่งเป็น COMPLEMENT ของ Q1 สำหรับจ่ายช่วงลบกระแสจากโหลด ต้องไหลกลับมาตามสายที่ต่อกลับ OP-AMP เพราะวงจรไม่สามารถขับกระแสต่อลงกราวด์ได้เลย ซึ่งสัญญาณที่ต้องใช้สาย 2 เส้นในการทำงานของวงจรซึ่งกระแสที่ไหลในสายทั้งสองมีทิศทางตรงกันข้ามกันเราสามารถให้ความแตกต่างหรือวงจรขยายของเครื่องมือวัดที่โหลด เพื่อสัญญาณรบกวนที่สายสัญญาณทั้ง 2 เส้นระหว่างการส่งสัญญาณ ควบคุม

ห้ามทำการเปิดโหลดเพราะถ้าเกิดโหลดออกมามจะทำให้มีการป้อนกลับแบบลบจะส่งผลให้ OP-AMP เกิดการ SATURATION โดยเราสามารถป้องกันได้โดยการ SHOT OUTPUT ซึ่งจะทำให้วงจรมีการทำงานเหมือน วงจรแรงดันสัญญาณกระแสก็จะไม่มีผลต่อวงจร NON INVERTING จะเป็นป้องกันแหล่งจ่ายแรงดันอินพุททุกโหลดดึงกระแส เราสามารถสร้าง วงจร I TO V โดยใช้วงจรแบบ INVERTING ได้

จากรูปในวงจรปัญหาที่เกิดขึ้นอีกประการหนึ่งก็คือเมื่อ

$$e_{in} = 0$$

$$I_1 = 0$$

ในลักษณะเดียวกันถ้าเปิดโหลดหรือการส่งสัญญาณของอุปกรณ์อิเล็กทรอนิกส์เกิดการผิดพลาดก็จะทำให้เกิด  $I_1 = 0$  เช่น

$$\text{กระแสตกเป็นศูนย์เมื่อ } E_{in} = 0$$

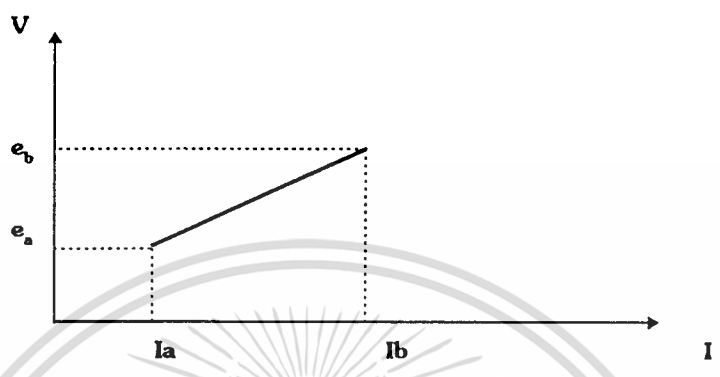
เราสามารถแก้ปัญหานี้ได้โดยการปรับ OFFSET ให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$e_{in} = 0 \text{ หรือ } e_{in} = \text{MINIMUM}$$

$$I_1 = I_{(1) > 0}$$

แรงดันอินพุตที่มีค่าแรงดันเกือบเป็นศูนย์ จะทำให้กระแสในลูปมีค่าไม่เท่ากับ ศูนย์วงจร OFFSET จะทำให้กระแสในลูปมีค่ามากหรือทำให้ค่าเป็น  $I(0)$  ถ้าวงจรเกิดการผิดพลาด



วงจรและเส้นกราฟแสดงความสัมพันธ์ของ V/I CONVERTER

วงจร V/I ที่มี OFFSET ดังแสดงในรูปโดยใช้วงจรขยาย NON - INVERTING แบบมี SUMMING แทนในวงจรรูปกระแสเอาท์พุทจะเปลี่ยนตามแรงดันอินพุต ( $e_{in}$ ) และแรงดันอ้างอิง ( $e_{ref}$ ) ค่าความต้านทานทั้งสองตัวจะป้องกันการไหลของแหล่งจ่ายไฟของแรงดันตัวอื่น

เส้นกราฟที่แสดงในรูปมีความเป็นเชิงเส้นดีคล้ายๆ กันที่อยู่ในควอเทอร์เนตที่กำหนดโดยให้อินพุต  $e_{(A)}$  ควบคุมเอาท์พุท  $I_{(A)}$  และอินพุต  $e_{(n)}$  ควบคุมเอาท์พุท  $I_{(o)}$  โดยเราสามารถแสดงค่าต่างๆ ในวงจรได้จาก

$$R = \frac{e(B) - e(A)}{2 [I_{(B)} - I_{(a)}]} \quad \text{และ} \quad E = 2RI(B) - e(B)$$

เราสามารถแก้สมการได้ตามนี้ เพราะว่า OP-AMP มีการป้อนกลับแบบลบ

$$V_x = V_r$$

$$I_1 = V_x / r$$

ผลรวมของลูปจาก  $e_{in}$  ถึง  $e_{ref}$  ถ้าเราไม่พิจารณา  $R \ 10 \ K\Omega$  เมื่อทำการเปรียบเทียบ

$$e_{in} - I_n(IM) - I_n(IM) - e_{ref} = 0$$

หาค่า  $I_n$  ได้จาก

$$I_n = (e_{in} - e_{ref}) / 2M$$

ผลรวมของจุดจาก  $e_{in}$  ถึง  $V_x$  ได้ดังนี้

$$e_{in} - I_{in} (1M) - V_x = 0$$

$$\begin{aligned} V_x &= e_{in} - (e_{in} - e_{ref}) * 1M/2M = (e_{in} + e_{ref}) / 2 \\ &= [e_{in} + e_{ref}] / 2 \end{aligned}$$

จากสมการด้านบนเราจะได้ว่า

$$I_L = (e_{in} - e_{ref}) / 2R$$

และที่จุด A เราจะได้

$$I_a = (e_a + e_{ref}) / 2R$$

และที่จุด B เราจะได้

$$I_b = (e_b + e_{ref}) / 2R$$

จากสมการที่จุดทั้งสอง (A, B)

$$2R * I_a = e_a + e_{ref}$$

$$2R * I_b = e_b + e_{ref}$$

หาค่า R ได้โดยการทำให้ค่า  $e_{ref}$  หายไป

$$2R (I_a - I_b) = e_b - e_a$$

เราสามารถหาค่า  $e_{ref}$  ได้จาก

$$e_{ref} = 2R I(B) - e(B)$$

เราสามารถคำนวณหาค่า R และค่า  $e_{ref}$  ของวงจรได้เมื่อกำหนดค่าตัวแปรดังนี้

$$e(A) = 0 \text{ V.} \quad I(A) = 4 \text{ mA}$$

$$e(B) = 5 \text{ V.} \quad I(B) = 20 \text{ mA}$$

$$R = \frac{e(B) - e(A)}{2 [ I(B) - I(A) ]}$$

$$R = \frac{5V - 0V}{2 [ 20mA - 4mA ]}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นเราจะได้อ่าความต้านทาน

$$R = 156.25 \Omega$$

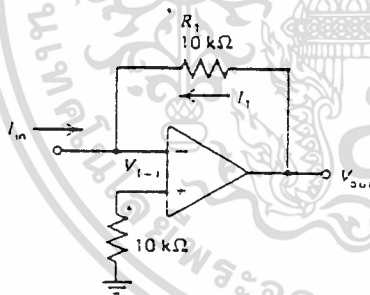
จากค่า R เราสามารถนำมาหาค่า  $e_{ref}$  ได้ดังนี้

$$\begin{aligned} e_{ref} &= 2RI(B) - e(B) \\ &= 2(156.25 \Omega)(20 \text{ mA}) - 5V \\ &= 1.25 V \end{aligned}$$

ในการปรับแต่งวงจร V/I ชั้นแรก เราต้องทำการปรับค่า R เพื่อการปรับอย่าง การวัดให้มีขนาดตามต้องการและปรับค่า  $e_{ref}$  เพื่อตั้งค่า (ZERO) ให้กับวงจรด้วย

### I / V CONVERTOR

ในการใช้สัญญาณแบบกระแสเช่นนี้เราจะต้องมีการเปลี่ยนสัญญาณกระแสกลับมาเป็น สัญญาณแรงดันซึ่งเราสามารถทำได้ง่าย ๆ โดยการใช้ความต้านทานมาต่อกับ OP-AMP สำหรับโหลดที่ ใช้ GROUND เป็นจุดอ้างอิงดังนี้



CIRCUITS USING THE OP AMP

Figure 10.22 Current-to-voltage converter.

### รูป I / V CIRCUIT แบบใช้กราวด์อ้างอิง

กระแสที่ใช้ในการส่งสัญญาณจะถูกเปลี่ยนเป็น สัญญาณแรงดันโดย R1 ซึ่งก็ยัง คงใช้งานไม่ได้ดีเท่าที่ไม่สามารถปรับค่า SPAN & ZERO ได้เพื่อที่ปรับค่าทั้งสองจึงได้ต่อ U2 & U3 เข้าไปในวงจร ส่วนวงจรตามแรงดัน U1 จะทำหน้าที่เป็นตัวกั้นระหว่าง R1 & Ri วงจรนี้มีปัญหา มากมาย ซึ่งต้องมีกราวด์สำหรับกระแสไหลกลับของระหว่างตัวส่งกับตัวรับสัญญาณ ค่าความต้าน ทานในสายที่ใช้ส่งจะทำให้เกิดแรงดันตกคร่อมในสาย ถึงแม้จะมีการปรับแต่งค่าแล้วก็ตามความแตกต่าง ของค่าความต้านทานที่เกิดขึ้นระหว่างวงจรโอเพิลทรอนิกส์ต้องการสภาวะแรงดันจากโหลดลอย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การส่งจะถูกเปลี่ยนไปเป็นแรงดันโดย  $R_{span}$ , OPAMP U1,  $R_f$ ,  $R_i$  จากวงจรรขยายความแตกต่าง ซึ่งมี  $R_{span}$  เป็นตัวกันให้กับแรงดันที่ถ่วงรวมอยู่ที่ถูกขยายด้วย  $R_f / R_i$  และต้องมีควารดอ้างอิงให้กับเอาต์พุตด้วย ในการป้องกันความผิดพลาดกับโหลดต้องใช้ค่า  $R_i \gg R_{span}$

การปรับค่า OFFSET ให้เป็นศูนย์สามารถปรับได้โดยใช้  $R_{POT}$  และ U2 เอาต์พุต U1 จะขึ้นอยู่กับกระแส  $I$ ,  $R_{SPAN}$  ของโหลดลอย อัตราขยายของ  $R_f / R_i$  และแรงดัน OFFSET 0 ของ  $V_x$

$$V_{OUT} = (R_f / R_i) I R_{SPAN} + V_x \quad (A)$$

$$V(a) = (R_f / R_i) * I(a) R_{SPAN} + V_x \quad (B)$$

ที่จุด b.

$$V(b) = (R_f / R_i) * I(b) R_{SPAN} + V_x \quad (C)$$

(B) - (C) :

$$\begin{aligned} V(a) - V(b) &= (R_f / R_i) * I(b) R_{SPAN} - (R_f / R_i) * I(a) R_{SPAN} \\ &= (R_f / R_i) * R_{SPAN} [ I(b) - I(a) ] \end{aligned} \quad (D)$$

ดังนั้น

$$R_{SPAN} = [ V(b) - V(a) ] / [ (R_f / R_i) * R_{SPAN} * [ I(b) - I(a) ] ] \quad (E)$$

จากสมการ (b)

$$V_x = V(a) - (R_f / R_i) * I(a) * R_{SPAN} ( I(b) - I(a) ) \quad (F)$$

ซึ่งเราสามารถคำนวณหาค่า  $R_{SPAN}$  กับ  $V_x$  ได้ดังนี้จากรูปที่

$$I(a) = 4mA \quad V(a) = 0V.$$

$$I(b) = 20mA \quad V(b) = 5V.$$

เราเลือกใช้  $R_f / R_i = 10$  คำนวณของค่า  $R_{SPAN}$  ได้จากสมการ

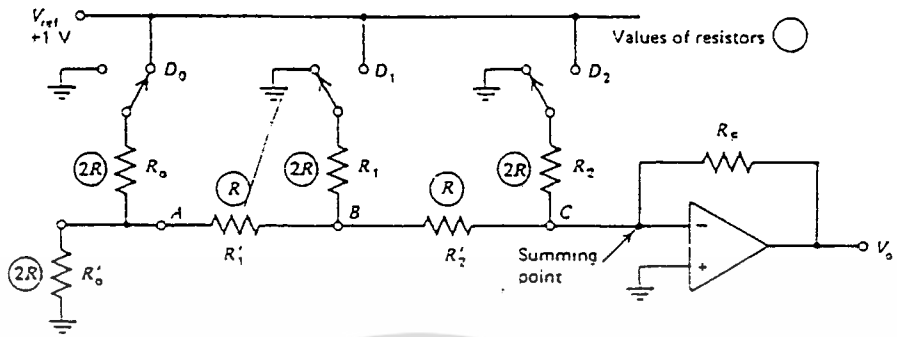
$$R_{SPAN} = \frac{5V - 0V}{10 (20mA - 4mA)} = 31.25 \Omega$$

เลือกใช้ R-TRIMPOT ขนาด  $100 \Omega$  ในการปรับค่า  $R_{SPAN}$  และเมื่อให้ค่า โดยกำหนด  $R_i \gg R$  ก็โอเค

หลักการของ D/A แบบง่าย ๆ คือ หลักการความต้านทานสองค่า โดยใช้ค่าความต้านทานเป็นอัตราส่วนที่ใช้คำนวณได้ง่ายขึ้น

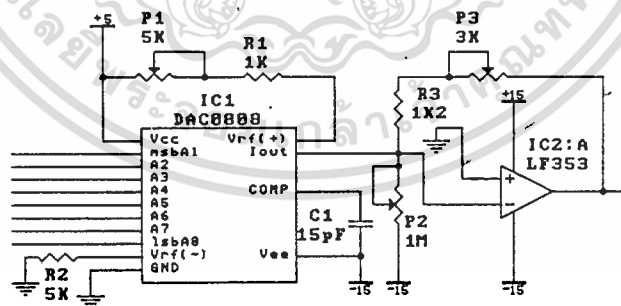
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## LINEAR INTEGRATED CIRCUITS



ใบโครงงานนี้ใช้วงจร D/A ขนาดของ 8 bit คือ DAC 0808 ซึ่งมีฟังก์การทำงานในรูป

แสดงฟังก์การทำงานของ DAC 0808



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ลักษณะคุณสมบัติของ D/A

ความละเอียดซึ่งขึ้นอยู่กับจำนวนบิตทางด้านอินพุท เช่น วงจรเปลี่ยนสัญญาณขนาด 8 บิตมีระดับเอาต์พุทอยู่ 256 ระดับดังนั้น ความละเอียดคือ 1 ใน 256 ความละเอียดบางครั้งจะคิดเป็น % คือ  $1/256$  มีค่าเท่ากับ 0.39% ความถูกต้อง เป็นค่าที่ได้จากการเปรียบเทียบระหว่างเอาต์พุทเต็มสเกล 5 VOLT มีค่าความถูกต้อง +19% ดังนั้นความผิดพลาดสูงสุดคือ  $0.0019 \times 5$  V หรือ 9.5 mV ในทฤษฎีแล้วความถูกต้องของวงจร D/A ไม่ควรต่ำกว่า  $+1/2$  ของค่าที่ LSB



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การทดลองและผลการทดลอง

ในการทดลองนี้เป็นการนำเอาเครื่องควบคุมที่ได้ออกแบบ มาต่อทดลองใช้กับขบวนการจำลอง ซึ่งเป็นลักษณะแบบ 2 ลูปหรือ 2 ขบวนการ

#### การทำงานของขบวนการจำลอง

ในการทดลองนี้เป็นการควบคุมระดับของของเหลวกับการควบคุมการไหลของของเหลว

โดยจะทำการวัดระดับของเหลวในกระบอกแก้วส่งสัญญาณเอาท์พุทเป็นความดันลม 3 ถึง 15 psig ไปยังหน่วยเปลี่ยนสัญญาณการดันลมไปเป็นสัญญาณไฟตรง สัญญาณนี้จะเปลี่ยนแปลงอย่างเป็นสัดส่วนกับระดับของเหลว โดยเรียกสัญญาณนี้ว่า ค่าตัวแปรขบวนการ ( PROCESS VARIABLE : PV ) ค่าตัวแปรนี้จะถูกส่งไปยังหน่วยควบคุมเพื่อทำการประมวลผลเปรียบเทียบกับค่าที่กำหนดเอาไว้ ( SET POINT : SP ) การประมวลผลนี้จะเป็นลักษณะแบบ PID โดยผลลัพธ์ที่ได้จากการประมวลผลนี้เรียกว่า ค่าตัวแปรจัดการ ( Manipulated Variable : MV ) ซึ่งเป็นสัญญาณดิจิทัล ค่าตัวแปรนี้จะมีหน้าที่ควบคุมกระบวนการให้มีค่าตามที่กำหนดไว้ จะถูกไปแปลงให้เป็นสัญญาณกระแสไฟตรง แล้งจึงถูกส่งเข้าหน่วยแปลงสัญญาณกระแสไฟตรงเป็นสัญญาณความดันลม จากนั้นจึงถูกส่งออกไปเป็นสัญญาณอินพุทให้กับวาล์วควบคุม โดยวาล์วควบคุมนี้จะทำหน้าที่ควบคุมระดับของของเหลวตามค่าของตัวแปรจัดการ ที่ส่งมาจากหน่วยควบคุมหลัก เพื่อให้ระดับของของเหลวเป็นไปตามค่าที่ต้องการ

#### ลำดับขั้นตอนการทดลอง

1. ทำการปรับเทียบ ( Calibrate ) อุปกรณ์ทุกหน่วยให้ได้ค่าตามมาตรฐานที่กำหนดไว้
2. ต่ออุปกรณ์ต่างๆที่เกี่ยวข้องเข้ากับขบวนการจำลอง
3. เริ่มขบวนการควบคุม โดยการเซตค่าพารามิเตอร์ต่างๆเช่น .SP KP Ti Td จากการป้อนค่าเริ่มแรกจากดีสบอร์ดที่ตัวคอนโทรลเลอร์ หรือป้อนทาง PC ก็ได้ เมื่อค่าพารามิเตอร์ต่างๆเรียบร้อยแล้ว กำหนดให้อยู่ในโหมดออโตเมตริกและสังเกตการเปลี่ยนแปลงของขบวนการว่าระดับของเหลวมีการเปลี่ยนแปลงตามค่าที่ตั้งไว้หรือไม่
4. ทดลองเปลี่ยนค่า SP หลังจากเลือกค่าพารามิเตอร์ต่างๆ สังเกตดูว่าคอนโทรลเลอร์สามารถควบคุมขบวนการได้หรือไม่ ถ้าไม่ได้แสดงว่าค่าพารามิเตอร์ที่เลือกไว้ยังไม่ถูกต้อง ให้เลือกค่าพารามิเตอร์ใหม่
5. ทดลองเปลี่ยนแปลงโหลดเพื่อทำให้เกิดสัญญาณรบกวนขึ้นในขบวนการได้หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ผลการทดลอง

ผลจากการทดลอง ในการควบคุมขบวนการจำลองปรากฏว่าสามารถควบคุมได้เป็นอย่างดีรูปต่าง ๆ ต่อไปนี้จะแสดงถึงผลที่ได้จากการทดลองคือ

การควบคุมระดับของของเหลว ในกรณีที่มีการเปลี่ยนแปลงของ SP และค่าพารามิเตอร์ของหน่วยควบคุมหลักถูกเลือกไว้อย่างเหมาะสมดังนี้

$$K_p = 1$$

$$T_i = 60 \text{ s}$$

SP มีการเปลี่ยนแปลงตั้งค่า 3 ค่า

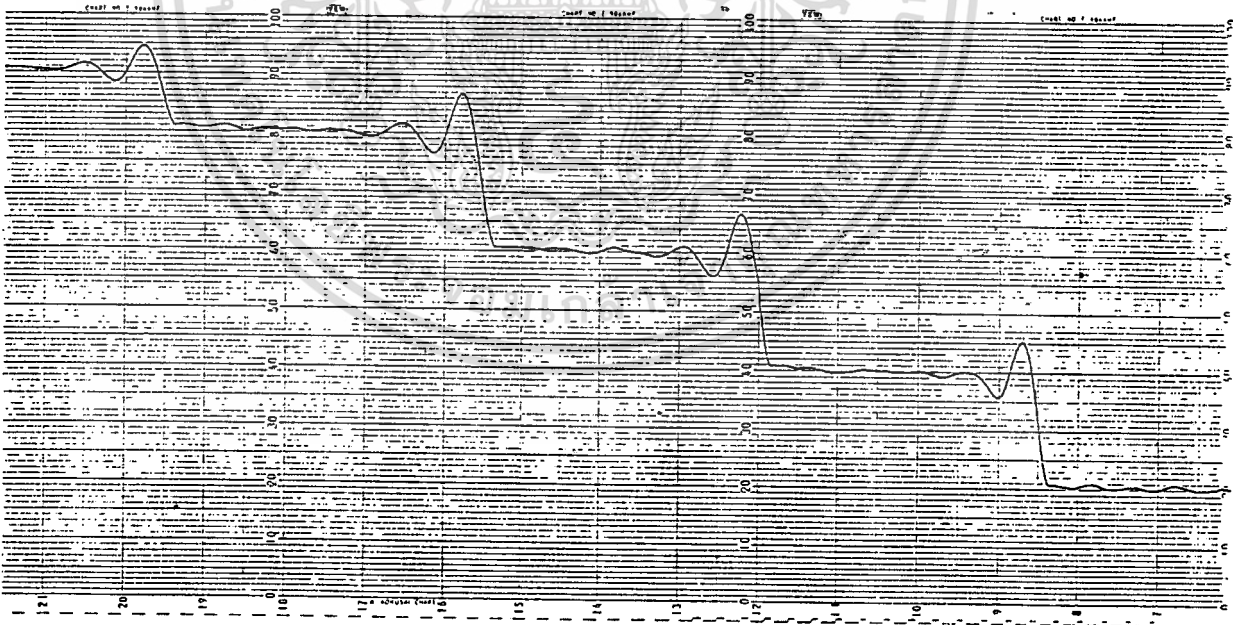
$$SP = 20$$

$$SP = 40$$

$$SP = 60$$

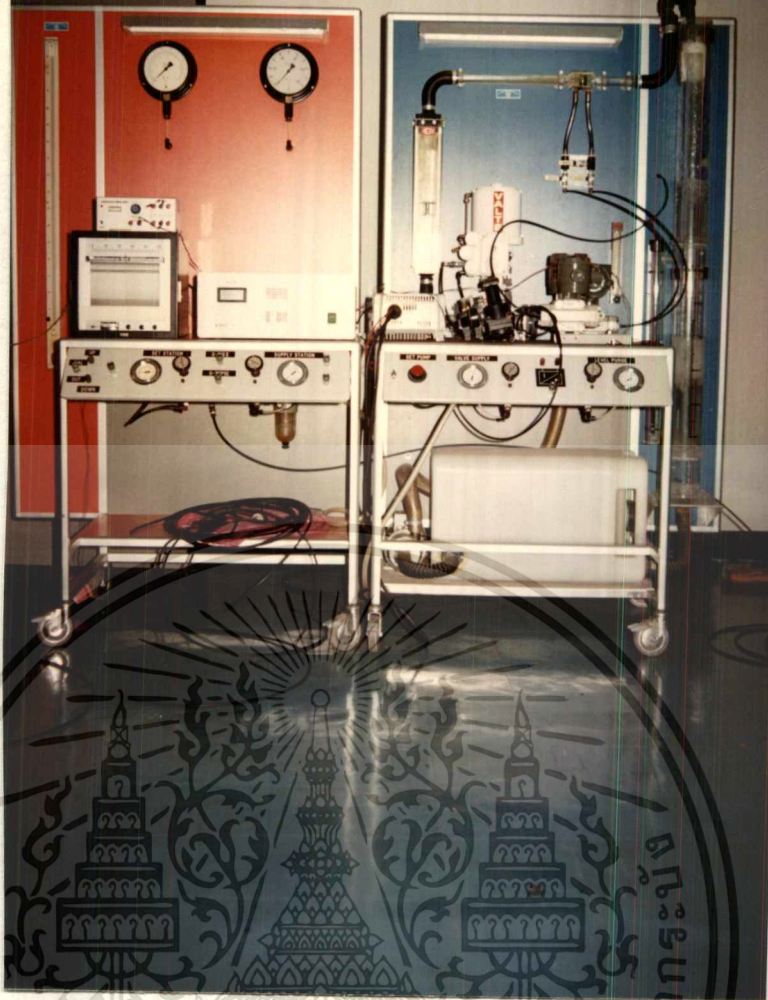
$$SP = 80$$

$$SP = 90$$



### รูปกราฟการทดลองของ LEVEL PROCESS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เฉพาะการศึกษาค้นคว้าเท่านั้น เมื่อผู้ดูแลเห็นประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปของ Process ที่ใช้ทดลอง Level Process



ในรูปแบบการแสดงค่าพารามิเตอร์ของการควบคุมของ Level Process

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมการไหลของของเหลว กรณีที่มีการเปลี่ยนแปลงของ SP และค่าพารามิเตอร์ของหน่วยควบคุมหลักถูกเลือกไว้อย่างเหมาะสมดังนี้

$$K_p = 2$$

$$T_i = 100 \text{ s}$$

SP มีการเปลี่ยนแปลงตั้งค่า 3 ค่า

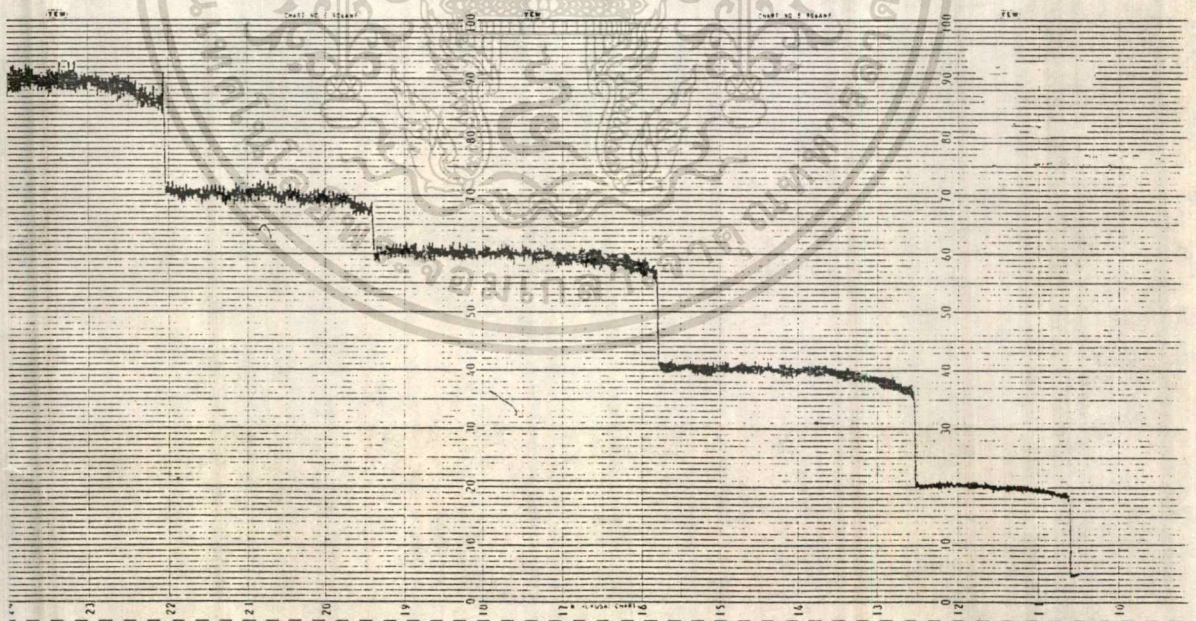
$$SP = 20$$

$$SP = 40$$

$$SP = 60$$

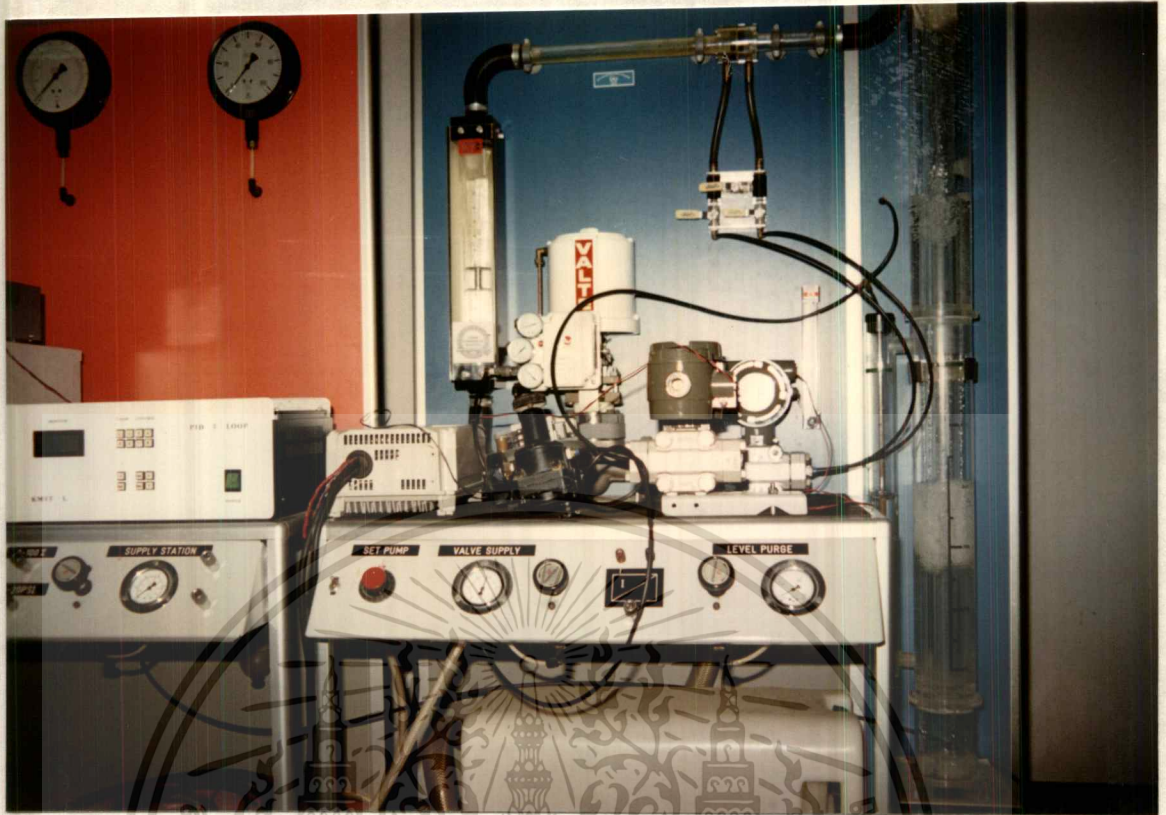
$$SP = 70$$

$$SP = 90$$



### รูปกราฟการทดลองของ FLOW PROCESS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปของ Process ที่ใช้ทดลอง Flow Process



ในรูปเป็นการแสดงค่าพารามิเตอร์ของการควบคุม Flow Process

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลอง การควบคุมแบบ Cascade โดยกำหนดค่าต่างให้ดังต่อไปนี้

โดยให้ Loop 1 เป็น Level Process

Loop 2 เป็น Flow Process

$$K_p = 2$$

$$T_i = 60$$

SP มีการเปลี่ยนแปลงค่า 5 ระดับ

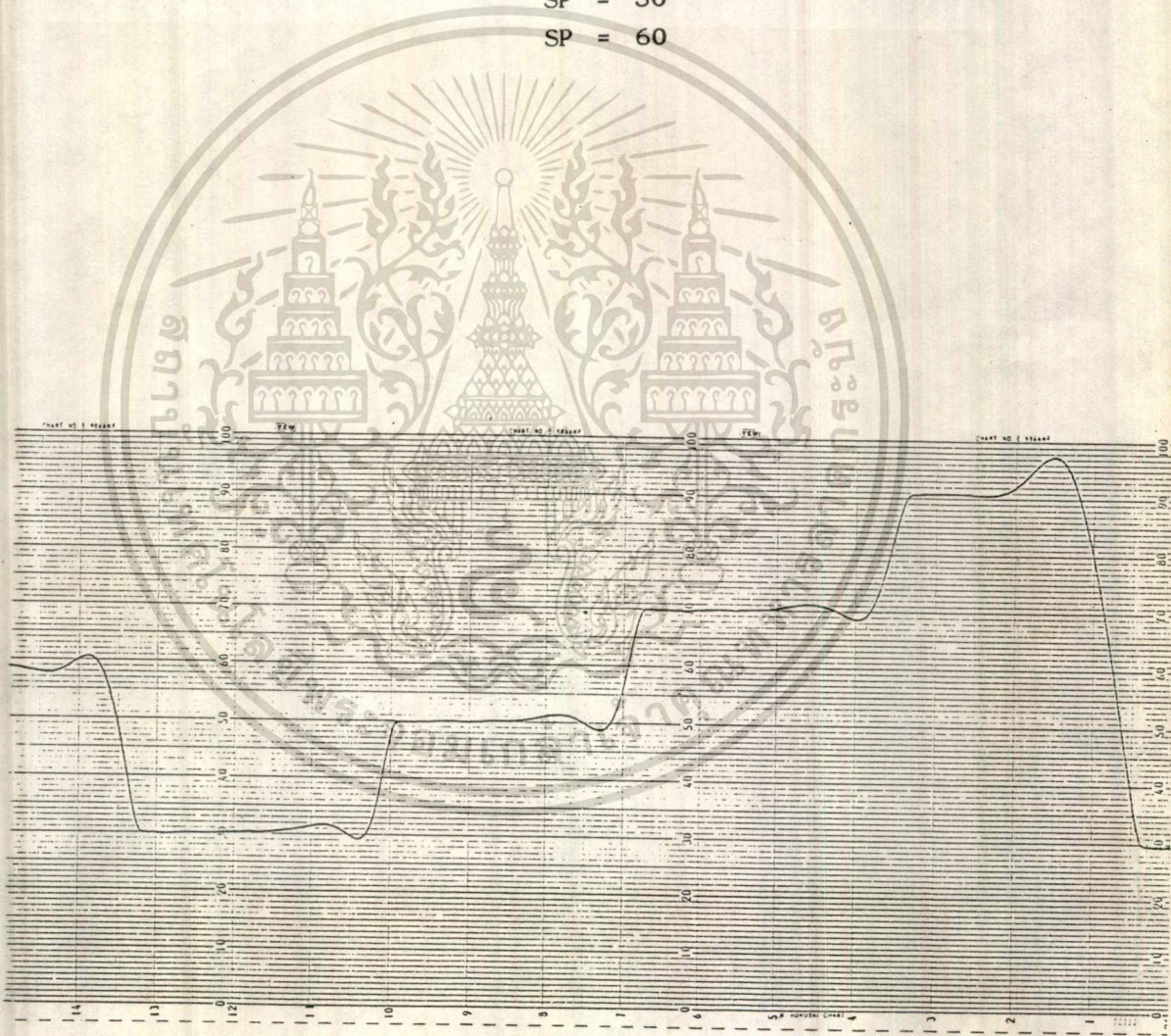
$$SP = 90$$

$$SP = 70$$

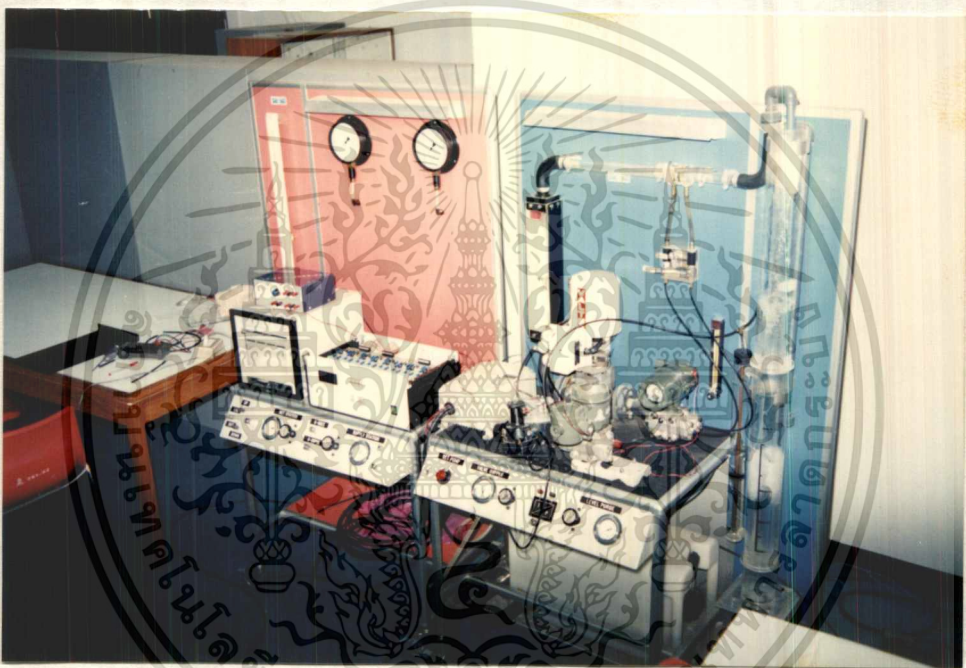
$$SP = 50$$

$$SP = 30$$

$$SP = 60$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้รูปกราฟการทดลองการควบคุมแบบ Cascade ซึ่งประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปของ *Process* ที่ใช้ทำการทดลอง การควบคุมแบบ *Cascade*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลอง การควบคุมแบบ RATIO โดยกำหนดค่าต่างๆให้ดังต่อไปนี้

โดยให้ LOOP 1 และ LOOP 2 LEVEL PROCESS

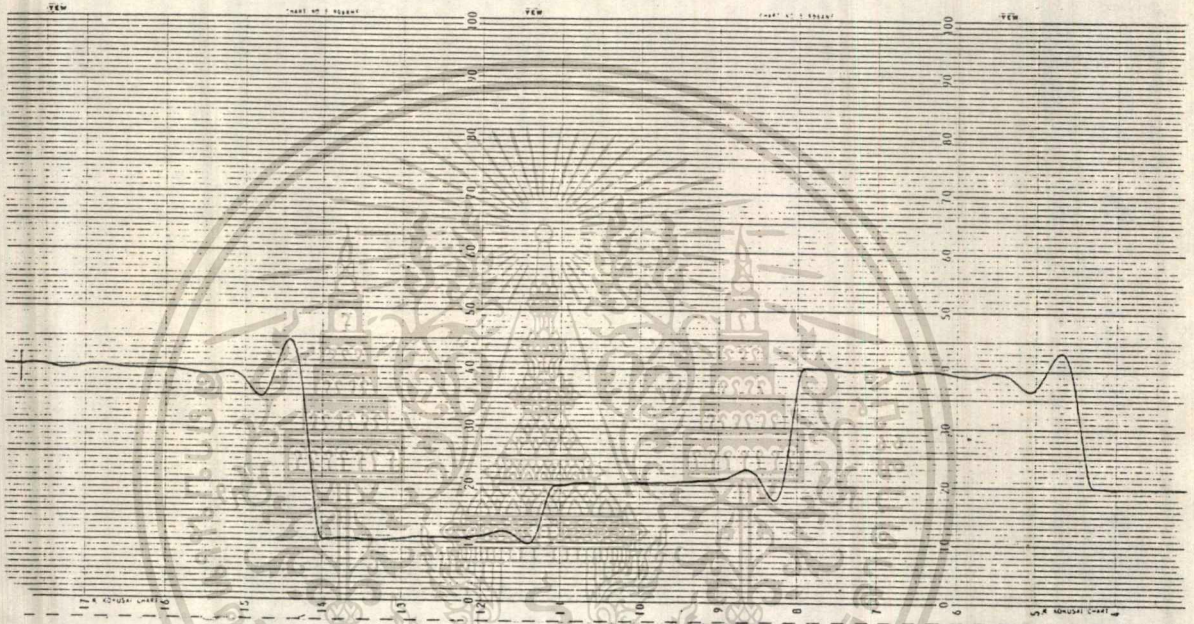
Level Ratio = 2

P = 2

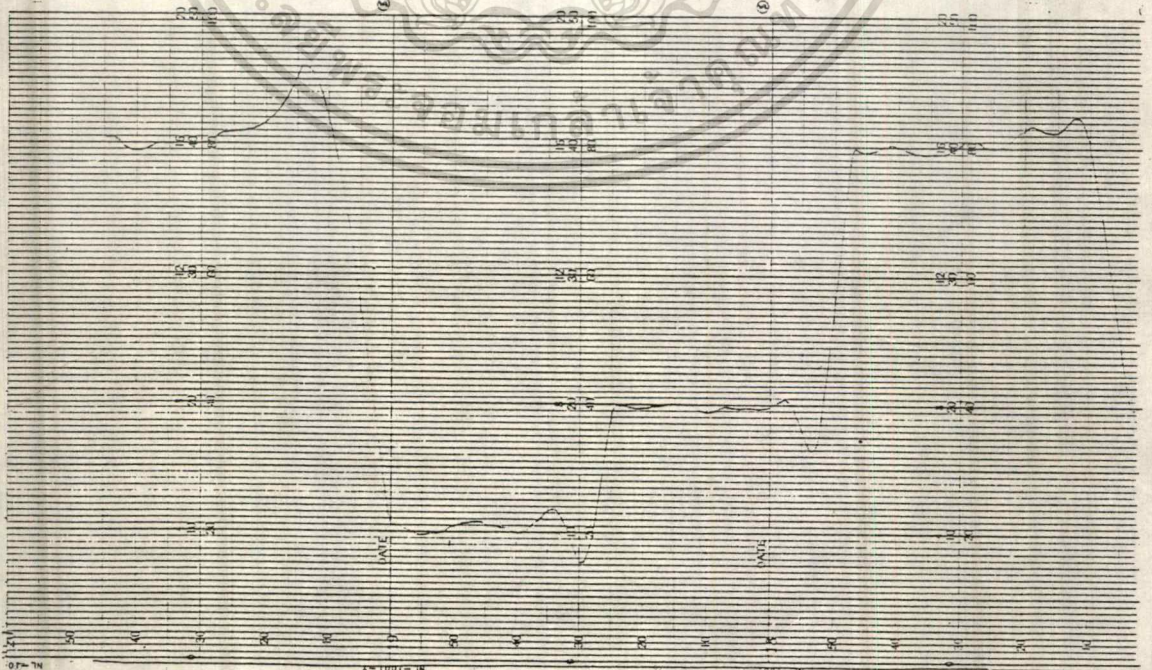
Ti = 60

LOOP 1 มีการเลื่อนของกระดาษ Recorder = 25 mm/min

LOOP 2 มีการเลื่อนของกระดาษ Recorder = 1440 mm/hr

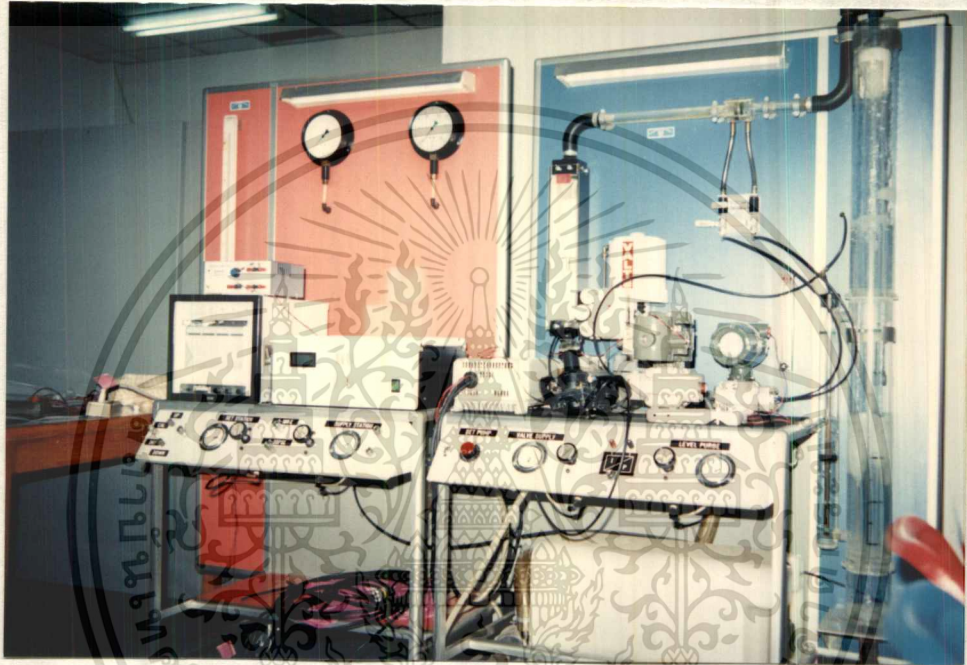


รูปกราฟของ Process Loop 1 ที่เป็น Loop อ่างอิง



รูปกราฟของ Process Loop 2 ที่ถูกทำการ Ratio เทียบกับ Loop 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ปรากฏทั้ง 2 รูปเป็นการแสดงผลตอบสนองของการควบคุมแบบ Ratio



รูป ของ Process

ที่ทำการทดลองการควบคุมแบบ Ratio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สรุปผลการทดลอง

จากการทดลองหน่วยควบคุมกับกระบวนการจำลองปรากฏว่า สามารถควบคุมระบบการได้ โดยเราสามารถใส่ค่าพารามิเตอร์ต่างๆผ่านทางเครื่องคอมพิวเตอร์ หรือใส่ค่าที่ตัวเครื่องได้โดย

### ช่วงของค่าพารามิเตอร์

|                          |   |             |
|--------------------------|---|-------------|
| - Sampling Time (DT)     | = | 62.5 ms     |
| - Proportional Gain (Kp) | = | 0 - 256     |
| - Integral Time (Ti)     | = | 4095.9375 s |
| - Derivative Time (Td)   | = | 4095.9375 s |
| - Initial (BIAS)         | = | 50 %        |

### บทสรุป

เนื่องจากโครงงานนี้ เป็นการนำเอา ไมโครคอนโทรลเลอร์ เบอร์ 68HC11 และ ไมโครคอมพิวเตอร์ มาประยุกต์ใช้งานทางอุตสาหกรรม โดยใช้เป็นหน่วยรับส่งข้อมูลและเป็นหน่วยประมวลผล โดยการควบคุมจะเป็นแบบ PID CONTROLLER และยังสามารถเปลี่ยนแปลงค่า PARAMETERS ต่าง ๆ ของการควบคุมได้ โดยการเปลี่ยนค่าจาก ไมโครคอมพิวเตอร์ซึ่งจะมีส่วน SOFT WARE ช่วยในการสนับสนุน ในส่วนของการรับค่าและป้อนค่าจากคีย์บอร์ด

สำหรับการทดลองนี้ ได้นำเอาไมโครคอนโทรลเลอร์เบอร์ 68HC11 เป็นหน่วยควบคุมหลักซึ่งจะทำการติดต่อระหว่าง HARD WARE ต่าง ๆ ในส่วน DIGITAL ให้มีการทำงานประสานสัมพันธ์กันอย่างมีประสิทธิภาพ และ จากการทดลองในส่วนนี้ ได้ผลเป็นที่น่าพอใจ

ในส่วนของการรับส่ง IN PUT และ OUT PUT หรือในส่วนของ ANALOG ซึ่งออกแบบเพื่อรับค่าและส่งค่าในรูปแบบของ VOLTAGE และ CURRENT ซึ่งในส่วนนี้ได้ทดลองแล้วได้ผลเป็นที่น่าพอใจ

ปัญหาที่เกิดขึ้นในการทดลองครั้งนี้ จะแบ่งออกเป็น 2 ข้อด้วยกันคือ

1. ในส่วนของหน่วยควบคุมหลัก
2. ในส่วนของหน่วย รับ - ส่ง IN PUT และ OUT PUT

ในส่วนของหน่วยควบคุมหลัก ปัญหาจะอยู่ที่แหล่งจ่ายแรงดัน DC ต้องมีเสถียรภาพดีคือหมายถึง RIPPLE จะต้องมีให้น้อยที่สุดและกระแสต้องมีเพียงพอสำหรับจ่ายให้กับอุปกรณ์ต่างๆ เนื่องจากปัญหาเหล่านี้เป็นตัวแปรสำคัญที่จะทำให้ วงจรหน่วยควบคุมหลักสามารถทำงานได้อย่างมีประสิทธิภาพและเสถียรภาพมากที่สุด

ในส่วนของหน่วยรับส่ง IN PUT และ OUT PUT ปัญหาอยู่ที่ความไม่เป็นเชิงเส้นของอุปกรณ์ต่างๆ ที่ประกอบกัน ทำให้เสถียรภาพของวงจรไม่ดีเพียงพอ ในการปรับแต่งวงจรมันเราจึงต้องทำการปรับแต่งหลายครั้ง เพื่อให้ได้ค่าที่ถูกต้องและมีเสถียรภาพจนเป็นที่น่าพอใจและยังสามารถนำมาใช้งานได้อย่างมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#ifdef _TINY_
#error Graphics demos will not run in the tiny model.
#endif
```

```
#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include <stdarg.h>
#include <graphics.h>
#include <iostream.h>
#include <alloc.h>
#include <string.h>
```

```
#define PORT 1
#define Comp (0.392157)
#define Cont (0.0625)
#define NEXT 10
#define INC 11
#define DEC 12
#define SPINC 13
#define SPDEC 14
#define EXIT 15
```

```
#include "gprintt.i"
#include "mouse.i"
#include "button.i"
```

```
Mstatus Position;
Button bkgdpull;
Button bkgdpush;
Button bkbutton[17];
```

```
int GraphDriver;
int GraphMode;
int MaxColors;
int ErrorCode = 0;
int MaxX,MaxY;
int a[9]; //Step of loop
int page= 1, c=0;
int Exit=0;
int loop=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int dcon= 1;
```

```
int TPos( int TP, int Low, int High )
```

```
return( ( TP >= Low ) && ( TP <= High ) );
```

```
Port_Init( )
```

```
union REGS r;
r.x.dx = PORT;          //Number port
r.h.ah = 0;
r.h.al = 0xe3;          //Baud rate 9600 non parity 8 bit 1 stop bit
int86(0x14,&r,&r);
```

```
Check_State( )
```

```
union REGS r;
r.x.dx = PORT;          //Number port
r.h.ah = 3;
int86(0x14,&r,&r);
return r.x.ax;
```

```
Rport( )
```

```
union REGS r;
char c;
while(!(Check_State()&256))
```

```
r.x.dx = PORT;          //Number port
r.h.ah = 2;
int86(0x14,&r,&r);
if(r.h.ah & 128)
    printf("read error detected in serial port");
return r.h.al;
```

```
Sport(char c)
```

```
int ch=c;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

union REGS r;

r.x.dx = PORT;      //Number port
r.h.al = c;
r.h.ah = 1;
int86(0x14,&r,&r);
if(r.h.ah & 128)

printf("send error detected in serial port %d", r.h.ah);
exit(1);

```

```

Check_Installed_System()

```

```

int xloc,yloc;
union REGS r;
do
{
r.x.dx = PORT;      //Number port
r.h.al = 0;
r.h.ah = 1;
int86(0x14,&r,&r);
if((r.h.ah & 128))

setcolor(DARKGRAY);
xloc = 320; yloc = 240;
settextstyle( DEFAULT_FONT, HORIZ_DIR, 2 );
gprintf( &xloc, &yloc, "INSTALL CONTROLLER SYSTEM PLEASE" );
xloc = 320; yloc = 450;
settextstyle( DEFAULT_FONT, HORIZ_DIR, 1 );
gprintf( &xloc, &yloc, "Press key to exit program" );
if(kbhit())

_getch();
_exit(1);

```

```

while(!(Check_State()&256));
Rport();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Initialize()
```

```
    GraphDriver = DETECT;
```

```
    initgraph( &GraphDriver, &GraphMode, "B:" );
```

```
    ErrorCode = graphresult();
```

```
    if ( ErrorCode != grOk )
```

```
        printf(" Graphics System Error: %s",grapherrormsg( ErrorCode ) );
```

```
        exit ( 1 );
```

```
    MaxColors = getmaxcolor() + 1;
```

```
    MaxX = getmaxx();
```

```
    MaxY = getmaxy();
```

```
void Pause()
```

```
    if( kbhit() ) getch;
```

```
    getch();
```

```
enum type C, SP, KP, TI, TD, CON, MOD, PV, MV ;
```

```
enum mode NC, MNU, PI, PD, PID ;
```

```
enum con DIR, REV=0xFF ;
```

```
enum Act on, off=0xFF ;
```

```
class Lcontrol
```

```
enum mode Mode;
```

```
enum Act Active;
```

```
enum con Con;
```

```
int InitX, InitY;
```

```
int Page;
```

```
int Loop;
```

```
unsigned int Kp, Ti, Td;
```

```
float Pv, Mv, Sp;
```

```
float FKp, FTi, FTd;
```

```
Button button[9];
```

```
public:
```

```
Lcontrol( int I);
```

```
void Increase( enum type t );
```

```
void Decrease( enum type t );
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void SpeedInc( enum type t );
void SpeedDec( enum type t );
enum Act Power()      return Active; //question ON, OFF
void Power( enum Act a ) Active = a;
void Gloop();
void Mem( int n );
enum con Gcon()      return Con;
int Gpage()          return Page;
void Hdata( int n );
void Sdata();
void Sdata(enum type t);
friend void Arrage( Lcontrol* l, int* a );
friend void Dpage1( Lcontrol* l );
friend void Dpage2( Lcontrol* l );
friend void DataHit( Lcontrol* l );
friend void ConHit( Lcontrol* l );
friend void ActHit( Lcontrol* l );
friend void KeyHit( Lcontrol* l );
friend void Input_Port( Lcontrol* l );
friend void Init_Input_Port( Lcontrol* l );

void Input_Port( Lcontrol* l )
{
    static int code=0;
    int getdata;
    float data;
    if(code<=30)
    {
        Sport(code);
        data = Rport();
        switch (code)
        {
            case 0: |[1].Pv = (data*100/255); if(page==|[1].Gpage()) |[1].Sdata(PV)
; break;
            case 2: |[1].Mv = (data*100/255); if(page==|[1].Gpage()) |[1].Sdata
(MV); break;
            case 4: |[2].Pv = (data*100/255); if(page==|[2].Gpage()) |[2].Sdata(PV)
; break;
            case 6: |[2].Mv = (data*100/255); if(page==|[2].Gpage()) |[2].Sdata
(MV); break;
            case 8: |[3].Pv = (data*100/255); if(page==|[3].Gpage()) |[3].Sdata(PV)
; break;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 10: |[3].Mv = (data*100/255); if(page==|[3].Gpage()) |[3].Sdata
(MV); break;
case 12: |[4].Pv = (data*100/255); if(page==|[4].Gpage()) |[4].Sdata
(PV); break;
case 14: |[4].Mv = (data*100/255); if(page==|[4].Gpage()) |[4].Sdata
(MV); break;
case 16: |[5].Pv = (data*100/255); if(page==|[5].Gpage()) |[5].Sdata
(PV); break;
case 18: |[5].Mv = (data*100/255); if(page==|[5].Gpage()) |[5].Sdata
(MV); break;
case 20: |[6].Pv = (data*100/255); if(page==|[6].Gpage()) |[6].Sdata
(PV); break;
case 22: |[6].Mv = (data*100/255); if(page==|[6].Gpage()) |[6].Sdata
(MV); break;
case 24: |[7].Pv = (data*100/255); if(page==|[7].Gpage()) |[7].Sdata
(PV); break;
case 26: |[7].Mv = (data*100/255); if(page==|[7].Gpage()) |[7].Sdata
(MV); break;
case 28: |[8].Pv = (data*100/255); if(page==|[8].Gpage()) |[8].Sdata
(PV); break;
case 30: |[8].Mv = (data*100/255); if(page==|[8].Gpage()) |[8].Sdata
(MV); break;
if((code>30)&&(code<=46))
Sport(code);
code=Rport();
switch(code)
case 32: Sport(getdata = |[1].Gcon()); break;
case 34: Sport(getdata = |[2].Gcon()); break;
case 36: Sport(getdata = |[3].Gcon()); break;
case 38: Sport(getdata = |[4].Gcon()); break;
case 40: Sport(getdata = |[5].Gcon()); break;
case 42: Sport(getdata = |[6].Gcon()); break;
case 44: Sport(getdata = |[7].Gcon()); break;
case 46: Sport(getdata = |[8].Gcon()); break;

```

```
if(code>46)
```

```
Sport(code);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

code=Rport();
switch(code)

case 48: Sport([1].Mode); break;
case 50: Sport([1].Kp/0x100); break;
case 52: Sport([1].Kp%0x100); break;
case 54: Sport([1].Ti/0x100); break;
case 56: Sport([1].Ti%0x100); break;
case 58: Sport([1].Td/0x100); break;
case 60: Sport([1].Td%0x100); break;
case 62: Sport(int([1].Sp*2.55)); break;
case 64: Sport([1].Active); break;
case 66: Sport([2].Mode); break;
case 68: Sport([2].Kp/0x100); break;
case 70: Sport([2].Kp%0x100); break;
case 72: Sport([2].Ti/0x100); break;
case 74: Sport([2].Ti%0x100); break;
case 76: Sport([2].Td/0x100); break;
case 78: Sport([2].Td%0x100); break;
case 80: Sport(int([2].Sp*2.55)); break;
case 82: Sport([2].Active); break;
case 84: Sport([3].Mode); break;
case 86: Sport([3].Kp/0x100); break;
case 88: Sport([3].Kp%0x100); break;
case 90: Sport([3].Ti/0x100); break;
case 92: Sport([3].Ti%0x100); break;
case 94: Sport([3].Td/0x100); break;
case 96: Sport([3].Td%0x100); break;
case 98: Sport(int([3].Sp*2.55)); break;
case 100: Sport([3].Active); break;
case 102: Sport([4].Mode); break;
case 104: Sport([4].Kp/0x100); break;
case 106: Sport([4].Kp%0x100); break;
case 108: Sport([4].Ti/0x100); break;
case 110: Sport([4].Ti%0x100); break;
case 112: Sport([4].Td/0x100); break;
case 114: Sport([4].Td%0x100); break;
case 116: Sport(int([4].Sp*2.55)); break;
case 118: Sport([4].Active); break;
case 120: Sport([5].Mode); break;
case 122: Sport([5].Kp/0x100); break;
case 124: Sport([5].Kp%0x100); break;
case 126: Sport([5].Ti/0x100); break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 128: Sport(l[5].Ti%0x100); break;
case 130: Sport(l[5].Td/0x100); break;
case 132: Sport(l[5].Td%0x100); break;
case 134: Sport(int(l[5].Sp*2.55)); break;
case 136: Sport(l[5].Active); break;
case 138: Sport(l[6].Mode); break;
case 140: Sport(l[6].Kp/0x100); break;
case 142: Sport(l[6].Kp%0x100); break;
case 144: Sport(l[6].Ti/0x100); break;
case 146: Sport(l[6].Ti%0x100); break;
case 148: Sport(l[6].Td/0x100); break;
case 150: Sport(l[6].Td%0x100); break;
case 152: Sport(int(l[6].Sp*2.55)); break;
case 154: Sport(l[6].Active); break;
case 156: Sport(l[7].Mode); break;
case 158: Sport(l[7].Kp/0x100); break;
case 160: Sport(l[7].Kp%0x100); break;
case 162: Sport(l[7].Ti/0x100); break;
case 164: Sport(l[7].Ti%0x100); break;
case 166: Sport(l[7].Td/0x100); break;
case 168: Sport(l[7].Td%0x100); break;
case 170: Sport(int(l[7].Sp*2.55)); break;
case 172: Sport(l[7].Active); break;
case 174: Sport(l[8].Mode); break;
case 176: Sport(l[8].Kp/0x100); break;
case 178: Sport(l[8].Kp%0x100); break;
case 180: Sport(l[8].Ti/0x100); break;
case 182: Sport(l[8].Ti%0x100); break;
case 184: Sport(l[8].Td/0x100); break;
case 186: Sport(l[8].Td%0x100); break;
case 188: Sport(int(l[8].Sp*2.55)); break;
case 190: Sport(l[8].Active); break;

```

```

if(code<190)
    code = code+2;
else code = 0;

```

```
void Init_Input_Port( Lcontrol* l )
```

```

int getdata;
float data;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

unsigned int unidata;
for(int code=0; code<=30; code+=2)

    if(code<=30)

        Sport(code);
        data = Rport();
        switch (code)

            case 0: |[1].Pv = (data*100/255); break;
            case 2: |[1].Mv = (data*100/255); break;
            case 4: |[2].Pv = (data*100/255); break;
            case 6: |[2].Mv = (data*100/255); break;
            case 8: |[3].Pv = (data*100/255); break;
            case 10: |[3].Mv = (data*100/255); break;
            case 12: |[4].Pv = (data*100/255); break;
            case 14: |[4].Mv = (data*100/255); break;
            case 16: |[5].Pv = (data*100/255); break;
            case 18: |[5].Mv = (data*100/255); break;
            case 20: |[6].Pv = (data*100/255); break;
            case 22: |[6].Mv = (data*100/255); break;
            case 24: |[7].Pv = (data*100/255); break;
            case 26: |[7].Mv = (data*100/255); break;
            case 28: |[8].Pv = (data*100/255); break;
            case 30: |[8].Mv = (data*100/255); break;

        for(code=32; code<=190; code+=2)

            Sport(0xFF);
            Rport();
            Sport(code);
            unidata=data=Rport();
            switch(code)

                case 32: |[1].Con = unidata; break;
                case 34: |[2].Con = unidata; break;
                case 36: |[3].Con = unidata; break;
                case 38: |[4].Con = unidata; break;
                case 40: |[5].Con = unidata; break;
                case 42: |[6].Con = unidata; break;
                case 44: |[7].Con = unidata; break;

```

```

case 46: I[8].Con = unidata; break;
case 48: I[1].Mode = unidata; break;
case 50: I[1].Kp = unidata*0x100; break;
case 52: I[1].Kp l = unidata; break;
case 54: I[1].Ti = unidata*0x100; break;
case 56: I[1].Ti l = unidata; break;
case 58: I[1].Td = unidata*0x100; break;
case 60: I[1].Td l = unidata; break;
case 62: I[1].Sp = data/2.55; break;
case 64: I[1].Active = unidata; break;
case 66: I[2].Mode = unidata; break;
case 68: I[2].Kp = unidata*0x100; break;
case 70: I[2].Kp l = unidata; break;
case 72: I[2].Ti = unidata*0x100; break;
case 74: I[2].Ti l = unidata; break;
case 76: I[2].Td = unidata*0x100; break;
case 78: I[2].Td l = unidata; break;
case 80: I[2].Sp = data/2.55; break;
case 82: I[2].Active = unidata; break;
case 84: I[3].Mode = unidata; break;
case 86: I[3].Kp = unidata*0x100; break;
case 88: I[3].Kp l = unidata; break;
case 90: I[3].Ti = unidata*0x100; break;
case 92: I[3].Ti l = unidata; break;
case 94: I[3].Td = unidata*0x100; break;
case 96: I[3].Td l = unidata; break;
case 98: I[3].Sp = data/2.55; break;
case 100: I[3].Active = unidata; break;
case 102: I[4].Mode = unidata; break;
case 104: I[4].Kp = unidata*0x100; break;
case 106: I[4].Kp l = unidata; break;
case 108: I[4].Ti = unidata*0x100; break;
case 110: I[4].Ti l = unidata; break;
case 112: I[4].Td = unidata*0x100; break;
case 114: I[4].Td l = unidata; break;
case 116: I[4].Sp = data/2.55; break;
case 118: I[4].Active = unidata; break;
case 120: I[5].Mode = unidata; break;
case 122: I[5].Kp = unidata*0x100; break;
case 124: I[5].Kp l = unidata; break;
case 126: I[5].Ti = unidata*0x100; break;
case 128: I[5].Ti l = unidata; break;
case 130: I[5].Td = unidata*0x100; break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 132: l[5].Td l = unidata; break;
case 134: l[5].Sp = data/2.55; break;
case 136: l[5].Active = unidata; break;
case 138: l[6].Mode = unidata; break;
case 140: l[6].Kp = unidata*0x100; break;
case 142: l[6].Kp l = unidata; break;
case 144: l[6].Ti = unidata*0x100; break;
case 146: l[6].Ti l = unidata; break;
case 148: l[6].Td = unidata*0x100; break;
case 150: l[6].Td l = unidata; break;
case 152: l[6].Sp = data/2.55; break;
case 154: l[6].Active = unidata; break;
case 156: l[7].Mode = unidata; break;
case 158: l[7].Kp = unidata*0x100; break;
case 160: l[7].Kp l = unidata; break;
case 162: l[7].Ti = unidata*0x100; break;
case 164: l[7].Ti l = unidata; break;
case 166: l[7].Td = unidata*0x100; break;
case 168: l[7].Td l = unidata; break;
case 170: l[7].Sp = data/2.55; break;
case 172: l[7].Active = unidata; break;
case 174: l[8].Mode = unidata; break;
case 176: l[8].Kp = unidata*0x100; break;
case 178: l[8].Kp l = unidata; break;
case 180: l[8].Ti = unidata*0x100; break;
case 182: l[8].Ti l = unidata; break;
case 184: l[8].Td = unidata*0x100; break;
case 186: l[8].Td l = unidata; break;
case 188: l[8].Sp = data/2.55; break;
case 190: l[8].Active = unidata; break;

```

```
void RestoreView()
```

```
setviewport( 0, 0, 639, 479, TRUE );
```

```
Lcontrol::Lcontrol( int l )
```

```
Mv = Pv = Ti = Td = 0x5;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Sp = 0;
Active = on;
Kp = 0x50;
Con = REV;
Mode = PID;
Loop = I;
Hdata(0);

```

```
void Lcontrol::Mem( int n )
```

```
switch( n )
```

```

case 1: InitX = 100;
-   InitY = 90;
-   Page = 1;
-   break;
case 2: InitX = 220;
-   InitY = 90;
-   Page = 1;
-   break;
case 3: InitX = 340;
-   InitY = 90;
-   Page = 1;
-   break;
case 4: InitX = 460;
-   InitY = 90;
-   Page = 1;
-   break;
case 5: InitX = 100;
-   InitY = 90;
-   Page = 2;
-   break;
case 6: InitX = 220;
-   InitY = 90;
-   Page = 2;
-   break;
case 7: InitX = 340;
-   InitY = 90;
-   Page = 2;
-   break;
case 8: InitX = 460;
-   InitY = 90;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- \_ Page = 2;
- \_ break;

```
void Lcontrol::Increase( enum type t )
```

```
switch( t )
```

```

case PV: if(Pv<100) Pv+=Comp; if(Pv>100) Pv = 100; break;
case MV: if(Mv<100) Mv+=Comp; if(Mv>100) Mv = 100; break;
case SP: if(Sp<100) Sp+=Comp; if(Sp>100) Sp = 100; break;
case KP: if(Kp<0xFFFF) Kp+=0x10; if(Kp>0xFFFF)Kp = 0xFFFF; break;
case TI: if(Ti<0xFFFF) Ti+=0x1; if(Ti>0xFFFF)Ti = 0xFFFF; break;
case TD: if(Td<0xFFFF) Td+=0x1; if(Td>0xFFFF)Td = 0xFFFF; break;
case CON: if(Con==DIR)Con = REV; else Con = DIR; break;
case MOD: if(Mode<4) Mode++; else if(Mode>=4) Mode = 1; break;

```

```
void Lcontrol::Decrease( enum type t )
```

```
switch( t )
```

```

case PV: if(Pv>0) Pv-=Comp; if(Pv<0) Pv=0; break;
case MV: if(Mv>0) Mv-=Comp; if(Mv<0) Mv=0; break;
case SP: if(Sp>0) Sp-=Comp; if(Sp<0) Sp=0; break;
case KP: if(Kp>0) Kp-=0x10; if(Kp<0) Kp=0; break;
case TI: if(Ti>0) Ti-=0x1; if(Ti<0) Ti=0; break;
case TD: if(Td>0) Td-=0x1; if(Td<0) Td=0; break;
case CON: if(Con==DIR)Con = REV; else Con = DIR; break;
case MOD: if(Mode>1) Mode--; else Mode = 4; break;

```

```
void Lcontrol::SpeedInc( enum type t )
```

```
switch( t )
```

```

case PV: if(Pv<100) Pv+=(50*Comp); if(Pv>100) Pv = 100; break;
case MV: if(Mv<100) Mv+=(50*Comp); if(Mv>100) Mv = 100; break;
case SP: if(Sp<100) Sp+=(50*Comp); if(Sp>100) Sp = 100; break;
case KP: if(Kp<0xFFFF) Kp+=0x300; if(Kp>0xFFFF) Kp = 0xFFFF;
break;
case TI: if(Ti<0xFFFF) Ti+=0x30; if(Ti>0xFFFF) Ti = 0xFFFF; break;

```

```

case TD: if(Td<0xFFFF) Td+=0x30; if(Td>0xFFFF) Td = 0xFFFF; break;
case CON: if(Con==DIR)Con = REV; else Con = DIR; break;
case MOD: if(Mode<4) Mode++; else if(Mode>=4) Mode = 1; break;

```

```
void Lcontrol::SpeedDec( enum type t )
```

```
switch( t )
```

```

case PV: if(Pv>0) Pv--=(50*Comp); if(Pv<0) Pv=0; break;
case MV: if(Mv>0) Mv--=(50*Comp); if(Mv<0) Mv=0; break;
case SP: if(Sp>0) Sp--=(50*Comp); if(Sp<0) Sp=0; break;
case KP: if(Kp>0) Kp--=0x300; if(Kp<0) Kp=0; break;
case TI: if(Ti>0) Ti--=0x30; if(Ti<0) Ti=0; break;
case TD: if(Td>0) Td--=0x30; if(Td<0) Td=0; break;
case CON: if(Con==DIR)Con = REV; else Con = DIR; break;
case MOD: if(Mode>1) Mode--; else if(Mode<=1) Mode = 4; break;

```

```
void Lcontrol::Gloop()
```

```

int xloc, yloc;
gmouse.Mshow(FALSE);
if(Active==on)
    bkgdpull.BkPull( InitX, InitY, 112, 353, 3 );
else
    bkgdpush.BkPush( InitX, InitY, 112, 353, 3 );
settextjustify( CENTER_TEXT, CENTER_TEXT );
bkgdpush.BkPush( InitX+28, InitY+40, 5, 160, 2 );
bkgdpush.BkPush( InitX+88, InitY+40, 5, 160, 2 );
bkgdpush.BkPush( InitX+25, InitY+215, 70, 5, 2 );
setcolor(DARKGRAY);
gprintxy( InitX+31, InitY+30, "SP" );
gprintxy( InitX+89, InitY+30, "PV" );
gprintxy( InitX+13, InitY+218, "MV" );
line( InitX+0, InitY+15, InitX+115, InitY+15 );
line( InitX+35, InitY+200, InitX+35, InitY+40 );
line( InitX+85, InitY+200, InitX+85, InitY+40 );
for( int n=0; n<=10; n++ )

    line( InitX+35, InitY+(n*16)+40, InitX+45, InitY+(n*16)+40 );
    line( InitX+75, InitY+(n*16)+40, InitX+85, InitY+(n*16)+40 );

```

```
if(n!=10)
```

```
    line( InitX+35, InitY+(n*16)+48, InitX+40, InitY+(n*16)+48 );
    line( InitX+80, InitY+(n*16)+48, InitX+85, InitY+(n*16)+48 );
```

```
int N=100-(n*10);
```

```
xloc = InitX+60; yloc = InitY+(n*16)+40;
```

```
gprintf( &xloc, &yloc, "%d", N );
```

```
rectangle( InitX+24, InitY+213, InitX+96, InitY+222 );
```

```
for( int m=0; m<=5; m++ )
```

```
    outtextxy( InitX+(m*14)+25, InitY+224, "." );
```

```
    outtextxy( InitX+25, InitY+232, "0" );
```

```
    outtextxy( InitX+95, InitY+232, "100" );
```

```
    setttextjustify( LEFT_TEXT, CENTER_TEXT );
```

```
    outtextxy( InitX+3, InitY+8, "LOOP" );
```

```
    xloc = InitX+36; yloc = InitY+8;
```

```
    gprintf( &xloc, &yloc, "%d", Loop );
```

```
    // rectangle( InitX+1, InitY+240, InitX+115, InitY+336 );
```

```
    // bkgdpull.Create( InitX+0, InitY+240, InitX+110, InitY+336, 1 );
```

```
    line( InitX+20, InitY+240, InitX+20, InitY+324 );
```

```
    line( InitX+27, InitY+324, InitX+27, InitY+350 );
```

```
    for( n=0; n<=7; n++ )
```

```
        line( InitX+0, InitY+240+(n*14), InitX+115, InitY+240+(n*14) );
```

```
    Hdata(0);
```

```
    Sdata();
```

```
    gmouse.Mshow(TRUE);
```

```
void Lcontrol::Hdata( int n )
```

```
    gmouse.Mshow(FALSE);
```

```
    button[1].CreateFlat( InitX+2, InitY+270, 16, 10, 2, "SP" );
```

```
    button[2].CreateFlat( InitX+2, InitY+284, 16, 10, 2, "KP" );
```

```
    button[3].CreateFlat( InitX+2, InitY+298, 16, 10, 2, "TI" );
```

```
    button[4].CreateFlat( InitX+2, InitY+312, 16, 10, 2, "TD" );
```

```
    button[5].CreateFlat( InitX+2, InitY+326, 25, 10, 2, "ACT" );
```

```
    button[6].CreateFlat( InitX+2, InitY+340, 25, 10, 2, "MOD" );
```

```
    button[7].CreateFlat( InitX+1, InitY+241, 18, 12, 1, "PV" );
```

```
    button[8].CreateFlat( InitX+1, InitY+255, 18, 12, 1, "MV" );
```

```
    if(n)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

button[n].Push();
gmouse.Mshow(TRUE);

```

```

void Lcontrol::Sdata()

```

```

int xloc, yloc;
gmouse.Mshow(FALSE);
setcolor( DARKGRAY );
settextjustify( LEFT_TEXT, CENTER_TEXT );
xloc = InitX+25; yloc = InitY+247;
erasestr( xloc, yloc, " " );
gprintf( &xloc, &yloc, "%3.3f", Pv );
yloc = InitY+261;
erasestr( xloc, yloc, " " );
gprintf( &xloc, &yloc, "%3.3f", Mv );
yloc = InitY+275;
erasestr( xloc, yloc, " " );
gprintf( &xloc, &yloc, "%3.3f", Sp );
yloc = InitY+289;
FKp = (Kp%0x100)>>4;
FKp = FKp/0x10;
FKp += Kp/0x100;
erasestr( xloc, yloc, " " );
gprintf( &xloc, &yloc, "%3.3f", FKp );
yloc = InitY+303;
FTi = (Ti%0x10);
FTi = FTi/0x10;
FTi += Ti/0x10;
erasestr( xloc, yloc, " " );
gprintf( &xloc, &yloc, "%4.3f", FTi );
yloc = InitY+317;
FTd = (Td%0x10);
FTd = FTd/0x10;
FTd += Td/0x10;
erasestr( xloc, yloc, " " );
gprintf( &xloc, &yloc, "%4.3f", FTd );
xloc = InitX+35; yloc = InitY+331;
if(Con==DIR) gprintc( &xloc, &yloc, "DIR" );
else gprintc( &xloc, &yloc, "REV" );
xloc = InitX+35; yloc = InitY+345;
switch(Mode)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case MNU: gprintc( &xloc, &yloc, "MNU" ); break;
case PI: gprintc( &xloc, &yloc, "PI" ); break;
case PD: gprintc( &xloc, &yloc, "PD" ); break;
case PID: gprintc( &xloc, &yloc, "PID" ); break;

```

```

setfillstyle( SOLID_FILL, LIGHTGRAY );
bar( InitX+27, InitY+40, InitX+32, InitY+200 );
bar( InitX+93, InitY+40, InitX+88, InitY+200 );
bar( InitX+25, InitY+220, InitX+95, InitY+215 );
setfillstyle( SOLID_FILL, LIGHTRED );
bar( InitX+28, InitY+(200-(int)(Sp*1.6)), InitX+33, InitY+200 );
bar( InitX+92, InitY+(200-(int)(Pv*1.6)), InitX+87, InitY+200 );
bar( InitX+25, InitY+215, InitX+25+(int)(Mv*0.7), InitY+220 );
gmouse.Mshow(TRUE);

```

```

void Lcontrol::Sdata(enum type t)

```

```

int xloc, yloc;
gmouse.Mshow(FALSE);
setcolor( DARKGRAY );
settextjustify( LEFT_TEXT, CENTER_TEXT );
switch(t)
{
case PV : xloc = InitX+25; yloc = InitY+247;
- erasestr( xloc, yloc, " " );
- gprintf( &xloc, &yloc, "%3.3f", Pv );
- setfillstyle( SOLID_FILL, getbkcolor() );
- bar( InitX+92, InitY+200, InitX+87, InitY+40 );
- setfillstyle( SOLID_FILL, LIGHTRED );
- bar( InitX+92, InitY+(200-(int)(Pv*1.6)), InitX+87, InitY+200 );
- break;
case MV : xloc = InitX+25; yloc = InitY+261;
- erasestr( xloc, yloc, " " );
- gprintf( &xloc, &yloc, "%3.3f", Mv );
- setfillstyle( SOLID_FILL, getbkcolor() );
- bar( InitX+25, InitY+215, InitX+95, InitY+220 );
- setfillstyle( SOLID_FILL, LIGHTRED );
- bar( InitX+25, InitY+215, InitX+25+(int)(Mv*0.7), InitY+220 );
- break;
case SP : xloc = InitX+25; yloc = InitY+275;
- erasestr( xloc, yloc, " " );
- gprintf( &xloc, &yloc, "%3.3f", Sp );

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

-   setfillstyle( SOLID_FILL, getbkcolor() );
-   bar( InitX+28, InitY+200, InitX+33, InitY+40 );
-   setfillstyle( SOLID_FILL, LIGHTRED );
-   bar( InitX+28, InitY+(200-(int)(Sp*1.6)), InitX+33, InitY+200 );
-   break;
case KP : FKp = (Kp%0x100)>>4;
-   FKp = FKp/0x10;
-   FKp += Kp/0x100;
-   xloc = InitX+25; yloc = InitY+289;
-   erasestr( xloc, yloc, "      " );
-   gprintf( &xloc, &yloc, "%3.3f", FKp ); break;
case TI : FTi = (Ti%0x10);
-   FTi = FTi/0x10;
-   FTi += Ti/0x10;
-   xloc = InitX+25; yloc = InitY+303;
-   erasestr( xloc, yloc, "      " );
-   gprintf( &xloc, &yloc, "%4.3f", FTi ); break;
case TD : FTd = (Td%0x10);
-   FTd = FTd/0x10;
-   FTd += Td/0x10;
-   xloc = InitX+25; yloc = InitY+317;
-   erasestr( xloc, yloc, "      " );
-   gprintf(&xloc, &yloc, "%4.3f", FTd ); break;
case CON: xloc = InitX+35; yloc = InitY+331;
-   if(Con==DIR) gprintf( &xloc, &yloc, "DIR" );
-   else gprintf( &xloc, &yloc, "REV" );
case MOD: xloc = InitX+35; yloc = InitY+345;
-   switch(Mode)
-   -
-   case MNU: gprintf( &xloc, &yloc, "MNU" ); break;
-   case PI: gprintf( &xloc, &yloc, "PI" ); break;
-   case PD: gprintf( &xloc, &yloc, "PD" ); break;
-   case PID: gprintf( &xloc, &yloc, "PID" ); break;
-
-
-   gmouse.Mshow(TRUE);

void DataHit( Lcontrol* l )

Mstatus P = gmouse.Mpressed( ButtonL );
if( page==1 )

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for( int i=1; i<=4; i++ )

    if( TPos( P.xaxis, (120*(i-1)+100), (120*(i-1)+210) ) )

        for( int j=1; j<=6; j++ )
            if( TPos( P.yaxis, (14*(j-1)+360), (14*(j)+ 360) ) )

                _l[a[loop]].Hdata(0);
                _l[a[i]].Hdata(j);
                _dcon = j;
                _loop = i;

```

```

if( page==2 )
for( int i=1; i<=4; i++ )

    if( TPos( P.xaxis, (120*(i-1)+100), (120*(i-1)+210) ) )

        for( int j=1; j<=6; j++ )
            if( TPos( P.yaxis, (14*(j-1)+360), (14*(j)+ 360) ) )

                _l[a[loop]].Hdata(0);
                _l[a[i+4]].Hdata(j);
                _dcon = j;
                _loop = i+4;

```

```
void Bkground()
```

```

int xloc, yloc;
bkgdpull.BkPull(2,2,637,477,2);
bkgdpush.BkPush(150,25,340,45,3);
settextstyle( DEFAULT_FONT, HORIZ_DIR, 2 );
settextjustify( CENTER_TEXT, CENTER_TEXT );
setcolor(DARKGRAY);
outtextxy( 320, 45, "PID CONTROLER" );
settextstyle( DEFAULT_FONT, HORIZ_DIR, 1 );
settextjustify( LEFT_TEXT, CENTER_TEXT );
setcolor( DARKGRAY );

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for( int n=1; n<=8; n++ )
```

```
    bkgdpush.BkPull( 5, 90+(n*16), 50, 12, 2 );
    outtextxy( 7, 97+(n*16), "LOOP" );
    xloc = 40; yloc = 97+(n*16);
    gprintf( &xloc, &yloc, "%d", n );
```

```
    bkbutton[10].CreatePull( 590, 90, 40, 15, 2, "NEXT" );
    bkbutton[11].CreatePull( 590, 115, 40, 15, 2, "TNC" );
    bkbutton[12].CreatePull( 590, 140, 40, 15, 2, "DEC" );
    bkbutton[13].CreatePull( 590, 165, 40, 15, 2, "SPINC");
    bkbutton[14].CreatePull( 590, 190, 40, 15, 2, "SPDEC");
    bkbutton[15].CreatePull( 590, 215, 40, 15, 2, "EXIT" );
```

```
void Arrange( Lcontrol* l )
```

```
    gmouse.Mshow(FALSE);
    int n = 1;
    int xloc, yloc;
    settxtjustify( CENTER_TEXT, CENTER_TEXT );
    for( int i=1; i<=8; i++ )
        if( l[i].Power() == on )
            bkbutton[i].CreatePush( 59, 90+(i*16), 31, 12, 2, "ON" );
            l[i].Mem(n);
            a[n]=i;
            n++;

    for( i=1; i<=8; i++ )
        if( l[i].Power() == off )
            bkbutton[i].CreatePull( 59, 90+(i*16), 31, 12, 1, "OFF" );
            l[i].Mem(n);
            a[n]=i;
            n++;
```

```
    gmouse.Mshow(TRUE);
```

```
void ConHit( Lcontrol* l )
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

000001 E000          .ORG $E000
000002 E000          .SEGMENT .MEMORY
000003 4000          .MEMORY
000004 FFFE          .ORG $FFFE
000005 FFFE E44A     .DW START
000006 FFE6          .ORG $FFE6
000007 FFE6 E598     .DW INTOC2
000008 FFD6          .ORG $FFD6
000009 FFD6 E4D8     .DW RECEIVE
000010 0000          .ORG $00
000011 0000          LCAT_ARROW: .RS 1
000012 0001          BUF_KEY: .RS 1
000013 0002          NUM_SEND: .RS 1
000014 0003          BINDEK .RS 2
000015 0005
000016 2000          .ORG $2000
000017 2000          NUM_KEY: .RS 2
000018 2002          HBUF: .RS 2
000019 2004          DBUF: .RS 7
000020 200B          PRO_DATA: .RS 1
000021 200C          F_DATA: .RS 1
000022 200D          LOCATE: .RS 1
000023 200E          STATUS_LCD: .RS 2
000024 2010          STATUS_BUF_DIS: .RS 2
000025 2012          INDEXX: .RS 2
000026 2014          DATA: .RS 2
000027 2016          BUF_DIS: .RS 2
000028 2018          BUF_DIS1: .RS 2
000029 201A          BUF_DISX_T1: .RS 2
000030 201C          SAVE: .RS 1
000031 201D          SAVE_INNER0: .RS 1
000032 201E          SAVE_INNER1: .RS 1
000033 201F          SAVE_INNER2: .RS 1
000034 2020          SAVE_INNER3: .RS 1
000035 2021          SAVE_INNER4: .RS 1
000036 2022          SAVE_INNER5: .RS 1
000037 2023          SAVE_INNER6: .RS 1
000038 2024          SAVE_INNER7: .RS 1
000039 2025          SAVE_INNER8: .RS 1
000040 2026          SAVE_INNER9: .RS 1
000041 2027          SAVE_INNER10: .RS 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|        |      |                 |        |
|--------|------|-----------------|--------|
| 000042 | 2028 | SAVE_INNER11:RS | 1      |
| 000043 | 2029 | SAVE_INNER12:RS | 1      |
| 000044 | 202A | SAVE_INNER13:RS | 1      |
| 000045 | 202B |                 |        |
| 000046 | 202B | BUF_DIS1_T1:RS  | 64     |
| 000047 | 206B | BUF_DIS1_T2:RS  | 64     |
| 000048 | 20AB |                 |        |
| 000049 | 20AB | BUF_DIS2_T1:RS  | 64     |
| 000050 | 20EB | BUF_DIS2_T2:RS  | 64     |
| 000051 | 212B |                 |        |
| 000052 | 212B | BUF_DIS3_T1:RS  | 64     |
| 000053 | 216B | BUF_DIS3_T2:RS  | 64     |
| 000054 | 21AB |                 |        |
| 000055 | 21AB | BUF_DIS4_T1:RS  | 64     |
| 000056 | 21EB | BUF_DIS4_T2:RS  | 64     |
| 000057 | 222B |                 |        |
| 000058 | 222B | BUF_DIS5_T1:RS  | 64     |
| 000059 | 226B | BUF_DIS5_T2:RS  | 64     |
| 000060 | 22AB |                 |        |
| 000061 | 22AB | BUF_DIS6_T1:RS  | 64     |
| 000062 | 22EB | BUF_DIS6_T2:RS  | 64     |
| 000063 | 232B |                 |        |
| 000064 | 232B | BUF_DIS7_T1:RS  | 64     |
| 000065 | 236B | BUF_DIS7_T2:RS  | 64     |
| 000066 | 23AB |                 |        |
| 000067 | 23AB | BUF_DIS8_T1:RS  | 64     |
| 000068 | 23EB | BUF_DIS8_T2:RS  | 64     |
| 000069 | 242B |                 |        |
| 000070 | 242B | BUF_SAVE:RS     | 64     |
| 000071 | 246B |                 |        |
| 000072 | 3600 | .ORG            | \$3600 |
| 000073 | 3600 | FUNC_SAVE:RS    | \$200  |
| 000074 | 3800 |                 |        |
| 000075 | 3800 |                 |        |
| 000076 | 3700 | .ORG            | \$3700 |
| 000077 | 3700 | BUF_ACTIVE:RS   | \$1    |
| 000078 | 3800 | .ORG            | \$3800 |
| 000079 | 3800 | PAGE_BUFFER:RS  | \$800  |
| 000080 | 4000 |                 |        |
| 000082 | 4000 |                 |        |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

000083 3800 .EQU INDEX1,$3800 ; PAGE BUFFER OF
LOOP1
000084 3900 .EQU INDEX2,$3900 ; PAGE BUFFER OF
LOOP2
000085 3A00 .EQU INDEX3,$3A00 ; PAGE BUFFER OF
LOOP3
000086 3B00 .EQU INDEX4,$3B00 ; PAGE BUFFER OF
LOOP4
000087 3C00 .EQU INDEX5,$3C00 ; PAGE BUFFER OF
LOOP5
000088 3D00 .EQU INDEX6,$3D00 ; PAGE BUFFER OF
LOOP6
000089 3E00 .EQU INDEX7,$3E00 ; PAGE BUFFER OF
LOOP7
000090 3F00 .EQU INDEX8,$3F00 ; PAGE BUFFER OF
LOOP8
000091 4000 ;
*****
000092 0000 .EQU SGN1,$00 ;LIVE IN PAGE
BUFFER
000093 0080 .EQU S_DCD,10000000b
000094 0040 .EQU S_DCI,01000000b
000095 0020 .EQU S_DCP,00100000b
000096 0004 .EQU S_DDE,00000100b
000097 0002 .EQU S_DE ,00000010b
000098 0001 .EQU S_CON ,00000001b
000099 0001 .EQU MODE , $01
000100 0002 .EQU KP , $02 ;2 BYTE
000101 0004 .EQU TI , $04 ;2 byte
000102 0006 .EQU TD , $06 ;2 byte
000103 0008 .EQU DSP , $08
000104 0009 .EQU ACT , $09
000105 000A .EQU DV , $0A ;IT IS PV
000106 000B .EQU MV , $0B ;DATA OF
TRANSFER
000107 4000
000108 000C .EQU SGN2 , $0C
000109 0004 .EQU S_DCPI,00000100b
000110 0002 .EQU S_DCPD ,00000010b
000111 0001 .EQU S_DCPI ,00000001b
000112 000D .EQU DVO , $0D

```

```

000113 000E .EQU DCP , $0E
000114 000F .EQU F_DCP , $0F
000115 0010 .EQU DCI , $10
000116 0011 .EQU F_DCI , $11
000117 0012 .EQU DCD , $12
000118 0013 .EQU F_DCD , $13
000119 0014 .EQU DCPI , $14
000120 0015 .EQU F_DCPI , $15
000121 0016 .EQU DCPD , $16
000122 0017 .EQU DCPID , $17
000123 4000
000124 0018 .EQU DE , $18
000125 0019 .EQU DDE , $19
000126 001A .EQU SUM , $1A ;2 byte
000127 001C .EQU BUF_2P , $1C ;1 byte
000128 001D .EQU BUF_1P , $1D ;2 byte
000129 001F .EQU BUF_DE , $1F ;2 byte
000130 0021 .EQU BUF_SUM , $21 ;2 byte
000131 0023 .EQU BUFF_1 , $23 ;4 byte
000132 0028 .EQU BUF_4I , $28 ;1 byte
000133 0029 .EQU BUF_3I , $29 ;2 byte
000134 002B .EQU BUF_2I , $2B ;2 byte
000135 002D .EQU BUF_1I , $2D ;2 byte
000136 002F .EQU BUFF_D , $2F ;2 byte
000137 0031 .EQU BUF_2D , $31 ;1 byte
000138 0032 .EQU BUF_1D , $32 ;2 byte
000139 0034 .EQU BUFF_2D , $34 ;1 byte
000140 0035 .EQU BUFF_1D , $35 ;2 byte
000141 0038 .EQU BUF_4I , $38 ;2 BYTE
000142 4000
000143 4000
000144 4000
000145 4000
;*****
000146 0080 .EQU BIAS0 , 128
000147 0007 .EQU DDRD , $07 ;LIVE IN ADDRESS
1000h-103Fh
_000148 0008 .EQU PORTD , $08
000149 0009 .EQU DDRD , $09
000150 000A .EQU PORTE , $0A
000151 000B .EQU CFORC , $0B

```

|             |                     |
|-------------|---------------------|
| 000152 000C | .EQU OC1M , \$0C    |
| 000153 000D | .EQU OC1D , \$0D    |
| 000154 000E | .EQU TCNT , \$0E    |
| 000155 0010 | .EQU TIC1 , \$10    |
| 000156 0012 | .EQU TIC2 , \$12    |
| 000157 0014 | .EQU TIC3 , \$14    |
| 000158 0016 | .EQU TOC1 , \$16    |
| 000159 0018 | .EQU TOC2 , \$18    |
| 000160 001A | .EQU TOC3 , \$1A    |
| 000161 001C | .EQU TOC4 , \$1C    |
| 000162 001E | .EQU TOC5 , \$1E    |
| 000163 0020 | .EQU TCTL1 , \$20   |
| 000164 0021 | .EQU TCTL2 , \$21   |
| 000165 0022 | .EQU TMSK1 , \$22   |
| 000166 0023 | .EQU TFLG1 , \$23   |
| 000167 0024 | .EQU TMSK2 , \$24   |
| 000168 0025 | .EQU TFLG2 , \$25   |
| 000169 0026 | .EQU PACTL , \$26   |
| 000170 0027 | .EQU PACNT , \$27   |
| 000171 0028 | .EQU SPCR , \$28    |
| 000172 0029 | .EQU SPSR , \$29    |
| 000173 002A | .EQU SPDR , \$2A    |
| 000174 002B | .EQU BAUD , \$2B    |
| 000175 002C | .EQU SCCR1 , \$2C   |
| 000176 002D | .EQU SCCR2 , \$2D   |
| 000177 002E | .EQU SCSR , \$2E    |
| 000178 002F | .EQU SCDR , \$2F    |
| 000179 0030 | .EQU ADCTL , \$30   |
| 000180 0031 | .EQU ADR1 , \$31    |
| 000181 0032 | .EQU ADR2 , \$32    |
| 000182 0033 | .EQU ADR3 , \$33    |
| 000183 0034 | .EQU ADR4 , \$34    |
| 000184 0039 | .EQU OPTION , \$39  |
| 000185 003A | .EQU COPRST , \$3A  |
| 000186 003C | .EQU HPRIO , \$3C   |
| 000187 4000 | ; .EQU INIT , \$3D  |
| 000188 003F | .EQU CONFIG , \$3F  |
| 000189 1400 | .EQU SET0 , \$1400  |
| 000190 1402 | .EQU BUSY , \$1402  |
| 000191 1401 | .EQU WRITE , \$1401 |
| 000192 0140 | .EQU READ , \$140   |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                  |                           |
|------------------|---------------------------|
| 000193 4000      |                           |
| 000194 1000      | .EQU REGBAS,\$1000        |
| 000195 4000      |                           |
| 000196 4000      |                           |
| 000197 E000      | .CODE                     |
| 000198 E000      |                           |
| 000199 E000 380A | T_DATA_TRAN:.DW INDEX1+DV |
| 000200 E002 380B | .DW INDEX1+MV             |
| 000201 E004 390A | .DW INDEX2+DV             |
| 000202 E006 390B | .DW INDEX2+MV             |
| 000203 E008 3A0A | .DW INDEX3+DV             |
| 000204 E00A 3A0B | .DW INDEX3+MV             |
| 000205 E00C 3B0A | .DW INDEX4+DV             |
| 000206 E00E 3B0B | .DW INDEX4+MV             |
| 000207 E010 3C0A | .DW INDEX5+DV             |
| 000208 E012 3C0B | .DW INDEX5+MV             |
| 000209 E014 3D0A | .DW INDEX6+DV             |
| 000210 E016 3D0B | .DW INDEX6+MV             |
| 000211 E018 3E0A | .DW INDEX7+DV             |
| 000212 E01A 3E0B | .DW INDEX7+MV             |
| 000213 E01C 3F0A | .DW INDEX8+DV             |
| 000214 E01E 3F0B | .DW INDEX8+MV             |
| 000215 E020 3800 | .DW INDEX1+SGN 1          |
| 000216 E022 3900 | .DW INDEX2+SGN 1          |
| 000217 E024 3A00 | .DW INDEX3+SGN 1          |
| 000218 E026 3B00 | .DW INDEX4+SGN 1          |
| 000219 E028 3C00 | .DW INDEX5+SGN 1          |
| 000220 E02A 3D00 | .DW INDEX6+SGN 1          |
| 000221 E02C 3E00 | .DW INDEX7+SGN 1          |
| 000222 E02E 3F00 | .DW INDEX8+SGN 1          |
| 000223 E030 3801 | .DW INDEX1+MODE           |
| 000224 E032 3802 | .DW INDEX1+KP             |
| 000225 E034 3803 | .DW INDEX1+KP+1           |
| 000226 E036 3804 | .DW INDEX1+TI             |
| 000227 E038 3805 | .DW INDEX1+TI+1           |
| 000228 E03A 3806 | .DW INDEX1+TD             |
| 000229 E03C 3807 | .DW INDEX1+TD+1           |
| 000230 E03E 3808 | .DW INDEX1+DSP            |
| 000231 E040 3809 | .DW INDEX1+ACT            |
| 000232 E042 3901 | .DW INDEX2+MODE           |
| 000233 E044 3902 | .DW INDEX2+KP             |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                  |                 |
|------------------|-----------------|
| 000234 E046 3903 | .DW INDEX2+KP+1 |
| 000235 E048 3904 | .DW INDEX2+TI   |
| 000236 E04A 3905 | .DW INDEX2+TI+1 |
| 000237 E04C 3906 | .DW INDEX2+TD   |
| 000238 E04E 3907 | .DW INDEX2+TD+1 |
| 000239 E050 3908 | .DW INDEX2+DSP  |
| 000240 E052 3909 | .DW INDEX2+ACT  |
| 000241 E054 3A01 | .DW INDEX3+MODE |
| 000242 E056 3A02 | .DW INDEX3+KP   |
| 000243 E058 3A03 | .DW INDEX3+KP+1 |
| 000244 E05A 3A04 | .DW INDEX3+TI   |
| 000245 E05C 3A05 | .DW INDEX3+TI+1 |
| 000246 E05E 3A06 | .DW INDEX3+TD   |
| 000247 E060 3A07 | .DW INDEX3+TD+1 |
| 000248 E062 3A08 | .DW INDEX3+DSP  |
| 000249 E064 3A09 | .DW INDEX3+ACT  |
| 000250 E066 3B01 | .DW INDEX4+MODE |
| 000251 E068 3B02 | .DW INDEX4+KP   |
| 000252 E06A 3B03 | .DW INDEX4+KP+1 |
| 000253 E06C 3B04 | .DW INDEX4+TI   |
| 000254 E06E 3B05 | .DW INDEX4+TI+1 |
| 000255 E070 3B06 | .DW INDEX4+TD   |
| 000256 E072 3B07 | .DW INDEX4+TD+1 |
| 000257 E074 3B08 | .DW INDEX4+DSP  |
| 000258 E076 3B09 | .DW INDEX4+ACT  |
| 000259 E078 3C01 | .DW INDEX5+MODE |
| 000260 E07A 3C02 | .DW INDEX5+KP   |
| 000261 E07C 3C03 | .DW INDEX5+KP+1 |
| 000262 E07E 3C04 | .DW INDEX5+TI   |
| 000263 E080 3C05 | .DW INDEX5+TI+1 |
| 000264 E082 3C06 | .DW INDEX5+TD   |
| 000265 E084 3C07 | .DW INDEX5+TD+1 |
| 000266 E086 3C08 | .DW INDEX5+DSP  |
| 000267 E088 3C09 | .DW INDEX5+ACT  |
| 000268 E08A 3D01 | .DW INDEX6+MODE |
| 000269 E08C 3D02 | .DW INDEX6+KP   |
| 000270 E08E 3D03 | .DW INDEX6+KP+1 |
| 000271 E090 3D04 | .DW INDEX6+TI   |
| 000272 E092 3D05 | .DW INDEX6+TI+1 |
| 000273 E094 3D06 | .DW INDEX6+TD   |
| 000274 E096 3D07 | .DW INDEX6+TD+1 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                   |                                    |
|-------------------|------------------------------------|
| 000275 E098 3D08  | .DW INDEX6+DSP                     |
| 000276 E09A 3D09  | .DW INDEX6+ACT                     |
| 000277 E09C 3E01  | .DW INDEX7+MODE                    |
| 000278 E09E 3E02  | .DW INDEX7+KP                      |
| 000279 E0A0 3E03  | .DW INDEX7+KP+1                    |
| _000280 E0A2 3E04 | .DW INDEX7+TI                      |
| 000281 E0A4 3E05  | .DW INDEX7+TI+1                    |
| 000282 E0A6 3E06  | .DW INDEX7+TD                      |
| 000283 E0A8 3E07  | .DW INDEX7+TD+1                    |
| 000284 E0AA 3E08  | .DW INDEX7+DSP                     |
| 000285 E0AC 3E09  | .DW INDEX7+ACT                     |
| 000286 E0AE 3F01  | .DW INDEX8+MODE                    |
| 000287 E0B0 3F02  | .DW INDEX8+KP                      |
| 000288 E0B2 3F03  | .DW INDEX8+KP+1                    |
| 000289 E0B4 3F04  | .DW INDEX8+TI                      |
| 000290 E0B6 3F05  | .DW INDEX8+TI+1                    |
| 000291 E0B8 3F06  | .DW INDEX8+TD                      |
| 000292 E0BA 3F07  | .DW INDEX8+TD+1                    |
| 000293 E0BC 3F08  | .DW INDEX8+DSP                     |
| 000294 E0BE 3F09  | .DW INDEX8+ACT                     |
| 000295 E0C0       |                                    |
| 000296 E0C0       | *****                              |
| 000297 E0C0       | ***** TABLE HEXADECIMAL TO DECIMAL |
| *****             |                                    |
| 000298 E0C0       | ***** OF SP,MV AND PV *****        |
| 000299 E0C0       | *****                              |
| 000300 E0C0       |                                    |
| 000301 E0C0 0000  | T_HTOD: .DW \$0000                 |
| 000302 E0C2 0004  | .DW \$0004                         |
| 000303 E0C4 0008  | .DW \$0008                         |
| 000304 E0C6 0012  | .DW \$0012                         |
| 000305 E0C8 0016  | .DW \$0016                         |
| 000306 E0CA 0020  | .DW \$0020                         |
| 000307 E0CC 0023  | .DW \$0023                         |
| 000308 E0CE 0027  | .DW \$0027                         |
| 000309 E0D0 0031  | .DW \$0031                         |
| 000310 E0D2 0035  | .DW \$0035                         |
| 000311 E0D4 0039  | .DW \$0039                         |
| 000312 E0D6 0043  | .DW \$0043                         |
| 000313 E0D8 0047  | .DW \$0047                         |
| 000314 E0DA 0051  | .DW \$0051                         |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                   |            |
|-------------------|------------|
| 000315 E0DC 0055  | .DW \$0055 |
| 000316 E0DE 0059  | .DW \$0059 |
| 000317 E0E0 0063  | .DW \$0063 |
| 000318 E0E2 0066  | .DW \$0066 |
| 000319 E0E4 0070  | .DW \$0070 |
| 000320 E0E6 0074  | .DW \$0074 |
| 000321 E0E8 0078  | .DW \$0078 |
| 000322 E0EA 0082  | .DW \$0082 |
| 000323 E0EC 0086  | .DW \$0086 |
| 000324 E0EE 0090  | .DW \$0090 |
| 000325 E0F0 0094  | .DW \$0094 |
| 000326 E0F2 0098  | .DW \$0098 |
| 000327 E0F4 0102  | .DW \$0102 |
| 000328 E0F6 0105  | .DW \$0105 |
| 000329 E0F8 0109  | .DW \$0109 |
| 000330 E0FA 0113  | .DW \$0113 |
| 000331 E0FC 0117  | .DW \$0117 |
| 000332 E0FE 0121  | .DW \$0121 |
| 000333 E100 0125  | .DW \$0125 |
| 000334 E102 0129  | .DW \$0129 |
| 000335 E104 0133  | .DW \$0133 |
| 000336 E106 0137  | .DW \$0137 |
| 000337 E108 0141  | .DW \$0141 |
| 000338 E10A 0145  | .DW \$0145 |
| 000339 E10C 0148  | .DW \$0148 |
| 000340 E10E 0152  | .DW \$0152 |
| 000341 E110 0156  | .DW \$0156 |
| 000342 E112 0160  | .DW \$0160 |
| 000343 E114 0164  | .DW \$0164 |
| 000344 E116 0168  | .DW \$0168 |
| 000345 E118 0172  | .DW \$0172 |
| _000346 E11A 0176 | .DW \$0176 |
| 000347 E11C 0180  | .DW \$0180 |
| 000348 E11E 0184  | .DW \$0184 |
| 000349 E120 0188  | .DW \$0188 |
| 000350 E122 0191  | .DW \$0191 |
| 000351 E124 0195  | .DW \$0195 |
| 000352 E126 0199  | .DW \$0199 |
| 000353 E128 0203  | .DW \$0203 |
| 000354 E12A 0207  | .DW \$0207 |
| 000355 E12C 0211  | .DW \$0211 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                  |            |
|------------------|------------|
| 000356 E12E 0215 | .DW \$0215 |
| 000357 E130 0219 | .DW \$0219 |
| 000358 E132 0223 | .DW \$0223 |
| 000359 E134 0227 | .DW \$0227 |
| 000360 E136 0231 | .DW \$0231 |
| 000361 E138 0235 | .DW \$0235 |
| 000362 E13A 0239 | .DW \$0239 |
| 000363 E13C 0243 | .DW \$0243 |
| 000364 E13E 0247 | .DW \$0247 |
| 000365 E140 0251 | .DW \$0251 |
| 000366 E142 0255 | .DW \$0255 |
| 000367 E144 0259 | .DW \$0259 |
| 000368 E146 0263 | .DW \$0263 |
| 000369 E148 0267 | .DW \$0267 |
| 000370 E14A 0271 | .DW \$0271 |
| 000371 E14C 0275 | .DW \$0275 |
| 000372 E14E 0279 | .DW \$0279 |
| 000373 E150 0283 | .DW \$0283 |
| 000374 E152 0287 | .DW \$0287 |
| 000375 E154 0291 | .DW \$0291 |
| 000376 E156 0295 | .DW \$0295 |
| 000377 E158 0299 | .DW \$0299 |
| 000378 E15A 0303 | .DW \$0303 |
| 000379 E15C 0307 | .DW \$0307 |
| 000380 E15E 0311 | .DW \$0311 |
| 000381 E160 0315 | .DW \$0315 |
| 000382 E162 0319 | .DW \$0319 |
| 000383 E164 0323 | .DW \$0323 |
| 000384 E166 0327 | .DW \$0327 |
| 000385 E168 0331 | .DW \$0331 |
| 000386 E16A 0335 | .DW \$0335 |
| 000387 E16C 0339 | .DW \$0339 |
| 000388 E16E 0343 | .DW \$0343 |
| 000389 E170 0347 | .DW \$0347 |
| 000390 E172 0351 | .DW \$0351 |
| 000391 E174 0355 | .DW \$0355 |
| 000392 E176 0359 | .DW \$0359 |
| 000393 E178 0363 | .DW \$0363 |
| 000394 E17A 0367 | .DW \$0367 |
| 000395 E17C 0371 | .DW \$0371 |
| 000396 E17E 0375 | .DW \$0375 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                  |            |
|------------------|------------|
| 000397 E180 0379 | .DW \$0379 |
| 000398 E182 0383 | .DW \$0383 |
| 000399 E184 0387 | .DW \$0387 |
| 000400 E186 0391 | .DW \$0391 |
| 000401 E188 0395 | .DW \$0395 |
| 000402 E18A 0398 | .DW \$0398 |
| 000403 E18C 0402 | .DW \$0402 |
| 000404 E18E 0406 | .DW \$0406 |
| 000405 E190 0410 | .DW \$0410 |
| 000406 E192 0414 | .DW \$0414 |
| 000407 E194 0418 | .DW \$0418 |
| 000408 E196 0422 | .DW \$0422 |
| 000409 E198 0426 | .DW \$0426 |
| 000410 E19A 0430 | .DW \$0430 |
| 000411 E19C 0434 | .DW \$0434 |
| 000412 E19E 0438 | .DW \$0438 |
| 000413 E1A0 0441 | .DW \$0441 |
| 000414 E1A2 0445 | .DW \$0445 |
| 000415 E1A4 0449 | .DW \$0449 |
| 000416 E1A6 0453 | .DW \$0453 |
| 000417 E1A8 0457 | .DW \$0457 |
| 000418 E1AA 0461 | .DW \$0461 |
| 000419 E1AC 0465 | .DW \$0465 |
| 000420 E1AE 0469 | .DW \$0469 |
| 000421 E1B0 0473 | .DW \$0473 |
| 000422 E1B2 0477 | .DW \$0477 |
| 000423 E1B4 0480 | .DW \$0480 |
| 000424 E1B6 0484 | .DW \$0484 |
| 000425 E1B8 0488 | .DW \$0488 |
| 000426 E1BA 0492 | .DW \$0492 |
| 000427 E1BC 0496 | .DW \$0496 |
| 000428 E1BE 0500 | .DW \$0500 |
| 000429 E1C0 0504 | .DW \$0504 |
| 000430 E1C2 0508 | .DW \$0508 |
| 000431 E1C4 0512 | .DW \$0512 |
| 000432 E1C6 0516 | .DW \$0516 |
| 000433 E1C8 0520 | .DW \$0520 |
| 000434 E1CA 0523 | .DW \$0523 |
| 000435 E1CC 0527 | .DW \$0527 |
| 000436 E1CE 0531 | .DW \$0531 |
| 000437 E1D0 0535 | .DW \$0535 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                   |            |
|-------------------|------------|
| 000438 E1D2 0539  | .DW \$0539 |
| 000439 E1D4 0543  | .DW \$0543 |
| 000440 E1D6 0547  | .DW \$0547 |
| 000441 E1D8 0551  | .DW \$0551 |
| 000442 E1DA 0555  | .DW \$0555 |
| 000443 E1DC 0559  | .DW \$0559 |
| 000444 E1DE 0569  | .DW \$0569 |
| 000445 E1E0 0566  | .DW \$0566 |
| 000446 E1E2 0570  | .DW \$0570 |
| 000447 E1E4 0574  | .DW \$0574 |
| 000448 E1E6 0578  | .DW \$0578 |
| 000449 E1E8 0582  | .DW \$0582 |
| 000450 E1EA 0586  | .DW \$0586 |
| 000451 E1EC 0590  | .DW \$0590 |
| 000452 E1EE 0594  | .DW \$0594 |
| 000453 E1F0 0598  | .DW \$0598 |
| 000454 E1F2 0602  | .DW \$0602 |
| 000455 E1F4 0605  | .DW \$0605 |
| 000456 E1F6 0609  | .DW \$0609 |
| 000457 E1F8 0613  | .DW \$0613 |
| 000458 E1FA 0617  | .DW \$0617 |
| 000459 E1FC 0621  | .DW \$0621 |
| 000460 E1FE 0625  | .DW \$0625 |
| 000461 E200 0629  | .DW \$0629 |
| 000462 E202 0633  | .DW \$0633 |
| 000463 E204 0637  | .DW \$0637 |
| 000464 E206 0641  | .DW \$0641 |
| 000465 E208 0645  | .DW \$0645 |
| 000466 E20A 0648  | .DW \$0648 |
| 000467 E20C 0652  | .DW \$0652 |
| 000468 E20E 0656  | .DW \$0656 |
| 000469 E210 0660  | .DW \$0660 |
| 000470 E212 0664  | .DW \$0664 |
| 000471 E214 0668  | .DW \$0668 |
| 000472 E216 0672  | .DW \$0672 |
| 000473 E218 0676  | .DW \$0676 |
| 000474 E21A 0680  | .DW \$0680 |
| 000475 E21C 0684  | .DW \$0684 |
| 000476 E21E 0688  | .DW \$0688 |
| 000477 E220 0691  | .DW \$0691 |
| _000478 E222 0695 | .DW \$0695 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                  |            |
|------------------|------------|
| 000479 E224 0699 | .DW \$0699 |
| 000480 E226 0703 | .DW \$0703 |
| 000481 E228 0707 | .DW \$0707 |
| 000482 E22A 0711 | .DW \$0711 |
| 000483 E22C 0715 | .DW \$0715 |
| 000484 E22E 0719 | .DW \$0719 |
| 000485 E230 0723 | .DW \$0723 |
| 000486 E232 0727 | .DW \$0727 |
| 000487 E234 0730 | .DW \$0730 |
| 000488 E236 0734 | .DW \$0734 |
| 000489 E238 0738 | .DW \$0738 |
| 000490 E23A 0742 | .DW \$0742 |
| 000491 E23C 0746 | .DW \$0746 |
| 000492 E23E 0750 | .DW \$0750 |
| 000493 E240 0754 | .DW \$0754 |
| 000494 E242 0758 | .DW \$0758 |
| 000495 E244 0762 | .DW \$0762 |
| 000496 E246 0766 | .DW \$0766 |
| 000497 E248 0770 | .DW \$0770 |
| 000498 E24A 0773 | .DW \$0773 |
| 000499 E24C 0777 | .DW \$0777 |
| 000500 E24E 0781 | .DW \$0781 |
| 000501 E250 0785 | .DW \$0785 |
| 000502 E252 0789 | .DW \$0789 |
| 000503 E254 0793 | .DW \$0793 |
| 000504 E256 0797 | .DW \$0797 |
| 000505 E258 0801 | .DW \$0801 |
| 000506 E25A 0805 | .DW \$0805 |
| 000507 E25C 0809 | .DW \$0809 |
| 000508 E25E 0813 | .DW \$0813 |
| 000509 E260 0816 | .DW \$0816 |
| 000510 E262 0820 | .DW \$0820 |
| 000511 E264 0824 | .DW \$0824 |
| 000512 E266 0828 | .DW \$0828 |
| 000513 E268 0832 | .DW \$0832 |
| 000514 E26A 0836 | .DW \$0836 |
| 000515 E26C 0840 | .DW \$0840 |
| 000516 E26E 0844 | .DW \$0844 |
| 000517 E270 0848 | .DW \$0848 |
| 000518 E272 0852 | .DW \$0852 |
| 000519 E274 0855 | .DW \$0855 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                  |            |
|------------------|------------|
| 000520 E276 0859 | .DW \$0859 |
| 000521 E278 0863 | .DW \$0863 |
| 000522 E27A 0867 | .DW \$0867 |
| 000523 E27C 0871 | .DW \$0871 |
| 000524 E27E 0875 | .DW \$0875 |
| 000525 E280 0879 | .DW \$0879 |
| 000526 E282 0883 | .DW \$0883 |
| 000527 E284 0887 | .DW \$0887 |
| 000528 E286 0891 | .DW \$0891 |
| 000529 E288 0895 | .DW \$0895 |
| 000530 E28A 0898 | .DW \$0898 |
| 000531 E28C 0902 | .DW \$0902 |
| 000532 E28E 0906 | .DW \$0906 |
| 000533 E290 0910 | .DW \$0910 |
| 000534 E292 0914 | .DW \$0914 |
| 000535 E294 0918 | .DW \$0918 |
| 000536 E296 0922 | .DW \$0922 |
| 000537 E298 0926 | .DW \$0926 |
| 000538 E29A 0930 | .DW \$0930 |
| 000539 E29C 0934 | .DW \$0934 |
| 000540 E29E 0938 | .DW \$0938 |
| 000541 E2A0 0941 | .DW \$0941 |
| 000542 E2A2 0945 | .DW \$0945 |
| 000543 E2A4 0949 | .DW \$0949 |
| 000544 E2A6 0953 | .DW \$0953 |
| 000545 E2A8 0957 | .DW \$0957 |
| 000546 E2AA 0961 | .DW \$0961 |
| 000547 E2AC 0965 | .DW \$0965 |
| 000548 E2AE 0969 | .DW \$0969 |
| 000549 E2B0 0973 | .DW \$0973 |
| 000550 E2B2 0977 | .DW \$0977 |
| 000551 E2B4 0980 | .DW \$0980 |
| 000552 E2B6 0984 | .DW \$0984 |
| 000553 E2B8 0988 | .DW \$0988 |
| 000554 E2BA 0992 | .DW \$0992 |
| 000555 E2BC 0996 | .DW \$0996 |
| 000556 E2BE 1000 | .DW \$1000 |
| 000557 E2C0      |            |
| 000558 E2C0      |            |
| 000559 E2C0      |            |
| *****            |            |

\*\*\*\*\*

\*\*\*\*\* TABLE HEXADECIMAL TO DECIMAL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

000560 E2C0 ;***** OF FLOAT OF KP,TI AND TD *****
000561 E2C0 ;*****
000562 E2C0
000563 E2C0 0000 TF_HTOD .DW $0000
000564 E2C2 0062 .DW $0062
000565 E2C4 0125 .DW $0125
000566 E2C6 0187 .DW $0187
000567 E2C8 0250 .DW $0250
000568 E2CA 0312 .DW $0312
000569 E2CC 0375 .DW $0375
000570 E2CE 0437 .DW $0437
000571 E2D0 0500 .DW $0500
000572 E2D2 0562 .DW $0562
000573 E2D4 0625 .DW $0625
000574 E2D6 0687 .DW $0687
000575 E2D8 0750 .DW $0750
000576 E2DA 0812 .DW $0812
000577 E2DC 0875 .DW $0875
000578 E2DE 0937 .DW $0937
000579 E2E0
000580 E2E0 4C4F41442020 LOAD_: .DB "LOAD "
000581 E2E6 534156452020 SAVE_: .DB "SAVE "
000582 E2EC 43414E434C45 CANCEL_: .DB "CANCEL"
000583 E2F2
000584 E2F2 4D4E55 MNU: .DB "MNU"
000585 E2F5 504920 PI: .DB "PI"
000586 E2F8 504420 PD: .DB "PD"
000587 E2FB 504944 PID: .DB "PID"
000588 E2FE
000589 E2FE 444952 DIR: .DB "DIR"
000590 E301 524556 REV: .DB "REV"
000591 E304
000592 E304 204F4E ON: .DB "ON"
000593 E307 4F4646 OFF: .DB "OFF"
000594 E30A
000595 E30A 4C4F4F5020202020 TABLE1: .DB "LOOP
E312 20202020202020
000596 E31A 2053503A2020202E .DB "SP: . % "
E322 2020202520202020
000597 E32A 2050563A2020202E .DB "PV: . % "
E332 2020202520202020

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

000598 E33A 204D563A2020202E .DB " MV: . % "  
 E342 2020202520202020  
 000599 E34A  
 000600 E34A 2020202020202020 TABLE2: .DB "  
 E352 2020202020202020  
 000601 E35A 204B503A20202020 .DB " KP: . "  
 E362 2E20202020202020  
 000602 E36A 2054493A20202020 .DB " TI: . S "  
 E372 2E20202020532020  
 \_000603 E37A 2054443A20202020 .DB " TD: . S "  
 E382 2E20202020532020  
 000604 E38A  
 000605 E38A 50524F4455434544 TITLE1: .DB " PRODUCED BY "  
 E392 2042592020202020  
 000606 E39A 532E42554E434841 .DB " S.BUNCHA "  
 E3A2 2020202020202020  
 000607 E3AA 412E505241534954 .DB " A.PRASIT "  
 E3B2 2020202020202020  
 000608 E3BA 532E53414D415741 .DB " S.SAMAWAT "  
 E3C2 5420202020202020  
 000609 E3CA  
 000610 E3CA 2020202020202020 TITLE2: .DB "  
 E3D2 2020202020202020  
 000611 E3DA 2020504944202043 .DB " PID CONTROL "  
 E3E2 4F4E54524F4C2020  
 000612 E3EA 20204D554C544920 .DB " MULTI CHANEL "  
 E3F2 4348414E454C2020  
 000613 E3FA 2020202020202020 .DB "  
 E402 2020202020202020  
 000614 E40A  
 000615 E40A 2046554E433A2020 SAVE\_T: .DB " FUNC: LOOP: "  
 E412 204C4F4F503A2020  
 000616 E41A 2043414E434C4520 .DB " CANCEL "  
 E422 2020202020202020  
 000617 E42A 4E454D453A202020 .DB " NEME: "  
 E432 2020202020202020  
 000618 E43A 2020202020202020 .DB "  
 E442 2020202020202020  
 000619 E44A  
 000620 E44A ;\*\*\*\*\*  
 000621 E44A ;\*\*\*\*\* MAIN PROGRAM \*\*\*\*\*

```

000622 E44A ;*****
000623 E44A ;
000624 E44A 8E00FF START: LDS #$$F
000625 E44D BDE461 JSR INITSYS
000626 E450 BDE494 JSR INITDATA
000627 E453 BDE4B4 JSR SETPIO
000628 E456 LOOP_C:
000629 E456 CE1000 LDX #$$1000
000630 E459 1C2D20 BSET SCCR2,X,$20
000631 E45C BDEAEO JSR SCAN
000632 E45F 20F5 BRA LOOP_C
000633 E461
000634 E461 ;*****
000635 E461 ;***** INITIAL SYSTEM *****
000636 E461 ;*****
000637 E461 CE1000 INITSYS: LDX #REGBAS
000638 E464 8630 LDAA #$$30 ;SET BAUD RATE
9600.
000639 E466 A72B STAA BAUD,X ;64 E cycles
000640 E468 6F2C CLR SCCR1,X ;SET 8BIT DATA
000641 E46A 862C LDAA #$$2C
000642 E46C A72D STAA SCCR2,X
000643 E46E 8601 LDAA #01
000644 E470 A724 STAA TMSK2,X ;SET PRESCAN
000645 E472 1D3903 BCLR OPTION,X,$03
000646 E475 8640 LDAA #$$40
000647 E477 A720 STAA TCTL1,X
000648 E479 A723 STAA TFLG1,X ;Clear flag OC2F
000649 E47B A722 STAA TMSK1,X ;Enable OC2
interrupts
000650 E47D BDE5C7 JSR STARTAD
000651 E480 BDF31A JSR INIT_DIS
000652 E483
000653 E483 8682 LDAA #$$82 ;SET STATUS OF 8255
000654 E485 B71203 STAA $1203
000655 E488 8680 LDAA #$$80
_000656 E48A B71617 STAA $1617
000657 E48D B7161B STAA $161B
000658 E490 B7161F STAA $161F
000659 E493 39 RTS
000660 E494

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

000661 E494
000662 E494
000663 E494
000664 E494
000665 E494 8600
000666 E496 B72000
000667 E499 9700
000668 E49B 9701
000669 E49D B7201C
000670 E4A0 C612
000671 E4A2 8601
000672 E4A4 CE201E
000673 E4A7 A700
000674 E4A9 08
000675 E4AA 5A
000676 E4AB 26F7
000677 E4AD BDE623
000678 E4B0 BDE5D3
000679 E4B3 39
000680 E4B4
000681 E4B4
000682 E4B4
000683 E4B4
000684 E4B4 860C
000685 E4B6 18A73C
000686 E4B9 CEE38A
000687 E4BC FF2018
000688 E4BF BDF8E2
000689 E4C2 860A
000690 E4C4 CEFFFF
000691 E4C7 09
000692 E4C8 26FD
000693 E4CA 4A
000694 E4CB 26F7
000695 E4CD CEE3CA
000696 E4D0 FF2018
000697 E4D3 BDF8E2
000698 E4D6 0E
000699 E4D7 39
000700 E4D8
000701 E4D8

*****
***** INITIAL DATA *****
*****

INITDATA: LDAA #00
          STAA NUM_KEY
          STAA LCAT_ARROW
          STAA BUF_KEY
          STAA SAVE
          LDAB #18
          LDAA #1
L_CLR:   LDX #SAVE_INNER1
          STAA 0,X
          INX
          DECB
          BNE L_CLR
          JSR CLR_PAGE
          JSR SET_VAR8
          RTS

*****
***** SET PRIORITY *****
*****

SETPIO:  LDAA #0C
          STAA HPRI0,Y
          LDX #TITLE1
          STX BUF_DIS1
          JSR OUT_DIS
          LDAA #0A
LP_DTT1: LDX #0FFF
LP_DTT2: DEX
          BNE LP_DTT2
          DECA
          BNE LP_DTT1
          LDX #TITLE2
          STX BUF_DIS1
          JSR OUT_DIS
CLI      ;Enable Interrupts
RTS

```

```

000702 E4D8 ;*****
000703 E4D8 ;***** Subroutine cross talk with pc *****
000704 E4D8 ;*****
000705 E4D8 0F RECEIVE: SEI
000706 E4D9 37 PSHB
000707 E4DA 3C PSHX
000708 E4DB 183C PSHY
000709 E4DD CE1000 LDX # $1000
000710 E4E0 1D2D20 BCLR SCCR2,X,$20
000711 E4E3 1F2E20FC REC1: BRCLR SCSR,X,$20,REC1 ;
GETCODE IS GET CODE FROM PC
000712 E4E7 E62F LDAB SCDR,X
000713 E4E9 CEE000 LDX #T_DATA_TRAN
000714 E4EC 3A ABX
000715 E4ED 1AEE00 LDY 00,X
000716 E4F0 ;*****
000717 E4F0 8CE020 CPX #T_DATA_TRAN+$20 ;
CASE1 IS SEND DATA TO PC
000718 E4F3 2410 BHS SSGN
000719 E4F5 CE1000 LDX # $1000
000720 E4F8 18E600 LDAB 00,Y
000721 E4FB E72F STAB SCDR,X
000722 E4FD 1F2E80FC TR1: BRCLR SCSR,X,$80,TR1
000723 E501 E62F LDAB SCDR,X
000724 E503 2024 BRA OUT_REC
000725 E505
000726 E505 8CE030 SSGN: CPX #T_DATA_TRAN+$30 ;
CASE2 IS GET SIGN FROM PC
000727 E508 2421 BHS GDATA
000728 E50A CE1000 LDX # $1000
000729 E50D E72F STAB SCDR,X
000730 E50F 1F2E80FC TR2: BRCLR SCSR,X,$80,TR2
000731 E513 E62F LDAB SCDR,X
000732 E515 1F2E20FC REC2: BRCLR SCSR,X,$20,REC2
000733 E519 E62F LDAB SCDR,X
000734 E51B C100 CMPB # $00
000735 E51D 2606 BNE SETSGN
000736 E51F 181D0001 CLRSGN: BCLR 00,Y,$01
000737 E523 206D BRA OUTREC
000738 E525 181C0001 SETSGN: BSET 00,Y,$01
000739 E529 2067 OUT_REC: BRA OUTREC

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

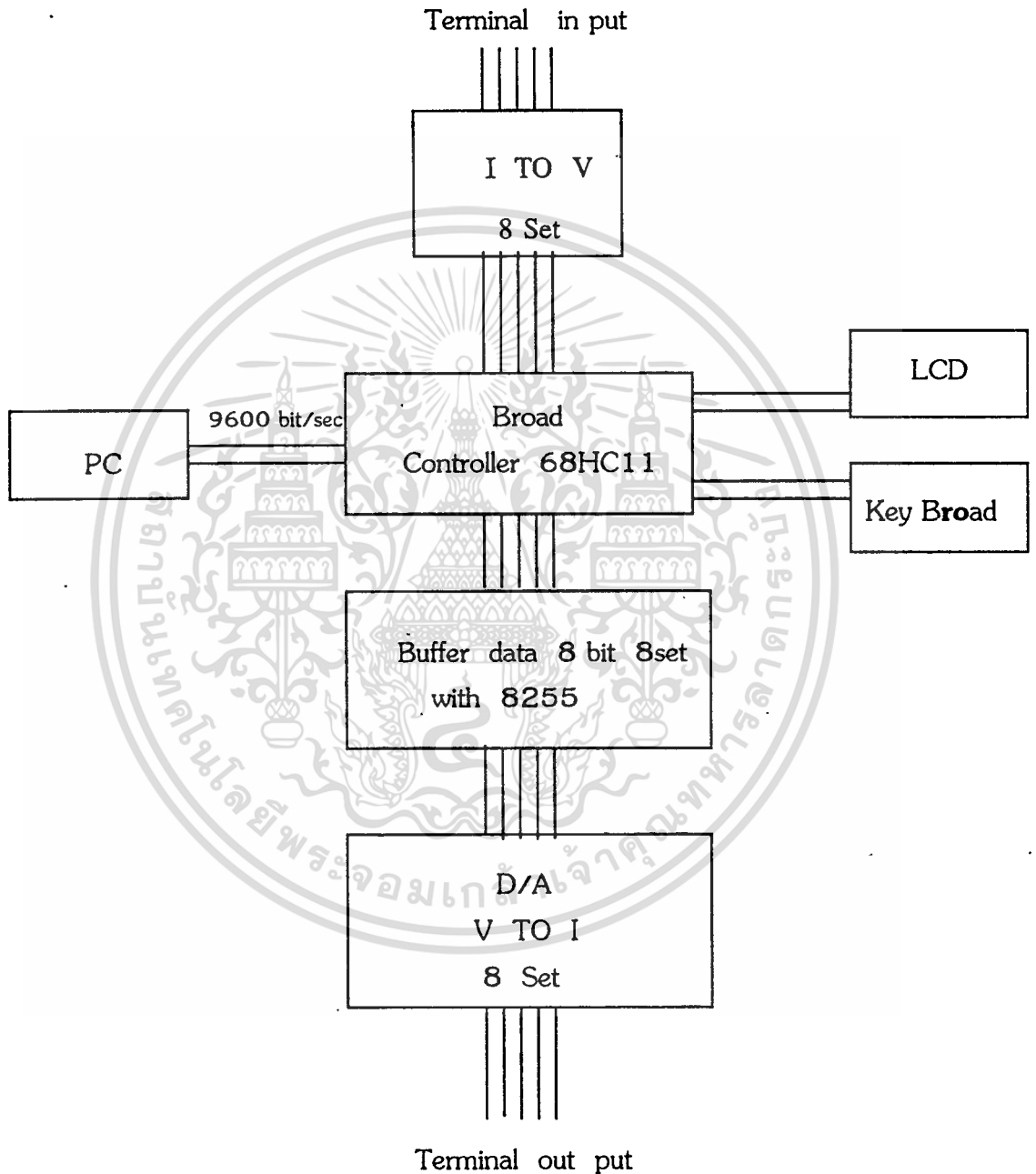
000740 E52B
000741 E52B 8CE0C0          GDATA:  CPX #T_DATA_TRAN+&C0
000742 E52E 2416          BHS SENDTOPC
000743 E530 CE1000        LDX #&1000          ;CASE3 IS
GET PARAMETER FROM PC
000744 E533 E72F          STAB SCDR,X
000745 E535 1F2E80FC      TR3:  BRCLR SCSR,X,&80,TR3
000746 E539 E62F          LDAB SCDR,X
000747 E53B 1F2E20FC      REC3:  BRCLR SCSR,X,&20,REC3
000748 E53F E62F          LDAB SCDR,X
000749 E541 18E700        STAB 00,Y
000750 E544 204C          BRA OUTREC
000751 E546
000752 E546 CE1000        SENDTOPC:  LDX #&1000
000753 E549 E72F          STAB SCDR,X
000754 E54B 1F2E80FC      TR4:  BRCLR SCSR,X,&80,TR4
000755 E54F E62F          LDAB SCDR,X
000756 E551 1F2E20FC      REC4:  BRCLR SCSR,X,&20,REC4
000757 E555 E62F          LDAB SCDR,X
000758 E557 CEE000        LDX #T_DATA_TRAN
000759 E55A 3A            ABX
000760 E55B 1AEE00        LDY 00,X
000761 E55E 8CE030        SENDSGN:  CPX #T_DATA_TRAN+&30
000762 E561 241A          BHS SENDDATA
000763 E563 CE1000        LDX #&1000
000764 E566 18E600        LDAB 00,Y
000765 E569 C501          BITB #&01
000766 E56B 2604          BNE SGNS
000767 E56D C600          LDAB #00
000768 E56F 2002          BRA TSGN
000769 E571 C6FF          SGNS:  LDAB #&FF
000770 E573 E72F          TSGN:  STAB SCDR,X
000771 E575 1F2E80FC      TR5:  BRCLR SCSR,X,&80,TR5
000772 E579 E62F          LDAB SCDR,X
000773 E57B 2015          BRA OUTREC
000774 E57D
000775 E57D 8CE0C0        SENDDATA:  CPX #T_DATA_TRAN+&C0
000776 E580 2410          BHS OUTREC
000777 E582 CE1000        LDX #&1000
000778 E585 18E600        LDAB 00,Y
000779 E588 E72F          STAB SCDR,X

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

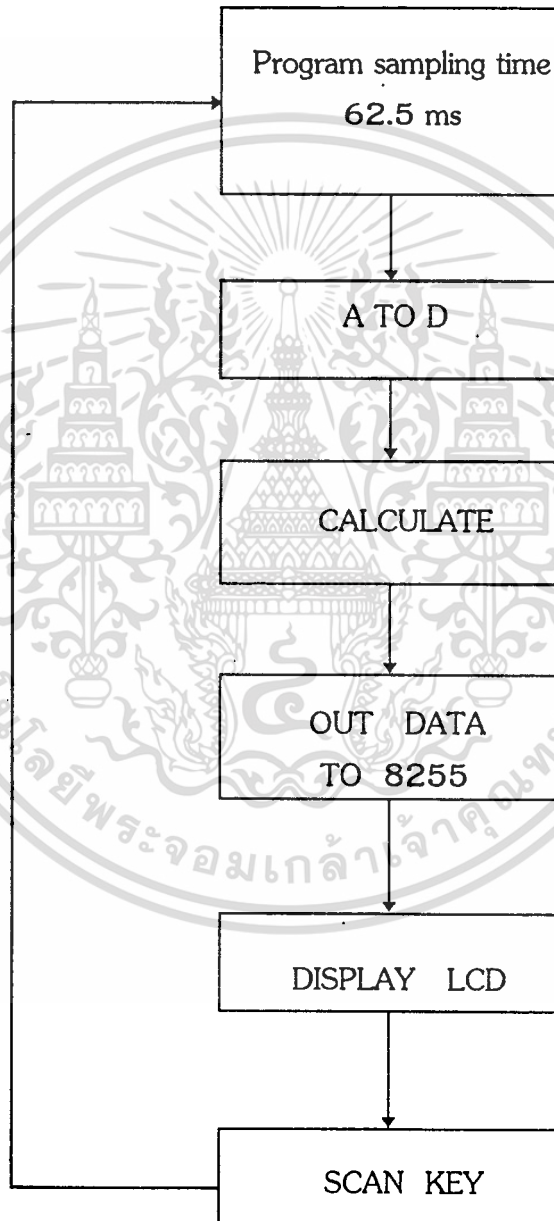


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



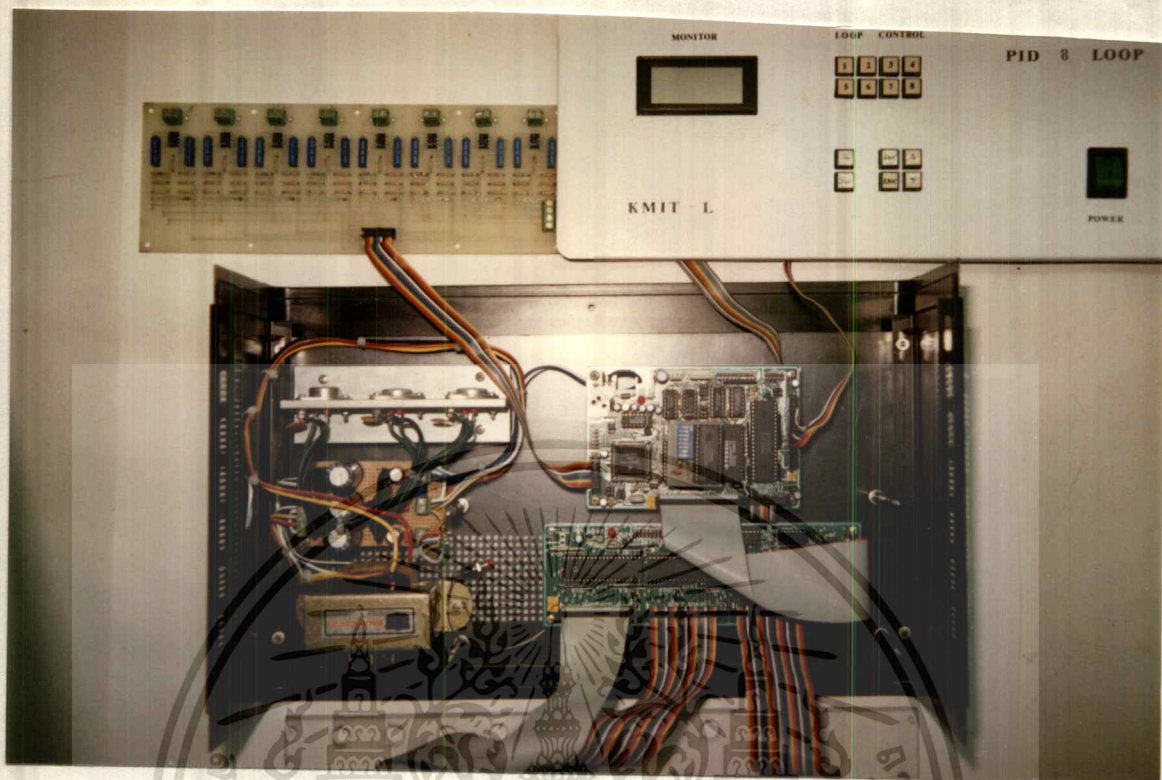
### การเชื่อมต่อส่วนต่างๆของ HARD WARE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

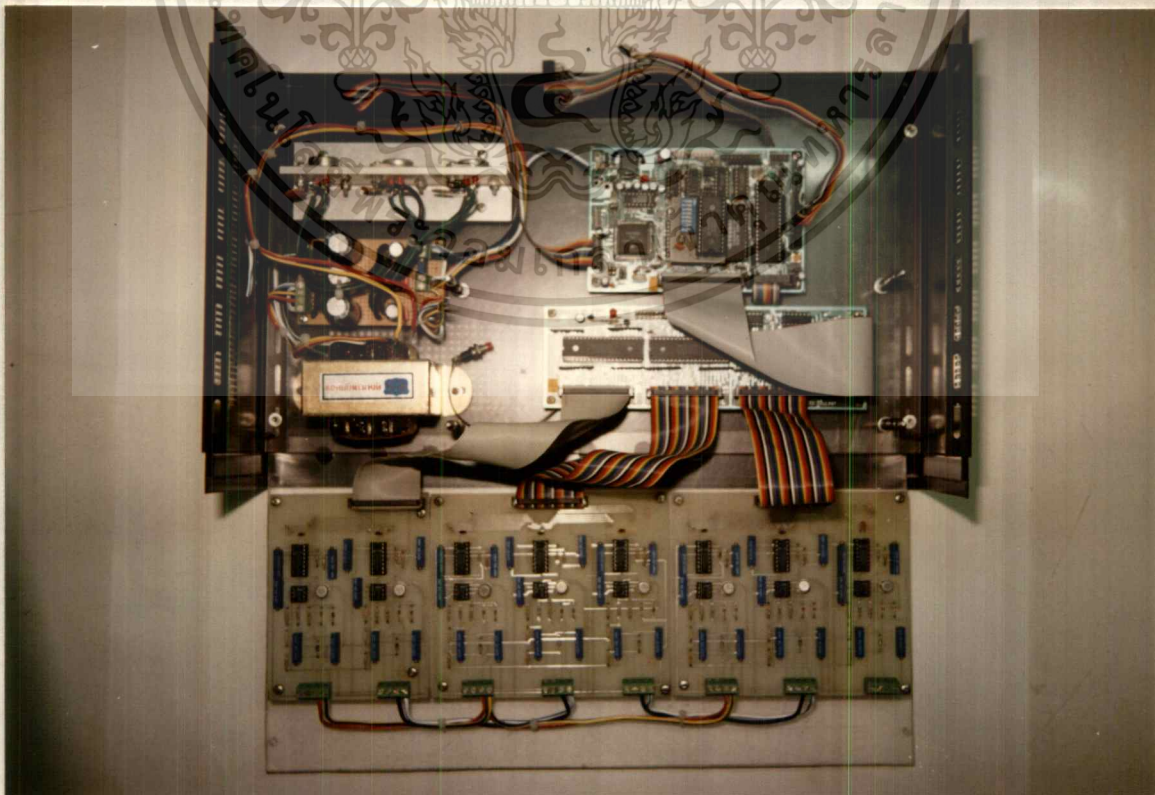


### ขั้นตอนการประมวลผลของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดงการเชื่อมต่อของ CPU กับ I/V , CPU กับ KEY BROAD , CPU กับ DISPLAY



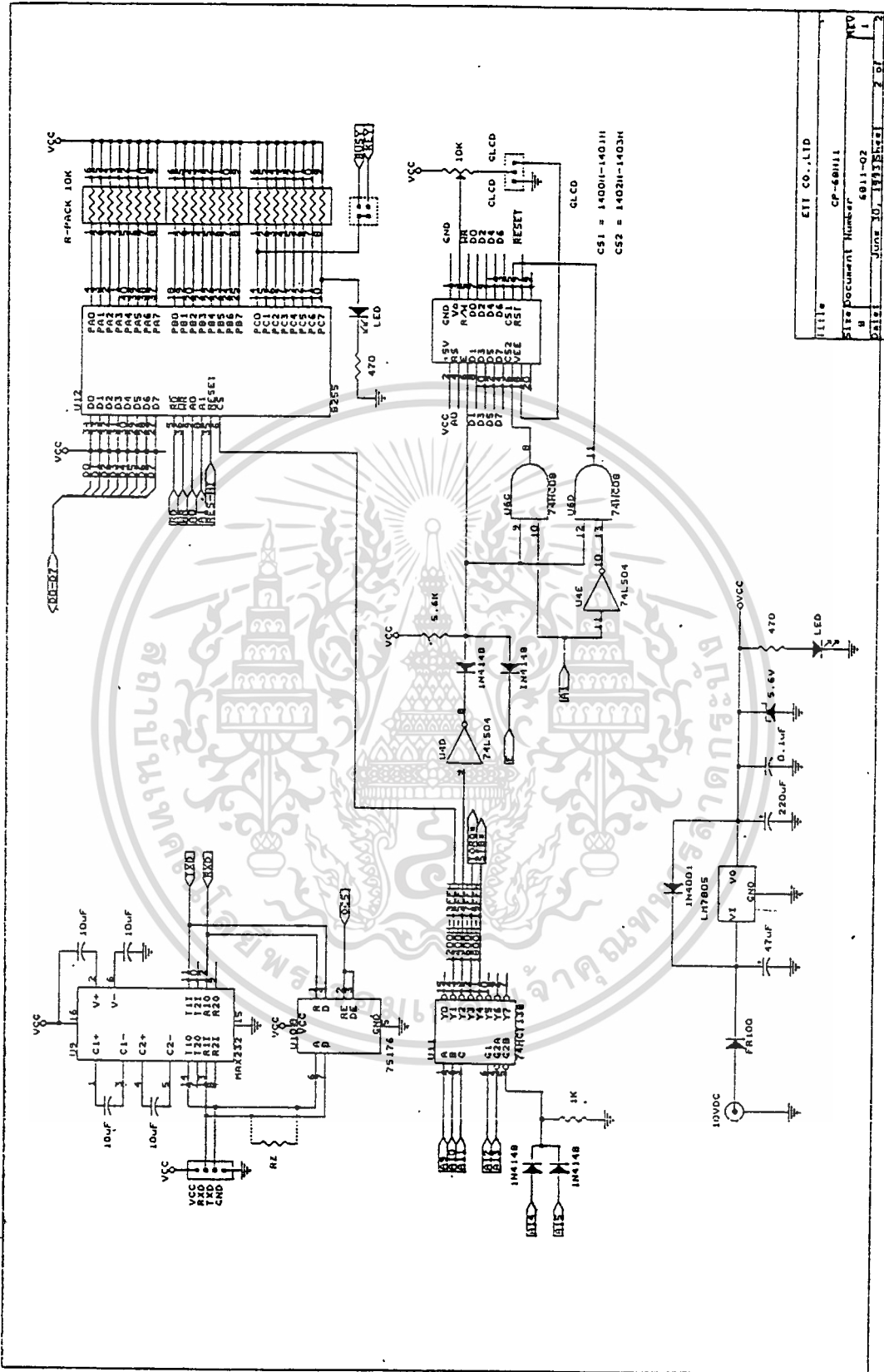
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการ **รูปแสดงการเชื่อมต่อของ CPU กับ** หน้าไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดงของชุด HARD WARE เมื่อประกอบเสร็จ

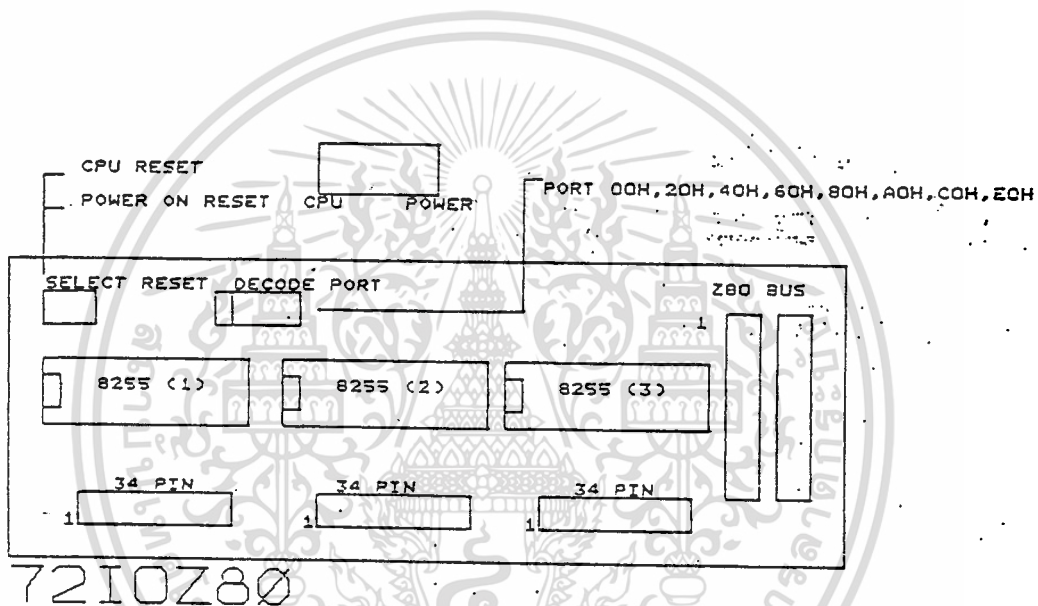
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





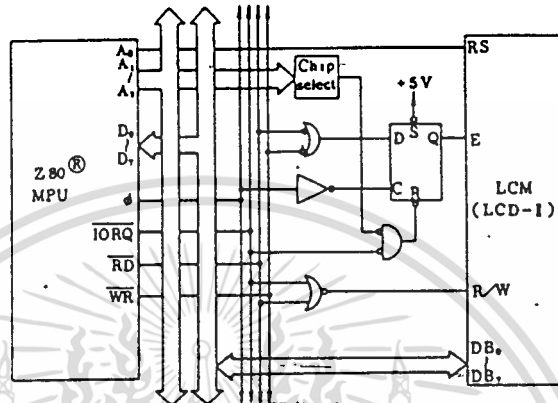
|               |               |
|---------------|---------------|
| ชื่อ          | EIT CO.,LTD   |
| ชื่อเอกสาร    | CP-68111      |
| หมายเลขเอกสาร | 6811-02       |
| วันที่        | JUNE 10, 1993 |
| หน้า          | 1             |
| รวม           | 2 of 2        |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

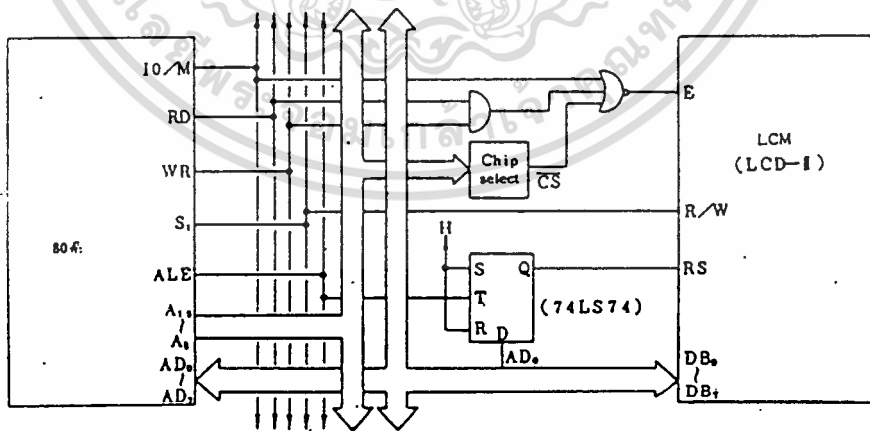


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Example of interfacing to Z80 MPU



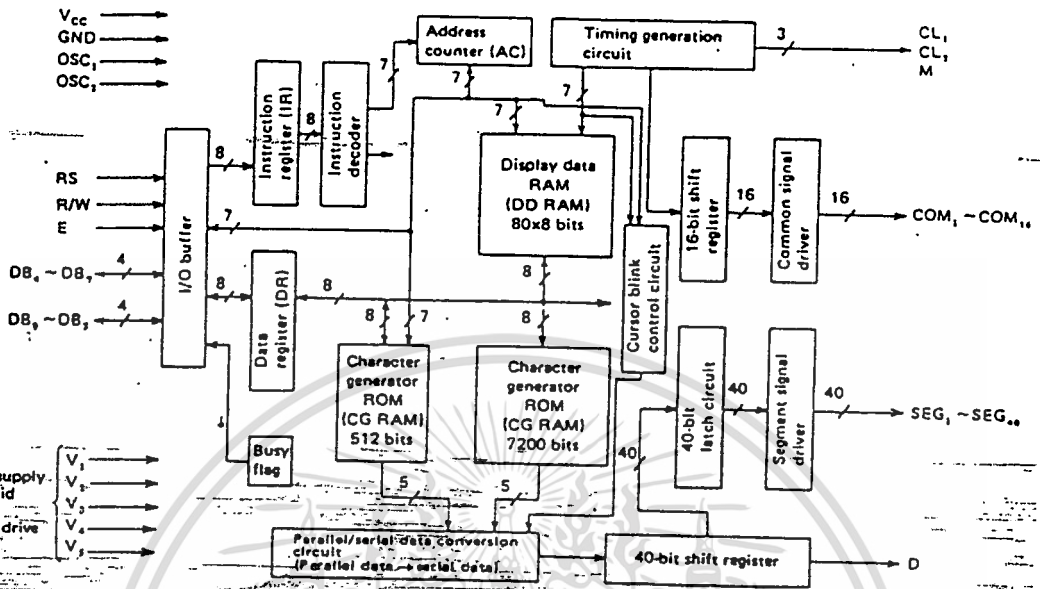
เราสามารถต่อ LCD MODULE (HD44780 เป็น CONTROLER) เข้ากับระบบไมโครได้  
หลายรูปแบบดังรูป



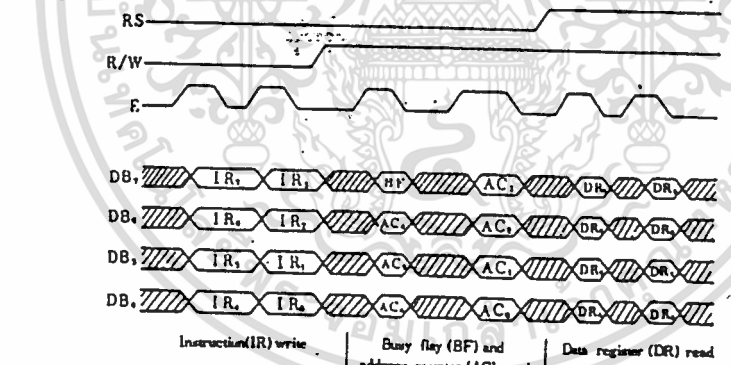
Example of connection with LCM being used as a part of memories on the determined address.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block diagram of HD44780 interior

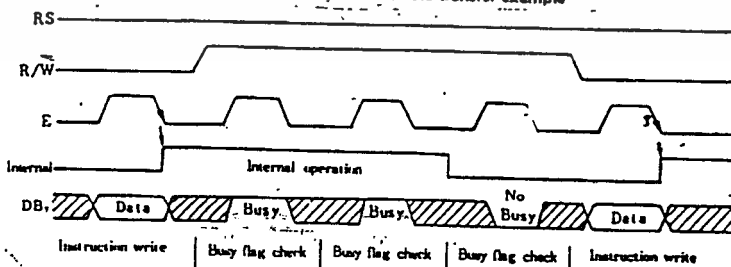


เท่านั้น โดยข้อมูลครึ่งแรกทีส่งนั้น HD44780 จะถือเป็นข้อมูล 4 BIT บน และข้อมูลครึ่งต่อมา  
เป็นข้อมูล 4 BIT ล่าง



Instruction (IR) write | Busy flag (BF) and address counter (AC) read | Data register (DR) read

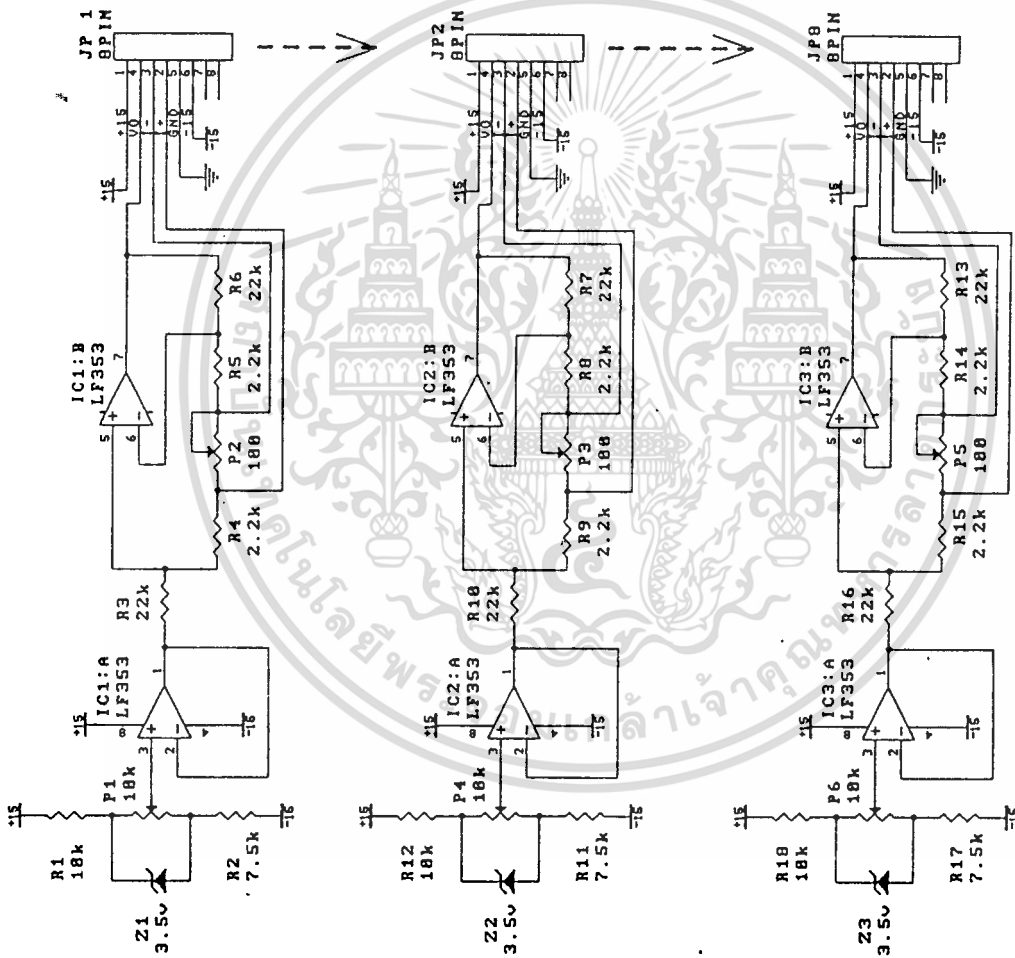
4-bit data transfer example



Instruction write | Busy flag check | Busy flag check | Busy flag check | Instruction write

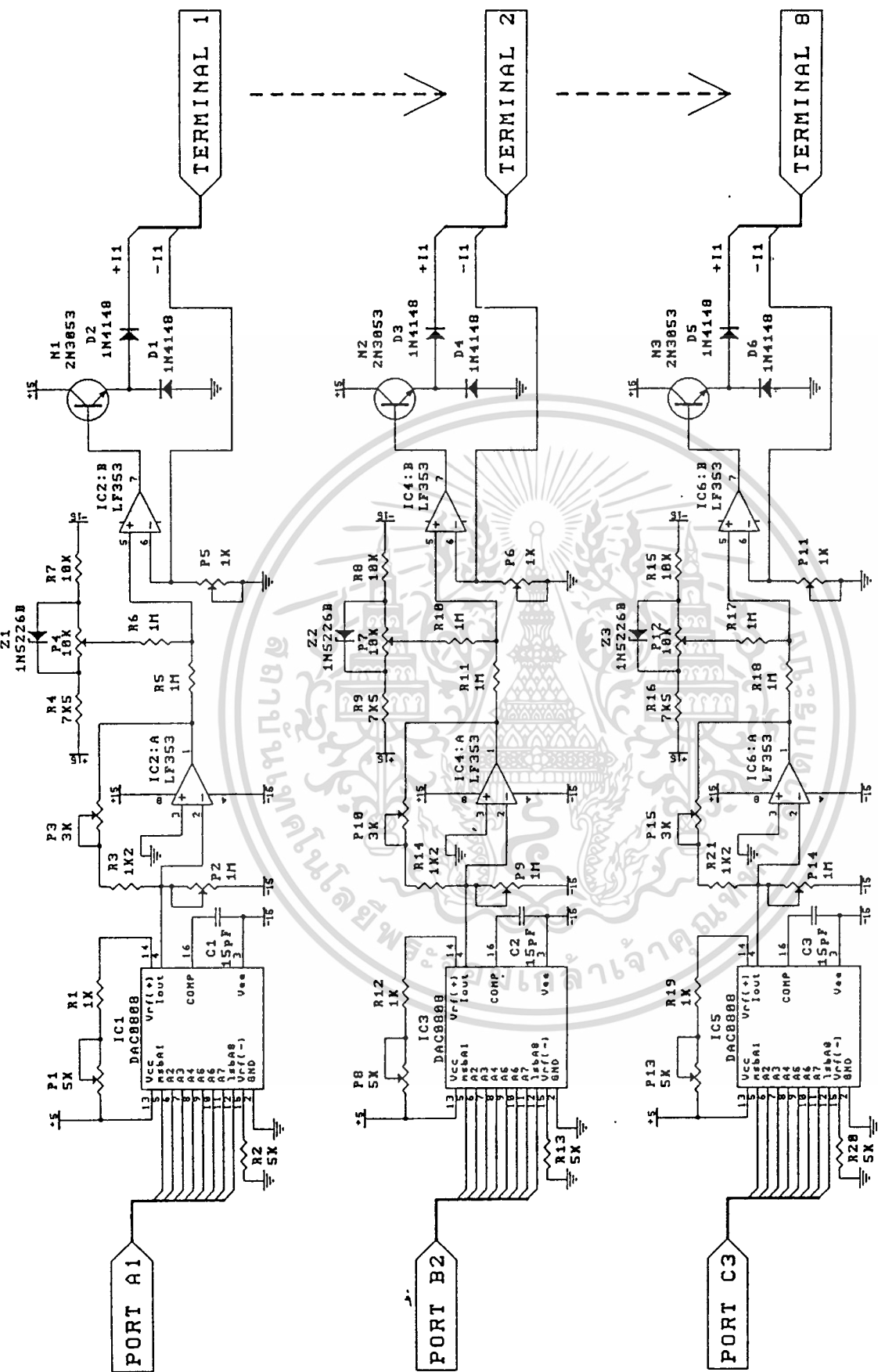
Example of busy flag check timing sequence

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



I TO V CONVERTER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



D TO A CONVERTER WITH U TO I CONVERTER CIRCUIT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**PAGE 1**

|      |         |     |    |   |
|------|---------|-----|----|---|
| Loop | x       | PID | ON | ← |
| SP   | : xxx.x | %   |    | ← |
| PV   | : xxx.x | %   |    |   |
| MV   | : xxx.x | %   |    |   |

**PAGE 2**

|    |            |   |     |   |
|----|------------|---|-----|---|
| x  | PID        | ← | REV | ← |
| KP | : xxx.xxx  |   |     | ← |
| TI | : xxxx.xxx | s |     | ← |
| TD | : xxxx.xxx | s |     | ← |

**Parameter ที่สามารถปรับได้**

|       |                                      |
|-------|--------------------------------------|
| Power | on , off                             |
| SP    | 000.0 To 100.0 เพิ่มทีละ 0.4         |
| MODE  | PI , PD , PID , MNU                  |
| ACT   | REV , DIR                            |
| KP    | 000.000 TO 255.937 เพิ่มทีละ 0.062   |
| TI    | 0000.000 TO 4095.937 เพิ่มทีละ 0.062 |
| TD    | 0000.000 TO 4095.937 เพิ่มทีละ 0.062 |

|     |     |   |   |
|-----|-----|---|---|
| 1   | 2   | 3 | 4 |
| 5   | 6   | 7 | 8 |
| ~   | INC | ↑ |   |
| SAV | DEC | ↓ |   |

## กติกกรรมการ

.. ขอขอบพระคุณ อ. วิทยา กิพย์สุวรรณพร , อ. ธนิตย์ ตริสุวรรณ , อ. สุพรรณกุลพานิชย์ , อ. วิริยะ กองรัตน์ , อ. ประภาส อุดคภิมาพันธ์ ที่ได้ให้การประสิทธิ์ประสาทวิชาให้คำแนะนำปรึกษาในเรื่องราวต่าง แก่ผู้จัดทำ ตลอดจนเพื่อนทุกคนที่ได้ให้ความช่วยเหลือแนะนำต่างๆ และ ให้กำลังใจแก่พวกเราตั้งแต่ต้นตลอดมา ในการทำปฏิญญานิพนธ์จนสำเร็จลุล่วงไปด้วยดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### เอกสารอ้างอิง

1. CURTIS D. JOHNSON *University of Houston*  
“ MICROPROCESSOR - BASED PROCESS CONTROL ”
2. MOTOROLA HANDBOOK, “ MC68HC11 TECHNICAL DATA “  
, MOTOROLA INC....., 1991
3. BEN EZZELL “ GRAPHICS PROGRAMMING IN TURBO C++ “  
ADDISON - WESLEY PUBLISHING COMPANY, INC. , 1990

