เครื่องควบคุม PID โดยใช้ทฤษฎีของ PLL

PID CONTROLLER WITH PLL

โดย

นาย โศธนะ เพิ่มชาติ
นาย อนันต์ สมไร่ขิง
นาย อรรจกร ชำนาญนิตย์

วัน เดือน ปี 1 ส.ค. 2540
เลขทะเบียน 037160
เลขเรียกหนังสือ T 38253 ศ494 ค.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต
สาขาวิศวกรรมการวัดคุมทางอุตสาหกรรม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2538

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

037160

เครื่องควบคุม PID โดยใช้ทฤษฎีของ PLL

PID CONTROLLER WITH PLL

โดย

นาย โศธนะ เพิ่มชาติ

นาย อนันต์ สมไร่ขิง

นาย อรรจกร ชำนาญนิตย์

อาจารย์ที่ปรึกษา

ผศ. วิทยา ทิพสุวรรณพร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องควบคุม PID โดยใช้ทฤษฎีของ PLL

PID CONTROLLER WITH PLL

ผู้จัดทำ

1. นาย โศธนะ เพิ่มชาติ
2. นาย อนันต์ สมไร่ขิง
3. นาย อรรจกร ชำนาญนิตย์

.... อาจารย์ที่ปรึกษา

(ผศ. วิทยา ทิพสุวรรณพร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องควบคุม PID โดยใช้ทฤษฎี PLL

PIDCONTROLLER WITH PLL.

โดย นาย โศธนะ เพิ่มชาติ
นาย อนันต์ สมไร่ขิง
นาย อรรจกร ชำนาญนิตย์
อาจารย์ที่ปรึกษา ผศ. วิทยา ทิพย์สุวรรณพร

บทคัดย่อ

โครงงานนี้เป็นการนำหลักการของเฟสล็อคลูปมาประยุกต์ใช้ในเครื่องควบคุมพีไอดี เพื่อศึกษาการควบคุมแบบพีไอดีทั่วไปและพีไอดีเฟสล็อคลูบ โดยใช้ไมโครคอนโทรลเลอร์เป็นตัวปฏิบัติการ สำหรับพีไอดีเฟสล็อคลูบจะทำการเปลี่ยนสัญญาณดิจิตอลเป็นความถี่และเปรียบเทียบทางความถี่เพื่อหาค่าที่เหมาะสมนำไปควบคุมกระบวนการให้มีการทำงานตามต้องการ โครงงานนี้สามารถนำไปพัฒนาใช้ในงานอุตสาหกรรมทั่วไปได้

ลักษณะการทำงานแบ่งเป็น 6 โหมด ประกอบด้วยโหมดทั่วไป พี,พีไอ,พีไอดี และโหมดเฟสล็อคลูป พี,พีไอ,พีไอดี โดยรับอินพุทและเอาท์พุทเป็นกระแสในระดับมาตรฐาน การแสดงผลจะแสดงโดยแอลอีดี 7 ส่วน(LED 7 segment)และจอมอนิเตอร์

Abstract:

This project was used the principle of phase lock loop apply to the PID controller for study the general PID controlling and phase locked loop PID with microcontroller performance. The phase lock loop PID performance conversion digital

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

signalling to frequency and frequency compare for the suitable value and process control as desire. In addition, still can modify and develop for the general industrial.

The performance divied to 6 mode, general mode P, PI, PID and phase lock loop mode P, PI, PID. Input and output are standardize level. It display by 7-Segment LED and monitor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

สารบัญ
สารบัญตาราง
สารบัญภาพ
คำย่อและสัญลักษณ์

บทที่

1. บทนำ 1-1
1.1 บทนำ 1-1
1.2 วัตถุประสงค์ของงานวิจัย 1-1
1.3 ขอบเขตของงานวิจัย 1-2
1.4 ประโยชน์ที่ได้รับจากงานวิจัย 1-2

2. ทฤษฎีดิจิทัลเฟสล็อกลูป 2-1
2.1 นิยามของหลักการทำงานเฟสล็อกลูปตามแบบฉบับ 2-1
2.2 นิยามดิจิทัลเฟสล็อกลูป 2-4
2.3 ดิจิทัลเฟสล็อกลูปแบบต่าง ๆ กับสัญญาณแอนะล็อกตัวกลาง 2-5

3. โหมดการควบคุมของคอนโทรลเลอร์ 3-1
3.1 นิยามระบบควบคุม 3-1
3.2 การควบคุมแบบเปิด-ปิด 3-2
3.3 การควบคุมแบบต่อเนื่อง 3-2
3.3.1 โหมดการควบคุมแบบพรอปพรอชันแนล 3-2
3.3.2 โหมดการควบคุมแบบอินทีกรัล 3-4
3.3.3 โหมดการควบคุมแบบดีริเวทีฟ 3-4
3.3.4 โหมดการควบคุมแบบคอมโพสิต 3-5
3.4 บทนิยามอัลกอลิธึมของการควบคุม 3-6
3.5 อัลกอลิธึมของโหมดควบคุม 3-9
3.5.1 โหมดพรอปพรอชันแนล 3-9
3.5.2 โหมดอินทีกรับหรือโหมดรีเซต 3-13
3.5.3 โหมดดีริเวทีฟ 3-19
3.5.4 โหมดคอมโพสิต 3-22

4. การออกแบบวงจรตัวดิจิทัลคอนโทรลเลอร์ 4-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 โครงสร้างของตัวดิจิทัลคอนโทรลเลอร์ 4-1
4.1.1 ลักษณะทางสถาปัตยกรรมของ 8031 โดยย่อ 4-2
4.1.2 การออกแบบ 8031 ไมโครคอนโทรลเลอร์บอร์ด 4-4
4.2 วงจรต่าง ๆ ในตัวดิจิทัลคอนโทรลเลอร์ 4-6
4.2.1 วงจรสแกนคีย์บอร์ด 4-6
4.2.2 วงจรแสดงผล 4-9
4.2.3 วงจร ADC และ DAC 4-11

5. หลักการออกแบบซอฟต์แวร์และการใช้งาน 5-1
5.1 แนวทางการออกแบบซอฟต์แวร์ของตัวดิจิทัลคอนโทรลเลอร์ 5-1
5.1.1 ไมโครคอนโทรลเลอร์และซอฟต์แวร์ 5-1
5.1.2 โครงสร้างของโปรแกรมมอนิเตอร์ของตัวดิจิทัลคอนโทรลเลอร์ 5-4
5.1.3 การจัดการทำงานของโปรแกรม 5-5
5.1.4 หน้าที่ของรูทีนคีย์บอร์ด 5-5
5.1.5 หน้าที่ของรูทีนแสดงผลบนตัวเลข 7 ส่วน 5-7
5.1.6 หน้าที่ของรูทีนเซตพารามิเตอร์ของตัวดิจิทัลคอนโทรลเลอร์ 5-9
5.2 วิธีการใช้งานตัวดิจิทัลคอนโทรลเลอร์ 5-10
5.2.1 การเซตโหมดการควบคุม 5-11
5.2.2 การเซตพารามิเตอร์การควบคุม KP, KI, KD 5-12
5.2.3 การเซตค่าเซตพอยน์ 5-14
5.2.4 แสดงค่า process variable(PV) 5-15
5.2.5 การเซตอัตราบอด 5-15

6 การทดสอบตัวดิจิทัลคอนโทรลเลอร์และโปรแกรมพล้านต์มอนิเตอริง 6-1

บรรณานุกรม
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่ หน้า

2-1 (a) บล็อกไดอะแกรมของ ระบบเฟสล็อกลูป (b) ฟังก์ชันถ่ายโอนของ VCO (c) ฟังก์ชันถ่ายโอนของ PD 2-1

2-2 โครงสร้างทางเสถียรภาพทางไดนามิคและสแตติกของ PLL แบบเชิงเส้น 2-3

3-1 ส่วนประกอบของระบบควบคุมอย่างง่าย 3-1

3-2 กราฟแสดงการควบคุมแบบเปิด-ปิดและเดดแบนด์ 3-2

3-3 คอนโทรลเอาท์พุทในโหมด P จะราบเรียบภายในข้อกำหนดของเกน และเออเรอร์ 3-3

3-4 ในกรณีที่เออเรอร์เป็นศูนย์ แต่เอาต์พุตดีริเวทีฟมีการตอบสนอง 3-5

3-5 เป็นโฟล์วชาร์ตแสดงค่าเออเรอร์ DE ทั้งขนาดและเครื่องหมาย 3-9

3-6 โฟล์ชาร์ตอัลกอลิธึมของโหมดพรอปพรอชันแนล 3-10

3-7 รายละเอียดโฟล์วชาร์ตของโหมด P แสดงแฟลกเครื่องหมาย และการป้องกัน โอเวอร์โฟล์วและอันเดอร์โฟล์ว 3-12

3-8 กราฟแสดงตัวอย่างพื้นที่เออเรอร์ต่อเวลาโดยประมาณพื้นที่เป็นสี่เหลี่ยม 3-14

3-9 โฟล์วชาร์ตโดยทั่วไปของโหมดอินทีกรัล 3-16

3-10 รายละเอียดโฟล์วชาร์ตของโหมดอินทีกรัล แสดงถึงแอคคิวมูเลเตอร์ SUM และ การป้องกันโอเวอร์โฟล์ว อันเดอร์โฟล์ว 3-18

3-11 โฟล์วชาร์ตสำหรับโหมดดีริเวทีฟแสดงถึงการกำหนดขั้วเครื่องหมาย 3-21

3-12 (a),(b) โฟล์วชาร์ตของโหมด PI แสดงถึงการป้องกันโอเวอร์โฟล์วและอันเดอร์โฟล์ว 3-26

4-1 แสดงบล็อกไดอะแกรมของตัวดิจิทัลคอนโทรลเลอร์ 4-1

4-2 วงจร 8031 บอร์ด 4-3

4-3 วงจรถอดรหัสแอดเดรสอุปกรณ์ภายนอก 4-4

4-4 บล็อกไดอะแกรมสแกนคีย์บอร์ด 4-7

4-5 วงจรสแกนคีย์บอร์ด 4-8

4-6 แสดงเป็นวงจรขับตัวเลข 7 ส่วนอย่างง่ายแบบทั่วไป 4-10

4-7 การขับตัวเลข 7 ส่วนโดยวิธีตรง 4-11

4-8 แสดงวงจร ADC และ DAC 4-12

4-9 วงจรดิจิทัลเฟสล็อกลูป 4-14

5-1 บล็อกไดอะแกรมของซอฟต์แวร์โปรแกรมมอนิเตอร์ของคอนโทรลเลอร์ 5-4

5-2 บล็อกไดอะแกรมของซอฟต์แวร์รูทีนคีย์บอร์ด 5-6

5-3 แสดงโฟล์วชาร์ตของการแสดงผลเป็นค่าเปอร์เซ็นต์ 5-8

5-4 แสดงโฟล์วชาร์ตของการเปลี่ยนค่าเกน 5-10

5-5 แสดงหน้าปัดด้านหลังของตัวคอนโทรลเลอร์ 5-11

6-1 แสดงการติดตั้งตัวคอนโทรลเลอร์กับกระบวนการควบคุมระดับ 6-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่		หน้า
6-2	แสดงด้านหน้าของตัวดิจิทัลคอนโทรลเลอร์	6-2
6-3	แสดงอุปกรณ์ภายในของดิจิทัลคอนโทรลเลอร์	6-2
6-4	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5	6-3
6-5	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5 Ki = 0.05	6-3
6-6	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5 Ki = 0.05 Kd = 0.02	6-4
6-7	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5	6-4
6-8	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5 Ki = 0.05	6-5
6-9	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5 Ki = 0.05 Kd = 0.02	6-5
6-10	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5	6-6
6-11	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5 Ki = 0.05	6-6
6-12	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5 Ki = 0.05 Kd = 0.02	6-7
6-13	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5	6-7
6-14	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5 Ki = 0.05	6-8
6-15	การตอบสนองสเต็ปขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5 Ki = 0.05 Kd = 0.02	6-8
6-16	การตอบสนองสเต็ปลงจากค่าเซตพอยน์ 60% เป็น 40% ค่า Kp = 6.5 Ki = 0.07	6-9
6-17	การตอบสนองสเต็ปลงจากค่าเซตพอยน์ 60% เป็น 50% ค่า Kp = 6.5 Ki = 0.05	6-9
6-18	การตอบสนองสเต็ปขึ้นจากค่าเซตพอยน์ 40% เป็น 60% ค่า Kp = 6.5 Ki = 0.04	6-10
A	ภาพแสดงโปรแกรมมอนิเตอร์	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำย่อและสัญลักษณ์

8031 -ไมโครโปรเซสเซอร์แบบ 8031 ของบริษัท อินเทล(Intel)

ADC,A/D - ตัวแปลงสัญญาณแอนะล็อกเป็นสัญญาณดิจิทัล(Analog to Digital Converter)

ADLL - ระบบดิจิทัลเฟสล็อกลูปล้วน(All Digital Phase Locked Loop)

CV - เอาต์พุตคอนโทรล(Control Variable)

DAC,D/A - ตัวแปลงสัญญาณดิจิทัลเป็นสัญญาณแอนะล็อก(Digital to Analog Converter)

DCO - ดิจิทัลคอนโทรลออสซิลเลเตอร์

DPLL - ดิจิทัลเฟสล็อกลูป(Digital Phase Locked Loop)

INP - อินพุต(Input)

PD - เฟสดีเทกเตอร์(Phase Detecter) แบ่งเป็น 4 แบบด้วยกัน แบบที่ 1 คือ เป็นตัวคูณสี่ควอดแครนต์ แบบที่ 2 คือเกทแบบเอกซ์คลูสีฟออร์ แบบที่ 3 คือฟลิบฟลอบแบบ J-K และ แบบที่ 4 คือ PFD

PFD - เฟส/ความถี่ดีเทกเตอร์ (Phase Frequency Detecter)

PLL - เฟสล็อกลูป(Phase Locked Loop)

PV - Process Variable

PWM - พัลส์วิทด์มอดูเลชัน(Pulse Width Modulation)

RC - หมายถึงวงจรซึ่งประกอบด้วยตัวต้านทานและตัวเก็บประจุ

SP - เซตพอยน์

TLX,TRX - เรจิสเตอร์พิเศษของไมโครโปรเซสเซอร์เบอร์ 8031 สำหรับการสื่อสารอนุกรม

VCO - โวลต์เตจคอนโทรลออสซิลเลเตอร์ (Voltage Control Oscilator)

XOR - เกทแบบเอกซ์คลูสีฟออร์

Z^{-1} - ดีเลย์โอเปอร์เรเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 บทนำ

ในปัจจุบันนี้ระบบต่าง ๆ ซับซ้อนมากขึ้น ซึ่งย่อมเกี่ยวพันกับการควบคุม ซึ่งในปัจจุบันมีการควบคุมหลายวิธี เช่นการควบคุมแบบต่อเนื่อง เช่น PID เป็นต้น ในบางงานระบบควบคุมสมัยใหม่จะใช้วิธีเฟสล็อคลูปกันมากขึ้น ซึ่งเป็นแบบอย่างมาจากวงการสื่อสาร สิ่งหนึ่งที่ขาดไม่ได้ก็คือการนำไมโครโปรเซสเซอร์มาประยุกต์ใช้กับงานต่างๆ ที่เกี่ยวกับในระบบควบคุม ดังนั้นในผลงานวิจัยนี้จึงได้นำเอา 2 สิ่งดังกล่าวมาประยุกต์ใช้งาน โดยการนำเอาไมโครคอนโทรลเลอร์ หรือไมโครโปรเซสเซอร์มาพัฒนาเป็นตัวคอนโทรลเลอร์แบบเฟสล็อคลูป โดยทำงานตามลักษณะของไอซีเฟสล็อคลูปแบบสำเร็จรูป มาใช้ประโยชน์ที่ได้ในการซิงโครไนซ์สัญญาณอ้างอิงกับสัญญาณป้อนกลับจากกระบวนการ ซึ่งเป็นลักษณะการเปรียบเทียบรูปคลื่นทางเฟส และความถี่อินพุตความถี่เปรียบเทียบและสัญญาณความถี่อ้างอิง เพื่อให้ได้เอาต์พุตออกมาใช้งานควบคุมตามความต้องการ ซึ่งในส่วนซอฟต์แวร์โปรแกรมแทนเฟสดีเทกเตอร์ จะมีความยืดหยุ่นในการใช้งานดีกว่าการใช้ ไอซีสำเร็จรูป โดยพยายามรักษาคุณสมบัติความเที่ยงตรง และความเร็วในการแทรคกิ้งของระบบเฟสล็อคไว้

ส่วนของไมโครคอนโทรลเลอร์จะทำงานเป็นตัวคอนโทรเลอร์แบบเฟสล็อคแล้วยังทำการมีโหมดทั่วไปไว้ให้เลือกใช้งาน คือ โหมดพรอปพรอชันแนล หรือ โหมด P โหมดพรอปพรอชันแนล-อินทีกรัล หรือ PI และ โหมดพรอปพรอชันแนล-อินทีกรัล/ดีริเวทีฟ หรือ PID ซึ่งเป็นโหมดใช้ในระบบควบคุมทั่ว ๆ ไป

อีกเพิ่มไมโครคอมพิวเตอร์นำมาใช้ช่วยงาน ควบคุมดูแลตัวคอนโทรลเลอร์ โดยให้ไมโครคอมพิวเตอร์สามารถทำหน้าที่เป็น พล้านต์มอนิเตอริง

1.2 วัตถุประสงค์ของงานวิจัย

เพื่อหาแนวทางพัฒนาตัวควบคุมแบบเฟสล็อก เพื่อให้ใช้เป็นตัวควบคุมแบบมาตรฐานซึ่งใช้ในอุตสาหกรรมทั่วไปเหมือนกับตัวควบคุมแบบปกติ ซึ่งนิยมใช้กันอยู่ทั่วไป โดยพยายามใช้ข้อดีต่าง ๆ ของระบบเฟสล็อกลูปมาใช้ปฏิบัติและพัฒนา ที่สำคัญได้ทำเป็นต้นแบบเพื่อไปทดลองใช้งานกับกระบวนการได้

เอกสารนี้ ... ที่ ... ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของงานวิจัย

ขอบเขตในงานวิจัย ในจุดเริ่มแรก ได้มาถึงระบบเฟสล็อกลูปกำลังเป็นดาวรุ่ง ซึ่งเดิมทีถูกจำกัดอยู่ในแวดวงการสื่อสาร แล้วมีการพัฒนามาใช้ระบบควบคุมความเร็วของมอเตอร์ ซึ่งประสบความสำเร็จอย่างท่วมท้น อย่างไรก็ดีการนำเอาระบบเฟสล็อกลูปมาใช้กับระบบควบคุมนั้น ยังต้องคำนึงถึงความสามารถของเฟสดีเทกเตอร์บางแบบในวงจรสำเร็จรูปอยู่ ซึ่งผลลัพธ์ที่ได้เป็นที่ยอมรับว่า ตัวดีเทกเตอร์เฟส/ความถี่ จะมีคุณสมบัติดีที่สุดในบรรดาตัวดีเทกเตอร์ต่าง ๆ ในระบบเฟสล็อกลูป

ในปัจจุบันตัวคอนโทรลเลอร์ในแบบการควบคุมทั่วไป ซึ่งถ้าหากใช้ไมโครโปรเซสเซอร์เลียนแบบอัลกอลิธึมในการทำงานของมันแล้ว จะเรียกอีกอย่างหนึ่งว่า “ดิจิทัลคอนโทรลเลอร์” ซึ่งในทางอุตสาหกรรมเราหันมาใช้ตัวคอนโทรลเลอร์แบบดิจิทัลมากขึ้น เพราะความยืดหยุ่นในการใช้งานการเปลี่ยนแปลงพารามิเตอร์ และยังมีความสามารถทัดเทียมระบบแอนะล็อกมากขึ้น ซึ่งส่วนหนึ่งเป็นเพราะความสามารถของซีพียู ซึ่งปัจจุบันมีความเร็วสูง

ดังนั้นจึงไม่อาจมองข้ามในการนำไมโครโปรเซสเซอร์ ในการนำไปประยุกต์เกี่ยวกับระบบเฟสล็อกลูปได้ ถึงแม้ว่าระบบเฟสล็อกลูปเดิมซึ่งเป็นวงจรอิเล็กทรอนิกส์นั้นก็มีความสามารถดีเลิศแล้ว การนำไมโครโปรเซสเซอร์มาเลียนแบบอัลกอลิธึมของตัวเฟสดีเทกเตอร์ปัจจุบันยังต้องมีการพัฒนาเพื่อให้มีความสามารถดีเทียบเท่ากับวงจรอิเล็กทรอนิกส์

ผู้อ่านบางท่านสงสัยว่าในเมื่อระบบเฟสล็อกลูปในปัจจุบันเป็นขบวนการดิจิทัล ซึ่งใช้เป็นวงจรอิเล็กทรอนิกส์สำเร็จรูปนั้นดีอยู่แล้ว ทำไมจึงต้องไปใช้ไมโครโปรเซสเซอร์เลียนแบบด้วย ทั้งที่จะได้ผลลัพธ์ที่ดีที่สุด โดยไม่ต้องคำนึงถึงวิธีการปฏิบัติ หรือกล่าวอีกนัยหนึ่งว่า สิ่งที่ได้ทำนั้นดีกว่าเก่าอย่างไร ซึ่งคำตอบที่ได้นั้นคือ ผลลัพธ์ในการเลียนแบบขบวนการของเฟสดีเทกเตอร์โดยซอฟต์แวร์นั้น ไม่อาจดีกว่าวงจรอีเล็กทรอนิกส์ได้ อย่างดีที่สุดก็เทียบเท่า

เปรียบเทียบเหมือนกับ การใช้ซอฟท์แวร์แทนการควบคุมในโหมดทั่วไป เช่น โหมด PID โดยการใช้ไมโครโปรเซสเซอร์ ไปควบคุมลูปของกระบวนการ โดยสมมุติให้ขบวนการมีค่าคงที่เวลาน้อยกว่าอัตราแซมปลิง ผลลัพธ์ในการควบคุมก็ยังเกิดปัญหาบางอย่างขึ้นได้ ดังนั้นการใช้ซอฟต์แวร์แทนโหมดควบคุมปกติไม่อาจเทียบกับการควบคุมโดยใช้วงจรแอนะล็อกได้ อย่างไรก็ดีประโยชน์ที่ได้รับในการนำไมโครโปรเซสเซอร์มาใช้งานคือ ความก้าวหน้าอีกขั้นหนึ่ง

1.4 ประโยชน์ที่ได้รับจากงานวิจัย

ในโหมดควบคุมแบบเฟสล็อก เป็นตัวเลือกใหม่ในการควบคุม ซึ่งในงานบางอย่างเราสามารถใช้แทนโหมดควบคุมแบบปกติได้ดี เช่น สมมุติถ้าขบวนการมีการเปลี่ยนแปลงฟังก์ชันถ่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้เพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โอน เราจะต้องมีปรับพารามิเตอร์ของโหมดควบคุมปกติอยู่เสมอ แต่เหล่านี้จะไม่เกิดขึ้นโหมดการควบคุมแบบเฟสล็อกในช่วงแปรผันค่าหนึ่ง

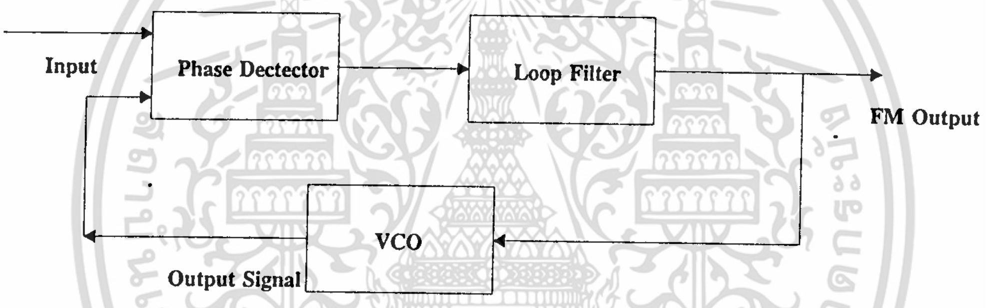
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีดิจิทัลเฟสล็อกลูป

2.1 นิยามของหลักการทำงานเฟสล็อกลูปตามแบบฉบับ

เฟสล็อกลูปคือ วงจรพิเศษที่ซึ่งทำให้ระบบหนึ่งปฏิบัติตามรอยอีกระบบหนึ่งได้ หรืออธิบายให้ชัดเจนคือ เฟสล็อกลูป (PLL) เป็นวงจรที่ทำซิงโครไนซ์สัญญาณเอาต์พุตกับสัญญาณอ้างอิงหรือสัญญาณอินพุตในความถี่และเฟสเดียวกัน ซึ่งการซิงโครไนซ์นี้ถูกเรียกบ่อย ๆ ว่า "สถาวะล็อก (locked state)" โดยที่เฟสเออเรอร์ระหว่างออสซิลเลเตอร์เอาต์พุตกับสัญญาณอ้างอิงนั้นเป็นศูนย์หรือน้อยมาก



(a)

(b) (c)

รูปที่1 (a) บล็อกไดอะแกรมของระบบเฟสล็อกลูป (b) ฟังก์ชั่นถ่ายโอนของ VCO

(c) ฟังก์ชั่นถ่ายโอนของ PD

นั่นเป็นหลักการโดยย่อของเฟสล็อกลูปบล็อกไดอะแกรมแสดงตามรูปที่1 โดยวงจรเฟสล็อกลูปจะประกอบด้วยบล็อกฟังก์ชัน 3 ส่วนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ออสซิลเลเตอร์ควบคุมด้วยแรงดันไฟฟ้า (VCO)
2. เฟสดีเทกเตอร์ (PD)
3. ลูปฟิลเตอร์ (LF)

สัญญาณภายในวงจรเฟสล็อกลูปที่น่าสนใจกำหนดดังนี้
- สัญญาณอ้างอิง หรืออินพุต u_1 (t)
- ความถี่เชิงมุม (angular frequency) ω_1 ของสัญญาณเอาต์พุต
- สัญญาณเอาต์พุต u_2 (t) ของ VCO
- ความถี่เชิงมุม ω_2 ของสัญญาณเอาต์พุต
- สัญญาณเอาต์พุต u_d (t) ของเฟสดีเทกเตอรื
- สัญญาณเอาต์พุต u_f (t) ของลูปฟิลเตอร์
- เฟสเออเรอร์ θ_e กำหนดเป็นความต่างเฟสของสัญญาณ u_1 (t) และ u_2 (t)

เรามาสนใจกับการทำงานทั้งสามบล็อกตามรูป 2-1 ตัว VCO ออสซิลเลตที่ความถี่เชิงมุม ω_2 ซึ่งถูกกำหนดสัญญาเอาต์พุต u_f ของลูปฟิลเตอร์จะออสซิลเลตที่ความถี่เชิงมุม ω_2 กำหนดเป็นสมการดังนี้

$$\omega_2 (t) = \omega_0 + K_o u_f (t) \qquad (2\text{-}1)$$

ที่ซึ่ง ω_0 คือความถี่กลางของ VCO และ K_0 คือแกนของ VCO ในหน่วย S^{-1} V^{-1}

ตัว PD หรืออ้างเป็น เฟสคอมพาราเตอร์ ได้ด้วย จะทำหน้าที่เป็นตัวเปรียบเทียบเ ฟส ระหว่างสัญญาณเอาต์พุตกับเฟสกับสัญญาณอ้างอิง ซึ่งจะจ่ายสัญญาเอาต์พุต u_d (t) โดยประมาณเป็นสัดส่วนโดยตรงของเฟสเออเรอร์ e ภายในช่วงจำกัดช่วงหนึ่ง

$$u_d (t) = K_d \theta e \qquad (2\text{-}2)$$

ที่ซึ่ง K_d แทนด้วยแกนของ PD มีหน่วยเป็นโวลต์ หรือ V/ rad สำหรับเหตุผลในการอธิบาย แสดงตามรูปที่ 2-1c เป็นกราฟซึ่งแทนจากสมการที่ 2-2

ต่อไป มาดูการทำงานทั้งสามฟังก์ชันบล็อก โดยเริ่มแรกสมมุติให้ความถี่เชิงมุมของสัญญาณอินพุต u_1 (t) มีค่าเท่ากับ ความถี่กลาง (centor frequency) ซึ่งแทนด้วย ω_0 แล้วให้ VCO ทำงานที่ความถี่กลาง ดังนั้นจะเห็นว่าเฟสเออเรอร์ θ_e เป็นศูนย์ ซึ่งถ้า θ_e เป็นศูนย์นั้น u_d ของ PD ต้องเป็นศูนย์ด้วย เพราะฉะนั้นสัญญาณเอาต์พุตของลูปฟิลเตอร์ u_f จะเป็นศูนย์ด้วย ซึ่งเงื่อนไขนี้เป็นการให้ VCO ทำงานที่ความถี่กลาง

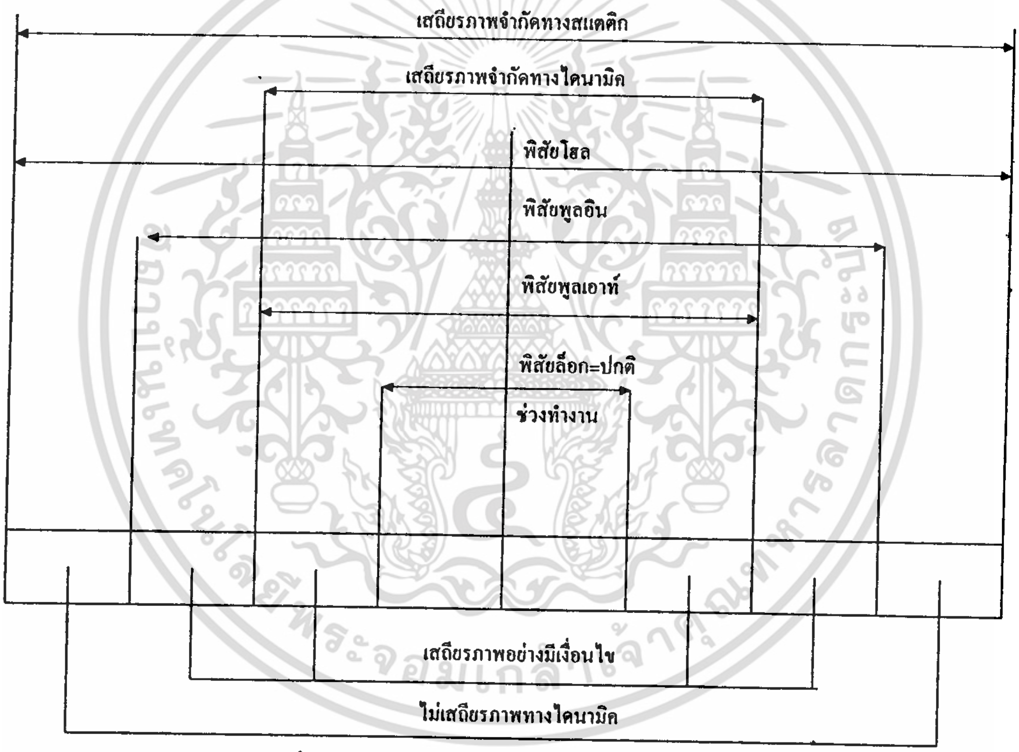
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเฟสเออเรอร์ θ_e เริ่มต้นไม่เป็นศูนย์แล้ว PD จะสร้างสัญญาณเอาต์พุต u_d ไม่เป็นศูนย์ หลังจากผ่านลูปฟิลเตอร์เป็นสัญญาณ u_f ด้วยเหตุนี้จะทำให้ VCO เปลี่ยนความถี่ทำงานในทิศทาง ซึ่งจะทำให้เฟสเออเรอร์หายไป

มีสามเงื่อนไขที่จำเป็นให้ระบบ PLL แบบเชิงเส้นยังคงรักษาการแทรคกิ้ง (tracking)

1. ความถี่เชิงมุมของสัญญาณอ้างอิงต้องอยู่ภายในพิสัยโฮล (hold range)
2. สเต็ปความถี่สูงสุดจ่ายสู่อินพุตอ้างอิงของ PLL ต้องน้อยกว่าพิสัยพูลเอาต์ (pull-out range)
3. อัตราการเปลี่ยนของความถี่อ้างอิง $\Delta\omega$ ต้องน้อยกว่า ω_n^2

มีการจัดแบ่งพารามิเตอร์ระบุพิสัยความถี่ในการทำงานของระบบ PLL ซึ่งสามารถสรุปเนื้อหาที่สำคัญได้ดังนี้



รูปที่2 โครงสร้างเสถียรภาพทางไดนามิค และสแตติก ของ PLL แบบเชิงเส้น

1. พิสัยโฮล (hold range) $\Delta\omega_H$ ช่วงความถี่นี้ PLL สามารถรักษาการติดตามอยู่กับเฟสได้ ซึ่ง PLL จะมีเสถียรภาพอย่างมีเงื่อนไขในช่วงนี้เท่านั้น
2. พิสัยพลู-เอาต์ (pull-out range) $\Delta\omega_{PO}$ เป็นช่วงที่จำกัดไดนามิคสำหรับการทำงาน PULL อย่างเสถียรภาพ ถ้าการแทรคกิ้งสูญเสียไปภายในช่วงนี้ ระบบ PLL จะล็อกอีกครั้ง แต่กระบวนการจะช้าลง ซึ่งเป็นลักษณะของกระบวนการดึงเข้า (pull in process)
3. พิสัยพลู-อิน (pull-in range) $\Delta\omega_P$ คือช่วงที่ระบบ PLL ล็อกเสมอ แต่กระบวนการค่อนข้างช้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. พิสัยล็อก (lock range) $\Delta\omega_L$ คือช่วงความถี่ที่ระบบ PLL จะทำการล็อกภายในหนึ่งบีต (beat) ระหว่างความถี่อ้างอิงกับความถี่เอาต์พุต ซึ่งโดยปกติเป็นช่วงความถี่การทำงานของระบบ PLL ถูกจำกัดภายในช่วงนี้

ดังนั้นในการออกแบบโดยส่วนมาก เราต้องจัดขนาดพิสัยต่าง ๆ ไม่เท่ากันดังนี้

$$\Delta\omega_L < \Delta\omega_{PO} < \Delta\omega_P < \Delta\omega_H$$

<u>หมายเหตุ</u> เงื่อนไขประโยชน์ที่ได้รับจะเกี่ยวข้องเฉพาะระบบ PLL แบบเชิงเส้นเท่านั้น กรณีที่เป็นระบบดิจิทัล PLL สภาวะเงื่อนไขนี้จะแตกต่างกันออกไป

ในหนังสืออ้างอิงหลายฉบับจะกล่าวถึง เทอมของ "พิสัยแคปเจอร์ (carptur range)" ในบางครั้งสามารถใช้ความหมายเป็น พิสัยพูล-อิน (pull-in range) ซึ่งไม่อาจแยกความแตกต่างกันได้ชัดเจน

2.2 นิยามดิจิทัลเฟสล็อกลูป (Digital Phase-Locked Loop Difinition)

ดิจิทับเฟสล็อกนิยามความหมายได้ยาก เพราะว่าคำว่าดิจิทับนั้นสามารถประยุกต์ใช้ได้หลายแบบอย่าง ในการกำหนดนิยามของดิจิทับเฟสล็อกลูปตามแบบดั้งเดิม (classical) จะต้องมีเทอม ซึ่งมีความหมายของสัญญาอ้างอิง u_1 และสัญญาณเอาต์พุต u_2 เป็นสัญญาณไบนารี โดยเป็นสัญญารูปคลื่นสี่เหลี่ยม อย่างไรก็ดี ภายในดิจิทัลเฟสล็อกลูปแบบดั้งเดิม บล็อกแอนะล็อกยังคงถูกใช้อยู่ ดังเช่น ฟิลเตอร์ RC และ VCO ซึ่งเราสามารถกล่าวว่า ระบบ DPLL แบบดั้งเดิม มีคุณลักษณะของสัญญาณแอนะล็อกเป็นตัวกลางอยู่

หลายปีมาแล้วระบบได้มีการพัฒนาสิ่งต่าง ๆ โดยสร้างจากอุปกรณ์ดิจิทัลอย่างแพร่หลาย ซึ่งไม่พบการใช้สัญญาณแอนะล็อก อย่างเช่น ลูปฟิลเตอร์ (loop filter) ซึ่งไม่ได้เป็น RC ฟิลเตอร์อย่างที่เป็นมานาน แต่ใช้เป็นดิจิทัลฟิลเตอร์แบบออร์เดอร์หนึ่งหรือออร์เดอร์สองแกน ดังนั้นดิจิทัลเฟสล็อกลูปแบบใหม่จึงต่างจากแบบดิจิทัลแบบดั้งเดิมโดยสิ้นเชิง โดยเรียกรวมทั้งหมดว่า "ระบบเฟสล็อกลูปแบบดิจิทัลล้วน (ADPLL)" จะอธิบายต่อไป ระบบ DPLL ยังได้เพิ่มเติมโดยใช้ซอฟต์แวร์บนไมโครโปรเซสเซอร์ ในขั้นนี้จึงเป็นการง่ายที่จะแยก DPLL แบบดั้งเดิมและ ADPLL ออกจากกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ระบบเฟสล็อกลูปแบบดิจิทัลล้วนและแบบใช้ซอฟต์แวร์เป็นหลัก (All-Digital and Software-Based PLLS)

ระบบ ADPLL หรือระบบ PLL แบบดิจิทัลล้วนซึ่งประกอบด้วยบล็อกของดิจิทัลทั้งหมดโดยไม่รวมบล็อกของแอนะล็อก อีกเทอมคำว่า "ดิจิทัล"ยังคงคลุมเครืออยู่ในแง่การอธิบายประกอบบางตอน ซึ่งไม่ได้มีความหมายว่าเป็นบล็อกฟังก์ชันสัญญาณอินพุตหรือเอาต์พุตอย่างใดอย่างหนึ่งโดยเฉพาะ เช่น เป็นสัญญาณลอจิกสูง (high) หรือลอจิกต่ำ (low) เท่านั้น แต่อีกนัยหนึ่งเป็นสัญญาณแบบเฉพาะเช่น ไบต์ หรือดิจิทัลเวิร์ดแบบ n บิต (nบิต digital word) ซึ่งแทนผลต่างสภาวะทั้งหมดเป็น 2^n

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

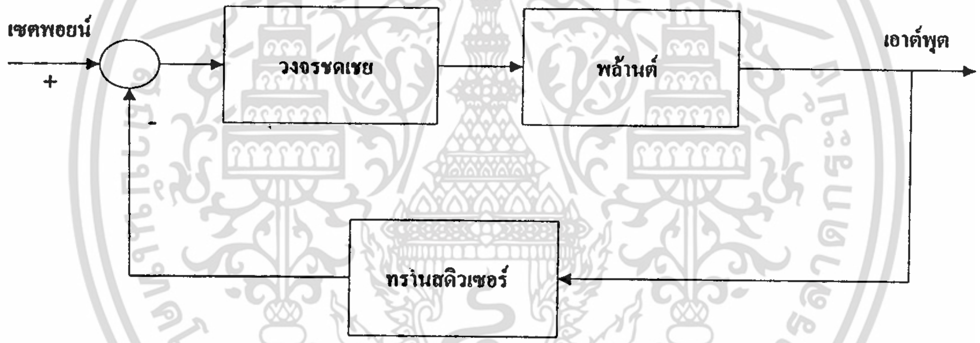
บทที่ 3

โหมดการควบคุมของคอนโทรลเลอร์

3.1 นิยามระบบควบคุม

การนิยามของระบบควบคุม นานมาแล้วการพัฒนาการผลิตในเชิงอัตโนมัติ (automation) ได้ถูกเข้ามาสนใจมากขึ้น ซึ่งเรียกอย่างเดียวกันว่า "ระบบควบคุม" ซึ่งเทอมนี้นั้นมีความหมายหลายแนวความคิด โดยมีจุดประสงค์แตกต่างกันออกไป ทำให้มีเทคนิคต่าง ๆ ที่สำคัญเป็นประโยชน์ต่องานวิศวกรรม โดยมีแนวทางปฏิบัติให้เป็นจริง

ระบบควบคุมสามารถจัดแบ่งเป็นแบบ เซลฟ์เรกูเลชัน (self regulation) ตามรูปที่ 2-1 เป็นบล็อกไดอะแกรมของส่วนประกอบต่าง ๆ ซึ่งเกี่ยวข้องกับเทอมต่างในระบบ



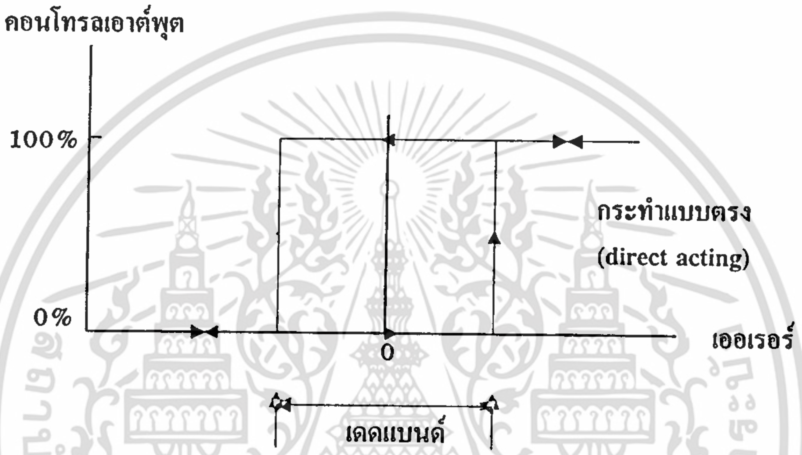
รูปที่1 ส่วนประกอบของระบบควบคุมอย่างง่าย

ระบบต้องการอินพุตคือค่าเซตพอย์น เป็นการเตรียมจุดหมายให้ระบบเซลฟ์เรกูเลชัน การบรรลุถึงจุดหมายนั้น จะต้องใช้ความรู้ในการปฏิบัติคือ การใช้จุดต่อร่วม (summing junction) เพื่อสร้างเออเรอร์จากการป้อนกลับของเอาต์พุต โดยเตรียมการป้อนกลับจากตัวทรานสดิวเซอร์ ถ้าต้องการค่าเออร์เรอร์ลดน้อยลง แล้วค่าป้อนกลับต้องมีค่าผกผันซึ่งเรียกว่า "ป้อนกลับลบหรือ เนกาตีฟฟีดแบค (negative teedback)" เออเรอร์เหล่านี้จะป้อนให้กับวงจรชดเชย (commpensate network) ซึ่งประกอบด้วยตัวขยาย ตัวขับฟิลเตอร์ และอื่น ๆ ซึ่งเรียกว่าการควบคุม (control action) สัญญาณที่ได้จากวงจรนี้จะถูกส่งให้กับพล้านต์ (plant) ต่อไป และนี่คือความหมายของระบบควบคุมโดยสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การควบคุมแบบเปิด-ปิด (ON-OFF Control Action)

เป็นหนึ่งของการควบคุมแบบธรรมดาที่สุดและง่ายที่สุด โหมดนี้มีตัวอย่างตามรูปที่ 2 ซึ่งการควบคุมแบบนี้มีฟังก์ชันถ่ายโอนกับสภาวะเอาต์พุตที่เป็นได้ได้สองสถานะคือ 0 % กับ 100 % ถ้าค่าเออเรอร์เพิ่มขึ้นเกินค่าที่กำหนด เอาต์พุตจะเปลี่ยนจาก 0 % เป็น 100 % แต่ถ้าเออเรอร์ต่ำกว่าค่าที่กำหนดแล้ว เอาต์พุตก็จะลดลงจาก 100 % เป็น 0 % ดังนั้นค่าเดดแบนด์ (dead band) เป็นตัวกำหนดการเปลี่ยนแปลงโดยอ้างที่จุดศูนย์ ตัวอย่างในระบบควบคุมซึ่งให้การควบคุมแบบเปิด/ปิด เช่น ระบบปรับอากาศเป็นต้น ดังนั้นสมการการควบคุมแบบนี้คือ



รูปที่2 กราฟแสดงการควบคุมแบบเปิด-ปิด และเดดแบนด์

3.3 การควบคุมแบบต่อเนื่อง (Continuous Control Action)

เป็นการควบคุม ที่ใช้มากที่สุดในระบบควบคุมกระบวนการแบบต่อเนื่อง เพราะว่ามี เรกูเลชันดีกว่าการควบคุมแบบเปิด-ปิด การควบคุมแบบนี้จัดแบ่งเป็นโหมดต่าง ๆ ดังนี้

3.3.1 โหมดการควบคุมแบบพรอปพรอชันแนล (Proprotional Contro Mode)

การควบคุมแบบพรอปพรอชันแนลเรียกโดยย่อว่า การควบคุมแบบ "P" การทำงานของโหมดนี้ในการควบคุมกระบวนการ เอาต์พุตของการควบคุมแบบพรอปพรอชันแนลจะเป็นสัดส่วนโดยตรงกับค่าเออเรอร์ของตัวเองคูณด้วยค่าคงที่ โดยที่ค่าเออเรอร์เกิดจากการแก้ไขค่าป้อนกลับนั่นเอง ซึ่ง สามารถเขียนสมการได้ดังนี้

$$C_P = K_p e_p + C_0$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

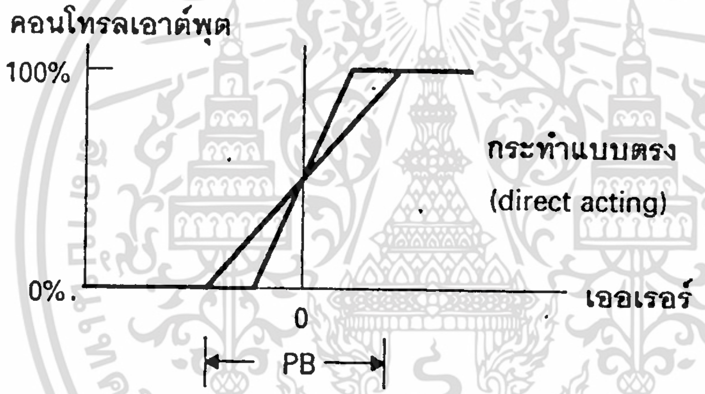
ที่ซึ่ง C_P = เอาต์พุตของคอนโทรลเลอร์ในเปอร์เซ็นต์

K_P = พรอปพรอชันแนลเกนใน เปอร์เซ็นต์ของเอาต์พุต/เปอร์เซ็นต์ของเออเรอร์

e_P = เออเรอร์ในเปอร์เซ็นต์ของช่วง

C_O = เอาต์พุตของเออเรอร์ศูนย์

กราฟของโหมดนี้ถูกแสดงในรูป 2 แสดงค่าเกนสองค่าที่แตกต่างกัน สังเกตว่ามีค่าอิ่มตัว (saturation value) ของเออเรอร์เมื่อเอาต์พุตถึง 100 % โดยที่ค่าเออเรอร์จะมากขึ้นไปกว่านี้นั้น จะไม่มีผลิตเอาต์พุตที่เพิ่มขึ้นมากไปกว่านี้ ซึ่งจะมีผลเหมือนกันเมื่อเอาต์พุตตกลงที่ 0 % แถบเออร์เรอร์ภายในเอาต์พุตระหว่าง 0% ถึง 100% ถูกเรียกว่า "พรอปพรอชันแนลแบนด์ (proportional band) หรือ PB" ที่ค่าเกนมีค่าสูงพรอปพรอชันแนลแบนด์จะมีค่าต่ำ



รูปที่3 คอนโทรลเอาต์พุตในโหมด P จะราบเรียบ ภายในข้อกำหนดของเกนและเออเรอร์

ออฟเซตเออเรอร์ (offset error) เป็นปัญหาอย่างหนึ่งกับการควบคุมแบบ พรอปพรอ- ชันแนล นั่นคือความจริงที่ว่า เออร์เรอร์ศูนย์ (zero error) เป็นค่าคงที่ ค่า C_O ถูกเลือกมาในตอนต้นให้เหมาะสม ค่า C_O เป็นค่านอกเหนือจากการควบคุม โดยเกี่ยวพันกับการเปลี่ยนแปรในกระบวนการ (process) การแปรเปลี่ยนแบบทันใดของตัวแปรอื่น ๆ เหล่านั้น จะเป็นเหตุให้ตัวแปรควบคุมเปลี่ยนด้วย ซึ่งแก้ไขให้ถูกต้องโดยเปลี่ยนค่า C_P คล้องจองกับสมการควบคุม อย่างไรก็ดีถ้าตัวแปรอินพุตเปลี่ยนแปลงแบบที่เรียกว่า "การเปลี่ยนแปลงโหลด" แล้วการควบคุมแบบนี้ไม่ได้แก้ไขไว้ถึงอย่างไรก็สนับสนุนระบบที่มีค่าเออเรอร์ศูนย์ หรือ C_O คงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 โหมดการควบคุมแบบอินทีกรัล (Integal Mode)

เป็นโหมดหนึ่งที่ใช้งานบ่อย ๆ ในระบบการควบคุม โดยมีพื้นฐานเกี่ยวเนื่องกับเหตุการณ์ในอดีตของค่าเออเรอร์ ซึ่งเกิดขึ้นในการแปรผันทางการควบคุม โหมดการควบคุมนี้สามารถเรียกอีกอย่างหนึ่งว่า "โหมดรีเซต (reset)" ถ้าการควบคุมแบบพร้อปพรอชันแนลกำหนดการป้อนกลับบนพื้นฐานของเออเรอร์ในอดีตที่เกิดขึ้น ดังนั้นเป็นไปได้ว่า เมื่อขณะใดขณะหนึ่งถ้าค่าเออเรอร์เป็นศูนย์แล้วการควบคุมแบบอินทีกรัลยังคงถูกป้อนกลับอยู่เพราะเกิดจากเออเรอร์ในอดีต

จากนี้เราสามารถมองเห็นภาพพจน์ของโหมดนี้ เทอมของอินทีกรัลสามารถคำนวณจากพื้นที่สุทธิ (net area) ภายใต้เส้นกราฟเออเรอร์ต่อเวลา การกระทำเอาต์พุตถูกพบว่าเป็นการคูณด้วยค่าคงที่ซึ่งเรียกว่า "อินทีกรัลเกน" ดังนั้น

$$C_I(t) = K_I A_e(t) + C_I(O)$$

ที่ซึ่ง $C_I(t)$ = คอนโทรลเอาต์พุตสำหรับโหมดอินทีกรัล (%) ที่เวลา t

K_I = ค่าเกน [%-time]

$A_e(t)$ = พื้นที่สุทธิของเออเรอร์ต่อเวลา (%-time) คำนวณจากเวลา t จาก = O

$C_I(O)$ = คอนโทรลเอาต์พุตที่เวลา t = O

สมการตามปกติจะเขียนอยู่ในรูปสมการอินทีกรัลดังนี้

$$C_I(t) = K_I \; e_P(t)dt + C_I(O)$$

กระบวนการรีเซต การกระทำอินทีกรัลเตรียมเอาต์พุต "flaot" ด้วยเหตุนี้เพราะค่าเอาต์พุตที่เออเรอร์ศูนย์ (zero error) จะขึ้นอยู่กับเออเรอร์ในอดีต นี้หมายความว่า ในโหมดอินทีกรัลสามารถกำหนดเอาต์พุตอัตโนมัติเพื่อเผชิญการเปลี่ยนแปลงโหลดได้

3.3.3 โหมดการควบคุมแบบดีริเวทีฟ (Derivative Mode)

การกระทำในโหมดนี้ถูกพิจารณาว่า เป็นการปฏิบัติการของการควบคุม อยู่บนพื้นฐานของอัตราเวลาของการเปลี่ยนเออเรอร์ (time rate of change of error) ในกรณีที่เออเรอร์ที่แท้จริงเป็นศูนย์ การกระทำดีริเวทีฟยังจะให้การตอบสนองได้เฉพาะอัตราเปลี่ยนเออเรอร์ ซึ่งขึ้นอยู่กับอัตราการเปลี่ยนเออเรอร์นั้น ทำให้เรียกบ่อย ๆ ว่า "เรตแอคชั่นโหมด (rate action)" การตอบสนองของการปฏิบัติดีริเวทีฟจะไม่ขึ้นอยู่กับค่าเออเรอร์ในขณะนั้นโดยเฉพาะ แต่จะขึ้นอยู่กับอัตราการเปลี่ยน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของค่าเออเรอร์ในขณะนั้น รูปที่ 4 แสดงกรณีที่เออเรอร์เป็นศูนย์ขณะหนึ่ง แต่ว่าเพื่อให้กระจ่าย เอาต์พุตคอนโทรลจำเป็นต้องปรับค่าบางสิ่ง โดยค่าเออเรอร์ที่กำลังเปลี่ยนแปลงแบบรวดเร็วนั้นจะประกันได้ว่าเอาต์พุตคอนโทรลจะไม่เป็นศูนย์เว้นเสียแต่ว่าจะมีเงื่อนไขอื่น

รูปที่4 ในกรณีที่เออเรอร์เป็นศูนย์ แต่เอาต์พุตดีริเวทีฟมีการตอบสนอง

อัตราการเปลี่ยนของเออเรอร์สามารถถูกประมาณ โดยการเอาผลต่างระหว่างค่าเออเรอร์สองค่าหารด้วยเวลาต่างกัน ดังนี้

$$C_D(t) = K_D \frac{e_P(t) - e_P(t_0)}{t - t_0}$$

ที่ซึ่ง $C_D(t)$ = เอาต์พุตการควบคุมแบบดีริเวทีฟที่เวลา t(%)

K_D = ค่าเกนดีริเวทีฟ [% per (%/time)]

$e_P(t)$ = เออเรอร์ที่เวลา t (%)

$e_P(t_0)$ = เออเรอร์ที่เวลา t_0 (%)

สังเกตได้ว่าไม่ว่าไม่มีเทอมเอาต์พุตของการควบคุมออกมา เมื่อมีอัตราที่เปลี่ยนแปลงเออเรอร์เริ่มต้นเป็นศูนย์ ด้วยเหตุนี้การควบคุมโหมดดีริเวทีฟจะไม่ใช้งานแบบโดดเดี่ยว เพราะว่าจะไม่ตอบสนองค่าเออเรอร์คงที่ (fixed error)

การแสดงสมการทางคณิตศาสตร์ที่เหมาะสมสำหรับดีริเวทีฟโหมดเป็นรูปแบบแคลคูลัสคือ

$$C_D = K_D \frac{de_P}{dt}$$

3.3.4 โหมดการควบคุมแบบคอมโพสิต (Composite Mode)

การปฏิบัติการควบคุมในโหมดนี้ โดยปกติจะใช้สามโหมดข้างต้นมารวมกัน โดยนำสมการมารวมกันแบบธรรมดาดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. โหมด PI (proprotional-integral)

$$C_{PI} = K_P e_P + K_P K_I A_e$$

2. โหมด PD (proprotional-derivative)

$$C_{PD} = K_P e_P + K_P K_D \frac{e_P(t) - e_P(t_0)}{t - t_0}$$

3. โหมด PID (proprotional-integral-derivative)

$$C_{PID} = K_P e_P + K_P K_I A_e + K_P K_D \frac{e_P(t) - e_P(t_0)}{t - t_0}$$

สังเกตว่าพรอปพรอชันแนลเกนจะปรากฏในทุก ๆ เทอมของโหมดคอมโพสิต

3.4 บทนิยามอัลกอลิธึมการควบคุม (Control Action Algorithm Definition)

เมื่อได้พิจารณาปัญหาของคอมพิวเตอร์เมื่อเตรียมการใช้กับงานควบคุม โดยสำหรับข้อมูลอินพิต. และเอาต์พุตการควบคุมที่แปรผัน ซึ่งได้รับอิทธิพลจากธรรมชาติของดิสครีต (discrete) บทนี้ในปัญหาทั่ว ๆ ไปของการควบคุมกระบวนการ นั้นเป็นสิ่งแปรเปลี่ยนไปอย่างต่อเนื่อง ดังนั้นเราจะได้ศึกษาต่อไป ซึ่งมองดูผิวเผินในการประยุกต์ของคอมพิวเตอร์ในงานควบคุมของกระบวนการต่อเนื่อง จะต้องแทนข้อมูลในรูปดิสครีต นี่คือปัญหาที่ต้องจัดเตรียม โดยการใช้โค๊ดแทนข้อมูลสำหรับให้คอมพิวเตอร์ประมวลผลหลักการเปลี่ยนข้อมูลโปรเซสเป็นข้อมูลดิจิทัลไม่มีอะไรต้องวิตกเพราะเป็นความสามารถทางฮาร์ดแวร์อย่างไรก็ดีต้องจำไว้ว่าการใช้คอมพิวเตอร์
ใช้ประมวลผลในการปฏิบัติกับรูปแบบการควบคุมต่อเนื่อง โดยการใช้ดิสครีตแทนนั้นจะยุ่งยากพอควร

ต่อไปต้องทำความรู้จักกับองค์ประกอบพื้นฐานดังนี้

1. สมการควบคุม (Control Equation)
2. ค่าเออเรอร์ของตัวแปรควบคุม (Controlled Variable Error)
3. ขั้วเครื่องหมายของ DE (Polarity of DE)
4. เอาต์พุตตัวแปรการควบคุม (Controlling Variable Output)

องค์ประกอบพื้นฐานเหล่านี้สามารถอธิบายได้ดังนี้

1) สมการควบคุม (Control Equation)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัตถุประสงค์พื้นฐานของหัวข้อนี้ ในการแสดงสมการการควบคุมในโหมดต่าง ๆ เป็นอย่างเช่น เราเคยเสนอไว้ในตอนต้น โดยที่เราต้องเพิ่มเติมบางสิ่งขึ้นมาเพื่อใช้กับคอมพิวเตอร์ นี่เอง หมายความว่าจะต้องมีอัลกอลิธึม ซึ่งจะเตรียมหาไว้ในการแก้ปัญหาทางสมการควบคุม ในความเข้าใจทั่วไปสมการจะลดลงเหลือในรูปสมการ

$$DC = F(DV) \tag{3-1}$$

เมื่อ DC = เอาต์พุทดิจิทัลของค่าตัวแปรควบคุม (controlling variable)

DV = อินพุตดิจิทัลจากค่าตัวแปรควบคุม

สมการที่ 3-1 เป็นการแทนดีสครีตแทนการแปรผันต่อเนื่อง โดยที่คอมพิวเตอร์จะต้องใช้ไบต์ข้อมูลที่มีบิตจำนวนจำกัดเป็นตัวแปร ตัวอย่างเช่น ถ้าเป็น 8 บิตตามสมการที่ 3-1 สามารถแทนข้อมูลด้วยความละเอียดเพียง 1 ใน 256 เท่านั้น ซึ่งต้องรู้จักข้อจำกัดนี้เป็นผลที่ตามมาภายหลังในกระบวนการควบคุม

2) ค่าเออเรอร์ของตัวแปรควบคุม (Controlled Variable Error)

เริ่มแรกในข้อปฏิบัติ โดยเพิ่มเติมกระทำกับสมการที่ 3-1 เพื่อหาค่าเออเรอร์ของตัวแปรควบคุม DV จากตัวแปรเซตพอย์นที่กำหนดขึ้นคือ DSP อย่างไรก็ดีมีหลายวิธีที่แสดงถึงค่าเออเรอร์ ซึ่งสำหรับในหัวข้อนี้ และความเข้าใจทั่ว ๆ ไปในอัลกอลิธึมในการควบคุม เราสามารถจะนิยามเพื่อหาค่าเออเรอร์ได้ เว้นเสียแต่การแทนค่าจริงของตัวแปร เช่นการเปลี่ยนข้อมูลจากระบบแอนะล็อกของกระแส 4 - 20 มิลลิแอมป์ ในกรณีการควบคุมแบบดิจิทัลช่วงกายภาพของตัวแปรในคอมพิวเตอร์ จะต้องสมมุติเป็นข้อมูลเต็มการนับในไบต์ของข้อมูล ซึ่งเป็นข้อมูล 8 บิตโดยมีช่วงระหว่าง 00 - Ffh โดยเลือกค่าเออเรอร์ให้ง่ายต่อ การแสดงเหมือนเป็นเศษส่วนของช่วง (fraction of range) ด้วยการใช้การนิยมของเปอร์เซ็นต์ของช่วง โดยสมการดังนี้

$$eP = 100 \frac{V - VSP}{Vmax - Vmin} \tag{3-2}$$

ที่ซึ่ง VSP คือ ตัวแปรควบคุมเซตพอย์น

สิ่งแรกจะต้องทำคือเออเรอร์ต้องถูกกำหนดโดยค่าแตกต่างระหว่างค่าวัดได้ กับเซตพอย์นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$DE = DV - DSP \tag{3-3}$$

ค่าเออเรอร์ DE จะเป็นจำนวนดิจิทัลแทนในรูป 2's คอมพลีเมนต์ ซึ่งหมายความว่าถ้า DE เป็นจำนวนลบจะอยู่ในรูป 2's คอมพลีเมนต์

ในการสอดคล้องกับการนิยามของสมการที่ 3-2 ค่าเออเรอร์สามารถแสดงอัตราส่วนของช่วงการหารของเออเรอร์ในช่วง

$$DFE = \frac{DE}{RIN} \tag{3-4}$$

ซึ่ง RIN = ช่วงตัวแปรอินพุต = DMAX - DMIN

DFE = ค่าเออเรอร์โดยเป็นสัดส่วนกับช่วง (range)

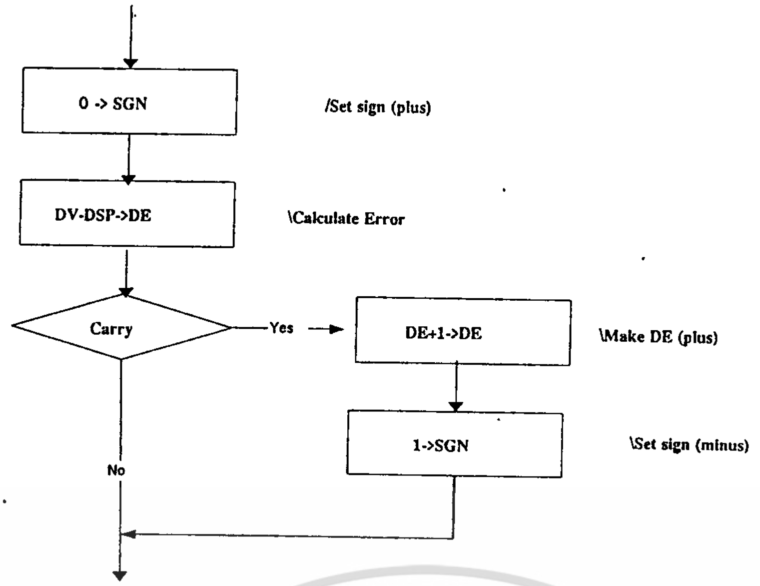
โดยทั่วๆ ไปเราจะใช้เป็นช่วงเต็มของอินพุตคือ FFH ซึ่งเป็นการเริ่มต้นง่าย ๆ ของสมการเพื่อให้ได้ผลลัพท์ที่ดีที่สุด ผลที่แตกต่างออกไปจะต้องพิจารณาช่วงการทำงานด้วย

3) ขั้วเครื่องหมายของ DE (Polarity of DE)

ค่าของ DE คำนวณจากสมการที่ 3-3 จะมีขั้วเครื่องหมายได้สองแบบ หมายความว่าถ้า DV มากกว่า DSP แล้ว DE จะมีเครื่องหมายเป็นลบและแทนใน 2's คอมพลีเมนต์ ซึ่งอาจจะเป็นการไม่สะดวกในการใช้แทนคำสั่งของกระบวนการคำนวณของไมโครโปรเซสเซอร์ เพื่อให้ได้เอาต์พุตออกมาเป็น 2's นั้นจะต้องกำหนด DE ต้องใช้เป็นต่า (value) ใด ๆ ไม่ติดขั้วเครื่องหมาย (polarity) โดยที่ขั้วเครื่องหมายจะแทนเป็นคำแฟลก

โฟล์วชาร์ตตามรูปที่ 5 เป็นการแสดงถึงการทำงานโดยทั่วไปของโปรแกรม ซึ่งเป็นไปตามความจริงที่ว่าถ้าผลลัพท์เป็นลบ บิดทด carry bit) จะถูกเซต และผลลัพท์จะเป็น 2's คอมพลีเมนต์ในกรณีนี้แฟลก SGN จะถูกเซตเป็น 1 ซึ่งจะชี้ว่าค่า DE จะต้องเป็นค่าลบ โดยที่ตัวมันเองจะไม่แสดงเครื่องหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่5 เป็นโฟล์วชาร์ตแสดงค่า DE ทั้งขนาดและเครื่องหมาย

4) เอาต์พุตตัวแปรการควบคุม (Controlling Variable Output)

มีรูปแบบการกำหนดตัวควบคุมและค่าเออเรอร์ โดยคงสภาพรูปแบบที่กำหนดไว้เพื่อใช้เป็นเอาต์พุตตัวแปรควบคุม ซึ่งเอาต์พุตตัวแปรควบคุมกำหนดอย่างง่ายเป็นค่าเปอร์เซ็นต์ 0% ถึง 100 % เช่นในลูปควบคุมทางแอนะล็อกจะแปรค่ากระแสไฟฟ้า 4 - 20 มิลลิแอมป์ หรือถ้าเป็นสัญญาณนิวเมติกคือ 3 PSI ถึง 15 PSI สำหรับกรณีทางดิจิทัลเราจะกำหนดสมการมีผลลัพท์ DC อยู่ในช่วงจาก 00H คล้องจองกับ 0% ถึง FFH คล้องจองกับ 100%

3.5 อัลกอลิธึมของโหมดควบคุม (Pure Mode Control Algorithm)

หมวดนี้จะเสนออัลกอลิธึมของคอมพิวเตอร์ ซึ่งทำงานโหมดคอนโทรลเลอร์แบบพื้นฐานขยายเพิ่มเติม คือ พรอปพร์อชันแนล อินทีกรัล(รีเซต) และดีริเวทีฟ(เรต) ในแต่ละกรณีอัลกอลิธึมจะบรรยายโดยสมการ และโฟล์วชาร์ตของรหัสที่ใช้จริงของไมโครโปรเซสเซอร์ ซึ่งไม่ใช่เรื่องยากที่จะรวมเอาโหมดการควบคุมเหล่านั้นมาประยุกต์ใช้งานจริง

3.5.1 โหมดพรอปพรอชันแนล (Proportional Mode)

โหมดพรอปพรอชันแนลค่าเอาต์พุตจะเป็นสัดส่วนโดยตรงกับค่าเออเรอร์ซึ่งสมการจะเป็น

$$C = K_P e_P + C_0$$

อัลกอลิธึมสำหรับใช้ของสมการในไมโครคอมพิวเตอร์คอนโทรลมีรูปแบบคล้ายคลึงกันคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$DCP = (KP \bullet DFE \bullet ROUT) + DCO \qquad (3\text{-}5)$$

ซึ่ง DCP = ดิจิทัลคอนโทรลเอาต์พุต

KP = ดิจิทัลพรอปพรอชั นแนลเกน

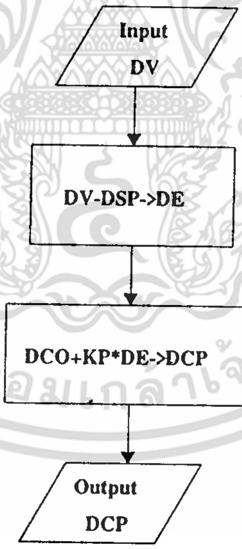
DFE = ดิจิทัลเออเรอร์กับสัดส่วนของ

DCO = เออเรอร์เริ่มต้นของเอาต์พุต

ROUT = ช่วงของเอาต์พุต

ถ้าตัวแปรอินพุตคอนโทรลเท่ากับค่าเซตพอย์น ดังนั้นค่าเออเรอร์จะมีค่าเท่ากับศูนย์ของตัวเอาต์พุต DCO ค่าดิจิทัลเออเรอร์จะกำหนดจากสมการที่ 3-4 และเอาต์พุตการควบคุมจะแปรค่าครอบคลุมช่วงที่ระบุโดย ROUT ค่าของ KP ถูกนิยามเหมือนกับอัตราส่วนการเปลี่ยนต่อค่าเออเรอร์กับช่วงอินพุต ดังนั้นผลจาก KP ● DFE คืออัตราส่วนของเอาต์พุตที่เปลี่ยนไประบุโดยค่าเออเรอร์ ดังนั้นจำนวนเท่าของช่วงเอาต์พุตคือ เอาต์พุตที่เปลี่ยนอย่างแท้จริง สมการนี้สามารถแสดงเป็นรูปฟอร์มที่ไม่มี DFE ดังนี้

$$DCP = KP*DE*(ROUT/RIN)+DCO$$



รูปที่6 โฟล์วชาร์ตอัลกอลิธึมของโหมดพรอปพรอชันแนล

ตอนนี้สามารถมองเห็นเหตุผลความต้องการสำหรับที่จะทำช่วงอินพุต และเอาต์พุตให้เท่ากัน ถ้า ROUT = RIN แล้ว สมการพรอปพรอชันแนลโหมดจะพัวพันเฉพาะค่าเกน ค่าเออเรอร์และเทอมเออเรอร์ที่ศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) สมการอัลกอลิธึมอย่างง่ายของโหมด P (Simplified Algorithm Equation)

สำหรับจุดประสงค์ของหัวข้อนี้ จะต้องสมมุติว่าช่วงของอินพุตและเอาต์พุตถูกเลือกให้มีค่าเท่ากัน ซึ่งตัวเลือกนี้โดยปกติจะนับได้เต็ม 1 ไบต์ข้อมูลถ้าเป็นไมโครโปรเซสเซอร์ 8 บิต จะได้ช่วงถึง FFH ซึ่งจะมีสมการที่ต้องตัดบางส่วนออกไปดังนี้

$$DCP = (KP \bullet DE) + DCO \qquad (3\text{-}7)$$

เมื่อ

DCP = เอาต์พุตคอนโทรล

DE = DV - DSP = เออเรอร์คอนโทรล

DCO = เอาต์พุตเออเรอร์ที่ศูนย์

KP = ค่าเกน

ข้อได้เปรียบของสมการนี้ไม่มีรูทีนการทหารทางคณิตศาสตร์ ซึ่งค่า KP จะเป็นจำนวนกับค่าที่ถูกกำหนดออกแบบทางแอนะล็อกคอนโทรล

2) โฟล์วชาร์ตของโหมด P

โดยปกติโฟล์วชาร์ตพรอปพรอชันแนลโหมดจะแสดงตามรูป 6 ซึ่งแสดงรูปแบบฟอร์แมตตามอัลกอลิธึม สังเกตว่าพรอปพรอชันแนลเกนสามารถจะเป็นได้ทั้งบวกและลบ นั่นหมายความว่าเออเรอร์อาจจะทำให้ DCP เพิ่มขึ้นหรือลดลง มีโฟล์วชาร์ตซึ่งแสดงรายละเอียดมากขึ้นคือรูปที่ 7 โดยได้พิจารณาส่วนอื่น เป็นความสำคัญในโครงสร้างการปฏิบัติ ในการควบคุมพรอปพรอ-ชันแนล

3) การใช้งานเชิงปฏิบัติ (Practical Application)

เมื่อเขียนโปรแกรมอย่างจริงจัง เพิ่มเติมจากสมการที่ 3-7 ในพรอฟพรอชันแนลโหมดมีข้อพิจารณาในการปฏิบัติมากมายโฟล์วชาร์ตตามรูปที่ 3-7 แสดงเปิดเผยให้เห็นถึงรายละเอียดที่มากขึ้นของอัลกอลิธึมของพรอปพรอชันแนลโหมด ดังต่อไปนี้

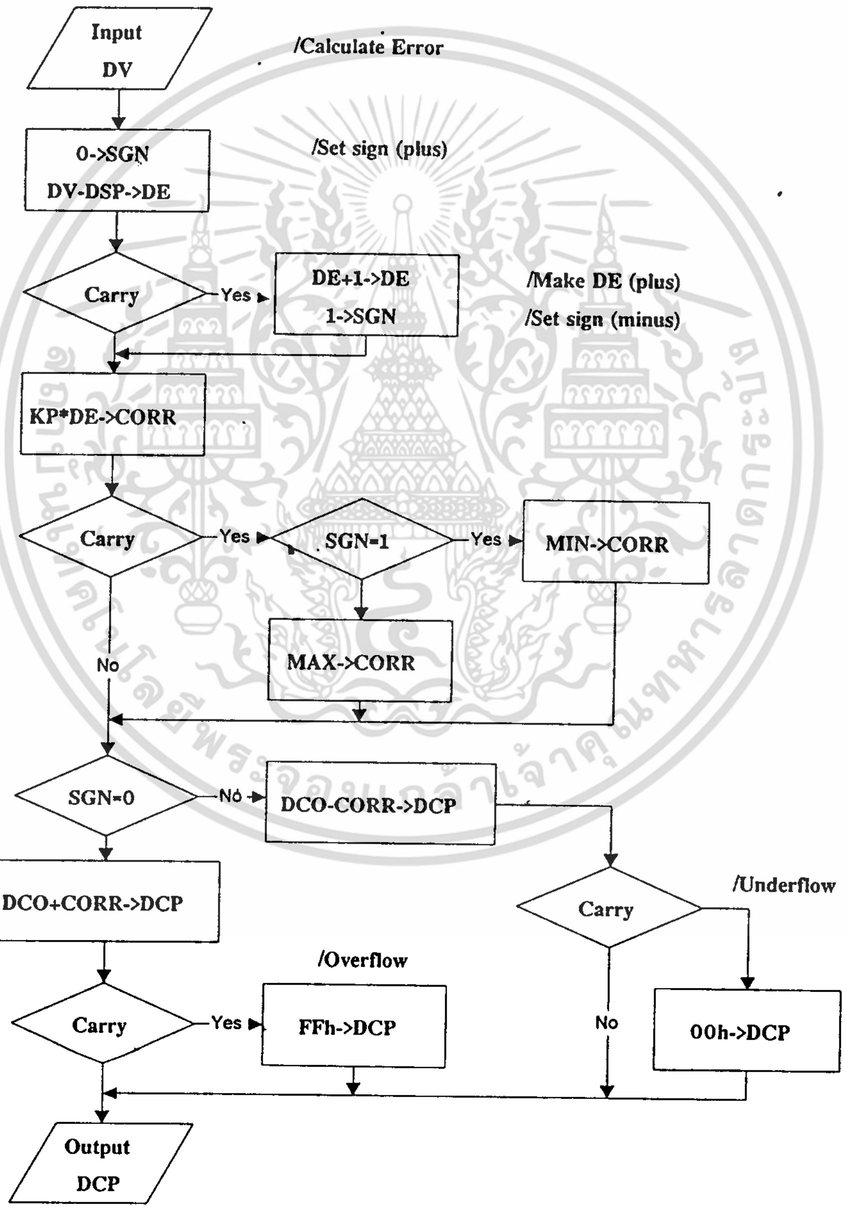
1. ขั้วเครื่องหมายของ DE (Polarity of DE) เหมือนกับตอนต้น เป็นความสะดวกที่ใช้ DE แบบไม่มีเครื่องหมายกับเครื่องหมายที่ชี้โดยแฟลก (flag) ในโฟล์วชาร์ตรูป 7 โดยได้ร่วมหน้าตานี้เข้ากับอัลกอลิธึมของพรอปพรอชันแนลด้วย

2. ผลของโอเวอร์โฟล์ว (Product overflow) เมื่อ DE ถูกคูณด้วย KP มันเป็นไปได้ทีเดียวจะมีเงื่อนไขของโอเวอร์โฟล์วเกิดขึ้น นั่นหมายความว่าค่าของผลคูณนั้นเกินกว่าค่าไบต์ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเป็น 8 บิตนั้นแฟลกทด (carry flag) จะถูกเซต ดังรูปที่ 7 เมื่อเหตุการณ์นี้เกิดขึ้นจำเป็นต้องเซตค่า KP ● DE เป็นแฟกเตอร์คอเร็กชั่น (correction factor) ซึ่งจะเท่ากับค่าสูงสุดและต่ำสุด

3. ผลรวมโอเวอร์โฟล์วและอันเดอร์โฟล์ว (Sum Overflow anf Underflow) ในกรณีทั่วไป DCP จะเป็นจำนวนเต็มบวกในช่วงของ ROUT จะแปรไปตั้งแต่ 00 ถึง FFH และแก้ไขให้ถูกต้อง (correction) โดยนำมาลบหรือรวมเข้ากับค่าเออเรอร์ที่ศูนย์ DCO ผลลัพท์ของ DCP แสดงในสมการที่ 7 โอเวอร์โฟล์วเกิดขึ้นโดยผลรวมเป็นผลลัพท์ที่เกินกว่า 8 บิต และผลลบที่ต่ำกว่า 8 บิต คือน้อยกว่า 00H ซึ่งจำเป็นที่จะให้ซอฟต์แวร์ตรวจสอบโอเวอร์โฟล์วและอันเดอร์โฟล์วเพื่อเตรียมแก้ไข



รูปที่7 รายละเอียดโฟล์วชาร์ตของโหมด P แสดงแฟลกเครื่องหมาย และการป้องกันโอเวอร์โฟล์วอันเดอร์โฟล์ว

เอกสารนี้เป็นเอกสารที่สงวนไว้ ... ญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โอเวอร์โฟล์วเกิดขึ้นเมื่อ DCO ถูกรวมเข้า และจะทำให้บิตทดถูกเซต อันเดอร์โฟล์วเกิดขึ้นเมื่อ DCO ถูกนำไปลบซึ่งบิตทดจะถูกเซตด้วยเช่นกันแสดงดังรูปที่ 2-7 การตอบสนองของอันเดอร์โฟล์ว และโอเวอร์โฟล์วคือ ค่าสูงสุดและต่ำสุดเป็น 00H และ FFH

4. การกระทำแบบตรงและแบบผกผัน (Direct or Reverse Action) เครื่องหมายของ KP จะเป็นตัวกำหนดทิศทางของการควบคุมแบบตรงหรือผกผัน แบบตรงหมายความว่า ถ้าเออเรอร์เป็นค่าบวกผลลัพท์ของตัวแปรเอาต์พุตจะเพิ่มขึ้น และในทางตรงกันข้ามจะลดลงถ้าเป็นแบบผกผันนั้นสามารถเตรียมตัวแปรที่กำหนดในบล็อก โดยเมื่อมีการบวกหรือลบกับ DCO จะกระทำตามนั้นแต่ถ้าเป็นแบบผกผันจะกระทำบวกหรือลบจะทำในทางตรงกันข้าม

3.5.2 โหมดอินทีกรัลหรือโหมดรีเซต (Integral Mode Reset)

ในบทแรกโหมดอินทีกรัลถูกกำหนดเหมือนการตอบสนองการควบคุม โดยจะขึ้นอยู่กับค่าเออเรอร์ในอดีตมากกว่าเออเรอร์ของตัวมันเองในขณะนั้น ในอดีต (history) ถูกวัดได้โดยพื้นที่ในส่วนที่กำหนด หรือเรียกว่า "พื้นที่สุทธิ (Net area)" ซึ่งเออเรอร์ที่วัดได้จะอยู่ในเส้นโค้งของเวลาระหว่างเวลาเริ่มต้นกับเวลาปัจจุบัน ในทางคณิตศาสตร์จะเท่ากับการหาค่าอินทีกรัลของเออเรอร์ ซึ่งเหตุนี้จึงถูกขนานนามว่าอินทีกรัลโหมด และมันถูกเรียกว่ารีเซตเพราะว่าโหมดอินทีกรัลเป็นความจุของออฟเซตเออเรอร์ในพรอปพรอชันแนลโหมด โดยที่ "รีเซต" คือเทอมของเออเรอร์ที่ศูนย์

สมการของการคำนวณของโหมดอินทีกรัลคือ

$$C_I(t) = K_I A_e(t) + C_I(O)$$

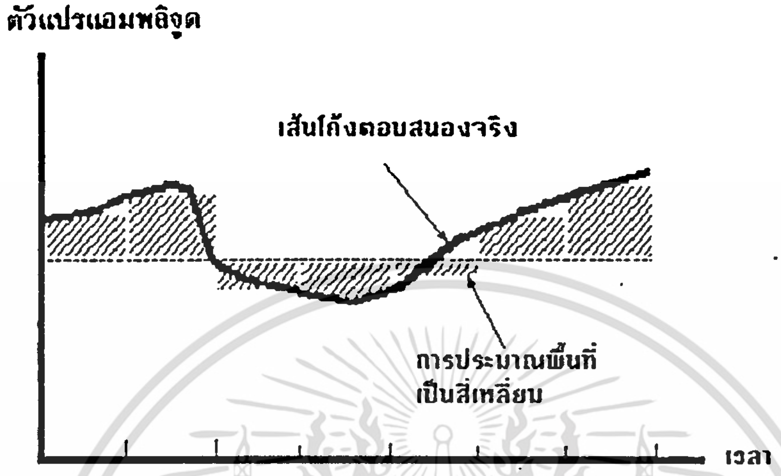
ซึ่ง K_I คือ อินทีกรัลเกน (integal gain) และ $A_e(t)$ คือพื้นที่เออเรอร์ (error area) ซึ่งแปรผันตามเวลา ซึ่งความสัมพันธ์ของพื้นที่มีความสำคัญคือ พื้นที่จะเพิ่มขึ้นเมื่อเออเรอร์เป็นบวก (positive) และพื้นที่จะลดลงเมื่อเออเรอร์เป็นลบ

1. การประมาณพื้นที่ของเออเรอร์ (Error Area Approximation)

วิธีเพิ่มเติมโหมดนี้โดยใช้คอมพิวเตอร์ ซึ่งจะต้องหาพื้นที่โดยใช้ค่าตัวแปรอินพุตแซมปลิงมีหลายวิธีที่จะทำเช่นนี้ ในเอกสารฉบับนี้จะใช้วิธีที่ง่ายคือ การอินทีกรัลแบบรูปสี่เหลี่ยม (rectangular integration) แสดงตัวอย่างรูปแบบในรูปที่ 8 อินพุตของคอมพิวเตอร์เป็นส่วนประกอบของแซมปลิงที่จุดเส้นโค้งของตัวแปรควบคุม (controlled varialble) กับเวลา เออเรอร์ในแต่ละกรณีจะถูกหาโดยวัดความแตกต่างของค่าและเซตพอย์น รูปที่ 8 เป็นการแสดงด้วยว่าแต่ละค่าของเออเรอร์ได้ใช้สร้างเป็นรูปสี่เหลี่ยมมีเซตพอย์นเป็นฐานอ้างอิง ดังนั้นจุดสูงสุดของรูปสี่เหลี่ยมคือค่า

เอกสารนี้เป็นลิขสิทธิ์ที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เออเรอร์ และความกว้างคือเวลาแซมปลิง เมื่อค่าตัวแปรมากเกินเซตพอย์นโดยที่ค่าเออเรอร์จะเป็นบวก และพื้นที่ใต้กราฟก็จะเป็นบวก ในช่วงความกว้างของเวลาแซมปลิง เมื่อตัวแปรมีค่าน้อยกว่าเซตพอย์น ค่าเออเรอร์จะเป็นลบและพื้นที่จะเป็นค่าลบเช่นกัน พื้นที่สุทธิ (net area) จะเป็นผลรวมระหว่างจำนวนบวกหรือจำนวนลบของพื้นที่ ซึ่งเป็นการประมาณค่าพื้นที่ของเส้นโค้งกับเวลาดังนั้นความเที่ยงตรงจะขึ้นอยู่กับการกำหนดค่าเวลาแซมปลิงให้เล็กที่สุด



รูปที่8 กราฟแสดงตัวอย่างพื้ที่เออเรอร์ต่อเวลาโดยประมาณพื้ที่เป็นสี่เหลี่ยม

2. การกำหนดเวลาแซมปลิง (Sample Time)

เวลาแซมปลิงที่น้อยที่สุดถูกปรารถนาที่จะใช้ในการประมาณค่าของพื้นที่ เพื่อให้ได้เที่ยงตรงมากที่สุด ความคิดที่ว่าจะให้เวลาแซมปลิงน้อยที่สุดนั้นประกอบด้วยความถี่ในการแซมปลิงต้องให้มากที่สุดที่จะเป็นไปได้ โดยข้อมูลที่แซมปลิงมาจะไม่เออเรอร์ ดังนั้นความถี่การแซมปลิงต้องประมาณ 10 เท่าของความถี่สูงที่สุดเพื่อหลีกเลี่ยงค่าเออเรอร์ในคาบเวลา และให้เหมาะกับโครงสร้างของพื้นที่ด้วย เพื่อให้ค่าเฉลี่ยของสัญญาณจะไม่เปลี่ยนแปลงมากในแต่ละคาบ ดังนั้นการประมาณค่าในแต่ละคาบจะเที่ยงตรงทีเดียว

3. อัลกอลิธึมของโหมดอินทีกรัล

ทั้งหมดยังคงเหลือว่าพื้นที่ซึ่งหามาเป็นสี่เหลี่ยมนั้น จะใช้ในการคำนวณในโหมดอินทีกรัลได้อย่างไร ถ้าตัวแปรทั้งหมดถูกกำหนดเป็นพื้นที่สุทธิใต้เส้นโค้ง ของเวลาแล้วสมการที่เหมาะสมในอินทีกรัลโหมดสำหรับระบบดิจิทัลคือ

$$DCI = KI \bullet AREA \bullet ROUT \quad (3\text{-}8)$$

เมื่อ DCI = เอาต์พุตอินทีกรัลโหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

KI = เกนอินทีกรัลโหมด

AREA = การประมาณพื้นที่ของเส้นโค้ง

ROUT = ช่วงตัวแปรเอาต์พุต

สังเกตว่าไม่มีค่าคงที่รวมอยู่ในสมการนี้เป็นเพราะว่าในคอมพิวเตอร์นั้นค่า DCI จะแทนเป็นเรจิสเตอร์ หรือตำแหน่งของหน่วยความจำ โดยที่ถูกคำนวณจะถูกเปลี่ยนแปลงค่าไปในแต่ละค่าพื้นที่ที่แซมปลิง ถ้ามีค่าเริ่มต้นต้องโหลดลงไปในหน่วยความจำก่อนที่โปรแกรมเริ่มต้นทำงาน

สมการที่ 3-8 สมมุติว่าช่วงของคอนโทรลเลอร์ถูกกำหนดโดย ROUT ดังนั้น KI ● AREA แทนสัดส่วนช่วง (range) ซึ่งเป็นผลลัพท์จากโหมดอินทีกรัล ถ้าพื้นที่สุทธิทั้งหมดเป็นศูนย์เทอมสมการนี้จะไม่มีผลลัพท์ใด ๆ ทั้งสิ้น

การประมาณพื้นที่ถูกคำนวณ โดยการรวมพื้นที่สี่เหลี่ยมทั้งหมดจากเวลาซึ่งกระบวนการเริ่มสตาร์ตเมื่อ $t = 0$ ดังนี้

$$AREA = AREA + (DFE \bullet DT) \tag{3-9}$$

เมื่อ DFE คือ เออเรอร์ซึ่งเป็นสัดส่วนกับช่วง (range)

DT คือ เวลาระหว่างการแซมปลิง

ความคิดของสมการอย่างหลังนี้ เป็นพื้นที่ถูกปรับปรุงด้วยแต่ละแซมปลิง โดยการรวมพื้นที่ ซึ่งแทนด้วยพื้นที่เดิม ซึ่งถูกคำนวรก่อนหน้านี้

4. การใช้สมการของโหมดอินทีกรัลให้ง่ายขึ้น (Simplication)

นำความคิดก่อนหน้าของเราโดย การใช้ช่วงอินพุตและเอาต์พุตเหมือนกัน ซึ่งสามารถทำสมการเดิมให้ง่ายลง ซึ่งทำแบบนี้เราต้องการทำให้ สมการนี้แสดงช่วง RIN และ ROUT อย่างชัดแจ้งด้วย และกำหนดตัวแปรใหม่คือ SUM แทนผลบวกของ ค่าเออเรอร์ทั้งหมด

$$SUM = SUM + DE \tag{3-10}$$

ดังนั้นพื้นที่จะแสดงอยู่ใต้รูป

$$AREA = SUM \bullet DT/RIN \tag{3-11}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการที่ 2-11 แทนในสมการ 2-8 เป็นผลดังนี้

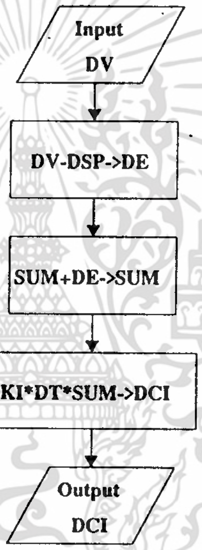
$$DCI = KI \bullet SUM \bullet DT \bullet ROUT / RIN \quad (3\text{-}12)$$

ดังนั้นถ้าทำช่วงให้เท่ากันโดยให้ RIN = ROUT สมการทั้งหมดของโหมดอินทีกรัลจะได้เป็นดังนี้

$$DE = DV - DSP \quad (3\text{-}3)$$

$$SUM = SUM + DE \quad (3\text{-}10)$$

$$DCI = KI \bullet SUM \bullet DT \quad (3\text{-}13)$$



รูปที่9 โฟล์วชาร์ตโดยทั่วไปของโหมดอินทีกรัล

การแทนสมการแบบนี้จะเห็นได้ว่า KI เหมือนกับว่าเป็นตัวลดทอน ของระบบแอะนะล็อกโฟล์วชาร์ตสำหรับเซตของสมการถูกกำหนดโดยรูปที่ 9 ซึ่งเหล่านี้มีผลในการประยุกต์ใช้ในงานควบคุมโดยตรง

5. รายละเอียดของโฟล์วชาร์ต (Detailed Flowchart)

รายละเอียดโฟล์วชาร์ตของโหมดอินทีกรัล ต้องรวมการป้องกันการเกิดโอเวอร์โฟล์ และอันเดอร์โฟล์ว รูปที่ 10 แสดงโฟล์วชาร์ตซึ่งรวมเอาไว้ทั้งหมด เครื่องหมายของ DE ใช้วิธีเดียวกับที่ได้อธิบายรูป 5 ไปแล้วด้วยค่าแฟลกเครื่องหมาย SGN และ DE ซึ่งไม่มีเครื่องหมาย

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. โอเวอร์โฟล์วและอันเดอร์โฟล์วของ SUM

โปรแกรมโหมดอินทีกรัล โอเวอร์โฟล์ว และอันเดอร์โฟล์วคือตำแหน่งในหน่วยความจำ ซึ่งค่าตัวแปรจะเพิ่มขึ้นเมื่อเออเรอร์เป็นบวก (positive) และลดลงเมื่อเออเรอร์เป็นลบ (negative) ซึ่งกระบวนการต่อเนื่องมีโอกาสข้อมูลของเออเรอร์บวก ทำให้ค่า SUM ถูกนับจนเต็มจำนวนบิตที่ใช้ในกรณีนี้ ผลบวกของ SUM จะทำให้โอเวอร์โฟล์ว ผลลพัทะของกรณีนี้ SUM จะถูกลดทอนลงมาโดยแทนจำนวนสูงสุดเข้าไปใน SUM เหตุนี้สามารถเป็นเงื่อนไขสำหรับในระบบควมคุมได้

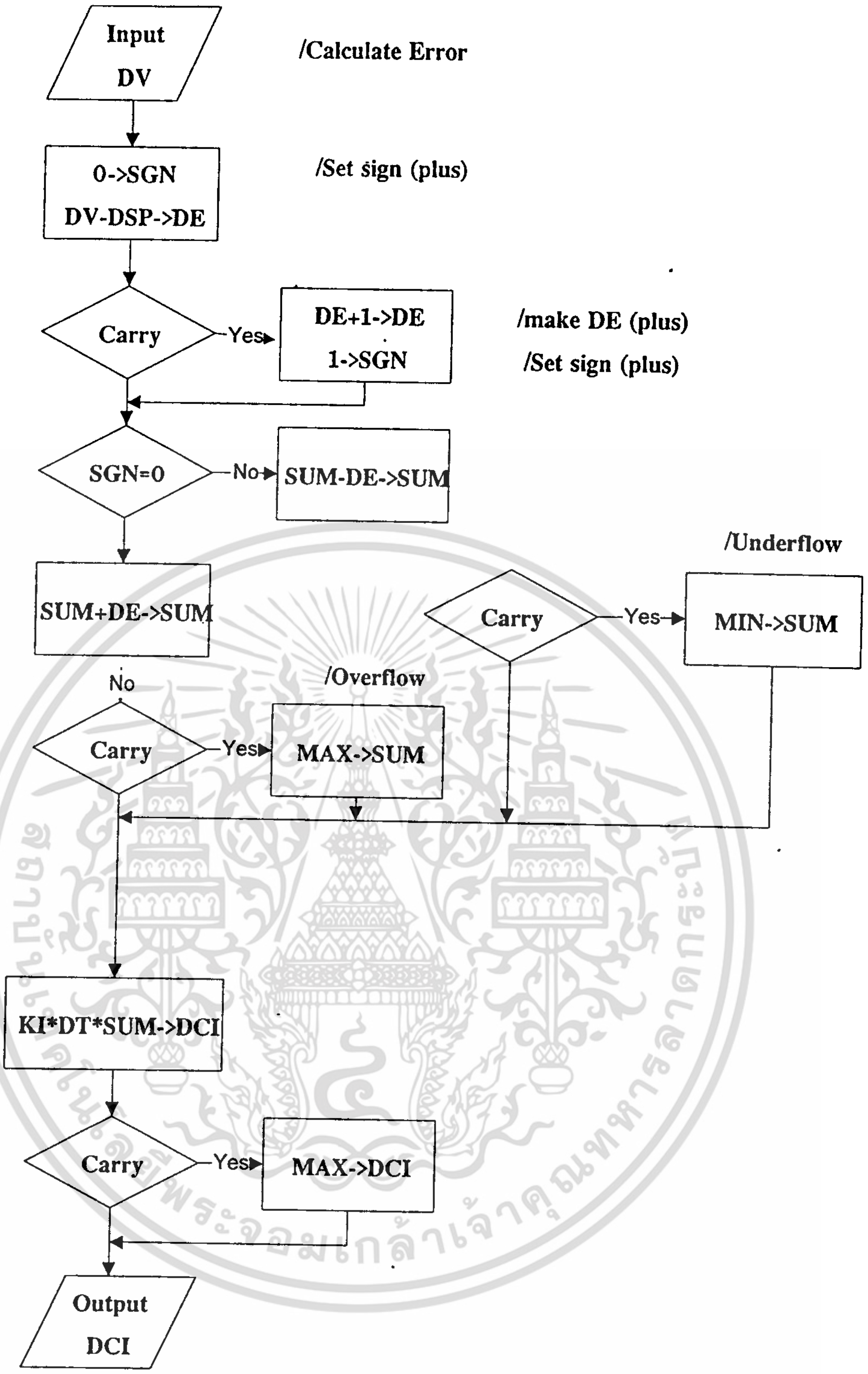
ผลลัพท์ของอันเดอร์โฟวล์นั้น เมื่อการผลของค่าเออเรอร์จาก SUM ● KI ไม่สามารถเป็นจำนวนลบได้ ซึ่งการนับในหน่วยความจำเป็นศูนย์ ถึง จำนวนสูงสุดเท่านั้น อย่างไรก็ดีถ้าค่าเออเรอร์ยังคงเป็นลบเกินค่าน้อยที่สุดของ SUM จะเป็นอันเดอร์โฟล์วจะทำให้จำนวน SUM เพิ่มมากขึ้นจากค่าน้อยที่สุด แต่ทิศทางของการรวมจะกลายเป็นตรงกันข้ามแทน

ถ้าสังกตรูป 10 อันเดอร์โฟล์วนั้นจะถูกกำหนดโดยแฟลกทด ซึ่งแปรตามการลบของ DE จาก SUM และโอเวอร์โฟล์วจะถูกกำหนดโดยแฟลกทด ซึ่งแปรตามการบวกของ DE เช่นกัน ในทั้งสองกรณี SUM ต้องถูกเซตให้จำกัดค่าไว้ทั่ว ๆ ไป จะเป็นจำนวนจากสูงสุดถึงต่ำสุด โดยไม่มีเครื่องหมาย ถ้าเป็น 8 บิต SUMMIN = 00H และ SUMMAX - FFH

7. ขนาดของ SUM

ในการปฏิบัติอาจจำเป็นจะต้องเพิ่มความเที่ยงตรงสำหรับการรวบรวมของ SUM นี่เพราะว่าผลของ KI และ DT ซึ่งจะสมทับเข้ากับ SUM เป็นจำนวนทศนิยม ดังนั้นถ้า SUM เป็น 8 บิต แล้ว KCI = KI ● DT ● SUM จะไม่เข้าใกล้ FFH ซึ่งหมายความว่าเอาต์พุตของตัวควบคุมจะไม่เป็นค่าสูงสุดเลยในโหมดอินทีกรัลที่เราใช้นี้ ถ้าเราเพิ่มความเที่ยงตรงของ SUM เป็นจำนวน 16 บิต ซึ่งมันเป็นไปได้ที่จะมีเอาต์พุตคอนโทรลได้สูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่10 รายละเอียดโฟล์วชาร์ตของโหมดอินทีกรัลแสดงถึงแอคคิวมูเลเตอร์ SUM และการป้องกันโอเวอร์โฟล์วอันเดอร์โฟล์ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. การกระทำแบบตรงและผกผัน (Direct and Reverse Action)

ในรูปแบบที่ได้นำเสนอมานั้นได้สมมุติฐานได้ว่าพื้นที่ซึ่งเพิ่มขึ้นต้องการเอาต์พุตที่เพิ่มขึ้น นี่คือการกระทำแบบตรง ในหลายกรณีพื้นที่ที่เพิ่มขึ้นต้องการเอาต์พุตลดลง นี่เป็นการกระทำแบบผกผัน ซึ่งสามารถเตรียมบล็อกไดอะแกรม สำหรับการรวมและการลบของเออเรอร์กับ SUM ในโฟล์วชาร์ตของรูป 10 ได้แสดงสำหรับแบบตรงซึ่งเออเรอร์เป็นบวก (SGN = 0) เป็นเหตุให้ SUM เพิ่มขึ้นสำหรับการทำแบบผกผันได้เตรียมบล็อกไดอะแกรมไว้ เพื่อว่าเออเรอร์บวกทำให้ DE ไปลบออกจาก SUM

9. ปัญหาการในการเริ่มต้น (Start-up Problem)

หนึ่งในจำนวนปัญหาความยุ่งยากทั้งหมด กับการใช้โหมดอินทีกรัลในระหว่างกระบวนการเริ่มต้น เช่น ระบบการควบคุมอุณหภูมิ ต้องเข้าไปพัวพันกับค่าความร้อน ซึ่งเป็นเงื่อนไขเริ่มต้นระหว่างเวลา โดยที่กำลังเพิ่มความร้อนเพิ่ม เพื่อจะเข้าสู่เซตพอย์น เทอมอินทิกรัลจะคำนวณพื้นที่ด้วยขั้วคงที (เป็นเครื่องหมาย + หรือ - ดังเดิม ผลก็ คือ จะเกิดโอเวอร์ชูต (overshoot)) ขนาดใหญ่ที่เอาต์พุตอย่างไม่เหมาะสม ในกรณีเช่นนี้ต้องงดการทำโหมดอินทีกรัลชั่วคราว จนกระทั่งระบบเข้าสู่อุณหภูมิ หรือให้ค่าออฟเซตของพื้นที่ในขณะเริ่มต้นมีความแตกต่างน้อยที่สุด

3.5.3 โหมดดีริเวทีฟ (Derivative Mode or Rate)

โหมดดีริเวทีฟของระบบความคุมแบบป้อนกลับ ซึ่งจะขึ้นอยู่กับอัตราการเปลี่ยนแปลงของค่าเออเรอร์ การกำหนดรูปแบบของโหมดนี้คือ การกำหนดการคำนวณแบบอนุพันธ์ในการใช้กับระบบคอมพิวเตอร์ค่าที่ใช้งานมีเพียงตัวแปรคาบเวลาแซมปลิงและตัวแปรควบคุม ดังนั้นเพื่อความเหมาะสม จะกำหนดโหมดจากสมการที่กำหนดไว้เดิมต้นต้นคือ

$$C_D(t) = \frac{K_D\left[e_P(t) - e_P(t_0)\right]}{t - t_0}$$

จึงเปลี่ยนบางเทอมให้เหมาะสมแช่นค่าคาบเวลา

ในโหมดดีริเวทีฟจะไม่ใช้ทำงานตามลำพังโดยไม่มีการป้อนกลับ คือต้องมีการเปลี่ยน-แปลงค่าเออเรอร์ตลอดเวลา ถ้าค่าเออเรอร์เป็นค่าคงที่จะไม่สามารถทำงานได้ง่ายถูกต้อง

1. อัลกอลิธึมของดีริเวทีฟโหมด

การประยุกต์ใช้โหมดดีริเวทีฟกับระบบควบคุมโดยคอมพิวเตอร์ จะมีอัลกอลิธึมหลาย-อย่างที่แตกต่างกัน โดยวิธีอนุพันธ์จะสามารถจัดการให้ง่ายสำหรับตัวแปร ซึ่งง่ายที่สุดคือ การใช้สมการข้างต้นโดยตรง ในกรณีทั่วไปโหมดดีริเวทีฟจะกำหนดรูปแบบสมการโดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

037160

$$DCD = \frac{KD \bullet (DFE - DFE0) \bullet ROUT}{DT} \quad (3\text{-}14)$$

ที่ซึ่ง DCD คือ เอาต์พุตคอนโทรลของโหมดดีริเวทีฟ

KD คือ ดีริเวทีฟเกน

DFE คือ เออเรอร์อินพุตในสัดส่วนของช่วง

DFEO คือ เออเรอร์เดิมในสัดส่วนของช่วง

ROUT คือ ช่วงของเอาต์พุต

DT คือ เวลาแซมปลิง

2. การใช้สมการในโหมดดีริเวทีฟให้ง่ายขึ้น (Simplification)

ในสองกรณีดังกล่าวเป็นไปได้ ที่ทำให้สมการง่ายลงสำหรับโหมดการควบคุม ถ้าช่วงอินพุต RIN และช่วงเอาต์พุต ROUT เท่ากัน ดูจากสมการที่ 3-14 เห็นได้ว่าจะเขียนตามแฟชั่นนิยมเป็นดังนี้ อย่างแรก DFE = DE / RIN และ DFEO = DEO / RIN ที่ซึ่ง DEO เป็นค่าเออเรอร์เดิม ถ้าสมการเหล่านี้ลบกันแทนในสมการที่ 3-14 ผลลัพท์จะเป็น

$$DCD = \frac{KD \bullet (DE - DEO) \bullet ROUT}{DT \bullet RIN}$$

ขณะนี้ถ้า RIN = ROUT สมการนี้จะลดรูปลงเหลือ

$$DCD = \frac{KD \bullet DDE}{DT} \quad (3\text{-}15)$$

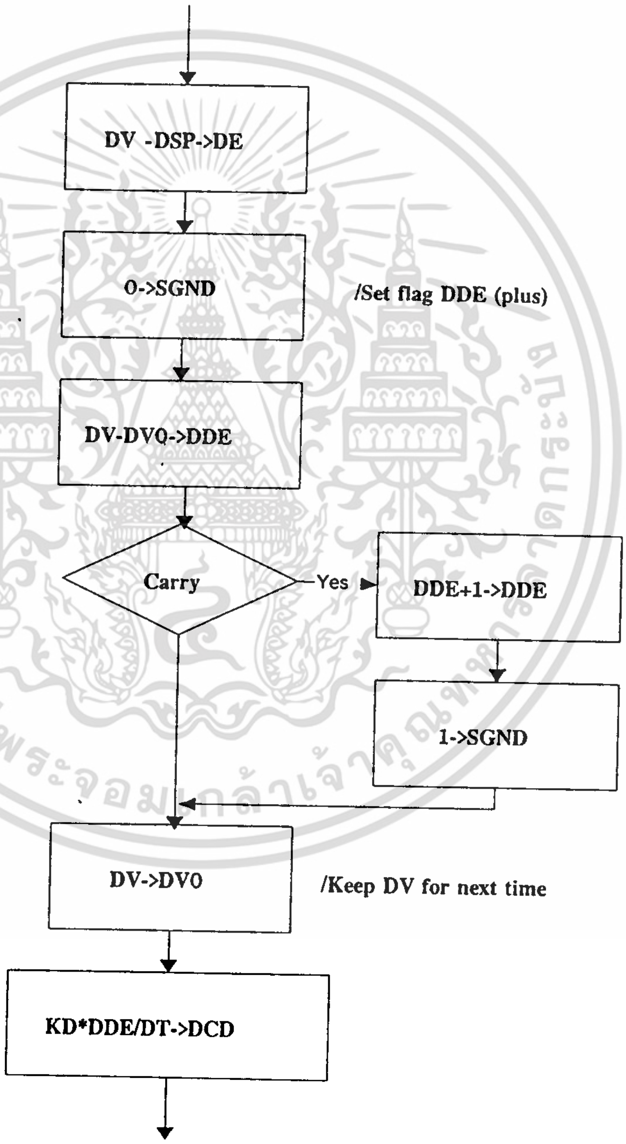
$$DDE = DE - DEO \quad (3\text{-}16)$$

ผลการทำให้สมการกระทัดรัดลงดังกล่าว ถ้าเราเขียนเทอมเออเรอร์แยกจากตัวมันจะได้ DE = DV - DSP และ DEO = DVO - DSP ที่ซึ่ง DVO คือตัวแปรควบคุมของแซมปลิงเดิม ดังนั้นสมการที่ 3-16 จะเป็นดังนี้

$$DDE = DV - DVO \quad (3\text{-}17)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งสองสมการสามารถใช้สนับสนุนการคำนวณ กรณีทั่วไปจะพบว่าง่ายกว่าถ้าใช้สมการที่ 3-17 เพื่อหาเทอมอนุพันธ์ นี่เป็นเพราว่าเมื่อผลต่างระหว่าง DE และ DEO มีความเป็นไปได้ถึงสี่ลักษณะ เครื่องหมายด้วยกันซึ่งจะเป็นผลให้ DDE ติดเครื่องหมายด้วย แต่ DV และ DVO เป็นปริมาณบวก (00H สู่ FFH) โดยเครื่องหมายจะกำหนดโดยผลลบเพียงเท่านั้น โฟล์วชาร์ตรูป 3-11 แสดงการประยุกต์โดยตรงของโหมดดีริเวทีฟ ซึ่งแน่นอนทีเดียวว่าโหมดนี้ไม่สามารถใช้เพียงลำพังโดยไม่มีการป้อนกลับ ซึ่งจะทำให้ค่าเออเรอร์นั้นคงที่ สังเกตได้ว่าไม่จำเป็นต้องเก็บค่า DV เดิมไว้สำหรับใช้ในการคำนวนในขั้นตอนต่อไป



รูปที่11 โฟล์วชาร์ตสำหรับโหมดดีริเวทีฟแสดงถึงการกำหนดเครื่องหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ขั้วเครื่องหมายของอนุพันธ์ (Derivative Polarity)

ผลลัพท์ของการอนุมาณการปฏิบัติโดยวิธีอนุพันธ์นั้น จะเป็นไปได้ทั้งบวกและลบ ซึ่งถ้าเป็นผลลบจะแทนด้วย 2's คอมพลีเมนต์ ในการเก็บมาเพื่อต้องการใช้จำนวนตัวเลขไม่มีเครื่องหมาย จำเป็นอย่างยิ่งต้องเปลี่ยนเป็นจำนวนบวก โดยใช้เครื่องหมายเป็นหลัก นี่แสดงตามรูป 3-11 ดังนั้น ขณะนี้มีแฟลกเครื่องหมายสองตัว แฟลกเครื่องหมายตัวแรกสำหรับ DE และอีกหนึ่งคือเครื่องหมายของผลต่างจากการทำอนุพันธ์ DE - DEO

4. การกำหนดดีริเวทีฟเกน (Derivative Gain)

ค่าดีริเวทีฟ แทนด้วยสัดส่วนการเปลี่ยนเอาต์พุตคอนโทรลต่ออัตราการเปลี่ยนค่าเออเรอร์ดีริเวทีฟเกนอยู่ในหน่วยของเวลา บ่อยครั้งเมื่อเราให้สมการ KD / DT เป็นเหมือนค่าคงที่ตัวหนึ่งคูณกับเทอมของผลต่าง ที่สำคัญหน่วยของ KD และ DT ต้องเข้ากันได้

5. การกระทำแบบตรงและผกผันในโหมดดีริเวทีฟ (Direct and Reverse Action)

เป็นไปได้สำหรับโหมดดีริเวทีฟทั้งกระทำแบบตรงและผกผัน แบบตรงหมายความว่าอัตราบวก (Positive rate) ของการเปลี่ยนแปรค่าควบคุมเป็นผลทำให้เอาต์พุตเพิ่มขึ้น แบบผกผันหมายความว่า สโลปบวก (Positive slop) ในเวลานั้นจะทำให้เอาต์พุตควบคุมลดลง นี่สามารถเตรียมได้โดยกำหนดบล็อกไว้ทั้งบวก และลบ ให้กับเทอมของเอาต์พุตบล็อกของโฟล์วชาร์ต โดยบล็อกนี้ขึ้นอยู่กับสองเครื่องหมายซึ่งเครื่องหมายแรกคือ DDE ถ้าอีกเครื่องหมายหนึ่งเหมือนกันก็จะกระทำเป็นแบบบวกและแบบลบ ถ้าตรงกันข้าม

3.5.4 โหมดคอมโพสิต (Composite Mode)

โดยทั่วไประบบควบคุมแบบต่อเนื่องที่ดีที่สุด เมื่อได้ร่วมการทำงานระหว่างโหมดเดี่ยว (pure mode) ดังกล่าวข้างต้นมารวมกันเป็น 3 แบบด้วยกันคือ

(1) PI เป็นการร่วมกันทำงานระหว่างพรอปพรอชันแนลและอินทีกรัล

(2) PD เป็นการร่วมกันทำงานระหว่างพรอปพรอชันแนลและดีริเวทีป

(3) PID เป็นการรวม 3 โหมด ด้วยกันคือพรอปพรอชันแนล อินทีกรัล และดีริเวทีฟ

มีข้อสงสัยว่า "ทำไมเพิ่งจะใช้คอนโทรลเลอร์แบบคอมโพสิต PID ในการประยุกต์" ในบางการประยุกต์ที่ PI, PD และโหมดเดี่ยว เช่น P,I,D สามารถถูกใช้งานแทน PID ได้ดีกว่าเพราะว่า ประหยัดค่าใช้จ่ายทางเวลา และความพยายามที่เลือกค่าเหมาะสม เพื่อสำหรับจะให้ระบบเสถียรนั้นยากกว่า มีตัวอย่างหลายกระบวนการยังไม่ได้เตรียมการปรับปรุงระบบเป็น PID แต่จะใช้รวมกันระหว่างการทำดีริเวทีฟกับโหมด PI โหมดคอมโพสิตแบ่งได้เป็นโหมดย่อยดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การควบคุมแบบพรอปพรอชันแนล-อินทีกรัล (Proportional-integral Control)

2. การทำ PID คอนโทรลเลอร์ (PID controller action)

1. การควบคุมแบบพรอปพรอชันแนล-อินทีกรัล (Proportional-Integral Control)

ใช้การทำให้พรอปพรอชันแนลและอินทีกรัลรวมเข้าด้วยกันนั้นธรรมดามาก เหตุผลหนึ่งคือให้มีอินทีกรัลคือ ระบบมีความสามารถปรับตัวเองในเงื่อนไขของโหลดใหม่ โดยพยายามให้เออเรอร์เป็นศูนย์

อัลกอลิธึมของ PI

เนื้อแท้ของโหมด PI นั้น คือการรวมระหว่างพรอปพรอชันแนลและอินทีกรัล ซึ่งได้บรรยายไว้ก่อนหน้านี้ อย่างไรก็ตามเราได้พบว่าโครงสร้างที่เหมาะสมกับโหมดคอมโพสิต คือ การนำเทอมพรอปพรอชันแนลคูณกับเออเรอร์และพื้นที่เออเรอร์ของเทอมอินทีกรัล ดังนั้นในขณะใดขณะหนึ่งเอาต์พุตคอนโทรลประกอบขึ้นด้วยทั้งสองเทอม เทอมหนึ่งได้เนื่องจากการคำนวณของพื้นที่ต่อเวลาแซมปลิง และอีกเทอมหนึ่งได้เนื่องจากเออเรอร์ตามปัจจุบัน ซึ่งเป็นไปได้ที่จะเขียนเป็นสมการเดียวในโหมดนี้ดังนี้

$$DCPI = (KP \bullet DE) + (KP \bullet KI \bullet DT \bullet SUM) \tag{3-15}$$

ซึ่ง DCPI คือ เอาต์พุตควบคุมของโหมด PI

KP คือ พรอปพรอชันแนลเกน

DE คือ เออเรอร์

KI คือ อินทีกรัลเกน

DT คือ เวลาแซมปลิง

SUM คือ ผลรวมของเออเรอร์ล่าสุด

สังเกตว่าถ้าเออเรอร์สำหรับเฉพาะบางตัวอย่างเป็นศูนย์ แล้วเอาต์พุตตามสมการ 3-15 จะเป็นผลลัพท์จากเทอมอินทีกรัล เทอมนี้คือผลลัพท์ของเออเรอร์ในอดีตของระบบสู่เทอมของ SUM จะเห็นได้ว่าเทอมอินทีกรัลนั้นจะเตรียมตอบสนองเออเรอร์ศูนย์ (zero error) ของระบบ

แล้วหลักสำคัญคือ โครงสร้างแบบอย่าง ซึ่งถูกใช้โดยสมการที่ 3-15 ในการกำหนดค่าคอนโทรลเอาต์พุต (control output) อย่างไรก็ดีที่จริงการนำไปปฏิบัติของสมการที่ 3-15 กับการปฏิบัติจริงโดยใช้ไมโครโปรเซสเซอร์สร้างนั้นยากมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซตของสมการโหมด PI สมการ 3-15 กำหนดการตอบสนองของ PI แต่ไม่สามารถใส่แทนค่าโดยตรงเข้าสู่ไมโครโปรเซสเซอร์ การใส่รหัสนั้นต้องแตกสมการเป็นหลายขั้นตอน ที่ซึ่งไมโครโปรเซสเซอร์สามารถปฏิบัติได้เฉพาะเลขหนึ่งชุดที่เวลาหนึ่ง ซึ่งมีเซตของสมการดังนี้

$$DE = DV - DSP \tag{3-3}$$

$$SUM = SUM + DE \tag{3-10}$$

$$DCI = KP \bullet KI \bullet DT \bullet SUM \tag{3-16}$$

$$DCP = KP \bullet DE$$

$$DCPI = DCI + DCP \tag{3-17}$$

ในแต่ละสมการสามารถแก้ปัญหาโดยคำสั่งแบบง่าย ๆ ในคอมพิวเตอร์ ค่าคงที่ในสมการที่ 3-16 คือ KP ● KI ● DT ซึ่งแน่นอนจะถูกรวมเข้าเป็นค่าคงที่ตัวหนึ่ง

อย่างหนึ่งเป็นการพยายามรวมสมการ 3-10 และสมการ 3-16 และเขียนเทอมอินทีกรัลเหมือน DCI = DCI + (KP ● KI ● DT ● DE) มีเหตุผลที่ดีอย่างหนึ่งว่า การเลือกสมการนี้อาจไม่ฉลาดนัก เพราะว่า DE เป็นจำนวนค่าน้อย และจะได้ผลเป็นค่าคงที่น้อยกว่า 1 ได้บ่อย ๆ นี่หมายความว่าหลายเออเรอร์จะสร้างความผิดพลาดเข้าสู่ DCI แต่การปฏิบัติการคูณด้วยค่าคงที่นั้น เทอม DE จะหนีจากจำนวนค่าน้อยได้ และสามารถถูกแทนโดยข้อมูลไบต์ได้ โดยที่อย่างแรก DE ต้องบวกกับ SUM ก่อน

โอเวอร์โฟล์วและอันเดอร์โฟล์วของโหมด PI (Overflow and Underflow)

การพิจารณาที่หนักใจที่สุดคือ เมื่อโหมด PI ต้องเตรียมป้องกันโอเวอร์โฟล์ว และอัน-เดอร์โฟลว์ของตัวแปรในการคำนวณ ความคิดพื้นฐานในที่นี้คือ DCPI โดยปกติจะครอบคลุมช่วงจำนวนตัวเลขของไบต์ข้อมูลตั้งแต่ 00h ถึง FFh สำหรับไมโครโปรเซสเซอร์แบบ 8 บิต ค่า DCI ยังคงถูกจำกัดอยู่ในขอบเขตด้วยซึ่งไม่มีเออเรอร์ DCPI เท่ากับ DCI แต่ขีดจำกัดบน SUM จะมีผลต่อปริมาณค่าของ DCI

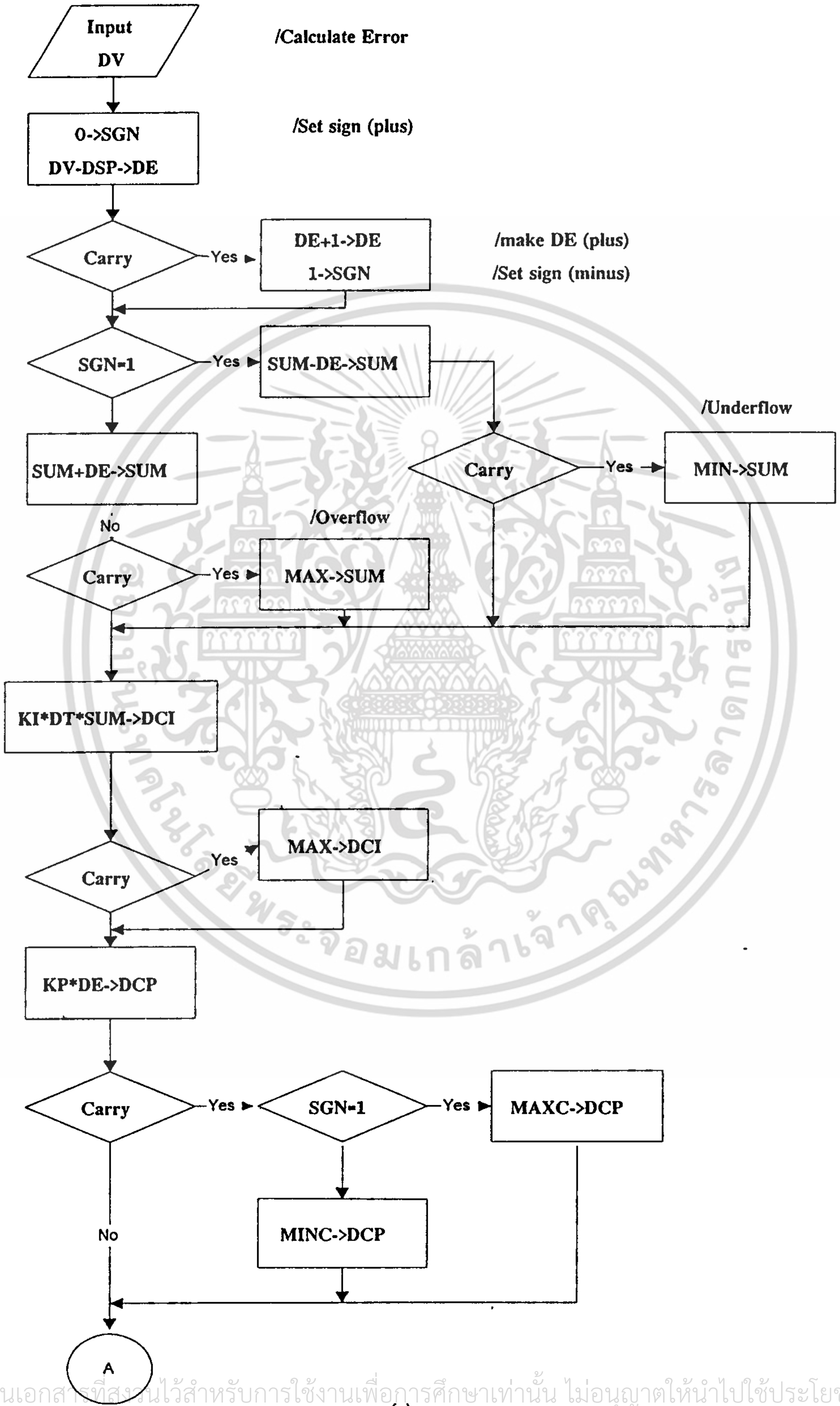
โอเวอร์โฟล์วของปริมาณเหล่านี้เกิดขึ้น เมื่อผลลัพท์จากผลรวมของการรวมในจำนวนที่เกินความเป็นไปได้ของคอมพิวเตอร์เช่น FFH เป็นแบบของไมโครคอมพิวเตอร์ 8 บิต เมื่อเงื่อนไขเช่นนั้นถูกตรวจสอบตัวแปรนั้น ต้องเซตให้เท่ากับ FFH

อันเดอร์โฟล์วเป็นผลลัพท์จากการลบซึ่ง SUM, DCI หรือ DCPI จะพ้นจากการเป็นจำนวนลบ ถ้าการลบทำให้เทอมเหล่านั้นเป็นจำนวนลบเทอมเหล่านั้นจะถูกเซตเป็นศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟล์วชาร์ต (Flowchart)

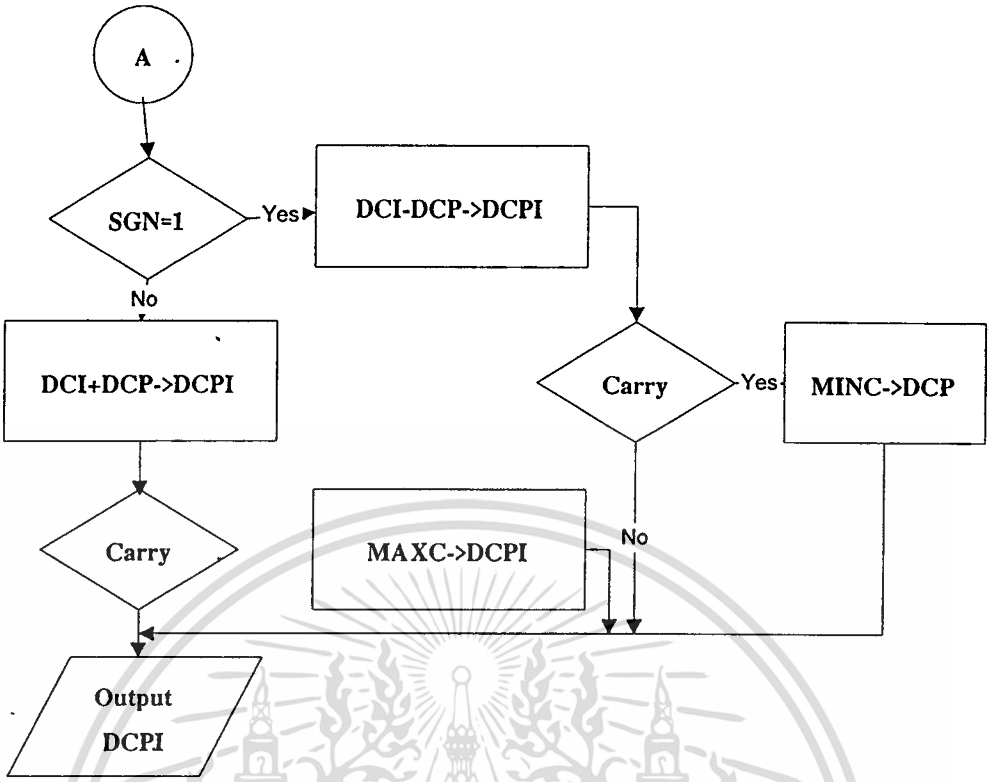
โฟล์วชาร์ตตามรูปที่ 3-12 แสดงส่วนหน้าตาที่จำเป็นของอัลกอลิธึมของโหมด PI ซึ่งจะรวมเอาส่วนที่สำคัญเช่นการป้องกัน โอเวอร์โฟล์ว อันเดอร์โฟล์ว และการพิจารณาเครื่องหมาย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

(a)

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่12 (a),(b) โฟล์วชาร์ตของโหมด PI แสดงถึงการป้องกันโอเวอร์โฟล์วและอันเดอร์โฟล์ว

2. การทำ PID คอนโทรลเลอร์ (PID Controller Action)

คอนโทรลเลอร์แบบ 3 โหมด ในท้ายสุดใช้ในการควบคุมกระบวนการสำหรับฟังก์ชันถ่ายโอนระบบเฉพาะ ซึ่งไม่สามารถเขียนได้ การปรับจูนมีความซับซ้อนเพราะว่ามีหลายเทอมที่พิจารณา แต่การควบคุมสามารถทำให้ประสบความสำเร็จอย่างใหญ่หลวงในหลายงานประยุกต์ ในบทนี้การเพิ่มเติมโหมด PID โดยคอมพิวเตอร์จะถูกแสดงให้ชัดเจน

อัลกอลิธึมของ PID ในตอนที่แล้วเราได้แนะนำการทำดีริเวทีฟในโหมด PI ไว้แล้ว ซึ่งคล้ายกับกรณีนั้น จึงจำเป็นที่จะต้องใช้พรอปพรอชันแนลเกนเข้าไปคูณกับดีริเวทีฟเทอม ผลลัพท์ของสมการควบคุมจะถูกแบ่งเป็นสามเทอม เทอมแรกเป็นสัดส่วนโดยตรงกับเออเรอร์ เทอมสองนี้ขึ้นอยู่กับเออเรอร์ในอดีต และเทอมสุดท้ายนี้ขึ้นอยู่กับอัตราการเปลี่ยนแปลงเออเรอร์ในปัจจุบัน สมการแสดงได้ดังนี้

$$DCPID = (KP \bullet DE) + (KP \bullet KI \bullet DT \bullet SUM) + \frac{KP \bullet KD \bullet DDE}{DT} \quad (3\text{-}18)$$

ที่ซึ่ง DPCID

KP คือ พรอปพรอชันแนลเกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DE คือ เออเรอร์แปรเปลี่ยน = DV - DSP

KI คือ อินทีกรัลเกน

DT คือ เวลาแซมปลิง

SUM คือ ผลบวกเออเรอร์

KD คือ ดีริเวทีฟเกน

DDE = DV - DVO คือ เออเวอร์เปลี่ยนในแต่ละแซมปลิง

สังเกตว่าถ้าเออเรอร์คือ ศูนย์ เอาต์พุตจะถูกกำหนดด้วยผลจาก SUM และการเปลี่ยนของเออเรอร์ในแซมปลิงก่อนหน้านี้

การทำโปรแกรมจริงตามสมการที่ 3-18 นั้นต้องการแตกสมการออกเป็นส่วนไม่ขึ้นต่อกันและดูตามแต่ละส่วนเพื่อนำไปประยุกต์

การประยุกต์สมการ การปฏิบัติในการประยุกต์ของการควบคุมแบบ PID ซึ่งต้องการสมการควบคุม 3-18 ถูกแตกเป็นหลายส่วนซึ่งไมโครโปรเซสเซอร์กระทำได้หน่วยส่วนต่อหนึ่งเวลา เซตแบบอย่างควบคุมประกอบด้วยเซตดังนี้

$$
\begin{aligned}
DE &= DV - DSP \\
SUM &= SUM + DE \\
DDE &= DV - DVO \\
DVO &= DV \\
DVI &= KP \bullet KI \bullet DT \bullet SUM \\
DCD &= \frac{KP \bullet KD \bullet DDE}{DT} \\
DCP &= KP \bullet DE \\
DCPID &= DCP + DCI + DCD
\end{aligned}
\qquad (3\text{-}19)
$$

ค่าคงที่ต่างๆ ถูกรวมกับเพื่อเพิ่มเติมให้เป็นจริงในสมการ เพื่อเป็นประโยชน์การพิจารณาการปฏิบัติจำเป็นอย่างยิ่ง ในการแทนค่าในสมการสู่การประยุกต์จริง ส่วนใหญ่จะมีความคิดเห็นเหมือนก่อนหน้านี้สำหรับคอนโทรลเลอร์แบบดีริเวทีฟ และ PI เหล่านี้จะถูกระบุในการรวมกับโหมดดีริเวทีฟต่อไป

เครื่องหมายของ DDE (ploarity of DDE) เทอมเครื่องหมายหรือขั้วของ DDE และดังนั้นบล็อกของการบวกหรือการลบ DCD จาก DCI + DCP ต้องอยู่บนเงื่อนไขของเครื่องหมายหรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้ว สังเกตว่าถ้ากระทำแบบตรง KP > 0 และเป็นอัตราบวก DDE > 0 มีความหมายว่า DCD จะเพิ่ม -ขึ้น ขณะที่การทำผกผันสำหรับ DDE > 0 แล้ว DCD จะถูกลบ

ปัญหาโอเวอร์โฟล์ว ถ้าค่าคงที่คูณกับ DDE มากกว่า 1 แล้วมันจะเป็นไปได้สำหรับ DCD โอเวอร์โฟล์ว ด้วยเหตุผลนี้การตรวจสอบโอเวอร์โฟล์วจะต้องเตรียมไว้

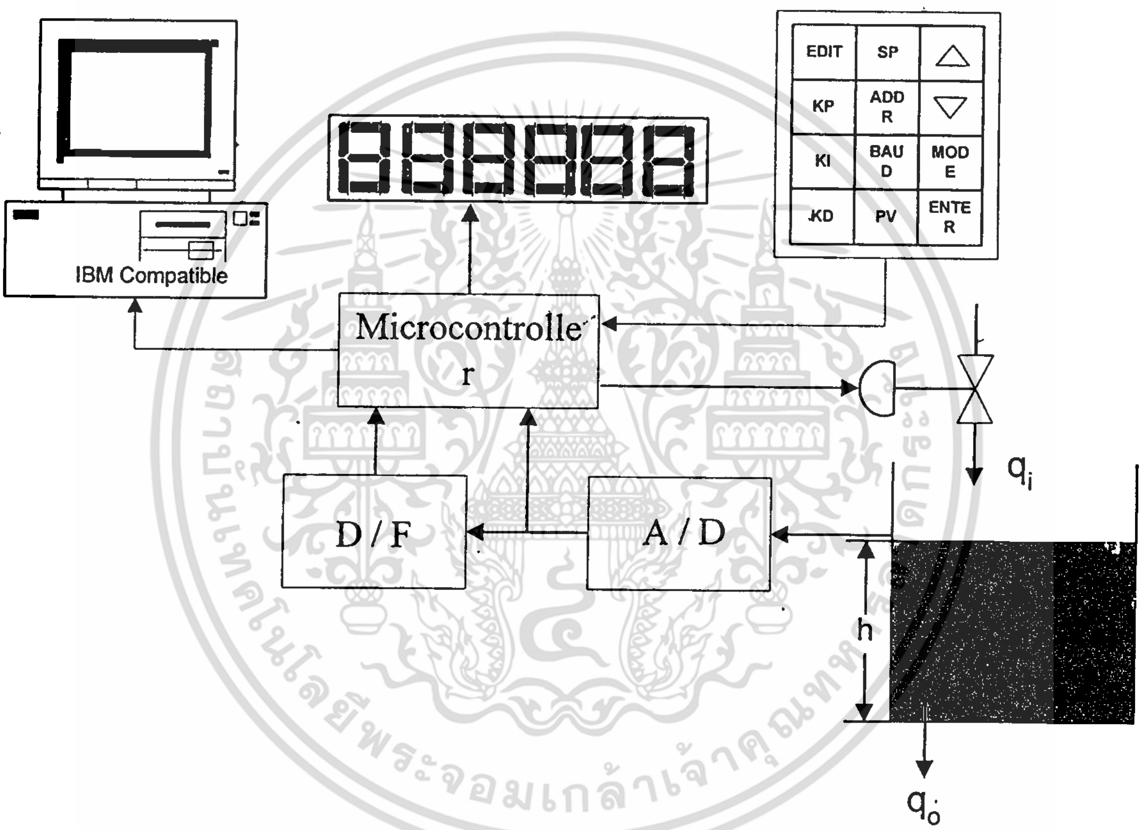
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การออกแบบวงจรตัวดิจิทัลคอนโทรลเลอร์

จุดมุ่งหมายของบทนี้เกี่ยวข้องกับฮาร์ดแวร์ เกี่ยวกับข้อมูลอินพุตสู่ไมโครโปรเซสเซอร์ข้อมูลเอ้าต์พุตจากไมโครโปรเซสเซอร์ ซึ่งการแปรผันข้อมูลจะเกี่ยวกับการแปรผันทางด้านควบคุมในแบบต่างกัน

4.1 โครงสร้างของตัวดิจิทัลคอนโทรลเลอร์



รูปที่ 4-1 *แสดงบล็อกไดอะแกรมของตัวดิจิทับคอนโทรลเลอร์*

ตามรูปที่ 4-1 เป็นบล็อกไดอะแกรมของตัวคอนโทรลเลอร์รวมทั้งหมด โดยเลือกใช้ตัวไมโครโปรเซสเซอร์ 8 บิตเบอร์ 8031 ซึ่งเรียกโดยทั่วไปว่า “ไมโครคอนโทรลเลอร์” ใช้เป็นตัวประมวลผลหรือ ซีพียู ซึ่งไมโครคอนโทรลเลอร์เบอร์นี้มีจุดมุ่งหมายสร้างขึ้น เพื่อให้ใช้ได้งานประยุกต์ต่าง ๆ ในทางดิจิทับโดยไม่ซับซ้อนมากนัก สิ่งสำคัญคือราคาอุปกรณ์ราคาไม่แพง โดยพยายามเลือกอุปกรณ์แต่ละอย่างเพื่อใช้ร่วมกันให้ดีที่สุดและให้ประหยัดในการประยุกต์ใช้เป็นตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอนโทรลเลอร์นี้จำเป็นจะต้องมีอุปกรณ์อินเตอร์เฟสร่วมด้วย ตามรูปมีอุปกรณ์อินเตอร์เฟสต่าง ๆ ที่สำคัญดังนี้

- วงจรคีย์บอร์ด(KBC)
- วงจรแสดงผล(display)
- วงจรซีเรียลอินเตอร์เฟส(UART)
- วงจรแปลงแอนะล็อกเป็นดิจิทัล(ADC)
- วงจรแปลงดิจิทัลเป็นแอนะล็อก(DAC)
- วงจรเฟส/ความถี่ดีแทกเตอร์(PFD)

4.1.1 ลักษณะทางสถาปัตยกรรมของ 8031 โดยย่อ

กรณีแรกในปัญหาคือต้องเรียนรู้เกี่ยวกับการใช้งานกับไมโครคอนโทรลเลอร์เบอร์ ซึ่งจะต้องเรียนรู้ลักษณะกลไกทางฮาร์ดแวร์ต่าง ๆ การกำหนดจำนวนเรจิสเตอร์(register) และการใช้งานเรจิสเตอร์ ความสามารถของคำสั่งทางซอฟต์แวร์ จนกระทั่งถึงข้อดีต่าง ๆ ในการเลือกใช้

1) ฮาร์ดแวร์ของ 8051

โดยแท้จริง 8051 เป็นไมโครคอนโทรลเลอร์รวมอยู่ในตระกูลเดียวกัน มีตั้งแต่เบอร์ 8031 ถึง 8751 มีหลายแพคเกจ(package)หาเลือกใช้ได้เช่น แบบ N-Channel Metal Oxide Silicon (NMOS) ไอซีตระกูล 51 เป็นแบบ DIP 40 ขา การจัดขาต่าง ๆ จะเป็นลักษณะเฉพาะตัว ฉะนั้นต้องดูคู่มือประกอบไมโครคอนโทรลเลอร์ 8051 ภายในประกอบด้วย เช่น

- รอมและแรมภายใน
- พอร์ตอินพุตเอาต์พุตและขาโปรแกรม
- ไทเมอร์และตัวนับ
- สื่อสารข้อมูลแบบอนุกรม

และในรูปยังมีส่วนประกอบเกี่ยวข้องกับ ซีพียู คือ โปรแกรมตัวนับ เรจิสเตอร์ใช้งาน และวงจรคล็อก

โครงสร้างทางสถาปัตยกรรมของ 8051 ประกอบด้วยดังนี้ :

- 8 บิต ซีพียู กับเรจิสเตอร์ A (accumlator) และ B
- 16 บิตโปรแกรมตัวนับ(PC) และดาต้าพอย์นเตอร์(DPTR)
- 8 บิตเวอร์ดเก็บสถานะโปรแกรม (Program Status Word) PSW

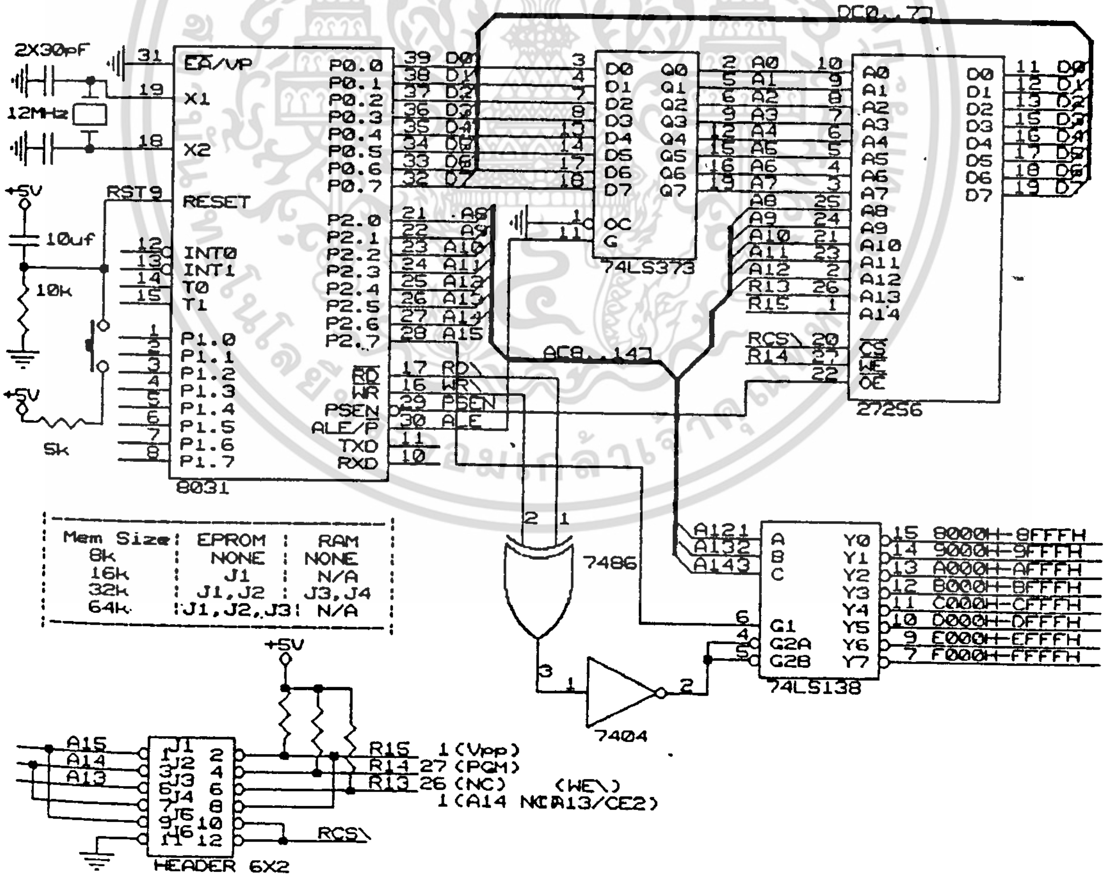
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 8 บิตสแต็กพอยน์เตอร์(SP)
- รอมหรืออีพรอมภายใน และไม่มีใน 8031 และ 4 กิโลไบต์ ใน 8051
- แรมภายใน 128 ไบต์

เรจิสเตอร์ 4 แบงค์ซึ่งประกอบด้วย 8 เรจิสเตอร์

- 16 ไบต์ซึ่งเป็นแอดเดรสระดับบิต
- 8 ไบต์เป็นดาต้าเมโมรี (data memory)ใช้ทั่วไป
- 32 ขาอินพุตเอาต์พุตจัดโดย 8 บิตพอร์ต P0-P3
- ฟูลดูเพล็กซ์ (full duplex)ในการรับส่งสัญญาณอนุกรม
- เรจิสเตอร์ควบคุม TCON,TMOD,SCON,PCON,IP และ IE
- มีอินเตอร์รัพต์จากภายนอกในสองแหล่ง

โมเดลการโปรแกรมของ 8051 มีเรจิสเตอร์ให้เลือกทั้ง 8 และ 16 บิต อีกทั้ง 8 บิตในหน่วยความจำ ซึ่งเรจิสเตอร์และตำแหน่งในหน่วยความจำเหล่านั้นทำงานโดย ใช้คำสั่งทางซอฟต์แวร์ซึ่งร่วมเข้าด้วยกันกับส่วนที่ออกแบบไว้



รูปที่ 4-2 *วงจร 8031 บอร์ด*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

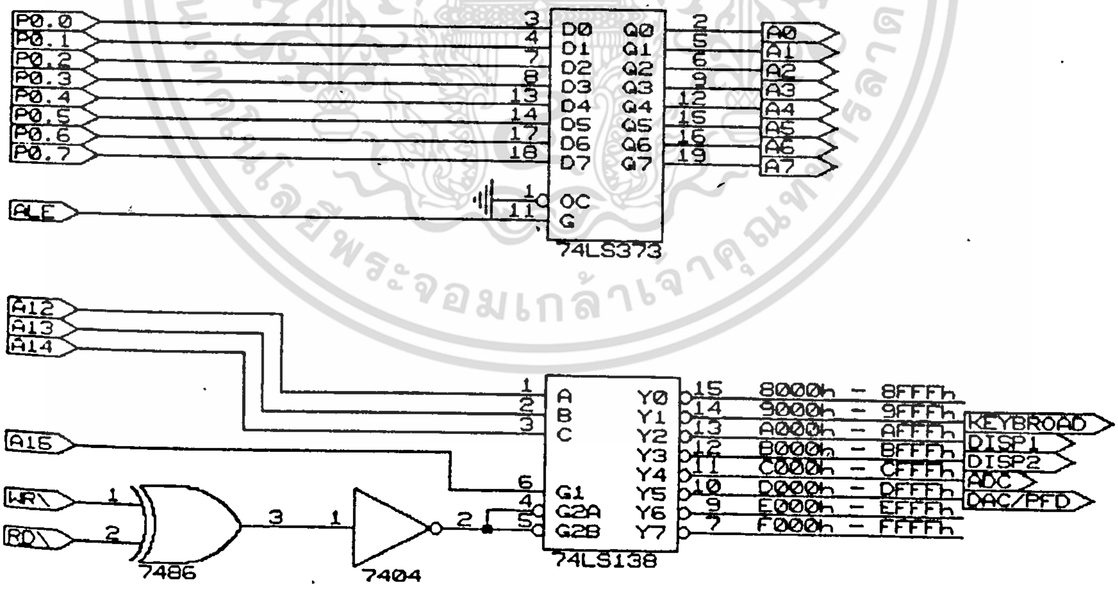
4.1.2 การออกแบบ 8031 ไมโครคอนโทรลเลอร์บอร์ด

ปกติแล้ว IC 8031 (ไม่มีรอมภายใน) ไม่สามารถทำงานตามลำพังได้ ซึ่งตัวเองต้องการวงจรของหน่วยความจำภายนอก และวงจร I/O และส่วนอินเตอร์เฟสต่าง ๆ ที่นำมาใช้งานโดยเฉพาะ ในการออกแบบบอร์ด 8031 จำเป็นต้องพร้อมที่จะร่วมวงจรเหล่านั้นไปด้วย ซึ่งการออกแบบแบบนั้นต้องทำตามแนวทางวิธี ซึ่งสามารถประหยัด เกิดประโยชน์และยืดหยุ่นในการใช้ได้

1) การถอดรหัสแอดเดรสหน่วยความจำ

หน่วยความจำภายนอกถูกใช้ร่วมกับพอร์ตเบอร์ศูนย์ เป็นดาต้าบัสและแอดเดรสบัสไบต์ต่ำ และพอร์ตเบอร์สองเป็นแอดเดรสบัสไบต์สูง ดาต้าและแอดเดรสไบต์ต่ำถูกทำมัลติเพลกซ์บนพอร์ตเบอร์ศูนย์ ตัวแลทซ์แอดเดรสภายนอกเบอร์ 74LS373 ถูกต่อเข้ากับพอร์ตเบอร์ศูนย์ เก็บค่าแอดเดรสไบต์ต่ำ เมื่อไรก็ตามจะเข้าถึงตำแหน่งของหน่วยความจำภายนอก โดยที่ 8031 จะส่งพัลส์ ALE ไปทริกเพื่อให้ 74LS373 แลทซ์แอดเดรส และพอร์ตเบอร์ศูนย์ ยังเป็นบัสแบบสองทิศทางในระหว่างช่วงคาบไซเคิลอ่านและเขียน

แรม และรอม กำหนดการเข้าแอดเดรสในเงื่อนไขที่ต่างกัน สัญญาณ PSEN สำหรับ รอม และสัญญาณWR หรือสัญญาณ RD สำหรับแรม ซึ่งแต่ละส่วนมีขนาดการเข้าถึง 64 กิโลไบต์ รูปที่ 4-2 คือบอร์ดวงจร ไมโครโปรเซสเซอร์ 8031 และการถอดรหัสแอดเดรสหน่วยความจำ



รูปที่ 4-3 *วงจรถอดรหัสแอดเดรสอุปกรณ์ภายนอก*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การถอดรหัสแอดเดรสอุปกรณ์ภายนอก

วงจรของบอร์ดควบคุมต้องติดต่อกับอุปกรณ์ภายนอกที่จำเป็นในการควบคุม ซึ่งแบ่งเป็นอุปกรณ์ คีย์บอร์ด ADC DAC เฟสดีเทกเตอร์ และอุปกรณ์แสดงผลเป็นต้น รูปที่ 4-3 แสดงถึงวงจรการถอดรหัสแอดเดรสอุปกรณ์ภายนอกตามแอดเดรสค่าต่าง ๆ

จากวงจรสังเกตได้ว่าการจัดสรรค์แอดเดรสเพื่อติดต่อกับอุปกรณ์ภายนอกนั้น จำเป็นต้องการถอดรหัสแอดเดรสที่มีลักษณะซ้ำกับการถอดรหัสแอดเดรสแรมภายนอก โดยใช้สัญญาณ RD กับ WR เป็นสัญญาณอีนาเบิลให้กับ 74LS138 ซึ่งเป็นไอซีตัวถอดรหัส 8 ช่อง ในการถอดรหัสแอดเดรส RAM ภายนอกนั้นสามารถจัดสรรค์เนื้อที่ได้ 64 กิโลไบต์ดังกล่าว แต่อย่างไรก็ดีการถอดรหัสแอดเดรสอุปกรณ์ภายนอกนั้นจะเนื้อที่ซ้ำเฉพาะ 32 กิโลไบต์หลังเท่านั้น ทำให้เนื้อที่ของแรม ภายนอกเหลือ 32 กิโลไบต์ ซึ่งเพียงพอกับการใช้งานในหลาย ๆ การประยุกต์ แอดเดรสการถอดรหัสอุปกรณ์ภายนอกแสดงตามตาราง 4-1 ดังนี้

แอดเดรส	อุปกรณ์ติดต่อ
8000h 8FFFh	สำรอง
9000h 9FFFh	วงจรแกนคีย์บอร์ดแบบเมตริก 12 คีย์ (KBC)
A000h AFFFh	วงจรแสดงผลส่วนที่ 1 (DISP1)
B000h BFFFh	วงจรแสดงผลส่วนที่ 1 (DISP1)
C000h CFFFh	วงจรแปลงอะนาลอกเป็นดิจิตอล (ADC)
D000h DFFFh	วงจรแปลงดิจิตอลเป็นแอนะล็อก (DAC) และวงจรเฟสดีเทกเตอร์
E000h EFFFh	สำรอง
F000h FFFFh	สำรอง

ตารางที่ 4-1 *จำแนกการถอดรหัสแอดเดรสอุปกรณ์ภายนอก*

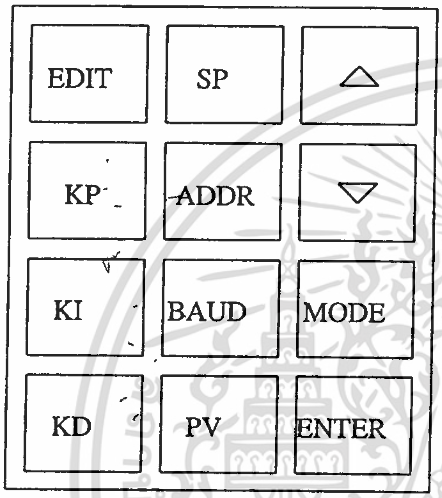
สังเกตได้เช่น ช่วงแอดเดรสระหว่าง 9000h ถึง 9FFFh นั้นสำหรับวงจรคีย์บอร์ดซึ่งมีเนื้อที่ระหว่างแอดเดรสถึง 4 กิโลไบต์ อันที่จริงตามอุดมคติสามารถใช้เพียงไม่กี่ไบต์เท่านั้น สำหรับการถอดรหัสแอดเดรสพอร์ต แต่เพื่อความจำเป็นและความประหยัดในอุปกรณ์ถอดรหัสแอดเดรสจึงเห็นว่าการใช้ลักษณะนี้ เพียงพอสำหรับงานนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 วงจรต่าง ๆ ในตัวดิจิทัลคอนโทรลเลอร์

4.2.1 วงจรสแกนคีย์บอร์ด

อุปกรณ์ที่ติดต่อกับมนุษย์กับคอมพิวเตอร์ คือ คีย์บอร์ด ซึ่งจะประกอบด้วยแป้นคีย์บอร์ด ประกอบด้วยคีย์ต่างๆ เช่นคีย์ตัวเลข ตัวอักษร เป็นต้น ซึ่งในการประยุกต์ใช้กับตัวคอนโทรลเลอร์ จะประกอบด้วยคีย์ซึ่งทำหน้าที่ต่าง ๆ ดังนี้



แป้นคีย์บอร์ด

EDIT : แก้ไขค่าต่างที่กำหนด

SP : แสดงค่าเซตพอยน์บนจอแสดงผล

KP : พรอปพรอซันนอลเกน

KI : อินทีกรัลแกน

KD : ดิริเวทีฟเกน

PV : แสดงค่าตัวแปร(process variable)

ADDR : แสดงหมายเลขแอดเดรสสถานี

BAUD : แสดงอัตรบอด

MODE : แสดงโหมดการทำงาน พร้อมแก้ไข

ENTER : รับข้อมูล หรือเตรียมพร้อมแก้ไข

△ : เพิ่มค่า

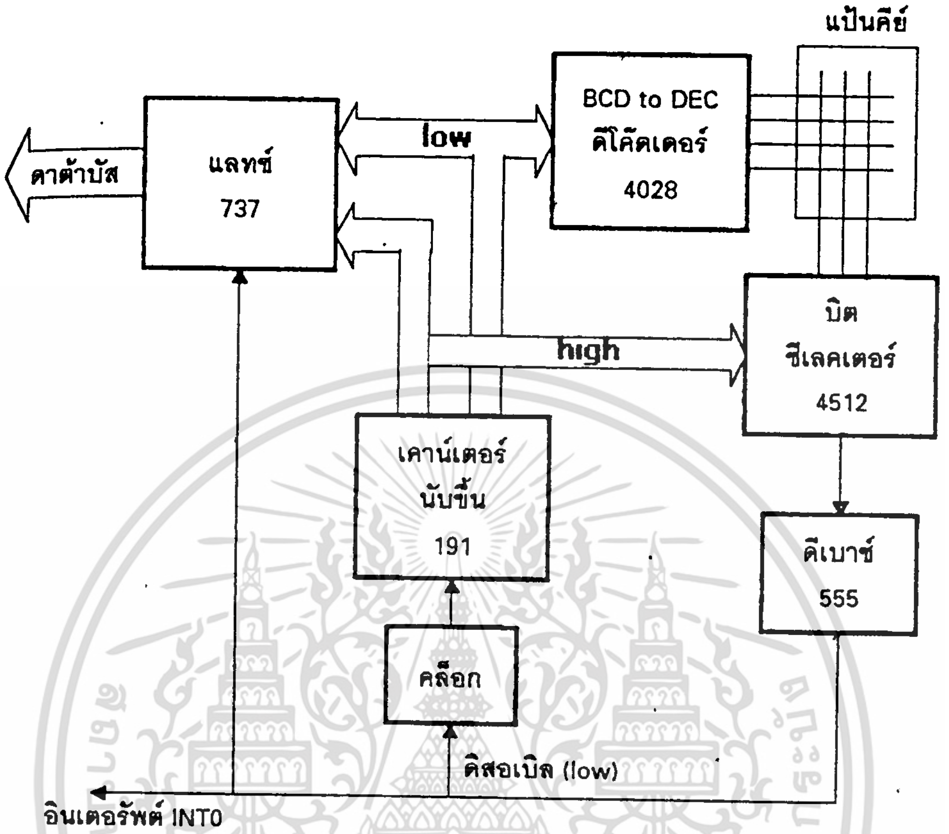
▽ : ลดค่า

หมายเหตุ รายละเอียดเกี่ยวกับการใช้งานดูวิธีการใช้งานในหัวข้อต่อไป

วงจรและโปรแกรมของคีย์บอร์ดต้องระแวดระวังเงื่อนไขที่เป็นไปได้เช่น การกดคีย์มากกว่าหนึ่งครั้งกดคีย์ค้างไว้ กดและปล่อยอย่างรวดเร็วเป็นต้น ซึ่งคุณสมบัติโดยทั่วไปของคีย์บอร์ดที่ถือปฏิบัติคือความสามารถในการป้องกันเปิดปิดของคีย์สวิทท์ซ้ำ หรือเรียกอีกอย่างหนึ่งว่า "ดีเบาซ์ (debounce)" การดีเบาซ์มีหลายแบบ เช่นใช้การหน่วง หรือใช้ R-S ฟลิปฟลอป การจัดองค์ประกอบของคีย์แบ่งเป็นแต่ละประเภทดังนี้ การจัดคีย์แบบ 1 คีย์ต่อ 1 บิตพอร์ต แบบนี้จะมีขาด้านหนึ่งของคีย์จะร่วมกันตลอด และอีกขาหนึ่งจะต่อกับพอร์ต ซึ่งการต่อแบบนี้จะมีการตรวจสอบคีย์เร็ว แต่ถ้ามีจำนวนคีย์มาก ๆ จะสิ้นเปลืองจำนวนขาพอร์ตมาก การจัดคีย์อีกแบบหนึ่งคือการจัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบ XY ซึ่งจำนวนคีย์ทั้งหมดเท่ากับจำนวนแถว X และคูณคอลัมน์ Y โดยที่จำนวนขาพอร์ตเท่ากับ X บวก Y การจัดแบบนี้จะใช้นิยมใช้กันมาก ถ้ามีจำนวนคีย์มากกว่า 8 คีย์ขึ้นไป



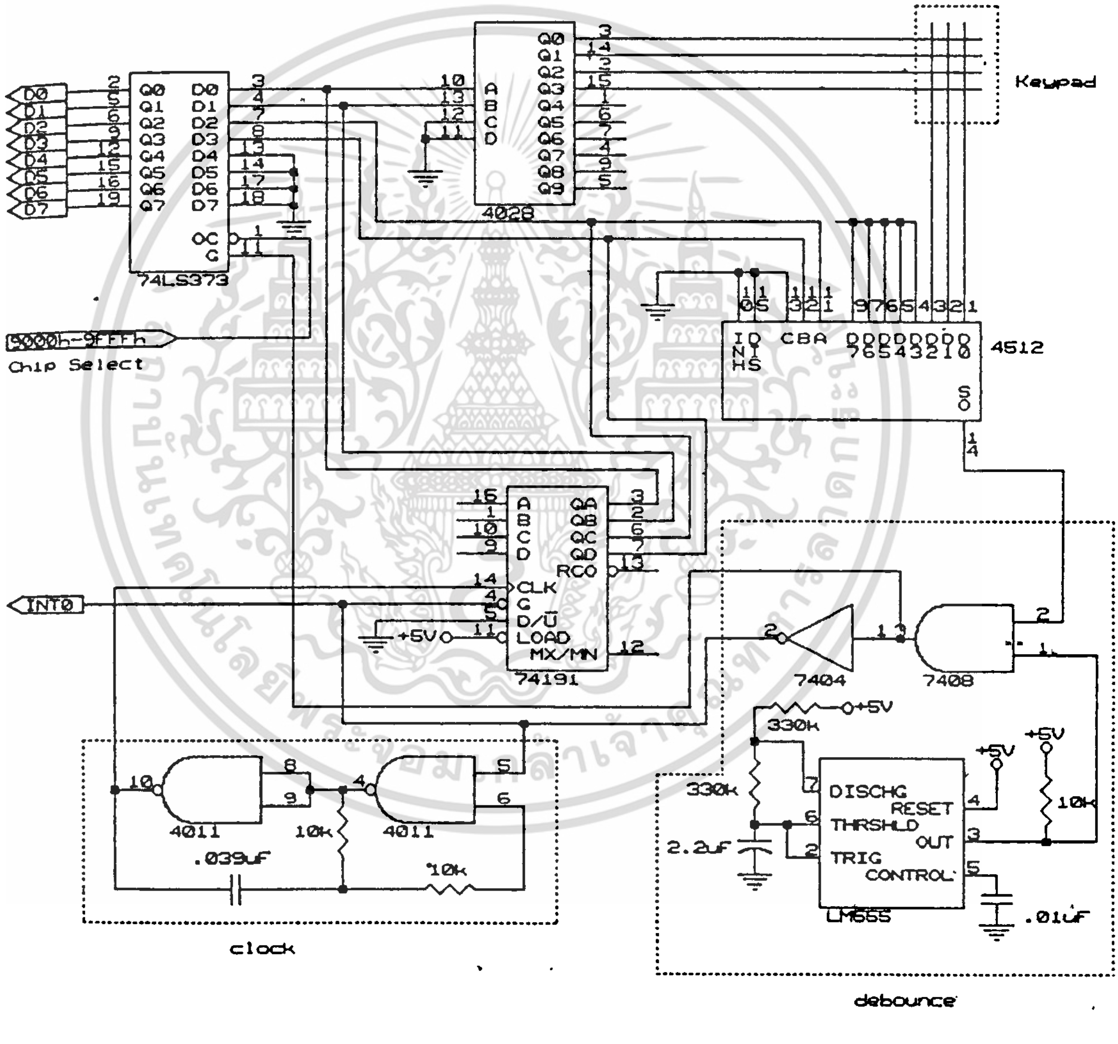
รูปที่ 4-4 *บล็อกไดอะแกรมสแกนคีย์บอร์ด*

สำหรับการสแกนคีย์สามารถใช้เป็นซอฟต์แวร์หรือเป็นวงจร ซึ่งในงานวิจัยนี้จะใช้เป็นวงจรในการสแกนคีย์ให้กับตัวคอนโทรลเลอร์ เหตุนี้เพราะว่าไม่ต้องการแบ่งเวลาของ ซีพียู มาใช้งานสแกนคีย์นั่นเอง วงจรสแกนคีย์บอร์ดนั้นมีบล็อกไดอะแกรมดังรูปที่ 4-4

ในขณะที่คีย์ยังไม่ถูกกด วงจรคล็อกจะไม่ถูกดิสเบิล(disable) เพราะฉะนั้นวงจรคล็อกจะส่งสัญญาณคล็อกตามปกติให้กับตัวนับแบบนับขึ้น(up counter) ตัวนับจะทำการนับคล็อกเหล่านั้น ซึ่งเอาต์พุตที่ได้เป็นค่าไบนารี 4 บิตตั้งแต่ 00 ถึง 0Fh โดยที่เอาต์พุตบิตนัยสำคัญต่ำ(low-significance bit) ของตัวนับป้อนให้กับตัวถอดรหัส
BCD-to-DECIMAL และเอาต์พุตบิตนับสำคัญสูง(high-significance bit) สู่ตัวเลือกบิต(bit selector) ในการสแกนคีย์นั้นตัวถอดรหัสเมื่อรับอินพุตจากตัวนับแล้ว เอาต์พุตแต่ละขาจะสลับกันเป็นลอจิกสูงเรียงกันไปตลอดทั้ง 4 ขา เช่นเดียวกับตัวเลือกบิตจะคอยกวาดรับค่าอินพุตทั้ง 3 ขาจากคีย์บอร์ด ซึ่งโดยปกติจะมีอินพุตรับคีย์เป็นลอจิกต่ำ ถ้าคีย์ไม่ถูกกดแล้วเอาต์พุตตัวเลือกจะเป็นลอจิกต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมมุติถ้ามีการกดคีย์จะมีลอจิกสูงเข้าทางอินพุตรับคีย์ของตัวเลือกบิต จะทำให้เอาต์พุตเป็นลอจิกสูง ซึ่งจะถูกดีเบาซ์จากวงจรดีเบาซ์ หลังจากนั้นวงจรสร้างสัญญาณนาฬิกาจะถูกดิสเอเบิลคือหยุดทำงาน ในขณะเดียวกันตัวแลทช์ทำการเก็บไบต์ข้อมูลจากตัวนับ ซึ่งหยุดนับเมื่อไม่มีคล็อกพร้อมกับสัญญาณที่ส่งไปอินเตอร์รัพต์บนบอร์ดไมโครคอนโทรลเลอร์ 8031 เพื่อเป็นการบอกไมโครโปรเซสเซอร์ว่ามีการกดคีย์ และข้อมูลถูกเก็บไว้ที่ตัวแลทช์ตามแอดเดรสที่กำหนด ซึ่งเมื่อไมโครโปรเซสเซอร์ถูกอินเตอร์รัพท์ จะทำการ อ่าน ข้อมูลจากแอดเดรสของคีย์บอร์ดคือ 9000h-9FFFh แต่ละคีย์จะกำหนดเป็นข้อมูลไบต์ ตามตารางที่ 4-2



รูปที่ 4-5 *วงจรสแกนคีย์บอร์ด*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรสแกนคีย์อย่างสมบูรณ์ แสดงดังรูป 4-5 สังเกตว่าเส้นประจะคล้องจองกับบล็อกไดอะแกรม ซึ่งได้แสดงดังรูป 4-4 เช่น บล็อกดีเบาซ์(debounce) ประกอบด้วย ไอซี 555 แอนด์เกทและน็อตเกต เป็นต้น

ปุ่มคีย์บอร์ดถูกกด	ไบต์ข้อมูลถูกส่ง
เพิ่มค่า	00h
ลดค่า	01h
MODE	02h
ENTER	03h
SP	04h
ADDR	05h
BAUD	06h
PV	07h
EDIT	08h
KP	09h
KI	0Ah
KD	0Bh

ตารางที่ 4-2 *แสดงไบต์ข้อมูลที่ถูกส่งจากวงจรสแกนคีย์เมื่อกดปุ่ม*

4.2.2 วงจรแสดงผล (Display Circuit)

ถ้าคีย์บอร์ดเป็นอินพุตอินเตอร์เฟสกับมนุษย์แล้ว การแสดงผล คือเอาต์พุตอินเตอร์เฟสกับมนุษย์ บอร์ดแสดงผล แบ่งเป็น 3 อย่างดังนี้

1. แสดงเป็นจุดแสง(single light)
2. แสดงเป็นตัวอักษร(lingle charactor)
3. แสดงเป็นอักขระแบบฉลาด(intelligent alphanumeric)

จุดแสงคือมีความหมายเดียวกับการใช้ LED โดยใช้แสดงผลเป็นบิต หรือจะแสดงค่าเปิด-ปิด เป็นต้น การแสดงเป็นตัวเลขและตัวอักขระในขอบเขตจำกัด ซึ่งอาจเป็นอย่างง่ายคือ ตัวเลข 7 ส่วน(7-segment) หรือที่ดีกว่าคือ ดอตแมตริกซ์(dot-matrix) ซึ่งการแสดงผลจะรับโค๊ดแอสกี้

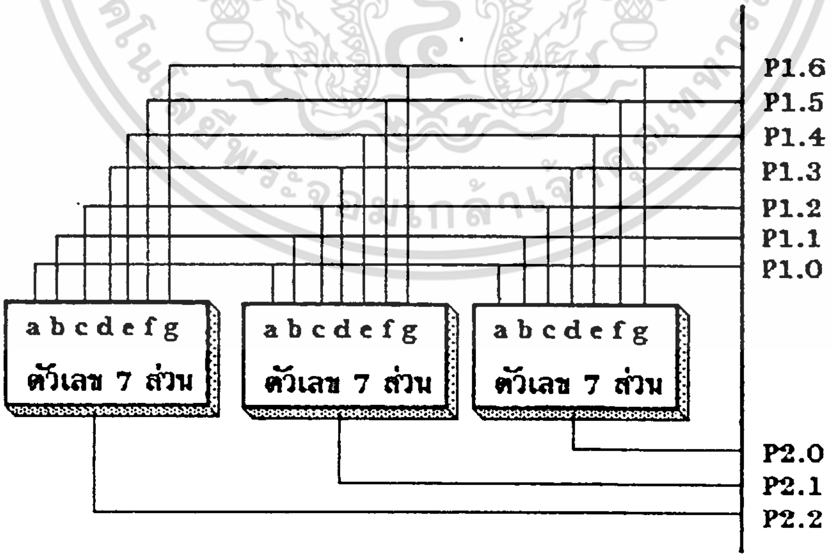
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ASCII) 8 บิต โดยจะคล้องจองกับแบบแผน(patterm)การแสดงผลบน ดอตเมตริกซ์ ส่วนการแสดงเป็นอักขระแบบฉลาดนั้นจะถูกจัดเตรียมไมโครคอนโทรลเลอร์อยู่ภายใน อย่างเช่นการ์ดแสดงผล วีจีเอ(VGA) ดังที่ใช้กับคอมพิวเตอร์พีซี เป็นต้น ซึ่งในตัวคอนโทรลเลอร์ที่ใช้อยู่ในงานวิจัยจะใช้แสดงเป็นอักขระหรือตัวเลข 7 ส่วน (7-segment)

ตัวเลข 7 ส่วน(7-segment) เป็นเซกเมนต์ของ LED ประกอบกับคล้ายเลข "8" กับขาร่วม เช่นแคโทดร่วมหรือแอโนดร่วม โดยปกติจะใช้เป็นแคโทดร่วม ถ้ามากกว่าใช้ 1 หลัก แล้วต้องทำมัลติเพลกซ์เวลาเพื่อไม่ให้ตาของมนุษย์จับการกระพริบได้ ซึ่งการแสดงผลต้องต่อเนื่องไม่มากกว่า 10 มิลลิวินาที(3) ถ้าต้องการจะรู้ว่าตัวเลข 7 ส่วนแต่ละตัว ใช้ช่วงเวลาเปลี่ยนข้อมูลใหม่เท่าไร แล้วต้องนำจำนวนตัวเลข 7 ส่วนทั้งหมดเป็นตัวหารของ 10 มิลลิวินาที

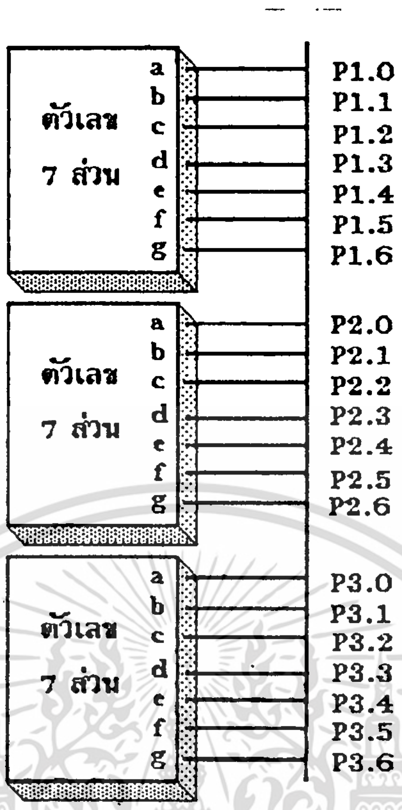
สำหรับวงจรขับตัวเลข 7 ส่วนแบบนิยมใช้ทั่วไปนั้น จะใช้วิธีให้แต่ละแซกเมนต์ (a,b,c,d,e,f,g) ของแต่ละตัวของตัวเลข 7 ส่วน ต่อเข้ากับ 1 บิตพอร์ต รวมทั้งหมดเป็น 1 ไบต์พอร์ต และขาร่วม (common) ของแต่ละตัวต่อเข้ากับ 1 บิตพอร์ต แสดงรูปที่ 4-6

ในเรื่องของโปรแกรมสำหรับตัวเลข 7 ส่วน (7-segment) อินเตอร์รัพต์ไทเมอร์ TO จะคอยกำหนดช่วงเวลาโหลดตัวอักษรไปให้ตัวเลข 7 ส่วนแต่ละหลัก โดยโปรแกรมอินเตอร์รัพต์จะไปหาไบต์อักขระตามตำแหน่งของหลักในตำแหน่งของหน่วยความจำ เมื่อได้ข้อมูลไบต์จะส่งเป็นเอาต์พุตที่ P1 พอร์ต แล้วให้บิตลอจิกต่ำที่ P2 พอร์ต ที่บิตของหลักตัวเลข 7 ส่วน ก็สิ้นสุดการทำอินเตอร์รัพต์รูทีนครั้งหนึ่ง เมื่ออินเตอร์รัพต์ไทเมอร์เป็นช่วงเวลา ที่ไม่ทำให้ตาของมนุษย์จับการกระพริบได้ดังกล่าว



รูปที่ 4-6 *แสดงเป็นวงจรขับตัวเลข 7 ส่วนอย่างง่ายแบบทั่วไป*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-7 *การขับตัวเลข 7 ส่วนโดยวิธีตรง*

อย่างไรก็ดีการกระทำแบบการแสดงผลข้างต้นไม่เหมาะสม ในสำหรับงานวิจัยเพื่อนำไปออกแบบตัวคอนโทรลเลอร์นี้ เพราะว่าอินเตอร์รัพต์ TO ถูกสำรองไว้ใช้งาน และซีพียูต้องรับภาระ จึงกระทำโดยใช้วิธีขับตัวเลข 7 ส่วนโดยตรง คือขับ 1 หลักต่อ 1 ไบต์พอร์ต ซึ่งใช้วิธีนี้การแสดงผลจะไม่กระพริบอย่างสิ้นเชิง แต่อย่างไรก็ดีถ้าเป็นจำนวนหลักของตัวเลข 7 ส่วนมากวิธีนี้จะไม่เหมาะสมเช่นกัน แสดงดังรูปที่ 4-7 ดังต่อไปนี้

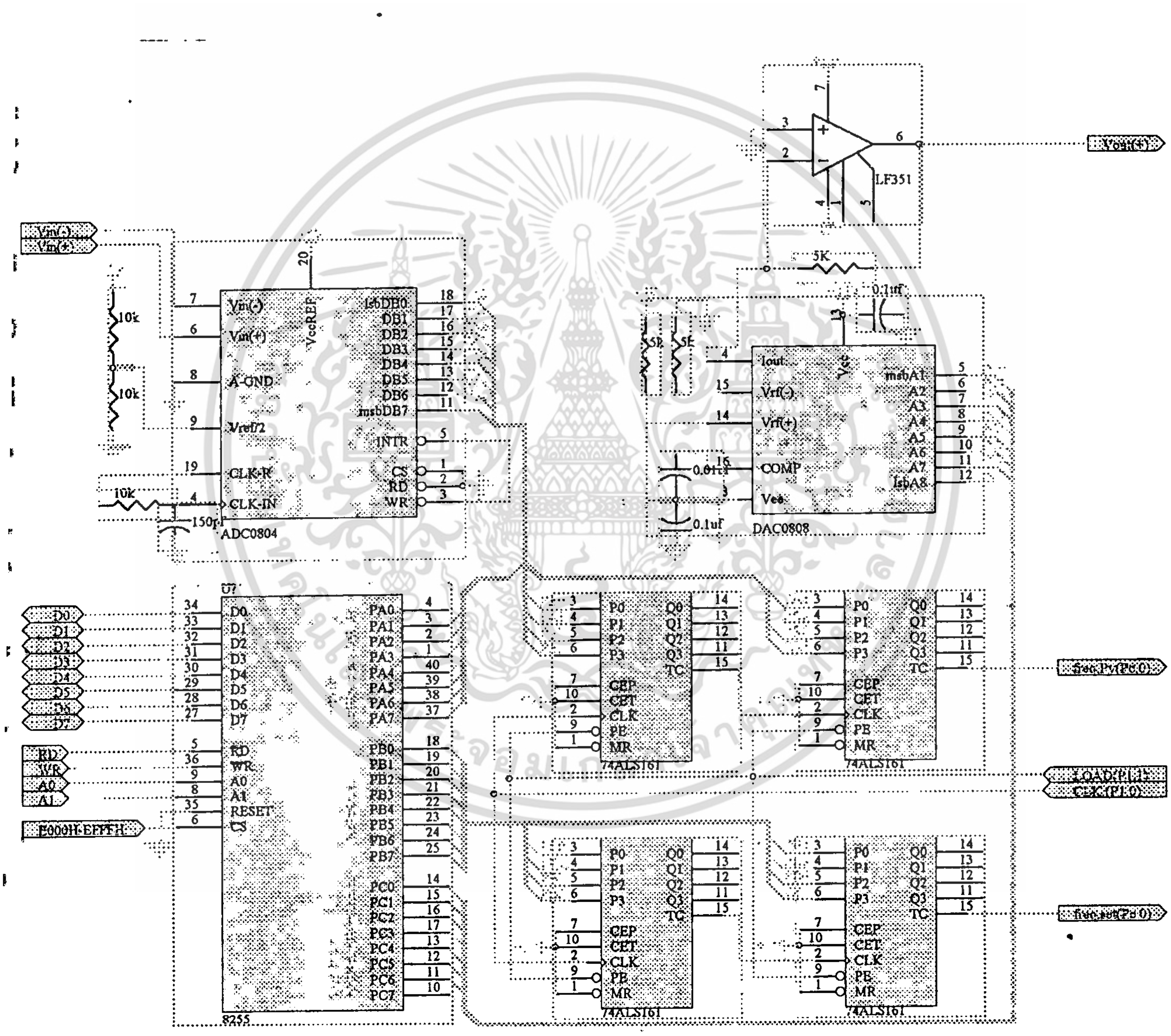
4.2.3 วงจร ADC และ DAC

เอาต์พุตตัวทรานสดิวเซอร์โดยมากจะเป็นแอนะล็อกโวลต์เตจ ซึ่งถ้าหากข้อมูลจากทรานสดิวเซอร์แบบนี้ถูกนำเข้าไปเก็บในระบบดิจิทัลแล้ว มันต้องถูกเปลี่ยนเข้าสู่สมมูลทางดิจิทัล คือการแปลงแอนะล็อกเป็นดิจิทัลนั่นเอง ตัว ADC จะประกอบด้วยคอมพาราเตอร์และ I/O โดยทำให้ได้สมบูรณ์จะต้องได้รับการถูกสั่งการจากโปรแกรมซอฟต์แวร์อีกที อย่างไรก็ดีวิธีนี้ไม่ได้เป็นการประหยัดเวลาของซีพียู ซึ่งมันต้องแบ่งเวลาในการขบวนการเปลี่ยนแปลงแอนะล็อกเป็นดิจิทัล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในทางกลับกันการเปลี่ยนแอนะล็อกเป็นดิจิทัลของ ADC นั้น สามารถกระทำให้บรรลุผลอย่างรวดเร็วได้โดยการใช้ ADC ในลักษณะของแฮนด์เชค(handshake) โดยยอมให้ ADC เป็นอิสระในขบวนการแปลงซึ่งจะไม่ขึ้นอยู่กับการทำงานของซีพียู เมื่อ ADC พร้อมที่จะส่งข้อมูลสู่ซีพียูก็จะมีสัญญาณ ready เพื่อบอกซีพียูเพื่อให้ซีพียูเตรียมพร้อม เพื่อที่ซีพียูจะส่งสัญญาณแฮนด์เชคไปที่ ADC ว่าพร้อมรับข้อมูลดิจิทับซึ่งจะได้รับข้อมูลจะ ADC ในทันที

วงจร ADC และ DAC แสดงดังรูป 4-8



รูปที่ 4-8 *แสดงวงจร ADC และ DAC*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีข้อมูล A/D คอนเวอร์เตอร์ตามวงจรดังนี้

- โวลต์เตจแอนะล็อกอินพุตศูนย์ โวลต์ถึง 5 โวลต์ ใช้กับแหล่งจ่ายไฟ 5 โวลต์ทางเดียว
- ไม่มีการปรับศูนย์ (zero adjust)
- ความละเอียด 8 บิต
- เออเรอร์ทั้งสิ้น + LSB ของ ADC0804
- เวลาการเปลี่ยน 100 uS ของ ADC0804

สำหรับ DAC ใช้ชิปสำเร็จรูปเบอร์ DAC0808 ซึ่งเป็นแบบ R-2R ซึ่งนิยมใช้โดยทั่วไป ถูกต่ออยู่กับพอร์ต PB0..7 ของ 8255 ถูกถอดรหัสแอดเดรสหมายเลข C000h-CFFFh ที่ต้องต่ออยู่กับพอร์ตเพราะว่า DAC 0808 ไม่มีตัวแลทช์อยู่ภายในจึงจำเป็นต้องใช้ I/O พอร์ต 8255 เป็นตัวแลทช์ โดยมีข้อมูลทางวิชาการดังนี้

- โวลต์อ้างอิง +Vref เป็น+5 โวลต์และ -Vref เป็น 0 โวลต์
- ใช้แหล่งจ่ายไฟคู่ + 12V
- ความละเอียด 8 บิต
- เวลาการแปลงข้อมูล 10 uS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

หลักการออกแบบซอฟต์แวร์และการใช้งาน

5.1 แนวทางการออกแบบซอฟต์แวร์ของตัวดิจิทัลคอนโทรลเลอร์

5.1.1 ไมโครคอนโทรลเลอร์และซอฟต์แวร์

เป็นที่รู้กันอยู่แล้วว่าไมโครคอนโทรลเลอร์พัฒนามาจากไมโครโปรเซสเซอร์ โดยมีเทคนิคการเขียนโปรแกรมที่เหมือนกัน แต่ต่างกันที่ไมโครคอนโทรลเลอร์ใช้งานได้กว้างขวางกว่า (generalperpose) ซึ่งในบทความนี้จะเน้นการใช้ไมโครโปรเซสเซอร์ 8031 ซึ่งเป็นไมโครโปรเซสเซอร์ 8บิต เหมือนกับไมโครโปรเซสเซอร์ Z80 ซึ่งเราทราบกันดีโดยมีคุณสมบัติ และข้อต่างกันดังนี้

	Z80	8031
การกำหนดขา		
จำนวนขาทั้งหมด	40	40
ขาแอดเดรส	16(fixed)	16
ขาดาต้า	8(fixed)	8
ขาอินเตอร์รัพต์	2(fixed)	2
ขา I/O	0	32
สถาปัตยกรรม		
เรจิสเตอร์ 8 บิต	20	34
เรจิสเตอร์ 16 บิต	4	2
ขนาดสแต็ก(stack)	64K	128
รอม(ROM) ภายใน	0	0
แรม(RAM) ภายใน	0	128
หน่วยความจำภายนอก	64K	128K
แฟลก(flags)	6	4
ไทเมอร์(timer)	0	2
พอร์ตขนาน(parallel port)	0	4
พอร์ตอนุกรม	0	1
เซตคำสั่ง		
คำสั่ง MOV ภายนอก	4/14	2/6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง MOV บล็อก	2/4	0
การจัดการบิต	4/4	12/12
คำสั่ง jump บนบิต	0	3/3
แสตก(stack)	3/15	2/2
คำสั่ง single-byte	203	49
คำสั่ง multi-byte	490	62

<table>
<tr><td>28· ไบต์ 7Fh
UPPER
(8032 ONLY)

ACCESSIBLE
BY INDIRECT</td><td>SFR
BIT ACCESSIBLE
80h - Ffh
ACCESSIBLE
BY DIRECT</td></tr>
<tr><td>ACCESSIBLE BY 5Fh
DIRECT AND INDIRECT</td><td></td></tr>
<tr><td>BIT ACCESSIBLE
SPACE (00 - 7F)</td><td></td></tr>
<tr><td>REGISTER BANK</td><td></td></tr>
</table>

ตารางที่ 5-1 *แสดงการจัดหน่วยความจำของไมโครคอนโทรลเลอร์ 8031*

จุดนี้ไม่ได้ชี้ว่าอย่างไหนดีกว่ากัน ซึ่งทั้งสองออกแบบการใช้งานในจุดประสงค์ ต่างกันมีบางซีพียู เช่น Z80 นั้นมีคำสั่งแบบหลายไบต์(multi-byte) ถึง 71 เปอร์เซ็นต์ ซึ่งการใช้คำสั่งแบบหลายไบต์ นี้ต้องเสียเวลาเฟทซ์คำสั่ง (fetch) จากหน่วยความจำของโปรแกรม ดังนั้นในลักษณะงานที่คล้ายกัน 8031 จะทำงานได้เร็วกว่า

การใช้งานพื้นที่แรมภายใน(internal data RAM) ของไมโครคอนโทรลเลอร์ตระกูล MCS51 จัดแบ่งหน่วยความจำภายในของไมโครคอนโทรลเลอร์แสดงเป็นตาราง 5-1 ได้ดังนี้

หน่วยความจำภายใน(intermal RAM) ดังกล่าวนำใช้ประโยชน์โดยมาใช้เป็นพื้นที่กำหนดตัวแปรต่าง ๆ ซึ่งใช้ในมอนิเตอร์โปรแกรมของวงจรดิจิทัลคอนโทรลเลอร์ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS RAM แอดเดรสแรม	VARIABLE ตัวแปร	คำบรรยาย
47H	DT	เป็นค่าเวลาแซมปลิง
48H	SUM0	เป็นตัวแปร SUM ไบต์แรกใช้ในโหมด integal
49H	SUM1	เป็นตัวแปร SUM ไบต์สองใช้ในโหมด integal
4AH	CD	DATA CONTROL ของโหมด I - CONTROL
4BH	DV0	เป็นข้อมูลจาก A/D แซมปลิงก่อนหน้า
4CH	DE	เออเรอร์เอาต์พุต DSP - DV0
4DH	KD0	เป็นไบต์แรกของค่า KD
4EH	KD1	เป็นไบต์สองของค่า KD
4FH	DCP1	DATA CONTROL ของโหมด PI - CONTROL
50H - 7FH	stack area	เป็นพื้นที่สำหรับ stack

2. พื้นที่หน่วยความจำระหว่างแอดเดรส 20H - 2AH โดยพื้นที่นี้จะใช้ตัวแปรไบต์ร่วมกับบิตแฟลกดังนี้

ADDRESS RAM แอดเดรส	VARIABLE ตัวแปร	คำบรรยาย
20H	Flag area	พื้นที่ที่ใช้เป็นแฟลก
21H		
22H		
23H		
24H		
25H	reserve	พื้นที่สำรองสำหรับใช้งาน
26H		
27H		
28H		
29H		
2AH		

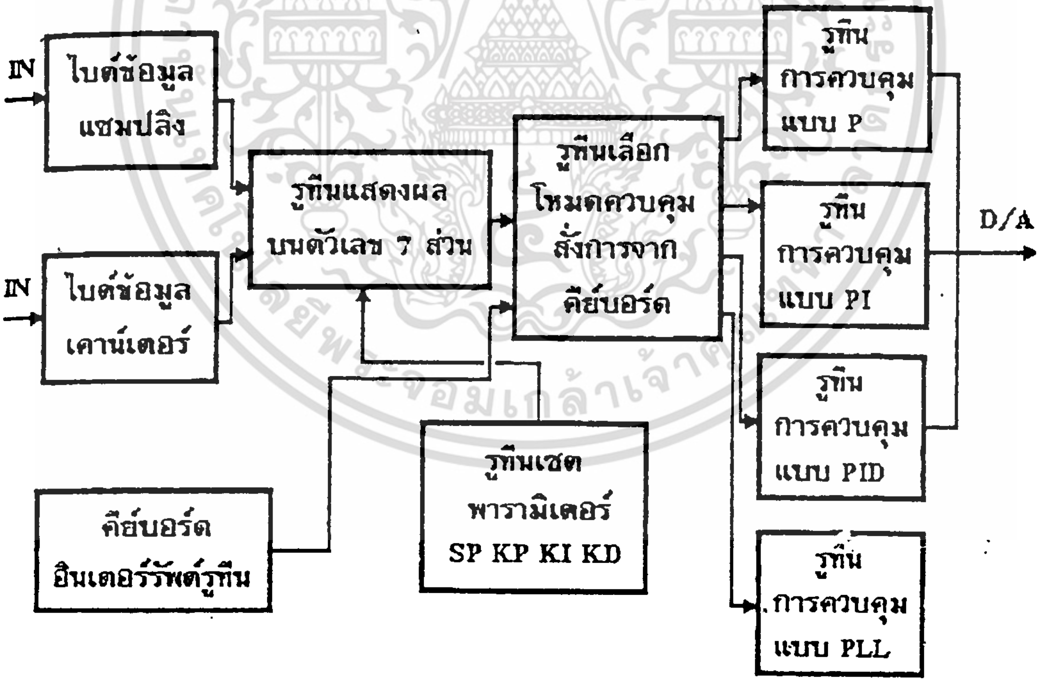
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.2 โครงสร้างของโปรแกรมมอนิเตอร์ของตัวดิจิทัลคอนโทรลเลอร์

หลักการออกแบบซอฟแวร์ เพื่อเป็นโปรแกรมมอนิเตอร์ให้กับตัวคอนโทรลเลอร์ โดยใช้ไมโครโปรเซสเซอร์ตระกูล MPS-51 ของบริษัทอินเทล โดยใช้สัญญาณนาฬิกา ๆ/ เมกะเฮิรตซ์ จึงมีความเร็วพิเพียงที่จะเขียนซอฟแวร์โปรแกรม ซึ่งโปรแกรมนี้จะใช้งานร่วมกับกระบวนการ ทำให้โปรแกรมนี้ต้องทำงานในลักษณะเป็นเวลาจริง (real time) โดยที่ควบคุมลูปคอนโทรลได้ 1 ลูปซึ่งสามารถต่อแบบ เครือข่าย สามารถควบคุมลูปคอน โทรลในพล้านต์ (plant) ได้ถึง 12 ลูปโดยใช้ โฮส (host) ช่วยเหลือ โปรแกรมสำหรับตัวคอนโทรลนี้แบ่งออกเป็น 4 โหมดด้วยกัน ดังนี้

1. โหมดการควบคุมพรอปพรอชันแนล หรือ P
2. โหมดการควบคุมพรอปพรอชันแนล-อินทีกรัล หรือ PI
3. โหมดดารควบคุมพรอปพรอชันแนล-อินทีกรัล-ดีริเวทีฟ หรือ PID
4. โหมดการควบคุมเฟสล็อก หรือ PLL

มีบล็อกไดอะแกรมการทำงานของโปรแกรมของซอฟแวร์รูทีนต่างๆที่จัดการกับตัวดิจิทัลคอนโทรลเลอร์ เป็นดังนี้



รูปที่ 5-1 *บล็อกไดอะแกรมของซอฟแวร์โปรแกรมมอนิเตอร์ของคอนโทรลเลอร์*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 5-1 อธิบายได้ว่า อย่างแรก อินพุตของตัวดิจิทัลคอลโทรลเลอร์นั้นแบ่งอินพุต 2 ประเภทด้วยกัน คือ

1. แรงดันอินพุตซึ่งป้อนกลับมาจากกระบวนการมีช่วงระหว่าง 0 โวลต์ ถึง 5 โวลต์ โดยที่แรงดัน 0 โวลต์ เทียบเป็น 0 เปอร์เซ็นต์ และ 5 โวลต์เทียบเป็น 100 เปอร์เซ็นต์ของการการควบคุม ซึ่งอินพุตนี้ใช้กับโหมดควบคุมแบบ P,PI และ PID

2. ความถี่อินพุตซึ่งป้อนกลับมาจาก เอนโคเดอร์ (encoder) หรือแหล่งความถี่ใด ๆ โดยมีช่วงความถี่ระหว่าง 60 เฮิรตซ์ ถึง 1.5 กิโลเฮิรตซ์ โดยจะเทียบเป็น 0 เปอร์เซ็นต์ และ 100 เปอร์เซ็นต์ ตามลำดับ โดยอินพุตแบบนี้ใช้เฉพาะโหมดควบคุมแบบ PLL เท่านั้น

5.1.3 การจัดการทำงานของโปรแกรม

จากรูปเนื่องจากเป็นระบบควบคุมแบบเวลาจริง (real time) ทำให้กระบวนการในประมวลผลของไมโครโปรเซสเซอร์ตามโปรแกรมนั้นต้องเร็วที่สุด ที่ซึ่งการทำงานของไมโครคอนโทรลเลอร์เบอร์ 8031 ใช้สัญญาณาฬิกา 12 เมกะเฮิรตซ์ จะได้เปรียบกว่าไมโครโปรเซสเซอร์เบอร์ Z80 หรือ 8085 ซึ่งต่างก็เป็น ไมโครโปรเซสเซอร์แบบ 8 บิต ในกระบวนการตั้งแต่อินพุตผ่านรูทีนของโปรแกรมต่างจนกระทั่งได้ เป็นค่า CV(control variable) ไปควบคุมกระบวนการจะใช้เวลาของแต่ละโหมดต่างกันดังนี้

1. โหมด P จะใช้เวลาการคำนวณรวมทั้งหมดประมาณ 17 มิลลิวินาทีโดย โหมดนี้จะผ่านซอฟต์แวร์ โปรแกรมจำนวนคำสั่งน้อยที่สุด

2. โหมด PI ใช้เวลาประมาณ 20 มิลลิวินาที

3. โหมด PID ใช้เวลาประมาณ 21 มิลลิวินาที

4. โหมด PLL ใช้เวลาประมาณ 0.6 มิลลิวินาที เหตุที่น้อยที่สุดเพราะว่าทั้ง 3 โหมดแรกใช้ซอต์แวร์การคำนวณทางคณิตศาสตร์ซึ่ง IC 8031 ทำได้ไม่เร็วนัก ด้วยเหตุนี้โหมด PLL ไม่ใช้การคำนวณทางคณิต-ศาสตร์และรูทีนมีคำสั่งน้อย จึงทำให้เวลาน้อยที่สุด

5.1.4 หน้าที่ของรูทีนคีย์บอร์ด

หน้าที่ของซอฟต์แวร์จัดการคีย์บอร์ด หรือเรียกว่าไบออส (bios) สำหรับการจัดการคีย์บอร์ดเพื่อให้ผู้ใช้สามารถติดต่อกับตัวดิจิทัลคอนโทรลเลอร์ได้ โดยใช้เซตการทำหน้าที่ต่างดังนี้

1. เลือกโหมดได้ 4 โหมดดังกล่าวมาแล้ว

2. เลือก setpoint (SP) ได้ตั้งแต่ 0 - 100 เปอร์เซ็นต์

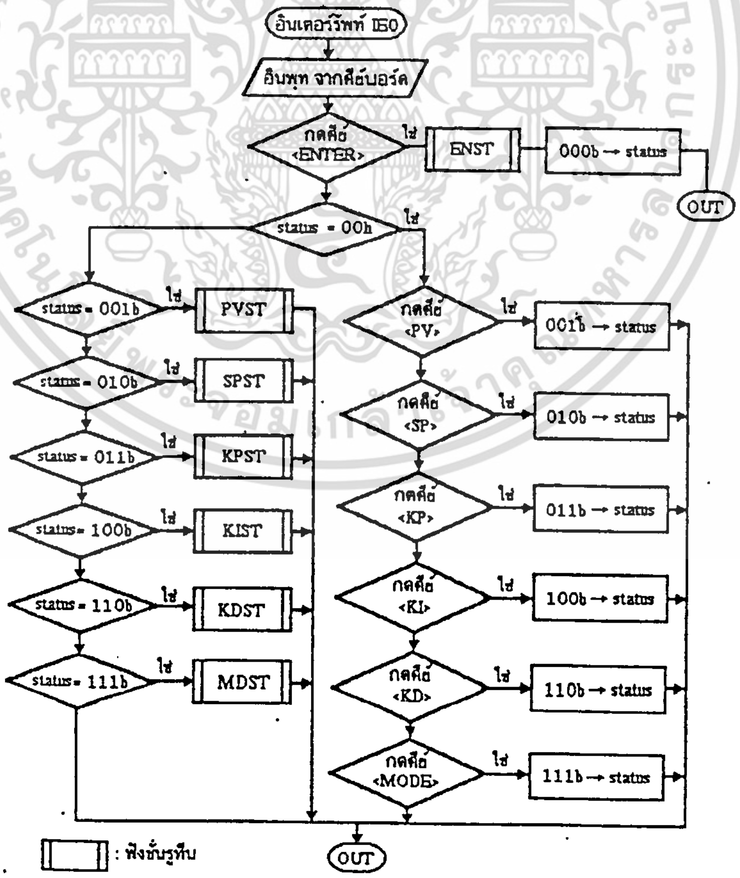
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เลือกค่า KP (proportional gain) ได้ตั้งแต่ 0 - 200 เท่า
4. เลือกค่า KI (integal time) ได้ตั้งแต่ 0.01 - 99 วินาที
5. เลือกค่า KD (dirivative time) ได้ตั้งแต่ 0.01 - 99 วินาที
6. แสดงค่า PV (process variable)

เนื่องจากการจัดการทั้ง 6 หัวข้อของคีย์บอร์ดดังกล่าว ต้องไม่ทำให้การทำงานปกติในการควบคุมกระบวนการในโหมดใดก็แล้วแต่ ต้องไม่หยุดการทำงาน เพราะว่าจะเกอดการเสียหายแก่กระบวนการ ดังนั้นซอฟต์แวร์รูทีนสำหรับคีย์บอร์ด จึงต้องใช้วิธีอินเตอร์รัพต์โดยใช้ IEO คืออินเตอร์รัพต์ภายนอกเบอร์ 0 ของขา 12 ของ IC 8031 ซึ่งจะชี้ไปทำอินเตอร์รัพรูทีนที่แอดเดรส 0003H อย่างไรก็ตามเพื่อไม่ให้เสียเวลาในรูทีนอินเตอร์รัพต์นี้ เพราะการต้องรอคีย์บอร์ดต่อเนื่อง ดังนั้นทำให้ต้องเก็บสถานะเดิม ของคีย์บอร์ดไว้เพื่อไม่ต้องรอ ซึ่งจะใช้เวลาในอินเตอร์รัพต์รูทีนนี้น้อยมาก

คีย์บอร์ดหรือแป้นคีย์(keypad) ประกอบด้วยปุ่ม (botton) ใช้งานทั้งหมด 12 คีย์คือ <ENTER>, <MODE>,<UP>,<DOWN>,<SP>,<ADDS>,<PV>,<EDIT>,<KP>,<KI>,<KD> และคีย์สำรองส่วนวิธีใช้จะนำเสนอต่อไป

โฟล์ชาร์ต การทำงานของคีย์บอร์ดรูทีน มีรูปแบบการทำงานตามโฟล์ชาร์ตไดอะแกรม ตามรูป 5-2



รูปที่ 5-2 *บล็อกไดอะแกรมของซอฟต์แวร์รูทีนคีย์บอร์ด*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีลักษณะอธิบายได้ดังนี้ สังเกตได้ว่าบล็อก IF เป็นรูปเพชร๗ แบ่งเรียงเป็นสองแถวทางด้านซ้ายและด้านขวา ซึ่งการจัดบล็อก IF ลงมาในลักษณะนะทำหน้าที่เหมือนคำสั่ง CASE ในภาษาระดับสูง รูทีนหรือซับรูทีนนี้จะทำงาน โดยตัวไมโครคอนโทรลเลอร์ถูกอินเตอร์รัพต์จากวงจรรับคีย์บอร์ดภายนอก ตัวซอฟต์แวร์จะทำการตรวจสอบว่าเป็น ENTER คีย์หรือแทนเป็น <ENTER> หรือไม่ ถ้า yes จะทำซับรูทีน ENST ต่อไป โดยที่ no โปรแกรมจะตรวจสอบสถานะจากตัวแปร status ซึ่งตัวแปรนี้จะเก็บสถานะคีย์เดิมไว้ เมื่อตรวจสอบสถานะแล้วปรากฏว่าเท่ากับ 00h แสดงว่าสถานะคีย์เดอมนั้นไม่มีดังนี้จะทำงานตามบล็อก IF ทางด้านขวาซึ่งมีหน้าที่เปลี่ยนค่าในตัวแปร status ตามสถานะ คีย์ปัจจุบันเช่น <PV>,<SP> เป็นต้น ตรงกันข้าม ถ้า status = 00h จะทำตามซับรูทีนทางด้านขวามือ ซึ่งเป็นรูทีนไปเซตพารามิเตอร์ต่างของการควบคุมของตัวคอนโทรลเลอร์ดังนี้

1. *สัปรูทีน PVST* เป็นรูทีนเพื่อให้ดิสเพลย์ ของคอนโทรลเลอร์แสดงค่า PV (process variable)

2. *สัปรูทีน SPST* เป็นรูทีนสำหรับเลือกค่า SP หรือเซตพอย์น โดยที่จะแสดงผลเซตพอย์นปัจจุบัน (ถ้ารี เซตค่าเซตพอย์นจะเป็นที่ 50 เปอร์เซนต์)

3. *สัปรูทีน KPST* เป็นรูทีนสำหรับเลือกค่า KP คือพรอปพรอลันแนลเกน

4. *สัปรูทีน KIST* เป็นรูทีนสำหรับเลือกค่า KI คืออินทีกรัลเกนมีหน่วยเป็นวินาที

5. *สัปรูทีน KDST* เป็นรูทีนสำหรับเลือกค่า KD คือดีริเวทีฟเกนมีหน่วยเป็นวินาที

6. *สัปรูทีน MDST* เป็นรูทีนสำหรับเลือกโหมดซึ่งมีสามโหมด มีรูทันดังนี้ P-control PI-control และ PID-control PLL-control

5.1.5 หน้าที่ของรูทีนแสดงผลบนตัวเลข 7 ส่วน

จอแสดงผลของตัวดิจิทัลคอนโทรลเลอร์จะใช้ตัวเลข 7 ส่วนจำนวน 6 หลักด้วยกัน รูทีนต่าง ๆ ซึ่งเกี่ยวข้องกับการแสดงผลบนตัวเลข 7 ส่วน ซึ่งใช้จำนวนตัวเลข 7 ส่วน 6 หลักนี้คือ สองหลักแรกทางซ้ายมือแสดงตัวแปรต่าง ๆ เช่น PV SP เป็นต้น หลักที่สามแสดงเคอร์เซอร์ ส่วนสามหลักสุดท้ายแสดงค่าตัวแปรเป็น เปอร์เซ็นต์ หมายเลข วินาที ซึ่งแล้วแต่หน้าที่ของคีย์บอร์ดในขณะนั้น

รูทีนที่เกี่ยวข้องกับตัวเลข 7 ส่วนนั้นจะรวมอยู่ในไฟล์ 7-SEG.ASM มีรูทีนต่าง ๆ ดังนี้

1. สัปรูทีน SHOWDISP1 ซึ่งรูทีนนี้จะนำค่าที่อยู่ภายในเรจิสเตอร์ R0 และ R1 แสดงบนตัวเลข 7 ส่วน หลักที่ 1 หลักที่ 2 และหลักที่ 3 นับจากขวามือ ซึ่งผลที่แสดงบนจอแสดงผลจะเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

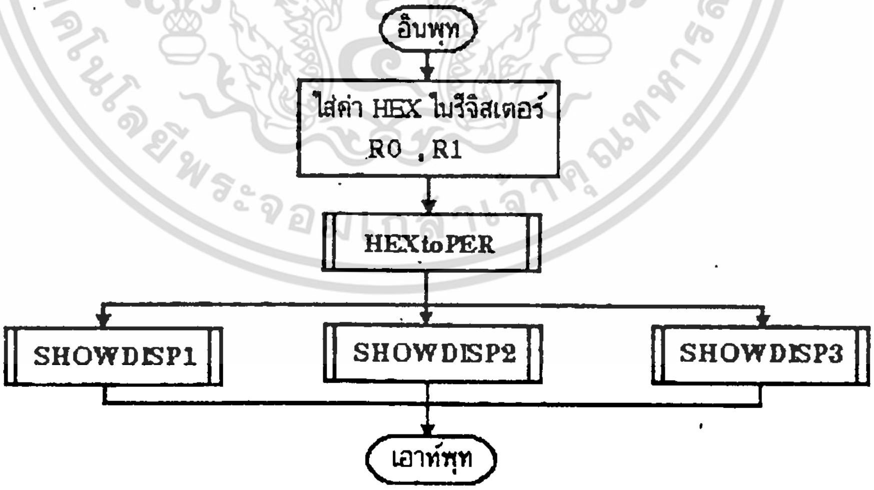
เลขฐาน 16 ไม่เกินค่า FFFh สมมุติว่า จะแสดงผลบนตัวดิจิทัลคอนโทรลเลอร์เป็นจำนวนตัวเลขฐาน 16 เป็น 57C เราต้องให้ค่าตัวเลขกับเรจิสเตอร์ R0 เป็น 7Ch และ R1 เป็น 05h เป็นต้น

2. สัปรูทีน SHOWDISP2 หน้าที่เป็นเช่นเดียวกับ รูทีน SHOWDISP1 แต่จะแสดงผลบนตัวเลข 7 ส่วน หลักที่ 4 หลักที่ 5 และหลักที่ 6 จากขวามือ

3. สัปรูทีน SHOWDISP3 หน้าที่เป็นเช่นเดียวกับ รูทีน SHOWDISP2 แต่ต่างกันตรงที่การแสดงผลนั้น จะเป็นช่องว่าง(blank) ถ้าหลักแรกของตัวเลขเป็นศูนย์ ตัวอย่างเช่นถ้าค่าเป็น 0 C5h ถ้าป็นรูทีน HOWDISP1 หรือ SHOWDISP2 จะแสดงผลบนจอแสดงผลเป็น 0C5h แล้วซึ่งหน้าที่ของรูทีน SHOWDISP3 จะแสดงค่าเป็น 5C สังเกตว่าข้างหน้าตัวเลข 5 จะไม่มีเลข 0 โดยให้เป็นช่องว่างไป

4. สัปรูทีน HEXtoPER หน้าที่ของรูทีนคือแปลงตัวเลขฐาน 16 เป็นค่าเปอร์เซ็นต์ โดยเทียบจาก 00h เป็น 0 เปอร์เซ็นต์ และ Ffh เป็น 100 เปอร์เซ็นต์ ซึ่งวิธีการของรูทีนนี้จะใช้การเปิดตาราง โโยตารางที่ ใช้คือ TABLEHD0, TABLEHD1, TABLEHD2, TABLEHD0,TABLEHP1, TABLEHP2 มีตาราง จำนวนทั้งสิ้น 6 ตารางด้วยกัน ตัวแปรอินพุตของรูทีนนี้เป็น เรจิสเตอร์ R0 และ R1 โดยที่ R0 แทน บิตนัยสำคัญต่ำและ R1 แทนบิตนับสำคัญสูงเมื่อเข้าสู่ประบวนการในรูทีน HEXtoPER แล้วจะให้เอาต์ พุตเป็นค่าของเปอร์เซ็นต์ที่ R0 และ R1 ถึงแม้จะสิ้นเปลืองเนื้อที่มอนิเตอร์โปรแกรม ถ้าเทียบกับ ความเร็วในการแสดงผลการแสดงผลถือว่าคุ้มค่า

5. สัปรูทีน INITDISP มีหน้าที่เซตค่าเริ่มต้นของฮาร์ดแวร์ ที่ซึ่งสนับสนุนการแสดงผล



รูปที่ 5- 3 *แสดงโฟล์วชาร์ตของการแสดงผลเป็นค่าเปอร์เซ็นต์*

ทั้งหมดทั้ง 5 รูทีนนั้นไม่มีความสำคัญอะไร ที่ต้องแสดงรายละเอียดของโฟล์วชาร์ตของแต่ละรูทีน เพราะว่าไม่เกี่ยวข้องกับคุณสมบัติความสามารถของตัวดิจิทัลคอนโทรลเลอร์แต่อย่างใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่เป็นเพียงรูทีนสนับสนุนเท่านั้น อย่างไรก็ดีถ้าต้องการรายละเอียดแต่ละรูทีนสามารถดูได้จากไฟล์ "7-SEG.ASM" ได้ ซึ่งโฟล์ชาร์ตการแสดงผลบนจอแสดงผลของดิจิทัลคอนโทรลเลอร์แสดงดังรูปที่ 5-3

5.1.6 หน้าที่ของรูทีนเซตพารามิเตอร์ของตัวดิจิทัลคอนโทรลเลอร์

พารามิเตอร์ต่างของตัวดิจิทัลคอนโทรลเลอร์เช่น เซตพอย์น(SP), พรอปพรอชันแนลเกน (KP),อินทีกรับเกน(KI), ดีริเวทีฟเกน(KD) เป็นต้น จำเป็นต้องเปลี่ยนแปลงตลอดเวลาเพื่อให้เหมาะสมกับกระบวนการในขณะนั้น การเซตนั้นจะทำโดยคีย์บอร์ด ที่ซึ่งการทำงานของคีย์บอร์ดนั้นได้อธิบายไว้ข้างต้นอย่างชัดเจน ซึ่งในหัวข้อนี้จะอยู่ในซอร์สโค้ดแอสเซมบลีในไฟล์ชื่อว่า "ALLKTabl.ASM " และไฟล์ "INTKEY2.ASM" โดยที่ไฟล์แรกเกี่ยวข้องกับการแสดงผล และไฟล์สองเกี่ยวกับการนำสถานะคีย์บอร์ดมากระทำ

ในไฟล์ ALLKTable.ASM ดังกล่าวประกอบด้วยสัปรูทีนต่างๆที่สำคัญดังนี้

1. สัปรูทีน LookATKtable มีหน้าที่รับค่าจากตัวแปรเกนเช่น KP0,KP1 ซึ่งเป็นพรอปพรอชันแนลเกนใช้ จำนวนบิต 16 บิตโดยบิตนัยสำคัญต่ำจะอยู่ใน KP0 และบิตนัยสำคัญสูงจะอยู่ใน KP1 ที่ซึ่งจำเป็นต้อง ใช้ถึง 16 บิตเพราะว่าต้องใช้เป็นทศนิยม 4 บิต เช่นเดียวกับ KI0 ,KI1 ของอินทีกรัลเกน และ KD0,KD1 ของดีริเวทีฟเกน

สับรูทีนนี้มีอินพุตตัวแปรเป็นเรจิสเตอร์ R0,R1 โดยข้อมูลของเรจิสเตอร์แทนจาก KP0,KP1 หรือKI0,KI1 หรือ KD0,KD1 ซึ่งรูทีนนี้จะเอาข้อมูลจากเรจิสเตอร์ไปเปิดตาราง โดยข้อมูลที่ได้จากการเปิดตารางจะเป็นข้อมูลใหม่ ซึ่งพร้อมจะนำไปแสดงผลบนจอแสดงผลของตัวคอนโทรลเลอร์โดยข้อมูลใหม่จะเข้าสู่รูทีน ShowKTab ก่อนจะแสดงผล

2. สัปรูทีน ShowKTab มีหน้าที่เข้าข้อมูลใหม่ดังกล่าวอยู่ในเรจิสเตอร์ R3,R4 นำไปแสดงผลบน จอแสดงผลของตัวคอนโทรลเลอร์เป็น 3 หลักด้วยกัน ซึ่งค่าที่ได้จากการเปิดตารางเมื่อนำไปแสดงผล จะได้ค่าต่ำสุดเป็น 0.001 และสูงสุดเป็น 999 โดยสามารถครอบคลุมได้ทั้งถึงค่าปกติซึ่งใช้อยู่ในทั่ว ไปของคอนโทรลเลอร์ และมีตาราง LOOK-UP คือ tableK0, tableK1, tablePK0, tablePK1

ในไฟล์ INTKEY2.ASM ยกมาอธิบายเฉพาะที่เกี่ยวกับการเซตค่าเกนดังนี้

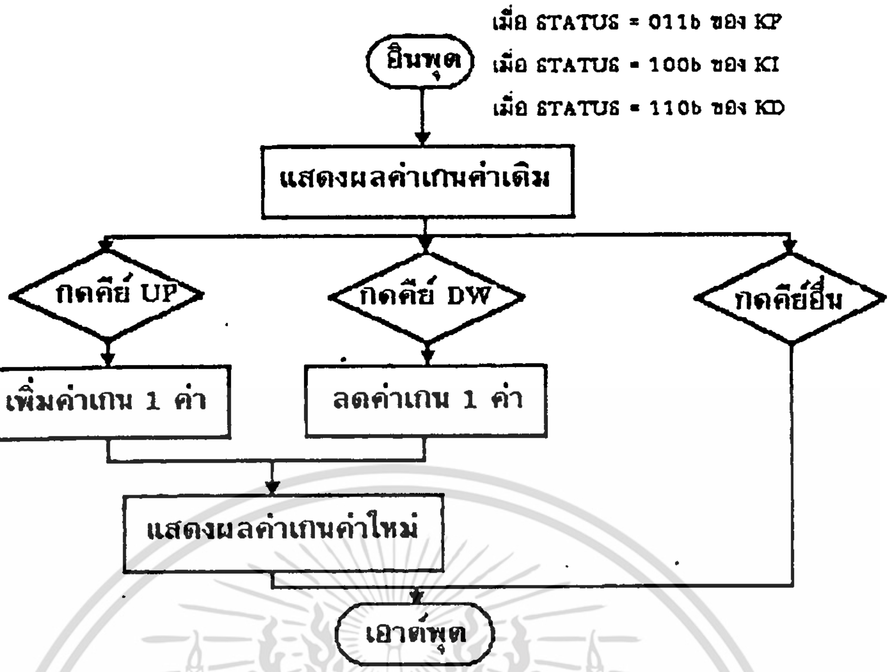
1. สัปรูทีน KPst เกี่ยวข้องกับการเปลี่ยนค่าพรอปพรอชันแนลเกน

2. สัปรูทีน KIst เกี่ยวข้องกับการเปลี่ยนค่าอินทีกรับเกน

3. สัปรูทีน KDst เกี่ยวข้องกับการเปลี่ยนค่าดีริเวทีฟเกน

โดยทั้ง 3 รูทีนมีลักษณะโฟล์วชาร์ตคล้ายคลึงกัน ดังนี้ ตามรูปที่ 5-4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-4 *แสดงโฟล์วชาร์ตของการเปลี่ยนค่าเกน*

5.2 วิธีการใช้งานตัวดิจิทัลคอนโทรลเลอร์

เริ่มแรกคือต่อแหล่งจ่ายไฟเข้าให้เรียบร้อยตามลำดับได้แก่ +5 โวลต์ +12 โวลต์ -12โวลต์ และกราวด์ให้เรียบร้อย และนำไปควบคุมลูปของกระบวนการโดยนำตัวดิจิทัลคอนโครลเลอร์ไปต่อให้ครบลูป ซึ่งจะใช้แบบการควบคุมแบบต่อเนื่องหรือแบบเฟสล็อกลูป ซึ่งขั้วต่อต่างกัน แสดงตามรูปดังนี้ 5-15 แสดงหน้าปัดด้านหลังของตัวดิจิทัลคอนโทรลเลอร์ ซึ่งประกอบด้วยขั้วต่อหลายแบบประกอบด้วยขั้วต่อ RS232 อยู่ด้านขวาใช้ติดต่อกับโฮสคอมพิวเตอร์ ขั้วต่อ I/O ซึ่งประกอบด้วย INP,OUTP ใช้กับการควบคุมแบบต่อเนื่องโหมดปกติเช่น P,PI,PID และ PD outp, PD inp ใช้การควบคุมแบบเฟสล็อกโหมด ขั้วต่อซัพพลาย (SUPPLY) ใช้สำหรับต่อแหล่งจ่ายแรงดันไฟฟ้าเข้า ขั้วทั้งหมดที่กล่าวมาสามารถจำแนกได้ดังนี้

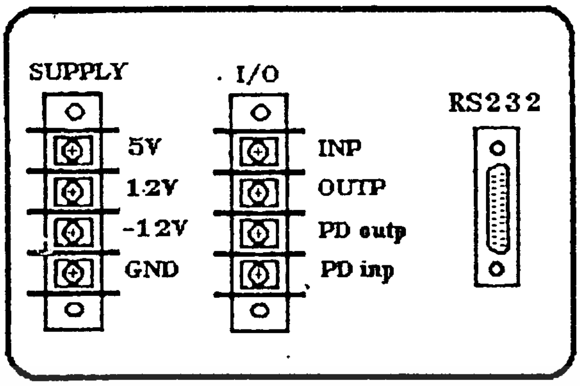
INP เป็นขั้วสัญญาณอินพุตจากกระบวนการ มีแรงดันอยู่ระหว่าง 0 โวลต์ถึง 5 โวลต์

OUTP เป็นขั้วสัญญาณเอาต์พุตสู่กระบวนการ มีแรงดันอยู่ระหว่าง 0 โวลต์ถึง 5 โวลต์

PD outp เป็นขั้วสัญญาณเอาต์พุตของเฟสดีเทกเตอร์(ซอฟต์แวร์) มีลักษณะเป็นไตร-สเตต

PD inp เป็นขั้วต่อสัญญาณอินพุตป้อนกลับจากกระบวนการ โดยที่ต้องเป็นสัญญาณรูปคลื่นสี่เหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-5 *แสดงหน้าปัดด้านหลังของตัวคอนโทรลเลอร์*

ต่อจากนี้จะกล่าวถึงขั้นตอนการปฏิบัติในการใช้งานของดิจิทัลคอนโทรลเลอร์ ดังนี้

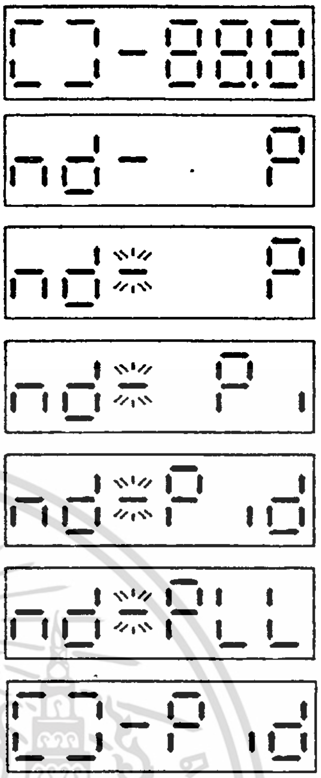
-เมื่อเริ่มเปิดเครื่อง(จ่ายแรงดันไฟฟ้าเข้าแล้ว) สังเกตว่าที่จอแสดงผลของตัวดิจิทัลคอนโทรลเลอร์จะแสดง คำว่า "READY"

-จากนั้นกดปุ่มคีย์ ENTER แล้วที่จอแสดงผลก็แสดงเป็นรูปวงเล็บใหญ่ "{}" หลักจากนั้นก็เป็นการเซต โหมดของการควบคุมต่างดังนี้

5.2.1 การเซตโหมดการควบคุม

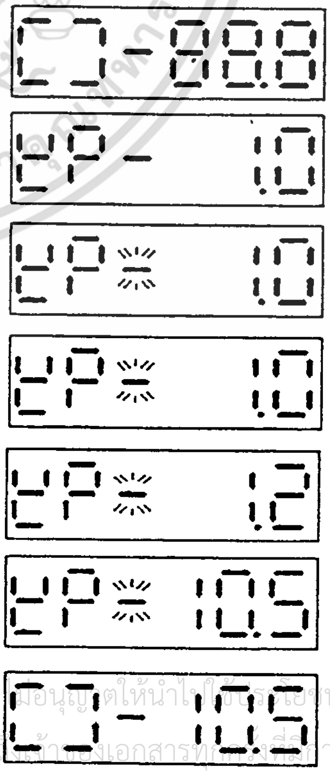
ตัวคอนโทรลเลอร์นั้นกล่าวได้ว่าส่วนที่สำคัญที่สุดคือวิธีการควบคุม เมื่อรีเซตเครื่องดิจิทัลคอนโทรลเลอร์ จะดีฟอลต์(default) การควบคุมเป็นแบบโหมด P ถ้าต้องการเซตโหมดใหม่ทำได้โดยปฏิบัติตามการกดคีย์เรียงลำดับดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



1. เตรียมพร้อม — ENTER
2. แสดงโหมดการควบคุมปัจจุบัน — MODE
3. แก้ไขโหมด — EDIT
4. เปลี่ยนโหมด — △ OR ▽
5. รับโหมดการควบคุมใหม่ — ENTER

5.2.2 การเซตพารามิเตอร์การควบคุม KP, KI, KD

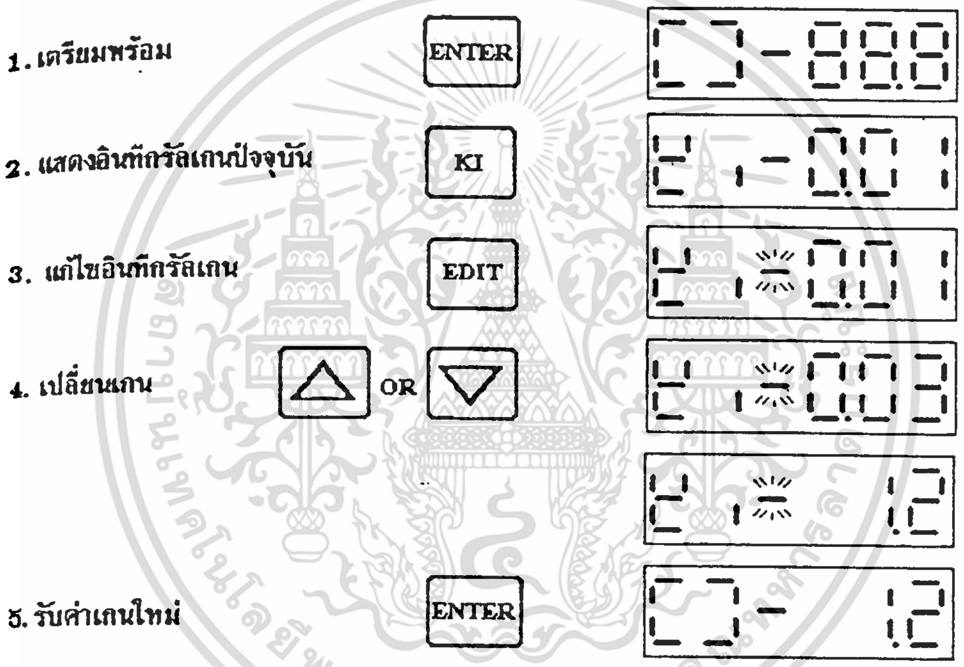


1. เตรียมพร้อม — ENTER
2. แสดงพรอพพอชั่นนอลเกน — KP
3. แก้ไขพรอพพอชั่นนอลเกน — EDIT
4. เปลี่ยนเกน — △ OR ▽
5. รับค่าเกนใหม่ — ENTER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

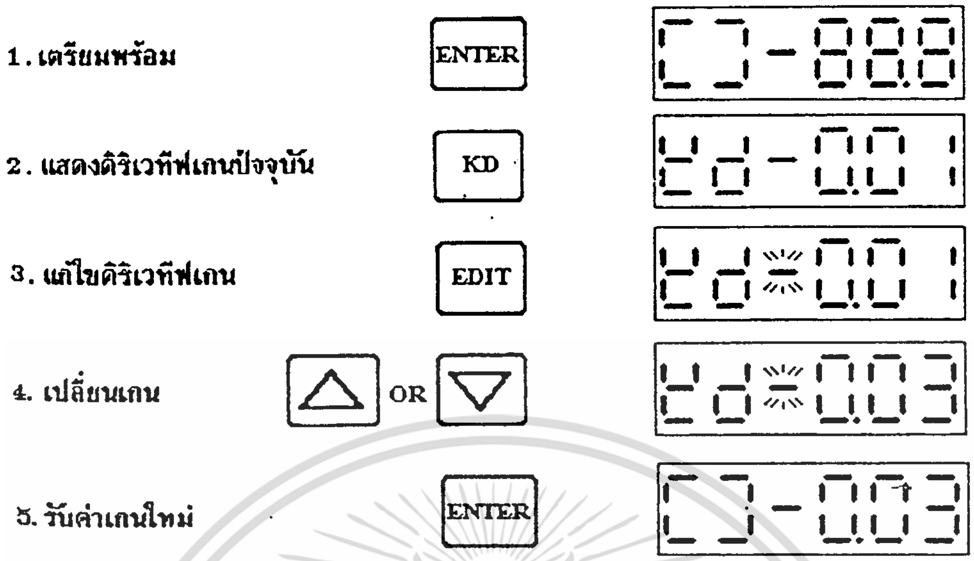
ซึ่งค่าของ KP มีหน่วยเป็นจำนวนเท่า โดยถูกนิยามเหมือนกับอัตราส่วนการเปลี่ยนแปลงต่อค่าเออเรอร์กับช่วงอินพุต โดยสามารถเซตค่า KP ได้ตั้งแต่ 0.01 ถึง 999 เท่า เช่น ถ้า KP เป็น 1 จะไม่มีการขยายสัญญาณ CV(control variable) ของตัวดิจิทัลคอนโทรลเลอร์เป็นต้น

2. วิธีเซตค่าอินทีกรัลเกน KI ค่าพารามิเตอร์ KI จะมีผลต่อการควบคุมแบบ PI, และ PID เท่านั้น สมมุติว่าใช้การควบคุมแบบ P หรือแบบ PLL ถึงแม้จะเซตค่าพารามิเตอร์ KI เป็นอย่างไรก็ตามก็จะไม่มีผลในการควบคุมเหล่านั้น ค่า KI มีหน่วยเป็นวินาที $^{-1}$



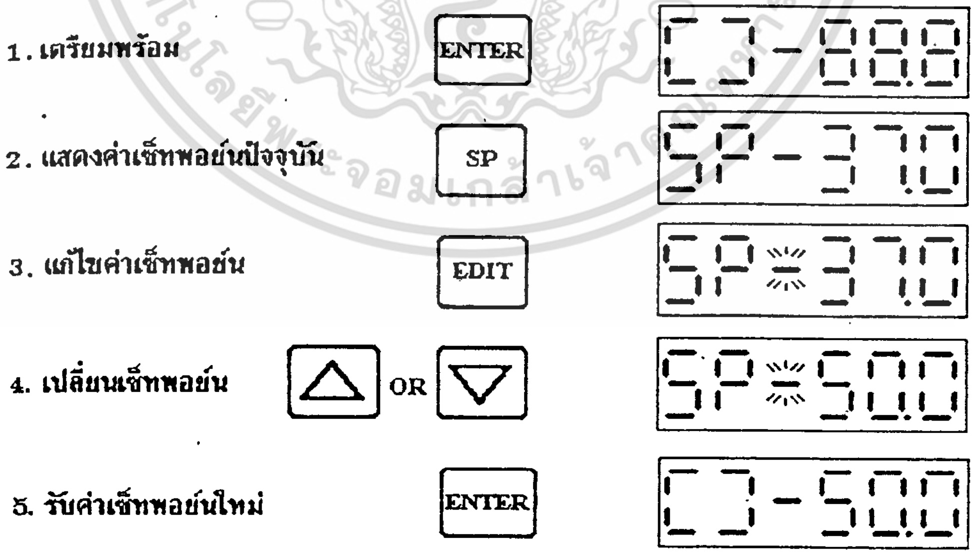
3. วิธีเซตค่าดีริเวทีฟเกน KD ค่าพารามิเตอร์ KD จะมีผลต่อการควบคุมแบบ PID เท่านั้น สมมุติว่าใช้การค่วบคุมแบบ P,PI หรือแบบ PLL ถึงแม้จะเซตค่าพารามิเตอร์ KD เป็นอย่างไรก็ตามก็จะไม่มีผลในการควบคุมเหล่านั้น ค่า KD มีหน่วยเป็น วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



5.2.3 การเซตค่าพอยน์

เราสามารถกำหนดเซตพอยน์ให้กับตัวดิจิทัลคอนโทรลเลอร์ได้ ค่าเซตพอยน์นั้นจะต้องใส่ค่าเป็นเปอร์เซ็นต์ ตั้งแต่ 0 เปอร์เซ็นต์ ถึง 100 เปอร์เซ็นต์ ตามรูปแบบการกดคีย์ดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ตามตัวอย่างเป็นการเปลี่ยนค่าเซตพอยน์จาก 37 เปอร์เซ็นต์เป็น 50 เปอร์เซ็นต์ เป็นต้น

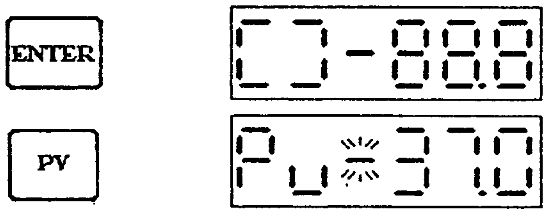
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.4 แสดงค่า process variable (PV)

โดยปกติเมื่อกระบวนการดำเนินไปอย่างต่อเนื่อง เราสามารถจะดูค่า PV ของกระบวนการได้อย่างต่อเนื่องเช่นกัน โดยกดคีย์ตามดังนี้

1. เตรียมพร้อม

2. แสดงค่า PV



อย่างไรก็ดีค่า PV ที่แสดงบนจอแสดงผลนั้นจะถูกปรับทันกาล(update) ประมาณทุก ๆ 1 วินาที โดยสังเกตจากจังหวะการกระพริบของเคอร์เซอร์ ค่า PV ที่แสดงออกมาบนจอแสดงผลนั้นจะเป็นหน่วยเปอร์เซ็นต์ เช่น ตามตัวอย่าง 37.0 เท่ากับ 37.0 เปอร์เซ็นต์ ดังนั้นการแสดงค่า PV จึงเป็นไปได้อยู่ในช่วง 0 เปอร์เซ็นต์ ถึง 100 เปอร์เซ็นต์ ถ้าเคอร์เซอร์ไม่กระพริบในตอนนี้ แสดงว่ามีเออเรอร์เกิดขึ้น ซึ่งต้องรีเซตเครื่องใหม่

5.2.5 การเซตอัตราบอด(Baud Rate)

อัตราบอด (baud rate) เป็นอัตราการส่งข้อมูลแบบอนุกรมแบบ RS232 ซึ่งจำเป็นสำหรับตัวดิจิทัลคอนโทรลเลอร์ต้องติดต่อสื่อสารกับโฮสคอมพิวเตอร์ ซึ่งการเซตค่าบอดกำหนดให้ผู้ใช้เซตค่าได้ ตั้งแต่ 150, 300, 600, 1200, 2400, 4800, 9600 บอด ดังนั้นการเซตอัตราบอดต้องเหมือนกับอัตราบอดของโฮส

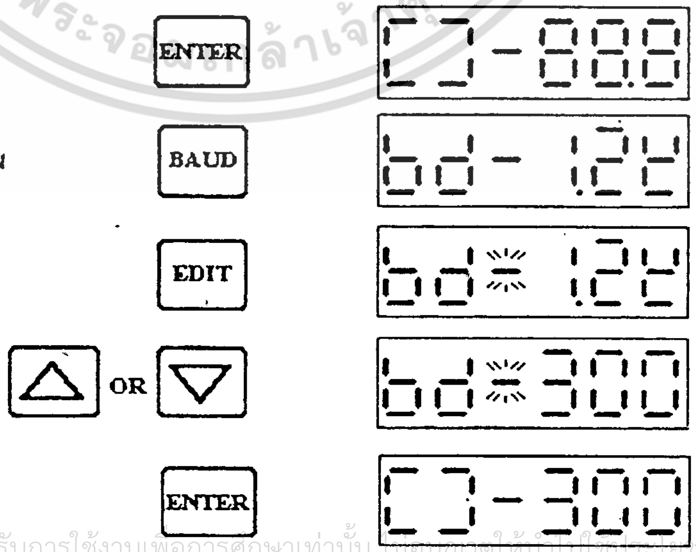
1. เตรียมพร้อม

2. แสดงอัตรบอดปัจจุบัน

3. แก้ไขอัตราบอด

4. เปลี่ยนอัตราบอด

5. รับอัตราบอดใหม่



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตามตัวอย่างการกดคีย์ข้างต้น ลำดับที่ 2 ค่าที่ใช้ให้ปัจจุบันคือ 1200 บอด หรือ 1.2K บอด โดยเปลี่ยนแปลงใหม่เป็นอัตราบอด 300 บอด แล้วลำดับที่ 5 กดคีย์ ENTER รับอัตราบอดใหม่ไปใช้

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

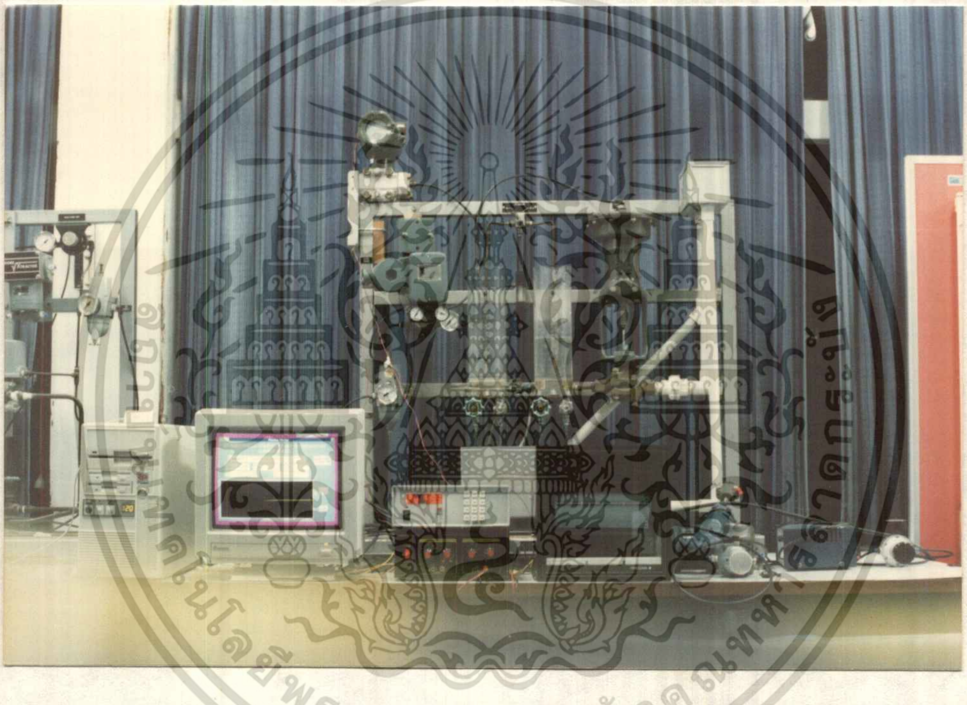
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การทดสอบตัวดิจิทัลคอนโทรลเลอร์

6.1 ลักษณะโดยทั่วไปของตัวคอนโทรลเลอร์

รูป 6.1 เป็นภาพแสดงการติดตั้งตัวคอนโทรลเลอร์และโปรแกรมมอนิเตอร์เข้ากับกระบวนการควบคุมระดับอันดับที่สองในห้องปฏิบัติการ

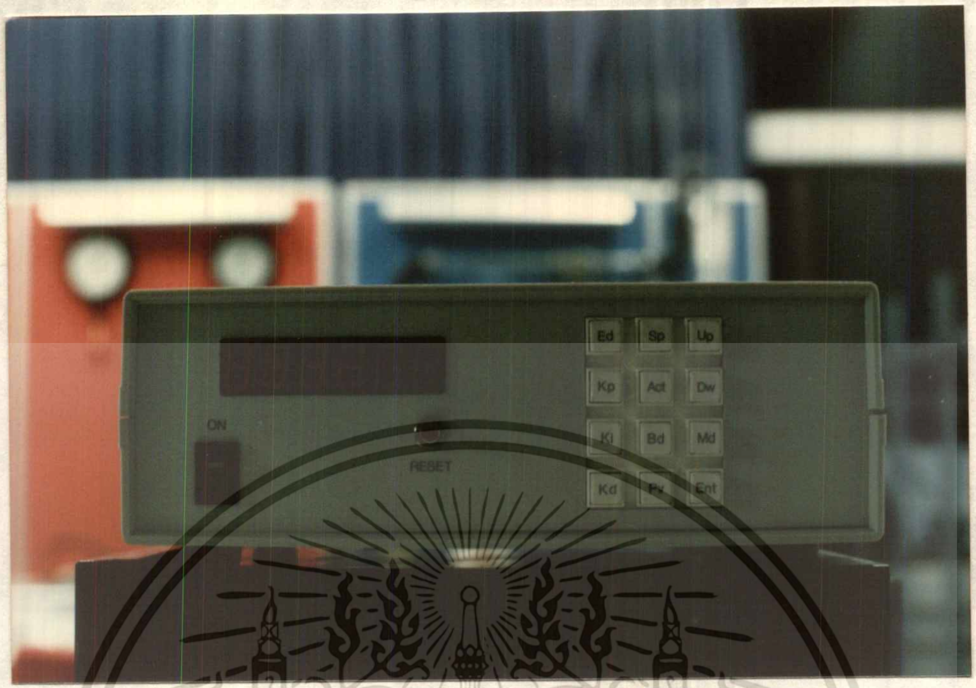


รูปที่ 6-1 *แสดงการติดตั้งตัวคอนโทรลเลอร์กับกระบวนการควบคุมระดับ*

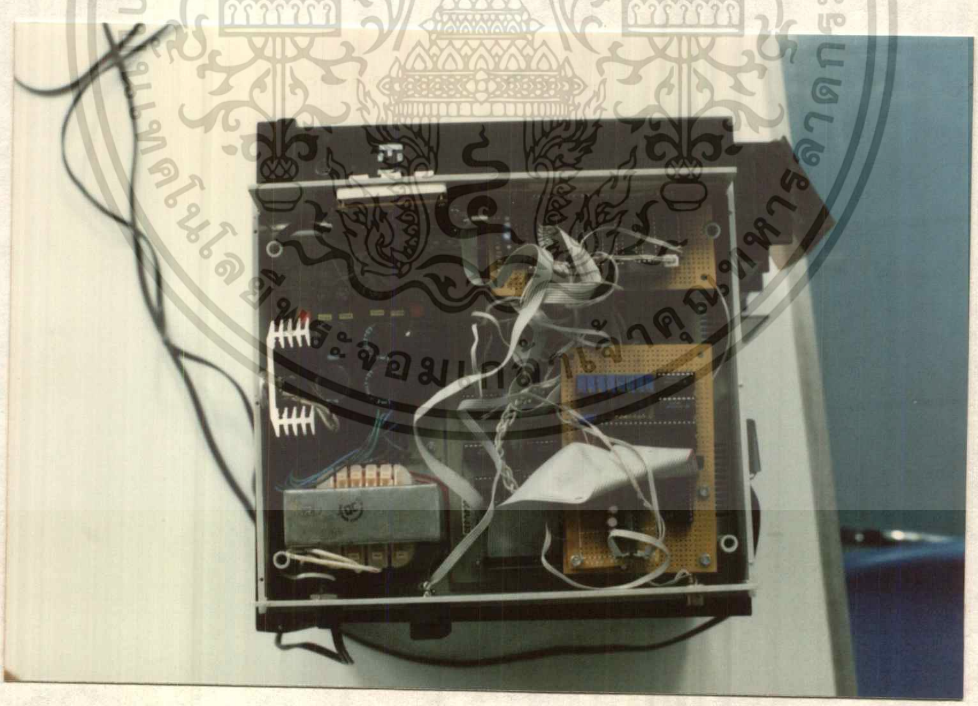
ด้านหน้าของตัวดิจิทัลคอนโทรลเลอร์จะมีหน้าตาดังรูป 6-2 จะประกอบด้วยสวิทช์เปิดปิด ,จอแสดงผลเป็นแอลอีดี 7 ส่วน(LED 7 segment)จำนวน 6 หลัก,แผงคีย์บอร์ดสำหรับป้อนค่าต่างๆ และสวิทช์รีเซ็ตวงจร

แสดงลักษณะภายในของดิจิทัลคอนโทรลเลอร์ตามรูป 6-3 จะประกอบด้วยแผงวงจรหลายส่วนด้วยกัน แผงวงจรที่อยู่ด้านล่างสุดคือ บอร์ดไมโครโปรเซสเซอร์เบอร์ 8031 แผงวงจรตามแนวตั้งติดด้านหน้าสุดคือ แผงวงจรควบคุมคีย์บอร์ด ต่อมา แผงควบคุมแสดงผล แผงอินพุตเอาต์พุต และแผงของวงจรเฟสล็อก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-2 *แสดงด้านหน้าของตัวดิจิทัลคอนโทรลเลอร์ต้นแบบ*

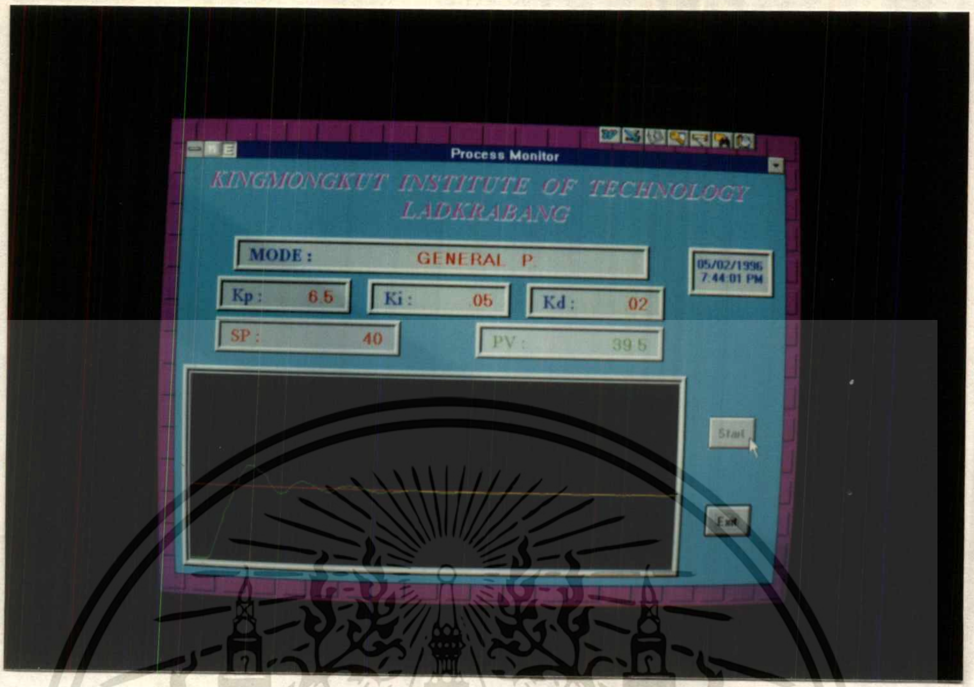


รูปที่ 6-3 *แสดงอุปกรณ์ภายในของตัวดิจิทัลคอนโทรลเลอร์ต้นแบบ*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 ผลการทดสอบดิจิทัลคอนโทรลเลอร์ในโหมดต่าง ๆ

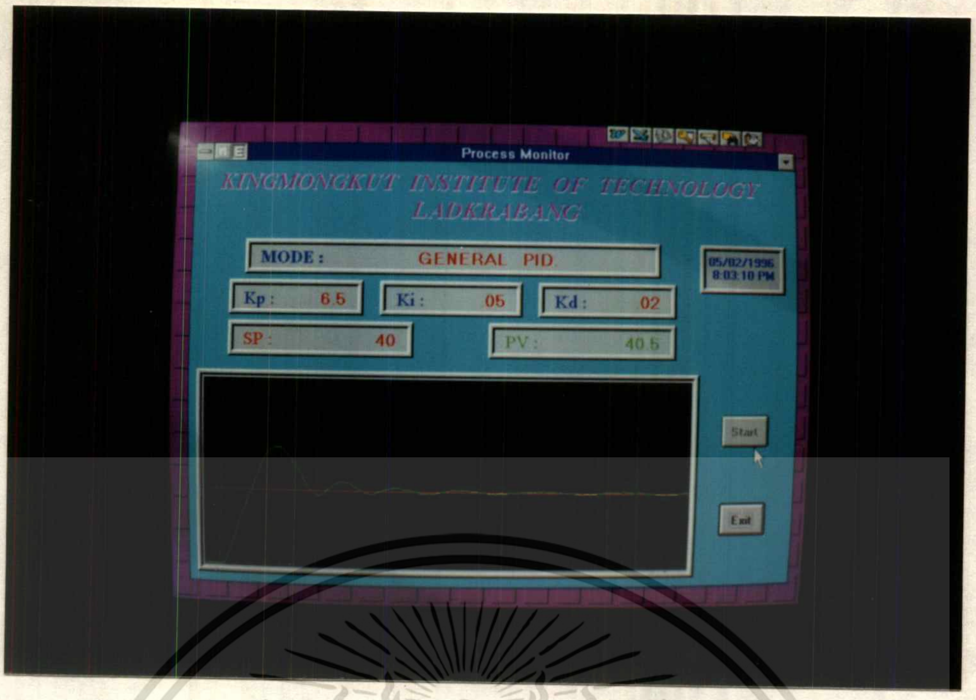
6.2.1 ทดสอบในโหมดการควบคุมแบบเฟสล็อก



รูปที่ 6-4 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5*

รูปที่ 6-5 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5 Ki = 0.05*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

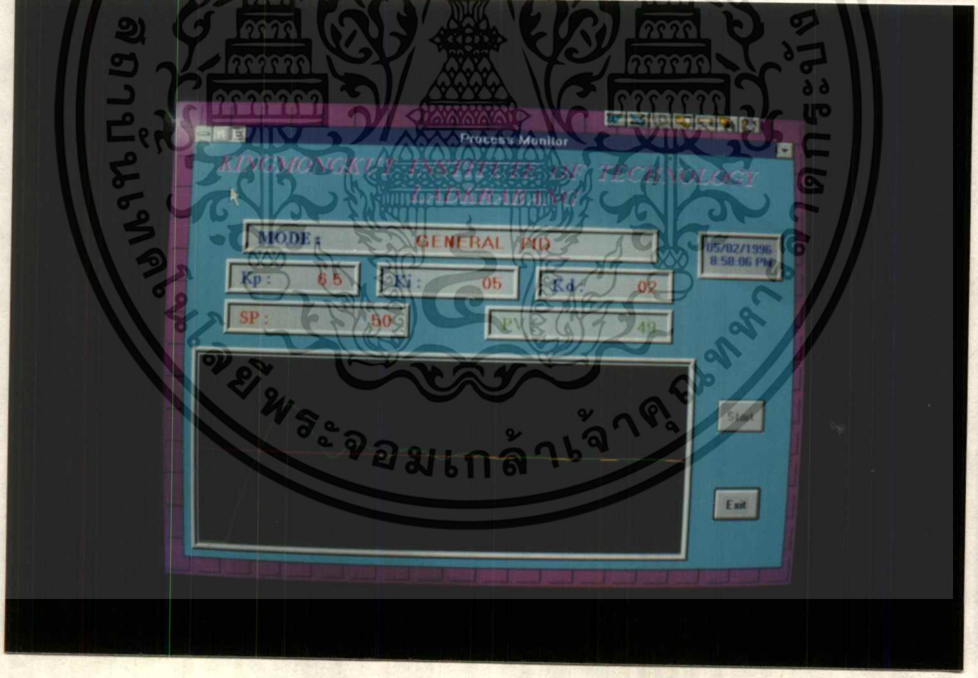


รูปที่ 6-6 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5, Ki = 0.05, Kd = 0.02*

รูปที่ 6-7 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

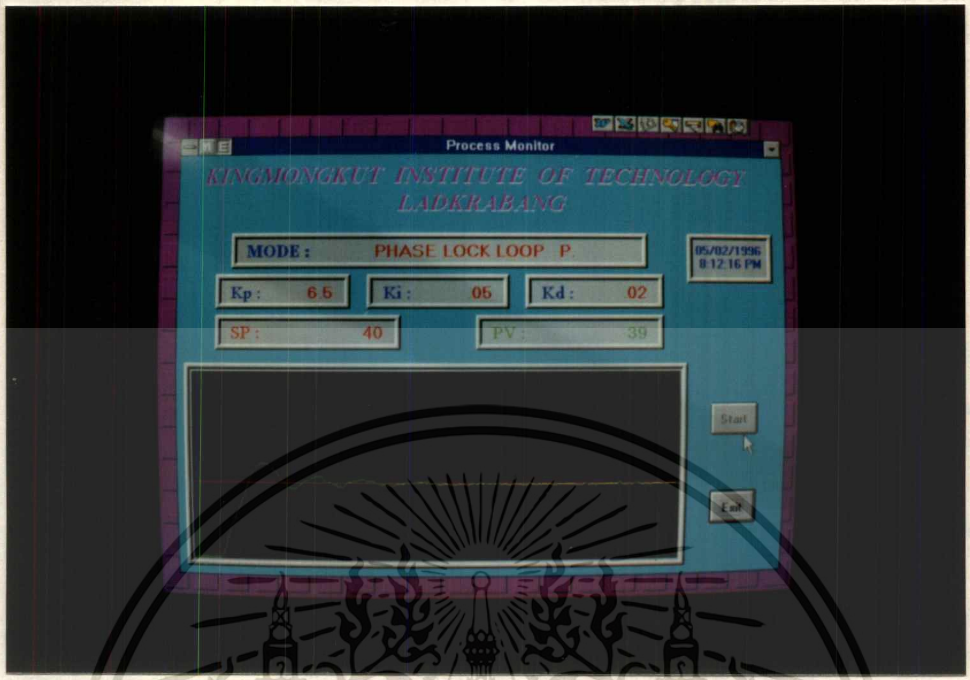
รูปที่ 6-8 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5 Ki = 0.05*



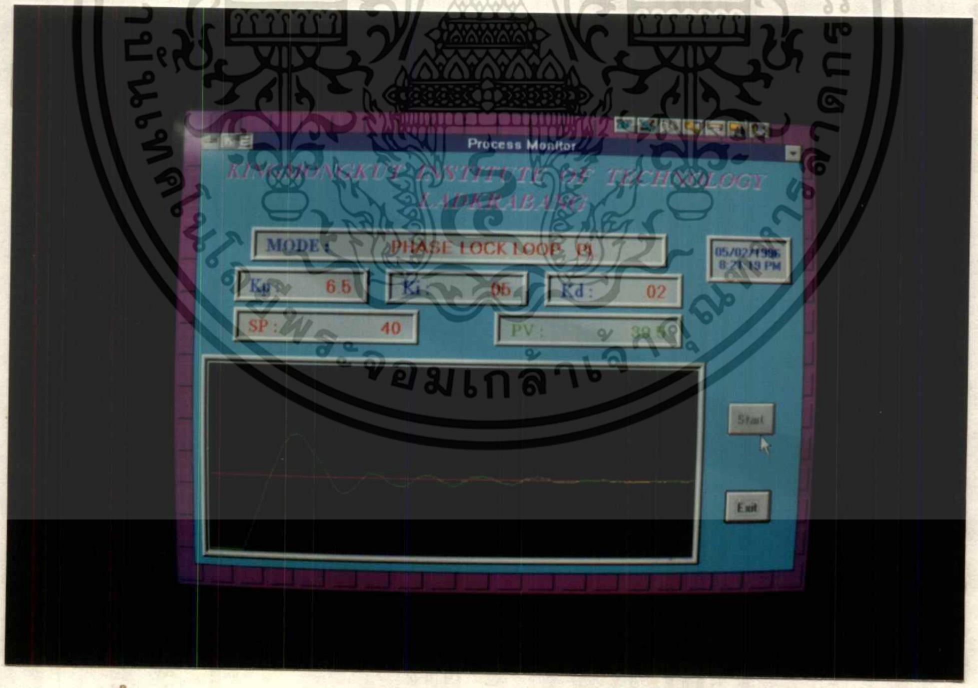
รูปที่ 6-9 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5, Ki = 0.05, Kd = 0.02*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2.2 ทดสอบในโหมดการควบคุมแบบเฟสล็อก

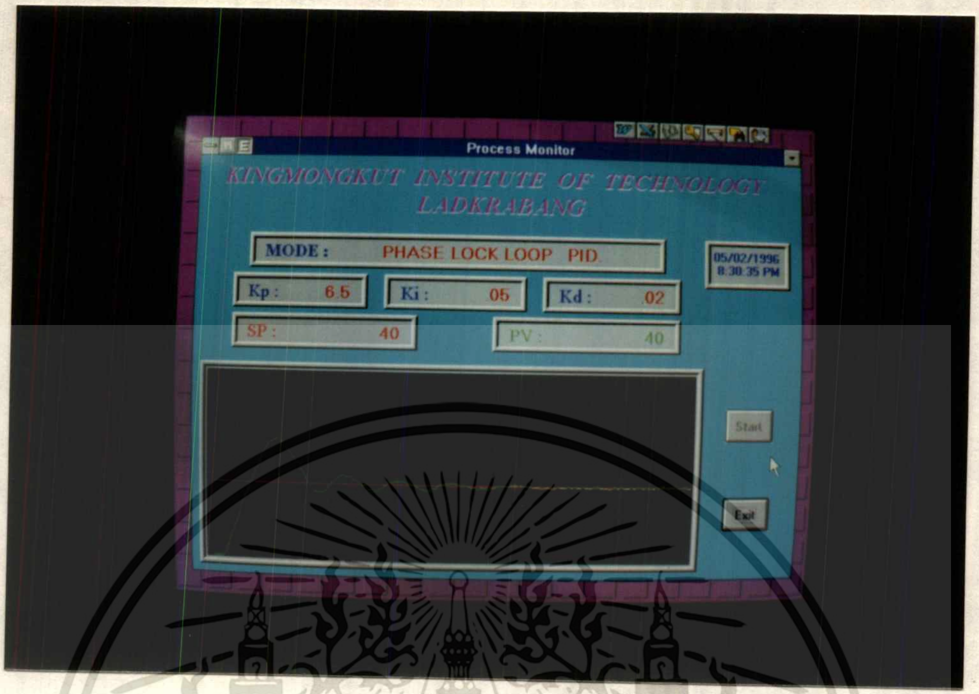


รูปที่ 6-10 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5*

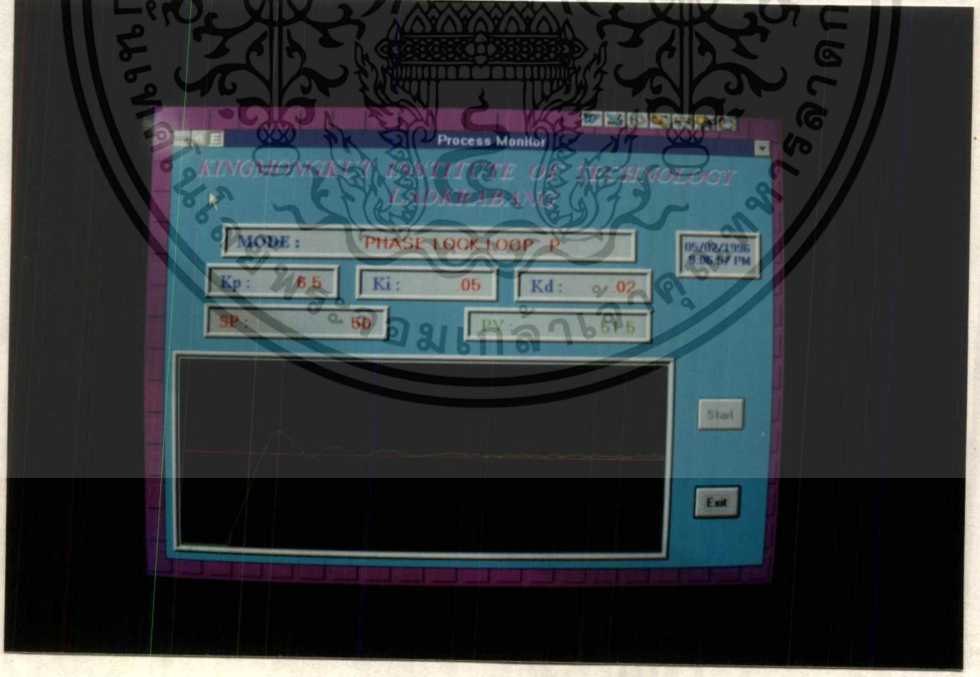


รูปที่ 6-11 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5 Ki = 0.05*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

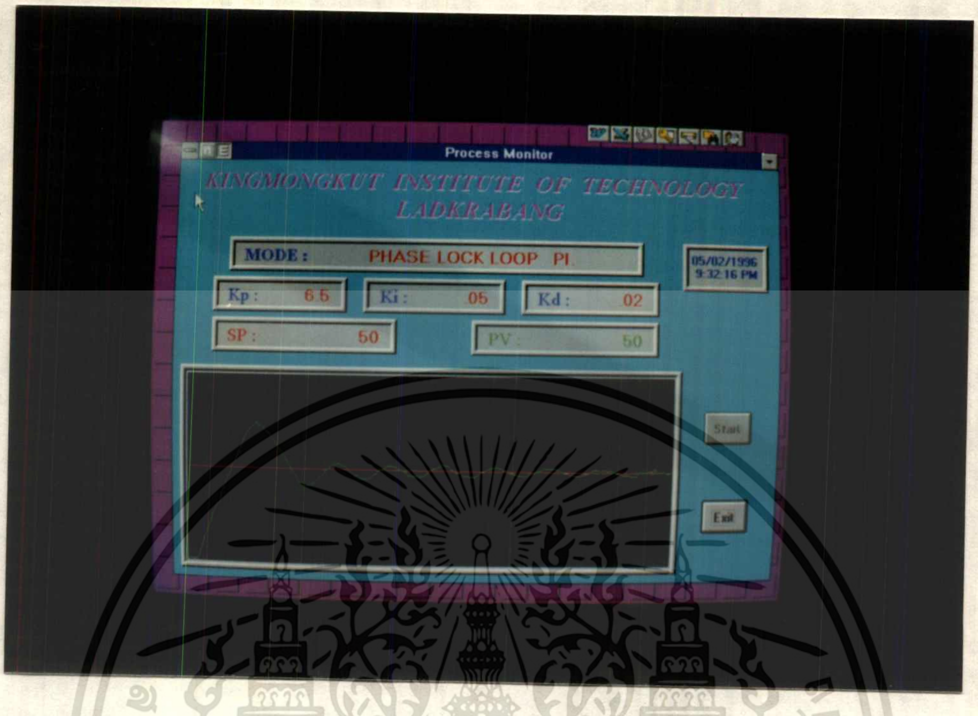


รูปที่ 6-12 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 40% ค่า Kp = 6.5, Ki = 0.05, Kd = 0.02*

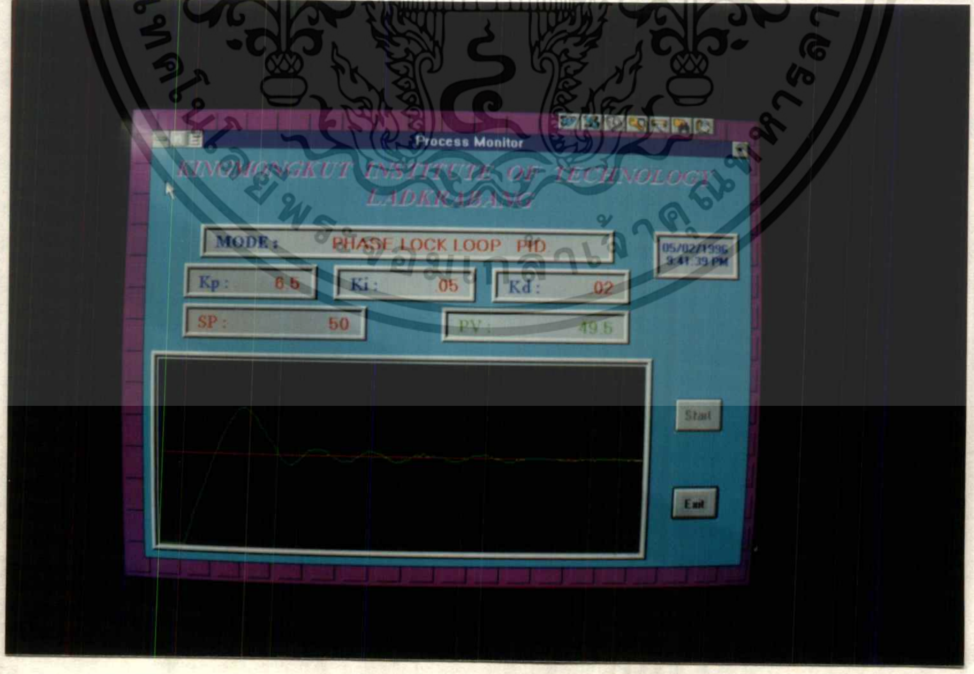


รูปที่ 6-13 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



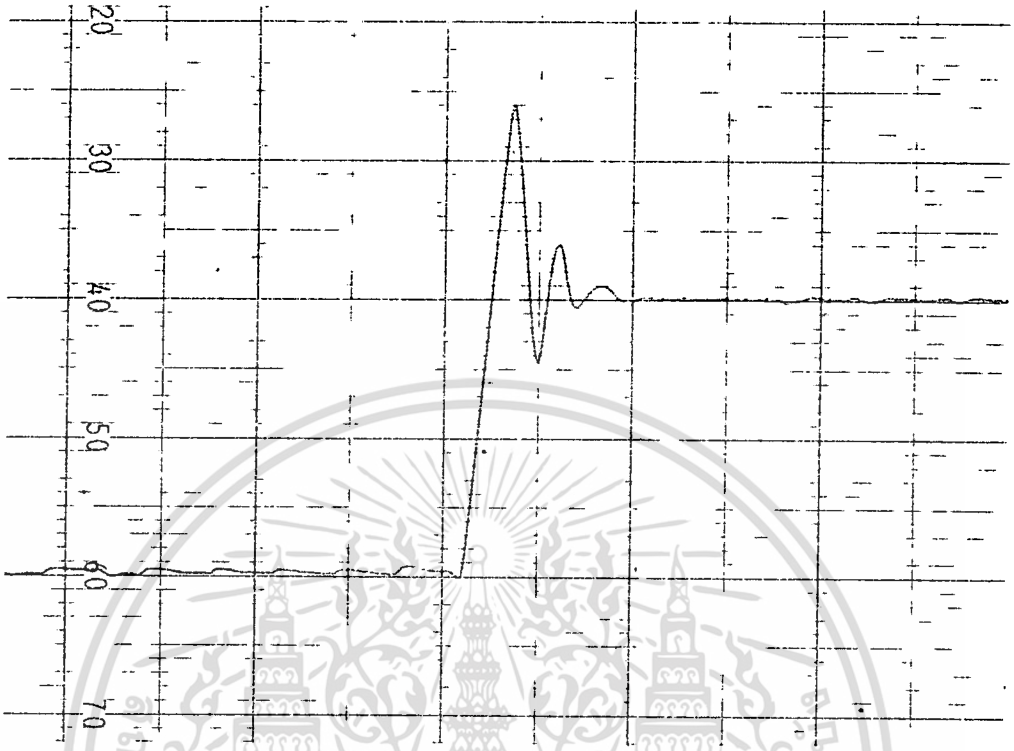
รูปที่ 6-14 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5 Ki = 0.05*



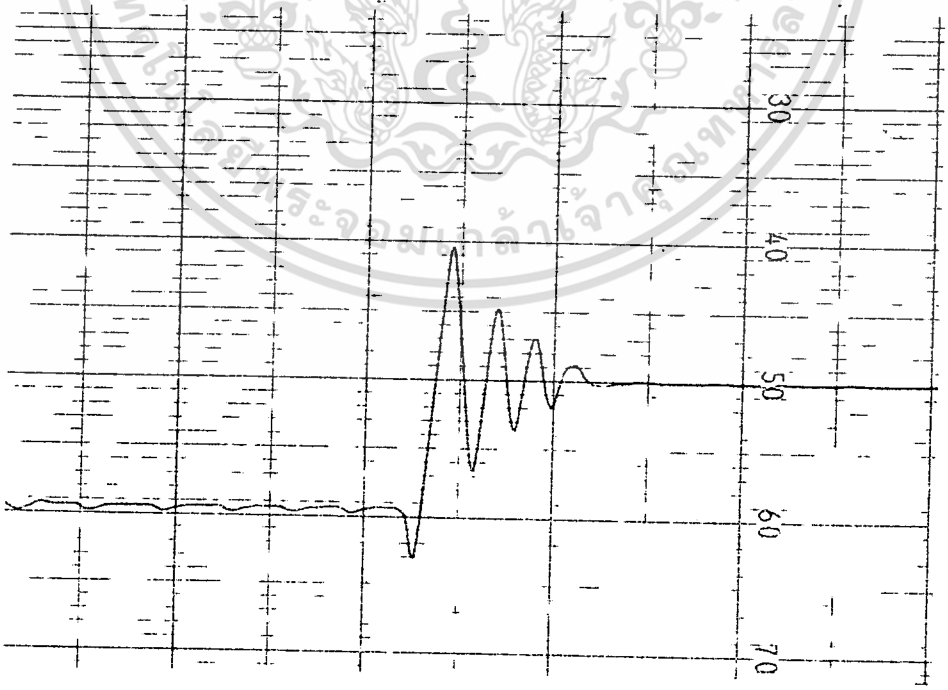
รูปที่ 6-15 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 50% ค่า Kp = 6.5, Ki = 0.05, Kd = 0.02*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2.3 กราฟแสดงลักษณะสัญญาณการควบคุมในโหมด PI

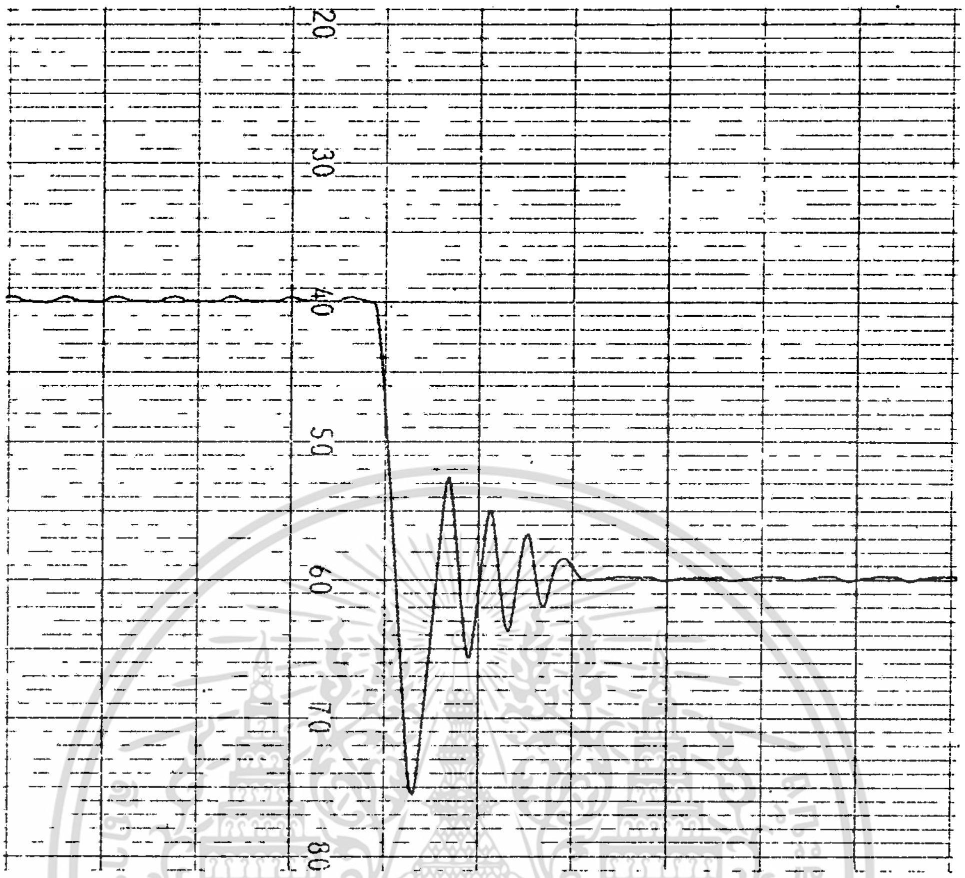


รูปที่ 6-16 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 60% เป็น 40% ค่า Kp = 6.5 Ki = 0.07*



รูปที่ 6-17 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 60% เป็น 50% ค่า Kp = 6.5 Ki = 0.05*

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-18 *การตอบสนองสเต็บขึ้นที่ค่าเซตพอยน์ 40% เป็น 60% ค่า Kp = 6.5 Ki = 0.04*

6.2 สรุปผลการทดลอง

จากผลการทดลองที่ได้ ในการทดลองเปลี่ยนโหมดในการควบคุม รวมถึงการเปลี่ยนค่าเกน การควบคุมระบบที่ค่าต่าง ๆ จะได้ผลตอบสนองการควบคุมระบบต่าง ๆ กันดังรูปผลการทดลอง ซึ่งจะเห็นได้ว่า ในโหมดการควบคุมแบบ PI นั้น การตอบสนองของระบบ จะมีเสถียรภาพดีเนื่องจากว่าระบบที่ใช้ในการทดลองนั้น เป็นการควบคุมระดับน้ำในระบบ ระดับ 2 (Second Order) ซึ่งการเปลี่ยนแปลงของระบบนั้นเป็นไปอย่างช้า ๆ ไม่รวดเร็ว ดังนั้นในส่วนของค่าดีริเวทีฟเกน ซึ่งเป็นค่าที่ใช้ในการหน่วงเวลา การเปลี่ยนแปลงของระบบ จึงไม่จำเป็นต้องใช้ในการควบคุมกระบวนการนี้ ซึ่งจะเห็นความแตกต่างได้จากภาพแสดงการควบคุมในโหมดต่างๆ

6.3 ข้อเสนอแนะและข้อคิดเห็น

เนื่องจากบทความนี้ได้เน้นการใช้พล้านเป็นแบบดีเลย์ออร์เดอร์หนึ่งและสอง ซึ่งเป็นพื้นฐานกระบวนการโดยทั่วไป อย่างเช่น การควบคุมระดับเป็นต้น โดยเน้นเนื้อหาการเปรียบเทียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระหว่างโหมดควบคุมแบบพรอปพรอชันแนล-อินทีกรัล กับโหมดเฟสล็อก โดยควบคุมขบวนการข้างต้น

อย่างไรก็ดีในพล้านมิใช่จะเป็นแบบดีเลย์ออร์เดอร์หนึ่งหรือสองเท่านั้น ซึ่งถ้าหากเป็นออร์เดอร์สูงกว่านี้จะเป็นอย่างไร ในกรณีดังกล่าวถ้าเป็น ออร์เดอร์รวม จำเป็นต้องการควบคุมแบบ PID หรือออร์เดอร์สูงกว่า ต้องมีการเพิ่มวงจรชดเชย เป็นต้น ทำให้เกินความจำเป็นสำหรับหัวข้อที่จะศึกษาเกี่ยวกับเรื่องนี้

ท้ายสุดหวังบอย่างยิ่งว่า ผู้อ่านคงจะได้ความกระจ่างเกี่ยวกับการศึกษาทางด้านการควบคุมโดยเฉพาะอาย่างยิ่งการควบคุมในโหมดเฟสล็อก ถ้ามีข้อสงสัย หรือติชม ยินดีน้อมรับด้วยความเต็มใจเป็นอย่างสูง สวัสดี

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

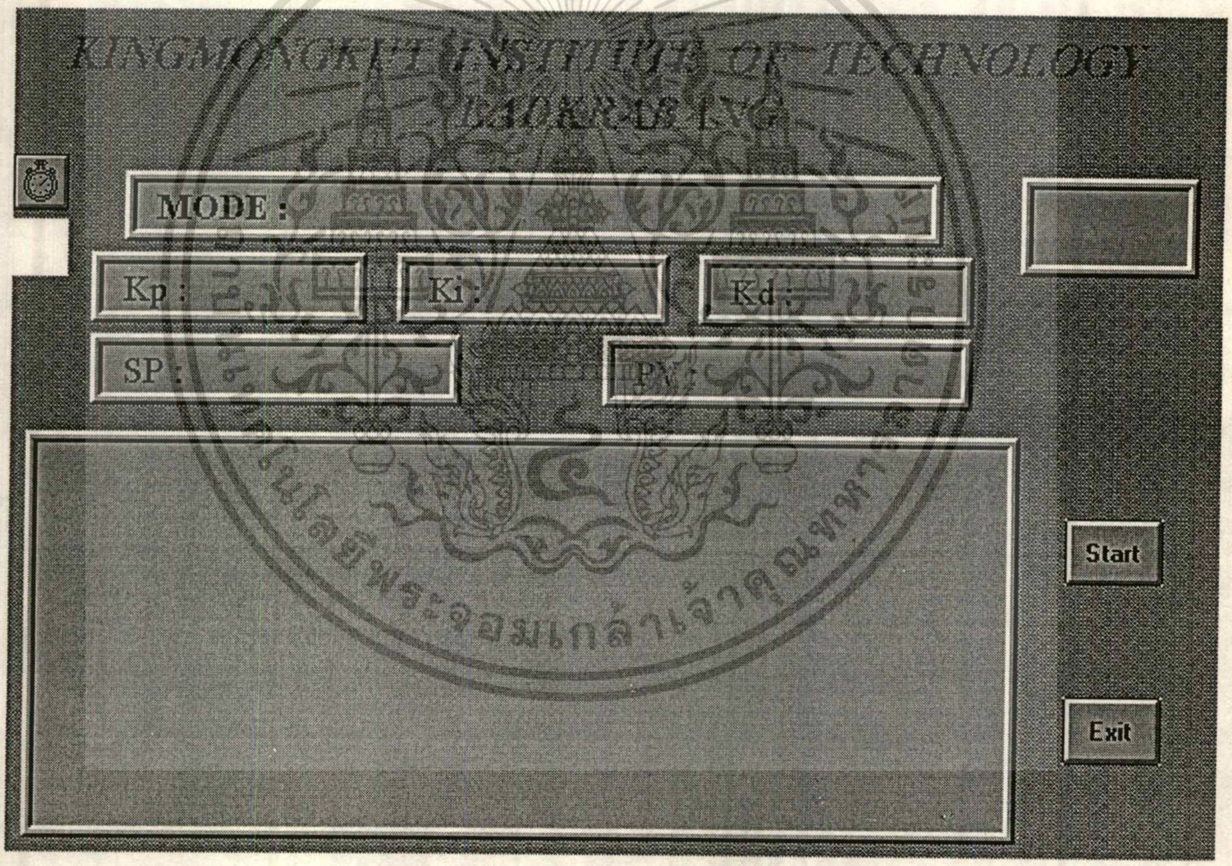
1. Dr. Roland E.Best:"PHASE-LOCKED LOOPS Theory, Design and Application.",Copyright 1984, McGraw Hill, 1984.
2. National Semiconductor "Linear databook".
3. R.C. Holland. "Microcomputers for process control ".Pergamon Press.
4. Garth Nash "Phased-Locked loop Design Fundamentals" Application Note AN-535, MOTOROLA Semiconductor Products Inc.
5. Izawa ,K. : Introduction to Automatic Control , Elsevier, New York, 1663.
6. Mr. Jetanar Singsin "Phase Locked and Conventional Mode Digital Controller for Industrial Plant."

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

ซอฟต์แวร์ลิสต์โปรแกรมมอนิเตอร์

ซอฟต์แวร์ลิสต์ ของโปรแกรมพล้านต์มอนิเตอริง ซึ่งใช้กับโฮสคอมพิวเตอร์เพื่อสำหรับดูแลเหตุการณ์ต่าง ๆ ที่เกิดขึ้นในพล้านต์ โดยรับข้อมูลมาจากตัวดิจิทัลคอนโทรลเลอร์ ดังกล่าวในเนื้อเรื่อง ซึ่งสามารถเรียกใช้โปรแกรมนี้ได้บนระบบปฏิบัติการ Microsoft Window โดยเรียกใช้ไฟล์โปรแกรม Project2.exe



รูปที่ A *ภาพแสดงโปรแกรมมอนิเตอร์*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
VERSION 2.00
Begin Form Form1
   BackColor       =   &H00C0C000&
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Process Monitor"
   ClientHeight    =   6240
   ClientLeft      =   690
   ClientTop       =   1815
   ClientWidth     =   9075
   ForeColor       =   &H0000FFFF&
   Height          =   6645
   Left            =   630
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   ScaleHeight     =   416
   ScaleMode       =   3  'Pixel
   ScaleWidth      =   605
   Top             =   1470
   Width           =   9195
   Begin PictureBox Pic1
      BackColor       =   &H00000000&
      Height          =   2775
      Left            =   240
      ScaleHeight     =   201
      ScaleMode       =   0  'User
      ScaleWidth      =   500
      TabIndex        =   4
      Top             =   3240
      Width           =   7215
   End
   Begin SSPanel Panel7
      BackColor       =   &H00C0C0C0&
      BevelInner      =   1  'Inset
      BevelWidth      =   2
      BorderWidth     =   2
      Font3D          =   0  'None
      ForeColor       =   &H00FF0000&
      Height          =   3015
      Left            =   120
      ShadowColor     =   1  'Black
      TabIndex        =   9
      Top             =   3120
      Width           =   7455
   End
   Begin SSPanel Panel1
      Alignment       =   4  'Right Justify - MIDDLE
      BackColor       =   &H00C0C0C0&
      BevelInner      =   1  'Inset
      BevelWidth      =   2
      BorderWidth     =   2
      Font3D          =   0  'None
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "Times New Roman"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น ทั้งนี้ห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      FontSize          =   18
      FontStrikethru    =   0   'False
      FontUnderline     =   0   'False
      ForeColor         =   &H0000FF00&
      Height            =   525
      Left              =   600
      ShadowColor       =   1  'Black
      TabIndex          =   16
      Top               =   2400
      Width             =   2775
      Begin Label FLabel1
         BackColor         =   &H00C0C0C0&
         Caption           =   "SP :"
         FontBold          =   0   'False
         FontItalic        =   0   'False
         FontName          =   "Times New Roman"
         FontSize          =   13.5
         FontStrikethru    =   0   'False
         FontUnderline     =   0   'False
         ForeColor         =   &H000000FF&
         Height            =   300
         Left              =   240
         TabIndex          =   17
         Top               =   120
         Width             =   735
      End
      Begin Label Label1
         Alignment         =   1  'Right Justify
         BackColor         =   &H00C0C0C0&
         FontBold          =   -1  'True
         FontItalic        =   0   'False
         FontName          =   "MS Sans Serif"
         FontSize          =   12
         FontStrikethru    =   0   'False
         FontUnderline     =   0   'False
         ForeColor         =   &H000000FF&
         Height            =   300
         Left              =   1320
         TabIndex          =   18
         Top               =   120
         Width             =   1215
      End
   End
   Begin SSPanel Panel5
      Alignment         =   4  'Right Justify - MIDDLE
      BackColor         =   &H00C0C0C0&
      BevelInner        =   1  'Inset
      BevelWidth        =   2
      BorderWidth       =   2
      Font3D            =   0  'None
      FontBold          =   -1  'True
      FontItalic        =   0   'False
      FontName          =   "Times New Roman"
      FontSize          =   18
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        FontStrikethru   =   0   'False
        FontUnderline    =   0   'False
        ForeColor        =   &H000000FF&
        Height           =   525
        Left             =   2880
        ShadowColor      =   1  'Black
        TabIndex         =   23
        Top              =   1800
        Width            =   2055
        Begin Label Label5
           Alignment        =   1  'Right Justify
           BackColor        =   &H00C0C0C0&
           FontBold         =   -1  'True
           FontItalic       =   0   'False
           FontName         =   "MS Sans Serif"
           FontSize         =   12
           FontStrikethru   =   0   'False
           FontUnderline    =   0   'False
           ForeColor        =   &H000000FF&
           Height           =   300
           Left             =   1080
           TabIndex         =   19
           Top              =   120
           Width            =   735
        End
        Begin Label FLabel5
           BackColor        =   &H00C0C0C0&
           Caption          =   "Ki  :  "
           FontBold         =   0   'False
           FontItalic       =   0   'False
           FontName         =   "Times New Roman"
           FontSize         =   13.5
           FontStrikethru   =   0   'False
           FontUnderline    =   0   'False
           ForeColor        =   &H00FF0000&
           Height           =   300
           Left             =   240
           TabIndex         =   3
           Top              =   120
           Width            =   735
        End
     End
     Begin SSPanel Panel6
        Alignment        =   4  'Right Justify - MIDDLE
        BackColor        =   &H00C0C0C0&
        BevelInner       =   1  'Inset
        BevelWidth       =   2
        BorderWidth      =   2
        Font3D           =   0  'None
        FontBold         =   -1  'True
        FontItalic       =   0   'False
        FontName         =   "Times New Roman"
        FontSize         =   18
        FontStrikethru   =   0   'False
```

เอกสารนี้เป็นเอกสารที่ใช้ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น ลูกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        FontUnderline   =   0   'False
        ForeColor       =   &H000000FF&
        Height          =   525
        Left            =   5160
        ShadowColor     =   1  'Black
        TabIndex        =   20
        Top             =   1800
        Width           =   2055
        Begin Label FLabel6
           BackColor       =   &H00C0C0C0&
           Caption         =   "Kd :"
           FontBold        =   0   'False
           FontItalic      =   0   'False
           FontName        =   "Times New Roman"
           FontSize        =   13.5
           FontStrikethru  =   0   'False
           FontUnderline   =   0   'False
           ForeColor       =   &H00FF0000&
           Height          =   300
           Left            =   240
           TabIndex        =   22
           Top             =   120
           Width           =   735
        End
        Begin Label Label6
           Alignment       =   1  'Right Justify
           BackColor       =   &H00C0C0C0&
           FontBold        =   -1  'True
           FontItalic      =   0   'False
           FontName        =   "MS Sans Serif"
           FontSize        =   12
           FontStrikethru  =   0   'False
           FontUnderline   =   0   'False
           ForeColor       =   &H000000FF&
           Height          =   300
           Left            =   1080
           TabIndex        =   21
           Top             =   120
           Width           =   735
        End
     End
     Begin SSPanel Panel3
        Alignment       =   4  'Right Justify - MIDDLE
        BackColor       =   &H00C0C0C0&
        BevelInner      =   1  'Inset
        BevelWidth      =   2
        BorderWidth     =   2
        Font3D          =   0  'None
        FontBold        =   -1  'True
        FontItalic      =   0   'False
        FontName        =   "Times New Roman"
        FontSize        =   18
        FontStrikethru  =   0   'False
        FontUnderline   =   0   'False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      ForeColor       =   &H000000FF&
      Height          =   525
      Left            =   840
      ShadowColor     =   1  'Black
      TabIndex        =   13
      Top             =   1200
      Width           =   6135
      Begin Label FLabel3
         BackColor       =   &H00C0C0C0&
         Caption         =   "MODE : "
         FontBold        =   -1  'True
         FontItalic      =   0   'False
         FontName        =   "Times New Roman"
         FontSize        =   13.5
         FontStrikethru  =   0   'False
         FontUnderline   =   0   'False
         ForeColor       =   &H00FF0000&
         Height          =   300
         Left            =   240
         TabIndex        =   15
         Top             =   120
         Width           =   1455
      End
      Begin Label Label3
         Alignment       =   2  'Center
         BackColor       =   &H00C0C0C0&
         FontBold        =   -1  'True
         FontItalic      =   0   'False
         FontName        =   "MS Sans Serif"
         FontSize        =   12
         FontStrikethru  =   0   'False
         FontUnderline   =   0   'False
         ForeColor       =   &H000000FF&
         Height          =   300
         Left            =   1320
         TabIndex        =   14
         Top             =   120
         Width           =   4575
      End
   End
   Begin SSPanel Panel4
      Alignment       =   4  'Right Justify - MIDDLE
      BackColor       =   &H00C0C0C0&
      BevelInner      =   1  'Inset
      BevelWidth      =   2
      BorderWidth     =   2
      Font3D          =   0  'None
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "Times New Roman"
      FontSize        =   18
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H000000FF&
```

เอกสารนี้เป็นเอกสารที่จัดทำไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Height           =   525
        Left             =   600
        ShadowColor      =   1  'Black
        TabIndex         =   10
        Top              =   1800
        Width            =   2055
        Begin Label FLabel4
           BackColor        =   &H00C0C0C0&
           Caption          =   "Kp : "
           FontBold         =   0   'False
           FontItalic       =   0   'False
           FontName         =   "Times New Roman"
           FontSize         =   13.5
           FontStrikethru   =   0   'False
           FontUnderline    =   0   'False
           ForeColor        =   &H00FF0000&
           Height           =   300
           Left             =   240
           TabIndex         =   12
           Top              =   120
           Width            =   735
        End
        Begin Label Label4
           Alignment        =   1  'Right Justify
           BackColor        =   &H00C0C0C0&
           FontBold         =   -1  'True
           FontItalic       =   0   'False
           FontName         =   "MS Sans Serif"
           FontSize         =   12
           FontStrikethru   =   0   'False
           FontUnderline    =   0   'False
           ForeColor        =   &H000000FF&
           Height           =   300
           Left             =   1080
           TabIndex         =   11
           Top              =   120
           Width            =   735
        End
     End
     Begin SSPanel Panel8
        BackColor        =   &H00C0C0C0&
        BevelInner       =   1  'Inset
        BevelWidth       =   2
        BorderWidth      =   2
        Font3D           =   0  'None
        ForeColor        =   &H00FF0000&
        Height           =   735
        Left             =   7560
        ShadowColor      =   1  'Black
        TabIndex         =   7
        Top              =   1200
        Width            =   1335
     End
     Begin SSPanel Panel2
```

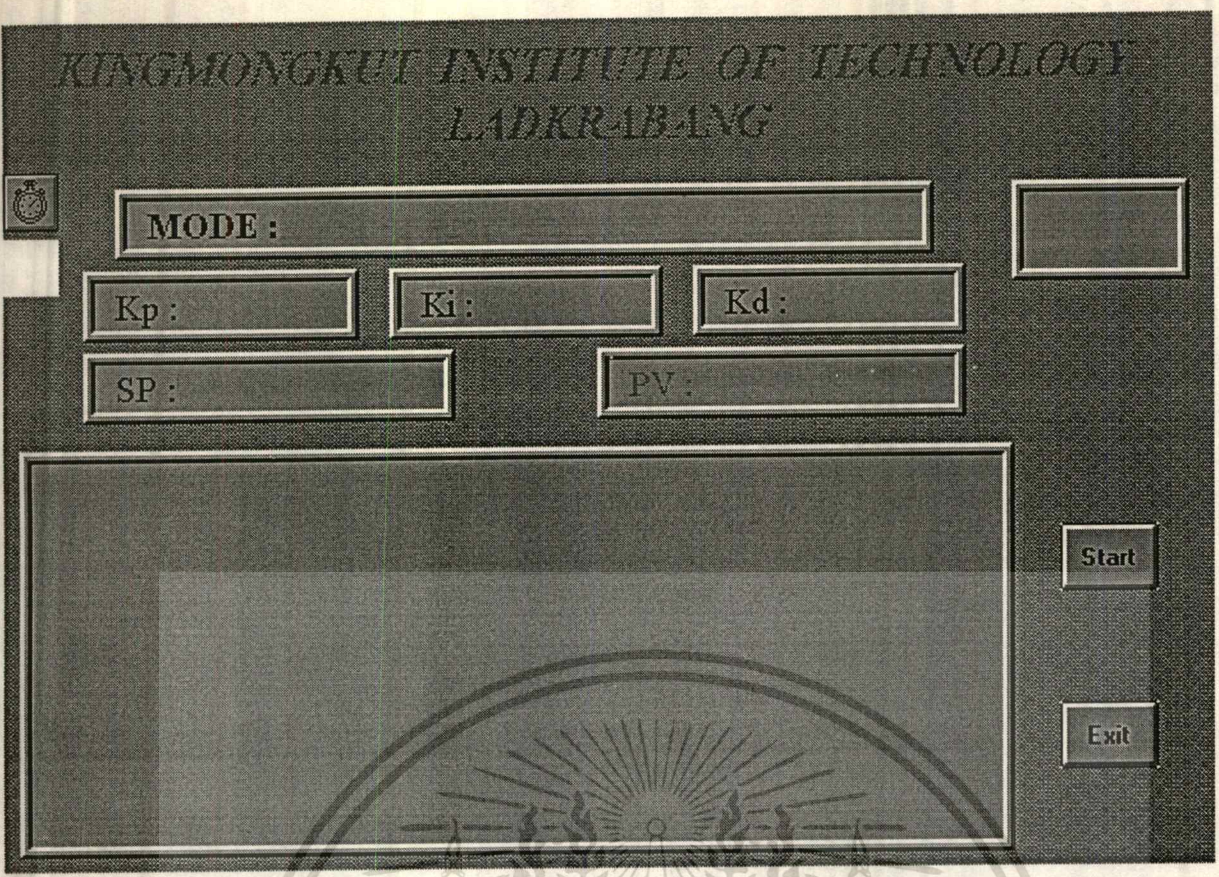
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Alignment        =   4  'Right Justify - MIDDLE
      BackColor        =   &H00C0C0C0&
      BevelInner       =   1  'Inset
      BevelWidth       =   2
      BorderWidth      =   2
      Font3D           =   0  'None
      FontBold         =   -1  'True
      FontItalic       =   0   'False
      FontName         =   "Times New Roman"
      FontSize         =   18
      FontStrikethru   =   0   'False
      FontUnderline    =   0   'False
      ForeColor        =   &H0000FF00&
      Height           =   525
      Left             =   4440
      ShadowColor      =   1  'Black
      TabIndex         =   5
      Top              =   2400
      Width            =   2775
      Begin Label Label2
         Alignment        =   1  'Right Justify
         BackColor        =   &H00C0C0C0&
         FontBold         =   -1  'True
         FontItalic       =   0   'False
         FontName         =   "MS Sans Serif"
         FontSize         =   12
         FontStrikethru   =   0   'False
         FontUnderline    =   0   'False
         ForeColor        =   &H0000FF00&
         Height           =   300
         Left             =   1320
         TabIndex         =   8
         Top              =   120
         Width            =   1215
      End
      Begin Label FLabel2
         BackColor        =   &H00C0C0C0&
         Caption          =   "PV :"
         FontBold         =   0   'False
         FontItalic       =   0   'False
         FontName         =   "Times New Roman"
         FontSize         =   13.5
         FontStrikethru   =   0   'False
         FontUnderline    =   0   'False
         ForeColor        =   &H0000FF00&
         Height           =   300
         Left             =   240
         TabIndex         =   6
         Top              =   120
         Width            =   735
      End
   End
   Begin MSComm Comm
      CommPort         =   2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Interval        =   1000
        Left            =   0
        Settings        =   "4800,n,8,1"
        Top             =   1560
    End
    Begin CommandButton BtnExit
        Caption         =   "Exit"
        Height          =   495
        Left            =   7920
        TabIndex        =   1
        Top             =   5040
        Width           =   735
    End
    Begin CommandButton BtnStart
        Caption         =   "Start"
        Height          =   495
        Left            =   7920
        TabIndex        =   0
        Top             =   3720
        Width           =   735
    End
    Begin Timer Timer1
        Interval        =   1000
        Left            =   0
        Top             =   1080
    End
    Begin Label FLabel7
        Alignment       =   2  'Center
        BackColor       =   &H00C0C000&
        Caption         =   "KINGMONGKUT  INSTITUTE  OF  TECHNOLOGY  LADKRABANG
        FontBold        =   -1  'True
        FontItalic      =   -1  'True
        FontName        =   "Times New Roman"
        FontSize        =   18
        FontStrikethru  =   0   'False
        FontUnderline   =   0   'False
        ForeColor       =   &H00FF00FF&
        Height          =   855
        Left            =   0
        TabIndex        =   2
        Top             =   120
        Width           =   9075
    End
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Program

```
Genaral ( )
Dim Start As Integer
Dim Index As Integer
Dim i, m, k, z, t As Integer
Dim InData As String
Dim DataPV(1000) As Single
Dim DispPV(1000) As Integer
Dim DataSP(1000) As Single
Dim DispSP(1000) As Integer
Dim Mode As String
Dim Kp, Ki, Kd As Single
Dim TMode(6) As String
Dim Table(110) As Single

Sub Assign ()
  TMode(1) = "GENERAL   P."
  TMode(2) = "GENERAL   PI."
  TMode(3) = "GENERAL   PID."
  TMode(4) = "PHASE LOCK LOOP   P."
  TMode(5) = "PHASE LOCK LOOP   PI."
  TMode(6) = "PHASE LOCK LOOP   PID."
  Table(1) = .01
  Table(2) = .02
  Table(3) = .03
  Table(4) = .04
  Table(5) = .05
  Table(6) = .06
  Table(7) = .07
  Table(8) = .08
  Table(9) = .09
  Table(10) = .1
  Table(11) = .12
  Table(12) = .14
  Table(13) = .16
  Table(14) = .18
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table(15) = .2

Table(16) = .22

Table(17) = .24

Table(18) = .28

Table(19) = .3

Table(20) = .35

Table(21) = .4

Table(22) = .45

Table(23) = .5

Table(24) = .55

Table(25) = .6

Table(26) = .7

Table(27) = .8

Table(28) = .9

Table(29) = 1

Table(30) = 1.1

Table(31) = 1.2

Table(32) = 1.3

Table(33) = 1.4

Table(34) = 1.5

Table(35) = 1.6

Table(36) = 1.7

Table(37) = 1.8

Table(38) = 1.9

Table(39) = 2

Table(40) = 2.2

Table(41) = 2.4

Table(42) = 2.6

Table(43) = 2.8

Table(44) = 3

Table(45) = 3.2

Table(46) = 3.4

Table(47) = 3.6

Table(48) = 3.8

Table(49) = 4

Table(50) = 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table(51) = 4.4

Table(52) = 4.6

Table(53) = 4.8

Table(54) = 5

Table(55) = 5.5

Table(56) = 6

Table(57) = 6.5

Table(58) = 7

Table(59) = 7.5

Table(60) = 8

Table(61) = 8.5

Table(62) = 9

Table(63) = 9.5

Table(64) = 10

Table(65) = 11

Table(66) = 12

Table(67) = 13

Table(68) = 14

Table(69) = 15

Table(70) = 16

Table(71) = 17

Table(72) = 18

Table(73) = 19

Table(74) = 20

Table(75) = 22

Table(76) = 24

Table(77) = 26

Table(78) = 28

Table(79) = 30

Table(80) = 32

Table(81) = 34

Table(82) = 36

Table(83) = 38

Table(84) = 40

Table(85) = 42

Table(86) = 44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Table(87) = 46
    Table(88) = 48
    Table(89) = 50
    Table(90) = 55
    Table(91) = 60
    Table(92) = 65
    Table(93) = 70
    Table(94) = 75
    Table(95) = 80
    Table(96) = 85
    Table(97) = 90
    Table(98) = 95
    Table(99) = 100
    Table(100) = 110
    Table(101) = 120
    Table(102) = 130
    Table(103) = 140
    Table(104) = 150
    Table(105) = 160
    Table(106) = 170
    Table(107) = 180
    Table(108) = 190
    Table(109) = 200
End Sub

Sub BtnExit_Click ()
    Comm.PortOpen = False
    End
End Sub

Sub BtnStart_Click ()
    Start = 1
    BtnStart.Enabled = False
    ReadData
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub Convert ()
   Dim S1, S2, S3, S4, S5, S6 As String
   Dim I1, I2, I3, I4, I5, I6 As Integer

   S1 = Mid(InData, 1, 1)
   S2 = Mid(InData, 2, 1)
   S3 = Mid(InData, 3, 1)
   S4 = Mid(InData, 4, 1)
   S5 = Mid(InData, 5, 1)
   S6 = Mid(InData, 6, 1)
   I1 = Asc(S1)
   I2 = Asc(S2)
   I3 = Asc(S3)
   I4 = Asc(S4)
   I5 = Asc(S5)
   I6 = Asc(S6)
   If I3 < 0 Then I6 = 0
   If I3 > 5 Then I6 = 5
   If I4 < 0 Then I6 = 0
   If I4 > 109 Then I6 = 109
   If I5 < 0 Then I6 = 0
   If I5 > 109 Then I6 = 109
   If I6 < 0 Then I6 = 0
   If I6 > 109 Then I6 = 109
   DataSP(Index) = I1
   DataPV(Index) = I2
   Mode = TMode(I3 + 1)
   Kp = Table(I4 + 1)
   Ki = Table(I5 + 1)
   Kd = Table(I6 + 1)
'Convert Data from FFh-00h to 200-0
   DispSP(Index) = (200 / 255) * DataSP(Index)
   DispPV(Index) = (200 / 255) * DataPV(Index)
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub Form_Load ()
   Assign
   Start = 0
   Index = 0
   i = 1: m = 1: k = 2: z = 1: t = 502
    ' Use COM2.
    Comm.CommPort = 2
    ' 4800 baud, no parity, 8 data, and 1 stop bit.
    Comm.Settings = "4800,N,8,1"
    ' Tell the control to read entire buffer when Input is used.
    Comm.InputLen = 0
    ' Open the port.
    Comm.PortOpen = True
End Sub

Sub Form_Paint ()
   If Start = 1 Then
     z = 0
     For i = m To k - 1
       If z <> 0 Then
            Pic1.Line (t + z - 2, 200 - DispSP(i - 1))-(t + z, 200 - DispSP(i)), RGB(0, 0, 0)
            Pic1.Line (t + z - 2, 200 - DispPV(i - 1))-(t + z, 200 - DispPV(i)), RGB(0, 0, 0)
       End If
       z = z + 2
     Next

     t = t - 2
     If t < 0 Then
       t = 0
       m = m + 1
     End If

     z = 0
     For i = m To k
       If z <> 0 Then
            Pic1.Line (t + z - 2, 200 - DispSP(i - 1))-(t + z, 200 - DispSP(i)), RGB(255, 0, 0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Pic1.Line (t + z - 2, 200 - DispPV(i - 1))-(t + z, 200 - DispPV(i)), RGB(0, 255, 0)
         End If
         z = z + 2
       Next
       label1.Caption = DispSP(Index) / 2
       label2.Caption = DispPV(Index) / 2
       label3.Caption = Mode
       label4.Caption = Kp
       label5.Caption = Ki
       label6.Caption = Kd
       k = k + 1: 'Index = Index + 1
       If Index = 1000 Then
         Index = 1
       End If
     End If
   End Sub

   Sub ReadData ()
      Do
       Comm.InBufferCount = 0
       ' Wait for data to come back to the serial port.
         Do
            Dummy = DoEvents()
         Loop Until Comm.InBufferCount >= 6
       ' Read the "OK" response data in the serial port.
       Index = Index + 1
       InData = Comm.Input
       Convert
       Form_Paint
      Loop Until Start = 0
   End Sub

   Sub Timer1_Timer ()
      Panel8.Caption = Now
   End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้