



การใช้ ไมโครคอมพิวเตอร์ เป็น ระบบควบคุมแบบกระจายส่วน (เฟส 1)

DCS BASED ON MICROCOMPUTER (PHASE 1)



โดย
นาย ฉวีชัย พงษ์พานิช
นาย อัครฉัตร ตั้งสุขเกษมสันต์
น.ส. อัจฉรา ชยาพิพัฒน์

วัน เดือน ปี 18 ม.ค. 2537
เลขทะเบียน... 034797
เลขเรียกหนังสือ... T 37097 คบ5

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมการวัดคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาก่อนนำเอกสารทุกครั้งที่มีการนำไปใช้



034797

M034797

ปีการศึกษา 2537

การใช้ไมโครคอมพิวเตอร์ เป็นระบบควบคุมแบบกระจายส่วน

โดย

นาย วัชรพงษ์พานิช

นาย อัครฉัตร ตั้งสุขเกษมสันต์

น.ส. อังฉรา ชยาพิวัฒน์

อาจารย์ที่ปรึกษา

ผศ. ธนิตย์ ตรีสุวรรณวัฒน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2537

ภาควิชา เทคโนโลยีการควบคุมทางอุตสาหกรรม


คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีการควบคุมทางอุตสาหกรรม

เรื่อง การใช้ไมโครคอมพิวเตอร์ เป็นระบบควบคุมแบบกระจายส่วน

DCS BASED ON MICROCOMPUTER

ผู้จัดทำ

1. นาย ณวัชร พงษ์พานิช 34102111
2. นาย อัครฉัตร ตั้งสุขเกษมสันต์ 34109495
3. น.ส. อังฉรา ชยาพิวัฒน์ 34109500


.....อาจารย์ที่ปรึกษา
(ผศ. ธนิตย์ ศรีสุวรรณวัฒน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้ ไมโครคอมพิวเตอร์ เป็น ระบบควบคุมแบบกระจายส่วน (เฟส 1)

โดย นาย ฌวัชร พงษ์พานิช 34102111
นาย อัครฉัตร ตั้งสุขเกษมสันต์ 34109495
น.ส. อัจฉรา ชยาพิวัฒน์ 34109500
อาจารย์ที่ปรึกษา
ผศ. ธนิตย์ ศรีสุวรรณวัฒน์
ปีการศึกษา 2537

บทคัดย่อ

ในปัจจุบัน เพื่อการควบคุมอย่างมีประสิทธิภาพ ต้องมีการพัฒนาระบบควบคุมแบบเดิมๆ ให้เป็นระบบใหม่ ที่ประกอบด้วย ระบบควบคุมกระบวนการ และการควบคุมตามลำดับขั้น เพื่อนำไปสู่ ระบบควบคุมแบบกระจายส่วน

ระบบของ การควบคุมแบบกระจายส่วน ทั้งชุดมีราคาสูงมากประกอบกับ คอมพิวเตอร์ส่วนบุคคลในปัจจุบันมีประสิทธิภาพ ในราคาไม่แพงนักดังนั้นหากสามารถจำลองการทำงานของ การควบคุมแบบกระจายส่วน โดยใช้ คอมพิวเตอร์ส่วนบุคคล จะทำให้ค่าใช้จ่ายลดลงเป็นอย่างมาก จากประโยชน์ดังกล่าว คณะผู้จัดทำ จึงได้ทำการจำลอง การทำงานของ ระบบการควบคุมแบบกระจายส่วนดังกล่าว

นอกจากจะสามารถควบคุมระบบได้แล้ว โปรแกรมยังสามารถที่จะจำลองกระบวนการ ควบคุมง่ายๆ เพื่อให้ผู้ใช้ได้สามารถศึกษาถึงธรรมชาติของกระบวนการควบคุมแต่ละชนิดได้ด้วย ซึ่งเป็นการพัฒนาความรู้ความสามารถของผู้ใช้งานได้อีกทางหนึ่ง

DCS BASED ON MICROCOMPUTER

NAWAT	PONGPANICH	34102111
AKARACHAT	TANGSUKKASAMSUN	34109495
ACHARA	CHAYAPIWAT	34109500
ADVISOR		
THANT	TRISUWANAWAT	

ABSTRACT

FOR HIGHER PERFORMANCE OF OPERATION , THE CLASSICAL CONTROL SYSTEM MUST BE DEVELOPED TO THE NEW ONE , WHICH ,CONSIST OF PROCESS CONTROL SYSTEM AND SEQUENTIAL CONTROL SYSTEMS, THAT CORRESPONDING TO THE DISTRIBUTED CONTROL SYSTEM (DCS).

DCS IS VERY EXPENSIVE AND EFFICIENT PERSONAL COMPUTER IS NOT EXPENSIVE SO MUCH , IF WE CAN SIMULATE DCS BASED ON MICROCOMPUTER , IT MEAN THE LOWER PRICE TOO.

BESIDE CONTROLLING THE PROCESS , THE PROGRAM CAN BE SIMULATE A SIMPLE PROCESS IN REASON FOR STUDYING ABOUT EACH CONTROLLING PROCESS , SO THE USER KNOWLEDGE IS OBTAINED.

สารบัญ

เนื้อเรื่อง	หน้า
บทที่ 1 การควบคุมกระบวนการ	2
บทที่ 2 ตัวควบคุมแบบ PID และการนำไปใช้งาน	5
-Proportional Control	5
-Proportional Plus Integral Control	6
-Proportional Plus Integral Plus Derivative Control	6
-Proportional Plus Derivative Control	6
บทที่ 3 ระบบควบคุมแบบกระจายส่วน	7
บทที่ 4 ลักษณะการจำลองการทำงานเครื่องวัดระดับ	9
บทที่ 5 บทสรุป	11
บทที่ 6 เกี่ยวกับโครงการ	12
บทที่ 7 วิธีการใช้โปรแกรม	15
บทที่ 8 เกี่ยวกับโครงการและสรุป	17
บทที่ 9 แนวทางในการพัฒนาต่อ	18
-ด้านฮาร์ดแวร์	18
-ด้านซอฟต์แวร์	18
บทที่ 10 การทดลอง	20
ภาคผนวก	25
รายละเอียดเกี่ยวกับ ADDA การ์ด	26
SOURCE PROGRAM	29
INCLUDE FILE	55
-GRAPH.H	55
-WALL.H	61
-THAL.H	66
-MOUSE.H	68
บรรณานุกรม	70

บทที่ 1

การควบคุมกระบวนการ (Process Control)

“การควบคุมกระบวนการ” คือ การควบคุมให้กระบวนการมีสถานะทางฟิสิกส์ตามต้องการซึ่งอาจแสดงเป็นสถานะ (State) หรืออาจแสดงได้เป็นจำนวนตัวเลข(Quantities)

องค์ประกอบหลักในการควบคุมจะประกอบด้วย

ก.กระบวนการ (Process) หมายถึงกระบวนการที่เราต้องการควบคุมให้มีสถานะทางฟิสิกส์ตามต้องการ ค่าตัวแปรต่างๆในกระบวนการประกอบด้วย

PV(Process Variable) คือ ตัวแปรกระบวนการ หมายถึงตัวแปรทางฟิสิกส์ซึ่งแสดงสถานะทางฟิสิกส์ของกระบวนการเราเรียกว่า

ตัวแปรปรับกระบวนการ(Manipulate Variable) คือ ตัวแปรปรับกระบวนการ หมายถึงค่าที่เราสามารถปรับเพื่อเปลี่ยนแปลงค่าตัวแปรกระบวนการได้

"สิ่งรบกวนกระบวนการ (Process Disturbance)" หมายถึง การเปลี่ยนแปลงของสภาพการทำงานและสภาพแวดล้อมของกระบวนการ ซึ่งจะส่งผลให้ค่าตัวแปรกระบวนการเปลี่ยนค่าไป

ข.อุปกรณ์วัด (Measuring Device) เป็นอุปกรณ์ที่ให้สัญญาณขาออก ซึ่งมีขนาดสัมพันธ์กับขนาดตัวแปรทางฟิสิกส์ของสิ่งที่เราต้องการวัด

ค.ตัวควบคุม (Controller) หมายถึงสิ่งที่ทำหน้าที่ออกคำสั่ง หรือกำเนิดสัญญาณควบคุม (Control Signal) ตามกฎเกณฑ์การควบคุม (Control Law) ที่ถูกกำหนดไว้ล่วงหน้า

ง.อุปกรณ์ปรับกระบวนการ (Final Control Element) เป็นอุปกรณ์ที่ทำหน้าที่ปรับสถานะของกระบวนการด้วยการเปลี่ยนแปลงค่าตัวแปรปรับกระบวนการ (Manipulated variable) ตามคำสั่ง หรือสัญญาณควบคุมที่ได้รับ

ในการควบคุมกระบวนการเป้าหมาย คือ การรักษาตัวแปรกระบวนการให้ใกล้เคียงกับค่าที่ต้องการ หรือค่าเป้าหมาย (Desired Value หรือ Set Point) ซึ่งอาจจะมีค่าคงที่ หรือเปลี่ยนแปลง ขณะที่ระบบควบคุมกำลังทำงาน ตัวแปรกระบวนการอาจจะมีค่าคลาดเคลื่อนไปจากค่าเป้าหมาย

หมายถึง ถ้าสภาวะการทำงาน หรือสภาพแวดล้อมเกิดเปลี่ยนแปลงไป การกำจัดความคลาดเคลื่อนทำได้โดยการปรับค่าตัวแปรกระบวนการ (Manipulated Variable)

หลักการพื้นฐานในการปรับค่าตัวแปรกระบวนการสองประเภท คือ หลักการป้อนกลับ (Feedback) และ หลักการป้อนล่วงหน้า (Feedforward) ดังนั้น ชนิดของการควบคุมจึงแบ่งได้เป็นสองชนิด ดังนี้

I. การควบคุมแบบป้อนกลับ (Feedback Control) :

หลักสำคัญของการควบคุมแบบนี้ เริ่มจากการวัดค่าตัวแปรกระบวนการ (Process Variable) นำไปเปรียบเทียบกับค่าเป้าหมาย (Desired Value หรือ Set Point) ระบบจะเริ่มทำงานก็ต่อเมื่อค่าตัวแปรกระบวนการที่วัดได้มีค่าต่างจากค่าเป้าหมาย โดยที่ผลต่างของการเปรียบเทียบ (Deviation) หรือความคลาดเคลื่อน (Error) จะถูกนำไปคำนวณหาสัญญาณควบคุม (Control Signal) ที่เหมาะสม "ป้อนกลับ (Feedback)" ไปปรับกระบวนการ การทำงานจะกระทำซ้ำๆ จนตัวแปรกระบวนการมีค่าเท่ากับค่าเป้าหมาย

การควบคุมแบบ Feedback Control เป็นแบบที่เราพบบ่อยที่สุดโดยเฉพาะอย่างยิ่งในโรงงานอุตสาหกรรม (มากกว่า 80%) ตัวควบคุมที่ใช้ในการควบคุมเป็นแบบ PID ตัวควบคุม

II. การควบคุมแบบป้อนล่วงหน้า (Feedforward Control) :

หลักสำคัญของการควบคุมแบบนี้ เริ่มจากการวัดค่าสิ่งรบกวนไปรเซส (Process Disturbance) แล้วนำค่าที่วัดได้ไปคำนวณหาสัญญาณควบคุมที่เหมาะสม "ป้อนล่วงหน้า (Feedforward)" ไปปรับกระบวนการด้านกับสิ่งรบกวน ก่อนที่ตัวแปรกระบวนการจะเปลี่ยนแปลงไป

ตัวควบคุมที่ใช้ระบบควบคุมแบบป้อนล่วงหน้าจะเป็นคอมพิวเตอร์

ส่วนสำคัญของการควบคุมจะประกอบด้วย

- ตัวควบคุม ทำหน้าที่คอยควบคุมระบบ
- ส่วนคำนวณและปรับตั้งพารามิเตอร์ ทำหน้าที่คำนวณหาพารามิเตอร์ที่เหมาะสมส่งไป

ปรับตั้งตัวควบคุม

การทำงานแบ่งได้เป็นสองช่วงคือ ช่วงแรกเป็นการคำนวณหาและปรับตั้งพารามิเตอร์ของตัวควบคุม (ช่วงนี้สัญญาณควบคุมจะมีค่าคงที่) และช่วงที่สองตัวควบคุมจะทำหน้าที่ควบคุมกระบวนการ (สัญญาณควบคุมอาจเปลี่ยนแปลงตามความเหมาะสม)

คอมพิวเตอร์ Control หมายถึงการควบคุมที่ใช้คอมพิวเตอร์แบบดิจิทัล (Digital computer) ทำหน้าที่เป็นตัวควบคุม

การควบคุมแบบนี้มีข้อดีที่ ความแม่นยำ ความสามารถทำการคำนวณที่ซับซ้อน ความ
สะดวกในการตัดแปลงโครงสร้างของระบบควบคุม ความสามารถในการจัดการ และจัดเก็บข้อมูล
จำนวนมาก ๆ ฯลฯ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ตัวควบคุมแบบ PID และการนำไปใช้งาน

(PID ตัวควบคุม and Its Utilization)

การออกแบบระบบควบคุมทางด้านอุตสาหกรรม (Industrial Control) เป็นงานที่ค่อนข้างจะยุ่งยากซับซ้อน แต่อย่างไรก็ตามเราสามารถสรุปการออกแบบระบบควบคุมได้ด้วยคำถามสองข้อหลัก คือ เป้าหมายหลักของการควบคุมคืออะไรและเราจะดำเนินการอย่างไร

แบบต่างๆ ของการควบคุมแบบป้อนกลับ (Feedback Control Modes)

แบบของการควบคุมแบบป้อนกลับถูกกำหนด โดยความสัมพันธ์ระหว่างสัญญาณขาออก (ตัวแปรปรับกระบวนการ) กับสัญญาณขาเข้า (PV) ของตัวควบคุม (ตัวควบคุม) ความสัมพันธ์ดังกล่าวจะขึ้นอยู่กับกฎเกณฑ์การควบคุม (Control Law) ที่ Operator เป็นผู้ "ปรับตั้ง" ไว้ล่วงหน้า

PID ตัวควบคุม กฎเกณฑ์การควบคุมพื้นฐานมีอยู่ด้วยกัน 3 แบบคือ Proportional (P) Integral (I) และ Derivative (D)

1. Proportional Control (P-only Control)

ความสัมพันธ์ระหว่างสัญญาณขาออก (ตัวแปรปรับกระบวนการ) กับสัญญาณขาเข้า (PV) ของ P-only ตัวควบคุม ที่ค่าเป้าหมาย (SP) ใด ๆ แสดงได้ดังนี้

$$MV = (100 / PB) \cdot (SP - PV) + b \dots\dots\dots(1)$$

ตัวแปรปรับกระบวนการ = สัญญาณขาออก

SP = ค่าเป้าหมาย

PV = สัญญาณขาเข้า

b = ค่าไบแอส (Bias) เป็นค่าสัญญาณควบคุมขณะที่ไม่มีความผิดพลาด (SP = PV)

PB = Proportional Band มีหน่วยเป็นเปอร์เซ็นต์ (%)

อัตราการตอบสนองต่อความผิดพลาดของ P-only ตัวควบคุม ถูกกำหนดด้วยค่า Proportional Band

เราให้คำจำกัดความของ Proportional Band ว่าเป็นเปอร์เซ็นต์การเปลี่ยนแปลงของสัญญาณขาเข้า (กรณีค่าเป้าหมายคงที่) ที่ทำให้สัญญาณขาออกเปลี่ยนแปลงไป 100%

บางครั้งเราอาจจะพบคำว่า Proportional Gain(K) แทนคำว่า Proportional Band โดยสมการที่ (1) จะกลายเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$MV = K(SP-PV) + b \quad \text{.....(2)}$$

ความสัมพันธ์ระหว่าง Proportional Gain กับ Proportional Band แสดงได้ด้วยสมการที่ 3

$$K = 100 / PB \quad \text{.....(3)}$$

2. Proportional-Plus-Integral Control (PI-Control)

ในทางอุตสาหกรรมเรามักจะเรียกว่า "Proportional-Plus-Reset Control"

ระบบควบคุมส่วนมากมักจะเป็นการควบคุมแบบ PI-Control โดยเฉพาะอย่างยิ่งในการควบคุมอัตราการไหลและการควบคุมแรงดันของของเหลว ฯลฯ

ความสัมพันธ์ระหว่างสัญญาณขาออกกับสัญญาณขาเข้าของ PI-ตัวควบคุม แสดงได้ดังนี้

$$MV = (100 / PB) \cdot [(SP-PV) + (1 / T_i) \cdot \int (SP-PV) \cdot dt] + b \quad \text{.....(4)}$$

โดยที่ T_i = Integral Time หรือ Reset Time หน่วยเป็นนาที (Minute)

หมายเหตุ ในตำราบางเล่มอาจเรียก T_i ว่า Minutes Per Repeat

3. Proportional-Plus-Integral-Plus-Derivative Control (PID Control)

ในทางอุตสาหกรรมมักจะเรียกว่า "Proportional-Plus-Reset-Plus-Rate Control"

ดังที่กล่าวมาแล้วว่า Integral Action ใน PI-Control จะทำให้ตัวควบคุมตอบสนองต่อความผิดพลาดช้าลง กรณีที่กระบวนการช้าอยู่แล้วเราอาจจำเป็นต้องเพิ่ม Derivative Action (หรือ Rate Action) เพื่อลดความช้าของระบบควบคุม การควบคุมแบบนี้เราเรียกว่า PID-Control

เราจะพบการควบคุมแบบ PID ในงานควบคุมอุณหภูมิเป็นส่วนใหญ่

ความสัมพันธ์ระหว่างสัญญาณขาออก (ตัวแปรปรับกระบวนการ) กับสัญญาณขาเข้า (PV) ของ PID-ตัวควบคุม ที่ค่าเป้าหมาย (SP) ใดๆ แสดงได้ดังนี้

$$MV = (100 / PB) \cdot [(SP-PV) + (1 / T_i) \cdot \int (SP-PV) \cdot dt + T_d \cdot d(SP-PV) / dt] + b \quad \text{.....(5)}$$

T_d = Derivative Time หรือ Rate Time หน่วยเป็นนาที (Minute)

"ข้อดีของ PID Control คือ ระบบควบคุมจะตอบสนองต่อสิ่งรบกวน (Disturbance) ไวขึ้น ทำให้เสถียรภาพในการควบคุมดีขึ้นกว่า PI Control สำหรับกระบวนการที่ตอบสนองช้า"

4. Proportional-Plus-Derivative Control (PD-Control)

ในทางอุตสาหกรรมเรามักจะเรียกว่า "Proportional-Plus-Rate Control" ลักษณะความสัมพันธ์ของสัญญาณขาออกกับสัญญาณขาเข้าของตัวควบคุม PD-ตัวควบคุม มีสมการดังนี้

$$MV = (100 / PB) \cdot [(SP-PV) + T_d \cdot d(SP-PV) / dt] + b \quad \text{.....(6)}$$

PD-Control จะใช้ในการควบคุมกระบวนการจำพวกเดียวกับ P-only Control โดย PD-

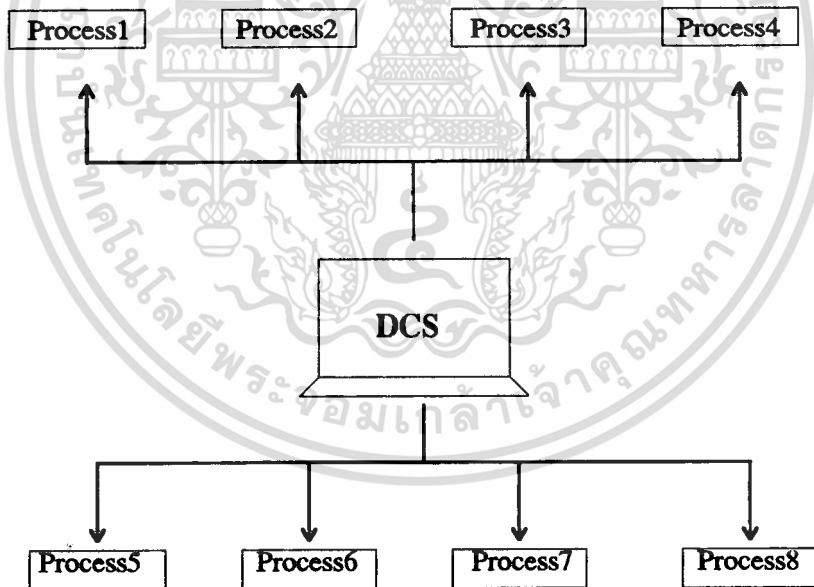
Control จะให้ค่า Offset น้อยกว่า P-only Control

บทที่ 3

ระบบการควบคุมแบบกระจายส่วน

DISTRIBUTE CONTROL SYSTEM (DCS)

ในการควบคุมกระบวนการต่างๆให้มีสภาพทางฟิสิกส์ที่ต้องการได้นั้น ในปัจจุบันนับว่ามีความจำเป็นอย่างยิ่ง ที่จะต้องมีเครื่องคอมพิวเตอร์มาใช้ในการควบคุม เนื่องจากมีความเที่ยงตรงและให้ผลการควบคุมที่แน่นอน แต่อย่างไรก็ตามเนื่องจากในสายการผลิตจริงมีหลายกระบวนการประกอบกัน และแต่ละกระบวนการก็อาจมีความสัมพันธ์กัน ดังนั้นจึงต้องใช้คอมพิวเตอร์หลายตัวมาควบคุมแต่ละกระบวนการ ซึ่งจะเป็นผลให้มีคอมพิวเตอร์จำนวนหลายตัว สร้างความยุ่งยากและเป็นการเพิ่มรายจ่ายโดยไม่จำเป็น ดังนั้นจึงมีการพัฒนาระบบที่สามารถควบคุมกระบวนการได้หลายๆ กระบวนการ ในเวลาเดียวกันโดยเครื่องคอมพิวเตอร์เครื่องเดียว ดังรูป



โดยการควบคุมด้วยระบบควบคุมแบบกระจายส่วน (DCS) จะมีข้อดีกว่าการควบคุมด้วยระบบคอมพิวเตอร์อื่นๆที่ ทำให้งานที่ได้มีประสิทธิภาพสูงกว่า เนื่องจากสามารถทำงานได้ที่ละหลายๆงาน โดยไม่ต้องหยุดรอ และในการขยายจำนวนกระบวนการ ก็ทำได้ง่าย เพราะเมื่อ

ต้องการขยายก็เพียงแค่นำระบบใหม่มาเชื่อมต่อกับระบบที่มีอยู่แล้ว และทำการปรับค่าตัวแปร กระบวนการต่าง ๆ ให้เหมาะสม

DCS ที่ได้ มีส่วนประกอบ ต่างๆ คือ

1.Operator Station

เป็นอุปกรณ์ที่จะติดต่อสื่อสารกับผู้ที่มีความคุมการทำงาน โดยการแสดงผลทางจอภาพ โดยข้อมูลที่แสดงอยู่ในจอภาพนั้น ถูกส่งมาจาก Field Control Unit โดยผ่านทางระบบสายส่งข้อมูล และในหน้าจอจะมีกราฟที่แสดงการเปลี่ยนแปลงของระบบ เพื่อให้เข้าใจการทำงานจากระบบ ได้ดีขึ้น

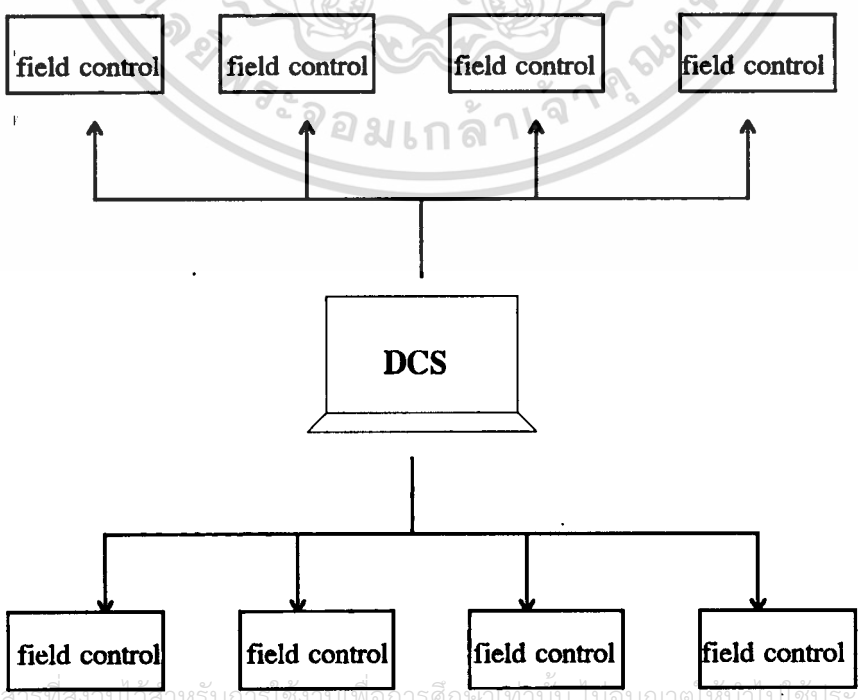
2.Field Control Unit

เป็นหน่วยหนึ่งในคอมพิวเตอร์ที่ใช้ประมวลผลโดยตรง โดยจะมีส่วนประมวลผลอยู่ในตัวเองและจะจำลองให้ทำงานเหมือนกับเป็นตัวควบคุมกระบวนการ การทำงานต่างๆของตัว Field Control Unit จะทำงานในรูปแบบของการจำลองการทำงานโดยการใช้โปรแกรม

เช่น การเชื่อมต่อกับตัวควบคุมกระบวนการ ในรูปของ Software Wiring และตัว Field Control Unit ยังมีส่วนการทำงานพิเศษ ก็คือรูปของการควบคุมแบบลำดับ ที่สามารถกำหนดการทำงานได้จากตารางลำดับการทำงานที่ได้สร้างไว้

3.ระบบสายส่งข้อมูล

จะเป็นส่วนเชื่อมสัญญาณระหว่าง Field Control Unit กับ Operator Station โดยระบบสายส่งนี้ จะมีอัตราเร็วในการส่งต่าง ๆ กันไปขึ้นอยู่กับชนิดของวัสดุที่ใช้ในการทำสายส่ง

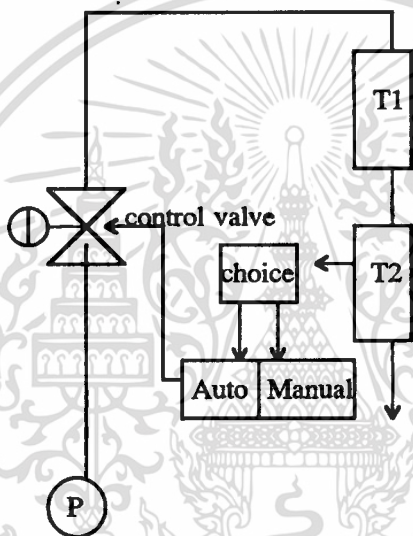




บทที่ 4

ลักษณะการจำลองการทำงานเครื่องวัดระดับ

เป็นโปรแกรมที่ ทำการจำลองการทำงาน ขึ้นมาเพื่อใช้ในการศึกษาการควบคุมระดับใน รูปแบบต่างๆโดยจะมีการวัดระดับในถัง 2 ถังที่ตั้งต่อกัน โดยที่อัตราการไหลออกจากถังใบบน จะเป็น อัตราการไหลเข้าของถังใบล่าง ดังรูป



เราสามารถที่จะหาความสูงที่เพิ่มขึ้นของระดับของเหลวในถังได้จากสมการทั่วไปคือ

$$H = (Q_{in} - Q_{out}) / A$$

(สมการที่ 1)

- โดยกำหนดให้ H คือ ความสูงของของเหลวในถังที่เพิ่มขึ้นใน 1 ช่วงเวลา
- Qout คือ อัตราการไหลเข้าถังของของเหลวในช่วงเวลา
- Qin คือ อัตราการไหลออกจากถังของของเหลวในช่วงเวลา
- A คือ พื้นที่หน้าตัดของถัง

ถ้าเราพิจารณาที่ทางออกของของเหลวที่เป็นท่อเราสามารถหาค่า อัตราการไหลออกของของเหลวได้ดังนี้คือ

$$Q_{out} = (C)(H)^{1/2}$$

(สมการที่ 2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ C เป็นค่าคงที่ของท่อนำของเหลวออกที่ได้จากการทดลอง จากทั้ง 2 สมการข้างต้น เราสามารถหาค่าความสูงของระดับของเหลวที่เวลาใดๆ ได้จากสมการ

$$H(t) = H(t-1) + H \quad (\text{สมการที่ 3})$$

เมื่อ $H(t)$ คือ ความสูงของถังที่เวลา T

$H(t-1)$ คือ ความสูงของถังที่เวลา $T-1$

H คือ ความสูงที่เพิ่มขึ้นในช่วงเวลา T ถึง $T-1$

แทนค่าสมการ 2 ลงในสมการ 1 ได้สมการที่ 4.1 และ 4.2 ดังนั้น ความสูงของของเหลวที่เพิ่มขึ้นใน 1 ช่วงเวลาในถังที่ 1 (T_1) คือ

$$H_1 = (Q_{in} - (C_1)(H_1(t-1))^{1/2}) / A_1 \quad (\text{สมการที่ 4.1})$$

และสำหรับค่าความสูงในถัง 2 (T_2) คือ

$$H_2 = ((C_1)(H_1(t-1))^{1/2}) / A_2 \quad (\text{สมการที่ 4.2})$$

เมื่อเราได้ความสูงที่เพิ่มขึ้นมาในช่วงเวลาที่เรากำหนดเราก็นำมารวมกับความสูงเดิมที่มีอยู่ (จากสมการที่ 3) เราก็จะได้ความสูงของถังที่เวลานั้นๆ และเมื่อเราทราบถึงความสูงของๆเหลวในถังเท่าไร เราก็ทำการเปรียบเทียบกับค่าเป้าหมายที่เราได้ตั้งไว้เพื่อจะนำค่าไปคำนวณค่า วาล์วควบคุม เพื่อใช้เป็นค่าการปิดวาล์วนั้นเอง

บทที่ 5

บทสรุป

ในยุคปัจจุบันเทคโนโลยีได้รับการพัฒนาอย่างรวดเร็ว อุปกรณ์แบบใหม่ๆ ได้ถูกนำมาใช้ในโรงงานเพื่อประสิทธิภาพและประสิทธิผลในการผลิต ตัวควบคุมแบบ Analog PID ตัวควบคุมซึ่งเคยถูกนำมาใช้แทนที่ Pneumatic PID ตัวควบคุม กำลังจะล้าสมัยไปการนำเอาไมโครคอมพิวเตอร์มาประยุกต์ใช้ในงานควบคุมได้ทวีจำนวนมากขึ้นทุกขณะ

อย่างไรก็ตาม ถ้าเราพิจารณากันให้ลึกซึ้งจะพบว่าหลักการควบคุม (Control law) ของตัวควบคุมไม่ว่าจะเป็นแบบ Pneumatic Analog หรือ Digital ยังคงเดิมไม่เปลี่ยนแปลงระบบควบคุมอาจจะมีประสิทธิภาพมากขึ้น ขนาดเล็กลง ฯลฯ แต่ระบบควบคุมส่วนใหญ่ในโรงงานใหญ่บางโรงได้นำหลักการควบคุมที่ซับซ้อนกว่าเช่น Feedforward Control, Adaptive Control หรือ Optimum Control มาใช้งานแต่ก็เป็นเพียงส่วนน้อย

ดังจะให้เห็นแล้วว่าในการควบคุมระบบต่างๆในโรงงานอุตสาหกรรม คอมพิวเตอร์ได้เข้ามามีส่วนร่วมเป็นอันมาก จึงเป็นที่มาของระบบควบคุมแบบกระจายส่วน ซึ่งนับว่ามีประโยชน์อย่างมากในวงการอุตสาหกรรม ในด้านของการควบคุมที่เที่ยงตรง แม่นยำ ลดความยุ่งยากในด้านการใช้เครื่องคอมพิวเตอร์จำนวนน้อยลง ตลอดจนลดค่าใช้จ่ายในการซื้อเครื่องคอมพิวเตอร์หลายๆเครื่อง ทั้งโครงการจีนี้อย่างช่วยประหยัดค่าใช้จ่ายในการซื้อเครื่องระบบควบคุมแบบกระจายส่วน สำเร็จรูปอีกด้วย เนื่องจากในปัจจุบัน เครื่องระบบควบคุมแบบกระจายส่วน สำเร็จรูป ยังมีราคาแพงมากอีกด้วย

จากที่ได้กล่าวอธิบายมาทั้งหมดผู้เขียนได้พยายามเน้นถึงหลักการพื้นฐาน การทำงานและการนำไปใช้งานของตัวควบคุมแบบ PID โดยควบคุมด้วยระบบควบคุมแบบกระจายส่วน ซึ่งผู้เขียนหวังว่าจะยังคงจะมีประโยชน์ใช้งานได้ในวงการอุตสาหกรรมไปอีกนาน

บทที่ 6

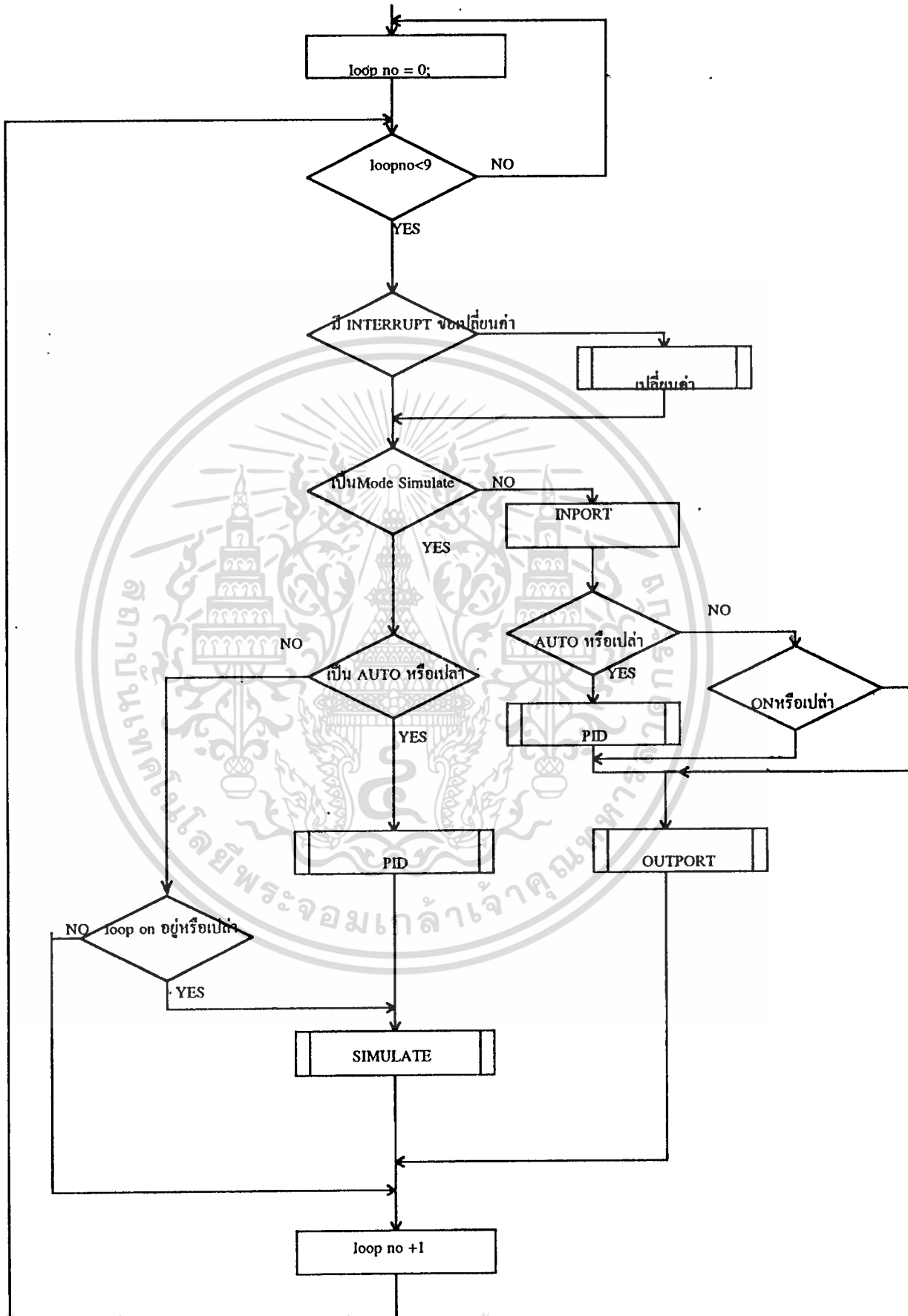
เกี่ยวกับโครงการ

เนื่องจาก ระบบควบคุมแบบกระจายส่วน เป็นระบบควบคุมที่มีขนาดใหญ่ และซับซ้อนมาก มีฟังก์ชันต่างๆมากมาย ดังนั้นการที่เราจะจำลองการทำงานของ ระบบควบคุมแบบกระจายส่วนทั้งหมดมาเป็นโปรแกรมคอมพิวเตอร์ที่สมบูรณ์นั้น จึงเป็นเรื่องที่ต้องใช้เวลามาก แต่ถ้าไม่มีการเริ่มต้นก็ไม่มีความสำเร็จ โครงการนี้ของเรา จึงทำขึ้นเพื่อเป็นแนวทางในการพัฒนาต่อไป

ขอบเขตของโครงการที่ทำในครั้งนี คือการเขียนโปรแกรมเพื่อให้ควบคุมกระบวนการต่างๆ ได้พร้อมกัน 8 กระบวนการ โดยสามารถเปลี่ยนค่าตัวแปรควบคุม (PARAMETER) ต่างๆได้โดยไม่ทำให้การทำงานของทุกๆกระบวนการเกิดการหยุดชะงัก ซึ่งในส่วนนี้จะเป็นส่วนสำคัญของการเขียนโปรแกรม

สำหรับในส่วนของการติดต่อระหว่าง คอมพิวเตอร์ กับ กระบวนการเพื่อรับส่งค่านั้น เริ่มแรกได้ออกแบบว่าจะใช้ 25CU ต่อผ่านทางพอร์ตอนุกรม แต่เนื่องจาก ตัว 25CU ที่จะใช้นั้นชำรุด ดังนั้นเราจึงใช้ ADDA CARD แทนทำให้สามารถติดต่อกับกระบวนการภายนอกได้เพียงกระบวนการเดียว คณะผู้จัดทำจึงได้ทำการจำลองกระบวนการมาใช้แทน 7 กระบวนการที่เหลือ แต่เราก็ได้เขียนโปรแกรมรองรับในส่วนที่จะขยายเพื่อต่อควบคุมกระบวนการทั้ง 8 ไว้แล้ว

ในด้านหลักการของการทำงานนั้น คือ การให้คอมพิวเตอร์ทำการตรวจสอบการทำงานของแต่ละกระบวนการ ว่าต้องทำอะไรบ้าง นั่นคือ เมื่อเริ่มการทำงานทุกๆกระบวนการ จะถูกกำหนดให้เป็น การจำลองการทำงานของระบบควบคุมระดับน้ำ ซึ่งสามารถเขียนเป็นแผนผังการไหล (flowchart) ของการทำงานได้ในหน้าถัดไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ถ้าหากต้องการเปลี่ยนค่าตัวแปรควบคุมต่างๆ คือ ค่าเป้าหมาย ,ตัวแปรปรับโปรเซส , ตัวแปรโปรเซส ,Kp ,Ki ,Kd ,ค่าสูงสุดของการควบคุม ,ค่าต่ำสุดของการควบคุม ก็สามารถส่ง สัญญาณไปรบกวนการ ทำงานของระบบเพื่อให้ แวะ มาทำการเปลี่ยนค่าก่อน ถ้าหยุดมาทำการ เปลี่ยนค่าเป็นเวลานาน จะทำให้การควบคุมกระบวนการเกิดการผิดพลาดได้ จึงได้ลดการ ทำงานต่างๆ ให้เหลือเป็นเพียงขั้นตอนนี้ไม่ต้องมา หยุดรอ อ่านค่า

และสามารถเปลี่ยนกระบวนการใดก็ได้ให้เป็นการควบคุมกระบวนการจริง แต่เนื่องจาก ในการทำโครงการนี้มีการติดต่อกับภายนอกเพียงกระบวนการเดียว จึงได้มีการบังคับ โดย โปรแกรมเอาไว้ว่าถ้ามีกระบวนการใดต่อควบคุมกระบวนการจริงแล้วกระบวนการอื่นจึงไม่ สามารถเปลี่ยนให้เป็นหมวดควบคุมกระบวนการจริงได้

จากหลักการที่ใช้นี้จะพบว่าความเร็วของการทำงานครบ 1 รอบ จะขึ้นอยู่กับความเร็ว ของหน่วยประมวลผลของคอมพิวเตอร์ซึ่งในการทำโครงการนี้ใช้ หน่วยประมวลผลเบอร์ 386 SX ซึ่งสามารถให้ความเร็วในการควบคุมได้อย่างเพียงพอ



บทที่ 7

วิธีการใช้โปรแกรม

เมื่อเข้าสู่โปรแกรม จะเป็นภาพแรกของระบบ ดังรูปที่ 1 ซึ่งเป็นตารางสำหรับป้อนข้อมูล ต่างๆของแต่ละ ระบบ เพื่อแสดงให้เห็นว่าเราได้เข้าสู่โปรแกรมแล้วให้ทำตามขั้นตอนดังนี้

1. ทำการป้อนข้อมูลในส่วนของ Description ,ค่า ค่าเป้าหมาย ,ค่า ตัวแปรปรับกระบวนการ ,ค่า K_p ,ค่า K_i ,ค่า K_d ,ค่าจำกัดสูงสุด ,ค่าจำกัดต่ำสุด , และ สถานะ (STATUS) ของแต่ละกระบวนการ ในการเลือกช่องป้อนข้อมูลสามารถเลือกได้โดยการเลื่อนเมาส์ ให้ตรงกับช่องที่ต้องการ

2. เมื่อต้องการป้อนช่องใดให้ทำการกดเมาส์ แล้วทำการป้อน โดยจะสามารถป้อนค่าได้ตั้งแต่ 0-100 สำหรับค่า ค่าเป้าหมาย ,ตัวแปรปรับกระบวนการ ,จำกัดต่ำสุด และ จำกัดสูงสุด

3. เมื่อป้อนค่าเรียบร้อยแล้ว ให้กด พิมพ์ SPACE BAR แล้วเลื่อนเมาส์ เพื่อป้อนข้อมูลต่อไป ถ้าค่าที่ป้อนเข้าไปแล้วไม่ถูกต้อง ให้กดเมาส์ใหม่ เพื่อป้อนค่าใหม่ลงไป

4. สำหรับการใส่ ชื่อการควบคุม (Description) จะต้องเป็นตัวอักษรเท่านั้น

5. ค่าต่างๆซึ่งได้แก่ ค่าเป้าหมาย , K_p , K_i , K_d , จำกัดสูงสุด ,จำกัดต่ำสุด เมื่อป้อนค่าให้ป้อนค่าเฉพาะตัวเลขหรือทศนิยมเท่านั้น โดยส่วนของค่า ค่าเป้าหมาย ,จำกัดสูงสุด ,จำกัดต่ำสุด สามารถป้อนค่าให้อยู่ในช่วง 0-100 เท่านั้น แต่ถ้าไม่มีการป้อนค่าระบบจะมีการรับรู้ว่าค่านั้นๆเป็น 0

6.การระบุ สถานะ ให้ใช้ตัวอักษร A แทน ระบบควบคุมอัตโนมัติ (AUTO) , M แทน ระบบควบคุมแบบสั่งการ (MANUAL) และ O แทน ระบบที่ไม่มีการควบคุม (OFF)

7. เมื่อทำการป้อนค่าครบตามที่ต้องการแล้ว ให้เลื่อน เมาส์ ไปยังตำแหน่ง Finish enter values แล้วทำการกดเมาส์ เพื่อนำค่าที่ป้อนไปใช้ควบคุมต่อไป จากนั้น จะเป็นภาพ กราฟของการควบคุม ดังรูปที่ 2 แต่ถ้า ตำแหน่งสถานะ ของ ระบบ เป็น Off ระบบจะออกจากโปรแกรมการควบคุมทันที

8. สังเกตการเปลี่ยนแปลงของระบบใน ระบบต่างๆ ดังรูปที่ 2 ภาพซ้าย เป็นภาพกราฟขนาดย่อ ที่แสดงให้เห็นสถานะของแต่ละระบบ และสามารถเลือกได้ว่า จะทำการดูหรือแก้ไขระบบใด

9. การเลือกแสดงและปรับค่าต่างๆของระบบการทำงานที่ต้องการ ทำได้โดยการเลื่อนเมาส์ ไปยังระบบที่ต้องการ จากนั้นทำการกดเมาส์ที่ระบบที่ต้องการ

10. ภาพบนขวา เป็นกราฟขยายแสดงการเปลี่ยนแปลงของระบบ ซึ่งแสดงตัวเปลี่ยนแปลงต่างๆของระบบ

11. ภาพล่างขวา เป็นปุ่มสำหรับการเปลี่ยนค่าต่างๆในขณะการทำงาน ระบบ นั้นๆอยู่ ซึ่งจะทำการเปลี่ยนค่าได้โดยเลื่อนเมาส์ ไปยังค่าที่ต้องการเปลี่ยน จากนั้นทำการกดเปลี่ยนค่าเพิ่มขึ้นหรือลดลง สำหรับค่า ค่าเป้าหมาย ,ตัวแปรปรับกระบวนการ ,Kp ,Ki ,Kd ,จำกัดสูงสุด ,จำกัดค่าสุดตามที่ต้องการ

12. ค่า ค่าเป้าหมาย ,ตัวแปรปรับกระบวนการ ,จำกัดต่ำสุด ,จำกัดสูงสุด สามารถเปลี่ยนค่าได้อยู่ในช่วง 0-100 เท่านั้น

13. สถานะ สามารถเลือกได้จาก การ กด ที่ M ,A ,OFF แทนการเลือก ระบบควบคุมแบบสั่งการ ,อัตโนมัติ และ ปิด ตามลำดับ

14. ภาพล่างขวา เป็นปุ่มสำหรับเลือกให้ ระบบ ใด ระบบ หนึ่ง เป็น กระบวนการจริง หรือ กระบวนการจำลอง โดยการ เลื่อนเมาส์ไปกด ที่ กระบวนการ ที่ต้องการ จากนั้นทำการ กด ที่ กระบวนการจริง หรือ การจำลองการทำงาน

15. เมื่อจะจบการควบคุม และออกจากโปรแกรมให้เลื่อนเมาส์ ไปยังตำแหน่ง EXIT จาก นั้น ทำการกดเมาส์ ภาพบนจอ คอมพิวเตอร์ จะเข้าสู่ DOS ทันที เป็นการสิ้นสุดโปรแกรมการควบคุม

บทที่ 8

บทวิจารณ์และสรุป

ในปัจจุบันระบบควบคุมแบบกระจายส่วนมีความสำคัญต่อการพัฒนาอุตสาหกรรมของไทย แต่ในปัจจุบันระบบควบคุมแบบกระจายส่วนมีราคาที่สูง และต้องนำเข้าจากต่างประเทศ ดังนั้นคณะผู้จัดทำจึงได้พยายามสร้างระบบควบคุมแบบกระจายส่วน โดยการใช้เครื่องคอมพิวเตอร์ส่วนบุคคล ซึ่งมีราคาถูกกว่ามาก แต่ในการสร้างระบบควบคุมแบบกระจายส่วนที่สมบูรณ์แบบนั้นจำเป็นต้องมีองค์ประกอบต่าง ๆ ที่ซับซ้อนมากมาย ดังนั้นการทำโครงการนี้จึงเป็นการหยิบยกการทำงานของระบบควบคุมแบบกระจายส่วน เพียงบางส่วนเท่านั้น แต่คณะผู้จัดทำได้เสนอแนวคิดและวิถีทางในการพัฒนาโครงการนี้ต่อไปไว้แล้วในหัวข้อเกี่ยวกับโครงการ

คณะผู้จัดทำหวังเป็นอย่างยิ่งว่า จะมีการพัฒนาระบบควบคุมแบบกระจายส่วนนี้ต่อไปในภายหน้า จนมีองค์ประกอบที่สมบูรณ์ สามารถใช้แทนระบบควบคุมแบบกระจายส่วนจากต่างประเทศได้

ขีดจำกัดของระบบ

เนื่องจากการทำงานของโปรแกรม ได้ใช้รอบการทำงานของโปรแกรมเป็นความเร็วในการควบคุมกระบวนการ ดังนั้นจะเห็นได้ว่าถ้ามีการเปิดกระบวนการเพื่อควบคุมมาก การทำงานจะช้ากว่าการเปิดกระบวนการเพื่อควบคุมจำนวนน้อยกระบวนการ และความเร็วในการควบคุมกระบวนการยังขึ้นกับความเร็วของหน่วยประมวลผลหลักของคอมพิวเตอร์อีกด้วย

บทสรุป

ระบบควบคุมแบบกระจายส่วน เป็นระบบที่ใช้หน่วยประมวลผลหลายๆหน่วยทำงานร่วมกัน ดังนั้นจึงเป็นการยากที่จะขยู่การทำงานของหน่วยประมวลผลดังกล่าว มาทำงานโดยใช้หน่วยประมวลผลเพียงหน่วยเดียว ในโครงการนี้จึงได้จำลองการทำงานบางส่วนและเสนอแนะแนวทาง ในการสานต่อเพื่อให้เป็นระบบควบคุมแบบกระจายส่วนที่สมบูรณ์ต่อไปในอนาคต

บทที่ 9

แนวทางในการพัฒนาต่อ

ด้าน ฮาร์ดแวร์ คือนำ 25 CU มาใช้ ซึ่งโปรแกรมในส่วนนี้ได้เขียนไว้แล้ว ขาดแต่เพียงต้องทำการเพิ่มรูปแบบ (PROTOCOL) ของการรับ-ส่งค่า ระหว่างคอมพิวเตอร์ กับ 25CU ด้วยคือมีข้อมูลบิตเริ่มต้น (START BIT) การส่งเป็นอะไร มีข้อมูลบิตสุดท้าย (STOP BIT) เป็นอะไร

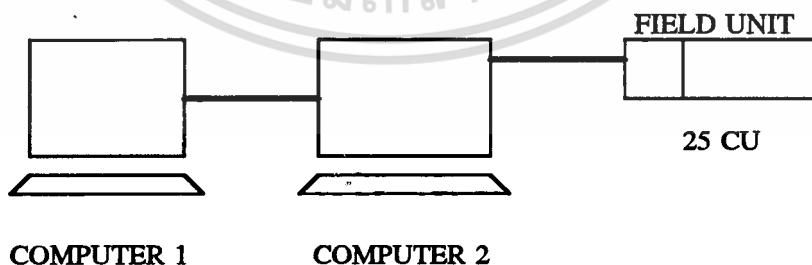
ด้าน ซอฟต์แวร์ มีรายละเอียดดังนี้

1. มีการเพิ่มทรานเฟอร์ฟังก์ชันมาตรฐานเข้ามาให้ผู้ใช้เลือกประกอบเป็นตัวควบคุมกระบวนการได้เองตามความเหมาะสม

2. มีการนำสัญญาณเอาต์พุตของกระบวนการหนึ่ง ไปเป็นอินพุตของอีกกระบวนการหนึ่ง หรือสามารถใช้สัญญาณเพื่อทำการควบคุมแบบ CASCADE , RATIO , FEEDFORWARD, FEEDBACK เป็นต้น

3. สามารถวาดรูปกระบวนการและดูการเปลี่ยนแปลงได้ เช่น ถังน้ำมีการเปลี่ยนแปลงระดับความสูงของน้ำในถัง เป็นต้น แต่ในการทำเช่นนี้จะเป็นการทำให้เสียเวลามาก ซึ่งจะทำให้การควบคุมกระบวนการผิดพลาดได้ ทางคณะผู้จัดทำจึงขอเสนอแนวคิดไว้ 2 แนวคิดคือ

3.1 การใช้คอมพิวเตอร์ 2 ตัว เชื่อมถึงกันโดยให้เครื่องหนึ่งทำการควบคุมกระบวนการซึ่งได้ทำไปแล้วในโครงการนี้ ส่วนคอมพิวเตอร์อีกเครื่องหนึ่ง จะใช้ในการวาดรูปแสดงการทำงานของกระบวนการที่ทำการควบคุม โดยคอมพิวเตอร์ทั้งสองตัวจะทำงานเป็นอิสระต่อกัน โดยมีการติดต่อกันก็เพียงเพื่อทำการรับและส่งค่าตัวแปรควบคุมต่าง ๆ เท่านั้น ดังรูป



COMPUTER 1 ทำหน้าที่รับ ส่ง และเก็บข้อมูล แสดงรูปกระบวนการต่าง ๆ รวมถึงทำหน้าที่เป็น เอดิเตอร์ และใช้ในการดูกราฟการเปลี่ยนแปลงของกระบวนการย้อนหลัง ซึ่งผู้ใช้สามารถเปลี่ยนค่าตัวแปรกระบวนการต่าง ๆ ได้ด้วย

COMPUTER 2 ทำหน้าที่ควบคุมกระบวนการโดยส่งผ่าน และรับค่าควบคุมต่าง ๆ ไปยัง 25CU และส่งค่าตัวแปรควบคุมกระบวนการต่าง ๆ ไปยัง COMPUTER 1 โดยที่ผู้ใช้สามารถปรับหรือเปลี่ยนแปลงตัวแปรควบคุมกระบวนการต่าง ๆ ได้ด้วย

25 CU ทำหน้าที่ในการรับและส่งผ่านค่าตัวแปรควบคุมกระบวนการจาก FIELD ไปยัง COMPUTER 2 โดย 25 CU ทำหน้าที่ในการแปลงระดับสัญญาณ 4-20 มิลลิแอมป์ ไปเป็นระดับสัญญาณทางคอมพิวเตอร์ รวมถึงในทางกลับกันด้วย

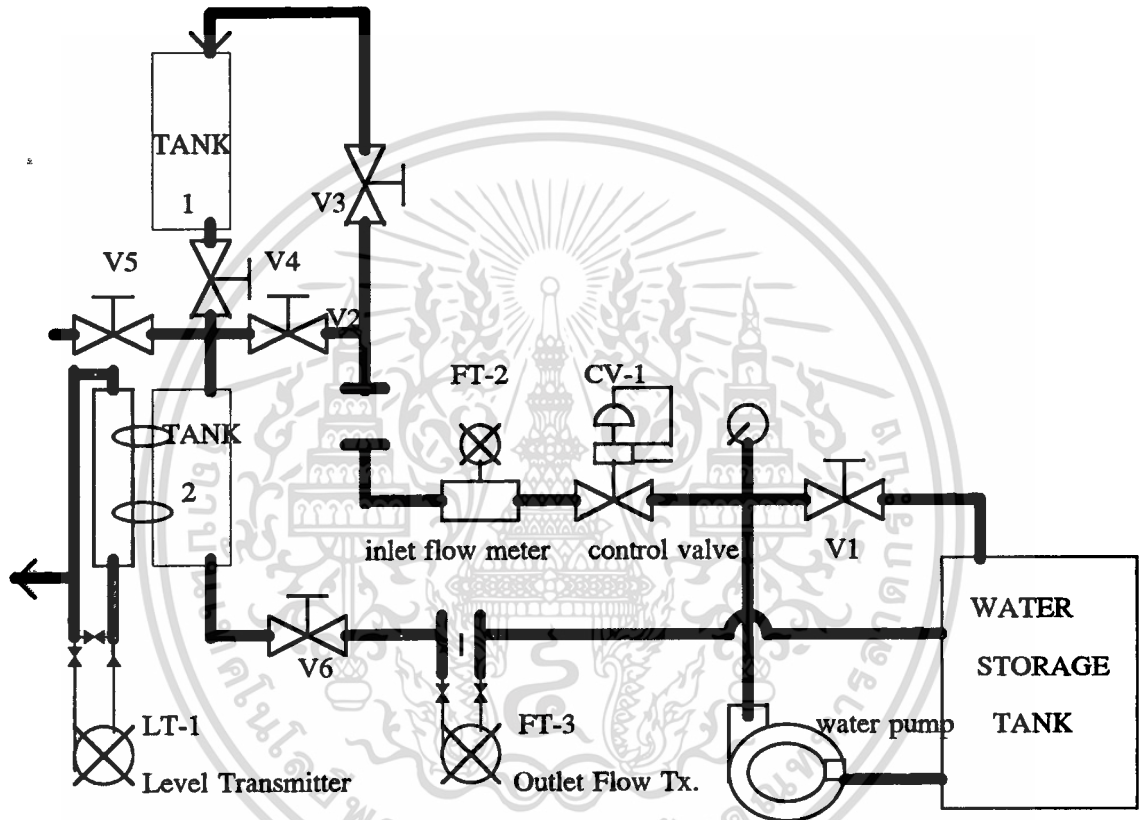
3.2 ใช้ระบบปฏิบัติการที่เป็นการทำงานแบบหลายงานในเวลาเดียวกัน (MULTITASK) ซึ่งในปัจจุบันระบบปฏิบัติการดังกล่าวนี้คือ ระบบ UNIX , OS/2 และ WINDOWS 4.1 ซึ่งระบบดังกล่าวเป็นระบบที่ใหม่และมีราคาสูง รวมถึงแหล่งข้อมูลในปัจจุบันยังมีจำนวนน้อย ผู้จัดทำจึงแนะนำให้เลือกเป็นแนวทางหลังที่จะตัดสินใจใช้



บทที่ 10

การทดลอง

ทำการทดลองโดยควบคุมกระบวนกรให้ทำงานตาม ระบบ ต่อไปนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่.1 รูปแสดงหน้าจอเริ่มต้นโปรแกรม

NO.	Description	SV	K1	K2	K3	MaxRange	MinRange	Status
1								
2								
3								
4								
5								
6								
7								
8								

Finish enter values

รูปที่.2 รูปแสดงหน้าจอตารางรับค่า

NO.	Description	SV	K1	K2	K3	MaxRange	MinRange	Status
1	LE							
2								
3								
4								
5								
6								
7								
8								

Finish enter values

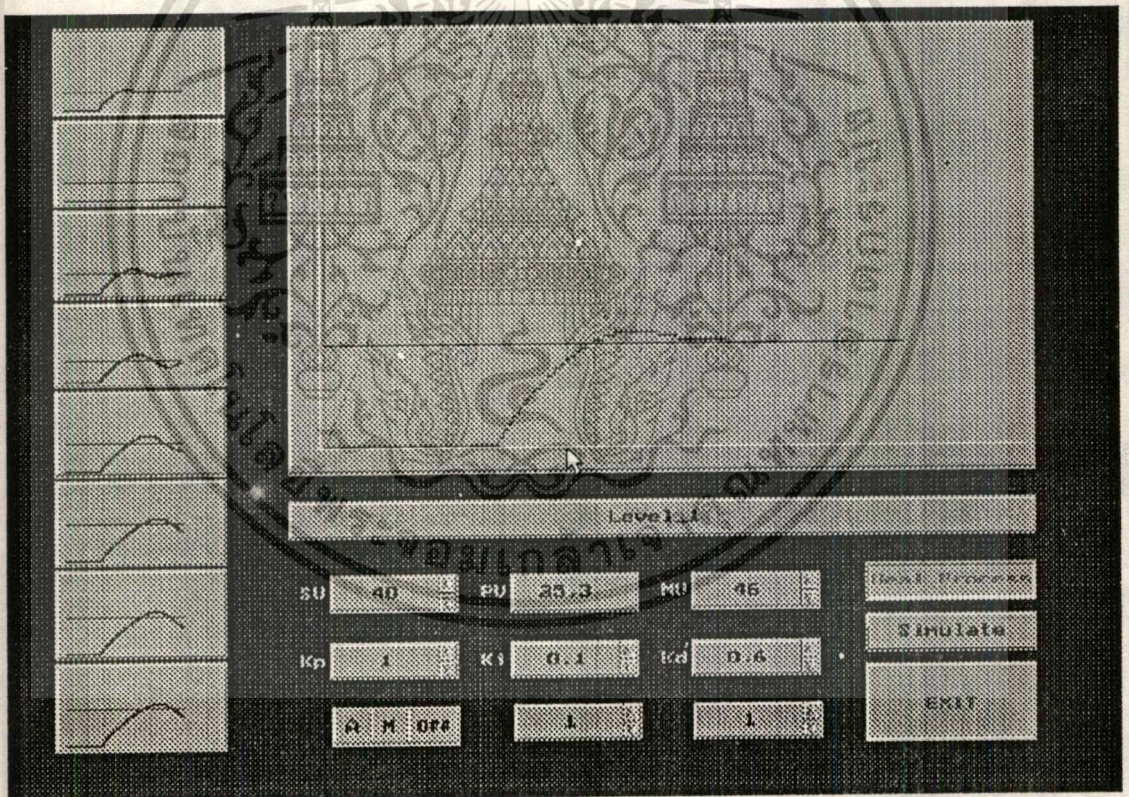
รูปที่.3 รูปแสดงหน้าจอขณะป้อนค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NO.	Description	SU	Kp	Ki	Kd	Headrange	Distrange	Status
1	Level_1	40	1	.1	.6	100	20	auto
2	Level_2	45	1	.2	.5	100	25	manual
3	Level_3	50	1	.3	.6	100	30	auto
4	Level_4	55	1	.4	.6	100	35	auto
5	Level_5	60	1	.5	.6	100	30	auto
6	Level_6	65	1	.6	.6	100	25	auto
7	Level_7	60	1	.7	.6	100	20	auto
8	Level_8	55	1	.8	.6	100	20	auto

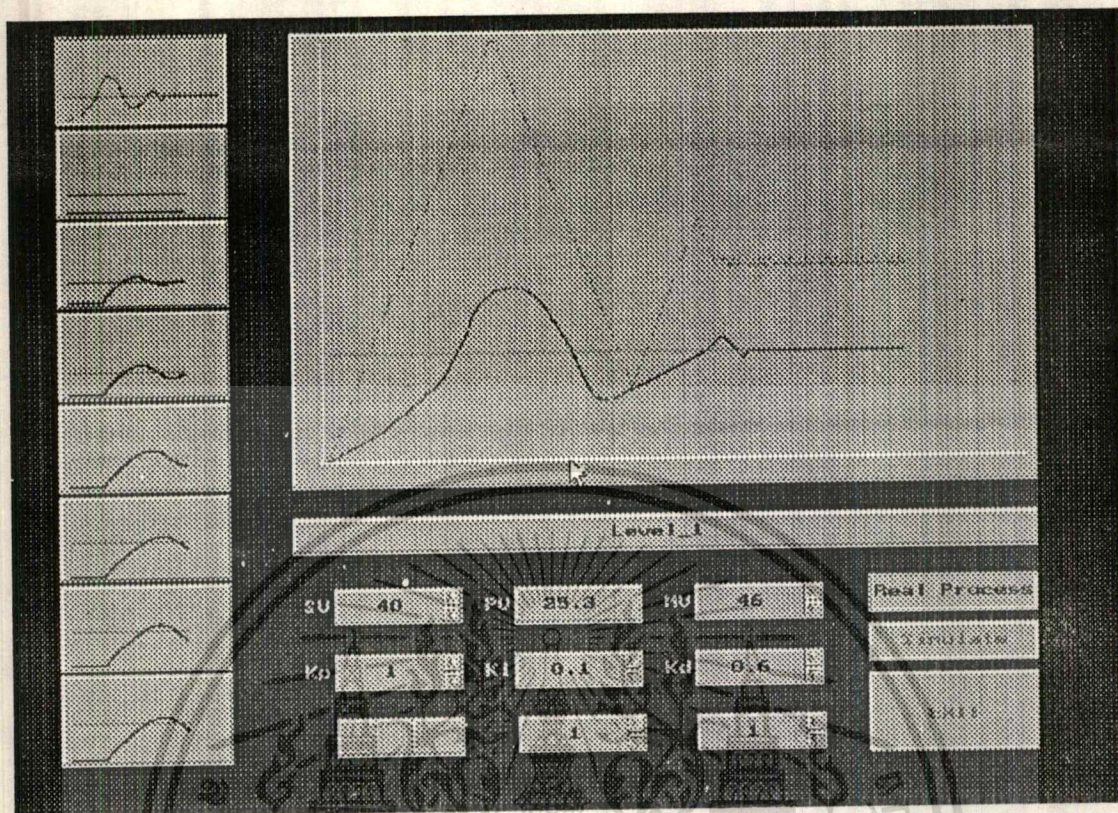
Finish enter values

รูปที่.4รูปแสดงตารางที่ป้อนค่าเสร็จแล้ว

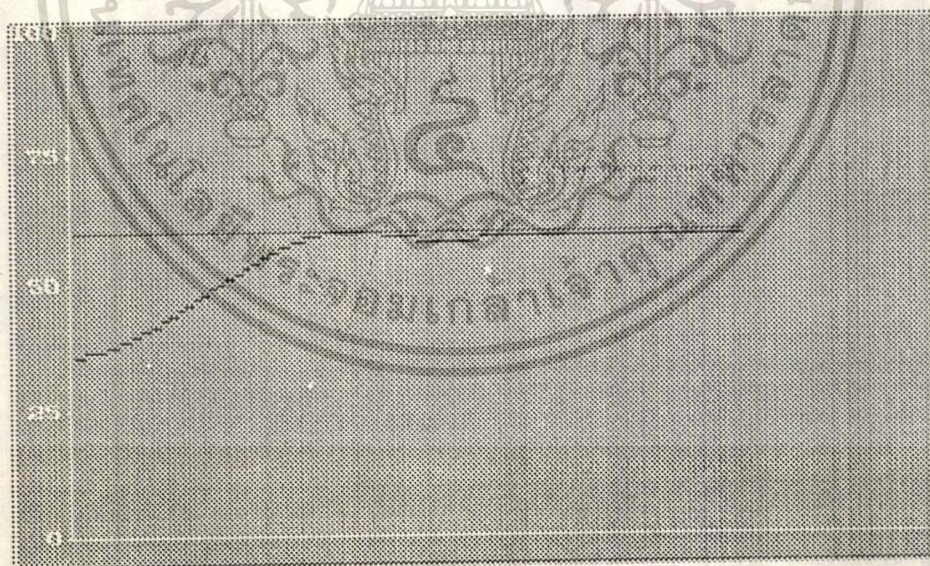


รูปที่.5รูปแสดงหน้าจอแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

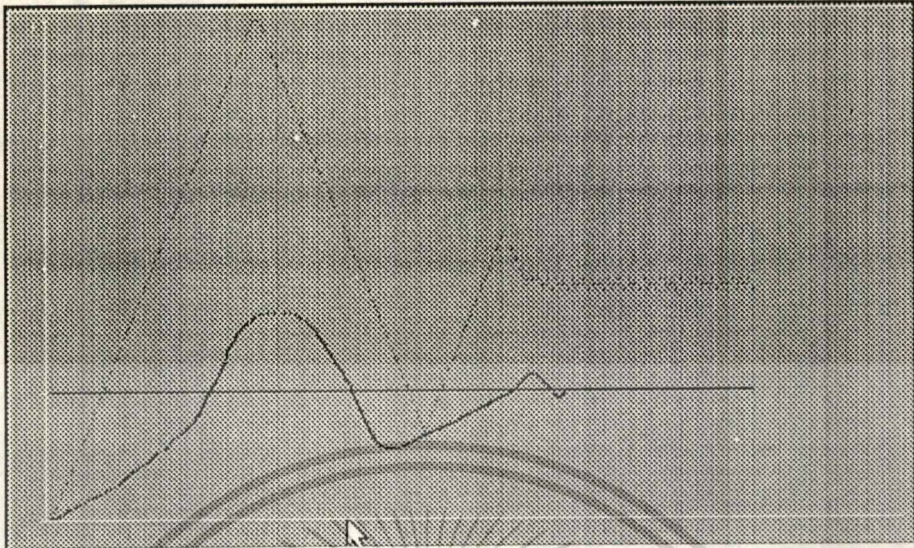


รูปที่.6 รูปแสดงหน้าจอแสดงผลขณะควบคุมโปรเซสจริง



รูปที่.7 กราฟแสดงผลการควบคุมที่ค่าเป้าหมายเมื่อ $K_p=1, K_i=0.1, K_d=0.6$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8 กราฟแสดงผลการควบคุมที่ค่าเป้าหมายเมื่อ $K_p=1, K_i=0.8, K_d=0.6$





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดเกี่ยวกับ ADDA CARD

บทนำ

เป็น ADDA CARD ขนาด 12 บิต ที่มีความแม่นยำสูงในการแปลงค่าข้อมูล การ์ดนี้ประกอบด้วย เอาร์ทพุต ดิจิตอลเป็นแอนาลอก 12 บิต 1 แชนแนล และ อินพุต แอนาลอกเป็นดิจิตอล ขนาด 12 บิต 16 แชนแนล

ข้อมูลจำเพาะ

- D/A
- ขนาด 12 บิต 1 แชนแนล
 - ขนาดช่วงเอาร์ทพุต 0-9 โวลต์ (ปรับโดยใช้ VR2)
 - ขั้วเดียวหรือสองขั้ว (เลือกโดย สวิตช์เลือก 1)
 - ใช้เวลาในการแปลงค่า 500 นาโนวินาที
 - ความไม่เป็นเชิงเส้น 0.2%
- A/D
- ขนาด 12 บิต 16 แชนแนล
 - ขนาดช่วงอินพุต 0-9 โวลต์
 - ขั้วเดียว
 - ใช้วิธีการประมาณค่าแบบ “ Successive Approximation Method ”
 - เวลาในการแปลงค่า 60 ไมโครวินาที
 - ตำแหน่งของ อินพุต เอาร์ทพุตพอร์ท : \$278 - \$27F or \$2F8 - \$2FF (เลือกโดย สวิตช์เลือก 3)

ขั้นตอนในการใช้งาน

- 1.) ตั้งค่าตำแหน่ง อินพุต เอาร์ทพุต โดยสวิตช์เลือก 3 (ตั้งไว้ที่ตำแหน่ง 1)
ตำแหน่ง 1 : ตั้งสวิตช์เลือก 3 ขึ้นข้างบน ตำแหน่ง \$278 - \$27F (632 - 639)
ตำแหน่ง 2 : ตั้งสวิตช์เลือก 3 ลงข้างล่าง ตำแหน่ง \$2F8 - \$2FF (760 - 767)
- 2.) ปรับระดับแรงดันด้วยค่าความต้านทาน : ปรับค่า A/D และ D/A แบบเต็มสเกลกับระดับอ้างอิง โดย ปรกติปรับที่ 9 โวลต์
- 3.) port = 632 (หรือ 760)
port +0 = เอาร์ทพุตข้อมูล (0-15) เพื่อเลือก A/D Channel
port +1 = อ่านข้อมูล A/D 8 bit ส่วนล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

port +2 = อ่านข้อมูล A/D 4 bit ส่วนบน (บิต0 - บิต3)

port +3 = เอาท์พุทเพื่อเคลียร์ A/D รีจิสเตอร์

port +4 = เรียก 7 ครั้งเพื่อเริ่มทำการแปลง A/D 6 บิตบน

port +5 = เรียก 7 ครั้งเพื่อเริ่มทำการแปลง A/D 6 บิตล่าง

port +6 = เอาท์พุท D/A 8 บิตข้อมูลล่าง

port +7 = เอาท์พุท D/A 4 บิตข้อมูลบน (บิต0 - บิต3)

4.) ขั้นตอนการแปลง แอนาลอกเป็น ดิจิตอล

ก.) ส่งค่าเซนแนลที่จะเลือก ไปที่ port +0

ข.) ตั้งค่ารีจิสเตอร์เริ่มต้น โดยการใช้ softswitch ที่ port +3 เพื่อลบข้อมูลก่อน

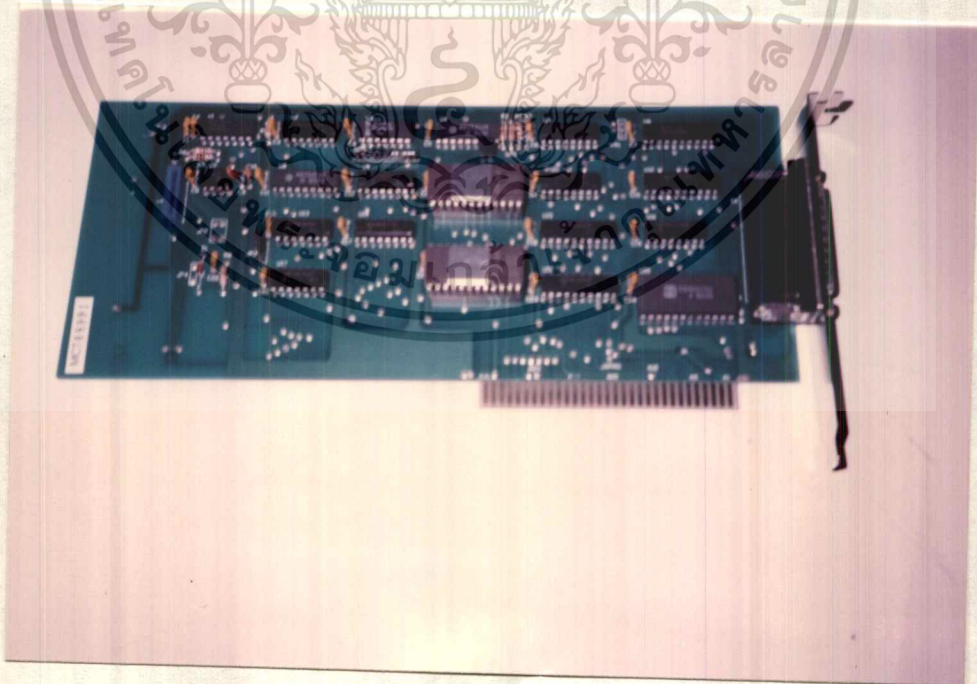
ค.) เริ่มทำการแปลงค่าข้อมูล โดยการใช้ softswitch ที่ port +4 , port +5 7 ครั้ง

ง.) อ่านข้อมูลที่ port +2 (บิต0 - บิต3) สำหรับข้อมูลบิตสูง 4 บิต และ port +1 สำหรับ 8 บิตข้อมูลล่าง

5.) ขั้นตอนการแปลง ดิจิตอลเป็นแอนาลอก

ก.) ส่งค่าเอาท์พุท 4 บิต สูงไปที่ port +7

ข.) ส่งค่าเอาท์พุท 8 บิต สูงไปที่ port +6



รูปแสดง ADDA CARD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D-TYPE CONNECTOR PINOUT

PIN	SIGNAL	PIN	SIGNAL
1	GND	14	GND
2	D/A out	15	-5V
3	5V	16	GND
4	GND	17	CH15
5	CH0	18	CH14
6	CH1	19	CH13
7	CH2	20	CH12
8	CH3	21	CH11
9	CH4	22	CH10
10	CH5	23	CH9
11	CH6	24	CH8
12	CH7	25	-12V
13	12V		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SOURCE PROGRAM

```

#include <graphics.h>
#include <stdlib.h>
#include <wall.h>
#include <ctype.h>
#include <thai.h>
#include <stdio.h>
#include <conio.h>
#include <math.h>
#include <dos.h>
#include <bios.h>
#include <string.h>
#include <alloc.h>
#include <time.h>
#include <process.h>
#include <graphs.h>
#include <mouse.h>

/*****

void    filltable(void);
void    description(int x,int y);
void    loopstatus(int x,int y);
void    value(int x,int y);
void    interruption(int x,int y);
void    ch_change(int ch);
void    fill_value(int color);
void    drawlittlegraph(int loop);
void    drawgraph(int ch);
void    pid(int loop);
void    del_graph(int ch);
void    ins_graph(int ch);
void    inputport(int loop);
void    simulate(int nn);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*****/

int    xx[10],yy[10],xxxx_run[9];
int    channel=1,loopno=1;
int    set_mode[9];
int    enablestatus[9];

float  mv_1[9],m[9];
int    sv100[9][422],mv100[9][422],pv100[9][422];
float  sv[9],pv[9],mv[9],maxi[9],mini[9],kp[9],ki[9],kd[9];
float  cr[9],cr_1[9],cr_2[9],scantime=0.1;
double h1[9],h2[9];
char   descript[9][16];
char   values[10][9][6];

/*****/

void main(void)
{
    time_t beginning[9],current[9];

// double    dif_time;
int          ii,mm;
int          row,col;
int          gdriver = DETECT,gmode;
int          xxxx,yyyy,ttt;
float        err1[9],err2[9],err3[9];

xxxx_run[1]=0;
xx[0]=15;    xx[2]=200;    xx[4]=300;    xx[6]=400;    xx[8]=560;
xx[1]=55;    xx[3]=250;    xx[5]=350;    xx[7]=480;    xx[9]=625;

for(ii=0;ii<10;ii++)yy[ii]=70+ii*20;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

initgraph(&gdriver,&gmode,"");

if(!initial_mouse())
{
    closegraph();
    printf("Mouse haven't been installed");
    exit(0);
}

mouse_range(5,5,635,475);
show();
tables();
cursor_on();
settextjustify(LEFT_TEXT, TOP_TEXT);
filltable();
if(enablestatus[1]+enablestatus[2]+enablestatus[3]+enablestatus[4]+
    enablestatus[5]+enablestatus[6]+enablestatus[7]+enablestatus[8]==0)
{
    closegraph();
    exit(0); //should have some images.
}

cursor_off();
show();
display();
presswindows(30,10,130,60);
fill_value(4);
for(ii=0;ii<9;ii++)beginning[ii]=time(0);
ins_graph(1);
fill_value(4);
presswindows(30,10,130,60);
cursor_on();
cursor_on();
start : loopno=1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

while(loopno<9)
{
//          if(enablestatus[loopno])
//          {
//              current[loopno]=time(0);
//              gotoxy(1,1);
//              printf("%d",loopno);
//              if(set_mode[loopno]==1)inputport(loopno);

// set_mode=1 ==> Real Processs
//              if(enablestatus[loopno]==1)pid(loopno);
//              if(set_mode[loopno]==1)outputport(loopno);
//              if(difftime(current[loopno],beginning[loopno]))
//              {
//                  for(mm=1;mm<9;mm++)
//                  {
//                      if((enablestatus[mm]!=0)&&(set_mode[mm]==0))
//                          simulate(mm);
//                  }
//                  if(channel==loopno)drawgraph(channel);
//                  drawlittlegraph(loopno);
//                  for(mm=1;mm<9;mm++)
//                  {
//                      if(mm!=channel)
//                          if(!!!!_run[mm]>421)
//                              !!!!_run[mm]=0;
//                  }
//                  beginning[loopno]=current[loopno];
//              }
//              if(left_click(&yyyy,&xxxx))
//              {
//                  cursor_off();
//                  interruption(yyyy,xxxx);
//                  cursor_on();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
}
loopno++;
}
goto start;
}

/*****

void filltable(void)
{
    int i,j;
    int a,b;
    int row,col;
    for(;;)
    {
        if(left_click(&row,&col))
            for(j=1;j<9;j++)
            {
                if((col<xx[2]&&(col>xx[1])&&(row<yy[j+1])&&(row>yy[j]))
                {
                    cursor_off();
                    presswindows(xx[1]+1,yy[j]+1,xx[2],yy[j+1]);
                    description(1,j);
                    windows(xx[1]+1,yy[j]+1,xx[2],yy[j+1]);
                    cursor_on();
                }
                else if((col<xx[9]&&(col>xx[8])&&(row<yy[j+1])&&(row>yy[j]))
                {
                    cursor_off();
                    presswindows(xx[8]+1,yy[j]+1,xx[9],yy[j+1]);
                    loopstatus(8,j);
                    windows(xx[8]+1,yy[j]+1,xx[9],yy[j+1]);
                    cursor_on();
                }
            }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
else for(i=2;i<8;i++)
if((col<xx[i+1])&&(col>xx[i])&&(row<yy[j+1])&&(row>yy[j]))
{
    cursor_off();

    presswindows(xx[i]+1,yy[j]+1,xx[i+1],yy[j+1]);

    value(i,j);

    windows(xx[i]+1,yy[j]+1,xx[i+1],yy[j+1]);

    cursor_on();
}
else if((col<625)&&(col>400)&&(row<290)&&(row>270))
{
    cursor_off();
    presswindows(400,270,625,290);
    delay(500);
    goto end;
}
}end : for(i=1;i<9;i++)
{
    sv[i]=atof(values[2][i]);
    kp[i]=atof(values[3][i]);
    ki[i]=atof(values[4][i]);
    kd[i]=atof(values[5][i]);
    maxi[i]=atof(values[6][i]);
    mini[i]=atof(values[7][i]);
}

}

/*****

void description(int x,int y)
{
    int row,col;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

char ch[2];
setcolor(7);
ch[0]='\0';
ch[1]='\0';
outtextxy(3+xx[x],8+yy[y],descript[y]);
*descript[y]='\0';
setcolor(4);
while(ch[0]!=0x20)
{
    if(strlen(descript[y])<16)
    {
        ch[0]=getch();
        ch[1]='\0';
        strcat(descript[y],ch);
        outtextxy(3+xx[x],8+yy[y],descript[y]);
    }
    else { ch[0]=getch();ch[1]='\0';}
}
ch[0]='\0';
ch[1]='\0';
}

/*****

void loopstatus(int x,int y)
{
    int k;
    char ch;
    setcolor(7);
    outtextxy(10+xx[x],yy[y]+8," auto");
    outtextxy(10+xx[x],yy[y]+8,"manual");
    outtextxy(12+xx[x],yy[y]+8," off");
    while(ch!=0x20)
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ch=getch();
if(ch!=0x20)
{
    ch=tolower(ch);
    switch(ch)
    {
        case 'a': enablestatus[y]=1;
            setcolor(7);
            outtextxy(12+xx[x],yy[y]+8," off");
            outtextxy(10+xx[x],yy[y]+8,"manual");
            setcolor(4);
            outtextxy(10+xx[x],yy[y]+8," auto");
            break;
        case 'o': enablestatus[y]=0;
            setcolor(7);
            outtextxy(10+xx[x],yy[y]+8," auto");
            outtextxy(10+xx[x],yy[y]+8,"manual");
            setcolor(4);
            outtextxy(12+xx[x],yy[y]+8," off");
            break;
        case 'm': enablestatus[y]=2;
            setcolor(7);
            outtextxy(12+xx[x],yy[y]+8," off");
            outtextxy(10+xx[x],yy[y]+8," auto");
            setcolor(4);
            outtextxy(10+xx[x],yy[y]+8,"manual");
            break;
    }
}
else {
    setcolor(4);
    if(enablestatus[y]==0)outtextxy(12+xx[x],yy[y]+8," off");
    if(enablestatus[y]==1)outtextxy(10+xx[x],yy[y]+8," auto");
    if(enablestatus[y]==2)outtextxy(10+xx[x],yy[y]+8,"manual");
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
}
}

/*****/

void value(int x,int y)
{
char ch[2];
setcolor(7);
ch[0]='\0';
ch[1]='\0';
outtextxy(xx[x+1]-8-8*strlen(values[x][y]),8+yy[y],values[x][y]);
*values[x][y]='\0';
setcolor(4);
while(ch[0]!=0x20)
{
if(strlen(values[x][y])<5)
{
ch[0]=getch();
ch[1]='\0';
if(isdigit(ch[0])||(ch[0]=='.'))
{
setcolor(7);
outtextxy(xx[x+1]-8-8*strlen(values[x][y]),8+yy[y],values[x][y]);
strcat(values[x][y],ch);
setcolor(4);
outtextxy(xx[x+1]-8-8*strlen(values[x][y]),8+yy[y],values[x][y]);
}
}
else{ ch[0]=getch();ch[1]='\0';}
}
ch[0]='\0';
ch[1]='\0';

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

/*****/

void runprogram(void)
{
//drawlittlegraph()      loopno
//pid()                  loopno
//drawgraph()            channel
}

/*****/

void interruption(int y,int x)
{
if((x>30)&&(x<130))
{
if((y>10)&&(y<60))if(channel!=1)ch_change(1);    //#1-8
if((y>65)&&(y<115))if(channel!=2)ch_change(2);
if((y>120)&&(y<170))if(channel!=3)ch_change(3);
if((y>175)&&(y<225))if(channel!=4)ch_change(4);
if((y>230)&&(y<280))if(channel!=5)ch_change(5);
if((y>285)&&(y<335))if(channel!=6)ch_change(6);
if((y>340)&&(y<390))if(channel!=7)ch_change(7);
if((y>395)&&(y<445))if(channel!=8)ch_change(8);
}
if((x>261)&&(x<270))
{
if((y>345)&&(y<355))
{
//          value_change(1,1,1);          sv+
presswindows(261,345,270,354);
setcolor(7);
outtextxy(227,355,values[2][channel]);
}
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sv[channel]++;
if(sv[channel]>100)sv[channel]=100;
delay(20);
gcvt(sv[channel],3,values[2][channel]);
setcolor(4);
outtextxy(227,355,values[2][channel]);
windows(261,345,270,354);
}
if((y>356)&&(y<365))
{
// value_change(1,1,2); sv-
presswindows(261,355,270,364);
setcolor(7);
outtextxy(227,355,values[2][channel]);
sv[channel]--;
if(sv[channel]<0)sv[channel]=0;
delay(20);
gcvt(sv[channel],3,values[2][channel]);
setcolor(4);
outtextxy(227,355,values[2][channel]);
windows(261,355,270,364);
}
if((y>385)&&(y<395))
{
// value_change(1,2,1); kp+
presswindows(261,385,270,394);
setcolor(7);
outtextxy(227,395,values[3][channel]);
kp[channel]=kp[channel]+0.1;
if(kp[channel]>100)kp[channel]=100;
delay(20);
gcvt(kp[channel],3,values[3][channel]);
setcolor(4);
outtextxy(227,395,values[3][channel]);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        windows(261,385,270,394);
    }
    if((y>396)&&(y<405))
    {
//        value_change(1,2,2);           kp-
        presswindows(261,395,270,404);
        setcolor(7);
        outtextxy(227,395,values[3][channel]);
        kp[channel]=kp[channel]-0.1;
        if(kp[channel]<0.1)kp[channel]=0;
        delay(20);
        gcvt(kp[channel],3,values[3][channel]);
        setcolor(4);
        outtextxy(227,395,values[3][channel]);
        windows(261,395,270,404);
    }
}
if((x>371)&&(x<380))
{
    if((y>385)&&(y<395))
    {
//        value_change(2,2,1);           ki+
        presswindows(371,385,380,394);
        setcolor(7);
        outtextxy(337,395,values[4][channel]);
        ki[channel]=ki[channel]+0.1;
        if(ki[channel]>100)ki[channel]=100;
        delay(20);
        gcvt(ki[channel],3,values[4][channel]);
        setcolor(4);
        outtextxy(337,395,values[4][channel]);
        windows(371,385,380,394);
    }
}
if((y>396)&&(y<405))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
//      value_change(2,2,2);          ki-
      presswindows(371,395,380,404);
      setcolor(7);
      outtextxy(337,395,values[4][channel]);
      ki[channel]=ki[channel]-0.1;
      if(ki[channel]<0.1)ki[channel]=0;
      delay(20);
      gcvt(ki[channel],3,values[4][channel]);
      setcolor(4);
      outtextxy(337,395,values[4][channel]);
      windows(371,395,380,404);
}
if((y>425)&&(y<435))
{
//      value_change(2,3,1);          maxi+
      presswindows(371,425,380,434);
      setcolor(7);
      outtextxy(337,435,values[6][channel]);
      maxi[channel]++;
      if(maxi[channel]>100)maxi[channel]=100;
      delay(20);
      gcvt(maxi[channel],3,values[6][channel]);
      setcolor(4);
      outtextxy(337,435,values[6][channel]);
      windows(371,425,380,434);
}
if((y>436)&&(y<445))
{
//      value_change(2,3,2);          maxi-
      presswindows(371,435,380,444);
      setcolor(7);
      outtextxy(337,435,values[6][channel]);
      maxi[channel]--;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if(maxi[channel]<0)maxi[channel]=0;
        delay(20);
        gcvt(maxi[channel],3,values[6][channel]);
        setcolor(4);
        outtextxy(337,435,values[6][channel]);
        windows(371,435,380,444);
    }
}
if((x>481)&&(x<490))
{
    if((y>345)&&(y<355))if(enablestatus[channel]==2)
    {
//      value_change(3,1,1);          mv+
        presswindows(481,345,490,354);
        setcolor(7);
        outtextxy(447,355,values[8][channel]);
        mv[channel]++;
        if(mv[channel]>100)mv[channel]=100;
        delay(20);
        gcvt(mv[channel],3,values[8][channel]);
        setcolor(4);
        outtextxy(447,355,values[8][channel]);
        windows(481,345,490,354);
    }
    if((y>356)&&(y<365))if(enablestatus[channel]==2)
    {
//      value_change(3,1,2);          mv-
        presswindows(481,355,490,364);
        setcolor(7);
        outtextxy(447,355,values[8][channel]);
        mv[channel]--;
        if(mv[channel]<0)mv[channel]=0;
        delay(20);
        gcvt(mv[channel],3,values[8][channel]);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        setcolor(4);
        outtextxy(447,355,values[8][channel]);
        windows(481,355,490,364);
    }
    if((y>385)&&(y<395))
    {
//        value_change(3,2,1);           kd+
        presswindows(481,385,490,394);
        setcolor(7);
        outtextxy(447,395,values[5][channel]);
        kd[channel]=kd[channel]+0.1;
        if(kd[channel]>100)kd[channel]=100;
        delay(20);
        gcvt(kd[channel],3,values[5][channel]);
        setcolor(4);
        outtextxy(447,395,values[5][channel]);
        windows(481,385,490,394);
    }
    if((y>396)&&(y<405))
//    {
        value_change(3,2,2);           kd-
        presswindows(481,395,490,404);
        setcolor(7);
        outtextxy(447,395,values[5][channel]);
        kd[channel]=kd[channel]-0.1;
        if(kd[channel]<0.1)kd[channel]=0;
        delay(20);
        gcvt(kd[channel],3,values[5][channel]);
        setcolor(4);
        outtextxy(447,395,values[5][channel]);
        windows(481,395,490,404);
    }
    if((y>425)&&(y<435))
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//          value_change(3,3,1);          mini+
          presswindows(481,425,490,434);
          setcolor(7);
          outtextxy(447,435,values[7][channel]);
          mini[channel]++;
          if(mini[channel]>100)mini[channel]=100;
          delay(20);
          gcvt(mini[channel],3,values[7][channel]);
          setcolor(4);
          outtextxy(447,435,values[7][channel]);
          windows(481,425,490,434);
        }
if((y>436)&&(y<445))
{
//          value_change(3,3,2);          mini-
          presswindows(481,435,490,444);
          setcolor(7);
          outtextxy(447,435,values[7][channel]);
          mini[channel]--;
          if(mini[channel]<0)mini[channel]=0;
          delay(20);
          gcvt(mini[channel],3,values[7][channel]);
          setcolor(4);
          outtextxy(447,435,values[7][channel]);
          windows(481,435,490,444);
        }
    }
if((y>425)&&(y<445))
{
    if((x>195)&&(x<217))
    {
        presswindows(195,425,217,445);          // auto
        windows(218,425,240,445);
        windows(241,425,270,445);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

setcolor(8);
enablestatus[channel]=1;
outtextxy(255,435,"Off");
outtextxy(229,435,"M");
setcolor(4);
outtextxy(206,435,"A");
delay(20);
windows(195,425,217,445);
}
if((x>218)&&(x<240))
{
presswindows(218,425,240,445); //manual
windows(195,425,217,445);
windows(241,425,270,445);
setcolor(8);
enablestatus[channel]=2;
outtextxy(255,435,"Off");
outtextxy(206,435,"A");
setcolor(4);
outtextxy(229,435,"M");
delay(20);
windows(218,425,240,445);
}
if((x>241)&&(x<270))
{
presswindows(241,425,270,445); // off
windows(195,425,217,445);
windows(218,425,240,445);
setcolor(8);
enablestatus[channel]=0;
outtextxy(229,435,"M");
outtextxy(206,435,"A");
setcolor(4);
outtextxy(255,435,"Off");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        delay(20);
        windows(241,425,270,445);
    }
}
if((x>520)&&(x<620))
{
    if((y>400)&&(y<445))
    {
        presswindows(520,400,620,445);
//        sound(346);
//        delay(500);
//        nosound();
//        sound(400);
//        delay(500);
//        nosound();
//        sound(450);
//        delay(500);
//        nosound();
//        sound(600);
//        delay(500);
//        nosound();
        exit(0);
    }
    if((y>340)&&(y<360))
    {
        if(set_mode[1]+set_mode[2]+set_mode[3]+set_mode[4]+
            set_mode[5]+set_mode[6]+set_mode[7]+set_mode[8]==0)
        {
            setcolor(4);
            outtextxy(570,350,"Real Process");
            setcolor(8);
            set_mode[channel]=1;
            outtextxy(570,380,"Simulate");
            presswindows(520,340,620,360);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        windows(520,370,620,390);
    }
}
if((y>370)&&(y<390))
{
    setcolor(4);
    outtextxy(570,380,"Simulate");
    setcolor(8);
    set_mode[channel]=0;
    outtextxy(570,350,"Real Process");
    presswindows(520,370,620,390);
    windows(520,340,620,360);
}
}
}

/*****

void ch_change(int ch)
{
    fill_value(7);
    windows(30,10+(channel-1)*55,130,60+(channel-1)*55);
    del_graph(channel);
    channel=ch;
    presswindows(30,10+(channel-1)*55,130,60+(channel-1)*55);
    ins_graph(channel);
    fill_value(4);
}

/*****

void value_change(int ax,int ay,int op)
{
    presswindows(261+(ax-1)*110,345+(ay-1)*40+(op-1)*10,

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        270+(ax-1)*110,354+(ay-1)*40+(op-1)*10);
delay(20);
windows(261+(ax-1)*110,345+(ay-1)*40+(op-1)*10,
        270+(ax-1)*110,354+(ay-1)*40+(op-1)*10);
}

/*****/

void fill_value(int color)
{
    //jam
    setcolor(color);
    gcvt(mv[channel],3,values[8][channel]);outtextxy(447,355,values[8][channel]);
    gcvt(pv[channel],3,values[9][channel]);outtextxy(337,355,values[9][channel]);
    gcvt(sv[channel],3,values[2][channel]);outtextxy(227,355,values[2][channel]);
    gcvt(kp[channel],3,values[3][channel]);outtextxy(227,395,values[3][channel]);
    gcvt(ki[channel],3,values[4][channel]);outtextxy(337,395,values[4][channel]);
    gcvt(kd[channel],3,values[5][channel]);outtextxy(447,395,values[5][channel]);
    gcvt(maxi[channel],3,values[6][channel]);outtextxy(337,435,values[6][channel]);
    gcvt(mini[channel],3,values[7][channel]);outtextxy(447,435,values[7][channel]);
    if(enablestatus[channel]==0)
    {
        setcolor(8);
        outtextxy(206,435,"A");
        outtextxy(229,435,"M");
        setcolor(4);
        outtextxy(255,435,"Off");
    }

    if(enablestatus[channel]==1) //1=auto ;2=manual ;0=off
    {
        setcolor(8);
        outtextxy(255,435,"Off");
        outtextxy(229,435,"M");
        setcolor(4);
        outtextxy(206,435,"A");
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    if(enablestatus[channel]==2)
    {
        setcolor(8);
        outtextxy(206,435,"A");
        outtextxy(255,435,"Off");
        setcolor(4);
        outtextxy(229,435,"M");
    }
    if(set_mode[channel]==1)
    {
        setcolor(4);
        outtextxy(570,350,"Real Process");
        setcolor(8);
        outtextxy(570,380,"Simulate");
        presswindows(520,340,620,360);
        windows(520,370,620,390);
    }
    if(set_mode[channel]==0)
    {
        setcolor(4);
        outtextxy(570,380,"Simulate");
        setcolor(8);
        outtextxy(570,350,"Real Process");
        windows(520,340,620,360);
        presswindows(520,370,620,390);
    }
}

/*****/

void drawlittlegraph(int loop)
{
    int t;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(xxxx_run[loop]>420)
{
    for(t=0;t<421;t++)
    {
        putpixel(35+t/5,3+loop*55-0.4*sv100[loop][t-1],7);
        putpixel(35+t/5,3+loop*55-0.4*pv100[loop][t-1],7);
        putpixel(35+t/5,3+loop*55-0.4*mv100[loop][t-1],7);
    }
}

// gotoxy(3,loop+5);
// printf("%d %d ",loop,xxxx_run[loop]);
// putpixel(35+xxxx_run[loop]/5,3+loop*55-0.4*mv100[loop][xxxx_run[loop]-1],LIGHTGREEN);
putpixel(35+xxxx_run[loop]/5,3+loop*55-0.4*sv100[loop][xxxx_run[loop]-1],LIGHTMAGENTA);
putpixel(35+xxxx_run[loop]/5,3+loop*55-0.4*pv100[loop][xxxx_run[loop]-1],LIGHTBLUE);
if(xxxx_run[loop]==421){
    putpixel(35+xxxx_run[loop]/5,3+loop*55-0.4*sv100[loop][xxxx_run[loop]-1],7);
    putpixel(35+xxxx_run[loop]/5,3+loop*55-0.4*pv100[loop][xxxx_run[loop]-1],7);
}
}

/*****

void drawgraph(int ch)
{
    int t,tt;
    for(t=1;t<9;t++)
        if(enablestatus[t])
        {
            sv100[t][xxxx_run[t]]=(int)(100*(sv[t]-mini[t])/(maxi[t]-mini[t]));
            mv100[t][xxxx_run[t]]=(int)(100*(mv[t]-mini[t])/(maxi[t]-mini[t]));
            pv100[t][xxxx_run[t]]=(int)(100*(pv[t]-mini[t])/(maxi[t]-mini[t]));
        }
    if(xxxx_run[ch]>420)
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for(t=0;t<421;t++)
{
    putpixel(190+t,265-2.5*pv100[ch][t],7);
    putpixel(190+t,265-2.5*mv100[ch][t],7);
    putpixel(190+t,265-2.5*sv100[ch][t],7);
}
xxxx_run[ch]=0;
}

putpixel(190+xxxx_run[ch],265-2.5*mv100[ch][xxxx_run[ch]],LIGHTGREEN);
putpixel(190+xxxx_run[ch],265-2.5*sv100[ch][xxxx_run[ch]],LIGHTMAGENTA);
putpixel(190+xxxx_run[ch],265-2.5*pv100[ch][xxxx_run[ch]],LIGHTBLUE);

for(t=1;t<9;t++)
if(enablestatus[t])
{
    setcolor(7);
    outtextxy(447,355,values[8][channel]);
    outtextxy(337,355,values[9][channel]);
    xxxx_run[t]++;
    gcvt(mv[channel],3,values[8][channel]);
    gcvt(pv[channel],3,values[9][channel]);
    setcolor(4);
    outtextxy(447,355,values[8][channel]);
    outtextxy(337,355,values[9][channel]);
}
}

/*****/

void del_graph(int ch)
{
    int t;
    for(t=0;t<xxxx_run[ch];t++)
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        putpixel(190+t,265-2.5*pv100[ch][t],7);
        putpixel(190+t,265-2.5*mv100[ch][t],7);
        putpixel(190+t,265-2.5*sv100[ch][t],7);
    }
}

/*****/

void ins_graph(int ch)
{
    int t;
    for(t=0;t<xxxx_run[ch];t++)
    {
        putpixel(190+t,265-2.5*mv100[ch][t],LIGHTGREEN);
        putpixel(190+t,265-2.5*sv100[ch][t],LIGHTMAGENTA);
        putpixel(190+t,265-2.5*pv100[ch][t],LIGHTBLUE);
    }
}

/*****/

void inputport(int loop)
{
}

/*****/

void simulate(int nn)
{
    // if(h1[nn][1]>100)h1[nn][1]=100;
    // if(h2[nn][1]>100)h2[nn][1]=100;
    // if(h1[nn][1]<0)h1[nn][1]=0;
    // if(h2[nn][1]<0)h2[nn][1]=0;

    h1[nn]=h1[nn]+(mv[nn]*5-64.4*sqrt(h1[nn]))*0.1/61.1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

h2[nn]=h2[nn]+(64.4*sqrt(h1[nn])-46.08*sqrt(h2[nn]))*0.1/61.1;
if(h1[nn]>100)h1[nn]=100;
if(h2[nn]>100)h2[nn]=100;
if(h1[nn]<0.1)h1[nn]=0;
if(h2[nn]<0.1)h2[nn]=0;
pv[nn]=h2[nn];
}

/*****

void pid(int loop)
{
float k1=0,k2=0,k3=0;
// static float m[9];

k1=kp[loop]*(1+ki[loop]*scantime)+kd[loop]/scantime;
k2=(-kp[loop])*(1+2*kd[loop]/scantime);
k3=kp[loop]*kd[loop]/scantime;
er[loop]=((sv[loop]-mini[loop])/(maxi[loop]-mini[loop]))*100-pv[loop];
mv[loop]=mv_1[loop]+k1*er[loop]+k2*er_1[loop]+k3*er_2[loop];
if(mv[loop]>m[loop]) mv[loop]=m[loop]+1;
if(mv[loop]<m[loop]) mv[loop]=m[loop]-1;
if(mv[loop]>100)mv[loop]=100;
if(mv[loop]<0)mv[loop]=0;
mv_1[loop]=mv[loop];
m[loop]=mv[loop];
er_2[loop]=er_1[loop];
er_1[loop]=er[loop];
}

/*****

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*****/

void inputport(int loop)
{
  int PORT=632;
  int s, b,c;
  outportb(PORT + 3, 0x00);
  delay(10);
  outportb(PORT + 0, loop);
  delay(10);
  for(s=1; s<=7; s++)
  {
    inportb(PORT + 4);
    inportb(PORT + 5);
    delay(10);
  }
  b = inportb(PORT + 2);
  delay(10);
  c = inportb(PORT + 1);
  d = b;
  d = ((b&0x0F)<<8) + c;
  mv[loop]=((d/4095.)/4) * 100;
}

/*****/

END OF PROGRAM

/*****/

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INCLUDE FILE

```
#define graphs.h
```

```
#include <graphics.h>
```

```
/**/
```

```
void windows(int a,int b,int c,int d);
```

```
void presswindows(int a,int b,int c,int d);
```

```
void box(int a,int b,int c,int d,int color);
```

```
void fillbox(int a,int b,int c,int d,int color);
```

```
void display(void);
```

```
void tables(void);
```

```
void up_tri(int a,int b,int c,int d,int e,int f);
```

```
void down_tri(int a,int b,int c,int d,int e,int f);
```

```
/**/
```

```
void presswindows(int a,int b,int c,int d)
```

```
{
```

```
    setcolor(8);
```

```
    line(a,b,a,d);
```

```
    line(a,b,c,b);
```

```
    setcolor(15);
```

```
    line(c,b,c,d);
```

```
    line(a,d,c,d);
```

```
}
```

```
/**/
```

```
void windows(int a,int b,int c,int d)
```

```
{
```

```
    setcolor(15);
```

```
    line(a,b,a,d);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

line(a,b,c,b);
setcolor(8);
line(c,b,c,d);
line(a,d,c,d);
}

```

```

/*****

```

```

void box(int a,int b,int c,int d,int color)

```

```

{
setcolor(color);
line(a,b,c,b);
line(c,b,c,d);
line(c,d,a,d);
line(a,d,a,b);
}

```

```

/*****

```

```

void fillbox(int a,int b,int c,int d,int color)

```

```

{
setcolor(color);
line(a,b,c,b);
line(c,b,c,d);
line(c,d,a,d);
line(a,d,a,b);
setfillstyle(1,color);
floodfill(a+1,b+1,color);
}

```

```

/*****

```

```

void display(void)

```

```

{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int a,b,c,d;
setcolor(BLACK);
line(0,0,0,480);
line(1,0,1,480);
line(2,0,2,480);
line(638,0,638,480);
line(639,0,639,480);
for(a=0;a<8;a++)
{
    fillbox(29,9+a*55,131,61+a*55,LIGHTGRAY);
    windows(30,10+a*55,130,60+a*55);
}
fillbox(169,9,621,281,LIGHTGRAY);
windows(170,10,620,280);
fillbox(169,299,621,321,LIGHTGRAY);
windows(170,300,620,320);
for(a=0;a<3;a++)
for(b=0;b<3;b++)
{
    fillbox(194+a*110,344+b*40,271+a*110,365+b*40,LIGHTGRAY);
    windows(195+a*110,345+b*40,260+a*110,364+b*40);
    windows(261+a*110,345+b*40,270+a*110,354+b*40);
    windows(261+a*110,356+b*40,270+a*110,364+b*40);
    up_tri(265+a*110,347+b*40,262+a*110,353+b*40,268+a*110,353+b*40);
    down_tri(265+a*110,363+b*40,268+110*a,357+b*40,262+a*110,357+b*40);
}
fillbox(304,344,381,366,GREEN);
fillbox(304,344,381,366,LIGHTGRAY);
windows(305,345,380,365);
fillbox(194,424,271,446,GREEN);
fillbox(194,424,271,446,LIGHTGRAY);
windows(195,425,217,445);
windows(218,425,240,445);
windows(241,425,270,445);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

setcolor(LIGHTCYAN);
settextjustify(RIGHT_TEXT,CENTER_TEXT);
outtextxy(192,355,"SV");
outtextxy(302,355,"PV");
outtextxy(412,355,"MV");
outtextxy(192,395,"Kp");
outtextxy(302,395,"Ki");
outtextxy(412,395,"Kd");
outtextxy(302,435,"Max");
outtextxy(412,435,"Min");
fillbox(519,399,621,446,LIGHTGRAY);
fillbox(519,339,621,361,LIGHTGRAY);
fillbox(519,369,621,391,LIGHTGRAY);
windows(520,400,620,445);
windows(520,340,620,360);
windows(520,370,620,390);
settextjustify(CENTER_TEXT,CENTER_TEXT);
setcolor(8);
outtextxy(570,350,"Real Process");
outtextxy(570,380,"Simulate");
outtextxy(570,423,"EXIT");
setcolor(RED);
outtextxy(320,460,"King Mongkut's Institute of Technology");
outtextxy(320,472,"L A D K R A B A N G");
setcolor(WHITE);
line(188,15,188,266);
line(186,266,615,266);
line(186,15,188,15);
line(186,140,188,140);
line(186,77,188,77);
line(186,202,188,202);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*****

```

```

void tables(void)

```

```

{

```

```

    int i,j;

```

```

    int ver[9],hor[9];

```

```

    ver[0]=15;      ver[2]=200;      ver[4]=300;      ver[6]=400;      ver[8]=560;

```

```

    ver[1]=55;      ver[3]=250;      ver[5]=350;      ver[7]=480;      ver[9]=625;

```

```

    setcolor(BLACK);

```

```

    line(0,0,0,479);

```

```

    line(1,0,1,479);

```

```

    line(2,0,2,479);

```

```

    line(3,0,3,479);

```

```

    line(638,0,638,479);

```

```

    line(639,0,639,479);

```

```

    settxtjustify(CENTER_TEXT,CENTER_TEXT);

```

```

    fillbox(15,70,626,251,LIGHTGRAY);

```

```

    for(i=0;i<9;i++)

```

```

    for(j=0;j<9;j++)

```

```

    windows(ver[j]+1,71+20*i,ver[j+1],90+20*i);

```

```

    outtextxy(35,81,"N0.");

```

```

    outtextxy(122.5,81,"Description");

```

```

    outtextxy(225,81,"SV");

```

```

    outtextxy(275,81,"Kp");

```

```

    outtextxy(325,81,"Ki");

```

```

    outtextxy(375,81,"Kd");

```

```

    outtextxy(440,81,"MaxRange");

```

```

    outtextxy(520,81,"MinRange");

```

```

    outtextxy(592.5,81,"Status");

```

```

    outtextxy(35,101,"1");

```

```

    outtextxy(35,121,"2");

```

```

    outtextxy(35,141,"3");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

outtextxy(35,161,"4");
outtextxy(35,181,"5");
outtextxy(35,201,"6");
outtextxy(35,221,"7");
outtextxy(35,241,"8");
fillbox(399,269,626,291,LIGHTGRAY);
windows(400,270,625,290);
outtextxy(512,280,"Finish enter values");
// settextrjust(LEFT_TEXT,TOP_TEXT);
)

/*****/

void up_tri(int a,int b,int c,int d,int e,int f)
{
setcolor(8);
line(a,b,c,d);
setcolor(15);
line(c,d,e,f);
line(e,f,a,b);
}

/*****/

void down_tri(int a,int b,int c,int d,int e,int f)
{
setcolor(15);
line(a,b,c,d);
setcolor(8);
line(c,d,e,f);
line(e,f,a,b);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#define wall.h

#include <stdio.h>
#include <graphics.h>
#include <alloc.h>
#include <dos.h>

struct{
    unsigned char Password;
    unsigned char Version;
    unsigned char Encoding;
    unsigned char BPP;
    int WinDemX1;
    int WinDemY1;
    int WinDemX2;
    int WinDemY2;
    int HoriResolut;
    int VertResolut;
    unsigned char ColorMap[16][3];
    unsigned char reserve;
    unsigned char NumofPlane;
    int BytePLine;
    int PaletteInfo;
} Header;

//unsigned char get_display_mode();
//unsigned char get_BIOS_display_mode();
FILE *filepcx;
int MAXLINE;
int MAXCOLUMN;
int MaxLoop,linecount;
//char index,arg2[3],zoomvalue[2];.
unsigned ADDSCRN,Offset;
unsigned char zoom,Value,counter,plane;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

unsigned int column,movebyte,ShiftInLine,dropcount,drope;
//unsigned char PALETTE[16];
int start_hori_offset,start_vert_offset,ii=0;
//extern unsigned char far * pt;
//unsigned char far * end_page;
void Enable_Plane(unsigned char Plane);
void Put40;
void show();
void Show_4_Plane();

int procmain(int xx,int yy,char *name)
{
    start_hori_offset = yy;
    start_vert_offset = xx;
    filepcx = fopen(name,"rb");
    if(filepcx == NULL)
        return (-1);
    fread(&Header,70,1,filepcx); /*read Header file */
    if(Header.Password != 0x0a)
        return (-2);
        MAXLINE=480;
        MAXCOLUMN=80;
        ADDSCRN = 0xa000;

    if((Header.BytePLine+start_vert_offset) < MAXCOLUMN)
    {
        movebyte = Header.BytePLine;
        drope = 0;
    }
    else
    {
        movebyte = MAXCOLUMN - start_vert_offset;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

drope = Header.BytePLine - MAXCOLUMN + start_vert_offset;
    }

    if( (Header.WinDemY2+1+start_hori_offset) >= MAXLINE)
        MaxLoop = MAXLINE - start_hori_offset;
    else
        MaxLoop = Header.WinDemY2 + 1;

    start_hori_offset = start_hori_offset * MAXCOLUMN;
    fseek(filepcx,128,SEEK_SET);
        Show_4_Plane(); /* B/W color */
    fclose(filepcx);
    return (0);
}

void show()
{
    int size,x,i,j;
    void far *pic;
    procmain(0,0,"brick.pcx");
    size = imagesize(0,0,63,31);
    pic = farmalloc(size);
    getimage(0,0,Header.WinDemX2,Header.WinDemY2,pic);
    for(j=0;j<20;j++)
    for(i=0;i<20;i++)
    putimage(i*32,j*32,pic,COPY_PUT);
    farfree(pic);
}

void Show_4_Plane()
{
    int i;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Offset = start_hori_offset + start_vert_offset;
linecount = 0;
column = 0;
plane=1;
Enable_Plane(plane);
dropcount = 0;
fread(&Value,1,1,filepcx);
while ((!feof(filepcx)) && ((MaxLoop) >= linecount))
{
    if (Value > 0xc0)
    {
        counter = (Value^0xc0);
        fread(&Value,1,1,filepcx);
        for(i=counter;i>0;i--)
            Put4();
    }
    else
        Put4();
    fread(&Value,1,1,filepcx);
}
}

void Put4()
{
    if(dropcount==0)
    {
        pokeb(ADDSCRN,Offset,Value);
        Offset++;
        column++;
        if( column==movebyte )
        {
            plane = plane << 1;
            if(plane==16)
            {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    plane = 1;
    linecount++;
}
Offset = (linecount*80)+start_hori_offset
        +start_vert_offset;
Enable_Plane(plane);
column = 0;
dropcount = drope;
}
if (linecount==(Header.WinDemY2+1))
return;
}
else
dropcount--;
}
void Enable_Plane(unsigned char Plane)
{
outportb(0x3c4,2);
outportb(0x3c5,Plane);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#define thai.h

#include <graphics.h>
#include <string.h>
#include <stdio.h>

void drawfont(int x,int y,int color,int inchar);
void touttextxy(int x,int y,int color,unsigned char St[]);
char Fontpic[256][20];
int loadfont(char filename[]);

int loadfont(char filename[])
{
    FILE *tfile;
    if((tfile=fopen(filename,"rb"))==NULL)
        return 0;
    fread(&Fontpic,5120,1,tfile);
    return 1;
}

void drawfont(int x,int y,int color,int inchar)
{
    int i,j;
    for(i=0;i<20;i++)
    {
        for(j=0;j<8;j++)
        {
            if(((Fontpic[inchar][i]>>(7-j))&1))
                putpixel(x+j,y+i,color);
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void touttextxy(int x,int y,int color,unsigned char St[])
{
    int count=0;
    int a=0;
    int test;
    while((St[count]!=0)&(count<128))
    {
        test =(int)St[count];
        if(((test==209)||((test>211)&&(test<220))||((test>230)&&(test<239))))
            drawfont(x+a-8,y,color,test);
        else
        {
            drawfont(x+a,y,color,test);
            a+=8;
        }
        count++;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define mouse.h

#include <dos.h>

int    initial_mouse(void);
void   cursor_on(void);
void   cursor_off(void);
int    left_click(int *row,int *col);
int    right_click(int *row,int *col);
void   mouse_range(int left,int top,int right,int bottom);
```

```
int initial_mouse(void)
{
    union REGS regs;
    regs.x.ax=0;
    int86(0x33,&regs,&regs);
    return regs.x.ax;
}
```

```
void cursor_on(void)
{
    union REGS regs;
    regs.x.ax=1;
    int86(0x33,&regs,&regs);
}
```

```
void cursor_off(void)
{
    union REGS regs;
    regs.x.ax=2;
    int86(0x33,&regs,&regs);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int left_click(int *row,int *col)
```

```
{
    union REGS regs;
    regs.x.ax=3;
    int86(0x33,&regs,&regs);
    *row=regs.x.dx;
    *col=regs.x.cx;
    if(regs.x.bx==1)return 1;
    else return 0;
}
```

```
int right_click(int *row,int *col)
```

```
{
    union REGS regs;
    regs.x.ax=3;
    int86(0x33,&regs,&regs);
    *row=regs.x.dx;
    *col=regs.x.cx;
    if(regs.x.bx==2)return 1;
    else return 0;
}
```

```
void mouse_range(int left,int top,int right,int bottom)
```

```
{
    union REGS regs;
    regs.x.ax=7;
    regs.x.cx=right;
    regs.x.dx=left;
    int86(0x33,&regs,&regs);
    regs.x.ax=8;
    regs.x.cx=top;
    regs.x.dx=bottom;
    int86(0x33,&regs,&regs);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. สมศักดิ์ กীরตวิมลเศรษฐ์ , “เครื่องมือวัดอุตสาหกรรม” ,สมาคมส่งเสริมเทคโนโลยี ไทย-ญี่ปุ่น , พิมพ์ครั้งที่ 3 , 2532
2. ธันวา ศรีประโมง , “การเขียนโปรแกรมภาษาซีสำหรับวิศวกรรม” , มหาวิทยาลัยมหานคร พิมพ์ครั้งที่ 2 , 2532
3. บุญเลิศ เอี่ยมพัฒนา, ยืน ภู่วรรณ, สมนึก ศรีโต, “โปรแกรมคอมพิวเตอร์ภาษาซี” , บริษัทซีเอ็ดยูเคชั่น จำกัด, 2535
4. มนตรี พจนารถาวณิช , “การเขียนโปรแกรมคอมพิวเตอร์ด้วยเทอร์โบซี” , “การเขียนโปรแกรมคอมพิวเตอร์ด้วยเทอร์โบซี” , บริษัทซีเอ็ดยูเคชั่น จำกัด , 2535
5. ศิววัฒน์ ศิวะบวร , พรชัย จักรธำรงค์ , จิรศักดิ์ชัยวิริยะกุล , “การประยุกต์ใช้งานภาษาซี” , บริษัทซีเอ็ดยูเคชั่น จำกัด , 2535
6. สมพัฒน์ รุ่งตะวันเรืองศรี , “เรียนรู้คอมพิวเตอร์กราฟิกส์ 2 มิติด้วยภาษาซี” , บริษัทซีเอ็ดยูเคชั่น จำกัด , 2535
7. Curtis D. Johnson , “Process Control Instrumentation Technology” , John Willey & sons , Inc., Third edition , 1998
8. J. Micheal Jacob , “Industrial Control Electronics” , Prentice-Hall International , Inc., 1989