

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติ
บนเครื่องควบคุม PLC

DEVELOPMENT OF AUTO-TUNING PID CONTROLLER BASED ON PLC



อภิชาติ พลพิทักษ์
APHICHARD PHONPHITHAK

Qพ.
๑๒๕๒ก
๒๕๔๙

เลขหมู่.....
เลขทะเบียน..... 61710
วัน,เดือน,ปี 19 ก.ค. 2549

b. 11๖๐๖๐๖๕
i.

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2549

ISBN 974-15-2312-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEVELOPMENT OF AUTO-TUNING PID CONTROLLER BASED ON PLC

APHICHARD PHONPHITHAK



A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING IN INSTRUMENTATION ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2006

ISBN 974-15-2312-2
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2006

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารทสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์

การพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC

นักศึกษา

นายอภิชาติ พลพิทักษ์

รหัสนักศึกษา

44061407

ปริญญา

วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา

วิศวกรรมการวัดคุม

พ.ศ.

2549

อาจารย์ผู้ควบคุมวิทยานิพนธ์

รศ.ประสิทธิ์ จุลเสวีวงศ์

อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม

รศ.ทวีพล ชื้อศักดิ์

บทคัดย่อ

ตัวควบคุมพีไอดี (Proportional-Integral-Derivative) ถูกนำมาใช้อย่างกว้างขวางในการควบคุมกระบวนการทางอุตสาหกรรม เนื่องจากมีโครงสร้างที่ไม่ซับซ้อนและให้ผลตอบสนองที่ดี ปัจจุบันเทคนิคการควบคุมแบบนี้ถูกพัฒนารวมเข้าไว้ในเครื่องควบคุม PLC เพื่อใช้ในการควบคุมกระบวนการแบบอัตโนมัติ อย่างไรก็ตามการใช้เทคนิคนี้เป็นเรื่องยากในการปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี (K_p , T_i และ T_d) เพื่อให้ได้ผลตอบสนองที่ดีตามความต้องการ ทางปฏิบัติการปรับแต่งค่าพารามิเตอร์ของตัวควบคุมพีไอดีนี้ยังใช้วิธีการลองผิดลองถูกโดยอาศัยประสบการณ์ของผู้ควบคุม วิทยานิพนธ์นี้นำเสนอการพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC เพื่อใช้ในการควบคุมกระบวนการทางอุตสาหกรรม โดยใช้ตัวควบคุมพีไอดีที่มีอยู่ในเครื่องควบคุม PLC ทำหน้าที่ควบคุมกระบวนการพัฒนาตัวควบคุมนี้ใช้เครื่องควบคุม PLC ทำหน้าที่สังเคราะห์และวิเคราะห์หาค่าพารามิเตอร์ที่เหมาะสมให้กับตัวควบคุมพีไอดีตามหลักการของคาล์ลิน (Dahlin) และการประมาณค่าล่วงหน้า ผลจากการทดลองใช้ตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC ในการควบคุมระดับน้ำและควบคุมอุณหภูมิของตู้อบ พบว่าตัวควบคุมที่ได้พัฒนาขึ้นนี้สามารถควบคุมกระบวนการได้อย่างมีประสิทธิภาพและให้ผลตอบสนองดี

Thesis Title	Development of Auto-Tuning PID Controller Based on PLC
Student	Mr.Aphichard Phonphithak
Student ID.	44061407
Degree	Master of Engineering
Programme	Instrumentation Engineering
Year	2006
Thesis Advisor	Assoc.Prof.Prasit Julsereewong
Thesis Co-Advisor	Assoc.Prof.Taweepol Suesut

ABSTRACT

The conventional PID (Proportional-Integral-Derivative) control technique is widely used for the process control in many industries since it is simple in structure and provides the good response. Nowadays, this control technique has been developed on the Programmable Logic Controller (PLC) to use for the process control loop. However, using this technique is difficult when tuning the PID parameters (K_p , T_i and T_d) to achieve the best response. Moreover, trial-and-error procedure along with the operator experiences is required to obtain the best results when tuning the PID controller parameters. This thesis proposes the auto-tuning PID controller based on PLC for the process control in the industries. The proposed auto-tuning PID controller uses the PLC-based PID structures to control the process production. The proposed PID tuning utilizes the PLC to synthesize and analyze controller parameter as well as to tune for appropriate parameters using Dahlin method and extrapolation. Experimental results using a auto-tuning PID controller to control level and temperature of the oven show that the controller developed is capable of controlling the process very effectively and provides a good response.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีเพราะได้รับความเมตตาจาก รองศาสตราจารย์ ประสิทธิ์ จุลเสวีวงศ์ และ รองศาสตราจารย์ทวีพล ช่อสัคย์ ที่ได้ให้คำแนะนำแก่ ผู้วิจัยตลอดมา ผู้วิจัยรู้สึกซาบซึ้งและขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์พิทยา ปานนิล ที่ได้คำปรึกษา คำชี้แนะ อีกทั้งยังเอื้อเฟื้ออุปกรณ์และเครื่องมือต่างๆ ในการทำวิทยานิพนธ์ฉบับนี้

ขอขอบพระคุณ อาจารย์ภาควิชาวิศวกรรมการวัดคุมทุกท่าน ที่ได้คำแนะนำอันเป็นประโยชน์ต่อการทำวิทยานิพนธ์ฉบับนี้

ที่ลืมนึกได้คือ ขอกราบขอบพระคุณคุณแม่ ที่สนับสนุนและเป็นกำลังใจเสมอมาในการทำวิทยานิพนธ์ฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ผู้วิจัยขอบแต่ผู้มีพระคุณทุกท่าน

อภิชาติ พลพิทักษ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	1
1.3 สมมติฐานของการศึกษา.....	2
1.4 ขอบเขตการวิจัย.....	2
1.5 ขั้นตอนของการศึกษา.....	3
บทที่ 2 ความรู้เบื้องต้นเกี่ยวกับการควบคุมแบบป้อนกลับด้วยตัวควบคุมพีไอดี.....	4
2.1 การควบคุมแบบป้อนกลับด้วยตัวควบคุมพีไอดี.....	4
2.2 กิริยาการควบคุมของการควบคุมแบบป้อนกลับ.....	6
2.2.1 กิริยาการควบคุมแบบ ON-OFF.....	6
2.2.2 กิริยาการควบคุมแบบ Proportional (P).....	7
2.2.3 กิริยาการควบคุมแบบ Integral (I).....	8
2.2.4 กิริยาการควบคุมแบบ Derivative (D).....	9
2.2.5 กิริยาการควบคุมแบบ Proportional - Integral (PI).....	9
2.2.6 กิริยาการควบคุมแบบ Proportional - Derivative (PD).....	10
2.2.7 กิริยาการควบคุมแบบ Proportional - Integral - Derivative (PID).....	11
2.3 การปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี.....	12
2.3.1 การปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี โดยวิธีของ Ziegler-Nichols... 12	
2.3.1.1 วิธี Process Reaction Curve.....	13
2.3.1.2 วิธี Ultimate Method.....	14
2.3.2 การปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี โดยวิธี Damped Oscillation... 16	
2.4 การปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี โดยวิธี Automatic Tuning.....	16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.4.1 Transient response method.....	17
2.4.2 Closed loop method based on relay feedback.....	18
บทที่ 3 หลักการและทฤษฎีที่ใช้ในการออกแบบ.....	21
3.1 การหาค่าคุณลักษณะของกระบวนการ.....	21
3.2 การสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีตามหลักการของคาล์ลัน.....	26
3.3 การประมาณค่าลวงหน้า.....	30
3.4 เครื่องควบคุม PLC	31
3.4.1 โครงสร้างและหลักการทำงานของเครื่องควบคุม PLC.....	31
3.4.2 ภาษาที่ใช้ในการเขียนโปรแกรมของเครื่องควบคุม PLC ตามมาตรฐาน IEC1131-3.....	33
3.4.3 เครื่องควบคุม PLC SIMATIC S7-300/400 “SIEMENS”.....	35
บทที่ 4 การออกแบบและการพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติ บนเครื่องควบคุม PLC.....	46
4.1 กล่าวนำ.....	46
4.2 การประยุกต์ใช้เครื่องควบคุม PLC ตามหลักการและทฤษฎี.....	46
4.3 การออกแบบทางด้านฮาร์ดแวร์.....	49
4.3.1 ส่วนของเครื่องควบคุม PLC.....	49
4.3.2 ส่วนกำหนดค่าการทำงานและแสดงผล.....	50
4.4 การออกแบบทางด้านซอฟต์แวร์.....	51
4.4.1 การเขียนโปรแกรมการทำงานของเครื่องควบคุม PLC.....	51
4.4.2 การเขียนโปรแกรมแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์.....	64
4.5 การใช้งานตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์แบบอัตโนมัติบน เครื่องควบคุม PLC.....	65
4.5.1 หน้าจอหลัก.....	65
4.5.2 หน้าจอการควบคุมแบบ Manual.....	66
4.5.3 หน้าจอการควบคุมแบบ Automatic.....	67
4.5.4 หน้าจอการควบคุมแบบ Auto-tuning.....	68

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 5 ผลการทดลอง.....	71
5.1 กล่าวนำ.....	71
5.2 ผลการทดลองกับกระบวนการควบคุมระดับน้ำ.....	71
5.3 ผลการทดลองกับกระบวนการควบคุมอุณหภูมิ.....	88
5.4 สรุปผลการทดลอง.....	101
บทที่ 6 สรุปและข้อเสนอแนะ.....	102
6.1 สรุป.....	102
6.2 ข้อเสนอแนะ.....	102
บรรณานุกรม.....	104
ภาคผนวก.....	106
ภาคผนวก ก. ผลงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์และได้รับการตีพิมพ์.....	107
ประวัติผู้เขียน.....	114

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงค่าพารามิเตอร์ของตัวควบคุมแบบต่างๆ ตามวิธี Process Reaction Curve.....	14
2.2 แสดงค่าพารามิเตอร์ของตัวควบคุมแบบต่างๆ ตามวิธี Ultimate Method.....	15
2.3 แสดงค่าพารามิเตอร์ของตัวควบคุมแบบต่างๆ ตามวิธี Ziegler-Nichols แบบ Step response method.....	18
2.4 แสดงค่าพารามิเตอร์ของตัวควบคุมแบบต่างๆ ตามวิธี Ziegler-Nichols แบบ Closed-loop method.....	20
3.1 แสดงสูตรสำหรับหาค่าพารามิเตอร์ของตัวควบคุม PID ของ Dahlin.....	30



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 แสดงโครงสร้างของระบบควบคุมแบบป้อนกลับโดยทั่วไป.....	5
2.2 แสดงคุณสมบัติของกริยาควบคุมแบบ ON-OFF.....	6
2.3 แสดงคุณสมบัติของกริยาควบคุมแบบ Proportional.....	7
2.4 แสดงผลตอบสนองของกริยาควบคุมแบบ Integral.....	8
2.5 แสดงตัวอย่างผลตอบสนองของกริยาควบคุมแบบ Derivative.....	9
2.6 แสดงตัวอย่างผลตอบสนองของกริยาควบคุมแบบ PI (Direct action).....	10
2.7 แสดงตัวอย่างผลตอบสนองของกริยาควบคุมแบบ PD.....	11
2.8 แสดงตัวอย่างผลตอบสนองของกริยาควบคุมแบบ PID.....	12
2.9 แสดงค่าพ่วงเกินที่ต้องการเมื่อใช้การปรับด้วยวิธี Ziegler – Nichols.....	13
2.10 แสดงผลตอบสนองรูปตัว S เมื่อปรับโดยใช้วิธี Process Reaction Curve.....	14
2.11 แสดงผลตอบสนองเวลาเกิดการแกว่งอย่างต่อเนื่อง เมื่อปรับโดยใช้วิธี Ultimate method.....	15
2.12 แสดงค่าอัตราการเสื่อม 1/4 เมื่อปรับโดยวิธี Damped Oscillation.....	16
2.13 แสดงผลตอบสนองของกระบวนการแบบ FOPDT หลังจากป้อนสัญญาณแบบขั้นบันได....	17
2.14 แสดงสัญญาณ Relay ที่มีการแกว่งอย่างต่อเนื่องที่ป้อนให้กับกระบวนการ.....	18
2.15 แสดงสัญญาณ Relay และผลตอบสนองของกระบวนการ.....	19
3.1 แสดงโครงสร้างของระบบควบคุมแบบป้อนกลับ.....	21
3.2 แสดงบล็อกไดอะแกรมของระบบควบคุมแบบป้อนกลับหลังจากการลดรูป.....	22
3.3 แสดงบล็อกไดอะแกรมของการทดสอบกระบวนการแบบรูปเปิด.....	23
3.4 แสดงผลตอบสนองของกระบวนการแบบ FOPDT หลังจากป้อนสัญญาณแบบขั้นบันได....	24
3.5 แสดงการประมาณค่าพารามิเตอร์ของกระบวนการแบบ FOPDT.....	25
3.6 แสดงบล็อกไดอะแกรมของระบบแบบรูปปิดเมื่อไม่พิจารณาสัญญาณรบกวน.....	26
3.7 แสดงผลตอบสนองแบบรูปปิดที่ใช้ในการสังเคราะห์ค่าพารามิเตอร์ PID.....	27
3.8 แสดงผลตอบสนองแบบปิดที่มีการหน่วงเวลาเป็นเวลา t_0	28
3.9 แสดงโครงสร้างโดยทั่วไปของเครื่องควบคุม PLC.....	32
3.10 แสดงส่วนประกอบภายในหน่วยประมวลผลกลาง.....	32
3.11 แสดงวงรอบการทำงานของเครื่องควบคุม PLC.....	33
3.12 แสดงตัวอย่างการเขียนโปรแกรมโดยใช้ภาษา Ladder Diagram.....	34
3.13 แสดงตัวอย่างการเขียนโปรแกรมโดยใช้ภาษา Sequential Function Charts.....	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.14 แสดงตัวอย่างการเขียน โปรแกรม โดยใช้ภาษา Function Block Diagram.....	36
3.15 แสดงโครงสร้างของเครื่องควบคุม PLC SIMATIC S7-300.....	37
3.16 แสดงการขยายอินพุทเอาต์พุท โดยใช้โมดูลอินเตอร์เฟซ.....	37
3.17 แสดงการกระจายอินพุทเอาต์พุทผ่านการสื่อสารแบบ Profibus-DP.....	38
3.18 แสดงพื้นที่ของหน่วยความจำภายในหน่วยประมวลผลกลาง.....	38
3.19 แสดงโครงสร้าง โปรแกรมแบบลิเนียร์.....	42
3.20 แสดงโครงสร้าง โปรแกรมแบบแบ่งส่วน.....	42
3.21 แสดงโครงสร้าง โปรแกรมแบบ โครงสร้าง.....	43
3.22 แสดงการกำหนดค่าของฮาร์ดแวร์ของเครื่องควบคุม PLC.....	44
3.23 แสดงการเขียน โปรแกรมการทำงานของเครื่องควบคุม PLC.....	45
4.1 แสดงบล็อก ไคแกรมของระบบ Auto-tuning.....	47
4.2 แสดงผังขั้นตอนการทำงานของตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC.....	48
4.3 แสดงการออกแบบทางด้านฮาร์ดแวร์ส่วนของเครื่องควบคุม PLC.....	49
4.4 แสดงการออกแบบทางด้านฮาร์ดแวร์ของส่วนแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์.....	50
4.5 แสดงการ์ดอินเตอร์เฟซ.....	50
4.6 แสดงการทำงานของบล็อกฟังก์ชัน Auto-tuning	52
4.7 แสดงการทำงานของบล็อกฟังก์ชันการหาค่าพารามิเตอร์ของตัวควบคุมพีไอดี.....	53
4.8 แสดงการทำงานของบล็อกฟังก์ชัน Step Test.....	54
4.9 แสดงการทำงานของบล็อกฟังก์ชันการเก็บค่าข้อมูลที่คาบตัวอย่าง.....	55
4.10 แสดงการทำงานของบล็อกฟังก์ชันการหาค่า t_0	56
4.11 แสดงการทำงานของบล็อกฟังก์ชันการประมาณค่าล่วงหน้าของ c_s	58
4.12 แสดงการทำงานของบล็อกฟังก์ชันการหาค่า c_s	59
4.13 แสดงการทำงานของบล็อกฟังก์ชันการหาค่า t_1	60
4.14 แสดงการทำงานของบล็อกฟังก์ชันการหาค่า t_2	61
4.15 แสดงการทำงานของบล็อกฟังก์ชัน Relay Feedback.....	62
4.16 แสดงบล็อก SCALE (A/D converter).....	63
4.17 แสดงบล็อก UNSCALE (D/A converter).....	63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.18 แสดงบล็อกเก็บข้อมูล.....	64
4.19 แสดงการเขียนโปรแกรมแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์.....	65
4.20 แสดงหน้าจอหลัก.....	66
4.21 แสดงหน้าจอการควบคุมแบบ Manual.....	67
4.22 แสดงหน้าจอการควบคุมแบบ Automatic.....	68
4.23 แสดงหน้าจอการควบคุมแบบ Auto-tuning	70
5.1 แสดงการประยุกต์ใช้ตัวควบคุมพีไอดีแบบค่าพารามิเตอร์อัตโนมัติบน เครื่องควบคุม PLC ควบคุมระดับน้ำในถัง.....	71
5.2 แสดงโครงสร้างการทดลองกับกระบวนการควบคุมระดับน้ำ.....	72
5.3 แสดงผลตอบสนองของกระบวนการจากการทดสอบ โดยป้อนสัญญาณทดสอบ ขนาด $\Delta m = 30\%$ ในโหมด Step Test แบบ Normal	73
5.4 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% หลังจากได้ ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.).....	74
5.5 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% ด้วย การควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.).....	74
5.6 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% และ Step change ไปที่ 50% ด้วยการควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.).....	75
5.7 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% และ Step change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.).....	75
5.8 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% และ Step change ไปที่ 50% และ Step change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.).....	76
5.9 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% ด้วย การควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.).....	76
5.10 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 75% ด้วย การควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.).....	77
5.11 แสดงผลตอบสนองของกระบวนการจากการทดสอบ โดยป้อนสัญญาณทดสอบ ขนาด $\Delta m = 30\%$ ในโหมด Step Test แบบ Predict.....	78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.12 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% หลังจากได้ ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.).....	79
5.13 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% ด้วย การควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.).....	79
5.14 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% และ Step change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.).....	80
5.15 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% และ Step change ไปที่ 50% และ Step change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.).....	80
5.16 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% ด้วย การควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.).....	81
5.17 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 75% ด้วย การควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.).....	81
5.18 แสดงผลตอบสนองของกระบวนการการจากการทดสอบแบบ Relay Feedback ที่ ค่าเป้าหมาย 25%.....	82
5.19 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% หลังจากได้ ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 4.00$ และ $T_i = 12.00$ sec.).....	83
5.20 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% ด้วย การควบคุมแบบพีไอ ($K_p = 4.00$ และ $T_i = 12.00$ sec.).....	83
5.21 แสดงผลตอบสนองของกระบวนการการจากการทดสอบแบบ Relay Feedback ที่ ค่าเป้าหมาย 50%.....	84
5.22 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% หลังจากได้ ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 4.26$ และ $T_i = 11.20$ sec.).....	85
5.23 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% ด้วย การควบคุมแบบพีไอ ($K_p = 4.26$ และ $T_i = 11.20$ sec.).....	85
5.24 แสดงผลตอบสนองของกระบวนการการจากการทดสอบแบบ Relay Feedback ที่ ค่าเป้าหมาย 75%.....	86
5.25 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 75% หลังจากได้ ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 4.40$ และ $T_i = 11.20$ sec.).....	87

เอกสารนี้เป็นทรัพย์สินทางปัญญาของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.26 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 75% ด้วยการควบคุมแบบพีไอ ($K_p = 4.40$ และ $T_i = 11.20$ sec.).....	87
5.27 แสดงการประยุกต์ใช้ตัวควบคุมพีไอดีแบบค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC ควบคุมระดับอุณหภูมิของตู้อบ.....	88
5.28 แสดงโครงสร้างการทดลองกับกระบวนการควบคุมอุณหภูมิของตู้อบ.....	89
5.29 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิจากการทดสอบโดยป้อนสัญญาณทดสอบขนาด $\Delta m = 30\%$ ในโหมด Step Test แบบ Normal.....	90
5.30 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 8.46$ และ $T_i = 537.00$ sec.).....	91
5.31 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% ด้วยการควบคุมแบบพีไอ ($K_p = 8.46$ และ $T_i = 537.00$ sec.).....	92
5.32 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% และ Step Change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 8.46$ และ $T_i = 537.00$ sec.).....	92
5.33 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% ด้วยการควบคุมแบบพีไอ ($K_p = 8.46$ และ $T_i = 537.00$ sec.).....	93
5.34 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิจากการทดสอบโดยป้อนสัญญาณทดสอบขนาด $\Delta m = 30\%$ ในโหมด Step Test แบบ Predict.....	94
5.35 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 8.41$ และ $T_i = 317.16$ sec.).....	95
5.36 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% ด้วยการควบคุมแบบพีไอ ($K_p = 8.41$ และ $T_i = 317.16$ sec.).....	95
5.37 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% และ Step Change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 8.41$ และ $T_i = 317.16$ sec.).....	96
5.38 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% ด้วยการควบคุมแบบพีไอ ($K_p = 8.41$ และ $T_i = 317.16$ sec.).....	96
5.39 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิการจากการทดสอบแบบ Relay Feedback ที่ค่าเป้าหมาย 50%.....	97
5.40 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 14.30$ และ $T_i = 119.00$ sec.).....	98

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการใช้งานในวงจำกัดเท่านั้น เมื่อผู้ดูเห็นได้โปรดแจ้งให้ทราบ

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.41 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% ด้วย การควบคุมแบบพีไอ ($K_p = 14.30$ และ $T_i = 119.00$ sec.).....	98
5.42 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิจากการทดสอบแบบ Relay Feedback ที่ค่าเป้าหมาย 75%.....	99
5.43 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% หลังจากได้ ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 13.20$ และ $T_i = 127.50$ sec.).....	100
5.44 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% ด้วย การควบคุมแบบพีไอ ($K_p = 13.20$ และ $T_i = 127.50$ sec.).....	100



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

กระบวนการทางอุตสาหกรรมมีลักษณะการทำงานแบบอัตโนมัติโดยใช้เครื่องควบคุม PLC ทำหน้าที่ควบคุมการทำงานของกระบวนการเนื่องจากเครื่องควบคุม PLC มีราคาไม่แพง ใช้งานง่าย มีความยืดหยุ่นและมีความน่าเชื่อถือสูงเมื่อเทียบกับระบบควบคุมแบบเดิมที่ใช้วงจรรีเลย์ เริ่มแรกเครื่องควบคุม PLC ถูกสร้างขึ้นเพื่อใช้กับการควบคุมแบบลำดับเท่านั้น ต่อมาเครื่องควบคุม PLC ได้ถูกพัฒนาประสิทธิภาพขึ้นอย่างต่อเนื่องจนปัจจุบันเครื่องควบคุม PLC สามารถใช้กับการควบคุมแบบป้อนกลับได้ [1][2][3] ซึ่งเครื่องควบคุม PLC มีฟังก์ชันบล็อกพิเศษ เช่น การควบคุมแบบพีไอดี (PID control) การควบคุมแบบฟัซซี่ (Fuzzy control) เป็นต้น บรรจุอยู่ภายในซอฟต์แวร์สำหรับการเขียน โปรแกรมให้กับเครื่องควบคุม PLC [3][4] โดยทั่วไปการนำเครื่องควบคุม PLC ไปใช้ควบคุมกระบวนการแบบป้อนกลับนิยมใช้ฟังก์ชันบล็อกการควบคุมแบบพีไอดีเนื่องจากการควบคุมแบบพีไอดีมีโครงสร้างการทำงานที่ไม่ซับซ้อน เข้าใจง่าย สามารถประยุกต์ใช้งานกับกระบวนการ ได้หลายแบบและให้ผลตอบสนองของกระบวนการที่ดี [5][6] การใช้งานฟังก์ชันบล็อกการควบคุมแบบพีไอดีนี้จำเป็นต้องปรับแต่งค่าพารามิเตอร์ของตัวควบคุมพีไอดี (K_p , T_i และ T_d) เพื่อให้ได้ผลตอบสนองของกระบวนการที่ดีตามความต้องการ อย่างไรก็ตามเป็นเรื่องยากในการปรับแต่งค่าพารามิเตอร์ของตัวควบคุมพีไอดีหากไม่ทราบโมเดลทางคณิตศาสตร์ของกระบวนการ (Mathematical model of process) ทางปฏิบัติการปรับแต่งค่าพารามิเตอร์ของตัวควบคุมพีไอดีนี้ยังนิยมใช้การลองผิดลองถูกโดยอาศัยประสบการณ์จากผู้ใช้งานเพื่อให้ได้ผลตอบสนองของกระบวนการที่ดี [7][8] หากผู้ใช้งานมีประสบการณ์ไม่เพียงพอ อาจใช้ระยะเวลาานานมากและอาจเกิดความเสียหายกับกระบวนการได้

งานวิจัยในวิทยานิพนธ์ฉบับนี้จึงได้พัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC (Development of Auto-tuning PID controller based on PLC) เพื่อให้เครื่องควบคุม PLC สามารถทำการวิเคราะห์และประมาณค่าพารามิเตอร์ของตัวควบคุมพีไอดีสำหรับควบคุมกระบวนการได้อย่างอัตโนมัติตามหลักการของดาห์ลิน (Dahlin) และการประมาณค่าล่วงหน้า การใช้เทคนิคนี้ช่วยให้การประมาณค่าพารามิเตอร์ของกระบวนการรวดเร็วขึ้นและง่ายต่อการใช้งานแล้วยังให้ผลตอบสนองของกระบวนการที่ดี

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

งานวิจัยในวิทยานิพนธ์ฉบับนี้จะเป็นการพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC เพื่อให้เครื่องควบคุม PLC สามารถทำการวิเคราะห์และปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดีบนเครื่องควบคุม PLC ได้อย่างอัตโนมัติเพื่อช่วยลดระยะเวลาและความยุ่งยากในการปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี โดยผู้ใช้งานไม่จำเป็นต้องมีความรู้ทางคณิตศาสตร์ที่ใช้ในการวิเคราะห์กระบวนการทางคณิตศาสตร์และไม่จำเป็นต้องมีประสบการณ์ในการปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดีสูงนัก ทำให้สามารถใช้เครื่องควบคุม PLC ทำการควบคุมกระบวนการควบคุมแบบป้อนกลับโดยใช้ฟังก์ชันการควบคุมแบบพีไอดีได้ง่ายและสะดวกขึ้น

1.3 สมมติฐานของการศึกษา

การใช้งานฟังก์ชันตัวควบคุมพีไอดีของเครื่องควบคุม PLC ในการควบคุมกระบวนการแบบป้อนกลับนั้นจำเป็นต้องปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี เพื่อให้ได้ผลตอบสนองของกระบวนการที่ดีตามความต้องการ การปรับค่าพารามิเตอร์ของตัวควบคุมนี้จะทำได้ยากหากไม่ทราบโมเดลทางคณิตศาสตร์ของกระบวนการ โดยผู้ใช้งานจำเป็นต้องมีความรู้ทางคณิตศาสตร์และมีประสบการณ์สูงในการปรับค่าพารามิเตอร์นี้ หากสามารถออกแบบและพัฒนาให้เครื่องควบคุม PLC ที่ใช้ควบคุมกระบวนการแบบป้อนกลับนั้นสามารถปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดีได้อย่างอัตโนมัติ ทำให้ผู้ใช้งานสามารถใช้งานเครื่องควบคุม PLC ในการควบคุมกระบวนการได้ง่ายและสะดวกขึ้น โดยไม่จำเป็นต้องมีความรู้ทางคณิตศาสตร์และประสบการณ์สูงนัก

1.4 ขอบเขตการวิจัย

วิทยานิพนธ์ฉบับนี้เป็นการนำเสนอการพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC ซึ่งขั้นตอนการทำงานที่สำคัญของตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC นี้คือ การใช้เครื่องควบคุม PLC หาค่าพารามิเตอร์ของกระบวนการ (Process Parameter) โดยกระบวนการจะถูกอนุมานให้อยู่ในรูปแบบของกระบวนการอันดับหนึ่งที่มีค่าหน่วงเวลา (First-order plus dead time: FOPDT) และทำทดสอบกระบวนการโดยการป้อนสัญญาณอินพุตแบบขั้นบันไดให้กับกระบวนการแบบวงเปิด (Opened-loop) จะได้ค่าพารามิเตอร์ของกระบวนการ คือ ค่าอัตราขยายของกระบวนการ (Process Gain) ค่าการหน่วงเวลาของกระบวนการ (Process Dead Time) และค่าเวลาคงที่ของกระบวนการ (Process Time Constant) ซึ่งค่าพารามิเตอร์ของกระบวนการทั้ง 3 ค่านี้ จะถูกนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีตามหลักการของคาล์ลิน โดยใช้เครื่องควบคุม PLC และกระบวนการจำลองภายในสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

1.5 ขั้นตอนของการศึกษา

การทำโครงการวิจัยในวิทยานิพนธ์ฉบับนี้มีขั้นตอนการศึกษาเริ่มจากศึกษาการทำงานของตัวควบคุมแบบต่างๆ เช่น ตัวควบคุมแบบ ON-OFF ตัวควบคุมแบบพี ตัวควบคุมแบบไอ ตัวควบคุมแบบดี ตัวควบคุมแบบพีไอและตัวควบคุมแบบพีไอดี เป็นต้น วิธีการปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดีตามหลักการต่างๆ เช่น Ziegler-Nichols และ Dahlin เป็นต้น เพื่อใช้เป็นแนวทางในการวิจัย จากนั้นทำการศึกษาค้นคว้าคุณลักษณะของกระบวนการเพื่อหาวิธีที่จะประมาณค่าพารามิเตอร์ของกระบวนการ ซึ่งเป็นตัวแปรที่สำคัญที่จะนำไปสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอดี เมื่อได้วิธีการและขั้นตอนในการหาคุณลักษณะของกระบวนการและการสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีแล้ว ต่อไปก็คือการออกแบบและพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC สุดท้ายได้นำตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC ไปประยุกต์ใช้ในทดลองกับกระบวนการจำลองที่มีอยู่ในห้องทดลองของภาควิชาเพื่อยืนยันถึงประสิทธิภาพและหาข้อบกพร่องตลอดจนแนวทางแก้ไขเพื่อเป็นแนวทางในการพัฒนาต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

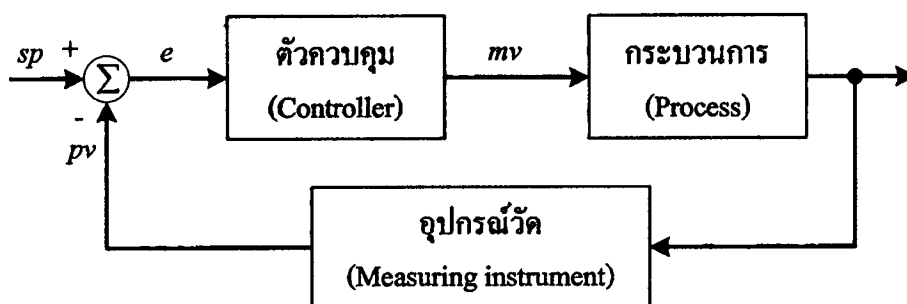
ความรู้เบื้องต้นเกี่ยวกับการควบคุมแบบป้อนกลับด้วย ตัวควบคุมพีไอดี

2.1 การควบคุมแบบป้อนกลับด้วยตัวควบคุมพีไอดี

ระบบควบคุมแบบป้อนกลับ หมายถึง ระบบควบคุมที่ต้องการควบคุมให้เอาต์พุตหรือผลตอบสนองของระบบมีค่าตามที่ต้องการ โดยการนำเอาสัญญาณเอาต์พุตป้อนกลับมาเปรียบเทียบกับสัญญาณอินพุต ซึ่งการป้อนกลับนี้จะต้องเป็นการป้อนกลับแบบลบ (Negative Feedback) จากนั้นนำค่าความแตกต่างระหว่างอินพุตกับเอาต์พุต ส่งต่อไปยังส่วนสร้างสัญญาณควบคุมซึ่งจะทำการสร้างสัญญาณควบคุมที่เหมาะสมตามกฎเกณฑ์การควบคุมที่เลือกไว้และส่งสัญญาณออกไปควบคุมระบบเพื่อให้เอาต์พุตของกระบวนการที่ถูกควบคุมมีค่าตามที่ต้องการ ระบบควบคุมป้อนกลับนี้อาจจะเรียกอีกอย่างหนึ่งว่า ระบบควบคุมแบบลูปิด หรือระบบควบคุมอัตโนมัติก็ได้

ระบบควบคุมแบบป้อนกลับโดยทั่วไปประกอบด้วยอุปกรณ์ 3 ส่วน คือ

- 1. ตัวควบคุม (Controller)** หมายถึง เครื่องมือหรืออุปกรณ์ที่ใช้ในการสร้างสัญญาณควบคุม เพื่อทำหน้าที่ควบคุมให้ระบบหรือกระบวนการที่ต้องการควบคุม มีเอาต์พุตหรือผลตอบสนองตามความต้องการ ตัวควบคุมจะมีหลายแบบ เช่น ตัวควบคุมแบบ ON-OFF ตัวควบคุมแบบ Proportional (P) ตัวควบคุมแบบ Integral (I) ตัวควบคุมแบบ Derivative (D) หรือการใช้ตัวควบคุมหลายๆ แบบรวมกัน เช่น ตัวควบคุมแบบ PI ตัวควบคุมแบบ PD และตัวควบคุมแบบ PID เป็นต้น ซึ่งสัญญาณที่ออกจากตัวควบคุมจะเรียกว่าตัวแปรปรับกระบวนการ (Manipulated Variable: mv)
- 2. กระบวนการ (Plant or Process)** หมายถึง ระบบหรือกระบวนการที่ถูกควบคุม หรืออาจจะเป็นวัตถุทางกายภาพที่ถูกควบคุมก็ได้ เช่น กระบวนการเกี่ยวกับการควบคุมอุณหภูมิ กระบวนการเกี่ยวกับการควบคุมความดัน เป็นต้น กระบวนการนี้รวมถึงอุปกรณ์ควบคุมสุดท้าย (Final Control Element) ซึ่งสถานะของกระบวนการเรียกว่าตัวแปรกระบวนการ (Process Variable: pv)
- 3. อุปกรณ์วัด (Measuring Instruments)** หมายถึง อุปกรณ์ที่อาจจะ ได้แก่ เซนเซอร์ (Sensor) ทรานสดิวเซอร์ (Transducer) อุปกรณ์แปลง (Transmitter) หรือวัดสัญญาณอื่นๆ ที่ทำหน้าที่วัดค่าของเอาต์พุตของระบบที่ถูกควบคุมแล้วป้อนกลับไปเพื่อทำการเปรียบเทียบกับอินพุตของระบบ โดยสัญญาณที่ใช้เป็นสัญญาณมาตรฐานทางอุตสาหกรรม ได้แก่ สัญญาณไฟฟ้ากระแสตรง 4-20 มิลลิแอมป์แปร์ (4-20 mA DC) สัญญาณแรงดันไฟฟ้ากระแสตรง 1-5 โวลต์ (1-5 VDC) หรือสัญญาณลมขนาด 3-15 ปอนด์/ตารางนิ้ว (3-15 PSI) เป็นต้น



รูปที่ 2.1 แสดงโครงสร้างของระบบควบคุมแบบป้อนกลับโดยทั่วไป

การควบคุมกระบวนการทางอุตสาหกรรมโดยทั่วไปนิยมใช้ตัวควบคุมแบบพีไอดี (PID) เพราะรูปแบบของตัวควบคุมเป็นตัวควบคุมที่สามารถควบคุมกระบวนการต่างๆ ได้อย่างกว้างขวาง เนื่องจากมีโครงสร้างการทำงานที่ไม่ซับซ้อน สามารถเข้าใจได้ง่าย การใช้งานตัวควบคุมพีไอดีนี้ ขึ้นอยู่กับการปรับค่าพารามิเตอร์ของตัวควบคุม PID ให้เหมาะสมเพื่อให้ได้ผลตอบสนองของกระบวนการตามความต้องการ

ตัวควบคุมแบบ PID ประกอบด้วยตัวควบคุมแบบ Proportional (P) ตัวควบคุมแบบ Integral (I) ตัวควบคุมแบบ Derivative (D) ซึ่งมีฟังก์ชันถ่ายโอน (Transfer function) ดังนี้

$$m(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right) \quad (2.1)$$

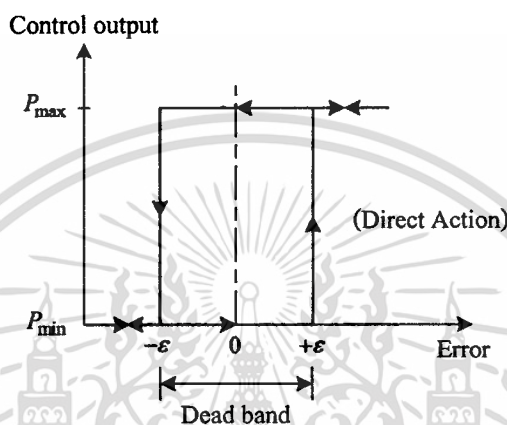
โดยที่ K_p = ค่าอัตราขยายของตัวควบคุมแบบ P (Proportional Gain)
 T_i = ค่าเวลา Integral (Integral Time)
 T_d = ค่าเวลา Derivative (Derivative Time)

จากรูปที่ 2.1 จะเห็นได้ว่าสัญญาณควบคุมหรือตัวแปรปรับกระบวนการ (mv) ที่ได้จากตัวควบคุม PID จะถูกกำหนดด้วยความสัมพันธ์ระหว่างสัญญาณขาเข้าตัวควบคุมกับตัวแปรกระบวนการ (pv) กับสัญญาณอ้างอิงหรือค่าเป้าหมาย (sp) โดยที่ความสัมพันธ์ดังกล่าวจะขึ้นอยู่กับกฎเกณฑ์การควบคุมที่ผู้ควบคุมปรับแต่งไว้ล่วงหน้า (ค่าพารามิเตอร์ของตัวควบคุม PID) ซึ่งสัญญาณควบคุมของตัวควบคุมจะเป็นไปตามกริยาการควบคุมแบบต่างๆ ดังที่จะกล่าวต่อไป

2.2 กริยาการควบคุมของการควบคุมแบบป้อนกลับ

2.2.1 กริยาการควบคุมแบบ ON-OFF

การควบคุมแบบ ON-OFF เป็นการควบคุมแบบที่ง่ายที่สุดและนิยมใช้ในการควบคุมกระบวนการที่ไม่ต้องการความเที่ยงตรงสูง โดยการควบคุมจะทำงานเพียง 2 สถานะ คือ เปิด (100%) กับปิด (0%) เท่านั้น กริยาการควบคุมแบบ ON-OFF ดังรูปที่ 2.2



รูปที่ 2.2 แสดงกริยาการควบคุมแบบ ON-OFF

จากรูปที่ 2.2 จะเห็นว่าถ้าค่าความคลาดเคลื่อนมากกว่าค่าวิกฤต ($+\epsilon$) ค่าเอาต์พุตของตัวควบคุมจะเปลี่ยนจาก 0% เป็น 100% เมื่อค่าความคลาดเคลื่อนน้อยกว่าค่าวิกฤต ($-\epsilon$) ค่าเอาต์พุตของตัวควบคุมจะเปลี่ยนจาก 100% เป็น 0% ค่าเอาต์พุตที่อยู่ในช่วงเขตแบนด์ (Dead Band) จะไม่มีการเปลี่ยนแปลงแต่อย่างใดซึ่งอาจจะเป็นผลมาจากการเสียดทานที่ไม่ได้คาดไว้ก่อนหรือบางครั้งก็จำเป็นต้องทำให้เกิดช่วงเขตแบนด์ขึ้นเพื่อป้องกันการ ON-OFF บ่อยเกินไปอันจะทำให้อุปกรณ์ควบคุมหรือกระบวนการได้รับความเสียหาย แต่ช่วงเขตแบนด์นี้ต้องไม่กว้างนักเพราะจะทำให้ค่าความเที่ยงตรงของการควบคุมลดลง กริยาการควบคุมแบบ ON-OFF สามารถเขียนเป็นสมการได้ดังนี้

$$m(t) = \begin{cases} 0\% & , e < -\epsilon \\ 100\% & , e > +\epsilon \end{cases} \quad (2.2)$$

เมื่อ $m(t)$ = สัญญาณควบคุมหรือเอาต์พุตของตัวควบคุม
 $e(t)$ = ค่าความคลาดเคลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\varepsilon = \frac{1}{2} \text{ ของค่าเดทแบนด์}$$

2.2.2 กิริยาการควบคุมแบบ Proportional (P)

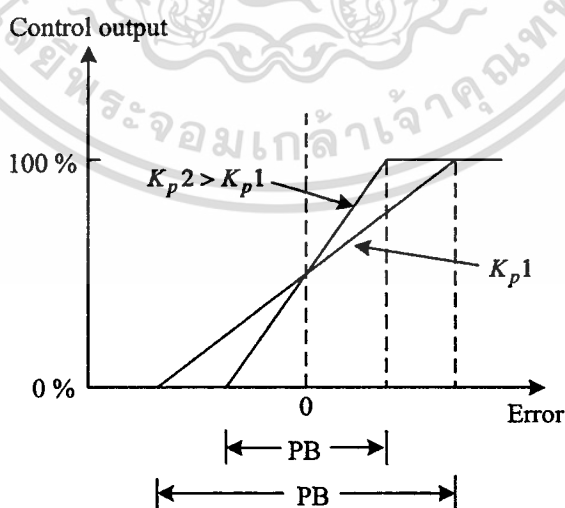
กิริยาการควบคุมแบบ P นั้น ค่าเอาต์พุทของตัวควบคุมจะแปรผันตรงกับค่าความคลาดเคลื่อน คือ ถ้าค่าความคลาดเคลื่อนมีค่ามากขึ้น ค่าเอาต์พุทของตัวควบคุมก็จะมีค่ามากขึ้นตาม และถ้าค่าความคลาดเคลื่อนมีค่าน้อยลง ค่าเอาต์พุทของตัวควบคุมก็จะมีค่าน้อยลงตาม กิริยาการควบคุมแบบ P สามารถเป็นเขียนสมการ ได้ดังนี้

$$m_p(t) = K_p e(t) + \bar{m} \quad (2.3)$$

เมื่อ $m_p(t)$ = ค่าเอาต์พุทของตัวควบคุมแบบ Proportional
 K_p = อัตราขยายของตัวควบคุมแบบ Proportional
 \bar{m} = ค่าเอาต์พุทของตัวควบคุมที่ค่าความคลาดเคลื่อนเท่ากับศูนย์

ตัวควบคุมแบบ P บางตัวอาจจะใช้ค่า Proportional Band (PB) แทนการใช้ค่า K_p ซึ่ง PB คือ ช่วงของค่าความคลาดเคลื่อนระหว่างที่เอาต์พุทของตัวควบคุมมีค่า 0-100 % ดังสมการ (2.4)

$$PB = \frac{100\%}{K_p} \quad (2.4)$$



รูปที่ 2.3 แสดงคุณสมบัติของกิริยาการควบคุมแบบ Proportional

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเสียของกริยาการควบคุมแบบ Proportional คือ ไม่สามารถกำจัดค่าออฟเซ็ทได้

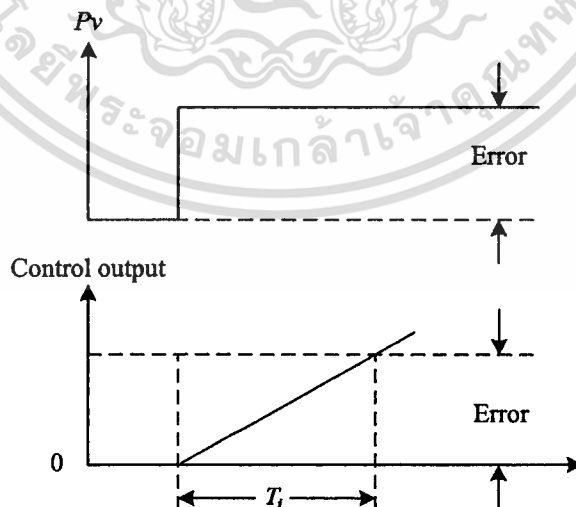
2.2.3 กริยาการควบคุมแบบ Integral (I)

กริยาการควบคุมแบบ I หรือเรียกอีกอย่างหนึ่งว่า การควบคุมแบบรีเซ็ต (Reset Control) โดยค่าเอาต์พุทของตัวควบคุมหาได้จากค่าพื้นที่ทั้งหมดภายใต้กราฟของค่าความคลาดเคลื่อนต่อเวลาคูณกับค่าคงที่ที่เรียกว่า อัตราขยายของตัวควบคุมแบบ I (Integral Gain) ดังสมการต่อไปนี้

$$m_I(t) = K_I \int_0^t e(t) dt + \bar{m}_I(0) \quad (2.5)$$

เมื่อ $m_I(t)$ = ค่าเอาต์พุทของตัวควบคุมแบบ Integral
 K_I = อัตราขยายของตัวควบคุมแบบ Integral
 $\int_0^t e(t) dt$ = พื้นที่ทั้งหมดของค่าความคลาดเคลื่อน
 $\bar{m}_I(0)$ = ค่าเอาต์พุทของตัวควบคุมที่เวลา t เท่ากับศูนย์

ผลของกริยาการควบคุมแบบ I นี้จะทำให้ไม่เกิดออฟเซ็ทขึ้นในระบบและลดค่าพุ่งเกิน (Overshoot) ของระบบลงได้ แต่ถ้ากริยาการควบคุมมีค่าสูงเกินไปจะทำให้ผลตอบสนองของกระบวนการช้าลง ผลตอบสนองของกริยาการควบคุมแบบ I แสดงดังรูปที่ 2.4



รูปที่ 2.4 แสดงผลตอบสนองของกริยาการควบคุมแบบ Integral

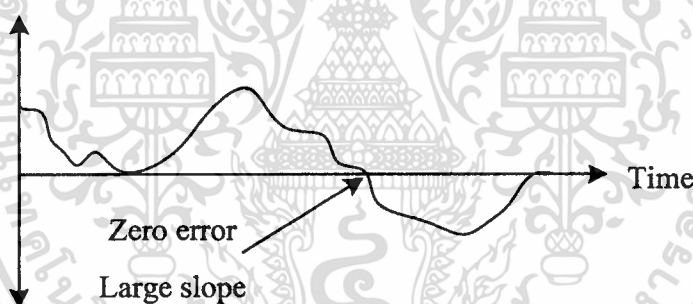
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 กริยาการควบคุมแบบ Derivative (D)

กริยาการควบคุมแบบ D เรียกอีกชื่อหนึ่งว่าการควบคุมแบบอัตราส่วน (Rate Action) โดยสัญญาณเอาต์พุตของตัวควบคุมจะขึ้นอยู่กับอัตราการเปลี่ยนแปลงของค่าความคลาดเคลื่อนต่อเวลา จะเห็นว่าค่าความคลาดเคลื่อนนี้มีโอกาสเป็นศูนย์ได้ และค่าเอาต์พุตก็สามารถเปลี่ยนแปลงให้มีค่าสูงขึ้น เมื่อความคลาดเคลื่อนเปลี่ยนแปลง ซึ่งเรียกการกระทำดังกล่าวว่า อัตราการกระทำ (Rate Action) ดังสมการต่อไปนี้

$$m_D(t) = K_D \frac{de(t)}{dt} \quad (2.6)$$

เมื่อ $m_D(t)$ = ค่าเอาต์พุตของตัวควบคุมแบบ Derivative
 K_D = อัตราขยายของตัวควบคุมแบบ Derivative
 $\frac{de(t)}{dt}$ = ค่าความคลาดเคลื่อนที่เวลา t



รูปที่ 2.5 แสดงตัวอย่างผลตอบสนองของกริยาการควบคุมแบบ Derivative

2.2.5 กริยาการควบคุมแบบ Proportional-Integral (PI)

ตามที่กล่าวมาแล้วว่ากริยาการควบคุมแบบ P นั้น จะมีออฟเซ็ทเกิดขึ้น ซึ่งการกำจัดค่าออฟเซ็ทนี้สามารถทำได้โดยการเพิ่มกริยาการควบคุมแบบ I เข้าไป ดังนั้นสมการเอาต์พุตของตัวควบคุมแบบ PI นี้ ดังสมการ

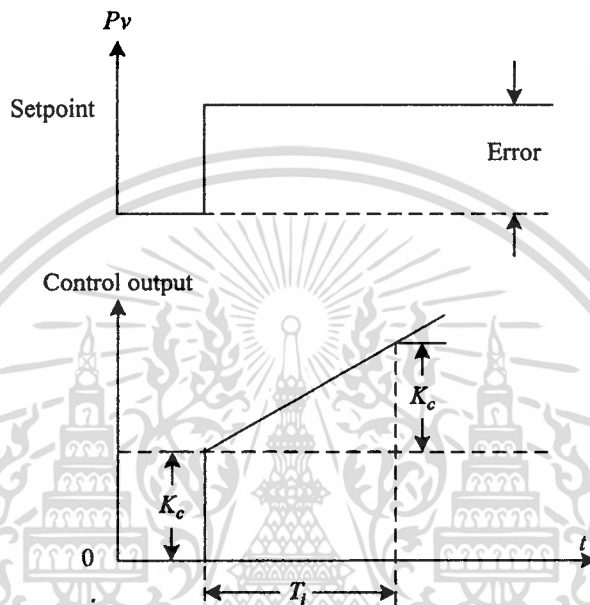
$$m_{PI}(t) = \bar{m} + K_P e(t) + K_P K_I \int_0^t e(t) dt \quad (2.7)$$

หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$m_{PI}(t) = \bar{m} + K_c e(t) + \frac{K_c}{T_i} \int_0^t e(t) dt \quad (2.8)$$

เมื่อ $K_c = K_p$ และ $K_I = 1/T_i$
 $T_i =$ ค่าเวลา Integral



รูปที่ 2.6 แสดงตัวอย่างผลตอบสนองของกริยาการควบคุมแบบ PI (Direct action)

2.2.6 กริยาการควบคุมแบบ Proportional-Derivative (PD)

การประยุกต์ใช้กริยาการควบคุมแบบ P ร่วมกับกริยาการควบคุมแบบ D เพื่อทำให้ผลตอบสนองของระบบรวดเร็วขึ้นแต่จะไม่มีผลโดยตรงต่อผลตอบสนองของระบบที่สภาวะคงที่ ซึ่งสมการเอาท์พุทของกริยาการควบคุมแบบ PD แสดงดังสมการต่อไปนี้

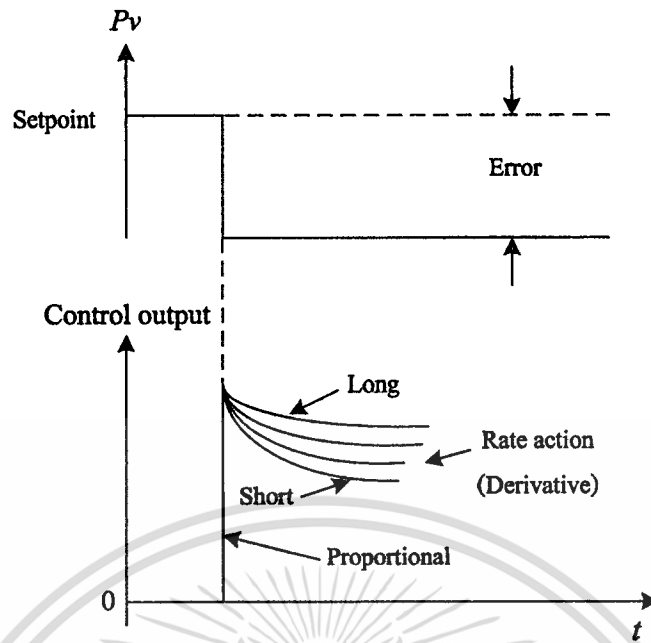
$$m_{PD}(t) = \bar{m} + K_p e(t) + K_p K_D \frac{de(t)}{dt} \quad (2.9)$$

หรือ

$$m_{PD}(t) = \bar{m} + K_c e(t) + K_c T_d \frac{de(t)}{dt} \quad (2.10)$$

เมื่อ $K_D = T_d$
 $T_d =$ ค่าเวลา Derivative

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 แสดงตัวอย่างผลตอบสนองของกริยาการควบคุมแบบ PD

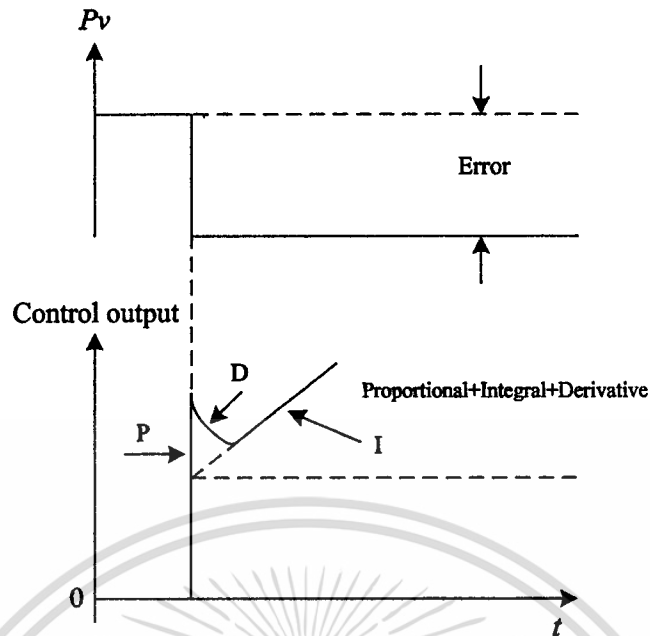
2.2.7 กริยาการควบคุมแบบ Proportional-Integral-Derivative (PID)

จากที่ได้กล่าวมาแล้ว เพื่อให้ผลตอบสนองของระบบควบคุมมีสมรรถนะเป็นไปตามต้องการจึงใช้กริยาการควบคุมทั้ง 3 แบบร่วมกัน ซึ่งจะทำให้ได้กริยาการควบคุมแบบ PID ที่มีสมการดังต่อไปนี้

$$m_{PID}(t) = \bar{m} + K_P e(t) + K_P K_I \int_0^t e(t) dt + K_P K_D \frac{de(t)}{dt} \quad (2.11)$$

หรือ

$$m_{PID}(t) = \bar{m} + K_c e(t) + \frac{K_c}{T_i} \int_0^t e(t) dt + K_c T_d \frac{de(t)}{dt} \quad (2.12)$$



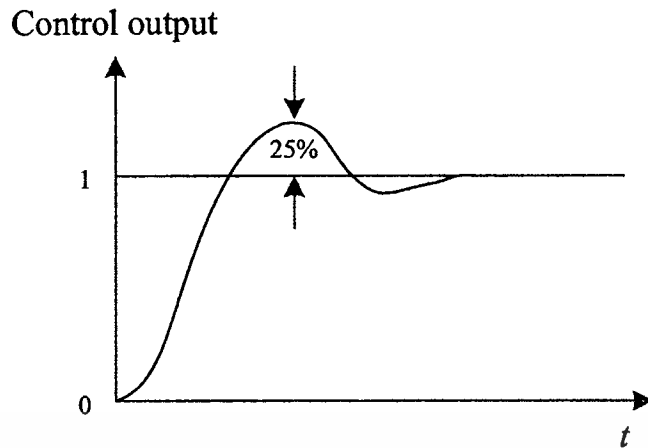
รูปที่ 2.8 แสดงตัวอย่างผลตอบสนองของกริยาการควบคุมแบบ PID

2.3 การปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี

การนำตัวควบคุมพีไอดีไปใช้ในการควบคุมกระบวนการต่างๆ จำเป็นต้องปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดีให้เหมาะสม เพื่อให้ได้ผลตอบสนองที่ดีตามความต้องการ ซึ่งการหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีนี้สามารถทำได้หลายวิธี บางวิธีพิจารณาจากผลตอบสนองของกระบวนการจากการทดสอบกระบวนการและอาศัยประสบการณ์ บางวิธีจะอาศัยทฤษฎีทางคณิตศาสตร์ และส่วนใหญ่จะพิจารณาจากเงื่อนไขในโดเมนเวลามากกว่าในโดเมนความถี่ ซึ่งวิธีของ Ziegler-Nichols เป็นวิธีที่ได้รับความนิยมมากที่สุดในทางปฏิบัติ

2.3.1 การปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี โดยวิธีของ Ziegler-Nichols

การหาค่า K_p , T_i และ T_d จะขึ้นอยู่กับคุณลักษณะของผลตอบสนองชั่วคราวของกระบวนการที่ถูกควบคุม ซึ่งมีอยู่ 2 วิธี แต่ละวิธีมีจุดมุ่งหมายที่จะทำให้ผลตอบสนองต่อเวลาของกระบวนการต่ออินพุตแบบขั้นบันได มีค่าพุ่งเกินสูงสุดไม่เกิน 25% ดังรูปที่ 2.9



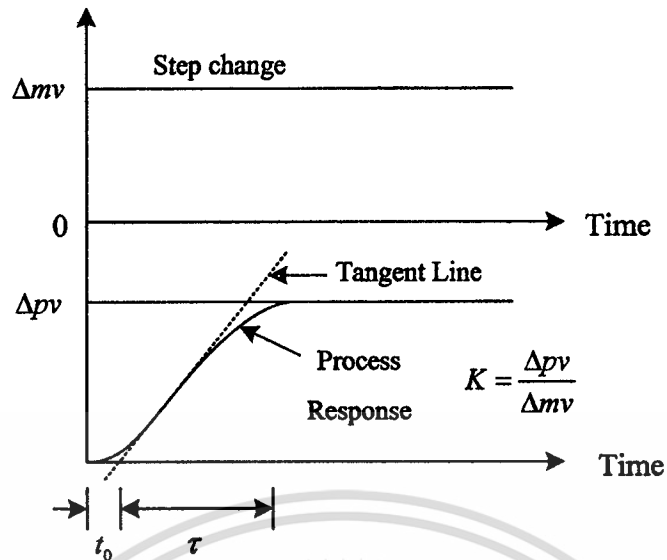
รูปที่ 2.9 แสดงค่าพ่วงเกินที่ต้องการเมื่อใช้การปรับด้วยวิธี Ziegler - Nichols

2.3.1.1 วิธี Process Reaction Curve

วิธีนี้จะหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีจากผลตอบสนองการของกระบวนการแบบรูปเปิดที่ถูกควบคุมด้วยสัญญาณอินพุตแบบขั้นบันได โดยที่กระบวนการในกรณีนี้จะไม่มี Pole ที่จุด Origin หรือไม่มี Dominant complex-conjugate poles ดังนั้นผลตอบสนองของเวลาจะได้ในลักษณะเป็นรูปตัว S (ถ้าผลตอบสนองของเวลาไม่มีลักษณะเป็นรูปตัว S วิธีนี้จะใช้ไม่ได้) จากผลตอบสนองต่อเวลาสามารถแยกพิจารณาเป็นค่าเวลาหน่วงของกระบวนการ (Delay time : t_0) และเวลาคงที่ของกระบวนการ (Time constant : τ) โดยลากเส้นสัมผัสกับจุดที่มีอัตราการเปลี่ยนแปลงสูงสุด เส้นนี้จะตัดกับแกนเวลาและเส้นตรงของเอาท์พุท $c(t)$ ที่ $c(t) = K$

ดังนั้นฟังก์ชันถ่ายโอนของระบบจะถูกประมาณได้ด้วยระบบอันดับหนึ่งแบบมีการหน่วงเวลา (FOPDT) คือ

$$G(s) = \frac{C(s)}{U(s)} = \frac{Ke^{-t_0s}}{\tau s + 1} \quad (2.13)$$



รูปที่ 2.10 แสดงผลตอบสนองรูปตัว S เมื่อปรับ โดยใช้วิธี Process Reaction Curve

Ziegler-Nichols ได้กำหนดค่าของ K_p , T_i และ T_d สำหรับตัวควบคุมแบบต่างๆ ดังตาราง

2.1

ตารางที่ 2.1 แสดงค่าพารามิเตอร์ของตัวควบคุมแบบต่างๆ ตามวิธี Process Reaction Curve

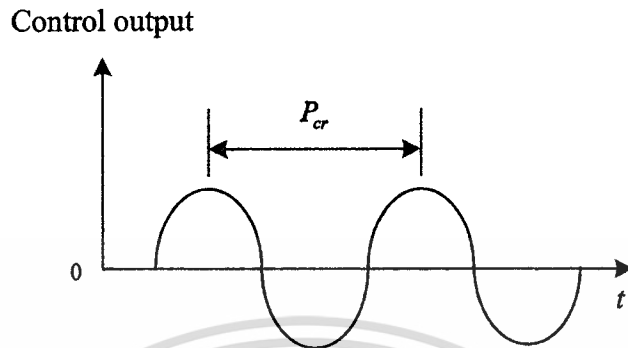
Controller Type		Proportional Gain K_c	Integral Time T_i	Derivative time T_d
Proportional only	P	$\frac{1}{K} \left(\frac{\tau}{t_0} \right)$	—	—
Proportional-Integral	PI	$\frac{0.9}{K} \left(\frac{\tau}{t_0} \right)$	$3.33 t_0$	—
Proportional-Integral-Derivative	PID	$\frac{1.2}{K} \left(\frac{\tau}{t_0} \right)$	$2.0 t_0$	$0.5 t_0$

2.3.1.2 วิธี Ultimate Method

วิธีนี้จะหาค่าพารามิเตอร์ของตัวควบคุมที่ไค้จากผลตอบสนองของกระบวนการแบบรูปปัดที่ถูกควบคุมด้วยตัวควบคุมแบบ P ต่อสัญญาณอินพุทแบบขั้นบันได โดยปรับค่า K_p

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปเรื่อยๆ จนผลตอบสนองของกระบวนการเกิดการแกว่งอย่างต่อเนื่อง (Sustained Oscillations) ถ้าผลตอบสนองเวลาไม่เกิดการแกว่งอย่างต่อเนื่อง วิธีการนี้จะใช้ไม่ได้ แสดงคังรูปที่ 2.11



รูปที่ 2.11 แสดงผลตอบสนองเวลาเกิดการแกว่งอย่างต่อเนื่อง เมื่อปรับโดยใช้วิธี Ultimate method

จากนั้นหาค่าของ

K_{cr} (Critical Gain) คือ อัตราการขยายที่ทำให้ผลตอบสนองเวลาเกิดการแกว่งอย่างต่อเนื่อง

P_{cr} (Oscillation Period) คือ คาบเวลาของการแกว่งอย่างต่อเนื่อง

Ziegler-Nichols ได้กำหนดค่าของ K_p, T_i และ T_d สำหรับตัวควบคุมแบบต่างๆ ดังตารางที่ 2.2

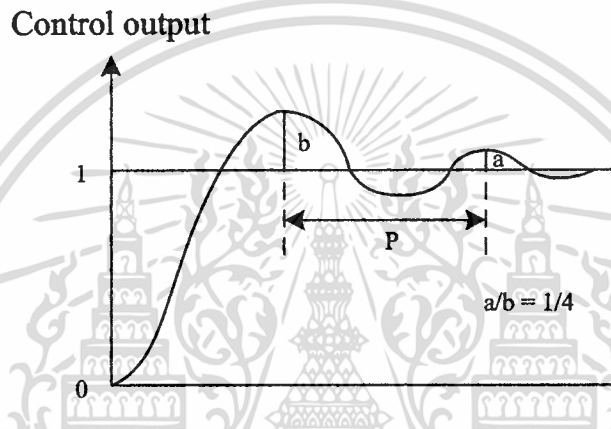
ตารางที่ 2.2 แสดงค่าพารามิเตอร์ของตัวควบคุมแบบต่างๆ ตามวิธี Ultimate Method

Controller Type		Proportional Gain K_c	Integral Time T_i	Derivative time T_d
Proportional only	P	$K_{cr} / 2$	–	–
Proportional-Integral	PI	$K_{cr} / 2.2$	$T_{cr} / 1.2$	–
Proportional-Integral-Derivative	PID	$K_{cr} / 1.7$	$T_{cr} / 2$	$T_{cr} / 8$

การคำนวณหาค่าพารามิเตอร์ของตัวควบคุม PID โดยวิธีของ Ziegler-Nichols นั้น ไม่ใช่เป็นค่าที่เที่ยงตรงที่จะนำไปใช้งานได้ทันที และมีค่าพุ่งเกินสูงสุด 25% ตามที่กล่าวไว้แต่เป็นเพียงค่าที่ใกล้เคียงเท่านั้น ผู้ควบคุมจะต้องทำการปรับค่าพารามิเตอร์เหล่านี้แบบละเอียด (Fine Tuning) อีกครั้งหนึ่ง

2.3.2 การปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี โดยวิธี Damped Oscillation

วิธีนี้ปรับปรุงมาจากวิธี Ultimate Method โดย Harriott เพื่อใช้ในกรณีที่การปรับค่าของ K_p ไปอย่างไรก็ตาม แต่ผลตอบสนองต่อเวลาไม่เกิดการแกว่งอย่างต่อเนื่อง แสดงดังรูปที่ 2.12



รูปที่ 2.12 แสดงค่าอัตราการเสื่อม 1/4 เมื่อปรับ โดยวิธี Damped Oscillation

วิธี Damped Oscillation นี้จะปรับค่าของ K_p ไปจนผลตอบสนองต่อเวลาของระบบควบคุมแบบลูปปิดมีอัตราเสื่อม $1/4$ ดังรูป จากนั้นวัดค่าของ P และใช้ค่าของ $K_p(1/4)$ เพื่อคำนวณหาค่า K_p, T_i และ T_d ดังนี้

$$K_p = K_p(1/4)$$

$$T_i = P/1.5$$

$$T_d = P/6$$

(2.14)

2.4 การปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี โดยวิธี Automatic Tuning

จากที่ได้กล่าวมาแล้ว การควบคุมกระบวนการโดยใช้ตัวควบคุมพีไอดีนั้นจำเป็นต้องปรับค่าพารามิเตอร์ของตัวควบคุมเพื่อให้ได้ผลตอบสนองที่ติดตามความต้องการ โดยอาศัยประสบการณ์จากผู้ใช้งาน วิธีการ Automatic Tuning หรือเรียกอีกอย่างหนึ่งว่า Auto-tuning เป็นวิธีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ในวงการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

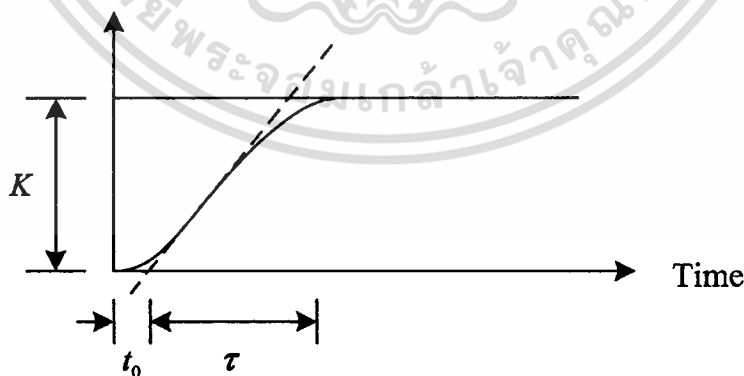
หาค่าพารามิเตอร์ของตัวควบคุมแบบอัตโนมัติ ซึ่งช่วยลดปัญหาที่ผู้ใช้งานมีประสบการณ์ไม่เพียงพอและลดระยะเวลาในการปรับค่าพารามิเตอร์ของตัวควบคุมให้สั้นลง การใช้งาน Auto-tuning นี้สามารถกระทำได้โดยผู้ใช้งานกำหนดค่าเป้าหมายให้กับตัวควบคุม หลังจากนั้นกดปุ่ม Auto-tuning เมื่อตัวควบคุมได้รับสัญญาณการทำงานในโหมด Auto-tuning แล้วนั้นตัวควบคุมจะทำการสร้างสัญญาณควบคุมป้อนให้กับกระบวนการเพื่อประมาณค่าผลตอบสนองของกระบวนการตามหลักการที่กำหนดไว้ เมื่อได้ค่าผลตอบสนองของกระบวนการแล้วจึงทำการคำนวณหาค่าพารามิเตอร์ของตัวควบคุมและปรับค่าพารามิเตอร์ของตัวควบคุมตามค่าที่ได้จากการคำนวณและตัวควบคุมทำการควบคุมกระบวนการตามค่าเป้าหมายที่กำหนดไว้ต่อไป วิธีการ Auto-tuning มีด้วยกันหลายวิธี [10] วิธีที่ได้รับความนิยมได้แก่

2.4.1 Transient response method

วิธี Transient response method นี้ทำได้โดยการป้อนสัญญาณควบคุมแบบขั้นบันไดให้กับกระบวนการแบบรูปเปิด โดยฟังก์ชันถ่ายโอนของกระบวนการจะถูกอนุมานให้อยู่ในรูปของกระบวนการอันดับหนึ่งแบบมีการหน่วงเวลา (First order plus dead time: FOPDT) คือ

$$G(s) = \frac{C(s)}{U(s)} = \frac{Ke^{-t_0s}}{\tau s + 1} \quad (2.15)$$

ผลตอบสนองของกระบวนการแบบ FOPDT หลังจากป้อนสัญญาณควบคุมแบบขั้นบันได แสดงดังรูปที่ 2.13



รูปที่ 2.13 แสดงผลตอบสนองของกระบวนการแบบ FOPDT หลังจากป้อนสัญญาณแบบขั้นบันได

สำหรับวิธีนี้ Ziegler-Nichols ได้กำหนดค่าสำหรับตัวควบคุมแบบต่างๆ ไว้ในตารางที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงแก้ไข หรือต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 แสดงค่าพารามิเตอร์ของตัวควบคุมแบบต่างๆ ตามวิธี Ziegler-Nichols

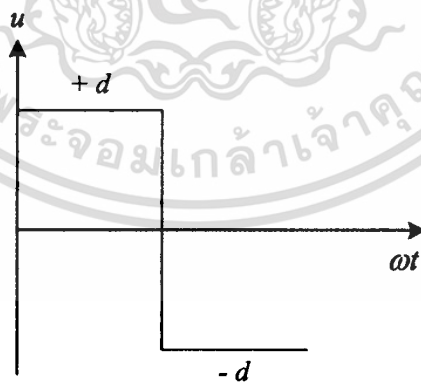
แบบ Step response method

Controller Type		Proportional Gain K_c	Integral Time T_i	Derivative time T_d
Proportional only	P	$\frac{1}{K} \left(\frac{\tau}{t_0} \right)$	–	–
Proportional-Integral	PI	$\frac{0.9}{K} \left(\frac{\tau}{t_0} \right)$	$3 t_0$	–
Proportional-Integral-Derivative	PID	$\frac{1.2}{K} \left(\frac{\tau}{t_0} \right)$	$2 t_0$	$0.5 t_0$

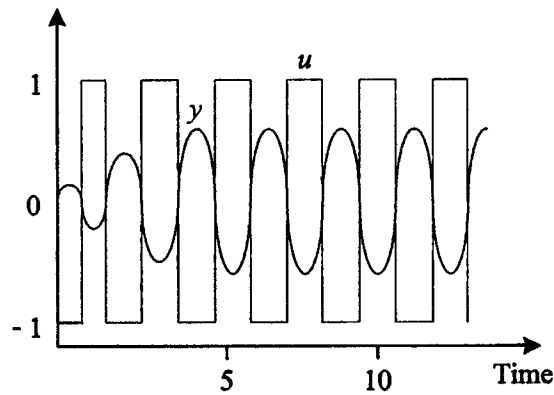
โดย $0.1 \leq \frac{t_0}{\tau} \leq 1.0$

2.4.2 Closed loop method based on relay feedback

วิธี Closed loop method based on relay feedback นี้ทำได้โดยการป้อนสัญญาณรีเลย์ (Relay) ที่มีการแกว่งแบบต่อเนื่องให้กับกระบวนการแบบรูปปิด สัญญาณ Relay แสดงดังรูปที่ 2.14



รูปที่ 2.14 แสดงสัญญาณ Relay ที่มีการแกว่งอย่างต่อเนื่องที่ป้อนให้กับกระบวนการ



รูปที่ 2.15 แสดงสัญญาณ Relay และผลตอบสนองของกระบวนการ

กำหนดให้สัญญาณ Relay เป็นสัญญาณสมมาตรแบบ Square wave ที่การแกว่งอย่าง ต่อเนื่อง กำหนดให้ โดย $u_c = 0$ และ d คือ ขนาดของสัญญาณ Relay ดังนั้นสัญญาณคลาดเคลื่อน จะมีขนาด คือ

$$a = \frac{4d}{\pi} |G(i\omega_c)| \quad (2.16)$$

เงื่อนไขการแกว่งอย่างต่อเนื่อง คือ

$$\arg G(i\omega_c) = -\pi \text{ และ } K_u = \frac{4d}{\pi a} = \frac{1}{|G(i\omega_c)|} \quad (2.17)$$

โดย

K_u (Ultimate gain) คือ อัตราการขยายที่ทำให้เกิดผลตอบสนองที่มีการแกว่ง อย่างต่อเนื่อง

T_u (Ultimate period) คือ คาบเวลาของการแกว่งอย่างต่อเนื่อง

สำหรับวิธีนี้ Ziegler-Nichols ได้กำหนดค่าสำหรับตัวควบคุมแบบต่างๆ ไว้ในตารางที่ 2.4

ตารางที่ 2.4 แสดงค่าพารามิเตอร์ของตัวควบคุมแบบต่างๆ ตามวิธี Ziegler-Nichols

แบบ Closed-loop method

Controller Type		Proportional Gain K_c	Integral Time T_i	Derivative time T_d
Proportional only	P	$0.5K_u$	–	–
Proportional-Integral	PI	$0.4K_u$	$0.8T_u$	–
Proportional-Integral-Derivative	PID	$0.6K_u$	$0.5T_u$	$0.12T_u$

โดย $2 \leq kK_u \leq 20$ เมื่อ k คือ อัตราการขยายคงที่ของกระบวนการ (Static process gain)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

หลักการและทฤษฎีที่ใช้ในการออกแบบ

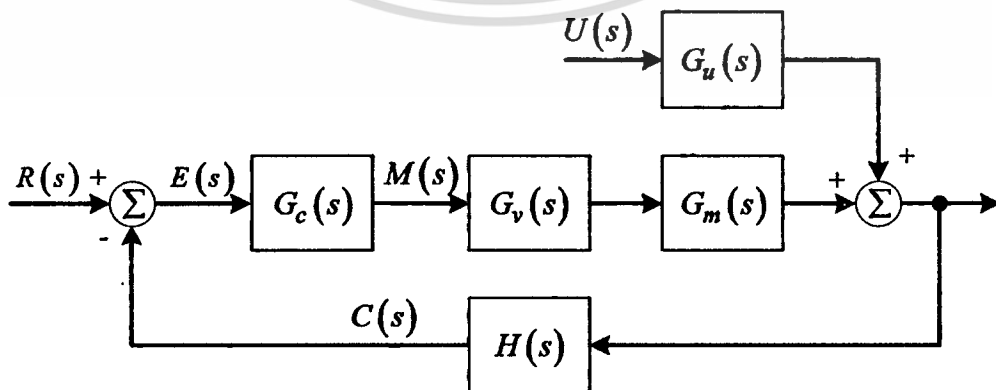
3.1 การหาค่าคุณลักษณะของกระบวนการ

การวิเคราะห์และการออกแบบการควบคุมกระบวนการอุตสาหกรรม จำเป็นต้องทราบคุณลักษณะหรือคุณสมบัติของกระบวนการและอุปกรณ์ต่างๆ ที่ใช้ในการควบคุมกระบวนการทั้งหมด เพื่อใช้ศึกษาและวิเคราะห์คุณสมบัติการเปลี่ยนแปลงสภาพของกระบวนการ เพื่อที่จะได้เลือกการควบคุมและค่าพารามิเตอร์ของตัวควบคุมที่เหมาะสม โดยทั่วไปการศึกษาคุณลักษณะของกระบวนการที่ใช้กันอยู่ 2 วิธี คือ

1. การศึกษาคุณลักษณะของกระบวนการ โดยใช้วิธีการทดสอบ วิธีนี้การวิเคราะห์กระบวนการสามารถทำได้โดยใช้เครื่องมือและเครื่องควบคุมทดลองป้อนสัญญาณเข้าและสังเกตผลตอบสนองของสัญญาณขาออกของกระบวนการและอุปกรณ์ต่างๆ ซึ่งผลของสัญญาณขาออกนี้จะนำไปประมาณค่าพารามิเตอร์ของกระบวนการ เพื่ออธิบายคุณลักษณะของกระบวนการต่อไป

2. การศึกษาคุณลักษณะของกระบวนการ โดยใช้วิธีการวิเคราะห์ทางคณิตศาสตร์ วิธีนี้การวิเคราะห์คุณลักษณะของอุปกรณ์และกระบวนการจะถูกแสดงในลักษณะของสมการคณิตศาสตร์ต่างๆ เช่น สมการพีชคณิต สมการอนุพันธ์ (Differential Equation) และสมการความแตกต่าง (Difference Equation) และวิเคราะห์คุณสมบัติของกระบวนการจากการหาค่าตอบของสมการเหล่านี้

การศึกษาคุณลักษณะของกระบวนการ โดยใช้วิธีการวิเคราะห์ทางคณิตศาสตร์เป็นวิธีที่ยุ่งยาก ดังนั้นทางปฏิบัตินิยมใช้วิธีการศึกษาคุณลักษณะของกระบวนการ โดยใช้วิธีการทดสอบ ซึ่งโครงสร้างของระบบควบคุมแบบป้อนกลับโดยทั่วไป แสดงดังรูปที่ 3.1

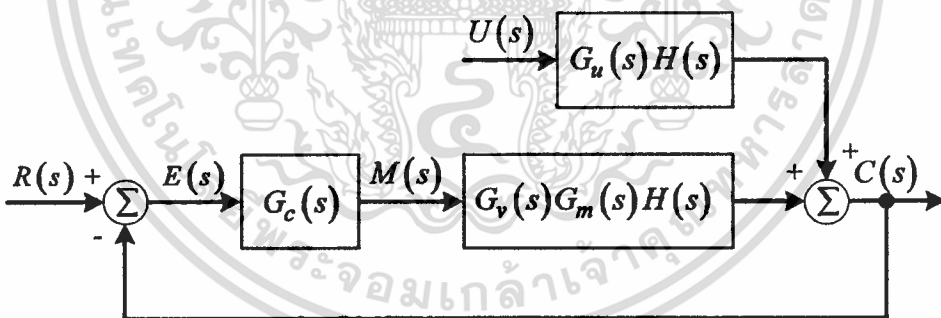


รูปที่ 3.1 แสดงโครงสร้างของระบบควบคุมแบบป้อนกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เมื่อ $R(s)$ = ค่าสัญญาณอินพุตอ้างอิงหรือค่าเป้าหมาย
 $C(s)$ = ค่าสัญญาณเอาต์พุตของอุปกรณ์แปลงสัญญาณ
 $E(s)$ = ค่าสัญญาณความคลาดเคลื่อน
 $M(s)$ = ค่าสัญญาณเอาต์พุตของตัวควบคุม
 $U(s)$ = ค่าสัญญาณรบกวน
 $G_c(s)$ = ฟังก์ชันถ่ายโอนของตัวควบคุม
 $G_v(s)$ = ฟังก์ชันถ่ายโอนของอุปกรณ์ควบคุมสุดท้าย
 $G_m(s)$ = ฟังก์ชันถ่ายโอนของกระบวนการระหว่างสัญญาณเอาต์พุตของกระบวนการ (Controlled variable) และตัวแปรปรับกระบวนการ (Manipulated variable)
 $G_u(s)$ = ฟังก์ชันถ่ายโอนของกระบวนการระหว่างสัญญาณเอาต์พุตของกระบวนการ (Controlled variable) และสัญญาณรบกวน (Disturbance)
 $H(s)$ = ฟังก์ชันถ่ายโอนของอุปกรณ์ตรวจจับสัญญาณและอุปกรณ์แปลงสัญญาณ

จากรูปที่ 3.1 สามารถลดรูปให้เข้าใจง่าย ดังรูปที่ 3.2



รูปที่ 3.2 แสดงบล็อกโคอะแกรมของระบบควบคุมแบบป้อนกลับหลังจากการลดรูป

กำหนดให้

$$G(s) = G_v(s)G_m(s)H(s) \quad (3.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทั่วไปฟังก์ชันถ่ายโอนของกระบวนการจะถูกอนุมานให้อยู่ในรูปแบบของกระบวนการอันดับหนึ่งที่มีการหน่วงเวลา (First-Order Plus Dead Time: FOPDT) ซึ่งฟังก์ชันถ่ายโอนของกระบวนการแบบ FOPDT แสดงดังสมการ (3.2)

$$G(s) = \frac{Ke^{-t_0s}}{\tau s + 1} \quad (3.2)$$

เมื่อ K = อัตราขยายของกระบวนการ (Process Gain)

t_0 = ค่าการหน่วงเวลาของกระบวนการ (Process Dead Time)

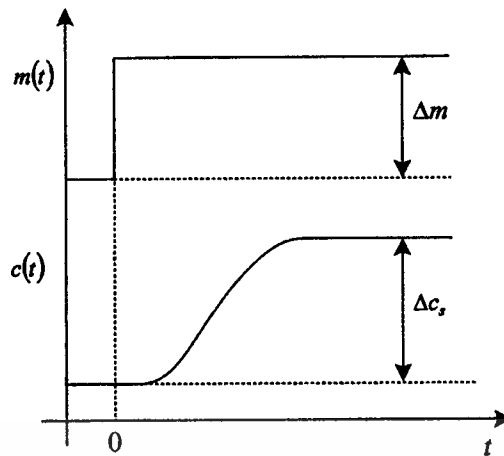
τ = ค่าเวลาคงที่ของกระบวนการ (Process Time Constant)

จากสมการ (3.2) จะเห็นได้ว่าคุณลักษณะของกระบวนการอยู่ในรูปตัวแปร 3 ตัวแปร ซึ่งสามารถหาได้จากการทดสอบกระบวนการแบบรูปเปิด (Opened-loop) โดยทำการป้อนสัญญาณขั้นบันได $M(s)$ และไม่พิจารณาสัญญาณรบกวน ดังรูปที่ 3.3



รูปที่ 3.3 แสดงบล็อกไดอะแกรมของการทดสอบกระบวนการแบบรูปเปิด

จากการทดสอบจะได้ผลตอบสนองของกระบวนการมีลักษณะเป็นรูปตัว S ดังรูปที่ 3.4 ซึ่งรูปแบบของผลตอบสนองนี้ครอบคลุมไปถึงกระบวนการอันดับสองและกระบวนการอันดับสูง (Second-Order and Higher-Order Process) ที่มีอัตราหน่วง (Damping Ratio) มากกว่าหรือเท่ากับหนึ่งได้



รูปที่ 3.4 แสดงผลตอบสนองของกระบวนการแบบ FOPDT หลังจากป้อนสัญญาณแบบขั้นบันได

จากรูปที่ 3.1 จะได้ฟังก์ชันถ่ายโอนของผลตอบสนองของกระบวนการ คือ

$$C(s) = G(s)M(s) \quad (3.3)$$

พิจารณการเปลี่ยนแปลงค่าสัญญาณควบคุมขนาด Δm และกระบวนการแบบ FOPDT ในรูปแบบของสมการลาปลาซ จะได้

$$\begin{aligned} C(s) &= \frac{Ke^{-t_0s}}{\tau s + 1} \cdot \frac{\Delta m}{s} \\ &= K \Delta m e^{-t_0s} \left[\frac{1}{s} - \frac{\tau}{\tau s + 1} \right] \end{aligned} \quad (3.4)$$

แปลงผกผันลาปลาซให้อยู่ในโดเมนเวลา จะได้

$$\Delta c(t) = K \Delta m u(t-t_0) \left[1 - e^{-(t-t_0)/\tau} \right] \quad (3.5)$$

โดย $\Delta c(t) = c(t) - c(0)$ เป็นการเปลี่ยนแปลงของสัญญาณเอาต์พุตของกระบวนการ จากค่าสถานะเริ่มต้น (Initial Value) ส่วนฟังก์ชันขั้นบันได $u(t-t_0)$ เป็นพจน์ที่กำหนดที่ทำให้ $\Delta c(t) = 0$ ที่เวลา $t \leq t_0$ ดังนั้นเมื่อผลตอบสนองของกระบวนการเข้าสู่สถานะคงที่ (Steady State) จะได้

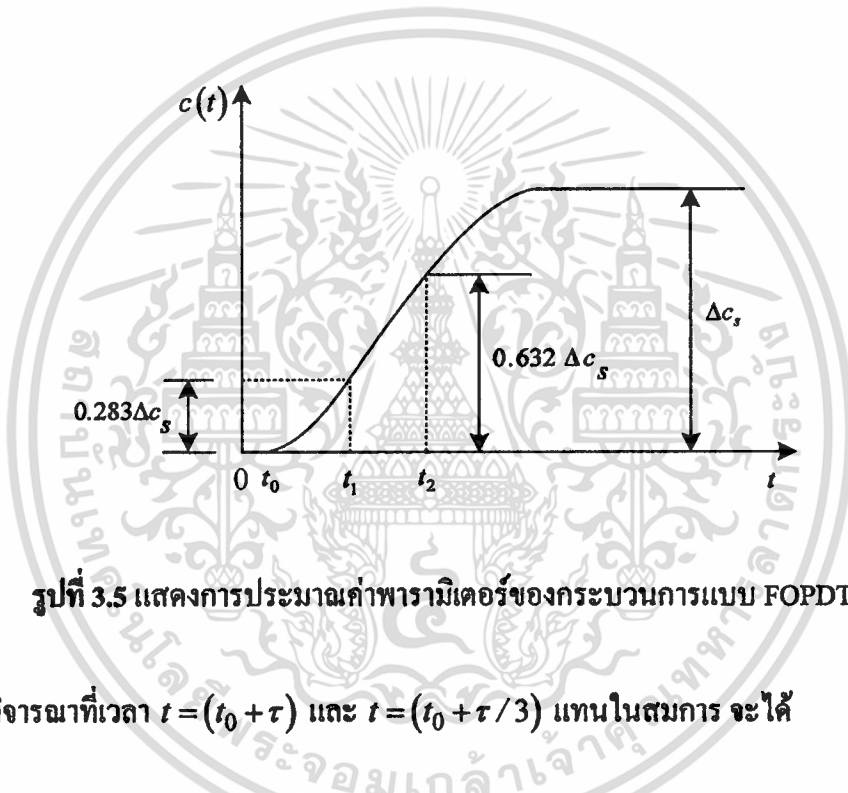
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\Delta c_s = \lim_{t \rightarrow \infty} \Delta c(t) = K \Delta m \quad (3.6)$$

ดังนั้นสามารถหาค่าคุณลักษณะของกระบวนการ คือ อัตราขยายของกระบวนการ คือ

$$K = \frac{\Delta c_s}{\Delta m} \quad (3.7)$$

ค่าคุณลักษณะที่เหลือคือ τ และ t_0 สามารถหาได้โดยการประมาณค่า (Curve Fitting) ดังรูปที่ 3.5



รูปที่ 3.5 แสดงการประมาณค่าพารามิเตอร์ของกระบวนการแบบ FOPDT

พิจารณาที่เวลา $t = (t_0 + \tau)$ และ $t = (t_0 + \tau/3)$ แทนในสมการ จะได้

$$\begin{aligned} \Delta c(t_0 + \tau) &= K \Delta m (1 - e^{-1}) = 0.632 \Delta c_s \\ \Delta c\left(t_0 + \frac{\tau}{3}\right) &= K \Delta m (1 - e^{-1/3}) = 0.283 \Delta c_s \end{aligned} \quad (3.8)$$

กำหนดให้เวลาที่ $t = (t_0 + \tau/3)$ และ $t = (t_0 + \tau)$ เป็น t_1 และ t_2 ตามลำดับ คือ

$$\begin{aligned} t_1 &= t_0 + \frac{\tau}{3} \\ t_2 &= t_0 + \tau \end{aligned} \quad (3.9)$$

ดังนั้น

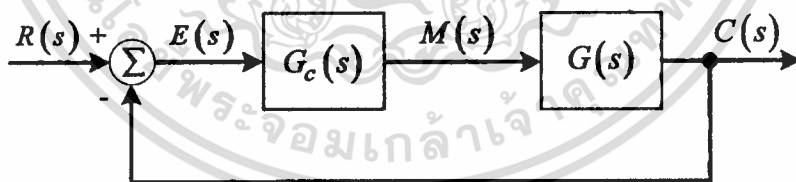
$$\begin{aligned} t_0 &= t_2 - \tau \\ \tau &= \frac{3}{2}(t_2 - t_1) \end{aligned} \quad (3.10)$$

เมื่อ $t_1 =$ ค่าเวลาคงที่ $\Delta c = 0.283 \Delta c_s$
 $t_2 =$ ค่าเวลาคงที่ $\Delta c = 0.632 \Delta c_s$

ค่าคุณลักษณะของกระบวนการทั้ง 3 ค่า คือ K, τ และ t_0 นี้จะนำไปใช้ในการสังเคราะห์ค่าพารามิเตอร์ของตัวควบคุมพีไอคือ ค่า K_c, T_i และ T_d ที่จะใช้ปรับค่าให้กับตัวควบคุม ดังที่จะกล่าวต่อไป

3.2 การสังเคราะห์ค่าพารามิเตอร์ของตัวควบคุมพีไอตามหลักการของค่าหัดิน

การหาค่าคุณลักษณะของกระบวนการ โดยการทดสอบคั้งที่กล่าวมา จะพิจารณาในลักษณะการควบคุมแบบรูปเปิด ซึ่งต่างจากการสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอที่ทำการพิจารณาในลักษณะการควบคุมแบบรูปปิด เนื่องจากต้องพิจารณารวมไปถึงอุปกรณ์ควบคุมสุดท้ายด้วยดังนั้น จากรูปที่ 3.1 ระบบควบคุมแบบรูปปิด เมื่อไม่พิจารณาสัญญาณรบกวนจะได้ดังรูปที่ 3.6



รูปที่ 3.6 แสดงบล็อกไดอะแกรมของระบบแบบรูปปิดเมื่อไม่พิจารณาสัญญาณรบกวน

จากรูป ฟังก์ชันถ่ายโอนแบบรูปปิด คือ

$$\frac{C(s)}{R(s)} = \frac{G_c(s)G(s)}{1 + G_c(s)G(s)} \quad (3.11)$$

จากสมการ (3.11) สามารถเขียนให้อยู่ในรูปของฟังก์ชันถ่ายโอนของตัวควบคุมได้ คือ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์ การนำเอกสารนี้ไปใช้โดยไม่ขออนุญาตถือว่าผิดกฎหมาย และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

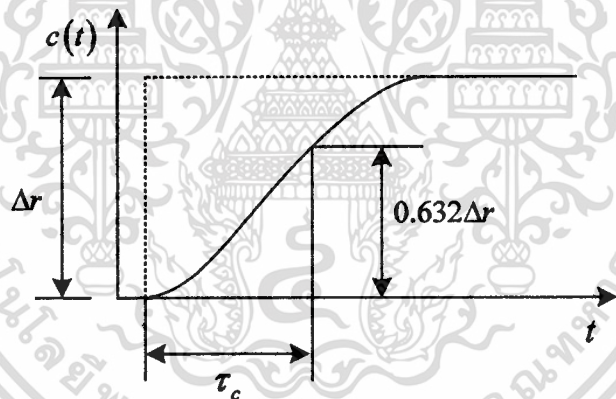
$$G_c(s) = \frac{1}{G(s)} \cdot \frac{C(s)/R(s)}{1 - [C(s)/R(s)]} \quad (3.12)$$

ฟังก์ชันถ่ายโอนที่ได้นี้จะใช้สำหรับสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมต่อไป

ในการควบคุมกระบวนการหนึ่งๆ ผู้ควบคุมต้องการให้กระบวนการมีผลตอบสนองแบบรูปปิดต่อการควบคุมตามต้องการ ซึ่งผลตอบสนองแบบรูปปิดของระบบควบคุมที่เข้าใกล้ความต้องการของผู้ควบคุมมากที่สุด คือ ระบบควบคุมแบบรูปปิดที่ให้ผลตอบสนองเป็นแบบ First Order Lag ที่มีฟังก์ชันถ่ายโอน

$$\frac{C(s)}{R(s)} = \frac{1}{\tau_c s + 1} \quad (3.13)$$

เมื่อ τ_c คือ ค่าเวลาคงที่ของผลตอบสนองแบบรูปปิด



รูปที่ 3.7 แสดงผลตอบสนองแบบรูปปิดที่ใช้ในการสังเคราะห์ค่าพารามิเตอร์ PID

จากรูปที่ 3.7 จะเห็นได้ว่า τ_c เป็นพารามิเตอร์ที่ปรับค่าได้ ซึ่งถ้าหากค่า τ_c มีค่าน้อย ผลตอบสนองก็จะเข้าใกล้ค่าเป้าหมายเร็วขึ้น ดังนั้นเมื่อแทนสมการ (3.13) ในสูตรสังเคราะห์ค่าพารามิเตอร์ของตัวควบคุม จะได้

$$G_c(s) = \frac{1}{G(s)} \cdot \frac{\frac{1}{\tau_c s + 1}}{1 - \frac{1}{\tau_c s + 1}} = \frac{1}{G(s)} \cdot \frac{1}{\tau_c s + 1 - 1}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือ

$$G_c(s) = \frac{1}{G(s)} \cdot \frac{1}{\tau_c s} \quad (3.14)$$

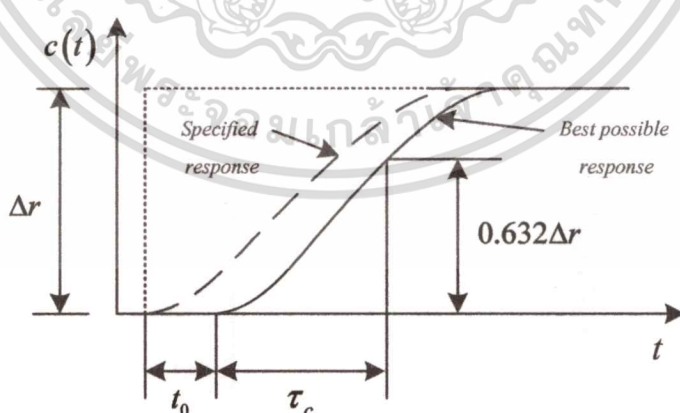
สมการ (3.14) มีพจน์ของอินทิกรัลอยู่ในฟังก์ชันถ่ายโอน ซึ่งจะส่งผลให้สามารถควบคุมกระบวนการได้โดยไม่มีค่าออฟเซ็ท

แทนค่าสมการ (3.2) ในสมการ (3.14) เพื่อสังเคราะห์ค่าพารามิเตอร์ของตัวควบคุมสำหรับกระบวนการแบบ FOPDT จะได้

$$\begin{aligned} G_c(s) &= \frac{\tau s + 1}{K e^{-t_0 s}} \cdot \frac{1}{\tau_c s} \\ &= \frac{\tau}{K \tau_c} \left(1 + \frac{1}{\tau s} \right) e^{t_0 s} \end{aligned} \quad (3.15)$$

พิจารณาสมการ (3.15) ในโดเมนจะพบว่าค่าการหน่วงเวลา t_0 เป็นลบ ส่งผลให้สามารถทำให้ตัวควบคุมที่เป็นจริงได้ เนื่องจากต้องการการทำนายค่าเวลา t_0 ล่วงหน้า ดังนั้นจึงต้องพิจารณาเทียบกับผลตอบสนองแบบปิด ที่มีการหน่วงเวลาเป็นเวลา t_0 คือ

$$\frac{C(s)}{R(s)} = \frac{e^{-t_0 s}}{\tau_c s + 1} \quad (3.16)$$



รูปที่ 3.8 แสดงผลตอบสนองแบบรูปปิดที่มีการหน่วงเวลาเป็นเวลา t_0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นจากสมการ (3.2), (3.12) และ (3.16) จะได้ฟังก์ชันถ่ายโอนสำหรับการสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมตามกระบวนการแบบ FOPDT คือ

$$G_c(s) = \frac{\tau s + 1}{K e^{-t_0 s}} \cdot \frac{e^{-t_0 s}}{\tau_c s + 1 - e^{-t_0 s}}$$

หรือ

$$G_c(s) = \frac{\tau s + 1}{K} \cdot \frac{1}{\tau_c s + 1 - e^{-t_0 s}} \quad (3.17)$$

เทอมเอ็กซ์โปเนนเชียล ($e^{-t_0 s}$) สามารถประยุกต์ค่าได้โดยอนุกรมเทย์เลอร์ (Taylor Series Expansion) คือ

$$e^{-t_0 s} = 1 - t_0 s + \frac{1}{2}(t_0 s)^2 - \frac{1}{3!}(t_0 s)^3 + \dots \quad (3.18)$$

ใช้เฉพาะ 2 เทอมแรก คือ $1 - t_0 s$ (สำหรับกระบวนการอันดับหนึ่ง) แทนในสมการ (3.17) จะได้

$$\begin{aligned} G_c(s) &= \frac{\tau s + 1}{K} \cdot \frac{1}{(\tau_c + t_0)s} \\ &= \frac{\tau}{K(\tau_c + t_0)} \cdot \left(1 + \frac{1}{\tau s}\right) \end{aligned} \quad (3.19)$$

จากสมการ (3.19) แสดงให้เห็นว่าจากกระบวนการที่เป็นแบบ FOPDT เมื่อนำมาหาฟังก์ชันถ่ายโอนของตัวควบคุมจะได้เป็นตัวควบคุมแบบ PI เท่านั้น โดยมีค่าพารามิเตอร์ คือ

$$K_c = \frac{\tau}{K(\tau_c + t_0)} \quad \text{และ} \quad T_i = \tau \quad (3.20)$$

แต่จากการสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุม PID แบบรูปปิดตามหลักการของ Dahlin ตามกระบวนการแบบ FOPDT สามารถทำเป็นตัวควบคุมแบบ PID ที่มีค่าพารามิเตอร์ของตัวควบคุม [11] ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$K_c = \frac{\tau}{K(\tau_c + t_0)}, T_i = \tau \quad \text{และ} \quad T_d = \frac{t_0}{2} \quad (3.21)$$

ข้อเสนอแนะ สูตรหาค่าพารามิเตอร์ของตัวควบคุมแบบ PID ของ Dahlin นี้ ในการใช้งานโหมด PID ควรใช้เมื่อค่า t_0 มากกว่า $\tau/4$ กรณีนอกเหนือจากนี้ ควรใช้การควบคุมแบบ PI และสำหรับความต้องการควบคุมให้ค่าพุ่งเกินมีค่าไม่เกิน 5% สำหรับการเปลี่ยนแปลงค่าเป้าหมาย แนะนำให้ $\tau_c = t_0$ ดังนั้นที่ความต้องการค่าพุ่งเกินมีค่าไม่เกิน 5% ได้ดังนี้

$$K_c = \frac{\tau}{K(\tau_c + t_0)} = \frac{0.5}{K} \left(\frac{\tau}{t_0} \right) \quad (3.22)$$

โหมดการควบคุมและสูตรหาค่าพารามิเตอร์ของตัวควบคุมแบบ PID ของ Dahlin แสดงดังตารางที่ 3.1

ตารางที่ 3.1 แสดงสูตรสำหรับหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีของ Dahlin

Process	Controller Type	Tuning Parameters	
$G(s) = K$	P	$K_c = \frac{1}{K\tau_c}$	Tunable
$G(s) = \frac{K}{\tau s + 1}$	PI	$K_c = \frac{1}{K\tau_c}, T_i = \tau$	Tunable
$G(s) = \frac{K}{(\tau_1 s + 1)(\tau_2 s + 1)}$ $\tau_1 > \tau_2$	PID	$K_c = \frac{\tau_1}{K\tau_c}, T_i = \tau_1, T_d = \tau_2$	Tunable
$G(s) = \frac{Ke^{-t_0 s}}{\tau s + 1}$	PID ^a	$K_c = \frac{\tau}{K(t_0 + \tau_c)}, T_i = \tau, T_d = \frac{t_0}{2}$	Tunable

^a ใช้ทั้งกับตัวควบคุมแบบ PID และ PI ($T_d = 0$) ใช้เมื่อ t_0 มากกว่า $\tau/4$

3.3 การประมาณค่าล่วงหน้า

การประมาณค่าล่วงหน้าหรือการประมาณค่านอกช่วง เป็นการหาค่าโดยประมาณของข้อมูลที่จะเกิดขึ้นในอนาคตโดยการวิเคราะห์จากข้อมูลที่ได้จากการสุ่มในช่วงเวลาต่างๆ วิธีการที่เหมาะสมในการวิเคราะห์ระเบียบวิธีการเชิงตัวเลข (Numerical Analysis) คือ วิธีผลต่างจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแบ่งย่อยของนิวตัน (Newton's Divided-Difference Polynomial) ซึ่งในงานวิจัยนี้ได้นำมาใช้เพื่อคำนวณหาค่าผลตอบสนองของกระบวนการที่สภาวะคงที่ (Δc_s) โดยใช้สมการดังต่อไปนี้

$$f(x) = f(x_0) + (x - x_0)f[x_1, x_0] + \dots + (x - x_0)(x - x_1)\dots(x - x_{n-1})f[x_n, \dots, x_0] \quad (3.23)$$

เมื่อ $f[x_1, x_0] = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$

$$f[x_n, x_{n-1}, \dots, x_1, x_0] = \frac{f[x_n, x_{n-1}, \dots, x_1] - f[x_{n-1}, x_{n-2}, \dots, x_0]}{x_n - x_0}$$

x_n คือ ข้อมูลที่ได้จากการสุ่มในแต่ละครั้ง

$f(x)$ คือ ค่าที่ได้จากการประมาณค่านอกช่วง

ในวิทยานิพนธ์นี้ได้ใช้ข้อมูลจากการสุ่มจำนวน 3 ค่า โดยกำหนดให้เป็น x_0, x_1 และ x_2 เพื่อนำมาทำการคำนวณหาค่าข้อมูลในอนาคต 2 ค่า คือ x_3 และ x_4 ถ้าหากค่า $x_4 > x_3$ จะทำการเก็บค่าข้อมูล x_3 เพื่อนำมาทำการคำนวณหาค่าในรอบต่อไปโดยใช้ข้อมูลจาก x_1, x_2 และ x_3 ซึ่งจะคำนวณจนกระทั่ง $x_n \geq x_{n+1}$ เพราะผลตอบสนองของกระบวนการแสดงแนวโน้มที่ลดลงหรือเข้าสู่สภาวะคงที่ ซึ่งค่าของ x_n คือ Δc_s ที่จะนำไปใช้ในการคำนวณหาค่าพารามิเตอร์ของกระบวนการต่อไป ซึ่งเวลาที่ใช้ในการสุ่มข้อมูลสามารถกำหนดได้จากหน้าจอกอมพิวเตอร์ตามความต้องการ

3.4 ความรู้เบื้องต้นเกี่ยวกับเครื่องควบคุม PLC

เครื่องควบคุม PLC (Programmable Logic Controller) ถูกนำมาใช้ควบคุมกระบวนการทางอุตสาหกรรมอย่างกว้างขวาง เนื่องจากใช้งานง่าย มีความยืดหยุ่นในการทำงานสูง และมีความน่าเชื่อถือสูง เริ่มแรกเครื่องควบคุม PLC ถูกออกแบบขึ้นมาเพื่อประยุกต์ใช้งานกับระบบควบคุมแบบลำดับขั้น (Sequence control system) เพื่อใช้ทดแทนระบบควบคุมแบบเก่าที่ใช้อุปกรณ์ไฟฟ้าเชิงกล (Electromechanical device) เช่น รีเลย์ (Relay) ตัวตั้งเวลา (Timer) และตัวนับ (Counter) เป็นต้น โดยเครื่องควบคุม PLC ได้ถูกพัฒนาขึ้นอย่างต่อเนื่องเพื่อให้มีความสามารถและขอบเขตการใช้งานกว้างขวางขึ้น ซึ่งมีโมดูลอินพุตเอาต์พุตแบบอนาลอกหรืออินพุตเอาต์พุตแบบพิเศษเพิ่มมากขึ้น เพื่อนำประยุกต์ใช้กับงานพิเศษแบบต่างๆ และมีฟังก์ชันการควบคุม เช่น PID control, Fuzzy control เป็นต้น

3.4.1 โครงสร้างและหลักการการทำงานของเครื่องควบคุม PLC

โครงสร้างของเครื่องควบคุม PLC

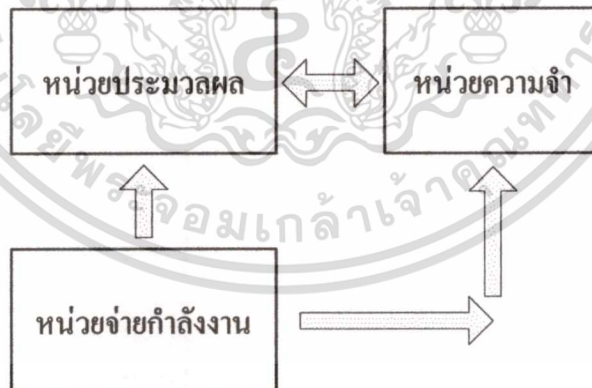
เครื่องควบคุม PLC เป็นอุปกรณ์อิเล็กทรอนิกส์ทำหน้าที่ควบคุมกระบวนการตามโปรแกรมที่ผู้ใช้งานกำหนดไว้ โดยโปรแกรมจะถูกเก็บไว้ในหน่วยความจำ ซึ่งโครงสร้างของเครื่องควบคุม PLC แสดงดังรูปที่ 3.9



รูปที่ 3.9 แสดงโครงสร้างโดยทั่วไปของเครื่องควบคุม PLC

เครื่องควบคุม PLC โดยทั่วไปประกอบด้วย 2 ส่วน คือ

1. หน่วยประมวลผลกลางหรือ CPU (Central Processing Unit) หน่วยประมวลผลกลางทำหน้าที่ควบคุมการทำงานทั้งหมดของเครื่องควบคุม PLC โดยรับคำสั่งสถานะต่างๆ ผ่านหน่วยอินพุตและเข้ามาประมวลผลตามโปรแกรมของผู้ใช้ที่เก็บไว้ในหน่วยความจำและส่งสัญญาณเอาต์พุตผ่านทางหน่วยเอาต์พุตเพื่อไปควบคุมเครื่องจักรหรือกระบวนการ



รูปที่ 3.10 แสดงส่วนประกอบภายในหน่วยประมวลผลกลาง

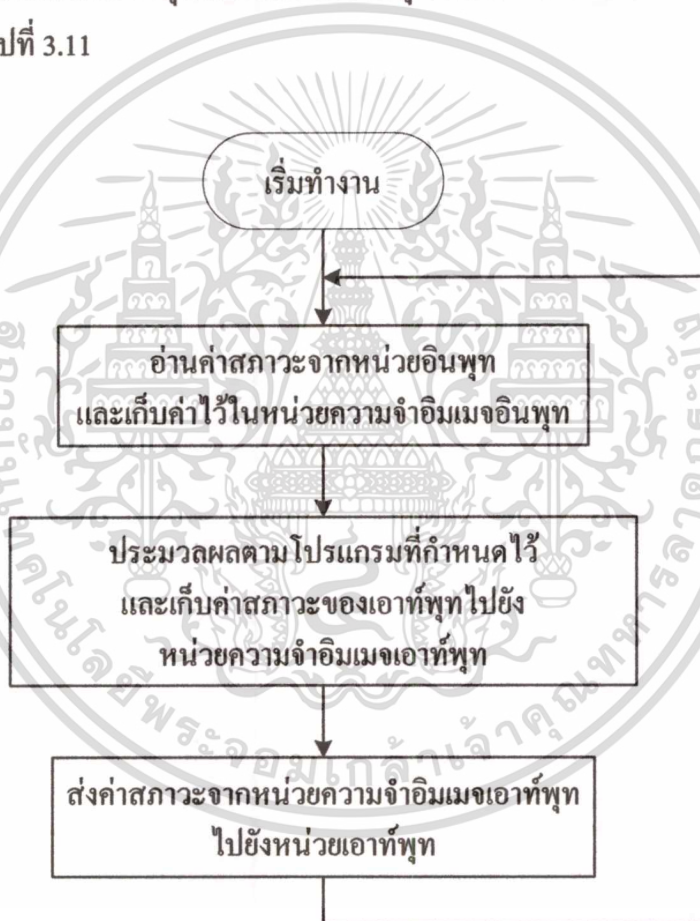
2. หน่วยอินพุตและเอาต์พุต (Input and Output Unit) ทำหน้าที่ติดต่อระหว่างเครื่องควบคุม PLC กับเครื่องจักรหรือกระบวนการ โดยหน่วยอินพุตทำหน้าที่รับสัญญาณอินพุตจากอุปกรณ์ภายนอก ได้แก่ สัญญาณอินพุตแบบดิจิทัล เช่น สวิตช์แบบกดปุ่ม ลิimitsวิตช์ พลัสติกมิเตอร์สวิตช์ เป็นต้น และสัญญาณอินพุตแบบอนาล็อก เช่น เทอร์โมคัปเปิล อาร์ทีดี เป็นต้น ส่วนหน่วยเอาต์พุตทำหน้าที่

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ผู้ใดเห็นประโยชน์หรือมีข้อสงสัย
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่งสัญญาณออกไปขับเคลื่อนอุปกรณ์ภายนอก ได้แก่ สัญญาณเอาต์พุตแบบดิจิทัล เช่น รีเลย์ หลอดไฟ โซลินอยด์วาล์ว เป็นต้น และสัญญาณเอาต์พุตแบบอนาลอก เช่น คอนโทรลวาล์ว เป็นต้น

หลักการการทำงานของเครื่องควบคุม PLC

หลักการการทำงานของเครื่องควบคุม PLC มีขั้นตอน คือ เมื่อเริ่มต้นการทำงานของเครื่องควบคุม PLC จะทำการอ่านค่าอินพุตทั้งหมดจากหน่วยอินพุตมาเก็บเข้ายังหน่วยความจำอิมเมจอินพุต (Process image input) หน่วยประมวลผลกลางทำการประมวลผลตามโปรแกรมที่ได้เขียนไว้และเก็บค่าเอาต์พุตเข้ายังหน่วยความจำอิมเมจเอาต์พุต (Process image output) และส่งค่าเอาต์พุตจากหน่วยความจำอิมเมจเอาต์พุตไปยังหน่วยเอาต์พุต หลังจากนั้นการทำงานจะเป็นแบบวงรอบต่อไป แสดงดังรูปที่ 3.11



รูปที่ 3.11 แสดงวงรอบการทำงานของเครื่องควบคุม PLC

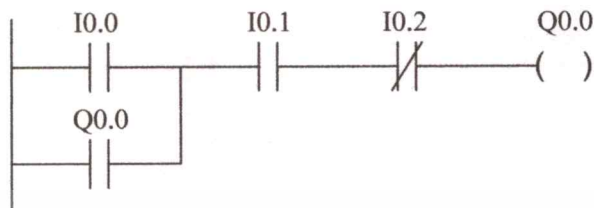
3.4.2 ภาษาที่ใช้ในการเขียนโปรแกรมของเครื่องควบคุม PLC ตามมาตรฐาน IEC1131-3

ภาษาที่ใช้ในการเขียนโปรแกรมของเครื่องควบคุม PLC ตามมาตรฐาน IEC1131-3 [12]

มีด้วยกัน 5 ภาษา คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ภาษา **Ladder Diagram (LD)** ภาษา LD นี้มีพื้นฐานมาจากวงจรควบคุมแบบรีเลย์ ซึ่งมีลักษณะเป็นสัญลักษณ์ของหน้าสัมผัส วิธีการเขียนสามารถทำได้ง่าย โดยบรรทัดนี้จะเรียกว่า Rung หรือ Network วิธีการเขียน โปรแกรมภาษา Ladder Diagram สามารถแสดงดังรูปที่ 3.12



รูปที่ 3.12 แสดงตัวอย่างการเขียน โปรแกรม โดยใช้ภาษา Ladder Diagram

2 ภาษา **Instruction List (IL)** ภาษา IL นี้หรือเรียกอีกชื่อหนึ่งว่าภาษา Statement List (ST) มีพื้นฐานมาจากภาษาแอสเซมบลีที่ใช้ในการเขียน โปรแกรมให้กับไมโครคอนโทรลเลอร์ วิธีการเขียน โปรแกรมภาษา Instruction List ดังตัวอย่างเช่น

```

LD      I0.0
AND    I0.3
OR     I0.1
AND    I0.2
ST     Q0.0
  
```

3. ภาษา **Structured Text (ST)** ภาษา ST นี้ถูกคิดแปลงมาจากภาษาคอมพิวเตอร์ระดับสูง มีลักษณะการเขียนโปรแกรมคล้ายกับภาษาซี (C) หรือปาสคาล (Pascal) ทำให้การเขียนโปรแกรมมีความคล่องตัวและมีประสิทธิภาพสูงขึ้น สามารถทำการคำนวณโปรแกรมที่มีขนาดใหญ่และซับซ้อนมากขึ้น วิธีการเขียน โปรแกรมภาษา Structured Text ดังตัวอย่างเช่น

```

SUM := 0;
FOR I := 1 TO 3 DO
  FOR J := 1 TO 2 DO
    IF FLAG = 1 THEN EXIT;
  END_IF;
  SUM := SUM+J;
  
```

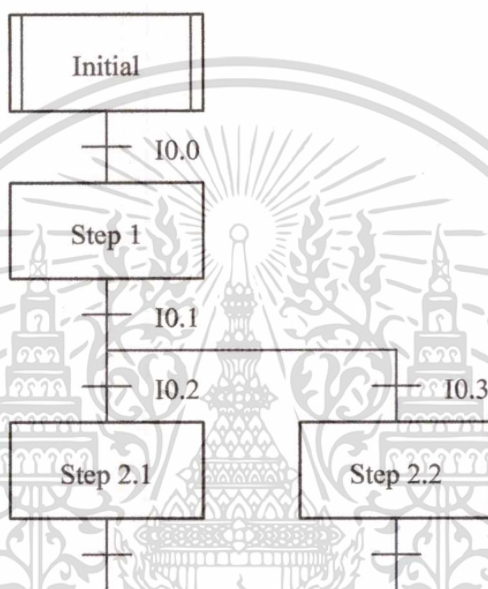
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

END_FOR;
SUM := SUM+I
END_FOR

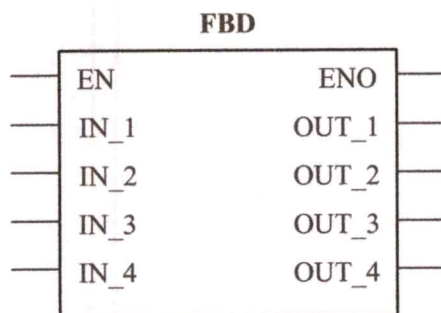
```

4. ภาษา Sequential Function Charts (SFC) ภาษา SFC นี้มีลักษณะเป็นกราฟิกไคอะแกรม คล้ายกับโฟลว์ชาร์ท ซึ่งจะมีการเรียงลำดับขั้นตอนการทำงานเป็นแบบขั้นๆ วิธีการเขียนโปรแกรมภาษา Sequential Function Charts สามารถแสดงดังรูปที่ 3.13



รูปที่ 3.13 แสดงตัวอย่างการเขียนโปรแกรมโดยใช้ภาษา Sequential Function Charts

5. ภาษา Function Block Diagram (FBD) ภาษา FBD นี้มีลักษณะเป็นบล็อกสี่เหลี่ยม ภาษาบล็อกนี้จะใช้กับคำสั่งหรือการควบคุมที่ซับซ้อนเหมาะสำหรับการใช้งานฟังก์ชันคำสั่งนี้บ่อยๆ เช่น ฟังก์ชันทางคณิตศาสตร์ เป็นต้น โดยภายในบล็อกสี่เหลี่ยมมีการเขียนโปรแกรมคำสั่งอยู่ภายใน ซึ่งผู้ใช้งานสามารถสร้างฟังก์ชันบล็อกนี้เองได้ วิธีการเขียนโปรแกรมภาษา Function Block Diagram สามารถแสดงดังรูปที่ 3.14



รูปที่ 3.14 แสดงตัวอย่างบล็อกไดอะแกรมของภาษา Function Block Diagram

3.4.3 เครื่องควบคุม PLC รุ่น SIMATIC S7-300/400

เครื่องควบคุม PLC แต่ละรุ่นจะมีคุณลักษณะทั่วไปและวิธีการการใช้งานของซอฟต์แวร์ที่แตกต่างกันตามการพัฒนาของแต่ละบริษัทผู้ผลิตเครื่องควบคุม PLC ดังนั้นจึงเป็นการดีที่จะมาทำความรู้จักกับเครื่องควบคุม PLC SIMATIC S7-300/400 ของ SIEMENS [13]

ส่วนประกอบของเครื่องควบคุม PLC SIMATIC S7-300/400

เครื่องควบคุม PLC SIMATIC S7-300/400 เป็นเครื่องควบคุม PLC แบบโมดูลาร์ (Modular) ประกอบด้วยส่วนประกอบดังนี้

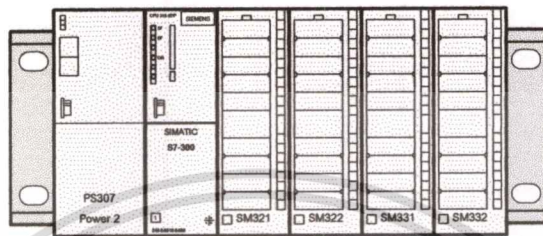
1. แร็ค (Rack) สำหรับเสียบ โมดูลต่างๆ ซึ่งแร็คมีบัสสื่อสารข้อมูล 2 แบบ คือ I/O bus (P bus) ใช้สำหรับสื่อสารข้อมูลของสัญญาณอินพุตเอาต์พุต และ Communication bus (K bus) ใช้สำหรับสื่อสารข้อมูลที่มีขนาดใหญ่
2. แหล่งจ่ายไฟ (Power Supply: PS) ทำหน้าที่จ่ายไฟให้กับแร็คและ โมดูลต่างๆ
3. หน่วยประมวลผลกลาง (Central Processor Unit: CPU) ทำหน้าที่เก็บ โปรแกรมและประมวลผลการทำงานตามโปรแกรมที่กำหนดไว้
4. โมดูลอินเตอร์เฟซ (Interface Modules: IMs) ทำหน้าที่สื่อสารข้อมูลเพื่อขยายหน่วยอินพุตเอาต์พุต คือ ติดต่อข้อมูลระหว่างแร็คกับอีกแร็ค
5. โมดูลสัญญาณ (Signal Modules: SMs) ทำหน้าที่รับสัญญาณอินพุตและส่งสัญญาณจากอุปกรณ์ภายนอก ซึ่งสัญญาณอินพุตเอาต์พุตนี้สามารถเป็นได้ทั้งแบบดิจิทัลและแบบอนาล็อก
6. โมดูลฟังก์ชัน (Function Modules: FMs) ทำหน้าที่ประมวลผลการทำงานเฉพาะส่วนโดยไม่ขึ้นอยู่กับประมวลผลของ CPU ใช้สำหรับงานพิเศษ เช่น Cam controller, Position controller เป็นต้น
7. โมดูลสื่อสาร (Communications Processors: CPs) ทำหน้าที่สื่อสารข้อมูลระหว่างเครื่องควบคุม PLC เครื่องหนึ่งกับเครื่องควบคุม PLC อีกเครื่องหนึ่งหรือระหว่างเครื่องควบคุม PLC กับอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

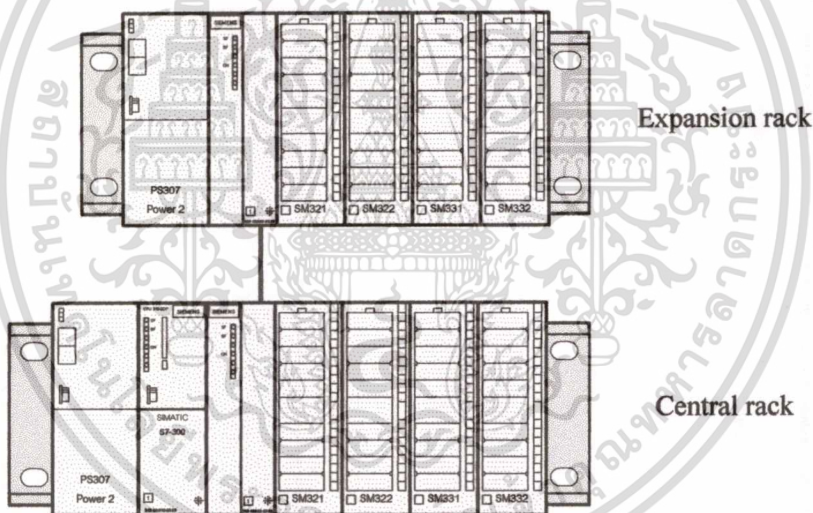
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายนอก รูปแบบการสื่อสารข้อมูล ได้แก่ RS-232, RS-422, RS-485, AS-Interface, Profibus-DP และ Industrial Ethernet เป็นต้น

ส่วนประกอบต่างๆ ของเครื่องควบคุม PLC SIMATIC S7-300 สามารถแสดงได้ดังรูปที่ 3.15



รูปที่ 3.15 แสดงส่วนประกอบต่างๆ ของเครื่องควบคุม PLC SIMATIC S7-300

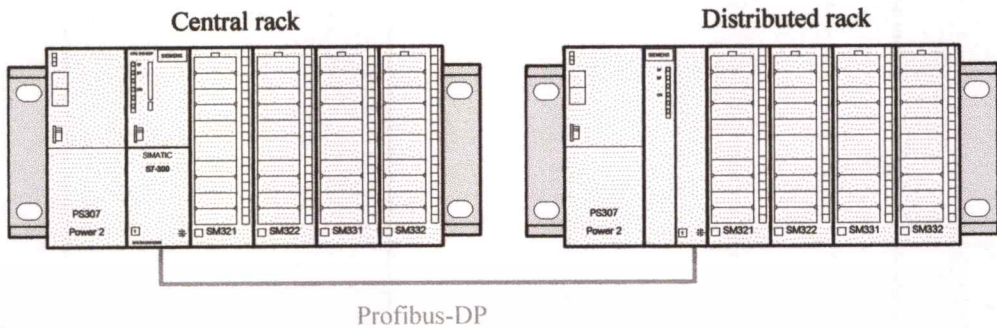


รูปที่ 3.16 แสดงการขยายอินพุทเอาต์พุทโดยใช้โมดูลอินเตอร์เฟซ

การกระจายอินพุทเอาต์พุท (Distributed I/O)

การกระจายอินพุทเอาต์พุท สามารถกระทำได้โดยติดต่อระหว่างหน่วยกระจายอินพุทเอาต์พุท (DP Slave) กับโมดูล DP Master ผ่านการสื่อสารแบบ Profibus-DP โดยหน่วยกระจายอินพุทเอาต์พุทใช้โมดูลอินเตอร์เฟซ (IM) สำหรับกำหนดค่าแอดเดรสและค่าความเร็วในการสื่อสารข้อมูลของหน่วยกระจายอินพุทเอาต์พุทให้กับเน็ตเวิร์ค ซึ่งความเร็วในการติดต่อสื่อสาร (Baud rate) แบบ Profibus-DP อยู่ระหว่าง 9.6 Kb/s ถึง 12 Mb/s ทั้งนี้ขึ้นอยู่กับชนิดของตัวนำและระยะทางของเน็ตเวิร์ค ซึ่งระยะทางของเน็ตเวิร์คอาจมากถึง 9.6 กิโลเมตรสำหรับตัวนำที่เป็นเอกสารเป็นเอกสารทงวนไวสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติหนาไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สายไฟและอาจมากถึง 23.8 กิโลเมตรสำหรับตัวนำเป็นใยแก้วนำแสง (Fiber optic) โดยทุกๆ ระยะทาง 1 กิโลเมตรอาจต้องใช้อุปกรณ์ทวนสัญญาณ (RS-485 Repeater)

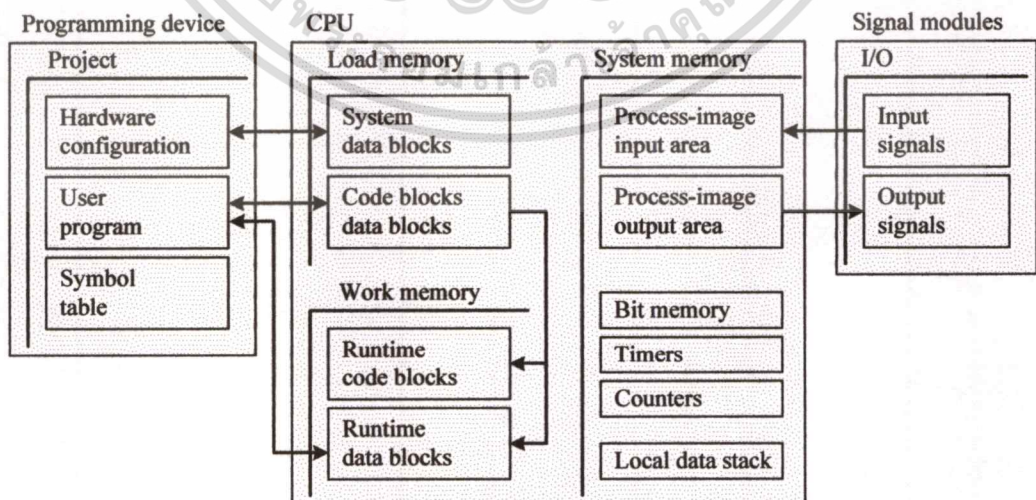


รูปที่ 3.17 แสดงการกระจายอินพุทเอาต์พุตผ่านการสื่อสารแบบ Profibus-DP

พื้นที่ของหน่วยความจำภายในหน่วยประมวลผลกลาง (CPU)

พื้นที่ของหน่วยความจำภายในหน่วยประมวลผลกลาง ประกอบด้วย 2 ส่วน คือ

1. หน่วยความจำของผู้ใช้งาน (User Memory) ประกอบด้วยหน่วยความจำ 2 ส่วนคือ หน่วยความจำโหลด (Load Memory) ทำหน้าที่เก็บโปรแกรมและค่าพารามิเตอร์ของโมดูลต่างๆ และหน่วยความจำทำงาน (Work Memory) ทำหน้าที่เก็บค่าของข้อมูลระหว่างโปรแกรมทำงาน มีลักษณะเป็นหน่วยความจำแบบความเร็วสูง (High-speed RAM)
2. หน่วยความจำของระบบ (System Memory) ทำหน้าที่เก็บตำแหน่งของอินพุท เอาต์พุท หน่วยความจำแบบบิต ไทม์เมอร์ เคนต์เตอร์ และ ข้อมูลแบบสแตคภายใน (Local data stack)



รูปที่ 3.18 แสดงพื้นที่ของหน่วยความจำภายในหน่วยประมวลผลกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิดของข้อมูล (Data type)

ข้อมูลแต่ละข้อมูลถูกเก็บในพื้นที่ของหน่วยความจำของระบบภายในหน่วยประมวลผลกลาง (CPU) การอ้างข้อมูลนี้ต้องกำหนดตำแหน่งของข้อมูล ชนิดของข้อมูล และขนาดให้ถูกต้องเพื่อจองพื้นที่ภายในหน่วยความจำ การกำหนดชนิดของข้อมูลมีด้วยกันหลายชนิด ที่พบบ่อยๆ ได้แก่

1. **ข้อมูลชนิด BOOL** ข้อมูลชนิด BOOL มีขนาด 1 บิต ซึ่งมีค่า 0 หรือ 1 เท่านั้น การแสดงค่าของข้อมูลชนิด BOOL เช่น

เลขฐานสอง : 2# 0 หรือ 2# 1

ตัวอักษร : True หรือ False

2. **ข้อมูลชนิด BYTE** ข้อมูลชนิด BYTE มีขนาด 8 บิต หรือ 1 Byte ซึ่งมีค่าตั้งแต่ 0-255 การแสดงค่าของข้อมูลชนิด BYTE เช่น

เลขฐานสอง : 2# 00 ถึง 2# 1111_1111

เลขฐานสิบหก : 16# 00 ถึง 16# FF

3. **ข้อมูลชนิด WORD** ข้อมูลชนิด WORD มีขนาด 16 บิต หรือ 2 Byte ซึ่งมีค่าตั้งแต่ 0-65,535 การแสดงค่าของข้อมูลชนิด WORD เช่น

เลขฐานสอง : 2# 0000 ถึง 2# 1111_1111_1111_1111

เลขฐานสิบหก : 16# 0000 ถึง 16# FFFF

4. **ข้อมูลชนิด DWORD** ข้อมูลชนิด DWORD มีขนาด 32 บิต หรือ 2 Word ซึ่งการแสดงค่าของข้อมูลชนิด DWORD เช่น

เลขฐานสอง : 2# 0000_0000 ถึง 2# 1111_1111_1111_1111_1111_1111_1111_1111

เลขฐานสิบหก : 16# 0000_0000 ถึง 16# FFFF_FFFF

5. **ข้อมูลชนิด INT** ข้อมูลชนิด INT คือ ค่าจำนวนเต็ม มีขนาด 16 Bit แบบมีเครื่องหมาย ซึ่งมีค่าอยู่ในช่วงระหว่างตั้งแต่ -32,768 ถึง 32,768

6. **ข้อมูลชนิด DINT** ข้อมูลชนิด DINT คือ ค่าจำนวนเต็ม มีขนาด 32 Bit แบบมีเครื่องหมาย ซึ่งมีค่าอยู่ในช่วงระหว่างตั้งแต่ -2,147,483,648 ถึง 2,147,483,648

7. **ข้อมูลชนิด REAL** ข้อมูลชนิด REAL คือ ค่าจำนวนจริง มีขนาด 32 Bit ตัวอย่างเช่น -1.553 หรือ $4.7 e^{-25}$ เป็นต้น

8. **ข้อมูลชนิด CHAR** ข้อมูลชนิด CHAR คือ ข้อมูลแบบตัวอักษรที่มีลักษณะเป็นแบบ ASCII มีขนาด 8 Bit เช่น "A" หรือ "C" เป็นต้น

การอ้างตำแหน่งของข้อมูล

การเขียนโปรแกรมให้กับเครื่องควบคุม PLC จำเป็นต้องทราบถึงวิธีการอ้างตำแหน่งของข้อมูล ซึ่งการอ้างตำแหน่งของข้อมูลสามารถกระทำได้ 4 แบบ ได้แก่

1. การอ้างตำแหน่งแบบบิต คือ การอ้างตำแหน่งเป็นแบบเจาะจงบิตของไบต์ ตัวเลขของบิตเริ่มตั้งแต่ บิตที่ 0 ถึง บิตที่ 7 เฉพาะการอ้างตำแหน่งแบบบิตในคาต้าบล็อกรจึงใช้สัญลักษณ์แทนด้วย "X" เท่านั้น ดังตัวอย่างเช่น

I 1.0 : อินพุทไบต์ที่ 1 บิตที่ 0
 Q 16.4 : เอาท์พุทไบต์ที่ 16 บิตที่ 4
 M 56.7 : หน่วยความจำแบบบิต ไบต์ที่ 56 บิตที่ 7
 DB10.DBX 2.0 : ข้อมูลบิตที่ 2.0 ในคาต้าบล็อกร DB 10

2. การอ้างตำแหน่งแบบไบต์ คือ การอ้างตำแหน่งเป็นแบบไบต์ โดยการอ้างตำแหน่งแบบไบต์นี้ใช้สัญลักษณ์แทนด้วย "B" ดังตัวอย่างเช่น

IB 2 : อินพุทไบต์ที่ 2
 QB 18 : เอาท์พุทไบต์ที่ 18
 MB 58 : หน่วยความจำแบบบิต ไบต์ที่ 58
 DB11.DBB 14 : ข้อมูลไบต์ที่ 14 ในคาต้าบล็อกร DB 11

3. การอ้างตำแหน่งแบบเวิร์ด คือ การอ้างตำแหน่งเป็นแบบเวิร์ด โดยการอ้างตำแหน่งแบบเวิร์ดนี้ใช้สัญลักษณ์แทนด้วย "W" ดังตัวอย่างเช่น

IW 4 : อินพุทเวิร์ดที่ 2 (ไบต์ที่ 4 และ 5)
 QW 20 : เอาท์พุทเวิร์ดที่ 20 (ไบต์ที่ 20 และ 21)
 MW 60 : หน่วยความจำแบบบิต เวิร์ดที่ 60 (ไบต์ที่ 60 และ 61)
 DB20.DBW 22 : ข้อมูลเวิร์ดที่ 22 ในคาต้าบล็อกร DB 20

4. การอ้างตำแหน่งแบบดับเบิลเวิร์ด คือ การอ้างตำแหน่งเป็นแบบดับเบิลเวิร์ด โดยการอ้างตำแหน่งแบบดับเบิลเวิร์ดนี้ใช้สัญลักษณ์แทนด้วย "D" ดังตัวอย่างเช่น

ID 8 : อินพุทดับเบิลเวิร์ดที่ 8 (ไบต์ที่ 8 9 10 และ 11)
 QD 24 : เอาท์พุทดับเบิลเวิร์ดที่ 24 (ไบต์ที่ 24 25 26 และ 27)
 MD 62 : หน่วยความจำแบบบิต ดับเบิลเวิร์ดที่ 62 (ไบต์ที่ 62 63 64 และ 65)
 DB22.DBD 10 : ข้อมูลดับเบิลเวิร์ดที่ 10 ในคาต้าบล็อกร DB 22

การเข้าถึงของข้อมูล

การเข้าถึงของข้อมูลมีการเข้าถึงด้วยกัน 2 แบบ คือ การเข้าถึงของข้อมูลโดยตรง (Direct addressing) และการเข้าถึงของข้อมูลโดยอ้อม (Indirect addressing)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้กับโรงเรียนเพื่อให้นักศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การเข้าถึงของข้อมูลโดยตรง (Direct addressing) คือการเข้าถึงของข้อมูล โดยการกำหนดค่าของแอดเดรสโดยตรง ตัวอย่างเช่น

L IW 8 : Load อินพุทเวอร์คที่ 8

L MD 0 : Load หน่วยความจำดับเบิลเวิร์คที่ 0

2. การเข้าถึงของข้อมูลโดยอ้อม (Indirect addressing) คือการเข้าถึงของข้อมูล โดยการกำหนดค่าของแอดเดรสผ่านแอดเดรสของหน่วยความจำหรือรีจิสเตอร์ เพื่อเป็นPointer ในการชี้ตำแหน่ง ซึ่งการเข้าถึงของข้อมูลโดยอ้อมนี้สามารถใช้กับ โปรแกรมแบบวนรอบได้ (Loop) ดังตัวอย่างเช่น

L I[MW 8] : Load อินพุทตำแหน่งที่มีค่าอยู่ในหน่วยความจำเวิร์คที่ 8

L DBD[AR1,P#0.0] : Load หน่วยความจำดับเบิลเวิร์คในดาต้าบล็อกค่าที่ตำแหน่งของ AR1 บวกด้วยค่า Pointer 0.0

ชนิดของบล็อกโปรแกรม (Type of Program Blocks)

เครื่องควบคุม PLC มีบล็อกสำหรับการ โปรแกรมหลายชนิด ซึ่งบล็อกสำหรับ โปรแกรมนี้เรียกสั้นๆ ว่าบล็อก (Blocks) การเลือกชนิดบล็อกขึ้นอยู่กับผู้ใช้งานตามกระบวนการ โดยบล็อกมีด้วยกันหลายชนิด ได้แก่

1. **Organization blocks (OBs)** คือ บล็อกที่ทำหน้าที่ติดต่อระหว่างการทำงานของระบบเครื่องควบคุม PLC กับ โปรแกรมของผู้ใช้งาน เช่น OB1 ซึ่งเป็นบล็อกที่ทำงานเป็นวงรอบทุกๆ การทำงานของระบบ

2. **Functions (FCs)** คือ บล็อกที่ใช้สำหรับเขียนฟังก์ชันการทำงาน ซึ่งฟังก์ชันการทำงานนี้ขึ้นอยู่กับผู้ใช้งาน โดยผู้ใช้งานต้องกำหนดและเขียน โปรแกรมให้กับบล็อกนั้น เช่น การคำนวณต่างๆ

3. **System functions (SFCs)** คือ บล็อกที่มีฟังก์ชันการทำงานอยู่แล้วภายในเครื่องควบคุม PLC ขึ้นอยู่กับเบอร์ของหน่วยประมวลผลกลาง ซึ่งผู้ใช้งานสามารถเรียกบล็อกนี้มาใช้งานได้แต่ไม่สามารถแก้ไขหรือเพิ่มเติมฟังก์ชันการทำงานได้

4. **Function Blocks (FBs)** คือ บล็อกที่ใช้สำหรับเขียนฟังก์ชันการทำงาน ซึ่งฟังก์ชันการทำงานนี้ขึ้นอยู่กับผู้ใช้งาน มีลักษณะคล้ายคลึงกับ FCs แต่การเรียกใช้งาน FBs นี้ต้องอาศัย DBs ในการเก็บค่าตัวแปรขณะทำงานด้วย

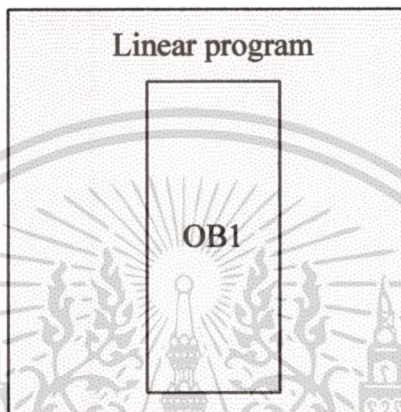
5. **System Function Blocks (SFBs)** คือ บล็อกที่มีฟังก์ชันการทำงานอยู่แล้วภายในเครื่องควบคุม PLC ขึ้นอยู่กับเบอร์ของหน่วยประมวลผลกลาง ซึ่งผู้ใช้งานสามารถเรียกบล็อกนี้มาใช้งานได้ แต่ไม่สามารถแก้ไขหรือเพิ่มเติมฟังก์ชันการทำงานได้ มีลักษณะคล้ายคลึงกับ SFCs แต่การเรียกใช้งาน SFBs นี้ต้องอาศัย DBs ในการเก็บค่าตัวแปรขณะทำงานด้วย

6. **Data Blocks (DBs)** คือ บล็อกที่ใช้ในการเก็บข้อมูลต่างๆ ของ โปรแกรม โดยการใช้งานผู้ใช้งานต้องกำหนดชนิดของข้อมูลต่างๆ ให้กับบล็อกนี้

โครงสร้างของโปรแกรม (Program structure)

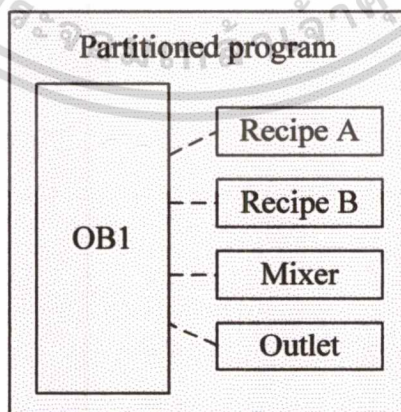
การเขียนโปรแกรมให้กับเครื่องควบคุม PLC ผู้ใช้งานสามารถเขียนโปรแกรมได้ตามความต้องการ ซึ่งโปรแกรมนี้อาจเขียนได้ 3 แบบ คือ

1. โปรแกรมแบบลิเนียร์ (Linear program) มีลักษณะการเรียกใช้งาน OB1 เพียงบล็อกเดียว โดยผู้ใช้งานเขียนโปรแกรมการทำงานอยู่ภายใน OB1 ทั้งหมด



รูปที่ 3.19 แสดงโครงสร้างโปรแกรมแบบลิเนียร์

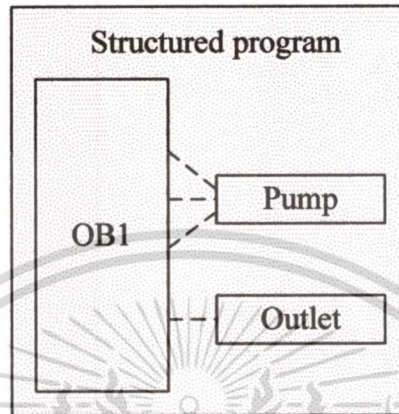
2. โปรแกรมแบบแบ่งส่วน (Partitioned program) มีลักษณะคล้ายกับโปรแกรมแบบลิเนียร์ คือการทำงานของโปรแกรมยังคงเป็นแบบลิเนียร์โดยผู้ใช้งานแบ่งโปรแกรมการทำงานออกเป็นส่วนๆ และผู้ใช้งานเขียนโปรแกรมการทำงานแต่ละส่วนในแต่ละบล็อก



รูปที่ 3.20 แสดงโครงสร้างโปรแกรมแบบแบ่งส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. โปรแกรมแบบโครงสร้าง (Structured program) มีลักษณะเป็นแบบโครงสร้าง คือ การทำงานของโปรแกรมจะตามโครงสร้าง เมื่อถึงเงื่อนไขการทำงานที่เป็นฟังก์ชันก็จะเรียกบล็อกที่เขียนฟังก์ชันมาใช้ เมื่อทำงานเสร็จฟังก์ชันนั้นแล้วก็จะกลับมาทำงานยังโปรแกรมหลักต่อไป ซึ่งการเรียกใช้งานบล็อกฟังก์ชันนี้มีลักษณะคล้ายกับการอินเทอร์รัพท์ (Interrupt)



รูปที่ 3.21 แสดง โครงสร้าง โปรแกรมแบบ โครงสร้าง

การเขียนโปรแกรมให้กับเครื่องควบคุม PLC

การเขียนโปรแกรมให้กับเครื่องควบคุม PLC ประกอบด้วย 2 ส่วน คือ ส่วนกำหนดค่าทางฮาร์ดแวร์ (Hardware configuration) และส่วนการเขียนโปรแกรมการทำงาน (Blocks)

1. ส่วนกำหนดค่าของฮาร์ดแวร์ (Hardware configuration: HW Config.) ส่วนกำหนดค่าของฮาร์ดแวร์ คือ การกำหนดค่าต่างๆ ของฮาร์ดแวร์ในซอฟต์แวร์ให้ตรงกับฮาร์ดแวร์ภายนอก เพื่อกำหนดชนิด ประเภท การติดต่อสื่อสารและแอดเดรสของโมดูลต่างๆ ที่ให้ตรงกับความต้องการ ซึ่งการใช้งานจำเป็นต้องดาวน์โหลด HW Config. เพื่อตรวจสอบความถูกต้องว่าระหว่างฮาร์ดแวร์กับซอฟต์แวร์ว่าตรงกันและเป็นการตรวจสอบเช็คอุปกรณ์ทางฮาร์ดแวร์ต่างๆ ว่าพร้อมทำงานหรือไม่ ถ้าหากเกิดการขัดข้องทางการติดต่อสื่อสารหรืออุปกรณ์ฮาร์ดแวร์เสียจะมีสัญญาณไฟ Error กระพริบเตือนที่หน่วยประมวลผลกลาง (CPU) การออกแบบทางด้านซอฟต์แวร์ของเครื่องควบคุม PLC สามารถแสดงดังรูปที่ 3.22

HW Config - [SIMATIC 400(1) (Configuration) - test]

Station Edit Insert PLC View Options Window Help

0) UR2

PROFIBUS(1): DP master system (1)

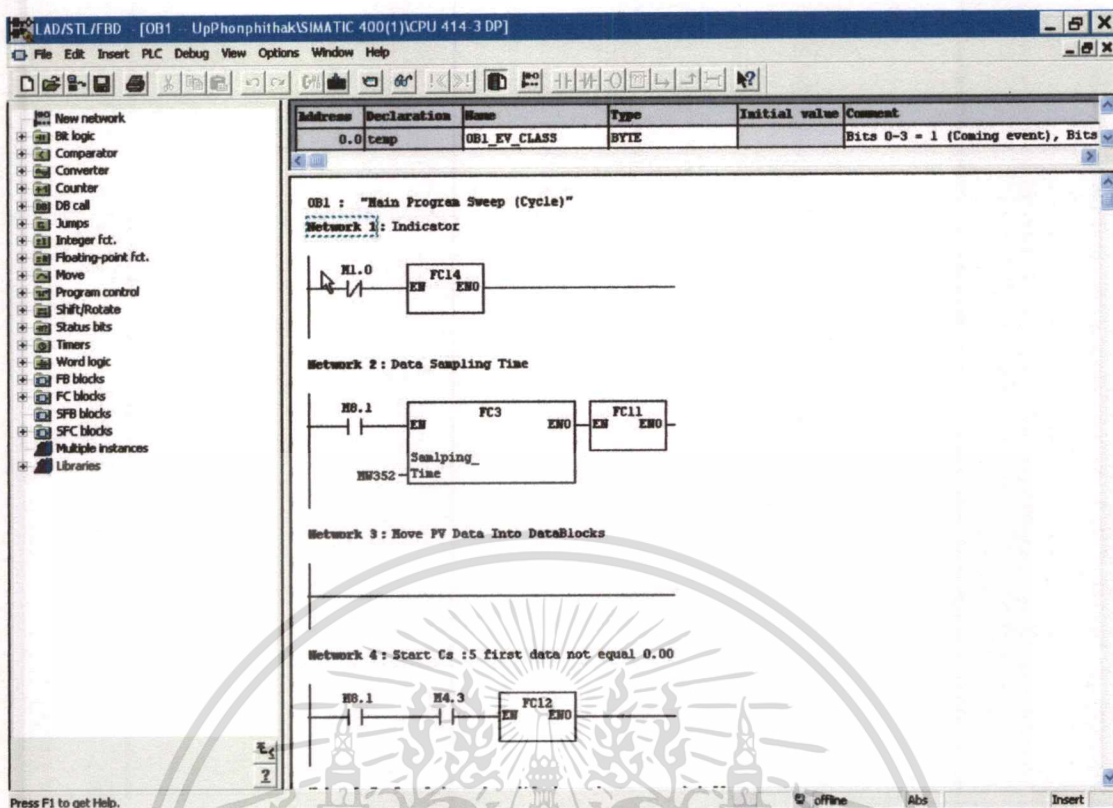
Slot	Module	Order number	Firmware	MPI address	I address	Q address	Comment
1	PS 407 10A	6ES7 407-0KA00-0AA0					
3	CPU 414-3 DP	6ES7 414-3EJ00-0AB0		2			
X2	DP			2	512I*		
X1	MPI/DP			2	512Q*		
IF1							
5							
6							
7							
8							
9							

Press F1 to get Help.

รูปที่ 3.22 แสดงการกำหนดค่าของฮาร์ดแวร์ของเครื่องควบคุม PLC

2. ส่วนการเขียนโปรแกรมการทำงาน (Blocks) ส่วนการเขียนโปรแกรมการทำงานของเครื่องควบคุม PLC คือส่วนที่ใช้สำหรับโปรแกรมเพื่อควบคุมการทำงานตามขั้นตอนที่กำหนดไว้ตามหลักการที่ได้กล่าวมาข้างต้น แสดงดังรูปที่ 3.23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.23 แสดงการเขียนโปรแกรมการทำงานของเครื่องควบคุม PLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การออกแบบและการพัฒนาตัวควบคุมพีไอดี แบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC

4.1 กล่าวนำ

การออกแบบและการพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC นี้ใช้เครื่องควบคุม PLC ที่ใช้งานฟังก์ชันควบคุมพีไอดีในการควบคุมกระบวนการแบบป้อนกลับทำหน้าที่วิเคราะห์และปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดีแบบอัตโนมัติ โดยการเขียนโปรแกรมให้กับเครื่องควบคุม PLC เพื่อกำหนดให้วิเคราะห์และปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดีตามหลักการที่ได้กล่าวมาแล้วข้างต้น โดยโครงการในวิทยานิพนธ์ฉบับนี้ใช้เครื่องควบคุม PLC ของบริษัท ซิเมนส์ จำกัด (SIEMENS) รุ่น SIMATIC S7-300/400 และกระบวนการจำลองภายในภาควิชาวิศวกรรมการวัดคุม

4.2 การประยุกต์ใช้เครื่องควบคุม PLC ตามหลักการและทฤษฎี

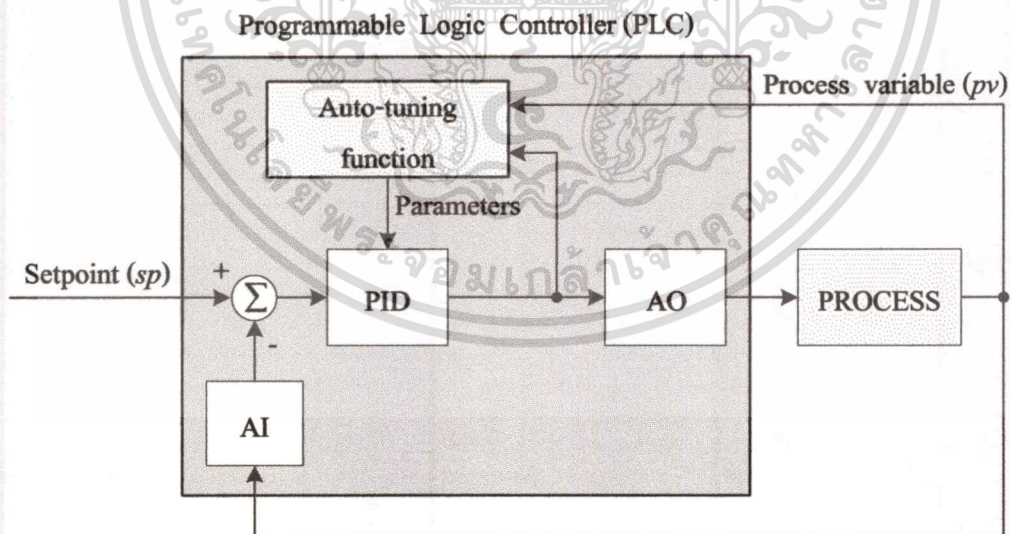
เครื่องควบคุม PLC จะรับสัญญาณผลตอบสนองของกระบวนการที่ถูกวัดโดยอุปกรณ์วัดและส่งสัญญาณ (Transmitter) ในรูปของสัญญาณมาตรฐาน (1-5VDC หรือ 4-20 mA) เข้าทางโมดูลอินพุทแบบอนาล็อก (Analog input module: AI) เพื่อเป็นข้อมูลที่ใช้สำหรับวิเคราะห์ต่อไป โดยมีฟังก์ชัน Auto-tuning ทำหน้าที่วิเคราะห์และหาค่าพารามิเตอร์ของตัวควบคุมพีไอดี ตามขั้นตอนที่ได้โปรแกรมไว้ โดยเครื่องควบคุม PLC จะส่งสัญญาณควบคุมออกไปควบคุมกระบวนการทางโมดูลเอาต์พุทแบบอนาล็อก (Analog output module: AO) ในรูปของสัญญาณมาตรฐาน (4-20 mA) โดยค่าพารามิเตอร์ต่างๆ จะแสดงผลผ่านทางหน้าจอคอมพิวเตอร์ ซึ่งฟังก์ชัน Auto-tuning นี้ถูกเขียนอยู่ในรูปของฟังก์ชันบล็อกไดอะแกรม (Function block diagram) ซึ่งขั้นตอนการทำงานโดยละเอียดมีดังนี้

การทดสอบกระบวนการเพื่อหาค่าคุณลักษณะของกระบวนการทำได้โดยการป้อนสัญญาณขั้นบันไดขนาด Δm ให้กับกระบวนการแบบรูปเปิด โดยผู้ใช้งานสามารถกำหนดขนาดของ Δm ได้ตามความต้องการ หลังจากป้อนสัญญาณทดสอบเข้าสู่กระบวนการแล้วผลตอบสนองของกระบวนการจะถูกวัดและส่งมายังเครื่องควบคุม PLC โดยเครื่องควบคุม PLC จะทำการสุ่มเก็บค่าผลตอบสนองของกระบวนการไว้ในหน่วยความจำเพื่อเป็นข้อมูลสำหรับการคำนวณหาค่าพารามิเตอร์ของกระบวนการต่อไป ซึ่งกำหนดช่วงเวลาของกระบวนการ (t_0) หาได้จากระยะเวลาตั้งแต่เริ่มทดสอบกระบวนการจนกระทั่งผลตอบสนองของกระบวนการเปลี่ยนแปลงไปประมาณ

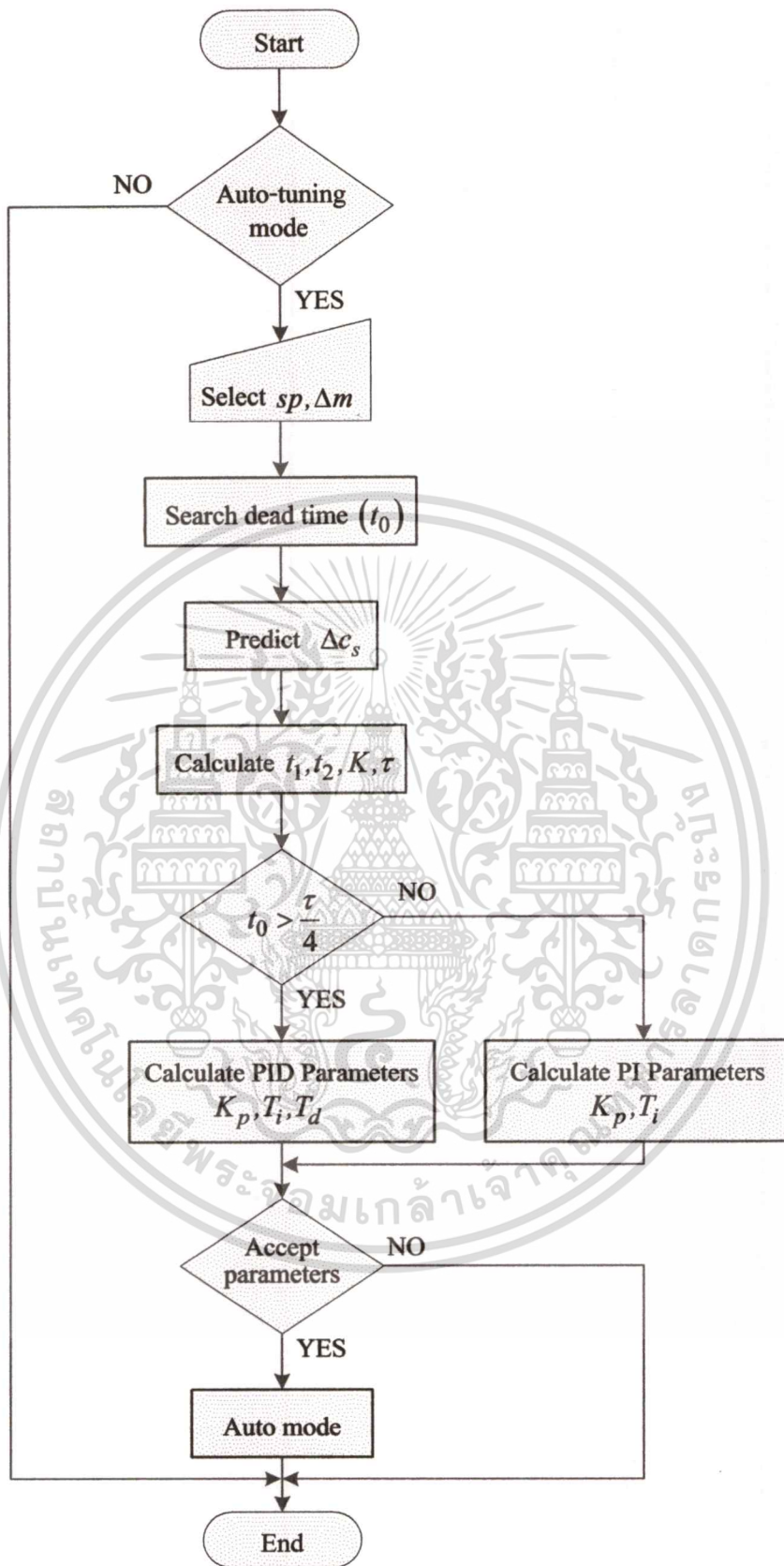
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3% ของค่าเริ่มต้น เมื่อกระบวนการเข้าสู่สภาวะคงที่ (Steady state) แล้วจะได้ค่าผลตอบสนองที่ สภาวะคงที่ (Δc_s) นำค่า Δc_s ที่ได้มาคำนวณหาค่าอัตราขยายของกระบวนการ (K) และค่า เวลาคงที่ของกระบวนการ (τ) จากการทดสอบพบว่า การหาค่าคุณลักษณะของกระบวนการ จำเป็นต้องรอให้กระบวนการเข้าสู่สภาวะคงที่เพื่อนำค่า Δc_s ที่ได้มาทำการคำนวณหา ค่าพารามิเตอร์ต่างๆ ของกระบวนการ ซึ่งในกระบวนการบางกระบวนการอาจต้องใช้ระยะเวลา นานมากเพื่อให้ผลตอบสนองของกระบวนการเข้าสู่สภาวะคงที่ ในวิทยานิพนธ์ฉบับนี้จึงได้นำ ระเบียบวิธีการเชิงตัวเลขมาใช้ในการประมาณค่าของ Δc_s โดยทำการวิเคราะห์ข้อมูลที่ได้จากการ สุ่มขณะทำการทดสอบกระบวนการตามวิธีผลต่างจากการแบ่งย่อยของนิวตัน เพื่อลดระยะเวลาใน การทดสอบกระบวนการให้สั้นลง เมื่อได้ค่าพารามิเตอร์ของกระบวนการแล้วเครื่องควบคุม PLC จะทำการตั้งเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอหรือพีไอดี (K_p, T_i และ T_d) ตาม เงื่อนไขที่ได้กำหนดไว้ ซึ่งค่าพารามิเตอร์ต่างๆ ที่คำนวณได้นี้จะแสดงผลทางหน้าจอคอมพิวเตอร์ เพื่อให้ผู้ใช้งานตัดสินใจว่าต้องการจะปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดีหรือไม่ หากตกลงที่จะ ใช้ค่าพารามิเตอร์ที่ได้นี้ กระบวนการจะถูกควบคุมด้วยเครื่องควบคุม PLC ที่ใช้ตัวควบคุมพีไอ หรือพีไอดีโดยใช้ค่าพารามิเตอร์ดังกล่าวทันที

ขั้นตอนการทำงานของตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC แสดงดังรูปที่ 4.1 และรูปที่ 4.2



รูปที่ 4.1 บล็อกไดแกรมของระบบ Auto-tuning



รูปที่ 4.2 แสดงผังขั้นตอนการทำงานของตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบน
เครื่องควบคุม PLC

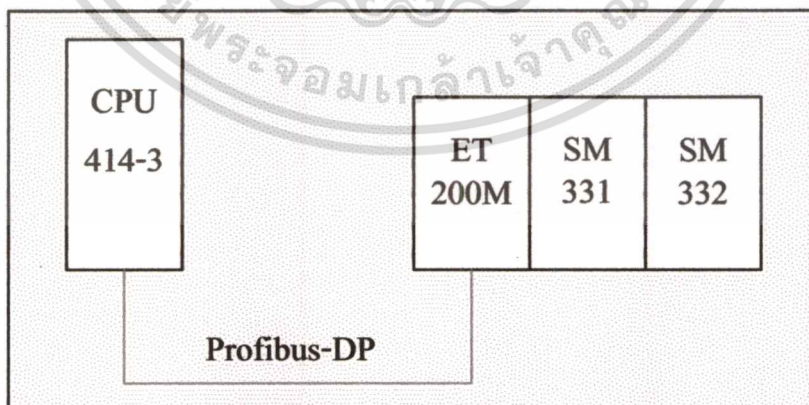
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การออกแบบทางด้านฮาร์ดแวร์

โครงสร้างทางด้านฮาร์ดแวร์ของการพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC สามารถแบ่งออกได้เป็น 2 ส่วนใหญ่ๆ คือส่วนของเครื่องควบคุม PLC และส่วนการแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์

4.3.1 ส่วนของเครื่องควบคุม PLC

ส่วนของเครื่องควบคุม PLC ทำหน้าที่รับข้อมูลสัญญาณจากตัวแปรกระบวนการ ที่เป็นสัญญาณอนาลอกเข้าทางโมดูลอินพุทแบบอนาลอก หลังจากนั้นเครื่องควบคุม PLC จะทำการคำนวณหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีตามขั้นตอนที่ได้กล่าวมาแล้ว หลังจากนั้นที่เครื่องควบคุม PLC คำนวณหาค่าพารามิเตอร์แล้วเสร็จก็จะเข้าสู่การควบคุมโดยส่งสัญญาณควบคุมแบบอนาลอกออกทางโมดูลเอาต์พุทแบบอนาลอกไปควบคุมกระบวนการต่อไป ซึ่งโครงการในวิทยานิพนธ์ฉบับนี้จะใช้เครื่องควบคุม PLC ของบริษัท ซิเมนส์ จำกัด (SIEMENS) รุ่น SIMATIC S7-300/400 โดยใช้หน่วยประมวลผลกลางคือ CPU414-3 (6ES7414-3XJ00-0AB0) โมดูลอินพุทแบบอนาลอก คือ SM331 (6ES7331-7KF02-0AB0) สามารถรับสัญญาณอินพุทแบบอนาลอกได้ 8 ช่องสัญญาณ มีความละเอียด 12 Bit และโมดูลเอาต์พุทแบบอนาลอก คือ SM332 (6ES7332-5HD01-0AB0) สามารถส่งสัญญาณเอาต์พุทแบบอนาลอกได้ 4 ช่องสัญญาณ มีความละเอียด 12 Bit การติดต่อระหว่างหน่วยประมวลผลกลางกับ โมดูลอินพุทและเอาต์พุทเป็นแบบกระจายหน่วยอินพุทเอาต์พุท (Distributed I/O) โดยใช้โมดูลกระจายหน่วยอินพุทเอาต์พุท คือ ET200M (6ES7153-1AA03-0XB0) โดยการติดต่อสื่อสารเป็นแบบ Profibus-DP การออกแบบทางด้านฮาร์ดแวร์ส่วนของเครื่องควบคุม PLC สามารถแสดงได้ดังรูปที่ 4.3

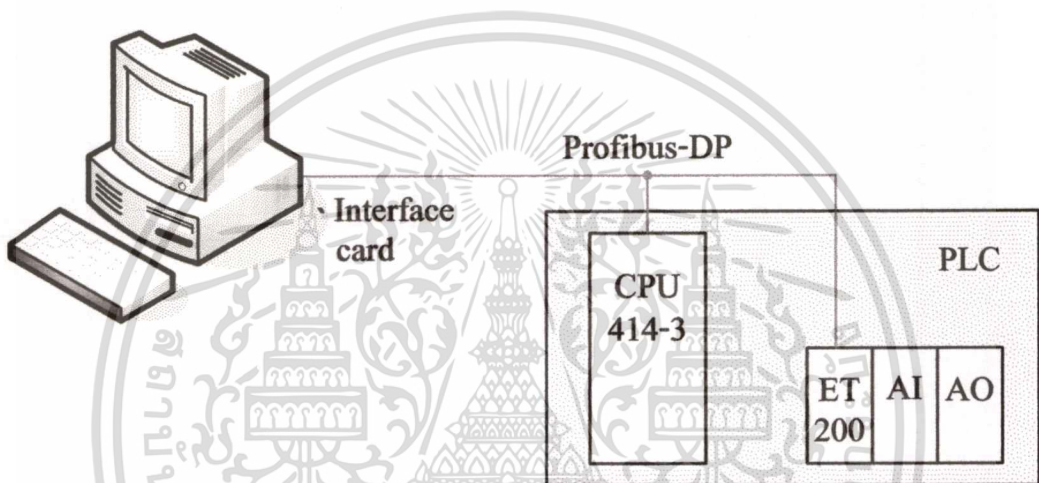


รูปที่ 4.3 แสดงการออกแบบทางด้านฮาร์ดแวร์ส่วนของเครื่องควบคุม PLC

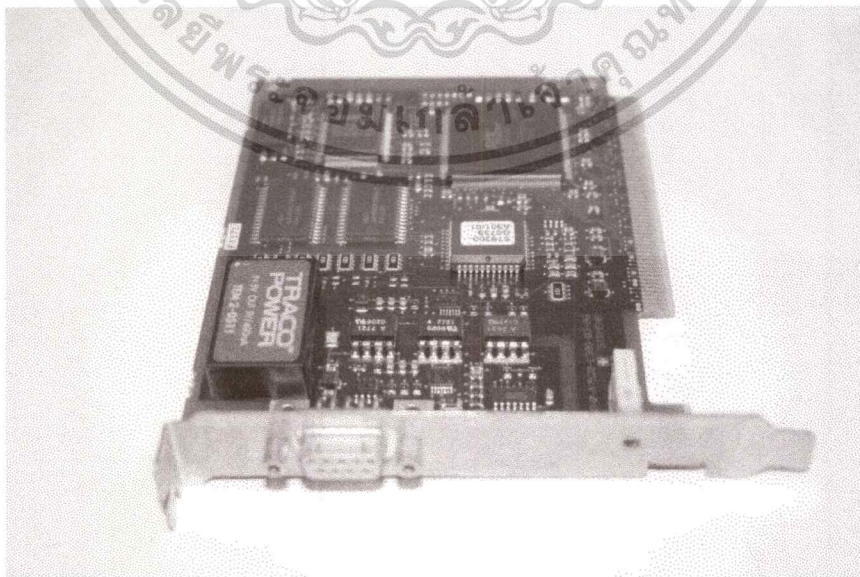
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 ส่วนแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์

ส่วนแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์ทำหน้าที่กำหนดค่าข้อมูลต่างๆ ที่ใช้ในการควบคุมการทำงานและแสดงผลการทำงาน ซึ่งโครงการในวิทยานิพนธ์นี้ใช้คอมพิวเตอร์ส่วนบุคคล (PC) ในการทำหน้าที่กำหนดค่าข้อมูลและแสดงผลโดยการติดต่อสื่อสารระหว่างคอมพิวเตอร์ส่วนบุคคลกับเครื่องควบคุม PLC เป็นแบบ Profibus-DP โดยคอมพิวเตอร์ส่วนบุคคลมีการ์ดอินเตอร์เฟซทำหน้าที่แปลงสัญญาณจาก Profibus-DP เป็น RS-232 ซึ่งการ์ดอินเตอร์เฟซที่ใช้ก็คือ CP5611 (6GK15611-1AA0) ของบริษัท ซีเมนส์ จำกัด (SIEMENS) การออกแบบทางด้านฮาร์ดแวร์ของส่วนแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์ สามารถแสดงได้ดังรูปที่ 4.4



รูปที่ 4.4 แสดงการออกแบบทางด้านฮาร์ดแวร์ของส่วนแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์



รูปที่ 4.5 แสดงการ์ดอินเตอร์เฟซ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเฉพาะที่ขอให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การออกแบบทางด้านซอฟต์แวร์

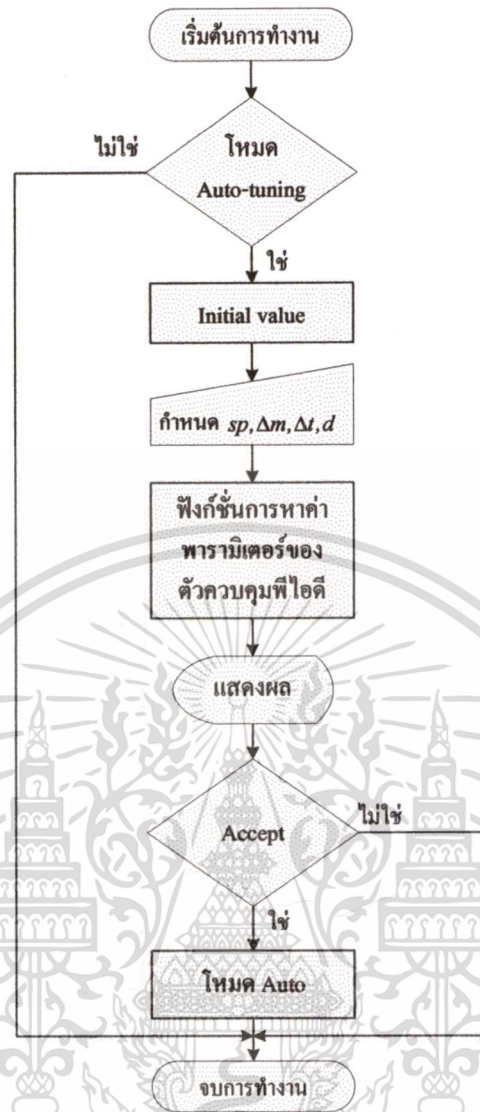
การออกแบบทางด้านซอฟต์แวร์ของการพัฒนาตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC สามารถแบ่งออกได้เป็น 2 ส่วนใหญ่ๆ คือการเขียนโปรแกรมการทำงานของเครื่องควบคุม PLC และการเขียนโปรแกรมแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์

4.4.1 การเขียนโปรแกรมการทำงานของเครื่องควบคุม PLC

ส่วนการเขียนโปรแกรมการทำงานให้กับเครื่องควบคุม PLC นี้ใช้ซอฟต์แวร์คือโปรแกรม SIMATIC STEP 7 Manager ภาษาที่ใช้ในการเขียนโปรแกรมการทำงานเขียนโดยใช้ภาษา Ladder Diagram ภาษา Structure Text และภาษา Function Block Diagram และมีลักษณะโปรแกรมเป็นแบบโครงสร้างโดยการเขียนโปรแกรมให้อยู่ในรูปแบบบล็อกฟังก์ชันย่อยๆ เพื่อความสะดวกในการเรียกใช้งาน ดังนี้

บล็อกฟังก์ชัน Auto-tuning

บล็อกฟังก์ชัน Auto-tuning ทำหน้าที่เป็นบล็อกฟังก์ชันหลักของการทำงานโดยผู้ใช้งานเลือกโหมดการทำงานเป็น Auto-tuning แล้วทำการกำหนดค่าต่างๆ ที่เกี่ยวข้อง หลังจากนั้นจะทำการหาค่าพารามิเตอร์ของตัวควบคุม โดยจะเรียกบล็อกฟังก์ชันการหาค่าพารามิเตอร์ของตัวควบคุมพีไอดี หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมพีไอดีแล้วจะแสดงผลทางหน้าจอคอมพิวเตอร์ หากผู้ใช้งานตกลงใช้ค่าพารามิเตอร์ของตัวควบคุมพีไอดีที่ได้กระบวนการจะถูกควบคุมในโหมด Auto-tuning โดยใช้ค่าพารามิเตอร์ของตัวควบคุมพีไอดีที่คำนวณได้ การทำงานของบล็อกฟังก์ชัน Auto-tuning แสดงดังรูปที่ 4.6

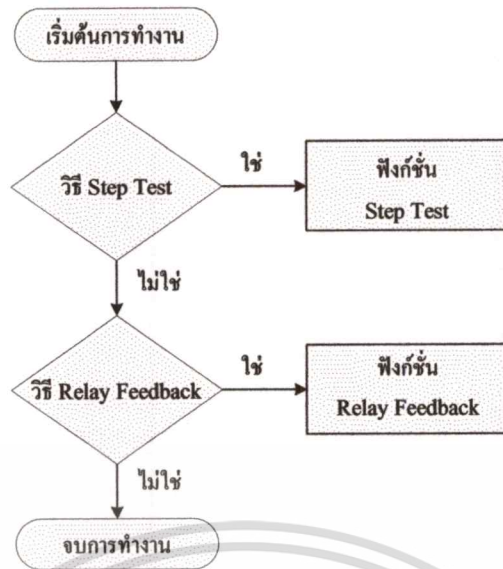


รูปที่ 4.6 แสดงการทำงานของบล็อกฟังก์ชัน Auto-tuning

บล็อกฟังก์ชันการหาค่าพารามิเตอร์ของตัวควบคุมพีไอดี

บล็อกฟังก์ชันการหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีทำหน้าที่หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีซึ่งวิธีการหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีมีด้วยกัน 2 วิธีคือ วิธี Step Test และวิธี Relay Feedback โดยถ้าเลือกวิธี Step Test จะเรียกบล็อกฟังก์ชัน Step Test หรือถ้าหากเลือกวิธี Relay Feedback จะเรียกบล็อกฟังก์ชัน Relay Feedback ซึ่งการทำงานของบล็อกฟังก์ชันการหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีแสดงดังรูปที่ 4.7

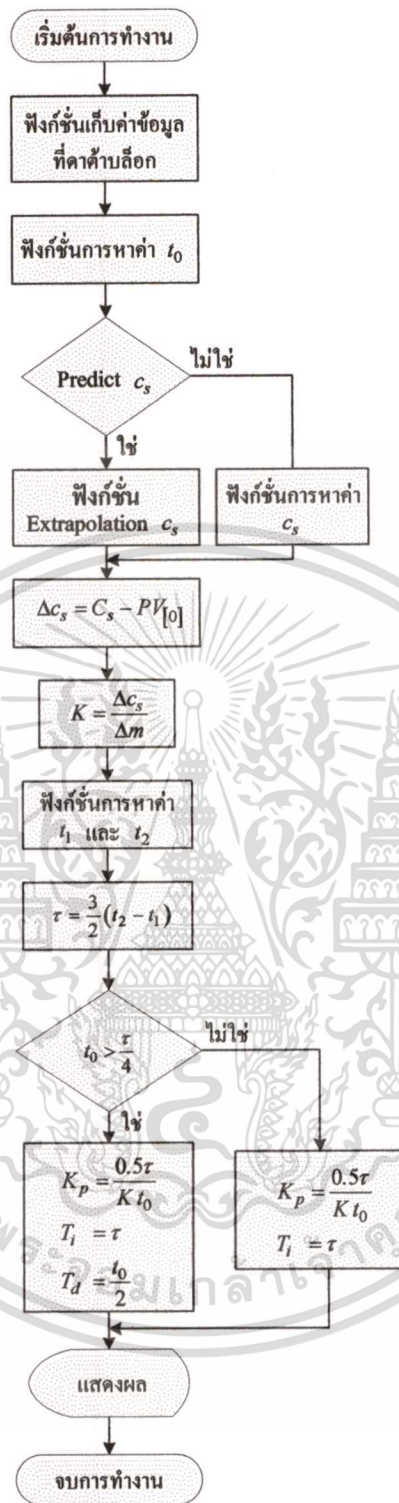
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงการทำงานของบล็อกฟังกั้นการหาค่าพารามิเตอร์ของตัวควบคุมพีไอดี

บล็อกฟังกั้น Step test

บล็อกฟังกั้น Step test ทำหน้าที่หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีโดยการป้อนสัญญาณควบคุมแบบขั้นบันไดขนาด Δm ให้กับกระบวนการแบบวงเปิดเพื่อหาค่าพารามิเตอร์ของกระบวนการสำหรับเป็นตัวแปรที่จะนำไปคำนวณหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีต่อไป โดยค่าผลตอบสนองของกระบวนการจะถูกเก็บไว้ที่ค่าบล็อก โดยเรียกฟังกั้นการเก็บข้อมูลที่ค่าค่าบล็อก หลังจากเริ่มทำการทดสอบกระบวนการจะได้ค่าช่วงเวลาของกระบวนการ (t_0) โดยเรียกบล็อกฟังกั้นการหาค่า t_0 ต่อไปเป็นการหาค่าผลตอบสนองของกระบวนการที่สถานะคงที่ (c_s) โดยถ้าหากเลือกการประมาณค่าล่วงหน้า (Predict c_s) จะเรียกบล็อกฟังกั้นการประมาณค่าล่วงหน้า (Extrapolation c_s) หรือถ้าเลือกการรอให้ผลตอบสนองของกระบวนการเข้าสู่สถานะคงที่ จะเรียกบล็อกฟังกั้นการหาค่า c_s หลังจากได้ค่าผลตอบสนองของกระบวนการที่สถานะคงที่แล้ว นำค่า c_s ไปคำนวณหาค่า Δc_s และค่าอัตราขยายของกระบวนการ (K) หลังจากได้ค่า Δc_s นำไปหาค่าเวลา t_1 และ t_2 ตามสมการ โดยเรียกบล็อกฟังกั้นการหาค่าเวลา t_1 และ t_2 ตามลำดับ หลังจากที่ได้ค่าของ t_1 และ t_2 นำไปคำนวณหาค่าเวลาคงที่ของกระบวนการ (τ) หลังจากที่ได้ค่าพารามิเตอร์ของกระบวนการทั้ง 3 ค่าแล้วต่อไปเป็นการคำนวณหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีโดยถ้าหาก t_0 มากกว่า $\tau/4$ จะคำนวณหาค่าพารามิเตอร์ของตัวควบคุมแบบพีไอดี ถ้าไม่ใช่จะคำนวณหาค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ซึ่งการทำงานของบล็อกฟังกั้น Step Test แสดงดังรูปที่ 4.8



รูปที่ 4.8 แสดงการทำงานของบล็อกฟังก์ชัน Step test

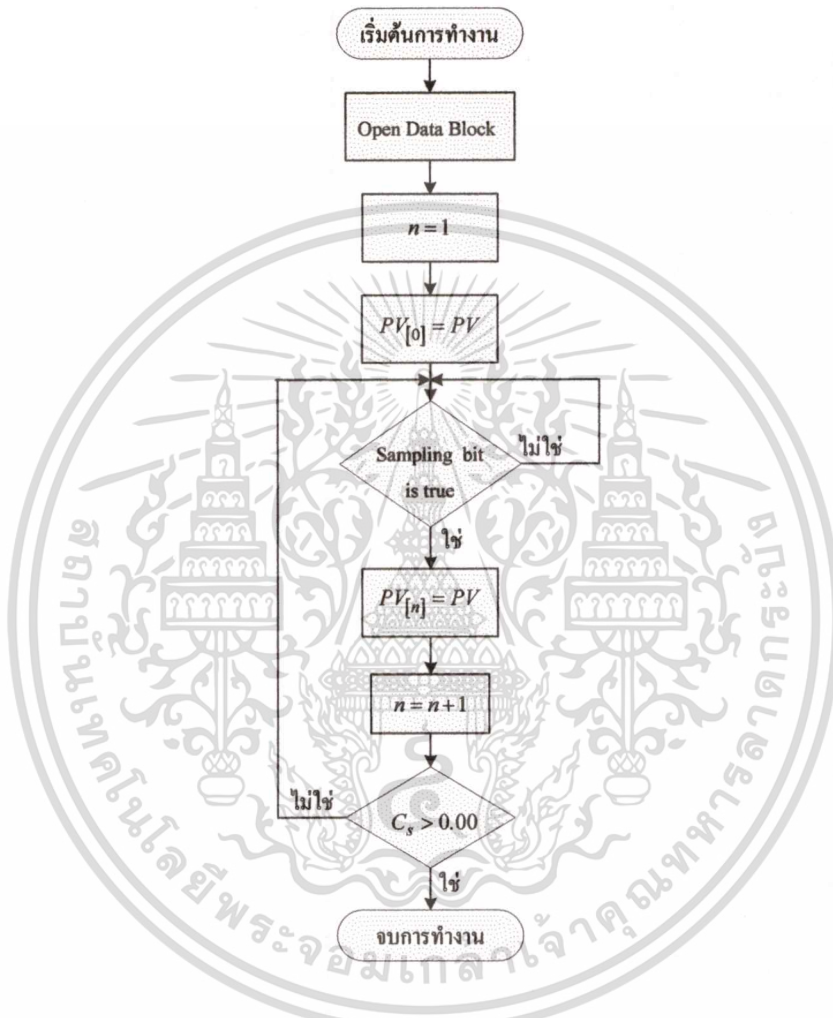
บล็อกฟังก์ชันการเก็บค่าข้อมูลที่คาบตัวอย่าง

บล็อกฟังก์ชันการเก็บค่าข้อมูลที่คาบตัวอย่างทำหน้าที่เก็บค่าผลตอบสนองของกระบวนการ

เพื่อใช้เป็นข้อมูลในการคำนวณหาค่าพารามิเตอร์ของตัวควบคุมพีไอดี โดยเมื่อเริ่มต้นผลตอบสนอง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของกระบวนการจะถูกเก็บที่ตำแหน่งเริ่มต้น ($PV_{[0]}$) และจะทำการสุ่มเก็บค่าผลตอบแทนของกระบวนการตามระยะเวลาที่กำหนดโดย Δt เมื่อครบระยะเวลาที่กำหนดบิต Sampling จะมีค่าเท่ากับ “1” ผลตอบแทนของกระบวนการจะถูกเก็บที่ค่าดัชนีบล็อกตำแหน่ง $PV_{[n]}$ จนกว่าจะได้ค่า c_s ซึ่งการทำงานของบล็อกฟังก์ชันการเก็บค่าข้อมูลที่ค่าดัชนีบล็อกแสดงดังรูปที่ 4.9

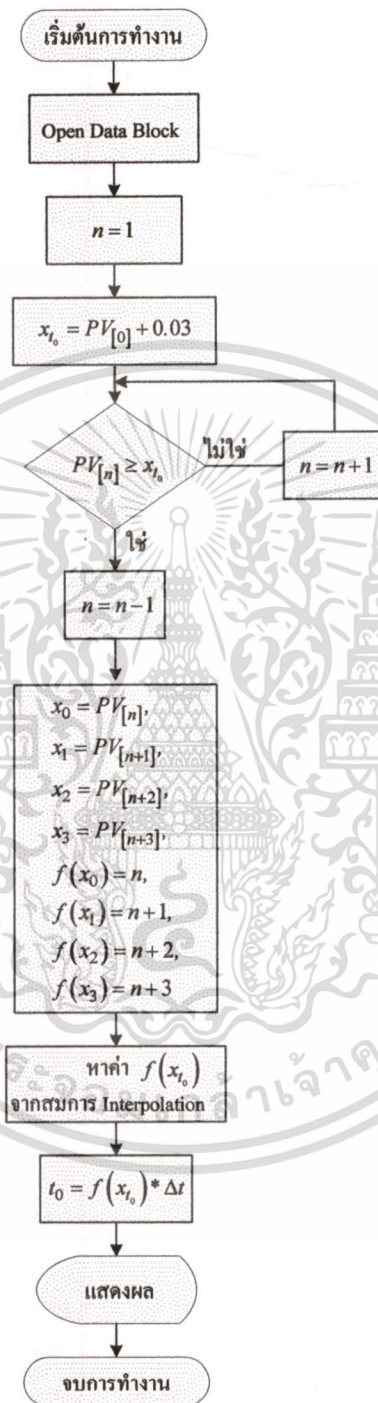


รูปที่ 4.9 แสดงการทำงานของบล็อกฟังก์ชันการเก็บค่าข้อมูลที่ค่าดัชนีบล็อก

บล็อกฟังก์ชันการหาค่า t_0

บล็อกฟังก์ชันการหาค่า t_0 ทำหน้าที่หาค่าหน่วยเวลาของกระบวนการ (t_0) โดยค่า t_0 คือค่าเวลา ณ ผลตอบแทนของกระบวนการเปลี่ยนแปลงไปประมาณ 3% ของค่าเริ่มต้น (X_{t_0}) การหาค่า t_0 ทำได้โดยหาค่า X_{t_0} ไปเปรียบเทียบกับค่า $PV_{[n]}$ ที่เก็บอยู่ในบล็อกเก็บข้อมูล โดยหาค่าที่ $PV_{[n]} \geq x_{t_0}$ เพื่อเป็นตัวแปรสำหรับการประมาณค่า (Interpolation) ตามสมการจะได้ค่าเวลา

$f(x_{t_0})$ หลังจากได้ค่า $f(x_{t_0})$ แล้วต่อไปจะคำนวณหาค่า t_0 ต่อไป ซึ่งการทำงานของบล็อกฟังก์ชันการหาค่า t_0 แสดงดังรูปที่ 4.10



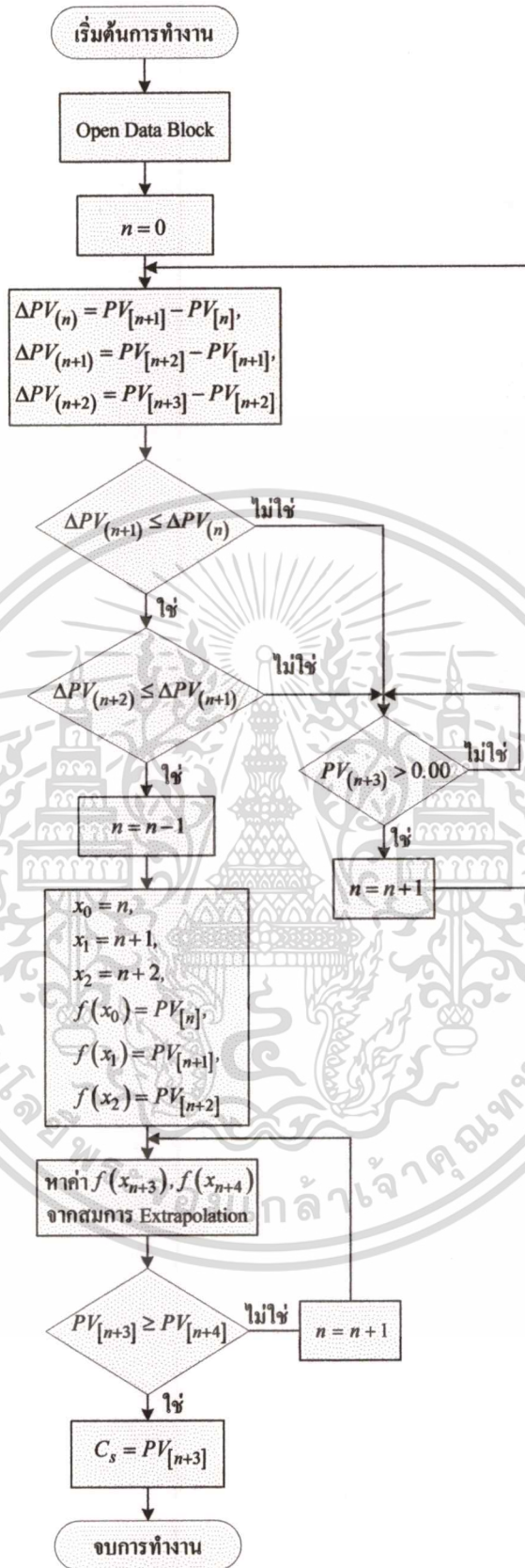
รูปที่ 4.10 แสดงการทำงานของบล็อกฟังก์ชันการหาค่า t_0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บล็อกฟังก์ชันการประมาณค่าล่วงหน้าของ c_s (Extrapolation c_s)

บล็อกฟังก์ชันการประมาณค่าล่วงหน้าของ c_s ทำหน้าที่ประมาณค่าล่วงหน้าของผลตอบสนองของกระบวนการที่สถานะคงที่ (c_s) โดยการนำค่าผลต่างระหว่างค่าผลตอบสนองของกระบวนการ ($\Delta PV_{(n)}$) จำนวน 3 ค่าคือ $\Delta PV_{(n+1)} \leq \Delta PV_{(n)}$, $\Delta PV_{(n+2)} \leq \Delta PV_{(n+1)}$ และ $\Delta PV_{(n+3)} \leq \Delta PV_{(n+2)}$ จากค่าผลต่างของค่าผลตอบสนองของกระบวนการที่ลดลงอย่างต่อเนื่อง แสดงว่าค่าผลตอบสนองของกระบวนการใกล้ที่เข้าสู่สถานะคงที่แล้ว หลังจากได้ค่าผลต่างของผลตอบสนองของกระบวนการตามเงื่อนไขข้างต้นแล้ว นำค่า $PV_{[n]}$, $PV_{[n+1]}$ และ $PV_{[n+2]}$ เป็นตัวแปรสำหรับการประมาณค่าล่วงหน้า (Extrapolation) ตามสมการเพื่อหาค่า $f(x_{n+3})$ และ $f(x_{n+4})$ ถ้าหาก $f(x_{n+3}) \geq f(x_{n+4})$ แสดงว่าได้ค่าผลตอบสนองของกระบวนการที่สถานะคงที่ (c_s) โดย c_s เท่ากับ $f(x_{n+3})$ ซึ่งการทำงานของบล็อกฟังก์ชันการประมาณค่าล่วงหน้าของ c_s แสดงดังรูปที่ 4.11



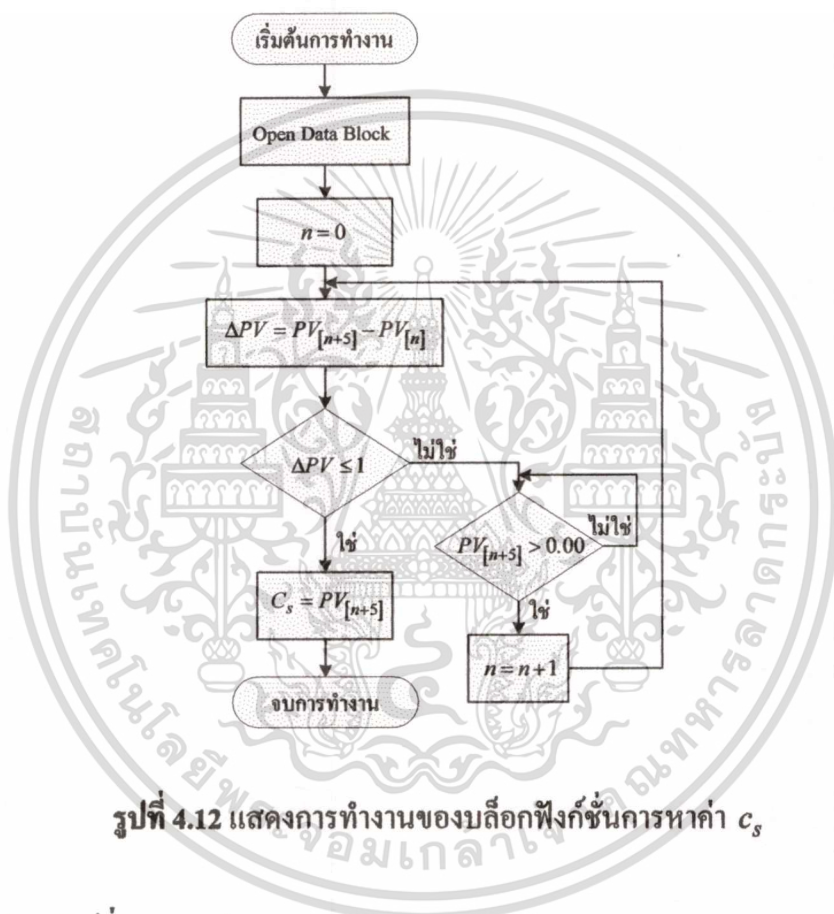


รูปที่ 4.11 แสดงการทำงานของบล็อกฟังก์ชันการประมาณค่าลวงหน้าของ c_s

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บล็อกฟังก์ชันการหาค่า c_s

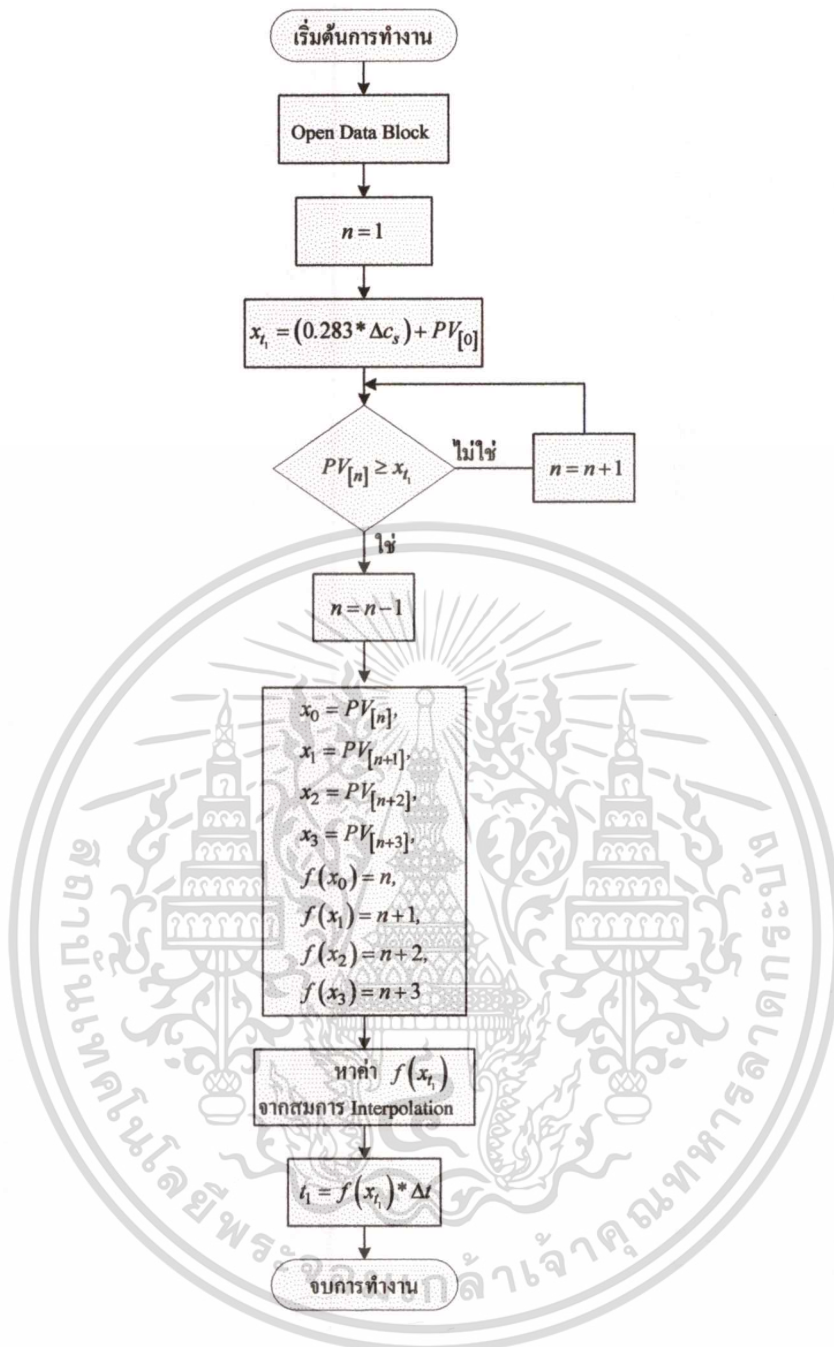
บล็อกฟังก์ชันการหาค่า c_s ทำหน้าที่หาค่าผลตอบแทนของกระบวนการที่สถานะคงที่ (c_s) โดยการนำค่าผลต่างระหว่างค่าผลตอบแทนของกระบวนการค่าที่ 5 กับค่าที่ 1 ($\Delta PV = PV_{[n+5]} - PV_{[n]}$) โดยค่าผลต่างของผลตอบแทนของกระบวนการมีค่าประมาณ 1 ($\Delta PV \leq 1$) หลังจากได้ค่า $\Delta PV \leq 1$ แล้ว จะได้ค่าผลตอบแทนของกระบวนการที่สถานะคงที่ (c_s) โดย c_s เท่ากับ $PV_{[n+5]}$ ซึ่งการทำงานของบล็อกฟังก์ชันการหาค่า c_s แสดงดังรูปที่ 4.12



รูปที่ 4.12 แสดงการทำงานของบล็อกฟังก์ชันการหาค่า c_s

บล็อกฟังก์ชันการหาค่า t_1

บล็อกฟังก์ชันการหาค่า t_1 ทำหน้าที่หาค่าเวลา t_1 โดยค่าเวลา t_1 คือค่าเวลา ณ ค่าผลตอบแทนของกระบวนการมีค่าประมาณ $0.283\Delta c_s$ (X_{t_1}) นำค่า X_{t_1} ไปเปรียบเทียบกับค่า $PV_{[n]}$ ที่เก็บอยู่ในบล็อกเก็บข้อมูล โดยหาค่าที่ $PV_{[n]} \geq x_{t_1}$ เพื่อเป็นตัวแปรสำหรับการประมาณค่า (Interpolation) ตามสมการจะได้ค่าเวลา $f(x_{t_1})$ หลังจากได้ค่า $f(x_{t_1})$ แล้วต่อไปจะคำนวณหาค่า t_1 ต่อไป ซึ่งการทำงานของบล็อกฟังก์ชันการหาค่า t_1 แสดงดังรูปที่ 4.13



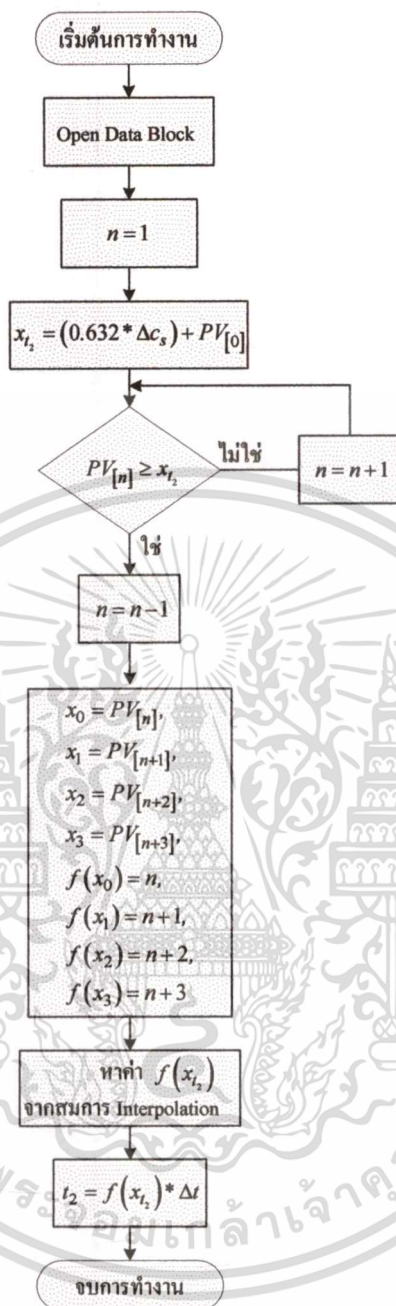
รูปที่ 4.13 แสดงการทำงานของบล็อกฟังก์ชันการหาค่า t_1

บล็อกฟังก์ชันการหาค่า t_2

บล็อกฟังก์ชันการหาค่า t_2 ทำหน้าที่หาค่าเวลา t_2 โดยค่าเวลา t_2 คือค่าเวลา ณ ค่าผลตอบแทนของกระบวนการ $0.632\Delta c_s (X_{t_2})$ นำค่า X_{t_2} ไปเปรียบเทียบกับค่า $PV[n]$ ที่เก็บอยู่ในบล็อกเก็บข้อมูล โดยหาค่าที่ $PV[n] \geq x_{t_2}$ เพื่อเป็นตัวแปรสำหรับการประมาณค่า (Interpolation) ตามสมการจะได้ค่าเวลา $f(x_{t_2})$ หลังจากได้ค่า $f(x_{t_2})$ แล้วต่อไปจะคำนวณหาค่า t_2 ต่อไป ซึ่งการทำงานของบล็อกฟังก์ชันการหาค่า t_2 แสดงดังรูปที่ 4.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



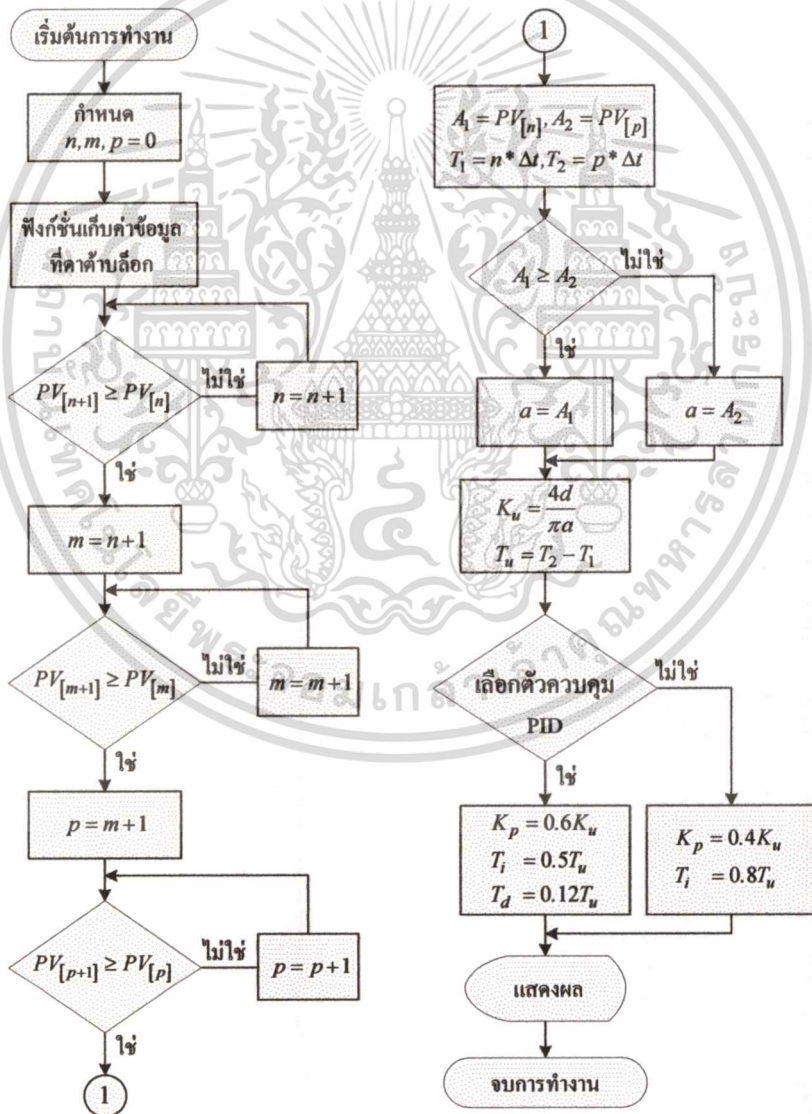
รูปที่ 4.14 แสดงการทำงานของบล็อกฟังก์ชันการหาค่า t_2

บล็อกฟังก์ชัน Relay Feedback

บล็อกฟังก์ชัน Relay Feedback ทำหน้าที่หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีโดยการป้อนสัญญาณควบคุมแบบรีเลย์ที่มีการแกว่งอย่างต่อเนื่องขนาด $+d$ และ $-d$ ให้กับกระบวนการแบบวงปิด โดยค่าผลตอบสนองของกระบวนการจะถูกเก็บไว้ที่บล็อกเก็บข้อมูล (คาต่าบล็อก) โดยเรียกบล็อกฟังก์ชันการเก็บข้อมูลที่คาต่าบล็อก ผลตอบสนองของกระบวนการหลังจากถูกป้อนด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณควบคุมแบบรีเลย์จะได้ผลตอบสนองของกระบวนการที่มีลักษณะแกว่งอย่างต่อเนื่อง โดย a คือ ขนาดของสัญญาณคลาดเคลื่อน สามารถหาได้โดยการหาค่าผลตอบสนองของกระบวนการที่มีค่าสูงที่สุดจำนวน 2 ค่าคือค่า $A_1 = PV_{[n+1]}$ และค่า $A_2 = PV_{[p+1]}$ ถ้าค่า $A_1 \geq A_2$ จะได้ $a = A_1$ ถ้าไม่ใช่จะได้ $a = A_2$ นำค่า a ไปคำนวณหาค่า K_u และค่า T_u เป็นคาบเวลาของการแกว่งอย่างต่อเนื่องก็คือผลต่างของค่าเวลาที่ค่าผลตอบสนองของกระบวนการมีค่าสูงสุด 2 ค่า ได้แก่ค่าเวลา T_1 และค่าเวลา T_2 หลังจากได้ค่า K_u และค่า T_u แล้วต่อไปเป็นการคำนวณหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีโดยถ้าเลือกการควบคุมแบบพีไอดีจะคำนวณหาค่าพารามิเตอร์ของตัวควบคุมแบบพีไอดีตามสมการ ถ้าเลือกการควบคุมแบบพีไอจะคำนวณหาค่าพารามิเตอร์ของตัวควบคุมแบบพีไอตามสมการ ซึ่งการทำงานของบล็อกฟังก์ชัน Relay Feedback แสดงดังรูปที่ 4.15

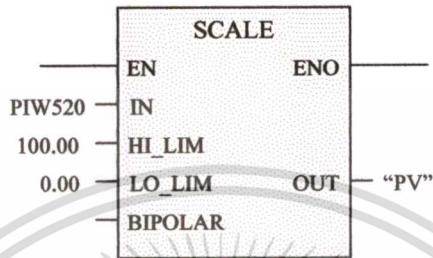


รูปที่ 4.15 แสดงการทำงานของบล็อกฟังก์ชัน Relay Feedback

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บล็อกรหัส A/D Converter

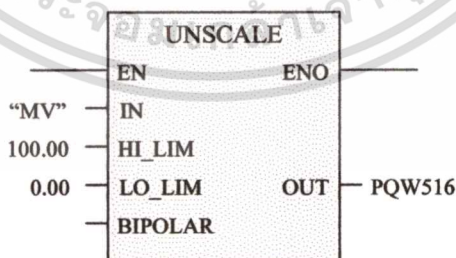
บล็อกรหัส A/D converter ในโปรแกรม STEP 7 Manager เรียกว่าบล็อกรหัส Scale (FC105) ทำหน้าที่แปลงสัญญาณอินพุตแบบอนาล็อกที่รับเข้ามาทางโมดูลอินพุตแบบอนาล็อกให้เป็นสัญญาณแบบดิจิทัล โดย PIW520 เป็นแอดเดรสของโมดูลอินพุตแบบอนาล็อกและ “PV” (MD200) เป็นหน่วยความจำแบบ Double Word ชนิด Real แอดเดรส 200 ซึ่งบล็อกรหัส SCALE แสดงดังรูปที่ 4.16



รูปที่ 4.16 แสดงบล็อกรหัส SCALE (A/D converter)

บล็อกรหัส D/A Converter

บล็อกรหัส A/D converter ในโปรแกรม STEP 7 Manager เรียกว่าบล็อกรหัส Unscale (FC106) ทำหน้าที่แปลงสัญญาณควบคุมแบบดิจิทัลให้เป็นสัญญาณแบบอนาล็อกออกไปควบคุมกระบวนการผ่านทางโมดูลเอาต์พุตแบบอนาล็อก โดย “MV” (MD300) เป็นหน่วยความจำแบบ Double Word ชนิด Real แอดเดรส 300 และ PQW 512 เป็นแอดเดรสของโมดูลเอาต์พุตแบบอนาล็อก ซึ่งบล็อกรหัส UNSCALE แสดงดังรูปที่ 4.17



รูปที่ 4.17 แสดงบล็อกรหัส UNSCALE (D/A converter)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บล็อกเก็บข้อมูล (Data Block:DB)

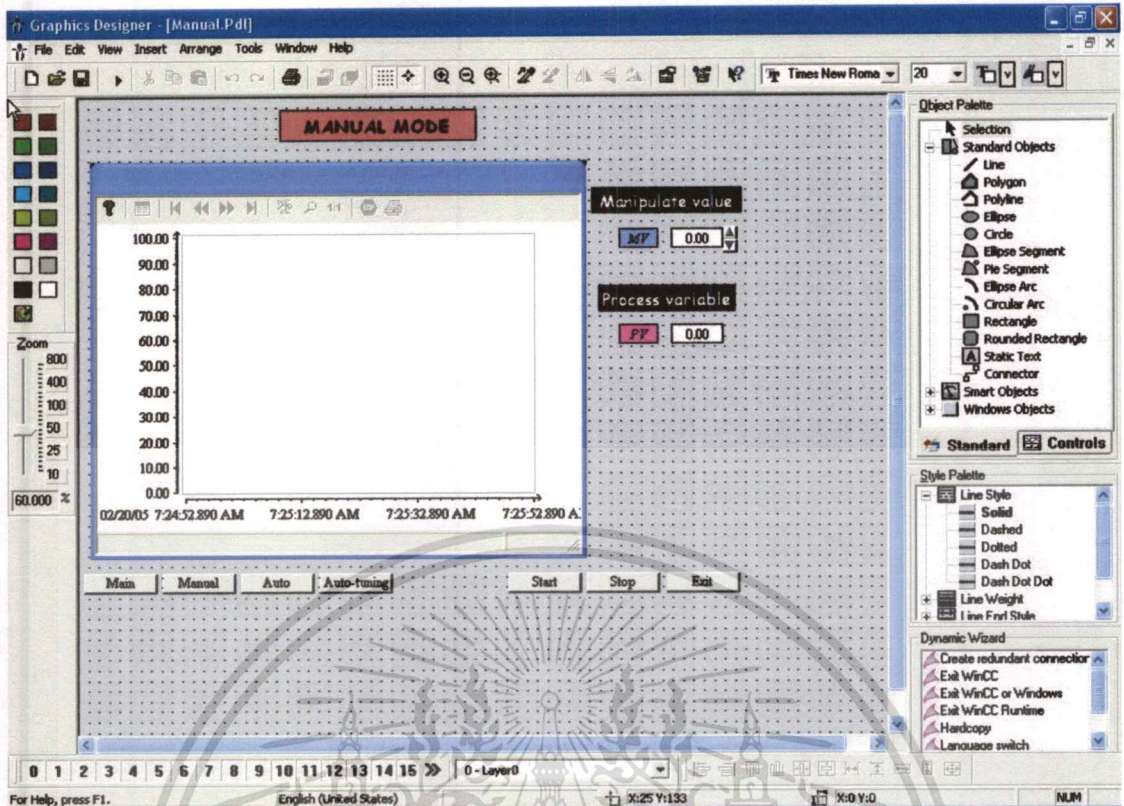
บล็อกเก็บข้อมูลทำหน้าที่เก็บค่าข้อมูล PV เพื่อใช้ในการคำนวณ โดยจะเก็บตามระยะเวลา ในการส่งข้อมูลตามฟังก์ชันการส่งเก็บข้อมูล โดยประกาศตัวแปร (Declaration) ให้ข้อมูลเป็นชนิด Real ค่าเริ่มต้นมีค่า 0.00 ซึ่งบล็อกเก็บข้อมูล แสดงดังรูปที่ 4.18

Address	Name	Type	Initial value	Actual value	Comment
0.0	PV_Data[0]	REAL	0.000000e+000	0.000000e+000	Temporary placeholder variable
4.0	PV_Data[1]	REAL	0.000000e+000	0.000000e+000	
8.0	PV_Data[2]	REAL	0.000000e+000	0.000000e+000	
12.0	PV_Data[3]	REAL	0.000000e+000	0.000000e+000	
16.0	PV_Data[4]	REAL	0.000000e+000	0.000000e+000	
20.0	PV_Data[5]	REAL	0.000000e+000	0.000000e+000	
24.0	PV_Data[6]	REAL	0.000000e+000	0.000000e+000	
28.0	PV_Data[7]	REAL	0.000000e+000	0.000000e+000	
32.0	PV_Data[8]	REAL	0.000000e+000	0.000000e+000	
36.0	PV_Data[9]	REAL	0.000000e+000	0.000000e+000	
40.0	PV_Data[10]	REAL	0.000000e+000	0.000000e+000	
44.0	PV_Data[11]	REAL	0.000000e+000	0.000000e+000	
48.0	PV_Data[12]	REAL	0.000000e+000	0.000000e+000	
52.0	PV_Data[13]	REAL	0.000000e+000	0.000000e+000	
56.0	PV_Data[14]	REAL	0.000000e+000	0.000000e+000	
60.0	PV_Data[15]	REAL	0.000000e+000	0.000000e+000	
64.0	PV_Data[16]	REAL	0.000000e+000	0.000000e+000	
68.0	PV_Data[17]	REAL	0.000000e+000	0.000000e+000	
72.0	PV_Data[18]	REAL	0.000000e+000	0.000000e+000	
76.0	PV_Data[19]	REAL	0.000000e+000	0.000000e+000	
80.0	PV_Data[20]	REAL	0.000000e+000	0.000000e+000	
84.0	PV_Data[21]	REAL	0.000000e+000	0.000000e+000	
88.0	PV_Data[22]	REAL	0.000000e+000	0.000000e+000	
92.0	PV_Data[23]	REAL	0.000000e+000	0.000000e+000	
96.0	PV_Data[24]	REAL	0.000000e+000	0.000000e+000	
100.0	PV_Data[25]	REAL	0.000000e+000	0.000000e+000	
104.0	PV_Data[26]	REAL	0.000000e+000	0.000000e+000	
108.0	PV_Data[27]	REAL	0.000000e+000	0.000000e+000	
112.0	PV_Data[28]	REAL	0.000000e+000	0.000000e+000	
116.0	PV_Data[29]	REAL	0.000000e+000	0.000000e+000	

รูปที่ 4.18 แสดงบล็อกเก็บข้อมูล

4.4.2 การเขียนโปรแกรมแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์

การเขียนโปรแกรมแสดงผลการทำงานทางหน้าจอคอมพิวเตอร์ใช้โปรแกรมของบริษัท ซิเมนส์ จำกัด (SIEMENS) คือ โปรแกรม WinCC (Windows Control Center) เป็นโปรแกรมที่ใช้ในการออกแบบส่วนกำหนดค่าการทำงานและแสดงผลต่างๆ แสดงดังรูปที่ 4.19



รูปที่ 4.19 แสดงตัวอย่างการเขียนโปรแกรมแสดงผลการทำงานหน้าจอกอมพิวเตอร์

4.5 การใช้งานตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC

เมื่อทำการติดตั้งเครื่องควบคุม PLC เข้ากับกระบวนการและกำหนดค่าทางด้านซอฟต์แวร์คือ Hw Config. ให้ตรงกับ การติดตั้งเรียบร้อยแล้ว ขั้นตอนต่อไปคือการควบคุมการทำงาน ซึ่งอธิบายตามหน้าจอของการใช้งานได้ดังนี้

4.5.1 หน้าจอหลัก

หน้าจอหลักเป็นหน้าจอที่ใช้สำหรับกำหนดคุณลักษณะของการควบคุม ประกอบด้วย

1). ปุ่มเลือกแบบการควบคุมกระบวนการ มีด้วยกัน 3 แบบ คือ

- แบบ Manual คือ การควบคุมกระบวนการ โดยผู้ใช้งานกำหนดค่าสัญญาณควบคุม (mv) ตามต้องการ โดยการกดปุ่มเลื่อนขึ้นเลื่อนลงหรือป้อนค่าได้ โดยค่าสัญญาณควบคุมอยู่ในช่วงระหว่าง 0.00 – 100.00%

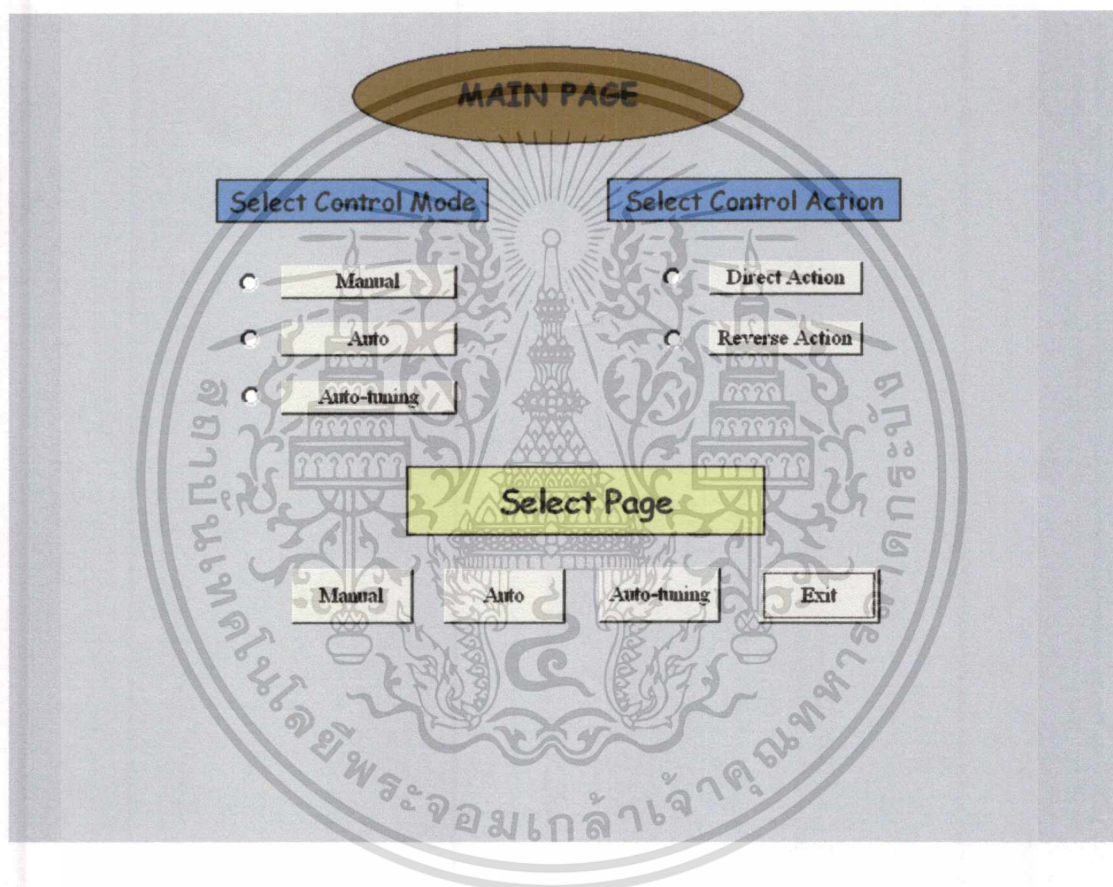
- แบบ Auto คือ การควบคุมกระบวนการแบบอัตโนมัติโดยผู้ใช้งานต้องกำหนดค่าพารามิเตอร์ของตัวควบคุม PID และค่าเป้าหมายหรือค่าอ้างอิงของกระบวนการเพื่อให้เครื่องควบคุม PLC ทำการควบคุมกระบวนการได้อย่างอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แบบ Auto-tuning คือ การปรับค่าพารามิเตอร์ของตัวควบคุมแบบอัตโนมัติ โดยผู้ใช้งานสามารถเลือกวิธีการปรับค่าพารามิเตอร์ของตัวควบคุมได้ ซึ่งมีด้วยกัน 2 วิธี คือ วิธี Step response test และวิธี Relay Feedback

2). ปุ่มเลือกลักษณะการทำงานของอุปกรณ์ควบคุม ซึ่งมีด้วยกัน 2 ลักษณะ คือ

- Direct Action
- Reverse Action



รูปที่ 4.20 แสดงหน้าจอหลัก

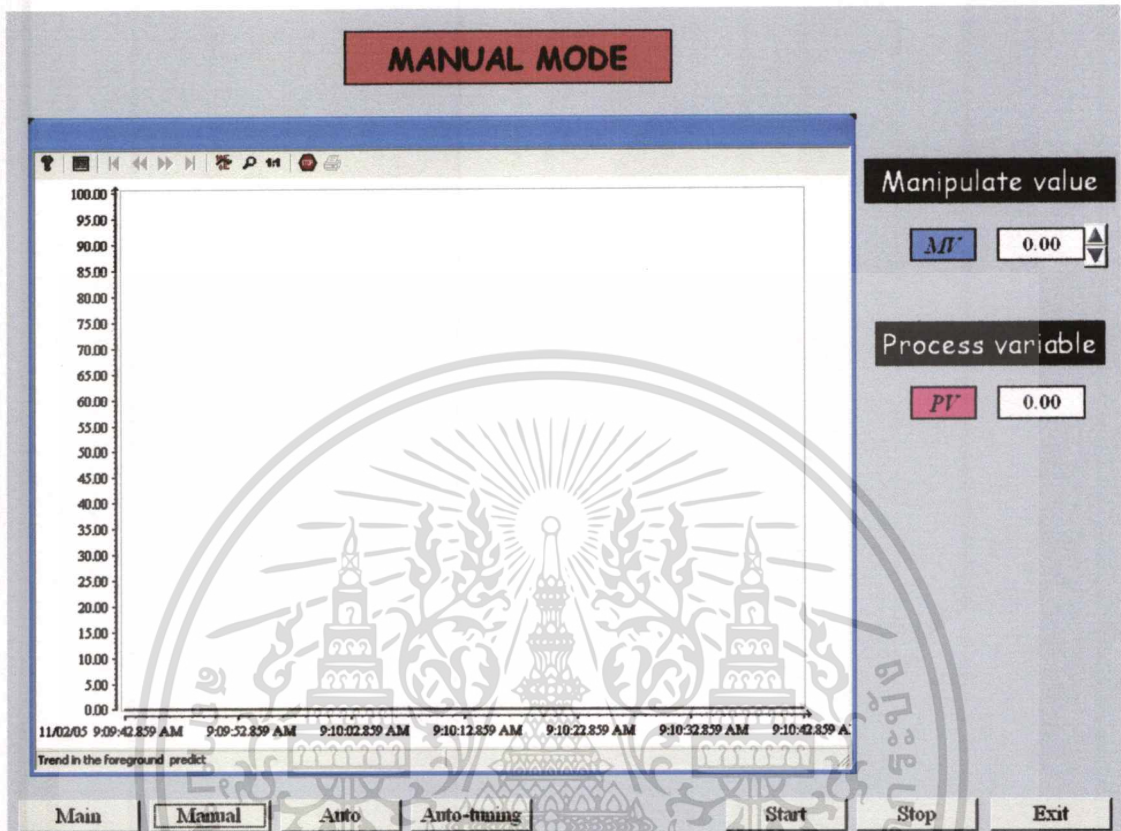
4.5.2 หน้าจอการควบคุมแบบ Manual

หน้าจอการควบคุมแบบ Manual ประกอบด้วย

- 1) ช่องสำหรับป้อนค่าสัญญาณควบคุม (Manipulate Value: mv) โดยผู้ใช้งานสามารถกำหนดค่าสัญญาณ mv ได้ตามต้องการโดยกดปุ่มลูกศรเลื่อนขึ้นเลื่อนลง โดยการเพิ่มขึ้นหรือลดลงนี้มีความละเอียด 0.01 หรือป้อนค่าสัญญาณควบคุมได้โดยตรง ค่าสัญญาณควบคุมนี้มีค่าอยู่ในช่วงระหว่าง 0.00 – 100.00%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2). ช่องแสดงค่าผลตอบสนองของกระบวนการ (Process Value: pv)
- 3). กราฟแสดงค่าสัญญาณควบคุม (mv) และค่าสัญญาณผลตอบสนองของกระบวนการ (pv)



รูปที่ 4.21 แสดงหน้าจอการควบคุมแบบ Manual

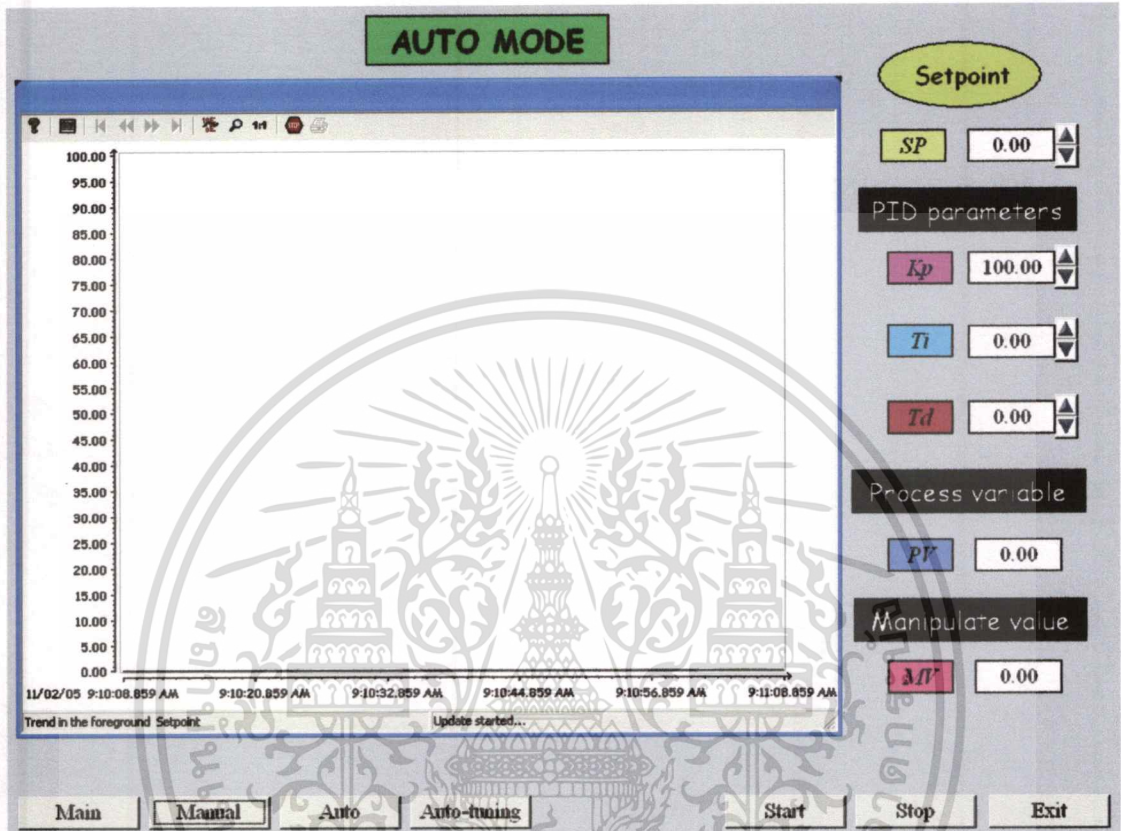
4.5.3 หน้าจอการควบคุมแบบ Automatic

หน้าจอการควบคุมแบบ Automatic ประกอบด้วย

- 1). ช่องสำหรับป้อนค่าเป้าหมายของกระบวนการหรือค่าอ้างอิง (Set point: sp) โดยผู้ใช้งานสามารถกำหนดค่า sp ได้ตามต้องการโดยกดปุ่มลูกศรเลื่อนขึ้นเลื่อนลง โดยการเพิ่มขึ้นหรือลดลงนี้มีความละเอียด 0.01 หรือป้อนค่าโดยตรง ค่า sp นี้มีค่าอยู่ในช่วงระหว่าง 0.00 – 100.00%
- 2). ช่องสำหรับป้อนค่าพารามิเตอร์ของตัวควบคุมพีไอดี (K_p, T_i และ T_d) โดยผู้ใช้งานสามารถกำหนดค่าพารามิเตอร์ต่างๆ ได้ตามต้องการโดยกดปุ่มลูกศรเลื่อนขึ้นเลื่อนลง โดยการเพิ่มขึ้นหรือลดลงนี้มีความละเอียด 0.01 หรือป้อนค่าพารามิเตอร์โดยตรง ซึ่งค่า T_i และ T_d มีหน่วยเป็นวินาที
- 3). ช่องแสดงค่าผลตอบสนองของกระบวนการ (pv)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 4). ช่องแสดงค่าสัญญาณควบคุม (mv)
- 5). กราฟแสดงค่าเป้าหมาย (sp) ค่าสัญญาณควบคุม (mv) และค่าสัญญาณผลตอบสนองของกระบวนการ (pv)



รูปที่ 4.22 แสดงหน้าจอการควบคุมแบบ Automatic

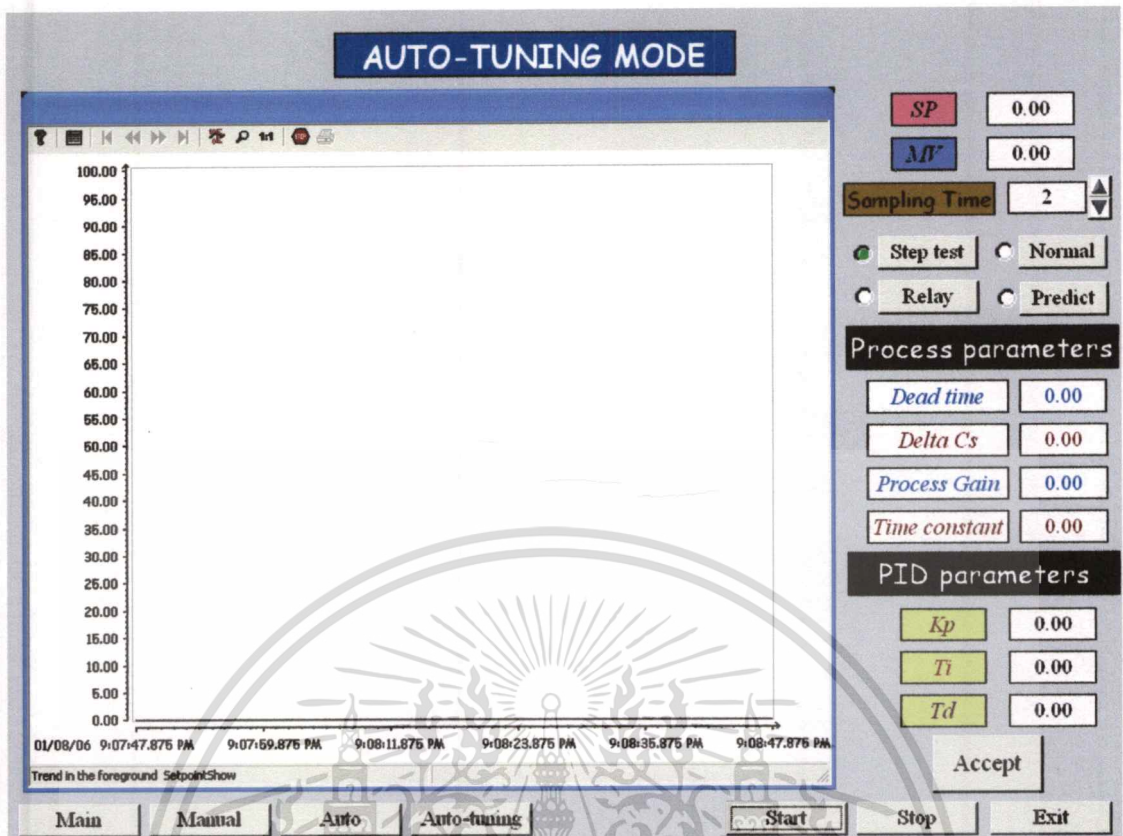
4.5.4 หน้าจอการควบคุมแบบ Auto-tuning

หน้าจอการควบคุมแบบ Auto-tuning ประกอบด้วย

- 1). ปุ่มเลือกแบบการปรับค่าอัตโนมัติ
 - แบบ Step response คือ การปรับค่าอัตโนมัติโดยใช้หลักการ Step response test
 - แบบ Relay feedback คือ การปรับค่าอัตโนมัติโดยใช้หลักการ Relay feedback
- 2). ปุ่มเลือกแบบการคำนวณหาค่าพารามิเตอร์ของกระบวนการ เมื่อเลือกการปรับค่าอัตโนมัติแบบ Step response test
 - แบบ Normal คือ การรอให้ผลตอบสนองของกระบวนการเข้าสู่สภาวะคงที่
 - แบบ Predict คือ การประมาณค่าของผลตอบสนองของกระบวนการที่สภาวะคงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3). ช่องสำหรับป้อนค่าเป้าหมายของกระบวนการหรือค่าอ้างอิง (sp) โดยผู้ใช้งานสามารถกำหนดค่า sp ได้ตามต้องการโดยกดปุ่มลูกศรเลื่อนขึ้นเลื่อนลง โดยการเพิ่มขึ้นหรือลดลงนี้มีความละเอียด 0.01 หรือป้อนค่าโดยตรง ค่า sp นี้มีค่าอยู่ในช่วงระหว่าง 0.00 – 100.00%
- 4). ช่องสำหรับป้อนค่าความเร็วในการสุ่มเก็บข้อมูลเพื่อใช้ในการคำนวณค่าพารามิเตอร์ของกระบวนการ โดยผู้ใช้งานสามารถกำหนดค่าความเร็วในการสุ่มเก็บข้อมูลได้ตามต้องการโดยกดปุ่มลูกศรเลื่อนขึ้นเลื่อนลง โดยการเพิ่มขึ้นหรือลดลงนี้มีความละเอียด 0.01 หรือป้อนค่าโดยตรง ค่าความเร็วมีหน่วยเป็นวินาทีต่อครั้ง
- 5). ช่องสำหรับป้อนค่าสัญญาณควบคุม (mv) ใช้ในการปรับค่าแบบ Step response test เพื่อใช้เป็นสัญญาณควบคุมในการทดสอบหาค่าพารามิเตอร์ของกระบวนการ โดยผู้ใช้งานสามารถกำหนดค่าสัญญาณ mv ได้ตามต้องการโดยกดปุ่มลูกศรเลื่อนขึ้นเลื่อนลง โดยการเพิ่มขึ้นหรือลดลงนี้มีความละเอียด 0.01 หรือป้อนค่าสัญญาณควบคุมได้โดยตรง ค่าสัญญาณควบคุมนี้มีค่าอยู่ในช่วงระหว่าง 0.00 – 100.00%
- 6). ช่องแสดงค่าผลตอบสนองของกระบวนการ (pv)
- 7). ช่องแสดงค่าพารามิเตอร์ของกระบวนการ (K, t_0 และ τ)
- 8). ช่องแสดงค่าพารามิเตอร์ของตัวควบคุมพีไอดี (K_p, T_i และ T_d)
- 9). กราฟแสดงค่าเป้าหมาย (sp) ค่าสัญญาณควบคุม (mv) และค่าสัญญาณผลตอบสนองของกระบวนการ (pv)
- 10). ปุ่ม Accept คือปุ่มยอมรับค่าพารามิเตอร์ที่คำนวณ เพื่อเปลี่ยนโหมดการควบคุมไปสู่โหมด Automatic โดยกระบวนการจะถูกควบคุมโดยใช้ค่าพารามิเตอร์ของตัวควบคุมพีไอดี (K_p, T_i และ T_d) ใหม่ทันที



รูปที่ 4.23 แสดงหน้าจอการควบคุมแบบ Auto-tuning

การกำหนดช่วงเวลาในการสุ่มเก็บข้อมูลในแต่ละครั้ง ควรพิจารณาจากผลตอบสนองของกระบวนการ หากกระบวนการมีผลตอบสนองที่เร็ว เช่น กระบวนการควบคุมระดับน้ำ ควรเลือกใช้ค่าต่ำๆ เช่น 5 วินาที/ครั้ง เป็นต้น และหากกระบวนการมีผลตอบสนองของกระบวนการที่ช้า เช่น กระบวนการควบคุมอุณหภูมิ ควรเลือกใช้ค่าสูงๆ เช่น 30 วินาที/ครั้ง เป็นต้น

บทที่ 5

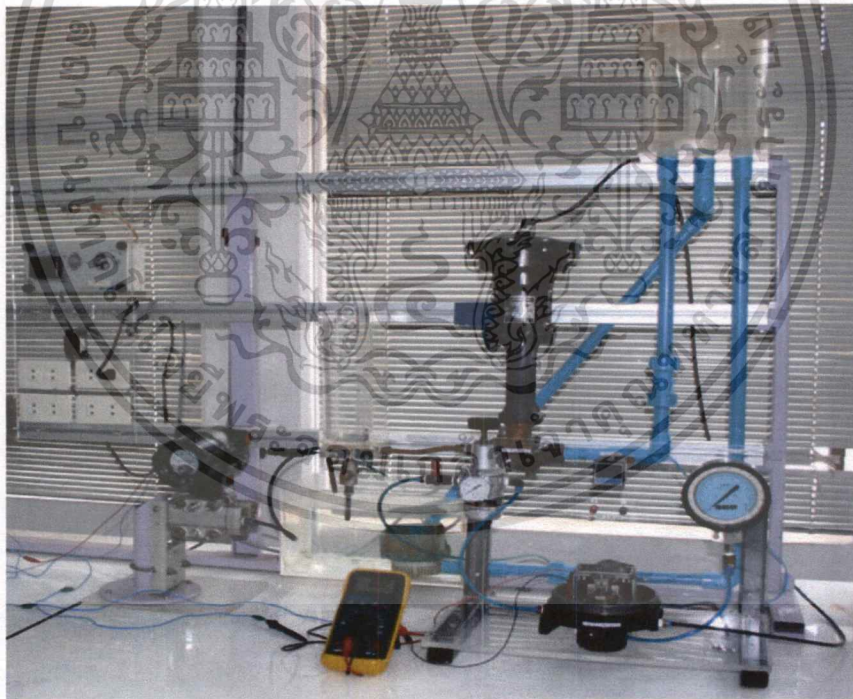
ผลการทดลอง

5.1 กล่าวนำ

ในบทนี้จะเป็นการนำตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC ไปประยุกต์ใช้ทดลองกับกระบวนการจำลอง 2 กระบวนการ ได้แก่ กระบวนการจำลองการควบคุมระดับน้ำและกระบวนการจำลองการควบคุมอุณหภูมิ โดยทำการทดลองให้เห็นผลตอบสนองของกระบวนการที่ค่าเป้าหมายต่างๆ

5.2 ผลการทดลองกับกระบวนการควบคุมระดับน้ำ

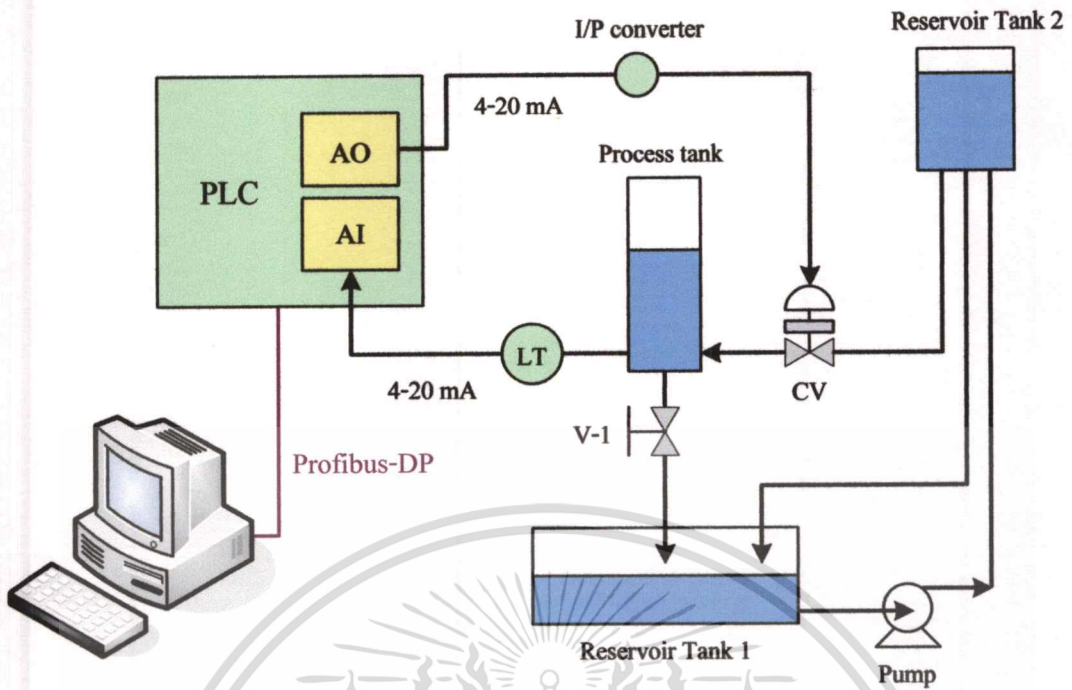
การทดลองนี้เป็นการประยุกต์ใช้ตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC กับกระบวนการควบคุมระดับน้ำ ดังรูปที่ 5.1



รูปที่ 5.1 แสดงการประยุกต์ใช้ตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC ควบคุมระดับน้ำในถัง

จากรูปที่ 5.1 นำมาเขียนเป็น โครงสร้างการทดลองกับกระบวนการควบคุมระดับน้ำให้ชัดเจนได้ดังรูปที่ 5.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



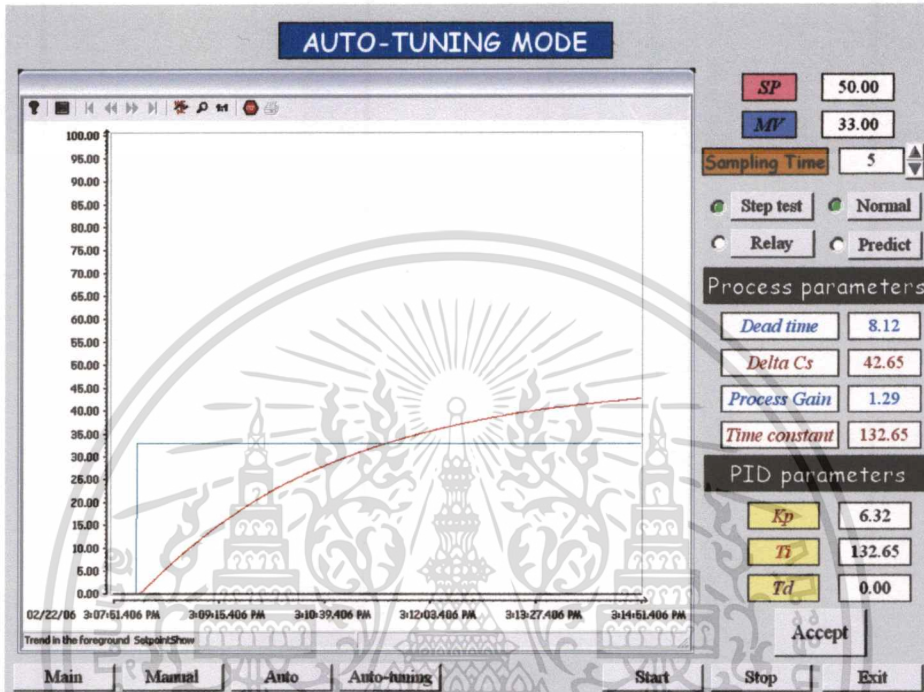
รูปที่ 5.2 แสดงโครงสร้างการทดลองกับกระบวนการควบคุมระดับน้ำ

การทำงานของกรทดลอง ป้อนน้ำจะทำน้ำที่สูบน้ำจากถังเก็บน้ำ (Reservoir Tank 1) ส่งน้ำขึ้นไปตามท่อตลอดระยะเวลาการทดลอง ซึ่งน้ำที่ถูกสูบจะถูกเก็บไว้ในถังพักน้ำ (Reservoir Tank 2) ถ้าหากน้ำในถังพักน้ำเต็ม น้ำในส่วนที่ล้นก็จะไหลลงสู่ถังเก็บน้ำ (Reservoir Tank 1) และน้ำในถังพักน้ำ (Reservoir Tank 2) จะถูกส่งไปยังถังกระบวนการ (Process Tank) ผ่านวาล์วควบคุม (CV) วาล์วควบคุมเป็นชนิดปกติเปิด (Normally Open) และน้ำที่อยู่ในถังกระบวนการ (Process Tank) จะไหลผ่านวาล์วหมายเลข 1 (V-1) ซึ่งเปิดตลอดเวลาเพื่อให้น้ำไหลลงไปที่ถังเก็บน้ำ (Tank 1) เริ่มแรกเมื่อเลือกการควบคุมแบบกลับ (Reverse Action) เครื่องควบคุม PLC จะสั่งวาล์วควบคุมให้ปิดสนิท เมื่อเริ่มต้นการทำงานของเครื่องควบคุม PLC จะส่งสัญญาณควบคุมออกทางโมดูลเอาต์พุตแบบอนาลอกเป็นสัญญาณกระแสไฟฟ้าแบบขั้วบับนโค ขนาด Δm (4-20 mADC) มายังอุปกรณ์แปลงสัญญาณกระแสไฟฟ้าเป็นความดันลม (Current To Pneumatic Converter) เพื่อสั่งให้วาล์วเปิดตามอัตราส่วนของสัญญาณกระแสไฟฟ้า น้ำที่ถูกสูบก็สามารถไหลเข้าสู่ถังกระบวนการได้ ซึ่งระดับน้ำในถังกระบวนการจะถูกวัดโดย LT (Level Transmitter) และแปลงเป็นสัญญาณกระแสไฟฟ้าขนาด 4-20 mADC และถูกป้อนเป็นสัญญาณอินพุตให้กับเครื่องควบคุม PLC เข้าทางโมดูลอินพุตแบบอนาลอกเพื่อทำการเก็บข้อมูลที่ได้ออกจากการสุ่มเก็บค่าผลตอบสนองของกระบวนการและเครื่องควบคุม PLC ก็จะเริ่มทำการจับเวลาตั้งแต่เริ่มต้นการทำงานและนำข้อมูลที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้จากการสุ่มไปใช้ในการคำนวณค่าพารามิเตอร์ต่างๆ ตามหลักการข้างต้นต่อไป ซึ่งจะแสดงผลที่ได้ต่างๆ ทางหน้าจอคอมพิวเตอร์

การหาค่าพารามิเตอร์ของกระบวนการจากการทดสอบกระบวนการในโหมด Step test แบบ Normal จากผลตอบสนองของกระบวนการของการทดลองควบคุมระดับน้ำ แสดงดังรูปที่ 5.3



รูปที่ 5.3 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำจากการทดสอบโดยป้อนสัญญาณทดสอบขนาด $\Delta m = 30\%$ ในโหมด Step Test แบบ Normal

จากการทดสอบกระบวนการที่ค่าสัญญาณทดสอบ ได้ค่าพารามิเตอร์ของกระบวนการ ดังนี้

$$K = 1.29, t_0 = 8.12 \text{ sec. และ } \tau = 132.65 \text{ sec.}$$

และได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ดังนี้

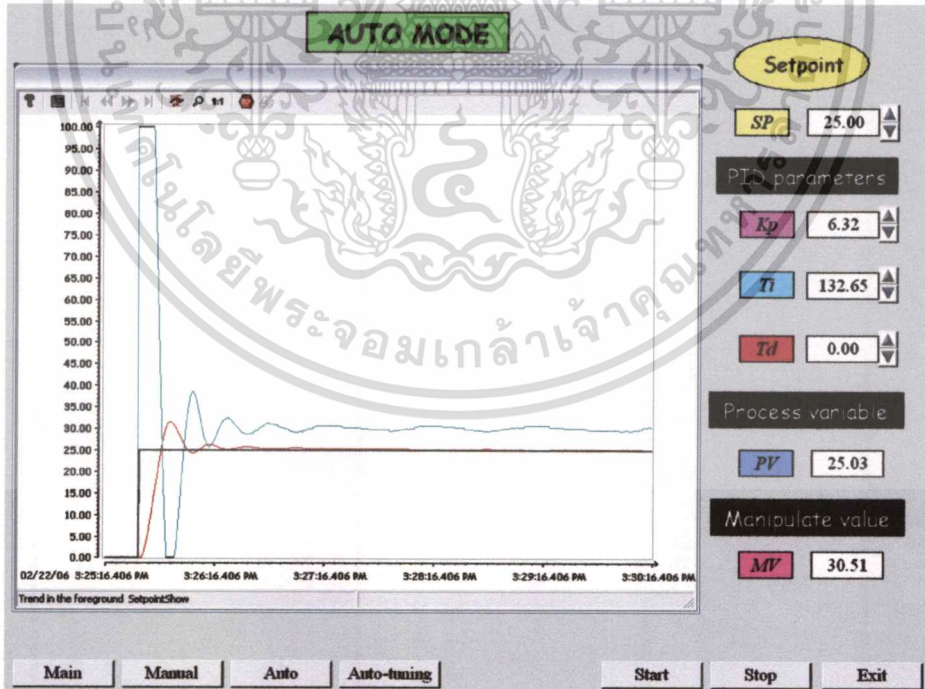
$$K_p = 6.32 \text{ และ } T_i = 132.65 \text{ sec.}$$

นำค่าพารามิเตอร์ที่ได้นี้ไปควบคุมกระบวนการที่ค่าเป้าหมาย 25%, 50% และ 75% ตามลำดับ แสดงดังรูปที่ 5.4 ถึง 5.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

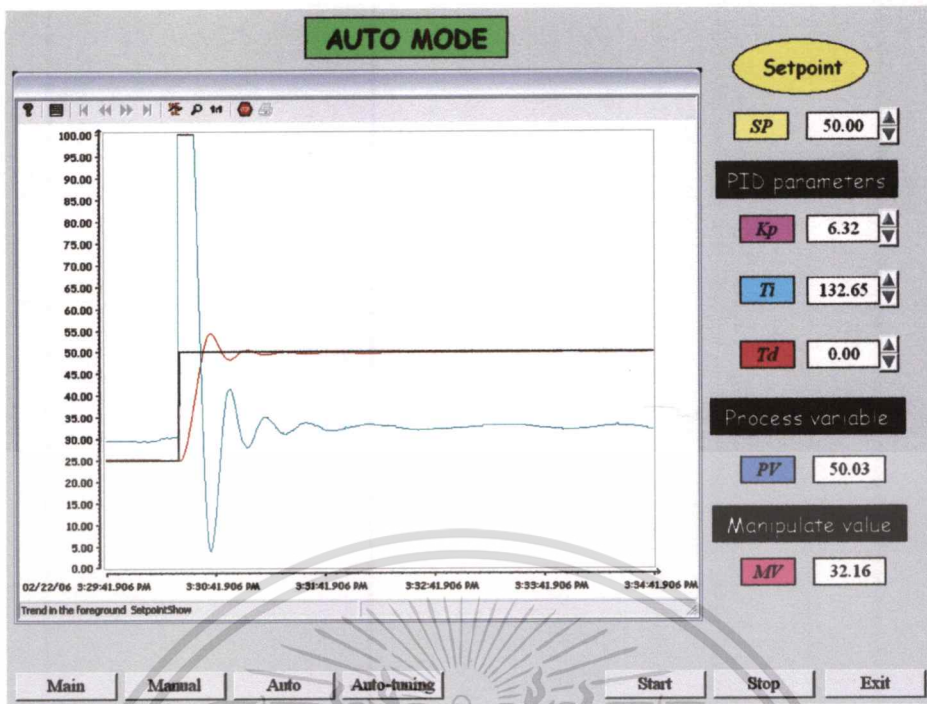


รูปที่ 5.4 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.)

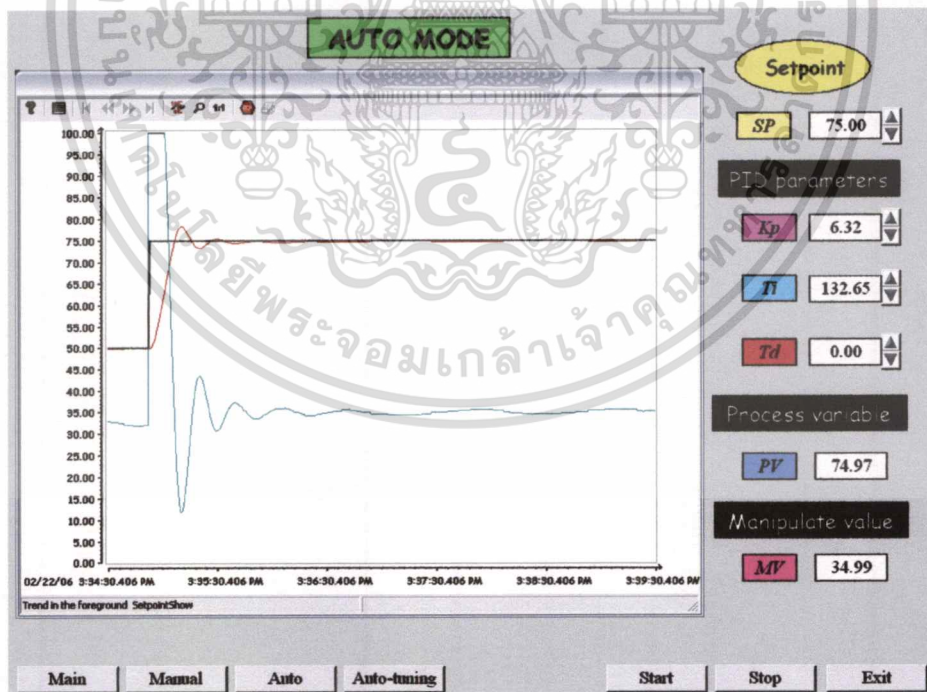


รูปที่ 5.5 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% ด้วยการควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% และ Step change ไปที่ 50% ด้วยการควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.)

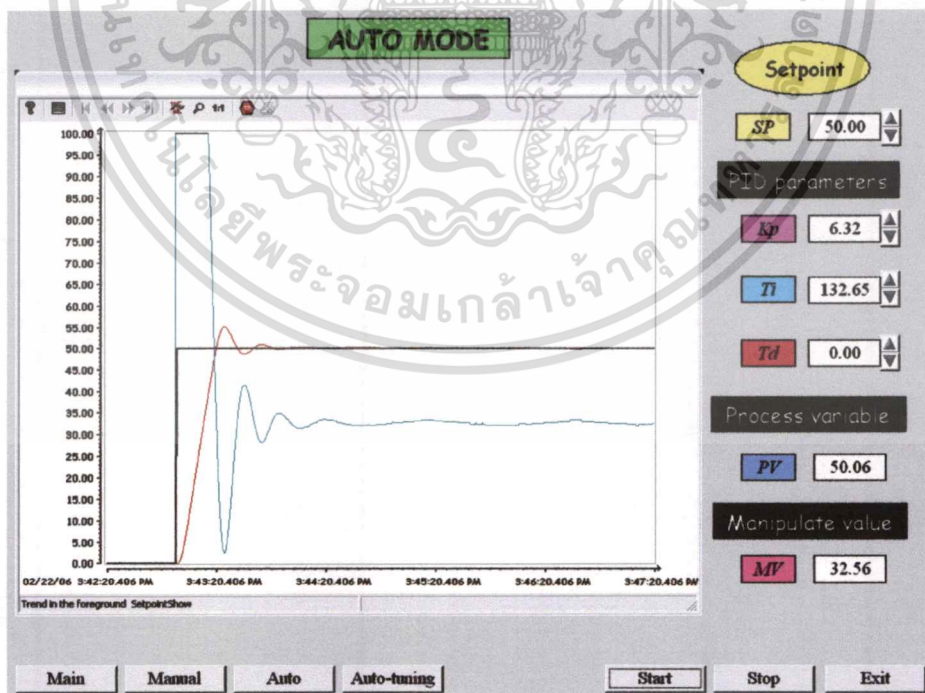


รูปที่ 5.7 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% และ Step change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.8 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% และ Step change ไปที่ 50% และ Step change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.)



รูปที่ 5.9 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% ด้วยการควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.)

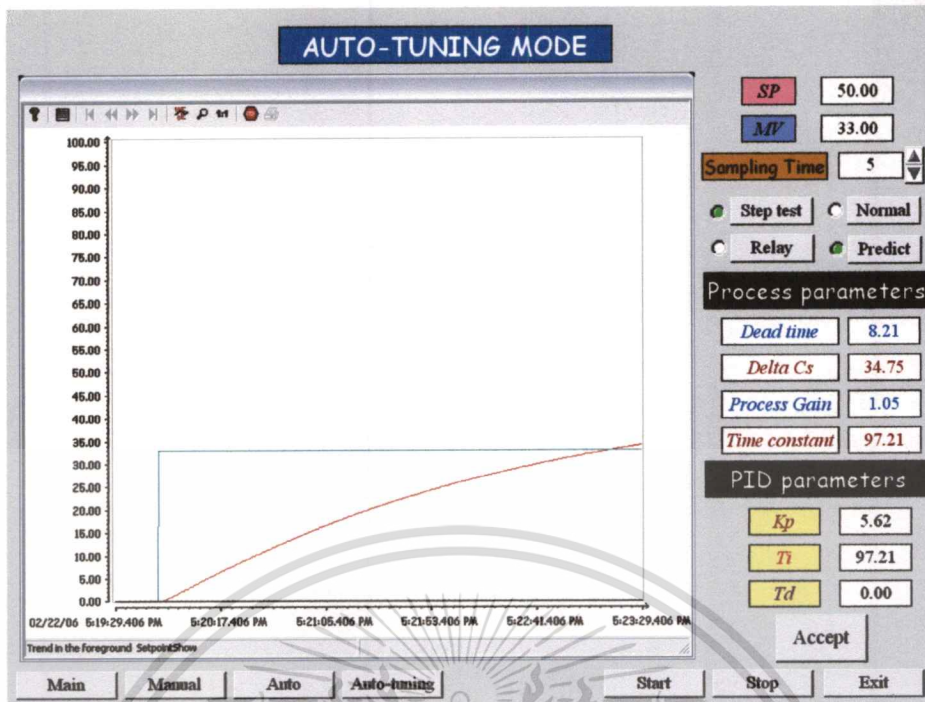
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.10 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 75% ด้วยการควบคุมแบบพีไอ ($K_p = 6.32$ และ $T_i = 132.65$ sec.)

การหาค่าพารามิเตอร์ของกระบวนการจากการทดสอบกระบวนการในโหมด Step Test แบบ Predict จากผลตอบสนองของกระบวนการของการทดลองควบคุมระดับน้ำ แสดงดังรูปที่

5.11



รูปที่ 5.11 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำจากการทดสอบ โดยป้อนสัญญาณทดสอบขนาด $\Delta m = 30\%$ ในโหมด Step Test แบบ Predict

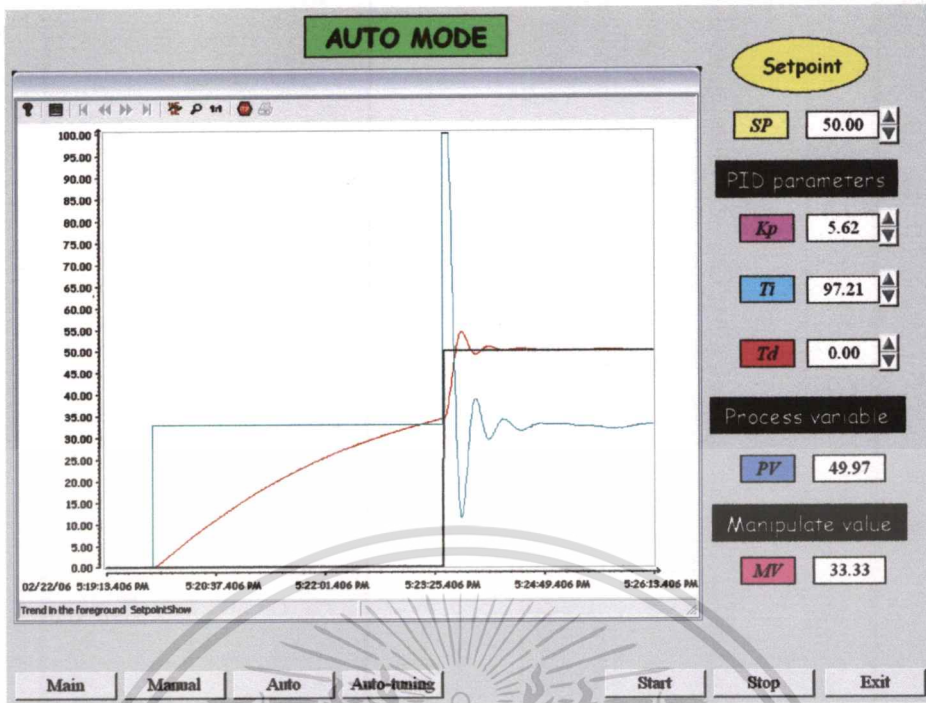
จากการทดสอบกระบวนการที่ค่าสัญญาณทดสอบ ได้ค่าพารามิเตอร์ของกระบวนการ ดังนี้

$$K = 1.05, t_0 = 8.21 \text{ sec. และ } \tau = 97.21 \text{ sec.}$$

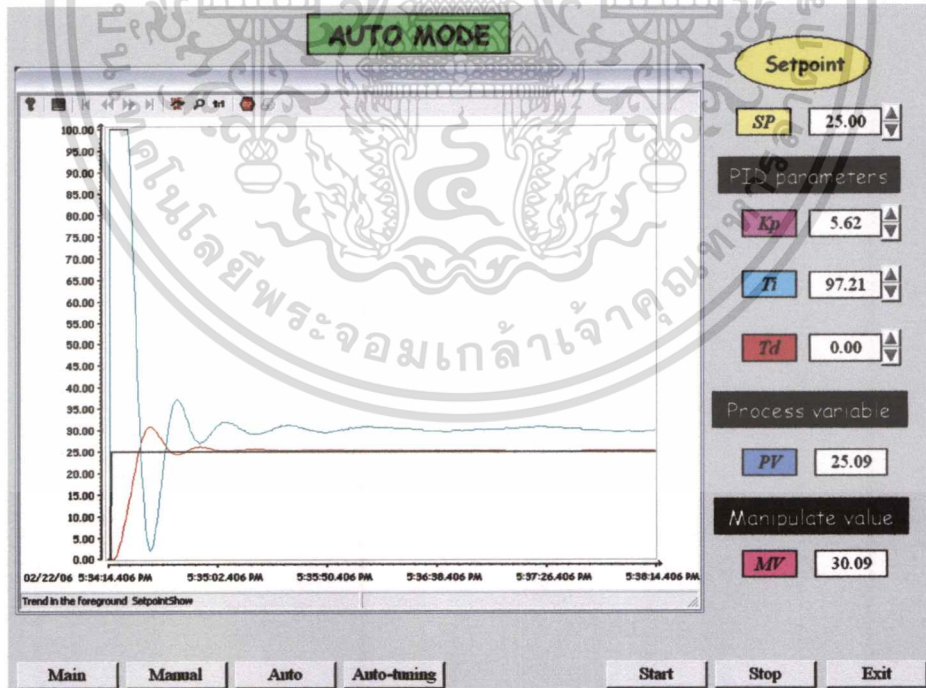
และได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ดังนี้

$$K_p = 5.62 \text{ และ } T_i = 97.21 \text{ sec.}$$

นำค่าพารามิเตอร์ที่ได้นี้ไปควบคุมกระบวนการที่ค่าเป้าหมาย 25%, 50% และ 75% ตามลำดับ แสดงดังรูปที่ 5.12 ถึง 5.17

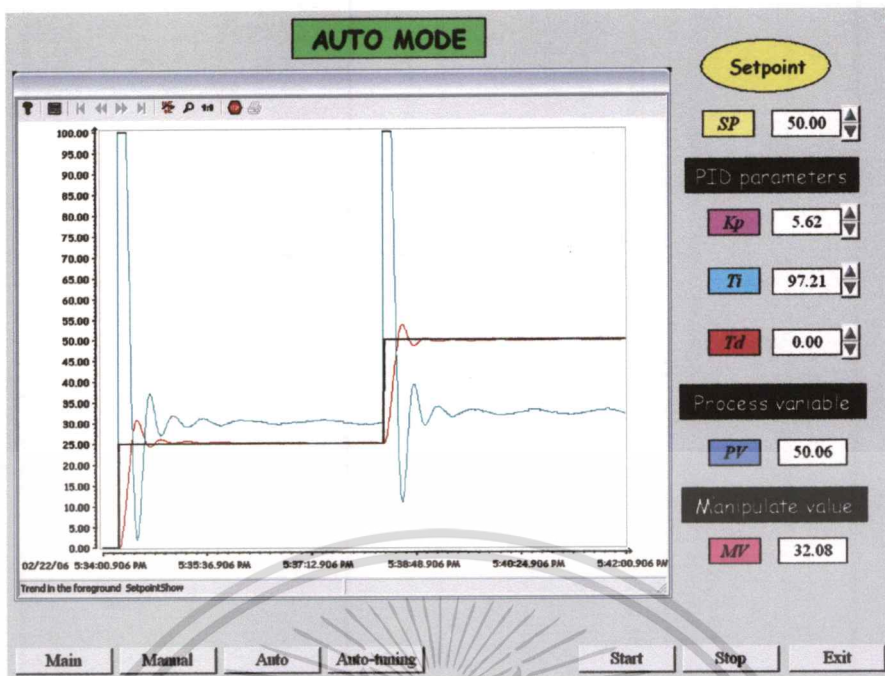


รูปที่ 5.12 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.)

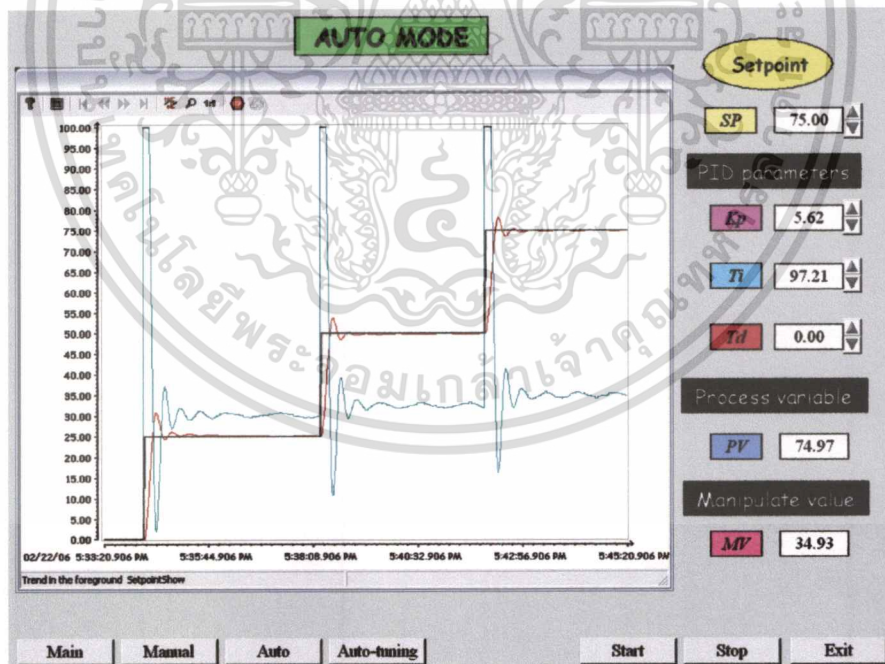


รูปที่ 5.13 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% ด้วยการควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

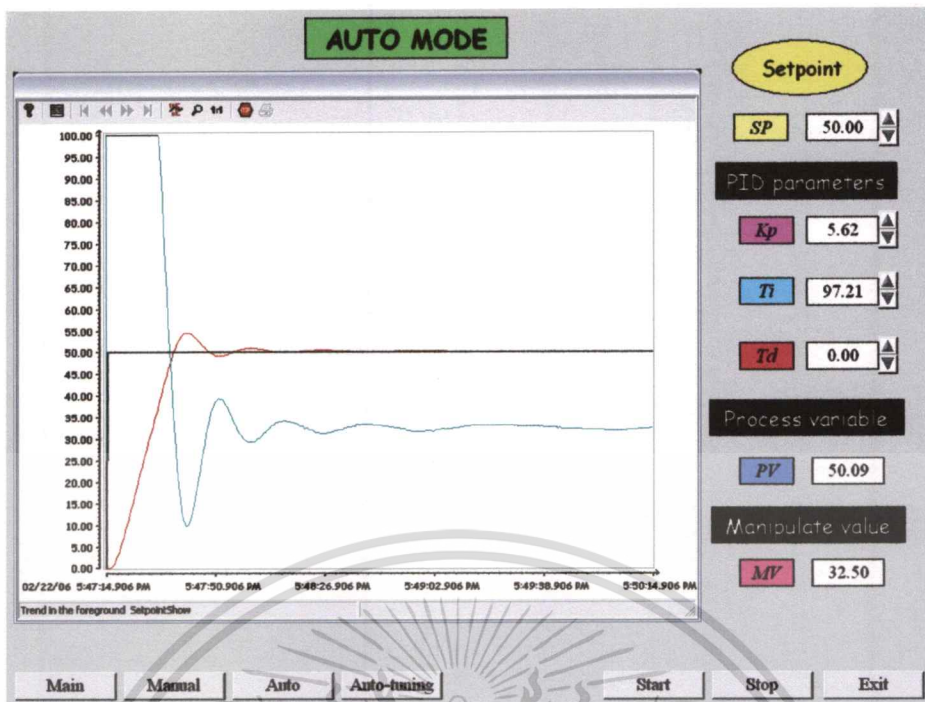


รูปที่ 5.14 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% และ Step change ไปที่ 50% ด้วยการควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.)

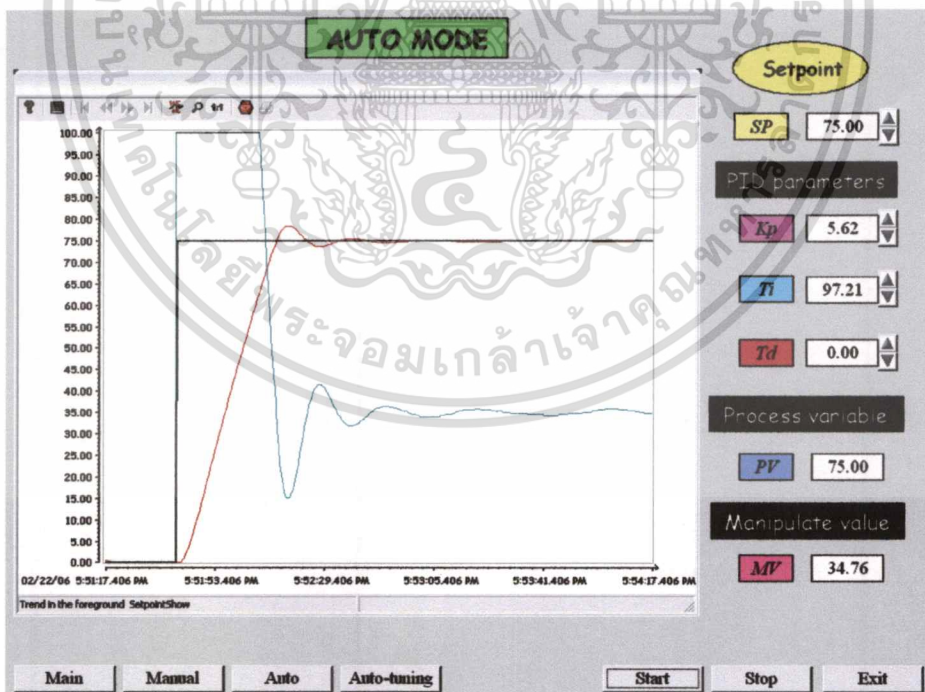


รูปที่ 5.15 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% และ Step change ไปที่ 50% และ Step change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



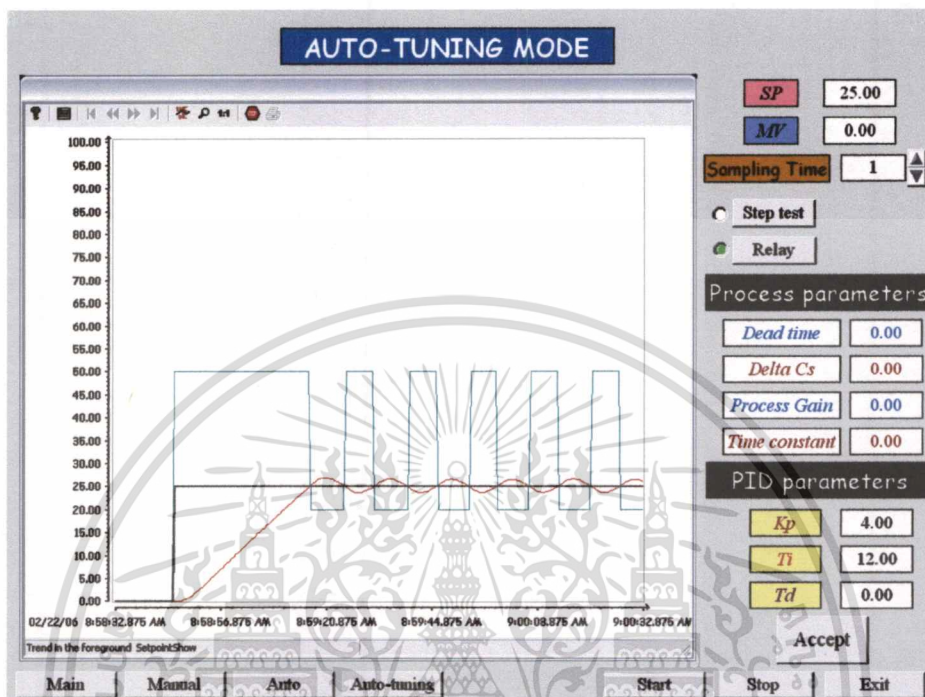
รูปที่ 5.16 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% ด้วยการควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.)



รูปที่ 5.17 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 75% ด้วยการควบคุมแบบพีไอ ($K_p = 5.62$ และ $T_i = 97.21$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาค่าพารามิเตอร์ของกระบวนการควบคุมระดับน้ำจากการทดสอบกระบวนการในโหมด Relay Feedback จากผลตอบสนองของกระบวนการของการทดลองควบคุมระดับน้ำที่ค่าเป้าหมาย 25% แสดงดังรูปที่ 5.18

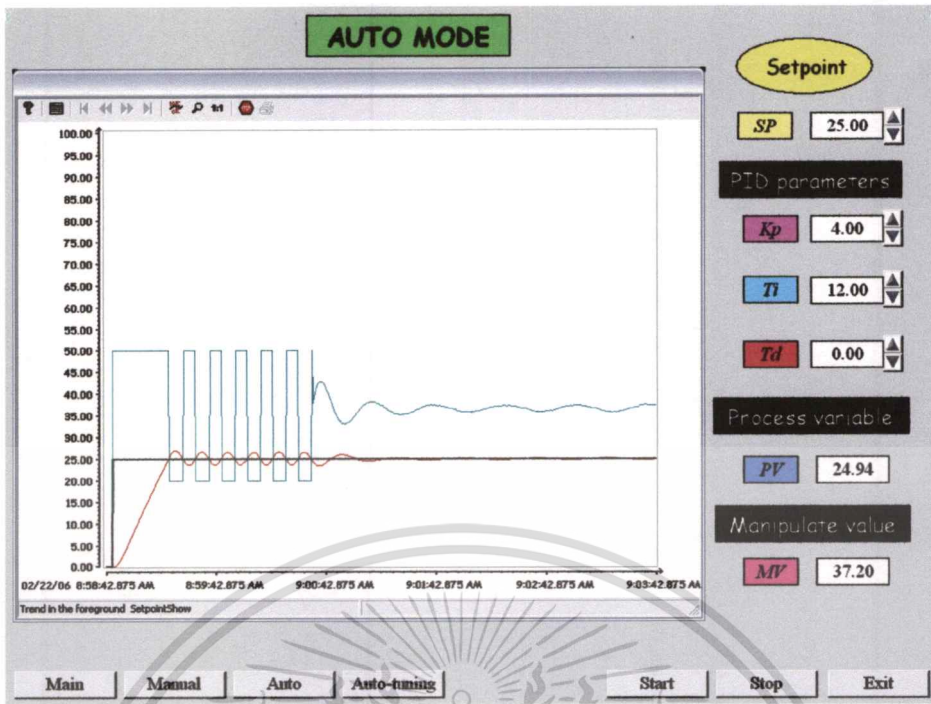


รูปที่ 5.18 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำจากการทดสอบแบบ Relay Feedback ที่ค่าเป้าหมาย 25%

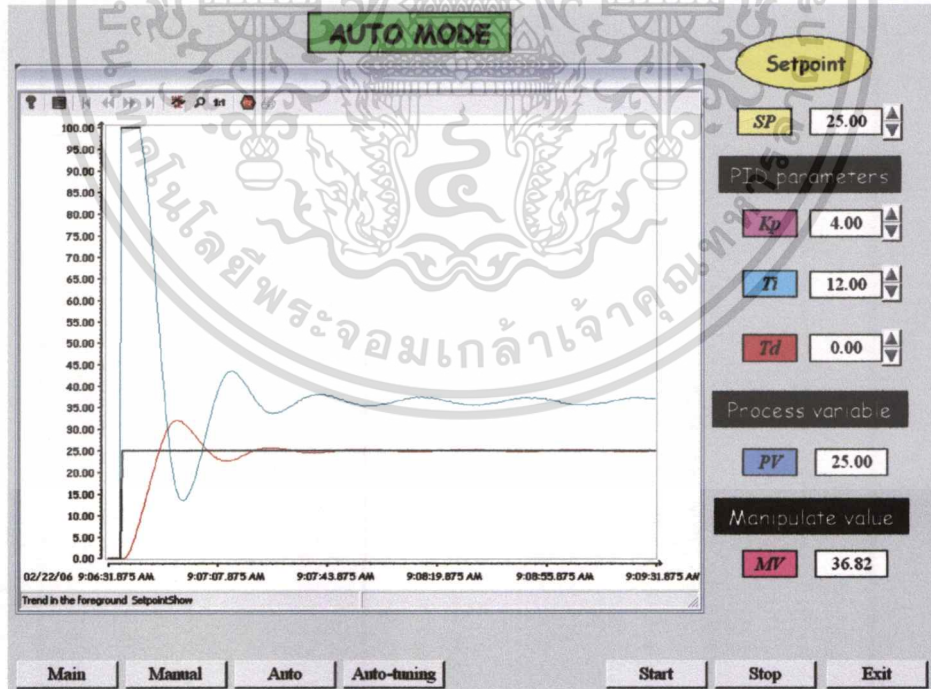
จากการทดสอบกระบวนการควบคุมระดับน้ำแบบ Relay Feedback ค่าพารามิเตอร์ของตัวควบคุมพีไอดีที่คำนวณได้ (กรณีนี้เลือกใช้ตัวควบคุมพีไอ) ดังนี้

$$K_p = 4.00 \text{ และ } T_i = 12.00 \text{ sec.}$$

นำค่าพารามิเตอร์ที่ได้นี้ไปควบคุมกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% แสดงดังรูปที่ 5.19 และ 5.20

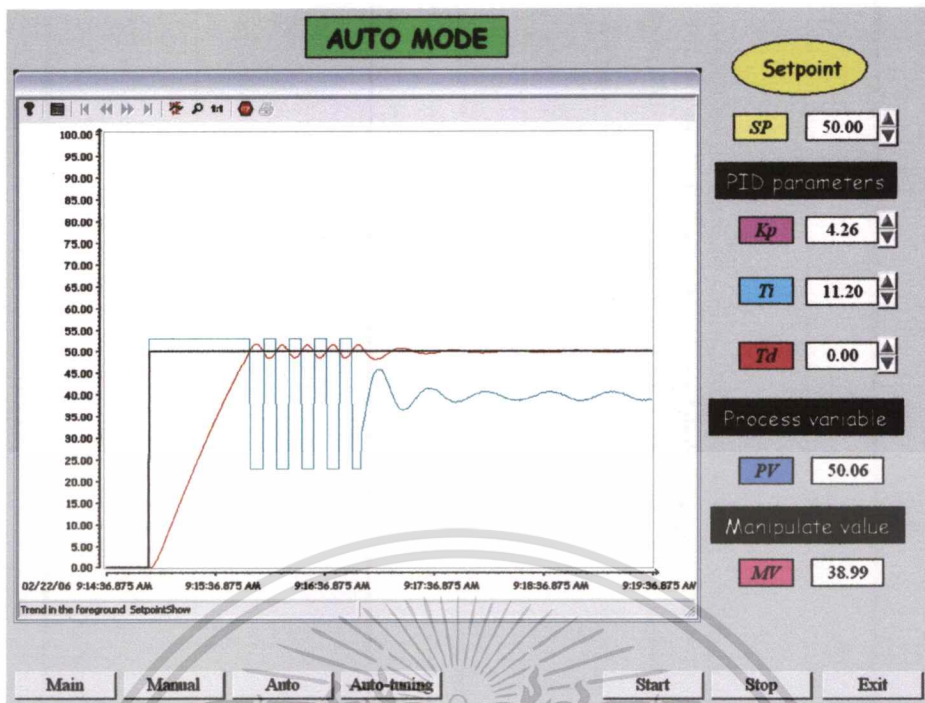


รูปที่ 5.19 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 4.00$ และ $T_i = 12.00$ sec.)

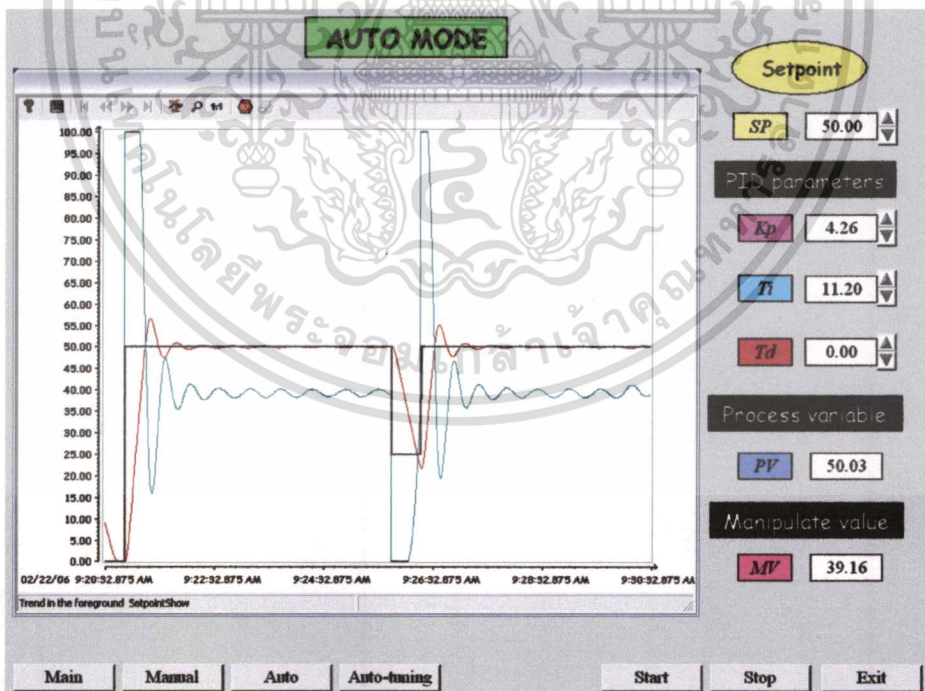


รูปที่ 5.20 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 25% ด้วยการควบคุมแบบพีไอ ($K_p = 4.00$ และ $T_i = 12.00$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



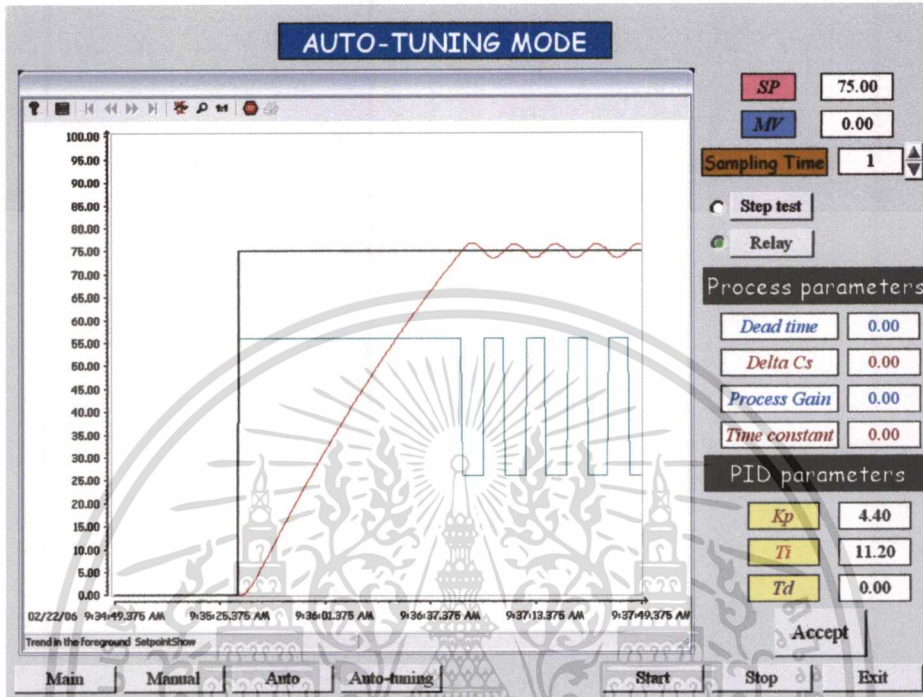
รูปที่ 5.22 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 4.26$ และ $T_i = 11.20$ sec.)



รูปที่ 5.23 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 50% ด้วยการควบคุมแบบพีไอ ($K_p = 4.26$ และ $T_i = 11.20$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาค่าพารามิเตอร์ของกระบวนการควบคุมระดับน้ำจากการทดสอบกระบวนการใน โหมด Relay Feedback จากผลตอบสนองของกระบวนการของการทดลองควบคุมระดับน้ำที่ค่า เป้าหมาย 75% แสดงดังรูปที่ 5.24

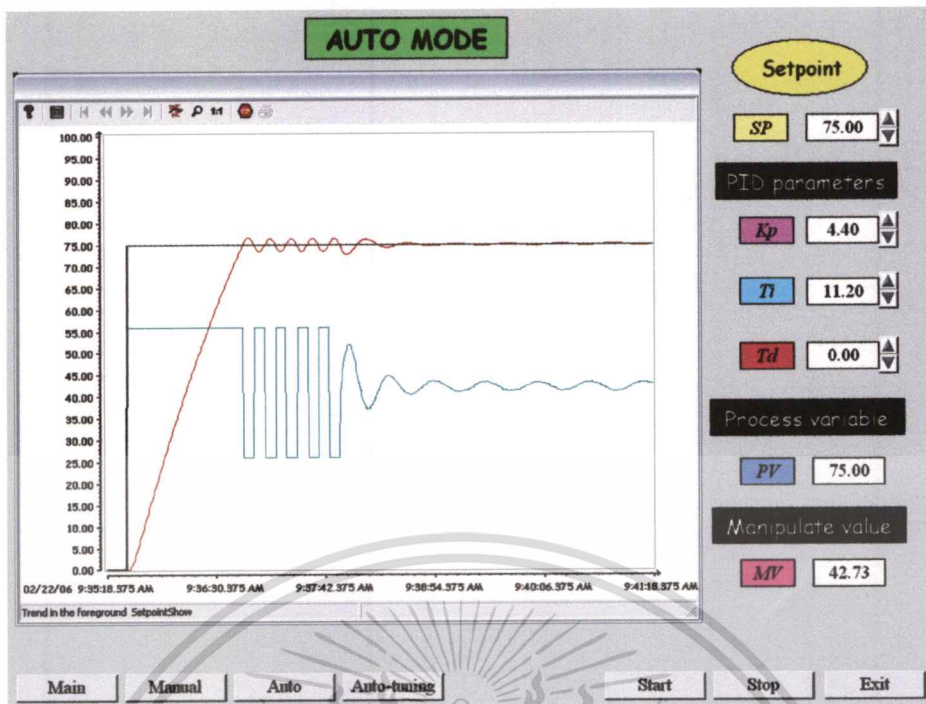


รูปที่ 5.24 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำจากการทดสอบแบบ Relay Feedback ที่ค่าเป้าหมาย 75%

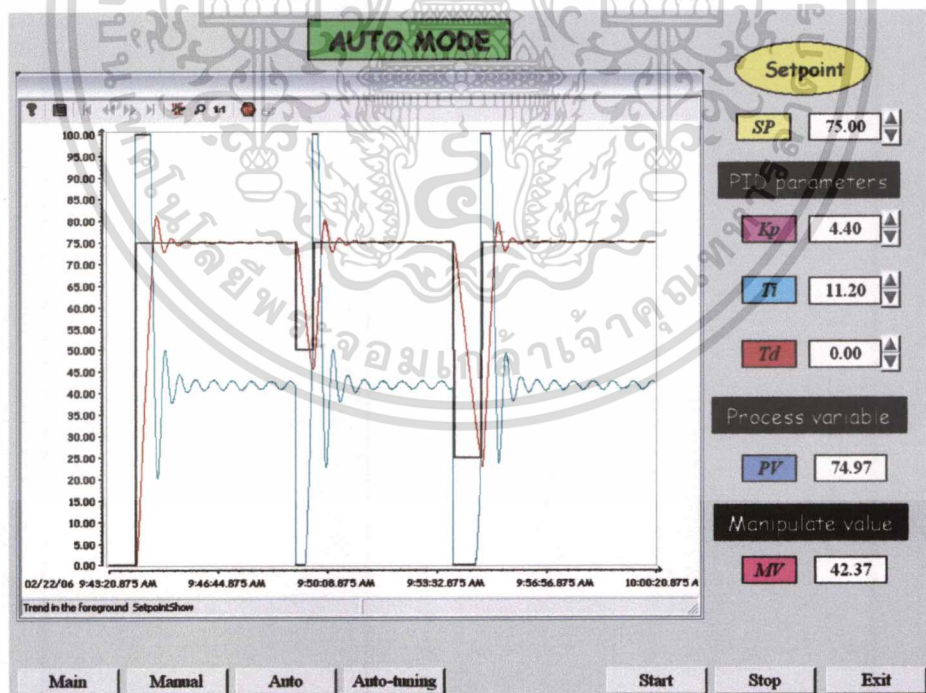
จากการทดสอบกระบวนการควบคุมระดับน้ำแบบ Relay Feedback ค่าพารามิเตอร์ของตัวควบคุมพีไอดีที่คำนวณได้ (กรณีนี้เลือกใช้ตัวควบคุมพีไอ) ดังนี้

$$K_p = 4.40 \text{ และ } T_i = 11.20 \text{ sec.}$$

นำค่าพารามิเตอร์ที่ได้นี้ไปควบคุมกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 75% แสดง ดังรูปที่ 5.25 และ 5.26



รูปที่ 5.25 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 75% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 4.40$ และ $T_i = 11.20$ sec.)

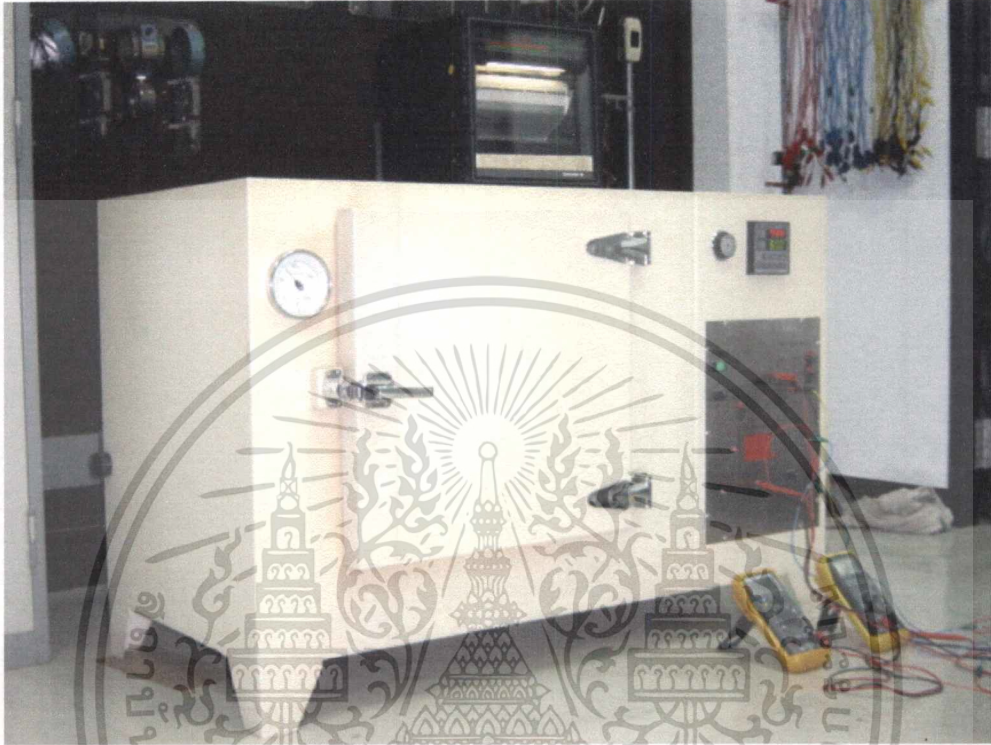


รูปที่ 5.26 แสดงผลตอบสนองของกระบวนการควบคุมระดับน้ำที่ค่าเป้าหมาย 75% ด้วยการควบคุมแบบพีไอ ($K_p = 4.40$ และ $T_i = 11.20$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

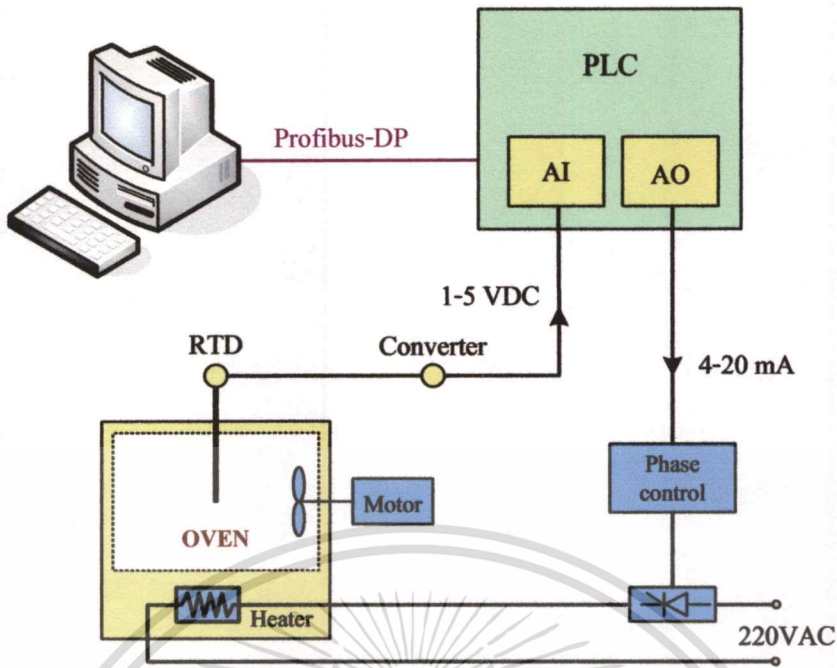
5.3 ผลการทดลองกับกระบวนการควบคุมอุณหภูมิ

การทดลองนี้เป็นการประยุกต์ใช้ตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC กับกระบวนการควบคุมอุณหภูมิของตู้อบ ดังรูปที่ 5.27



รูปที่ 5.27 แสดงการประยุกต์ใช้ตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC กับกระบวนการจำลองการควบคุมอุณหภูมิของตู้อบ

จากรูปที่ 5.27 นำมาเขียนเป็นโครงสร้างการทดลองกับกระบวนการควบคุมอุณหภูมิของตู้อบให้ชัดเจนได้ดังรูปที่ 5.28

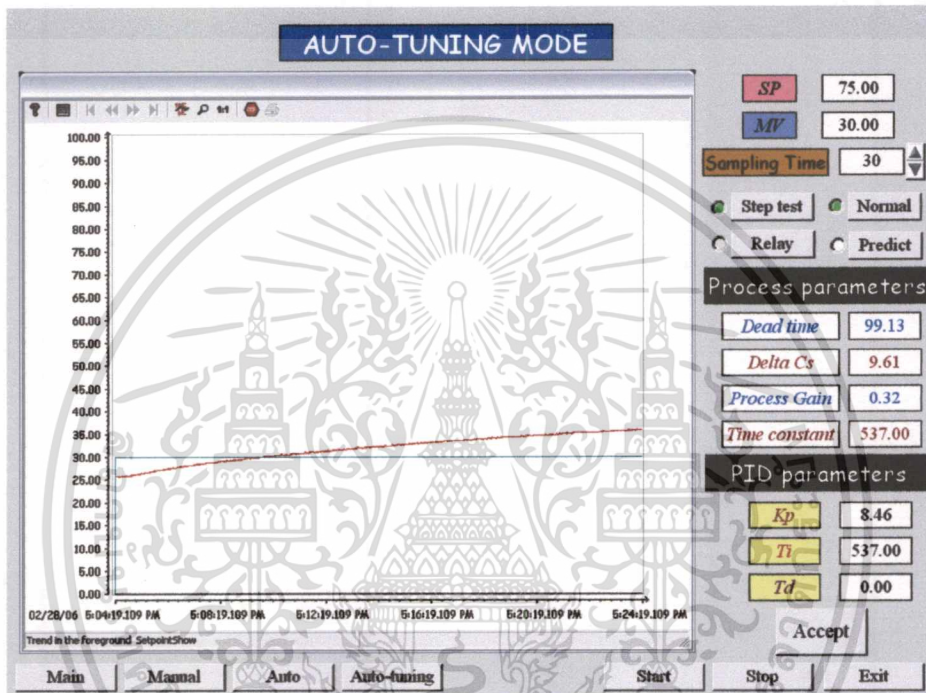


รูปที่ 5.28 แสดงโครงสร้างการทดลองกับกระบวนการควบคุมอุณหภูมิของตู้อบ

การทำงานของ การทดลอง เมื่อกดสวิทช์ ON พัดลมจะทำงานเพื่อกระจายความร้อนภายในตู้อบและต่อให้ขดลวดความร้อน (Electrical Heater) ทำงานครบวงจร โดยขดลวดความร้อนจะรับระดับแรงดันไฟฟ้ากระแสสลับ (0-220 VAC) เพื่อกำเนิดความร้อนจากชุดควบคุมแรงดันไฟฟ้า (Phase control) ซึ่งจะทำงานเมื่อได้รับอินพุตที่เป็นกระแสไฟฟ้า (4-20 mADC) ส่วนอุณหภูมิในตู้อบจะตรวจวัดโดยอาร์ทีดี (Resistance Temperature Detector: RTD) และส่งสัญญาณออกไปในรูปของแรงดันไฟฟ้า (1-5 VDC) ผ่านตัวควบคุมซึ่งระดับอุณหภูมิภายในตู้อบจะถูกป้องกันไม่ให้สูงเกินกว่าระดับอุณหภูมิสูงสุดที่ตั้งไว้โดยตัวป้องกันอุณหภูมิเกิน (Overload Temperature Protection) เริ่มแรกถ้าเครื่องควบคุม PLC ไม่ทำงาน ขดลวดก็ยังไม่เกิดความร้อนขึ้น เนื่องจากไม่มีสัญญาณอินพุตป้อนให้กับชุดควบคุมแรงดันไฟฟ้า ดังนั้นระดับแรงดันไฟฟ้าที่เอาท์พุทของชุดควบคุมแรงดันไฟฟ้าจะเป็นศูนย์ และเมื่อเริ่มทำงานเครื่องควบคุม PLC จะส่งสัญญาณควบคุมออกทางโมดูลเอาท์พุทแบบอนาลอกเป็นสัญญาณกระแสไฟฟ้าแบบขั้วมันไบโค ขนาด Δm (4-20 mADC) จ่ายให้กับชุดควบคุมแรงดันไฟฟ้าทำให้เกิดแรงดันไฟฟ้าค่าหนึ่งป้อนให้กับขดลวดความร้อน อุณหภูมิภายในตู้อบก็จะค่อยๆ สูงขึ้นจากนั้นอุณหภูมิภายในตู้อบจะถูกวัดโดย RTD และแปลงเป็นสัญญาณแรงดันไฟฟ้าขนาด 1-5 VDC และถูกป้อนเป็นสัญญาณอินพุตให้กับเครื่องควบคุม PLC เข้าทางโมดูลอินพุทแบบอนาลอกเพื่อทำการเก็บข้อมูลที่ี้ได้จากการสุ่มเก็บค่าผลตอบแทนของกระบวนการและเครื่องควบคุม PLC ก็จะเริ่มทำการจับเวลาตั้งแต่เริ่มต้นการทำงานและนำข้อมูลที่

ได้จากการสุ่มไปใช้ในการคำนวณค่าพารามิเตอร์ต่างๆ ตามหลักการข้างต้นต่อไป ซึ่งจะแสดงผลที่ได้ต่างๆ ทางหน้าจอคอมพิวเตอร์

การหาค่าพารามิเตอร์ของกระบวนการจากการทดสอบกระบวนการในโหมด Step Test แบบ Normal จากผลตอบสนองของกระบวนการของการทดลองควบคุมอุณหภูมิ แสดงดังรูปที่ 5.29



รูปที่ 5.29 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิจากการทดสอบโดยป้อนสัญญาณทดสอบขนาด $\Delta m = 30\%$ ในโหมด Step-Test แบบ Normal

จากการทดสอบกระบวนการที่ค่าสัญญาณทดสอบ ได้ค่าพารามิเตอร์ของกระบวนการ ดังนี้

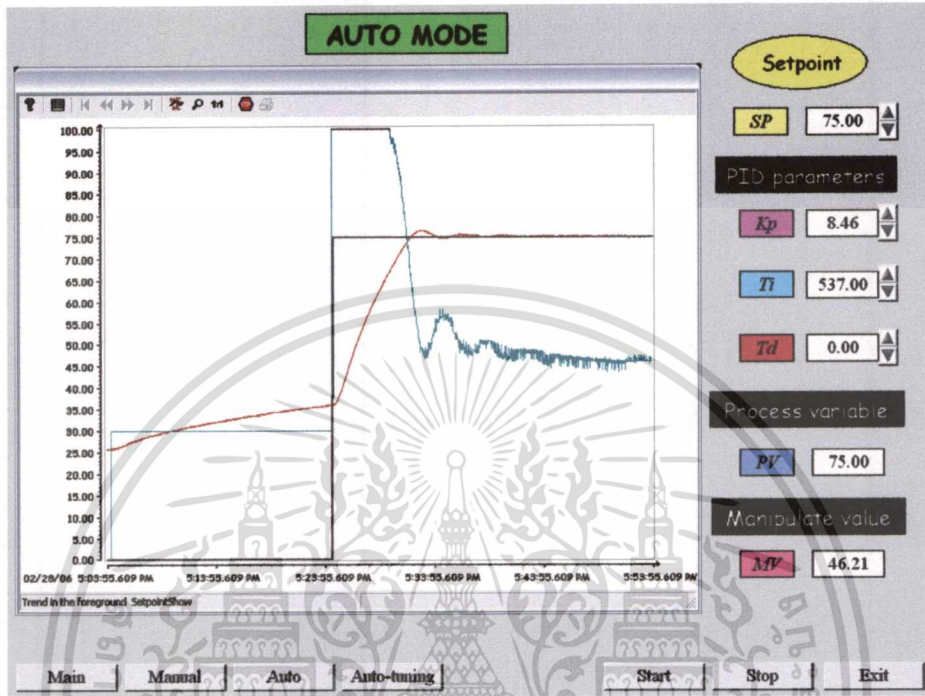
$$K = 0.32, t_0 = 99.13 \text{ sec. และ } \tau = 537.00 \text{ sec.}$$

และได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ดังนี้

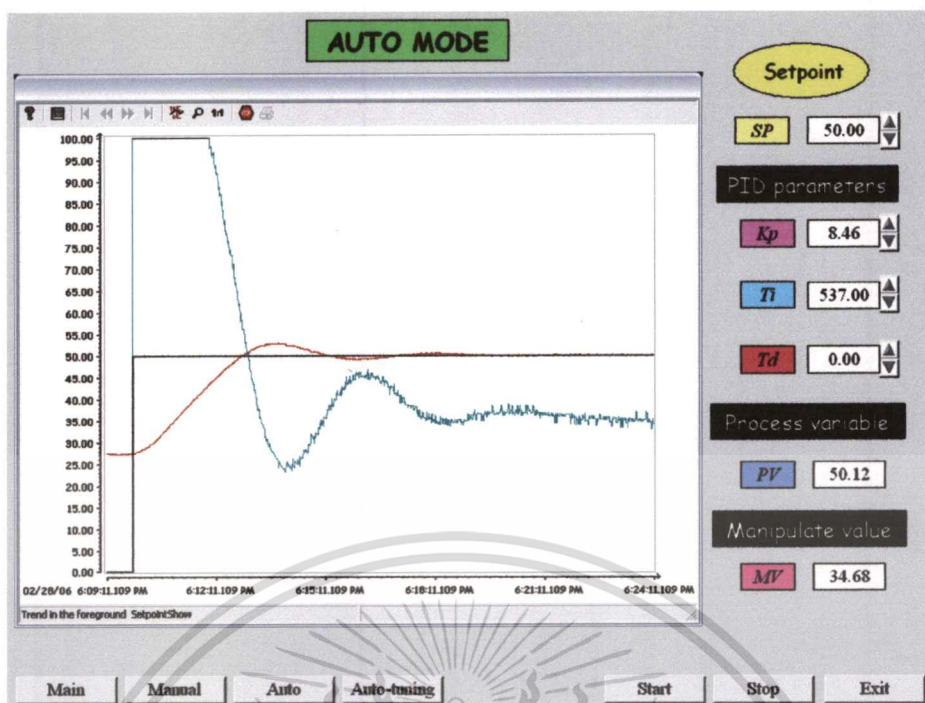
$$K_p = 8.46 \text{ และ } T_i = 537.00 \text{ sec.}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

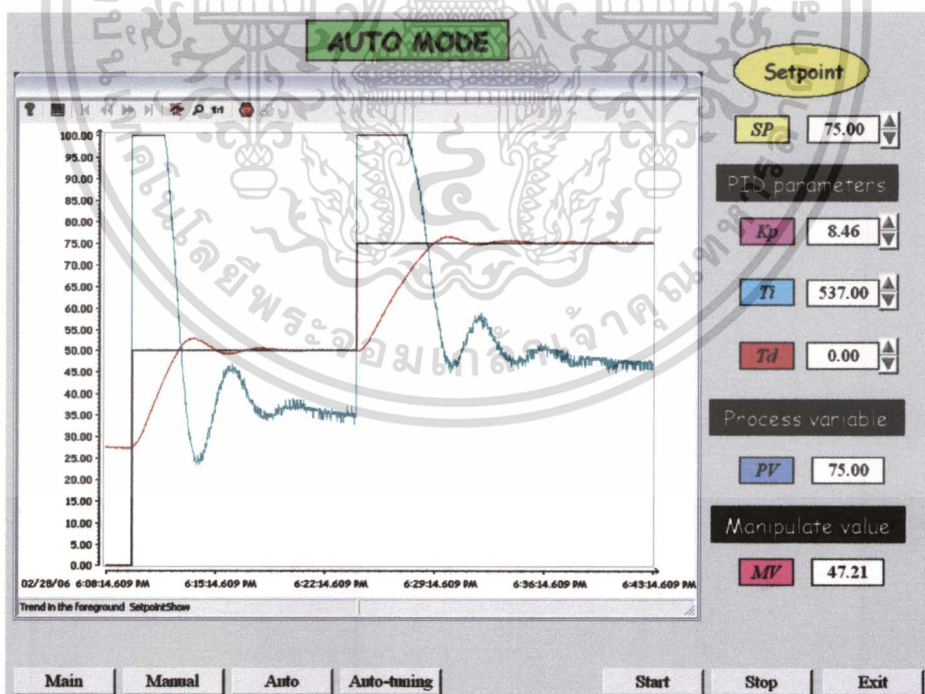
จากการทดลองค่าพารามิเตอร์ของตัวควบคุมที่ได้จากโหมด Step Test แบบ Normal คือ $K_p = 8.46$ และ $T_i = 537.00$ วินาที นำไปควบคุมกระบวนการควบคุมอุณหภูมิแบบอัตโนมัติที่ค่าเป้าหมายต่างๆ แสดงดังรูปที่ 5.30 ถึง 5.32



รูปที่ 5.30 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 8.46$ และ $T_i = 537.00$ sec.)

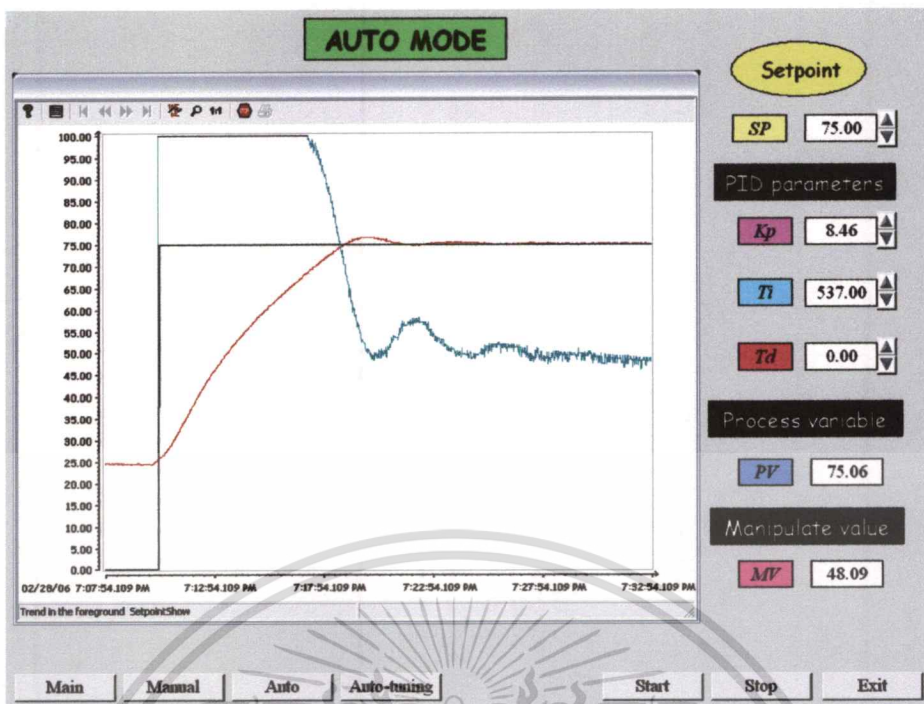


รูปที่ 5.31 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% ด้วยการควบคุมแบบพีไอ ($K_p = 8.46$ และ $T_i = 537.00$ sec.)



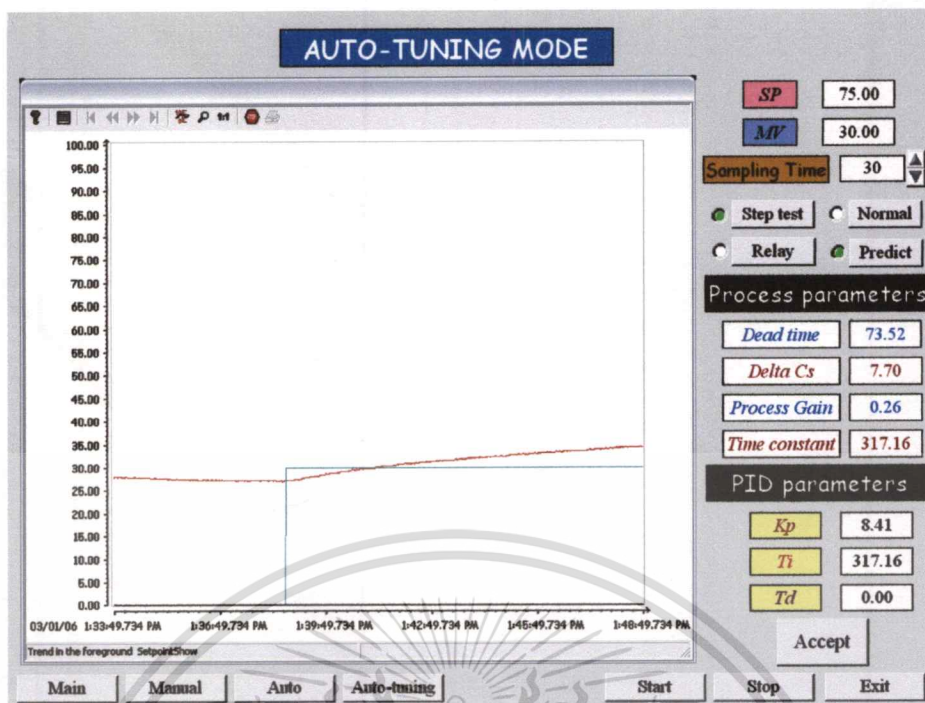
รูปที่ 5.32 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% และ Step change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 8.46$ และ $T_i = 537.00$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.33 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% ด้วยการควบคุมแบบพีไอ ($K_p = 8.46$ และ $T_i = 537.00$ sec.)

การหาค่าพารามิเตอร์ของกระบวนการจากการทดสอบกระบวนการควบคุมอุณหภูมิในโหมด Step Test แบบ Predict จากผลตอบสนองของกระบวนการของการทดลองควบคุมอุณหภูมิแสดงดังรูปที่ 5.34



รูปที่ 5.34 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิจากการทดสอบโดยป้อนสัญญาณทดสอบขนาด $\Delta m = 30\%$ ในโหมด Step Test แบบ Predict

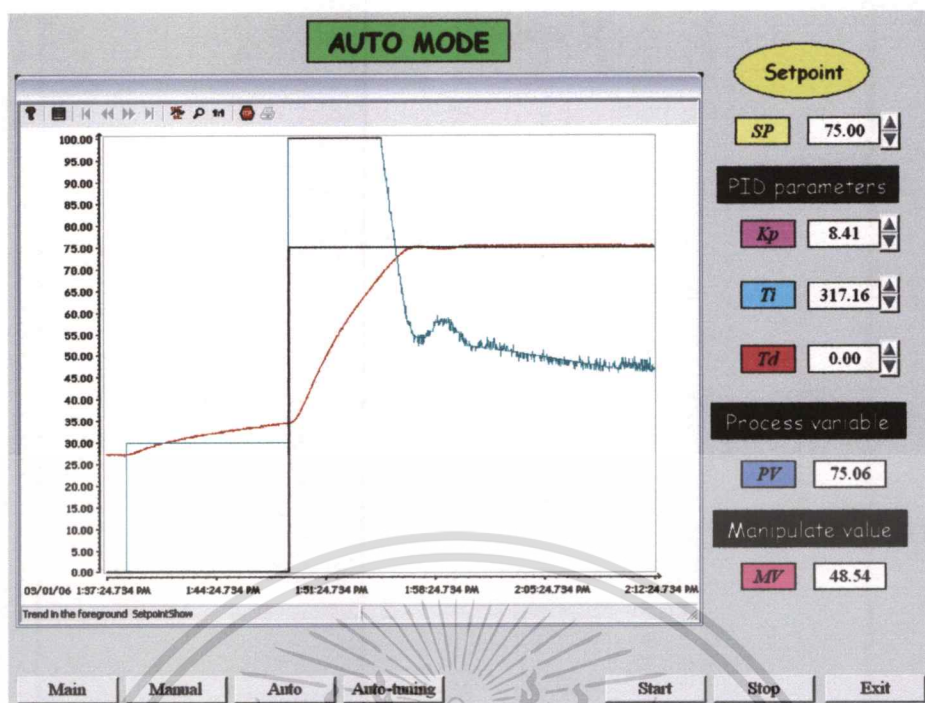
จากการทดสอบกระบวนการที่ค่าสัญญาณทดสอบ ได้ค่าพารามิเตอร์ของกระบวนการ ดังนี้

$$K = 0.26, t_0 = 73.52 \text{ sec. และ } \tau = 317.16 \text{ sec.}$$

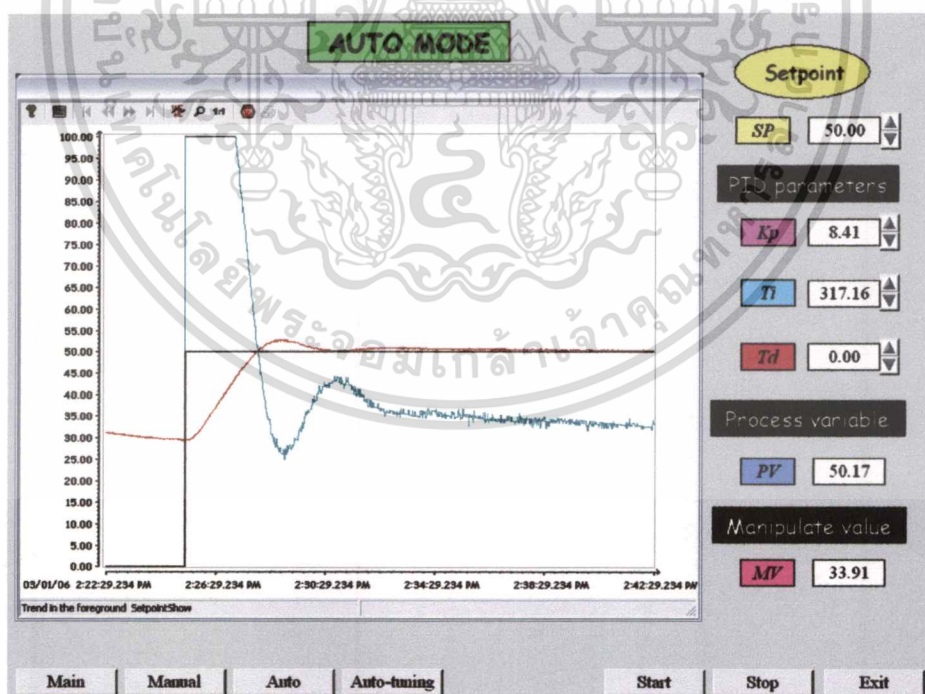
และได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ดังนี้

$$K_p = 8.41 \text{ และ } T_i = 317.16 \text{ sec.}$$

จากการทดลองค่าพารามิเตอร์ของตัวควบคุมที่ได้จากโหมด Step Test แบบ Predict คือ $K_p = 8.41$ และ $T_i = 317.16$ วินาที นำไปควบคุมกระบวนการควบคุมอุณหภูมิแบบอัตโนมัติที่ค่าเป้าหมายต่างๆ แสดงดังรูปที่ 5.35 ถึง 5.38

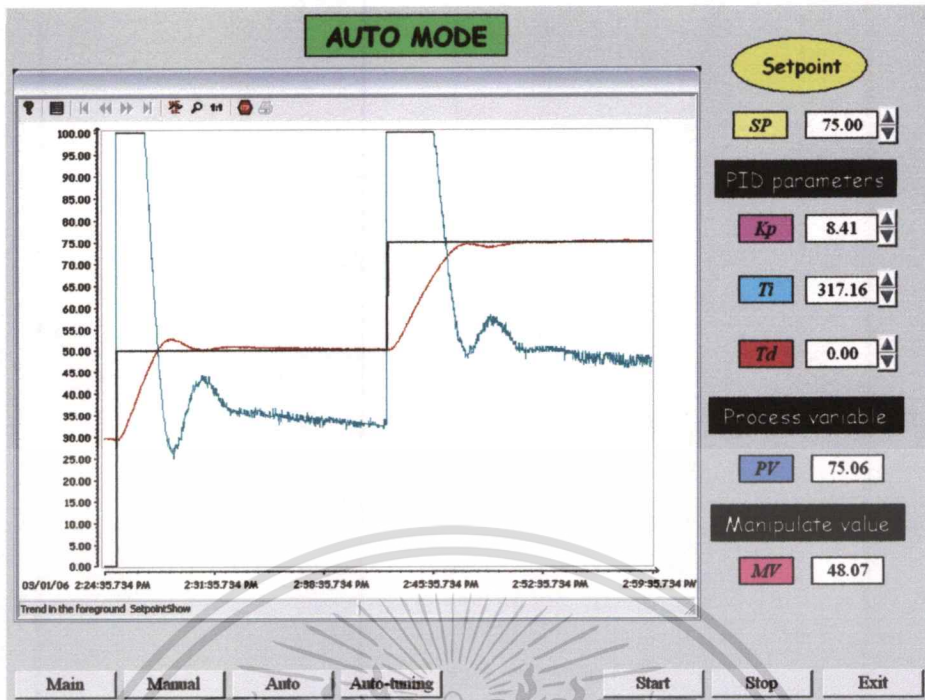


รูปที่ 5.35 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 8.41$ และ $T_i = 317.16$ sec.)

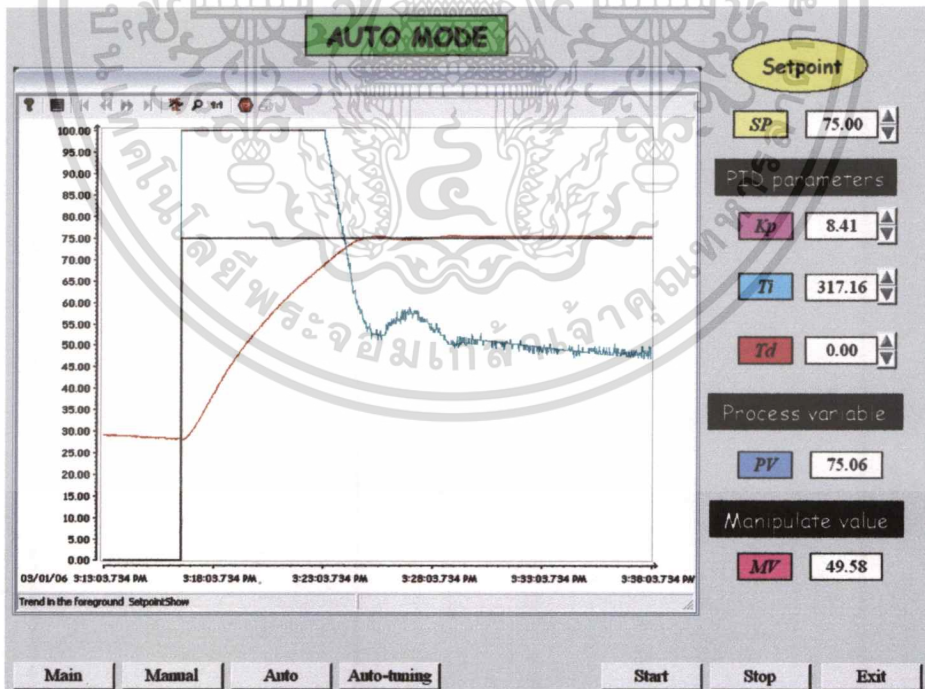


รูปที่ 5.36 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% ด้วยการควบคุมแบบพีไอ ($K_p = 8.41$ และ $T_i = 317.16$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



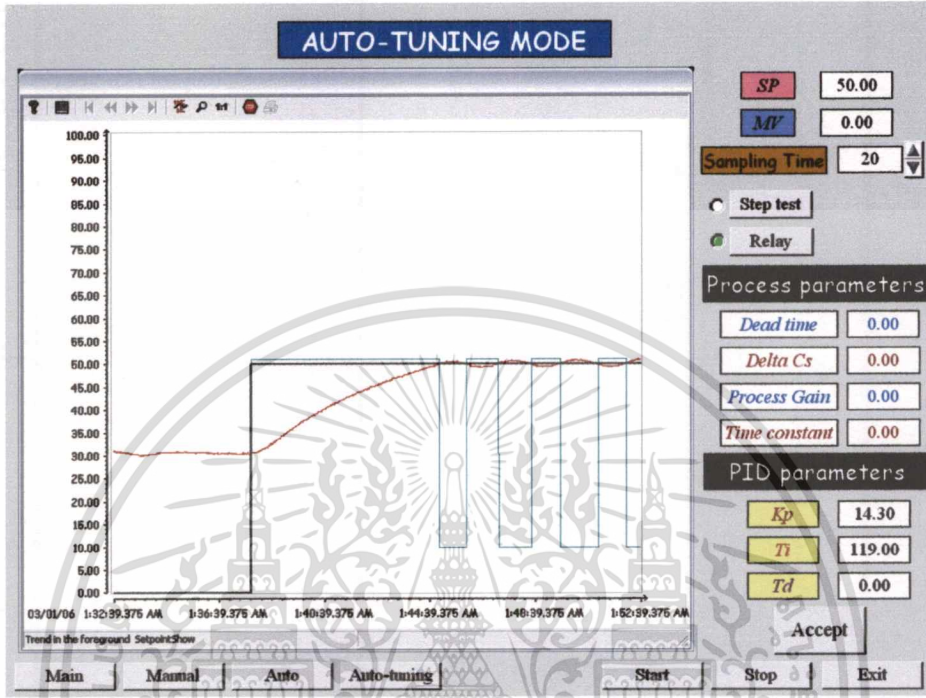
รูปที่ 5.37 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% และ Step change ไปที่ 75% ด้วยการควบคุมแบบพีไอ ($K_p = 8.41$ และ $T_i = 317.16$ sec.)



รูปที่ 5.38 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% ด้วยการควบคุมแบบพีไอ ($K_p = 8.41$ และ $T_i = 317.16$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาค่าพารามิเตอร์ของกระบวนการควบคุมอุณหภูมิจากการทดสอบกระบวนการในโหมด Relay Feedback จากผลตอบสนองของกระบวนการของการทดลองควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% แสดงดังรูปที่ 5.39

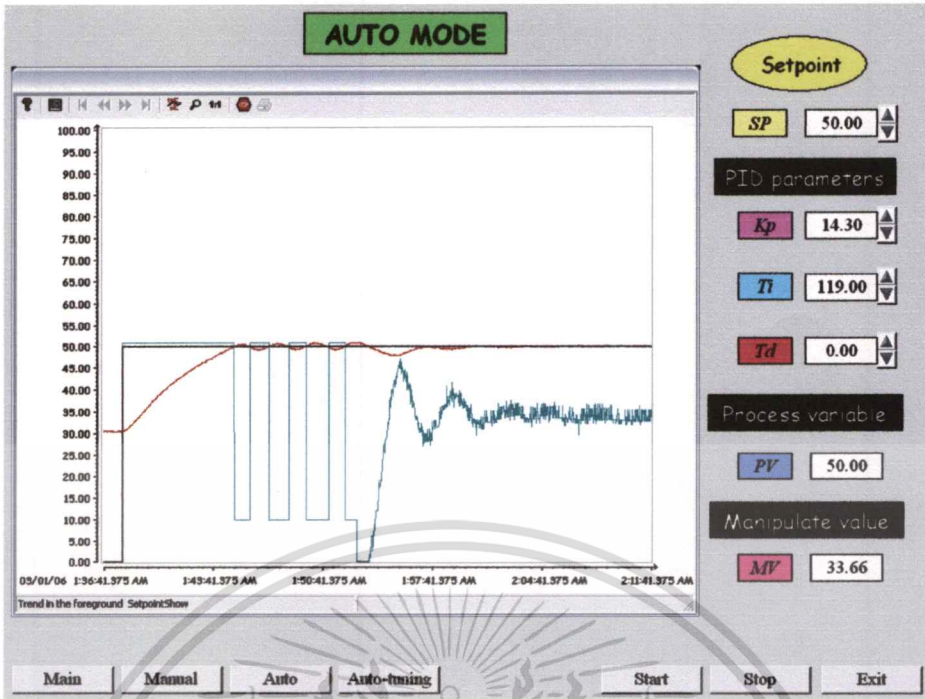


รูปที่ 5.39 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิจากการทดสอบแบบ Relay Feedback ที่ค่าเป้าหมาย 50%

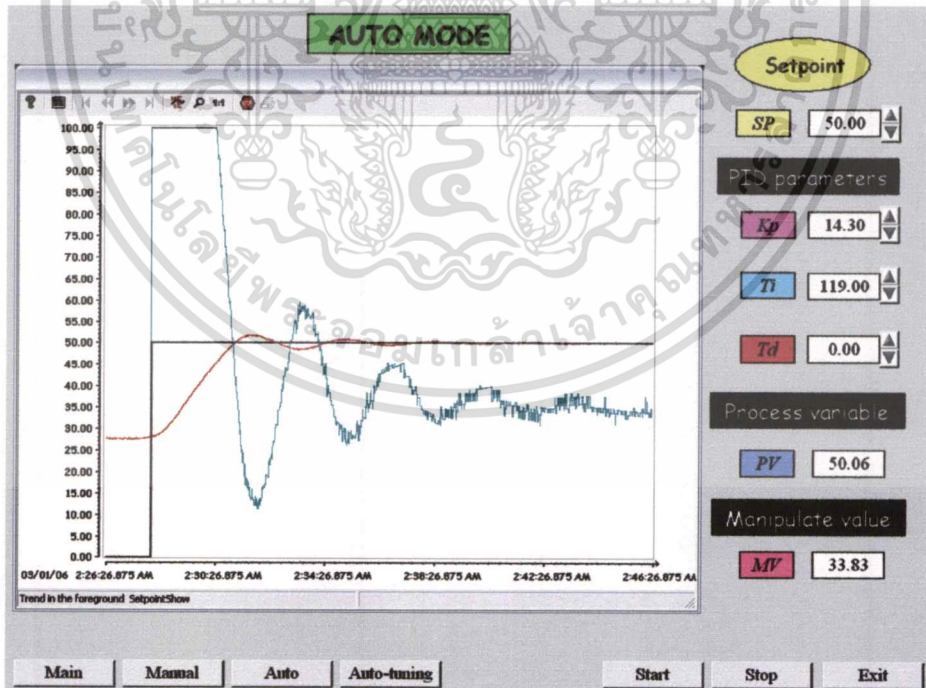
จากการทดสอบกระบวนการควบคุมอุณหภูมิแบบ Relay Feedback ค่าพารามิเตอร์ของตัวควบคุมพีไอที่ดีที่สุดที่คำนวณได้ (กรณีนี้เลือกใช้ตัวควบคุมพีไอ) ดังนี้

$$K_p = 14.30 \text{ และ } T_i = 119.00 \text{ sec.}$$

นำค่าพารามิเตอร์ที่ได้นี้ไปควบคุมกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% แสดงดังรูปที่ 5.40 และ 5.41



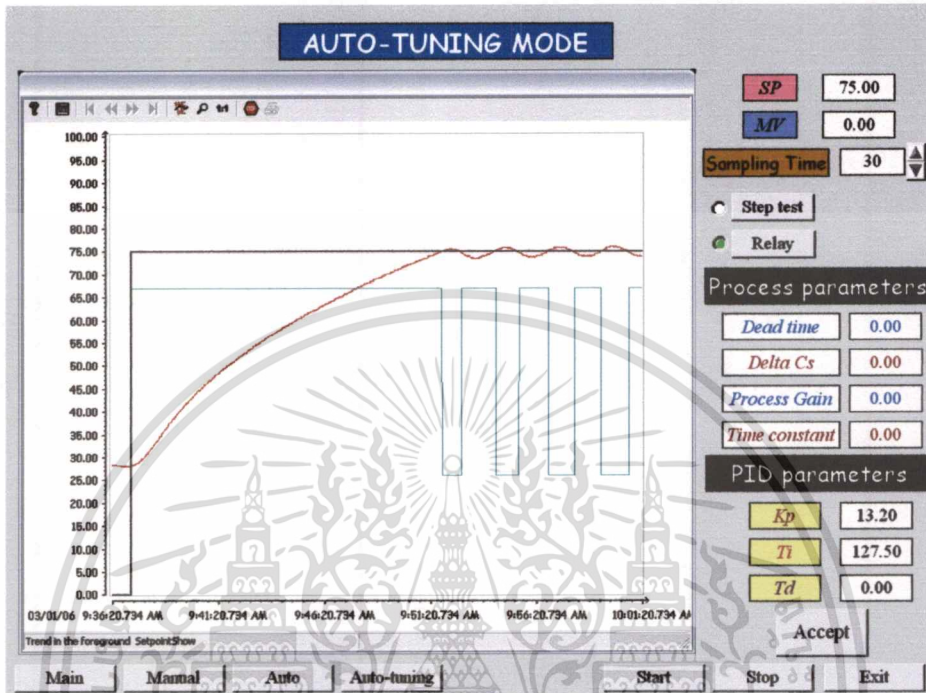
รูปที่ 5.40 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 14.30$ และ $T_i = 119.00$ sec.)



รูปที่ 5.41 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 50% ด้วยการควบคุมแบบพีไอ ($K_p = 14.30$ และ $T_i = 119.00$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาค่าพารามิเตอร์ของกระบวนการควบคุมอุณหภูมิจากการทดสอบกระบวนการในโหมด Relay Feedback จากผลตอบสนองของกระบวนการของการทดลองควบคุมอุณหภูมิ ที่ค่าเป้าหมาย 75% แสดงดังรูปที่ 5.42

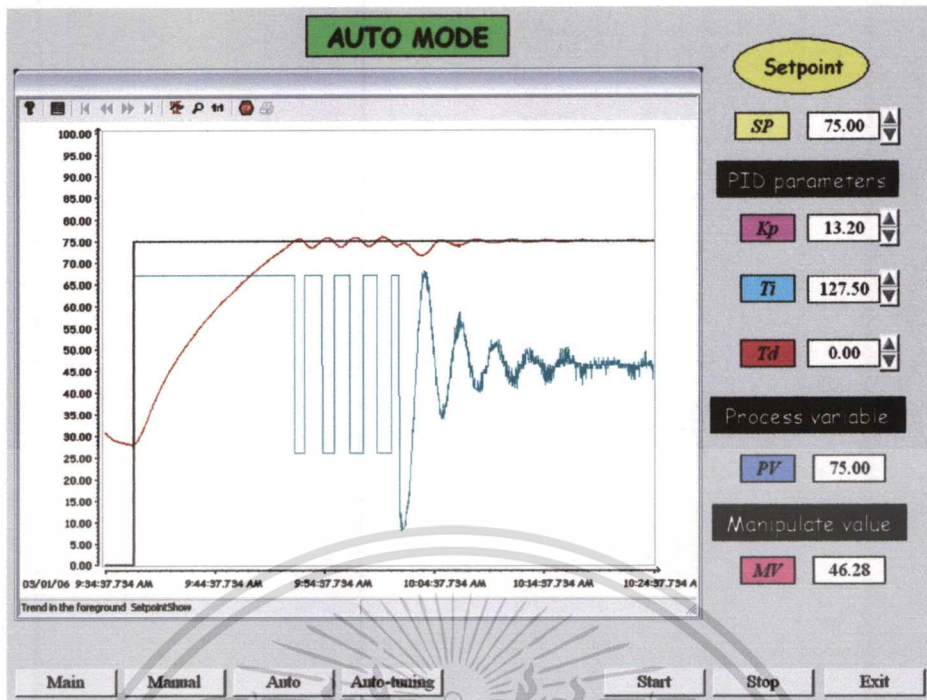


รูปที่ 5.42 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิจากการทดสอบแบบ Relay Feedback ที่ค่าเป้าหมาย 75%

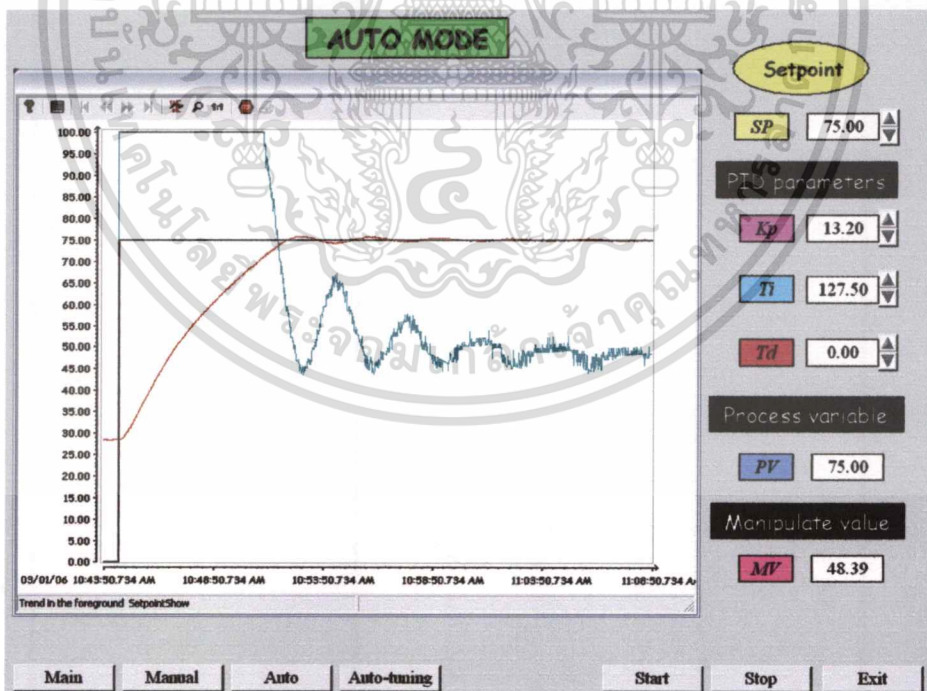
จากการทดสอบกระบวนการควบคุมอุณหภูมิแบบ Relay Feedback ค่าพารามิเตอร์ของตัวควบคุมพีไอดีที่คำนวณได้ (กรณีนี้เลือกใช้ตัวควบคุมพีไอ) ดังนี้

$$K_p = 13.20 \text{ และ } T_i = 127.50 \text{ sec.}$$

นำค่าพารามิเตอร์ที่ได้นี้ไปควบคุมกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% แสดงดังรูปที่ 5.43 และ 5.44



รูปที่ 5.43 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% หลังจากได้ค่าพารามิเตอร์ของตัวควบคุมแบบพีไอ ($K_p = 13.20$ และ $T_i = 127.50$ sec.)



รูปที่ 5.44 แสดงผลตอบสนองของกระบวนการควบคุมอุณหภูมิที่ค่าเป้าหมาย 75% ด้วยการควบคุมแบบพีไอ ($K_p = 13.20$ และ $T_i = 127.50$ sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 สรุปผลการทดลอง

จากการทดลองที่ประยุกต์นำตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC นำไปประยุกต์ใช้กับกระบวนการควบคุมระดับน้ำของน้ำในถังและกระบวนการควบคุมอุณหภูมิของตู้อบพบว่าตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC สามารถตั้งคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีและสามารถควบคุมกระบวนการได้อย่างมีประสิทธิภาพ สอดคล้องกับทฤษฎีตามหลักการของ Dahlin กล่าวคือ เกิดค่าพุ่งเกินสูงสุดประมาณ 5% เมื่อเปลี่ยนแปลงค่าเป้าหมาย การประยุกต์ใช้การประมาณค่าล่วงหน้าช่วยให้การหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีตามหลักการของคาล์ลิตินสามารถทำได้รวดเร็วขึ้นและให้ผลของค่าพารามิเตอร์ของตัวควบคุมพีไอดีมีค่าใกล้เคียงกัน การหาค่าพารามิเตอร์ของตัวควบคุมพีไอดีทั้งสองวิธีใช้หลักการที่แตกต่างกัน ดังนั้นจึงผลตอบสนองของกระบวนการที่แตกต่างกัน อย่างไรก็ตามค่าพารามิเตอร์ของตัวควบคุมพีไอดีที่คำนวณได้นี้เป็นค่าโดยประมาณซึ่งจะให้ผลตอบสนองที่ดีในระดับหนึ่งเท่านั้น ถ้าต้องการให้ผลตอบสนองของกระบวนการที่ดีมากขึ้นกว่านี้อาจต้องทำการปรับละเอียดอีกครั้ง



บทที่ 6

สรุปและข้อเสนอแนะ

6.1 สรุป

วิทยานิพนธ์ฉบับนี้เป็นการประยุกต์ใช้เครื่องควบคุม PLC ทำหน้าที่วิเคราะห์และสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีแบบอัตโนมัติ โดยเครื่องควบคุม PLC ทำการควบคุมกระบวนการ การวิเคราะห์และสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอดี โดยกระบวนการจะถูกอนุมานให้เป็นกระบวนการอันดับหนึ่งที่มีการหน่วงเวลา (FOPDT) โดยการทดสอบกระบวนการโดยป้อนสัญญาณแบบขั้นบันไดให้กับกระบวนการแบบลูเปิด ตามหลักการของ Dahlin หลังจากที่ได้ค่าพารามิเตอร์ของตัวควบคุมพีไอดีแล้วการควบคุมกระบวนการจะเป็นแบบอัตโนมัติทันที จากการสังเคราะห์หาค่าพารามิเตอร์ของกระบวนการบางกระบวนการอาจใช้ระยะเวลาานานมาก วิทยานิพนธ์ฉบับนี้จึงใช้เทคนิคการประมาณค่าล่วงหน้าจากข้อมูลที่ได้จากการสุ่มเพื่อช่วยลดระยะเวลาในการทดสอบกระบวนการให้สั้นลง

จากการทดลองนำตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC ไปทำการควบคุมกระบวนการควบคุมระดับน้ำและกระบวนการควบคุมอุณหภูมิของตู้อบ แสดงให้เห็นว่าตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC สามารถสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีและสามารถนำไปใช้ควบคุมกระบวนการได้อย่างมีประสิทธิภาพ สอดคล้องกับทฤษฎีตามหลักการของ Dahlin กล่าวคือ เกิดค่าพุ่งเกินสูงสุดประมาณ 5% เมื่อเปลี่ยนแปลงค่าเป้าหมาย อย่างไรก็ตามค่าพารามิเตอร์ของตัวควบคุมพีไอดีนี้เป็นการประมาณค่าที่ให้ผลตอบสนองของกระบวนการที่ดีในระดับหนึ่งเท่านั้น ดังนั้นในการใช้งานตัวควบคุมพีไอดีแบบปรับค่าพารามิเตอร์อัตโนมัติบนเครื่องควบคุม PLC ในการควบคุมกระบวนการนั้นควรจะมีการปรับละเอียดอีกครั้งเพื่อให้ได้ผลตอบสนองที่ดีตามความต้องการ โดยใช้ค่าพารามิเตอร์ที่คำนวณได้เป็นแนวทางซึ่งจะทำให้ลดระยะเวลาในการปรับละเอียดลงได้

6.2 ข้อเสนอแนะ

ข้อเสนอแนะและการพัฒนาสำหรับวิทยานิพนธ์ฉบับนี้ที่ใช้พัฒนาเครื่องควบคุม PLC ให้สามารถวิเคราะห์และสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุมพีไอดีแบบอัตโนมัติ เพื่อใช้ในการควบคุมกระบวนการต่าง ๆ ดังที่แสดงผลการทดลองให้เห็นในบทที่ 5 นั้น ยังสามารถนำไปประยุกต์ใช้กับกระบวนการอื่นๆ ได้ เช่น การควบคุมความดัน เป็นต้น และสามารถใช้เป็นแนวทางในการปฏิบัติโดยการประยุกต์นำไปใช้กับกระบวนการที่ใช้งานอยู่จริงได้

ข้อจำกัดของการทำวิทยานิพนธ์ฉบับนี้ก็คือ การใช้งานเครื่องควบคุม PLC กับกระบวนการแบบต่างๆ นั้นจำเป็นต้องกำหนดชนิดและประเภทของหน่วยอินพุทเอาต์พุทให้ตรงกับการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นเข้าไปเผยแพร่ขึ้นต้นการค้นคว้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เครื่องควบคุม PLC ประมวลสัญญาณได้อย่างถูกต้อง และข้อจำกัดอีกข้อหนึ่งคือ อุปกรณ์ เซนเซอร์ ทรานสมิตเตอร์ และอุปกรณ์แอคทูเอเตอร์ต่างๆ ก็เป็นสาเหตุหนึ่งที่ทำให้เกิดความผิดพลาดขึ้นในการเก็บข้อมูลที่นำมาวิเคราะห์หาค่าพารามิเตอร์ของกระบวนการซึ่งจะทำให้ค่าพารามิเตอร์ของตัวควบคุมที่โอดีผิดพลาดได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] B-S Kang and K-H Cho, "Design of PLCs for Automated Industrial Systems Based on Discrete Event Models", IEEE ISIE 2001, Pusan, Korea, pp. 1431-1434, 2001.
- [2] T. Krairojananan and S. Suthapradit, "A PLC Program Generator Incorporating Sequential Circuit Synthesis Techniques", Circuit and systems, The 1998 IEEE Asia-Pacific Conference, pp. 399-402, 24-27 Nov. 1998.
- [3] S. Qiang, X.Z. Gao and X. Zhuang, "PLC-based Control System for Industrial Production of Fuel Alcohol", IEEE ICIT'02, International Conference, Bangkok, Thailand, pp.827-832, 11-14 Dec. 2002.
- [4] H. Wegmann, "Fuzzy Control and Neural Networks Industrial Applications in the World of PLCs", Control Application 1994, Proceedings of the Third IEEE Conference, pp. 1245-1249 vol.2, 24-26 Aug. 1994.
- [5] P. Dadone and H. VanLandingham, "Genetic Algorithms for Tuning PLC Loops", Soft Computing Method in Industrial Application, IEEE SMCia/99, Kuusamo, Finland, pp. 58-62, 16-18 June 1999.
- [6] Standard Software for S7-300 and S7-400 - PID Control User Manual, SIEMENS AG.
- [7] S.Bennett, "Development of the PID Controller", IEEE Control Systems Magazine, pp. 58-65, Dec. 1993.
- [8] Yokogawa, "Expert Self-Tuning Controllers", 2nd Edition, Yokogawa Electronic Corp., July 1987.
- [9] C.C Hang and K.K.Sin, "A Comparative Performances Study of PID Auto-Tuners", IEEE Control Systems, pp. 41-47, Aug. 1991.
- [10] K. J. Astrom and T. Hagglund, "PID Controller: Theory, Design, and Tuning," 2nd Edition.
- [11] C.A. Smith, A.B. Corripio, "Principles and Practice of Automatic of Automatic Process Control," John Wiley & Sons, Inc., pp. 210-254, 1985.
- [12] IEC 1131 part 3 "Programming Languages for Programmable Controllers", CEI, 1992.
- [13] Hans Berger, "Automating with STEP7 in LAD: SIMATIC S7-300/400 programmable controllers," Pubicis MCD Verlag, 1998.

บรรณานุกรม (ต่อ)

- [14] L. Tanachaikhan, W. Sriratana, P.Pannil, A. Chaikla, P. Julsereewong and K. Tiraseath, "Application of Personal Computer as a Self-Tuning PID Controller," Proc. The 15th KACC 2000, Korea, pp.288-291, 19-21 October, 2000.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ก.
ผลงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์และได้รับการตีพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ICCAS '04

ICCAS 2004

Final Program & Abstract Book

International Conference on Control, Automation and Systems

August 25-27, 2004

The Shangri-La Hotel, Bangkok, Thailand

Organized by:

Research Center for Communications and Information Technology, King Mongkut's Institute of Technology Ladkrabang

The Institute of Control, Automation, and Systems Engineers, Korea



ICASE

In association with:



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Self-Tuning PID Controller Based on PLC

A. Phonphithak, P. Pannil, T. Suesut, R. Masuchun and P. Julserewong

Department of Instrumentation, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand
(Tel : +66-2-326-7346-7; Fax : +66-2-326-7346-7 Ext. 103; E-mail: kjprasit@kmitl.ac.th)

Abstract: The conventional PID (Proportional-Integral-Derivative) control technique is widely used for the process control in many industries since it is simple in structure and provides the good response. Nowadays, this control technique has been developed on the Programmable Logic Controller (PLC) to use for the process control loop. However, using this technique is difficult when tuning the PID parameters (K_p , T_i and T_d) to achieve the best response. Moreover, trial-and-error procedure along with the operator experiences are required to obtain the best results when tuning the PID controller parameters. This paper proposes the self-tuning PID controller based on PLC for the process control in the industries. The proposed self-tuning PID controller uses the PLC-based PID structures to control the process production. The proposed PID tuning utilizes the PLC to synthesize and analyze controller parameter as well as to tune for appropriate parameters using Dahlin method and extrapolation. Experimental results using a self-tuning PID controller to control temperature of the oven show that the controller developed is capable of controlling the process very effectively and provides a good response.

Keywords: Self-tuning, PID controller, PLC and extrapolation

1. INTRODUCTION

PLC is widely used in most industrial control systems as a process controller since it is low-cost, easy to use, and flexible compared with the earlier relay controller. Although, the original PLC is designed for a sequence control system, nowadays, it has been applied to control the continuous system [1-3]. In general, the special function blocks such as PID control, fuzzy logic control, etc. are included in PLC software packages [3-4]. The PID control function block is commonly used to control the process in industries by reason of it can provide good performance [5-6]. However, tuning a variety of values of the PID controller for the best response of the process can be time consuming and may cause a mistake if user lacks skill and sufficient experience. Therefore, most PID controllers used in the real practice still require trial-and-error as well as the operator experiences when tuning to obtain the best results [7-8]. Alternatively, the self-tuning control can be applied [9]. This paper presents an idea of PID tuning which can automatically analyze and tune appropriate parameters for the process using Dahlin method and extrapolation. Using this technique, the parameter of the process used to calculate the PID parameter can be obtained quickly. Not only is this technique very easy to use, it also provides a good response of the process.

2. PRINCIPLE AND THEORY

Most of the automatic control systems in the industrial process are the feedback control as constructed in Fig.1.

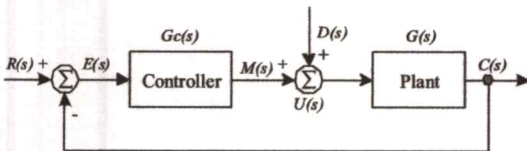


Fig. 1 Structure of the feedback control system.

As shown in Fig. 1, any control system has controller and plant. The plant consists of the final control element and the measurement with physical parameter(s). Controller is used to generate the control signal to the process. Usually, the controller is PID controller having the continuous equation as

$$m(t) = K_p \left[e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{de(t)}{dt} \right] \tag{1}$$

where $m(t)$ = controller output signal
 $e(t)$ = error signal
 K_p = Proportional gain
 T_i = Integral time
 T_d = Derivative time

When the mathematical model of process is known, the process can be easily controlled. However, in the real world, it is difficult to obtain the process model accurately. Thus, the approximation of process parameters by the process testing is still common and being used in industry.

2.1 Process Characterization

Generally, the transfer function of the process is represented by the first-order lag plus dead time (FOPDT) model since it provides the good approximation for the process industries. Hence, there are three parameters to be concerned; the process time constant (τ), the process gain (K) and the process dead time (t_0). The transfer function of a FOPDT model can be written as,

$$G(s) = \frac{K e^{-t_0 s}}{\tau s + 1} \tag{2}$$

The straightforward method to find these three parameters is to test an open loop with unit step signal while the disturbance signal is neglected. Consequently, the S-shaped response sketched in Fig. 2 can be obtained. This response curve covers the second order or higher plants with damping ratio equal or greater than one.

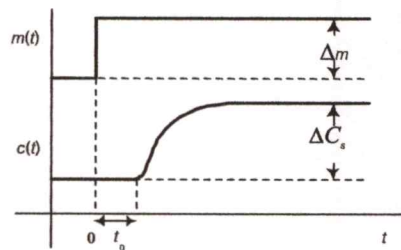


Fig. 2 Open loop step response of the process.

When the step change of controller output with the magnitude of Δm is applied to the process, the Laplace transform of the output response of the FOPDT process can be written as

$$C(s) = \frac{Ke^{-t_0s}}{\tau s + 1} \cdot \frac{\Delta m}{s} = K\Delta m e^{-t_0s} \left[\frac{1}{s} - \frac{\tau}{\tau s + 1} \right] \quad (3)$$

and

$$\Delta C(t) = K \cdot \Delta m \cdot u(t - t_0) [1 - e^{-(t-t_0)/\tau}] \quad (4)$$

The term $u(t - t_0)$ of Eq. (4) is determined to make the $\Delta C_s = 0$ at $t \leq t_0$.

The steady-state response can be given by Eq. (5) and the process gain can be obtained from Eq. (6).

$$\Delta C_s = \lim_{t \rightarrow \infty} \Delta C(t) = K\Delta m \quad (5)$$

$$K = \frac{\Delta C_s}{\Delta m} \quad (6)$$

Dead time of the process (t_0) is derived by checking time since the beginning until there is a response of the process about 1%. The time constant of τ can be obtained from the curve fitting procedure.

$$\text{Given that } t_1 = t_0 + \frac{\tau}{3}, t_2 = t_0 + \tau \quad (7)$$

$$\Delta C_1 = \Delta C(t_0 + \frac{\tau}{3}) = 0.283K\Delta m = 0.283\Delta C_s$$

$$\Delta C_2 = \Delta C(t_0 + \tau) = 0.632K\Delta m = 0.632\Delta C_s$$

$$\text{hence, } \tau = \frac{3}{2}(t_2 - t_1) \quad (8)$$

2.2 PID controller Parameter Synthesis

The PID controller parameter synthesis can be performed in closed-loop control. From Fig.1, the closed-loop transfer function with unity feedback can be expressed in terms of the controller transfer function as

$$G_c(s) = \frac{1}{G(s)} \cdot \frac{C(s)/R(s)}{1 - [C(s)/R(s)]} \quad (9)$$

Theoretically, the most suitable response to feedback control system is the response in the form of First Order Lag. However, dead time of this control system remains negative making this controller becomes unpractical. Therefore, it is necessary to consider the control system, which provides First Order Lag response with dead time t_0 as shown in Eq. (10).

$$\frac{C(s)}{R(s)} = \frac{e^{-t_0s}}{\tau_c s + 1} \quad (10)$$

where τ_c is the closed-loop response time constant.

From Eq.(1), (9) and Eq. (10), where the exponential term can be approximated by a Taylor series expansion

$e^{-t_0s} = 1 - t_0s$, the controller transfer function can be rewritten as

$$G_c(s) = \frac{\tau}{K(\tau_c + t_0)} \left[1 + \frac{1}{\tau s} \right] \quad (11)$$

Eq. (11) shows that the controller required for FOPDT model is the PI controller with tuning parameter as

$$K_c = \frac{\tau}{K(\tau_c + t_0)} \text{ and } T_i = \tau \quad (12)$$

However, the Dahlin synthesis results for this FOPDT model is the PID controller with tuning parameter as shown in [10]

$$K_c = \frac{\tau}{K(\tau_c + t_0)}, \tau_I = \tau \text{ and } \tau_D = \frac{t_0}{2} \quad (13)$$

The PID controller can be applied to the process control system when the process dead time t_0 is greater than $\tau/4$, otherwise, only the PI controller is used. If the response having an overshoot of 5% is required for the setpoint changes, $\tau_c = t_0$.

2.3 Extrapolation

Extrapolation is an estimation of future data by analyzing data obtained from various samplings. A suitable method of numerical analysis is Newton's Divided-Difference Extrapolating Polynomial. It is used in this paper to calculate Δc_s in advance. The value obtained from this calculation method has a slight error by using the following equation

$$f(x) = f(x_0) + (x - x_0)f[x_1, x_0] + (x - x_0)(x - x_1)f[x_2, x_1, x_0] + \dots + (x - x_0)(x - x_1)\dots(x - x_{n-1})f[x_n, x_{n-1}, \dots, x_0] \quad (14)$$

when $f[x_1, x_0] = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$

$$f[x_n, x_{n-1}, \dots, x_0] = \frac{f[x_n, \dots, x_1] - f[x_{n-1}, \dots, x_0]}{x_n - x_0}$$

x_n = data from each sampling

$f(x)$ = value from extrapolation

User is able to select suitable size of the sampling period. In this paper, 3 sampling data, x_0, x_1 and x_2 , are selected to calculate future data x_3 and x_4 representing the tendency of process response. If $x_4 > x_3$, then it will calculate x_4 and x_5 from sampling data x_1, x_2, x_3 . The calculation repeats until $x_n \geq x_{n+1}$ which is when the response becomes stable. Once the data is obtained, $x_n = \Delta c_s$, the computer will analyze the data and figure out the parameters from the test.

3. DESIGN AND OPERATION

The developed system uses PLC as PID controller to automatically analyze and tune the process's parameter. The process variable (pv) or response signal (standard analog signal 420 mAdc. or 15 Vdc.) from the process will be transmitted to the PLC via analog input module to calculate all parameters. The operation and monitoring of the process can be controlled by PC with the WinCC (Windows Control Center) software via PCI card interface (CP5611). This signal is used to analyze the process parameters and the PID controller parameters according to the proposed procedure.

When analyzing characteristics of the process, there is a test by inputting a unit step signal into the process so that user can select the appropriate size of the signal. In addition, dead time value of the process can be obtained from this test. Once the resulting response becomes stable, Δc_s is used to calculate steadystate gain and time constant of the process. With the test of analyzing the Δc_s , the process takes some time until the response becomes stable. Therefore, in this paper, the numerical analysis is used to estimate value of Δc_s by comparing data from sampling while inputting the unit step signal during the test, which is called the extrapolation. The estimation by this method will help the process characteristic analysis take less time. When the process parameters are obtained, the PID parameters are computed and shown on the monitor; thus, user can decide whether this PID parameter's tuning is satisfied. If the user agrees on these parameters, the process will be automatically controlled and the response is shown on the monitor. The operation flowchart of the proposed control technique according to the IEC1131-3 standard [11], which is written using the ladder diagram, function block diagram, and function list, is shown in Fig. 3. The operation diagram can be seen from Fig. 4.

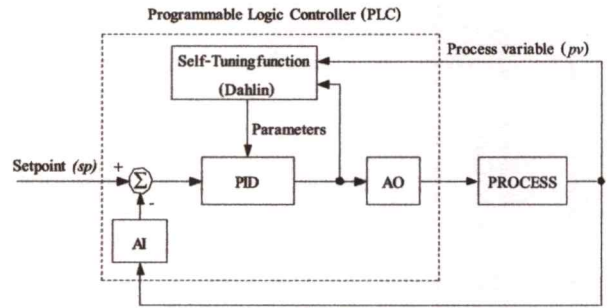


Fig. 4 Operation diagram of the proposed control system.

4. EXPERIMENTAL RESULTS

The performance of the proposed controller was observed from the experiments with temperature control of the oven. Fig. 5 - Fig. 7 show the experimental setup.

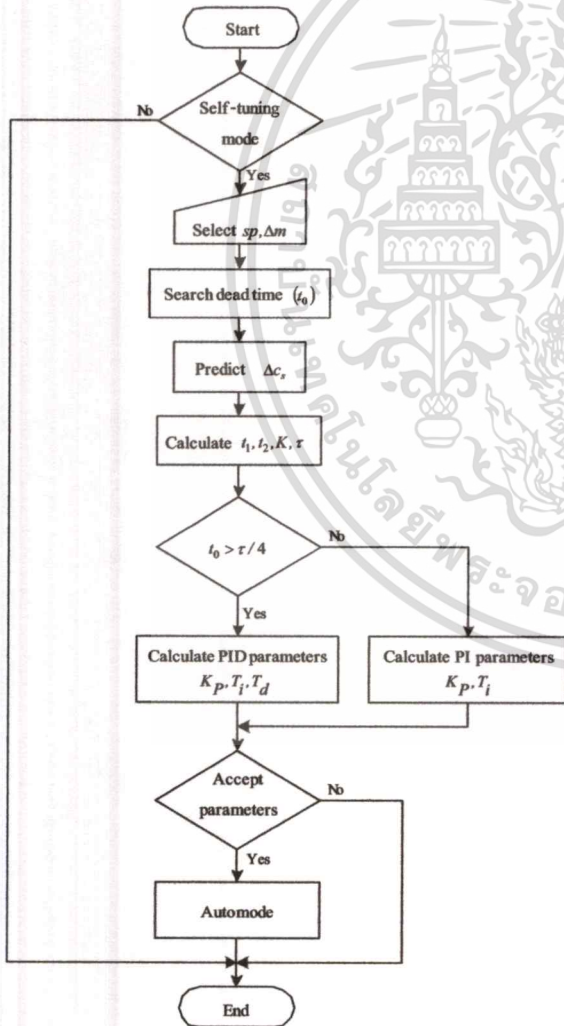


Fig. 3 Operation flowchart.

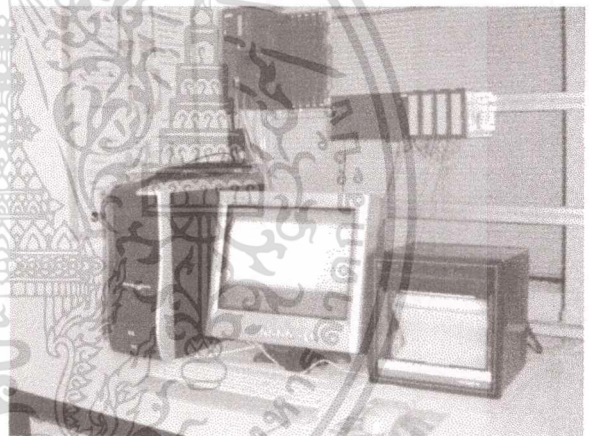


Fig. 5 Control station (PC, PLC and PCI interface card).

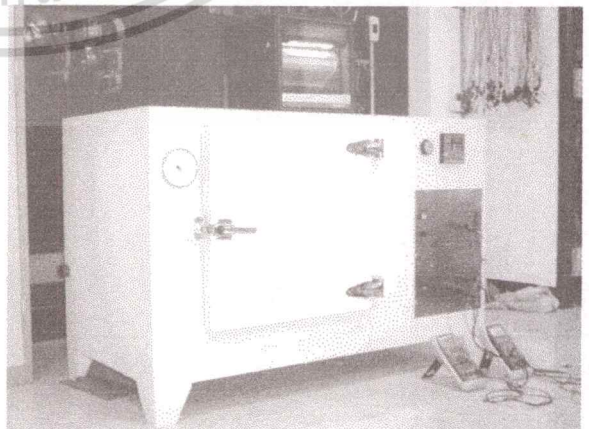


Fig. 6 Temperature control plant model (oven).

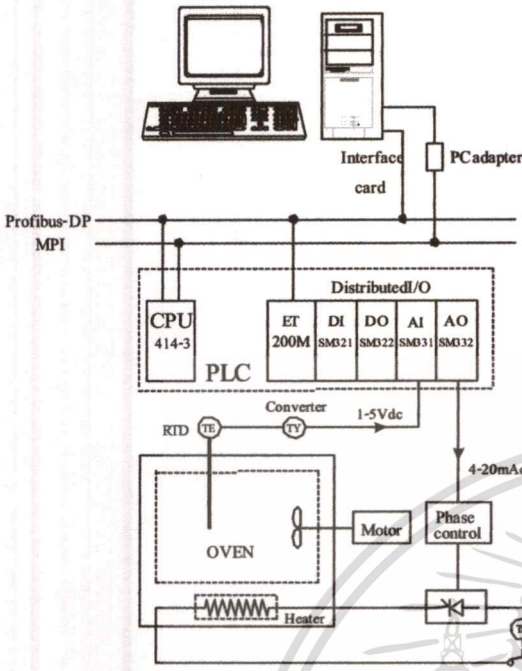


Fig. 7 Experimental setup diagram.

In the experiment, the PLC of SIEMENS by SIMATIC S7 family is used. This PLC system consists of the CPU414-3, Distributed I/O station (ET200M, SM321, SM322, SM331, and SM332), PC Adaptor, CP5611 PCI card, STEP7 Manager software (to write the ladder diagram, function block diagram, and function list), and WinCC software (to design the user interface). All are communicated via Profibus-DP protocol.

The process characterization by the process reaction curve and extrapolation of the experimental thermal plant are shown in Fig. 8 and Fig. 9, respectively. Fig. 10 shows the comparison of the process characterization of both procedures.

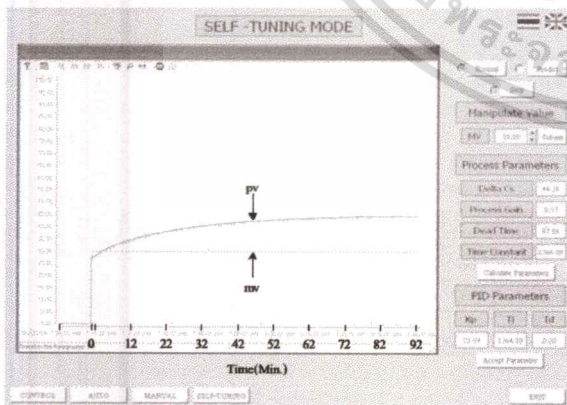


Fig. 8 Open loop step response at $\Delta m = 30\%$ in normal mode (process characterization by process reaction curve).

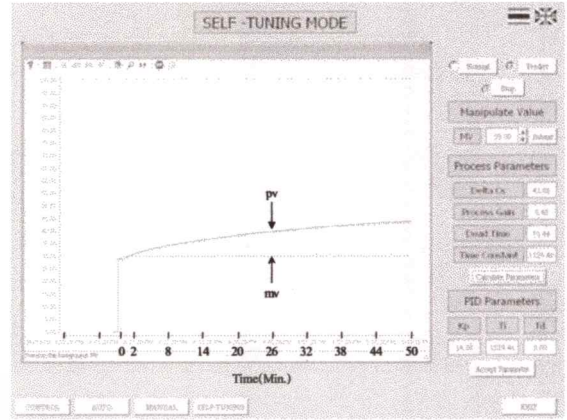


Fig. 9 Open loop step response at $\Delta m = 30\%$ in the predictive mode (process characterization by extrapolation).

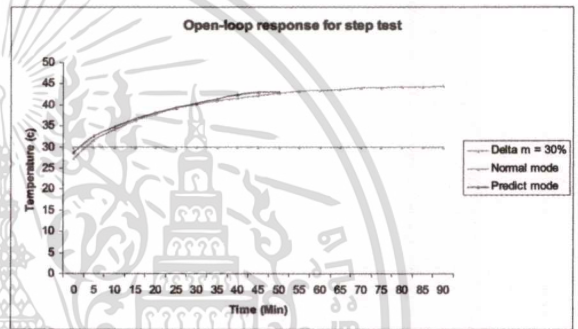


Fig. 10 Open loop step response in normal mode compare with predictive mode ($\Delta m = 30\%$).

From Fig. 8, the process characterization in normal mode by the process reaction curve can be obtained. The time consumed to obtain the process parameters is approximately 90 minutes with $K\theta.57$, $\tau = 364.09$ sec., $t_d = 87.86$ sec., and $\Delta C_s = 44.38$ °C. The predictive mode gives $K\theta.48$, $\tau = 1229.46$ sec., $t_d = 91.44$ sec., and $\Delta C_s = 43.08$ °C with time spent approximately 50 minutes (Fig. 9). The results show that the performance of the process characterization in predictive mode is fast and ΔC_s are slightly different, only 1.3°C.

The step response of temperature control at setpoint 70 °C. and temperature control at setpoint 70 °C with step change to setpoint 90 °C. using the PID parameters obtained from the predictive mode with $K_p = 4.08$ and $T_i = 1229.46$ sec. are shown in Fig. 11 and Fig. 12, respectively.

5. CONCLUSIONS

This paper proposes the self-tuning PID controller based on PLC for the process control in the industries. The proposed self-tuning PID controller can automatically analyze and tune the appropriate PID parameter for the process without any error or damage that may occur due to insufficient skill of user. The extrapolation is applied to estimate the process parameters from a process open loop step test based on first-order lag plus dead time (FOPDT) model. Using this technique, the process parameters used to synthesize the PID parameter by the formula of Dahlin can achieve in a short

time. In addition, the proposed controller structure can determine to synthesize the PID parameter by Ziegler-Nichols and Coon-Cohen formulas and others methods if required. According to experimental results of the temperature control process, it is shown that using the proposed controller can provide the good performance of the control system.

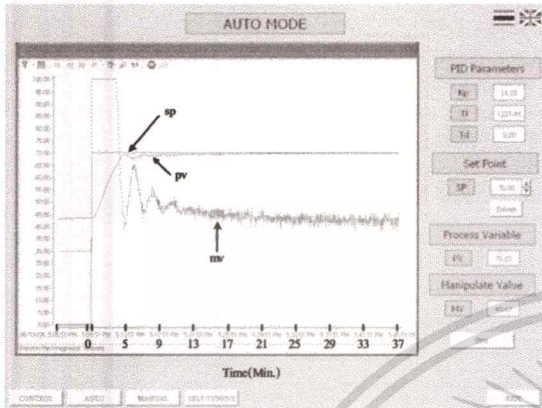


Fig. 11 Step response of temperature control at setpoint 70 °C.

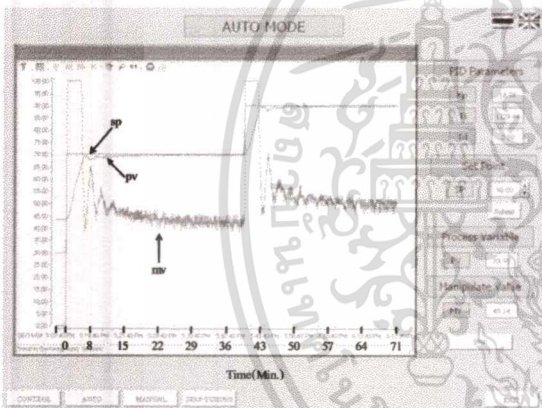


Fig. 12 Step response of temperature control at setpoint 70 °C and step change to setpoint 90 °C.

REFERENCES

- [1] B-S Kang and K-H Cho, "Design of PLCs for Automated Industrial Systems Based on Discrete Event Models," *IEEE ISIE 2001*, Pusan, Korea, pp. 1431-1434, 2001.
- [2] T. Krairojananan and S. Suthapradit, "A PLC Program Generator Incorporating Sequential Circuit Synthesis Techniques," *Circuit and systems, The 1998 IEEE Asia-Pacific Conference*, pp. 399-402, 24-27 Nov. 1998.
- [3] S. Qiang, X. Gao and XZhuang, "PLC-based Control System for Industrial Production of Fuel Alcohol," *IEEE ICIT'02, International Conference*, Bangkok, Thailand, pp.827-832, 11-14 Dec. 2002.
- [4] H. Wegmann, "Fuzzy Control and Neural Networks Industrial Applications in the World of PLCs," *Control Application 1994., Proceedings of the Third IEEE Conference*, pp.1245-1249 vol.2, 24-26 Aug. 1994.
- [5] P. Dadone and H. VanLandingham, "Genetic Algorithms for Tuning PLC Loops," *Soft Computing Method in Industrial Application, IEEE SMCia/99*, Kuusamo, Finland, pp.58-62, 16-18 June 1999.
- [6] *Standard Software for S7-300 and S7-400 - PID Control User Manual*, SIEMENS AG.
- [7] S.Bennett, "Development of the PID Controller," *IEEE Control Systems Magazine*, pp. 58-65, Dec. 1993.
- [8] Yokogawa, "Expert Self-Tuning Controllers," 2nd Edition, *Yokogawa Electronic Corp.*, July 1987.
- [9] C.C Hang and K.K.Sin, "A Comparative Performances Study of PID Auto-Tuners," *IEEE Control Systems*, pp. 41-47, Aug. 1991.
- [10] C.A. Smith, A.B. Corripio, "Principles and Practice of Automatic of Automatic Process Control," *John Wiley & Sons, Inc.*, pp. 210254, 1985.
- [11] IEC 1131 part 3 "programming Languages for Programmable Controllers," CEI, 1992

ประวัติผู้เขียน

ชื่อ-นามสกุล นายอภิชาติ พลพิทักษ์
 วัน เดือน ปีเกิด 22 กันยายน 2521 ที่จังหวัดกาญจนบุรี
 ที่อยู่ 130/79 ซ.เดิมนุญ ถ.บางขุนนนท์ แขวงบางขุนนนท์
 เขตบางขุนนนท์ กรุงเทพมหานคร 10700
 ประวัติการศึกษา 2544 วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมการควบคุม
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ผลงานทางวิชาการที่ได้รับการยอมรับ

พ.ศ.2547 “Self-Tuning PID Controller Based on PLC”, ICCAS’04
 (International Conference on Control, Automation and Systems 2004),
 Bangkok, Thailand, pp272-276, 25-27 Aug. 2004

ประสบการณ์ทำงาน

พ.ศ.2545-2548 ตำแหน่ง Service Engineer
 บริษัท บางกอก ออโตเมชัน เอ็นจิเนียริง จำกัด
 พ.ศ.2548-ปัจจุบัน ประกอบธุรกิจส่วนตัว