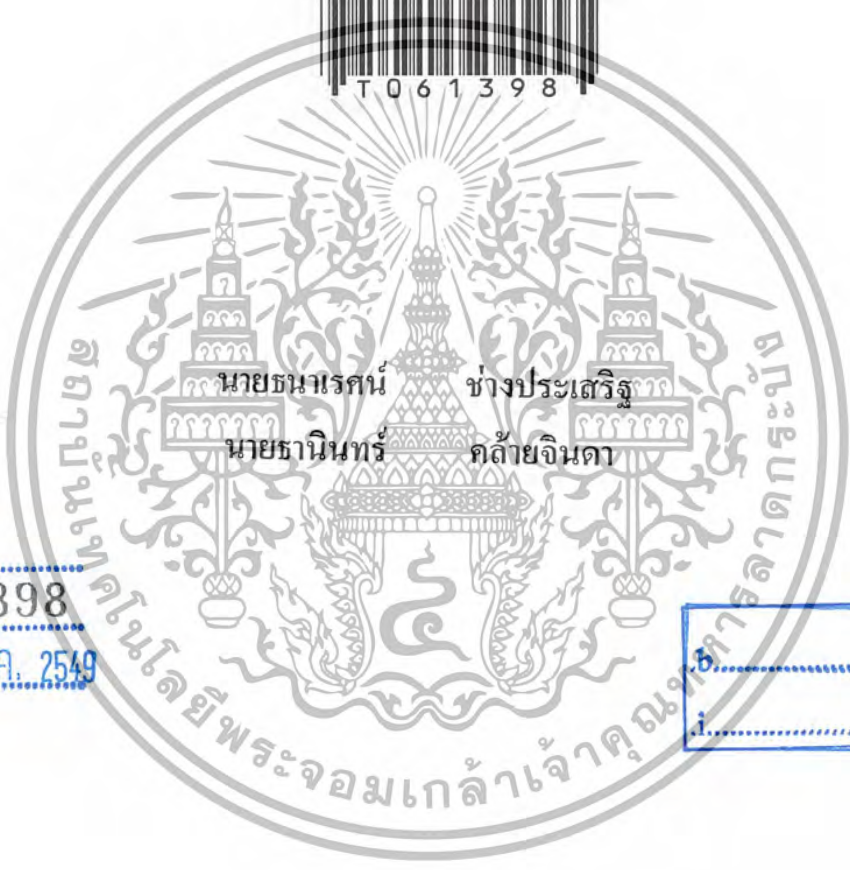


สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การออกแบบและพัฒนาชุดทดลองกระบวนการควบคุมระดับแบบมีโมดูล
ไ้ผลตอบสนอง

THE DEVELOPMENT WATER - LEVEL CONTROL SYSTEM WITH
DELAY TIME MODULE



นายธนาเรศน์ ช่างประเสริฐ
นายธานีินทร์ คล้ายจินดา

เลขหมู่.....
เลขทะเบียน..... 61398
วัน,เดือน,ปี 17 ก.ค. 2549



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**THE DEVELOPMENT WATER - LEVEL CONTROL SYSTEM WITH
DELAY TIME MODULE**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING
DEPARTMENT OF INSTRUMENTATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2004

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญาโท

หัวข้อปริญญาโท การออกแบบและพัฒนาชุดทดลองกระบวนการควบคุมระดับน้ำ
อัตโนมัติแบบมีโมดูลไ้ผลตอบสนอง
THE DEVELOPMENT WATER - LEVEL CONTROL SYSTEM
WITH DELAY TIME MODULE

นักศึกษาผู้จัดทำ นายธนารศน์ ช่างประเสริฐ รหัสประจำตัว 45015552
นายธานีทร์ กล้ายจินดา รหัสประจำตัว 45015554

ปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมการวัดคุม
ปีการศึกษา 2547

อาจารย์ผู้ควบคุมปริญญาโท	ลายมือชื่อ
ผศ.อาจินต์ น่วมสำราญ	
ผศ.ดร.ธนิศย์ ตรีสุวรรณวัฒน์	

วัน/เดือน/ปี ที่สอบ วันพฤหัสบดีที่ 24 มีนาคม พ.ศ. 2548
สถานที่สอบ ณ.ห้องสอบปริญญาโท ภาควิชาวิศวกรรมการวัดคุม

ภาควิชารับรองแล้ว

(รศ.ประสิทธิ์ จุลเสวีวงศ์)

หัวหน้าภาควิชาวิศวกรรมการวัดคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญาโท การออกแบบและพัฒนาชุดทดลองกระบวนการควบคุมระดับน้ำ
แบบมีโมดูลไร้ผลตอบสนอง
THE DEVELOPMENT WATER - LEVEL CONTROL SYSTEM
WITH DEALY TIME MODULE

นักศึกษาผู้จัดทำ นายชนาเรศน์ ช่างประเสริฐ
นายธานินทร์ คล้ายจินดา
อาจารย์ที่ปรึกษา ผศ.อาจินต์ น่วมสำราญ
ผศ.ดร.ธนิตย์ ศรีสุวรรณวัฒน์
ปีการศึกษา 2547

บทคัดย่อ

ตัวควบคุมแบบ PID ถูกนำมาใช้อย่างกว้างขวางในวงการด้านอุตสาหกรรมร่วมกับวิธีการ
ปรับแต่งค่าพารามิเตอร์ PID ของ Ziegler - Nichols และวิธีอื่น ๆ อย่างไรก็ตามในวิธีการปรับแต่ง
ค่าพารามิเตอร์ของตัว PID โดยทั่วไปจะเป็นการประมาณค่าคงที่ให้ผลดีในระดับหนึ่งและอาจเกิดค่า
ผิดพลาดในการวัด ค่าความผิดพลาดของความคลาดเคลื่อนในระบบ อาจส่งผลกระทบต่อประสิทธิภาพของ
ระบบควบคุมและบางครั้งอาจมีผลต่อเสถียรภาพของระบบอีกด้วย ในทำนองเดียวกับระบบที่มีตัว
หน่วงเวลาอยู่ภายในระบบ ตัวหน่วงเวลาจะมีผลกระทบกับเสถียรภาพของระบบในปริญญาโท
เล่มนี้จะกล่าวถึงการออกแบบระบบตัวควบคุมเสถียรภาพ ให้กับระบบที่มีทั้งตัวหน่วงเวลาและค่า
ความผิดพลาดคลาดเคลื่อนอยู่ภายในระบบ โดยระบบที่จะศึกษานี้มีตัวหน่วงเวลาแบบค่าคงที่ที่
ทราบค่าและค่าความผิดพลาดคลาดเคลื่อนที่ไม่ทราบค่าแน่นอนแต่ถ้าทราบค่าขอบเขตอยู่ในถ้า
ตัวระบบหลักทำให้เสถียรได้ และค่าความผิดพลาดคลาดเคลื่อนสอดคล้องกับเงื่อนไขที่กำหนดไว้
แล้วเราก็จะสามารถออกแบบตัวควบคุมเสถียรภาพที่เหมาะสมกับระบบได้

Thesis	The Development Water – Level Control System With Delay Time Module	
Author	Mr.Tanares	Changprasert
	Mr.Tanin	Klaijinda
Thesis Advisor	Asst.Prof.Arjin	Noumsomran
	Asst.Prof.Dr.Tanit	Treesuwannawat
Year	2004	

ABSTRACT

The PID (Proportional – integral – derivative) controller is widely used in the industries with the well known Ziegler – Nichols tuning method and other varieties. However, these methods yield good estimate of the PID controller parameters but the large error may from measurement. The uncertainties in the system are possibly made system performance deteriorate and also sometimes destroy system stability. In the other hand, the time-delay systems which often occur in various real processes and in other transport phenomena are the source of instability also. This thesis is mainly concerned with the controller design for the uncertain time-delay system. The system under consideration is linear system with known constant state delay and also contains uncertain parameters whose elements are continues, unknown but bounded. When the nominal system is spectrally stabilizable and its uncertain parts satisfy certain matching conditions, a suitable control law can be derived.

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี เพราะได้รับความเมตตาจาก ผศ.ดร.ธนิตย์ ศรีสุวรรณวัฒน์ ซึ่งเป็นอาจารย์ผู้ควบคุมปริญญาานิพนธ์ ผู้วิจัยรู้สึกทราบซึ่งและขอกราบขอบพระคุณเป็นอย่างสูงที่ได้ให้คำแนะนำและคำปรึกษาในการวิจัย

ขอขอบพระคุณอาจารย์ อาจินต์ น่วมสำราญซึ่งได้ให้คำปรึกษาและให้ความช่วยเหลือในการทำวิจัยและช่วยตรวจสอบแก้ไขข้อผิดพลาดต่าง ๆ พร้อมทั้งวิธีแก้ปัญหา

ขอขอบพระคุณ คณะอาจารย์ และภาควิชาวิศวกรรมการวัดคุมทุกท่าน ที่ได้ให้คำแนะนำอันเป็นประโยชน์ เอื้อเฟื้ออุปกรณ์ และเครื่องมือทั้งหมด ตลอดจนสถานที่ จนสามารถทำปริญญาานิพนธ์นี้สำเร็จ

และที่ลืมเสียมิได้คือ ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ อันเป็นที่รักยิ่ง ที่สนับสนุนและเป็นแรงบันดาลใจในการศึกษาและทำปริญญาานิพนธ์ฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากปริญญาานิพนธ์ฉบับนี้ ผู้วิจัยขอบแต่ผู้มีพระคุณทุกท่าน



คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติมากรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญภาพ.....	VII

บทที่ 1 บทนำ

1.1 ความเป็นมาและเหตุจูงใจของกรวิจัย.....	1
1.2 วัตถุประสงค์ของปริญญาโท.....	1
1.3 ขอบเขตของปริญญาโท.....	2

บทที่ 2 ทฤษฎีการควบคุมกระบวนการ

2.1 ระบบควบคุมกระบวนการ (Process Control System).....	3
2.1.1 ระบบการควบคุมแบบดั้งเดิม (Classical).....	3
2.1.2 ระบบควบคุมแบบสมัยใหม่ (Modern Control System).....	3
2.2 การควบคุมกระบวนการแบบอัตโนมัติ (Automatic Control System).....	3
2.2.1 โครงสร้างพื้นฐานของระบบควบคุมอัตโนมัติ.....	4
2.3 คำจำกัดความ.....	5
2.3.1 อินพุท (Input).....	5
2.3.2 ตัวควบคุม (Controller).....	5
2.3.3 กระบวนการ (Plant or Process).....	5
2.3.4 เอาท์พุท (Output).....	5
2.3.5 การรบกวน (Disturbance).....	5
2.3.6 อุปกรณ์วัด (Measuring Instruments).....	6
2.3.7 ระบบ (System).....	6
2.3.8 ระบบควบคุม (Control System).....	6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
2.4 การออกแบบระบบควบคุมในโดเมนเวลา.....	8
2.4.1 ขั้นตอนการออกแบบระบบควบคุม.....	8
2.5 ความรู้เบื้องต้นเกี่ยวกับการควบคุมแบบป้อนกลับด้วยตัวควบคุม PID.....	10
2.5.1 เครื่องควบคุม.....	10
2.5.2 หลักการของตัวควบคุมกระบวนการแบบ PID.....	10
2.5.3 รูปแบบของตัวควบคุมแบบ PID.....	11
2.6 กริยาต่าง ๆ ของการควบคุม.....	13
2.6.1 การควบคุมแบบ Proportional.....	13
2.6.2 การควบคุมแบบ Proportional + Integral.....	13
2.6.3 การควบคุมแบบ Proportional + Integral + Derivative.....	13
2.7 การปรับแต่งหาค่าพารามิเตอร์ของเครื่องควบคุม.....	14
2.7.1 วิธีการแบบ Reaction Curve (Open Loop Method).....	14
2.8 ระบบเกี่ยวกับระดับของเหลว.....	16
2.8.1 ค่าความต้านทาน.....	16
2.9 ผลตอบสนองของระบบต่อสัญญาณอินพุตชนิดต่างๆ.....	19
2.9.1 ระบบอันดับหนึ่ง (Response of First-Order-System).....	19
2.10 การวิเคราะห์หาค่าคุณลักษณะของกระบวนการ.....	23
2.11 การสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุม PID ในการควบคุม.....	27
บทที่ 3 อุปกรณ์ฮาร์ดแวร์และทฤษฎีที่เกี่ยวข้อง	
3.1 องค์ประกอบของการควบคุม.....	32
3.1.1 กระบวนการ (Process).....	32
3.1.2 (Measuring Device)	32
3.1.3 เครื่องควบคุมหรือตัวควบคุม (Controller).....	32
3.1.4 อุปกรณ์ปรับกระบวนการ (Final Control Element).....	32
3.2 อุปกรณ์และฮาร์ดแวร์.....	32
3.2.1 Computer (Controller)	33

สารบัญ(ต่อ)

	หน้า
3.2.2 Signal Converter.....	33
3.2.3 Final Control Element.....	37
3.2.4 Sensor and Transmitter.....	38
3.3 ทฤษฎีและหลักการของ LAB VIEW.....	40
3.4 การสร้าง VI.....	40
3.4.1 ฟรอนต์พาเนล (Front panel).....	40
3.4.2 บล็อกไดอะแกรม (Block diagram).....	41
3.5 องค์ประกอบของ Front Panel Toolbar.....	43
3.6 องค์ประกอบของ Block Diagram Toolbar.....	44
3.7 การพัฒนาโปรแกรม.....	44
3.7.1 หลักการทำงานของ Dataflow Programming.....	44
3.7.2 Data Type (ประเภทของข้อมูล).....	45
3.8 เทคนิคการดีบักกิ้ง (Debugging).....	47
3.8.1 Finding Error.....	47
3.8.2 Execution Highlight.....	47
3.8.3 Execution Highlight button Single Step Through.....	48
3.8.4 Button Probe.....	48
3.8.5 Breakpoint.....	49
3.9 การสร้างโปรแกรมย่อย (Sub VI).....	50
3.9.1 การใช้งาน โปรแกรมย่อย (Sub VI).....	51
3.10 ตัวอย่างการใช้งานฟังก์ชันใน LAB VIEW.....	52
3.10.1 While loop.....	52
3.10.2 Waveform Chart.....	53
3.11 การใช้ซีพีอาร์จิสเตอร์.....	55
3.11.1 For Loop.....	56
3.11.2 Case Structure.....	58
3.12 โปรแกรมที่ใช้ในการควบคุม.....	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา แล VI อ่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 4 ผลการทดลอง

4.1 สมการคุณลักษณะของกระบวนการ.....	62
4.2 ความยาวสาย $L = 0.5$ เมตร.....	62
4.2.1 จากกระบวนการ ได้ใส่ค่าพารามิเตอร์ P.....	64
4.2.2 จากกระบวนการ ได้ใส่ค่าพารามิเตอร์ PI.....	64
4.2.3 จากกระบวนการ ได้ใส่ค่าพารามิเตอร์ PID.....	64
4.2.4 จากกระบวนการ ได้ใส่ค่า Disturbance เพื่อเป็นการรบกวน กระบวนการ.....	65
4.3 ความยาวสาย $L = 8$ เมตร.....	66
4.3.1 จากกระบวนการ ได้ใส่ค่าพารามิเตอร์ P.....	67
4.3.2 จากกระบวนการ ได้ใส่ค่าพารามิเตอร์ PI.....	68
4.3.3 จากกระบวนการ ได้ใส่ค่าพารามิเตอร์ PID.....	68
4.3.4 จากกระบวนการ ได้ใส่ค่า Disturbance เพื่อเป็นการรบกวน กระบวนการ.....	69
4.4 ความยาวสาย $L = 13$ เมตร.....	70
4.4.1 จากกระบวนการ ได้ใส่ค่าพารามิเตอร์ P.....	71
4.4.2 จากกระบวนการ ได้ใส่ค่าพารามิเตอร์ PI.....	72
4.4.3 จากกระบวนการ ได้ใส่ค่าพารามิเตอร์ PID.....	72
4.4.4 จากกระบวนการ ได้ใส่ค่า Disturbance เพื่อเป็นการรบกวน กระบวนการ.....	73

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ

5.1 บทสรุป.....	74
5.2 ข้อเสนอแนะและแนวทางการพัฒนา.....	75

สารบัญตาราง

ตารางที่	หน้า
2.1 สูตรหาค่าพารามิเตอร์ของตัวควบคุม PID ตามวิธี Process Reaction Curve.....	17
4.1 แสดงสูตรในการหาค่าพารามิเตอร์ Kc, P, PI, PID.....	63
4.2 แสดงค่าที่คำนวณหาค่าพารามิเตอร์ Kc, P, PI, PID.....	63
4.3 แสดงสูตรในการหาค่าพารามิเตอร์ KC, P, PI, PID.....	67
4.4 แสดงค่าที่คำนวณหาค่าพารามิเตอร์ KC, P, PI, PID.....	67
4.5 แสดง สูตรในการหาค่าพารามิเตอร์ KC, P, PI, PID.....	71
4.6 แสดงค่าที่คำนวณหาค่าพารามิเตอร์ KC, P, PI, PID.....	71



สารบัญรูป

รูปที่	หน้า
2.0 แสดงรูปแบบการควบคุมแบบอัตโนมัติ(Automatic Control System Configuration).....	5
2.1 แสดงบล็อกไดอะแกรมของระบบควบคุมแบบวงเปิด(Open Loop Control System).....	7
2.2 แสดงการควบคุมแบบปิด (Close Loop Control System).....	7
2.3 ก, ข, ค และ ง แสดงรูปแบบของตัวควบคุมแบบ PID.....	12
2.4 แสดงโครงสร้างพื้นฐานของตัวควบคุม PID.....	13
2.5 แสดงโครงสร้างของระบบที่ควบคุมด้วยตัวควบคุม PID.....	15
2.6 แสดงผลตอบสนองรูปตัว S เมื่อใช้วิธี Process Reaction Curve.....	16
2.7 แสดงระบบเกี่ยวกับของเหลว.....	17
2.8 แสดงรูปกราฟเมื่อของเหลวไหลผ่าน.....	18
2.9 แสดงผลตอบสนองของอิมพัลส์ที่เป็น Unit Impulse.....	21
2.10 แสดงผลตอบสนองของอิมพัลส์ที่เป็น Unit Step.....	23
2.11 แสดงบล็อกไดอะแกรมของระบบควบคุมแบบป้อนกลับ.....	24
2.12 แสดงบล็อกไดอะแกรมของระบบควบคุมแบบป้อนกลับเมื่อทำการลดยุติ.....	25
2.13 แสดงบล็อกไดอะแกรมของการทดสอบวงเปิดโดยการป้อนสัญญาณแบบขั้นบันได.....	26
2.14 แสดงผลตอบสนองของกระบวนการแบบวงเปิดต่อสัญญาณขั้นบันไดรูปที่.....	26
2.15 แสดงกราฟแสดงการหาค่า Δc_s	27
2.16 แสดงบล็อกไดอะแกรมของการสังเคราะห์ค่าพารามิเตอร์ของตัวควบคุมแบบป้อนกลับ.....	29
2.17 แสดงผลตอบสนองของระบบแบบป้อนกลับในรูปแบบอันดับหนึ่ง.....	30
2.18 แสดงผลตอบสนองแบบปิดที่มีการหน่วงเวลาเป็นเวลา t_0	31
3.1 แสดง Block Diagram ของระบบ.....	32
3.2 แสดงการ์ดแปลงสัญญาณ Analog/Digital และ Digital/Analog.....	34
3.3 แสดงการ Set Jumper เพื่อเลือกขานของสัญญาณเอาต์พุต.....	35
3.4 แสดงการ Set Jumper เพื่อเลือกขานของสัญญาณอินพุต.....	35
3.5 แสดงการช่องสัญญาณ Analog Input/Output ของการ์ด.....	36
3.6 แสดงเป็นตารางแสดงรายละเอียดของ A/D Card รุ่น AD12-16(PCI)E.....	36

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.7 แสดงวงจรรอบแอมป์แบบไม่กลับเฟส (Non-inverting) ที่ใช้ขับมอเตอร์.....	37
3.8 แสดงชุดขับกระแสมอเตอร์ (Motor Driver).....	38
3.9 แสดงชุด DP Cell (Differential Pressure Transmitter).....	38
3.10 แสดงรูปของ Plant จริงของการจำลองกระบวนการควบคุมระดับน้ำแบบ มีโมดูลไร้ผลตอบสนอง.....	39
3.11 แสดง Front Panel และ Block Diagram.....	41
3.12 แสดง Front Panel Toolbar.....	43
3.13 แสดง หลักการทำงานของ Dataflow Programming.....	45
3.14 แสดง Data Type (ประเภทของข้อมูล).....	46
3.15 แสดงการ Wiring ที่ผิด.....	46
3.16 แสดง Error List.....	47
3.17 แสดง Finding Error.....	47
3.18 แสดง Execution Highlight.....	48
3.19 แสดง Execution Highlight button Single Step Through.....	48
3.20 แสดง Execution Highlight button Single Step Through.....	49
3.21 แสดง Breakpoint.....	49
3.22 แสดง Icon Edit.....	50
3.23 แสดงเทอมินอล ป๊อปอัพเมนู.....	51
3.24 แสดงการเรียกใช้งาน โปรแกรมย่อย.....	52
3.25 แสดงขีดเพดานของ Function While loop.....	53
3.26 แสดงFunction While loop บนหน้า Diagram.....	53
3.27 แสดง Waveform Chart.....	54
3.28 แสดง Function strip chart, scope chart และ sweep chart บนหน้า Front Panel.....	54
3.29 แสดงSingle plot chart.....	55
3.30 แสดง Multiple chart plot waveform chart.....	55

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.31 แสดงShift Register.....	56
3.32 แสดงการทำงานของ Shift Register.....	56
3.33 แสดง การเลือก For Loop.....	57
3.34 แสดง For Loop.....	57
3.35 แสดงตัวอย่างการใช้ For Loop.....	58
3.36 แสดงการเลือกCase Structure.....	58
3.37 แสดง Case Structureซึ่งสามารถรับค่าได้ทั้งที่เป็นนิวเมอริก, บูลีน หรือ สตริง.....	59
3.38 แสดงโปรแกรมที่ใช้ควบคุมกระบวนการในรูปของบล็อกไดอะแกรม.....	60
3.39 แสดงโปรแกรมที่ใช้ควบคุมกระบวนการที่แสดงอยู่ในรูปของ Front Panel.....	61
4.0 แสดงวิธีการหาแบบจำลองกระบวนการแบบเปิด.....	62
4.1 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ P เข้าไป.....	64
4.2 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PI เข้าไป.....	64
4.3 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PID เข้าไป.....	65
4.4 แสดงกราฟของการควบคุมกระบวนการเมื่อมีการรบกวนกระบวนการ.....	65
4.5 แสดงวิธีการหาแบบจำลองกระบวนการแบบเปิด.....	66
4.6 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ P เข้าไป.....	68
4.7 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PI เข้าไป.....	68
4.8 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PID เข้าไป.....	69
4.9 แสดงกราฟของการควบคุมกระบวนการเมื่อมีการรบกวนกระบวนการ.....	70
4.10 แสดงวิธีการหาแบบจำลองกระบวนการแบบเปิด.....	71
4.11 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ P เข้าไป.....	72
4.12 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PI เข้าไป.....	72
4.13 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PID เข้าไป.....	73
4.14 แสดงกราฟของการควบคุมกระบวนการเมื่อมีการรบกวนกระบวนการ.....	73

บทที่ 1

บทนำ

1.1 ความเป็นมาและเหตุจูงใจของการวิจัย

แนวความคิดในการนำระบบควบคุมระดับน้ำอัตโนมัติ มีจุดเริ่มต้นมาจากความต้องการนำระบบควบคุมแบบอัตโนมัติมาใช้แทนระบบควบคุมแบบเดิมซึ่งอาศัยการสั่งงานจากผู้ปฏิบัติการในการออกแบบระบบควบคุมอัตโนมัติ สำหรับการควบคุมกระบวนการทางอุตสาหกรรมขนาดใหญ่ ซึ่งอาศัยพื้นฐานของการวัดและการควบคุมเป็นหลัก โดยมีจุดประสงค์เพื่อควบคุมการผลิตให้ได้ผลผลิตสูงสุดและมีคุณภาพดีที่สุด พร้อมกับต้นทุนที่ต่ำ การนำเอาระบบควบคุมอัตโนมัติมาใช้ในกระบวนการผลิตทำให้ลดการใช้แรงงานของผู้ปฏิบัติ ทั้งยังลดความผิดพลาดที่เกิดขึ้นจากการทำงานของผู้ปฏิบัติงานด้วย หลักของระบบควบคุมอัตโนมัติส่วนใหญ่ นั้น จะใช้การป้อนค่ากลับจากค่าที่วัดได้มาใช้ควบคุมกระบวนการ เรียกลักษณะเช่นนี้ว่า การควบคุมแบบป้อนกลับ (Feed Back Control) กล่าวได้ว่าระบบควบคุมอัตโนมัติ คือการควบคุมปริมาณผลผลิตของระบบให้ได้ตามค่าที่ต้องการ โดยอาศัยปริมาณการวัดจากผลผลิตที่ได้แล้ว นำมาเทียบเคียงกับค่าที่ต้องการผ่านการป้อนกลับ

ด้วยเหตุนี้ในปริญญาานิพนธ์จึง ได้ศึกษาวิธีการออกแบบระบบควบคุมแบบ PID เนื่องจากรูปแบบของระบบควบคุมแบบ PID เป็นรูปแบบที่สามารถควบคุมกระบวนการต่างๆ ได้อย่างกว้างขวาง ไม่ว่าจะเป็นการตอบสนองของกระบวนการนั้นจะเร็วหรือช้าอย่างไรก็ตาม ตัวควบคุม PID ก็สามารถใช้ควบคุมกระบวนการของระบบ ได้เป็นอย่างดีแต่ทั้งนี้ทั้งนั้นจะต้องมีการปรับแต่งค่าพารามิเตอร์ ของตัวควบคุม PID ให้เหมาะสมด้วยซึ่งต้องอาศัยประสบการณ์ และความชำนาญต่อกระบวนการที่จะควบคุมเป็นอย่างดีของผู้ปฏิบัติ ซึ่งในบางกรณียังคงต้องใช้วิธีการแบบ ลองผิด-ลองถูก (Trial And Error) ในการปรับแต่งค่าพารามิเตอร์ของตัวควบคุมให้เหมาะสมกับกระบวนการ และ PID นั้นยังสามารถใช้ได้กับกระบวนการอันดับที่สองหรือต่ำกว่า

1.2 วัตถุประสงค์ของปริญญาานิพนธ์

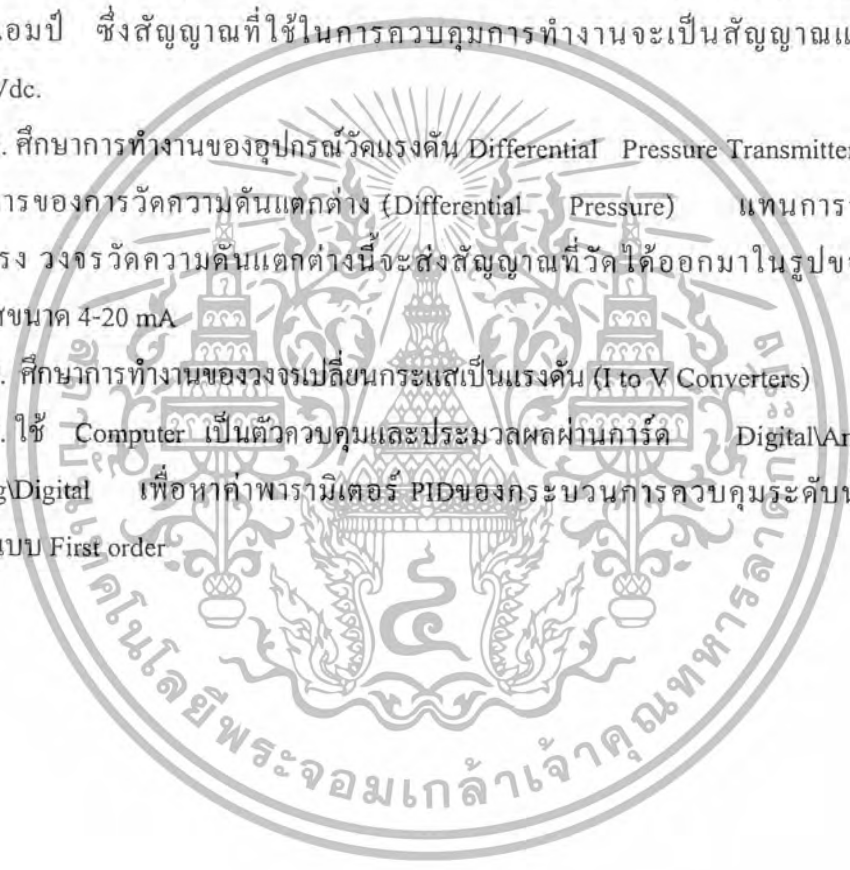
1. เพื่อศึกษาการควบคุมระดับน้ำอัตโนมัติแบบมีโมดูลไร้ผลตอบสนอง
2. ออกแบบและสร้างแบบจำลองการทำงานของ Plant ควบคุมระดับน้ำอัตโนมัติ
3. ศึกษาการทำงานของชุด Signal Converter ในการแปลงสัญญาณ Digital\Analog และ Analog\Digital
4. ศึกษาผลตอบสนองทางเวลาของระบบอันดับที่หนึ่ง Time response for first-order system modem

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ศึกษาทฤษฎีการควบคุมแบบ PID Control และ Ziegler Nichols เพื่อใช้ในการหาค่าพารามิเตอร์ มาเป็นตัวควบคุม (Controller) ของการควบคุมระดับน้ำอัตโนมัติ
6. ศึกษาหลักการการทำงานของ Controller ที่ใช้ควบคุมระดับน้ำ

1.3 ขอบเขตของปริญญานิพนธ์

1. ออกแบบและสร้าง Plant จำลองระดับน้ำ เพื่อใช้ในการควบคุมระดับน้ำ ซึ่งจะทำได้ถึงน้ำ 1 ใบ โดยการควบคุมระดับน้ำนั้นสามารถหน่วงเวลาการไหลของน้ำได้
2. ออกแบบและสร้างวงจรการทำงาน อุปกรณ์ที่ทำหน้าที่ขับเคลื่อนการทำงานของปั้มน้ำ (Motor Driver) โดยประยุกต์ใช้หลักการการทำงานของ วงจรอินเวอร์ตติ้ง (Non-Inverting) ของออปแอมป์ ซึ่งสัญญาณที่ใช้ในการควบคุมการทำงานจะเป็นสัญญาณแรงดันขนาด 0-10 Vdc.
3. ศึกษาการทำงานของอุปกรณ์วัดแรงดัน Differential Pressure Transmitter โดยอาศัยหลักการของการวัดความดันแตกต่าง (Differential Pressure) แทนการวัดระดับน้ำโดยตรง วงจรวัดความดันแตกต่างนี้จะส่งสัญญาณที่วัดได้ออกมาในรูปของสัญญาณกระแสขนาด 4-20 mA
4. ศึกษาการทำงานของวงจรเปลี่ยนกระแสเป็นแรงดัน (I to V Converters)
5. ใช้ Computer เป็นตัวควบคุมและประมวลผลผ่านการ์ด DigitalAnalog และ AnalogDigital เพื่อหาค่าพารามิเตอร์ PIDของกระบวนการควบคุมระดับน้ำอัตโนมัติ ที่เป็นแบบ First order



บทที่ 2

ทฤษฎีการควบคุมกระบวนการ

2.1 ระบบควบคุมกระบวนการ (Process Control System)

2.1.1 ระบบการควบคุมแบบดั้งเดิม (Classical)

- ใช้กับระบบควบคุมที่มี 1 อินพุต 1 เอาท์พุท
- อาจประยุกต์ใช้กับระบบที่มีหลายอินพุต หลายเอาท์พุทได้
- ใช้คณิตศาสตร์ Differential Equation และ Laplace Transform
- วิเคราะห์และออกแบบระบบโดยใช้ฟังก์ชันถ่ายโอน (Transfer Function)
- จะศึกษาเกี่ยวกับเสถียรภาพ การทำงานที่เป็นไปตามที่ต้องการ และความเที่ยงตรงของระบบที่ถูกควบคุม

2.1.2 ระบบควบคุมแบบสมัยใหม่ (Modern Control System)

- วิเคราะห์เห็นและออกแบบระบบโดยใช้ตัวแปรสถานะ (State Variables)
- จะศึกษาและพิจารณาเกี่ยวกับความสามารถในการตรวจสอบ หรือสังเกตสถานะของระบบ (Modern)
- ใช้กับระบบควบคุมที่มี 1 อินพุต 1 เอาท์พุทได้
- เหมาะสมและมีประสิทธิภาพในการใช้กับระบบที่มีหลายอินพุตและหลายเอาท์พุท
- ใช้คณิตศาสตร์ Differential Equation, Vector และ Matrix ถูกควบคุม และสามารถในการควบคุมได้

2.2 การควบคุมกระบวนการแบบอัตโนมัติ (Automatic Control System)

การควบคุมกระบวนการได้มีวิวัฒนาการ และค้นคว้าทางด้านทฤษฎีมาเป็นเวลานาน ในปัจจุบันนี้ ได้แบ่งทฤษฎีของระบบควบคุมสามารถจะแบ่งออกได้เป็น 2 ประเภทใหญ่ ๆ คือ

ระบบควบคุมอัตโนมัตินั้นมีจำเป็นและความสำคัญอย่างยิ่งต่อความก้าวหน้าและ พัฒนาในวิศวกรรมศาสตร์และวิทยาศาสตร์ รวมทั้งในงานอุตสาหกรรมซึ่งจำเป็นจะต้องมีระบบการควบคุมอัตโนมัติเพื่อให้งานควบคุมการผลิต หรืออื่น ๆ ที่เกี่ยวข้อง ให้เป็นไปตามความต้องการได้ ไม่ว่าจะเป็นระบบไฟฟ้า ระบบไฟฟ้าเชิงกล ระบบนิวแมติกส์ ระบบไฮดรอลิกส์ เป็นต้น

เครื่องควบคุมหรือตัวควบคุม เป็นส่วนประกอบที่สำคัญของ ระบบควบคุม แบบอัตโนมัติ

โดยทำหน้าที่ในการสั่งการด้วยการสร้างสัญญาณสำหรับควบคุมกระบวนการต่างๆ จะทำให้

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง จะทำให้

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลตอบสนองเป็นไปตามต้องการ ซึ่งสัญญาณดังกล่าวจะเป็นไปตามกฎ และรูปแบบของกระบวนการควบคุมที่ผู้ควบคุมได้เลือกและกำหนดไว้ล่วงหน้า โดยสัญญาณควบคุมนั้นจะขึ้นอยู่กับค่าความคลาดเคลื่อนของการควบคุม (คือค่าแตกต่างระหว่างค่าเป้าหมายและตัวแปรกระบวนการนั่นเอง)

ปัจจุบันเครื่องควบคุมสามารถแบ่งตามโครงสร้างการทำงานได้ 3 ชนิด คือ เครื่องควบคุมแบบนิวแมติกส์ที่ทำงานโดยใช้สัญญาณลม เครื่องควบคุมแบบอิเล็กทรอนิกส์ทำงานโดยใช้วงจรอิเล็กทรอนิกส์เชิงเส้น (Linear Circuit) ในการสร้างสัญญาณควบคุม และเครื่องควบคุมแบบดิจิทัลทำงานโดยใช้วงจรตรรกะ (Logic Circuit) หรือไมโครโปรเซสเซอร์ สร้างสัญญาณควบคุมโดยเครื่องควบคุมที่ถูกนำมาใช้และเป็นที่รู้จักกันดีมากที่สุดในการอุตสาหกรรมการผลิตก็คือ ตัวควบคุมแบบ PID

โดยส่วนใหญ่การควบคุมในอุตสาหกรรมการผลิต จะแบ่งเป็นการควบคุมแบบป้อนกลับ กับแบบ Sequence แต่ในที่นี้จะขอกล่าวเฉพาะที่เป็นแบบป้อนกลับเท่านั้นและระบบการควบคุมที่เป็นแบบป้อนกลับก็จะเป็นตัวควบคุมแบบ PID ซึ่งตัวควบคุมแบบนี้มีฟังก์ชันถ่ายโอน คือ โครงสร้างพื้นฐานของตัวควบคุม PID ที่ใช้ในอุตสาหกรรมแสดงดังรูปที่ 2.4 โดยที่

รูปแบบของ Ziegler-Nichols ได้กำหนดค่าของ K_c , T_i และ T_d สำหรับตัวควบคุมแบบต่าง ๆ ที่ใช้การปรับค่าพารามิเตอร์ของตัวควบคุม PID ตามวิธี Process Reaction Curve ได้ตามตารางที่ 2.0

คุณลักษณะที่สำคัญของผลตอบสนองต่อเวลาที่เป็นเส้นโค้งในรูปของ Exponential Curve คือ เมื่อ $t = T$ ค่าของผลตอบสนองต่อเวลาจะมีค่าเท่ากับ 63.2% ของค่าสุดท้ายที่สภาวะคงที่

สิ่งที่พึงสังเกตอีกประการหนึ่งคือ ถ้าค่าของ T ซึ่งเป็นค่าเวลาคงที่ของระบบ มีค่าน้อย ผลตอบสนองจะเร็ว และความชันที่สภาวะเริ่มแรกของผลตอบสนองจะหาได้จากสมการ

ซึ่งการหาค่าตัวแปรเหล่านี้สามารถหาได้โดยใช้วิธีการทดสอบกระบวนการแบบวงเปิด

ผลตอบสนองของกระบวนการจะมีลักษณะเป็นรูปตัว S ดังในรูปที่ 2.5 ดังนั้นการทดสอบนี้สามารถจะกระทำกับกระบวนการอันดับสองหรือกระบวนการอันดับสูงที่มีอัตราความหน่วงมากกว่าหรือเท่ากับหนึ่งด้วย เนื่องจากมีผลตอบสนองเป็นรูปตัว S ด้วยเช่นกัน

จากสมการที่ 2.1 ทำให้สามารถคำนวณหาอัตราขยาย ณ.สภาวะคงตัวของกระบวนการได้ดังนี้

ซึ่งเรียกสมการที่ 2.1 นี้ว่า “รูปแบบสังเคราะห์ตัวควบคุม (Controller Synthesis Function)”

ผลตอบสนองระบบป้อนกลับที่ง่ายที่สุดคือ ผลตอบสนองแบบอันดับหนึ่งซึ่งมีทรานเฟอ์ฟังก์ชันคือ

จากสมการเราจะพบว่าตัวควบคุมที่มีการทำงาน ในแบบอินทิกรัลจึงทำให้ผลลัพธ์ที่ได้จากสมการไม่มีออฟเซต (Offset) ตามคุณสมบัติของการควบคุมแบบอินทิกรัล แต่ในระบบอันดับสองและอันดับที่สูงกว่านี้ ถ้าระบบประกอบด้วยค่าหน่วงเวลา (Dead Time) ผลที่ได้ของระบบป้อนกลับก็จะมีพจน์ของการหน่วงเวลาปรากฏอยู่ด้วย

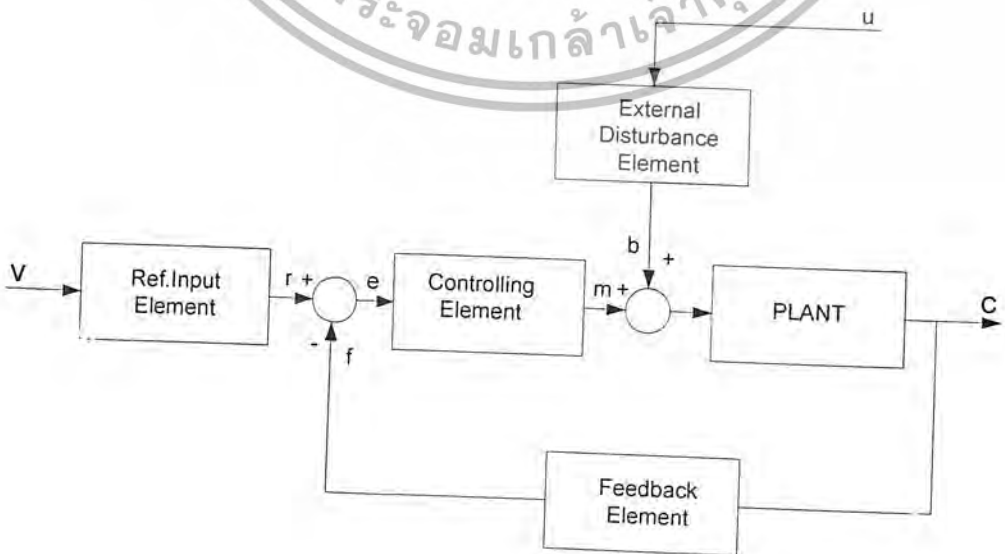
ดังนั้นจากสมการ และสมการ จะได้ฟังก์ชันถ่ายโอนสำหรับสังเคราะห์ค่าพารามิเตอร์ของตัวควบคุมตามกระบวนการแบบ FOPDT คือ

จากสมการแสดงให้เห็นว่า จากกระบวนการที่เป็นแบบ FOPDT เมื่อนำค่ามาหาฟังก์ชันถ่ายโอนของตัวควบคุมจะได้เป็นตัวควบคุมแบบ เท่านั้น โดยมีค่าพารามิเตอร์คือ

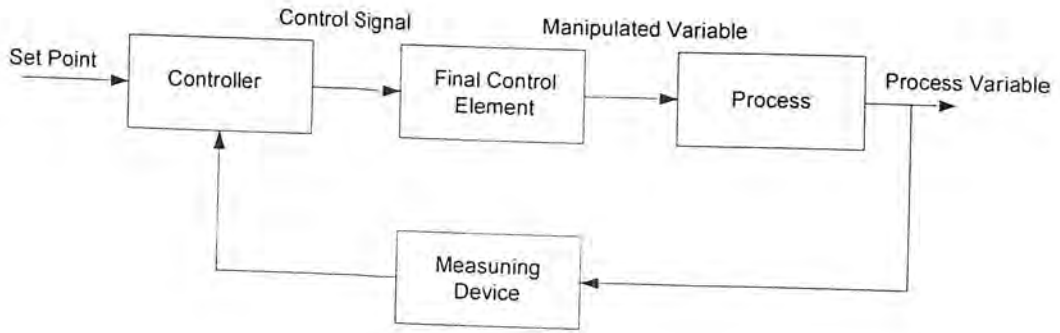
การควบคุมกระบวนการแบบอัตโนมัติ นั้น จะเป็นเรื่องเกี่ยวข้องกับค่าตัวแปรต่าง ๆ ในกระบวนการ (Process Variable) เช่น อุณหภูมิ (Temperature) ความดัน (Pressure) อัตราการไหล (Flows) และตำแหน่งต่าง ๆ (Composition) ในการควบคุมกระบวนการให้เป็นผลสำเร็จได้นั้น เราต้องหาค่าความคลาดเคลื่อน (Error) โดยการวัดค่าตัวแปรเหล่านี้ แล้วนำมาเปรียบเทียบกับค่าที่ตั้งไว้ (Set Point) แล้วตัดสินใจว่าจะทำอะไรกับค่าความแตกต่างนี้ เพื่อให้ได้ค่าที่เราต้องการ ซึ่งการทำงานเหล่านี้ สามารถทำได้โดยใช้ผู้ปฏิบัติการ (Operator) แต่สนใจงานส่วนใหญ่แล้วจะมีตัวแปรต่าง ๆ เหล่านี้มากมาย ทำให้ต้องใช้คนมากมายตามไปด้วย ดังนั้นการใช้เครื่องมือที่มีความสามารถควบคุมกระบวนการ ได้โดยอัตโนมัติ จึงเป็นวิธีที่ดีกว่า

2.2.1 โครงสร้างพื้นฐานของระบบควบคุมอัตโนมัติ

ในการศึกษา วิเคราะห์และออกแบบระบบควบคุม โดยการใช้ทฤษฎีของระบบควบคุมแบบดั้งเดิมนั้น จะจัดรูปแบบของระบบควบคุมอัตโนมัติไว้ดังแสดงในรูปข้างล่างนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 แสดงรูปแบบการควบคุมแบบอัตโนมัติ (Automatic Control System Configuration)

2.3 คำจำกัดความ

2.3.1 อินพุต (Input)

บางครั้งเรียกว่า อินพุตอ้างอิง (Reference Input) หรือค่าที่ตั้งไว้ (Set point) หมายถึงค่าหรือผลตอบสนองที่ต้องการของระบบที่ต้องการควบคุมที่กำหนดไว้ เช่น ต้องการควบคุมอุณหภูมิของเตาเผาให้อยู่ที่ 700 องศาเซลเซียส ดังนั้นค่าอุณหภูมิ 700 องศาเซลเซียส (หรือค่าของสัญญาณในรูปแบบอื่น ๆ ที่สอดคล้องกับค่าอุณหภูมิ 700 องศาเซลเซียส) จะเป็นอินพุตอ้างอิงของระบบที่ถูกรับควบคุม

2.3.2 ตัวควบคุม (Controller)

หมายถึง เครื่องมือหรืออุปกรณ์ที่ใช้ในการสร้างสัญญาณควบคุม เพื่อทำหน้าที่ควบคุมให้ระบบหรือกระบวนการที่ต้องการควบคุม มีเอาต์พุตหรือผลตอบสนองตามต้องการ ตัวควบคุมจะมีหลายแบบ เช่น ตัวควบคุมแบบ ON-OFF (ตัวควบคุมแบบสองตำแหน่ง) ตัวควบคุมแบบ Proportional (P) ตัวควบคุมแบบ Integral (I) ตัวควบคุมแบบ Derivative (D) หรือการใช้ตัวควบคุมหลาย ๆ แบบรวมกัน เช่น ตัวควบคุมแบบ PI ตัวควบคุมแบบ PD ตัวควบคุมแบบ PID เป็นต้น นอกจากนี้ ยังมีการจัดหรือกำหนดให้ตัวควบคุมทำงานลักษณะพิเศษ เช่น ตัวควบคุมแบบ Duplex (ตัวควบคุมที่มี 1 อินพุต และมี 2 เอาต์พุต) ตัวควบคุมแบบ Auto Selector (ตัวควบคุมที่มีอินพุตมากกว่า 2 อินพุต แต่จะมีเพียง 1 เอาต์พุต) ตัวควบคุมแบบ Ratio ตัวควบคุมแบบ Cascade ตัวควบคุมแบบ Feedforward เป็นต้น

2.3.3 กระบวนการ (Plant or Process)

หมายถึงระบบหรือกระบวนการที่ถูกควบคุม หรืออาจจะเป็นวัตถุทางกายภาพที่ถูกควบคุมก็ได้ เช่น กระบวนการเกี่ยวกับระดับของของเหลว กระบวนการเกี่ยวกับการควบคุมอุณหภูมิในห้อง ในเตาเผา เครื่องจักรที่ใช้ในอุตสาหกรรม เป็นต้น

2.3.4 เอาท์พุท (Output)

หมายถึงผลตอบสนองของระบบหรือกระบวนการที่ถูกควบคุม ซึ่งโดยทั่วไปแล้วต้องการจะควบคุมให้อาท์พุทที่กำหนด (หรือตามค่าของอินพุทที่เปลี่ยนแปลงไป) หรือมีค่าคงเดิมได้เมื่อมีการรบกวนทั้งภายในและภายนอกมากระทำต่อระบบที่ถูกควบคุม

2.3.5 การรบกวน (Disturbance)

หมายถึงสัญญาณรบกวนที่อาจจะเกิดขึ้นในระบบที่ถูกควบคุม สัญญาณรบกวนนี้อาจจะเกิดขึ้นที่จุดใด ๆ ในระบบก็ได้ เช่น เกิดขึ้นที่กระบวนการ เกิดขึ้นที่อุปกรณ์วัด เป็นต้น การเกิดขึ้นของสัญญาณรบกวนอาจจะเกิดขึ้นในเวลาใด ๆ ทั้งที่คาดเดาได้และคาดเดาไม่ได้ การรบกวนนี้แบ่งได้เป็น 2 ลักษณะ คือ

- การรบกวนภายใน (Internal Disturbance) เป็นการรบกวนซึ่งอาจจะทำให้มีการเกิดการเปลี่ยนแปลงของค่าพารามิเตอร์ต่าง ๆ ของอุปกรณ์ ที่ใช้ในระบบ
- การรบกวนภายนอก (External Disturbance) เป็นการรบกวนที่เกิดขึ้นจากภายนอกระบบ แต่มีผลต่อระบบที่กำลังถูกควบคุมอยู่ โดยทั่วไป จะถือว่าการรบกวนภายนอกเป็นอินพุทอย่างหนึ่ง (ที่ไม่พึงประสงค์) ของระบบควบคุม

2.3.6 อุปกรณ์วัด (Measuring Instruments)

หมายถึงอุปกรณ์ที่อาจจะได้แก่ เซนเซอร์ (Sensor) ทรานสดิวเซอร์ (Transducer) หรืออุปกรณ์แปลงหรือวัดสัญญาณอื่น ๆ ที่ทำหน้าที่วัดค่าของเอาท์พุทของระบบที่ถูกควบคุม แล้วป้อนกลับไปเพื่อทำการเปรียบเทียบกับอินพุทของระบบ

2.3.7 ระบบ (System)

หมายถึงการนำเอาอุปกรณ์ต่าง ๆ ที่สามารถทำงานร่วมกันได้ มารวบรวมเข้าด้วยกันเพื่อให้ทำงานอย่างใดอย่างหนึ่งที่ต้องการ เช่นระบบทางกายภาพ (Physical System) ของวงจรไฟฟ้า ระบบเชิงกล เป็นต้น

2.3.8 ระบบควบคุม (Control System)

หมายถึงระบบควบคุมที่ต้องการควบคุมให้เอาต์พุตหรือผลตอบสนองของระบบมีค่าตามที่ ต้องการ โดยแบ่งออกเป็น 2 แบบง่าย ๆ คือ ระบบควบคุมแบบเปิด (Open Loop) และระบบควบคุม แบบปิด (Close Loop)

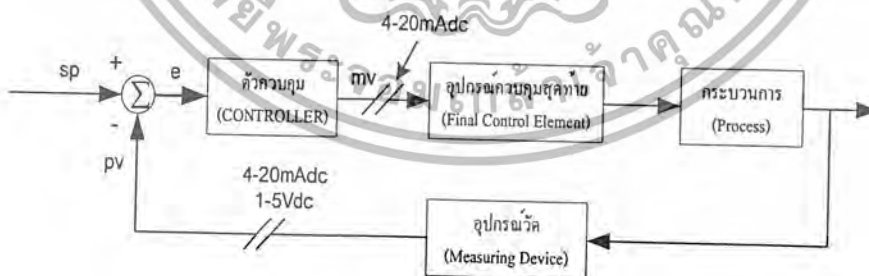
1. ระบบควบคุมแบบเปิด (Open Loop Control System)



รูปที่ 2.1 แสดงบล็อกไดอะแกรมของระบบควบคุมแบบวงเปิด (Open Loop Control System)

จากรูปที่ 2.1 เป็นระบบควบคุมที่เอาต์พุตของระบบจะไม่มีการนำเอาต์พุตมาเปรียบเทียบกับ อินพุต (ไม่มีการป้อนกลับ) โดยเอาต์พุตที่ได้ไม่มีผลโดยตรงต่อการควบคุมไม่สามารถ ควบคุมเอาต์พุตหรือผลตอบสนองของระบบ จะเป็นไปตามต้องการได้ตลอดเวลา หรือเมื่อมีการ เปลี่ยนแปลงเกิดขึ้น แล้วทำให้เกิดการรบกวนไม่สามารถจัดผลของการรบกวนไปได้แต่ต้องใช้ วิธีปรับ อินพุต ความเที่ยงตรงของเอาต์พุตขึ้นอยู่กับอุปกรณ์ที่ใช้ มักจะใช้กับระบบที่ไม่ต้องการ ความเที่ยงตรงสูง และใช้งานง่าย ๆ เช่น พัดลม เตาวิค เครื่องซักผ้า ระบบควบคุมไฟจราจร แบบเดิม เป็นต้น

2. ระบบควบคุมแบบปิดหรือป้อนกลับ (Close Loop or Feedback Control System)



รูปที่ 2.2 แสดงการควบคุมแบบปิด (Close Loop Control System)

จากรูปที่ 2.2 อุปกรณ์หรือเครื่องวัดจะวัดค่าตัวแปรกระบวนการ (pv) เช่น ความดัน อัตรา การไหล และระดับของของเหลว เป็นต้น เพื่อเปรียบเทียบกับค่าอ้างอิงหรือค่าเป้าหมาย (Set Point : sp) จากนั้นตัวควบคุมจะนำค่าความคลาดเคลื่อน (Error : e) ในการควบคุมมาใช้ในการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เฉพาะ P, PI, PD, หรือ PID นั้นขึ้นอยู่กับชนิดและคุณลักษณะของกระบวนการ (ซึ่งอาจจะวิเคราะห์ได้จาก การเปลี่ยนแปลงของ p_v ด้วย)

นั่นคือเป็นระบบควบคุมที่ต้องการควบคุมให้อาต์พุท หรือผลตอบสนองของระบบมีค่าตามที่ต้องการ โดยกานนำสัญญาณเอาต์พุทป้อนกลับมาเปรียบเทียบกับสัญญาณอินพุท ซึ่งการป้อนกลับนี้จะต้องเป็นการป้อนกลับแบบลบ (Negative Feedback) จากนั้นนำค่าความแตกต่างระหว่างอินพุทกับเอาต์พุท ส่งต่อไปยังส่วนสร้างสัญญาณของตัวควบคุม เพื่อสร้างสัญญาณควบคุมที่เหมาะสมที่จะทำให้เอาต์พุทของระบบที่ถูกควบคุมมีค่าตามต้องการ ระบบควบคุมป้อนกลับนี้ อาจจะเรียกอีกอย่างหนึ่งว่า ระบบควบคุมแบบลูปปิด หรือระบบควบคุมอัตโนมัติก็ได้

- มีการนำเอาต์พุทมาเปรียบเทียบกับอินพุท (การป้อนกลับ) ดังนั้นเอาต์พุทจะมีผลโดยตรงต่อการควบคุม
- สามารถควบคุมเอาต์พุท หรือผลตอบสนองของระบบควบคุมให้เป็นไปตามต้องการได้ตลอดเวลา หรือเมื่อมีการเปลี่ยนแปลงเกิดขึ้น
- สามารถจัดหรือลดผลของการรบกวนได้ ไม่ว่าจะเป็นการรบกวนจากภายใน หรือการรบกวนจากภายนอก ทั้งนี้เพราะมีการป้อนกลับ
- อุปกรณ์ที่ใช้ไม่จำเป็นต้องมีความเที่ยงตรงสูง
- ต้องระวังปัญหาเรื่องเสถียรภาพอันเกิดจากผลของการป้อนกลับ
- ระบบควบคุมแบบลูปปิด จะใช้กับระบบที่ต้องการความเที่ยงตรงสูง และใช้ในงานทั่วไปหรือในงานอุตสาหกรรม เช่น การควบคุมอุณหภูมิห้อง การควบคุมอุณหภูมิของเตาเผา ระบบควบคุมเชิงกล ระบบควบคุมกระบวนการทางอุตสาหกรรม เป็นต้น

2.4 การออกแบบระบบควบคุมในโดเมนเวลา

2.4.1 ขั้นตอนการออกแบบระบบควบคุม

ขั้นตอนของการออกแบบและชดเชย (Compensate) ระบบควบคุมเชิงเส้นที่ไม่แปรตามเวลาแบบหนึ่งอินพุท-หนึ่งเอาต์พุท ให้มีคุณสมบัติและสมรรถนะตามต้องการ จะใช้วิธีของทางเดินของรากวิเคราะห์และออกแบบระบบในโดเมนเวลา และใช้วิธีของ Bode Plot, Nyquist Plot ในโดเมนความถี่อย่างไรก็ตาม ในที่นี้ จะทำการศึกษาและออกแบบระบบในโดเมนเวลาเท่านั้น ซึ่งโดยทั่วไป ขั้นตอนการออกแบบและชดเชยเพื่อให้ระบบมีสมรรถนะตามต้องการจะเป็นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ศึกษาและวิเคราะห์ถึงคุณสมบัติแบบพฤติกรรมของระบบนั้น ๆ ว่าเป็นอย่างไร ผลตอบสนองเวลาต่ออินพุทแบบต่าง ๆ มีเสถียรภาพสัมพัทธ์ดีหรือไม่ มีความเที่ยงตรงหรือไม่ และในการใช้งานนั้น ต้องการให้มีผลตอบสนองเวลาเป็นอย่างไร

2. เลือกข้อกำหนด (Specification) เพื่อให้ระบบมีสมรรถนะตามต้องการเลือกข้อกำหนดที่เหมาะสมซึ่งเป็นที่จำเป็นอย่างยิ่งในการออกแบบข้อกำหนดต่าง ๆ จะได้แก่ ความเที่ยงตรง เสถียรภาพสัมพัทธ์ ความเร็วของผลตอบสนองเวลา เป็นต้น ในโดเมนเวลานั้น ค่าความคลาดเคลื่อนที่ภาวะคงที่จะแสดงถึงความเที่ยงตรงของระบบ ในขณะที่อัตราการหน่วงและ Maximum Overshoot จะแสดงถึงเสถียรภาพสัมพัทธ์ของระบบ สำหรับความเร็วของผลตอบสนองเวลานั้น จะพิจารณาได้จาก Rise Time, Setting Time, และ Undamped Natural Frequency (ในโดเมนความถี่ Peak Resonance และ Phase Margin จะใช้วัดเสถียรภาพสัมพัทธ์ของระบบ ในขณะที่ Resonance Frequency และ Bandwidth จะใช้วัดความเร็วของผลตอบสนองของระบบ

3. การออกแบบและชดเชย เมื่อทราบข้อกำหนดต่าง ๆ แล้วขั้นต่อไปจะต้องทำให้ระบบมีคุณสมบัติและสมรรถนะสอดคล้องกับข้อกำหนด ในทางปฏิบัติทั่วไปจะใช้วิธีการลองผิดลองถูก (Trial-and-Error) ปรับค่าพารามิเตอร์ของระบบเพื่อให้ระบบมีคุณสมบัติสอดคล้องกับข้อกำหนดที่เลือกไว้เท่าที่จะเป็นไปได้ การปรับค่าอัตราขยาย (Gain) ของระบบจะเป็นขั้นตอนแรกที่จะทำให้ระบบมีสมรรถนะตามต้องการ อย่างไรก็ตาม การปรับค่าอัตราขยายเพียงอย่างเดียวจะไม่สามารถทำให้พฤติกรรมทางพลศาสตร์ และความเที่ยงตรงจะสอดคล้องกับข้อกำหนดที่ต้องการทั้งหมดได้ เช่น เมื่อเราเพิ่มค่าอัตราขยายให้มากขึ้น จะทำให้ค่าความคลาดเคลื่อนที่ภาวะคงที่ลดลง แต่ในทางตรงกันข้ามจะทำให้เสถียรสัมพัทธ์ของระบบเลวลง หรืออาจจะทำให้ระบบนั้นไม่มีเสถียรภาพก็ได้ จึงจำเป็นจะต้องมีการออกแบบระบบใหม่โดยอาจจะเปลี่ยนโครงสร้างของระบบ หรือเพิ่มอุปกรณ์เข้าไปในระบบเพื่อให้ระบบนั้นมีสมรรถนะตามต้องการ อุปกรณ์ที่จะเพิ่มเข้าไปในระบบนั้นได้แก่ ตัวควบคุม (Controller) หรือตัวชดเชย (Compensator) ซึ่งอาจจะเพิ่มเข้าไปในระบบนั้นได้แก่ ตัวควบคุมเชิงกล Pneumatic หรืออื่น ๆ ทั้งนี้ขึ้นอยู่กับโครงสร้างของระบบที่จะทำการออกแบบ

4. การเพิ่มตัวควบคุมหรือตัวชดเชยเข้าไปในระบบ ถ้าเพิ่มตัวควบคุมหรือตัวชดเชย $G(s)$ เข้าไปในระบบ โดยต่ออนุกรมกันกับระบบที่ถูกควบคุมที่มีฟังก์ชันถ่ายโอน $G(s)$ {หรือ $G(p)$ } จะเรียกว่าเป็นการชดเชยแบบอนุกรม (Series) หรือ Cascade ถ้าต่ออยู่ในลักษณะป้อนกลับจะเรียกว่าการชดเชยแบบขนาน (Parallel) หรือแบบป้อนกลับ หรืออาจจะต่อในลักษณะของการชดเชยแบบ Forward หรือต่อหลาย ๆ แบบร่วมกันก็ได้ การเพิ่มตัวควบคุมหรือตัวชดเชยเข้าไปในระบบนั้นจะต้องพิจารณาว่าจะเพิ่มเข้าไปแบบอนุกรมหรือแบบขนาน โดยพิจารณาจากธรรมชาติของสัญญาณในระบบ ระดับของสัญญาณที่จุดต่าง ๆ อุปกรณ์ที่เราสามารถจัดหาได้ และสเปกตรัมและอื่น ๆ

โดยทั่วไปแล้ว การเพิ่มตัวควบคุมหรือตัวชดเชยแบบอนุกรมจะทำให้ได้ง่ายกว่าแบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำออกจำหน่ายหรือเผยแพร่โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เมื่อต้องเพิ่มตัวควบคุม หรือตัวชดเชยใส่เข้าไปในระบบ ผู้ออกแบบก็จะต้องสามารถหาอุปกรณ์ทางกายภาพที่มีฟังก์ชันถ่ายโอน เหมือนกับฟังก์ชันถ่ายโอนของตัวควบคุมหรือตัวชดเชยที่ต้องการได้ ที่นิยมใช้กันมาก คือ

1. ตัวควบคุมแบบ Proportional (P)
2. ตัวควบคุมแบบ Proportional (P) + Derivative (PD)
3. ตัวควบคุมแบบ Proportional (P) + integral (PI)
4. ตัวควบคุมแบบ Proportional (P) + Integral + Derivative (PID)
5. ตัวควบคุมแบบ Lead
6. ตัวควบคุมแบบ Lag
7. ตัวควบคุมแบบ Lag - Lead
8. อื่น ๆ

6. วิเคราะห์ระบบโดยรวมในโดเมนเวลา (และ/หรือ ในโดเมนความถี่) เพื่อตรวจสอบให้แน่ใจว่าให้ระบบที่ถูกออกแบบนั้นบรรลุนะตามต้องการ

2.5 ความรู้เบื้องต้นเกี่ยวกับการควบคุมแบบป้อนกลับด้วยตัวควบคุม PID

2.5.1 เครื่องควบคุม

2.5.2 หลักการของตัวควบคุมกระบวนการแบบ PID

เป็นที่ทราบกันดีว่า ระบบการควบคุมอัตโนมัติแบบ PID ได้ถูกนำมาควบคุมกระบวนการในการผลิตเป็นเวลานานกว่า 50 ปี มาแล้ว อุตสาหกรรมการผลิตแบบอัตโนมัติที่ทั่วไปในปัจจุบันก็ยังนิยมใช้ตัวควบคุมแบบ PID อยู่ทั้งนี้อาจเป็นจำเพาะรูปแบบของตัวควบคุมเป็นรูปแบบที่สามารถควบคุมกระบวนการต่างๆ ได้อย่างกว้างขวางไม่ว่ากระบวนการนั้นจะมีผลตอบแทนความถี่ต่ำ และความถี่กลาง หรือความถี่สูง ตัวควบคุมแบบ PID ก็สามารถควบคุมกระบวนการได้เป็นอย่างดี ถ้ามีการปรับค่าพารามิเตอร์ของตัวควบคุมอย่างเหมาะสม

อุตสาหกรรมในประเทศไทยนั้นได้รับการนำระบบควบคุมแบบ PID มาใช้ แต่ในการใช้งานนั้น มักจะใช้เพียงแต่การควบคุมแบบ P หรือการควบคุมแบบ PD เท่านั้น ทั้งนี้เนื่องจากการหาค่าพารามิเตอร์แบบ PID ต้องอาศัยผู้เชี่ยวชาญ หรือผู้ที่มีความชำนาญกับกระบวนการเป็นอย่างมาก ค่าของพารามิเตอร์ที่ได้ถึงจะเป็นค่าที่กระบวนการได้อย่างมีประสิทธิภาพ ในกระบวนการผลิตที่ทำงานโดยอัตโนมัติทั่วไป จำเป็นต้องมีตัวควบคุมกระบวนการซึ่งการควบคุมกระบวนการต่างๆ ก็มีมากมายหลายแบบ ตั้งแต่การควบคุมกระบวนการอย่างง่าย ๆ (Single-Loop-Control) จนถึงการ

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนในรายวิชาการควบคุมกระบวนการแบบอัตโนมัติ โดยไม่มีการนำเอกสารนี้ไปเผยแพร่หรือใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

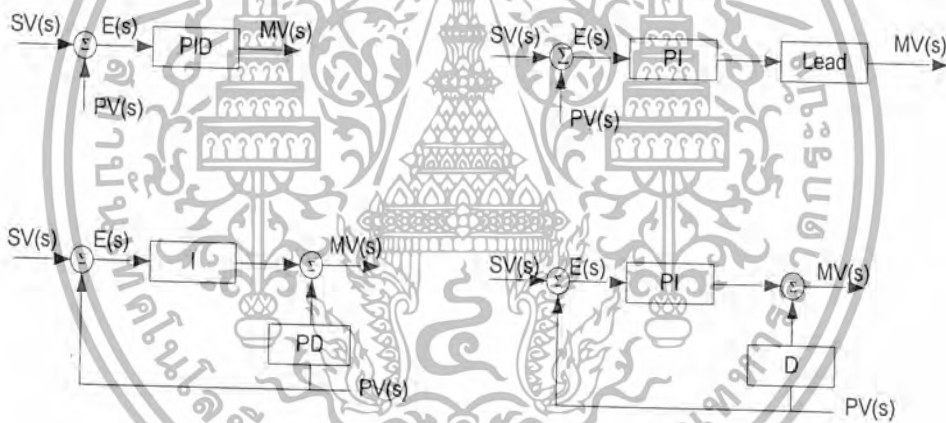
พบว่าประกอบไปด้วยตัวควบคุมกระบวนการแบบ PID แทบทั้งสิ้น น้อยครั้งจะพบตัวควบคุมกระบวนการแบบอื่นๆ

ตัวควบคุมแบบ PID คืออะไร

ตัวควบคุมกระบวนการแบบ PID เป็น ตัวควบคุมชนิดหนึ่งใช้ในการควบคุมแบบป้อนกลับ (Feedback Control) ทำหน้าที่นำสัญญาณเอาต์พุตของกระบวนการมาเปรียบเทียบกับสัญญาณจุดปรับตั้งที่กำหนด และคำนวณตามกฎเกณฑ์การควบคุมแบบ PID ซึ่งประกอบด้วยการคำนวณ 3 ส่วนคือ Proportional, Integral และ Derivative สัญญาณควบคุมที่ได้จากตัวควบคุมกระบวนการแบบ PID ก็คือ ผลรวมของการคำนวณทั้ง 3 ส่วนที่ได้กล่าวมาแล้วนั่นเอง

2.5.3 รูปแบบของตัวควบคุมแบบ PID

ตัวควบคุมแบบ PID จะแบ่งรูปแบบของตัวควบคุมตามกฎเกณฑ์ของการควบคุมออกได้เป็น 4 แบบตามรูปที่ 2.3



รูปที่ 2.4 แสดงรูป ก, ข, ค และ ง แสดงรูปแบบของตัวควบคุมแบบ PID

- SV(s) - จุดปรับตั้ง (Set Point หรือ Set Valve)
- PV(s) - สัญญาณตัวแปรกระบวนการ (Process Variable)
- E(s) - ผลต่างระหว่างจุดปรับตั้งและสัญญาณป้อนกลับมีค่าเท่ากับ $SV(s) - PV(s)$ หรือ $PV(s) - SV(s)$
- M(s) - สัญญาณตัวแปรกระบวนการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Gc(s) = Kp \left(1 + \frac{1}{T_i s} + T_d s \right)$$

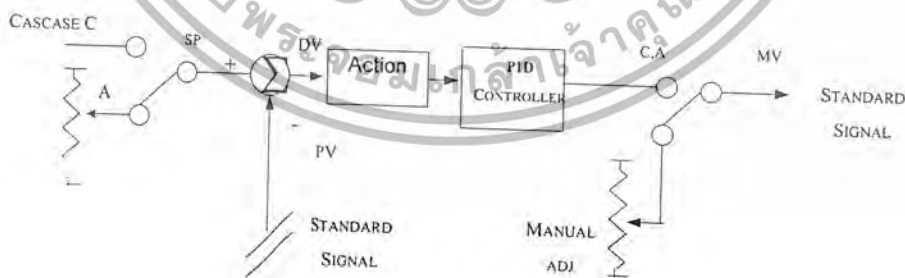
- โดยที่ K_c = อัตราขยายของตัวควบคุม
- T_i = ค่า Integral or Reset Time (วินาที)
- T_d = ค่า Derivative or rate Time (วินาที)

ค่าพารามิเตอร์ของตัวควบคุม PID ทั้ง 3 ค่านี้ จะใช้สำหรับปรับให้กับตัวควบคุมเพื่อควบคุมกระบวนการที่ต้องการให้มีผลตอบสนองเป็นไปตามต้องการ ดังนั้นจะเห็นได้ว่าสัญญาณควบคุมหรือตัวแปรปรับกระบวนการ (mv) ที่ได้จากตัวควบคุม PID จะถูกกำหนดด้วยความสัมพันธ์ระหว่างสัญญาณเข้าตัวควบคุมหรือตัวแปรกระบวนการ (pv) กับสัญญาณอ้างอิงหรือค่าเป้าหมาย (sp) โดยที่ความสัมพันธ์ดังกล่าวจะขึ้นอยู่กับกฎเกณฑ์ของการควบคุมที่พนักงานหรือผู้ควบคุมได้ปรับแต่งไว้ล่วงหน้า (คือการปรับแต่งค่าพารามิเตอร์ K_c , T_i และ T_d ของตัวควบคุม PID นั้นเอง) ดังนั้นสัญญาณควบคุมของตัวควบคุม PID คือ

$$mv = \frac{100}{PB} \left[(sp - pv) + \frac{1}{T_i} \int_0^t (sp - pv) dt + T_d * \frac{d(sp - pv)}{dt} \right] + b$$

หรือ

$$mv = Kc \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d * \frac{de(t)}{dt} \right] + b$$



รูปที่ 2.5 โครงสร้างพื้นฐานของตัวควบคุม PID

- pv : Process Variable Value ค่าสัญญาณจากเครื่องมือวัด
- DV : Deviation หรือค่าความคลาดเคลื่อน (Error)
- sp : Setpoint ค่าเป้าหมาย
- mv : Manipulated Value

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C : Cascade หรือ Remote Set Mode
M : Manual Mode

A : Automated Value

2.6 กริยาต่าง ๆ ของการควบคุม

2.6.1 การควบคุมแบบ Proportional

การทำงานของเครื่องควบคุมที่จะเปิดหรือปิด (สัญญาณควบคุม) วาล์วควบคุมนั้นจะเป็นปฏิกิริยากับค่าเบี่ยงเบน โดยการควบคุมแบบนี้จะเหมาะนี้จะเหมาะกับกระบวนการที่มี Time Lag ไม่สูงมากแต่การควบคุมแบบนี้มีจุดอ่อนก็คือการเกิด Offset

2.6.2 การควบคุมแบบ Proportional + Integral

การเพิ่ม Integral เข้าไปก็เพื่อลดการเกิด Offset นั้นเองการควบคุมแบบนี้นิยมใช้ในการควบคุมความดัน ระดับ ของไหล แต่ถ้าเราปรับค่า Integral มากไปก็จะทำให้การควบคุมเกิดการแกว่ง (Hunting) ขึ้นได้

2.6.3 การควบคุมแบบ Proportional + Integral + Derivative

การควบคุมแบบนี้จะเหมาะกับกระบวนการที่มี Time Delay มาก ๆ เช่นการควบคุมอุณหภูมิเป็นต้นแต่ถ้าเราปรับค่า Derivative และ Integral ไม่เหมาะสมก็จะเกิดการแกว่งเช่นเดียวกัน

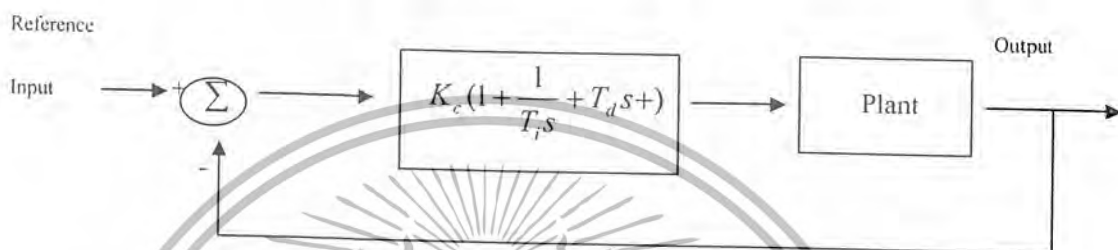
จากที่กล่าวมาในหัวข้อนี้ การปรับค่าพารามิเตอร์ต่างๆเหล่านี้จึงต้องการวิศวกรที่มีความเข้าใจทั้งภาคทฤษฎีและภาคปฏิบัติสูงเป็นผู้ปรับค่าต่าง ๆ เหล่านี้ จึงจะทำให้การควบคุมเป็นไปตามความต้องการ

2.7 การปรับแต่งค่าพารามิเตอร์ของเครื่องควบคุม

การปรับค่าพารามิเตอร์ทั้ง 3 ค่า ของตัวควบคุม PID เป็นสิ่งจำเป็นเนื่องจากผลตอบสนองของกระบวนการที่ถูกควบคุมจะเป็นเช่นไรนั้นจะขึ้นอยู่กับการปรับแต่งตัวควบคุม ซึ่งการปรับแต่งค่าพารามิเตอร์ของตัวควบคุม PID สามารถหาได้ทั้งทางคณิตศาสตร์และทางปฏิบัติแต่ส่วนใหญ่ในโรงงานอุตสาหกรรมจะนิยมใช้วิธีการทางปฏิบัติมากกว่า เนื่องจากการหาพารามิเตอร์ทางคณิตศาสตร์จำเป็นจะต้องรู้ตัวแปรทั้งหมดในกระบวนการซึ่งเป็นเรื่องที่ยุ่งยากมาก

การปรับแต่งหาค่าพารามิเตอร์ของตัวควบคุม PID ที่นิยมใช้ในทางปฏิบัติมีอยู่ 3 วิธีด้วยกันคือ

- วิธี Process Reaction Curve Method ของ Ziegler – Nichols (Open Loop Method)
- วิธี Ultimate Method ของของ Ziegler – Nichols (Closed Loop)
- วิธี Trial and Error (วิธีการลองผิดลองถูก)



รูปที่ 2.6 แสดงโครงสร้างของระบบที่ควบคุมด้วยตัวควบคุม PID

2.7.1 วิธีการแบบ Reaction Curve (Open Loop Method)

วิธีนี้จะหาค่าพารามิเตอร์ของตัวควบคุม PID จากผลตอบสนองเวลาของระบบหรือกระบวนการที่ถูกควบคุมแบบเปิดคืออินพุตแบบ Step โดยที่ระบบหรือกระบวนการในกรณีนี้ จะไม่มีโพล (Pole) ที่จุดกำเนิด (Origin) หรือไม่มี Dominant Complex Conjugate Poles และ ไม่มีตัวควบคุมต่อร่วมอยู่ ดังนั้นผลตอบสนองของเวลาจะเป็นรูปตัว S (ถ้าผลตอบสนองเวลาไม่เป็นรูปตัว S วิธีนี้จะใช้ไม่ได้) ผลตอบสนองของเวลารูปตัว S นี้จะถูกนำมาอธิบายคุณลักษณะของกระบวนการ (Process Characteristic) ด้วยพารามิเตอร์ 3 ค่า คือ ค่าอัตราขยายของกระบวนการ K ค่าเวลาหน่วงของกระบวนการ t_0 (Dead Time) และจะถูกประมาณโดยการลากเส้นสัมผัสกับจุดที่มีอัตราการเปลี่ยนแปลงสูงสุด (Maximum Slope) ดังรูปที่ 3.5 ส่วนค่า K ทราบได้เมื่อผลตอบสนองของกระบวนการเข้าสู่สภาวะคงที่

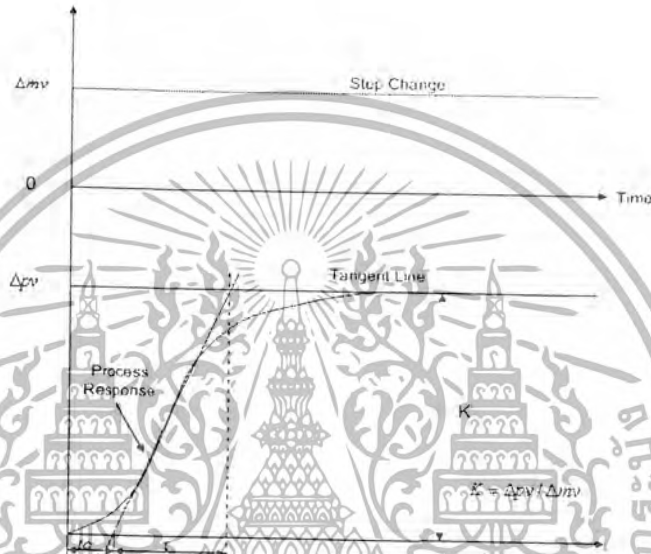
ดังนั้นจะเห็นได้ว่า ฟังก์ชันถ่ายโอนของกระบวนการจะประมาณได้ด้วยระบบอันดับหนึ่งแบบมีการหน่วงเวลา (First Order Lag Plus Dead Time : FOPDT) คือ

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการทดลองการทดลอง

ปรับตัวควบคุมอยู่ในตำแหน่ง Manual Mode แล้วทำการปรับค่า M_v ไว้ค่า ๆ หนึ่งให้คงที่ ตลอดในช่วงการทดลองหาค่า (เช่น 10 %) แล้วสังเกตค่า p_v จากเครื่องบันทึกค่าว่าได้ผลตามรูปที่ 3.5 หรือไม่ ถ้าได้ก็ทำการลากเส้นสัมผัส Tangent Line แล้วทำการหาค่าอัตราขยายของกระบวนการ K ค่าเวลาคงที่ของกระบวนการ τ (Time Constant) และค่า t_0 (Dead Time) จากนั้นก็เลือกวิธีการว่าต้องการควบคุมรูปแบบไหน เช่น PI หรือ PID โดยสามารถคำนวณได้ใน



รูปที่ 2.7 แสดงผลตอบสนองรูปตัว S เมื่อใช้วิธี Process Reaction Curve

ตารางที่ 2.1 สูตรสำหรับหาค่าพารามิเตอร์ของตัวควบคุม PID ตามวิธี Process Reaction Curve

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 ระบบเกี่ยวกับระดับของของเหลว

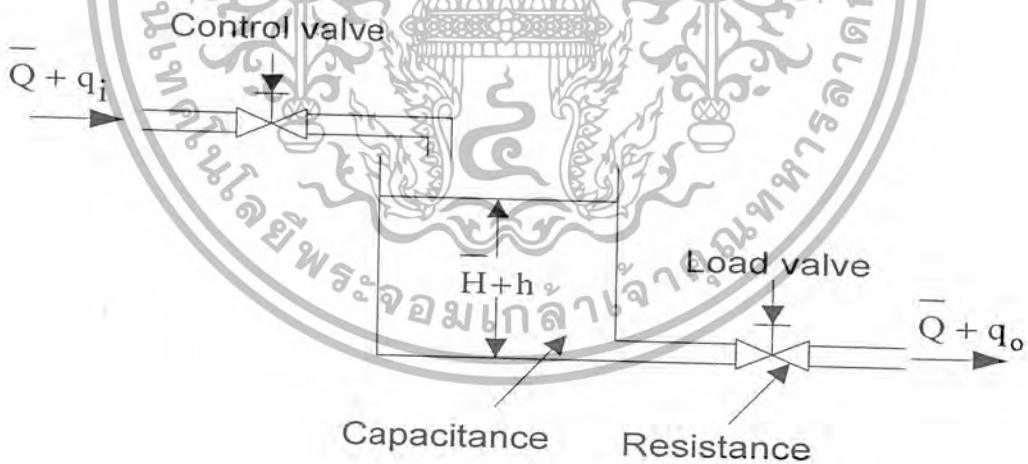
ในที่นี้จะพิจารณาถึงความต้านทาน (Resistance) และค่าความจุ (Capacitance) เพื่อที่จะอธิบายถึงคุณลักษณะทางพลศาสตร์ ของระบบเกี่ยวกับระดับของของเหลว

2.8.1 ค่าความต้านทาน

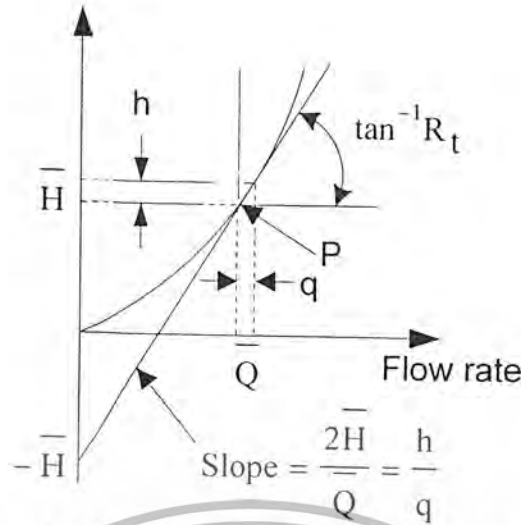
พิจารณาจากการไหลผ่านท่อสั้นๆ ที่อยู่ระหว่างภาชนะบรรจุ (Tank) สองใบ ค่าความต้านทานการไหลของของเหลวในท่อหรือสิ่งกีดขวาง R จะกำหนดได้ว่าเป็นค่าความเปลี่ยนแปลงของความแตกต่างของ ระดับของของเหลวของภาชนะบรรจุทั้งสอง และทำให้อัตราการไหล เกิดการเปลี่ยนแปลงไปหนึ่งระดับ นั่นคือ

$$R = \frac{\text{change in level difference, m}}{\text{change in flow rate, m}^3/\text{sec}}$$

เนื่องจากว่าความสัมพันธ์ระหว่าง อัตราการไหลและความแตกต่างของระดับของของเหลว ในกรณีของการไหลแบบราบเรียบหรือสม่ำเสมอ (Laminar) กับกรไหลแบบปั่นป่วน(Turbulent) จะไม่เหมือนกัน ในที่นี้จะแยกการพิจารณาออกเป็นสองกรณี



รูปที่ 2.8 แสดงระบบเกี่ยวกับของเหลว



รูปที่ 2.9 แสดงรูปกราฟเมื่อของเหลวไหลผ่าน

จากรูปของเหลวจะไหลผ่าน Load Valve ออกจากภาชนะบรรจุ ถ้าการไหลผ่านสิ่งกีดขวางนี้เป็นการไหลแบบราบเรียบ ความสัมพันธ์ระหว่างอัตราการไหลกับระดับความสูงของของเหลวในถังบรรจุที่ภาวะคงที่คือ

$$Q = KH$$

โดยที่ Q อัตราการไหลของของเหลวภาวะคงที่
 K ค่าสัมประสิทธิ์
 H ระดับความสูงที่ภาวะคงที่

จะเห็นว่ากฎเกี่ยวกับการไหลแบบราบเรียบจะคล้ายคลึงกับกฎของ Coulomb ซึ่งกระแสนี้เป็นสัดส่วนโดยตรงกับความต่างศักย์ค่าความต้านทาน R_t คือ

$$R_t = \frac{dH}{dQ} = \frac{H}{Q}$$

ซึ่งถ้ามียูนิทค่าคงที่ และเทียบได้กับค่าความต้านทานไฟฟ้า สำหรับการไหลแบบปั่นป่วน อัตราการไหลที่สภาวะคงที่ คือ

$$Q = K\sqrt{H}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าความต้านทาน R_t คือ

$$R_t = \frac{dH}{dQ}$$

จากสมการของอัตราการไหล เขียนใหม่ได้ว่า

$$dQ = \frac{K}{2\sqrt{H}} dH$$

$$\frac{dH}{dQ} = \frac{2\sqrt{H}}{K} = \frac{2\sqrt{H}\sqrt{H}}{Q} = \frac{2H}{Q}$$

ดังนั้น

$$R_t = \frac{2H}{Q}$$

ค่าความต้านทานของของการไหลแบบปั่นป่วนขึ้นอยู่กับอัตราการไหล และระดับความสูง อย่างไรก็ตาม ในกรณีที่ การเปลี่ยนแปลงอัตราการไหลและระดับความสูงมีค่าน้อย ค่าความต้านทานนี้อาจถือว่าเป็นค่าคงที่

ในทางปฏิบัติ จะไม่ทราบค่าของ K ดังนั้น ค่าความต้านทาน จะหาได้จากเส้น โค้งที่แสดงความสัมพันธ์ระหว่างระดับความสูงกับอัตราการไหล เส้น โค้งนี้จะได้มาจากการทดลอง โดยการหาความชัน (Slope) ของเส้น โค้งในช่วงการทำงานของระบบ ในกรณีของรูปข้างต้น จุด P จะเป็นจุดการทำงานที่สภาวะคงที่ ดังนั้นค่าความชันซึ่งก็คือค่าของความต้านทาน จะหาได้ดังนี้

$$P = \frac{h}{q} = \frac{2H}{Q} = R_t$$

นอกจากนี้แล้ว ค่าความต้านทานยังอาจจะหาจาก Process Reaction Curve ของระบบได้จากวิธีหนึ่งด้วย

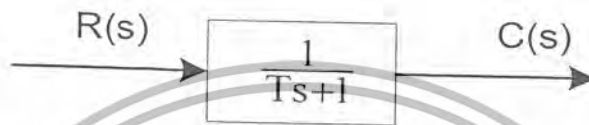
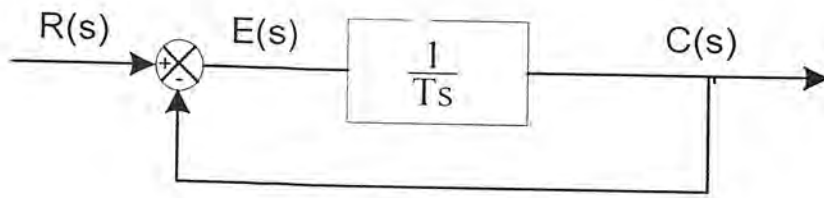
2.9 ผลตอบสนองของระบบต่อสัญญาณอินพุทชนิดต่างๆ

ในที่นี้จะพิจารณาผลตอบสนองของระบบอันดับหนึ่งดังนี้

2.9.1 ระบบอันดับหนึ่ง (Response of First-Order-System) พิจารณาระบบอันดับหนึ่ง

ในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



จากรูปเป็นฟังก์ชันถ่ายโอนของระบบควบคุมแบบลูปปิด คือ

$$\frac{C(s)}{R(s)} = \frac{1}{Ts+1}$$

สมการคุณลักษณะ คือ

$$\Delta(s) = Ts+1=0$$

หรือ

$$s = -1/T$$

ซึ่งแสดงว่า รากของสมการคุณลักษณะ หรือเรียกอีกชื่อหนึ่งว่า Pole ของฟังก์ชันถ่ายโอนแบบลูปปิด (Close Loop Pole) จะอยู่ที่ด้านซ้ายมือของแกน Real ในระนาบ s (s -plane) และแบบ (Close Loop Pole $s = -1/T$ Open Loop Pole $s = 0$)

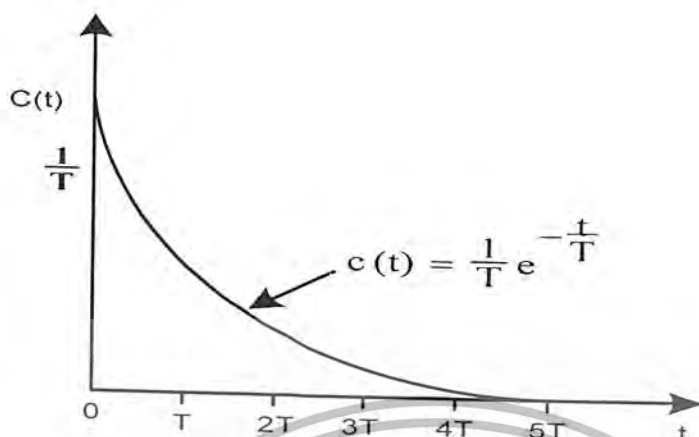
ต่อไป จะทำการศึกษาถึงผลตอบสนอง หรือเอาต์พุตของระบบ เมื่ออินพุต (หรือสัญญาณรบกวน) เมื่อมีการเปลี่ยนแปลงของอินพุตอ้างอิงประเภทต่าง ๆ ที่กล่าวมาข้างต้น ผลตอบสนองต่ออินพุตที่เป็น Unit Impulse (Unit Impulse Response) กรณีนี้ $R(s)$ มีค่าเท่ากับหนึ่ง ดังนั้น

$$C(s) = \frac{1}{Ts+1}$$

หรือ

$$c(t) = \frac{1}{T} e^{-t/T}$$

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 แสดงผลตอบสนองของอินพุตที่เป็น Unit Impulse

2.9.1.1 ผลตอบสนองอินพุตที่เป็น Unit Step (Unit Step Response)

กรณีนี้ $R(s)$ มีค่าเท่ากับ $1/s$ ดังนั้น

$$C(s) = \frac{1}{Ts+1} \cdot \frac{1}{s}$$

หรือโดยทำการ Partial Fraction Expansion

$$C(s) = \frac{1}{s} - \frac{T}{Ts+1}$$

นั่นคือ

$$c(t) = 1 - e^{-t/T} = 0.632 \quad (t \geq 0)$$

$$c(t) = 1 - e^{-1}$$

นอกจากนี้ จะเห็นว่าเมื่อระบบอยู่ในสถานะแรกเริ่มค่าของ $c(t)$ จะเท่ากับศูนย์ และเมื่อเข้าสู่สถานะคงที่ ค่าของ $c(t)$ จะเท่ากับหนึ่ง หรือหมายถึงว่าระบบอันดับหนึ่งเฉพาะกรณีตัวอย่างในรูปที่ 2.8 ข้างต้นนั้น มีค่าความคลาดเคลื่อนที่ภาวะคงที่เท่ากับศูนย์ หรือระบบมีความเที่ยงตรงสูง ค่าของเอาต์พุตสามารถเปลี่ยนแปลงไปตามค่าของอินพุตที่ต้องการได้

ค่าความคลาดเคลื่อน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$e(t) = r(t) - c(t) \\ = e^{-t/T}$$

โดยที่ $e(t)$ ค่าความคลาดเคลื่อน

ค่าความคลาดเคลื่อนที่ภาวะคงที่ $e_{ss}(t)$ จะหาได้จาก

$$e_{ss}(t) = \lim_{t \rightarrow \infty} e(t) \\ = 0$$

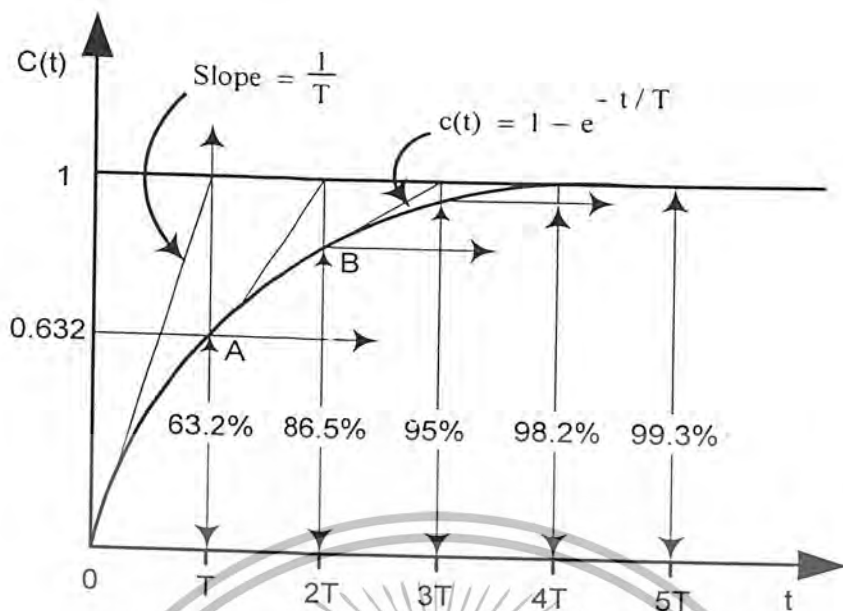
$$\left. \frac{dc}{dt} = \frac{1}{T} e^{-t/T} \right|_{t=0} = \frac{1}{T}$$

ซึ่งหมายความว่า ถ้าผลตอบสนองต่อเวลาสามารถรักษาความเร็วของการเปลี่ยนแปลงให้เท่ากับความเร็วที่สถานะเริ่มแรกคงสมการข้างต้นได้ ผลตอบสนองต่อเวลา $c(t)$ จะมีค่าเท่ากับค่าที่ต้องการได้ เมื่อเวลาผ่าน $t = T$ ไปแต่อย่างไรก็ตาม จะเห็นว่าความชันของผลตอบสนองจะลดลงเรื่อยๆ จากค่า $1/T$ ที่เวลาไปเป็น ที่เวลา $t = 0$ ไปเป็น 0 ที่เวลา $t = \infty$

ผลตอบสนองต่ออินพุตที่เป็น Unit Step แสดงได้ดังรูปข้างล่างนี้

ที่ $t = T$	$c(t) = 63.2\%$ ของค่าสุดท้ายที่ภาวะคงที่
$t = 2T$	$c(t) = 86.5\%$ ของค่าสุดท้ายที่ภาวะคงที่
$t = 3T$	$c(t) = 95\%$ ของค่าสุดท้ายที่ภาวะคงที่
$t = 4T$	$c(t) = 98.2\%$ ของค่าสุดท้ายที่ภาวะคงที่
$t = 5T$	$c(t) = 99.3\%$ ของค่าสุดท้ายที่ภาวะคงที่

แสดงว่า เมื่อ $t \geq 4T$ ผลตอบสนองต่อเวลาจะมีค่าอยู่ในช่วง 2% ของค่าสุดท้าย และโดยทั่วไปแล้วจะถือว่า เมื่อผลตอบสนองต่อเวลามีค่าเข้าสู่ช่วง 2% ของค่าสุดท้าย จะเป็นการเข้าสู่สภาวะคงที่ จะเป็นการเข้าสู่สภาวะคงที่ หรือเป็นการเปลี่ยนสภาวะจากผลตอบสนองชั่วคราวไปเป็นผลตอบสนองที่สภาวะคงที่



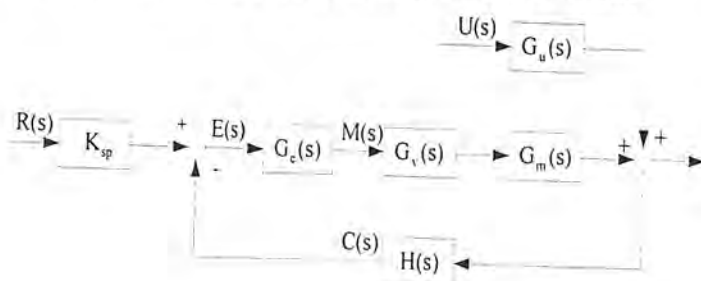
รูปที่ 2.11 แสดงผลตอบสนองของอินพุตที่เป็น Unit Step

สรุปพฤติกรรมของระบบอันดับหนึ่ง

1. รากของสมการคุณลักษณะ (หรือ Pole ของฟังก์ชันถ่ายโอนแบบลูปปิด) จะอยู่บนแกน Real ทางครึ่งซ้าย ในระนาบ s (ถ้าอย่างทางครึ่งขวา ระบบจะไม่มีเสถียรภาพ)
2. พฤติกรรมทางพลศาสตร์ของผลตอบสนองต่อเวลา (เอาท์พุท) จะอยู่ในรูปของ Exponential Curve
3. ความเร็วของผลตอบสนองต่อเวลาในการติดตามการเปลี่ยนแปลงของอินพุต จะขึ้นอยู่กับค่าเวลาคงที่ (Time Constant) ของระบบ ถ้า T มีค่าน้อยเอาท์พุทจะเปลี่ยนแปลงตามอินพุตได้รวดเร็ว
4. ผลตอบสนองต่อเวลาจะเข้าสู่สภาวะคงที่ เมื่อเวลาผ่านไป 4 เท่าของ Time Constant

2.10 การวิเคราะห์หาคูณลักษณะของกระบวนการ

พิจารณาสถิตโคอะแกรมของระบบควบคุมแบบป้อนกลับดัง



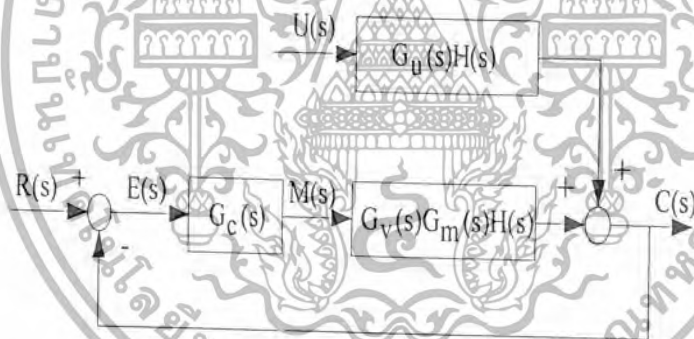
รูปที่ 2.12 แสดงบล็อกโคอะแกรมของระบบควบคุมแบบป้อนกลับ

- โดยที่
- $R(s)$ = การแปลงลาปลาซ (Laplace Transform) ของค่าอินพุตอ้างอิงหรือค่าเป้าหมาย
 - $E(s)$ = การแปลงลาปลาซของค่าความคลาดเคลื่อน (Error)
 - $M(s)$ = การแปลงลาปลาซของสัญญาณควบคุมจากตัวควบคุม
 - $C(s)$ = การแปลงลาปลาซของสัญญาณขาออกของอุปกรณ์แปลงสัญญาณ
 - $U(s)$ = การแปลงลาปลาซของสัญญาณรบกวน (Disturbance)
 - $G_c(s)$ = ฟังก์ชันถ่ายโอนของตัวควบคุม
 - $G_v(s)$ = ฟังก์ชันถ่ายโอนของวาล์วควบคุม
 - $G_m(s)$ = ฟังก์ชันถ่ายโอนของกระบวนการระหว่างตัวแปรควบคุม (Process Variable)

และตัวแปรปรับกระบวนการ (Manipulated Variable)

$G_u(s)$ = ฟังก์ชันถ่ายโอนของกระบวนการระหว่างตัวแปรควบคุมและสัญญาณรบกวน

$H(s)$ = ฟังก์ชันถ่ายโอนของอุปกรณ์ตรวจจับสัญญาณ และอุปกรณ์แปลงสัญญาณ
 โครงสร้างของระบบควบคุมแบบป้อนกลับในรูปที่ 2.12 สามารถสรุปให้เข้าใจง่าย



รูปที่ 2.13 แสดงบล็อกไดอะแกรมของระบบควบคุมแบบป้อนกลับเมื่อทำการลดรูปแล้ว

ดังนี้

เมื่อทำการลดรูปไดอะแกรมของรูปที่ 2.11 จะได้รูปใหม่ดังรูปที่ 2.12 โดยมีความสัมพันธ์

$$G(s) = G_v(s)G_m(s)H(s)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบที่ง่ายที่สุดของทรานเฟอร์ฟังก์ชันที่ใช้ในการหาค่าพารามิเตอร์ของกระบวนการ มักจะอยู่ในรูปของกระบวนการอันดับหนึ่งที่มีการหน่วงเวลา (First Order Plus Dead Time : FOPDT) ซึ่งเป็นรูปแบบหนึ่งที่นิยมใช้กันมากที่สุด ดังสมการ

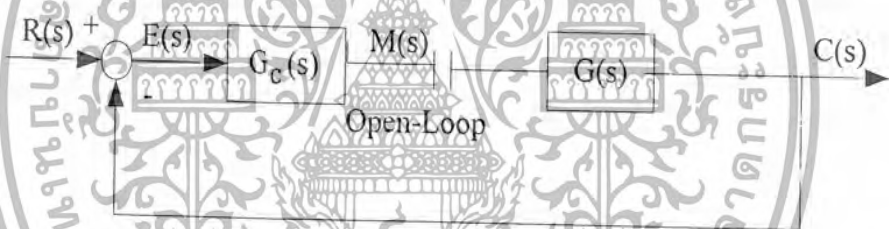
$$G(s) = \frac{Ke^{-t_0 s}}{\tau s + 1}$$

เมื่อ K = อัตราขยายของกระบวนการ (Process Gain)

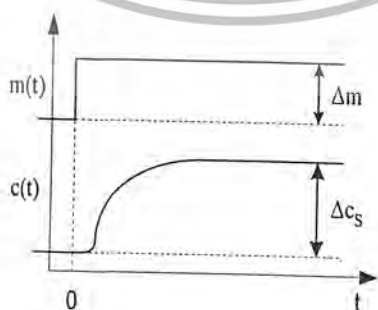
T_0 = ค่าหน่วงเวลาของกระบวนการ (Process Dead Time)

τ = ค่าคงที่เวลาของกระบวนการ (Process Time Constant)

(Open Loop) เนื่องจากการทดสอบเฉพาะกระบวนการเท่านั้นและไม่พิจารณาสัญญาณรบกวนที่เกิดขึ้น โดยทำการป้อนสัญญาณขั้นบันได (Step Signal) ดังรูป 2.13



รูปที่ 2.14 แสดงบล็อกไดอะแกรมของการทดสอบวงเปิด โดยการป้อนสัญญาณแบบขั้นบันได



รูปที่ 2.15 แสดงผลตอบสนองของกระบวนการแบบวงจรเปิดต่อสัญญาณขั้นบันได

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.13 จะได้ผลตอบสนองของกระบวนการคือ

$$C(s) = G(s) H(s)$$

สำหรับระบบ FOPDT ที่มีสัญญาณอินพุทเป็นสัญญาณขั้นบันไดขนาด Δm จะได้ความสัมพันธ์ดังนี้

$$C(s) = \frac{Ke^{-t_0 s}}{\tau s + 1} \cdot \frac{\Delta m}{s}$$

$$C(s) = K\Delta m e^{-t_0 s} \left[\frac{1}{s} - \frac{\tau}{\tau s + 1} \right]$$

อินเวิร์คสถาปลาสมการที่

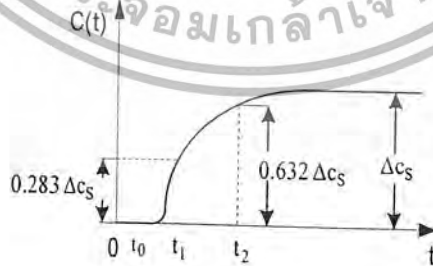
$$\Delta c(t) = K\Delta m u(t - t_0) \left[1 - e^{-(t-t_0)/\tau} \right]$$

พิจารณาที่เวลา $t < t_0$ จะได้ว่า $\Delta c(t) = 0$

จากกราฟของสมการ ที่แสดงในรูปที่ 2.14 จะพบว่า

$$\Delta c(s) = \lim_{t \rightarrow \infty} \Delta c(t) = K\Delta m$$

โดยที่ $\Delta c(s)$ = Steady-State change in $c(t)$



รูปที่ 2.16 แสดงกราฟแสดงการหาค่า Δc_s

$$K = \frac{\Delta c_s}{\Delta m}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนวิธีในการหาค่าหน่วยเวลา (Dead Time : t_0) และค่าคงที่ (Time Constant : τ) เวลา นั้น ได้มีผู้เสนอความคิดต่างๆดังนี้

Dr.Cecil L.Smith (1972) ได้เสนอการหาค่าของ t_0 และ τ โดยการพิจารณาผลตอบสนอง ของกระบวนการ ณ.เวลาที่ทำให้ผลตอบสนองในรูปแบบจำลองและผลตอบสนองของ กระบวนการจริงมีค่าเท่ากัน ดังแสดงในรูปนั้นคือ จะพิจารณาที่เวลา

$$t_1 = t_0 + \frac{\tau}{3}$$

$$t_2 = t_0 + \tau$$

นำค่าเวลานี้แทนลงในสมการที่จะได้ว่า

$$\Delta c(t_0 + \tau) = K \Delta m(1 - e^{-1}) = 0.632 \Delta c_s$$

$$\Delta c(t_0 + \frac{\tau}{3}) = K \Delta m(1 - e^{-1/3}) = 0.283 \Delta c_s$$

หรืออาจกล่าวได้ว่า

t_1 คือ เวลาที่ทำให้ $\Delta c = 0.283 \Delta c_s$

t_2 คือ เวลาที่ทำให้ $\Delta c = 0.632 \Delta c_s$

จะได้ว่า

$$\tau = \frac{3}{2}(t_2 - t_1)$$

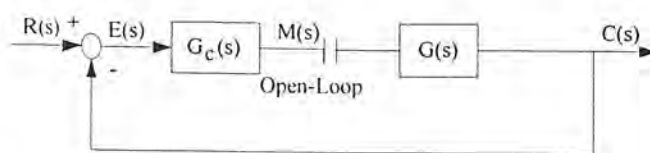
$$t_0 = t_2 - \tau$$

ค่าคุณลักษณะของกระบวนการทั้ง 3 ค่าคือ K , τ และ t_0 นี้จะนำไปใช้ในการสังเคราะห์ ค่าพารามิเตอร์ของตัวควบคุม PID คือค่า K_c , T_i และ T_d ที่จะใช้ให้กับตัวควบคุมดังที่จะกล่าวต่อไป

2.11 การสังเคราะห์หาค่าพารามิเตอร์ของตัวควบคุม PID ในการควบคุม

ในส่วนนี้จะแสดงถึงวิธีการหาตัวควบคุมชนิดต่างๆซึ่งเกิดจากทรานเฟอร์ฟังก์ชันของวง ควบคุมป้อนกลับ (Feedback Loop) โดยไม่พิจารณาสัญญาณรบกวน ดังในรูปที่ 2.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 แสดงบล็อกไดอะแกรมของการสังเคราะห์ค่าพารามิเตอร์ของตัวควบคุมแบบป้อนกลับ

หาทรานเฟอร์ฟังก์ชันของรูปที่ 2.16 จะได้ว่า

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

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

จากสมการข้างต้น ถ้าสัญญาณอินพุตและสัญญาณเอาต์พุตมีค่าเท่ากัน (Perfect Control) $C(s) = R(s)$ แล้วจะได้ว่า $[C(s)/R(s)] = 1$ ซึ่งจะได้รูปแบบของตัวควบคุมเป็น

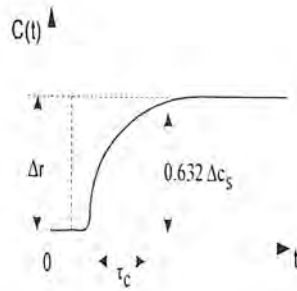
$$G_c(s) = \left(\frac{1}{G(s)} \right) \left(\frac{1}{1/(1-1)} \right) = \left(\frac{1}{G(s)} \right) \left(\frac{1}{0} \right)$$

จะพบว่า $G_c(s)$ มีอัตราขยายเป็นอนันต์ (Infinite) หรือกล่าวอีกนัยหนึ่ง สำหรับระบบที่เป็น Perfect Control นั้น ไม่สามารถให้กับระบบควบคุมแบบป้อนกลับได้

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

โดยที่ τ_c = ค่าคงที่เวลาของระบบ (Time Constant)

τ_c เป็นพารามิเตอร์ที่สามารถปรับค่าได้ ซึ่งยังมีค่าน้อยจะทำให้ระบบเข้าสู่สภาวะคงที่ได้เร็วขึ้น ดังแสดงดังรูปที่ 2.18



รูปที่ 2.18 แสดงผลตอบสนองของระบบแบบป้อนกลับในรูปแบบอันดับหนึ่ง

แทนสมการในสมการ จะได้ว่า

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

$$G_c(s) = \left(\frac{1}{G(s)} \right) \cdot \left(\frac{1}{\tau_c s} \right)$$

แทนค่าสมการในสมการเพื่อสังเคราะห์ค่าพารามิเตอร์ของตัวควบคุมสำหรับกระบวนการแบบ FOPDT จะได้ว่า

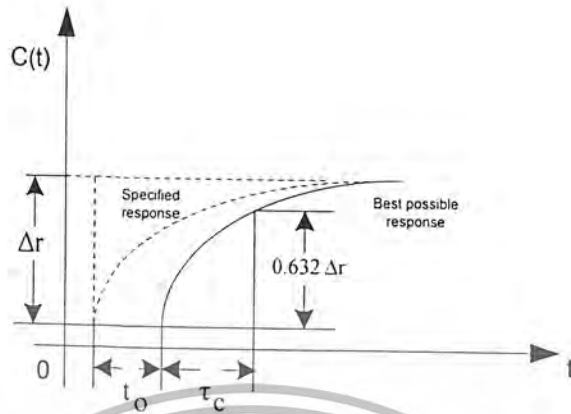
$$G_c(s) = \left(\frac{\tau s + 1}{K e^{-t_0 s}} \right) \cdot \left(\frac{1}{\tau_c s} \right)$$

$$= \frac{\tau}{K \tau_c} \left(1 + \frac{1}{\tau s} \right) e^{t_0 s}$$

พิจารณาสมการในโดเมนเวลาจะพบว่ค่าการหน่วงเวลา t_0 เป็นลบส่งผลให้ไม่สามารถทำให้เป็นตัวควบคุมที่เป็นจริงได้ เนื่องจากต้องการการทำนายค่าเวลา t_0 ล่วงหน้า ดังนั้นจึงต้องพิจารณาร่วมกับผลตอบสนองแบบปิด ที่มีการหน่วงเวลาเป็นเวลา t_0 คือ

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 แสดงผลตอบสนองแบบปัดที่มีการหน่วงเวลาเป็นเวลา t_0

หรือ

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

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

เทอมเอ็กโพเนนเชียล ($e^{-t_0 s}$) สามารถประมาณค่าได้โดยอนุกรมเทย์เลอร์ (Taylor Series Expansion) คือ

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

ใช้เฉพาะ 2 เทอมแรกคือ $1 - t_0 s$ สำหรับกระบวนการอันดับหนึ่ง แทนในสมการจะได้

$$G_c(s) = \left(\frac{\tau s + 1}{K} \right) \cdot \frac{1}{(\tau_c + t_0) s}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$= \left(\frac{\tau}{K(\tau_c + t_0)} \right) \cdot \left(1 + \frac{1}{\tau_s} \right)$$

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



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

หลักการและทฤษฎีที่ใช้ในการออกแบบชุดทดลอง

3.1 องค์ประกอบของการควบคุม

ระบบควบคุมทางอุตสาหกรรมโดยทั่วไปจะประกอบไปด้วยอุปกรณ์ 4 ส่วน คือ

3.1.1 กระบวนการ (Process) หมายถึงกระบวนการทางฟิสิกส์ที่ต้องการควบคุมให้มีสถานะตามต้องการสถานะของกระบวนการแสดงด้วยตัวแปรกระบวนการ (Process Variable) การควบคุมสถานะของกระบวนการทำได้โดยการปรับ หรือ เปลี่ยนค่าตัวแปรปรับกระบวนการ (Manipulated Variable) ในทิศทางที่ทำให้ตัวแปรกระบวนการมีค่าเปลี่ยนแปลงไปตามต้องการ

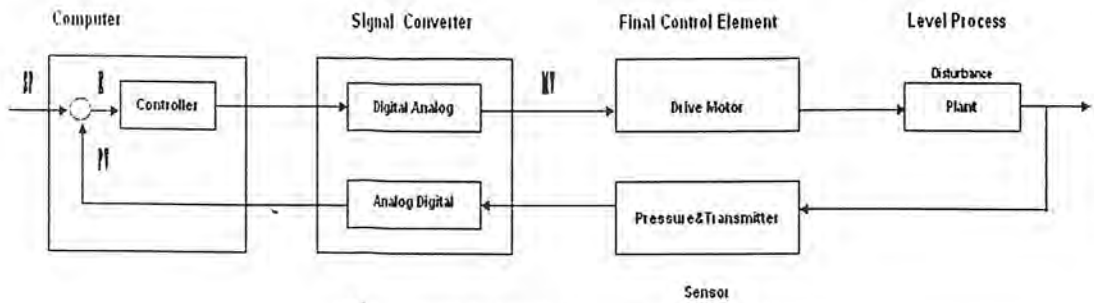
3.1.2 (Measuring Device) เป็นอุปกรณ์ที่ให้สัญญาณขาออกหรือสัญญาณ (Measuring Signal) ซึ่งมีขนาดสัมพันธ์กับตัวแปรทางฟิสิกส์ที่ต้องการวัด หรือ ตัวแปรกระบวนการโดยทั่วไปสัญญาณขาออกของอุปกรณ์วัดจะเป็นสัญญาณมาตรฐาน 4-20mA, 1.5 V หรือสัญญาณลมขนาด 3-15 psi

3.1.3 เครื่องควบคุมหรือตัวควบคุม (Controller) เป็นอุปกรณ์ที่ทำหน้าที่คำนวณหาสัญญาณขาออกหรือสัญญาณควบคุม (Control Signal) จากสัญญาณเข้าที่ได้แก่สัญญาณวัดและจุดปรับตั้ง (Set Point) ตามเกณฑ์การควบคุม (Control Law) ที่ถูกกำหนดไว้ก่อนล่วงหน้า

3.1.4 อุปกรณ์ปรับกระบวนการ (Final Control Element) คือ อุปกรณ์ที่ทำหน้าที่ปรับแต่งสถานะของกระบวนการ ด้วยการเปลี่ยนแปลงค่าตัวแปร ปรับกระบวนการตามคำสั่ง หรือ สัญญาณควบคุมที่ได้รับจากตัวควบคุม

3.2 อุปกรณ์และฮาร์ดแวร์

ก่อนอื่นจะขอกล่าวถึงการออกแบบอุปกรณ์ฮาร์ดแวร์ที่ใช้ในในโรงงาน ทฤษฎีที่ใช้ประกอบการใช้งานและการควบคุมการทำงานกับอุปกรณ์และการออกแบบกระบวนการ (Process) ที่ใช้ในการควบคุมซึ่งจะสามารถแสดงได้ดัง Block Diagram ของระบบทั้งหมดได้ดัง



รูปที่ 3.1 แสดง Block Diagram ของระบบ

จาก Block Diagram ของระบบจะเห็นได้ว่าในระบบจะประกอบด้วยส่วนต่าง ๆ 5 ส่วน คือ

1. Computer ทำหน้าที่เป็น Controller
2. Signal Converter ทำหน้าที่ในการแปลงสัญญาณ Digital\Analog และ Analog\Digital
3. Final Control Element ทำหน้าที่ในการ Drive Pump Motor
4. Sensor and Transmitter ทำหน้าที่ในการวัดระดับน้ำ และส่งสัญญาณมาตรฐานที่วัดได้ 4-20 mA ไปใช้งานต่อไป
5. Level Process Plant ทำหน้าที่เป็นแทงค์กักเก็บน้ำเพื่อจะนำเอากระบวนการควบคุมแบบอัตโนมัติมาทำการทดลอง โดยเป็นแบบ 1 แท็งค์ แบบหนึ่งเวลา

3.2.1 Computer (Controller)

Computer ในส่วนนี้จะทำหน้าที่ในการควบคุม Controller การทำงานของกระบวนการควบคุมระดับน้ำ และยังเป็นมอนิเตอร์ ไว้แสดงค่าการทำงานของกระบวนการวัดระดับน้ำ พร้อมทั้งเป็นตัวควบคุมทั้ง Hardware และ Software ด้วย

3.2.2 Signal Converter

ในส่วนนี้จะทำหน้าที่ในการรับสัญญาณที่วัดได้จากกระบวนการแล้วทำการหาร และแปลงสัญญาณที่รับเข้ามา Analog\Digital เพื่อส่งต่อไปยัง Final Control Element แต่เนื่องจากว่า Input ที่เข้า Analog\Digital และ Output ที่ออกจาก Digital\Analog นั้นสัญญาณจะอยู่ในช่วง 0-5 V แต่สัญญาณที่ใช้ในการติดต่อสื่อสารภายในกระบวนการนั้น จะใช้สัญญาณมาตรฐานขนาด 1-5 V หรือ 4-20 mA ดังนั้นจึงต้องมีวงจรแปลงสัญญาณให้เหมาะสมก่อนที่จะนำสัญญาณส่งไปยังส่วน Final Control Element

การแปลงสัญญาณ Analog/Digital และ Digital/Analog

การแปลงสัญญาณ A/D และ D/A ที่ใช้ในการทดลองนี้จะใช้การ์ดรุ่น AD12-16(PCI)E ของบริษัท CONTEC ซึ่งจะเป็นการ์ดที่มีลักษณะการเชื่อมต่อกับคอมพิวเตอร์แบบ PCI การ์ดนี้ จะทำหน้าที่ในการเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นตัวควบคุมให้กับอุปกรณ์ขับเคลื่อนของปั๊มน้ำและอุปกรณ์กรองความถี่ต่ำ โดยรูปของการแปลงสัญญาณ A/D และ D/A แสดงได้ดังรูปที่ 3.2 และการ์ดจะประกอบไปด้วยส่วนต่างๆดังนี้

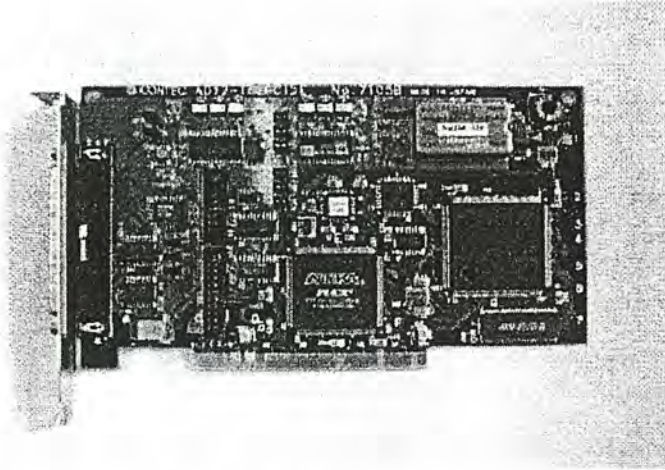
- ส่วนแปลงสัญญาณอนาล็อกเป็นดิจิทัล (A/D)
- ส่วนแปลงสัญญาณดิจิทัลเป็นอนาล็อก (D/A)

3.2.2.1 ส่วนแปลงสัญญาณ A/D

ส่วนแปลงสัญญาณ A/D นี้จะทำหน้าที่ในการรับสัญญาณแรงดันจากอุปกรณ์กรองความถี่ และแปลงสัญญาณให้อยู่ในรูปของสัญญาณดิจิทัล เพื่อส่งให้กับคอมพิวเตอร์หรือตัวควบคุมต่อไป โดยส่วนแปลงสัญญาณ A/D นี้จะมีจำนวนช่องในการรับสัญญาณถึง 16 ช่อง แบบ Single Ended, มีความละเอียดสูง (Resolution) ถึง 12 บิต, มีความเที่ยงตรงสูงถึง $\pm 0.03\%$ หรือ ± 3 บิต, มีความเป็นเชิงเส้นมากถึง ± 3 บิต, และจะมีความเร็วในการแปลงสัญญาณสูงสุดถึง $1 \mu\text{sec/ch}$. หรือ 1 MHz นอกจากนั้นยังย่านของสัญญาณทางด้านอินพุต (Input Rang) Un-isolated $\pm 5\text{V}$, $\pm 2.5\text{V}$, $0\sim 10\text{V}$ or $0\sim 5\text{V}$, อินพุตอิมพีแดนซ์ (Input Impredance) มากกว่า 1 เมกะโอห์ม บัฟเฟอร์ (Buffer Memory) 256 K word สามารถเลือกช่วงของสัญญาณอินพุตที่จะรับเข้ามาได้หลายแบบ ขึ้นอยู่กับการเลือกที่สวิทช์หรือเลือกทางซอฟต์แวร์

3.2.2.2 ส่วนแปลงสัญญาณ D/A

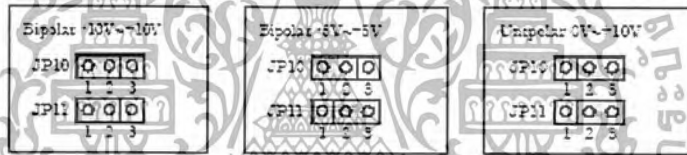
ส่วนแปลงสัญญาณ D/A นี้จะทำหน้าที่ในการรับสัญญาณควบคุมจากคอมพิวเตอร์หรือตัวควบคุม และแปลงให้อยู่ในรูปของสัญญาณอนาล็อกเพื่อส่งให้กับชุดขับเคลื่อนของปั๊มน้ำต่อไป โดยส่วนแปลงสัญญาณ D/A นี้จะมีจำนวนช่องในการส่งสัญญาณ 1 ช่อง แบบ Single Ended, มีความละเอียดสูง (Resolution) ถึง 12 บิต, มีความเที่ยงตรงสูงถึง $\pm 0.005\%$ หรือ ± 0.5 บิต, มีความเป็นเชิงเส้นมากถึง ± 0.5 บิต, และจะมีความเร็วในการแปลงสัญญาณสูงสุดถึง $6\mu\text{sec/ch}$. หรือ 166.7 KHz นอกจากนั้นยังย่านของสัญญาณทางด้านเอาต์พุต (Output Rang) Un-isolated $\pm 10\text{V}$, $\pm 5\text{V}$, $0\sim 10\text{V}$, กระแสทางด้านเอาต์พุตสูงสุด (Output Current) 5 mA Max สามารถเลือกช่วงของสัญญาณอินพุตที่จะรับเข้ามาได้หลายแบบ ขึ้นอยู่กับการเลือกที่สวิทช์หรือเลือกทางซอฟต์แวร์



รูปที่ 3.2 แสดงการ์ดแปลงสัญญาณ Analog/Digital และ Digital/Analog

3.2.2.3 วิธีการ Set ค่าเอาต์พุต Rang ของ D/A Card ในที่นี้จะเลือกการ Set Jumper ที่ให้ย่านเอาต์พุตออกมาที่ 0V - 10V

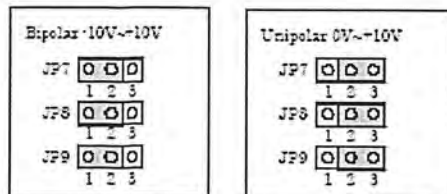
AD12-16(PCIE):



รูปที่ 3.3 แสดงการ Set Jumper เพื่อเลือกย่านของสัญญาณเอาต์พุต

3.2.2.4 วิธีการ Set ค่าอินพุต Rang ของ A/D Card ในที่นี้จะเลือกการ Set Jumper ที่ต้องการค่าอินพุตที่มีย่านรับเข้ามาที่ 0V - 10V

AD12-16(PCIE):

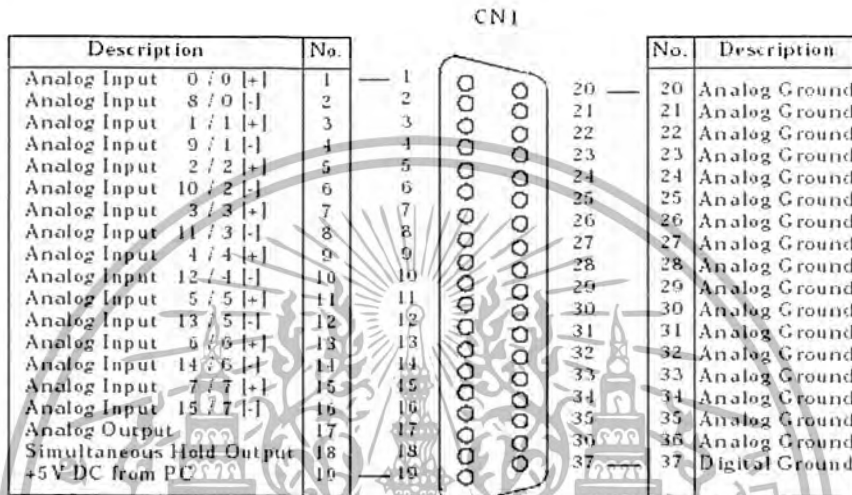


รูปที่ 3.4 แสดงการ Set Jumper เพื่อเลือกย่านของสัญญาณอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2.5 จากรูปเป็นการแสดง Analog Input /Output Connector ของ Card

Analog Input มีช่องสัญญาณทั้งหมด 16 Ch. คือ 1 – 16 เป็นแบบ Single Ended Channel หรือ 8 Differential Channels ในส่วน Analog Output จะอยู่ที่ช่องที่ 17 เป็นแบบ Single Ended Channels นอกจากนั้นส่วนใหญ่จะเป็นช่องสัญญาณ Analog Ground หรือ Digital Ground และ ช่องสัญญาณ Simultaneous Hold Output และ ช่องสัญญาณสำหรับเป็นแหล่งจ่ายจากคอมพิวเตอร์ไปการ์ด ขนาด +5Vdc



รูปที่ 3.5 แสดงการช่องสัญญาณ Analog Input/Output ของการ์ด

3.2.2.6 AD12-16(PC)E Table เป็นตารางแสดงรายละเอียดของ A/D Card รุ่น

AD12-16(PCI)E

Item	Specification
Analog Input	
Number of Channels	16 single-ended channels or 8 differential
Input Range	Un-isolated =10V or 0~10V
Absolute Max. Input Voltage	=20V
Input Impedance	More than 1M Ω
Resolution	12 bits
Conversion Speed	10 μ sec/ch. Max.
Conversion Accuracy Non-linearity error (Note 1)	=2LSB (Gain = $\times 1$ and $\times 2$) =4LSB (Gain = $\times 4$ and $\times 8$)
Buffer Memory	256K Words FIFO or 256K Words Ring
Conversion Start Trigger	Software command, analog input signal
Conversion Stop Trigger	Software command, storage data number
Analog Output	
Number of Channel	1 single-ended channel
Output Range	Un-isolated =10V, =5V or 0~10V
Output Current Ability	=5mA Max.
Resolution	12 bits
Non-Linearity Error (Note 1)	=1/2LSB
Conversion Speed	6 μ sec Max.
Power consumption (Note 2)	+5VDC, 1100mA Max.
Dimensions (inch)	6.9 \times 4.2 (176.4mm \times 107mm)

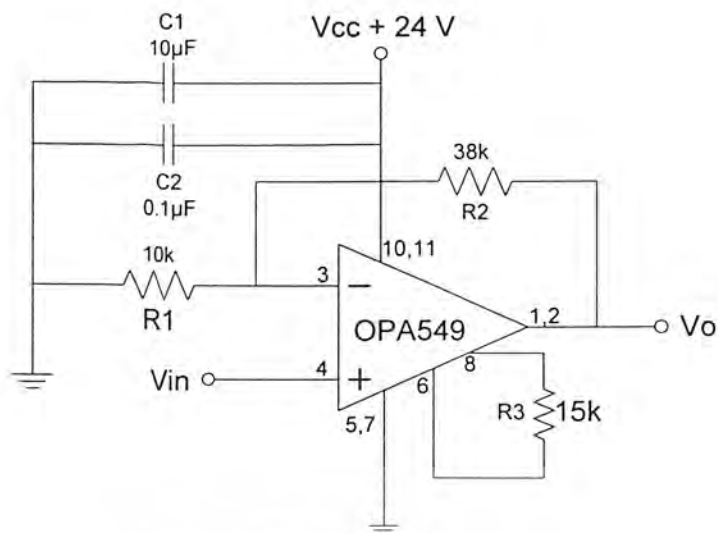
รูปที่ 3.6 เป็นตารางแสดงรายละเอียดของ A/D Card รุ่น AD12-16(PCI-E)

3.2.3 Final Control Element

ในส่วนนี้จะทำหน้าที่ในการรับสัญญาณขนาด 4-20 mA ที่ส่งมาจากส่วน Signal Converter ที่ออกจากส่วนเอาต์พุตของ Digital/Analog แล้วนำสัญญาณที่ได้นี้ไปแปลงให้อยู่ในช่วง 0-5 V แล้วส่งสัญญาณขนาด 0-10 V นี้ไปยัง Motor Driver นี้จะทำหน้าที่เป็นตัวส่งสัญญาณให้ Pump Motor ทำงาน

3.2.3.1 ชุดขับเคลื่อนมอเตอร์

วงจรขับเคลื่อนเป็นวงจรที่ทำหน้าที่ในการควบคุมการทำงานของ (Motor) ปริมาณน้ำ เพื่อให้สามารถ จ่ายน้ำได้มากหรือน้อยตามสัญญาณควบคุมที่ได้รับมาจากตัวควบคุม ซึ่งรูปของวงจรขับเคลื่อนแสดงดังรูปที่ 3.7



รูปที่ 3.7 วงจรออปแอมป์แบบไม่กลับเฟส (Non-inverting) ที่ใช้ขับมอเตอร์

จากรูปจะเห็นได้ว่าวงจรขับกระแสจะใช้ Op-Amp เพียงอย่างเดียวเท่านั้น โดย Op-Amp ที่ใช้นี้จะใช้ ไอซีเบอร์ OPA548 ของบริษัท Burr-Brown ซึ่งเป็น Power Op-Amp ที่สามารถจ่ายกระแสได้สูงถึง 3 A. และสามารถทำงานภายในช่วงแรงดัน $+60\text{V}$ หรือ $\pm 30\text{V}$ ลักษณะของวงจรขับกระแสจะเป็นวงจรขับกระแสที่เป็นแบบเชิงเส้น โดยจะรับสัญญาณแรงดันอินพุตขนาด $0 - 10\text{ V}$ มาจากส่วนของ D/A และขยายสัญญาณแรงดันเอาท์พุทให้มีขนาด $0 - 24\text{ V}$ เพื่อจ่ายให้กับปั๊มน้ำต่อไป การจ่ายแรงดันให้กับตัว Pump Motor ตามค่าสัญญาณอินพุทที่ได้รับเข้ามา $0 - 10\text{ Vdc}$ โดยเอาท์พุทของตัวกระแสมอเตอร์ คือ $0 - 24\text{ Vdc}$ รูปของอุปกรณ์ที่ใช้ในการขับกระแสให้กับปั๊มน้ำแสดงได้ดังรูปที่ 3.8

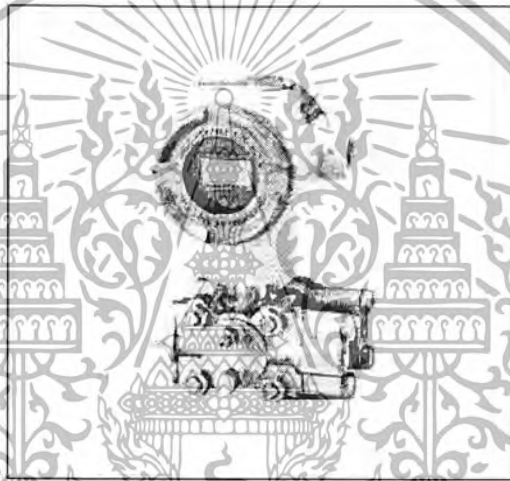


รูปที่ 3.8 แสดงชุดขับกระแสมอเตอร์ (Motor Driver)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 Sensor and Transmitter

Sensor and Transmitter ในส่วนนี้จะทำหน้าที่ในการวัดระดับน้ำของการบวมนการโดยใช้หลักการของ Differential Pressure Transmitter โดยจะใช้ความดันบรรยากาศเป็นความดันอ้างอิง เนื่องจากแท่งที่ทำการวัดระดับนั้นเป็นแบบเปิด เมื่อระดับน้ำในแท่งเปลี่ยนแปลงไปก็จะส่งผลให้ความดันที่ได้เปลี่ยนแปลงไปด้วย ตัวตรวจจับ Sensor ที่นำมาใช้ในกระบวนการวัดความดันนี้คือ Differential Pressure Transmitter Model Yamatake Honeywell Corporation รุ่น STD-920 มีช่วงการวัดด้าน Span (Measuring Span) คือ 0.75 – 100 kPa, ย่านการวัด (Measuring Range) คือ -100 – 100 kPa, Working Prssure Range คือ 2.0 kPa abs to 21MPa สัญญาณที่วัดได้จะอยู่ในรูปของกระแส 4 – 20 mA จากนั้นจะทำการและแปลงสัญญาณที่ได้นี้ให้เป็นแรงดันมีขนาด 1 – 5 Vdc ตามแรงดันที่วัดได้ โดยกระแสที่ออกมาจะคร่อมตัวต้านทาน ขนาด 250 Ohm ก่อนที่จะส่งไปยังส่วน Signal Converter ต่อไป



รูปที่ 3.9 แสดงชุด DP Cell (Differential Pressure Transmitter)

รายละเอียดของ DP Cell (Differential Pressure Transmitter) มีดังนี้

Model STD920 / 921

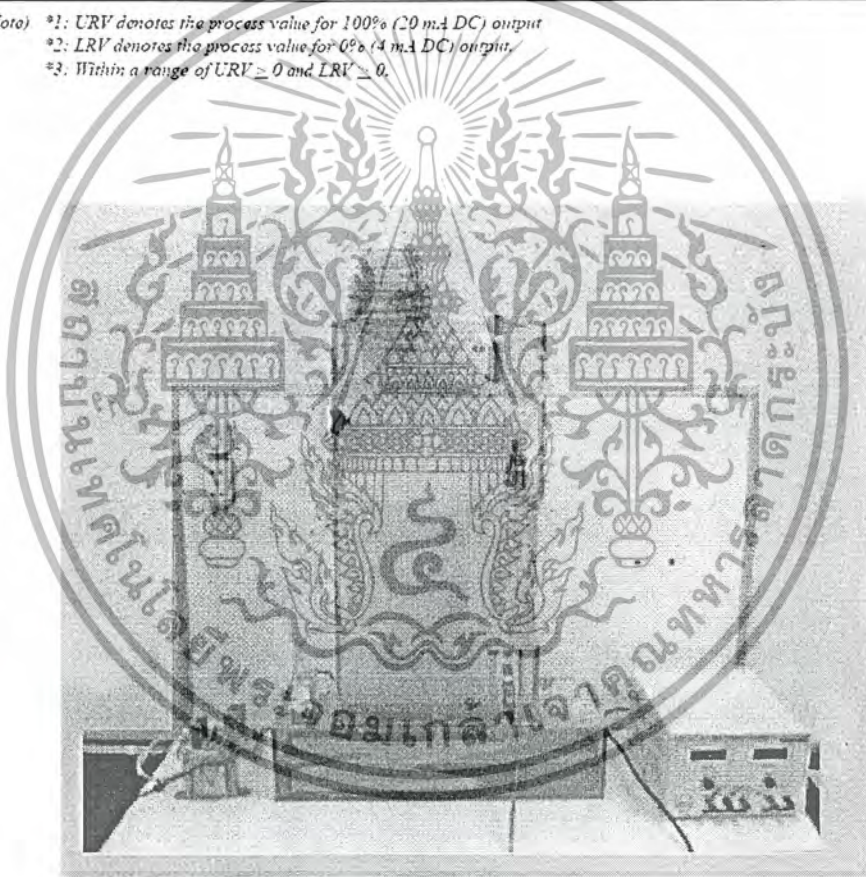
(Material of wetted parts: Diaphragm: SUS316L, Others: SUS316)

Accuracy (*3)	Linear output: $= 0.075\%$ (For $\gamma \geq 50.0kPa$ {5000mmH ₂ O}) $= 0.1\%$ (For $50.0kPa$ {5000mmH ₂ O} $\cdot \gamma \geq 5.0kPa$ {500 mmH ₂ O}) $= 0.025 - 0.075 \cdot \frac{5.0}{\gamma} \%$ (For $\gamma < 5.0kPa$ {500 mmH ₂ O}) Square-root output: When output is 50 to 100%: same as the linear output When output is 7.1 to 50%: linear output $\cdot \frac{50}{\text{square-root-output}} \%$ When output is less than 7.1%: dropout
Temperature characteristics (Shift from the set range) (*3) Change of 55°C	Zero shift: $= 0.25 - 0.3 \cdot \frac{12.5}{\gamma} \%$ Combined shift: (including zero and span shifts) $= 0.8\%$ (For $\gamma \geq 12.5kPa$ {1250 mmH ₂ O}) $= 0.35 - 0.45 \cdot \frac{12.5}{\gamma} \%$ (For $\gamma < 12.5kPa$ {1250 mmH ₂ O})
Static pressure effect (Shift with respect to Setting range) (*3) Change of 7 MPa {70 kgf/cm ² }	Zero shift: $= 0.03 - 0.17 \cdot \frac{20}{\gamma} \%$ Combined shift: (including zero and span shifts) $= 0.4\%$ (For $\gamma \geq 20.0kPa$ {2000 mmH ₂ O}) $= 0.03 - 0.37 \cdot \frac{20}{\gamma} \%$ (For $\gamma < 20.0kPa$ {2000 mmH ₂ O})

Nota) *1: URV denotes the process value for 100% (20 mA DC) output

*2: LRV denotes the process value for 0% (4 mA DC) output.

*3: Within a range of URV > 0 and LRV > 0.



รูปที่ 3.10 แสดงรูปของ Plant จริงของการจำลองกระบวนการควบคุมระดับน้ำแบบมีโมดูลไร้

ผลตอบสนอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ทฤษฎีและหลักการของ LabVIEW

LabVIEW ย่อมาจาก Laboratory Virtual Instrument Engineering Workbench โปรแกรมที่พัฒนาขึ้นโดยใช้ LabVIEW จะเรียกว่า Virtual Instrument หรือเรียกย่อ ๆ ว่า VI ซึ่งหมายถึงเครื่องมือวัดเสมือน ดังตัวอย่างจากรูปนี้เป็น Oscilloscope ที่ได้ทำการสร้างขึ้นบนหน้าจอคอมพิวเตอร์

LabVIEW จะมี Front Panel ซึ่งเปรียบเสมือนได้กับสิ่งที่ผู้ใช้จะเห็นและควบคุมการทำงาน ผู้ใช้สามารถสร้างรูปแบบขึ้นเองได้อย่างรวดเร็วเพราะ LabVIEW มีส่วนประกอบต่าง ๆ ที่ใช้สำหรับออกแบบหน้าจอบนหน้าจอมากมาย เช่น จอแสดงผลแบบออสซิลโลสโคป, ปุ่มหมุน (Dial) และ สวิตช์ เป็นต้น โดย LabVIEW จะแสดงผลและควบคุมการทำงานผ่านทางคอมพิวเตอร์พื้นที่ส่วนที่เขียนโปรแกรมจะเรียกว่า Block Diagram เปรียบเสมือนกับ Hardware ภายในเครื่องมือวัด โดย LabVIEW จะเขียนโปรแกรมโดยอาศัยรูปภาพ LabVIEW อาศัยหลักการทำงานของเครื่องมือวัดหรือการวัดคุมทำให้ผู้ใช้สามารถออกแบบรูปแบบโปรแกรมตามที่ต้องการหลักการดังกล่าวแบ่งออกเป็น 3 ส่วนใหญ่ ๆ คือ

1. Acquisition ซึ่งเป็นส่วนที่รับข้อมูล (Input) จากสิ่งแวดล้อมภายนอกเข้าสู่ระบบในที่นี้คือ คอมพิวเตอร์ โดยข้อมูลที่เข้าสู่ระบบนี้อาจจะมาจาก การ์ด DAQ (สำหรับสัญญาณทางไฟฟ้า), การ์ด IMAQ (สำหรับข้อมูลประเภทรูป) หรือ GPIB (สำหรับควบคุมเครื่องมือวัด)
2. หลังจากที่ได้รับข้อมูลแล้วอาจจะผ่านฟังก์ชันในการทำ Analysis หรือ วิเคราะห์ข้อมูล ซึ่งจะแสดงผลในรูปที่สื่อความหมายในสิ่งที่ผู้ใช้งานสามารถนำไปแสดงแทนสิ่งที่วัดได้และใช้งานได้
3. Presentation คือการแสดงผลในรูปแบบที่เป็นประโยชน์ต่อผู้ใช้งาน โดยอาจแสดงบนหน้าจอคอมพิวเตอร์ เช่น DMM (Digital Multimeter) แสดงผลเฉพาะสัญญาณที่วัดได้โดยไม่จำเป็นต้องรู้ความสัมพันธ์กับเวลา หรือ Oscilloscope แสดงผลของข้อมูลที่สัมพันธ์กับเวลา หรือ Spectrum Analyzer จะแสดงสัญญาณในรูปความถี่หรือการพิมพ์ออกมาเป็นรายงานหรือเก็บข้อมูลในฮาร์ดดิสก์

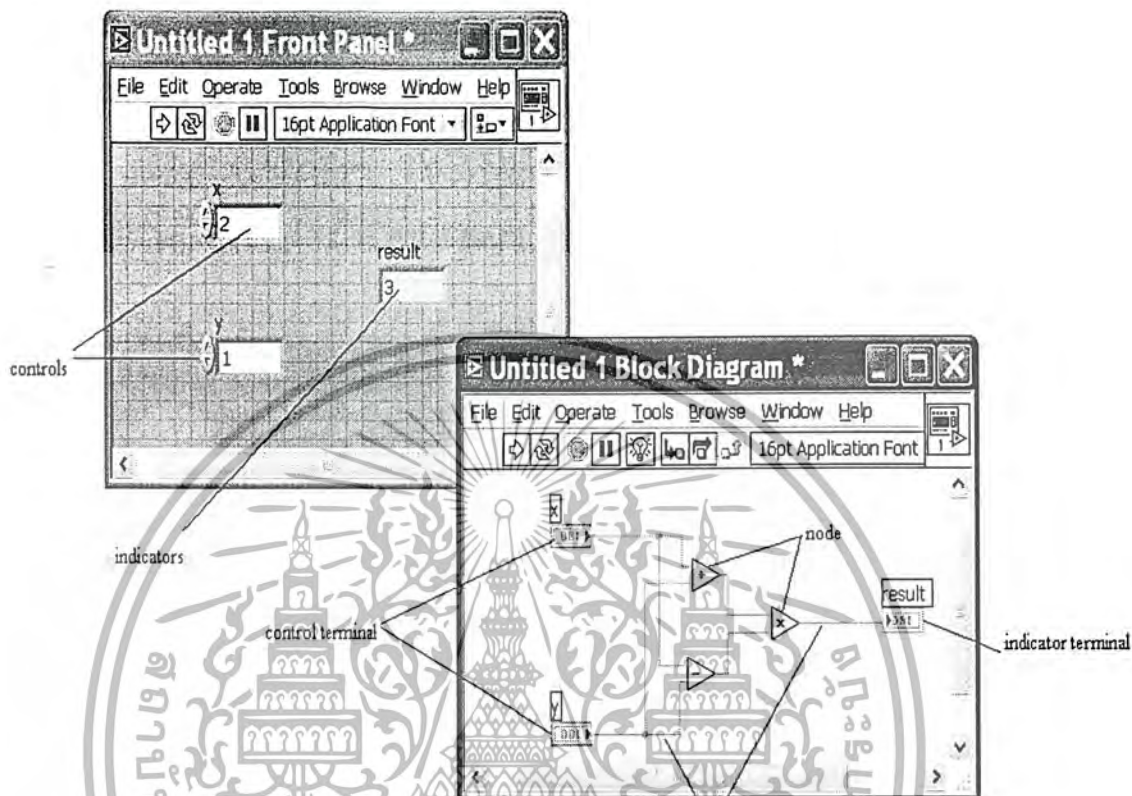
3.4 การสร้าง VI

VI (Virtual Instrument) ประกอบด้วย 2 ส่วนคือ

3.4.1 ฟรอนต์พานเนล (Front panel)

ใช้สำหรับสร้างคอนโทรล (control) หรือ อินดิเคเตอร์ (indicators) ของนิวเมอริก (Numeric), บูลีน (Boolean), สตริง (String), กราฟ (Graph), เทเบิล (Table) หรืออื่นๆที่อยู่บน

คอนโทรลเพลท ลงบน Front Panel เพื่อใช้รับผิดชอบระหว่างคอมพิวเตอร์กับผู้ใช้งาน (user interface)



รูปที่ 3.11 แสดง Front panel และ Block diagram

3.4.2 บล็อกไดอะแกรม (Block diagram)

เป็นส่วนที่เก็บตัวโปรแกรม (source code) โดยจะอยู่ในรูปแบบของกราฟิกโปรแกรมมิ่ง ประกอบด้วย โหนด (Nodes) เทอมินอล (Terminals) และไวร์ (Wires)

3.4.2.1 โหนด

แบ่งเป็น 4 ชนิด คือ

1. ฟังก์ชัน (Function) ติดตั้งมาพร้อมกับ LabVIEW
2. SubVI โหนด คือ VI ที่สร้างขึ้นเองและถูกนำมาใช้เป็น SubVI ในภายหลัง
3. สตริกเจอร์ (Structure) เช่น Case , loop , Sequence ที่ใช้ในการควบคุมการทำงาน ของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Code interface node (CIN) ใช้ติดต่อกันระหว่าง LabVIEW และ user supplied code ของ ภาษา C

3.4.2.2 Terminal

เป็นตัวรับ-ส่งข้อมูลระหว่างบล็อกโคโอะแกรมและ Front Panel หรือ ระหว่าง Node ในบล็อกโคโอะแกรมแบ่งเป็น 2 ชนิด คือ Control หรือ Indicator Terminal ซึ่งจะถูกสร้างจาก Front Panel 2.2.1 Terminal Node จะเกิดจากฟังก์ชันต่างๆที่สร้างขึ้น บล็อกโคโอะแกรม

3.4.2.3 ไวลร์

คือทางเดินของข้อมูลระหว่าง Terminals ซึ่งเป็นแบบส่งทางเดียว สีและรูปแบบของ Wiring จะบอกถึงชนิดของข้อมูลที่ส่งผ่านโดยใช้ wiring tool (รูปหลอดค้ำย) ใน ทูลเพลท (tool palette) เป็นตัวต่อเชื่อมข้อมูลระหว่าง Terminals ต่างๆ

3.4.2.3.1 สีของ Control หรือ Indicator

- สีน้ำเงิน แทนตัวเลขที่เป็นจำนวนเต็ม
- สีส้ม แทนตัวเลขที่มีค่าทศนิยม
- สีเขียว แทนลอจิก (logic) หรือ boolean
- สีชมพู แทนค่าของสตริง (string)

3.4.2.3.2 รูปแบบของเชื่อมต่อ (wiring)

- เส้นบางแทนค่าจำนวนเดียว
- เส้นหนาแทนค่าของอาร์เรย์ ขนาด 1 มิติ
- เส้นคู่แทนค่าของอาร์เรย์ ขนาด 2 มิติ

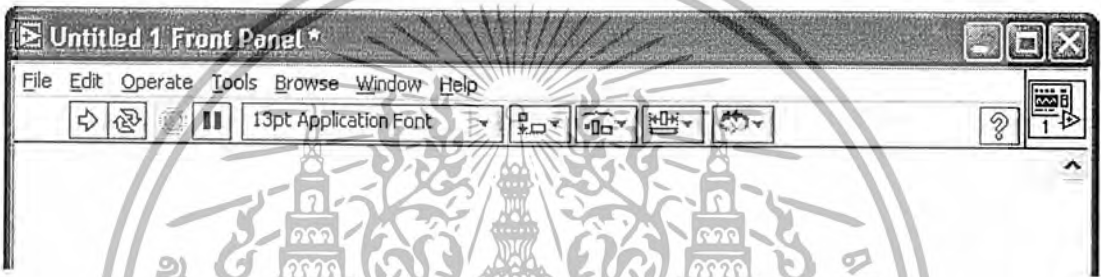
การเชื่อมต่อทำโดยใช้ Wiring Tool เลื่อนไปให้ตรง Terminals แรก สังเกตให้ Terminal กระทบแล้วคลิกเมาส์ จากนั้นให้ลากไปยัง Terminals ที่สอง โดยจะเกิดเส้นประสีดำลากตาม Curser ให้คลิกเมาส์บน Terminals ที่สองในขณะที่ Terminals นั้นกำลังกระทบ ข้อแตกต่างระหว่าง Control กับ Indicator บล็อกโคโอะแกรมคือ Control จะมีขอบเป็นเส้นหนา Indicator จะมีขอบเป็นเส้นบาง

การเก็บข้อมูลโดยปกติ LabVIEW จะเก็บเป็นไฟล์ 2 ชนิด คือ












1. VI file : เก็บเป็นไฟล์ที่มีนามสกุล เป็น *.VI
2. VI libraries : เป็นไลบรารีไฟล์ที่รวมเอาหลายๆ VI ไว้ในไฟล์เดียวกันโดยจะมีนามสกุลเป็น *.lbb

LabVIEW สามารถเขียนโปรแกรมได้จากการสร้างฟรอนต์พานอลซึ่งจะใช้เป็นที่สร้างคอนโทรลและอินดิเคเตอร์สำหรับรับและแสดงข้อมูลระหว่างผู้ใช้งานและเครื่องคอมพิวเตอร์ ส่วนบล็อกไดอะแกรม จะเป็นที่ใช้เขียนโปรแกรมเพื่อให้คอมพิวเตอร์ทำงานตามที่ต้องการในการเก็บโปรแกรมลงไฟล์ทำได้ทั้งแบบ VI file หรือ VI libraries




3.5 องค์ประกอบของ Front Panel Toolbar



รูปที่ 3.12 แสดง Front Panel Toolbar






-  โปรแกรมสามารถ Run ได้เมื่อ Click ปุ่มบน Toolbar
-  โปรแกรมกำลัง Run อยู่
-  โปรแกรมกำลัง Run อยู่ SubVI
-  โปรแกรม Error ถ้า Click จะแสดงรายการ Error List
-  โปรแกรมสามารถ Run แบบต่อเนื่อง (Continuous) เมื่อ Click ปุ่มนี้
-  กำลัง Run อย่างต่อเนื่องจนกระทั่ง Click ปุ่ม Stop/Pause
-  เมื่อโปรแกรมกำลัง Run อยู่สามารถหยุดการทำงานโปรแกรมโดยการ Click ปุ่ม Stop
-  หยุดชั่วคราว เพื่อตรวจสอบการทำงานของโปรแกรม ซึ่งสามารถ Run ต่อไปได้เมื่อ Click ปุ่ม หรือคลิกปุ่ม อีกครั้ง
-  13pt Application Font เลือกหรือแก้ไขรูปแบบอักษรที่ต้องการ
-  จัดแนวของสิ่งต่าง ๆ (Objects)
-  จัดระยะทางระหว่างสิ่งต่าง ๆ (Objects)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-  ปรับขนาดสิ่งต่าง ๆ ที่อยู่บน Front Panel หลาย ๆ สิ่ง ซ้อนกัน
-  จัดลำดับของ Objects ที่วางซ้อนกัน
-  แสดง Context Help

3.6 องค์ประกอบของ Block Diagram Toolbar

สิ่งที่เพิ่มเติมจาก Front Panel Toolbar ในส่วนของ Block Diagram มีดังต่อไปนี้

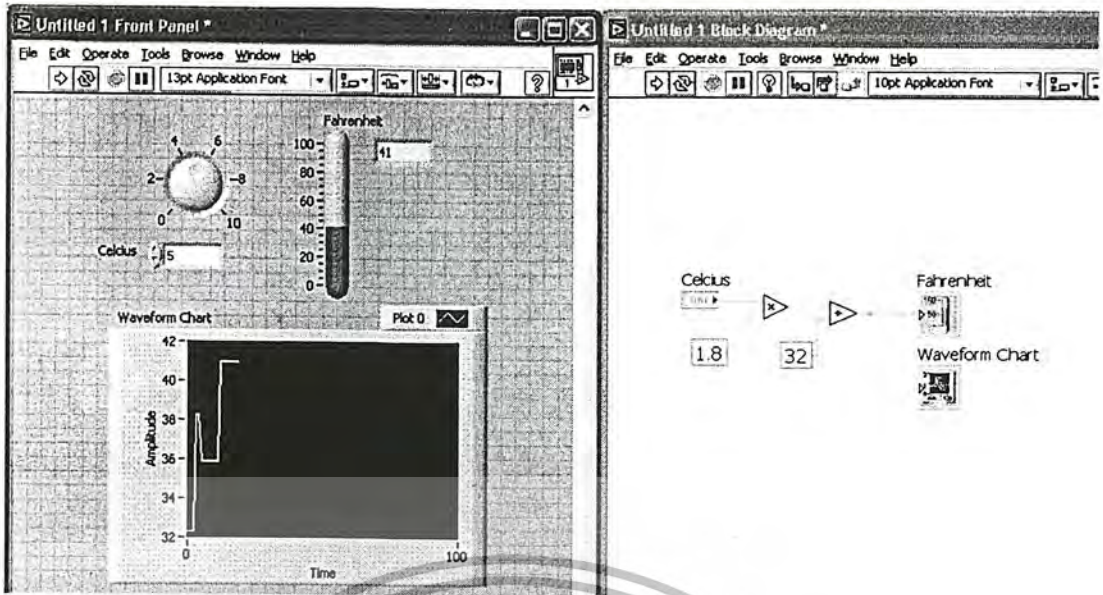
-  Highlight Execution เมื่อ Click แล้ว ในการ Run Program จะทำให้โปรแกรม Run
-  ข้างลงเพื่อให้เราเห็นถึงการไหลของข้อมูล และทำงานของโปรแกรม เมื่อ Click แล้วจะออกจากโหมด (Mode) ที่โปรแกรม Run ข้าง (Highlight Execution)
-  เป็นการ Run โปรแกรมแบบทีละคำสั่ง และจะลงไปถึงคำสั่งของ SubVI
-  เป็นการ Run โปรแกรมแบบทีละคำสั่ง แต่จะไม่ลงไปถึงคำสั่งของ SubVI
-  ให้ Run จนออกจาก Loop และ SubVI

3.7 การพัฒนาโปรแกรม

3.7.1 หลักการทำงานของ Dataflow Programming

โปรแกรมที่เขียนขึ้นด้วย LabVIEW จะทำงานโดยอาศัยหลักการของ Dataflow ซึ่งมีหลักการดังต่อไปนี้ คือ

1. ฟังก์ชันหรือ SubVI จะทำงานเมื่อมีข้อมูล (Input)
2. เมื่อฟังก์ชันหรือ SubVI ทำงานเสร็จจะให้ข้อมูล (Output) ไปยังฟังก์ชันหรือ SubVI อื่นๆที่ต้องการข้อมูล
3. ข้อมูลจะถูกส่งผ่านโดยสาย (Wiring)



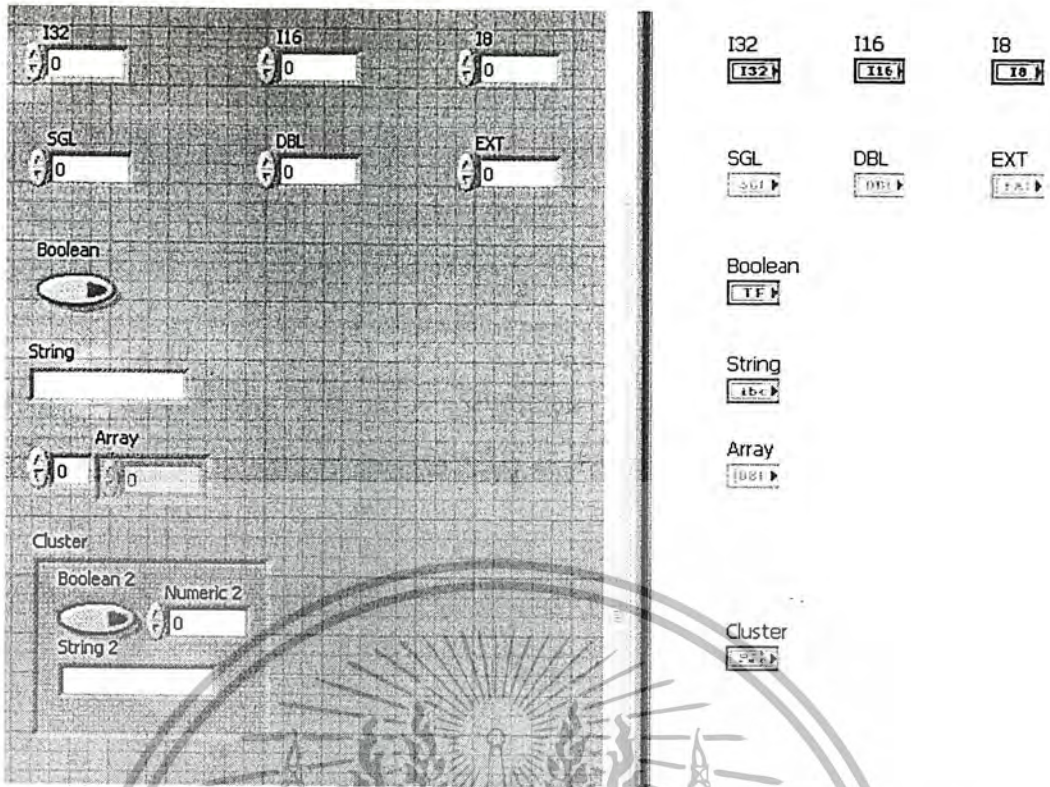
รูปที่ 3.13 แสดง หลักการทำงานของ Dataflow Programming

จากภาพขั้นตอนการทำงานของ โปรแกรมจะเป็นดังนี้

1. ฟังก์ชันคูณ (Multiply) จะทำงานก่อนฟังก์ชันบวก เพราะฟังก์ชันคูณมีข้อมูล (Input) พร้อม แต่ฟังก์ชันบวกต้องรอ Output จากฟังก์ชันคูณ (หลักการ Dataflow ข้อที่ 1)
2. หลังจากฟังก์ชันคูณทำงานเสร็จจะส่งผ่านข้อมูลไปยังฟังก์ชันบวก (หลักการ Dataflow ข้อที่ 2 และ 3)
3. ฟังก์ชันบวกทำงาน (หลักการ ข้อที่ 1) เพราะมีข้อมูลพร้อม
4. หลังจากฟังก์ชันบวกทำงานจะส่งผลลัพธ์ไปให้ Terminal ทั้ง 2 คือ Fahrenheit และ Waveform Chart พร้อมกัน (หลักการ ข้อที่ 2 และ 3)

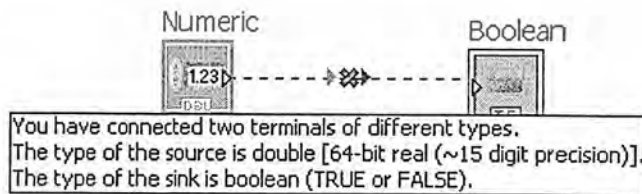
3.7.2 Data Type (ประเภทของข้อมูล)

เช่นเดียวกับกับภาษาโปรแกรมอื่นๆ เช่น ภาษา C ที่มีประเภทของข้อมูล ประเภทของข้อมูลใน LabVIEW ที่สำคัญจะประกอบด้วย Numeric , Boolean , String และ Waveform ซึ่งแต่ละรูปแบบของประเภทข้อมูลจะมีการแยกสีให้แตกต่างกัน (Color Code)



รูปที่ 3.14 แสดง Data Type (ประเภทของข้อมูล)

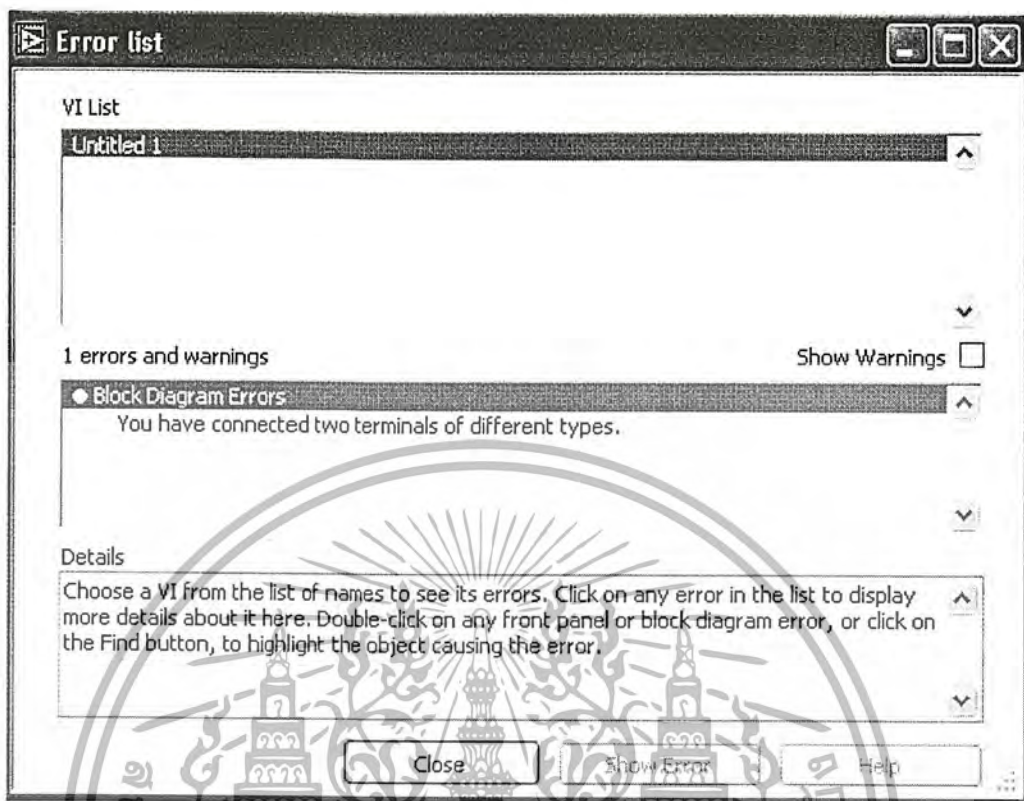
ซึ่งแต่ละประเภทของข้อมูลจะมีฟังก์ชันที่แยกตามประเภทกัน รวมทั้งสีของ (Wiring) จะขึ้นอยู่กับประเภทของข้อมูล ประเภทของข้อมูลมีความสำคัญในการเขียนโปรแกรม เพราะฟังก์ชัน หรือ SubVI รับรูปแบบ Input และให้ Output แตกต่างกันไป เช่น ฟังก์ชันคุณรับข้อมูลประเภทตัวเลขหรือ Array ของตัวเลข ซึ่งถ้านำเอาข้อมูลประเภทตัวอักษร (String Data Type) มาป้อนที่ด้าน Input Terminal แล้ว LabVIEW จะแสดงเส้น (Wire) เป็นเส้นประ ซึ่งเราสามารถดูคำอธิบายได้โดยการเลื่อน Cursor ไปเหนือเส้นประนั้น



รูปที่ 3.15 แสดงการ Wiring ที่ผิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือ Click ที่ปุ่ม Run เพื่อให้เห็น Error List

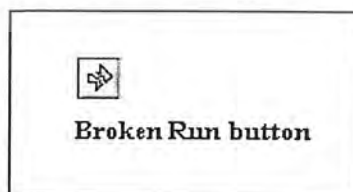


รูปที่ 3.16 แสดง Error List

3.8 เทคนิคของการดีบั๊กกิ้ง (Debugging)

การดีบั๊กกิ้ง คือ การตรวจสอบหาข้อผิดพลาดของโปรแกรม ในกรณีที่โปรแกรมที่เขียนขึ้นมีปัญหาโดย LabVIEW จะมีเครื่องมือต่างๆ ในการ Debugging ดังนี้

3.8.1 Finding Error

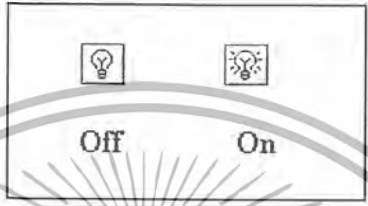


รูปที่ 3.17 แสดง Finding Error

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อโปรแกรมไม่สามารถรันได้ เครื่องหมายลูกศรแตก (Broken arrow) จะปรากฏขึ้นที่ปุ่ม รัน (Run button) บน Front Panel เราสามารถดูว่ามีข้อผิดพลาดอะไรบ้าง โดยกดปุ่มเครื่องหมาย ลูกศรแตกนั้น และเมื่อคลิกเมาส์ที่ข้อผิดพลาดเพื่อเลือก และกดปุ่ม Find เพื่อทำให้เกิดไฮไลต์ที่ ออบเจกต์หรือ Terminal ที่ตรวจพบว่ามีปัญหา

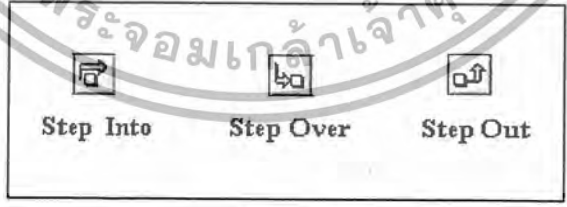
3.8.2 Execution Highlight



รูปที่ 3.18 แสดง Execution Highlight

เราสามารถให้บล็อกโคดอะแกรมแสดงการทำงานให้เห็นได้โดยการคลิกที่ปุ่ม Execution Highlight ปุ่มนี้จะเปลี่ยนไปเป็นเหมือนที่แสดงในรูป Execution Highlight มักจะใช้ร่วมกับ singer step mode เพื่อดูข้อมูลที่เกิดขึ้นขณะที่โปรแกรมรัน

3.8.3 Execution Highlight button Singer Step Through



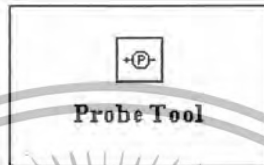
รูปที่ 3.19 แสดง Execution Highlight button Singer Step Through

สำหรับวัตถุประสงค์ในการDebuggingเรามักจะต้องการดูการทำงานของบล็อกโคดอะแกรม โหนดต่อโหนด ซึ่งเรียกกันว่า Singer Stepping ในการที่จะให้โปรแกรมทำงานแบบ Single Step ให้กดที่ปุ่ม Steo Into หรือ Step Over ซึ่งจะมีผลให้โหนดแรกที่เริ่มทำงานกระพริบหมายถึงพร้อมที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะทำงานต่อไป เมื่อต้องการให้โปรแกรมทำงานต่อ ก็ให้กดปุ่ม Steo Into หรือ Step Over อีกครั้ง ถ้าในกรณีที่โหนดต่อไปที่จะทำงานเป็น SubVI และเรากด Steo Over โปรแกรมจะทำงานโดยข้ามการทำงานใน SubVI นั้นไปเลย และส่งเพียงผลที่ได้จากการทำงานของ SubVI ออกมาเท่านั้น แต่ถ้าเลือกกด Step Into โปรแกรมจะทำการเปิด SubVI นั้นและแสดงการทำงานของ SubVI นั้นกดปุ่ม Step Out เพื่อให้โปรแกรมจบการทำงานของบล็อกโคอะแกรมนั้น ๆ

3.8.4 Button Probe



รูปที่ 3.20 แสดง Execution Highlight button Singer Step Through

เราสามารถที่จะดูข้อมูลที่อยู่ในจุดใดของบล็อกโคอะแกรมได้โดยใช้ Probe การสร้าง Probe ทำได้โดยเลือก Probe Tool จาก Tool Platte และคลิกลงบนไวร์ที่ต้องการดูข้อมูล หรืออาจจะสร้างได้อีกวิธีคือ คลิกเมาส์ขวาตรงตำแหน่งที่ต้องการสร้าง Probe เพื่อแสดงป๊อปอัพเมนู และเลือก Custom Probe

3.8.5 Breakpoint



รูปที่ 3.21 แสดง Breakpoint

บางครั้งเราอาจจะต้องการที่จะหยุดการทำงานของโปรแกรมที่จุดใดๆในโปรแกรม เพื่อใช้ Probe ดูข้อมูลหรือรันแบบ Single Step ต่อไป ให้ใช้ Breakpoint Tool คลิกในตำแหน่งที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

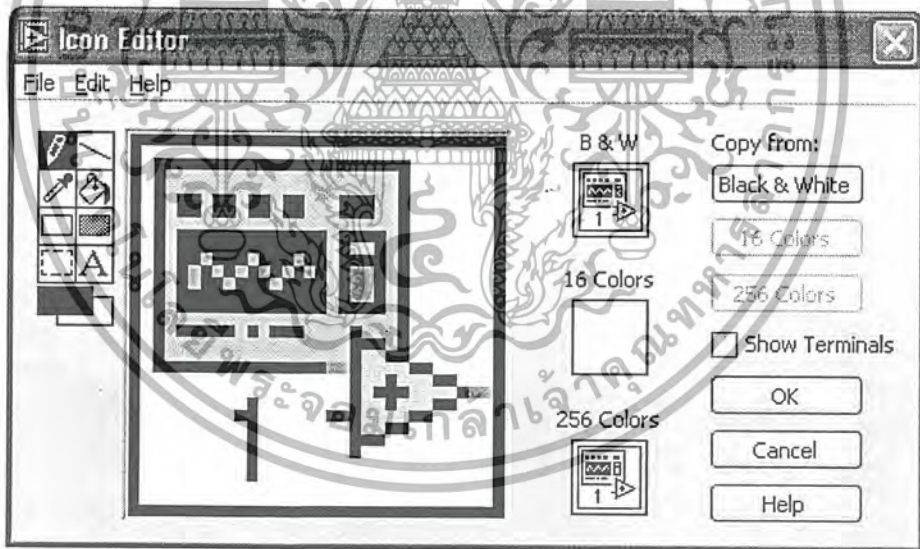
เซ็ท หรือเตลีสร์ Breakpoint ซึ่งเมื่อเซ็ทแล้วจะปรากฏให้เห็นเป็นจุดสีแดงกรณีที่ใช้กับไวร์หรือเป็นกรอบสีแดงรอบโหนดหรือโคอะแกรม

3.9 การสร้างโปรแกรมย่อย (Sub VI)

ในการสร้างโปรแกรมด้วย LabVIEW เพื่อใช้งานนั้น มักจะสร้างโปรแกรมย่อย (subVI) ขึ้นมาเพื่อสามารถที่จะนำมาเรียกใช้อีกโปรแกรมย่อยจะคล้ายกับฟังก์ชันหรือซับริoutine) ในโปรแกรมต่างๆ ไป

VI ที่นำมาใช้เป็นโปรแกรมย่อยจะต้องมีไอคอน (icon) เพื่อแทนตัวโปรแกรมย่อยในบล็อกโคอะแกรมของ VI ที่เรียกใช้และยังต้องมีคอนเน็กเตอร์ กับ เทอร์มินอล เพื่อที่จะในการรับหรือส่งค่าไปยัง VI ที่เรียกใช้

ไอคอน-ทุก VI จะมีไอคอนอยู่ที่มุมขวาบนโดย LabVIEW จะสร้างมาให้อัตโนมัติ แต่ผู้ใช้งานสามารถแก้ไขได้โดยดับเบิลคลิกเมาส์ที่ไอคอน หรือคลิกเมาส์ขวาและเลือก Edit icon จากป๊อปอัพเมนู เมื่อเข้าไปใน edit icon แล้วสามารถที่จะใช้ หลุดต่างๆ เช่น pencil , line, dropper, text เป็นต้นเพื่อทำการวาดรูปบนไอคอน หรืออาจจะก๊อปปี้ภาพที่มีนามสกุล .BMP, .WMF , .EMF หรือ PCT มาใส่ลงบนไอคอนได้



รูปที่ 3.22 แสดง Icon Editor

Connector-เป็นตัวที่ใช้ติดต่อ รับ-ส่งข้อมูลระหว่าง Control และIndicator ของโปรแกรมย่อยนั้นๆ ไปยัง VI ที่เรียกใช้งาน การกำหนด Connector ว่าจะเป็น Control หรือ Indicator ใดๆ ทำโดยเลือก Show Connector จากป๊อปอัพเมนู LabVIEW จะนับจำนวน Connector ให้อัตโนมัติ แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ผู้ใช้งานกำหนดว่า Connector ใด แทนตัว Control หรือ Indicator โดยใช้เมาส์คลิกเลือกจับคู่กัน รูปแบบของจำนวน Terminal สามารถที่จะแก้ไขหรือเปลี่ยนแปลงได้โดยเลือกจากป๊อปอัพเมนูของ Connector ซึ่งประกอบด้วย

1. Add Terminal เพื่อเพิ่ม Terminal เมื่อมีการเพิ่ม Control หรือ Indicator
2. Remove Terminal เพื่อลบ Terminal ออกจาก Connector
3. Patterns เพื่อเลือกรูปแบบและจำนวน Terminal
4. Rotate 90 degree, Flip Horizontal และ Flip Vertical เพื่อหมุน Connector
5. Disconnect All Terminal เพื่อใช้ลบ Control และ Indicator ออกจากทุก Terminal
6. Disconnect This Terminal เพื่อใช้ลบ Control หรือ Terminal ของ Terminal ที่เมาส์

เคอร์เซอร์ชี้อยู่

7. This Connector is Required เพื่อกำหนดว่าเวลาใช้งานจะต้องมีค่าส่งมาจาก VI ที่เรียกใช้
8. This Connector Recommended, optional เพื่อกำหนดว่าเวลาใช้งานจะมีค่าส่งมาจาก VI ที่เรียกใช้หรือไม่ก็ได้



รูปที่ 3.23 แสดง เทอมินอล ป๊อปอัพเมนู

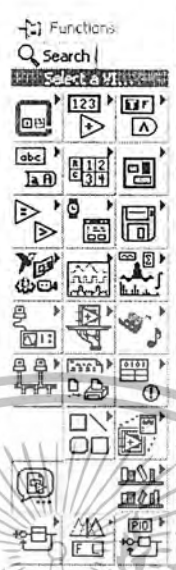
โปรแกรมย่อยสร้างได้โดยการเขียนโปรแกรมให้เสร็จก่อน จากนั้นทำการแก้ไขไอคอน และทำการกำหนดคอนเน็กเตอร์ของคอนโทรลและอินดิเคเตอร์บนพอนต์ฟานเนล

3.9.1 การใช้งานโปรแกรมย่อย (SubVI)

ในกรณีที่ไม่โปรแกรมมีการทำงานชนิดเดียวกันหลาย ๆ ครั้งและยังสามารถสร้างส่วนของโปรแกรมนั้นเป็นโปรแกรมย่อยซึ่งจะช่วยให้ใช้เนื้อที่ในการเก็บโปรแกรมน้อยลง ลดเวลาในการเขียนโปรแกรมและง่ายในการแก้ไขข้อผิดพลาดของโปรแกรมนอกจากนี้ยังสามารถเรียกโปรแกรมย่อยจากโปรแกรมย่อยได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับ LabVIEW วิธีการเรียกโปรแกรมย่อยมาใช้งานทำได้โดยเลือก Select a VI... บน function เพลต ต่อจากนั้นให้เรียกโปรแกรมย่อยที่จะใช้งานมาใส่ลงบน VI ที่กำลังเขียนอยู่

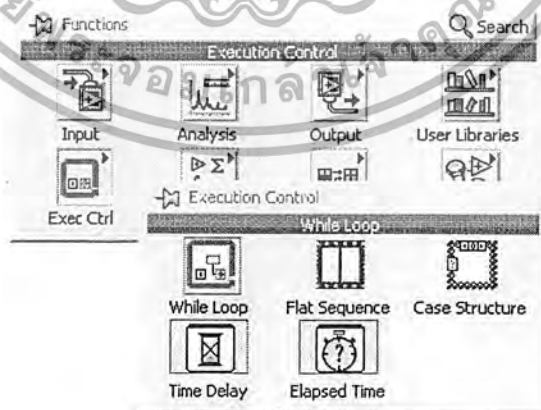


รูปที่ 3.24 แสดงการเรียกใช้งานโปรแกรมย่อย

3.10 ตัวอย่างการใช้งานฟังก์ชันใน LabVIEW

3.10.1 While loop

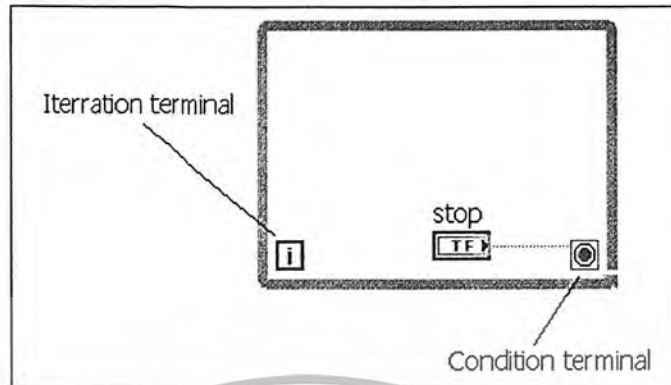
เป็นสตรัคเจอร์ อยู่บน Function เพลต สร้างจาก Structure ซับเพลต เราสามารถสร้างได้ บล็อกโคอะแกรมโดยเลือกจาก Structure ซับเพลตของ Function เพลต



รูปที่ 3.25 แสดงซับเพลตของ Function While loop

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

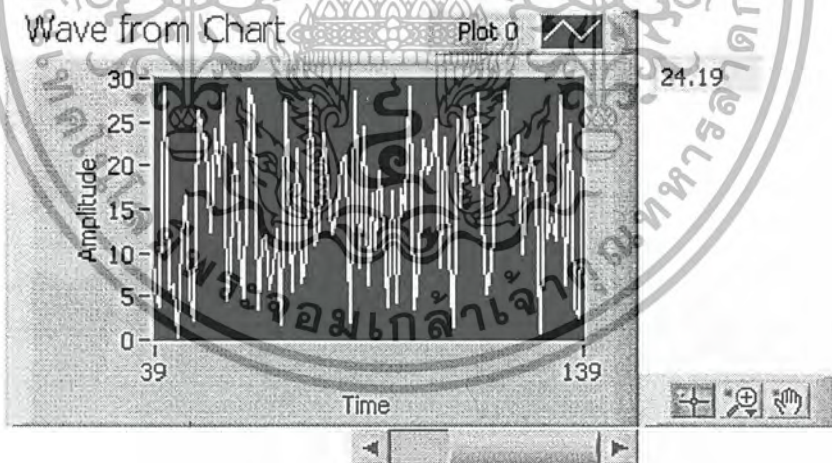
การทำงานจะสั่งให้ทำงานภายในลูปเมื่อได้รับค่าจริงเข้าที่คอนดิชันเทอร์มินอล (condition terminal) และจะหยุดการทำงานเมื่อได้รับค่าเท็จส่วนจำนวนครั้งที่ลูปทำงานจะให้ค่าออกมาที่อิเทอร์มินอล (Iteration terminal) โดยจะมีค่าเริ่มต้นที่ศูนย์



รูปที่ 3.26 แสดง Function While loop บนหน้า Diagram

3.10.2 Waveform Chart

เป็นนิวมเมอริคอินดิเคเตอร์ ที่ใช้แสดงผลข้อมูลที่เป็นตัวเลขเลือกมาจาก graph ชัฟเฟลตของ Control เฟลต

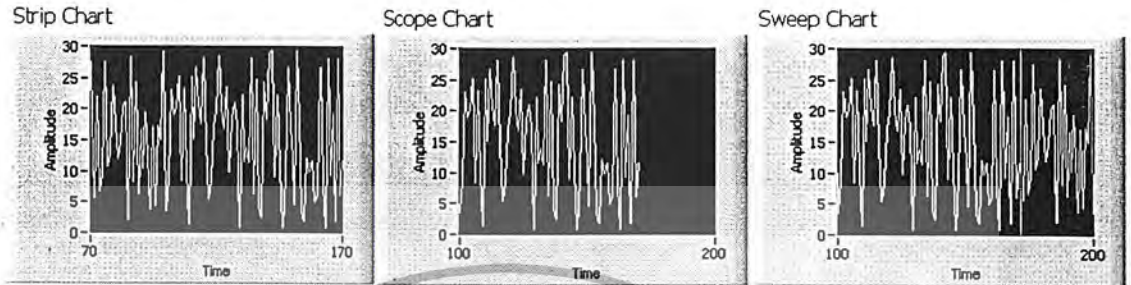


รูปที่ 3.27 แสดง Waveform Chart

การแสดงผลสามารถกำหนดได้ 3 แบบ คือ strip chart, scope chart และ sweep chart โดยเลือกจาก Data Operation >> Update Mode Menu

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Strip chart จะเลื่อนหน้าจอที่แสดงผลไปตามข้อมูลที่รับเข้ามาคล้ายกับกระดาษบันทึกข้อมูลของ recorder
2. Scope chart จะเขียนข้อมูลไปจนเต็มชาร์ต จากนั้นจะลบจอก่อนจึงจะเขียนข้อมูลใหม่
3. Sweep chart คล้ายกับ Scope Chart แต่จะไม่ลบหน้าจอเหมือนเขียนข้อมูลใหม่



รูปที่ 3.28 แสดง Function strip chart, scope chart และ sweep chart บนหน้า Front Panel

3.10.2.1 Single plot chart

เป็นการแสดงข้อมูลค่าเดียวบนชาร์ตสามารถลากเอาต์พุตของนิวเมอริเทอมินอลไปยัง waveform chart เทอมินอลได้เลย

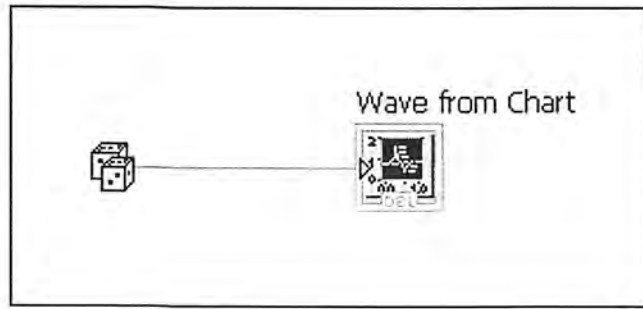


รูปที่ 3.29 แสดง Single plot chart

3.10.2.2 Multiple chart plot waveform chart

เป็นการแสดงข้อมูลมากกว่า 1 ค่าบนชาร์ต โดยใช้ฟังก์ชัน Bundle เพื่อรวมข้อมูลหลายค่าเข้าด้วยกัน Bundle จะอยู่ใน Cluster palette บน Function palette ในส่วนของ Block diagram window การขยายอินพุตของ Bundle ทำได้โดยใช้ Positioning Tool ไป drag ที่มุมของ Bundle แล้วขยายขนาดของอินพุตได้ตามต้องการ

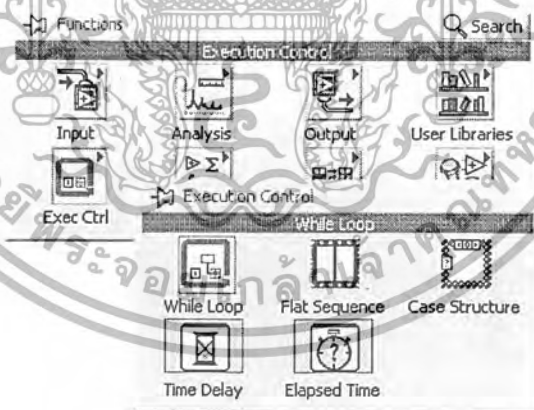
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.30 แสดง Multiple chart plot waveform chart

3.11 การใช้ชิฟต์รีจิสเตอร์(Shift Register)

ชิฟต์รีจิสเตอร์ประกอบด้วย Terminal เป็นคู่ โดยจะอยู่คั่นข้างกันบนขอบแนวตั้งของลูป ข้อมูลจะถูกส่งจากชิฟต์รีจิสเตอร์ Terminal ด้านซ้ายไปยังด้านขวา ชิฟต์รีจิสเตอร์สามารถใช้ได้กับข้อมูลทุกชนิด (มีเฉพาะใน While Loop และ For Loop เท่านั้น) ใช้ส่งค่าจากลูปที่กำลังทำงานอยู่ไปยังลูปถัดไป สร้างโดยคลิกเมาส์ขวาที่ขอบของลูป แล้วเลือก Add Shift Register จากป๊อปอัพเมนู

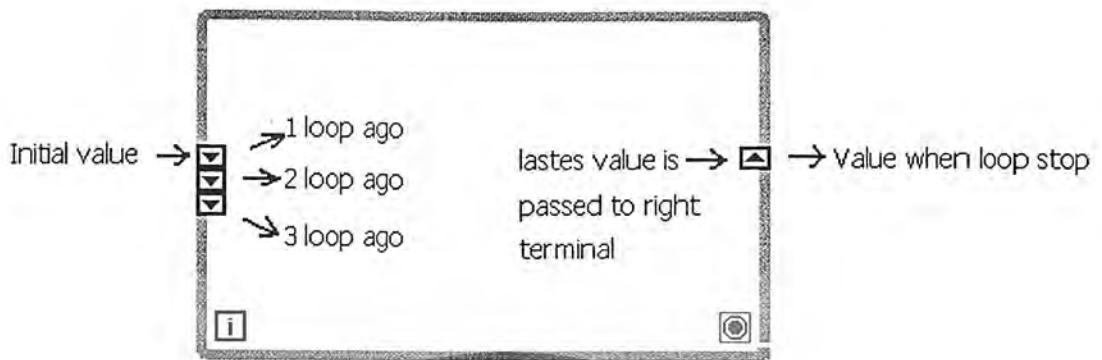


รูปที่ 3.31 แสดง Shift Register

ชิฟต์รีจิสเตอร์สามารถเก็บค่าได้หลายลูป ซึ่งจะมีประโยชน์ในการเก็บข้อมูล (Averaging) ทำโดยกด ป๊อปอัพเมนูของชิฟต์รีจิสเตอร์ด้านซ้าย และเลือก Add Element ในรูปด้านล่างถ้าเรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพิ่มจำนวนของชิฟต์รีจิสเตอร์ให้เท่ากับ 3 Terminal เราก็สามารถนำเอาข้อมูลใน 3 ลูป ที่ผ่านมา มาใช้งานได้



รูปที่ 3.32 แสดงการทำงานของ Shift Register

ค่าเริ่มต้นของชิฟต์รีจิสเตอร์ กำหนดได้โดยใส่ค่าที่ต้องการให้กับชิฟต์รีจิสเตอร์นั้นๆ แต่ถ้าปล่อยให้ชิฟต์รีจิสเตอร์นั้นว่างอยู่ ค่าเริ่มต้นจะถูกกำหนดให้เป็นค่าดีฟอลต์ (default) เช่น ถ้าเป็นบูลีน (boolean) ก็จะเป็นเท็จ แต่ถ้าเป็นนิวเมอริก ก็จะเป็นค่าศูนย์

ชิฟต์รีจิสเตอร์ ใช้ในการส่งค่าตัวแปรได้ทุกชนิด ไม่ว่าจะเป็นตัวเลข, ตัวหนังสือ, ลอจิก ทั้งที่เป็นค่าเดียวหรือเป็นอาร์เรย์จากลูปที่เริ่มต้น ไปยังลูปต่อไปได้เรื่อยๆ จนกว่าจะสิ้นสุดการทำงานของลูป

3.11.1 For Loop

For Loop สร้างจาก Structure ซัพพลาย ของ Function เฟลด์ บนบล็อกไดอะแกรม โดยนำไปล้อมรอบออบเจกต์ที่ต้องการให้ทำงานซ้ำๆ กัน



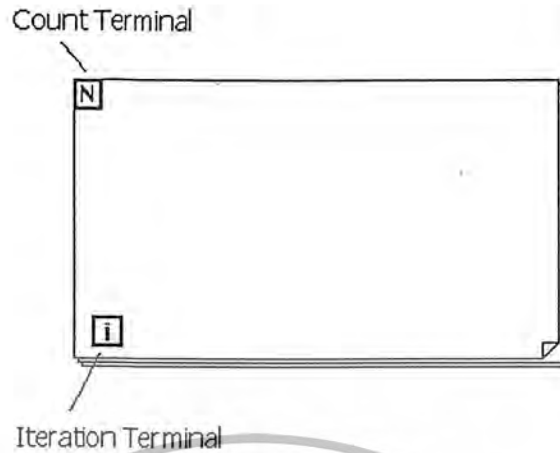
รูปที่ 3.33 แสดง การเลือก For Loop

For Loop จะประกอบด้วยเทอมินอล 2 ตัว คือ

1. Count Terminal เป็นอินพุตเทอมินอลที่ใช้กำหนดจำนวนครั้งในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Iteration Terminal เป็นเอาต์พุตเทอร์มินอลที่ใช้แสดงจำนวนครั้งที่ลูปทำงาน



รูปที่ 3.34 แสดง For Loop

ความแตกต่างระหว่าง For Loop และ While loop คือ For Loop จะมีการกำหนดจำนวนครั้งในการทำงานที่แน่นอน แต่ While loop จะทำงานตามเงื่อนไขที่ได้รับ ตัวอย่างข้างล่างเป็นการใช้ For Loop ในการสุ่มค่าตัวเลขตั้งแต่ 1 ถึง 100 จำนวน 20 ครั้ง และนำมาพล็อตบน Waveform Chart

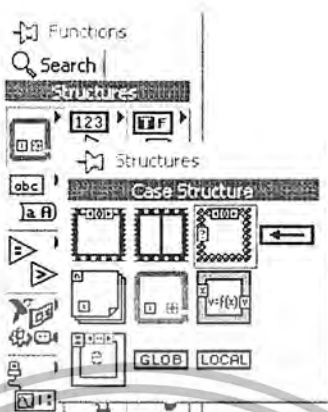


รูปที่ 3.35 แสดงตัวอย่างการใช้ For Loop

เราสามารถเลือกใช้ For Loop และ While loop ในโปรแกรมของเราได้ ขึ้นอยู่กับเงื่อนไขในการทำงานซึ่ง For Loop จะใช้กับการทำงานที่รู้จำนวนครั้งที่แน่นอน

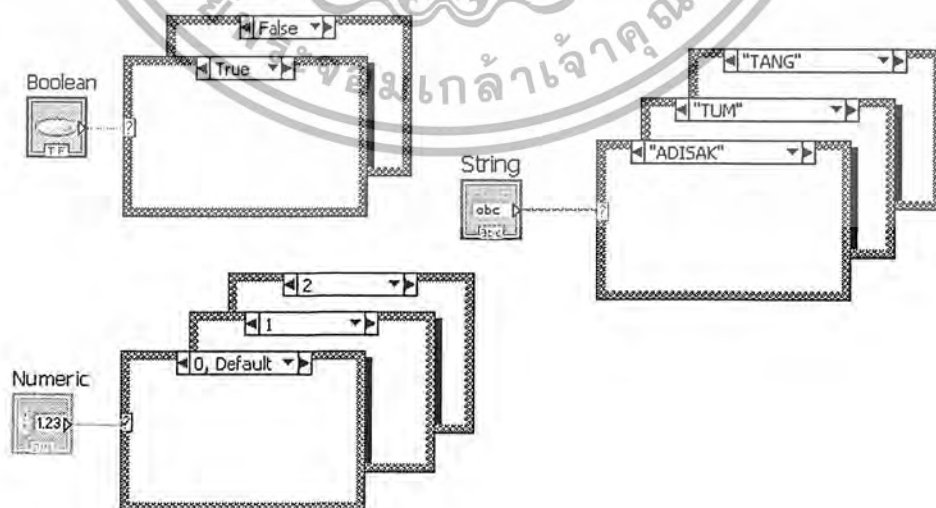
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11.2 Case Structure



รูปที่ 3.36 แสดงการเลือก Case Structure

จะเหมือนกับคำสั่ง if...then...else ในโปรแกรมทั่วไป เลือกสร้างจาก Structure ชัฟเฟลต (Function Plate) โดยสามารถเลือกเงื่อนไขได้ทีละ 1 เงื่อนไขเท่านั้น เมื่อเงื่อนไขทำงานจะเลือกการทำงานที่เงื่อนไขใดเงื่อนไขหนึ่งเท่านั้น โดยจะตรวจสอบว่าจะทำเงื่อนไขใดจากซีเล็กเตอร์เทอม นอล (Selector Terminal) ภายในมีรูปเครื่องหมายค่าตาม ซึ่งสามารถรับค่าได้ทั้งที่เป็น นิวเมอริค , บูลีน หรือ สตริง ถ้าค่าที่ตั้งให้ซีเล็กเตอร์เทอมินอลเป็นบูลีนจะมีเพียงเงื่อนไขจริงและเท็จเท่านั้น แต่ถ้าเป็นนิวเมอริคหรือสตริง จะสามารถสร้างเงื่อนไขได้มากถึง $2^1 - 1$ เงื่อนไข



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

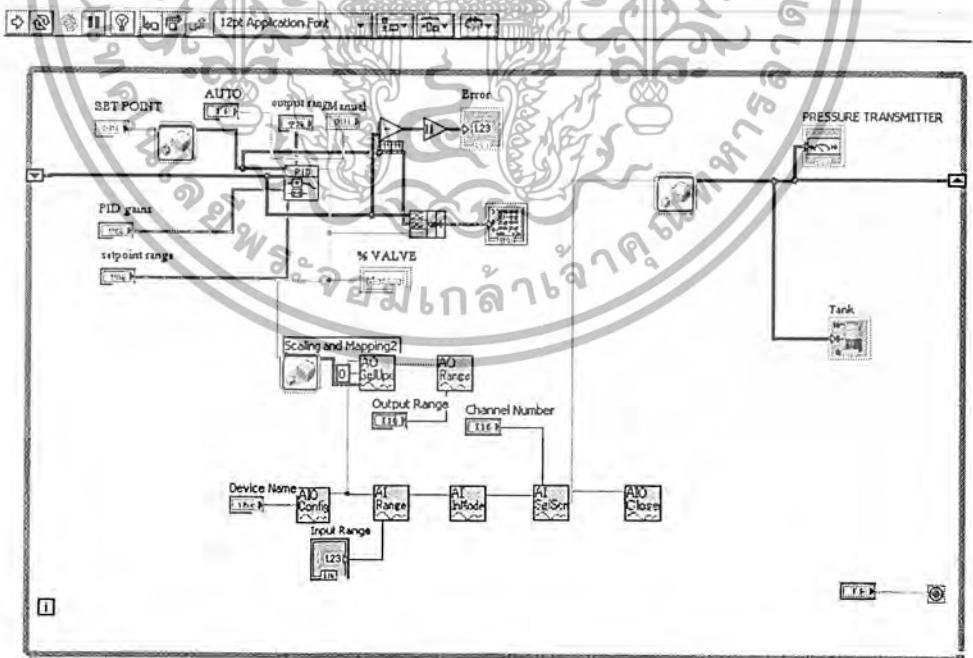
รูปที่ 3.37 แสดง Case Structure ซึ่งสามารถรับค่าได้ทั้งที่เป็นนิวมอริก, บูลีน หรือ สตริง

ในกรณีที่มีการส่งค่าออกจากเงื่อนไขใดเงื่อนไขหนึ่ง จะเกิดเอาต์พุตทันเนล (output tunnel) ที่ขอบของเงื่อนไขที่ส่งข้อมูลออก โดยจะอยู่ที่ตำแหน่งเดียวกันหมด เงื่อนไขใดที่มีข้อมูลส่งมาที่ทันเนลครบทุกเงื่อนไขแล้วจะแสดงเป็นสี่เหลี่ยมสีดำ แต่ถ้าทันเนลเป็นสี่เหลี่ยมสีขาว แสดงว่ายังมีเงื่อนไขที่ยังไม่ได้ส่งค่าที่ทันเนลซึ่งเราจำเป็นต้องส่งค่าให้กับทันเนลจากเงื่อนไขเงื่อนไขที่มีอยู่

Case Structure ใน LabVIEW จะใช้ในการเลือกทำงานตามเงื่อนไขที่เท่ากับค่าที่ถูกส่งมาที่ซีเล็กเตอร์ เทอมินอล (selector terminal) โดยใช้ได้กับค่าที่เป็นนิวมอริก , บูลีน หรือ สตริง ในกรณีที่มีค่าส่งออกจากเงื่อนไข จะเกิดเป็นทันเนลขึ้น ซึ่งต้องทำการส่งค่าจากภายในเงื่อนไข ออกไปยังทันเนลให้ครบทุกเงื่อนไข

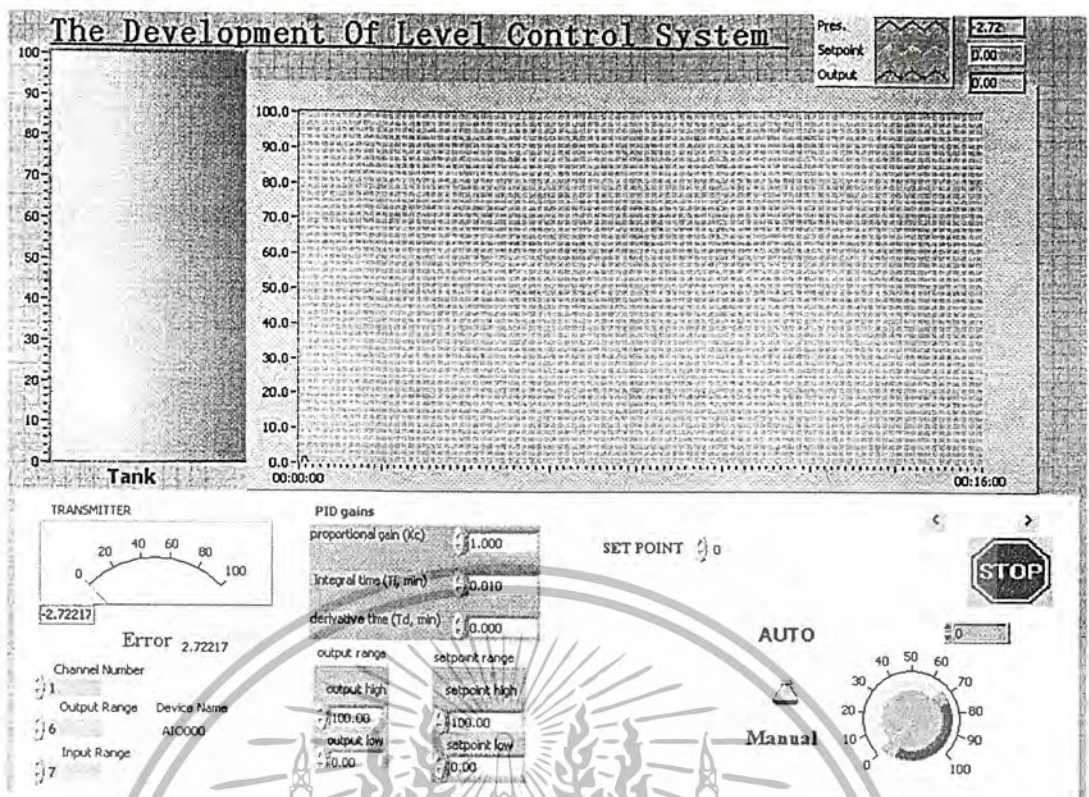
3.12 โปรแกรมที่ใช้ในการควบคุม

โปรแกรมที่ใช้ในการควบคุมกระบวนการนั้นเราใช้โปรแกรม Lab VIEW ในการเขียนและจำลองกระบวนการ จากทฤษฎีที่กล่าวมาข้างต้น โปรแกรมจะประกอบด้วยกัน 2 ส่วนคือ ส่วนที่ 1 โปรแกรมที่เขียนอยู่ในรูปของบล็อกไดอะแกรมที่แสดง ดังรูปที่3.35 และส่วนที่ 2 คือ โปรแกรมที่แสดงอยู่ในรูปของ Front Panel แสดงดังรูป 3.36



รูปที่ 3.38 แสดง โปรแกรมที่ใช้ควบคุมกระบวนการในรูปของบล็อกไดอะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.39 แสดงโปรแกรมที่ใช้ควบคุมกระบวนการที่แสดงอยู่ในรูปของ Front Panel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

การออกแบบและพัฒนาชุดทดลองกระบวนการควบคุมระดับน้ำแบบมีโมดูล ไว้ผลตอบสนอง

4.1 สมการคุณลักษณะของกระบวนการ

วิธีการหาแบบจำลองของกระบวนการโดยใช้วิธี Integrat step โดยการใช้งาน simulink ในโปรแกรม LABVIEW เพื่อหาค่าพารามิเตอร์ K_c , T_i , T_d โดยการป้อนอินพุตแบบ unit step ให้กับกระบวนการจริง แล้วดูผลตอบสนองของกระบวนการเมื่อเข้าสู่สภาวะคงตัว จากรูปที่ 4.1 ค่าคือค่าที่กระบวนการเข้าสู่สภาวะคงตัว Step 45% to 65%

4.2 ความยาวสาย $L = 0.5 M$



รูปที่ 4.1 แสดงวิธีการหาแบบจำลองกระบวนการแบบเปิด

Show Step 45% steady state change to 65% steady state Response to step input

$L = 0.5 M$, Dead Time = 4 at flow 50 %

T_d = Deadtime in minutes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

T = Time constant in minutes

K = Process gain = (Change in output) / (Change in PV)

Lower = 0 %

Hight = 57.67%

$T_d = 7/60 = 0.117$

$T = 0.632 * (0.584 - 0.048) = 0.338 \text{ Min}$

$K = (\text{Change in output}) / (\text{Change in PV}) = (20) / (58.4 - 4.8) = 0.373$

$K_c = 100/PB$

ตารางที่ 4.1 แสดงสูตรในการหาค่าพารามิเตอร์ K_c , P, PI, PID

Control	PB(Percent)	Rate (Minutes)	Rate (Minutes)
P	$100KT_d/T$	-	-
PI	$110KT_d/T$	$3.33T_d$	-
PID	$80KT_d/T$	$2.00T_d$	$0.50T_d$

ตารางที่ 4.2 แสดงค่าที่คำนวณหาค่าพารามิเตอร์ K_c , P, PI, PID

Control	PB(Percent)	Rate (Minutes)	Rate (Minutes)
P	12.911, $K_c = 7.745$	-	-
PI	14.205, $K_c = 7.040$	$T_i = 0.389$	-
PID	10.329, $K_c = 9.681$	$T_i = 0.234$	$T_d = 0.0585$

Dead Time Calculator

พื้นที่หน้าตัดของแท่งค้ำน้ำ มีค่าเท่ากับ $A_t = 3.14/4 * (9.3^2 - 2.3^2) = 63.742 \text{ cm}^2$

ความสูงของแท่งน้ำจากจุด Zero ถึงจุดสูงสุดที่ปรับคือ Span มีค่าเท่ากับ $H = 17.8 \text{ cm}$

ปริมาตรของแท่งค้ำที่คำนวณ มีค่าเท่ากับ $V = A_t * H = 63.742 * 17.8 = 1134.6 \text{ cm}^3$

อัตราไหลเฉลี่ยคำนวณจาก Flow Max มีค่าเท่ากับ $\text{Flow} = 1134.6 / (0.257 * 60) = 73.579 \text{ cm}^3/\text{Sec}$

ขนาดพื้นที่หน้าตัดของสายยาง มีค่าเท่ากับ $A_p = 3.14/4 * (0.6^2) = 0.2826$

ค่าเวลาหน่วง(Dead Time) เมื่อคำนวณ มีค่าเท่ากับ $T_o = L / (f/A_p) = 0.5 / (73.579 / 0.2826) = 0.2 \text{ Sec}$

สรุปค่าที่คำนวณได้ $A_t L = 50 \text{ cm};$

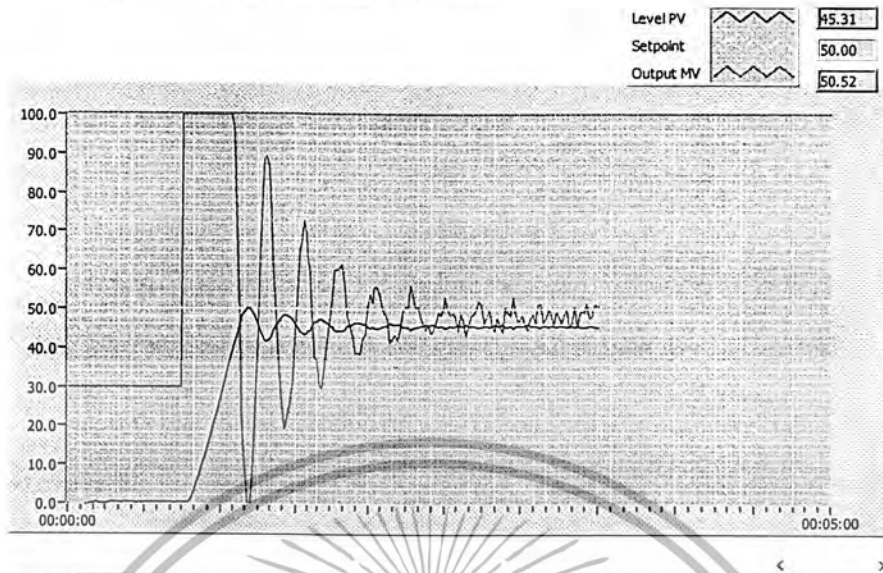
$T_o = L / (f/A_p) = 50 / (73.579 / 0.2826) = 0.2 \text{ Sec}$

สมการคุณลักษณะ คือ $G(s) = \frac{C(s)}{R(s)} = \frac{K e^{-t_0 s}}{\tau s + 1} = \frac{0.268 e^{-0.2s}}{0.338s + 1}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 จากกระบวนการได้ใส่ค่าพารามิเตอร์ P

$$Kc = 7.745, \%Overshoot = 69.1\%$$



รูปที่ 4.2 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ P เข้าไป

4.2.2 จากกระบวนการได้ใส่ค่าพารามิเตอร์ PI

$$Kc = 7.040, Ti = 0.389, \%Overshoot = 0\%$$

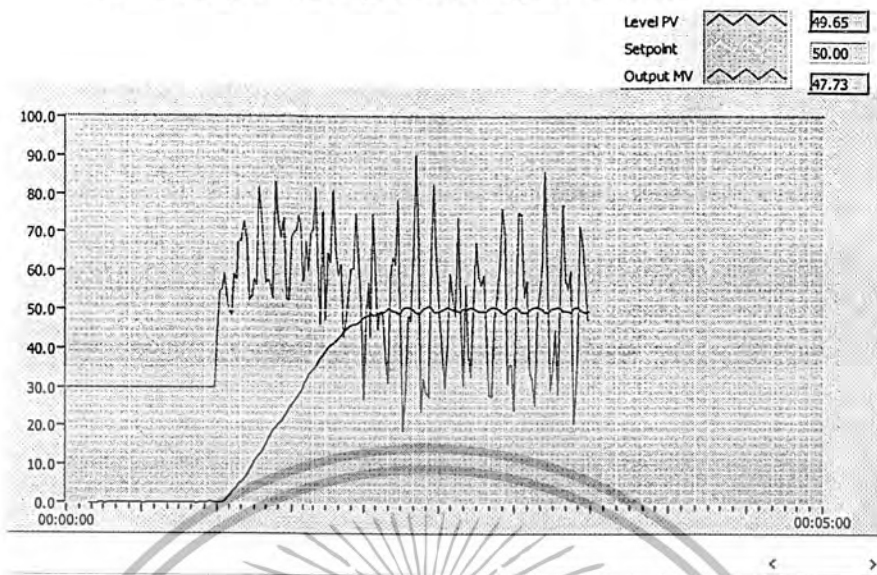


รูปที่ 4.3 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PI เข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

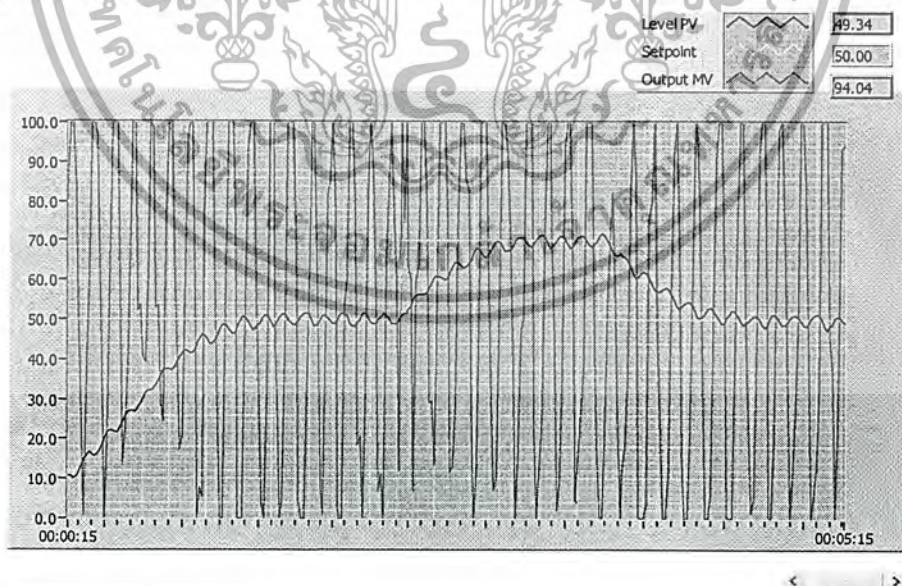
4.2.3 จากกระบวนการได้ใส่ค่าพารามิเตอร์ PID

$K_c = 9.681$, $T_i = 0.234$, $T_d = 0.0585$, %Overshoot = 0%



รูปที่ 4.4 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PID เข้าไป

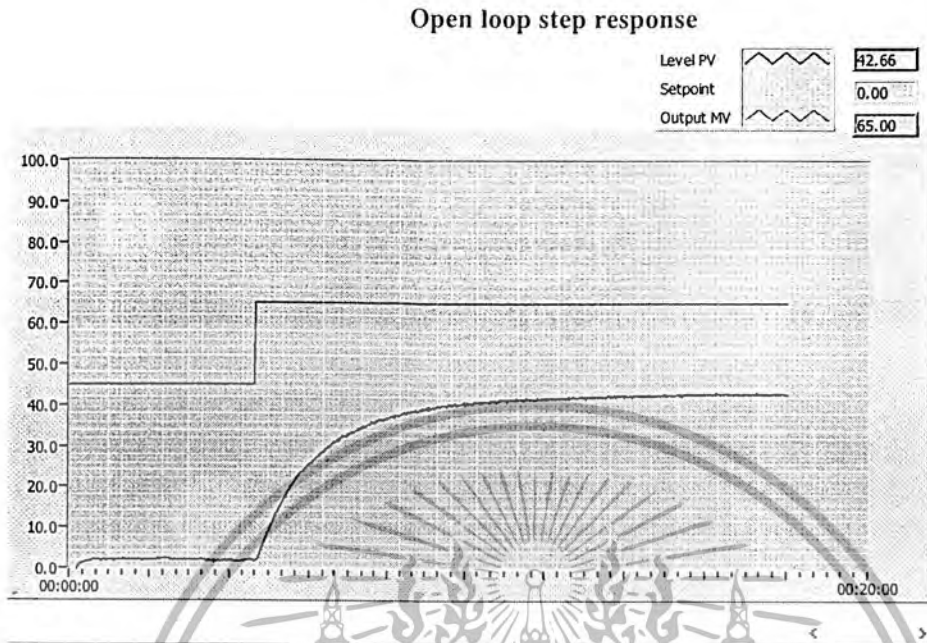
4.2.4 จากกระบวนการได้ใส่ค่า Disturbance เพื่อเป็นการรบกวนกระบวนการ ทำการรบกวนกระบวนการที่ทำการควบคุม โดยการเปลี่ยนค่าเป้าหมาย Set Point ขึ้นลง เป็น Step และสังเกตกราฟของการควบคุมกระบวนการที่ได้ ว่ายังคงติดตามค่าเป้าหมายอยู่หรือไม่
Dist Step sp 50 to 70 to 50



รูปที่ 4.5 แสดงกราฟของการควบคุมกระบวนการเมื่อมีการรบกวนกระบวนการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 เปลี่ยนความยาวสาย $L = 8\text{m}$



รูปที่ 4.6 แสดงวิธีการหาแบบจำลองกระบวนการแบบเปิด

Show Step 45% steady state change to 65% steady state Response to step input

$L = 8\text{ M}$, Dead Time = 15 at flow 45%

T_d = Deadtime in minutes

T = Time constant in minutes

K = Process gain = (Change in output) / (Change in PV)

Lower = 1.89 %

Hight = 42.66 %

$T_d = 5.5/60 = 0.092$

$T = 0.632 * (0.426 - 0.092) = 0.257$

$K = (\text{Change in output}) / (\text{Change in PV}) = (20) / (42.6 - 9.2) = 0.599$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่าที่ได้มาคำนวณหาค่าพารามิเตอร์ PID จากสูตรในตาราง

$$K_c = 100/PB$$

ตารางที่ 4.3 แสดงสูตรในการหาค่าพารามิเตอร์ KC, P, PI, PID

Control	PB(Percent)	Rate (Minutes)	Rate (Minutes)
P	100KTd/T	-	-
PI	110KTd/T	3.33Td	-
PID	80KTd/T	2.00Td	0.50Td

ตารางที่ 4.4 แสดงค่าที่คำนวณหาค่าพารามิเตอร์ KC, P, PI, PID

Control	PB(Percent)	Rate (Minutes)	Rate (Minutes)
P	11.653, Kc=8.851	-	-
PI	12.819, Kc=7.801	Ti=0.242	-
PID	9.323, Kc= 10.726	Ti=0.145	Td=0.036

Dead Time Calculator

พื้นที่หน้าตัดของแท็งก์น้ำ มีค่าเท่ากับ $A_t = 3.14/4 * (9.3^2 - 2.3^2) = 63.742 \text{ cm}^2$

ความสูงของแท็งก์น้ำจากจุด Zero ถึงจุดสูงสุดที่ปรับคือ Span มีค่าเท่ากับ $H = 17.8 \text{ cm}$

ปริมาตรของแท็งก์ที่คำนวณ มีค่าเท่ากับ $V = A_t * H = 63.742 * 17.8 = 1134.6 \text{ cm}^3$

อัตราไหลเฉลี่ยคำนวณจาก Flow Max มีค่าเท่ากับ $\text{Flow} = 1134.6 / (0.257 * 60) = 73.579 \text{ cm}^3/\text{Sec}$

ขนาดพื้นที่หน้าตัดของสายยาง มีค่าเท่ากับ $A_p = 3.14/4 * (0.6^2) = 0.2826$

ค่าเวลาหน่วง(Dead Time) เมื่อคำนวณ มีค่าเท่ากับ $T_o = L / (f/A_p) = 800 / (73.579 / 0.2826) = 3.07 \text{ Sec}$

สรุปค่าที่คำนวณได้ $A_t L = 800 \text{ cm};$

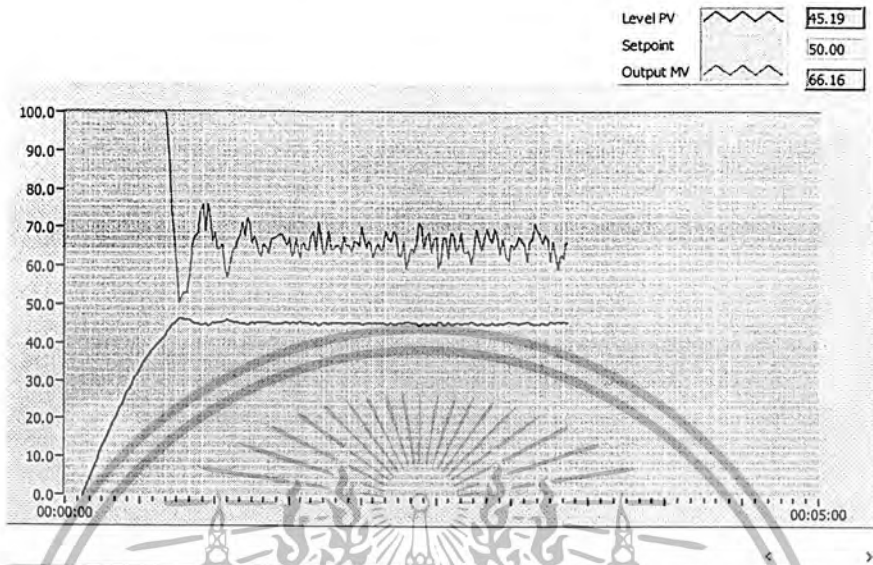
$$T_o = L / (f/A_p) = 800 / (73.579 / 0.2826) = 3.07 \text{ Sec}$$

สมการคุณลักษณะ คือ $G(s) = \frac{C(s)}{R(s)} = \frac{K e^{-t_0 s}}{\tau s + 1} = \frac{0.204 e^{-3.07 s}}{0.257 s + 1}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1 จากกระบวนการได้ใส่ค่าพารามิเตอร์ P

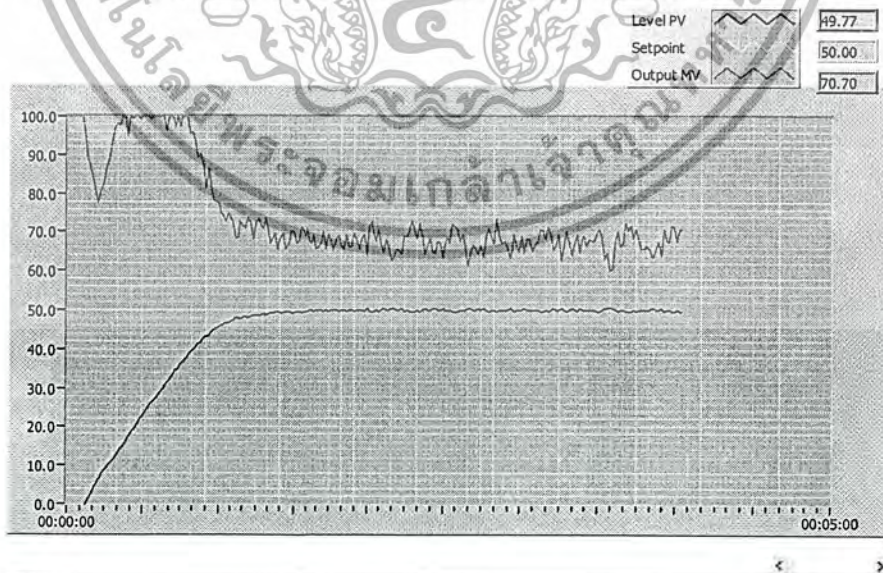
$K_c = 8.581$, Dead Time = 7 Sec , %Overshoot = 85.45%



รูปที่ 4.7 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ P เข้าไป

4.3.2 จากกระบวนการได้ใส่ค่าพารามิเตอร์ PI

$K_c = 7.801$ $T_i = 0.242$ Dead Time = 7 Sec , %Overshoot = 0%

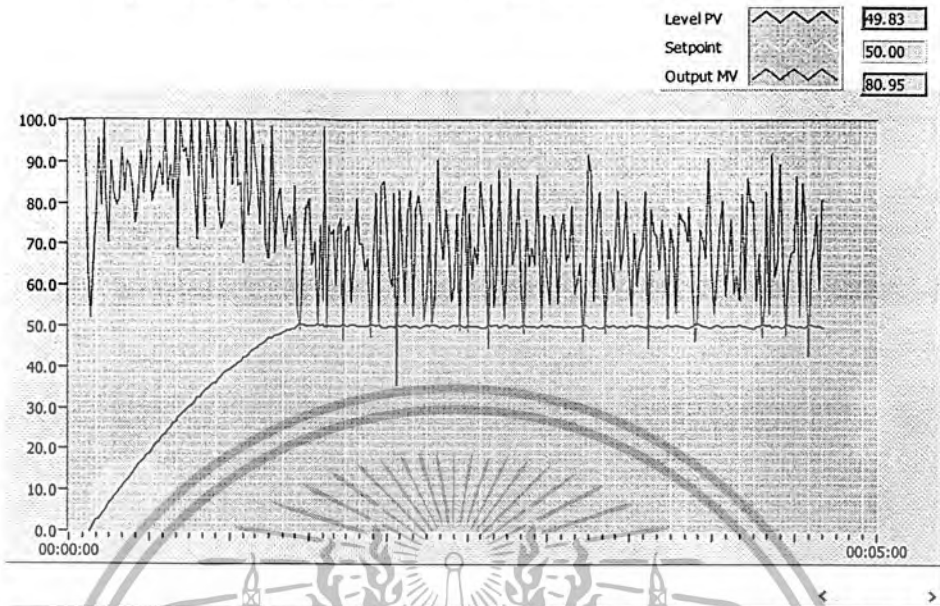


รูปที่ 4.8 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PI เข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 จากกระบวนการได้ใส่ค่าพารามิเตอร์ PID

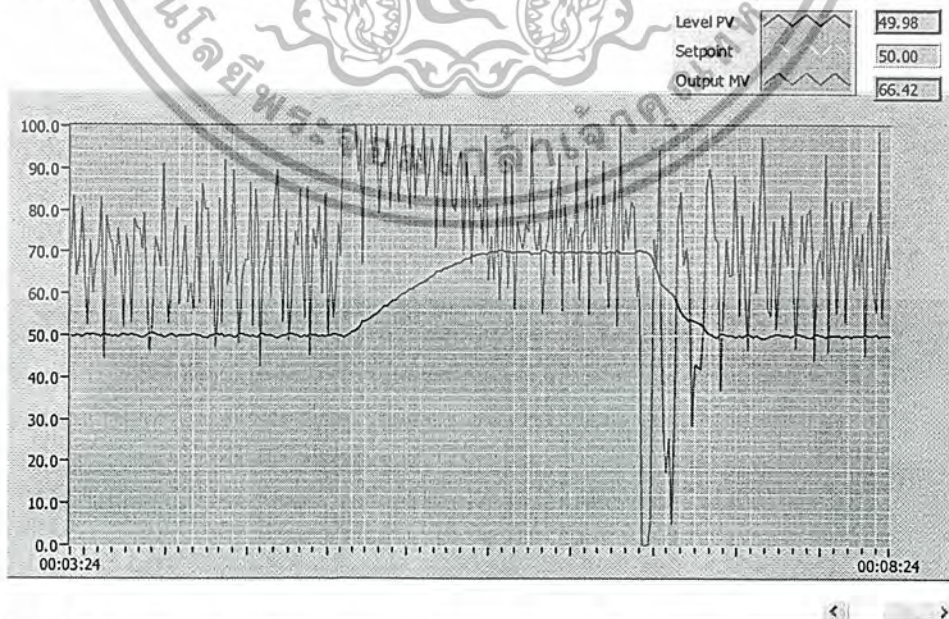
$K_c = 10.726$, $T_i = 0.145$, $T_d = 0.036$, Dead Time = 7 Sec , %Overshoot = 0%



รูปที่ 4.9 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PID เข้าไป

4.3.4 จากกระบวนการได้ใส่ค่า Disturbance เพื่อเป็นการรบกวนกระบวนการ

ทำการรบกวนกระบวนการที่ทำการควบคุม โดยการเปลี่ยนค่าเป้าหมาย Set Point ขึ้นลง เป็น Step และสังเกตกราฟของการควบคุมกระบวนการที่ได้ว่ายังคงติดตามค่าเป้าหมายอยู่ หรือไม่ Step sp 50 to 70 to 50

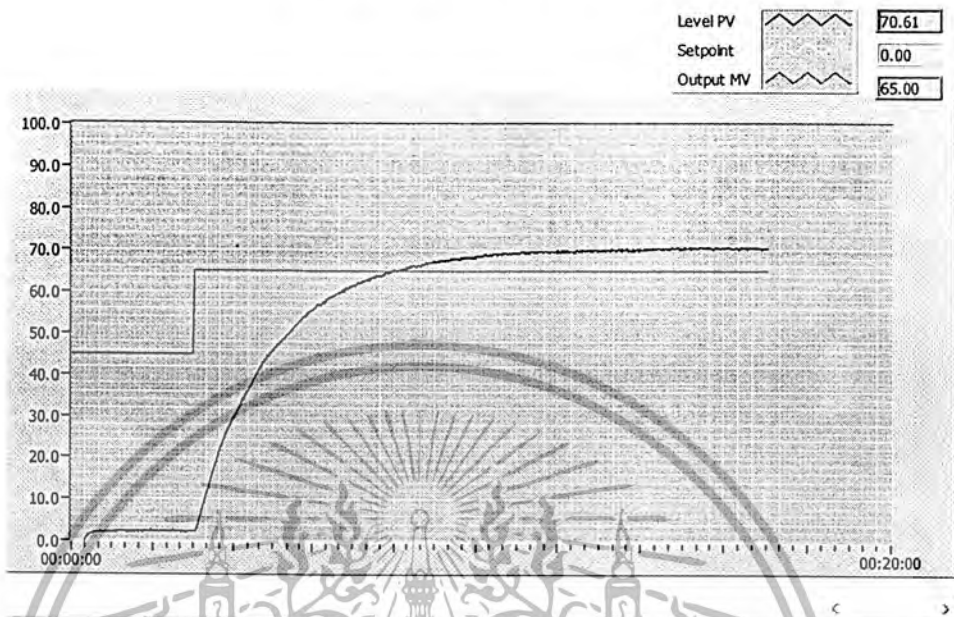


รูปที่ 4.10 แสดงกราฟของการควบคุมกระบวนการเมื่อมีการรบกวนกระบวนการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ความยาวสาย $L = 13\text{m}$

Open loop step response



รูปที่ 4.11 แสดงวิธีการหาแบบจำลองกระบวนการแบบเปิด

Show Step 45% steady state change to 65% steady state Response to step input

$L=13\text{ m}$, Dead Time = 20 at flow = 45 %

T_d = Deadtime in minutes

T = Time constant in minutes

K = Process gain = (Change in output) / (Change in PV)

Lower = 2.22 %

Hight = 70.61 %

$T_d = 6/60 = 0.1$

$T = 0.632 \cdot (0.706 - 0.022) = 0.432$

$K = (\text{Change in output}) / (\text{Change in PV}) = (20) / (70.2 - 2.2) = 0.293$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่าที่ได้มาคำนวณหาค่าพารามิเตอร์ PID จากสูตรในตาราง $K_c = 100/PB$
 ตารางที่ 4.5 แสดงสูตรในการหาค่าพารามิเตอร์ KC, P, PI, PID

Control	PB(Percent)	Rate (Minutes)	Rate (Minutes)
P	100KTd/T	-	-
PI	110KTd/T	3.33Td	-
PID	80KTd/T	2.00Td	0.50Td

ตารางที่ 4.6 แสดงค่าที่คำนวณหาค่าพารามิเตอร์ KC, P, PI, PID

Control	PB(Percent)	Rate (Minutes)	Rate (Minutes)
P	6.782, $K_c = 14.745$	-	-
PI	7.461, $K_c = 13.403$	$T_i = 0.333$	-
PID	5.426, $K_c = 18.430$	$T_i = 0.200$	$T_d = 0.050$

Dead Time Calculator

พื้นที่หน้าตัดของแท่งก้นน้ำ มีค่าเท่ากับ $A_t = 3.14/4 * (9.3^2 - 2.3^2) = 63.742 \text{ cm}^2$

ความสูงของแท่งน้ำจากจุด Zero ถึงจุดสูงสุดที่ปรับคือ Span มีค่าเท่ากับ $H = 17.8 \text{ cm}$

ปริมาตรของแท่งก้นน้ำมีค่าเท่ากับ $V = A_t * H = 63.742 * 17.8 = 1134.6 \text{ cm}^3$

อัตราไหลเฉลี่ยคำนวณจาก Flow Max มีค่าเท่ากับ $\text{Flow} = 1134.6 / (0.432 * 60) = 43.77 \text{ cm}^3/\text{Sec}$

ขนาดพื้นที่หน้าตัดของสายยาง มีค่าเท่ากับ $A_p = 3.14/4 * (0.6^2) = 0.2826$

ค่าเวลาหน่วง(Dead Time) เมื่อคำนวณ มีค่าเท่ากับ $T_o = L / fA_p = 1300 / (43.77 / 0.2826) = 8.4 \text{ Sec}$

สรุปค่าที่คำนวณได้ $A_t L = 1300 \text{ cm};$

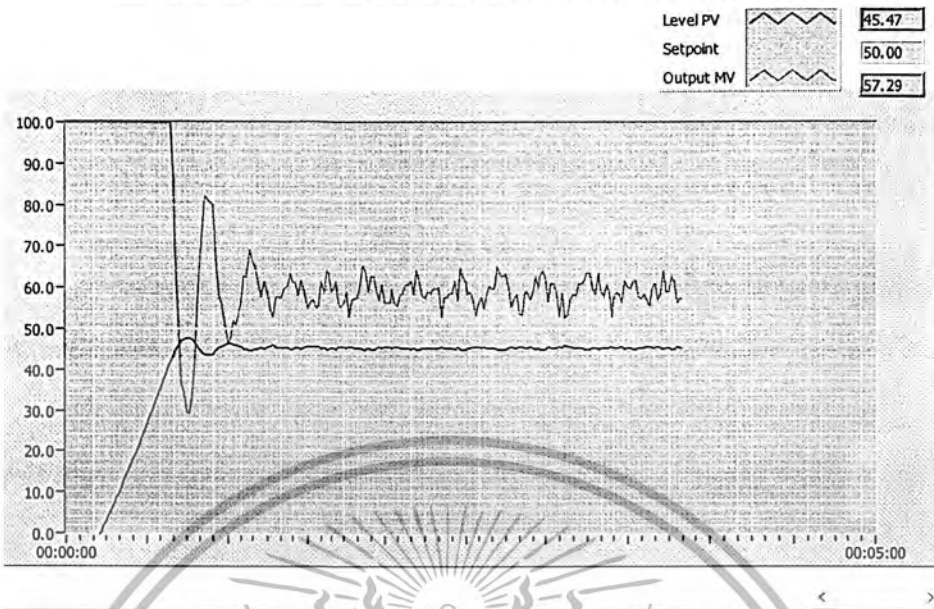
$$T_o = L / (f/A_p) = 1300 / (43.77 / 0.2826) = 8.4 \text{ Sec}$$

สมการคุณลักษณะ คือ $G(s) = \frac{C(s)}{R(s)} = \frac{K e^{-t_0 s}}{\tau s + 1} = \frac{0.342 e^{-8.40 s}}{0.432 s + 1}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.1 จากกระบวนการได้ใส่ค่าพารามิเตอร์ P

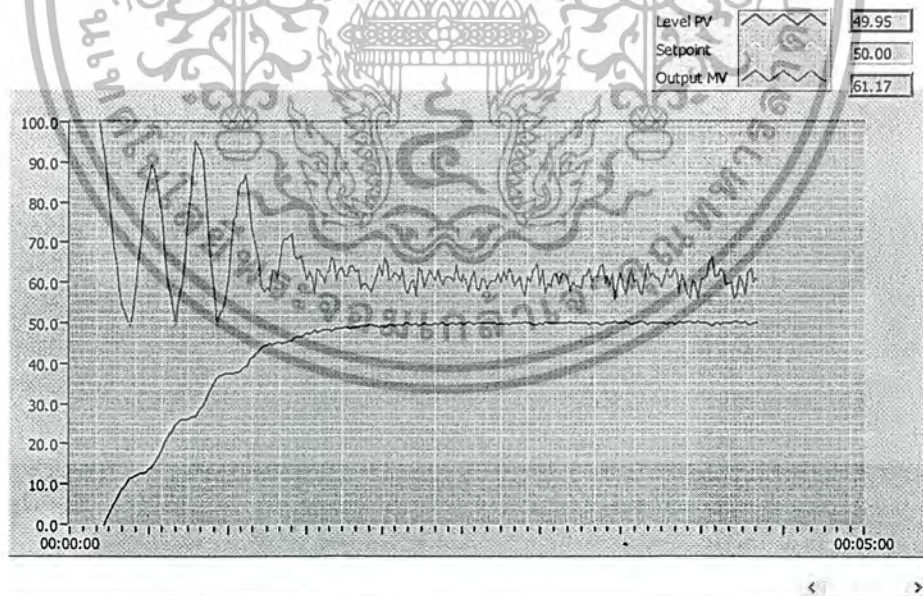
$K_c = 12.632$, Dead Time 12 Sec, Flow max 100%, %Overshoot = 0.729%



รูปที่ 4.12 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ P เข้าไป

4.4.2 จากกระบวนการได้ใส่ค่าพารามิเตอร์ PI

$K_c = 11.434$, $T_i = 0.333$, Dead Time 13, Sec, %Overshoot = 0%

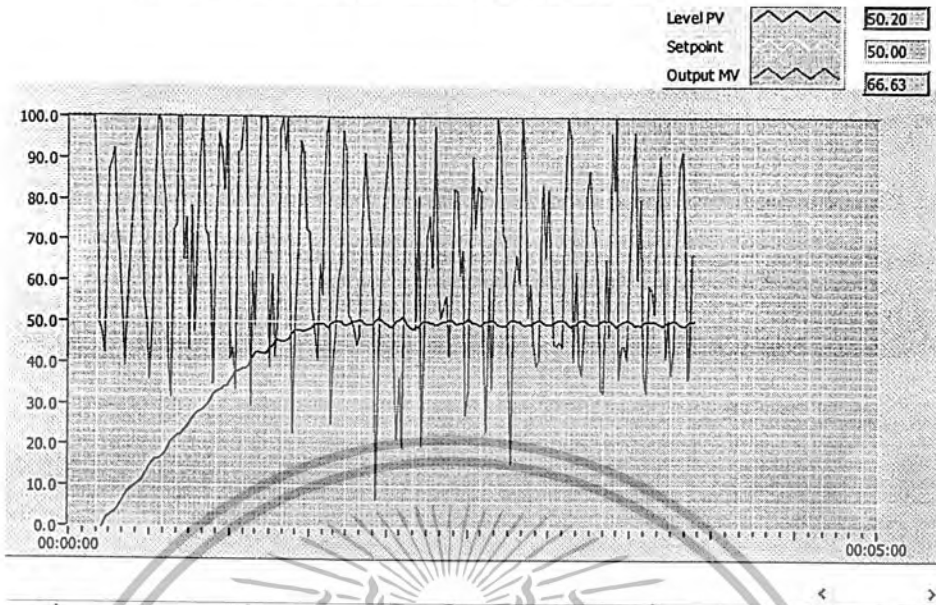


รูปที่ 4.13 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PI เข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

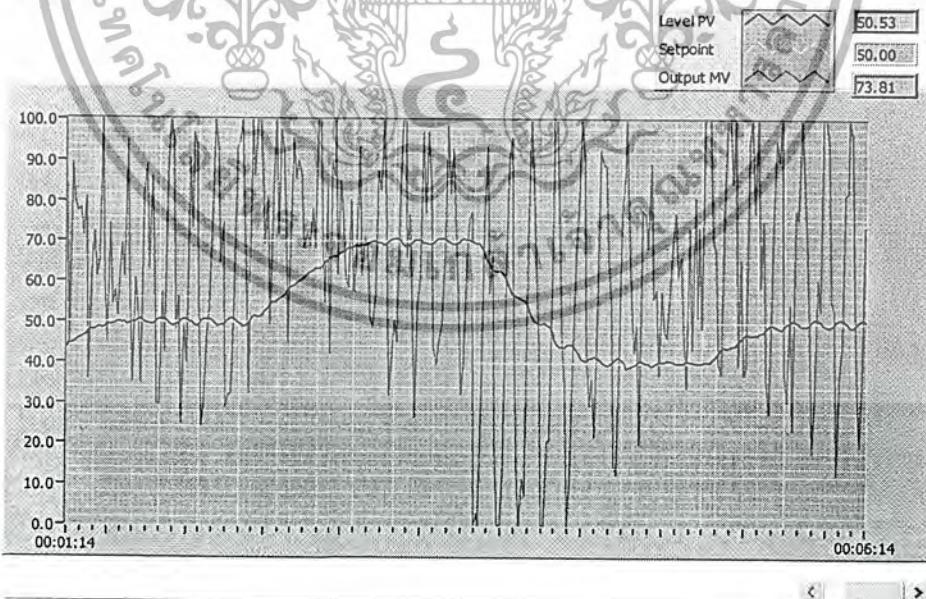
4.4.3 จากกระบวนการได้ใส่ค่าพารามิเตอร์ PID

$K_c=15.787$, $T_i=0.2$, $T_d = 0.05$, Dead Time 12 Sec, %Overshoot = 0%



รูปที่ 4.14 แสดงกราฟของการควบคุมกระบวนการเมื่อใส่ค่าพารามิเตอร์ PID เข้าไป

4.4.4 จากกระบวนการได้ใส่ค่า Disturbance เพื่อเป็นการรบกวนกระบวนการ ทำการรบกวนกระบวนการที่ทำการควบคุม โดยการเปลี่ยนค่าเป้าหมาย Set Point ขึ้นลง เป็น Step และสังเกตกราฟของการควบคุมกระบวนการที่ได้ว่ายังคงติดตามค่าเป้าหมายอยู่หรือไม่ โดย Step sp 50 to Step sp 70 to Step sp 40



รูปที่ 4.15 แสดงกราฟของการควบคุมกระบวนการเมื่อมีการรบกวนกระบวนการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 บทสรุป

ปริญญานิพนธ์นี้ได้นำเสนอวิธีการออกแบบและพัฒนาชุดทดลองของกระบวนการควบคุมระดับน้ำอัตโนมัติแบบมีโมดูลไร้ผลตอบสนอง โดยใช้วิธีการ Process Reaction Curve ของ Ziegler Nichole ในการปรับแต่งค่าพารามิเตอร์ของตัวควบคุม PID ในช่วงต่าง ๆ ตามที่ได้ออกแบบ แล้วนำค่าที่คำนวณได้นั้นมาทำการควบคุมกระบวนการ โดยในเริ่มต้นเราต้องทราบถึงฟังก์ชันการถ่ายโอน (Transfer Function) ของระบบโดยการทดสอบกระบวนการแบบเปิด (Open Loop Control) ซึ่งในการทดลองจะทำการควบคุมกระบวนการด้วยคอมพิวเตอร์ เพื่อความถูกต้องในการทำงานของตัวควบคุมที่ได้จากการออกแบบ และการควบคุมกระบวนการจริงที่ได้สร้างขึ้น

จากผลการทดลองการควบคุมกระบวนการควบคุมระดับน้ำ ที่จำลองเราด้วยชุดทดลองของกระบวนการควบคุมระดับน้ำอัตโนมัติแบบมีโมดูลไร้ผลตอบสนองด้วยโปรแกรมคอมพิวเตอร์และการควบคุมกระบวนการจริง ที่ผู้ทดลองได้ทำการสร้างขึ้นมา จะเห็นได้ว่าผลการทดลอง เป็นที่น่าพอใจแม้ว่าผลที่ได้จากการควบคุมกระบวนการจริงนั้นจะเกิดค่าความผิดพลาดต่าง ๆ ขึ้นมาจากการทดลอง ไม่ว่าจะเป็นค่าของสัญญาณเอาต์พุตที่ได้จากการทดลองแตกต่างจากทฤษฎีรอบค่าเป้าหมาย แต่ค่าความผิดพลาดนั้นยังอยู่ในช่วงของการยอมรับได้ ดังนั้น ตัวควบคุมที่ได้จากการออกแบบนั้นสามารถนำไปใช้ควบคุมกระบวนการได้จริง

5.2 ข้อเสนอแนะและแนวทางการพัฒนา

การพัฒนาชุดทดลองการควบคุมระดับน้ำอัตโนมัติแบบมีโมดูลไร้ผลตอบสนอง เมื่อเราได้ค่าพารามิเตอร์ ของตัวควบคุมกระบวนการ P, PI และ PID จะเห็นได้ว่าเมื่อนำไปควบคุมกระบวนการจริงแล้วค่าของสัญญาณเอาต์พุตที่ออกมา นั้นยังคงที่ไม่เข้าถึงค่าเป้าหมาย ดังนั้นในการควบคุมกระบวนการจริง ยังคงต้องใช้วิธีการ ลองผิด - ลองถูก (Trial and Error) เพื่อทำการปรับค่าพารามิเตอร์ให้ได้ค่าสัญญาณเอาต์พุตเข้าใกล้ค่าเป้าหมายมากที่สุด

เนื่องจากการคำนวณค่าของค่าเวลาดังตัวค่าเวลา Dead Time หรือไม่ว่าจะเป็นค่า K (Process Gain) ของกราฟที่ทดสอบกระบวนการแบบเปิดนั้นเกิดความคลาดเคลื่อนขึ้นจากการทดลอง เนื่องจากค่าที่นำมาคำนวณนั้นจะเป็นค่าประมาณจากกราฟพลูเปิดที่ได้มาซึ่งการปรับแต่งค่าพารามิเตอร์ต่างๆเหล่านี้ต้องใช้เวลาและขึ้นอยู่กับระบบของกระบวนการที่จะทำการทดสอบด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวทางในการพัฒนาต่อไป คือการพัฒนาโครงสร้างพื้นฐานของระบบกระบวนการที่จะควบคุม การปรับแต่งค่าพารามิเตอร์ควบคุม PID สามารถที่จะทำการควบคุมได้โดยการศึกษาความสัมพันธ์ระหว่างค่าพารามิเตอร์แต่ละตัวของตัวควบคุม ที่ต้องการปรับแต่งให้กับค่าสมรรถนะของระบบควบคุมเพื่อนำไปเป็นค่าดัชนีที่ใช้ในการปรับแต่งนอกจากนี้ในการทดลองยังมีได้ทดลองกระบวนการควบคุมแบบนิวโรฟัซซีและความคงทนต่อสัญญาณภายนอก ซึ่งเป็นอีกแนวทางหนึ่งที่น่าสนใจในการออกแบบสำหรับกระบวนการต่อไป

คุณภาพของตัวตรวจวัดสัญญาณ จะมีผลต่อระบบควบคุม เพราะในการทดลองผู้ทดลองได้ทำการออกแบบในส่วนของตัวตรวจวัดสัญญาณแล้วทำการทดลองจึงพบว่า ค่าของกราฟการควบคุมนั้นมีค่าการแกว่ง และความคลาดเคลื่อนขึ้นมาก ซึ่งตรงจุดนี้จะเป็นข้อเสียที่จะทำให้การคำนวณค่าพารามิเตอร์ต่าง ๆ เพื่อให้ได้ค่าที่ถูกต้อง ในการนำมาควบคุมกระบวนการ อันเนื่องมาจากค่ากราฟที่ได้นั้นเปลี่ยนแปลงไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

John Dorsey, **Continuous and Discrete Control System**, Modeling, Identification, Design and Implementation, Georgia Institute of Technology Atlanta, Georgia.

Katsuhiko Ogata, **Modern Control Engineering**, Third Edition, University Of Minnesota, Prentice Hall Internationnal, Inc. 1997

Carlos A. Smith, Ph.D.,P.E, University of South Florida, Armando B. Corripio, Ph.D., P.E., Louisiana State University, **Principles and Practice of Automatic Process Control**, Second Edition, John Wiley & Sons, Inc.

รศ.กิตติ ตีระเศรษฐม, รศ.วิทยา ทิพย์สุวรรณพร, **วิศวกรรมระบบควบคุม Control System Engineering และ ระบบควบคุมป้อนกลับ Feedback Control System**, คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมการวัดคุม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

นาย เทียนชัย สุขศรี, **การควบคุมระดับน้ำด้วยตัวควบคุมแบบนิวโรฟัซซี่ Neuro Fuzzy For Level Control Application**, ปรินญาณิพนธ์ ปรินญาวิศวกรรมศาสตรบัณฑิต สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2544

เจริญ เพชรมูลี, **เรียนลัด LabVIEW**, กรุงเทพฯ: ซีเอ็ดยูเคชั่น, 2547

