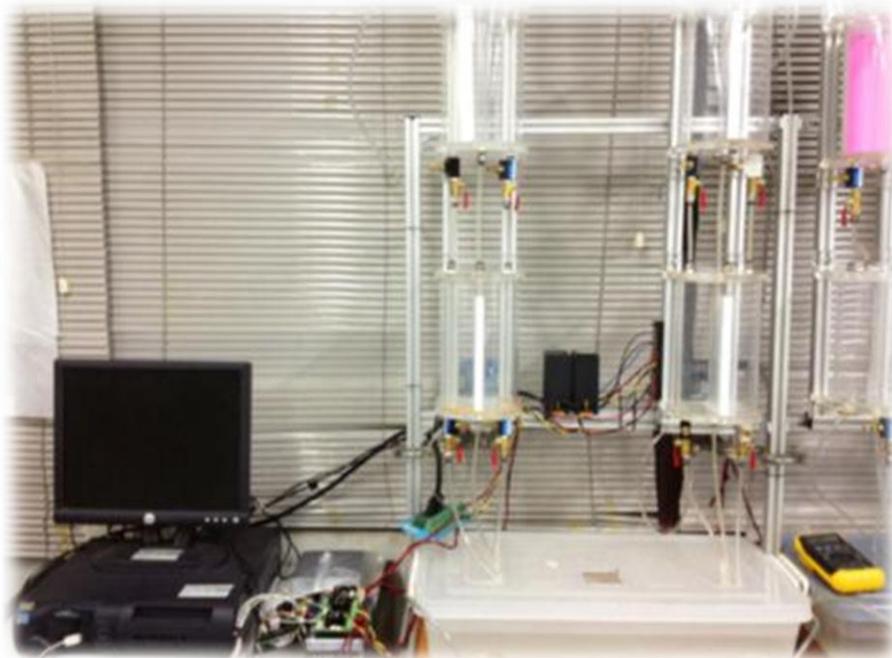




รายงานการวิจัยฉบับสมบูรณ์

การออกแบบชุดทดลองกระบวนการระดับน้ำแบบ 4 ถัง Design and Development Quadruple-tanks Process Apparatus



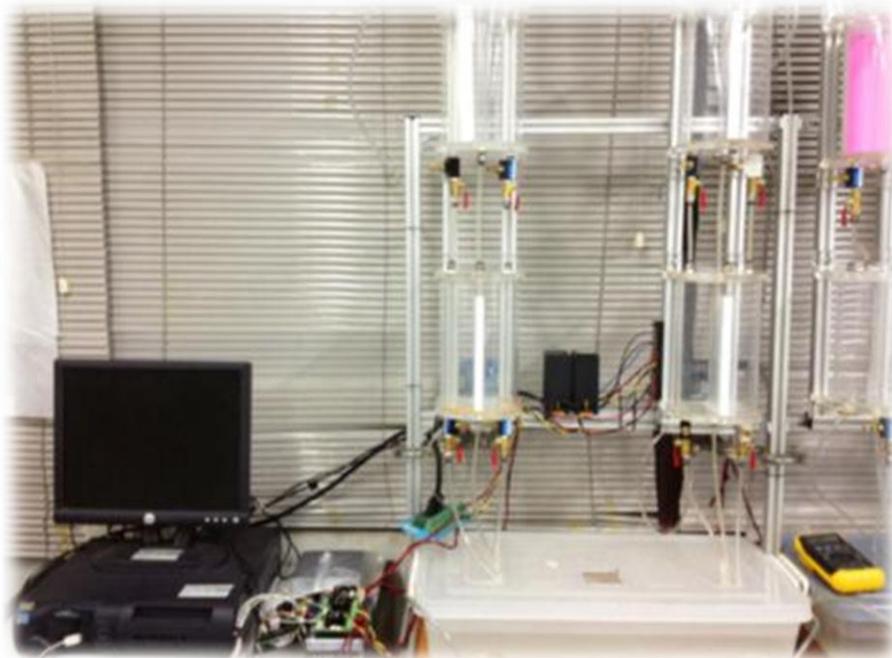
รองศาสตราจารย์อาจินต์ น่วมสำราญ

งานวิจัยนี้ได้รับทุนสนับสนุนงานวิจัยจากงบประมาณเงินรายได้ ประจำปีงบประมาณ 2555
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



รายงานการวิจัยฉบับสมบูรณ์

การออกแบบชุดทดลองกระบวนการระดับน้ำแบบ 4 ถัง Design and Development Quadruple-tanks Process Apparatus



รองศาสตราจารย์อาจินต์ น่วมสำราญ

งานวิจัยนี้ได้รับทุนสนับสนุนงานวิจัยจากงบประมาณเงินรายได้ ประจำปีงบประมาณ 2555
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อโครงการ (ภาษาไทย) การออกแบบชุดทดลองกระบวนการระดับน้ำแบบ 4 ถัง.....
 แหล่งเงินทุน งบประมาณเงินรายได้ คณะวิศวกรรมศาสตร์.....
 ประจำปีงบประมาณ..... 2555..... จำนวนเงินที่ได้รับการสนับสนุน..... 81,000..... บาท
 ระยะเวลาทำการวิจัย..... 1..... ปี ตั้งแต่ ตุลาคม 2554..... ถึง..... กันยายน 2555.....
 ชื่อ-สกุล หัวหน้าโครงการ และผู้ร่วมโครงการวิจัย พร้อมระบุ หน่วยงานต้นสังกัด
 รองศาสตราจารย์อาจินต์ นุ่มสำราญ..... สาขาวิชาวิศวกรรมการวัดและควบคุม..... คณะวิศวกรรมศาสตร์.....

บทคัดย่อ

ชุดทดลองกระบวนการระดับน้ำ 4 ถัง เป็นอุปกรณ์ที่มีความเหมาะสมสำหรับการประยุกต์ใช้เป็นกระบวนการตัวอย่างในการออกแบบระบบควบคุม สำหรับกระบวนการแบบสองอินพุตสองเอาต์พุตที่สามารถพบเห็นได้ในหลายๆอุตสาหกรรม ซึ่งโดยทั่วไปแล้วกระบวนการแบบสองอินพุตสองเอาต์พุตมักพบปัญหาในการควบคุมระบบเนื่องจากการรบกวนกันระหว่างค่าตัวแปรอินพุตและเอาต์พุตทั้งสอง การออกแบบระบบควบคุมแบบสองอินพุตสองเอาต์พุตที่มีการรบกวนกันเพื่อให้ได้ผลตอบสนองของระบบควบคุมตามต้องการ จึงต้องอาศัยความรู้ความเข้าใจด้านคุณลักษณะของกระบวนการ ทฤษฎีการออกแบบระบบควบคุมด้วยเทคนิคต่างๆ รวมถึงทักษะความรู้ทางด้านการใช้งานอุปกรณ์ตรวจวัด ตัวแปลงสัญญาณ และตัวควบคุม ดังนั้นชุดทดลองกระบวนการระดับน้ำ 4 ถังจึงเป็นโจทย์ปัญหามาตรฐานสำหรับการวิเคราะห์และออกแบบระบบควบคุม และสามารถใช้ในการเรียนการสอนเพื่อเสริมสร้างทักษะความรู้ความเข้าใจถึงขีดจำกัดสมรรถนะของระบบควบคุมแบบสองอินพุตสองเอาต์พุต อันเนื่องมาจากการรบกวนกันของตัวแปรในกระบวนการ ความไม่แน่นอนของโมเดลกระบวนการ พฤติกรรมของระบบแบบเฟสไม่ต่ำสุด การออกแบบตัวควบคุมแบบกระจายส่วน การประยุกต์ใช้ตัวควบคุมแบบดีคัปปลิงเพื่อลดผลกระทบจากการรบกวนกันของตัวแปรในกระบวนการ

คำสำคัญ : กระบวนการระดับน้ำ 4 ถัง, สองอินพุตสองเอาต์พุต, การวิเคราะห์และออกแบบระบบควบคุม

Research Title: Design and Development Quadruple-tank Process Apparatus.....

Researcher Assoc. Prof. Arjin Numsomran.....

Faculty:Engineering..... DepartmentInstrumentation and Control Engineering.....

ABSTRACT

A quadruple-tanks process apparatus is an appropriate device for the control system design application of two inputs - two outputs (TITO) process which can be found in several industries. Generally, the TITO control system often has a problem because of the interference between the input and output variables. Therefore achieving the response of control system as desired specifications, it requires a deep understanding of process characteristics, control system theory including knowledge and skills for apparatuses using. Thus, the quadruple-tanks process can be a standard problem for control system analysis and design practices and it can be used to teach students about the skills of instrument devices using, understanding control limitation due to interactions, model uncertainties, non-minimum phase behavior, Design decentralized controllers, implementing decoupling controller to reduce the effect of interactions.

Keyword : quadruple-tanks process, TITO, control system analysis and design

กิตติกรรมประกาศ

การวิจัยครั้งนี้ได้รับทุนสนับสนุนการวิจัยจากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง จากแหล่งทุนงบประมาณเงินรายได้ ประจำปีงบประมาณ พ.ศ. 2555

ขอกราบขอบพระคุณท่านอธิการบดี สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่สนับสนุนการดำเนินการวิจัยและอำนวยความสะดวก

ขอขอบคุณ สาขาวิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่สนับสนุนการดำเนินการวิจัยจนสำเร็จลุล่วง

ขอขอบคุณนักศึกษา STW Lab สาขาวิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้ความร่วมมือในการดำเนินการวิจัย

รองศาสตราจารย์อาจินต์ น่วมสำราญ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญภาพ.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์งานวิจัย.....	1
1.3 ขอบเขตของงานวิจัย.....	1
1.4 วิธีดำเนินงานวิจัย.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	3
2.1 การทบทวนวรรณกรรม (Literature review).....	3
2.2 วรรณกรรมที่ถูกอ้างอิง.....	3
บทที่ 3 องค์ประกอบของกระบวนการระดับน้ำ 4 ถัง.....	5
3.1 กระบวนการระดับน้ำ 4 ถัง.....	6
3.2 อุปกรณ์วัดระดับน้ำ.....	7
3.3 อุปกรณ์ปรับแต่งสัญญาณ 4-20mA เป็น 1-5V.....	8
3.4 ชุดขับปั๊มน้ำ.....	9
3.5 ปั๊มน้ำ.....	10
3.6 แหล่งจ่ายไฟสำหรับอุปกรณ์.....	10
3.7 ชุดบอร์ดควบคุมกระบวนการ.....	12
บทที่ 4 แบบจำลองทางคณิตศาสตร์และการวิเคราะห์คุณลักษณะของกระบวนการน้ำ 4 ถัง.....	19
4.1 แบบจำลองทางคณิตศาสตร์ของกระบวนการน้ำ 4 ถัง.....	19
4.2 การวิเคราะห์คุณลักษณะของกระบวนการระดับน้ำ 4 ถัง.....	21
4.2.1 ซีโรของกระบวนการ MIMO.....	21
4.2.2 การทำงานของซีโร.....	22
4.2.3 ผลกระทบจากตำแหน่งซีโรต่อระบบควบคุม.....	23
บทที่ 5 ความรู้พื้นฐานการประยุกต์ใช้งานด้านซอฟต์แวร์.....	24
5.1 การใช้งานโปรแกรม Matlab.....	24
5.1.1 โปรแกรม Simulink.....	24
5.1.1.1 การจำลองระบบพลศาสตร์ (Dynamic System Simulation).....	24
ด้วย Simulink	

สารบัญ (ต่อ)

	หน้า
5.1.1.2	หลักการของสัญญาณและการไหลของตรรก.....24 (Signal and Logic Flow)
5.1.1.3	การเชื่อมต่อบล็อก.....25
5.1.1.4	กลุ่มของบล็อกที่ใช้งานบ่อยในปัจจุบัน.....26
5.1.1.5	การสร้างแบบจำลองระบบควบคุม.....29
5.1.2	การติดตั้งโปรแกรมเพื่อใช้งาน Real Time Windows Target.....34
5.1.2.1	Simulink external model interface.....34
5.1.2.2	การติดตั้ง Real Time Windows Target Kernal.....34
5.1.2.3	การค่าพารามิเตอร์ใน Simulink สำหรับ Real Time Workshop.....35
5.2	การเขียนโปรแกรมภาษาซีสำหรับหน่วยประมวลผล ARM7.....36
5.2.1	สถาปัตยกรรมไมโครคอนโทรลเลอร์ ARM7 (ADuC7024).....36
5.2.2	โปรแกรม Keil uVision 3.11.....36
5.2.2.1	การเริ่มต้นใช้งานโปรแกรม Keil uVision3.....41
5.2.2.2	การ debug โปรแกรมภาษา C สำหรับ ARM7.....45
5.2.2.3	การติดตั้งซอฟต์แวร์ ARMWSD.....47
5.3	พื้นฐานการเขียนภาษาซี (Basic C For ARM7).....48
5.3.1	ความหมายภาษา C สำหรับ ARM7.....48
5.3.2	โครงสร้างโปรแกรมภาษา C สำหรับ ARM7.....49
5.3.3	ตัวแปรชนิดต่าง ๆ และค่าคงที่.....50
5.3.4	ชนิดของตัวแปรในภาษาซี.....50
5.3.5	การประกาศตัวแปร.....51
5.3.6	ตัวแปรแบบอาร์เรย์.....52
5.3.7	เครื่องหมายตัวดำเนินการ.....52
5.3.8	รูปแบบการเขียนคำสั่ง.....54
5.3.8.1	คำสั่งเงื่อนไขและการทำซ้ำ.....56
5.3.8.2	Standard Function และ User define Function.....59
บทที่ 6	การออกแบบระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ 4 ถัง.....60
6.1	การควบคุมแบบพีไอดี.....60
6.1.1	ผลกระทบการควบคุมแบบ I และ D.....62
6.2	การออกแบบตัวควบคุมพีไอดีด้วยเทคนิคทางเดินราก.....64
6.2.1	สถาปัตยกรรมไมโครคอนโทรลเลอร์ ARM7 (ADuC7024).....64
6.2.2	การสร้าง ROOT LOCI ที่สมบูรณ์.....68
6.3	การออกแบบตัวควบคุมดิคัมปลิ่งด้วย สำหรับกระบวนการหลาย.....74
	แบบหลายอินพุตหลายเอาต์พุตพีไอดีด้วยเทคนิคทางเดินราก

สารบัญ (ต่อ)

	หน้า
เอกสารอ้างอิง.....	77
ภาคผนวก (ใบงานการทดลอง).....	78
ประวัติผู้วิจัย.....	98

สารบัญตาราง

ตารางที่	หน้า
3.1 รายละเอียดทางเทคนิคของอุปกรณ์วัดระดับน้ำ (Differential Pressure Transmitter).....	7
3.2 รายละเอียดทางเทคนิคของอุปกรณ์ปรับแต่งสัญญาณ SC-95.....	9
3.3 รายละเอียดทางเทคนิคของชุดขับปั้มน้ำ.....	9
3.4 รายละเอียดทางเทคนิคของปั้มน้ำ.....	10
3.5 ตารางแสดงคุณสมบัติของ Switching Power Supply.....	11
5.1 ชนิดของตัวแปรในภาษาซี.....	50
5.2 เครื่องหมายตัวดำเนินการทางคณิตศาสตร์.....	53
5.3 เครื่องหมายตัวดำเนินการทางด้านการเปรียบเทียบ.....	53
5.4 เครื่องหมายตัวดำเนินการระดับบิต.....	53
5.5 เครื่องหมายตัวดำเนินการเพิ่มค่าและลดค่า.....	53
5.6 เครื่องหมายตัวดำเนินการเลื่อนบิต (ซ้าย, ขวา).....	54
5.7 ตัวกำหนดชนิดข้อมูลของภาษาซี.....	54

สารบัญภาพ

ภาพที่	หน้า
3.1 ชุดทดลองกระบวนการระดับน้ำ 4 ถัง.....	5
3.2 โครงสร้างกระบวนการระดับน้ำ 4 ถัง.....	6
3.3 วาล์วที่ใช้ในการปรับพื้นที่หน้าตัดรูน้ำด้านนอก.....	6
3.4 อุปกรณ์วัดระดับน้ำ (Pressure Sensor 3.92kPa Gauge 1-Port).....	7
3.5 อุปกรณ์ปรับแต่งสัญญาณ 4-20mA เป็น 1-5V	8
3.6 ชุดขับปั๊มน้ำ SE-HB40-1.....	9
3.7 ปั๊มน้ำ API Instrument.....	10
3.8 บล็อกไดอะแกรมของแหล่งจ่ายไฟขนาด ± 24 V, 6.5 A.....	11
3.9 อุปกรณ์แหล่งไฟขนาด ± 24 V, 6.5 A.....	11
3.10 ไมโครคอนโทรลเลอร์ ARM7.....	12
3.11 สถาปัตยกรรมภายในของไมโครคอนโทรลเลอร์ ARM7 เบอร์ ADuC7024.....	13
3.12 บอร์ด ET-BASE ARM7024 (ADuC7024).....	14
3.13 พอร์ต P1 มีขนาด 6 บิต.....	16
3.14 พอร์ต P3 มีขนาด 8 บิต.....	16
3.15 พอร์ต P4 มีขนาด 8 บิต.....	16
3.16 พอร์ต ARM-JTAG และวงจรส่วนที่ใช้ในการเชื่อมต่อกับ ARM-JTAG.....	17
3.17 การจัดเรียงขาสัญญาณของ Character LCD มาตรฐาน.....	17
3.18 วงจรส่วนที่เชื่อมต่อกับ RS232.....	18
3.19 ขั้วต่อ ADC0 – ADC9.....	18
3.20 ขั้วต่อ DAC0 และ DAC1.....	18
4.1 ชุดทดลองกระบวนการระดับน้ำ 4 ถัง.....	19
5.1 ลักษณะของโปรแกรม Simulink.....	25
5.2 แสดงบล็อกต่างๆ ของแหล่งกำเนิดสัญญาณ (Sources) และแหล่งสัญญาณ (Sinks).....	26
5.3 แสดงบล็อกต่างๆ ภายในกลุ่มบล็อกของระบบต่อเนื่อง (Continuous System)..... และระบบไม่ต่อเนื่อง (Discrete System)	27
5.4 แสดงบล็อกต่างๆ ภายในกลุ่มบล็อกการดำเนินการทางคณิตศาสตร์ (Math Operations).....	27
5.5 แสดงกลุ่มบล็อก Discontinuities และกลุ่ม Signal Routing.....	29
5.6 แสดงหน้าต่างที่ใช้ในการกำหนดค่าคงที่ในการจำลองปัญหา (Configuration Parameters).....	29
5.7 หน้าต่างสำหรับสร้างแบบจำลอง.....	30
5.8 การเลือกบล็อกที่ต้องการจาก Simulink Library Browser.....	31
5.9 บล็อกของระบบตัวอย่าง.....	31
5.10 การเชื่อมต่อสัญญาณระหว่างบล็อก.....	32
5.11 การเปลี่ยนแปลงค่าพารามิเตอร์ของบล็อก.....	32
5.12 ไดอะแกรมใส่ค่าพารามิเตอร์ของบล็อก Transfer Function.....	33
5.13 แบบจำลองระบบทางกลในรูปแบบของ Transfer Function.....	33

สารบัญญภาพ (ต่อ)

ภาพที่	หน้า
5.14 Hardware Implementation.....	35
5.15 RTW System Target File.....	36
5.16 แสดงกระบวนการทำงานของไปป์ไลน์ใน ARM7โปรเซสเซอร์.....	37
5.17 ลักษณะการใช้งาน ARM ทั้ง 7 โหมด.....	38
5.18 หน้าต่าง Setup Real View Microcontroller Development Kit Version 3.11.....	40
5.19 หน้าต่าง customer Information.....	40
5.20 หน้าต่างแสดงการสิ้นสุดการติดตั้งโปรแกรม.....	41
5.21 หน้าต่างหลักของโปรแกรม Keil uVision.....	41
5.22 หน้าต่าง Components, Environment and Books.....	42
5.23 หน้าต่าง Create New Project.....	42
5.24 หน้าต่าง Select Device for Target.....	43
5.25 หน้าต่าง Add File to Group.....	43
5.26 หน้าต่างกำหนด Options for Target	44
5.27 หน้าต่างกำหนด Output Information.....	44
5.28 การ compile และการ Build โปรแกรม.....	45
5.29 ปุ่ม Build Target.....	46
5.30 การเข้าสู่โหมด Simulation หรือ Debug.....	46
5.31 แสดงสถานะเตรียมพร้อม Simulator.....	46
5.32 ปุ่มการ Debug แบบต่างๆ.....	47
5.33 หน้าต่างการใช้งานของโปรแกรม ARMWSD.....	47
5.34 การ Configuration โปรแกรม ARMWSD ก่อนใช้งาน.....	48
5.35 Flow Chart การประมวลผลภาษาซีสำหรับ ARM7.....	49
5.36 โครงสร้างโปรแกรมภาษาซี.....	50
5.37 ตัวอย่างการประกาศตัวแบบ char และ int.....	51
5.38 อาร์เรย์ 1 มิติ.....	52
5.39 อาร์เรย์ 2 มิติ.....	52
6.1 (a) แสดง block diagram ของการควบคุมแบบ P, I, D.....	61
(b) แสดงว่า เป็นฟังก์ชัน unit ramp	
(c) แสดงการควบคุมเอาต์พุต	
6.2 (a) block diagram ของระบบควบคุมแบบอัตโนมัติแบบ first order sensor.....	61
(b) block diagram ของระบบควบคุมแบบอัตโนมัติแบบ over damped second order sensor	
(c) block diagram ของระบบควบคุมแบบอัตโนมัติแบบ underdamped second order sensor	
6.3 การควบคุมแบบ integral.....	62

สารบัญญภาพ (ต่อ)

ภาพที่	หน้า
6.4 (a) แสดงระบบการควบคุมระดับของเหลว.....	63
(b) แสดง block diagram ของระบบ	
6.5 pole, zero ของ $G(s) = \frac{[K(s + z_1)]}{[s(s + p_2)(s + p_3)]}$	67
6.6 ตัวอย่างจุด breakaway บนแกน real ในระนาบ s.....	74
6.7 โครงสร้างของกระบวนการแบบ TITO.....	74
6.8 โครงสร้างระบบควบคุมสำหรับกระบวนการระดับน้ำแบบ 4 ถึงที่ประกอบด้วยชุดดีคัปปลิ่ง.....	75

บทที่ 1

บทนำ

1.1 หลักการและเหตุผลของโครงการวิจัย

การศึกษาด้านวิศวกรรมศาสตร์นั้น การปฏิบัติการทดลองถือเป็นองค์ประกอบหนึ่งของกระบวนการศึกษาที่มีความสำคัญและมีความจำเป็นอย่างมากเพราะการปฏิบัติการทดลองช่วยให้นักศึกษาสามารถเข้าใจหลักการทางทฤษฎีได้มากขึ้น อีกทั้งยังเป็นอีกช่องทางหนึ่งในการเพิ่มทักษะในการปฏิบัติงาน การใช้อุปกรณ์และเครื่องมือต่างๆให้กับนักศึกษา สำหรับหลักสูตรวิศวกรรมการวัดคุมได้มุ่งเน้นให้นักศึกษาเข้าใจหลักการวัดและการออกแบบระบบควบคุมซึ่งชุดทดลองที่จำหน่ายโดยทั่วไปมักมีราคาสูงและยังขาดความยืดหยุ่นในการประยุกต์ใช้ตามวัตถุประสงค์ที่ต้องการ ดังนั้นคณะผู้วิจัยจึงมีแนวคิดในการออกแบบชุดทดลองกระบวนการระดับน้ำ 4 ถึงที่มีราคาต่ำแต่มีความยืดหยุ่นสำหรับการประยุกต์ใช้สูง สามารถสร้างความรู้ความเข้าใจพฤติกรรมของกระบวนการและฝึกฝนทักษะในการออกแบบระบบควบคุมให้แก่ นักศึกษา วิศวกรรมการวัดคุมได้เป็นอย่างดี

ชุดทดลองกระบวนการระดับน้ำ 4 ถึง เป็นอุปกรณ์ที่มีความเหมาะสมสำหรับการประยุกต์ใช้เป็นกระบวนการตัวอย่างในการออกแบบระบบควบคุมสำหรับกระบวนการแบบสองอินพุตสองเอาต์พุตที่สามารถพบเห็นได้ในหลายๆอุตสาหกรรม ซึ่งโดยทั่วไปแล้วกระบวนการแบบสองอินพุตสองเอาต์พุตมักมีปัญหาในการควบคุมระบบเพราะมีการรบกวนกันระหว่างค่าตัวแปรอินพุตและเอาต์พุตทั้งสอง การออกแบบระบบควบคุมแบบสองอินพุตสองเอาต์พุตที่มีการรบกวนกันเพื่อให้ได้ผลตอบสนองของระบบควบคุมตามต้องการ จึงต้องอาศัยความรู้ความเข้าใจด้านคุณลักษณะของกระบวนการ ทฤษฎีการออกแบบระบบควบคุมด้วยเทคนิคต่างๆ รวมถึงทักษะความรู้ทางด้านการใช้งานอุปกรณ์ตรวจวัด ตัวแปลงสัญญาณและตัวควบคุม ดังนั้นชุดทดลองกระบวนการระดับน้ำ 4 ถึงจึงเป็นโจทย์ปัญหามาตรฐานสำหรับการออกแบบระบบควบคุม ซึ่งถูกออกแบบมาเพื่ออธิบายถึงขีดจำกัดสมรรถนะของระบบควบคุมแบบสองอินพุตสองเอาต์พุต อันเนื่องมาจากตำแหน่งซีโรของกระบวนการ สามารถใช้สำหรับการปฏิบัติการทดลองเพื่อฝึกฝนทักษะทางด้านการวัดและออกแบบระบบควบคุม เป็นชุดทดลองที่สามารถสร้างองค์ความรู้พื้นฐานให้แก่ นักศึกษา ในหลักสูตรวิศวกรรมการวัดคุมได้อย่างมีประสิทธิภาพ

1.2 วัตถุประสงค์ของการวิจัย

1) เพื่อออกแบบและพัฒนาต้นแบบชุดทดลองกระบวนการระดับน้ำแบบ 4 ถึง สำหรับการปฏิบัติ การทดลองด้านการวัดและการออกแบบระบบควบคุมสำหรับกระบวนการแบบสองอินพุตสองเอาต์พุต

2) เพื่อจัดทำใบงานประกอบการทดลองเพื่อสร้างความรู้พื้นฐานทางด้านการวัดและการออกแบบระบบควบคุม

1.3 ขอบเขตของการวิจัย

ออกแบบและพัฒนาต้นแบบชุดทดลองกระบวนการระดับน้ำแบบ 4 ถึง พร้อมจัดทำใบงานการทดลอง

1.4 วิธีดำเนินการวิจัย

- 1) ศึกษาค้นคว้าหาข้อมูลชุดทดลองกระบวนการระดับน้ำ 4 ถัง
- 2) ออกแบบชุดทดลองกระบวนการระดับน้ำ 4 ถัง
- 3) จัดหาอุปกรณ์สำหรับประกอบชุดทดลองกระบวนการระดับน้ำ 4 ถัง
- 4) ประกอบและทดสอบการใช้งานกระบวนการระดับน้ำ 4 ถัง
- 5) จัดทำใบงานการทดลองและคู่มือใช้งาน
- 6) เขียนรายงานสรุปการดำเนินโครงการ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ต้นแบบชุดทดลองกระบวนการระดับน้ำ 4 ถัง สำหรับอธิบายผลของพฤติกรรมของระบบแบบสองอินพุตสองเอาต์พุตที่มีต่อประสิทธิภาพของระบบควบคุม
- 2) ใบงานการทดลองการออกแบบระบบควบคุมสำหรับกระบวนการระดับน้ำ 4 ถัง

บทที่ 2

แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 การทบทวนวรรณกรรม (Literature review)

กระบวนการระดับน้ำแบบ 4 ถัง ถูกออกแบบและพัฒนาขึ้นในปี ค.ศ. 1996 [1] ที่สถาบัน Lund Institute of Technology ประเทศสวีเดน เพื่อใช้เป็นกระบวนการทดลองสำหรับอธิบายเทคนิคการควบคุมระบบแบบหลายตัวแปร ถูกใช้เพื่ออธิบายให้นักศึกษาเข้าใจถึงผลของตำแหน่งซีโรของระบบแบบหลายตัวแปรที่มีต่อประสิทธิภาพของระบบควบคุม โดยเป็นชุดทดลองในหลักสูตรวิศวกรรมควบคุมสำหรับอธิบายตัวอย่างการหาโมเดลของระบบจากผลทดลอง การควบคุมแบบเวลาจริง การออกแบบระบบควบคุมด้วยเทคนิคต่างๆ เช่น PID, Decentralized PID [3], Auto-Tuning PID [4], H_∞ Control [5], Adaptive Control เป็นต้น

ด้วยคุณสมบัติของกระบวนการระดับน้ำแบบ 4 ถัง ที่สามารถสร้างความรู้ความเข้าใจพฤติกรรมของกระบวนการและฝึกฝนทักษะในการออกแบบระบบควบคุมให้นักศึกษาวิศวกรรมควบคุมได้เป็นอย่างดี คณะผู้วิจัยจึงได้มีแนวคิดออกแบบและพัฒนากระบวนการระดับน้ำแบบ 4 ถัง เพื่อใช้สำหรับการวิจัยในหัวข้อต่างๆ เช่น “I-P Controller Design for Quadruple-Tank System” [6], “T-DOF PID Controller Design using Characteristic Ratio Assignment Method for Quadruple Tank Process” [7], “Design of Decoupled Controller for TITO System using Characteristic Ratio Assignment” [8], “Modeling of the Modified Quadruple-Tank Process” [9], “Design PID Controller for the Modified Quadruple-Tank Process using Root Locus” [10], “Design of PI Controller Using Decoupling and CRA Techniques for Quadruple-Tanks Process”, [11], “Design of PID Controller for the Modified Quadruple-Tank Process using Inverted Decoupling Technique” [12]

2.2 วรรณกรรมที่ถูกอ้างอิง

[1] K. H. Johansson, “The Quadruple-Tank Process: A Multivariable Laboratory Process with an Adjustable zero”, IEEE Trans. on Control Systems Technology, 8, 456-465, 2000.

[2] Johansson, K.H; Horch,A; Wijk,O; Hansson,A. “Teaching multivariable control using the quadruple-tank process.” Conference on Proceeding of th 38th IEEE, 1999.

[3] J. L. R. Nunes. “Modeling and control of the Quadruple-Tank Process. Master thesis, Department of Automatic Control”, Lund Institute of Technology, Sweden, 1997.

[4] V. Recica. “Automatic tuning of multivariable controllers”. Master thesis, Department of Automatic Control, Lund Institute of Technology, Sweden, 1998.

[5] M. Grebeck. “A comparison of controllers for the quadruple tank system. Technical report, Department of Automatic Control”, Lund Institute of Technology, Sweden, 1998.

[6] Arjin Numsomran, Tianchai Suksri, Viriya Kongratana, Thanit Trisuwannawat and Kitti Tirasesth, "I-P Controller Design for Quadruple-Tank System", TENCON 2004.

[7] Tianchai Suksri, U-thai Sritheeravirojana, Arjin Numsomran, Viriya Kongrattana, and Thongchai Werataweemart, "T-DOF PID Controller Design using Characteristic Ratio Assignment Method for Quadruple Tank Process", Transactions on Engineering, Computing and Technology, 2006.

[8] Arjin Numsomran, Tianchai Suksri, Maitree Thumma, "Design of Decoupled Controller for TITO System using Characteristic Ratio Assignment", ICCAS 2007

[9] Arjin Numsomran, Vittaya Tipsuwanporn, Kitti Tirasesth, "Modeling of the Modified Quadruple-Tank Process", SICE2008

[10] Arjin Numsomran, Vittaya Tipsuwanporn, Thanit Trisuwannawat and Kitti Tirasesth, "Design PID Controller for the Modified Quadruple-Tank Process using Root Locus", ICCAS2008

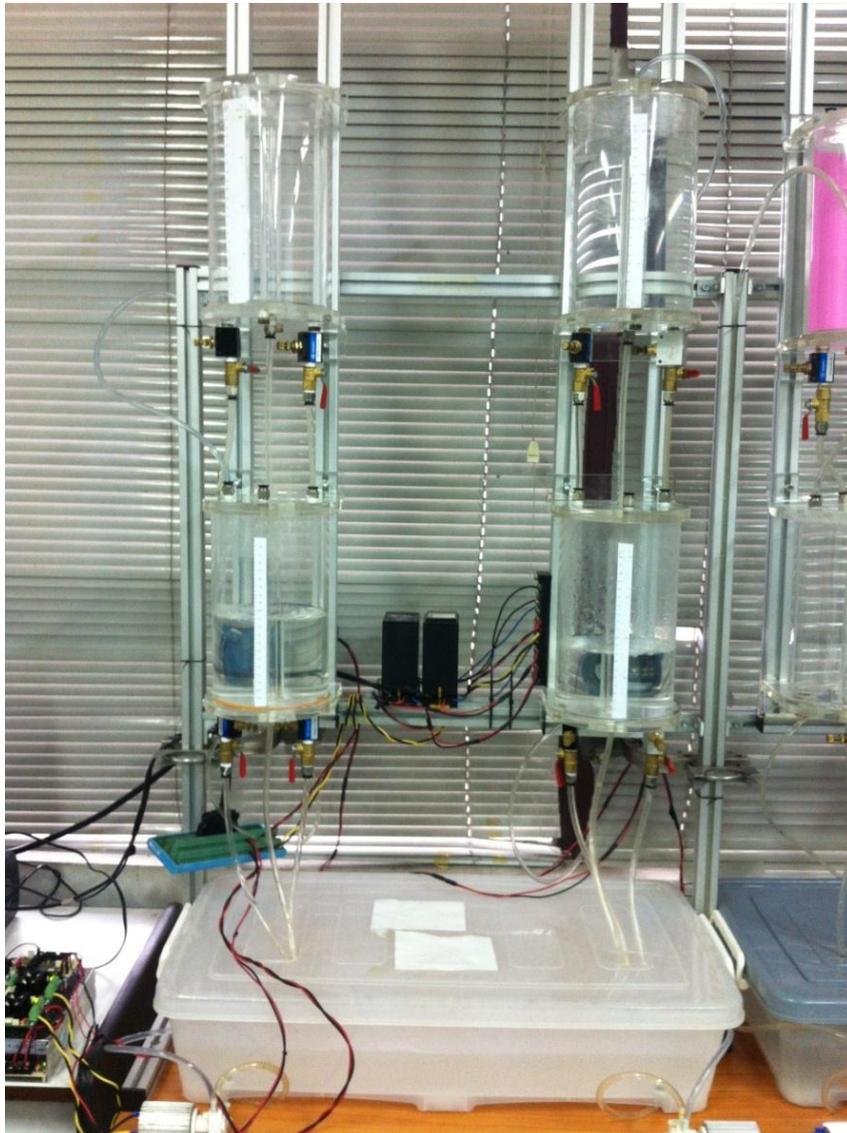
[11] Tianchai Suksri, Arjin Numsomran, Vittaya Tipsuwanporn, Thanit Trisuwannawat and Kitti Tirasesth, "Design of PI Controller Using Decoupling and CRA Techniques for Quadruple-Tanks Process", ICCAS2011

[12] Arjin Numsomran, Vittaya Tipsuwanporn, Thanit Trisuwannawat and Kitti Tirasesth, "Design of PID Controller for the Modified Quadruple-Tank Process using Inverted Decoupling Technique", ICCAS2011

บทที่ 3

องค์ประกอบของกระบวนการระดับน้ำ 4 ถึง

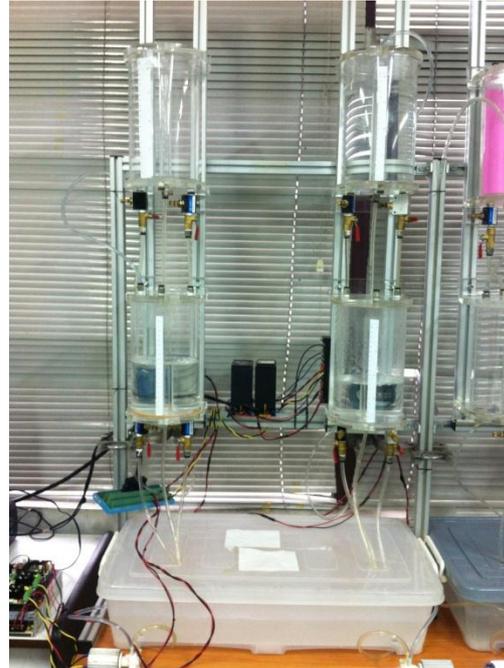
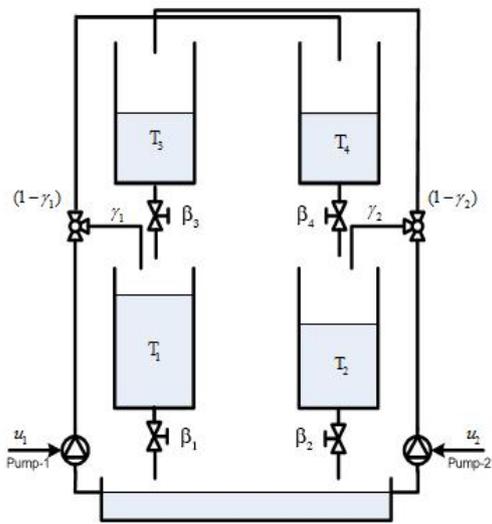
ชุดทดลองกระบวนการระดับน้ำ 4 ถึง ประกอบด้วยอุปกรณ์ต่างๆที่สำคัญ 7 ส่วนด้วยกัน คือ กระบวนการระดับน้ำ 4 ถึง อุปกรณ์ตรวจวัดความดัน วงจรปรับแต่งสัญญาณแรงดันที่ได้รับจากเซ็นเซอร์ วงจรขับกระแส ปั๊มน้ำ แหล่งจ่ายไฟ และ ชุดบอร์ดควบคุม ส่วนเอาต์พุตของระบบคือ ระดับน้ำของถัง ด้านล่างทั้ง 2 ใบ อินพุตของระบบคือสัญญาณควบคุมที่ป้อนให้แก่ ชุดขับปั๊มน้ำ ทิศทางการไหลของน้ำและ คุณลักษณะของกระบวนการสามารถปรับเปลี่ยนตามตำแหน่งการเปิด-ปิดของวาล์วทางเข้าของน้ำ ซึ่ง ตำแหน่งซีโรของกระบวนการสามารถปรับตำแหน่งให้อยู่ได้ทั้งด้านซ้ายและด้านขวาของระนาบ s โดยการ ปรับเปลี่ยนตำแหน่งวาล์วทางเข้าของน้ำ รายละเอียดของอุปกรณ์ต่างๆที่ใช้ในการทดลองนั้นสามารถ อธิบายได้ดังต่อไปนี้



ภาพที่ 3.1 ชุดทดลองกระบวนการระดับน้ำ 4 ถึง

3.1 กระบวนการระดับน้ำ 4 ถัง

กระบวนการระดับน้ำ 4 ถัง ประกอบด้วยถังน้ำอะคริลิกมีขนาดพื้นที่หน้าตัด 169.56 cm^2 และสูง 20.0 เซนติเมตร 4 ใบ โดยถังน้ำแต่ละใบจะมีวาล์ว ที่ใช้ในการปรับพื้นที่หน้าตัดรูน้ำด้านออกจากถัง แสดงได้ดังภาพที่ 3.2 น้ำที่ไหลเข้าสู่ถังน้ำแต่ละใบนั้นจะถูกปั๊มขึ้นมาจากบ่อเก็บน้ำด้านล่าง โดยปั๊มน้ำกระแสตรง 24 โวลต์ ถูกใช้เพื่อจ่ายน้ำให้ถึงแต่ละใบผ่านสายยางอ่อนขนาดเส้นผ่านศูนย์กลางภายใน 8 มิลลิเมตรและผ่านวาล์ว ที่ใช้ในการปรับความต้านทานการไหลสำหรับปรับเปลี่ยนโหมดการควบคุมของกระบวนการ



ภาพที่ 3.2 โครงสร้างกระบวนการระดับน้ำ 4 ถัง

วาล์วที่ใช้ในการปรับพื้นที่หน้าตัดรูน้ำทางด้านออกของถังน้ำแต่ละใบ แสดงได้ดังภาพที่ 3.3



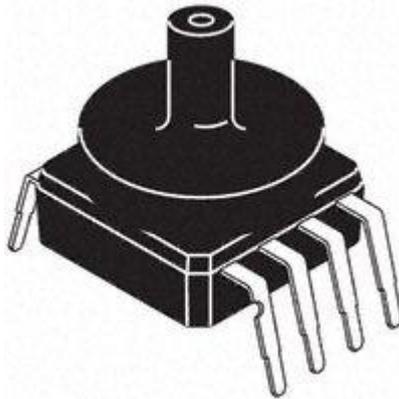
ภาพที่ 3.3 วาล์วที่ใช้ในการปรับพื้นที่หน้าตัดรูน้ำด้านออก

3.2 อุปกรณ์วัดระดับน้ำ (Gauge Pressure Sensor)

อุปกรณ์ที่ใช้วัดระดับน้ำที่เลือกใช้สำหรับกระบวนการนี้จะเลือกใช้ อุปกรณ์วัดแรงดันแบบความดันเกจ (Pressure Sensor 3.92kPa Gauge 1-Port) ของ Freescale Semiconductor รุ่น MPXV5004GC7U Range 0-3.92 kPa ค่าความแม่นยำคือ $\pm 1.5\%$ of Span ใช้พาวเวอร์ซัพพลาย 5.25 VDC และ ให้เอาต์พุต 1-5 Volts รูปของอุปกรณ์วัดระดับน้ำแสดงได้ดังภาพที่ 3.4

ตารางที่ 3.1 รายละเอียดทางเทคนิคของอุปกรณ์วัดระดับน้ำ (Differential Pressure Transmitter)

Reference Accuracy	+1.5% of Span
Analogue Output	1-5 Volts
Maximum Operating Temperature	+85 °C
Maximum Pressure Reading	3.92 kPa
Minimum Pressure Reading	0 kPa
Media Measured	Air, Fluid
Pressure Reading Type	Gauge
Supply Voltage	4.75 → 5.25 V dc



ภาพที่ 3.4 อุปกรณ์วัดระดับน้ำ (Pressure Sensor 3.92kPa Gauge 1-Port)

การวัดระดับน้ำนั้นจะใช้สายยางขนาด 8 มิลลิเมตรที่ต่อมาจากด้านบนถึงของถังน้ำไปที่ 1 และ 2 ต่อเข้าทางด้านอินพุตของตัวตรวจวัดแรงดัน ค่าความดันเกจที่วัดโดยอุปกรณ์ตรวจวัดความดันจึงสัมพันธ์กับระดับน้ำในถังดังต่อไปนี้

$$\Delta P = \rho gh \quad (3.1)$$

โดย ρ คือ ความหนาแน่นของน้ำ 1000 kg/cm^3

g คือ แรงโน้มถ่วงของโลก 9.81 m/s^2

h คือ ระดับน้ำที่วัด m

ΔP คือ ความดันแตกต่าง KPa

ในการทดลองกำหนด ช่วงการวัดอยู่ที่ระยะ 0 เซนติเมตรถึง 20.0 เซนติเมตร

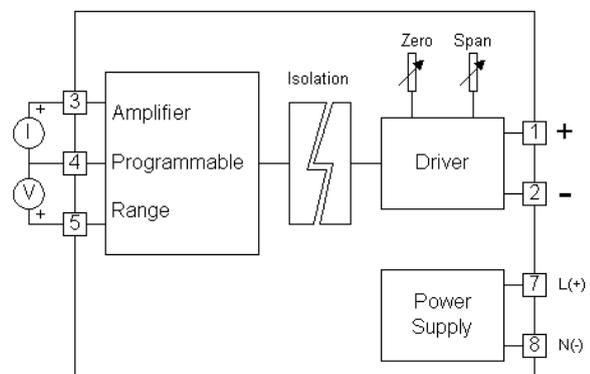
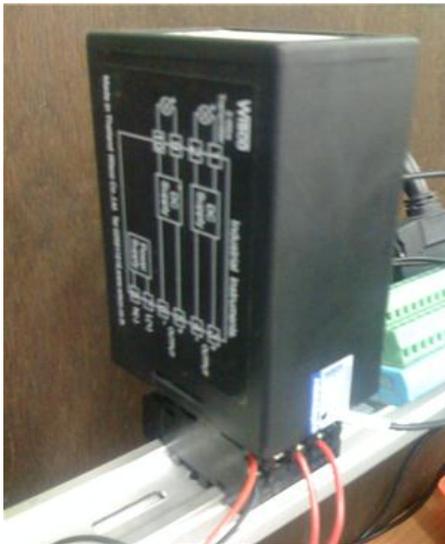
$$\begin{aligned}\Delta P &= (1000 \times 9.81 \times 0.20) / 1000 \\ &= 1.962 \text{ KPa}\end{aligned}\quad (3.2)$$

ดังนั้นจะได้ช่วงความดันเกจที่สัมพันธ์กับช่วงการวัดระดับน้ำ เท่ากับ

$$\Delta P = 0 - 1.962 \text{ KPa}$$

เมื่อกำหนดช่วงความดันแตกต่างข้างต้นให้เป็นช่วงการวัดระดับ 0% ถึง 100% ของระดับน้ำในถังที่ 1 และ 2 อุปกรณ์วัดแรงดันแบบความดันเกจ จะให้เอาต์พุตเป็นค่าแรงดันไฟฟ้าไฟฟ้า 1-5 Volts ที่สัมพันธ์กับระดับน้ำในถังที่ 1 และ 2 ที่ความสูง 0 ถึง 20.0 เซนติเมตร

3.3 อุปกรณ์ปรับแต่งสัญญาณ 4-20mA เป็น 1-5V



ภาพที่ 3.5 อุปกรณ์ปรับแต่งสัญญาณ 4-20mA เป็น 1-5V

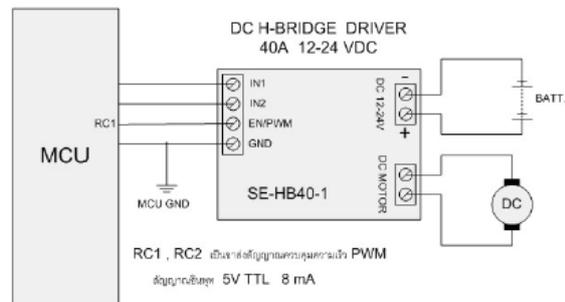
SC-95 เป็นอุปกรณ์ที่ทำหน้าที่แปลงสัญญาณจาก 4 ถึง 20 มิลลิแอมป์ เป็น 1 ถึง 5 โวลต์ เพื่อส่งต่อไปยังการ์ดแปลงสัญญาณ A/D และ D/A ข้อมูลคุณสมบัติและรายละเอียดของอุปกรณ์ปรับแต่งสัญญาณ แสดงได้ดังตารางที่ 3.3

ตารางที่ 3.2 รายละเอียดทางเทคนิคของอุปกรณ์ปรับแต่งสัญญาณ SC-95

Input voltage	Rangeable from 10 mVDC to 100VDC
Input current	Rangeable from 100uA DC to 100mA DC
Output	4-20 mA, 0-20 mV/ 0-1 V, 0-5 V, 1-5 V, 0-10 V
Linearity	< + 0.2% of span
Power Supply	100 VAC., 110 VAC., 220 VAC., 24 VDC

3.4 ชุดขับปั๊มน้ำ

ใช้ในการควบคุมการทำงานของมอเตอร์ปั๊มน้ำ โดยรับสัญญาณ PWM (Pulse Width Modulation) จากไมโครคอนโทรลเลอร์เพื่อนำไปควบคุมมอเตอร์ดีซี โดยบอร์ดขับมอเตอร์ดีซีนี้เป็นบอร์ดสำเร็จรูป ทำงานโดยใช้มอเตอร์แบบ H-Bridge รุ่น SE-HB40-1



ภาพที่ 3.6 ชุดขับปั๊มน้ำ SE-HB40-1

SC-95 เป็นอุปกรณ์ที่ทำหน้าที่แปลงสัญญาณจาก 4 ถึง 20 มิลลิแอมป์ เป็น 1 ถึง 5 โวลต์ เพื่อส่งต่อไปยังการ์ดแปลงสัญญาณ A/D และ D/A ข้อมูลคุณสมบัติและรายละเอียดของอุปกรณ์ปรับแต่งสัญญาณ แสดงได้ดังตารางที่ 3.3

ตารางที่ 3.3 รายละเอียดทางเทคนิคของชุดขับปั๊มน้ำ

Output	<ul style="list-style-type: none"> - Motor DC Supply 12-24 V 40A (Max.) - Full-Complementary Power MOSFET Driver With ultrafast reverse recovery protection diodes
Input	<ul style="list-style-type: none"> - Full Opto-isolated input interface signals - 5V 8 mA TTL – Level
Drive Mode	<ul style="list-style-type: none"> - ON – OFF Control - Direction Control - Speed Control (PWM Drives)
PWM Frequency	- 400 Hz - 1000 Hz (800 Hz Recommend)

3.5 ป้อนน้ำ

ปั้มน้ำที่ใช้ในการทดลองนั้นจะใช้ของบริษัท API Instruments รุ่น P6 10.0006 ซึ่งจะเป็นปั้มน้ำแบบเกียร์ปั้ม โดยรูปของปั้มน้ำที่ใช้ในการทดลองแสดงได้ดังภาพที่ 3.7 ส่วนข้อมูลคุณสมบัติและรายละเอียดของปั้มน้ำแสดงได้ดังตารางที่ 3.4



ภาพที่ 3.7 ปั้มน้ำ API Instrument

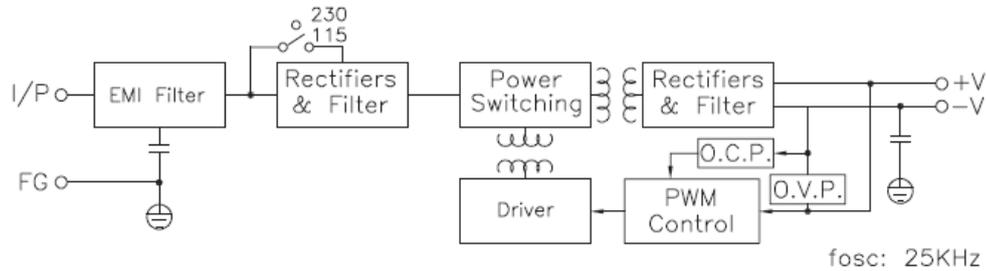
ตารางที่ 3.4 รายละเอียดทางเทคนิคของปั้มน้ำ

Max Flow rate	2200 ml/min (Vdc)
Max. Suction	10 ft. H ₂ O wet, 4" dry
Max. Pressure	20 psig (46 ft. H ₂ O)
Port Size (OD)	4.8 mm (0.18") hose barb
Power Require (Vdc)	24 Vdc (up to 28W)
Max Operating Temp	930C (2000F)
Ambient Fluid Temp	0 to 930C (32 to 2000F)
Particulates	No
Run dry	No
Max. Viscosity	200 cps
Dimensions	88×81×92 mm (4×1.75×2.63)
Weight	1.4 kg (3 lb.)

3.6 แหล่งจ่ายไฟสำหรับอุปกรณ์

แหล่งจ่ายไฟสำหรับอุปกรณ์ขับเคลื่อนของปั้มน้ำนั้นจะใช้แหล่งจ่ายไฟแบบสวิตชิง ขนาด ± 24 V, 6.5 A โดยรูปบล็อกไดอะแกรมของแหล่งจ่ายไฟแสดงดังภาพที่ 3.8 และรูปอุปกรณ์แหล่งจ่ายไฟแสดงดัง

ภาพที่ 3.9 ข้อดีของสวิตชิ่งเพาเวอร์ซัพพลายเมื่อเปรียบเทียบกับแหล่งจ่ายไฟแบบเชิงเส้น คือมีประสิทธิภาพที่สูง ขนาดเล็ก และน้ำหนักเบากว่าแหล่งจ่ายไฟแบบเชิงเส้น เนื่องจากแหล่งจ่ายไฟแบบเชิงเส้นใช้หม้อแปลงความถี่ต่ำจึงมีขนาดใหญ่ และน้ำหนักมาก ขณะใช้งานจะมีแรงดันและกระแสผ่านตัวหม้อแปลงตลอดเวลา กำลังงานสูญเสียที่เกิดจากหม้อแปลงจึงมีค่าสูง



ภาพที่ 3.8 บล็อกไดอะแกรมของแหล่งจ่ายไฟขนาด ± 24 V, 6.5 A



ภาพที่ 3.9 อุปกรณ์แหล่งจ่ายไฟขนาด ± 24 V, 6.5 A

ตารางที่ 3.5 ตารางแสดงคุณสมบัติของ Switching Power Supply

Output DC Voltage	24 V
Output Rated Current	6.5 A
Output Current Range	0–6.5 A
Output Rated Power	156 W
Output Voltage Adjust Range	21–28 V
Input Voltage Range	85-132 VAC/170-264 VAC
Input Frequency Range	50-60 Hz
Input AC Current	2.0 Amax
Input Leakage Current	3.5 mA/240V
Overload	105-150% Rate Output Power
Overvoltage	30-34.8V
Working Temp.	-10 - +60

3.7 ชุดบอร์ดควบคุมกระบวนการ



ภาพที่ 3.10 ไมโครคอนโทรลเลอร์ ARM7

บอร์ด ARM7 ที่ใช้ในงานวิจัยนี้เป็นบอร์ดที่มีไมโครคอนโทรลเลอร์ ARM7 เบอร์ ADuC7024 ของบริษัท Analog Device ต่อใช้งาน ซึ่งบอร์ดนี้มีหน้าที่เพื่อรับและส่งสัญญาณระหว่างคอนโทรลเลอร์กับกระบวนการ โดยสถาปัตยกรรมของ ARM7 เป็นซีพียูแบบ RISC ขนาด 32 บิต ภายในมีบัสขนาด 32 บิต ตัวเดียวที่ใช้สำหรับรับข้อมูลและคำสั่ง ชุดคำสั่งจะมีขนาด 32 บิต คงที่ ในขณะที่ข้อมูลสามารถเลือกได้ว่า จะมีขนาด 8, 16 หรือ 32 บิต

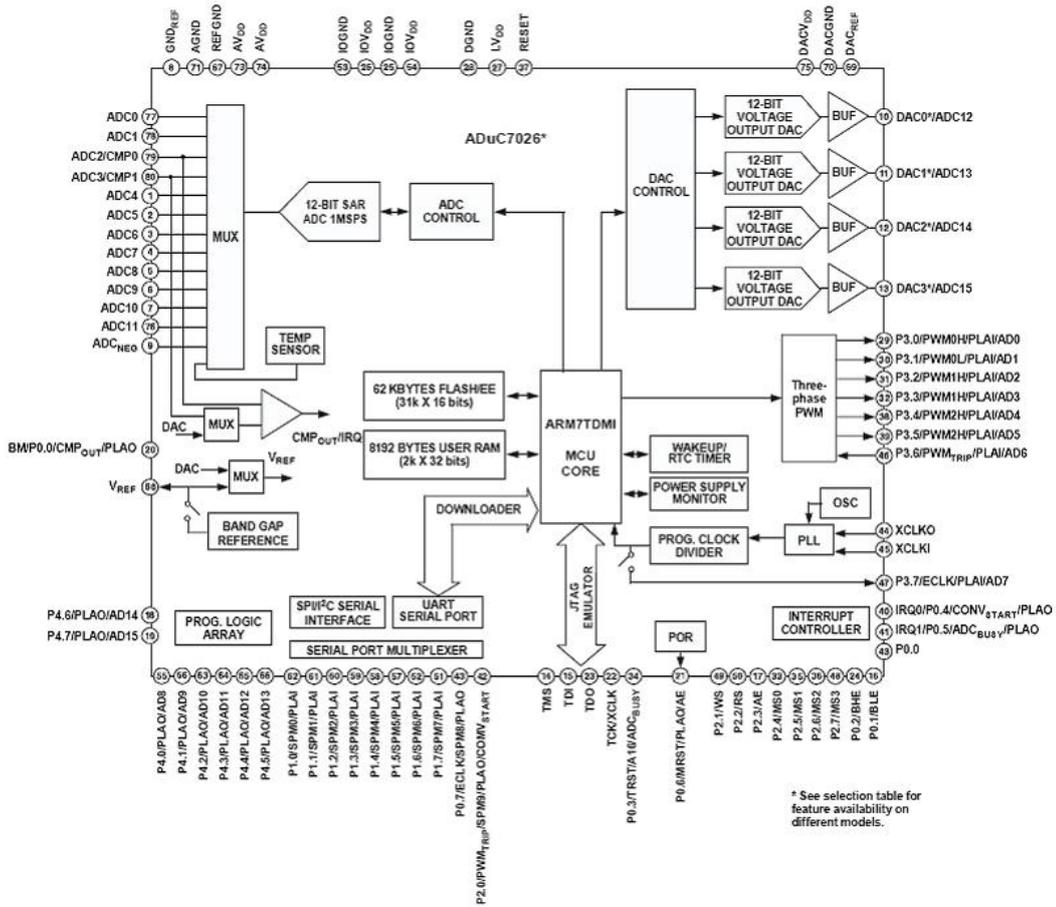
โครงสร้างของ ARM7 จะเป็นแบบที่เรียบง่าย มีชุดคำสั่งไม่มากนัก ประหยัดพื้นที่ สารกึ่งตัวนำที่ใช้สร้างประหยัดพลังงาน ในการประมวลผลข้อมูลใดๆ ต้องกระทำผ่านทางรีจิสเตอร์ เริ่มต้นด้วยการโหลดค่าจากหน่วยความจำ เก็บค่าในรีจิสเตอร์ นำค่ามาประมวลผล เสร็จแล้วจะเขียนค่าเก็บในหน่วยความจำ ดั่งเดิม รีจิสเตอร์ของ ARM7 ที่ใช้งานได้สำหรับผู้ใช้นี้ทั้งหมด 16 ตัวคือ R0-R15 โดยทุกตัวมีขนาด 32 บิต โดย R0-R12 เป็นรีจิสเตอร์ทั่วไปไม่ได้กำหนดหน้าที่การทำงานพิเศษ ส่วน R12 ทำหน้าที่เป็น stack pointer (SP) R14 ทำหน้าที่เป็น link register (LR) และ R15 ทำหน้าที่เป็น Program Counter (PC)

คุณสมบัติของบอร์ด

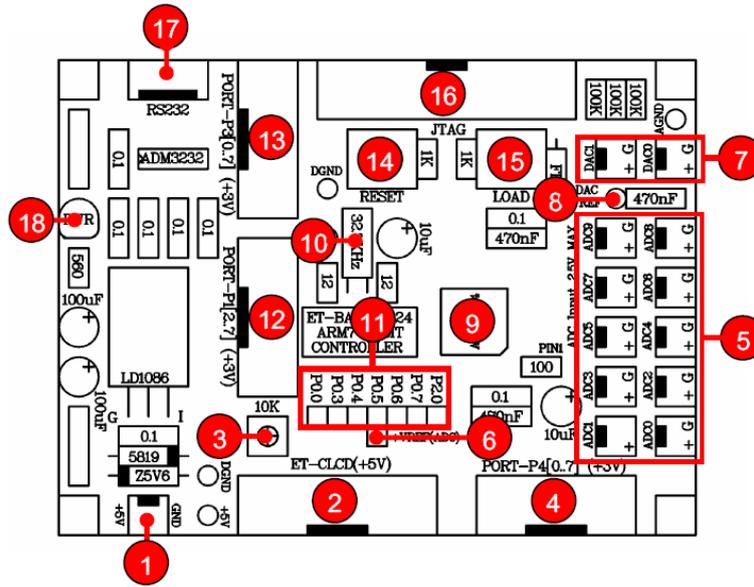
- เลือกใช้ MCU ตระกูล ARM7 TDMI Core เบอร์ ADuC7024 ของ Analog Device เป็น MCU ประจำบอร์ด โดยเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบ XTAL ค่า 32.768 KHz ซึ่งสามารถกำหนดการทำงานร่วมกับ Phase Lock Loop ให้ MCU สามารถประมวลผลด้วยความเร็วสูงสุดที่ 41.78 MHz ได้ด้วย โดยคุณสมบัติเด่นๆของ MCU ได้แก่

- มีหน่วยความจำ Flash สำหรับเขียนโปรแกรม 62KByte และ มี RAM ขนาด 8Kbyte
- มีพอร์ต I/O จำนวน 5 พอร์ตคือ P0(6Bit),P1(8Bit),P2(1Bit),P3(8Bit) และ P4(8Bit)
- มีวงจรรีจิสเตอร์อนุกรม UART จำนวน 1 พอร์ต และมีวงจรรีจิสเตอร์ SPI จำนวน 1 พอร์ต
- มีวงจรรีจิสเตอร์ Timer/Counter จำนวน 4 ชุด
- มีวงจรรีจิสเตอร์ ADC ขนาด 12บิต จำนวน 10 ช่อง และ DAC ขนาด 12 บิต จำนวน 2 ช่อง
- มีวงจรรีจิสเตอร์ Watchdog, Power-ON Reset, PWM

- มีขั้วต่อสัญญาณ I/O แบบ TTL แบบ Header 2x5 จำนวน 3 ชุด (P1,P3 และ P4)
- มีขั้วต่อ LCD แบบ Header 2x7 รองรับการทำงานเชื่อมต่อกับ LCD Character (เชื่อมต่อแบบ 4 บิต)
- มีขั้วต่อใช้งาน RS232 สำหรับใช้งาน และ สำหรับ Download ผ่าน RS232
- ใช้แหล่งจ่ายไฟขนาด +5VDC ร่วมกับ Regulate 3.3V on Board พร้อม LED สถานะ Power
- ขนาด PCB Size เล็กเพียง 8 x 6 cm.



ภาพที่ 3.11 สถาปัตยกรรมภายในของไมโครคอนโทรลเลอร์ ARM7 เบอร์ ADuC7024



ภาพที่ 3.12 บอร์ด ET-BASE ARM7024 (ADUC7024)

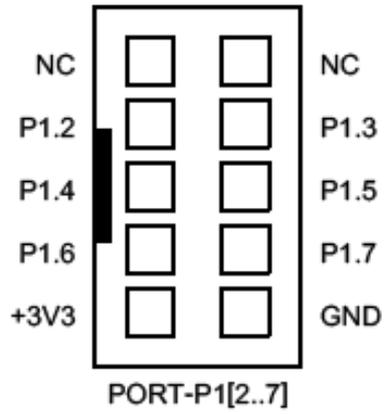
ในภาพที่ 3.12 แสดงองค์ประกอบของบอร์ดไมโครคอนโทรลเลอร์ ADUC7024 ซึ่งมีรายละเอียดดังต่อไปนี้

- หมายเลข 1 คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงวงจรของบอร์ด ใช้กับแหล่งจ่ายไฟตรง +5VDC
- หมายเลข 2 คือ Port-LCD ชนิด Character Type ใช้การเชื่อมต่อแบบ 4 บิต ผ่าน Port - P4 [0..6]
- หมายเลข 3 คือ ตัวต้านทานสำหรับปรับค่าความสว่างให้ LCD
- หมายเลข 4 คือ Port-P4 มี ขนาด 8 Bit คือ P4 [0..7]
- หมายเลข 5 คือ ขั้วต่อ ADC จำนวน 10 ช่อง คือ ADC [0..9]
- หมายเลข 6 คือ จุดต่อแรงดันอ้างอิงให้ ADC จากภายนอก
- หมายเลข 7 คือ ขั้วต่อ DAC จำนวน 2 ช่อง คือ DAC [0..1]
- หมายเลข 8 คือ จุดต่อแรงดันอ้างอิงให้ DAC จากภายนอก
- หมายเลข 9 คือ MCU เบอร์ ADUC7024 ซึ่งเป็น MCU ตระกูล ARM7TDMI จาก Analog Device
- หมายเลข 10 คือ Crystal ค่า 32.768 KHz
- หมายเลข 11 คือ Port-P0 มี ขนาด 6 Bit คือ P0[0,3,4,5,6,7] และ Port-P2 มี 1 บิต คือ P2 [0]
- หมายเลข 12 คือ Port-P1 มี ขนาด 6 Bit คือ P1 [2..7]
- หมายเลข 13 คือ Port-P3 มี ขนาด 8 Bit คือ P3 [0..7]
- หมายเลข 14 คือ Switch RESET ใช้สำหรับ Reset การทำงานของ CPU
- หมายเลข 15 คือ Switch LOAD ใช้ร่วมกับ RESET สำหรับสั่ง Download HEX ให้ MCU
- หมายเลข 16 คือ ขั้วต่อ ARM-JTAG สำหรับ Debug โปรแกรม
- หมายเลข 17 คือ ขั้วต่อ RS232 สำหรับใช้งานทั่วไป และ Download HEX ให้ MCU
- หมายเลข 18 คือ LED Power ใช้แสดงสถานะของแหล่งจ่ายไฟ +5VDC

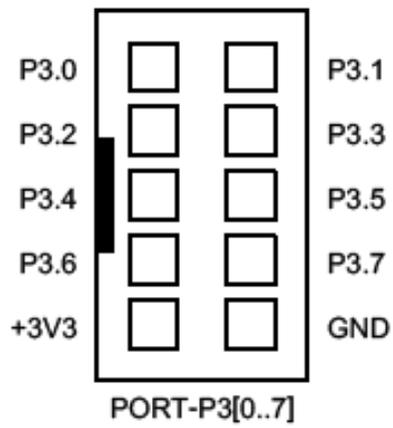
ขั้วสัญญาณต่าง ๆ

สำหรับขั้วต่อสัญญาณของพอร์ต I/O จาก MCU นั้น จะถูกออกแบบและจัดเตรียมไว้ผ่านทาง ขั้วต่อแบบ IDE Header ขนาด 10Pin (2x5) จำนวน 3 ชุด คือ PORT-P1, PORT-P3 และ PORT-P4 ตามลำดับ ส่วน PORT-P0 และ PORT-P2 นั้นจะต่อเป็น Header ขนาด 1x7 ไร่ โดยที่ขั้วต่อสัญญาณแต่ละชุด จะ ประกอบไปด้วยสัญญาณของ I/O ที่เชื่อมต่อมาจากขาสัญญาณของ MCU โดยตรงทั้งหมด โดยจุดเชื่อมต่อกับสัญญาณภายนอกของบอร์ด มีดังนี้

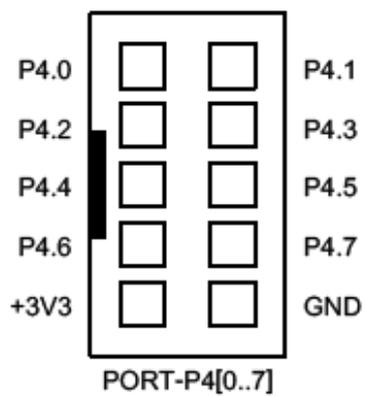
- ขั้วต่อแหล่งจ่ายขนาด +5VDC
- ขั้วต่อ PORT-P0 มี 6 บิต คือ P0 [0,3,4,5,6,7]
- ขั้วต่อ PORT-P1 มี 6 บิต คือ P1 [2,3,4,5,6,7] ส่วน P1.0 และ P1.1 จะจะถูกเชื่อมต่อผ่าน วงจร
- Line Driver (MAX232) สำหรับแปลงระดับสัญญาณจากระดับลอจิก TTL ของ MCU ให้ เป็น
- สัญญาณแรงดันตามมาตรฐานของ RS232 โดยสัญญาณที่ได้รับการแปลงเป็นแบบ RS232 จะถูก
- เชื่อมต่อโปรไวท์ขั้วต่อแบบ CPA ขนาด 4 PIN (RS232)
- ขั้วต่อ PORT-P2 มีขนาด 1 บิต คือ P2.0
- ขั้วต่อ PORT-P3 มีขนาด 8 บิต คือ P2 [0,1,2,3,4,5,6,7]
- ขั้วต่อ PORT-P4 มีขนาด 8 บิต คือ P4 [0,1,2,3,4,5,6,7] โดยมีการต่อสัญญาณไปยังขั้ว LCD ด้วย
- จำนวน 7 บิต คือ P4 [0,1,2,3,4,5,6] ตามลำดับ
- ขั้วต่อ ADC มีขนาด 10 ช่อง คือ ADC0-ADC9 โดยรับสัญญาณ Analog ได้ระหว่าง 0-2.5V
- จุดรับแรงดันอ้างอิงของ DAC (DAC Reference) จากภายนอกขนาด 0-3V
- ขั้วต่อ DAC มีขนาด 2 ช่อง คือ DAC0, DAC1 โดยสามารถสร้างสัญญาณ Analog ได้ 0-2.5V
- จุดรับแรงดันอ้างอิงของ ADC (+Vref ADC) จากภายนอกขนาด 0-2.5V
- ขั้วต่อ ARM-JATG
- ขั้วต่อสัญญาณ RS232



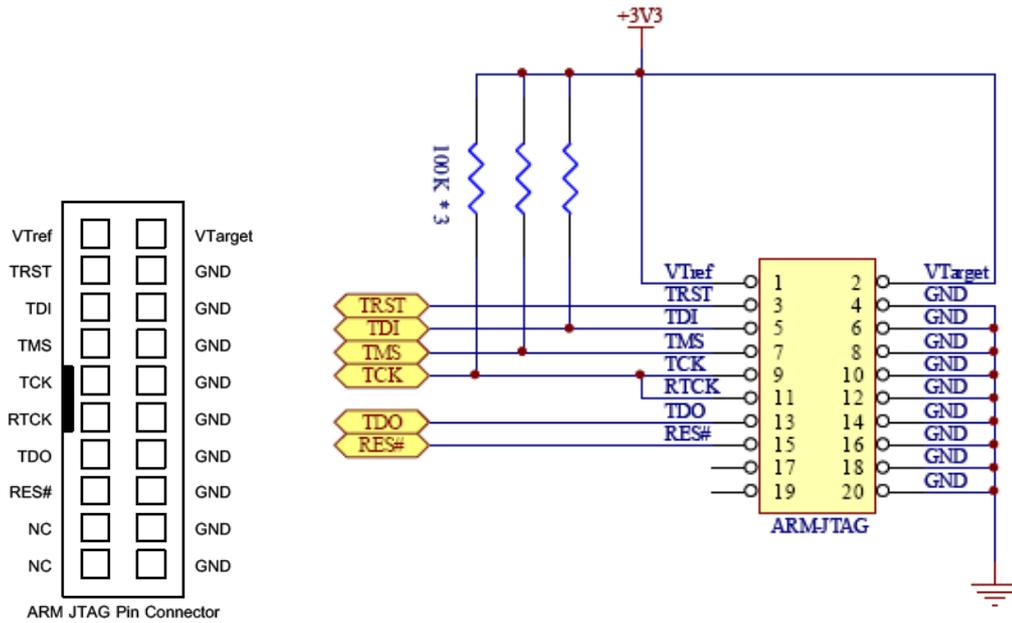
ภาพที่ 3.13 พอร์ต P1 มีขนาด 6 บิต



ภาพที่ 3.14 พอร์ต P3 มีขนาด 8 บิต

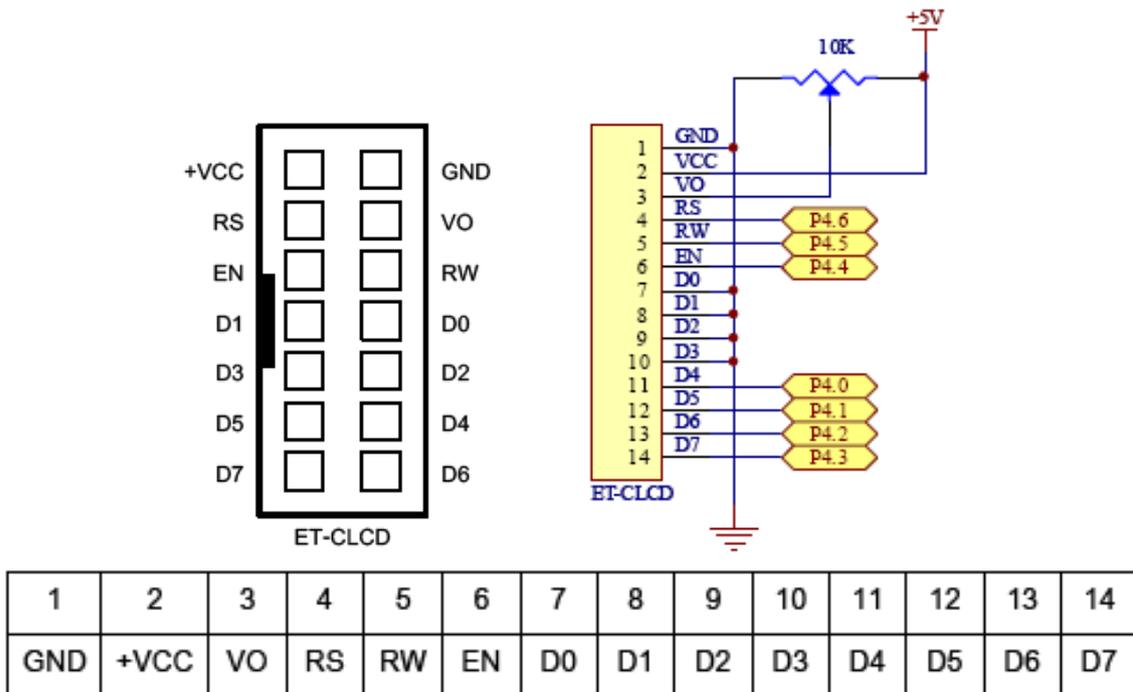


ภาพที่ 3.15 พอร์ต P4 มีขนาด 8 บิต



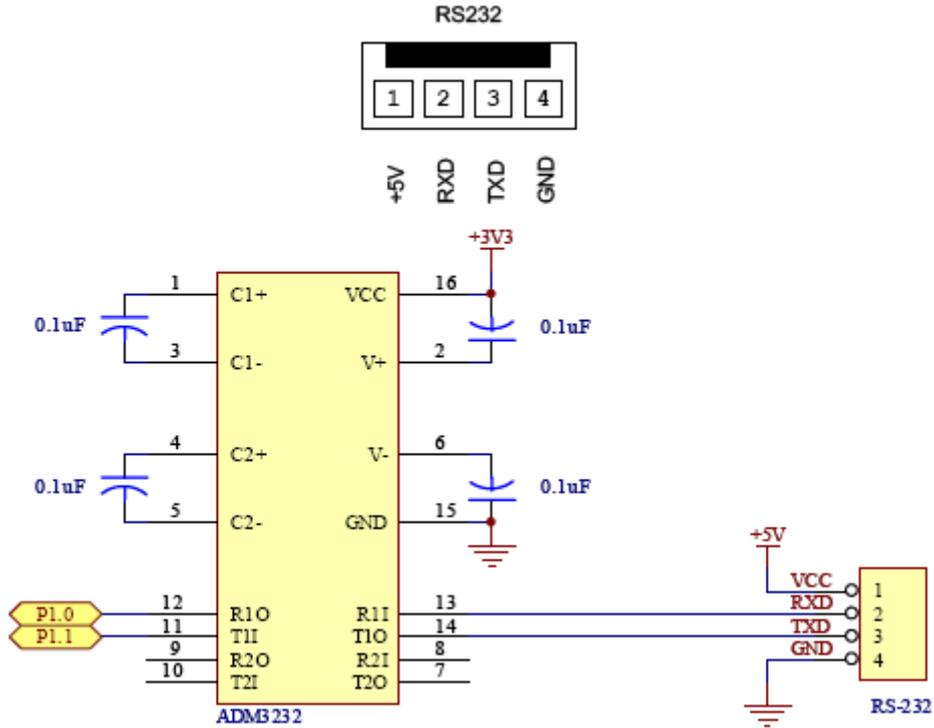
ภาพที่ 3.16 พอร์ต ARM-JTAG และวงจรส่วนที่ใช้ในการเชื่อมต่อกับ ARM-JTAG

พอร์ต CLCD ใช้กับ Character LCD โดยเชื่อมต่อแบบ 4 บิต โดยสัญญาณที่ใช้เชื่อมต่อกับ LCD เป็นสัญญาณชุดเดียวกับที่ต่อไปยังขั้วต่อของ PORT-P4 โดยในการเชื่อมต่อสายสัญญาณจากขั้วต่อของพอร์ต LCD ไปยังจอแสดงผล LCD นั้นมีการจัดเรียงขาสัญญาณของ Character LCD แสดงดังภาพที่ 3.18

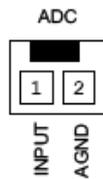


ภาพที่ 3.17 การจัดเรียงขาสัญญาณของ Character LCD มาตรฐาน

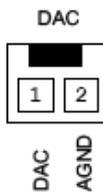
พอร์ต RS232 เป็นสัญญาณ RS232 ซึ่งผ่านวงจรแปลงระดับสัญญาณ MAX232 สามารถใช้เชื่อมต่อกับสัญญาณ RS232 เพื่อรับส่งข้อมูล และสามารถใช้งาน ร่วมกับ Switch PSEN และ Switch RESET เพื่อทำการ Download แบบ Manual



ภาพที่ 3.18 วงจรส่วนที่เชื่อมต่อกับ RS232



ภาพที่ 3.19 ขั้วต่อ ADC0 – ADC9

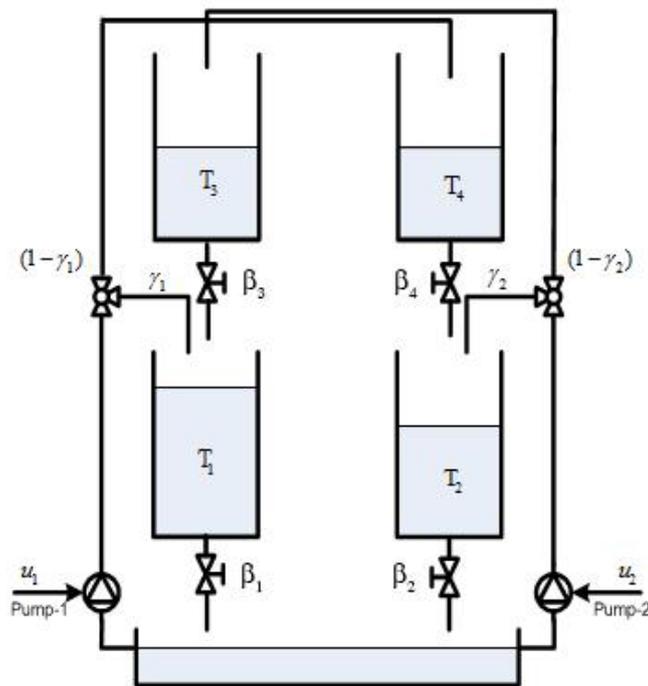


ภาพที่ 3.20 ขั้วต่อ DAC0 และ DAC1

บทที่ 4

แบบจำลองทางคณิตศาสตร์และการวิเคราะห์คุณลักษณะของกระบวนการน้ำ 4 ถัง

กระบวนการระดับน้ำ 4 ถัง เป็นกระบวนการในการศึกษาและการทดลองแบบใหม่ซึ่งถูกแนะนำโดย K.H.Johansson [1] ซึ่งกระบวนการนี้ได้ถูกออกแบบมาเพื่อทำการอธิบายถึงขีดจำกัดสมรรถนะของระบบควบคุมแบบ MIMO อันเนื่องมาจากตำแหน่งของซีโรของกระบวนการ โดยกระบวนการนั้นจะประกอบด้วยถังน้ำ 4 ใบและปั้มน้ำ 2 ตัว ซึ่งอินพุตที่เข้าสู่กระบวนการคือค่าแรงดันที่จ่ายให้กับปั้มน้ำทั้ง 2 ตัวและเอาต์พุตของกระบวนการคือ ระดับน้ำของถังน้ำด้านล่างทั้ง 2 ใบ เมื่อทำการวิเคราะห์แบบจำลองเชิงเส้นของกระบวนการระดับน้ำ 4 ถัง จะพบว่าตำแหน่งซีโรของกระบวนการนั้นสามารถปรับตำแหน่งให้อยู่ได้ทั้งด้านซ้ายและด้านขวาของระนาบ s โดยการปรับเปลี่ยนค่าตำแหน่งขั้ว ซึ่งก็คือกระบวนการนี้สามารถเลือกโหมดการทำงานให้เป็นแบบเฟสต่ำสุดหรือเฟสไม่ต่ำสุดนั่นเอง



ภาพที่ 4.1 ชุดทดลองกระบวนการระดับน้ำ 4 ถัง

4.1 แบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ 4 ถัง

พิจารณากระบวนการระดับน้ำ 4 ถัง ในภาพที่ 4.1 เป้าหมายของการควบคุมกระบวนการหรือเอาต์พุตของระบบ คือระดับน้ำในถังที่ 1 และ 2 โดยอินพุตของกระบวนการคือ ค่าแรงดันไฟฟ้าที่จ่ายให้กับปั้มน้ำ (u_1) โดย γ_i คือวาล์วที่ใช้ในการปรับสัดส่วนการไหลของน้ำที่เข้าสู่แต่ละถัง ซึ่งในกรณีศึกษานี้จะกำหนดให้ γ_3 และ γ_4 มีค่าเท่ากับ $(1-\gamma_2)$ และ $(1-\gamma_1)$

แบบจำลองทางคณิตศาสตร์สำหรับแบบจำลองไม่เป็นเชิงเส้นของกระบวนการระดับน้ำ 4 ถังนี้ สามารถหาได้จากสมการสมดุลมวลและกฎของเบอร์นูลลี (Bernoulli's) ดังรายละเอียด ต่อไปนี้

$$\begin{aligned}
\frac{dh_1(t)}{dt} &= -\frac{\beta_1 a_1}{A_1} \sqrt{2gh_1(t)} + \frac{\beta_3 a_3}{A_1} \sqrt{2gh_3(t)} + \frac{\gamma_1 k_1}{A_1} u_1(t) \\
\frac{dh_2(t)}{dt} &= -\frac{\beta_2 a_2}{A_2} \sqrt{2gh_2(t)} + \frac{\beta_4 a_4}{A_2} \sqrt{2gh_4(t)} + \frac{\gamma_2 k_2}{A_2} u_2(t) \\
\frac{dh_3(t)}{dt} &= -\frac{\beta_3 a_3}{A_3} \sqrt{2gh_3(t)} + \frac{(1-\gamma_2)k_2}{A_3} u_2(t) \\
\frac{dh_4(t)}{dt} &= -\frac{\beta_4 a_4}{A_4} \sqrt{2gh_4(t)} + \frac{(1-\gamma_1)k_1}{A_4} u_1(t)
\end{aligned} \tag{4.1}$$

โดย

$h_i(t)$	คือ ระดับน้ำในถัง i	$i = 1 \sim 4$; cm
a_i	คือ พื้นที่หน้าตัดของรูน้ำด้านนอกของถัง i	$i = 1 \sim 4$; cm ²
A_i	คือ พื้นที่หน้าตัดของถัง i	$i = 1 \sim 4$; cm ²
β_i	คือ ค่าตำแหน่งวาล์วที่ต่อด้านออกจากถัง i	$i = 1 \sim 4$	
γ_i	คือ ค่าตำแหน่งวาล์วที่ต่ออยู่กับปั๊ม i	$i = 1, 2$	
k_i	คือ ค่าอัตราการขยายของปั๊ม i	$i = 1, 2$; cm ³ /V.s
g	คือ แรงโน้มถ่วงของโลก		981 cm/s ²
u_i	คือ แรงดันที่จ่ายให้กับปั๊ม i	$i = 1, 2$; V

อัตราการไหลของน้ำที่ไหลเข้าถังที่ 1 จะเป็นสัดส่วนกับ $\gamma_1 k_1 u_1$ และอัตราการไหลของน้ำที่ไหลเข้าถังที่ 4 จะเป็นสัดส่วนกับ $(1-\gamma_1)k_1 u_1$ และในทำนองเดียวกันสำหรับถังที่ 2 และถังที่ 3 สัญญาณอินพุตหรือสัญญาณควบคุมที่ป้อนเข้าสู่กระบวนการคือ u_1 และ u_2 (ค่าสัญญาณแรงดันที่จ่ายให้กับปั๊มน้ำ) และสัญญาณเอาต์พุตที่วัดจากกระบวนการคือ y_1 และ y_2 (ค่าสัญญาณแรงดันจากอุปกรณ์วัด) จะถูกปรับให้เป็นสัดส่วนเดียวกันกับความสูงของระดับน้ำที่วัดได้ คือ $y_1 = k_{m1} h_1$ และ $y_2 = k_{m2} h_2$ โดย $k_{m1} = k_{m2} = 1$

จากสมการ (4.1) สามารถทำให้เป็นแบบจำลองเชิงเส้นในรูปแบบของสมการสเตท-สเปซได้ โดยกำหนดให้ $x_i = h_i - h_i^0$ และ $u_i = u_i - u_i^0$

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \end{bmatrix} = \begin{bmatrix} -\frac{\beta_1}{T_1} & 0 & \frac{\beta_3 A_3}{A_3 T_3} & 0 \\ 0 & -\frac{\beta_2}{T_2} & 0 & \frac{\beta_4 A_4}{A_2 T_4} \\ 0 & 0 & -\frac{\beta_3}{T_3} & 0 \\ 0 & 0 & 0 & -\frac{\beta_4}{T_4} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} \frac{\gamma_1 k_1}{A_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{A_2} \\ 0 & \frac{(1-\gamma_2)k_2}{A_3} \\ \frac{(1-\gamma_1)k_1}{A_4} & 0 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} \tag{4.2}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix}$$

โดยค่าคงตัวเวลาสำหรับแบบจำลองเชิงเส้นของกระบวนการนี้คือ

$$T_i = \frac{A_i}{\beta_i a_i} \sqrt{\frac{2h_i^0}{g}} \quad (4.3)$$

และเมทริกฟังก์ชันถ่ายโอนของแบบจำลองเชิงเส้นนี้คือ

$$G_p(s) = \begin{bmatrix} \frac{\gamma_1 k_1 c_1}{(T_1 s + 1)} & \frac{(1 - \gamma_2) k_2 c_1}{(T_1 s + 1)(T_3 s + 1)} \\ \frac{(1 - \gamma_1) k_1 c_2}{(T_2 s + 1)(T_4 s + 1)} & \frac{\gamma_2 k_2 c_2}{(T_2 s + 1)} \end{bmatrix} \quad (4.4)$$

โดย $C_1 = \frac{T_1}{A_1}$ และ $C_2 = \frac{T_2}{A_2}$ และสมการที่ (4.4) จะสมมูลกับเมทริกฟังก์ชันถ่ายโอนดังนี้

$$G_p(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \quad (4.5)$$

จากสมการ (4.4) จะสังเกตเห็นได้ว่า ฟังก์ชันถ่ายโอนแต่ละคู่อินพุต-เอาต์พุตนั้นไม่มีซีโรซึ่งการพิจารณาแบบนี้เป็นการพิจารณาเพียงส่วนย่อยของระบบเท่านั้น โดยสำหรับกระบวนการแบบ MIMO นั้น การพิจารณาโพลและซีโรของกระบวนการทั้งหมด จะแตกต่างจากระบบ SISO ซึ่งวิธีในการหาจะอธิบายใน ส่วนของคุณลักษณะของกระบวนการ

4.2 การวิเคราะห์คุณลักษณะของกระบวนการระดับน้ำ 4 ถัง

4.2.1 ซีโรของกระบวนการ MIMO

การหาตำแหน่งและทิศทางซีโรสำหรับกระบวนการแบบหลายอินพุต-เอาต์พุต MIMO นั้น เป็นสิ่งสำคัญมาก เนื่องจากซีโรของกระบวนการนั้นสามารถเกิดขึ้นได้แม้ว่าฟังก์ชันถ่ายโอนย่อยของกระบวนการแต่ละอินพุต-เอาต์พุตไม่มีซีโรอยู่เลยก็ตาม กระบวนการระดับน้ำ 4 ถังก็เป็นกระบวนการหนึ่งที่ฟังก์ชันถ่ายโอนย่อยในแต่ละคู่อินพุต-เอาต์พุตไม่มีซีโร แต่เมื่อวิเคราะห์ทั้งระบบแล้วนั้นจะพบว่ากระบวนการนี้มีซีโรซ่อนอยู่ ซึ่งซีโรของกระบวนการสามารถอยู่ได้ทั้งด้านซ้ายหรือด้านขวาของระนาบ s ก็ได้โดยการปรับตำแหน่งขั้ว γ_1 และ γ_2 นอกจากนี้แล้วผลจากการปรับค่าตำแหน่งขั้ว γ_1 และ γ_2 ยังมี

ผลทำให้ทิศทางของซีโรเปลี่ยนไป ซึ่งทิศทางของซีโรนี้จะเป็นตัวกำหนดถึงเอาต์พุตที่จะได้รับผลกระทบจากซีโรตัวนั้น

ในส่วนนี้จะอธิบายถึงวิธีการหาตำแหน่ง, ทิศทางของซีโรของกระบวนการระดับน้ำ 4 ถึง ซึ่งสามารถอธิบายได้ในรูปแบบของค่าตำแหน่งวาล์ว γ_1 และ γ_2 และอธิบายถึงผลกระทบจากตำแหน่งและทิศทางของซีโรต่อกระบวนการ

4.2.2 การหาตำแหน่งซีโร

ซีโรของเมทริกฟังก์ชันถ่ายโอนในสมการ (3.4) นั้นก็คือเทอมนเศษ (Numerator) ของสมการพหุนามซึ่งหาได้จากการทำดีเทอร์มิแนนต์เมทริกฟังก์ชันถ่ายโอน (4.4) นั้นเอง

$$\det G_p(s) = \frac{\gamma_1 \gamma_2 k_1 k_2 c_1 c_2}{\prod_{i=1}^4 (T_i s + 1)} \left[(T_3 s + 1)(T_4 s + 1) - \frac{(1 - \gamma_1)(1 - \gamma_2)}{\gamma_1 \gamma_2} \right] \quad (4.6)$$

ดังนั้นเมทริกฟังก์ชันถ่ายโอน $G_p(s)$ จะมีซีโรอยู่ 2 ตัว โดยซีโรตัวหนึ่งจะอยู่ด้านซ้ายของระนาบ s เสมอในขณะที่ซีโรอีกตัวหนึ่งนั้นสามารถอยู่ได้ทั้งด้านซ้ายหรือด้านขวาของระนาบ s กำหนดให้ $\eta \in (0, \infty)$

$$\eta = \frac{(1 - \gamma_1)(1 - \gamma_2)}{\gamma_1 \gamma_2} \quad (4.7)$$

ถ้า η มีค่าน้อย ($\eta \rightarrow 0$) ซีโรทั้ง 2 ตัวจะเข้าใกล้ตำแหน่ง $-1/T_3$ และ $-1/T_4$ ในขณะที่ η มีค่ามาก ($\eta \rightarrow \infty$) ซีโรตัวหนึ่งจะเข้าใกล้ $-\infty$ และซีโรอีกตัวหนึ่งจะเข้าใกล้ $+\infty$ ถ้า $\eta = 1$ จะมีซีโรตัวหนึ่งอยู่ที่จุดศูนย์กลางของระนาบ s ซึ่งสอดคล้องกับ $\gamma_1 + \gamma_2 = 1$ จากสมการ (3.6) เห็นได้ว่าระบบจะเป็นแบบเฟสไม่ต่ำสุด สำหรับในกรณีนี้ $0 < \gamma_1 + \gamma_2 < 1$ และเป็นแบบเฟสต่ำสุดสำหรับในกรณีที่ $1 < \gamma_1 + \gamma_2 < 2$

การที่ซีโรของกระบวนการนี้ตัวหนึ่งอยู่ได้ทั้งด้านซ้ายและด้านขวาของระนาบ s สามารถอธิบายได้โดยคุณสมบัติทางกายภาพของกระบวนการคือ กำหนดให้ q_1 คืออัตราการไหลจากบ่มี i และสมมติให้อัตราการไหล $q_1 = q_2$ เมื่อนั้นผลรวมของปริมาณน้ำทั้งหมดที่ไหลเข้าสู่ถึงด้านบนคือ $[2 - (\gamma_1 + \gamma_2)]q_1$ และผลรวมของปริมาณน้ำที่ไหลเข้าสู่ถึงด้านล่างคือ $(\gamma_1 + \gamma_2)q_1$ ดังนั้นปริมาณน้ำที่ไหลเข้าสู่ถึงด้านล่างจะมากกว่าถึงด้านบนก็ต่อเมื่อ $(\gamma_1 + \gamma_2) > 1$ ซึ่งสอดคล้องกับการที่กระบวนการมีคุณสมบัติเป็นแบบเฟสต่ำสุด และถ้าปริมาณน้ำที่ไหลเข้าสู่ถึงด้านบนมากกว่าถึงด้านล่างกระบวนการก็จะมีคุณลักษณะเป็นแบบเฟสไม่ต่ำสุด จากที่ได้ทำการอธิบายมาสามารถสังเกตเห็นได้ว่า ถ้าปริมาณน้ำไหลเข้าสู่ถึงด้านล่างมากกว่าถึงด้านบน y_1 และ y_2 ควรที่จะถูกควบคุมด้วย u_1 และ u_2 ในทางกลับกันปริมาณน้ำที่ไหลเข้าสู่ถึงด้านบนมากกว่าถึงด้านล่าง y_1 และ y_2 ก็ควรที่จะถูกควบคุมด้วย u_2 และ u_1

4.2.3 ผลกระทบจากตำแหน่งซีโรต่อระบบควบคุม

ดังได้อธิบายในตอนที่แล้วว่า กระบวนการนี้สามารถเลือกรูปแบบการทำงานให้เป็นแบบเฟสต่ำสุดหรือเฟสไม่ต่ำสุดได้ โดยผลกระทบจากตำแหน่งซีโรต่อระบบควบคุมในกรณีเฟสต่ำสุดและเฟสไม่ต่ำสุดจะแตกต่างกัน ซึ่งสามารถอธิบายได้ดังนี้

กรณีเฟสต่ำสุด

ในกรณีนี้ซีโรของกระบวนการทั้ง 2 ตัว จะอยู่ด้านซ้ายของระนาบ s โดยเป็นที่ทราบกันดีอยู่แล้วว่า ตำแหน่งซีโรสำหรับกระบวนการ SISO นั้นจะมีผลกระทบต่อผลตอบสนองของระบบควบคุมในรูปของค่าพุงเกินและขีดจำกัด ของอินพุต อันเนื่องมาจากค่าอัตราขยายที่สถานะคงตัวน้อยซึ่งสอดคล้องกับการที่ซีโรอยู่ใกล้จุดศูนย์กลางของระนาบ s มาก แต่สำหรับกระบวนการ MIMO เช่นกระบวนการ 2 อินพุต 2 เอาต์พุต ผลกระทบจากตำแหน่งซีโรต่อระบบควบคุมก็จะคล้ายกับกระบวนการ SISO แตกต่างกันตรงที่ตำแหน่งของซีโรนั้นสามารถมีผลกระทบกับเอาต์พุตใดเอาต์พุตหนึ่ง หรืออาจจะมีผลกระทบต่อทั้งสองเอาต์พุตก็ได้ขึ้นอยู่กับทิศทางของซีโร

กรณีเฟสไม่ต่ำสุด

ในกรณีนี้ซีโรของกระบวนการตัวหนึ่งจะมีตำแหน่งอยู่ด้านซ้ายของระนาบ s และซีโรอีกตัวหนึ่งจะมีตำแหน่งอยู่ทางด้านขวาของระนาบ s โดยผลกระทบจากการที่ซีโรมีตำแหน่งอยู่ทางด้านซ้ายของระนาบ s ก็จะเป็นเช่นเดียวกับที่ได้อธิบายในกรณีเฟสต่ำสุด แต่สำหรับซีโรที่มีตำแหน่งอยู่ทางด้านขวาของระนาบ s จะมีผลที่แตกต่างกันออกไป คือซีโรที่มีตำแหน่งอยู่ทางด้านขวาของระนาบ s จะเป็นตัวจำกัดแบนด์วิดท์ของระบบควบคุม ซึ่งสอดคล้องกับความเร็วผลตอบสนองของระบบควบคุมนั่นเอง โดยแบนด์วิดท์ของระบบควบคุมในกรณีที่มีซีโรอยู่ทางด้านขวาของทางระนาบ s คือ

$$\omega_b < \frac{z}{2} \quad (4.8)$$

สมการ (4.8) สังเกตได้ว่า ถ้าซีโรอยู่ใกล้จุดศูนย์กลางของระนาบ s แบนด์วิดท์ของระบบควบคุมจะแคบ ในทางกลับกันถ้าซีโรอยู่ห่างจากจุดศูนย์กลางของระนาบ s แบนด์วิดท์ของระบบควบคุมจะกว้าง

จากที่ทำการอธิบายมาแสดงให้เห็นได้ว่าการบวนการระดับน้ำ 4 ดังนั้นเป็นกระบวนการที่เหมาะสมสำหรับการศึกษาถึงพฤติกรรม และการอธิบายถึงขีดจำกัดทางด้านสมรรถนะในการออกแบบตัวควบคุมของระบบควบคุมแบบหลายอินพุต-หลายเอาต์พุต MIMO อันเนื่องมาจากตำแหน่งซีโรที่อยู่ทางด้านขวาของระนาบ s ซึ่งจากแบบจำลองเชิงเส้นของกระบวนการระดับน้ำ 4 ถึง พบว่าตำแหน่งซีโรของกระบวนการนั้นจะขึ้นอยู่กับค่าตำแหน่งของพหุคูณ $\gamma_1, \gamma_2 \in (0, 1)$ โดยถ้า $\gamma_1 + \gamma_2 < 1$ กระบวนการจะเป็นแบบเฟสไม่ต่ำสุดและถ้า $\gamma_1 + \gamma_2 > 1$ กระบวนการจะเป็นแบบเฟสต่ำสุด ทำให้สามารถเลือกคุณลักษณะในการทำงานของกระบวนการได้

บทที่ 5

ความรู้พื้นฐานการประยุกต์ใช้งานด้านซอฟต์แวร์

5.1 การใช้งานโปรแกรม Matlab

สำหรับโปรแกรมที่ใช้ในงานวิจัยนี้คือ โปรแกรม MatLab ซึ่งสามารถใช้ควบคุมได้เป็นกระบวนการได้อย่างดี อีกทั้งยังสามารถทำงานร่วมกับการ์ด ADC และ DAC ของบริษัท Advantech รุ่น PCI-1711 ได้ โปรแกรม Matlab จะถูกใช้ในการจำลองกระบวนการ สำหรับเครื่องมือหรือโปรแกรมย่อยของโปรแกรม MatLab นั้นมีอยู่มาก ผู้วิจัยจึงขออธิบายถึงส่วนประกอบของโปรแกรมที่ใช้เฉพาะในงานวิจัยนี้เท่านั้น ซึ่งรายละเอียดแสดงได้ดังต่อไปนี้

5.1.1 โปรแกรม Simulink

Simulink เป็นโปรแกรมที่เป็นส่วนที่เพิ่มเติมเข้ามาในโปรแกรม MATLAB ซึ่งเป็นโปรแกรมที่พัฒนาโดยบริษัท Mathworks โดยเป็นโปรแกรมที่มีความสามารถในการแก้ปัญหาทางคณิตศาสตร์ที่มีประสิทธิภาพสูง ในส่วนของ Simulink นั้นเป็นเครื่องมือที่ใช้ในการเขียนโปรแกรมเพื่อหาคำตอบของสมการทางคณิตศาสตร์โดยใช้ลักษณะการโปรแกรมในรูปแบบของ Block Diagram เป็นหลัก โดยเฉพาะอย่างยิ่งระบบสมการเชิงอนุพันธ์ซึ่งมักจะพบปัญหาในการหา Closed Form Solution หรือการหาคำตอบเมื่อเทียบกับเวลาในช่วงเวลาใดๆ และเนื่องจากระบบทางวิศวกรรมโดยส่วนใหญ่จะสามารถอธิบายด้วยระบบสมการเชิงอนุพันธ์ เช่นกันดังนั้น Simulink จะเข้ามามีบทบาทในการทำให้การจำลองการตอบสนองของระบบมีความสะดวกมากยิ่งขึ้น

สำหรับองค์ประกอบของโปรแกรมนั้นไม่ว่าจะเป็นวงจรทางตรรกศาสตร์ไปจนถึง ระบบควบคุมสำหรับระบบพลศาสตร์ที่ซับซ้อนจะสามารถสร้างแบบจำลองได้โดยอาศัยเพียง Building Blocks พื้นฐานที่มีมาให้จาก Simulink Libraries นอกเหนือจากนั้นยังมีเครื่องมือใน ส่วนอื่นๆ ที่ในปัจจุบันได้เพิ่มเติมเข้ามาเพื่อให้การใช้งานมีประสิทธิภาพมากยิ่งขึ้น เช่น Fuzzy Logic Toolbox, Neural Network Toolbox, Digital Signal Processing (DSP) Toolbox หรือ อื่นๆ ซึ่งในปัจจุบันพบว่าเครื่องมือต่างๆ ที่มีให้ สามารถทำให้การจำลองหรือวิเคราะห์การทำงานของ ระบบสามารถทำได้ตั้งแต่ระบบที่มีความง่ายจนถึงระบบที่มีความซับซ้อนอย่างมากโดยไม่ต้อง มีการเขียนโปรแกรมเพิ่มเติมแต่อย่างใด

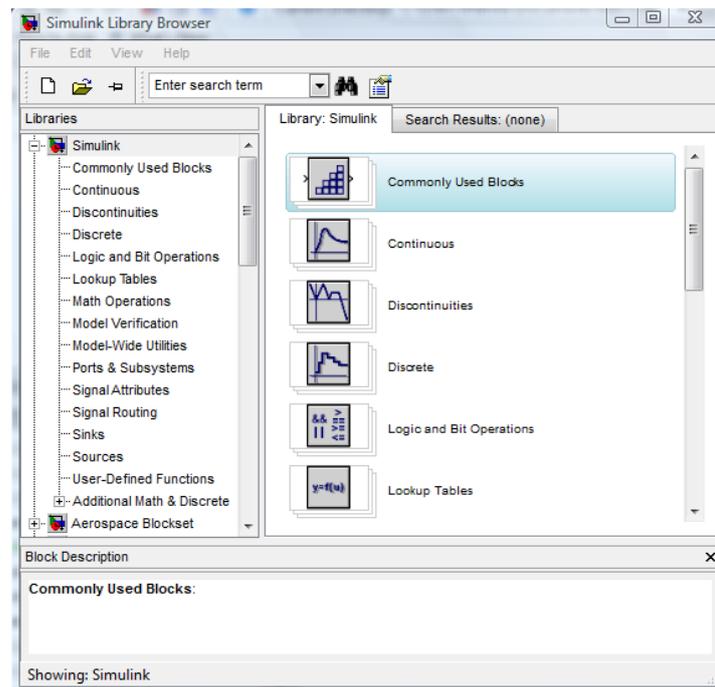
5.1.1.1 การจำลองระบบพลศาสตร์ (Dynamic System Simulation) ด้วย Simulink

ในปัจจุบันคอมพิวเตอร์ได้เข้ามามีส่วนช่วยทำให้การแก้ปัญหาทางวิศวกรรมได้ใช้ประโยชน์จากคณิตศาสตร์อย่างเต็มรูปแบบในการจำลองระบบทางพลศาสตร์ใดๆ โดยไม่ต้องอาศัย การศึกษาหรือทดลองจากระบบทางกายภาพจริงโดยการจำลองการทำงานของระบบพลศาสตร์ (Dynamic System Simulation) ดังกล่าวนั้น ได้รับการพิสูจน์แล้วว่ามีความเหมาะสมอย่างยิ่งในการทำงานทาง วิศวกรรม โดยเฉพาะอย่างยิ่งในงานทางด้านการออกแบบระบบ (System Design) และงานการออกแบบ ระบบควบคุม (Controller System Design) ทั้งนี้เนื่องจากการจำลองปัญหาช่วยประหยัดทั้งงบประมาณ และเวลา (Prototype) ของระบบทางกายภาพที่สนใจซึ่งในอดีตจะต้องสูญเสียให้กับการสร้างระบบต้นแบบ

5.1.1.2 หลักการของสัญญาณและการไหลของตรรก (Signal and Logic Flow)

ในโปรแกรม Simulink นั้นการจำลองปัญหาจะอาศัยการสร้างแผนภูมิ (Simulation Diagram) เพื่อการกำเนิด การรับ หรือส่งผ่านสัญญาณ หรือข้อมูลที่ต้องการทราบ ใน แผนภูมิที่ถูกสร้างขึ้นข้อมูล จากบล็อกประเภทต่างๆ จะถูกส่งผ่านไปสู่อบล็อกอื่นๆ โดยเส้นสัญญาณที่

เชื่อมต่อระหว่าง บล็อกนั้นๆ สัญญาณสามารถที่จะถูกสร้างและส่งไปสู่บล็อกอื่นๆ ทั้งแบบสัณยต์และแบบแปรผันข้อมูลสามารถป้อนให้แก่ฟังก์ชันเพื่อการกระทำทางตรรกศาสตร์หรือทางคณิตศาสตร์ ข้อมูลสามารถที่จะถูกส่งไปสู่แหล่งทิ้งข้อมูล (Sink) ซึ่งสามารถที่จะแสดงให้เห็นในลักษณะของ Oscilloscope หรือ Display หรือแม้กระทั่งเก็บในรูปของไฟล์ (Save to Files) สิ่งที่ต้องพึงระวังในการจำลองปัญหานั้น ข้อมูลต่างๆ จะถูก ดำเนินการ รับและส่ง ในลักษณะเป็นช่วงเวลา (Discrete Times) เท่านั้น ดังนั้น Step Time หรือ Simulation Time Step หรือ Integration Time Step ที่กำหนดเพื่อการจำลองปัญหานั้นจะเป็นส่วนที่สำคัญอย่างยิ่งโดยการเลือกใช้ Step Time ที่เหมาะสมจะถูกกำหนดโดยระบบย่อยที่มีการตอบสนองเร็วที่สุดในแผนภูมิที่กำหนดขึ้น



ภาพที่ 5.1 ลักษณะของโปรแกรม Simulink

ในโปรแกรม Simulink นั้นการจำลองปัญหาจะอาศัยการสร้างแผนภูมิ (Simulation Diagram) เพื่อการกำเนิด การรับ หรือส่งผ่านสัญญาณ หรือข้อมูลที่ต้องการทราบ ในแผนภูมิที่ถูกสร้างขึ้นข้อมูล จากบล็อกประเภทต่างๆ จะถูกส่งผ่านไปสู่บล็อกอื่นๆ โดยเส้นสัญญาณที่

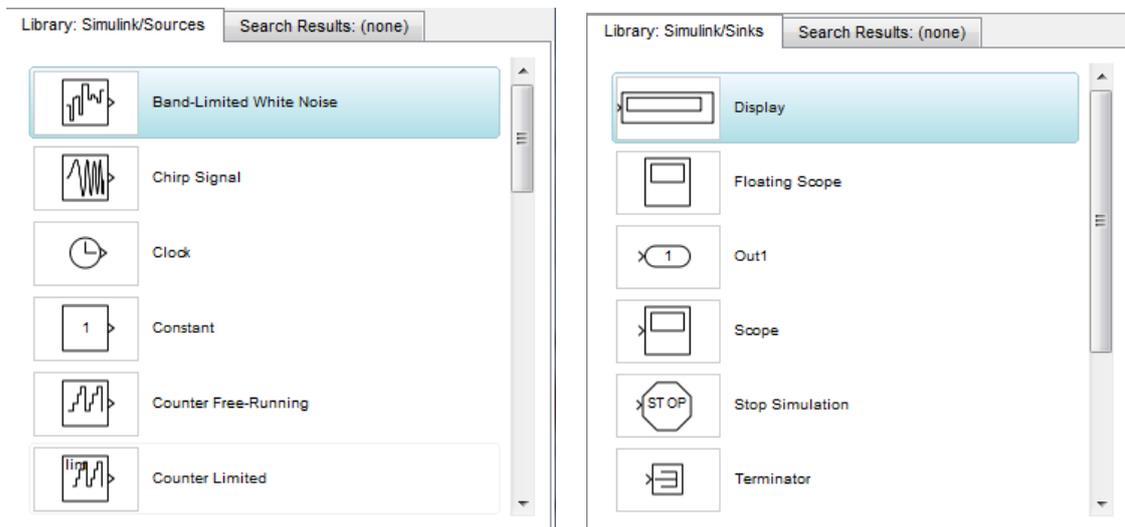
5.1.1.3 การเชื่อมต่อบล็อก

ในการเชื่อมต่อบล็อกบน Simulation Diagram นั้น ให้กดเมาส์ปุ่มซ้ายที่ Output ของ บล็อกหนึ่งแล้วค้างปุ่มเอาไว้โดยเคอร์เซอร์ของเมาส์จะเปลี่ยนรูปเป็นเครื่องหมายบวก หลังจากนั้นลาก (Drag) เมาส์ (จะเห็นเส้นประเกิดขึ้นและวิ่งตามเครื่องหมายบวกดังกล่าว) จากบล็อกหนึ่งไปสู่ Input ของบล็อกที่ต้องการจะเชื่อมต่อ เมื่อปล่อยเมาส์ปุ่มซ้ายที่ Input ของอีกบล็อกหนึ่งแล้วเส้นประจะเปลี่ยนเป็นเส้นทึบแสดงถึงการเชื่อมต่อกันของบล็อกทั้งสอง

5.1.1.4 กลุ่มของบล็อกที่ใช้งานบ่อยในเบื้องต้น

1) แหล่งกำเนิดสัญญาณ (Sources) และแหล่งทิ้งสัญญาณ (Sinks)

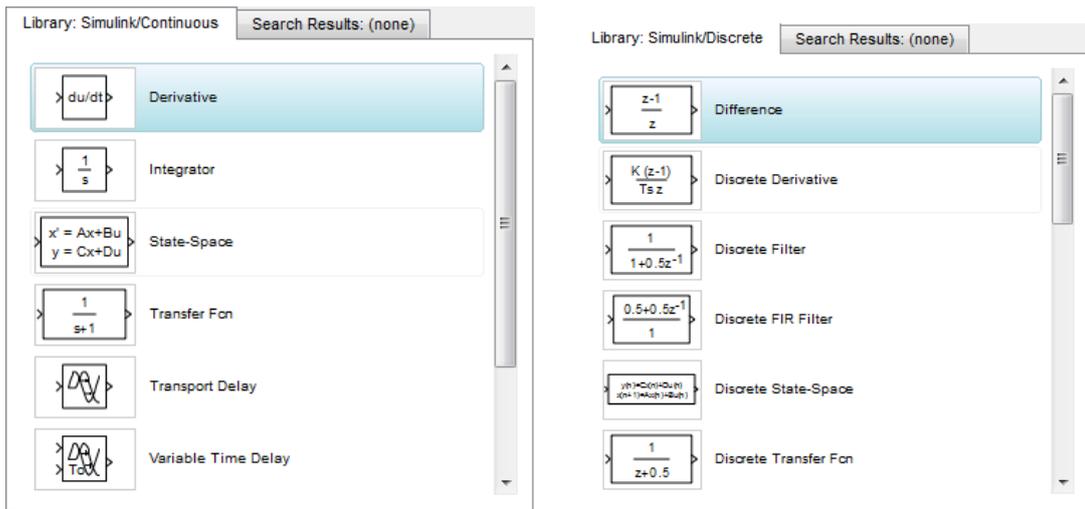
กลุ่มบล็อกแหล่งกำเนิดสัญญาณ (Sources) ที่กำหนดมาให้ประกอบด้วยแหล่งกำเนิดข้อมูลหรือสัญญาณ ที่ใช้บ่อยในการวิเคราะห์ระบบทางพลศาสตร์ เช่น แหล่งกำเนิดค่าคงที่ (Constant) แหล่งกำเนิดสัญญาณคลื่นไซน์ (Sinusoidal Wave) แหล่งกำเนิดสัญญาณแบบ Step หรืออื่นๆ นอกจากนี้ยังมีแหล่งกำเนิดสัญญาณที่ทำหน้าที่เหมือน Signal Generator รูปที่ใช้ในการทดลองทางวิศวกรรมอื่นๆ หรือแหล่งกำเนิดสัญญาณนาฬิกา (Clock) ที่มักจะใช้บ่อยเพื่อจับเวลาของการจำลองปัญหา หรืออื่นๆ กลุ่มบล็อกแหล่งทิ้งสัญญาณ (Sinks) เป็นกลุ่มบล็อกที่มักจะใช้ในการเก็บสัญญาณที่ ต้องการมากกว่าการทิ้งสัญญาณอย่างที่ชื่อกำหนด โดยทั่วไปแล้วบล็อกที่จะใช้บ่อยคือบล็อกของ Scope ประเภทต่างๆ เพื่อตรวจสอบลักษณะการเปลี่ยนแปลงของสัญญาณในจุดใดๆ ของระบบที่กำลังถูกจำลองในขณะที่กำลังทำการจำลอง หรือบล็อกที่เก็บสัญญาณ (To Workspace) เพื่อใช้ในการดำเนินการอื่นๆ ที่พื้นที่ทำงานของ MATLAB ภาพที่ 5.2 แสดงบล็อกต่างๆ ของทั้งแหล่งกำเนิด และแหล่งสัญญาณ ของ Simulink



ภาพที่ 5.2 แสดงบล็อกต่างๆ ของแหล่งกำเนิดสัญญาณ (Sources) และแหล่งสัญญาณ (Sinks)

2) ระบบต่อเนื่อง (Continuous) และระบบไม่ต่อเนื่อง (Discrete)

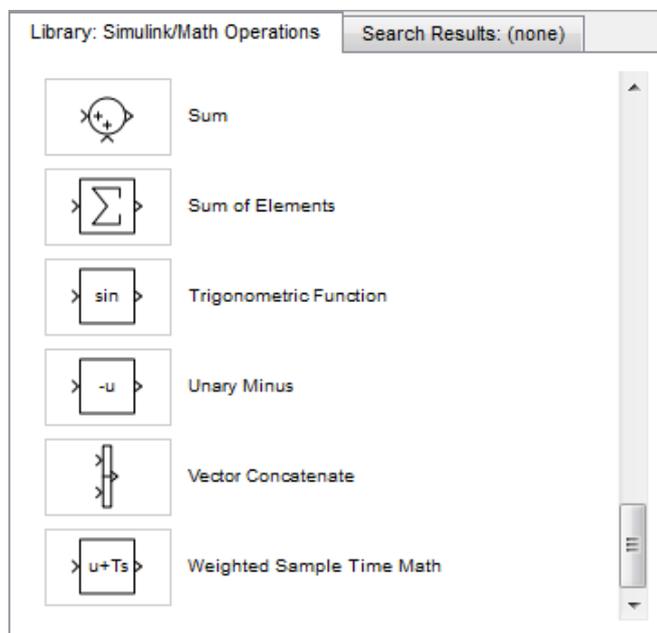
กลุ่มบล็อกของระบบต่อเนื่อง (Continuous System) หรือระบบไม่ต่อเนื่อง (Discrete System) เป็นอีกกลุ่มหนึ่งที่มีการใช้ในการจำลองปัญหาบ่อย ระบบพลศาสตร์ทุกระบบสามารถที่จะถูกวิเคราะห์ในลักษณะของระบบต่อเนื่อง (Continuous System) หรือระบบไม่ต่อเนื่อง (Discrete System) ซึ่งไม่ว่าระบบจะถูกจำลองในลักษณะใด โปรแกรม Simulink นั้นสามารถตอบสนองต่อลักษณะรูปแบบของระบบได้ในหลายลักษณะ เช่น Transfer Function หรือ Integration Blocks หรือ Delay Blocks หรืออื่นๆ ภาพที่ 5.3 แสดงลักษณะบล็อกที่ใช้ในการกำหนด ระบบในลักษณะต่างๆ ของทั้งระบบต่อเนื่อง (Continuous System) หรือระบบไม่ต่อเนื่อง (Discrete System) ของ Simulink



ภาพที่ 5.3 แสดงบล็อกต่างๆ ภายในกลุ่มบล็อกของระบบต่อเนื่อง (Continuous System) และระบบไม่ต่อเนื่อง (Discrete System)

3) การดำเนินการทางคณิตศาสตร์ (Math Operations)

กลุ่มบล็อกการดำเนินการทางคณิตศาสตร์เป็นอีกกลุ่มหนึ่งที่มีการใช้ในการจำลองปัญหาบ่อย กลุ่มบล็อกนี้จะประกอบไปด้วยบล็อกย่อยๆ ที่มีหน้าที่เกี่ยวข้องกับการดำเนินการต่างๆ ทางคณิตศาสตร์ เช่น การนำสัญญาณมาบวกกัน (Sum) การคูณสัญญาณด้วยค่าคงที่ (Gain) หรืออื่นๆ ภาพที่ 5.4 แสดงลักษณะบล็อกย่อยในกลุ่มบล็อกการดำเนินการทางคณิตศาสตร์ ของ Simulink



ภาพที่ 5.4 แสดงบล็อกต่างๆ ภายในกลุ่มบล็อกการดำเนินการทางคณิตศาสตร์ (Math Operations)

4) ความไม่ต่อเนื่อง (Discontinuities)

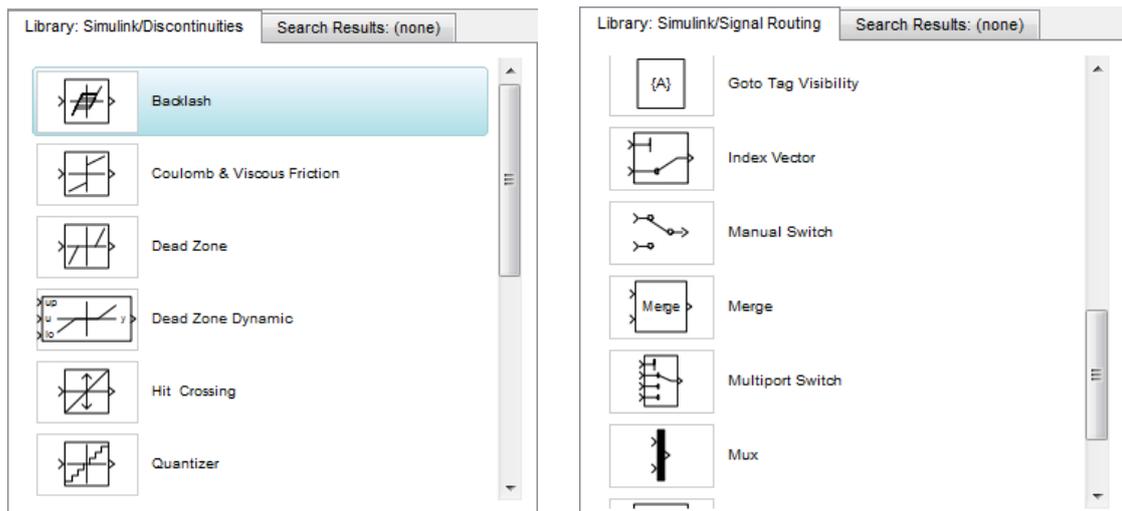
ข้อได้เปรียบข้อหนึ่งของโปรแกรม Simulink คือความสามารถในการจำลองระบบแบบไม่เป็นเชิงเส้น (Nonlinear System Simulation) ซึ่งในการหาคำตอบด้วยมือ หรือการหา Closed Form Solution โดยทั่วไปจะเป็นการยากฟังก์ชันหรือลักษณะของความไม่เป็นเชิงเส้นที่พบบ่อยคือ Saturation, Signum, Limited Slew Rate หรืออื่นๆ ภาพที่ 5.4 แสดงกลุ่มบล็อก Discontinuities ซึ่ง แสดงความไม่เป็นเชิงเส้นซึ่งสามารถใช้ในการจำลองปรากฏการณ์ทางวิศวกรรมได้หลายเหตุการณ์ ยกตัวอย่างเช่นการใช้ Saturation บล็อกมักจะใช้แทนข้อจำกัดทางกายภาพเช่นขีดจำกัดของแรงดันทางเข้ามอเตอร์ไฟฟ้าหรือระดับแรงดันสูงสุดในระบบไฮดรอลิกส์ บล็อก Switches เป็นอีกบล็อกหนึ่งที่ใช้บ่อย เป็นบล็อกที่ใช้ในการจำลองปัญหา IF-THEN เช่นเดียวกับที่พบในการเขียน โปรแกรม โดยทั่วไป บล็อก Slew Rates ใช้ในการจำกัดอัตราการเปลี่ยนแปลงของค่าทางกายภาพ เช่น ความเร็วของมอเตอร์ไฟฟ้ากระแสตรง เป็นต้น

5) ทางเดินสัญญาณ (Signal Routing)

ในการจำลองระบบที่มีความซับซ้อน บางครั้งมีความจำเป็นที่จะต้องมีการรับ/ส่ง สัญญาณจากส่วนหนึ่งของระบบ ไปสู่อีกส่วนหนึ่งของระบบ ซึ่งการใช้เส้นเชื่อมอาจจะไม่สะดวกในการทำงานซึ่งบล็อกในชุดนี้เป็นประเภท GOTO บล็อกหรือ FROM บล็อกกับบล็อกประเภท Multiplexing จะมีประโยชน์ในการจัดระเบียบของแผนภูมิให้มีความเป็นระเบียบมากขึ้นภาพที่ 5.5 แสดงกลุ่มบล็อก Signal Routing

6) การกำหนดค่าคงที่ในการจำลองปัญหา (Parameters Configuration)

การจำลองปัญหาบนคอมพิวเตอร์จะต้องอาศัยเทคนิคของระเบียบวิธีเชิงตัวเลข(Numerical Method) ในการแก้ปัญหาและหาคำตอบของสมการเชิงอนุพันธ์ สำหรับโปรแกรม Simulink นั้น ระบบในรูปแบบต่างๆ สามารถที่จะจำลองทั้งในแบบของระบบต่อเนื่อง (Continuous System) หรือระบบไม่ต่อเนื่อง (Discrete System) ขึ้นอยู่กับลักษณะองค์ประกอบของระบบย่อย การจำลองสามารถที่จะถูกกำหนดให้เริ่ม หรือหยุดที่ช่วงเวลา หรือส่วนหนึ่งของเวลาได้ตามกำหนด ข้อสังเกตที่ต้องพึงระวังคือระยะเวลาที่สนใจในการจำลองปัญหา กับระยะเวลาที่ใช้ ในการจำลอง ปัญหา หรือระยะเวลาที่คอมพิวเตอร์ใช้ในการหาคำตอบของปัญหาที่สนใจจะไม่ตรงกัน เช่นคอมพิวเตอร์อาจจะใช้ระยะเวลาในการจำลองการตอบสนองในช่วง 10 นาที ของระบบที่สนใจ เพียง 1 วินาที เป็นต้น ในการจำลองแบบปัญหาโดยใช้ระเบียบวิธีเชิงตัวเลขประเภทที่ Step Size มีค่าไม่คงที่นั้น (Variable Step Size) Step Size ที่มากที่สุดและน้อยที่สุดสามารถกำหนดได้ อย่างไรก็ตามระเบียบวิธีเชิงตัวเลขประเภท Step Size คงที่ (Fixed Step Size) เป็นวิธีที่มักจะนิยมใช้ในการจำลองปัญหา ทั้งนี้เนื่องจากสามารถระบุช่วงเวลา หรือจุดของเวลาที่สนใจได้อย่างแม่นยำ Step Size ที่ใช้ในการจำลองปัญหาจะต้องมีการกำหนด และจะมีค่าขึ้นอยู่กับลักษณะของระบบพลศาสตร์ที่สนใจ เช่น ในระบบทางความร้อน Step Size อาจจะมีอยู่ประมาณ 2 – 3 วินาที ในขณะที่ระบบควบคุมมอเตอร์อาจจำเป็นที่จะต้องใช้ Step Size ที่มีค่าน้อยถึง 0.002 – 0.003 วินาที โดยทั่วไป กฎที่ใช้ในการเลือก Step Size จะอยู่ที่ประมาณอย่างน้อย 4 – 20 เท่าของระบบที่มีการตอบสนองที่เร็วที่สุด



ภาพที่ 5.5 แสดงกลุ่มบล็อก Discontinuities และกลุ่ม Signal Routing

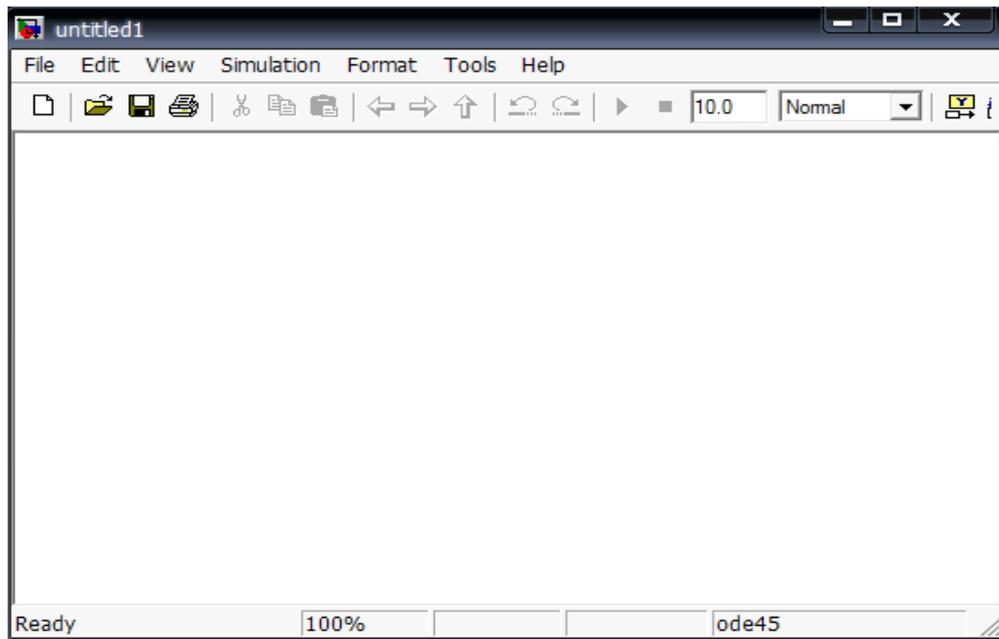


ภาพที่ 5.6 แสดงหน้าต่างที่ใช้ในการกำหนดค่าคงที่ในการจำลองปัญหา (Configuration Parameters)

5.1.1.5 การสร้างแบบจำลองระบบควบคุม

การสร้างแบบจำลองของระบบควบคุมใน Simulink นั้นต้องทำการสร้างหน้าต่าง สำหรับสร้างแบบจำลองขึ้นมา ก่อนซึ่งขั้นตอนในการสร้างแบบจำลองระบบควบคุมทำได้ดังนี้

1) เลือก เมนู File --> New --> Model จากนั้นจะปรากฏหน้าต่างขึ้นมา แสดงได้ดังภาพที่ 5.7 ซึ่งหน้าต่างนี้จะใช้สำหรับการวางบล็อกที่เลือกมาจาก Simulink Library Browser



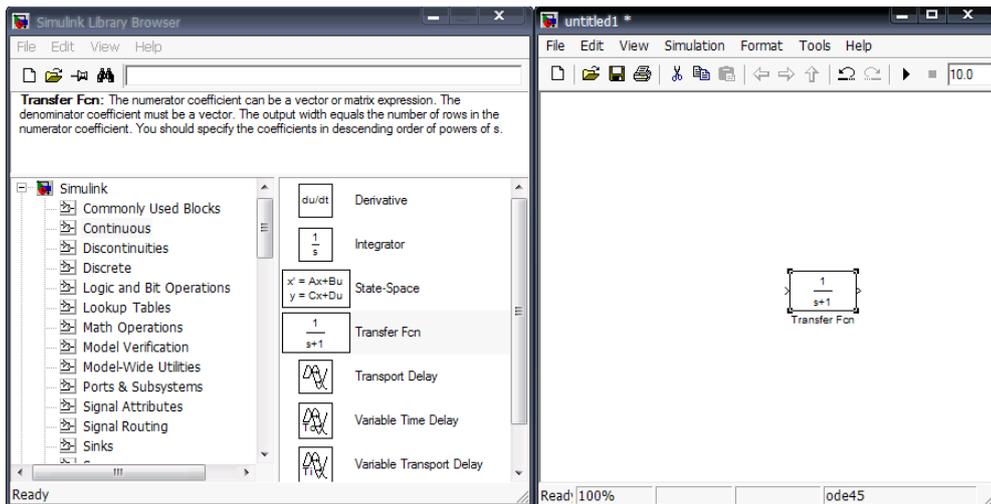
ภาพที่ 5.7 หน้าต่างสำหรับสร้างแบบจำลอง

สำหรับคำสั่งต่างๆ ดังนี้

ในส่วนบนของหน้าต่างสำหรับสร้างแบบจำลองจะเป็นเมนูบาร์

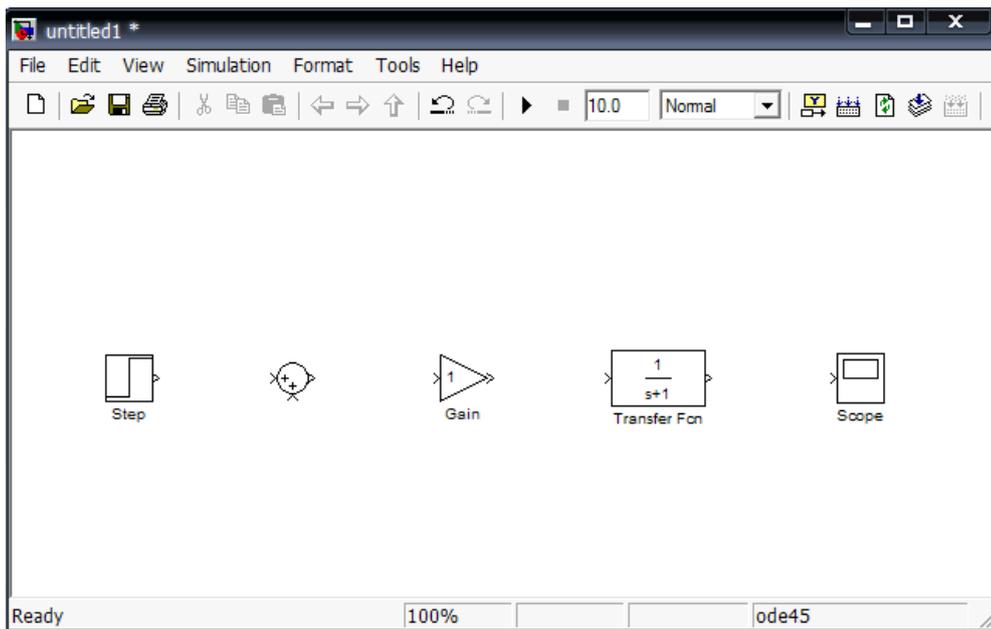
- เมนู File เป็นคำสั่งเกี่ยวกับการจัดการทามด่วนแฟ้มข้อมูลทั้งหมด เช่น การเปิดไฟล์ การบันทึกไฟล์ และการกำหนดคุณลักษณะของโปรแกรม Simulink เมนู Edit เป็นคำสั่งเกี่ยวกับการแก้ไขแบบจำลองที่สร้างขึ้น
- เมนู Edit เป็นคำสั่งเกี่ยวกับการแก้ไขแบบจำลองที่สร้างขึ้น
- เมนู View เป็นคำสั่งเกี่ยวกับการแสดงเครื่องมืออุปกรณ์บนหน้าจอแบบจำลอง
- เมนู Simulation เป็นคำสั่งเกี่ยวกับการจัดการการจำลองการทำงานของระบบควบคุมที่ทำการสร้างขึ้น
- เมนู Format เป็นคำสั่งเกี่ยวกับรูปแบบการแสดงผลบนหน้าจอจำลอง
- เมนู Tools เป็นคำสั่งเกี่ยวกับการเลือกใช้เครื่องมือต่าง ๆ
- เมนู Help เป็นคำสั่งเกี่ยวกับการขอความช่วยเหลือ

2) เมื่อทำการสร้างหน้าต่างสำหรับสร้างแบบจำลองแล้วทำบล็อกใน Simulink Library Browser มาวางในหน้าต่างสำหรับสร้างแบบจำลองได้โดยทำการคลิกบล็อกที่ต้องการค้างไว้จากนั้นทำการลากมาไว้ในหน้าต่างสำหรับสร้างแบบจำลองจากนั้นปล่อยปุ่มจะปรากฏบล็อกที่ต้องการบน หน้าต่างสำหรับสร้างแบบจำลองแสดงได้ดังภาพที่ 5.8



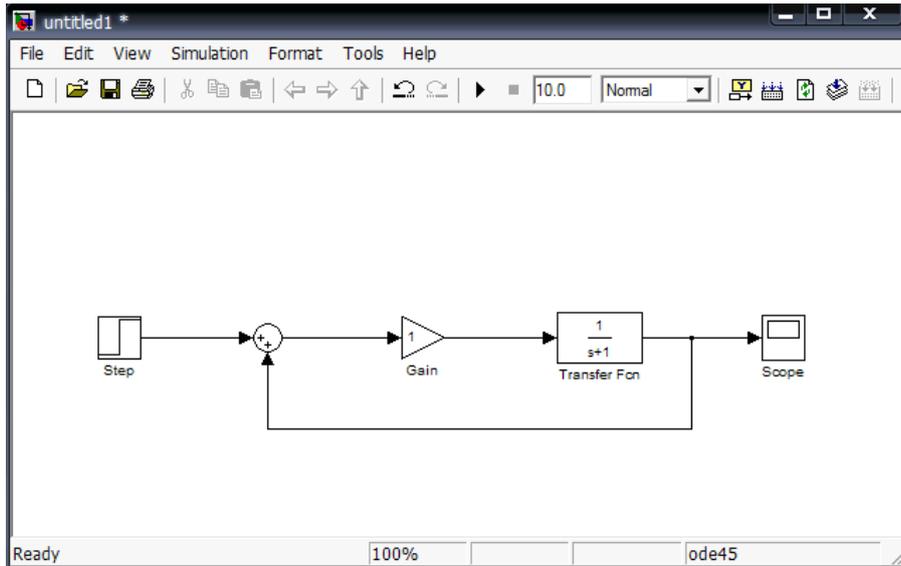
ภาพที่ 5.8 การเลือกบล็อกที่ต้องการจาก Simulink Library Browser

3) เมื่อได้บล็อกที่ต้องการแล้วเราสามารถนำบล็อกอื่นๆ มาวางบนหน้าต่าง สำหรับ สร้างแบบจำลองได้เช่นกัน ต่อไปจะแสดงบล็อกของระบบตัวอย่าง



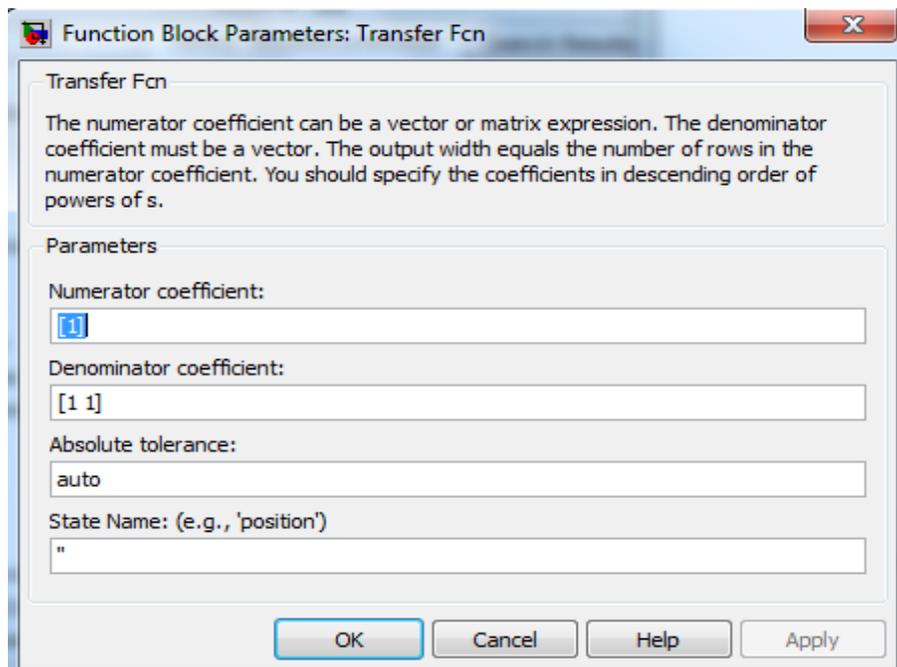
ภาพที่ 5.9 บล็อกของระบบตัวอย่าง

4) ภาพที่ 5.9 สังเกตได้ว่าแต่ละบล็อกจะมีช่องสัญญาณเข้าและสัญญาณออกไว้สำหรับลากเส้นสัญญาณเชื่อมต่อระหว่างบล็อกต่างๆ เข้าด้วยกันการลากเส้นเชื่อมต่อระหว่างบล็อกต่างๆ นั้น สามารถทำได้โดยเลื่อนเมาส์ให้ไปชี้ที่จุดต่อสัญญาณของบล็อกที่ต้องการ ตัวชี้ของเมาส์จะเปลี่ยนเป็นเครื่องหมายกากบาท จากนั้นคลิกค้างไว้แล้วลากตัวชี้ไปที่จุดต่อสัญญาณของบล็อกที่ต้องการ แสดงได้ดังภาพที่ 5.10



ภาพที่ 5.10 การเชื่อมต่อสัญญาณระหว่างบล็อก

5) การเปลี่ยนแปลงค่าพารามิเตอร์ของบล็อกสามารถทำได้โดยดับเบิลคลิกที่บล็อกซึ่งจะ ปรากฏไดอะล็อกสำหรับให้ใส่ค่าพารามิเตอร์ของบล็อก

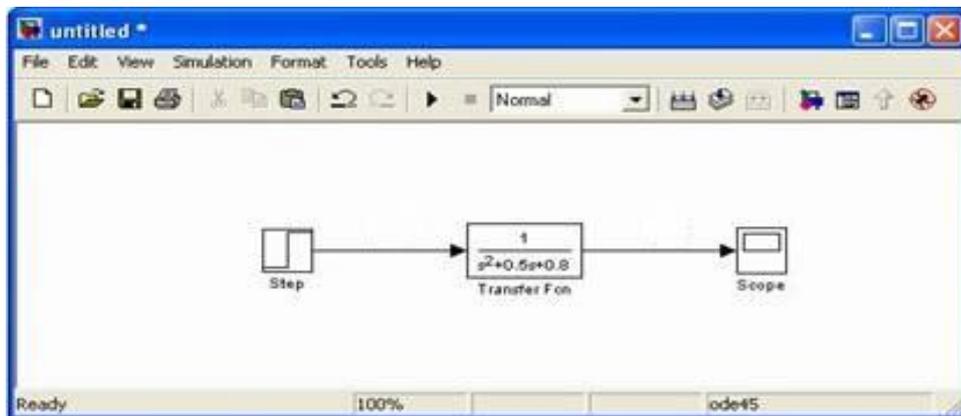


ภาพที่ 5.11 การเปลี่ยนแปลงค่าพารามิเตอร์ของบล็อก

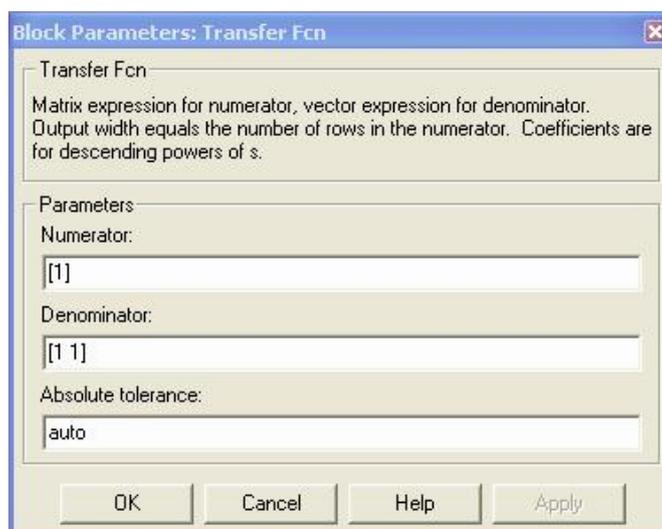
เพื่อให้เข้าใจหลักการทํางานและวิธีการสร้างโปรแกรมด้วย Simulink จะขอแสดงถึงวิธีการสร้างแบบจำลองของระบบควบคุมใน Simulink โดยการสร้างแบบจำลองของกระบวนการใน Simulink นั้นสามารถแบ่งเป็น 3 วิธีด้วยกัน คือ

1. การสร้างแบบจำลองกระบวนการจากสมการ Differential Equation
2. การสร้างแบบจำลองกระบวนการจากทรานส์เฟอร์ฟังก์ชัน
3. การสร้างแบบจำลองระบบควบคุมจากสมการ State-Space

ในงานวิจัยนี้จะใช้วิธีการสร้างแบบจำลองกระบวนการจากทรานส์เฟอร์ฟังก์ชัน วิธีการสร้างแบบจำลองของระบบที่แสดงอยู่ในรูป 5.12 Transfer Function นั้นจะใช้บล็อก Transfer Fcn แสดงแทนระบบ การใส่ค่าโพลีโนเมียลในส่วนของ Numerator และ Denominator นั้นทำได้โดยดับเบิลคลิกที่บล็อกซึ่งจะปรากฏไดอะล็อกที่ให้ใส่ค่าขึ้นมาแสดงได้ดังภาพที่ 5.13 โดยการใส่ค่าลงไปนั้นมีเงื่อนไขอยู่ว่า ส่วน Numerator นั้นต้องมีอันดับน้อยกว่า ส่วน Denominator ซึ่งสังเกตได้ว่าการสร้างแบบจำลองของระบบจาก Transfer Function นั้นจะทำการวัดผลตอบสนองได้ที่ส่วนเอาต์พุตของระบบเท่านั้น



ภาพที่ 5.12 ไดอะล็อกใส่ค่าพารามิเตอร์ของบล็อก Transfer Function



ภาพที่ 5.13 แบบจำลองระบบทางกลในรูปแบบของ Transfer Function

5.1.2 การติดตั้งโปรแกรมเพื่อใช้งาน Real Time Windows Target

5.1.2.1 Simulink external model interface

ในหน้าต่าง Command Windows ของโปรแกรม MATLAB ให้พิมพ์คำสั่ง

```
>> mex -setup
```

จากนั้นโปรแกรม Matlab จะแสดงข้อความ

```
Please choose your compiler for building external interface (MEX) files: Would you like
mex to locate installed compilers [y]/n?
```

ให้พิมพ์ y เพื่อยืนยัน จากนั้นโปรแกรม Matlab จะแสดงข้อความเพื่อให้เลือก

Compiler ที่ต้องการ

```
Select a compiler:
```

```
[1] Lcc-win32 C 2.4.1 in C:\PROGRA~1\MATLAB\R2007a\sys\lcc
[2] Microsoft Visual C++ 2005 in C:\Program Files\Microsoft Visual Studio 8
[3] Microsoft Visual C++ 6.0 in C:\Program Files\Microsoft Visual Studio
[0] None
```

ในการวิจัยนี้ผู้วิจัยจะเลือก Compiler Microsoft Visual C++ ซึ่งทำได้โดย

การพิมพ์เลข 2

```
Compiler: 2
```

โปรแกรม Matlab จะแสดงข้อความเพื่อให้เรายืนยัน Compiler ที่เหลือ

```
Please verify your choices:
```

```
Compiler: Microsoft Visual C++ 2005
Location: C:\Program Files\Microsoft Visual Studio 8
Are these correct?([y]/n):
```

ถ้ายืนยันให้พิมพ์ y แต่ถ้าไม่ยืนยันให้พิมพ์ n จากนั้นโปรแกรมจะอัปเดตข้อมูลของ Compiler และสิ้นสุดการติดตั้ง Compiler โดยสังเกตข้อความในโปรแกรม Matlab ดังตัวอย่างด้านล่าง

```
Trying to update options file: C:\Documents and Settings\xp\Application
Data\MathWorks\MATLAB\R2007a\mexopts.bat
From template: C:\PROGRA~1\MATLAB\R2007a\bin\win32\mexopts\msvc80opts.bat
Done . . .
```

5.1.2.2 การติดตั้ง Real Time Windows Target Kernal

ในหน้าต่าง Command Windows ของโปรแกรม MATLAB ให้พิมพ์คำสั่ง

```
>> rtwintgt -install
```

จากนั้นโปรแกรม Matlab จะแสดงข้อความเพื่อเริ่มต้นการติดตั้ง

You are going to install the Real-Time Windows Target kernel.
Do you want to proceed? [y] :

พิมพ์ y เพื่อยืนยันการเริ่มต้นการติดตั้ง Real Time Windows Target Kernel หลังจากนั้นโปรแกรมจะดำเนินการติดตั้ง เมื่อเสร็จเรียบร้อยแล้วโปรแกรมจะแสดงข้อความ

The Real-Time Windows Target kernel has been successfully installed.

5.1.2.3 การกำหนดค่าพารามิเตอร์ใน Simulink สำหรับ Real Time Workshop

หลังจากที่สร้าง Simulink Model จะต้องกำหนดพารามิเตอร์ในหน้าต่าง Simulation เพื่อให้ Real-Time Workshop สามารถสร้าง Code ภาษา C และสร้าง Real-Time Application ได้ ซึ่งมีขั้นตอนในการกำหนด ดังนี้

1) ในหน้าต่าง Simulink ไปที่ Simulation เมนู คลิกที่ Configuration Parameters

2) คลิกที่หมวดของ Hardware Implementation

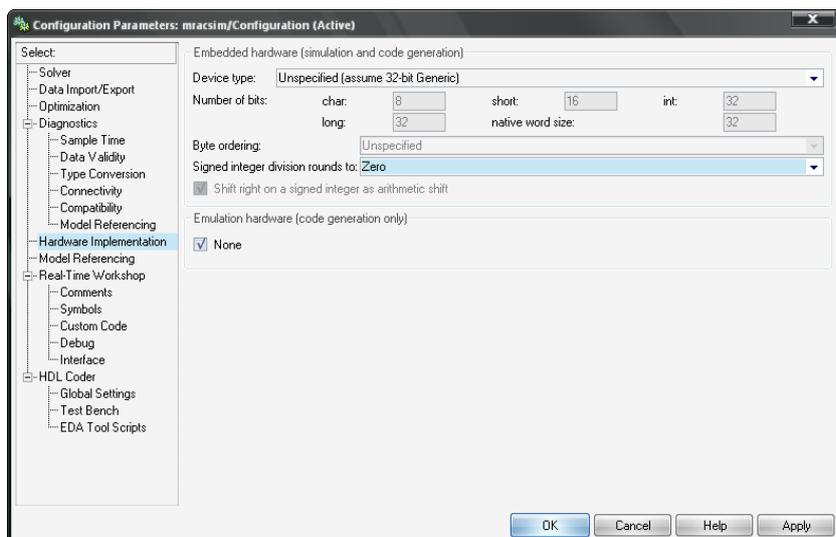
3) จากรายการ Device Type เลือก 32-bit Real Time Windows Target

4) Emulation hardware เลือก None

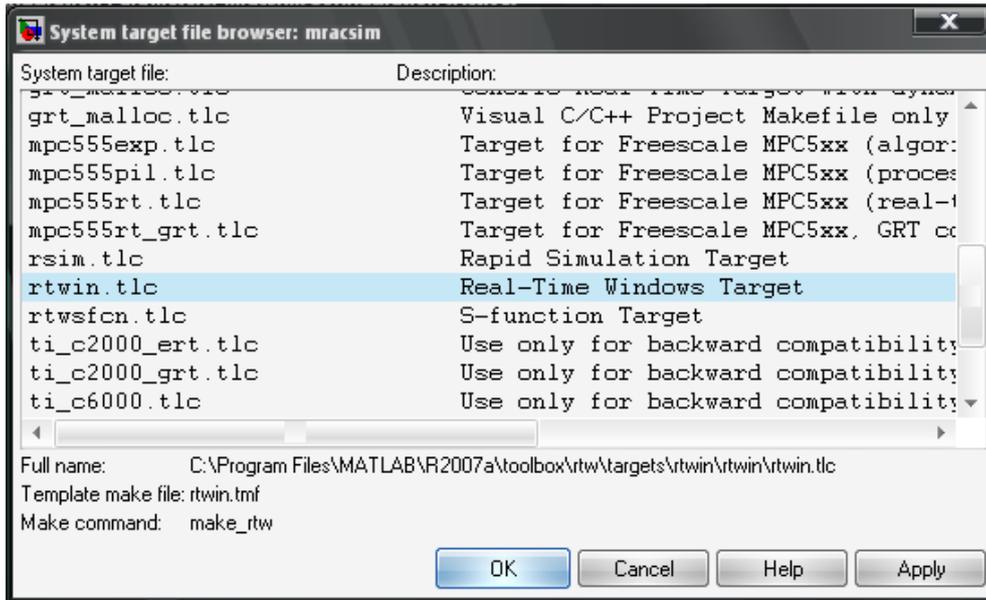
5) คลิกที่หมวด Real Time Workshop

6) Target selection คลิก Browse ที่รายการ RTW system target file เลือก system target file สำหรับ Real Time Windows Target และคลิก Ok

7) คลิก Ok เพื่อยืนยันการเปลี่ยนแปลงและปิด Dialog box



ภาพที่ 5.14 Hardware Implementation



ภาพที่ 5.15 RTW System Target File

5.2 การเขียนโปรแกรมภาษาซีสำหรับหน่วยประมวลผล ARM7

5.2.1 สถาปัตยกรรมไมโครคอนโทรลเลอร์ ARM7 (ADuC7024)

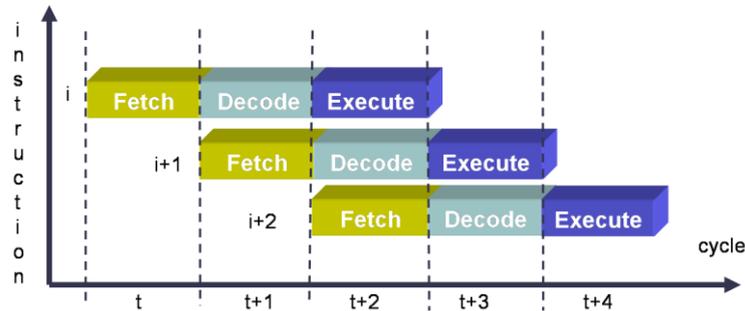
ARM7 มีรูปแบบการออกแบบพื้นฐานตามสถาปัตยกรรมของ Von Neumann ด้วยบัสขนาด 32 บิต ซึ่งสามารถรับส่งคำสั่งและข้อมูลขนาด 8 บิต, 16 บิตหรือ 32 บิต โดยทั่วไปมีคุณสมบัติดังนี้

- ใช้รูปแบบคำสั่งแบบ load and store หมายความว่าก่อนการประมวลผลจะต้องทำการโหลดข้อมูลจากรีจิสเตอร์หรือหน่วยความจำก่อนหลังจากนั้นจึงการประมวลผลซึ่งผลลัพธ์ที่ได้อาจจะเก็บอยู่ในรีจิสเตอร์เพื่อไปสู่กระบวนการถัดไปหรืออาจจะบันทึกกลับลงสู่หน่วยความจำ
- มีโหมดของการอ้างตำแหน่งแอดเดรสที่ไม่มากนัก โดยสามารถกระทำได้โดยอ้างตำแหน่งจากรีจิสเตอร์และคำสั่งเพียงคำสั่งเดียวเท่านั้น
- ขนาดของคำสั่งมีขนาดคงที่ (32 บิต)
สำหรับ ARM แล้ว ได้เพิ่มความสามารถเพิ่มเติม คือ
- มี shifter หรือตัวเลื่อนบิตแยกต่างหากก่อนที่จะเข้า ALU (Arithmetic Logic Unit) เพื่อเพิ่มความเร็วในการประมวลผล
- มีโหมดการอ้างตำแหน่งแบบ Auto-increment และ Auto-decrement เพื่อช่วยในการทำงานแบบลูปง่ายขึ้น
- สามารถเพิ่มเงื่อนไขภายในแต่ละคำสั่งเพื่อลดการสูญเสียจากการทำงานของ pipeline ซึ่งจะส่งผลให้การทำงานเร็วขึ้น

1) กระบวนการแบบไปป์ไลน์ (instruction pipeline)

ARM7 ใช้ไปป์ไลน์ 3 ระดับในการในการเพิ่มความเร็วในการทำงานกับชุดคำสั่ง ทั้งนี้ค่าในรีจิสเตอร์ program counter (PC) ที่ใช้ในการเอ็ชชีควิต์คำสั่งจะมีค่าแอดเดรสหน้าไปสองคำสั่งเสมอ ทำให้เกิดหลายการปฏิบัติงานในเวลาเดียวกัน ดังรูป จะเห็นว่าขณะที่ซีพียูทำการเอ็ชชีควิต์คำสั่งที่ 1

ซีพียูก็ถอดรหัสคำสั่งที่ 2 และทำการเพช้คำสั่งที่ 3 จากหน่วยความจำไปพร้อมกัน. ทำให้ซีพียู ARM7 ทำงานได้อย่างรวดเร็วและสามารถรองรับการประมวลผลข้อมูลขนาดใหญ่



ภาพที่ 5.16 แสดงกระบวนการทำงานของไปป์ไลน์ใน ARM7 โพรเซสเซอร์

2) การจัดรูปแบบข้อมูลในหน่วยความจำ

อินเตอร์เฟซ กับหน่วยความจำของซีพียู ARM7 ถูกออกแบบให้มีสมรรถนะดีเยี่ยม และลดการใช้หน่วยความจำให้น้อยที่สุด สัญญาณควบคุมความเร็วถูกส่งผ่านทางท่อเพื่อให้มีฟังก์ชันควบคุมระบบที่รองรับโหมดการเข้าถึงที่รวดเร็วโดยเทคโนโลยีหน่วยความจำแบบต่างๆ ARM7 มีชนิดของไซเคิล 4 แบบ ดังนี้

- แบบภายใน Internal
- แบบไม่เป็นลำดับ Non-sequential
- แบบเป็นลำดับ Sequential
- Coprocessor transfer

3) Coprocessor

โพรเซสเซอร์ร่วมเป็นวงจรถิเศษในชิปไมโครโพรเซสเซอร์ ที่ได้รับการออกแบบให้ควบคุมหรือทำฟังก์ชันพิเศษเร็วกว่าวงจรมิโครโพรเซสเซอร์พื้นฐาน โดย ARM7 coprocessors ประกอบด้วย

- Instruction pipeline
- Instruction decode logic
- Handshake logic
- Register bank
- Processing logic พิเศษ

สถานะการทำงาน : สถานะการทำงานของ ARM แบ่งเป็น 7 แบบ ได้แก่

1. User mode (usr) เป็นโหมดสำหรับการเอ็กซิคิวต์โปรแกรมที่ใช้ทั่วไป ในโหมดนี้ จะไม่สามารถติสเอเบิลอินเตอร์รัปต์ได้
2. Fast Interrupt mode (fiq) เป็นโหมดสำหรับโปรแกรมอินเตอร์รัปต์ที่มีลำดับความสำคัญสูงกว่าอินเตอร์รัปต์ตัวอื่น
3. Interrupt mode (irq) เป็นโหมดสำหรับโปรแกรมอินเตอร์รัปต์ทั่วไป
4. Supervisor (svc) เป็นโหมดสำหรับระบบปฏิบัติการ

5. Abort (abt) เป็นโหมดสำหรับการเข้าถึงหน่วยความจำที่ไม่มีอยู่จริงเพื่อป้องกันการดำเนินงานของระบบ

6. Undefined (und) เป็นโหมดสำหรับการเอ็กซ์คิวต์คำสั่งที่ ARM ไม่รู้จัก

7. System (SYS) – เป็น user mode แบบพิเศษสำหรับระบบปฏิบัติการ ในแต่ละโหมดจะมีลักษณะการใช้งานที่แตกต่างกัน ดังรูป

System & User	FIQ	Supervisor	Abort	IRQ	Undefined
R0	R0	R0	R0	R0	R0
R1	R1	R1	R1	R1	R1
R2	R2	R2	R2	R2	R2
R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6
R7	R7_fiq	R7	R7	R7	R7
R8	R8_fiq	R8	R8	R8	R8
R9	R9_fiq	R9	R9	R9	R9
R10	R10_fiq	R10	R10	R10	R10
R11	R11_fiq	R11	R11	R11	R11
R12	R12_fiq	R12	R12	R12	R12
R13	R13_fiq	R13_svc	R13_abt	R13_irq	R13_und
R14	R14_fiq	R14_svc	R14_abt	R14_irq	R14_und
R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)
CPSR	CPSR SPSR_fiq	CPSR SPSR_svc	CPSR SPSR_abt	CPSR SPSR_irq	CPSR SPSR_und

ภาพที่ 5.17 ลักษณะการใช้งาน ARM ทั้ง 7 โหมด

โหมดอื่นนอกจาก user mode ถือเป็น privileged mode ซึ่งก็คือโหมด พิเศษที่ใช้สำหรับให้บริการอินเทอร์รัพท์เอ็กซ์เซ็ปชันหรือการเข้าถึง resource ที่ถูกป้องกัน

4) ARM registers

ARM มีรีจิสเตอร์แบบ 32 บิต ทั้งหมด 37 ตัว แบ่งเป็น รีจิสเตอร์สำหรับใช้งานทั่วไป 30 ตัว รีจิสเตอร์แสดงสถานะ 6 ตัว และ program counter 1 ตัว หน้าที่ของรีจิสเตอร์ที่สำคัญสรุปได้ดังนี้

- Stack pointer รีจิสเตอร์ R13 ทำหน้าที่เก็บ ตำแหน่งของคำสั่งในหน่วยความจำที่ต้องทำเป็นลำดับถัดไป เมื่อทำงานในโปรแกรมย่อยเสร็จแล้วให้แก่ Program Counter Register
- Link register รีจิสเตอร์ R14 ทำหน้าที่ในการเก็บค่าที่อยู่ของคำสั่งถัดไปหลังจากคำสั่ง Branch with Link (BL) ที่ใช้เวลาเรียกโปรแกรมย่อยของ ARM
- Program counter รีจิสเตอร์ R15 ทำหน้าที่เป็น Program Counter ทำการชี้ไปยังตำแหน่งของคำสั่งถัดไป เมื่อเริ่มทำการ execute คำสั่งปัจจุบัน
- CPSR (Current Program Status Register) เก็บค่าสถานะโปรแกรม ณ เวลาปัจจุบัน

- SPSR (Saved Program Status Register) เก็บค่าของ CPSR ก่อนจะเกิดเอกเซปชัน

ARM Instruction set แบ่งได้ 4 ประเภทคือ

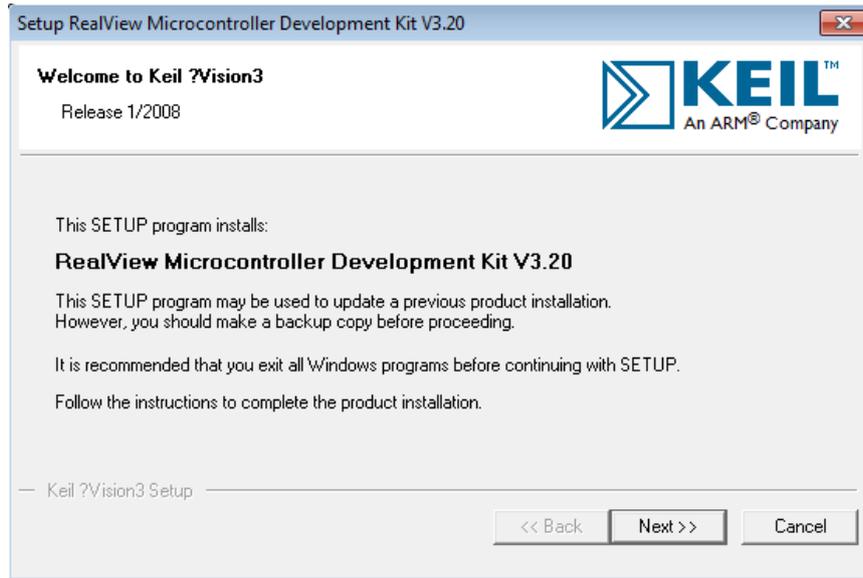
- 1) Branch Instruction
 - General branch instruction
 - Branch with Link
 - Software interrupt (SWI) ใช้เรียกเพื่อเข้าสู่การทำงานของระบบปฏิบัติการ
 - THUMB instruction (16 บิต)
- 2) Data - processing Instruction
 - Arithmetic/logic instructions มี 16 คำสั่ง ซึ่งมีรูปแบบร่วมกัน
 - Multiply instructions มีแบบ 32 บิต และ 64 บิต
 - Status register transfer instruction โหลดหรือเก็บค่าจาก SPSR หรือ CPSR
- 3) Load - store Instruction แบ่งเป็น 3 ประเภทคือ
 - Load or store the value of a single register คือ ในการโหลดค่ามาเก็บไว้ในรีจิสเตอร์สามารถทำได้ครั้งละ 32 บิต, 16 บิต หรือ 8 บิต จาก memory ในการstore ก็
 - Load and store multiple register values สามารถโหลดหรือเก็บค่าในช่วงของMemory โดยเพิ่มหรือลด address ได้โดยอัตโนมัติ
 - ในการอ้างอิงแอดเดรสแบบ Increment หรือ decrement จะเป็นสิ่งที่ระบุว่าค่าของ pointer จะเลื่อนขึ้นหรือเลื่อนลง และในส่วนของ pre หรือ post จะระบุว่าขณะนั้น pointer ชี้ในจุดที่มีข้อมูลอยู่หรือไม่ ถ้ามีอยู่ต้องเลื่อนpoint ก่อนแล้วค่อยใส่ค่าลงไป หรือถ้าไม่มีข้อมูลอยู่จะใส่ค่าก่อน แล้วจึงเลื่อน pointer ไป
 - Swap a register with the value of a memory location คือการสลับค่าระหว่างregisterโดยผ่าน memory หรือสลับค่าระหว่าง memory กับ register ได้
- 4) Co - processor Instruction
 - Data-processing instructions ใช้จัดการการจัดการภายในของโปรเซสเซอร์

รวม

 - Register transfers ย้ายค่าภายในโปรเซสเซอร์รวมมายัง ARM รีจิสเตอร์ หรือจากARM รีจิสเตอร์ออกไป
 - Data-transfer instructions ย้ายข้อมูลของโปรเซสเซอร์รวมมายังหรือออกจาก memory โดยตำแหน่งจะถูกคำนวณโดย ARM

5.2.2 โปรแกรม Keil uVision 3.11

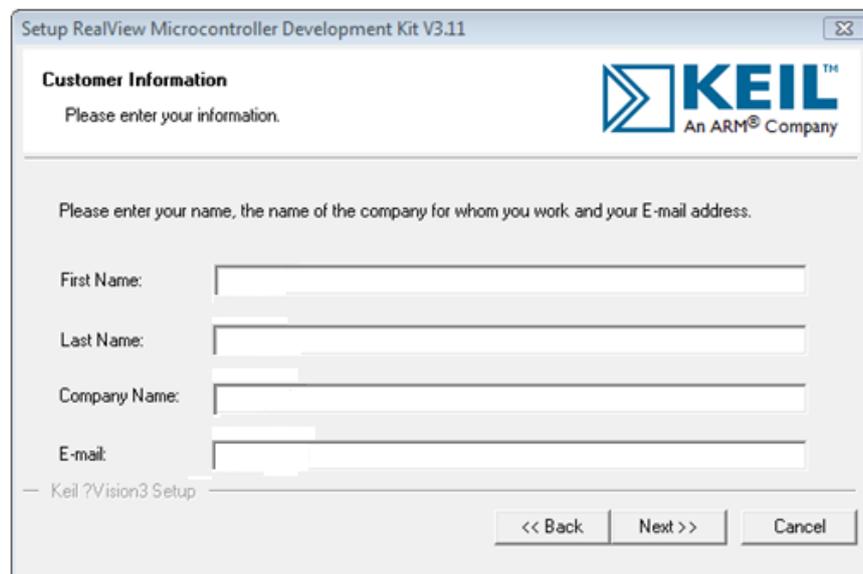
โปรแกรม Keil RealView [11] สามารถดาวน์โหลดได้จากเว็บไซต์ของบริษัท Keil ที่ <http://keil.com> แล้วเลือกที่เมนู Software Download จากนั้นเลือกที่ Evaluation Software โดยก่อนที่จะทำการดาวน์โหลดซอฟต์แวร์จะต้องทำการลงทะเบียนกับทางเว็บไซต์ก่อนจึงจะสามารถดาวน์โหลดได้ หลังจากดาวน์โหลดเสร็จสิ้นจะได้ไฟล์ที่มีนามสกุล *.exe (ในที่นี้คือ rvmdk311.exe) ดับเบิลคลิกที่ไฟล์เพื่อติดตั้งโปรแกรม โดยโปรแกรมจะแสดงหน้าต่างสำหรับติดตั้งโปรแกรกดังภาพที่ 5.17



ภาพที่ 5.18 หน้าต่าง Setup Real View Microcontroller Development Kit Version 3.11

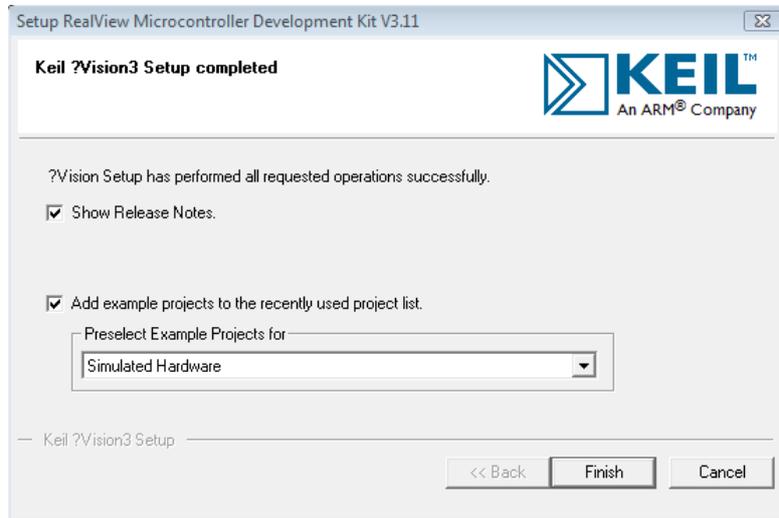
เมื่อคลิกปุ่ม Next จะปรากฏหน้าต่าง License Agreement เพื่อแจ้งลิขสิทธิ์ของโปรแกรมคลิกที่ หน้าข้อความ I agree to all terms of proceeding License Agreement แล้วคลิกปุ่ม Next

หลังจากนั้นจะมีหน้าต่าง Folder Selection เลือกโฟลเดอร์สำหรับการติดตั้งโปรแกรมในที่นี้ให้เลือกเป็น C:\Keil ตามที่เครื่องกำหนดมาให้ แล้วคลิก Next จะปรากฏหน้าต่าง customer Information เพื่อกรอกข้อมูลของผู้ใช้งาน ซึ่งควรกรอกข้อมูลตามความเป็นจริง โดยเฉพาะในช่องของ e-mail ถ้าไม่กรอกข้อมูลจะไม่สามารถคลิกปุ่ม Next เพื่อดำเนินการในขั้นตอนต่อไปได้ แสดงดังภาพที่ 5.18



ภาพที่ 5.19 หน้าต่าง customer Information

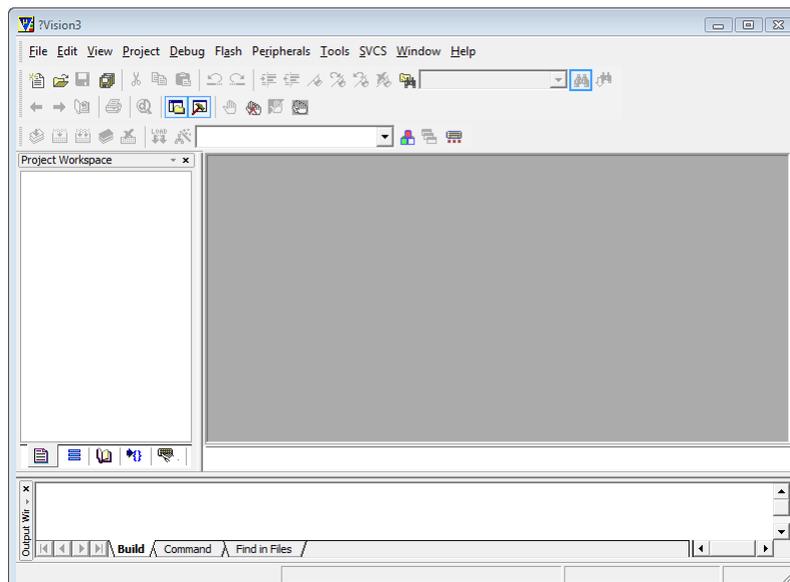
หลังจากนั้นขั้นตอนการติดตั้งโปรแกรมก็จะเริ่มขึ้น หลังจากเสร็จเรียบร้อยแล้วจะปรากฏหน้าต่าง Setup Complete เพื่อเป็นการยืนยันการติดตั้งโปรแกรมให้คลิกที่ปุ่ม Finish เป็นการเสร็จสิ้นการติดตั้งโปรแกรม แสดงดังรูป 5.19



ภาพที่ 5.20 หน้าต่างแสดงการสิ้นสุดการติดตั้งโปรแกรม

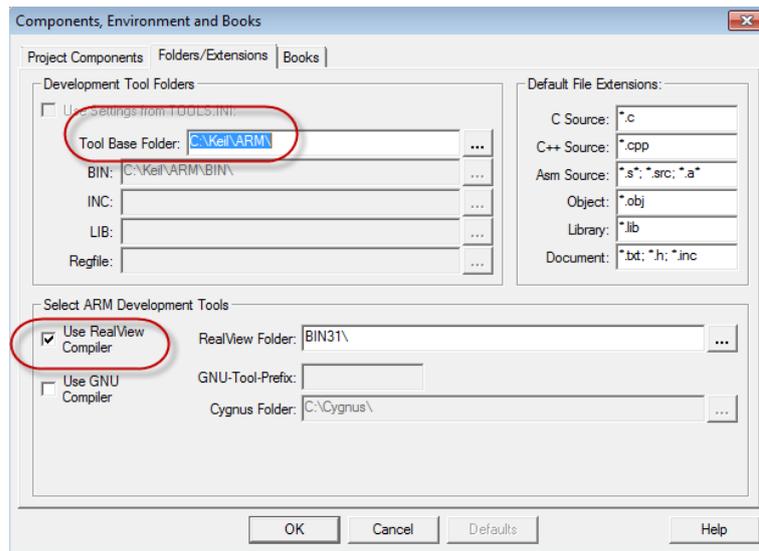
5.2.2.1 การเริ่มต้นใช้งานโปรแกรม Keil uVision3

การใช้งานโปรแกรม Keil RealView ในการเขียนโปรแกรมนั้นจะต้องทำการสร้างโปรเจกต์ขึ้นมาและทำการกำหนดค่าการใช้งานให้ถูกต้องเสียก่อนจึงจะสามารถใช้งานโปรแกรมได้ โดยเริ่มต้นจากการเปิดโปรแกรม Keil uVision3 ขึ้นมา แสดงดังภาพที่ 4.20



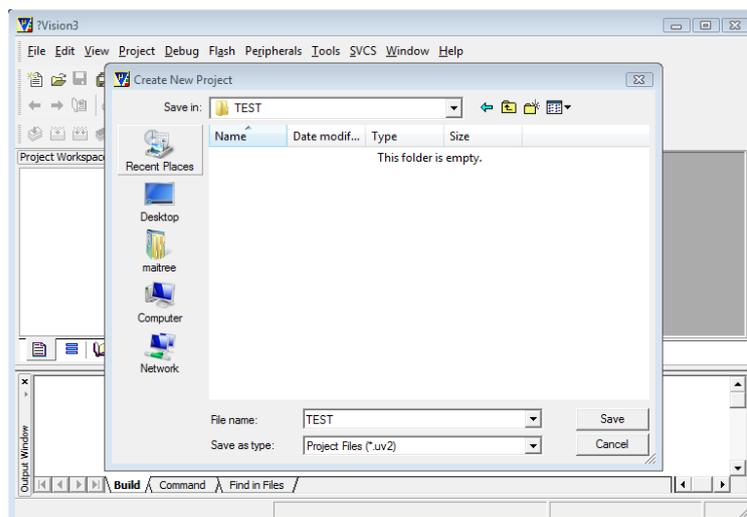
ภาพที่ 5.21 หน้าต่างหลักของโปรแกรม Keil uVision

กำหนดค่าเลือกตัวคอมไพเลอร์ในการแปลคำสั่งให้กับโปรแกรม Keil uVision3 โดยคลิกที่ Project -> Manage -> Components, Environment, Books... เพื่อเปิดหน้าต่าง Components, Environment and Books ซึ่งมีส่วนที่ต้องกำหนดอยู่ 2 จุดคือ จุดแรกในส่วนของ Select ARM Development Tool จะมีคอมไพเลอร์ให้เลือกอยู่ 2 ตัว ซึ่งในส่วนนี้ให้เลือกคอมไพเลอร์ RealView Compiler และอีกจุดหนึ่งที่ต้องกำหนดคือในส่วนของ Development Tool Folder โดยที่ Tool Base Folder ให้กำหนดไว้ที่ C:\Keil\ARM เสร็จแล้วกดปุ่ม OK



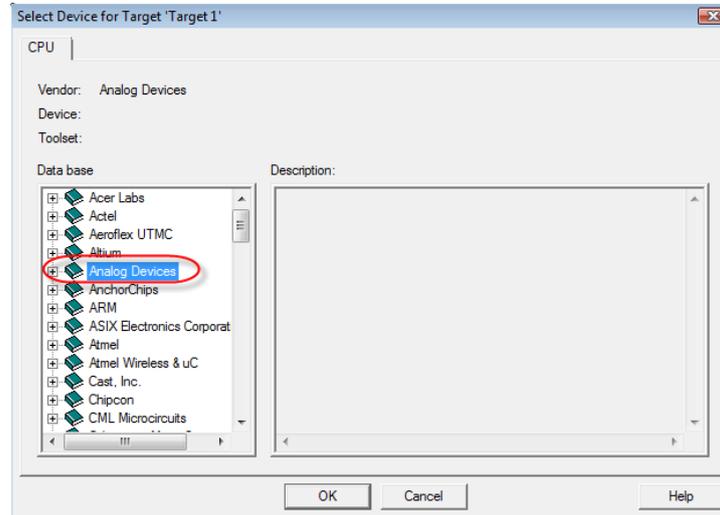
ภาพที่ 5.22 หน้าต่าง Components, Environment and Books

ต่อจากนั้นจะต้องทำการสร้างโปรเจกต์ใหม่ขึ้นมาโดยเลือกที่ Project -> New uVision Project จะปรากฏหน้าต่าง Create New Project ขึ้นมา แล้วทำการสร้างไฟล์เดอริใหม่ขึ้นมา แล้วตั้งชื่อไฟล์เดอริ จากนั้นตั้งชื่อโปรเจกต์ที่ต้องการ แล้วทำการบันทึกโปรเจกต์ แสดงดังภาพที่ 5.22



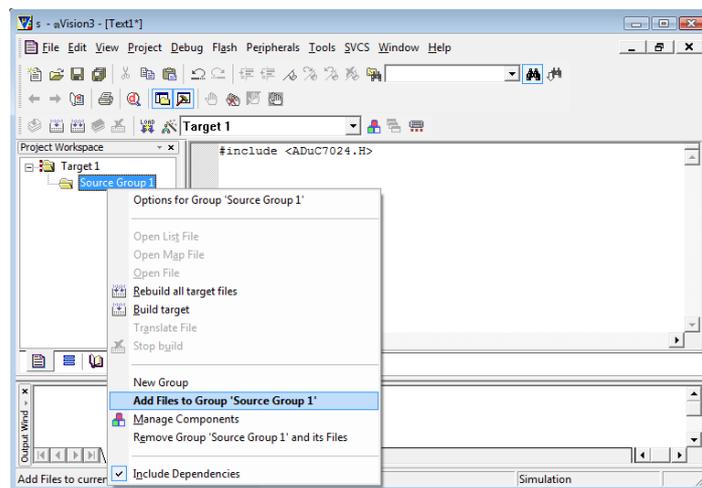
ภาพที่ 5.23 หน้าต่าง Create New Project

เมื่อสั่งกดปุ่ม Save แล้วจะปรากฏหน้าต่าง Select Device for Target 'Target 1' เพื่อให้ผู้ใช้เลือก MCU ที่ใช้งาน ซึ่งในที่นี้เลือกไมโครคอนโทรลเลอร์ของบริษัท Analog Devices จากนั้นจะมีรายละเอียดเกี่ยวกับเบอร์ของไมโครคอนโทรลเลอร์ Analog Devices ซึ่งเบอร์ที่ใช้คือ ADuC7024 จากนั้นกดปุ่ม OK โปรแกรมจะถามว่าต้องการที่จะเพิ่มไฟล์ Startup เข้าไปในโปรเจกต์หรือไม่ (ถ้าต้องการกด YES ถ้าไม่กด NO)



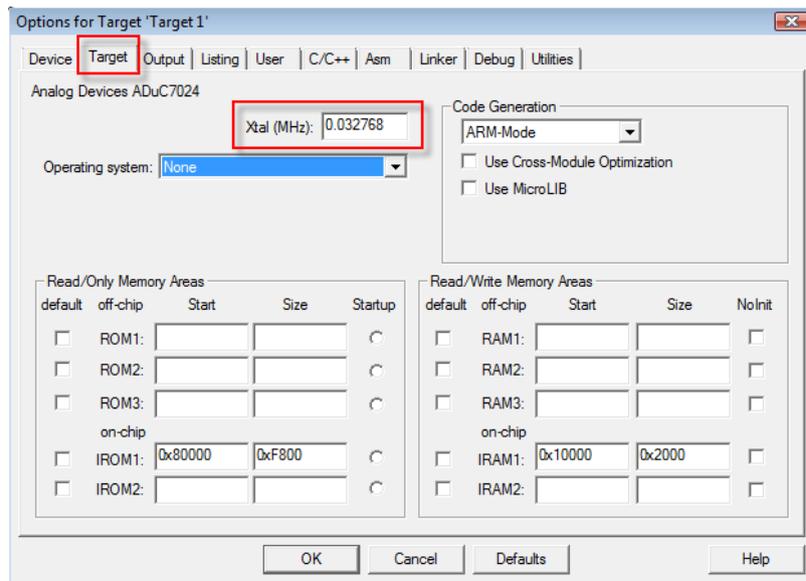
ภาพที่ 5.24 หน้าต่าง Select Device for Target

หลังจากขั้นตอนนี้เราก็จะสามารถเริ่มเขียนโปรแกรมได้แล้วโดยเลือกที่ File -> New... ซึ่งจะปรากฏหน้าต่างสำหรับเขียน Text File ขึ้นมา ไปที่หน้าต่าง Text Editor ให้คลิกขวาแล้วเลือก Insert '#include <ADuC7024.H>' ข้อความ #include <ADuC7024.H> จะถูกแทรกในบรรทัดที่ 1 หลังจากที่เราเขียนโปรแกรมเรียบร้อยแล้วให้บันทึกไฟล์ที่มีนามสกุล .c



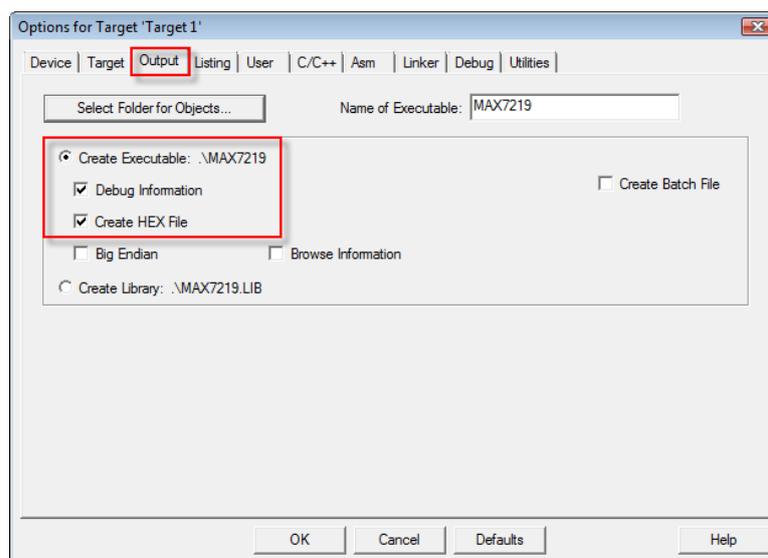
ภาพที่ 5.25 หน้าต่าง Add File to Group

ในส่วนของ Project Workspace ในภาพที่ 5.24 จะมีโฟลเดอร์ Target 1 คลิกตรงเครื่องหมายบวกจะมีโฟลเดอร์ย่อยชื่อ Source Group 1 ให้คลิกขวาจะมีเมนูย่อยให้เลือกที่ Add File to Group 'Source Group 1' แล้วเลือกแอดไฟล์โปรแกรม (นามสกุล .c) และไฟล์ Startup เข้ามาในโปรเจกต์ จากนั้นทำการกำหนดค่า Option ของ Project File โดยคลิกปุ่ม Option for Target เพื่อเปิดหน้าต่าง Option for Target 'Target1' เลือกแท็บ Target เพื่อกำหนด MCU Target โดย กำหนดค่า Xtal เป็น 0.032768 ดังภาพที่ 5.25



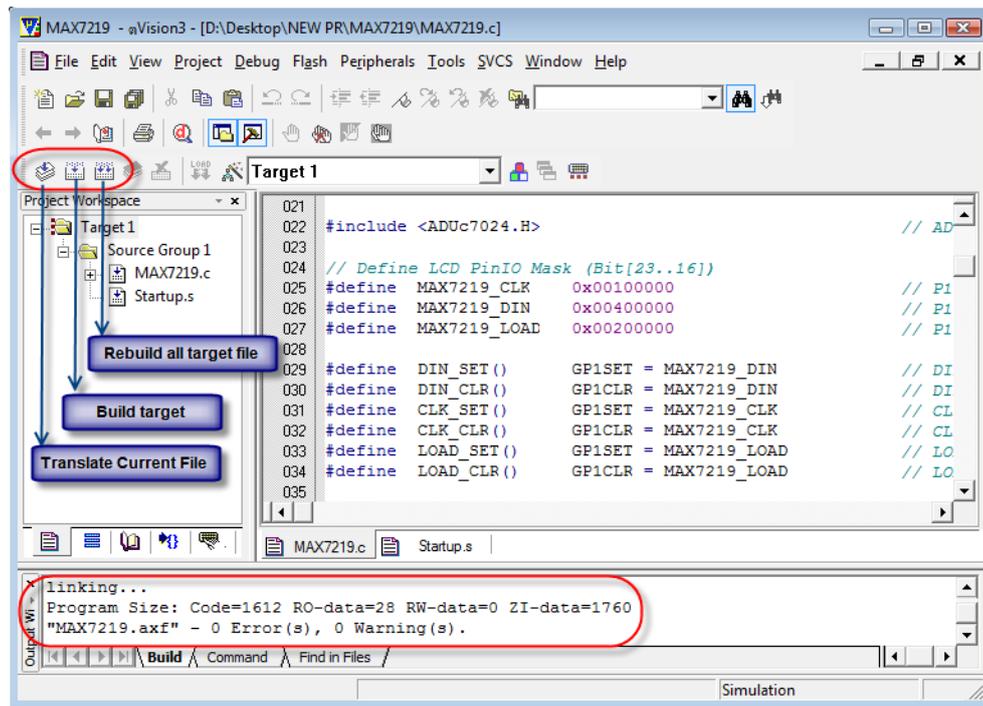
ภาพที่ 5.26 หน้าต่างกำหนด Options for Target

ต่อไปเลือกแท็บกำหนด Output Information โดยเลือกที่ Output แล้วคลิกเลือก Create Executable และเลือกที่ Debug information และ Create HEX file ดังภาพที่ 5.26



ภาพที่ 5.27 หน้าต่างกำหนด Output Information

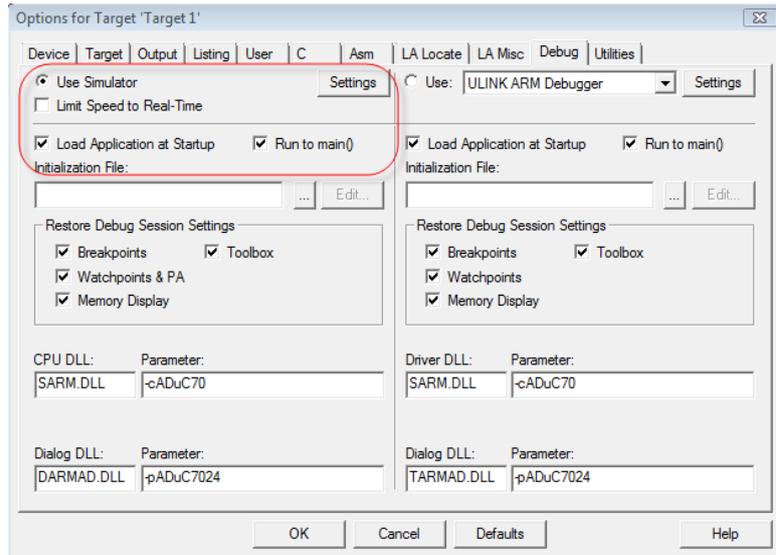
หลังจากที่เขียนโปรแกรมเรียบร้อยแล้วต้องทำการ Compile และ Build โปรแกรมเสียก่อนโดยในภาพที่ 5.27 มีจุดสำหรับ Compile และ Build อยู่ 3 แบบคือ Translate current file, Build target และ Rebuild all target file หลังจากที่ทำ Compile และ Build โปรแกรมเรียบร้อยแล้วในส่วนด้านล่างคือ Output Window ที่ทำหน้าที่รายงานผลของการ Compile โปรแกรม ถ้าโปรแกรมถูกต้องทั้งหมดจะต้องขึ้น 0 Error(s), 0 Warning(s) แต่ถ้ามีจุดผิดพลาดโปรแกรมก็จะรายงานว่ามีที่ผิดที่ที่และผิดที่ตรงไหน ซึ่งจะช่วยให้เราสามารถแก้ไขโปรแกรมได้สะดวกขึ้น



ภาพที่ 5.28 การ compile และการ Build โปรแกรม

5.2.2.2 การ debug โปรแกรมภาษา C สำหรับ ARM7

การดีบัคโปรแกรมเป็นการตรวจสอบการทำงานของโปรแกรมที่เราเขียนขึ้นมาว่ามีความถูกต้องและเป็นไปตามที่เราต้องการหรือไม่ ซึ่งสามารถตรวจสอบได้เป็นลำดับขั้นตอนอีกทั้งยังตรวจหาจุดผิดของโปรแกรมได้ด้วย ซึ่งหลังจากที่ทำการคอมไพล์ผ่านแล้วให้กำหนดค่าตามภาพที่ 5.28 จากนั้นให้คลิกที่ปุ่มดีบัค (ตามภาพที่ 5.29 หรือเลือกที่เมนู Debug -> Start/Stop Debug Session โปรแกรมก็จะเข้าสู่โหมดดีบัค (โปรแกรมที่ใช้ในการดีบัคเป็นโปรแกรมสำหรับ LED ติดดับโดยพอร์ตที่ใช้เป็นพอร์ต P3.x)



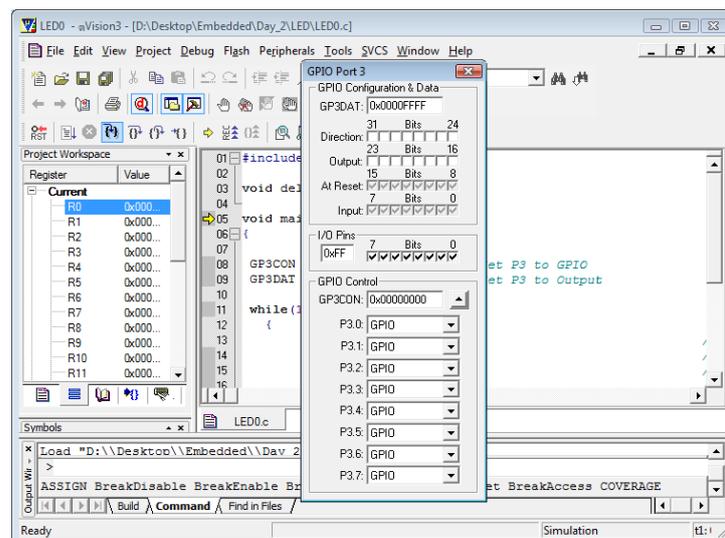
ภาพที่ 5.29 ปุ่ม Build Target

เมื่อกำหนดค่าแล้วให้ไปที่ Debug -> Start/Stop Debug Session ดังแสดงในภาพที่ 5.29



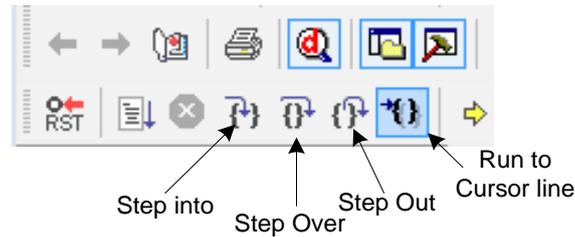
ภาพที่ 5.30 การเข้าสู่โหมด Simulation หรือ Debug

หลังจากที่เข้าสู่โหมดดีบัคแล้วให้เลือกที่ Peripherals -> General Purpose Input/Output -> GPIO Port (0-4) ดังภาพที่ 5.30



ภาพที่ 5.31 แสดงสถานะเตรียมพร้อม Simulator

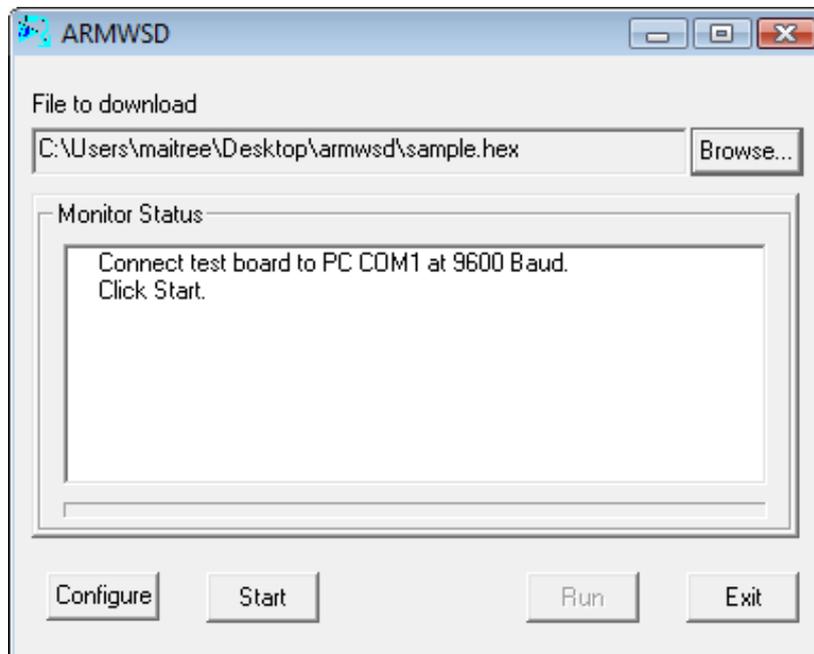
ถึงขั้นตอนนี้ก็สามารถ Run โปรแกรมได้แล้ว โดยไปที่ Debug -> Run หรือกด F5 ดังแสดงในภาพที่ 5.31



ภาพที่ 5.32 ปุ่มการ Debug แบบต่างๆ

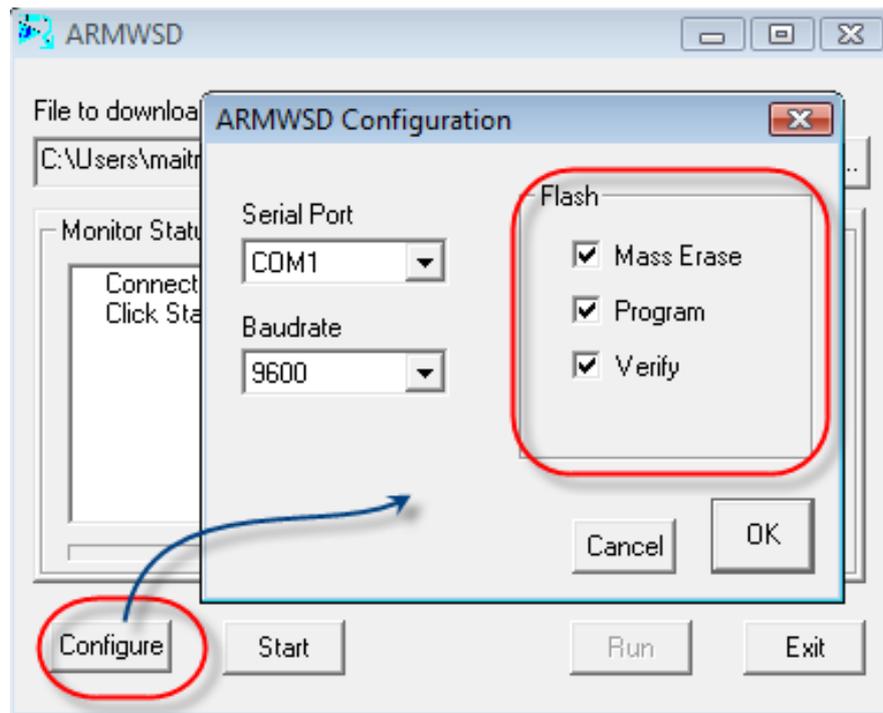
5.2.2.3 การติดตั้งซอฟต์แวร์ ARMWSD

หลังจากที่เขียนโปรแกรมภาษา C เสร็จแล้ว ทำการคอมไพล์จะได้ไฟล์ภาษาเครื่องนามสกุล .hex ซึ่งจะต้องใช้ซอฟต์แวร์ ARMWSD เพื่อดาวน์โหลดไฟล์ .hex ที่ได้ลงไปหน่วยความจำแฟลชของไมโครคอนโทรลเลอร์ ARM7 ผ่านสาย RS-232 โดยที่องค์ประกอบของซอฟต์แวร์ ARMSWD จะมีไฟล์ 3 ตัวคือ ARMWSD.exe, armwsd.ini และ sample.hex การใช้งานโปรแกรม ARMWSD นั้นไม่ต้องทำการติดตั้งโปรแกรมก็สามารถใช้งานได้ เพียงแต่ต้องให้ไฟล์ทั้ง 3 ทำงานร่วมกันโดยอาจวางไว้ในโฟลเดอร์เดียวกัน วิธีการใช้งานเพียงแค่อีกที่ไฟล์ ARMWSD.exe ก็พร้อมใช้งาน แสดงดังภาพที่ 5.32



ภาพที่ 5.33 หน้าต่างการใช้งานของโปรแกรม ARMWSD

การใช้งานโปรแกรมเริ่มจากคลิกที่ปุ่ม Browse แล้วเลือกไฟล์ที่ต้องการดาวน์โหลดลงไปยังไมโครคอนโทรลเลอร์ ARM7 (ไฟล์ที่มีนามสกุล .hex) จากนั้นคลิกที่ปุ่ม Configure จะมีหน้าต่าง ARMWSD Configuration เพื่อให้กำหนดค่าในการดาวน์โหลด โดยจะมีส่วนที่ต้องกำหนดค่าอยู่ 3 ส่วนคือ Serial Port เลือกพอร์ตให้ตรงกับพอร์ตคอมพิวเตอร์ที่สาย RS-232 ต่ออยู่ในส่วนของ Baud rate ให้เลือกค่า 9600 และในส่วน Flash ให้คลิกเลือกที่ Mass Erase, Program และ Verify แล้วคลิกปุ่ม OK เป็นการเสร็จสิ้นการกำหนดค่า การดาวน์โหลดโปรแกรมทำได้โดยการคลิกที่ปุ่ม Start โปรแกรมก็จะโหลดซอฟต์แวร์ลงไปยังไมโครคอนโทรลเลอร์ แล้วคลิกปุ่ม RUN ไมโครคอนโทรลเลอร์ก็จะเริ่มการทำงาน



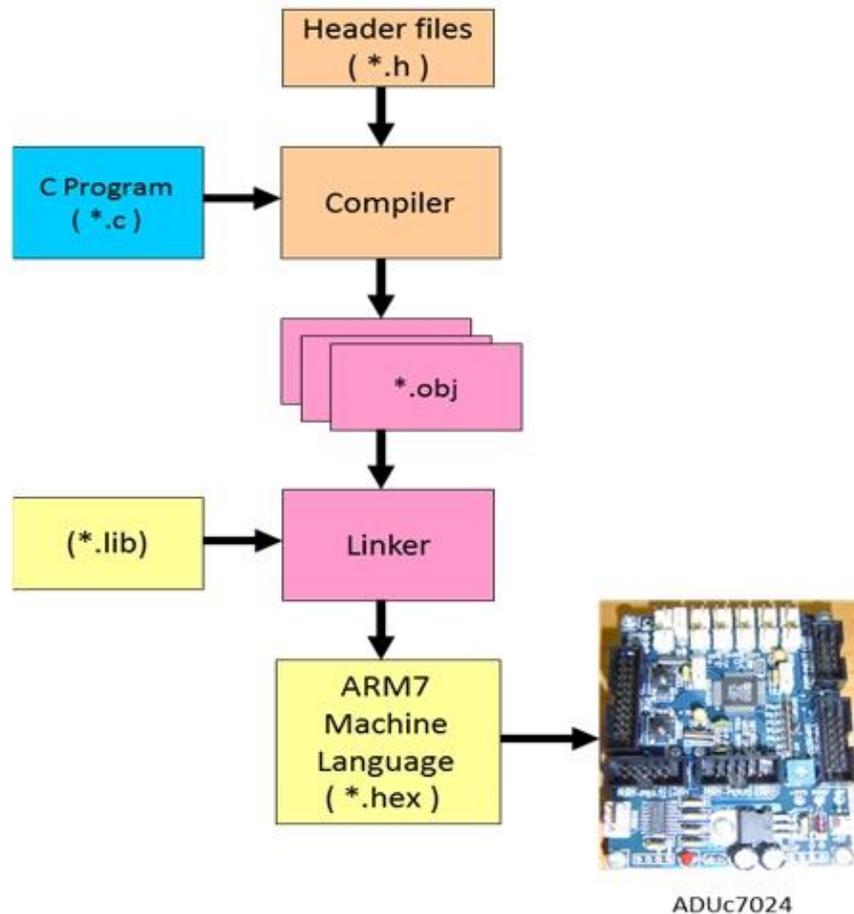
ภาพที่ 5.34 การ Configuration โปรแกรม ARMWSD ก่อนใช้งาน

5.3 พื้นฐานการเขียนภาษาซี (Basic C For ARM7)

5.3.1 ความหมายภาษา C สำหรับ ARM7

ภาษา C สำหรับ ARM7 คือ ภาษาที่ใช้เขียนโปรแกรมเพื่อสั่งงานและควบคุมไมโครคอนโทรลเลอร์ ARM7 ซึ่งภาษา C สำหรับ ARM7 ถือว่าเป็นภาษาระดับกลางมีโครงสร้าง (Structure) และไวยากรณ์ (Syntax) เหมือนกับภาษา C ทั่วไปความหมายภาษา C สำหรับ ARM73

การทำงานของภาษาซีเริ่มจากใช้ Editor เขียนโปรแกรมเมื่อบันทึกเป็นไฟล์นามสกุล *.C จากนั้นทำการคอมไพล์ถ้าหากไม่มีการผิดพลาดจะได้ไฟล์ *.Obj แล้วทำการ Link ไฟล์ *.Obj เข้ากับไฟล์ Libraries หากไม่มีข้อผิดพลาดจะได้ไฟล์ *.EXE



ภาพที่ 5.35 Flow Chart การประมวลผลภาษาซีสำหรับ ARM7

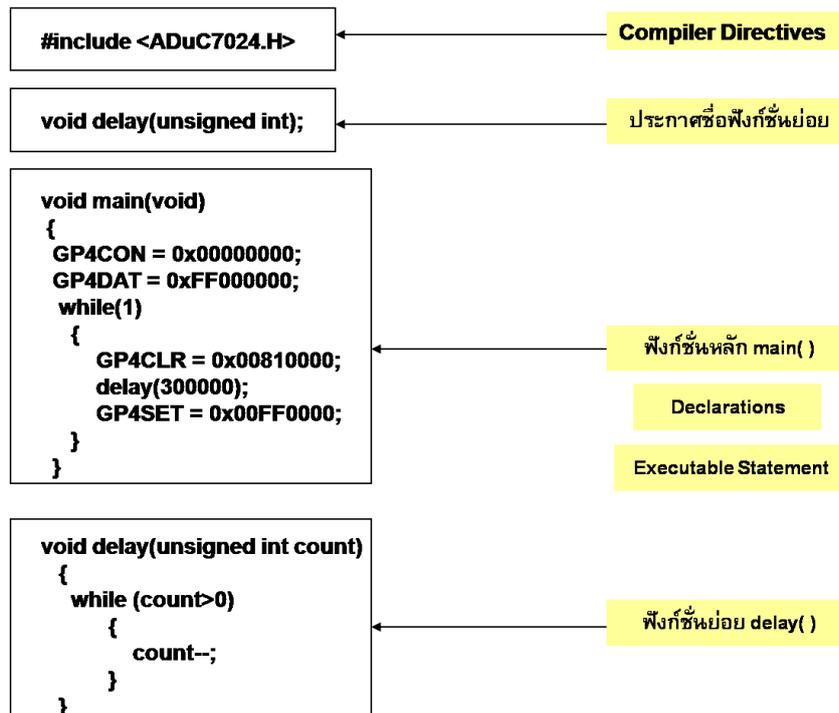
5.3.2 โครงสร้างโปรแกรมภาษา C สำหรับ ARM7

1) Compiler Directive คือ ส่วนที่ใช้บอก Compiler ว่าให้ทำอะไรก่อน ก่อนที่ compiler จะทำการ Compile โปรแกรม เช่น การรวมไฟล์อื่นๆ เข้ากับโปรแกรม การสร้างเงื่อนไขของการ Compile โปรแกรมจะขึ้นต้นด้วยเครื่องหมาย #

2) Functions คือ โปรแกรมย่อย (Sub Program) ชนิดหนึ่งสามารถทำงานได้อย่างใดอย่างหนึ่งเสร็จสิ้นภายในตัวเอง โดยที่ function อาจจะมีการรับค่าเข้า function และส่งกลับในชื่อของ function หรืออาจจะไม่มีก็ได้โดยฟังก์ชันในภาษา C แบ่งได้เป็น 2 ประเภทด้วยกัน คือ ฟังก์ชันที่เราสร้างขึ้นมาเอง และฟังก์ชันที่ภาษาซีสร้างมาเพื่อให้ใช้ในการทำงานทั่วไป อย่างเช่น ฟังก์ชันสำหรับแสดงผล ฟังก์ชันที่ใช้ในการคำนวณ หรือฟังก์ชันสำหรับจัดการกับไฟล์ เป็นต้น

3) ส่วนการประกาศ (Declarations) เป็นส่วนที่ใช้กำหนดประเภทของข้อมูลแบบต่างๆ เช่นการประกาศตัวแปร (Variable) การประกาศค่าคงที่ (Constants) หรือชนิดข้อมูล (Data Type) เป็นต้น

4) ส่วนของคำสั่ง (Executable Statements) เป็นส่วนของประโยคคำสั่ง ที่มีผลให้มีการปฏิบัติการอย่างใดอย่างหนึ่ง ส่วนของคำสั่งนี้มักเป็นส่วนที่อยู่ในขอบเขตของเครื่องหมาย {...}



ภาพที่ 5.36 โครงสร้างโปรแกรมภาษาซี

5.3.3 ตัวแปรชนิดต่าง ๆ และค่าคงที่

ภาษา C สำหรับ ARM7 มีโครงสร้างเหมือนกับภาษา C ทั่วไปทำให้การประกาศตัวแปรและค่าคงที่จึงคล้ายกัน ตัวแปรและค่าคงที่ที่ประกาศไว้จะถูกจองพื้นที่และเก็บค่าไว้ในหน่วยความจำ (RAM)

5.3.4 ชนิดของตัวแปรในภาษาซี

ตารางที่ 5.1 ชนิดของตัวแปรในภาษาซี

ชนิดของตัวแปร	ขนาด	ช่วงของข้อมูล	การกำหนดชนิดของตัวแปร
1. Character	8 บิต	-128 ถึง +127	char
2. Unsigned character	8 บิต	0 ถึง 255	unsigned char
3. Integer	16 บิต	-32768 ถึง +32767	int
4. Unsigned integer	16 บิต	0 ถึง 65535	unsigned int
5. Long integer	32 บิต	-2147483648 ถึง 2147483647	long int
6. Unsigned Long	32 บิต	0 ถึง 4,294,967,295	unsigned long int
7. Floating point	32 บิต	3.4×10^{-38} ถึง $3.4 \times 10^{+38}$	float

1) Character ตัวแปรชนิดตัวอักษรหรือจำนวนเต็ม 8 บิต ใช้เก็บข้อมูลที่เป็นตัวอักษรหรือจำนวนเต็มที่มีค่าอยู่ในช่วง -128 ถึง +127

2) Unsigned character ตัวแปรชนิดตัวอักษรหรือจำนวนเต็ม 8 บิต แบบไม่คิดเครื่องหมายใช้เก็บข้อมูลที่เป็นตัวอักษรหรือจำนวนเต็มบวกที่มีค่าอยู่ในช่วง 0 ถึง 255

3) Integer ตัวแปรชนิดจำนวนเต็มขนาด 16 บิต ใช้เก็บข้อมูลที่เป็นจำนวนเต็มที่มีค่าอยู่ในช่วง -32768 ถึง +32767

4) Unsigned integer ตัวแปรชนิดจำนวนเต็มขนาด 16 บิต แบบไม่คิดเครื่องหมายหรือจำนวนเต็มบวก ใช้เก็บข้อมูลที่เป็นจำนวนเต็มที่มีค่าอยู่ในช่วง 0 ถึง 65535

5) Long integer ตัวแปรชนิดจำนวนเต็มขนาด 32 บิต ใช้เก็บข้อมูลที่เป็นจำนวนเต็มที่มีค่าอยู่ในช่วง -2147483648 ถึง +2147483648

6) Unsigned Long ตัวแปรชนิดจำนวนเต็มขนาด 32 บิต แบบไม่คิดเครื่องหมายหรือจำนวนเต็มบวก ใช้เก็บข้อมูลที่เป็นจำนวนเต็มที่มีค่าอยู่ในช่วง 0 ถึง 4,294,967,295

7) Floating point ตัวแปรชนิดทศนิยม ใช้เก็บข้อมูลเลขทศนิยมที่มีค่าอยู่ในช่วง 3.4×10^{-38} ถึง $3.4 \times 10^{+38}$

ตัวอย่างการประกาศตัวแปร char และ int ARM7 จะไปจองพื้นที่ในหน่วยความจำตามชนิดของตัวแปร คือ char = 1 Byte และ int = 2 Byte

char i;	i	0x10000H
int j;	j	0x10001H
	j	0x10002H
i = 100;	64h	0x10000H
j = 10,000;	10h	0x10001H
	27h	0x10002H

Remark : 100 = 64h, 10,000 = 2710h

ภาพที่ 5.37 ตัวอย่างการประกาศตัวแปร char และ int

5.3.5 การประกาศตัวแปร

ในการเขียนโปรแกรมภาษา C สำหรับ ARM7 การประกาศตัวแปรเพื่อใช้เก็บค่าข้อมูลมีรูปดังนี้

Syntax : ชนิดของตัวแปร ชื่อตัวแปร;

ตัวอย่าง :

char i; สร้างตัวแปรชื่อ i สำหรับเก็บค่าข้อมูล-128 ถึง 127
 int j; สร้างตัวแปรชื่อ j สำหรับเก็บค่าข้อมูล -32,768 ถึง 32,767
 unsigned int count; สร้างตัวแปรชื่อ count สำหรับเก็บค่าข้อมูล 0 ถึง 65,535
 long x; สร้างตัวแปรชื่อ x สำหรับเก็บค่าข้อมูล -2,147,483,648 ถึง 2,147,483,647
 float y; สร้างตัวแปรชื่อ y สำหรับเก็บค่าข้อมูลทศนิยม

การประกาศตัวแปรพร้อมกับค่าเริ่มต้น

Syntax : ชนิดของตัวแปร ชื่อตัวแปร = ค่าเริ่มต้น;

ตัวอย่าง:

unsigned int count = 10,000;

float y = 99.9999;

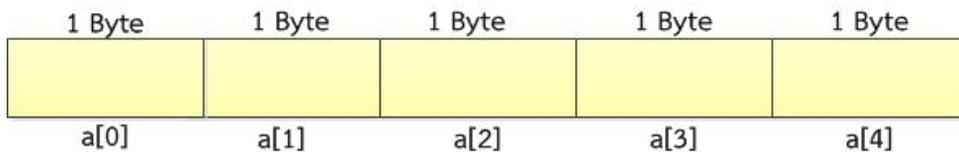
การประกาศตัวแปรแบบค่าคงที่ (Constant)

การประกาศตัวแปรแบบค่าคงที่ เป็นการประกาศตัวแปรพร้อมกับกำหนดค่าให้ตัวนั้นเลย ซึ่งค่าในตัวแปรจะไม่สามารถแก้ไขหรือเปลี่ยนแปลงได้ภายหลัง การประกาศตัวแปรค่าคงที่จะประกาศไว้ตรงส่วนหัวของโปรแกรม

```
#include <ADuC7024.H>
static const float pi =3.14;
static const int y = 5;
```

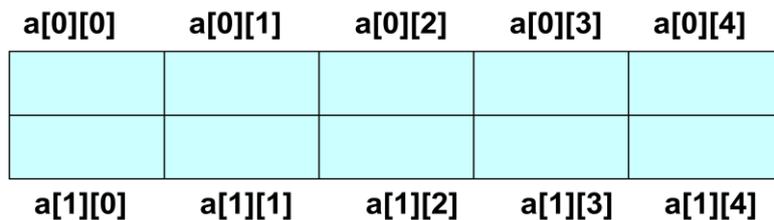
5.3.6 ตัวแปรแบบอาร์เรย์

ตัวแปรแบบอาร์เรย์ (Array) คือ ตัวแปรที่ใช้เก็บข้อมูลแบบชุด หรือกลุ่มข้อมูลที่เป็นชนิดเดียวกัน โดยมีโครงสร้างดังนี้



ภาพที่ 5.38 อาร์เรย์ 1 มิติ

```
unsigned char a[2][5];
```



ภาพที่ 5.39 อาร์เรย์ 2 มิติ

5.3.7 เครื่องหมายตัวดำเนินการ

เครื่องหมายและตัวดำเนินการภาษาซี ที่เรารู้จักและจำเป็นต้องใช้งานอยู่เป็นประจำสามารถแบ่งออกเป็น 5 แบบคือ

ตารางที่ 5.2 เครื่องหมายตัวดำเนินการทางคณิตศาสตร์

เครื่องหมาย	การดำเนินการ	ตัวอย่าง
+	บวก	$a = b + c;$
-	ลบ	$a = b - c;$
*	คูณ	$a = b * c;$
/	หาร	$a = b / c;$
%	หารเอาเศษ	$a = b \% c;$

ตารางที่ 5.3 เครื่องหมายตัวดำเนินการทางด้านการเปรียบเทียบ

เครื่องหมาย	การดำเนินการ	ตัวอย่าง
>	มากกว่า	$a > b$ (ถ้า a มากกว่า b ผลลัพธ์ = จริง)
<	น้อยกว่า	$a < b$ (ถ้า a น้อยกว่า b ผลลัพธ์ = จริง)
>=	มากกว่าหรือเท่ากับ	$a >= b$ (ถ้า a มากกว่าหรือเท่ากับ b ผลลัพธ์ = จริง)
<=	น้อยกว่าหรือเท่ากับ	$a <= b$ (ถ้า a น้อยกว่าหรือเท่ากับ b ผลลัพธ์ = จริง)
==	เท่ากับ	$a == b$ (ถ้า a เท่ากับ b ผลลัพธ์ = จริง)
&&	AND	$a \&\& b$ (ผลลัพธ์จะเป็นจริงก็ต่อเมื่อ a และ b เป็นจริง)
	OR	$a \ \ b$ (ผลลัพธ์จะเป็นจริงก็ต่อเมื่อ a หรือ b เป็นจริง)
!	NOT	$!a$ (เมื่อ a เป็นจริง $!a$ ผลลัพธ์จะเป็นเท็จ)

ตารางที่ 5.4 เครื่องหมายตัวดำเนินการระดับบิต

เครื่องหมาย	การดำเนินการ	ตัวอย่าง
&	AND	$a = b \& 0xF8;$ (1111 1000) ถ้า $b = 1010\ 1010$; a เท่ากับ 1010 1000
	OR	$a = b 0xF8;$ (1111 1000) ถ้า $b = 1010\ 1010$; a เท่ากับ 1111 1010
^	XOR	$a = b \wedge 0xF8;$ (1111 1000) ถ้า $b = 1010\ 1010$; a เท่ากับ 0101 0010
~	Complement	$a = \sim b$ (1111 1000); a เท่ากับ 0000 0111

ตารางที่ 5.5 เครื่องหมายตัวดำเนินการเพิ่มค่าและลดค่า

เครื่องหมาย	การดำเนินการ	ตัวอย่าง
++	เพิ่มค่า	$a = i++;$ นำค่าตัว i ไปเก็บที่ตัวแปร a แล้วบวกค่า i เพิ่มขึ้นหนึ่งค่า $a = ++i;$ บวกค่า i เพิ่มขึ้นหนึ่งค่า แล้วนำค่า i ไปเก็บที่ตัวแปร a
--	ลดค่า	$a = i--;$ นำค่าตัว i ไปเก็บที่ตัวแปร a แล้วลบค่า i ลงหนึ่งค่า $a = --i;$ ลบค่า i ลงหนึ่งค่า แล้วนำค่า i ไปเก็บที่ตัวแปร a

ตารางที่ 5.6 เครื่องหมายตัวดำเนินการเลื่อนบิต (ซ้าย, ขวา)

เครื่องหมาย	การดำเนินการ	ตัวอย่าง
<<	เลื่อนบิตไปทางซ้าย	a << 3 เลื่อนบิตข้อมูลในตัวแปร a ไปทางซ้าย 3 บิต a (0000 0001) ผลลัพธ์ a = 0000 1000
>>	เลื่อนบิตไปทางขวา	a >> b เลื่อนบิตข้อมูลในตัวแปร a ไปทางซ้าย b บิต a (0011 0000), b = 4 ; ผลลัพธ์ a = 0000 0011

ตารางที่ 5.7 ตัวกำหนดชนิดข้อมูลของภาษาซี

ตัวกำหนดชนิดข้อมูล	ความหมายของตัวกำหนดชนิดข้อมูล
%c	แสดงข้อมูลในรูปแบบตัวอักษร
%d	แสดงข้อมูลในรูปแบบตัวเลขจำนวนเต็ม
%e	แสดงข้อมูลในรูปแบบเอ็กซ์โพเนนเชียล
%f	แสดงข้อมูลในรูปแบบตัวเลขทศนิยม
%o	แสดงข้อมูลในรูปแบบตัวเลขฐาน 8
%x	แสดงข้อมูลในรูปแบบตัวเลขฐาน 16
%u	แสดงข้อมูลในรูปแบบของจำนวนเต็มบวก
%s	แสดงข้อมูลในรูปแบบของข้อความ (String)
%p	แสดงข้อมูลในรูปแบบของที่อยู่ของตัวแปร (Address)
%%	แสดงเครื่องหมาย %

การกำหนดชนิดของตัวแปร ควรกำหนดให้สอดคล้องกับการใช้งาน เช่น ถ้ามีการคำนวณเป็นจุดทศนิยม ควรกำหนดชนิดของตัวแปรเป็นแบบ float หรือการกำหนดตัวแปรเพื่อใช้ในการวนรอบที่ไม่เกิน 127 รอบ ควรกำหนดตัวแปรเป็น char เป็นต้น เพื่อลดขนาดของโปรแกรมให้เล็กกลงและทำให้การประมวลผลเร็วขึ้น

5.3.8 รูปแบบการเขียนคำสั่ง

• คำสั่งต้องเขียนด้วยตัวพิมพ์เล็ก ในภาษา C ถือว่าตัวพิมพ์เล็กและตัวพิมพ์ใหญ่ต่างกัน

for(), while(), if..else.., unsigned int count;

• เมื่อจบคำสั่งจะมีต้องลงท้ายด้วยเครื่องหมาย “ ; ” และสามารถเขียนหลายคำสั่งในบรรทัดเดียวกันได้

unsigned char i, j;

i = 0;

j = 0; j++;

• หลังฟังก์ชันไม่ต้องลงท้ายด้วยเครื่องหมาย “ ; ”

```
void delay (unsigned long count)
```

```
{
    while (count > 0)
    {
        count--;
    }
}
```

- ในหนึ่งโปรแกรมจะมีฟังก์ชันก็ได้แต่ต้องมีฟังก์ชันที่ชื่อ main() เสมอ

```
void main(void)
```

```
{
    :
}
```

• การตั้งชื่อตัวแปรใดๆ ในโปรแกรม : เป็นการกำหนดชื่อต่างๆ เช่น ชื่อค่าคงที่ ชื่อโปรแกรม ชื่อฟังก์ชัน ชื่อของประเภทข้อมูล ฯลฯ เพื่อนำไปใช้ในโปรแกรม ซึ่งเป็นชื่อที่ผู้เขียนโปรแกรมกำหนดขึ้นมา โดยการกำหนดชื่อมีกฎเกณฑ์ดังต่อไปนี้

- ประกอบด้วยตัวอักษรตั้งแต่ A ถึง Z หรือตัวเลขตั้งแต่ 0 ถึง 9 ที่ไม่มีสัญลักษณ์พิเศษอื่นๆ อยู่

- เริ่มต้นด้วยตัวอักษรภาษาอังกฤษ (ห้ามเป็นตัวเลข) และตัวถัดไปอาจเป็นตัวเลข ตัวอักษร หรือเครื่องหมาย Underscore (_) ก็ได้

- ตัวอักษรตัวใหญ่กับตัวเล็กจะถือว่าเป็นคนละตัวกัน

```
i, j, count, Count, count1, count2, x_y, _irq
```

```
29x, "xy", +x, y-, 12345
```

• การแทรกคำบรรยาย (Comment) : การเขียนคำบรรยายแทรกไว้ภายในโปรแกรม จะช่วยเพิ่มความเข้าใจของเรา หรือเพื่อให้คนอื่นมาอ่านโปรแกรมได้เข้าใจสิ่งที่เราเขียนด้วย คำบรรยายแทรก โดยใช้เครื่องหมาย /* ข้อความที่ต้องการ */

```
// ข้อความแทรกทั้งบรรทัด
```

```
/* function delay time calling by main function */
void delay(unsigned int count)
{
    while (count>0)
    count--;           // Counter
}
```

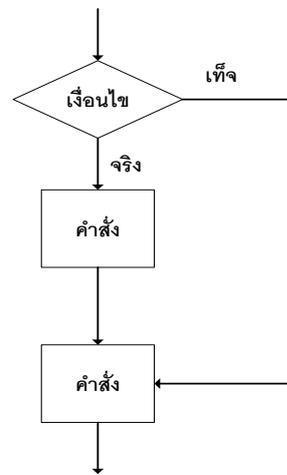
5.3.8.1 คำสั่งเงื่อนไขและการทำซ้ำ

1) คำสั่งเงื่อนไข (Condition)

การเขียนโปรแกรมภาษา c ในบางกรณีจำเป็นต้องให้ ARM7 ตัดสินใจดำเนินการอย่างใดหนึ่ง โดยการกำหนดเป็นเงื่อนไข (Condition) เช่น ถ้าเงื่อนไขเป็น “จริง” ให้ ARM7 ดำเนินการตามชุดคำสั่ง A แต่ถ้าเป็นเงื่อนไขเป็น “เท็จ” ให้ ARM7 ดำเนินการตามชุดคำสั่ง B เป็นต้น คำสั่งกำหนดเงื่อนไขได้แก่ คำสั่ง if, if - else, if - else - if และ Switch-Case

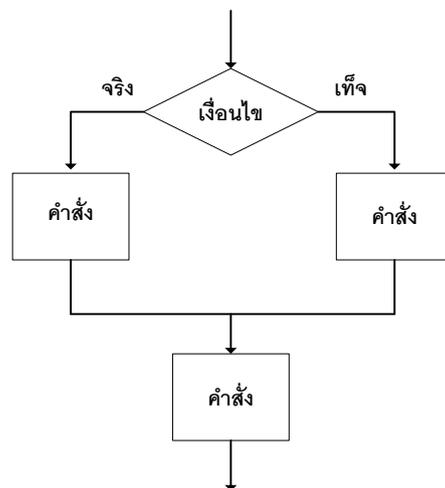
ตัวอย่าง คำสั่ง if

```
if (เงื่อนไข)
{
    คำสั่ง
}
```



ตัวอย่าง คำสั่ง if-else

```
if (เงื่อนไข)
{
    คำสั่ง
}
else
{
    คำสั่ง
}
คำสั่ง
```



ตัวอย่าง คำสั่ง if-else-if

if (เงื่อนไข)

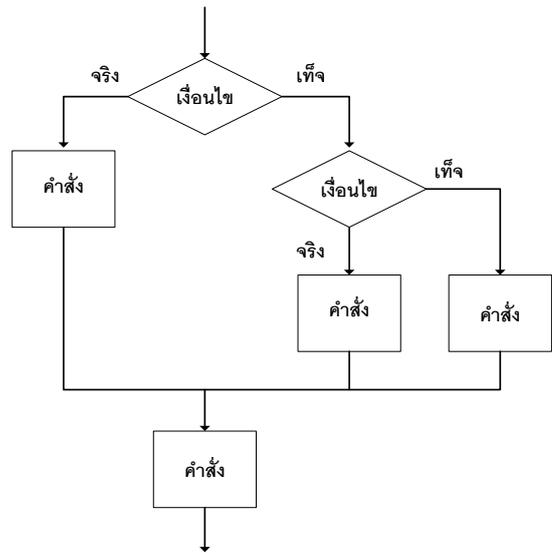
```
{
  คำสั่ง
}
```

else if (เงื่อนไข)

```
{
  คำสั่ง
}
```

else

```
{
  คำสั่ง
}
```



ตัวอย่าง คำสั่ง Switch - Case

switch (เงื่อนไข)

{

case 1 :

```
  คำสั่ง;
  break;
```

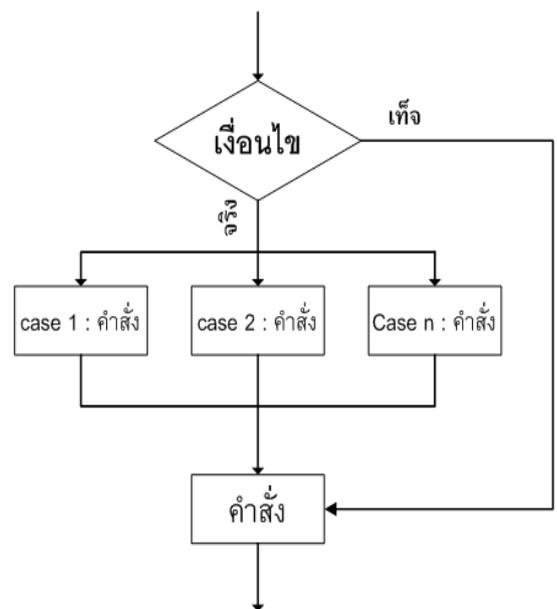
case 2 :

```
  คำสั่ง;
  break;
```

case n :

```
  คำสั่ง;
  break;
```

}



2) การทำซ้ำ (Loop)

การทำซ้ำหรือการทำ Loop เป็นการสั่งให้ ARM7 ทำชุดคำสั่งเดิมซ้ำๆจนครบตามจำนวนรอบที่กำหนด หรือ ตามเงื่อนไขที่กำหนด ซึ่งคำสั่งการทำซ้ำ ได้แก่ for, while และ do - while

ตัวอย่าง คำสั่ง for

คำสั่ง for เป็นการทำซ้ำที่ผู้เขียนโปรแกรมสามารถกำหนดจำนวนครั้งให้กระทำคำสั่งได้ โดยการกำหนดเริ่มต้นให้แก่ตัวแปร Counter และจำนวนรอบที่ต้องการทำซ้ำ เมื่อกระทำซ้ำคำสั่งใน {...} จนครบตามจำนวนรอบก็จะไปทำคำสั่งหลังเครื่องหมาย } (ออกจาก Loop)

```
unsigned int i;
for (i=0;i<100;i++)
{
    คำสั่ง
}
คำสั่ง
```

คำสั่ง while

คำสั่ง while เป็นการทำซ้ำโดยการตรวจสอบเงื่อนไขก่อน ถ้าเงื่อนไขเป็นจริงจะกระทำซ้ำคำสั่งที่อยู่ใน { .. } และหากเงื่อนไขเป็นเท็จจะกระทำคำสั่งหลัง }.. (หลุดออกจาก Loop)

```
unsigned int i;
i = 0;
While (i < 100)
{
    คำสั่ง
    i++;
}
คำสั่ง
```

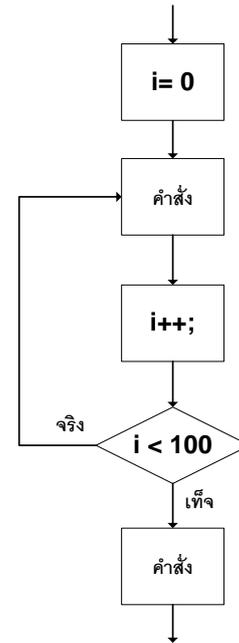
คำสั่ง do - while

คำสั่ง do - while เป็นการทำซ้ำโดยจะกระทำซ้ำคำสั่งที่อยู่ใน { .. } ก่อนแล้วตรวจสอบเงื่อนไขทีหลัง จนกระทั่งเงื่อนไขเป็นเท็จจะกระทำคำสั่งหลัง } (หลุดจาก Loop)

```

unsigned int i;
i = 0;
do
{
    คำสั่ง
    i++;
}
while (i<100);
คำสั่ง

```



5.3.8.2 Standard Function และ User define Function

Standard Function เป็นฟังก์ชันมาตรฐานที่ไมโครคอนโทรลเลอร์ ARM7 ได้เตรียมไว้ให้ผู้ใช้งานสามารถเรียกใช้งานได้ ซึ่งฟังก์ชันมาตรฐานของไมโครคอนโทรลเลอร์ ARM7 (ADuC7024) เช่น <stdio.h>, printf(), scanf(), clear(), getchar() เป็นต้น ส่วน User define Function เป็นฟังก์ชันที่นอกเหนือจาก Standard Function ที่ไมโครคอนโทรลเลอร์ ARM7 ให้มาโดยผู้เขียนโปรแกรมสร้างขึ้นมาใช้งานเองเพื่อตอบสนองความต้องการในการเขียนโปรแกรม ตัวอย่างเช่น

```

void show_menu(void)
{
    printf("***** Main Menu *****\n");
    printf("1. Test ADC 12Bit 10Ch\n");
    printf("2. Test DAC 12Bit 2Ch\n");
    printf("*****\n");
    printf("\n");
    printf( "Select Choice(1...4).....");
}
void main(void)
{

```

บทที่ 6

การออกแบบระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ 4 ถัง

จากอดีตจนถึงปัจจุบันเป็นที่น่าสังเกตได้ว่า มากกว่าครึ่งหนึ่งของตัวควบคุมที่ใช้อยู่ในอุตสาหกรรมเป็นตัวควบคุมแบบพีไอดีแบบดั้งเดิม หรือแบบดัดแปลง ด้วยเหตุผลที่ว่า ตัวควบคุมแบบพีไอดีนั้นมีพฤติกรรมที่ผู้ใช้สามารถเข้าใจได้และมีวิธีการหาค่าพารามิเตอร์ของตัวควบคุมที่หลากหลายวิธี สะดวกในการใช้งานและดัดแปลงไปใช้เช่น ตัวควบคุมแบบ ไอ-พีดี หรือตัวควบคุมแบบสององศาอิสระ แม้ในกรณีที่ไม่สามารถทำการออกแบบระบบควบคุมโดยการวิเคราะห์โมเดลของกระบวนการ ตัวควบคุมแบบพีไอดีก็ยังสามารถใช้งานได้

เนื้อหาในบทนี้เราจะกล่าวถึง กิจการควบคุมแบบพีไอดี การออกแบบระบบควบคุมแบบพีไอดีด้วยเทคนิคทางเดินราก และการออกแบบตัวควบคุมดิคัปปลิ่งสำหรับกระบวนการแบบหลายอินพุตหลายเอาต์พุต เพื่อใช้เป็นแนวทางในการออกแบบระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ 4 ถัง ให้มีผลตอบสนองของระบบมีคุณสมบัติตามที่ต้องการ

6.1 การควบคุมแบบพีไอดี

เป็นการรวมกันของกิริยาควบคุมทั้งแบบ P, I, D ดังสมการ

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \quad (6.1)$$

แสดงการหา transfer function ของระบบควบคุมดังนี้

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (6.2)$$

โดยที่

K_p เป็นค่าอัตราขยายของ P

T_d derivative time

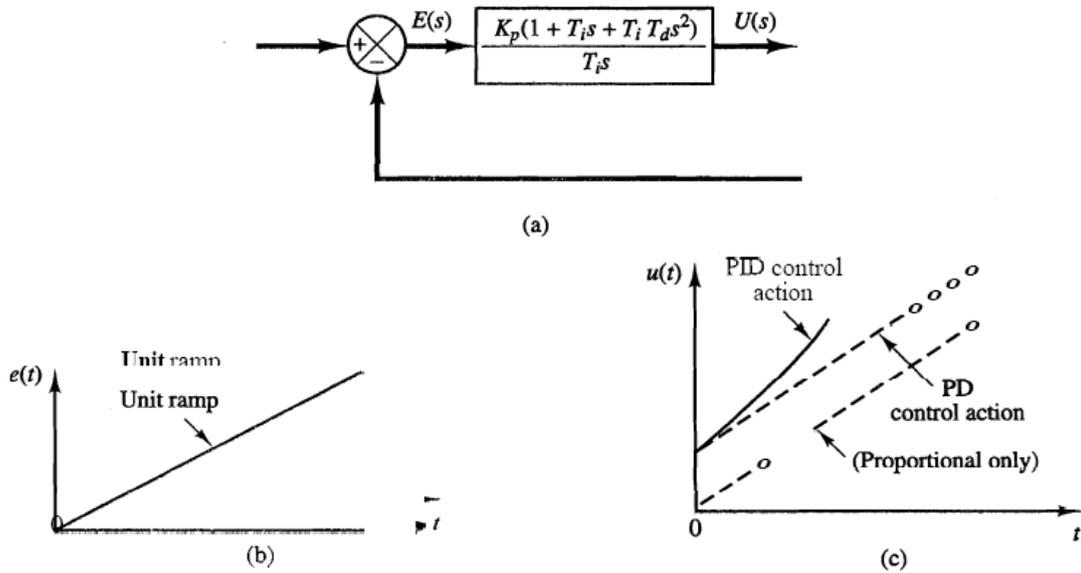
T_i เรียกว่า integral time

ตั้งแต่ dynamic และ static characteristics ของตัวตรวจวัดมีผลกระทบต่อค่าขึ้นของค่าจริงของการเปลี่ยนแปลงของเอาต์พุต ตัวที่กระทำการวัดเป็นบทบาทสำคัญในการแสดงผลตัดสินทั้งหมดของระบบการควบคุม

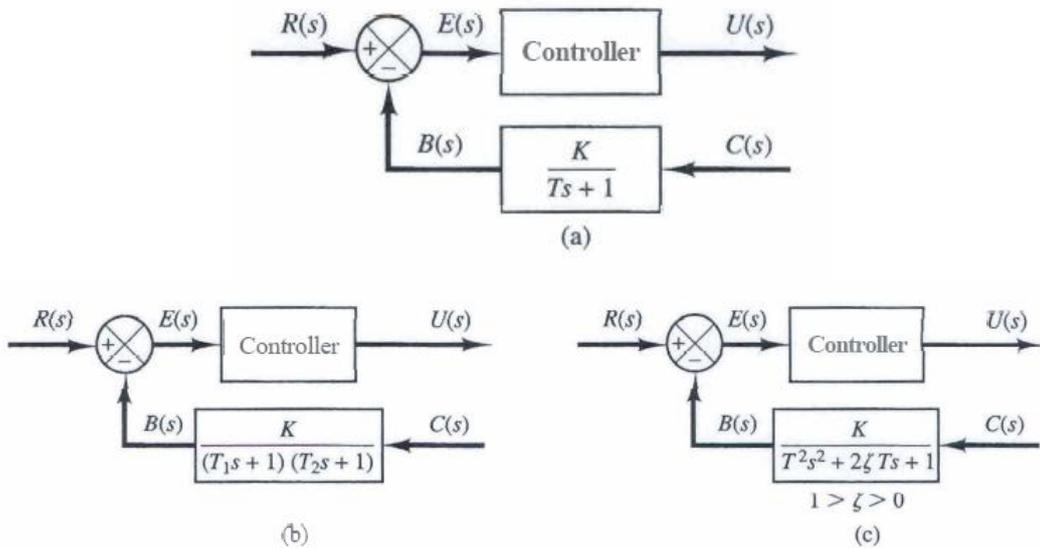
โดยทั่วไปตัวตรวจวัดหาได้จากค่า transfer function ของ feed back path ถ้าค่าคงที่เวลาของตัวตรวจวัดเป็นการเปรียบเทียบที่สามารถมองข้ามได้ของค่าคงที่ค่าเวลาอื่นของระบบการควบคุม กับ transfer function ของตัวตรวจวัดเปลี่ยนแปลงมาเป็นค่าคงที่

รูป 6.2 (a), (b), (c) แสดง block diagram ของการควบคุมอัตโนมัติแบบมีตัวตรวจวัดอันดับหนึ่ง, ตัวตรวจวัดอันดับสองของ over damped, ตัวตรวจวัดอันดับสองแบบ under damped ตามลำดับ

ผลตอบสนองของตัวตรวจวัดอุณหภูมิปกติจะใช้ชนิดวางจรรอันดับสองแบบ over damped



ภาพที่ 6.1 (a) แสดง block diagram ของการควบคุมแบบ P, I, D
 (b) แสดงว่า $e(t)$ เป็นฟังก์ชัน unit ramp
 (c) แสดงการควบคุมเอาต์พุต $u(t)$



ภาพที่ 6.2 (a) block diagram ของระบบควบคุมแบบอัตโนมัติแบบ first order sensor
 (b) block diagram ของระบบควบคุมแบบอัตโนมัติแบบ over damped second order sensor
 (c) block diagram ของระบบควบคุมแบบอัตโนมัติแบบ underdamped second order sensor

6.1.1 ผลกระทบของการควบคุมแบบ I และ D

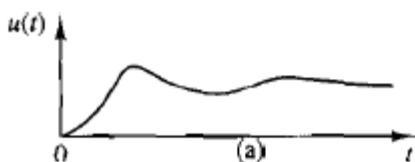
ในบทนี้เราจะศึกษาถึงผลกระทบของการควบคุมแบบ integral และ derivative ต่อประสิทธิภาพของระบบ โดยเราจะศึกษาจากระบบที่มีความซับซ้อนน้อยเพื่อให้สามารถเห็นผลของการควบคุมแบบ integral และ derivative ต่อประสิทธิภาพของระบบได้อย่างชัดเจน

การควบคุมแบบ integral สำหรับการควบคุมโรงงานแบบปฏิกิริยานั้น transfer function จะไม่มี integrator $1/s$ ซึ่งจะทำให้เกิดความผิดพลาดแบบ steady-state หรือ offset ขึ้นจากการตอบสนองกับ step input โดยเราสามารถกำจัดความผิดพลาดแบบ offset ได้โดยการติดตั้งตัวควบคุมเข้าไปเพิ่มในระบบ

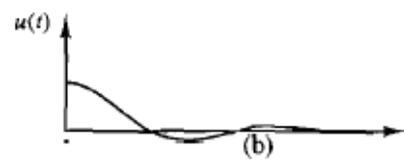
สำหรับระบบควบคุมแบบ integral ของโรงงานนั้นสัญญาณควบคุมและสัญญาณขาออกจากตัวควบคุม สามารถหาได้จากพื้นที่ใต้กราฟของ actuating error ค่าสัญญาณควบคุมไม่จำเป็นต้องเป็นศูนย์ในขณะที่ค่าสัญญาณ actuating error เป็นศูนย์ ดังแสดงในภาพที่ 6.3(a) ปรากฏการณ์ข้างต้นนี้ไม่สามารถเป็นไปได้ในการควบคุมแบบปฏิกิริยาเพราะในการควบคุมแบบปฏิกิริยานั้นค่าสัญญาณควบคุมมีค่ามากกว่าศูนย์จำเป็นจะต้องมีค่าสัญญาณ actuating error มากกว่าศูนย์ด้วย ค่าสัญญาณ actuating error มากกว่าศูนย์ที่สภาวะคงที่หมายความว่า มี offset เกิดขึ้น ภาพที่ 6.3(b) แสดงถึงกราฟความสัมพันธ์ระหว่าง สัญญาณควบคุมกับ สัญญาณ actuating error ของระบบควบคุมแบบปฏิกิริยา

สำหรับการควบคุมแบบ integral นั้นในขณะที่ทำการกำจัดความผิดพลาดแบบ steady-state หรือ offset อาจก่อให้เกิดการตอบสนองแบบแกว่งที่ลดแอมพลิจูดลงอย่างช้าๆ ไปจนกระทั่งเพิ่มแอมพลิจูดซึ่งเราไม่ต้องการให้เกิดปรากฏการณ์ทั้งสองอย่างนี้

การควบคุมแบบ integral ในระบบควบคุมระดับน้ำ เราพบว่าการควบคุมระดับของน้ำด้วยการควบคุมแบบปฏิกิริยาจะให้ความผิดพลาดแบบ steady-state เมื่อมี step input ซึ่งเราพบว่าการกำจัดความผิดพลาดนั้นได้ด้วยการเติมตัวควบคุมเข้าไปในระบบ



ภาพที่ 6.3 (a)



ภาพที่ 6.3 (b)

ภาพที่ 6.3 การควบคุมแบบ integral

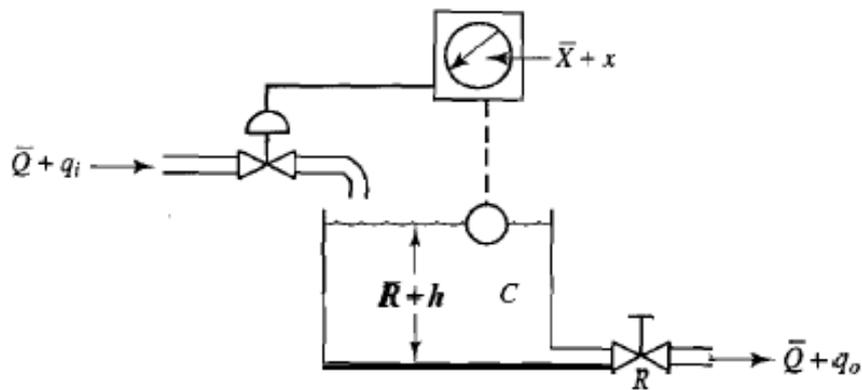
ภาพที่ 6.4 (a) แสดงถึงระบบควบคุมระดับน้ำ เราสมมติว่าตัวควบคุมเป็นแบบ integral และตัวแปร x , q_i , h และ q ซึ่งวัดมาจากตัวแปร \bar{X} , \bar{Q} , \bar{H} และ \bar{Q} ที่สภาวะคงที่ซึ่งมีค่าน้อยมากจนสามารถพิจารณาว่าระบบเป็นเส้นตรง ภายใต้สมมติฐานนี้เราสามารถเขียนแผนผังกล่องของระบบนี้ได้ดัง

ภาพที่ 6.4(b) จากรูป 6.4(b) สามารถแสดงการถ่ายโอนของฟังก์ชันปิดระหว่าง $H(s)$ และ $X(s)$ ได้ดังสมการ

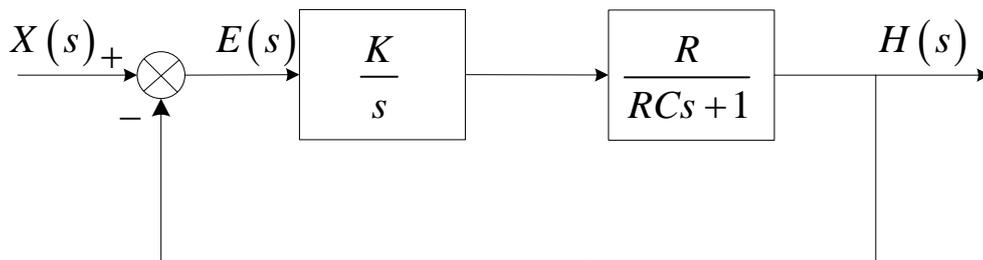
$$\frac{H(s)}{X(s)} = \frac{KR}{RCs^2 + s + KR} \quad (6.3)$$

$$\begin{aligned} \frac{H(s)}{X(s)} &= \frac{X(s) - H(s)}{X(s)} \\ &= \frac{RCs^2 + s}{RCs^2 + s + KR} \end{aligned} \quad (6.4)$$

เมื่อระบบเข้าสู่สมดุลค่าความผิดพลาดที่สภาวะคงที่ของการตอบสนองแบบ unit-step สามารถหาได้จากการใส่ค่าสุดท้ายจากทฤษฎีดังนี้



(a)



(b)

ภาพที่ 6.4 (a) แสดงระบบการควบคุมระดับของเหลว

(b) แสดง block diagram ของระบบ

$$\begin{aligned}
 e_{ss} &= \lim_{s \rightarrow 0} sE(s) \\
 &= \lim_{s \rightarrow 0} \frac{s(RCs^2 + s)}{RCs^2 + s + KR} \cdot \frac{1}{s}
 \end{aligned} \tag{6.5}$$

การควบคุมแบบ integral ในระบบควบคุมระดับน้ำ ที่กำจัดความผิดพลาดที่สภาวะคงที่ ภายใต้การตอบสนองต่อ step input ซึ่งถือว่าการพัฒนาที่สำคัญเมื่อเปรียบเทียบกับการใช้การควบคุมแบบปฏิกิริยาเพียงอย่างเดียวซึ่งจะทำให้เกิด offset

6.2 การออกแบบตัวควบคุมพีไอดีด้วยเทคนิคทางเดินราก

วิธีทางเดินราก (Root Locus) ซึ่งเป็นวิธีที่ศึกษาถึงทางเดินรากของสมการคุณลักษณะของระบบ เมื่อพารามิเตอร์แปรค่าไปจึงเป็นวิธีที่มีประสิทธิภาพวิธีหนึ่งในการศึกษาระบบควบคุม นอกจากนี้เรายังสามารถจะนำวิธี root locus นี้ไปประยุกต์เพื่อหารากของสมการทางพีชคณิตที่มีอันดับสูงๆ ได้อีกด้วย โดยทั่วไปวิธีของ root-locus จะใช้สำหรับพิจารณาทางเดินของรากเมื่อพารามิเตอร์หนึ่งตัวแปรค่าไป แต่ในกรณีที่พารามิเตอร์มากกว่าหนึ่งตัวแปรค่าไปนั้น ก็จะสามารถใช้วิธีของ root locus ได้เช่นกัน

พิจารณาสมการที่มี K เป็นพารามิเตอร์ที่แปรค่าต่อไปนี้

$$F(s) = s^n + a_1s^{n-1} + \dots + a_{n-1}s + a_n + K(s^m + b_1s^{m-1} + \dots + b_{m-1}s + b_m) = 0 \tag{6.6}$$

โดยที่ K เป็นพารามิเตอร์ที่แปรค่า ในที่นี้กำหนดให้ K แปรค่าจาก $-\infty$ ถึง $+\infty$ และจะสมมติให้สัมประสิทธิ์ a_1, \dots, a_n และ b_1, \dots, b_m นั้นมีค่าคงที่ โดยสัมประสิทธิ์นี้อาจจะเป็นค่าจริงหรือจำนวนเชิงซ้อน (complex number) ก็ได้แต่ส่วนใหญ่เราจะพิจารณาถึงสัมประสิทธิ์ที่เป็นค่าจริงเท่านั้น โดยทั่วไปเราจะแบ่งชนิดของทางเดินรากเป็น 3 ชนิด ดังนี้

1. Root loci : root loci เป็นทางเดินรากเมื่อ K ในสมการ (2.63) แปรค่าอยู่ในช่วง 0 ถึง ∞
2. Complementary root loci : บางครั้งเรียกว่า Inverse root loci เป็นทางเดินรากเมื่อ K ในสมการ 2.64 แปรค่าอยู่ในช่วง $-\infty$ ถึง 0
3. ถ้านำทางเดินของราก 2 แบบ ในข้อ 1 และ 2 มารวมกัน จะเรียกว่า root loci ที่สมบูรณ์ (complete root loci)
4. Root contours : root contours เป็นทางเดินของรากเมื่อพารามิเตอร์มากกว่า 1 ตัวแปรค่าไป

6.2.1 สถาปัตยกรรมไมโครคอนโทรลเลอร์ ARM7 (ADuC7024)

พิจารณาระบบเชิงเส้นที่มี transfer function ดังนี้

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} \tag{6.7}$$

สมการคุณลักษณะของระบบหาได้โดยการกำหนดให้เทอมของส่วนในสมการ (6.7) มีค่าเท่ากับศูนย์ นั่นคือ

$$1 + G(s)H(s) = 0 \quad (6.8)$$

สมการ (6.8) เป็นสมการคุณลักษณะของ closed-loop transfer function ของระบบที่มีการป้อนกลับเป็นการป้อนกลับแบบ (negative feedback) โดยมี

$C(s)$ = สัญญาณเอาต์พุต

$R(s)$ = อินพุตอ้างอิง

$G(s)$ = forward-path transfer function

$H(s)$ = feedback-path transfer function

$G(s)H(s)$ = loop transfer function

ถ้าจัดรูปสมการ (6.6) ในอยู่ในรูปสมการ (6.8) จะได้ว่า

$$1 + \frac{K(s^m + b_1s^{m-1} + \dots + b_{m-1}s + b_m)}{s^n + a_1s^{n-1} + \dots + a_{n-1}s + a_n} = 0 \quad (6.9)$$

จากสมการ (6.8) และสมการ (6.9) จะให้ผลดังนี้

$$G(s)H(s) = \frac{K(s^m + b_1s^{m-1} + \dots + b_{m-1}s + b_m)}{s^n + a_1s^{n-1} + \dots + a_{n-1}s + a_n} \quad (6.10)$$

เพื่อที่จะศึกษาถึง root loci ที่สมบูรณ์ โดยที่ K แปรค่าจาก $-\infty$ ถึง $+\infty$ ถ้าเราเขียนสมการ (6.10) ใหม่เป็น

$$G(s)H(s) = KG_1(s)H_1(s) \quad (6.11)$$

จะเห็นว่า $G_1(s)H_1(s)$ ไม่ได้ประกอบด้วยพารามิเตอร์ K ดังนั้นเราจะเขียนสมการ (6.8) ใหม่ได้ ดังนี้

$$G(s)H(s) = -\frac{1}{K} \quad (6.12)$$

เนื่องจาก s เป็นจำนวนเชิงซ้อน ดังนั้นจากสมการ (6.11) จะมีเงื่อนไขดังนี้

$$|G_1(s)H_1(s)| = \frac{1}{|K|} \quad -\infty < K < \infty \quad (6.13)$$

$$\angle G_1(s)H_1(s) = (2K + 1)\pi \quad K \geq 0 \quad (6.14)$$

$$\angle G_1(s)H_1(s) = 2K\pi \quad K \geq 0 \quad (6.15)$$

โดยที่ $k = 0, +1, +2, \dots$

การสร้าง root loci จะเป็นวิธีทางกราฟที่จะเริ่มต้นจากการกำหนดตำแหน่ง pole และ zeros ของ loop transfer function $G(s)H(s)$ ก่อน ดังนั้นขั้นแรกจะต้องเขียนสมการ (6.10) ให้อยู่ในรูปของ poles และ zeros จะได้ว่า

$$G(s)H(s) = \frac{K(s + z_1)(s + z_2)\dots(s + z_m)}{(s + p_1)(s + p_2)\dots(s + p_n)} \quad (6.16)$$

$$= KG_1(s)H_1(s) \quad (6.17)$$

โดยที่ poles และ zeros ในสมการ (6.16) สามารถเป็นไปได้อันค่าจริงหรือจำนวนเชิงซ้อน สมการ (6.13) สมการ (6.14) และสมการ (6.15) สามารถเขียนใหม่ได้ดังนี้

$$|G_1(s)H_1(s)| = \frac{\prod_{i=1}^m |s + z_i|}{\prod_{j=1}^n |s + p_j|} = \frac{1}{|K|} \quad -\infty < K < \infty \quad (6.18)$$

และ

$$\angle G_1(s)H_1(s) = \sum_{i=1}^m \angle s + z_i - \sum_{j=1}^n \angle s + p_j = (2k + 1)\pi \quad 0 \leq K < \infty \quad (6.19)$$

$$\angle G_1(s)H_1(s) = \sum_{i=1}^m \angle s + z_i - \sum_{j=1}^n \angle s + p_j = 2k\pi \quad 0 \leq K < \infty$$

โดยที่ $k = 0, +1, +2, \dots$

เพื่อเป็นตัวอย่างที่จะแสดงถึงการใช้สมการ (6.18) และสมการ (6.19) ในการสร้าง root loci ให้พิจารณา loop transfer function $G(s)H(s)$ หรือ forward-path transfer function ถ้า $H(s) = 1$ ต่อไปนี้

$$G(s)H(s) = \frac{K(s + z_1)}{s(s + p_2)(s + p_3)} \quad (6.20)$$

ภาพที่ 6.5 จะแสดงถึงตำแหน่งของ zero และ pole ที่เราสมมติขึ้นมาเอง จะเป็นจุดที่อยู่บน root loci ($0 < K < \infty$) ได้ก็จะต้องสอดคล้องในสมการ (6.17) และสมการ (6.18) นั่นคือ

$$\frac{|s_1 + z_1|}{|s_1||s_1 + p_2||s_1 + p_3|} = \frac{1}{|K|} \quad (6.21)$$

และ

$$\angle s_1 + z_1 - (\angle s_1 + \angle s_1 + p_2 + \angle s_1 + p_3) = (2K + 1)\pi \quad (6.22)$$

โดยที่ $K = 0, +1, +2, \dots$

ทำนองเดียวกัน จุด s_1 จะเป็นจุดที่อยู่บน Complementary root loci ($-\infty < K \leq 0$) ได้ค่าก็ต่อเมื่อสอดคล้องกับเงื่อนไขในสมการ (6.18) นั่นคือ

$$\angle s_1 + z_1 - (\angle s_1 + \angle s_1 + p_2 + \angle s_1 + p_3) = (2K + 1)\pi \quad (6.23)$$

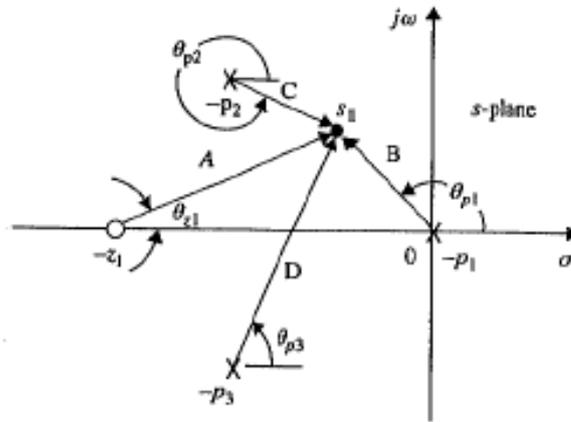
โดยที่ $K = 0, +1, +2, \dots$

จากภาพที่ 6.5 มุม $\theta_{z_1}, \theta_{p_1}, \theta_{p_2}$ และ θ_{p_3} เป็นมุมของ vectors ที่วัดได้โดยมีแกน real หรือแนวนอนเป็นแนวอ้างอิง สมการ (6.22) และสมการ (6.23) จะเขียนใหม่ได้ว่า

$$\theta_{z_1} - (\theta_{p_1} + \theta_{p_2} + \theta_{p_3}) = (2k + 1)\pi \quad 0 \leq K < \infty \quad (6.24)$$

และ

$$\theta_{z_1} - (\theta_{p_1} + \theta_{p_2} + \theta_{p_3}) = 2k\pi \quad -\infty < K \leq 0 \quad (6.25)$$



ภาพที่ 6.5 pole, zero ของ $G(s) = \frac{[K(s + z_1)]}{[s(s + p_2)(s + p_3)]}$

ถ้าจุด s_1 สอดคล้องกับเงื่อนไขของสมการ (6.24) หรือสมการ (6.25) เราก็จะสามารถคำนวณ หาค่าของ K ที่จุด s_1 นั้นได้โดยใช้สมการ (6.21)

$$|K| = \frac{|s_1| |s_1 + p_2| |s_1 + p_3|}{|s_1 + z_1|} \quad (6.26)$$

ถ้าภาพที่ 6.5 นี้ใช้สเกลความยาวของ vector ด้วย a, b และ c ตามลำดับแล้ว จะได้ค่าของ K ดังนี้

$$|K| = \frac{BCD}{A} \quad (6.27)$$

สำหรับเครื่องหมายของ K นั้นจะขึ้นอยู่กับว่าจุด s_1 นั้นอยู่บน root loci หรืออยู่บน complementary root loci ดังที่กล่าวมาแล้ว เราสามารถจะสร้างทางเดินรากที่สมบูรณ์ได้ตามขั้นตอนต่อไปนี้

1. หาจุด s_1 ทุกๆจุดในระนาบ s ที่สอดคล้องกับสมการ (6.14) และสมการ (6.15)
2. ค่าต่างๆ ของ K บน root loci และบน complementary root loci หาได้โดยการใช้สมการ (6.13) โดยที่ K จะเป็นบวกเมื่อจุด s_1 อยู่บน root loci และเป็นลบเมื่อจุด s_1 อยู่บน complementary root loci

6.2.2 การสร้าง ROOT LOCI ที่สมบูรณ์

วิธีของ root loci ได้คิดค้นขึ้นโดย W.R.Evans ในปี ค.ศ.1948 และได้ถูกพัฒนาขึ้นมาเพื่อใช้วิเคราะห์และออกแบบระบบอย่างกว้างขวาง กฎต่างๆ ในการสร้าง root loci ถูกสร้างขึ้นโดยอาศัยความสัมพันธ์ ระหว่าง poles และ zeros ของ $G(s)H(s)$ และ zeros ของสมการคุณลักษณะ

กฎที่ 1 $K=0$

จุดที่ K มีค่าเท่ากับ 0 root loci ที่สมบูรณ์จะอยู่ที่ poles ของ $G(s)H(s)$

พิสูจน์ จากสมการ (6.17)

$$|G_1(s)H_1(s)| = \frac{\prod_{i=1}^m |s + z_i|}{\prod_{j=1}^n |s + p_j|} = \frac{1}{|K|} \quad (6.28)$$

เมื่อ K เข้าใกล้ 0 สมการ (6.28) จะมีค่าเข้าใกล้ ∞ และจะสมนัยกับการที่ s เข้าใกล้ poles ของ $G_1(s)H_1(s)$ นั่นคือ poles ของ $G(s)H(s)$ นั่นคือ s มีค่าเข้าใกล้ $-p_j (j=1,2,\dots,n)$ ใช้ได้ทั้งกับกรณีของ root loci และ complementary root loci เพราะค่าของ K ในกรณีนี้ไม่ได้คิดเครื่องหมาย

กฎที่ 2 $K=\pm\infty$

จุดที่ $K=\pm\infty$ บน root loci ที่สมบูรณ์จะอยู่ที่ zeros ของ $G(s)H(s)$

พิสูจน์ อ้างถึงสมการ (6.28) เมื่อ K เข้าใกล้ $\pm\infty$ สมการ 6.28 จะมีค่าเข้าใกล้ 0 ซึ่งจะสมนัยกับ s เข้าใกล้ zeros ของ $G(s)H(s)$ นั่นคือ s มีค่าเข้าใกล้ $-z_i (i=1,2,\dots,m)$

กฎที่ 3 จำนวนแขนง (branch) ของ root loci ที่สมบูรณ์

ถ้า $n =$ จำนวนของ finite poles ของ $G(s)H(s)$

$m =$ จำนวน finite zeros ของ $G(s)H(s)$

และ $N =$ จำนวนแขนงของ root loci ที่สมบูรณ์
แล้วจะได้ว่า

$$\begin{aligned} N &= n && \text{ถ้า } n > m \\ N &= m && \text{ถ้า } m > n \\ N &= n = m && \text{ถ้า } n = m \end{aligned}$$

กฎที่ 4 การสมมาตรกันของ root loci ที่สมบูรณ์

Root loci ที่สมบูรณ์จะสมมาตรกันเมื่อเทียบกับแกน real ในระนาบ s โดยทั่วไป root loci ที่สมบูรณ์จะสมมาตรกันเมื่อเทียบกับแกนที่ poles และ/หรือ zeros ของ $G(s)H(s)$ สมมาตรกัน

ถ้าหากว่า poles และ/หรือ zeros ของ $G(s)H(s)$ สมมาตรกันกับแกนที่ไม่ใช่แกน real ในระนาบ s แล้ว เราสามารถจะถือว่าแกนที่ poles และ/หรือ zeros ของ $G(s)H(s)$ สมมาตรกัน เป็นแกน real ในระนาบเชิงซ้อนอันใหม่ โดยการใช้วิธี linear transformation และเรียกแกนนี้ว่า axis of symmetry

กฎที่ 5 Asymptotes ของ root loci ที่สมบูรณ์ (พฤติกรรมของ root loci ที่ $s = \infty$)

คุณสมบัติของ root loci ที่สมบูรณ์เข้าใกล้ ในระนาบ s มีความสำคัญอย่างหนึ่ง เนื่องจาก $n = m$ จะมีจำนวนทางเดินของรากเท่ากับ $2|n - m|$ ที่เข้าใกล้ ∞ ระนาบ s ฉะนั้น สำหรับ s ที่มีค่ามากๆมุมของเส้น asymptotes ของ root loci ($K \geq 0$) จะถูกกำหนดโดย

$$\theta_k = \frac{(2k + 1)\pi}{n - m} \quad (6.29)$$

โดยที่ $k = 0, +1, +2, \dots, |n - m| - 1$

$n =$ จำนวนของ finite poles ของ $G(s)H(s)$

$m =$ จำนวนของ finite zeros ของ $G(s)H(s)$

สำหรับมุมของเส้น asymptotes สำหรับ complementary root loci คือ

$$\theta_k = \frac{2k\pi}{n - m} \quad (6.30)$$

โดยที่ k, n และ m จะเหมือนกับกรณีสมการ (6.29)

ถ้าหารสมการ (6.6) ด้วยเทอม

$$s^m + b_1 s^{m-1} + \dots + b_{m-1} s + b_m$$

แล้วจะได้ว่า

$$\frac{s^n + a_1 s^{n-1} + \dots + b_{n-1} s + a_m}{s^m + b_1 s^{m-1} + \dots + b_{m-1} s + b_m} + K = 0 \quad (6.31)$$

จัดรูปสมการ (6.31) และพิจารณาเพียงเทอมที่มี s อันดับสูงเทอมแรก จะได้

$$s^{n-m} + (a_1 + b_1)s^{n-m-1} \cong -K \quad (6.32)$$

หรือ

$$s \left(1 + \frac{a_1 - b_1}{s} \right)^{1/(n-m)} \cong (-K)^{1/(n-m)} \quad (6.33)$$

เทอม $\left[1 + (a_1 - b_1)/s \right]^{1/(n-m)}$ นั้นสามารถกระจายได้โดยวิธีของ binomial expansion ดังนั้นสมการ (6.33) จึงกลายเป็น

$$s \left[1 + \frac{a_1 - b_1}{(n-m)s} + \dots \right] \cong (-K)^{1/(n-m)} \quad (6.34)$$

ทำนองเดียวกัน ถ้าเราพิจารณาเพียง 2 เทอมแรกทางซ้ายมือ จะได้

$$s + \frac{a_1 - b_1}{n-m} \cong (-K)^{1/(n-m)} \quad (6.35)$$

ต่อไปแทน s ด้วย $\sigma + j\omega$ และใช้ทฤษฎีทางพีชคณิตของ De Moivre สมการ (6.35) สามารถเขียนใหม่ได้ดังนี้

$$\sigma + j\omega + \frac{a_1 - b_1}{n-m} \cong (-K)^{1/(n-m)} \left[\cos \frac{(2k+1)\pi}{n-m} + j \sin \frac{(2k+1)\pi}{n-m} \right] \quad (6.36)$$

สำหรับ $0 \leq k < \infty$ และ

$$\sigma + j\omega + \frac{a_1 - b_1}{n-m} \cong \left| K^{1/(n-m)} \right| \left[\cos \frac{2k+1\pi}{n-m} + j \sin \frac{2k\pi}{n-m} \right] \quad (6.37)$$

สำหรับ $-\infty < k \leq 0$ และ $k = 0, \pm 1, \pm 2, \dots$

เทียบ real path และ imaginary ทั้งสองข้างของสมการ (2.94) ให้เท่ากัน เราจะได้ว่า สำหรับ $0 \leq k < \infty$

$$\sigma + \frac{a_1 - b_1}{n-m} \cong K^{1/(n-m)} \cos \frac{(2k+1)\pi}{n-m} \quad (6.38)$$

และ

$$\omega \cong K^{1/(n-m)} \sin \frac{(2k+1)\pi}{n-m} \quad (6.39)$$

จากสมการ (6.38) และสมการ (6.39) หาค่าของ $K^{1/(n-m)}$ ได้ว่า

$$K^{1/(n-m)} \cong \frac{\omega}{\sin \frac{(2k+1)\pi}{n-m}} \cong \frac{\sigma + \frac{a_1 - b_1}{n-m}}{\cos \frac{(2k+1)\pi}{n-m}} \quad (6.40)$$

และ ω มีค่าเป็น

$$\omega \cong \tan \frac{(2k+1)\pi}{n-m} \left(\sigma + \frac{a_1 + b_1}{n-m} \right) \quad (6.41)$$

สมการ (6.41) เป็นสมการเส้นตรงในระนาบ K_1 และอยู่ในรูป

$$\omega \cong M(\sigma - \sigma_1) \quad (6.42)$$

โดยที่ M จะเป็น slope ของเส้นตรงหรือเส้น asymptote และ σ_1 จะเป็นจุดตัดบนแกน σ จากสมการ (6.41) และสมการ (6.42) เราได้

$$M = \tan \frac{(2k+1)\pi}{n-m} \quad (6.43)$$

โดยที่ $k = 0, +1, +2, \dots, |n-m|-1$ และ

$$\sigma_1 = -\frac{a_1 - b_1}{n-m} \quad (6.44)$$

ทำนองเดียวกัน สำหรับกรณีของ Complementary root loci ในสมการ (6.37) เราจะได้ว่า

$$M = \tan \frac{2k\pi}{n-m} \quad , \quad \sigma_1 = -\frac{a_1 - b_1}{n-m} \quad (6.45)$$

โดยที่ $k = 0, +1, +2, \dots, |n-m|-1$

กฎที่ 6 จุดตัดของเส้น Asymptotes (จุด centroid)

ก) จุดตัดของเส้น Asymptotes จำนวน $2|n-m|$ เส้นของ root loci ที่สมบรูณ์จะอยู่บนแกน real ของระนาบ s

ข) จุดตัดของเส้น Asymptote จะถูกกำหนดโดย

$$\sigma_1 = -\frac{b_1 - a_1}{n - m} \quad (6.46)$$

โดยที่ a_1, b_1 กำหนดในสมการ (2.104)

n = จำนวนของ Finite poles ของ $G(s)H(s)$

m = จำนวนของ Finite zeros ของ $G(s)H(s)$

กำหนดฟังก์ชัน $G(s)H(s)$ ตามสมการ (6.10) แล้ว สมการ (6.10) สามารถเขียนใหม่ได้ดังนี้

$$\begin{aligned} \sigma_1 &= -\frac{b_1 - a_1}{n - m} \\ &= \frac{\sum \text{finite poles ของ } G(s)H(s) - \sum \text{finite zeros ของ } G(s)H(s)}{\text{จำนวนของ finite poles ของ } G(s)H(s) - \text{จำนวนของ finite zeros ของ } G(s)H(s)} \end{aligned} \quad (6.47)$$

จากกฎทางพีชคณิต

$$\begin{aligned} -a_1 &= \text{ผลบวกของรากของ } s^n + a_1s^{n-1} + \dots + a_{n-1}s + a_n = 0 \\ &= \text{ผลบวกของ finite poles ของ } G(s)H(s) \end{aligned} \quad (6.48)$$

$$\begin{aligned} -b_1 &= \text{ผลบวกของรากของ } s^m + b_1s^{m-1} + \dots + b_{m-1}s + b_m = 0 \\ &= \text{ผลบวกของ finite zeros ของ } G(s)H(s) \end{aligned} \quad (6.49)$$

จากนี้สมการ (6.47) นั้นสามารถจะใช้ได้กับ root loci และ complementary root loci เพราะค่า σ_1 ของ root loci และ complementary root loci มีค่าเท่ากัน

กฎที่ 7 Root loci บนแกน real

- ก) Root loci : บนส่วนต่างๆ ที่กำหนดให้ของแกน real จะพบว่ามี root loci ($K \geq 0$) อยู่ในส่วนนั้นๆ ถ้าจำนวนรวมกันของ poles และ zeros ของ $G(s)H(s)$ อยู่บนแกน real ซึ่งอยู่ทางขวามือของส่วนที่พิจารณานั้นมีจำนวนเป็นคี่
- ข) Complementary root loci : บนส่วนที่กำหนดให้ของแกน real จะพบว่ามี complementary root loci ($K \leq 0$) อยู่ในส่วนนั้นถ้าหากว่าจำนวนรวมกันของ poles และ zeros ของ $G(s)H(s)$ ที่อยู่บนแกน real ซึ่งอยู่ทางขวามือของส่วนที่พิจารณานั้นมีจำนวนเป็นคู่ หรือเราอาจกล่าวได้ว่าจะมี complementary root loci ในส่วนต่างๆบนแกน real ที่ไม่มี root loci

สำหรับ complex poles และ complex zeros ของ $G(s)H(s)$ จะไม่เกี่ยวข้องกับการพิจารณาหา root loci หรือ complementary root loci บนแกน real ไม่ว่ากรณีใดๆ

กฎที่ 8 มุมที่ออกจาก Pole (Angle of departure) และมุมที่เข้าหา zero (Angle of arrival) ของ root loci ที่สมบูรณ์

มุมเหล่านี้จะหาได้โดยใช้สมการ (6.18) ตัวอย่างเช่นถ้าต้องการที่จะหามุมที่ root loci ออกจาก pole ที่จุด $-1 + j1$ มุมที่เราต้องการ และสอดคล้องกับสมการ (6.18) นั่นคือ

$$\angle G(s_1)H(s_1) = -(\theta_1 + \theta_2 + \theta_3 + \theta_4) = (2k + 1)180^\circ \quad (6.50)$$

หาได้โดยใช้สมการ (6.14) สมการ (6.50) จึงกลายเป็น

$$-(135^\circ + \theta_2 + 90^\circ + 26.6^\circ) = (2k + 1)180^\circ \quad (6.51)$$

กำหนดให้ $k=0$ จะได้

$$\theta_2 = -431.6^\circ$$

หรืออาจจะกล่าวได้ว่า

$$\theta_2 = -71.6^\circ$$

เมื่อหามุมของ root loci ที่ออกจาก pole หรือเข้าหา zero ของ $G(s)H(s)$ ได้แล้ว มุมของ complementary root loci ที่จุดเดียวกันจะมีค่าต่างไปจากมุมที่หาได้ที่จุดนั้นๆ เท่ากับ 180 องศา โดยใช้สมการ (6.19)

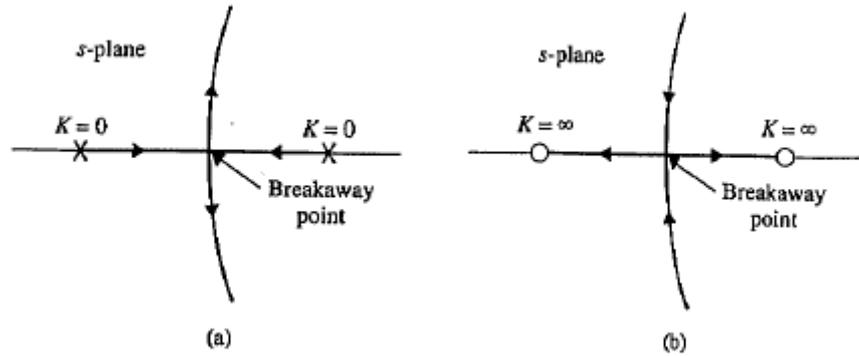
กฎที่ 9 จุดตัดของ Root locus กับแกน imaginary

จุดที่ Root loci ที่สมบูรณ์ตัดกับแกน imaginary และค่าของ K ณ.ที่จุดตัดนี้สามารถหาได้โดยการใช่วิธี Routh-Hurwitz criterion

สำหรับกรณีที่มีจุดตัดหลายจุด ค่าของ K และ จะหาได้ง่ายกว่าโดยวิธีของ Bode diagram

กฎที่ 10 วิธีการหาจุด Breakaway (จุด Saddle) บน root loci ที่สมบูรณ์

จุด breakaway หรือจุด saddle บน root loci ของสมการใดๆ หมายถึงว่า สมการนั้นๆ จะมีรากซ้ำ (multiple-order roots) ภาพที่ 6.6 (a) แสดงถึงกรณีที่ root loci จำนวน 2 แขนงมาพบกันที่จุด breakaway บนแกน real จากนั้นก็แยกออกจากกันไปในทิศทางตรงกันข้ามตามแกน สำหรับภาพที่ 6.6 (b) ก็แสดงถึงจุด breakaway อีกกรณีหนึ่ง

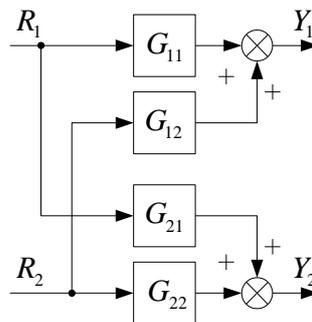


ภาพที่ 6.6 ตัวอย่างจุด breakaway บนแกน real ในระนาบ s

6.3 การออกแบบตัวควบคุมติคัปปลิ่งด้วยสำหรับกระบวนการแบบหลายอินพุตหลายเอาต์พุต

ในการศึกษาระบบควบคุมโดยทั่วไปเราจะศึกษาเพียงระบบที่เป็นแบบ SISO เท่านั้น บ่อยครั้งที่พบว่าหลายๆ กระบวนการที่พบในการทำงานนั้นเป็นระบบที่ต้องการการควบคุมตัวแปรในกระบวนการนั้นมากกว่าหนึ่ง หรือกล่าวอีกนัยหนึ่งคือจุดมุ่งหมายของการควบคุมมีมากกว่าหนึ่งนั่นเอง กระบวนการเหล่านี้สามารถพิจารณาแบบ SISO ได้ตราบดีที่องค์ประกอบภายในระบบไม่มีการอินเตอร์แอคชัน(interaction) ภายในซึ่งกันและกัน การติคัปปลิ่ง(Decoupling) ถูกนำมาใช้ออกแบบระบบของกระบวนการเพื่อช่วยให้สามารถแยกระบบของกระบวนการแบบ MIMO ให้เป็นระบบ SISO สำหรับการออกแบบระบบควบคุมที่ใช้เทคนิคการออกแบบตัวควบคุมโดยวิธีการกำหนดอัตราส่วนคุณลักษณะ CRA ซึ่งมีรายละเอียดดังต่อไปนี้

หลักการทำงานของวิธีการดังกล่าวจะเป็นการนำชุดติคัปปลิ่งมาต่อร่วมกับทรานส์เฟอร์เมทริกของกระบวนการเพื่อลดผลกระทบของการอินเตอร์แอคชันระหว่างลูปทรานด์เฟอร์ฟังก์ชันจะถูกกำจัดไปพิจารณากระบวนการแบบ TITO ดังภาพที่ 6.7



ภาพที่ 6.7 โครงสร้างของกระบวนการแบบ TITO

โดยเมทริกฟังก์ชันถ่ายโอนของกระบวนการคือ

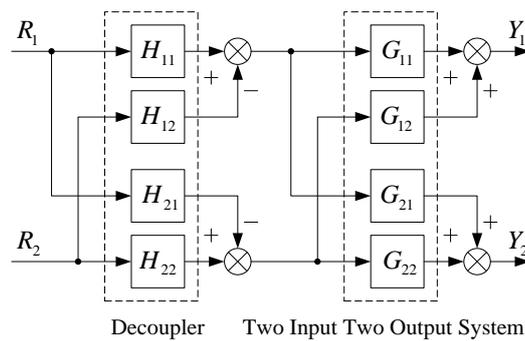
$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \quad (6.52)$$

เมื่อพิจารณาจากองค์ประกอบของเมทริกซ์ฟังก์ชัน $G_{11}(s), G_{22}(s)$ เป็นรูปของทรานส์เฟอร์ฟังก์ชันของ $R_1 - Y_1$ และ $R_2 - Y_2$ และมี $G_{12}(s), G_{21}(s)$ เป็นการคัปปลิงระหว่างลูปของ ทรานส์เฟอร์ฟังก์ชัน ผลของการคัปปลิงนี้เราเรียกว่า Interaction ที่มีผลกระทบต่อระบบควบคุมได้หลายลักษณะ ในกรณีนี้ผลกระทบดังกล่าวสามารถกำจัดหรือลดผลกระทบได้โดยการใช้เทคนิคการดีคัปปลิง วิธีการดีคัปปลิงใช้หลักการจัดรูปแบบทรานส์เฟอร์เมทริกซ์ฟังก์ชันของระบบให้เป็นเมทริกซ์แนวทแยงโดยการใช้ดีคัปปลิงเมทริกซ์ ภาพที่ 3.10 แสดงถึงโครงสร้างของดีคัปปลิงที่ประกอบด้วยกระบวนการแบบ TITO ดีคัปปลิงเมทริกซ์สามารถแสดงได้ดังสมการที่ 3.40

$$H(s) = \begin{bmatrix} H_{11}(s) & H_{12}(s) \\ H_{21}(s) & H_{22}(s) \end{bmatrix} \quad (6.53)$$

เมื่อใช้ดีคัปปลิงทรานส์เฟอร์เมทริกซ์ร่วมกับกระบวนการ TITO ทรานส์เฟอร์เมทริกซ์ทั้งระบบสามารถจัดรูปแบบเมทริกซ์แนวทแยงดังต่อไปนี้

$$\begin{bmatrix} G_{n1} & 0 \\ 0 & G_{n2} \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \quad (6.54)$$



ภาพที่ 6.8 โครงสร้างระบบควบคุมสำหรับกระบวนการระดับน้ำแบบ 4 ถังที่ประกอบด้วยชุดดีคัปปลิง

เมื่อพิจารณาจากการออกแบบชุดดีคัปปลิงจะสามารถกำหนดให้ H_{11} และ $H_{22} = I$ สำหรับ H_{12} และ H_{21} หาได้จากสมการ (6.55), (6.56) ซึ่งได้ดังต่อไปนี้

$$H_{12} = -\frac{G_{12}(s)}{G_{11}(s)} \quad (6.55)$$

$$H_{21} = -\frac{G_{21}(s)}{G_{22}(s)} \quad (6.56)$$

จากทรานส์เฟอร์ฟังก์ชัน H_{11}, H_{12}, H_{21} และ H_{22} จะสามารถจัดรูปแบบระบบแบบกระจายส่วนได้ดังสมการต่อไปนี้

$$Y_1 = \left(G_{11}(s) - \frac{G_{21}(s)}{G_{22}(s)} G_{12}(s) \right) R_1 - 0R_2 = G_{n1}(s)R_1 \quad (6.57)$$

$$Y_2 = 0R_1 - \left(G_{22}(s) - \frac{G_{12}(s)}{G_{11}(s)} G_{21}(s) \right) R_2 = G_{n2}(s)R_2 \quad (6.58)$$

เมื่อพิจารณาจากสมการ (6.57) และ (6.58) จะเห็นได้ว่าผลของการเกิดอินเตอร์แอคชั่นระหว่าง ลูปทรานด์เฟอร์ฟังก์ชันจะถูกกำจัดไป ระบบใหม่ที่ได้จะมีลักษณะการกระจายส่วนซึ่งสะดวกในการควบคุม และสามารถนำมาออกแบบระบบควบคุมในลักษณะตัวควบคุมแบบหนึ่งอินพุตหนึ่งเอาต์พุต SISO ได้

เอกสารอ้างอิง

- [1] K. H. Johansson, “The Quadruple-Tank Process: A Multivariable Laboratory Process with an Adjustable zero”, IEEE Trans. on Control Systems Technology, 8, 456-465, 2000.
- [2] Johansson, K.H; Horch,A; Wijk,O; Hansson,A. “Teaching multivariable control using the quadruple-tank process.” Conference on Proceeding of th 38th IEEE, 1999.
- [3] J. L. R. Nunes. “Modeling and control of the Quadruple-Tank Process. Master thesis, Department of Automatic Control”, Lund Institute of Technology, Sweden, 1997.
- [4] V. Recica. “Automatic tuning of multivariable controllers”. Master thesis, Department of Automatic Control, Lund Institute of Technology, Sweden, 1998.
- [5] M. Grebeck. “A comparison of controllers for the quadruple tank system. Technical report, Department of Automatic Control”, Lund Institute of Technology, Sweden, 1998.
- [6] Arjin Numsomran,Tianchai Suksri, Viriya Kongratana, Thanit Trisuwannawat and Kitti Tirasesth, “I-P Controller Design for Quadruple-Tank System”, TENCON 2004
- [7] Tianchai Suksri, U-thai Sritheeravirojana, Arjin Numsomran, Viriya Kongrattana, and Thongchai Werataweemart,”T-DOF PID Controller Design using Characteristic Ratio Assignment Method for Quadruple Tank Process”,Transactions on Engineering, Computing and Technology, 2006.
- [8] Arjin Numsomran, Tianchai Suksri, Maitree Thumma ,”Design of Decoupled Controller for TITO System using Characteristic Ratio Assignment”, ICCAS 2007
- [9] Arjin Numsomran, Vittaya Tipsuwanporn, Kitti Tirasesth, “Modeling of the Modified Quadruple-Tank Process”, SICE2008
- [10] Arjin Numsomran, Vittaya Tipsuwanporn,Thanit Trisuwannawat and Kitti Tirasesth, “Design PID Controller for the Modified Quadruple-Tank Process using Root Locus”, ICCAS2008
- [11] Tianchai.Suksri, Arjin Numsomran, Vittaya Tipsuwanporn, Thanit Trisuwannawat and Kitti Tirasesth, “Design of PI Controller Using Decoupling and CRA Techniques for Quadruple-Tanks Process”, ICCAS2011
- [12] Arjin Numsomran, Vittaya Tipsuwanporn, Thanit Trisuwannawat and Kitti Tirasesth, “Design of PID Controller for the Modified Quadruple-Tank Process using Inverted Decoupling Technique”, ICCAS2011

ภาคผนวก

ใบงานการทดลอง

ใบงานการทดลองสำหรับกระบวนการระดับน้ำ 4 ถึง

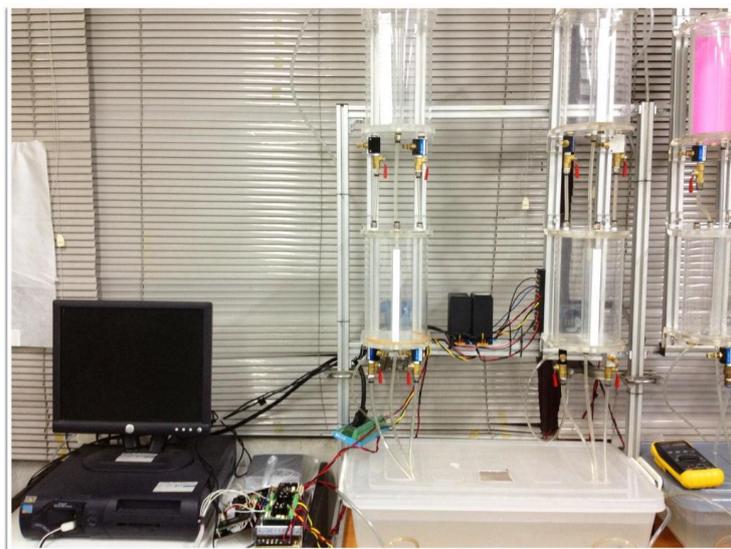
จากข้อมูลรายละเอียดอุปกรณ์ต่างๆของกระบวนการระดับน้ำ 4 ถึง โมเดลทางคณิตศาสตร์ พื้นฐานความรู้การใช้งานโปรแกรม Matlab การเขียนโปรแกรมภาษาซีสำหรับหน่วยประมวลผล ARM7 และ ทฤษฎีการออกแบบระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ 4 ถึง ในบทที่ 2, 3, 4, 5 แล้วนั้น เพื่อให้นักศึกษาสามารถเข้าใจพฤติกรรมของกระบวนการ หลักการทางด้านการวัดและ ออกแบบระบบควบคุม และส่งเสริมทักษะในการปฏิบัติงาน การใช้อุปกรณ์และเครื่องมือต่างๆ คณะผู้วิจัยจึงได้จัดทำใบงานการทดลองสำหรับกระบวนการระดับน้ำ 4 ถึง โดยรวบรวมรายละเอียดไว้ในภาคผนวกนี้ ซึ่งประกอบด้วย 2 ใบงานการทดลอง และหัวข้อการปฏิบัติการทดลอง ดังต่อไปนี้

การทดลอง CS01 การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ

- การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการด้วยวิธีการวิเคราะห์คุณลักษณะทางกายภาพ
- การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการด้วยวิธีการวิเคราะห์จากผลตอบสนองของลูปเปิดในโดเมนเวลา
- การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการด้วยวิธีการวิเคราะห์จากผลตอบสนองของลูปเปิดในโดเมนความถี่

การทดลอง CS02 การออกแบบระบบควบคุมพีไอดี ด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ

- การทดลองออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ 2 ถึง แบบอนุกรม
- การทดลองออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ 4 ถึง



รูปที่ 1 กระบวนการระดับน้ำ 4 ถึง

การวิเคราะห์และออกแบบระบบควบคุมกระบวนการระดับน้ำ 4 ถัง
ห้องปฏิบัติการระบบควบคุม
สาขาวิชาวิศวกรรมการวัดและควบคุม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

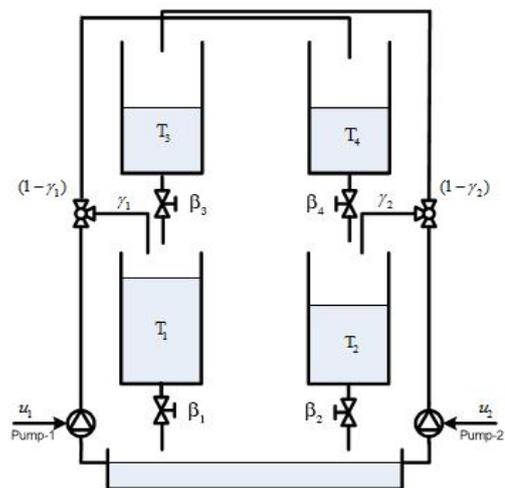
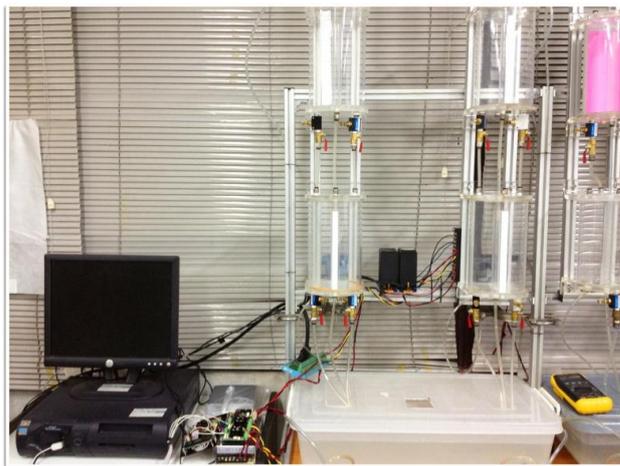
การทดลอง CS01 การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ

วัตถุประสงค์

เพื่อทดลองหาแบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ 4 ถัง ด้วยเทคนิควิธีต่างๆ พร้อมทั้งวิเคราะห์คุณลักษณะของกระบวนการ

บทนำ

กระบวนการระดับน้ำ 4 ถัง เป็นกระบวนการในการศึกษาและการทดลองแบบใหม่ซึ่งถูกแนะนำโดย K.H.Johansson [1] ซึ่งกระบวนการนี้ได้ถูกออกแบบมาเพื่อทำการอธิบายถึงขีดจำกัดสมรรถนะของระบบควบคุมแบบ MIMO อันเนื่องมาจากตำแหน่งของซีโรของกระบวนการ โดยกระบวนการนั้นจะประกอบด้วยถังน้ำ 4 ใบและปั้มน้ำ 2 ตัว ซึ่งอินพุตที่เข้าสู่กระบวนการคือค่าแรงดันที่จ่ายให้กับปั้มน้ำทั้ง 2 ตัวและเอาต์พุตของกระบวนการคือ ระดับน้ำของถังน้ำด้านล่างทั้ง 2 ใบ เมื่อทำการวิเคราะห์แบบจำลองเชิงเส้นของกระบวนการระดับน้ำ 4 ถัง จะพบว่าตำแหน่งซีโรของกระบวนการนั้นสามารถปรับตำแหน่งให้อยู่ได้ทั้งด้านซ้ายและด้านขวาของระนาบ s โดยการปรับเปลี่ยนค่าตำแหน่งขั้ว ซึ่งก็คือกระบวนการนี้สามารถเลือกโหมดการทำงานให้เป็นแบบเฟสต่ำสุดหรือเฟสไม่ต่ำสุดนั่นเอง



รูปที่ 2 โครงสร้างกระบวนการระดับน้ำ 4 ถัง

สำหรับการหาแบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ 4 ถัง นั้น สามารถกระทำได้โดยการประยุกต์ใช้ 3 เทคนิควิธี ดังนี้

1. วิธีการวิเคราะห์คุณลักษณะทางกายภาพด้วยสมการสมดุลมวลและกฎของเบอร์นูลลี (Bernoulli's) เป็นแนวทางการหาแบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ 4 ถัง โดยอ้างอิงถึงหลักการสมดุลของมวลและกฎของเบอร์นูลลี “อัตราการไหลของน้ำเข้าสู่ระบบ เท่ากับ อัตราการไหลของน้ำออกจากระบบ” และทำการหาค่าพารามิเตอร์ต่างๆ ในสมการ เช่น ค่าความต้านทานการไหลของวาล์ว ค่าอัตราขยายของปั้มน้ำ ค่าความสูงคงที่ของระดับน้ำ ณ จุดทำงาน เพื่อคำนวณหาทรานเฟอร์ฟังก์ชันของกระบวนการด้วยการแปลงลาปลาซ

2. วิธีการวิเคราะห์จากผลตอบสนองของลูปเปิดในโดเมนเวลา เป็นวิธีการหาแบบจำลองของกระบวนการจากการทดลอง โดยทำการป้อนสัญญาณอินพุตแบบขั้นบันไดให้แก่ระบบเปิด เพื่อบันทึกผลตอบสนอง และทำการวิเคราะห์หาแบบจำลองทางคณิตศาสตร์ของกระบวนการ

3. วิธีการวิเคราะห์จากผลตอบสนองของลูปเปิดในโดเมนความถี่ เป็นวิธีการหาแบบจำลองของกระบวนการจากการทดลอง โดยทำการป้อนสัญญาณอินพุตที่มีขนาดคงที่และแปรความถี่ให้แก่ระบบเปิดของกระบวนการพลวัตที่ต้องการทดสอบ เพื่อบันทึกผลตอบสนองทางความถี่ และทำการวิเคราะห์หาแบบจำลองทางคณิตศาสตร์ของกระบวนการ

อุปกรณ์

- ชุดทดลองกระบวนการระดับน้ำ 4 ถัง
- เครื่องคอมพิวเตอร์พร้อมโปรแกรมใช้งาน

การทดลอง

การทดลอง 1 การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการด้วยวิธีการวิเคราะห์คุณลักษณะทางกายภาพ

วัตถุประสงค์ เพื่อศึกษาการหาแบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ 4 ถัง ด้วยวิธีการวิเคราะห์คุณลักษณะทางกายภาพ

ขั้นตอนที่ 1 : ทำการกำหนดตำแหน่งวาล์วของกระบวนการระดับน้ำ 4 ถัง ให้มีโครงสร้าง Schematic diagram ตามภาพที่ 3 (กระบวนการระดับน้ำหนึ่งถัง 1 อินพุต 1 เอาต์พุต)

ขั้นตอนที่ 2 : หาอัตราการขยายของปั้มน้ำ K_{p1}

เมื่อ A_1 คือ พื้นที่หน้าตัดด้านในของถังน้ำ $= 176.625 \text{ cm}^2$

A_2 คือ พื้นที่หน้าตัดด้านนอกของท่อน้ำทิ้ง $= 7.065 \text{ cm}^2$

a คือ พื้นที่หน้าตัดของเอาต์พุต $= 0.1963 \text{ cm}^2$

A คือ พื้นที่หน้าตัดจริงของถังน้ำ $= 169.56 \text{ cm}^2$

ป้อนแรงดันอินพุต (u_1) = 1 V และจับเวลาจนกระทั่งน้ำมีความสูงที่ระดับ $h = 10 \text{ cm}$

ใช้เวลา $t = \underline{\hspace{2cm}}$ วินาที

$$\text{คำนวณหาค่าปริมาตร } C = A \times h = \underline{\hspace{2cm}} \text{ cm}^3 \quad (1)$$

$$\text{หาค่าอัตราการไหล } Q = C / t ; \text{cm}^3 / \text{s} = \underline{\hspace{2cm}} \text{ cm}^3 / \text{s} \quad (2)$$

$$\text{หาค่าอัตราขยายของปั้มน้ำ } K_{p1} = \frac{Q}{U} = \underline{\hspace{2cm}} \text{ cm}^3 / \text{V} \cdot \text{S} \quad (3)$$

ขั้นตอนที่3 : หาค่าความต้านทานการไหลของวาล์ว β_1

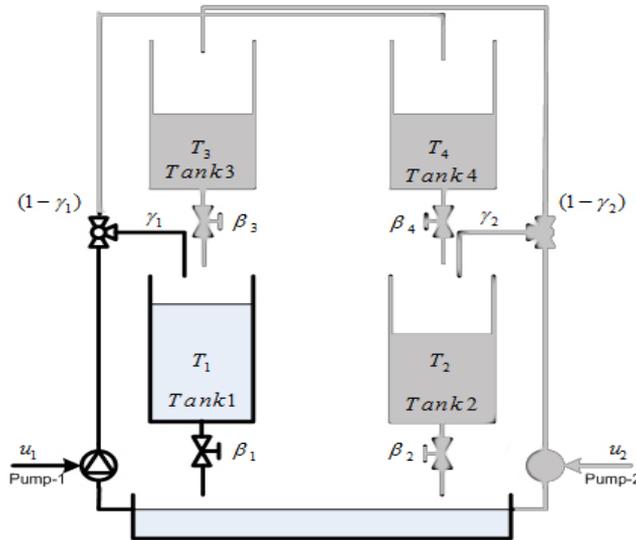
ป้อน u_1 คงที่ รอกันได้ระดับหนึ่งที่จุดทำงานเหมาะสม บันทึกผลการทดลอง กรณี $\gamma_1=0$ ดังนั้นรู้ k_{p1} , u_1 , A , a , h_1 จึงคำนวณหา β_1 ได้ [***h (cm) , a,A(cm²), k(cm³/(Volt/sec)), g คือค่าแรงโน้มถ่วงของโลก (cm/s²)=981

$$\beta_1 = \frac{k_{p1}u_1(t)}{a_1\sqrt{2gh_1(t)}} = \text{_____} \quad (4)$$

ขั้นตอนที่4 : คำนวณหาทรานเฟอร์ฟังก์ชันของกระบวนการ

$$\frac{H_1(s)}{U_1(s)} = G(s) = \frac{T_1 k_{p1} / A_1}{(T_1 s + 1)} = \text{_____} \quad (5)$$

โดย $T_1 = \frac{A_1}{\beta_1 a_1} \sqrt{\frac{2h_1}{g}}$, s



รูปที่ 3 Schematic diagram สำหรับกระบวนการระดับน้ำหนึ่งถึง 1 อินพุต 1 เอาต์พุต



รูปที่ 4 ผลตอบสนองทางเอาต์พุตแบบวงรอบเปิดของกระบวนการ

ขั้นตอนที่5 : บันทึกค่า โพล _____ และซีโร _____ ของกระบวนการ

การทดลอง 2 การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการด้วยวิธีการวิเคราะห์จากผลตอบสนองของลูปเปิดในโดเมนเวลา

วัตถุประสงค์ เพื่อศึกษาการหาแบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ 4 ถัง ด้วยวิธีการวิเคราะห์จากผลตอบสนองของลูปเปิดในโดเมนเวลา

การหาโมเดลกระบวนการทางพลวัต นอกจากจะสามารถทำได้โดยวิธีการวิเคราะห์คุณลักษณะทางกายภาพแล้ว ยังสามารถหาแบบจำลองกระบวนการได้จากการทดลองด้วยวิธีการป้อนสัญญาณอินพุตแบบขั้นบันได และการวิเคราะห์คุณลักษณะของผลตอบสนองที่ได้เพื่อคำนวณหาแบบจำลองทางคณิตศาสตร์ของกระบวนการ

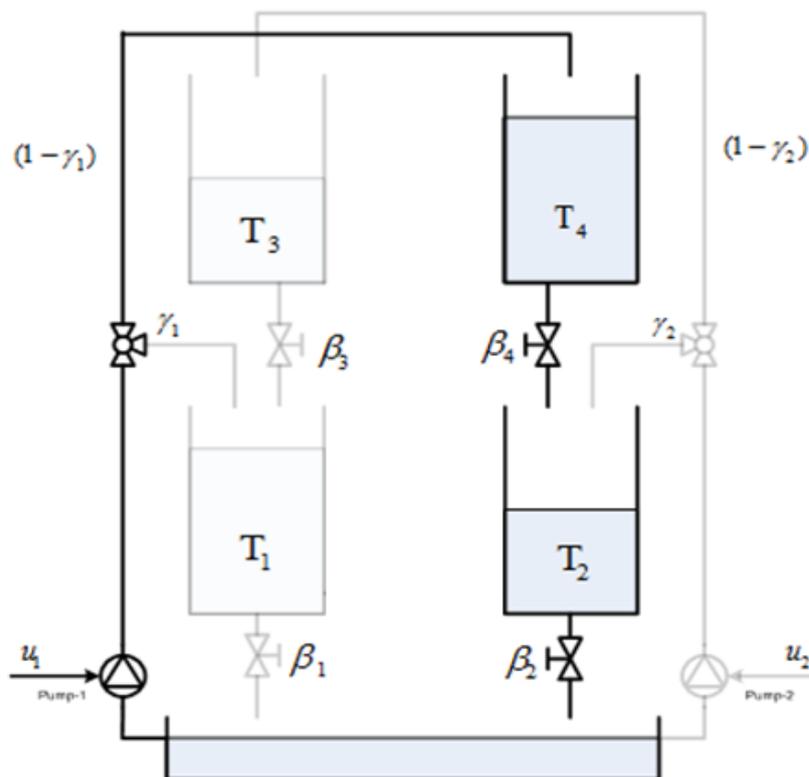
สำหรับใบงานการทดลองนี้ จะเป็นการประมาณโมเดลทางคณิตศาสตร์ของกระบวนการระดับน้ำในรูปแบบของสมการอันดับหนึ่งแบบมีเวลาหน่วง (First Order Plus Dead Time Process) โดยฟังก์ชันถ่ายโอน แสดงดังนี้

$$G_p = \frac{K_{ss}}{\tau s + 1} e^{-T_0 s} \quad (6)$$

โดย $K_{ss} = \frac{\Delta PV}{\Delta MV} \%$ = อัตราขยายของกระบวนการในสภาวะหยุดนิ่ง

τ = ค่าเวลาคงที่ของกระบวนการ

T_0 = ค่าเวลาหน่วงของกระบวนการ (วินาที)



ภาพที่ 5 Schematic diagram สำหรับกระบวนการระดับน้ำ 2 ถัง แบบอนุกรม

ขั้นตอนที่ 1 : ทำการกำหนดตำแหน่งวาล์วของกระบวนการระดับน้ำ 4 ถัง ให้มีโครงสร้าง Schematic diagram ตามภาพที่ 5 (กระบวนการระดับน้ำ 2 ถัง)

ขั้นตอนที่ 2 : ป้อนค่าแรงดันไฟฟ้าอินพุตคงที่ u_1 ที่ค่า 1 โวลต์ รอนจนผลตอบสนองของกระบวนการมีระดับนิ่ง

ขั้นตอนที่ 3 : เปลี่ยนระดับสัญญาณอินพุตไปประมาณ 5 หรือ 10%

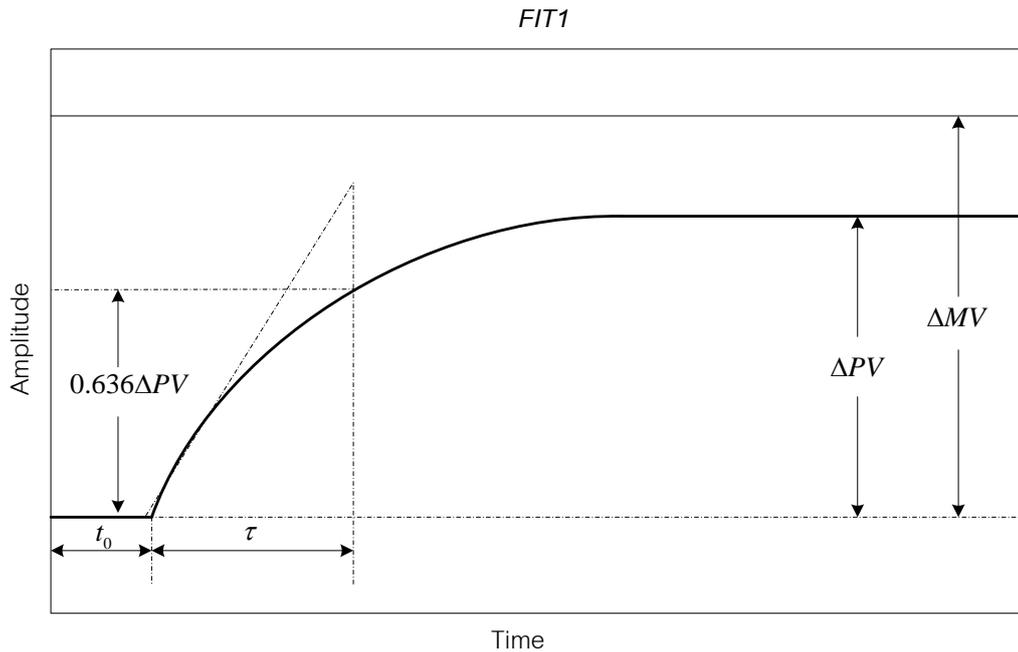
ขั้นตอนที่ 4 : บันทึกสัญญาณและรอนจนกระทั่งผลตอบสนองของกระบวนการนิ่งอีกครั้ง

ขั้นตอนที่ 5 : นำกราฟผลตอบสนองของกระบวนการพล็อตมาคำนวณหาโมเดลทางคณิตศาสตร์โดยที่หาเวลา K_{ss} , τ , T_0

$$G_p = \frac{K_{ss}}{\tau s + 1} e^{-T_0 s} = \text{_____} \quad (7)$$

ตารางที่ 1 ค่าพารามิเตอร์ของโมเดลทางคณิตศาสตร์ของกระบวนการ

	Process Gain K_{ss}	Dead Time T_0	Time Constant τ	T_0 / τ
Model A				



ภาพที่ 6 การหาค่าต่างๆ จากผลตอบสนองแบบขั้นโดยใช้ Model A

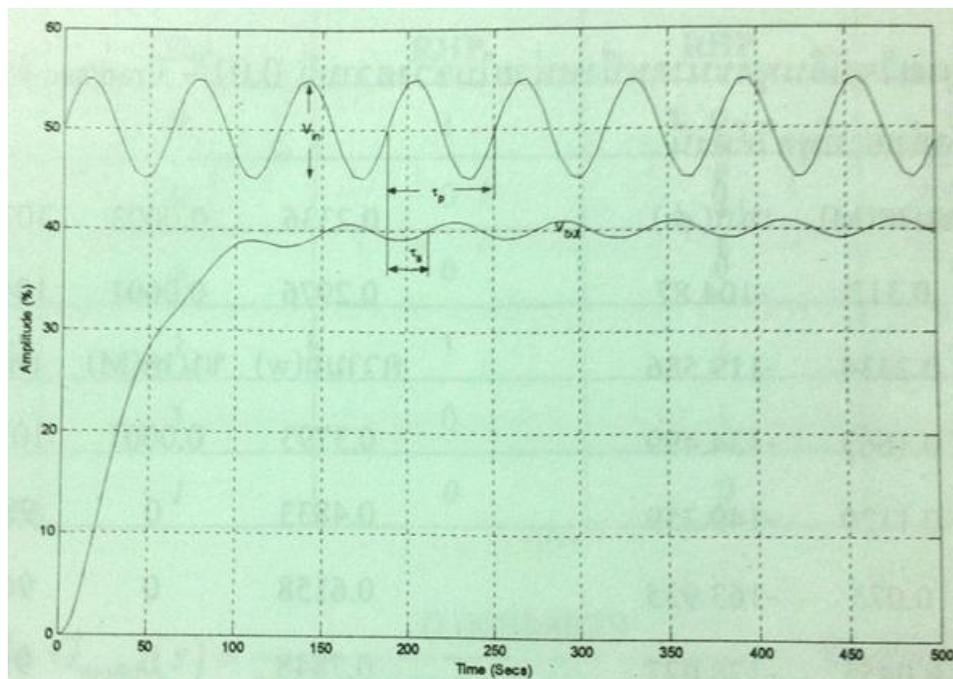
การทดลอง 3 การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการด้วยวิธีการวิเคราะห์จากผลตอบสนองของลูปเปิดในโดเมนความถี่

วัตถุประสงค์ เพื่อศึกษาการหาแบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ 4 ถัง ด้วยวิธีการวิเคราะห์จากผลตอบสนองของลูปเปิดในโดเมนความถี่

การหาผลตอบสนองทางความถี่ของกระบวนการ โดยที่ป้อนอินพุตที่มีขนาดคงที่และแปรความถี่เข้าไปยังกระบวนการพลวัตที่ต้องการทดสอบ จากนั้นนำผลตอบสนองเชิงความถี่ที่สามารถวัดขนาดและมุมเฟสที่เปลี่ยนแปลงไปซึ่งจากภาพที่ 6 สามารถหาขนาดเอาต์พุตและมุมเฟสได้จาก

$$AR = \frac{v_{pp}^{out}}{v_{pp}^{in}}, MR = \frac{AR}{K_{ss}}, \phi = -360 \frac{\tau_s}{\tau_p} [Deg] \quad (8)$$

จากนั้นนำค่าที่ได้มาพล็อตแผนภาพแบบโบลด์แล้วลากเส้นสัมผัสในการประมาณหาแบบจำลองกระบวนการทางคณิตศาสตร์



ภาพที่ 7 ผลตอบสนองทางความถี่ของกระบวนการ

ขั้นตอนที่ 1 : ทำการกำหนดตำแหน่งวาล์วของกระบวนการระดับน้ำ 4 ถัง ให้มีโครงสร้าง Schematic diagram ตามภาพที่ 8 (กระบวนการระดับน้ำ 2 ถัง แบบอนุกรม)

ขั้นตอนที่ 2 : ป้อนสัญญาณไซน์อินพุตขนาดหนึ่งหน่วยในช่วงความถี่ 0.01 ~ 1 rad/sec จำนวน 20 ตัวอย่างและบันทึกค่าเอาต์พุต ความถี่ ขนาด และเฟส

การวิเคราะห์และออกแบบระบบควบคุมกระบวนการระดับน้ำ 4 ถัง
ห้องปฏิบัติการระบบควบคุม
สาขาวิชาวิศวกรรมการวัดและควบคุม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

การทดลอง CS02 การออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ

วัตถุประสงค์

เพื่อทดลองออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ

บทนำ

ตัวควบคุมแอนะล็อก PID มีรูปแบบเป็นดังนี้

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (11)$$

$$G(s) = K_p + \frac{K_i}{s} + sK_d \quad (12)$$

โดยที่ $e(t)$ = error signal, $u(t)$ = control output, K_p = proportional gain,
 K_i = integral gain, K_d = derivative gain

การออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากเป็นอีกหนึ่งเทคนิควิธีที่ใช้การวางโพลของระบบ ในตำแหน่งที่เหมาะสม เพื่อให้ได้ผลตอบสนองในสถานะชั่วขณะตามข้อกำหนดที่ต้องการ โดยค่าพุงเกินและเวลาเข้าที่นั้นจะใช้ในการคำนวณหาค่าของอัตราการหน่วง (Damping ratio: ζ) ความถี่ธรรมชาติที่ไม่มีการหน่วง (Undamped natural frequency: ω_n) และตำแหน่งของรากที่สำคัญ (Dominant closed-loop poles: s_d) ของระบบควบคุมแบบวงปิด จากสมการต่อไปนี้

$$P.O. = 100e^{-\zeta\pi/\sqrt{1-\zeta^2}} \% \quad (13)$$

$$t_s = \frac{4}{\zeta\omega_n} \quad (14)$$

$$s_d = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \quad (15)$$

ค่าของ ζ , ω_n และ s_d ที่หาได้จากสมการ (1) จะใช้ในการออกแบบตัวควบคุม โดยรากของสมการคุณลักษณะจะอยู่ที่จุด s_d เมื่อคุณลักษณะของระบบควบคุมสอดคล้องตามเงื่อนไขของมุมและขนาดดังต่อไปนี้

$$\text{เงื่อนไขของมุม } \angle G_c(s_d)G_p(s_d) = \pm(2k+1)\pi; k=0,1,2,\dots \quad (16)$$

$$\text{เงื่อนไขของขนาด } K = \frac{|s_d| \left| \prod_{i=1}^{n-1} (s_d + p_i) \right|}{|(s_d + z_c)|^n} \quad (17)$$

อุปกรณ์

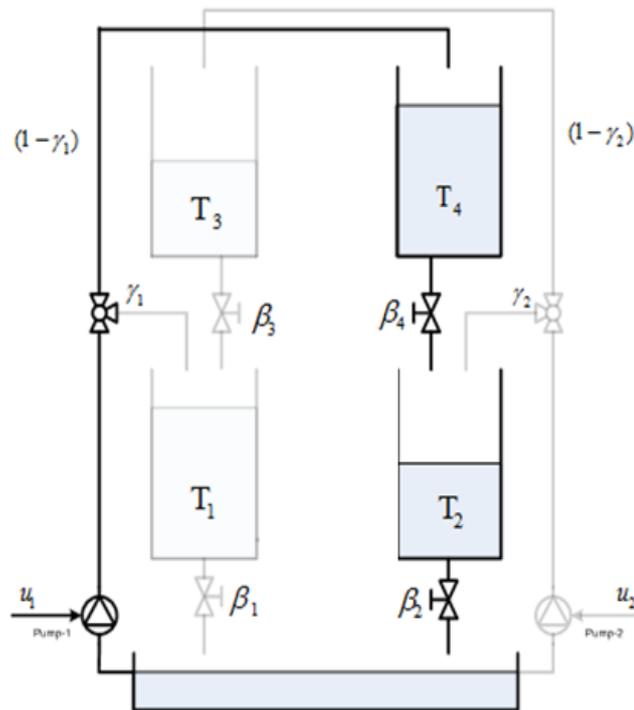
- ชุดทดลองกระบวนการระดับน้ำ 4 ถัง

2. เครื่องคอมพิวเตอร์พร้อมโปรแกรมใช้งาน

การทดลอง

การทดลอง 1 การทดลองออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ 2 ถัง แบบอนุกรม

วัตถุประสงค์ เพื่อศึกษาการออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ



ภาพที่ 9 Schematic diagram สำหรับกระบวนการระดับน้ำ 2 ถัง แบบอนุกรม

ขั้นตอนที่ 1 : ทำการกำหนดตำแหน่งวาล์วของกระบวนการระดับน้ำ 4 ถัง ให้มีโครงสร้าง Schematic diagram ตามภาพที่ 9 (กระบวนการระดับน้ำ 2 ถัง แบบอนุกรม)

ขั้นตอนที่ 2 : ทำการหาแบบจำลองทางคณิตศาสตร์ของกระบวนการโดยใช้ผลตอบสนองของอินพุตแบบขั้นบันได และทำการบันทึกผลการทดลอง

$$\text{Transfer Function } G(s) = \frac{Ke^{-t_0s}}{\tau s + 1} = \text{_____} \quad (18)$$

K คือ อัตราขยายของกระบวนการ

t_0 คือ ค่าเวลาหน่วงของกระบวนการ

τ คือ ค่าเวลาคงที่ของกระบวนการ

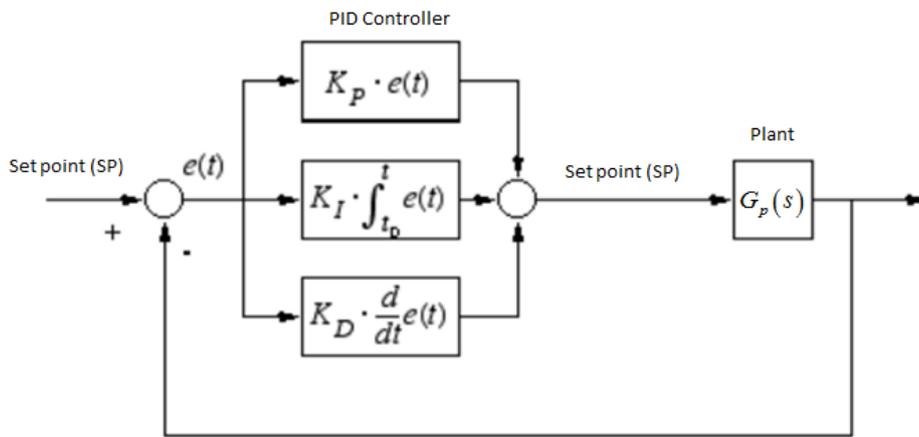
ขั้นตอนที่ 3 : ทำการออกแบบตัวควบคุมพีไอดีด้วยเทคนิคทางเดินราก ด้วยข้อกำหนดทางสมรรถนะของระบบคือ $t_s = 150 \text{ sec}$, $P.O. < 20\%$ และบันทึกเส้นทางเดินรากของระบบที่ชดเชยด้วยตัวควบคุมพีไอดี

ทรานเฟอร์ฟังก์ชันของตัวควบคุม

$$G(s) = K_p + \frac{K_i}{s} + sK_d = \text{_____} \quad (19)$$



ภาพที่ 10 Reaction Curve จากการป้อนสัญญาณอินพุตแบบขั้นบันได



ภาพที่ 11 โครงสร้างของระบบควบคุมแบบพีไอดี



ภาพที่ 12 ทางเดินรากลของระบบเมื่อชดเชยด้วยตัวควบคุม PID

ขั้นตอนที่4 : ทำการจำลองระบบควบคุมพีไอดีด้วยโปรแกรม Simulink โดยทำการควบคุมระดับน้ำจากตำแหน่ง 20% มาที่ตำแหน่ง 40% และบันทึกผลการทดลอง



ภาพที่ 13 ผลตอบสนองของการจำลองระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ

บันทึกคุณสมบัติผลตอบสนองของการจำลองระบบควบคุม

- Percent Maximum Overshoot: $M_p = \underline{\hspace{2cm}}$ % (20)

- Settling Time: $t_s = \underline{\hspace{2cm}}$ sec (21)

ขั้นตอนที่5 : ทำการทดลองระบบควบคุมกับกระบวนการจริง

ทดลองกับกระบวนการจริง โดยควบคุมผ่านไมโครคอนโทรลเลอร์ ARM7 แล้วส่งข้อมูลผ่านทาง RS-232 มาพลอตกราฟที่คอมพิวเตอร์ โดยทำการควบคุมระดับน้ำจากตำแหน่ง 20% มาที่ตำแหน่ง 40% และบันทึกผลการทดลอง



ภาพที่ 14 ผลตอบสนองของการทดลองระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ

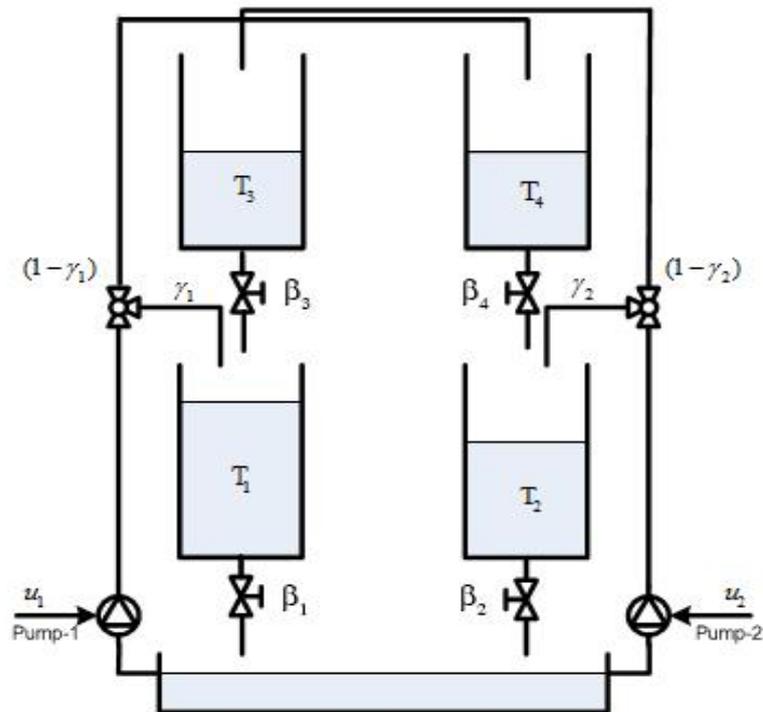
บันทึกคุณสมบัติผลตอบสนองของการทดลองระบบควบคุม

- Percent Maximum Overshoot: $M_p = \underline{\hspace{2cm}}$ % (22)

- Settling Time: $t_s = \underline{\hspace{2cm}}$ sec (23)

การทดลอง 2 การทดลองออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ 4 ถัง

วัตถุประสงค์ เพื่อศึกษาการออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ 4 ถัง



ภาพที่ 15 Schematic diagram สำหรับกระบวนการระดับน้ำ 4 ถัง

ขั้นตอนที่ 1 : ทำการกำหนดตำแหน่งวาล์วของกระบวนการระดับน้ำ 4 ถัง ให้มีโครงสร้าง Schematic diagram ตามภาพที่ 15

ขั้นตอนที่ 2 : ทำการหาแบบจำลองทางคณิตศาสตร์ของกระบวนการโดยใช้ผลตอบสนองของอินพุตแบบขั้นบันได และทำการบันทึกผลการทดลอง

$$\text{Transfer Function } G(s) = \frac{Ke^{-t_0s}}{\tau s + 1} = \text{_____} \quad (24)$$

K คือ อัตราขยายของกระบวนการ

t_0 คือ ค่าเวลาหน่วงของกระบวนการ

τ คือ ค่าเวลาคงที่ของกระบวนการ

ขั้นตอนที่ 3 : ทำการออกแบบตัวควบคุมพีไอดีด้วยเทคนิคทางเดินราก ด้วยข้อกำหนดทางสมรรถนะของระบบคือ $t_s = 150 \text{ sec}$, $P.O. < 20\%$ และบันทึกเส้นทางเดินรากของระบบที่ชดเชยด้วยตัวควบคุมพีไอดี ทรานเฟอร์ฟังก์ชันของตัวควบคุม

$$G(s) = K_p + \frac{K_i}{s} + sK_d = \text{_____} \quad (25)$$



ภาพที่ 16 Reaction Curve จากการป้อนสัญญาณอินพุตแบบขั้นบันได



ภาพที่ 17 ทางเดินรากลของระบบเมื่อชดเชยด้วยตัวควบคุม PID

ขั้นตอนที่ 4 : ทำการจำลองระบบควบคุมพีไอดีด้วยโปรแกรม Simulink โดยทำการควบคุมระดับน้ำจากตำแหน่ง 20% มาที่ตำแหน่ง 40% และบันทึกผลการทดลอง



ภาพที่ 18 ผลตอบสนองของการจำลองระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ
บันทึกคุณสมบัติผลตอบสนองของการจำลองระบบควบคุม

$$\text{- Percent Maximum Overshoot: } M_p = \text{_____ \%} \quad (26)$$

$$\text{- Settling Time: } t_s = \text{_____ sec} \quad (27)$$

ขั้นตอนที่ 5 : ทำการทดลองระบบควบคุมกับกระบวนการจริง

ทดลองกับกระบวนการจริง โดยควบคุมผ่านไมโครคอนโทรลเลอร์ ARM7 แล้วส่งข้อมูลผ่านทาง RS-232 มาพลอตกราฟที่คอมพิวเตอร์ โดยทำการควบคุมระดับน้ำจากตำแหน่ง 20% มาที่ตำแหน่ง 40% และบันทึกผลการทดลอง



ภาพที่ 19 ผลตอบสนองของการทดลองระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ

บันทึกคุณสมบัติผลตอบสนองของการทดลองระบบควบคุม

$$\text{- Percent Maximum Overshoot: } M_p = \text{_____ \%} \quad (28)$$

$$\text{- Settling Time: } t_s = \text{_____ sec} \quad (29)$$

สรุปผลการทดลอง

การวิเคราะห์ผลและสรุปผล

ในการนำเสนอผลตอบสนองของระบบด้วยกราฟ ควรใส่รายละเอียด ชื่อแกน และหน่วย พร้อมทั้งมี scale ที่ชัดเจน และบ่งชี้ค่าสำคัญที่สามารถนำไปใช้ในการวิเคราะห์ผลต่อไป

รายงาน

รายงานการทดลองควรสั้นและเข้าใจง่าย รายงานควรประกอบด้วยข้อมูลดิบที่วัดได้ การวิเคราะห์ข้อมูลดิบเพื่อนำไปสู่การปรับจูนพารามิเตอร์ของตัวควบคุม และสรุปเพื่อแสดงความเข้าใจเกี่ยวข้องกับการทดลอง ในการสรุปผลการทดลองเชิงเปรียบเทียบ ควรชี้ให้เห็นถึงข้อดี ข้อด้อย

ใบรายการผลการทดลอง

ใบงาน CS01 การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการระดับน้ำ (20 คะแนน)

การทดลอง 1 การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการด้วยวิธีการวิเคราะห์คุณลักษณะทางกายภาพ

- คำนวณหาอัตราการขยายของปั๊ม K_p1 (2 คะแนน)
- คำนวณหาค่าความต้านทานการไหลของวาล์ว $\beta 1$ (2 คะแนน)
- คำนวณหาทรานเฟอร์ฟังก์ชันของกระบวนการ (2 คะแนน)
- กราฟผลตอบสนองทางเอาต์พุตแบบวงรอบเปิดของกระบวนการ (2 คะแนน)
- บันทึกค่า โพล และซีโร ของกระบวนการ (2 คะแนน)

การทดลอง 2 การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการด้วยวิธีการวิเคราะห์จากผลตอบสนองของลูเปิดในโดเมนเวลา

- ตารางค่าพารามิเตอร์ของโมเดลทางคณิตศาสตร์ของกระบวนการ (2 คะแนน)
- กราฟผลตอบสนองในโดเมนเวลาสำหรับอินพุตแบบขั้นบันได (2 คะแนน)

การทดลอง 3 การหาแบบจำลองทางคณิตศาสตร์ของกระบวนการด้วยวิธีการวิเคราะห์จากผลตอบสนองของลูเปิดในโดเมนความถี่

- ตารางค่าเอาต์พุต ความถี่ ขนาด และเฟส (2 คะแนน)
- โมเดลของกระบวนการ (2 คะแนน)
- สรุปผลการทดลอง (2 คะแนน)

ใบงาน CS02 การออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ (40 คะแนน)

การทดลอง 1 การทดลองออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ 2 ถัง แบบอนุกรม

- คำนวณหาแบบจำลองทางคณิตศาสตร์ของกระบวนการ (2 คะแนน)
- คำนวณหาทรานเฟอร์ฟังก์ชันของตัวควบคุมพีไอดี (2 คะแนน)
- กราฟทางเดินรากของระบบเมื่อชดเชยด้วยตัวควบคุม PID (3 คะแนน)
- กราฟผลตอบสนองของการจำลองระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ (3 คะแนน)
- บันทึกคุณสมบัติผลตอบสนองของการจำลองระบบควบคุม (2 คะแนน)

- กราฟผลตอบแทนของการทดลองระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ (3 คะแนน)
- บันทึกคุณสมบัติผลตอบแทนของการทดลองระบบควบคุม (3 คะแนน)
- สรุปผลการทดลอง (2 คะแนน)

การทดลอง 2 การทดลองออกแบบระบบควบคุมพีไอดีด้วยเทคนิคทางเดินรากสำหรับกระบวนการระดับน้ำ 4 ถัง

- คำนิยามแบบจำลองทางคณิตศาสตร์ของกระบวนการ (2 คะแนน)
- คำนิยามทราานเฟอ์ฟังก์ชันของตัวควบคุมพีไอดี (2 คะแนน)
- กราฟทางเดินรากของระบบเมื่อชดเชยด้วยตัวควบคุม PID (3 คะแนน)
- กราฟผลตอบแทนของการจำลองระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ (3 คะแนน)
- บันทึกคุณสมบัติผลตอบแทนของการจำลองระบบควบคุม (2 คะแนน)
- กราฟผลตอบแทนของการทดลองระบบควบคุมพีไอดีสำหรับกระบวนการระดับน้ำ (3 คะแนน)
- บันทึกคุณสมบัติผลตอบแทนของการทดลองระบบควบคุม (3 คะแนน)
- สรุปผลการทดลอง (2 คะแนน)

ประวัติผู้วิจัย

ประวัติส่วนตัว

ชื่อ-สกุล นายอาจินต์ น่วมสำราญ

ตำแหน่ง รองศาสตราจารย์

ประวัติการศึกษา

ชื่อย่อปริญญา	สาขา	สถาบันที่จบ	ปีที่จบ
วิศวกรรมศาสตรมหาบัณฑิต	วิศวกรรมไฟฟ้า	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง	พ.ศ. 2543
วิศวกรรมศาสตรบัณฑิต	วิศวกรรมการวัดคุม	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง	พ.ศ. 2541

สาขาวิจัยที่มีความชำนาญพิเศษ (แตกต่างจากวุฒิการศึกษา).....

.....

รางวัลด้านวิชาการ/ด้านวิจัย/งานสร้างสรรค์ (ด้านศิลปะ หรืออื่นๆ) ที่ได้รับ

ปี พ.ศ.	ชื่อรางวัล	สถาบันที่ให้
-		

ทุนการศึกษาและทุนวิจัยที่เคยได้รับ

ปี พ.ศ.	ทุนการศึกษาและทุนวิจัย	สถาบันที่ให้
-		

ผลงานวิจัย/งานสร้างสรรค์

ผลงานวิจัย/งานสร้างสรรค์ที่ตีพิมพ์เผยแพร่ (ระดับชาติและนานาชาติ).....

1999

1. Kitti Tirasesth, Thanit Trisuwannawat, Arjin Numsomran, Noriyuki Komine and Yasuzumi Ochiai, "PID (n-2)/2 Stage PDA Controller Designed by CDM," Proceedings of the 1999 Japan Industry Applications Society Conference, Vol. 3, pp. 519-524, August 25-27, 1999, Nagasaki, Japan.

2. Arjin Numsomran, Prasit Julsereewong, Prapart Ukakimaparn, Thanit Trisuwannawat and Kitti Tirasesth, "Robust PID (n-1) Stage PD Controller," Proceedings of the 14th Korea Automatic Control Conference (KACC), International Session Papers, pp. E-298-E-301, October 14-16, 1999, Seoul, Korea.

2000

3. T. Thepmanee, K. Sirisantisamrit, A. Numsomran, V. Kongratana, K. Tirasesth, "2-DOF Control System Designed By Root Locus Technique," Proceedings of the 15th Korea Automatic Control Conference (KACC), International Session Papers, October 19-21, 2000, Seoul, Korea.

4. Tipsuwanporn, V.; Thepsathorn, P.; Riyarat, W.; Numsomran, A.; Bunjungjit, S. "4-quadrant DC motor drive control by BRM technique", Power Electronics and Motion Control Conference, 2000. Proceedings. IPEMC2000. The Third International, 2000, Page(s): 1185-1188 vol.3

5. Tipsuwanporn, V.; Numsomran, A.; Withephanich, K.; Piyarat, W.; Roddumporn, S. "Controller Design Improving Robustness Properties for Parametrically Uncertain System", IECON-2000. Proceedings, Page(s): 452-456

6. Tipsuwanporn, V.; Numsomran, A.; Piyarat, W.; Thepsathorn, P.; Charean, A. "New adjustable frequency controlled SPIM drive", TENCON 2000. Proceedings, Volume: 2, 2000, Page(s): 46-50 vol. 2

7. Tipsuwanporn, V.; Ruckaphan, A.; Numsomran, A.; Kaotien, J.; Withephanich. "Neurofuzzy application for process control", Proceedings of the 3rd Asian Control Conference, ASCC2000, 2000, Page(s): 532

2001

8. Tipsuwanporn, V.; Tarasantisuk, C.; Numsumran, A.; Sawaengsinkasikit, W. "Motor Speed Identification Using Multilayer Feedforward Neural Networks", The 4th IEEE International Conference on Power Electronics and Drive Systems Volume: 1, 2001, Page(s): 62-65

9. Tipsuwanporn, V.; Choochuan, C.; Numsumran, A.; Pathanasattanon, S.; Pirajanchai, V. "Economical Function Generator and Delta Modulator Circuits", The 4th IEEE International Conference on Power Electronics and Drive Systems Volume: 1, 2001, Page(s): 223-226

10. Tipsuwanporn, V.; Koetsam-ang, N.; Kongratana, V.; Numsomran, A.; Suesut, t. "A design neuro-fuzzy controller for level process control", Electrical and Computer Engineering, 2001. Canadian Conference on, Volume; 2, 2001 Page(s): 969-974

11. Tipsuwanporn, V.; Ruckaphan, A.; Kongratana, V.; Numsomran, A.; Tuppadung, Y. "Neurofuzzy Estimation for Fault Location Based on PLC", Proceedings of the International, Control, Automatic Conference on, ICCAS2001, 2001 Page(s): 1161-1164

12. Tipsuwanporn, V.; Sattho, U.; Numsomran, A.; Jaruvanawat, A; Charean, A. "A Digital Power Output Measurement Based on Binary Rate Multiplication Technique", Proceedings. 9th International Symposium On Integrated Circuits, Devices & Systems, ISIC 2001, 2001, Page(s): 480-481

13. Tipsuwanporn, V.; Choochuan, C.; Numsomran, A.; Pathanasattanon, S.; Pirajnanchai, V.; “Economical function generator and delta modulator circuits” Proceedings of International Conference on, Volume 1, 22-25 Oct. 2001 Page(s):223 - 226 vol.1

2002

14. Tipsuwanporn, V.; Numsomran, A.; Chuchotsakunleot, W.; Chuenarom, S.; Maitreechit, S. “Algorithmic ADC Using Current Mode without DAC”, The 2002 IEEE Asia-Pacific Conference on Circuits and Systems, 2002. IEEE APCCAS, Volume: 1, 2002, Page(s): 453 – 456.

15. Tipsuwanporn, V.; Sangrayub, T.; Gulpanich, S.; Suesut, T.; Numsomran, A. “A Development PLC Fiber-Optic Network for Redundant System”, 2002 IEEE International Conference on Industrial Technology, ICIT 2002, 2002, Bangkok, Thailand, Page(s):

16. Suesut, T.; Tipsuwanporn, V.; Sukprasert, P.; Gulpanich, S.; Numsomran, A. “A Design Automation Warehouse via Internet”, 2002 IEEE International Conference on Industrial Technology, ICIT 2002, Bangkok, Thailand, Page(s):

17. Tipsuwanporn, V.; Mitravakin, P.; Ukakimaparn, P.; Kulpanich, S.; Numsomran, A. “Contour Text Technique based on Autocad for Controlling 2-Axis Actuators”, 2002 IEEE International Symposium on Intelligent Control, ISIC 2002, 2002, Vancouver., Canada , Page(s): 631-636

18. Tipsuwanporn, V.; Srisuwan, K.; Kulpanich, S.; Suesut, T.; Numsomran, A. “Development of load Control and Management System” , IEEE/PES T&D 2002 Conference : Asia Pacific , Yokohama , Japan Page(s): 2139-2142

2003

19. Tipsuwanporn, V.; Jitnaknan, P.; Gulpanich, S.; Numsomran, A.; Runghimmawan, T. “Automatic tune parameter for digital PID controller based on FPGA”, Proceedings of the International , Control , Automatic Conference on, ICCAS2003, 2003 Page(s): 1012-1015

2004

20. Numsomran, A.; Suksri, T.; Kongratana, V.; Trisuwannawat, T.; Tirasesth, K. ” I-P Controller Design For Quadruple-Tank System” , 2004 IEEE , 0-7803-8560-8/04/\$20.00

21. Suesut, T.; Tipsuwanporn, V.; Nilas, P.; Rerngreun, P.; Numsomran, A. “ Multi Level Contract Net Protocol based on Holonic Manufacturing System Implement to Industrial Networks.”, Proceedings of the 2004 IEEE Conference on Robotics, Automation and Mechatronics Singapore, 1-3 December, 2004, Page(s): 253-255

2005

22. Numsomran, A.; Chaorai-ngern, J.; Suesut, T.; Trisuwannawat, T.; Tipsuwanporn, V. ” PID Controller Design using Characteristic Ratio Assignment Method for Coupled-Tank Process.”, Proceedings of the 2005 International Conference on Computational Intelligence for Modeling, Control and Automation, and International Conference Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'05)

23. Gulpanich, S.; Numsomran, A.; Roengruen, P.; Kongratana, V.; "Design of Programmable Logic Controller and I/O Expansions" Proceedings of the International , Control ,Automatic Conference on, ICCAS2005

24. Gulpanich S.; Numsomran A.; Tipsuwanporn V.; Tirasesth K.; Distributed control of network devices with remote terminal units.", Proceeding of International Conference on ; 14-17 Dec. 2005 Page(s):823 - 828

2006

25. Suesut, T.; Numsomran, A.; Tipsuwanporn, V. " Vision-based Network System for Industrial Applications.", Transactions on Engineering, Computing and Technology V12 March 2006 ISSN 1305-5313 Page(s): 98-102

26. Numsomran, A.; Gulpanich, S.; Tirasesth, K. " The Auto-Tuning PID Controller for Interacting Water Level Process.", Transactions on Engineering, Computing and Technology V12 March 2006 ISSN 1305-5313 Page(s): 134-138

27. Boonsrimuang, P.; Numsomran, A.; Puangma, T.; Kobayashi, H. " PAPR Reduction Method for OFDM Signal by Using Dummy Sub-Carriers. " Transactions on Engineering, Computing and Technology V12 March 2006 ISSN 1305-5313 Page(s): 211-215

28. Numsomran, A.; Withephanich, K.; Tipsuwanporn, V.; Klinsmitth, N. " Robust Controller Design for Plant Uncertainty. ", SICE-ICASE International Joint Conference 2006 Oct. 18-21, 2006 in Bexco, Busan, Korea

29. Tipsuwanporn, V.; Numsomran, A.; Sawangsinkasikit, W.; "Design and Implementation Multilevel Inverter for 3 ϕ Induction Motor Speed Control with RBM Chopper Technique" Proceedings of the International Symposium, Volume 3, July 2006 Page(s):2094 – 2098, Digital Object Identifier 10.1109/ISIE.2006.295896

30. Thepsatorn, P.; Numsomran, A.; Tipsuwanporn, V.; Teanthong, T. "DC Motor Speed Control using Fuzzy Logic based on LabVIEW", Proceedings of the International Joint Conference. SICE-ICA 2006

31. Numsomran, A.; Withephanich, K.; Trisuwannawat, T.; Tirasesth, K. " I-P Controller design for Quadruple Tank System ", Transactions on Engineering, Computing and Technology ENFORMATIKA V12 March 2006 ISSN1305-5313

32. Tunyasrirut, S.; Suksri, T.; Numsomran, A.; Gulpanich, S.; Tirasesth, K. " The Auto-Tuning PID Controller for Interacting Water Level Process ", Transactions on Engineering, Computing and Technology ENFORMATIKA V12 March 2006 ISSN1305-5313, Page(s): 134-138

33. Suesut, T.; Numsomran, A.; Tipsuwanporn, T. " Vision-based Network System for Industrial Applications ", Transactions on Engineering, Computing and Technology ENFORMATIKA V12 March 2006 ISSN1305-5313, Page(s): 98-102

34. Suksri, T.; Sritheeravirojana, U.; Numsomran, A.; Kongrattana, V.; Werataweemart, Thongchai. " T-DOF PID Controller Design using Characteristic Ratio Assignment Method for Quadruple Tank Process ", Transactions on Engineering, Computing and Technology ENFORMATIKA V12 March 2006 ISSN1305-5313, Page(s): 252-258

2007

35. Numsomran, A. ; Tipsuwannaporn, V. ; Boonsrimuang, P., “Traffic forecasting and navigation assistance system via web application”, Control, Automation and Systems, 2007, Page(s): 857 – 860

36, Wongkhum, T. ; Suksri, T. ; Nilas, P. ; Chaoraingern, J., “Design of Decoupled Controller for TITO System using Characteristic Ratio Assignment”, Control, Automation and Systems, 2007, Page(s): 957 – 862

37. Arjin Numsomran, Tianchai Suksri, Maitree Thumma, “Design of 2-DOF PI Controller with Decoupling for Coupled-Tank Process” , Control, Automation and Systems, 2007, Page(s): 339 – 344

2008

38 Arjin Numsomran, Saen Niyomsuk and Vittaya Tipsuwanporn, “The Feasibility Study of Noise Disturbance in STW Process”, Control, Automation and Systems, 2008, Page(s): 1125 – 1129

39. A. Numsomran, V. Tipsuwanporn, K. Tirasesth, “Modeling of the Modified Quadruple-Tank Process”, SICE Annual Conference, 2008, Page(s): 818 - 823

2010

40. Puawade, P.; Boonpranchoo, V.; Tipsuwannaporn, V. and Numsomran, A., “Smith Predictor Design by CDM for Temperature Control System” , Proceedings of the International , Control ,Automatic Conference , ICCAS2010, 2010, Page(s): 1472-1477

41. Trisuwannawat, T.; Chaoraingern, J.; Numsomran, A., “Following control of hard disk drive using derivative state constrained optimal H2 controls”, Proceedings of the International , Control ,Automatic Conference , ICCAS2010, 2010, Page(s): 1275-1280

42. Kangwanrat, S.; Tipsuwannaporn, V.; Numsomran, A., “Design of PI Controller Using MRAC Techniques for Coupled-Tanks Process” , Proceedings of the International , Control ,Automatic Conference , ICCAS2010, 2010, Page(s): 485-490

43. Singhapan, A.; Numsomran, A.; Boonsrimuang, P. and Kobayashi, K., “Proposal of Non-Uniform Weighting Factor in Concurrent PTS Algorithm for STBC MIMO-OFDM Systems,” International Workshop on Information Communication Technology (ICT 2010) , 2010, KMITL, Bangkok, Thailand.

2011

44. Chaoraingern, J.; Vaidee, W.; Trisuwannawat, T.; Numsomran, A. and Tipsuwanporn, V., “The Design of PID controller for track following control of hard disk drive using Coefficient Diagram Method”, SICE Annual Conference (SICE), 2011 Proceedings of Publication Year: 2011 , Page(s): 2949- 2954

45. Numsomran, A.; Tipsuwanporn, V.; Trisuwannawat, T. and Tirasesth, K., “Design of PID Controller for the Modified Quadruple-Tank Process using Inverted Decoupling Technique”,

Proceedings of the International , Control ,Automatic Conference , ICCAS2011, 2011, Page(s): 1364-1368

46. Numsomran, A.; Suksri, T.; Tipsuwanporn, V.; Trisuwannawat, T. and Tirasesth, K., “Design of PI Controller using decoupling and CRA techniques for quadruple-tanks process” , Proceedings of the International , Control ,Automatic Conference , ICCAS2011, 2011, Page(s): 1358-1363

47. Boonpranchoo, V.; Kongratana, V.; Tipsuwanporn, V. and Numsomran, A., “Design of Temporal Logic Embedded Controller for Small Oven Process” , Proceedings of the International , Control ,Automatic Conference , ICCAS2011, 2011, Page(s): 1354-1357

48. Kungwalrut, P.; Thumma, M.; Tipsuwanporn, V.; Numsomran, A. and Boonsrimuang, P., “Design MRAC PID Control for Fan and Plate Process”, SICE Annual Conference, 2011, Page(s): 2944-2948

2012

49. Nakharin, W.; Boonpranchoo, V.; Tipsuwanporn, V. and Numsomran, A., “The Design of Load/Unload Algorithm for 2.5” Hard Disk Drive in SSTW Process”, 2012 7th IEEE Conference on Industrial Electronics and Applications, ICIEA 2012, Page(s): 72

50. Ronnapop Jaisue, Jutarut Chaoraingern, Vittaya Tipsuwanporn, Arjin Numsomran and Phayupp Pholkeaw, “A Design of Fuzzy PID Controller Based on ARM7TDMI for Coupled-Tanks Process”, 2012 12th International Conference on Control, Automation and Systems, Oct. 17-21, 2012, Page(s): 610-613

การเสนอผลงานวิชาการ

.....

.....

.....

ผลงานตีพิมพ์/สิ่งประดิษฐ์/งานสร้างสรรค์ (ศิลปะ หรือ อื่นๆ)

.....

.....

.....

อื่นๆ

.....

.....