



ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมไฟฟ้า)

ปริญญา

วิศวกรรมไฟฟ้า

วิศวกรรมไฟฟ้า

สาขา

ภาควิชา

เรื่อง การควบคุมอุณหภูมิแบบอัตโนมัติภายในถังแพคเบดสำหรับกระบวนการหมักโคจิด้วยวิธี
Fuzzy PID

Automatic Temperature Control in Packed-Bed Reactor by Fuzzy PID Method

นามผู้วิจัย นายนคร ตรีสิงหวงศ์

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(ผู้ช่วยศาสตราจารย์พีระยศ แสนโกชณ์, D.Sc.)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

(อาจารย์จันทน์ รุ่งเรืองพิทยากุล, D.Sc.)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

(อาจารย์ยศวีร์ วีระกำแหง, Ph.D.)

หัวหน้าภาควิชา

(รองศาสตราจารย์มงคล รักษาพัชรวงค์, Ph.D.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

(รองศาสตราจารย์กัญญา ชีระกุล, D.Agr.)

คณบดีบัณฑิตวิทยาลัย

วันที่ เดือน พ.ศ.

วิทยานิพนธ์

เรื่อง

การควบคุมอุณหภูมิแบบอัตโนมัติภายในถังแพคเบดสำหรับกระบวนการหมักโคจิด้วยวิธี

Fuzzy PID

Automatic Temperature Control in Packed-Bed Reactor by Fuzzy PID Method

โดย

นายนคร ตริสิงหวงศ์

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมไฟฟ้า)

พ.ศ. 2552

นคร ศรีสิงหวงศ์ 2552: การควบคุมอุณหภูมิแบบอัตโนมัติภายในถังแพคเบตสำหรับ
กระบวนการหมักโคจิด้วยวิธี Fuzzy PID ปริญญาวิศวกรรมศาสตรมหาบัณฑิต
(วิศวกรรมไฟฟ้า) สาขาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า อาจารย์ที่ปรึกษาวิทยานิพนธ์
หลัก: ผู้ช่วยศาสตราจารย์พีระยศ แสนโกชณ์, D.Sc. 106 หน้า

กระบวนการหมักโคจิ เป็นขั้นตอนสำคัญในกระบวนการผลิตซีอิ๊วและเต้าเจี้ยว โดยปัจจุบัน
กระบวนการหมักโคจิแบบแห้งในถังแพคเบตเริ่มเป็นที่มีความสำคัญในการผลิตเชิง อุตสาหกรรมแต่
เนื่องจากปัญหาที่พบในกระบวนการหมักคือความร้อนที่เกิดขึ้นจากการปฏิกิริยาการเจริญเติบโตของจุลินทรีย์
Aspergillus จะถูกเก็บสะสมอยู่ในถังแพคเบตจนกระทั่งอุณหภูมิภายในถังแพคเบตมีค่าสูง
กว่าช่วงอุณหภูมิที่เหมาะสมต่อการทำงานของจุลินทรีย์คือที่ 30 องศาเซลเซียส และหากอุณหภูมิสูงกว่า
40 องศาเซลเซียสจะทำให้จุลินทรีย์ตายได้ ดังนั้นจึงจำเป็นต้องมีการควบคุมอุณหภูมิภายในถังแพคเบต
เพื่อให้การหมักมีประสิทธิภาพมากขึ้น

วิทยานิพนธ์นี้นำเสนอการควบคุมอุณหภูมิแบบอัตโนมัติภายในถังแพคเบตสำหรับ
กระบวนการหมักโคจิด้วยวิธี Fuzzy PID เพื่อควบคุมให้อุณหภูมิภายในถังแพคเบตให้อยู่ในช่วง
อุณหภูมิที่เหมาะสมต่อการเจริญเติบโตของจุลินทรีย์ งานวิจัยนี้ได้ทำการออกแบบบอร์ดควบคุม โดยใช้
MCU PIC24HJ128GP306 เพื่อเป็นอุปกรณ์ในการทดลอง โดยในการออกแบบ Fuzzy PID เราจะใช้การ
Simulation กับสมการแบบจำลองทางคณิตศาสตร์ของกระบวนการหมักโคจิเพื่อใช้ หา Fuzzy Rule และ
Membership function ของ Fuzzy PID และใช้เป็นข้อมูลในการออกแบบ software Fuzzy PID ของ
บอร์ดควบคุม

Nakorn Threesinghawong 2009: Automatic Temperature Control in Packed-Bed Reactor by Fuzzy PID Method. Master of Engineering (Electrical Engineering), Major Field: Electrical Engineering, Department of Electrical Engineering. Thesis Advisor: Assistant Professor Peerayot Sanposh, D.Sc. 106 pages.

Today Koji Fermenting Process in packed bed reactor is important to producing basic material with use to producing soy bean source. In this process, *Aspergillus oryzae* produces heat from its metabolism. This heat is stored in fermenting material. Thus, the temperature in packed-bed reactor is increased. When the temperature is higher than 40 degree Celsius, most of *Aspergillus oryzae* will die. And the good condition for fermenting is 30 degree Celsius. So we must maintain the bed temperature to not exceed 30 degree Celsius.

This thesis proposes a Fuzzy PID method to Control bed temperature. In this thesis we use MCU PIC 24HJ128GP306 to develop the controller board and use matlab simulation with mathematical model of Koji process to find out Fuzzy PID rule and Membership function.

Student's signature

Thesis Advisor's signature

กิตติกรรมประกาศ

ขอขอบพระคุณ ผศ.ดร. พีระยศ แสนโกชณ์ ประธานกรรมการที่ปรึกษา เป็นอย่างสูงที่ให้คำปรึกษาและช่วยชี้แนะแนวทางในการแก้ไขปัญหาต่างๆ รวมถึงการให้ความสนับสนุนงบประมาณในการทำวิทยานิพนธ์ ทำให้ผลงานวิทยานิพนธ์ครั้งนี้สำเร็จไปได้ด้วยดี

ขอขอบพระคุณ รศ.ดร.เพ็ญจิตร ศรีนพ ที่ให้คำปรึกษาและตรวจสอบเพื่อให้คำชี้แนะและชี้ให้เห็นถึงปัญหาเพื่อนำไปปรับปรุงแก้ไขในการทำวิทยานิพนธ์นี้ให้มีประสิทธิภาพมากยิ่งขึ้น

ขอขอบคุณเพื่อนๆ พี่ๆ และน้องๆ นิสิตปริญญาโททั้งในห้องปฏิบัติการควบคุมภาควิชาวิศวกรรมไฟฟ้าและห้องปฏิบัติการภาควิชาวิศวกรรมเคมีทุกท่านที่ให้กำลังใจ ให้ความช่วยเหลือ และให้คำปรึกษาในการทำงานวิทยานิพนธ์นี้

สุดท้าย ต้องขอกราบขอบพระคุณ บิดา มารดา เป็นอย่างสูงที่ให้ความสนับสนุนและเป็นที่กำลังใจที่สำคัญซึ่งทำให้การศึกษาระดับปริญญาโทของข้าพเจ้าประสบผลสำเร็จลุล่วงได้ด้วยดี

นคร ตรีสึงหวงส์

กุมภาพันธ์ 2552

สารบัญ

	หน้า
สารบัญ	(1)
สารบัญตาราง	(2)
สารบัญภาพ	(3)
คำนำ	1
วัตถุประสงค์	2
การตรวจเอกสาร	3
อุปกรณ์และวิธีการ	17
อุปกรณ์	17
วิธีการ	17
ผลและวิจารณ์	68
ผล	68
วิจารณ์	74
สรุปและข้อเสนอแนะ	78
สรุป	78
ข้อเสนอแนะ	78
เอกสารและสิ่งอ้างอิง	79
ภาคผนวก	
ภาคผนวก ก การสร้าง Setup File สำหรับโปรแกรมที่ถูกพัฒนามบนโปรแกรม LabVIEW8.2	81
ภาคผนวก ข โปรโตคอล Modbus RTU ที่ใช้ในงานวิจัย	90
ภาคผนวก ค การปรับจูน Fuzzy PID	95
ประวัติการศึกษา และการทำงาน	107

สารบัญตาราง

ตารางที่		หน้า
1	Output (mV) ของ Thermocouple type k ที่อุณหภูมิขานบวกเมื่ออุณหภูมิ Cold junction เป็น 0 องศาเซลเซียส	30
2	Output (mV) ของ Thermocouple type k ที่อุณหภูมิขานลบเมื่ออุณหภูมิ Cold junction เป็น 0 องศาเซลเซียส	31
3	Output (ohm) ของ PT100	33
4	แสดงค่า index สำหรับตัวแปร use	57

สารบัญภาพ

ภาพที่		หน้า
1	แสดงภาพถึงแพคเกจแบบต่างๆ	4
2	ภาพเชื้อรา <i>Aspergillus oryzae</i>	6
3	โครงสร้างของ Fuzzy Inference System	6
4	Mamdani Fuzzy Inference System (max-min composition)	7
5	Mamdani Fuzzy Inference System (max-product composition)	8
6	Sugeno Fuzzy Inference System (weight average)	10
7	Sugeno Fuzzy Inference System (weight sum)	10
8	Tsugamoto Fuzzy Inference System	11
9	ภาพลักษณะของถึงแพคเกจต้นแบบ	12
10	แสดงการทำงานของระบบที่ทำการศึกษา	18
11	แสดงการทำงานของระบบที่เขียนขึ้นบน Simulink	20
12	แสดงรายละเอียดของโปรแกรมระบบที่เขียนบน Simulink	20
13	โครงสร้างของระบบ Fuzzy ที่ทำการศึกษา	22
14	Membership function ของ input fuzzy e	22
15	Membership function ของ input fuzzy de/dt	23
16	Membership function ของ input fuzzy $\int edt$	23
17	Membership function ของ Output fuzzy u	24
18	Control Surface ของ Fuzzy PID	24
19	แสดง Fuzzy rule สำหรับ input e , de และ $\int edt$ เป็น Pos	25
20	แสดง Fuzzy rule สำหรับ input e , de และ $\int edt$ เป็น Neg e	25
21	แสดงองค์ประกอบหลักในบอร์ดควบคุม	26
22	แผนภาพแสดงรายละเอียดการส่งสัญญาณ 4 ถึง 20 mA ไปควบคุม Drive	27
23	แผนภาพแสดงรายละเอียดการส่งสัญญาณ 0 ถึง 10 V ไปควบคุม Drive	28
24	แสดงวงจรที่ใช้สำหรับอ่านค่า Thermocouple Type k	29
25	แสดงวงจรที่ใช้สำหรับอ่านค่า PT100	29

สารบัญภาพ(ต่อ)

ภาพที่		หน้า
26	แสดงถึงการทำงานของ Thermocouple sensor	30
27	แสดงวงจรภาครับสัญญาณ Thermocouple ที่มี Gain ขยาย 200 เท่า	32
28	แสดงวงจรที่ใช้สำหรับอ่านค่า Cold junction (Tj)	33
29	แสดงให้เห็นถึงวงจรที่ใช้สำหรับการอ่านค่า Output ของ PT100	34
30	แสดงรายละเอียดของวงจร VCS	35
31	แสดงการหาความสัมพันธ์ของ Va, Vb และ I L	35
32	แสดงรายละเอียดของสัญญาณ PWM เมื่อผ่าน low pass filter	37
33	แสดงรายละเอียดของวงจรที่ใช้งาน	38
34	แสดงการเชื่อมต่อของบอร์ดควบคุมด้วยอุปกรณ์ Converter ต่างๆ	39
35	แสดงรายละเอียดของวงจรสื่อสาร	39
36	แสดงรายละเอียดการของวงจรการแสดงผลและ Key Pad Interface	40
37	แสดงแผนภาพการทำงานใน 16 bit Timer1 Module	41
38	แสดงการทำงานของ Displaypage function	42
39	แสดงการตัวอย่างการทำงานของ function WriteStr2Seg ในกรณีไม่มี Character “.”	43
40	แสดงการตัวอย่างการทำงานของ function WriteStr2Seg ในกรณีมี Character “.”	43
41	แสดงแผนผังการทำงานใน interrupt service ของ Timer 2 Module	44
42	แสดงการทำงานของ function Rtc_Module();	45
43	แสดงการทำงานของ function Funckey(); เมื่อ Display.page=0	46
44	แสดงการทำงานของ function Funckey(); เมื่อ Display.page=1	46
45	แสดงการทำงานของ function Funckey(); เมื่อ Display.page=2	46
46	แสดงให้เห็นถึงการทำงานของ mb_req()	47
47	แสดงการทำงานของ DMA 0 Module	48
48	แสดงให้กระบวนการทำงานของ UART และ DAM1 Module	49
49	แสดงกระบวนการทำงานใน Interrupt service ของ DMA 2 Module	50

สารบัญภาพ(ต่อ)

ภาพที่		หน้า
50	แสดงการทำงานของโปรแกรมใช้ Main Program ของ MCU	51
51	แสดงโครงสร้างข้อมูลของ Input Fuzzy	53
52	แสดงตัวอย่างค่าข้อมูลที่เก็บในตัวแปรโดยมี Input error=2	53
53	แสดงตัวอย่างโครงสร้างข้อมูลที่ใช้แทน Membership function	54
54	Function คำนวณค่าความเป็นสมาชิกของ Membership function แบบ triangle	55
55	Function คำนวณค่าความเป็นสมาชิกของ Membership function แบบ trapezoidal	55
56	แสดงการทำ Fuzzification ของ 1 input fuzzy	56
57	แสดงโครงสร้างข้อมูลสำหรับ Fuzzy Rule	57
58	ตัวอย่างการเขียน Fuzzy Rule ในบอร์ดควบคุม	57
59	แสดงตัวอย่างในการคำนวณสำหรับการทำ defuzzification	58
60	แสดงตัวอย่างโปรแกรมที่เขียนขึ้นสำหรับ Defuzzification	59
61	แสดงการจัดเตรียมข้อมูลสำหรับอ่านค่าอุณหภูมิจากบอร์ดควบคุม	60
62	แสดงการจัดเตรียมข้อมูลสำหรับเขียนข้อมูลลงบนบอร์ดควบคุม	61
63	แสดงตัวอย่างโครงสร้างของการใช้งาน VISA Write และ VISA Read	61
64	แสดงกระบวนการตรวจสอบข้อมูลที่อ่านมาได้	62
65	แสดงตัวอย่างการอ่าน Data จาก Modbus RTU message	63
66	แสดงตัวอย่างการแปลความหมายของ Data Register จาก Modbus RTU message	63
67	แสดงตัวอย่างการใช้งาน Numeric Indicator	64
68	แสดงตัวอย่างการใช้งาน Waveform Chart	64
69	แสดงกระบวนการเขียนข้อมูลลงใน file excel	65
70	แสดงตัวอย่างการใช้งาน Excel Report ในโปรแกรม Labview	66
71	แสดงโปรแกรมในส่วนที่ใช้ในการอ่านและเขียนข้อมูลลง Serial port	66
72	แสดงโปรแกรมในส่วนที่ใช้เก็บข้อมูลลง excel file7	66
73	แสดงโปรแกรม Sub VI ที่ใช้ในการคำนวณ CRC ขนาด 16 bits	67

สารบัญภาพ(ต่อ)

ภาพที่		หน้า
74	แสดงโปรแกรม Sub VI ที่ใช้ในการเตรียมข้อมูลสำหรับส่งผ่านทาง Serial Port	67
75	แผนผังแสดงขั้นตอนในการทดลอง	68
76	แสดงผลของอุณหภูมิภายในถังแพคเกจที่ควบคุมโดย Fuzzy PID	69
77	แสดงค่า input process ที่ใช้ในการควบคุม	70
78	แสดงผลการทดลองที่ได้ของการทดลองที่ 2	71
79	แสดงผลการทดลองที่ได้ของการทดลองที่ 3	72
80	แสดงผลการทดลองที่ได้ของการทดลองที่ 4	74
81	แสดงวงจรที่ใช้สำหรับอ่านค่า Thermocouple Type k	74
82	แสดงวงจรที่ใช้สำหรับอ่านค่า PT100	75

คำอธิบายสัญลักษณ์และคำย่อ

μ_C	=	ค่าผลลัพธ์จากค่าความเป็นสมาชิกของ Output u ของทุกกฎ Fuzzy รวมกัน
μ_{Ai}	=	ค่าความเป็นสมาชิกของ Input Fuzzy A สำหรับกฎที่ i ของกฎ Fuzzy
μ_{Bi}	=	ค่าความเป็นสมาชิกของ Input Fuzzy B สำหรับกฎที่ i ของกฎ Fuzzy
μ_{Ci}	=	ค่าความเป็นสมาชิกของ Out Fuzzy สำหรับกฎที่ i ของกฎ Fuzzy
Z	=	ค่า Crips Output ของ Fuzzy system ที่ได้จากการทำ defuzzification
μ	=	อัตราการเจริญจำเพาะ
X	=	ปริมาณชีวมวล
$X_{\max}$	=	ปริมาณชีวมวลที่สามารถเกิดขึ้นสูงสุด
R	=	ค่าคงที่ของแก๊สมีค่าเท่ากับ 0.001986 กิโลแคลอรี/กรัม โมล-เคลวิน
T	=	อุณหภูมิเบดหน่วย เคลวิน
ρ_{bed}	=	ความหนาแน่นของเบดหน่วย กิโลกรัม/เมตร ³
Cp_{bed}	=	ความจุความร้อนของเบดหน่วย จูล/กิโลกรัม-เคลวิน
ε	=	สัดส่วนช่องว่างหรือค่าความพรุนของเบด
ρ_{drymat}	=	ความหนาแน่นของวัสดุหมักหน่วย กิโลกรัม/เมตร ³
Cp_{water}	=	ความจุความร้อนของน้ำหน่วย จูล/กิโลกรัม-เคลวิน
W	=	สัดส่วนน้ำในเบดหน่วย กิโลกรัมน้ำ/กิโลกรัมวัสดุหมัก
$\frac{dT}{dt}$	=	อัตราการเปลี่ยนแปลงอุณหภูมิเบด
V_z	=	ความเร็วอากาศหน่วย เมตร/ชั่วโมง
H_{bed}	=	ความสูงของเบดหน่วย เมตร
ρ_{moist}	=	ความหนาแน่นของอากาศชื้นหน่วย กิโลกรัม/เมตร ³
Cp_{moist}	=	ความจุความร้อนของอากาศชื้นหน่วย จูล/กิโลกรัม-เคลวิน
T_{in}	=	อุณหภูมิอากาศขาเข้าหน่วย เคลวิน
T_{out}	=	อุณหภูมิอากาศขาออกหน่วย เคลวิน
λ	=	ความร้อนแฝงของการกลายเป็นไอน้ำหน่วย จูล/กิโลกรัม
$Y_{w_{in}}$	=	สัดส่วนน้ำในอากาศขาเข้า
$Y_{w_{out}}$	=	ปริมาตรสัดส่วนน้ำในอากาศขาออก

คำอธิบายสัญลักษณ์และคำย่อ(ต่อ)

$y_{Q/x}$	=	สัมประสิทธิ์ผลได้ของพลังงานจากจูลินทรีย์หน่วย
$m_{Q/x}$	=	สัมประสิทธิ์ผลได้ของพลังงานจากจูลินทรีย์ในระยะคงที่หน่วย
Σ	=	ค่าผลรวม
P_w	=	ความดันไออิ่มตัวหน่วยเป็นมิลลิเมตรปรอท
T_c	=	อุณหภูมิหน่วย องศาเซลเซียส
P_{atm}	=	ความดันบรรยากาศ

การควบคุมอุณหภูมิแบบอัตโนมัติภายในถังแพคเบดสำหรับกระบวนการหมักโคจิด้วย วิธี Fuzzy PID

Automatic Temperature Control in Packed-Bed Reactor by Fuzzy PID Method

คำนำ

ถังแพคเบด (packed-bed reactor) หรือ ฟิกซ์เบด (fixed-bed reactor) เป็นถังปฏิกรณ์ชนิดหนึ่งที่ใช้กันมากในอุตสาหกรรมเคมี ซึ่งมักใช้ในอุตสาหกรรมที่เกี่ยวข้องกับกระบวนการเกิดปฏิกิริยาระหว่างของแข็งกับของไหล การใช้งานถังแพคเบดมีจุดเด่นหลายอย่าง แต่จุดด้อยของถังแพคเบดที่พบมากที่สุดคือ ข้อจำกัดในการถ่ายเทความร้อนออกจากระบบ เมื่อปฏิกิริยาที่เกิดขึ้นเป็นปฏิกิริยาคายความร้อน ความร้อนจะสะสมอยู่ภายในเบด ทำให้อุณหภูมิของระบบสูงขึ้นจนเป็นปัญหาต่อกระบวนการผลิต ดังนั้นการใช้ระบบควบคุมความร้อนจึงมีความจำเป็น และมักใช้งานควบคู่ไปกับการใช้ถังแพคเบด

โดยในปี พ.ศ. 2546 พงษ์ประวัติ พิมพารัง ได้ทำการศึกษาหาแบบจำลองทางคณิตศาสตร์ (Model predictive control) ของกระบวนการหมักโคจิในถังแพคเบด และได้ใช้แบบจำลองนี้ในการควบคุม แต่แบบจำลองที่ได้มีความซับซ้อนอย่างมากจึงทำให้ยากต่อการทำงาน

ดังนั้นในวิทยานิพนธ์นี้จึงสนใจนำการควบคุมแบบ FUZZY PID มาประยุกต์ใช้เนื่องจากการควบคุมที่สามารถประยุกต์ใช้กับระบบที่มีแบบจำลองทางคณิตศาสตร์ที่ซับซ้อนได้ดีและยังสามารถนำ FUZZY RULE ที่ได้ไปใช้งานบนอุปกรณ์ควบคุมพวก Microcontroller ได้หลากหลาย

วัตถุประสงค์

1. เพื่อสร้างอุปกรณ์ควบคุมอุณหภูมิอัตโนมัติที่ใช้ในการควบคุมอุณหภูมิในถังแพคเบคสำหรับกระบวนการหมักโคจิ
2. เพื่อศึกษาและทดสอบการควบคุมอุณหภูมิในการหมักโคจิด้วยทฤษฎี Fuzzy PID
3. เพื่อเป็นอุปกรณ์ต้นแบบที่จะนำไปพัฒนาใช้งานได้จริงในงานอุตสาหกรรม

การตรวจเอกสาร

รายงานผลงานวิจัยที่ผ่านมา

จากการค้นคว้าและศึกษาข้อมูลการควบคุมแบบ Fuzzy PID และการควบคุมอุณหภูมิอัตโนมัติในถังหมักโคจิพบว่า ปัญหาที่เกิดขึ้นในการใช้ถังแพคเบดในการหมักโคจิคือ ความร้อนที่เกิดขึ้นจากปฏิกิริยาการเจริญเติบโตของจุลินทรีย์จะเพิ่มอุณหภูมิในถังแพคเบด จนกระทั่งอุณหภูมิภายในถังแพคเบดมีค่าสูงกว่าช่วงอุณหภูมิที่เหมาะสมต่อการทำงานของจุลินทรีย์ ทำให้ผลผลิตที่ได้จากการหมักลดลงหรืออาจเน่าเสียได้ ดังนั้นจึงจำเป็นต้องมีการควบคุมอุณหภูมิ โดยการควบคุมแบบ Fuzzy ได้มีผู้ที่ทำการวิจัยมาแล้วดังนี้

ในปี ค.ศ.1992 M. Mizumoto ได้เสนอเทคนิคการควบคุมแบบ Fuzzy PID ด้วย Product-sum-gravity method และ Simplified fuzzy reasoning method

ในปี ค.ศ.1996 J.S. Taur, C.W. Tao และ C.C. Tsai ได้เสนอวิธีการควบคุมอุณหภูมิของถังสักร์พลาสติกโดยใช้ PID Fuzzy Controllers

ในปี พ.ศ.1998 Jan Jantzen ได้เสนอเทคนิคในการ Tuning Fuzzy PID Controllers

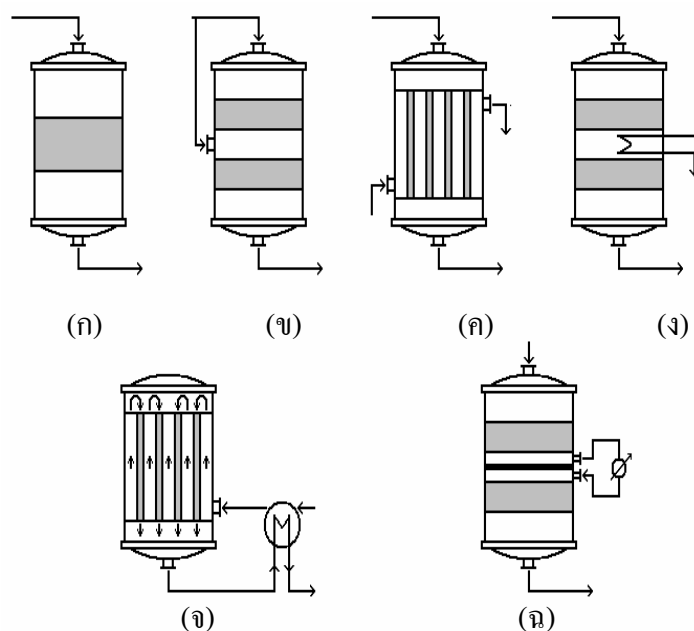
ในปี พ.ศ.2546 ปุณณา จารุรัตน์ ได้ทำการหาแบบจำลองทางคณิตศาสตร์การถ่ายเทความร้อนและมวลแบบลัมปีของการหมักแห้งถึงเหลือในถังหมักชนิดแพคเบด

ในปี พ.ศ.2546 พงษ์ประวัติ พิมพการัง ได้ทำการออกแบบอุปกรณ์ควบคุมอุณหภูมิอัตโนมัติแบบคำนวณตัวแปรภายในถังแพคเบด

ในปี พ.ศ.2547 ณัฐวุฒิ ชินธเนศ ได้ทำการหาแบบจำลองของระบบแขนหุ่นยนต์โดยใช้ Adaptive Neuro-Fuzzy Models

ถังแพคเบด

ถังแพคเบดเป็นถังปฏิกรณ์ชนิดหนึ่งที่ใช้กันมากในอุตสาหกรรมเคมีถังปฏิกรณ์ชนิดนี้มีลักษณะเป็นถังทรงกระบอกสูง ขนาดของถังขึ้นอยู่กับกรออกแบบให้มีความเหมาะสมกับการใช้งาน โดยคำนึงถึงสัดส่วนของความสูงต่อเส้นผ่านศูนย์กลางของถัง การนำถังแพคเบดไปใช้งานสามารถประยุกต์ใช้ได้หลากหลาย โดยมีจุดประสงค์การใช้งานที่แตกต่างกัน เช่น เพื่อเพิ่มพื้นที่ผิวสัมผัสและระยะเวลาในการเกิดปฏิกิริยาของสารตั้งต้นที่ไหลผ่านเบด, ใช้เป็นที่บรรจุตัวเร่งปฏิกิริยา หรือใช้ตรึงเอนไซม์ในกระบวนการเชิงวิศวกรรมชีวเคมี เป็นต้น ภาพที่ 1 แสดงตัวอย่างของถังแพคเบดแบบต่างๆ



ภาพที่ 1 ถังแพคเบดแบบต่างๆ ก) adiabatic, ข) interbed cold shot injection, ค) shell and tube, ง) built-in interbed heat exchanger, ฉ) external interbed exchanger, ช) autothermal shell; outside influent-effluent heat exchanger

ที่มา: พงษ์ประวัติ พิมพ์ารัง (2546)

ปัญหาที่ในการใช้งานถังแพคเบดคือ พฏิกิริยาภายในถังแพคเบดสามารถอธิบายได้ด้วยศาสตร์ต่างๆ ทางวิศวกรรมเคมี เช่น การถ่ายเทมวล (mass transfers), การถ่ายเทความร้อน (heat transfers), จลนศาสตร์การเกิดปฏิกิริยา (kinetics) เป็นต้น ปัญหาที่พบบ่อยจากการใช้ถังแพคเบดก็คือ การ

สะสมความร้อนภายในเบค เนื่องจากปฏิกิริยาที่เกิดขึ้นส่วนมากเป็นปฏิกิริยาแบบคายความร้อน (exothermic reaction) และการจัดเรียงตัวของของแข็งภายในเบคก่อนข้างหนาแน่นส่งผลให้การถ่ายเทความร้อนภายในเบคเป็นไปยาก ดังนั้นความร้อนจึงสะสมอยู่ภายในเบค สำหรับปฏิกิริยาที่เกี่ยวข้องกับตัวเร่งปฏิกิริยา ความร้อนที่สะสมจะทำให้อุณหภูมิภายในเบคสูงและเกิดอันตรายต่อระบบ ดังนั้นปฏิกิริยาการเจริญเติบโตของจุลินทรีย์ที่อุณหภูมิสูงจะส่งผลให้การทำงานของจุลินทรีย์ลดลงหรือตายได้ (เพ็ญจิตร และ มณีรัตน์, 2541) ดังนั้นการควบคุมความร้อนภายในถังแพคเบคจึงมีความสำคัญ และสามารถดำเนินการได้หลายวิธี แต่วิธีที่นิยมใช้เนื่องจากมีประสิทธิภาพและใช้งานง่าย มีอยู่ 2 วิธี คือ

1. การควบคุมความร้อนด้วยการนำความร้อน (conductive cooling) เช่น การควบคุมความร้อนโดยใช้น้ำหล่อเย็น (cooling water) ไหลผ่านแจ็กเก็ต (cooling jacket) หรือผ่านท่อขด (cooling coil)
2. การควบคุมความร้อนด้วยการพาความร้อน (convective cooling) เช่น การควบคุมความร้อนด้วยการพาความร้อนออกจากกระบอกโดยการเป่าหรือดูดอากาศ หรืออาศัยการระเหยของน้ำที่ใช้พลังงานภายในระบบในการเปลี่ยนสถานะ (evaporative cooling)

การหมักโคจิแบบแห้งในถังแพคเบค

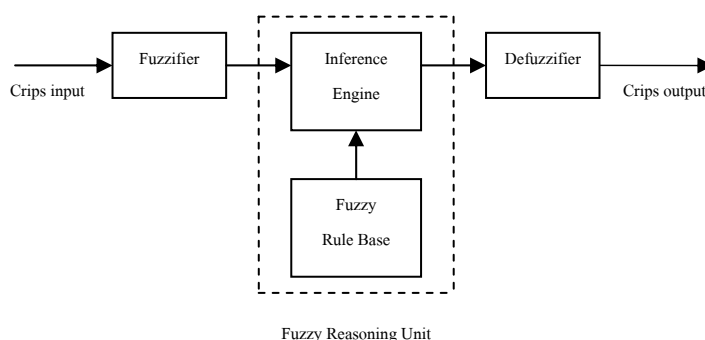
การหมักโคจิ เป็นขั้นตอนสำคัญในกระบวนการผลิตซีอิ๊วและเต้าเจี้ยว โดยปกติการผลิตซีอิ๊วและเต้าเจี้ยวจะประกอบไปด้วยขั้นตอนการผลิต 2 ขั้นตอน คือ การหมักโคจิ และการหมักโมโรมิเพื่อเปลี่ยนโคจิให้เป็นซีอิ๊วหรือเต้าเจี้ยว วัตถุดิบที่ใช้ในการหมักโคจิคือ ถั่วเหลือง ส่วนเชื้อจุลินทรีย์ที่ใช้ในการหมักคือ รา *Aspergillus oryzae* ดังภาพที่ 2 เนื่องจากราชนิดนี้เพาะเลี้ยงง่าย มีประสิทธิภาพสูงในการสร้างเอนไซม์ที่มีบทบาทต่อกลิ่นและรสชาติของซีอิ๊วและเต้าเจี้ยว ราชนิดนี้เป็นราประเภทที่ต้องการออกซิเจนในกระบวนการเจริญเติบโต ชอบสภาวะที่มีค่าความเป็นกรด/ด่างต่ำ อุณหภูมิที่เหมาะสมต่อการทำงานอยู่ในช่วง 25-30 องศาเซลเซียส และไม่ควรมากเกิน 40 องศาเซลเซียส ซึ่งจะทำให้ราตาย ความชื้นในช่วงเริ่มการหมักควรมีค่าเท่ากับ 50-70% (ปุ่นณา, 2546) กลไกที่เกิดภายในถังหมักโคจิคือ ราจะผลิตเอนไซม์เพื่อย่อยถั่วเหลืองและแบ่งให้เป็นสารโมเลกุลเล็กและละลายน้ำได้ ซึ่งเป็นแหล่งพลังงานของแบคทีเรียและยีสต์ในการหมักขั้นโมโรมิ



ภาพที่ 2 รา *Aspergillus oryzae* ซึ่งมีสีเหลืองแกมเขียว
ที่มา: พงษ์ประวัติ พิมพ์การัง (2546)

การควบคุมแบบ Fuzzy Inference System

1. โครงสร้างของ Fuzzy Inference System



ภาพที่ 3 แสดงโครงสร้างของ Fuzzy Inference System
ที่มา: ญัฐวุฒิ ชินธเนศ (2547)

1.1 Fuzzifier ประกอบด้วยหลาย ๆ fuzzy set ของแต่ละอินพุต โดยจะทำหน้าที่แปลง crips input ให้เป็น fuzzy input กล่าวคือ แปลงให้เป็นค่าความเป็นสมาชิกของแต่ละเซต

1.2 Fuzzy reasoning unit ประกอบด้วย inference engine, fuzzy rule base และ หลายๆ fuzzy set ของเอาต์พุต โดยในส่วนนี้จะทำหน้าที่ ในการหาความสัมพันธ์ระหว่างค่าความเป็นสมาชิกของอินพุต และค่าความเป็นสมาชิกของเอาต์พุต โดยในการหาความสัมพันธ์จะขึ้นอยู่กับ Fuzzy rule base ที่ถูกกำหนดขึ้นมา

1.3 Defuzzifier ทำหน้าที่ในการแปลงค่าความเป็นสมาชิกของเอาต์พุตให้เป็นค่า crisp output

Fuzzy Inference System ที่ถูกนำมาประยุกต์ใช้งานอย่างกว้างขวางมีอยู่ 3 ประเภท โดยทั้ง 3 ประเภทจะแตกต่างกันในส่วนของการ implication ของ if-then rule, ในส่วนของการรวม ผลของแต่ละ if-then rule เข้าด้วยกัน (aggregation) และในส่วนของการทำ defuzzification

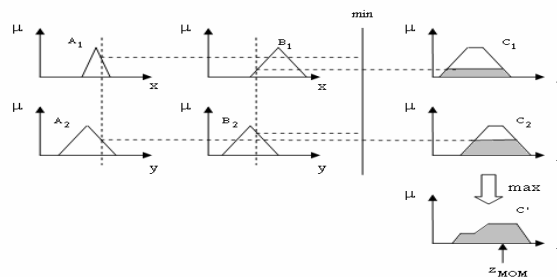
2. Mamdani Fuzzy Inference System

Mamdani Model เป็น Fuzzy Inference System ที่เป็นที่นิยมใช้งานอย่างกว้างขวางในระบบควบคุม สมมติให้ Mamdani Fuzzy Inference System มี 2 crisp input คือ $x = x_0$ และ $y = y_0$, มี 1 crisp output คือ z และกำหนดให้มีกฎจำนวน n ข้อ แล้วจะสามารถหา fuzzy reasoning ได้ดังทฤษฎีบทต่อไปนี้

ทฤษฎีบทที่ 1 กำหนดให้ fuzzy if-then rule ข้อที่ i คือ $R_i = A_i \times B_i \rightarrow C_i$ สำหรับการทำการ implication ใน if-then rule แบบ minimum และการรวมกฎแบบ maximum (max-min composition) แล้วจะได้ฟังก์ชันค่าความเป็นสมาชิกจากการรวมกฎ (aggregated output membership function) ดังนี้

$$\mu_{c'}(z) = \max_{i=1}^n \{\min(\omega_i, \mu_{c_i}(z))\} \quad (1)$$

โดยที่ $\omega_i = \min\{\mu_{A_i}(x_0), \mu_{B_i}(y_0)\}$



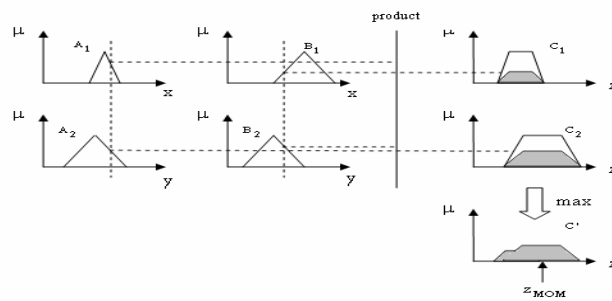
ภาพที่ 4 Mamdani Fuzzy Inference System (max-min composition) (ณัฐวุฒิ, 2547)

ที่มา: ณัฐวุฒิ ชินธเนศ (2547)

ทฤษฎีบทที่ 2 กำหนดให้ fuzzy if-then rule ข้อที่ i คือ $R_i = A_i \times B_i \rightarrow C_i$ สำหรับการ
ทำ implication ใน if-then rule แบบ product และ การรวมกฎแบบ maximum (max-product
composition) แล้วจะได้ฟังก์ชันค่าความเป็นสมาชิกจากการรวมกฎดังนี้

$$\mu_{c'}(z) = \max_{i=1}^n \{ \omega_i \cdot \mu_{C_i}(z) \} \quad (2)$$

โดยที่ $\omega_i = \min\{ \mu_{A_i}(x_0), \mu_{B_i}(y_0) \}$



ภาพที่ 5 Mamdani Fuzzy Inference System (max-product composition)

ที่มา: ณัฐวุฒิ ชินธเนศ (2547)

หลังจากได้ฟังก์ชันค่าความเป็นสมาชิกจากการรวมกฎ, $\mu_{c'}(z)$ จาก fuzzy reasoning unit
แล้ว ขั้นตอนต่อมาคือการทำ defuzzification กล่าวคือ จะต้องทำการเปลี่ยน fuzzy set ให้เป็น crisp
output ซึ่งในขั้นตอน defuzzification จะทำได้หลายวิธี ดังเช่น

2.1 Centroid of area (z_{COA})

$$z_{COA} = \frac{\int_Z \mu_{C'}(z) \cdot z \, dz}{\int_Z \mu_{C'}(z) \, dz} \quad (3)$$

2.2 Bisector of area (z_{BOA})

$$\int_{\alpha}^{z_{BOA}} \mu_{C'}(z) \, dz = \int_{z_{BOA}}^{\beta} \mu_{C'}(z) \, dz \quad (4)$$

โดยที่ $\alpha = \min\{z / z \in Z\}$ and $\beta = \max\{z / z \in Z\}$

2.3 Mean of maximum (z_{MOM})

$$z_{MOM} = \frac{\int_{z'} z dz}{\int_{z'} dz} \quad (5)$$

โดยที่ $z' = \{z / \mu_{C^*}(z) = \mu^*\}$ และ μ^* คือค่าที่มากที่สุดของฟังก์ชันค่าความเป็นสมาชิกจากการรวมกฎ

2.4 Smallest of maximum (z_{SOM}) : z_{SOM} คือ ค่า z ที่น้อยที่สุดที่ทำให้ $\mu_{C^*}(z)$ เท่ากับ ค่ามากที่สุดของฟังก์ชันค่าความเป็นสมาชิกจากการรวมกฎ

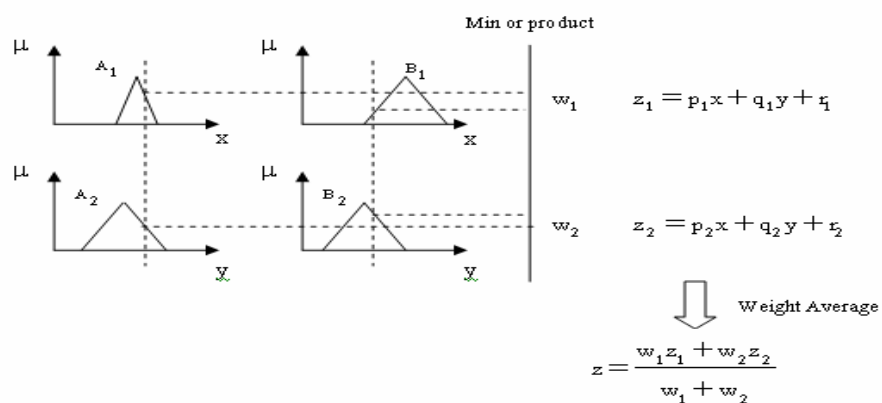
2.5 Largest of maximum (z_{LOM}) : z_{LOM} คือ ค่า z ที่มากที่สุดที่ทำให้ $\mu_{C^*}(z)$ เท่ากับ ค่ามากที่สุดของฟังก์ชันค่าความเป็นสมาชิกจากการรวมกฎ

3. Sugeno Fuzzy Inference System

Sugeno Fuzzy Inference System ถูกสร้างขึ้นมาเพื่อพยายามที่จะพัฒนา Fuzzy Inference System โดยการสร้าง fuzzy rule จาก input-output data set โดยทั่วไป fuzzy rule ของ Sugeno Fuzzy Inference System จะมีรูปแบบ

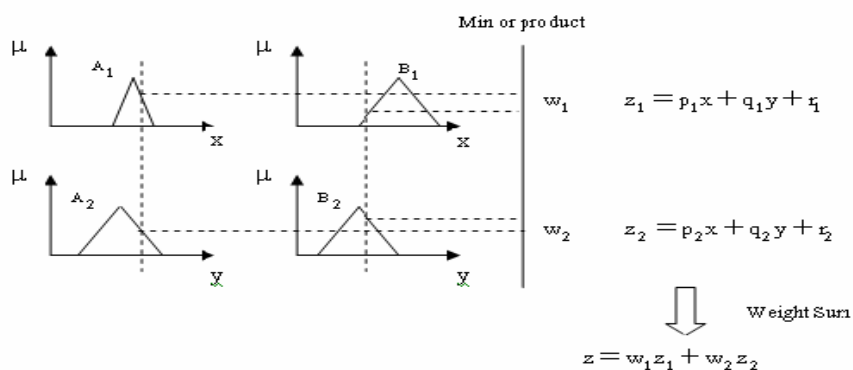
$$\text{ถ้า } x \text{ เป็น } A \text{ และ } y \text{ เป็น } B \text{ แล้ว } z = f(x,y)$$

โดยที่ A และ B เป็น fuzzy set ของ input และ $z = f(x,y)$ เป็น crisp function ในส่วนของ implication ของ fuzzy if-then rule ซึ่งโดยทั่วไปจะกำหนดให้เป็นสมการพหุนามของตัวแปร input x และ y ในส่วนของการรวมกฎ จะทำได้ 2 วิธี คือ weight average และ weight sum ภาพที่ 6 และ 7 แสดง Sugeno fuzzy reasoning สำหรับ 2 อินพุต- 2 กฎ โดยการรวมกฎเป็นแบบ weight average และแบบ weight sum ตามลำดับ



ภาพที่ 6 Sugeno Fuzzy Inference System (weight average)

ที่มา: ณัฐวุฒิ ชินชเนต (2547)

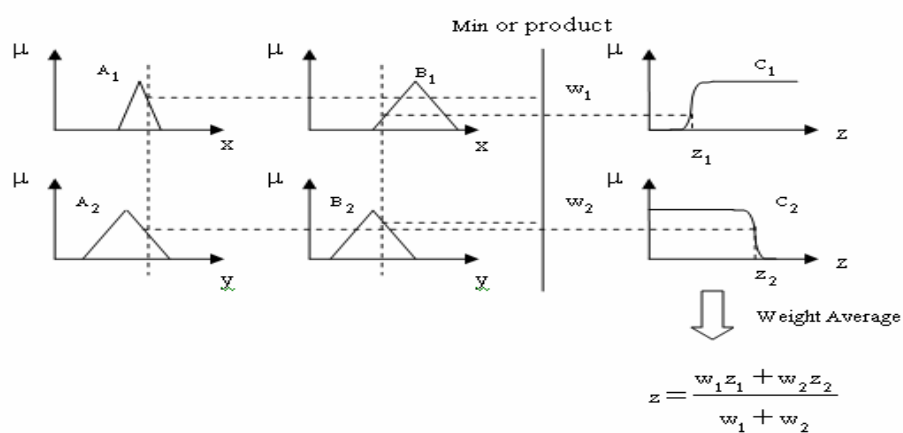


ภาพที่ 7 Sugeno Fuzzy Inference System (weight sum)

ที่มา: ณัฐวุฒิ ชินชเนต (2547)

4. Tsugamoto Fuzzy Inference System

สำหรับ Tsugamoto Fuzzy Inference System ในส่วน implication ของ fuzzy if-then rule จะอยู่ในรูปแบบ fuzzy set แบบ monotonical membership function ดังแสดงในรูปที่ 8 แล้วจะได้ crisp output ของกฎแต่ละข้อ จากนั้นก็นำมารวมกฎแบบ weight average ภาพที่ 8 แสดง Tsugamoto fuzzy reasoning สำหรับ 2 อินพุต- 2 กฎ และ รวมกฎแบบ weight average



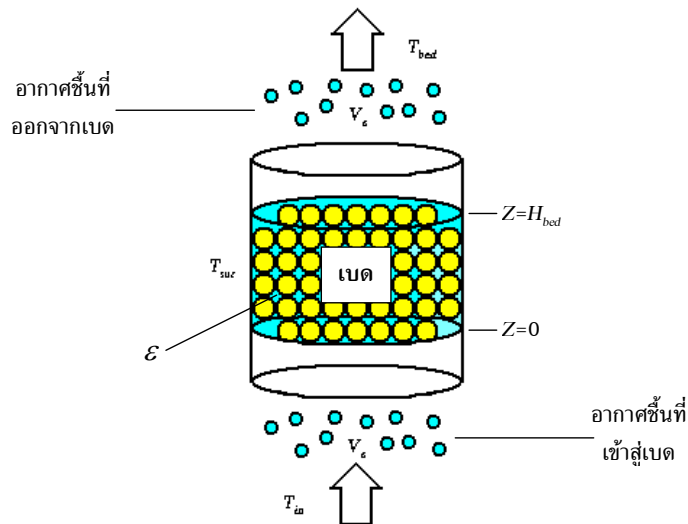
ภาพที่ 8 Tsugamoto Fuzzy Inference System

ที่มา: ณัฐวุฒิ ชินชเนต (2547)

แบบจำลองทางคณิตศาสตร์ของการหมักแบบแห้งในถังแพคเบด

1. การสร้างภาพลักษณ์ของถังแพคเบดต้นแบบ

ระบบที่ศึกษาเป็นระบบการหมักโคจิแบบแห้งด้วยถังแพคเบด โดยถังแพคเบดที่ศึกษามีลักษณะเป็นถังทรงกระบอกสูงบรรจุเบดไว้ภายในแบบเบดเดี่ยวดังภาพที่ 9 ลักษณะทางกายภาพของเบดจะมีความสูงเท่ากับ H_{bed} มีเส้นผ่านศูนย์กลางเท่ากับ D_{bed} องค์ประกอบของเบดในกระบวนการหมักโคจิประกอบไปด้วย เมล็ดธัญพืชที่ผ่านการคลุกกับสปอร์ของรา *Aspergillus oryzae*, แป้ง, น้ำ และอากาศ



ภาพที่ 9 ภาพลักษณ์ของถังแพคเบดต้นแบบ

ที่มา: ปุณณา จารุรัตน์ (2547)

ตั้งสมมุติฐานให้เบดมีลักษณะเป็นเสมือนเนื้อเดียวกัน (pseudo-homogeneous) และมีสัดส่วนช่องว่างเท่ากับ ε ในสถานะสมดุลอุณหภูมิของธัญพืช, น้ำ และอากาศ พิจารณาให้มีค่าเท่ากันเป็นอุณหภูมิของเบดคือ T_{bed} โดยมีค่าเปลี่ยนแปลงไปตามระยะเวลาของการหมัก

ในสถานะที่รามีอาหารอากาศและน้ำนี้ทำให้เกิดกระบวนการเมตาบอลิซึม คือเกิดการเผาผลาญอาหารเพื่อการเจริญเติบโตและเพิ่มจำนวนเซลล์ผลที่ตามมาคือ พลังงานความร้อนก๊าซ

คาร์บอนไดออกไซด์ และน้ำ พลังงานความร้อนที่เกิดจากกระบวนการเมตาบอลิซึมทำให้อุณหภูมิเบตสูงขึ้น จึงเกิดการถ่ายเทพลังงานความร้อนไปยังบริเวณที่มีอุณหภูมิต่ำกว่า คืออากาศที่ไหลผ่านเบตและผนังของถังแพคเบต

ดังนั้นการควบคุมความร้อนของเบตทำได้โดยผ่านอากาศที่มีอุณหภูมิค่า T_{in} เข้าสู่เบตทางด้านล่างของถัง อากาศจะผ่านผิวน้ำที่บรรจุอยู่ที่ก้นถังเพื่อปรับความชื้น เมื่ออากาศชื้นซึ่งมีอุณหภูมิต่ำกว่าอุณหภูมิเบตไหลผ่านเบตจะพาความร้อนออกทางด้านบนแล้วออกจากถังสู่สิ่งแวดล้อมที่มีอุณหภูมิ T_{sur} ความเร็วอากาศที่ไหลผ่านเบตจะพิจารณาความเร็วในถังเปล่า (superficial velocity, V_z) เพื่อนำไปใช้คำนวณในแบบจำลองทางคณิตศาสตร์

2. การพัฒนาแบบจำลองทางคณิตศาสตร์ของการหมักแบบแห้ง

สำหรับงานวิจัยนี้ได้พิจารณาแบบจำลองทางคณิตศาสตร์ของพงศ์ประวัติ (2546) แบบจำลองทางคณิตศาสตร์ของการหมักแบบแห้งประกอบด้วยสมการหลัก 2 กลุ่มคือ สมการจลนพลศาสตร์การเจริญของจุลินทรีย์แบบลอจิสติก (logistic growth kinetic equation) และสมการถ่ายเทความร้อน (heat transfer equation)

2.1 สมการจลนพลศาสตร์การเจริญของจุลินทรีย์แบบลอจิสติก

ติดตามอัตราการเจริญเติบโตของปริมาณชีวมวลจากสมการที่ 6 โดยมีอัตราการเจริญจำเพาะ μ ตามสมการที่ 7 ซึ่งเป็นสมการความสัมพันธ์ที่ได้จากการทดลอง (ปุ่นณา, 2546)

$$\frac{dX}{dt} = \mu x \left(1 - \frac{X}{X_{\max}} \right) \quad (6)$$

$$\mu = 90.4683 \exp\left(\frac{-4.5535}{RT}\right) \quad (7)$$

เมื่อ X คือปริมาณชีวมวล, $X_{\max}$ คือปริมาณชีวมวลที่สามารถเกิดขึ้นสูงสุด, R คือค่าคงที่ของแก๊สมีค่าเท่ากับ 0.001986 กิโลแคลอรี/กรัม โมล-เคลวิน และ T คืออุณหภูมิเบตหน่วย เคลวิน จากสมการที่ 6 สามารถหาค่าของปริมาณชีวมวล X จากผลเฉลยแม่นยำตรงดังสมการที่ 8

$$X = \frac{X_{\max}}{1 + \left(\frac{X_{\max}}{X} - 1 \right) e^{(-\mu t)}} \quad (8)$$

เมื่อ X คือปริมาณชีวมวลเริ่มต้นและ t คือเวลาที่ผ่านไปนับตั้งแต่เริ่มต้นการหมักหน่วย ชั่วโมง

2.1 สมการถ่ายเทความร้อน

ในสมการถ่ายเทความร้อนแบบลัมปีประกอบไปด้วยเทอมที่อธิบายการเปลี่ยนแปลงพลังงานในภาพรวมทั้งหมดระหว่างระบบที่ศึกษาและสิ่งแวดล้อมรอบระบบ โดยไม่คำนึงถึงกลไกการเปลี่ยนแปลงที่เกิดขึ้นภายในระบบดังสมการที่ 9

$$\begin{aligned} (\rho_{bed} Cp_{bed} + (1-\varepsilon)\rho_{drymat} Cp_{water} W) \frac{dT}{dt} = & \frac{V_z}{H_{bed}} \rho_{moist} Cp_{moist} (T_{in} - T_{bed}) \\ & + \frac{V_z}{H_{bed}} \rho_{moist} \lambda (Y_{W_{in}} - Y_{W_{out}}) \\ & + (1-\varepsilon)\rho_{drymat} y_{Q/X} \frac{dX}{dt} \\ & + (1-\varepsilon)\rho_{drymat} m_{Q/X} X \end{aligned} \quad (9)$$

- เมื่อ ρ_{bed} คือความหนาแน่นของเบดหน่วย กิโลกรัม/เมตร³
 Cp_{bed} คือความจุความร้อนของเบดหน่วย จูล/กิโลกรัม-เคลวิน
 ε คือสัดส่วนช่องว่างหรือค่าความพรุนของเบด
 ρ_{drymat} คือความหนาแน่นของวัสดุหมักหน่วย กิโลกรัม/เมตร³
 Cp_{water} คือความจุความร้อนของน้ำหน่วย จูล/กิโลกรัม-เคลวิน
 W คือสัดส่วนน้ำในเบดหน่วย กิโลกรัมน้ำ/กิโลกรัมวัสดุหมัก
 $\frac{dT}{dt}$ คืออัตราการเปลี่ยนแปลงอุณหภูมิเบด
 V_z คือความเร็วอากาศหน่วย เมตร/ชั่วโมง
 H_{bed} คือความสูงของเบดหน่วย เมตร
 ρ_{moist} คือความหนาแน่นของอากาศชื้นหน่วย กิโลกรัม/เมตร³

Cp_{moist}	คือความจุความร้อนของอากาศชื้นหน่วย จูล/กิโลกรัม-เคลวิน
T_{in}	คืออุณหภูมิอากาศขาเข้าหน่วย เคลวิน
T_{out}	คืออุณหภูมิอากาศขาออกหน่วย เคลวิน
λ	คือความร้อนแฝงของการกลายเป็นไอน้ำหน่วย จูล/กิโลกรัม
$Y_{W_{in}}$	คือสัดส่วนน้ำในอากาศขาเข้า
$Y_{W_{out}}$	คือปริมาตรสัดส่วนน้ำในอากาศขาออก
$y_{Q/X}$	คือสัมประสิทธิ์ผลได้ของพลังงานจากจุลินทรีย์หน่วย จูล/กิโลกรัมชีวมวล
$m_{Q/X}$	คือสัมประสิทธิ์ผลได้ของพลังงานจากจุลินทรีย์ในระยะคงที่หน่วย จูล/กิโลกรัมชีวมวล-ชั่วโมง

สมการถ่ายเทความร้อนเป็นการคลพลังงาน โดยฝั่งซ้ายของสมการเป็นพลังงานสะสมในระบบประกอบไปด้วยเทอมของพลังงานสะสมในเบด $\rho_{bed} Cp_{bed} \frac{dT}{dt}$ ซึ่งเป็นส่วนของวัสดุหมักกับอากาศและเทอมพลังงานสะสมในน้ำที่อยู่ในเบด $(1-\varepsilon)\rho_{drymat} Cp_{water} W \frac{dT}{dt}$ ส่วนทางฝั่งขวาของสมการเป็นพลังงานที่เกิดการเปลี่ยนแปลงภายในระบบประกอบไปด้วย เทอมของการเปลี่ยนแปลงพลังงานเนื่องจากอากาศที่ไหลผ่านระบบ $\frac{V_z}{H_{bed}} \rho_{moist} Cp_{moist} (T_{in} - T_{bed})$, เทอมของการเปลี่ยนแปลงพลังงานด้วยกลไกการระเหยของน้ำที่เข้าออกระบบ $\frac{V_z}{H_{bed}} \rho_{moist} \lambda (Y_{Win} - Y_{Wout})$, เทอมของการเปลี่ยนแปลงพลังงานเนื่องจากปฏิกิริยาการเจริญเติบโตของจุลินทรีย์ในระยะการเจริญเติบโตแบบลอการิทึม (logarithmic phase) $(1-\varepsilon)\rho_{drymat} y_{Q/X} \frac{dX}{dt}$ และเทอมของการเปลี่ยนแปลงพลังงานปฏิกิริยาการเจริญเติบโตของจุลินทรีย์ในระยะคงที่ (stationary phase) $(1-\varepsilon)\rho_{drymat} m_{Q/X} X$

นอกจากนี้ค่าของสัดส่วนน้ำในอากาศ Y_W หาได้จากความดันไออิ่มตัว P_W ซึ่งสัมพันธ์กับอุณหภูมิตามสมการความสัมพันธ์ขององตวน (Antoine Equation) ดังสมการที่ 10

$$\log P_w = A - \left(\frac{B}{T_C - 235} \right) \quad (10)$$

เมื่อ P_w คือความดันไออิ่มตัวหน่วยเป็นมิลลิเมตรปรอท, T_C คืออุณหภูมิหน่วย องศาเซลเซียส และ A กับ B เป็นค่าคงที่ไว้หน่วยที่ขึ้นกับช่วงอุณหภูมิที่พิจารณา หลังจากนั้น สามารถแปลงค่าความดันไออิ่มตัว P_w เป็นปริมาณน้ำในอากาศ Y_w ได้ด้วยสมการที่ 11

$$Y_w = 0.622 \frac{P_w}{P_{atm} - P_w} \quad (11)$$

เมื่อ P_{atm} คือความดันบรรยากาศ

และจากสมมุติฐานที่ว่าวัสดุหมักเสมือนเป็นเนื้อเดียวกันทั้งคุณสมบัติทางความร้อนและคุณสมบัติทางกายภาพ ดังนั้นการหาค่าความหนาแน่นของเบด ρ_{bed} และค่าความจุความร้อน ของเบด Cp_{bed} หาได้จากค่าเฉลี่ยของวัสดุหมักและอากาศชั้นภายในเบด ดังสมการที่ 12 และ สมการที่ 13

$$\rho_{bed} = (1 - \varepsilon) \rho_{drymat} + \varepsilon \rho_{moist} \quad (12)$$

$$Cp_{bed} = (1 - \varepsilon) Cp_{drymat} + \varepsilon Cp_{moist} \quad (13)$$

อุปกรณ์และวิธีการ

อุปกรณ์

1. อุปกรณ์ในการควบคุมอุณหภูมิในถังแพคเบด
 - 1.1 เครื่องคอมพิวเตอร์ตั้งโต๊ะ
 - 1.2 โปรแกรม LabVIEW 8.5
 - 1.3 โปรแกรม MATHLAB 6.5
 - 1.4 บอร์ดควบคุม PIC24HJ128GP306
 - 1.5 อุปกรณ์แปลงสัญญาณ USB เป็น RS485 (USB to RS485 converter)
 - 1.6 เครื่องดูดลม (vortex blower) ของ Hitachi รุ่น VB-0007-DN
 - 1.7 Compact inverter ของ Fuji electric รุ่น FRNO. 75C1S-4A
2. วัสดุและอุปกรณ์สำหรับหมักโคจิ
 - 2.1 ถั่วเหลืองแห้ง 2 กิโลกรัมต่อการทดลอง
 - 2.1 แป้งสาลี 200 กรัมต่อการทดลอง
 - 2.3 รา Aspergillus oryzae 3 กรัมต่อการทดลอง
 - 2.4 แผ่นกันเบด 3 แผ่น
 - 2.5 เครื่องกระจายอากาศ
 - 2.6 แผ่นกรองอากาศ
 - 2.7 ถังแพคเบดสแตนเลสเบอร์ 304 ขนาด 40 ลิตร

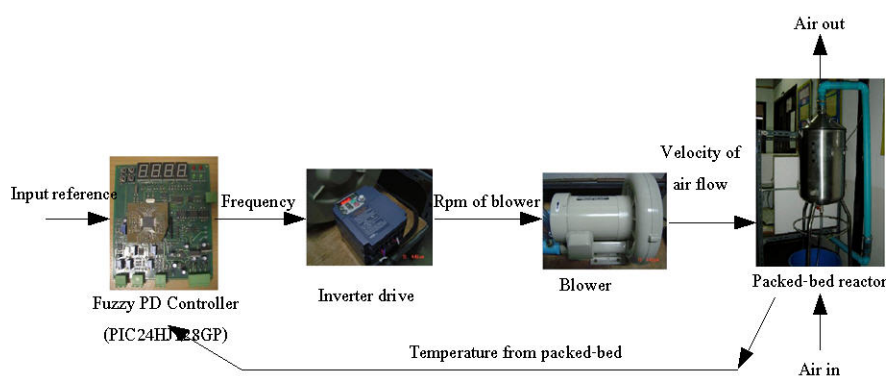
วิธีการ

ในวิทยานิพนธ์นี้ได้มีการศึกษาและค้นคว้าตามแนวทางของวัตถุประสงค์การศึกษา กระบวนการหมักโคจิในถังแพคเบดในวิทยานิพนธ์นี้ใช้แบบจำลองทางคณิตศาสตร์การถ่ายเทความร้อนและมวลแบบลัมปีของการหมักแห้งถั่วเหลืองในถังหมักชนิดแพคเบด(ปูลัณหา, 2546)

โดยในขั้นแรกจะเป็นการศึกษาและออกแบบตัวควบคุม FuzzyPID บนโปรแกรม MATLAB หลังจากนั้นผู้วิจัยจะทำการออกแบบบอร์ดควบคุมเองทั้งหมด โดยใช้ไมโครคอนโทรลเลอร์ PIC24HJ128 จากนั้นเราจะใช้บอร์ดควบคุมที่ได้พัฒนาออกแบบในการทดลองหมักโคจิในการหมักจริงเพื่อทำการทดลองตัวควบคุมที่ออกแบบ

1. การทำงานของระบบโดยรวม

ภาพรวมของระบบที่ใช้ในการทดลองประกอบแสดงให้เห็นดังรูป



ภาพที่ 10 แสดงการทำงานของระบบที่ทำการศึกษา

บอร์ดควบคุม PIC24HJ128GP จะทำการอ่านค่าอุณหภูมิของถังแพคเบดด้วยเซ็นเซอร์ 2 จุดและนำมาประมวลผลด้วย Fuzzy PID controller และขับสัญญาณอะนาล็อก Output ให้กับ Inverter drive เพื่อควบคุมความเร็วลมของ Blower ที่ใช้ในการดูดอากาศผ่านถังแพคเบด โดยสมการความสัมพันธ์ของสัญญาณอะนาล็อก Output ที่จ่ายให้กับ Inverter drive กับความเร็วของตัว Inverter drive แสดงดังสมการที่ 14 ด้านล่าง

$$f = \frac{I_{out} - 4}{16} \times 50 \quad (14)$$

สมการแสดงความสัมพันธ์ของ ความถี่ของ Inverter drive กับความเร็วลมที่ไหลผ่านถังแพคเบดคือ

$$V_z = m \times f \quad (15)$$

โดยที่ m มีค่าเท่ากับ 0.26 เป็นค่าคงที่ได้จากการทดลอง (พงศัประวัตติ, 2546)

2. การศึกษาและออกแบบตัวบน matlab simulink 6.5

จากแบบจำลองทางคณิตศาสตร์การถ่ายเทความร้อนและมวลแบบลัมปีของการหมักแห้ง ถึงเหลือในถังหมักชนิดแพคเกจสมการที่ 6, 7, 8 และ 9 ทำการจัดรูปใหม่ในรูปของ State Space Equation ได้ดังสมการที่ 16 และสมการที่ 17

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{T} \end{bmatrix} &= \begin{bmatrix} \mu x \left(1 - \frac{x}{x_{\max}}\right) \\ \frac{C_2}{C_1} T_{in} V_z - \frac{C_2}{C_1} T V_z + \frac{C_3}{C_1} Y_{win} V_z - \frac{C_3}{C_1} Y_{wout} V_z + \frac{C_4}{C_1} \dot{x} + \frac{C_5}{C_1} x \end{bmatrix} \\ &= \begin{bmatrix} 0 & \mu \left(1 - \frac{x}{x_{\max}}\right) \\ -\frac{C_2}{C_1} V_z & \frac{C_4}{C_1} \mu \left(1 - \frac{x}{x_{\max}}\right) + \frac{C_5}{C_1} \end{bmatrix} \begin{bmatrix} x \\ T \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{C_2}{C_1} T_{in} + \frac{C_3}{C_1} Y_{win} + \frac{C_3}{C_1} Y_{wout} \end{bmatrix} V_z \end{aligned} \quad (16)$$

$$Y = [0 \quad 1] \begin{bmatrix} X \\ T \end{bmatrix} \quad (17)$$

เมื่อ C_1, C_2, C_3, C_4 และ C_5 เป็นค่าคงที่โดยมีค่าดังนี้

$$C_1 = \rho_{bed} C_{p_{bed}} + (1 - \varepsilon) \rho_{drymat} C_{p_{water}} W \quad (18)$$

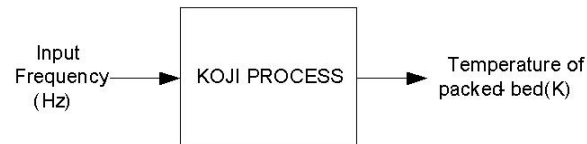
$$C_2 = \frac{\rho_{moist} C_{p_{moist}}}{H_{bed}} \quad (19)$$

$$C_3 = \frac{\rho_{moist} \lambda}{H_{bed}} \quad (20)$$

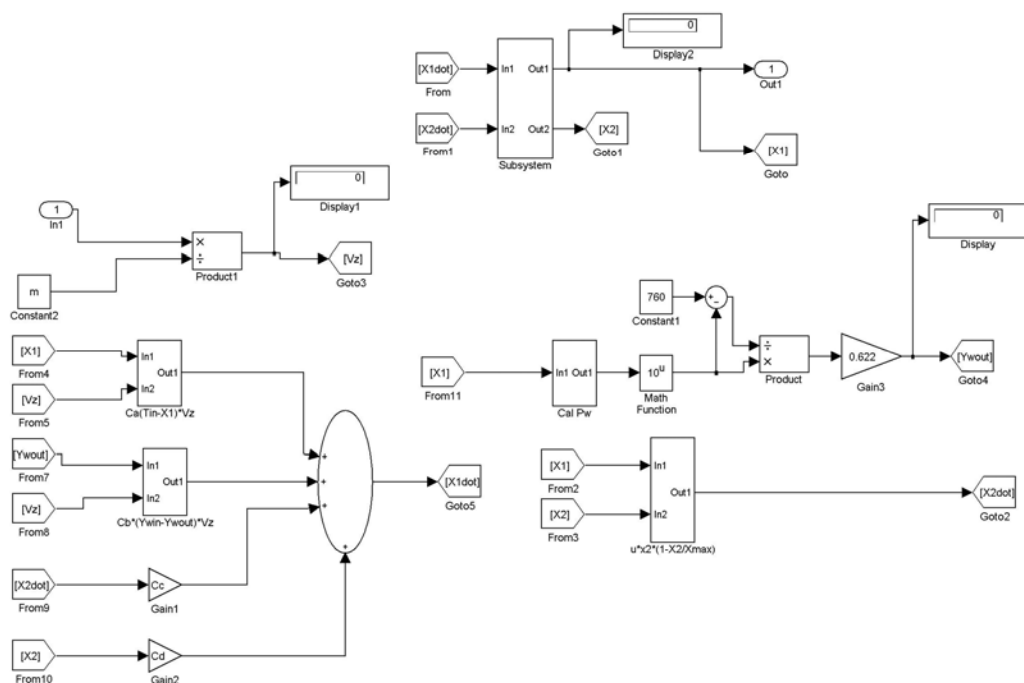
$$C_4 = (1 - \varepsilon) \rho_{drymat} C_{p_{water}} y_{O/X} \quad (21)$$

$$C_5 = (1 - \varepsilon) \rho_{drymat} m_{O/X} \quad (22)$$

โดยในการ Simulate เราให้ T_{in} มีค่าคงที่และ Y_w หาได้จากสมการที่ 10 และสมการที่ 11 V_z หาได้จากสมการที่ 15 และที่คงที่ต่างๆใช้ตามผลงานวิจัยของพงษ์ประวัดี, 2546 โดย Input ของระบบให้เป็น Frequency (Hz) และ Output ของระบบเป็นอุณหภูมิของเบด แสดงดังภาพที่ 11 และภาพที่ 12 แสดงรายละเอียดของระบบที่เขียนบน Simulink



ภาพที่ 11 แสดงการทำงานของระบบที่เขียนขึ้นบน Simulink



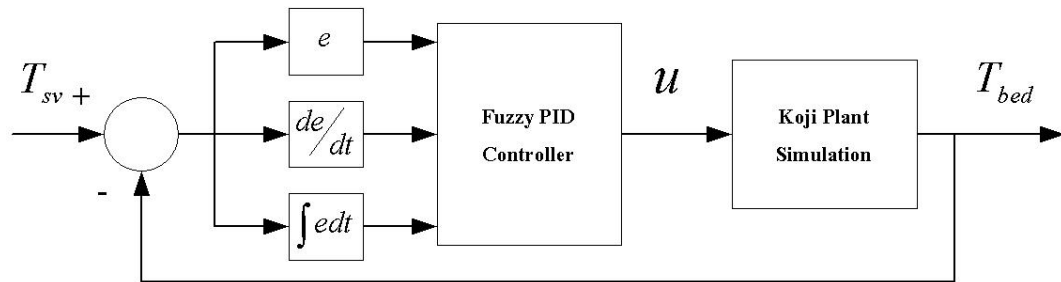
ภาพที่ 12 แสดงรายละเอียดของโปรแกรมระบบที่เขียนบน Simulink

การให้ค่าตัวแปรคงที่และการให้ค่าเริ่มต้นสำหรับตัวแปรต่างๆ เขียนใน m-file โดยการใช้
งาน ต้องทำการรัน m-file ก่อนจึงจะทำการรัน Simulink file ได้โดยรายละเอียดของ m-file มีดังนี้

```
% this file for initial const parameter of simulink of packsim %
clc;
clear all;
A=8.10765 %
B=1750.286 %
Patm=760 % mmHg
W=0.55
e=0.3
X0=0.1437
Xmax=0.3881
Pdrymat=2.18e+3 % kg/m^3
Cpdrymat=2.34e+3 % J/kg*K
Pmoist=1.14 % kg/m^3
Cpmoist=1.05e+3 % J/kg*K
Cpwater=4.18e+3 % J/kg*K
lamda=2.41e+6 % J/kg
yqm=2.99e+6 % J/kg % try value by you self
mqx=(3.29e+4) % J/kg*h
R=0.001986 % kcal/kmol*K
%R=1.986e+3 % kcal/kmol*K
Pbed=(1-e)*Pdrymat+e*Pmoist % kg/m^3
Cpbed=(1-e)*Cpdrymat+e*Cpmoist % J/kg*K
Hbed=0.1 % m
m=0.26 % hz/m/s f=m*Vz
f=5 % hz
%-----Initial Input const Parameter-----%
Tin=29+273.15 % K
Pwin=10^( A-(B/(Tin-273.15+235)) )
Ywin=0.622*( Pwin/(760-Pwin) )
C1=Pbed*Cpbed+(1-e)*Pdrymat*Cpwater*W
C2=(Pmoist*Cpmoist)/Hbed
C3=(Pmoist*lamda)/Hbed
C4=(1-e)*Pdrymat*yqm
C5=(1-e)*Pdrymat*mqx
Ca=C2/C1
Cb=C3/C1
Cc=C4/C1
Cd=C5/C1
X1o=27+273.15
X2o=0.1437
```

3. ออกแบบตัวควบคุม Fuzzy PID

เลือก Input Fuzzy PID เป็นค่า e (error), $\int edt$, de/dt โดยที่โครงสร้างของตัวควบคุมแสดงรายละเอียดดังภาพที่ 13



ภาพที่ 13 แสดงโครงสร้างของระบบ Fuzzy ที่ทำการศึกษา

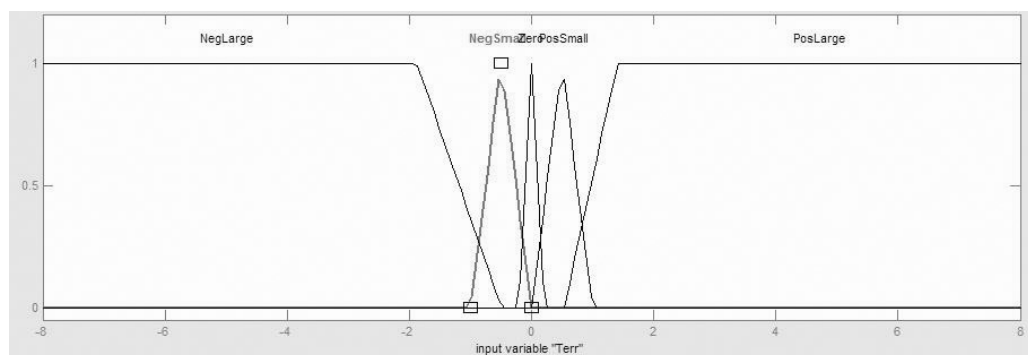
โดยที่ T_{sv} คืออุณหภูมิของถังแพคเบดที่ต้องการ (เคลวิน)

u คือ Output Fuzzy PID (Hz)

T_{bed} คืออุณหภูมิของถังแพคเบด (เคลวิน)

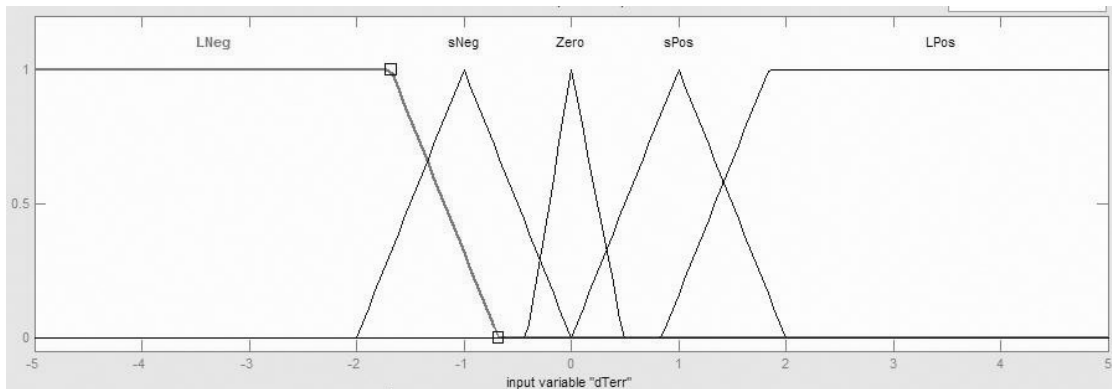
$$e = T_{sv} - T_{bed}$$

ออกแบบ Input Fuzzy PID โดยเลือก input e ให้มี 5 membership function คือ Large neg, Small neg, Zero, Small Pos, Large pos แสดงดังภาพที่ 14



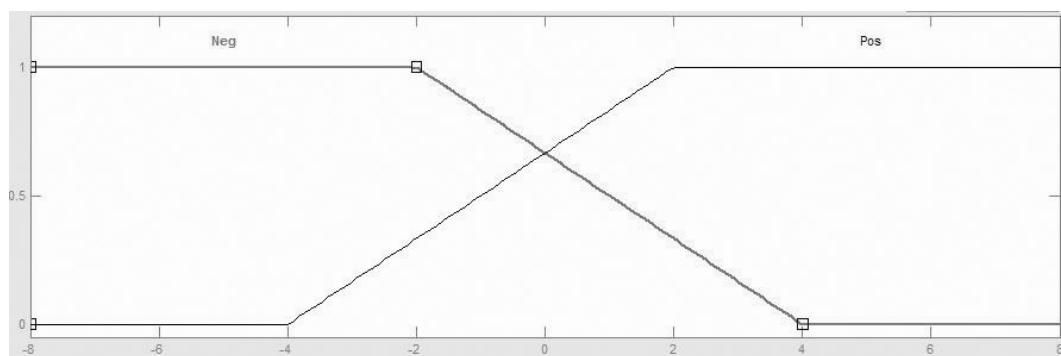
ภาพที่ 14 แสดง Membership function ของ input fuzzy e

เลือก Input de/dt ให้มี 5 membership function คือ Neg, Zero, Pos แสดงดังภาพที่ 15



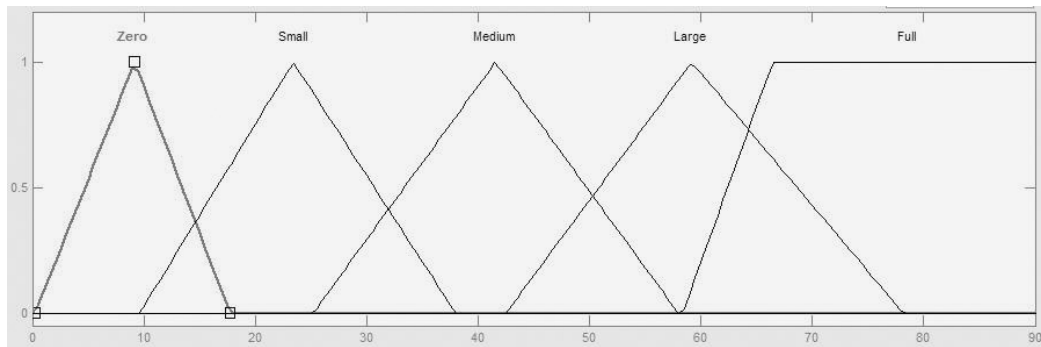
ภาพที่ 15 แสดง Membership function ของ input Fuzzy de/dt

เลือก Input $\int edt$ ให้มี 2 membership function คือ Neg, Pos แสดงดังภาพที่ 16



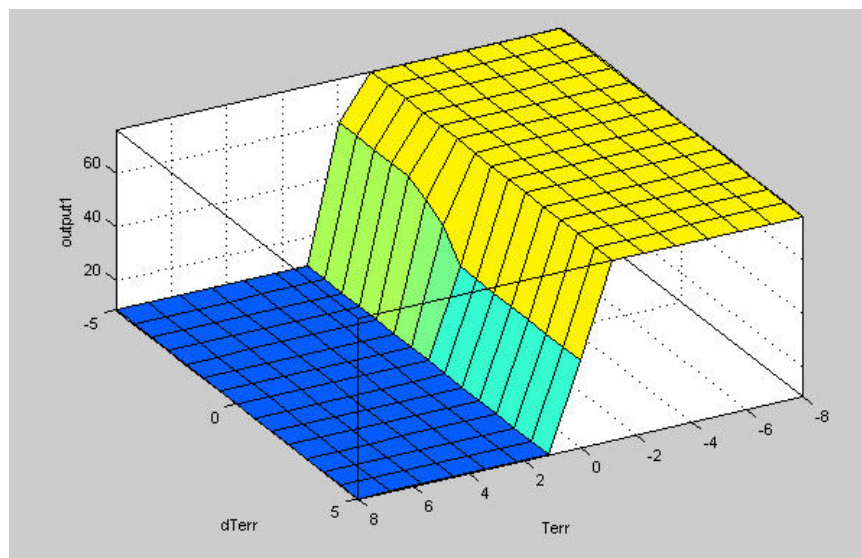
ภาพที่ 16 แสดง Membership function ของ input Fuzzy $\int edt$

เลือก Membership function สำหรับ output fuzzy u คือ Zero, Small, Medium, Large และ Full ดังแสดงในภาพที่ 17



ภาพที่ 17 แสดง Membership function ของ Output Fuzzy u

ออกแบบ Fuzzy Rule ด้วยได้ Control Surface ดังรูปที่ 18 และทำ defuzzification ด้วย Centroid of area Method (Z_{COA})



ภาพที่ 18 แสดง Control Surface ของ Fuzzy PID

Fuzzy Rule ที่ได้ทำการออกแบบแสดงให้ห้เห็นดังภาพที่ 19 และ 20

Input Diff error (de)						
Input error (e)	Output (u)	PosLarge	PosSmall	Zero	NegSmall	NegLarge
	PosLarge	Zero	Zero	Zero	Zero	Zero
	PosSmall	Zero	Zero	Zero	Small	Small
	Zero	Medium	Medium	Medium	Medium	Medium
	NegSmall	Large	Large	Large	Large	Large
	NegLarge	Full	Full	Full	Full	Full

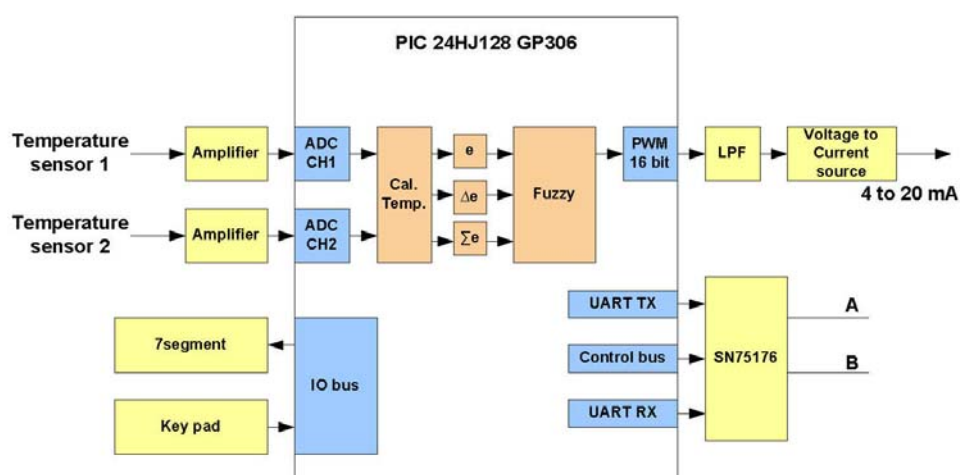
ภาพที่ 19 แสดง Fuzzy rule สำหรับ input e , de และ $\int edt$ เป็น Pos

Input Diff error (de)						
Input error (e)	Output (u)	PosLarge	PosSmall	Zero	NegSmall	NegLarge
	PosLarge	Zero	Zero	Zero	Zero	Zero
	PosSmall	Zero	Small	Small	Small	Small
	Zero	Medium	Medium	Medium	Medium	Medium
	NegSmall	Large	Large	Large	Large	Large
	NegLarge	Full	Full	Full	Full	Full

ภาพที่ 20 แสดง Fuzzy rule สำหรับ input e , de และ $\int edt$ เป็น Neg

4. ภาพรวมของการออกแบบบอร์ดควบคุมเพื่อใช้ในกระบวนการหมัก

การทำงานของบอร์ดควบคุมคือ อ่านค่าอุณหภูมิของวัสดุหมักแล้วนำค่าที่อ่านได้ไปคำนวณหาค่าความถี่ f ที่ต้องจ่ายให้กับ inverter drive เพื่อควบคุมอุณหภูมิของวัสดุหมักให้มีค่าอยู่ที่ Setpoint ที่ต้องการ โดยองค์ประกอบหลักของบอร์ดควบคุมแสดงให้ห็นดังภาพที่ 21



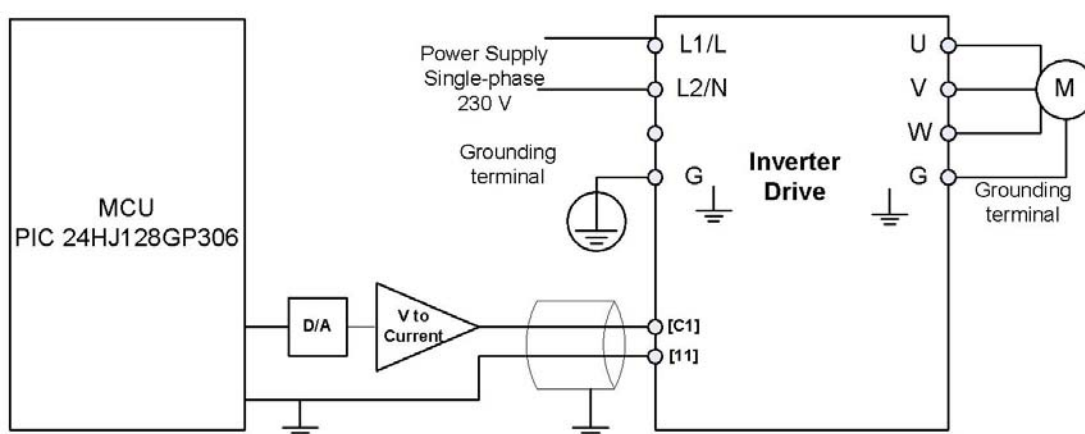
ภาพที่ 21 แสดงองค์ประกอบหลักในบอร์ดควบคุม

จากภาพแสดงให้เห็นส่วนประกอบสำคัญในบอร์ดควบคุมคือ MCU PIC 24HJ128GP306 ซึ่งเป็น 16 bit microcontroller ที่มีคุณสมบัติที่สำคัญในการคำนวณคือมี 16x16 Multiply operations, 32/16 และ 16/16 divide operations นอกจากนี้ยังมี Hardware DMA Module ที่ช่วยควบคุมการทำงานของ Peripheral (UART, I2C, SPI, A/D) แทน CPU

5. ออกแบบส่วน Interface กับ Inverter driver ของบอร์ดควบคุม

ในงานวิจัยนี้ใช้ MCU PIC24HJ128GP306 เป็นตัวประมวลผลและการควบคุมการทำงานของ inverter drive FRNO.75C1S-4A โดยการควบคุมความถี่ของ Inverter Drive สามารถทำได้โดยผ่าน Card Interface Rs485 และ Analog Interface แต่เนื่องจาก Inverter Drive ที่ใช้ในงานวิจัยนี้ไม่มี Card Interface Rs485 จึงจะขอกล่าวถึง Analog Interface เท่านั้น

5.1 การส่งสัญญาณแบบอะนาล็อก 4 ถึง 20 mA



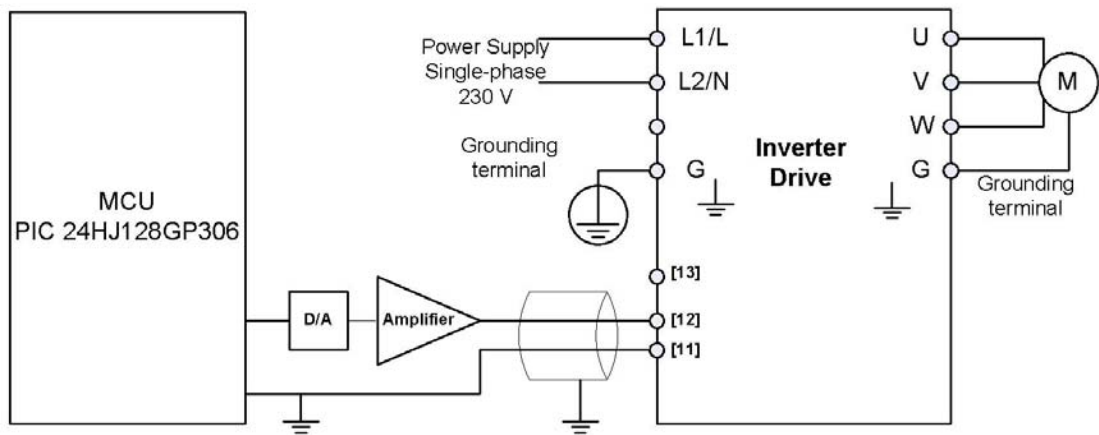
ภาพที่ 22 แผนภาพแสดงรายละเอียดการส่งสัญญาณ 4 ถึง 20 mA ไปควบคุม Drive

การตั้งค่าตัวแปรของ Inverter Drive สำหรับรับสัญญาณ 4 ถึง 20 mA ทำได้ดังนี้

5.1.1 ตั้งค่า Parameter F01 เป็น 2 เพื่อให้ Inverter Driver อ่าน Current input จาก Terminal [C1]

5.1.2 ตั้งค่า F18 และ C50 เป็น 0, C32 และ C34 เป็น 100

5.2 การส่งสัญญาณแบบอนาล็อก 0 ถึง 10 V



ภาพที่ 23 แผนภาพแสดงรายละเอียดการส่งสัญญาณ 0 ถึง 10 V ไปควบคุม Drive

การตั้งค่าตัวแปรของ Inverter Drive สำหรับรับสัญญาณ 0 ถึง 10 V ทำได้ดังนี้

5.2.1 ตั้งค่า Parameter F01 เป็น 2 เพื่อให้ Inverter Drive อ่าน Volt input จาก Termianl

[12]

5.2.2 ตั้งค่า F18 และ C50 เป็น 0, C32 และ C34 เป็น 100

การตั้งค่าตัวแปรของ Inverter Drive สำหรับรับสัญญาณ 0 ถึง 5 V ทำได้ดังนี้

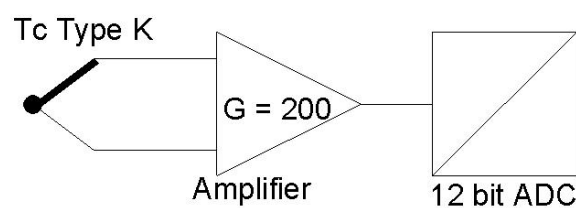
5.2.3 ตั้งค่า Parameter F01 เป็น 2 เพื่อให้ Inverter Drive อ่าน Volt input จาก Termianl

[12]

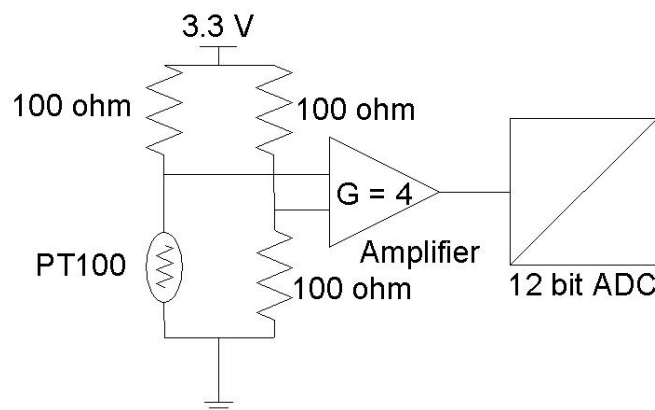
5.2.4 ตั้งค่า F18 และ C50 เป็น 0, C32 เป็น 100 และ C34 เป็น 50

6. ออกแบบส่วนวงจรอ่านอุณหภูมิของบอร์ดควบคุม

การอ่านค่าอุณหภูมิของบอร์ดควบคุมทำได้มีการออกแบบมา 2 แบบเพื่อทดลองใช้งาน โดยจะใช้ส่วนของ Opamp ขยายสัญญาณมาให้กับ ADC เพื่ออ่านค่าเข้าใน MCU แล้วทำการประมวลผลออกมาเป็นค่าอุณหภูมิดังที่แสดงในภาพที่ 24 และ ภาพที่ 25 ความละเอียดของ ADC มีค่าเท่ากับ 0.805 mV / bit



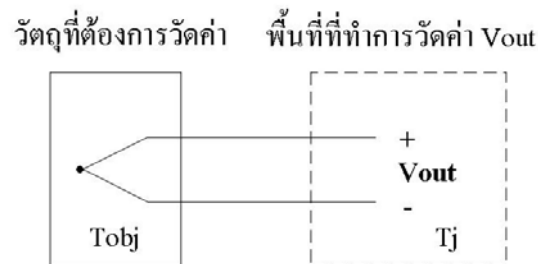
ภาพที่ 24 แสดงวงจรที่ใช้สำหรับอ่านค่า Thermocouple Type k



ภาพที่ 25 แสดงวงจรที่ใช้สำหรับอ่านค่า PT100

6.1 Thermocouple sensor type K เป็น sensor ที่ใช้สำหรับการวัดค่าอุณหภูมิ โดยที่ตัว sensor ประกอบด้วยวัสดุของชนิดที่ต่างกันได้เชื่อมต่อเข้าด้วยกันที่ปลายข้างหนึ่งอีกข้างหนึ่งมี ไขว่

สำหรับอ่านค่า mV ที่เป็น output ของ sensor เมื่อนำปลายข้างที่เชื่อมต่อกัน ไปวัด อุณหภูมิของวัตถุ ที่เป้าหมายปลายอีกข้างหนึ่งจะเกิดแรงดันไฟฟ้าขึ้น โดยภาพที่ 26 แสดงให้เห็นถึง การทำงานของ Thermocouple sensor



ภาพที่ 26 แสดงถึงการทำงานของ Thermocouple sensor

T_{obj} คืออุณหภูมิของวัตถุที่ต้องการวัด, T_j (Temperature of cold junction) คืออุณหภูมิของพื้นที่ที่ปลายสายของ sensor ที่ทำการวัดค่า output และ V_{out} คือค่าแรงดันไฟฟ้าที่ Sensor สร้างขึ้นมาเมื่อค่า T_j เท่ากับ $0\text{ }^{\circ}\text{C}$ จะทำให้ค่า V_{out} ที่วัดได้มีค่าตามตารางที่ 1 และ ตารางที่ 2

ตารางที่ 1 Output (mV) ของ Thermocouple type k ที่อุณหภูมิย่านบวกเมื่ออุณหภูมิ Cold junction เป็น $0\text{ }^{\circ}\text{C}$ เซลเซียส

$^{\circ}\text{C}$	0	1	2	3	4	5	6	7	8	9	10
Thermoelectric Voltage in Millivolts											
0	0	0.039	0.079	0.119	0.158	0.198	0.238	0.277	0.317	0.357	0.397
10	0.397	0.437	0.477	0.517	0.557	0.597	0.637	0.677	0.718	0.758	0.798
20	0.798	0.838	0.879	0.919	0.96	1	1.041	1.081	1.122	1.163	1.203
30	1.203	1.244	1.285	1.326	1.366	1.407	1.448	1.489	1.53	1.571	1.612
40	1.612	1.653	1.694	1.735	1.776	1.817	1.858	1.899	1.941	1.982	2.023
50	2.023	2.064	2.106	2.147	2.188	2.23	2.271	2.312	2.354	2.395	2.436

ตารางที่ 2 Output (mV) ของ Thermocouple type k ที่อุณหภูมิย้อนกลับเมื่ออุณหภูมิ Cold junction เป็น 0 องศาเซลเซียส

°C	0	1	2	3	4	5	6	7	8	9	10
Thermoelectric Voltage in Millivolts											
0	0	-0.039	-0.079	-0.118	-0.157	-0.197	-0.236	-0.275	-0.314	-0.353	-0.392
-10	-0.392	-0.431	-0.47	-0.508	-0.547	-0.586	-0.624	-0.663	-0.701	-0.739	-0.778
-20	-0.778	-0.816	-0.854	-0.892	-0.93	-0.968	-1.006	-1.043	-1.081	-1.119	-1.156
-30	-1.156	-1.194	-1.231	-1.268	-1.305	-1.343	-1.38	-1.417	-1.453	-1.49	-1.527
-40	-1.527	-1.564	-1.6	-1.637	-1.673	-1.709	-1.745	-1.782	-1.818	-1.854	-1.889

แต่เมื่ออุณหภูมิของ T_j ไม่เท่ากับ 0 C จะทำให้ค่า V_{out} ที่วัดได้ต่างจากตารางดังนั้นใน การใช้งาน Thermocouple Sensor ต้องมีการชดเชยค่าที่เปลี่ยนแปลงไปตามอุณหภูมิของ T_j ด้วยจึงจะสามารถใช้งานได้เนื่องจากการทำงานจรไฟฟ้าที่ใช้งานกับกับ Thermocouple Sensor นั้นมีความยุ่งยากในการ Calibrate และอีกทั้งการหาอุปกรณ์ที่มีค่า Resistive และค่า capacitive ของตามที่ต้องการนั้นทำได้ยาก ดังนั้นในงานวิจัยนี้จึงได้ทำการวัดค่า อุณหภูมิจากตัว Thermocouple Sensor ด้วยการทำการชดเชยได้การคำนวณภายใน โปรแกรมโดยมีวิธีการคำนวณดังนี้โดยขั้นแรกทำการคำนวณหาค่า Emf_c ที่ใช้ในการ ชดเชยจากสมการที่ 23

$$Emf_c = \sum_{i=0}^9 C_i T_j^i + a_0 \exp(a_1 (T_j - a_2)^2) \quad (23)$$

โดยที่ T_j คืออุณหภูมิของ Cold Junction

$$C_0 = -0.176004136860E - 01$$

$$C_1 = 0.389212049750E - 01$$

$$C_2 = 0.185587700320E - 04$$

$$C_3 = -0.994575928740E - 07$$

$$C_4 = 0.318409457190E - 09$$

$$C_5 = -0.560728448890E - 12$$

$$C_6 = 0.560750590590E - 15$$

$$C_7 = -0.320207200030E - 18$$

$$C_8 = 0.971511471520E - 22$$

$$C_9 = -0.121047212750E - 25$$

$$a_0 = 0.118597600000E + 00$$

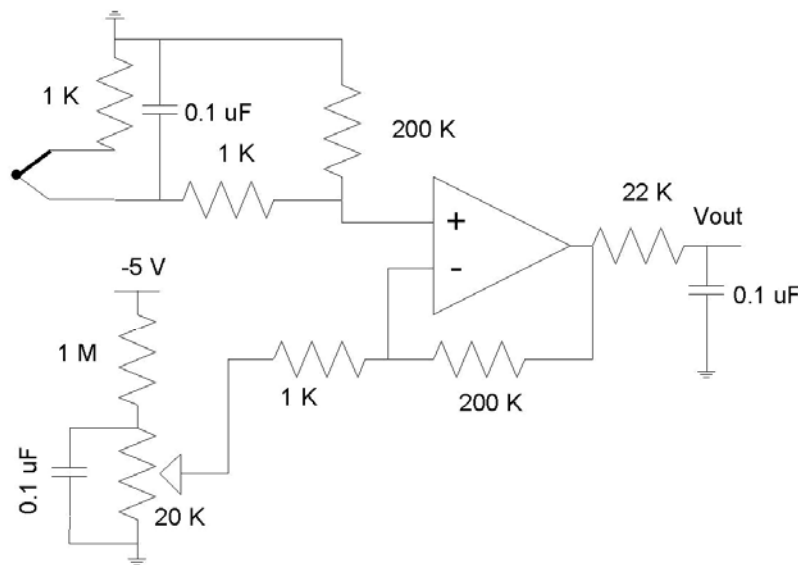
$$a_1 = -0.118343200000E - 03$$

$$a_2 = 0.126968600000E + 03$$

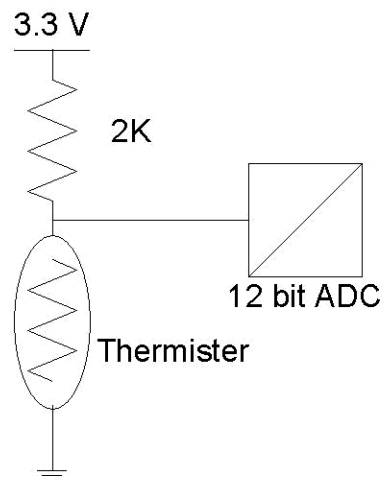
หลังจากนั้นอ่านค่า V_{out} และนำมารวมกับค่าชดเชยดังสมการที่ 24

$$V_{compensate} = V_{out} + Emf_c \quad (24)$$

จากนั้นนำค่า $V_{compensate}$ ที่ได้ไปเปรียบเทียบกับตารางว่าเป็นค่าอุณหภูมิเท่าไรโดยวงจร Opamp ที่เป็นภาคขยายสัญญาณมีการออกแบบดังภาพที่ 27 โดยวงจรขยายมีค่า Gain เท่ากับ 200 เท่า ดังนั้นเมื่อมีสัญญาณมา 40 μV จะถูกขยายมาเป็น 8 mV และวงจรอ่าน ค่าอุณหภูมิที่ Cold junction (T_j) แสดงดังภาพที่ 28



ภาพที่ 27 แสดงวงจรภาครับสัญญาณ Thermocouple ที่มี Gain ขยาย 200 เท่า

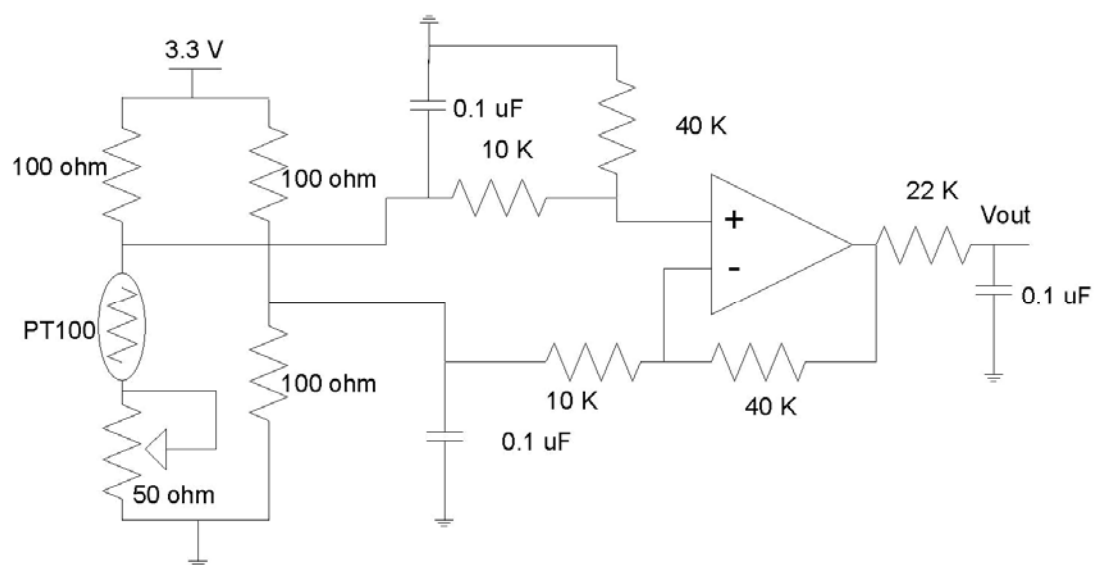


ภาพที่ 28 แสดงวงจรที่ใช้สำหรับอ่านค่า Cold junction (T_j)

6.2 PT100 sensor เป็น Sensor ที่ใช้สำหรับวัดอุณหภูมิโดยที่ค่า output ของ Sensor จะเป็นค่า ความต้านทาน โดยเมื่ออุณหภูมิสูงขึ้นค่าความต้านทานจะมากขึ้นดังที่แสดงในตารางที่ 3 และวงจรที่ใช้ในการอ่านค่าแสดงในภาพที่ 29

ตารางที่ 3 Output (ohm) ของ PT100

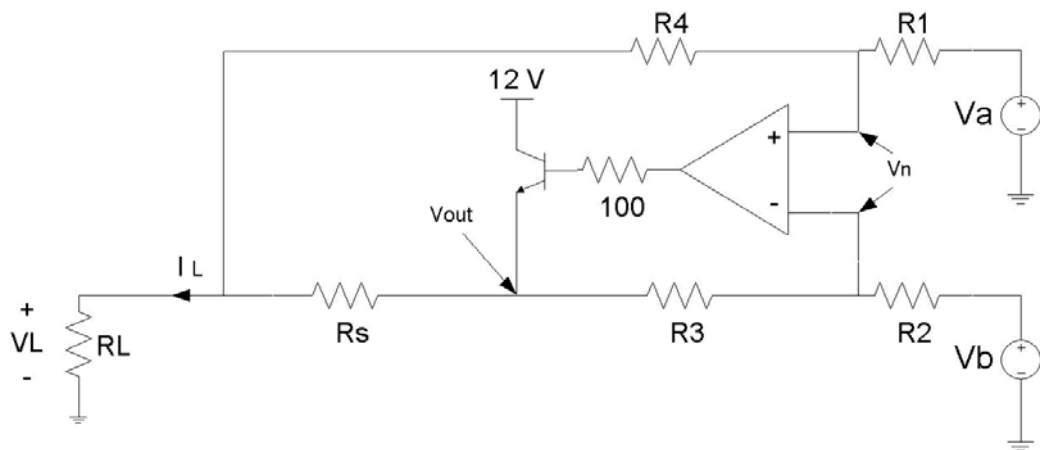
°C	0	1	2	3	4	5	6	7	8	9
Resistance in Ohm										
20.0	107.8	108.2	108.6	109.0	109.4	109.7	110.1	110.5	110.9	111.3
30.0	111.7	112.1	112.5	112.8	113.2	113.6	114.0	114.4	114.8	115.2
40.0	115.5	115.9	116.3	116.7	117.1	117.5	117.9	118.2	118.6	119.0
50.0	119.4	119.8	120.2	120.6	120.9	121.3	121.7	122.1	122.5	122.9



ภาพที่ 29 แสดงให้เห็นถึงวงจรที่ใช้สำหรับการอ่านค่า Output ของ PT100

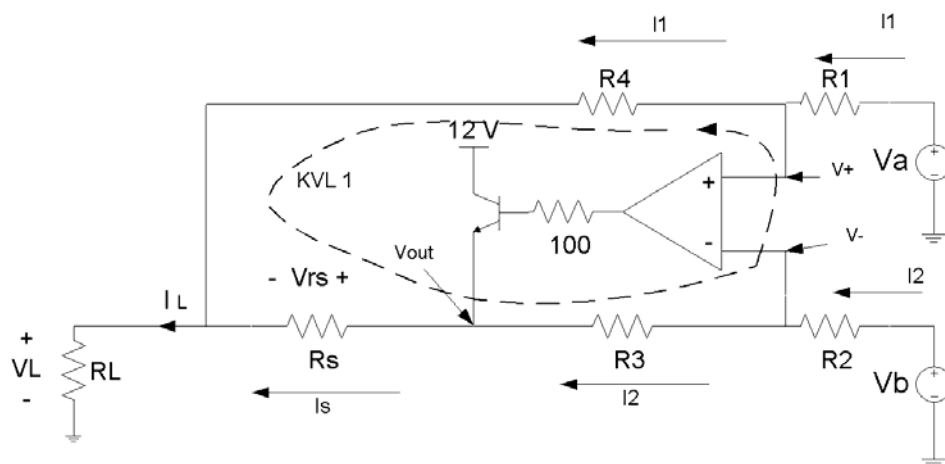
7. การออกแบบวงจรขับสัญญาณ Analog 4 ถึง 20 mA

การขับสัญญาณกระแส 4 ถึง 20 mA จะขับผ่านวงจร Volt to Current Source (VCS) ดังภาพที่ 30



ภาพที่ 30 แสดงรายละเอียดของวงจร VCS

จากรูปเราสามารถหาความสัมพันธ์ของ V_a , V_b และ I_{out} ได้โดยพิจารณา ดังนี้ ให้ $R_1=R_2=R$, $R_3=R_4+R_s$ และหาค่ากระแสของ node ดังภาพที่ 31



ภาพที่ 31 แสดงการหาความสัมพันธ์ของ V_a , V_b และ I_L

จากการพิจารณาหากระแส V^+ และ V^- ได้ดังนี้

$$V^- = V_b - R_2 I_2$$

$$V^+ = V_a - R_1 I_1$$

จาก $V^+ = V^-$

$$V_b - R_2 I_2 = V_a - R_1 I_1$$

ให้ $R_1 = R_2 = R_1$

$$I_1 - I_2 = \frac{V_a - V_b}{R_1} \quad (25)$$

จากนั้นทำการคิด KVL1 จะได้

$$0 = R_4 I_1 - R_s I_s - R_3 I_2 + V^+ - V^-$$

ให้ $R_3 = R_4 + R_s$ และ $R_4 = R_3 - R_s$

$$\begin{aligned} I_s &= \frac{I_1 R_4}{R_s} - \frac{I_2 R_3}{R_s} \\ &= \frac{I_2 R_3}{R_s} - I_1 - \frac{I_2 R_3}{R_s} \end{aligned} \quad (26)$$

จาก $I_L = I_1 + I_s$

$$\begin{aligned} I_L &= \frac{I_1 R_3}{R_s} - \frac{I_2 R_3}{R_s} \\ I_L &= \frac{R_3}{R_s} (I_1 - I_2) \end{aligned} \quad (27)$$

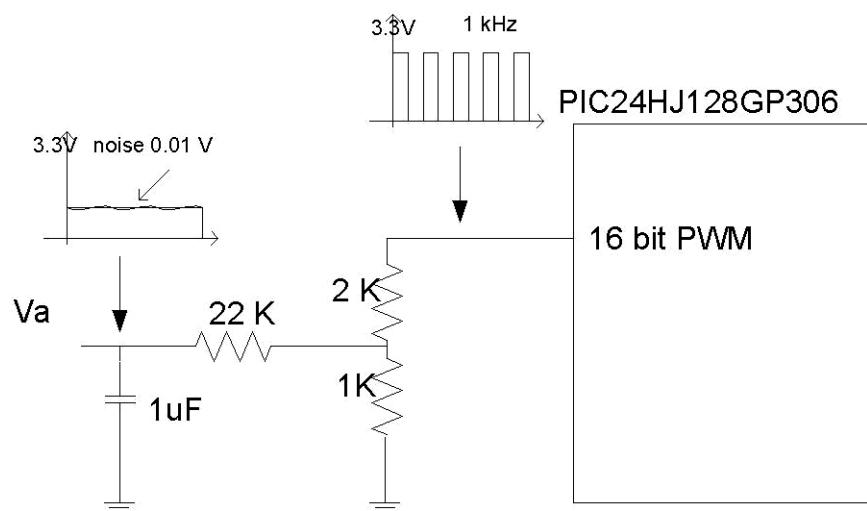
แทนสมการที่ 25 ลงในสมการ 27 ได้

$$I_L = \frac{R_3}{R_s} \left(\frac{V_a - V_b}{R_1} \right)$$

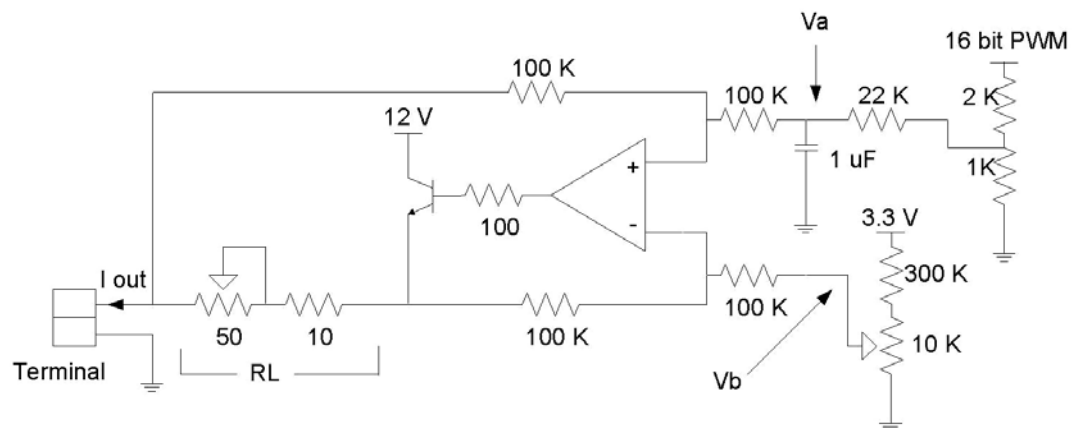
โดยเลือกให้ $R_1 = R_2 = R_3$

$$I_L = \frac{V_a - V_b}{R_s} \quad (28)$$

จากวงจร VCS เราจะใช้การจ่ายแรงดัน V_a เป็นแรงดัน input สำหรับควบคุมกระแส โดยที่แรงดันนี้จะได้มาจาก MCU ดังภาพที่ 32 VCS จะรับสัญญาณเป็น PWM 16 bit ที่ควบคุมด้วย MCU โดยสามารถปรับ DutyCyc ได้ 0 ถึง 100 % ผ่านวงจร lowpass filter เมื่อ PWM ผ่าน lowpass filter จะได้สัญญาณ เป็น V_a (Volt DC)



ภาพที่ 32 แสดงรายละเอียดของสัญญาณ PWM เมื่อผ่าน low pass filter

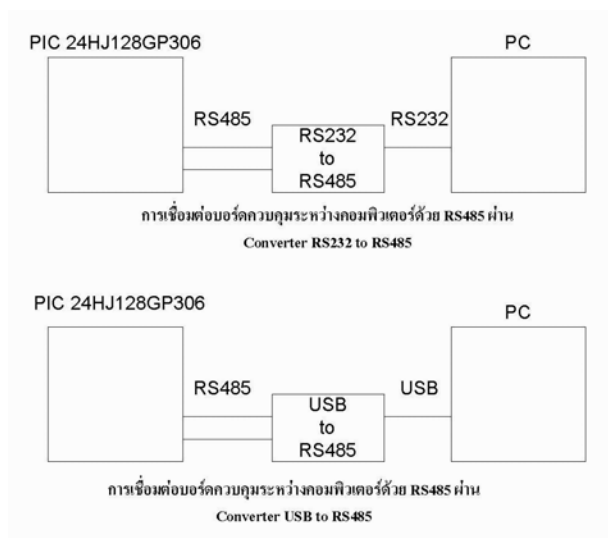


ภาพที่ 33 แสดงรายละเอียดของวงจรที่ใช้งาน

เมื่อ MCU ขับ PWM Duty Cyc 100 % จะทำให้วงจรกับกระแสออกมา 20 mA และเมื่อ MCU ขับ PWM Duty Cyc 0 % จะทำให้วงจรขับสัญญาณออกมา 4 mA

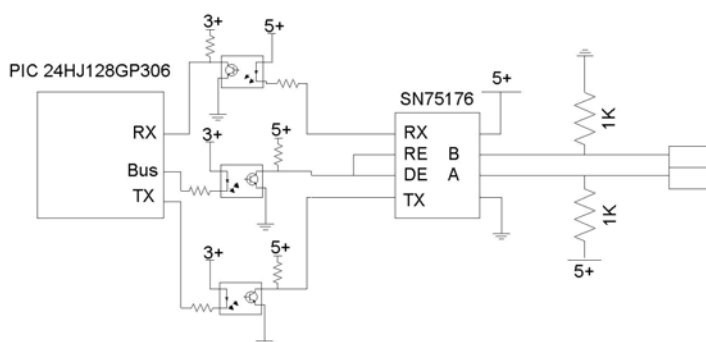
8. การออกแบบวงจรสื่อสารกับคอมพิวเตอร์

การสื่อสารระหว่างบอร์ดควบคุมและ PC จะทำการสื่อสารผ่านระบบสัญญาณ RS485 และ อุปกรณ์ Converter USB to RS485 หรือ Converter RS232 to RS485 ดังแสดงในภาพที่ 34



ภาพที่ 34 แสดงการเชื่อมต่อของบอร์ดควบคุมด้วยอุปกรณ์ Converter ต่างๆ

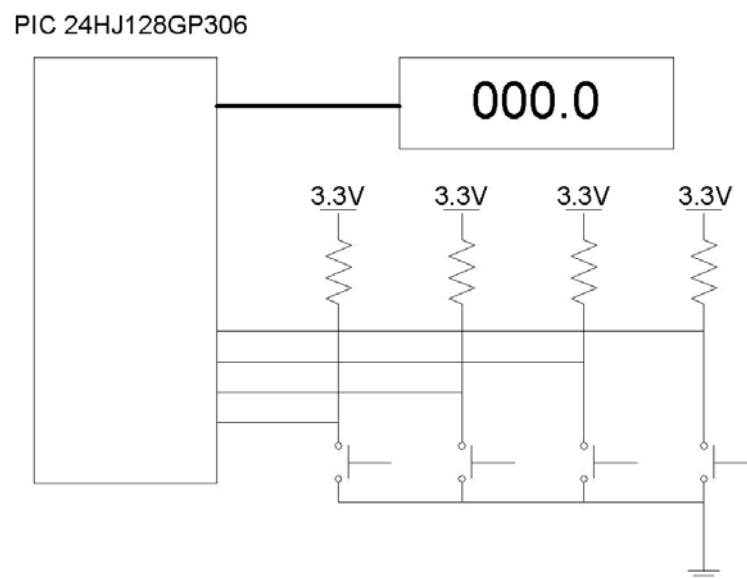
การออกแบบวงจรสื่อสาร RS485 จะใช้ IC SN75176 โดยการใช้งานจะต้องมีการส่งสัญญาณสลับการทำงานระหว่างรับสัญญาณและส่งสัญญาณให้กับ IC SN75176 และ Module สื่อสารที่ MCU ใช้งานคือการ UART Module ของ MCU เอง และ Protocol ที่ใช้ในการสื่อสารคือ Modbus RTU Protocol



ภาพที่ 35 แสดงรายละเอียดของวงจรสื่อสาร

9. ออกแบบการแสดงผลการทำงานด้วย 7 Segment Display และ Key Pad interface

โดยวงจรขับ 7 Segment จะใช้ IC 74HC595 เป็น Bus ที่ใช้สำหรับ Drive และใช้เทคนิคการ Drive แบบ Scan ที่ละ Segment

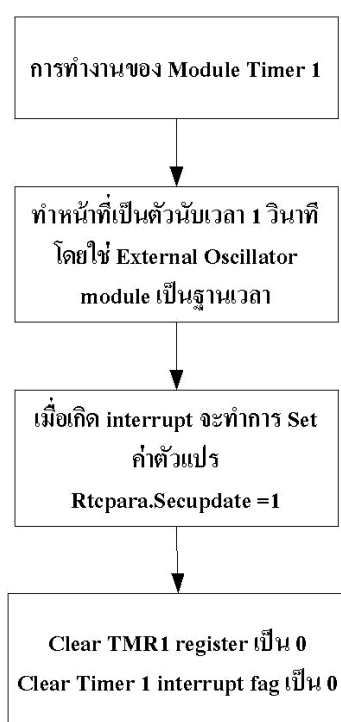


ภาพที่ 36 แสดงรายละเอียดการของวงจรการแสดงผลและ Key Pad Interface

10. รายละเอียดการออกแบบโปรแกรมของบอร์ดควบคุม

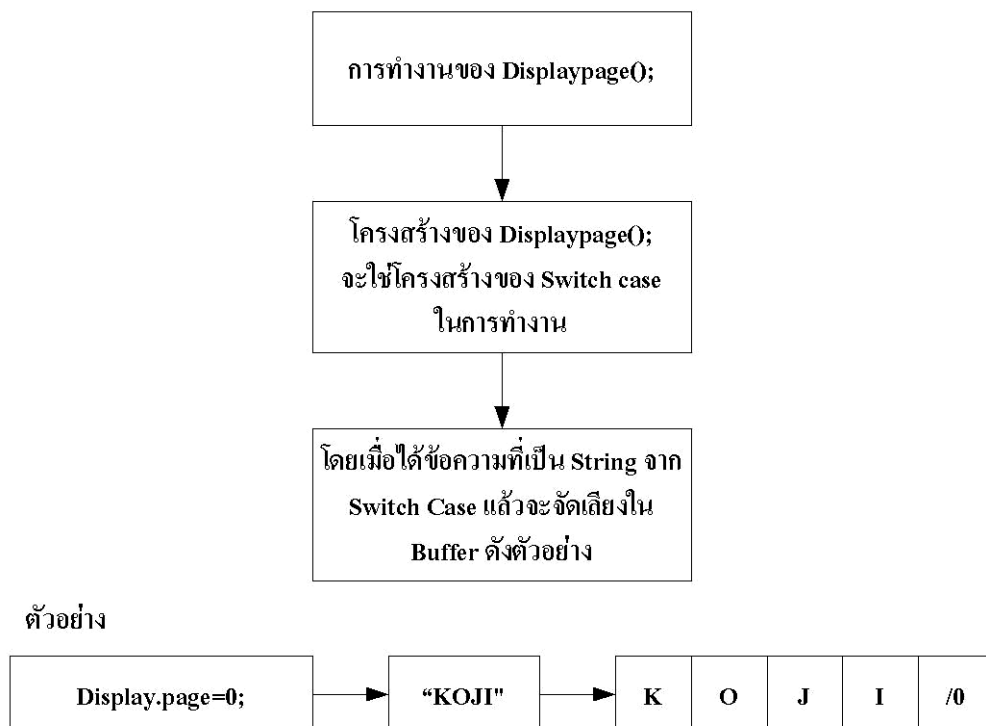
โดยงานของ MCU จะถูกกระจายงานไปเป็นส่วนย่อยและทำงานใน module ที่แตกต่างกัน เพื่อให้ MCU มีประสิทธิภาพที่สุดในการทำงาน โดยที่งานของ MCU จะถูกแบ่งออกเป็น

10.1 นับฐานเวลา 1 วินาที โดยจะใช้งาน Interrupt Service ของ 16bit Timer 1 Module เป็นฐานเวลา โดยที่การเกิด interrupt ของ Module Timer 1 จะใช้ค่าความถี่จาก External Low Power 32.768 kHz Crystal Osillator เป็นฐานเวลา เนื่องจากค่า 32,768 มีค่าเป็นครึ่งหนึ่งของ TMR1 register ดังนั้นจึงสามารถตั้งค่าให้เกิด interrupt ทุกๆ 1 วินาทีได้อย่างถูกต้อง และการทำงานโปรแกรมใช้ Timer1 Module นี้จำเป็นต้องใช้เวลาให้สั้นที่สุดเพื่อลบกวณ ฐานเวลาให้น้อยที่สุดจึงใช้ให้ค่าตัวแปร `Ratpara.secupdate = 1` เท่านั้น โดยการทำงานแสดงให้เห็นดังภาพที่ 37



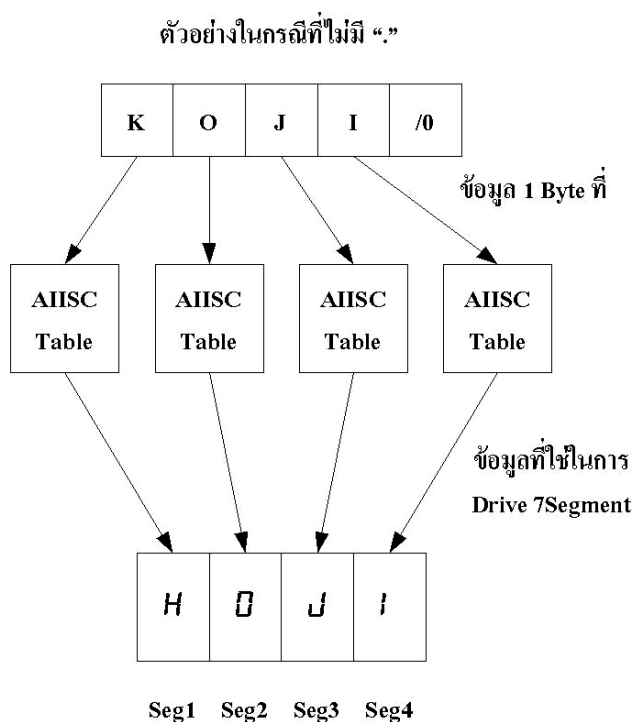
ภาพที่ 37 แสดงแผนภาพการทำงานใน 16 bit Timer1 Module

10.2 การแสดงผลผ่าน 7 Segment การทำงานในส่วนนี้จะถูกทำงานโดยใช้ Interrupt Service ของ 16 bit Timer 2 Module โดยในการแสดงผลจะมีการทำงานสองส่วนคือการจัดข้อมูลที่จะแสดงผลและการแสดง ผลด้วยการจับข้อมูลผ่าน 74HC595 โดยการจัดการข้อมูลที่จะใช้ในการแสดงผลจะทำใน function Displaypage(); ซึ่งจะทำหน้าที่เขียนข้อมูลลงใน Display Buffer ขนาด 6 Byte การทำงานของ function Displaypage(); แสดงให้เห็นในภาพที่ 38

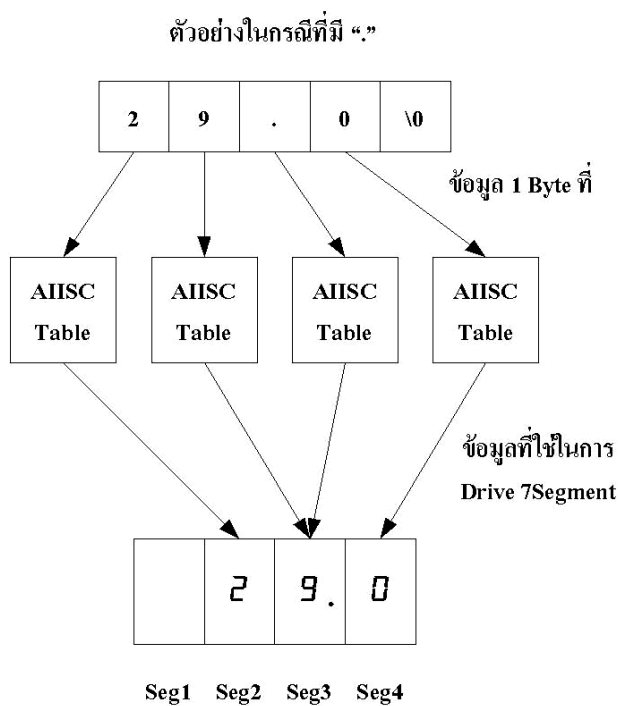


ภาพที่ 38 แสดงการทำงานของ Displaypage function

การแสดงผลด้วยการจับข้อมูลผ่าน 74HC595 จะกระทำใน function WriteStr2Seg(); โดยที่ใน function WriteStr2Seg จะทำการอ่านข้อมูลจาก Display buffer ที่ได้มีการเขียนไว้ใน function displaypage(); โดยแสดงตัวอย่างการทำงานให้เห็น ดังภาพที่ 39 และ ภาพที่ 40

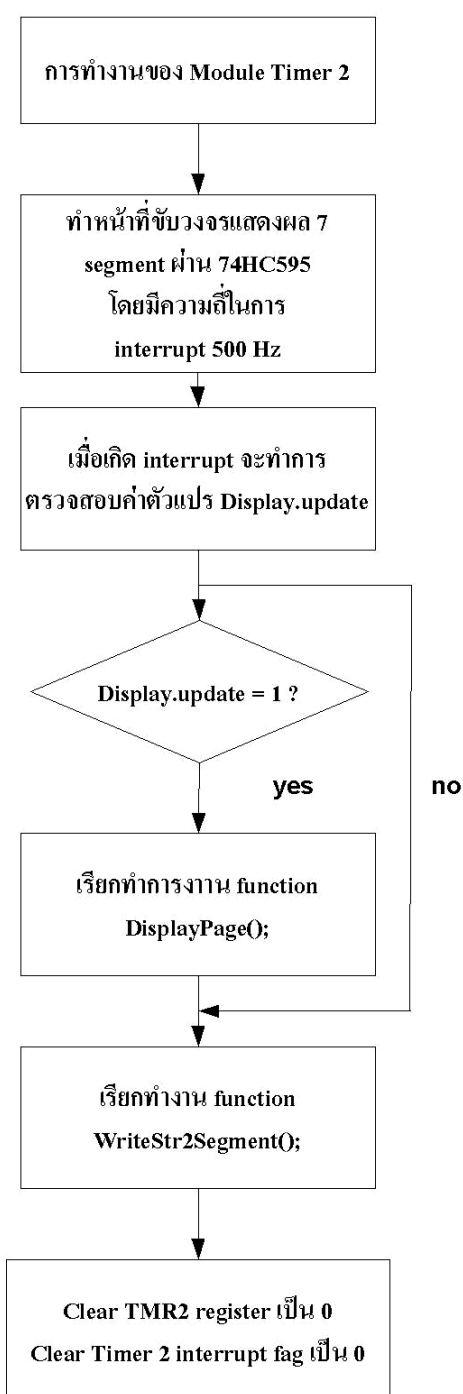


ภาพที่ 39 แสดงการตัวอย่างการทำงานของ function WriteStr2Seg ในกรณีไม่มี Character “.”



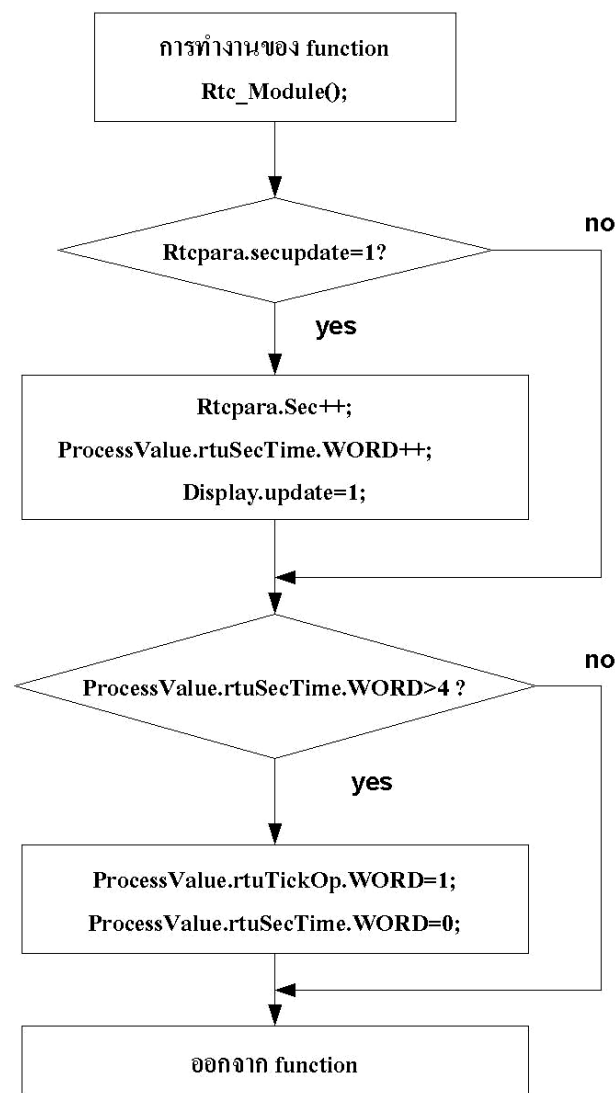
ภาพที่ 40 แสดงการตัวอย่างการทำงานของ function WriteStr2Seg ในกรณีมี Character “.”

การเกิด Interrupt ของ 16 bit Timer 2 Module จะมีความถี่อยู่ที่ 500 Hz เพื่อให้การแสดงผลผ่าน 7 Segment ด้วยเทคนิคการ Scan ค่าที่ละ Segment ได้ผลดี โดยภาพรวมของการทำงานใน Interrupt Service ของ Timer 2 แสดงให้เห็นดังภาพที่ 41



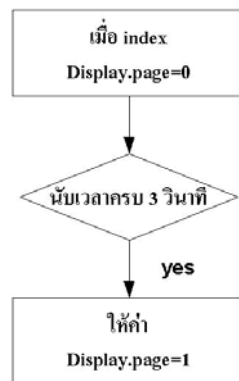
ภาพที่ 41 แสดงแผนผังการทำงานใน interrupt service ของ Timer 2 Module

10.3 การนับเวลาสำหรับการทำงานของโปรแกรมอื่นและ การจัดการกับข้อมูลสื่อสาร และรับค่า key Pad การทำงานใช้ Interrupt Service ของ 16 bit Timer 3 Module โดยงานที่ทำใน Interrupt Service จะประกอบด้วย การนับเวลา, การอ่าน Key Pad และการจัดการสื่อสาร โดยที่การนับเวลาจะกระทำใน function Rtc_Module(); การทำงานใน function นี้จะเป็นการตรวจสอบตัวแปร Rtcpara.secupdate ว่ามีค่าเป็น 1 หรือไม่โดยเมื่อมีค่าเป็น 1 ให้ทำการเพิ่มค่าตัวแปรต่างดังแสดงในภาพที่ 42

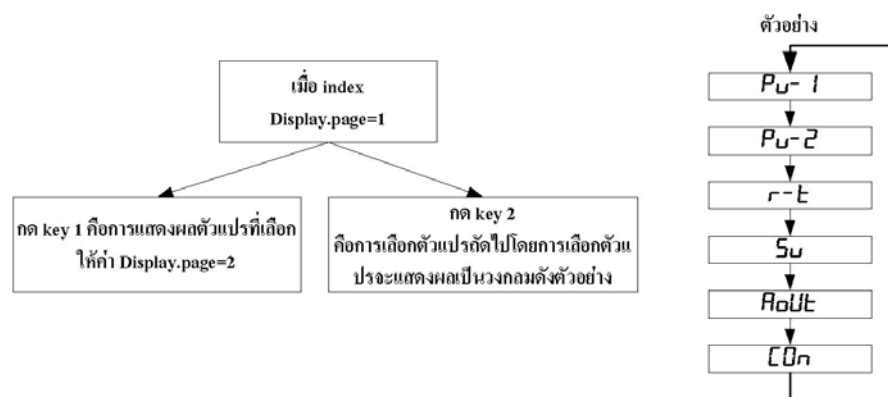


ภาพที่ 42 แสดงการทำงานของ function Rtc_Module();

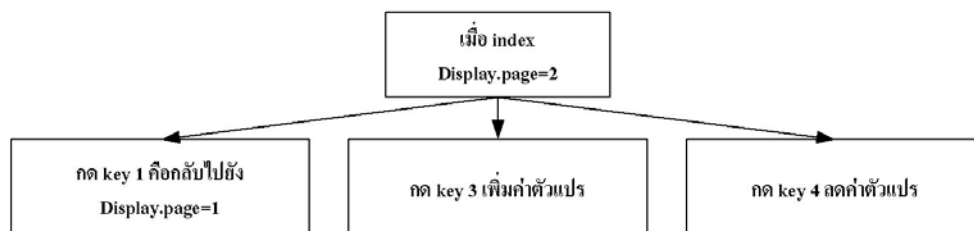
การอ่านค่า key pad กระทำใน function Funkey(); โดยใช้ switch case เป็นโครงสร้างในการทำงาน และใช้ Display.page เป็นตัวแปร index สำหรับ switch case โดยการทำงานแสดงให้เห็นในภาพที่ 43, 44 และ 45



ภาพที่ 43 แสดงการทำงานของ function Funkey(); เมื่อ Display.page=0

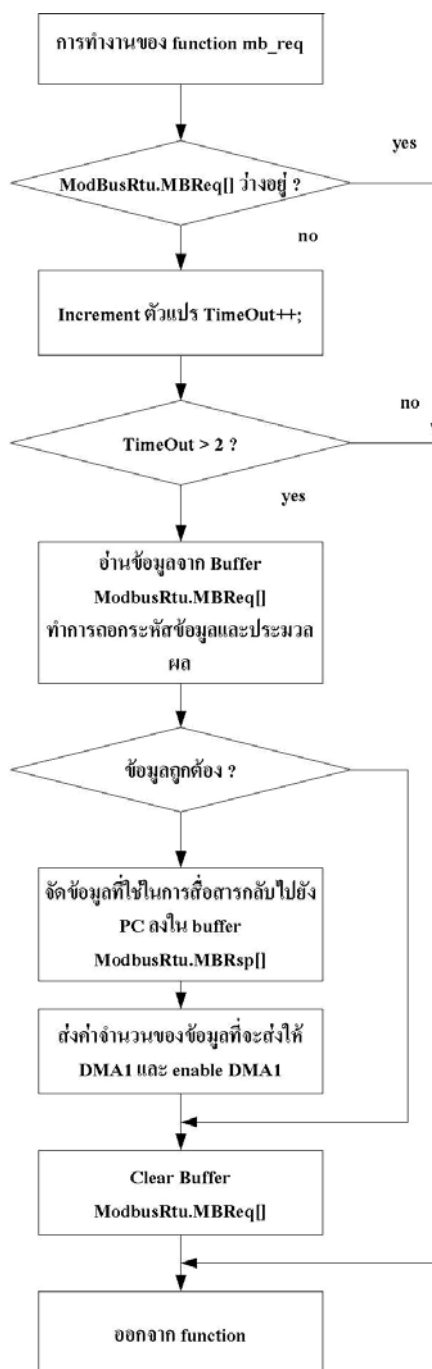


ภาพที่ 44 แสดงการทำงานของ function Funkey(); เมื่อ Display.page=1



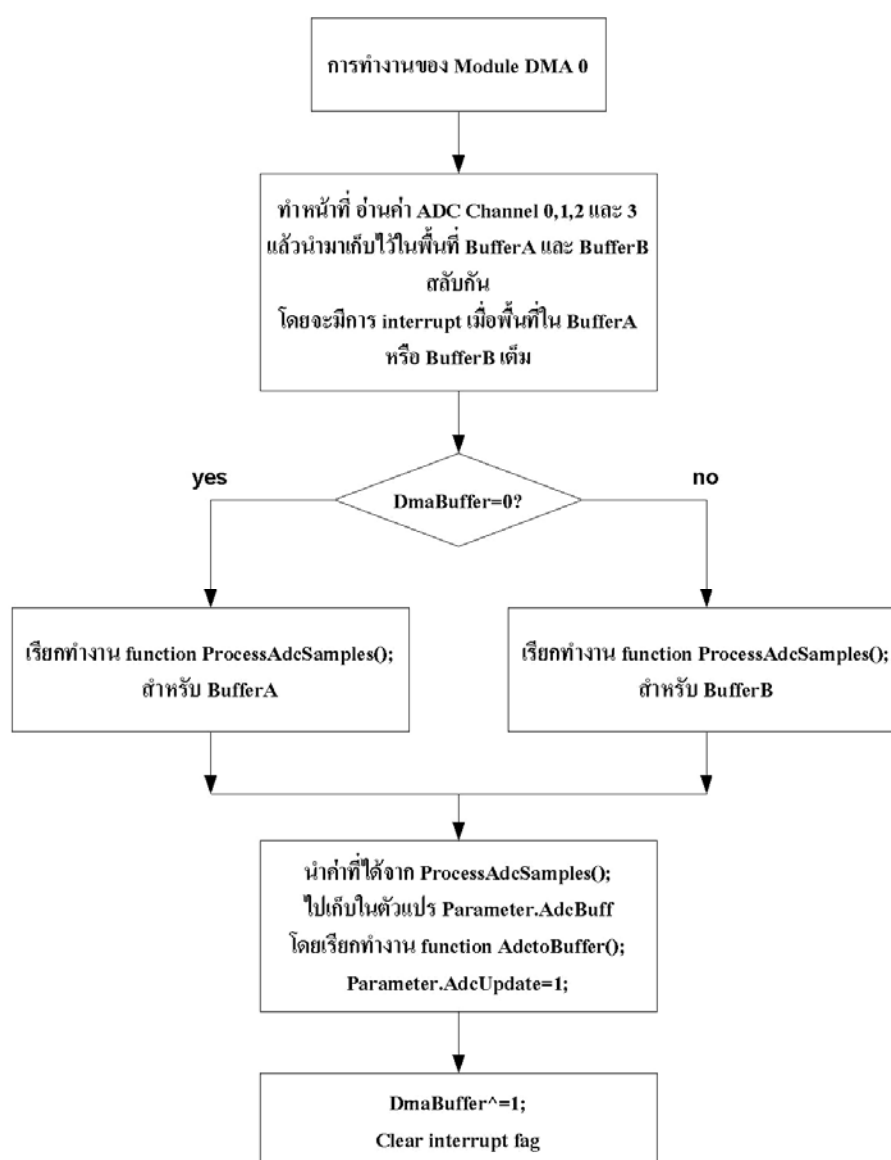
ภาพที่ 45 แสดงการทำงานของ function Funkey(); เมื่อ Display.page=2

การทำงานของจัดการข้อมูลที่ใช้ในการสื่อสารจะกระทำใน Function mb_req(); โดยการทำการสื่อสารจะอาศัยโปรโตคอล Modbus RTU ในการติดต่อสื่อสารภาพที่ 46 แสดงให้เห็นถึงการทำงานของ mb_req();



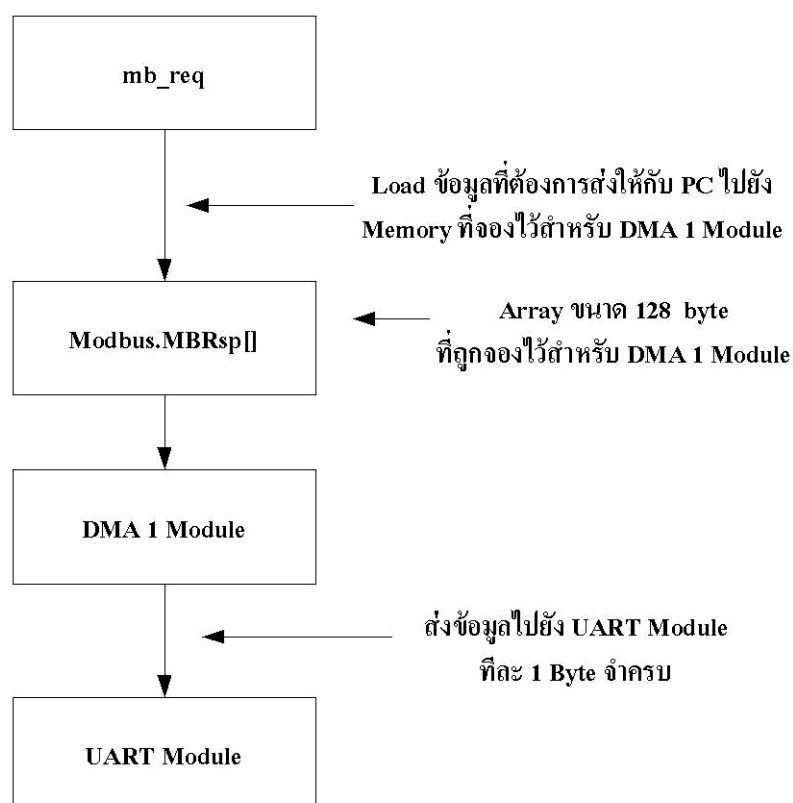
ภาพที่ 46 แสดงให้เห็นถึงการทำงานของ mb_req()

10.4 การอ่านค่า ADC จำนวน 4 channel จะกระทำโดยใช้งาน Direct Memory Access (DMA) Module ช่วยในการทำงาน โดยหลักการทำงานของ DMA Module คือทำหน้าที่อ่านค่า ADC ของแต่ละ Channel แล้วนำไปเก็บที่ buffer A และ buffer B สลับกันไปหลังจากที่มี DMA module อ่านค่า ADC และนำไปเก็บจำ buffer A หรือ buffer B เต็ม จะเกิดการ interrupt ของ DMA module ดังแสดงให้เห็นในภาพที่ 47



ภาพที่ 47 แสดงการทำงานของ DMA 0 Module

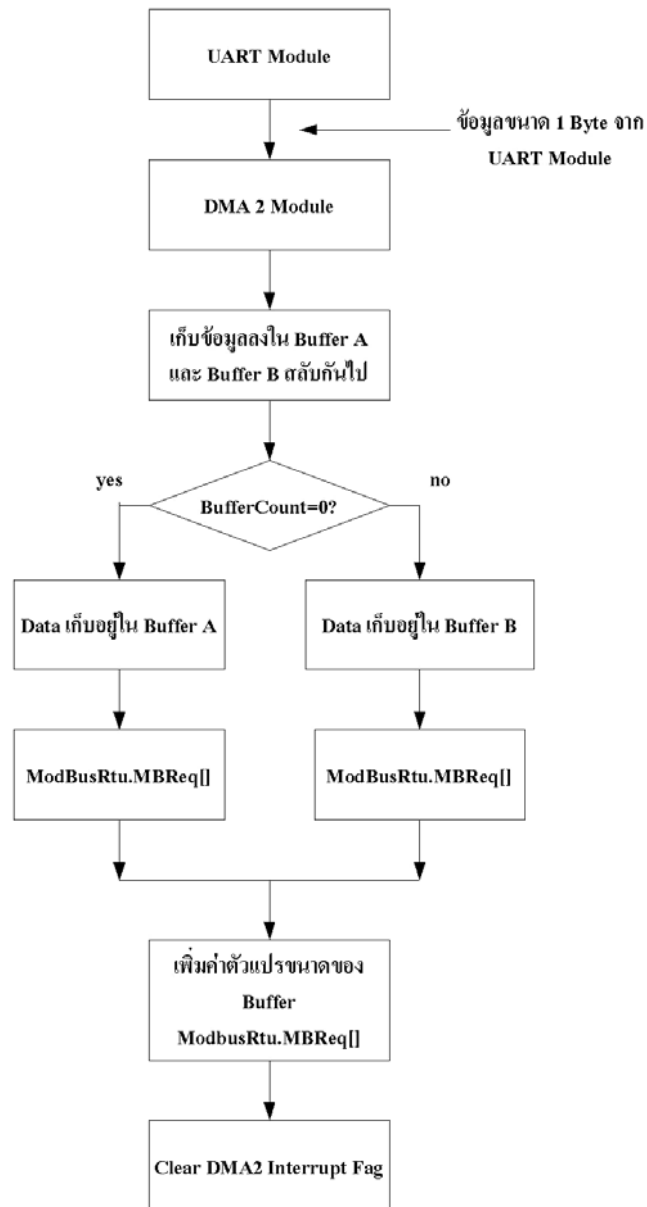
10.5 การส่งค่าข้อมูลสื่อสารสัญญาณสื่อสารที่ใช้ในการติดต่อกับ PC นั้นจะเป็นสัญญาณ Serial โดยที่การส่งสัญญาณ ดังกล่าวนี MCU จะอาศัยการทำงานของ UART Module ทำหน้าที่ส่งสัญญาณ และใช้ DMA 1 Module ใช้การจัดเรียงข้อมูลให้กับ UART Module โดยกระบวนการทำงานนี้แสดงให้เห็นดังภาพที่ 48



ภาพที่ 48 แสดงให้กระบวนการทำงานของ UART และ DAM1 Module

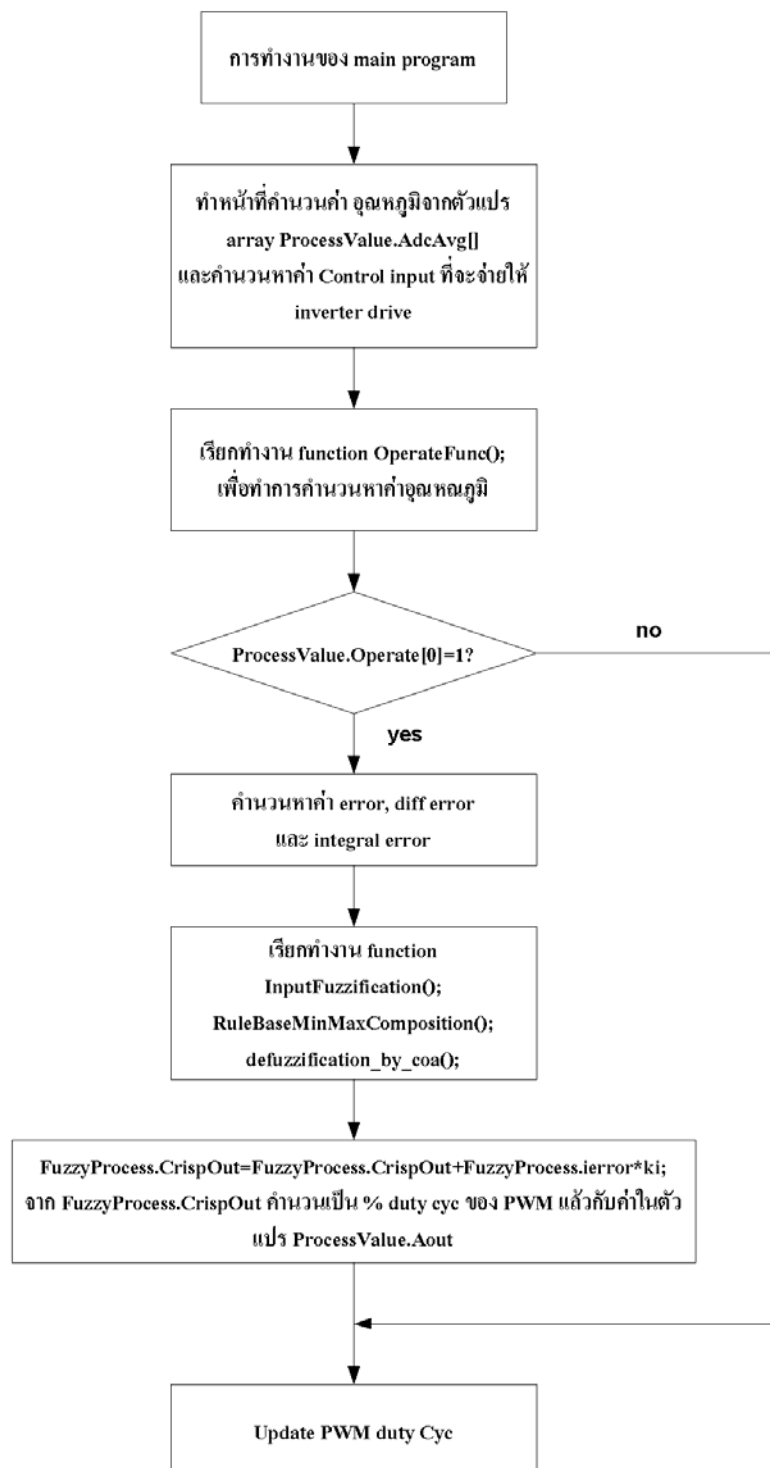
หลังจากที่ DMA1 Module ส่งข้อมูลที่อยู่ภายใน buffer จนหมดแล้วจะเกิดการ interrupt ขึ้นเพื่อเป็นการแจ้งว่ากระบวนการเสร็จสิ้น

10.6 การรับค่าข้อมูลสื่อสารสัญญาณสื่อสารที่ PC ส่งมายัง MCU มีรูปแบบเป็นสัญญาณ Serial เช่นกัน ดังนั้น MCU การรับค่าสัญญาณจะกระทำผ่าน UART Module เช่นกัน โดยกระบวนการรับสัญญาณ จะอาศัยการทำงานของ DMA 2 Module ร่วมกับ UART Module โดยที่กระบวนการทำงานจะเป็นดังภาพที่ 49



ภาพที่ 49 แสดงกระบวนการทำงานใน Interrupt service ของ DMA 2 Module

10.7 การคำนวณค่าอนุภูมิและการคำนวณหาค่าควบคุม



ภาพที่ 50 แสดงการทำงานของโปรแกรมใช้ Main Program ของ MCU

การคำนวณค่าอุณหภูมิจาก `Parameter.AdcBuffer[]` และการคำนวณหาค่าควบคุมจะกระทำใน Main Program ของ MCU เนื่องจากการเป็นการทำงานที่ใช้ Cycle ของ MCU เป็นจำนวนมาก แต่มีคาบในการคำนวณแต่ละครั้งยาวนานคือ มีการคำนวณค่าอุณหภูมิใหม่ทุกๆ 1 วินาที และมีการคำนวณค่าควบคุมใหม่ทุกๆ 5 วินาที ภาพที่ 50 แสดงการทำงานของโปรแกรมใช้ Main Program ของ MCU ภายใน Function `OperateFunc()`; จะทำการอ่านค่า Adc จากตัวแปร `Parameter.Adcbuff[]` ของ Adc แต่ละช่องมาทำการคำนวณหาค่าอุณหภูมิ

11. การออกแบบโปรแกรมควบคุม Fuzzy PID

โดยการเขียนโปรแกรมจะทำแบ่งออกตามส่วนประกอบของ Fuzzy โดยมี 3 ส่วนคือ Fuzzifier, Fuzzy Reasoning Unit และ Defuzzifier

11.1 ออกแบบโปรแกรมส่วน Fuzzifier โดยที่ Fuzzifier ประกอบด้วยโครงสร้าง 2 ส่วน คือ โครงสร้างส่วนที่ใช้เก็บค่าความเป็นสมาชิกของแต่ละ Membership function สำหรับ Crisp input หนึ่งๆ และโครงสร้างในส่วนที่ใช้กำหนด Membership function โดยภาพที่ 51 แสดงโครงสร้างข้อมูลของ Input Fuzzy โดย array ที่ประกาศมีขนาด 5 แทนจำนวน Membership function 5 ของแต่ละ input fuzzy

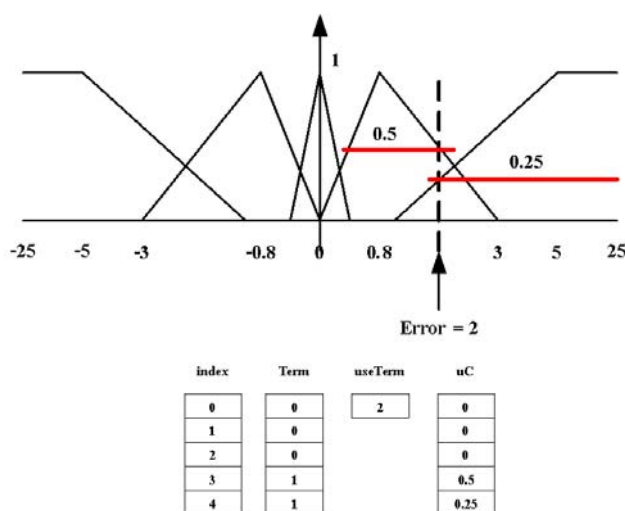
```
typedef struct {
    unsigned char Term[5];
    unsigned char useTerm;
    float uC[5];
} FuzzyTerm;
```

← ใช้สำหรับตรวจว่าค่า Crisp input อยู่ในช่วง Member ship function นั้นหรือไม่

← บอกจำนวน Member ship function ที่ถูกเลือก

← ค่าความเป็นสมาชิกของ Member ship function สำหรับ Crisp input ที่นำมาคำนวณ

ภาพที่ 51 แสดงโครงสร้างข้อมูลของ Input Fuzzy



ภาพที่ 52 แสดงตัวอย่างค่าข้อมูลที่เก็บในตัวแปร โดยมี Input error=2

ภาพที่ 53 แสดงตัวอย่างโครงสร้างข้อมูลที่ใช้แทน Membership function โดยตัวอย่างเป็นของ input fuzzy error ประกอบไปด้วยข้อมูล 2 ชุดคือ MFErrT[5][5] เป็นข้อมูลสำหรับ define Membership function มีขนาดเป็น Array 5x5

```
const float MFErrT[5][5]={
// First index    0      1      2      3      4
//                largeNeg minNeg Zero   minPos LargePos
    {-25,-25,-5,-1,1},{-3,-0.8,0,0,0},{-0.2,0,0.2,0,0},{0,1,3,0,0},{1,5,25,25,1}
};
// Data Format {a,b,c,d,type}  type = 0 --> triangle  type = 1 --> trapezoidal
const float MFErrorTerm[5][2]={
// First index    0      1      2      3      4
//                largeNeg minNeg Zero   minPos LargePos
    {-2,-25},{0,-3},{0.2,-0.2},{3,0},{25,2}
};
//// Data Format {max,min}
```

ภาพที่ 53 แสดงตัวอย่างโครงสร้างข้อมูลที่ใช้แทน Membership function

โดยที่ index แรกของ array แทน largeNeg, minNeg, Zero, minPos และ LargePos ส่วนใน index ชุดที่สองแทน Format สำหรับ คำนวณค่าความเป็นสมาชิกใน Function คำนวณ MF_triangle(x,a,b,c) และ MF_Trapezoidal(x,a,b,c,d) ส่วน MFErrorTerm[5][2] โดยที่ index แรกของ array แทน largeNeg, minNeg, Zero, minPos และ LargePos ส่วนใน index ชุดที่สองแทน ค่าขอบเขตสูงสุดและต่ำสุดของแต่ละ Membership function โดยข้อมูลชุดนี้ใช้ในการตรวจสอบว่า ค่า Crisp input ที่รับมาอยู่ใน Membership function ไດบ้าง

```

float MF_triangle(float x,float a,float b,float c)
{
//
//          = 0                      x<=a
// triangle(x,a,b,c) = (x-a)/(b-a)    a<=x<=b
//          = (c-x)/(c-b)            b<=x<=c
//          = 0                      c<=x
/*
//      /\
//     /  \
//    /    \ a<b<c
//   /      \
//  a  b  c
*/
if((x<=a)|| (x>=c))
{
return(0);
}
else if((x>a)&&(x<=b))
{
return((x-a)/(b-a));
}
else if((x>b)&&(x<c))
{
return((c-x)/(c-b));
}
return(0);
}

```

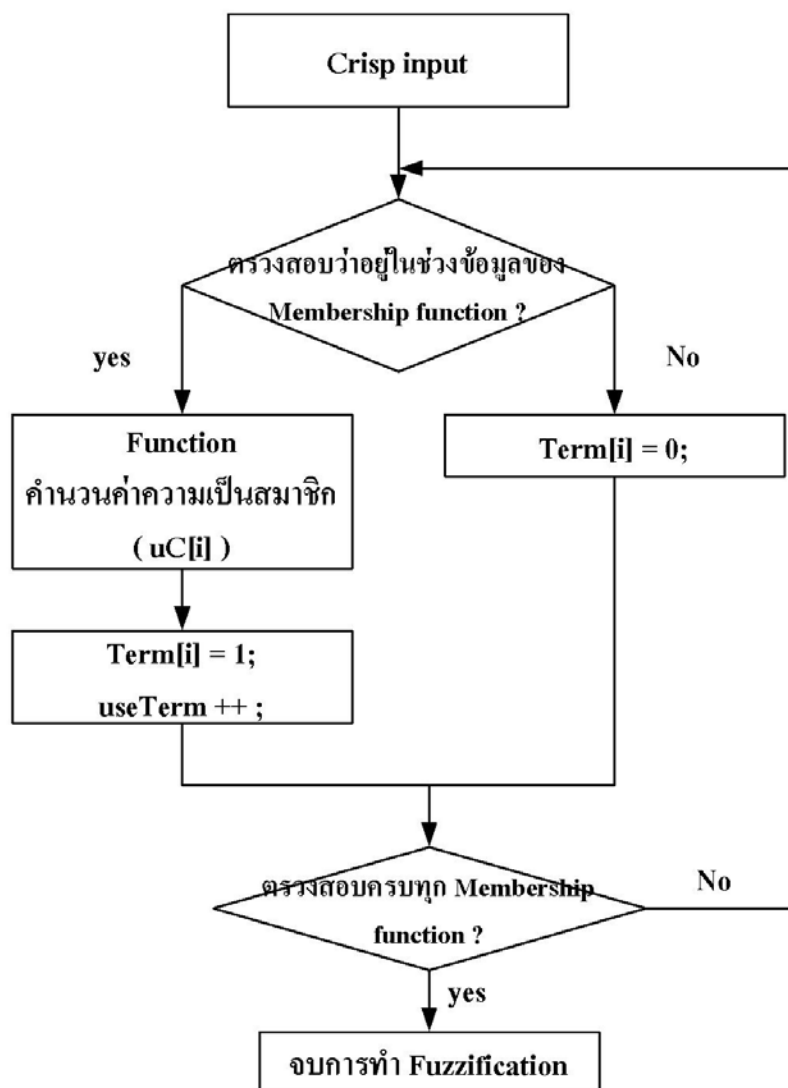
ภาพที่ 54 Function คำนวณค่าความเป็นสมาชิกของ Membership function แบบ triangle

```

float MF_Trapezoidal(float x,float a,float b,float c,float d)
{
//
//          =0                      x<=a
//          =(x-a)/(b-a)            a<=x<=b
// trapezoid(x,a,b,c,d) =1          b<=x<=c
//          =(d-x)/(d-c)            c<=x<=d
//          =0                      d<=x
/*
//      ----
//     /      \
//    /        \
//   /          \
//  /            \
// /              \
// a  b          c  d
*/
if( (x>=b)&&(x<=c) )
{
return(1);
}
else if( (x>=a)&&(x<=b) )
{
return( (x-a)/(b-a) );
}
else if( (x>c)&&(x<=d) )
{
return( (d-x)/(d-c) );
}
else if((x<=a)|| (x>=d))
{
return(0);
}
return(0);
}

```

ภาพที่ 55 Function คำนวณค่าความเป็นสมาชิกของ Membership function แบบ trapezoidal



ภาพที่ 56 แสดงการทำ Fuzzification ของ 1 input fuzzy

11.2 ออกแบบโปรแกรมส่วน Fuzzy Reasoning Unit ในส่วนนี้ได้มีการประกาศตัวแปรชุดหนึ่งเพื่อทำหน้าที่เก็บค่าความเป็นสมาชิกสำหรับผลลัพธ์ของแต่ละกฎโดยที่ตัวแปรดังกล่าวแสดงให้เห็นดังภาพที่ 57 ในภาพแสดงให้ตัวแปรคือ uC ซึ่งใช้ในการเก็บค่าความเป็นสมาชิกของ output สำหรับแต่ละกฎ

```
typedef struct{
    unsigned char use;
    float uC;
}RuleTerm;
```

ใช้บอกว่า input fuzzy
เป็นไปตามกฎนี้หรือไม่มีค่าเป็น -1 แสดงว่าไม่ใช่
ค่าความเป็นสมาชิกของ Member ship function
สำหรับ output U ที่เป็นผลของการตัดสินใจ

ภาพที่ 57 แสดงโครงสร้างข้อมูลสำหรับ Fuzzy Rule

ตารางที่ 4 แสดงค่า index สำหรับตัวแปร use

index	error	derror	iererror	Uout
0	errLneg	errneg	errneg	UoutZero
1	errSneg	errZero	errPos	UoutSmall
2	errZero	errPos		UoutMedium
3	errSPos			UoutLarge
4	errLPos			

```
/* Rule1 */
{ (FuzzyProcess.FuzzErrTerm.Term[errLneg])&&(FuzzyProcess.FuzzDErrTerm.Term[derrneg])
  &&(FuzzyProcess.FuzzIErrTerm.Term[erneg]) )
  // drive output hight //
  FuzzyProcess.Rule[0].use=UoutLarge;
  FuzzyProcess.Rule[0].uC=FuzzyProcess.FuzzErrTerm.uC[errLneg]* FuzzyProcess.FuzzDErrTerm.uC[derrneg]
  FuzzyProcess.FuzzIErrTerm.uC[erneg];

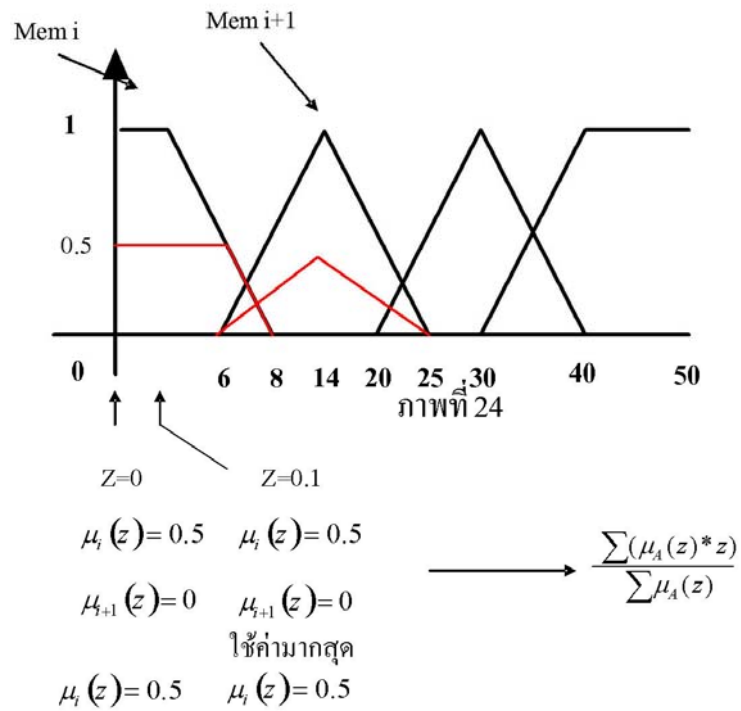
/* Rule2 */
{ (FuzzyProcess.FuzzErrTerm.Term[errLneg])&&(FuzzyProcess.FuzzDErrTerm.Term[derrneg])
  &&(FuzzyProcess.FuzzIErrTerm.Term[errpos]) )
  // drive output hight //
  FuzzyProcess.Rule[0].use=UoutLarge;
  FuzzyProcess.Rule[0].uC=FuzzyProcess.FuzzErrTerm.uC[errLneg]* FuzzyProcess.FuzzDErrTerm.uC[derrneg]
  FuzzyProcess.FuzzIErrTerm.uC[errpos];
```

ภาพที่ 58 ตัวอย่างการเขียน Fuzzy Rule ในบอร์ดควบคุม

11.3 ออกแบบโปรแกรมส่วน Defuzzifier การทำ Defuzzification ทำด้วยวิธี Centroid of area โดยใช้สมการ 29 คือ

$$z_{COA} = \frac{\sum (\mu_A(z) * z) * \Delta z}{\sum \mu_A(z) * \Delta z} \quad (29)$$

โดย Δz มีค่าเท่ากับ 0.1



ภาพที่ 59 แสดงตัวอย่างในการคำนวณสำหรับการทำ defuzzification

```

i=0;
FuzzyProcess.sigmaUz=0;
FuzzyProcess.sigmaUzz=0;
z=MFUoutTerm[i][1];                                     //          initial @ low bourn of mf term
while(i<NumofUout)
{
    uzn=0;
    uzn_1=0;
    while(z<MFUoutTerm[i][0])                          //integral untial z >= hi bound of mf term
    {
        uzn=MF_Value(z, MFUout[i][pa], MFUout[i][pb], MFUout[i][pc], MFUout[i][pd], MFUout[i][pType]);
        uzn=uzn*FuzzyProcess.FuzzUTerm.uC[i];
        if(i<NumofUout-1)
        {
            uzn_1=MF_Value(z, MFUout[i+1][pa], MFUout[i+1][pb], MFUout[i+1][pc], MFUout[i+1][pd], MFUout[i+1][pType]);
            uzn_1=uzn_1*FuzzyProcess.FuzzUTerm.uC[i+1]; }
        else
        {   uzn_1=0;   }
        if(uzn<uzn_1)                                uzn=uzn_1;
        uzz=z*uzn;
        FuzzyProcess.sigmaUz+=uzn;
        FuzzyProcess.sigmaUzz+=z*uzn;
        z+=0.1;                                       //shift of next step by increment 0.1 per step
    }
    z=MFUoutTerm[i][0];                             //initial @ low bourn of mf term
    i++;                                             //shift of next term
}

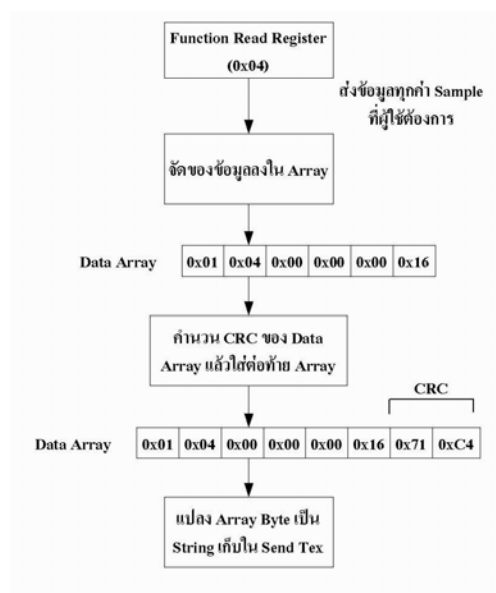
```

ภาพที่ 60 แสดงตัวอย่างโปรแกรมที่เขียนขึ้นสำหรับ Defuzzification

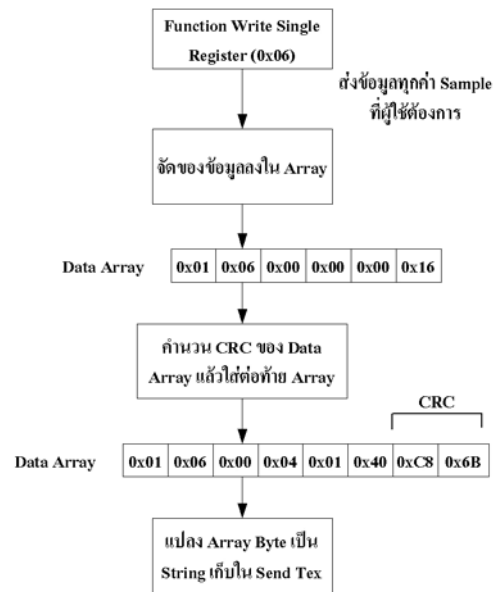
12. การออกแบบโปรแกรมอ่านค่าตัวแปรต่างของบอร์ดควบคุมมาแสดงผลบนคอมพิวเตอร์

การออกแบบโปรแกรมอ่านค่านี้ใช้ Labview เป็นโปรแกรมในการออกแบบ เนื่องจากโปรแกรม Labview มีส่วนของ Graphic Interface ที่ใช้งานได้ง่ายและมีประสิทธิภาพ โดยการออกแบบโปรแกรมจะทำการแบ่งงานออกเป็น 3 ส่วนคือ การสื่อสารกับบอร์ดควบคุมด้วยโปรโตคอล Modbus RTU, Human Interface และการเก็บข้อมูลที่อ่านจากบอร์ดควบคุมลงใน file excel

12.1 การสื่อสารกับบอร์ดควบคุมด้วยโปรโตคอล Modbus RTU การเขียนโปรแกรมของส่วนสื่อสารจะแบ่งออกเป็นงานย่อยอีกคือ การจัดเตรียมข้อมูลที่จะส่งไปยังบอร์ดควบคุม, การส่งข้อมูลออก Serial Port ของ คอมพิวเตอร์ด้วย VISA Write ของ โปรแกรม Labview , การอ่านข้อมูลที่ตอบกลับจากบอร์ดควบคุมด้วย VISA Read และการแปลความหมายของโปรโตคอลออกเป็นข้อมูลอุณหภูมิ โดยการจัดเตรียมข้อมูลของโปรแกรมจะแบ่งออกเป็นข้อมูล 2 ชุด ดังภาพที่ 61 แสดงการจัดเตรียมข้อมูลสำหรับอ่านค่าอุณหภูมิจากบอร์ดควบคุม และภาพที่ 62 แสดงการจัดเตรียมข้อมูลสำหรับเขียนข้อมูลลงบนบอร์ดควบคุม

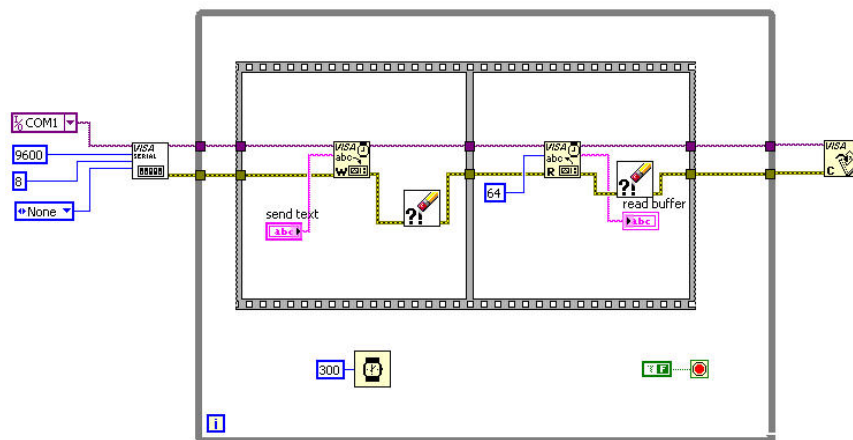


ภาพที่ 61 แสดงการจัดเตรียมข้อมูลสำหรับอ่านค่าอุณหภูมิจากบอร์ดควบคุม



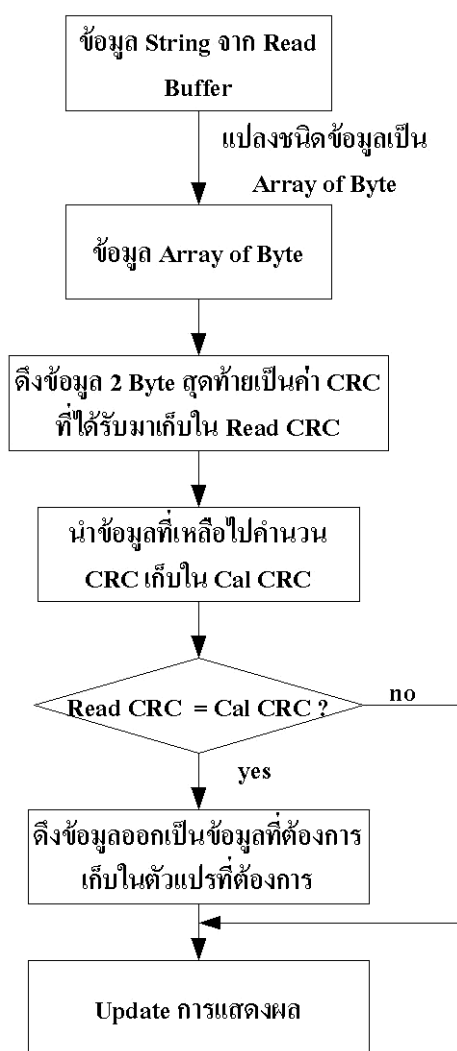
ภาพที่ 62 แสดงการจัดเตรียมข้อมูลสำหรับเขียนข้อมูลลงบนบอร์ดควบคุม

การเขียนและการอ่านข้อมูลจาก VISA Write และ VISA Read ของ Labview ใช้โครงสร้างดังภาพที่ 63 โดยที่ในภาพนี้จะเป็นการแสดงตัวอย่างง่าย ๆ ในการส่งและอ่าน เนื่องจากในโปรแกรมจริงมีขนาดใหญ่เกินกว่าที่จะแสดงให้เห็นได้ หลักการทำงานคือเมื่อส่งข้อมูลแล้วจะทำการรอการตอบกลับมายัง VISA Read

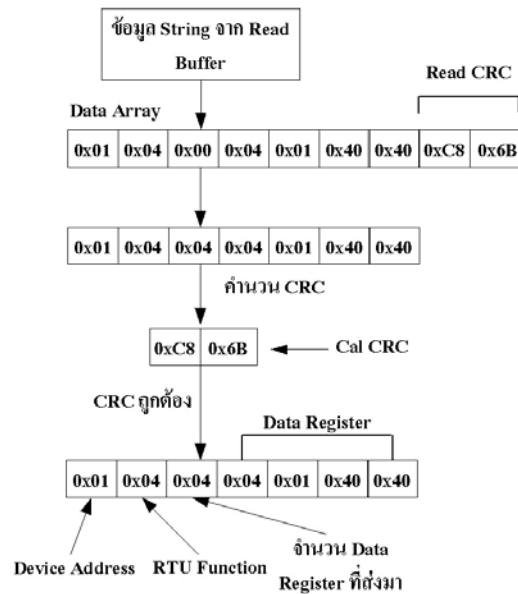


ภาพที่ 63 แสดงตัวอย่างโครงสร้างของการใช้งาน VISA Write และ VISA Read

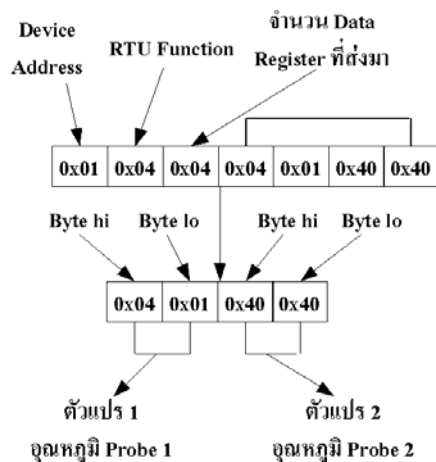
การแปลความหมายของข้อมูลที่ได้จาก VISA Read โดยการทำงานเริ่มจากตรวจสอบความถูกต้องของข้อมูลก่อนจากการตรวจค่า CRC หากข้อมูลที่ส่งมาถูกต้องให้ทำการแปลความหมายข้อมูลดังแสดงขั้นตอนการทำงานในภาพที่ 64 และภาพที่ 65 และภาพที่ 66 แสดงการแปลความหมายของ Data



ภาพที่ 64 แสดงกระบวนการตรวจสอบข้อมูลที่อ่านมาได้



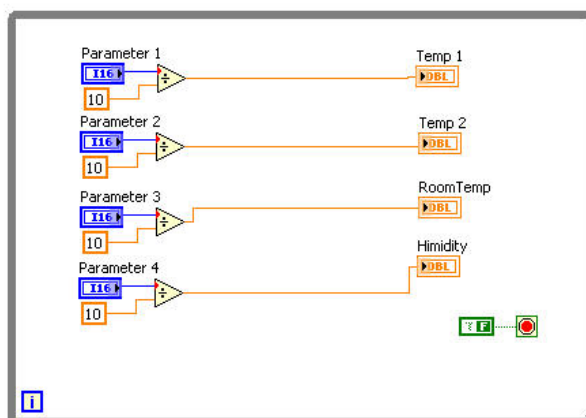
ภาพที่ 65 แสดงตัวอย่างการอ่าน Data จาก Modbus RTU message



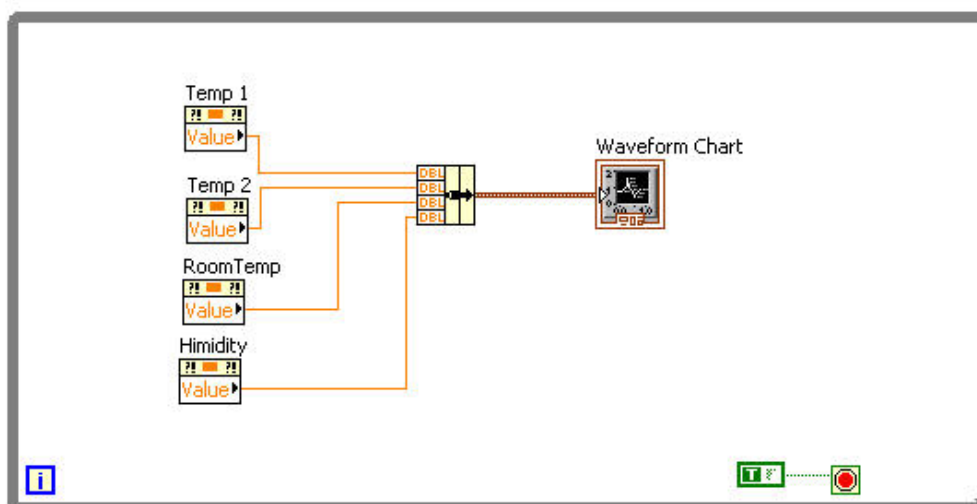
ภาพที่ 66 แสดงตัวอย่างการแปลความหมายของ Data Register จาก Modbus RTU message

12.2 Human Interface ในส่วน Human Interface ประกอบด้วยส่วนที่ 1 คือการแสดงผลค่าตัวแปรด้วยกราฟและ การแสดงผลเป็นตัวเลข ส่วนที่ 2 คือการเขียนค่าตัวแปร SV และกดปุ่มส่งข้อมูลโดยที่การแสดงค่าตัวแปรตัวเลขอาศัยคุณสมบัติของ VI Numeric Indicator ในการแสดงผลและแสดงผลด้วยกราฟจะใช้ VI WaveForm Chart โดยภาพที่ 67 แสดงให้เห็น

ตัวอย่างในการใช้งาน Numeric Indicator และ ภาพที่ 68 แสดงตัวอย่างการใช้งาน WaveForm Chart



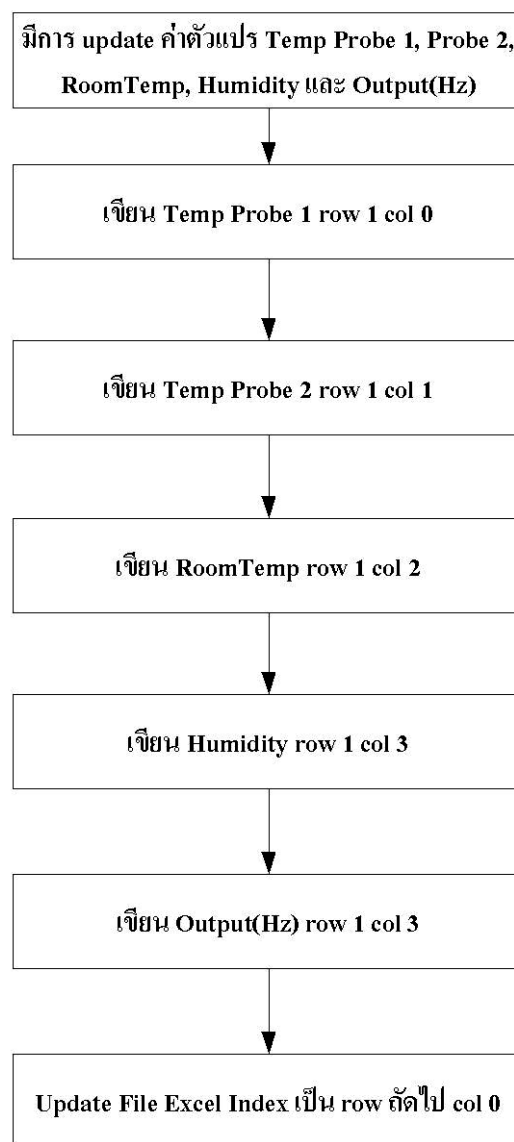
ภาพที่ 67 แสดงตัวอย่างการใช้งาน Numeric Indicator



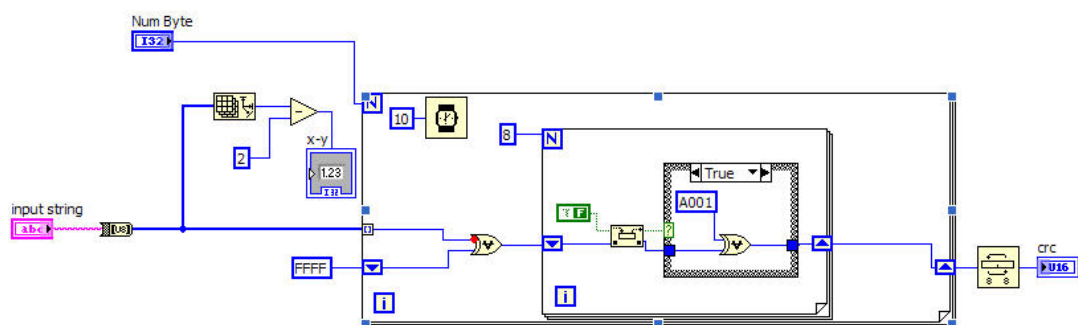
ภาพที่ 68 แสดงตัวอย่างการใช้งาน Waveform Chart

12.3 การและการเก็บข้อมูลที่อ่านจากบอร์ดควบคุมลงใน file excelการเขียนข้อมูลลงใน file excel กระทำโดยใช้งาน NewReport ในการสร้าง file และใช้งาน Excel Easy Text ในการเขียนข้อมูลลงใน file ที่ละ cell หลังจากจบการทำงานจึง Save file excel นี้ด้วย Save Report to File โดย

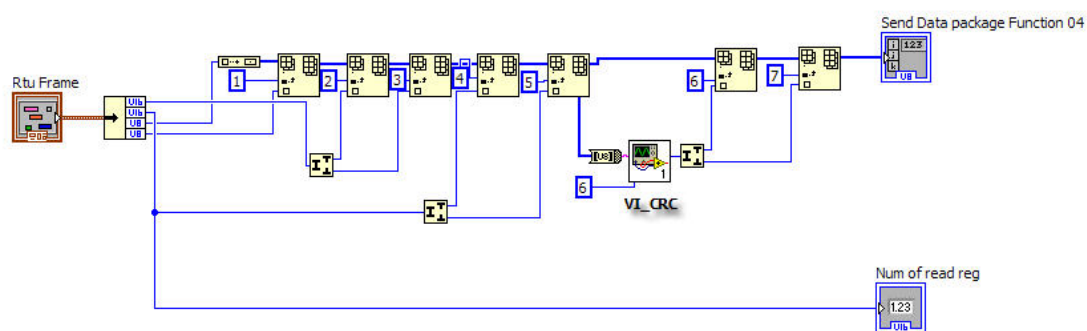
ที่โปรแกรมจะมีการเขียนข้อมูลลงใน file ทุกครั้งที่มีการ update ค่าตัวแปรต่างๆ ตาม Sampling Time ที่ผู้ใช้งานตั้งไว้โดยการเขียนจะเริ่มเขียนตั้งแต่ Cell หลักแรกแล้วจึงเขียนข้อมูลลงในหลักถัดไปจนครบทุกตัวแปรแล้วจึงไปเขียนข้อมูลลงในแถวถัดไปโดยการทำงานแสดงให้เห็นในภาพที่ 69 และภาพที่ 70 แสดงตัวอย่างการใช้งาน Excel Report ในโปรแกรม Labview



ภาพที่ 69 แสดงกระบวนการเขียนข้อมูลลงใน file excel



ภาพที่ 73 แสดงโปรแกรม Sub VI ที่ใช้ในการคำนวณ CRC ขนาด 16 bits

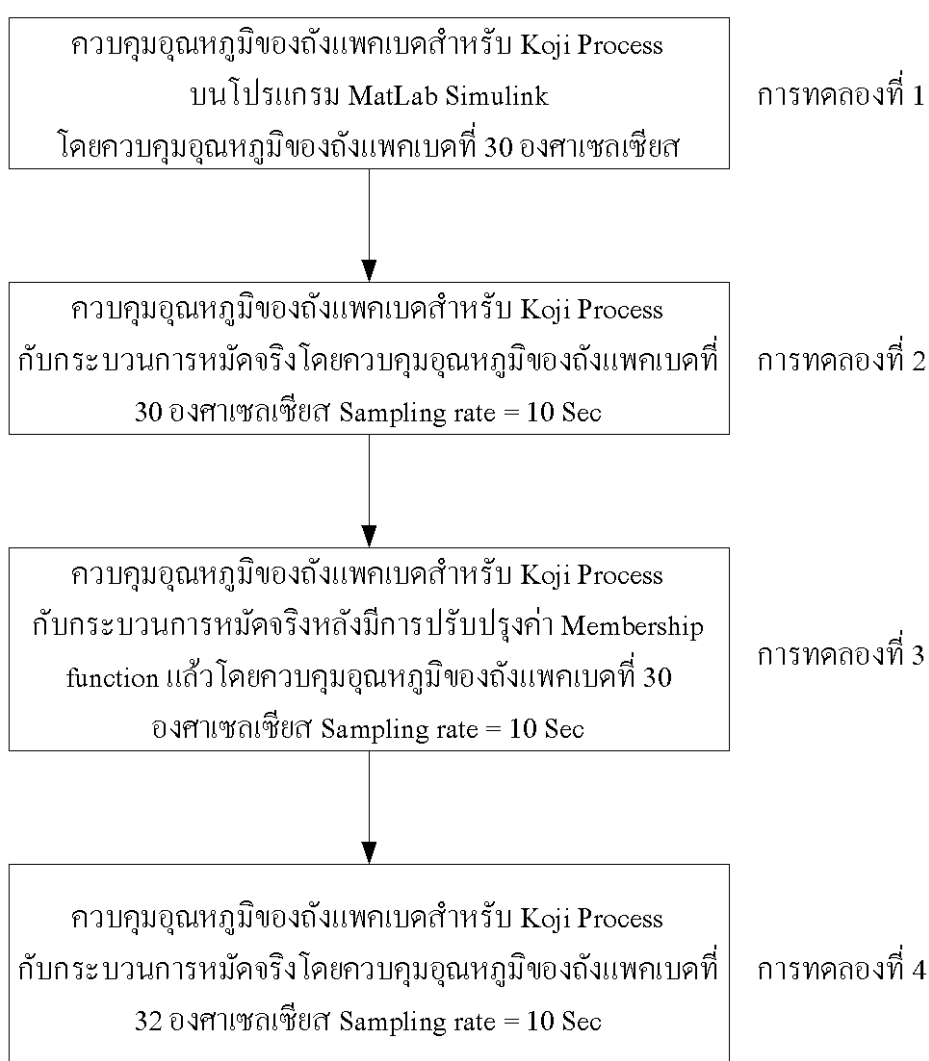


ภาพที่ 74 แสดงโปรแกรม Sub VI ที่ใช้ในการเตรียมข้อมูลสำหรับส่งผ่านทาง Serial Port

ผลและวิจารณ์

ผล

ในงานวิจัยนี้ได้แบ่งการทดลองออกเป็น 4 การทดลองดังภาพที่ 75



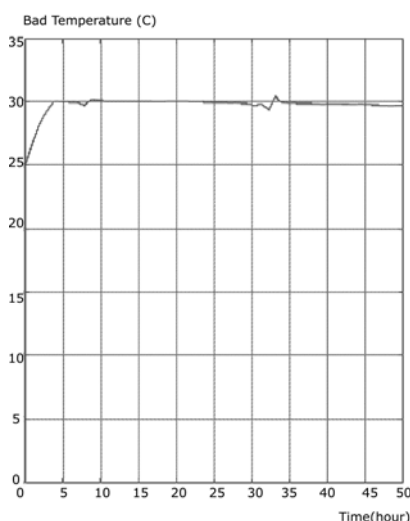
ภาพที่ 75 แผนผังแสดงขั้นตอนในการทดลอง

การทดลองที่ 1 ควบคุมอุณหภูมิของถังแพคเบตสำหรับ Koji Process บนโปรแกรม MatLab Simulink โดยควบคุมอุณหภูมิของถังแพคเบตที่ 30 องศาเซลเซียส

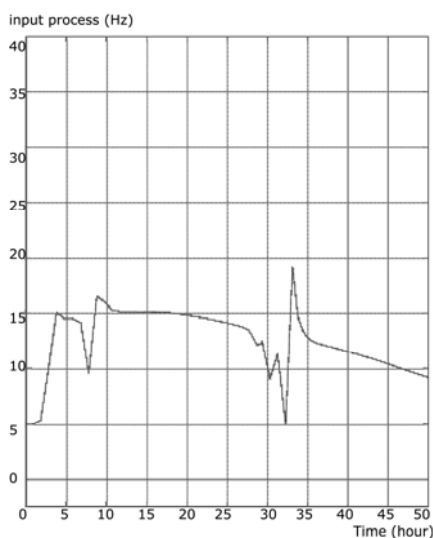
การทดลองที่ 1 เป็นการทดลองเพื่อหา Member ship function และ Fuzzy rule ที่ใช้ในการควบคุม โดย Mandani Fuzzy Inference System แบบ min max composition ในการควบคุมใช้ input Fuzzy 3 ตัวแปรคือ error(e), diff of error($\frac{de}{dt}$) และ integral of error($\int edt$)

เลือก Member ship function จำนวน 5 แบบสำหรับ e , เลือก Member ship function จำนวน 4 แบบสำหรับ $\frac{de}{dt}$ และ เลือก Member ship function จำนวน 2 แบบสำหรับ $\int edt$ เลือก Member ship function จำนวน 5 แบบสำหรับ Output Fuzzy u และการทำ Defuzzification เลือกใช้วิธี Centroid of area Method

ในการทดลองบนโปรแกรม Simulate เราให้อุณหภูมิภายนอกไม่มีผลต่อระบบและตัวแปรที่เป็นปัจจัยอื่นๆถูกจำกัดไว้ตามสมมุติฐานของ Math Model (ปทุมมา 2546) และให้อุณหภูมิภายนอกมีค่าคงที่คือ 29 องศาเซลเซียส ผลการทดลองที่ได้แสดงให้เห็นในภาพที่ 76 และ ภาพที่ 77 จากกราฟแสดงให้เห็นว่า อุณหภูมิของวัสดุหมักถูกควบคุมให้ต่ำกว่าค่า 30 องศาเซลเซียส



ภาพที่ 76 แสดงผลของอุณหภูมิภายในถังแพคเบตที่ควบคุมโดย Fuzzy PID



ภาพที่ 77 แสดงค่า input process ที่ใช้ในการควบคุม

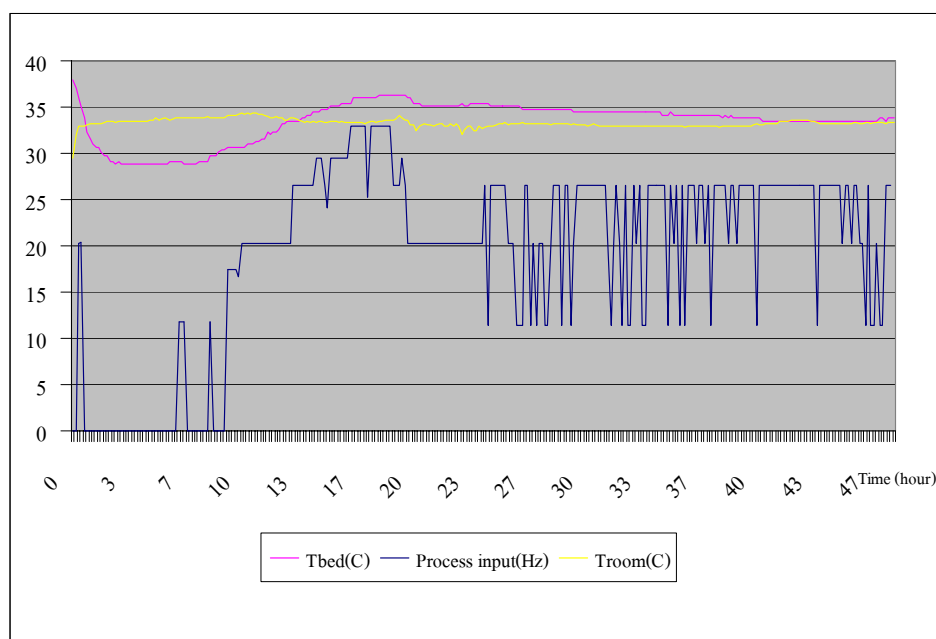
การทดลองที่ 2 ควบคุมอุณหภูมิของถังแพคเบตสำหรับ Koji Process กับกระบวนการหมักจริงโดย ควบคุมอุณหภูมิของถังแพคเบตที่ 30 องศาเซลเซียส

ในการทดลองนี้จะใช้ Member ship function และ Fuzzy rule ที่ได้จากการทดลองที่ 1 มา เขียนเป็นโปรแกรมควบคุมในบอร์ดควบคุม PIC24HJ128GP306 โดยการทดลองนี้กระทำในห้อง ปิดที่และท่อที่ใช้สำหรับการดูดอากาศจะถูกต่อออกไปนอกตัวห้อง

การทดลองจะทำการหมักถั่วเหลือง 2 กิโลกรัมภายในถังหมักขนาด 40 ลิตรและทำการ ควบคุมอุณหภูมิของถั่วเหลืองในระหว่างกระบวนการหมักให้อยู่ที่ 30 องศาเซลเซียส ผลที่ได้จาก การทดลองแสดงให้เห็นในภาพที่ 78 โดยข้อมูลดังภาพเป็นข้อมูลถูกเก็บค่าทุก 10 นาที และ โปรแกรมควบคุมมีค่า Sampling rate คือ 10 วินาที

จากกราฟ แสดงให้เห็นว่าช่วงแรกอุณหภูมิ ของวัสดุหมักมีค่าสูงกว่า 40 องศาเซลเซียส เนื่องจากความร้อน ในกระบวนการฆ่าเชื้อด้วยการนึ่งถั่วเหลืองก่อนการหมักหลังจากที่อุณหภูมิ ของวัสดุหมักต่ำกว่าลงช่วงระยะเชื้อราที่ใช้ในการหมักจึงเริ่มทำงานจากกราฟอุณหภูมิของวัสดุหมัก (Tbed) แสดงให้เห็นว่าอุณหภูมิสูงกว่าเป้าหมายต้อง การคือ 30 องศาเซลเซียส โดยที่อุณหภูมิของ วัสดุหมักเมื่อจบการทดลองแล้วอยู่ที่ 32 องศาเซลเซียส และความถี่ที่ใช้ขับ Inverter Drive อยู่ที่

26.6 Hz อุณหภูมิของวัสดุหมักมีค่ามากที่สุดที่ 36.3 องศาเซลเซียส ณ ชั่วโมงที่ 18 ของ การหมัก และ ขณะนั้นความถี่ที่ใช้ขับ Inverter Drive อยู่ที่ 33 Hz อุณหภูมิของวัสดุหมักเริ่มมีค่ามากกว่า 30 องศาเซลเซียสในชั่วโมงที่ 9 ของการหมัก



ภาพที่ 78 แสดงผลการทดลองที่ได้ของการทดลองที่ 2

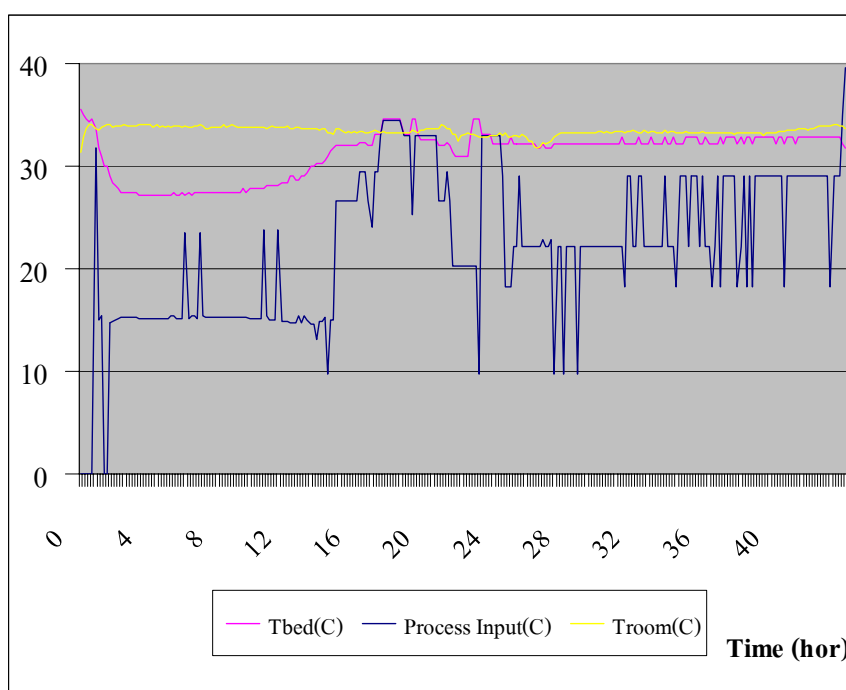
การทดลองที่ 3 ควบคุมอุณหภูมิของถังแพคเบดสำหรับ Koji Process กับกระบวนการหมักจริง
หลังมีการปรับปรุงค่า Membership function

ในการทดลองนี้ได้ใช้ข้อมูลก่อนหน้าในการปรับปรุงค่า Member ship function ให้สามารถควบคุมอุณหภูมิได้ดีขึ้น โดยทำการปรับค่า Member ship function และในการทดลองนี้ ได้มีการปรับปรุงโครงสร้างของ Input Fuzzy ด้วยโดยมีการคูณค่า gain $K_i = 1.5$ เข้าที่ input e และ gain $K_d = -1.5$ เข้าที่ input $\int edt$ ก่อนส่งค่าเข้าไปคำนวณใน Fuzzy PID

การทดลองจะทำการหมักถั่วเหลือง 2 กิโลกรัม ภายในถังหมักขนาด 40 ลิตรและทำการควบคุมอุณหภูมิของถั่วเหลือง ในระหว่างกระบวนการหมักให้อยู่ที่ 30 องศาเซลเซียส ผลที่ได้จาก

การทดลองแสดงให้เห็นในภาพที่ 79 โดยข้อมูลดังกล่าวเป็นข้อมูลถูกเก็บค่าทุก 10 นาทีและโปรแกรมควบคุมมีค่า Sampling rate คือ 10 วินาที

จากกราฟ อุณหภูมิของวัสดุหมัก (Tbed) แสดงให้เห็นว่าอุณหภูมิสูงกว่าเป้าหมายต้องการ คือ 30 องศาเซลเซียส โดยที่อุณหภูมิของวัสดุหมักเมื่อจบการทดลองแล้วอยู่ที่ 31.8 องศาเซลเซียส และความถี่ที่ใช้ขับ Inverter Drive อยู่ที่ 39.6 Hz อุณหภูมิของวัสดุหมักมีค่ามากที่สุดที่ 34.6 องศาเซลเซียส ณ ชั่วโมงที่ 18 ของ การหมัก และขณะนั้นความถี่ที่ใช้ขับ Inverter Drive อยู่ที่ 34.5 Hz อุณหภูมิของวัสดุหมักเริ่มมีค่ามากกว่า 30 องศาเซลเซียสในชั่วโมงที่ 13 ของการหมัก



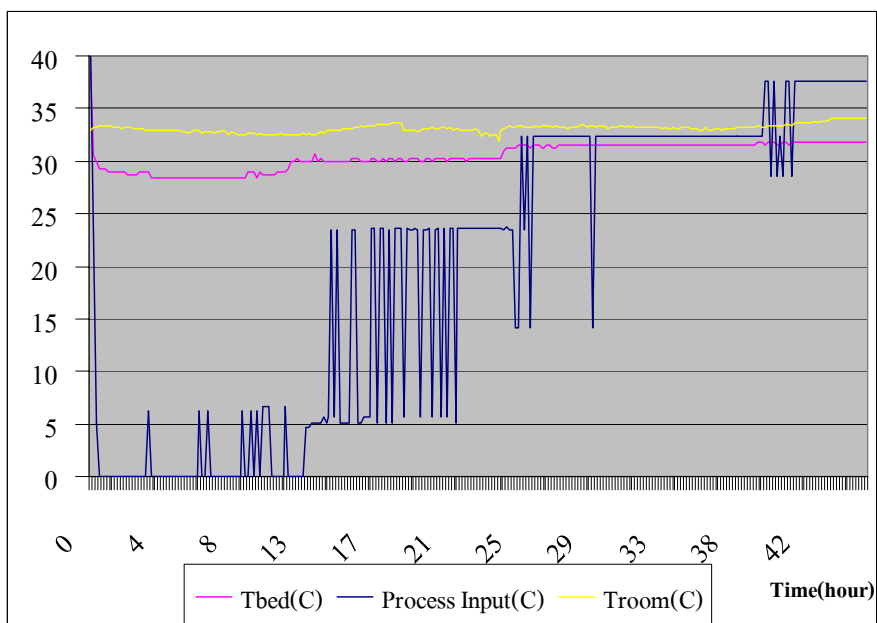
ภาพที่ 79 แสดงผลการทดลองที่ได้ของการทดลองที่ 3

การทดลองที่ 4 ควบคุมอุณหภูมิของถังแพคเบตสำหรับ Koji Process กับกระบวนการหมักจริงโดยควบคุมอุณหภูมิของถังหมักให้อยู่ที่ 32 องศาเซลเซียส

ในการทดลองนี้ได้ใช้ข้อมูลก่อนหน้าในการปรับค่า Member ship function ให้สามารถควบคุมอุณหภูมิได้ดีขึ้น โดยทำการปรับค่า Member ship function และในการทดลองนี้ได้มีการปรับปรุงโครงสร้างของ Input Fuzzy ด้วยโดยมีการคูณค่า gain $K_i = 1.5$ เข้าที่ input e และ gain $K_d = -0.5$ เข้าที่ input $\int edt$ ก่อนส่งค่าเข้าไปคำนวณใน Fuzzy PID

การทดลองจะทำการหมักถั่วเหลือง 2 กิโลกรัมภายในถังหมักขนาด 40 ลิตรและทำการควบคุมอุณหภูมิของถั่วเหลือง ในระหว่างกระบวนการหมักให้อยู่ที่ 30 องศาเซลเซียส ผลที่ได้จากการทดลองแสดงให้เห็นในภาพที่ 80 โดยข้อมูลดังกล่าวเป็นข้อมูลถูกเก็บค่าทุก 10 นาที และโปรแกรมควบคุมมีค่า Sampling rate คือ 10 วินาที

จากกราฟอุณหภูมิของวัสดุหมัก (Tbed) แสดงให้เห็นว่าอุณหภูมิสูงกว่าเป้าหมายต้องการ คือ 32 องศาเซลเซียส โดยที่อุณหภูมิของวัสดุหมักเมื่อจบการทดลองแล้วอยู่ที่ 31.8 องศาเซลเซียส และความเร็วที่ไช้ขับ Inverter Drive อยู่ที่ 37.6 Hz อุณหภูมิของวัสดุหมักมีค่ามากที่สุดที่ 31.8 องศาเซลเซียส ณ ชั่วโมงที่ 38 ของ การหมัก และขณะนั้นความเร็วที่ไช้ขับ Inverter Drive อยู่ที่ 32.3 Hz

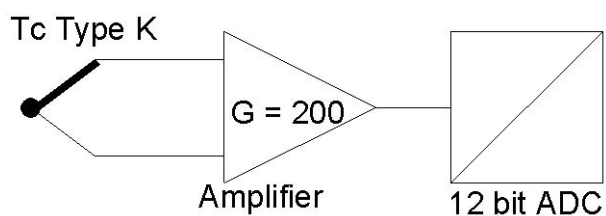


ภาพที่ 80 แสดงผลการทดลองที่ได้ของการทดลองที่ 4

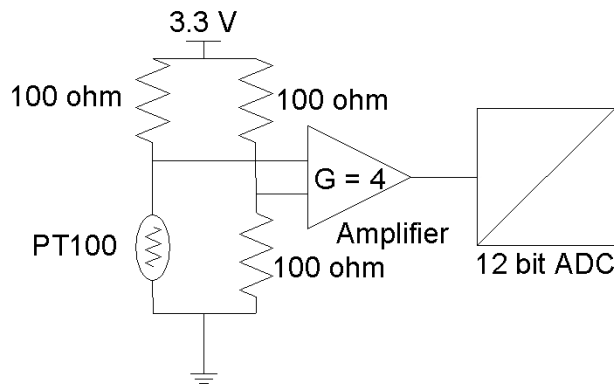
วิจารณ์

การหาความผิดพลาดในการวัดอุณหภูมิของบอร์ดควบคุม

การอ่านค่าอุณหภูมิของบอร์ดควบคุมจะในวงจรดังภาพที่ 81 และ 82 โดยใช้ ADC ขนาด 12 bit ในการอ่านค่า mV จากวงจรที่ผ่านการขยายมาแล้ว



ภาพที่ 81 แสดงวงจรที่ใช้สำหรับอ่านค่า Thermocouple Type k



ภาพที่ 82 แสดงวงจรที่ใช้สำหรับอ่านค่า PT100

ค่าความละเอียดของ ADC 12 bit ที่ใช้ Volt Reference อยู่ที่ 3.3 volte คือ

$$\frac{V_{ref}}{4096} = \frac{3.3}{4096} = 0.805664 \text{ mV}$$

จากตารางค่า Volt Output ของ Thermocouple Type k ค่า Volt Output มีค่าเปลี่ยนแปลง $39 \text{ uV} / ^\circ\text{C}$ ดังนั้นเมื่อผ่านวงจรขยาย 200 เท่าแล้ว Volt Output ของวงจรวัดค่าอุณหภูมิจาก Thermocouple Type k จะมีค่า $7.8 \text{ mV} / ^\circ\text{C}$ ดังนั้นค่า ADC ที่บอร์ดควบคุมต้องอ่านได้ต่อการเปลี่ยนแปลง 1 องศา คือ 9.68 แต่อาจอ่านค่าข้อมูลของ ADC นั้นอ่านออกมาเป็นค่า ADC ดังนั้นทำให้ค่าที่อ่านได้จริงเป็น 10 นั้นค่ามีความผิดพลาดจาก ADC อยู่ที่ ± 0.4 องศาเซลเซียส โดยคำนวณ ได้จาก สมการ 30

$$error = \pm \left(\frac{ADC - RealValue}{RealValue} \right) \times 1 \quad (30)$$

จากการอ่านอุณหภูมิห้องด้วยวงจรอ่าน NTC มีความผิดพลาดในการอ่านอุณหภูมิอยู่ที่ ± 1 องศาเซลเซียส จากสมการ 24 ที่ใช้ในการคำนวณหาอุณหภูมิของบอร์ดควบคุมแสดงให้เห็น ค่าความผิดพลาดในการอ่านอุณหภูมิของ Thermocouple Type k เป็นไปตามสมการที่ 31

$$Error_{total} = Error_{V_{thermocouple}} + Error_{NTCSensor} \quad (31)$$

ดังนั้นค่าความผิดพลาดของการอ่านอุณหภูมิด้วย Thermocouple Type k อยู่ที่ ± 1.4 องศาเซลเซียส

การหาค่าความผิดพลาดในการอ่านอุณหภูมิของวงจรอ่านอุณหภูมิจาก PT100 จากวงจรเราสามารถคำนวณหาค่า Volt output (V_{pt100}) จากวงจรได้ดังนี้

$$V_{pt100} = \left(\left(\frac{3.3}{R_{pt100} + 100} \right) \times R_{pt100} - 1.65 \right) \times 4 \quad (32)$$

จากตารางค่า PT100 ได้ว่าเมื่ออุณหภูมิเปลี่ยนแปลง 1 องศาเซลเซียส ค่าความต้านทานของ PT100 จะเพิ่มขึ้น 0.39 ohm ดังนั้นเราจะได้ว่า V_{pt100} มีค่าเปลี่ยนแปลง 12.844 mV/C ดังนั้นค่า ADC ที่บอร์ดควบคุมต้องอ่านได้ต่อการเปลี่ยนแปลง 1 องศา คือ 15.94 แต่หากอ่านค่าข้อมูลของ ADC นั้นอ่านออกมาเป็นค่า ADC ดังนั้นทำให้ค่าที่อ่านได้จริงเป็น 16 ดังนั้นจากสมการที่ 29 เราสามารถหาค่าความผิดพลาดจาก ADC อยู่ที่ ± 0.004 องศาเซลเซียส

แต่จากการวัดค่าความผิดพลาดเมื่อใช้งานร่วมกับ Inverter Drive โดยเทียบกับ Thermometer แล้วได้ค่าความผิดพลาดจากการอ่านอุณหภูมิหลังทำการปรับจูนแล้วประมาณ 1 องศาเซลเซียส สาเหตุอาจมาจากสัญญาณลบกวนเมื่อ Inverter Drive ทำงาน

ผลการทดลองที่ 1 จากกราฟอุณหภูมิของภาพที่ 76 แสดงให้เห็นว่า Fuzzy PID สามารถควบคุมให้อุณหภูมิของวัสดุหมักมีค่าอยู่ที่ 30 องศาเซลเซียส ได้แต่ยังคามีค่า OverShoot อยู่ 0.4 องศาเซลเซียส และต่ำกว่าอยู่ 0.5 องศาเซลเซียส ในช่วงที่ระบบมีการเปลี่ยนแปลงอุณหภูมิอย่างรวดเร็วซึ่งเป็นผลมาจากการทำงานของเชื้อราในช่วงปฏิกิริยาการเจริญเติบโตของจุลินทรีย์ใน ระยะการเจริญเติบโตแบบลอการิทึม (logarithmic phase) และจากภาพที่ 76 และ ภาพที่ 77 จะสังเกตได้ว่าทั้งอุณหภูมิ ของวัสดุหมักและค่า output ของ Fuzzy PID มีการกระชากอย่างแรง เนื่องมาจากค่า error ขณะนั้น อยู่ในช่วง Membership function zero และ ค่า derror แกว่งอยู่ในช่วง Positive large และ Negative large

ผลการทดลองที่ 2 จากกราฟอุณหภูมิของภาพที่ 78 แสดงให้เห็นว่า Fuzzy PID ที่ใช้ในการทดลองที่ 1 เมื่อนำมาทดลองกับระบบจริงให้ผลการควบคุมที่ไม่ดีนักและมีช่วงที่อุณหภูมิสูงสุดที่ 36.3 องศาเซลเซียส ในช่วงที่ระบบมีการเปลี่ยนแปลงอุณหภูมิอย่างรวดเร็วซึ่งเป็นผลมาจากการทำงาน ของเชื้อราในช่วงปฏิกิริยาการเจริญเติบโตของจุลินทรีย์ใน ระยะการเจริญเติบโตแบบลอการิทึม (logarithmic phase) การที่ผลการทดลองที่ 2 ให้ผลที่แตกต่างจากการทดลองที่ 1 เป็นผลมาจากการ ที่ค่าตัวแปรคงที่ใช้ในของสมการ Model Koji process ที่แตกต่างจากการทดลองจริง เนื่องมาจาก ตัวแปรของวัสดุหมักบางตัวที่ใช้ในการทดลองอาจไม่เท่ากันเสมอเช่น ความหนาแน่นของ วัสดุหมัก, มวลของน้ำ, ความชื้นในอากาศ, อุณหภูมิของอากาศขาเข้า โดยตัวแปรที่สังเกตได้ชัดเจน คืออุณหภูมิของอากาศขาเข้าซึ่งมีค่าเท่ากับอุณหภูมิห้องโดยมีค่าสูงกว่าในการทดลองที่ 1 และมีค่า สูงสุดถึง 34.3 องศาเซลเซียส ดังนั้นจึงต้องมีการปรับจูน Fuzzy PID ที่ใช้ในการทดลองที่ 1 จึงสามารถใช้กับระบบจริงได้

ผลการทดลองที่ 3 จากกราฟอุณหภูมิของภาพที่ 79 แสดงให้เห็นว่า Fuzzy PID ใช้ในการทดลองมี ประสิทธิภาพในการควบคุมดีขึ้นแต่ยังคงไม่สามารถลดค่าอุณหภูมิให้อยู่ที่ 30 องศาเซลเซียสได้ โดยค่าอุณหภูมิสูงสุดอยู่ที่ 34.6 องศาเซลเซียส ในช่วงที่ระบบมีการเปลี่ยนแปลงอุณหภูมิอย่าง รวดเร็วซึ่งเป็นผลมาจากการทำงาน ของเชื้อราในช่วงปฏิกิริยาการเจริญเติบโตของจุลินทรีย์ใน ระยะการเจริญเติบโตแบบลอการิทึม (logarithmic phase) แต่หลังจากนั้นจะมีค่าอุณหภูมิอยู่ที่ ประมาณ 32.1 องศาเซลเซียส โดยมีอุณหภูมิห้องอยู่ที่ 33.2 องศาเซลเซียส

ผลการทดลองที่ 4 จากกราฟอุณหภูมิของภาพที่ 80 แสดงให้เห็นว่า Fuzzy PID ใช้ในการทดลองมี ประสิทธิภาพในการควบคุมให้อยู่ที่ 32 องศาเซลเซียสได้ โดยค่าอุณหภูมิสูงสุดอยู่ที่ 31.8 องศาเซลเซียส ในช่วงท้ายของการทดลองและผ่านช่วงที่เชื้อราเริ่มปฏิกิริยาการเจริญเติบโตของ จุลินทรีย์ในระยะการเจริญเติบโตแบบลอการิทึม (logarithmic phase) ไปแล้ว โดยอุณหภูมิห้อง มีค่าอยู่ที่ 33.2 องศาเซลเซียส และมีค่าสูงสุด 34 องศาเซลเซียส ในช่วงท้ายของการทดลองเช่นกัน

สรุปและข้อเสนอแนะ

สรุป

จากผลการทดลองได้ว่าการควบคุมอุณหภูมิของกระบวนการหมักโคจีนนั้นสามารถใช้ตัวควบคุมแบบ Fuzzy PID ได้แต่ปัจจัยที่มีผลทำให้การควบคุมไม่สามารถควบคุมอุณหภูมิที่ 30 องศาเซลเซียส ได้คืออุณหภูมิของอากาศขาเข้าซึ่งมีค่าเท่ากับอุณหภูมิห้องคือมีค่าอยู่ประมาณ 33.2 องศาเซลเซียส และมีค่าสูงสุดประมาณ 34 องศาเซลเซียส ทำให้ค่าอุณหภูมิของอากาศขาเข้ามีค่า สูงกว่าค่าอุณหภูมิเป้าหมายของวัสดุหมักประมาณ 3.2 ถึง 4 องศาเซลเซียส ซึ่งมีผลต่ออัตรา การถ่ายเทความร้อนของระบบไปออกสู่ภายนอกด้วยวิธีการพา ความร้อนมีลดลงดังนั้นจึงมีผล ทำให้ไม่สามารถควบคุมอุณหภูมิที่ 30 องศาเซลเซียสได้

ข้อเสนอแนะ

1. การเลือกใช้งาน Sensor ที่ใช้ในการวัดอุณหภูมิควรเลือกใช้ Sensor PT100 หรือชนิดอื่นที่สามารถอ่านค่าผ่าน ADC 12 bit ได้เลยจะช่วยในการลดปัญหาการอ่านค่าเนื่องจาก สัญญาณลบจาก Inverter Drive ที่ถูกขยายผ่านวงจรขยายมาด้วย
2. ในกระบวนการหมักควรมีการปรับปรุงห้องทดลองให้มีอุณหภูมิลดลงโดยที่ไม่มีผลต่อปัจจัยความชื้นในอากาศเพื่อให้การถ่ายเทความร้อนทำได้ดีขึ้น

เอกสารและสิ่งอ้างอิง

ณัฐวุฒิ ชินธเนศ. 2547. การหาแบบจำลองของระบบแขนหุ่นยนต์โดยใช้ **Adaptive Neuro-Fuzzy Models**. วิทยานิพนธ์ปริญญาโท, มหาวิทยาลัยเกษตรศาสตร์.

ปณณา จารุรัตน์. 2546. แบบจำลองทางคณิตศาสตร์การถ่ายเทความร้อนและมวลแบบลัมปีของการหมักแห้งถั่วเหลืองในถังหมักชนิดแพคเบด. วิทยานิพนธ์ปริญญาโท, มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี.

พงษ์ประวัติ พิมพการัง. 2546. การควบคุมอุณหภูมิอัตโนมัติแบบคำนวณตัวแปรภายในถังแพคเบด. วิทยานิพนธ์ปริญญาโท, มหาวิทยาลัยเกษตรศาสตร์.

Jan Jantzen. 1998. **Tuning Of Fuzzy PID Controllers**. Tech. report on 98-H 871 (fpid)
Technical University of Denmark, Department of Automation, Bldg 326, DK-2800
Lyngby, DENMARK.

Kevin M. Passino and Stephen Yurkovich. 1998. **Fuzzy Control**. Addison Wesley Longman, Inc., Printed in the United States of America.

M. Mizumoto. 1996. Product-Sum-Gravity Method = Fuzzy Singleton-type Reasoning Method = Simplified Fuzzy Reasoning Method, pp. 2098-2102 vol.3 In **IEEE International Conference on Fuzzy Systems**.

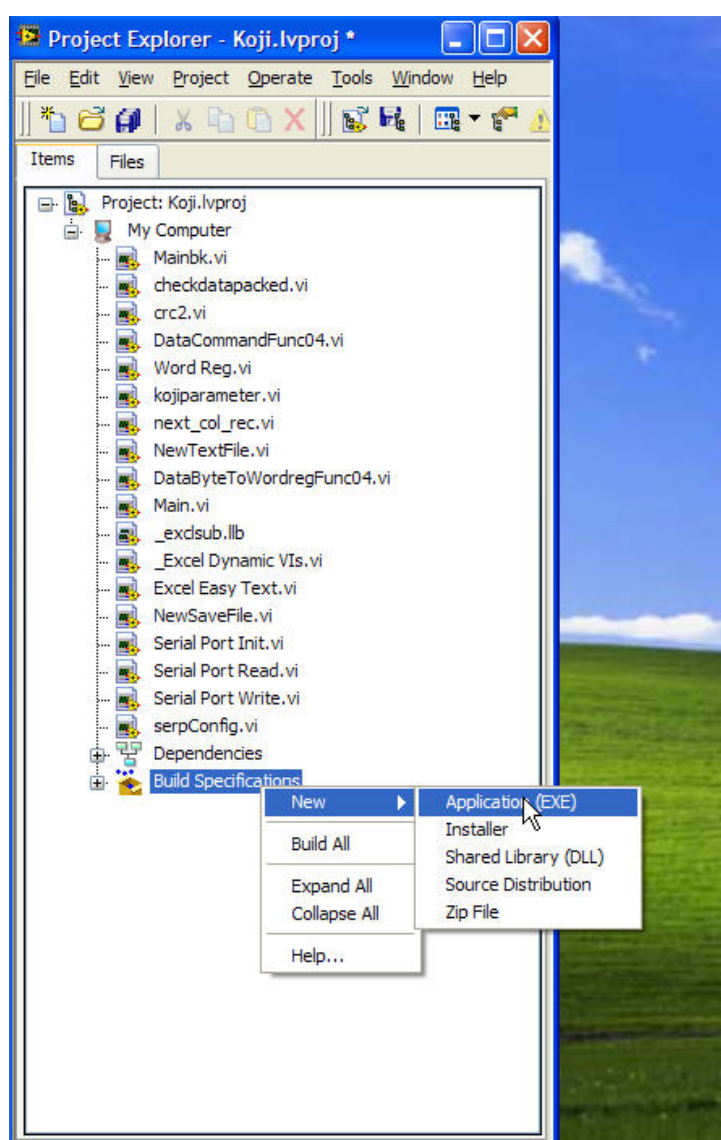
ภาคผนวก

ภาคผนวก ก

การสร้าง Setup File สำหรับโปรแกรมที่ถูกพัฒนาบนโปรแกรม LabVIEW8.2

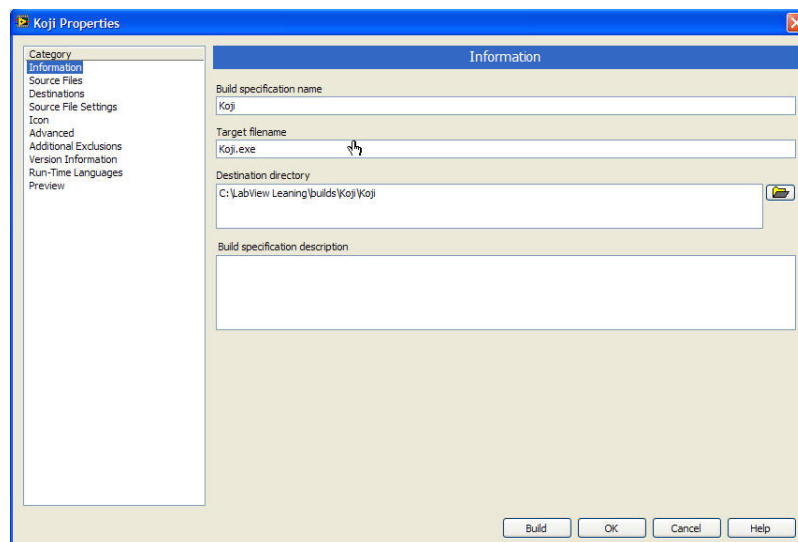
การสร้าง Setup File สำหรับโปรแกรมที่ถูกพัฒนาขึ้นบนโปรแกรม LabVIEW 8.2

1. โดยขั้นตอนในการสร้าง Setup File นั้นต้องทำการสร้าง Application file(EXE) ก่อน แล้วจึงสร้าง Installer
2. หลังจากทำการพัฒนาโปรแกรมเสร็จแล้วให้ทำการคลิกขวาที่ Build Specifications เลือก New -> Application (EXE) ดังภาพผนวกที่ ก1



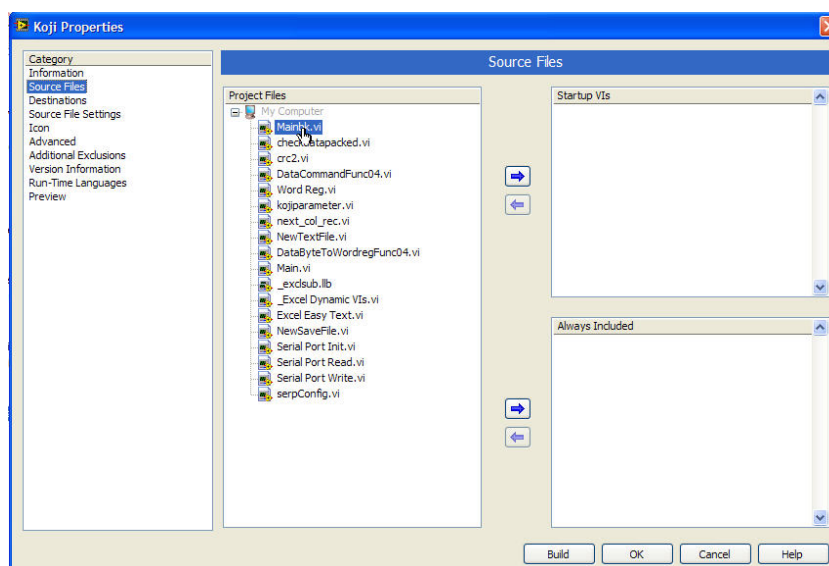
ภาพผนวกที่ ก1 เลือกสร้าง Application(EXE)

3. กำหนดค่า Build specification name, Target File name และ Destination directory ใน หน้าต่าง Information ดังภาพผนวก ก2

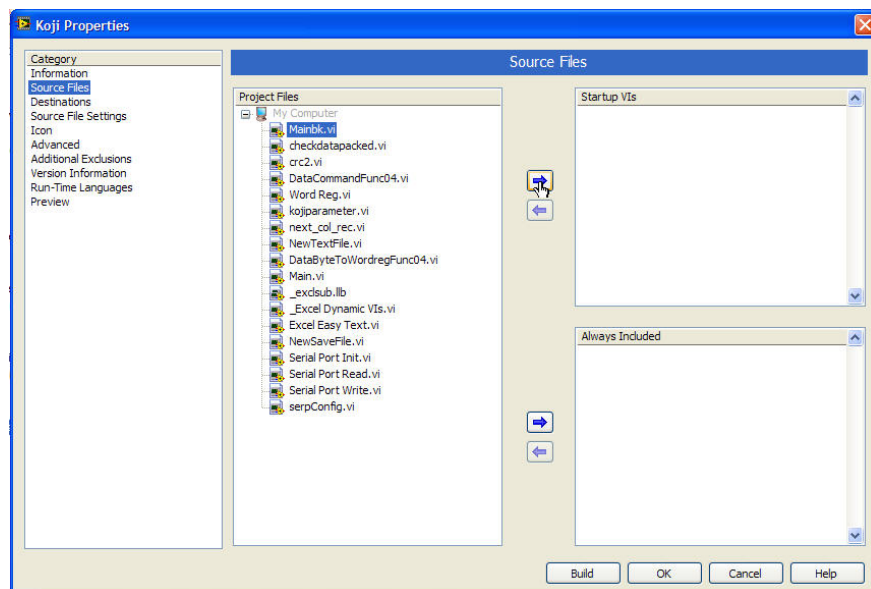


ภาพผนวกที่ ก2 กำหนดค่าที่ใน Information

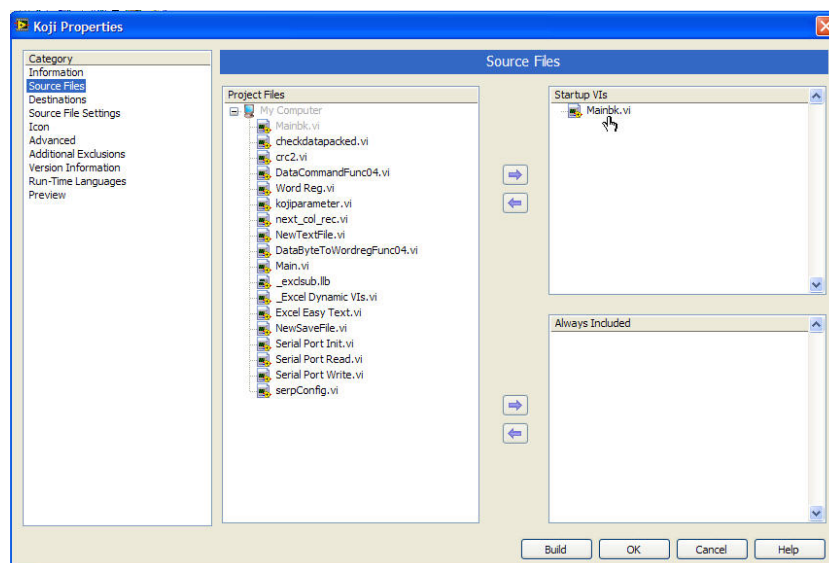
4. ทำการเลือก Source file ที่ต้องการนำไปสร้าง Application file โดยคลิกที่ชื่อ file VIs ที่ต้องการแล้วกดปุ่มลูกศรดังภาพผนวกที่ ก3, ภาพผนวกที่ ก4 และภาพผนวกที่ ก5



ภาพผนวกที่ ก3 คลิกที่ File VIs ที่ต้องการให้ทำงานบน Application

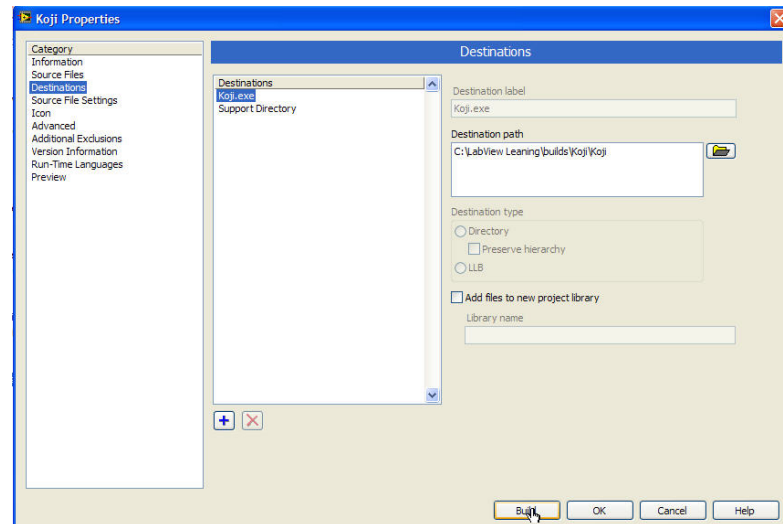


ภาพผนวกที่ ก4 คลิกที่ปุ่มรูปลูกศรเพื่อทำการเลือก File VIs

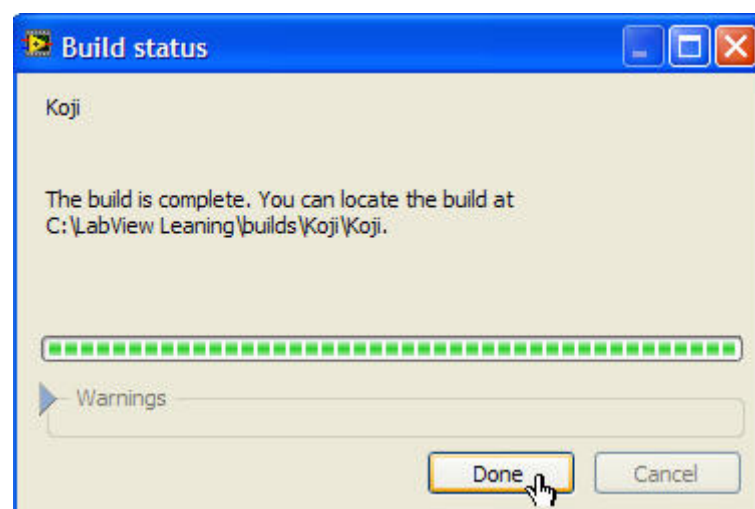


ภาพผนวกที่ ก5 หน้าต่าง Startup VIs จะแสดงชื่อ File VIs ที่เลือกไว้

5. ทำการสร้าง Application file โดยคลิกที่ Build จะได้ File EXE ที่ต้องการ

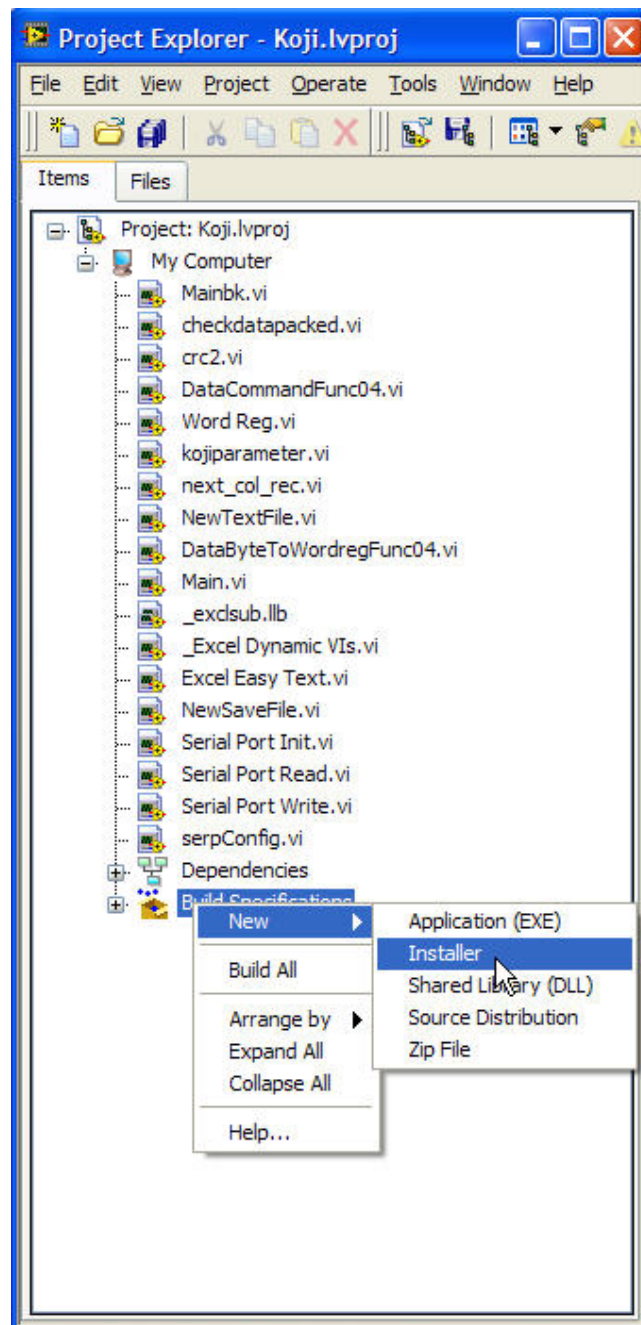


ภาพผนวกที่ 6 เริ่มการสร้าง Application file โดยคลิกที่ Build



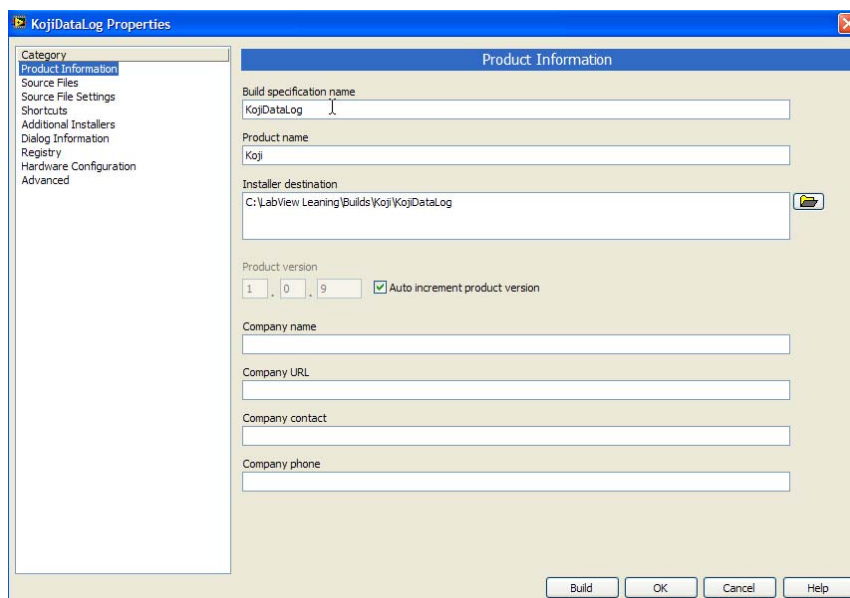
ภาพผนวกที่ 7 แสดงสถานะว่ากระบวนการเสร็จสมบูรณ์

6. หลังจากได้ Application file แล้วจึงเริ่มทำการสร้าง Setup file โดยคลิกขวาที่ Build Specification แล้วเลือก New -> Installer ดังภาพผนวกที่ ก8



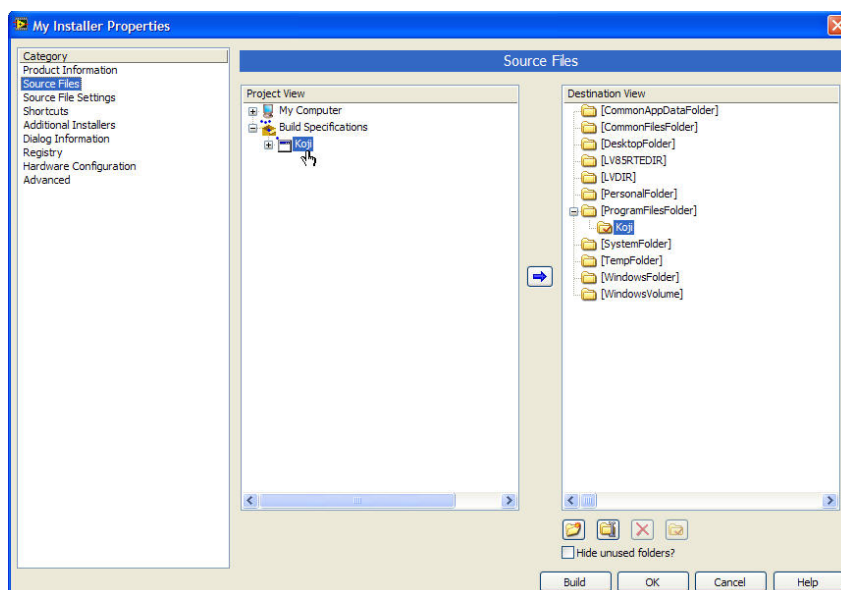
ภาพผนวกที่ ก8 เลือกสร้าง Installer

7. กำหนดค่าในหน้าต่าง Product Information ดังภาพผนวกที่ ก9

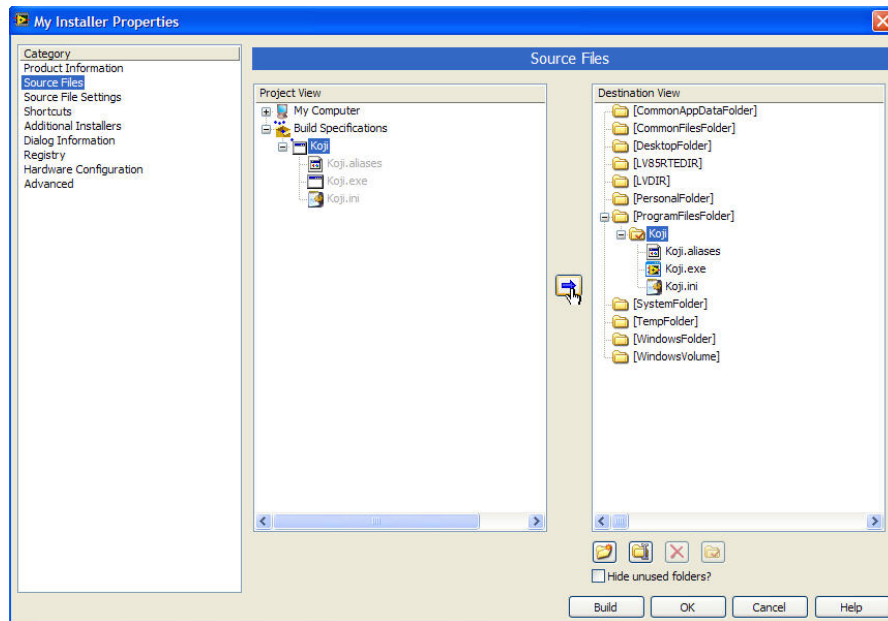


ภาพผนวกที่ ก9 กำหนดค่า Product Information

8. เลือก Source File ดังภาพผนวก ก10 และ ก11

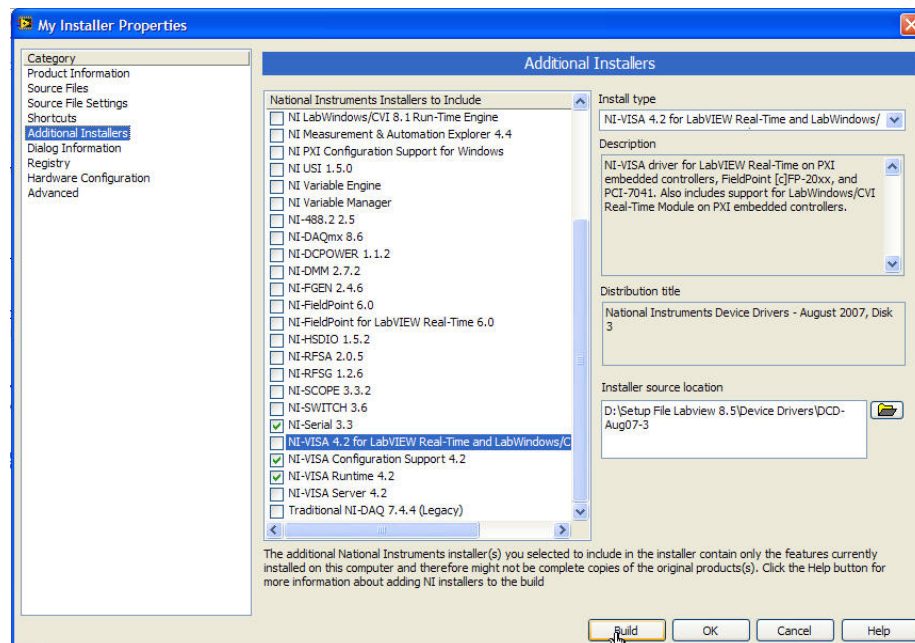


ภาพผนวกที่ ก10 เลือก source file



ภาพผนวกที่ 11 คลิกที่ปุ่มลูกศรเพื่อเลือก file

9. ทำการเลือก Additional Installers โดยคลิกเลือก module ที่ใช้งานกับ Application โดยในงานวิจัยนี้ใช้ NI-VISA runtime, NI-seiral 3.3 และ NI-VISA Server 4.2



ภาพผนวกที่ 12 คลิกเลือก Additional Installers

10. คลิก Build โปรแกรม LabVIEW จะทำการสร้าง Setup File ให้และเก็บไว้ที่ Destination Folder ที่ทำการกำหนดไว้ใน Product Information

ภาคผนวก ข

โปรโตคอล Modbus RTU ที่ใช้ในงานวิจัย

โปรโตคอล Modbus RUT

Modbus serial line protocol คือ protocol ที่ใช้สำหรับสื่อสารแบบ Master-Slaves protocol โดยที่จะมี Master 1 ตัวในระบบสายสัญญาณและที่เหลือเป็น Slaves

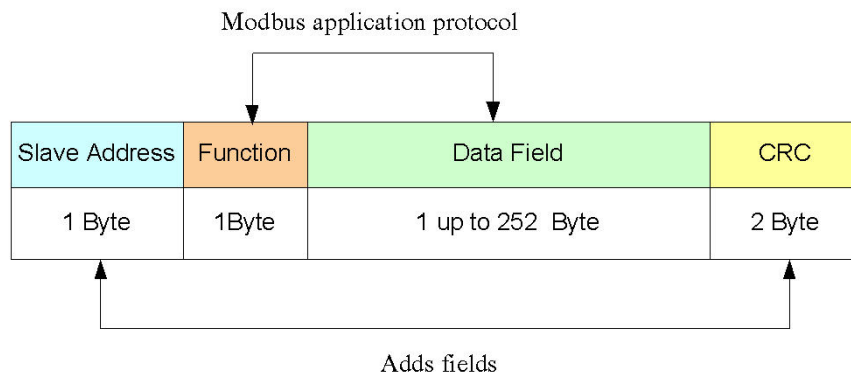
การสื่อสารทำได้ใน 2 Mode คือ

1. Unicast Mode คือการสื่อสารที่ Master ร้องขอไปยัง Slave ที่มี Address หนึ่งๆโดยตรง เช่น Slave address 1 และ Slave address 1 เท่านั้นที่จะทำตามความต้องการของ Master และทำการตอบกลับไปยัง Master
2. Broadcast Mode คือการสื่อสารที่ Master จะทำการร้องขอไปยัง Slave ทุกตัวใน line

รูปแบบของ Modbus Frame

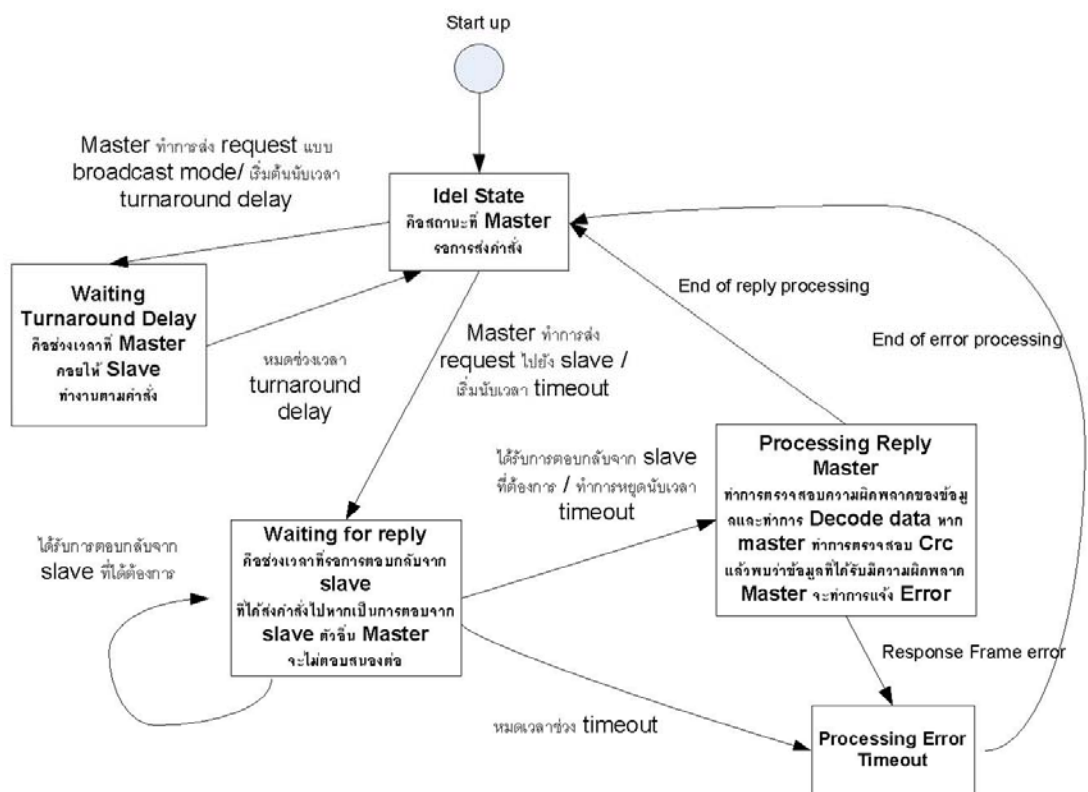
Modbus frame จะต้องมีความยาวไม่เกิน 256 byte และมีส่วนประกอบด้วย 2 ส่วนคือ

1. Modbus application protocol คือส่วนที่เป็น Function และ Data ที่ Master ต้องการ (request) หรือ Slave ตอบกลับ (reply) มายัง Master
2. Adds fields คือ Address field และ CRC ของ Modbus frame
 - 2.1 Address field คือ Slave address ของตัวอุปกรณ์ โดย address 0 เป็น address สำหรับการสื่อสารแบบ broadcast และ 1-247 คือ address ที่ slave สามารถใช้งานได้
 - 2.2 Function code คือ ส่วนที่ Master ระบุว่าต้องการให้ Slave ทำอะไรเช่น
 - 2.2.1 function 0x04 : Multi read input register
 - 2.2.2 function 0x06 : Preset single register
 - 2.2.3 function 0x08 : loop back diagnostic
 - 2.2.4 function 0x10 : Preset multi register
 - 2.3 Data Field คือส่วนที่เป็น request หรือ response parameters
 - 2.4 Error checking field คือ ส่วนที่ใช้ในการตรวจสอบความผิดพลาดจะข้อมูลที่รับได้

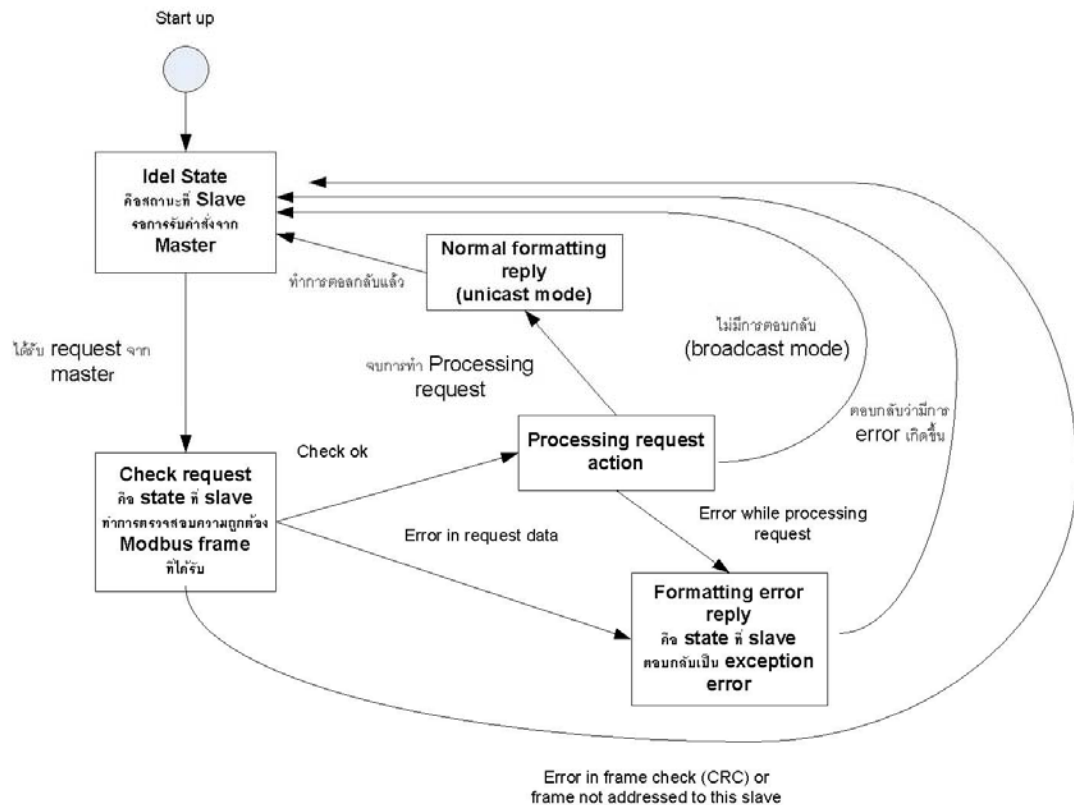


ภาพผนวกที่ ข1 Modbus RTU Frame

ขั้นตอนในการส่งข้อมูลสื่อสารสำหรับอุปกรณ์ Master และ Slave แสดงให้เห็นดัง ภาพผนวกที่ ข2 และ ภาพผนวกที่ ข3



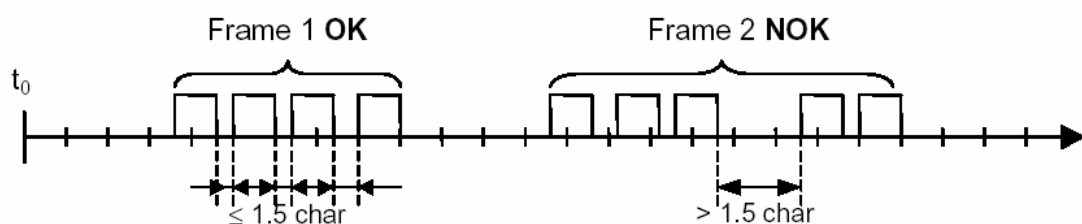
ภาพผนวกที่ ข2 การทำงานในการส่งของ Master



ภาพผนวกที่ ข3 การทำงานในการส่งของ Slave

Modbus Message RTU Framing

Modbus Message ประกอบด้วยส่วนที่เป็น message, Start และ End โดยที่ Start และ End คือ ช่วงเวลาที่ไม่มีการส่งข้อมูลมาเป็นเวลา 3.5 character โดยที่การส่งข้อมูลระหว่าง character ต้องไม่ห่างกันเกิน 1.5 characters โดยแสดงตัวอย่างดังภาพผนวกที่ ข4 โดย Message Frame ซ้ายมือเป็น Frame ที่ถูกต้อง



ภาพผนวกที่ ข4 แสดงตัวอย่าง Modbus Message RTU Framing

การคำนวณ 16 bits CRC Checking

1. Load ค่า 0xFFFF ลงใน 16 bit register(CRC register)
2. ทำการ XOR Byte แรกของข้อมูลทีอ่านมาด้วย Low Byte ของ CRC register แล้วเก็บผลใน CRC register
3. Shift 1 bit ของ CRC register ออกทางขวา แล้วแทน MSB ด้วย 0
4. ถ้า bit ที่ Shift ออกมาตามข้อ 3 มีค่าเป็น 1 ให้ทำการ XOR CRC register ด้วย 0xA001
5. ทำซ้ำในขั้นตอนที่ 3 และ 4 จนครบ 8 ครั้งเท่ากับจบการทดสอบ 1 Byte
6. ทำซ้ำในขั้นตอนที่ 2 ถึง 5 โดยใช้ 8 bit byte ต่อไปของข้อมูลจนครบ
7. ทำการสลับ Low กับ High Byte ของ CRC register

Example C-Code

```

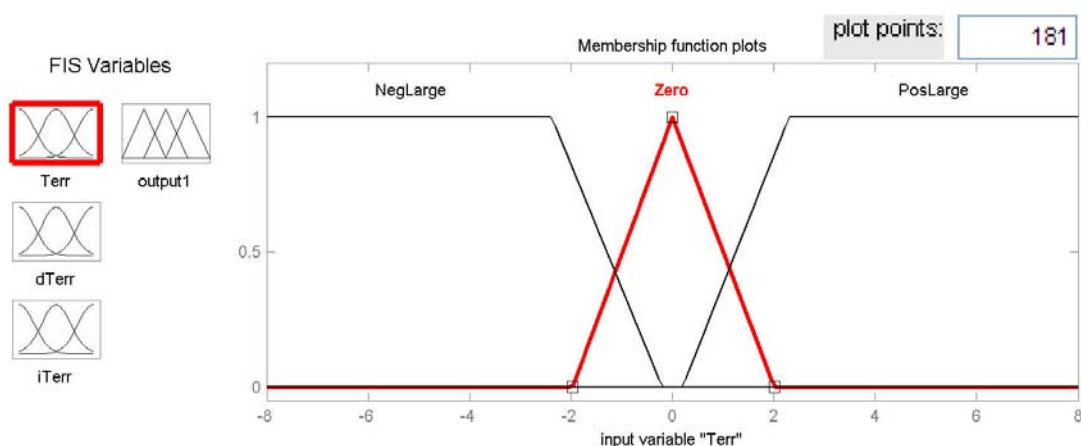
unsigned int CHECK_CRC(unsigned char *buff,unsigned char start,unsigned char len)
{
    //          Function CRC FOR RTU PROTOCOL          //
    unsigned char byte_count,bit_count;
    unsigned int  crc_reg;
    unsigned int  CRCHi,CRCLo;
    crc_reg=0xFFFF;
    for(byte_count=start;byte_count<(len+start);byte_count++)
    {
        crc_reg=crc_reg^(*(buff+byte_count));
        for(bit_count=0;bit_count<8;bit_count++)
        {
            if((crc_reg & 0x0001))
            {
                crc_reg=crc_reg>>1;    crc_reg=crc_reg^0xA001;
            }
            else {
                crc_reg=crc_reg>>1;
            }
        }
        CRCLo=crc_reg>>8;    CRCHi=crc_reg<<8;
        crc_reg=CRCLo+CRCHi;    return(crc_reg);
    }
}

```

ภาคผนวก ค
การปรับจูน Fuzzy PID

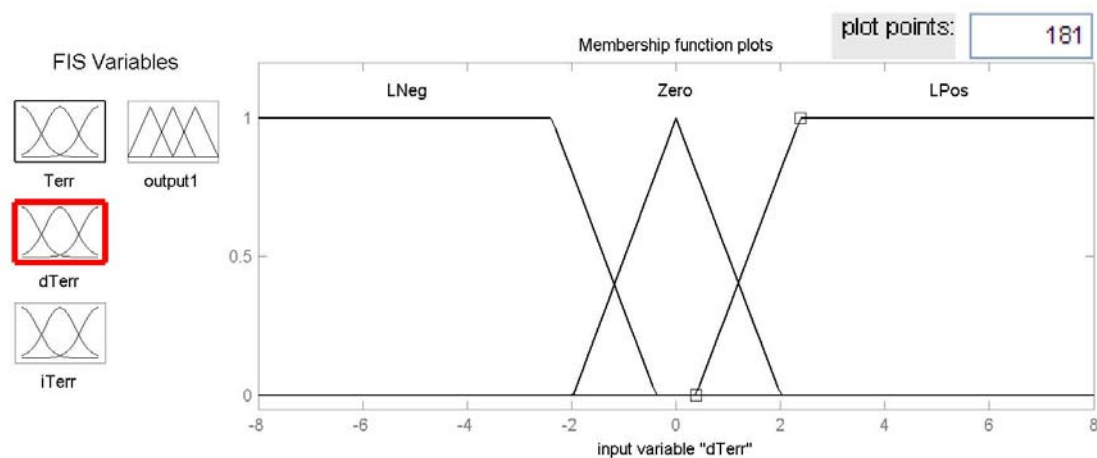
การปรับจูน Fuzzy PID

เลือก error, derror และ ierror เป็น input Fuzzy และมี 1 output Fuzzy โดยทำการออกแบบ Membership function ของ error ประกอบด้วย 3 เทอมคือ NegLarge, Zero และ PosLarge ดังภาพผนวกที่ ค1



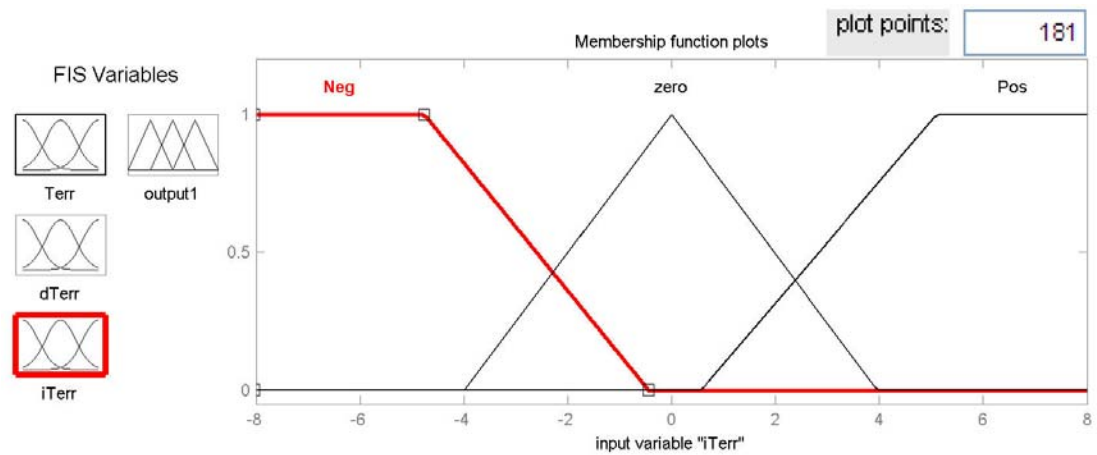
ภาพผนวกที่ ค1 แสดงค่า Membership function ของ input error

Input derror ประกอบด้วย 3 เทอมคือ NegLarge, Zero และ PosLarge ดังภาพผนวกที่ ค2



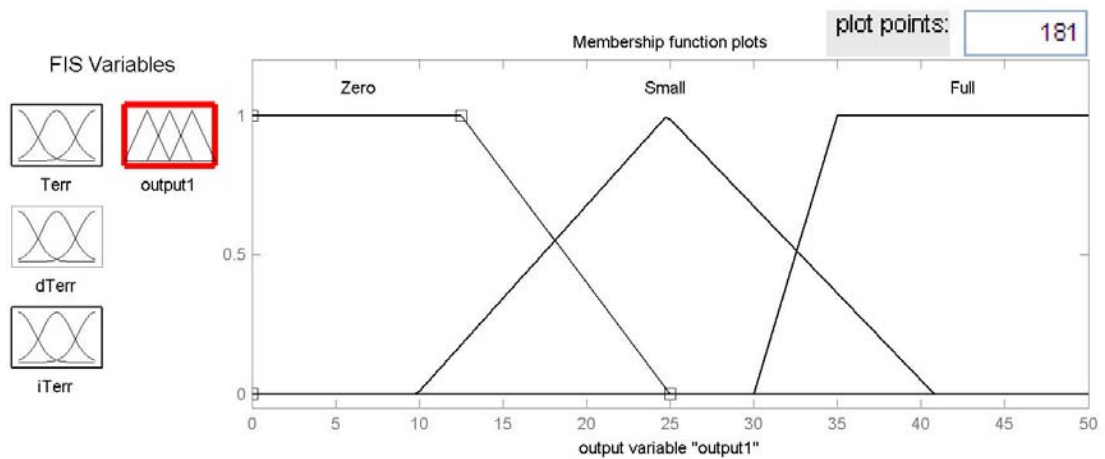
ภาพผนวกที่ ค2 แสดงค่า Membership function ของ input derror

Input ierror ประกอบด้วย 3 เทอมคือ NegLarge, Zero และ PosLarge ดังภาพผนวกที่ ค3



ภาพผนวกที่ ค3 แสดงค่า Membership function ของ input ierror

Output Fuzzy ประกอบด้วย 3 เทอมคือ Zero, Small และ Full ดังภาพผนวกที่ ค4



ภาพผนวกที่ ค4 แสดงค่า Membership function ของ Output Fuzzy

ทำการออกแบบ Fuzzy Rule โดยในที่นี้เลือกใช้ Fuzzy Rule ดังภาพผนวกที่ ค5, ค6 และ ค7

$\int edt$

de				
Input error (e)	Output(u)			

ภาพผนวกที่ ค5 Fuzzy Rule สำหรับ errorและ derror ขณะที่ ierror เป็น Positive large

$\int edt$

de				
Input error (e)	Output(u)			

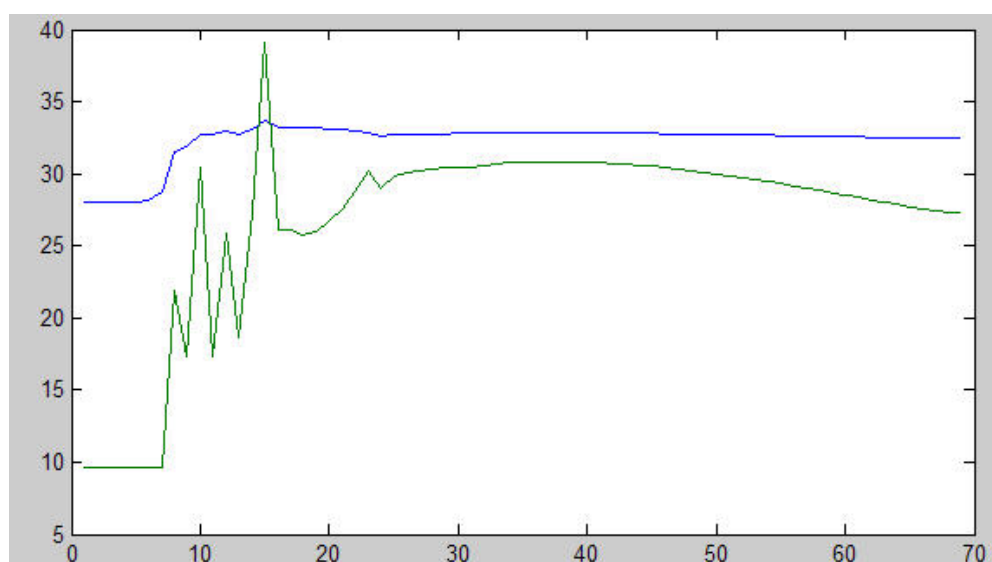
ภาพผนวกที่ ค6 Fuzzy Rule สำหรับ errorและ derror ขณะที่ ierror เป็น Zero

$$\int e dt$$

de				
Input error (e)	Output(u)			

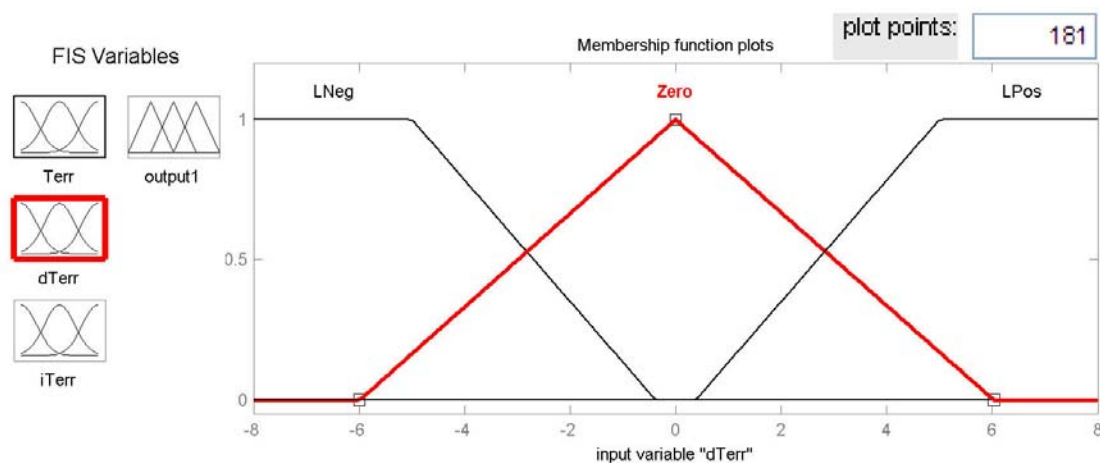
ภาพผนวกที่ ค7 Fuzzy Rule สำหรับ error และ derror ขณะที่ error เป็น Negative large

ผลทดลองที่ได้แสดงดังภาพผนวกที่ ค8 โดยที่เส้นสีน้ำเงินคือค่าอุณหภูมิของวัสดุหมัก และเส้นสีเขียวเป็นค่า output Fuzzy PID ที่จ่ายให้กับ Koji Process แกน x ของ กราฟคือเวลามีหน่วยเป็น ชม. แกน y ของกราฟมีหน่วยเป็น C สำหรับเส้นสีน้ำเงิน และมีหน่วยเป็น Hz สำหรับเส้นสีเขียว

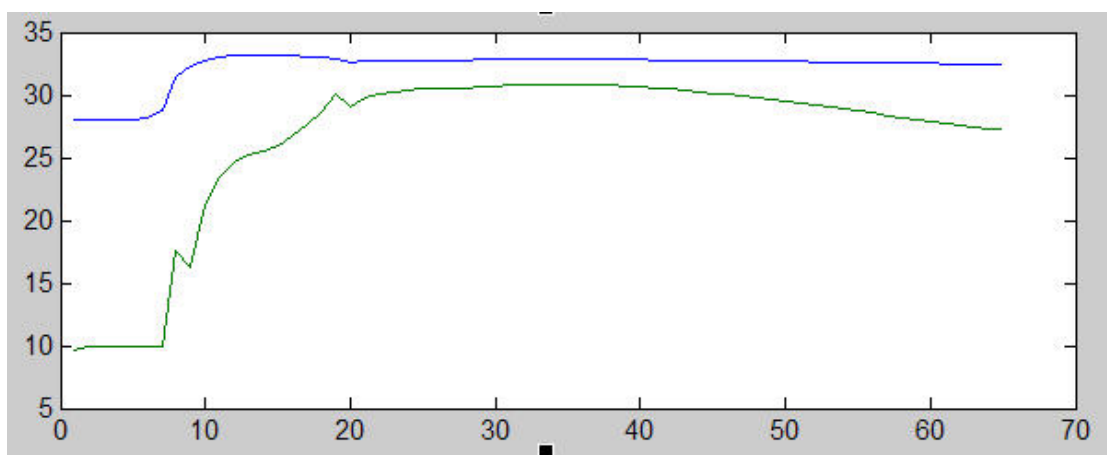


ภาพผนวกที่ ค8 ผลการ Simulation

1. ทำการปรับจูน Membership function ของ input derror โดยไม่มีการเปลี่ยนแปลง Fuzzy Rule โดยทำการขยายขนาดของ Zero และปรับส่วนของกางหมูให้ยืดยาวขึ้นดังภาพผนวกที่ ค9



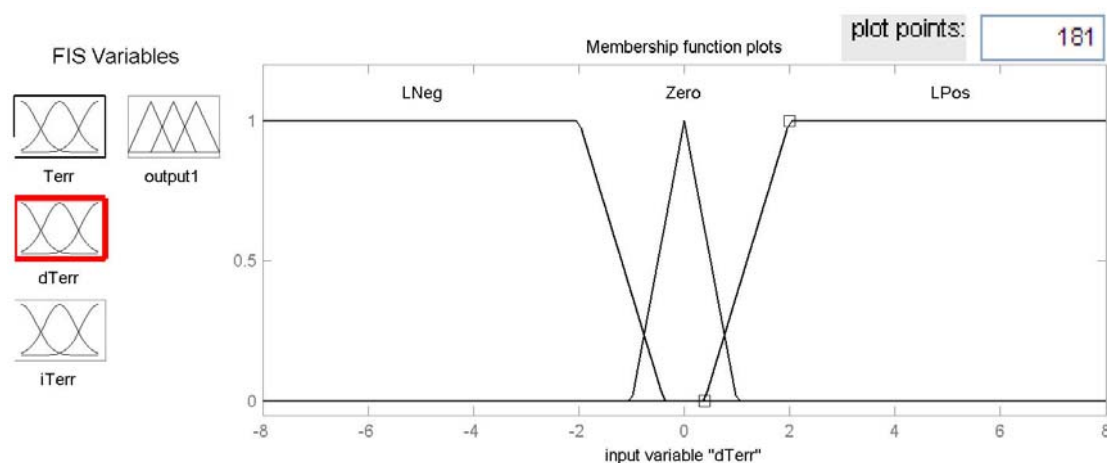
ภาพผนวกที่ ค9 Member ship function ของ derror ที่ถูกปรับจูน



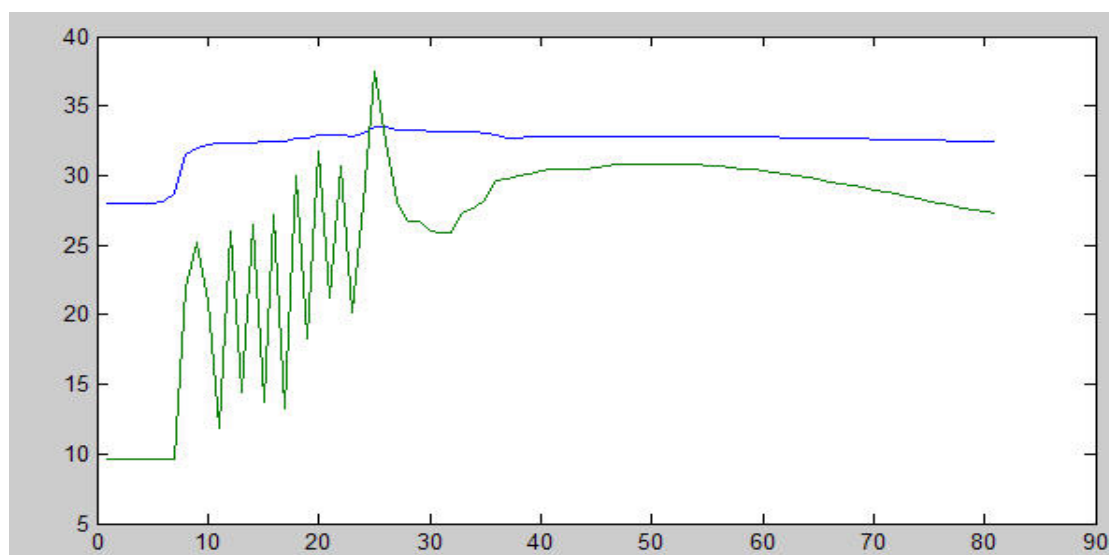
ภาพผนวกที่ ค10 ผลการทดลองเมื่อปรับจูน Zero และปรับส่วนของกางหมูให้ยืดยาวขึ้น

ผลการทดลองแสดงให้เห็นว่าค่า Output Fuzzy มีค่าแกว่งน้อยลงซึ่งเป็นผลจากการปรับจูนค่า Zero ให้กว้างขึ้น

2. ทำการปรับจูน Membership function ของ input derror โดยไม่มีการเปลี่ยนแปลง Fuzzy Rule โดยทำการลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้นดังภาพผนวกที่ ค11



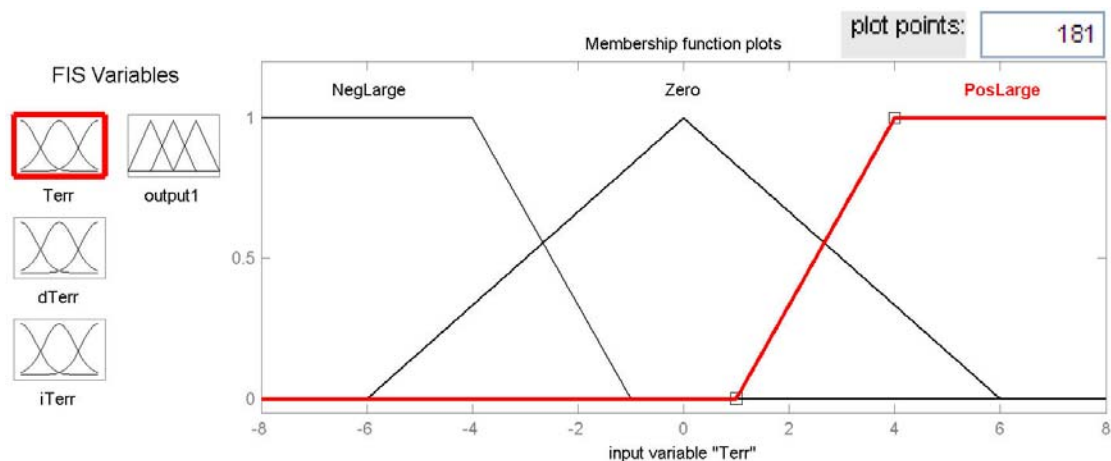
ภาพผนวกที่ ค11 Member ship function ของ derror ที่ถูกลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้น



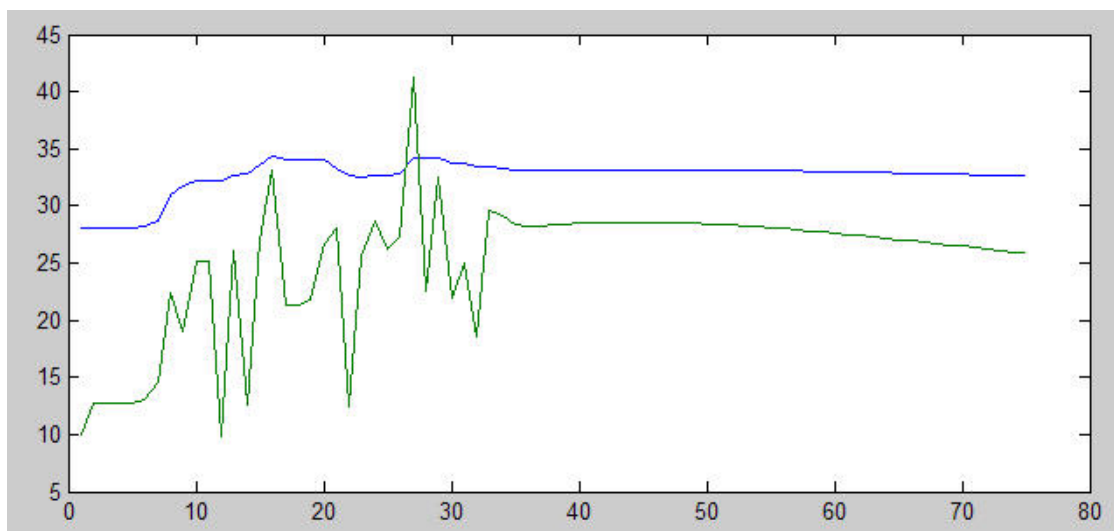
ภาพผนวกที่ ค12 ผลการทดลองเมื่อปรับจูนลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้น

ผลการทดลองแสดงให้เห็นว่าค่า Output Fuzzy มีค่าแกว่งมากขึ้นแต่ค่า Over shoot มีค่าลดลง ซึ่งเป็นผลจากการปรับจูนลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้น

3. ทำการปรับจูน Membership function ของ input error โดยไม่มีการเปลี่ยนแปลง Fuzzy Rule และค่า Membership function ของ derro เปลี่ยนเป็นค่าเริ่มต้นทำการปรับจูนโดยทำการขยายขนาดของ Zero และปรับส่วนของคางหมูให้ยืดยาวขึ้นดังภาพผนวกที่ ค13



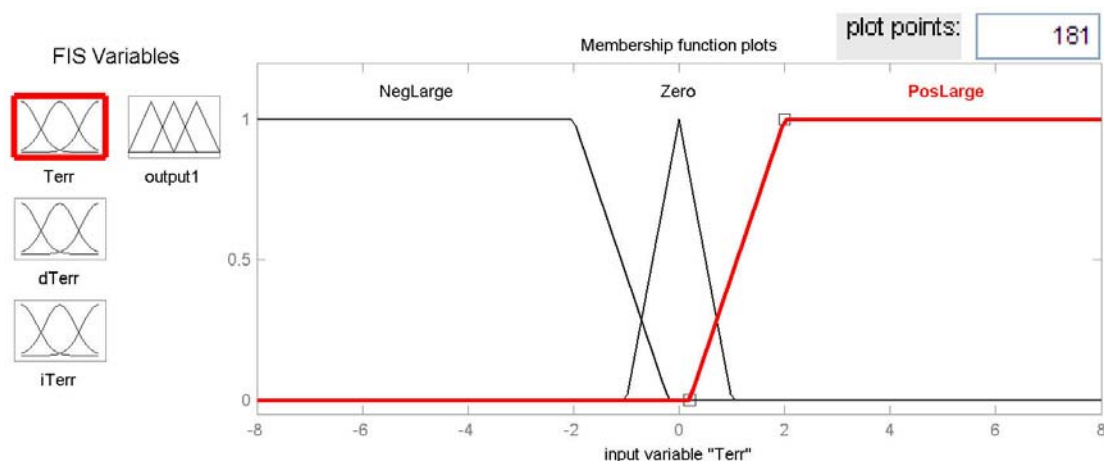
ภาพผนวกที่ ค13 Member ship function ของ error ที่ถูกปรับจูนขยายขนาดของ Zero และปรับส่วนของคางหมูให้ยืดยาวขึ้น



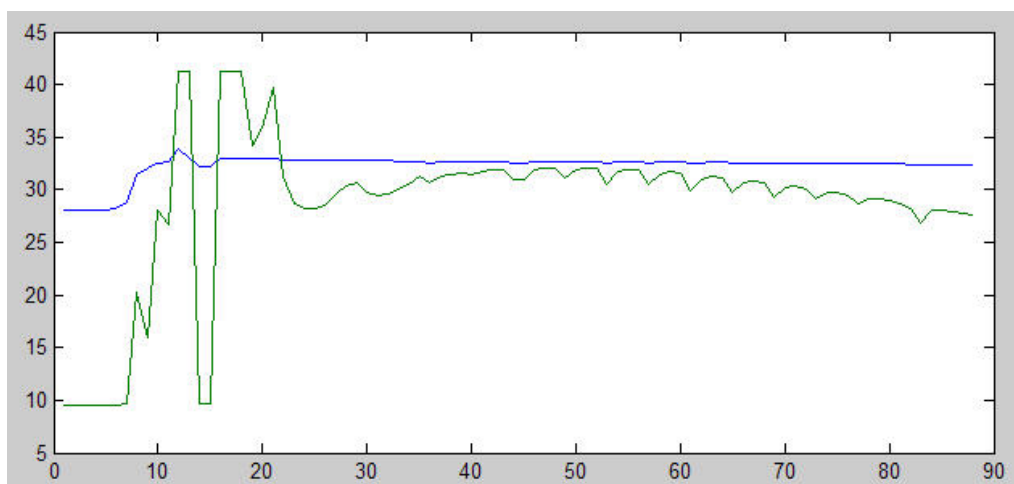
ภาพผนวกที่ ค14 ผลการทดลองเมื่อปรับจูนขยายขนาดของ Zero และปรับส่วนของคางหมูให้ยืดยาวขึ้น

ผลการทดลองแสดงให้เห็นว่าค่า Output Fuzzy มีค่าแกว่งมากขึ้นและค่า stady state error มีค่ามากขึ้นซึ่งเป็นผลจากการปรับจูนค่า Zero ให้กว้างขึ้น

4. ทำการปรับจูน Membership function ของ input error โดยไม่มีการเปลี่ยนแปลง Fuzzy Rule และค่า Membership function ของ derro เปลี่ยนเป็นค่าเริ่มต้นทำการปรับจูนโดยทำการลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้นดังภาพผนวกที่ ค15



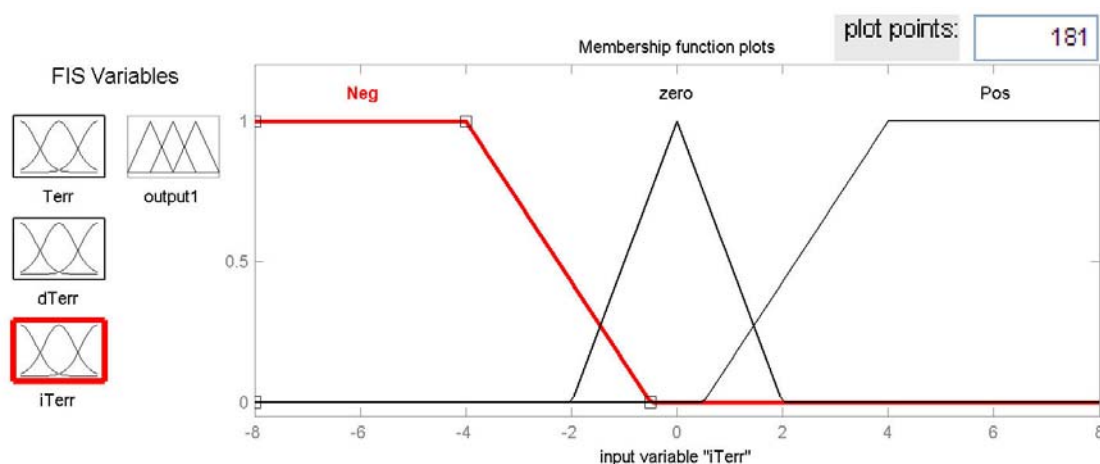
ภาพผนวกที่ ค15 Member ship function ของ error ที่ถูกปรับจูนลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้น



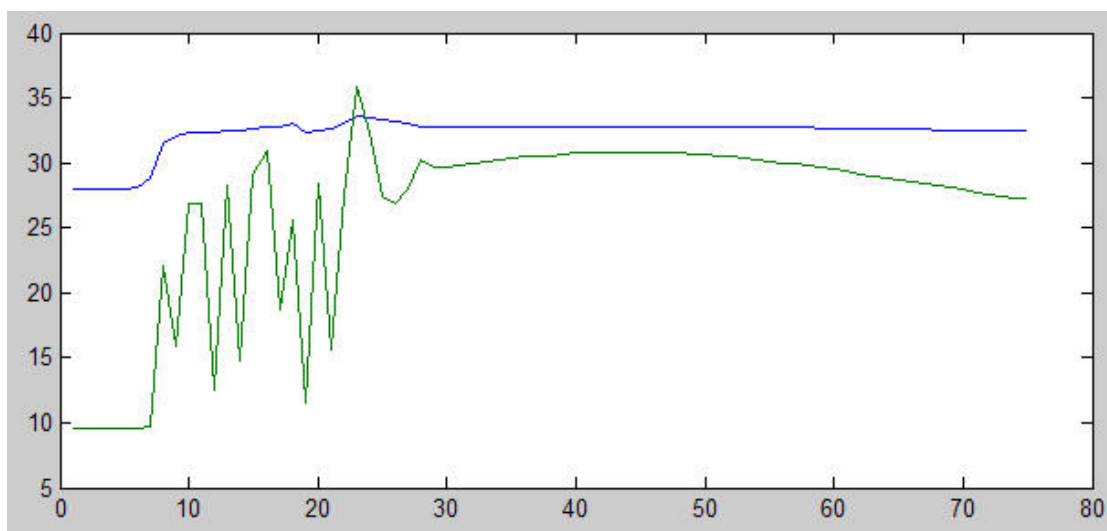
ภาพผนวกที่ ค16 ผลการทดลองเมื่อปรับจูนลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้น

ผลการทดลองแสดงให้เห็นว่าค่า Output Fuzzy มีค่าแกว่งมากขึ้นแต่ค่า steady state error มีค่าลดลงซึ่งเป็นผลจากการปรับจูนค่า Zero ให้กว้างขึ้น

5. ทำการปรับจูน Membership function ของ input ierror โดยไม่มีการเปลี่ยนแปลง Fuzzy Rule และค่า Membership function ของ error และ derro เปลี่ยนเป็นค่าเริ่มต้นทำการปรับจูน โดยทำการลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้นดังภาพผนวกที่ ค17



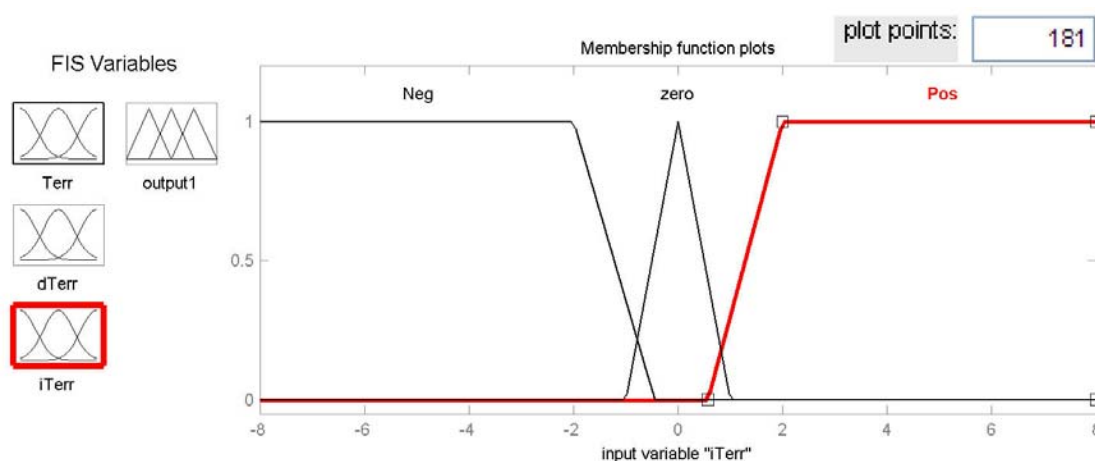
ภาพผนวกที่ ค17 Member ship function ของ ierror ที่ถูกปรับจูนลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้น



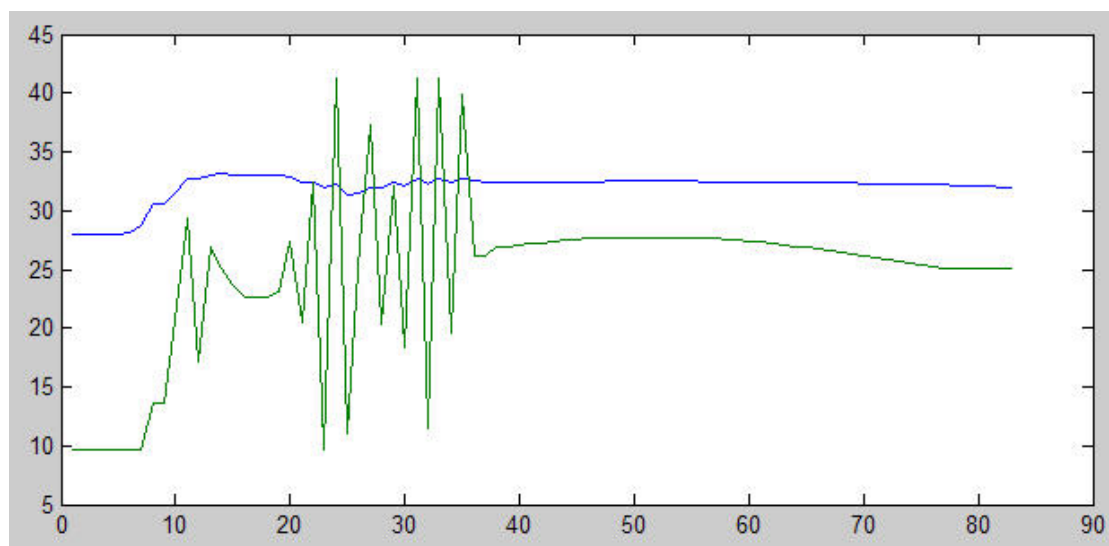
ภาพผนวกที่ ค18 ผลการทดลองเมื่อปรับจูนลดขนาดของ Zero และปรับส่วนของคางหมูให้แคบขึ้น

ผลการทดลองแสดงให้เห็นว่าค่าผลที่ได้ไม่มีความแตกต่างจากก่อนปรับมากนักโดยมีค่า Output Fuzzy เท่านั้นที่มีการเปลี่ยนแปลงมากที่สุดคือมีค่าแกว่งมากขึ้น

6. ทำการปรับจูน Membership function ของ input ierror โดยไม่มีการเปลี่ยนแปลง Fuzzy Rule และค่า Membership function ของ error และ derro เปลี่ยนเป็นค่าเริ่มต้นทำการปรับจูน โดยทำการลดขนาดของ Zero ให้มีเล็กกว่าเดิมและปรับส่วนของกางหมูให้แคบขึ้น ดังภาพผนวกที่ ค19



ภาพผนวกที่ ค19 Member ship function ของ ierror ที่ถูกปรับจูนลดขนาดของ Zero ให้มีเล็กกว่าเดิมและปรับส่วนของกางหมูให้แคบขึ้น



ภาพผนวกที่ ค20 ผลการทดลองเมื่อปรับจูนลดขนาดของ Zero ให้มีเล็กกว่าเดิมและปรับส่วนของกางหมูให้แคบขึ้น

ผลการทดลองเมื่อเปรียบเทียบภาพผนวกที่ 20 กับ ภาพผนวกที่ 18 จะเห็นได้ว่าผลการทดลอง ไม่แตกต่างกันมากนักเนื่องจาก Fuzzy Rule ที่ใช้งาน โดยในช่วงที่ ierror มีค่าเป็น Positive error, Zero และ Negative error ของ Rule ให้ผลการตัดสินใจ Output แตกต่างกันน้อยมากคือมี แตกต่างกันเพียงกฎ เดียวดังนั้นการปรับมูลค่า Membership function ของ ierror จึงไม่มีผลมาก นัก

ประวัติการศึกษา และการทำงาน

ชื่อ –นามสกุล	นายนคร ตริสิงหวงศ์
วัน เดือน ปี ที่เกิด	15 กรกฎาคม 2523
สถานที่เกิด	จ. ฉะเชิงเทรา อ. เมือง ต. หน้าเมือง
ประวัติการศึกษา	วศ.บ. (ไฟฟ้า) มหาลัยเกษตรศาสตร์ (พ.ศ. 2546)
ตำแหน่งหน้าที่การงานปัจจุบัน	วิศวกรออกแบบและวิจัยผลิตภัณฑ์
สถานที่ทำงานปัจจุบัน	บ. ไพรมัส จำกัด
ผลงานดีเด่นและรางวัลทางวิชาการ	
ทุนการศึกษาที่ได้รับ	ทุนอุดหนุนวิจัย มก.
ผลงานตีพิมพ์ทางวิชาการ	- นคร ตริสิงหวงศ์, พีระยศ แสนโกชณ์, เพ็ญจิตร ศรีนพ , และ ญัฐวุฒิ ชินธเนศ. 2551. Temperature Control of Koji Process by using Fuzzy PD Controller. น. 669-672. ใน งาน ECTI-CON 2008