

ดี  
วอ  
ออง  
ง

ภาคผนวก

ดร. อธิษฐ์

ภาคผนวก ก  
โปรแกรม LabVIEW

## โปรแกรม LabVIEW [33]

### 1. พาเนล

พาเนลเป็นหน้าต่างที่ใช้แสดงการควบคุมการทำงานโปรแกรม ผู้ใช้สามารถสร้างรูปแบบขึ้นเองได้อย่างรวดเร็วเนื่องจาก LabVIEW มีส่วนประกอบต่าง ๆ ที่ใช้สำหรับออกแบบหน้าต่างพาเนลก็จะมีส่วนประกอบหลัก ๆ ด้วยกัน 3 ส่วนคือ พื้นที่สำหรับเขียนโปรแกรม เมนูบาร์ และทูลบาร์ สำหรับหน้าต่างพาเนล จะแสดงตัวควบคุมการทำงาน (Controls) และตัวแสดงผล (Indicator) ส่วนลำดับการทำงานของโปรแกรมนั้นจะแสดงในส่วนของหน้าต่างไดอะแกรม โดยในโปรแกรม LabVIEW จะถือว่าตัวควบคุมเป็นอุปกรณ์ “อินพุต” และตัวแสดงผลเป็นอุปกรณ์ “เอาต์พุต” การวางตัวควบคุมและตัวแสดงผลลงในหน้าต่างพาเนลทำได้โดยคลิกเมาส์ทางด้านขวาบนหน้าต่างพาเนล จะปรากฏลือกทูลพาเลทขึ้นมาซึ่งภายในทูลพาเลทจะประกอบด้วยตัวควบคุมและตัวแสดงผลแบบต่าง ๆ การวางอุปกรณ์ทำได้โดยการใช้เมาส์คลิกค้างไว้ที่อุปกรณ์ที่ต้องการจากนั้นลากไปวางในหน้าต่างพาเนลก็จะได้ภาพดังภาพที่ 1

ภาพที่ 1 ลักษณะของหน้าต่างพาเนล [33]

### - Numerical controls and indicator

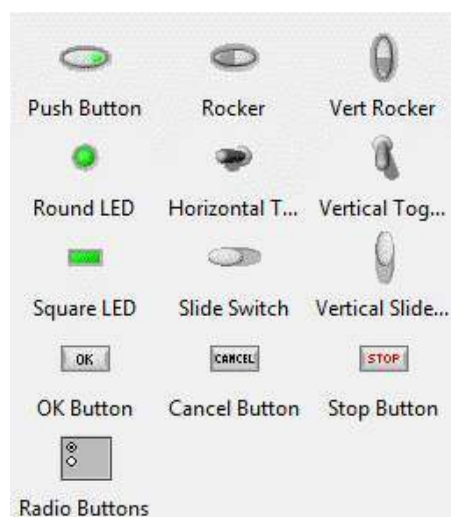
สำหรับตัวควบคุมและตัวแสดงผลที่มีลักษณะเป็นตัวเลขนั้นสามารถเรียกใช้ในส่วน ของ หน้าต่างพร้อมท์พาเนล โดยเลือกจาก Control » Modern » Numeric บน Control พาเนล



ภาพที่ 2 Numerical Control and Indicator [33]

### - Boolean control and indicator

สำหรับตัวควบคุมและตัวแสดงผลที่มีลักษณะบูลีนนั้นจะแสดงการทำงานในลักษณะของ ความถูกหรือผิด (True /False) สามารถเรียกใช้ในส่วน ของ All function หรือ Button and LED บนทูลพาเลท



ภาพที่ 3 Boolean Control and Indicator [33]

### - การกำหนดอุปกรณ์

โดยปกติแล้วอุปกรณ์หนึ่งตัวก็จะทำงานหรือมีสถานะได้หนึ่งอย่าง ไม่ว่าจะ มีสถานะ เป็นค่าคงที่ ตัวควบคุมหรือตัวแสดงผล ซึ่งเราสามารถกำหนดได้ว่าจะให้อุปกรณ์ตัวนั้นแสดง สถานะเป็นแบบใด การกำหนดทำได้โดยการคลิกขวาที่อุปกรณ์แล้วเลือกคำสั่ง Create ซึ่งคำสั่ง Create นี้จะมีคำสั่งย่อยอยู่ 3 คำสั่งคือ Constant, Control และ Indicator ซึ่งเราสามารถเลือกได้ว่าจะกำหนดสถานะของอุปกรณ์นั้นได้

ภาพที่ 4 การกำหนดอุปกรณ์ [33]

### - ไตอะแกรม

ในส่วนของหน้าต่าง Block diagram จะเป็นส่วนที่ใช้ในการเขียนโปรแกรมและแสดงการทำงานของโปรแกรมทั้งหมด แสดงการเชื่อมโยงระหว่างบล็อกคำสั่งแต่ละบล็อกเข้าด้วยกัน โดยการเขียนโปรแกรมด้วยภาพลักษณะนี้ในโปรแกรม LabVIEW เรียกว่าการเขียนโปรแกรมด้วยภาษากราฟฟิก (G-Languages) การเขียนโปรแกรมด้วยภาพนั้นจะอาศัยหลักการทำงานของเครื่องมือวัดหรือการวัดคุม ทำให้ผู้ใช้สามารถออกแบบรูปแบบโปรแกรมตามต้องการ

ภาพที่ 5 หน้าต่างของไดอะแกรม [33]

## 2. พาเลท

พาเลทเป็นหน้าต่างย่อยที่มีส่วนประกอบของอ็อบเจ็กต์ที่ใช้สำหรับการเขียน VI การนำไปใช้งานนั้นทำได้ง่ายเพียงแค่คลิกเมาส์ทางด้านขวาบริเวณที่ว่างของหน้าต่างพาเนลหรือไดอะแกรม ซึ่งจะปรากฏพาเลทขึ้นมาจากนั้นใช้เมาส์คลิกที่อ็อบเจ็กต์ที่ต้องการแล้วลากไปวางบนหน้าต่าง พาเนลหรือหน้าต่างไดอะแกรมพาเลทมีองค์ประกอบหลัก ๆ อยู่ 3 ส่วนคือ Tools, Control และ Function

### - Tools Palette

ทูลพาเลทจะเป็นตัวบอกหรือกำหนดสถานะของเมาส์ว่าทำงานอยู่ในโหมดใด การเรียกใช้พาเลทนี้ทำได้โดยการเลือกที่เมนู Window แล้วเลือกคำสั่งย่อย Show tools palette แล้วจะปรากฏ Tools palette ขึ้นมาดังภาพ การใช้งานนั้นเพียงแค่เราใช้เมาส์ไปคลิกที่สัญลักษณ์ใน Tools palette เคอร์เซอร์ของเมาส์ก็จะเปลี่ยนไปตามลักษณะที่เราเลือก

ภาพที่ 6 Tool Palette [33]

### - Control palette

มีส่วนประกอบของอุปกรณ์ที่เป็นตัวควบคุมและตัวแสดงผลที่ใช้สำหรับสร้าง VI โดยภายใน Control palette ก็จะมีพาเลทย่อย ๆ อีกเพราะว่าอุปกรณ์ใน Control palette ก็จะมีอุปกรณ์ย่อยอีกแต่อย่างไรก็ตามเราสามารถเรียกดูอุปกรณ์ทั้งหมดที่อยู่ใน Control palette ได้โดยคลิกเมาส์เลือกที่ All Control ซึ่งจะรวมเอาอุปกรณ์ทั้งหมดไว้ในนี้ นอกจากนี้แล้วเรายังสามารถเปิดใช้งาน Control palette ได้อีกทางหนึ่งคือเลือกที่เมนู Window แล้วเลือก Show control palette ก็สามารเลือกใช้ Control palette ได้เช่นกัน

ภาพที่ 7 Control Palette [33]

### - Function Palette

การใช้งานโดยทั่วไปจะมีความคล้ายกันกับ Control palette ซึ่งภายในประกอบไปด้วยอุปกรณ์ในการสร้าง VI ตัวอย่างอุปกรณ์หลัก ๆ ใน Function palette เช่น For loops, While loops และ Formula nodes สามารถเรียกใช้งานได้ 2 วิธีเช่นเดียวกันกับการเรียกใช้งาน Control palette คือ เลือกที่เมนู Window แล้วเลือก Show function palette หรือ คลิกเมาส์ทางด้านขวาบริเวณพื้นที่ว่างในหน้าต่างโคอะแกรม

ภาพที่ 8 Function palette [33]

### 3. เครื่องมือ (Tools)

- เครื่องมือของหน้าต่างพาเนล

ภาพที่ 9 ทูลบาร์ในหน้าต่างพาเนล [33]

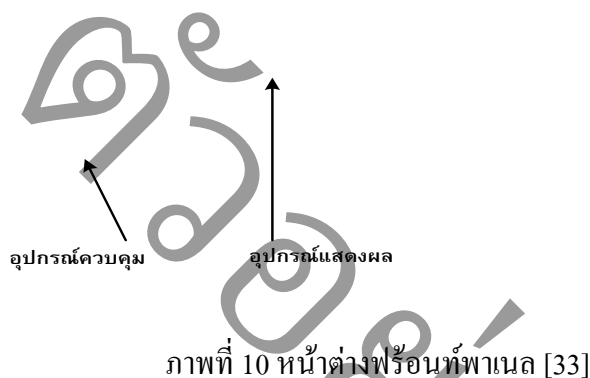
RUN	ใช้สำหรับรันโปรแกรม
RUN continuously	ใช้สำหรับรันโปรแกรมอย่างต่อเนื่อง
Abort execution	หรือเรียกอีกอย่างหนึ่งว่าปุ่ม Stop ใช้สำหรับหยุดการรันโปรแกรม
Pause	ใช้สำหรับหยุดการรันโปรแกรมชั่วคราวเพื่อตรวจสอบการทำงานของโปรแกรมเมื่อต้องการรันโปรแกรมต่อให้กดปุ่มนี้ซ้ำอีกครั้งโปรแกรมก็จะรันต่อไป



Highlight execution	แสดงลำดับขั้นการทำงานของโปรแกรมโดยแสดงให้เห็นทิศทางการไหลของสัญญาณ
Start single step	ใช้สำหรับสั่งให้โปรแกรมทำงานทีละคำสั่ง
Text setting	ใช้สำหรับจัดการกับตัวอักษรทั้งหมดไม่ว่าจะเป็น ฟรอนท์ ขนาดรูปแบบตัวอักษร รวมไปถึงการกำหนดสีให้กับตัวอักษรด้วย
Align object	ใช้จัด Object ต่าง ๆ ที่เราใช้เขียนโปรแกรมให้อยู่ในแนวเดียวกันทั้งแนวตั้งและแนวนอนเพื่อความเป็นระเบียบ
Distribute object	ใช้สำหรับการกำหนดระยะห่างระหว่าง Objects อัตโนมัติ
Reorder	ใช้สำหรับกำหนดลำดับก่อนหลังสำหรับการวางซ้อนทับกันของ Object
Object context help	ใช้แสดงคุณสมบัติหรือรายละเอียดของ Object แต่ละตัว

#### 4. ฟรอนท์พาเนล

หน้าต่างฟรอนท์พาเนลสำหรับเขียนโปรแกรม LabVIEW นั้นจะเป็นหน้าต่างที่ใช้ติดต่อกับผู้ใช้งานซึ่งในหน้าต่างพาเนลจะประกอบไปด้วยอุปกรณ์อยู่ 2 ส่วนคืออุปกรณ์สำหรับการควบคุม (Control) ยกตัวอย่างเช่น Knobs, Switches และอุปกรณ์อีกอย่างหนึ่งก็คืออุปกรณ์ สำหรับแสดงผล (Indicator) เช่น LED, Text setting, Graph และ Chart เป็นต้น ส่วนอุปกรณ์ที่ไม่ใช่อุปกรณ์ควบคุมและแสดงผลจะแสดงไอคอนอยู่ในหน้าต่างของบล็อกไดอะแกรม สำหรับอุปกรณ์ควบคุมในภาษากราฟฟิกจะหมายถึง “อินพุต” และอุปกรณ์แสดงผลจะหมายถึง “เอาต์พุต” การวางอุปกรณ์ควบคุมและอุปกรณ์แสดงผลลงไปในหน้าต่างฟรอนท์พาเนลทำได้ โดยการเลือกอุปกรณ์ที่ต้องการบนหน้าต่าง Control palette แล้วใช้เมาส์ลากไปวางบนหน้าต่างพาเนลซึ่งการเรียกหน้าต่าง Control palette ทำได้โดยการคลิกเมาส์ทางด้านขวา บริเวณพื้นที่ว่างของหน้าต่างพาเนลก็จะปรากฏหน้าต่าง Control palette ที่ประกอบไปด้วยอุปกรณ์ควบคุมและอุปกรณ์แสดงผลขึ้นมา การเรียกใช้งานหน้าต่าง Control palette อีกวิธีหนึ่งก็คือเลือกที่เมนู Window ของแท็บเมนูบาร์ แล้วเลือก Show control palette จะปรากฏหน้าต่างของ Controls palette ขึ้นมาให้เลือกใช้งานเช่นเดียวกัน



## 5. บล็อกไดอะแกรม

ในหน้าต่างบล็อกไดอะแกรมจะประกอบไปด้วยอุปกรณ์ที่ใช้ในการสร้างโปรแกรมซึ่งถ้าเปรียบเทียบกับภาษาอื่น ๆ แล้วอุปกรณ์ที่วางลงไปบนหน้าต่างบล็อกไดอะแกรมจะถือว่าเป็น Source code การทำให้อุปกรณ์ต่าง ๆ ที่อยู่ในหน้าต่างของบล็อกไดอะแกรมทำงานสัมพันธ์กัน และเป็นไปตามที่ผู้เขียนต้องการนั้นสามารถทำได้ด้วยการลากสายสัญญาณจากจุดเชื่อมต่อของ อุปกรณ์แต่ละตัวให้ถึงกันและกำหนดค่าหรือคุณสมบัติของอุปกรณ์แต่ละตัวก็จะทำให้โปรแกรมทำงานตามที่ผู้เขียนต้องการ คุณสมบัติของอุปกรณ์ที่อยู่บนหน้าต่างบล็อกไดอะแกรมมีอยู่ 3 อย่างคือ Node, Terminal และ Wires

### - Nodes

เป็นตัวกระทำให้โปรแกรมทำงานตามที่ผู้เขียนต้องการถ้าเปรียบเทียบกับโปรแกรมทั่วไป Nodes ก็เปรียบเสมือน Statement, Functions, และ Subroutines โหนดในโปรแกรม LabVIEW มี 3 ชนิดคือ Function, Sub VI และ Structure โหนดแบบ Function จะสร้างขึ้นมาเพื่อให้สามารถทำงานขั้นพื้นฐานได้ เช่น การบวกเลข การคูณเลข เป็นต้น ดังเช่น การบวกและการคูณในภาพที่ 11 ก็เป็นฟังก์ชันโหนดแบบหนึ่งเช่นกัน

ภาพที่ 11 หน้าต่างบล็อกไดอะแกรม [33]

#### - Terminal

เป็นทางผ่านของข้อมูลระหว่างบล็อกไดอะแกรมกับพรีอนท์พาเนลและระหว่างโหนดแต่ละโหนดในบล็อกไดอะแกรมเทอร์มินอลเปรียบได้กับค่าพารามิเตอร์และค่าคงที่ในโปรแกรมภาษาอื่น เทอร์มินอลแต่ละชนิดก็มีความแตกต่างกันซึ่งในโปรแกรม LabVIEW มีเทอร์มินอลอยู่ 4 ชนิดคือ Control and Indicator, node, Constants, และเทอร์มินอลพิเศษ ซึ่งเทอร์มินอลแต่ละตัวสามารถลากสายสัญญาณเพื่อเป็นทางผ่านของข้อมูลสำหรับตัวอย่างในกรณีของ Control and indicator เมื่อมีข้อมูลแบบตัวเลขผ่านเข้าไปยังบล็อกไดอะแกรมโดยผ่านช่องทาง Control terminals แล้วทำการประมวลผลในหน้าต่างบล็อกไดอะแกรมเมื่อโปรแกรมทำการประมวลผลเสร็จเรียบร้อยแล้วก็จะส่งข้อมูลจากหน้าต่างบล็อกไดอะแกรมไปยังหน้าต่างพรีอนท์พาเนลโดยผ่านช่องทาง Indicator terminal

#### - Wires

เป็นทางผ่านของข้อมูลระหว่าง Terminal โดยการลากสายสัญญาณทำได้โดยการเลือกเครื่องมือที่ชื่อว่า Connect wire ที่หน้าต่าง Tool palette มีลักษณะคล้ายกับหลอดด้าย เม้าส์จะเปลี่ยนเป็นรูปหลอดด้ายจากนั้นลากไปวางตรงจุดต่อของเทอร์มินอลแรกที่ต้องการลาก คลิกเม้าส์หนึ่งครั้งแล้วลากไปยังจุดต่อของเทอร์มินอลปลายทางที่ต้องการคลิกเม้าส์อีกหนึ่งครั้งก็เสร็จการเชื่อมต่อสัญญาณ

## 6. For Loop

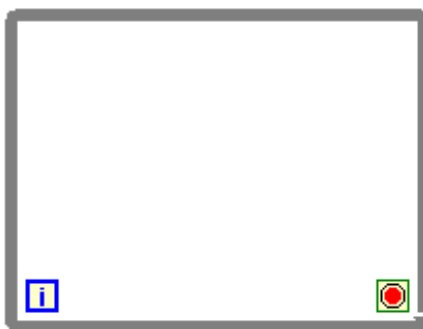
For loop เป็นฟังก์ชันคำสั่งให้โปรแกรมทำงานซ้ำจนกว่าการกระทำซ้ำนั้นจะเท่ากับ จำนวน N เมื่อ N คือจำนวนครั้งที่ต้องการให้ทำซ้ำ ใน For loop นั้นจะมีเทอร์มินอลอยู่ 2 จุดคือ Count terminal และ Iteration terminal สำหรับ Count terminal นั้นมีหน้าที่เป็นตัวกำหนดว่าจะให้โปรแกรมทำงานซ้ำกี่รอบซึ่งสามารถจะกำหนดให้ปรับเปลี่ยนจำนวนรอบหรือเป็นค่าคงที่ก็ได้ ส่วน Iteration terminal จะทำหน้าที่แสดงจำนวนรอบการทำงานว่าโปรแกรมทำงานในวงรอบไปแล้วกี่ครั้งตัวเลขที่แสดงของ Iteration Terminal จะมีค่าเท่ากับ N-1 การใช้งาน For loop ดังภาพที่ 12 การสร้าง For loop ทำได้โดยการเลือกไปที่หน้าต่างบล็อกไดอะแกรมแล้วเลือก Function » Structure แล้วใช้เมาส์เลือกที่ For loop หลังจากนั้นนำเมาส์ไปคลิกที่หน้าต่าง บล็อกไดอะแกรมแล้วลากให้มีขนาดที่ต้องการแล้วปล่อยเมาส์ก็จะได้โครงสร้างการทำงานของ For loop นอกจากนี้แล้วยังสามารถทำการย่อขยายขนาดของ For loop ได้โดยใช้เมาส์ไปชี้ตรงบริเวณเส้นขอบของ For Loop จะปรากฏจุดสี่เหลี่ยมเล็ก ๆ บริเวณเส้นขอบของ For loop ใช้เมาส์ไปวางตรงบริเวณจุดดังกล่าวแล้วเมาส์จะเปลี่ยนเป็นรูปลูกศรจากนั้นคลิกเมาส์ค้างแล้วลากย่อหรือขยายได้ตามต้องการการทำงานของ For loop นอกจากจะทำงานเพียงวงรอบเดียว ๆ แล้วยังสามารถทำงานซ้อนวงรอบได้ โดยจะทำงานวงรอบด้านในก่อนจนครบเงื่อนไขแล้วจึงมาทำงานรอบนอกหนึ่งรอบแล้วก็กลับไปทำงานในวงรอบในอีกครั้ง การทำงานจะทำจนกว่าวงรอบนอกครบตามเงื่อนไขโปรแกรมจึงจะหยุดทำงาน

ภาพที่ 12 โครงสร้างของ For loop [33]

## 7. While loop

While loop เมื่อเปรียบเทียบกับกรเขียนโปรแกรมด้วยภาษาอื่น ๆ ก็จะเปรียบได้กับ โครงสร้างของคำสั่ง Repeats until นั่นเองซึ่งลักษณะของ While loop แสดงดังภาพที่ 13 หลักการทำงานของ While loop คือจะทำงานภายในวงรอบไปจนกว่าเงื่อนไขที่ Condition terminal จะเป็นจริงจึงจะหยุดทำงานสำหรับ While loop นั้นจะมีเทอร์มินอลอยู่ 2 เทอร์มินอลเช่นเดียวกันกับ For loop คือ

Iteration terminal และ Condition terminal การทำงานของ Iteration terminal จะทำงานเหมือนกันกับ Iteration terminal ของ For loop คือทำหน้าที่แสดงจำนวนรอบที่โปรแกรมกระทำในวงรอบซึ่งแสดงผลออกมาในลักษณะของตัวเลขส่วน Condition terminal เป็นการกำหนดเงื่อนไขให้โปรแกรมหยุดทำงานซึ่งเงื่อนไขในการหยุดทำงานคือจะหยุดทำงาน เมื่อเงื่อนไขเป็นจริง (True) หรือจะให้หยุดทำงานเมื่อเงื่อนไขเป็นเท็จ (False) ก็สามารถที่จะกำหนดได้นอกจากนี้แล้วเรายังสามารถกำหนดเงื่อนไขในการหยุดทำงานได้อีกหนึ่งวิธีคือการนำเอาค่าเอาต์พุตของ Iteration Terminal มาเปรียบเทียบกับค่าที่เราต้องการไม่ว่าจะเป็นค่าคงที่หรือปรับเปลี่ยนได้โปรแกรมจะทำงานวนในวงรอบและนำค่าเอาต์พุตของ Iteration terminal มาเปรียบเทียบกับค่าที่เรากำหนดไว้แล้วจะส่งเอาต์พุตที่ได้ไปยัง Condition terminal เมื่อจำนวนรอบการทำงานมีค่าเท่ากับค่าที่เรากำหนดโปรแกรมจะหยุดทำงานในวงรอบนี้ทันทีที่กำหนดเงื่อนไขแบบนี้เปรียบได้กับการกำหนดค่า N ใน For loop นั่นเอง การสร้าง While loop ทำได้โดยการเลือกไปที่หน้าต่างบล็อกโคอะแกรมแล้วเลือก Function » Structure แล้วใช้เมาส์เลือกที่ While loop นำเมาส์ไปคลิกที่หน้าต่างบล็อกโคอะแกรมแล้วลากให้มีขนาดที่ต้องการแล้วปล่อยเมาส์ก็จะได้โครงสร้างการทำงานของ While loop นอกจากนี้แล้วยังสามารถทำการย่อขยายขนาดของ While loop ได้โดยใช้เมาส์ไปชี้ตรงบริเวณเส้นขอบของ While loop จะปรากฏจุดสี่เหลี่ยมเล็ก ๆ บริเวณเส้นขอบของ While loop ใช้เมาส์ไปวางตรงบริเวณจุดดังกล่าวแล้วเมาส์จะเปลี่ยนเป็นรูปลูกศรจากนั้นคลิกเมาส์ค้างแล้วลากย่อหรือขยายได้ตามต้องการ



ภาพที่ 13 โครงสร้างของ While loop [33]

## 8. Shift register

Shift register ประกอบไปด้วยเทอร์มินอลสองตัวคือเทอร์มินอลทางด้านอินพุตและเทอร์มินอลทางด้านเอาต์พุต Shift register สามารถใช้ทำงานร่วมกับ For loop และ While loop เท่านั้นการสร้าง Shift register ทำได้โดยการใช้เมาส์คลิกขวาบริเวณเส้นขอบ ของ For loop หรือ While loop

แล้วเลือก Add shift register ก็จะได้ Shift register ที่อยู่บนเส้นขอบทางซ้ายและขวาของ For loop และ While loop ดังภาพที่ 14

ภาพที่ 14 โครงสร้างของ Shift register [33]

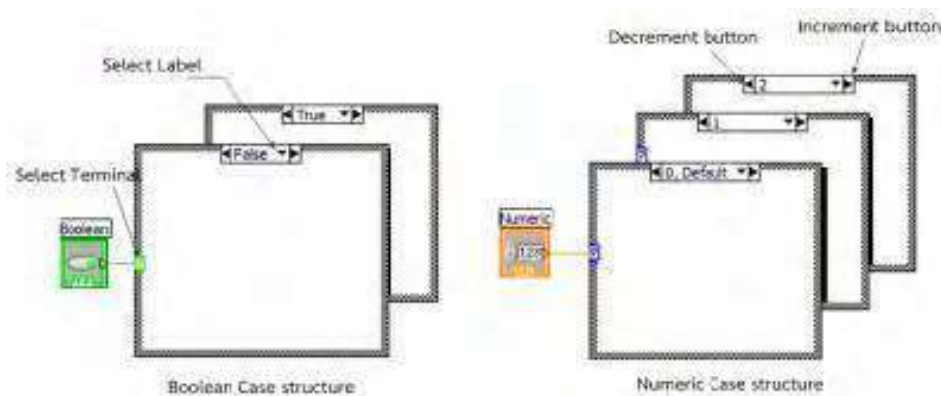
### 9. Feedback node

Feedback node สามารถทำงานร่วมกับ For loop และ While loop ได้เช่นเดียวกับ Shift register การทำงานของ Feedback node คือจะเก็บข้อมูลไว้ที่ Iteration เมื่อโปรแกรมทำงานภายในวงรอบเสร็จก็จะส่งข้อมูลดังกล่าวไปที่ Iteration ของวงรอบต่อไปทั้งนี้ Feedback node ยังสามารถกำหนดค่าเริ่มต้นได้ด้วยและสามารถใช้ได้กับข้อมูลทุกชนิดลักษณะการใช้งาน Feedback node แสดงดังภาพที่ 15

ภาพที่ 15 การใช้งาน Feedback node [33]

## 10. Case structure

Case structure เป็นโครงสร้างการเขียนโปรแกรมตามเงื่อนไขของตัวหนังสือ (Conditional text) ซึ่งถ้าเปรียบเทียบกับภาษาเขียนโปรแกรมภาษาอื่นๆ (Text-based programming languages) สามารถเปรียบเทียบได้กับโครงสร้างของคำสั่ง If...Then...Else การเรียกใช้งานฟังก์ชัน Case structure ลงบนหน้าต่างบล็อกไดอะแกรมทำได้โดยเลือก Function » Structure แล้วใช้เมาส์คลิกที่ Case structure นำเมาส์ไปคลิกที่หน้าต่างบล็อกไดอะแกรมแล้วลากให้มีขนาดที่ต้องการแล้วปล่อยเมาส์ก็จะได้โครงสร้างการทำงานของ Case structure นอกจากนี้แล้วยังสามารถทำการย่อและขยายขนาดของ Case structure ได้โดยใช้เมาส์ไปชี้ตรงบริเวณเส้นขอบของ Case structure จะปรากฏจุดสี่เหลี่ยมเล็กๆ บริเวณเส้นขอบของ While loop ใช้เมาส์ไปวางตรงบริเวณจุดดังกล่าวแล้วเมาส์จะเปลี่ยนเป็นรูปลูกศร จากนั้นคลิกเมาส์ค้างแล้วลากย่อหรือขยายได้ตามต้องการ สามารถแสดงลักษณะของ Case structure ได้ดังภาพที่ 16 ซึ่งมีส่วนประกอบที่สำคัญอยู่ 2 ส่วนคือ Select terminal และ Select label โดยที่ Select terminal จะทำหน้าที่รับอินพุตเข้ามาแล้วทำการตรวจสอบหรือเปรียบเทียบว่าข้อมูลที่รับเข้ามาเป็นข้อมูลชนิดใดและตรงกับเงื่อนไขใดใน Case structure เพื่อให้โปรแกรมทำงานตามเงื่อนไขที่ต้องการ ส่วน Select label จะทำหน้าที่สร้างหน้าต่างย่อยของ Case structure ซึ่งสามารถเพิ่มและลดเงื่อนไขได้ถึง  $2^{31} - 1$  เงื่อนไขการส่งข้อมูลออกเอาต์พุตนั้นทุก ๆ กรณีจะส่งข้อมูลออกที่เอาต์พุตเดียวกับซึ่งทางออกของสัญญาณเอาต์พุตเรียกว่า Output tunnel



ภาพที่ 16 โครงสร้างของ Case structure [33]

## 11. Flat and stack sequence structures

Sequence structures เป็นการทำงานแบบลำดับขั้นรูปแบบของ Sequence structures มีลักษณะเหมือนแผ่นฟิล์มสำหรับ Sequence structures มีทั้งหมด 2 แบบคือ Flat sequence และ Stack sequence ลักษณะของ Flat sequence และ Stack sequence จะมีลักษณะคล้ายกับ Case structure กล่าวคือจะมี Select case สำหรับเลือกเฟรมที่ซ่อนอยู่ภายใน โดยตัวเลขจะบอกว่ากำลังเปิดเฟรมใดอยู่ การสร้าง Flat และ Stack sequence structures ทำได้โดยการเลือกไปที่หน้าต่าง บล็อกไดอะแกรม แล้วเลือก Function » Structure แล้วใช้เมาส์เลือกที่ Flat หรือ Stack sequence structures ตามที่ต้องการ แสดงดังภาพที่ 17 นำเมาส์ไปคลิกที่หน้าต่างบล็อกไดอะแกรมแล้วลากให้มีขนาดที่ต้องการแล้วปล่อยเมาส์ก็จะได้โครงสร้างการทำงานของ Flat และ Stack sequence structures การที่จะนำข้อมูลของเฟรมหนึ่งไปยังอีกเฟรมหนึ่งสามารถทำได้โดยคลิกเมาส์ทางขวาที่ บริเวณกรอบ Sequence structures แล้วเลือก Add sequence local จะปรากฏเทอร์มินอลขึ้นที่ขอบของ Sequence structures จากนั้นลากสายสัญญาณที่เราต้องการให้ออกเอาต์พุตไปต่อกับเทอร์มินอลซึ่งเทอร์มินอลดังกล่าวก็จะเปลี่ยนเป็นลักษณะที่มีลูกศรพุ่งออกจากเฟรม ในขณะที่เดียวกันทุกเฟรมที่เลือกก็จะปรากฏเทอร์มินอลขึ้นเช่นกันแต่จะมีลักษณะลูกศรชี้เข้าไปข้างในเฟรมแสดงดังภาพที่ 18

ภาพที่ 17 โครงสร้างของ Flat และ Stack sequence structures [33]



ภาพที่ 18 การ Add sequence local ของ Stack sequence structures [33]

## 12. Formula node

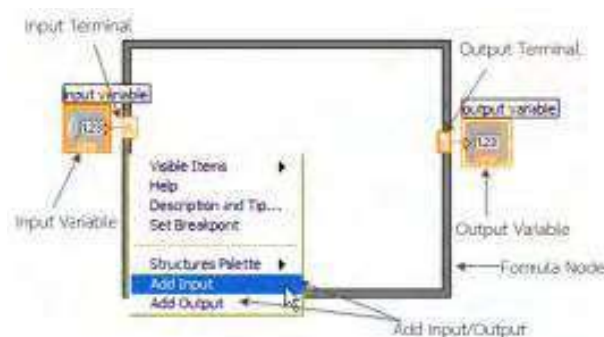
Formula node เป็นโครงสร้างของคำสั่งที่มีความใกล้เคียงกับการเขียนโปรแกรมด้วยภาษาอื่น ๆ ทั่วไปซึ่ง Formula node จะถูกนำมาใช้เมื่อต้องการให้โปรแกรมทำงานตามฟังก์ชันทางคณิตศาสตร์ที่มีความยุ่งยากการใช้อุปกรณ์ในการสร้างให้โปรแกรมทำงานตามฟังก์ชันทางคณิตศาสตร์หนึ่งฟังก์ชันอาจต้องใช้อุปกรณ์หลายตัวซึ่งอาจทำให้การตรวจสอบโปรแกรมเป็นไปด้วยความยุ่งยากและโปรแกรมมีขนาดใหญ่เกินไป ยกตัวอย่างการสร้างโปรแกรมให้ทำงานตามสมการ  $y = x - e \sin(x)$  ซึ่งถ้าเขียนโปรแกรมด้วยการใช้อุปกรณ์ทั่วไปจะได้ลักษณะของโปรแกรมดังภาพที่ 19 แต่หากใช้ Formula node จะได้ลักษณะของโปรแกรมดังภาพที่ 18

ภาพที่ 19 การสร้างโปรแกรมจากอุปกรณ์ทั่วไป [33]

การทำงานของ Formula node จะเป็นลักษณะของการกำหนดฟังก์ชันทางคณิตศาสตร์ลงไป ใน Formula node แสดงดังภาพที่ 20 แล้วจึงกำหนดเทอร์มินอลเพื่อเชื่อมต่ออินพุตและเอาต์พุต การสร้าง Formula node ทำได้โดยการเลือกไปที่หน้าต่างบล็อกไดอะแกรมแล้วเลือก Function » Structure แล้วใช้เมาส์เลือกที่ Formula node แสดงดังภาพที่ 18 นำเมาส์ไปคลิกที่หน้าต่างบล็อกไดอะแกรมแล้วลากให้มีขนาดที่ต้องการแล้วปล่อยเมาส์ก็จะได้โครงสร้างการทำงานของ Formula node

ภาพที่ 20 โครงสร้างและการใช้งาน Formula node [33]

การกำหนดเทอร์มินอลอินพุตและเอาต์พุตสามารถทำได้โดยการคลิกเมาส์ขวาที่ขอบของ Formula node แล้วเลือก Add input หรือ Add output แสดงดังภาพที่ 21 ที่เทอร์มินอลสามารถที่จะกำหนดชื่อได้และสามารถ ค่าอินพุตและเอาต์พุตจะเป็นข้อมูลแบบตัวเลข (Numeric) การกำหนดตัวแปรอินพุตสามารถกำหนดให้เป็นค่าคงที่หรือค่าที่ปรับเปลี่ยนได้ส่วนเอาต์พุตจะกำหนดให้เป็นตัวแสดงผลชื่อที่เทอร์มินอลก็คือ ชื่อของตัวแปรของฟังก์ชันทางคณิตศาสตร์ที่สร้างขึ้นใน Formula node



ภาพที่ 21 การเพิ่มอินพุตและเอาต์พุตของ Formula node [33]

### 13. ฟังก์ชันของอาร์เรย์

#### - Array size

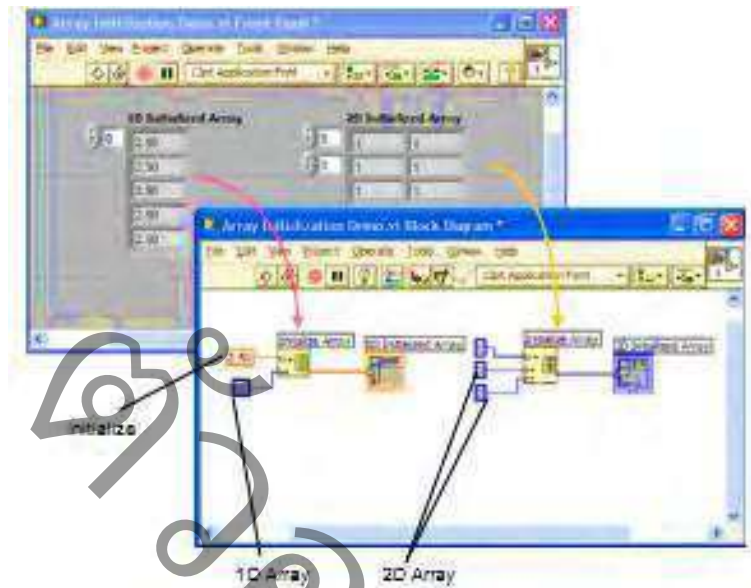
ฟังก์ชัน Array size จะเป็นตัวบอกขนาดของอาร์เรย์ว่ามีขนาดใดเช่นถ้าอาร์เรย์อินพุต เป็นอาร์เรย์ 1 มิติ (4 แถว 1 หลัก) อาร์เรย์เอาต์พุตจะเป็นอาร์เรย์ 1 มิติ มีจำนวนสมาชิกทั้งหมด 4 ตัวหรือถ้าอาร์เรย์อินพุตเป็นอาร์เรย์ 2 มิติ (2 แถว 4 หลัก) อาร์เรย์เอาต์พุตจะแสดง เป็นอาร์เรย์ 2 มิติ (แสดงจำนวนของแถวและหลัก) แสดงดังภาพที่ 22



ภาพที่ 22 การใช้งาน Array size function [33]

#### - Initialize array

เป็นการสร้างอาร์เรย์ตามขนาดที่กำหนดพร้อมทั้งสามารถกำหนดค่าเริ่มต้นให้กับอาร์เรย์ที่สร้างได้อีกด้วย เช่น ต้องการสร้างอาร์เรย์ 1 มิติ มีจำนวนสมาชิกทั้งหมด 50 ตัว และกำหนดให้ค่าเริ่มต้นของอาร์เรย์เป็น 2.5 สามารถสร้างได้โดยคลิกเมาส์ขวาที่เทอร์มินอล Element ซึ่งอยู่ทางด้านบนซ้ายของบล็อก Initialize array แล้วเลือก create » Constant และกำหนดค่าเริ่มต้นเป็น 2.5 ที่เทอร์มินอล Dimension size ของบล็อก Initialize array ที่อยู่ด้านล่างเทอร์มินอล Element คลิกเมาส์ขวาแล้วเลือก Create » Constant และกำหนดให้เป็น 50 (จำนวนสมาชิก) ถ้าต้องการเพิ่มมิติของอาร์เรย์ก็สามารถทำได้โดยคลิกเมาส์ขวาที่บล็อกของ Initialize array แล้วเลือก Add dimension ซึ่งจะเพิ่มเทอร์มินอลเพื่อกำหนดขนาดของแถวและหลัก แสดงดังภาพที่ 23



ภาพที่ 23 การใช้งาน Initialize array function [33]

#### - Build array

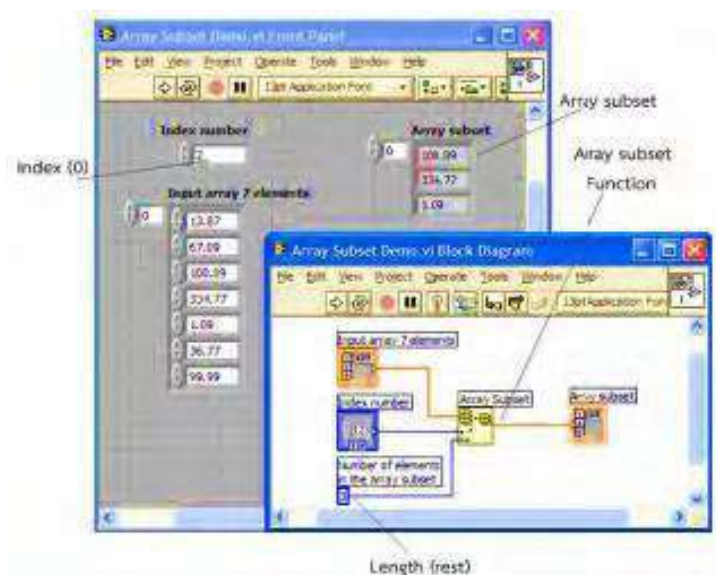
เป็นการนำเอาอาร์เรย์หรือสมาชิกในอาร์เรย์หลาย ๆ ตัวมารวมกันแล้วเกิดเป็นอาร์เรย์ใหม่ ซึ่งอินพุตสามารถที่จะเป็นทั้งอาร์เรย์และเป็นจำนวนสเกลาร์โดยปกติแล้ว Build array จะมีอินพุต 2 อินพุตถ้าต้องการเพิ่มจำนวนอินพุตสามารถทำได้โดยการคลิกเมาส์ขวาที่บล็อก Build array แล้วเลือก Add input หรือจะใช้ Positioning tool คลิกค้างแล้วลากขยายเพื่อเพิ่มอินพุตและถ้าต้องการลดจำนวนอินพุตก็สามารถทำได้โดยคลิกเมาส์ขวา Build array แล้วเลือก Remove input ตัวอย่างการใช้งาน Build array แสดงดังภาพที่ 24



ภาพที่ 24 การใช้งาน Build array function [33]

**- Array subset**

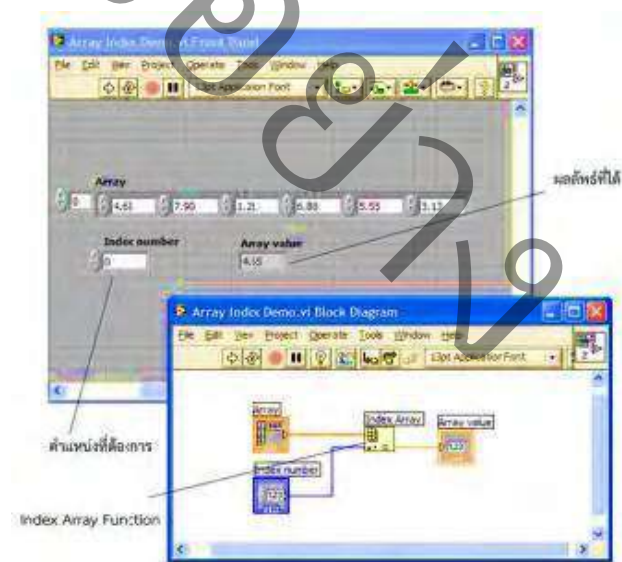
เป็นวิธีการเลือกค่าของข้อมูลที่อยู่ในอาร์เรย์ไปใช้งาน โดยมีสิ่งที่จะต้องกำหนดอยู่ 2 อย่างคือ Index (0) ซึ่งเป็นการกำหนดตำแหน่งเริ่มต้นที่เราต้องการ (ตำแหน่งแรกนับเป็นตำแหน่งที่ 0) และ Length (Rest) เพื่อกำหนดจำนวนข้อมูลที่ต้องการ ดังตัวอย่างในภาพที่ 23 อาร์เรย์ 1 มิติ มีจำนวนข้อมูล 7 ข้อมูลต้องการใช้ข้อมูลตั้งแต่ตำแหน่งที่ 3 ถึง 5 ทำได้โดยกำหนด Index (0) ให้มีค่าเท่ากับ 2 และ Length (Rest) ให้มีค่าเท่ากับ 3 แสดงดังภาพที่ 25



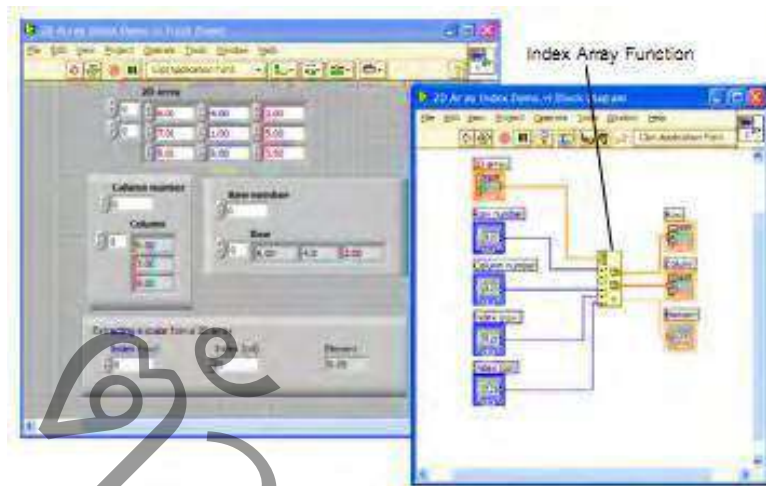
ภาพที่ 25 การใช้งาน Subset array function [33]

### - Index array

ฟังก์ชัน Index array สามารถที่จะเลือกข้อมูลในอาร์เรย์ในตำแหน่งที่ต้องการมาใช้งานได้โดยการเลือกตำแหน่งของข้อมูลได้เลยในกรณีที่เป็นอาร์เรย์ขนาด 2 มิติ ก็สามารถใช้ Index array เลือกที่จะเลือกข้อมูลออกเอาต์พุตที่ตำแหน่งแถวและหลัก แสดงดังภาพที่ 26 มีอาร์เรย์ 1 มิติ มีจำนวนข้อมูลทั้งหมด 6 ข้อมูลต้องการเลือกข้อมูลที่ 1 ไปใช้งานสามารถทำได้โดยกำหนดให้ Index number มีค่าเท่ากับ 0 (ตำแหน่งแรกนับเป็นตำแหน่งที่ 0) และในภาพที่ 27 เป็นการใช้ฟังก์ชัน Index array ในการเลือกข้อมูลจากอาร์เรย์ 2 มิติ ขนาด 3 x 3 โดย Index array จะต้องมี 2 ตัวเพื่อเลือกตำแหน่งแถวและตำแหน่งหลักที่ต้องการ ซึ่งการเพิ่มอินพุตของ Index array สามารถทำได้โดยการใช้ Position tool คลิกที่ Index array แล้วลากขยายเพื่อเพิ่มอินพุตให้กับ Index array



ภาพที่ 26 การใช้งาน Index array function [33]



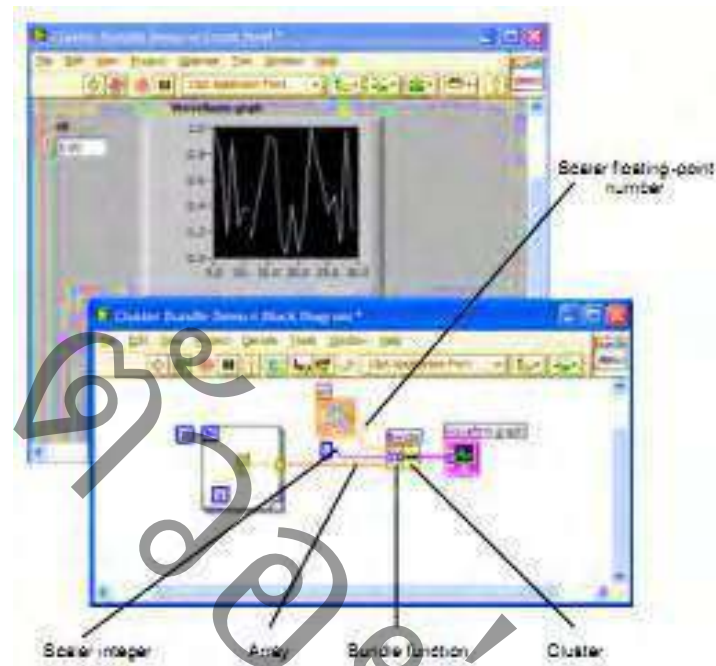
ภาพที่ 27 การใช้งาน Index array function กับอาร์เรย์ 2 มิติ [33]

#### 14. ฟังก์ชันของคลัสเตอร์

ฟังก์ชันที่ช่วยในการทำงานของคลัสเตอร์มีอยู่หลายแบบด้วยกันในที่นี้จะขอล่าถึง ฟังก์ชันที่ถูกนำมาใช้งานบ่อย ๆ 2 ฟังก์ชันคือ Bundle และ unbundle ซึ่งวิธีที่เรียกใช้ฟังก์ชันทำโดยเลือกที่ Function palette » All function » Cluster แล้วเลือกฟังก์ชัน Bundle หรือ Unbundle ตามต้องการ

##### - Bundle function

Bundle function เป็นฟังก์ชันที่ใช้รวมเอาอินพุตหลาย ๆ ตัวมาสร้างเป็นคลัสเตอร์ใหม่ขึ้นมา โดยอินพุตนั้นสามารถเป็นได้ทั้งตัวเลขจำนวนจริง อาร์เรย์ หรือแม้กระทั่งคลัสเตอร์ เมื่อวางบล็อกของฟังก์ชัน Bundle ลงไปในหน้าต่างบล็อกไดอะแกรมฟังก์ชัน Bundle จะมีอินพุตเทอร์มินอล 2 เทอร์มินอลอยู่ทางซ้ายและเทอร์มินอลเอาต์พุต 1 เทอร์มินอลอยู่ทางขวาและอินพุตที่เป็นคลัสเตอร์จะอยู่ด้านบนของบล็อกเราสามารถที่จะเพิ่มหรือลดจำนวนเทอร์มินอลอินพุตให้มีจำนวนเท่าที่เราจำเป็นต้องใช้งาน โดยใช้เมาส์คลิกที่บล็อกแล้วเลื่อนขึ้นลงในแนวตั้งหรือใช้เมาส์ไปวางตรงเทอร์มินอลอินพุตแล้วคลิกเมาส์ขวาแล้วเลือก Add input ถ้าต้องการเพิ่มเทอร์มินอลและเลือก Remove input ถ้าต้องการลดเทอร์มินอล ในภาพที่ 28 จะเป็นตัวอย่างการสร้างคลัสเตอร์ด้วย Bundle โดยมีอินพุต 3 ชนิดคือ Floating-point number, Scalar integer และ Array ที่สร้างจาก For loop และมีเอาต์พุตที่แสดงผลด้วยกราฟ

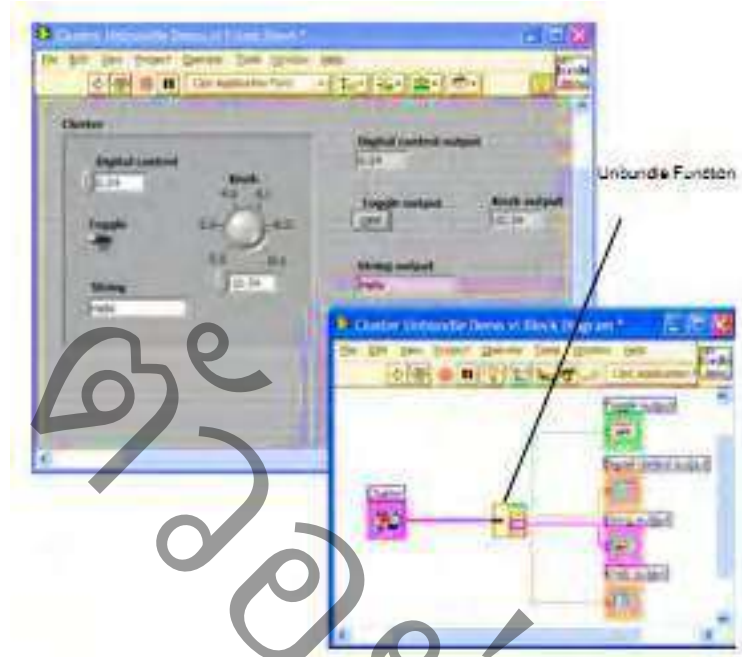


ภาพที่ 28 การใช้งาน Bundle function [33]

#### - Unbundle function

ในคลัสเตอร์จะประกอบไปด้วยชนิดของข้อมูลที่แตกต่างกัน Unbundle function จะช่วยในการแยกข้อมูลอินพุตที่แตกต่างกันออกจากกันเพื่อจะนำเอาต์พุตนั้นไปใช้งานต่อไป ฟังก์ชันของ Unbundle ที่วางลงไปบนหน้าต่างบล็อกไดอะแกรมครั้งแรกจะมีอินพุต 1 อินพุตและมีเอาต์พุต 2 อินพุต เอาต์พุต Unbundle จะเปลี่ยนไปตามอินพุตที่เป็นคลัสเตอร์ ตัวอย่างเช่น มีคลัสเตอร์อินพุตที่มีชนิดของข้อมูลแตกต่างกัน 4 ชนิด เมื่อต่อสายของคลัสเตอร์เข้ากับอินพุตของ Unbundle เอาต์พุตของ Unbundle จะเปลี่ยนเป็น 4 เอาต์พุตโดยอัตโนมัติ





ภาพที่ 29 การใช้งาน Unbundle function [33]

## 15. การใช้งาน Charts และ Graph

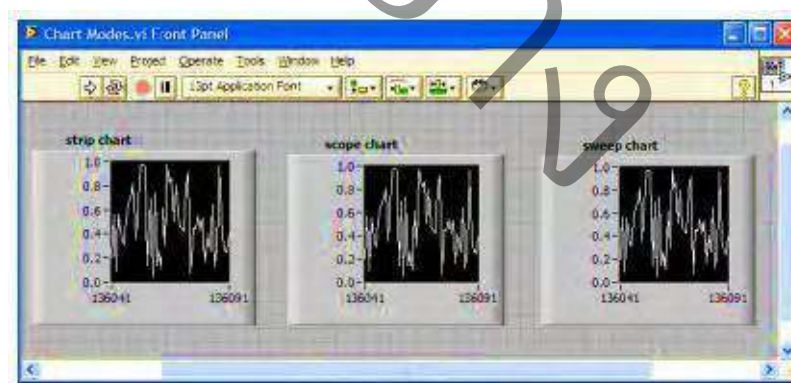
### - Waveform charts

การแสดงผลข้อมูลในลักษณะของ Charts เป็นการแสดงการพล็อตรูปคลื่นสัญญาณข้อมูล แบบหนึ่งที่สามารถพล็อตข้อมูลที่เป็นสเกลาร์และเป็นอาร์เรย์ได้ การเรียกใช้งาน คือเลือกที่ Controls Palette » Graph » Waveform chart ดังภาพที่ 30



ภาพที่ 30 วิธีการเรียกใช้งาน Waveform chart [33]

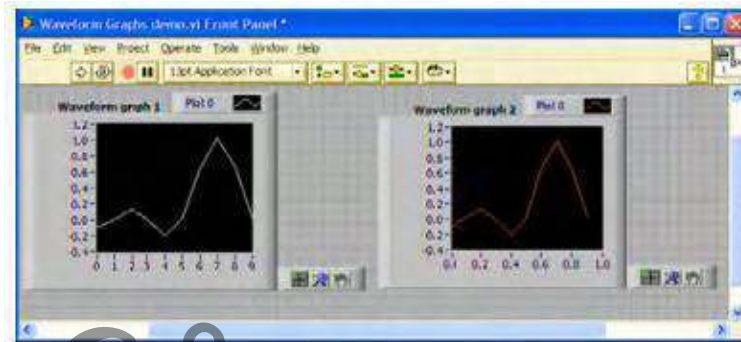
โดย Waveform charts ในโปรแกรม LabVIEW มีเพียงชนิดเดียวแต่มีคุณสมบัติในการแสดงข้อมูลที่แตกต่างกัน 3 อย่างคือ Strip chart, Scope chart และ Sweep chart ซึ่งข้อแตกต่างของ Chart ทั้ง 3 แบบอธิบายได้คือ Strip chart มีคุณสมบัติในการพล็อตข้อมูลไปจนกว่าจะหยุดพล็อต ซึ่งจะสังเกตเห็นว่าเส้นของสัญญาณข้อมูลที่พล็อตนั้นจะเลื่อนจากขวาไปซ้ายต่อเนื่องกันไป Scope chart คุณสมบัติของการพล็อตแบบนี้ก็จะแบ่งเป็นช่วงของข้อมูลโดยการกำหนดที่แกนเวลาเมื่อโปรแกรมทำการพล็อตข้อมูลไปถึงค่าสุดท้ายของช่วงแรกแล้วการพล็อตข้อมูลของช่วงต่อไป โปรแกรมจะทำการลบกราฟในช่วงแรกไปแล้วเริ่มการพล็อตใหม่โดยเวลาเริ่มต้นจะนับต่อจากค่าสุดท้ายของช่วงแรก Sweep chart จะทำงานคล้ายกับ Scope chart แต่เมื่อเริ่มทำการพล็อตกราฟในช่วงต่อไปเส้นกราฟในช่วงแรกจะไม่ถูกลบจะทำงานในลักษณะพล็อตทับเส้นกราฟเดิมและจะมีเส้นสีแดงในแนวตั้งเลื่อนจากซ้ายไปขวาเพื่อแสดงถึงจุดการพล็อตกราฟครั้งล่าสุดคุณสมบัติทั้ง 3 ของ Waveform chart แสดงดังภาพที่ 31 เป็นการแสดงค่าของข้อมูลที่เก็บอยู่ในอาร์เรย์เทียบกับเวลาที่รับเข้ามา



ภาพที่ 31 คุณสมบัติของ Waveform chart [33]

#### - Waveform graphs

แตกต่างจาก Waveform chart คือการพล็อตข้อมูลด้วย Waveform graphs จะมีการกำหนดจำนวนของช่วงข้อมูลอย่างชัดเจนจำนวนข้อมูลทั้งหมดที่อยู่ในช่วงของกราฟจะถูกแสดงออกมาทั้งหมด การพล็อตกราฟหลายเส้นทำได้โดยการนำอาร์เรย์แต่ละตัวมาต่อรวมเพื่อสร้างอาร์เรย์ใหม่จำนวนเส้นกราฟที่แสดงจะเท่ากับมิติของอาร์เรย์การเรียกใช้งาน Waveform graphs ทำได้คือเลือกที่ Controls palette » Modern » Graph » Waveform graphs ลักษณะของการพล็อตกราฟลงบน Waveform graph แสดงได้ดังภาพที่ 32



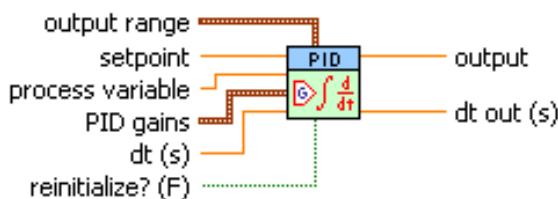
ภาพที่ 32 คุณสมบัติของ Waveform graph [33]

### 16. PID Control toolkit

การใช้ฟังก์ชัน PID ในการออกแบบระบบควบคุมนั้นในโปรแกรม LabVIEW นั้นมีเครื่องมือที่สามารถนำมาประยุกต์ใช้ ให้เข้ากับทฤษฎีระบบควบคุมได้โดยง่ายและหลากหลาย การประยุกต์ใช้ระบบควบคุมแบบ PID และวิธีการใช้งาน Function block ต่าง ๆ ของ PID Toolkit เพื่อที่สามารถนำไปประยุกต์ใช้ให้เหมาะสมกับกระบวนการที่จะทำการควบคุมต่อไป



ภาพที่ 33 Panel icon ของ PID [33]



ภาพที่ 34 การเชื่อมต่อสัญญาณของฟังก์ชัน PID [33]

การใช้งานฟังก์ชันตัวควบคุม PID ภายในโปรแกรม LabVIEW นั้น จะต้องมีการเชื่อมต่อสัญญาณต่าง ๆ ให้แก่ฟังก์ชัน ดังแสดงได้ในภาพที่ 34 โดยสัญญาณต่าง ๆ สามารถอธิบายได้ดังนี้

Set point	คือ ค่าสัญญาณอ้างอิง
Process variable	คือ ค่าสัญญาณเป้าหมายที่ต้องการควบคุม
Output	คือ สัญญาณควบคุมที่ออกจากตัวควบคุม
Output rage	คือ ช่วงของสัญญาณเอาต์พุตของตัวควบคุมพีไอดี
PID gains	คือ ค่าพารามิเตอร์ของตัวควบคุมพีไอดี
dt (s)	คือ ค่าฐานเวลาการคำนวณของตัวควบคุมพีไอดี
dt out (s)	คือ เวลาที่ใช้ในการคำนวณสัญญาณควบคุม
Reinitialize	คือ การรีเซ็ตค่าสัญญาณควบคุมให้กลับไปสู่สถานะเริ่มต้น

ดร.อรรถพร

ภาคผนวก ข

ผลงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์ที่ได้รับการตีพิมพ์

### ผลงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์ที่ได้รับการตีพิมพ์

1. ชัยพร ไทรเกตุ, สันติ หวังนิพนพานโต, “การควบคุมความดันในถังลมแบบอนุกรม 2 ถัง โดยใช้พีแอลซี”, การประชุมวิชาการเครือข่ายวิศวกรรมไฟฟ้ามหาวิทยาลัยเทคโนโลยีราชมงคลครั้งที่ 8, Proceeding of The 8<sup>th</sup> Conference of Electrical Engineering Network of Rajamangala University of Technology (EENET 2016), 2016.

2. ภาคภูมิ ศรีทองนวล, ชัยพร ไทรเกตุ, สันติ หวังนิพนพานโต : การศึกษาการเติมอากาศเพื่อลดอุณหภูมิของอุปกรณ์อิเล็กทรอนิกส์ในตู้ควบคุมลิฟต์ (Studies on The Filling Air to Reducing The Electronics Equipment Temperature in The Elevator Control Cabinet) การประชุมวิชาการเรื่องการถ่ายเทพลังงานความร้อนและมวลในอุปกรณ์ด้านความร้อนและกระบวนการ (ครั้งที่ 15) ระหว่างวันที่ 30-31 มีนาคม 2559 ณ โรงแรม แอล รีสอร์ท สมุย จังหวัดสุราษฎร์ธานี

การควบคุมความดันในถังผสมอนุกรม 2 ถังโดยใช้พีแอลซี

Pressure Control in Series 2 Tank by Using PLC

จิพร ไตรภู่<sup>1</sup>, สันติ ศรีวันพิพัฒน์<sup>2</sup> และศพร สมณสิทธิ์<sup>3</sup>

<sup>1</sup> ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า

พระจอมเกล้าพระนครเหนือ กรุงเทพมหานคร

โทรศัพท์ 09-09773786, E-mail : charaporn.sangtho@gmail.com, jupprato@gmail.com

บทคัดย่อ

งานวิจัยนี้จะศึกษาการควบคุมความดันในถังผสมอนุกรมซึ่งควบคุมความดันของถังที่สองที่ความดันที่ให้มีขนาดเป็นสองเท่าของถังแรก (ขนาด 60 ซม. และขนาดเส้นผ่าศูนย์กลาง 25 ซม. และ 30 ซม. ซึ่งต่อเชื่อมกัน โดยการเชื่อมแบบอนุกรมความดันในระบอบเปิด) โดยใช้วิธีการของ Ziegler-Nichols ในการออกแบบตัวควบคุมแบบ PID และศึกษาผลของปริมาณการรบกวนที่เพิ่มมากขึ้น จากการผลิตแบบไม่ต่อเนื่องที่มี การตั้งความดันเป้าหมายเพิ่มขึ้นหรือลดลง ระบบควบคุมที่ออกแบบนี้ทดสอบภายใต้การรบกวนที่เพิ่มมากขึ้นโดยที่ความดันได้ตามที่ต้องการ โดยมีค่าความคลาดเคลื่อนที่ค่าเฉลี่ยที่น้อยกว่า 1% และใช้ระยะเวลาเพิ่มขึ้น (Rise time) น้อยกว่า 12 วินาที นอกจากนี้ยังมีการศึกษาผลกระทบของระบบแบบเปิดที่พบที่ส่งผลต่อการควบคุมที่เพิ่มมากขึ้นของระบบดังกล่าวไว้ด้วย เพื่อสำหรับการวิจัยขั้นต่อไป

คำสำคัญ: ความดันในถัง, การควบคุมแบบปิด, ระบบอันดับหนึ่ง

Abstract

This research studies on the pressure control in horizontal tank in order to maintain the set point pressure following the users. Pressure tanks diameter 12 cm and length 60 cm and diameter 12 cm and length 60 cm are connected in series. The Ziegler-Nichols method's used for designed the PI controller and set parameters on PLC in order to studies the influence of parameters on the control performance under disturbance conditions. It was found that when the pressure set point increase (+) decrease, the PLC controller can be control pressure in the horizontal tank. has steady state error less than 1%, and has the rise time less than 12 seconds per pressure step 2.0bar. In addition to, the open systems is tested and get the transfer function of the pressure in horizontal tank to conduct using its in next research step.

Keywords: Pressure in the tank, PI control, First-Order System.

1. บทนำ

โรงงานอุตสาหกรรมทั่วไป จะทำการควบคุมค่าคงที่ในสิ่งที่มีอยู่ การที่ได้อัตราการควบคุมที่มีคุณภาพดี คือ การใช้ค่าปริมาณการผิดพลาดที่น้อย การเพิ่ม การควบคุมระดับน้ำ ความดัน อุณหภูมิ อัตราการไหล การเพิ่มปริมาณที่เพิ่มขึ้น จากปริมาณควบคุมความดันในระบอบเปิด เมื่อใช้วิธีการต่างๆ เช่น วิธี Ziegler-Nichols จะทำให้ได้ผลลัพธ์ที่ดี การศึกษาการควบคุมความดันในถังผสมอนุกรมเปิดที่ได้โดยอัตโนมัติ โดยการใช้วิธีการ Ziegler-Nichols ในการออกแบบตัวควบคุมแบบ PID และศึกษาผลของปริมาณการรบกวนที่เพิ่มมากขึ้น จากการผลิตแบบไม่ต่อเนื่องที่มี การตั้งความดันเป้าหมายเพิ่มขึ้นหรือลดลง ระบบควบคุมที่ออกแบบนี้ทดสอบภายใต้การรบกวนที่เพิ่มมากขึ้นโดยที่ความดันได้ตามที่ต้องการ โดยมีค่าความคลาดเคลื่อนที่ค่าเฉลี่ยที่น้อยกว่า 1% และใช้ระยะเวลาเพิ่มขึ้น (Rise time) น้อยกว่า 12 วินาที นอกจากนี้ยังมีการศึกษาผลกระทบของระบบแบบเปิดที่พบที่ส่งผลต่อการควบคุมที่เพิ่มมากขึ้นของระบบดังกล่าวไว้ด้วย เพื่อสำหรับการวิจัยขั้นต่อไป

2. ทฤษฎี

2.1 ความดัน (Pressure)

ความดัน เป็นปริมาณที่แสดงถึงในทางฟิสิกส์ หมายถึง แรงต่อพื้นที่ของไหลกระทำต่อพื้นที่ตั้งฉากกับพื้นที่ที่แรงนั้นกระทำ

$$P = \frac{F}{A} \tag{1}$$

- P คือ ความดัน (Pressure) มีหน่วยเป็น  $N/m^2$  , Pa
- F คือ แรงที่กระทำต่อพื้นที่ตั้งฉากกับพื้นที่  $N$
- A คือ พื้นที่ มีหน่วยเป็น  $m^2$

2.2 ระบบเปิดของระบบความดันในถังผสมอนุกรม 2 ถัง

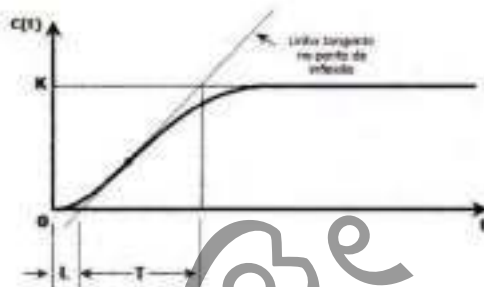
การหาความสัมพันธ์ระหว่างปริมาณการควบคุมความดันในถัง (P) และค่าเป้าหมายความดัน (Set Point) ที่ต้องการจะควบคุม โดยที่ค่าเป้าหมายความดันที่เพิ่มมากขึ้นหรือลดลง จะทำให้ระบบแบบเปิดมีการตอบสนองการควบคุมความดันในถังที่ล่าช้ากว่าค่าเป้าหมายที่ต้องการ

GN09

**บทความวิจัย**

การประชุมวิชาการเครือข่ายวิศวกรรมไฟฟ้ามหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ ครั้งที่ ๕

Proceedings of the 5<sup>th</sup> Conference of Electrical Engineering Network of Rajabhatngala University of Technology 2016 (EENET 2016)



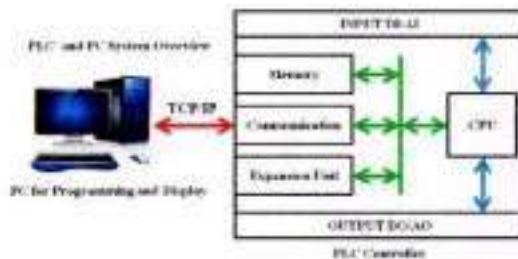
รูปที่ 1. ลักษณะการตอบสนองของระบบที่มีค่าคงที่รวมกับคืบ

$$G_p(s) = \frac{K e^{-Ls}}{TS + 1} \quad (2)$$

- เมื่อ:  $K_p$  คือค่าคงที่ (Process steady state gain) ของระบบที่ควบคุม
- $T$  คือ ค่าคงที่เวลา (Process time constant)
- $L$  คือค่าความหน่วง (Delay time)
- $C_p(s)$  คือค่าความถี่ของฟังก์ชันของระบบอันดับหนึ่งที่มีค่าคงที่รวม

**1.3 ตัวควบคุมฟีดแบ็ค**

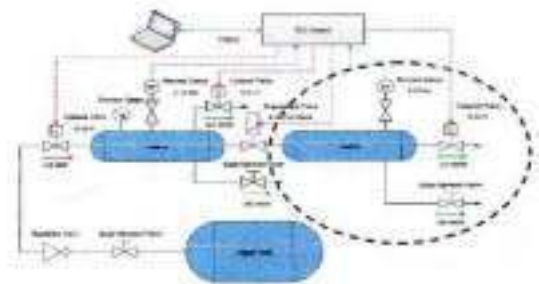
ฟีดแบ็ค เป็นชุดเครื่องมือควบคุมที่มีความสามารถในการทำงานซ้ำๆ มีความแม่นยำในการคำนวณ และมีกระบวนการเชิงตรรกะ สามารถเชื่อมต่อกับคอมพิวเตอร์ เพื่อใช้งานระบบควบคุมดังรูปที่ 2.



รูปที่ 2. การต่อระบบควบคุมฟีดแบ็คของระบบอัตโนมัติ

**3. ขั้นตอนการทดลอง**

**3.1 ระบบทดสอบโดยทั่วไป**

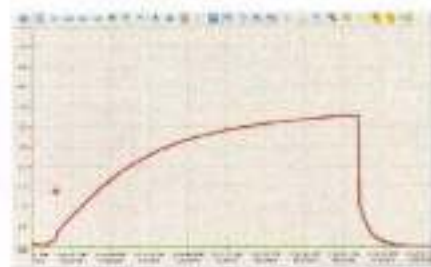


รูปที่ 3. ระบบทดสอบรวมของการควบคุมความเร็วเชิงกล

รูปที่ 3. เป็นระบบทดสอบโดยทั่วไปประกอบด้วย 1. เครื่องคอมพิวเตอร์ที่รันโปรแกรมควบคุมที่เชื่อมต่อกับ PLC 2. ชุดควบคุมความเร็วเชิงกลที่มีขนาดเล็กรุ่นชุดระบบ 4-20 มีดักกระแสไฟฟ้าที่รับรับรับค่าความเร็วเชิงกลและแรงบิดชุดระบบแรงบิด 2-10 โวลต์ดีซีออก สำหรับส่งสัญญาณกลับสู่ควบคุม 3. ชุดโซลินอยด์ตัวจำนวน 3 ตัว แรงดันที่ 24 โวลต์ที่ควบคุมเปิด/ปิดลิ้นลมของชุดออก 4. ชุดควบคุมความเร็วเชิงกลจำนวน 1 ตัว แรงดันที่ 24 โวลต์ เพื่อควบคุมความเร็ว จากคอมพิวเตอร์ไปยังชุดที่ 3 และ 4 3. สัญญาณของชุดควบคุม 8 บิตสามารถวัดความเร็วเชิงกลได้และชุดควบคุมที่เชื่อมต่อกับความเร็วเชิงกล

**3.2 การทดสอบค่าพารามิเตอร์ที่ขึ้นของระบบ**

การทดลองจะวัดค่าความถี่อินพุตด้วยชุดอุปกรณ์ปรับความดัน (Pressure Regulator) แล้วเปลี่ยนความถี่อินพุตแบบขึ้นจากศูนย์จนถึงความถี่ที่ตัวไว้ เพื่อดูการตอบสนองของความเร็วเชิงกลที่แรงกดที่ชุด โดยจะใช้เครื่องมือที่เชื่อมต่อกับตัววัดดังรูปที่ 4.



รูปที่ 4. การตอบสนองของระบบความเร็วเชิงกลที่ขึ้นกับความเร็วเชิงกล



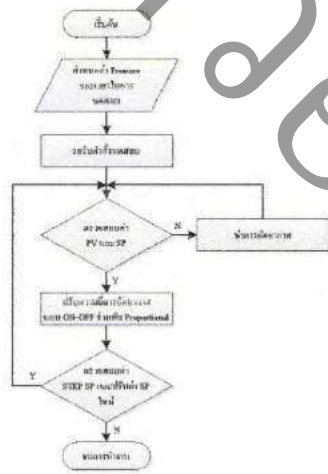
**บทความวิจัย**

การประชุมวิชาการเครือข่ายวิศวกรรมไฟฟ้ามหาวิทยาลัยเทคโนโลยีราชมงคล ครั้งที่ 8

*Proceedings of the 8<sup>th</sup> Conference of Electrical Engineering Network of Rajamangala University of Technology 2016 (EENET 2016)*

**3.3 ขั้นตอนการทดสอบระบบ**

การทดสอบโดยป้อนค่าความดันที่ต้องการ SP(Set-point)ของถังที่จอแสดงผลและคอมพิวเตอร์จะส่งข้อมูลไปที่พีแอลซี เพื่อนำไปใช้ในการประมวลผล เมื่อสั่งให้เริ่มต้นการทดสอบพีแอลซีจะประมวลผลหาค่า MV (Manipulated Value) เพื่อใช้สำหรับการควบคุมวิธีการเปิด-ปิดร่วมกับแบบอัตราส่วน (Proportional) ควบคุมความดันถังลม จากนั้นตัวตรวจจับความดันจะส่งสัญญาณ PV (Process Variable) กลับมาที่พีแอลซีเพื่อใช้ประมวลผลการควบคุมต่อไป ดังแสดงขั้นตอนการทำงานดังรูปที่ 5.



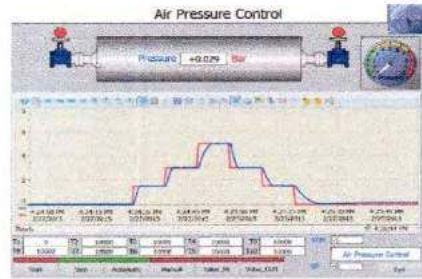
รูปที่ 5. โฟลว์ชาร์ตการทำงานของระบบควบคุม

จากรูปที่ 5. เป็นการทำงานทางซอฟต์แวร์ของระบบโดยรวม โดยจะมีส่วนสำคัญอยู่ 4 ส่วน คือ ส่วนรับข้อมูล ส่วนประมวลผล ส่วนของการปรับความดันและช่วงเวลาในการทดสอบ ด้วยโปรแกรมภายในพีแอลซี

**4. ผลการทดลอง**

**4.1 ส่วนของการแสดงผลและการติดตั้งระบบจริง**

การแสดงผลจะแสดงในรูปที่ 6. สามารถตั้งค่าความดันที่ต้องการ และช่วงเวลาในการทดสอบได้มากกว่า 4 ตำแหน่ง สำหรับส่วนของการติดตั้งระบบจริง แสดงในรูปที่ 7.

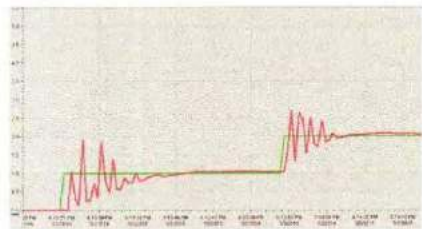


รูปที่ 6. การแสดงผลหน้าจอและการตั้งค่าต่างๆ



รูปที่ 7. การติดตั้งระบบทดสอบการควบคุมความดันในถังลม

**4.2 สรรพนะของระบบควบคุมความดันในถังลม**



รูปที่ 8. การปรับตั้งความดันในถังลมแบบขั้นบันได

การทดสอบปรับความดันแบบขั้นบันได ที่ความดันขาขึ้น 0 ถึง 1 และ 2 บาร์รูปที่ 8. จะพบว่าความดันในระบบสามารถวิ่งเข้าสู่เป้าหมายได้อย่างแม่นยำ โดยมีค่าผิดพลาดที่สภาวะคงที่ไม่เกิน 5% และมีค่าเวลาขอมขาขึ้น (Rise time) ไม่เกิน 12 วินาทีในช่วงเริ่มต้นความดันในถังจะมีการแกว่งไกวอย่างรุนแรงเนื่องจากความดันในถังสูงจะถ่ายเทลงเข้าสู่ถังลมที่พิจารณาอย่างรวดเร็วตั้งนั้นจึงเกิดการกระแทกของลมเกิดขึ้น

**บทความวิจัย**

การประยุกต์ใช้การเบี่ยงเบนค่าตัวแปรไม่เชิงเส้นในระบบไดนามิกแบบกระจายตัว

*Proceedings of the 11<sup>th</sup> Conference of Electrical Engineering, Network of Rajabhatguru University of Technology 2016 (RENET 2016)*

**4.3 ผลการทดลองในการปรับค่าพารามิเตอร์ของระบบ**

ในการทดสอบการควบคุมระบบไดนามิกที่มีความล่าช้าและลดความถี่ในระบบค่าพารามิเตอร์ไดนามิกของระบบที่แสดงดังรูปที่ 9.



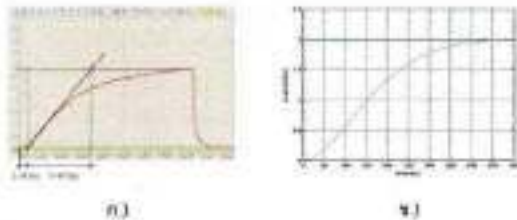
รูปที่ 9. การตอบสนองของระบบควบคุมที่มีค่าพารามิเตอร์

การทดสอบการควบคุมระบบควบคุมที่มีค่าพารามิเตอร์ไดนามิกที่มีความล่าช้าและลดความถี่ในระบบค่าพารามิเตอร์ไดนามิกของระบบที่แสดงดังรูปที่ 9. ค่าของค่าพารามิเตอร์ของระบบไดนามิกที่มีความล่าช้าและลดความถี่ในระบบค่าพารามิเตอร์ไดนามิกของระบบที่แสดงดังรูปที่ 9. ค่าของค่าพารามิเตอร์ของระบบไดนามิกที่มีความล่าช้าและลดความถี่ในระบบค่าพารามิเตอร์ไดนามิกของระบบที่แสดงดังรูปที่ 9.

**4.4 การหาพารามิเตอร์ของระบบและระบบจำลองแบบ**

จากสมการที่ (4) สามารถเขียนระบบสำหรับฟังก์ชันถ่ายโอนระบบเป็น

$$G_f(s) = \frac{1 \cdot e^{-0.5s}}{60s + 1} \quad (5)$$



รูปที่ 10. การหาพารามิเตอร์ของระบบและระบบจำลองแบบ (ก.) ผลการทดลองจริง โดยปรับค่าที่ 100% (ข.) ผลการจำลองจากสมการที่ (5)

**5. สรุปผลการทดลอง**

ในการควบคุมความถี่ในระบบแบบกระจายตัว 2 ถึงค่าที่โปรแกรมเบี่ยงเบนจากโปรแกรม สามารถสรุปได้ดังนี้

- 1. ระบบควบคุมที่สร้างขึ้นสามารถควบคุมความถี่ในระบบไดนามิกได้ตั้งแต่ 0-2 บาร์และค่าพารามิเตอร์ของระบบที่แสดงดังรูปที่ 9. ค่าของค่าพารามิเตอร์ของระบบไดนามิกที่มีความล่าช้าและลดความถี่ในระบบค่าพารามิเตอร์ไดนามิกของระบบที่แสดงดังรูปที่ 9.

2. ระบบการควบคุมความถี่แบบกระจายตัวเป็นโปรแกรมที่ใช้งานได้เป็นอย่างดีและมีข้อผิดพลาดน้อย

3. ระบบการเบี่ยงเบนค่าพารามิเตอร์ของระบบควบคุมที่แสดงดังรูปที่ 9. ค่าของค่าพารามิเตอร์ของระบบไดนามิกที่มีความล่าช้าและลดความถี่ในระบบค่าพารามิเตอร์ไดนามิกของระบบที่แสดงดังรูปที่ 9.

**6. กิตติกรรมประกาศ**

ขอขอบคุณอาจารย์ที่ปรึกษาวิทยานิพนธ์ ผศ.ดร. เสือ วัลลภวิวัฒน์ ที่ได้ให้คำแนะนำในการวิจัย ตลอดจนขอขอบคุณทีมงานในหลักสูตรระดับบัณฑิตศึกษา สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

**เอกสารอ้างอิง**

- [1] Aiping Jiang and Arthur Atan, "Response Surface Tuning Methods in Dynamic Matrix Control of a Pressure Task System", American Chemical Society-Published, 2000, 39, 3835-3843.
- [2] SanitWANGNIPARNIO and SaxonTUNYASRIBUT, "Simulation of control speed DC motor by using DSP board for electrical engineering education," International Conference on Simulation Technology (ISST2012), 27-28 September, Integrated Research Center, Kobe Univ., Japan, 454-457, 2012.
- [3] SIEMENS System Manual, "SIMATIC PID Temperature Control" Edition 1999/03.
- [4] นายศิริวิบูลย์เกษม, "การจำลองระบบการควบคุมการไหลของน้ำด้วยเทคนิคความถี่ที่ลดลงด้วยระบบการเบี่ยงเบนค่าพารามิเตอร์", ปีที่ 25, ฉบับที่ 79, หน้า 47-66, มกราคม-มีนาคม 2555.
- [5] ศศพร เสงี่ยมสิทธิ์, "การควบคุมอุณหภูมิและความเร็วเชิงสเกลาร์ในโรลเวอร์ด้วยเทคนิค", การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 35 (EECON35), 12-14 ธันวาคม 2555 ณ นครนายก, หน้า 571-574
- [6] สันติ พงศ์นิพนธ์ และ ศศพร เสงี่ยมสิทธิ์, "การควบคุมอุณหภูมิแบบปรับค่าพารามิเตอร์เบี่ยงเบนค่าพารามิเตอร์", การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 35 (EECON35), 12-14 ธันวาคม 2555 ณ นครนายก, หน้า 575-578

**ประวัติผู้เขียนบทความ**



**ชัชชา ปวงนรินทร์**  
ปัจจุบันกำลังศึกษาในระดับปริญญาโท หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

అవలంక

இவ்வாறு

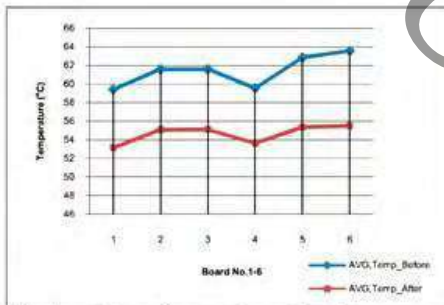


รูปที่ 3 ค่าอุณหภูมิโดยรวมที่โรงห้องเครื่องไฟฟ้า เมื่อมีการเติมอากาศด้วยพัดลมควบคุมที่การทำงานของเครื่องปรับอากาศระบบเดิม โดยตั้งเทอร์โมสแตทไว้ที่ 20 °C

**4. ผลการทดลอง**

**4.1 อุณหภูมิของบอร์ดควบคุม**

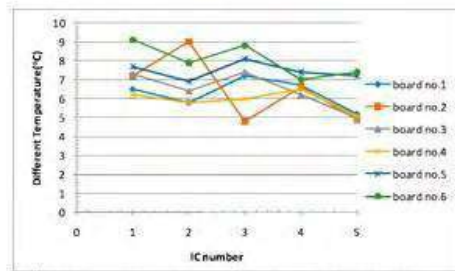
บอร์ดควบคุมซึ่งประกอบไปด้วยอุปกรณ์อิเล็กทรอนิกส์วางอยู่บนตู้ควบคุม จะมีอุณหภูมิเฉลี่ยที่บอร์ดต่างๆ ก่อนและหลังการเติมอากาศ จะแสดงดังรูปที่ 4. จากรูปที่ 4 จะพบว่าเมื่ออุณหภูมิโดยรวมภายในห้องไฟฟ้าลดลงจาก 28.3 °C เป็น 24.5 °C จะทำให้อุณหภูมิของบอร์ดควบคุมลดลงเฉลี่ย 6.91 °C หรือคิดเป็น 11.2%



รูปที่ 4 ค่าอุณหภูมิของบอร์ดควบคุม ก่อนและหลังการเติมอากาศ

**4.2 ผลต่างอุณหภูมิที่ตัว IC ของบอร์ดควบคุมต่างๆ**

ค่าความแตกต่างของอุณหภูมิที่ตัว IC 5 ตัว บนบอร์ดควบคุมต่างๆ 6 บอร์ด ก่อนและหลังจากการเติมอากาศ จะแสดงดังรูปที่ 5 จากรูปที่ 5 จะพบว่าผลต่างของอุณหภูมิที่ตัว IC 5 ตัว บนบอร์ดควบคุมต่างๆ 6 บอร์ด หลังจากการเติมอากาศ มีค่าไม่เกิน 10 °C ซึ่งอยู่ในเกณฑ์มาตรฐานของ N.E.T.A. ที่ยอมรับได้ ดังแสดงในตารางที่ 1



รูปที่ 5 ค่าความแตกต่างของอุณหภูมิที่ตัว IC บนบอร์ดการควบคุมต่างๆ หลังจากมีการเติมอากาศ

ตารางที่ 1 มาตรฐาน : N.E.T.A. Maintenance Testing Specifications for electrical equipment

	อุณหภูมิแตกต่างระหว่างอุปกรณ์เดียวกัน	อุณหภูมิแตกต่างระหว่างเวคัลลัมกับอุปกรณ์	ข้อเสนอแนะการแก้ไข
1	1 ถึง 3 °C	1 ถึง 10 °C	ยอมรับได้
2	4 ถึง 15 °C	11 ถึง 20 °C	วางแผนแก้ไข PM ครั้งต่อไป
3	-----	21 ถึง 40 °C	เฝ้าติดตามใกล้ชิด แก้ไขทันทีที่มีความพร้อม
4	มากกว่า 15 °C	มากกว่า 40 °C	แก้ไขทันที

**5. สรุปผลการทดลอง**

จากผลการทดลองการเติมอากาศเพื่อลดอุณหภูมิของอุปกรณ์อิเล็กทรอนิกส์ในตู้ควบคุมไฟฟ้า สามารถสรุปได้ดังนี้

- 1.การลดอุณหภูมิในตู้ควบคุมไฟฟ้าด้วยการเติมอากาศจากภายนอกถือว่าเป็นทางเลือกหนึ่ง ในกรณีที่ไม่สามารถที่จะติดตั้งเครื่องปรับอากาศเพิ่มเติมได้
- 2.ผลการทดลองได้อุณหภูมิแตกต่างระหว่าง อุณหภูมิเวคัลลัมกับอุปกรณ์อิเล็กทรอนิกส์ในตู้ควบคุมไฟฟ้าที่ 1-10 °C นั้นเป็นไปตามมาตรฐาน N.E.T.A. Maintenance Testing Specifications for electrical equipment ที่กำหนดอุณหภูมิของอุปกรณ์ไฟฟ้า
- 3.วิธีการที่ดีที่สุดนั้นควรที่จะมีการออกแบบขนาดเครื่องปรับอากาศให้เหมาะสมกับพื้นที่ห้อง หรือติดตั้งชุดควบคุมผลต่างอุณหภูมิเพื่อสั่งการให้พัดลมหรือเครื่องปรับอากาศที่ติดตั้งเสริมทำงาน
- 4.หากมีการติดตั้งเครื่องปรับอากาศขนาด 12,000BTU/h (ราคา 18000 บาทรวมค่าแรงติดตั้ง) เพิ่มเติม แทนการลดอุณหภูมิจาก 25 °C เป็น 20 °C และใช้พัดลมเป่าหมุนเวียน จะใช้ระยะเวลาคืนทุนประมาณ 11 เดือน

இவ்வாறு