# การออกแบบและสร้างเครื่องมือวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมบนพื้นฐานฟาวน์เดชันฟิลด์บัสโดยใช้เทคโนโลยี NI

# DESIGN AND IMPLEMENTATION OF SIMPLE DIAGNOSTIC TOOL FOR FOUNDATION FIELDBUS–BASED CONTROL VALVES USING NI TECHNOLOGY

เจษฎา เจริญวัฒนโยธิน
JETSADA JAREONWATTHANAYOTHIN

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอัตโนมัติ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2557
KMITL-2014-EN-M-060-065

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# DESIGN AND IMPLEMENTATION OF SIMPLE DIAGNOSTIC TOOL FOR FOUNDATION FIELDBUS–BASED CONTROL VALVES USING NI TECHNOLOGY

JETSADA JAREONWATTHANAYOTHIN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING IN AUTOMATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2014

KMITL-2014-EN-M-060-065

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2014
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**คณะวิศวกรรมศาสตร์**
**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**
**ใบรับรองวิทยานิพนธ์**

---

**หัวข้อวิทยานิพนธ์** การออกแบบและสร้างเครื่องมือวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมบนพื้นฐานฟาวน์เดชันฟิลด์บัสโดยใช้เทคโนโลยี NI

**Thesis Title** Design and Implementation of Simple Diagnostic Tool for Foundation Fieldbus-Based Control Valves Using NI Technology

**นักศึกษา** นายเจษฎา เจริญวัฒนโยธิน

**รหัสประจำตัว** 53613504

**ปริญญา** วิศวกรรมศาสตรมหาบัณฑิต

**สาขาวิชา** วิศวกรรมอัตโนมัติ

**อาจารย์ที่ปรึกษาวิทยานิพนธ์** รศ.ดร.ไสว พงศ์สวัสดิ์

**หมายเลขวิทยานิพนธ์** KMITL-2014-EN-M-060-065

| คณะกรรมการสอบวิทยานิพนธ์ | | ลายมือชื่อ |
|---|---|---|
| รศ.ประสิทธิ์ | จุลเสรีวงศ์ | |
| ผศ.ดร.ธีรวัฒน์ | เทพมณี | |
| รศ.ดร.จิระศักดิ์ | ชาญวุฒิธรรม | |
| รศ.ดร.อัมพวัน | จุลเสรีวงศ์ | |
| รศ.ดร.ไสว | พงศ์สวัสดิ์ | |

**วัน / เดือน/ ปี ที่สอบ** วันพุธที่ 14 พฤษภาคม พ.ศ. 2557 เวลา 09.00-11.00 น.
**สถานที่สอบ** ณ อาคาร A ชั้น 3 ห้องประชุม 5

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

**คณะวิศวกรรมศาสตร์ รับรองแล้ว**

**(ศาสตราจารย์ ดร.สุชัชวีร์ สุวรรณสวัสดิ์)**
**คณบดี คณะวิศวกรรมศาสตร์**
**วันที่ 14 พฤษภาคม พ.ศ. 2557**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| **หัวข้อวิทยานิพนธ์** | การออกแบบและสร้างเครื่องมือวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมบนพื้นฐานฟาวน์เดชันฟิลด์บัสโดยใช้เทคโนโลยี NI |
| **นักศึกษา** | นายเจษฎา เจริญวัฒนโยธิน |
| **รหัสนักศึกษา** | 53613504 |
| **ปริญญา** | วิศวกรรมศาสตรมหาบัณฑิต |
| **สาขาวิชา** | วิศวกรรมอัตโนมัติ |
| **พ.ศ.** | 2557 |
| **อาจารย์ที่ปรึกษาวิทยานิพนธ์** | รศ.ดร.ไสว พงศ์สวัสดิ์ |

## บทคัดย่อ

วิทยานิพนธ์ฉบับนี้นำเสนอการออกแบบและสร้างเครื่องมือวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมบนพื้นฐานฟาวน์เดชันฟิลด์บัส โดยอาศัยการเชื่อมต่อการสื่อสารระดับ H1 ด้วยฮาร์ดแวร์ NI USB-8486 และซอฟต์แวร์ที่ทำงานบนคอมพิวเตอร์ซึ่งพัฒนาจากโปรแกรม Labview ร่วมกับ NI FBUS Communications Manager API เพื่อความสามารถในการตรวจสอบพารามิเตอร์ ทดสอบวาล์วควบคุมและเก็บบันทึกผลการทดสอบเพื่อการเปรียบเทียบหรือวิเคราะห์ โดยเป็นการลดต้นทุนของซอฟต์แวร์ด้านบริหารจัดการเครื่องมือวัดของผู้ผลิตซึ่งมีราคาสูง ในกรณีศึกษานี้ได้ทำการพัฒนาตัวอย่างซอฟต์แวร์ที่ใช้เปรียบเทียบระยะเวลากับการเคลื่อนที่ของวาล์วควบคุมของเหลว เพื่อที่จะสามารถนำไปประยุกต์ใช้ในการวิเคราะห์ปัจจัยที่มีผลกระทบและคาดการณ์การบำรุงรักษาล่วงหน้าในระดับสูงต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| **Thesis Title** | Design and Implementation of Simple Diagnostic Tool for Foundation Fieldbus-Based Control Valves Using NI Technology |
| **Student** | Mr. Jetsada Jareonwatthanayothin |
| **Student ID.** | 53613504 |
| **Degree** | Master of Engineering |
| **Program** | Automation Engineering |
| **Year** | 2014 |
| **Thesis Advisor** | Assoc. Prof. Dr. Sawai Pongswatd |


## ABSTRACT

This thesis presents a design and implementation of simple diagnostic tool for Foundation Fieldbus-based control valves. The proposed technique using NI technology consists of the NI USB-8486 for interfacing Fieldbus H1 segments and application software developed by NI FBUS Communications Manager API and LabVIEW for checking parameter values and testing step responses. Test results can be applied not only for control valve operation analysis but also for predictive maintenance plan. Performance of the proposed diagnostic tool is confirmed through experiment results obtained from testing Yamatake SVP3000 AVP303 and EMP Fisher DVC6000f control valves.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้ด้วยคำแนะนำและคำปรึกษาจาก รศ.ดร. ไสว พงศ์สวัสดิ์ ซึ่งเป็นอาจารย์ที่ปรึกษาวิทยานิพนธ์ ข้าพเจ้ารู้สึกซาบซึ้งและขอขอบพระคุณเป็นอย่างสูงในความอนุเคราะห์ให้คำแนะนำและช่วยเหลือในทุก ๆ เรื่องที่ผ่านมา นอกจากอาจารย์จะให้คำปรึกษาในเรื่องงานวิจัยเป็นอย่างดีแล้ว อาจารย์ยังให้โอกาสและคำปรึกษาเรื่องงานต่างๆ แก่ข้าเจ้าอยู่เสมอ

ขอขอบคุณห้องสมุด คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่เป็นแหล่งค้นคว้าข้อมูล

สุดท้ายนี้ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ที่ให้ลูกได้เกิดมา ให้คำสั่งสอนและอบรมให้ลูกเป็นคนดี จนกระทั่งจบปริญญาโทและคอยให้กำลังใจลูกเสมอมา

คุณค่าประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบคุณค่าต่าง ๆ เหล่านี้แด่ผู้มีพระคุณทุกท่าน ข้าพเจ้าหวังเป็นอย่างยิ่งว่าวิทยานิพนธ์ฉบับนี้จะเป็นประโยชน์แก่ผู้ที่ได้ทำการศึกษาเพื่อที่จะนำไปประยุกต์ใช้ในงานวิจัยด้านต่าง ๆ

เจษฎา เจริญวัฒนโยธิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I
บทคัดย่อภาษาอังกฤษ.....II
กิตติกรรมประกาศ.....III
สารบัญ.....IV
สารบัญตาราง.....VI
สารบัญภาพ.....VII
บทที่ 1 บทนำ.....1
1.1 ความเป็นมาและความสำคัญของปัญหา.....1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....1
1.3 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย.....1
1.4 ขอบเขตการวิจัย.....2
1.5 รายละเอียดของวิทยานิพนธ์.....2

บทที่ 2 ทฤษฎีพื้นฐานที่ใช้ในการวิจัย.....3
2.1 กล่าวนำ.....3
2.2 ระดับชั้นของฟาวน์เดชันฟิลด์บัส.....3
2.2.1 Physical Layer.....5
2.2.2 Communication Stack.....6
2.2.3 User Layer.....9
2.3 วาล์วควบคุม.....10
2.3.1 วาล์วควบคุมที่มีการสื่อสารแบบฟาวน์เดชันฟิลด์บัส.....10
2.3.2 หลักการในการวินิจฉัยวาล์วเบื้องต้น.....12
2.4 เทคโนโลยีของ National Instrument ที่ใช้ในงานวิจัย.....12
2.4.1 NI USB-8486.....13
2.4.2 NI-FBUS Communications Manager.....13
2.4.3 LabVIEW.....14
2.4.4 การจัดเก็บข้อมูล.....15

บทที่ 3 การออกแบบและการสร้างเครื่องมือวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมบนพื้นฐานฟาวน์เดชันฟิลด์บัสที่นำเสนอ.....17
3.1 กล่าวนำ.....17
3.2 ส่วนของฮาร์ดแวร์.....17
3.3 ส่วนของซอฟต์แวร์.....18
3.3.1 Communication Manager API.....18
3.3.2 ส่วนของโปรแกรมในการติดต่อกับผู้ใช้งาน.....22

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ห้ามมิให้ทำซ้ำ ดัดแปลง หรือนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

หน้า

3.3.3 การนำข้อมูลมาแสดงผลและเก็บข้อมูล....23
3.3.4 ภาพรวมของการทำงานของซอฟต์แวร์....24

บทที่ 4 การทดลองและผลการทดลอง....32
4.1 กล่าวนำ....32
4.2 ขั้นตอนการทดสอบกับวาล์วควบคุมฟาวน์เดชันฟิลด์บัส....33
4.3 ผลการทดลองกับวาล์วควบคุม Yamatake SVP3000 AVP303....37
4.4 ผลการทดลองกับวาล์วควบคุม EPM Fisher DVC6000f....38
4.5 อ่านไฟล์ที่บันทึกจากฟังก์ชัน Step Response Data....40

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ....41
5.1 สรุปผลการวิจัย....41
5.2 ข้อเสนอแนะและแนวทางการทำวิจัยต่อ....41

เอกสารอ้างอิง....42

ภาคผนวก....43

ประวัติผู้เขียน....51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง

ตารางที่ หน้า
2.1 ประเภทและหน้าที่ของ VCR....8
2.2 Function Block มาตรฐานสำหรับการควบคุม....10
3.1 รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Open Session....19
3.2 รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Browse Device List....19
3.3 รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Get Block Tag....20
3.4 รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Function Write....21
3.5 รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Function Read....21
3.6 รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Close Session....22
3.7 รูปแบบการเก็บค่าจากการทดสอบ....24
3.8 รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Open TDMS....29
3.9 รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ TDMS Write....29
3.10 รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ TDMS Close....30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญภาพ


ภาพที่ หน้า

2.1 สถาปัตยกรรมระบบการควบคุมด้วยระบบดิจิตอล....3
2.2 โมเดล OSI โมเดลเทียบกับโมเดลของฟาวน์เดชันฟิลด์บัส....4
2.3 แพ็คเก็ตการถ่ายโอนข้อมูลของฟาวน์เดชันฟิลด์บัส....4
2.4 การเชื่อมต่อสัญญาณฟาวน์เดชันฟิลด์บัส....5
2.5 การเข้ารหัสแบบ Manchester....5
2.6 การส่งถ่ายข้อมูลแบบ Scheduled....6
2.7 การส่งถ่ายข้อมูลแบบ Unscheduled....7
2.8 สรุปประเภทของและหน้าที่ของ VFD....8
2.9 ประเภทของ Block ใน User Layer....9
2.10 Function Block Parameters....11
2.11 การทำงานของเทคโนโลยีจาก National Instrument ในการติดต่อวาล์วควบคุม....12
2.12 อุปกรณ์เชื่อมต่อเครือข่ายฟาวน์เดชันฟิลด์บัส H1 รุ่น NI USB-8486....13
2.13 หัวต่อและสัญญาณแบบ DB 9 ที่ใช้ใน NI USB-8486....13
2.14 การพัฒนาโปรแกรมประยุกต์บน LabVIEW....15
2.15 ไฟล์ TDMS ประกอบไปด้วยข้อมูลรายละเอียดของไฟล์แต่ละระดับ....16
3.1 การติดตั้ง USB-8486 เข้ากับระบบ....17
3.2 การทำงานโดยทั่วไปของ Communication Manager API....18
3.3 ฟังก์ชัน Open Session....19
3.4 ฟังก์ชัน Browse Device List....19
3.5 ฟังก์ชัน Get Block Tag....20
3.6 ฟังก์ชัน Write....20
3.7 ฟังก์ชัน Read....21
3.8 ฟังก์ชัน Close Session....21
3.9 ส่วนติดต่อกับผู้ใช้เพื่อกำหนดค่าและทดสอบ....22
3.10 ตารางสำหรับใส่ข้อมูลเพื่อทำการทดสอบตามผู้ใช้กำหนด....23
3.11 ส่วนติดต่อกับผู้ใช้งานสำหรับแสดงเปอร์เซ็นต์การเปิดวาล์วเปรียบเทียบกับเวลา....24
3.12 ส่วนแรกของการพัฒนาซอฟท์แวร์....25
3.13 Open Session และ Browse Device List เพื่อการแสดงชื่อของอุปกรณ์....25
3.14 การเรียก Block Tag List ของอุปกรณ์มาแสดงผล....25
3.15 ภาพรวมส่วนของการเริ่มต้นในโปรแกรม....25
3.16 การทำงานส่วนที่สองซึ่งมีการทำงานซ้ำ....26
3.17 การเขียนข้อมูลสั่งงานวาล์วเคลื่อนที่....26
3.18 การส่งค่าให้กับวาล์วควบคุม....27
3.19 การอ่านค่าตำแหน่งของวาล์ว....27
3.20 การอ่านค่าจากวาล์วควบคุม....28



เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญภาพ (ต่อ)


ภาพที่ หน้า

3.21 ขั้นตอนการจัดเก็บข้อมูล....28
3.22 ฟังก์ชัน TDMS Open....28
3.23 ฟังก์ชัน TDMS Write....29
3.24 ฟังก์ชัน TDMS Close....30
3.25 การจัดเก็บข้อมูลในโปรแกรม....31
3.26 การทำงานส่วนสุดท้ายของการพัฒนาซอฟต์แวร์....31
3.27 การปิดเซสชั่นและการตรวจสอบข้อผิดพลาดของโปรแกรม....31
4.1 ระบบฟาวน์เดชันฟิลด์บัสและวาล์วควบคุม Yamatake ในกระบวนการควบคุมระดับ....32
4.2 ระบบฟาวน์เดชันฟิลด์บัสและวาล์วควบคุม EPM Fisher ในกระบวนการควบคุมระดับ....33
4.3 เมนูบล็อกอุปกรณ์และหน้าการปรับโหมด....33
4.4 NI-FBUS Communications Manager....34
4.5 ส่วนของโปรแกรมวินิจฉัยหลังจากการโหลดรายชื่ออุปกรณ์....34
4.6 โปรแกรมทดสอบผลตอบสนองขั้นบันไดของวาล์วควบคุม....35
4.7 โปรแกรมขณะเริ่มทำการทดสอบ....36
4.8 โปรแกรมทำการควบคุมสั่งเปิดวาล์วที่ 100%....36
4.9 ไฟล์ข้อมูลจากการทดสอบเปิดด้วย Microsoft Excel....37
4.10 ความสัมพันธ์ผลตอบสนองขั้นบันไดของวาล์ว Yamatake SVP3000 AVP303....37
4.11 ความสัมพันธ์ผลตอบสนองขั้นบันไดของวาล์ว Yamatake SVP3000 AVP303 ครั้งที่สอง....38
4.12 ความสัมพันธ์แสดงผลตอบสนองขั้นบันไดของวาล์ว EPM Fisher DVC6000f....39
4.13 ความสัมพันธ์ผลตอบสนองขั้นบันไดของวาล์ว EPM Fisher DVC6000f ครั้งที่สอง....39
4.14 เปิดข้อมูลจากการทดสอบผ่าน Step Response Data....40



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1
# บทนำ

## 1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันรูปแบบโครงสร้างของการสื่อสารได้เปลี่ยนไปจากระบบควบคุมที่มีตัวประมวลผลเป็นจุดศูนย์กลางเพียงตัวเดียวไปเป็นการประมวลผลเป็นกระจายส่วน [1] โดยมีเทคโนโลยีฟาวน์เดชันฟิลด์บัส เป็นเทคโนโลยีการสื่อสารแบบดิจิตอลสื่อสารกันระหว่างตัวอุปกรณ์อัจฉริยะ [2] ที่เป็นแบบระบบเปิดในระบบอุตสาหกรรม นำมาใช้แทนการสื่อสารแบบเก่าที่ใช้สัญญาณอนาลอกมาตรฐาน 4-20 mA เพื่อเพิ่มประสิทธิภาพในการสื่อสารข้อมูลระหว่างอุปกรณ์ควบคุมและเครื่องมือวัด การส่งค่าที่ได้จากการวัดในกระบวนการ ค่าตัวแปรที่อยู่ในเครื่องมือวัด รวมถึงเพิ่มความสามารถในการส่งข้อมูลที่ใช้ในการวินิจฉัยความผิดปกติเข้ามาซึ่งไม่มีในการสื่อสารแบบเก่า [3] โดยทั่วไปแล้วเครื่องมือวัดที่ทำการสื่อสารด้วยเทคโนโลยีฟาวน์เดชันฟิลด์บัส จะมีหน่วยประมวลผลและหน่วยความจำอยู่ภายในทำหน้าที่ในการประมวลผลข้อมูลที่ได้รับจากเครื่องมือวัดอื่นในระบบ รวมทั้งส่งสัญญาณควบคุมและจัดการกับค่าพารามิเตอร์ในตัวอุปกรณ์ โดยความได้เปรียบที่มีหน่วยประมวลผลอยู่ภายในตัวอุปกรณ์ ทำให้วาล์วควบคุมที่สื่อสารด้วยเทคโนโลยีฟาวน์เดชันฟิลด์บัส [4]-[5] มีความสามารถที่จะตรวจสอบความผิดปกติเบื้องต้นได้ โดยที่ไม่จำเป็นต้องถอดออกจากกระบวนการ แต่เนื่องจากอุปกรณ์ที่ใช้ฟาวน์เดชันฟิลด์บัสเป็นการสื่อสารแบบดิจิตอล ทำให้ปกติแล้วการปรับเปลี่ยนค่าพารามิเตอร์ การทดสอบวาล์วควบคุม การเก็บข้อมูลจากวาล์วควบคุมเพื่อการวินิจฉัยประสิทธิภาพในการทำงานรวมถึงการคาดการณ์ระยะเวลาบำรุงรักษาล่วงหน้า สามารถทำได้ผ่านตัวควบคุมและซอฟต์แวร์ที่มากับผู้ผลิตเท่านั้น ซึ่งมีราคาสูง และใช้ได้เฉพาะผลิตภัณฑ์ของบริษัทนั้น ๆ [6]

## 1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

ออกแบบและพัฒนาซอฟต์แวร์เพื่อติดต่อวาล์วควบคุมที่ใช้เทคโนโลยีการสื่อสารแบบฟาวน์เดชันฟิลด์บัสในระดับการสื่อสารแบบ H1 และรวมถึงการเก็บค่าตอบสนองขั้นบันไดของวาล์วในกระบวนการ เพื่อแสดงให้เห็นถึงแนวทางในการพัฒนาโปรแกรมประยุกต์ขึ้นมาใช้แทนหรือใช้ร่วมกับซอฟต์แวร์ของผู้ผลิตและเพื่อแสดงถึงประโยชน์จากการใช้เทคโนโลยีฟาวน์เดชันฟิลด์บัสในกระบวนการ

## 1.3 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

ฟาวน์เดชันฟิลด์บัสเป็นเทคโนโลยีที่ใช้การสื่อสารแบบดิจิตอลที่สื่อสารโดยรูปแบบโมเดลที่กำหนดโดย OSI เป็นโมเดลในการสื่อสารระหว่างอุปกรณ์ที่อยู่บนระบบ โดยประกอบไปด้วย Physical Layer Communication Stack และ User Application Layer การติดต่อสื่อสารกับตัวอุปกรณ์จะต้องผ่านการเชื่อมต่อเพื่อสื่อสารในระดับ H1 ด้วยฮาร์ดแวร์จากบริษัท National Instrument NI USB-8486 ซึ่งทำหน้าที่ในส่วน Communication Stack และซอฟต์แวร์ที่ทำงานบนคอมพิวเตอร์ซึ่งพัฒนาจากโปรแกรม LabVIEW ทำเป็นส่วนติดต่อผู้ใช้งานและสื่อสารกับ User Application Layer ร่วมกับ NI FBUS Communications Manager API เป็นตัวจัดการการสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระหว่าง Communication Stack กับ User Application Layer เพื่อความสามารถในการตรวจสอบพารามิเตอร์ ทดสอบวาล์วควบคุมและเก็บบันทึกผลการทดสอบเพื่อการเปรียบเทียบในภายหลัง โดยไม่ขึ้นกับตัวควบคุมและเป็นการลดต้นทุนของซอฟต์แวร์ด้านบริหารจัดการเครื่องมือวัดของผู้ผลิตซึ่งมีราคาสูง

## 1.4 ขอบเขตการวิจัย

ในวิทยานิพนธ์ฉบับนี้ได้นำเสนอการพัฒนาโปรแกรมเพื่อเป็นเครื่องมือต้นแบบในการวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมที่ทำการสื่อสารด้วยโปรโตคอลฟาวน์เดชันฟิลด์บัส ผ่าน NI USB-8486 และคอมพิวเตอร์ที่ติดตั้งซอฟต์แวร์ที่พัฒนาขึ้น เข้ากับระบบสื่อสารในระดับ H1 ของกระบวนการควบคุมระดับ เพื่อให้ผู้ใช้งานสามารถกำหนดค่าในการใช้งาน การทดสอบผลตอบสนองขั้นบันไดการเคลื่อนที่ของวาล์วควบคุมตามที่ต้องการและเก็บข้อมูลเพื่อการเปรียบเทียบผลตอบสนองในภายหลัง นอกจากนี้ระบบและโปรแกรมที่พัฒนาขึ้นสามารถนำไปใช้กับวาล์วควบคุมฟาวน์เดชันฟิลด์บัสของทุกบริษัทผู้ผลิต และเป็นแนวทางในการพัฒนาเครื่องมือสำหรับการวินิจฉัยวาล์วควบคุมเพื่อการวางแผนในการซ่อมบำรุง และการบริหารระบบโดยรวมต่อไป

## 1.5 รายละเอียดของวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกัน

บทที่ 1 กล่าวถึง ความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ ทฤษฎีหรือแนวคิดที่ใช้ในงานวิจัย ขอบเขตของงานวิจัย และรายละเอียดของวิทยานิพนธ์

บทที่ 2 กล่าวถึง ทฤษฎีพื้นฐานที่ใช้ในงานวิจัย ฟาวน์เดชันฟิลด์บัส User Layer ฟาวน์เดชันฟิลด์บัส Communication Stack และฟาวน์เดชันฟิลด์บัส Physical Layer เทคโนโลยีที่ใช้ในงานวิจัย

บทที่ 3 กล่าวถึง การออกแบบและการสร้างเครื่องมือวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมบนพื้นฐานฟาวน์เดชันฟิลด์บัสโดยใช้เทคโนโลยี NI ทั้งส่วนฮาร์ดแวร์และส่วนของซอฟต์แวร์

บทที่ 4 นำเสนอการทดลองและผลการทดลอง ด้วยวาล์ว Yamatake SVP3000 AVP303 และวาล์ว EPM Fisher DVC6000f

บทที่ 5 เป็นบทสรุปผลการวิจัยและข้อเสนอแนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

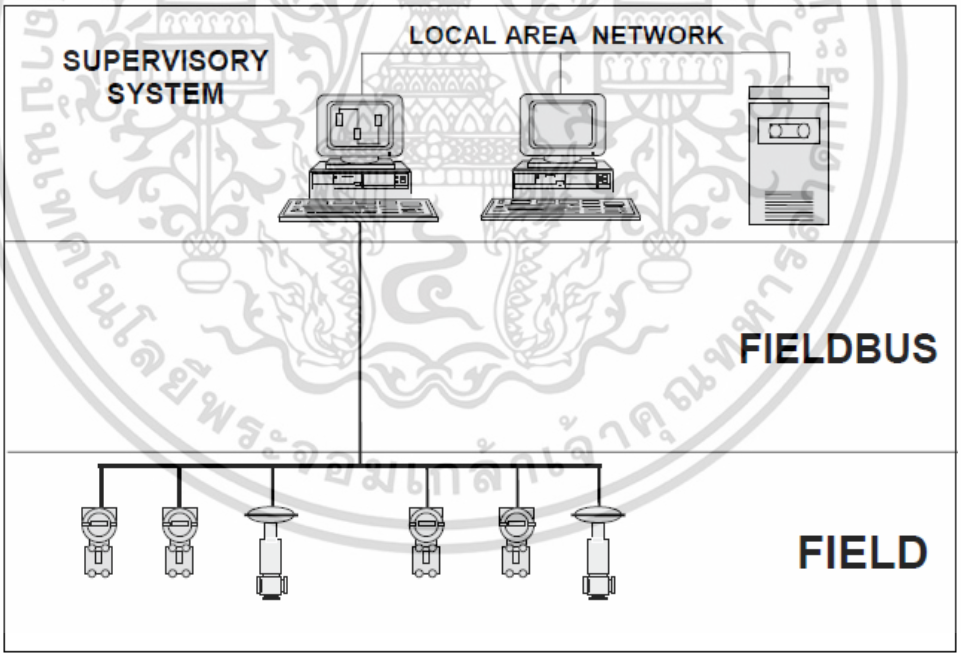
# บทที่ 2
# ทฤษฎีพื้นฐานที่ใช้ในการวิจัย

## 2.1 กล่าวนำ

ในบทนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องในการวิจัย โดยนำเสนอทฤษฎีเกี่ยวกับ เทคโนโลยีฟาวน์เดชันฟิลด์บัส ระดับชั้น (Layer) ของฟาวน์เดชันฟิลด์บัส วาล์วควบคุมฟาวน์เดชันฟิลด์บัส บล็อกของวาล์วควบคุมฟาวน์เดชันฟิลด์บัสและวิธีการตรวจสอบวาล์ว ชุดเครื่องมือที่ใช้เชื่อมต่อกับเครือข่ายฟาวน์เดชั่นฟิลด์บัส ซอฟท์แวร์ที่ใช้พัฒนาส่วนควบคุมและแสดงผล

## 2.2 ระดับชั้นของฟาวน์เดชันฟิลด์บัส

เทคโนโลยีฟาวน์เดชันฟิลด์บัสเป็นการสื่อสารแบบดิจิตอลสองทิศทาง มีการเชื่อมต่อทางกายภาพแบบ Multi-Drop ซึ่งทำให้อุปกรณ์และกระบวนการและ HMI เชื่อมต่อสื่อสารกันได้ รวมทั้งอุปกรณ์ยังมีความสามารถในการวัด และการควบคุมภายในตัวเอง นอกจากนี้ยังสามารถที่จะสนับสนุนการเชื่อมต่อกับระบบ LAN สำหรับกระบวนการที่มีความซับซ้อน ดังภาพที่ 2.1 ซึ่งแสดงให้เห็นความสัมพันธ์ของอุปกรณ์และฟิลด์บัสและระบบควบคุมดูแล

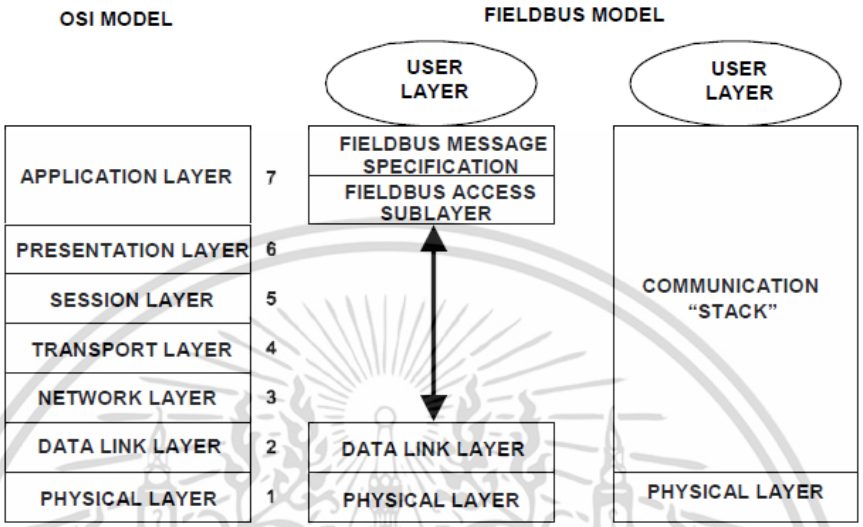



**ภาพที่ 2.1** สถาปัตยกรรมระบบการควบคุมด้วยระบบดิจิตอล

ฟาวน์เดชันฟิลด์บัสอยู่บนพื้นฐานของ 7 ระดับชั้นการสื่อสารของโมเดล OSI แต่ไม่ได้ใช้งานระดับชั้นของโมเดล OSI ทั้งหมด จะมี Physical Layer ซึ่งเป็นส่วนที่เกี่ยวกับเทคนิคในการเชื่อมต่ออุปกรณ์และ Communication Stack ที่เกี่ยวกับการสื่อสารแบบดิจิตอลระหว่างตัวอุปกรณ์ เป็นสองส่วนที่มี

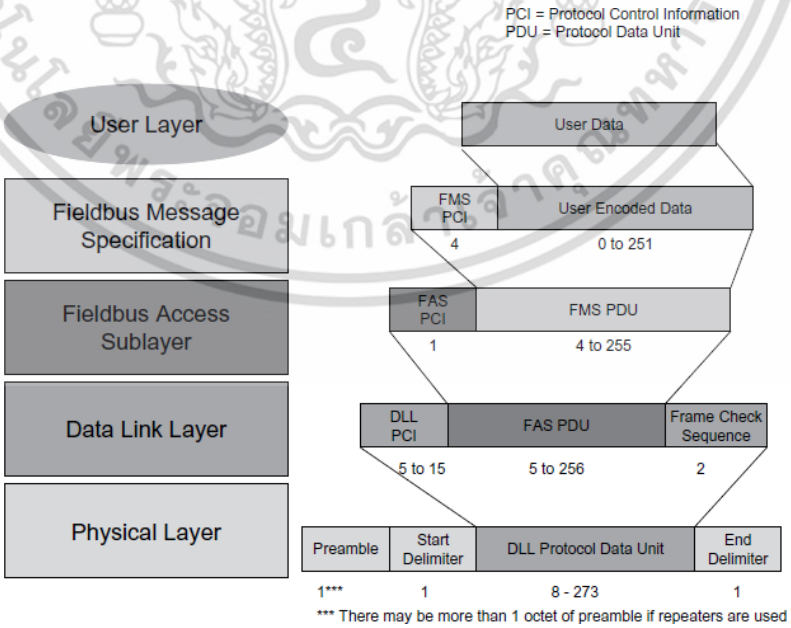
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน จากภาพที่ 2.2 จะพบว่าระดับชั้นที่ 3-6 จะไม่ถูกใช้งานในระบบฟาวน์เดชันฟิลด์บัส และ Fieldbus Message Specification และ Fieldbus Access Sublayer เป็นส่วนหนึ่งของ Application Layer และจะมีส่วนติดต่อกับผู้ใช้งาน (User Application) เพิ่มขึ้นมาอยู่เหนือ Application Layer ในระบบฟาวน์เดชันฟิลด์บัส




**ภาพที่ 2.2** โมเดล OSI โมเดลเทียบกับโมเดลของฟาวน์เดชันฟิลด์บัส

ข้อมูลหรือข้อความที่ถูกส่งจากอุปกรณ์ตัวหนึ่งไปยังอุปกรณ์ตัวอื่นในเครือข่ายฟาวน์เดชันฟิลด์บัส จะต้องผ่านระดับ OSI ทั้งหมดและขั้นตอนการแพ็คเก็ตข้อมูลตามที่แสดงดังภาพที่ 2.3 ซึ่งสามารถแสดงรายละเอียดของแต่ละระดับชั้นได้ดังนี้

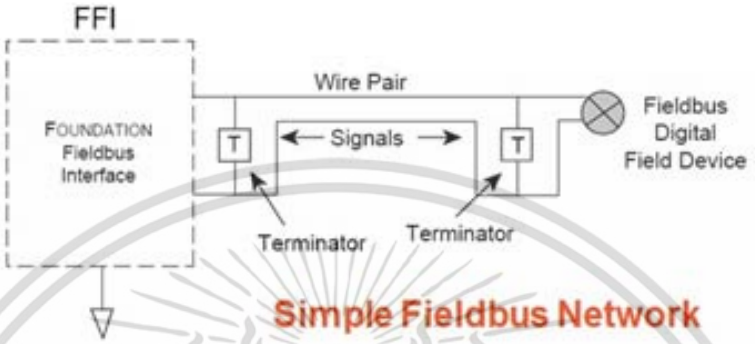



**ภาพที่ 2.3** แพ็คเก็ตการถ่ายโอนข้อมูลของฟาวน์เดชันฟิลด์บัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

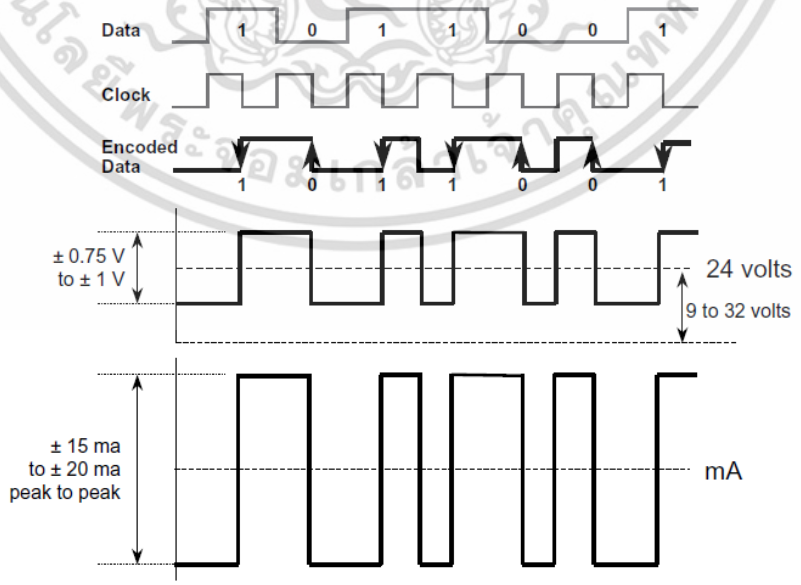
### 2.2.1 Physical Layer

ในชั้นนี้จะเป็นโครงสร้างของระบบฟาวน์เดชันฟิลด์บัส และเป็นส่วนที่เชื่อมต่อระหว่างอุปกรณ์ทุกตัวในระบบเข้าด้วยกัน เพื่อใช้เป็นเส้นทางเดินของสัญญาณ โดยทั่วไปจะใช้สายสัญญาณแบบปกติเหมือนกับระบบควบคุมทั่ว ๆ ไป ซึ่งจะถูกกำหนดโดย IEC และ ISA ในการเชื่อมต่อกันทางกายภาพ อุปกรณ์ฟาวน์เดชันฟิลด์บัสสามารถแสดงได้ดังภาพที่ 2.4




**ภาพที่ 2.4** การเชื่อมต่อสัญญาณฟาวน์เดชันฟิลด์บัส

เครือข่ายฟาวน์เดชันฟิลด์บัส H1 ใช้การเข้ารหัสแบบ Manchester สัญญาณนี้จะเรียกว่า Synchronous Serial เพราะมีสัญญาณ Clock ฝังอยู่ในข้อมูลอนุกรม ข้อมูลจะถูกรวมเข้ากับสัญญาณ Clock เพื่อสร้างเป็นสัญญาณฟิลด์บัส ซึ่งการเข้ารหัสแบบนี้มีการแทนลอจิก 0 ด้วยสัญญาณที่เปลี่ยนจาก 1 เป็น 0 บริเวณกึ่งกลางของสัญญาณ Clock และลอจิก 1 ด้วยสัญญาณที่เปลี่ยนจาก 0 เป็น 1 บริเวณกึ่งกลางของสัญญาณ Clock ดังภาพที่ 2.5




**ภาพที่ 2.5** การเข้ารหัสแบบ Manchester

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.2 Communication Stack

เป็นส่วนที่ใช้ในการสื่อสารระหว่างชั้นของ Physical Layer และ User Application โดยจะประกอบด้วย Data Link Layer และ Application Layer ซึ่งแบ่งออกเป็นส่วนสำคัญ 3 ส่วนดังนี้

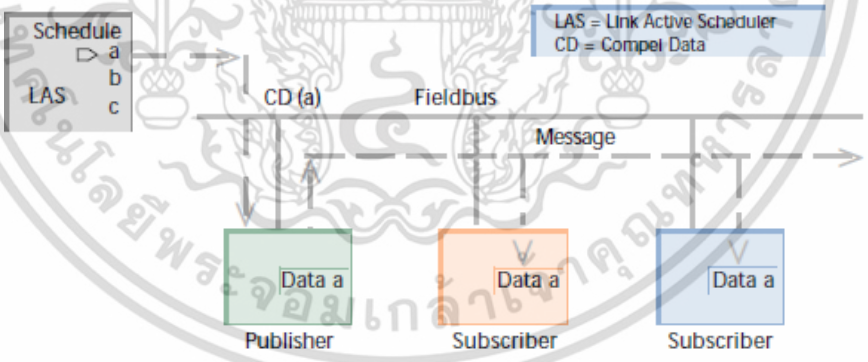
Data Link Layer (DLL) มีหน้าที่ควบคุมการรับส่งข้อมูลหรือข้อความในระบบฟาวน์เดชันฟิลด์บัสโดยผ่านการกำหนดตามตารางเวลาจากส่วนกลางที่เรียกว่า Link Active Scheduler หรือ LAS ซึ่ง DLL แบ่งอุปกรณ์ออกเป็นสามประเภทคือ

Basic Device มีคุณสมบัติในการที่จะทำหน้าที่เป็นเครื่องมือวัดและตัวควบคุมในกระบวนการ

Bridges เป็นอุปกรณ์ที่ใช้ในการขยายโครงข่ายของระบบ

Link Master มีคุณสมบัติที่สามารถเป็นตัวควบคุมตารางเวลาการทำงานของระบบได้ หรือที่เรียกว่า LAS มีหน้าที่ในการจัดการการสื่อสารในระบบฟาวน์เดชันฟิลด์บัส โดยแบ่งการสื่อสารออกเป็นสองแบบคือ

Scheduled Communication เป็นการสื่อสารที่มีการกำหนดให้ข้อมูลต่าง ๆ ที่จะรับส่งในระบบมีเวลาในการรับส่งที่แน่นอน ซึ่งโดยทั่วไปแล้วข้อมูลเหล่านี้จะเป็นตัวแปรที่มีความสำคัญในการทำงานของระบบ เช่น ตัวแปรทางกระบวนการผลิต ตัวแปรในการควบคุมกระบวนการผลิตจากผู้ใช้งาน สัญญาณเตือนจากกระบวนการผลิตที่สำคัญ หรืออาจทำให้กระบวนการผลิตหยุดทำงาน เป็นต้น โดยที่รูปแบบการสื่อสารแบบ Scheduled นั้น LAS จะมีตารางของเวลาสำหรับบัฟเฟอร์ข้อมูลในอุปกรณ์ทั้งหมดที่ต้องทำส่งข้อมูลถึงกันในระบบฟาวน์เดชันฟิลด์บัส เมื่อถึงเวลาของอุปกรณ์ที่ต้องส่งข้อมูลในบัฟเฟอร์ LAS จะส่ง Compel Data (CD) ไปยังอุปกรณ์นั้น เมื่อได้รับ CD อุปกรณ์ตัวนั้นจะทำการปล่อยข้อมูลในบัฟเฟอร์ไปยังอุปกรณ์ทั้งหมดที่อยู่ในระบบหรือที่เรียกว่า Publishes อุปกรณ์ตัวไหนที่ถูกกำหนดให้รับค่าจะเรียกว่า Subscriber ดังภาพที่ 2.6

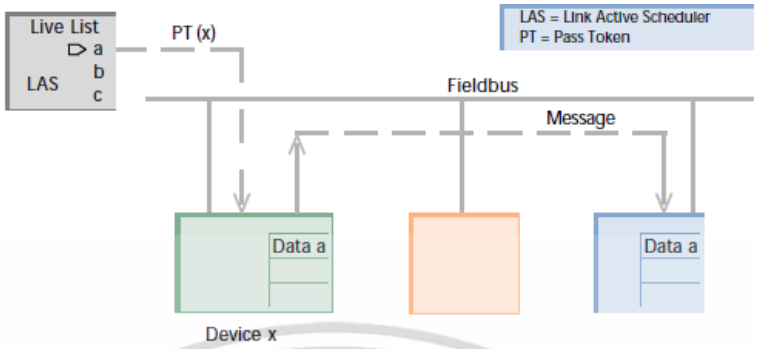



**ภาพที่ 2.6** การส่งถ่ายข้อมูลแบบ Scheduled

Unscheduled Communication เป็นการสื่อสารที่ไม่มีตัวกำหนดให้ข้อมูลต่าง ๆ ที่จะรับส่งในระบบมีเวลาในการรับส่งที่แน่นอน ซึ่งทั่วไปแล้วข้อมูลเหล่านี้ จะเป็นตัวแปรต่างๆ ของอุปกรณ์การวัด สัญญาณเตือนจากกระบวนการผลิต เป็นต้น โดยทั่วไปนั้นอุปกรณ์ทุกตัวบนเครือข่ายฟาวน์เดชันฟิลด์บัสมีโอกาสจะส่งข้อมูลที่เป็นแบบไม่มีการกำหนดเวลาในการส่งที่แน่นอนหรือ Unscheduled ได้ระหว่างการรับส่งข้อมูลแบบ Scheduled โดย LAS จะให้สิทธิ์กับอุปกรณ์ด้วยการส่ง Pass Token (PT) ให้กับอุปกรณ์ เมื่ออุปกรณ์นั้นได้รับ PT จะได้รับอนุญาตให้ส่งข้อมูลได้จนเสร็จหรือจนกระทั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวลาสูงสุดในการถือ PT หมดไป ขึ้นอยู่กับว่าเวลาของกระบวนการไหนสั้นกว่ากัน ภาพที่ 2.7 ข้อมูลในบัฟเฟอร์จะถูกส่งเมื่อ LAS ส่ง PT ให้กับอุปกรณ์ x




**ภาพที่ 2.7** การส่งถ่ายข้อมูลแบบ Unscheduled

Fieldbus Access Sublayer (FAS) ใช้การสื่อสาร Scheduled และ Unscheduled ใน Data Link Layer เพื่อที่เตรียมข้อมูลสำหรับ Fieldbus Message Specification (FMS) รูปแบบของการจัดเตรียมข้อมูลจะถูกอธิบายด้วย Virtual Communication Relationship (VCR) ซึ่งเป็นเหมือนกับฟังก์ชันโทรด่วนบนโทรศัพท์มือถือ ตัวอย่างเช่น ถ้าต้องการที่จะโทรไปยังต่างประเทศจะต้องใช้ตัวเลขหลายหลักในการโทร เช่น รหัสการโทรออกต่างประเทศ รหัสประเทศ รหัสจังหวัด รหัสชุมสายโทรศัพท์แล้วตามด้วยเบอร์โทรศัพท์ รายละเอียดทั้งหมดนี้ต้องการใส่ให้ครบเพียงแค่ครั้งเดียวแล้วตั้งค่าไว้ให้เป็นเบอร์โทรด่วน หลังจากนั้นการโทรออกไปยังต่างประเทศก็สามารถทำได้ด้วยการกดที่เบอร์โทรออกเท่านั้น เช่นเดียวกันหลังจากที่ตั้งค่าแล้ว ก็ใช้แค่เลข VCR ในการสื่อสารกับอุปกรณ์อื่นในระบบฟาวน์เดชันฟิลด์บัส โดยประเภทของ VCR มีดังต่อไปนี้

Client/Server เป็นการสื่อสารแบบ Unscheduled และเป็นการสื่อสารแบบหนึ่งต่อหนึ่ง เมื่ออุปกรณ์ได้รับ PT จาก LAS แล้วจะสามารถส่งการร้องขอข้อมูลไปให้อุปกรณ์อื่นบนบัสได้ อุปกรณ์ที่ร้องขอจะเรียกว่า Client และอุปกรณ์อื่นที่ถูกร้องขอจะเรียกว่า Server อุปกรณ์ที่เป็น Server สามารถส่งข้อมูลตอบกลับได้เมื่อได้รับ PT ซึ่งทั่วไปแล้ว VCR รูปแบบนี้จะถูกใช้โดยผู้ดูแลระบบ เช่น เปลี่ยนค่าที่ต้องการ (Set Point) ปรับและเปลี่ยนค่าพารามิเตอร์ รับทราบสัญญาณเตือน อัพโหลดและดาวน์โหลดอุปกรณ์

Report Distribution เป็นการสื่อสารแบบ Unscheduled และเป็นการสื่อสารแบบหนึ่งต่อกลุ่ม เมื่ออุปกรณ์ที่มีเหตุการณ์เกิดขึ้นหรือมีรายงานได้รับ PT จาก LAS จะส่งข้อมูลไปยัง Group Address ที่กำหนดไว้ อุปกรณ์ที่ถูกกำหนดให้รอรับข้อมูลจาก VCR นั้นก็จะได้รับรายงาน ซึ่งทั่วไปแล้วจะถูกใช้โดยอุปกรณ์ฟาวน์เดชันฟิลด์บัสในการส่งสัญญาณแจ้งเตือนมายังผู้ดูปฏิบัติงาน

Publisher and Subscriber เป็นการสื่อสารแบบ Scheduled และเป็นการสื่อสารแบบหนึ่งต่อกลุ่ม เมื่ออุปกรณ์ได้รับ CD จะทำการ Publish หรือกระจายข้อมูลไปยังทุกอุปกรณ์ในระบบ อุปกรณ์ที่รับข้อมูลที่ส่งออกมาเรียกว่า Subscribers ซึ่งโดยทั่วไปแล้วจะถูกใช้โดยอุปกรณ์ฟาวน์เดชันฟิลด์บัสสำหรับการส่งค่าอินพุตและเอาต์พุตฟังก์ชันบล็อกของ User Application เช่น ค่าตัวแปรของกระบวนการ (PV) และ เอาต์พุต (OUT)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

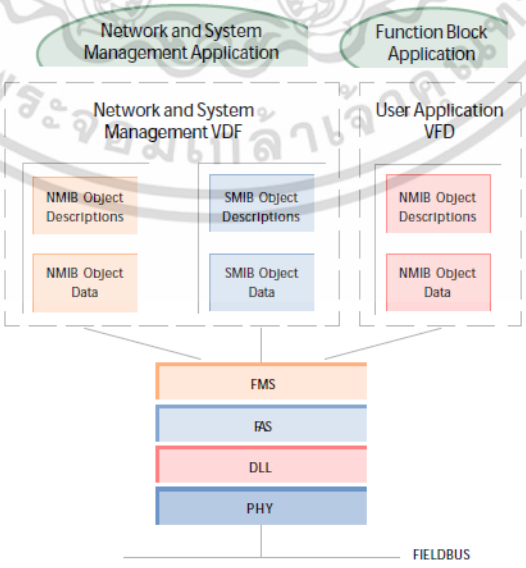
**ตารางที่ 2.1** ประเภทและหน้าที่ของ VCR

| Client/Server | Report Distribuiltion | Publisher/Subscriber |
|---|---|---|
| Used for Operator Messages | Used for Event Notifications and Trend Reports | Used for Publishing Data |
| - Setpoint<br>- Mode changes<br>- Tuning changes<br>- Upload/Dwnload<br>- Alarm Management<br>- Access display views<br>- Remote diagnostics | - Send process alarms to operator consoles<br>- Send trend reports to data historians | - Send Transmitter PV to PID control block |

Fieldbus Message Specification (FMS)

เป็นส่วนที่สร้างข้อมูลให้อยู่ในรูปแบบมาตรฐานและมีส่วนของโปรโตคอลที่จำเป็นให้กับ User Application ทำให้ User Application สามารถส่งข้อมูลหากันได้ในระบบฟาวน์เดชันฟิลด์บัส ข้อมูลที่สื่อสารกันผ่านฟาวน์เดชันฟิลด์บัสสามารถจะถูกอธิบายได้ด้วยลักษณะของวัตถุ (Object Description) ซึ่งลักษณะของวัตถุนี้ได้เก็บรวบรวมไว้ในโครงสร้างที่เรียกว่า Object Dictionary หรือ OD โดย Virtual Field Device (VFD) จะเรียกดู OD นี้เพื่อที่จะรู้ถึงชนิดข้อมูลเพื่อการตีความข้อมูลได้ ซึ่ง VFD จะมีอยู่สองประเภท ดังภาพที่ 2.8 คือ VFD ที่เกี่ยวกับเครือข่ายและระบบ (NMIB/SMIB) และ VFD ที่เกี่ยวกับฟังก์ชันบล็อกแอพพลิเคชั่น (FBAP)

Virtual Field Device (VFD) คือโมเดลสำหรับการเรียกดูข้อมูลที่อยู่ใน OD ด้วยการทำงานของ FMS ทำให้สามารถอ่านและเขียนข้อมูลเกี่ยวกับ OD อ่านและเขียนตัวแปรที่กำหนดไว้ใน OD ได้ และกระบวนการอื่นๆ เช่น การอัพโหลด ดาวน์โหลดข้อมูล เป็นต้น แต่ละอุปกรณ์ฟาวน์เดชันฟิลด์บัสสามารถมี VFD ได้มากกว่า 1 VFD แต่โดยส่วนใหญ่แล้วจะมีเพียง 1 VFD ซึ่ง VFD จะมี 1 Resource Block ส่วน Function Block กับ Transducer Block อาจจะมีมากกว่าหนึ่งได้

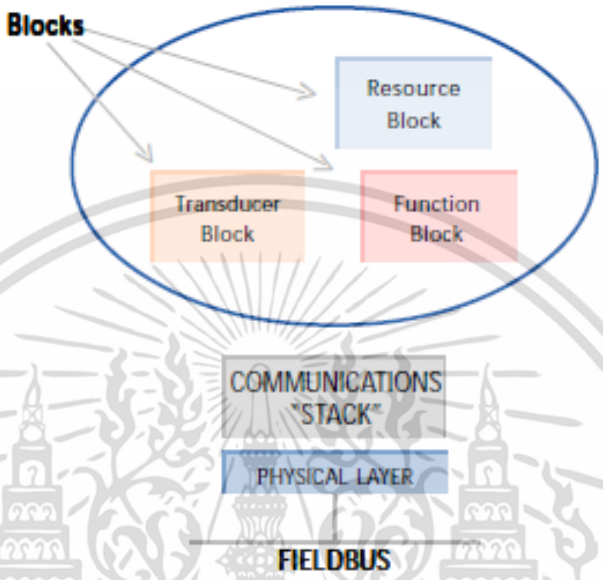



**ภาพที่ 2.8** สรุปประเภทและหน้าที่ของ VFD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.3 User Layer

เป็นส่วนที่กำหนดรูปแบบในการเข้าถึงเข้าข้อมูลภายในอุปกรณ์ฟาวน์เดชันฟิลด์บัส โดย User Application จะอยู่บนพื้นฐานของ Block ซึ่งทำหน้าที่แสดงถึงฟังก์ชั่นในตัวอุปกรณ์แบ่งออกเป็น Resource Block, Transducer Block, Function Block แต่ละประเภทถูกออกแบบมาให้ทำหน้าที่ต่างกันไปดังนี้




**ภาพที่ 2.9** ประเภทของ Block ใน User Layer

Resource Block เป็นส่วนที่แสดงถึงข้อมูลพื้นฐานของตัวอุปกรณ์ฟาวน์เดชันฟิลด์บัส เช่น ยี่ห้อ รุ่น ชื่ออุปกรณ์ หมายเลขอุปกรณ์ โดยที่แต่ละอุปกรณ์ฟาวน์เดชันฟิลด์บัสจะมี Resource Block เพียงแค่ Block เดียว

Transducer Block เป็นส่วนที่ทำหน้าที่ในการวัดแล้วแปลงค่าจากเซนเซอร์เป็นข้อมูลดิจิตอลก่อนที่จะส่งไปยัง Function Block รวมทั้งยังเก็บข้อมูลวันที่ทำการสอบเทียบและชนิดของเซนเซอร์ด้วย

Function Block เป็นส่วนที่ใช้ในเรื่องของการควบคุม ซึ่งอินพุตและเอาต์พุตของ Function Block สามารถติดต่อสื่อสารกับอุปกรณ์อื่นในระบบได้และจะมีกำหนดการทำงานที่แน่นอน โดยใน User Application เดียวสามารถที่จะมีหลาย Function Block ก็ได้ ซึ่งการทำงานของแต่ละ Block จะต่างกันออกไปเช่น

AI Block ทำหน้าที่อ่านข้อมูลจากช่องสัญญาณอนาลอกอินพุต ทำการฟิลเตอร์และปรับข้อมูลดิบให้อยู่ในหน่วยทางวิศวกรรม

AO Block ทำหน้าที่เขียนข้อมูลไปยังช่องสัญญาณอนาลอกเอาต์พุต

DI Block ทำหน้าที่อ่านข้อมูลจากช่องสัญญาณ Discrete อินพุต ทำการฟิลเตอร์และปรับข้อมูลดิบให้อยู่ในหน่วยทางวิศวกรรม

DO Block ทำหน้าที่เขียนข้อมูลไปยังช่องสัญญาณ Discrete เอาต์พุต

โดยฟาวน์เดชันฟิลด์บัสกำหนด Block มาตรฐานสำหรับการควบคุมไว้ 10 Blocks ดังตารางที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 2.2** Function Block มาตรฐานสำหรับการควบคุม

| Function Block Name | Symbol |
|---|---|
| Analog Input | AI |
| Analog Output | AO |
| Bias | B |
| Control Selector | CS |
| Discrete Input | DI |
| Discrete Output | DO |
| Manual Loader | ML |
| Proportional/Derivative | PD |
| Proportional/Integral/Derivative | PID |
| Ratio | RA |

## 2.3 วาล์วควบคุม

กระบวนการผลิตในอุตสาหกรรมประกอบไปด้วยระบบควบคุมมากมาย รวมกันเป็นระบบควบคุมขนาดใหญ่เพื่อใช้ในการผลิตสินค้าออกสู่ตลาด โดยที่แต่ละการควบคุมนั้นถูกออกแบบมาเพื่อให้ตัวแปรในกระบวนการที่สำคัญ เช่น ความดัน อัตราการไหล อุณหภูมิและอื่นๆ อยู่ในย่านที่เหมาะต่อกระบวนการเพื่อให้ได้ผลิตภัณฑ์ที่มีคุณภาพ ถ้าระบบควบคุมย่อยถูกรบกวนจนไม่สามารถควบคุมตัวแปรในกระบวนการให้เป็นไปตามค่าเป้าหมาย จะส่งผลกระทบต่อตัวแปรในระบบควบคุมอื่นด้วย เพื่อลดผลกระทบจากการรบกวนที่อาจทำให้ตัวแปรควบคุมมีค่าไม่ถูกต้องจึงใช้เซนเซอร์และทรานสมิตเตอร์ในการเก็บข้อมูลของตัวแปรในกระบวนการและความสัมพันธ์กับค่าเป้าหมายที่ต้องการ แล้วตัวควบคุมจะนำข้อมูลที่ได้จากกระบวนการมาตัดสินว่าต้องทำอะไร เพื่อให้ตัวแปรในกระบวนการกลับเข้าสู่ค่าที่ควรจะเป็นหลังจากที่เกิดการรบกวนระบบขึ้น เมื่อทำการวัด คำนวณ เปรียบเทียบเสร็จแล้วจะส่งให้ค่ากับส่วนควบคุมสุดท้ายซึ่งจะทำหน้าที่ในการใช้ข้อมูลที่ส่งมาจากตัวควบคุมในการควบคุมกระบวนการ ส่วนควบคุมสุดท้ายที่พบมากที่สุดในกระบวนการผลิตในอุตสาหกรรมคือ วาล์วควบคุม ซึ่งทำหน้าที่ควบคุมอัตราการไหลของของไหล เช่น แก๊ส ไอน้ำ น้ำ หรือสารประกอบทางเคมี เพื่อชดเชยภาระที่เกิดจากการรบกวนและควบคุมตัวแปรควบคุมในกระบวนการให้ใกล้เคียงกับค่าเป้าหมายมากที่สุด

### 2.3.1 วาล์วควบคุมฟาวน์เดชันฟิลด์บัส

อุปกรณ์ฟาวน์เดชันฟิลด์บัสจะมีตัวควบคุมและหน่วยความจำบนตัวอุปกรณ์เพื่อเพิ่มความสามารถในการคำนวณและควบคุมบนเครือข่าย รวมถึงประโยชน์จากข้อมูลในตัววาล์วเพื่อการบำรุงรักษาด้วย โดยการเข้าถึงข้อมูลและการควบคุมวาล์วจะมีการทำงานเข้าถึงในรูปแบบที่เรียกว่า Block ซึ่งประกอบไปด้วย

Resource Block มีหน้าที่ในการ Identify ตัวอุปกรณ์ว่าเป็นยี่ห้ออะไร โมเดลอะไร ซึ่งจะเป็นส่วนที่อธิบายข้อมูลทั่วไปของอุปกรณ์ฟาวน์เดชันฟิลด์บัสตัวนั้น ๆ

Transducer Block เป็นส่วนที่ทำหน้าที่ในการวัด การประมวลผลข้อมูล ก่อนที่จะส่งไปยัง Function Block

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

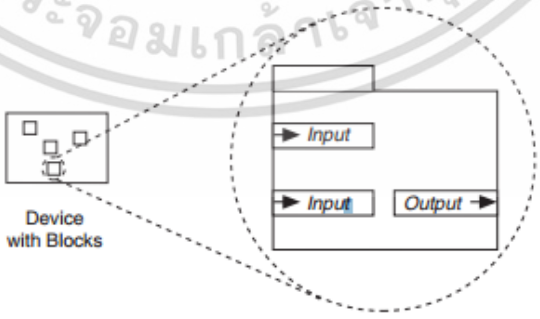
Function Block เป็นส่วนที่ใช้ในเรื่องของการควบคุม ซึ่งรูปแบบของ Function Block ที่ใช้งานโดยทั่วไป เช่น ใน Transmitter จะมี AI Block เพื่อที่จะรับค่าที่ได้จากการวัดมาทำการควบคุมกระบวนการ หรือแสดงผลได้ หรือใน Control Valve สำหรับผู้ผลิตบางผู้ผลิตจะมี AI Block, AO Block และ PID Block ซึ่ง Control Valve จะสามารถนำค่าการควบคุมที่รับมาจากส่วนควบคุมเข้ามาทำการเปิดปิดตัว Control Valve ในลักษณะของการควบคุมแบบ PID ได้ซึ่งในการที่จะสามารถนำ Block ต่าง ๆ ไปใช้งานนั้นจะต้องมีการเชื่อมต่อ Block ต่าง ๆ ให้เป็นไปในลักษณะของ Loop Control ของ AI Block, AO Block และ PID Block ร่วมกันเพื่อการควบคุม นอกจากนี้ Function Block ของวาล์วบางรุ่นจะถูกสร้างมาเฉพาะสำหรับตอบสนองกับผลิตภัณฑ์ซอฟต์แวร์ของผู้ผลิตนั้น ๆ ซึ่งจะเป็น Block ที่ทางผู้ผลิตเครื่องมือวัดและระบบควบคุมเป็นผู้ใส่เพิ่มเข้ามา โดยเป็นคุณสมบัติเฉพาะของแต่ละผู้ผลิตเครื่องมือวัด ซึ่งพารามิเตอร์ต่าง ๆ ใน Function Block สามารถเข้าถึงได้ด้วยการติดตั้ง Device Description หรือ DD File ซึ่งเป็นส่วนของรูปแบบ File ข้อมูลที่บ่งบอกถึงข้อมูลต่าง ๆ ของเครื่องมือวัดและควบคุมที่ Support อยู่ถ้าเปรียบเทียบกับ Personal Computer ก็จะเป็นเสมือน Driver ของ Hardware ที่เราได้ทำการติดตั้งเข้ากับ Personal Computer ของเรา และด้วยหลักการเดียวกันนี้เองหากมีการเปลี่ยนแปลงเครื่องมือวัดและควบคุมของระบบฟาวน์เดชันฟิลด์บัสที่มี Revision โมเดลหรือยี่ห้อที่ต่างไปจากเครื่องมือวัดและควบคุมตัวเดิมผู้ใช้งาน หรือผู้ดูแลระบบจะต้องทำการติดตั้ง (Installation) DD File ตัวใหม่เข้าไปด้วยเพื่อเป็นการอัพเดทข้อมูลของอุปกรณ์ให้ตรงกับอุปกรณ์ที่นำมาต่อใช้งาน โดยที่ภายใน Function Block จะมี Parameter ดังภาพที่ 2.10 ซึ่งแบ่งออกได้เป็น 3 ประเภทดังต่อไปนี้

Input Parameters ทำหน้าที่รับข้อมูลจาก Block อื่น

Output Parameters ทำหน้าที่ส่งข้อมูลให้ Block อื่น

Contained Parameters ไม่ส่งหรือรับข้อมูลจาก Block อื่น แต่จะเก็บค่าอยู่ภายใน Block

Parameters บางตัวจะมีรวมหลายการตั้งค่า เรียกว่า Field ตัวอย่างเช่น Output Parameter ของ Function block ซึ่งส่วนมากคือ OUT ซึ่งไม่ได้มีเพียงแค่ Value หรือ ค่าที่เป็นข้อมูลเพียงอย่างเดียว แต่ยังมีข้อมูลที่เป็น Status หรือ สถานะของ Value นั้นด้วย ดังนั้น OUT Parameter ประกอบไปด้วย 2 Fields คือ Value และ Status Field ของ Parameters จะแสดงด้วยรูปแบบ ParameterName.FieldName เช่น OUT.VALUE เป็นต้น




**ภาพที่ 2.10** Function Block Parameters

Function Block Mode เป็นอีกหนึ่งส่วนหนึ่งของ Function Block ที่มีความจำเป็นในการสั่งงานควบคุมอุปกรณ์ฟาวน์เดชันฟิลด์บัส โหมดของ Function Block โดยทั่วไปมีดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Automatic Mode ใช้ค่าเป้าหมายภายใน ใน Block Algorithm เพื่อการกำหนดค่าเอาต์พุต
2. Cascade Mode รับค่าเป้าหมายจาก Function Block อื่นสำหรับการใช้ Block Algorithm เพื่อกำหนดค่าเอาต์พุต
3. Out of Service Mode ไม่มีการทำงานของ Block โดยทั่วไปโหมดนี้ใช้ระหว่างเมื่อทำ Block Configuration หรือสำหรับอุปกรณ์บางชนิดที่ต้องการเปลี่ยนพารามิเตอร์บางตัว
4. Manual Mode ไม่มีการคำนวณเอาต์พุตด้วย Block Algorithm ผู้ใช้งานจะเป็นคนกำหนดค่าของเอาต์พุตเอง

### 2.3.2 หลักการในการวินิจฉัยวาล์วเบื้องต้น

เมื่อพบว่าการทำงานของกระบวนการควบคุมมีประสิทธิภาพต่ำลงหรือเกิดข้อผิดพลาดและวาล์วเป็นส่วนที่ต้องสงสัยว่าทำให้เกิดปัญหา การตรวจสอบและวินิจฉัยประสิทธิภาพการทำงานของวาล์วเบื้องต้นก่อนที่จะถอนการติดตั้งออกจากกระบวนการ จะช่วยลดระยะเวลาของผู้ปฏิบัติงานในการนำไปตรวจสอบ อีกทั้งยังทำให้การหาจุดบกพร่องของกระบวนการทำได้รวดเร็วขึ้น

มีการใช้การทดสอบวาล์วในหลากหลายวิธีเพื่อทำให้วาล์วอยู่ในสภาพที่สามารถทำงานได้อย่างมีประสิทธิภาพ [6] เช่น

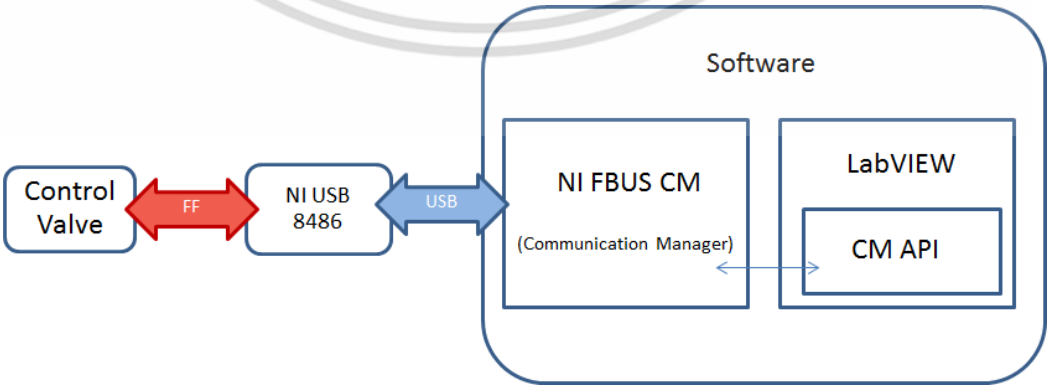
Valve Signature Test การเปรียบเทียบลายเซ็นของวาล์วโดยจะทำการทดสอบครั้งแรกก่อนการใช้งานเพื่อทำการเก็บค่าประสิทธิภาพของวาล์วคือความสัมพันธ์ของระยะเมื่อวาล์วเปิดและปิดกับความดันที่ใช้ในการเปิดปิดวาล์ว และจะทดสอบอีกครั้งหลังการใช้งานแล้วนำมาเปรียบเทียบกัน เพื่อตรวจสอบค่าคงที่ความแข็ง-อ่อนของสปริงและความเสียดทานของวาล์ว

Step Response Test การทดสอบผลตอบสนองขั้นบันไดตอนซึ่งเป็นวิธีการสำหรับการประเมินว่าวาล์วตอบสนองต่อการเปลี่ยนแปลงของอินพุตที่รับได้ดีแค่ไหน โดยการลด Overshoot, Dead Time, Dead Band ซึ่งจะช่วยในการควบคุมกระบวนการให้ดีขึ้นได้มาก

Dynamic Error Band ใช้ในการวิเคราะห์ Hysteresis, Dead Band, Dynamic Error

## 2.4 เทคโนโลยีของ National Instrument ที่ใช้ในงานวิจัย

การสื่อสารกับเครือข่ายระดับ H1 อาศัยชุดเครื่องมือจากบริษัท National Instrument ทำงานร่วมกันแสดงดังภาพที่ 2.11 โดยที่แต่ละส่วนประกอบไปด้วย




**ภาพที่ 2.11** การทำงานของเทคโนโลยีจาก National Instrument ในการติดต่อวาล์วควบคุม

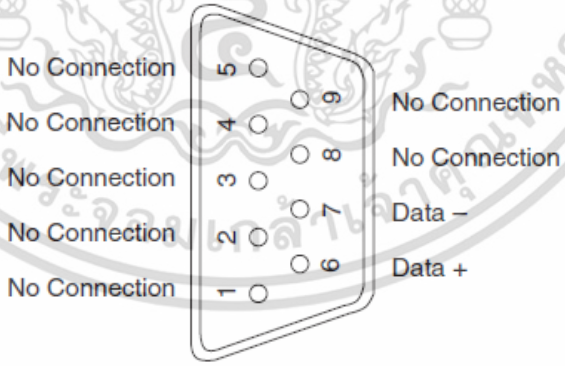
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.4.1 NI USB 8486

เป็นอุปกรณ์เชื่อมต่อกับเครือข่ายฟาวน์เดชันฟิลด์บัสที่มีการเชื่อมต่อแบบ USB กับคอมพิวเตอร์ มีหน้าที่ในการจัดการการสื่อสารระหว่างคอมพิวเตอร์ผ่านทางพอร์ต USB กับอุปกรณ์ฟาวน์เดชันฟิลด์บัสทางเครือข่าย H1 โดยที่ USB-8486 จะมีหน่วยประมวลผลอยู่ภายในตัวเพื่อทำหน้าที่จัดการการทำงานร่วมกันของฟาวน์เดชันฟิลด์บัส H1 Stack และสนับสนุนอัตราการถ่ายโอนข้อมูลของฟาวน์เดชันฟิลด์บัสในอัตรา 31.25 kb/s

**ภาพที่ 2.12** อุปกรณ์เชื่อมต่อเครือข่ายฟาวน์เดชันฟิลด์บัส H1 รุ่น NI USB-8486

โดยที่ส่วนเชื่อมต่อสัญญาณที่ USB-8486 จะเป็นหัวต่อสัญญาณแบบ DB-9 ที่ใช้งานกันทั่วไป โดยที่จะมีการใช้งานแค่ขาสัญญาณที่ 6 ซึ่งเป็นขาสัญญาณ Data + และขาสัญญาณที่ 7 ซึ่งเป็นขาสัญญาณ Data – ดังแสดงในภาพที่ 2.13




**ภาพที่ 2.13** หัวต่อและสัญญาณแบบ DB 9 ที่ใช้ใน NI USB-8486

### 2.4.2 NI-FBUS Communications Manager

เป็นส่วนที่ใช้งาน API ระดับสูง เพื่อจัดการการสื่อสารกับ National Instruments Foundation Fieldbus Communication Stack และฮาร์ดแวร์ จุดประสงค์หลักของ NI-FBUS Communications Manager คือการจัดเตรียมและให้รายละเอียดในการสื่อสารบนโปรโตคอลฟาวน์เดชันฟิลด์บัสด้วยการจัดเตรียมชุด API ที่รองรับการเข้าถึงในรูปแบบ TAG.Parameter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยการทำงานของ NI-FBUS Communication Manager ทำหน้าที่ในการจัดการการสื่อสารระหว่าง Communication Stack และ User Application รวมถึงยังจัดการรายละเอียดของการสื่อสารกับ Fieldbus Messaging Specification (FMS) และระดับอื่นที่อยู่ใน Communication stack โดยจะไม่แสดงรายละเอียดของ Virtual Communication Relationships (VCRs), การจัดการการสื่อสารและดัชนี OD แต่จะแสดงชื่อเพื่อการเข้าถึงอุปกรณ์, Virtual Field Devices (VFDs), ฟังก์ชันบล็อก, ทรานสดิวเซอร์บล็อกและพารามิเตอร์ การทำงานของ NI-FBUS Communications Manager API เป็นอิสระไม่ขึ้นกับฟาวน์เดชันฟิลด์บัสฮาร์ดแวร์ของ National Instrument และระบบปฏิบัติการ ผู้ใช้สามารถเพิ่มการเชื่อมต่อฮาร์ดแวร์ฟาวน์เดชันฟิลด์บัสได้หลายตัวในคอมพิวเตอร์เครื่องเดียวกันโดยสามารถใช้งานผ่าน NI-FBUS Communications Manager API ได้อย่างไม่มีปัญหาเพราะการเชื่อมต่อที่เป็นอิสระนั้นไม่จำเป็นต้องระบุตายตัวว่าต้องใช้การเชื่อมต่อกับฮาร์ดแวร์ฟาวน์เดชันฟิลด์บัสตัวไหนเพื่อเรียกใช้งาน การเชื่อมต่อจะถูกกำหนดผ่านทางข้อมูลที่จะส่ง เป็นการเพิ่มอิสระให้ในการพัฒนาแอพพลิเคชันกับการเชื่อมต่อฟาวน์เดชันฟิลด์บัส

NI-FBUS Communications Manager มีประโยชน์มากสำหรับโฮสแอพพลิเคชัน เช่น ซอฟต์แวร์ในการปรับแต่งฟังก์ชันบล็อก แอพพลิเคชันสำหรับการตรวจสอบฟังก์ชันบล็อก การวินิจฉัยเครือข่ายและการพัฒนาแอพพลิเคชันสำหรับการเชื่อมต่อกับ HMI เป็นต้น

### 2.4.3 LabVIEW

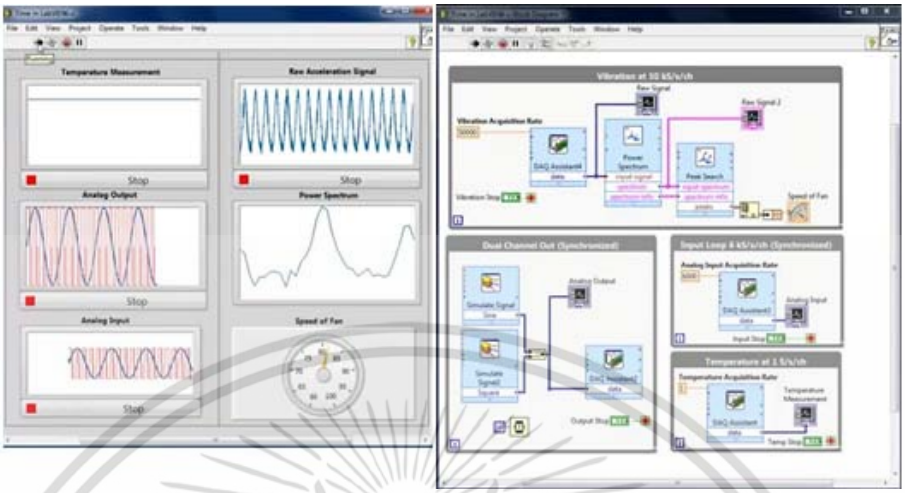
เป็นโปรแกรมคอมพิวเตอร์ที่เริ่มต้นสร้างมาเพื่อนำมาพัฒนาแอพพลิเคชันในด้านการวัดและเครื่องมือวัดสำหรับงานวิศวกรรมซึ่งย่อมาจากคำว่า Laboratory Virtual Instrument Engineering Workbench จุดประสงค์หลักของการทำงานของโปรแกรมคือการจัดการในด้านการวัดและเครื่องมือวัด และในตัวของโปรแกรมจะประกอบไปด้วยชุดฟังก์ชันที่ช่วยในการวัดมากมาย โดย LabVIEW จะมีประโยชน์อย่างสูงเมื่อใช้ร่วมกับเครื่องมือวัดทางวิศวกรรมต่างๆ ทั้งด้านการใช้งานที่ง่ายและการพัฒนาแอปพลิเคชั่นที่รวดเร็ว [7]

สิ่งที่ทำให้ LabVIEW แตกต่างไปจากโปรแกรมอื่นอย่างเห็นได้ชัดที่สุดก็คือวิธีการใช้งานหรือวิธีการพัฒนาแอปพลิเคชั่น ซึ่งเป็นแบบ GUI หรือ Graphical User Interface โดยสมบูรณ์ ดังภาพที่ 2.14 หมายถึง ผู้ใช้งานไม่จำเป็นที่จะต้องเขียน Code หรือคำสั่งใดๆ ทั้งสิ้น และที่สำคัญลักษณะภาษาที่ใช้ในโปรแกรมนี้เรียกว่าเป็นภาษารูปภาพ หรือเรียกว่าภาษา G (Graphical Programming Language) แทนการเขียนโปรแกรมทีละบรรทัดอย่างภาษาพื้นฐานทั่ว ๆไป เช่น C, C++, Basic ซึ่งช่วยลดระยะเวลาในการพัฒนาโปรแกรมลงไปได้เป็นอย่างมาก

จากความนิยมอย่างกว้างขวางจึงทำให้ปัจจุบันนอกจาก LabVIEW จะถูกใช้งานบนคอมพิวเตอร์ทั่วไปเพื่อการดึงสัญญาณหรือควบคุมเครื่องมือวัดแล้ว แนวคิดที่เรียกว่า Graphical System Design ยังขยายความสามารถของแพล็ตฟอร์ม LabVIEW ให้สามารถทำการออกแบบได้ทั้งระบบ ตั้งแต่ส่วนของการออกแบบซอฟต์แวร์ การคำนวณ ไปจนถึงการดึงสัญญาณและการติดตั้งลงไปทำงานบนฮาร์ดแวร์ทั้งแบบระบบปฏิบัติการวินโดวส์ ระบบปฏิบัติการเวลาจริงและ FPGA โดยใช้แพล็ตฟอร์ม LabVIEW เพียงแพล็ตฟอร์มเดียว จึงทำให้สามารถทำงานต่อยอดได้ทันทีเมื่อเวลาที่ต้องการย้ายจากการทำงานกับคอมพิวเตอร์ระบบปฏิบัติการวินโดวส์ไปเป็นระบบปฏิบัติการฮาร์ดแวร์แบบฝังตัว เป็นต้น นอกจากนั้น LabVIEW ยังพัฒนาชุดเครื่องมือสำหรับงานทางด้านต่าง ๆ เช่น ชุดเครื่องมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับติดต่อฐานข้อมูล โมดูลสำหรับการออกแบบระบบควบคุมและการจำลอง การประมวลผลภาพ โมดูลสำหรับการพัฒนาหุ่นยนต์ เป็นต้น



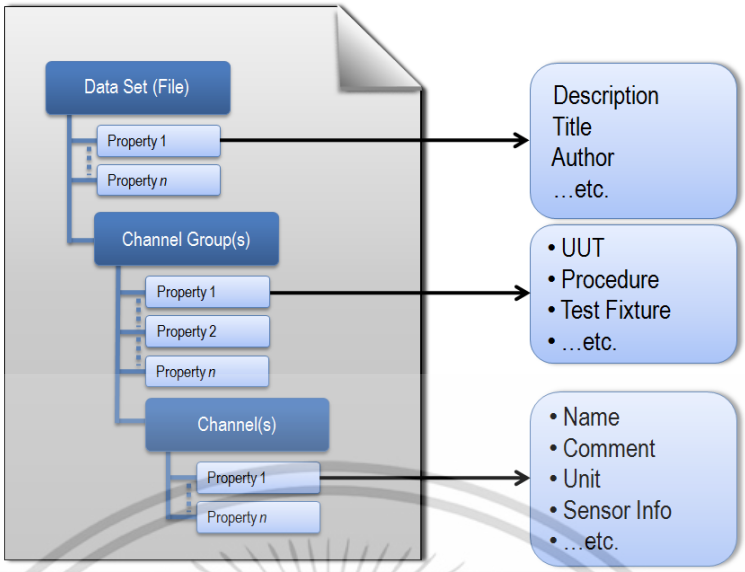
**ภาพที่ 2.14** การพัฒนาโปรแกรมประยุกต์บน LabVIEW

### 2.4.4 การจัดเก็บข้อมูล

การบันทึกข้อมูลจากใน LabVIEW เก็บลงไฟล์ และการอ่านข้อมูลจากไฟล์นำมาใช้ใน LabVIEW เป็นสิ่งที่ต้องใช้ในทุกโปรแกรมทั้งแบบที่บันทึกเป็นตัวอักษรหรือ Text File ในระบบ ASCII และอธิบายรวมไปถึงการบันทึกในระบบ Binary File ซึ่งเป็นภาษาเครื่อง และระบบ TDMS ซึ่งเป็นฟอร์แมตพิเศษที่รวมเอาข้อดีจากระบบการเก็บข้อมูลหลายๆแบบมาใช้

TDM Streaming เป็นฟอร์แมตสำหรับการเก็บข้อมูลที่เรียกว่า Technical Data Management Streaming หรือ TDMS โดยเฉพาะข้อมูลที่เกี่ยวกับการวัดและการทดสอบที่สัมพันธ์กับเวลา เพื่อให้การเก็บข้อมูลเป็นไปอย่างมีระบบ มีคำอธิบายกำกับ สามารถขยายช่องเก็บข้อมูลเพิ่มเติมได้ มีขนาดเล็กและมีประสิทธิภาพ ซึ่งจะช่วยให้ไม่ต้องเสียเวลาในการออกแบบโครงสร้างการเก็บข้อมูลที่ซับซ้อนในแต่ละงาน รวมไปถึงความสามารถในการแปลงกลับไปเป็นไฟล์แบบ MS Excel TDMS เก็บข้อมูลแบบ Binary โดยโครงสร้างจะแบ่งเป็นสามลำดับขั้นคือระดับ File, Channel Group และ Channel โดยในแต่ละลำดับขั้นสามารถในคุณสมบัติได้หลายตัว เพื่อเป็นการอธิบายรายละเอียดเกี่ยวกับสิ่งที่บันทึกในส่วนนั้นๆ โดยทั้งหมดนี้จะเก็บไว้อยู่ในไฟล์เดียวกันและมีสกุลเป็น TDMS โดยในหนึ่งไฟล์สามารถมีได้หลาย Group และแต่ละ Group สามารถมีได้หลาย Channel และสามารถแทรกคุณสมบัติเข้าไปในแต่ละส่วนได้ ดังภาพที่ 2.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 2.15** ไฟล์ TDMS ประกอบไปด้วยข้อมูลรายละเอียดของไฟล์แต่ละระดับ

ด้วยการออกแบบโครงสร้างการเก็บข้อมูลแบบนี้ ทำให้ TDMS สามารถใช้ได้กับงานเก็บข้อมูลในการวัดและทดสอบทุกประเภท ข้อมูลในโครงสร้างนี้ทั้งหมดจะถูกเก็บไว้ในไฟล์ TDMS แต่เมื่อมีการเรียกใช้ไฟล์นี้ จะมีการสร้างไฟล์ขึ้นอีกหนึ่งไฟล์ที่มีชื่อเดียวกันแต่มีสกุลไฟล์เป็น TDMS_Index ซึ่งเป็นไฟล์ Index ที่จะเป็นตัวช่วยให้สามารถอ่านข้อมูลได้เร็วขึ้น ซึ่งเวลาที่จัดเก็บไฟล์หรือส่งไฟล์ TDMS ให้ผู้อื่น ไม่จำเป็นต้องส่งไฟล์ Index ไปด้วย เพราะสามารถที่จะสร้างขึ้นใหม่ได้เองโดยอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3
# การออกแบบและการพัฒนาส่วนวินิจฉัยวาล์วควบคุมที่สื่อสารด้วยฟาวน์เดชันฟิลด์บัส

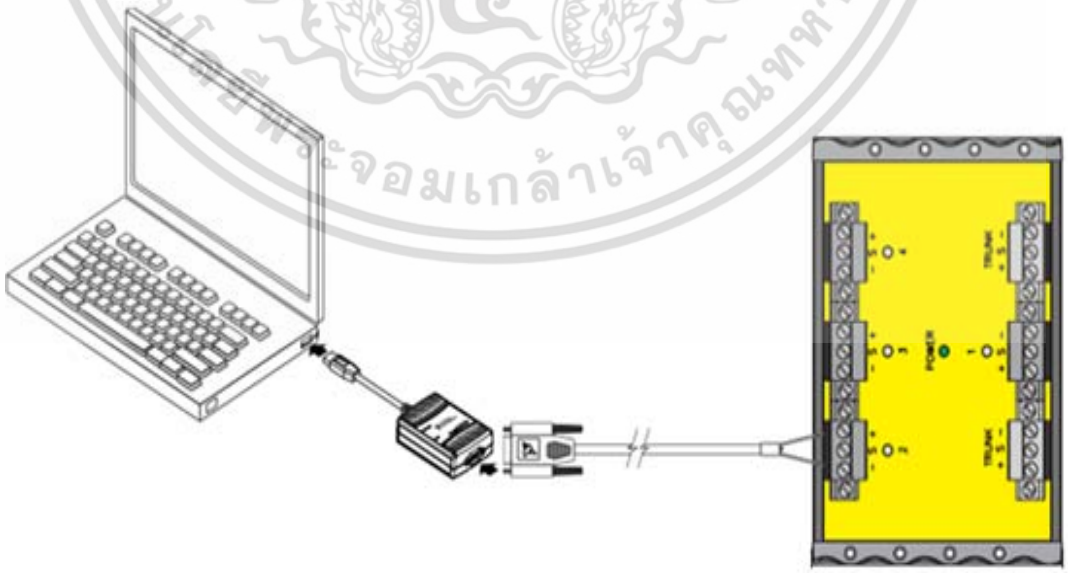
## 3.1 กล่าวนำ

ในบทนี้จะกล่าวถึงการออกแบบและพัฒนาส่วนวินิจฉัยวาล์วฟาวน์เดชันฟิลด์บัส ให้มีฟังก์ชันความสามารถในการตอบสนองตามที่ผู้ใช้งานกำหนด สั่งงานควบคุม อ่านค่าข้อมูล แล้วนำค่าที่ได้มาแสดงผลเพื่อประโยชน์ในการวิเคราะห์ข้อมูลที่ได้ต่อไป ซึ่งแบ่งออกเป็นสองส่วนหลักด้วยกัน คือ

1. ส่วนของฮาร์ดแวร์ที่นำมาใช้เพื่อเชื่อมต่อคอมพิวเตอร์เข้ากับเครือข่ายฟาวน์เดชันฟิลด์บัส H1
2. ส่วนของซอฟต์แวร์ที่นำมาพัฒนา User Interface และฟังก์ชันในการสื่อสารกับวาล์วฟาวน์เดชันฟิลด์บัส

## 3.2 ส่วนของฮาร์ดแวร์

ในการพัฒนาส่วนของการวินิจฉัยวาล์วควบคุม ในวิทยานิพนธ์นี้ได้เลือกใช้อุปกรณ์จากบริษัท National Instrument รุ่น NI USB 8486 ในการเชื่อมต่อคอมพิวเตอร์เข้ากับระบบซึ่งมีอินเตอร์เฟสแบบ USB โดยมีลักษณะการติดตั้งแสดงดังภาพที่ 3.1 เป็นการติดตั้งเข้ากับระบบที่มีอุปกรณ์ฟาวน์เดชันฟิลด์บัสอื่นรวมอยู่ด้วย NI USB 8486 มีพอร์ตสื่อสารฟาวน์เดชันฟิลด์บัส 1 พอร์ตในการเชื่อมต่อกับเครือข่ายฟาวน์เดชันฟิลด์บัส ตัวอุปกรณ์ทำหน้าที่ในการจัดการข้อมูลในส่วนที่เป็น Communication Stack แล้วแปลงเป็นข้อมูลเป็นสัญญาณที่ใช้ในระดับกายภาพหรือ Physical layer การเชื่อมต่อ NI USB 8486 จะมีจุดเชื่อมต่อสัญญาณอยู่สองจุดเป็น Data+ กับ Data- สามารถนำไปต่อเข้ากับเทอร์มินอลของระบบฟาวน์เดชันฟิลด์บัสได้เลย



**ภาพที่ 3.1** การติดตั้ง USB-8486 เข้ากับระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

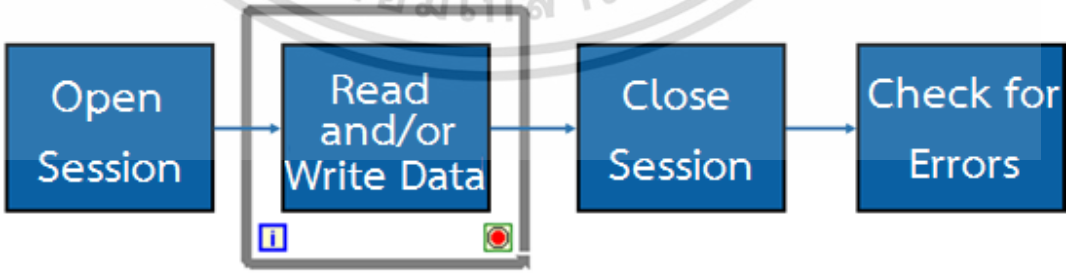
## 3.3 ส่วนของซอฟต์แวร์

การพัฒนาส่วนของซอฟต์แวร์ในวิทยานิพนธ์นี้ จะพัฒนาในส่วนที่เป็นการเก็บค่าของการตอบสนองขั้นบันไดในแต่ละลำดับขั้น เพื่อประโยชน์ในการตรวจสอบเปรียบเทียบในภายหลังหลังจากมีการใช้งานวาล์วควบคุมไปแล้ว โดยไม่จำเป็นที่จะต้องถอดวาล์วควบคุมออกจากระบบเพื่อนำมาตรวจสอบ

ส่วนของซอฟต์แวร์ของส่วนวินิจฉัยวาล์วฟาวน์เดชันฟิลด์บัสมีหน้าที่การทำงานอยู่สองส่วนด้วยกันคือส่วนที่ติดต่อกับผู้ใช้งาน GUI กับส่วนของซอฟต์แวร์ที่จะพัฒนาเพื่อติดต่อกับ NI USB 8486 ซึ่งการพัฒนาซอฟต์แวร์ทั้งสองส่วนนี้จะใช้ LabVIEW เข้ามาช่วยในการพัฒนาเนื่องจากเป็นผลิตภัณฑ์จากบริษัท National Instrument เหมือนกัน ทำให้ลดโอกาสที่จะเกิดข้อผิดพลาดในการสื่อสารกับอุปกรณ์ อีกทั้ง LabVIEW เป็นโปรแกรมที่ได้รับความนิยมในงานวิจัยต่างๆและความรวดเร็วในการพัฒนาโปรแกรมประยุกต์มากกว่าภาษาโปรแกรมเชิงข้อความทั่วไป (Text-Based Programming Language)

### 3.3.1 Communication Manager API

การติดต่อกับอุปกรณ์ในระบบฟาวน์เดชันฟิลด์บัสจะอาศัยกระบวนการที่เรียกว่า NIFB ซึ่งจะทำงานเมื่อ NI-FBUS Communication Manager เริ่มทำงาน โดยที่กระบวนการ NIFB จะทำหน้าที่เป็นตัวกลางในการสื่อสารระหว่างโปรแกรมประยุกต์กับอุปกรณ์บนเครือข่ายฟาวน์เดชันฟิลด์บัส รับการร้องขอจากโปรแกรมประยุกต์และส่งให้อุปกรณ์ตามตัวที่ระบุผ่านทาง NI USB 8486 ที่เชื่อมต่ออยู่กับคอมพิวเตอร์ การติดต่อกับ NI-FBUS นั้นจะอาศัย Communication Manager API ที่เป็นชุดโปรแกรมสามารถเรียกใช้ผ่าน LabVIEW ได้อย่างสะดวก โดยมีรูปแบบการทำงานทั่วไปแสดงดังภาพที่ 3.2 เมื่อเริ่มต้นจะทำการเปิดเซสชั่นเป็นอันดับแรก จากนั้นจะสามารถทำอ่านหรือเขียนข้อมูลได้จนกระทั่งต้องการจบการทำงานจึงทำการปิดเซสชั่นแล้วทำการตรวจสอบข้อผิดพลาดที่เกิดขึ้น โดยการสื่อสารระหว่างฟังก์ชันจะเข้าถึงอุปกรณ์ด้วย Reference Number หรือหมายเลขอ้างอิงทำหน้าที่เหมือนเป็นตัวชี้ทุกอุปกรณ์ชั่วคราว ซึ่งได้มาจากการเปิดเซสชั่นก่อนเริ่มการอ่านและเขียนข้อมูล เมื่อทำการปิดเซสชั่นจะเป็นการจบการเชื่อมโยงนั้น โดยการเปิดและปิดเซสชั่นควรทำเพียงครั้งเดียวเมื่อเริ่มและจบโปรแกรมเพื่อประสิทธิภาพการทำงานที่ดี



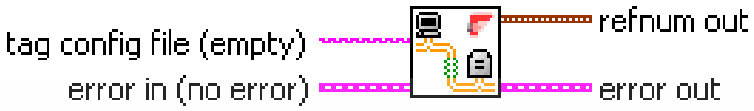

**ภาพที่ 3.2** การทำงานโดยทั่วไปของ Communication Manager API

โดยมีฟังก์ชันที่เรียกใช้งานในการพัฒนาส่วนวินิจฉัยวาล์วตามรูปแบบการทำงานในส่วนต่าง ๆ ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Open Session

เป็นฟังก์ชันในการเริ่มเปิดการสื่อสารกับฟาวน์เดชันฟิลด์บัส ช่องทางสื่อสารระหว่างโปรแกรมและ NIFB จะถูกกำหนดขึ้นและสามารถเริ่มการสื่อสารบนฟาวน์เดชันฟิลด์บัสได้ เช่น การอ่านหรือเขียน หลังจากเปิดเซสชั่นการสื่อสารกับฟิลด์บัสแล้วจะได้ Reference Number หรือหมายเลขอ้างอิงมาเพื่อเป็นสิ่งอ้างอิงถึงการสื่อสาร




**ภาพที่ 3.3** ฟังก์ชัน Open Session

**ตารางที่ 3.1** รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Open Session

| Required Input/Output | Description |
|---|---|
| Reference Number Output | The reference number of the NI Foundation Fieldbus Toolset. |

Browse Device List

เป็นฟังก์ชันที่ทำหน้าที่ในการเรียกดูการเชื่อมต่อฟาวน์เดชันฟิลด์บัสเพื่อดูรายการข้อมูลของอุปกรณ์ ซึ่งมีอยู่ 3 รายการด้วยกัน คือ PD Tag, ID, Address โดยจะต้องทำการให้ Reference Number และ Link Name กับฟังก์ชัน โดยที่ Link Name คือชื่อของการเชื่อมต่อกับอุปกรณ์ที่ต้องการที่จะทำการเรียกดู สามารถดู Link Name ได้จาก Interface Configuration Utility

**ภาพที่ 3.4** ฟังก์ชัน Browse Device List

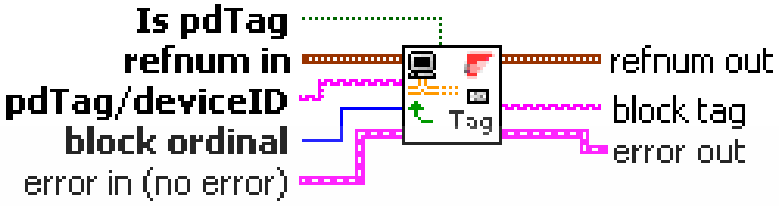
**ตารางที่ 3.2** รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Browse Device List

| Required Input/Output | Description |
|---|---|
| Reference Number Input | The reference number of the NI Foundation Fieldbus Toolset. |
| Link Name Input | The name of the link with the devices that you want to browse. |
| PD Tag List Output | An array of pdTags for the devices on this link. |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Get Block Tag

เป็นฟังก์ชันในการใช้เรียกดู Block Tag ในระหว่างที่การเชื่อมต่อถูกเปิดด้วยฟังก์ชัน Open Session โดยที่จะต้องให้ Reference Number, PD Tag หรือ Device ID, Block Ordinal กับฟังก์ชั่นเพื่อเป็นการเจาะจงถึงบล็อก



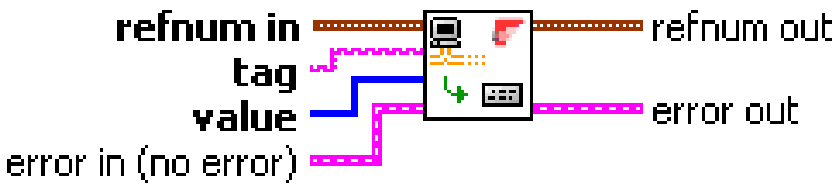

**ภาพที่ 3.5** ฟังก์ชัน Get Block Tag

**ตารางที่ 3.3** รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Get Block Tag

| Required Input/Output | Description |
|---|---|
| Reference Number Input | The reference number of the NI Foundation Fieldbus Toolset. |
| PD Tag/Device ID Input | The pdTag of the device if **Is** pdTag is true. It is the deviceID of the device if Is pdTag is false. |
| Is PD Tag Input | Use pdTag to identify the device if **Is** pdTag is true. Use deviceID to identify the device if **Is** pdTag is false. |
| Block Ordinal Input | The Foundation Fieldbus block ordinal, which is the index number of a block in a device. |
| Block Tag Output | The Foundation Fieldbus block tag. |

Function Write

เป็นฟังก์ชันที่ใช้ในการเขียนข้อมูลลงไปยังตัวอุปกรณ์ ในระหว่างที่การเชื่อมต่อถูกเปิดด้วยฟังก์ชัน Open Session สามารถรองรับชนิดของข้อมูลได้ทั้งหมด 15 ชนิดและอาเรย์ 8 ชนิดภายใต้ข้อกำหนดของฟาวน์เดชันฟิลด์บัส โดยที่จะต้องระบุ Tag ให้กับฟังก์ชันว่าต้องการที่จะเขียนค่าลงใน Tag ไหน ภายใต้รูปแบบ BLOCKTAG.PARAM หรือ BLOCKTAG.STRUCT.ELEMENT




**ภาพที่ 3.6** ฟังก์ชัน Write

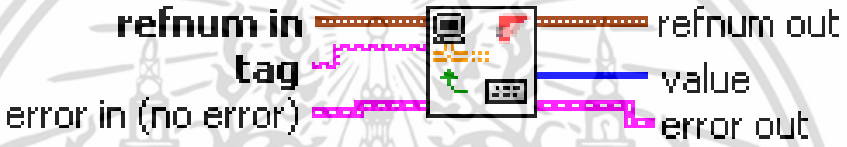
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.4** รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Function Write

| Required Input/Output | Description |
|---|---|
| Reference Number Input | The reference number of the NI Foundation Fieldbus Toolset. |
| Tag | Tag defined with the form BLOCKTAG.PARAM or BLOCKTAG.STRUCT.ELEMENT. |
| Value | The value written to the field device. |

Function Read

เป็นฟังก์ชันที่ใช้ในการอ่านค่าข้อมูลจากตัวอุปกรณ์ ในระหว่างที่การเชื่อมต่อถูกเปิดด้วยฟังก์ชัน Open Session ด้วยการระบุ Reference Number ของการเชื่อมต่อและ Tag ที่ต้องการที่จะอ่านเพื่อให้ได้ข้อมูลที่ต้องการ



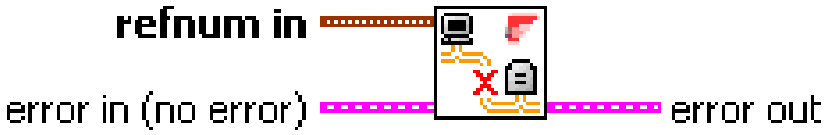

**ภาพที่ 3.7** ฟังก์ชัน Read

**ตารางที่ 3.5** รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Function Read

| Required Input/Output | Description |
|---|---|
| Reference Number Input | The reference number of the NI Foundation Fieldbus Toolset. |
| Tag | Tag defined with the form BLOCKTAG.PARAM or BLOCKTAG.STRUCT.ELEMENT. |
| Value | The value read from the field device. |

Close Session

เป็นฟังก์ชันในการปิดการเชื่อมต่อระหว่างฟาวน์เดชันฟิลด์บัส ใช้เมื่อจบการทำงานโปรแกรมและไม่ต้องการใช้การเชื่อมต่อนั้นแล้ว โดยต้องให้ Reference Number กับฟังก์ชันเพื่อเป็นการระบุการเชื่อมต่อที่ต้องการปิด




**ภาพที่ 3.8** ฟังก์ชัน Close Session

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.6** รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Close Session

| Required Input/Output | Description |
|---|---|
| Reference Number Input | The reference number of the NI Foundation Fieldbus Toolset. |

### 3.3.2 ส่วนของโปรแกรมในการติดต่อกับผู้ใช้งาน

ในส่วนของโปรแกรมที่เป็นส่วนติดต่อกับผู้ใช้งาน หรือ User Interface นั้นจะมีฟังก์ชันและการทำงานเพื่ออำนวยความสะดวกให้กับผู้ใช้งานเพื่อให้สามารถใช้งานได้ง่ายที่สุด ดังต่อไปนี้

Device Tag Information เป็นส่วนที่ใช้ในการกำหนดค่าและตรวจสอบการกำหนดค่าว่าถูกต้องหรือไม่ โดยจะมีรายการ แสดงดังภาพที่ 3.9

Device Name แสดงชื่อของอุปกณ์ทุกตัวที่อยู่บนเครือข่ายให้เลือก

PD Tag แสดงชื่อของ Tag จากอุปกรณ์ตัวที่เลือก

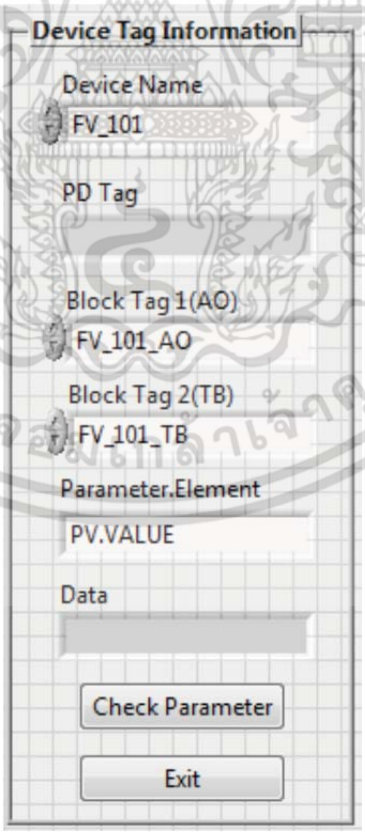
Block Tag 1 (Read) แสดงชื่อให้เลือก Block Tag โดยให้กำหนดค่าที่ต้องการทำการอ่าน

Block Tag 2 (Write) แสดงชื่อให้เลือก Block Tag โดยให้กำหนดค่าที่ต้องการทำการเขียน

Parameter.Element กำหนดชื่อของพารามิเตอร์กับ Element ที่ต้องการทดสอบ

Check Parameter กดเพื่อตรวจสอบค่าข้อมูลที่ทำการปรับไว้จาก Parameter.Element

Data แสดงข้อมูลจากการกด Check Parameter

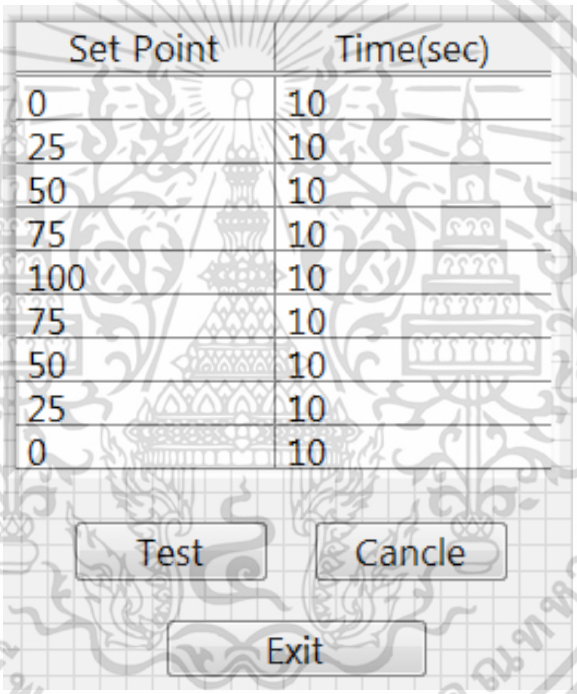



**ภาพที่ 3.9** ส่วนติดต่อกับผู้ใช้เพื่อกำหนดค่าและทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Parameters Setting

เป็นตารางสำหรับการกำหนดค่าที่ต้องการทำการทดสอบตามลำดับขั้นพร้อมกับเวลาที่ต้องการในแต่ละขั้น พร้อมกับมีปุ่มควบคุมการทำงานของโปรแกรมอยู่ในหน้าโปรแกรมนี้ด้วย ดังภาพที่ 3.10 คอลัมน์ทางด้านซ้ายของตารางจะเป็นคอลัมน์สำหรับการกำหนดค่าเป้าหมายหรือ Set Point ให้กับวาล์วควบคุมโดยค่าเป้าหมายในที่นี้หมายถึงเปอร์เซ็นต์ของการเปิดวาล์วควบคุม เช่น 0 หมายถึงวาล์วเปิด 0%, 50 หมายถึงวาล์วเปิด 50% เป็นต้น ส่วนคอลัมน์ทางด้านขวาจะเป็นคอลัมน์สำหรับใส่ค่าเพื่อกำหนดเวลาในการหน่วงก่อนที่จะเริ่มเปลี่ยนค่าเป้าหมายไปยังค่าถัดไป โดยผู้ใช้งานสามารถที่จะกำหนดค่าเป้าหมายและเวลาได้ตามที่ต้องการ ปุ่ม Test เป็นปุ่มสำหรับสั่งโปรแกรมให้เริ่มทำงาน โปรแกรมจะเริ่มนำค่าจากแถวบนสุดไป สั่งให้วาล์วควบคุมเปิด แล้วหน่วงเวลาตามกำหนดจนครบแล้วจึงจะนำค่าเป้าหมายในแถวถัดลงมาด้านล่างส่งต่อไป ปุ่ม Cancel ใช้สำหรับเมื่อต้องการหยุดการทดสอบในระหว่างที่กำลังทดสอบอยู่ ปุ่ม Exit สำหรับย้อนกลับไปหน้า Device Tag Information



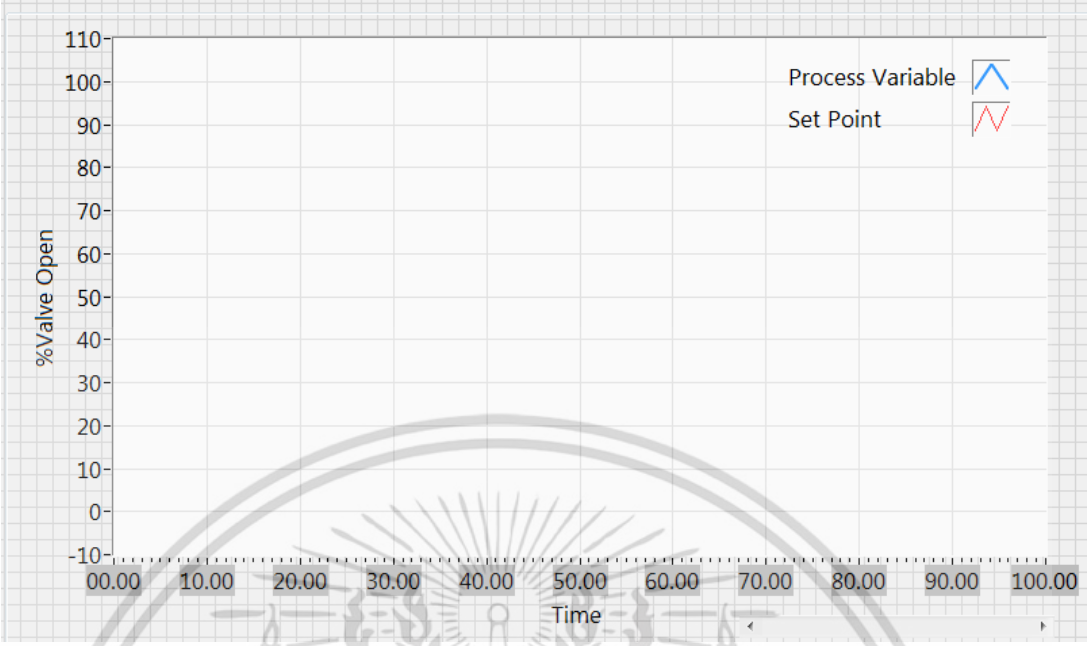

**ภาพที่ 3.10** ตารางสำหรับใส่ข้อมูลเพื่อทำการทดสอบตามผู้ใช้กำหนด

**3.3.3 การนำข้อมูลมาแสดงผลและเก็บข้อมูล**

เพื่อเปรียบเทียบข้อมูลที่ได้ระหว่างค่าเป้าหมายกับค่าตัวแปรกระบวนการจึงแสดงผลโดยผ่านกราฟ ดังภาพที่ 3.11 โดยที่แกน X จะเป็นแกนของเวลาและแกน Y เป็นเปอร์เซ็นต์ของการเปิดวาล์วควบคุม ค่าเป้าหมายจะถูกพล็อตด้วยสีแดงบนกราฟ ส่วนค่าตัวแปรกระบวนการจะถูกพล็อตด้วยสีน้ำเงิน

เมื่อทำการทดสอบสิ้นสุดลงโปรแกรมจะทำการเก็บข้อมูลเอาไว้ในรูปแบบของไฟล์สกุล TDMS ประกอบไปด้วยคอลัมน์ทั้งหมด 3 คอลัมน์ด้วยกัน ดังตารางที่ 3.7 คือ เวลา, ตัวแปรกระบวนการ, ค่าเป้าหมาย เมื่อมีข้อมูลเหล่านี้แล้วผู้ใช้สามารถที่จะนำมาพล็อตเรียกดูผลย้อนหลังเพื่อการเปรียบเทียบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 3.11** ส่วนติดต่อกับผู้ใช้งานสำหรับแสดงเปอร์เซ็นต์การเปิดวาล์วเปรียบเทียบกับเวลา

**ตารางที่ 3.7** รูปแบบการเก็บค่าจากการทดสอบ

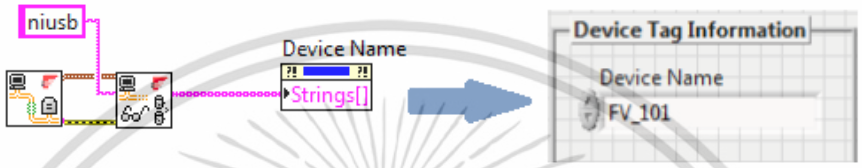
| Time | PV | SP |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

### 3.3.4 ภาพรวมของการทำงานของซอฟแวร์

เมื่อเริ่มโปรแกรมจะเป็นการพัฒนาในส่วนแรกคือ Open Session จะมีการเปิดเชื่อมต่อกับ NI-FBUS ด้วยฟังก์ชัน Open Session จะได้ Reference Number มาเพื่อใช้ในการดำเนินการอื่นๆ ต่อไป จากนั้นจะแสดงรายชื่อของ Device ทั้งหมดในช่อง Device Name ด้วยฟังก์ชัน Browse Device List โดยการระบุ Reference Number และ Link Name คือ NIUSB ฟังก์ชัน Browse Device List จะให้ PD Tag List เป็นเอาต์พุตออกมา ดังภาพที่ 3.13 และสามารถเรียกดูอุปกรณ์ทั้งหมดในระบบด้วยการเลือกที่ช่อง Device Name

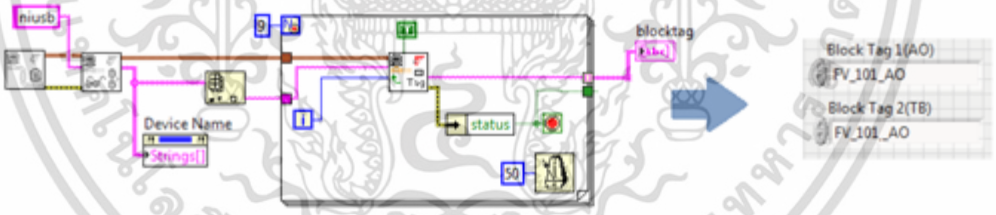
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ภาพที่ 3.12** ส่วนแรกของการพัฒนาซอฟท์แวร์

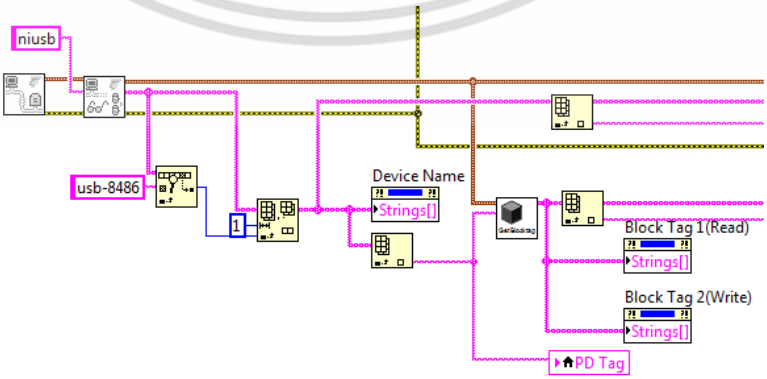



**ภาพที่ 3.13** Open Session และ Browse Device List เพื่อการแสดงชื่อของอุปกรณ์

จากนั้นสั่งให้แสดง Block Tag ของอุปกรณ์ตัวแรกที่ช่อง Block Tag 1 และ Block Tag 2 ด้วยฟังก์ชัน Get Block Tag ด้วยการระบุ PD Tag ของอุปกรณ์ตัวแรกกับ Block Ordinal ตั้งแต่ 0 ถึง 9 ดังภาพที่ 3.14 โดยสามารถกดเรียกดู Block Tag ของอุปกรณ์ได้จากช่อง Block Tag 1 และ Block Tag 2




**ภาพที่ 3.14** การเรียก Block Tag List ของอุปกรณ์มาแสดงผล



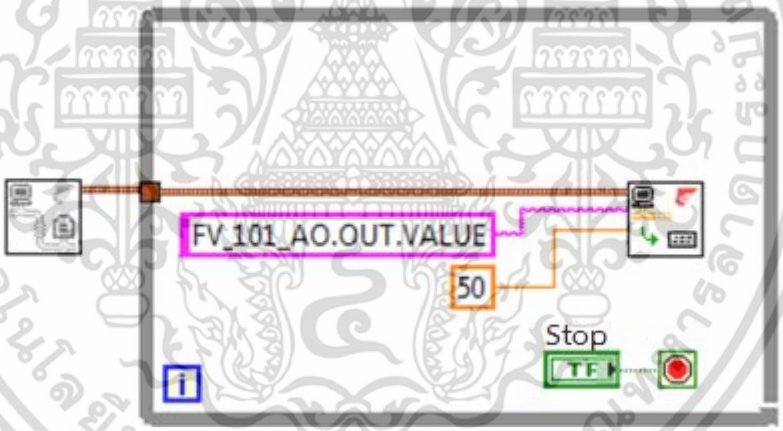

**ภาพที่ 3.15** ภาพรวมส่วนของการเริ่มต้นในโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วน Read And/Or Write Data เป็นส่วนของการทำงานซ้ำ ๆ คือการการอ่านและเขียนข้อมูล ซึ่งมีการสื่อสาร VCR แบบ Client/Server จะถูกกำหนดให้ทำงานอยู่ภายในลูป ดังภาพที่ 3.16

**ภาพที่ 3.16** การทำงานส่วนที่สองซึ่งมีการทำงานซ้ำ

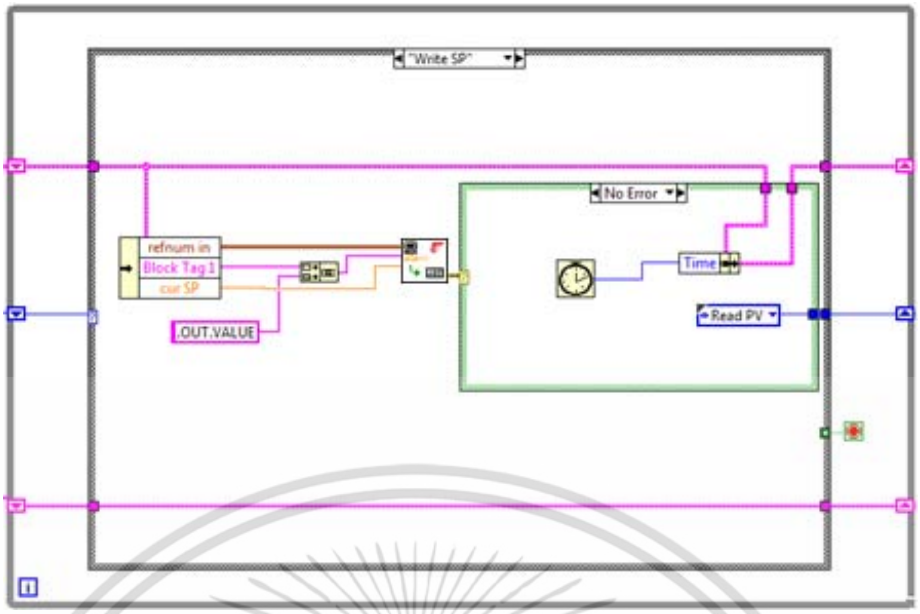
การเขียนข้อมูลสั่งงานให้วาล์วเคลื่อนที่ปิดหรือเปิดนั้นจะใช้ AO บล็อก พารามิเตอร์ OUT สั่งงานผ่านฟังก์ชัน Write ของ Communication Manager API แสดงการเขียนข้อมูลอย่างง่าย ดังภาพที่ 3.17 ซึ่งเป็นการสั่งให้วาล์วให้เปิด 50% โดยโปรแกรมจะเริ่มทำงานเมื่อสั่ง Run โปรแกรมบน LabVIEW และหยุดเมื่อกดปุ่ม Stop




**ภาพที่ 3.17** การเขียนข้อมูลสั่งงานวาล์วเคลื่อนที่

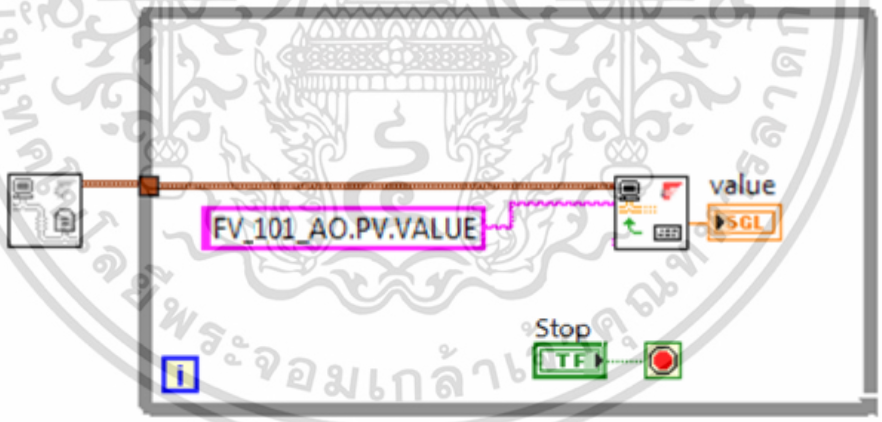
ค่าที่ได้ใส่ไว้ในตารางจะถูกจัดให้อยู่ในรูปแบบที่พร้อมใส่ให้กับฟังก์ชัน Write โดยจะนำค่าจาก Block Tag 1 ที่ได้เลือกไว้กับ Parameter OUT และ Element Value ที่กำหนดไว้ในโปรแกรมมาจัดให้อยู่ในรูป FV_101_AO.OUT.VALUE ดังภาพที่ 3.18 พร้อมกับเริ่มทำการจับเวลาด้วยฟังก์ชัน Tick Count ซึ่งเป็นฟังก์ชันมาตรฐานในโปรแกรม LabVIEW และตรวจสอบให้ตรงกับช่วงเวลาที่กำหนดไว้ในตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 3.18** การส่งค่าให้กับวาล์วควบคุม

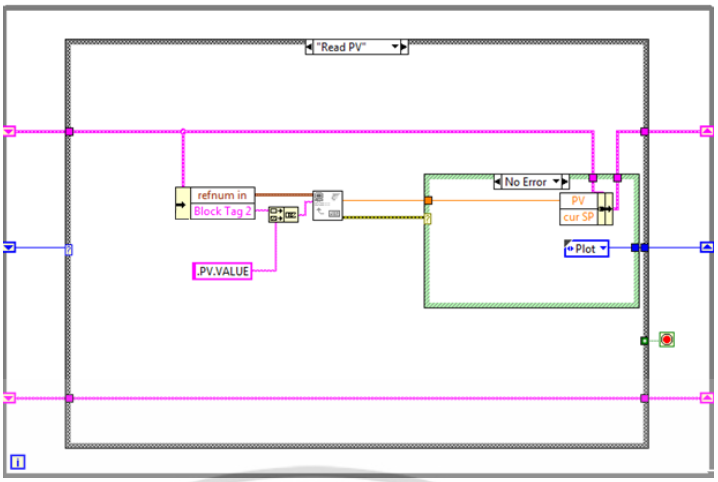
การตรวจสอบตำแหน่งการปิดหรือเปิดของวาล์วจะใช้ฟังก์ชัน Read ของ Communication Manager API อ่านค่าจาก AO บล็อก พารามิเตอร์ PV ดังภาพที่ 3.19 โดยโปรแกรมจะเริ่มอ่านค่าเมื่อสั่ง Run โปรแกรมบน LabVIEW และหยุดเมื่อกดปุ่ม Stop




**ภาพที่ 3.19** การอ่านค่าตำแหน่งของวาล์ว

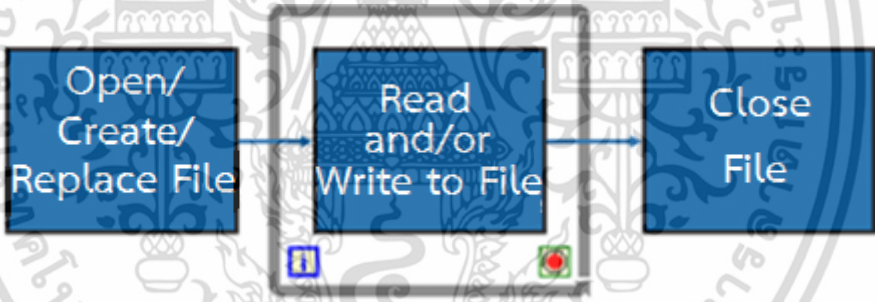
หลังจากทำการส่งค่าเป้าหมายไปแล้ว โปรแกรมจะทำการอ่านค่าตัวแปรกระบวนการกลับมา โดยจะนำค่าจาก Block Tag 2 ที่ได้เลือกไว้กับ Parameter PV และ Element Value ที่กำหนดไว้ในโปรแกรมมาจัดให้อยู่ในแบบที่พร้อมใส่ให้กับฟังก์ชัน Read คือ FV_101_AO.PV.VALUE ดังภาพที่ 3.20 แล้วฟังก์ชัน Read จะให้ค่าตัวแปรกระบวนการซึ่งเป็นตำแหน่งของการเปิดวาล์วเป็นเอาต์พุตออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 3.20** การอ่านค่าจากวาล์วควบคุม

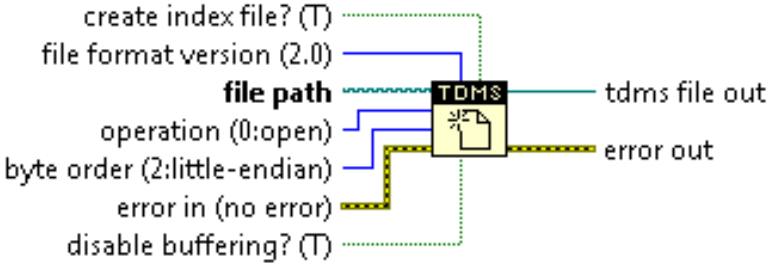
จากนั้นจะนำค่าตัวแปรกระบวนการที่อ่านมาได้ไปพล็อตเทียบกับค่าเป้าหมายบนกราฟและเมื่อสิ้นสุดกระบวนการทดสอบแล้วจะมีการเก็บข้อมูลลงไฟล์ตามรูปแบบของไฟล์สกุล TDMS ที่มีขั้นตอนการทำงานดังภาพที่ 3.21 คือ เปิด สร้างหรือแทนที่ไฟล์ จากนั้นทำการเขียนข้อมูลแล้วปิดไฟล์ด้วยฟังก์ชันดังต่อไปนี้




**ภาพที่ 3.21** ขั้นตอนการจัดเก็บข้อมูล

TDMS Open

เปิดไฟล์สำหรับการเขียนหรือการอ่านไฟล์ TDMS โดยฟังก์ชันจะให้ Reference Number ออกมาเพื่อการอ้างอิงถึงไฟล์ในการอ่านหรือเขียนไฟล์ต่อไป




**ภาพที่ 3.22** ฟังก์ชัน TDMS Open

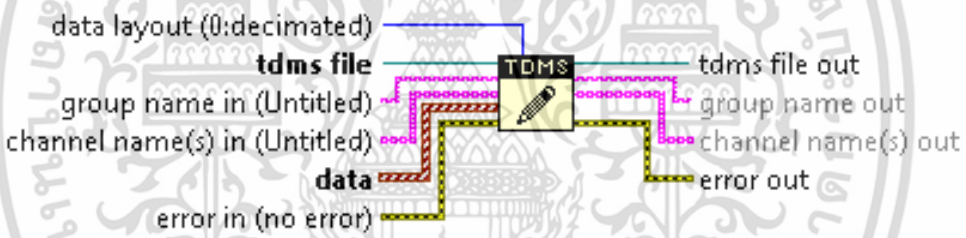
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.8** รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ Open TDMS

| Required Input/Output | Description |
|---|---|
| File Path | Specifies the absolute path to the file you want to open. If you use this function to create a new file, the file extension of the filename you specify in file path must be .tdms. Otherwise, this function automatically appends .tdms to the filename you specify. If you use this function to open or update an existing file, you do not have to ensure that the file extension is .tdms. |
| TDMS File Out | Returns a TDMS file reference to the .tdms file on which you performed the operation. |

TDMS Write

เขียนข้อมูลลง TDMS ไฟล์ โดยสามารถทำการกำหนด Group Name และ Channel Name เพื่อความเข้าใจในการเรียกดูข้อมูลในภายหลัง




**ภาพที่ 3.23** ฟังก์ชัน TDMS Write

**ตารางที่ 3.9** รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ TDMS Write

| Required Input/Output | Description |
|---|---|
| TDMS File | Specifies a reference number to the .tdms file on which to perform the operation. Use the TDMS Open function to open the refnum. |
| Data | The data to write to the .tdms file. This input accepts the following data types:<br>• Analog waveform or a 1D array of analog waveforms<br>• Digital waveform<br>• Digital table<br>• Dynamic data<br>• 1D or 2D array |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.9** (ต่อ)

| | |
|---|---|
| Group Name In | Specifies the channel group on which to perform the operation. The default is Untitled. |
| Channel Name(s) In | Indicates the channels on which to perform the operation. If you do not wire data to this input, LabVIEW names the channels automatically. If you wire waveform data to the data input, LabVIEW uses waveform names. The data type that the channel name(s) in input accepts can be a string or a 1D array of strings. The data type varies according to the data you wire to the data input. |

TDMS Close

เป็นฟังก์ชันสำหรับปิด Reference Number ของไฟล์ TDMS ดังภาพที่ 2.24

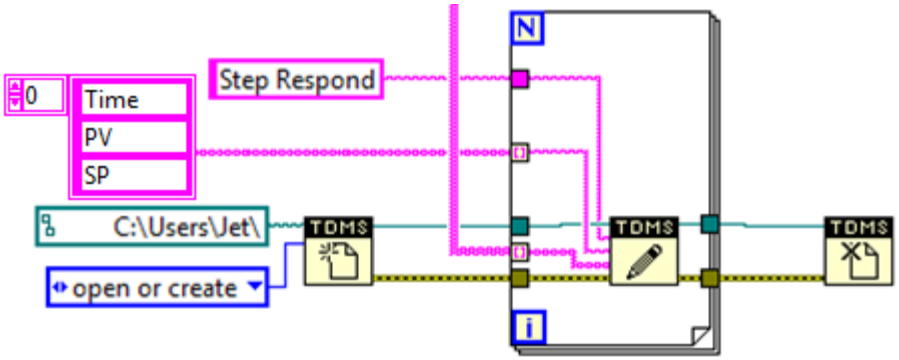
**ภาพที่ 3.24** ฟังก์ชัน TDMS Close

**ตารางที่ 3.10** รายละเอียดอินพุตและเอาต์พุตที่ต้องใช้ของ TDMS Close

| Required Input/Output | Description |
|---|---|
| TDMS File | Specifies a reference number to the .tdms file on which to perform the operation. |

โปรแกรมได้ทำการจัดเก็บค่าเป้าหมาย ค่าตัวแปรกระบวนการและเวลา ในรูปแบบของ TDMS ดังรูปที่ 3.25 โดยกำหนด Group Name เป็น Step Response และชื่อของ Channel Name เป็น Time, PV, SP ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 3.25** การจัดเก็บข้อมูลในโปรแกรม

ส่วนสุดท้ายของการพัฒนาซอฟต์แวร์คือการปิดเซสชั่นที่เปิดไว้ด้วยฟังก์ชัน Close Session โดยกำหนด Reference Number ให้กับฟังก์ชันแล้วทำการตรวจสอบข้อผิดพลาดที่อาจจะเกิดขึ้น การตรวจสอบข้อผิดด้วยฟังก์ชัน Simple Error Handler ด้วยการเชื่อมสัญญาณ Error Cluster เข้ากับตัวฟังก์ชัน ดังภาพที่ 3.27

**ภาพที่ 3.26** การทำงานส่วนสุดท้ายของการพัฒนาซอฟต์แวร์

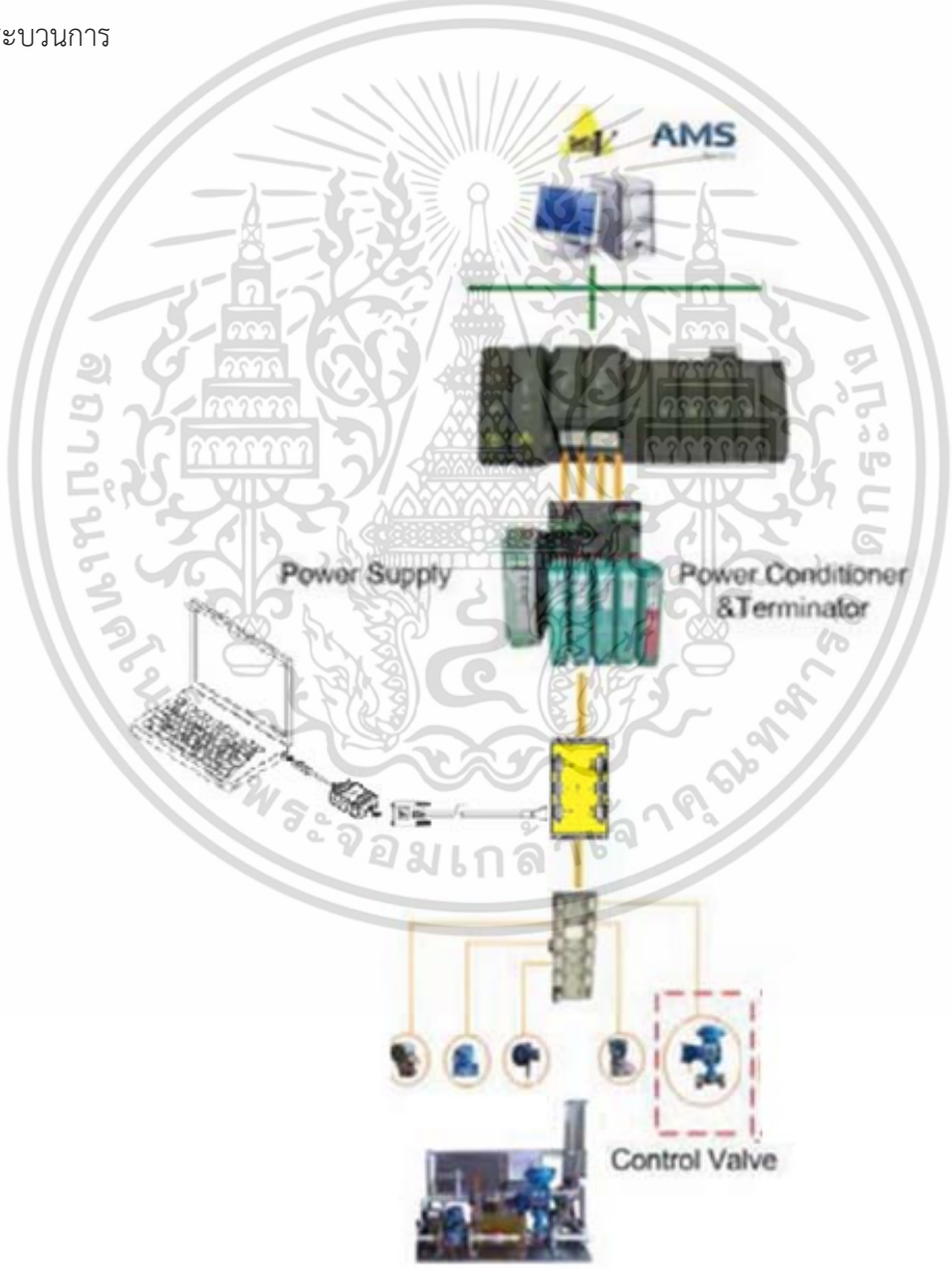
**ภาพที่ 3.27** การปิดเซสชั่นและการตรวจสอบข้อผิดพลาดของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 4
# การทดลองและผลการทดลอง

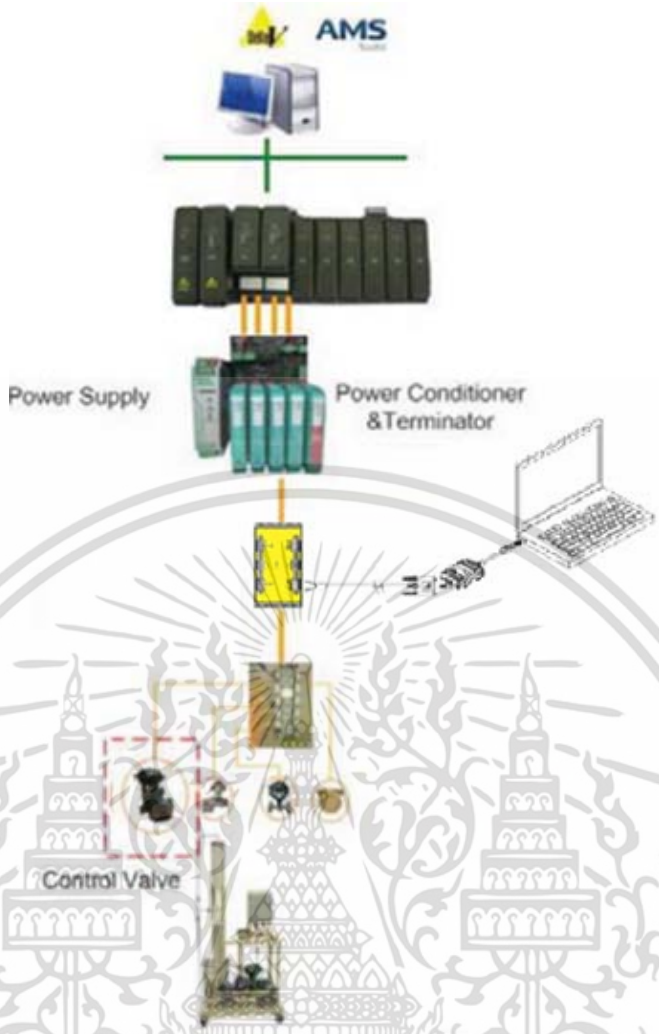
## 4.1 กล่าวนำ

ในบทนี้จะกล่าวถึงการทดลองใช้โปรแกรมที่ได้ออกแบบและพัฒนาขึ้นมาเพื่อการเก็บข้อมูลการตอบสนองขั้นบันไดของวาล์วควบคุม เพื่อทดสอบการทำงานของส่วนวินิจฉัยวาล์วควบคุมที่พัฒนาขึ้นมา จึงได้ทำการทดสอบกับวาล์วควบคุม Yamatake SVP3000 AVP303 ซึ่งเป็นวาล์วแบบปกติปิด และ EPM Fisher DVC6000f ซึ่งเป็นวาล์วแบบปกติเปิด ที่ติดตั้งอยู่ในกระบวนการควบคุมระดับ ดังภาพที่ 4.1 และ ภาพที่ 4.2 ด้วยการเชื่อมต่อสายจาก NI USB-8486 เข้ากับเซกเมนท์ของกระบวนการ




**ภาพที่ 4.1** ระบบฟาวน์เดชันฟิลด์บัสและวาล์วควบคุม Yamatake ในกระบวนการควบคุมระดับ

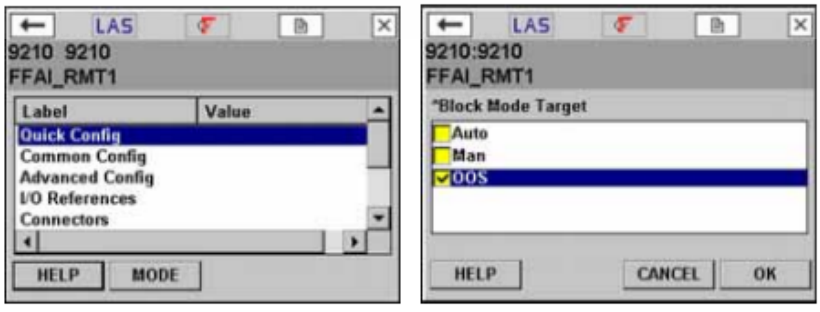
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 4.2** ระบบฟาวน์เดชันฟิลด์บัสและวาล์วควบคุม EPM Fisher ในกระบวนการควบคุมระดับ

## 4.2 ขั้นตอนการทดสอบกับวาล์วควบคุมฟาวน์เดชันฟิลด์บัส

เมื่อติดตั้ง NI USB-8486 เข้ากับเซกเมนท์และต่อเข้ากับคอมพิวเตอร์เรียบร้อย แล้วปรับวาล์วให้อยู่ในโหมด Out of Service หรือ OOS ด้วย 475 Field Communicator โดยการกดที่ปุ่ม Mode ที่ด้านล่างของจอแล้วเลือกเครื่องหมายถูกมาที่ OOS จากนั้นกดปุ่ม OK ดังภาพที่ 4.3

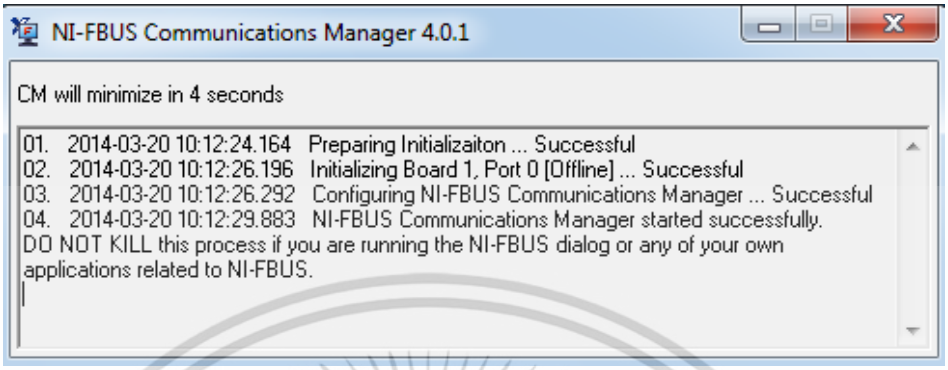



**ภาพที่ 4.3** เมนูบล็อกอุปกรณ์และหน้าการปรับโหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

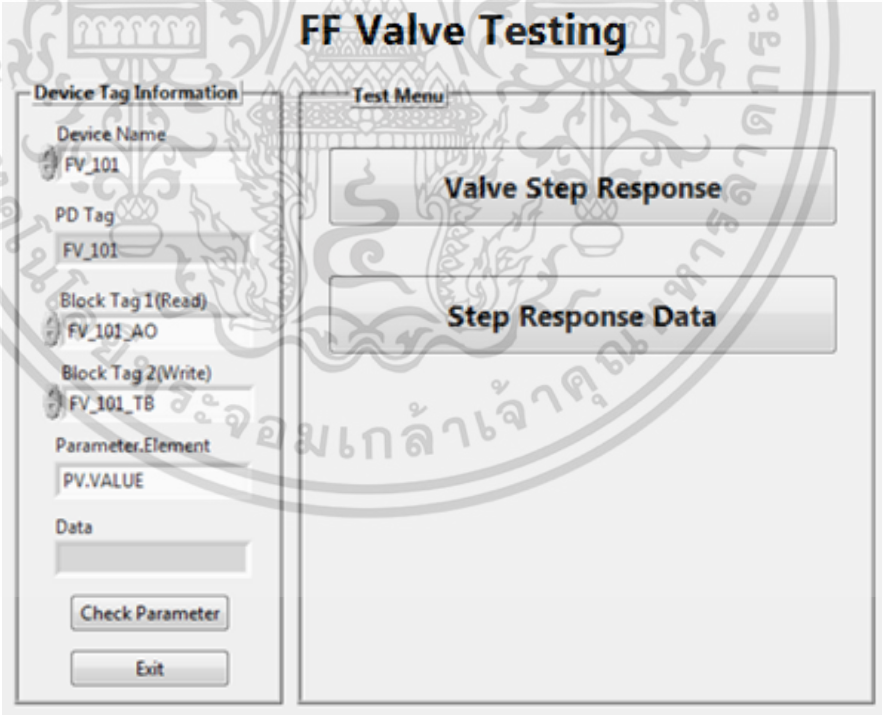
หลังจากนั้นจึงเริ่มทำการทดลองส่วนวินิจฉัยวาล์วควบคุมตามขั้นตอนดังต่อไปนี้

1. เปิดโปรแกรม NI-FBUS Communication Manager เพื่อเริ่มการทำงานของกระบวนการ NI-FBUS ซึ่งเป็นตัวกลางในการสื่อสารกับ NI USB-8486 ดังภาพที่ 4.4




**ภาพที่ 4.4** NI-FBUS Communications Manager

2. ทำการเปิดโปรแกรมส่วนวินิจฉัยวาล์วควบคุมที่พัฒนาขึ้นมา โปรแกรมแสดงหน้าจอเริ่มค้นหาอุปกรณ์ภายในเครือข่าย และทำการแสดงรายชื่ออุปกรณ์พร้อมกับ Block Tag ของอุปกรณ์ในหน้าโปรแกรม ดังภาพที่ 4.5

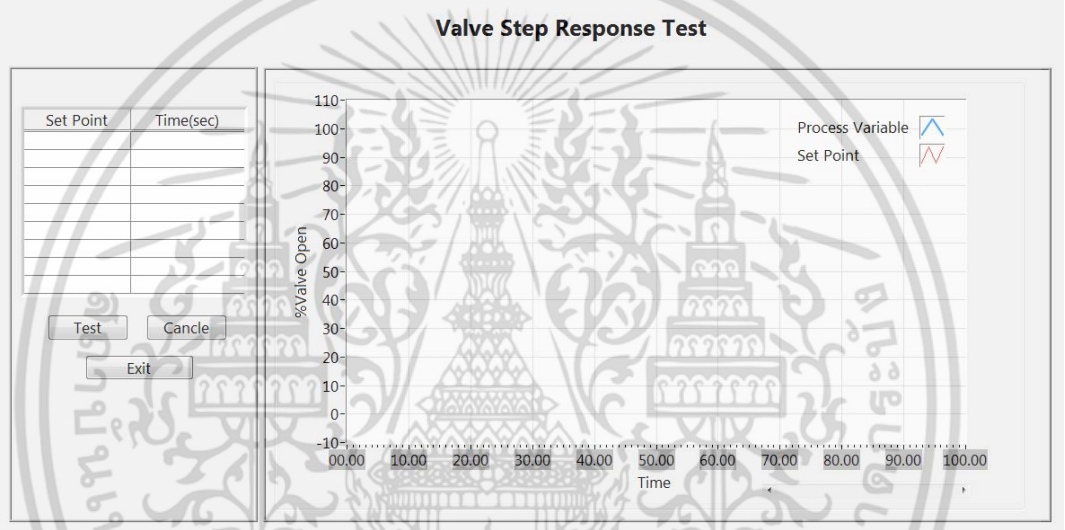



**ภาพที่ 4.5** ส่วนของโปรแกรมวินิจฉัยหลังจากการโหลดรายชื่ออุปกรณ์

3. ทำการเลือก Device Name ให้เป็นชื่อของวาล์วควบคุมที่ต้องการทำการทดสอบ เช่น FV_101

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

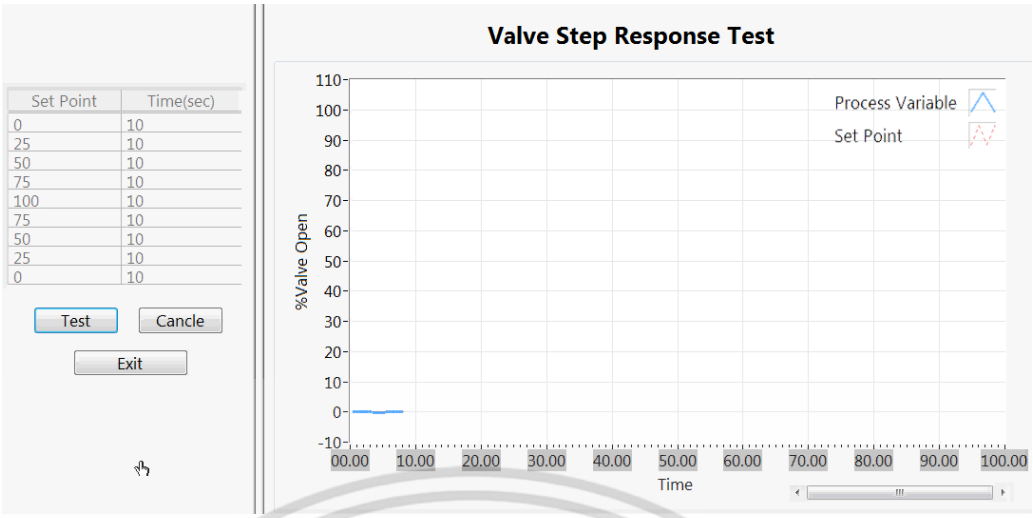
4. ทำการเลือก Block Tag สำหรับการอ่านค่า สำหรับการทดสอบวาล์วแบบผลตอบสนองขั้นบันไดจะเลือกอ่านค่าจากบล็อกแท็ก FV_101_AO ที่พารามิเตอร์ PV
5. ทำการเลือก Block Tag สำหรับการเขียนค่า สำหรับการทดสอบวาล์วแบบผลตอบสนองขั้นบันไดจะเลือกอ่านค่าจากบล็อกแท็ก FV_101_AO ที่พารามิเตอร์ OUT
6. การเลือก Block Tag ในการเขียนอ่านหรือเขียนค่าจะต้องระบุให้ถูกต้องมิฉะนั้นแล้วจะทำให้โปรแกรมไม่สามารถทำการอ่านค่าหรือควบคุมวาล์วได้และเนื่องจากพารามิเตอร์จะถูกกำหนดไว้ในการทดสอบวาล์วแบบผลตอบสนองขั้นบันไดอยู่แล้วจึงไม่ต้องทำการกำหนดค่าพารามิเตอร์
7. เลือก Valve Step Response ที่ Test Menu โปรแกรมจะเข้าสู่หน้าจอของการทดสอบวาล์ว ดังภาพที่ 4.6




**ภาพที่ 4.6** โปรแกรมทดสอบผลตอบสนองขั้นบันไดของวาล์วควบคุม

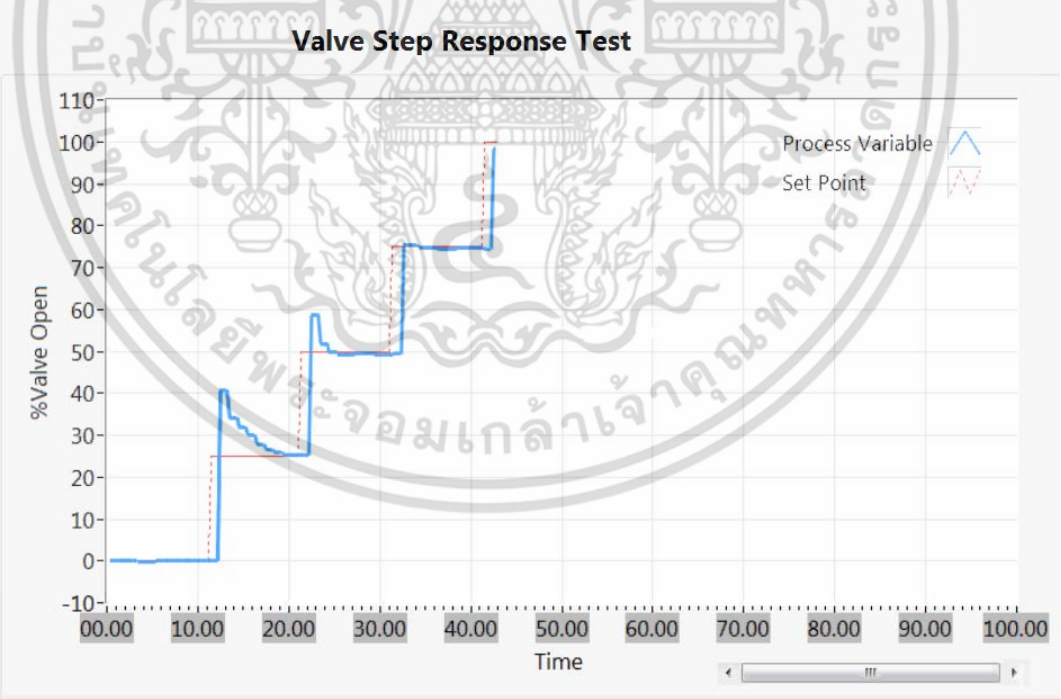
8. ป้อนค่าเป้าหมายที่คอลัมน์ Set Point เป็น 0%, 25%, 50%, 75%, 100%, 75%, 50%, 25%, 0% ตามลำดับ โดยใส่ค่าหน่วงเวลาที่คอลัมน์ Time แต่ละแถวเป็น 10 วินาที
9. หลังจากนั้นทำการกด Test โปรแกรมจะเริ่มทำงานตามค่าที่ป้อนไว้ในตาราง ดังภาพที่ 4.7 โดยในขณะที่ทดสอบวาล์วจะไม่สามารถปรับเปลี่ยนค่าเป้าหมายหรือเวลาได้และสามารถกดปุ่ม Cancel เพื่อหยุดการตรวจสอบวาล์วทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



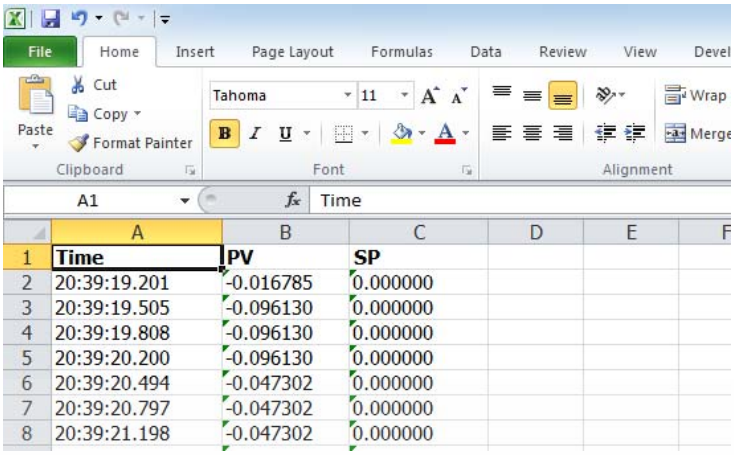

**ภาพที่ 4.7** โปรแกรมขณะเริ่มทำการทดสอบ

10. โปรแกรมจะเริ่มปรับค่าเป้าหมายไปเรื่อยๆ ตามค่าเป้าหมายและเวลาที่กำหนดไว้ ดังภาพที่ 4.8 จนครบ แล้วจะบันทึกค่าที่ได้จากการทดสอบเป็นข้อมูลที่สามารถเปิดจากโปรแกรม Microsoft Excel ดังภาพที่ 4.9 หรือเรียกดูจาก Step Response Data ใน Test Menu ได้ จากนั้นโปรแกรมจึงหยุดการทำงาน




**ภาพที่ 4.8** โปรแกรมทำการควบคุมสั่งเปิดวาล์วที่ 100%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



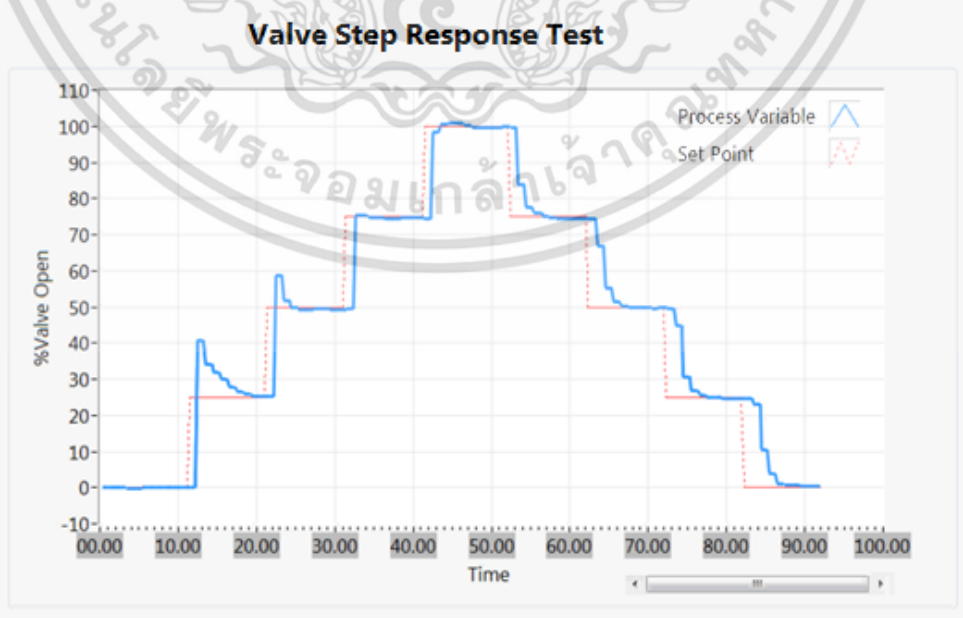
| | A | B | C |
|---|---|---|---|
| 1 | Time | PV | SP |
| 2 | 20:39:19.201 | -0.016785 | 0.000000 |
| 3 | 20:39:19.505 | -0.096130 | 0.000000 |
| 4 | 20:39:19.808 | -0.096130 | 0.000000 |
| 5 | 20:39:20.200 | -0.096130 | 0.000000 |
| 6 | 20:39:20.494 | -0.047302 | 0.000000 |
| 7 | 20:39:20.797 | -0.047302 | 0.000000 |
| 8 | 20:39:21.198 | -0.047302 | 0.000000 |

**ภาพที่ 4.9** ไฟล์ข้อมูลจากการทดสอบเปิดด้วย Microsoft Excel

11. ทำการทดสอบวาล์วควบคุมหลังจากให้ผู้อื่นใช้งานกระบวนการและปรับแต่งพารามิเตอร์ของวาล์วควบคุม

## 4.3 ผลการทดลองกับวาล์วควบคุม Yamatake SVP3000 AVP303

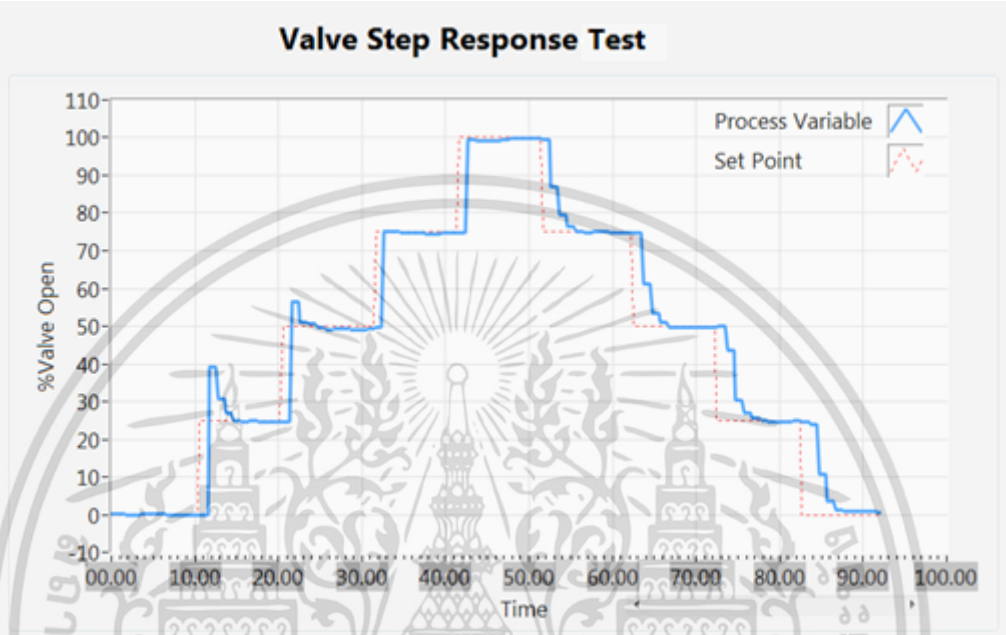
หลังจากทำการทดลองตามขั้นตอนกับวาล์ว Yamatake SVP3000 AVP303 แล้วได้ผลลัพธ์ดังภาพที่ 4.10 ซึ่งจะเห็นว่าการเคลื่อนที่ของวาล์วในขั้นที่หนึ่ง 0% ไปยัง 25% เกิด Overshoot ขึ้นอย่างชัดเจนก่อนเคลื่อนที่เข้าสู่ค่าเป้าหมายในช่วงปลายของการหน่วงเวลา เช่นเดียวกันกับในขั้นที่สอง 25% ไปยัง 50% แต่ในขั้นที่สองจะเคลื่อนที่เข้าสู่ค่าเป้าหมายเร็วกว่าในขั้นแรก ขั้นที่สามและขั้นที่สี่มี Overshoot เพียงเล็กน้อย ส่วนการเคลื่อนที่ขั้นที่ห้าถึงขั้นที่แปด การเคลื่อนที่สู่ค่าเป้าหมายในแต่ละขั้นเป็นไม่อย่างไม่ราบเรียบ สังเกตได้จากรอยหยักของกราฟก่อนเข้าสู่ค่าเป้าหมาย




**ภาพที่ 4.10** ความสัมพันธ์ผลตอบสนองขั้นบันไดของวาล์ว Yamatake SVP3000 AVP303

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยหลังจากที่ให้ผู้อื่นใช้งานและปรับแต่งระบบ ได้ทำการทดสอบผลตอบสนองขั้นบันไดอีกครั้ง ได้ผลลัพธ์ดังภาพที่ 4.11 ซึ่งแสดงให้เห็นว่าในขั้นที่หนึ่ง 0% ไปยัง 25% มีการเกิด Overshoot ขึ้น เหมือนแต่เคลื่อนที่เข้าสู่ค่าเป้าหมายเร็วขึ้นกว่าเดิมและในขั้นที่สองเกิด Overshoot น้อยลงเล็กน้อย แต่ไม่มี Overshoot เกิดขึ้นในขั้นที่สาม แต่มี Undershoot เกิดขึ้นเล็กน้อยในขั้นที่สี่ ส่วนการเคลื่อนที่ขั้นที่ห้าถึงขั้นที่แปดนั้นยังไม่มีความแตกต่างกับก่อนการใช้งาน

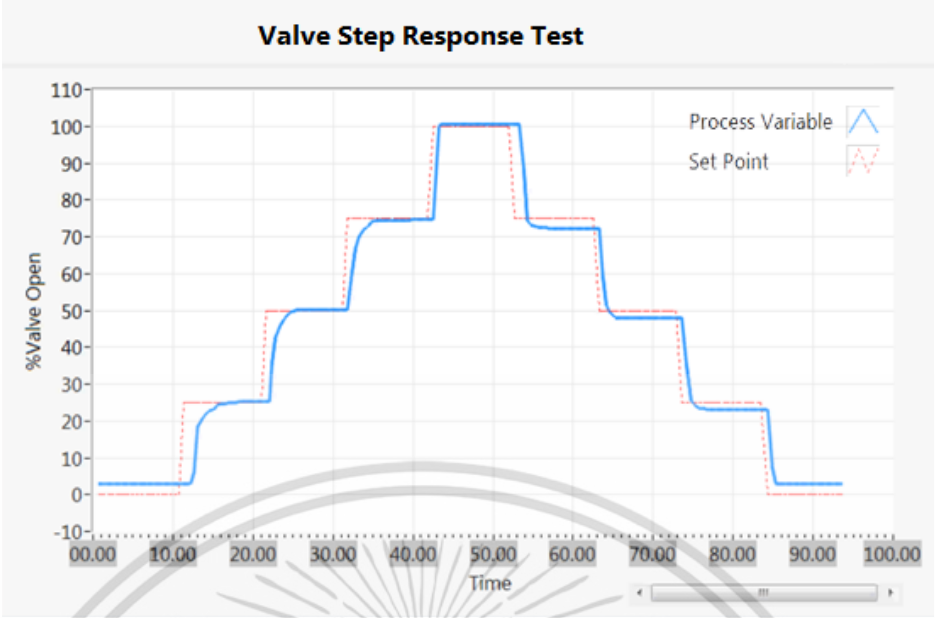



**ภาพที่ 4.11** ความสัมพันธ์ผลตอบสนองขั้นบันไดของวาล์ว Yamatake SVP3000 AVP303 ครั้งที่สอง

## 4.4 ผลการทดลองกับวาล์วควบคุม EPM Fisher DVC6000f

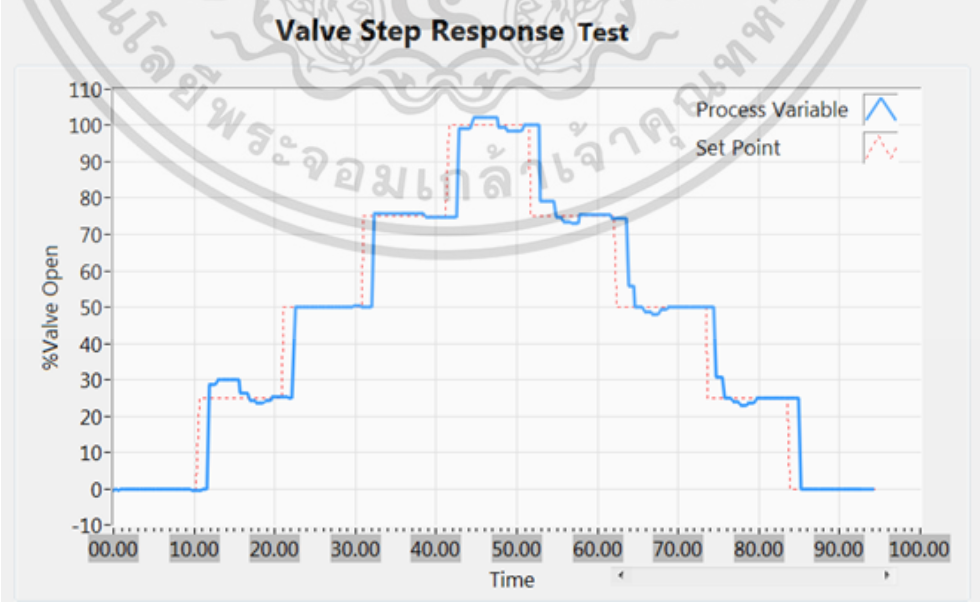
หลังจากทำการทดลองตามขั้นตอนกับวาล์ว EPM Fisher DVC6000f แล้วได้ผลลัพธ์แสดงออกมาดังภาพที่ 4.12 ซึ่งจะเห็นว่าการเคลื่อนที่ของวาล์วตั้งแต่ขั้นที่หนึ่งถึงขั้นที่สี่ 0% ไปยัง 25% จนถึงขั้น 75% ไปยัง 100% ไม่มี Overshoot เกิดขึ้นวาล์วเคลื่อนที่และเข้าสู่ค่าเป้าหมายอย่างถูกต้อง แต่การเคลื่อนที่ระหว่างขั้นที่ห้าถึงขั้นที่แปดนั้นมี Overshoot เกิดขึ้นและวาล์วไม่สามารถเคลื่อนที่เข้าสู่ค่าเป้าหมายได้ รวมทั้งวาล์วมีค่า Offset ไม่สามารถเคลื่อนที่ไปยัง 0% ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 4.12** ความสัมพันธ์แสดงผลตอบสนองขั้นบันไดของวาล์ว EPM Fisher DVC6000f

โดยหลังจากที่ให้ผู้อื่นใช้งานและปรับแต่งระบบ ได้ทำการทดสอบผลตอบสนองขั้นบันไดอีกครั้งได้ผลลัพธ์ดังภาพที่ 4.13 ซึ่งแสดงให้เห็นว่าในขั้นที่หนึ่ง 0% ไปยัง 25% มีการเกิด Overshoot เกิดขึ้นและไม่มีเสถียรภาพในการควบคุมค่าในช่วงเวลาที่กำหนดแต่ในขั้นที่สองไม่มี Overshoot เกิดขั้นและเข้าสู่ค่าเป้าหมายได้ในทันที การเคลื่อนที่ในขั้นที่สามนั้นเหมือนกับขั้นที่สองแต่มีการเคลื่อนที่เกินค่าเป้าหมายไปเพียงเล็กน้อย ในขั้นที่สี่มี Overshoot และไม่มีเสถียรภาพในการควบคุมค่าในช่วงเวลาที่กำหนด ขั้นที่ห้าถึงขั้นที่เจ็ดมี Overshoot เกิดขึ้นก่อนที่ค่าจะเข้าสู่ค่าเป้าหมาย ส่วนขั้นที่แปดไม่มี Overshoot เกิดขึ้นวาล์วสามารถเคลื่อนที่เข้าสู่ค่าเป้าหมายได้ทันที

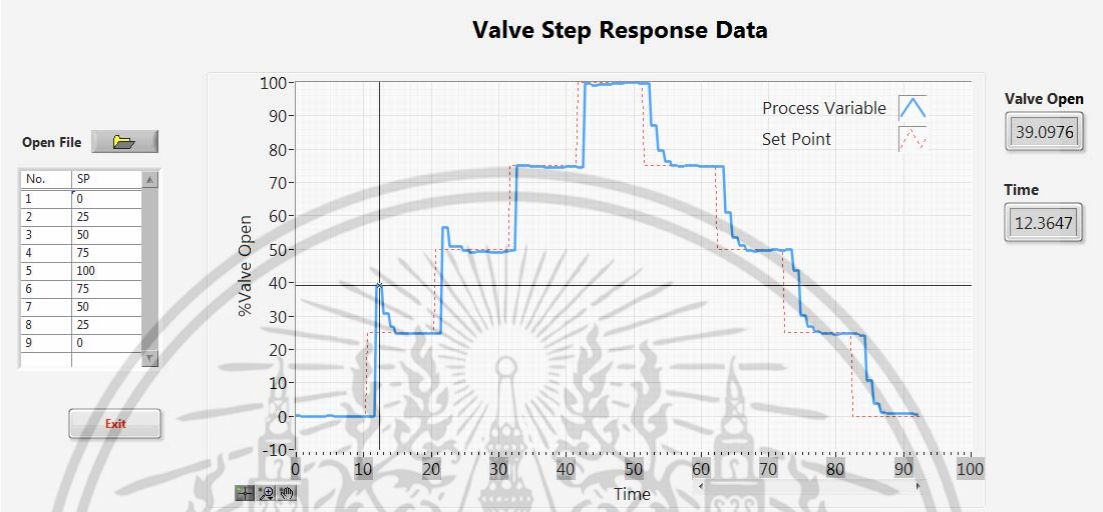



**ภาพที่ 4.13** ความสัมพันธ์ผลตอบสนองขั้นบันไดของวาล์ว EPM Fisher DVC6000f ครั้งที่สอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.5 อ่านไฟล์ที่บันทึกจากฟังก์ชัน Step Response Data

Step Response Data เป็นส่วนโปรแกรมในเรียกดูไฟล์ที่จัดเก็บมาจากการทดสอบวาล์วควบคุมสามารถเรียกเปิดไฟล์ที่จัดเก็บจากการทดสอบมาแสดงผลในรูปแบบของกราฟได้ โดยการเลือกไฟล์ที่ต้องการจากเมนู Open File จากนั้นโปรแกรมจะนำค่าจากไฟล์มาพล็อตลงบนกราฟ ดังภาพที่ 4.14




**ภาพที่ 4.14** เปิดข้อมูลจากการทดสอบผ่าน Step Response Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5
# สรุปผลการวิจัยและข้อเสนอแนะ

## 5.1 สรุปผลการวิจัย

จากการออกแบบและสร้างเครื่องมือวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมบนพื้นฐานฟาวน์เดชันฟิลด์บัสโดยใช้เทคโนโลยี NI แล้วนำมาทดลองใช้สามารถสั่งงานควบคุมวาล์วให้เป็นไปตามที่กำหนดและอ่านค่าจากวาล์วสามารถให้ข้อมูลผลตอบสนองขั้นบันไดของวาล์วในรูปแบบของกราฟและสามารถเก็บข้อมูลแล้วเรียกดูข้อมูลในภายหลังเพื่อการเปรียบเทียบซึ่งมีส่วนช่วยในการวินิจฉัยวาล์วในเบื้องต้นได้

การออกแบบและสร้างเครื่องมือวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมบนพื้นฐานฟาวน์เดชันฟิลด์บัสโดยใช้เทคโนโลยี NI ในวิทยานิพนธ์ฉบับนี้ถือเป็นอีกหนึ่งแนวทางในการพัฒนาเครื่องมือสำหรับกำหนดค่า และประเมินประสิทธิภาพของวาล์วควบคุมที่เป็นเสมือนหัวใจของตัวกระทำในกระบวนการปิโตรเคมีคอล โดยไม่ขึ้นกับผู้ผลิตของวาล์วควบคุม และยังถือเป็นอีกทางเลือกสำหรับวาล์วควบคุมที่ไม่มีซอฟต์แวร์ด้านการจัดการเครื่องมือวัด และสำหรับหน่วยงานที่ต้องการลดต้นทุนจากซอฟต์แวร์ด้านการจัดการเครื่องมือวัดของผู้ผลิตซึ่งมีราคาสูงที่จำเป็นสำหรับกระบวนการอุตสาหกรรมในการที่จะให้กระบวนการดำเนินไปอย่างมีประสิทธิภาพ โดยที่การพัฒนาในรูปแบบนี้ยังเป็นการเปิดโอกาสให้กับผู้พัฒนาสามารถนำความรู้และประสบการณ์มาประยุกต์ในการพัฒนาเครื่องมือที่ง่ายต่อการใช้งานและมีความเที่ยงตรงในการวิเคราะห์ปัจจัยและคาดการณ์การบำรุงรักษาล่วงหน้าในระดับสูงต่อไปได้

## 5.2 ข้อเสนอแนะและแนวทางการทำวิจัยต่อ

การออกแบบและสร้างเครื่องมือวินิจฉัยอย่างง่ายสำหรับวาล์วควบคุมบนพื้นฐานฟาวน์เดชันฟิลด์บัสโดยใช้เทคโนโลยี NI ในวิทยานิพนธ์ฉบับนี้ยังไม่สามารถระบุข้อบกพร่องและต้องการความชำนาญในการตีความหมายและวิเคราะห์ผลของการทดสอบ แต่ผู้ที่พัฒนาส่วนวินิจฉัยสำหรับฟาวน์เดชันฟิลด์บัสในแนวทางนี้ สามารถที่จะเพิ่มฟังก์ชันในการวิเคราะห์ค่าที่แสดงผลออกมาในรูปแบบของตัวเลขเพิ่มเติมเพื่อแสดงถึงประสิทธิภาพของวาล์วที่ชัดเจน หรือเพิ่มฟังก์ชันในการทดสอบวาล์วควบคุมแบบอื่น เช่น Valve Signature Test ที่ทำได้โดยการหาความสัมพันธ์ระหว่างการเคลื่อนที่ของวาล์วและความดันที่แอคชูเอเตอร์จากการสั่งวาล์วเคลื่อนที่และอ่านค่าความดันจากทรานดิวเซอร์วัดแรงดันเพื่อการวิเคราะห์ถึงสาเหตุที่แท้จริงของข้อบกพร่องของวาล์วควบคุมต่อไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# เอกสารอ้างอิง


[1] Mahalik, N. 2003. **Fieldbus Technology - Industrial Network Standards for Real-Time Distributed Control**. Berlin : Springer

[2] Zurawski, R. 2005. **The Industrial Communication Technology Handbook**. New York : CRC Press

[3] Pongswatd, S. Julsereewong, A. Nontanakorn, S. Sasajima, H. and Kitai, T. 2010. "Education of Automation Infrastructure Based on International Standards" **SICE Annual Conference 2010**. 2010(1) : 269-273.

[4] ศรีนคร นนทนาคร. 2552. **"Foundation Fieldbus Benefit for Control Valve Device Diagnostic."** กรุงเทพฯ : สถาบันเทคโนโลยีพระเจ้าเกล้าเจ้าคุณทหารลาดกระบัง. เอกสารอัดสำเนา.

[5] Hoon, P.S. 2006. "**Developing Valve Diagnostic Application for Foundation Fieldbus Devices.**" Singapore : Singapore : National Instrument.

[6] Verhappen, I. and Pereira, A. 2008 **Foundation Field bus** $3^{rd}$ **Edition**. North Carolina : ISA

[7] Krisi, B. 2004. **Beyond Basics - Asset Management Software Pushes Advanced Diagnostics**. Rijeka : InTech

[8] Blume, P. 2007. **The LabVIEW Style Book**. New Jersey : Prentice Hall


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก ก.

# ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

1. Sart Kummool, Sawai Pongswatd, Jetsada Jareonwatthanayothin “Simple Control Valve Diagnostic Tool for Foundation Fieldbus Technology”, SICE Annual Conference 2012, หน้า 959-962, 20-23 สิงหาคม 2555



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SICE2012

SICE Annual Conference 2012
http://www.sice.or.jp/sice2012/

Akita University, Akita, Japan
August 20-23, 2012

## Final Program and Abstracts

**Organized by**
The Society of Instrument and Control Engineers (SICE)

**Supported by**
Akita University

**Technically Co-Sponsored by**
IEEE Control Systems Society (CSS)
IEEE Robotics and Automation Society (RAS)
IEEE Industrial Electronics Society (IES)
Institute of Control, Robotics and Systems (ICROS)
The International Society of Automation (ISA)
Japan Electric Measuring Instruments Manufacturers' Association (JEMIMA)

**In Association with**
Asian Control Association (ACA)
China Instrument and Control Society (CIS)
The Chinese Association of Automation (CAA)
Chinese Automatic Control Society (CACS)
International Measurement Confederation (IMEKO)
IFAC NMO-Japan

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SICE Annual Conference 2012
August 20-23, 2012, Akita University, Akita, Japan

# Simple Control Valve Diagnostic Tool for Foundation Fieldbus Technology

Sart Kummool, Sawai Pongswatd, and Jetsada Jareonwatthanayothin

Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang
Ladkrabang, Bangkok 10520, Thailand
kksart@kmitl.ac.th, klsawai@kmitl.ac.th, jj-studio@hotmail.com

**Abstract:** This paper presents a simple diagnostic tool for configuration and diagnosis the control valves that are Foundation Fieldbus devices. National Instrument, NI USB-8486 is the hardware which is designed to connect to Foundation Fieldbus segment and Labview software with NI FBUS Communications Manager API is used to develop the diagnostic tool for parameters checking, stroke testing, data logging for signature comparisons and offline analysis in the control valve. The tool not impacts with other Fieldbus devices and controller in the H1 segment. Therefore, the end user can use the tool online and offline in the segment and reduce engineering cost for expensive vendor's asset management software. In additional, the tool have developed to test step response of control valves that can be developed and applied to analyze effective factors or predict the maintenance in higher level.

**Keywords:** Foundation Fieldbus, diagnostic tool, Labview, H1 segment, Control Valve.

## 1. INTRODUCTION

Recently, the structure of manufacturing processes has been changed from centralized control systems into decentralized control systems [1]. Traditional 4-20 mA current loop has been replaced by a fieldbus, which is all-digital, two-way, multi-drop communications among intelligent field devices [2]. Foundation Fieldbus proposed by Fieldbus Foundation is one of modern technologies commonly used that can be digital automation infrastructure enabling significant plant performance and economic improvements [3]. Nowadays, the Fieldbus Foundation has roadblock to promote the technology from the price of fieldbus components is higher. Fieldbus test devices are more complex compared to a multi- meter and Fieldbus systems are more complex, so users need to be more extensively trained or more highly qualified. Normally, the user can configure AO function block in the control valve by DCS (Distributed Control System) or Asset tool but they cannot test and diagnostic the control valve from different manufacturer. This research develops software for Fieldbus control valve to test and diagnostic which can operate with several models and manufacturers. NI FBUS and Labview have been developed as the diagnostic tool to test the 2 control valves from Emerson and Yamatake. The Yamatake SVP3000 AVP303 is the normal close control valve whereas EPM Fisher DVC6000f from Emerson is the normal open control valve.

## 2. FOUNDATION FIELDBUS NETWORK

The Foundation Fieldbus based on the related work of ISA (International Society of Automation) and IEC (International Electrotechnical Commission) is bi-directional digital communications technology for process and manufacturing automation [4]-[6]. Fig. 1 shows its open and nonproprietary system architecture that serves as a plant network for instrumentation and control systems in which each device has its own intelligence. It consists of two physical layers; H1 and HSE (High Speed Ethernet). H1 (running at 31.25 kbit/s) interconnects field instruments such as sensors, actuators, and I/O. HSE (running at 100 Mbit/s) provides integration of H1 segments (via a linking device), high speed controllers (such as PLCs), data servers, and workstations. Business Enterprise and Plant Application packages access the fieldbus information via the data services.

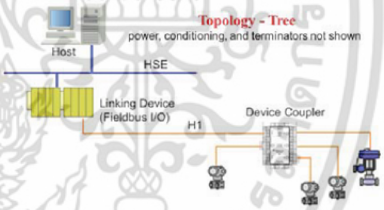



Fig.1 Foundation Fieldbus network.

FOUNDATION fieldbus H1 technology consists of the Physical Layer, the Communication "Stack," and the User Application Layer. The Open Systems Interconnect (OSI) layered communication model is used to model these components that can be shown by the Fig. 2. The Physical Layer is OSI layer1. The Data Link Layer (DLL) is OSI layer2. The Fieldbus Message Specification (FMS) is OSI layer7. The communication stack is comprised of layers 2 and 7 in them OSI model. The fieldbus does not use OSI layers 3, 4, 5 and 6. The Fieldbus Access Sub layer (FAS) maps the FMS onto the DLL. Each layer in the communication system is



PR0001/12/0000- 0959¥400 ©2012 SICE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

responsibility for a portion of the message that is transmitted on the fieldbus.

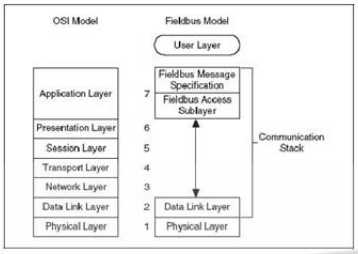



Fig.2 OSI model and Fieldbus model

The Foundation fieldbus has a standard User Application Layer based on blocks. The types of blocks used in a User Application are described in Fig. 3. Devices are configured using resource blocks and transducer blocks. The control strategy is built using Function Blocks. The resource block describes characteristics of the fieldbus device such as the device name, manufacturer, and serial number. There is only one resource block in a device. Transducer block are used to configure devices, decouple Function Blocks from the local input/output function required to read sensors and command output hardware. They contain information such as calibration date and sensor type. Function blocks provide the control system behavior. The input and output parameters of function blocks can be linked over the fieldbus. The execution of each function block is precisely scheduled. There can be many function blocks in a single User Application.

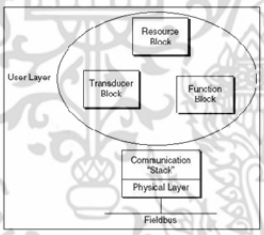



Fig. 3 The types of blocks used in a user application

## 3. SEGMENT INSTALLATION AND SYSTEM CONFIGURATION

Segment installation can be shown by the Fig. 4 It is a physical layer of Foundation fieldbus that uses NI USB-8486 and Fieldbus devices connected in H1 segment. Labview configurations for interfacing, reading and writing can be shown in Fig. 5. The user can configure field device and system in the user layer. The research creates the device tag information panel to support the user to choice and configure the field device. Fig. 6 shows device tag information panel for user to choice the field device by fill in the panel of device name, block tag, and parameter for control valve that wants to diagnose.

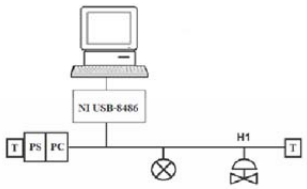



Fig. 4 H1 Segment installation

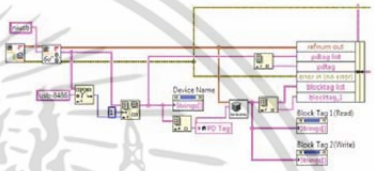


Fig. 5 Labview configurations for interfacing and read/write function

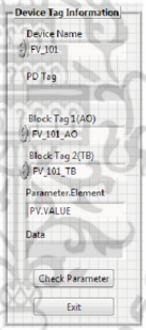



Fig. 6 Device tag information panel.

In addition, the diagnostic tool can diagnose with several models and manufacturers. Two control valves from Yamatake and EPM Fisher are tested as case study in the research. Fig.7 shows the Foundation fieldbus system that consists of two control valves in the H1 segments, DCS(Distributed Control System), power supply, and power conditioner. The proposed diagnostic tool is connected to junction box of each H1 segment for testing.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

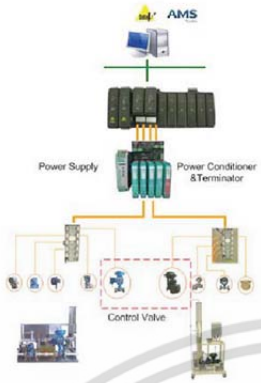



Fig. 7 Foundation Fieldbus system Configuration

The proposed diagnostic tool is developed from Labview program and NI FBUS hardware. The information panel and graphic is designed for user interface. NI FBUS Communication Manger API in program is used for write and read data between AO function block and Labview program. Fig. 8 (A) and (B) show the program for writing and reading data to AO block, respectively.

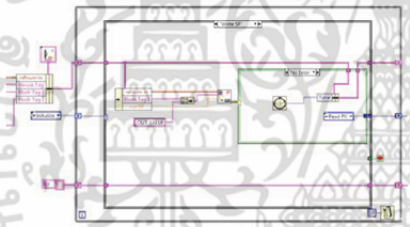


Fig.8 (A) Labview program to write data to AO block.

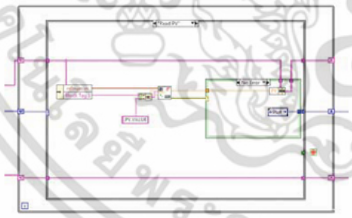


Fig.8 (B) Labview program to read data from AO block.

From the physical layer to user layer that shows by Fig.7, Fig.8 (A) and (B). In order to be able to utilize the diagnostic tool, the user selects device tag name and function block tag name match to assigned control valve. Then set the mode of function block to OOS (out of service) or auto or manual mode. For example, the reading function defines tag is "Tag Name_Function Block_Parameter" that is FV_201_AO.PV.VALUE. The writing function tag is FF_201_AO.OUT.Valve. The function of diagnosis tool for stroke testing can operate by open to close or close to open relate the time that is defined by user. The tool read percent of valve open or close and time then plot the graph of stroke response. In addition, the data of SP(set point), PV(process value), and time can save into file to analysis the valve signature comparisons, maintenance, detecting change in step response waveform. The step for stroke testing can set by the screen that can be shown in the Fig. 9.

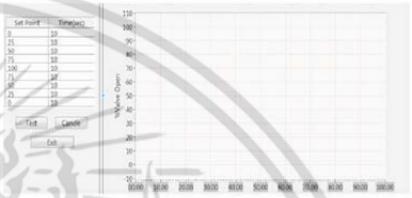


Fig. 9 screen to fill in the step of set points and times for stroke testing

## 4. EXPERIMENTAL RESULTS

The experiments perform with normal open control valve model Fisher DVC6000f and normal closed control valve model SVP3000 AVP303 from Emerson and Yamatake, respectively.

The experiment for stroke testing of control valve as the following steps:

1. Set mode of the control valve to out of service.
2. Select device name of valve control and analog output block (AO) in the device tag information panel.
3. Set the set point and time in the diagnostic tool software. Fig. 9 shows the screen of stroke testing that has step 0%, 25%, 50%, 75% and 100% of stroke respectively and then decrease 25% in each step until to 0%. The step of time is 10 seconds.
4. The diagnostic tool is started step response test by pushing the test button from the program, then control valve start to open depend on the set points and times until it's done. In each step of testing, the tool send set point value to the AO function block of control valve then read PV of the AO control valve to the tool. The data of PV and response time are plotted in the graph.

Fig. 10 and Fig. 11 show steps response in percent of valve open with time of Yamatake SVP3000 AVP303 and EPM Fisher DVC6000f, respectively.

The graph of step response obtained from the diagnostic tool which is for evaluating how well the valve tracks an input change illustrates that the Fig. 10 control valve has overshoot and does not travel



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

smoothly when travel down to 0%. Fig. 11 control valve does not fully closed when SP = 0% and when travel down valve opening has an offset from the set point.

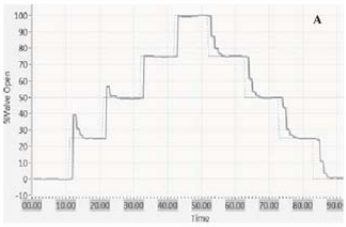



Fig. 10 Step response of Yamatake SVP3000 AVP303

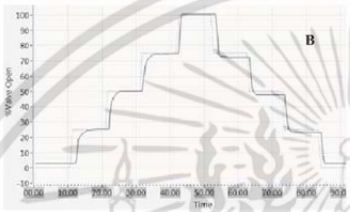



Fig. 11 Step response of EPM Fisher DVC6000f

## 5. CONCLUSION

In this paper, the NI interface module and Labview software have been developed as diagnostic tool for control valve in the Foundation Fieldbus segment. The diagnostic tool can connect to H1 segment to configure, stroke test, and diagnostic itself. In addition, it can be developed to have more diagnosis function such as valve signature test that carried out to establish a relationship between valve travel and actuator pressure actuator by get these value from the transducer and analog output blocks of the control valve. Furthermore, the tool can operate with the several models control valves from multi manufacturers so that the tool can support end user to save cost of diagnostic tool and give a chance to enhance their knowledge on control valve which can lead to an easier and more precise fault diagnosis.

## REFERENCES

[1] N. P. Mahalik, Fieldbus Technology: Industrial Network Standards for Real-Time Distributed Control, Springer-Verlag Berlin Heidelberg, 2003.

[2] R. Zurawski, The Industrial Communication Technology Handbook, CRC Press, 2005.

[3] Filedbus Foundation, http://www.fieldbus.org

[4]Sawai Pongswatd, Amphawan Julsereewong, Srinakorn Nontanakorn, Hisashi Sasajima, Tsuyoshi Kitai, "Education of Automation Infrastructure Based on International Standards", SICE Annual Conference 2010 (SICE 2010), August 18-21, 2010 Taipei, Taiwan.

[5]Teerawat Thepmanee, Sawai Pongswatd, Amphawan Julsereewong and Prasit Julsereewong, "Foundation Fieldbus Based Laboratory: A Case Study of Harmonas-DEO System", SICE Annual Conference 2011. 13-18 September 2011. Tokyo, Japan.

[6]Verhappen, I. and A. Pereira. 2008. Foundation Fieldbus 3rd Edition. ISA, 67 Alexander Drive P.O. Box 12277 Research Triangle Park, NC 27709.

[7]Pee Suat Hoon, "Developing Valve Diagnostic Application for Foundation Fieldbus Devices", ASEAN Virtual Instrument Applications Contest Submission 2006.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ประวัติผู้เขียน

นายเจษฎา เจริญวัฒนโยธิน เกิดเมื่อวันที่ 6 ธันวาคม พ.ศ. 2528 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมการวัดคุม จากภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2551 และเข้าศึกษาต่อในระดับปริญญาโท หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมอัตโนมัติ ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยี-พระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้