การประยุกต์ระบบสการ์ด้ากับการควบคุมชุดจำลองการทำงานใน ห้องปฏิบัติการ PLC

Applying SCADA System Base on PLC Control with Simulation Kit in PLC Laboratory

ณัฐดนัย โกสุจริต

ณัฐภูมิ โชติเอี่ยม

ธัญชนิต วรคามิน

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557

การประยุกต์ระบบสการ์ด้ากับการควบคุมชุดจำลองการทำงานใน ห้องปฏิบัติการ PLC

Applying SCADA System Base on PLC Control with Simulation Kit in PLC Laboratory

ณัฐดนัย โกสุจริต

ณัฐภูมิ โชติเอี่ยม

ธัญชนิต วรคามิน

ญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษาที่ 2557

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Applying SCADA System Base on PLC Control with Simulation Kit in PLC Laboratory

Natdanai Gosujarich

Nattapoom Chotieiam

Tanchanit Varakamin

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERRING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUE OF TECHNOLOGY LARDKRABANG

ACADEMIC YEAR 2014

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2557

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์ การประยุกต์ระบบสการ์ด้ากับการควบคุมชุดจำลองการทำงานในห้องปฏิบัติการ PLC

Applying SCADA System Base on PLC Control with Simulation Kit in PLC Laboratory

นักศึกษาผู้จัดทำ นายณัฐดนัย โกสุจริต **รหัสนักศึกษา** 54010400

นายณัฐภูมิ โชติเอี่ยม **รหัสนักศึกษา** 54010445

นายธัญชนิต วรคามิน **รหัสนักศึกษา** 54010611

ปริญญา วิศวกรรมศาสตรบัณฑิต

สาขาวิชา วิศวกรรมการวัดคุม

ปีการศึกษา 2557

อาจารย์ผู้ควบคุมปริญญานิพนธ์	ลายมือชื่อ
รศ. สุพรรณ กุลพานิชย์	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การประยุกต์ระบบสการ์ด้ากับการควบคุมชุดจำลองการทำงานในห้องปฏิบัติการ PLC Applying SCADA System Base on PLC Control with Simulation Kit in PLC Laboratory
นักศึกษาผู้จัดทำ	นายณัฐดนัย โกสุจริต **รหัสนักศึกษา** 54010400 นายณัฐภูมิ โชติเอี่ยม **รหัสนักศึกษา** 54010445 นายธัญชนิต วรคามิน **รหัสนักศึกษา** 54010611
ปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมการวัดคุม
อาจารย์ที่ปรึกษา	รองศาสตราจารย์สุพรรณ กุลพานิชย์
ปีการศึกษา	2557

บทคัดย่อ

ระบบ SCADA เป็นระบบที่ใช้ในการควบคุมสู่ศูนย์กลางที่มีการกระจายเครื่องควบคุมไปยังตำแหน่งต่างๆ ภายในบริเวณพื้นที่การควบคุม เช่น โรงงานอุตสาหกรรม โรงกลั่นน้ำมัน เป็นต้น ระบบสื่อสารเครือข่ายแบบ Ethernet ที่สื่อสารกันระหว่าง Client-Server จึงสอดคล้องและเหมาะสมกับการประยุกต์ใช้งาน สำหรับงานวิจัยฉบับนี้ขอนำเสนอ การประยุกต์ระบบสการ์ด้ากับการควบคุมชุดจำลองการทำงานในห้องปฏิบัติการ PLC โดยมีคอมพิวเตอร์ทำหน้าที่เป็น Server ทำงานร่วมกับโปรแกรม WinCC เป็นฐานข้อมูลหลัก ให้กับเครื่องควบคุม PLC S7-300 ในแต่ละสถานีกระจายอยู่ในห้องปฏิบัติการ โดยมีชุดจำลองกระบวนการทางอุตสาหกรรมเป็นโมเดลให้ทดลองควบคุมได้อย่างสมจริง ในระหว่างที่มีการควบคุมสามารถเฝ้ามองและติดตามค่าตัวแปรกระบวนการแบบ On-line บนเครื่องคอมพิวเตอร์หลักเพียงเครื่องเดียวครอบคลุมเครื่องควบคุม PLC ในเครือข่ายทั้งหมดจึงกล่าวได้ว่าระบบที่ออกแบบสามารถประยุกต์ใช้งานสนับสนุนการทดลองในห้องปฏิบัติการได้อย่างสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Applying SCADA System Base on PLC Control with Simulation Kit in PLC Laboratory
Authors	Mr. Natdanai Gosujarich Mr. Nattapoom Chotieiam Mr. Tanchanit Varakamin
Thesis Advisor	Asso. Prof. Supan Gulpanich
Year	2014

ABSTRACT

SCADA systems are used to control system to distributed central controller to a different position with in the control area, such as an industrial plant, Oil refinery, etc. Ethernet network communications systems that communicate with each other between Client-Server. Therefore, it is consistent and appropriate for the applying in use. For this research presents the application of the SCADA systems to control a set of simulators in the PLC laboratory. The computer acts as a server works with the WinCC software as the primary database to monitor PLC S7-300 in each station that distributed in the laboratory and a set of industrial process simulation model to experiment with realistic control. During that control can monitor and follow the process variables in On-line on the single host computer cover all PLC in networks.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้จัดทำสำเร็จลุล่วงได้ด้วยความเมตตากรุณาจากบุคคลหลายท่าน ซึ่งไม่สามารถนำมากล่าวได้ทั้งหมด โดยผู้มีพระคุณอย่างสูงที่สุดที่คณะผู้จัดทำใคร่ขอกราบขอบพระคุณจากท่านรองศาสตราจารย์สุพรรณ กุลพานิชย์ อาจารย์ที่ปรึกษาปริญญานิพนธ์ที่ได้ให้คำแนะนำมาโดยตลอดอีกทั้งเอื้อเฟื้ออุปกรณ์และเครื่องมือต่างๆในการทำปริญญานิพนธ์นี้ ผู้วิจัยรู้สึกซาบซึ้งและขอกราบขอบ พระคุณเป็นอย่างสูง ณ ที่นี้

ขอขอบพระคุณอาจารย์ภาควิชาวิศวกรรมการวัดคุมทุกท่านที่ให้คำแนะนำและเป็นกำลังใจในการทำวิจัยตลอดมา

ขอขอบพระคุณภาควิชาวิศวกรรมการวัดคุมคณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่เอื้อเฟื้อสถานที่ในการทำวิจัย

สุดท้ายนี้ขอกราบขอบพระคุณคุณพ่อคุณแม่ ที่ให้การสนับสนุน คอยให้กำลังใจ และเป็นแรงบัลดาลใจในการทำปริญญานิพนธ์เสมอมา คุณความดีที่พึงมีจากการทำปริญญานิพนธ์ ผู้วิจัยขอมอบแด่ คุณพ่อ คุณแม่ รวมทั้งคณาจารย์และผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I
บทคัดย่อภาษาอังกฤษ II
กิตติกรรมประกาศ III
สารบัญ IV
สารบัญตาราง
สารบัญภาพ

บทที่ 1 บทนำ 1
1.1 ที่มาและความสำคัญของปัญหาการวิจัย 1
1.2 วัตถุประสงค์ของการวิจัย 1
1.3 ขอบเขตของการวิจัย 1
1.4 ขั้นตอนการดำเนินงาน 2
1.5 ประโยชน์ที่ได้รับจากงานวิจัย 2

บทที่ 2 ทฤษฎีและโครงงานที่เกี่ยวข้อง 3
2.1 ระบบ SCADA 3
2.1.1 โครงสร้างของ SCADA (Architecture) 3
2.1.2 โครงสร้างด้านซอฟต์แวร์ (Software Architecture) 4
2.1.3 โครงสร้างด้านการสื่อสาร (Communications) 5
2.1.4 โครงสร้างอินเทอร์เฟส (Interface) 6
2.1.5 โครงสร้างความสามารถในการขยายระบบ (Scalability) 6
2.1.6 โครงสร้างการสำรองระบบ (Redundancy) 6
2.1.7 หน้าที่การทำงาน (Functionality) 7
2.1.8 ระบบแสดงผลแบบ MMI (Man Machine Interace) 7
2.1.9 ระบบแสดงกราฟสัญญาณแบบต่อเนื่อง 7
2.1.10 ระบบแจ้งเตือน (Alarm) 7
2.1.11 การทำงานแบบอัตโนมัติ 8
2.1.12 การสร้างและพัฒนา (Application Development) 8
2.1.13 เครื่องมือในการพัฒนา (Development Tool) 8
2.2 โครงสร้างพื้นฐานพีแอลซี 9
2.2.1 การจำแนกอุปกรณ์ที่เชื่อมต่อ 10
2.3 OPC (Ole for Process Control) 13
2.4 ภาษา Visual Basic 17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 3 ขั้นตอนการดำเนินงาน 20

3.1 การเชื่อมต่ออุปกรณ์ทางด้านฮาร์ดแวร์ 20

3.1.1 การเชื่อมต่อระหว่าง Module Local กับ Remote ของพีแอลซี 20

3.1.2 การเชื่อมต่อระหว่างพีแอลซีกับแผงควบคุม 20

3.1.3 การเชื่อมต่อระหว่างคอมพิวเตอร์กับเครื่องพีแอลซี 21

3.2 การเชื่อมต่ออุปกรณ์ทางด้านซอฟต์แวร์ 22

3.2.1 การตั้งค่าเครื่องพีแอลซีเชื่อมต่อกับคอมพิวเตอร์ 22

3.2.2 การตั้งค่าโปรแกรมสำหรับการเชื่อมต่อ OPC Server 23

3.2.3 การตั้งค่าโปรแกรมสำหรับการเชื่อมต่อ SCADA 24

3.3 การเขียนโปรแกรมและกราฟิกการทำงานของระบบ 25

3.3.1 การเขียนโปรแกรมสำหรับควบคุมพีแอลซี 25

3.3.2 การเขียนกราฟิกสำหรับการแสดงผลและเทรนไลน์ 25

3.4 ขั้นตอนการทำงานของระบบ SCADA 33

3.4.1 ขั้นตอนในการติดต่อกันระหว่างพีแอลซีและ OPC Server 34

3.4.2 ขั้นตอนในการติดต่อกันระหว่าง WinCC และ OPC Server 37

บทที่ 4 ผลการทดลอง 39

4.1 การทดสอบสับสวิทช์ที่แผงควบคุม 39

4.1.1 ลำดับขั้นตอนการทดลอง 39

4.1.2 ผลการทดลอง 40

4.2 การทดสอบการแสดงผลแบบดิจิตอล 44

4.2.1 ลำดับขั้นตอนการทดลอง 44

4.2.2 ผลการทดลอง 45

4.3 การทดสอบการแสดงผลแบบอนาล็อก 47

4.3.1 ลำดับขั้นตอนการทดลอง 47

4.3.2 ผลการทดลอง 48

4.4 การทดสอบการแสดงผลเมื่อใช้การเขียนภาษาแลดเดอร์ขั้นพื้นฐาน 50

4.4.1 ลำดับขั้นตอนการทดลอง 50

4.4.2 ผลการทดลอง 53

4.5 อภิปรายผลการทดลอง 55

บทที่ 5 สรุปผลและข้อเสนอแนะ 56

5.1 สรุปผล 56

5.2 ปัญหาของการวิจัยและการทดลอง 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

5.3 ข้อเสนอแนะ 56

บรรณานุกรม 57

ภาคผนวก 58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
3.1 ค่ารายละเอียดของ Tag ในพีแอลซีทั้งสองเครื่อง	35
4.1 การทดลองแบบที่ 1	40
4.2 ผลการทดลองแบบที่ 2	41
4.3 ผลการทดลองแบบที่ 3	42
4.4 ผลการทดลองแบบที่ 4	43
4.5 ผลการทดลองแบบดิจิตอล	45
4.6 ผลการทดลองข้อมูลแบบอนาล็อก (ดิจิตอลโวลต์มิเตอร์)	48
4.7 ผลการทดลองที่เป็นไปตามเงื่อนไขของโปรแกรมในแต่ละข้อ	53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 โครงสร้างด้านฮาร์ดแวร์ของระบบ SCADA	4
2.2 โครงสร้างด้านซอฟต์แวร์ของระบบ SCADA	5
2.3 PLC Siemens S7-300 รุ่น CPU 314C-2PN/DP	9
2.4 Power Supply ขนาด 24 โวลต์ 5 แอมแปร์	10
2.5 DP Master และ DP Slave	11
2.6 สถานะต่างๆของ PLC เมื่อเปิดโหมด Run	12
2.7 การเชื่อมต่อข้อมูลระหว่างโปรแกรมการใช้งานแม่ข่าย	14
2.8 การเชื่อมต่อข้อมูลระหว่างโปรแกรมการใช้งานบนลูกข่าย	14
2.9 การอ่านข้อมูลจากแม่ข่ายหลายชุด	15
2.10 การประมวลผลข้อมูลจากแม่ข่ายเดียวกัน	15
2.11 การเชื่อมต่อคอมพิวเตอร์เข้ากับเครือข่ายโดยใช้ฮับ	17
3.1 แผนผังการเชื่อมต่อของอุปกรณ์ฮาร์ดแวร์ต่างๆ	19
3.2 แผนภาพการเชื่อมต่อระหว่างอุปกรณ์ Local และ Remote ของพีแอลซี	20
3.3 แผนภาพการเชื่อมต่อระหว่างพีแอลซีกับแผงควบคุม	20
3.4 แผนภาพการเชื่อมต่อระหว่างพีแอลซีกับคอมพิวเตอร์สองเครื่อง	21
3.5 การเชื่อมต่ออุปกรณ์ทางด้านฮาร์ดแวร์	21
3.6 แผนภาพขั้นตอนการเชื่อมต่ออุปกรณ์ทางด้านซอฟต์แวร์	22
3.7 โปรแกรมที่ใช้ในการกำหนดโครงสร้างของพีแอลซี	22
3.8 การเชื่อมเครือข่ายผ่านระบบอีเทอร์เน็ตและโปรฟิบัสดีพีของพีแอลซีทั้งสองเครื่อง	23
3.9 การเลือกไดร์ฟเวอร์อุปกรณ์ในการตั้งค่าสำหรับการเชื่อมต่อ OPC Server	23
3.10 หน้าต่างหลักของโปรแกรม WinCC Explorer	24
3.11 หน้าต่างสำหรับการเขียนโปรแกรมภาษาแลดเดอร์	25
3.12 หน้าต่าง Graphics Designer ของโปรแกรม WinCC Explorer	26
3.13 แผงควบคุมต้นแบบที่ใช้ในงานวิจัย	26
3.14 กราฟิกแผงควบคุมจำลองที่ใช้ในงานวิจัย	27
3.15 หน้าต่างสำหรับการเขียนโปรแกรมภาษา VBScript	27
3.16 ผังงานการทำงานของการแสดงผลดิจิตอลโวลต์มิเตอร์ทั้ง8ช่อง	28
3.17 แผนผังกระบวนการทำงานของดิจิตอลเอาท์พุทของแต่ละตำแหน่ง	29
3.18 แผนผังกระบวนการทำงานของดิจิตอลเอาท์พุทของแต่ละตำแหน่ง(ต่อ)	30
3.19 แผนผังกระบวนการทำงานของดิจิตอลเอาท์พุทของแต่ละตำแหน่ง(ต่อ)	31
3.20 แผนผังกระบวนการทำงานของดิจิตอลเอาท์พุทของแต่ละตำแหน่ง(ต่อ)	32
3.21 หน้าต่างสำหรับเลือกใช้ WinCCOnlineTrendControl	33
3.22 ตัวอย่างหน้าต่างของเทรนกราฟเมื่ออยู่ใน Runtime Mode	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.23 แผนผังกระบวนการร้องขอข้อมูลจากแต่ละตำแหน่งของระบบ SCADA 34
3.24 ตำแหน่งของ Quick Client 34
3.25 หน้าต่างของ Quick Client 37
3.26 OPC Server Name ที่ต้องใส่สำหรับการทดสอบ 37
3.27 การทดสอบ Test Server ว่าเชื่อมต่อกันได้ 38
4.1 ผลการทดลองแบบที่ 1 ของแผงควบคุมจำลอง 40
4.2 ผลการทดลองแบบที่ 2 ของแผงควบคุมจำลอง 41
4.3 ผลการทดลองแบบที่ 3 ของแผงควบคุมจำลอง 42
4.4 ผลการทดลองแบบที่ 4 ของแผงควบคุมจำลอง 43
4.5 โปรแกรมทดสอบสัญญาณดิจิตอล 44
4.6 ตัวอย่างผลการทดลองข้อมูลแบบดิจิตอล 45
4.7 ตัวอย่างผลการทดลองข้อมูลแบบดิจิตอล(ต่อ) 46
4.8 ตัวอย่างผลการทดลองข้อมูลแบบดิจิตอล(ต่อ) 46
4.9 ตัวอย่างผลการทดลองข้อมูลแบบดิจิตอล(ต่อ) 46
4.10 โปรแกรมทดสอบสัญญาณอนาล็อก 47
4.11 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์(1V) 47
4.12 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์(2.5V) 49
4.13 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์(5V) 49
4.14 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์(7.5V) 49
4.15 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์(9V) 50
4.16 แลดเดอร์พื้นฐาน 1 51
4.17 แลดเดอร์พื้นฐาน 2 51
4.18 แลดเดอร์พื้นฐาน 3 51
4.19 แลดเดอร์พื้นฐาน 4 52
4.20 แลดเดอร์พื้นฐาน 5 52
4.21 แลดเดอร์พื้นฐาน 6 52
4.22 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 1 ณ แผงควบคุมจำลอง 53
4.23 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 2 ณ แผงควบคุมจำลอง 54
4.24 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 3 ณ แผงควบคุมจำลอง 54
4.25 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 4 ณ แผงควบคุมจำลอง 54
4.26 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 5 ณ แผงควบคุมจำลอง 55
4.26 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 6 ณ แผงควบคุมจำลอง 55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหาการวิจัย

เนื่องจากตัวควบคุมในงานอุตสาหกรรมที่หลากหลายส่วนใหญ่ โปรแกรมเมเบิลลอจิกคอลโทรลเลอร์ (Programmable Logic Controller ; PLC) ที่เป็นรุ่นปัจจุบันทั้งหลาย จะมีระบบสื่อสารเป็นแบบระบบอีเทอร์เน็ตแทบทั้งสิ้นมีความเร็วและเสถียรภาพในการสื่อสารที่มีประสิทธิภาพ ดังนั้นโครงงานวิจัยนี้ที่มีการประยุกต์เอาเครื่องควบคุม PLC เข้ามาใช้ในการควบคุมและมีระบบสื่อสารเป็นแบบอีเทอร์เน็ตนี้ จึงมีความน่าสนใจสำหรับการทำโครงงานวิจัยและน่าจะเป็นประโยชน์สำหรับงานอุตสาหกรรมที่เกิดขึ้นใน

1.2 วัตถุประสงค์ของการวิจัย

อุปกรณ์ที่เกี่ยวข้องทั้งทฤษฎีและหลักการของโปรแกรมเมเบิลลอจิกคอลโทรลเลอร์ (PLC) และซอฟต์แวร์โดยประยุกต์ใช้อุปกรณ์อิเล็กทรอนิกส์ร่วมกับอุปกรณ์ทางการวัดและควบคุมรวมไปถึงระบบ SCADA ที่ใช้กันอยู่สำหรับงานอุตสาหกรรม

1.3 ขอบเขตของการวิจัย

1.3.1 ศึกษาโครงสร้างที่สำคัญของโปรแกรมเมเบิลลอจิกคอลโทรลเลอร์ (PLC)

1.3.2 ศึกษาการใช้งานและคุณสมบัติของ PLC Siemens S7-300

1.3.3 ศึกษาอุปกรณ์เชื่อมต่อระหว่าง PLC กับ Human Machine Interface (HMI)

1.3.4 ศึกษาการนำ OPC Server ที่เชื่อมต่อผ่านระบบอีเทอร์เน็ตเพื่อให้คอมพิวเตอร์ที่อยู่ในเครือข่ายเดียวกันสามารถอ่านข้อมูลจาก OPC Server นำมาแสดงผลของกระบวนการในห้องควบคุม

1.3.5 ศึกษาส่วนของซอฟต์แวร์ที่ใช้สำหรับอุปกรณ์ HMI ที่ติดตั้งอยู่บริเวณเครื่องจักรในสถานที่ตำแหน่งต่างๆของโรงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนการดำเนินงาน

1.4.1 ศึกษาทฤษฎีต่างๆของโครงงาน เช่น โครงสร้างของพีแอลซี ระบบ SCADA และการเชื่อมต่อผ่านระบบอีเทอร์เน็ตซึ่งเป็น OPC Server

1.4.2 เชื่อมต่ออุปกรณ์ทางด้านฮาร์ดแวร์

1.4.3 เชื่อมต่ออุปกรณ์ทางด้านซอฟต์แวร์

1.4.4 เขียนโปรแกรมและกราฟิกของแผงควบคุม

1.4.5 ทดสอบระบบ SCADA และกราฟิกแผงควบคุมตามแผนการทดลอง

1.4.6 สรุปผลงานวิจัย

1.5 ประโยชน์ที่ได้รับจากงานวิจัย

1.5.1 การใช้โปรแกรม SIMATIC Manager STEP7

1.5.2 การใช้โปรแกรม Window Control Center (WinCC) เพื่อการควบคุมแบบ SCADA ผ่านตัวกลาง OPC

1.5.3 ศึกษาการทำงานและการขยาย I/O รวมไปถึงการติดตั้งระบบของ PLC Siemens S7-300 และคอมพิวเตอร์ที่เป็นตัวเซิร์ฟเวอร์

1.5.4 การสร้างงานกราฟฟิคเพื่อประโยชน์ในการแสดงผล

1.5.5 ศึกษาและการเชื่อมต่อตัว PLC ผ่านระบบเครือข่ายอีเทอร์เน็ตในรูปแบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

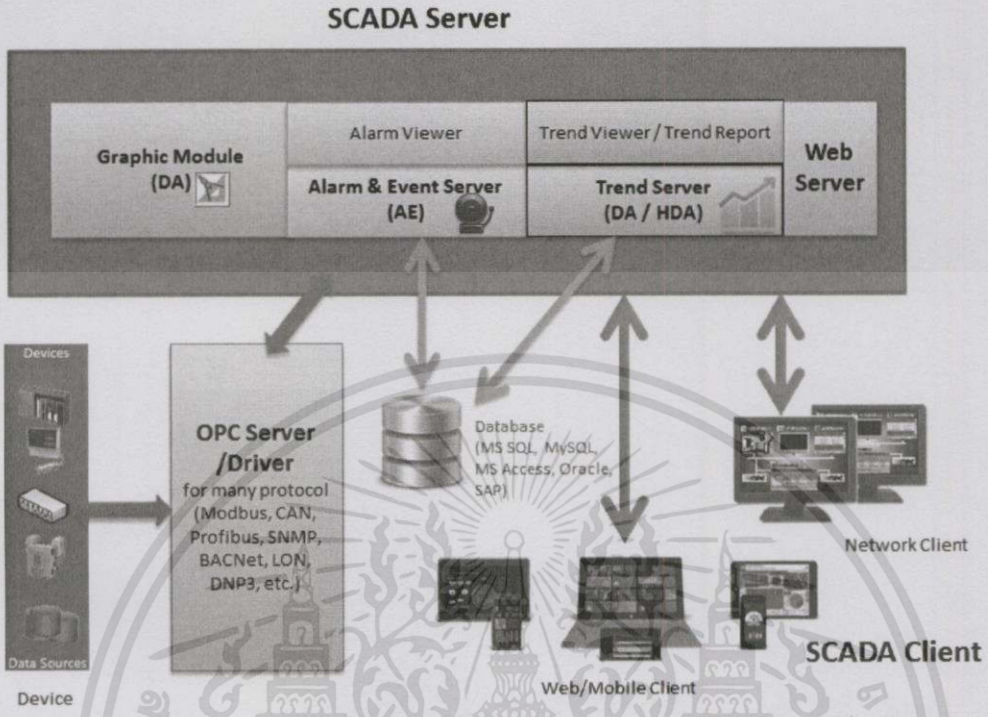
2.1 ระบบ SCADA

SCADA นั้นย่อมาจากคำว่า Supervisory Control And Data Acquisition เป็นระบบตรวจสอบและวิเคราะห์ข้อมูลแบบ Real-time ใช้ในการตรวจสอบสถานะตลอดจนถึงควบคุมการทำงานของระบบควบคุมในอุตสาหกรรมและงานวิศวกรรมต่างๆ เช่น งานด้านโทรคมนาคมสื่อสาร การประปา การบำบัดน้ำเสีย การจัดการด้านพลังงาน อุตสาหกรรมการกลั่นน้ำมันและก๊าซ อุตสาหกรรมเคมี อุตสาหกรรมประกอบยนต์ การขนส่ง กระบวนการนิวเคลียร์ในโรงไฟฟ้า เป็นต้น ตัวอย่างการใช้งานเช่นใช้ SCADA ตรวจสอบข้อมูลเช่น การรั่วไหลของของเหลวที่เกิดขึ้นในท่อขนส่งจากตัวตรวจจับแล้วส่งสัญญาณแจ้งเตือนให้พนักงานทราบ โดยส่งข้อมูลสู่ส่วนกลางของระบบ SCADA เป็นต้นนอกจากนั้น SCADA อาจทำหน้าที่คำนวณและประมวลผลข้อมูลที่ได้จากฮาร์ดแวร์ต่างๆ เช่น PLC, Controller, DCS, RTU แล้วแสดงข้อมูลทางหน้าจอ หรือส่งสัญญาณควบคุมฮาร์ดแวร์ดังกล่าว เช่นหากอุณหภูมิของอุปกรณ์สูงเกินพิกัดให้ทำการปิดอุปกรณ์นั้นเป็นต้น โดยสั่งงานผ่าน PLC หรือ Controller ที่ติดต่ออยู่ ทั้งนี้ SCADA สามารถเก็บรวบรวมข้อมูลที่ได้จากระบบควบคุมทั้งหมดไว้ในฐานข้อมูลเพื่อให้พนักงานหรือโปรแกรมอื่น ๆ สามารถนาไปใช้งานได้ SCADA นั้นเข้าไปมีส่วนในงานควบคุมทั้งเล็กและใหญ่ที่ต้องการแสดงผล แลกเปลี่ยนข้อมูลหรือควบคุมระบบต่าง ๆ จากส่วนกลาง เพื่อการทางานของระบบรวมที่สัมพันธ์กัน มองเห็นภาพรวมได้อย่างชัดเจนและมีความรวดเร็วต่อเหตุการณ์ต่าง ๆ ที่เกิดขึ้น ระบบ SCADA ในปัจจุบันมีความสามารถในการสื่อสาร ควบคุม และประมวลผลข้อมูลจาก I/O ของอุปกรณ์เช่น PLC, DCS, RTU ได้ถึงระดับที่เกินหนึ่งแสน I/O แล้ว และได้รับการพัฒนาให้มีความสามารถรองรับความต้องการใหม่ ๆ ของผู้ใช้งานอย่างต่อเนื่องตลอดมา

2.1.1 โครงสร้างของ SCADA (Architecture)

SCADA แบ่งตามโครงสร้างฮาร์ดแวร์ได้สองระดับคือ Client และ Data Server หรือเรียกสั้น ๆ ว่าServer โดยที่ Client คือคอมพิวเตอร์ที่รับและส่งข้อมูลไปยัง Data Server โดยฝั่ง Client นี้จะแสดงผลการทางานของระบบควบคุมเช่น แสดงเป็นกราฟิก กราฟแบบต่อเนื่อง หรือระบบแจ้งเตือนเมื่อเกิดเหตุการณ์ฉุกเฉินหรือต้องการแจ้งเตือน เป็นต้น ฝั่ง Client สามารถสั่งงานควบคุมไปยัง Data Server เพื่อส่งสัญญาณไปยัง PLC, DCS หรือ Controller อีกทอดหนึ่ง ส่วน Data Server จะทาหน้าที่ติดต่อกับ PLC, DCS, Controller หรือ RTU ต่าง ๆ เพื่อรับสัญญาณและส่งสัญญาณไปยัง Client และรับการร้องขอจาก Client เพื่อควบคุมอุปกรณ์ PLCและ Controller ต่าง ๆ Client และ Data Server ส่วนใหญ่ติดต่อกันผ่านระบบเครือข่าย Ethernet ดังรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

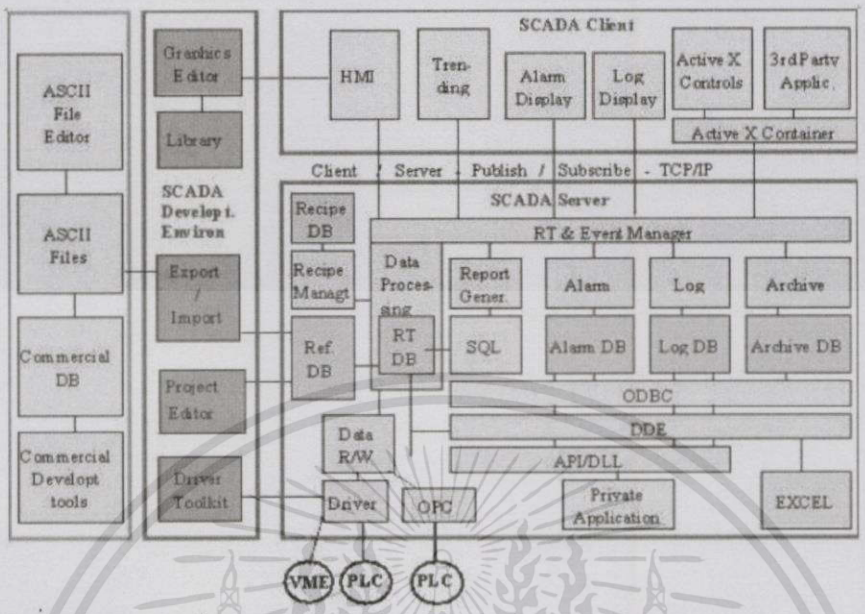


รูปที่ 2.1 โครงสร้างด้านฮาร์ดแวร์ของระบบ SCADA
ที่มา : http://www.eda.co.th/scada.html (09/04/2558)

2.1.2 โครงสร้างด้านซอฟต์แวร์ (Software Architecture)

โครงสร้างด้านซอฟต์แวร์ของระบบ SCADA นั้นมีข้อที่ต้องทราบคือ SCADA ใช้เทคโนโลยีในการสื่อสารกับฮาร์ดแวร์ (เช่น PLC, DCS) ต่าง ๆ กันไปตามผู้ผลิต เช่นการใช้ Driver เฉพาะของผู้ผลิต SCADA เพื่อสื่อสารกับ PLC, DCS เป็นต้น ซึ่งในปัจจุบันมีการกำหนดมาตรฐานกลางคือ OPC ขึ้นมาเพื่อยุติปัญหาการใช้เทคโนโลยีเฉพาะด้านในการสื่อสาร นอกจากนั้นยังมีความสามารถในการบริการข้อมูลให้กับ Client ที่รวดเร็วและมีเสถียรภาพ โครงสร้างด้านซอฟต์แวร์ของ SCADA แสดงได้ดังรูปที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 โครงสร้างด้านซอฟต์แวร์ของระบบ SCADA

ที่มา : http://www.scadathai.com/articles/475019/What's-SCADA.html (08/09/2558)

จากรูปที่ 2.2 สังเกตได้ว่าในส่วนของ SCADA Server นั้น การติดต่อกับ PLC หรือ Controller นั้น ทำได้ทั้งผ่าน Driver หรือ OPC โดยที่ OPC และ Driver สามารถรับคำสั่งแบบ Read / Write เพื่ออ่านข้อมูลจาก PLC หรือ เขียนข้อมูลเพื่อสั่งงานไปยัง PLC ได้

SCADA Server ทำหน้าที่จัดการข้อมูล RTDB (Real Time Data Base) ที่ได้จาก PLC แล้วส่งให้กับ SCADA Client โดยที่ SCADA Server บางประเภทจะติดต่อกับ SCADA Client ผ่าน DDE Server ซึ่งทำให้สามารถนำเข้าข้อมูลจาก PLC เข้าสู่โปรแกรมเช่น MS Excel หรือ โปแกรม Client อื่น ๆ ที่ติดต่อกับ DDE Server ได้ SCADA บางตัวจะออกแบบให้ SCADA Server ทำหน้าที่ตรวจจับ Alarm และเก็บไว้ใน Alarm DB หรือเก็บข้อมูลที่เป็น Historian ไว้ใน Log DB เป็นต้นเพื่อส่งให้ Alarm Display และ Log Display ทางฝั่ง SCADA Client ต่อไป

สำหรับส่วน Development Environment นั้นจะขึ้นอยู่กับการออกแบบของ SCADA ซอฟต์แวร์นั้น ๆ ซึ่งโดยทั่วไปก็จะมีเครื่องมือในการสร้างและจัดการกราฟิก (Graphic Editor) เครื่องมือในการจัดการโปรเจ็คที่สร้างขึ้นมา (Project Editor) มีเครื่องมือในการนำเข้าและส่งออก Text file ที่เก็บค่าคอนฟิกูเรชั่นของการติดต่อกับ Driver หรือ OPC Server ไว้

2.1.3 โครงสร้างด้านการสื่อสาร (Communications)

การสื่อสารระหว่าง Client-Server จะสื่อสารผ่านโปรโตคอลโดยทั่วไปเช่น TCP/IP โดย Client จะติดต่อกับพารามิเตอร์หรือ Tag ภายใน Server ที่บริการข้อมูลด้วยรูปแบบที่แตกต่างกันไปตามผู้ผลิต เช่นมีการส่งค่าจาก Server เมื่อค่าของ I/O ของ PLC มีการเปลี่ยนแปลง เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสื่อสารกับอุปกรณ์นั้น Server จะทำการตรวจสอบค่าจากอุปกรณ์ตามช่วงเวลาที่ผู้ใช้งานกำหนดไว้ (Defined polling rate) โดยอาจจะต่างกันไปตามพารามิเตอร์ประเภทต่าง ๆ โดยตัว Controller จะส่งค่าพารามิเตอร์ตามที่ถูกร้องขอให้กับ Data Server พร้อมค่าเวลาขณะนั้น (Time Stamp) การสื่อสารกับอุปกรณ์ของ Data Server นั้นอาจเป็นการสื่อสารแบบ Modbus, Profibus, CAN bus เป็นต้น ขึ้นอยู่กับมาตรฐานการสื่อสารของอุปกรณ์นั้น ๆ ว่าเป็นแบบใด ในปัจจุบันมีการสร้าง OPC Server ที่สนับสนุนการติดต่อด้วยมาตรฐานต่างๆเพิ่มขึ้นมากมายจนครอบคลุมอุปกรณ์ทุกประเภท และมีการพัฒนาให้ทั่วถึงไปยังอุปกรณ์ใหม่ ๆ อย่างต่อเนื่อง

2.1.4 โครงสร้างอินเทอร์เฟส (Interface)

การติดต่อระหว่าง Data Server กับอุปกรณ์หรือระหว่าง Data Server และ Data Server และกับ Client นั้น มีการผลิตเป็น Driver ออกมามากมายตามเทคนิคเฉพาะของแต่ละผู้ผลิต ต่อมาจึงมีการกำหนดมาตรฐานของอินเทอร์เฟสขึ้นมาเป็น OPC (OLE for Process Control) ซึ่งมีความรวดเร็วในการสื่อสารและบริการข้อมูลโดยมีการจัดตั้ง OPC Foundation ขึ้นเป็นองค์กรหลักในการกำหนดมาตรฐานและถ่ายทอดเทคโนโลยีให้แก่สมาชิก OPC จึงเป็นมาตรฐานกลางที่เปิดกว้างมากที่สุด

การติดต่อกับฐานข้อมูลภายนอกของ SCADA Software นั้น มีการสร้างให้สามารถติดต่อได้ผ่าน ODBC (Open Data Base Connectivity), OLEDB (Linking and Embedding Data Base), DDE (Dynamic Data Exchange) เป็นต้น เพื่อให้สามารถแลกเปลี่ยนข้อมูลหรือทำการเก็บข้อมูลไว้ในฐานข้อมูลรูปแบบต่าง ๆ ในปัจจุบันมีการพัฒนาให้สามารถติดต่อกับโปรแกรม ERP ต่าง ๆ เช่น SAP เป็นต้นได้ด้วย

2.1.5 โครงสร้างความสามารถในการขยายระบบ (Scalability)

Scalability คือความสามารถในการรองรับและต่อขยายระบบ SCADA กับส่วนต่าง ๆ เช่น I/O ของอุปกรณ์Controller และจำนวนเครื่อง SCADA Client ที่เพิ่มขึ้น หรือการต่อพ่วงกับระบบ SCADA ของยี่ห้ออื่น ๆ เป็นต้น ถ้าหาก Data Server เป็นแบบ Driver ที่สร้างด้วยเทคโนโลยีเฉพาะในการติดต่อกับอุปกรณ์ ก็เป็นเรื่องลำบากในการต่อขยาย เพราะ Driver บางประเภทสามารถติดต่อได้เฉพาะ SCADA Software บางยี่ห้อเท่านั้น ปัญหานี้เป็นที่วิพากย์ วิจารณ์กันอย่างกว้างขวาง ซึ่งปัจจุบันได้หันมาใช้มาตรฐานกลางคือOPCเพื่อแก้ไขปัญหานี้

2.1.6 โครงสร้างการสำรองระบบ (Redundancy)

SCADA Software ส่วนใหญ่มีความสามารถในการทาสารองระบบของ Data Server โดยที่เมื่อ Data Server เกิดความขัดข้องก็จะสั่งงานให้ Data Server อีกตัวหนึ่งทำงานแทนที่ โดยจะมีการกำหนดคอนฟิกูเรชั่นไว้ที่ Client ว่าจะให้เลือกติดต่อกับ Data Server ตัวไหนเมื่อเกิดความขัดข้องเกิดขึ้น

ในบางครั้งโมดูลที่ทาหน้าที่จัดการด้าน Redundancy นี้อาจจะทาหน้าที่อีกประการหนึ่งคือเป็นจุดพักข้อมูลที่รับมาจาก Data Server เพื่อนำไปส่งให้กับ Client ต่าง ๆ เพราะในกรณีที่มี Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนมากติดต่ออยู่กับ Data Server ตัวเดียวนั้นอาจมีความล่าช้าในการบริการข้อมูลของ Data Server เพราะต้องให้บริการข้อมูล Client ให้ครบจำนวนก่อนที่จะไปรับข้อมูลใหม่จากอุปกรณ์มาได้ ดังนั้นโมดูลที่ทาหน้าที่ Redundant จึงทำหน้าที่เป็นจุดรับข้อมูลแล้วช่วยส่งต่อให้ Client ต่างๆ อีกทอดหนึ่ง Data Server จะได้ทำหน้าที่บริการข้อมูลให้แก่โหนดเพียงจุดเดียว จึงมีความรวดเร็วในการบริการข้อมูล

2.1.7 หน้าที่การทำงาน (Functionality)

การเข้าถึงพารามิเตอร์ของอุปกรณ์ หมายถึงความสามารถในการเข้าถึงกลุ่มของพารามิเตอร์ในอุปกรณ์เช่น I/O ของ PLC เป็นต้น ความสามารถของ Data Server ในการกำหนดว่าพารามิเตอร์ใดอ่านได้อย่างเดียวเขียนได้อย่างเดียวหรือทั้งอ่านทั้งเขียนเป็นต้น

2.1.8 ระบบแสดงผลแบบ MMI (Man Machine Interface)

คือความสามารถในการแสดงผลการทางานของอุปกรณ์ในรูปแบบ กราฟิก ข้อความ สัญลักษณ์ แผนภาพ เป็นต้น โดยสามารถเชื่อมโยงลักษณะการเปลี่ยนแปลงของกราฟิกเหล่านี้กับพารามิเตอร์จาก Data Serverได้ ความสามารถในการสั่งงานผ่านระบบกราฟิกเช่น การปิด/เปิด สวิทช์บนจอมอนิเตอร์ส่งผลไปยังI/Oของพีแอลซีเป็นต้น

ความสามารถในการจัดการกราฟิกเช่น การย่อ ขยาย การกำหนดการเคลื่อนไหวแบบต่าง ๆ เช่น การหมุน การเคลื่อนที่แบบซิกแซ็กตามสัญญาณของ Data Server การแสดงผลสัญญาณในรูปแบบมิเตอร์และเกจวัดแบบต่าง ๆ การนำเข้ากราฟิกประเภทต่างๆ การจัดแบ่งเลเยอร์ เป็นต้น สิ่งเหล่านี้เป็นข้อเปรียบเทียบความสามารถของ SCADA Software ทั้งสิ้น

2.1.9 ระบบแสดงกราฟสัญญาณแบบต่อเนื่อง (Trending)

Trending เป็นความสามารถในการพล็อตกราฟต่อเนื่องกันไปบนจอภาพเพื่อแสดงค่าสัญญาณจาก Data Server โดยอาจจะสามารถพล็อตสัญญาณได้หลายสัญญาณเช่น 8 – 24 สัญญาณพร้อมกันในหน้าต่างเดียว เพื่อให้สามารถเปรียบเทียบสัญญาณที่พล็อตได้ และไม่จำกัดว่าจะสร้างหน้าต่างพล็อตจานวนเท่าใด Trending อาจมีความสามารถในการซูมสัญญาณที่พล็อต และหยุดการพล็อตเพื่อเลื่อนดูค่าที่พล็อตในแต่ละช่วงเวลาได้ด้วยตัวของผู้ใช้งานเอง นอกจากนั้นการพล็อตอาจสามารถเลือกได้ว่าจะให้เป็นการพล็อตแบบใดเช่น Time plot, Logarithmic plot, Strip Chart, Bar Chart, Circular, X-Y plot เป็นต้น นอกจากนั้นบางผู้ผลิตยังสามารถนาค่า Historian หรือข้อมูลสัญญาณที่เก็บไว้ในฐานข้อมูลออกมาพล็อต ได้อีกด้วย โดย Trending Module นี้อาจเป็นแบบ ActiveX Control คือสามารถนำไปใช้งานในแอพพลิเคชั่นอื่นที่สนับสนุนการนำเข้าActiveXได้

2.1.10 ระบบแจ้งเตือน (Alarm)

SCADA Software ส่วนใหญ่มีระบบแจ้งเตือนโดย Alarm Display จะรับสัญญาณมาจาก Alarm DB ในฝั่ง SCADA Server โดย Alarm DB สามารถที่จะทำการกำหนดคอนฟิกูเรชั่นว่าจะนำสัญญาณตัวใดมาเป็นตัวพารามิเตอร์ในการแจ้งเตือนบ้าง และมีการแบ่งระดับของ Priority, Limit อย่างไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นต้นระบบแจ้งเตือนยังสามารถที่จะเก็บข้อมูลการแจ้งเตือนไว้ในฐานข้อมูลประเภทต่าง ๆ ได้เช่น MS SQL Server, MS Access, Oracle, MS Excel เป็นต้น และบางยี่ห้อสามารถแสดงออกมาเป็นรายงานในรูปแบบตารางหรือแผนภูมิได้อีกด้วย

2.1.11 การทำงานแบบ Automation

เป็นความสามารถที่ SCADA ทำหน้าที่ต่าง ๆ ตามที่กำหนด เช่น ส่งอีเมล์ แสดงข้อความแบบ Instance Message บนหน้าจอ เปิดไปยังหน้าจออื่น ๆ เก็บข้อมูลลงฐานข้อมูล เปิดโปรแกรม หรือรันค่าสั่งสคริปต์ เป็นต้น ตามสัญญาณที่ได้รับจาก Data Server และข้อกำหนดที่สร้างขึ้น

2.1.12 การสร้างและพัฒนา (Application Development)

การกำหนดคอนฟิกูเรชั่น ขั้นแรกต้องมีการกำหนดว่าจะติดต่อกับพารามิเตอร์หรือ Tag ใดบ้างจาก Data Server ดังนั้นจะต้องทำการ Define หรือสร้าง Tag ที่ Data Server ก่อนว่า Tag แต่ละตัวหมายถึง Address ที่เท่าใดของอุปกรณ์ (PLC, DCS, RTU, Controller ต่างๆ) โดยทั่วไปสามารถทำการนำเข้าคอนฟิกูเรชั่นไฟล์ที่สร้างไว้ก่อนเข้ามาได้ และสามารถ Export ไปยัง Data Server อื่น ๆ ได้ จากนั้นโปรแกรมย่อยอื่น ๆ ของ SCADA Software ฝั่งไคลเอนท์ จึงทำคอนฟิกูเรชั่นตามหน้าที่การทำงานของตนเอง เช่น โมดูลที่มีหน้าที่แสดงผลกราฟิกก็ต้องกำหนดว่ากราฟิกนั้น ๆ จะเชื่อมโยงกับ Tag ใดจาก Data Server ส่วนโมดูลที่ทาหน้าที่แจ้งเตือนก็ต้องทำคอนฟิกูเรชั่นว่าจะนำ Tag ใด มาเป็นสัญญาณแจ้งเตือนและกำหนดระดับสัญญาณลิมิตเป็นต้น

2.1.13 เครื่องมือในการพัฒนา (Development Tool)

เครื่องมือในการสร้างและพัฒนาระบบ SCADA โดยทั่วไปจะประกอบด้วย

1. เครื่องมือในการสร้างระบบกราฟิก ที่ประกอบด้วยเครื่องมือวาดภาพ เครื่องมือกำหนดเอฟเฟ็กพิเศษต่าง ๆ ไลบรารี่ของกราฟิกสาเร็จรูปในอุตสาหกรรมด้านต่าง ๆ
2. เครื่องมือในการสร้าง Trending
3. เครื่องมือในการสร้างระบบ Alarm
4. เครื่องมือในการกำหนดการติดต่อกับฐานข้อมูลเพื่อทาการเก็บรวบรวมข้อมูลของ Trending และ Alarm ลงไว้ในฐานข้อมูล
5. เครื่องมือในการช่วยสร้าง Script เช่น Java script, VB Script
6. เครื่องมือจัดการด้านความปลอดภัย การแบ่งระดับ User และขอบเขตการใช้งานของ User
7. เครื่องมือในการสร้าง Web application เพื่อให้สามารถควบคุมและตรวจสอบระบบควบคุมผ่าน Web browser ได้

ที่กล่าวมาทั้งหมดนี้เป็นลักษณะของ SCADA และ SCADA Software ส่วนใหญ่ ทั้งนี้คุณผู้อ่านก็คงจะพอเห็นภาพว่าSCADA นั้นสามารถเป็นศูนย์กลางของระบบควบคุมทั้งหมดขององค์กร และมีส่วนช่วยในการตรวจสอบการทางานของระบบให้เป็นไปตามปกติได้อย่างมีประสิทธิภาพและทั่วถึง ภายใน

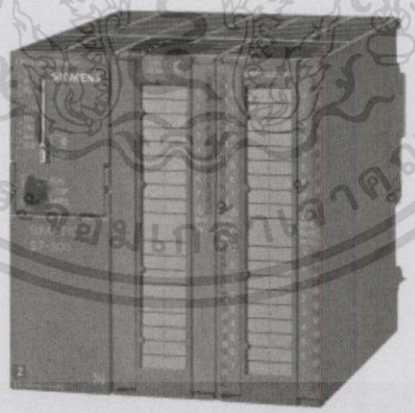
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวลาอันรวดเร็ว มีส่วนช่วยในการตัดสินใจในการดาเนินงานจากข้อมูลต่าง ๆ ที่ได้รับจากระบบ SCADA นอกจากนี้เรายังสามารถเชื่อมโยงข้อมูลที่ได้จาก SCADA เข้ากับข้อมูลทางธุรกิจอื่น ๆ เพื่อประมวลผลร่วมกัน เช่น ข้อมูลจานวนของเสียเป็นกิโลกรัมที่ตรวจสอบได้จาก SCADA ถูกนำมาคำนวณร่วมกับค่าใช้จ่ายอื่น ๆ แบบ Real time เพื่อสรุปเป็นรายงานค่าใช้จ่ายประจำวันเป็นต้นได้อย่างรวดเร็ว

2.2 โครงสร้างพื้นฐานของพีแอลซี

ในโปรแกรมเมเบิลลอจิกคอลโทรลเลอร์ (Programmable logic Control : PLC) เป็นอุปกรณ์ควบคุมการทำงานของเครื่องจักรหรือกระบวนการทำงานต่างๆ โดย Microprocessor เป็นมันสมองสั่งการที่สำคัญพีแอลซีจะมีส่วนที่เป็นอินพุทและเอาท์พุทที่สามารถต่อออกไปใช้งานได้ทันที ตัวตรวจวัดหรือสวิทตช์ต่างๆ จะต่อเข้ากับอินพุท ส่วนเอาท์พุทจะใช้ต่อออกไปควบคุมการทำงานของอุปกรณ์หรือเครื่องจักรที่เป็นเป้าหมาย เราสามารถสร้างวงจรหรือแบบของการควบคุมได้โดยการป้อนเป็นโปรแกรมคำสั่งเข้าไปในพีแอลซีนอกจากนี้ยังสามารถใช้งานร่วมกับอุปกรณ์อื่นเช่นเครื่องอ่านบาร์โค๊ด (Barcode Reader) เครื่องพิมพ์ (Printer) ซึ่งในปัจจุบันนอกจากเครื่องพีแอลซีจะใช้งานแบบเดี่ยว (Stand alone) แล้วยังสามารถต่อพีแอลซีหลายๆ ตัวเข้าด้วยกัน (Network) เพื่อควบคุมการทำงานของระบบให้มีประสิทธิภาพมากยิ่งขึ้นด้วยจะเห็นได้ว่าการใช้งานพีแอลซีมีความยืดหยุ่นมากดังนั้นในโรงงานอุตสาหกรรมต่างๆจึงเปลี่ยนมาใช้พีแอลซีมากขึ้น

ในส่วนนี้จะขอแสดงถึงโครงสร้างและคุณลักษณะเฉพาะตัวพื้นฐานโดยทั่วไปของ PLC Siemens S7-300 รุ่น CPU 314C-2PN/DP ซึ่งเป็นยี่ห้อหนึ่งที่มีมาตรฐานและสามารถใช้งานได้อย่างหลากหลายโดยพิจารณาได้ดังรูปที่ 2.3



รูปที่ 2.3 PLC Siemens S7-300 รุ่น CPU 314C-2PN/DP

เพื่อให้ CPU Siemens S7-300 ทำงานได้จำเป็นต้องมีไฟฟ้าจาก Power Supply ขนาด 24 โวลต์ 5 แอมแปร์ (รูปที่ 2.4) เข้าไปเลี้ยงวงจร CPU Siemens S7-300 เมื่อต่อสายทั้งหมดเข้าอุปกรณ์เปิด Power Supply และกด Run เครื่องจะเป็นดังรูป 2.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.4 Power Supply ขนาด 24 โวลต์ 5 แอมแปร์

การเชื่อมต่อจะต่ออุปกรณ์ต่างๆเข้าไปใน PLC นั้นจะต้องทำการแยกประเภทของอุปกรณ์

2.2.1 การจำแนกอุปกรณ์ที่เชื่อมต่อ

อุปกรณ์อินพุท (Input Devices) ในปัจจุบัน PLC ได้มีการพัฒนาให้มีความสามารถและประสิทธิภาพสูงขึ้นมากซึ่งสามารถรับสัญญาณได้ทั้งสัญญาณในรูปแบบ ON/OFF และสัญญาณ Analog ที่เป็นสัญญาณมาตรฐานต่างๆเช่น 4-20 mA, 1-5 V หรือ 0-10 V ซึ่งอุปกรณ์อินพุทที่ให้สัญญาณได้แก่ Switch Proximity, Switch Photo Sensor, Encoder Pressure Sensor, Thumbwheel Switch และ Temperature Sensor เป็นต้น

อุปกรณ์เอาท์พุท (Output Devices) สำหรับในส่วนของอุปกรณ์เอาท์พุทเป็นอุปกรณ์ที่ต้องทำการขยายสัญญาณก่อนที่ต่อใช้งานกับอุปกรณ์ในการทำงานหรือโหลดที่ต้องใช้กำลังไฟฟ้าสูงๆ เช่น มอเตอร์ฮีทเตอร์กระบอกสูบในระบบนิวเมติกส์เนื่องจากในส่วนเอาท์พุทของพีแอลซีนั้นมีความสามารถที่จะจ่ายหรือทนกระแสไฟฟ้าได้น้อยดังนั้นจึงต้องมีการนำอุปกรณ์เอาท์พุทมาต่อใช้งานร่วมด้วยได้แก่ รีเลย์, คอนแทคเตอร์, โซลินอยล์วาล์ว, หลอดไฟและคอนโทรลวาล์วเป็นต้นหลังจากการแยกประเภทของอุปกรณ์แล้วจะต้องทำการเชื่อมต่อสัญญาณโดยสัญญาณนั้นแบ่งออกด้วยกัน 2 ประเภทคือสัญญาณอินพุทและสัญญาณเอาท์พุท

ดิจิตอลอินพุทและดิจิตอลเอาท์พุท ใช้รับส่งข้อมูลกับอุปกรณ์ที่สัญญาณมีเพียง 2 สถานะ เช่น ดิจิตอลเอาท์พุท นำไปใช้ควบคุมการเปิด-ปิดอุปกรณ์จำพวกรีเลย์หรือหลอดไฟแอลอีดีว่าเปิดหรือปิด ขณะที่ดิจิตอลอินพุทใช้อ่านข้อมูลจากอุปกรณ์ เช่น สวิทช์หรือปุ่มกด ว่าอยู่ในสถานะเปิดหรือปิด เป็นต้น

อนาล็อกเอาท์พุทเป็นลักษณะการให้สัญญาณออกมาในรูปแบบของสัญญาณต่อ เนื่องที่เป็นสัญญาณมาตรฐานได้แก่สัญญาณด้านกระแส 4-20 mA สัญญาณแรงดันมาตรฐาน 0-5 V, 1-5 V, 0-10 V ฯลฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากโครงสร้างพื้นฐานที่กล่าวมาแล้วในโครงงานนี้ยังมีโมดูลเสริม Distributed I/O ET200M ที่ประกอบไปด้วย PROFIBUS terminal IM 153-1 และโมดูลอินพุทเอาท์พุท

เมื่อทราบการเชื่อมต่ออุปกรณ์ต่างๆแล้วในการทำโครงงานนี้จะทราบถึงการเชื่อมต่อระหว่าง PLC เนื่องจากในการทดลองนั้นใช้ PLC 2 เครื่องและคอมพิวเตอร์ที่เป็นเซิร์ฟเวอร์ที่มีการสื่อสารข้อมูลระหว่างกันผ่าน Industrial Ethernet และ PLC แต่ละเครื่องมีโมดูลเสริม (รูปที่ 2.5) โดยมีการสื่อสารข้อมูลระหว่างกันที่เรียกว่าการเชื่อมต่อแบบ Profibus ซึ่งการทำโครงงานฉบับนี้ต้องทราบถึงระบบการสื่อสารข้อมูลของ PLC บริษัท Siemens โดยประกอบด้วย

1. ระบบ Message Passing Interface (MPI) ทำหน้าที่สื่อสารระหว่าง PLC และคอมพิวเตอร์
2. ระบบ Process Field Bus (Profibus) ทำหน้าที่สื่อสารระหว่าง PLC ด้วยกันหรืออุปกรณ์ต่อเสริมระยะไกล
3. Actuator/Sensors Interface (ASI) ทำหน้าที่สื่อสารระหว่าง PLC กับอุปกรณ์รับสัญญาณหรือทำงานด้วยกันซึ่งเพิ่มระยะทางในการสื่อสารได้ไกลมากขึ้นอีกด้วย

รูปที่ 2.5 DP Master และ DP Slave

อุปกรณ์ต่างๆที่ติดตั้งอยู่ที่หน้างานสำหรับการควบคุมอัตโนมัติอันประกอบไปด้วย Sensors actuators transducers และ drives นั้นมีการแลกเปลี่ยนข้อมูลกับระบบควบคุมระดับสูงผ่านระบบ fieldbus PROFIBUS เป็นระบบ fieldbus ที่สามารถใช้ในอุปกรณ์อัตโนมัติเกือบทุกประเภท เช่น PLCs, PCs, Human-machine interface หรือแม้กระทั่ง sensors actuators เพื่อส่งข้อมูลโดยคำสำคัญที่ควรรู้จักมีดังนี้

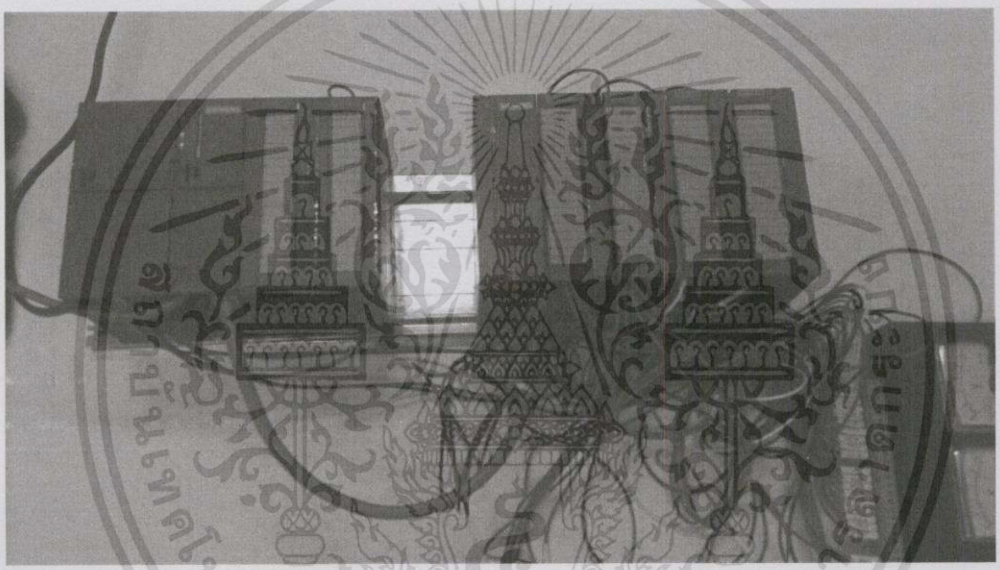
PROFIBUSDP คือ Protocol ตัวหนึ่งที่ถูกทำให้มีความเร็วการส่งข้อมูลที่สูงขึ้นแล้วจะถูกออกแบบสำหรับการสื่อสารระหว่าง PLCs (DP masters) และ Distributed I/O (DP slaves) โดยส่งผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณแบบขนานผ่านสายแรงดัน 24 V และกระแส 20 mA ซึ่งมีราคาถูกและสอดรับกับมาตรฐาน European field bus standard EN 50170

Master devices คืออุปกรณ์ที่ทำหน้าที่จัดการการส่งผ่านข้อมูลทั้งหมดในระบบซึ่งมันสามารถส่งข้อความไปยัง slave ได้โดยไม่ต้องมีคำขอเป็นตัวกำหนดการอนุญาตส่งผ่านข้อมูลของอุปกรณ์ตัวอื่นๆอาจเรียกว่าเป็น active node

Slave devices คืออุปกรณ์ simple I/O เช่น sensors actuators transducers และ drives ซึ่งไม่สามารถส่ง ข้อมูลให้กันได้หากไม่มีคำสั่งจาก master อาจเรียกว่าเป็น passive node โดยการเชื่อมต่อส่วนต่างๆในระบบ PROFIBUS



รูปที่ 2.6 สถานะต่างๆของ PLC เมื่อเปิดโหมด Run

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 OPC (Ole For Process Control)

OPC เป็นชุดการสื่อสารมาตรฐานสำหรับการเชื่อมต่อและวิธีการที่ได้นำไปใช้ในการสื่อสารกับระบบควบคุมในอุตสาหกรรมกระบวนการผลิตและอุตสาหกรรมแบบอัตโนมัติต่างๆ OPC เป็นเทคโนโลยีที่อยู่บนพื้นฐาน Windows's OLC, Com และ DCOM (Distributed Component Object Model) ซึ่งเทคโนโลยีต่างๆเหล่านี้เป็นตัวที่ใช้กำหนดให้โปรแกรมที่ทำงานบนเครื่องคอมพิวเตอร์แต่ละชนิดมีความสามารถที่ใช้ข้อมูลร่วมกันได้ ผู้ใช้งานอาจเคยได้ยินหรือเคยได้ใช้ความสามารถในการสื่อสารแบบ OLE มาบ้างแล้ว ดังเช่น การเพิ่มเติม Spreadsheet ลงไปบนเอกสาร Word ซึ่ง OLE จะยินยอมให้มีการแก้ไขหรือเพิ่มเติมข้อมูลใน Spreadsheet ได้ตลอดเวลา จากตัวอย่างนั้นเห็นได้ว่าผู้ใช้งานไม่จำเป็นต้องกำหนดรายละเอียดบนโปรแกรมเพิ่มเติมเพียงแต่ใช้เมาส์คลิกไปยังส่วนต่างๆในการทำงานเท่านั้น หลังจากนั้น OLE จะทำการกำหนดให้ Spreadsheet ทำหน้าที่เป็นโปรแกรมแม่ข่าย (OLE Server) สำหรับส่งข้อมูลไปยังเอกสาร Word ซึ่งทำหน้าที่เป็นโปรแกรมลูกข่าย (OLE Client)

OPC จะเป็นรูปแบบการสื่อสารที่ถูกพัฒนามาจาก OLE โดยการเพิ่มคุณลักษณะที่เป็นประโยชน์ต่างๆเข้าไปใน OLE เพื่อทำให้มีความน่าเชื่อถือสูงกับการนำไปใช้งานในระบบควบคุมอุตสาหกรรมกระบวนการผลิต เนื่องจากโปรแกรมแม่ข่ายของ OLE (OLE Client) และการตรวจสอบข้อมูลที่ได้รับว่าถูกต้องหรือไม่ ซึ่งจะเห็นได้ว่าความถูกต้องและความน่าเชื่อถือได้ของการรับส่งข้อมูลจะมีค่าต่ำมาก ดังนั้นจึงไม่เหมาะสมในการนำไปใช้งานกับระบบควบคุมอุตสาหกรรมกระบวนการผลิตได้ ซึ่ง OPC ได้ทำการเพิ่มเติมความสามารถในการตรวจสอบข้อมูลระหว่างการสื่อสารระหว่างโปรแกรมลูกข่าย (Client Application) และโปรแกรมแม่ข่าย (Server Application) จึงทำให้ข้อมูลที่ได้มีความถูกต้องแม่นยำและมีความเชื่อถือได้มากยิ่งขึ้น

ค่าข้อมูลจริงของ OPC จะถูกจัดเตรียมไว้สำหรับเชื่อมต่อร่วมกันระหว่างการสื่อสารกับอุปกรณ์หลายชนิดรวมไปถึงโปรแกรมคอมพิวเตอร์สำหรับใช้ในการควบคุมหรืออุปกรณ์ในกระบวนการผลิต ก่อนหน้าที่ OPC ได้ถูกพัฒนาขึ้นผู้พัฒนาโปรแกรมคอมพิวเตอร์ต้องทำการพัฒนาไดรฟ์เวอร์สำหรับการสื่อสารกับอุปกรณ์แต่ละชนิดอย่างเฉพาะเจาะจง สำหรับระบบควบคุมที่จะนำไปเชื่อมต่อกับอุปกรณ์นั้นๆ เช่น หน่วนแสดงผลหนึ่งชุดผู้จำหน่ายต้องทำการพัฒนาไดรฟ์เวอร์หลายๆแบบเพื่อใช้สำหรับระบบ PLC หรือระบบ DCS ที่แตกต่างกันไป

สำหรับการใช้งาน OPC ผู้จัดจำหน่ายหน่วยแสดงผลไม่จำเป็นต้องทำการพัฒนาไดรฟ์เวอร์หลายๆแบบเพื่อเตรียมไว้สำหรับเครื่อข่ายหรือระบบควบคุมที่แตกต่างกัน และในทางกลับกันผู้จัดจำหน่ายเหล่านี้เพียงแต่ทำการพิจารณาไดรฟ์เวอร์ OPC ที่เหมาะสมกับผลิตภัณฑ์ของตนเอง ไว้ใช้ในการเชื่อมต่อเพื่อทำการสื่อสารไปยังไดรฟ์เวอร์ OPC ของผลิตภัณฑ์จากผู้ผลิตรายอื่นๆ

ในปัจจุบันกลไกสื่อสารมาตรฐานสำหรับระบบปฏิบัติการวินโดว์อย่างเช่น OPC ได้มีการจัดเตรียมส่วนเชื่อมต่อสำหรับการใช้งาน (Application Programmer's Interface : API) ของผลิตภัณฑ์หลายๆชนิด โดยระบบนี้สามารถเข้าถึงข้อมูลได้ทางโปรแกรมการใช้งานหรืออาจจะเข้าถึงข้อมูลจากอุปกรณ์ควบคุมโดยผ่านไดรฟ์เวอร์

การใช้งาน OPC ในการเชื่อมต่อระบบควบคุมต่างๆเข้าด้วยกัน เริ่มมีจำนวนเพิ่มขึ้นสำหรับในการเชื่อมต่อระบบควบคุมอื่นๆ (Third party system) เข้ากับระบบควบคุมหลัก ข้อดีที่เห็นได้อย่างชัดเจนนั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือเป็นความสามารถในการใช้งานโปรแกรมแบบพิเศษต่างๆบนระบบควบคุมกระบวนการผลิต ข้อดีนี้ทำให้ผู้ใช้งานมีทางเลือกในการใช้งานโปรแกรมแบบพิเศษเหล่านี้ได้อย่างกว้างขวางดังตัวอย่างเช่น Advance Control, Trending, Data logging หรือ Data Conditioning เป็นต้น

ในการที่ทำให้เข้าใจการทำงานของการสื่อสารแบบ OPC จะต้องมีพื้นฐานความเข้าใจการทำงานของเครื่อข่ายแบบลูกข่าย (Client) และแม่ข่าย (Server) โดยแม่ข่ายจะเป็นส่วนที่ใช้ในการจัดเตรียมข้อมูลสำหรับลูกข่ายในการนำไปใช้งาน ลูกข่ายจะมีโปรแกรมประมวลผลจากข้อมูลที่ได้ถูกจัดเตรียมไว้ในแม่ข่าย ซึ่งแม่ข่ายจะไม่สามารถสื่อสารกันได้โดยตรงกับแม่ข่ายอื่นๆในระบบหรือบนเครื่อข่าย โปรแกรมการทำงานอยู่บนลูกข่ายจะถูกใช้เป็นสะพานในการเชื่อมต่อข้อมูลระหว่างแม่ข่ายตั้งแต่สองชุดขึ้นไป ดังแสดงในรูปที่ 2.7

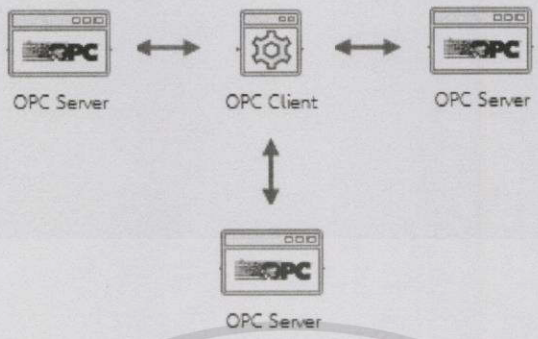
รูปที่ 2.7 การเชื่อมต่อข้อมูลระหว่างโปรแกรมการใช้งานแม่ข่าย
ที่มา : http://www.opcdatahub.com/Docs/dho-opcbridging.html (10/04/2558)

ในทำนองเดียวกันโปรแกรมการทำงานอยู่บนลูกข่ายก็ไม่สามารถสื่อสารกันได้โดยตรงกับลูกข่ายอื่นๆในระบบหรือบนเครื่อข่าย โปรแกรมการทำงานอยู่บนแม่ข่ายจะถูกใช้เป็นสะพานในการเชื่อมต่อข้อมูลระหว่างลูกข่ายตั้งแต่สองชุดขึ้นไป สามารถแสดงได้ในรูปที่ 2.8

รูปที่ 2.8 การเชื่อมต่อข้อมูลระหว่างโปรแกรมการใช้งานบนลูกข่าย

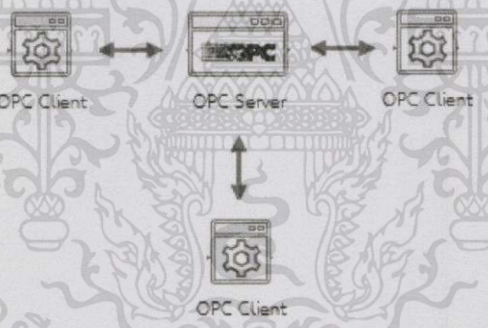
โปรแกรมการทำงานอยู่บนลูกข่ายอาจจะเข้าถึงข้อมูลที่อยู่บนแม่ข่ายได้หลายชุดในเวลาเดียวกัน ซึ่งเป็นคุณลักษณะพิเศษของโครงสร้างเครื่อข่ายแบบลูกข่ายกับแม่ข่าย เมื่อมีกานำไปใช้งานสำหรับการเชื่อมต่อข้อมูลร่วมกันโปรแกรมการใช้งานต่างๆที่ทำงานอยู่บนลูกข่ายจะทำการประมวลผลข้อมูลโดยตรงจากข้อมูลที่ได้จัดเตรียมไว้ในแต่ละแม่ข่ายและจะไม่เชื่อมต่อไปยังข้อมูลจากแม่ข่ายหลายชุดในเวลาเดียวกัน ซึ่งเป็นการสะดวกที่จะกำหนดให้แต่ละแม่ข่ายจัดเตรียมข้อมูลจากอุปกรณ์หรือระบบควบคุมที่แตกต่างกันไป สามารถแสดงได้ในรูปที่ 2.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 การอ่านข้อมูลจากแม่ข่ายหลายชุด

ข้อดีอีกข้อหนึ่งของโครงสร้างเครื่อข่ายแบบลูกข่ายกับแม่ข่ายสามารถกำหนดให้ลูกข่ายหลายชุดเชื่อมต่อเข้ากับแม่ข่ายได้ในเวลาที่ต้องการ ความสามารถนี้จะมีประสิทธิภาพอย่างมากในการใช้ข้อมูลร่วมกันของระบบควบคุม ดังแสดงได้ในรูปที่ 2.10



รูปที่ 2.10 การประมวลผลข้อมูลจากแม่ข่ายเดียวกัน

โดยทั่วไปในการควบคุมอุตสาหกรรมกระบวนการผลิตต่างๆ จะแบ่งระบบควบคุมออกเป็น 2 ชนิดคือระบบควบคุมพื้นฐานหรือระบบ DCS และระบบวัดคุมนิรภัย (Safety Instrumented System: SIS) หรือระบบ ESD (Emergency Shut Down system) ซึ่งทั้งสองระบบจะทำงานที่แยกเป็นอิสระต่อกัน ในการใช้งานจริงแล้วระบบวัดคุมนิรภัยจะต้องมีการรับส่งข้อมูลระหว่างระบบควบคุมพื้นฐานตลอดเวลาเนื่องจากระบบวัดคุมนิรภัยจะไม่มีหน่วยแสดงผลเพราะไม่ได้มีวัตถุประสงค์ในการควบคุมกระบวนการผลิต แต่ถึงอย่างไรก็ตามผู้ปฏิบัติการก็ยังต้องมีการตรวจสอบค่าตัวแปรต่างๆจากกระบวนการผลิตในขณะที่เกิดเหตุการณ์ผิดปกติขึ้น ดังนั้นในการออกแบบระบบควบคุมกระบวนการผลิตจึงกำหนดให้ระบบวัดคุมนิรภัยต้องส่งข้อมูลต่างๆที่ต้องการไปแสดงผลยังหน่วยแสดงผลของระบบ DCS โดยผ่าน

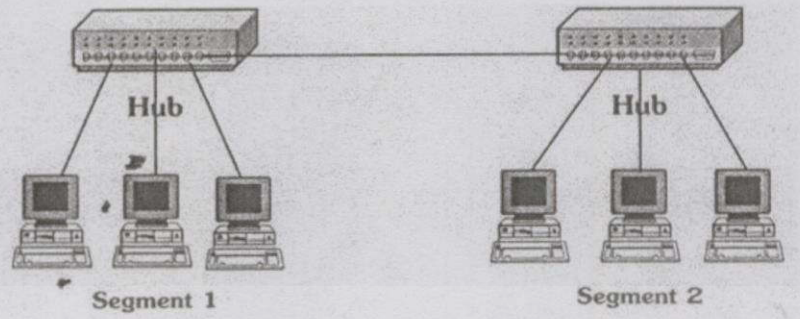
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครือข่ายหลักของระบบควบคุมผ่านการเชื่อมต่อทางโปรแกรม ซึ่ง OPC ก็เป็นทางเลือกหนึ่งที่นิยมใช้กันอย่างแพร่หลายในปัจจุบัน แต่ถ้าเป็นข้อมูลที่เป็นคำสั่งต้องทำการเชื่อมต่อกันด้วยสายไฟเท่านั้นเพื่อความปลอดภัยในการควบคุมกระบวนการผลิต

จากที่กล่าวมาแล้วข้างต้นการสั่งการใดๆจากระบบควบคุมหนึ่งไปยังอีกระบบควบคุมหนึ่งสามารถทำได้โดยผ่านการเชื่อมต่อ OPC ซึ่งในการออกแบบระบบควบคุมสำหรับอุตสาหกรรมกระบวนการผลิตแล้วจะหลีกเลี่ยงการใช้งานรูปแบบนี้ เนื่องจากคำสั่งที่ใช้ในการควบคุมอุปกรณ์ต่างๆในอุตสาหกรรมกระบวนการผลิตต้องการเวลาในการรับส่งข้อมูลที่แน่นอนและมีความเชื่อมั่นสูง และความเร็วของการรับส่งข้อมูลผ่านเครือข่ายการสื่อสารนั้นจะขึ้นอยู่กับปริมาณข้อมูลที่สื่อสารอยู่บนเครือข่าย ดังนั้นถ้าคำสั่งจากต้นทางไปยังอุปกรณ์ปลายทางล่าช้ากว่าเวลาที่ต้องการก็อาจจะทำให้เกิดเหตุการณ์อันตรายขึ้นได้

ในการเชื่อมต่อด้วยมาตรฐาน OPC นั้นเราใช้อุปกรณ์ตัวกลางในการสื่อสารคือ ฮับ (Hub) (รูปที่ 2.11) โดยฮับหรือรีพีตเตอร์ (Repeater) คืออุปกรณ์ที่ใช้เชื่อมต่อกลุ่มของคอมพิวเตอร์ฮับ มีหน้าที่รับส่งเฟรมข้อมูลทุกเฟรมที่ได้รับจากพอร์ตใดพอร์ตหนึ่งไปยังๆทุกๆพอร์ตที่เหลือ คอมพิวเตอร์ที่เชื่อมต่อเข้าฮับจะแชร์แบนด์วิธหรืออัตราข้อมูลของเครือข่าย ฉะนั้นยิ่งมีคอมพิวเตอร์เชื่อมต่อเข้าฮับมากเท่าใด ยิ่งทำให้แบนด์วิธต่อคอมพิวเตอร์แต่ละเครื่องลดลง ในท้องตลาดปัจจุบันมีฮับหลายชนิดจากหลายบริษัท ข้อแตกต่างระหว่างฮับเหล่านี้ก็เป็นจำพวกพอร์ต สายสัญญาณที่ใช้ ประเภทของเครือข่าย และอัตราข้อมูลที่ฮับรองรับได้การที่อุปกรณ์เครือข่ายอีเธอร์เน็ตสามารถทำงานได้ที่ความเร็ว 2 ระดับ เช่น 10/100 Mbps นั้น ก็เนื่องจากอุปกรณ์เครื่องนั้นมีฟังก์ชันที่สามารถเช็คได้ว่าอุปกรณ์ หรือคอมพิวเตอร์ที่มาเชื่อมต่อกับฮับนั้นสามารถรับส่งข้อมูลได้ที่ความเร็วสูงสุดเท่าใด และอุปกรณ์นั้นก็จะเลือกอัตราข้อมูลสูงสุดที่รองรับทั้งสองฝั่ง ฟังก์ชันนี้จะเรียกว่า "การเจรจาอัตโนมัติ (Auto-Negotiation)" ส่วนใหญ่ฮับหรือสวิสต์ที่ผลิตจะมีฟังก์ชันนี้อยู่ เพื่อให้สามารถเชื่อมต่อเครือข่ายอีเธอร์เน็ตที่ความเร็วต่างกันได้ ถ้ามีอุปกรณ์เครือข่าย หรือคอมพิวเตอร์หลาย ๆ เครื่องเชื่อมต่อเข้ากับฮับและแต่ละโหนดสามารถส่งข้อมูลได้ในอัตราที่ต่างกัน ฮับก็จะเลือกอัตราส่งข้อมูลที่อัตราความเร็วต่ำสุด เนื่องจากคอมพิวเตอร์เหล่านี้จัดอยู่ในคอลลิชั่นโดเมน (Collision Domain) เดียวกัน ตัวอย่างเช่น ถ้า LAN การ์ดของคอมพิวเตอร์เครื่องหนึ่งสามารถรับส่งข้อมูลได้ที่ 10 Mbps ส่วน LAN การ์ดของคอมพิวเตอร์ที่เหลือสามารถรับส่งข้อมูลได้ 10/100 Mbps แล้วคอมพิวเตอร์เหล่านี้เชื่อมต่อเข้ากับฮับเดียวกันที่รองรับอัตราความเร็วที่ 10/100 Mbps เครือข่ายนี้ก็จะทำงานที่ความเร็ว 10 Mbps เท่านั้น แต่ถ้าเป็นสวิทช์อัตราความเร็วจะขึ้นอยู่กับความเร็วของคอมพิวเตอร์เนื่องจากสวิทช์จะแยกคอลลิชั่นโดเมน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 การเชื่อมต่อคอมพิวเตอร์เข้ากับเครือข่ายโดยใช้ฮับ
ที่มา : http://www.nkatc.svec.go.th/web/pictures/e-learning/E-Learning3/unit3.html (11/04/2558)

2.4 ภาษา Visual Basic

ในอดีตการเขียนโปรแกรมหรือการพัฒนาแอพพลิเคชันขึ้นมาสักตัว โปรแกรมเมอร์จะต้องมีความรู้ ความเข้าใจในตัวภาษา ที่จะนำมาใช้ในการพัฒนา เป็นอย่างมาก ไม่ว่าจะพัฒนาด้วยภาษาใดก็ตาม เช่น ภาษา C, C++, ปาสคาล เป็นต้น และจะต้องเขียนโค้ดที่มีความสัมพันธ์กันตั้งแต่ บรรทัดแรกจนถึงบรรทัดสุดท้าย หรือที่เรียกว่า การเขียน โค้ดแบบ command line อีกทั้งยังต้องออกแบบรูปร่างหน้าตาของแอพพลิเคชัน ให้ตรงกับความต้องการของผู้ใช้อีกด้วย ซึ่งถ้าถูกใจผู้ใช้ก็ดีไป แต่ถ้าต้องมีการแก้ไขแล้วละก็ ทุกสิ่งทุกอย่างที่คุณได้ทำมา ก็ต้องแทบรื้อทำใหม่หมด ทำให้โปรแกรมเมอร์เสียเวลา ในการพัฒนาแอพพลิเคชันเป็นอย่างยิ่ง

ต่อมาไมโครซอฟท์ได้นำเสนอรูปแบบในการเขียนแอพพลิเคชันชนิดใหม่ด้วยการออก Visual Basic 1.0 แม้ว่า เวอร์ชันแรกนี้ จะถูกโปรแกรมเมอร์ในยุคนั้น มองว่าเป็นเวอร์ชันทดลอง แต่มันก็ได้สร้างความแปลกใหม่ ในการเขียนโปรแกรมเป็นอย่างยิ่ง ความยุ่งยากซับซ้อน ถูกซ่อนไว้เบื้องหลัง มีแต่ความสะดวกสบายไว้เบื้องหน้า ที่เตรียมไว้ให้โปรแกรมเมอร์ เนื่องจาก concept ในการเขียน โปรแกรมแทบจะเปลี่ยนไปอย่างสิ้นเชิง จนกระทั่งในยุคปัจจุบัน Visual Basic ได้พัฒนามาถึงเวอร์ชัน 6.0 แล้ว ความสามารถของตัวภาษา VB เองก็มีมากขึ้น เพราะความก้าวหน้าของเทคโนโลยีในยุคปัจจุบัน ทำให้ไมโครซอฟท์เพิ่มเติมฟีเจอร์ต่างๆ เข้าไปมากมาย จนกระทั่ง VB แทบจะเป็นเครื่องมือ ในการพัฒนาแอพพลิเคชันที่ไร้เทียมทาน เพราะเทคโนโลยีใหม่ๆ ล้วนแล้วแต่มาจาก ไมโครซอฟท์แทบทั้งสิ้น ดังนั้นจึงไม่เป็นที่แปลกใจเลยว่า ไมโครซอฟท์ได้เพิ่มความสามารถในทุกๆ เวอร์ชันของ VB เสมอ เช่น สามารถสร้างแอพพลิเคชันชนิด DHTML ซึ่งใช้ run บน web ได้, รวมถึงการผนวกเทคโนโลยี ActiveX เข้ากับตัวคอนโทรลของ VB ทำให้สามารถเชื่อมโยงเข้ากับเครื่องมือ ที่สนับสนุนเทคโนโลยีนี้ได้อีกด้วย แต่ VB ยังคงรักษาเอกลักษณ์อย่างหนึ่ง ไว้ได้เป็นอย่างดีนั่นคือ สามารถที่จะพัฒนาแอพพลิเคชันได้ ในระยะเวลาอันสั้น รวมถึงความง่าย ต่อการเรียนรู้ในตัวภาษา และทำความเข้าใจได้อย่างรวดเร็ว เมื่อเทียบกับการเริ่มต้นเรียนรู้ด้วยภาษาอื่นๆ คุณสามารถที่จะสร้างแอพพลิเคชันออกมาได้อย่างรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VB ได้จัดเตรียมเครื่องมือต่างๆ ที่เรียกว่า คอนโทรล (controls) ไว้คอยอำนวยความสะดวกให้แก่โปรแกรมเมอร์มากมาย คุณจะต้องศึกษา และทำความเข้าใจกับตัวคอนโทรลให้มากที่สุด ซึ่งตัวคอนโทรลเหล่านี้นี่เอง ที่อยู่เบื้องหลังทำให้ VB ประสบความสำเร็จเป็นอย่างยิ่ง ในยุคปัจจุบัน เพราะเนื่องจากมันได้ลดขั้นตอนต่างๆในการพัฒนาไปได้มากทีเดียว

แนวทางการเขียนโปรแกรมด้วยภาษา VB จะเป็นไปในลักษณะการนำคอนโทรลชนิดต่างๆ เช่น TextBox, Label, ComboBox เป็นต้น นำมาวาด เพื่อออกแบบหน้าตาแอพพลิเคชันที่เรียกว่า กราฟฟิก ยูสเซอร์ อินเตอร์เฟส (Graphic User Interface-GUI) คุณสามารถที่จะออกแบบ อินเตอร์เฟสได้อย่างอิสระ ให้ตรงกับจุดประสงค์และ การนำไปใช้งานของคุณก่อน แล้วจึงเริ่มเขียนโค้ด เพื่อตอบสนองการกระทำของผู้ใช้ (ใน VB เรียกว่า เหตุการณ์ event) ซึ่งถือเป็นหลักการเขียนโปรแกรมที่เรียกว่า การเขียนโปรแกรมเพื่อตอบสนองเหตุการณ์ที่เกิดขึ้น(Event-DrivenProgramming)

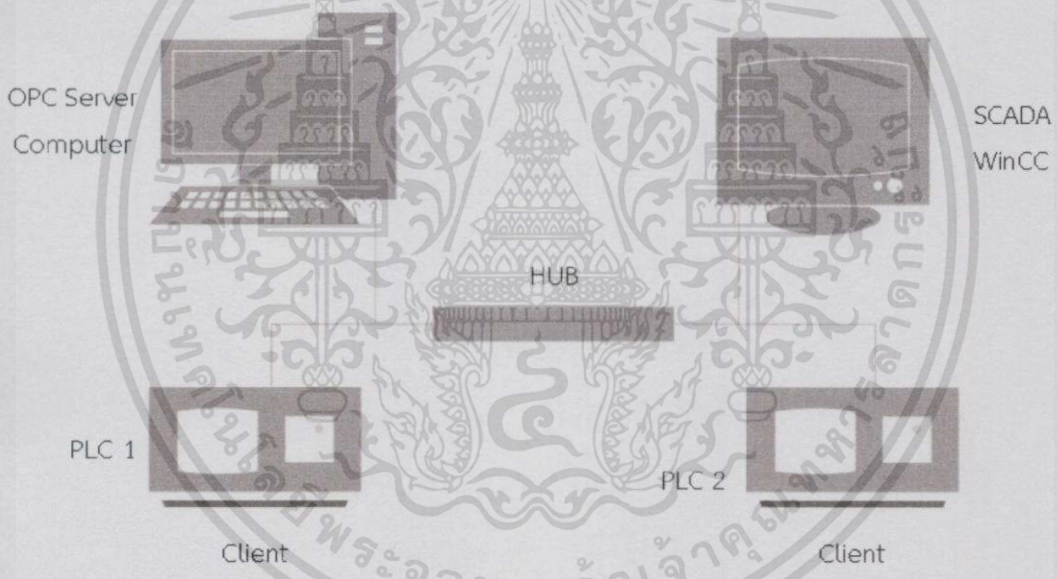
สิ่งต่างๆ ที่คุณนำไปใช้ร่วมกันเพื่อเป็นแอพพลิเคชันหนึ่งๆ เช่น แถบเมนู, dialog box , toolbars, TextBox, ปุ่ม OK ฯลฯ เป็นต้น จะถูกมองเปรียบเสมือนว่า เป็นวัตถุชิ้นหนึ่งที่เรียกว่า อ๊อบเจ็กต์ (object model) ทุกสิ่งทุกอย่าง ในแอพพลิเคชัน VB จะมองเป็นอ๊อบเจ็กต์ ที่คุณสามารถควบคุมพฤติกรรม, แก้ไข และกระทำโดยตรงต่ออ๊อบเจ็กต์นั้นได้ ด้วยการเขียนโค้ด หรือสามารถเปลี่ยนแปลง คุณสมบัติ (properties) ประจำตัวของอ๊อบเจ็กต์นั้นได้โดยตรง ตัวคอนโทรลก็ถูกมองเป็นอ๊อบเจ๊กต์ เช่นกัน ในทุกๆ อ๊อบเจ็กต์จะมีคุณสมบัติ (properties) และ เมธอด (methods) ประจำตัว ในแต่ละอ๊อบเจ็กต์ อาจจะมีคุณสมบัติและเมธอดที่เหมือน หรือต่างกันก็ได้ ขึ้นอยู่กับชนิดของอ๊อบเจ็กต์ คุณจะพัฒนาแอพพลิเคชัน ได้ดีหรือไม่ ส่วนหนึ่งจะมาจากการที่คุณสามารถ ใช้งานคอนโทรล, แก้ไขคุณสมบัติและเมธอดได้ตรงตามความต้องการของคุณและเต็มประสิทธิภาพของคอนโทรลนั้นๆได้หรือไม่

ในการพัฒนาแอพพลิเคชันด้วย VB การเขียนโค้ดจะถูกแบ่งออกเป็นส่วนๆ ที่เรียกว่า โพรซีเดอร์ (precedure) แต่ละโพรซีเดอร์จะประกอบไปด้วย โค้ดที่คุณพิมพ์เข้าไปแล้วทำให้คอนโทรลหรืออ๊อบเจ็กต์นั้นๆ ตอบสนองการกระทำ ของผู้ใช้ได้โดยสมบูรณ์ในตัวมันเอง ซึ่งเรียกว่าการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programmming-OOP) แต่ตัวภาษา VB ยังไม่ถือว่าเป็น OOP อย่างแท้จริง เนื่องจากข้อจำกัดหลายๆ อย่างที่ Visual Basic ไม่สามารถทำได้เหมือนกับภาษา C++ การเขียนโปรแกรมเชิงวัตถุมีข้อดีก็คือ ตัวโค้ดจะถูกแบ่งออกเป็นส่วนๆ ทำให้ง่ายต่อการตรวจสอบ และดักจับข้อผิดพลาด (debug) ซึ่งการแก้ไขดังกล่าวนี้ ไม่ได้ไปกระทบกับโค้ด ส่วนอื่นๆ ในตัวแอพพลิเคชันนั้นเลย ทำให้โปรแกรมเมอร์สามารถพัฒนาแอพพลิเคชันออกมา ได้อย่างสมบูรณ์แบบมากที่สุด โดยที่ไม่ต้องเสีย เวลามากมายดังเช่นในอดีต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ขั้นตอนการดำเนินงาน

ในการเริ่มต้นการออกแบบแบบจำลองระบบ SCADA ในห้องทดลองนั้นต้องกำหนดการทำงานทั้งหมดและส่วนประกอบย่อย ซึ่งส่วนประกอบย่อยได้แบ่งเป็นสองส่วนหลักๆคือ ส่วนของแบบโครงสร้างการเชื่อมต่อของฮาร์ดแวร์และอีกส่วนคือ การคอนฟิกูเรชั่น โปรแกรมและกราฟฟิค ซึ่งฮาร์ดแวร์ที่นำมาใช้นั้นประกอบไปด้วย PLC สองเครื่องที่ทำหน้าที่เป็นลูกข่ายและมีคอมพิวเตอร์หนึ่งเครื่องทำหน้าที่เป็นแม่ข่าย เชื่อมต่อกันด้วยการสื่อสารแบบอีเทอร์เน็ต โดยมีอุปกรณ์ตัวกลางในการเชื่อมต่อคือฮับ ซึ่งการออกแบบทั้งหมดของระบบ SCADA จะใช้งาน OPC เป็นหน่วยเชื่อมต่อสื่อสารของระบบงานวิจัยนี้ รูปแบบการเชื่อมต่อของแบบจำลองโดยภาพรวมสามารถแสดงเป็นแผนผังการเชื่อมต่อได้ดังรูปที่ 3.1



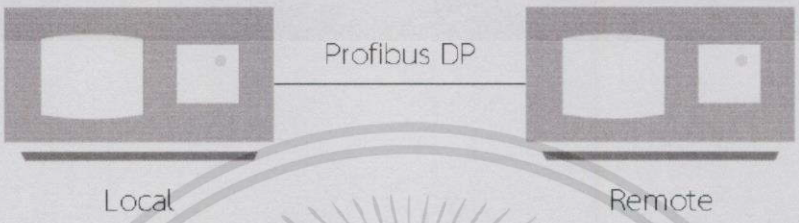
รูปที่ 3.1 แผนผังการเชื่อมต่อของอุปกรณ์ฮาร์ดแวร์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 การเชื่อมต่ออุปกรณ์ทางด้านฮาร์ดแวร์

การเชื่อมต่อของอุปกรณ์ต่างๆ สามารถอธิบายได้ดังนี้

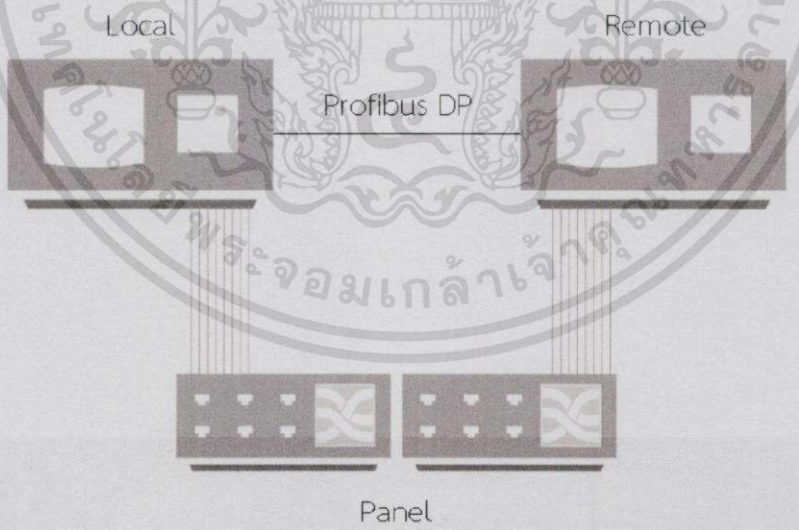
3.1.1 การเชื่อมต่อระหว่าง Module Local กับ Remote ของ PLC



รูปที่ 3.2 แผนภาพการเชื่อมต่อระหว่างอุปกรณ์ Local และ Remote ของพีแอลซี

ในการติดต่อกันระหว่างอุปกรณ์ Local และ Remote นั้นจะทำการติดต่อกันผ่านโปรฟิบัสดีพี ซึ่งเป็นโปรโตคอลใช้สื่อสารระหว่างส่วนควบคุมส่วนกลางกับอุปกรณ์อิทพุท เอาท์พุทที่ระดับฟิลด์ ซึ่งโปรฟิบัสดีพีเป็นโปรฟิบัสที่ใช้ในอุปกรณ์เชื่อมต่อส่วนนี้

3.1.2 การเชื่อมต่อระหว่าง PLC กับแผงควบคุม

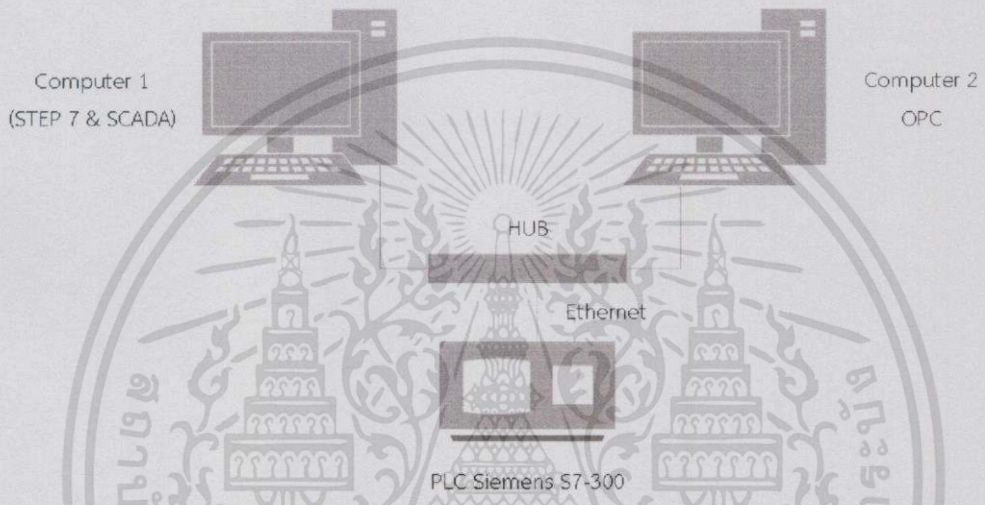


รูปที่ 3.3 แผนภาพแสดงการเชื่อมต่อระหว่าง PLC กับแผงควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

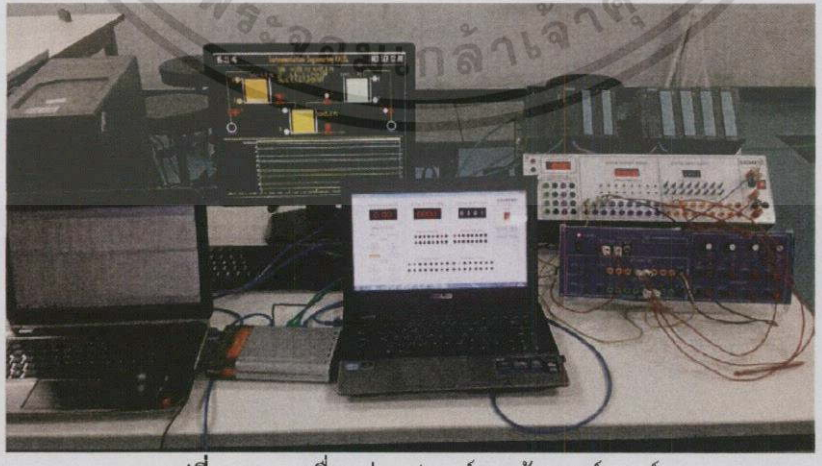
ในการติดต่อกันระหว่าง PLC กับหน้าแผงควบคุมนั้นจะเชื่อมต่อกันด้วยสายไฟหรือสายสัญญาณจากโมดูลอินพุทหรือเอาท์พุทเข้าที่ด้านหลังของแผงควบคุม โดยที่หน้าแผงควบคุมจะแบ่งออกเป็นสองส่วนคือแผงของเอาท์พุทและอินพุทของ Local และ Remote ซึ่งการเชื่อมต่อนั้นจะแยกกันตามโครงสร้าง

3.1.3 การเชื่อมต่อระหว่างคอมพิวเตอร์กับเครื่องพีแอลซี



รูปที่ 3.4 แผนภาพการเชื่อมระหว่างพีแอลซีกับคอมพิวเตอร์สองเครื่อง

ในการติดต่อระหว่างเครื่องพีแอลซีกับคอมพิวเตอร์ทั้งสองเครื่องนั้นจะทำการติดต่อผ่านอีเทอร์เน็ตที่เป็นโปรโตคอลของแลนโดยมีฮับเป็นอุปกรณ์เชื่อมต่อตัวกลางและเมื่อทำการเชื่อมต่ออุปกรณ์ทั้งหมดแล้วจะสามารถแสดงให้เห็นได้ดังรูปที่ 3.5



รูปที่ 3.5 การเชื่อมต่ออุปกรณ์ทางด้านฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การเชื่อมต่ออุปกรณ์ทางด้านซอฟต์แวร์

การเชื่อมต่ออุปกรณ์ซอฟต์แวร์แบ่งออกเป็น 3 ขั้นตอนคือ การตั้งค่าเครื่องพีแอลซีกับคอมพิวเตอร์ การตั้งค่าโปรแกรมสำหรับ SCADA และการตั้งค่าโปรแกรมที่ใช้สำหรับการเชื่อมต่อ OPC ซึ่งรายละเอียดตามดังต่อไปนี้

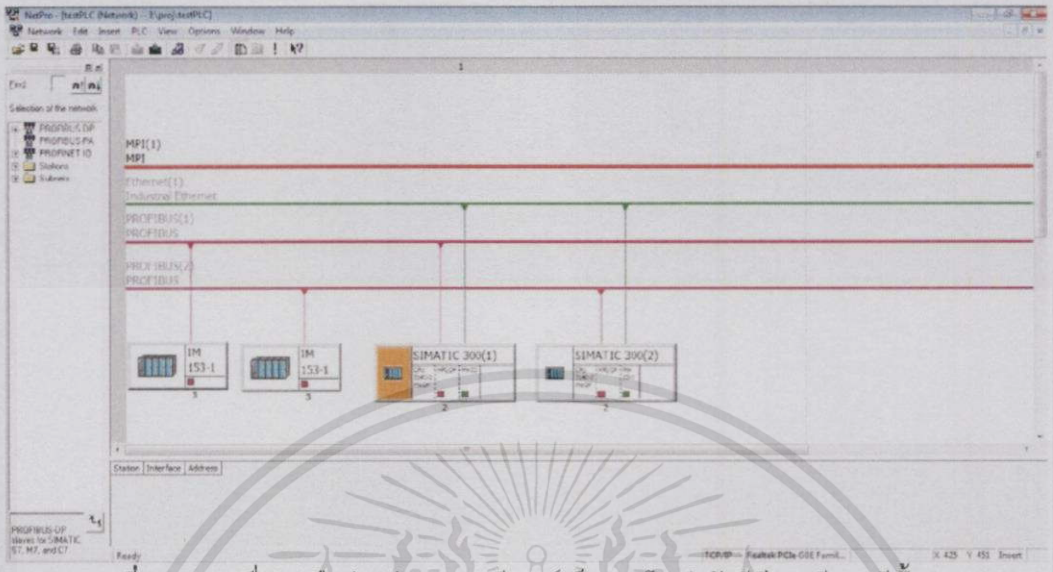
รูปที่ 3.6 แผนภาพขั้นตอนการเชื่อมต่ออุปกรณ์ทางด้านซอฟต์แวร์

3.2.1 การตั้งค่าเครื่องพีแอลซีเชื่อมต่อกับคอมพิวเตอร์

รูปที่ 3.7 โปรแกรมที่ใช้ในการกำหนดโครงสร้างของพีแอลซี

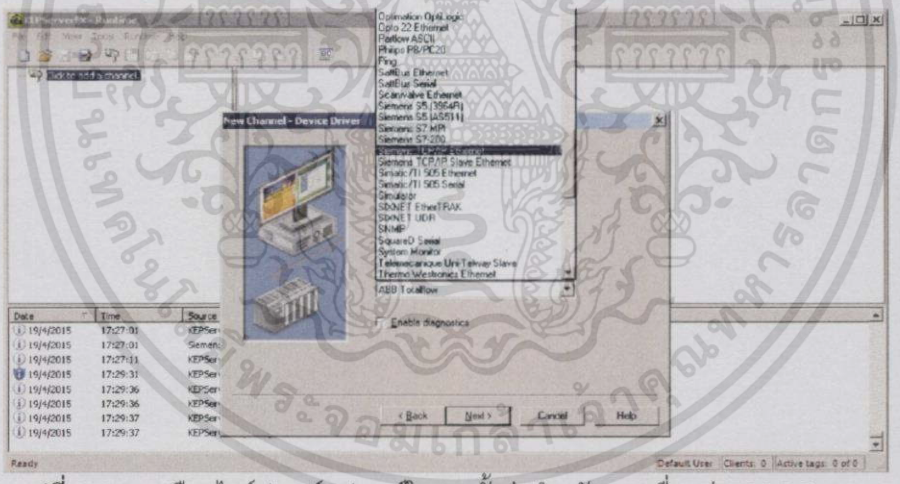
ในการตั้งค่าเครื่องพีแอลซีเชื่อมต่อกับคอมพิวเตอร์จะมีโปรแกรมที่ใช้คือ โปรแกรม Simatic Step 7 โดยเมื่อก่อนเริ่มใช้งานต้องมีการกำหนดโครงสร้างของฮาร์ดแวร์เสียก่อน ซึ่งโครงสร้างฮาร์ดแวร์ทั้งหมดได้กล่าวไว้ในบทที่ 2 ในเรื่องของพีแอลซีที่ใช้ในการวิจัย เมื่อกำหนดโครงสร้างเสร็จแล้วก่อนการดาวน์โหลดโปรแกรมลงเครื่องพีแอลซี ให้เสียบสาย PC Adapter เข้าที่พีแอลซีและคอมพิวเตอร์ และตรวจสอบพอร์ทสำหรับการสื่อสารในหน้าต่าง Set PG/PC Interface และทำการกำหนดไอพีแอดเดรสของพีแอลซีทั้งสองเครื่องเป็น IP : 192.168.0.100 และ IP : 192.168.0.101 ตามลำดับ จากนั้นบิดสวิตช์ที่พีแอลซีทั้งสองเครื่องไปที่ตำแหน่ง Run-P แล้วทำการดาวน์โหลดฮาร์ดแวร์ ซึ่งวิธีการตั้งค่าทั้งหมดนี้ได้กล่าวไว้ในภาคผนวกต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 การเชื่อมเครือข่ายผ่านระบบอีเทอร์เน็ตและโปรฟิบัสดีพีของพีแอลซีทั้งสอง

3.2.2 การตั้งค่าโปรแกรมสำหรับการเชื่อมต่อ OPC Server



รูปที่ 3.9 การเลือกไดร์ฟเวอร์อุปกรณ์ในการตั้งค่าสำหรับการเชื่อมต่อ OPC Server

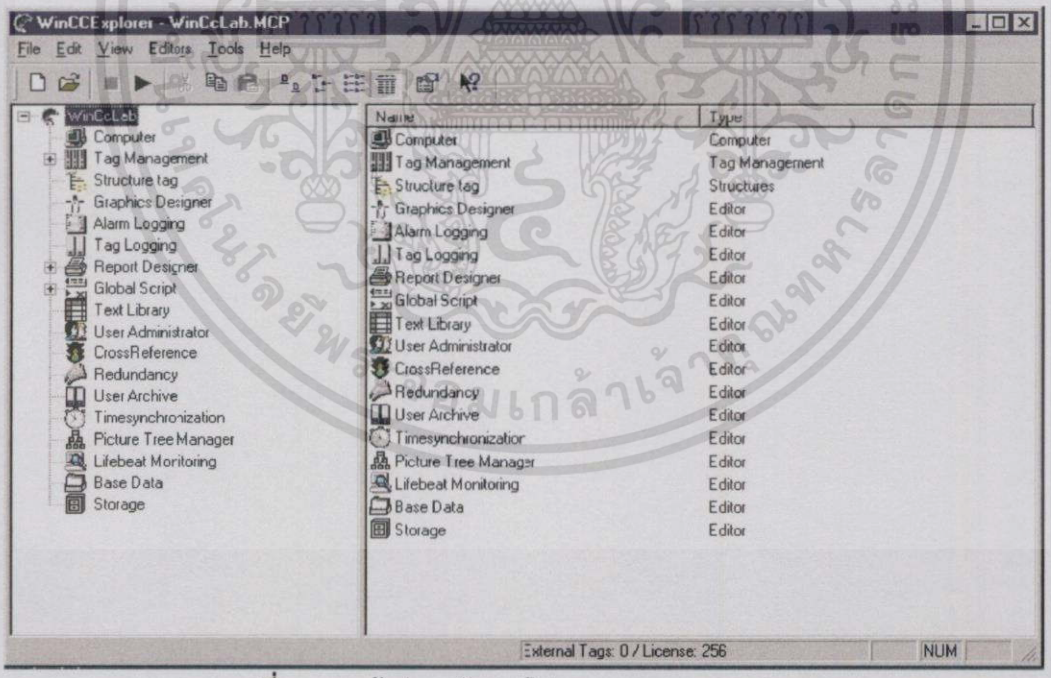
ในงานวิจัยนี้เราเลือกใช้โปรแกรม KEPServerEx V5 มาเป็น OPC Server ซึ่งมีประโยชน์ในการแลกเปลี่ยนข้อมูลกันระหว่าง SCADA พีแอลซี คอมพิวเตอร์ หรืออุปกรณ์อื่นๆ โดยเมื่อเปลี่ยนค่า OPC Tag ที่โหนดใดโหนดหนึ่ง โหนดอื่นๆของ SCADA ที่ติดต่อมาที่ OPC Server นี้ก็จะมีค่าที่ตรงกันกับที่เราเปลี่ยนแปลงนั้น โดยก่อนเริ่มใช้งานต้องมีการกำหนดโครงสร้างของฮาร์ดแวร์ที่เชื่อมต่อและสร้าง Tag ในแอดเดรสที่เราต้องการมอนิเตอร์ค่า ซึ่งสามารถสร้าง Tag ได้มากตามที่ต้องการ แต่มีข้อระวังคือ อย่าให้แอดเดรสทับกัน เช่นถ้าเราสร้าง OPC Tag ที่แอดเดรส I00.00 เป็น Data type ประเภท Double ซึ่งใช้ 4 word ถ้าเราสร้าง OPC Tag ใหม่ที่ใช้แอดเดรส I อีกต้องเว้นไป 4 word ให้กับ I ก่อนหน้าก็จะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I ถัดไปเป็น I00.04 ทั้งหมดนี้สามารถกำหนดให้เป็น Word, DWord หรือ Double ก็ได้ตามที่ต้องการ โดยการกำหนดฮาร์ดแวร์ทั้งหมดนี้ได้กล่าวไว้ในภาคผนวก

3.2.3 การตั้งค่าโปรแกรมสำหรับการเชื่อมต่อ SCADA

ในการเขียนกราฟิก SCADA สำหรับพีแอลซี Siemens ในงานวิจัยนี้เลือกใช้โปรแกรม WinCC (Window Control Center) เวอร์ชั่น 7.0 ซึ่งเป็นซอฟท์แวร์ที่ใช้ในการควบคุมงานทางด้านการผลิตและกระบวนการอัตโนมัติ ผ่านหน้าจอคอมพิวเตอร์ ภายในซอฟท์แวร์ประกอบโมดูลที่ใช้จัดการทางด้านกราฟิก แมสเซสไฟล์ข้อมูล และการทำรายงาน ซึ่งเหมาะสมสำหรับงานทางด้านอุตสาหกรรมการผลิตต่างๆทั่วไป ในขั้นตอนการทดลองเริ่มจากการกำหนดไดร์ฟเวอร์อุปกรณ์ โดยขั้นตอนการกำหนดโครงสร้างต่างๆได้กล่าวไว้ในภาคผนวก ต่อมาคือเขียนกราฟิกและจัดการ Tag ต่างๆ โดย Tag ใน WinCC เป็นได้ทั้งตัวแปรค่าที่ใช้งานจริง เช่น ค่าระดับน้ำแทงค์ หรือเป็นได้ทั้งตัวแปรค่าที่ใช้เฉพาะภายในโปรแกรมที่ใช้ในการคำนวณหรือการจำลองการทำงาน ซึ่งจะแบ่ง Tag ออกเป็นสองประเภทคือ Tag ที่ตัวแปรค่าใช้งานจริงค่าของมันถูกส่งมาจากหน่วยความจำของพีแอลซีหรืออุปกรณ์อื่นๆที่ติดต่อกับโปรแกรมผ่านสายสัญญาณหรือระบบสื่อสารต่างๆนั้นคือ Process Tags และอีกแบบคือ Internal Tags เป็นตัวแปรค่าที่ใช้เฉพาะภายในตัวโปรแกรมใช้ในการคำนวณหรือการจำลองการทำงาน ซึ่งสามารถจัดกลุ่มของ Tags ได้โดยสร้าง Tags Groups



รูปที่ 3.10 หน้าต่างหลักของโปรแกรม WinCC Explorer

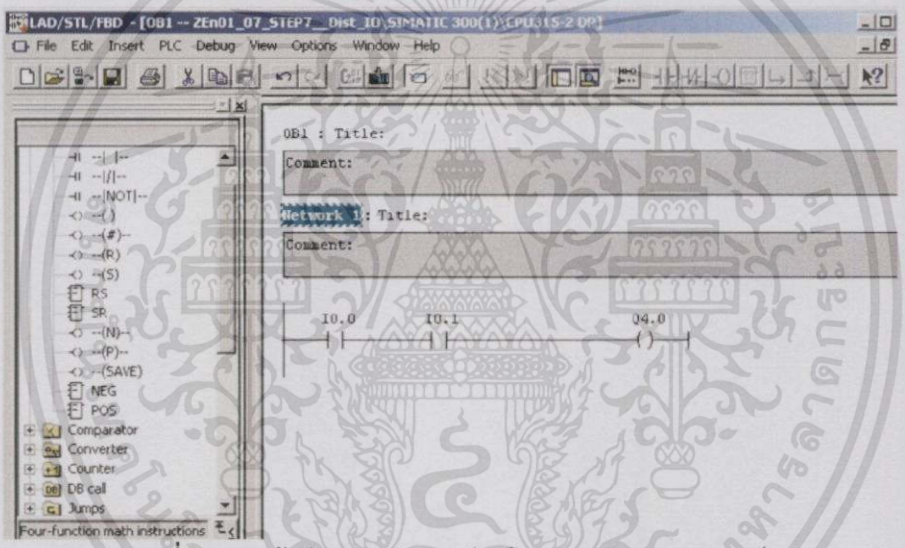
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การเขียนโปรแกรมและกราฟิกการทำงานของระบบ

ในงานวิจัยนี้จะมีการเขียนโปรแกรมออกเป็น 2 อย่างคือการเขียนโปรแกรมควบคุมพีแอลซีและการเขียนกราฟิกสำหรับการแสดงผล

3.3.1 การเขียนโปรแกรมสำหรับควบคุมพีแอลซี

ส่วนการเขียนโปรแกรมการทำงานให้กับเครื่องควบคุมพีแอลซี ซึ่งใช้ซอฟต์แวร์คือโปรแกรม Simatic Step 7 Manager ภาษาที่ใช้ในการเขียนโปรแกรมการทำงานเขียนโดยใช้ได้หลายภาษาเช่น ภาษาแลดเดอร์, ภาษาบูลลีน, ภาษาบล็อค, ภาษาข้อความภาษาอังกฤษ, ภาษาฟังก์ชั่นชาร์ทเป็นต้น โดยในงานวิจัยนี้เราใช้ภาษาแลดเดอร์ในการเขียนโปรแกรมต่างๆเพื่อใช้ทดลองกราฟิกที่ได้เขียนมารวมไปถึงค่าที่ได้ใน OPC Server ด้วย

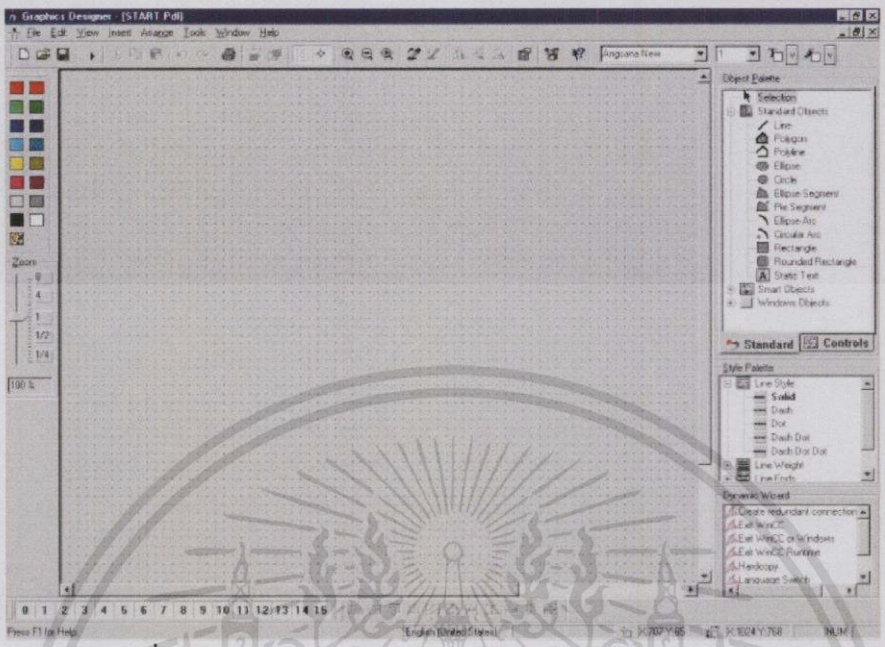


รูปที่ 3.11 หน้าต่างสำหรับการเขียนโปรแกรมภาษาแลดเดอร์

3.3.2 การเขียนกราฟฟิกสำหรับการแสดงผลและเทรนไลน์

การเขียนกราฟิกนั้นจะเริ่มจากการใช้งานหน้าต่าง Graphics Designer ซึ่งเป็นส่วนประกอบของโปรแกรม WinCC ที่ใช้สำหรับออกแบบหน้าจอที่เป็นตัวกลางระหว่างผู้ใช้กับเครื่องจักร โดย Graphics Designer มีความสามารถในการสร้างรูปภาพหรือวัตถุที่มีความสามารถในการเข้าถึงข้อมูลจากภายนอก มีความสามารถในการเขียนโปรแกรมเพื่อจัดการกับข้อมูลที่ได้รับมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

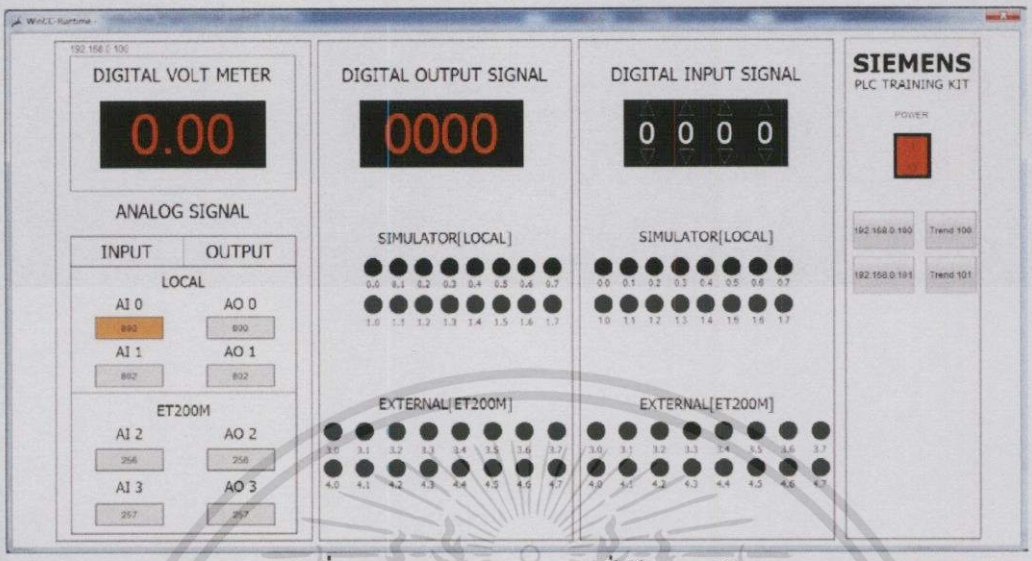


รูปที่ 3.12 หน้าต่าง Graphics Designer ของโปรแกรม WinCC

นอกจากนี้ Graphics Designer ยังได้เตรียมรูปภาพต่างๆไว้ โดยที่ผู้ใช้ไม่จำเป็นต้องสร้างมาขึ้นใหม่ สามารถนำมาใช้ในการเขียนกราฟิกได้เลย เริ่มจากการสร้างรูปภาพใหม่แล้วกำหนดคุณสมบัติของภาพที่สร้างขึ้นมา ซึ่งการกำหนดคุณสมบัติของภาพนั้น กำหนดได้หลากหลายมากมาย อาทิเช่น กำหนดให้เป็นปุ่มกด แสดงผลเป็นสีต่างๆคล้ายไฟบอกสถานะ หรือเป็นถังน้ำที่สามารถแสดงสถานะของระดับน้ำได้เป็นต้น คุณสมบัติต่างๆที่เป็นการแสดงผลเหล่านี้สามารถกำหนดได้ในหน้าต่าง Tag Assignment โดยในงานวิจัยนี้เราได้ทำการเขียนกราฟิกจำลองหน้าแผงควบคุมซึ่งแสดงได้ดังภาพ

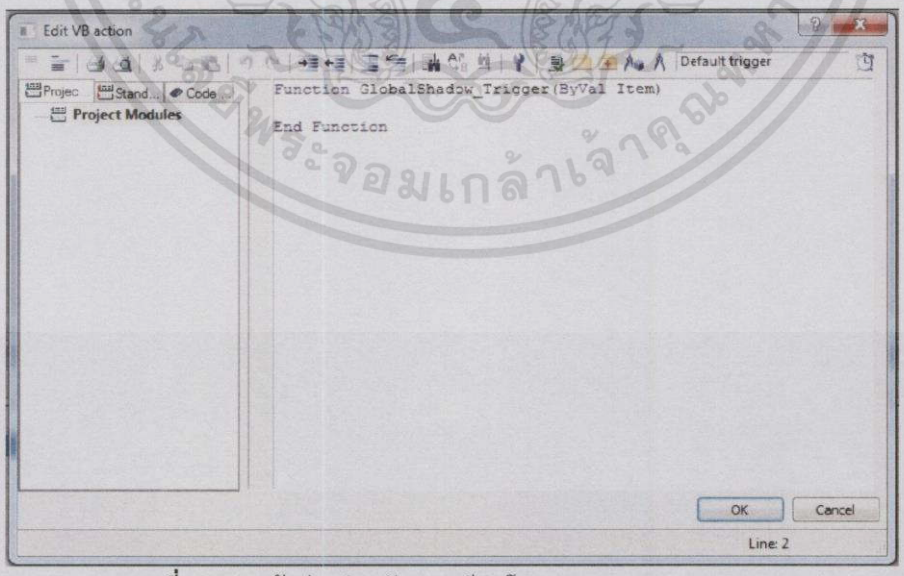
รูปที่ 3.13 แผงควบคุมที่ใช้ในงานวิจัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 กราฟิกแผงควบคุมที่ใช้ในงานวิจัย

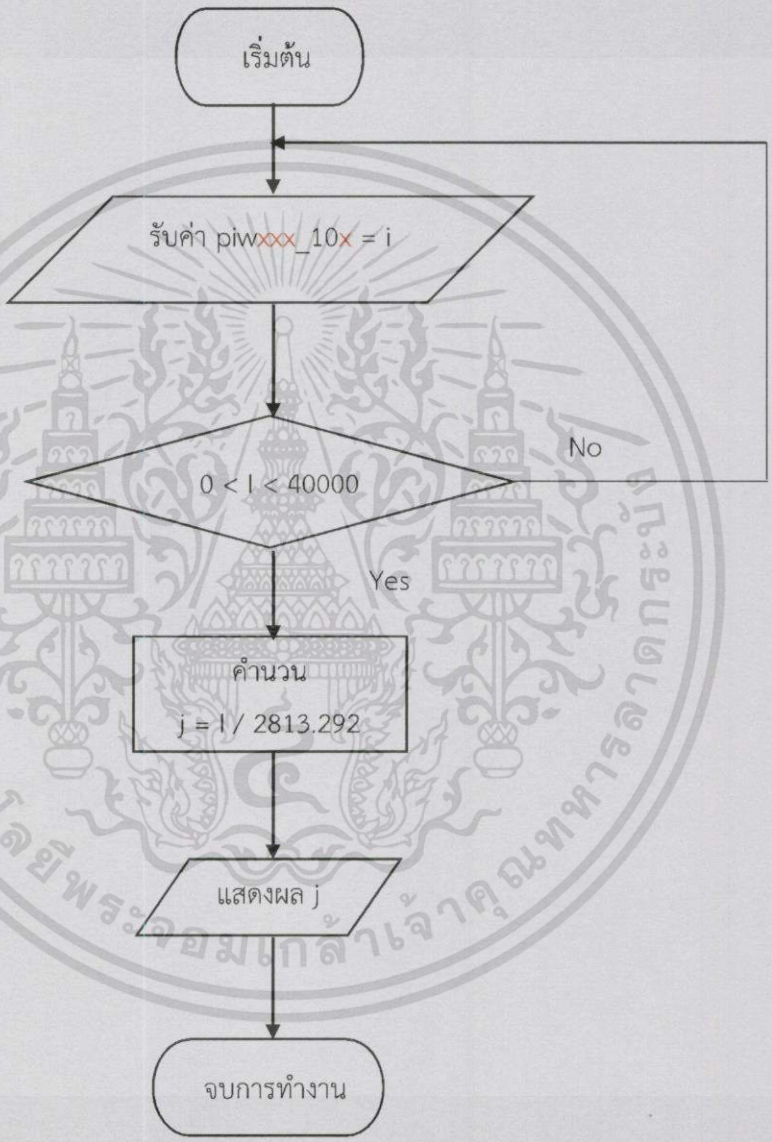
ณ แผงควบคุมประกอบไปด้วยหน่วยอินพุทและหน่วยแสดงผลที่เป็นไฟบอกสถานะหรือแสดงผลเป็นตัวเลขปุ่มกดขึ้นลง ในส่วนของกราฟิกที่แสดงผลเป็นตัวเลขทางด้านซ้ายมือ ซึ่งเชื่อมต่อกับช่องเอาท์พุทอนาล็อกมีหน้าที่แสดงผลคล้ายกับดิจิตอลโวลต์มิเตอร์ โดยในการวาดกราฟิกของดิจิตอลโวลต์มิเตอร์นี้ต้องการแสดงค่าออกมาเป็นอนาล็อกได้นั้นต้องอาศัยการเขียนโปรแกรมรับค่า ประมวลผลและแสดงผลออกมา ซึ่งใช้ภาษา VBScript (Visual Basic Script) ในการเขียนโปรแกรมเพื่อให้แสดงผลเป็นดิจิตอลโวลต์มิเตอร์และดิจิตอลเอาท์พุทตามที่ต้องการ โดยหน้าต่างสำหรับการเขียนโปรแกรมภาษา VBScript แสดงได้ดังรูปที่ 3.15



รูปที่ 3.15 หน้าต่างสำหรับการเขียนโปรแกรมภาษา VBScript

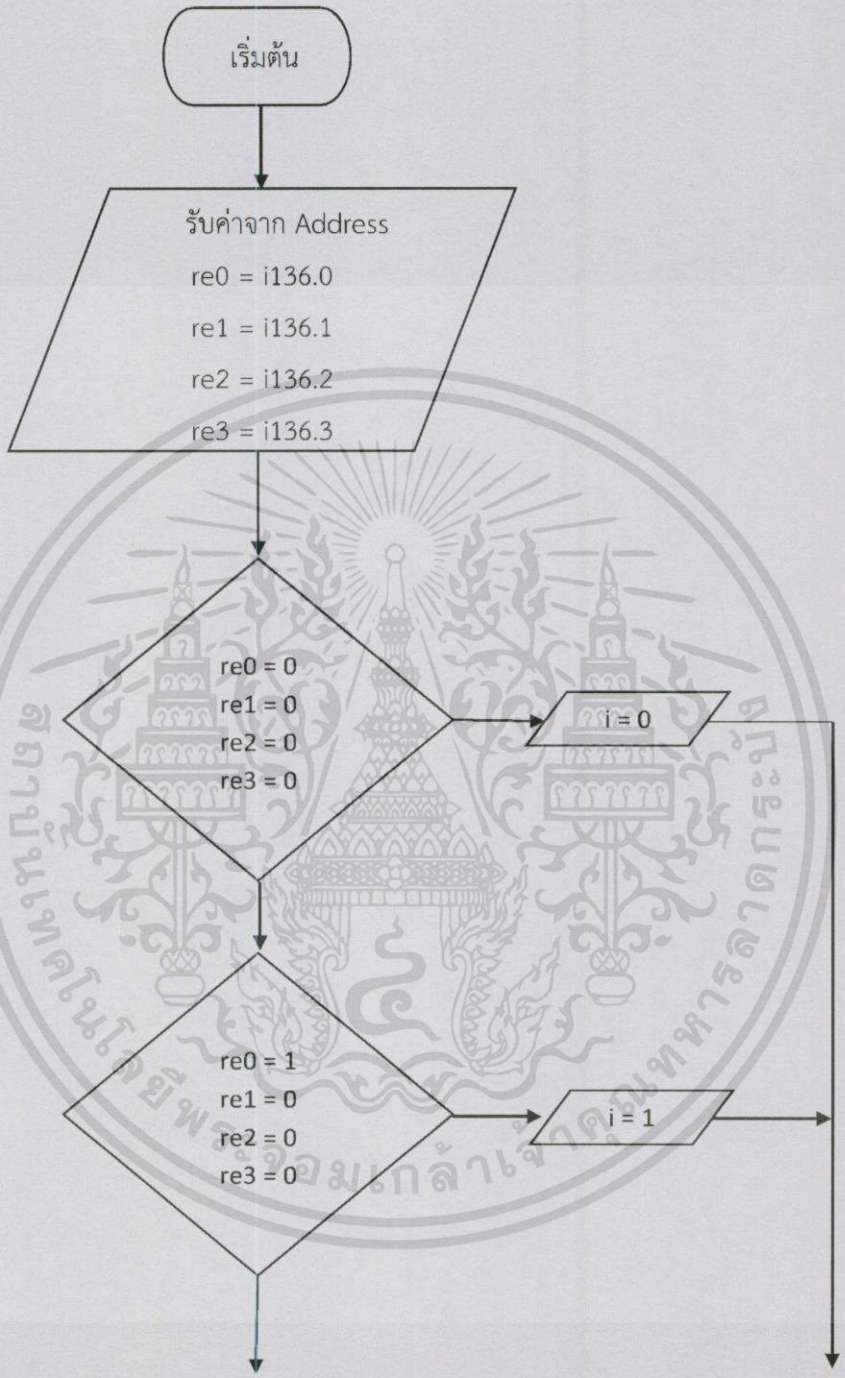
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการออกแบบกราฟิกเพื่อแสดงผลของดิจิตอลโวลต์มิเตอร์ ดิจิตอลเอาท์พุท และปุ่มกดเลือกแชแนล สามารถเขียนขั้นตอนการทำงานของโปรแกรมเป็นแผนผังดังนี้



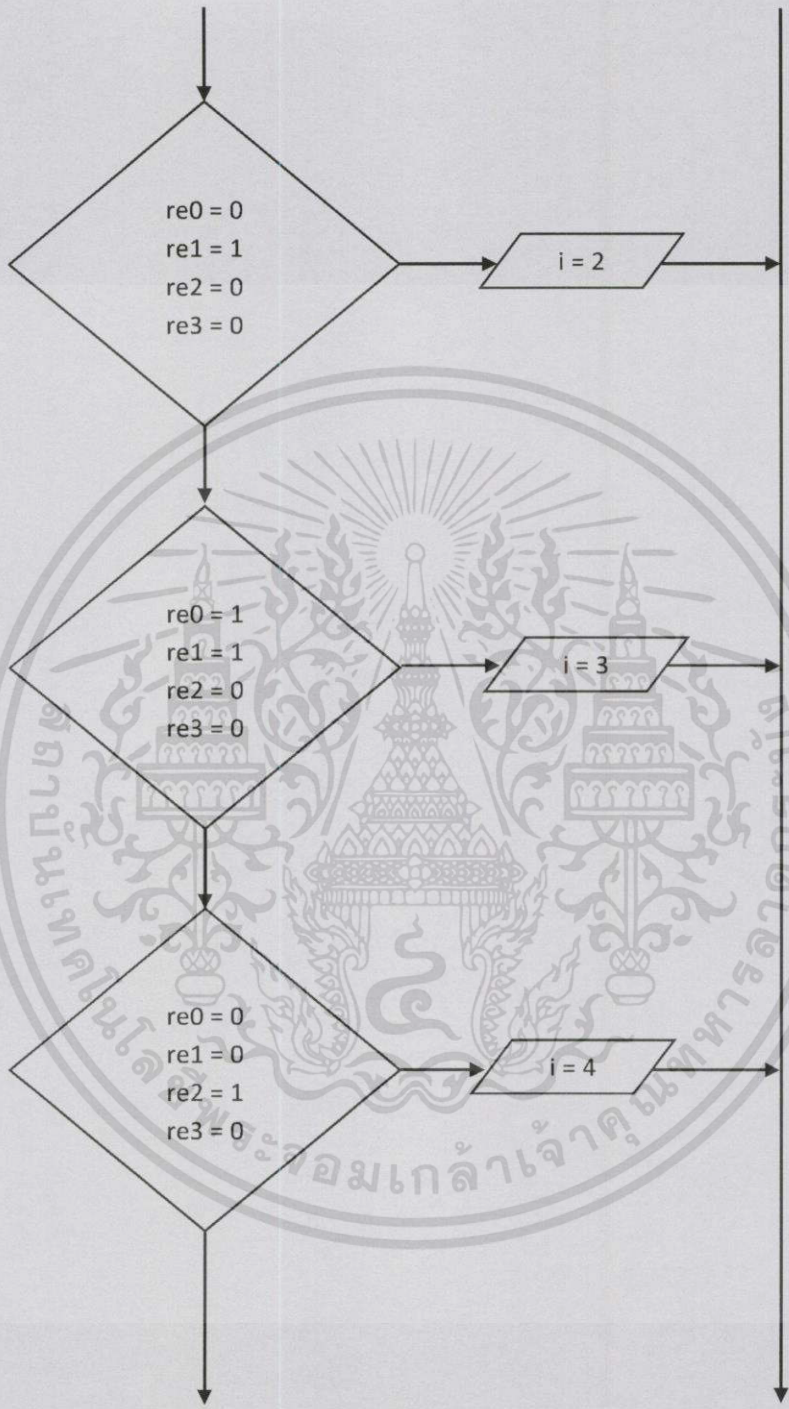
รูปที่ 3.16 ผังงานการทำงานของการแสดงผลดิจิตอลโวลต์มิเตอร์ทั้ง 8 ช่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



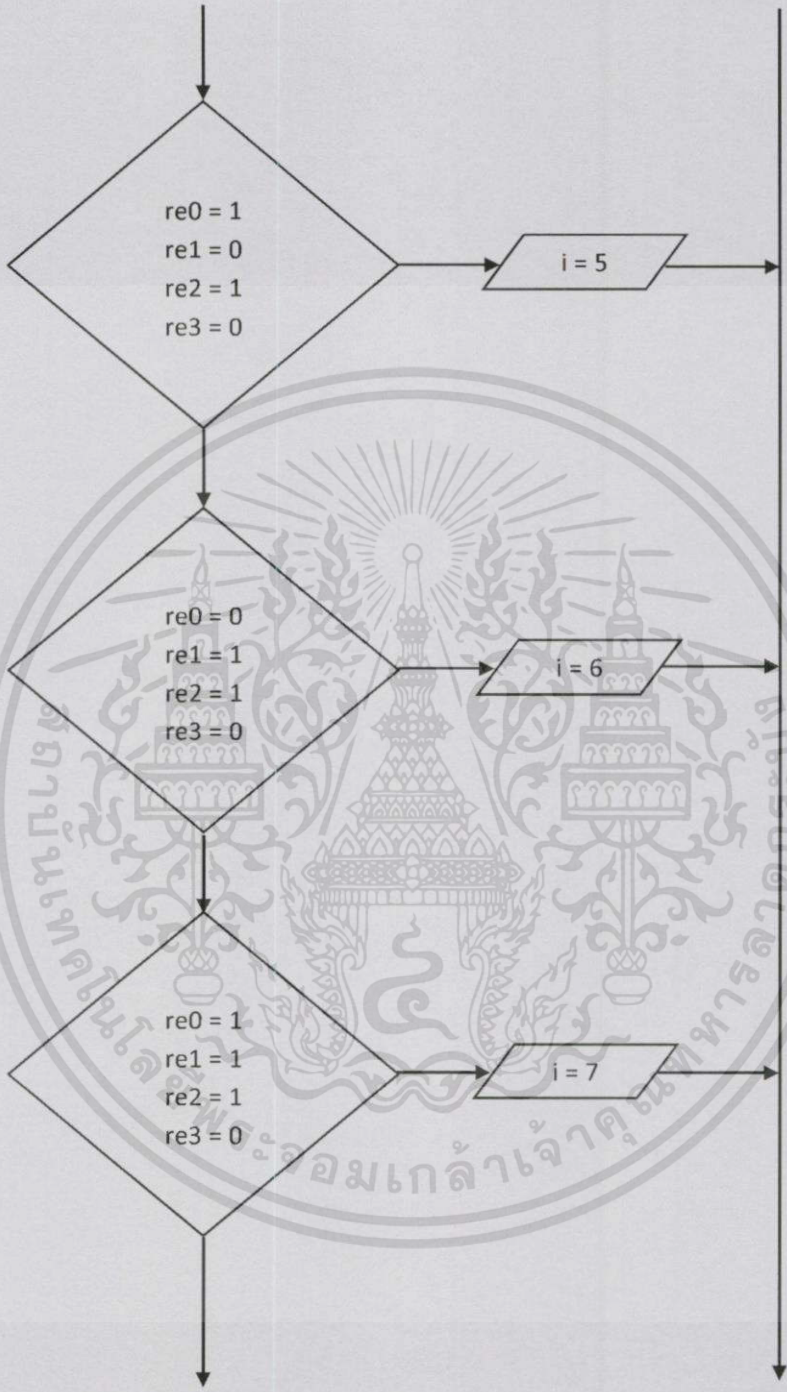
รูปที่ 3.17 แผนผังกระบวนการทำงานของดิจิตอลเอาท์พุทของแต่ละตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



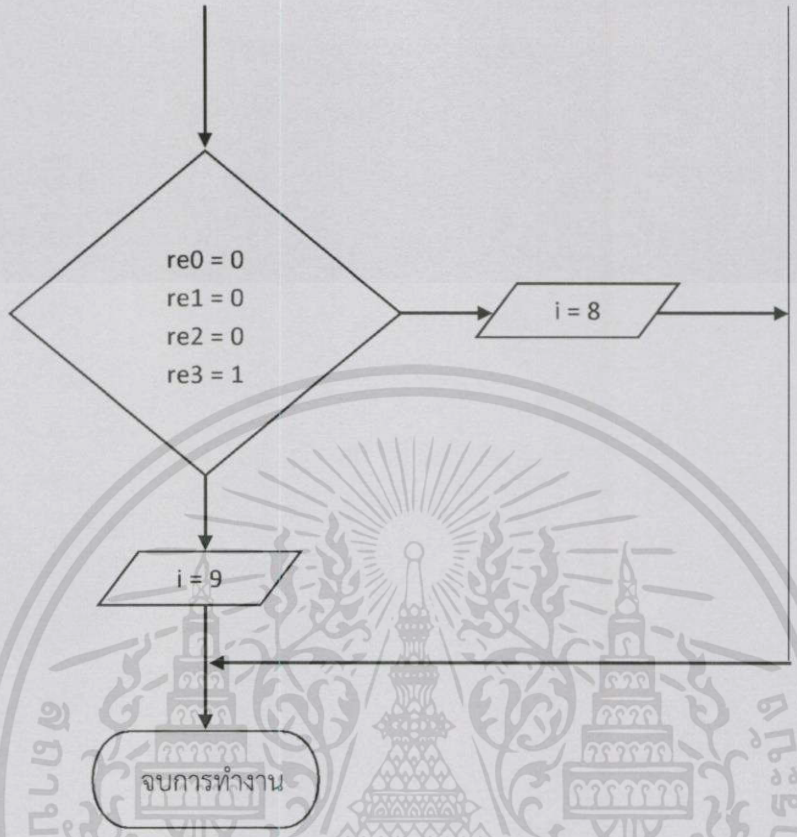
รูปที่ 3.18 แผนผังกระบวนการทำงานของดิจิตอลเอาท์พุทของแต่ละตำแหน่ง (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.19 แผนผังกระบวนการทำงานของดิจิตอลเอาท์พุทของแต่ละตำแหน่ง (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

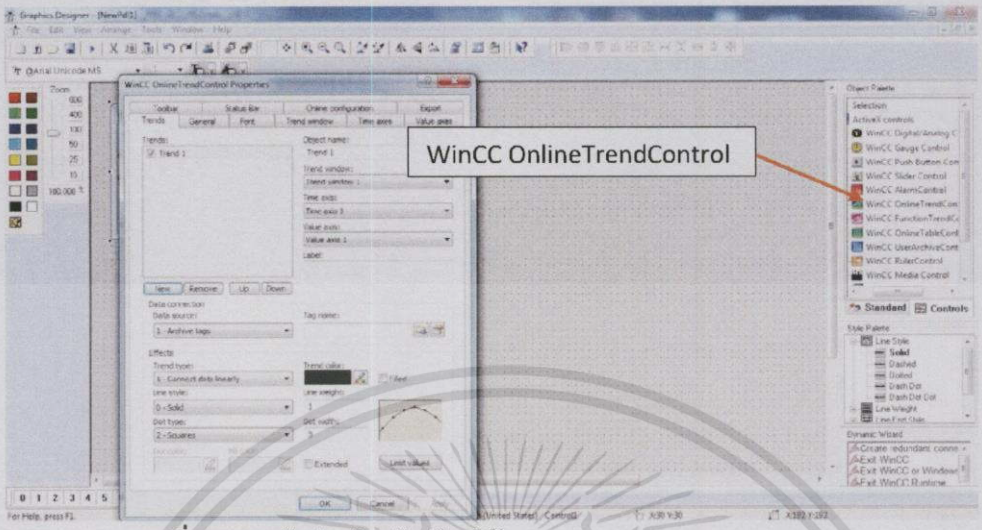


รูปที่ 3.20 แผนผังแสดงกระบวนการทำงานของดิจิตอลเอาท์พุทของแต่ละตำแหน่ง (ต่อ)

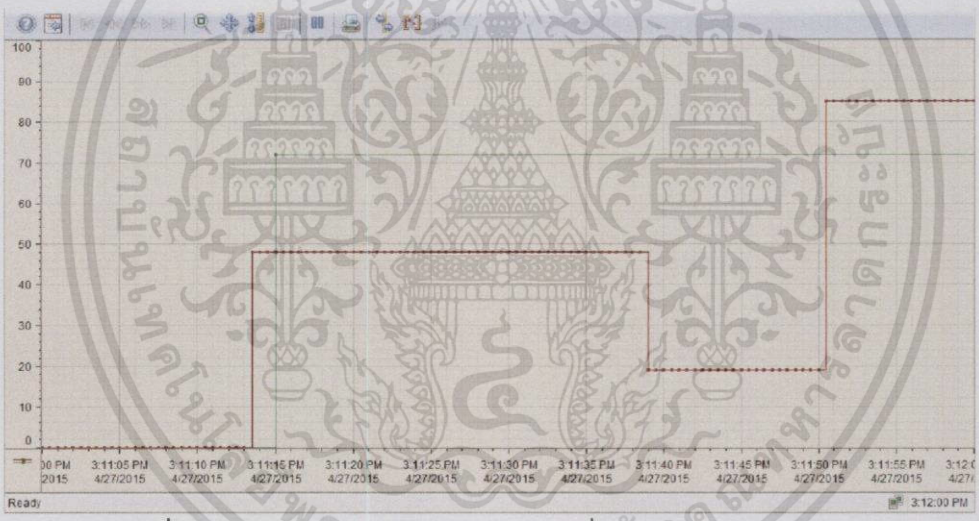
เมื่อเขียนกราฟิกทั้งหมดเสร็จแล้ว หากต้องการให้กราฟิกที่เขียนนั้นแสดงผลได้นั้นต้องทำการเปลี่ยนโหมดเป็น Runtime Mode เพื่อแสดงผลกราฟิกแผงควบคุมที่ได้เขียนขึ้นมา ซึ่งแสดงได้ดังรูปที่ 3.14

ต่อมาคือขั้นตอนการสร้างเทรนกราฟ เริ่มจากในหน้าต่าง Graphics Design จะมีรายการให้เลือกชื่อว่า WinCC OnlineTrendControl (รูปที่ 3.21) ซึ่งมีไว้สำหรับแสดงค่าอินพุทหรือเอาท์พุทเป็นเทรนกราฟใน WinCC Runtime โดย แล้วทำการตั้งค่าต่างๆของเทรนกราฟเช่น ชื่อเทรนกราฟ ตั้งค่าเวลาแกน ย่านเวลา ค่าในแกนกราฟ Tags ที่ต้องการแสดงผล เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.21 หน้าต่างสำหรับเลือกใช้ WinCC OnlineTrendControl

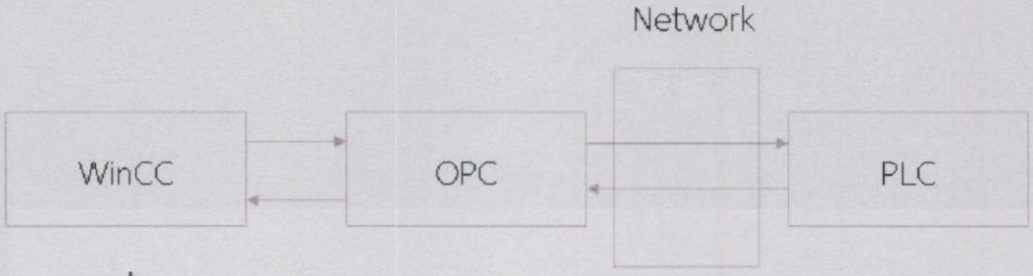


รูปที่ 3.22 ตัวอย่างหน้าต่างของเทรนกราฟเมื่ออยู่ใน Runtime Mode

3.4 ขั้นตอนการทำงานของระบบ SCADA

หลังจากตรวจสอบและติดตั้งการเชื่อมต่ออุปกรณ์ทางด้านฮาร์ดแวร์และการเชื่อมต่ออุปกรณ์ทางด้านซอฟต์แวร์เสร็จเรียบร้อยแล้ว จึงเริ่มดำเนินการตามขั้นตอนการดำเนินการตามขั้นตอนการทำงานของระบบ SCADA โดยระบบ SCADA ที่มีการใช้งาน OPC เป็นหน่วยเชื่อมต่อสื่อสารกลาง มีลำดับกระบวนการร้องขอข้อมูล เป็นดังต่อไปนี้คือ WinCC ร้องขอข้อมูลจาก OPC หลังจากนั้น OPC จะรับคำสั่งและทำตามคำสั่งการร้องขอข้อมูลแล้วพีแอลซีจะจัดเตรียมข้อมูลที่เหมาะสมตามคำเรียกร้องขอนั้นส่งกลับไปให้ OPC เพื่อให้ OPC ได้บันทึกข้อมูลลงฐานข้อมูลของ OPC และส่งกลับให้ WinCC ต่อไปตามแผนภาพ (รูปที่ 3.23)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

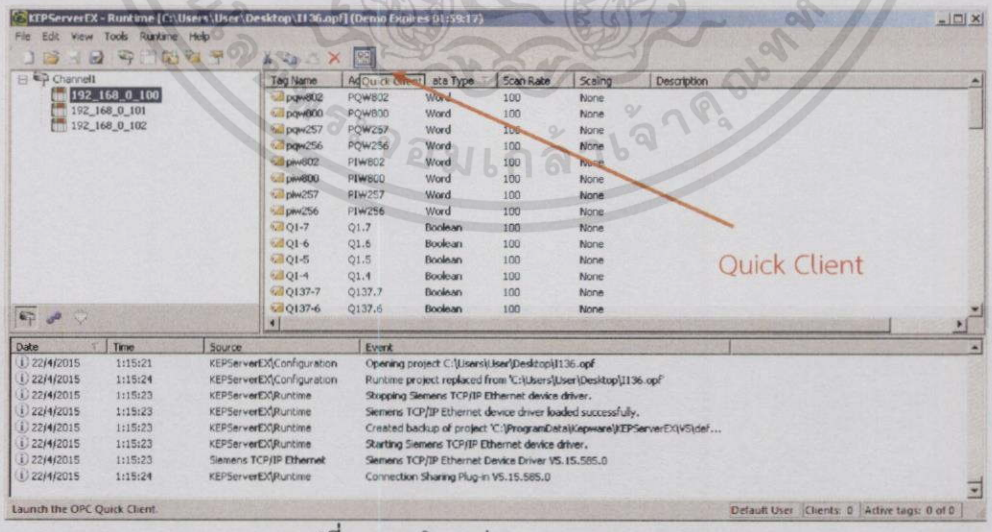


รูปที่ 3.23 แผนภาพกระบวนการร้องขอข้อมูลจากแต่ละตำแหน่งของระบบ SCADA

3.4.1 ขั้นตอนในการติดต่อกันระหว่าง PLC และ OPC Server

เริ่มจากการเขียนโปรแกรมแลดเดอร์โดยใช้แอดเดรสที่ต้องการพร้อมทั้งกำหนดสถานะในโปรแกรม Simatic Step 7 ซึ่งถ้าหากเป็นข้อมูลดิจิตอลจะเป็นข้อมูลประเภท Boolean หรือ Unsigned-32 bit ส่วนข้อมูลอนาล็อกจะเป็นข้อมูลประเภท Float ซึ่งข้อมูลเหล่านี้ได้นำมาใช้ในการกำหนดประเภทของข้อมูลในโปรแกรม KEPServerEX V5

เมื่อทำการเขียนโปรแกรมแลดเดอร์ที่ต้องการทดสอบพร้อมทั้งข้อมูลในแอดเดรสที่ได้ใช้ในการเขียนโปรแกรมแลดเดอร์แล้ว จากนั้นก็เข้าสู่ขั้นตอนของ OPC Server โดยเริ่มจากสร้างโปรเจคใหม่และทำการเพิ่มชาแนลและตั้งค่าอุปกรณ์ต่างๆในชาแนลที่ได้สร้างเอาไว้และใช้ชื่อว่า Channel1 ต่อจากนั้นก็ทำการเพิ่ม Tag สำหรับการร้องขอข้อมูล ซึ่งการตั้งค่าแต่ละ Tag มีวิธีการตั้งค่าที่แตกต่างกันตามประเภทของข้อมูล เมื่อทำการเพิ่ม Tag ทั้งหมดเรียบร้อยแล้วสามารถแสดงได้ดังตารางที่ 3.1 โดยถ้าหากต้องการมอนิเตอร์ค่าต่างๆที่มีการเชื่อมข้อมูลถึงกันแล้ว สามารถดูได้จากการคลิกที่ Quick Client ทางด้านบนของหน้าต่างโปรแกรม (รูปที่ 3.24) และหน้าต่างของ Quick Client แสดงได้ดังรูปที่ 3.25



รูปที่ 3.24 ตำแหน่งของ Quick Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 ค่ารายละเอียดของ Tag ในพีแอลซีทั้งสองเครื่อง

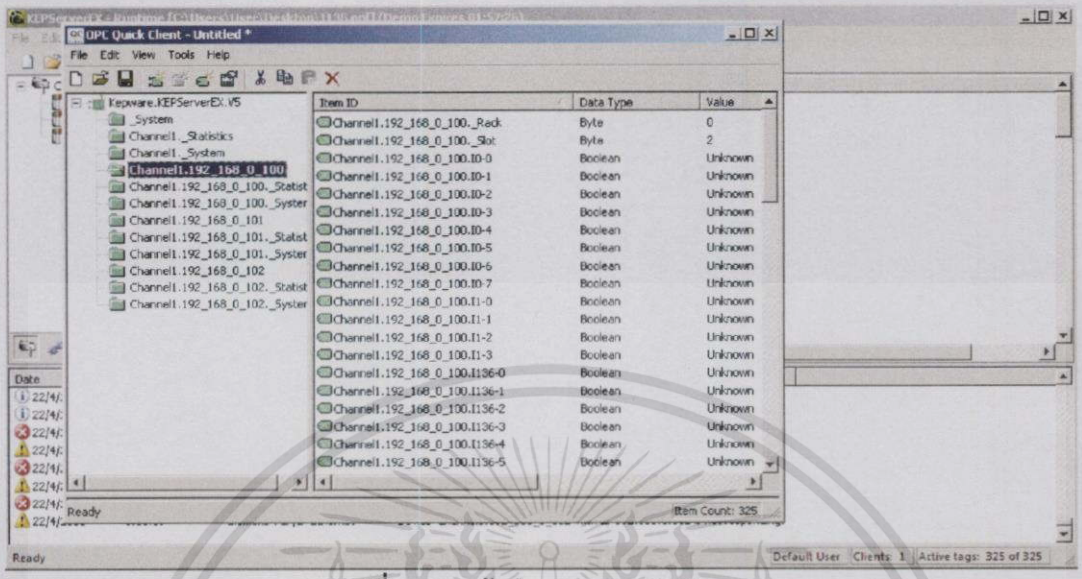
Tag Name	Address	Data Type	Scan Rate	Scaling
I136-0	I136.0	Boolean	100	None
I136-1	I136.1	Boolean	100	None
I136-2	I136.2	Boolean	100	None
I136-3	I136.3	Boolean	100	None
I136-4	I136.4	Boolean	100	None
I136-5	I136.5	Boolean	100	None
I136-6	I136.6	Boolean	100	None
I136-7	I136.7	Boolean	100	None
I137-0	I137.0	Boolean	100	None
I137-1	I137.1	Boolean	100	None
I137-2	I137.2	Boolean	100	None
I137-3	I137.3	Boolean	100	None
I137-4	I137.4	Boolean	100	None
I137-5	I137.5	Boolean	100	None
I137-6	I137.6	Boolean	100	None
I137-7	I137.7	Boolean	100	None
Q136-0	Q136.0	Boolean	100	None
Q136-1	Q136.1	Boolean	100	None
Q136-2	Q136.2	Boolean	100	None
Q136-3	Q136.3	Boolean	100	None
Q136-4	Q136.4	Boolean	100	None
Q136-5	Q136.5	Boolean	100	None
Q136-6	Q136.6	Boolean	100	None
Q136-7	Q136.7	Boolean	100	None
Q137-0	Q137.0	Boolean	100	None
Q137-1	Q137.1	Boolean	100	None
Q137-2	Q137.2	Boolean	100	None
Q137-3	Q137.3	Boolean	100	None
Q137-4	Q137.4	Boolean	100	None
Q137-5	Q137.5	Boolean	100	None
Q137-6	Q137.6	Boolean	100	None
Q137-7	Q137.7	Boolean	100	None

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 ค่ารายละเอียดของ Tag ในพีแอลซีทั้งสองเครื่อง (ต่อ)

Tag Name	Address	Data Type	Scan Rate	Scaling
pqw802	PQW802	Word	100	None
pqw800	PQW800	Word	100	None
pqw257	PQW257	Word	100	None
pqw256	PQW256	Word	100	None
piw802	PIW802	Word	100	None
piw800	PIW800	Word	100	None
piw256	PIW256	Word	100	None
piw257	PIW257	Word	100	None
I0-0	I0.0	Boolean	100	None
I0-1	I0.1	Boolean	100	None
I0-2	I0.2	Boolean	100	None
I0-3	I0.3	Boolean	100	None
I0-4	I0.4	Boolean	100	None
I0-5	I0.5	Boolean	100	None
I0-6	I0.6	Boolean	100	None
I0-7	I0.7	Boolean	100	None
I1-0	I1.0	Boolean	100	None
I1-1	I1.1	Boolean	100	None
I1-2	I1.2	Boolean	100	None
I1-3	I1.3	Boolean	100	None
I1-4	I1.4	Boolean	100	None
I1-5	I1.5	Boolean	100	None
I1-6	I1.6	Boolean	100	None
I1-7	I1.7	Boolean	100	None
Q0-0	Q0.0	Boolean	100	None
Q0-1	Q0.1	Boolean	100	None
Q0-2	Q0.2	Boolean	100	None
Q0-3	Q0.3	Boolean	100	None
Q0-4	Q0.4	Boolean	100	None
Q0-5	Q0.5	Boolean	100	None
Q0-6	Q0.6	Boolean	100	None
Q0-7	Q0.7	Boolean	100	None
Q1-0	Q1.0	Boolean	100	None
Q1-1	Q1.1	Boolean	100	None
Q1-2	Q1.2	Boolean	100	None
Q1-3	Q1.3	Boolean	100	None
Q1-4	Q1.4	Boolean	100	None
Q1-5	Q1.5	Boolean	100	None
Q1-6	Q1.6	Boolean	100	None
Q1-7	Q1.7	Boolean	100	None

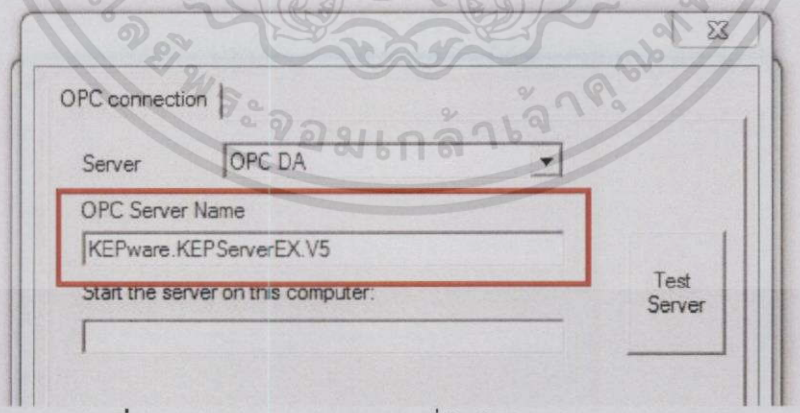
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.25 หน้าต่างของ Quick Client

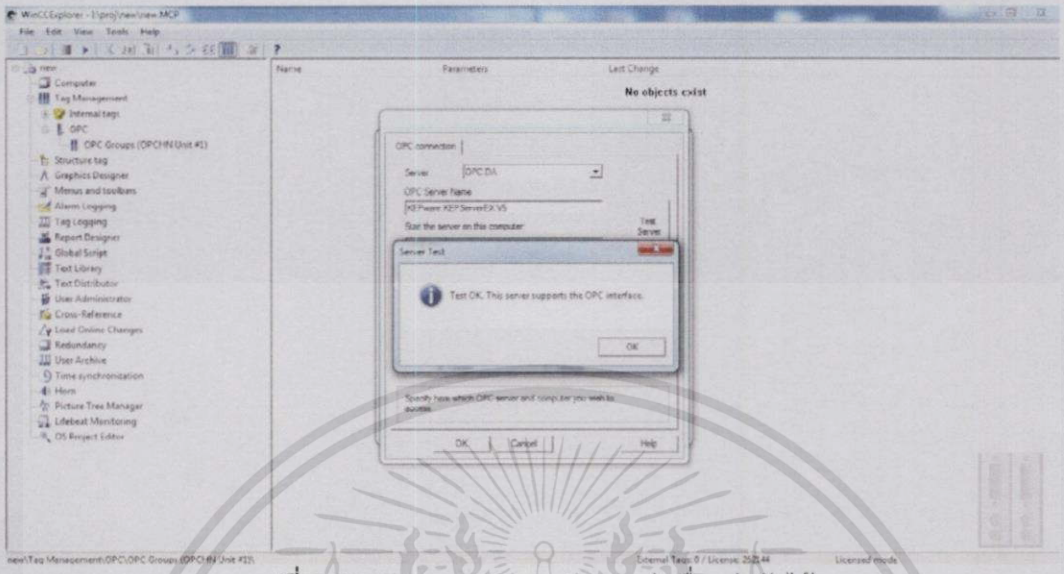
3.4.2 ขั้นตอนในการติดต่อกันระหว่าง WinCC และ OPC Server

การติดต่อกันระหว่างกันของ WinCC และ OPC Server ซึ่งใช้โปรแกรม KEPServerEX V5 เป็น OPC Server ต่อจากนั้นต้องมาตั้งค่าต่างๆในโปรแกรม WinCC โดยเมื่อสร้างโปรเจคใหม่ ในขั้นตอนของการเลือกไดร์ฟเวอร์สำหรับการเขียนกราฟิก ในหน้าต่าง Tag Management ต้องเลือกเป็นไดร์ฟเวอร์ OPC.chn เพื่อใช้ในการติดต่อกับ OPC Server และตั้งค่าชื่อสำหรับการเชื่อมต่อกัน OPC Server name ซึ่งเราใช้โปรแกรม KEPServerEx V5 ต้องใส่ชื่อเป็น KEPware.KEPServerEX.V5 (รูปที่ 3.23) แล้วทำการคลิก Test Server เพื่อทดสอบว่าเชื่อมต่อกันได้หรือเปล่าดังรูปที่ 3.27



รูปที่ 3.26 OPC Server Name ที่ต้องใส่สำหรับการทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.27 การทดสอบ Test Server ว่าเชื่อมต่อกันได้

จากนั้นก็ทำการสร้าง Tag ใหม่โดยรายละเอียดของการตั้งค่าแต่ละ Tag นั้นต้องมีค่าต่างๆเหมือนกันกับ Tag ใน OPC Server เพื่อเตรียมพร้อมสำหรับการเชื่อมต่อกับกราฟิกที่เขียนขึ้นมา ซึ่งขั้นตอนรายละเอียดต่างๆแสดงให้เห็นในภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

ในงานวิจัยนี้ได้วางแผนการศึกษาและทดลองว่าระบบ SCADA ที่ได้ศึกษามานั้นสามารถใช้งานได้จริง กราฟิกสามารถแสดงผลได้ตรงตามที่อุปกรณ์ฮาร์ดแวร์แสดงผลและแสดงเทรนกราฟที่มีค่าต่างๆ ตามที่ต้องการแสดง

โดยได้มีการแบ่งการทดลองออกเป็น 4 การทดลอง เริ่มจากทดลองสับสวิทช์ที่อุปกรณ์ฮาร์ดแวร์แล้วสังเกตการแสดงผลของแผงควบคุมจำลอง ทดลองการแสดงผลดิจิตอลเอาท์พุทแผงควบคุมจำลอง ทดลองการแสดงผลของดิจิตอลโวลต์มิเตอร์ซึ่งเป็นข้อมูลแบบอนาล็อก และสุดท้ายทดลองเขียนโปรแกรมด้วยคำสั่งพื้นฐานเพื่อทดสอบการแสดงผล

4.1 การทดสอบสับสวิทช์ที่แผงควบคุม

การทดลองนี้จัดทำเพื่อเป็นการทดสอบว่าสถานะต่างๆ ที่สับสวิทช์ ณ แผงควบคุมต้นแบบนั้นสามารถแสดงค่าหรือสถานะในแผงควบคุมจำลองหรือกราฟิกที่เขียนมานั้นให้มีสถานะที่ตรงกัน โดยในการทดลองจะสมมติการทดลองออกเป็น 4 แบบการทดลองโดยมีลำดับการทดลองดังนี้

4.1.1 ลำดับขั้นตอนการทดลอง

4.1.1.1 ตรวจสอบสวิทช์บนแผงควบคุมต้นแบบว่าอยู่ในสถานะปิดทั้งหมด

4.1.1.2 ตรวจสอบความถูกต้องของการเชื่อมต่อแผงควบคุมต้นแบบกับพีแอลซี

4.1.1.3 ตรวจสอบความถูกต้อง และเลื่อนสวิทช์ควบคุมโหมดของ CPU ให้อยู่ที่ตำแหน่ง RUN

4.1.1.4 กำหนดการทดลองโดยการโยกสวิทช์อินพุทจำลองการทำงานให้ข้อมูลเป็นไปตามตารางผลการทดลอง 4.1 - 4.4 ตามลำดับ

4.1.1.5 ทำการเฝ้าสังเกตสถานะของแผงควบคุมต้นแบบและแผงควบคุมจำลองว่ามีสถานะตรงกันหรือไม่

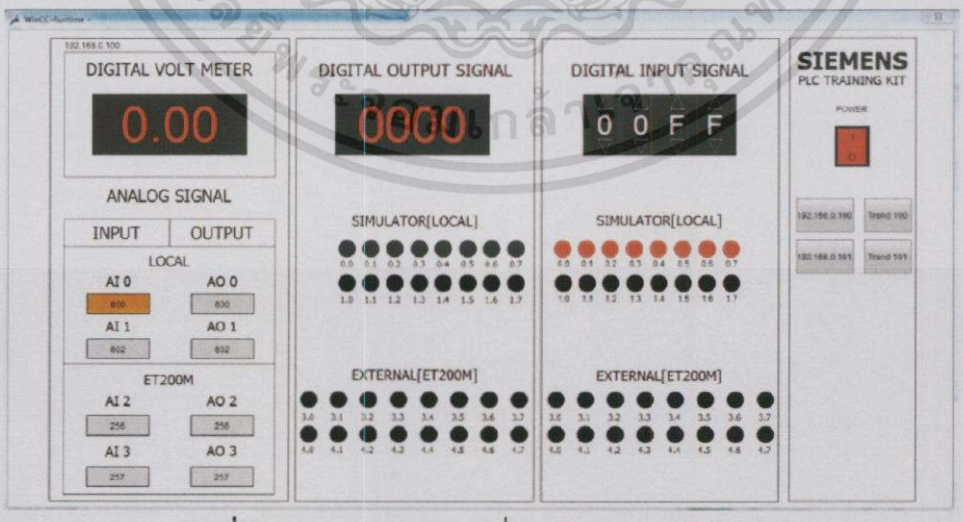
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ผลการทดลอง

ตารางต่อไปนี้เป็นตารางบันทึกผลการทดลองของทั้ง 4 แบบ

ตารางที่ 4.1 ผลการทดลองแบบที่ 1

แอดเดรส	สถานะอินพุท (แผงควบคุมต้นแบบ)	สถานะเอาท์พุท (แผงควบคุมจำลอง)
I136.0	เปิด	เปิด
I136.1	เปิด	เปิด
I136.2	เปิด	เปิด
I136.3	เปิด	เปิด
I136.4	เปิด	เปิด
I136.5	เปิด	เปิด
I136.6	เปิด	เปิด
I136.7	เปิด	เปิด
I137.0	ปิด	ปิด
I137.1	ปิด	ปิด
I137.2	ปิด	ปิด
I137.3	ปิด	ปิด
I137.4	ปิด	ปิด
I137.5	ปิด	ปิด
I137.6	ปิด	ปิด
I137.7	ปิด	ปิด

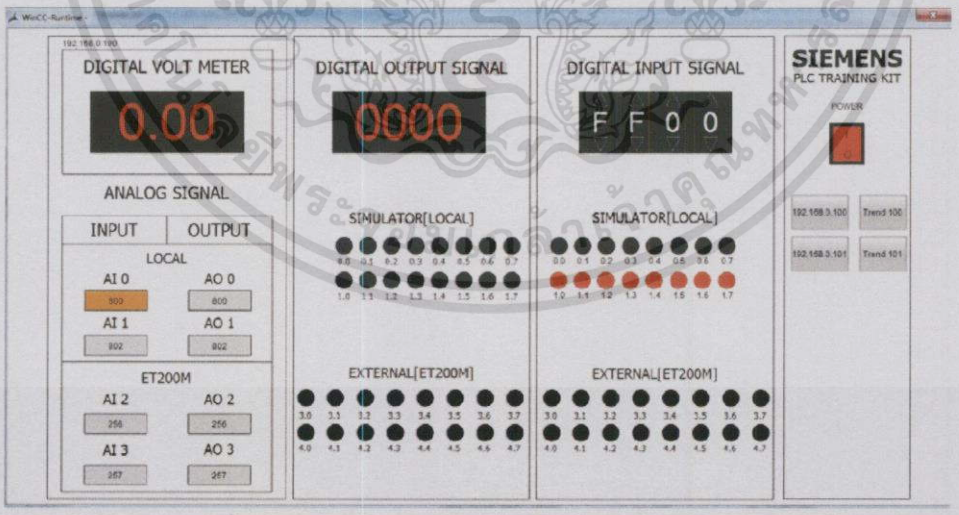


รูปที่ 4.1 ผลการทดลองแบบที่ 1 ของแผงควบคุมจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 ผลการทดลองที่ 2

แอดเดรส	สถานะอินพุท (แผงควบคุมต้นแบบ)	สถานะเอาท์พุท (แผงควบคุมจำลอง)
I136.0	ปิด	ปิด
I136.1	ปิด	ปิด
I136.2	ปิด	ปิด
I136.3	ปิด	ปิด
I136.4	ปิด	ปิด
I136.5	ปิด	ปิด
I136.6	ปิด	ปิด
I136.7	ปิด	ปิด
I137.0	เปิด	เปิด
I137.1	เปิด	เปิด
I137.2	เปิด	เปิด
I137.3	เปิด	เปิด
I137.4	เปิด	เปิด
I137.5	เปิด	เปิด
I137.6	เปิด	เปิด
I137.7	เปิด	เปิด

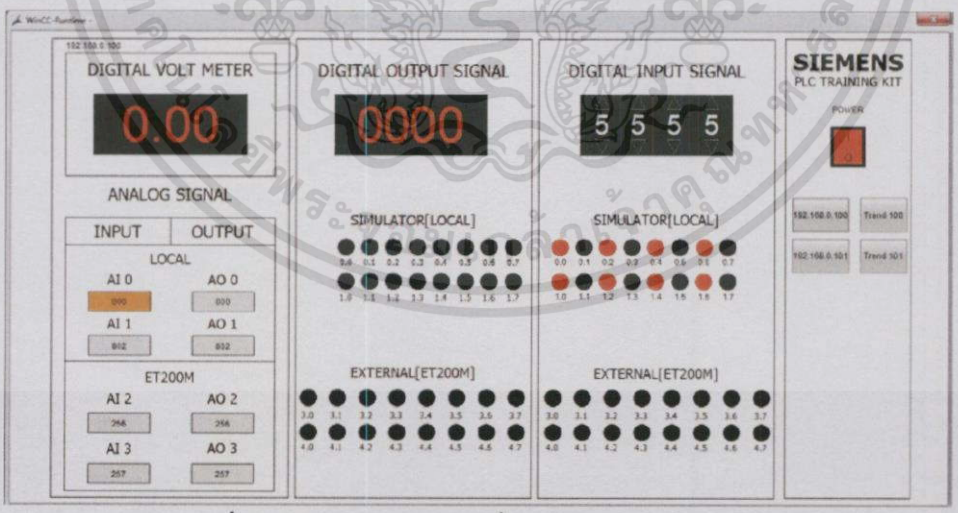


รูปที่ 4.2 ผลการทดลองแบบที่ 2 ของแผงควบคุมจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ผลการทดลองแบบที่ 3

แอดเดรส	สถานะอินพุท (แผงควบคุมต้นแบบ)	สถานะเอาท์พุท (แผงควบคุมจำลอง)
I136.0	เปิด	เปิด
I136.1	ปิด	ปิด
I136.2	เปิด	เปิด
I136.3	ปิด	ปิด
I136.4	เปิด	เปิด
I136.5	ปิด	ปิด
I136.6	เปิด	เปิด
I136.7	ปิด	ปิด
I137.0	เปิด	เปิด
I137.1	ปิด	ปิด
I137.2	เปิด	เปิด
I137.3	ปิด	ปิด
I137.4	เปิด	เปิด
I137.5	ปิด	ปิด
I137.6	เปิด	เปิด
I137.7	ปิด	ปิด

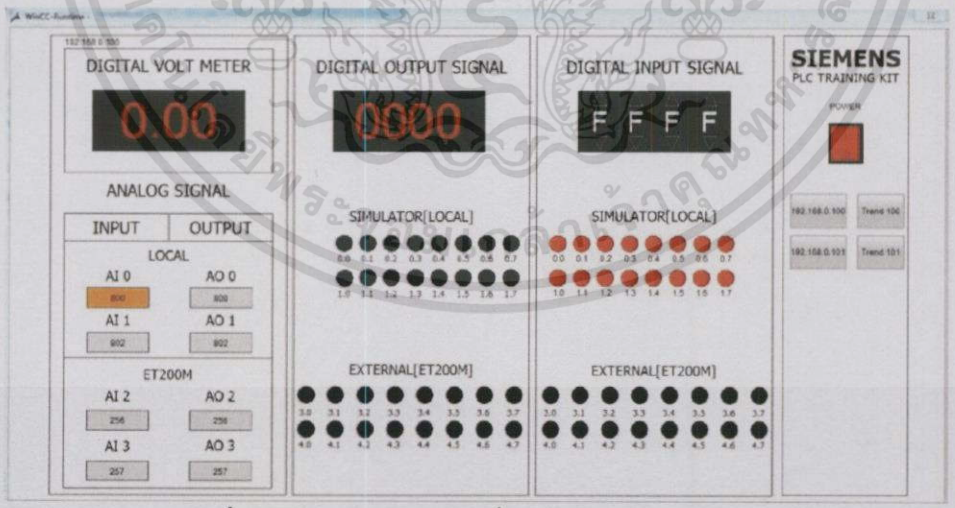


รูปที่ 4.3 ผลการทดลองแบบที่ 3 ของแผงควบคุมจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 ผลการทดลองแบบที่ 4

แอดเดรส	สถานะอินพุท (แผงควบคุมต้นแบบ)	สถานะเอาท์พุท (แผงควบคุมจำลอง)
I136.0	เปิด	เปิด
I136.1	เปิด	เปิด
I136.2	เปิด	เปิด
I136.3	เปิด	เปิด
I136.4	เปิด	เปิด
I136.5	เปิด	เปิด
I136.6	เปิด	เปิด
I136.7	เปิด	เปิด
I137.0	เปิด	เปิด
I137.1	เปิด	เปิด
I137.2	เปิด	เปิด
I137.3	เปิด	เปิด
I137.4	เปิด	เปิด
I137.5	เปิด	เปิด
I137.6	เปิด	เปิด
I137.7	เปิด	เปิด



รูปที่ 4.4 ผลการทดลองแบบที่ 4 ของแผงควบคุมจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดสอบการแสดงผลแบบดิจิตอล

การทดลองนี้จัดทำเพื่อเป็นการทดสอบว่าหากทำการย้ายค่าจากอินพุทไปสู่เอาท์พุทที่แสดงเป็นตัวเลข สามารถแสดงค่าตัวเลขในแผงควบคุมจำลองหรือกราฟิกที่เขียนมานั้นให้มีค่าที่ตรงกัน โดยในการทดลองจะสมมติการทดลองออกเป็นส่วนของ local I/O Area และ Remote I/O Area ซึ่งแต่ละสเตชั่นจะแยกออกเป็นแชแนลที่ 136, 137, 0, 1 ตามลำดับ โดยมีลำดับการทดลองดังนี้

4.2.1 ลำดับขั้นตอนการทดลอง

4.2.1.1 ตรวจสอบสวิทช์เลือกโหมดการทำงาน โดยตำแหน่งจะต้องอยู่ที่หยุด (STOP)

4.2.1.2 เพิ่มเติมโปรแกรมการทำงานใหม่ตามรูปที่ 4.5 ตรวจสอบความถูกต้องของโปรแกรม

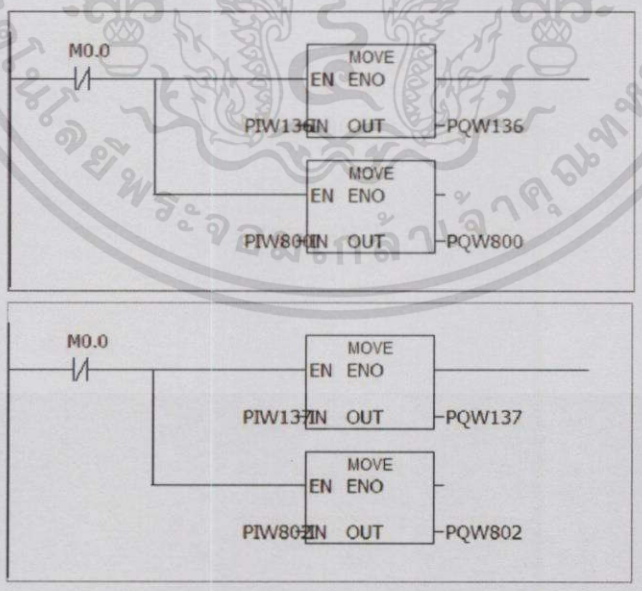
4.2.1.3 ดาวน์โหลดโปรแกรมการทำหรือหรือแลดเดอร์ที่เขียน

4.2.1.4 ตรวจสอบความถูกต้อง และเลื่อนสวิทช์โหมดของ CPU ให้อยู่โหมด RUN

4.2.1.5 กำหนดสัญญาณ I/P ทั้งหมดในส่วนของ Local แต่ละตัว โดยการโยกที่อินพุทสวิทช์จำลองการทำงานให้ข้อมูลเป็นไปตามตารางที่ 4.5

4.2.1.6 กำหนดสัญญาณ I/P ทั้งหมดในส่วนของ Remote แต่ละตัว โดยการโยกที่อินพุทสวิทช์จำลองการทำงานให้ข้อมูลเป็นไปตามตารางที่ 4.5

4.2.1.7 ทำการเฝ้าสังเกตสถานะของแผงควบคุมต้นแบบและแผงควบคุมจำลองว่ามีสถานะตรงกันหรือไม่



รูปที่ 4.5 โปรแกรมทดสอบสัญญาณดิจิตอล

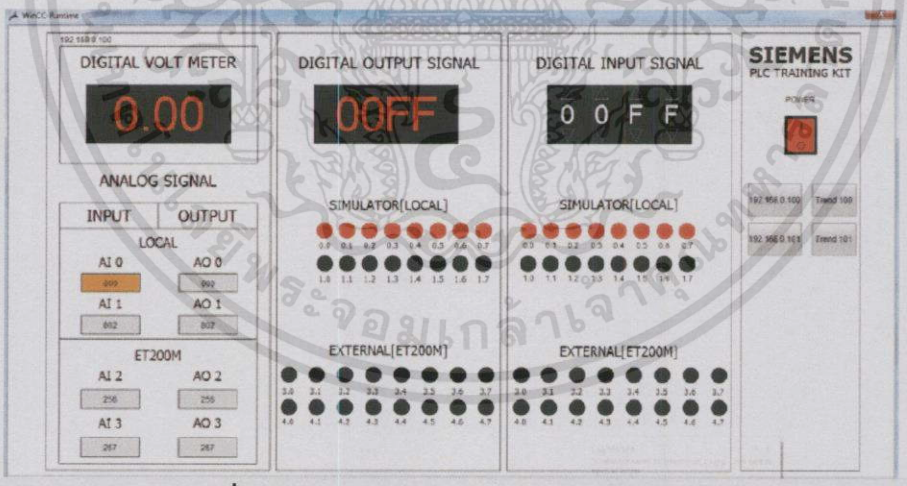
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 ผลการทดลอง

ตารางต่อไปนี้เป็นผลการทดลองของการทดสอบการแสดงผลดิจิตอลทั้ง Local และ Remote ซึ่งค่าที่ได้นำมาทดสอบได้กำหนดและอยู่ในตารางผลการทดลองแล้ว

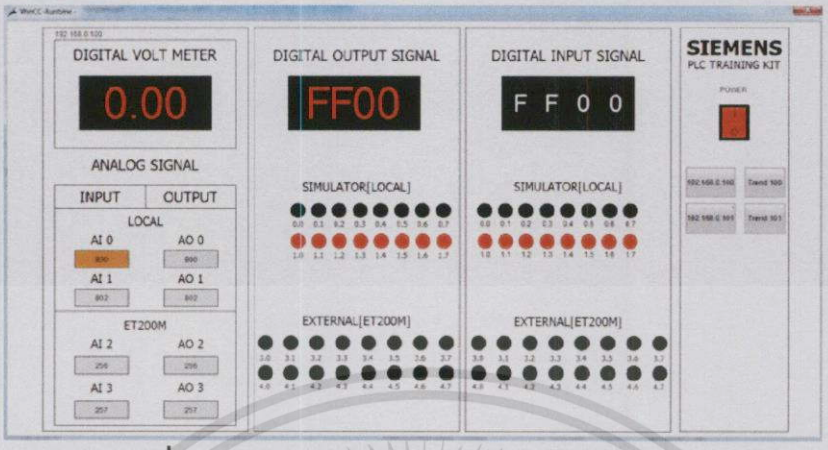
ตารางที่ 4.5 ผลการทดลองข้อมูลแบบดิจิตอล

CPU I/O area	
PIW136(HEX)	PQW136(HEX) แผงควบคุมจำลอง
1111	1111
2222	2222
4444	4444
8888	8888
00FF	00FF
FF00	FF00
AAAA	AAAA
5555	5555

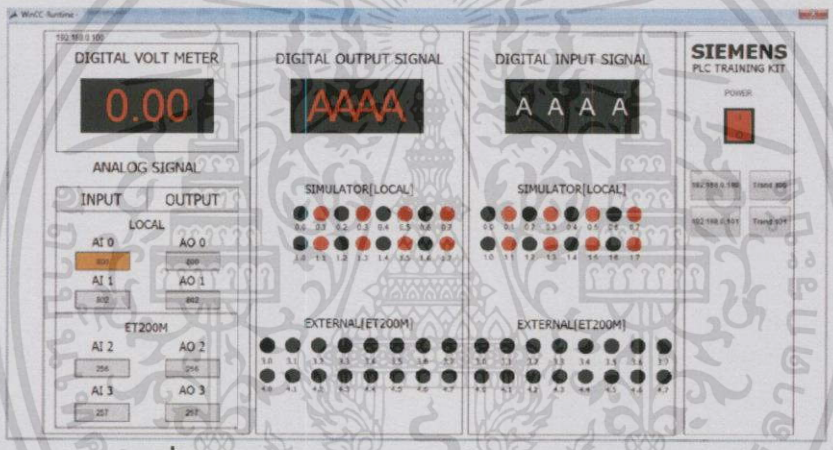


รูปที่ 4.6 ตัวอย่างผลการทดลองข้อมูลแบบดิจิตอล

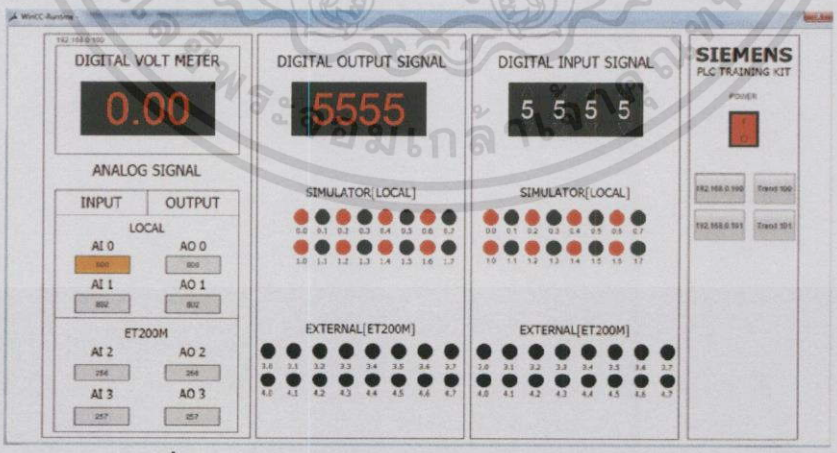
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 ตัวอย่างผลการทดลองข้อมูลแบบดิจิตอล (ต่อ)



รูปที่ 4.8 ตัวอย่างผลการทดลองข้อมูลแบบดิจิตอล (ต่อ)



รูปที่ 4.9 ตัวอย่างผลการทดลองข้อมูลแบบดิจิตอล (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดสอบการแสดงผลแบบอนาล็อก

การทดลองนี้จัดทำเพื่อเป็นการทดสอบกับข้อมูลอนาล็อก โดยทดสอบกับดิจิตอลโวลต์มิเตอร์ที่ติดอยู่ ณ แผงควบคุมต้นแบบ และสามารถแสดงค่าตัวเลขในแผงควบคุมจำลองหรือกราฟิกที่เขียนซึ่งเป็นดิจิตอลโวลต์มิเตอร์ให้มีค่าที่ตรงกัน โดยในการทดลองจะสมมติการทดลองออกเป็นส่วนของ local I/O Area และ Remote I/O Area ซึ่งแต่ละสเตชั่นจะแยกออกเป็นแชแนลที่ 800, 802, 256, 257 ตามลำดับ โดยมีลำดับการทดลองดังนี้

4.3.1 ลำดับขั้นตอนการทดลอง

4.3.1.1 ตรวจสอบสวิทช์เลือกโหมดการทำงาน โดยตำแหน่งจะต้องอยู่ที่หยุด (STOP)

4.3.1.2 เพิ่มเติมโปรแกรมการทำงานใหม่ตามรูปที่ 4.10 ตรวจสอบความถูกต้องของโปรแกรม

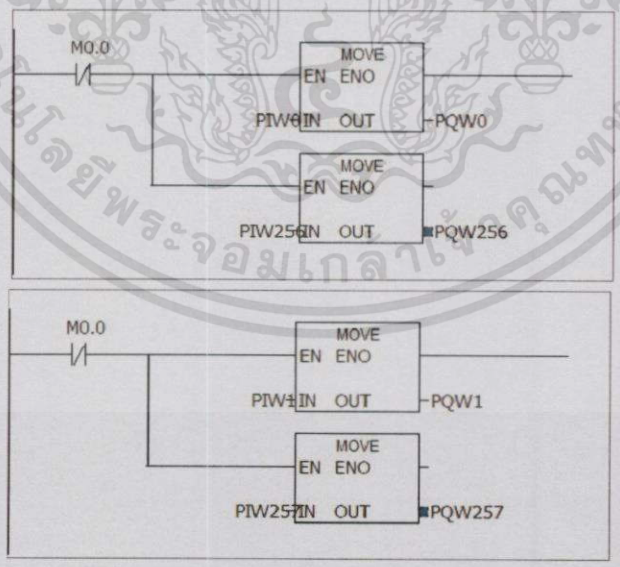
4.3.1.3 ดาวน์โหลดโปรแกรมการทำหรือหรือแลดเดอร์ที่เขียน

4.3.1.4 ตรวจสอบความถูกต้อง และเลื่อนสวิทช์โหมดของ CPU ให้อยู่โหมด RUN

4.3.1.5 เชื่อมต่อสัญญาณตามรูปที่ 4.11 ป้อนแรงดัน Vin ที่สัญญาณ I/P ทั้งในส่วนของ Local Vin Ch800 และ Remote Vin Ch0 ให้แรงดันเป็นไปตามตารางที่ 4.6

4.3.1.6 อ่านแรงดันที่ดิจิตอลโวลต์มิเตอร์ Vout Ch800 และ Vout Ch0

4.3.1.7 ทำการเฝ้าสังเกตสถานะของแผงควบคุมต้นแบบและแผงควบคุมจำลองว่ามีสถานะตรงกันหรือไม่



รูปที่ 4.10 โปรแกรมทดสอบสัญญาณอนาล็อก

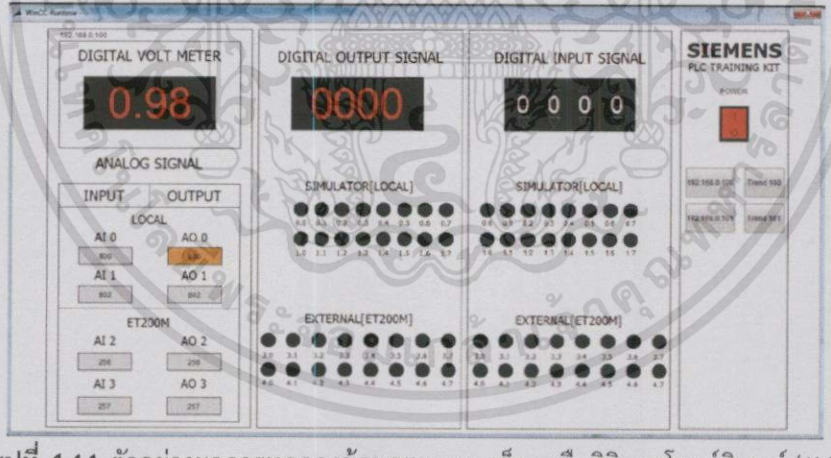
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 ผลการทดลอง

ตารางต่อไปนี้เป็นผลการทดลองของการทดสอบแสดงผลอนาล็อก โดยใช้ดิจิตอลโวลต์มิเตอร์แสดงผลออกมาผ่านแผงควบคุมจำลอง ซึ่งค่าผลการทดลองต่างๆได้อยู่ในตารางที่ 4.6

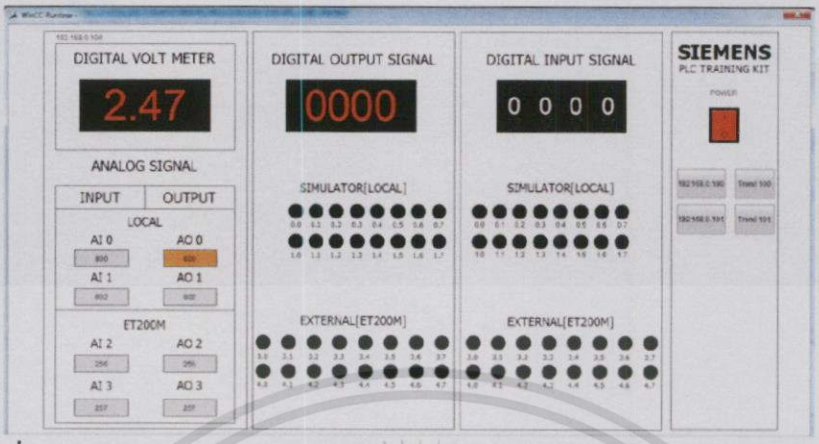
ตารางที่ 4.6 ผลการทดลองข้อมูลแบบอนาล็อก (ดิจิตอลโวลต์มิเตอร์)

CPU I/O area				Remote I/O area			
Vin CH800	Vout Ch800	Vout แผงควบคุมจำลอง	PQW800 (HEX)	Vin CH256	Vout Ch256	Vout แผงควบคุมจำลอง	PQW256 (HEX)
1 V.	0.99	0.98	00000ad0	1 V.	0.99	1.01	00000ad8
2.5 V.	2.5	2.47	00001b20	2.5 V.	2.48	2.41	00001a78
5.0 V.	4.99	4.92	00003610	5.0 V.	4.99	4.83	00003510
7.5 V.	7.51	7.39	00005140	7.5 V.	7.49	7.25	00004fa8
9 V.	9.00	8.86	00006160	9 V.	9.00	8.71	00005fb002

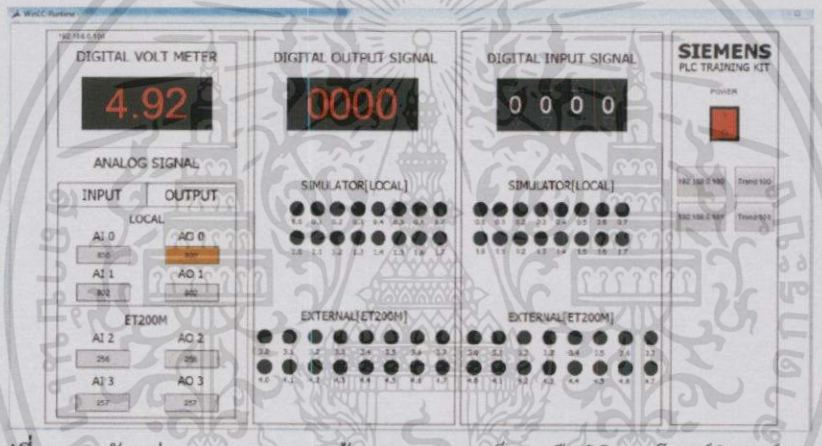


รูปที่ 4.11 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์ (1V)

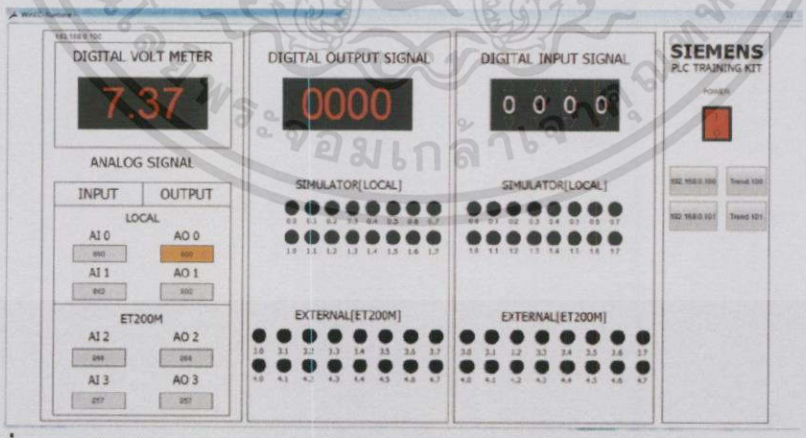
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์ (2.5V)

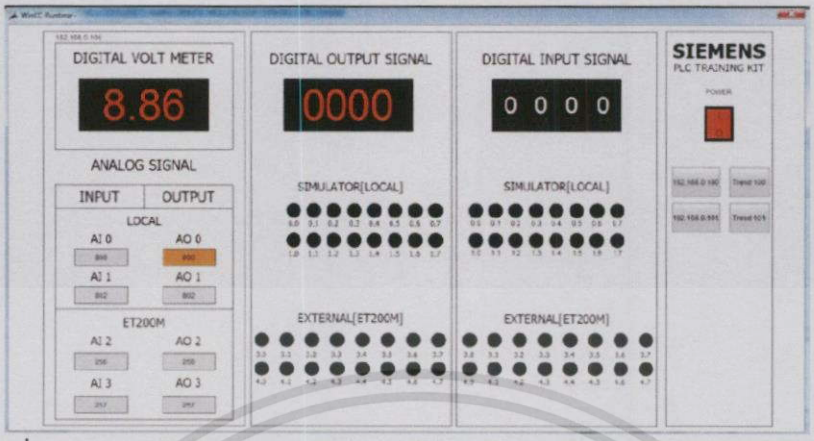


รูปที่ 4.13 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์ (5V)



รูปที่ 4.14 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์ (7.5V)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 ตัวอย่างผลการทดลองข้อมูลแบบอนาล็อกหรือดิจิตอลโวลต์มิเตอร์ (9V)

4.4 การทดสอบการแสดงผลเมื่อใช้การเขียนภาษาแลดเดอร์ขั้นพื้นฐาน

การทดลองนี้จัดทำเพื่อเป็นการทดสอบกับโปรแกรมแลดเดอร์ที่เป็นไปตามเงื่อนไขในแต่ละข้อ โดยพิจารณาเฉพาะสัญญาณอินพุทที่ส่งผลต่อเอาท์พุทให้มีสถานะเป็น “1” กรณีนี้เท่านั้น ส่วนสัญญาณอื่นๆที่ไม่ส่งผลต่อเอาท์พุท ให้ใส่ข้อมูลเป็น X (Don’t care) และเปรียบเทียบกับกราฟิกแผงควบคุมจำลองว่ามีสถานะที่เหมือนกันหรือไม่ตามตารางผลการทดลอง

4.4.1 ลำดับขั้นตอนการทดลอง

4.4.1.1 ตรวจสอบสวิทช์เลือกโหมดการทำงาน โดยตำแหน่งจะต้องอยู่ที่หยุด (STOP)

4.4.1.2 เพิ่มเติมโปรแกรมการทำงานใหม่ตามรูปที่ 4.17-4.22 ตรวจสอบความถูกต้องของโปรแกรม

4.4.1.3 ดาวน์โหลดโปรแกรมการทำหรือหรือแลดเดอร์ที่เขียน

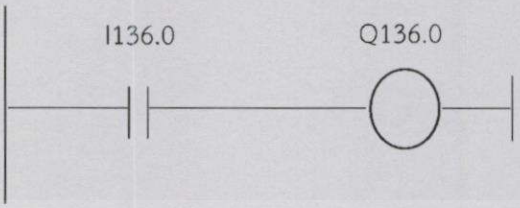
4.4.1.4 ตรวจสอบความถูกต้อง และเลื่อนสวิทช์โหมดของ CPU ให้อยู่โหมด RUN

4.4.1.5 กำหนดสัญญาณ I/P ทั้งหมด โดยการโยกที่อินพุทสวิทช์จำลองการทำงานให้ข้อมูลเป็นไปตามตารางที่ 4.7

4.4.1.6 สังเกตสถานะอินพุทที่ทำให้สถานะของเอาท์พุทเป็น “1”

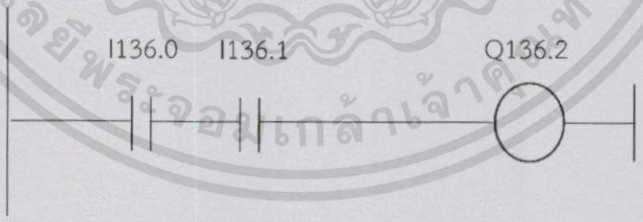
4.4.1.7 ทำการเฝ้าสังเกตสถานะของแผงควบคุมต้นแบบและแผงควบคุมจำลองว่ามีสถานะตรงกันหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



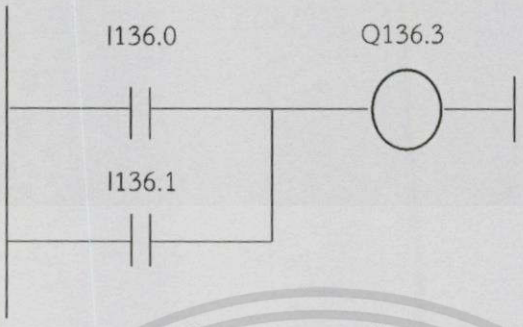
รูปที่ 4.16 แลดเดอร์พื้นฐาน 1

รูปที่ 4.17 แลดเดอร์พื้นฐาน 2

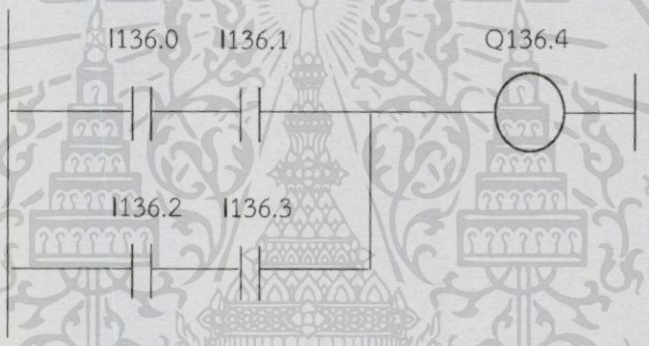


รูปที่ 4.18 แลดเดอร์พื้นฐาน 3

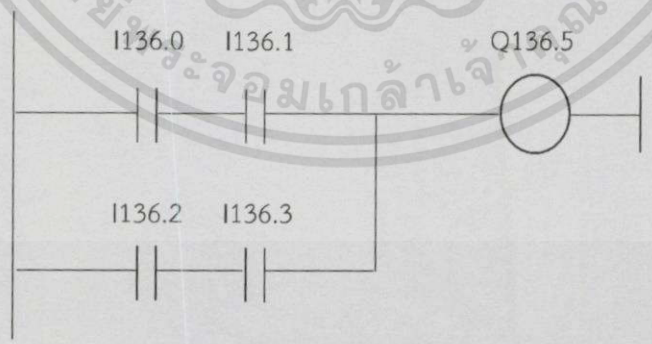
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 แลดเดอร์พื้นฐาน 4



รูปที่ 4.20 แลดเดอร์พื้นฐาน 5



รูปที่ 4.21 แลดเดอร์พื้นฐาน 6

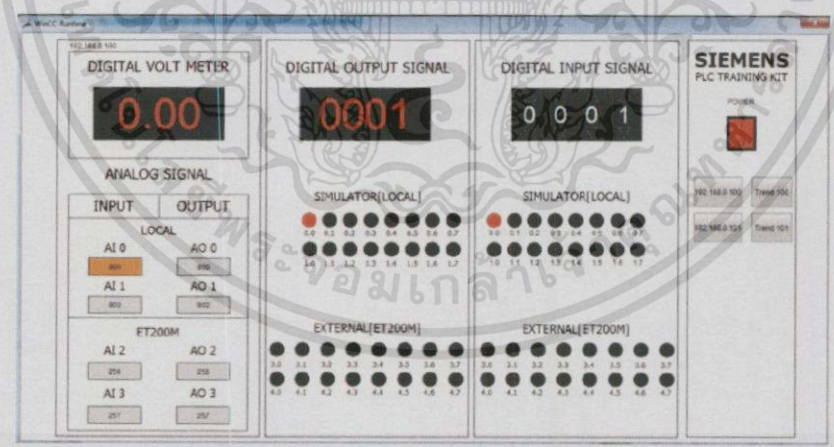
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.2 ผลการทดลอง

ตารางต่อไปนี้เป็นผลการทดลองของการทดสอบแสดงสถานะของอินพุทและเอาท์พุท เมื่อเขียนโปรแกรมแลดเดอร์เป็นคำสั่งพื้นฐานต่าง โดยแสดงผลออกมาผ่านแผงควบคุมจำลอง ซึ่งค่าผลการทดลองต่างๆได้อยู่ในตารางที่ 4.7

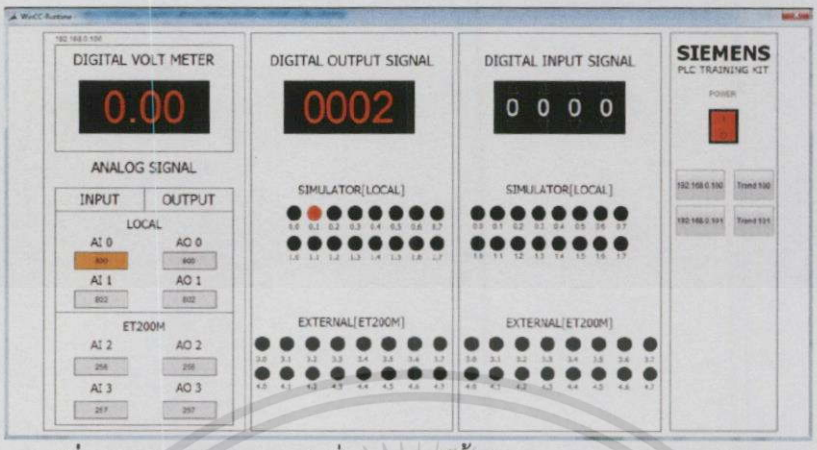
ตารางที่ 4.7 ผลการทดลองที่เป็นไปตามเงื่อนไขของโปรแกรมในแต่ละข้อ

	สถานะอินพุท				สถานะเอาท์พุท					
	I136.3	I136.2	I136.1	I136.0	Q136.5	Q136.4	Q136.3	Q136.2	Q136.1	Q136.0
1.	0	0	0	1	0	0	0	0	0	1
2.	0	0	0	0	0	0	0	0	1	0
3.	0	0	1	1	0	0	0	1	0	0
4.1	0	0	0	1	0	0	1	0	0	0
4.2	0	0	1	0	0	0	1	0	0	0
5.1	0	0	1	1	0	1	0	0	0	0
5.2	1	1	0	0	0	1	0	0	0	0
6.1	0	0	1	1	1	0	0	0	0	0
6.2	0	1	1	0	1	0	0	0	0	0
6.3	1	0	0	1	1	0	0	0	0	0
6.4	1	1	0	0	1	0	0	0	0	0

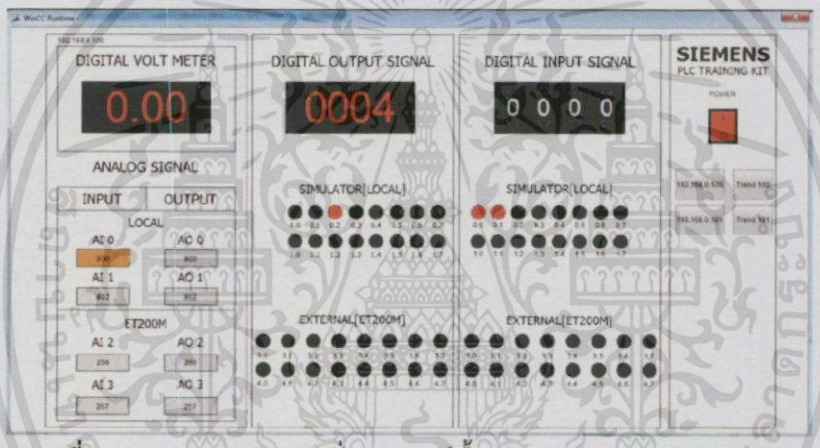


รูปที่ 4.22 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 1 ณ แผงควบคุมจำลอง

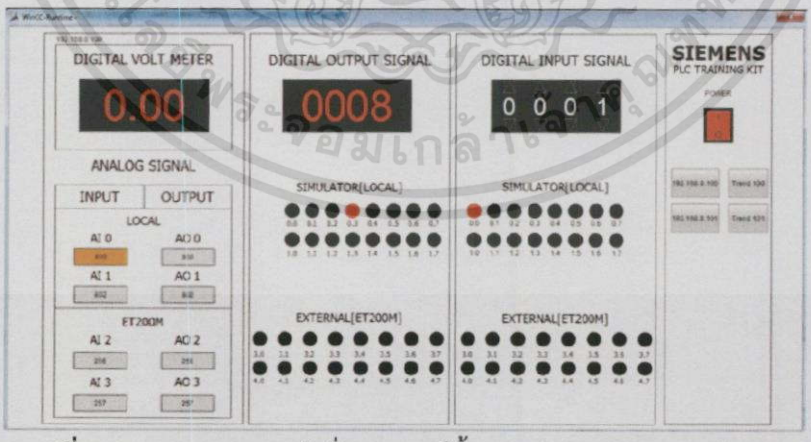
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 2 ณ แผงควบคุมจำลอง

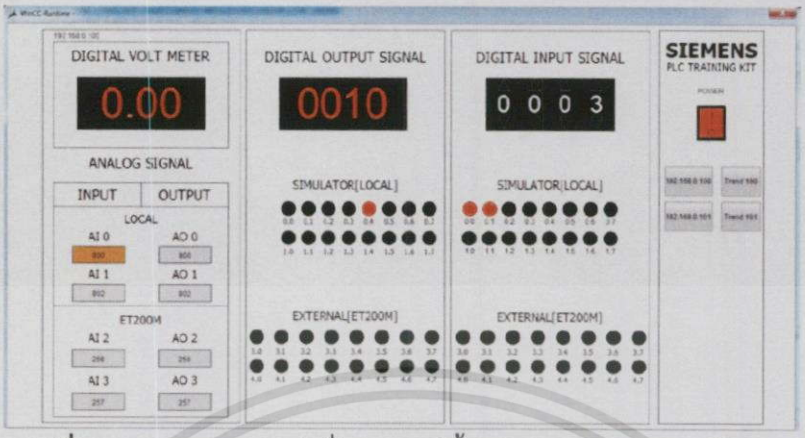


รูปที่ 4.22 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 3 ณ แผงควบคุมจำลอง

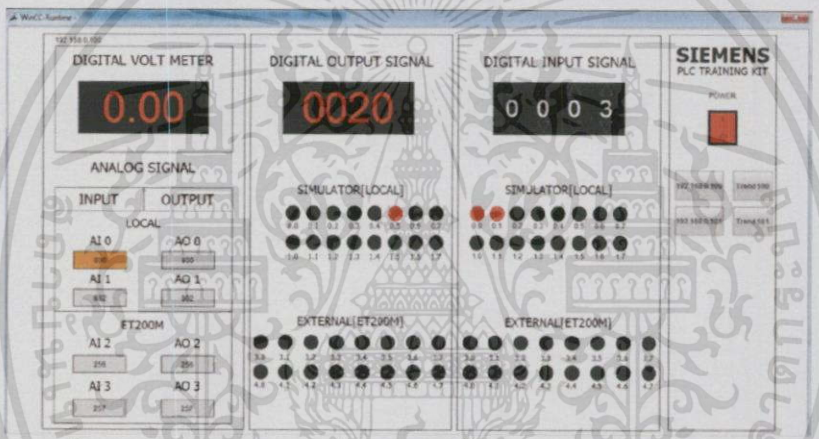


รูปที่ 4.22 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 4 ณ แผงควบคุมจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 5 ณ แผงควบคุมจำลอง



รูปที่ 4.22 ผลการทดลองคำสั่งแลดเดอร์พื้นฐาน 6 ณ แผงควบคุมจำลอง

4.5 อภิปรายผลการทดลอง

จากการทดลองทั้งหมดที่ผ่านมาข้างต้น แสดงให้เห็นว่ากราฟิกแผงควบคุมจำลองสามารถใช้งานได้จริงตามที่ต้องการ แต่ทว่ามีปัญหาที่ส่วนเอาท์พุทดิจิตอลโวลต์มิเตอร์ที่ซึ่งไม่สามารถแสดงผลได้ตรงตามแผงควบคุมต้นแบบมีความคลาดเคลื่อนเพียงเล็กน้อย เนื่องจากอุปกรณ์ฮาร์ดแวร์มีช่วงฮีสเตอเรซีสที่เขียนโปรแกรมแสดงค่าได้ยาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

โครงงานนี้เกี่ยวข้องกับแนวคิดทฤษฎีและการปฏิบัติเพื่อมีวัตถุประสงค์ในการเรียนรู้เกี่ยวกับระบบ SCADA โดยจะแบ่งออกเป็น 3 ส่วนที่มีกิจกรรมเป็นเฉพาะเจาะจง เป็นตัวอย่างที่ดีสำหรับการพัฒนาและแก้ไขปัญหาของระบบอัตโนมัติให้ดียิ่งขึ้น ซึ่งเป็นพื้นฐานที่จะช่วยแก้ไขปัญหาหรือพัฒนาในอุตสาหกรรมปัจจุบัน ประกอบไปด้วยการกำหนดค่าฮาร์ดแวร์ การเขียนโปรแกรมสำหรับควบคุม การเขียนโปรแกรมกราฟิก และการจำลองระบบ SCADA โดยในโครงงานนี้ได้ทำการทดสอบเขียนกราฟิกเป็นแผงควบคุมจำลอง มีต้นแบบมาจากแผงควบคุมที่เชื่อมต่ออยู่กับพีแอลซี ซึ่งทดลองโดยการสับสวิทช์ที่แผงควบคุมต้นแบบ เขียนโปรแกรมย้ายค่าอินพุทเอาท์เป็นทั้งดิจิตอลและอนาล็อก และเขียนโปรแกรมแลดเดอร์พื้นฐานเพื่อทดสอบว่าสถานะ ณ แผงควบคุมจำลองนั้นมีสถานะที่เหมือนกันกับแผงควบคุมต้นแบบ ซึ่งจากผลการทดลองแสดงให้เห็นว่าแผงควบคุมจำลองนั้นสามารถใช้งานได้จริง แต่อาจจะมีการหน่วงเล็กน้อยเมื่อทำการแสดงค่าออกมา

5.2 ปัญหาของการวิจัยและการทดลอง

ในการทำการทดลองของงานวิจัยนี้มีปัญหาเล็กน้อย เช่น อุปกรณ์แผงควบคุมต้นแบบ โวลต์มิเตอร์ปุ่มหมุนปรับค่ามีค่าไม่ตรงและมีช่วงฮีสเตอรีซีสที่ค่อนข้างมาก ส่งผลให้การทดลองเก็บค่ามีความคลาดเคลื่อนและเมื่อเปิดเครื่องคอมพิวเตอร์หรือโปรแกรมร่วมกับอุปกรณ์ฮาร์ดแวร์นานๆ บางครั้งอาจเกิดความขัดข้องของระบบขึ้น ซึ่งต้องมีการเริ่มเดินเครื่องใหม่ในบางครั้ง

5.3 ข้อเสนอแนะ

แนวคิดและเทคโนโลยีซอฟต์แวร์ที่ใช้ในการประยุกต์ในปัจจุบันสามารถนำกลับมาใช้ในการเขียนการควบคุมหรือกราฟิกที่คล้ายกันได้อย่างอีกมากมาย ซึ่งในโครงงานนี้สามารถนำมาเป็นพื้นฐานเพื่อต่อยอดในการศึกษาหรือปฏิบัติได้ ยกตัวอย่างเช่น เชื่อมต่อกับชุดทดลองต่างๆที่มีอยู่ในห้องแลปเพื่อประสิทธิภาพที่สูงกว่าและได้เรียนรู้ถึงระบบ SCADA อย่างแท้จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] Veranuch (ม.ป.ป.). **SCADA System**. เข้าถึงได้จาก : http://riverplusblog.com/author/veranuch/ (วันที่ค้นข้อมูล: 17 ธันวาคม 2557).

[2] EDA International Ltd. (ม.ป.ป.). **SCADA คืออะไร**. เข้าถึงได้จาก : http://www.eda.co.th/scada.html (วันที่ค้นข้อมูล: 17 ธันวาคม 2557).

[3] **โครงสร้างพื้นฐานพีแอลซี**. (ม.ป.ป.). เข้าถึงได้จาก : https://app.enit.kku.ac.th/mis/administrator/doc_upload/20140313110117.pdf (วันที่ค้นข้อมูล: 17 ธันวาคม 2557).

[4] ExpertAutomation. (2553). **Ole for Process Control (OPC)**. เข้าถึงได้จาก : http://archive.wunjun.com/bigdaddy/3/481.html (วันที่ค้นข้อมูล 20 ธันวาคม 2557)

[5] Kampol. (ม.ป.ป.). **สิ่งที่ควรรู้ก่อนเริ่มเล่น Visual Basic**. เข้าถึงได้จาก : http://kampol.htc.ac.th/web1/subject/programming2/sheet/vb6/vbch01.html (วันที่ค้นข้อมูล 4 มีนาคม 2558)

[6] SIMATIC Technical Support. (2007). **SIMATIC S7-300 Getting Started for First Time Users Manual** ; Siemens AG.

[7] SIMATIC Technical Support. (2008). **SIMATIC WinCC V7.0 Getting Started Manual** ; Siemens AG.

[8] Kepware Technical Support. (2013). **KEPServerEX V5 Help**; Kepware Technology.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

การสร้างโปรเจคและกำหนดฮาร์ดแวร์ของโปรแกรม Simatic Step 7

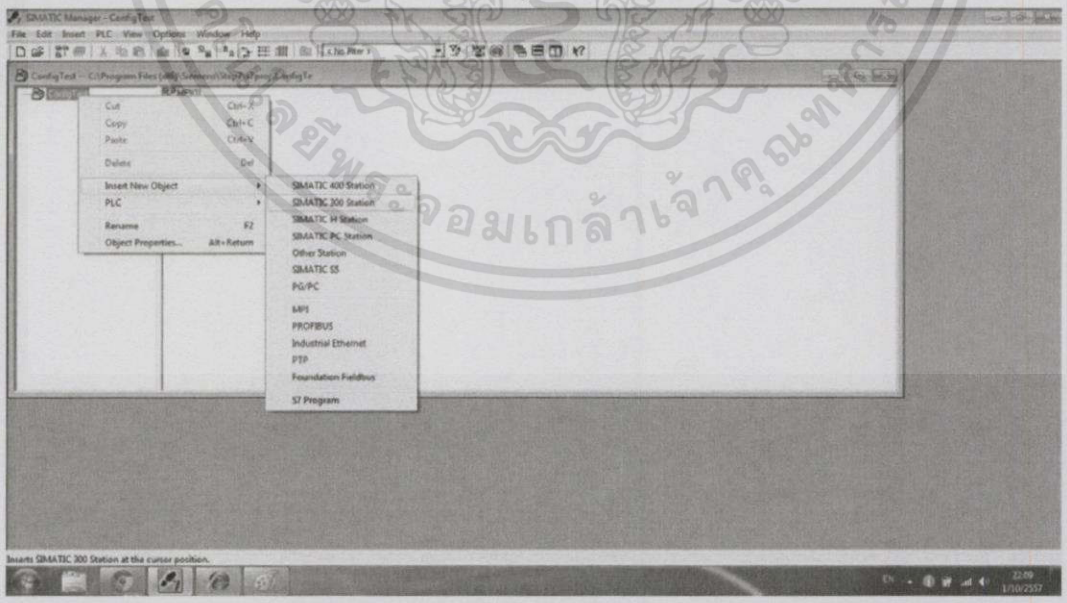
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างโปรเจคและกำหนดฮาร์ดแวร์ของโปรแกรม Simatic Step 7

1. เปิดโปรแกรม Simatic Manager Step 7

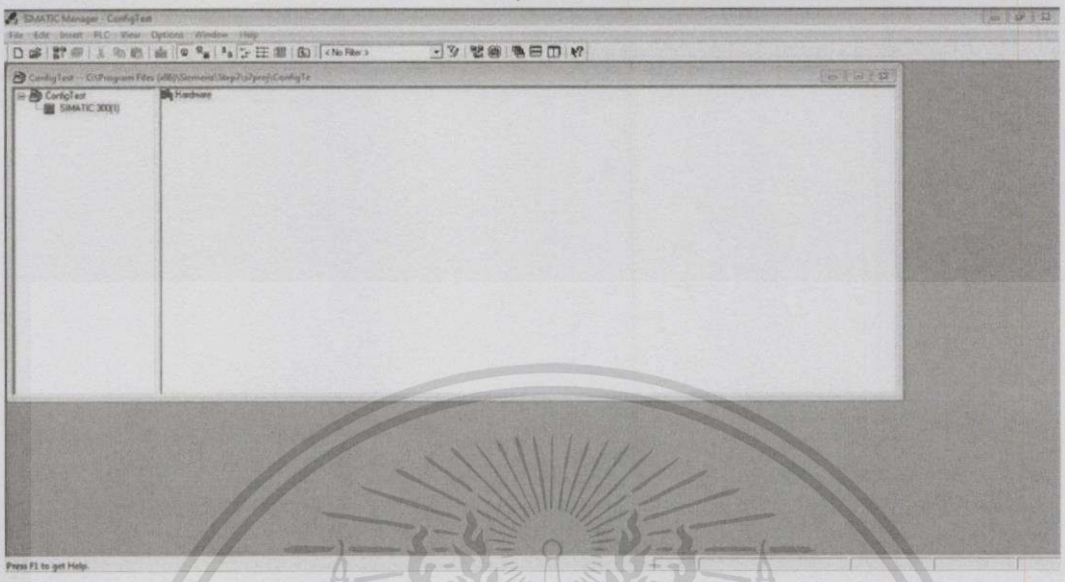
2. File --> New Project --> ตั้งค่าโปรเจค --> OK

3. คลิกขวาโปรเจคที่สร้าง --> Insert New Object --> SIMATIC 300 Station



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เลือก Hardware --> Insert --> Insert Object --> SIMATIC 300 --> Rack-300 --> Rail

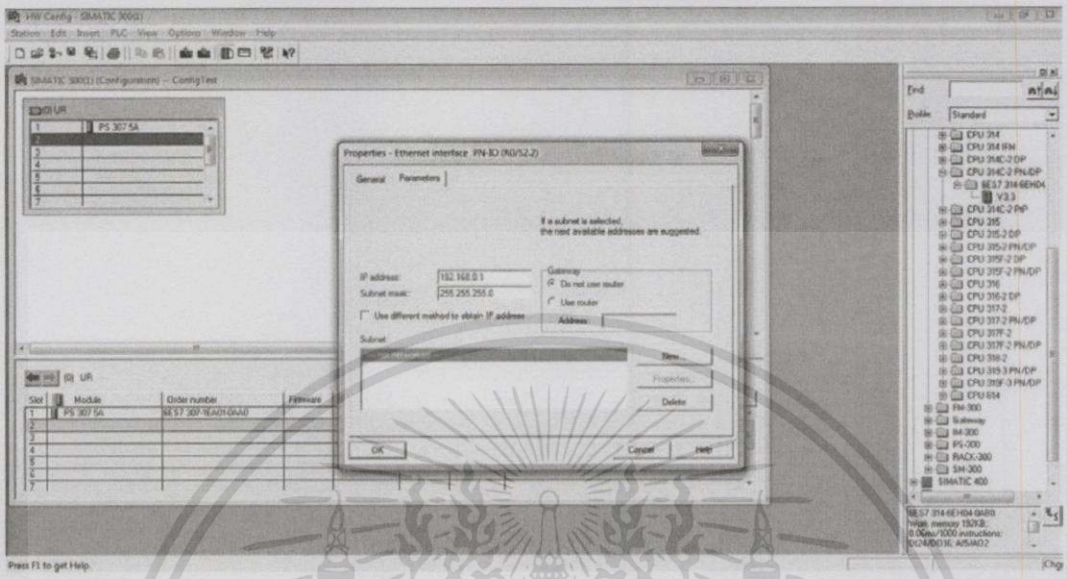


5. คลิกที่ Slot 1 Rack 0 --> Insert --> Insert Object --> PS300 --> PS307/5A

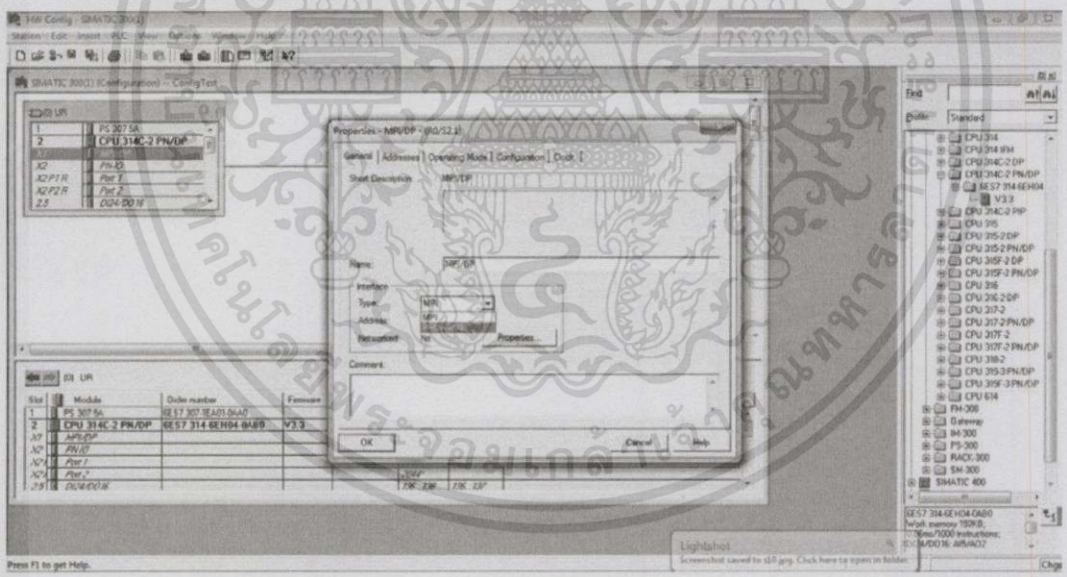
6. คลิกที่ Slot 2 Rack 0 --> Insert Object --> CPU314C-2PN/DP --> V3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ตั้งค่าใน Ethernet Interface --> New --> Ethernet --> OK --> OK
   (IP Address : 192.168.0.1 และ Subnet mark 255.255.255.0)



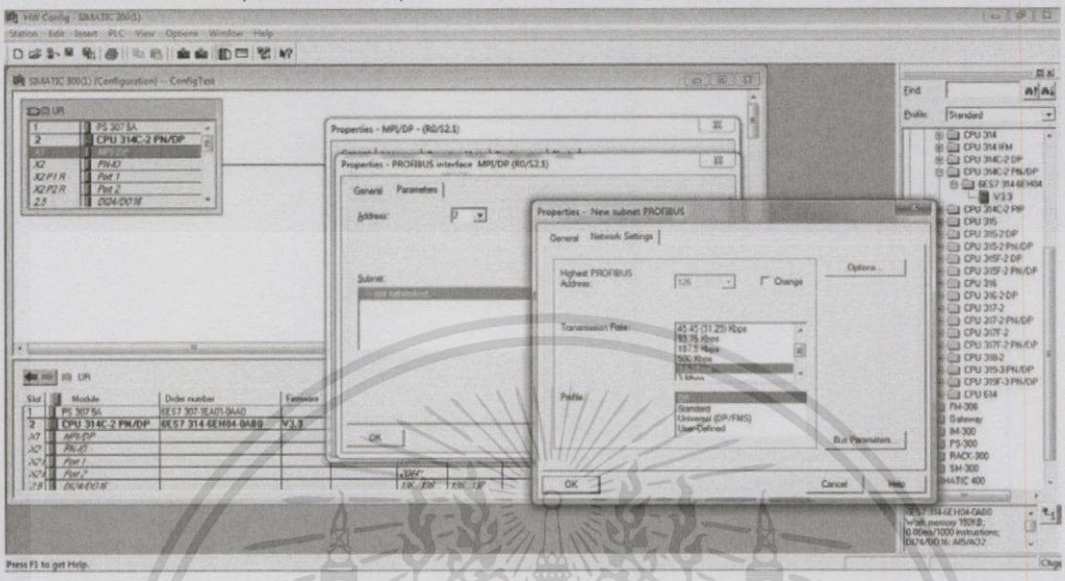
8. ดับเบิ้ลคลิกจากนั้นในหน้าต่าง Interface เปลี่ยนจาก MPI เป็น Profibus



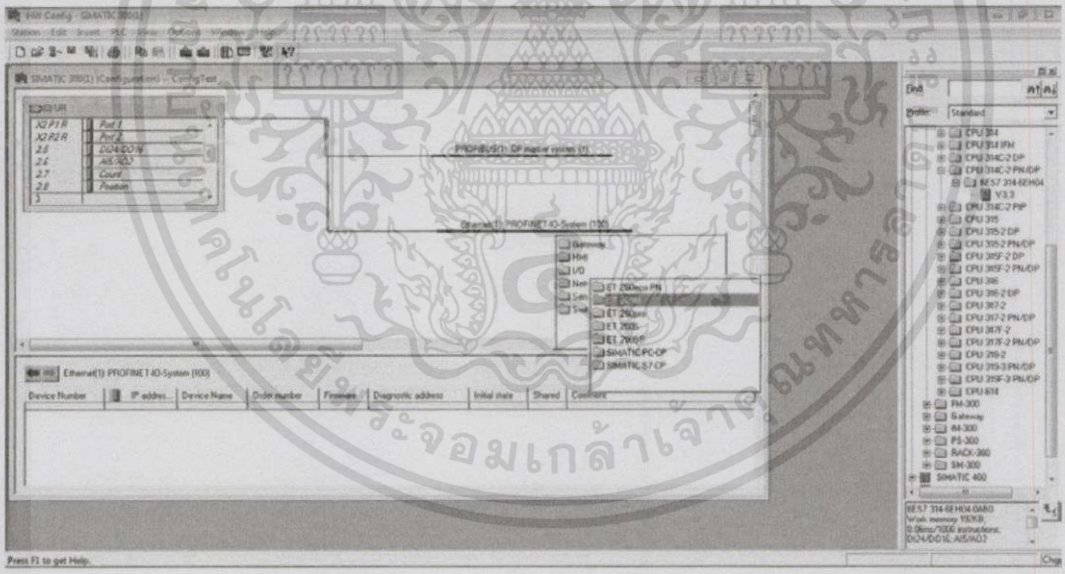
9. เปิด MPI/DP --> Properties --> Operation Network --> DP Master

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. หน้าต่าง MPI/DP ในส่วนของ Interface --> Properties --> Profibus --> Profibus DP/Address 2/Speed 1.5 Mbps. --> OK



11. คลิกที่สายบัสจากนั้น Insert Object --> ET200M/1M153-1



12. ตั้งค่าในส่วนนี้เป็น Slave No3/Speed 1.5 Mbps.

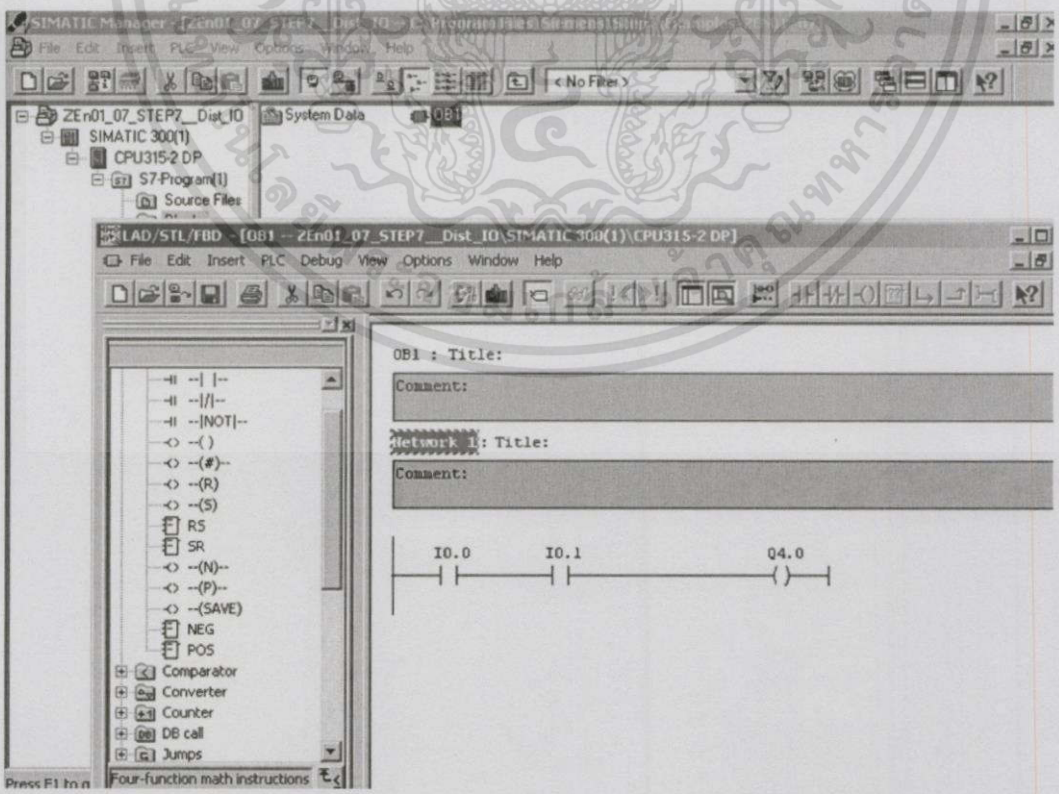
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. Slot 3 เว้นว่างเอาไว้จากนั้นใน Slot อื่นๆ ให้ใส่รายชื่ออุปกรณ์ดังตารางแล้ว Save & Compile และดาวน์โหลดลงไป

	Hardware Module	*Model*
1.	Power Supply PS307/5A	307-1EA01-0AA0
2.	SIMATIC S7-300 CPU314C-2PN/DP	314-6EH04-0AB0
3.	Module DI18*24VDC , AI5/AO2*12 Bits	Include CPU.
4.	Module DI16/DO16*24VDC	Include CPU.
5.	Module Profibus DP Slave (ET200M)	153-1AA03-0XB0
6.	Module DO16*24VDC/0.5A	322-1BH01-0AA0
7.	Module DI16*24VDC	321-1BH02-0AA0
8.	Module AO4*12 Bits	332-5HD02-0AB0
9.	Module AI8*12 Bits	331-7KF02-0AB0

14. ทำการกำหนดฮาร์ดแวร์ใหม่โดยมีขั้นตอนเหมือนเดิม ยกเว้นในข้อที่ 7 ให้ใส่ไอพีแอดเดรสที่อยู่ในวงเครือข่ายเดียวกันแต่ไม่ซ้ำกัน

15. หากต้องการเขียนโปรแกรมให้คลิกที่ SIMATIC Manager --> SIMATIC 300(1) --> CPU314C-2PN/DP --> S7 Program --> Block --> OB1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

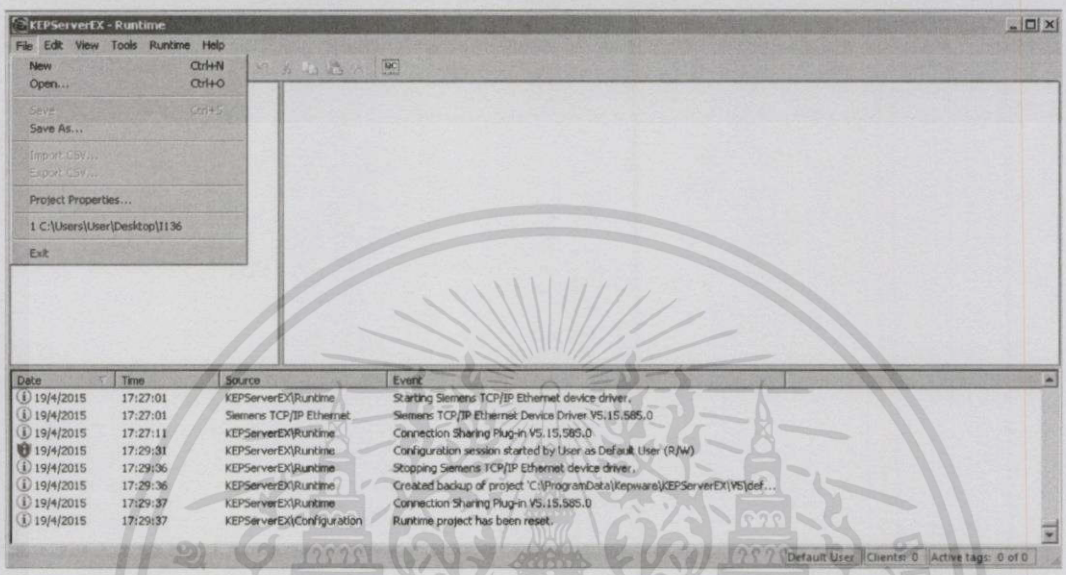
ภาคผนวก ข

การตั้งค่า OPC Server ด้วยโปรแกรม KEPWare KEPServerEx V5 สำหรับ PLC Siemens S7-300 ทั้งสองเครื่อง

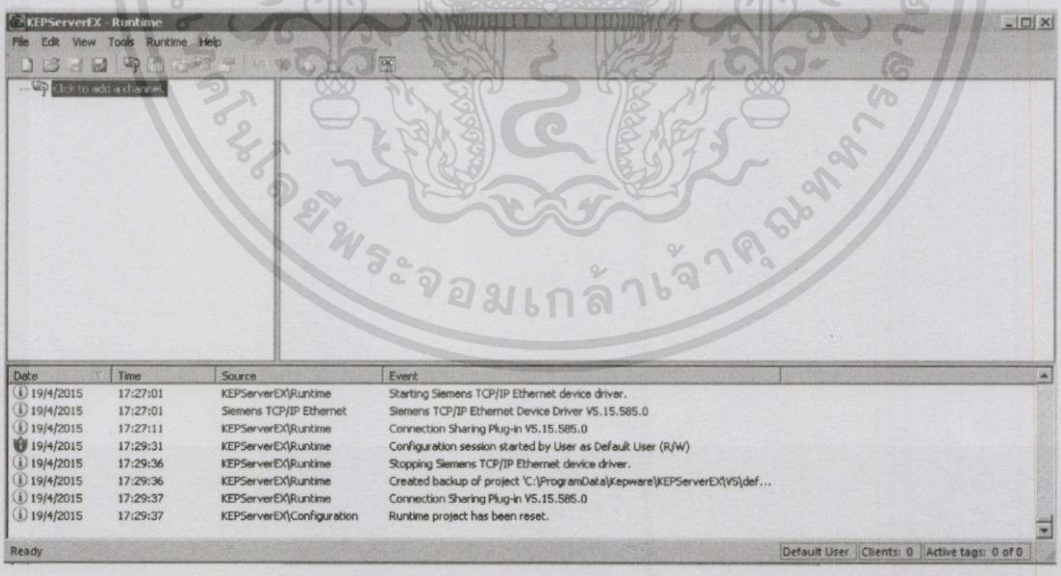
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตั้งค่า OPC Server ด้วยโปรแกรม KEPWare KEPServerEx V5 สำหรับ PLC Siemens S7-300 ทั้งสองเครื่อง

1. เปิดโปรแกรม KEPServerEX V5 --> Menu File --> New

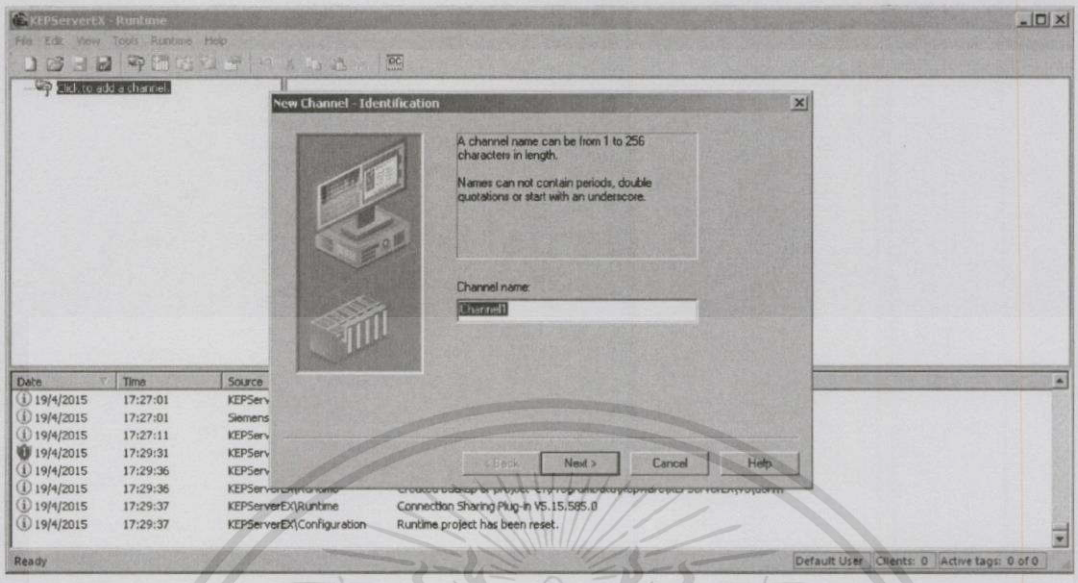


2. คลิกเพิ่มแชแนลที่ Click to add a Channel.

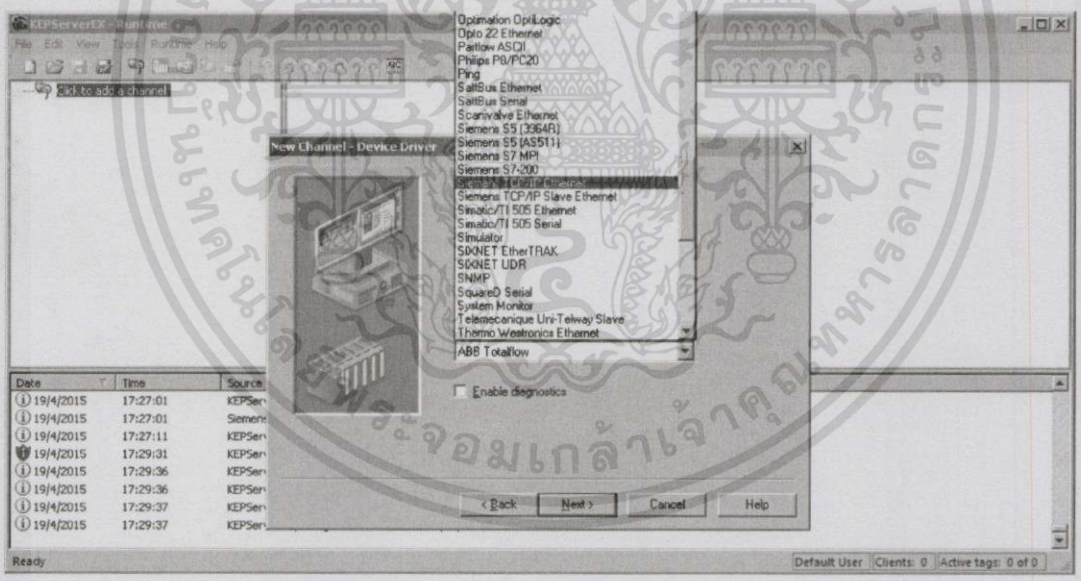


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ตั้งชื่อ Channel name --> Channel 1 --> Next

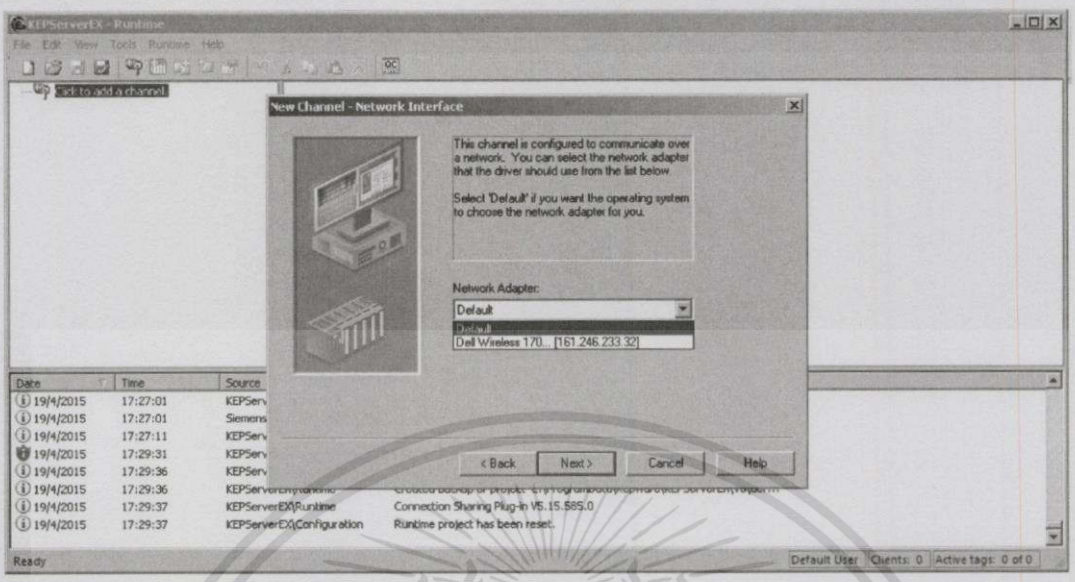


4. เลือก Device Driver เป็น Siemens TCP/IP Ethernet --> Next

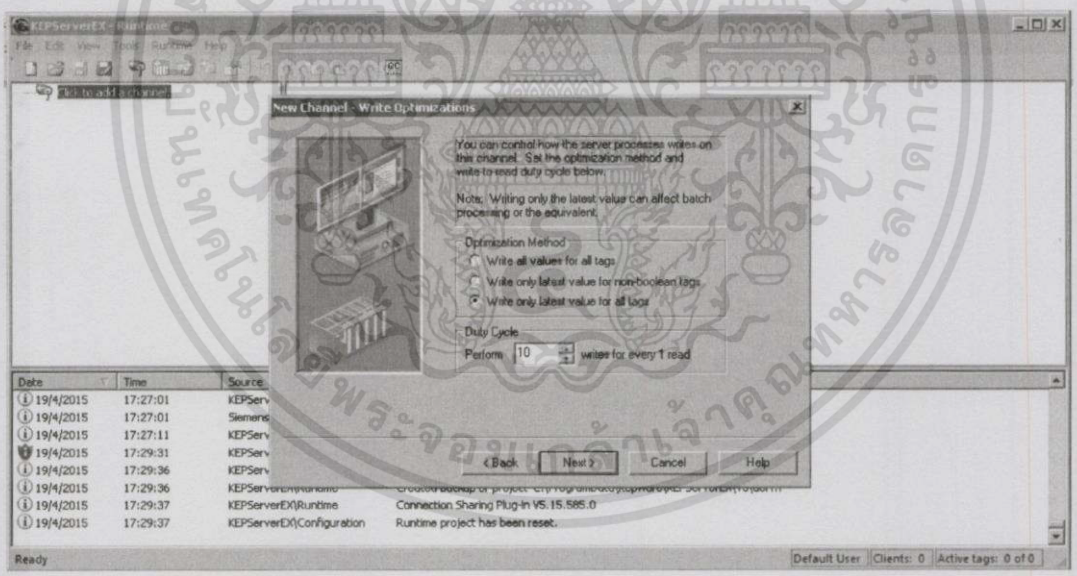


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เลือก Network Adapter ของคอมพิวเตอร์ที่ใช้ --> Next

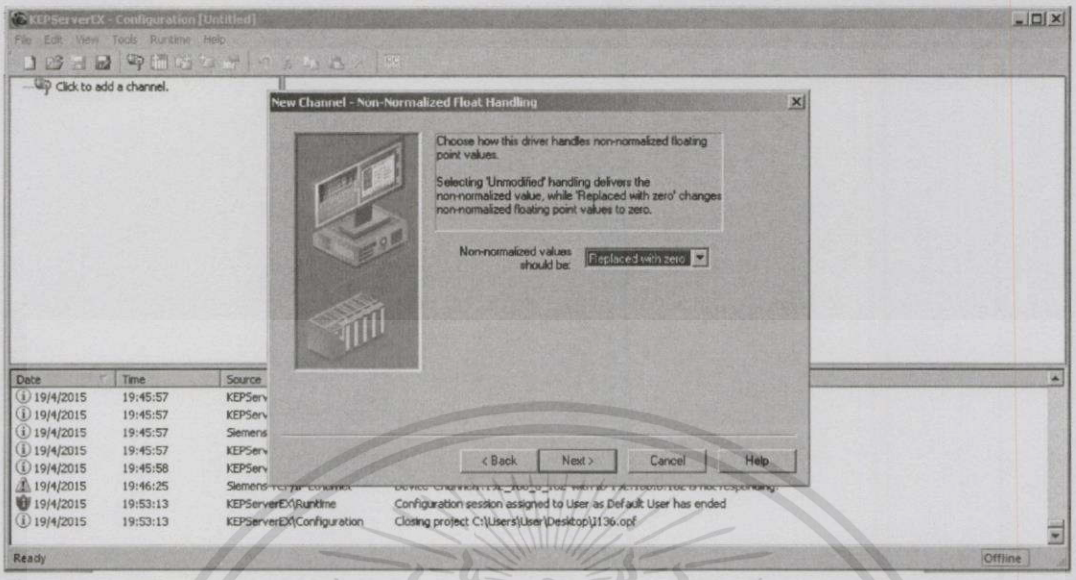


6. Next

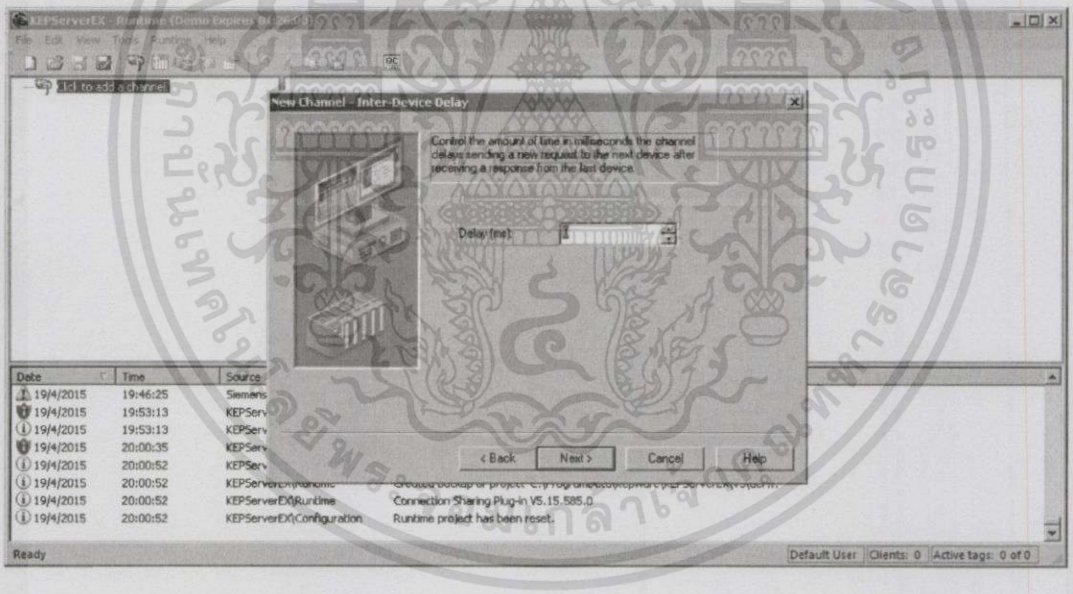


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. Next

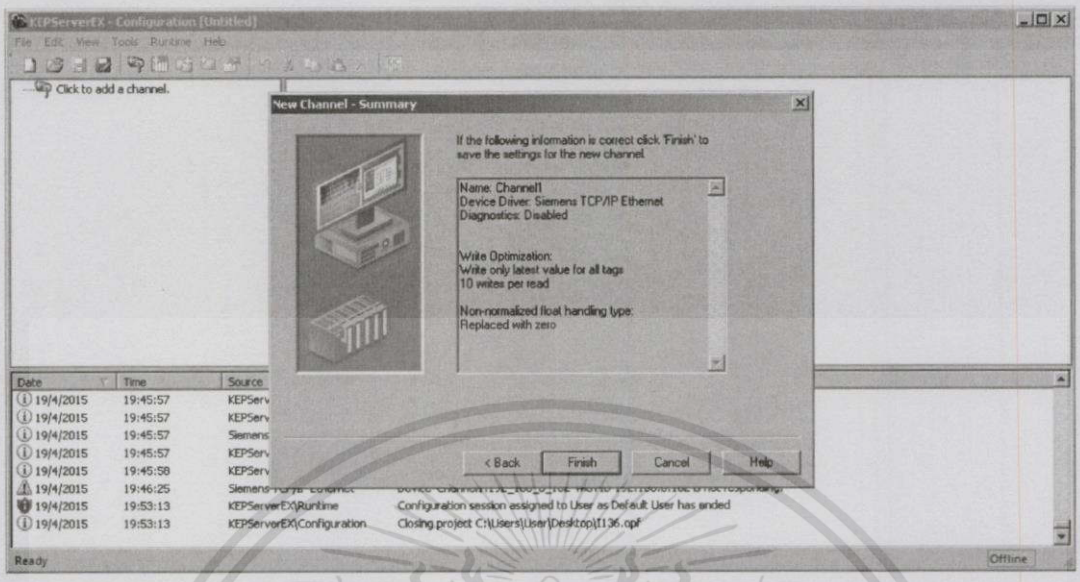


8. Next

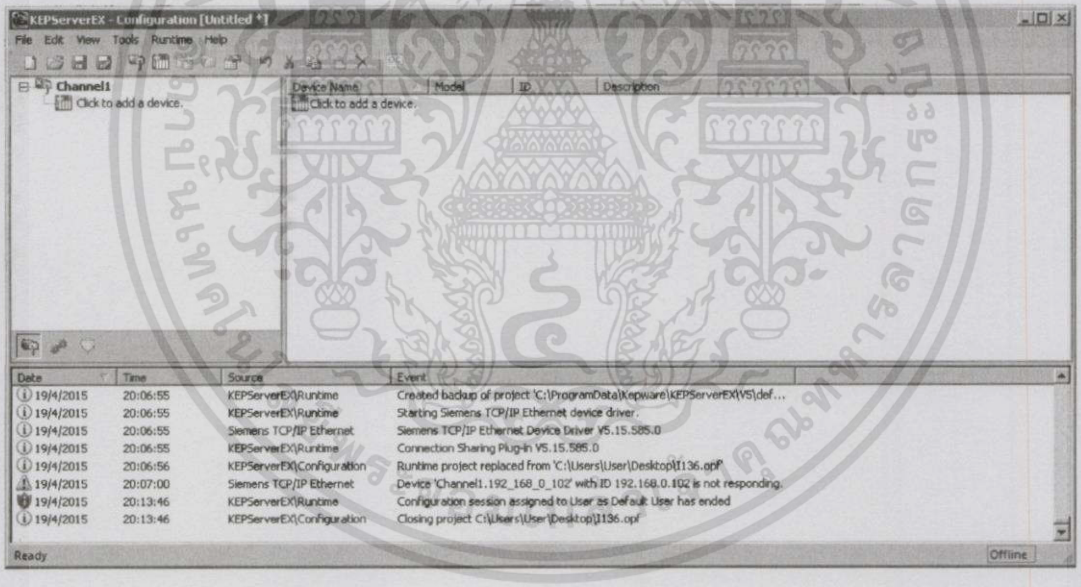


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. Next & Finish

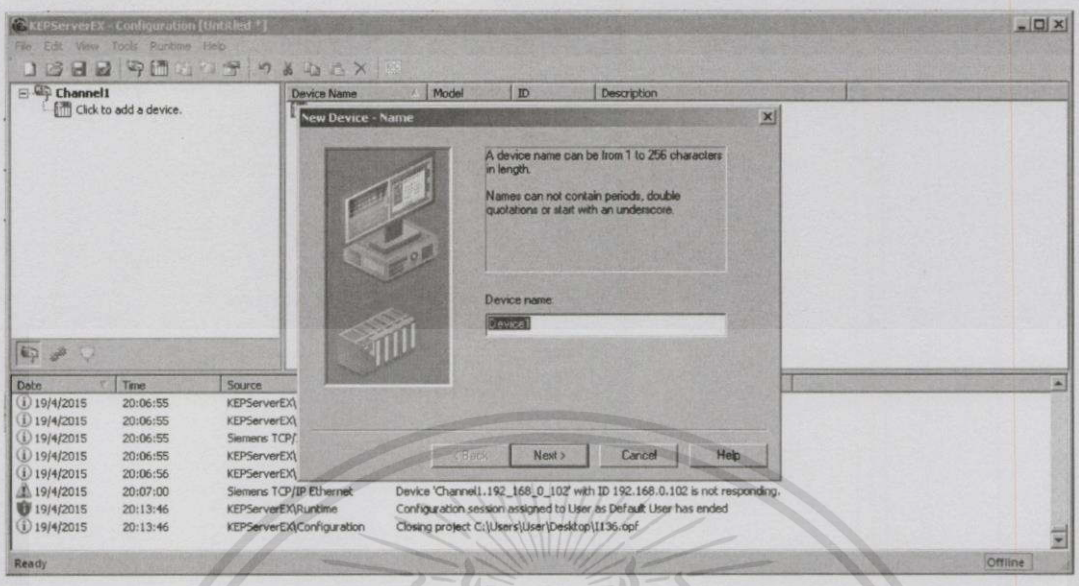


10. คลิกที่ Click to add a device

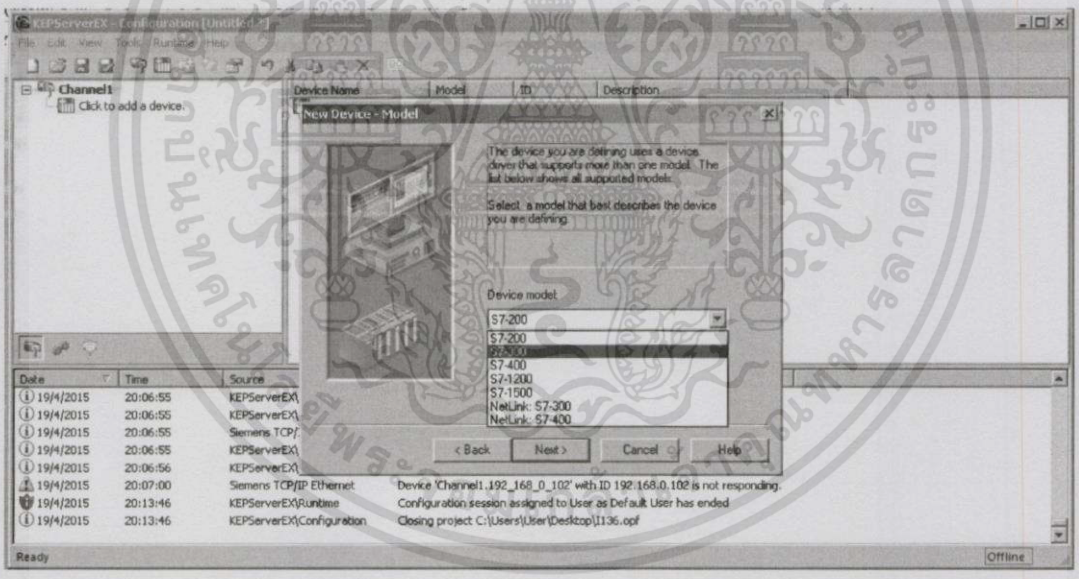


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. ตั้งชื่อ Device name --> Next

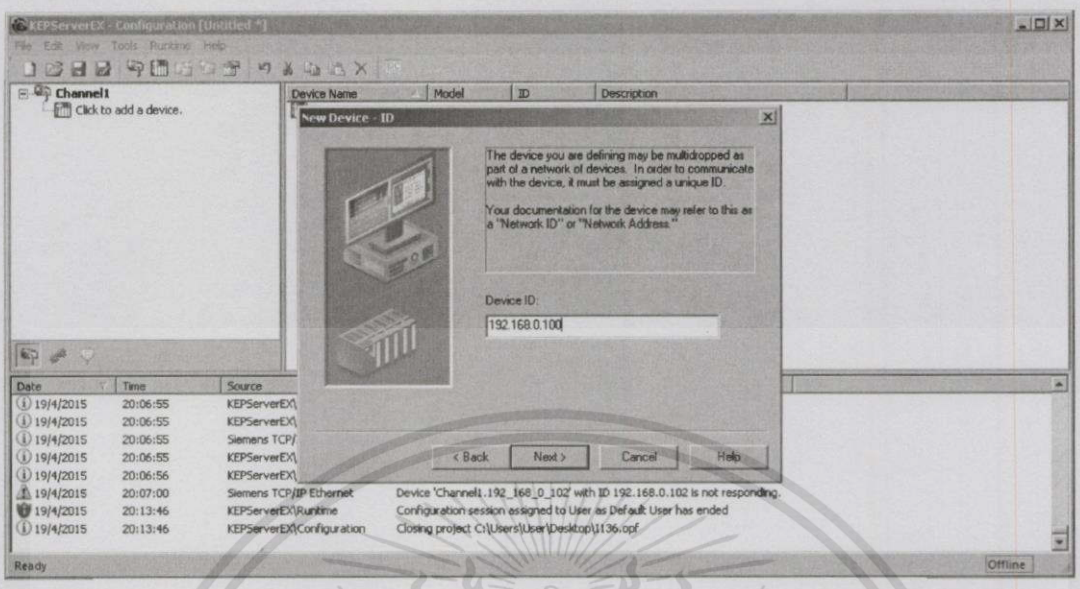


12. เลือก Device Model เป็น S7-300 --> Next

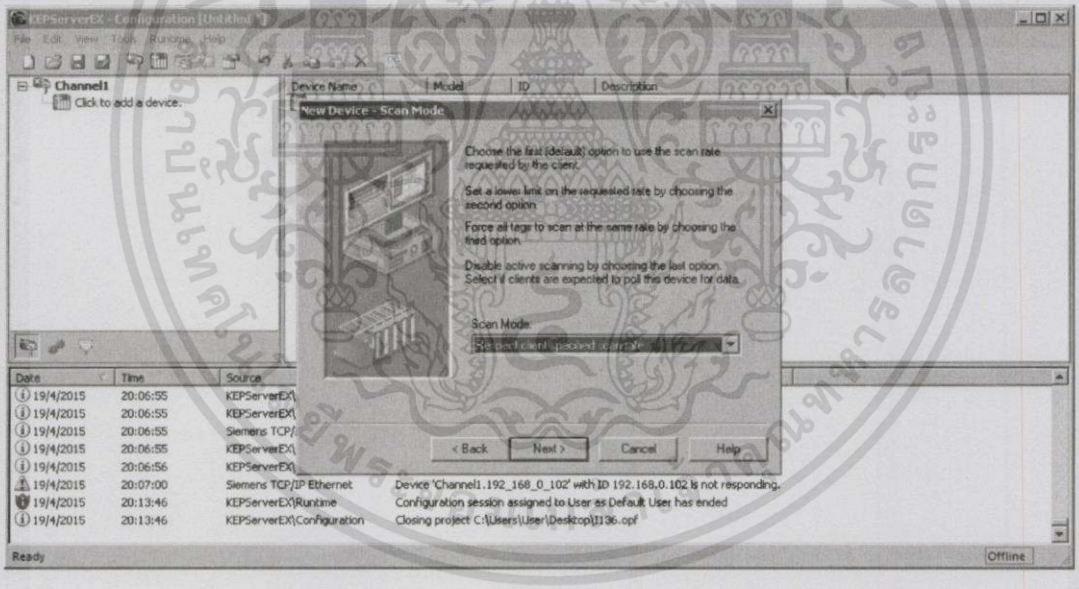


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. ตั้ง Device ID ตามแอดเดรสพีแอลซี --> Next

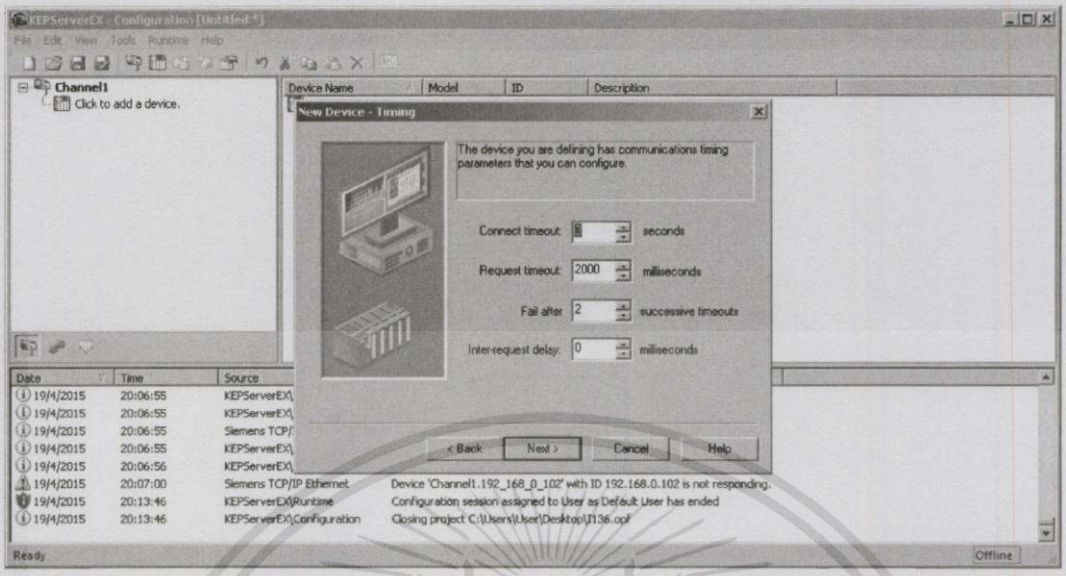


14. Next

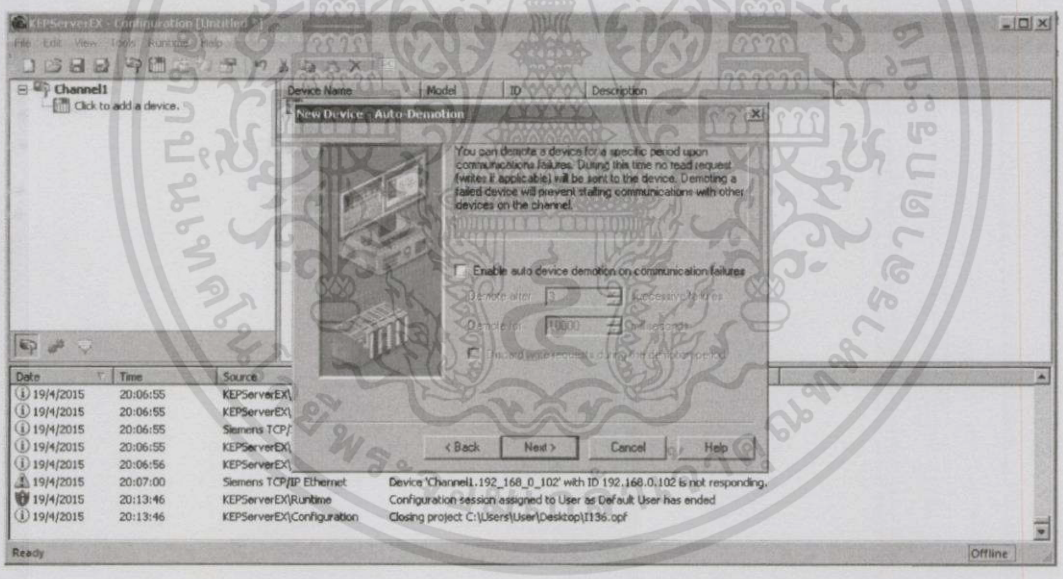


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15. Next

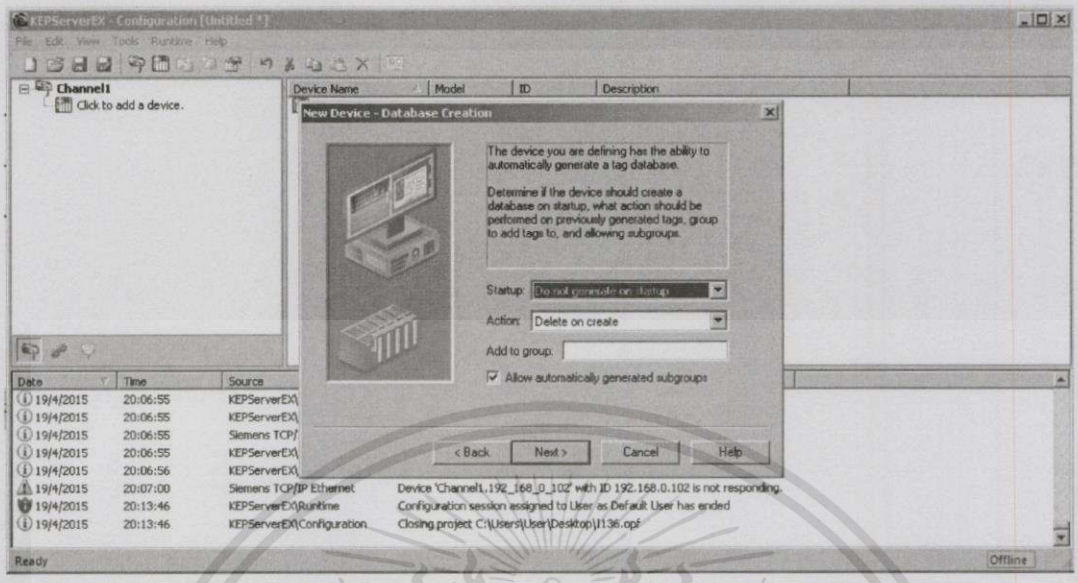


16. Next

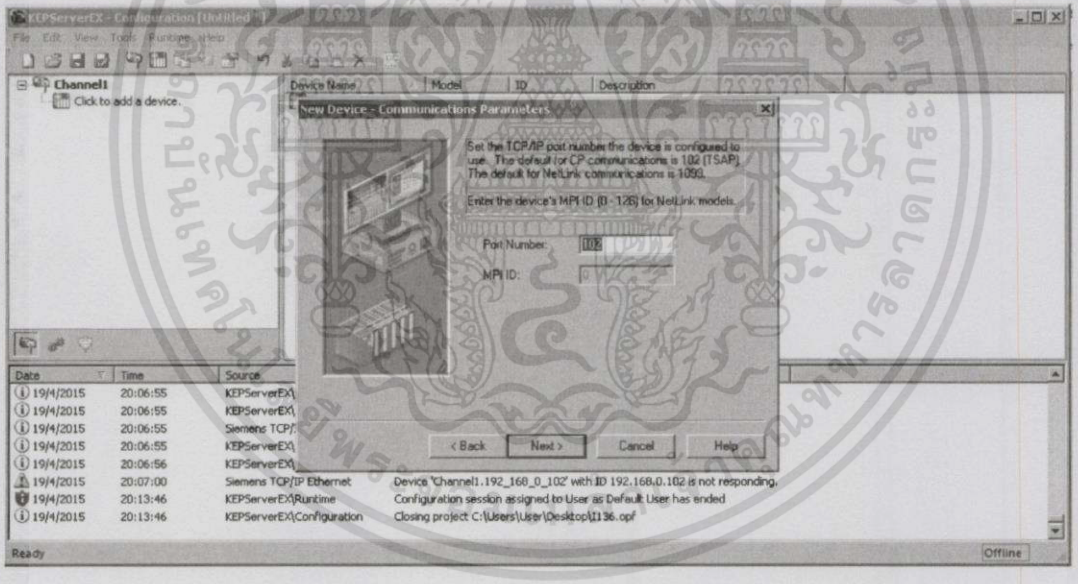


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

17. Next

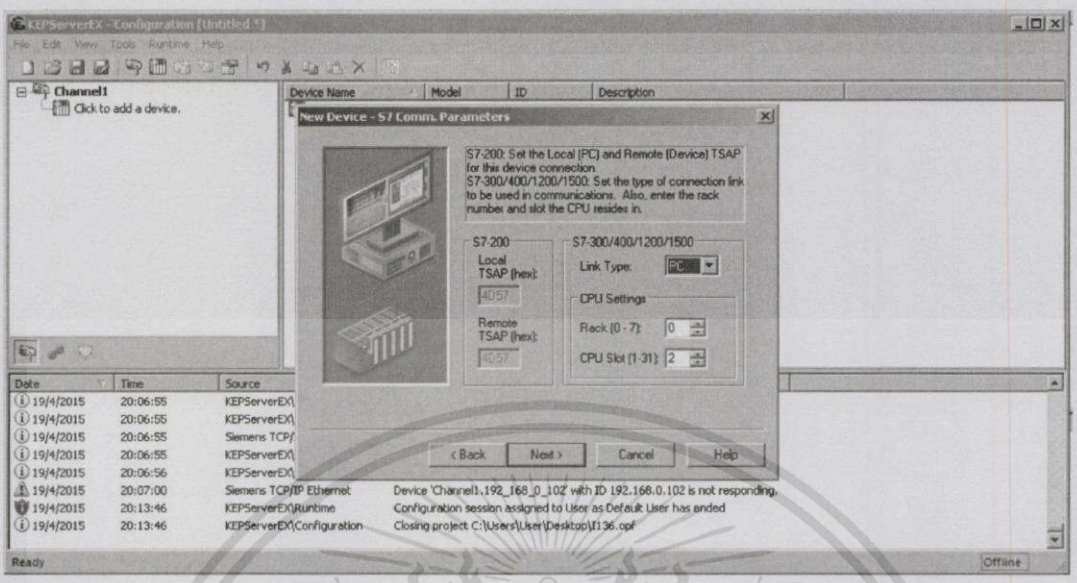


18. Next

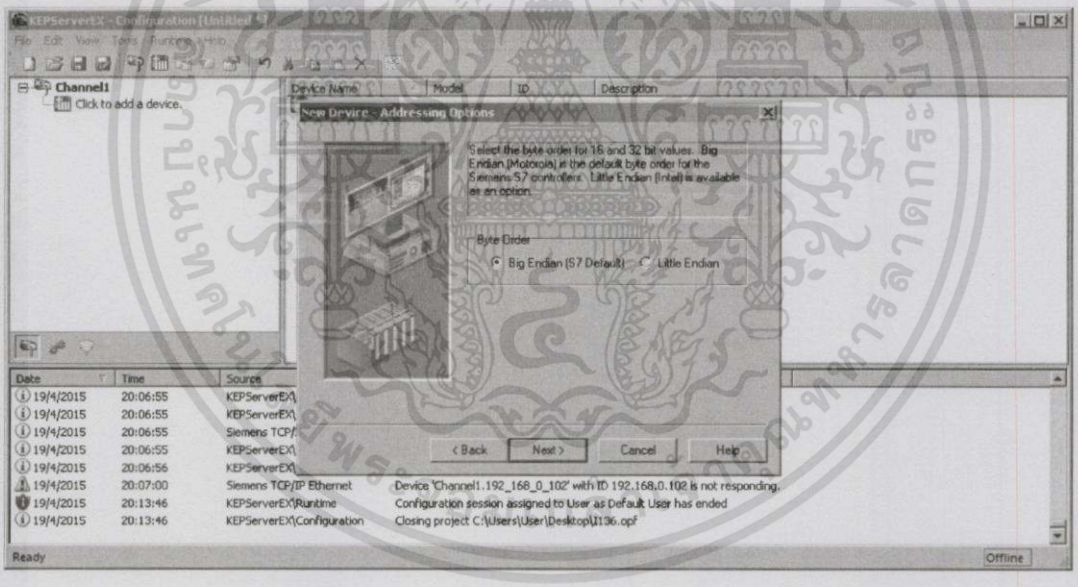


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

19. Next

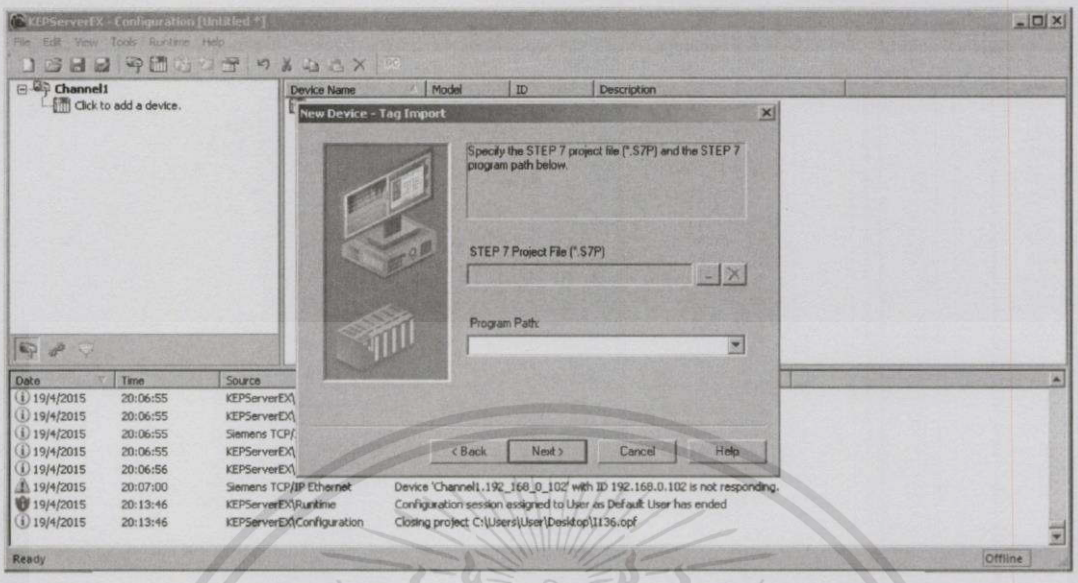


20. Next

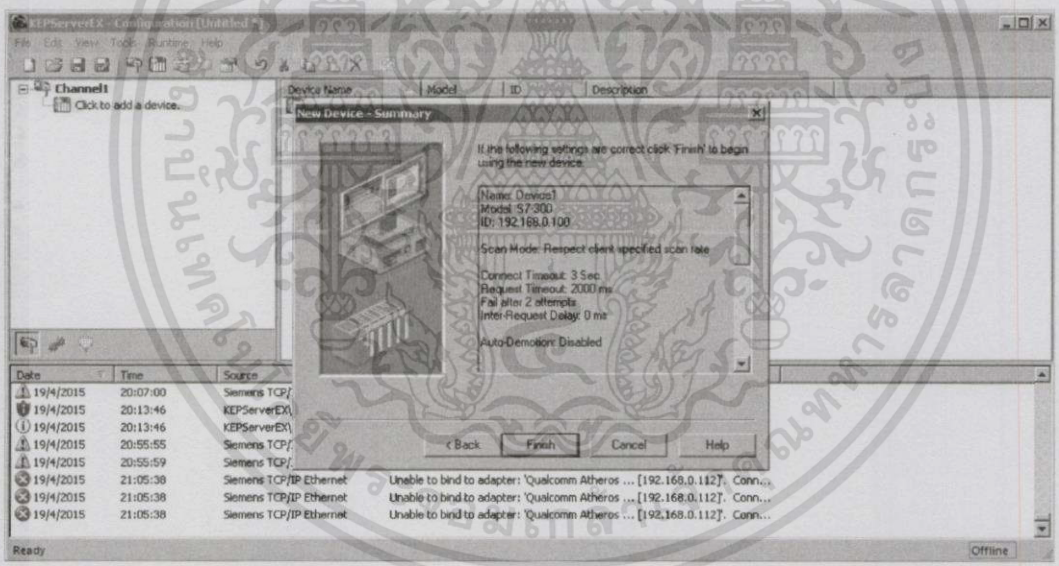


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

21. Next

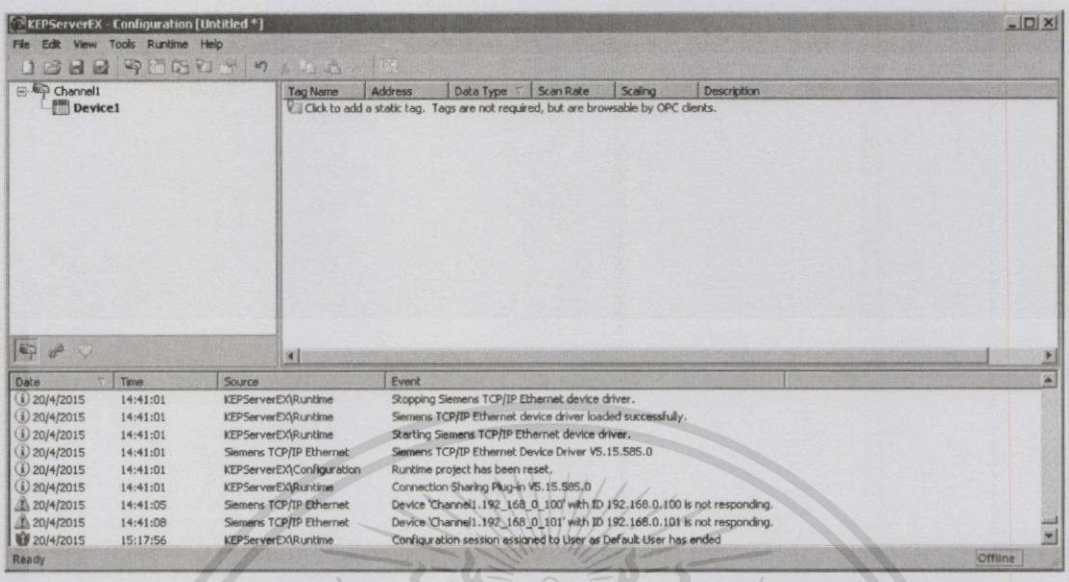


22. Next & Finish

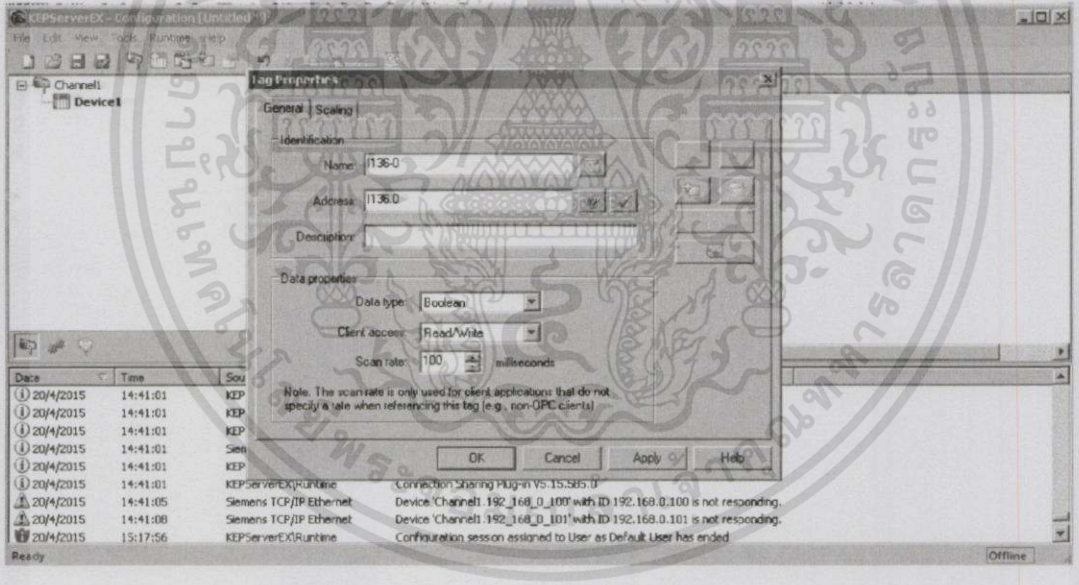


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

23. คลิกที่ Click to add static tag



24. ตั้งค่าต่างๆในหน้าต่าง General ของ Tag Properties



Name : ชื่อของ Tag : I136-0

Address : แอดเดรสที่มีเหมือนกับในโปรแกรม SIMATIC Manager : I136.0

Description : รายละเอียดเพิ่มเติม

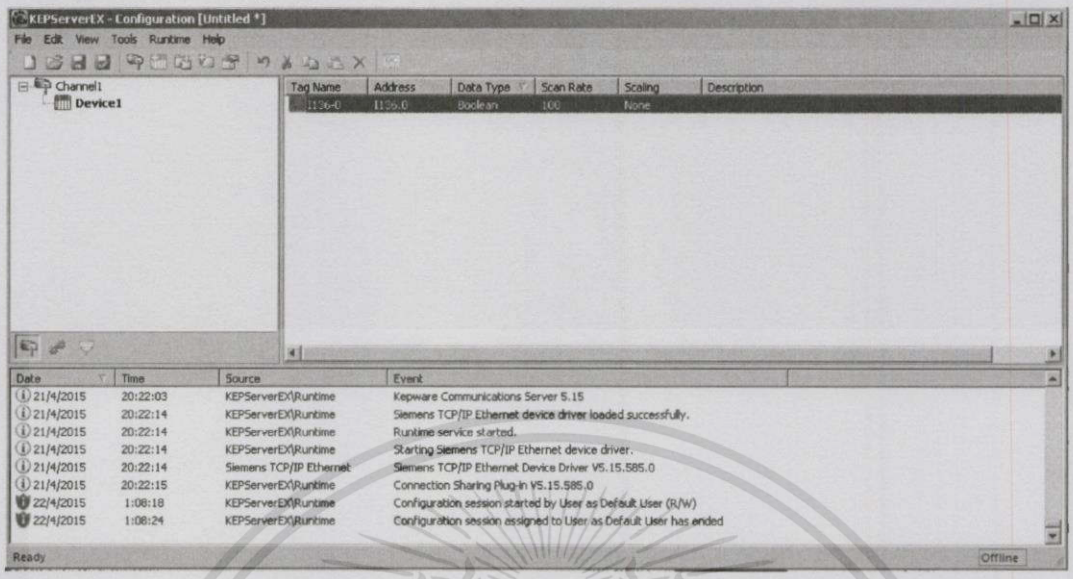
Data type : เลือกประเภทของข้อมูล : Boolean

Client access : เลือกประเภทของการเข้าถึงข้อมูล : Read/Write

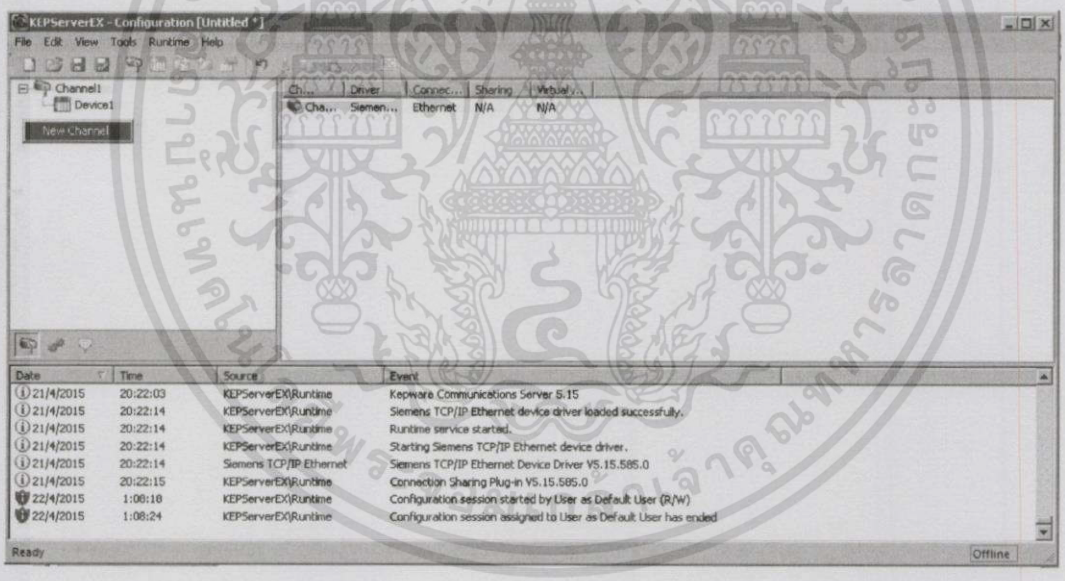
Scan rate : เลือกอัตราความเร็วในการสแกน : 100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

25. Apply & Ok จะได้รูปแบบดังภาพ

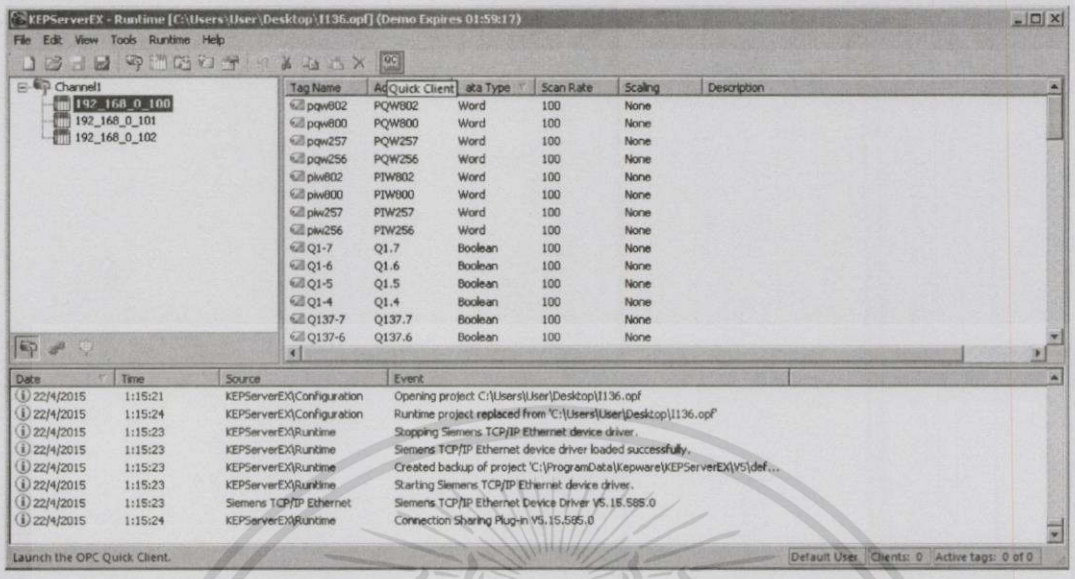


26. หากต้องการสร้างเพิ่มอีกอุปกรณ์ ให้ทำตามขั้นตอนที่ 10-25 อีกครั้ง

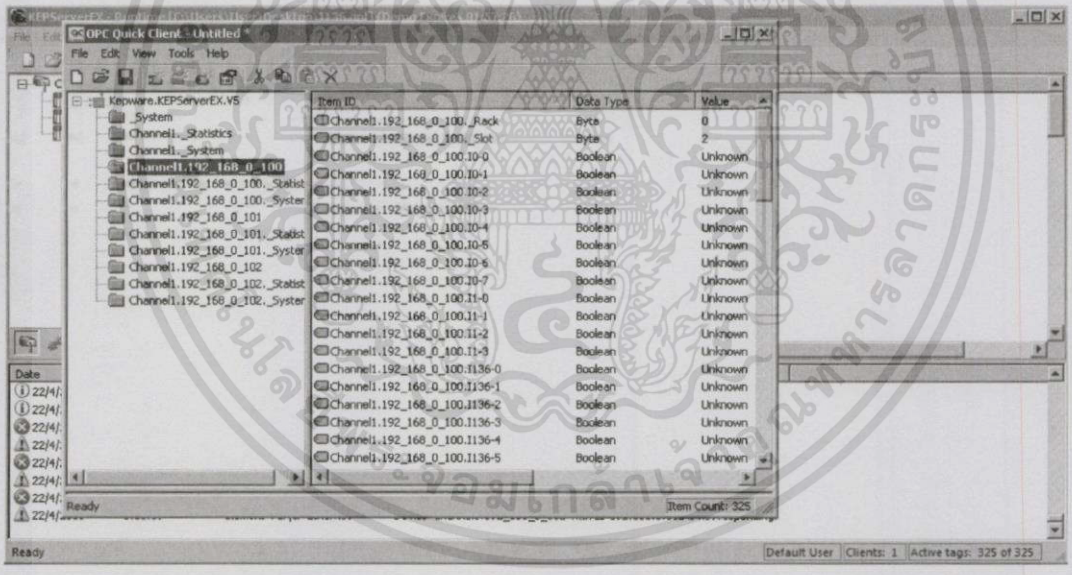


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

27. การมอเตอร์ค่าใน Tag โดยการกด Quick Client ด้านบน



28. หน้าต่างของ Quick Client



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

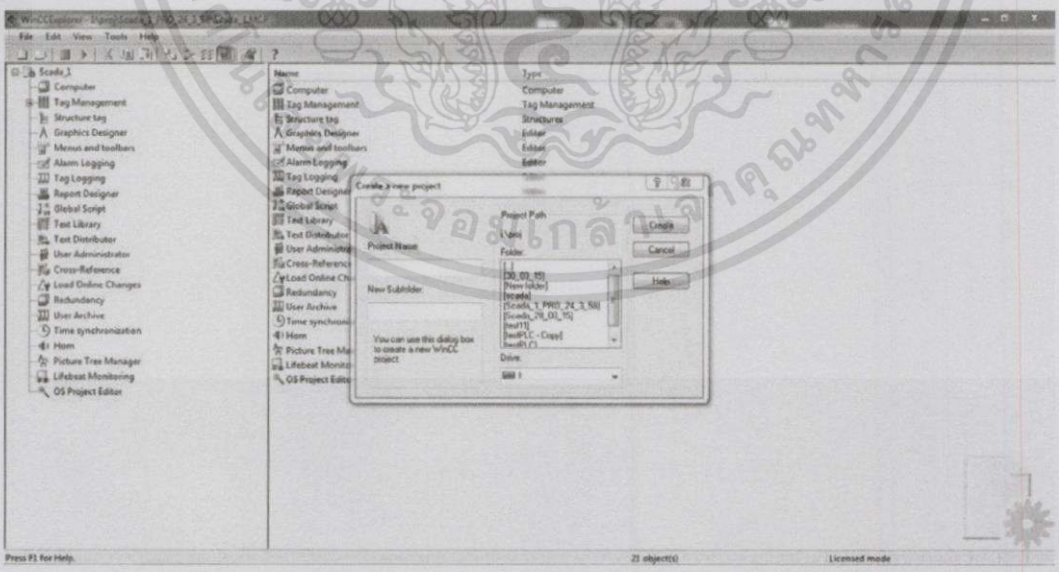
ภาคผนวก ค
การกำหนดและตั้งค่าของโปรแกรม WinCC เพื่อเชื่อมต่อกับ OPC Server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดและตั้งค่าของโปรแกรม WinCC เพื่อเชื่อมต่อกับ OPC Server

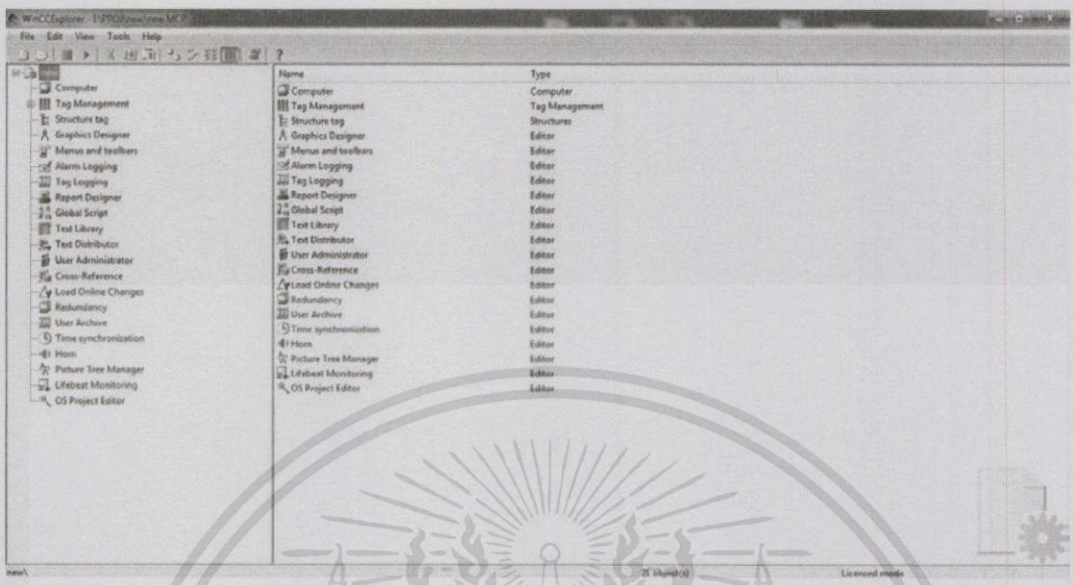
1. สร้างโปรเจคโดยการเลือก Single-User Project --> OK

2. ตั้งชื่อโปรเจคในช่อง Project Name แล้วกด Create



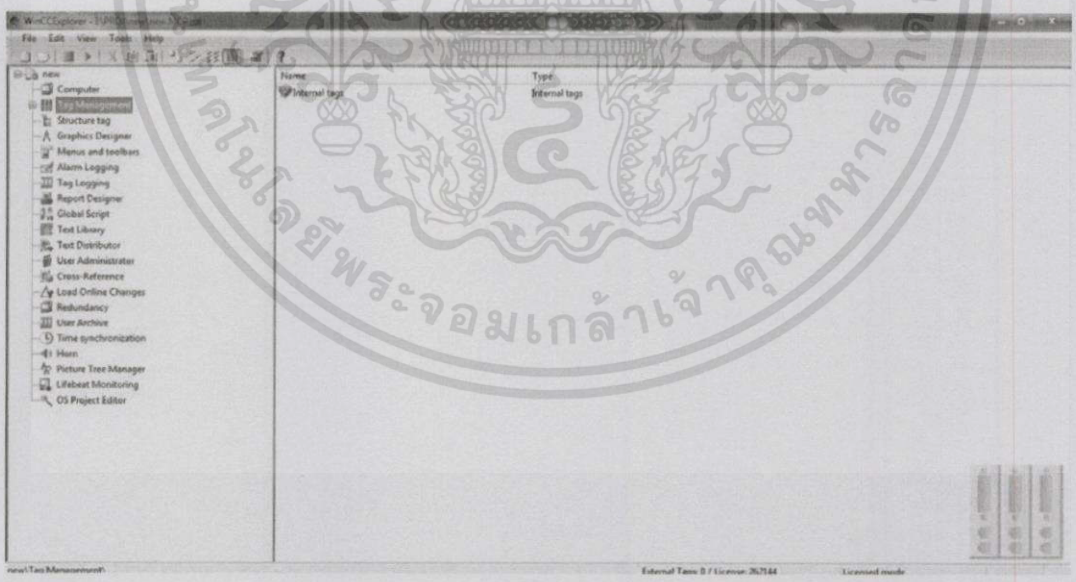
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. จะได้หน้าตาดังรูป



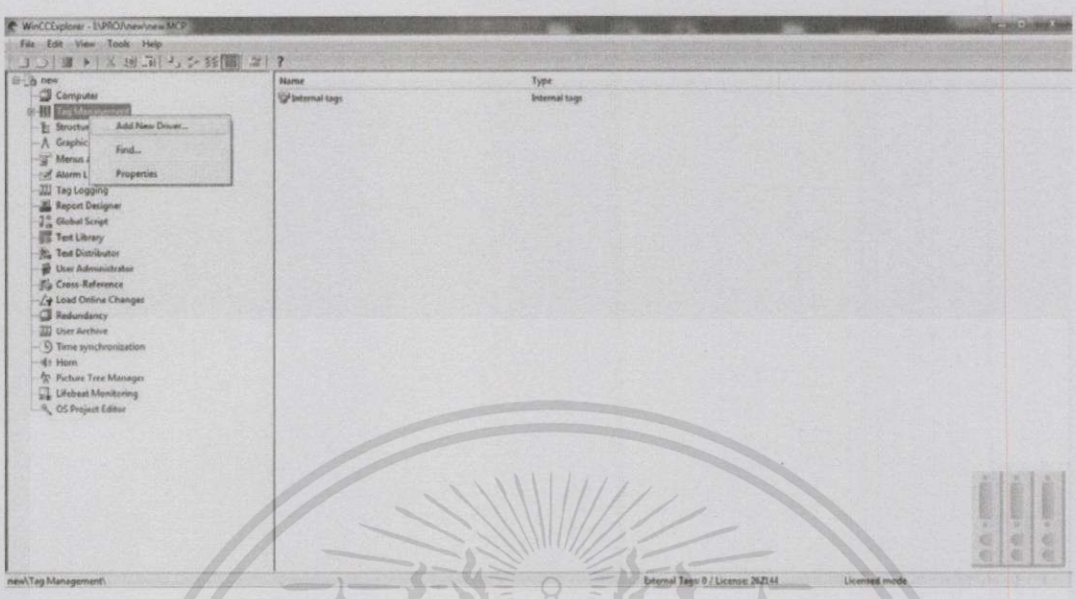
การ add tag

1. เลือกที่ tag manegement

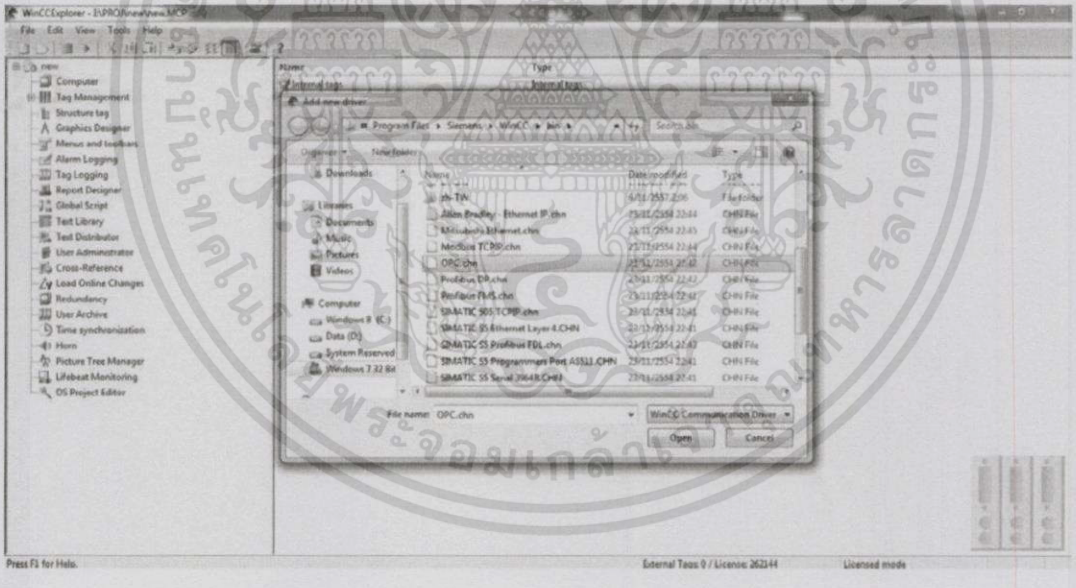


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. คลิกขวาเลือก add new driver

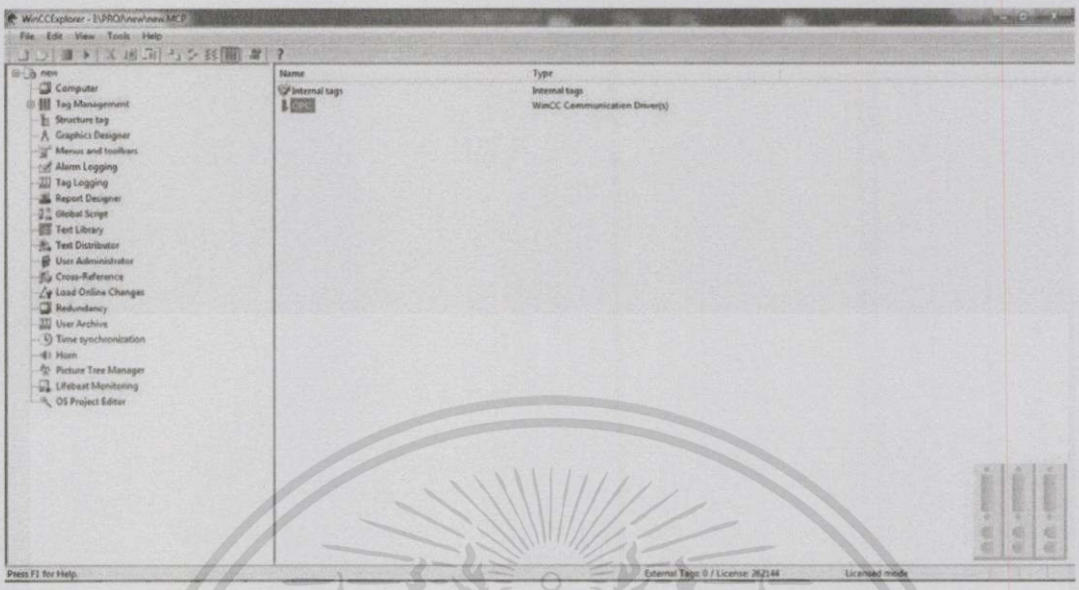


3. เลือกแอ็ด OPC driver โดยเลือก OPC.chn แล้วกด OPEN



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

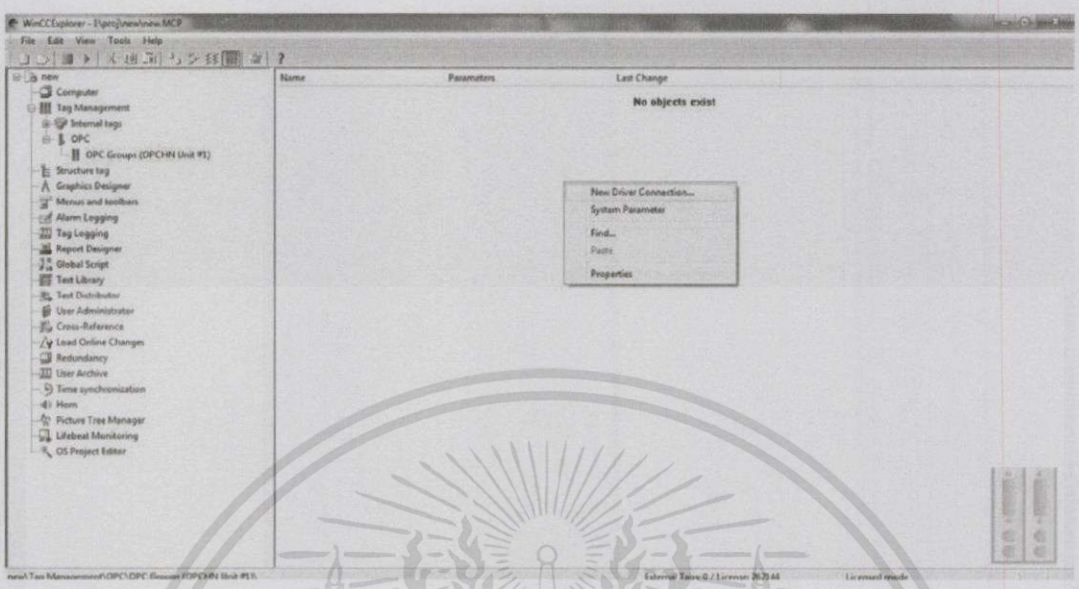
4. จะได้ driver OPC ดังรูป



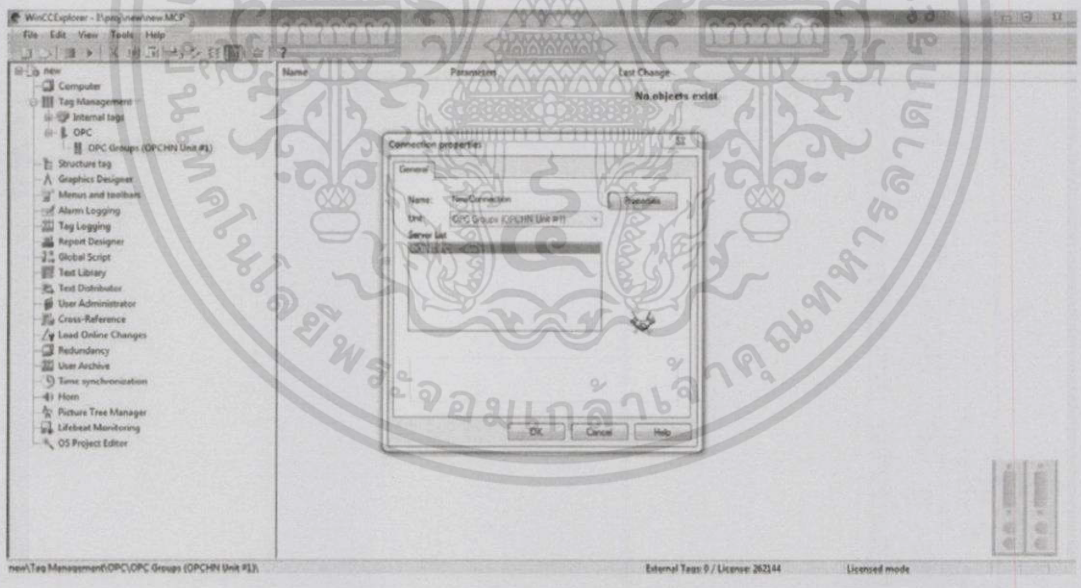
5. คลิกเข้าไปจะได้ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. คลิกขวาแล้วเลือก New Driver Connection

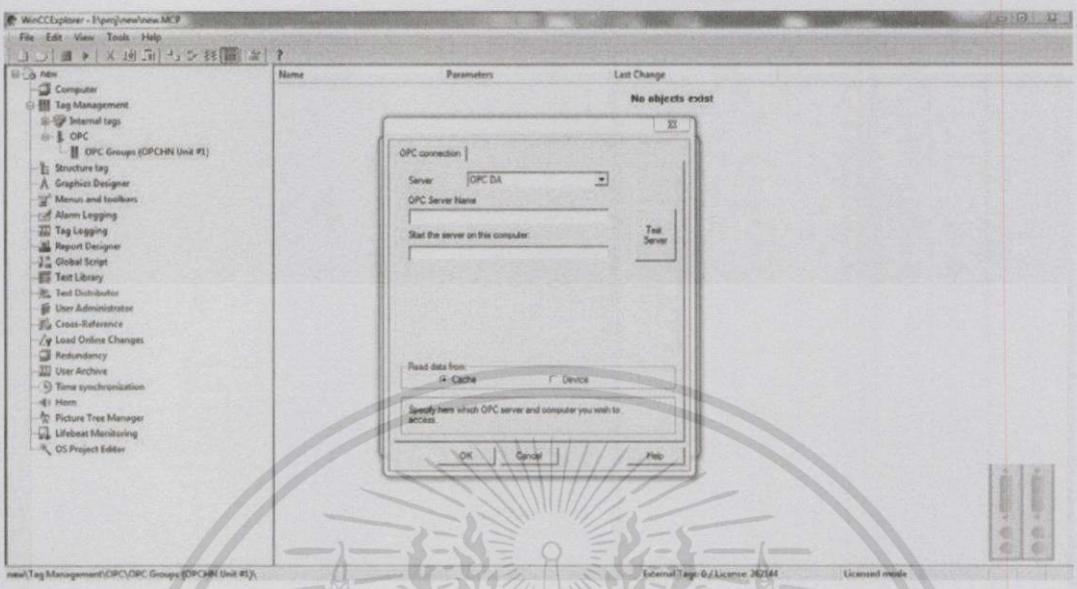


7. ที่หน้าต่างนี้กด Properties

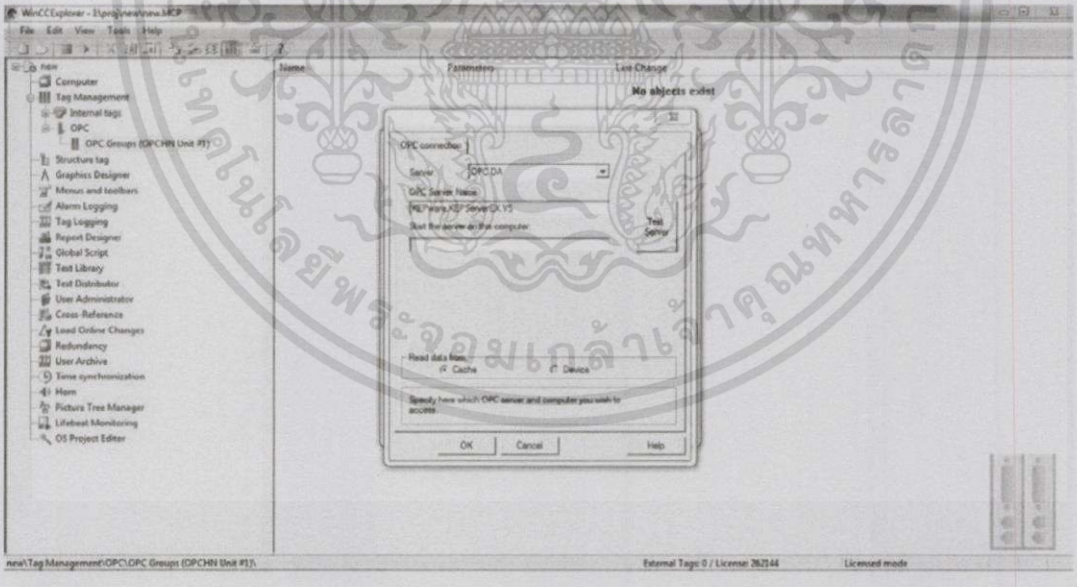


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ที่หน้าต่างนี้ให้ค่า OPC server ไป ในที่นี้คือ” KEPware.KEPServerEX.V5”

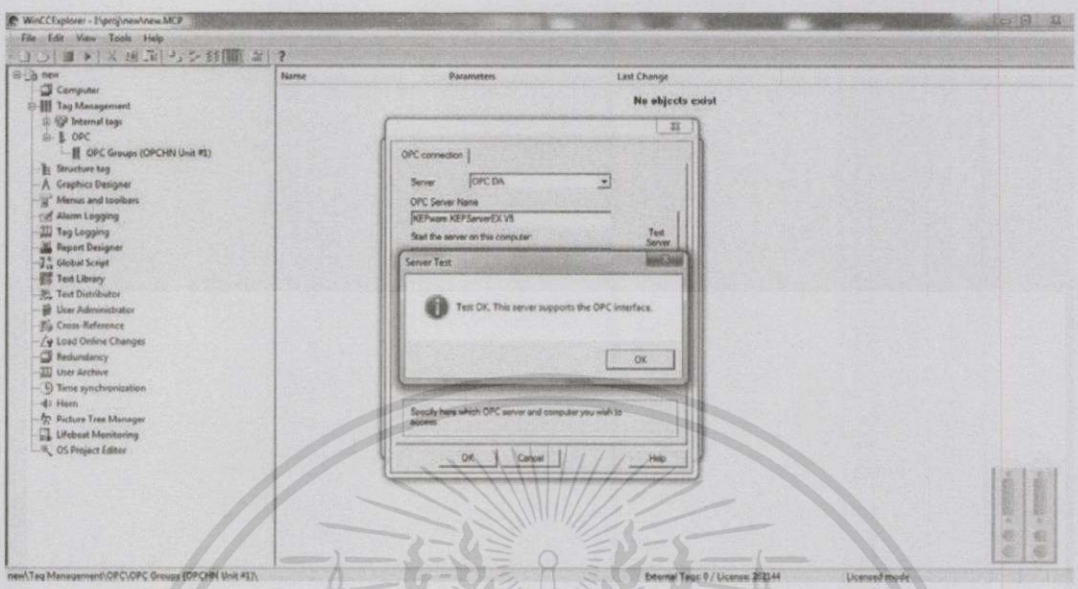


9. กรอก OPC แล้วให้กด Test Server

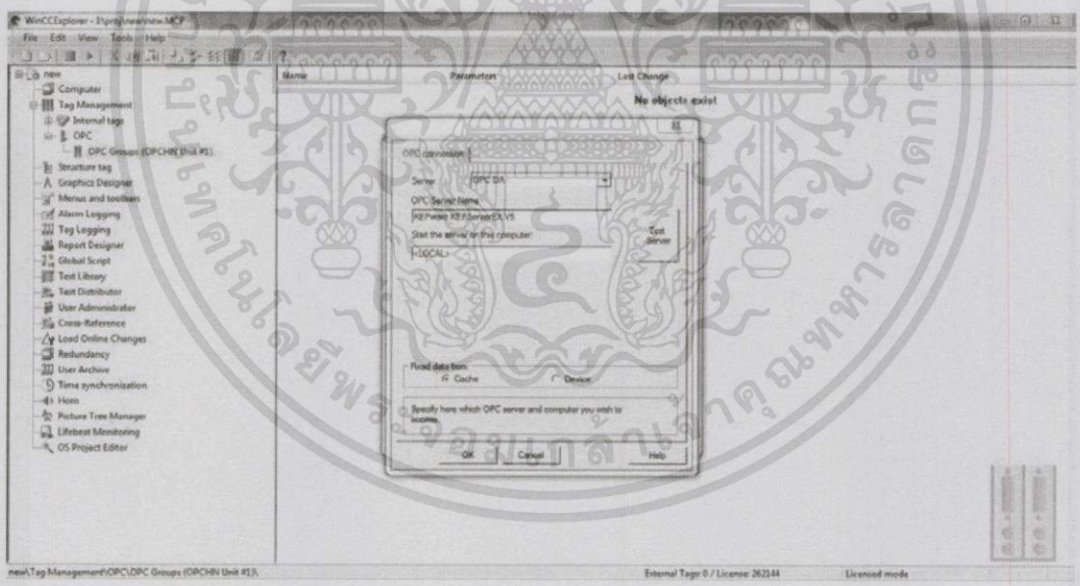


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. จะได้หน้าต่างนี้แล้วกด OK

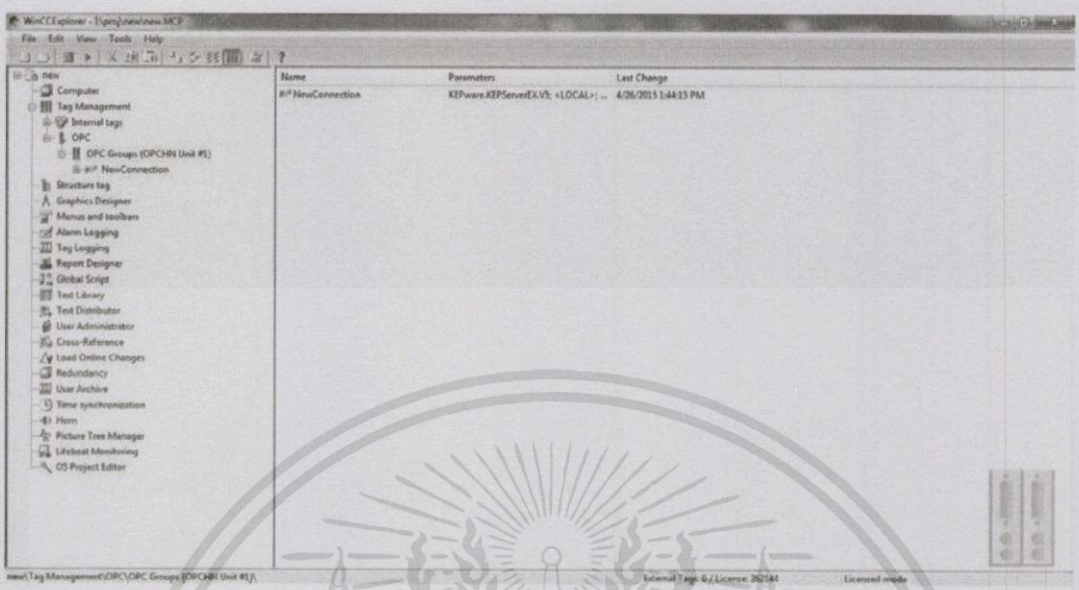


11. จะได้หน้าตาดังรูปแล้วกด OK



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

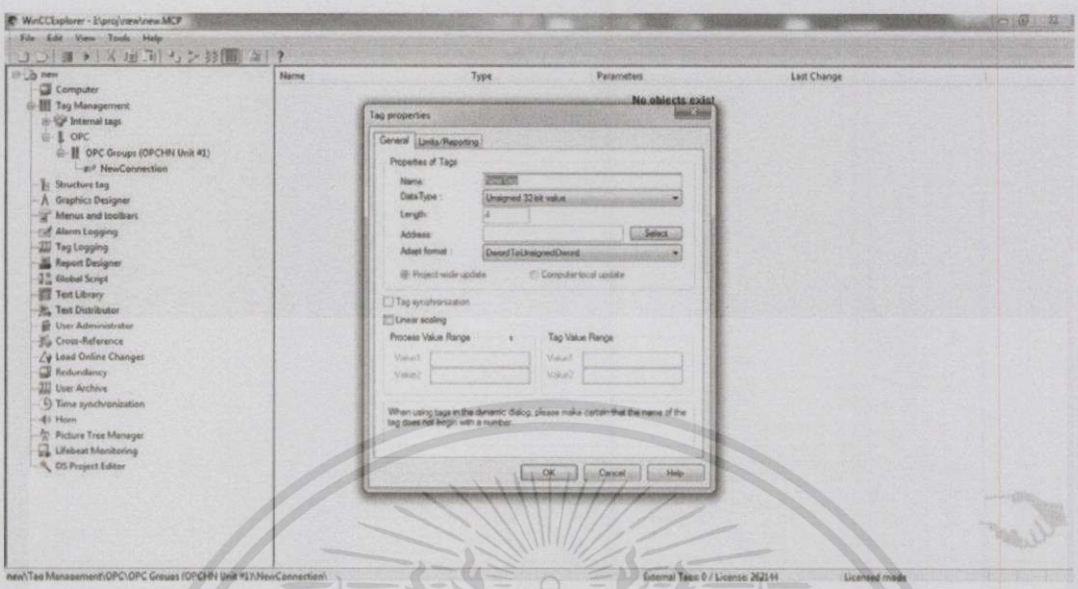
12. จะได้ ไอคอนดังรูป แล้วกดคลิกเข้าไป



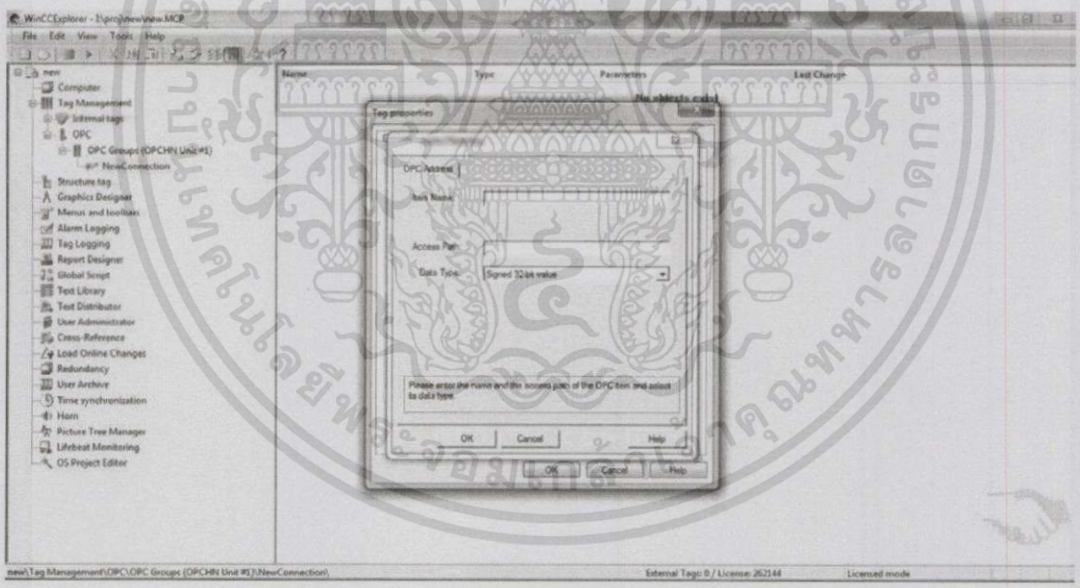
13. คลิกขวาเลือกที่ New Tag

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14. ตั้งชื่อ tag ที่ช่อง Name เลือก Data Type และที่ช่อง Address กด Select

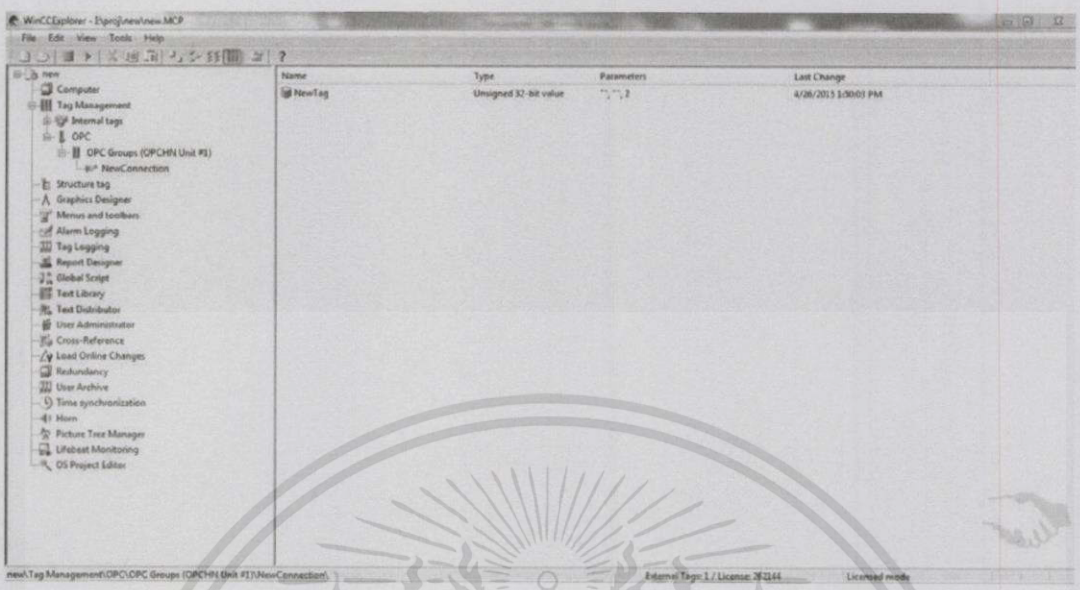


15. กรอก Address ของ tag บน OPC ลงไปช่อง Item Name



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

16. จะได้ tag ดังรูปหากต้องการ tag เพิ่มอีกให้ทำตามขั้นตอน 13-15



สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

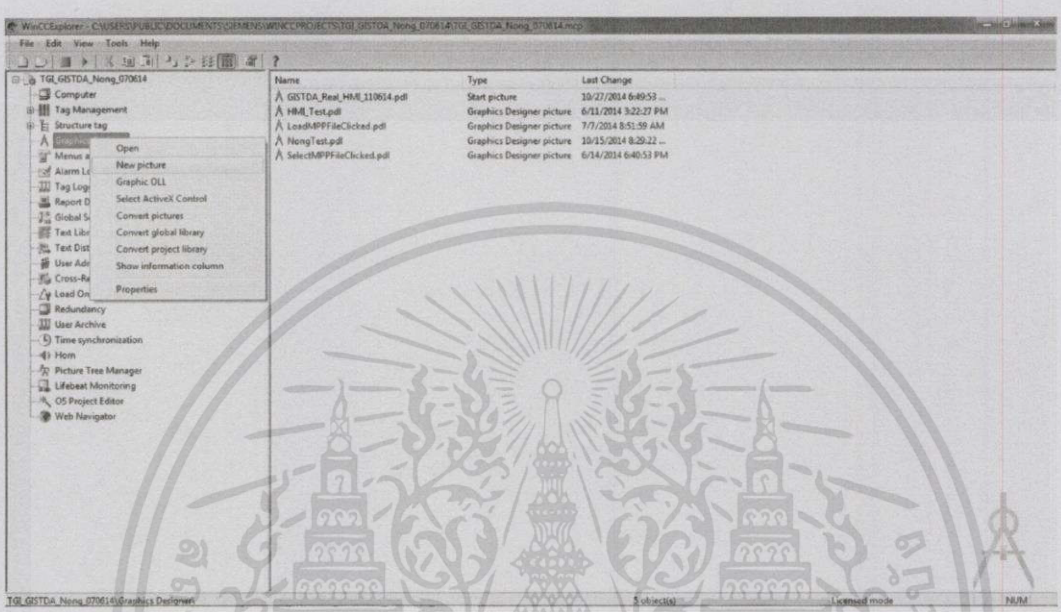
ภาคผนวก ง

การออกแบบกราฟิกบนโปรแกรม WinCC Explorer V7.0

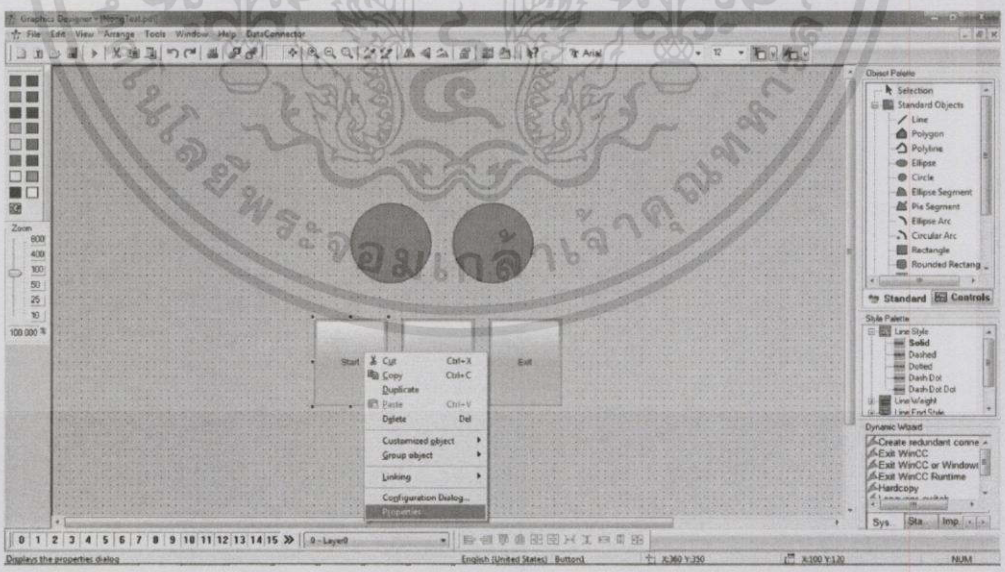
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบกราฟิกบนโปรแกรม WinCC Explorer V7.0

1. คลิกขวาที่ Graphics Design --> New Picture

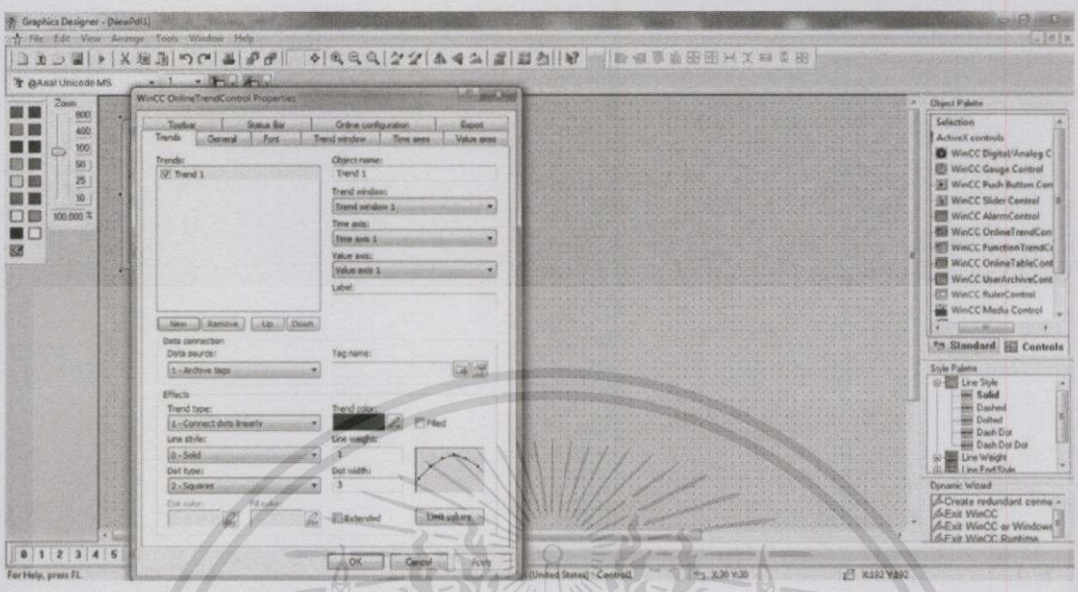


2. วาดรูปต่างๆ ตามที่ต้องการ --> คลิกขวาเลือก Properties --> ตั้งค่าตามต้องการ

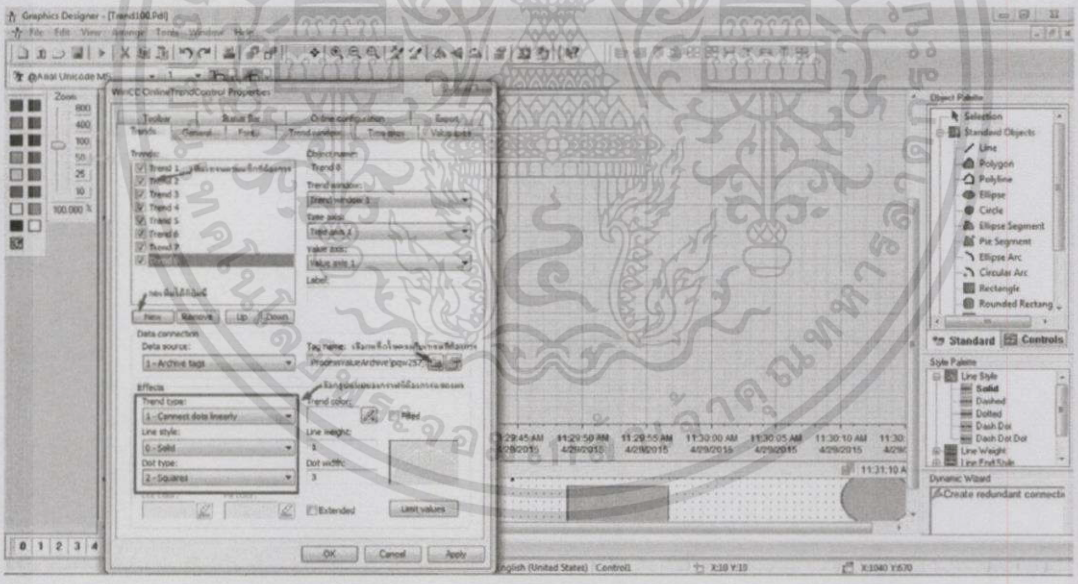


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เขียนกราฟิกโดยใช้เทรนกราฟ --> เลือก WinCCOnlineTrendControl ทางด้านหน้าต่างซ้ายมือ



4. รายละเอียดต่างๆสำหรับการตั้งค่าเทรนกราฟ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ
ซอร์ดโค้ดของโปรแกรมในการเขียนกราฟิกบางส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอร์ดโค้ดของโปรแกรมในการเขียนกราฟิกบางส่วน

```
Sub analogOPC_100_1
Dim i
Dim piw800
Dim pqw800
Set piw800 =
HMIRuntime.Tags("piw800_100")
i = piw800.Read(1)
HMIRuntime.Tags("a").Write i
If i > 0 And i<40000 Then
pqw800 = piw800.Read(1)/2813.292
HMIRuntime.Tags("100_output_ana_1").
Write pqw800
End If
If  i > 40001 Then
pqw800 = piw800.Read(1)/5000
HMIRuntime.Tags("100_output_ana_1").
Write pqw800
End If
End Sub
```

```
Sub analogOPC_100_2
Dim i
Dim piw800
Dim pqw800
Set piw800 =
HMIRuntime.Tags("piw802_100")
i = piw800.Read(1)
If i > 0 And i<40000 Then
pqw800 = piw800.Read(1)/2813.292
HMIRuntime.Tags("100_output_ana_2").
Write pqw800
End If
If  i > 40001 Then
pqw800 = piw800.Read(1)/5000
HMIRuntime.Tags("100_output_ana_2").
Write pqw800
End If
End Sub
```

```
Sub analogOPC_100_3
Dim i
Dim piw800
Dim pqw800
Set piw800 =
HMIRuntime.Tags("piw256_100")
i = piw800.Read(1)
If i > 0 And i<40000 Then
pqw800 = piw800.Read(1)/2813.292
HMIRuntime.Tags("100_output_ana_3").
Write pqw800
End If
If  i > 40001 Then
pqw800 = piw800.Read(1)/5000
HMIRuntime.Tags("100_output_ana_3").
Write pqw800
End If
End Sub
```

```
Sub analogOPC_100_4
Dim i
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim piw800

Dim pqw800

Set piw800 =
HMIRuntime.Tags("piw257_100")

i = piw800.Read(1)

If i > 0 And i<40000 Then

pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_4").
Write pqw800

End If

If  i > 40001 Then

pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_4").
Write pqw800

End If

End Sub


Sub analogOPC_100_5

Dim i

Dim piw800

Dim pqw800

Set piw800 =
HMIRuntime.Tags("pqw800_100")

i = piw800.Read(1)

If i > 0 And i<40000 Then

pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_5").
Write pqw800


End If

If  i > 40001 Then

pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_5").
Write pqw800

End If

End Sub


Sub analogOPC_100_6

Dim i

Dim piw800

Dim pqw800

Set piw800 =
HMIRuntime.Tags("pqw802_100")

i = piw800.Read(1)

If i > 0 And i<40000 Then

pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_6").
Write pqw800


End If

If  i > 40001 Then

pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_6").
Write pqw800

End If

End Sub


Sub analogOPC_100_7

Dim i

Dim piw800

Dim pqw800

Set piw800 =
HMIRuntime.Tags("pqw256_100")

i = piw800.Read(1)

If i > 0 And i<40000 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_7").
Write pqw800


End If

If  i > 40001 Then

pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_7").
Write pqw800

End If

End Sub


Sub analogOPC_100_8

Dim i

Dim piw800

Dim pqw800

Set piw800 =
HMIRuntime.Tags("pqw257_100")

i = piw800.Read(1)

If i > 0 And i<40000 Then

pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_8").
Write pqw800


End If

If  i > 40001 Then

pqw800 = piw800.Read(1)/2813.292

HMIRuntime.Tags("100_output_ana_8").
Write pqw800

End If

End Sub


Sub binaryOPC_100_I_1

Dim i

Dim re0,re1,re2,re3

Dim i1360,i1361,i1362,i1363


Set i1360 =
HMIRuntime.Tags("i136_0_100")

Set i1361 =
HMIRuntime.Tags("i136_1_100")

Set i1362 =
HMIRuntime.Tags("i136_2_100")

Set i1363 =
HMIRuntime.Tags("i136_3_100")

re0 = i1360.Read(1)

re1 = i1361.Read(1)

re2 = i1362.Read(1)

re3 = i1363.Read(1)


If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=0

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=1

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=3

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=4

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=5

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=6

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=7

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=8

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=9

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="A"

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="B"

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="C"

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="D"

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="E"

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="F"

HMIRuntime.Tags("100_input_digi_1").W
rite i

End If


End Sub


Sub binaryOPC_100_I_2

Dim i

Dim re0,re1,re2,re3

Dim i1360,i1361,i1362,i1363

Set i1360 =
HMIRuntime.Tags("i136_4_100")

Set i1361 =
HMIRuntime.Tags("i136_5_100")

Set i1362 =
HMIRuntime.Tags("i136_6_100")

Set i1363 =
HMIRuntime.Tags("i136_7_100")

re0 = i1360.Read(1)

re1 = i1361.Read(1)

re2 = i1362.Read(1)

re3 = i1363.Read(1)


If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=0

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If


If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=1

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If


If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=2

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=3

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

```
If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=8

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

```
If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=4

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

```
If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=9

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

```
If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=5

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

```
If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="A"

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=6

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

```
If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="B"

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

```
If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=7

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

```
If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="C"

HMIRuntime.Tags("100_input_digi_2").W
rite i

End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 1 Then
i="D"
HMIRuntime.Tags("100_input_digi_2").W
rite i
End If

If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 1 Then
i="E"
HMIRuntime.Tags("100_input_digi_2").W
rite i
End If

If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 1 Then
i="F"
HMIRuntime.Tags("100_input_digi_2").W
rite i
End If
End Sub

Sub binaryOPC_100_I_3
Dim i
Dim re0,re1,re2,re3
Dim i1360,i1361,i1362,i1363

Set i1360 =
HMIRuntime.Tags("i137_0_100")
Set i1361 =
HMIRuntime.Tags("i137_1_100")
Set i1362 =
HMIRuntime.Tags("i137_2_100")
Set i1363 =
HMIRuntime.Tags("i137_3_100")
re0 = i1360.Read(1)
re1 = i1361.Read(1)
re2 = i1362.Read(1)
re3 = i1363.Read(1)

If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 0 Then
i=0
HMIRuntime.Tags("100_input_digi_3").W
rite i
End If

If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 0 Then
i=1
HMIRuntime.Tags("100_input_digi_3").W
rite i
End If

If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 0 Then
i=2
HMIRuntime.Tags("100_input_digi_3").W
rite i
End If

If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 0 Then
i=3
HMIRuntime.Tags("100_input_digi_3").W
rite i
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 0 And re1 = 0 And re2 = 1 And re3 = 0 Then
i=4
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

```
If re0 = 1 And re1 = 0 And re2 = 0 And re3 = 1 Then
i=9
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

```
If re0 = 1 And re1 = 0 And re2 = 1 And re3 = 0 Then
i=5
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

```
If re0 = 0 And re1 = 1 And re2 = 0 And re3 = 1 Then
i="A"
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3 = 0 Then
i=6
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

```
If re0 = 1 And re1 = 1 And re2 = 0 And re3 = 1 Then
i="B"
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

```
If re0 = 1 And re1 = 1 And re2 = 1 And re3 = 0 Then
i=7
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

```
If re0 = 0 And re1 = 0 And re2 = 1 And re3 = 1 Then
i="C"
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

```
If re0 = 0 And re1 = 0 And re2 = 0 And re3 = 1 Then
i=8
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

```
If re0 = 1 And re1 = 0 And re2 = 1 And re3 = 1 Then
i="D"
HMIRuntime.Tags("100_input_digi_3").Write i
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="E"

HMIRuntime.Tags("100_input_digi_3").W
rite i

End If


If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="F"

HMIRuntime.Tags("100_input_digi_3").W
rite i

End If

End Sub


Sub binaryOPC_100_I_4

Dim i

Dim re0,re1,re2,re3

Dim i1360,i1361,i1362,i1363


Set i1360 =
HMIRuntime.Tags("i137_4_100")

Set i1361 =
HMIRuntime.Tags("i137_5_100")

Set i1362 =
HMIRuntime.Tags("i137_6_100")

Set i1363 =
HMIRuntime.Tags("i137_7_100")

re0 = i1360.Read(1)

re1 = i1361.Read(1)

re2 = i1362.Read(1)

re3 = i1363.Read(1)


If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=0

HMIRuntime.Tags("100_input_digi_4").W
rite i

End If


If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=1

HMIRuntime.Tags("100_input_digi_4").W
rite i

End If


If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=2

HMIRuntime.Tags("100_input_digi_4").W
rite i

End If


If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=3

HMIRuntime.Tags("100_input_digi_4").W
rite i

End If


If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=4

HMIRuntime.Tags("100_input_digi_4").W
rite i

End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 0 Then
i=5
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

```
If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 1 Then
i="A"
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 0 Then
i=6
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

```
If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 1 Then
i="B"
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

```
If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 0 Then
i=7
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

```
If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 1 Then
i="C"
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

```
If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 1 Then
i=8
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

```
If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 1 Then
i="D"
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

```
If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 1 Then
i=9
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 1 Then
i="E"
HMIRuntime.Tags("100_input_digi_4").W
rite i
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="F"

HMIRuntime.Tags("100_input_digi_4").W
rite i

End If

End Sub


Sub binaryOPC_100_O_1

Dim i

Dim re0,re1,re2,re3

Dim i1360,i1361,i1362,i1363


Set i1360 =
HMIRuntime.Tags("q136_0_100")

Set i1361 =
HMIRuntime.Tags("q136_1_100")

Set i1362 =
HMIRuntime.Tags("q136_2_100")

Set i1363 =
HMIRuntime.Tags("q136_3_100")

re0 = i1360.Read(1)

re1 = i1361.Read(1)

re2 = i1362.Read(1)

re3 = i1363.Read(1)


If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=0

HMIRuntime.Tags("100_output_digi_1").
Write i

End If

If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=1

HMIRuntime.Tags("100_output_digi_1").
Write i

End If


If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=2

HMIRuntime.Tags("100_output_digi_1").
Write i

End If


If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=3

HMIRuntime.Tags("100_output_digi_1").
Write i

End If


If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=4

HMIRuntime.Tags("100_output_digi_1").
Write i

End If


If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=5

HMIRuntime.Tags("100_output_digi_1").
Write i

End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=6

HMIRuntime.Tags("100_output_digi_1").
Write i

End If
```

```
If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=7

HMIRuntime.Tags("100_output_digi_1").
Write i

End If
```

```
If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=8

HMIRuntime.Tags("100_output_digi_1").
Write i

End If
```

```
If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=9

HMIRuntime.Tags("100_output_digi_1").
Write i

End If

If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="A"

HMIRuntime.Tags("100_output_digi_1").
Write i

End If
```

```
If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="B"

HMIRuntime.Tags("100_output_digi_1").
Write i

End If
```

```
If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="C"

HMIRuntime.Tags("100_output_digi_1").
Write i

End If
```

```
If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="D"

HMIRuntime.Tags("100_output_digi_1").
Write i

End If
```

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="E"

HMIRuntime.Tags("100_output_digi_1").
Write i

End If

If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="F"

HMIRuntime.Tags("100_output_digi_1").
Write i
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If

End Sub


Sub binaryOPC_100_O_2

Dim i

Dim re0,re1,re2,re3

Dim i1360,i1361,i1362,i1363


Set i1360 =
HMIRuntime.Tags("q136_4_100")

Set i1361 =
HMIRuntime.Tags("q136_5_100")

Set i1362 =
HMIRuntime.Tags("q136_6_100")

Set i1363 =
HMIRuntime.Tags("q136_7_100")

re0 = i1360.Read(1)

re1 = i1361.Read(1)

re2 = i1362.Read(1)

re3 = i1363.Read(1)


If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=0

HMIRuntime.Tags("100_output_digi_2").
Write i

End If


If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=1

HMIRuntime.Tags("100_output_digi_2").
Write i

End If


If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=2

HMIRuntime.Tags("100_output_digi_2").
Write i

End If


If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=3

HMIRuntime.Tags("100_output_digi_2").
Write i

End If


If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=4

HMIRuntime.Tags("100_output_digi_2").
Write i

End If


If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=5

HMIRuntime.Tags("100_output_digi_2").
Write i

End If


If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=6

HMIRuntime.Tags("100_output_digi_2").
Write i

End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=7

HMIRuntime.Tags("100_output_digi_2").
Write i

End If
```

```
If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=8

HMIRuntime.Tags("100_output_digi_2").
Write i

End If
```

```
If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=9

HMIRuntime.Tags("100_output_digi_2").
Write i

End If
```

```
If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="A"

HMIRuntime.Tags("100_output_digi_2").
Write i

End If
```

```
If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="B"

HMIRuntime.Tags("100_output_digi_2").
Write i

End If
```

```
If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="C"

HMIRuntime.Tags("100_output_digi_2").
Write i

End If
```

```
If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="D"

HMIRuntime.Tags("100_output_digi_2").
Write i

End If
```

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="E"

HMIRuntime.Tags("100_output_digi_2").
Write i

End If
```

```
If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="F"

HMIRuntime.Tags("100_output_digi_2").
Write i

End If

End Sub


Sub binaryOPC_100_O_3

Dim i

Dim re0,re1,re2,re3

Dim i1360,i1361,i1362,i1363
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Set i1360 =
HMIRuntime.Tags("q137_0_100")

Set i1361 =
HMIRuntime.Tags("q137_1_100")

Set i1362 =
HMIRuntime.Tags("q137_2_100")

Set i1363 =
HMIRuntime.Tags("q137_3_100")

re0 = i1360.Read(1)

re1 = i1361.Read(1)

re2 = i1362.Read(1)

re3 = i1363.Read(1)


If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=0

HMIRuntime.Tags("100_output_digi_3").
Write i

End If


If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=1

HMIRuntime.Tags("100_output_digi_3").
Write i

End If


If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=2

HMIRuntime.Tags("100_output_digi_3").
Write i

End If


If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=3

HMIRuntime.Tags("100_output_digi_3").
Write i

End If


If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=4

HMIRuntime.Tags("100_output_digi_3").
Write i

End If


If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=5

HMIRuntime.Tags("100_output_digi_3").
Write i

End If


If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=6

HMIRuntime.Tags("100_output_digi_3").
Write i

End If


If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=7

HMIRuntime.Tags("100_output_digi_3").
Write i

End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=8

HMIRuntime.Tags("100_output_digi_3").
Write i

End If
```

```
If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=9

HMIRuntime.Tags("100_output_digi_3").
Write i

End If

If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="A"

HMIRuntime.Tags("100_output_digi_3").
Write i

End If
```

```
If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="B"

HMIRuntime.Tags("100_output_digi_3").
Write i

End If
```

```
If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="C"

HMIRuntime.Tags("100_output_digi_3").
Write i

End If
```

```
If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="D"

HMIRuntime.Tags("100_output_digi_3").
Write i

End If
```

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="E"

HMIRuntime.Tags("100_output_digi_3").
Write i

End If

If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="F"

HMIRuntime.Tags("100_output_digi_3").
Write i

End If

End Sub

Sub binaryOPC_100_O_4

Dim i

Dim re0,re1,re2,re3

Dim i1360,i1361,i1362,i1363


Set i1360 =
HMIRuntime.Tags("q137_4_100")

Set i1361 =
HMIRuntime.Tags("q137_5_100")

Set i1362 =
HMIRuntime.Tags("q137_6_100")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Set i1363 =
HMIRuntime.Tags("q137_7_100")

re0 = i1360.Read(1)

re1 = i1361.Read(1)

re2 = i1362.Read(1)

re3 = i1363.Read(1)


If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=0

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 0 Then

i=1

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=2

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 0 Then

i=3

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=4

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 0 Then

i=5

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=6

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 0 Then

i=7

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 0 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=8

HMIRuntime.Tags("100_output_digi_4").
Write i

End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If re0 = 1 And re1 = 0 And re2 = 0 And re3
= 1 Then

i=9

HMIRuntime.Tags("100_output_digi_4").
Write i

End If

If re0 = 0 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="A"

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 1 And re1 = 1 And re2 = 0 And re3
= 1 Then

i="B"

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 0 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="C"

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 1 And re1 = 0 And re2 = 1 And re3
= 1 Then

i="D"

HMIRuntime.Tags("100_output_digi_4").
Write i

End If
```

```
If re0 = 0 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="E"

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


If re0 = 1 And re1 = 1 And re2 = 1 And re3
= 1 Then

i="F"

HMIRuntime.Tags("100_output_digi_4").
Write i

End If


End Sub


Sub button_100_I_800


ScreenItems("I/O Field6").Transparency=0

ScreenItems("I/O
Field7").Transparency=100

ScreenItems("I/O
Field8").Transparency=100

ScreenItems("I/O
Field9").Transparency=100

ScreenItems("I/O
Field10").Transparency=100

ScreenItems("I/O
Field11").Transparency=100

ScreenItems("I/O
Field12").Transparency=100

ScreenItems("I/O
Field13").Transparency=100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ScreenItems("Static
Text24").BackColor=RGB(255,161,0)

ScreenItems("Static
Text25").BackColor=RGB(218,218,218)

ScreenItems("Static
Text72").BackColor=RGB(218,218,218)

ScreenItems("Static
Text73").BackColor=RGB(218,218,218)

ScreenItems("Static
Text90").BackColor=RGB(218,218,218)

ScreenItems("Static
Text91").BackColor=RGB(218,218,218)

ScreenItems("Static
Text92").BackColor=RGB(218,218,218)

ScreenItems("Static
Text93").BackColor=RGB(218,218,218)


End Sub


Sub button_100_I_802


ScreenItems("I/O
Field6").Transparency=100

ScreenItems("I/O Field7").Transparency=0

ScreenItems("I/O
Field8").Transparency=100

ScreenItems("I/O
Field9").Transparency=100

ScreenItems("I/O
Field10").Transparency=100

ScreenItems("I/O
Field11").Transparency=100

ScreenItems("I/O
Field12").Transparency=100

ScreenItems("I/O
Field13").Transparency=100
```

```
ScreenItems("Static
Text24").BackColor=RGB(218,218,218)

ScreenItems("Static
Text25").BackColor=RGB(255,161,0)

ScreenItems("Static
Text72").BackColor=RGB(218,218,218)

ScreenItems("Static
Text73").BackColor=RGB(218,218,218)

ScreenItems("Static
Text90").BackColor=RGB(218,218,218)

ScreenItems("Static
Text91").BackColor=RGB(218,218,218)

ScreenItems("Static
Text92").BackColor=RGB(218,218,218)

ScreenItems("Static
Text93").BackColor=RGB(218,218,218)


End Sub


Sub button_100_I_256


ScreenItems("I/O
Field6").Transparency=100

ScreenItems("I/O
Field7").Transparency=100

ScreenItems("I/O Field8").Transparency=0

ScreenItems("I/O
Field9").Transparency=100

ScreenItems("I/O
Field10").Transparency=100

ScreenItems("I/O
Field11").Transparency=100

ScreenItems("I/O
Field12").Transparency=100

ScreenItems("I/O
Field13").Transparency=100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ScreenItems("Static
Text24").BackColor=RGB(218,218,218)

ScreenItems("Static
Text25").BackColor=RGB(218,218,218)

ScreenItems("Static
Text72").BackColor=RGB(255,161,0)

ScreenItems("Static
Text73").BackColor=RGB(218,218,218)

ScreenItems("Static
Text90").BackColor=RGB(218,218,218)

ScreenItems("Static
Text91").BackColor=RGB(218,218,218)

ScreenItems("Static
Text92").BackColor=RGB(218,218,218)

ScreenItems("Static
Text93").BackColor=RGB(218,218,218)


End Sub


Sub button_100_I_257


ScreenItems("I/O
Field6").Transparency=100

ScreenItems("I/O
Field7").Transparency=100

ScreenItems("I/O
Field8").Transparency=100

ScreenItems("I/O Field9").Transparency=0

ScreenItems("I/O
Field10").Transparency=100

ScreenItems("I/O
Field11").Transparency=100

ScreenItems("I/O
Field12").Transparency=100

ScreenItems("I/O
Field13").Transparency=100
```

```
ScreenItems("Static
Text24").BackColor=RGB(218,218,218)

ScreenItems("Static
Text25").BackColor=RGB(218,218,218)

ScreenItems("Static
Text72").BackColor=RGB(218,218,218)

ScreenItems("Static
Text73").BackColor=RGB(255,161,0)

ScreenItems("Static
Text90").BackColor=RGB(218,218,218)

ScreenItems("Static
Text91").BackColor=RGB(218,218,218)

ScreenItems("Static
Text92").BackColor=RGB(218,218,218)

ScreenItems("Static
Text93").BackColor=RGB(218,218,218)


End Sub


Sub button_100_O_800


ScreenItems("I/O
Field6").Transparency=100

ScreenItems("I/O
Field7").Transparency=100

ScreenItems("I/O
Field8").Transparency=100

ScreenItems("I/O
Field9").Transparency=100

ScreenItems("I/O
Field10").Transparency=0

ScreenItems("I/O
Field11").Transparency=100

ScreenItems("I/O
Field12").Transparency=100

ScreenItems("I/O
Field13").Transparency=100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ScreenItems("Static
Text24").BackColor=RGB(218,218,218)

ScreenItems("Static
Text25").BackColor=RGB(218,218,218)

ScreenItems("Static
Text72").BackColor=RGB(218,218,218)

ScreenItems("Static
Text73").BackColor=RGB(218,218,218)

ScreenItems("Static
Text90").BackColor=RGB(255,161,0)

ScreenItems("Static
Text91").BackColor=RGB(218,218,218)

ScreenItems("Static
Text92").BackColor=RGB(218,218,218)

ScreenItems("Static
Text93").BackColor=RGB(218,218,218)


End Sub


Sub button_100_O_802


ScreenItems("I/O
Field6").Transparency=100

ScreenItems("I/O
Field7").Transparency=100

ScreenItems("I/O
Field8").Transparency=100

ScreenItems("I/O
Field9").Transparency=100

ScreenItems("I/O
Field10").Transparency=100

ScreenItems("I/O
Field11").Transparency=0

ScreenItems("I/O
Field12").Transparency=100

ScreenItems("I/O
Field13").Transparency=100
```

```
ScreenItems("Static
Text24").BackColor=RGB(218,218,218)

ScreenItems("Static
Text25").BackColor=RGB(218,218,218)

ScreenItems("Static
Text72").BackColor=RGB(218,218,218)

ScreenItems("Static
Text73").BackColor=RGB(218,218,218)

ScreenItems("Static
Text90").BackColor=RGB(218,218,218)

ScreenItems("Static
Text91").BackColor=RGB(255,161,0)

ScreenItems("Static
Text92").BackColor=RGB(218,218,218)

ScreenItems("Static
Text93").BackColor=RGB(218,218,218)


End Sub


Sub button_100_O_256

ScreenItems("I/O
Field6").Transparency=100

ScreenItems("I/O
Field7").Transparency=100

ScreenItems("I/O
Field8").Transparency=100

ScreenItems("I/O
Field9").Transparency=100

ScreenItems("I/O
Field10").Transparency=100

ScreenItems("I/O
Field11").Transparency=100

ScreenItems("I/O
Field12").Transparency=0

ScreenItems("I/O
Field13").Transparency=100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ScreenItems("Static
Text24").BackColor=RGB(218,218,218)

ScreenItems("Static
Text25").BackColor=RGB(218,218,218)

ScreenItems("Static
Text72").BackColor=RGB(218,218,218)

ScreenItems("Static
Text73").BackColor=RGB(218,218,218)

ScreenItems("Static
Text90").BackColor=RGB(218,218,218)

ScreenItems("Static
Text91").BackColor=RGB(218,218,218)

ScreenItems("Static
Text92").BackColor=RGB(255,161,0)

ScreenItems("Static
Text93").BackColor=RGB(218,218,218)

End Sub


Sub button_100_O_257


ScreenItems("I/O
Field6").Transparency=100

ScreenItems("I/O
Field7").Transparency=100

ScreenItems("I/O
Field8").Transparency=100

ScreenItems("I/O
Field9").Transparency=1000

ScreenItems("I/O
Field10").Transparency=100

ScreenItems("I/O
Field11").Transparency=100

ScreenItems("I/O
Field12").Transparency=100

ScreenItems("I/O
Field13").Transparency=0

ScreenItems("Static
Text24").BackColor=RGB(218,218,218)

ScreenItems("Static
Text25").BackColor=RGB(218,218,218)

ScreenItems("Static
Text72").BackColor=RGB(218,218,218)

ScreenItems("Static
Text73").BackColor=RGB(218,218,218)

ScreenItems("Static
Text90").BackColor=RGB(218,218,218)

ScreenItems("Static
Text91").BackColor=RGB(218,218,218)

ScreenItems("Static
Text92").BackColor=RGB(218,218,218)

ScreenItems("Static
Text93").BackColor=RGB(255,161,0)

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้