สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การจำลองกระบวนการการจัดเก็บสินค้าอัตโนมัติโดยใช้ PLC

Warehouse Process Simulation by Using PLC

นาย นพดล คำเกลี้ยง
นาย บุญส่ง ด่านลำมะจาก
นาย วิษณุโยธิน รินทะรักษ์

เลขหมู่.....
เลขทะเบียน 82004
วัน,เดือน,ปี 4 ก.ค. 2551

b 11943993
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมระบบควบคุม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2550

ภาควิชา วิศวกรรมระบบควบคุม คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การจำลองกระบวนการการจัดเก็บสินค้าอัตโนมัติโดยใช้ PLC

Warehouse Process Simulation by Using PLC

ผู้จัดทำ	**นาย นพดล**	**คำเกลี้ยง**	**รหัส**	**48015291**
	นาย บุญส่ง	**ด่านลำมะจาก**	**รหัส**	**48015295**
	นาย วิษณุโยธิน	**รินทะรักษ์**	**รหัส**	**48015309**

.....อาจารย์ที่ปรึกษา

(ผศ.ดร. คงศักดิ์ อนันตหิรัญรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจำลองกระบวนการการจัดเก็บสินค้าอัตโนมัติโดยใช้ PLC

โดย

นาย นพดล คำเกลี้ยง รหัส 48015291

นาย บุญส่ง ด่านลำมะจาก รหัส 48015295

นาย วิษณุโยธิน รินทะรักษ์ รหัส 48015309

อาจารย์ที่ปรึกษา

ผศ.ดร. คงศักดิ์ อนันตหิรัญรัตน์

ปีการศึกษา 2550

บทคัดย่อ

เนื่องจากปัจจุบันอุตสาหกรรมส่วนใหญ่ในประเทศไทยได้นำเอาโปรแกรมเมเบิลลอจิกคอนโทรลเลอร์ (Programmable Logic Controller) หรือเรียกว่า " PLC " มาใช้งานในระบบควบคุมอัตโนมัติ ซึ่งมีความน่าเชื่อถือสูงและใช้งานได้ง่าย โดยนำเอา PLC มาใช้ควบคุมการลำเลียงสินค้าเพื่อให้การจัดเก็บสินค้ามีประสิทธิภาพสูงสุด ดังนั้นปริญญานิพนธ์ฉบับนี้ จึงนำเสนอโดยการจำลองกระบวนการจัดเก็บสินค้าขึ้นด้วยระบบ PLC โดยมีการควบคุมการทำงานผ่านทางโปรแกรมบนหน้าจอคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม RS-232C ซึ่งแสดงผลกระบวนการการทำงานโดยจำลองภาพ มีการบันทึกรายการสินค้าที่อยู่ในกระบวนการและสามารถตรวจสอบสินค้าได้อย่างถูกต้อง โดยกระบวนการทั้งหมดจำลองด้วยโปรแกรมวิชวลเบสิก 6 (Visual Basic 6)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Warehouse Process Simulation by Using PLC

By

Mr. Nophadol Kumkleang

Mr. Boonsong Danlammajak

Mr. Witsanuyothin Rintarak

Advisor

Asst.Prof.Dr. Kongsak Anuntahirunrat

Academic Year 2007

ABSTRACT

Another present time industrial use of PLC machines in automatic control systems is prevalent due to their reliability and their easy to use property. This reason comes from the need of industrial sector for an efficient transportation system, which leads to the most effective storage and mass product transportation. This thesis proposes the theory of transportation control systems design, with PLC systems controlled by application on computer connected through serial port RS-232. This application shows simulation process of model with control button to power. The precise records process of model with Visual Basic 6 (VB 6)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติประกาศ

การจัดทำปริญญานิพนธ์ฉบับนี้สามารถลุล่วงไปได้ด้วยดีเพราะได้รับการช่วยเหลือเป็นอย่างดีจากท่านอาจารย์ ผศ.ดร.ดงศักดิ์ อนันตหิรัญรัตน์ ซึ่งท่านเป็นอาจารย์ที่ปรึกษาโครงงานนี้ และท่านอาจารย์ รศ.สุเธียร เกียรติสุนทร ท่านอาจารย์ทั้งสองได้ให้คำปรึกษาเป็นอย่างดี คำแนะนำต่างๆ โดยเฉพาะความรู้ใน PLC และเทคนิคอื่นๆที่เป็นประโยชน์ในโครงงาน

ขอบคุณพี่ๆ ห้องเครื่องมือภาคระบบควบคุม ที่คอยสนับสนุนอุปกรณ์ที่ใช้ในการทำงานและให้คำแนะนำต่างๆในการใช้เครื่องมือกลและเพื่อนๆห้อง 3/O ที่คอยเป็นกำลังใจเสมอมา

สุดท้ายนี้ผู้จัดทำขอกราบขอบพระคุณ บิดา มารดาและครอบครัวที่คอยเป็นกำลังใจที่ดีตลอดมา รวมถึงการสนับสนุนการศึกษาเล่าเรียนตลอดจนคอยเป็นแรงบันดาลใจที่ดี จนทำให้โครงงานนี้สำเร็จสมบูรณ์

ผู้จัดทำ

นาย นพดล คำเกลี้ยง

นาย บุญส่ง ด่านลำมะจาก

นาย วิษณุโยธิน รินทะรักษ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญภาพ	VIII
บทที่ 1 บทนำ	I
1.1 ความเป็นมา	I
1.2 วัตถุประสงค์ของโครงงาน	1
1.3 ขอบเขตของโครงงาน	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 เนื้อหาโดยสังเขป	2
บทที่ 2 หลักการและทฤษฎี	3
2.1 ทฤษฎีเบื้องต้นโปรแกรมเมเบิลลอจิกคอนโทรลเลอร์ (PLC)	3
2.1.1 ประวัติความเป็นมาของ PLC	3
2.1.2 โครงสร้างโดยทั่วไปและคุณสมบัติของ PLC รุ่น CQM 1	3
2.1.3 หลักการทำงาน	8
2.1.4 การทำงานของหน่วยต่างๆ ภายใน PLC	8
2.2 โปรแกรมวิชวลเบสิก6 (Visual Basic 6.0)	10
2.2.1 ที่มาและลักษณะของโปรแกรมวิชวล เบสิก6 Visual Basic 6	10
2.2.2 จุดเด่นของ Visual Basic	10
2.2.3 รูปแบบการพัฒนาแอพพลิเคชั่นกับ Visual Basic	11
2.2.4 รายละเอียดของส่วนประกอบต่างๆของหน้าจอ	12
2.2.5 เริ่มสร้างแอพพลิเคชั่นด้วย Visual Basic	13
2.2.6 การสื่อสารกับ Communication Port	14
2.2.7 คุณสมบัติของ MSComm	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

เรื่อง	หน้า
2.3 การติดต่อสื่อสารและการส่งผ่านข้อมูล	17
2.3.1 กล่าวนำ	17
2.3.2 การสื่อสารข้อมูลทั่วไป	18
2.3.3 การส่งข้อมูลแบบซิมเพล็กซ์ และแบบดูเพล็กซ์	18
2.3.4 การส่งข้อมูลแบบอนุกรมในเครื่องคอมพิวเตอร์	19
2.3.5 การส่งข้อมูลระหว่างโปรแกรมวิชวล เบสิกกับ PLC	21
2.3.5.1 รูปแบบชุดคำสั่ง (Command Format)	21
2.3.5.2 รูปแบชุดผลตอบสนอง (Response Block)	22
2.3.5.3 การคำนวณ FCS	23
2.4 หลักการทำงานของวงจร	24
2.4.1 วงจรเซนเซอร์	24
2.4.2 วงจรรีเลย์และวงจรขับมอเตอร์	25
บทที่ 3 การออกแบบ สร้างและการทดลอง	27
3.1 การออกแบบโปรแกรมแลดเดอร์ควบคุมการทำงานของ Warehouse	27
3.1.1 ส่วนเก็บข้อมูลที่ผู้ใช้เลือกช่องที่ต้องการ	27
3.1.2 ส่วนป้องกันการทับซ้อนกัน	28
3.1.3 ส่วนเรียกข้อมูลที่เลือกไว้มาใช้งาน	28
3.1.4 ส่วนลำดับการเคลื่อนที่	28
3.1.5 ส่วน Emergency Stop	29
3.2 การออกแบบส่วนควบคุมการทำงาน	29
3.2.1 ส่วน Manual Operation	29
3.2.2 ส่วน Auto Operation	30
3.3 ออกแบบแอพพลิเคชั่นด้วยโปรแกรมวิชวลเบสิก 6 และผลการทดลอง	30
3.4 ออกแบบและสร้างส่วนของ Hardware	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

เรื่อง	หน้า
บทที่ 4 ผลการทดลอง	32
4.1 เปิดโปรแกรมจัดเก็บสินค้า	32
4.2 การจัดเก็บสินค้าเข้า	32
4.3 การขนถ่ายสินค้าขาออก	34
บทที่ 5 บทวิจารณ์และสรุป	36
5.1 สรุปผลการปฏิบัติงาน	36
5.2 ปัญหาด้านการออกแบบโครงสร้างแบบจำลอง	36
5.3 ข้อเสนอแนะและแนวทางในการค้นคว้าและพัฒนา	36
ภาคผนวก ก. โปรแกรมวิชวลเบสิกและภาษาแลดเดอร์	39
ภาคผนวก ข. Host Link Commands	64
ภาคผนวก ค. Memory Areas CPM1/CPM1A and CQM1 Memory Area	67
ภาคผนวก ง. Extended ASCII	68
ภาคผนวก จ. วงจรและแหล่งจ่ายไฟ	69
เอกสารอ้างอิง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่		หน้า
รูปที่ 2.1	PLC ชนิดโมดูล	3
รูปที่ 2.2	แสดงส่วนประกอบภายใน	4
รูปที่ 2.3	แสดงหน่วยประมวลผล (CPU) และ หน่วยความจำ (Memory)	5
รูปที่ 2.4	แสดงหน่วยอินพุต/เอาต์พุต (Input / Output Unit)	6
รูปที่ 2.5	แสดงตัวป้อนโปรแกรมแบบมือถือ (Hand Held Programmer)	7
รูปที่ 2.6	แสดงการต่อใช้งานร่วมกับ PLC	7
รูปที่ 2.7	แสดงหน่วยจ่ายพลังงาน (Power Supply)	8
รูปที่ 2.8	แสดงองค์ประกอบต่างๆของโปรแกรมวิชวล เบสิก 6	13
รูปที่ 2.9	แสดงการเปิดใช้งาน Microsoft Comm Control 6.0	14
รูปที่ 2.10	แสดงส่วนประกอบหลักในการสื่อสารข้อมูล	18
รูปที่ 2.11	แสดงการส่งข้อมูลแบบทิศทางเดียว (Simplex)	19
รูปที่ 2.12	แสดงการส่งข้อมูลแบบฮาล์ฟดูเพล็กซ์ (Half Duplex)	19
รูปที่ 2.13	แสดงการส่งข้อมูลแบบฟูลดูเพล็กซ์ (Full Duplex)	19
รูปที่ 2.14	แสดงการเชื่อมต่อ PLC กับคอมพิวเตอร์ผ่านพอร์ตสื่อสารอนุกรมขนาด 9 ขา	21
รูปที่ 2.15	แสดงรูปแบบของชุดข้อมูล (Block)	21
รูปที่ 2.16	แสดงรูปแบบของชุดคำสั่ง	21
รูปที่ 2.17	แสดงรูปแบบของชุดผลตอบสนอง	22
รูปที่ 2.18	แสดงตัวอย่างรูปแบบของชุดคำสั่ง	23
รูปที่ 2.19	แสดงตัวอย่างรูปแบบของชุดผลตอบสนอง	23
รูปที่ 2.20	แสดงการคำนวณหา FCS	23
รูปที่ 2.21	วงจรเซนเซอร์	24
รูปที่ 2.22	แสดงวงจรควบคุมทิศทางการหมุนของมอเตอร์	25
รูปที่ 3.1	Manual Hand	29
รูปที่ 3.2	แสดงการออกแบบแอพพลิเคชั่นเพื่อจัดเก็บสินค้า	30
รูปที่ 3.3	แบบจำลองกระบวนการจัดเก็บสินค้าอัตโนมัติ	31
รูปที่ 4.1	เลือกประเภทการจัดเก็บสินค้าเข้าหรือสินค้าออก	32
รูปที่ 4.2	แสดงผลการทดลองการนำสินค้าเข้า	32
รูปที่ 4.3	แสดงสถานะภายในคลังสินค้าเมื่อจัดเก็บสินค้าที่ช่อง 2	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ(ต่อ)

รูปที่		หน้า
รูปที่ 4.4	การเลือกช่องสินค้าที่ต้องการขนถ่ายสินค้าออก	33
รูปที่ 4.5	แสดงสถานะภายในคลังสินค้า	34
รูปที่ 4.6	การขนถ่ายสินค้าช่องที่ 2 ออก	35
รูปที่ 4.7	ตัวควบคุมการทำงาน PLC	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 2.1 แสดงขาคอนเน็คเตอร์แบบ D-Type	20
ตารางที่ 2.2 แสดงผลการเปรียบเทียบค่าแรงดัน	24
ตารางที่ 2.4 แสดงทิศทางการหมุนของมอเตอร์	25
ตารางที่ 2.3 แสดงสถานะหลอด LED	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมา

เนื่องจากปัจจุบันอุตสาหกรรมส่วนใหญ่ในประเทศไทยได้นำเอาโปรแกรมเมเบิลลอจิกคอนโทรลเลอร์ (Programmable Logic Controller) หรือเรียกว่า " PLC " มาใช้งานในระบบควบคุมอัตโนมัติ ซึ่งมีความน่าเชื่อถือสูงและใช้งานได้ง่าย โดยนำเอา PLC มาใช้ควบคุมการลำเลียงสินค้าเพื่อให้การจัดเก็บสินค้ามีประสิทธิภาพสูงสุด ดังนั้นปริญญานิพนธ์ฉบับนี้ จึงนำเสนอโดยการจำลองกระบวนการจัดเก็บสินค้าขึ้นด้วยระบบ PLC โดยมีการควบคุมการทำงานผ่านทางโปรแกรมบนหน้าจอคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม RS-232C ซึ่งแสดงผลกระบวนการการทำงานโดยจำลองภาพ มีการบันทึกรายการสินค้าที่อยู่ในกระบวนการและสามารถตรวจสอบสินค้าได้อย่างถูกต้อง โดยกระบวนการทั้งหมดจำลองด้วยโปรแกรมวิชวลเบสิก 6 (Visual Basic 6)

1.2 วัตถุประสงค์ของโครงงาน

- เพื่อศึกษาการทำงานของ PLC และ โครงสร้างภายใน PLC
- เพื่อศึกษาการรับส่งข้อมูลกับอุปกรณ์ภายนอกของ PLC
- เพื่อเพิ่มทักษะการเขียนแลดเดอร์ไดอะแกรม (Ladder Diagram)
- เพื่อฝึกฝนประสบการณ์ในการแก้ไขปัญหาที่เกิดขึ้นกับโครงงาน
- เพื่อเพิ่มความรับผิดชอบในหน้าที่ที่ได้มอบหมาย
- ศึกษาการเขียนโปรแกรมวิชวลเบสิก 6 (Visual Basic 6) ติดต่อกับ PLC

1.3 ขอบเขตของโครงงาน

- สร้างแบบจำลองกระบวนการการจัดเก็บสินค้า
- ศึกษาการทำงานของอุปกรณ์ PLC
- ศึกษาการรับส่งข้อมูลกับอุปกรณ์ภายนอกของ PLC
- สร้างหน้าต่างแสดงผลข้อมูลและติดต่อ PLC ด้วยการเขียนโปรแกรมวิชวลเบสิก6 (Visual Basic 6) ผ่านทางพอร์ตอนุกรม RS-232C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- หลักการทำงานโครงสร้างภายใน PLC รวมถึงหลักการการเขียนโปรแกรมควบคุม PLC
- ความรู้ในการติดต่อเพื่อรับ-ส่งข้อมูลระหว่าง PLC กับโปรแกรมวิชวลเบสิก6 (Visual Basic 6) ผ่านทางพอร์ตอนุกรม RS-232C
- ความรู้และทักษะในการออกแบบวงจรรวมถึงการเลือกใช้อุปกรณ์อิเล็กทรอนิกส์ต่างๆให้เหมาะสมกับวงจร

1.5 เนื้อหาโดยสังเขป

เนื้อหาภายในปริญญานิพนธ์ฉบับนี้แบ่งออกด้วยเนื้อหาเป็นบทต่างๆเพื่อความสะดวกและง่ายต่อการศึกษาและทำความเข้าใจในแต่ละบทจะประกอบด้วยเนื้อหาที่สำคัญดังนี้

บทที่ 1 บทนำ

บทที่ 2 หลักการและทฤษฎี ประกอบด้วยเนื้อหาในทางทฤษฎีที่เกี่ยวข้องซึ่งทำให้ผู้อ่านเข้าใจที่เป็นพื้นฐานเสียก่อน อันจะเป็นประโยชน์ต่อการทำความเข้าใจเนื้อหาปริญญานิพนธ์ต่อไป

บทที่ 3 การออกแบบ การสร้างและผลการทดลอง โดยจะกล่าวถึงการสร้าง และการออกแบบวงจรต่างๆ รวมถึงการเขียนวิชวลเบสิก 6 ซึ่งทำให้ผู้อ่านมีความเข้าใจในการทำงานโดยรวมของโครงงานนี้ และขั้นตอนการทดลองและการทดสอบประสิทธิภาพของโปรแกรมเพื่อตรวจสอบว่าโครงงานนี้จะสามารถทำงานได้ตามวัตถุประสงค์หรือไม่

บทที่ 4 ผลการทดลอง โดยจะกล่าวถึงขั้นตอนการทดลองและการทดสอบประสิทธิภาพของโปรแกรมเพื่อตรวจสอบว่าโครงงานนี้จะสามารถทำงานได้ตามวัตถุประสงค์หรือไม่

บทที่ 5 บทวิจารณ์และสรุป เป็นการสรุปผลการปฏิบัติงานและได้เสนอแนะแนวทางการแก้ปัญหา และแนวทางในการพัฒนาโครงงานให้มีประสิทธิภาพและประยุกต์การใช้งานมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
หลักการและทฤษฎี

2.1 ทฤษฎีเบื้องต้นโปรแกรมเมเบิลลอจิกคอนโทรลเลอร์ (PLC)

2.1.1 ประวัติความเป็นมาของ PLC

เมื่อปี พ.ศ. 2511 ในฝ่าย Hydromantic ของบริษัท General Motors ประเทศสหรัฐอเมริกา ได้คิดค้นอุปกรณ์ควบคุมแบบใหม่เพื่อใช้ทดแทนวงจรไฟฟ้าแบบเดิมที่ใช้กันอยู่ภายในโรงงานอุตสาหกรรมของบริษัท และในปี พ.ศ. 2512 PLC ได้ถูกผลิตขึ้นจำหน่ายในประเทศสหรัฐอเมริกาเป็นแห่งแรก ส่วนในประเทศญี่ปุ่น PLC ได้ถูกพัฒนาขึ้นภายหลังจากที่บริษัท ออมรอม (OMRON Co.,Ltd) ประเทศญี่ปุ่นประสบความสำเร็จในการผลิตโซลิต-สเตทรีเลย์ (Solid State Relay) ในปี พ.ศ.2508 หลังจากนั้น 5 ปี PLC ก็ถูกนำออกจำ หน่ายสู่ท้องตลาดจนเป็นที่แพร่หลายในเวลาต่อมา ชื่อเรียกที่แตกต่างกันของ PLCซึ่ง PLC ของแต่ละบริษัทจะมีชื่อเรียกแตกต่างกันในแต่ละประเทศ ดังนี้

- ในประเทศอังกฤษ เรียกว่า PC หรือ Programmable Controller (โปรแกรมเมเบิลคอนโทรเลอร์)

-ในประเทศกลุ่มสแกนดิเนเวียเรียกว่า PBS หรือ Programmable Binary System (โปรแกรมเมเบิลไบนารี่ซิสเทม)

- ในประเทศสหรัฐอเมริกา เรียกว่า PLC หรือ Programmable Logic Controller (โปรแกรมเมเบิลลอจิกคอนโทรเลอร์)

2.1.2 โครงสร้างโดยทั่วไปและคุณสมบัติของ PLC รุ่น CQM 1

โครงงานนี้ใช้โปรแกรมเมเบิลคอนโทรลเลอร์ (PLC) รุ่น CQM 1 บริษัท ออมรอน จำกัด (OMRON Co,Ltd) ซึ่งเป็น PLC ชนิดโมดูล (Modular Type PLCs) PLC ชนิดนี้ส่วนประกอบแต่ละส่วนสามารถแยกออกจากกันเป็นโมดูล โดยใช้คอนเน็คเตอร์ในการเชื่อมต่อแต่ละโมดูลเข้าด้วยกัน เพื่อให้สามารถทำงานร่วมกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

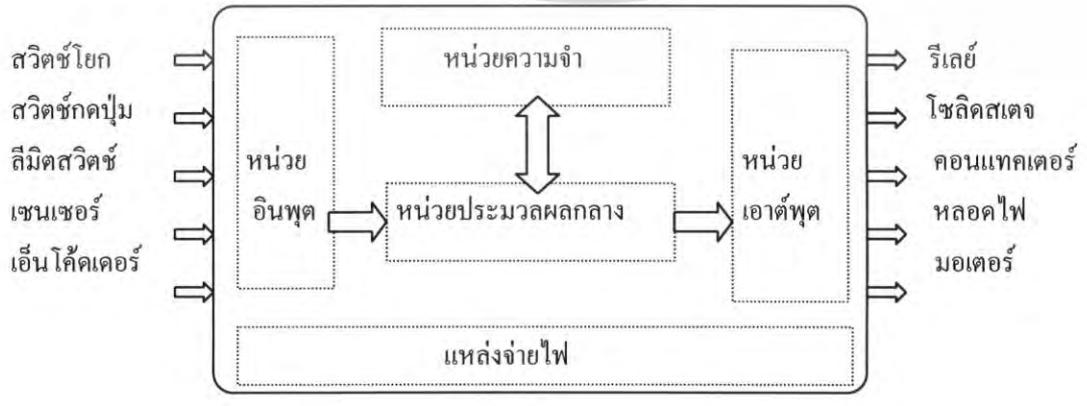
รูปที่ 2.1 PLC ชนิดโมดูล

2.1.2.1 คุณสมบัติของ PLC รุ่น CQM 1

- มีจำนวนอินพุต / เอาต์พุต สามารถขยายได้สูงสุด 512 จุด
- มีหน่วยความจำในการโปรแกรม เท่ากับ15 kWords
- มีตัวหน่วงเวลา (Timer) และตัวนับ (Counter) รวมกันเท่ากับ 512 ตัว
- ความเร็วในการประมวลผล 0.375 μs ต่อคำสั่ง
- ใช้กับแหล่งจ่ายไฟฟ้ากระแสสลับตั้งแต่ 100-240 VAC พร้อมทั้งมีไฟฟ้ากระแสตรง 24 VDC

2.1.2.2 ส่วนประกอบที่สำคัญ แบ่งออกเป็น 5 ส่วนคือ

- หน่วยประมวลผล (CPU)
- หน่วยความจำ (Memory)
- หน่วยอินพุต / เอาต์พุต (Input / Output Unit)
- อุปกรณ์ต่อร่วม (Peripheral Devices)
- แหล่งจ่ายพลังงานไฟฟ้า (Power Supply)



รูปที่ 2.2 แสดงส่วนประกอบภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. หน่วยประมวลผล (CPU)

ทำหน้าที่ คำนวณและควบคุมเปรียบเสมือนสมองของ PLC ภายใน ประกอบด้วยวงจรลอจิกหลายชนิด มีไมโครโปรเซสเซอร์เบสใช้แทนอุปกรณ์จำพวกรีเลย์ (Relay) เคาน์เตอร์ (Counter) ไทม์เมอร์(Timer) และซีเควนเซอร์(Sequencer) เพื่อให้ผู้ใช้สามารถออกแบบวงจรโดยใช้แลดเดอร์ไดอะแกรม (Ladder Diagram) ได้ CPU จะยอมรับข้อมูลจากอุปกรณ์อินพุตต่างๆ จากนั้นจะทำการประมวลผลและเก็บข้อมูลโดยใช้โปรแกรมจากหน่วยความจำ หลังจากนั้นจะส่งข้อมูลที่เหมาะสมและถูกต้องออกไปยังอุปกรณ์เอาต์พุต

2. หน่วยความจำ (Memory)

ทำหน้าที่ เก็บรักษาโปรแกรมและข้อมูลที่ใช้ในการทำงาน โดยขนาดของหน่วยความจำจะถูกแบบออกเป็นบิตข้อมูล (Data Bit) ภายในหน่วยความจำ 1 บิต ก็จะมีค่าสภาวะทางลอจิก 0 หรือ 1 แตกต่างกันออกไป ซึ่ง PLC ประกอบด้วยหน่วยความจำสองชนิดคือ ROM และ RAM

- **ROM** (Read Only Memory) เป็นหน่วยความจำที่ไม่ยอมให้ผู้ใช้เปลี่ยนข้อมูล และสามารถเก็บรักษาข้อมูลได้แม้ว่าจะไม่มีกระแสไฟฟ้า เหมาะสำหรับเก็บโปรแกรมบริหารระบบหรือโปรแกรมมอนิเตอร์ ราคาถูก มีความเร็วและความน่าเชื่อถือสูง

- **RAM** (Random Access Memory) เป็นหน่วยความจำที่ยอมให้ผู้ใช้เปลี่ยนข้อมูลได้ เหมาะสำหรับเก็บโปรแกรมผู้ใช้ที่อยู่ในช่วงกำลังพัฒนา หรือต้องการเปลี่ยนแปลงบ่อยครั้ง โดยมีหน่วยจ่ายกำลังสำรอง (Battery backup) เพื่อรักษาข้อมูล เป็นหน่วยความจำที่มีความเร็วสูง

รูปที่ 2.3 แสดงหน่วยประมวลผล (CPU) และหน่วยความจำ (Memory)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. หน่วยอินพุต / เอาต์พุต (Input / Output Unit)

- หน่วยอินพุต ทำหน้าที่รับข้อมูลเข้ามา จากนั้นจะทำการส่งข้อมูลต่อไปยัง CPU เพื่อนำไปประมวลผลต่อไปโดยข้อมูลที่รับเข้ามาเป็นสัญญาณอินพุต เช่น เซนเซอร์ ลิมิตสวิตช์ เป็นต้น

- หน่วยเอาต์พุต ทำหน้าที่ รับข้อมูลจากหน่วยประมวลผลแล้วส่งต่อข้อมูลต่อไปควบคุมอุปกรณ์ภายนอก เช่น มอเตอร์ หลอดไฟ เป็นต้น

รูปที่ 2.4 แสดงหน่วยอินพุต/เอาต์พุต (Input / Output Unit)

4. อุปกรณ์ต่อร่วม (Peripheral Devices)

ทำหน้าที่ ควบคุมโปรแกรมของผู้ใช้ลงในหน่วยความจำของ PLC นอกจากนั้นแล้วยังทำหน้าที่ติดต่อระหว่างผู้ใช้กับ PLC เพื่อให้ผู้ใช้สามารถตรวจการปฏิบัติงานของ PLC และผลการควบคุมเครื่องจักรและกระบวนการตามโปรแกรมควบคุมที่ผู้ใช้เขียนขึ้นได้อีกด้วย ซึ่งสามารถแบ่งอุปกรณ์ที่ใช้ต่อร่วมออกได้ เป็น 2 ประเภท

- เครื่องป้อนโปรแกรมแบบมือถือ (Hand Held Programmer)

ซึ่งการเขียนโปรแกรมให้กับ PLC โดยการใช้ Hand Held Programmer ภาษาที่ใช้จะเป็นภาษา Statement List เช่นคำสั่ง LD, AND, OR ซึ่งเป็นคำสั่งพื้นฐานสามารถเรียกใช้งานโดยการกดปุ่มที่อยู่ที่ตัว Hand Held Programmer นั้น แต่เมื่อต้องการใช้งานฟังก์ชันอื่นๆที่มีอยู่ใน PLC สามารถเรียกใช้งานได้โดยการกดปุ่มเรียกใช้คำสั่งพิเศษ ซึ่งวิธีการใช้งานต้องศึกษาจากคู่มือของยี่ห้อนั้นๆต่อไป

ข้อดี มีความสะดวกในการเคลื่อนย้าย พกพาได้สะดวก มีขนาดเล็ก

ข้อเสีย ในการใช้งานผู้ใช้ต้องศึกษาวิธีการใช้งานว่ามีวิธีการกดอย่างไร ถึงจะสั่งงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

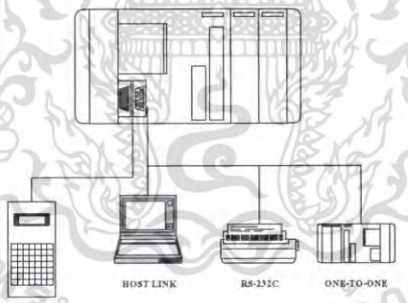
รูปที่ 2.5 แสดงตัวป้อนโปรแกรมแบบมือถือ (Hand Held Programmer)

- คอมพิวเตอร์ (Computer)

คอมพิวเตอร์สามารถใช้ในการเขียนโปรแกรมให้กับ PLC ได้โดยใช้งานร่วมกับซอฟแวร์ (Software) เฉพาะของ PLC ยี่ห้อนั้น เช่น PLC ของ OMRON จะใช้ซอฟแวร์ที่มีชื่อเรียกแตกต่างกันไปสามารถยกตัวอย่างได้เช่น Syswin Support Software และ CX-Programmer ใช้ได้กับระบบปฏิบัติการตั้งแต่ Window 95 ขึ้นไป

ข้อดี ใช้งานง่ายเป็นการนำสัญลักษณ์ต่างๆมาใช้แทนการเขียนคำสั่ง

ข้อเสีย มีขนาดใหญ่



รูปที่ 2.6 แสดงการต่อใช้งานร่วมกับ PLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. หน่วยจ่ายพลังงาน (Power Supply)

ทำหน้าที่จ่ายพลังงานและรักษาระดับแรงดันไฟฟ้ากระแสตรงให้กับอุปกรณ์อิเล็กทรอนิกส์ในหน่วยประมวลผล หน่วยความจำและหน่วยอินพุต/เอาต์พุต

รูปที่ 2.7 แสดงหน่วยจ่ายพลังงาน (Power Supply)

2.1.3 หลักการทำงาน

PLC เป็นอุปกรณ์ชนิดโซลิด-สเตท (Solid State) ที่ทำงานแบบลอจิก (Logic Functions) การออกแบบการทำงานของ PLC คล้ายกับหลักการทำงานของคอมพิวเตอร์ทั่วไป ซึ่งจากหลักการพื้นฐานแล้ว PLC จะประกอบด้วยอุปกรณ์ที่เรียกว่าSolid-State Digital Logic Elements เพื่อให้ทำงานและติดสินใจแบบลอจิก PLC ใช้สำหรับควบคุมกระบวนการทำงานของเครื่องจักรและอุปกรณ์ในโรงงานอุตสาหกรรมการใช้ PLC ควบคุมเครื่องจักรหรืออุปกรณ์ต่างๆ ในโรงงานอุตสาหกรรมจะมีข้อได้เปรียบกว่าการใช้ระบบรีเลย์ (Relay) ซึ่งจำเป็นจะต้องเดินสายไฟฟ้า หรือที่เรียกว่า Hard-Wired ฉะนั้นเมื่อมีความจำเป็นที่ต้องเปลี่ยนกระบวนการผลิตหรือลำดับการทำงานใหม่ ก็ต้องทำการเดินสายไฟฟ้าใหม่ ซึ่งเสียเวลาและเสียค่าใช้จ่ายสูง แต่เมื่อเปลี่ยนมาใช้ PLC แล้วการเปลี่ยนกระบวนการผลิตหรือลำดับการทำงานใหม่นั้น ทำได้โดยการเปลี่ยนโปรแกรมใหม่เท่านั้นนอกจากนี้แล้ว PLC ยังใช้ระบบโซลิด-สเตท ซึ่งน่าเชื่อถือกว่าระบบเดิม การกินกระแสไฟฟ้าน้อยกว่า และสะดวกกว่าเมื่อต้องขยายขั้นตอนการทำงานของเครื่องจักร

2.1.4 การทำงานของหน่วยต่างๆ ภายใน PLC

ส่วนของการประมวลผลกลางหรือ CPU (Control Processing Unit) ทำได้โดยรับข้อมูลมาจากหน่วยอินพุต-เอาต์พุต และส่งข้อมูลสุดท้ายที่ได้จากการประมวลผลไปยังหน่วยเอาต์พุต เรียกว่าการสแกน (Scan) ซึ่งใช้เวลาจำนวนหนึ่งเรียกว่า เวลาสแกน (Scan Time) เวลาในการสแกนแต่ละรอบใช้เวลาประมาณ 1 ถึง 100 msec. (10 msec. = 100 ครั้งต่อวินาที) ทั้งนี้ขึ้นอยู่กับข้อมูลและความยาวของโปรแกรมหรือจำนวนอินพุตและเอาต์พุตหรือจำนวนอุปกรณ์ที่ต่อจาก PLC เช่น จอภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องพิมพ์ เป็นต้น อุปกรณ์เหล่านี้จะทำให้เวลาในการสแกนยาวนานขึ้น การเริ่มต้นการสแกนเริ่มจากรับค่าของสภาวะของอุปกรณ์จากหน่วยความจำ (Memory) เสร็จแล้วจะทำการปฏิบัติ การตามโปรแกรมที่เขียนไว้ทีละคำสั่งจากหน่วยความจำนั้นจนสิ้นสุดแล้วส่งไปที่หน่วยเอาต์พุต

ส่วนของอินพุตและเอาต์พุต (I/O Unit) จะต่อร่วมกับชุดควบคุมเพื่อรับสภาวะและสัญญาณต่างๆ เช่น หน่วยอินพุตรับสัญญาณหรือสภาวะแล้วส่งไปยัง CPU เพื่อประมวลผล เมื่อ CPU ประมวลผลแล้วจะส่งให้ส่วนของเอาต์พุต เพื่อให้อุปกรณ์ทำงานตามโปรแกรม ที่กำหนดไว้ สัญญาณอินพุตจากภายนอกที่เป็นสวิตช์และตัวตรวจจับชนิดต่างๆ จะถูกแปลงให้เป็นสัญญาณที่เหมาะสมถูกต้อง ไม่ว่าจะเป็น AC หรือ DC เพื่อส่งให้ CPU ดังนั้น สัญญาณเหล่านี้จึงต้องมีความถูกต้องไม่เช่นนั้นแล้ว CPU จะเสียหายได้ สัญญาณอินพุตที่ดีจะต้องมีคุณสมบัติและหน้าที่ดังนี้

1. ทำให้สัญญาณเข้า ได้ระดับที่เหมาะสมกับ PLC

2. การส่งสัญญาณระหว่างอินพุตกับ CPU จะติดต่อกันด้วยลำแสง ซึ่งอาศัยอุปกรณ์ประเภทโฟโตทรานซิสเตอร์เพื่อต้องการแยกสัญญาณ (Isolate) ทางไฟฟ้าให้ออกจากกัน เป็นการป้องกันไม่ให้ CPU เสียหายเมื่ออินพุตเกิดลัดวงจร

3. หน้าสัมผัสจะต้องไม่สั่นสะเทือน (Contact Chattering) ในส่วนของเอาต์พุต จะทำหน้าที่รับค่าสะภาวะที่ได้จากการประมวลผลของ CPU แล้วนำค่าเหล่านี้ไปควบคุมอุปกรณ์ทำงาน เช่น รีเลย์ โซลีนอยด์ หรือหลอดไฟ เป็นต้น นอกจากนั้นแล้วยังทำหน้าที่แยกสัญญาณของหน่วยประมวลผลกลาง (CPU) ออกจากอุปกรณ์เอาต์พุต โดยปกติเอาต์พุตนี้จะมีความสามารถขับโหลดด้วยกระแสไฟฟ้าประมาณ 1-2 แอมแปร์ แต่ถ้าโหลดต้องการกระแสไฟฟ้ามากกว่านี้ จะต้องต่อเข้ากับอุปกรณ์ขับอื่นเพื่อขยายให้รับกระแสไฟฟ้ามากขึ้นเช่น รีเลย์ หรือคอนแทคเตอร์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 โปรแกรมวิชวลเบสิก6 (Visual Basic 6.0)

2.2.1 ที่มาและลักษณะของโปรแกรมวิชวลเบสิก6 Visual Basic 6

Visual Basic เป็นเครื่องมือที่ใช้พัฒนาโปรแกรมแบบ Visual Programming ซึ่ง Visual Programming เป็นวิธีการเขียนโปรแกรมที่มีเครื่องมือช่วยพัฒนาโปรแกรมได้ง่าย โดยโปรแกรมที่สร้างจะมีคุณ สมบัติและโค้ดกำกับซึ่งจะช่วยให้การพัฒนาโปรแกรมสามารถทำได้อย่างรวดเร็ว สำหรับการใช้โปรแกรม Visual Basic นั้นไม่ได้จำกัดตัวเองอยู่เพียงแค่บน PC ที่ใช้ Windows เท่านั้น แต่ยังขยายความสามารถไปยังการใช้บน Internet รวมทั้งอุปกรณ์ที่ใช้เทคโนโลยีไร้สาย

2.2.2 จุดเด่นของ Visual Basic

1. สร้างแอพพลิเคชั่นได้ง่ายและรวดเร็ว Visual Basic ได้รับการวางตัวให้เป็นเครื่องมือที่ช่วยให้สร้างแอพพลิเคชั่นได้อย่างรวดเร็วและง่ายดาย เพื่อลดเวลาในการสร้างแอพพลิเคชั่นให้สั้นลง ซึ่งเรียกรูปแบบนี้ว่า Rapid Application Development หรือ RAC ทั้งนี้เพราะมีการจัดงานที่โปรแกรมเมอร์ต้องทำซ้ำๆซากๆ ออกไปเหลือเฉพาะที่ต้องโฟกัสเกี่ยวกับปัญหาของงานจริงๆแล้วเขียนโปรแกรมจัดการปัญหานั้นๆส่วนเรื่องอื่นๆเหลือให้ Visual Basic จัดการ

2. ถ้ามีโอกาสเขียนโปรแกรมด้วย Visual Basic แล้วจะเห็นว่าภาษา Basic ที่ใช้ใน Visual Basic นั้นเข้าใจง่ายคือใกล้เคียงกับภาษาที่เราใช้งานปกติสื่อความหมายและเข้าใจง่ายกว่าภาษาของโปรแกรมอื่นๆ ทำให้ผู้เริ่มต้นเขียนโปรแกรมทำความเข้าใจกับการเขียนโปรแกรมได้อย่างรวดเร็ว

3. รวมเครื่องมืออำนวยความสะดวกในการเขียนโปรแกรมนอกจากจะง่ายต่อการเรียนรู้แล้ว Visual Basic ยังมีเครื่องมือที่ช่วยในการเขียนโปรแกรมเป็นเรื่องที่ไม่ยุ่งยาก เพราะจะมีเครื่องมือที่ช่วยให้ไม่ต้องจดจำไวยากรณ์ภาษาที่ยุ่งยาก มีการตรวจสอบอัตโนมัติว่าโปรแกรมที่เขียนนั้นถูกต้องตามหลักภาษาหรือไม่มีการแยกแยะส่วนของโปรแกรมอย่างเป็นระเบียบทำให้งานของโปรแกรมเมอร์นั้นลดลงได้มาก นอกจากจะมีเครื่องมือช่วยในการเขียนโปรแกรมแล้วยังมีเครื่องมือที่ใช้ในการทดสอบแก้ไขโปรแกรม(Debugger)ที่เขียนขึ้นมาว่าทำงานได้ถูกต้องมีระบบขอความช่วยเหลือ(Online Help) ไว้อ้างอิงและขอความช่วยเหลือในจุดที่เราสงสัยข้องใจได้ เครื่องมือทั้งหมดที่กล่าวมาถูกจัดรวมไว้ในสภาพแวดล้อมการทำงานเดียวกัน(เรียกย่อๆว่า IDE ซึ่งย่อมาจาก Integral Development Environment) ทำให้เรียกใช้งานได้สะดวกตั้งแต่เขียนโปรแกรม ทดสอบแก้ไข สร้างชุดติดตั้ง รวมทั้งระบบขอความช่วยเหลือ ซึ่งเราสามารถเพิ่มเติมเครื่องมือชนิดใหม่ๆเข้าไปได้เรื่อยๆหรือถอดเครื่องมือที่ไม่จำเป็นต้องใช้เพื่อประหยัดพื้นที่ฮาร์ดดิสก์ก็ได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 รูปแบบการพัฒนาแอพพลิเคชั่นกับ Visual Basic

เมื่อเรามองเห็นว่า Visual Basic จะช่วยให้เราสร้างแอพพลิเคชั่นบน Windows ได้ง่ายและรวดเร็วแล้วยังมีรูปแบบที่ Visual basic สามารถสร้างขึ้นมาได้อีกคือ

1. พัฒนาแอพพลิเคชั่นกับ Active X Controlเทคโนโลยีที่มีชื่อว่า Active X เป็นตัวอยู่เบื้องหลังความสำเร็จของ Visual Basic ซึ่งช่วยลดงานที่ซ้ำซ้อนของการเขียนโปรแกรมลงไปมาก ตัวอย่างเช่น การเขียนโปรแกรมเพื่อรับค่าข้อมูลของผู้ใช้แต่ก่อนเราต้องเขียนโปรแกรมเพื่อวาดหน้าจอ เขียนโปรแกรมวาดรูปปุ่มและช่องรับข้อความรวมทั้งการเขียนโปรแกรมเพื่อจัดการกับข้อมูลที่ผู้ใช้ป้อนเข้ามาแต่ Active X จะทำให้เราสนใจเฉพาะการจัดการกับข้อมูลที่ผู้ใช้ป้อนเข้ามาเท่านั้นส่วนที่เหลือจะจัดการให้เองโดยช่องรับข้อความและปุ่มต่างๆจะใช้ Active X control จัดการนอกจากจะลดความซับซ้อนลงแล้วการที่เราใช้ Active X Control ในการเขียนโปรแกรมยังช่วยให้โปรแกรมที่เราเขียนกับโปรแกรมที่คนอื่นๆเขียนนั้นตั้งอยู่บนมาตรฐานเดียวกัน ทำให้การบำรุงรักษาง่ายเพราะใครๆก็เข้าใจมาตรฐานของ Active X Control นี้ ทำให้ไม่ต้องกังวลใจว่าโปรแกรมที่เขียนขึ้นจะมีเฉพาะคนเขียนเท่านั้นที่เข้าใจ

2. สร้างแอพพลิเคชั่นที่ใช้งานกับฐานข้อมูล เป็นแอพพลิเคชั่นที่มีการใช้งานมากที่สุดเพราะระบบต่างๆเช่น ร้านค้า คลังสินค้า ระบบบัญชี ระบบบริหารงานบุคคลหรือแม้แต่ E-Commerce ทั้งหมดต่างก็ต้องมีส่วนที่ติดต่อกับฐานข้อมูล VisualBasic ได้ช่วยในการสร้างแอพพลิเคชั่นกับฐานข้อมูลเป็นเรื่องที่ทำได้โดยง่ายดาย เพราะมีเครื่องมือต่างๆที่ค่อยให้ความสะดวกในการเขียนโปร-แกรม เพื่อใช้งานข้อมูลจากฐานข้อมูลซึ่งไม่กำจัดด้วยว่าจะเห็นข้อมูลแบบใดทั้งฐานข้อมูลส่วนบุค-คลฐานข้อมูลผ่านเครือข่ายหรือฐานข้อมูลผ่านอินเตอร์เน็ต จากความสามารถที่หลากหลายนี้ Visual Basic จึงเป็นตัวเลือกอันดับต้นๆ ของการสร้างแอพพลิเคชั่นที่เกี่ยวกับงานบานข้อมูลในตลาดธุรกิจ

3. สร้างแอพพลิเคชั่นแบบใหม่กับอินเตอร์เน็ต อินเตอร์เน็ตนับว่ามีความสำคัญกับชีวิตของคนที่ใช้ไอทีมากขึ้นทุกวัน ซึ่ง Visual Basic เปิดโอกาสให้เราสามารถสร้างแอพพลิเคชั่นเพื่อรองรับการทำงานร่วมกับอินเตอร์เน็ตได้ด้วยการใช้ความรู้เดิมที่เรามีอยู่แล้วจากการสร้างแอพพลิเคชั่นปกติ รวมทั้งเครื่องมือเสริมการทำงานต่างๆมากมายจุดเด่นของ Visual Basic อีกข้อหนึ่งคือเปิดโอ-กาสให้เรานำแอพพลิเคชั่นปกติที่เดิมทำงานกับคอมพิวเตอร์ส่วนบุคคลสามารถดัดแปลงใช้งานกับอินเตอร์เน็ตได้อย่างไม่ยากเย็นนัก ทำให้ไม่ต้องเปลี่ยนแปลงรูปแบบงานเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 รายละเอียดของส่วนประกอบต่างๆของหน้าจอ

เมื่อเราเปิดใช้งานวิชวลเบสิก 6.0 ก็จะพบกับลักษณะการทำงานที่เรียกว่า IDE (Integrated Development Environment) คือรวบรวมเครื่องไม้เครื่องมือ ข้อมูลที่ใช้งานต่างๆไว้ในหน้าจอเดียว ทำให้เรียกใช้งานได้ง่าย

1. Menu Bar เก็บคำสั่งที่สามารถใช้งานได้ทั้งหมดใน Visual Basic 6 ประกอบด้วยเมนูทำงาน File, View, Windows

2. Toolbar ประกอบด้วยปุ่มคำสั่งต่างๆที่ช่วยให้งานคำสั่งของ Visual Basic 6.0 ได้อย่างรวดเร็ว

3. Toolbox เป็นกล่องเก็บ ActiveX Control ซึ่งเราจะนำมาประกอบเป็นส่วนต่างๆของแอพพลิเคชั่น

4. Project Explorer เป็นเครื่องมือที่ใช้ควบคุมการทำงานของโปรเจ็กต์

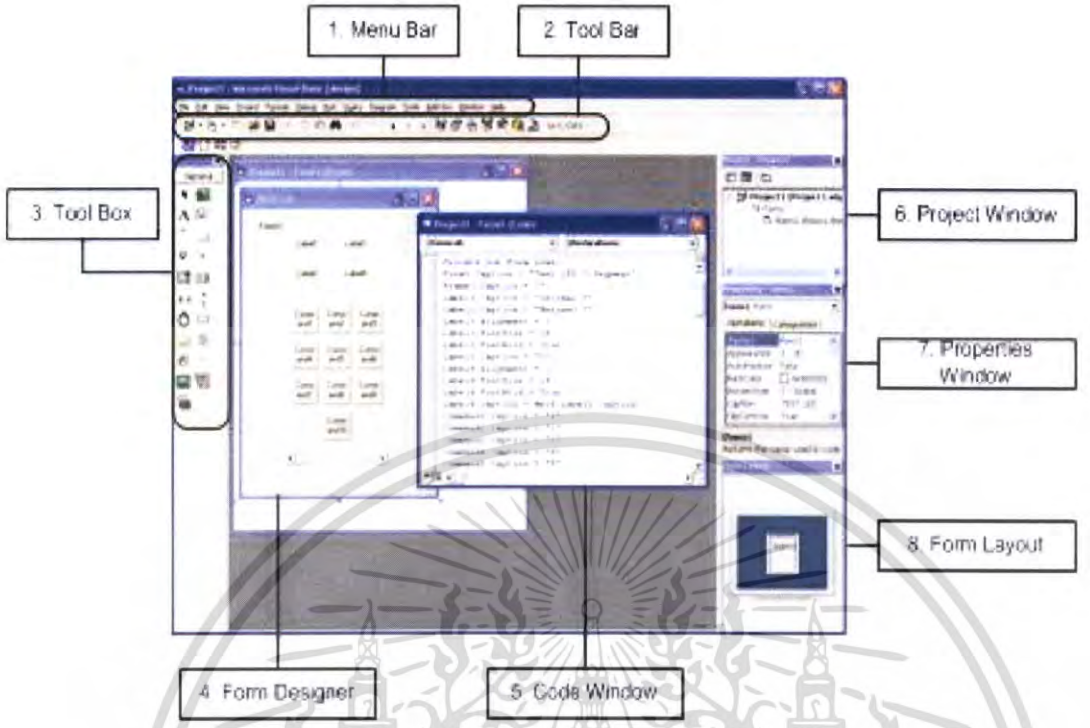
5. Properties windows เป็นส่วนกำหนดพร็อพเพอร์ตี้ให้กับออบเจ็กต์ต่างๆในแอพพลิเคชั่น

6. From Layout เป็นรูปหน้าต่างคร่าวๆ ของฟอร์มที่ได้จากการรันแอพพลิเคชั่น ทำให้เราทราบตำแหน่งที่ปรากฏบนจอภาพเมื่อแอพพลิเคชั่นทำงาน

7. From Designer ซึ่งเป็นส่วนที่เรามองเห็นในขณะออกแบบแอพพลิเคชั่นของ Visual Basic 6.0 ซึ่งเราออกแบบหน้าตาของแอพพลิเคชั่นผ่านฟอร์มดิไซเนอร์

8. Code Window เป็นส่วนที่เราทำการเขียนโปรแกรมเพื่อควบคุมการทำงานของแอพพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงองค์ประกอบต่างๆของโปรแกรมวิชวล เบสิก 6

2.2.5 เริ่มสร้างแอพพลิเคชั่นด้วย Visual Basic

1. ออกแบบแอพพลิเคชั่นต้องทราบให้แน่ชัดก่อนว่าแอพพลิเคชั่นที่เราจะทำการสร้างนั้นจะนำไปใช้ประโยชน์อะไรต้องมีความสำคัญอะไรบ้าง และจะต้องมีรูปร่างหน้าตาเป็นอย่างไร ซึ่งจำเป็นอย่างยิ่งที่ต้องคิดให้รอบคอบและเขียนออกมาให้ชัดเจนโดยอาจจะร่างง่ายๆในกระดาษ

2. ตกแต่งหน้าตาแอพพลิเคชั่นเป็นการตกแต่งรูปร่างของแอพพลิเคชั่นที่ได้ออกแบบไว้แล้วพร้อมๆกับการคำนวณค่าพร็อพเพอร์ตี้ต่างๆให้กับคอนโทรลแต่ละตัวในแอพพลิเคชั่น

3. เขียนโค้ดกำกับแอพพลิเคชั่นหลังจากตกแต่งหน้าตาเสร็จแล้วขั้นตอนต่อไปคือการเขียนโค้ดหรือการเขียนโปรแกรมเพื่อควบคุมการทำงานต่างๆ โดยเรามักจะใช้การเขียนโปรแกรมแบบ Event Driven programming ซึ่งจะเป็นการเขียนโค้ดเพื่อรองรับกับเหตุการณ์ต่างๆที่จะเกิดขึ้น กับคอนโทรลในแอพพลิเคชั่นของเรา

4. ทดสอบการทำงานของแอพพลิเคชั่น เมื่อเขียนโค้ดเสร็จแล้วก็ถึงเวลาที่จะทดสอบการทำงานของแอพพลิเคชั่นที่เราสร้างขึ้นซึ่งประกอบไปด้วยคอนโทรลต่างๆ

5. บันทึกเก็บไว้ในคอมพิวเตอร์ หลังจากทดสอบจนแน่ใจแล้วว่าแอพพลิเคชั่นที่สร้างนั้นทำงานได้ถูกต้องเราจึงบันทึกเก็บไว้ ซึ่งสามารถแก้ไขและเพิ่มเติมความสามารถอื่นๆได้ภายหลัง

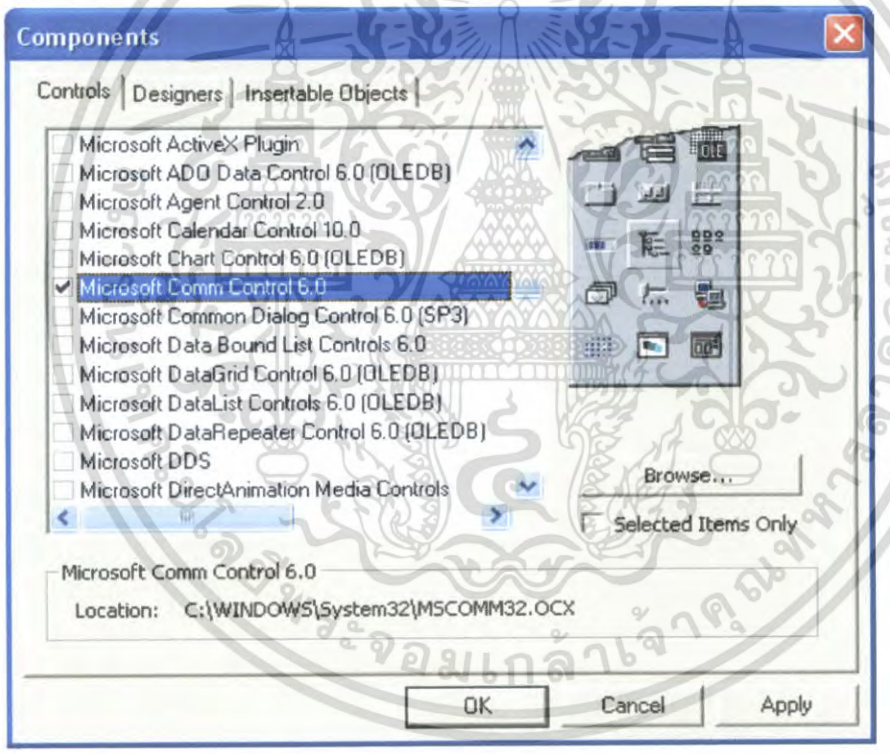
6. สร้างไฟล์ .EXE (make) เมื่อสร้างแอพพลิเคชั่นเสร็จแล้ว หากต้องการนำแอพพลิเคชั่นที่สร้างขึ้นให้เรียกใช้งานได้เอง โดยไม่ต้องเปิดผ่านโปรแกรม Visual Basic หรือต้องการนำไปใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำไปใช้งานในเครื่องคอมพิวเตอร์เครื่องอื่นๆเราสามารถทำได้โดยสร้างไฟล์เอ็กซ์คิวต์ (ไฟล์ที่มีนามสกุลเป็น .EXE)

2.2.6 การสื่อสารกับ Communication Port

Visual Basic ไม่สามารถจัดการกับ Hardware ได้โดยตรงเหมือนภาษา C หรือภาษาอื่นๆ อย่างไรดี Visual Basic ได้เตรียม Tool และ Control ต่างๆสำหรับจัดการกับ Hardware โดยเฉพาะไว้ให้ผู้เขียนโปรแกรมเท่าที่จำเป็น ในการทำงานกับพอร์ตสื่อสารบนเครื่อง PC ซึ่ง Visual Basic มี Control ชื่อ Microsoft Comm Control 6.0 สำหรับการรับส่งข้อมูลทางพอร์ตอนุกรมซึ่งสำคัญมากในโครงงานนี้ โดยสามารถเพิ่มเติมเข้ามาไว้ใน Toolbox โดยเลือกเมนู Project >Component...จะปรากฏหน้าจอดังรูปที่ 2.9



รูปที่ 2.9 แสดงการเปิดใช้งาน Microsoft Comm Control 6.0

การใช้ MsComm Control ซึ่ง MsComm จัดเตรียมทางเลือกเอาไว้ 2 ทาง เพื่อความสะดวกในการสื่อสารข้อมูล

- ทางเลือกแรกคือ การสื่อสารข้อมูลที่กระตุ้นด้วยเหตุการณ์ (event-drive communication) เป็นรูปแบบการใช้งานที่มีประสิทธิภาพมากสำหรับตอบสนองแบบทันทีทันใด เช่น เมื่อตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถูกส่งมาที่พอร์ตอนุกรมหรือเกิดการเปลี่ยนแปลงที่ขา Data Carrier (DCD) หรือขา Request To Send เหตุการณ์ OnComm ของ MSComm จะสามารถตรวจจับสัญญาณนั้นได้ทันที่

-ทางเลือกที่สองเป็นการคอยตรวจสอบค่าเหตุการณ์และความผิดพลาดที่เกิดขึ้นด้วยค่าที่เปลี่ยนแปลงในคุณสมบัติ CommEvent หลังจากให้โปรแกรมทำงานฟังก์ชันต่างๆ ไปเรียบร้อยแล้ว ซึ่งวิธีนี้ใช้งานได้ดีในกรณีที่โปรแกรมขนาดเล็ก

2.2.7 คุณสมบัติ (*Property*) ของMSComm

1. CommPort ใช้ในการกำหนดและอ่านค่าพอร์ตอนุกรมที่ติดต่ออยู่ (Com 1, Com 2, Com3, Com4)

โดยมีรูปแบบการใช้งาน คือ

Object.CommPort [=value]

โดย Value เป็นค่าของพอร์ตอนุกรม ชนิดของข้อมูลเป็น Integer ค่า Value สามารถกำหนดได้ในช่วง 1-16 (ค่าเริ่มต้นกำหนดไว้ที่ I) เมื่อมีการกำหนดค่าแล้วทำการเปิดพอร์ต โดยใช้คุณสมบัติ Port Open แต่ว่าพอร์ตนั้นไม่มีอยู่ในระบบ MSComm จะสร้างสัญญาณแสดงข้อผิดพลาด error 68 ขึ้นมา ซึ่งหมายถึง อุปกรณ์ตัวนี้ไม่มีอยู่ในระบบ ดังนั้นการเขียนโปรแกรมจึงจำเป็นต้องกำหนดตำแหน่งของพอร์ตอนุกรม ก่อนที่ใช้คำสั่ง Open Port

2. Settingใช้ในการกำหนดและอ่านค่าอัตราบอด, พาริตี้, จำนวนของบิตข้อมูล, จำนวนของบิตปิดท้ายรูปแบบการใช้งาน

Object. Setting [=value]

ค่า Value มีชนิดของข้อมูลเป็นแบบ String มีรูปแบบเป็น “BBBB, P, D, S” โดย BBBB เป็นค่าอัตราบอด, P เป็นค่าพาริตี้บิต, D เป็นจำนวนของบิตข้อมูลและ S เป็นจำนวนของบิตปิดท้าย ปกติแล้วค่านี้ถูกกำหนดไว้เป็น “9,600, N, 8, 1”

ค่าบอดเรตมาตรฐานที่ใช้กับ MSComm มีดังนี้

110	บิตต่อวินาที	
300	บิตต่อวินาที	
600	บิตต่อวินาที	
1,200	บิตต่อวินาที	
2,400	บิตต่อวินาที	
9,600	บิตต่อวินาที	(ค่าปกติ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14,400	บิตต่อวินาที	
19,200	บิตต่อวินาที	
28,800	บิตต่อวินาที	
38,400	บิตต่อวินาที	(สงวน)
56,000	บิตต่อวินาที	(สงวน)
128,000	บิตต่อวินาที	(สงวน)
256,000	บิตต่อวินาที	(สงวน)

สำหรับค่ามาตรฐานในการกำหนดค่าพาริตี้มีดังนี้

สัญลักษณ์	รายละเอียด
E	พาริตี้คู่ (Even)
M	ลอจิก "1 (Mark)
N	ไม่ใช้ (ค่าปกติ)
O	พาริตี้คี่ (Odd)
S	ลอจิก "0 (Space)

ค่าที่ใช้ในการทำกำหนดจำนวนบิตมี 5 ค่า คือ 4,5,6,7 และ 8 (เป็นค่าปกติ)

ค่าที่ระบุจำนวนบิตปิดท้ายมี 3 ค่า คือ 1 (เป็นค่าปกติ), 1.5 และ 2

3. Port Open ใช้ในการกำหนดและอ่านค่าสถานะของพอร์ตอนุกรม เพื่อเปิดและปิดพอร์ตอนุกรมรูปแบบการในงาน

Object.PortOpen [=Value]

ค่า Value มีชนิดข้อมูลเป็นแบบบูลีนคือ True กับ False โดนTrue หมายถึงการเปิดพอร์ตอนุกรมและ false หมายถึงการปิดพอร์ตอนุกรม สำหรับการปิดพอร์ตอนุกรม สำหรับการปิดพอร์ตนั้นจะมีการเคลียร์บัฟเฟอร์รับข้อมูลและบัฟเฟอร์ส่งข้อมูลด้วย คอนโทรล MSComm จะปิดพอร์ตอนุกรมโดยอัตโนมัติเมื่อออกจากโปรแกรม ก่อนที่ใช้คุณสมบัติ Port Open ต้องตรวจสอบให้แน่ใจก่อนว่าคุณสมบัติ CommPort นั้นได้ทำการกำหนดตำแหน่งของพอร์ตอนุกรมไว้ถูกต้องหรือไม่ มิเช่นนั้น MSComm จะแสดงผิดพลาด error 68 แจ้งแก่ผู้ใช้งาน หรือพอร์ตอนุกรมนั้นถูกเปิดเอาไว้แล้ว โปรแกรมก็จะแจ้งข้อผิดพลาดออกเช่นกัน ถ้าคุณสมบัติ DTREnable หรือ RTSEnable ถูกกำหนดให้เป็น True ก่อนที่จะทำการเปิดพอร์ตค่าคุณสมบัติของ DTREnable หรือ RTSEnable จะถูกเซ็ตเป็น false หลังจากปิดพอร์ต แต่ถ้าเซ็ตเป็น false หลังจากปิดโปรแกรมแล้ว ค่าที่กำหนดไว้จะเป็นเท่าเดิม การเปิด Serial Port การเปิดพอร์ตสื่อสารด้วย MSComm Control สามารถทำได้ง่ายๆ ด้วยการกำหนด Properties ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MSComm 1. CommPort = 4

MSComm 1.Settings = "19, 200, N, 8, 1"

MSComm 1. PortOpen = True

5. Input เป็นการอ่านข้อมูลจากบัฟเฟอร์ พร้อมทั้งลบข้อมูลในบัฟเฟอร์ด้านรับทิ้ง ซึ่งสามารถอ่านค่าได้เฉพาะในขณะทำงานเท่านั้น รูปแบบการใช้งานคือ

Object. Input

ทุกครั้งที่มีการใช้คุณสมบัติ Input ในการอ่านข้อมูลจากบัฟเฟอร์ด้านรับเข้านั้น จำนวนของตัวอักษรที่อ่านได้จะถูกกำหนดลงในคุณสมบัติ InputLen ทันทีซึ่งถ้าหากกำหนดให้คุณสมบัติ InputLen มีค่าเท่ากับ 0 ก็จะหมายถึงการกำหนดให้คุณสมบัติของ input อ่านข้อมูลทั้งหมดจากบัฟเฟอร์ด้านรับเข้านั่นเองซึ่งชนิดของข้อมูลที่อ่านโดยคุณสมบัติ input จะเป็นข้อมูลแบบข้อความหรือไบนารีก็ขึ้นกับการกำหนดค่าของคุณสมบัติ InputMode

Object.Output [=Value]

สำหรับชนิดของข้อมูลที่ถูกส่งโดยคุณสมบัติ Output จะเป็นข้อมูลแบบข้อความหรือไบนารี ก็ขึ้นกับการกำหนดค่าของคุณสมบัติ InputMode ดังที่กล่าวมาข้างต้น

2.3 การติดต่อสื่อสารและการส่งผ่านข้อมูล

2.3.1 กล่าวนำ

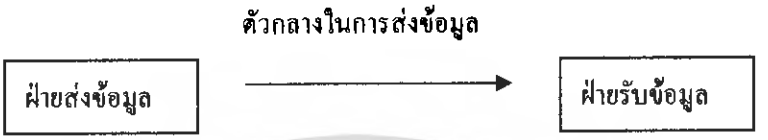
การสื่อสารแบบ Host link เป็นการเชื่อมต่อระหว่างเครื่อง PLC กับคอมพิวเตอร์ทั่วไป โดยผ่านทางพอร์ตของคอมพิวเตอร์(COM1:หรือCOM2:) ส่วนมากจะนิยมใช้มาตรฐานการสื่อสารแบบอนุกรม เพื่อให้สามารถควบคุม PLC จากคอมพิวเตอร์ได้ สำหรับคอมพิวเตอร์ 1 เครื่องสามารถต่อเข้ากับ PLC ได้จำนวนมาก โดยใช้การเชื่อมต่อหลายๆตัวเข้าด้วยกันเรียกว่า PC LINK ในการติดต่อแบบ Host link จะต้องผ่านอุปกรณ์ที่เรียกว่า Host link Unit ซึ่งจะต้องตั้งค่าต่างๆที่จำเป็นใช้ในการติดต่อแบบ Host link Unit SLK 23 ผ่านสวิตซ์ของเครื่อง PLC

82004
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 การสื่อสารข้อมูลทั่วไป

ส่วนประกอบเบื้องต้นในการสื่อสารข้อมูลแบ่งได้ออกเป็น 3 ส่วนคือ

1. ฝ่ายส่งข้อมูล (Transmitter)
2. ตัวกลางในการส่งผ่านข้อมูล (Medium)
3. ฝ่ายรับข้อมูล (Receiver)



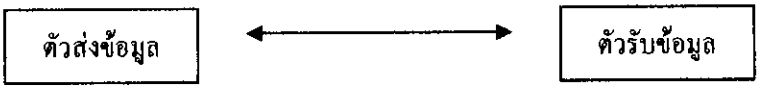
รูปที่ 2.10 แสดงส่วนประกอบหลักในการสื่อสารข้อมูล

2.3.3 การส่งข้อมูลแบบซิมเพล็กซ์ และแบบดูเพล็กซ์

ในการสื่อสารไม่ว่าจะเป็นการสื่อสารข้อมูลหรือการสื่อสารทั่วไปนั้นย่อมจะต้องประกอบด้วย ผู้รับและผู้ส่ง ผู้รับในขณะนี้อาจจะเป็นผู้ส่งในอนาคตก็ได้ แต่มีบางกรณีที่เป็นผู้รับและ ผู้ส่งแน่นอนตายตัวอยู่ตลอดเวลา เช่น การสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับเครื่องพิมพ์ เป็นต้น การสื่อสารข้อมูลของอุปกรณ์ที่มี ผู้รับ และ ผู้ส่ง ตายตัวนั้น เราเรียกว่าการสื่อสารแบบซิมเพล็กซ์ กล่าวคือการสื่อสารเป็นลักษณะทิศทางเดียวตลอดเวลาซึ่งจะมีที่ใช้ไม่มากนัก การสื่อสารโดยทั่วไปนั้นจะเป็นแบบดูเพล็กซ์ คือ มีทิศทางการสื่อสาร 2 ทิศทางทั้งไปและกลับการสื่อสารในลักษณะดูเพล็กซ์นี้ยังแบ่งออกเป็น 2 ชนิดคือ แบบฮาล์ฟดูเพล็กซ์ (Half Duplex) นิยมเขียนย่อกันว่า HDX ซึ่งจะมีทิศทางการสื่อสารในลักษณะที่ผลัดกันเป็นผู้ส่งและผู้รับพร้อมกันไปและแบบฟูลดูเพล็กซ์ (Full Duplex) นิยมเขียนย่อว่า FDX จะมีทิศทางการสื่อสารในลักษณะสัญญาณรับทิศทางหนึ่งสัญญาณส่งอีกทิศทางหนึ่งสัญญาณส่งอีกทางหนึ่งหรือกล่าวได้อีกนัยว่า สัญญาณรับและส่งจะมีสายตัวนำแยกออกจากกันโดยเด็ดขาด

รูปที่ 2.11 แสดงการส่งข้อมูลแบบทิศทางเดียว (Simplex)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 แสดงการส่งข้อมูลแบบฮาล์ฟดูเพล็กซ์ (Half Duplex)

รูปที่ 2.13 แสดงการส่งข้อมูลแบบฟลูดูเพล็กซ์(Full Duplex)

2.3.4 การส่งข้อมูลแบบอนุกรมในเครื่องคอมพิวเตอร์นั้นจะใช้มาตรฐาน RS - 232C

มาตรฐาน RS - 232C เป็นมาตรฐานที่ได้รับการออกแบบมาเพื่อที่จะทำให้อุปกรณ์จากผู้ผลิตต่างกันสามารถทำงานร่วมกันได้ มาตรฐานหลายชนิดได้รับการออกแบบขึ้นมา แต่มาตรฐานที่ได้รับความนิยมและใช้กันกว้างขวางมากที่สุดคือ มาตรฐาน RS - 232C ซึ่งถูกประกาศใช้ในปี 1969 โดยสมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association: EIA) มาตรฐาน RS - 232C ได้แบ่งอุปกรณ์ออกเป็น 2 ประเภท ทั้งนี้ก็เพื่อป้องกันไม่ให้เกิดการส่งข้อมูลบนสายเส้นเดียวกัน ซึ่งอุปกรณ์ทั้งสองประเภทนี้ก็คือ

- อุปกรณ์ DTE (Data Terminal Equipment) เป็นอุปกรณ์สำหรับส่งข้อมูล (เอาต์พุต)
- อุปกรณ์ DCE (Data Communication Equipment) เป็นอุปกรณ์สำหรับรับข้อมูล (อินพุต)

ตามมาตรฐาน RS - 232C คอนเน็คเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็คเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งคอนเน็คเตอร์ที่นิยมใช้กันอยู่จะเป็นชนิด D-Type แบบ 9 ขา และ แบบ 25 ขา โดยจะติดตั้งอยู่หลังเครื่องคอมพิวเตอร์ ระดับแรงดันจะมีค่าระหว่าง -3 V ถึง -15 V สำหรับลอจิก High และลอจิก Low จะมีระดับแรงดันระหว่าง +3 V ถึง +15 V สามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุด 50 ฟุต หรือ 150 เมตรแต่ถ้าเราต้องการสื่อสารกับอุปกรณ์อื่นที่อยู่ห่างกันมากๆ เราจำเป็นต้องใช้อุปกรณ์อื่นๆ เข้าช่วย เช่นการใช้โมเด็มเป็นต้น

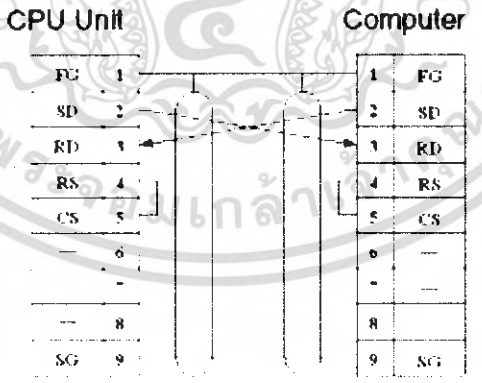
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของคอนเน็คเตอร์แบบ D-Type

หัวต่อแบบ D-Type ที่ใช้ในการสื่อสารแบบอนุกรมของเครื่องคอมพิวเตอร์นั้น จะมีอยู่ 2 ลักษณะคือ แบบ 9 ขา และแบบ 25 ขา บางครั้งเราจะเรียกว่า DB9 และ DB25 ซึ่งหัวต่อทั้งสองชนิดจะมีลักษณะการทำงานของสัญญาณเหมือนกัน แต่การจัดเรียงไม่เหมือนกัน และในโครงงานได้มีการเชื่อมต่อดังรูป 2.14

ตารางที่ 2.1 แสดงขาคอนเน็คเตอร์แบบ D-Type

D-Type 25 Pin	D-Type 9 Pin	สัญลักษณ์	ชื่อสัญญาณ
Pin 2	Pin 3	TD	Transmit Data
Pin3	Pin 2	RD	Receive Data
Pin 4	Pin 7	RTS	Request To Send
Pin 5	Pin 8	CTS	Clear To Send
Pin 6	Pin 6	DSG	Data Set Ready
Pin 7	Pin 5	SG	Signal Ground
Pin 8	Pin 1	CD	Carrier Defect
Pin 20	Pin 4	DTR	Data Terminal Ready
Pin 22	Pin 9	RI	Ring Indicator



รูปที่ 2.14 แสดงการเชื่อมต่อPLCกับคอมพิวเตอร์ผ่านพอร์ตสื่อสารอนุกรมขนาด 9 ขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

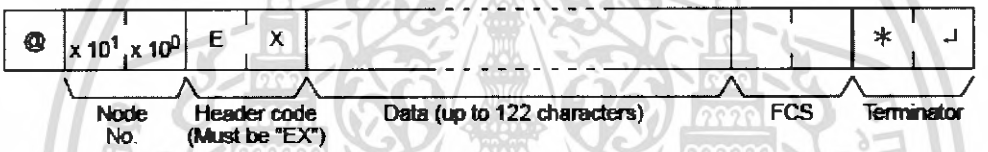
2.3.5 การข้อมูลระหว่างคอมพิวเตอร์กับ PLC ผ่านทาง RS-232C ด้วยโปรแกรมวิชวลเบสิก 6 (Visual Basic 6)

2.3.5.1 รูปแบบชุดคำสั่ง (Command Format)

ในการสื่อสารข้อมูลจะต้องมีกฎหรือข้อกำหนดในการสื่อสารข้อมูล เรียกว่า โปรโตคอล (Protocol) ซึ่งจะเป็นส่วนที่กำหนดมาตรฐานในการควบคุมและจัดการระบบการสื่อสารข้อมูล ลักษณะของบล็อกของเครื่องควบคุม PLC/PC แต่ละผู้ผลิตจะแตกต่างกันออกไปแต่จะมีพื้นฐานเดียวกัน

@	หมายเลขเครื่อง	HEADER	TEXT	FCS	*

รูปที่ 2.15 แสดงรูปแบบของชุดข้อมูล (Block)



รูปที่ 2.16 แสดงรูปแบบของชุดคำสั่ง

จากรูปแบบของชุดคำสั่งที่ส่งจากเครื่องคอมพิวเตอร์ไปยัง PLC สามารถอธิบายส่วนต่างๆ ได้ดังนี้

1. หมายเลขเครื่อง (Node No.) ในการเชื่อมต่อที่เป็นเครือข่ายแบบหลายๆจุดนั้นเครื่องควบคุมที่เชื่อมต่ออยู่ในระบบมากกว่า 1 เครื่อง ดังนั้นจะต้องกำหนดหมายเลขเครื่องให้เครื่องควบคุม เช่น @03 คือเครื่องควบคุมหมายเลขที่ 03

2. ส่วน HEADER CODE เป็นส่วนของคำสั่งหลักที่จะกำหนดว่าต้องกระทำกับข้อมูลส่วนใด ซึ่งจะเป็นตัวอักษรย่อภาษาอังกฤษ เช่น ต้องการอ่านข้อมูลอินพุตจากเครื่องควบคุมจะต้องใช้ตัวอักษรพิมพ์ใหญ่ว่า “RR” หรือถ้าต้องการเขียนข้อมูลให้กับเครื่องควบคุมใช้ ”WR” เป็นต้น ซึ่งสามารถดูรายละเอียดในภาคผนวก

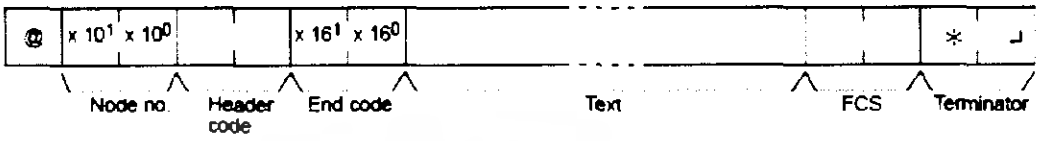
3. ส่วน TEXT เป็นส่วนของข้อมูล เช่น ค่าที่อ่านได้จากอินพุต หรือค่าที่ต้องการเขียนลงในพื้นที่ต่างๆ

4. ส่วนของ FCS เป็นส่วนของการควบคุมความผิดพลาดของข้อมูล ซึ่งได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ส่วนของ TERMINAL เป็นส่วนที่ปิดท้ายบอกให้ทราบว่าจบบล็อกและมักจะติดตามด้วยรหัส Carrier Return (CR)

2.3.5.2 รูปแบชุดผลตอบสนอง (Response Block)



รูปที่ 2.17 แสดงรูปแบบของชุดผลตอบสนอง

รูปแบชุดผลตอบสนอง (Response Block) ซึ่งส่วนประกอบจะเหมือนกับ Command Block แต่จะแตกต่างกันตรงที่บล็อกผลตอบสนองนั้น จะมีส่วนของ End code ซึ่งใช้แสดงสถานะในการสื่อสารและบอกความผิดพลาดที่เกิดขึ้นในการสื่อสาร ในกรณีที่ไม่มีความผิดพลาดเกิดขึ้น End code จะเท่ากับ 00 แต่ถ้า End code ไม่เท่ากับ 00 แสดงว่าการสื่อสารมีความผิดพลาด ซึ่งสามารถตรวจสอบสาเหตุของความผิดพลาดได้จากตารางแสดงค่า End code กับสาเหตุของความผิดพลาดในภาคผนวก เช่น End code แสดงตัวเลข 18 แสดงว่าเกิดความผิดพลาดในเรื่องขนาดความยาวของข้อมูลเกิน เป็นต้น

ตัวอย่างของรูปแบบชุดคำสั่งและชุดตอบสนอง

@	10	RR	00040003	46	*CR

รูปที่ 2.18 แสดงตัวอย่างรูปแบบของชุดคำสั่ง

@	10	RR	00	AB59 6324 7831	47	*CR

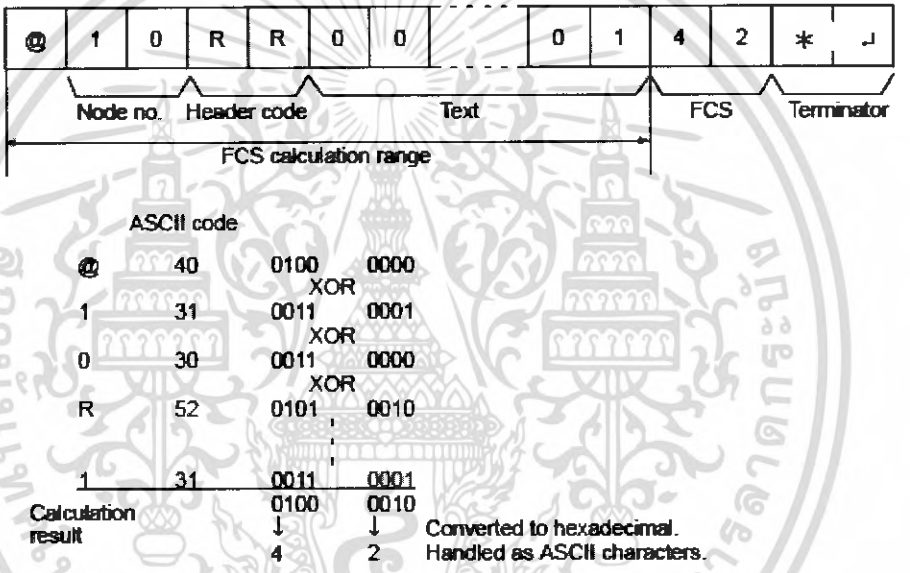
รูปที่ 2.19 แสดงตัวอย่างรูปแบบของชุดผลตอบสนอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.5.3 การคำนวณ FCS

FCS หรือ Frame Check Sequence เป็นสิ่งที่ใช้ในการตรวจสอบความผิดพลาดของการสื่อสาร ซึ่งจะทำหน้าที่คล้ายกับพาริตี้บิตในการสื่อสารแบบอนุกรม ในการรับส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์กับ PLC ทุกครั้ง จะต้องมีการคำนวณ FCS เพื่อใส่ให้กับชุดข้อมูลที่จะส่งออกไป หรือนำ FCS ที่คำนวณได้จากชุดข้อมูลรับเข้ามามาเปรียบเทียบกับค่า FCS ในชุดข้อมูลรับที่เข้ามาว่าตรงกันหรือไม่

หลักการนำข้อมูลในชุดข้อมูล โดยเริ่มจาก @ จนถึงตัวอักษรตัวสุดท้ายของ TEXT ข้อมูล 8 บิต เป็น 2 ตัวอักษร เป็นรหัสข้อมูล ASCII แล้วจึงนำมาทำการ EXCLUSIVE-OR (XOR) กันจะได้ค่า FCS

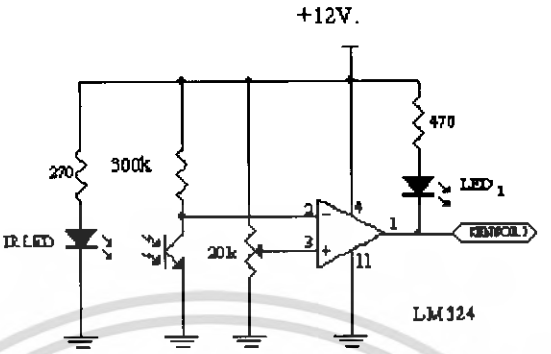


รูปที่ 2.20 แสดงการคำนวณหา FCS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 หลักการทำงานของวงจร

2.4.1 วงจรเซนเซอร์



รูปที่ 2.21 วงจรเซนเซอร์

จากรูปที่ 2.21 วงจรเซนเซอร์อินฟาเรดมีองค์ประกอบ 4 ส่วน คือ

- ส่วนภาคส่งสัญญาณอินฟาเรด ซึ่งในส่วนนี้จะทำการส่งสัญญาณอินฟาเรดไปให้กับภาครับสัญญาณอินฟาเรด

- ส่วนภาครับสัญญาณอินฟาเรด ซึ่งในส่วนนี้เมื่อได้รับสัญญาณอินฟาเรดแล้วจะทำการส่งผลให้เกิดค่าความต้านทานของภาครับ และจะส่งสัญญาณในส่วนนี้ไปเปรียบเทียบค่าแรงดัน

- ส่วนเปรียบเทียบค่าแรงดัน (LM324) ส่วนนี้จะเปรียบเทียบค่าแรงดันตกคร่อมของภาครับสัญญาณอินฟาเรดกับค่าแรงดันคงที่ที่ตั้งค่าไว้และเมื่อทำการเปรียบเทียบแล้วจะได้ผลเป็นไปตามตารางที่ 2.2

ตารางที่ 2.2 แสดงผลการเปรียบเทียบค่าแรงดัน

แรงดันขาบวก	แรงดันขาลบ	แรงดันที่ส่งออก
มาก	น้อย	12 V
น้อย	มาก	0 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนแสดงค่าการเปรียบเทียบ ในส่วนนี้มีไว้แสดงผลของส่วนเปรียบเทียบแรงดันไฟฟ้าเพื่อง่ายต่อการปรับตั้งค่าความต้านทานอ้างอิงโดยแสดงผลในตารางที่ 2.3

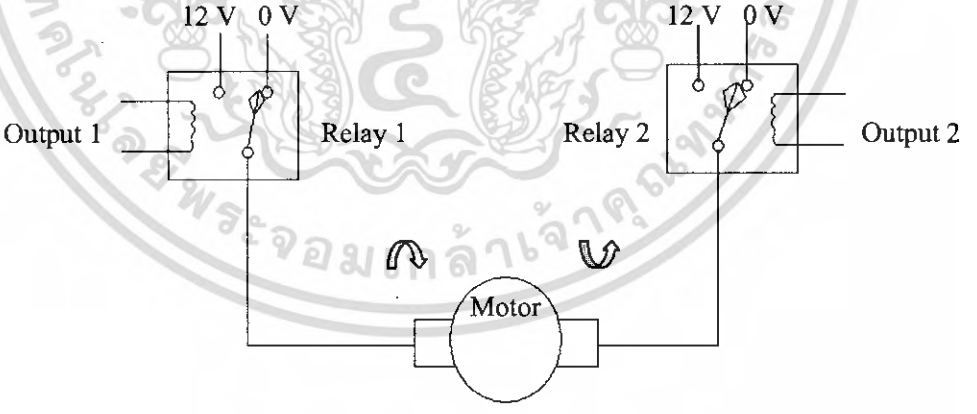
ตารางที่ 2.3 แสดงสถานะหลอด LED

แรงดันเอาท์พุตจากส่วนเปรียบเทียบแรงดัน	สถานะหลอด LED
12 V	ติด
0 V	ดับ

การที่สัญญาณเอาท์พุตออกมามีค่า 12 โวลต์ แล้วทำให้ LED ติดนั้นเป็นเพราะแรงดันที่ขาบวกของ LED ต่อเข้ากับแรงดันบวก 12 โวลต์ ดังนั้นเมื่อแรงดันที่ขาลบของ LED มีค่าบวก 12 โวลต์ จึงทำให้ค่าความต่างศักย์ตกคร่อมหลอด LED นั้นมีค่าเป็นศูนย์หลอด LED จึงติดในทางตรงกันข้ามถ้าแรงดันที่ขาลบของ LED มีค่าแรงดันเป็นศูนย์ จึงทำให้ค่าแรงดันต่างศักย์ตกคร่อม LED มีค่าบวก 12 โวลต์ LED จึงดับ

2.4.2 วงจรรีเลย์และวงจรขับมอเตอร์

การควบคุมทิศทางการหมุนของมอเตอร์จะใช้รีเลย์ควบคุมซึ่งวงจรเป็นไปตามรูป



รูปที่ 2.22 แสดงวงจรควบคุมทิศทางการหมุนของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 แสดงทิศทางการหมุนของมอเตอร์

รีเลย์ตัวที่ 1	รีเลย์ตัวที่ 2	ทิศทางการหมุนของมอเตอร์
ทำงาน	ไม่ทำงาน	หมุนไปทางขวา
ไม่ทำงาน	ทำงาน	หมุนไปทางซ้าย

สัญญาณเอาท์พุตของ PLC ที่ออกมามีค่า 12 โวลต์ จะทำให้เป็นสัญญาณอินพุตของคอยล์รีเลย์ ซึ่งเมื่อมีสัญญาณเข้ามาหน้าสัมผัสก็จะต่อให้เป็นปกติปิด (Normally Close) ซึ่งได้ทำการต่อแรงดันไฟฟ้า 12 โวลต์เอาไว้ ซึ่งเป็นสัญญาณที่จะนำไปใช้ในการควบคุมทิศทางการหมุนของมอเตอร์ สัญญาณที่จะควบคุมคอยล์รีเลย์ไม่สามารถทำงานพร้อมกันได้ จะสลับกันทำงานโดยจะแสดงผลการทำงานดังตารางที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบ

การวางแผนการทำงานจะกำหนดส่วนที่ต้องศึกษาและส่วนที่ปฏิบัติงานออกเป็น 3 ส่วนคือ

- ออกแบบระบบจัดเก็บสินค้าและทำการเขียนโปรแกรมแลดเดอร์ไดอะแกรม โดยใช้ Software ชื่อว่า CX-Programmer เพื่อไปควบคุมส่วน Hardware (Warehouse)

- ออกแบบและสร้างส่วนของ Hardware โดยสร้างแบบจำลองกระบวนการการจัดเก็บสินค้าเพื่อนำไปประกอบการศึกษาในส่วนอื่นๆ

- ออกแบบแอพพลิเคชั่นต่างๆ โดยใช้โปรแกรมวิชวลเบสิก 6 และทำการส่งข้อมูลระหว่างโปรแกรมวิชวลเบสิก 6 และ PLC หรือเรียกอีกอย่างว่า การเขียนโปรโตคอล เพื่อให้สามารถส่งงานทางหน้าจอ PC ได้

3.1 การออกแบบโปรแกรมแลดเดอร์ควบคุมการทำงานของ Warehouse

จากรูปร่างลักษณะโครงสร้างต่างๆของ Warehouse จะถูกนำมาใช้เป็นข้อมูลในการออกแบบโปรแกรมแลดเดอร์การควบคุมทั้งหมดโปรแกรมแลดเดอร์จะควบคุมให้ชุดเคลื่อนที่ของโครงสร้าง Warehouse เคลื่อนที่ไปยังจุดที่ผู้ใช้งานต้องการ เพื่อทำการเก็บวัสดุเข้าไปในช่องจัดเก็บเพื่อนำออกไปใช้งานโปรแกรมแลดเดอร์ที่ใช้ควบคุม Warehouse จะแบ่งออกได้ 5 ส่วนใหญ่ๆ คือ

3.1.1 ส่วนเก็บข้อมูลที่ผู้ใช้เลือกช่องที่ต้องการ

จากหลักการทำงานของ Warehouse ที่มีจุดประสงค์ในการจัดการวัสดุเข้าไปเก็บหรือนำออกมาจากช่องจัดเก็บจะมีการทำงานอยู่ 2 ลักษณะ คือ

1. การนำวัสดุเข้าไปเก็บในช่อง (IN) การทำงานในลักษณะของการเก็บวัสดุต้องการข้อมูลว่าจะต้องนำวัสดุไปเก็บไว้ในช่องที่ต้องการ และยังมีตำแหน่งที่รับวัสดุจากภายนอกที่จะนำเข้ามาเก็บร่วมพิจารณาด้วย ดังนั้นข้อมูลที่จำเป็นจะต้องใช้ในการทำงานจะต้องทำการเลือกตำแหน่งที่ต้องการ 1 ช่อง และใช้ช่องรับเข้ามาร่วมพิจารณาด้วย การเลือกช่องที่ต้องการสามารถเลื่อนได้จากแผงควบคุม

2. การนำวัสดุออกไปใช้ภายนอก (OUT) การทำงานในลักษณะนี้จะคล้ายกับการนำวัสดุเข้าไปเก็บ จะต้องทำการเลือกตำแหน่งที่ต้องการ 1 ช่อง และพิจารณาช่องที่จะนำของออกไปวาง เพื่อตัดสินใจในการทำงานด้วย การเลื่อนตำแหน่งที่ต้องการสามารถเลือกได้จากแผงควบคุมเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 ส่วนป้องกันการทับซ้อนกัน

ในการเลือกช่องเพื่อจะให้ทำงานตามต้องการตำแหน่งที่ทำการเลือกของแต่ละแบบจะ ต้องมีคุณสมบัติในขณะทำการเลือกนั้นตรงกับวัตถุประสงค์ในการทำงาน คุณสมบัติของตำแหน่งที่สำคัญในการทำงานคือ

- ส่วนป้องกันวัสดุทับซ้อน ช่องที่ผู้ใช้ทำการเลือกใช้งานจะมีสถานะของตัวมันเองอยู่ ถ้าทำการเลือกช่องที่ต้องที่ต้องการแล้ว สถานะไปขัดกับการทำงานจะทำให้เกิดความเสียหายได้ คือ ถ้าหากนำวัสดุออกจากช่องที่ยังไม่มีวัสดุจัดเก็บก็จะทำให้สิ้นเปลืองพลังงาน เนื่องจากความเสียหายที่อาจเกิดขึ้นได้ จึงจำเป็นที่จะต้องทำการเขียนโปรแกรมป้องกันการทำงานที่นำเข้าช่องที่มีของ และเขาของออกในช่องที่ไม่มีของโดยเขียนให้ PLC ไม่รับคำสั่งที่สั่งให้ทำงานเช่นนั้น ทำให้ไม่สามารถสั่งให้ทำคำสั่งนั้นได้จนกว่าจะสั่งได้ถูกต้อง ไม่ขัดแย้งกับเงื่อนไขดังกล่าว

- ส่วนป้องกันการเลือกช่องทับซ้อน ส่วนนี้จะมีความสำคัญมากเมื่อมีการกดคำสั่งแรกให้ทำตามคำสั่งตามกระบวนการทำงานไปแล้ว และเมื่อมีการกดคำสั่งต่อไปหรือมีการเลือกคำสั่งอื่นเข้าไปซ้อนกับคำสั่งที่กระทำตามกระบวนการอยู่ อาจจะทำให้สิ้นค้าเกิดความเสียหาย เสียเวลา และจะทำให้กระบวนการที่ทำตามคำสั่งเกิดการทำงานผิดพลาดไปจากเดิมดังนั้นจึงต้องออกแบบให้มีส่วนป้องกันเมื่อกดคำสั่งแรกให้กระบวนการทำงานตามที่ต้องการจนเสร็จสิ้นคำสั่งนั้นโดยจะไม่มีผลต่อกระบวนการทำงานถ้าหากกดคำสั่งบอกตำแหน่งอื่นๆทับซ้อนกระบวนการทำงานแรกที่ทำงานอยู่และคำสั่งที่อยู่ในส่วนคำสั่งทับซ้อนจะมีผลก็ต่อเมื่อกระบวนการนั้นเสร็จสิ้น และคำสั่งส่วนทับซ้อนจะทำงานตามลำดับก่อน-หลัง ที่มีการป้อนคำสั่ง

- ส่วนป้องกันมอเตอร์ มอเตอร์ตัวเดียวกันไม่สามารถสั่งให้ไปหน้า (Forward) และกลับทางหมุน (Reward) พร้อมได้ ดังนั้นเมื่อกดพร้อมกันมอเตอร์จะไม่ทำงาน และเมื่อมอเตอร์ทำงานอยู่ไม่สามารถสั่งให้มอเตอร์กลับทางหมุนได้ทันที่ทันใด จึงต้องมีการหน่วงเวลาเพื่อให้มอเตอร์หยุดนิ่งเสียก่อน ก่อนที่จะสั่งให้มอเตอร์กลับทางหมุนเพื่อป้องกันมอเตอร์เกิดการเสียหาย

3.1.3 ส่วนเรียกข้อมูลที่เลือกไว้มาใช้งาน

หลังจากกดคำสั่งให้ทำงานแล้ว โปรแกรมจะเรียกโปรแกรมย่อยที่ทำงานตามคำสั่งนั้นออกมาทำงาน ให้ทำการสั่ง Warehouse เคลื่อนที่ไปทำงานตามคำสั่งที่กดไว้

3.1.4 ส่วนลำดับการเคลื่อนที่

โปรแกรมย่อยซึ่งในโปรแกรมจะลำดับการทำงานของ Warehouse แต่ละส่วนเอาไว้ โดยเริ่มการทำงานเมื่อทำการกดคำสั่งเสร็จ และหยุดเมื่อจบสิ้นกระบวนการทำงาน หรือ กด Emergency Stop

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

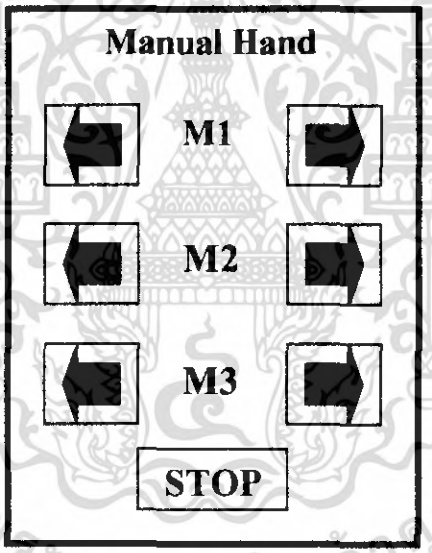
3.1.5 ส่วน Emergency Stop

เป็นโปรแกรมการทำงานส่วนพิเศษที่ไว้หยุดการทำงานของ Warehouse เนื่องจากเกิดความผิดพลาดของระบบ หลังจากแก้ข้อผิดพลาดเสร็จแล้ว ให้กด Reset และกด Emergency Stop อีกครั้ง Warehouse จะกลับสู่สภาวะเริ่มต้น

3.2 การออกแบบส่วนควบคุมการทำงาน

การสั่งให้ Warehouse ทำงานสามารถสั่งได้ 2 ลักษณะคือ

1. Manual operation คือ การสั่งผ่านการกดปุ่มควบคุมในส่วนของ Manual จะใช้ก็ต่อเมื่อระบบ Automatic เกิดความผิดพลาดจนไม่สามารถทำงานได้ เพื่อเป็นการสั่งให้มอเตอร์สามารถเคลื่อนที่ตามที่เราสั่งให้เคลื่อนที่ไปยังจุดๆนั้น การควบคุม Manual นี้ต้องทำการเปิด Selector Switch Auto /Manual มาอยู่ที่ตำแหน่ง Manual เสียก่อน จึงจะทำการควบคุมได้ ซึ่งการควบคุมแบบนี้จะใช้ผู้ควบคุมเป็นผู้ควบคุมด้วยตัวเอง ดังแสดงในรูป



รูปที่ 3.1 Manual Hand

→ M1 ใช้สำหรับให้เคลื่อนที่ออกไปรับของในแนวแกน X

← M1 ใช้สำหรับให้เคลื่อนที่เข้าหลังจากรับของในแนวแกน X

→ M2 ใช้สำหรับให้เคลื่อนที่ไปทางขวาในแนวแกน Y

← M2 ใช้สำหรับให้เคลื่อนที่ไปทางซ้ายในแนวแกน Y

→ M3 ใช้สำหรับให้เคลื่อนที่ขึ้นในแนวแกน Z

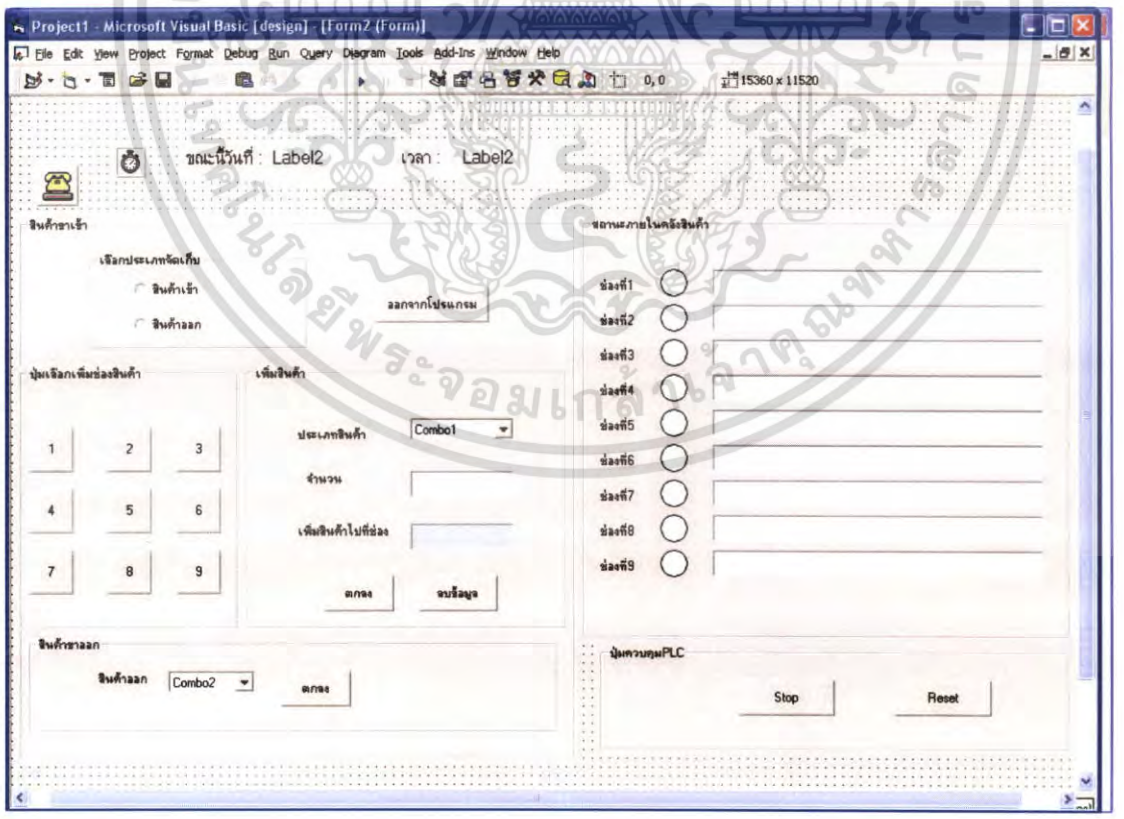
← M3 ใช้สำหรับให้เคลื่อนที่ลงในแนวแกน Z

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Automatic Operation การสั่งงานให้ระบบทำงานอัตโนมัติตามคำสั่งที่ป้อนให้กับระบบ ซึ่งเป็นการสั่งงานลักษณะปกติของ Warehouse โดยจะทำงานเมื่อเลือก Selector Switch Manual ไปอยู่ที่ตำแหน่ง Auto คำสั่งจากปุ่มกดในส่วน Automatic จะถูกส่งไปยัง PLC เพื่อที่ PLC ได้รับคำสั่งและทำงานต่อไป โดยเคลื่อนที่ให้ทำงานตามที่โปรแกรมเอาไว้ โดยหลักการสั่งจากคอมพิวเตอร์จะกล่าวในหัวข้อการทดลอง

3.3 ออกแบบแอพพลิเคชั่นต่างๆ โดยโปรแกรมวิชวลเบสิก 6 และผลการทดลอง

หลังจากสร้างแบบจำลองเสร็จแล้วและกำหนดแอดเดรสที่จะต้องส่งข้อมูลไปให้ PLC แล้วเราก็ทำการออกแบบแอพพลิเคชั่น ดังรูปที่ 3.1 โดยใช้ Microsoft Comm Control 6.0 เป็นตัวติดต่อกับ PLCโดยมีการกำหนดค่าต่างๆดังภาคผนวก ก. แล้วสร้างไฟล์ .EXE (make) เมื่อเราสร้างแอพพลิเคชั่นเสร็จแล้ว หากเราต้องการนำแอพพลิเคชั่นที่สร้างขึ้นให้เรียกใช้งานได้เอง โดยไม่ต้องเปิดผ่านโปรแกรม Visual Basic หรือต้องการนำไปใช้งานในเครื่องคอมพิวเตอร์เครื่องอื่นๆเราสามารถทำได้โดยสร้างไฟล์เอ็กซ์คิวต์ (ไฟล์ที่มีนามสกุลเป็น .EXE) แล้วจึงเปิดใช้งานโปรแกรมจัดเก็บสินค้าได้



รูปที่ 3.2 แสดงการออกแบบแอพพลิเคชั่นเพื่อจัดเก็บสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการทำงาน ผู้ใช้งานเลือกประเภทที่ต้องการจัดเก็บโดยมีอยู่ 2 ประเภท คือ

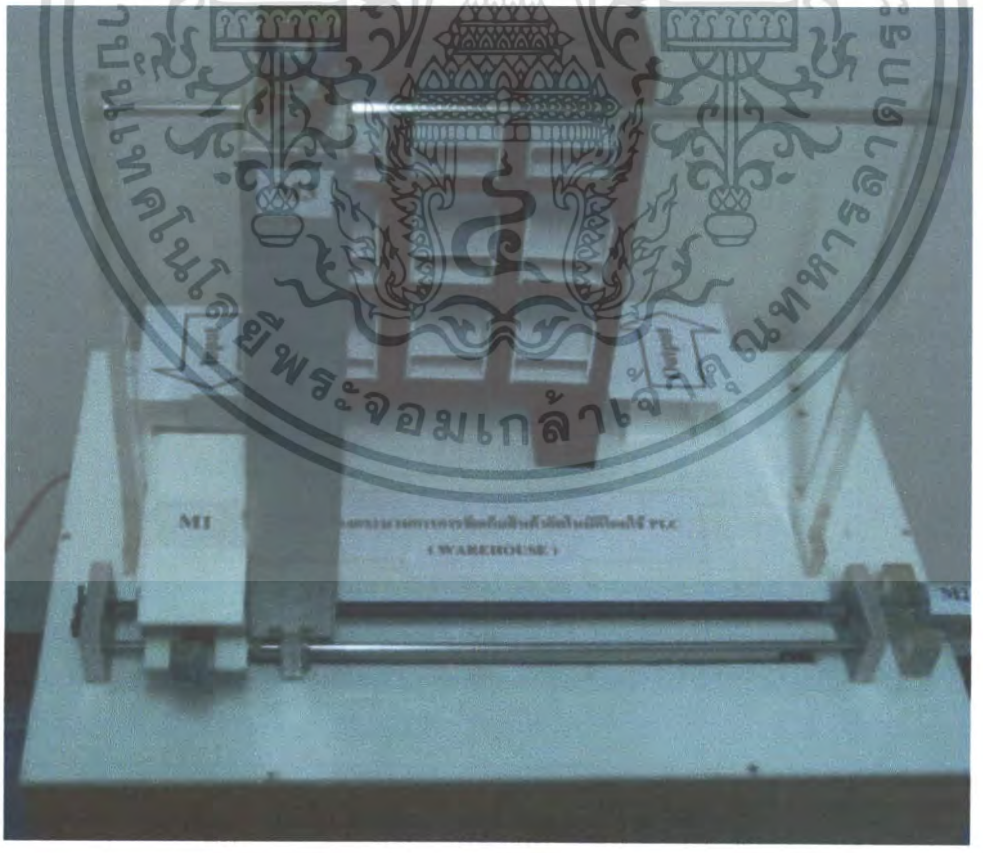
1.สินค้าเข้า

2.สินค้าออก

จากนั้นจะเห็นตัวเลือกให้กรอกข้อมูลหากเป็นสินค้าเข้าก็เลือกประเภทของสินค้าซึ่งในโปรแกรมมีสินค้าอยู่ 6 ชนิดและใส่ค่าของจำนวนสินค้าและใส่ช่องที่จะต้องทำการจัดเก็บช่อง 1-9 แล้วจึงกดตกลง ผู้ใช้งานควรตรวจสอบด้วยว่าที่ช่องสินค้าที่จะจัดเก็บนั้นมีสินค้าอยู่หรือเปล่า หากมีจะปรากฏสีแดงที่สถานะภายในคลังสินค้า และหากผู้ใช้ต้องการนำสินค้าออกก็เลือกประเภทการจัดเก็บเป็นสินค้าออกและจึงทำการเลือกช่องที่ต้องการนำสินค้าออก แล้วจึงกดตกลง

สถานะภายในคลังสินค้าจะบอกถึง ประเภทสินค้าว่าเป็นชนิดใด จำนวนเท่าไรและวัน เวลาที่จัดเก็บโดยจะมีสถานะที่หน้าช่องโชว์ หากมีสินค้าจะเป็นสีแดงและหากไม่มีเป็นสีขาว ปุ่มควบคุม PLC หากเกิดการผิดพลาดจากการจัดเก็บสินค้าให้ผู้ใช้กดปุ่ม Stop เพื่อหยุดกระบวนการแล้วจึงกดปุ่ม Reset เพื่อให้กระบวนการรอรับค่าสั่งงานใหม่

3.4 ออกแบบและสร้างส่วนของ Hardware



รูปที่ 3.3 แบบจำลองกระบวนการจัดเก็บสินค้าอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 เปิดโปรแกรมจัดเก็บสินค้า

โดยเชื่อมต่อเครื่องคอมพิวเตอร์กับ PLC เข้าด้วยกันด้วยพอร์ตอนุกรม RS-232C จากนั้นผู้ใช้เลือกประเภทการจัดเก็บสินค้าที่ต้องการ

1.สินค้าเข้า ข้อมูลที่ส่งไป PLC = @00WR0017040047*CR

2.สินค้าออก ข้อมูลที่ส่งไป PLC = @00WR0017100042*CR

โปรแกรมจะส่งข้อมูลออกไปให้ PLC เพื่อให้ทำตามภาษาแลดเดอร์ไดอะแกรมดังรูปที่ 3.2

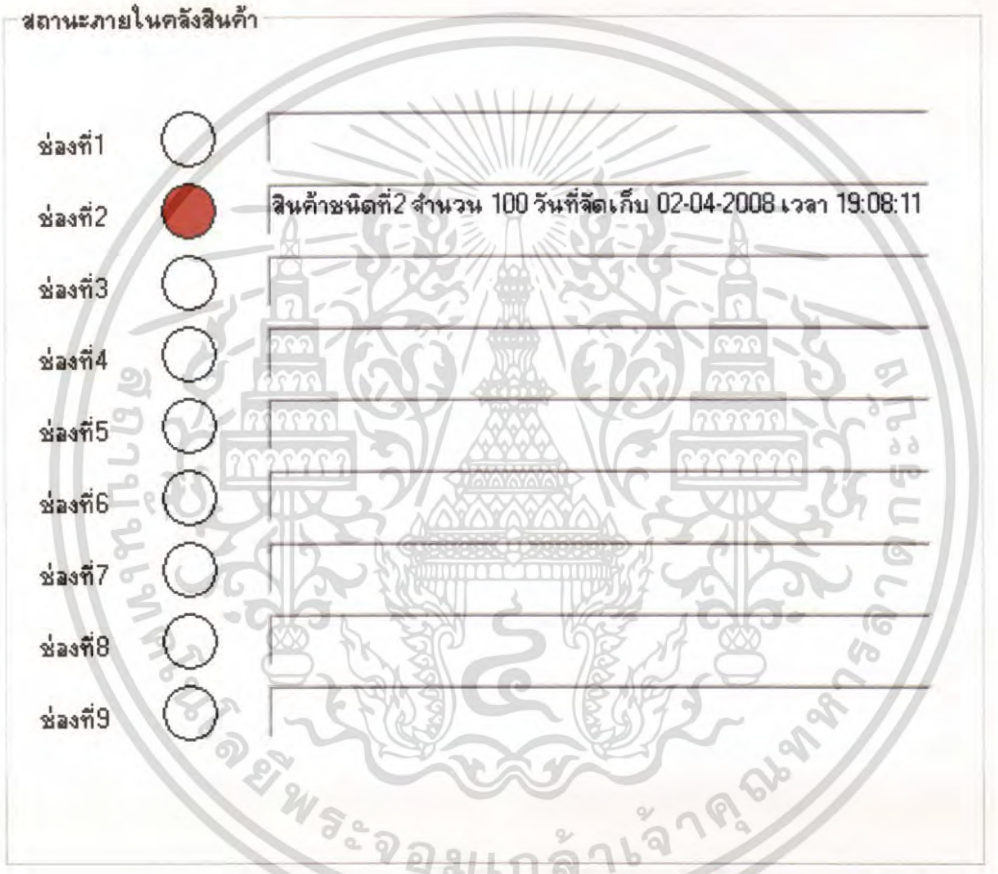
รูปที่ 4.1 เลือกประเภทการจัดเก็บสินค้าเข้าหรือสินค้าออก

4.2 การจัดเก็บสินค้าเข้า

รูปที่ 4.2 แสดงผลการทดลองการนำสินค้าเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.2 แสดงผลการทดลองให้เห็นว่าหากเลือกประเภทสินค้าเข้า เช่น จะนำสินค้าในช่องที่ 2 เข้าให้เลือกประเภทเป็นสินค้าออกซึ่งมีประเภทของสินค้าอยู่ 6 ชนิด แล้วกดปุ่มเลือกช่อง 2 แล้วตกลงจากนั้นโปรแกรมจะทำการส่งข้อมูล “@00WR0018000448*CR” ซึ่งคำสั่งจะถูกส่งออกไปทางพอร์ตอนุกรม RS-232C เพื่อควบคุมการทำงานของ PLC และรูปที่ 4.3 แสดงสภานะภายในคลังสินค้าที่ช่องที่ 2 ซึ่งเกิดจากผู้ใช้สั่งการ โดยจะเปลี่ยนเป็นสีแดงซึ่งแสดงว่ามีสินค้าภายในช่องเก็บสินค้าแล้ว อีกทั้งมีจำนวนของสินค้าและวันเวลาที่จัดเก็บ

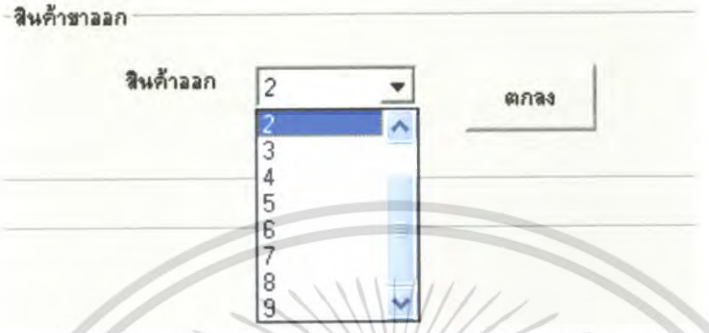


รูปที่ 4.3 แสดงสถานะภายในคลังสินค้าเมื่อจัดเก็บสินค้าที่ช่อง 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

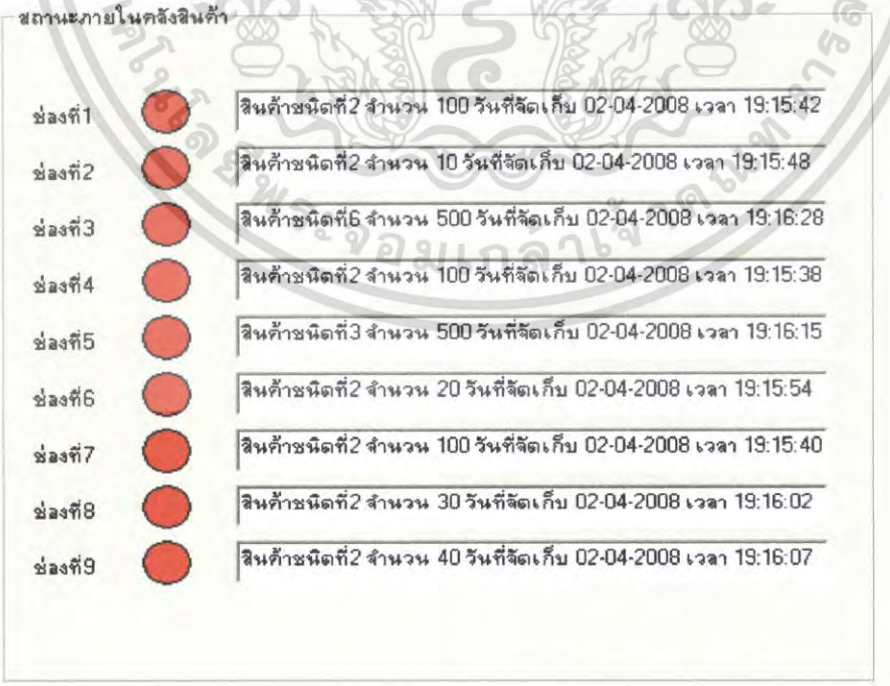
4.3 การขนถ่ายสินค้าขาออก

ผู้ใช้เลือกประเภทการจัดเก็บสินค้าขาออกแล้วจึงทำการเลือกช่องที่ต้องการที่จะขนถ่ายสินค้าออกแล้วจึงกดตกลงดังรูปที่ 4.4



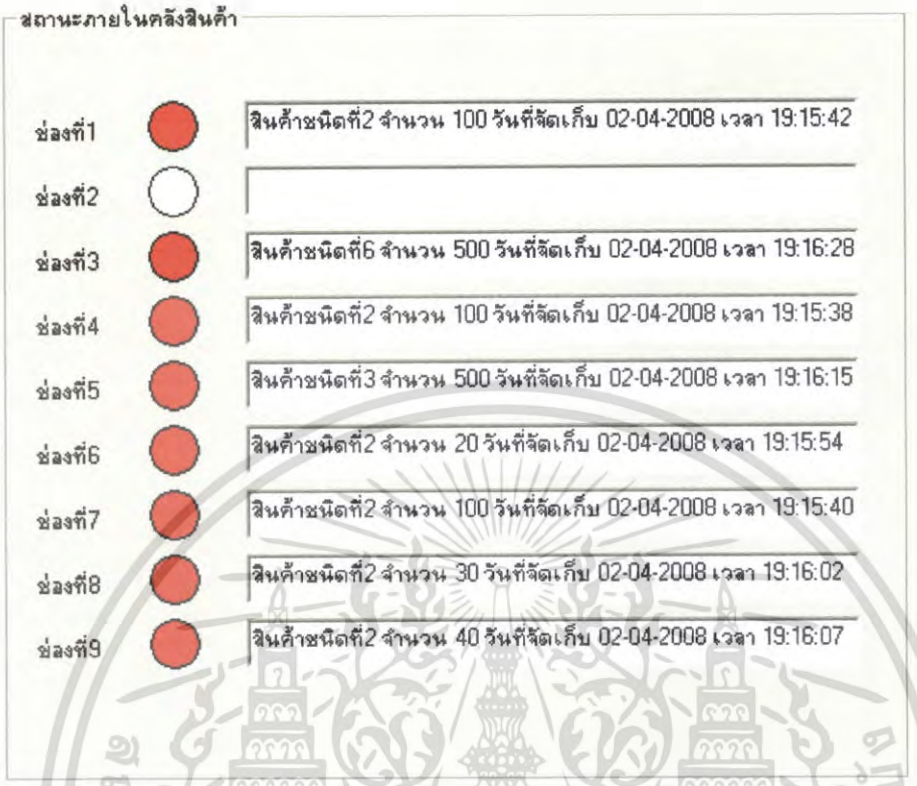
รูปที่ 4.4 การเลือกช่องสินค้าที่ต้องการขนถ่ายสินค้าออก

จากรูปที่ 4.5 แสดงสถานะภายในคลังสินค้าทั้ง 9 ช่องมีสินค้าเต็มแล้ว ผู้ใช้ต้องการจะทำการขนถ่ายสินค้าออกโดยนำสินค้าที่ต้องการในช่องที่ 2 ออกก็ทำตามขั้นตอนคือ เลือกประเภทสินค้าออก และช่องสินค้าที่ 2 แล้วกดปุ่มตกลง จากนั้นโปรแกรมจะทำการส่งข้อมูล "@00WR0018000448*CR" ไปที่PLC ผลที่ได้จะเป็นดังรูปที่ 4.6 จะเห็นได้ว่าสถานะภายในคลังสินค้าจะเปลี่ยนจากสีแดงเป็นสีขาวซึ่งแสดงว่าไม่มีสินค้าแล้ว



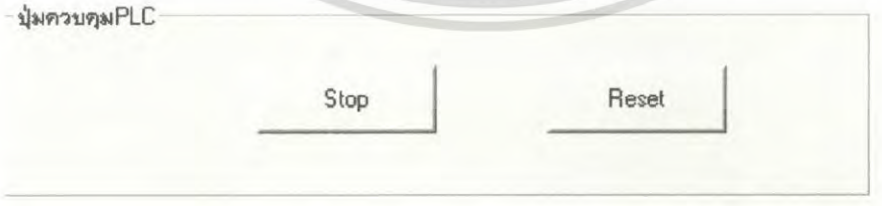
รูปที่ 4.5 แสดงสถานะภายในคลังสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 การขนถ่ายสินค้าช่องที่ 2 ออก

จากรูปที่ 4.7 ตัวควบคุมการทำงาน PLC ซึ่งมีไว้ควบคุมเมื่อเกิดข้อผิดพลาดจากการจัดเก็บ เช่น จัดเก็บช่องสินค้าผิดหรือสั่งขนถ่ายสินค้าออกผิด ให้ผู้ใช้กดปุ่ม Stop เพื่อหยุดการทำงานทั้งหมดของกระบวนการ แล้วให้ผู้ใช้กดปุ่ม Reset จะทำให้สินค้ามายังจุดรับสินค้า แล้วผู้ใช้จึงสั่งคำสั่งใหม่ต่อไป



รูปที่ 4.7 ตัวควบคุมการทำงาน PLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

5.1 สรุปผลการปฏิบัติงาน

จากบทที่ 1 ได้กล่าวถึงขอบเขตของโครงงาน ซึ่งแบ่งการปฏิบัติงานออกเป็น 3 ส่วนใหญ่ได้ดังนี้

1. สร้างแบบจำลองกระบวนการการจัดเก็บสินค้าและออกแบบภาควงจรไฟฟ้านำสองส่วนนี้มาประกอบเข้าด้วยกัน

2. เขียนแลดเดอร์ไดอะแกรม (Ladder Diagram) เพื่อให้สามารถควบคุมการทำงานของกระบวนการ

3. สร้างหน้าต่างแสดงผลข้อมูลและติดต่อ PLC ซึ่งเป็นไปตามผู้จัดทำโครงงานได้ตั้งใจไว้ และผลการทดลองทำให้รู้ถึงข้อผิดพลาดจากสิ่งที่เกิดขึ้นกับตัวโครงงานเช่น โปรแกรมแลดเดอร์ไดอะแกรมมีจุดผิดพลาดเล็กๆน้อยๆซึ่งอาจนำไปพัฒนาต่อได้ การเขียนโปรแกรมวิชวลเบสิก6 ต้องอาศัยประสบการณ์การเขียนโปรแกรมบ่อยๆเพื่อให้มีแอพพลิเคชั่นในการใช้งานที่หลากหลาย เป็นต้น

5.2 ปัญหาด้านการออกแบบโครงสร้างแบบจำลอง

1. ขณะที่ทำงานชุดเคลื่อนที่ซ้าย-ขวา (แกน X) จะทำให้ชุดเคลื่อนที่ยกของขึ้น-ลง (แกน Z) จะมีการส่ายเล็กน้อย แต่จะมีผลกระทบต่อการทำงานน้อยมาก

2. ขณะชุดเคลื่อนที่ยกของขึ้น-ลง (แกน Z) และชุดเคลื่อนที่รับ-ส่งของ ทำงานจะเกิดการสั่นอยู่บ้างเพียงเล็กน้อย

3. มีเสียงดังอยู่บ้างในขณะที่มอเตอร์ชุดยกของขึ้น-ลง ทำงาน

4. เมื่อใช้งานไปนานๆ จะมีการคลายตัวของน็อต ในบางชิ้นส่วนเป็นผลทำให้ชิ้นส่วนของแกนเคลื่อนที่หลวมซึ่งเป็นผลทำให้การทำงานของระบบผิดพลาดขึ้นหรือบางครั้งถ้าไม่รีบแก้ไขจะทำให้ชิ้นส่วนทางเครื่องกลเกิดความเสียหายขึ้นมาได้

5.3 ข้อเสนอแนะและแนวทางในการค้นคว้าและพัฒนา

ในการศึกษาโครงงานนี้เป็นการศึกษากระบวนการการจัดเก็บสินค้าอัตโนมัติโดยได้สร้างแบบจำลองกระบวนการขึ้น ซึ่งในการใช้งานในโรงงานจริงๆอาจมีกระบวนการที่ต่างไป ดังนั้นสิ่งที่จะได้ในโครงงานนี้คือความเข้าใจใน PLC และโปรแกรมวิชวลเบสิก 6 ซึ่งสามารถพัฒนาในงานที่ทำได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวทางการพัฒนาเพื่อให้เกิดประโยชน์อย่างมีประสิทธิภาพ คือ ศึกษาและทดลองเกี่ยวกับคำสั่ง Host Link Commands ที่มีในคู่มือของ PLC รุ่นนั้นๆนำมาใช้ติดต่อกับโปรแกรมวิชวลเบสิก 6 โดยคำสั่ง *Host Link Commands* ยังมีคำสั่งที่ผู้ศึกษาสามารถนำมาใช้ได้ และอีกสิ่งหนึ่ง คือ พัฒนาโปรแกรมภาษาแลดเดอร์ ซึ่งขึ้นอยู่กับความชำนาญของตัวผู้ศึกษาเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1) กิตติ ภักดีวัฒนะกุล, จำลอง ครูอุตสาหะ. Visual Basic 6 ฉบับโปรแกรมเมอร์: บริษัท ดวงกมลสมัย, 2544

2) ชติพันธุ์ หล่อเลิศสุนทร , ฐิติตะพันธุ์ หล่อเลิศสุนทร. สอนเขียน Visual Basic 6.0 ให้เป็น Project: บริษัท แอ็ดวานซ์ มีเดีย ซัพพลายส์, 2543

3) รศ.ธีรศิลป์ ทุมวิภาต , สุภาพร จำปาทอง. เรียนรู้ PLC ขั้นกลาง ด้วยตนเอง. กรุงเทพฯ: ซีเอ็ดยูเคชั่น, 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

โปรแกรมวิชวลเบสิกและภาษาแลดเดอร์ไดอะแกรม

โปรแกรมวิชวลเบสิกนี้สร้างแอพพลิเคชั่นบน Windows ได้เขียนลงในเครื่องคอมพิวเตอร์เพื่อสั่งการให้เครื่อง PLC ทำตามโปรแกรมภาษาแลดเดอร์ซึ่งภาษาแลดเดอร์จะทำหน้าที่ควบคุมกระบวนการทำงานของแบบจำลอง

```
Private A, B  As Integer
Private Sub Form_Load()
MSComm1.CommPort = 1
MSComm1.PortOpen = True
MSComm1.Settings = "9600,E,7,2"
Combo1.AddItem "ชนิดสินค้าที่ 1"
Combo1.AddItem "ชนิดสินค้าที่ 2"
Combo1.AddItem “ชนิดสินค้าที่ 3"
Combo1.AddItem “ชนิดสินค้าที่ 4"
Combo1.AddItem “ชนิดสินค้าที่ 5"
Combo1.AddItem “ชนิดสินค้าที่ 6"
Combo2.AddItem "0"
Combo2.AddItem "1"
Combo2.AddItem "2"
Combo2.AddItem "3"
Combo2.AddItem "4"
Combo2.AddItem "5"
Combo2.AddItem "6"
Combo2.AddItem "7"
Combo2.AddItem "8"
Combo2.AddItem "9"
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Command1_Click()
Text2.Text = "1"
End Sub
Private Sub Command2_Click()
Text2.Text = "2"
End Sub
Private Sub Command3_Click()
Text2.Text = "3"
End Sub
Private Sub Command4_Click()
Text2.Text = "4"
End Sub
Private Sub Command5_Click()
Text2.Text = "5"
End Sub
Private Sub Command6_Click()
Text2.Text = "6"
End Sub
Private Sub Command7_Click()
Text2.Text = "7"
End Sub
Private Sub Command8_Click()
Text2.Text = "8"
End Sub
Private Sub Command9_Click()
Text2.Text = "9"
End Sub
Private Sub Command10_Click()
If MSComm1.PortOpen = True Then
MSComm1.PortOpen = False
End If
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Unload Me
End Sub
Private Sub Command11_Click()
A = Text2.Text
If A = 1 Then
Text1(0).Text = Combo1.Text + " จำนวน " + Text4.Text + " วันที่จัดเก็บ " + Label2(2).Caption + " เวลา " + Label2(1).Caption
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR001800024E*" & vbCr
shpLED0.BackColor = vbRed
ElseIf A = 2 Then
Text1(1).Text = Combo1.Text + " จำนวน " + Text4.Text + " วันที่จัดเก็บ " + Label2(2).Caption + " เวลา " + Label2(1).Caption
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0018000448*" & vbCr
shpLED1.BackColor = vbRed
ElseIf A = 3 Then
Text1(2).Text = Combo1.Text + " จำนวน " + Text4.Text + " วันที่จัดเก็บ " + Label2(2).Caption + " เวลา " + Label2(1).Caption
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0018000844*" & vbCr
shpLED2.BackColor = vbRed
ElseIf A = 4 Then
Text1(3).Text = Combo1.Text + " จำนวน " + Text4.Text + " วันที่จัดเก็บ " + Label2(2).Caption + " เวลา " + Label2(1).Caption
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR001800104D*" & vbCr
shpLED3.BackColor = vbRed
ElseIf A = 5 Then
Combo1.Text + " จำนวน " + Text4.Text + " วันที่จัดเก็บ " + Label2(2).Caption + " เวลา " + Label2(1).Caption
MSComm1.Output = "@00SC0252*" & vbCr
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MSComm1.Output = "@00WR001800204E*" & vbCr
shpLED4.BackColor = vbRed
ElseIf A = 6 Then
Combo1.Text + " จำนวน " + Text4.Text + " วันที่จัดเก็บ " + Label2(2).Caption + " เวลา "
+ Label2(1).Caption
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0018004048*" & vbCr
shpLED5.BackColor = vbRed
ElseIf A = 7 Then
Combo1.Text + " จำนวน " + Text4.Text + " วันที่จัดเก็บ " + Label2(2).Caption + " เวลา "
+ Label2(1).Caption
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0018008044*" & vbCr
shpLED6.BackColor = vbRed
ElseIf A = 8 Then
Combo1.Text + " จำนวน " + Text4.Text + " วันที่จัดเก็บ " + Label2(2).Caption + " เวลา "
+ Label2(1).Caption
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR001801004D*" & vbCr
shpLED7.BackColor = vbRed
ElseIf A = 9 Then
Combo1.Text + " จำนวน " + Text4.Text + " วันที่จัดเก็บ " + Label2(2).Caption + " เวลา "
+ Label2(1).Caption
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR001802004E*" & vbCr
shpLED8(1).BackColor = vbRed
End If
End Sub
Private Sub Command15_Click()
B = Combo2.Text
If B = 0 Then
Label7.Caption = "คุณไม่ได้ใส่หมายเลขสินค้าออก"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ElseIf B = 1 Then
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR001800024E*" & vbCr
Text1(0).Text = ""
shpLED0.BackColor = vbWhite
ElseIf B = 2 Then
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0018000448*" & vbCr
Text1(1).Text = ""
shpLED1.BackColor = vbWhite
ElseIf B = 3 Then
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0018000844*" & vbCr
Text1(2).Text = ""
shpLED2.BackColor = vbWhite
ElseIf B = 4 Then
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR001800104D*" & vbCr
Text1(3).Text = ""
shpLED3.BackColor = vbWhite
ElseIf B = 5 Then
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR001800204E*" & vbCr
Text1(4).Text = ""
shpLED4.BackColor = vbWhite
ElseIf B = 6 Then
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0018004048*" & vbCr
Text1(5).Text = ""
shpLED5.BackColor = vbWhite
ElseIf B = 7 Then
MSComm1.Output = "@00SC0252*" & vbCr
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

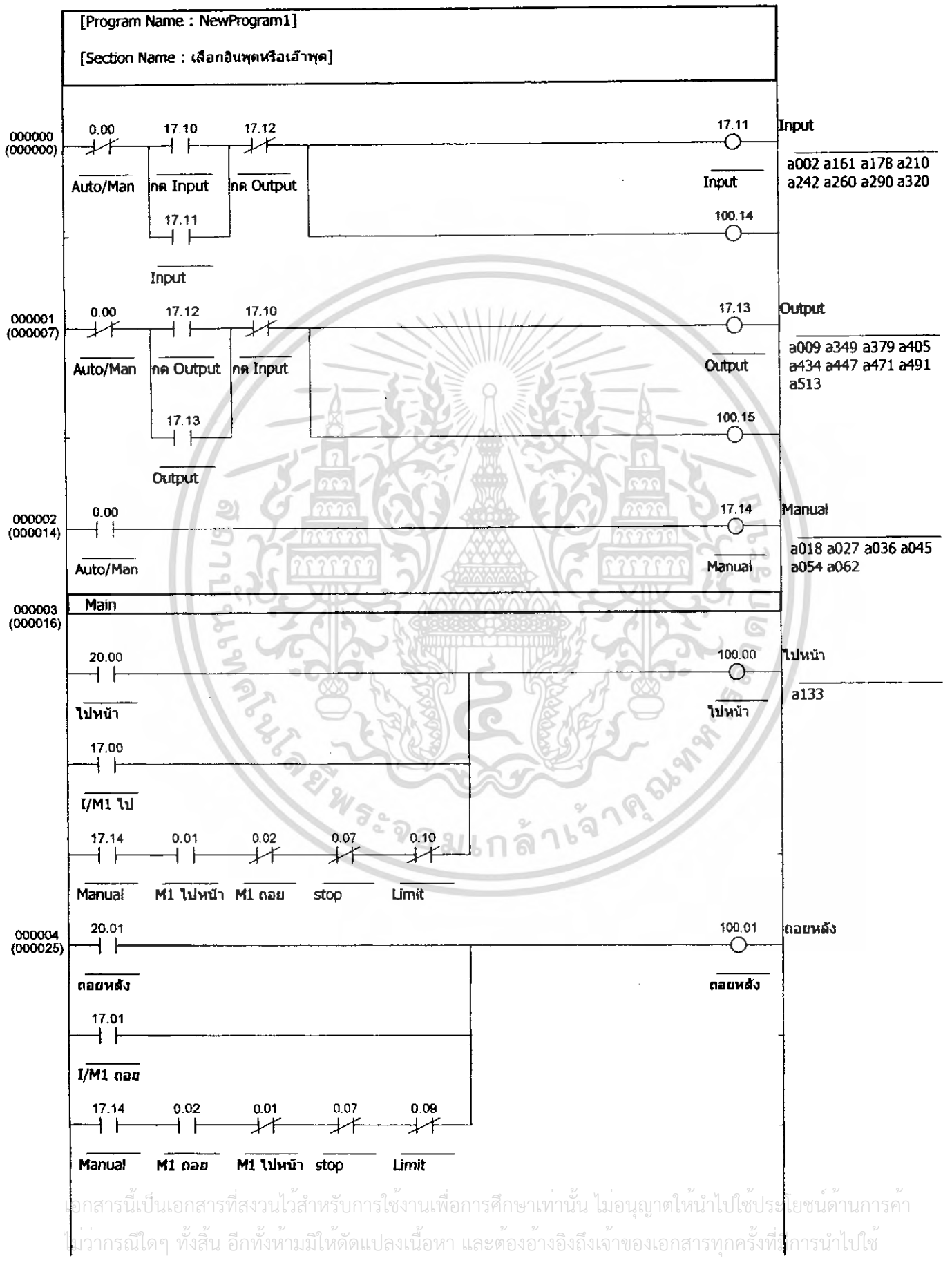
```
MSComm1.Output = "@00WR0018008044*" & vbCr
Text1(6).Text = ""
shpLED6.BackColor = vbWhite
Elself B = 8 Then
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR001801004D*" & vbCr
Text1(7).Text = ""
shpLED7.BackColor = vbWhite
ElseIf B = 9 Then
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR001802004E* & vbCr"
Text1(8).Text = ""
shpLED8(1).BackColor = vbWhite
End If
End Sub
Private Sub Command12_Click()
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0018400048*" & vbCr  'stop
End Sub
Private Sub Command13_Click()
Text2.Text = ""
Text4.Text = ""
End Sub
Private Sub Command14_Click()
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0018800044*" & vbCr  'reset
End Sub
Private Sub Option1_Click()
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0017040047*" & vbCr  'input
End Sub
Private Sub Option2_Click()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

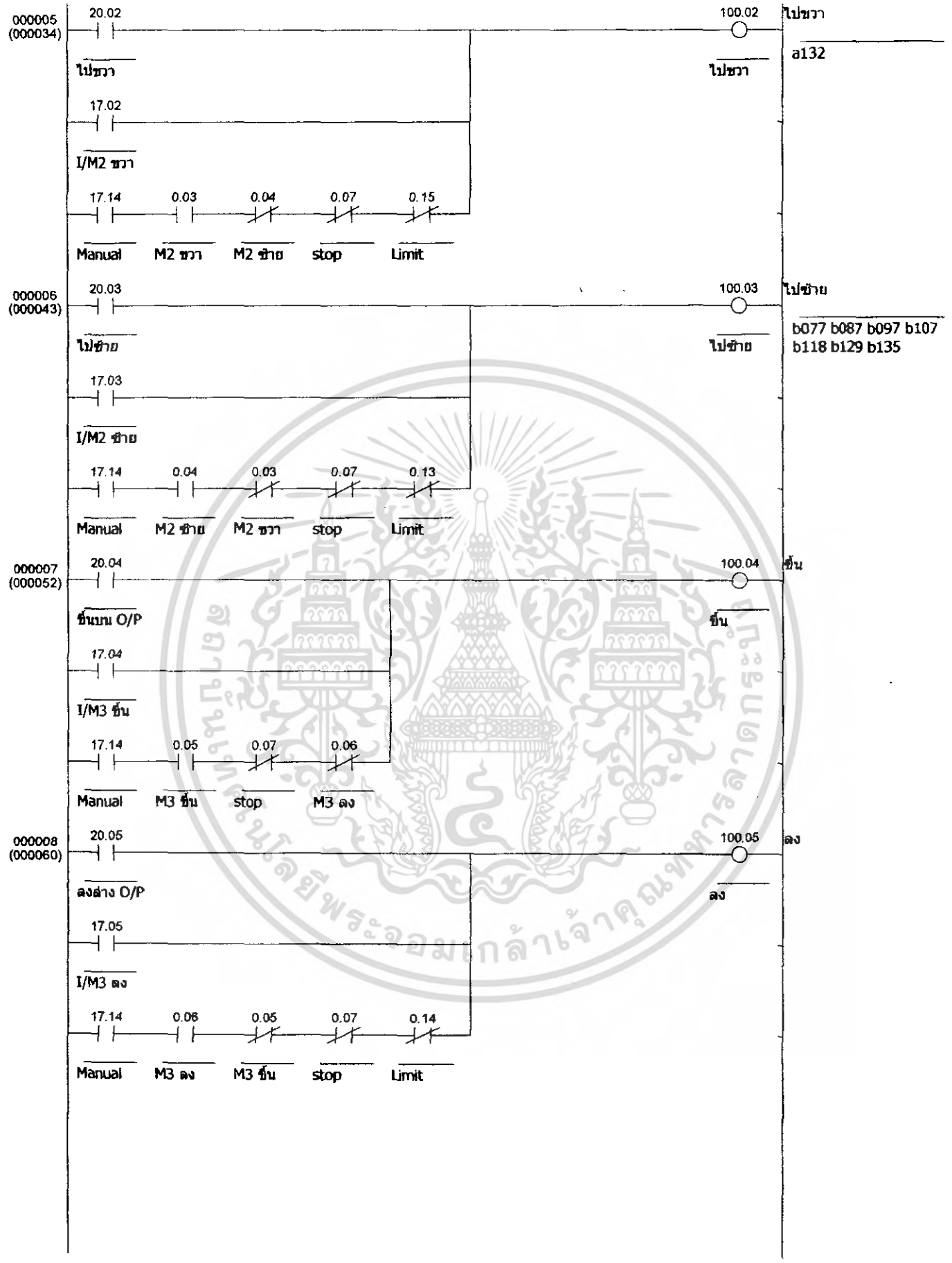
```
MSComm1.Output = "@00SC0252*" & vbCr
MSComm1.Output = "@00WR0017100042*" & vbCr 'output
End Sub
Private Sub Timer1_Timer()
 N = Time$
 Label2(1).Caption = N
 M = Date$
Label2(2).Caption = M
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

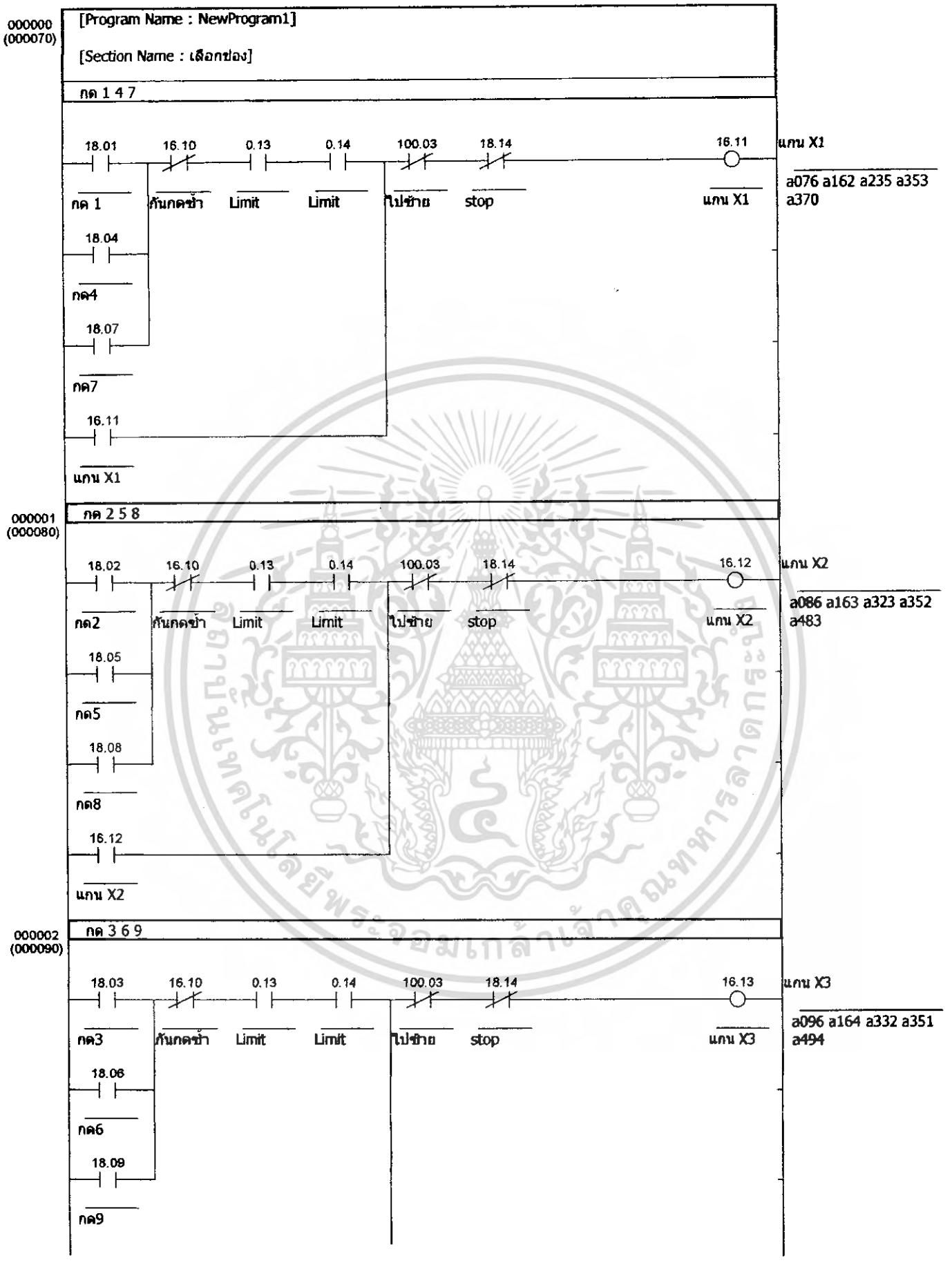
ภาษาแลดเดอร์ไดอะแกรม



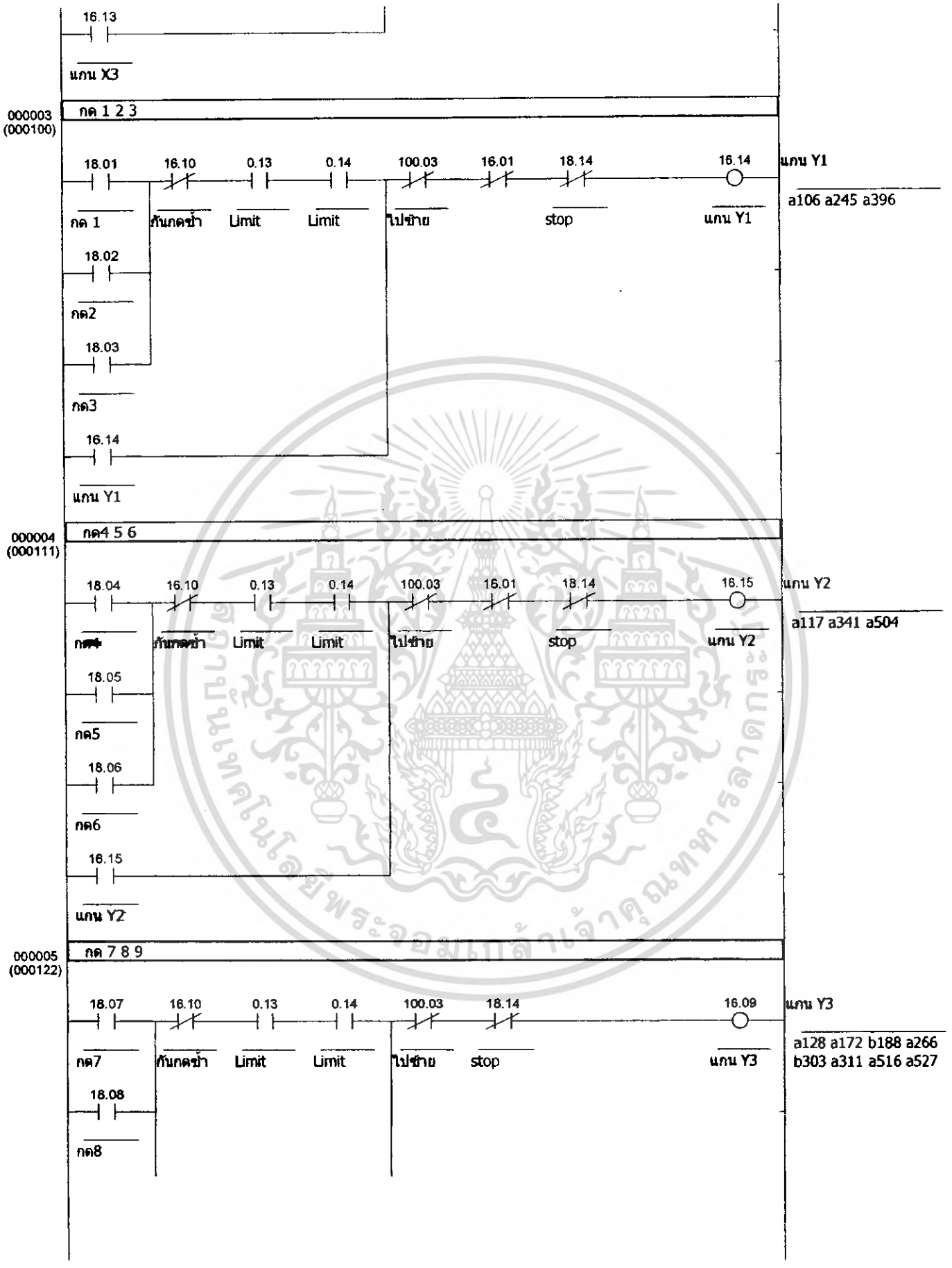
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



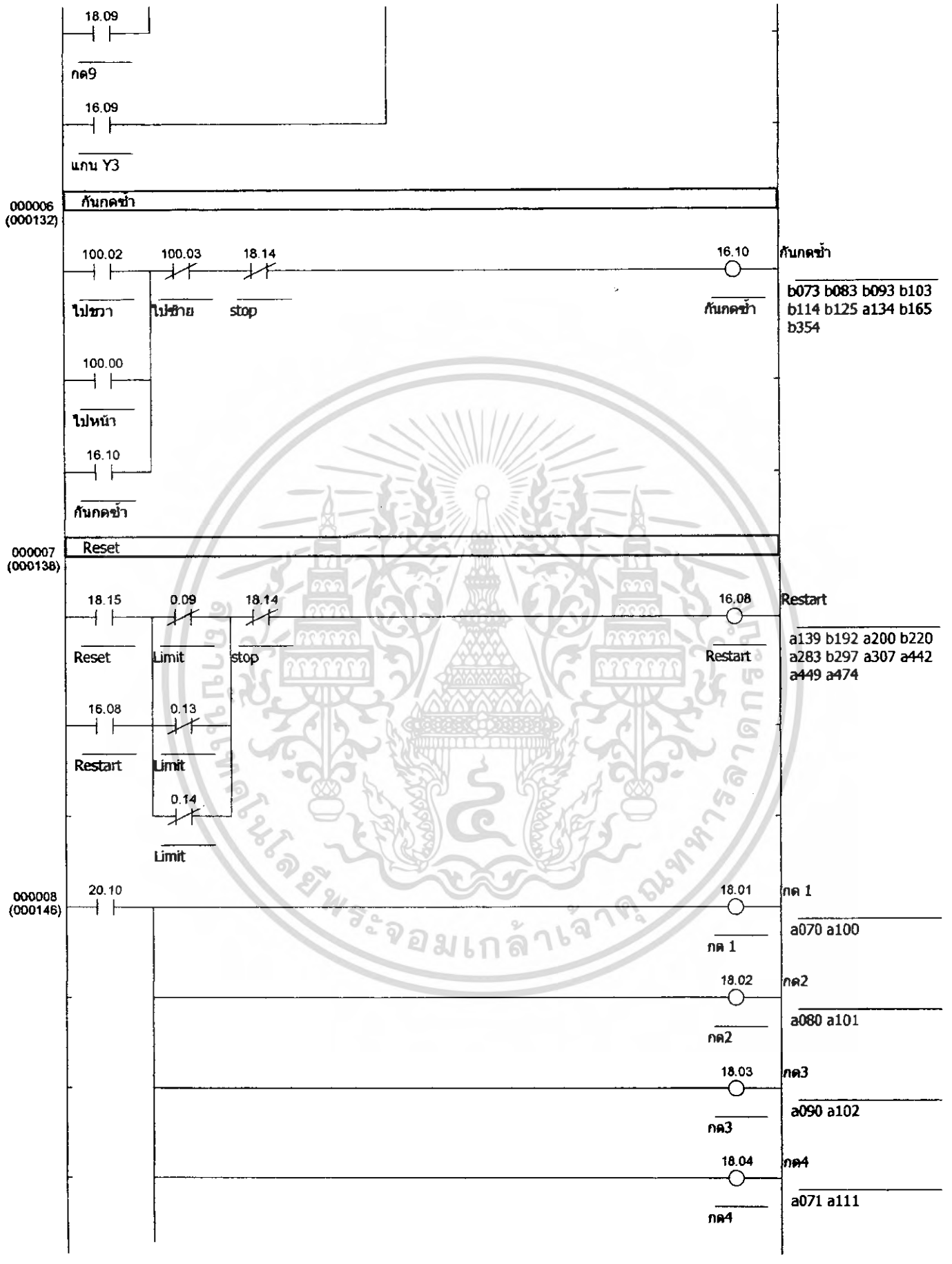
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



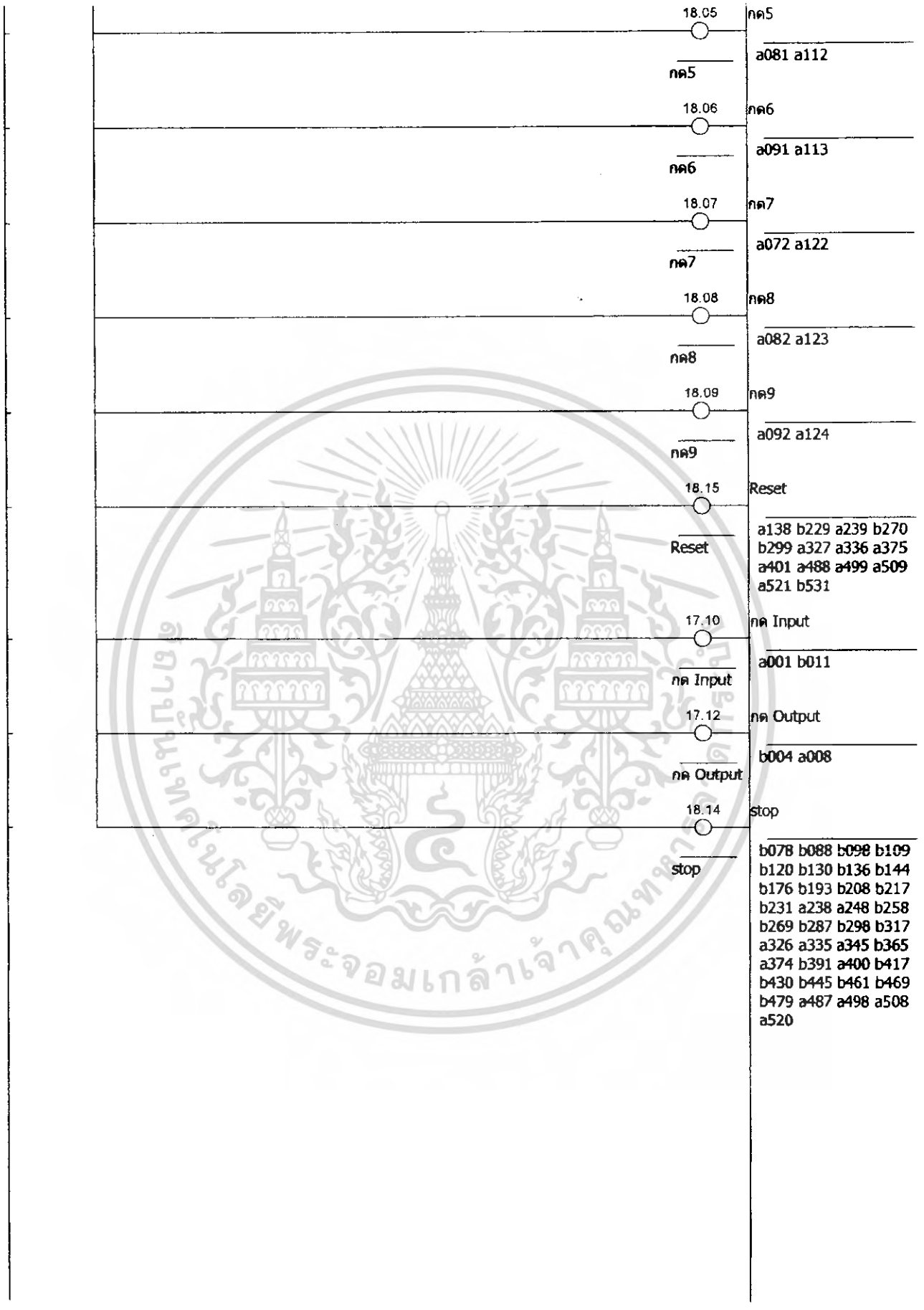
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



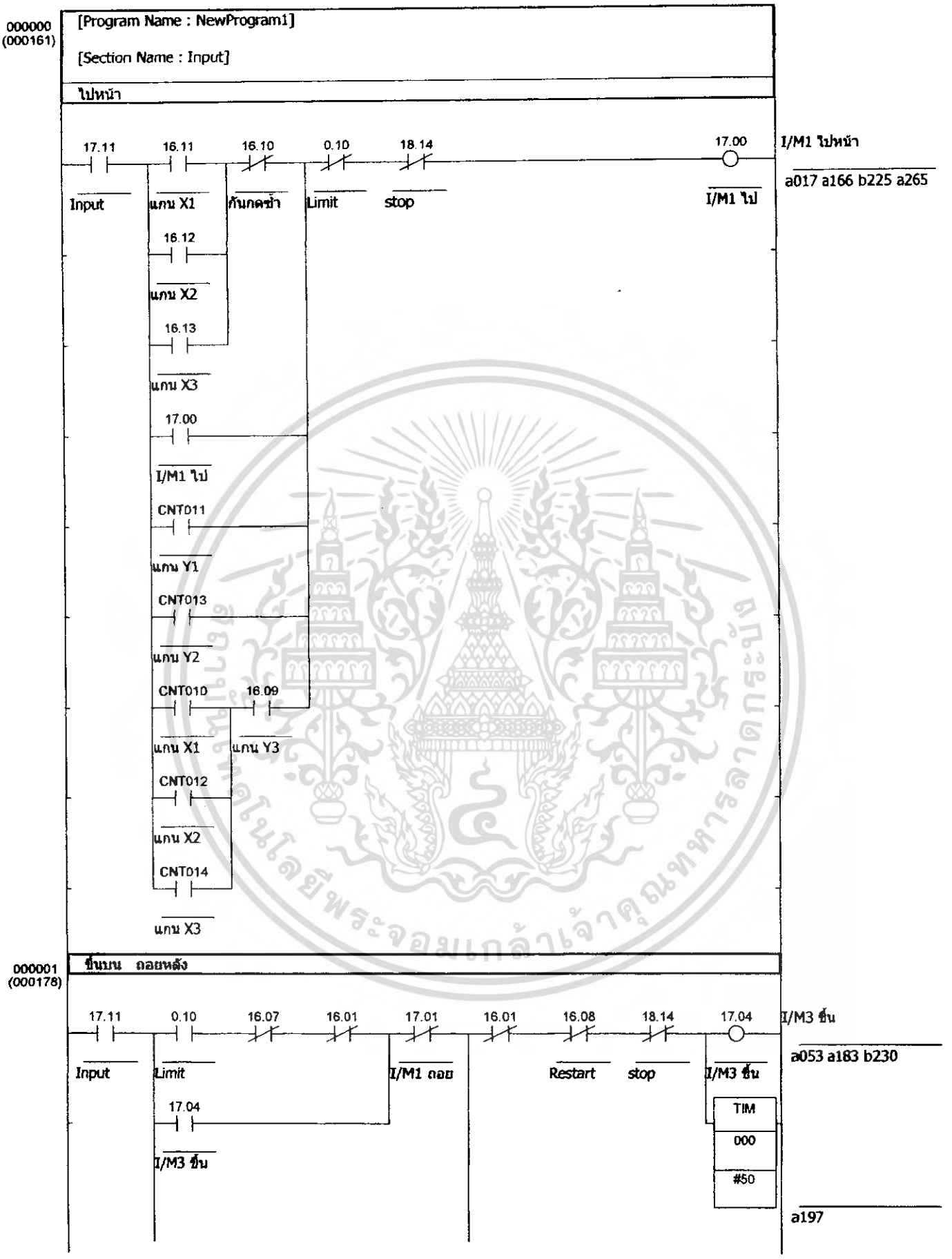
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



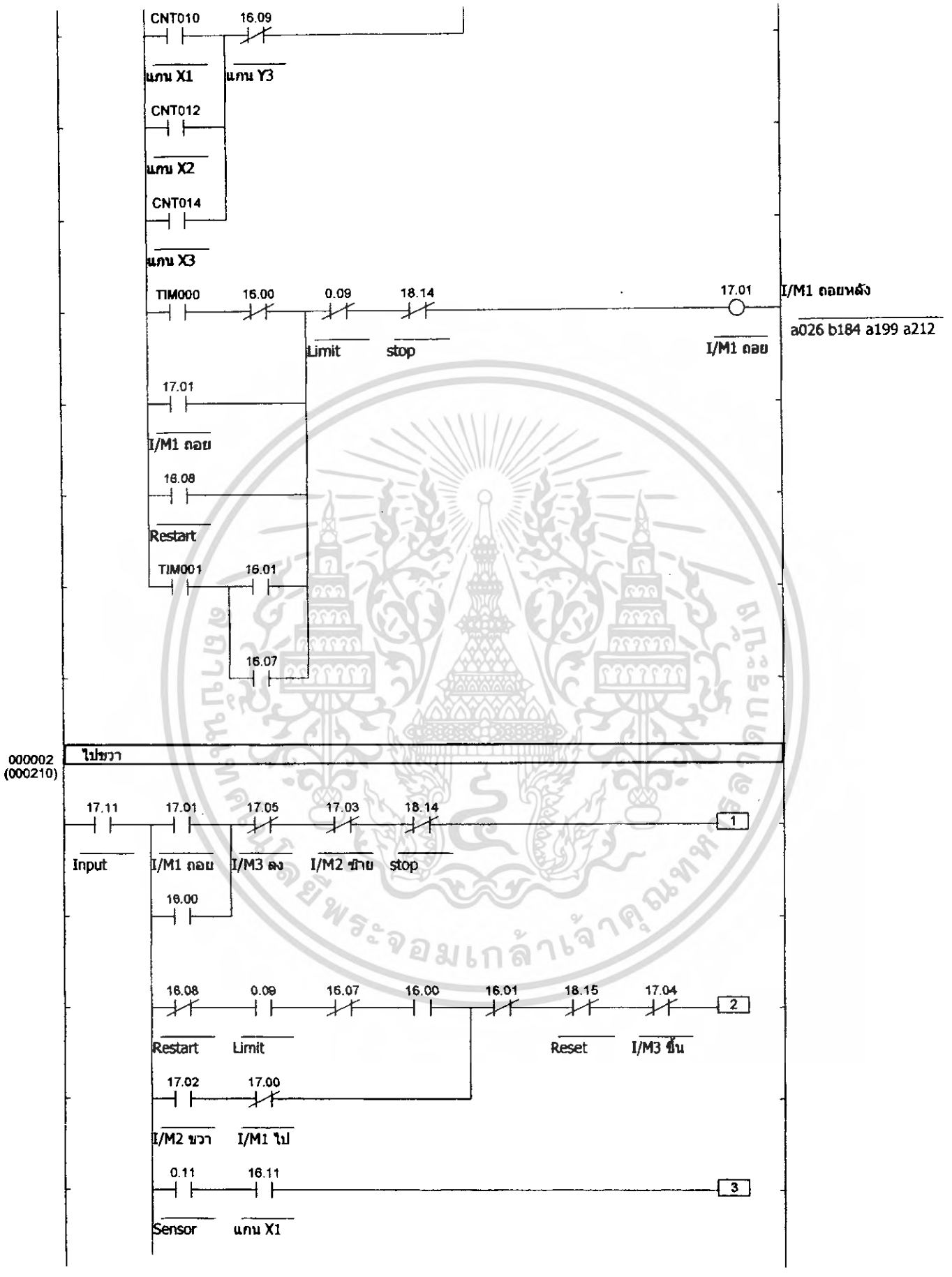
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



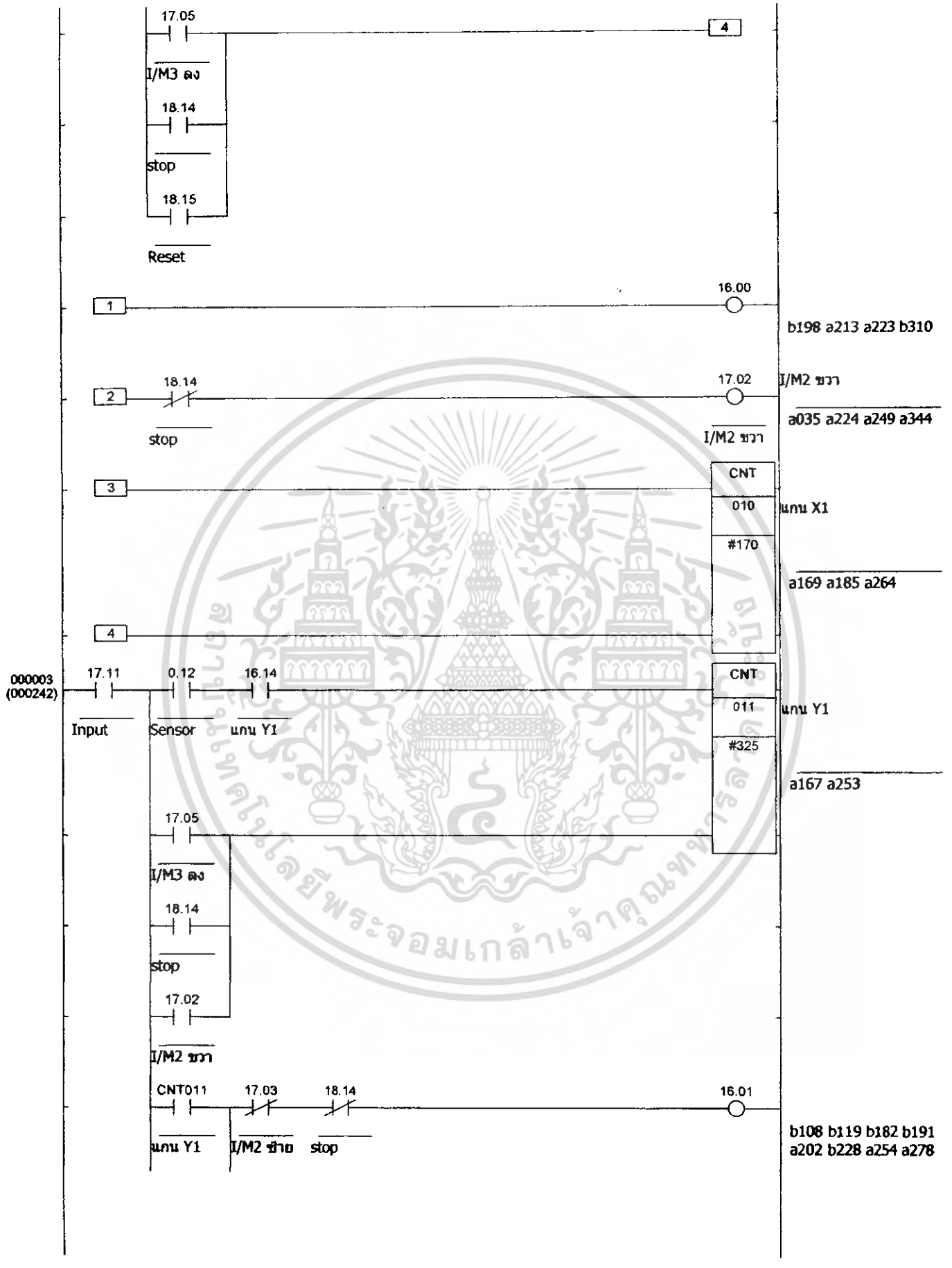
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



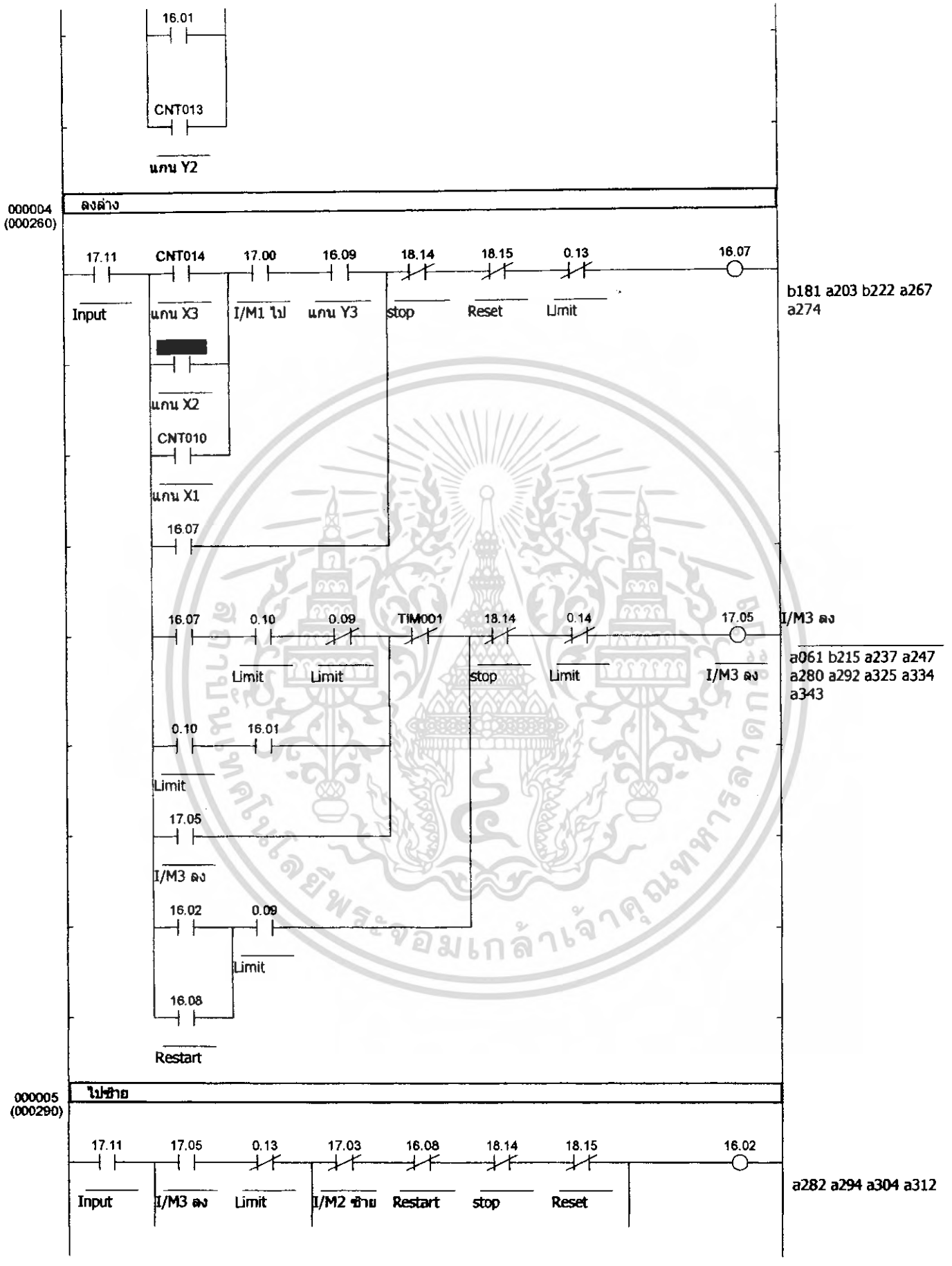
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



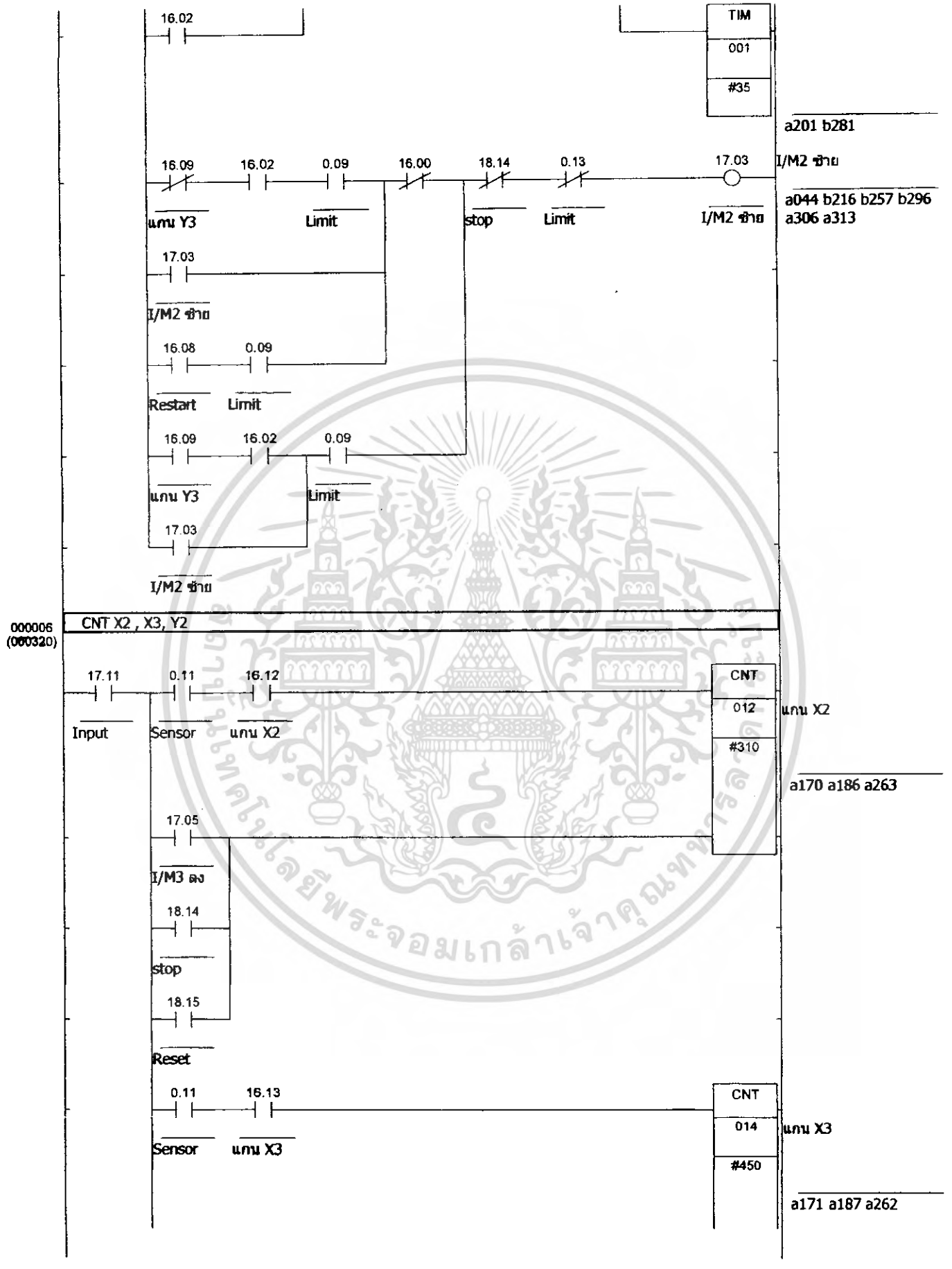
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



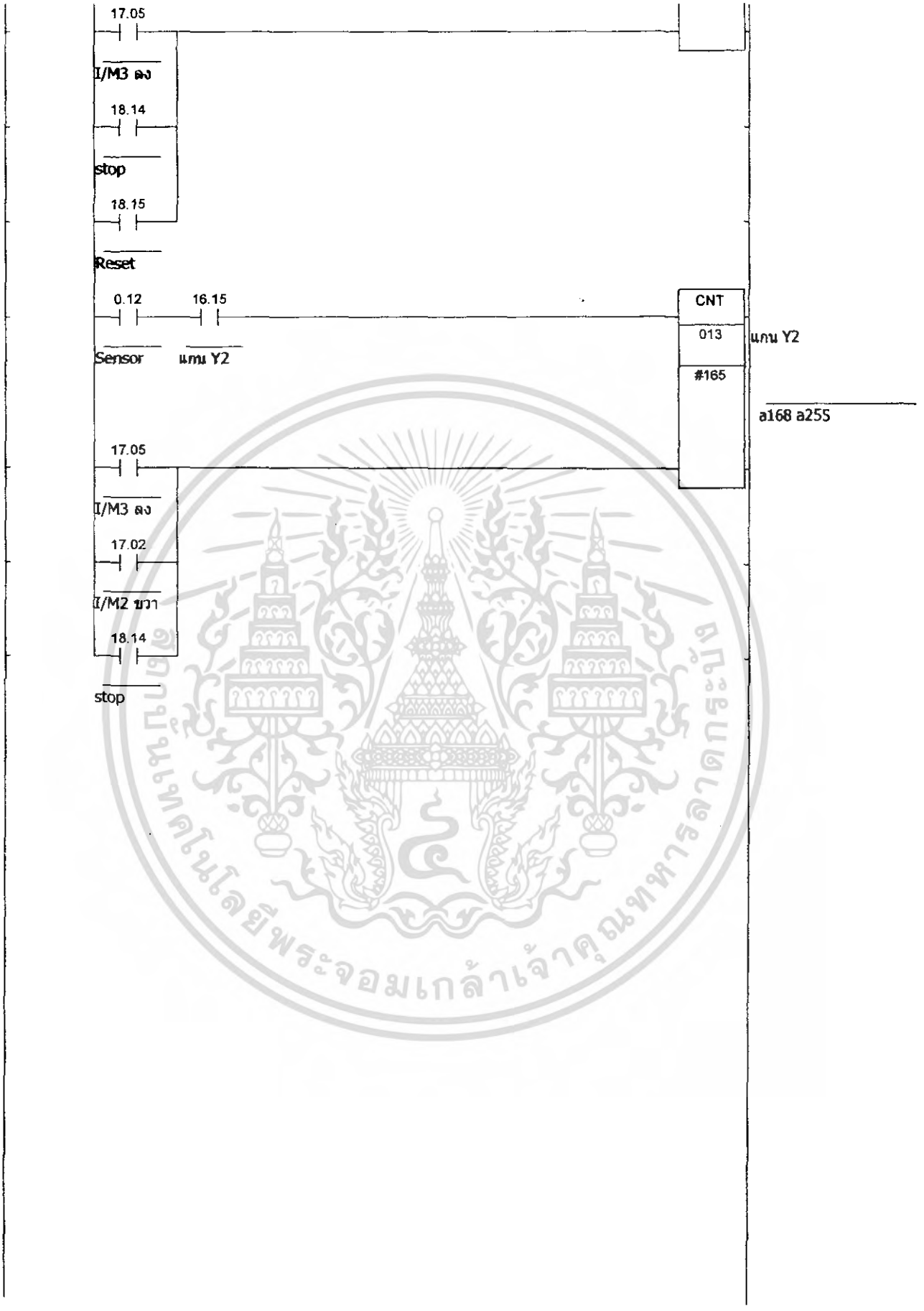
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



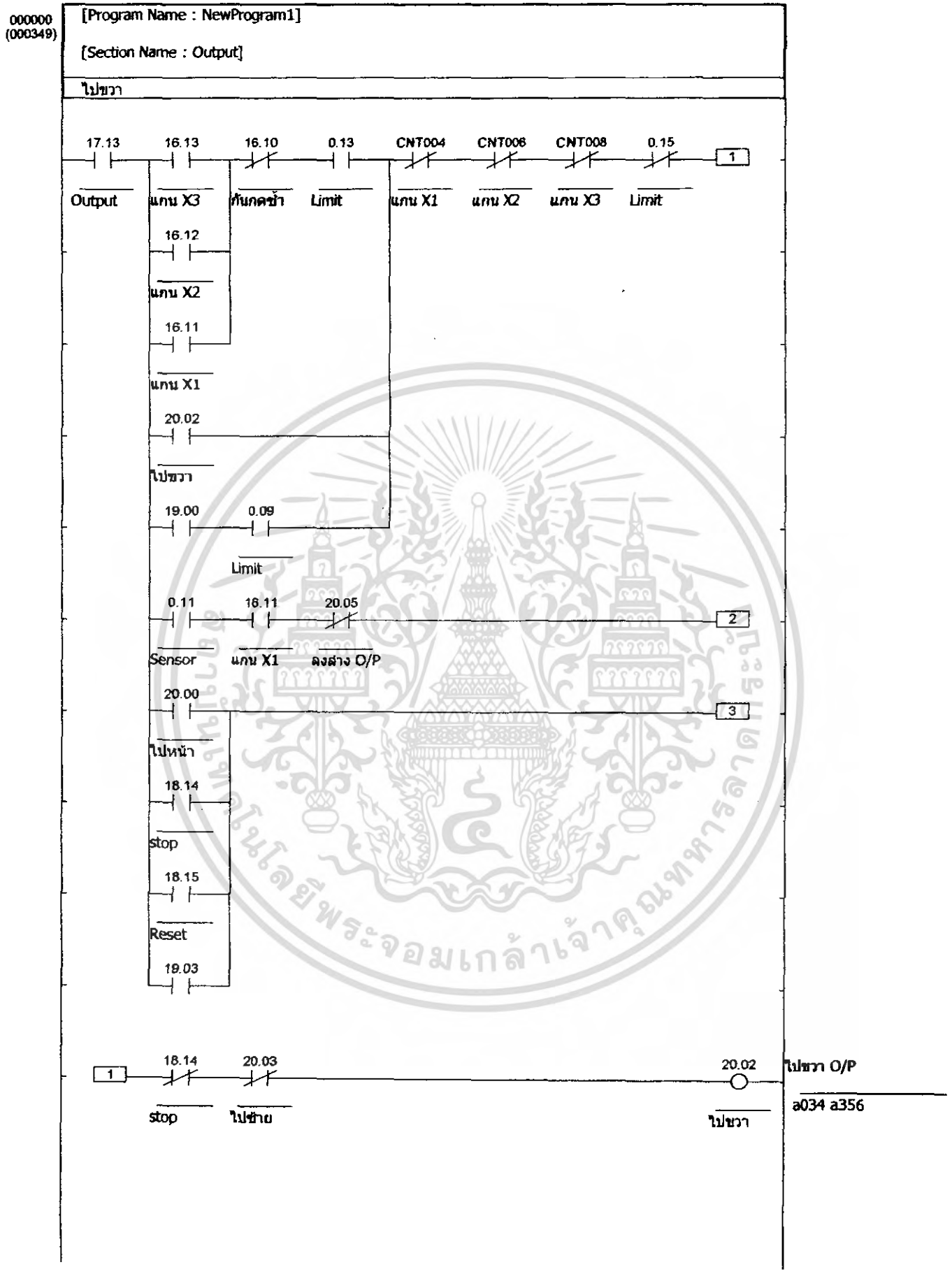
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



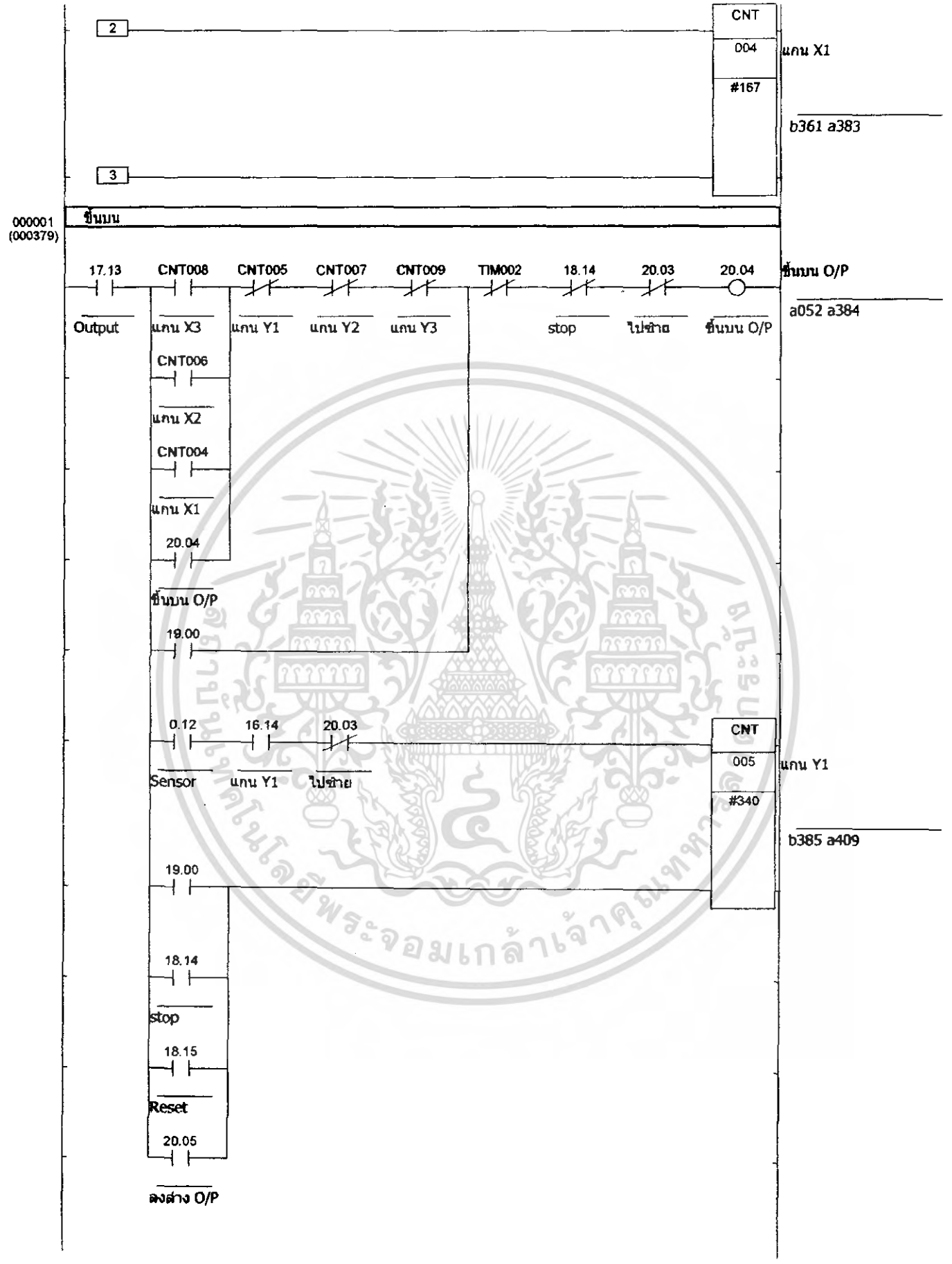
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



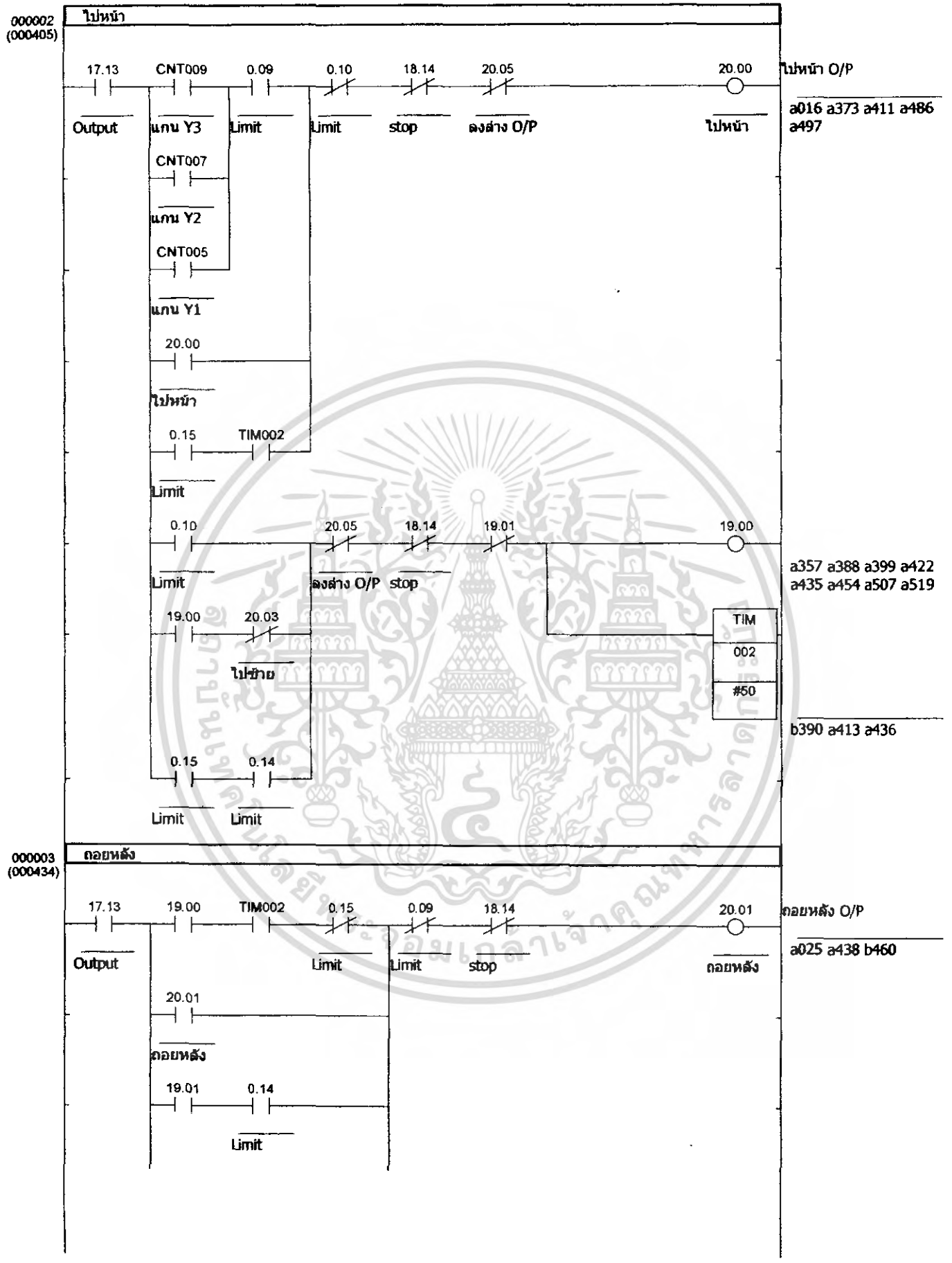
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



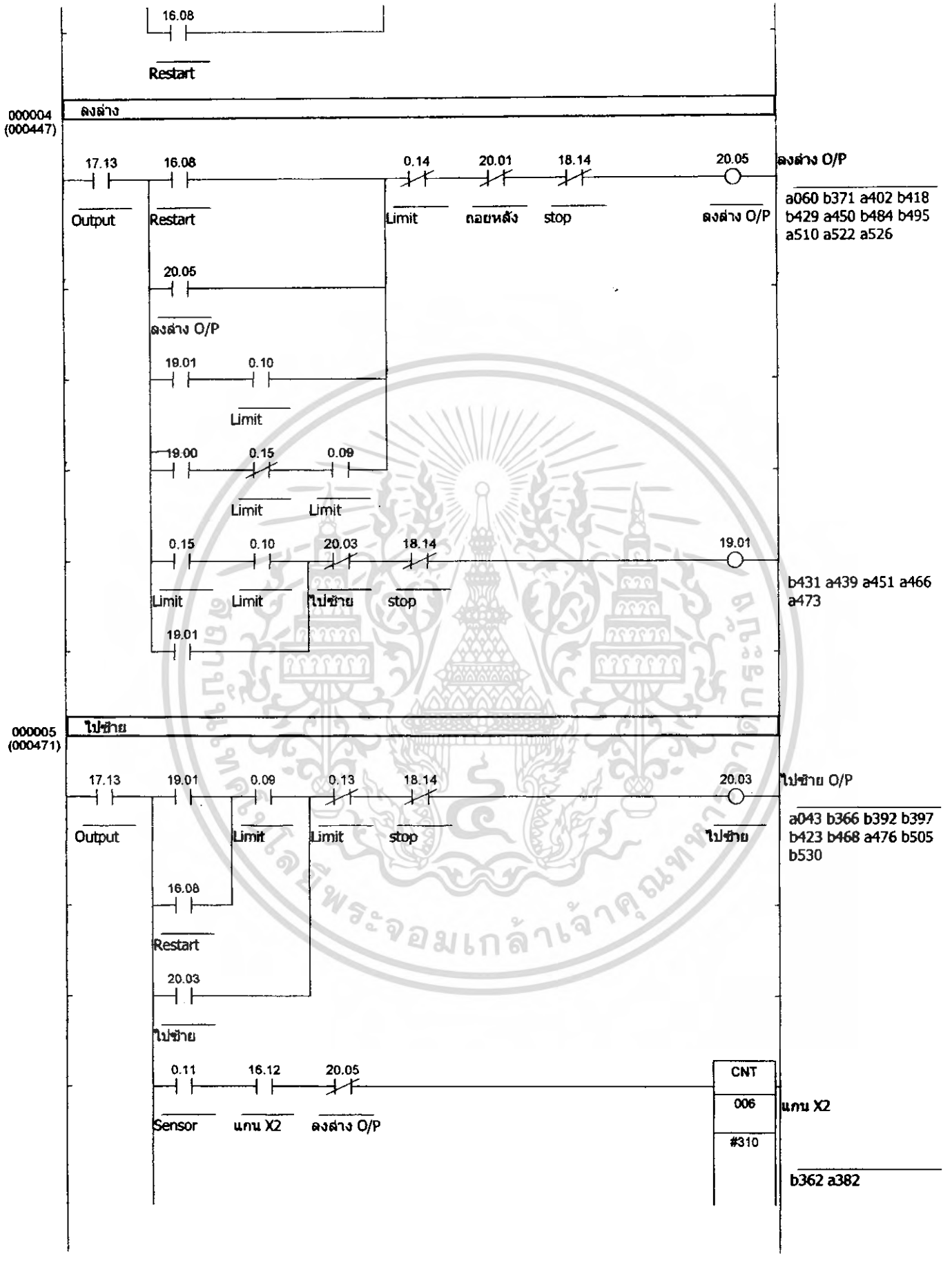
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



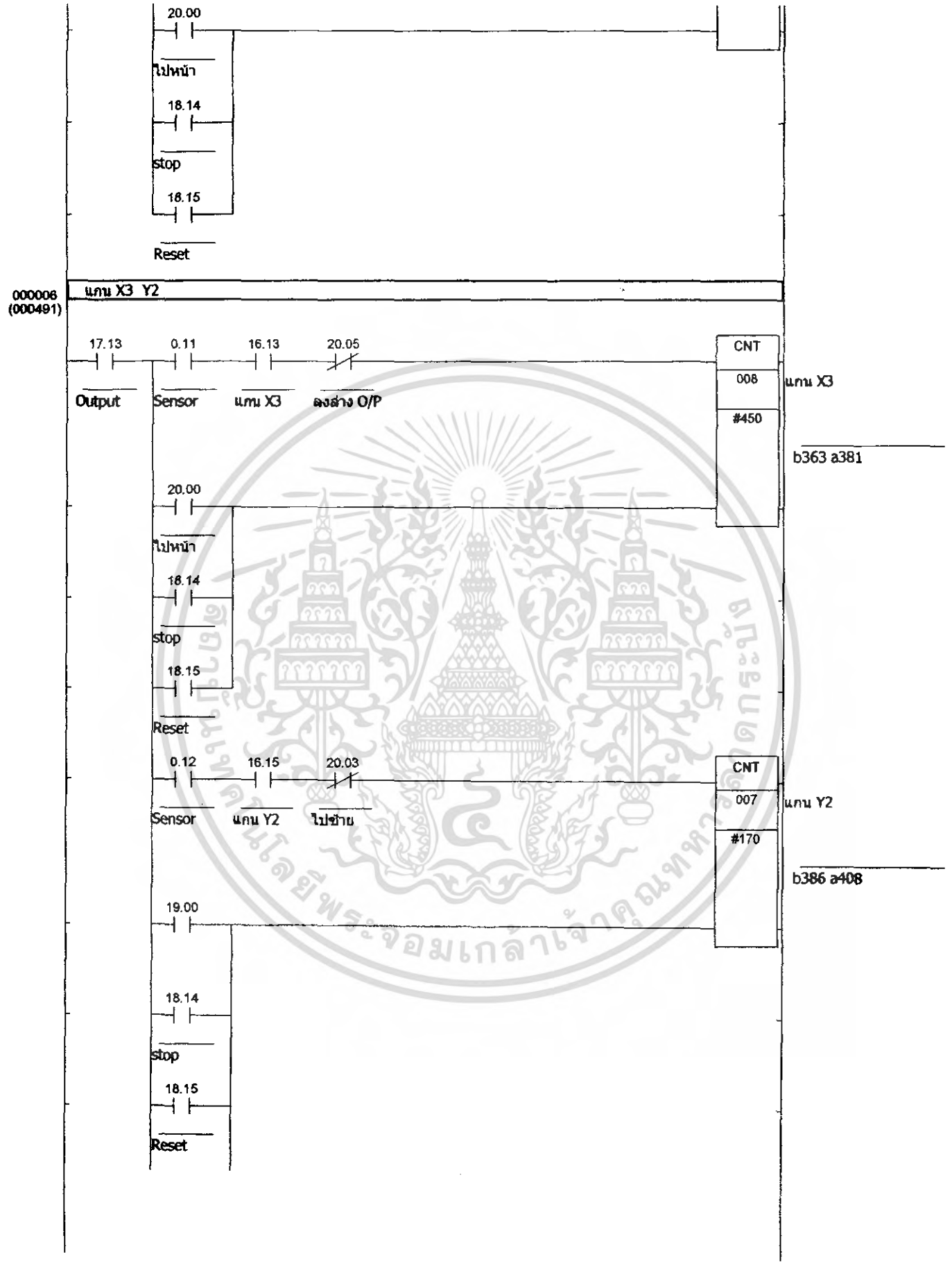
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



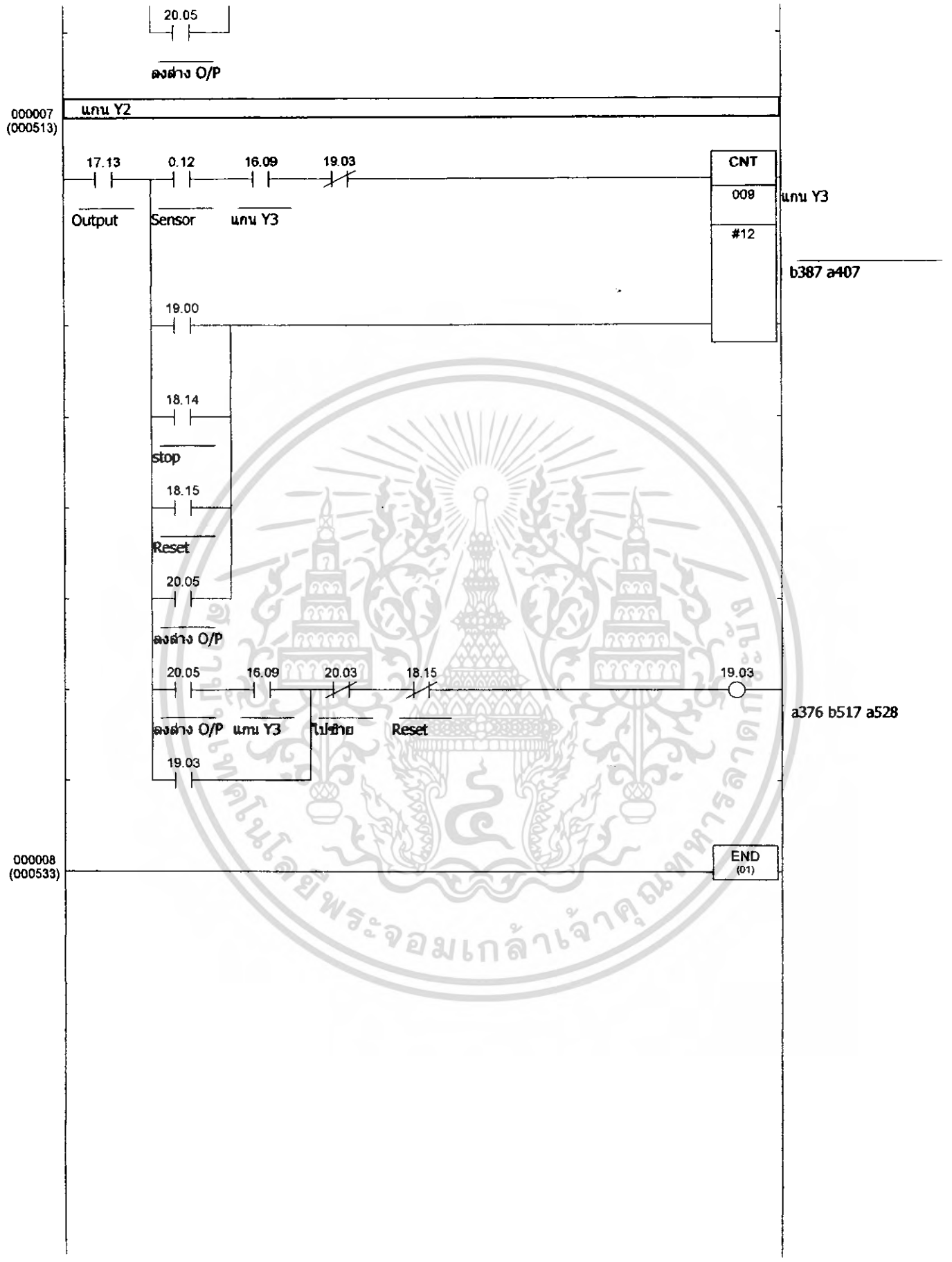
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

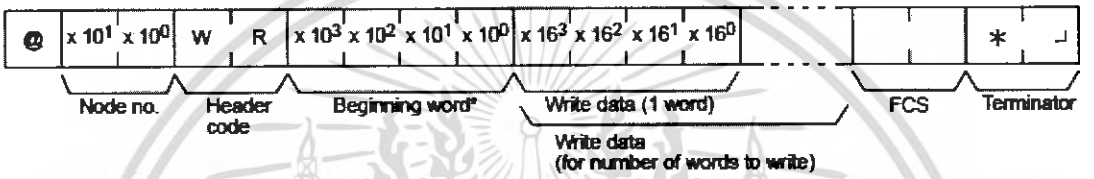
ภาคผนวก ข.

Host Link Commands

IR/SR AREA WRITE — WR

Writes data to the IR and SR areas, starting from the specified word. Writing is done word by word.

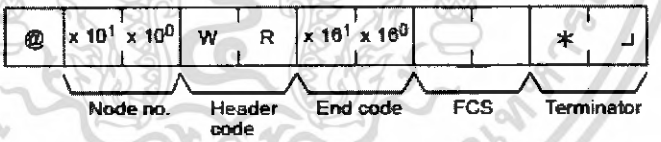
Command Format



Note 1. Beginning word: 0000 to 0252 in CQM1 PCs, 0000 to 0019 and 0200 to 0252 in CPM1/CPM1A/SRM1 PCs.

2. Divide the command when writing more than 30 words of data.

Response Format An end code of 00 indicates normal completion.



Note Words 0020 to 0199 in CPM1/CPM1A/SRM1 PCs cannot be specified. If an attempt to write to any of these words is made, the writing operations will not be executed and normal completion occurs.

Parameters **Write Data (Command)**

Specify in order the contents of the number of words to be written to the IR or SR area in hexadecimal, starting with the specified beginning word.

Note If data is specified for writing which exceeds the allowable range, an error will be generated and the writing operation will not be executed. If, for example, 252 is specified as the

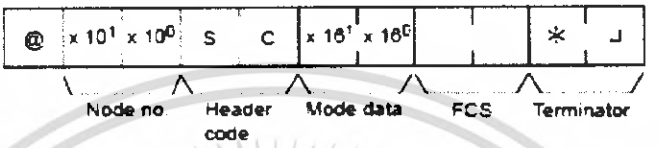
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

beginning word for writing, and two words of data are specified, then 253 will become the last word for writing data, and the command will not be executed because SR 253 is beyond the writable range.

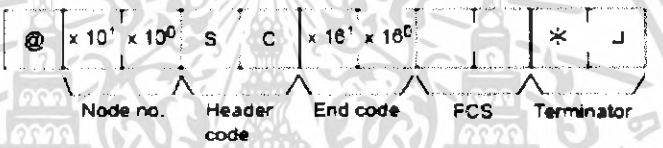
STATUS WRITE — SC

Changes the PC operating mode.

Command Format



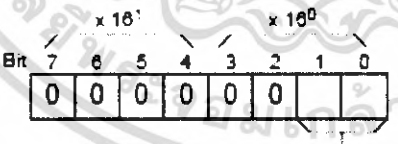
Response Format An end code of 00 indicates normal completion.



Parameters

Mode Data (Command)

"Mode data" consists of two digits (one byte) hexadecimal. With the leftmost two bits, specify the PC operating mode. Set all of the remaining bits to "0."



Bit		Operation mode
1	0	
0	0	PROGRAM mode
1	0	MONITOR mode
1	1	RUN mode

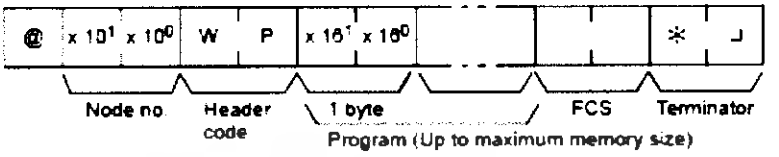
This area is different from that of STATUS READ.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

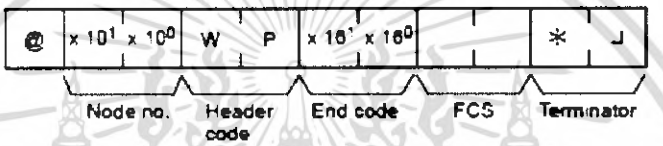
PROGRAM WRITE — WP

Writes to the PC user's program area the machine language (object code) program transmitted from the host computer. The contents are written as a block, from the beginning.

Command Format



Response Format An end code of 00 indicates normal completion.



Parameters Program (Command) Program data up to the maximum memory size.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

Memory Areas

CPM1/CPM1A and CQM1 Memory Area Comparison

This table shows the differences between the CPM1/CPM1A and CQM1 memory areas.

Data area		CPM1/CPM1A	CQM1
IR area	Input area	IR 000 to IR 009	IR 000 to IR 015
	Output area	IR 010 to IR 019	IR 100 to IR 115
	Work areas and special areas	IR 200 to IR 231 (IR 020 to IR 199 cannot be used.)	IR 016 to IR 099 IR 116 to IR 144
SR area		SR 232 to SR 255	SR 244 to SR 255
HR area		HR 00 to HR 19 (HR 20 to HR 99 cannot be used.)	HR 00 to HR 99
AR area		AR 00 to AR 15 (AR 16 to AR 27 cannot be used.)	AR 00 to AR 27
LR area		LR 00 to LR 15 (LR 16 to LR 63 cannot be used.)	LR 00 to LR 63
Timer/Counter area		TC 000 to TC 127 (TC 128 to TC 511 cannot be used.)	TC 000 to TC 511
DM area	Read/write	DM 0000 to DM 0999 DM 1022 to DM 1023 (DM 1024 to DM 6143 cannot be used.)	DM 0000 to DM 1023 DM 1024 to DM 6143
	Error history	DM 1000 to DM 1021	DM 6569 to DM 6599
	Read-only	DM 6144 to DM 6599	DM 6144 to DM 6568
	PC Setup	DM 6600 to DM 6655	DM 6600 to DM 6655

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.

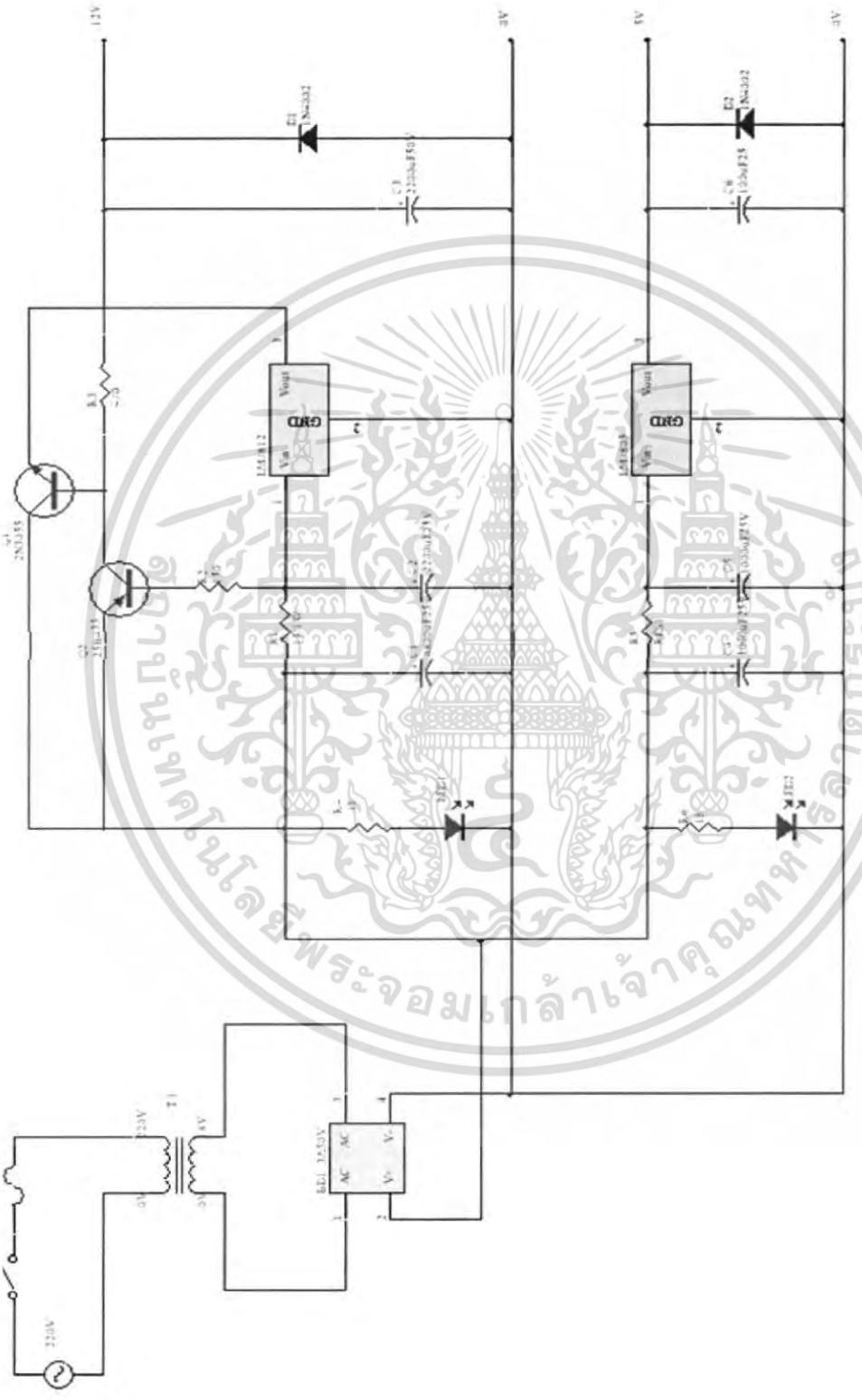
Extended ASCII

Right digit	Left digit												
	0, 1, 8, 9	2	3	4	5	6	7	A	B	C	D	E	F
0			0	@	P	`	p		-	@	P	`	p
1		!	1	A	Q	a	q	!	1	A	Q	a	q
2		"	2	B	R	b	r	"	2	B	R	b	r
3		#	3	C	S	c	s	#	3	C	S	c	s
4		$	4	D	T	d	t	$	4	D	T	d	t
5		%	5	E	U	e	u	%	5	E	U	e	u
6		&	6	F	V	f	v	&	6	F	V	f	v
7		'	7	G	W	g	w	'	7	G	W	g	w
8		(	8	H	X	h	x	(	8	H	X	h	x
9		)	9	I	Y	i	y	)	9	I	Y	i	y
A		*	:	J	Z	j	z	*	:	J	Z	j	z
B		+	;	K	[	k	{	+	;	K	[	k	{
C		,	<	L	¥	l	|	,	<	L	¥	l	|
D		-	=	M	]	m	}	-	=	M	]	m	}
E		.	>	N	^	n	→	.	>	N	^	n	
F		/	?	O	_	o	←	/	?	O	_	o	→

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ.

วงจรและแหล่งจ่ายไฟ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้