

ระบบสั่งงานพีแอลซีเพื่อควบคุมกระบวนการทางเคมีโดยใช้โทรศัพท์เคลื่อนที่

Chemical Process Control System using Mobile Phone



โดย

นาย ทินกร ลิขิตวัฒน์

นาย เจนวิทย์ วิจิตต์พันธ์

นาย เอกลักษณ์ วรศรี

เลขที่.....
ร.ร. 467
1017

เลขหมู่.....
เลขทะเบียน..... **62037**
วัน,เดือน,ปี..... **27 ก.ค. 2549**

b..... 101746730
i.....

ปริญญานิพนธ์นี้ส่วนหนึ่งของการศึกษาตามศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Chemical Process Control System using Mobile Phone



**A THESIS SUBMITTED IN PARTIAL OF THE REQUIREMENT FOR THE DEGREE OF
BACHLOR IN DEPARTMENT OF INFORMATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2004

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบเสนอปริญญาบัตร

หัวข้อปริญญาบัตร	ระบบสั่งงานพีแอลซีเพื่อควบคุมกระบวนการทางเคมีโดยใช้โทรศัพท์เคลื่อนที่	
ชื่อนักศึกษา	นาย ทินกร สิววัฒน์	รหัสประจำตัว 45015794
	นาย เจนวิทย์ วิจิตต์พันธ์	รหัสประจำตัว 45015840
	นาย เอกลักษณ์ วรรณศรี	รหัสประจำตัว 45015882
อาจารย์ที่ปรึกษา	รศ.ดร.ปิติเขต สุรักษา	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต	
ภาควิชา	สาขาวิศวกรรมสารสนเทศ	
ปีการศึกษา	2547	

ปริญญาบัตรฉบับนี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว

(รศ.ดร.ปิติเขต สุรักษา)

อาจารย์ที่ปรึกษาปริญญาบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบสั่งงานพีแอลซีเพื่อควบคุมกระบวนการทางเคมีโดยใช้โทรศัพท์เคลื่อนที่	
ชื่อนักศึกษา	นาย ทินกร สิจิววัฒน์	รหัสนักศึกษา 45015794
	นาย เจนวิทย์ วิจิตต์พันธ์	รหัสนักศึกษา 45015840
	นาย เอกลักษณ์ วรรณศรี	รหัสนักศึกษา 45015882
อาจารย์ที่ปรึกษา	รศ.ดร. ปิติเขต สุรรักษา	
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ภาควิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2547	

บทคัดย่อ

ในปัจจุบันพีแอลซีได้เข้ามามีบทบาทสำคัญในวงการอุตสาหกรรมเป็นอย่างมากเนื่องจากสามารถใช้งานควบคุมอุปกรณ์ต่างๆ ในโรงงานอุตสาหกรรมได้อย่างหลากหลายเพราะสามารถเขียนโปรแกรมควบคุมจากผู้ใช้ได้เองและสามารถแก้ไขโปรแกรมที่เขียนขึ้นได้ง่ายทำให้ใช้ได้กับอุปกรณ์ในระบบควบคุมได้ในหลายลักษณะ

ปริญญานิพนธ์นี้กล่าวถึงการประยุกต์เทคโนโลยีการสื่อสารไร้สายของโทรศัพท์เคลื่อนที่โดยใช้เทคโนโลยีของ แวบ (Wab) มาใช้งานร่วมกับโปรแกรม วิชาลเบสิก (Visual Basic) เพื่อสื่อสารกับเครื่องพีแอลซีที่ใช้ควบคุมอุปกรณ์ต่างในการควบคุมกระบวนการทางเคมี เพื่อให้สามารถควบคุมได้ในระยะไกลเท่าที่โทรศัพท์เคลื่อนที่ที่สามารถใช้งานได้

Thesis Title Chemical Process Control System using Mobile Phone
Student MR.TINNAKORN SIKHIWAT
 MR. JANEWIT WIJITPAN
 MR. AKKLUCK WORASRI
Advisor Assoc.Prof.Dr.PITIKHATE SOORAKSA
Graduate Level Bachelor Degree of Information Engineering
Department Information Engineering
Academic Year 2004

ABSTRACT

At the present, PLCs are important devices in industry because it is variously used for controlling many devices in factories because they can be programmed, debugged and controlled by operators, in control system processes. To extend the capability of the control system by utilizing IT technology, this thesis presents an application of wireless technology via mobile phone and WAP technology together with Visual Basic to implement a communication unit with PLCs for a chemical process.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี เป็นผลเนื่องมาจากความร่วมมือของคณะผู้จัดและความมานะอดทน ตลอดจนความช่วยเหลือเกื้อกูลกันของเพื่อนๆ ทุกคนที่มีส่วนร่วม ตลอดระยะเวลาที่ทำโครงการนี้ ได้รับการสนับสนุนและให้คำชี้แนะอันดีจาก รศ.ดร. ปิติเขต ผู้รักษาและขอกราบขอบคุณอาจารย์ทุกท่านที่ได้สั่งสอนวิชาความรู้ไว้ ณ โอกาสนี้ด้วย

สุดท้ายนี้ขอกราบขอบพระคุณบิดา มารดา ที่ให้โอกาสทางการศึกษา และตลอดจนถึงผู้มีบุญคุณทุกท่านที่คอยช่วยเหลือสนับสนุนตลอดมา



ผู้จัดทำ

นายทินกร สิจิวัดน์
นายเจนวิทย์ วิจิตต์พันธ์
นายเอกลักษณ์ วรศรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อไทย	ก
บทคัดย่ออังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูปภาพ	ฉ
บทที่ 1 บทนำ	1
1.1 แนวคิดและที่มาของโครงการ	1
1.2 จุดประสงค์	1
1.3 ขอบเขตของโครงการ	2
1.4 ผลที่คาดว่าจะได้รับ	2
1.5 วิธีการดำเนินงาน	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 โปรแกรมเมเบิลลอจิกคอนโทรลเลอร์ Programmable Logic Controller (PLC)	3
2.2 อาร์ทีดี(RTD)	16
2.3 WAP (wireless Application Protocol)	18
2.4 XML (Extensible Markup Language)	23
2.5 WML (Wiress Markup Language)	24
2.6 โปรแกรม Web Server	25
2.7 PHP Script	25
บทที่ 3 การดำเนินงานวิจัย	27
3.1 การออกแบบทางด้านฮาร์ดแวร์	27
3.2 การออกแบบการควบคุมอุปกรณ์ทางด้านฮาร์ดแวร์ด้วย PLC	28
3.3 การออกแบบส่วนการสั่งงาน PLC ด้วยโปรแกรม Visual Basic	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

เรื่อง	หน้า
3.4 การออกแบบในส่วนที่ติดต่อกันระหว่างโทรศัพท์เคลื่อนที่กับข้อมูลที่เป็น Input และ Output	32
3.5 การออกแบบ WAP Site	33
3.6 การออกแบบตารางเก็บข้อมูล	38
บทที่ 4 การทดลองและผลการทดลอง	41
4.1 ขั้นตอนการตรวจสอบควบคุมอุปกรณ์โดยใช้โปรแกรมควบคุมจากคอมพิวเตอร์ ในแบบ Manual	41
4.2 การควบคุมแบบอัตโนมัติ	43
4.3 การทดลองการควบคุมโดยโทรศัพท์เคลื่อนที่	49
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง	55

สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 ส่วนประกอบด้านหน้าของ PLC	6
รูปที่ 2.2 การต่อแหล่งจ่ายไฟฟ้าภายนอก	6
รูปที่ 2.3 การติดต่อระหว่าง PC กับ PLC	7
รูปที่ 2.4 การติดต่อกับพอร์ตภายนอก	7
รูปที่ 2.5 การแสดงผลของ PLC รุ่น KOYO DL05	8
รูปที่ 2.6 หน้าต่างแรกของโปรแกรม Direct SOFT32	9
รูปที่ 2.7 หน้าต่าง New Project	10
รูปที่ 2.8 หน้าต่างการ Link Computer กับ PLC	11
รูปที่ 2.9 หน้าต่างโปรแกรมเลือกอ่านค่าหรือบันทึกค่าลง PLC	11
รูปที่ 2.10 หน้าต่างของโปรแกรมที่พร้อมจะทำการ Run โปรแกรม	12
รูปที่ 2.11 หน้าต่างโปรแกรม DsData Server สำหรับการตั้งค่าการเชื่อมต่อ	13
รูปที่ 2.12 Analog Input Module 4 Channel	14
รูปที่ 2.13 โครงสร้างภายในของ Analog Input Module 4 Channel	15
รูปที่ 2.14 ตัวเครื่อง PLC DL-05 เมื่อติดตั้ง Analog Input Module 4 Channel	15
รูปที่ 2.15 วงจรบริดจ์ที่ใช้ร่วมกับอาร์ทีดีในการวัดค่าความต้านทาน	17
รูปที่ 2.16 เลขอร์ของโปรโตคอลต่างในระบบ WAP	19
รูปที่ 2.17 การใช้โปรโตคอลที่ทำหน้าที่รักษาความปลอดภัย	21
รูปที่ 2.18 การทำงานของ WAP Gateway	21
รูปที่ 2.19 WAP Gateway ที่ทำหน้าที่อีกอย่างเป็น proxy server	22
รูปที่ 2.20 WAP Gateway ของผู้ให้บริการโทรศัพท์เคลื่อนที่	23
รูปที่ 2.21 WAP Gateway ของ ISP ต่าง ๆ	23
รูปที่ 3.1 การออกแบบทางด้านฮาร์ดแวร์	27
รูปที่ 3.2 การต่ออุปกรณ์ในการควบคุมเข้ากับ Input และ Output ของ PLC	29
รูปที่ 3.3 การต่อเครื่องแปลงอุณหภูมิเป็นสัญญาณไฟฟ้ามาตรฐานเข้ากับการ์ด Analog Input Module	29
รูปที่ 3.4 หน้าโปรแกรมเมื่อเริ่มต้นใช้งาน	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ(ต่อ)

	หน้า
รูปที่ 3.5 หน้าโปรแกรมเมื่อทำงานในรูปแบบ Manual	31
รูปที่ 3.6 หน้าโปรแกรมที่ทำงานในรูปแบบ Automatic	32
รูปที่ 3.7 การติดต่อระหว่างโทรศัพท์เคลื่อนที่กับ WAP Server	33
รูปที่ 3.8 การออกแบบ WAP SITE	34
รูปที่ 3.9 การออกแบบ WAP SITE ต่อ	35
รูปที่ 3.10 การออกแบบ WAP SITE ต่อ	36
รูปที่ 3.11 การออกแบบ WAP SITE ต่อ	37
รูปที่ 3.12 การออกแบบตารางผู้ใช้	39
รูปที่ 3.13 การออกแบบตารางแสดงค่าต่าง ๆ ของระบบ	39
รูปที่ 3.14 การออกแบบตารางการควบคุมการทำงานแบบ Manual	39
รูปที่ 3.15 การออกแบบตารางการควบคุมการทำงานแบบ Manual	40
รูปที่ 4.1 อุปกรณ์ที่จะนำมาควบคุม	41
รูปที่ 4.2 ผลการควบคุมแบบ Manual(บนคอมพิวเตอร์)	42
รูปที่ 4.3 ผลการควบคุมแบบ Manual(อุปกรณ์)	43
รูปที่ 4.4 การตั้งค่าการทำงานแบบอัตโนมัติ	44
รูปที่ 4.5 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติ(บนอุปกรณ์)	45
รูปที่ 4.6 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติเมื่อมีสารเพียงพอ(บนคอมพิวเตอร์)	46
รูปที่ 4.7 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติเมื่อมีสารเพียงพอ (บนอุปกรณ์)	46
รูปที่ 4.8 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติต่อเมื่ออุณหภูมิถึงที่ตั้งไว้ (บนคอมพิวเตอร์)	47
รูปที่ 4.9 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติต่อเมื่ออุณหภูมิถึงที่ตั้งไว้ (บนอุปกรณ์)	47
รูปที่ 4.10 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติเมื่ออุณหภูมิลดจากที่ตั้งไว้ (บนคอมพิวเตอร์)	48
รูปที่ 4.11 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติเมื่ออุณหภูมิลดจากที่ตั้งไว้ (บนอุปกรณ์)	49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญรูปภาพ(ต่อ)

	หน้า
รูปที่ 4.12 การใช้งานในส่วนของการควบคุมผ่านโทรศัพท์เคลื่อนที่	49
รูปที่ 4.13 เรคคอร์ดที่เก็บค่าควบคุมที่ส่งมาจากโทรศัพท์เคลื่อนที่	50
รูปที่ 4.14 เมื่อส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่	50
รูปที่ 4.15 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่	50
รูปที่ 4.16 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่	51
รูปที่ 4.17 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่	51
รูปที่ 4.18 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่	52
รูปที่ 4.19 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่	53
รูปที่ 4.20 การตรวจสอบการทำงานของระบบจากโทรศัพท์เคลื่อนที่	53
รูปที่ 4.21 เร็คคอร์ดเซ็ทของสถานะการทำงานของระบบ(control)	54



บทที่ 1

บทนำ

1.1 แนวคิดและที่มาของโครงการ

ในภาวะปัจจุบัน PLC ได้เข้ามามีบทบาทในวงการอุตสาหกรรมเป็นอย่างมากเนื่องจากสามารถใช้งานควบคุมอุปกรณ์ต่างๆ ในโรงงานอุตสาหกรรมได้อย่างหลากหลายเพราะสามารถเขียนโปรแกรมควบคุมจากผู้ใช้ได้เองและสามารถแก้ไขโปรแกรมที่เขียนขึ้นได้ง่ายทำให้ใช้ได้กับอุปกรณ์ในระบบควบคุมได้ในหลายลักษณะ และเมื่อเรานำเทคโนโลยีการสื่อสารแบบไร้สายมาใช้งานร่วมกับ PLC ทำให้เพิ่มศักยภาพในการควบคุม ได้มากขึ้นเพราะสามารถควบคุมได้บนโทรศัพท์เคลื่อนที่ ทำให้สามารถสั่งการควบคุมได้ในระยะไกล

จากแนวคิดดังกล่าวทำให้เราเลือกที่จะควบคุมกระบวนการทางเคมี ซึ่งเป็นส่วนหนึ่งของกระบวนการผลิตน้ำมันไบโอดีเซล ในการควบคุมอุณหภูมิ และอัตราการไหลในกระบวนการ

1.2 จุดประสงค์

1.2.1 เพื่อเพิ่มศักยภาพในการสั่งงานแก่เครื่อง PLC โดยการสั่งงานผ่านทางคอมพิวเตอร์ และ โทรศัพท์เคลื่อนที่

1.2.2 เพื่อศึกษาและเขียน โปรแกรมสำหรับสร้าง wap page เพื่อใช้บน โทรศัพท์เคลื่อนที่

1.2.3 เพื่อศึกษาการติดต่อสื่อสารและใช้งานร่วมกันของ wap server กับโปรแกรม Visual

Basic

1.2.4 เพื่อศึกษาและสามารถเขียน โปรแกรม Visual Basic ให้ใช้งานร่วมกับเครื่อง PLC ได้

1.2.5 สามารถเขียน โปรแกรมเครื่อง PLC เพื่อควบคุมอุปกรณ์ในกระบวนการทางเคมีได้

1.2.6 เพื่อวิเคราะห์การทำงานของกระบวนการแล้วออกแบบระบบควบคุมที่เหมาะสมได้

1.2.7 เพื่อทดสอบประสิทธิภาพระบบที่สั่งงานผ่าน โทรศัพท์เคลื่อนที่

1.2.8 เพื่อเป็นต้นแบบในการประยุกต์ใช้กับระบบอื่น ๆ ในโรงงานอุตสาหกรรมได้ต่อไป

ในอนาคต

1.3 ขอบเขตของโครงการงาน

1.3.1 สามารถควบคุมอัตราการไหลได้ตามระดับที่ต้องการ

- โดยการควบคุมการทำงานของปั๊ม และ โซลีนอยวาล์วแต่ละตัวให้ทำงานตามที่ออกแบบเพื่อควบคุมอัตราการไหลของไบโอดีเซล โดยการสั่งงานควบคุมได้จาก Computer หรือ จากโทรศัพท์เคลื่อนที่

1.3.2 สามารถควบคุมอุณหภูมิได้ในหลายช่วงอุณหภูมิ

- โดยการควบคุมการทำงานของฮีตเตอร์ เพื่อเพิ่มอุณหภูมิในกระบวนการทางเคมีให้อยู่ในช่วงอุณหภูมิที่ต้องการ

1.3.3 สามารถแสดงอุณหภูมิในกระบวนการทางเคมีได้

- โดยการแสดงอุณหภูมิที่ได้จากการวัดในกระบวนการทางเคมีซึ่งจะให้เป็นช่วงของอุณหภูมิ

1.3.4 สามารถแสดงสถานะการทำงานของอุปกรณ์ที่ใช้ควบคุมกระบวนการทางเคมีได้

1.4 ผลที่คาดว่าจะได้รับ

1.4.1 สามารถออกแบบและเขียนโปรแกรมเพื่อสร้าง WAP Page บนโทรศัพท์เคลื่อนที่ได้

1.4.2 สามารถเขียนโปรแกรมใช้งาน Visual Basic เพื่อเป็นตัวกลางในการสื่อสารระหว่าง WAP Server กับเครื่อง PLC

1.4.3 สามารถนำเครื่อง PLC มาออกแบบโปรแกรมเพื่อควบคุมกระบวนการทางเคมีได้ตามความต้องการ

1.4.5 นำไปประยุกต์ใช้งาน ได้จริงเพื่อเพิ่มศักยภาพในการควบคุมและการผลิตในโรงงานอุตสาหกรรม

1.5 วิธีการดำเนินงาน

1.5.1 ด้านการติดต่อสื่อสารส่งผ่านข้อมูลจากโทรศัพท์เคลื่อนที่มายัง WAP Server

- ศึกษาเทคโนโลยี WAP

- ศึกษาและทดลองเขียน WAP Page ด้วย WML

- ศึกษาและการส่งผ่านข้อมูลมาประมวลผลที่ WAP Server ด้วย Script แบบ PHP

1.5.2 ส่วนของการเชื่อมระหว่าง เครื่องคอมพิวเตอร์กับเครื่อง PLC

- ศึกษาและทดลองการเขียนโปรแกรม Visual Basic

- ศึกษาและทดลองเขียนโปรแกรม Ladder ด้วยโปรแกรม DirectSOFT32 และ โปรแกรม

DirectSOFT32 Data Server เพื่อส่งค่าตัวแปรระหว่าง PLC กับ Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 โปรแกรมเมเบิลลอจิกคอนโทรลเลอร์ Programmable Logic Controller (PLC)

2.1.1 ประวัติความเป็นมา

เมื่อปี พ.ศ. 2511 ในฝ่าย Hiromatic ของบริษัท General Moter ประเทศสหรัฐอเมริกา ได้คิดค้นอุปกรณ์ควบคุมใหม่เพื่อใช้ทดแทนวงจรไฟฟ้าแบบเดิมที่ใช้กันอยู่ในโรงงานอุตสาหกรรมของบริษัทและในปี พ.ศ. 2512 PLC ได้ถูกผลิตมาจำหน่ายในประเทศสหรัฐอเมริกาเป็นครั้งแรก ส่วนในประเทศญี่ปุ่น PLC ได้ถูกพัฒนาภายหลังจากที่บริษัททอมรอน (OMRON) ประเทศญี่ปุ่นประสบความสำเร็จในการผลิตโซลิด-สเตทรีเลย์ (Solid State Relay) ในปี พ.ศ. 2508 หลังจากนั้น 5 ปี PLC ก็ถูกนำมาจำหน่ายในท้องตลาดจนเป็นที่แพร่หลายในเวลาต่อมา

PLC ของแต่ละบริษัทจะมีชื่อเรียกแตกต่างกันในแต่ละประเทศคือ

- PC หรือ Programmable Controller เรียกในประเทศอังกฤษ
- PLC หรือ Programmable Logic Controller เรียกกันในประเทศสหรัฐอเมริกา
- PBS หรือ Programmable Binary System เรียกกันในกลุ่มประเทศสแกนดิเนเวีย

โปรแกรมเมเบิล ลอจิก คอนโทรลเลอร์ เป็นอุปกรณ์ชนิดโซลิด-สเตท (Solid State) ที่ทำงานแบบลอจิก การออกแบบการทำงานของ PLC จะคล้ายกับหลักการของคอมพิวเตอร์ จากหลักการพื้นฐานแล้ว PLC จะประกอบด้วยอุปกรณ์ที่เรียกว่า Solid-State Digital Logic Elements เพื่อให้ทำงานและตัดสินใจแบบลอจิก PLC ใช้สำหรับควบคุมเครื่องจักรและอุปกรณ์ในโรงงานอุตสาหกรรม

การใช้ PLC สำหรับควบคุมเครื่องจักรหรืออุปกรณ์ต่าง ๆ ในโรงงานอุตสาหกรรมจะมีข้อได้เปรียบกว่าการใช้ระบบรีเลย์ ซึ่งจำเป็นต้องเดินสายไฟฟ้า ดังนั้นเมื่อจำเป็นต้องเปลี่ยนกระบวนการผลิตหรือลำดับการทำงานใหม่ ก็ต้องเดินสายไฟฟ้าใหม่ ซึ่งเสียเวลาและค่าใช้จ่ายสูง แต่เมื่อเปลี่ยนมาใช้ PLC แล้ว การเปลี่ยนกระบวนการหรือลำดับงานใหม่ทำได้โดยการเปลี่ยนโปรแกรมใหม่เท่านั้น นอกจากนี้ PLC ยังใช้ระบบโซลิด-สเตท ซึ่งหน้าเชื่อถือกว่าระบบเดิม การกินกระแสไฟฟ้าน้อยกว่า และสะดวกกว่าเมื่อต้องการขยายขั้นตอนการทำงานของเครื่องจักร

2.1.2 โครงสร้างของ PLC

PLC เป็นอุปกรณ์คอมพิวเตอร์สำหรับใช้ในงานอุตสาหกรรม PLC ประกอบด้วย หน่วยประมวลผลกลาง หน่วยความจำ หน่วยรับข้อมูล หน่วยส่งข้อมูล และหน่วยป้อน โปรแกรม PLC

ขนาดเล็กส่วนประกอบทั้งหมดจะรวมเป็นเครื่องเดียวกัน แต่ถ้าเป็นขนาดใหญ่สามารถแยกออกเป็น ส่วนประกอบย่อยๆ ได้

หน่วยความจำของ PLC ประกอบด้วย หน่วยความจำชนิด RAM และ ROM หน่วยความจำ ชนิด RAM ทำหน้าที่เก็บโปรแกรมของผู้ใช้และข้อมูลสำหรับใช้ในการปฏิบัติงานของ PLC ส่วน ROM ทำหน้าที่สำหรับเก็บโปรแกรมที่ใช้สำหรับปฏิบัติงานของ PLC ตามโปรแกรมของผู้ใช้

2.1.3 ส่วนประกอบของ PLC

PLC แบ่งออกได้ 3 ส่วนด้วยกันคือ

1. ส่วนที่เป็นหน่วยประมวลผลกลาง (Control Processing Unit : CPU) เป็นส่วนมันสมองของระบบ ภายในประกอบด้วยวงจร Logic Gate ชนิดต่างๆ หลายชนิด และมี Microprocessor-based ใช้สำหรับแทนอุปกรณ์จำพวกรีเลย์ (Relay) เคาน์เตอร์ (Counter) ไทเมอร์ (Timer) และ ซีควเอนเซอร์ (Sequencers) เพื่อให้ผู้ใช้ออกแบบใช้วงจรรีเลย์ แลคเตอร์ ลอจิก (Relay Ladder Logic) เข้าถึงได้

CPU จะยอมรับ (Read) อินพุต เดต้า (Input Data) จากอุปกรณ์ให้สัญญาณ (Sensing Device) ต่างๆ จากนั้นจะปฏิบัติการและเก็บข้อมูลโดยใช้โปรแกรมจากหน่วยความจำ และส่งข้อมูลที่เหมาะสมถูกต้องไปยังอุปกรณ์ควบคุม (Control Device) แหล่งของกระแสไฟฟ้าตรง (DC Current) สำหรับใช้สร้างโวลต์เตต้า (Low Level Voltage) ซึ่งใช้โดยโปรเซสเซอร์ (Processor) และ (I/O Modules) และแหล่งจ่ายไฟฟ้านี้จะเก็บไว้ที่ CPU หรือแยกออกไปติดตั้งที่จุดอื่นก็ได้ขึ้นอยู่กับผู้ผลิตแต่ละราย

2. ส่วนของอินพุตและเอาต์พุต (I/O Unit) จะต่อร่วมกับชุดควบคุมเพื่อรับสถานะและสัญญาณต่าง ๆ เช่น หน่วยอินพุตรับสัญญาณหรือสถานะแล้วส่งไปยัง CPU เพื่อประมวลผล เมื่อ CPU ประมวลผลแล้วจะส่งให้ส่วนของเอาต์พุต เพื่อให้อุปกรณ์ทำงานตามที่เขียนโปรแกรมไว้

สัญญาณอินพุตจากภายนอกที่เป็นสวิตช์และตัวตรวจจับชนิดต่าง ๆ จะถูกแปลงสัญญาณที่เหมาะสมและถูกต้อง ไม่ว่าจะ เป็น DC หรือ AC เพื่อส่งให้ CPU ดังนั้นสัญญาณเหล่านี้จึงต้องถูกต้องไม่เช่นนั้น CPU จะเกิดความเสียหายได้

3. เครื่องป้อนโปรแกรม (Programming Device) หน้าที่ของเครื่องป้อนโปรแกรมของผู้ใช้ลงในหน่วยความจำของ PLC นอกจากนั้นแล้วยังทำหน้าที่ติดต่อระหว่างผู้ใช้กับ PLC เพื่อให้ผู้ใช้สามารถตรวจการปฏิบัติงานของ PLC และผลการควบคุมเครื่องจักรและกระบวนการตามโปรแกรมควบคุมที่ผู้ใช้เขียนขึ้นได้อีกด้วย เครื่องป้อนโปรแกรมแต่ละยี่ห้อจะไม่เหมือนกันแต่มีจุดประสงค์อันเดียวกัน

นอกจากจะใช้เครื่องป้อนโปรแกรมแล้ว PLC สามารถใช้ซอฟต์แวร์คอมพิวเตอร์ได้หลายๆ อย่าง เช่น ใช้ซอฟต์แวร์ทำการป้อนโปรแกรม แก้ไขโปรแกรม ดูแลการทำงานของโปรแกรมเป็นต้น ซอฟต์แวร์แต่ละบริษัทจะมีวิธีการไม่เหมือนกันแต่มีจุดประสงค์เหมือนกัน

2.1.4 ความสามารถของ PLC

PLC สามารถควบคุมงานได้ 3 ลักษณะ คือ

1. งานที่ทำตามลำดับก่อนหลัง (Sequence Control) ตัวอย่าง เช่น

- การทำงานของระบบรีเลย์
- การทำงานของไทมเมอร์ คอนโทรลเลอร์
- การทำงานของ P.C.B. C
- การทำงานในระบบกึ่งอัตโนมัติ ระบบอัตโนมัติ หรืองานที่เป็นกระบวนการทำงานของ

เครื่องจักรกลต่าง

2. งานควบคุมสมัยใหม่ (Sophisticated Control) ตัวอย่างเช่น

- การทำงานทางคณิตศาสตร์ เช่น บวก ลบ คูณ หาร
- การควบคุมแบบอนาล็อก (Analog Control) เช่น การควบคุมอุณหภูมิ (Temperature)
- การควบคุม P.I.D. (Proportional — Integral — Derivation)
- การควบคุมเซอร์โวมอเตอร์ (Servo — Motor Control)
- การควบคุม Stepper — Motor
- Information Handling

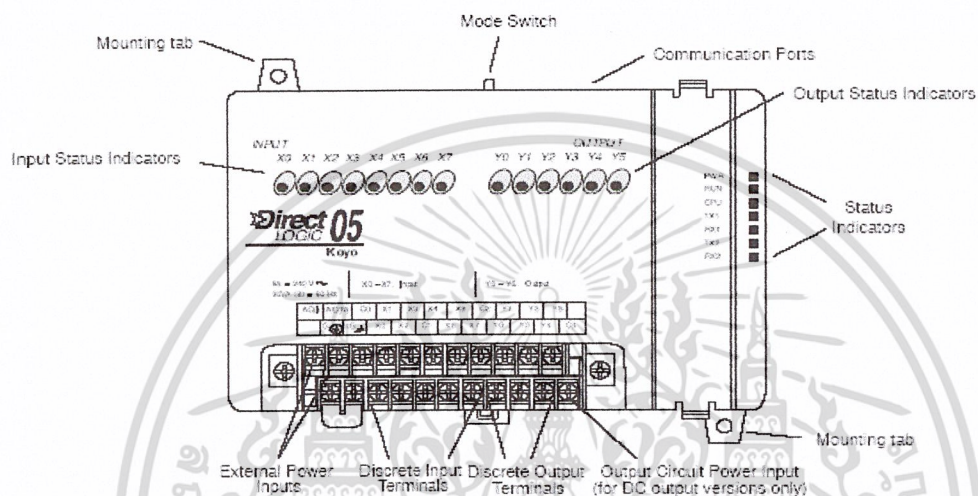
3. งานควบคุมเกี่ยวกับงานอำนวยการ (Supervisory Control) ตัวอย่างเช่น

- งานสัญญาณเตือน (Alarm) และ Process Monitoring
- Fault Diagnostic and Monitoring
- งานต่อร่วมกับคอมพิวเตอร์ (RS-232C/RS422)
- Printer / ASCII Interfacing
- งานควบคุมอัตโนมัติในโรงงานอุตสาหกรรม (Factory Automation Networking)
- LAN (Local Area Network)
- WAN (Wide Area Network)
- FA, FMS. , CIM เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

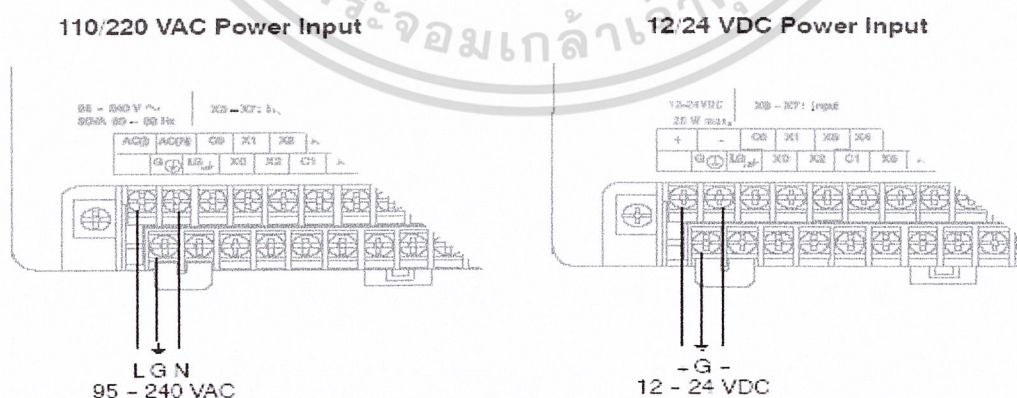
2.1.5 PROGRAMMABLE LOGIC CONTROLLER KOYO DL05

ส่วนประกอบด้านหน้าของ PLC ประกอบด้วย ส่วนขอการติดต่อ, ส่วนแสดงผล, ไฟแสดงสถานะ, และสัญญาณภัยคำอธิบาย แสดงอยู่บนด้านหน้าของ PLC ขนาดเล็กรุ่น DL05 พอร์ตการติดต่อจะอยู่ด้านบนของ PLC ดังรูปข้างล่าง



รูปที่ 2.1 ส่วนประกอบด้านหน้าของ PLC

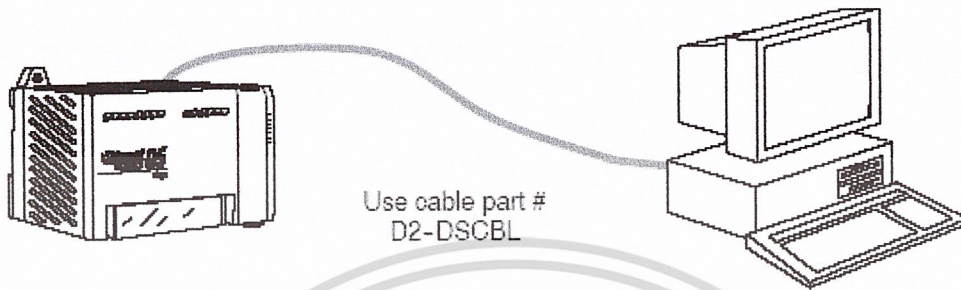
การต่อกับแหล่งจ่ายไฟฟ้าภายนอกสำหรับ PLC รุ่น KOYO DL05 มีการต่อแหล่งจ่ายไฟฟ้า ดังรูปข้างล่าง



รูปที่ 2.2 การต่อแหล่งจ่ายไฟฟ้าภายนอก

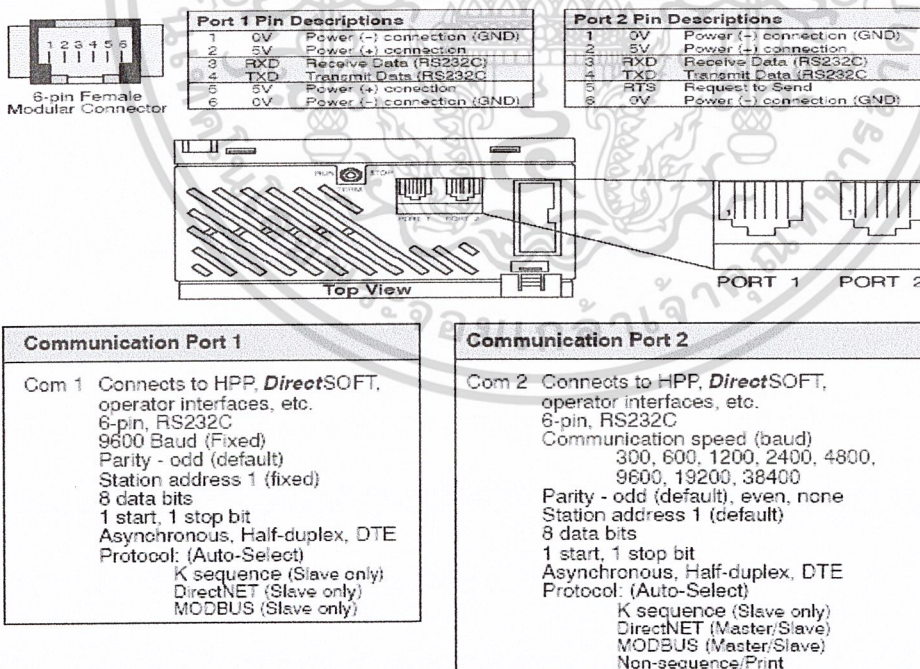
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การติดต่อกับเครื่องคอมพิวเตอร์ส่วนบุคคล PLC ขนาดเล็ก รุ่น DL05 สามารถโปรแกรมได้จาก โปรแกรม Direct SOFT บนเครื่องคอมพิวเตอร์ โดยใช้สายเคเบิลต่อคังรูปข้างล่าง



รูปที่ 2.3 การติดต่อระหว่าง PC กับ PLC

การติดต่อกับพอร์ตภายนอกเราจำเป็นต้องใช้สายที่มีความยาวเพียงพอที่จะติดต่อกับคอมพิวเตอร์ส่วนบุคคลกับ PLC ได้อย่างง่ายดายและรวดเร็ว โดยใช้สายสัญญาณมาตรฐาน RS-232 C โดยใช้หัวปลั๊กโทรศัพท์ RJ-12 ในการทำหัวแจ็ค คู่กับพอร์ตอนุกรม ดังรูป

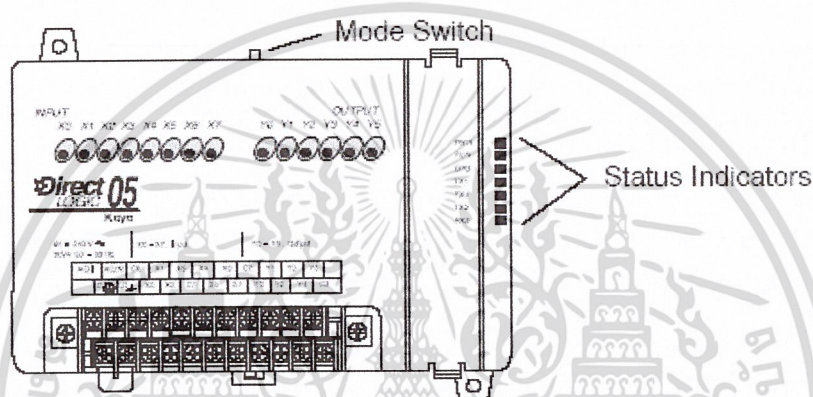


รูปที่ 2.4 การติดต่อกับพอร์ตภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PLC ขนาดเล็ก จะมีพอร์ตติดต่อ RS-232C 2 พอร์ต พอร์ต 1 ปกติใช้ติดต่อกับ D2-HPP , Direct NET ติดต่อการทำงาน, MOD BUS หรือ Direct NET ตัวรอง baud rate กำหนดอยู่ที่ 9600 สำหรับพอร์ต 2 สามารถทำงานเป็นได้ทั้งตัวหลักและตัวรอง พอร์ต 2 มีช่วงความเร็วจาก 300 baud ถึง 34.4 kbaud

การแสดงผลของ PLC รุ่น KOYO DL05 การแสดงผลของไฟแสดงผลของ PLC จะบอกให้ทราบสถานะ การทำงานของ PLC ในขณะนั้น ซึ่งจะมีไฟแสดงสถานะดังรูปข้างล่าง



Indicator	Status	Meaning
PWR	ON	Power good
	OFF	Power failure
RUN	ON	CPU is in Run Mode
	OFF	CPU is in Stop or program Mode
CPU	ON	CPU self diagnostics error
	OFF	CPU self diagnostics good
TX1	ON	Data is being transmitted by the CPU - Port 1
	OFF	No data is being transmitted by the CPU - Port 1
RX1	ON	Data is being received by the CPU - Port 1
	OFF	No data is being received by the CPU - Port 1
TX2	ON	Data is being transmitted by the CPU - Port 2
	OFF	No data is being transmitted by the CPU - Port 2
RX2	ON	Data is being received by the CPU - Port 2
	OFF	No data is being received by the CPU - Port 2

รูปที่ 2.5 การแสดงผลของ PLC รุ่น KOYO DL05

ซึ่งในการทำงานปกติของ PLC จะมีไฟแสดงสถานะของ PLC คือ RUN และ PWR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

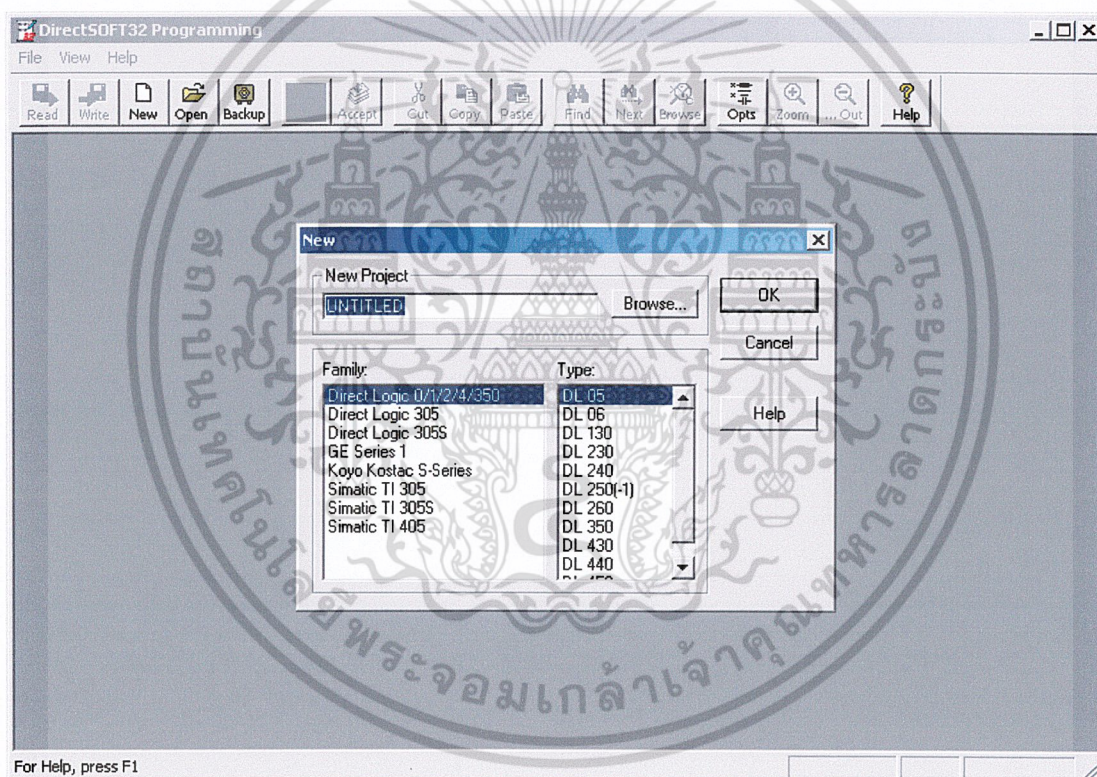
2.1.6 การรันเขียนโปรแกรมลงใน PLC

ในการเขียนโปรแกรมลงใน PLC ขนาดเล็ก รุ่น DL05 นั้นจะใช้โปรแกรม DirectSOFT32 ในการเขียน Ladder ซึ่งเป็นโปรแกรมการทำงานของ PLC แล้วทำการ memory ลงไปใน CPU ของ PLC

การใช้งานโปรแกรม Direct SOFT32

ขั้นตอนแรก

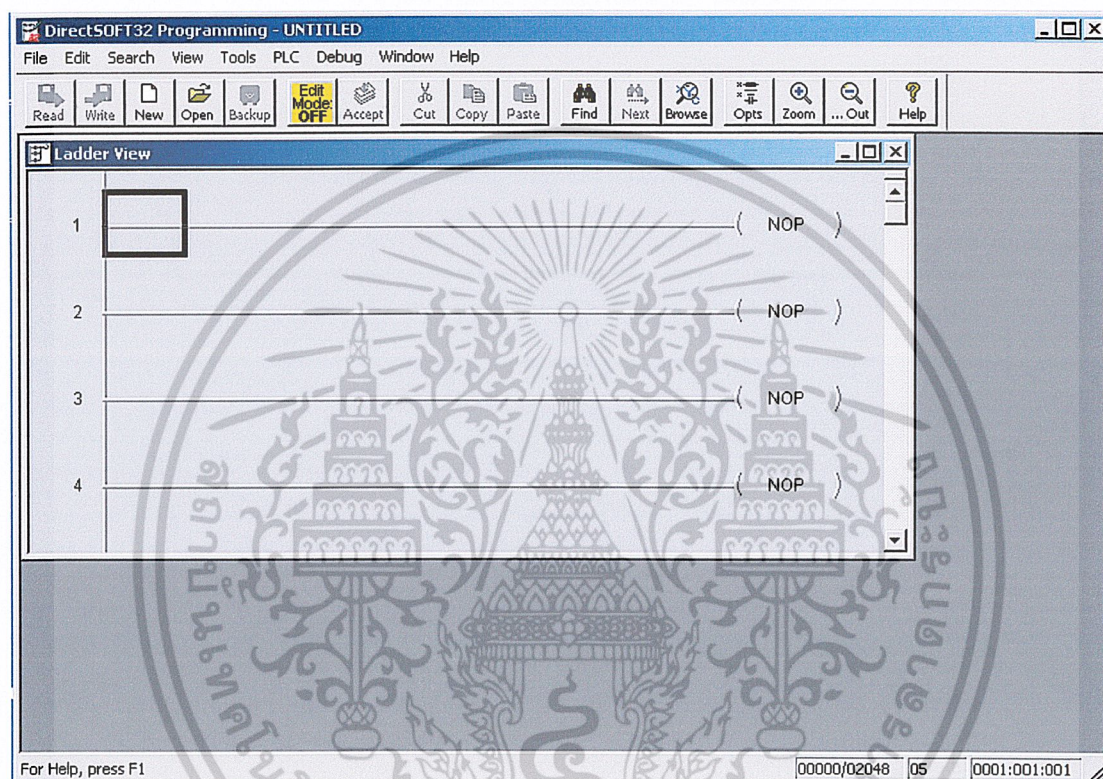
เมื่อทำการคลิก Direct SOFT32 program ก็จะมีปรากฏหน้าต่างดังรูปข้างล่าง



รูปที่ 2.6 หน้าต่างแรกของโปรแกรม Direct SOFT32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งหน้าต่างนี้จะเป็นการเปิด New Project ขึ้นมาเพื่อทำการเขียนโปรแกรม โดยการคลิกที่ช่อง name เพื่อตั้งชื่อ New Project ของคุณ หลังจากนั้น เลือกชนิดของ PLC และรุ่นของ PLC หรือชนิดของ CPU แล้วคลิก OK ก็จะปรากฏหน้าต่างดังรูปข้างล่าง

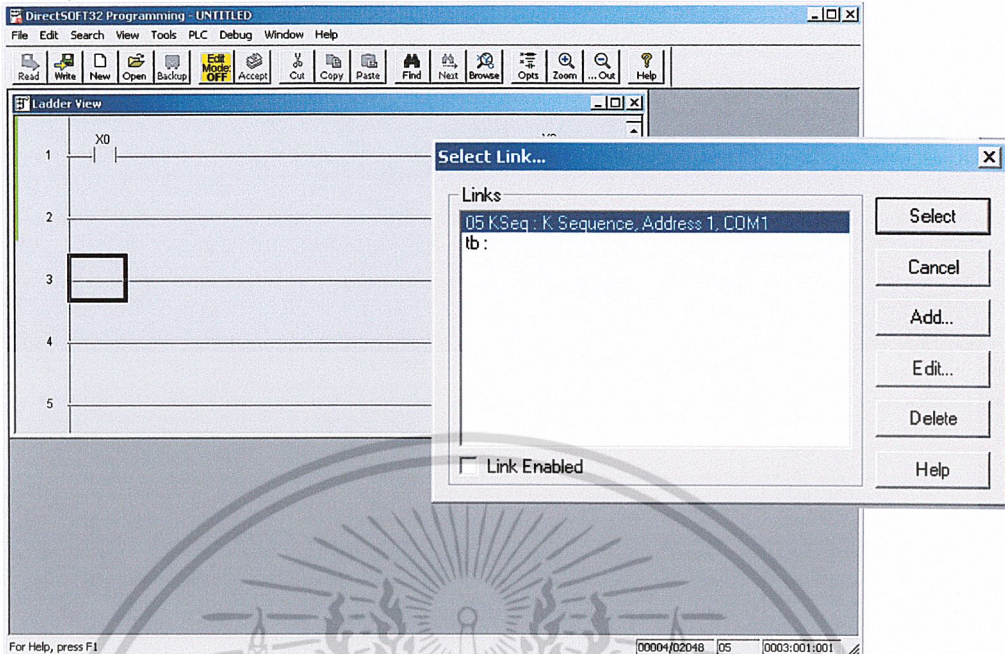


รูปที่ 2.7 หน้าต่าง New Project

ขั้นตอนที่สอง

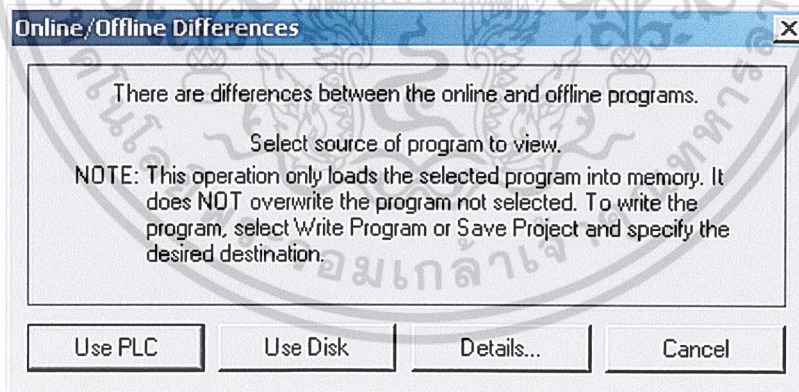
ทำการต่อ PLC กับ computer และทำการ Link กับโปรแกรม DirectSOFT32 โดยการไปเลือกที่ PLC บน menu bar แล้วเลือก connect ก็จะปรากฏ ดังรูปข้างล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 หน้าต่างการ Link Computer กับ PLC

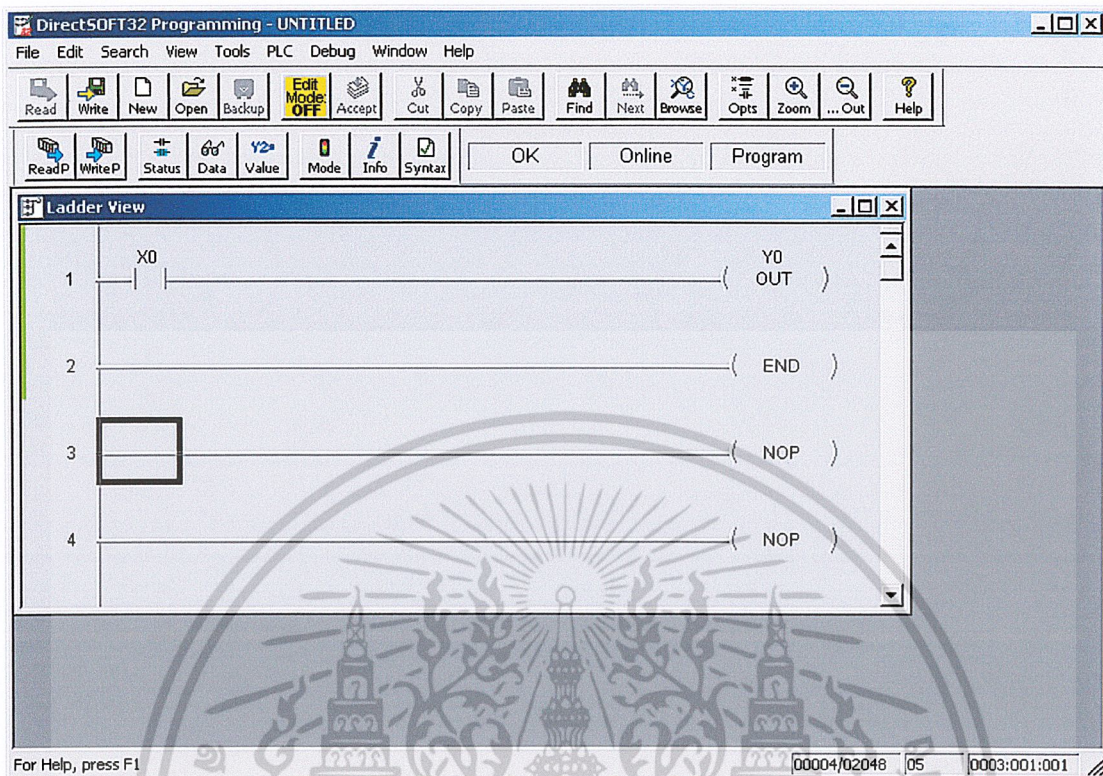
หลังจากนั้นคลิกเลือก Name ในช่อง Link แล้วคลิก select ก็จะปรากฏดังรูปที่



รูปที่ 2.9 หน้าต่างโปรแกรมเลือกอ่านค่าหรือบันทึกค่าลง PLC

แล้วคลิกเลือก User PLC โปรแกรมก็จะทำการอ่านโปรแกรมที่อยู่ใน CPU ของ PLC เพื่อที่จะทำการ RUN โปรแกรม แต่ถ้าเลือก User Disk โปรแกรมก็จะทำการเตรียมพร้อมที่จะบันทึกโปรแกรมใหม่ลง CPU ของ PLC ดังแสดงให้เห็น ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 หน้าต่างของโปรแกรมที่พร้อมจะทำการ Run โปรแกรม

ขั้นตอนที่สาม

การเขียนโปรแกรมทำได้โดยการคลิกที่ Edit Mode Off (ปุ่มจะมีสีเขียว) เพื่อให้บล็อกลักษณะ การเขียน Ladder ปรากฏขึ้นซึ่งใช้ในการเขียน Ladder ที่สะดวกกว่า การใช้ที่ menu bar

ขั้นตอนที่สี่

การเขียนโปรแกรมลงบน PLC เมื่อเครื่อง PC และ PLC ทำการติดต่อกันได้แล้ว ก็สามารถที่จะเขียนโปรแกรมลงบน PLC ได้ โดยผ่านโปรแกรม DirectSOFT32 ซึ่งทำได้โดยการคลิกที่ไอคอน ที่สองข้างล่างทางซ้าย ซึ่งเป็นไอคอนการ write โปรแกรมลงบน PLC

และหลังจากนั้นทำการ RUN โปรแกรม โดยคลิกที่ไอคอน Mode และจะปรากฏหน้าต่าง ดังรูปข้างล่าง แล้วคลิกที่ RUN PLC ก็จะอยู่ใน Mode Run

เมื่อโปรแกรมอยู่ใน Mode Run โปรแกรมก็จะปรากฏสถานะให้เห็น

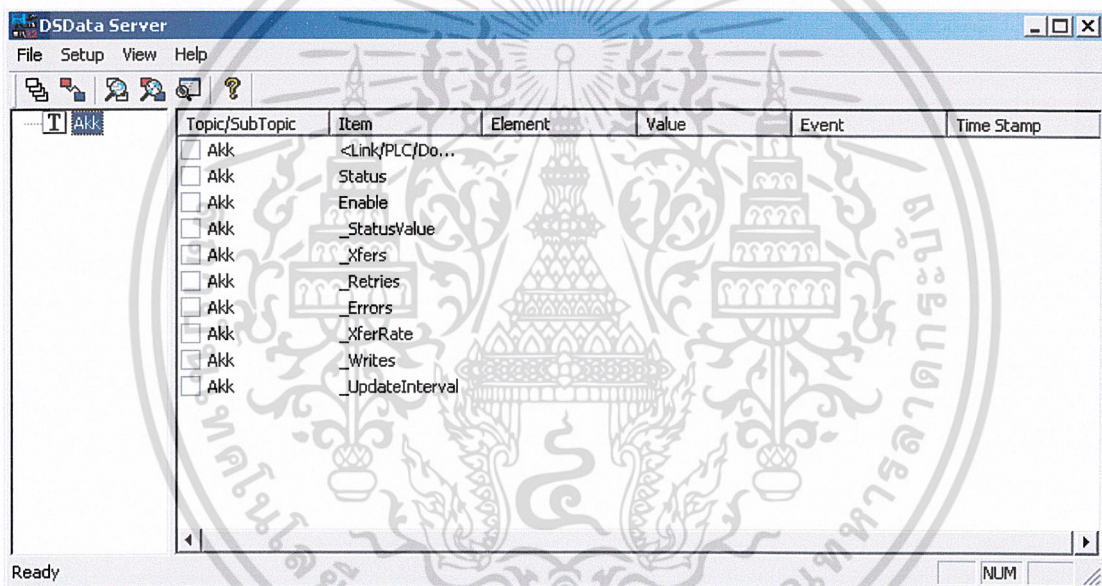
2.1.7โปรแกรม DS Data server

โปรแกรม Dsdata Server เป็นโปรแกรมที่ช่วยในการที่จะให้ Software window แลกเปลี่ยนข้อมูลกับเครื่อง PLC ซึ่งเปรียบเสมือนว่าโปรแกรม DsData Server จะเป็นล่ามที่ช่วยในการแปลข้อมูลจาก Software window เพื่อที่จะให้เครื่อง PLC เข้าใจ และทำงานได้ตามต้องการ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือทำการแปลข้อมูลที่ได้จากเครื่อง PLC เพื่อให้ Software window เข้าใจ และสามารถนำไปแสดงผล การใช้โปรแกรม DsData Server จะใช้ได้เฉพาะเครื่อง PLC ของยี่ห้อ KOYO เท่านั้น

การติดต่อสื่อสารกันระหว่างโปรแกรม จะใช้การติดต่อได้ 2 แบบ คือ DDE หรือ OPC ของแต่ละโปรแกรม ซึ่ง DDE และ OPC จะเป็น Protocol ที่ใช้ในการติดต่อสื่อสาร

การตั้งค่าการเชื่อมกับเครื่อง PLC การที่เราจะใช้ Dsdasta Server ทำงานได้ต้องทำการตั้งค่าการเชื่อมต่อเสียก่อน โดยการคลิกที่ไอคอน Setup links จะปรากฏหน้าจอจัดรูป ถ้าไม่มีการติดต่อสื่อสารมาก่อนจะปรากฏ เพียงแค่หน้าจอว่าง ถ้ามีการตั้งค่าไว้แล้วให้ทำการกด Select ได้ทันที



รูปที่ 2.11 หน้าต่าง โปรแกรม DsData Server สำหรับการตั้งค่าการเชื่อมต่อ

แต่ถ้ายังไม่ได้มีการตั้งค่าเอาไว้ จะต้องสร้างการเชื่อมต่อใหม่ โดยการคลิก Add เพื่อสร้างการเชื่อมต่อใหม่แล้วก็ให้ทำตามขั้นตอนไปเรื่อย ๆ โดยสามารถที่จะเลือกว่าจะติดต่อกับเครื่อง PLC รุ่นไหน และทำการเลือก Protocol ว่าจะใช้ Protocol อะไรในการเชื่อมต่อ

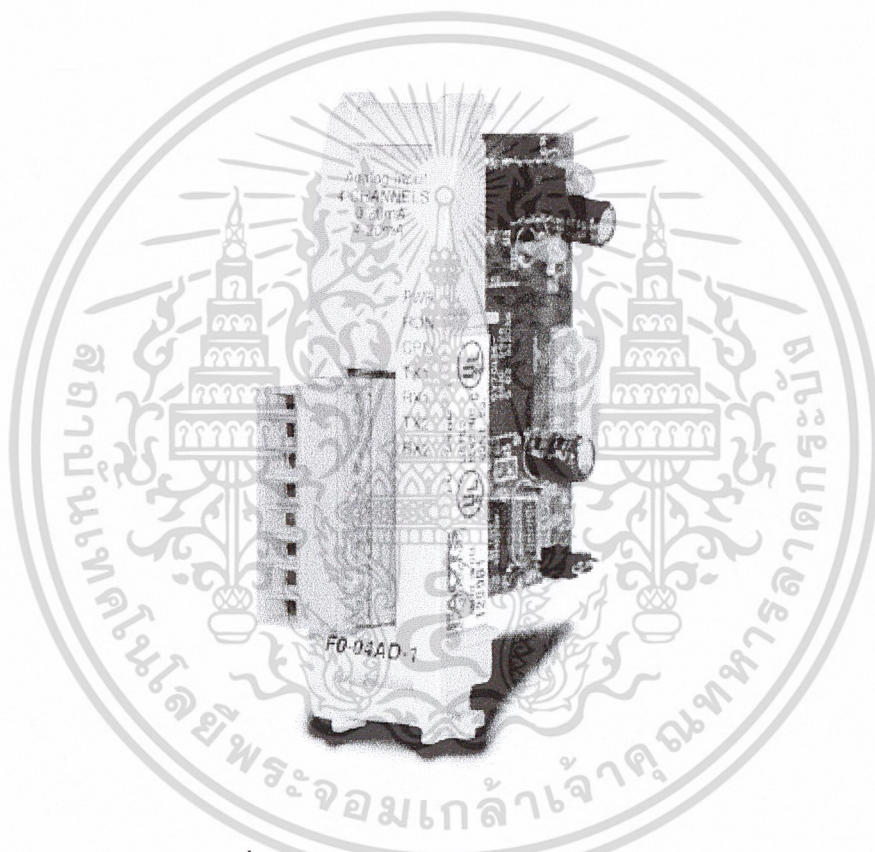
การสร้างหัวข้อในการติดต่อกับเครื่อง PLC หัวข้อ หรือ Topic ที่เราทำการสร้างไว้จะถูกนำไปใน โปรแกรม Visual Basicต่อไป ซึ่งการสร้าง Topic จะทำได้โดยการคลิกที่ไอคอน Setup Topics ก็จะขึ้นหน้าต่างให้เราใส่ชื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้เราใส่ชื่อของ Topic นั้น ๆ เช่น Demo Topic เป็นชื่อของ Topic ถ้าเราได้สร้างการเชื่อมต่อไว้แล้วก็สามารถเลือกได้ทันทีว่าจะใช้การติดต่อแบบไหน เมื่อเราทำการสร้าง Topic เสร็จแล้ว ให้เราทำการคลิกขวา ที่ Topic นั้นแล้ว เลือก Edit Item

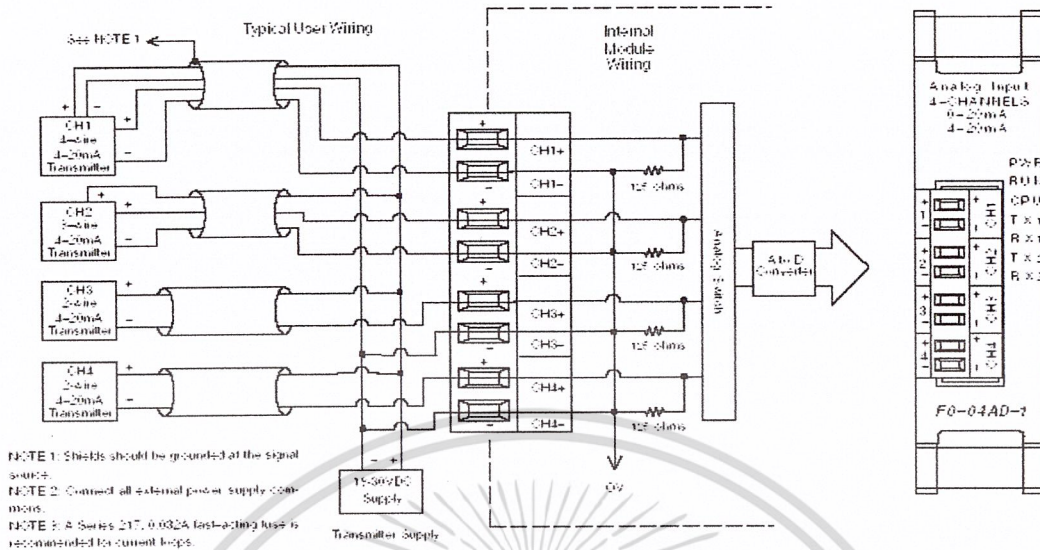
จะขึ้นมาให้เราเขียนชื่อ Item และทำการเลือกว่าจะให้แทนส่วนใดของเครื่อง PLC เช่น C, T, V เป็นต้น

2.1.8 Analog Input Module 4 Channel



รูปที่ 2.12 Analog Input Module 4 Channel

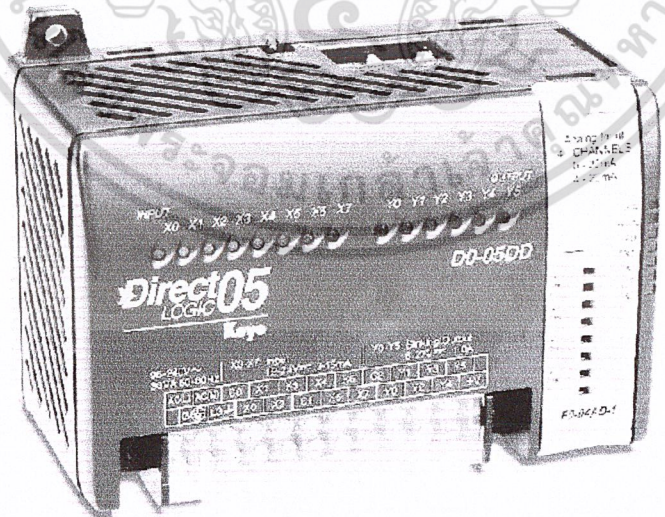
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 โครงสร้างภายในของ Analog Input Module 4 Channel

การติดตั้งกับเครื่อง PLC DL-05 ของ KOYO

ในตัวเครื่อง PLC DL-05 ของ KOYO จะมีช่องสำหรับที่จะใช้ต่อกับ Module พิเศษอยู่แล้ว 1 ช่อง การที่จะติดตั้งจึงสามารถที่จะทำได้ง่าย เพียงแค่การเสียบลงในช่องที่มีอยู่แล้วในตัวเครื่อง PLC ก็จะสามารถทำงาน ได้ทันที



รูปที่ 2.14 ตัวเครื่อง PLC DL-05 เมื่อติดตั้ง Analog Input Module 4 Channel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 อาร์ทีดี(RTD)

อาร์ทีดี(Resistance Temperature Detector) จัดเป็นทรานสดิวเซอร์อีกแบบหนึ่งที่ใช้สำหรับการวัดอุณหภูมิโดยใช้วัสดุที่มีค่าความต้านทานเปลี่ยนแปลงไปตามการเปลี่ยนแปลงของอุณหภูมิ วัสดุเหล่านี้ได้แก่ แพลทินัม(Platinum) ทองแดง (Copper) และนิเกิล(Nickel) แต่โดยทั่วไปจะนิยมใช้อาร์ทีดีที่ทำมาจากแพลทินัมเนื่องจากแพลทินัม มีความเที่ยงตรงและมีความเป็นเชิงเส้นมากที่สุด ซึ่งจะเรียกว่า PT100 (Platinum Temperature) โดยที่ตัวเลข 100 หมายถึง ที่อุณหภูมิ 0 องศา ค่าความต้านทานภายในของอาร์ทีดีจะมีค่าเท่ากับ100โอห์ม ความสัมพันธ์ระหว่างค่าความต้านทานของเส้นลวดตัวนำกับอุณหภูมิสามารถจะคำนวณได้จากสมการ ต่อไปนี้

$$R_t = R_{ref} (1 + \alpha \Delta t)$$

เมื่อ

R_t = ค่าความต้านทานของตัวนำที่อุณหภูมิ t °C ใดๆ

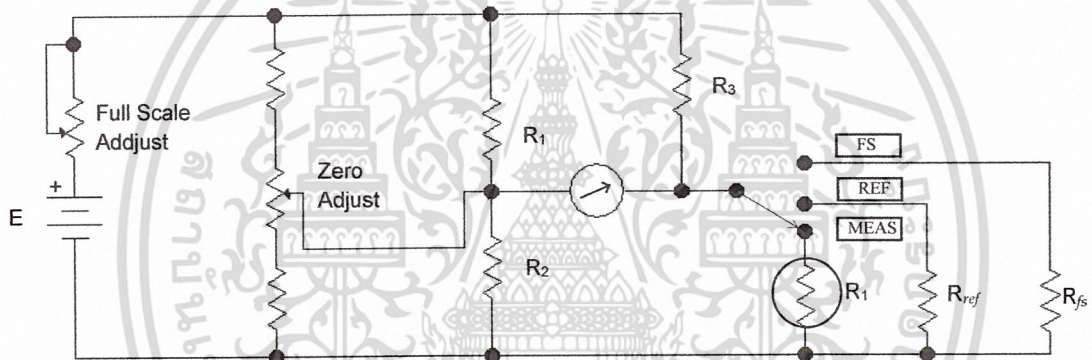
R_{ref} = ค่าความต้านทานของตัวนำที่อุณหภูมิอ้างอิง (โดยทั่วไปคือ 0 °C)

α = สัมประสิทธิ์ของค่าความต้านทานต่ออุณหภูมิของโลหะตัวนำ

Δt = ค่าความแตกต่างระหว่างอุณหภูมิ t °C ใดๆ กับอุณหภูมิอ้างอิง

โลหะตัวนำโดยทั่วไปนั้น สัมประสิทธิ์ของค่าความต้านทานต่ออุณหภูมิเป็นบวก กล่าวคือค่าความต้านทานของโลหะตัวนำจะเพิ่มขึ้นเมื่ออุณหภูมิเพิ่มขึ้น แต่วัสดุบางอย่างเช่น คาร์บอน, เจอร์มันเนียม(Germanium) นั้นจะมีสัมประสิทธิ์ของค่าความต้านทานต่ออุณหภูมิเป็นลบ นั่นคือค่าความต้านทานของวัสดุเหล่านี้จะลดลง ถ้าอุณหภูมิเพิ่มขึ้น โดยทั่วไปแล้วควรจะใช้วัสดุที่มีค่า α สูงเพราะหมายความว่า ค่าความต้านทานของโลหะตัวนำจะเปลี่ยนแปลงไปอย่างเห็นได้ชัดถึงแม้ว่า อุณหภูมิจะเปลี่ยนไปเพียงเล็กน้อยก็ตาม ค่าความต้านทานที่เปลี่ยนไปนี้สามารถจะวัดได้โดยใช้ Whetstone Bridge และปรับเทียบให้แสดงค่าเป็นค่าของอุณหภูมิได้ อาร์ทีดีนั้น โดยทั่วไปจะเป็นแบบโพรบ (Probe)เพื่อที่จะใช้จุ่มหรือใส่เข้าไปในตัวกลางใด ๆ ที่ต้องการวัดหรือควบคุมอุณหภูมิได้ อุปกรณ์ตรวจวัด(Sensing Element) ของเทอร์โมมิเตอร์แบบโพรบนี้ จะทำได้โดยการฉาบหรือเคลือบหลอดเล็ก ๆ ที่ทำด้วยเงินหรือแพลทินัมด้วยเซรามิกชั้นหนึ่งก่อน จากนั้นนำขดลวดความต้านทานมาพันรอบ ๆ หลอดนี้และฉาบหรือเคลือบด้วยเซรามิกอีกครั้งหนึ่ง เมื่อเรียบร้อยแล้วจึงนำไปเผาไฟที่อุณหภูมิสูงเพื่อให้แน่ใจว่า เส้นลวดอ่อนตัวแล้วจึงนำไปติดไว้ที่ปลายของโพรบเพื่อเป็นตัวตรวจวัดอุณหภูมิต่อไป

อาร์ทีดีที่ใช้ในงานอุตสาหกรรมนั้นมักจะติดตั้งไว้ใน Protecting Tube (Probe) หรือ Protecting Well เสมอเพื่อป้องกันความเสียหายที่อาจเกิดขึ้นต่ออุปกรณ์ตรวจวัด และยังใช้ป้องกันความสกปรก ซึ่งอาจจะมีผลต่อการวัดหรือการควบคุมอุณหภูมิได้ โดยทั่วไปแล้วจะใช้ Protecting Tube ในระดับความดันบรรยากาศ ส่วน Protecting Well นั้นจะใช้ในกรณีของการวัดหรือการควบคุมอุณหภูมิของของเหลวหรือก๊าซที่ความดันสูง ๆ เช่น ในท่อ, Steam Power Plant, Pumping Station และอื่นๆ Protecting Well นี้มักจะใช้เมื่อความดันในของเหลวหรือก๊าซที่ต้องการวัดอุณหภูมินั้นมีค่าสูงกว่า 3 เท่าของความดันบรรยากาศ ซึ่ง Protecting Well นี้ทำจาก Stainless Steel หรือ Carbon Steel และมีอุปกรณ์ตรวจวัดบรรจุอยู่ภายใน



รูปที่ 2.15 วงจรบริดจ์ที่ใช้ร่วมกับอาร์ทีดีในการวัดค่าความต้านทาน

แต่เนื่องจากว่าค่าความต้านทานของอุปกรณ์ตรวจวัดจะมีความสัมพันธ์อย่างเป็นเชิงเส้นกับอุณหภูมิ ดังนั้นจึงสามารถปรับเทียบค่าที่อ่านได้เป็นค่าของอุณหภูมิ จากรูป R_t จะเป็นค่าความต้านทานของอาร์ทีดีที่อุณหภูมิ t °C ใดๆ ที่ต้องการวัด R_{ref} เป็นตัวต้านทานที่มีค่าความต้านทานคงที่ซึ่งมีค่าเท่ากับค่าความต้านทานของ resistance thermometer ที่อุณหภูมิอ้างอิง (0 °C) ส่วน R_{fs} นั้นจะเป็นตัวต้านทานที่มีค่าความต้านทานคงที่อีกตัวหนึ่ง ค่าความต้านทาน R_{fs} นี้จะมีค่าเท่ากับค่าความต้านทานของอาร์ทีดีที่ทำให้มิเตอร์แสดงค่ากระแสชี้ค่าเต็มสเกล ซึ่งก็หมายถึงค่าอุณหภูมิสูงสุดเช่นกัน เพราะค่าความต้านทานของอาร์ทีดี นั้นสัมพันธ์อย่างเป็นเชิงเส้นกับอุณหภูมิ ก่อนที่จะทำการวัดอุณหภูมินั้นจะต้องทำการปรับเทียบเครื่องมือวัดนี้ก่อน กล่าวคือขั้นแรกให้สวิตช์ไปอยู่ที่ตำแหน่ง “REF” และปรับค่าความต้านทานของตัวต้านทาน “Zero Adjust” จนมิเตอร์ชี้ที่ศูนย์ ซึ่งหมายถึงอุณหภูมิศูนย์องศาแน่นอนเอง ต่อไปสับสวิตช์ให้ไปอยู่ ณ ตำแหน่ง “FS” และปรับค่าความ

ด้านทานของตัวต้านทาน “Full Scale Adjust” จนมิเตอร์แสดงค่าเต็มสเกลพอดี ที่ตำแหน่งเต็มสเกลนี้จะเป็นค่าสูงสุดของอุณหภูมิของวงจรนี้สามารถวัดได้ จากนั้นก็สามารถนำเครื่องมือวัดนี้ไปวัดอุณหภูมิได้

เมื่อจะทำการวัดค่าของอุณหภูมิใด ๆ สวิตช์จะถูกกลับไปอยู่ที่ตำแหน่ง “MEAS” ซึ่งต่อกับอาร์ทีดีซีเอ็มของมิเตอร์ก็จะชี้แสดงค่าใดค่าหนึ่งบนสเกลซึ่งถูกปรับเทียบให้เป็นค่าของอุณหภูมิแล้วตามวิธีการปรับเทียบที่กล่าวมาข้างต้น อย่างไรก็ตาม ข้อเสียของเครื่องมือวัดที่ใช้ Whetstone Bridge คือจะมีค่าความต้านทานของหน้าสัมผัสของวงจรบริดจ์เข้ามารวมกับค่าความต้านทานของหน้าสัมผัสของวงจรบริดจ์เข้ามารวมกับค่าความต้านทานของอาร์ทีดี หรือค่าความต้านทานของตัวต้านทานและเส้นลวดในวงจรอาจจะเปลี่ยนแปลงไปตามความร้อนที่เกิดขึ้นเนื่องจากกระแสที่ไหลในวงจร ทำให้การวัดอาจจะผิดพลาดไปได้ แต่อย่างไรก็ตาม ข้อเสียนี้สามารถแก้ไขได้โดยการใช้วงจรบริดจ์แบบ Kelvin Double Bridge แทน

2.3 WAP (wireless Application Protocol)

WAP เป็นเทคโนโลยีที่สามารถติดต่อกับเครือข่ายอินเทอร์เน็ตได้ด้วยอุปกรณ์ไร้สายเช่น โทรศัพท์เคลื่อนที่ เช่นเดียวกับเทคโนโลยี Bluetooth และ I-Mode โดย WAP ได้กำหนดโปรโตคอลขึ้นมาเพื่อใช้ในการส่งผ่านข้อมูลสำหรับการทำงานในแต่ละเลเยอร์เช่นเดียวกับระบบอินเทอร์เน็ตซึ่งมีโปรโตคอลในแต่ละเลเยอร์ต่าง ๆ เช่น HTTP, TCP, IP เป็นต้น ซึ่งโปรโตคอลเหล่านี้ใช้สำหรับเป็นข้อตกลงในการส่งผ่านข้อมูลไปตามสายนำสัญญาณ แต่โปรโตคอลของ WAP จะเป็นข้อตกลงใช้ในการส่งข้อมูลโดยระบบไร้สายซึ่งอาศัยคลื่นวิทยุเป็นพาหะ หรืออาจจะพูดได้ว่า WAP เป็นข้อกำหนดในการรับส่งข้อมูลผ่านเครือข่ายไร้สายอีชนิดหนึ่งซึ่งถูกออกแบบให้เหมาะสมกับการสื่อสารในระบบไร้สาย โดยเฉพาะระบบโทรศัพท์เคลื่อนที่

ผู้ที่คิดค้นหรือสร้างข้อกำหนดต่าง ๆ ของเทคโนโลยี WAP ขึ้นมาคือกลุ่ม WAP Forum ซึ่งหลัก ๆ ประกอบไปด้วยบริษัทที่ดำเนินธุรกิจที่เกี่ยวกับโทรศัพท์เคลื่อนที่ (Phone.com, โนเกีย, อีริกสัน, โมโตโลล่า ฯลฯ) โดยในครั้งแรกบริษัท Phone.com ได้คิดค้นภาษามาร์กอัปขึ้นมาตัวหนึ่งชื่อว่า HDML (Hand-held Device Markup Language) เพื่อใช้สำหรับแสดงผลบนอุปกรณ์ไร้สายต่าง ๆ รวมทั้งโทรศัพท์เคลื่อนที่ด้วย ซึ่งในเวลาต่อมาบริษัท Phone.com ได้ร่วมกับบริษัทต่าง ๆ ในกลุ่ม WAP Forum เพื่อคิดค้นแนวทางใหม่ในการส่งผ่านข้อมูลโดยระบบเครือข่ายไร้สายขึ้นเพื่อให้เหมาะสมกับคุณลักษณะและข้อจำกัดของสื่อข้อมูล (Media) แบบไร้สาย ซึ่งเทคโนโลยีนี้คือ WAP นั่นเอง

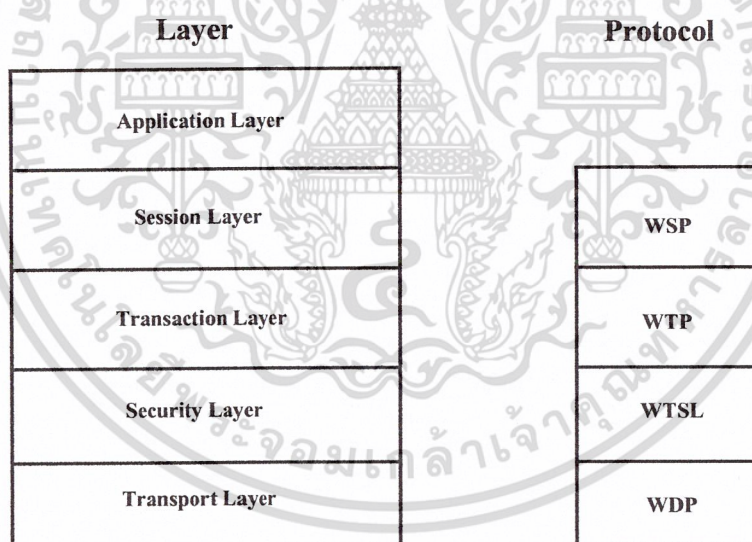
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 เลเยอร์ และ โพรโทคอลในระบบ WAP

โดย WAP จะกำหนดเลเยอร์ในการจัดการส่งข้อมูลเป็น 5 ชั้น โดยโพรโทคอลที่สร้างขึ้นมา 4 ตัวซึ่งจะทำงานในระดับไบนารีเพื่อให้เหมาะสมในการรับส่งข้อมูลผ่านเครือข่ายไร้สายซึ่งมีข้อจำกัดอยู่มากมาย โดยชั้นต่าง ๆ จะประกอบไปด้วย

1. Application
2. Session
3. Transaction
4. Security
5. Transport

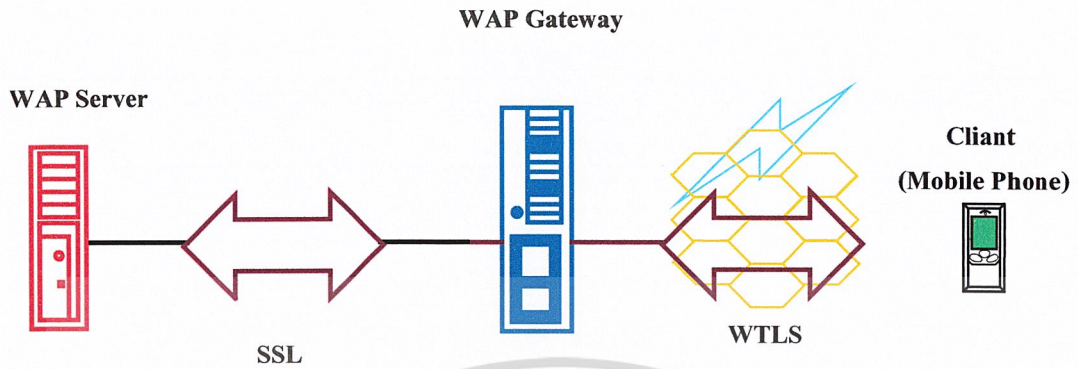
และได้สร้างภาษามาร์กอัปขึ้นมาใหม่ด้วยคือภาษา WML (Wireless Markup Language) ซึ่งเป็นภาษาที่สร้างแท็กและนิยามความหมายขึ้นมาตามรูปแบบของภาษา XML ซึ่งหมายความว่าภาษา WML เป็น XML Vocabulary นั้นเองโดยแต่ละเลเยอร์ทำหน้าที่ดังนี้



รูปที่ 2.16 เลเยอร์ของโพรโทคอลต่างในระบบ WAP

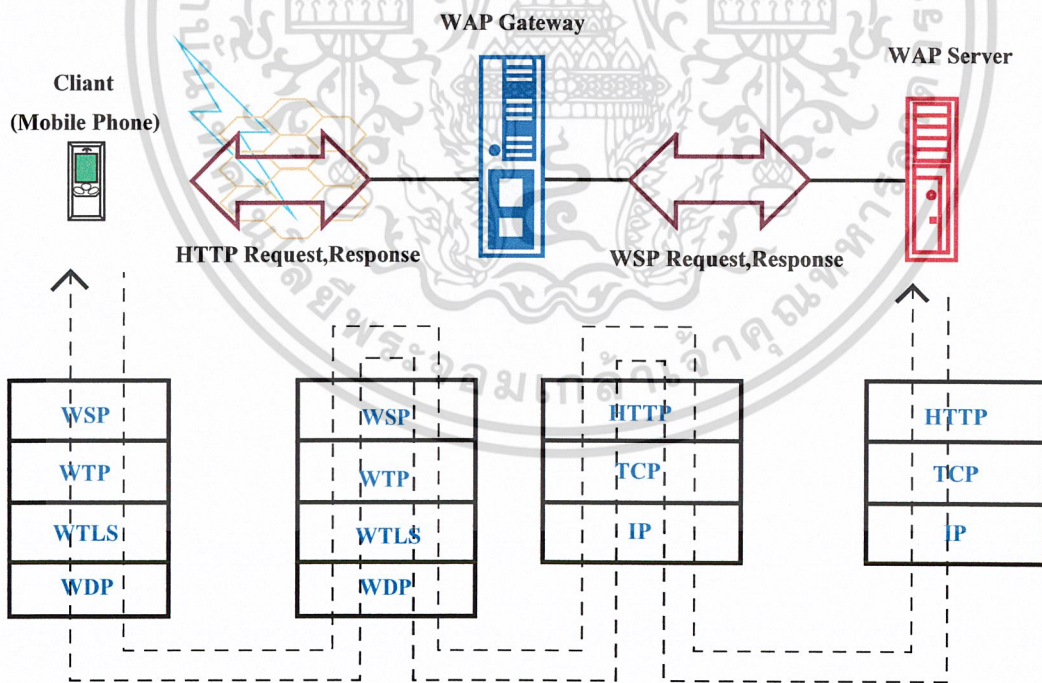
Application Layer คือชั้นที่นำข้อมูลไปประยุกต์หรือชั้นที่ติดต่อกับผู้ใช้ เช่น WAP Browser หรือ Micro WAP Browser ต่าง ๆ โดยจะไม่เสนอเป็นโพรโทคอลหรือข้อตกลงอะไรตายตัว แต่เปิดโอกาสให้สามารถพัฒนา Application ได้โดยสามารถดูแนวทางในการพัฒนาใน WAE (Wireless Application Environment) ซึ่งระบอบองค์ประกอบต่าง ๆ ในการใช้เทคโนโลยี WAP เช่น WML, WMLScript, Interpreter, URL เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 การใช้โปรโตคอลที่ทำหน้าที่รักษาความปลอดภัย

จากการติดต่อของระบบ WAP ดังกล่าว เมื่อนำมาเชื่อมต่อกับระบบอินเทอร์เน็ตจะทำให้เห็นภาพรวมในการติดต่อสื่อสารระหว่างเครื่องโทรศัพท์เคลื่อนที่กับ WAP Server

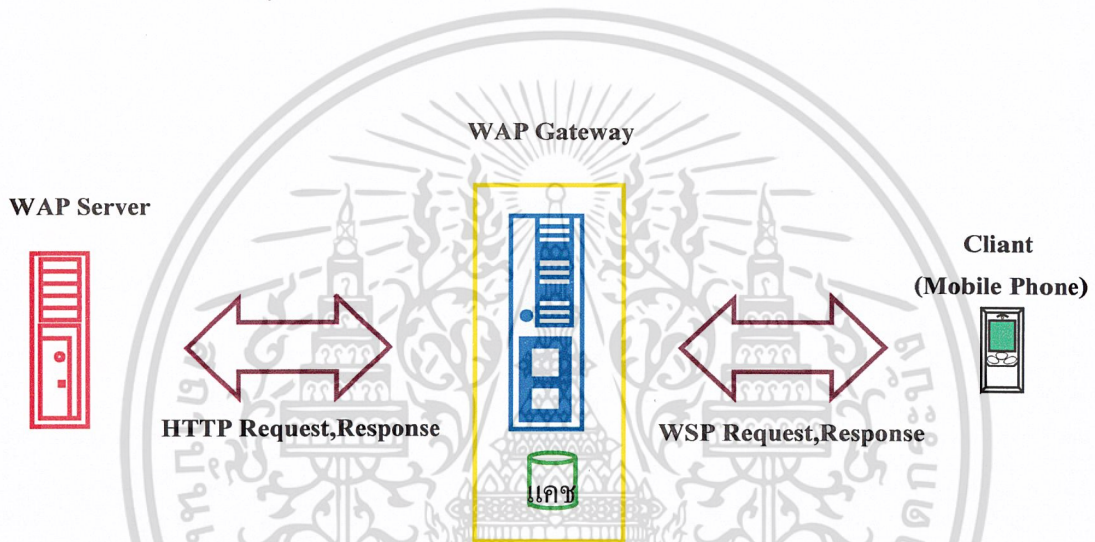


รูปที่ 2.18 การทำงานของ WAP Gateway

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WAP Gateway โดยปรกติแล้วอุปกรณ์ Gateway ถ้าอยู่บนเครือข่ายอินเทอร์เน็ตแล้วจะทำหน้าที่เป็นประตูระหว่างระบบที่ต่างชนิดต่างโปรโตคอลกัน ดังนั้นเมื่อพิจารณาการเชื่อมต่อระหว่าง WAP ซึ่งระบบไร้สายกับระบบอินเทอร์เน็ตนั้นเป็นคนละระบบกันจึงต้องใช้ Gateway เพื่อเปลี่ยนรูปแบบข้อมูลระหว่าง 2 โปรโตคอลรวมถึงต้องจัดการแปลงข้อมูลเพราะ WAP ใช้ข้อมูลเป็นไบนารี

นอกเหนือจากนั้น WAP Gateway ยังทำหน้าที่เป็น Proxy Server อีกด้วย คือจะเก็บไฟล์ WAP Page ที่อัปเดตล่าสุดไว้ในแคชเพื่อลดขั้นตอนระหว่าง WAP Gateway กับ WAP Server

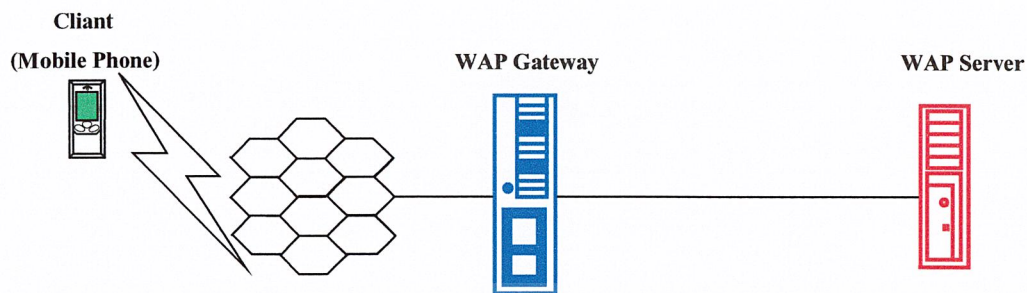


รูปที่ 2.19 WAP Gateway ที่ทำหน้าที่อีกอย่างเป็น proxy server

การเลือกใช้ WAP Gateway มีให้เลือกใช้ได้ 2 ประเภทคือ

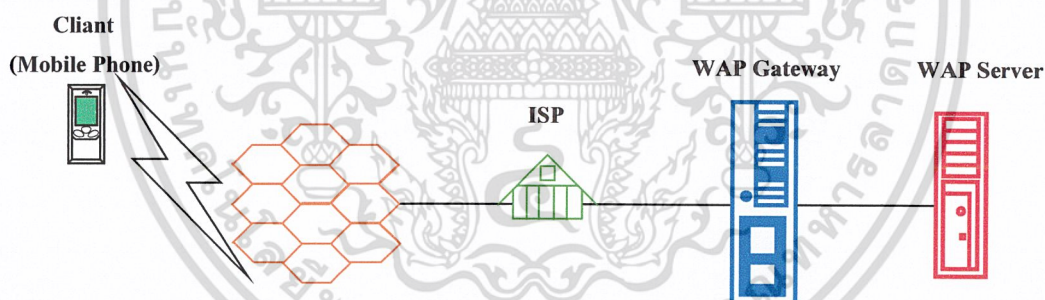
- WAP Gateway จากผู้ให้บริการโทรศัพท์เคลื่อนที่ โดยทั่วไปแล้วเมื่อซื้อโทรศัพท์มาจะยังไม่สามารถใช้ WAP ได้ จะต้องติดต่อผู้ให้บริการเพื่อเปิดบริการชนิดนี้ก่อน จากนั้นผู้ให้บริการก็จะให้ IP Address ของ WAP Gateway มาให้ตั้งค่าในโทรศัพท์ของเรา รวมถึงพอร์ตที่ใช้ในการเชื่อมต่อที่ WAP Gateway ตัวนี้รองรับในกรณีที่โทรศัพท์ของเราสามารถใช้ได้ทั้งโปรโตคอล WSP และ WSP/B โดยการเชื่อมต่อของโปรโตคอล 2 ตัวนี้จะใช้พอร์ตต่างกัน ดังนั้นผู้ให้บริการจะต้องระบุพอร์ตมาให้เราตั้งค่าในเครื่องของเราไปเป็นโปรโตคอลตัวเดียวกับที่ WAP Gateway รองรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 WAP Gateway ของผู้ให้บริการ โทรศัพท์เคลื่อนที่

- WAP Gateway สาธารณะ ซึ่งจะใช้ WAP Gateway ของ ISP ต่าง ๆ โดยผู้ใช้งานจะต้องสมัครเป็นสมาชิกของ ISP นั้นๆ หรือซื้อแพ็คเกจชั่วโมงมาใช้เช่นเดียวกับระบบอินเทอร์เน็ต สำหรับการค้นหารายชื่อของ WAP Gateway สาธารณะจากผู้ให้บริการในเมืองไทยนั้นค้นหาได้ที่ www.wapofthailand.com



รูปที่ 2.21 WAP Gateway ของ ISP ต่าง ๆ

2.4 XML (Extensible Markup Language)

XML เป็นภาษามาร์กอัปอีกแบบหนึ่งนอกเหนือจาก GML, SGML และ HTML โดยภาษามาร์กอัป จะใช้หลักการมาร์กหรือกำหนดรูปแบบของเอกสารเพื่อใน ไปใช้ประโยชน์ในประการใดประการหนึ่ง หรือหลายประการ ซึ่ง XML ได้พัฒนามาจาก SGML ซึ่งมีข้อเด่นในการนำไปใช้ประโยชน์ได้หลายประการ ซึ่งมีข้อดีกว่า HTML ที่มาร์กหรือกำหนดรูปแบบเอกสารเพื่อที่จะให้ไปใช้แสดงผลได้ใน Web Browser ประการเดียวเท่านั้น และมีข้อดีกว่า SGML ตรงที่มีความซับซ้อนน้อยกว่า โดยสาเหตุที่ XML นิยามข้อมูลมาแล้วสามารถใช้งานได้งานได้หลายอย่างคือ XML มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนิยามข้อมูลแบ่งเป็น 2 ส่วน คือ ส่วน Markup (แท็กหรืออิลิเมนต์) ในการนิยาม และส่วนของรูปแบบของ Markup นั้น จึงทำให้ Markup หรือแท็กในภาษา XML จึงไม่ตายตัวแล้วแต่ความเหมาะสมของเอกสาร แล้วสามารถเข้าใจเอกสารได้โดยดูความหมายของสัญลักษณ์

ในการกำหนดรูปแบบหรือนิยามข้อมูลต่าง ๆ ใน XML มีวิธีนิยามอยู่ 2 วิธีคือ DTD (Document Type Definition) และ XML Schema ซึ่งอย่างหลังนี้จะไม่ขอกล่าวถึง

DTD เป็นชื่อเอกสารส่วนที่เป็นกรนิยามข้อมูล ซึ่งอาจจะเก็บไว้ในฉบับเดียวกับเอกสาร XML ขึ้นอยู่กับการ Declare เอกสาร XML ในเอกสาร XML ที่มีรูปแบบถูกต้องตามกฎของ DTD นั้นจะเรียกว่า “Valid”

ในการสร้างเอกสาร XML นั้นจะต้องมีการประกาศ XML Declaration เพื่อบอกให้รู้ว่าเป็นเอกสาร XML Version ของ XML

```
<?xml version= “1.0”?>
```

และส่วนของ Document Typt Declaration เพื่อประกาศเอกสาร DTD ที่ใช้นิยามเอกสาร XML ฉบับนั้น ซึ่งจะประกาศได้ 2 แบบคือ

- ประกาศแบบภายใน (Internal Subset Declaration) ซึ่งจะรวมเอาเอกสาร DTD และ XML ไว้ในเอกสารฉบับเดียวกัน ซึ่งจะนำ DTD ต่อท้ายจากการประกาศ Document Type ได้เลย
- ประกาศแบบภายนอก (External Subset Declaration) ซึ่งเอกสาร DTD จะอยู่นอกเอกสาร XML โดยระบุเป็น URL

ตัวอย่างภาษามาร์กอัปแบบอื่นที่ใช้ XML ในการนิยามข้อมูลเช่น CML ที่ถูกสร้างมาอธิบายเอกสารทางเคมี, MathML ที่ใช้อธิบายเอกสารทางคณิตศาสตร์ รวมถึง WML ที่ถูกออกแบบมาให้แสดงข้อมูลโดยอุปกรณ์ไร้สาย ซึ่งภาษาเหล่านี้เรียกว่า XML Vocabulary

2.5 WML (Wiress Markup Language)

จากที่กล่าวมาข้างต้นแล้วว่า WML เป็น XML Vocabulary ตัวหนึ่ง ซึ่งกลุ่มบริษัท WAP Forum wfh ได้ใช้ XML ในการสร้างกลุ่มแท็กและนิยามความหมายขึ้นมา ดังนั้นจะเห็นได้ว่าในส่วนหัวของเอกสาร WML จะเป็นเอกสาร XML นั้นเอง ซึ่ง WML มีโครงสร้างหลัก ๆ คือ เอกสาร WML 1 ไฟล์ จะเรียกว่า 1 Deck ซึ่งจะอยู่ในแท็ก <wml> และเอกสารที่ใช้ในการแสดงผลทางจอภาพครั้งละ 1 หน้าจอ จะเรียกว่า 1 Card ซึ่งในเอกสาร WML 1 Deck จึงได้หลาย Card เพื่อลดจำนวนครั้งในการติดต่อสื่อสารโดยข้อจำกัดของระบบไร้สาย โดยจะมีรูปแบบหลัก ๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<?xml version= “1.0”?>
<!DOCTYPE wml PUBLIC “_//WAPFORUM//DTD WML 1.1 //EN”
“http://www.wapforum.org/DTD/wml_1.1.xml”>
  <wml>
    <card id=1>
      {
        ข้อมูลการ์ด 1
      }
    </card>
    <card id=2>
      {
        ข้อมูลการ์ด 2
      }
    </card>
  </wml>

```

2.6 โปรแกรม Web Server

การเขียนฝึก WAP Page ที่ต้องการให้แสดงผลเพียงอย่างเดียวอาจจะยังไม่จำเป็นต้องติดตั้ง Web Server แต่ถ้าต้องการเขียน WAP Page ให้เป็นลักษณะ Dynamic ด้วยการใส่ Script ฝั่ง Server เช่น CGI/Perl, ASP, PHP ก็จำเป็นต้องติดตั้ง Web Server วิธีการสร้าง WAP แบบนี้ในขั้นแรกควรออกแบบและทดสอบโดยวิธีจำลองเครื่อง PC ของเราให้เป็น WAB Server เสียก่อนแล้วค่อยอัปโหลดไปไว้ใน WAP Server จริง

โปรแกรม Web Server นั้นมีให้เลือกใช้ได้หลายตัวเช่น Omni HTTPd, PWS, IIS, Apache หรือ Xitami เป็นต้น แล้วแต่ Script ที่เลือกใช้ โดยเลือก โปรแกรม Web Server ที่รองรับ Script ตัวนั้น เช่นถ้าจะเลือกใช้ Script PHP ก็ควรจะเลือกใช้ Apache

2.7 PHP Script

PHP คือ Script ที่ทำงานทางฝั่ง Server หรือ (Server – Side – Script) คือจะทำการประมวลผลตรงฝั่ง Server คล้ายกับ ASP และ SSI ซึ่งสามารถสร้างเว็บแบบไดนามิกได้ สามารถดึงข้อมูลจากฐานข้อมูลได้โดยเอาชื่อของ JAVA และ Pearl เข้าไว้ด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PHP 1.0 ถูกสร้างขึ้นมาเพื่อใช้ในการตรวจสอบจำนวนผู้เข้าชมเว็บเป็นครั้งแรก โดยเป็นเว็บ Rasmus Lerdorf online และช่วยในการจัดการระบบข้อมูล SQL ต่อมาได้ถูกพัฒนาเป็น Version 2.0 ซึ่งออกแบบให้สามารถทำงานร่วมกับแท็กแบบ HTML ได้ สามารถรับค่าที่ได้จากการ Submit จากผู้ใช้ได้ เข้าถึงฐานข้อมูลได้ง่ายกว่า Version ก่อน

PHP ได้ถูกนำมาใช้ในเว็บไซต์ทางธุรกิจในปี 1996 โดยใช้ครั้งแรกในเว็บ SuperCuts (WWW.supercut.com) ซึ่งใช้ PHP ในการสร้างฐานข้อมูลสำหรับลูกค้า ต่อมาในปี 1998 ก็ได้มี PHP Version 3 ออกมาโดยได้รับการแก้ไขข้อบกพร่องของรุ่นก่อน

PHP เป็น Script ที่มีการกำหนดขอบเขตของโปรแกรม เพื่อแบ่งพื้นที่กันระหว่าง PHP และ HTML จึงเป็นการสะดวกที่จะเอาไว้ในส่วนไหนก็ได้ ข้อดีอีกประการของ PHP คือมีการ Compile และ Execute ได้อย่างรวดเร็ว และทำงานบน OS ได้หลายตัว และเป็นฟรีแวร์ซึ่งสามารถดาวน์โหลดได้จากเว็บไซต์ของ PHP ได้เลย

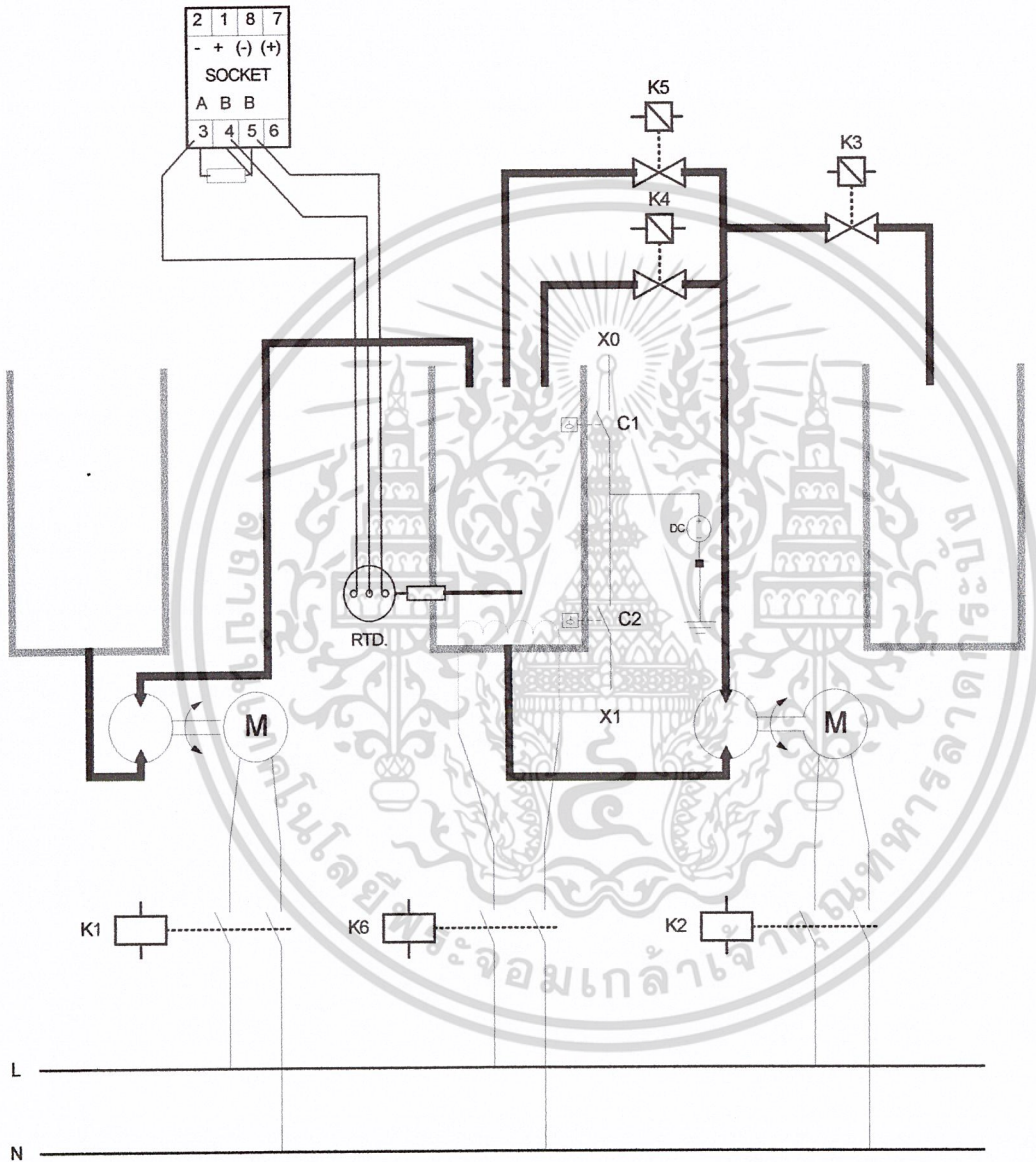


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การดำเนินงานวิจัย

3.1 การออกแบบทางด้านฮาร์ดแวร์



รูปที่ 3.1 การออกแบบทางด้านฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากความต้องการที่จะควบคุมอุณหภูมิและอัตราการไหลในกระบวนการทางเคมีเราได้มีการออกแบบโดยการใช้ฮีสเตอร์และอุปกรณ์วัดอุณหภูมิ RTD ในการควบคุมและรายงานอุณหภูมิสำหรับอัตราการไหลนั้นเราได้ออกแบบให้ใช้การควบคุมปั๊ม และ โซลินอยวาล์วเพื่อควบคุมอัตราการไหลของกระบวนการ นอกจากนี้ยังมีการออกแบบให้มีการใช้สวิตช์ลูกลอยเพื่อควบคุมระดับของเหลวให้อยู่ในระดับที่เครื่องทำงานได้โดยไม่เกิดความเสียหายแก่อุปกรณ์และรองรับการทำงานแบบอัตโนมัติ

3.2 การออกแบบการควบคุมอุปกรณ์ทางด้านฮาร์ดแวร์ด้วย PLC

ในการนำ PLC มาควบคุมการทำงานของอุปกรณ์ทางด้านฮาร์ดแวร์ก่อนอื่นเราต้องแบ่งอุปกรณ์ทางด้านฮาร์ดแวร์ออกเป็น 2 ประเภทคือ

1. อุปกรณ์ที่เราต้องการควบคุมการทำงานซึ่งได้แก่ ฮีสเตอร์, ปั๊ม, โซลินอยวาล์ว
2. อุปกรณ์ที่เป็นตัวแสดงสถานะซึ่งได้แก่ อุปกรณ์วัดอุณหภูมิ RTD และอุปกรณ์ที่ใช้จำกัดระดับของของเหลวในระบบซึ่งได้แก่ สวิตช์ลูกลอย C1, C2

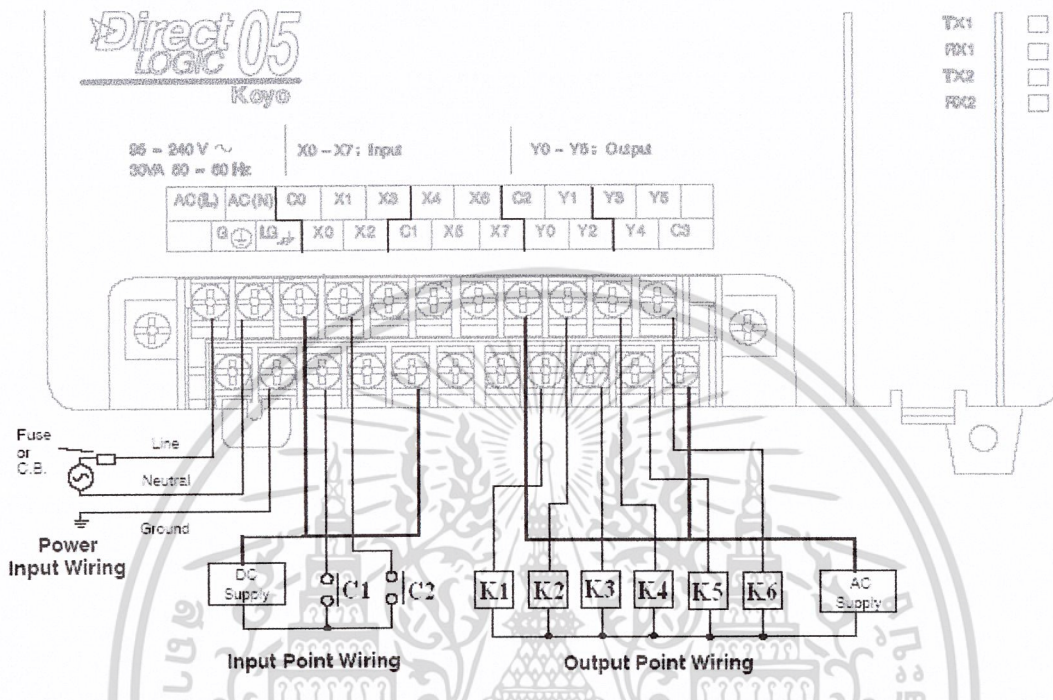
สำหรับการต่ออุปกรณ์ประเภทที่ 1 เพื่อให้ PLC ควบคุมการทำงานจะต้องต่อเข้ากับทางด้าน Output ของ PLC ซึ่งการควบคุมการทำงาน จะขึ้นอยู่กับการโปรแกรม Ladder ที่ PLC ทำการประมวลผลทำให้ Output ตัวไหนของ PLC ทำงาน แต่การต่ออุปกรณ์ดังกล่าวไม่สามารถต่อได้โดยตรงเพราะไฟที่จ่ายออกจาก Output ไม่สามารถขับให้อุปกรณ์เหล่านั้นทำงานได้ จึงต้องมีการใช้ Relay เพิ่มเข้ามาเพื่อต่อกับแหล่งจ่ายไฟ 220 V.AC. จากภายนอก เพื่อขับให้อุปกรณ์แต่ละตัวทำงานและต่อคอยล์ของ Relay เข้ากับ Output ของ PLC เพื่อให้ Relay แต่ละตัวทำงานตอบสนองกับ Output ของ PLC

สำหรับการต่ออุปกรณ์ประเภทที่ 2 ที่เป็นอุปกรณ์ซึ่งเป็นตัวแสดงสถานะของอุณหภูมิจะต้องต่อเข้าทางด้าน Input ของ PLC ซึ่งค่าของอุณหภูมิที่ต่อเข้ากับเครื่อง PLC จะต้องอยู่ในรูปของสัญญาณไฟฟ้า ดังนั้นจึงต้องต่อ Input ของ PLC เข้ากับวงจรแปลงค่าอุณหภูมิให้เป็นสัญญาณทางไฟฟ้า ซึ่งวงจรจะแปลงอุณหภูมิจากการวัดของ RTD ให้เป็นสัญญาณทางไฟฟ้าซึ่งเป็นรูปแบบของสัญญาณ Analog ที่ค่าของกระแสไฟฟ้าจะเปลี่ยนแปลงตามค่าอุณหภูมิที่วัด แล้วจึงนำค่าที่ได้จากการวัดต่อเข้าเครื่อง PLC โดยผ่านเข้าทางการ์ด Analog Input Module ที่ติดตั้งอยู่บนเครื่อง PLC

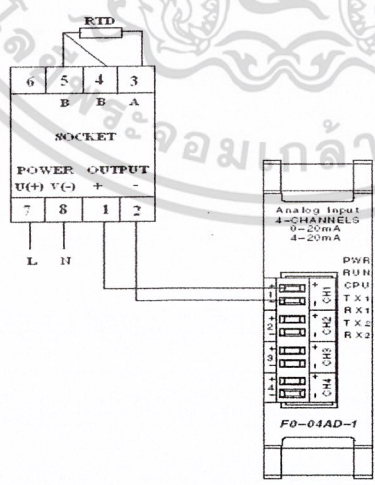
สำหรับสวิตช์ลูกลอย C1 และ C2 ซึ่งส่วนนี้เราต้องการรับสถานะปิดหรือเปิดของสวิตช์มาใช้เป็นเงื่อนไขในการควบคุม จึงสามารถต่อเข้าได้โดยตรงเข้าทาง Input ของ PLC X0. และ X1. ตามลำดับ

การเชื่อมต่อระหว่างอุปกรณ์ทางด้านฮาร์ดแวร์ทั้งสองประเภทกับ PLC แสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 การต่ออุปกรณ์ในการควบคุมเข้ากับ Input และ Output ของ PLC



รูปที่ 3.3 การต่อเครื่องแปลงอุณหภูมิเป็นสัญญาณไฟฟ้ามาตรฐานเข้ากับการ์ด Analog Input Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบส่วนการสั่งงาน PLC ด้วยโปรแกรม Visual Basic

ในการนำ PLC มาควบคุมอุปกรณ์ทางด้านฮาร์ดแวร์ เราต้องเขียนโปรแกรม Ladder ซึ่งมีลักษณะเป็นการนำหน้าสัมผัส (Contactor) มาต่อกันเป็นวงจรคอนโทรลเพื่อควบคุมให้ส่วนที่เป็น Output ทำงานตามที่เราร้องการ โดยการใช้โปรแกรม Visual Basic สร้างหน้า From เพื่อการสั่งงานควบคุม Contactor ในโปรแกรม Ladder ของ PLC

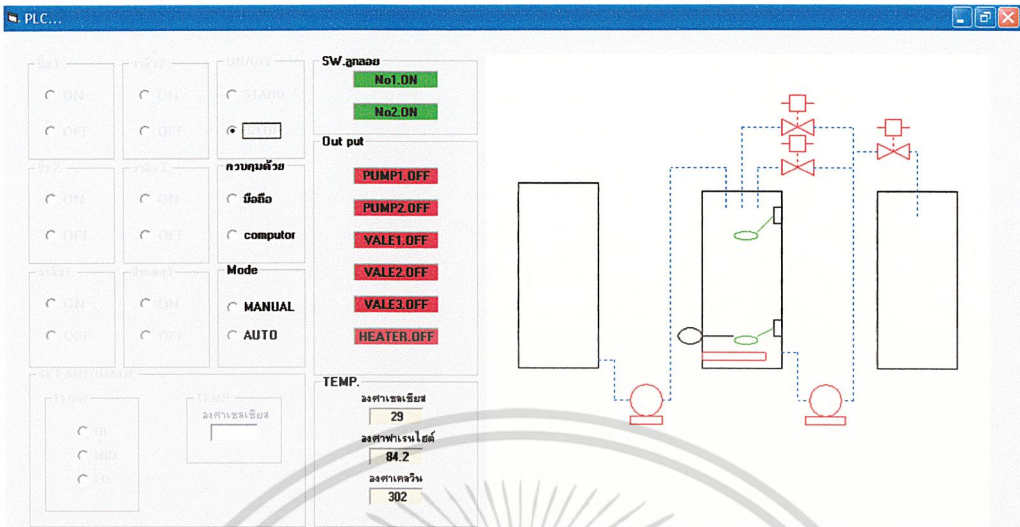
การสร้างหน้า From เพื่อการสั่งงาน PLC ของโปรแกรม Visual Basic เพื่อติดต่อกับ Contactor ในโปรแกรม Ladder ของ PLC โดยให้ผู้ใช้เลือกสั่งงานผ่านทางหน้า From แต่ในการเข้าถึง Contactor, Input และ Output แต่ละตัวใน PLC นั้น ต้องใช้คอนโทรล Tex Box ของโปรแกรม Visual Basic ในการรับค่าหรือส่งค่ากับ PLC เราต้องใช้โปรแกรม DS Data Server เป็นตัวกลางในการเชื่อมต่อการสื่อสารระหว่าง คอนโทรล Tex Box ของโปรแกรม Visual Basic กับ Address ของ Contactor, Input, Output ใน PLC ที่เราร้องการส่งค่าและรับค่าสถานะไปยัง From ของโปรแกรม Visual Basic

โดยการออกแบบเราได้กำหนดให้มีการสั่งงานควบคุมได้ใน 3 รูปแบบคือ

1.การควบคุมแบบ Manual เป็นการสั่งการควบคุมอุปกรณ์โดยตรงซึ่งสามารถเลือกสั่งงานให้อุปกรณ์แต่ละตัวสามารถทำงานได้อย่างอิสระโดยไม่มีการป้องกัน

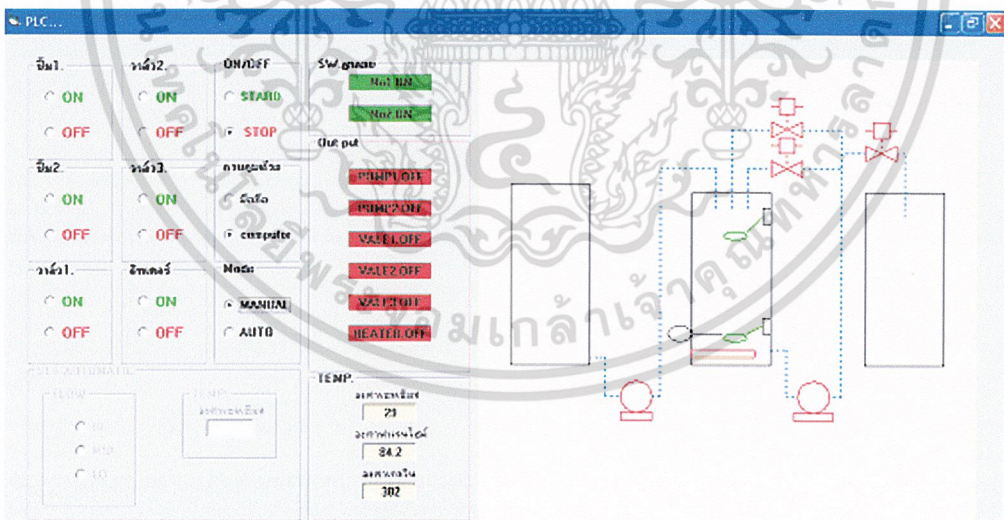
2.การควบคุมแบบ Automatic ซึ่งเป็นการสั่งการควบคุมโดยการตั้งค่าอุณหภูมิและกำหนดอัตราการไหล โดยเลือกตั้งค่าผ่านทางหน้า From ของโปรแกรม

3.การควบคุมผ่านทางโทรศัพท์เคลื่อนที่ซึ่งออกแบบให้ทำงานตามรูปแบบ Automatic แต่จะรับการตั้งค่าอุณหภูมิและอัตราการไหลผ่านทางโทรศัพท์เคลื่อนที่และจะมาแสดงผลของค่าอุณหภูมิและอัตราการไหลบนหน้า From เหมือนการทำงานในรูปแบบ Automatic



ระบบสั่งงาน PLC เพื่อควบคุมกระบวนการทางเคมีโดยใช้โทรศัพท์เคลื่อนที่

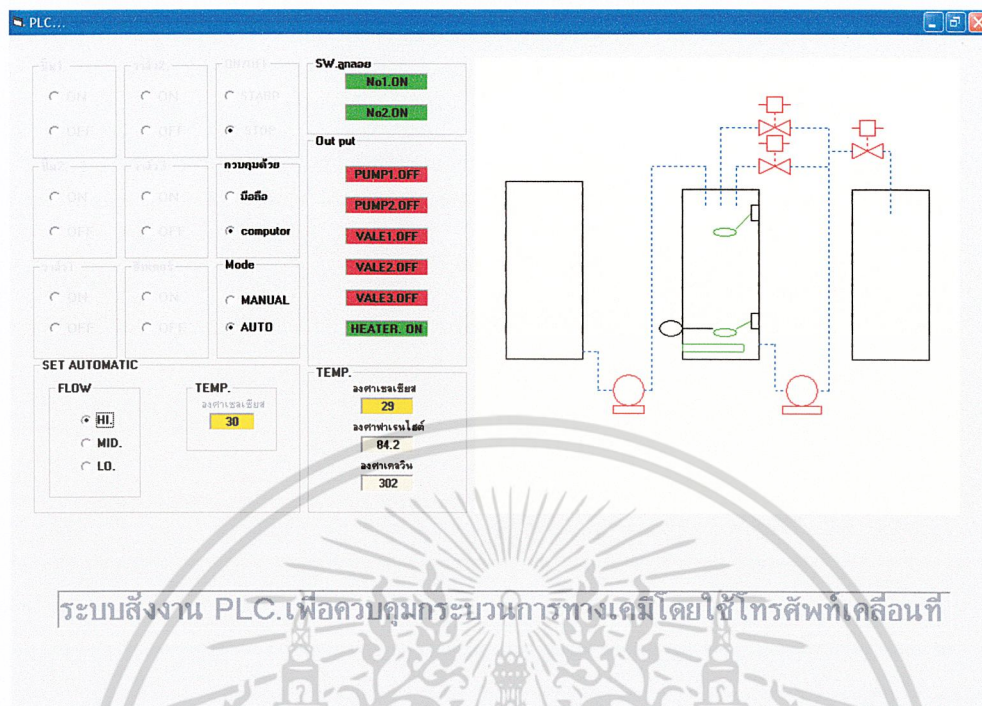
รูปที่ 3.4 หน้าโปรแกรมเมื่อเริ่มต้นใช้งาน



ระบบสั่งงาน PLC เพื่อควบคุมกระบวนการทางเคมีโดยใช้โทรศัพท์เคลื่อนที่

รูปที่ 3.5 หน้าโปรแกรมเมื่อทำงานในรูปแบบ Manual

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

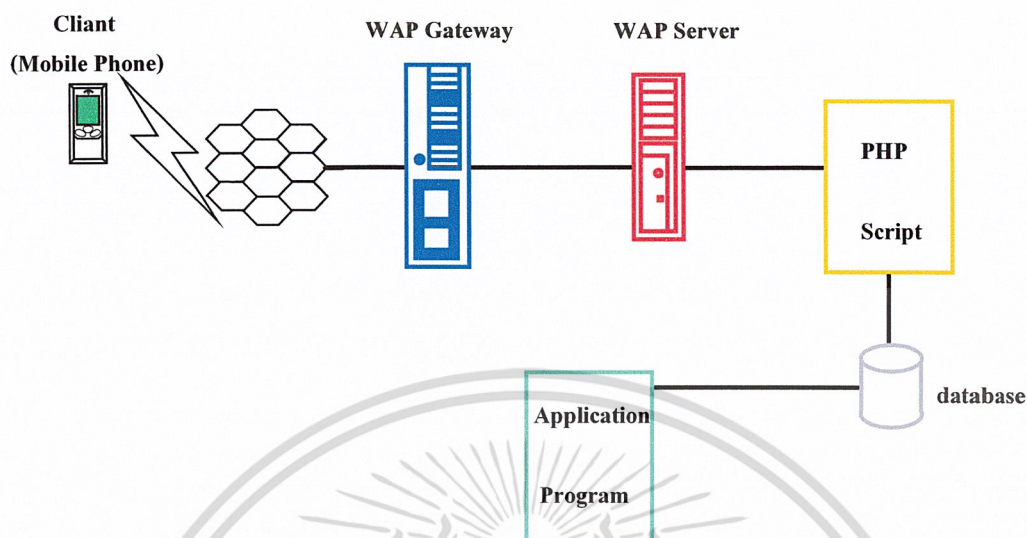


รูปที่ 3.6 หน้าโปรแกรมที่ทำงานในรูปแบบ Automatic

3.4 การออกแบบในส่วนที่ติดต่อระหว่างโทรศัพท์เคลื่อนที่กับข้อมูลที่เป็น Input และ Output

การออกแบบในส่วนที่ติดต่อระหว่าง โทรศัพท์เคลื่อนที่กับข้อมูลที่เป็น Input และ Output ของโปรแกรมประยุกต์ในส่วนควบคุมนั้น ทำได้โดยช่วงแรกอาศัยเทคโนโลยี WAP ในการสร้าง WAP Site ขึ้นมาเพื่อติดต่อแสดงผลและรับค่า Input จากผู้ใช้ผ่านทาง WAP Browser ซึ่งค่า Input และ Output ของโปรแกรมประยุกต์ที่จะนำไปแสดงผลและควบคุมนั้นจะเก็บไว้เป็นฐานข้อมูลเพื่อให้ Browser ติดต่อควบคุมได้โดยอาศัยการออกแบบ WAP Site ให้เป็นแบบไดนามิก แบบ 3Tier ซึ่งอาศัย Script เพื่อเป็น Application ที่ใช้ติดต่อจัดการฐานข้อมูลได้ง่าย โดย WAP Site จะแสดงข้อมูล WML ที่ใช้ร่วมกับ Script PHP เพื่อประมวลผลการจัดการฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

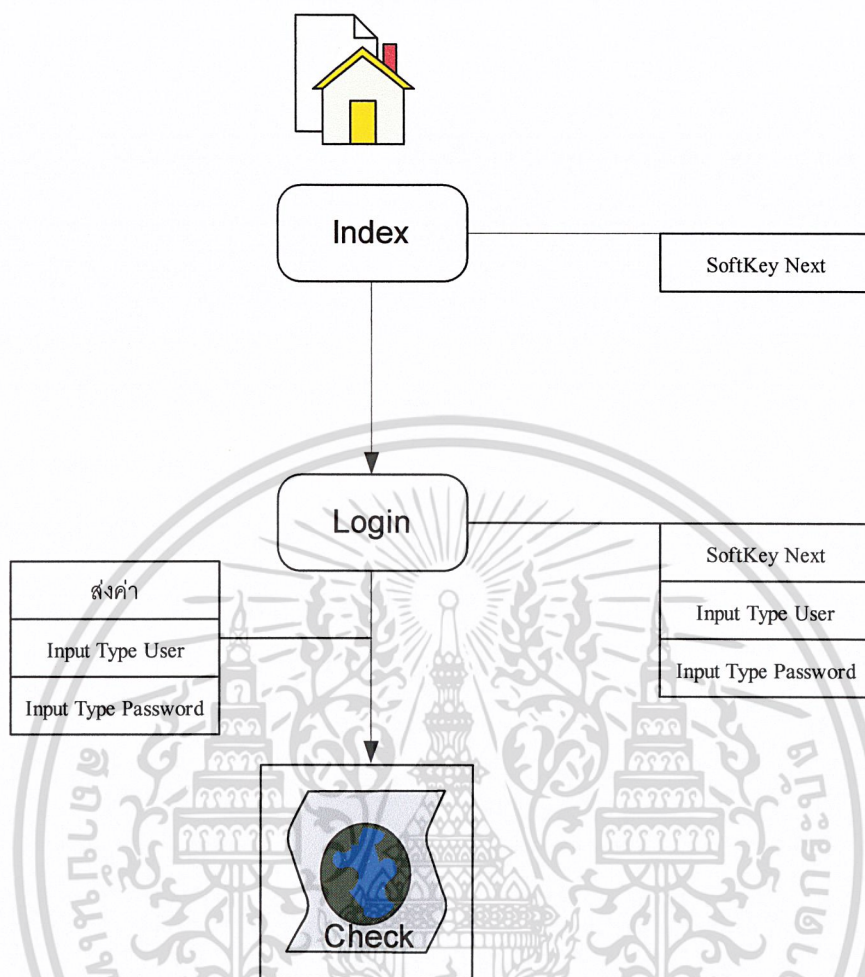


รูปที่ 3.7 การติดต่อระหว่างโทรศัพท์เคลื่อนที่กับ WAP Server

3.5 การออกแบบ WAP Site

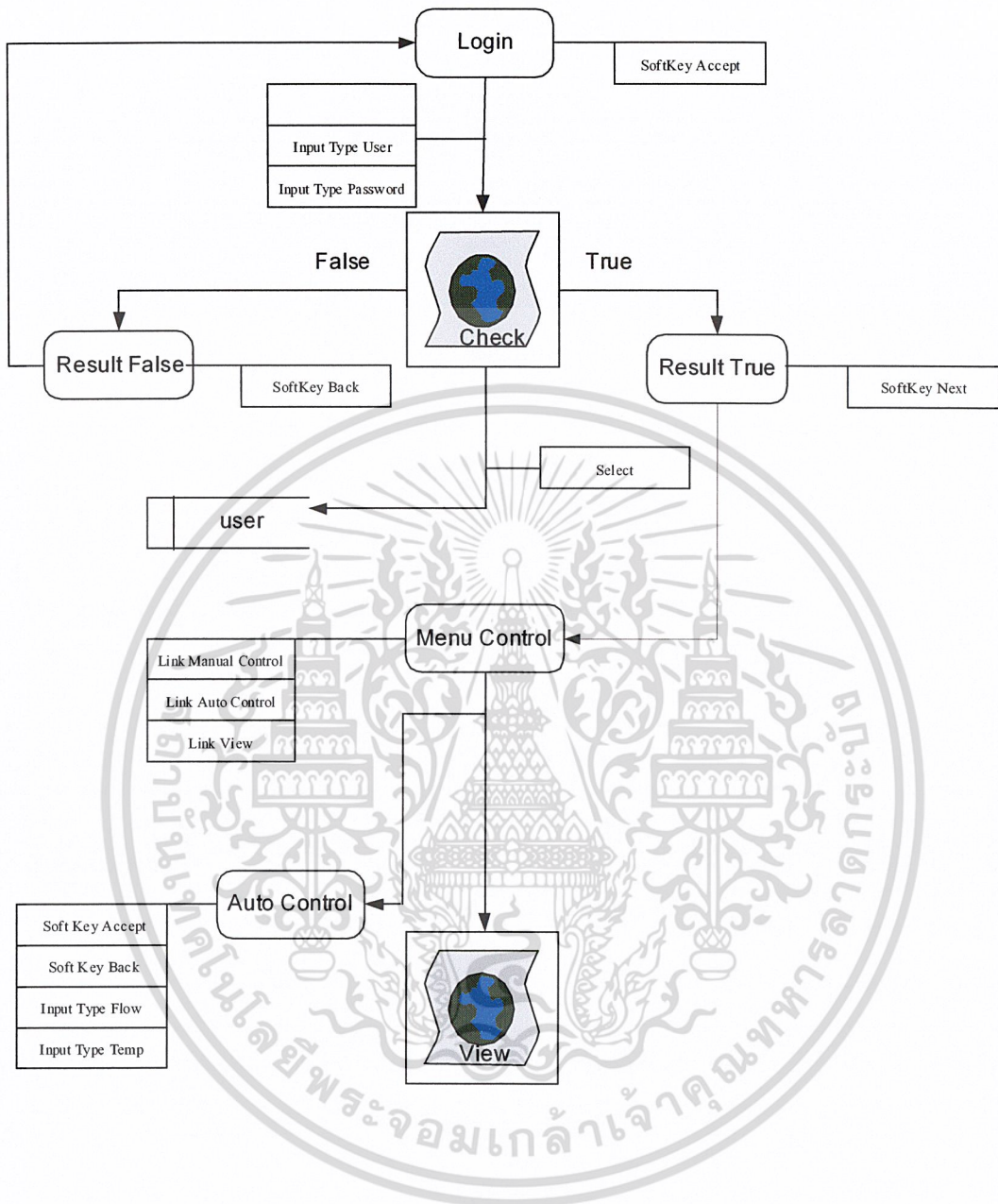
3.5.1 การออกแบบเบื้องต้น

ในการออกแบบเพื่อให้สามารถดูค่าและป้อน Input เพื่อควบคุมอุปกรณ์จะใช้ WML ใช้งานร่วมกับ PHP ซึ่งการออกแบบจะออกแบบการทำงานของ WAP เป็นขั้นตอนได้ดังนี้



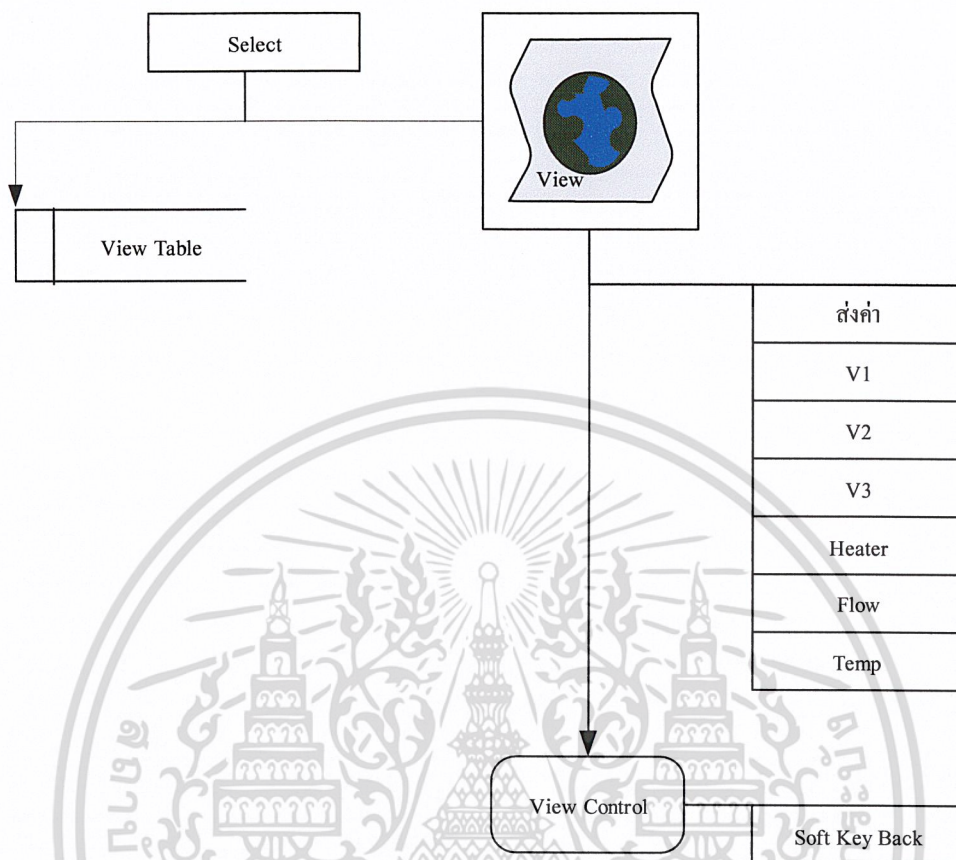
รูปที่ 3.8 การออกแบบ WAP SITE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



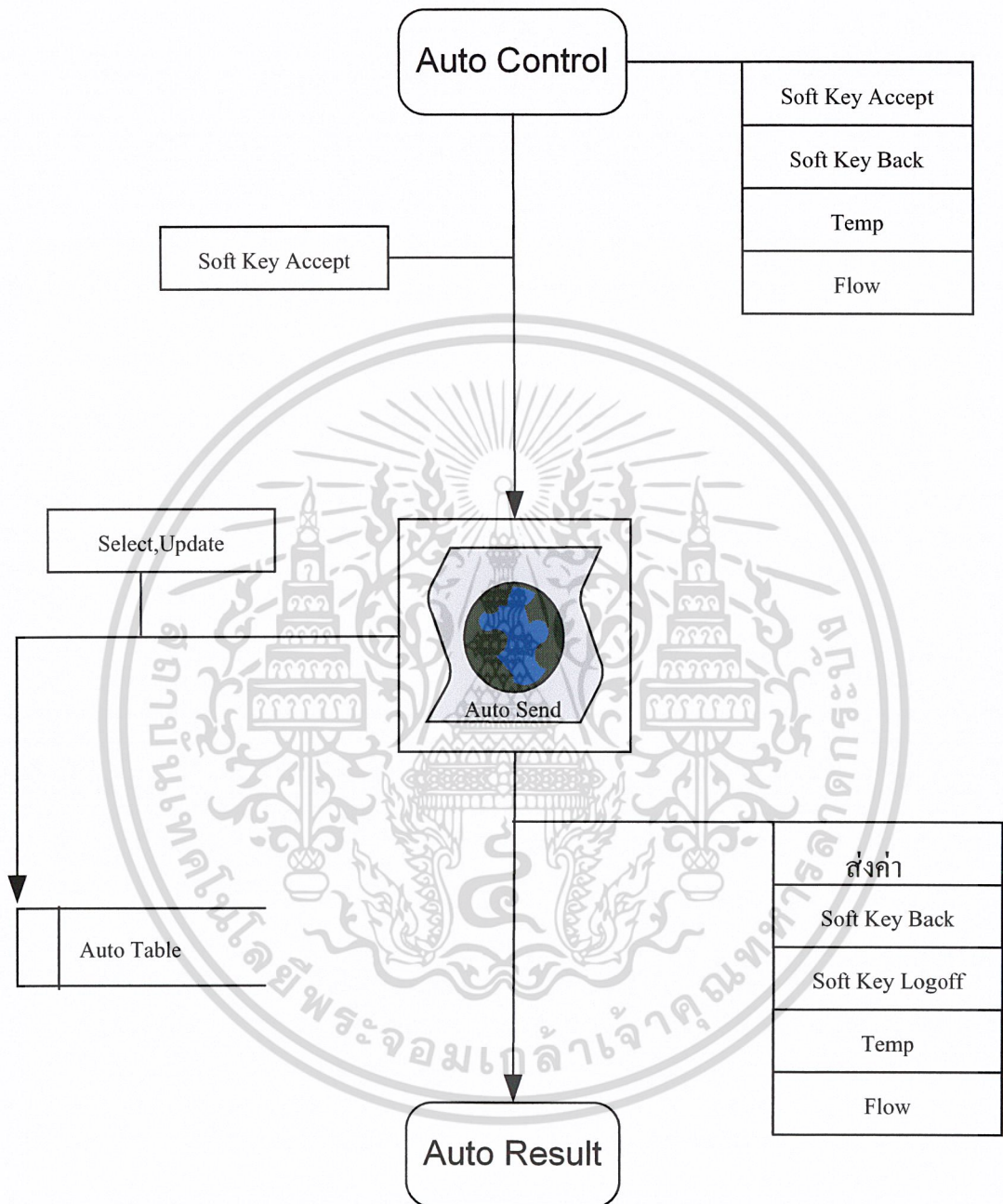
รูปที่ 3.9 การออกแบบ WAP SITE ต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 การออกแบบ WAP SITE ต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 การออกแบบ WAP SITE ต่อ

จากการออกแบบจะเห็นได้ว่า แบ่ง WML เป็น 2 ส่วน คือ ส่วนแรกคือไฟล์ WML ที่ใช้ในการแสดง WAP Page เพียงอย่างเดียว อีกส่วนคือ WML ที่มี Script แทรกอยู่ ซึ่งจะทำหน้าที่ติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการรับส่งข้อมูลและประมวลผลต่าง รวมทั้งติดต่อกับฐานข้อมูลด้วย โดยจะมีขั้นตอนการทำงานดังนี้

- ขั้นตอนที่ 1 เมื่อพิมพ์ URL ที่อยู่ของ WAP แล้ว Server จะส่งไฟล์ Wml หน้า Index เพื่อให้ WAP Browser แสดงผล จากนั้นก็จะรับค่าจากผู้ใช้ซึ่งหน้านี้มีให้รับค่าอยู่ค่าเดียว คือ Next ถ้ารับค่าเข้ามาเป็น Next ก็จะเรียกการติดต่อไปซึ่งเป็นการเข้าสู่ระบบ ซึ่งผู้ใช้จะต้องใส่ user name และ password ที่ถูกต้อง(ตามที่เก็บในฐานข้อมูลพนักงาน) เพื่อเข้าการใช้งาน จากนั้นรอรับค่า Next, Back เพื่อใช้งานต่อไปหรือกลับไปหน้าแรก

- ขั้นตอนที่ 2 เมื่อรับค่า Next มากี่จะส่งค่า user name และ password ที่รับเข้ามาในตอนแรกไปประมวลผลใน Script chek ซึ่งจะต้องมีหน้าที่ดึงข้อมูล user name และ password จากฐานข้อมูลมาเพื่อตรวจสอบด้วยแล้วส่งผลรับมายัง Browser ถ้าถูกต้อง จะแสดงข้อความแล้วรอรับค่า Next เพื่อร้องขอเอกสารหน้าต่อไป แต่ถ้าไม่ถูกต้องก็จะไปที่ไฟล์ อีกตัวเพื่อแสดงข้อความว่ารหัสผิดพลาด แล้วรอรับค่า Next เพื่อกลับไปสู่หน้า login อีกครั้ง

- ขั้นตอนที่ 3 เมื่อเข้ามาในระบบได้แล้วซึ่งแสดงใน menu.wml จะมี Link ในการเลือกเพื่อใช้งาน 2 กรณีด้วยกันคือ ขอดูข้อมูล หรือ ตั้งค่าการทำงานแบบ Automatic แล้วรอค่า Next, Back

- ขั้นตอนที่ 4 ถ้าเลือก การทำงานแบบขอดูข้อมูล จะนำค่าต่าง ๆ สถานะอุปกรณ์ในสถานะปัจจุบันมาแสดงในหน้าแล้วส่งไปให้ Browser ซึ่งหลังจากนี้มีการทำงานได้อีก 2 แบบคือกลับไปหน้าก่อนหน้านี้ หรือออกจากระบบ

- ขั้นตอนที่ 5 ถ้าเลือกในกรณีที่เป็นการทำงานแบบ ตั้งค่าการทำงานแบบ Automatic ก็จะนำไฟล์ WML Card Auto control มาแสดงแล้วรับค่าในการตั้งอุปกรณ์ต่าง ๆ แล้วรอค่าเป็น Next

- ขั้นตอนที่ 6 เมื่อตกลงแล้วก็ส่งค่าที่รับมาให้กับ Script Manual Send เพื่อประมวลผลพร้อมกันนั้นก็ echo ค่าต่าง ๆ แล้วส่งไฟล์นี้ไปแสดงที่ Browser ใน Card Auto View แล้วสามารถกลับไปหรือออกจากระบบได้

3.6 การออกแบบตารางเก็บข้อมูล

ฐานข้อมูลที่ใช้มีด้วยกัน 3 ตารางคือ

3.6.1 ฐานข้อมูลที่เกี่ยวข้องผู้ใช้ และรหัสผ่านเพื่อใช้ในการตรวจสอบผู้เข้าใช้ซึ่งจะมี

Operation ในการทำงาน 4 อย่างคือ Insert , Update , Delete , Select

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

User_ID	User_Name	Password
1	somsak	Sak01

รูปที่ 3.12 การออกแบบตารางผู้ใช้

3.6.2 ฐานข้อมูลที่เก็บค่าการทำงานต่าง ๆ ของอุปกรณ์จากโปรแกรมอินเทอร์เฟซ ซึ่งจะทำงานเปรียบเสมือนบัพเฟอร์ระหว่างโปรแกรมอินเทอร์เฟซกับ WAP Browser ไว้สำหรับตรวจสอบสถานะต่าง ๆ ของอุปกรณ์ ซึ่งจะมี Operation ในการทำงาน 2 อย่างคือ Update และ Select

Output_ID	V1	V2	V3	Heater	Temp	Flow
1	ON	OFF	ON	OFF	60	30

รูปที่ 3.13 การออกแบบตารางแสดงค่าต่าง ๆ ของระบบ

3.6.3 ฐานข้อมูลการตั้งค่าควบคุมแบบอัตโนมัติ จะเก็บการตั้งค่าควบคุมไว้เพื่อให้โปรแกรมอินเทอร์เฟซดึงไปใช้งาน ซึ่งจะทำงานเปรียบเสมือนบัพเฟอร์ระหว่างโปรแกรมอินเทอร์เฟซกับ WAP Browser ซึ่งจะมี Operation ในการทำงาน 2 อย่างคือ Update และ Select เหมือนแบบแรกเช่นกัน

Auto_ID	Temp	Flow
1	60	h/m/l

รูปที่ 3.15 การออกแบบตารางการควบคุมการทำงานแบบ Automatic

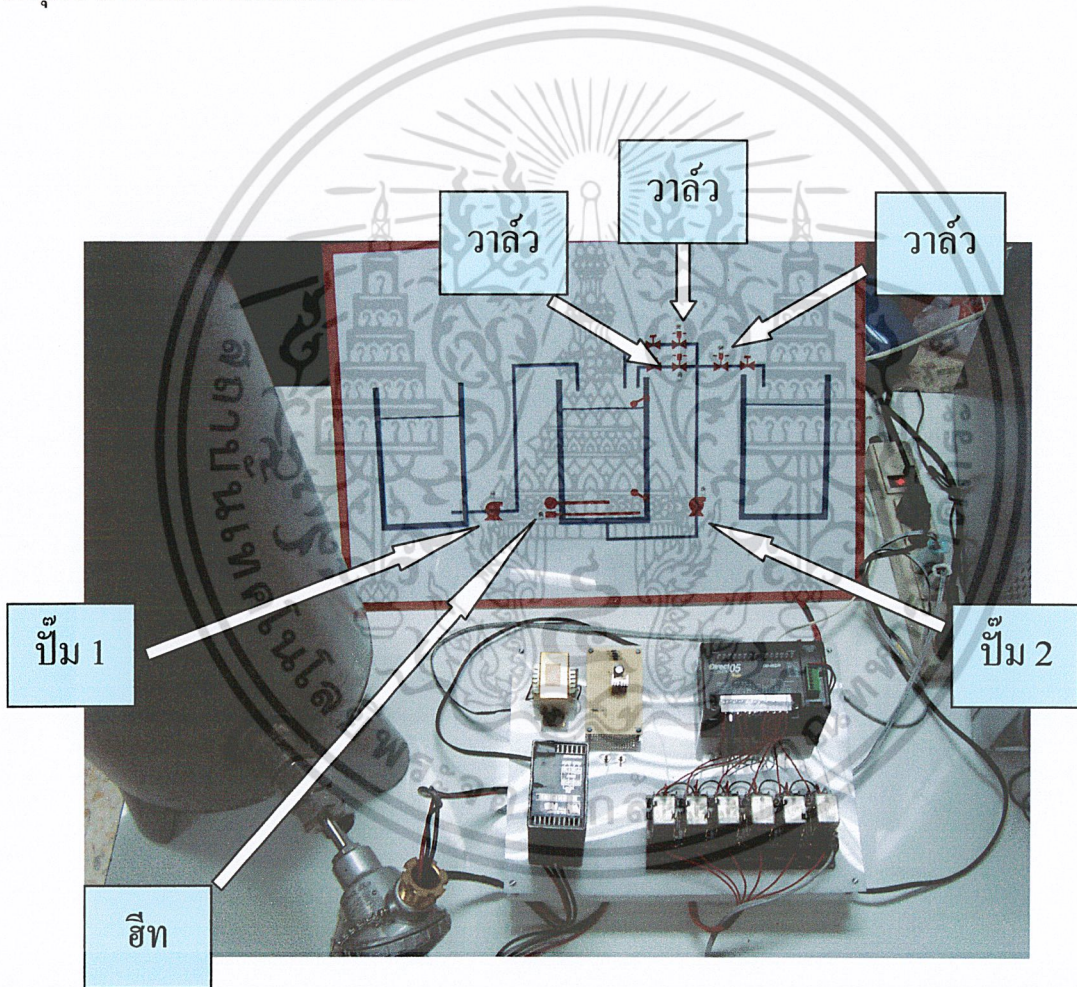
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 ขั้นตอนการตรวจสอบควบคุมอุปกรณ์โดยใช้โปรแกรมควบคุมจากคอมพิวเตอร์ในแบบ Manual

การควบคุมอุปกรณ์แบ่งได้ 3 รูปแบบคือการควบคุมแบบ Manual , Automatic โดยคอมพิวเตอร์ ซึ่งจะควบคุมผ่าน โปรแกรมอินเตอร์เฟซ และการควบคุมโดยโทรศัพท์เคลื่อนที่ซึ่งจะควบคุมได้เพียงแบบ Automatic เท่านั้น

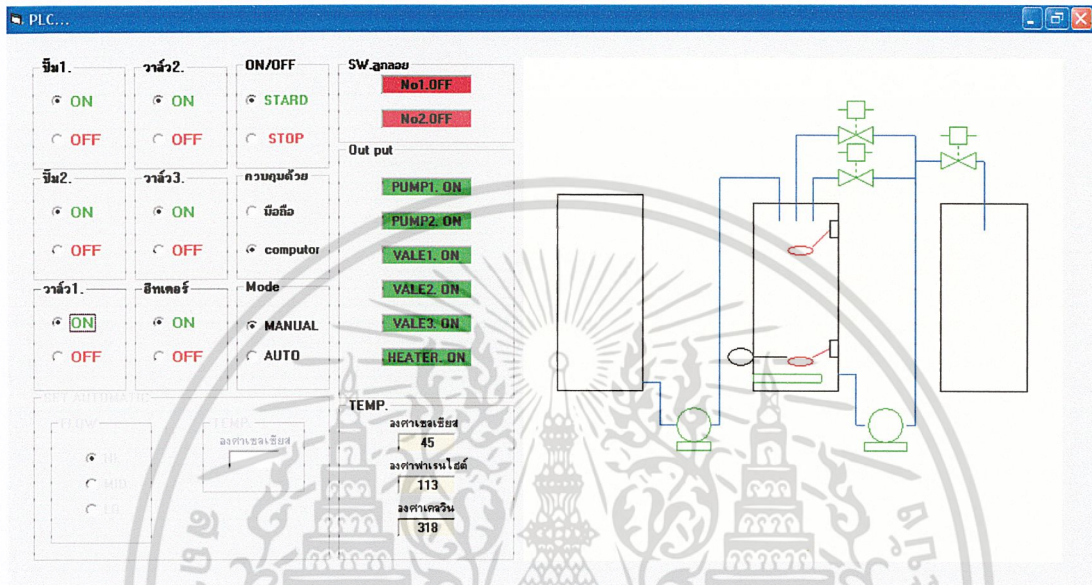


รูปที่ 4.1 อุปกรณ์ที่จะนำมาควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1 เปิดโปรแกรมควบคุมจากคอมพิวเตอร์และปรับประเภทการทำงานเป็นแบบ Manual

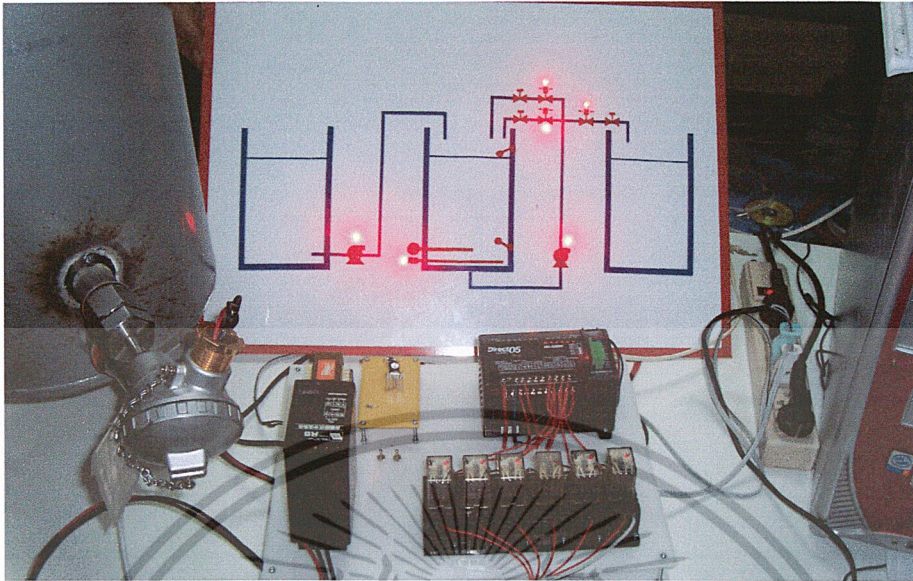
4.1.2 ตั้งค่าอุปกรณ์ต่าง ๆ ให้อุปกรณ์ทุกตัวเปิด



รูปที่ 4.2 ผลการควบคุมแบบ Manual(บนคอมพิวเตอร์)

4.1.3 ผลการทดสอบของการควบคุมแบบ Manual อุปกรณ์ทุกตัวทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



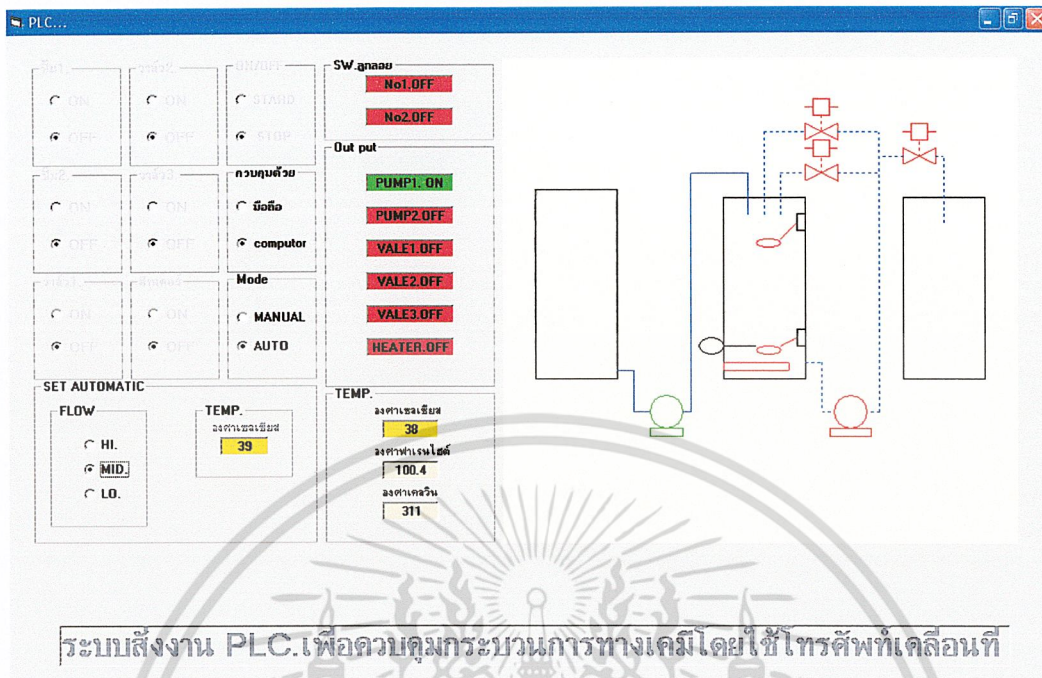
รูปที่ 4.3 ผลการควบคุมแบบ Manual(อุปกรณ์)

4.2 การควบคุมแบบอัตโนมัติ

การควบคุมแบบอัตโนมัติเป็นการควบคุม โดยกำหนดตัวแปร 2 อย่างคือ อัตราการไหล และอุณหภูมิ แล้วโปรแกรม Ladder ของ PLC จะทำการควบคุมอุปกรณ์ทุกอย่างของระบบอย่างอัตโนมัติ

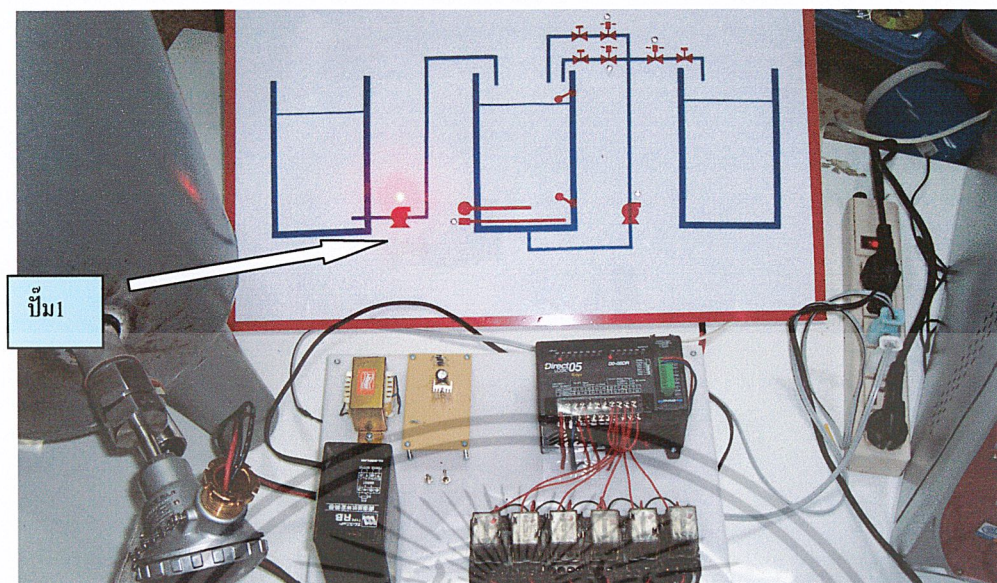
4.2.1 ตั้งค่าการทำงานเป็นประเภทอัตโนมัติ(AUTO)

4.2.2 ตั้งค่าอัตราการไหลเป็นระดับปานกลาง และอุณหภูมิเป็น 39 องศาเซลเซียส



รูปที่ 4.4 การตั้งค่าการทำงานแบบอัตโนมัติ

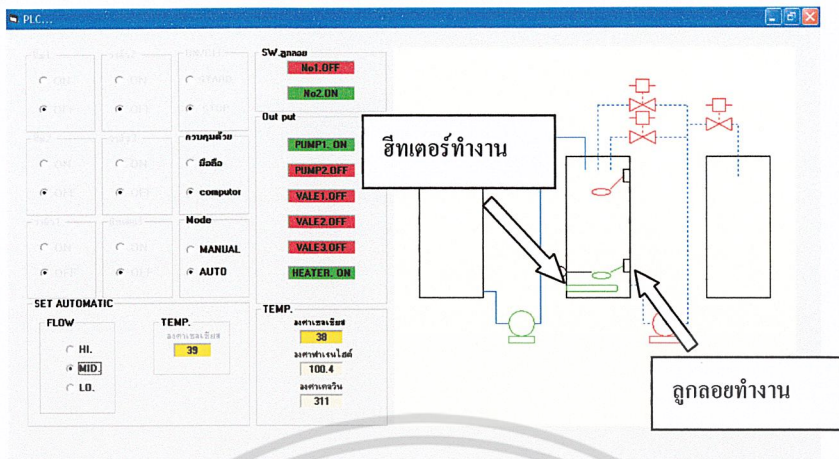
จากรูปเป็นการกำหนดการทำงานแบบอัตโนมัติโดยให้ อัตราการไหลเป็นระดับ ปานกลาง และอุณหภูมิเป็น 39 องศาเซลเซียส สังเกตว่าเมื่อไม่มีปริมาณสารระดับหนึ่งซึ่งไม่ทำให้ฮีทเตอร์เสียหาย(เนื่องจากฮีทเตอร์ที่ใช้เป็นแบบที่ใช้ในของเหลวเท่านั้น)ฮีทเตอร์จะยังไม่ทำงานเนื่องจากตั้ง สวิตช์ถูกลอย(SW.ถูกลอย No.2)ตรวจเช็คระดับไว้



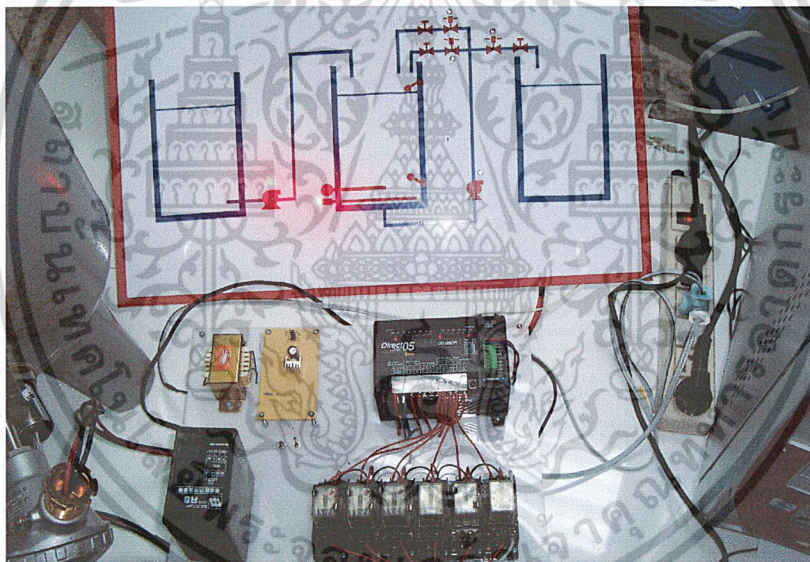
รูปที่ 4.5 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติ(บนอุปกรณ์)

จากรูป...บนอุปกรณ์ก็ได้ผลเช่นเดียวกัน คือฮีตเตอร์ยังไม่ทำงานจนกว่าปั๊มตัวที่ 1 จะทำการสูบน้ำเข้ามาจนเพียงพอเสียก่อน

4.2.3 เมื่อปั๊มตัวที่ 1 สูบสูบน้ำเข้ามาจนได้ระดับที่สวิทช์ถูกปล่อยตัวต่างทำงานแล้ว ฮีตเตอร์ก็สามารถทำการอุ่นสารได้



รูปที่ 4.6 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติเมื่อมีสารเพียงพอ(บนคอมพิวเตอร์)

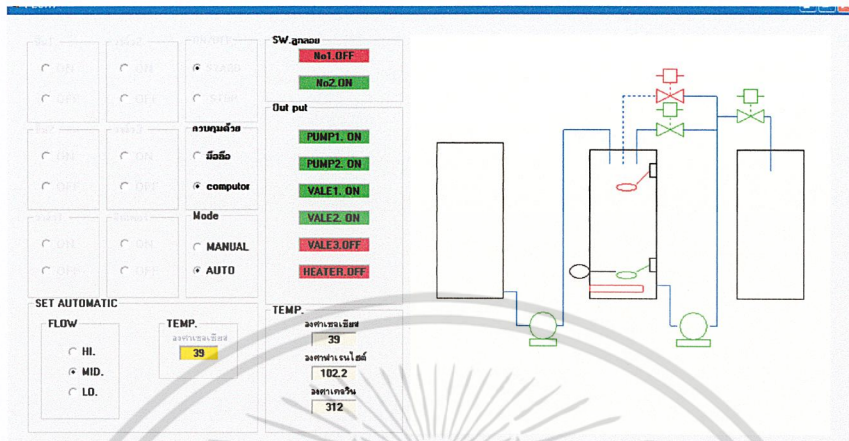


รูปที่ 4.7 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติเมื่อมีสารเพียงพอ (บนอุปกรณ์)

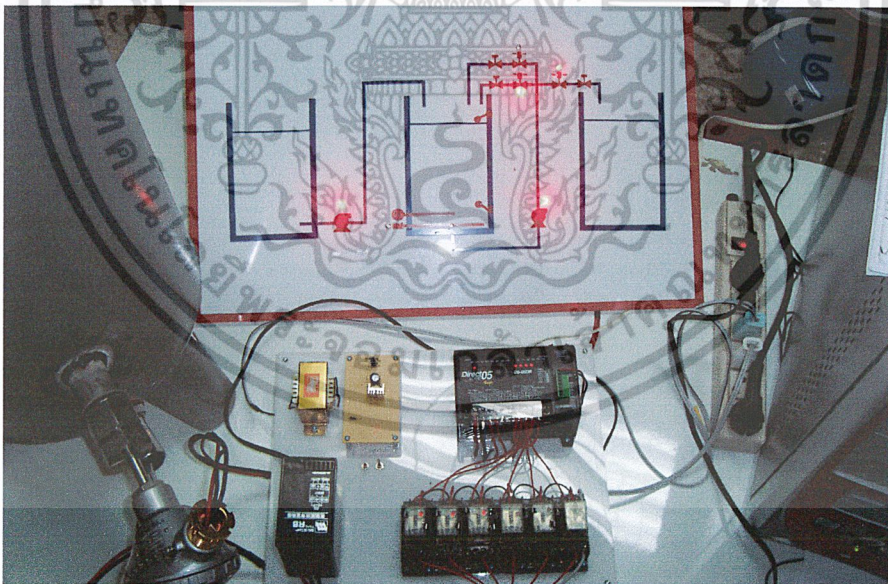
จากอุปกรณ์เมื่อป้อนตัวที่ 1 สูบสูบน้ำเข้ามาจนได้ระดับที่สวิทซ์ลูกลอยตัวล่างทำงานแล้ว ฮีตเตอร์ก็สามารถทำการอุ่นสารได้เช่นกัน

4.2.3 เมื่ออุณหภูมิที่วัดได้เท่ากับอุณหภูมิที่ตั้งไว้ ฮีตเตอร์จะหยุดทำงาน วาล์ว 1 และ วาล์ว 2 ทำงาน เนื่องจากตั้งค่าอัตราการไหลไว้เป็นระดับปานกลาง(MID.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติต่อเมื่ออุณหภูมิถึงที่ตั้งไว้ (บนคอมพิวเตอร์)

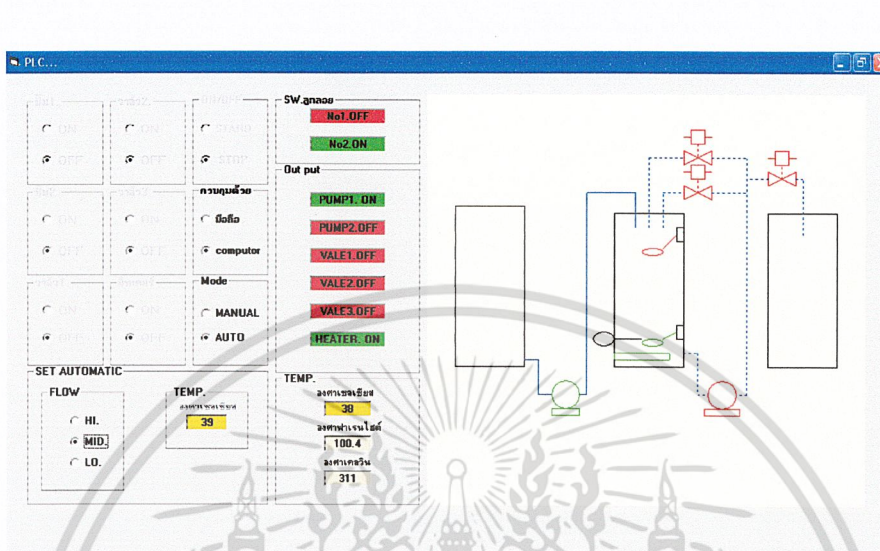


รูปที่ 4.9 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติต่อเมื่ออุณหภูมิถึงที่ตั้งไว้(บนอุปกรณ์)

จากรูปจะเห็นได้ว่าทั้งอุปกรณ์และการแสดงผลในโปรแกรมเป็นไปได้อย่างถูกต้อง

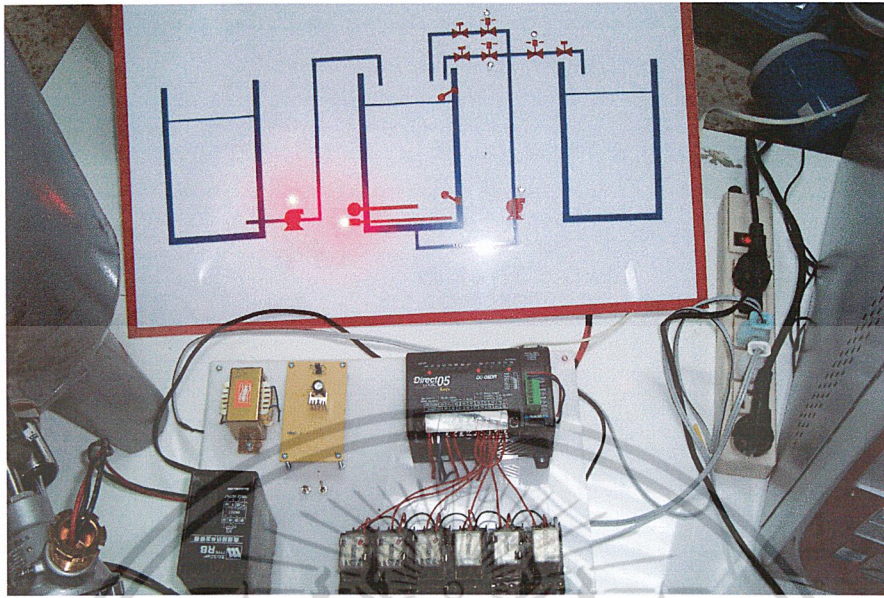
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 เมื่ออุณหภูมิลดลง วาล์วทุกตัวและปั๊มตัวที่ 2 จะปิด และฮีตเตอร์ทำงาน



รูปที่ 4.10 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติเมื่ออุณหภูมิลดจากที่ตั้งไว้ (บนคอมพิวเตอร์)

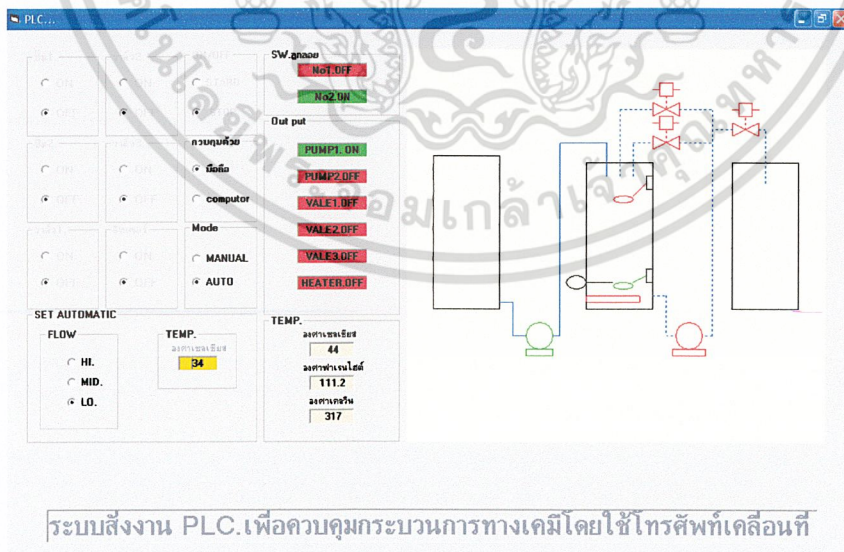
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 ผลการทดสอบการควบคุมการทำงานแบบอัตโนมัติเมื่ออุณหภูมิลดจากที่ตั้งไว้ (บนอุปกรณ์)

4.3 การทดลองการควบคุมโดยโทรศัพท์เคลื่อนที่

4.3.1 ตั้งค่าประเภทการควบคุมไว้เป็นแบบ “ควบคุมด้วยมือถือ”



รูปที่ 4.12 การใช้งานในส่วนของการควบคุมผ่านโทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Host	Database	Table	Data	SQL Query
wap / auto: 1 Records (1 retrieved)				
	id	f	t	
	1		34	

รูปที่ 4.13 เรคคอร์ดที่เก็บค่าควบคุมที่ส่งมาจากโทรศัพท์เคลื่อนที่



รูปที่ 4.14 เมื่อส่งค่าควบคุมจาก โทรศัพท์เคลื่อนที่

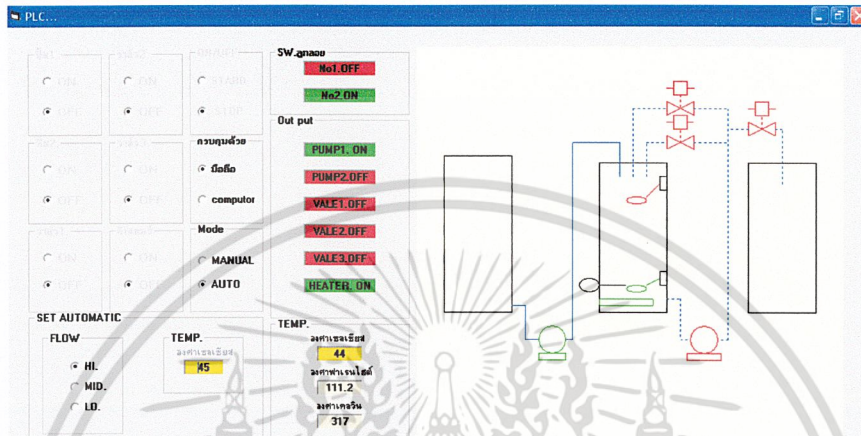
4.3.2 เมื่อกำหนดค่าให้อัตราไหลเป็นระดับความเร็วสูง(high)

Host	Database	Table	Data	SQL Query
wap / auto: 1 Records (1 retrieved)				
	id	f	t	
	1	h	45	

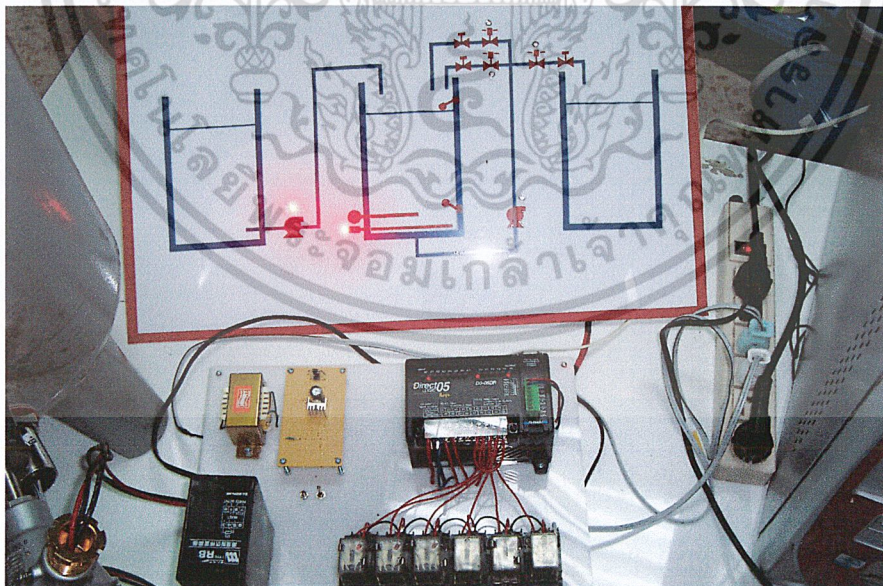
รูปที่ 4.15 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปจะเห็นได้ว่าเรคคอร์ดค่าควบคุมแบบอัตโนมัติ(auto) ใน คอลัมน์ของอัตราการไหล (f) และคอลัมน์อุณหภูมิ(t) จะมีค่าเป็น “h” และ “45” ตามลำดับ



รูปที่ 4.16 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่

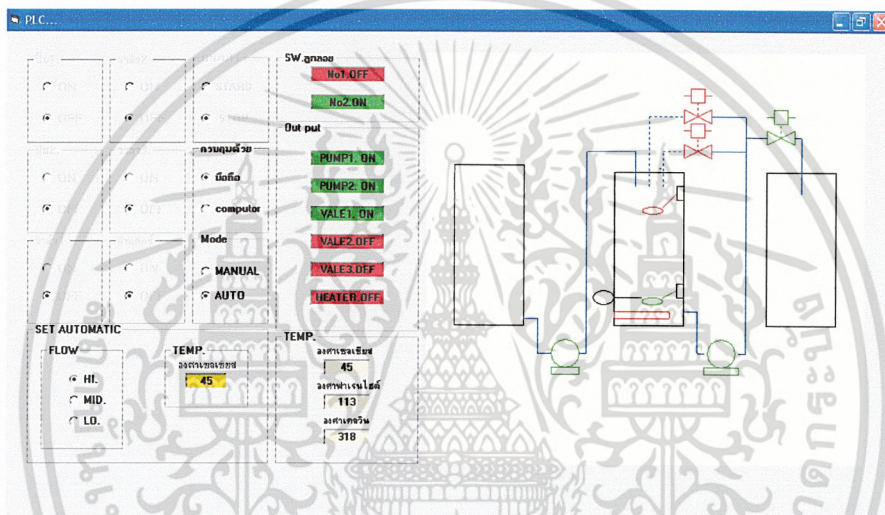


รูปที่ 4.17 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

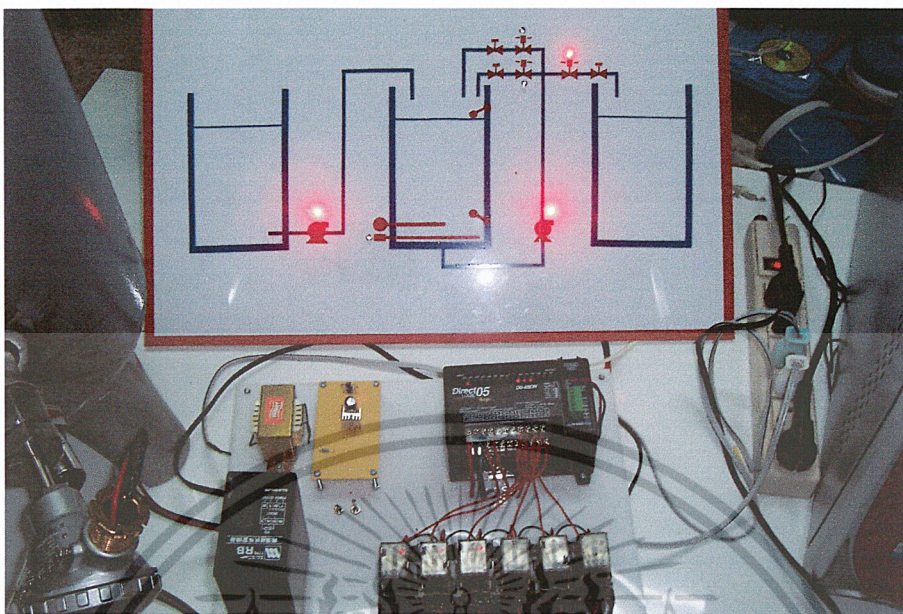
จากรูปที่ 4.16 และ 4.17 จะเห็นได้ว่าค่าควบคุมอัตราการไหลและอุณหภูมิในส่วนของการทำงานแบบอัตโนมัติเป็นค่าที่ส่งมาจากโทรศัพท์เคลื่อนที่ และเมื่อมีระดับสารเพียงพอ(สวิตช์ลुकลอยตัวล่างทำงาน) ฮีทเตอร์ก็จะทำงาน

4.3.3 เมื่ออุณหภูมิจนถึงอุณหภูมิที่ตั้งไว้ ฮีทเตอร์จะหยุดทำงานและวาล์วตัวที่ 1 ทำงานและปั๊มตัวที่ 2 ทำงาน



รูปที่ 4.18 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่

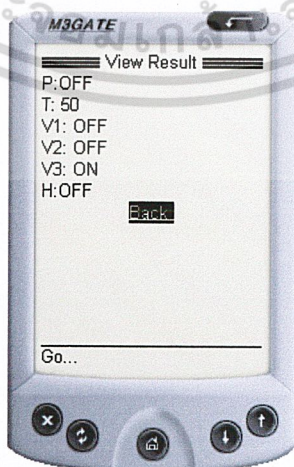
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 ผลการทดสอบการส่งค่าควบคุมจากโทรศัพท์เคลื่อนที่

จากรูปที่ 4.18 และ 4.19 จะเห็นได้ว่าเมื่ออุณหภูมิเพิ่มขึ้นถึงระดับที่ต้องการแล้วฮีทเตอร์จะหยุดทำงานและวาล์วตัวที่ 1 ทำงานและปั๊มตัวที่ 2 จะทำงาน

4.3.4 การตรวจสอบการทำงานของระบบจากโทรศัพท์เคลื่อนที่ โดยการเข้าไปที่เมนู view จะแสดงผลการทำงานต่าง ๆ ของระบบ



รูปที่ 4.20 การตรวจสอบการทำงานของระบบจากโทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Host		Database		Table		Data		Query		
wap / control: 1 Records (1 retrieved)										
	id	v1	v2	v3	p1	p2	d:	dswz	h	t
	1	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	50

รูปที่ 4.21 รีเคคอร์ดเซ็ตของสถานะการทำงานของระบบ(control)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและวิจารณ์ผลการทดลอง

จากการทดลองในบทที่ 4 สามารถใช้โปรแกรม Visual Basic ในการนำมาใช้เขียนโปรแกรมเพื่อสั่งงานควบคุมและแสดงค่าสถานะต่าง ๆ ของ PLC เพื่อสั่งงานควบคุมไปยังอุปกรณ์ที่ต่ออยู่กับ Output ของ PLC และสามารถวัดค่าอุณหภูมิมาแสดงบนหน้าโปรแกรม Visual Basic ที่เขียนขึ้น แต่ยังคงมีความคลาดเคลื่อนของอุณหภูมิอยู่บ้างเล็กน้อย และสามารถรับค่าตัวแปรที่เป็นตัวแปรสำหรับเก็บค่าอุณหภูมิและอัตราการไหลที่ได้จากการส่งค่าตัวแปรที่ได้จากโทรศัพท์เคลื่อนที่มาเป็นเงื่อนไขในการควบคุมอุปกรณ์ในรูปแบบ Automatic เพื่อสั่งการควบคุมไปยัง PLC เพื่อควบคุมอุปกรณ์ต่าง ๆ ในกระบวนการให้ทำงานได้ตามต้องการ

แต่จากการทดลองสั่งงานจากโปรแกรม Visual Basic ไปยัง PLC ยังมีค่า delay เกิดขึ้นเล็กน้อย ซึ่งไม่ส่งผลกระทบต่อกระบวนการที่ต้องการควบคุมมากนัก ในส่วนของค่าอุณหภูมิที่คลาดเคลื่อนสามารถแก้ปัญหาได้ โดยการเขียนโปรแกรม Visual Basic ในการคำนวณค่าอุณหภูมิที่ต้องการให้แสดงผล ซึ่งต้องอาศัยการทดลองวัดค่าอุณหภูมิอย่างละเอียด เพื่อนำมาเปรียบเทียบในการปรับแก้ค่าอุณหภูมิที่ผิดพลาดให้น้อยที่สุด

บรรณานุกรม

สรราช อ้อยศรีสกุล, เริ่มคิดเริ่มสร้างเริ่มใช้ XML, บริษัท วิตตี้ กรุ๊ป จำกัด, พิมพ์ครั้งที่ 1, 2544

อนุโชติ วุฒิพรพงษ์, พันซ์เทพ แก้วมงคล, สร้าง WAP ด้วย WML Script, สำนักพิมพ์อินโฟเพรส, พิมพ์ครั้งที่ 1, 2543

กิตติภูมิ วรรณิตร, MySQL ถาม-ตอบ ครอบคลุมจรรยา, บริษัท วิตตี้ กรุ๊ป จำกัด, พิมพ์ครั้งที่ 1, 2545

สัจจะ จรัสรุ่งรวีวร, คู่มือการเขียนโปรแกรมและใช้งาน Visual Basic 6.0, บริษัท ไอ ดี ซี อินโฟ ดิสทริบิวเตอร์ เซ็นเตอร์ จำกัด, พิมพ์ครั้งที่ 1, 2544

ณรงค์ ต้นชีวะวงศ์, ระบบ PLC, สำนักพิมพ์ ส.ส.ท., พิมพ์ครั้งที่ 7, 2545



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ก.

Source code WML , PHP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

test.wml

```
<?xml version="1.0"?>
<!-- created by WAPtor (http://www.wapdrive.net/) -->
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

```
<wml>
```

```
<!-- THIS IS THE FIRST CARD IN THE DECK -->
```

```
<card id="MainCard" title="This is a first card">
```

```
<do type="accept">
```

```
<go href="http://161.246.73.205/chk.php" >
```

```
<postfield name = "us" value="$ (us)"/>
```

```
<postfield name = "ps" value="$ (ps)"/>
```

```
</go>
```

```
</do>
```

```
<p>
```

```
User:
```

```
<input type="text" name="us"/>
```

```
</p>
```

```
<p>
```

```
Password:
```

```
<input type="password" name="ps"/>
```

```
</p>
```

```
</card>
```

```
</wml>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

chk.php

```
<?
header("Content-Type: text/vnd.wap.wml");
echo "<?xml version='1.0'?>\n";
echo "<!DOCTYPE wml PUBLIC '-//WAPFORUM//DTD WML 1.1//EN'
'http://www.wapforum.org/DTD/wml_1.1.xml'>\n";
?>

<wml>
<card title="result">
<do type="accept">
<go href="http://161.246.73.205/menu.php" >
</go>
</do>
<p>
<?php
include("conf.inc");
$result=mysql_db_query($db,"select * from $tb where User='$us' AND Passwd='$ps'");
$numrow= mysql_fetch_array($result);
if($numrow==0){

    echo "User and Password Not matchs ";

    }else{
echo "Login Press";
}

    mysql_close();
?>
</p>
</card>
</wml>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

menu.php

```
<?
header("Content-Type: text/vnd.wap.wml");
echo "<?xml version=\"1.0\"?>\n";
echo "<!DOCTYPE wml PUBLIC \"-//WAPFORUM//DTD WML 1.1//EN\"
\"http://www.wapforum.org/DTD/wml_1.1.xml\">\n";
?>

<wml>

<card id="menu" title="Menu Control" newcontext="true">
<p>
    <anchor>
<go href="http://161.246.73.205/send4.php"/>
View Output</anchor>
</p>
<p>
    <anchor>
<go href="http://161.246.73.205/auto.wml"/>
Auto</anchor>

</p>
</card>
</wml>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

auto.wml

```
<?xml version="1.0"?>
<!-- created by WAPtor (http://www.wapdrive.net/) -->
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
<!-- THIS IS THE AUTO CONTROL -->
<card id="MainCard" title="AUTO">
<do type="accept">
<go href="http://161.246.73.205/send3.php" >
<postfield name = "f" value="$(f)"/>
<postfield name = "t" value="$(t)"/>
</go>
</do>
<p>
<b>choose f</b><br/>:
<select name="f">
<option value="h">high</option>
<option value="m">midium</option>
<option value="l">low</option>
</select>
</p>
<p>
temp:
<input type="text" name="t"/>
</p>
</card>
</wml>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

send 3.php

```
<?
header("Content-Type: text/vnd.wap.wml");
echo "<?xml version='1.0'?>\n";
echo "<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
'http://www.wapforum.org/DTD/wml_1.1.xml'>\n";
?>

<wml>
<card title="Result Control">
<p>
<?
echo "f:$f";
?>
</p>
<p>
<?
echo "t:$t";
?>
</p>
<?php
include("conf.inc");
mysql_db_query($db, "UPDATE $tb2 set f='$f',t='$t' where id='1' ");
mysql_close();
?>
</card>
</wml>
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

send4.php

```
<?
header("Content-Type: text/vnd.wap.wml");
echo "<?xml version=\\"1.0\\"?>\n";
echo "<!DOCTYPE wml PUBLIC \\"-//WAPFORUM//DTD WML 1.1//EN\"
\\"http://www.wapforum.org/DTD/wml_1.1.xml\\">\n";
?>
<wml>
<card title="Result Control">
<?php
include("conf.inc");
$result1=mysql_db_query($db,"select * from $tb3 where id='1'");
$result= mysql_fetch_array($result1);
mysql_close();
?>
<?
<p>
echo "v1: $result[v1] ";
</p>
<p>
echo "v2: $result[v2] ";
</p>
<p>
echo "v3: $result[v3] ";
</p>
<p>
echo "p1: $result[p1] ";
</p>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<p>
echo "p2: $result[p2]";
</p>
<p>
echo "pSW1: $result[dsw1] ";
</p>
<p>
echo "dSW2: $result[dsw2] ";
</p>
<p>
echo "h: $result[h] ";
</p>
<p>
echo "t: $result[t] ";
</p>
?>
</card>
</wml>

```



conf.inc

```

<?
$host = "localhost" ;
$db_user = "" ;//
$db_pass = "" ;//
$db = "wap" ; //
$tb="user";//
$tb1="manual";//
$tb2="auto";//

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
$tb3="control";//  
mysql_connect($host, $db_user,$db_pass);  
?>
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข.
Sour code โปรแกรม Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Private Sub Command1_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)

Text45.Text = "1"

Text45.LinkMode = 1

Text45.LinkPoke

End Sub

Private Sub Command1_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

Text45.Text = "0"

Text45.LinkMode = 1

Text45.LinkPoke

End Sub

Private Sub Command2_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)

Text46.Text = "1"

Text46.LinkMode = 1

Text46.LinkPoke

End Sub

Private Sub Command2_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

Text46.Text = "0"

Text46.LinkMode = 1

Text46.LinkPoke

End Sub

Private Sub Form_Load()

If Option21.Value And Option15.Value Then

Frame2.ForeColor = &H0&

Frame3.ForeColor = &H0&

Frame4.ForeColor = &H0&

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Frame5.ForeColor = &H0&
Frame6.ForeColor = &H0&
Frame7.ForeColor = &H0&
Frame1.ForeColor = &H0&

Option3.ForeColor = &HFF&
Option5.ForeColor = &HFF&
Option7.ForeColor = &HFF&
Option9.ForeColor = &HFF&
Option11.ForeColor = &HFF&
Option13.ForeColor = &HFF&
Option1.ForeColor = &HFF&

Option2.ForeColor = &HFF00&
Option4.ForeColor = &HFF00&
Option6.ForeColor = &HFF00&
Option8.ForeColor = &HFF00&
Option10.ForeColor = &HFF00&
Option12.ForeColor = &HFF00&
Option14.ForeColor = &HFF00&
End If

End Sub

Private Sub Text35_Change()

Text35.Text = Text18.Text

End Sub

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Private Sub Timer1_Timer()

If Option1.Value Then

Text15.Text = "1"

Text15.LinkMode = 1

Text15.LinkPoke

Else

Text15.Text = "0"

Text15.LinkMode = 1

Text15.LinkPoke

End If

If Option3.Value Then

Text16.Text = "1"

Text16.LinkMode = 1

Text16.LinkPoke

Else

Text16.Text = "0"

Text16.LinkMode = 1

Text16.LinkPoke

End If

If Option5.Value Then

Text17.Text = "1"

Text17.LinkMode = 1

Text17.LinkPoke

Else

Text17.Text = "0"

Text17.LinkMode = 1

Text17.LinkPoke

End If

If Option7.Value Then



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Text18.Text = "1"  
Text18.LinkMode = 1  
Text18.LinkPoke  
Else  
Text18.Text = "0"  
Text18.LinkMode = 1  
Text18.LinkPoke  
End If  
If Option9.Value Then  
Text19.Text = "1"  
Text19.LinkMode = 1  
Text19.LinkPoke  
Else  
Text19.Text = "0"  
Text19.LinkMode = 1  
Text19.LinkPoke  
End If  
If Option11.Value Then  
Text20.Text = "1"  
Text20.LinkMode = 1  
Text20.LinkPoke  
Else  
Text20.Text = "0"  
Text20.LinkMode = 1  
Text20.LinkPoke  
End If  
If Option13.Value Then  
Text21.Text = "1"  
Text21.LinkMode = 1
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Text21.LinkPoke
Else
Text21.Text = "0"
Text21.LinkMode = 1
Text21.LinkPoke
End If
```

```
If Text28.Text = 1 Then
Text1.Text = "No1.ON"
Text1.BackColor = &HFF00&

Line40.BorderColor = &HFF00&
Shape13.BorderColor = &HFF00&
Else
Text1.Text = "No1.OFF"
Text1.BackColor = &HFF&

Line40.BorderColor = &HFF&
Shape13.BorderColor = &HFF&
End If
```

```
If Text29.Text = 1 Then
Text2.Text = "No2.ON"
Text2.BackColor = &HFF00&

Line41.BorderColor = &HFF00&
Shape15.BorderColor = &HFF00&
Else
Text2.Text = "No2.OFF"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Text2.BackColor = &HFF&

Line41.BorderColor = &HFF&

Shape15.BorderColor = &HFF&

End If

m = Text3.Text

m = Int(m)

n = m - 645

e = n / 47.67

c = e + 0.594 + 8.5

c = Int(c)

f = Int(f)

k = Int(k)

d = c * 1.8

f = d + 32

k = c + 273

Text4.Text = c

Text5.Text = f

Text6.Text = k



i = Text8.Text

j = Text4.Text

If i > j And i <> j Then

Text10.Text = "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Text10.LinkMode = 1

Text10.LinkPoke

Else

Text10.Text = "0"

Text10.LinkMode = 1

Text10.LinkPoke

End If

If i = j Then

Text9.Text = "1"

Text9.LinkMode = 1

Text9.LinkPoke

Else

Text9.Text = "0"

Text9.LinkMode = 1

Text9.LinkPoke

End If

If Option15.Value And Option21.Value Then

Text7.Text = "1"

Text7.LinkMode = 1

Text7.LinkPoke

Else

Text7.Text = "0"

Text7.LinkMode = 1

Text7.LinkPoke

End If



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

If Option18.Value Then

Text11.Text = "1"

Text11.LinkMode = 1

Text11.LinkPoke

Else

Text11.Text = "0"

Text11.LinkMode = 1

Text11.LinkPoke

End If

If Option19.Value Then

Text12.Text = "1"

Text12.LinkMode = 1

Text12.LinkPoke

Else

Text12.Text = "0"

Text12.LinkMode = 1

Text12.LinkPoke

End If

Adodc1.Refresh

Adodc2.Refresh

Adodc1.Recordset.Update

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

If Option20.Value And Option16.Value Then

Text8.Text = Text14.Text

End If

'hi,lo,mid

If Option21.Value And Option16.Value And Option18.Value Then

Text11.Text = "1"

Text11.LinkMode = 1

Text11.LinkPoke

Else

Text11.Text = "0"

Text11.LinkMode = 1

Text11.LinkPoke

End If

If Option21.Value And Option16.Value And Option19.Value Then

Text12.Text = "1"

Text12.LinkMode = 1

Text12.LinkPoke

Else

Text12.Text = "0"

Text12.LinkMode = 1

Text12.LinkPoke

End If

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

If Option20.Value And Option16.Value And Text13.Text = "m" Then

Option18.Value = True

End If

If Option20.Value And Option16.Value And Text13.Text = "l" Then

Option19.Value = True

End If

If Option20.Value And Option16.Value And Text13.Text = "h" Then

Option17.Value = True

End If

'moby manual

'show;,,,,,,,,,,,,,,,,,,,,,

If Text48.Text Then

Text22.Text = "PUMPI. ON"

Text22.BackColor = &HFF00&

Line1.BorderStyle = 1 - solid

Line2.BorderStyle = 1 - solid

Line3.BorderStyle = 1 - solid

Line4.BorderStyle = 1 - solid

Line5.BorderStyle = 1 - solid

Line6.BorderStyle = 1 - solid

Line7.BorderStyle = 1 - solid

Shape4.BorderColor = &HFF00&

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Shape5.BorderColor = &HFF00&

Else

Text22.Text = "PUMP1.OFF"

Text22.BackColor = &HFF&

Line1.BorderStyle = 3 - dot

Line2.BorderStyle = 3 - dot

Line3.BorderStyle = 3 - dot

Line4.BorderStyle = 3 - dot

Line5.BorderStyle = 3 - dot

Line6.BorderStyle = 3 - dot

Line7.BorderStyle = 3 - dot

Shape4.BorderColor = &HFF&

Shape5.BorderColor = &HFF&

End If

If Text49.Text Then

Text23.Text = "PUMP2. ON"

Text23.BackColor = &HFF00&

Line8.BorderStyle = 1 - solid

Line9.BorderStyle = 1 - solid

Line10.BorderStyle = 1 - solid

Line11.BorderStyle = 1 - solid

Line12.BorderStyle = 1 - solid

Line39.BorderStyle = 1 - solid

Line32.BorderStyle = 1 - solid

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Line25.BorderStyle = 1 - solid

Shape6.BorderColor = &HFF00&

Shape7.BorderColor = &HFF00&

Else

Text23.Text = "PUMP2.OFF"

Text23.BackColor = &HFF&

Line8.BorderStyle = 3 - dot

Line9.BorderStyle = 3 - dot

Line10.BorderStyle = 3 - dot

Line11.BorderStyle = 3 - dot

Line12.BorderStyle = 3 - dot

Line39.BorderStyle = 3 - dot

Line32.BorderStyle = 3 - dot

Line25.BorderStyle = 3 - dot

Shape6.BorderColor = &HFF&

Shape7.BorderColor = &HFF&

End If

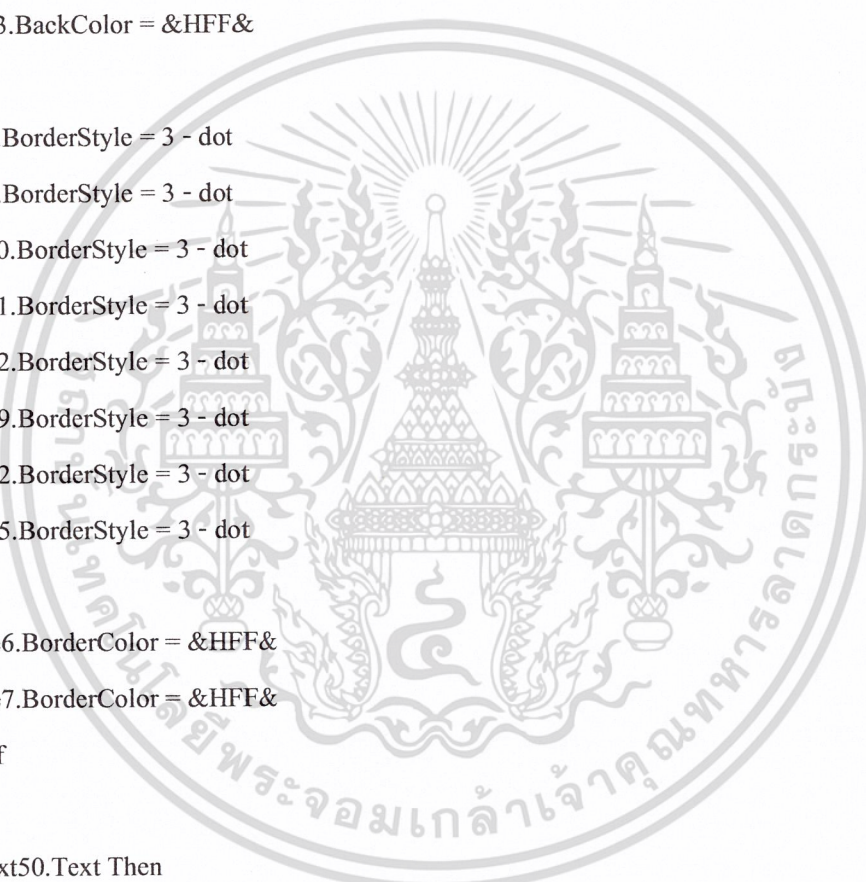
If Text50.Text Then

Text24.Text = "VALE1. ON"

Text24.BackColor = &HFF00&

Line17.BorderStyle = 1 - solid

Line18.BorderStyle = 1 - solid



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Line19.BorderColor = &HFF00&

Line20.BorderColor = &HFF00&

Line21.BorderColor = &HFF00&

Line22.BorderColor = &HFF00&

Line23.BorderColor = &HFF00&

Line24.BorderColor = &HFF00&

Line50.BorderColor = &HFF00&

Line51.BorderColor = &HFF00&

Line45.BorderColor = &HFF00&

Shape16.BorderColor = &HFF00&

Else

Text24.Text = "VALE1.OFF"

Text24.BackColor = &HFF&

Line17.BorderStyle = 3 - dot

Line18.BorderStyle = 3 - dot

Line19.BorderColor = &HFF&

Line20.BorderColor = &HFF&

Line21.BorderColor = &HFF&

Line22.BorderColor = &HFF&

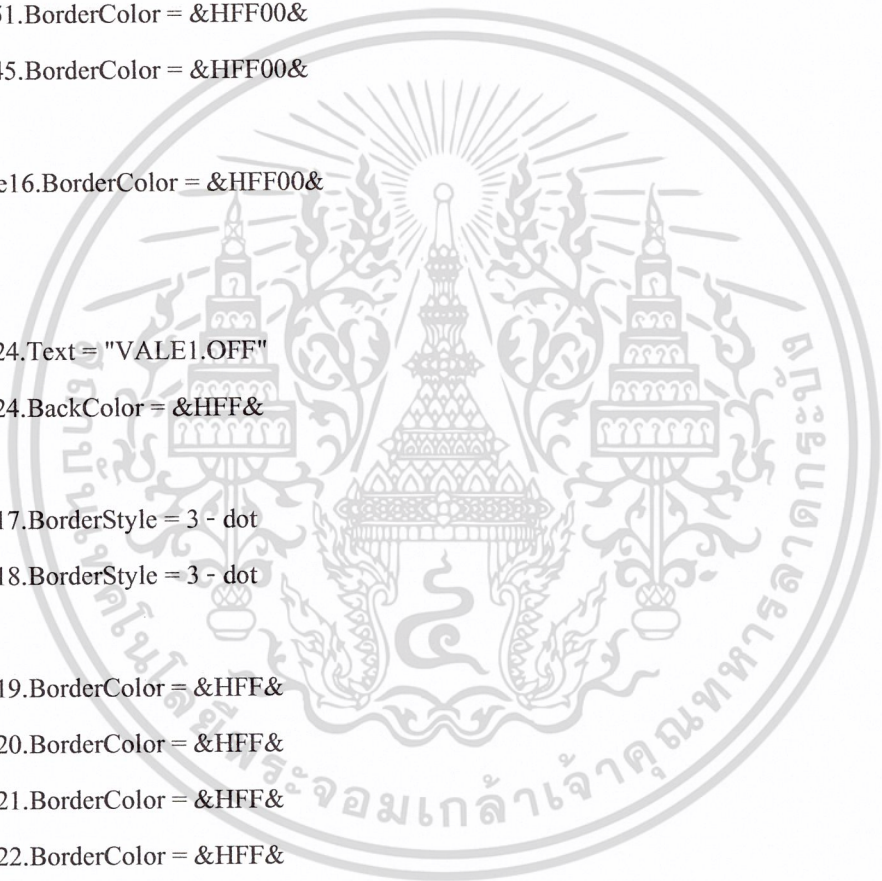
Line23.BorderColor = &HFF&

Line24.BorderColor = &HFF&

Line50.BorderColor = &HFF&

Line51.BorderColor = &HFF&

Line45.BorderColor = &HFF&



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Shape16.BorderColor = &HFF&

End If

If Text51.Text Then

Text25.Text = "VALE2. ON"

Text25.BackColor = &HFF00&

Line38.BorderStyle = 1 - solid

Line14.BorderStyle = 1 - solid

Line13.BorderColor = &HFF00&

Line33.BorderColor = &HFF00&

Line34.BorderColor = &HFF00&

Line35.BorderColor = &HFF00&

Line36.BorderColor = &HFF00&

Line37.BorderColor = &HFF00&

Line48.BorderColor = &HFF00&

Line49.BorderColor = &HFF00&

Line44.BorderColor = &HFF00&

Shape17.BorderColor = &HFF00&

Else

Text25.Text = "VALE2.OFF"

Text25.BackColor = &HFF&

Line38.BorderStyle = 3 - dot

Line14.BorderStyle = 3 - dot

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Line13.BorderColor = &HFF&

Line33.BorderColor = &HFF&

Line34.BorderColor = &HFF&

Line35.BorderColor = &HFF&

Line36.BorderColor = &HFF&

Line37.BorderColor = &HFF&

Line48.BorderColor = &HFF&

Line49.BorderColor = &HFF&

Line44.BorderColor = &HFF&

Shape17.BorderColor = &HFF&

End If

If Text52.Text Then

Text26.Text = "VALE3. ON"

Text26.BackColor = &HFF00&

Line15.BorderStyle = 1 - solid

Line16.BorderStyle = 1 - solid

Line46.BorderColor = &HFF00&

Line47.BorderColor = &HFF00&

Line43.BorderColor = &HFF00&

Line26.BorderColor = &HFF00&

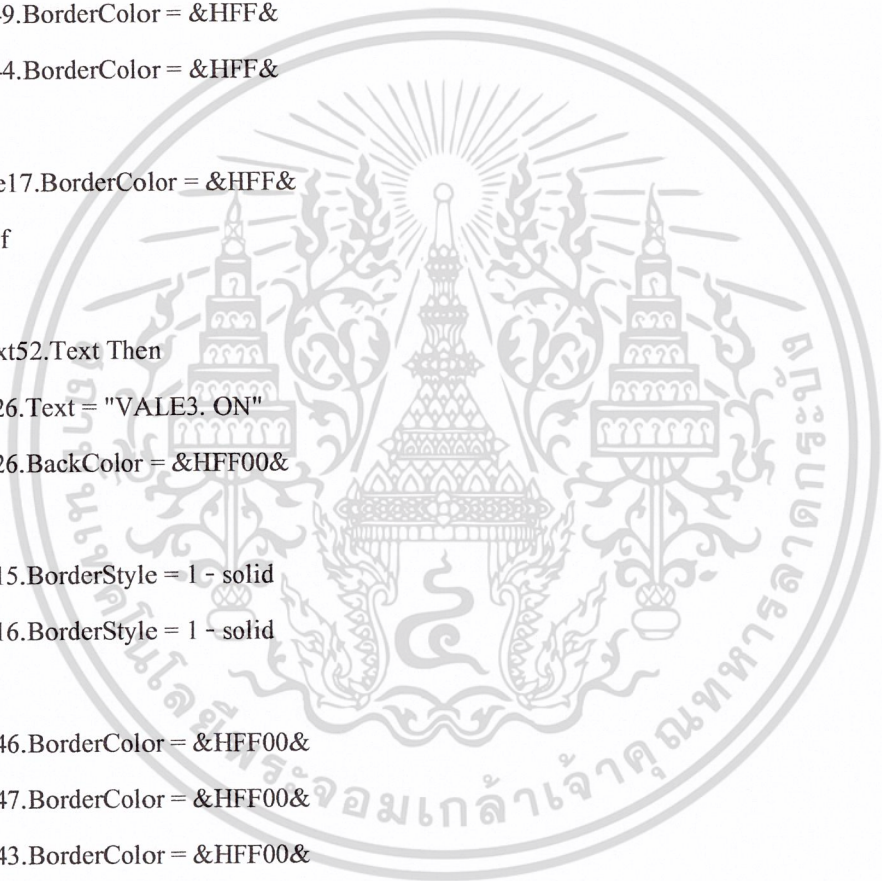
Line27.BorderColor = &HFF00&

Line28.BorderColor = &HFF00&

Line29.BorderColor = &HFF00&

Line30.BorderColor = &HFF00&

Line31.BorderColor = &HFF00&



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Shape10.BorderColor = &HFF00&

Else

Text26.Text = "VALE3.OFF"

Text26.BackColor = &HFF&

Line15.BorderStyle = 3 - dot

Line16.BorderStyle = 3 - dot

Line46.BorderColor = &HFF&

Line47.BorderColor = &HFF&

Line43.BorderColor = &HFF&

Line26.BorderColor = &HFF&

Line27.BorderColor = &HFF&

Line28.BorderColor = &HFF&

Line29.BorderColor = &HFF&

Line30.BorderColor = &HFF&

Line31.BorderColor = &HFF&

Shape10.BorderColor = &HFF&

End If

If Text53.Text Then

Text27.Text = "HEATER. ON"

Text27.BackColor = &HFF00&

Shape9.BorderColor = &HFF00&

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Else

Text27.Text = "HEATER.OFF"

Text27.BackColor = &HFF&

Shape9.BorderColor = &HFF&

End If

'animation

If Option21.Value And Option15.Value Then

Frame2.ForeColor = &H0&

Frame3.ForeColor = &H0&

Frame4.ForeColor = &H0&

Frame5.ForeColor = &H0&

Frame6.ForeColor = &H0&

Frame7.ForeColor = &H0&

Frame1.ForeColor = &H0&

Option3.ForeColor = &HFF00&

Option5.ForeColor = &HFF00&

Option7.ForeColor = &HFF00&

Option9.ForeColor = &HFF00&

Option11.ForeColor = &HFF00&

Option13.ForeColor = &HFF00&

Option1.ForeColor = &HFF00&

Option2.ForeColor = &HFF&

Option4.ForeColor = &HFF&

Option6.ForeColor = &HFF&

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Option8.ForeColor = &HFF&
Option10.ForeColor = &HFF&
Option12.ForeColor = &HFF&
Option14.ForeColor = &HFF&
Else
  Frame2.ForeColor = &H800000B
  Frame3.ForeColor = &H800000B
  Frame4.ForeColor = &H800000B
  Frame5.ForeColor = &H800000B
  Frame6.ForeColor = &H800000B
  Frame7.ForeColor = &H800000B
  Frame1.ForeColor = &H800000B

  Option3.ForeColor = &H800000B
  Option5.ForeColor = &H800000B
  Option7.ForeColor = &H800000B
  Option9.ForeColor = &H800000B
  Option11.ForeColor = &H800000B
  Option13.ForeColor = &H800000B
  Option1.ForeColor = &H800000B

  Option2.ForeColor = &H800000B
  Option4.ForeColor = &H800000B
  Option6.ForeColor = &H800000B
  Option8.ForeColor = &H800000B
  Option10.ForeColor = &H800000B
  Option12.ForeColor = &H800000B
  Option14.ForeColor = &H800000B
End If
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

If Option21.Value And Option16.Value Or Option20.Value Then

Frame12.ForeColor = &H0&

Frame13.ForeColor = &H0&

Frame14.ForeColor = &H0&

Option17.ForeColor = &H0&

Option18.ForeColor = &H0&

Option19.ForeColor = &H0&

Text8.BackColor = &HFFFF&

Else

Frame12.ForeColor = &H800000B

Frame13.ForeColor = &H800000B

Frame14.ForeColor = &H800000B

Option17.ForeColor = &H800000B

Option18.ForeColor = &H800000B

Option19.ForeColor = &H800000B

Text8.BackColor = &HFFFFFF

End If

If Text8.Text > Text4.Text Then

Text4.BackColor = &HFFFF&

Else

Text4.BackColor = &HC0FFF

End If End Sub

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ค.

Data Sheet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



5

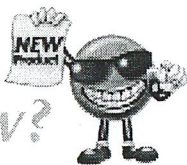
Technical Overview

- What's New
- DB15 Family of Products
- Features and Specifications
- Choosing I/O
- Dimensions and Installation Requirements
- I/O Specifications
- Option Cards
- High-Speed I/O Features
- Appendix
- Technical Support

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



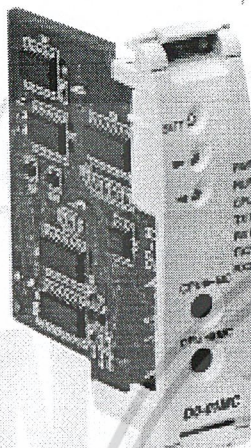
What's New?



Memory cartridge

D0-05MC:

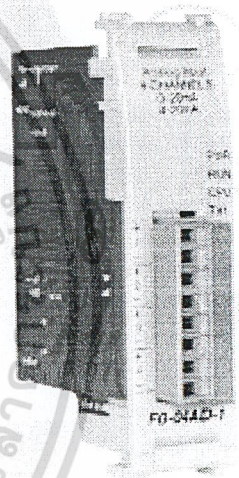
Removable memory cartridge with battery backed CMOS RAM, external upload/download controls and an embedded real-time clock/calendar.



Analog input module

F0-04AD-1:

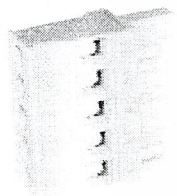
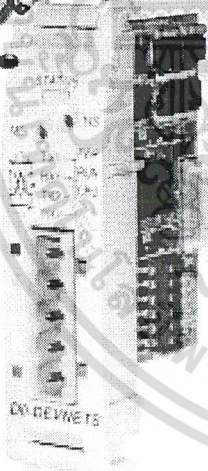
4-channel analog input module, 12-bit resolution, range: 0-20mA or 4-20mA.



DeviceNet module

D0-DEVNETS:

DeviceNet DL05 communications card. Slave for DeviceNet connection to your DeviceNet master.



DC-powered DL05s

D0-05DD-D and D0-05DR-D:

12/24VDC Power Supply DL05 CPU, 8 DC in/6DC or Relay out, 6K words total (2048 words ladder - Flash, 4096 words V-memory), RLL/RLL^{PLUS} programming, built-in RS232C programming port and additional RS232C communication port.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
Section 1 — DL05 Toll-free orders 1-800-633-0405 www.automationdirect.com
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



The DL05 Family of Products

The DL05 Micro PLC family includes eight different versions. All have the same appearance and CPU performance. Two are DC-powered. The CPU offers the same instruction set as our popular DL240 CPU, plus several more instructions specifically designed for machine control applications. All DL05 PLC's have two RS-232C communications ports and one option card slot. Units with DC inputs have selectable high-speed input features on three input points. Units with DC outputs can use the first two outputs as a single bi-directional pulse output. All DL05 Micro PLCs offer a large amount of program memory, a substantial instruction set and advanced diagnostics.

Since the DL05 products are fixed I/O type PLCs, your primary task is to select the I/O that is needed for your application. A brief overview of the available DL05 I/O is located in the chart to the right. Detailed specifications for each version of the DL05 PLC are located later in this section.

Specification	AC Powered Units	DC Powered Units
Power	110/220VAC	12/24VDC
Input Voltage Range	95-240VAC	12-24VDC
Maximum Power	30VA	20W
Maximum Inrush Current	13A, 1ms (240VAC)	10A < 1ms
Storage Temperature	-4°F to 158°F (-20°C to 70°C)	
Ambient Operating Temperature	32°F to 131°F (0°C to 55°C)	
Ambient Humidity	5% - 95% relative humidity (non-condensing)	
Vibration Resistance	MIL STD 810C, Method 514.2	
Shock Resistance	MIL STD 810C, Method 516.2	
Noise Immunity	NEMA (ICS3-304)	
Atmosphere	No corrosive gases	

The DL05 Family

AC Powered Versions

D0-05AA

8 AC inputs
6 AC outputs, 0.5A/point

D0-05AD

8 AC inputs
6 DC outputs, 1.0A/point
Two outputs can be used as a single bi-directional 7kHz Pulse Output

D0-05AR

8 AC inputs
6 Relay outputs, 2A/point

D0-05DA

8 DC inputs
Three inputs are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
6 AC outputs, 0.5A/point

D0-05DD

8 DC inputs
Three inputs are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
6 DC outputs, 1.0A/point
Two outputs can be used as a single bi-directional 7kHz Pulse Output

D0-05DR

8 DC inputs
Three inputs are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
6 Relay outputs, 2A/point

DC Powered Versions

D0-05DD-D

8 DC inputs
Three inputs are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
6 DC outputs, 1.0A/point
Two outputs can be used as a single bi-directional 7kHz Pulse Output

D0-05DR-D

8 DC inputs
Three inputs are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
6 Relay outputs, 2A/point

Option Cards

D0-01MC

Memory cartridge, 32K battery backed CMOS RAM, external upload/download controls, real time clock/calendar.

F0-04AD-1

4 channel analog input module, 12-bit resolution, range: 0-20mA or 4-20mA; Designed to operate with a 24 VDC user supplied power supply

D0-DEVNETS

DeviceNet slave controller for a DeviceNet connection from a DL05 to a DeviceNet master

Note: High speed modes may only be used one at a time. You cannot use the high-speed counter and the pulse output features at the same time.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า



Choosing the Type of I/O

The DL05 product family offers several different combinations of I/O points. Choose the I/O points that are right for your application.

Fixed discrete I/O

All DL05 Micro PLCs have 8 inputs and 6 outputs, regardless of the actual type of points on the unit (DC in/Relay out, DC in/DC out, etc.)

Option card slot

The DL05 has an option card slot that allows you to add a variety of features to your PLC. Detailed specifications for the option cards are available later in this section.

Automatically assigned addresses

The DL05 uses automatic addressing, so for the vast majority of applications, there is no setup required. We use octal addressing for our products, which means there are no 8's or 9's. The 8 input points use addresses X0-X7. The 6 output points use addresses Y0-Y5.

Review the I/O specs and wiring diagrams

The I/O Options Table gives a brief description of the I/O combinations offered for the DL05 PLCs. The I/O specifications discussed in more detail later in this section. Make sure the I/O specifications are within the range required by your application.

I/O Options Table						
Part Number	INPUTS			OUTPUTS		
	I/O Type/ Commons	Sink or Source	Voltage Ranges	I/O Type/ Commons	Sink or Source	Voltage/Current Ratings
D0-05AR	AC/2	N/A	90-120VAC	Relay/2	Sink or Source	6-27VDC, 2A 6-240VAC, 2A
D0-05DR	DC/2	Sink or Source	12-24VDC	Relay/2	Sink or Source	6-27VDC, 2A 6-240VAC, 2A
D0-05AD	AC/2	N/A	90-120VAC	DC/1	Sink	6-27VDC, 0.5A (Y0-Y2) 6-27VDC, 1.0A (Y3-Y5)
D0-05DD	DC/2	Sink or Source	12-24VDC	DC/1	Sink	6-27VDC, 0.5A (Y0-Y2) 6-27VDC, 1.0A (Y3-Y5)
D0-05AA	AC/2	N/A	90-120VAC	AC/2	N/A	17-240VAC 47-63Hz 0.5A
D0-05DA	DC/2	Sink or Source	12-24VDC	AC/2	N/A	17-240VAC 47-63Hz 0.5A
D0-05DR-D	DC/2	Sink or Source	12-24VDC	Relay/2	Sink or Source	6-27VDC, 2A 6-240VAC, 2A
D0-05DD-D	DC/2	Sink or Source	12-24VDC	DC/1	Sink	6-27VDC, 0.5A (Y0-Y2) 6-27VDC, 1.0A (Y3-Y5)

Sinking/sourcing

If you are using a DC field device, you should consider whether the device requires a sinking or sourcing configuration on the PLC I/O. For more information on sinking and sourcing concepts, please refer to the appendix section of this catalog.

Sink/Source Inputs — all DC inputs on the DL05 Micro PLCs can be wired in a sinking or sourcing configuration. However, all inputs on a single common must use the same configuration.

Sinking Outputs — all DC outputs on the DL05 Micro PLCs must be wired in a sinking configuration. If a sourcing output is required, consider using a unit with relay outputs.

Special high-speed counting and pulse output features

Selected DL05 units offer special high-speed input features (DC input units) or pulse output features (DC output units). The first three inputs are automatically set up as filtered inputs with a 10 ms filter. By entering a setup code in a special V-memory location, you can choose other features. In some modes of operation, you have a choice as to how you use each point. For example, if you use X0 as an Up Counter, you can choose to use X2 as a reset input for the counter, or as a filtered discrete input. If these features are of interest, you should review the detailed high-speed I/O descriptions found later in this section.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า



Feature and Specifications

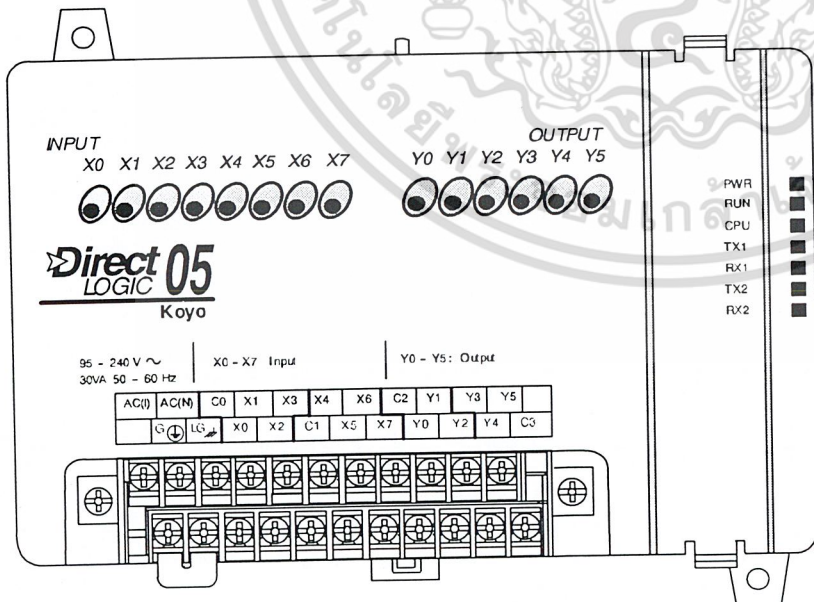
The DL05 Micro PLCs contain the CPU, power supply, and I/O all in the same housing. If you examine the CPU Specifications table, you'll see that we included many features found in our modular CPUs.

Reviewing the specs

Make sure these features can satisfy the requirements of your application. Detailed specifications for each version of the DL05 PLC are located later in this section.

System capacity

System capacity is the ability to accommodate a variety of applications. For ladder memory, most boolean instructions require one word. Some other instructions, such as timers, counters, etc. require two or more words. Our V-memory locations are 16-bit words and are useful for data storage, etc.



Performance

Performance is simply the scan time, which is the amount of time required to read the inputs, solve the RLL program and update the outputs.

Instructions and diagnostics

Make sure the unit offers the instructions needed. A complete list of instructions is available at the end of this section.

Communications

The DL05 offers powerful communication features normally found only on more expensive PLCs. Check to see if the available features fulfill your specific needs.

Specialty features

For the DC input and/or DC output versions, we also offer several high-speed I/O features.

DL05 CPU Specifications

System capacity

Total memory available (words)	6K
Ladder memory (words)	2,048
V-memory (words)	4,096
User V	3,968
Non-volatile User V	128
Battery backup	Yes ¹
Total I/O	14
Inputs	8
Outputs	6
I/O expansion	Yes ¹

Performance

Contact execution (Boolean)	2.0µs
Typical scan (1K Boolean) ²	2.7-3.2ms

Instructions and diagnostics

RLL ladder style	Yes
RLLPLUS/flowchart style (Stages)	Yes/256
Run-time editing	Yes
Scan	Variable/fixd
Instructions	129
Control relays	512
Timers	128
Counters	128
Immediate I/O	Yes
Subroutines	Yes
For/next loops	Yes
Timed interrupt	Yes
Integer math	Yes
Floating-point math	No
PID	Yes ¹
Drum sequencers	Yes
Bit of word	No
ASCII print	Yes
Real-time clock/calendar	Yes ¹
Internal diagnostics	Yes
Password security	Yes
System and user error log	No

Communications

Built-in ports	Two RS-232C
K-sequence (proprietary protocol)	Yes
DirectNet master/slave	Yes
MODBUS RTU master/slave	Yes
ASCII out	Yes
Baud rate	
Port 1	9,600 baud (fixed)
Port 2	selectable 300-38,400 baud (default 9,600)

Specialty Features

Filtered inputs	Yes ¹
Interrupt input	Yes ¹
High speed counter	Yes, 5kHz ¹
Pulse output	Yes, 7kHz ¹
Pulse catch input	Yes ¹

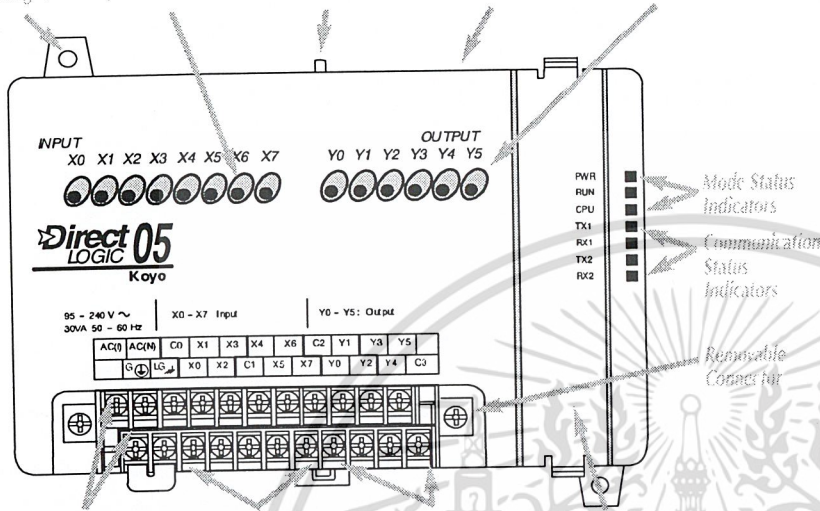
- These features are available with use of certain option cards. Option card specifications are located later in the DL05 section.
- Our 1K program includes contacts, coils, and scan overhead. If you compare our products to others, make sure you include their scan overhead.
- Input features only available on units with DC inputs. Output features only available on units with DC outputs.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

DL05 Hardware Features

Hardware features diagram

Mounting Tab Input Status Indicators Mode Switch Communication Ports Output Status Indicators



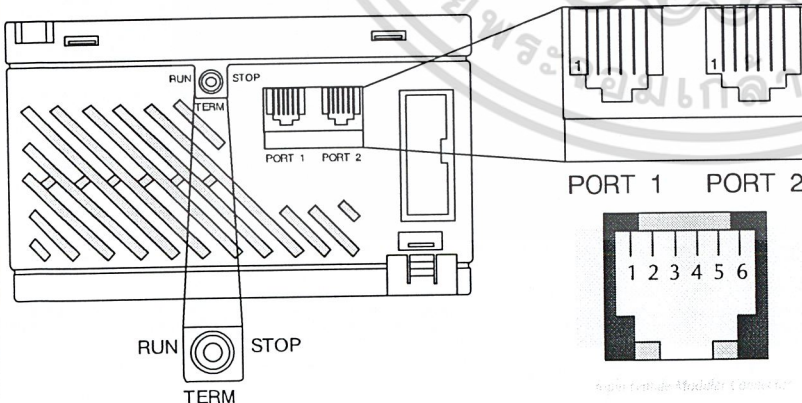
External Power Inputs Discrete Input Terminals Discrete Output Terminals Option Card Slot

Two communication ports!

Port 1 Pin Descriptions		
1	0V	Power (-) connection (GND)
2	5V	Power (+) connection
3	RXD	Receive data (RS-232C)
4	TXD	Transmit data (RS-232C)
5	5V	Power (+) connection
6	0V	Power (-) connection (GND)

Port 2 Pin Descriptions		
1	0V	Power (-) connection (GND)
2	5V	Power (+) connection
3	RXD	Receive data (RS-232C)
4	TXD	Transmit data (RS-232C)
5	RTS	Request to send
6	0V	Power (-) connection (GND)

Fixed Flash Memory: The DL05 units offer built-in Flash memory. However, memory cannot be removed or changed like the D2-230 and D2-240 CPUs.



Mode Switch Position	CPU Action
RUN (Run Program)	CPU is forced into the RUN mode if no errors are encountered. No changes are allowed by the programming/monitoring device.
TERM (Terminal)	RUN PROGRAM and the TEST modes are available. Mode and program changes are allowed by the programming/monitoring device.
STOP	CPU is forced into the STOP mode. No changes are allowed by the programming/monitoring device.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า



DL05 Communication Features

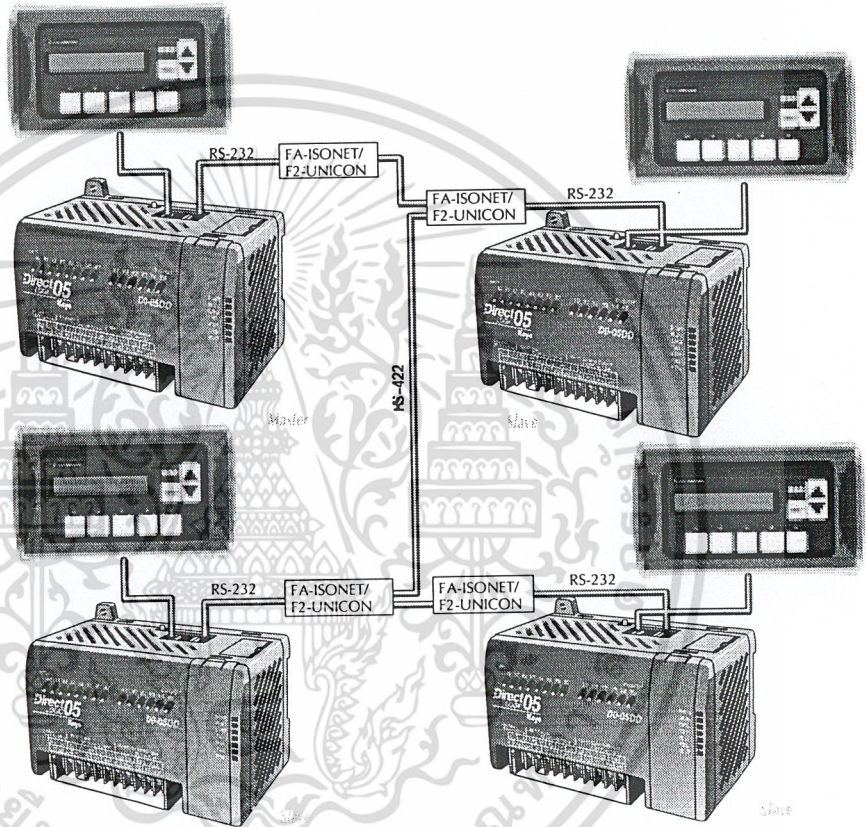
All DL05 units offer two 6-pin, RS-232C ports. Each of these ports can be connected to a handheld programmer, *DirectSOFT* software, operator interfaces, etc. Port 1 is fixed at 9,600 baud and port 2 has selective baud rates from 300-38,400 baud.

Networking the DL05

Protocols supported

Each port is capable of communicating with K-sequence, *DirectNet*, and MODBUS protocols. Port 1 can only be a slave for each of the protocols. Port 2 can serve as a K-sequence slave or a network master or slave for either *DirectNET* or MODBUS RTU protocols.

An option card is available that will allow you to connect your DL05 to a DeviceNet network. See information on option cards later in this section.



Communication Port 1

- Connects to: HPP, *DirectSOFT*, operator interfaces, etc.
- 6-pin, RS-232C
- 9,600 Baud (fixed)
- Parity - odd (fixed)
- Station address 1 (fixed)
- 8 data bits
- 1 start, 1 stop bit
- Asynchronous, Half-duplex, DTE
- Protocol: (Auto-Select)
 - K sequence (Slave only)
 - DirectNET* (Slave only)
 - MODBUS (Slave only)

Communication Port 2

- Connects to: HPP, *DirectSOFT*, operator interfaces, etc.
- 6-pin, RS-232C
- Communication speed (baud)
 - 300, 600, 1,200, 2,400, 4,800, 9,600 (default), 19,200, and 38,400
- Parity-odd (default), even, none
- Station address 1 (default),
 - 1-90 *DirectNet*/K-sequence,
 - 1-247 MODBUS RTU
- 8 data bits
- 1 start, 1 stop bit
- Asynchronous, Half-duplex, DTE
- Protocol:
 - K sequence (Slave only)
 - DirectNET* (Master/Slave)
 - MODBUS (Master/Slave)
 - Non-sequence/Print

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
Section 1 — DL05 Toll-free orders 1-800-633-0405 www.automationdirect.com
แม้ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Product Dimensions and Installation

It is important to understand the installation requirements for your DL05 system. This will help ensure that the DL05 products operate within their environmental and electrical limits.

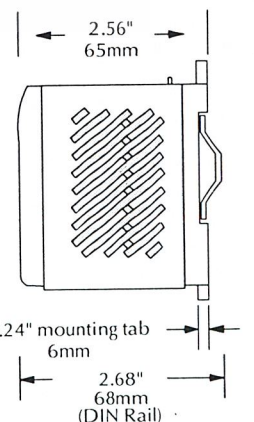
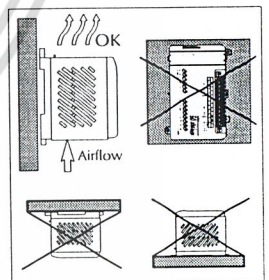
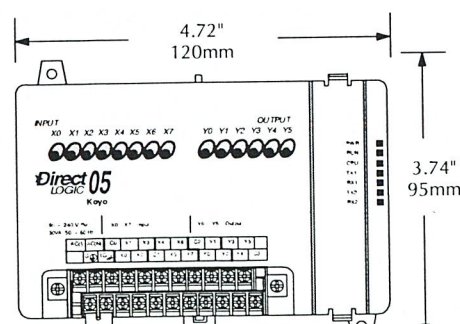
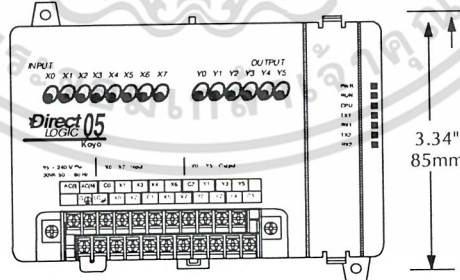
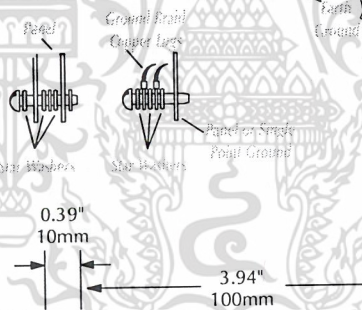
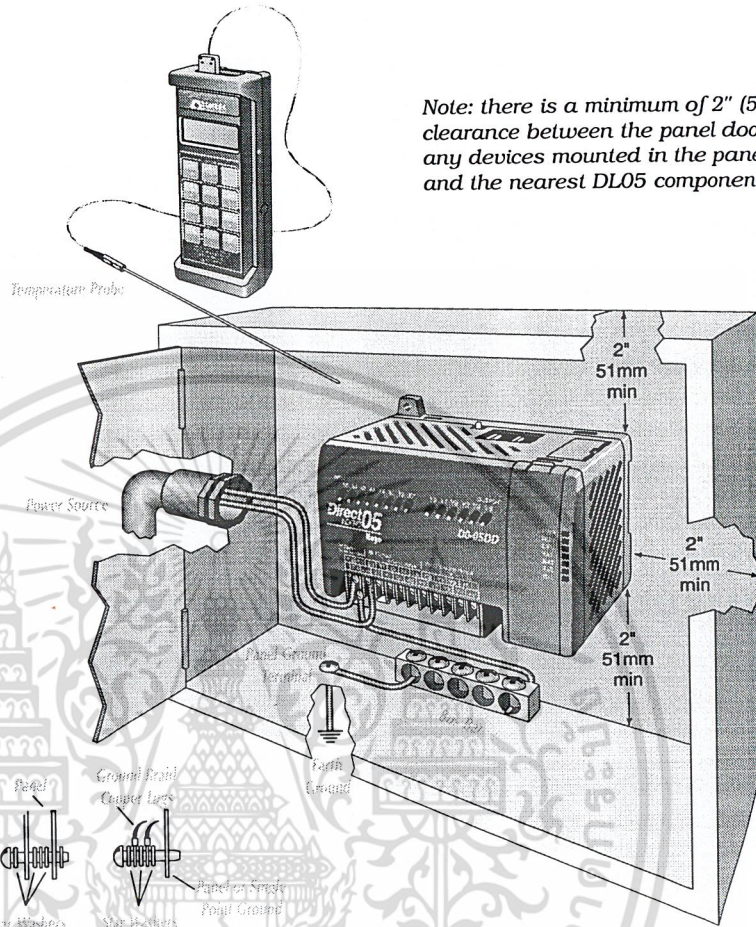
Plan for safety

This catalog should never be used as a replacement for the user manual. The user manual, DO-USER-M, contains important safety information that must be followed. The system installation should comply with all appropriate electrical codes and standards.

Unit dimensions and mounting orientation

DL05 units must be mounted properly to ensure ample airflow for cooling purposes. It is important to check these dimensions against the conditions required for your application. For example, it is recommended that you leave 2" depth for ease of access and cable clearance; however, your distance may be greater or less. Also, check the installation guidelines for the recommended cabinet clearances.

Note: there is a minimum of 2" (51mm) clearance between the panel door or any devices mounted in the panel door and the nearest DL05 component.



Environmental Specifications	
Storage temperature	-4° F-158°F (-20°C to 70°C)
Ambient operating temperature	32°F-131°F (0° to 55°C)
Ambient humidity	5 to 95% relative humidity (non-condensing)
Vibration resistance	MIL STD 810C Method 514.2
Shock resistance	MIL STD 810C Method 516.2
Noise immunity	NEMA (ICS3-304)
Atmosphere	No corrosive gases

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 Section 1 — DL05 Toll-free orders 1-800-633-0405 www.automationdirect.com
 เมื่อกำหนดใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำมาใช้

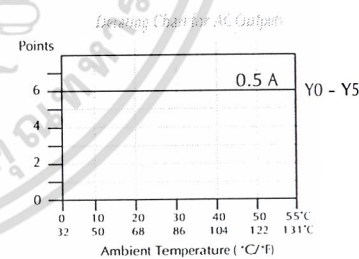
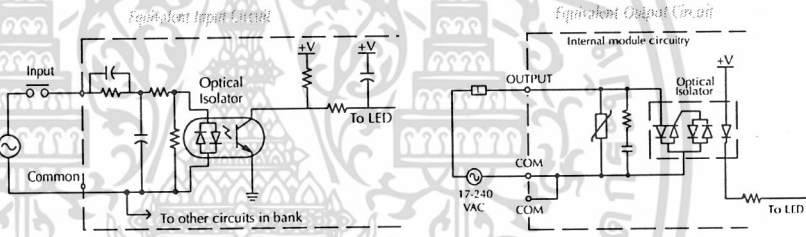
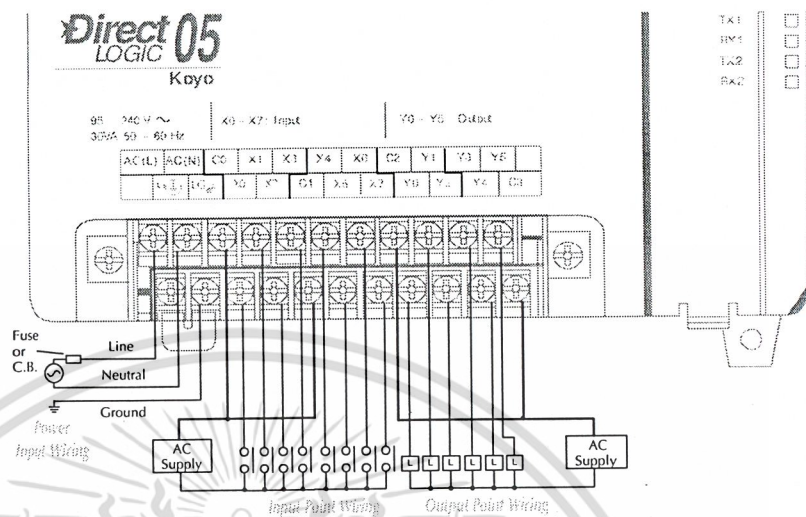
DL05 I/O Specifications

D0-05AA wiring diagram and specifications

AC Power Supply specifications
Voltage Range95-240VAC (30VA)

AC Input specifications
Number of input points8
Number of commons2 (isolated)
Input voltage range90-120VAC
Frequency range47-63Hz
Input current8mA @ 100 VAC at 50 Hz
.....10mA @ 100VAC at 60z
ON current/voltage level>6mA/75VAC
OFF current/voltage level<2mA/20VAC
OFF to ON response<40ms
ON to OFF response<40ms
FusesNone

AC Output specifications
Number of output points6
Number of commons2 (isolated)
Output voltage range17-240VAC
.....47-63Hz
Peak voltage264VAC
ON voltage drop1.5 VAC>50mA
.....4.0VAC<50mA
Maximum current0.5A/point
.....1.5A/common
Maximum leakage current4mA at 264VAC
Maximum inrush current10A for 10ms
Minimum load10mA
OFF to ON response1ms
ON to OFF response1ms + 1/2 cycle
FusesNone (external recommended)





DL05 I/O Specifications

D0-05AD wiring diagram and specifications

AC Power Supply specifications

Voltage range95-240VAC (30VA)

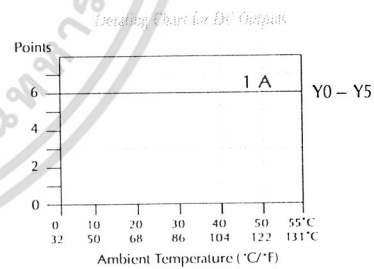
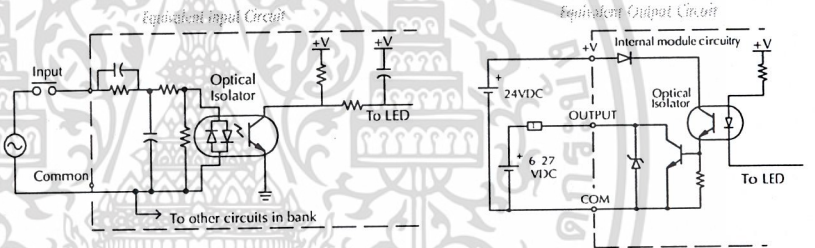
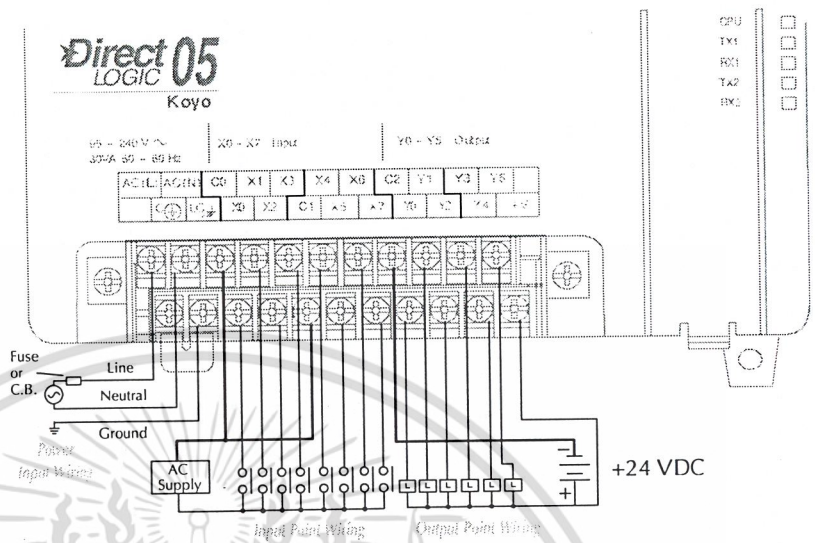
AC Input specifications

Number of input points8
 Number of commons2 (isolated)
 Input voltage range90-120VAC
 Frequency range47-63Hz
 Input current8mA @ 100 VAC at 50 Hz
10mA @ 100 VAC at 60 Hz
 ON current/voltage level>6mA/75 VAC
 OFF current/voltage level<2mA/20 VAC
 OFF to ON response<40ms
 ON to OFF response<40ms
 FusesNone

DC Output specifications

Number of output points6 (sinking)
 Number of commons1
 Output voltage range6-27VDC
 Peak voltage50VDC
 Max. frequency (Y0,Y1)7kHz
 ON voltage drop0.3VDC @ 1A
 Maximum current0.5A/point (Y0-Y1)*
1.0A point (Y2-Y5)
 Maximum leakage current15µA @ 30 VDC
 Maximum inrush current2A for 100ms
 OFF to ON response<10µs
 ON to OFF response<30µs
 External DC power required20-28VDC
150mA max.
 Status indicatorsLogic side
 FusesNone (external recommended)

* When output points Y0 and Y1 are not used in pulse mode, the maximum output current is 1.0A.



DL05 I/O Specifications



D0-05AR wiring diagram and specifications

AC Power Supply specifications

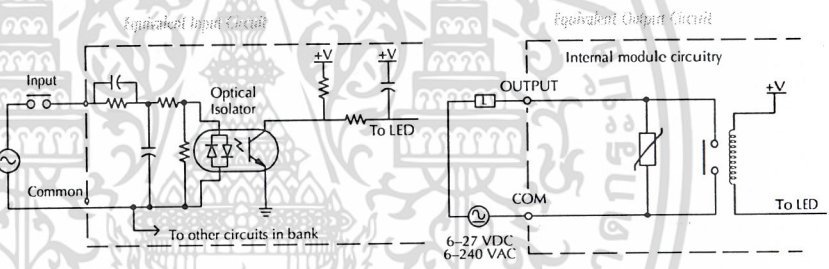
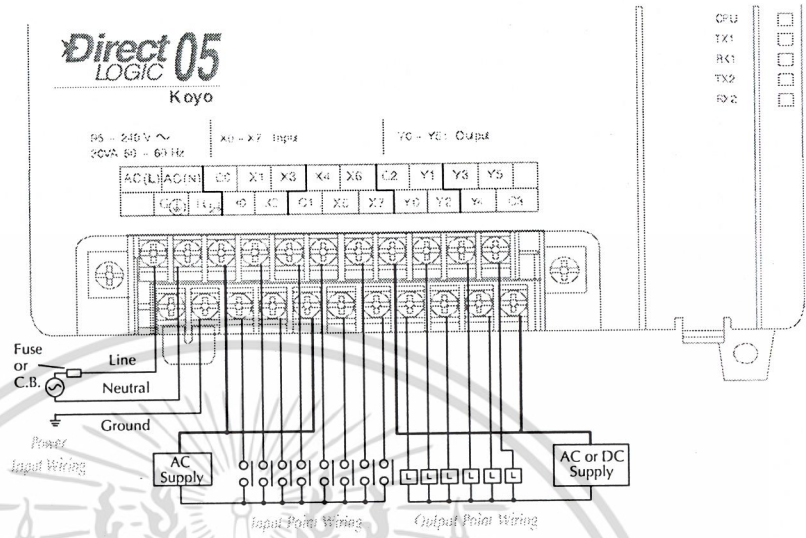
Voltage range95-240VAC (30VA)

AC Input specifications

Number of input points8
 Number of commons2 (isolated)
 Input voltage range90-120VAC
 Frequency range47- 63Hz
 Input current8mA @ 100VAC at 50Hz
10mA @ 100VAC at 60 Hz
 OFF current/voltage level>6mA/75VAC
 OFF current/voltage level<2mA/20VAC
 OFF to ON response<40ms
 ON to OFF response<40ms
 FusesNone

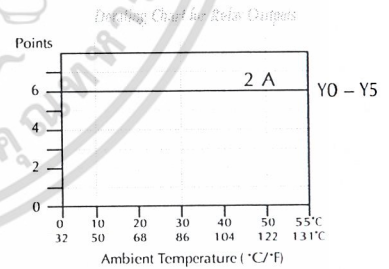
Relay Output specifications

Number of output points6
 Number of commons2 (isolated)
 Output voltage range6-240VAC, 47-63Hz
6-27VDC
 Maximum voltage264VAC, 30VDC
 Maximum Current2A/point
6A/common
 Max leakage current0.1mA @ 264 VAC
 Smallest recommended load5mA @ 5VDC
 OFF to ON response<15ms
 ON to OFF response<10ms
 Status indicatorsLogic Side
 FusesNone (external recommended)



Typical Relay Life (Operations) at Room Temperature

Voltage and Type of Load	Load Current	
	1A	2A
24 VDC Resistive	600K	270K
24 VDC Solenoid	150K	60K
110 VAC Resistive	900K	350K
110 VAC Solenoid	350K	150K
220 VAC Resistive	600K	250K
220 VAC Solenoid	200K	100K



DL05 I/O Specifications

D0-05DD wiring diagram and specifications

AC Power Supply specifications

Voltage range95-240VAC (30VA)

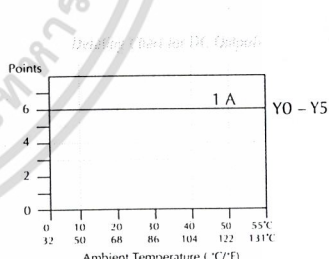
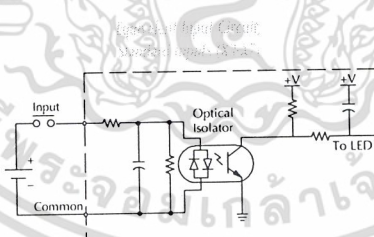
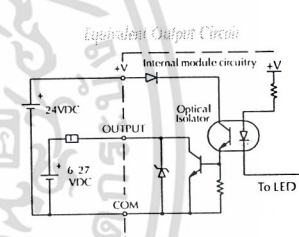
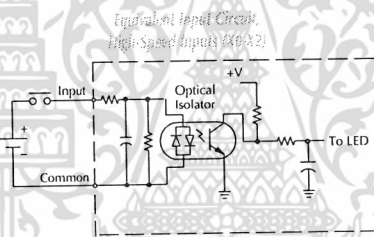
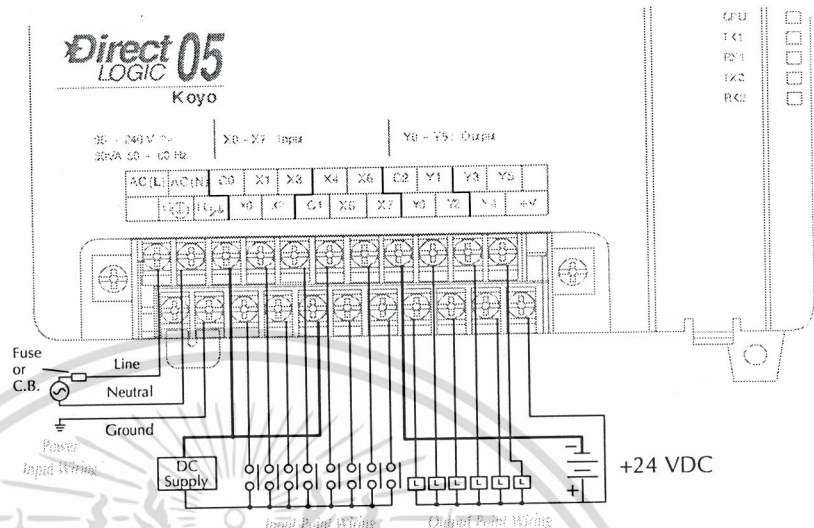
DC Input specifications

Number of input points8 (sink/source)
 Number of commons2 (isolated)
 Input voltage range12-24VDC
 Input impedance(X0-X2) 1.8K @ 12-24VDC
(X3-X7) 2.8K @ 12-24VDC
 ON current/voltage level>0.5mA/<10VDC
 OFF current/voltage level>0.5mA/<2VDC
 Response timeX0-X2 X3-X7
 OFF to ON<100µs <8ms
 ON to OFF<100µs <8ms
 FusesNone

DC Output specifications

Number of output points6 (sinking)
 Number of commons1
 Output voltage range6-27VDC
 Peak voltage50VDC
 Max.frequency (Y0,Y1)7kHz
 ON voltage drop0.3VDC @ 1A
 Maximum current0.5A/point (Y0-Y1)*
1.0A point (Y2-Y5)
 Maximum leakage current15µA @ 30 VDC
 Maximum inrush current2A for 100ms
 OFF to ON response<10µs
 On to OFF response<30µs
 External DC power required20-28VDC
150mA max.
 Status indicatorsLogic side
 FusesNone (external recommended)

* When output points Y0 and Y1 are not used in pulse mode, the maximum output current is 1.0A.





DL05 I/O Specifications

D0-05DR wiring diagram and specifications

AC Power Supply specifications

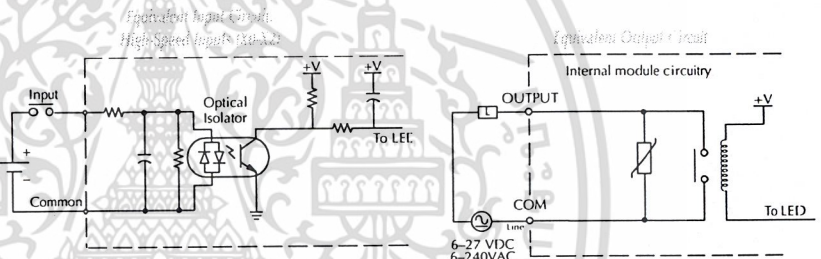
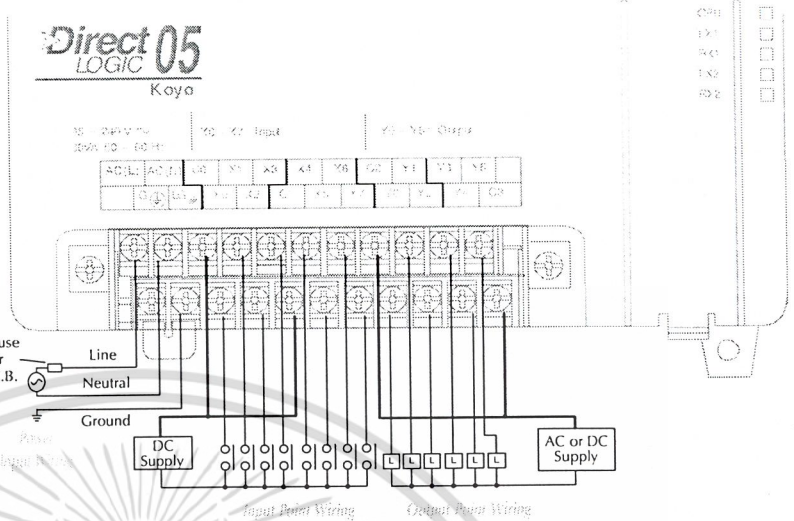
Voltage range95-240VAC (30VA)

DC Input specifications

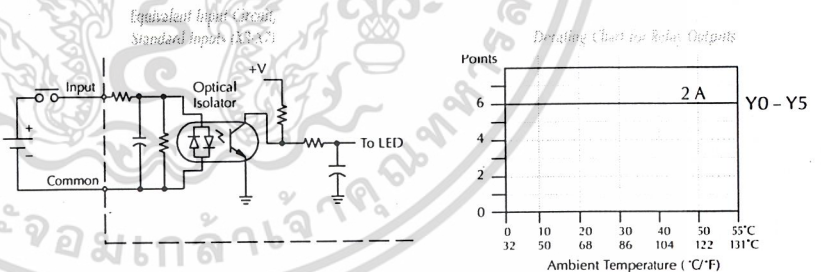
Number of input points8 (sink/source)
 Number of commons2 (isolated)
 Input voltage range12-24VDC
 Input impedance(X0-X2) 1.8K @ 12-24VDC
(X3-X7) 2.8K @ 12-24VDC
 ON current/voltage level>5mA/>10VDC
 OFF current/voltage level<0.5mA/<2VDC
 Response timeX0-X2 X3-X7
 OFF to ON<100µs <8ms
 ON to OFF<100µs <8ms
 FusesNone

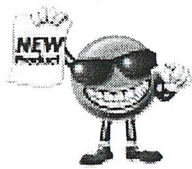
Relay Output specifications

Number of output points6
 Number of commons2 (isolated)
 Output voltage range6-240VAC, 47-63Hz
6-27VDC
 Maximum voltage264VAC, 30VDC
 Maximum Output Current2A/point
 Max leakage current0.1mA @ 264VAC
 Smallest Recommended Load5mA @ 5VDC
 OFF to ON response<15ms
 ON to OFF response<10ms
 Status indicatorsLogic Side
 FusesNone (external recommended)



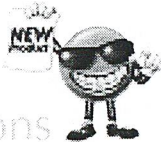
Typical Relay Life (Operations) at Room Temperature		
Voltage and Type of Load	Load Current	
	1A	2A
24 VDC Resistive	600K	270K
24 VDC Solenoid	150K	60K
110 VAC Resistive	900K	350K
110 VAC Solenoid	350K	150K
220 VAC Resistive	600K	250K
220 VAC Solenoid	200K	100K





DL05 I/O Specifications

DL05DR-D Wiring diagram and specifications



DC Power Supply Specifications

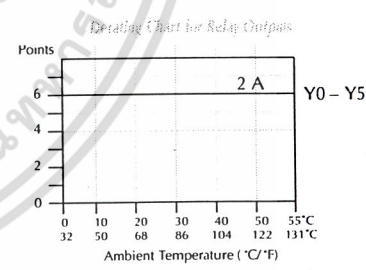
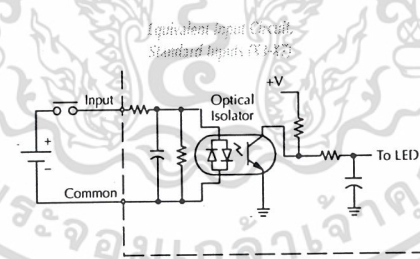
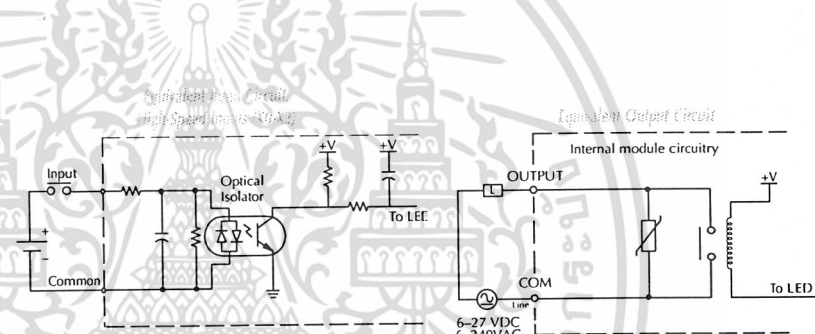
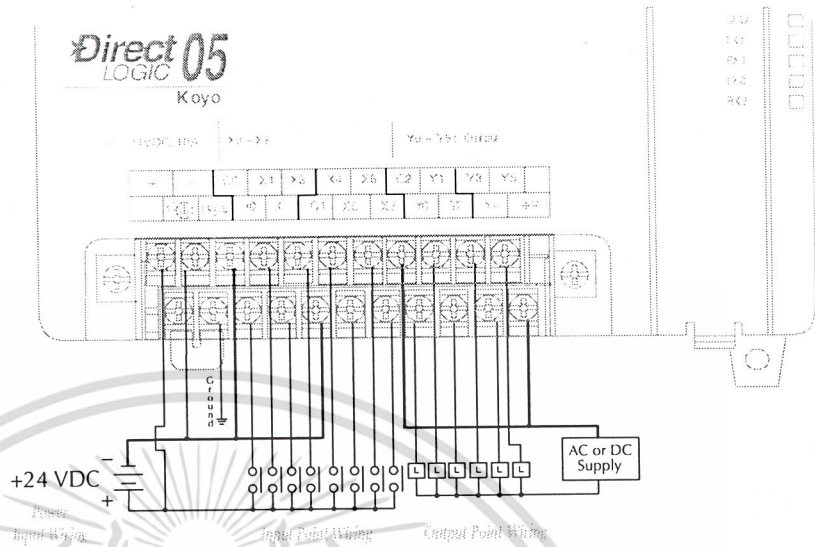
Voltage range..... 12-24VDC
Power..... 20W max.

DC Input specifications

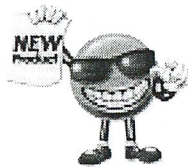
Number of input points 8 (sink/source)
Number of commons 2 (isolated)
Input voltage range 12-24VDC
Input impedance (X0-X2) 1.8K @ 12-24VDC
..... (X3-X7) 2.8K @ 12-24VDC
ON current/voltage level >5mA/>10VDC
OFF current/voltage level <0.5mA/<2VDC
Response time X0-X2 <100µs <8ms
X3-X7 <100µs <8ms
ON to OFF <100µs <8ms
Fuses None

Relay Output specifications

Number of output points 6
Number of commons 2 (isolated)
Output voltage range 6-240VAC, 47-63Hz
..... 6-27VDC
Maximum voltage 264VAC, 30VDC
Maximum output current 2A/point
Max leakage current 0.1mA @ 264VAC
Smallest recommended load 5mA @ 5VDC
OFF to ON response <15ms
ON to OFF response <10ms
Status indicators Logic Side
Fuses None (external recommended)

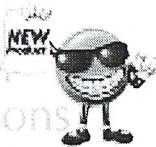


ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



D0-05DD-D

wiring diagram
and specifications



Power Requirements

Voltage range12-24VDC
.....20W max.

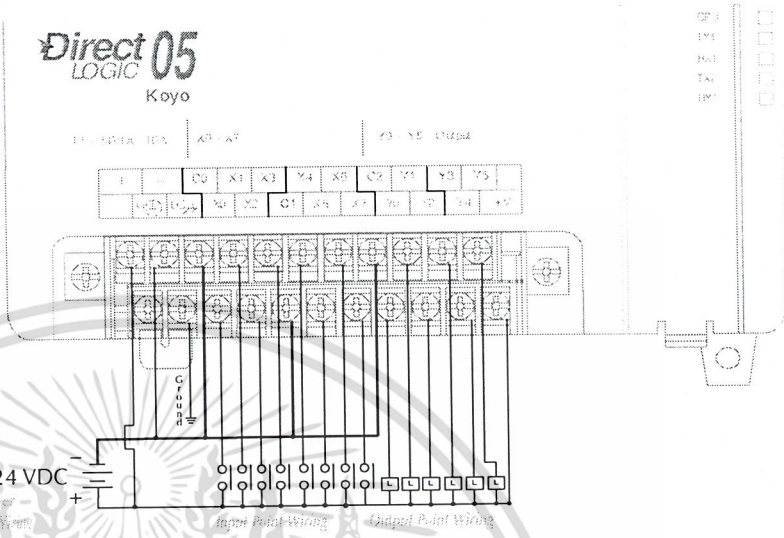
DC Input specifications

Number of input points8 (sink/source)
Number of commons2 (isolated)
Input voltage range12-24VDC
Input impedance(X0-X2) 1.8K @ 12-24VDC
.....(X3-X7) 2.8K @ 12-24VDC
ON current/voltage level>5mA/>10VDC
OFF current/voltage level<0.5mA/<2VDC
Response timeX0-X2 X3-X7
OFF to ON<100µs <8ms
ON to OFF<100µs <8ms
FusesNone

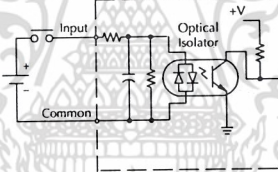
DC Output specifications

Number of output points6 (sinking)
Number of commons1
Output voltage range6-27VDC
Peak voltage50VDC
Max.frequency (Y0,Y1)7kHz
ON voltage drop0.3VDC @ 1A
Maximum current0.5A/point (Y0-Y1)*
.....1.0A point (Y2-Y5)
Maximum leakage current15µA @ 30VDC
Maximum inrush current2A for 100ms
OFF to ON response<10µs
On to OFF response<30µs
External DC power required20-28VDC
.....150mA max.
Status indicatorsLogic side
FusesNone (external recommended)

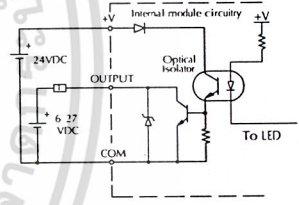
* When output points Y0 and Y1 are not used in pulse mode, the maximum output current is 1.0A.



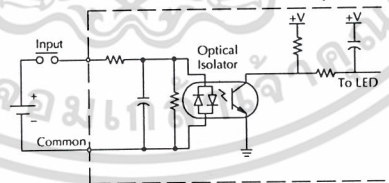
Equivalent Input Circuit, High-Speed Inputs (X0-X2)



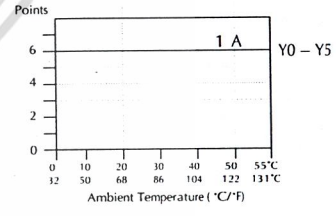
Equivalent Output Circuit

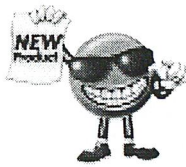


Equivalent Input Circuit, Standard Inputs (X3-X7)



Derating Factor of Outputs



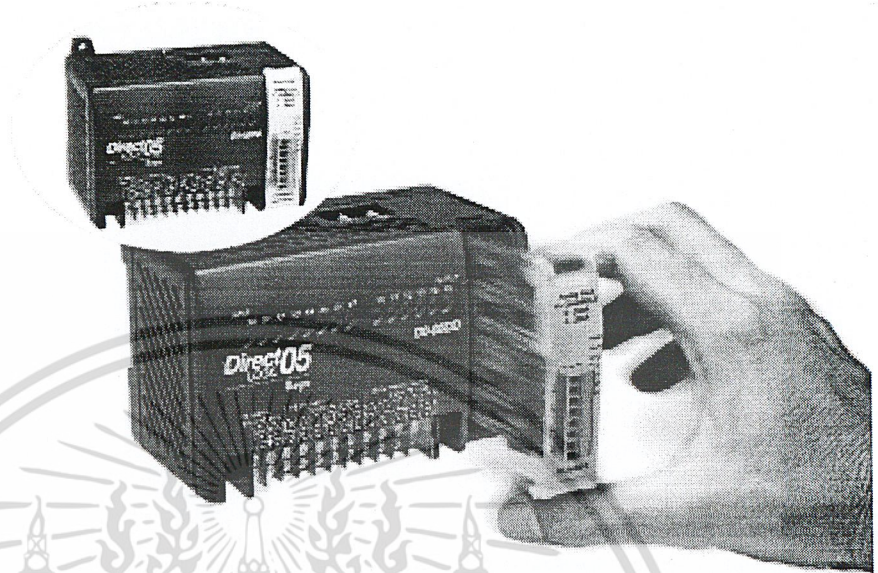


DL05 Option Cards



Want more from your DL05?

We built in an option card slot, so you can customize your DL05 PLC to fit your application. Now you can add features without adding size. Do you want to back up or transfer your memory programs? Get the optional memory cartridge and send your program around the building or around the world. Do you need to measure temperature or pressure? Get the 4-channel analog input module. Our list of option cards already includes a DeviceNet communication module, and more choices are on the way. Stay tuned to our Web site.



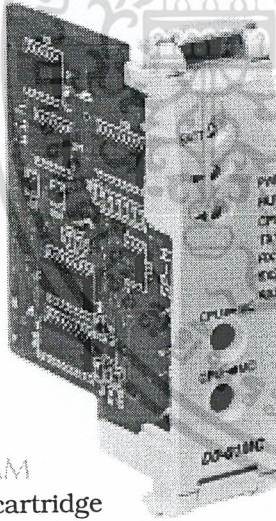
D0-01MC

Simple and inexpensive

The D0-01MC slides easily into the option card slot in any DL05 PLC to back up PLC programs and data for extended periods of time.

Battery-backed RAM

The memory cartridge makes programs portable from one DL05 PLC to another. The memory map is identical to the internal memory in the DL05 PLC, so no program changes are necessary. The on-board lithium battery lasts up to three years. If PLC power is lost and the battery is already dead, an on-board super capacitor backs up the memory 4-7 days, allowing time to insert a new battery.



Real-time clock

Access the year, month, day of the week, hour, minute and second for event scheduling or data logging applications.

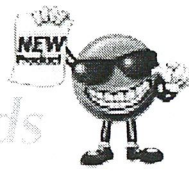
Operation

The D0-01MC installs into any of the DL05 PLCs. The MC backs up all ladder and data in CMOS RAM. The module's V-memory maps one-for-one to the PLC's memory locations.

If the Memory Cartridge is inserted in the option slot, it automatically becomes the source of the controlling program. You may choose to overwrite the PLC program, but you do not have to. You can transfer the program from the PLC to the MC, or from the MC to the PLC, or you can operate directly from the MC. By removing the module, you return control to the PLC's internal program. Two pushbuttons on

the face of the module initiate memory transfers. The pushbuttons are clearly marked to indicate the direction of the transfer, and an LED flashes to confirm the direction and success of the memory transfer.

A jumper enables/disables the write function in the MC. Write disable prevents overwriting the MC memory. Write enable allows overwriting the MC memory. An LED alerts you to a low battery condition. If the battery drops below 2.5V the "BATT" LED comes on, and an internal bit is set. You can use the internal bit to activate alarm functions or to execute an orderly shutdown. The date and time are easily set or accessed in the ladder logic program. The "year" field contains four digits so it is ready for Y2K and beyond. Environmental specifications for the D0-01MC are the same as for the DL05 PLCs. UL and CE approvals are pending.



F0-04AD-1 4-channel analog input module

The F0-04AD-1 Analog Input module offers the following features:

- The DL05 reads all four channels in one scan.
- The removable terminal block makes it possible to remove the module without disconnecting the field wiring.
- Analog inputs can be used as process variables for the (4) PID loops included in the DL05 CPU.
- Field device burnout is detected on all four channels when 4–20mA range is selected.
- On-board active analog filtering and RISC-like microcontroller provide digital signal processing to maintain precise analog measurements in noisy environments.

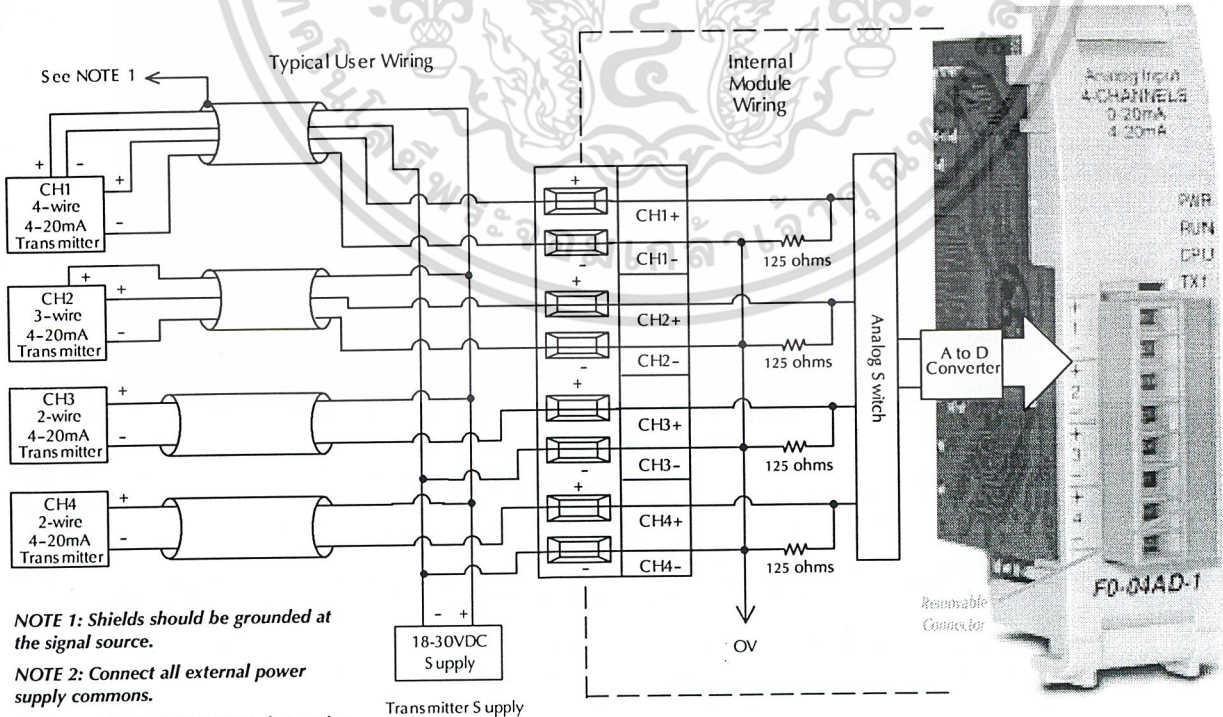
Input Specifications

Number of channels	4, single ended (one common)
Input range	0 to 20 mA or 4 to 20 mA current (jumper selectable)
Resolution	12 bit (1 in 4096) for 0 - 20mA, scaled for 4 - 20mA
Step response	25.0 mS (typ) to 95% of full step change
Crosstalk	.80 dB, 1/2 count maximum *
Active low-pass filtering	.3 dB at 40Hz (-12 dB per octave)
Input impedance	125 Ohm \pm 0.1%, 1/8 W current input
Absolute maximum ratings	-30 mA to +30 mA current input
Converter type	Successive approximation
Linearity error (end to end)	\pm 1 count (0.025% of full scale) maximum *
Input stability	\pm 1 count *
Full scale calibration error (Offset error not included)	\pm 10 counts maximum, @ 20mA current input
Offset calibration error	\pm 5 counts maximum @ 4mA current input *
Maximum inaccuracy	\pm 4% @ 25°C (77°F) \pm 85% 0 to 60°C (32 to 140°F)
Accuracy vs. temperature	\pm 80 ppm/°C maximum full scale calibration (including maximum offset change)
Recommended fuse (external)	0.032 A, Series 217 fast-acting current inputs

* One count in the specification table is equal to one least significant bit of the analog data value (1 in 4096).

General Specifications

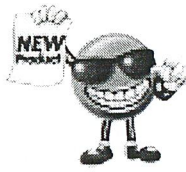
PLC update rate	4 channels per scan
16-bit data word	12 binary data bits, 2 channel ID bits, 2 diagnostic bits
Operating temperature	0 to 60°C (32 to 140° F)
Storage temperature	-20 to 70°C (-4 to 158° F)
Relative humidity	.5 to 95% (non-condensing)
Environmental air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise immunity	NEMA ICS3-304



NOTE 1: Shields should be grounded at the signal source.

NOTE 2: Connect all external power supply commons.

NOTE 3: A Series 217, 0.032A fast-acting fuse is recommended for current loops.



DL05 Option Cards



D0-DEVNETS slave

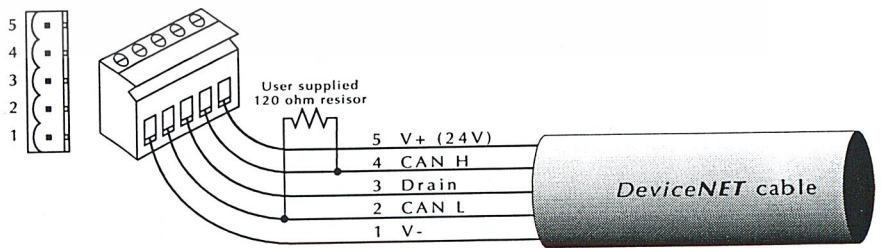
The D0-DEVNETS option card transforms any DL05 into a smart device node on your DeviceNet controller network. Now you don't have to turn to a more expensive PLC to get DeviceNet capability. DeviceNet is a low-cost control bus used to connect field devices to PLCs and PCs. DeviceNet designed to reduce the need for hard-wiring while providing device-level diagnostics. This industrial protocol links up to 64 nodes on a single network. The D0-DEVNETS slave module slides into the option card slot of any DL05 PLC. The module collects and reports all discrete I/O data to a DeviceNet master. The D0-DEVNETS module has a removable connector that makes the four-wire connection easy to implement and maintain. The DeviceNet module incorporates advanced diagnostics not commonly found on traditional industrial networks. This module has the quick response time and high dependability expected out of any DeviceNet device.

General Specifications

DeviceNet compatibility	Predefined Group 2 Master/Slave communications
Maximum field devices per bus	64 (see table below)
Communication to field devices	Standard 4 wire shielded cable to cabinet connector, molded 4-wire cable @ up to 500Kbps to field devices
Module connector	5 position removable terminal (European style)
Operating Temperature	0 to 55°C (32 to 131° F)
Storage Temperature	-20 to 70°C (-4 to 158° F)
Relative Humidity	5 to 95% (non-condensing)
Environmental air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3-304
	Impulse noise 1µs, 1000V
	FCC class A
	RFI (144Mhz, 430Mhz 10W, 10cm)

Trunk Length		Bits per sec	Branch Length		Devices
Feet	Meters		Feet	Meters	
328ft	100m	500Kbps	20ft	6m	64
820ft	250m	250Kbps	20ft	6m	64
1,640ft	500m	125Kbps	20ft	6m	64

Other DeviceNet specifications, compatible products, and latest DeviceNet information are made available through:
 Open DeviceNet Vendor Association
 Contact: William H. (Bill) Moss, Executive Director
 Phone: (954) 340-5412 Fax: (954) 340-5413
 Internet Address: <http://www.odva.org>
 e-mail: odva@powerintertnet.com
 ODVA, Inc.
 20423 State Road 7
 Boca Raton, FL 33498



Selected DL05 Micro PLCs offer special high-speed input features (on units with DC inputs) and pulse output features (on units with DC outputs). These features are available on the first 3 input points (X0-X2) and the first 2 output points (Y0-Y1). This allows you to use the economical DL05 Micro PLC to solve an incredibly diverse range of motion and high-speed machine control applications.

There are several modes of operation from which to choose. Here's a brief description of modes provided:

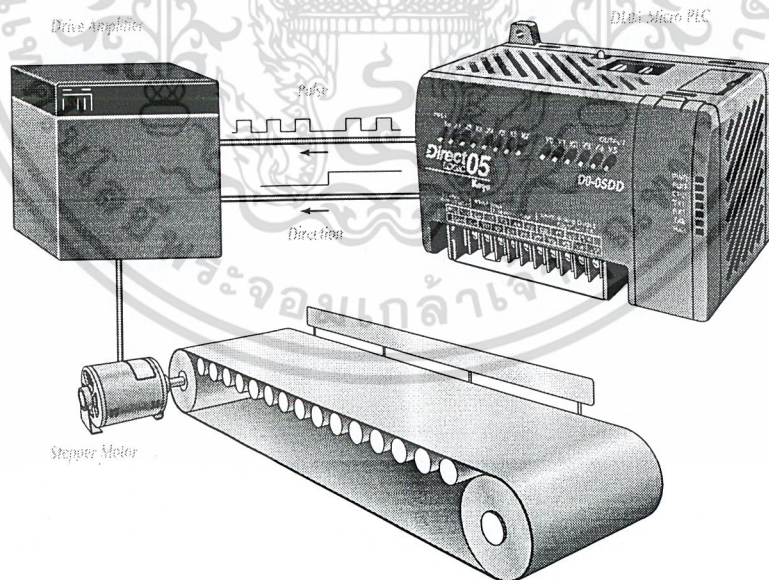
- Single 5kHz high-speed counter with 24 presets. When the preset is reached, an interrupt routine is executed.
- Single quadrature encoder input (up/down counter) for clockwise and counterclockwise position control.
- Single channel programmable 7kHz pulse output with an external interrupt and separate acceleration/deceleration profiles for positioning and velocity control.
- A single external interrupt input for an immediate response to time critical tasks.
- Single pulse catch input allows the CPU to read an input with a pulse width as small as 0.1ms.
- Three inputs with selectable filters (0-99ms) to ensure input signal integrity. This is the default mode, which is set at 10ms filter.

Counter Mode Options					
Mode	DC Inputs Points			DC Output Points	
	X0	X1	X2	Y0	Y1
High-Speed Up Counter	Counter Clock	Filtered Input	Filtered Input or Reset Count	Regular Output	Regular Output
Quadrature Counter	Phase A Input	Phase B Input	Filtered Input or Reset Count	Regular Output	Regular Output
High-Speed Interrupt	Interrupt Input	Filtered Input	Filtered Input	Regular Output	Regular Output
Pulse Catch	Pulse Input	Filtered Input	Filtered Input	Regular Output	Regular Output
Pulse Output	Filtered Input	Filtered Input	Filtered Input or Positioning Interrupt	Pulse or CW Pulse	Direction or CCW Pulse
Filtered Input	Filtered Input	Filtered Input	Filtered Input	Regular Output	Regular Output

- There are two timed interrupts. Interrupt 0 can be scheduled on a 5ms-999ms cycle. Interrupt 1 can be scheduled on a 5ms-9999ms cycle. (All units have this feature.)
- Combine features to use the full potential of the module. Some modes do not use all available points, so in some cases you can assign one of the other features to the point(s) not used by the main mode of operation.

You cannot use the DL05 for closed-loop control (you cannot use the UP counter and pulse output features at the same time).

You can easily select the mode of operation just by entering an appropriate code in a special CPU V-memory location. These features are explained in more detail later in this section. Remember, not all features can be used at the same time. The table above provides point-by-point usage for each mode of operation.

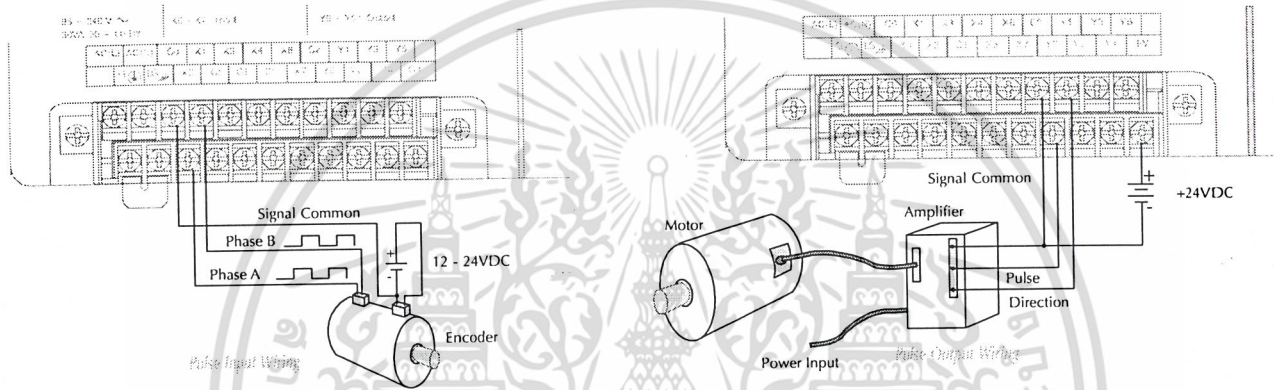


High Speed Specifications

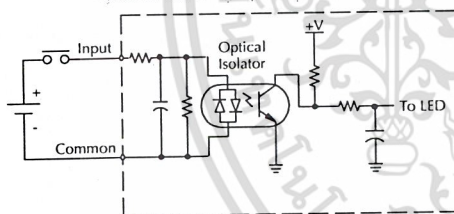


High-Speed Input Specifications	
Inputs	3 pts. Max. X0-X2 sink or source 5kHz Max
Minimum pulse width	100 μ s
Input voltage range	12-24 VDC
Input impedance (X0-X2)	1.8K @ 12-24VDC
ON current/voltage level	>5mA/>10VDC
OFF current/voltage level	<0.5mA/<2VDC
OFF to ON response	<100 μ s
ON to OFF response	<100 μ s

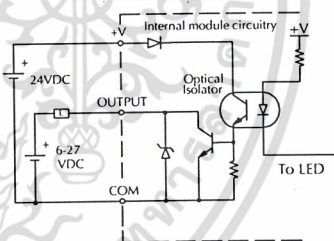
High-Speed Output Specifications	
Outputs	2 pts. Max., Y0&Y1 current sinking, 7kHz Max
Voltage range	6-27VDC
Maximum load current	0.5A/point
ON voltage drop	0.3VDC @ 1.0A
Leakage current	15 μ A @ 30VDC
Inrush current	2A (100ms)
OFF to ON response	<10 μ Sec
ON to OFF response	<30 μ Sec



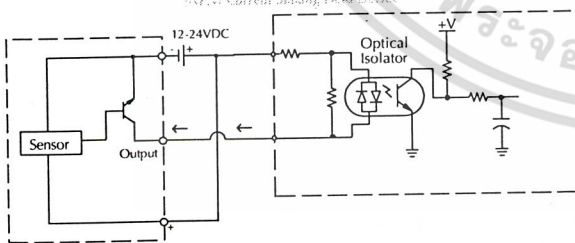
Equivalent Circuit, High-Speed Inputs (NPN)



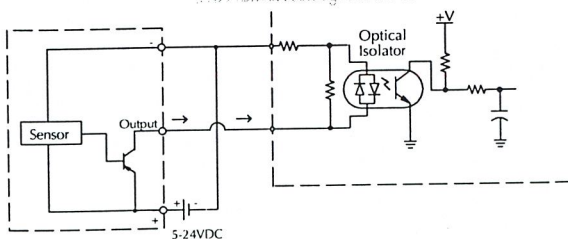
Equivalent Output Circuit



Equivalent Circuit, High-Speed Input (NPN) Current Sinking Field Device



Equivalent Circuit, High-Speed Inputs (PNP) Current Sourcing Field Device



Overview

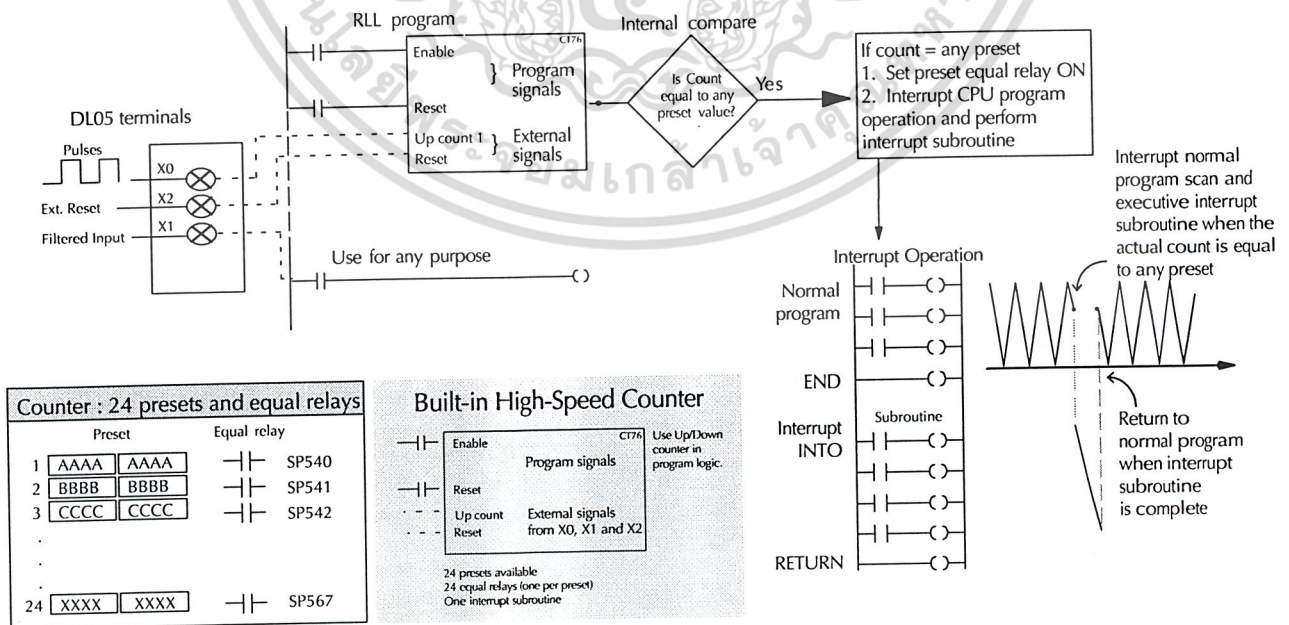
Each DL05 Micro PLC with DC inputs has internal features embedded that support a single high-speed counter up to 5kHz. You connect the external pulse input and reset input signals to the internal counter by using discrete input points (X0-X2). The embedded counter is independent of the Micro PLC's ladder logic execution, so counting is not affected by the scan time. When the counter reaches any one of up to 24 preset values, the Micro PLC stops executing the main RLL program and executes a special interrupt subroutine that is associated with the UP counter. The CPU resumes normal operations from where it left off after the

interrupt subroutine is finished. You can program the subroutine with any of the instructions that are normally available in subroutines. Also, each preset value has a corresponding Equal relay. These are individual internal control relays that are turned on when the associated preset matches the actual count. This allows you to easily trigger actions based on the current count. For example, you could use Immediate I/O instructions to provide a fast response. Using the counter in your program is fairly simple. An up/down counter box is used, and you can start and stop the counter just by turning an enable contact (of your choice) on and off as needed. Counters can be reset

either by an external signal (X02) or by special internal relays that can be activated by the program. Presets are absolute, which means they are compared directly to the actual count.

Input Assignments for Up Counter Mode	
X0:	Up count of UP counter
X1:	Filtered input
X2:	External counter reset (or filtered input)
Input Specifications	
Input voltage	12 or 24VDC
Frequency	5kHz maximum
Minimum pulse width	100 μSec
Maximum count	99,999,999
Preset types	Absolute
Number of presets	24

Example Operation



Mode 20: Single Quadrature Counter

Overview

By selecting Mode 20, you can use the high-speed input features to connect up to a 5kHz quadrature encoder. In this mode, you can have two external pulse inputs from the encoder (Phase A and Phase B) and one reset input signal. These are connected to the DL05 Micro PLC at points X0, X1 and X2 respectively (X2 can be used as an external reset or as a discrete filtered input). In addition to the physical inputs, there are also 2 internal references used in the control program, a counter enable input, and a counter reset input. Note: you cannot use two individual encoders as the input devices with the DL05 Micro PLC. Like the UP counter, the quadrature counter is independent of the CPU ladder logic execution, so the actual pulse counting is not affected by the scan time. However, the quadrature counter

cannot trigger an interrupt based on the current count/preset relationship.

To perform simple positioning or to control output devices, you must use relational contacts (based on the current count) within your RLL program. Since these contacts are within the RLL program, the resolution obtained with this method is actually limited by the PLC scan time. That is, the margin for error is equal to the maximum number of pulses that could be expected during one scan. You can determine the resolution with a simple formula: $\text{Pulses per scan} = \text{Scan Time (ms)} \times \text{Frequency (kHz)}$. For example, a 10ms scan and a 5kHz encoder input (0.01seconds x 5000Hz) yields a maximum of 50 pulses per scan. So, the maximum positioning precision would be the amount of encoder revolution that yields 50 pulses (the amount of precision will also

depend on the field device delay, PLC output off/on delay, etc.). This amount of precision may be acceptable for many simple positioning applications. If you need additional flexibility for your application, check out our DL205 Micro Modular family. The powerful DL240/250 CPU and inexpensive DL205 Counter Interface Module may be just what you need.

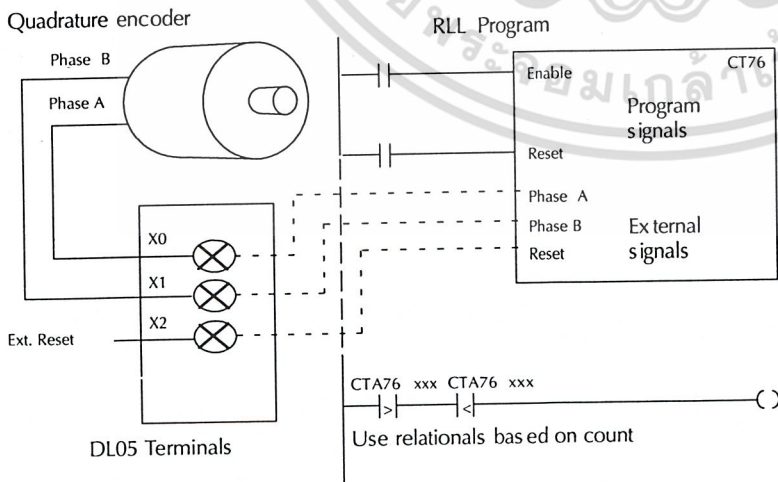
Input Assignments for Quadrature Counter Mode

X0: Phase A
 X1: Phase B
 X2: External counter reset (or filtered input)

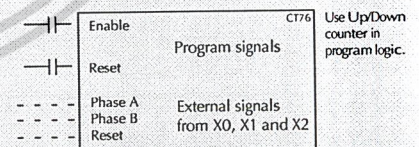
Input Specifications

Input voltage 12 or 24VDC
 Frequency 5kHz maximum
 Minimum pulse width 100 µSec
 Count range 0 to 99,999,999 unipolar -
 -8,388,608 to 8,388,607 bipolar
 Number of presets None, use relational
 contacts or use CT76 status contact

Example Operation



Built-in Quadrature Counter



Overview

By selecting Mode 30, you can use the pulse output feature to build simple motion and positioning control systems. Transfer and indexing tables are common applications. There are two operation profiles available (shown below). You choose the profile and motion parameters by using special CPU V-memory locations that are reserved for use by the high-speed I/O features. You can configure the pulse output for independent CW/CCW pulse train output, or step and direction. With independent operation, Y0 is the CW pulse output and Y1 is the CCW pulse output. If you choose step and direction, Y0 is the pulse train output and Y1 controls the CW/CCW operation (OFF/ON respectively). In either case, the pulses are sent out independently of the CPU scan, so scan time does not affect the

pulse generation. The pulse output is enabled through ladder logic by activating Y0.

Automatic accel/decel profile

The trapezoid profile is also referred to as the Automatic Acceleration/Deceleration profile. You specify a target destination (number of pulses), a starting velocity (pulses per second), a positioning velocity, an acceleration time, and a deceleration time. Once these parameters are specified, the DL05 automatically controls the actual acceleration/deceleration. Times can be in the range of 100ms to 10 seconds. This profile also allows you to perform simple registration. By using the external interrupt (X2), you can delay counting toward the target number of pulses until the interrupt occurs.

Velocity control

You can also choose a velocity-only profile. In this scenario, you only control the velocity. That is, there is no target destination (number of pulses). You simply change the velocity value as necessary to achieve the desired results.

Input Assignments for Pulse Output Mode

X0: Filtered Input
 X1: Filtered Input
 X2: Filtered Input or Positioning Interrupt

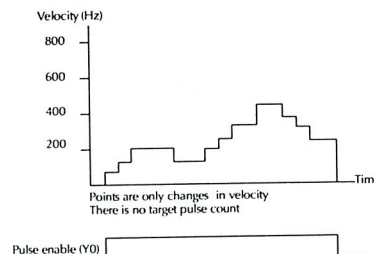
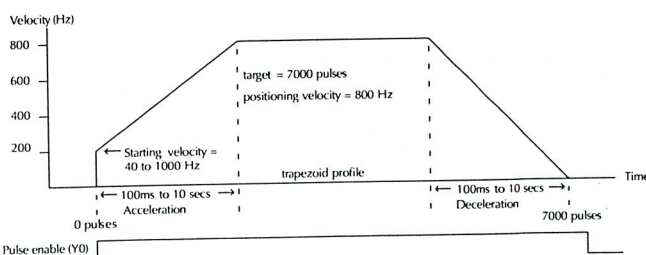
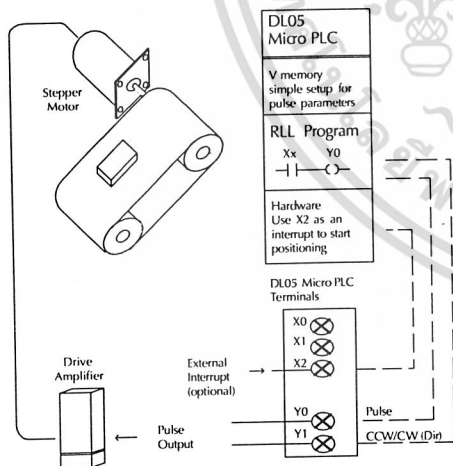
Output Assignments for Pulse Output Mode

Y0: Independent Mode, CW pulse output
 Step & Direction Mode, pulse output
 Y1: Independent Mode, CCW pulse output
 Step & Direction Mode, OFF=CCW, ON=CCW

Output Specifications

Output voltage range 6-27VDC
 Frequency 7kHz maximum
 Target pulse range -8,388,608 to 8,388,607
 Velocity range 40 to 7000 pulses/sec
 (in units of 10 pulses)

Example Operation



Mode 40 and Mode 50

Mode 40: External Interrupt overview

By selecting Mode 40, you can use X0 as a high-speed interrupt input. An interrupt input is especially useful in applications that have a high-priority event which requires special operations to be performed. When this high-priority event occurs, the interrupt input senses an ON input signal. The input automatically sends an interrupt request to the CPU. The CPU immediately suspends its routine scan cycle execution and jumps to an interrupt subroutine. You can program the subroutine with any of the instructions that are normally available in subroutines. For example, you could use immediate I/O instructions to immediately read inputs and update outputs without waiting on the normal I/O update cycle.

When the subroutine is complete, the CPU automatically resumes the normal scan cycle starting from the exact location where it was interrupted. The CPU continues the routine scan until another interrupt signal is sensed.

A note on timed interrupts

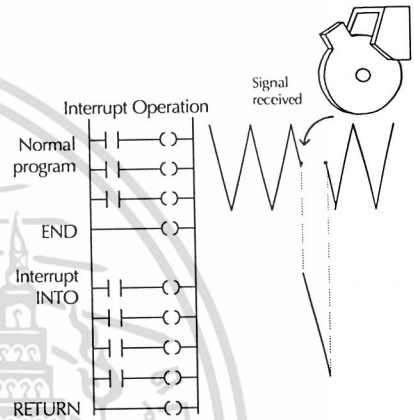
If you use the external hardware interrupt (mode 40), you cannot use the internal timed interrupt INT0. This is because they both share the same interrupt routine, INT0. In this case, you can use only the internal timed interrupt INT 1.

Input Assignments for Interrupt Mode

- X0: Interrupt input
- X1: Filtered input
- X2: Filtered input

Input Specifications

- Input voltage 12 or 24VDC
- Minimum pulse width 100 µSec
- Pulse period 0.5ms or greater
- Trigger Leading edge
- Interrupt subroutine INT0



Mode 50: Pulse Catch Input overview

By selecting Mode 50, you can use X0 as a Pulse Catch input. In this configuration, the DL05 Micro PLC can capture very fast (narrow) pulse inputs that cannot normally be detected during the normal input update cycle. You can detect pulse widths as small as 0.1ms (100µS) and a pulse

period greater than 0.5ms (500µS). When an external pulse is encountered, X0 is set on for the next CPU scan, and then it is automatically set to the OFF state. Like the other modes, the pulse catch feature operates independently of the CPU scan and is not affected by scan time fluctuations.

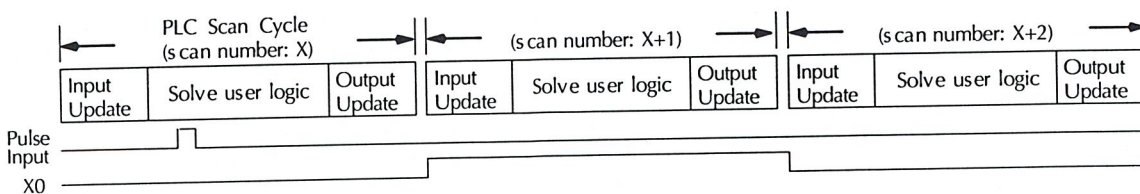
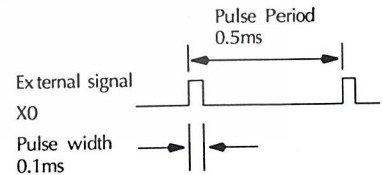
Mode 50 is not recommended for high-speed pulse counting.

Input Assignments for Pulse Catch Mode

- X0: Pulse catch input
- X1: Filtered input
- X2: Filtered input

Input Specifications

- Input voltage 12 or 24VDC
- Minimum pulse width 100 µSec
- Pulse period 0.5ms or greater
- Trigger Leading edge



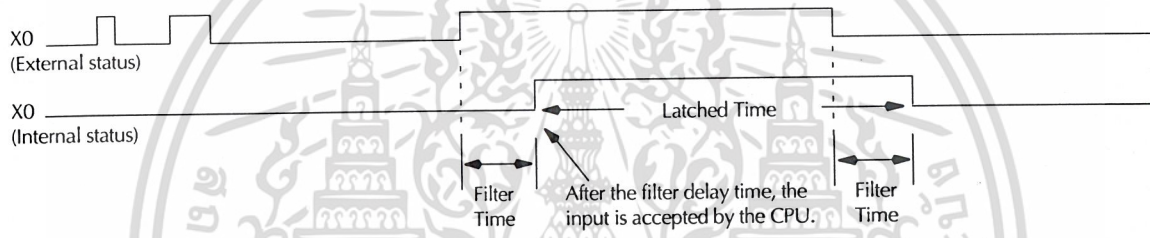
Default Mode 60: Three Discrete Inputs with Filter overview

Mode 60, which is the default mode set at the factory, provides selectable filtering for the first three inputs (X0-X2). Filtering can be especially useful because it reduces the possibility of false ON conditions (which can in turn trigger events in your ladder logic program). When an external

signal is first detected (ON state), a programmable filter is activated, which begins a timed countdown. The slight delay temporarily prevents the CPU from reading the input during the normal input update portion of the scan cycle. The ON signal must stay present long enough for the filter to time out. If the ON signal stays present during the entire filter time, it is latched by the filter and allowed to be accepted by the CPU during the CPU's normal input update portion of the scan cycle. The

signal is latched for the remaining duration of the ON signal plus an amount of time equal to the filter time. The filter time can be programmed from 0 to 99ms in 1ms increments (default is 10ms).

Input Assignments for Filtered Input Mode
 Point assignments: 3 inputs (X0,X1,X2)
 Filter time: Programmable from 0-99ms in 1ms increments



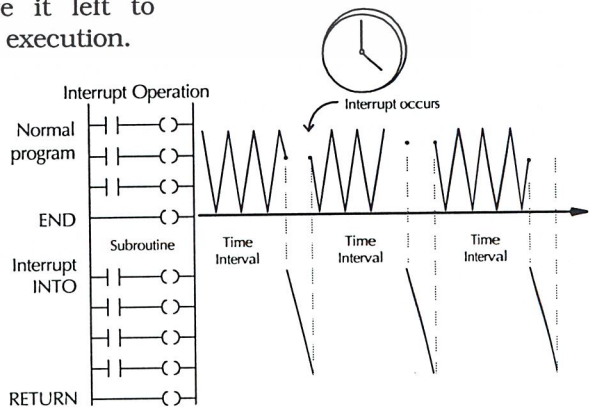
Understanding the Timed Interrupt

There is also a timed interrupt feature available in the DL05 Micro PLCs. This cyclical interrupt allows you to easily program a time-based interrupt that occurs on a scheduled basis. This feature is available in all units, regardless of input type. The CPU's timed interrupt operates in a similar manner to the external interrupt input, but instead of the interrupt subroutine being triggered by an external event tied to X0, it is now triggered by a cyclical interval of time. This interval can be programmed from 5ms to 999ms for INTO and 5 to 9999ms for INT1. Whenever the programmed time elapses, the CPU immediately suspends its routine scan cycle and jumps to the selected interrupt subroutine. As

with the other modes, when the interrupt subroutine execution is complete, the CPU automatically resumes its routine scan cycle starting from the exact location where it was interrupted. Since the CPU scan time and the interrupt time interval are different, the RLL program gets interrupted at various points in the execution over time. This does not present a problem. The CPU always returns to the point where it left to resume the program execution.

Input Assignments for Timed Interrupt Mode
 X0: Filtered input (uses filter time set for X1)
 X1: Filtered input
 X2: Filtered input

Timed Interrupt Specifications
 Timed interrupts: 2 (internal to CPU)
 Interrupt Subroutine: INTO,1
 Time interval:
 INTO 5 to 999ms (1ms increments)
 INT1 5 to 9999ms (1ms increments)



Boolean Instructions	Comparative Boolean Instructions	Immediate Instructions
<p>AND Begins a new rung or an additional branch in a rung with a normally open contact.</p> <p>ANDC Begins a new rung or an additional branch in a rung with a normally closed contact.</p> <p>ANDN Logically ors a normally open contact in parallel with another contact in a rung.</p> <p>ANDNC Logically ors a normally closed contact in parallel with another contact in a rung.</p> <p>ANDNO Logically ands a normally open contact in series with another contact in a rung.</p> <p>ANDNCNO Logically ands a normally closed contact in series with another contact in a rung.</p> <p>ANDNOO Logically ands two branches of a rung in series.</p> <p>ANDNOOC Logically ands two branches of a rung in parallel.</p> <p>ANDNOON Reflects the status of the rung (on/off) and outputs the discrete (on/off) state to the specified image register point or memory location.</p> <p>ANDNOONC Reflects the status of the rung and outputs the discrete (ON/OFF) state to the image register. Multiple OR OUT instructions referencing the same discrete point can be used in the program.</p> <p>ANDNOONNO Inverts the status of the rung at the point of the instruction.</p> <p>ANDNOONNC An output that turns on a point or a range of points. The reset instruction is used to turn the point(s) OFF that were set ON with the set instruction.</p> <p>ANDNOONNOO An output that resets a point or a range of points.</p> <p>ANDNOONNOOC Disables the update for a range of specified output points.</p>	<p>ANDC Begins a new rung or additional branch in a rung with a normally open comparative contact. The contact will be on when $A = B$.</p> <p>ANDNC Begins a new rung or additional branch in a rung with a normally closed comparative contact. The contact will be on when $A \neq B$.</p> <p>ANDNO Connects a normally open comparative contact in parallel with another contact. The contact will be on when $A = B$.</p> <p>ANDNOC Connects a normally closed comparative contact in parallel with another contact. The contact will be on when $A \neq B$.</p> <p>ANDNOO Connects a normally open comparative contact in series with another contact. The contact will be on when $A = B$.</p> <p>ANDNOOC Connects a normally closed comparative contact in series with another contact. The contact will be on when $A \neq B$.</p> <p>ANDNOON Begins a new rung or additional branch in a rung with a normally open comparative contact. The contact will be on when $A \geq B$.</p> <p>ANDNOONC Begins a new rung or additional branch in a rung with a normally closed comparative contact. The contact will be on when $A \geq B$.</p> <p>ANDNOONNO Connects a normally open comparative contact in parallel with another contact. The contact will be on when $A \geq B$.</p> <p>ANDNOONNC Connects a normally open comparative contact in parallel with another contact. The contact will be on when $A < B$.</p> <p>ANDNOONNOO Connects a normally open comparative contact in series with another contact. The contact will be on when $A \geq B$.</p> <p>ANDNOONNOOC Connects a normally closed comparative contact in parallel with another contact. The contact will be on when $A < B$.</p>	<p>ANDNOONNOO Begins a rung/branch of logic with a normally open contact. The contact will be updated with the current input field status when processed in the program scan.</p> <p>ANDNOONNOOC Begins a rung/branch of logic with a normally closed contact. The contact will be updated with the current input field status when processed in the program scan.</p> <p>ANDNOONNOO Connects a normally open contact in parallel with another contact. The contact will be updated with the current input field status when processed in the program scan.</p> <p>ANDNOONNOOC Connects a normally open contact in series with another contact. The contact will be updated with the current input field status when processed in the program scan.</p> <p>ANDNOONNOO Connects a normally closed contact in series with another contact. The contact will be updated with the current input field status when processed in the program scan.</p> <p>ANDNOONNOOC Reflects the status of the rung. The output field device status is updated when the instruction is processed in the program scan.</p> <p>ANDNOONNOO Reflects the status of the rung and outputs the discrete (ON/OFF) state to the image register. Multiple OR OUT instructions referencing the same discrete point can be used in the program. The output field device status is updated when the instruction is processed in the program scan.</p> <p>ANDNOONNOOC An output that turns on a point or a range of points. The reset instruction is used to turn the point(s) off that were set. The output field device status is updated when the instruction is processed in the program scan.</p> <p>ANDNOONNOO An output that resets a point or a range of points. The output field device status is updated when the instruction is processed in the program scan.</p>

Timer, Counter, and Shift Register Instructions	Accumulator/Stack Load and Output Data	Differential Instructions
<p>ANDNOONNOO Single input incrementing timer with 0.1 second resolution (0-999.9 seconds)</p> <p>ANDNOONNOOC Single input incrementing timer with 0.01 second resolution (0-99.99 seconds)</p> <p>ANDNOONNOO Two input incrementing timer with 0.1 second resolution (0-9,999.999.9 sec.). Time and enable/reset inputs control the timer.</p> <p>ANDNOONNOOC Two input incrementing timer with 0.1 second resolution (0-9,999.999.9 sec.). Time and enable/reset inputs control the timer.</p> <p>ANDNOONNOO Two input incrementing counter (0-9999). Count and reset inputs control the counter.</p> <p>ANDNOONNOOC Single input incrementing counter (0-9999). RST instruction must be used to reset count.</p> <p>ANDNOONNOO Three input counter (0-99,999,999). Up, down and reset inputs control the counter.</p> <p>ANDNOONNOO Shifts data through a range of control relays with each clock pulse. The data clock and reset inputs control the shift register.</p>	<p>ANDNOONNOO Loads a 16-bit word into the lower 16 bits of the accumulator/stack.</p> <p>ANDNOONNOOC Loads a 32-bit word into the accumulator/stack.</p> <p>ANDNOONNOO Loads the accumulator with a specified number of consecutive discrete memory bits.</p> <p>ANDNOONNOOC Loads the accumulator with the HEX value for an octal constant (address).</p> <p>ANDNOONNOO Copies the value in the lower 16 bits of the accumulator to a specified V memory location.</p> <p>ANDNOONNOOC Copies the value in the accumulator to two consecutive V memory locations.</p> <p>ANDNOONNOO Outputs a specified number of bits (1-32) from the accumulator to the specified discrete memory locations.</p> <p>ANDNOONNOOC Moves the value from the first level of the accumulator stack to the accumulator and shifts each value in the stack up one level.</p>	<p>ANDNOONNOO One-shot output coil. When the input logic produces an off to on transition, the output will energize for one CPU scan.</p> <p>ANDNOONNOOC Leading edge triggered one-shot contact. When the corresponding memory location transitions from low to high, the contact comes on for one CPU scan.</p> <p>ANDNOONNOO Trailing edge triggered one-shot contact. When the corresponding memory location transitions from high to low, the contact comes on for one CPU scan.</p> <p>ANDNOONNOOC Logically ors a leading edge triggered one-shot contact in parallel with another contact in a rung.</p> <p>ANDNOONNOO Logically ors a trailing edge triggered one-shot contact in parallel with another contact in a rung.</p> <p>ANDNOONNOOC Logically ands a leading edge triggered one-shot contact in series with another contact in a rung.</p> <p>ANDNOONNOO Logically ands a trailing edge triggered one-shot contact in series with another contact in a rung.</p>

Logical Instructions (Accumulator)	Math Instructions (Accumulator)	Math Instructions (Accumulator) cont'd
<p>ANDNOONNOO Logically ands the lower 16 bits in the accumulator with a V memory location.</p> <p>ANDNOONNOOC Logically ands the value in the accumulator with an 8-digit constant or a value in two consecutive V-memory locations.</p> <p>ANDNOONNOO Logically ors the lower 16 bits in the accumulator with a V memory location.</p> <p>ANDNOONNOOC Logically ors the value in the accumulator with an 8-digit constant or a value in two consecutive V-memory locations.</p> <p>ANDNOONNOO Performs an Exclusive Or of the value in the lower 16 bits of the accumulator and a V-memory location.</p> <p>ANDNOONNOOC Performs an Exclusive Or of the value in the accumulator and an 8-digit constant or a value in two consecutive V-memory locations.</p> <p>ANDNOONNOO Compares the value in the lower 16 bits of the accumulator with a V memory location.</p> <p>ANDNOONNOOC Compares the value in the accumulator with two consecutive V memory locations or an 8-digit constant.</p>	<p>ANDNOONNOO Adds a BCD value in the lower 16 bits in the accumulator with a V memory location. The result resides in the accumulator.</p> <p>ANDNOONNOOC Adds a BCD value in the accumulator with two consecutive V memory locations or an 8-digit constant. The result resides in the accumulator.</p> <p>ANDNOONNOO Subtract a BCD value, which is either a V memory location or a 4-digit constant from the lower 16 bits in the accumulator. The result resides in the accumulator.</p> <p>ANDNOONNOOC Subtracts a BCD value, which is either two consecutive V memory locations or an 8-bit constant, from a value in the accumulator. The result resides in the accumulator.</p> <p>ANDNOONNOO Multiplies a BCD value, which is either a V memory location or a 4-digit constant, by the value in the lower 16 bits in the accumulator. The result resides in the accumulator.</p> <p>ANDNOONNOOC Multiplies a BCD value contained in two consecutive V memory locations by the value in the accumulator. The result resides in the accumulator.</p> <p>ANDNOONNOO Divides a BCD value in the accumulator by a BCD value which is either a V memory location or a 4-digit constant. The result resides in the accumulator.</p> <p>ANDNOONNOOC Divides a BCD value in the accumulator by a BCD value which is either two consecutive V memory locations or a 8-digit constant. The result resides in the accumulator.</p>	<p>ANDNOONNOO Adds the binary value in the lower 16 bits of the accumulator to a value which is either a V memory location or a 16-bit constant. The result resides in the accumulator.</p> <p>ANDNOONNOOC Subtract a 16-bit binary value, which is either a V memory location or a 16-bit constant, from the lower 16 bits in the accumulator. The result resides in the accumulator.</p> <p>ANDNOONNOO Multiplies a 16-bit binary value, which is either a V memory location or a 16-bit constant, by the lower 16 bits in the accumulator. The result resides in the accumulator.</p> <p>ANDNOONNOOC Divides the binary value in the lower 16 bits in the accumulator by a value which is either a V memory location or a 16-bit constant. The result resides in the accumulator.</p> <p>ANDNOONNOO Increments a BCD value in a specified V memory location by 1 each time the instruction is executed.</p> <p>ANDNOONNOOC Decrements a BCD value in a specified V memory location by 1 each time the instruction is executed.</p> <p>ANDNOONNOO Increments a binary value in a specified V memory location by 1 each time the instruction is executed.</p> <p>ANDNOONNOOC Decrements a binary value in a specified V memory location by 1 each time the instruction is executed.</p>



Instruction Set (cont'd)

Table Instructions	CPU Control Instructions	Network Instructions
<p>Table Copy (TBLCPY) Moves the values from one V memory table to another V memory table.</p> <p>Table Copy (TBLCPY) Copies data between V memory and program ladder memory.</p>	<p>Table No Op (TBLNOOP) Inserts a no operation coil at specified program address.</p> <p>Table End (TBLEND) Marks the termination point for the normal program scan. An End instruction is required at the end of the main program body.</p> <p>Table Stop (TBLSTOP) Changes the operational mode of the CPU from Run to Program (Stop).</p> <p>Table Watchdog (TBLWDG) Resets the CPU watchdog timer.</p>	<p>Table Read (TBLRD) Reads a block of data from another CPU on the network.</p> <p>Table Write (TBLWR) Writes a block of data from the master device to a slave device on the network.</p>

Number Conversion Instructions (Accumulator)	Bit Instructions (Accumulator)	Interrupt Instructions
<p>Table BCD (TBLBCD) Converts the BCD value in the accumulator to the equivalent binary value. The result resides in the accumulator.</p> <p>Table BIN (TBLBIN) Converts the binary value in the accumulator to the equivalent BCD value. The result resides in the accumulator.</p> <p>Table ONES (TBLONES) Takes the one's complement of the 32-bit value in the accumulator. The result resides in the accumulator.</p> <p>Table ASCII (TBLASCII) Converts a table of ASCII values to a table of hexadecimal values.</p> <p>Table HEX (TBLHEX) Converts a table of hexadecimal values to a table of ASCII values.</p> <p>Table GRAY (TBLGRAY) Converts a 16-bit GRAY code value in the accumulator to a corresponding BCD value. The result resides in the accumulator.</p> <p>Table Shuffle (TBLSHFL) Shuffles a maximum of 8 digits, rearranging them in a specified order. The result resides in the accumulator.</p>	<p>Table Count (TBLCNT) Counts the number of bits set to "1" in the accumulator. The HEX result resides in the accumulator.</p> <p>Table Shift L (TBLSHL) Shifts the bits in the accumulator a specified number of places to the left.</p> <p>Table Shift R (TBLSHR) Shifts the bits in the accumulator a specified number of places to the right.</p> <p>Table Encode (TBLENC) Encodes the bit position set to 1 in the accumulator, and returns the appropriate binary representation in the accumulator.</p> <p>Table Decode (TBLDEC) Decodes a 5 bit binary value (0-31) in the accumulator by setting the appropriate bit position to 1 in the accumulator.</p>	<p>Table Interrupt (TBLINT) When a hardware or software interrupt occurs, the interrupt routine will be executed. The INT instruction is the beginning of the interrupt routine. The interrupt routine is terminated with an IRT of the interrupt routine. The interrupt routine is terminated with an IRT instruction (unconditional interrupt return). When a interrupt return is reached the execution of the program continues from the instruction where the program execution was prior to the interrupt.</p> <p>Table Enable (TBLENB) Enables hardware and software interrupts to be acknowledged.</p> <p>Table Disable (TBLDIS) Disables hardware and software interrupts from being acknowledged.</p>

Program Control Instructions	PLC Programming Instructions	Drum Instructions
<p>Table For Next (TBLFOR) Executes the logic between the FOR and NEXT instructions a specified number of times.</p> <p>Table GTS (TBLGTS) When a GTS instruction is executed the program jumps to the SBR (Subroutine). The subroutine is terminated with a RT instruction (unconditional return). When a return is executed, the program continues from the instruction after the calling GTS instruction.</p> <p>Table MLC (TBLMLC) Allows the program to control sections of ladder logic by forming a new power rail. The MLC marks the beginning of a power rail and the MLR marks the end of the power rail control.</p>	<p>Table LSG (TBLLSG) The initial stage instruction is used for a starting point for user application program. The LSG instruction will be active on power up and PROGRAM to RUN transitions.</p> <p>Table Stage (TBLSTG) Stage instructions are used to create structured programs. They are program segments which can be activated or deactivated with control logic.</p> <p>Table Stage NO (TBLSTGNO) Normally open coil that deactivates the active stage and activates a specified stage when there is power flow to the coil.</p> <p>Table Stage NC (TBLSTGNC) Normally closed coil that deactivates the active stage and activates a specified stage when there is power flow to the coil.</p> <p>Table Converge (TBLCV) Converge stages are a group of stages that when all stages are active the associated converge jump(s). (CVJMP) will activate another stage(s). One scan after the CVJMP is executed, the converge stages will be deactivated.</p> <p>Table Stage NO (TBLSTGNO) Normally open coil that deactivates the active CV stages and activates a specified stage when there is power flow to the coil.</p>	<p>Table Drum (TBLDRM) Time driven drum with up to 16 steps and 16 discrete output points. Output status is written to the appropriate output during each step. Specify a time base per count (in milliseconds). Each step can have a different number of counts to trigger the transition to the next step. Also define preset step as destination when reset occurs.</p> <p>Table Event Drum (TBLEDRM) Time and/or event driven drum with up to 16 steps and 16 discrete output points. Output status is written to the appropriate output during each step. Specify a time base per count (in milliseconds). Each step can have a different number of counts and an event to trigger the counting. Once the time has expired, a transition to the next step occurs. Also define preset step as destination when reset occurs.</p>

Message Instruction
<p>Table Display (TBLDISP) Displays a V memory value or a data label constant to the hand-held programmer or personal computer using DirectSOFT.</p> <p>Table Store (TBLSTR) Stores constants in numerical or ASCII form for use with other instructions.</p> <p>Table Print (TBLPRN) Prints the embedded text or text/data variable message to the specified communications port. Maximum message length is 255 words.</p>

Direct 105



5

Direct 105 Logic

Technical Overview

Features and Specifications	130
Dimensions and Installation Requirements	132
Power Supply and I/O Types	133
I/O Specifications	134
NEW! DeviceNet I/O	142
Four point Simulator	143
High-speed I/O Features	144
Modes	146

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Features and Specifications

The DL105 Micro PLCs contain the CPU, power supply and I/O all in the same housing. If you examine the CPU Specifications table, you'll see that we included many features found in our modular CPUs.

Review the specs

Make sure these features can satisfy the requirements of your application. Since these units are completely self-contained, you cannot expand the system or replace the CPU as you would in a modular system.

System capacity

System capacity is the ability to accommodate a variety of applications. For ladder memory, most Boolean instructions require one word. Some other instructions, such as timers, counters, etc. require two or more words. Our V-memory words are useful for data storage, etc.

Performance

The performance is simply the scan time, which is the amount of time required to read the inputs, solve the RLL program and update the outputs.

Instructions and diagnostics

Make sure the unit offers the instructions you need.

Communications

All DL105 units offer one RS-232 port, capable of 9,600 baud.

Specialty features

For the DC input and/or DC output versions, we also offer several high-speed I/O features.

AC powered

F1-130AA

10 AC inputs, 8 AC outputs, 1.7A/point

F1-130AD

10 AC inputs, 8 DC outputs, 1.0A/Point, two outputs can be used as 7kHz pulse output, 0.5A/point

F1-130AR

10 AC inputs, 8 relay outputs, 7A/point

F1-130DA

10 DC inputs, four inputs are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
8 AC outputs, 1.7A/point

F1-130DD

10 DC inputs, four points are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
8 DC outputs, 1.0A/point, 2 outputs can be used as 7kHz pulse output, 0.5A/point

F1-130DR

10 DC inputs, four inputs are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
8 relay outputs, 7A/point

DC powered

F1-130DD-D

10 DC inputs, four inputs can be used as 5kHz high-speed counter, interrupt inputs, or pulse catch inputs

8 DC outputs, 1.0A/point, two outputs can be used as 7kHz pulse output, 0.5A/point.

F1-130DR-D

10 DC inputs, four inputs can be used as 5kHz high-speed counter, interrupt inputs, or pulse catch inputs

8 Relay outputs, 7A/point

DeviceNet

F1-DEVNET-AR

10 AC inputs, 8 Relay outputs, 7A/point

F1-DEVNET-DD

10 DC inputs, four points are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
8 DC outputs, 1.0A/point, 2 outputs can be used as 7kHz Pulse Output, 0.5A/point

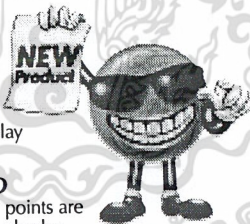
F1-DEVNET-DR

10 DC inputs, four inputs are filtered inputs, can also be configured as a single 5kHz high-speed counter, interrupt input, or pulse catch input
8 relay outputs, 7A/point

Programming

Handheld programmer

DirectSOFT Programming for Windows



DL105 CPU Specifications

System capacity

Total memory available (words)	2.4K
Ladder memory (words)	2,048 EEPROM
V-memory (words)	384
User V	256
Non-volatile User V	128
Battery backup	No
Total I/O	18
Inputs	10
Outputs	8
I/O expansion	No

Performance

Contact execution (Boolean)	3.3µs
Typical scan (1K Boolean)¹	5-6ms

Instructions and Diagnostics

RLL ladder style	Yes
RLL ^{PLUS} /flowchart style (Stages)	Yes/256
Run-time editing	Yes
Variable/fixed scan	Variable
Instructions	91
Control relays	256
Timers	64
Counters	64
Immediate I/O	Yes
Subroutines	No
For/next loops	No
Timed interrupt	Yes
Integer math	Yes
Floating-point math	No
PID	No
Drum sequencers	Yes
Bit of word	No
ASCII print	No
Real-time clock/calendar	No
Internal diagnostics	Yes
Password security	Multi-level
System and user error log	No

Communications

Built-in ports	one, RS-232C
K-sequence (proprietary protocol)	Yes
DirectNET TM	No
MODBUS master/slave	No
ASCLL out	No
Baud rate (fixed)	9,600 baud

Specialty Features

Filtered inputs	Yes ²
Interrupt input	Yes ²
High-speed counter	Yes, 5kHz ²
Pulse output	Yes, 7kHz ²
Pulse catch input	Yes ²

1- Our 1K program includes contacts, coils, and scan overhead. If you compare our products to others, make sure you include their scan overhead.

2- Input features are only available on units with DC inputs. Output features are only available on units with DC outputs

DL105 Hardware Features

CPU status indicators

- RUNON CPU is in RUN mode
-OFF CPU is in Program mode
- PWRON CPU power good
-OFF CPU power failure
- CPUON CPU internal diagnostics
-OFF has detected an error
-OFF CPU is OK

Mode control

The DL105 units do not have mode switches like many of our modular CPUs. You can set the unit (using special V-memory locations) so that it will power up in Run mode.

Communications port

- Protocol K-sequence slave
- Devices Can connect with HPP, DirectSOFT, DV-1000, EZText, and EZTouch Panels
- Specs 6P6C RJ12 connector
- RS-232C, 9,600 baud,
- Odd parity,
- Fixed station address (1),
- 8 data bits (one start, one stop bit),
- Asynchronous, half-duplex, DTE

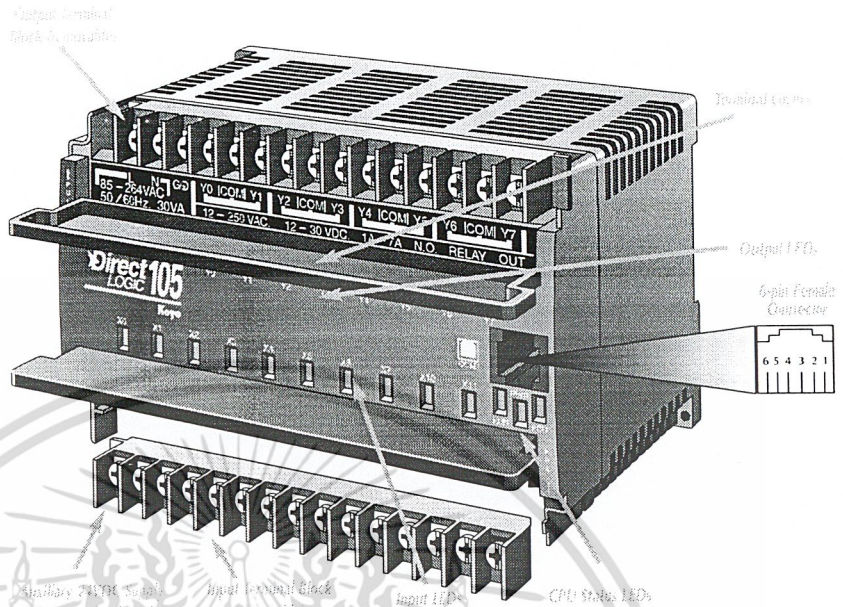
RJ12 Connector Port 1 Pinout

Pin	Signal
1	0V
2	5V
3	RS-232 Data in
4	RS-232 Data out
5	5V
6	0V

Fixed EEPROM memory

The DL105 units offer built-in EEPROM memory. However, memory cannot be removed or changed like the D2-230 and D2-240 CPUs.

NOTE: Terminals accept 16-24 AWG. For 16 AWG, use type TFFN or Type MTW. Other types of 16 AWG may be acceptable, but it really depends on the thickness of the wire insulation.





Dimensions and Installation

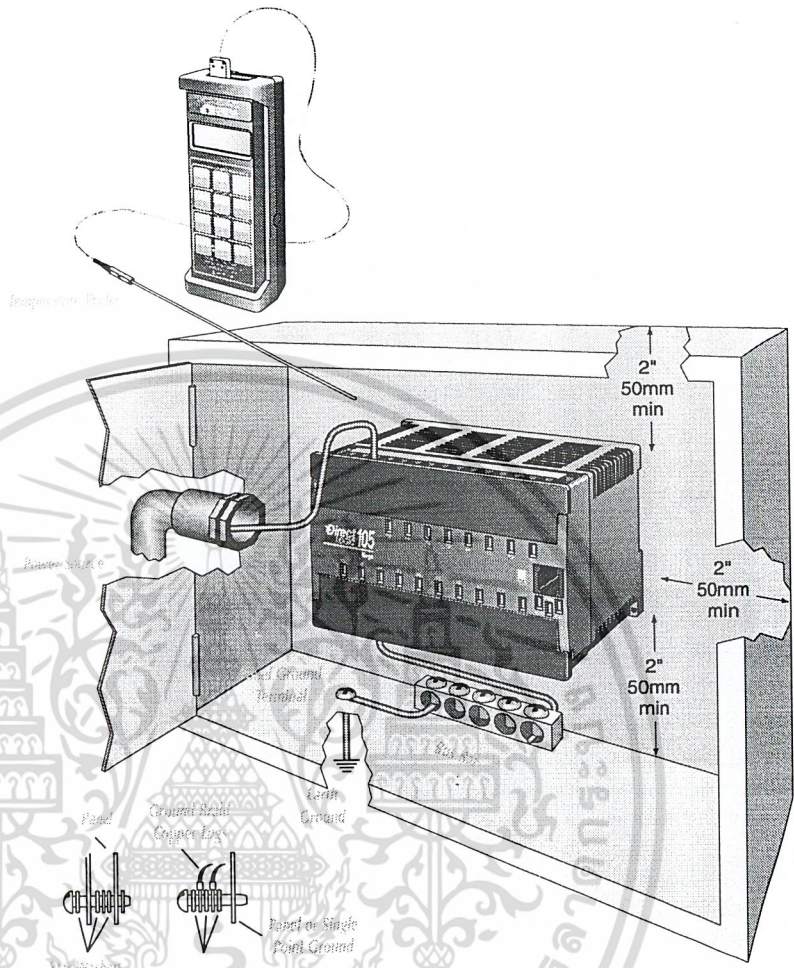
It is important to understand the installation requirements for your DL105 system. This will help ensure that the DL105 products operate within their environmental and electrical limits.

Plan for safety

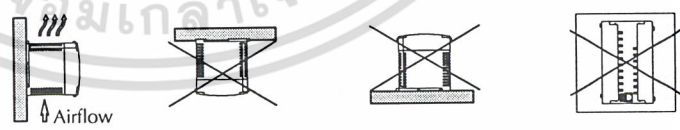
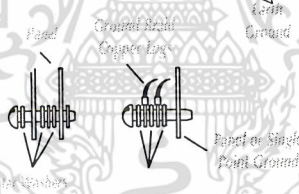
This catalog should never be used as a replacement for the user manual. The user manual, D1-USER-M, contains important safety information that must be followed. The system installation should comply with all appropriate electrical codes and standards.

Unit dimensions and mounting orientation

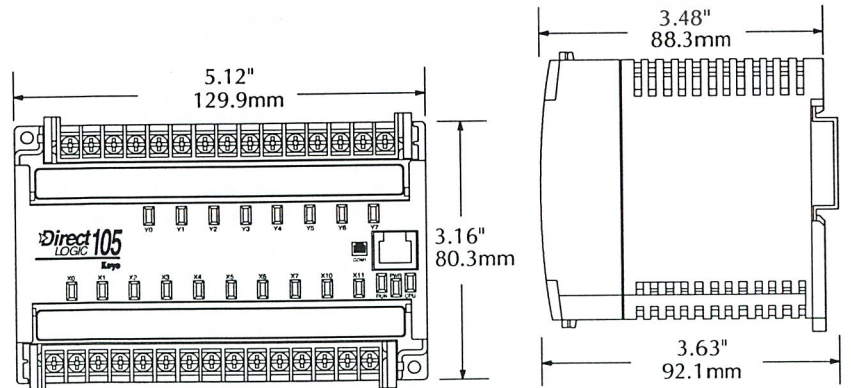
Use the following diagrams to make sure the DL105 system can be installed in your application. DL105 units must be mounted horizontally to ensure proper airflow for cooling purposes. It is important to check these dimensions against the conditions required for your application. For example, it is recommended to leave 2" depth for ease of access and cable clearance; however, your distance may be greater or less. Also, check the installation guidelines for the recommended cabinet clearances.



Note: There is a minimum of 2" (50mm) clearance required between the panel door or any devices mounted in the panel door and the nearest DL105 component.



Environmental Specifications	
Storage temperature	-4°F - 158°F (-20°C to 70°C)
Ambient operating temperature	32°F - 131°F (0° to 55°C)
Ambient humidity	30% - 95% relative humidity (non-condensing)
Vibration resistance	MIL STD 810C, Method 514.2
Shock resistance	MIL STD810, Method 516.2
Noise immunity	NEMA(ICS3-304)
Atmosphere	No corrosive gases



เมื่อการพิมพ์เอกสารที่สงวนไว้สำหรับการใช้ Toll-free orders 1-800-633-0405 อนุญาตให้นำไป www.automationdirect.com
 เมื่อการพิมพ์เอกสารที่สงวนไว้สำหรับการใช้ Toll-free orders 1-800-633-0405 อนุญาตให้นำไป www.automationdirect.com

Power supply options

This product family offers units that operate on 110/220VAC and 12/24VDC. Choosing the power supply is probably the most important consideration when specifying a DL105 system since not all I/O combinations are offered with each power supply option. The table to the right provides the I/O choices and power supply specifications for each type unit.

Power Supply Options		
Specification	AC Powered Units	24 VDC Powered Units
Part Numbers	F1-130AA, F1-130AR F1-130AD, F1-130DA F1-130DD, F1-130DR F1-DEVNET-AR, F1-DEVNET-DD F1-DEVNET-DR	F1-130DD-D F1-130DR-D
Voltage Withstand (dielectric)	one minute @ 1,500 VAC between primary, secondary and field ground	
Insulation Resistance	> 10MΩ @500 VDC	
External Power Requirement	85-132VAC (110 nominal) 170-264VAC (220 nominal) 100-264VDC (125 nominal)	10-30VDC (12 to 24VDC) with <10 percent ripple
Auxiliary 24 VDC Output	500 mA max.	Not available
Maximum Inrush Current	12A	8A
Maximum Power	30VA max.	1A (approx. 10W)

Choosing the I/O

The DL105 product family offers several different combinations of I/O points. Once you have chosen the power supply option, you need to choose the unit that offers the type of I/O points needed in your application.

Fixed I/O

All DL105 Micro PLCs have "fixed" I/O that is updated on every scan. This means that all units have 10 inputs and 8 outputs, regardless of the actual type of points on the units (DC in/Relay out, DC in/DC out, etc.) The DL105 Micro PLC is non-expandable, so you cannot add I/O points. If you are concerned about future system expansion, check our DL205 micro-modular product family. The DL205 also offers an incredible array of features and flexible I/O arrangements with several different base sizes.

Addresses are automatically assigned.

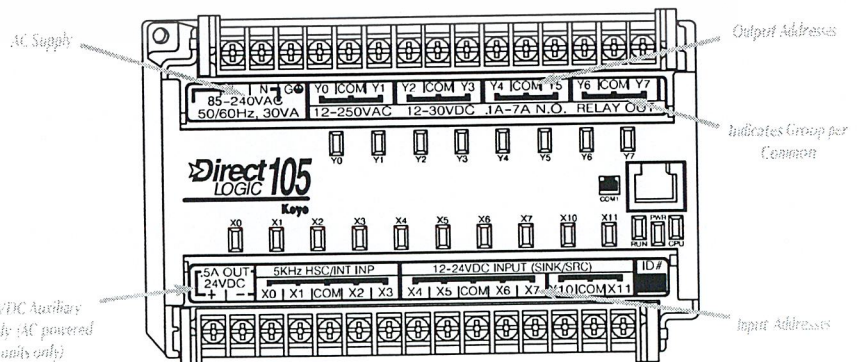
The DL105 uses automatic addressing, so for the vast majority of applications, there is no setup required. We use octal addressing for our products which means there are no 8's or 9's. The first 8 input points use addresses X0-X7, and the last two input points use X10 and X11. If you plan on using the high-speed counting features, there is some very minimal setup required in special V-memory locations.

AC powered units

Part No.	I/O Mix
F1-130AA	10 AC in
F1-130AD	8 AC out
F1-130AR	10 AC in
F1-130DA	8 relay out
F1-130DD	10 DC in
F1-130DR	8 DC out
F1-DEVNET-AR	10 AC in
F1-DEVNET-DD	8 relay out
F1-DEVNET-DR	10 DC in
	8 relay out

DC powered units

Part No.	I/O Mix
F1-130DD-D	10 DC in
F1-130DR-D	8 DC out
	10 DC in
	8 relay out





DL105 I/O Specifications

F1-130AA wiring diagram and specifications

Power requirements

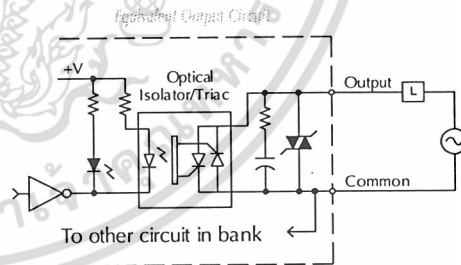
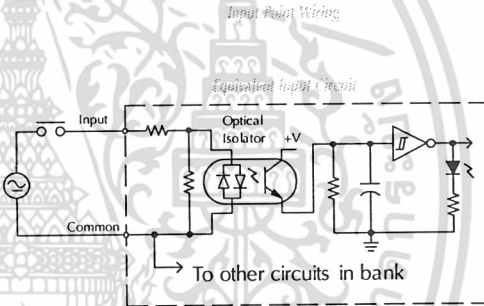
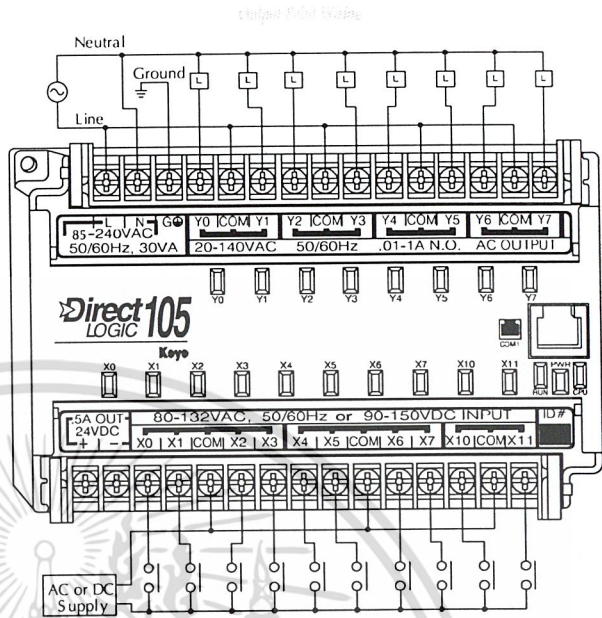
Voltage range94-240VAC (30VA)
100-240VDC (30W)

AC Input specifications

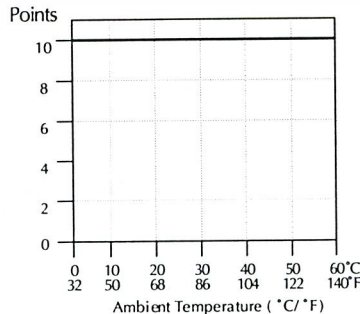
Number of input points10
 Number of commons3(isolated)
 Input voltage range80-132VAC
90-150VDC
 Input current6mA @ 132VAC
6.8mA @ 150VDC
 ON current/voltage level>4mA/80VAC
>4mA 90VDC
 OFF current/voltage level<2mA/45VAC
<2mA/60VDC
 OFF to ON response<8ms
 ON to OFF response<15ms
 FusesNone

AC Output specifications

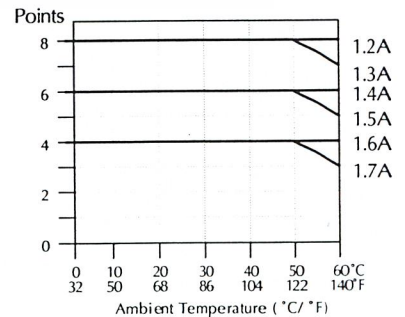
Number of output points8
 Number of commons4 (isolated)
 Output circuitryTriac
 Output voltage range20-140VAC
47-63Hz
 Peak voltage400VAC
 ON voltage drop1.3VAC at 2A
 Maximum current1.7A/point
(subject to derating)
 Maximum leakage current1mA at 400 VAC
 Maximum inrush current30A for 10ms
15A for 100ms
 Minimum load10mA
 OFF to ON response8.33ms @ 60Hz
10ms @ 50Hz
 On to OFF response8.33ms @ 60Hz
10ms @ 50Hz
 FusesNone (external recommended)



Derating Chart for AC Inputs



Derating Chart for AC Outputs



DL105 I/O Specifications



FI-130AD wiring diagram and specifications

Power requirements

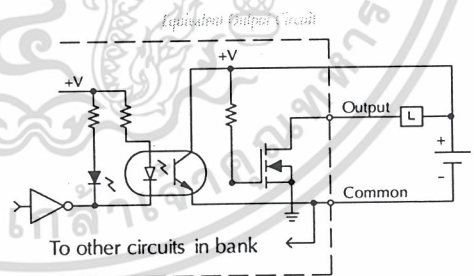
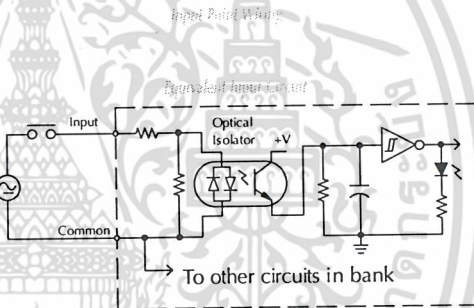
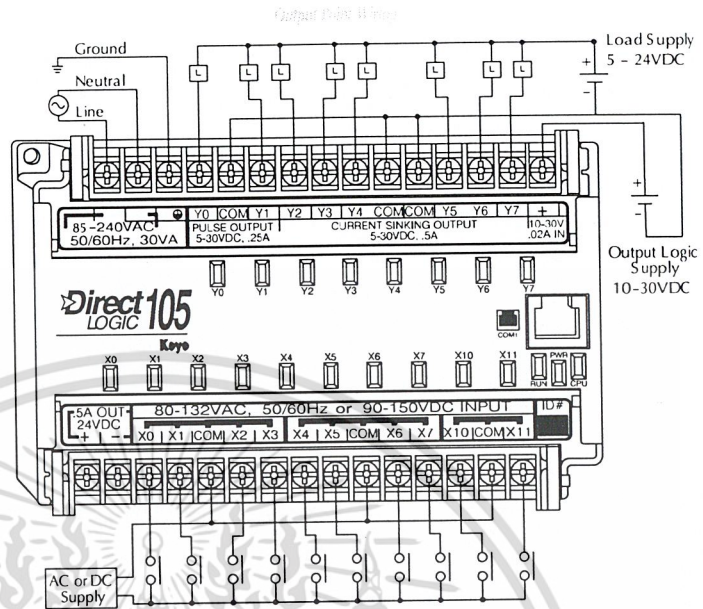
Voltage range94-240VAC (30VA)
100-240VDC (30W)

AC Input specifications

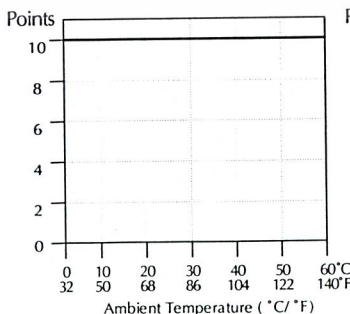
Number of input points10
 Number of commons3 (isolated)
 Input voltage range80-132VAC
90-150VDC
 Input current6mA @ 132VAC
6.8mA @ 150VDC
 ON current/voltage level>4mA/80VAC
>4mA/90VDC
 OFF current/voltage level<2mA/45VAC
<2mA/60VDC
 OFF to ON response<8ms
 ON to OFF response<15ms
 FusesNone

DC Output specifications

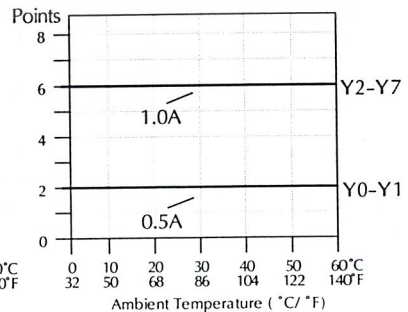
Number of output points8 (sinking)
 Number of commons3 (internally connected)
 Output circuitryMOSFET
 Output voltage range5-30VDC
 Peak voltage60VDC
 ON voltage drop0.45VDC @ 0.5A
 Maximum current0.5A/point (Y0-Y1)
1.0A/point (Y2-Y7)
 Maximum leakage current15µA at 30VDC
 Maximum inrush current1.5A for 10ms (Y0-Y1)
3A for 10ms (Y2-Y7)
0.5A for 100ms (Y0-Y1)
1A for 100ms (Y2-Y7)
 Minimum loadNone
 OFF to ON responseY0-Y1: 10µs
Y2-Y7: 3.5µs
 ON to OFF responseY0-Y1: 70µs
Y2-Y7: 110µs
 External DC power required10-30VDC
@30mA + load current
 FusesNone (external recommended)



Derating Chart for AC Inputs



Derating Chart for DC Outputs





DL105 I/O Specifications

F1-130AR wiring diagram and specifications

Power requirements

Voltage range94-240VAC (30VA)
100-240VDC (30W)

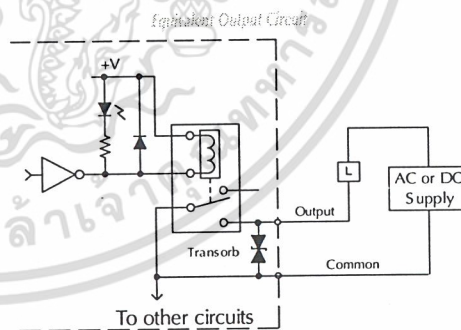
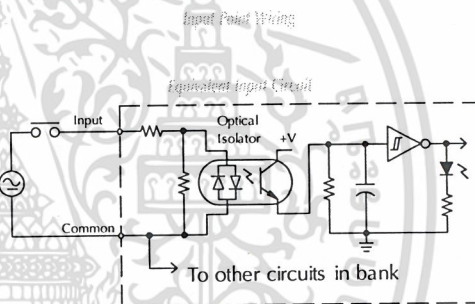
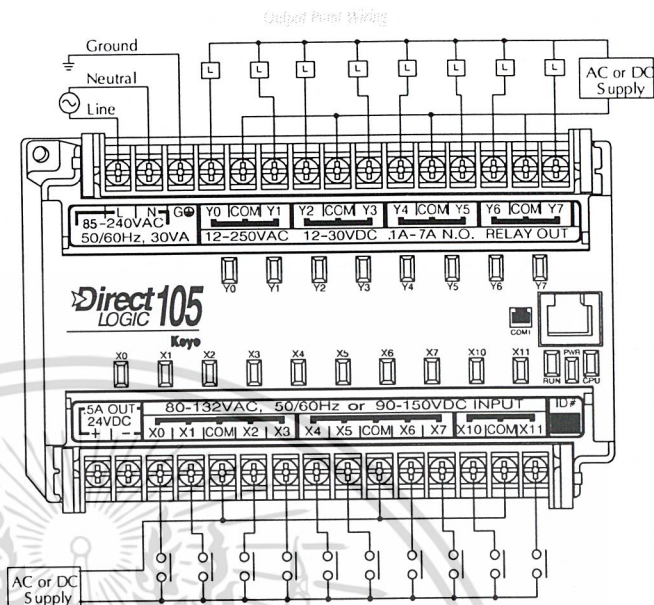
AC Input specifications

Number of input points10
 Number of commons3 (isolated)
 Input voltage range80-132VAC
90-150VDC
 Input current6mA @ 132VAC
6.8mA @ 150VDC
 ON current/voltage level>4mA/80VAC
>4mA/90VDC
 OFF current/voltage level<2mA/45VAC
<2mA/60VDC
 OFF to ON response<8ms
 ON to OFF response<15ms
 -FusesNone

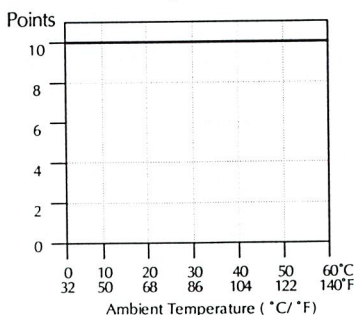
Relay Output specifications

Number of output points8
 Number of commons4 (isolated)
 Output circuitry Relay
 Output voltage range12-250VAC
12-30VDC
 Maximum voltage265VAC, 150VDC
 Maximum current7A/point (see derating)
 Maximum inrush current12A
 Minimum load10mA
 Minimum OFF resistance100Mohm @500VDC
 OFF to ON response15ms
 ON to OFF response5ms
 FusesNone (external recommended)

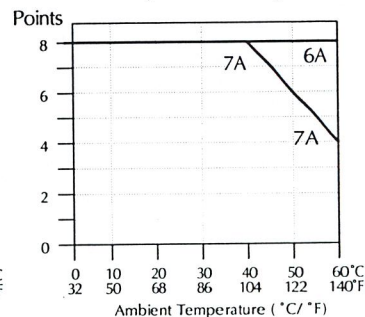
Typical Relay Life (Operations) at Room Temperature			
Voltage and Type of Load	Load Current		
	50mA	5A	7A
24 VDC Resistive	10M	600K	300K
24 VDC Solenoid	—	150K	75K
110 VAC Resistive	—	600K	300K
110 VAC Solenoid	—	500K	200K
220 VAC Resistive	—	300K	150K
220 VAC Solenoid	—	250K	100K



Derating Chart for AC Inputs



Derating Chart for Relay Outputs



DL105 I/O Specifications

F1-130DA wiring diagram and specifications

Power requirements

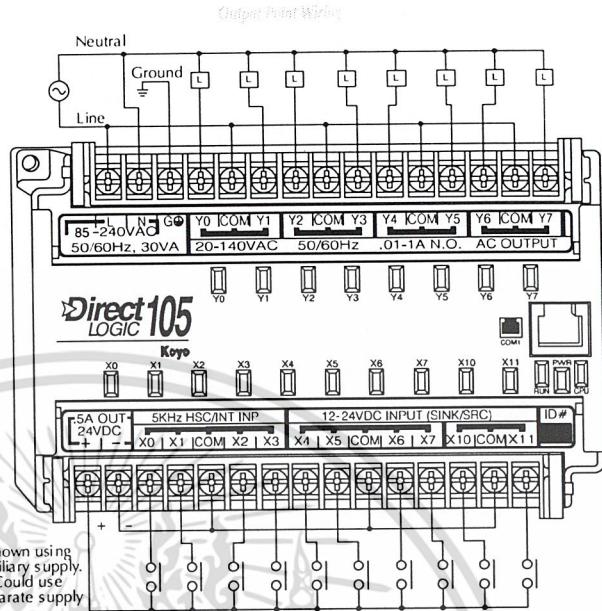
Voltage range94-240VAC (30VA)
100-240VDC (30W)

DC Input specifications

Number of input points10 (sink/source)
 Number of commons3 (isolated)
 Input voltage range(X0-X3) 10-26.4VDC
(X4-X11) 10-26.4VDC or
21.6-26.4VAC
 Input impedance2.8K @ 12VDC
2.8K @ 24VDC
 ON current/voltage level>3mA/>9VDC
 OFF current/voltage level<0.5mA/<2VDC
 ResponseX0-X3 X4-X11
 OFF to ON50µs 2-8ms
 ON to OFF50µs 2-8ms
 FusesNone

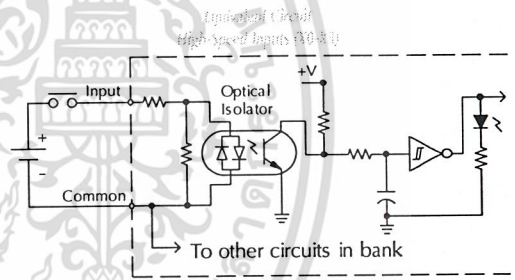
AC Output specifications

Number of output points8
 Number of commons4 (isolated)
 Output circuitryTriac
 Output voltage range20-140VAC
47-63Hz
 Peak voltage400VAC
 ON voltage drop1.3VAC at 2A
 Maximum current1.7A/point
(subject to derating)
 Maximum leakage current1mA at 400VAC
 Minimum inrush current30A for 10ms
15A for 100ms
 Minimum load10mA
 OFF to ON responseY0-Y7: 8.33ms @ 60Hz
Y2-Y7: 10ms @ 50Hz
 ON to OFF responseY0-Y7: 8.33ms @ 60Hz
Y2-Y7: 10ms @ 50Hz
 FusesNone (external recommended)

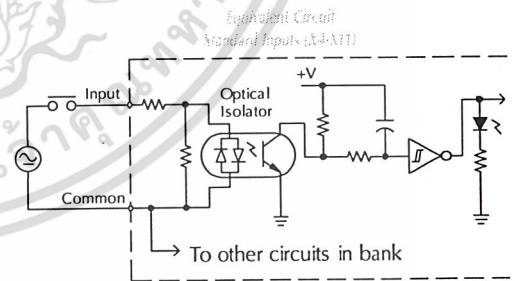


Shown using auxiliary supply. Could use separate supply

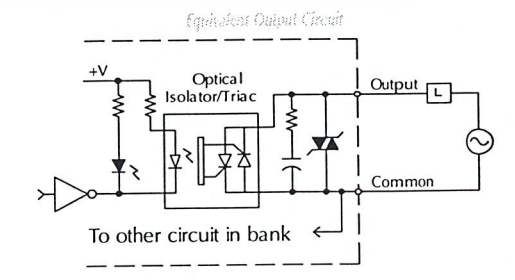
Input Point Wiring



Equivalent Circuit High Speed Inputs (X0-X3)

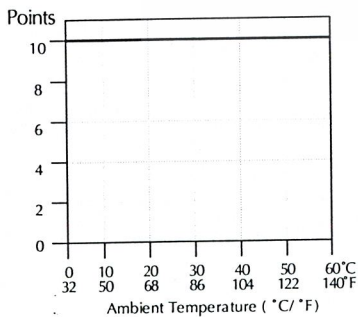


Equivalent Circuit Standard Inputs (X4-X11)

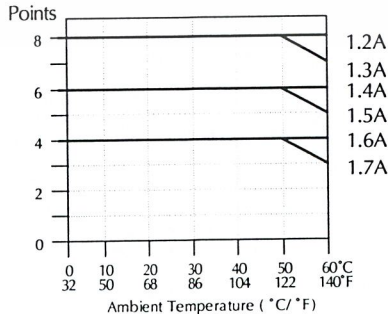


Equivalent Output Circuit

Derating Chart for DC Inputs



Derating Chart for AC Outputs





DL105 I/O Specifications

F1-130DD wiring diagram and specifications

Power requirements

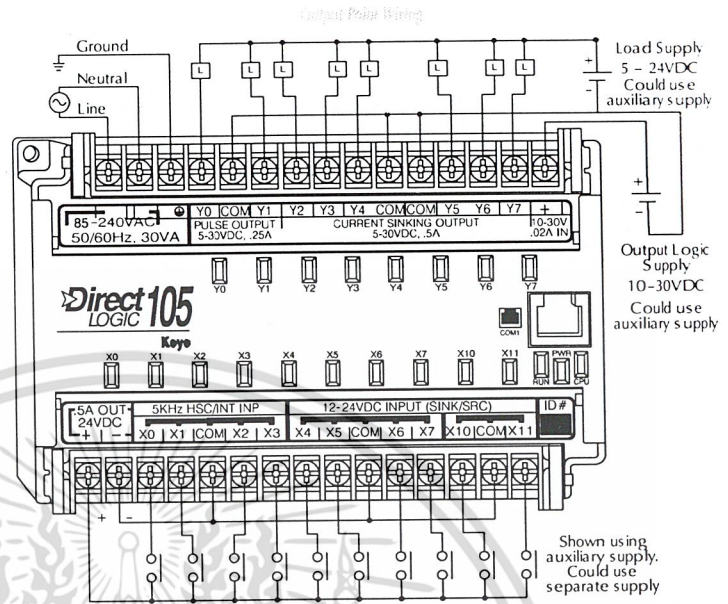
Voltage range94-240VAC (30VA)
100-240VDC (30W)

DC Input specifications

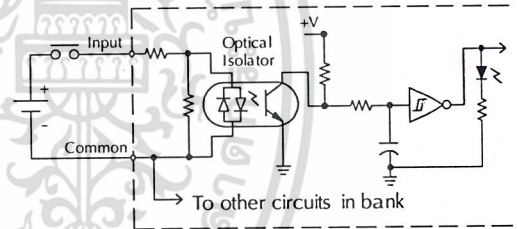
Number of input points10 (sink/source)
 Number of commons3 (isolated)
 Input voltage range(X0-X3) 10-26.4VDC
(X4-X11) 10-26.4VDC or
21.6-26.4VAC
 Input impedance2.8K @ 12-24VDC
 ON current/voltage level>3mA/>9VDC
 OFF current/voltage level<0.5mA/<2VDC
 OFF to ON responseX0-X3: 50µs
X4-X11: 2-8ms
 ON to OFF responseX0-X3: 50µs
X4-X11: 2-8ms
 FusesNone

DC Output specification

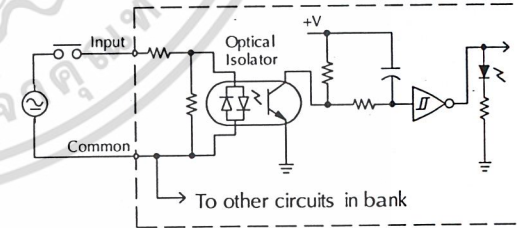
Number of output points8 (sinking)
 Number of commons3 (internally connected)
 Output circuitryMOSFET
 Output voltage range5-30VDC
 Peak voltage60VDC
 ON voltage drop0.4VDC @ 0.5A
 Maximum current0.5A/point (Y0-Y1)
1.0A/point (Y2-Y7)
 Maximum leakage current15µA at 30VDC
 Minimum inrush current
Y0-Y1: 1.5A for 10ms, 0.5A for 100ms
Y2-Y7: 3A for 10ms, 1A for 100ms
 Minimum loadNone
 OFF to ON responseY0-Y1: 10µs
Y2-Y7: 3.5µs
 ON to OFF responseY0-Y1: 70µs
Y2-Y7: 110µs
 External DC power required10-30 VDC,
@30mA + load current
 FusesNone (external recommended)



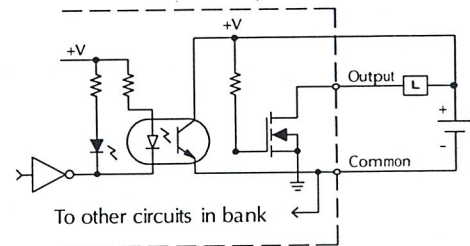
Equivalent Circuit High-Speed Input (X0-X3)



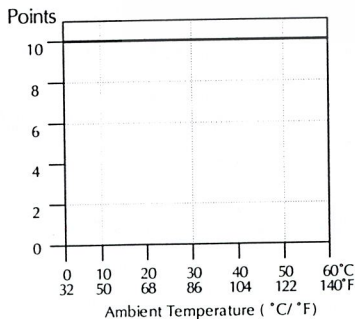
Equivalent Circuit Standard Input (X4-X11)



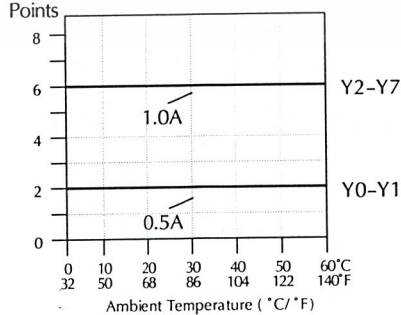
Equivalent Output Circuit



Derating Chart for DC Inputs



Derating Chart for DC Outputs



DL105 I/O Specifications

DL130DR wiring diagram and specifications

Power requirements

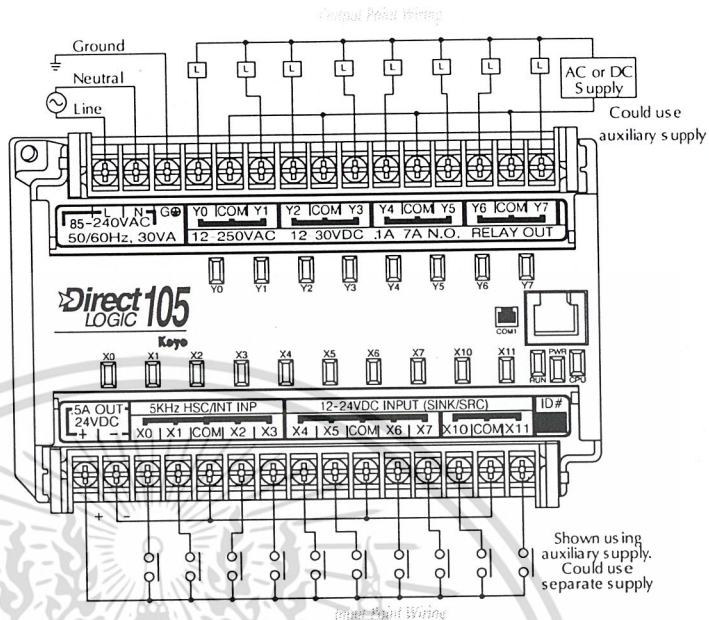
Voltage range94-240VAC (30VA)
100-240VDC (30W)

DC Input specifications

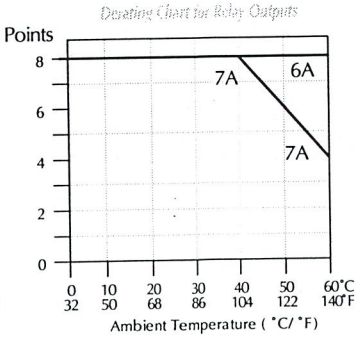
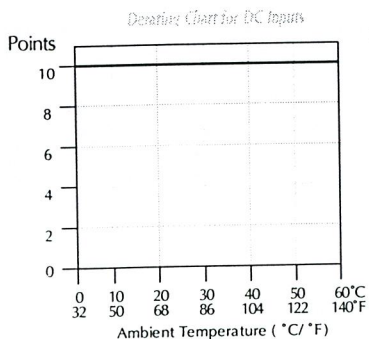
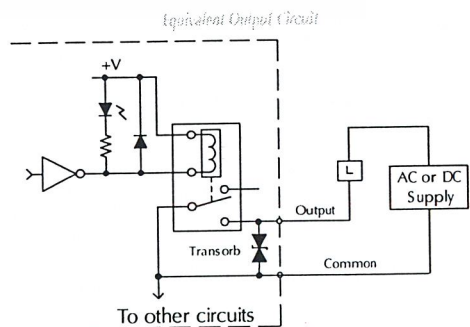
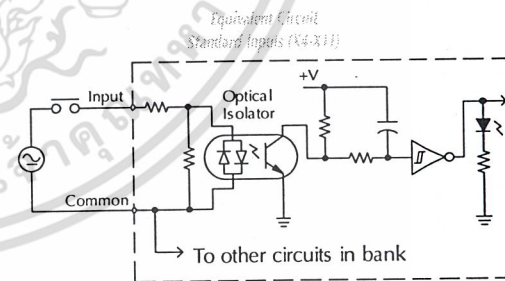
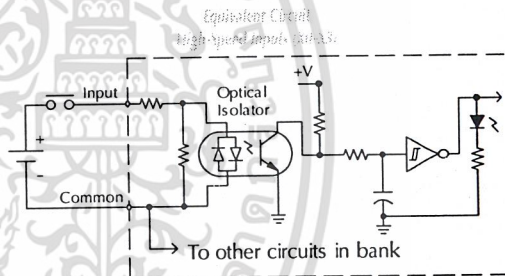
Number of input points10 (sink/source)
 Number of commons3 (isolated)
 Input voltage range(X0-X3):10-26.4VDC
(X4-X11):10-26.4VDC or
21.6-26.4VAC
 Input impedance2.8K @ 12-24VDC
 ON current/voltage level>3mA/>9VDC
 OFF current/voltage level<0.5mA/<2VDC
 OFF to ON responseX0-X3: 50µs
X4-X11: 2-8ms
 ON to OFF responseX0-X3: 50µs
X4-X11: 2-8ms
 FusesNone

Relay output specifications

Number of output points8
 Number of commons4 (isolated)
 Output circuitryRelay
 Output voltage range12-250VAC
12-30VDC
 Maximum voltage265VAC, 150VDC
 Maximum current7A/point (See derating)
 Minimum inrush current12A
 Minimum load10mA
 Minimum OFF resistance100MΩ @500VDC
 OFF to ON response15ms
 ON to OFF response5ms
 FusesNone (external recommended)



Typical Relay Life (Operations) at Room Temperature			
Voltage and Type of Load	Load Current		
	50mA	5A	7A
24 VDC Resistive	10M	600K	300K
24 VDC Solenoid	—	150K	75K
110 VAC Resistive	—	600K	300K
110 VAC Solenoid	—	500K	200K
220 VAC Resistive	—	300K	150K
220 VAC Solenoid	—	250K	100K



ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีมีการนำไปใช้



DL105 I/O Specifications

PL-130DD-D wiring diagram and specifications

Power requirements

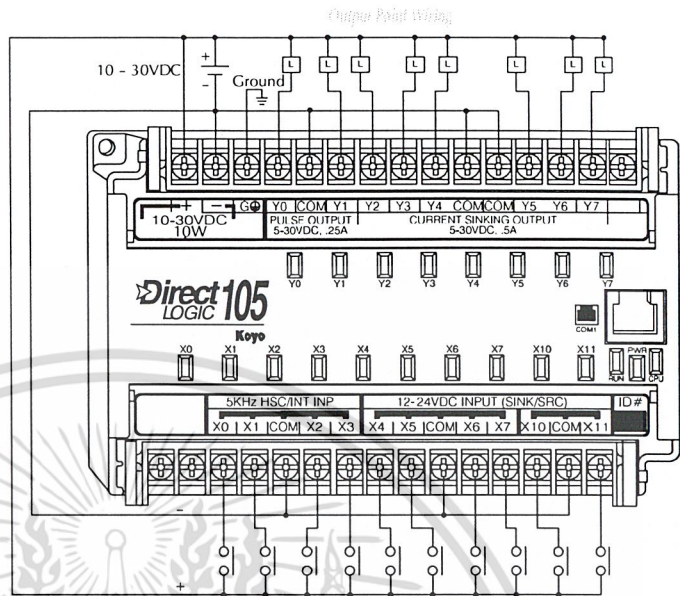
Voltage range10-30VDC
10W max.

DC Input specifications

Number of input points10 (sink/source)
 Number of commons3 (isolated)
 Input voltage range(X0-X3): 10-26.4VDC
(X4-X11): 10-26.4VDC or
21.6-26.4VAC
 Input impedance2.8K @ 12-24VDC
 ON current/voltage level>3mA/>9VDC
 OFF current/voltage level<0.5mA/<2VDC
 OFF to ON responseX0-X3: 50µs
X4-X11: 2-8ms
 ON to OFF responseX0-X3: 50µs
X4-X11: 2-8ms
 FusesNone

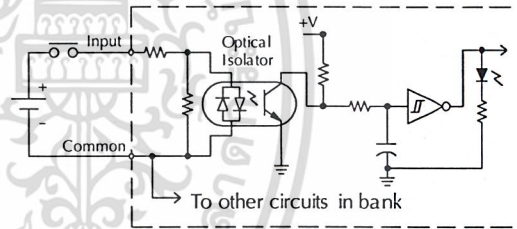
DC Output specifications

Number of output points8 (sinking)
 Number of commons3 (internally connected)
 Output circuitryMOSFET
 Output voltage range5-30VDC
 Peak voltage60VDC
 ON voltage drop0.4VDC @ 0.5A
 Maximum currentY0-Y1: 0.5A/point
Y2-Y7: 1.0A/point
 Maximum leakage current15µA at 30VDC
 Maximum inrush currentY0-Y1: 1.5A for 10ms
0.5A for 100ms
Y2-Y7: 3A for 10ms
1A for 100ms
 Minimum loadNone
 OFF to ON responseY0-Y1: 10µs
Y2-Y7: 3.5µs
 ON to OFF responseY0-Y1: 70µs
Y2-Y7: 110µs
 FusesNone (external recommended)

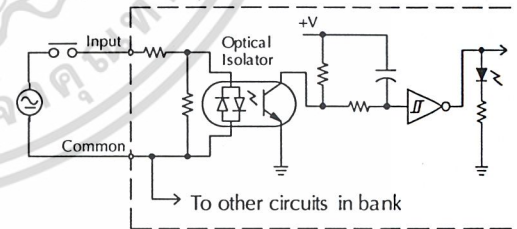


Note: Same supply can be used to power both input and output circuits because all circuits are isolated from the internal logic.

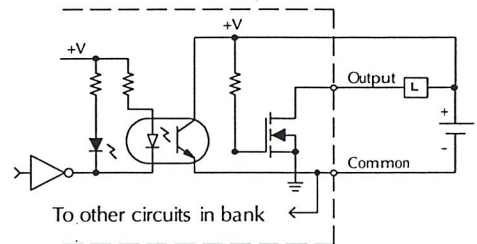
Equivalent Circuit High-Speed Inputs (X0-X3)



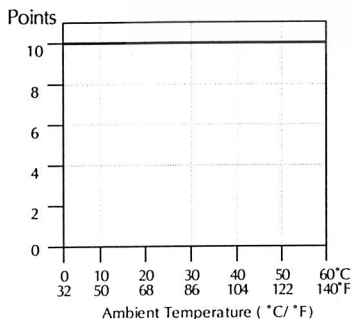
Equivalent Circuit Standard Inputs (X4-X11)



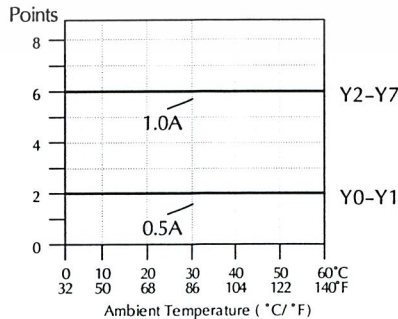
Equivalent Output Circuit



Derating Chart for DC Inputs



Derating Chart for DC Outputs



DL105 I/O Specifications

FI-130DR-D wiring diagram and specifications

DC Power supply specifications

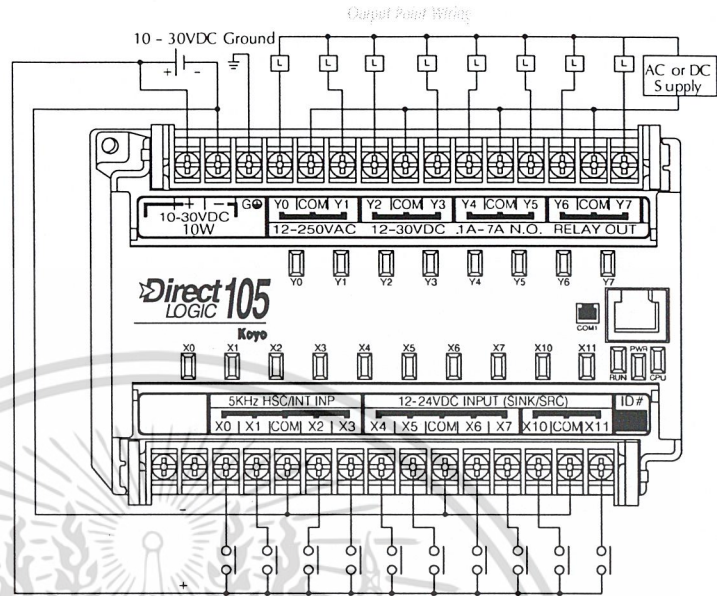
Voltage range: 10-30VDC
 Power: 10W max.

DC Input specifications

Number of input points: 10 (sink/source)
 Number of commons: 3 (isolated)
 Input voltage range: X0-X3: 10-26.4VDC
 X4-X11: 10-26.4VDC or 21.6-26.4VDC
 Input impedance: 2.8K @ 12-24VDC
 ON current/voltage level: >3mA/>9VDC
 OFF current/voltage level: <0.5mA/<2VDC
 OFF to ON response: X0-X3: 50µs
 X4-X11: 2-8ms
 ON to OFF response: X0-X3: 50µs
 X4-X11: 2-8ms
 Fuses: None

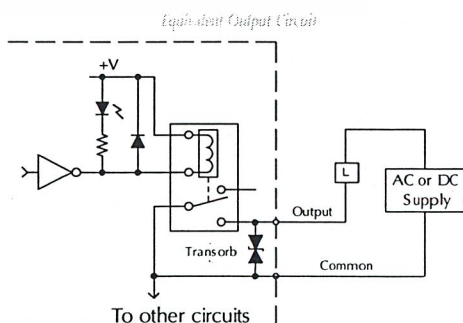
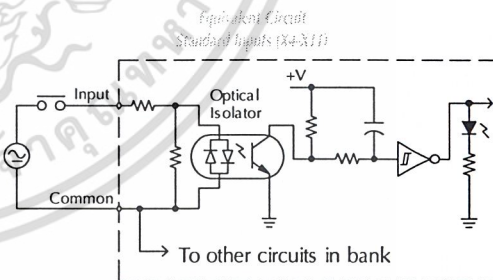
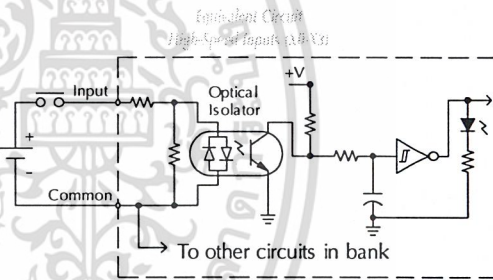
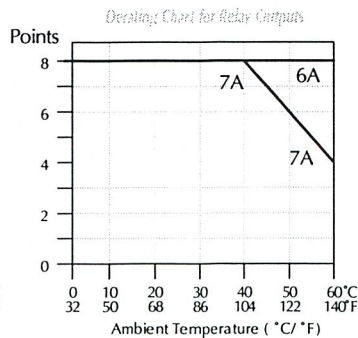
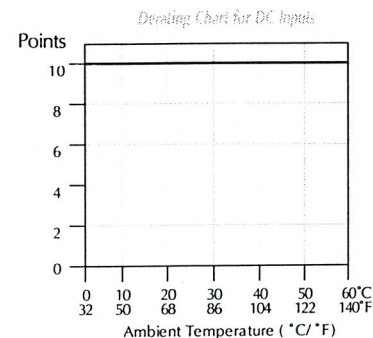
Relay Output specifications

Number of output points: 8
 Number of commons: 4 (isolated)
 Output circuitry: Relay
 Output voltage range: 12-250VAC
 12-30VDC
 Maximum voltage: 265VAC, 150VDC
 Maximum current:7A/point (See derating)
 Minimum inrush current: 12A
 Minimum load:10mA
 Minimum OFF resistance: 100MΩ @ 500VDC
 OFF to ON response:15ms
 ON to OFF response:5ms
 Fuses: None (external recommended)



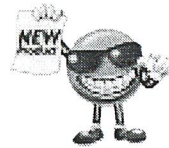
Note: Same supply can be used to power both input and output circuits because all circuits are isolated from the internal logic.

Typical Relay Life (Operations) at Room Temperature			
Voltage and Type of Load	Load Current		
	50mA	5A	7A
24 VDC Resistive	10M	600K	300K
24 VDC Solenoid	—	150K	75K
110 VAC Resistive	—	600K	300K
110 VAC Solenoid	—	500K	200K
220 VAC Resistive	—	300K	150K
220 VAC Solenoid	—	250K	100K





DeviceNet I/O Specifications

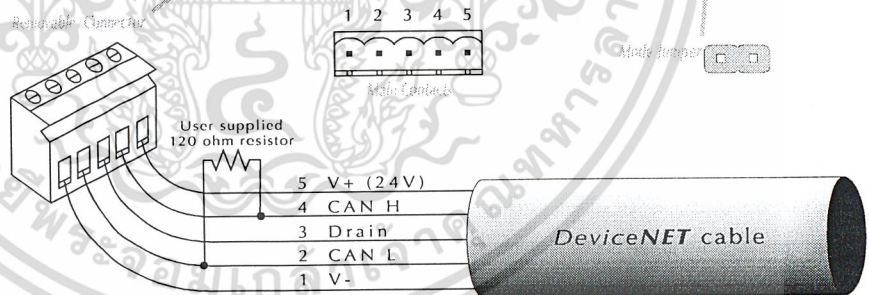
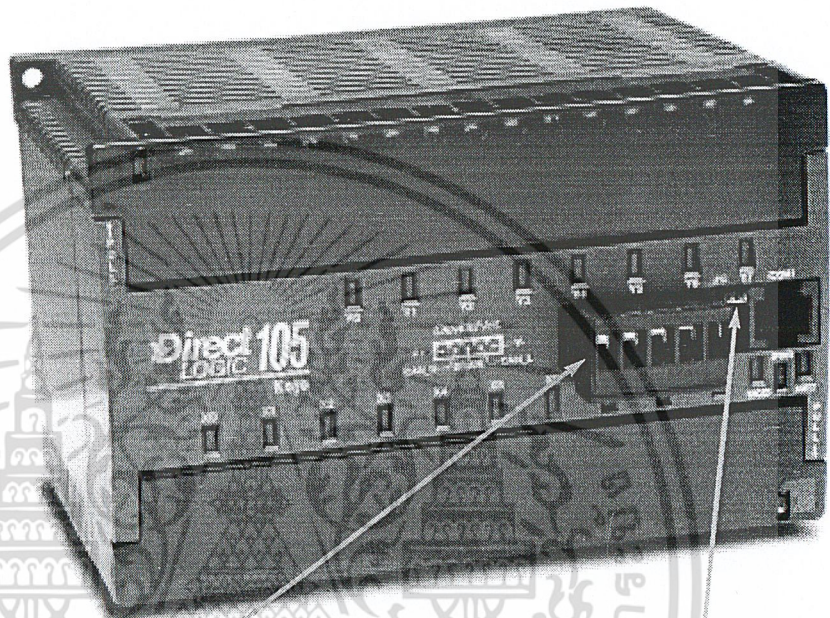


DL105 DeviceNet slave



The DL105 DeviceNet slave allows you to distribute the same I/O found on the DL105 PLCs across your DeviceNet control system. DeviceNet is designed to reduce the need for hardwiring while providing device-level diagnostics. This industrial protocol links up to 64 nodes on a single network. All F1-DVNET modules have a removable connector that makes the five-wire DeviceNet connection easy to implement and maintain. The DeviceNet module incorporates advanced diagnostics not commonly found on traditional industrial networks. This module has the quick response time and high dependability expected out of any DeviceNet apparatus. The F1-DVNET modules have the same I/O configurations and specifications as their DL105 counterparts. The F1-DVNET modules also support polling and explicit messaging.

DL105 DeviceNet Modules	
F1-DVNET-AR	10 AC inputs; 8 relay outputs
F1-DVNET-DD	10 DC inputs; 8 DC outputs
F1-DVNET-DR	10 DC inputs; 8 relay outputs



High-speed I/O

The F1-DVNET modules have four high-speed operating modes that run in concert with the DeviceNet commands to provide enhanced I/O control. The available modes are:

- Mode 10¹ Counter mode
- Mode 20² Quadrature counter
- Mode 30 Pulse output with profile modes
- Mode 60³ Normal I/O mode (Filtered inputs and pulse catch input)

1. Must be DC Input
2. Must be DC Output
3. Mode 60 includes mode 50. Input points X0-X2 are affected by these modes. All points not directly used in Mode 60 may be filtered if desired.

Trunk Length		Bits per sec	Branch Length	
Feet	Meters		Feet	Meters
328ft	100m	500Kbps	20ft	6m
820ft	250m	250Kbps	20ft	6m
1,640ft	500m	125Kbps	20ft	6m

Other DeviceNet specifications, compatible products, and latest DeviceNet information are made available through:
 Open DeviceNet Vendor Association
 Contact: William H. (Bill) Moss, Executive Director
 Phone: (954) 340-5412 Fax: (954) 340-5413
 Internet Address: <http://www.odva.org>
 e-mail: odva@powerinternetwork.com
 ODVA, Inc.
 20423 State Road 7
 Boca Raton, FL 33498

Four-Point Simulator

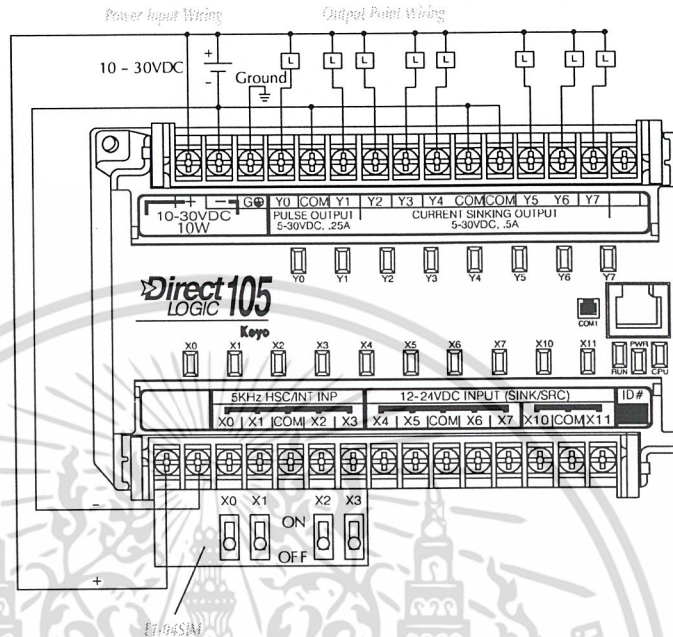
F1-04SIM wiring diagram and specifications

The F1-04SIM is a simple 4-point simulator that can be used with DC input versions of the DL105 Micro PLCs. It uses input points X0-X3 and is great for testing purposes or actual use in an application.

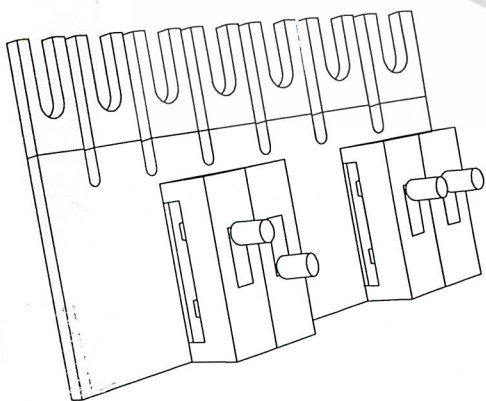
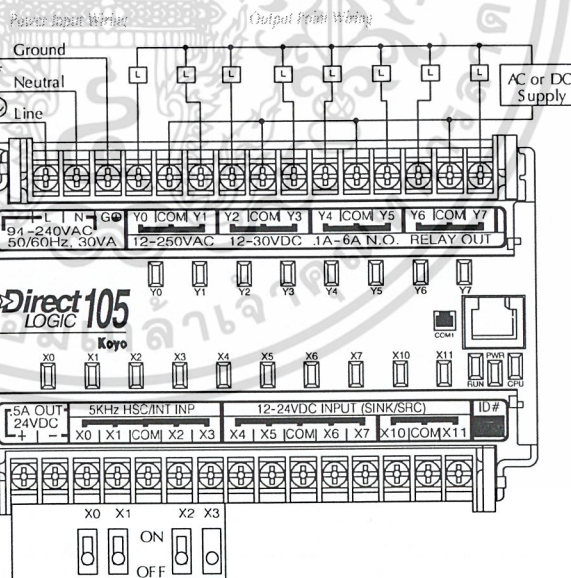
Please note, you cannot use this simulator with units that have AC discrete input points.

The simulator is a single circuit board that simply slides underneath the screw terminals on the DL105 Micro PLC. One advantage with this simulator is that power is obtained directly from the auxiliary 24VDC supply located on the input terminal strip. So for most applications, the task is extremely simple. If you are using an F1-130DD-D or F1-130DR-D, then you have to jumper the power input before you can use the simulator. This is because the DC-powered units do not offer this auxiliary supply.

To use with DC powered units, simply connect the input power wiring to the unused terminals normally occupied by the 24VDC Auxiliary supply.



For AC powered units, there are no extra wiring connections. Power is obtained directly from the 24VDC Auxiliary supply.



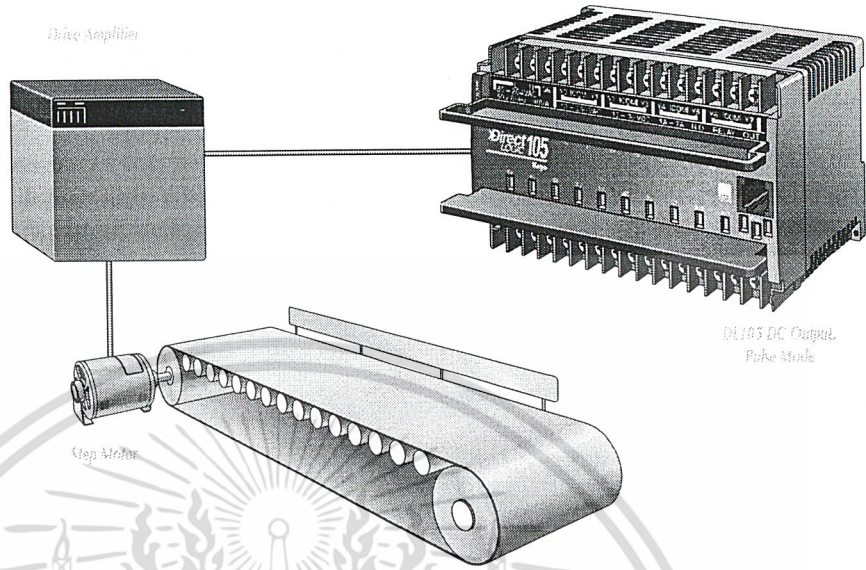
Selected DL105 Micro PLCs offer special high-speed input features (on units with DC inputs) and pulse output features (on units with DC outputs). These features are available on the first four input points (Y0-Y3) and the first two output points (Y0-Y1). This allows you to use the economical DL105 Micro OLS to solve an incredibly diverse range of motion and high-speed machine control applications.

There are several modes of operation from which to choose. Here's a brief description of the modes provided.

- Single 5kHz high-speed counter with 24 presets. When the preset is reached, an interrupt routine is executed.
- Single quadrature encoder input (up/down counter) for clockwise and counterclockwise position control.
- Single channel programmable 7kHz pulse output with an external interrupt and separate acceleration/deceleration profiles for positioning and velocity control.
- A single external interrupt input for an immediate response to time-critical tasks.
- Single pulse catch input allows the CPU to read an input with a pulse width as small as 0.1ms.
- Four inputs with selectable filters (0-99ms) to ensure input signal integrity. This is the default mode, which is set at 10ms filter.
- A single timed interrupt that can be scheduled on a 5ms-999ms cycle. (All units have this feature.)
- Combine features to use the full potential of the module. Some modes do not use all available points, so in some cases you can

assign one of the other features to the point(s) not used by the main mode of operations. You cannot use the DL105 for closed-loop control. **You cannot use the Up counter and pulse output features at the same time.**

You can easily select the mode of operation just by entering an appropriate "code" in a special CPU V-memory location. These features are explained in more detail later in this section. Remember, not all features can be used at the same time. The Counter Mode Options table provides point-by-point usage for each mode of operation.



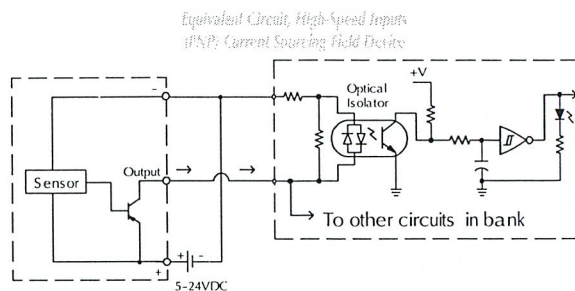
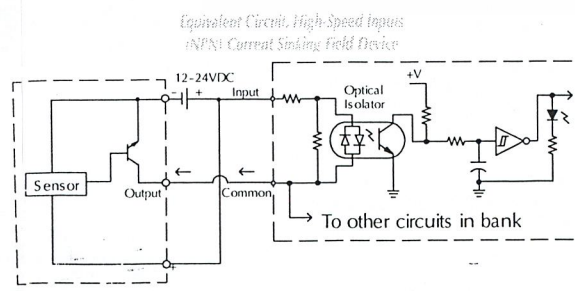
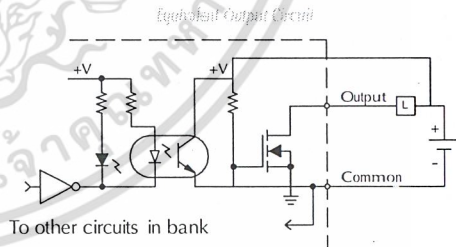
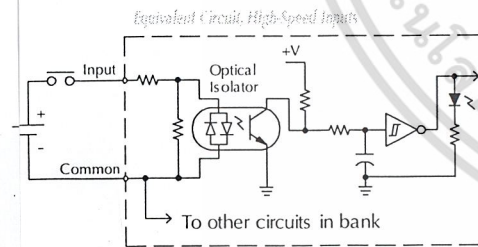
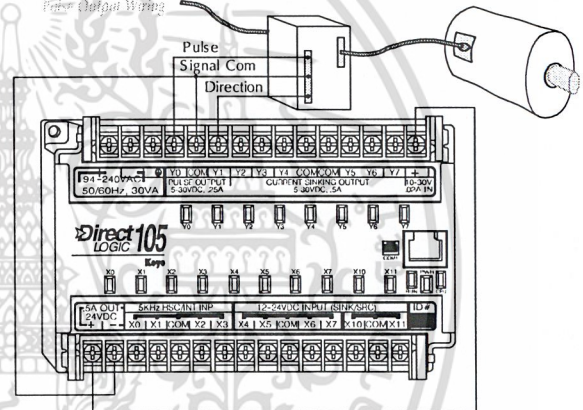
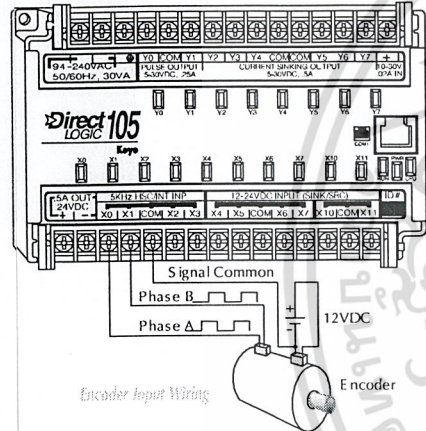
Counter Mode Options

Mode	DC Input Points				DC Output Points	
	X0	X1	X2	X3	Y0	Y1
Filtered Input	Filtered Input	Filtered Input	Filtered Input	Filtered Input	Regular Output	Regular Output
Up Counter	Count Input	Filtered Input	Filtered Input, or Counter Reset	Filtered Input	Regular Output	Regular Output
Up/Down Counter	Phase A Input	Phase B Input	Filtered Input, or Counter Reset	Filtered Input	Regular Output	Regular Output
Interrupt Input	Interrupt Input	Filtered Input	Filtered Input	Filtered Input	Regular Output	Regular Output
Pulse Catch	Pulse Catch	Filtered Input	Filtered Input	Filtered Input	Regular Output	Regular Output
Pulse Output	Not available for use	Filtered Input	Filtered Input, or Interrupt to Trigger Pulse Output	Filtered Input	Pulse or CW Output	Direction or CCW Output
Timed Interrupt	Filtered Input	Filtered Input	Filtered Input	Filtered Input	Regular Output	Regular Output

High-Speed I/O Specifications

High-Speed Input Specifications	
Inputs	4 pts. Max., X0-X3, sink or source 5kHz Max.
Minimum pulse width	100 μ Sec
Input voltage range	10-26.4 VDC
Input impedance	3.0K @12VDC 2.8K @24VDC
ON current/voltage level	>3mA/>9VDC
OFF current/voltage level	<0.5mA/<2VDC
OFF to ON response	<50 μ s
ON to OFF response	<50 μ s

High-Speed Output Specifications	
Outputs	2 pts. Max., Y0&Y1 current sinking, 7kHz Max.
Voltage range	5-30VDC
Maximum load current	0.5A/point
ON voltage drop	0.45VDC @0.5A
Leakage current	15 μ A @30VDC
Inrush current	1.5A (10ms) 0.5A (100ms)
OFF to ON response	Less than 50 μ Sec
ON to OFF response	Less than 50 μ Sec



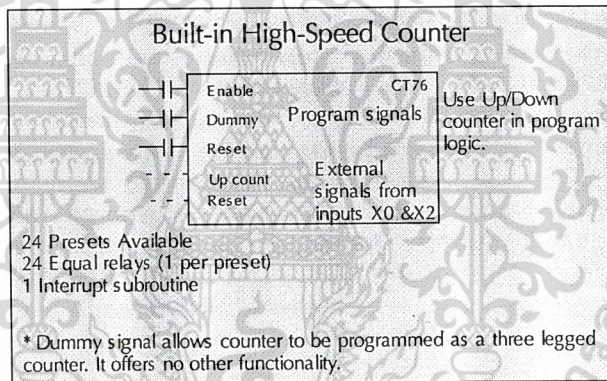
Overview

Each DL105 Micro PLC with DC inputs has internal features embedded that support a single high-speed counter up to 5kHz. You connect the external pulse input and reset input signals to the internal counter by using the first 4 discrete input points (X0-X3). The embedded counters are independent of the Micro PLC's ladder logic execution, so counting is not affected by the scan time. When the counter reaches any one of up to 24 preset values, the Micro PLC stops executing the main RLL program and executes a special interrupt subroutine that is associated with the UP counter (the CPU resumes normal operations from where it left off after the interrupt subroutine is finished). You can program the subroutine with any of the instructions that are normally available in subroutines. Also, each preset value has a corresponding "Equal" relay. These are individual internal control relays that are turned on when the asso-

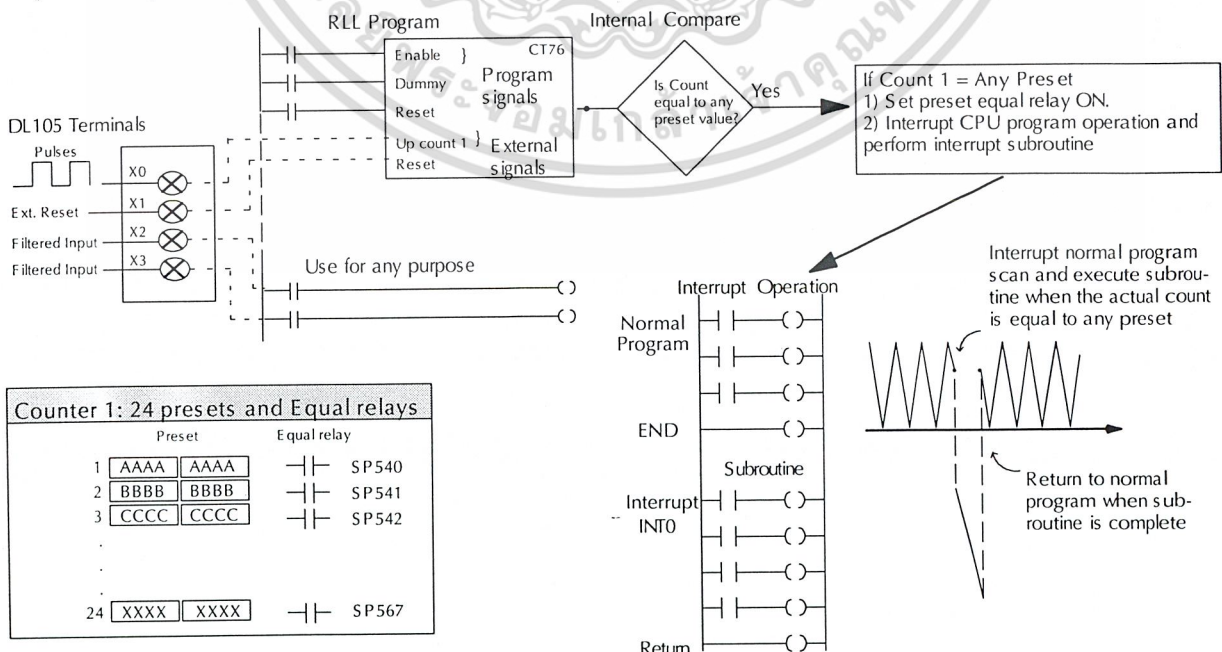
ciated preset matches the actual count. This allows you to easily trigger actions based on the current count. For example, you could use Immediate I/O instructions to provide a fast response. Using the counter in your program is fairly simple. An up/down counter box is used, and you can start and stop the counter just by turning an enable contact (of your choice) on and off as needed. Counters can be reset either by an external signal (X2) or by special internal relays that can be activated by the program. Presets are absolute, which means they are compared directly to the actual count.

Input Assignments for Up Counter Mode
 X0: Up count of UP counter
 X1: Filtered Input
 X2: External Counter reset (or filtered input)
 X3: Filtered Input

Input Specifications
 Input voltage 12 or 24 VDC
 Frequency 5kHz maximum
 Minimum pulse width 100 µSec
 Maximum count 99,999,999
 Preset types Absolute
 Number of presets 24 per counter



Example Operation



Mode 20: Single Quadrature Counter

Overview

By selecting Mode 20, you can use the high-speed input features to connect up to a 5kHz quadrature encoder. In this mode, you can have two external pulse inputs from the encoder (Phase A and Phase B) and one reset input signal. These are connected to the DL105 Micro PLC at points X0, X1, and X2 respectively (X2 can be used as an external reset or as a discrete filtered input). In addition to the physical inputs, there are also two internal references used in the control program, a counter enable input, and a counter reset input.

Note: you cannot use two individual encoders as the input devices with the DL105 Micro PLC.

Like the UP counter, the quadrature counter is independent of the CPU ladder logic execution, so the actual pulse counting is not affected by the scan time. However, the quadrature counter cannot trigger an interrupt based on the current count/preset relationship. To perform simple positioning or to control output devices, you must use relational

contacts (based on the current count) within your RLL program.

Since these contacts are within the RLL program, the resolution obtained with this method is actually limited by the Micro PLC scan time. That is, the margin for error is equal to the maximum number of pulses that could be expected during one scan. You can determine the resolution with a simple formula:

$$\text{Pulses per Scan} = \text{Scan Time (ms)} \times \text{Frequency (kHz)}$$

For example, a 10ms scan and a 5kHz encoder input yields a maximum of 50 pulses per scan. So, the maximum positioning precision would be the amount of encoder revolution that yields

Input Assignments for Quadrature Counter Mode

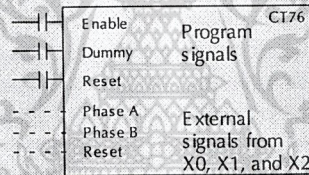
- X0: Phase A
- X1: Phase B
- X2: External Counter reset (or filtered input)
- X3: Filtered Input

Input Specifications

- Input voltage: 12 or 24VDC
- Frequency: 5kHz maximum
- Minimum pulse width: 100µs
- Count range: 0 to 99,999,999 unipolar
8388608 to 8388607 bipolar
- Number of presets: None, use relational contacts or use CT76 status contact

50 pulses. The amount of precision will also depend on the field device delay, Micro PLC output off/on delay, etc. This amount of precision may be acceptable for many simple positioning applications. If you need additional flexibility for your applications, check out our DL205 micro modular family. The powerful D2-250 CPU and inexpensive D2-CTRINT may be just what you need.

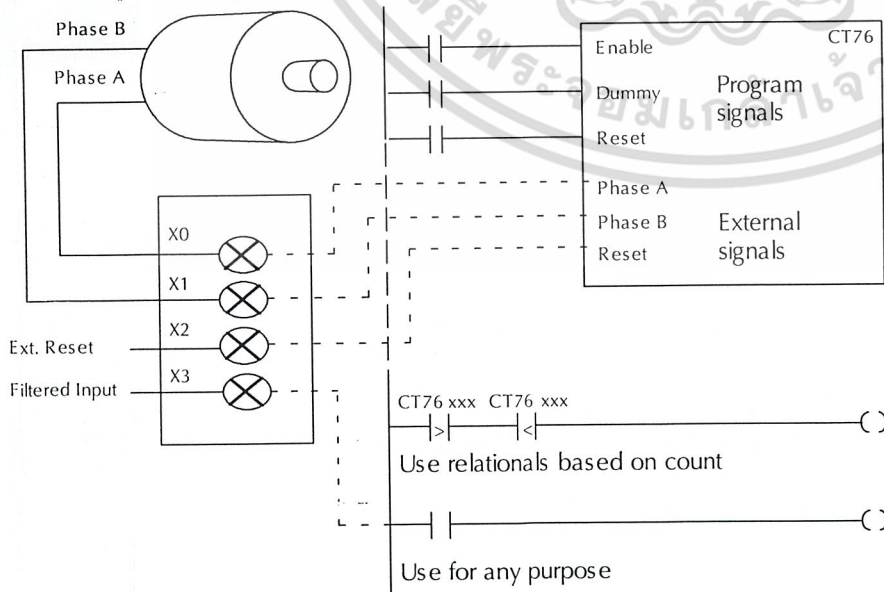
Built-in Quadrature Counter



Use Up/Down counter in program logic.

* Dummy signal allows counter to be programmed as a three legged counter. It offers no other functionality.

Example Operation





Mode 30: Pulse Output

Overview

By selecting Mode 30, you can use the pulse output feature to build simple motion and positioning control systems. Transfer and indexing tables are common applications. There are two operation profiles available (shown below). You choose the profile and motion parameters by using special CPU V-memory locations that are reserved for use by the high-speed I/O features. You can configure the pulse output for independent CW/CCW pulse train output, or step and direction. With independent operation, Y0 is the CW pulse output and Y1 is the CCW pulse output. If you choose step and direction, Y0 is the pulse train output and Y1 controls the CW/CCW operation (OFF/ON respectively). In either case, the pulses are sent out independently of the CPU scan, so scan time does not affect the pulse generation. The pulse output is enabled through ladder logic by activating Y0.

Once you have specified these parameters, the DL105 Micro PLC automatically controls the actual acceleration/deceleration velocity and pulse output.

Acceleration/deceleration times can be in the range of 100ms to 10 seconds. This profile also allows you to perform simple registration. By using the external interrupt (X2), you can delay counting toward the target number of pulses until the interrupt occurs.

Velocity control

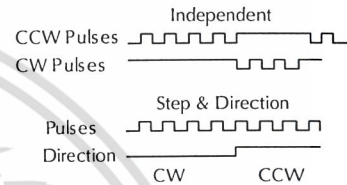
You can also choose a velocity-only mode. In this scenario, you only control the velocity. That is, there is no target destination (number of pulses). You simply change the velocity value as necessary to achieve the desired results.

Input Assignments for Pulse Output Mode

- X0: Not available (used internally as position complete)
- X1: Filtered input
- X2: Filtered input or positioning interrupt
- X3: Filtered input

Output Assignments for Pulse Output Mode

- Y0: Independent mode, CW pulse output step & direction mode, pulse output
- Y1: Independent mode, CCW pulse output Step & direction mode, OFF=CCW, ON=CCW

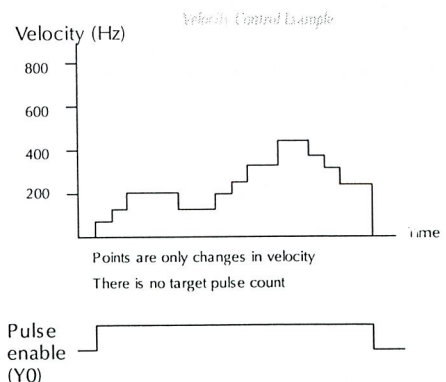
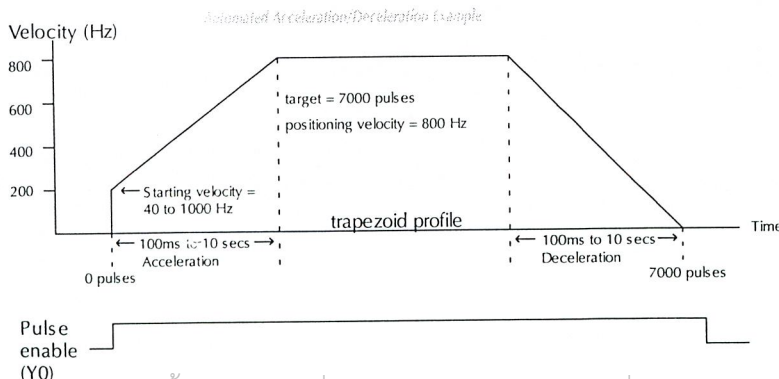
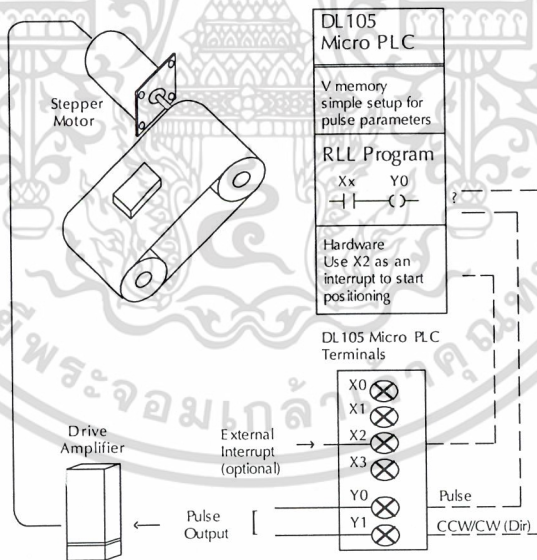


Output Specifications

- Output voltage 5-30VDC
- Frequency 7KHz maximum
- Target pulse range -8,388,608 to 8,388,607
- Velocity range 40 to 7000 pulses/sec (in units of 10 pulses)

Automatic acceleration/deceleration

The trapezoid profile is also referred to as the **Automatic Acceleration Profile**. You simply specify a target destination (number of pulses), a starting velocity (pulses per second), a positioning velocity, an acceleration time, and a deceleration time.



Mode 40: External Interrupt

Overview*

By selecting Mode 40, you can use X0 as a high-speed interrupt input. An interrupt input is especially useful in applications that have a high-priority event which requires special operations to be performed. When this high-priority event occurs, the interrupt input senses an ON input signal. The input automatically sends an interrupt request to the CPU. The CPU immediately suspends its routine scan cycle execution and jumps to a subroutine which is associated with the interrupt input point. You can program the subroutine with any of the instructions that are normally available in subroutines.

For example, you could use immediate I/O instructions to immediately read inputs and update outputs without waiting on the normal I/O update cycle. When the subroutine is complete, the CPU automatically resumes the normal scan cycles starting at the exact location from where it was interrupted. The CPU continues the routine scan until another interrupt signal is sensed.

A note on timed interrupt

If you use the external hardware interrupt (mode 40), you cannot use the internal timed interrupt. You can use either hardware interrupt or a timed interrupt, but not both. This is because they both share the same interrupt routine, INTO.

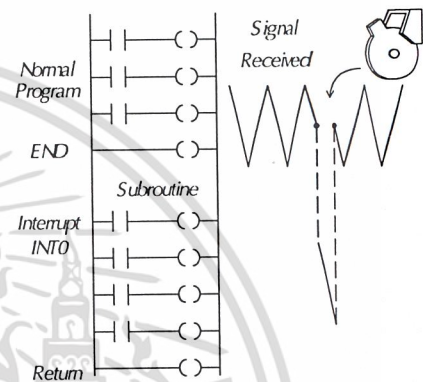
* Mode 40 is not available on the F1-DVNET modules.

Input Assignment for Interrupt Mode

X0: Interrupt input
 X1: Filtered Input
 X2: Filtered Input
 X3: Filtered Input

Input Specifications

Input voltage 12 or 24VDC
 Minimum pulse width 100 μSec
 Trigger Leading edge
 Interrupt subroutine INTO



Mode 50: Pulse Catch Input

Overview*

By selecting Mode 50, you can use X0 as a Pulse Catch input. In this configuration, the DL105 Micro PLC can capture very fast (narrow) pulse inputs that cannot normally be detected during the normal input update cycle. You can detect pulse widths as small as 0.1ms (and a pulse period greater than 0.5ms). When an external pulse is encountered, X0 is set ON for the next CPU scan,

and then it is automatically set to the OFF state. The special purpose relay remains ON for the next CPU scan and then it is automatically set to the OFF state. Like the other modes, the pulse catch feature operates independently of the CPU scan and is not affected by scan time fluctuations. *Mode 50 is not recommended for high-speed pulse counting.*

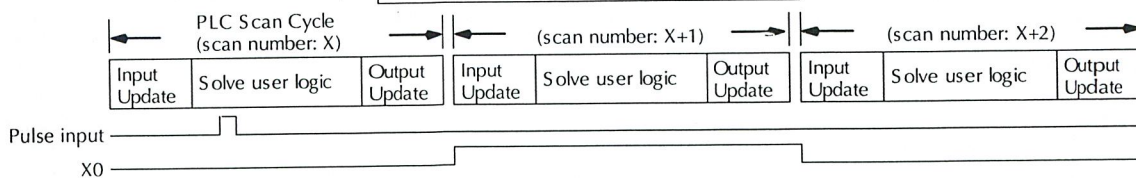
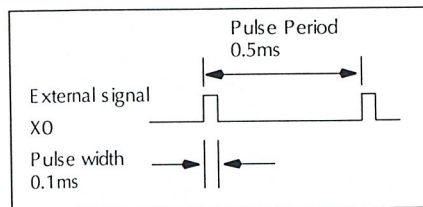
* Mode 50 is combined with Mode 60 on the F1-DVNETS modules.

Input Assignments for Pulse Catch Mode

X0: Pulse catch input
 X1: Filtered input
 X2: Filtered input
 X3: Filtered input

Input Specifications

Input voltage 12 or 24VDC
 Minimum pulse width 100 μSec
 Pulse period 0.5ms or greater
 Trigger Leading edge





Mode 60: Four Discrete Inputs with Filter

Overview*

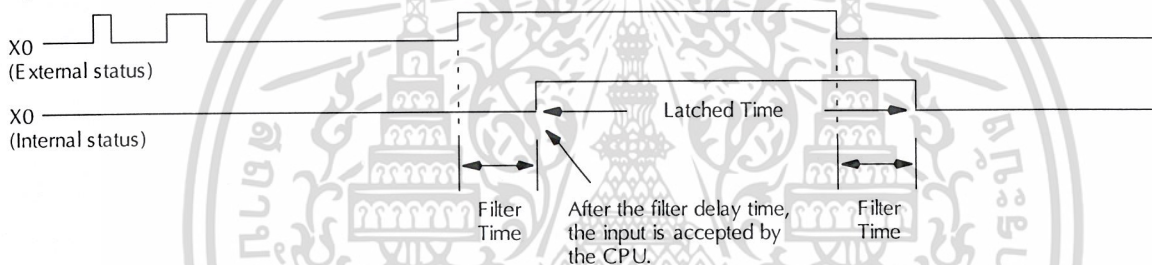
Mode 60, which is the default mode set at the factory, provides selectable filtering for the first four inputs (X0-X3). Filtering can be especially useful because it reduces the possibility of false ON conditions (which can in turn trigger events in your ladder logic program). When an external signal is first detected (ON state), a programmable filter is activated which begins a timed countdown. The slight delay temporarily prevents the CPU from reading the input during the normal input update portion of the scan cycle.

The ON signal must stay present long enough for the filter to time out. If the ON signal stays present during the entire filter time, it is latched by the filter and allowed to be accepted by the CPU during the CPU's normal input update portion of the scan cycle. The signal is latched for the remaining duration of the ON signal plus an amount of time equal to the filter time. The filter time can be programmed from 0 to 99ms in 1ms increments (default is 10ms).

* Mode 50 is combined with Mode 60 on the F1-DVNETS modules.

Input Assignments for Filtered Input Mode

Point assignments: 4 inputs (X0, X1, X2, X3)
Filter time: Programmable from 0-99ms
. in 1ms increments



Understanding the Timed Interrupt

Overview

There is also a timed interrupt feature available in the DL105 Micro PLCs. This cyclical interrupt allows you to easily program a time-based interrupt that occurs on a scheduled basis. This feature is available in all units, regardless of input type. The CPU's timed interrupt operates in a manner similar to the external interrupt input, but instead of the interrupt subroutine being triggered by an external event tied to X0, it is now triggered by a cyclical interval of time. This interval can be programmed from 5ms to 999ms. Whenever the programmed time elapses, the CPU immediately suspends its routine scan cycle and jumps to interrupt subroutine INT0. As with the other modes, when the subrou-

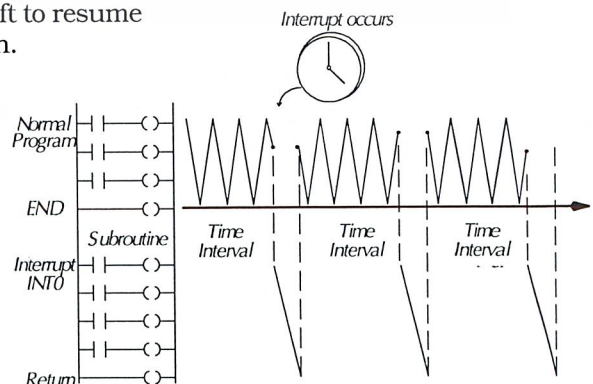
tine execution is complete, the CPU automatically resumes its routine scan cycle starting from the exact location where it was interrupted. Since the CPU scan time and the interrupted time interval are different, the RLL program gets interrupted at various points in the execution over time. This does not present a problem. The CPU always returns to the point where it left to resume the program execution.

Input Assignments for Timed Interrupt Mode

X0: Filtered input (uses filter time set for X1)
X1: Filtered input
X2: Filtered input
X3: Filtered input

Timed Interrupt Specification

Timed interrupts 1 (internal to CPU)
Time interval 5 to 999ms (1ms increments)
Interrupt subroutine INT0





DL105 Instruction Set

Boolean Instruction	Comparative Boolean Instructions	Immediate Instructions
NO Begins a new rung or an additional branch in a rung with a normally open contact.	NO (Equal) (EQ) Begins a new rung or additional branch in a rung with a normally open comparative contact. The contact will be on when A = B.	NO (Normally Open) (NO) Begins a rung/branch of logic with a normally open contact. The contact will be updated with the current input field status when processed in the program scan.
NC Begins a new rung or an additional branch in a rung with a normally closed contact.	NO (Not Equal) (NE) Begins a new rung or additional branch in a rung with a normally closed comparative contact. The contact will be on when A ≠ B.	NC (Normally Closed) (NC) Begins a rung/branch of logic with a normally closed contact. The contact will be updated with the current input field status when processed in the program scan.
OR Logically ors a normally open contact in parallel with another contact in a rung.	OR (Equal) (OE) Connects a normally open comparative contact in parallel with another contact. The contact will be on when A = B.	OR (Normally Open) (OR) Connects a normally open contact in parallel with another contact. The contact will be updated with current input field status when processed in the program scan.
OR NC Logically ors a normally closed contact in parallel with another contact in a rung.	OR (Not Equal) (ONE) Connects a normally closed comparative contact in parallel with another contact. The contact will be on when A ≠ B.	OR NC (Normally Open / Not Closed) Connects a normally closed contact in parallel with another contact. The contact will be updated with the current input field status when processed in the program scan.
AND Logically ands a normally open contact in series with another contact in a rung.	AND (Equal) (AE) Connects a normally open comparative contact in series with another contact. The contact will be on when A = B.	AND (Normally Open) (AND) Connects a normally open contact in series with another contact. The contact will be updated with the current input field status when processed in the program scan.
AND NC Logically ands a normally closed contact in series with another contact in a rung.	AND (Not Equal) (ANE) Connects a normally closed comparative contact in series with another contact. The contact will be on when A ≠ B.	AND NC (Normally Open / Not Closed) Connects a normally closed contact in series with another contact. The contact will be updated with the current input field status when processed in the program scan.
AND Series (AS) Logically ands two branches of a rung in series.	AND (≥) Begins a new rung or additional branch in a rung with a normally open comparative contact. The contact will be on when A ≥ B.	AND (Discrete Output) (DO) Reflects the status of the rung and outputs the discrete (ON/OFF) state to the image register. Multiple OR OUT instructions referencing the same discrete point can be used in the program. The output field device status is updated when the instruction is processed in the program scan.
OR Series (OS) Logically ors two branches of a rung in parallel.	AND (≤) Begins a new rung or additional branch in a rung with a normally closed comparative contact. The contact will be on when A ≤ B.	OR (Output) (OR) An output that turns on a point or a range of points. The reset instruction is used to turn the point(s) OFF that were set ON with the set instruction.
ON Reflects the status of the rung (on/off) and outputs the discrete (on/off) state to the specified image register point or memory location.	OR (≥) Connects a normally open comparative contact in parallel with another contact. The contact will be on when A ≥ B.	Reset (RST) An output that resets a point or a range of points. The output field device status is updated when the instruction is processed in the program scan.
OR OUT Reflects the status of the rung and outputs the discrete (ON/OFF) state to the image register. Multiple OR OUT instructions referencing the same discrete point can be used in the program.	OR (≤) Connects a normally closed comparative contact in parallel with another contact. The contact will be on when A ≤ B.	
ON Transition (OT) Is typically known as a one shot. When the input logic produces an off to on transition, the output will energize for one CPU scan.	OR (Series) (OS) Connects a normally open comparative contact in series with another contact. The contact will be on when A ≥ B.	
OR SET An output that turns on a point or a range of points. The reset instruction is used to turn the point(s) OFF that were set ON with the set instruction.	OR (Not ≤) (ONL) Connects a normally open comparative contact in series with another contact. The contact will be on when A < B.	
OR RST An output that resets a point or a range of points.		
OR Update (OU) Disable the update for range of specified output points.		

Accumulator / Stack Load and Output Data Instructions	Timer, Counter, and Shift Register Instructions	Logical Instructions (Accumulator)
LD Loads a 16 bit word into the lower 16 bits of the accumulator / stack.	T0.0 (Timer) Single input incrementing timer with 0.1 second resolution (0-999.9 seconds).	AND-AND Logically ands the lower 16 bits in the accumulator with a V memory location.
LDW Loads a 32 bit word into the accumulator / stack.	T0.01 (Timer) Single input incrementing timer with 0.01 second resolution (0-99.99 seconds).	AND-ANDC Logically ands the value in the accumulator with two consecutive V memory locations or an 8-digit constant.
LDI Loads the accumulator with a specified number of consecutive discrete memory bits.	T0.02 (Timer) Two input incrementing timer with 0.1 second resolution (0-9,999,999.9 sec.). Time and enable/reset inputs control the timer.	OR Logically ors the lower 16 bits in the accumulator with a V memory location.
LDH Loads the accumulator with the HEX value for an octal constant (address).	T0.03 (Timer) Two input incrementing timer with 0.01 second resolution (0-999,999.99 sec.). Time and enable/reset inputs control timer.	OR-ORC Logically ORs the value in the accumulator with two consecutive V memory locations or an 8-digit constant.
MOV Copies the values in the lower 16 bits of the accumulator to a specified V memory location.	C0.0 (Counter) Two input incrementing counter (0-9999). Count and reset inputs control the counter.	EX-OR Performs an Exclusive Or of the value in the lower 16 bits of the accumulator and a V memory location.
MOVW Copies the value in the accumulator to two consecutive V memory locations.	C0.0RST (Counter) Single input incrementing counter (0-9999). RST instruction must be used to reset count.	EX-ORC Performs as Exclusive Or of the value in the accumulator with two consecutive V memory locations or an 8-digit constant.
MOVH Outputs a specified number of bits (1-32) from the accumulator to the specified discrete memory locations.	C0.03 (Counter) Three input counter (0-99999999). Up, down, and reset inputs control the counter.	COMPARE Compares the value in the lower 16 bits of the accumulator with a V-memory location.
MOVV Moves the value from the first level of the accumulator stack to the accumulator and shifts each value in the stack up one level.	SR (Shift Register) Shift data through a range of control relays with each clock pulse. The data, clock, and reset inputs control the shift register.	COMPAREC Compares the value in the accumulator with two consecutive V-memory locations or an 8-digit constant.



DL105 Instruction Set

Math Instructions (Accumulator)	Bit Instruction (Accumulator)	Number Conversion Instructions (Accumulator)
<p>ADD BCD Adds a BCD value in the lower 16 bits in the accumulator with a V memory location. The result resides in the accumulator.</p> <p>ADD BCD Add a BCD value in the accumulator with two consecutive V memory locations or an 8-digit constant. The result resides in the accumulator.</p> <p>SUB BCD Subtract a BCD value, in a V memory location from the lower 16 bits in the accumulator. The result resides in the accumulator.</p> <p>SUB BCD Subtract a BCD value, which is either two consecutive V memory locations or an 8-digit constant, from a value in the accumulator. The result resides in the accumulator. (continued below)</p>	<p>Shift Left Shifts the bits in the accumulator a specified number of places to the left.</p> <p>Shift Right Shifts the bits in the accumulator a specified number of places to the right.</p> <p>Encode Bit Encodes the bit position set to 1 in the accumulator, and returns the appropriate binary representation in the accumulator.</p> <p>Decode Bit Decodes a 5-bit binary value (0-31) in the accumulator by setting the appropriate bit position to 1 in the accumulator.</p>	<p>BCD to BIN Converts the BCD value in the accumulator to the equivalent binary value. The result resides in the accumulator.</p> <p>BIN to BCD Converts the binary value in the accumulator to the equivalent BCD value. The result resides in the accumulator.</p> <p>ONE'S COMP Takes the one's complement of the 32-bit value in the accumulator. The result resides in the accumulator.</p>

Math Instructions (Accumulator) (Continued)	Interrupt Instructions	PLUS Stage Programming/ Drum Programming Instructions
<p>MULT BCD Multiplies a BCD value, which is either a V memory location or a 4-digit constant, by the value in the lower 16 bits in the accumulator. The result resides in the accumulator.</p> <p>DIV BCD Divides a BCD value in the lower 16 bits of the accumulator by a BCD value which is either a V memory location or a 4-digit constant. The result resides in the accumulator.</p> <p>INC BCD Increments a binary value in a specified V memory location by 1 each time the instruction is executed.</p> <p>DEC BCD Decrements a binary value in a specified V memory location by 1 each time the instruction is executed.</p>	<p>Enable Interrupt When a hardware or software interrupt occurs the interrupt routine will be executed. The INT instruction is the beginning of the interrupt routine. The interrupt routine is terminated with an IRT instruction (unconditional interrupt return). When a interrupt return is reached the execution of the program continues from the instruction where the program execution was prior to the interrupt.</p> <p>Enable Program Run Enable hardware and software interrupt to be acknowledged.</p> <p>Disable Interrupt Disable hardware and software interrupt from being acknowledged.</p>	<p>Initial Stage The initial stage instruction is used for a starting point for user application program. The ISG instruction will be active on power up and PROGRAM to RUN transitions.</p> <p>Stage Stage instructions are used to create structured program. They are program segments which can be activated or deactivated with control logic.</p> <p>N.O. Coil N.O. coil that deactivates the active stage and activates a specified stage when there is power flow to the coil.</p> <p>Power Rail Control Allows the program to control sections of logic by forming a new power rail. The MLS marks the beginning of a power rail and the MLR marks the end of the power rail control.</p> <p>Time and Event Driven Drum Time and/or event driven drum with up to 16 steps and 16 discrete output points. Output status is written to the appropriate output during each step. Specify a time base per count (in ms). Each step can have a different number of counts and an event to trigger the counting. Once the time has expired, a transition to the next step occurs. Also define preset step as destination when reset occurs.</p>

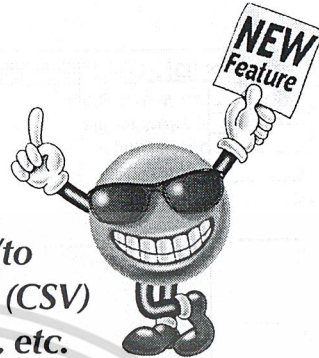
Table Instructions	CPU Control Instructions	Message Instructions
<p>Move Table Moves the values from one V memory table to another V memory table.</p> <p>Copy Table Copies data from data label area in program ladder memory to V-memory.</p>	<p>No Operation Inserts a no operation coil at specified program address.</p> <p>End Marks the termination point for the normal program scan. An End instruction is required at the end of the main program body.</p> <p>Stop Changes the mode of the CPU from Run to Program (Stop).</p>	<p>Display Value Displays a V-memory value or a Data Label constant to the handheld programmer or personal computer using DirectSOFT.</p> <p>Store Constant Stores constants in numerical or ASCII form for use with other instructions.</p>



What's New

DirectSOFT32 programming . . .

New DSLaunch facility as a central manager for all your projects, communication links, and applications



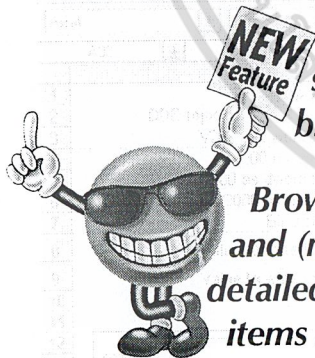
New Import/Export documentation feature from/to the widely supported Comma Separated Variable (CSV) file format for use with Excel, Lotus 1-2-3, Access, etc.

Support for Windows Telephony Application Programming Interface (TAPI) for modem configuration

New Project Information data with each project — including project creation and save dates, PLC Comm. link and PLC type, project description and version, and company/department/programmer

All the neat Win95/98/NT features: dockable/tear-off toolbars, toolbar ToolTips, long file names, common file browsers

DirectSOFT32 DSData server . . .



DDE and OPC (OLE for Process Control) all in one software package. New fully compliant OPC supports browsing for point and click use with OPC Clients.

Browser style interface allows you to browse the Topics and (new) SubTopics in the left browser pane, and see the detailed event information on the associated data point items in the right browser pane.

Multiple SubTopics can now be created below each Topic, allowing independent Update Rates per SubTopic.

New NickName Support — when a Topic is associated with a DirectSOFT32 Documentation database, the NickNames can be viewed in the DSData Server browser without having to launch DirectSOFT32 Programming.

This section provides more detailed descriptions of the various software tools that compliment **DirectLogic** PLCs.

There are three basic software tools:

- **DirectSOFT** Programming Software
- **DS Data Server**
- **VIA Direct**

DirectSOFT32 programming software

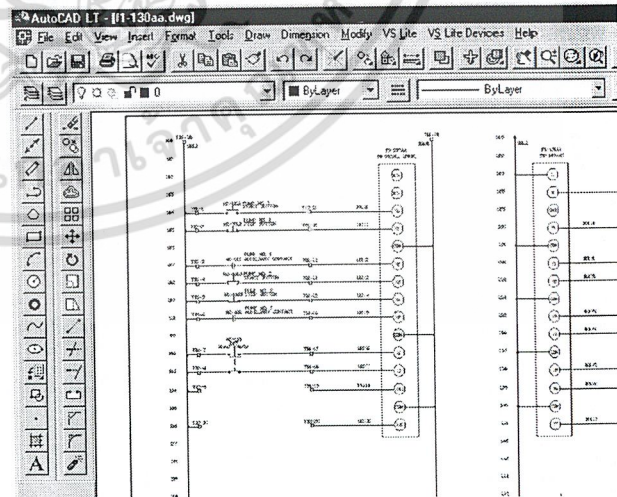
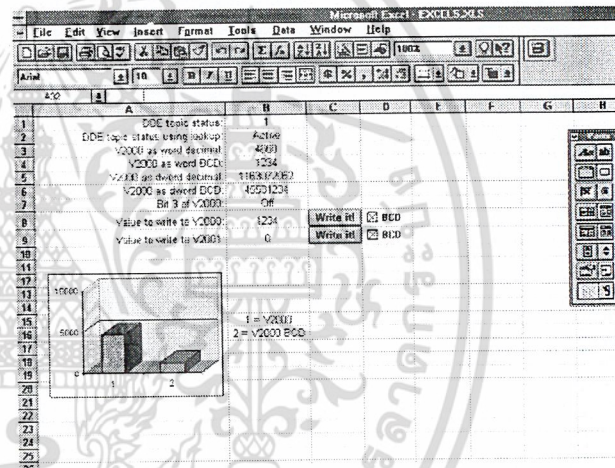
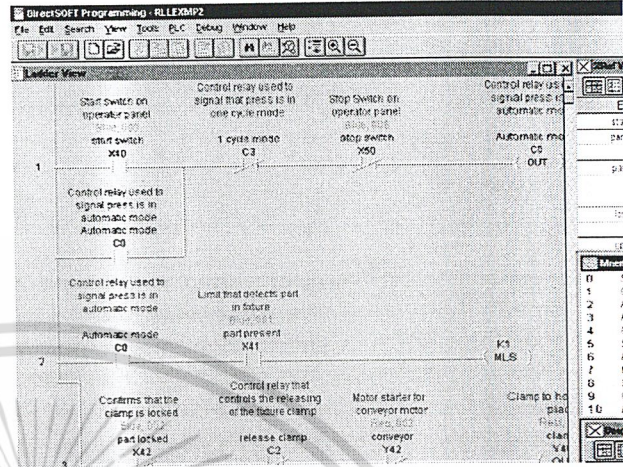
Our powerful Windows-based programming packages make it easy for you to program and monitor your PLC system. We offer several different packages, from our full support PC-PGMSW that programs all of our products, to single family support packages.

DSData server

Would you like to use one of your standard Windows packages, such as Excel to exchange data with your PLC? If so, check out the DSData Server. Basically, it acts as a translator to allow you to use most any DDE or OPC-compliant software to exchange data with our PLCs.

VIA Direct

VIA Direct UltraLITE is a powerful add-on package that works within AutoCAD or AutoCAD LT. By adding pull-down menus, a toolbar, and simple dialog boxes, AutoCAD and AutoCAD LT are transformed from drafting packages into tools specifically designed for creating electrical control schematics.



System requirements

Our new products are 32-bit applications that run under the Windows environment. The following provides the minimum system requirements.

- Pentium CPU, 133Mhz clock speed
- Windows 95/98/NT 4.0, or later (sorry, no DOS, OS/2, Macintosh, or UNIX versions, 16-bit version available for WIN 3.1))
- 32MB free RAM and 11MB free hard disk space
- At least one unused serial communications port
- Color SVGA Monitor

NOTES:

Laptops: DirectSOFT is perfectly suitable for use with laptop computers as long as they meet the requirements shown above.

Monitors: Any size monitor will work, however, larger monitors enhance the features of DirectSOFT. Also, we strongly recommend a color monitor. DirectSOFT makes use of color to note certain conditions, such as program editing changes, error conditions, etc.



DirectSOFT32

Programming versions

DirectSOFT32 Programming Software is available in several different packages. You can select the fully loaded package or one that is tailored just for a specific CPU.

The fully-loaded version will program all of our **DirectLogic** PLCs, some GE Series 1, and TI305/405 SIMATIC PLCs. Refer to the chart below to verify compatibility.

The single family versions were designed to provide a lower cost, more streamlined version for some of the PLC families. The table below provides a listing of available single user versions.

OEM and site licenses

If you are an OEM supplier who needs to ship a copy of the software with the machines that you have built, we offer several different OEM packages. Or, if you have a need for multiple copies at one location, there are several site license packages available. Refer to the table below to verify the packages for your application.

PLC compatibility

If you're using our **DirectLogic** products, then compatibility is not an issue. Make sure the package you choose supports the CPU you are using. Our software also supports many compatible products that were offered by previous vendors of the Koyo-designed PLCs. If you're buying the software to program a compatible CPU, check the following table to make sure your CPU is supported. (Sorry, but if your PLC is not shown, then it is not supported.)

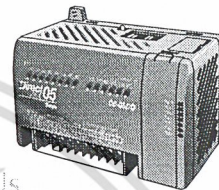
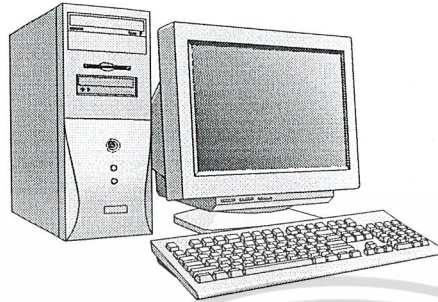
Family	CPU	DirectSOFT Programming PC-PCMSW	DirectSOFT Programming Single Family	DirectSOFT Site Licenses	DirectSOFT OEM Licenses	DirectSOFT DDE Server
DL05	(requires Rel. 2.4 or later)	✓	PC-PGM105	✓		✓
DL105	F1-130** (requires Rel. 2.0 or later)	✓	PC-PGM105	✓		✓
DL205	DL-230	✓	PC-PGM205	✓	PC-D2OEM	✓
	D2-240	✓	PC-PGM205	✓	PC-D2OEM	✓
	D2-250 (requires Rel. 2.1 or later)	✓	PC-PGM205	✓	PC-D2OEM	✓
DL305	D3-330*, D3-330P*	✓		✓	PC-D3OEM	✓
	D3-340	✓		✓	PC-D3OEM	✓
	D3-350 (requires Rel. 2.3 or later)	✓	PC-PGM350	✓	PC-D3OEM	✓
DL405	D4-430	✓		✓	PC-D4OEM	✓
	D4-440**	✓		✓	PC-D4OEM	✓
	D4-450** (requires Rel 2.0 or later)	✓		✓	PC-D4OEM	✓
GE® Series 1	IC610CPU105*	✓		✓	PC-D3OEM	✓
	IC10CPU106*	✓		✓	PC-D3OEM	✓
TI305™/ SIMATIC® TI305™	325-07*, PPX:325-07*	✓		✓	PC-D3OEM	✓
	330-37*, PPX:330-37*	✓		✓	PC-D3OEM	✓
	325S-07 (or 325 with Stage Kit)	✓		✓	PC-D3OEM	✓
	330S-37*, PPX:330S-37*	✓		✓	PC-D3OEM	✓
	335-37, PPX:335-37	✓		✓	PC-D3OEM	✓
TI405™/ SIMATIC® TI405™	425-CPU, PPX:425-CPU**	✓		✓	PC-D4OEM	✓
	PPX:430-CPU	✓		✓	PC-D4OEM	✓
	435-CPU, PPX:435-CPU**	✓		✓	PC-D4OEM	✓

*Requires Data Communications Unit (D3-232-DCU) ** also DC versions Note: In general, the compatible products listed offer similar features and are even identical in some cases. However, DirectSOFT has not been completely tested with the compatible products. There may be some aspects of system operation that may not be supported, or that may not work the same as previous software packages.

Basic system compatibility

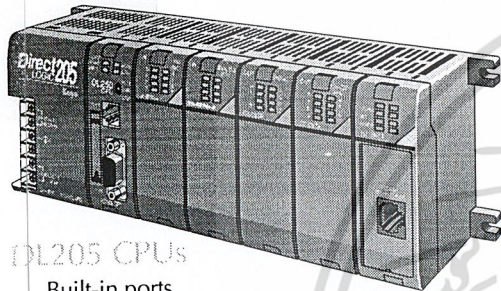
PC

- Windows 95/98/2000/NT
- Pentium 133 or higher
- 32mB RAM
- 11mB hard drive space
- CD ROM
- Color Monitor



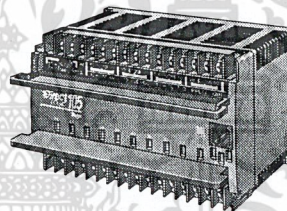
DL205 CPUs

- 2 Built-in ports
- Setup for Drum Sequencer
- Max. baud = 9.6K



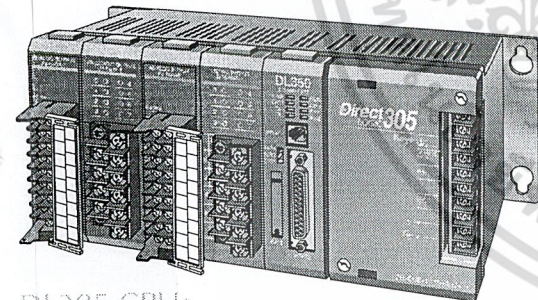
DL205 CPUs

- Built-in ports
- Requires RS232/422 converter if multidrop
- Max. baud = 38.4K



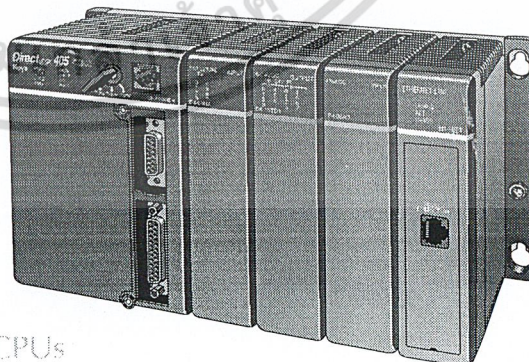
DL105 CPUs

- 1 Built-in ports
- Setup for Drum Sequencer
- Max. baud = 9.6K



DL305 CPUs

- Built-in ports
- Requires RS232/422 converter if multidrop
- Max. baud = 38.4K



DL405 CPUs

- 2 or 3 Built-in ports
- Max. baud = 19.2K
- DCM offers higher performance with baud rates up to 38.4K.

Software Tools

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกรทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Programming Cables

Since CPUs provide many different communication port possibilities, it is helpful to know exactly which

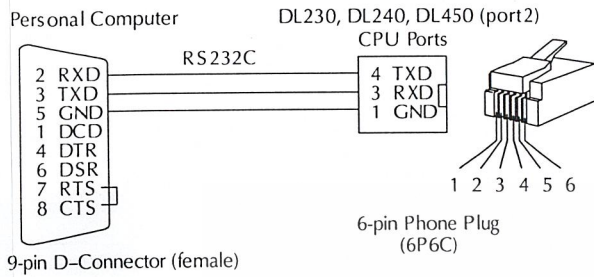
communications cable is required. Use the table below to choose the proper cable.

Family	CPU (or other device)	Port	Cable	Supported Protocol		
				DirectNET	K-sequence	ECOM
DirectLOGIC DL05	All versions	Port 1	D2-DSCBL	•	•	
		Port 2	D2-DSCBL	•	•	
DirectLOGIC DL105	DL130 (all versions)	Only one	D2-DSCBL		•	
DirectLOGIC DL205	D2-230	Only one	D2-DSCBL		•	
		Top port	D2-DSCBL		•	
	D2-240	Bottom port	D2-DSCBL	•	•	
		Top port	D2-DSCBL	•	•	
	D2-250	Bottom port	D2-DSCBL-1	•	•	
		Only one	D3-DSCBL-2	•	•	
DirectLOGIC DL305	D2-DCM (module)	Only one	D3-DSCBL-2	•	•	
	H2-ECOM /-F (modules)	Only one	**	•	•	•
	D3-330/D3-330P	Requires DCU*	D3-DSCBL-2	•		
		Top port	D3-DSCBL-1	•		
	D3-340	Bottom port	D3-DSCBL-1	•		
		Top port	D2-DSCBL-1	•	•	
DirectLOGIC DL405	D3-350	Bottom port	D3-DSCBL-2	•	•	
		Only one	D3-DSCBL-2	•	•	
	D4-430/D4-440P	Top port (15-pin)	D4-DSCBL		•	
		Bottom port (25-pin)	D3-DSCBL-2	•	•	
	D4-450	Phone jack	D2-DSCBL	•	•	
		Top port (15-pin)	D4-DSCBL		•	
GE® Series 1	D4-DCM (module)	Bottom port-1 (25-pin)	D3-DSCBL-2	•	•	
		Only one	D3-DSCBL-2	•	•	
	H4-ECOM (module)	Only one	**	•	•	•
TI305™/SIMATIC® TI305™	Slice I/O (module)	Only one	D4-DSCBL		•	
	IC610CPU105/106	Requires DCU*	D3-DSCBL-2	•		
		Requires DCU*	D3-DSCBL-2	•		
	325-07, PPX:325-07	Requires DCU*	D3-DSCBL-2	•		
		Requires DCU*	D3-DSCBL-2	•		
	330-37, PPX:330-37	Requires DCU*	D3-DSCBL-2	•		
Requires DCU*		D3-DSCBL-2	•			
TI405™/SIMATIC® TI305™	325S-07 (or 325 with stage kit)	Requires DCU*	D3-DSCBL-2	•		
		Requires DCU*	D3-DSCBL-2	•		
	330S-37, PPX:330S-37	Requires DCU*	D3-DSCBL-2	•		
335-37, PPX:335-37		Phone jack	D3-DSCBL-1	•		
	if DCU is used*	D3-DSCBL-2	•			
TI405™/SIMATIC® TI305™	425-CPU, PPX:425-CPU	Only one	D4-DSCBL		•	
		Top port (15-pin)	D4-DSCBL		•	
	N/A, PPX:430-CPU	Bottom port (25-pin)	D3-DSCBL-2	•	•	
		Top port (15-pin)	D4-DSCBL		•	
	435-CPU, PPX:435-CPU	Bottom port (25-pin)	D3-DSCBL-2	•	•	
		Only one	D4-DSCBL		•	
Smart slice					•	

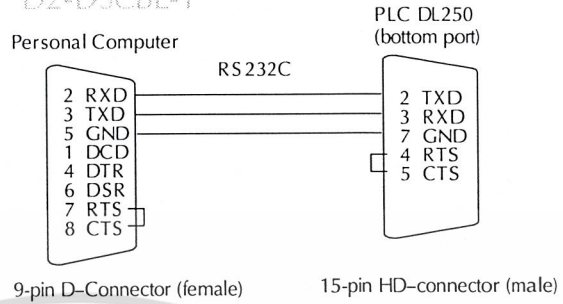
* requires RS232 Data Communications Unit (D3-232-DCU) **requires standard Ethernet twisted pair cable (HX-ECOM) or standard fiber optic cable (HX-ECOM-F)

Pin labeling conforms to IBM DTE and DCE standards

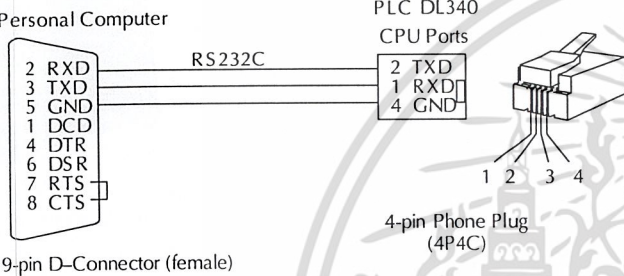
D2-DSCBL



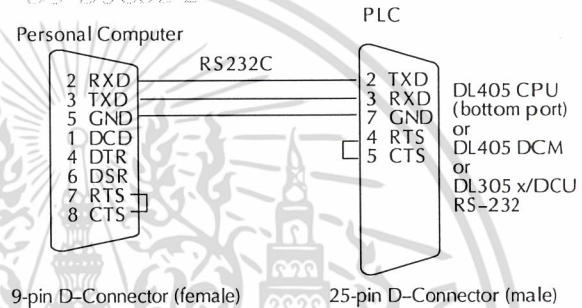
D2-DSCBL-1



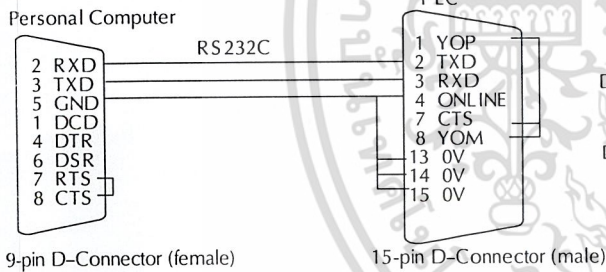
D3-DSCBL-1



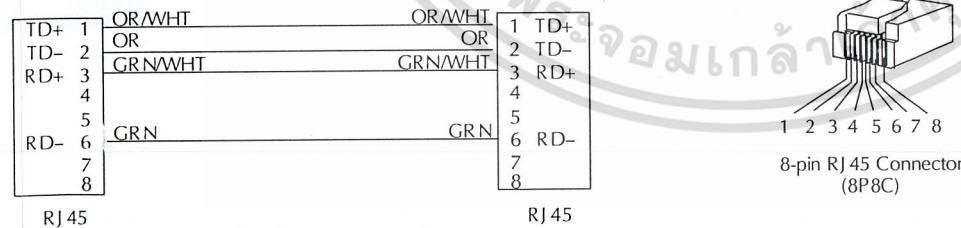
D3-DSCBL-2



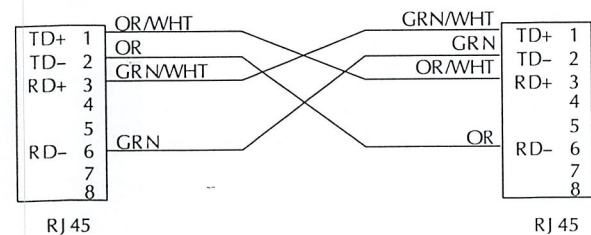
D4-DSCBL



Straight-through Patch Cable



Crossover Patch Cable



All Ethernet and ECOM patch cables should be Category 5, UTP Cable. The diagram illustrates the appropriate wire position in the RJ45 connector.



Programming software

- PC-PGMSW
- PC-PGM-105
- PC-PGM-205
- PC-PGM-305

Overview

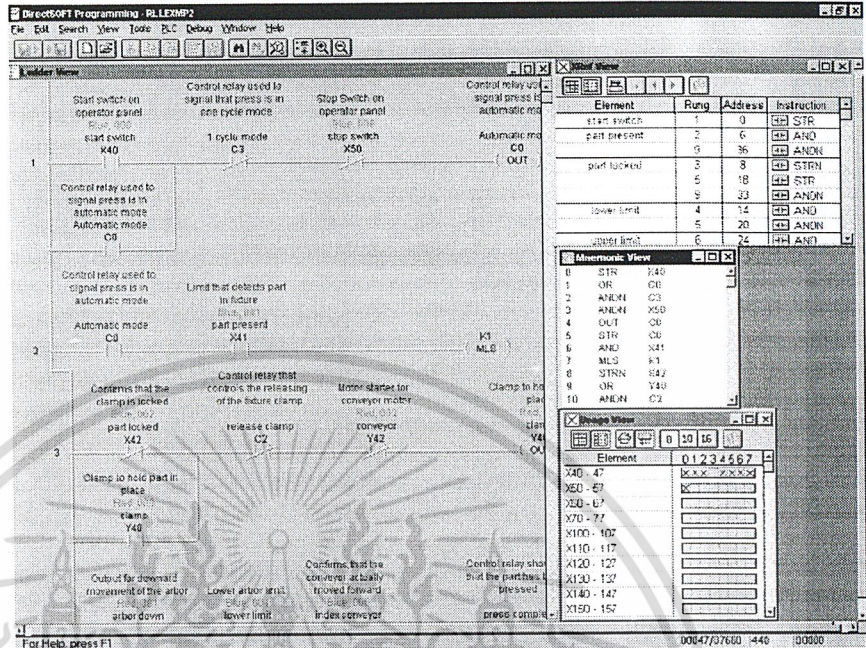
DirectSOFT32 Programming Software single-user licenses have been designed to make PLC programming and machine monitoring easier than ever before. The 32-bit Windows-based package provides simple menu and icon choices that operate similarly to those found in most standard Windows packages. Now you can have the power of Windows on the factory floor!

For more information and to download a demo version, please visit the Web site at www.automationdirect.com. You can view or download the Quick Start manual.

Which package is right for you? The programming software packages share the same general features, but there are a couple of key differences.

- **PC-PGMSW** programs all of our CPUs (DL05, DL105, DL205, DL305, and DL405). If you're a system integrator or OEM who uses one or more of our product lines, then this package is the right choice.
- **PC-PGM-305, PC-PGM-205 and PC-PGM-105** only programs their respective product families except for the PC-PGM-105 which programs both the DL05 and DL105. That is, PC-PGM-105, programs the DL05/105 Micro PLC, PC-PGM-205 only programs the DL205 CPUs, and the PC-PGM-305 programs the D3-350 CPU only.

Please note, you cannot upgrade the PC-PGM-350, PC-PGM-205 or PC-PGM-105 packages to support any other product families.



Number of users

The packages listed above are really "single-station" packages. That is, you can legally install them on only one computer. However, if there is only one person who primarily uses the computer (*over 80% of the time*), then you can install the software on a laptop, home office, or any other single backup computer. If you need more flexibility, or you need to install multiple copies in many areas, then check out our Site Licenses.

Laptop/portable computers and off-site usage

Many of you may be using laptop computers for field service work. This is perfectly acceptable as long as it adheres to the guidelines described above.

Can I ship a copy to my customer?

No, except for a single backup copy, you cannot make, install, or distribute any additional copies. You can transfer the original copy (original cd, any applicable upgrades, and any documentation) with our written consent prior to the transfer.

Of course, this means you cannot legally keep a copy once you have transferred the original. Some machine builders and OEMs are required to ship a copy of the PLC programming software with every machine. If this describes your situation, then see our OEM licenses. The OEM license allows you to ship an inexpensive copy of the software with every machine you sell.

Upgrades

You can receive maintenance releases at no charge by downloading them from our Web site. These do often include some feature enhancements along with software corrections. Any release that contains major feature enhancements will be subject to the normal upgrade charge in effect at that time. *Note: the upgrade will only update any packages that are already installed on your system.*

Program editing features

- Full-screen Point & Click Editing with the mouse and icons
- Edit Tool Box contains icons and hot key equivalents for all elements
- Browsers help you find proper addresses, contacts, coils, and box instructions.
- Neat Win95/98/NT features (tear-off toolbars, ToolTips, long filenames, common file browsers)
- Project Information data with each project including creation and save dates, links, PLC type, project description and version, and company/department/programmer.
- On-screen Cross Reference
- On-screen Element Usage in groups of 8, 10, or 16 points per line
- Predefined aliases for common items, such as timer & counter current values.
- Enter your program with nicknames and assign them later
- Access V-memory (Data Registers) with fill-in-the-blank Memory Editor
- Easily cut & paste program elements, single rungs, or groups of rungs.
- Cut & Paste rungs (with documentation) between programs
- Search for program elements by name, address, rung number, or instruction type
- Search for constants

Documentation features

- Grid-based documentation editor for easy access
- 16-character nicknames for addresses make troubleshooting and programming easier
- 16-character writing information allows you to tie PLC addresses to panel wiring numbers
- 6-line x 18-character description provides more detailed information about the program element
- Rung comments can be up to 20 lines of 69 characters each
- Cut, copy, and paste documentation between program elements
- Free-form program title page can be selected at time of printing
- Import/Export feature for documents

Printing features

- Print any available view, including ladders, charts, and mnemonics
- Print multiple columns
- Use zoom to enlarge or reduce print size
- Preview the printouts before you send them to the printer
- Create custom printouts by selecting exact items to include for each view
- Print program documentation, including nicknames, descriptions, rung comments, etc.
- Print program Cross Reference or Element Usage Table
- Print PLC V-memory or Data Register contents

Display and viewing features

- View program ladders
- View program mnemonics
- Stage view available for programs created with RLL PLUS instruction
- Choose rung number or addresses for ladder view
- View multiple programs at the same time
- Use split screen to view different sections of the same program
- Use zoom to enlarge or reduce the screen area for better viewing
- Customize display colors for editing, unsaved changes, documentation, etc.

System setup features

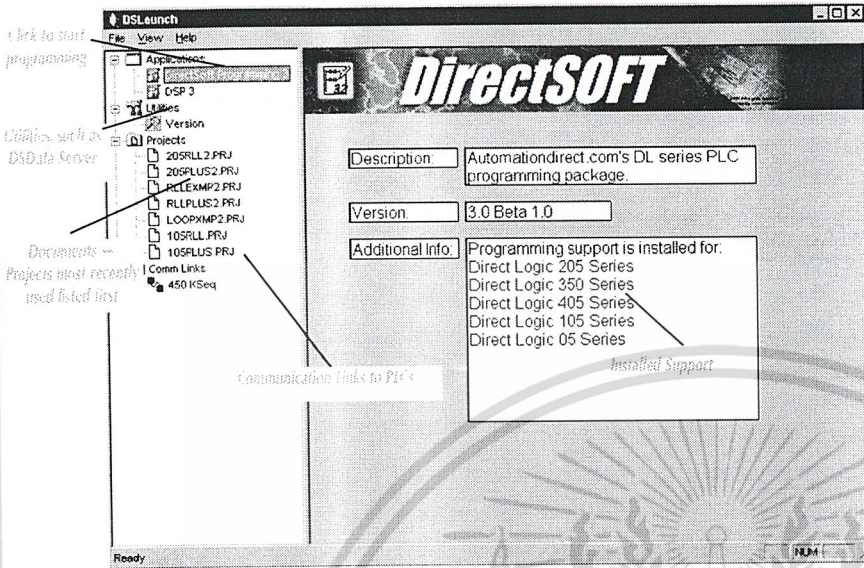
- Set up PID loops with easy-to-use dialog boxes
- Configure DV-1000
- Change PLC mode of operation
- Clear PLC memory by type or all
- Select or change retentive memory ranges
- Read PLC I/O configuration
- Change I/O configuration (if allowed by the PLC)
- Set PLC communication parameters through PC ports, through modems, or Ethernet
- View PLC firmware revisions

Online operation features

- DSLaunch as a central manager for all projects, links, and applications
- Automatically build & store communication links to the PLC, even using modems or Ethernet.
- Visible indication of PLC communication link status
- Status mode provides visible power flow condition for program elements, current values for timers and counters, and current values for V-memory or Data Register locations. Even status for Drum steps, outputs, event and/or time transitions, etc.
- Custom configure Data View windows to monitor various elements
- Display nicknames in Data View
- Save and store Data View windows by name
- Choose data format for Data View Window display (BCD, binary, Hexadecimal, etc.)
- Choose status symbol (switch, text, etc.) for discrete points in Data View windows
- Change PLC values from Data View windows
- View multiple PLCs simultaneously (not available with OEM packages)

Maintenance and debugging

- Automatic error checking during program entry
- Program syntax check
- Monitor PLC scan time
- Perform PLC I/O diagnostics
- View PLC errors and/or user error messages

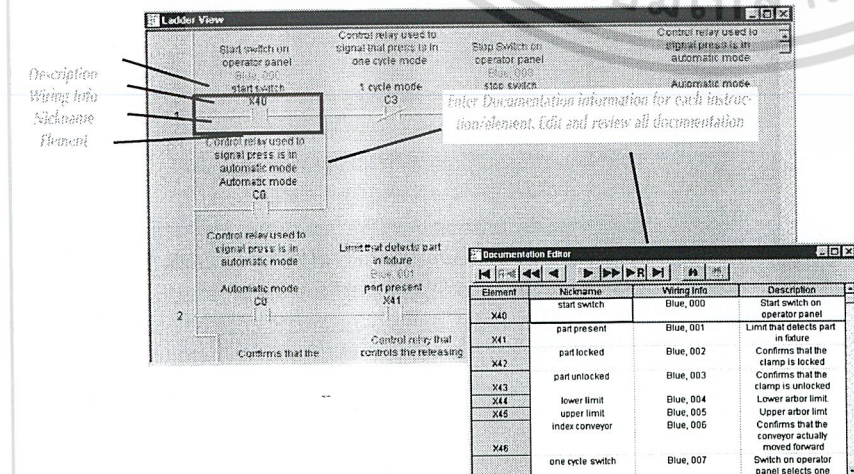
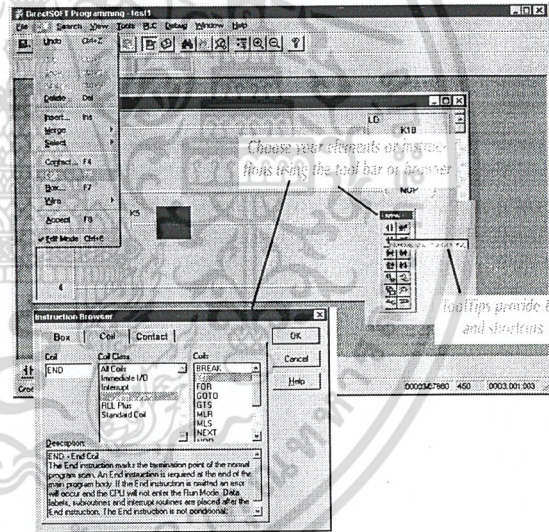


The launch window

The Launch Window is a centralized location to view not only all of your **DirectSOFT32** applications and projects, but also any additional applications such as Operator Interface software, Program Conversion packages, etc. This window is also used to create and manage PLC programs and the communications between your pc and the PLC.

Point and click editing

Once you have started a new project, there are a few different ways to start building your program. The Ladder Palette, which is floating and can be resized and repositioned anywhere on your desktop, contains most of the instructions. Simply click on the icon or use the Hot Key associated with that icon. Some instructions have to be accessed through one of the Browser icons. The Browsers contain all of the necessary information from various boxes, coils, and contacts.

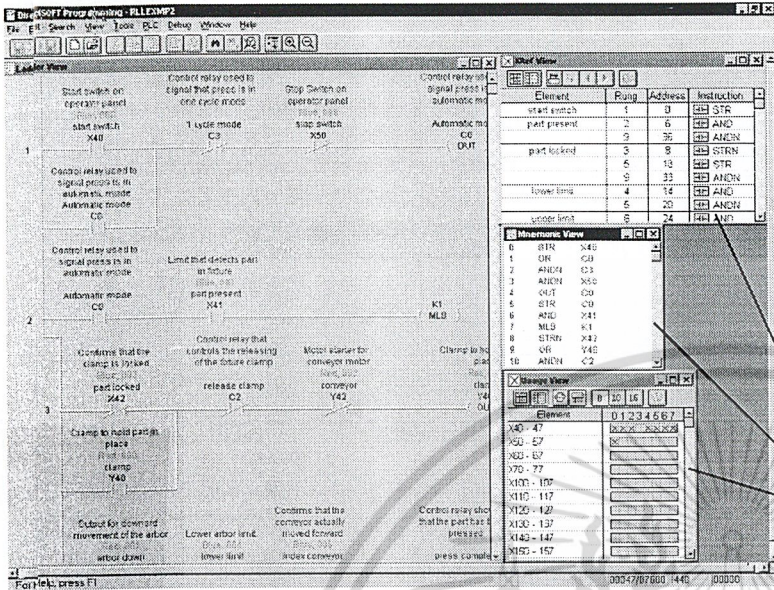


Documentation

Documentation is the text associated with the components and structure of your program. It helps to clarify the representation of elements by assigning 4 different fields: Description, Wiring, Info, Nickname, and Element.

On-screen views

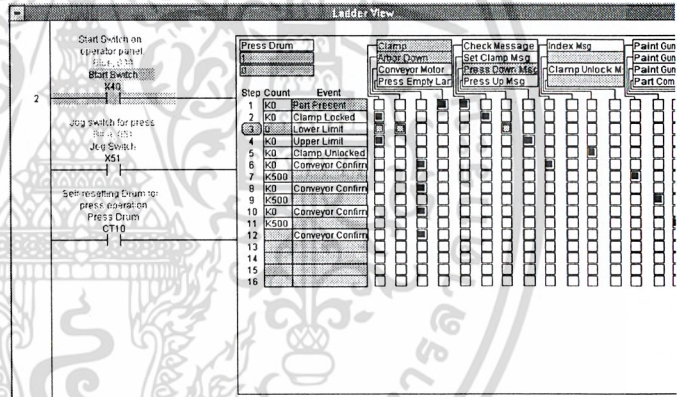
DirectSOFT32 allows you to keep different views to track program functions. You can have multiple windows open that provide you with; all of the addresses used in the program, Mnemonic view of the program instructions, all of the instructions used, just to name a few.



You can have multiple views of your project visible at the same time. This screen shows the Ladder View, Usage View, Mnemonic View and Xref View.

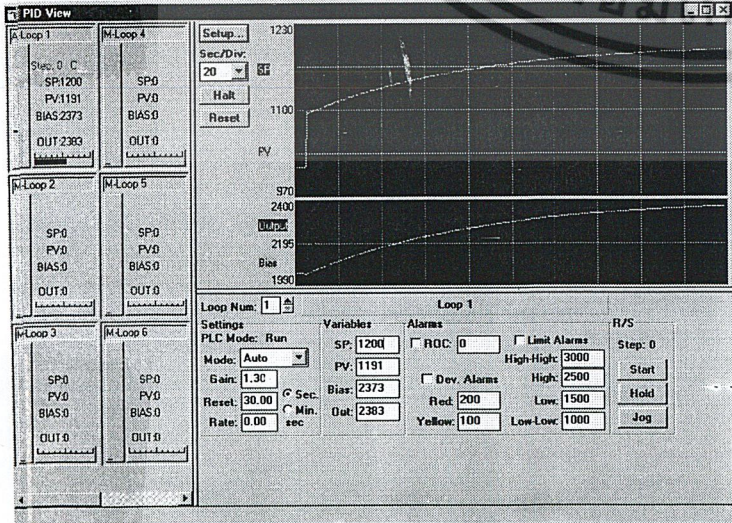
Support for drum instructions

If your PLC supports Drum Instructions, you will like this feature of the programming software. Simply fill in the top of the drum with outputs and configure each step with an event and/or time-based element. Click on the output boxes to choose the proper output states for each step and you are finished.



PID view

DirectSOFT32 offers an excellent PID feature. The loop and trending screens are very easy to configure. This feature has all of the views, trending, and tuning that you need to easily and successfully setup your PID loop.





Overview

In some larger installations (or in educational institutions) there are occasions when you have several people who each need access to a copy of the programming software. Of course, you could always purchase individual copies, but that can be expensive, especially if the software is only being used for occasional troubleshooting or system maintenance. A Site License package for **DirectSOFT** Programming Software is available. This program is designed for those who need to install the software on multiple machines throughout a single location.

A site license provides many benefits over purchasing separate single user licenses. The Site License package programs all **DirectLOGIC** families of PLCs (and many compatible PLCs).

Package contents

The Site License package contains one CD and one set of documentation. You can use this CD to install the software on the maximum number of PCs included in the site license. You can make copies of the documentation, or you can purchase additional copies.

Number of users

Each package is designed for a specific number of concurrent users. So, you may be wondering which license is right for you. The license defines a 'user' as someone who uses the software more than 20% of their normal working hours. This means that even though you purchase a 5-user license, you are actually allowed to have many other people who occasionally use the software. This is especially useful for technicians who only use the software for troubleshooting or maintenance on a random basis.

Portable computers and off-site usage

Many of you may be using laptop computers for field-service work. You can install the software on these machines as long as their permanent residence remains at the Site. It is perfectly acceptable to allow your engineers or technicians to take computers to job sites or remote office for occasional use.

Upgrades

Maintenance releases are provided at no cost by ordering them or downloading them from the Web. The maintenance releases provide bug fixes, corrections, and support for new release PLCs, but do not include any feature enhancements. Any release that contains feature enhancements will be subject to the normal upgrade charge in effect at that time. Here's an example: Let's say you purchased a 5-user license and two years from now you see an advertisement that shows some exciting new features. Rather than purchasing a new Site License, you would simply purchase 5 upgrade packages which would allow you to upgrade all of your packages at the site.

Can I ship a copy to my customer?

No, the Site License is not intended to allow you to ship copies of the software to your customers. If you need to do this, check out our OEM Licenses.

Packages

We offer several different packages designed for specific numbers of users. This allows you to purchase a license that closely matches your needs without paying a high price!

Site license packages

Users	Order No.
5-user	PC-5U
10-user	PC-10U
25-user	PC-25U
50-user	PC-50U

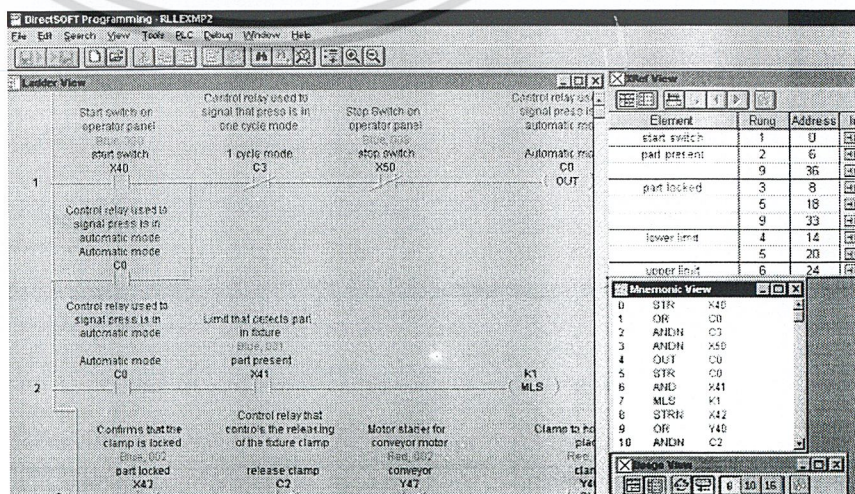
Manuals

DA-DSOFT-M

(Extra manuals are only available to Site License holders. They cannot be purchased by single user license holders.)

Extra user copies

Users	Order No.
5-user	PC-5U-XC
10-user	PC-10U-XC
25-user	PC-25U-XC
50-user	PC-50U-XC





Overview

An OEM license program for the **DirectSOFT32** Programming Software is also available. This program is designed to allow you to ship an inexpensive limited version of our powerful software with every machine you build!

Tailored packaging

Each package supports a single family of PLCs. For example, if you purchase the DL205 OEM License package (PC-D2OEM), you get support for the DL205 family of products. PC-D3OEM supports the DL305 family only, and PC-D4OEM supports the DL405 family only.

Note: It is not recommended to use an OEM package for your development system due to the slightly reduced feature set and single family support. It is recommended to purchase PC-PGMSW.

What is included

You will receive 5 complete sets of software when you purchase the license. Each set includes a production CD and a complete set of user documentation. Unlike some vendors, with our package there's no need to make your own diskettes or copies of the user manuals. Instead, you provide a professional-looking package that your customer can appreciate.

Extra copies

Once you have purchased the license, you can purchase extra copies at an even further discount. You can purchase an extra copy that is tailored to the exact product you are using.

Password protection

Protect the many hours of hard work in designing and troubleshooting your customer's machine by using the password utility. Using the password utility included with the OEM packages, you have the ability to add your own password to each package that you send out the door. You can even completely restrict program editing capacity if necessary.

IMPORTANT! *The OEM versions always allow the customer to at least view the program. The passwords only affect the ability to edit the program. These OEM packages are not a substitute for a CPU password, which is designed to minimize the risk of loss for proprietary program information.*

OEM license packages

Each OEM license provides support for a specific family of PLCs and comes with five complete packages. Each package includes a CD and the manual. By tailoring each package for specific product families, you don't have to pay for features that your products may not even utilize.

OEM licenses

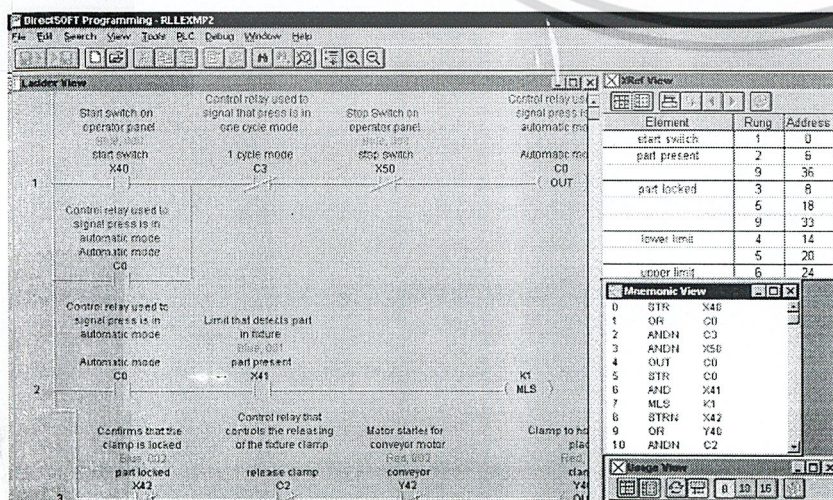
Family	Order No.
DL205	PC-D2OEM
DL305	PC-D3OEM
DL405	PC-D4OEM

Extra user copies

Family	Order No.
DL205	PC-D2OEM-X
DL305	PC-D3OEM-X
DL405	PC-D4OEM-X

Package contents

Each individual copy in the original license and any copies contain a production CD and an original set of documentation.



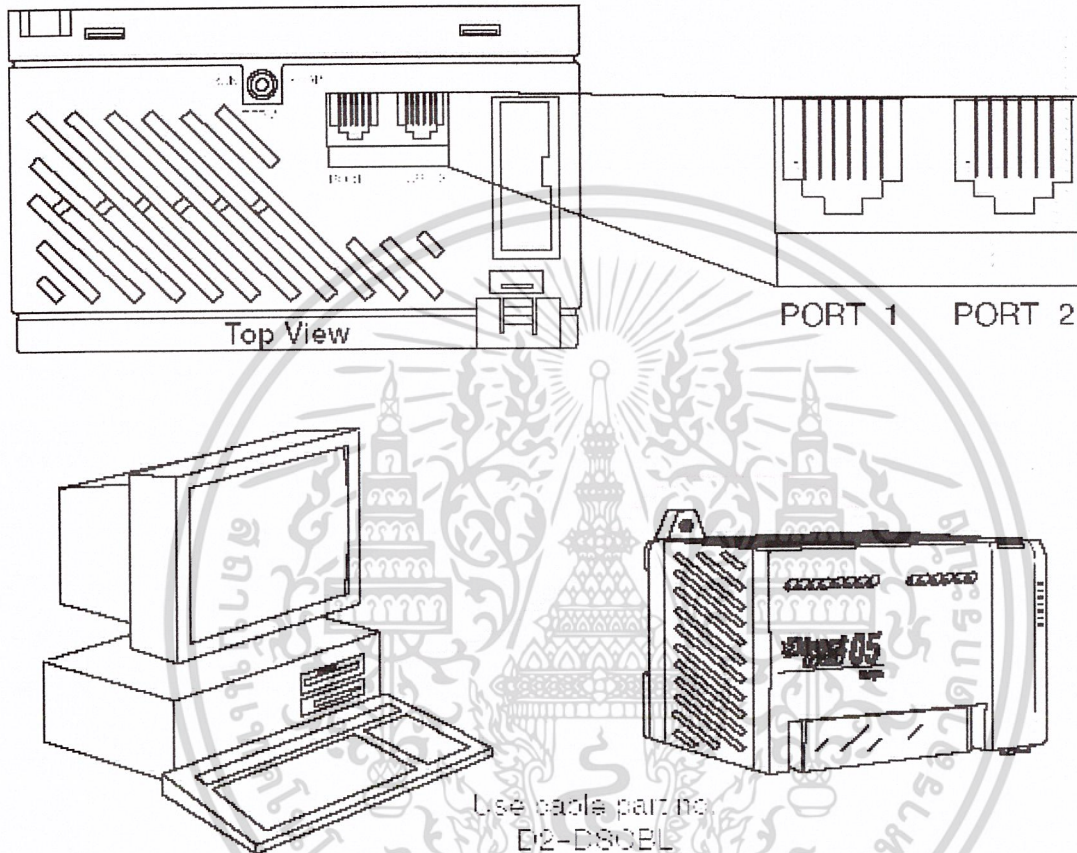


ภาคผนวก ง.
คู่มือและวิธีการใช้งานระบบส่งงานพีแอลซีเพื่อควบคุมกระบวนการทางเคมีโดยใช้
โทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการใช้งาน

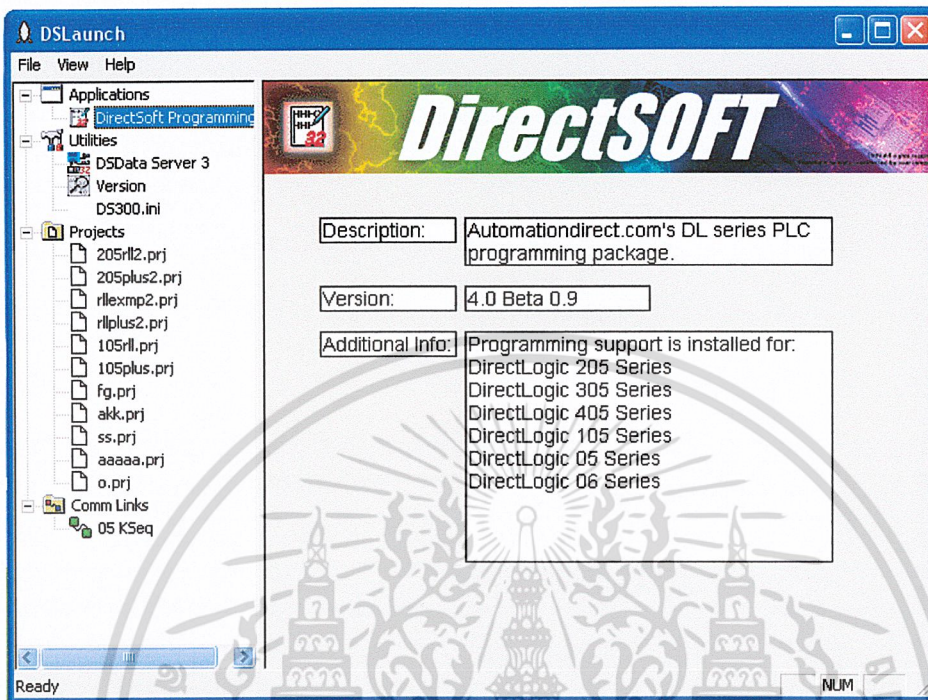
1. ต่ออุปกรณ์ทางด้านฮาร์ดแวร์เข้ากับแหล่งจ่ายไฟฟ้า 220V.AC และต่อสายจากเครื่อง PLC เข้ากับพอร์ตอนุกรมของคอมพิวเตอร์ตามรูป



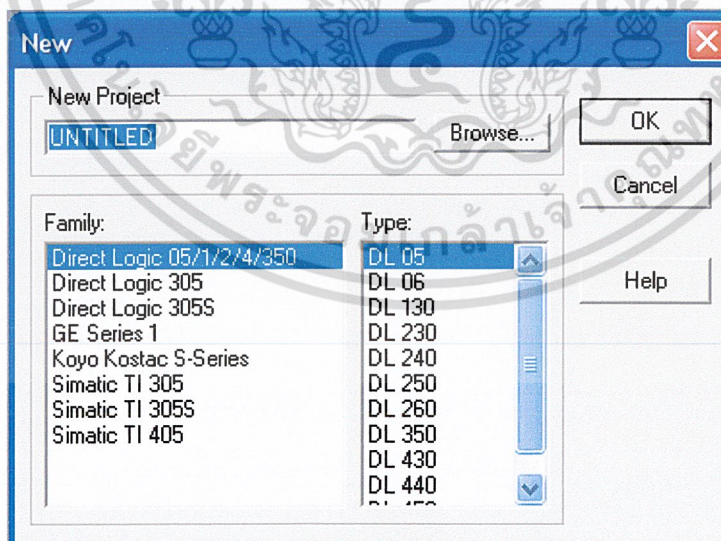
2. เปิดโปรแกรมที่ใช้สำหรับการเชื่อมต่อระหว่าง PLC กับ คอมพิวเตอร์แล้วทำการเปิดโปรแกรมตามขั้นตอนที่แสดงตามรูปภาพต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกโปรแกรมDirectSoft

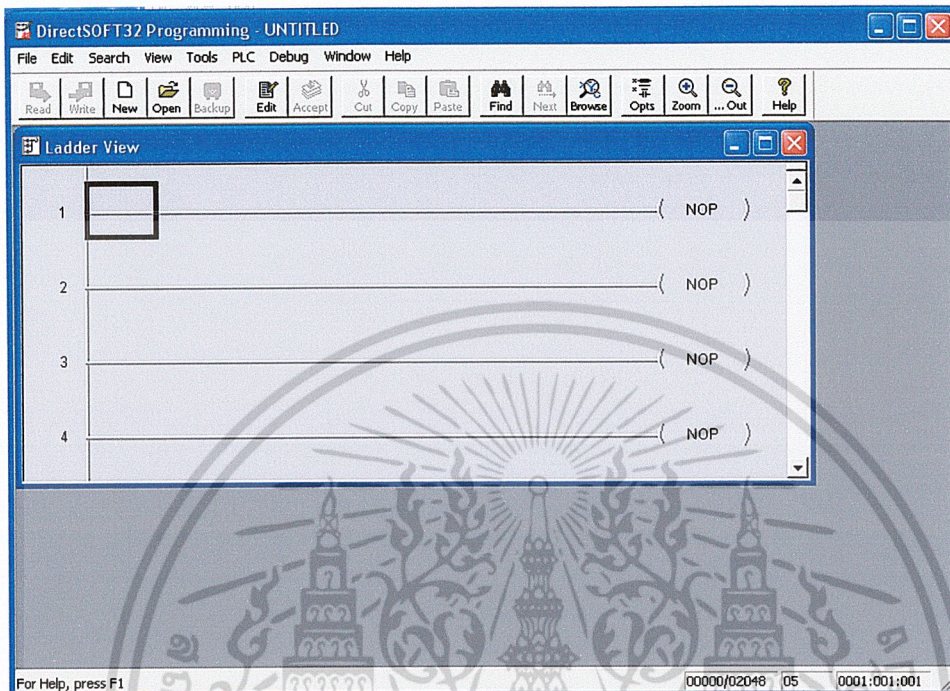


ปรากฏหน้าจอเพื่อให้เลือกรุ่นของ PLC ตามรูป

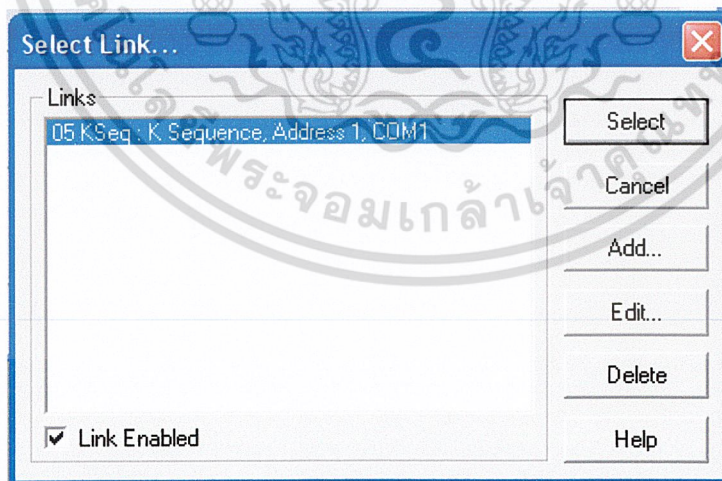


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือก PLC เพื่อติดต่อเรียกใช้โปรแกรม Ladder ที่อยู่ในหน่วยความจำของ PLC

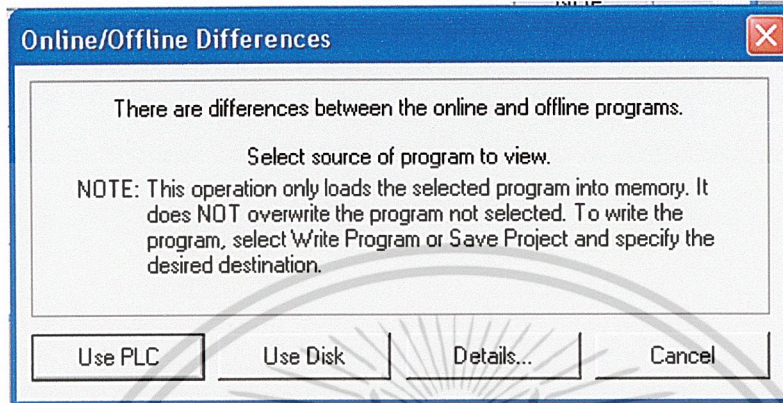


เลือกกด Select เพื่อกำหนดรูปแบบการติดต่อดังรูป

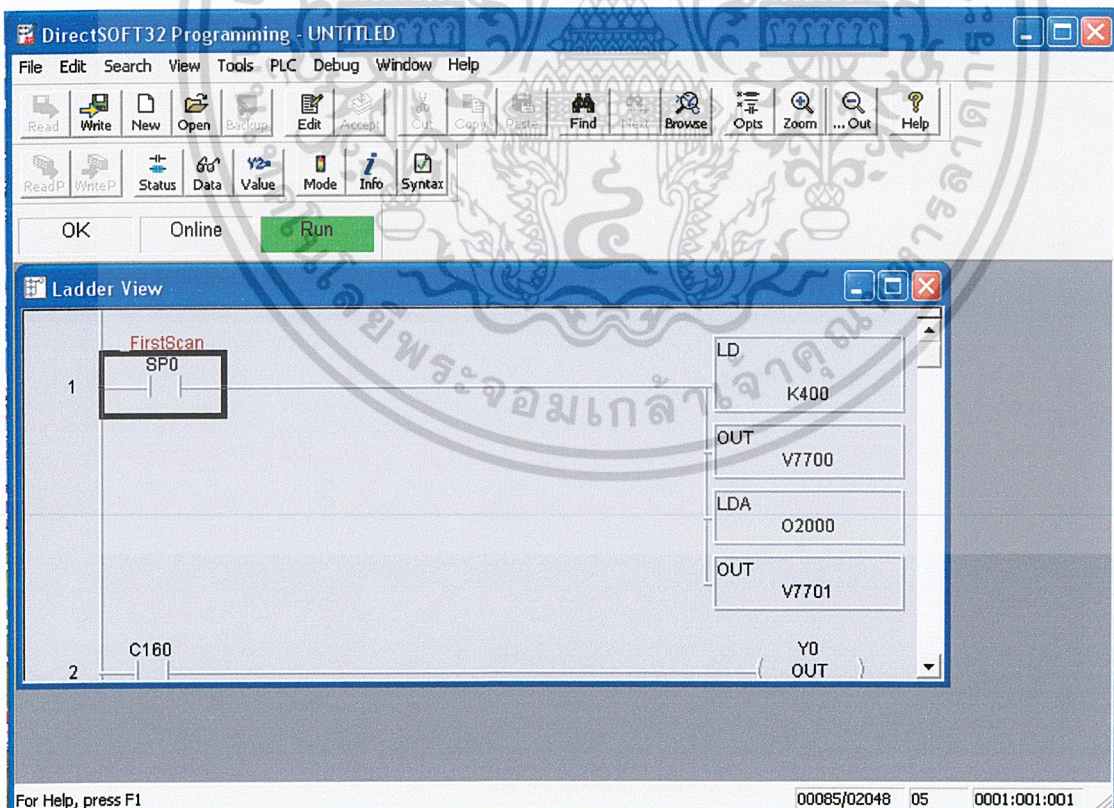


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือก Use PLC เพื่อใช้งาน โปรแกรมบน PLC

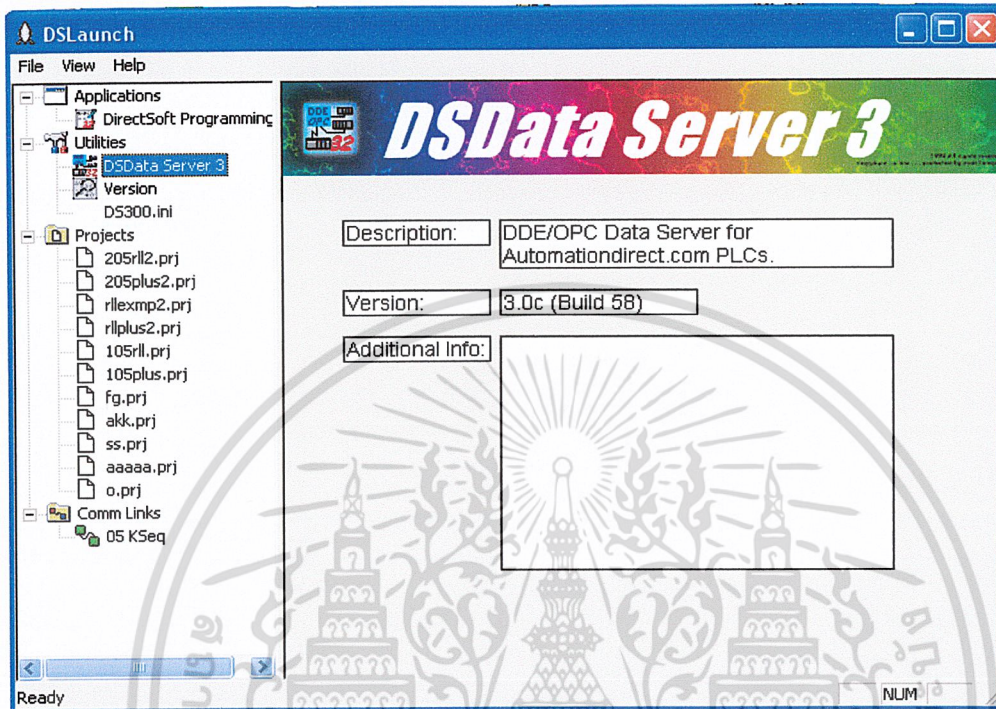


แสดงถึงหน้าโปรแกรมที่พร้อมจะใช้งาน

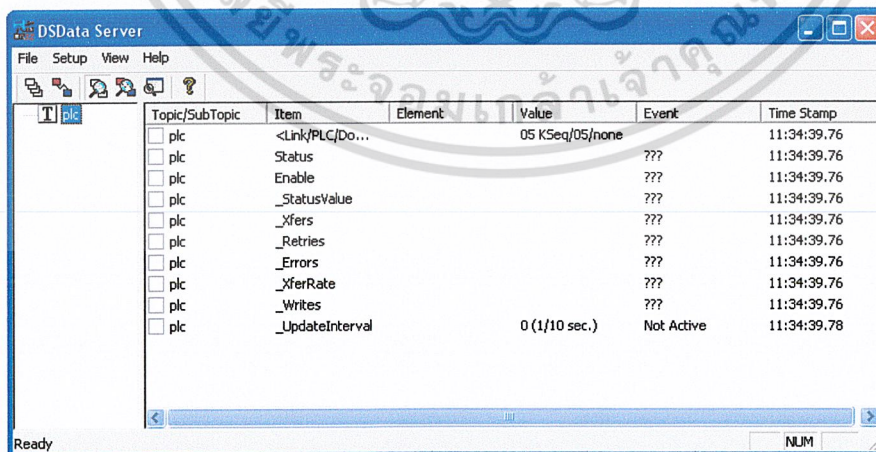


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการเปิดโปรแกรมที่ใช้ในการติดต่อกันระหว่าง PLC กับ Visual Basic

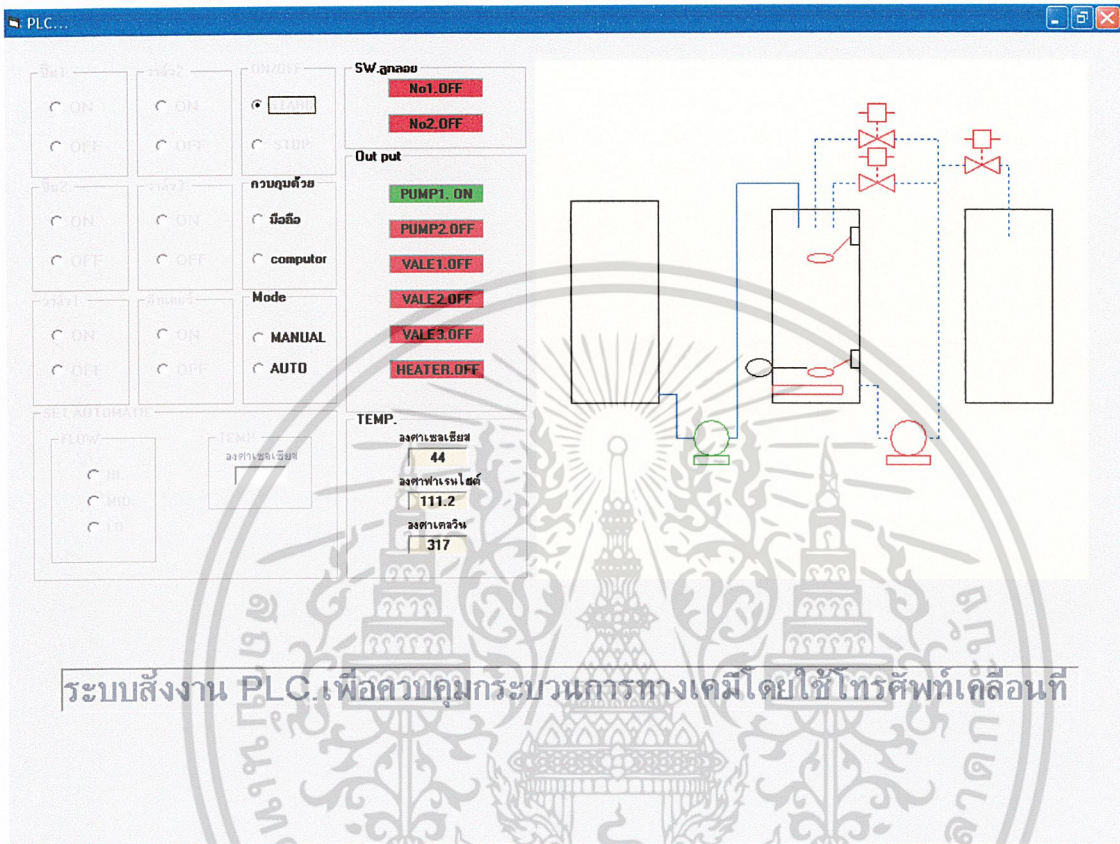


แสดงถึงหน้าโปรแกรมที่ใช้เชื่อมต่อ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เปิดหน้า Form ที่ใช้สำหรับควบคุมที่สร้างจากโปรแกรม Visual Basic



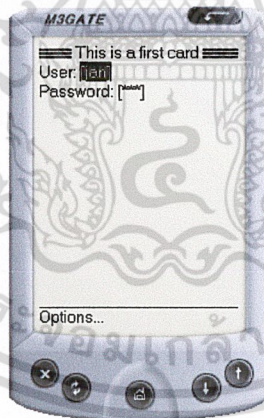
ระบบสั่งงาน PLC เพื่อควบคุมกระบวนการทางเคมีโดยใช้โทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ใช้ WAP เปิด WAP Site โดยพิมพ์ URL = “IP ของ WAP Server” /test.wml

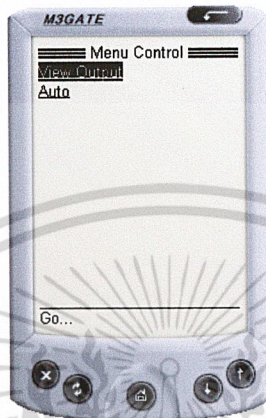


6. จากนั้นทำการ login เข้าสู่ระบบ โดยใส่ชื่อและรหัสผ่านให้ตรงกับในฐานข้อมูล



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. จากนั้นก็สามารถเข้าใช้งานในรูปแบบต่าง ๆ ได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้