CONFIGURATION AND CONNECTIONS

CHAPTER

4

In This Chapter:

DL05 System Design Strategies

I/O System Configurations

The DL05 PLCs offer a number of different I/O configurations. Choose the configuration that is right for your application, and keep in mind that the DL05 PLCs offer the ability to add an I/O card in the option slot. Although remote I/O isn't available, there are several option cards available. For instance:

- Various A/C and D/C I/O modules
- Combination I/O modules
- Analog I/O modules
- Combination Analog I/O modules

A DL05 system can be developed with an arrangement using a selected option modules. See our DL05/06 Options Modules User Manual (D0-OPTIONS-M) on the website, www.automationdirect.com for detailed selection information.

Networking Configurations

The DL05 PLCs offers the following ways to add networking:

- **Ethernet Communications Module** − connects a DL05 to high-speed peer-to-peer networks. Any PLC can initiate communications with any other PLC or operator interfaces, such as C-more, when using the ECOM modules.
- **Data Communications Modules** − connects a DL05 to devices using either DeviceNet or Profibus to link to master controllers, as well as a D0-DCM.
- **Communications Port 1** − The DL05 has a 6-pin RJ12 connector on Port 1 that supports (as slave) K-sequence, Modbus RTU or *Direct*NET protocols.
- **Communications Port 2** − The DL05 has a 6-pin RJ12 connector on Port 2 that supports either master/slave Modbus RTU or *Direct*NET protocols, or K-sequence protocol as slave. Port 2 can also be used for ASCII OUT communications.

Option Slot
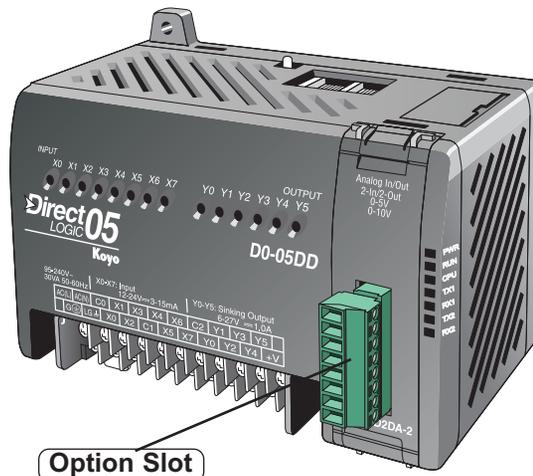
Automatic I/O Configuration

The DL05 CPUs will automatically detect the optional I/O module, if installed, at powerup and establish the correct I/O configuration and addresses. The configuration may never need to be changed.

The I/O addresses use octal numbering, with X0 to X7 being the eight inputs and Y0 to Y5 being the addresses for the six outputs. The discrete option slot addresses are assigned in groups of 8 or 16 depending on the number of I/O points for the I/O module. The discrete option module addressing will be X100 to X107 and X110 to X117 for the maximum sixteen point input module. The addressing for the sixteen point output module will be Y100 to Y107 and Y110 to Y117. Refer to the DL05/06 Options Modules User Manual (D0-OPTIONS-M) for the various discrete I/O modules available and the addressing for each one.

Power Budgeting

No power budgeting is necessary for the DL05. The built-in power supply is sufficient for powering the base unit, your choice of option module, the handheld programmer and the DV-1000 operator interface.

Network Configuration and Connections

Configuring the DL05's Comm Ports

This section describes how to configure the CPU's built-in networking ports for either Modbus or *Direct*NET. This will allow you to connect the DL05 PLC system directly to Modbus networks using the RTU protocol, or to other devices on a *Direct*NET network. Modbus host systems must be capable of issuing the Modbus commands to read or write the appropriate data. For details on the Modbus protocol, check with your Modbus supplier for the latest version of the Gould Modbus Protocol reference Guide. For more details on *Direct*NET, order our *Direct*NET manual, part number DA–DNET–M.

DL05 Port Specifications

Port 1	Communications Port 1
	Connects to HPP, *Direct*SOFT, operator interfaces, etc.
	6-pin, RS232C
	Communication speed: 9600 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), *Direct*NET (slave only), Modbus RTU (slave only)

Port 2	Communications Port 2
	Connects to HPP, *Direct*SOFT, operator interfaces, etc.
	6-pin, multifunction port, RS232C
	Communication speed (baud): 300, 600, 1200, 2400, 4800, 9600, 19200, 38400
	Parity: odd (default), even, none
	Station Address: 1 (default)
	8 data bits
	1 start, 1 stop bit
	Asynchronous, half-duplex, DTE
	Protocol (auto-select): K-sequence (slave only), *Direct*NET (master/slave), Modbus RTU (master/slave), non-sequence/print

Networking
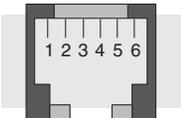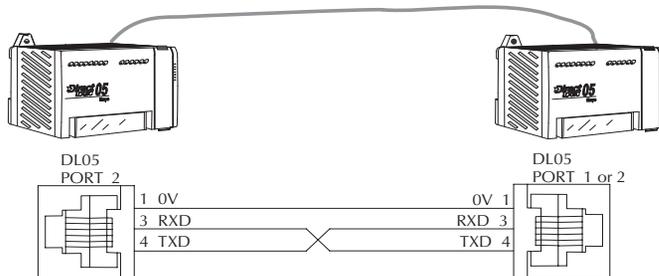
Port 1	Pin	Descriptions
1	0V	Power (–) connection (GND)
2	5V	Power (+) connection
3	RXD	Receive Data (RS232C)
4	TXD	Transmit Data (RS232C
5	5V	Power (+) connection
6	0V	Power (–) connection (GND)

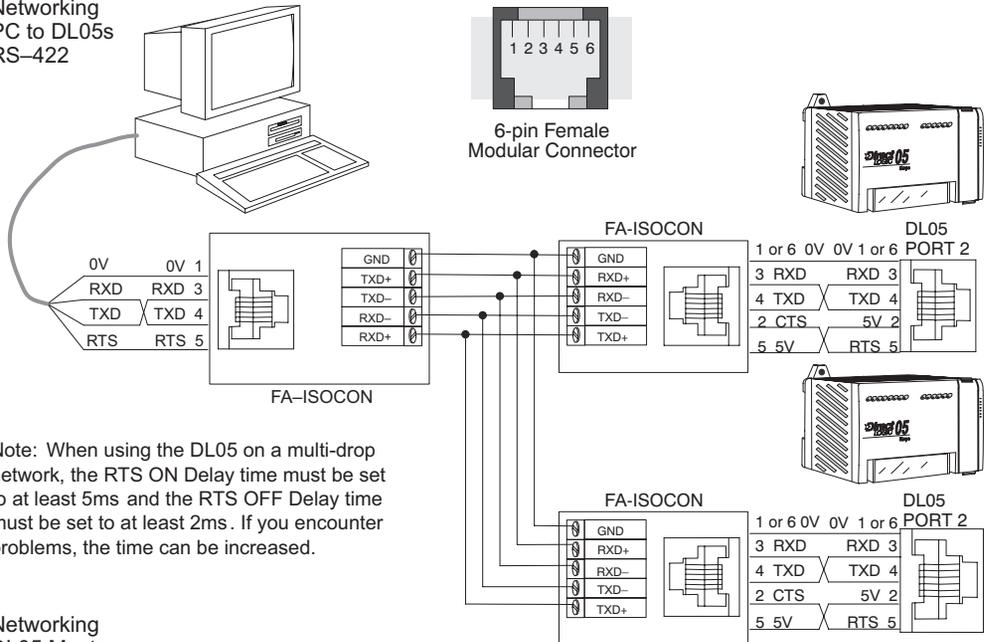| Port 2	Pin	Descriptions
1	0V	Power (–) connection (GND)
2	5V	Power (+) connection
3	RXD	Receive Data (RS232C)
4	TXD	Transmit Data (RS232C
5	RTS	Request to Send
6	0V	Power (–) connection (GND)

DL05 to DL05 RS-232C

You will need to make sure the network connection is a 3-wire RS–232 type. The recommended cable is AutomationDirect L19772 (Belden 8102) or equivalent. Normally, the RS–232 signals are used for communications between two devices with distances up to a maximum of 15 meters.
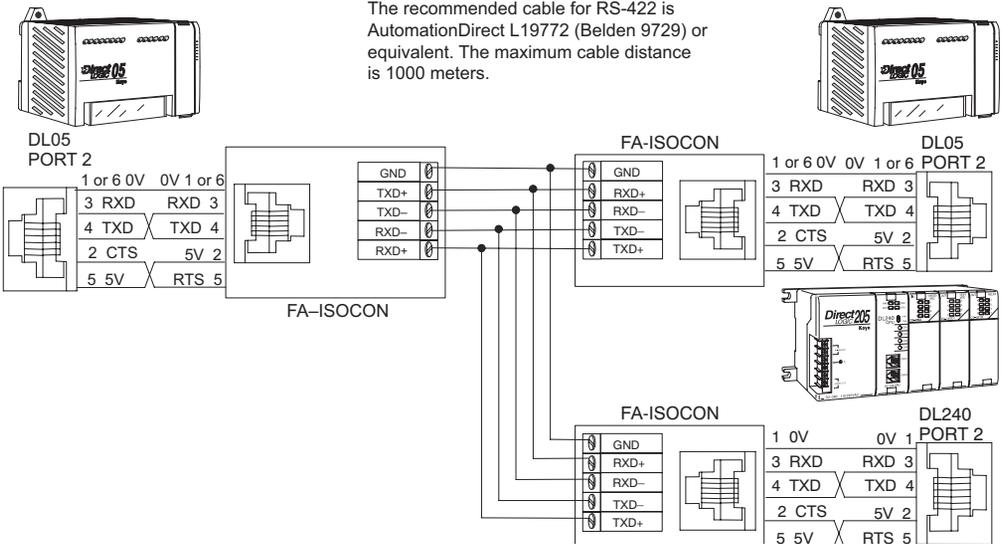
DL05 PORT 2

1	0V
3	RXD
4	TXD

DL05 PORT 1 or 2

0V	1
RXD	3
TXD	4

Networking Using RS–422 Converters

Networking
PC to DL05s
RS–422

6-pin Female
Modular Connector
1 2 3 4 5 6

FA-ISOCON

		FA-ISOCON			DL05 PORT 2
0V	0V 1	GND	GND	1 or 6 0V	0V 1 or 6
RXD	RXD 3	TXD+	RXD+	3 RXD	RXD 3
TXD	TXD 4	TXD–	RXD–	4 TXD	TXD 4
RTS	RTS 5	RXD–	TXD–	2 CTS	5V 2
		RXD+	TXD+	5 5V	RTS 5

FA–ISOCON

Note: When using the DL05 on a multi-drop network, the RTS ON Delay time must be set to at least 5ms and the RTS OFF Delay time must be set to at least 2ms. If you encounter problems, the time can be increased.

FA-ISOCON

	DL05 PORT 2
GND	1 or 6 0V 0V 1 or 6
RXD+	3 RXD RXD 3
RXD–	4 TXD TXD 4
TXD–	2 CTS 5V 2
TXD+	5 5V RTS 5

Networking
DL05 Master
to Other PLCs

The recommended cable for RS-422 is AutomationDirect L19772 (Belden 9729) or equivalent. The maximum cable distance is 1000 meters.

DL05
PORT 2

1 or 6 0V	0V 1 or 6	FA-ISOCON		FA-ISOCON		DL05 PORT 2
3 RXD	RXD 3	GND	GND			1 or 6 0V 0V 1 or 6
4 TXD	TXD 4	TXD+	RXD+			3 RXD RXD 3
2 CTS	5V 2	TXD–	RXD–			4 TXD TXD 4
5 5V	RTS 5	RXD–	TXD–			2 CTS 5V 2
		RXD+	TXD+			5 5V RTS 5

FA–ISOCON

FA-ISOCON

	DL240 PORT 2
GND	1 0V 0V 1
RXD+	3 RXD RXD 3
RXD–	4 TXD TXD 4
TXD–	2 CTS 5V 2
TXD+	5 5V RTS 5

Direct205

4

Modbus Port Configuration

In *Direct*SOFT 5, choose the PLC menu, then Setup, then "Secondary Comm Port".

- **Port**: From the port number list box at the top, choose "Port 2".

- **Protocol**: Click the check box to the left of "Modbus" (use AUX 56 on the HPP, and select "MBUS"), and then you'll see the dialog box below.
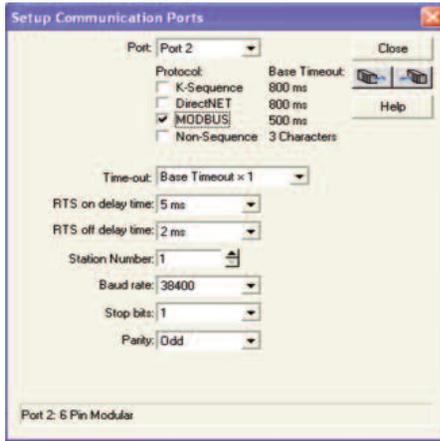
- **Timeout**: Amount of time the port will wait after it sends a message to get a response before logging an error.

- **RTS ON / OFF Delay Time:** The RTS ON Delay Time specifies the time the DL05 waits to send the data after it has raised the RTS signal line. The RTS OFF Delay Time specifies the time the DL05 waits to release the RTS signal line after the data has been sent. *When using the DL05 on a multi-drop network, the RTS ON Delay time must be set to at least **5ms** and the RTS OFF Delay time must be set to at least **2ms**. If you encounter problems, the time can be increased.*

- **Station Number**: The possible range for Modbus slave numbers is from 1 to 247, but the DL05 network instructions used in Master mode will access only slaves 1 to 99. Each slave must have a unique number. At powerup, the port is automatically a slave, unless and until the DL05 executes ladder logic network instructions which use the port as a master. Thereafter, the port reverts back to slave mode until ladder logic uses the port again.

- **Baud Rate**: The available baud rates include 300, 600, 1200, 2400, 4800, 9600, 19200, and 38400 baud. Choose a higher baud rate initially, reverting to lower baud rates if you experience data errors or noise problems on the network. Important: You must configure the baud rates of all devices on the network to the same value. Refer to the appropriate product manual for details.

- **Stop Bits**: Choose 1 or 2 stop bits for use in the protocol.

- **Parity**: Choose none, even, or odd parity for error checking.

Then click the button indicated to send the Port configuration to the CPU, and click Close.

*Direct*NET Port Configuration

In *Direct*SOFT 5, choose the PLC menu, then Setup, then "Secondary Comm Port".

- **Port**: From the port number list box, choose "Port 2 ".

- **Protocol**: Click the check box to the left of "*Direct*NET" (use AUX 56 on the HPP, then select "DNET"), and then you'll see the dialog box below.
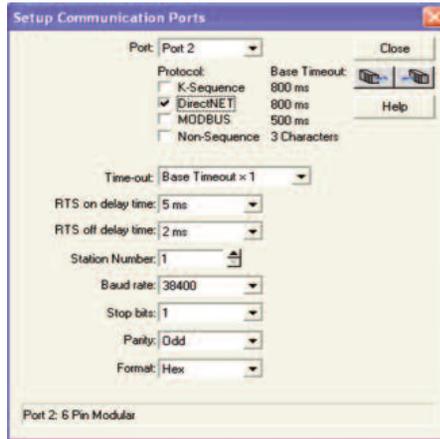
- **Timeout**: Amount of time the port will wait after it sends a message to get a response before logging an error.

- **RTS ON / OFF Delay Time:** The RTS ON Delay Time specifies the time the DL05 waits to send the data after it has raised the RTS signal line. The RTS OFF Delay Time specifies the time the DL05 waits to release the RTS signal line after the data has been sent. *When using the DL05 on a multi-drop network, the RTS ON Delay time must be set to at least **5ms** and the RTS OFF Delay time must be set to at least **2ms**. If you encounter problems, the time can be increased.*

- **Station Number:** For making the CPU port a *Direct*NET master, choose "1". The allowable range for *Direct*NET slaves is from 1 to 90 (each slave must have a unique number). At powerup, the port is automatically a slave, unless and until the DL05 executes ladder logic instructions which attempt to use the port as a master. Thereafter, the port reverts back to slave mode until ladder logic uses the port again.

- **Baud Rate:** The available baud rates include 300, 600, 1200, 2400, 4800, 9600, 19200, and 38400 baud. Choose a higher baud rate initially, reverting to lower baud rates if you experience data errors or noise problems on the network. Important: You must configure the baud rates of all devices on the network to the same value.

- **Stop Bits:** Choose 1 or 2 stop bits for use in the protocol.

- **Parity:** Choose none, even, or odd parity for error checking.

- **Format:** Choose between hex or ASCII formats.

Then click the button indicated to send the Port configuration to the CPU, and click Close.

Network Slave Operation

This section describes how other devices on a network can communicate with a CPU port that you have configured as a *Direct*NETslave or Modbus slave (DL05). A Modbus host must use the Modbus RTU protocol to communicate with the DL05 as a slave. The host software must send a Modbus function code and Modbus address to specify a PLC memory location the DL05 comprehends. The *Direct*NET host uses normal I/O addresses to access applicable DL05 CPU and system. No CPU ladder logic is required to support either Modbus slave or *Direct*NET slave operation.

Modbus Function Codes Supported

The Modbus function code determines whether the access is a read or a write, and whether to access a single data point or a group of them. The DL05 supports the Modbus function codes described below.

MODBUS Function Code	Function	DL05 Data Types Available
01	Read a group of coils	Y, CR, T, CT
02	Read a group of inputs	X, SP
05	Set / Reset a single coil	Y, CR, T, CT
15	Set / Reset a group of coils Y,	CR, T, CT
03, 04	Read a value from one or more registers	V
06	Write a value into a single register	V
16	Write a value into a group of registers	V

Determining the Modbus Address

There are typically two ways that most host software conventions allow you to specify a PLC memory location. These are:

• By specifying the Modbus data type and address

• By specifying a Modbus address only

*NOTE: For information about the Modbus protocol see the Group Schneider website at: www.schneiderautomation.com. At the main menu, select Support/Services, Modbus, Modbus Technical Manuals, PI-MBUS-300 Modbus Protocol Reference Guide or search for PIMBUS300. For more information about the **Direct**NET protocol, order our **Direct**NET User Manual, DA-DNET-M, or download the manual free from our website: www.automationdirect.com. Select Manuals\Docs\onlineusermanuals\misc.\DA-DNET-M*

4

If Your Host Software Requires the Data Type and Address...

Many host software packages allow you to specify the Modbus data type and the Modbus address that corresponds to the PLC memory location. This is the easiest method, but not all packages allow you to do it this way.

The actual equation used to calculate the address depends on the type of PLC data you are using. The PLC memory types are split into two categories for this purpose.

   • Discrete – X, SP, Y, CR, S, T, C (contacts)

   • Word – V, Timer current value, Counter current value

In either case, you basically convert the PLC octal address to decimal and add the appropriate Modbus address (if required). The table below shows the exact equation used for each group of data.

DL05 Memory Type	QTY (Dec.)	PLC Range(Octal	Modbus Address Range (Decimal)	Modbus Data Type
For Discrete Data Types …. Convert PLC Addr. to Dec. + Start of Range + Data Type				
Inputs (X)	256	X0 – X377	2048 – 2303	Input
Special Relays(SP)	512	SP0 – SP777	3072 – 3583	Input
Outputs (Y)	256	Y0 – Y377	2048 – 2303	Coil
Control Relays (CR)	512	C0 – C777	3072 – 4583	Coil
Timer Contacts (T)	128	T0 – T177	6144 – 6271	Coil
Counter Contacts (CT)	128	CT0 – CT177	6400 – 6527	Coil
Stage Status Bits(S)	256	S0 – S377	5120 – 5375	Coil
For Word Data Types …. Convert PLC Addr. to Dec. + Data Type				
Timer Current Values (V)	128	V0 – V177	0 – 127	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register
V-Memory, user data (V)	3968	V1200 – V7377	640 – 3839	Holding Register
V-Memory, non-volatile (V)	128	V7600 – V7777	3968 – 4095	Holding Register

The following examples show how to generate the Modbus address and data type for hosts which require this format.

Example 1: V2100

Find the Modbus address for User V location V2100.

1. Find V memory in the table.
2. Convert V2100 into decimal (1088).
3. Use the Modbus data type from the table.

PLC Address (Dec) + Data Type

V2100 = 1088 decimal

1088 + Hold. Reg. = ⎢ Holding Reg 1088 ⎥

V Memory, user data (V)	3200	V1200 – V7377	640 – 3839	Holding Register

Example 2: Y20

Find the Modbus address for output Y20.

1. Find Y outputs in the table.
2. Convert Y20 into decimal (16).
3. Add the starting address for the range (2048).
4. Use the Modbus data type from the table.

PLC Address (Dec) + Start Addr + Data Type

Y20 = 16 decimal

16 + 2048 + Coil = ⎢ Coil 2064 ⎥

Example 3: T10 Current Value

Outputs (V)	256	Y0 – Y377	2048 - 2303	Coil

Find the Modbus address to obtain the current value from Timer T10.

1. Find Timer Current Values in the table.
2. Convert T10 into decimal (8).
3. Use the Modbus data type from the table.

PLC Address (Dec) + Data Type

T10 = 8 decimal

8 + Input Reg. = ⎢ Input Reg. 8 ⎥

Example 4: C54

Timer Current Values (V)	128	V0 – V177	0 - 127	Input Register

Find the Modbus address for Control Relay C54.

1. Find Control Relays in the table.
2. Convert C54 into decimal (44).
3. Add the starting address for the range (3072).
4. Use the Modbus data type from the table.

PLC Address (Dec) + Start Addr. + Data Type

C54 = 44 decimal

44 + 3072 + Coil = ⎢ Coil 3116 ⎥

Control Relays (CR)	512	C0 – C77	3072 – 3583	Coil

If Your Modbus Host Software Requires an Address ONLY

Some host software does not allow you to specify the Modbus data type and address. Instead, you specify an address only. This method requires another step to determine the address, but it's still fairly simple. Basically, Modbus also separates the data types by address ranges as well. So this means an address alone can actually describe the type of data and location. This is often referred to as "adding the offset". One important thing to remember here is that two different addressing modes may be available in your host software package. These are:

- 484 Mode
- 584/984 Mode

We recommend that you use the 584/984 addressing mode if your host software allows you to choose. This is because the 584/984 mode allows access to a higher number of memory locations within each data type. If your software only supports 484 mode, then there may be some PLC memory locations that will be unavailable. The actual equation used to calculate the address depends on the type of PLC data you are using. The PLC memory types are split into two categories for this purpose.

- Discrete – X, SP, Y, CR, S, T (contacts), C (contacts)
- Word – V, Timer current value, Counter current value

In either case, you basically convert the PLC octal address to decimal and add the appropriate Modbus addresses (as required). The table below shows the exact equation used for each group of data.

Discrete Data Types				
DL05 Memory Type	PLC Range (Octal)	Address (484 Mode)	Address (584/984 Mode)	Modbus Data Type
Global Inputs (GX)	GX0-GX1746	1001 - 1999	10001 - 10999	Input
	GX1747-GX3777	---	11000 - 12048	Input
Inputs (X)	X0 – X1777	---	12049 - 13072	Input
Special Relays (SP)	SP0 – SP777	---	13073 - 13584	Input
Global Outputs (GY)	GY0 - GY3777	1 - 2048	1 - 2048	Output
Outputs (Y)	Y0 – Y1777	2049 - 3072	2049 - 3072	Output
Control Relays (CR)	C0 – C3777	3073 - 5120	3073 - 5120	Output
Timer Contacts (T)	T0 – T377	6145 - 6400	6145 - 6400	Output
Counter Contacts (CT)	CT0 – CT377	6401 - 6656	6401 - 6656	Output
Stage Status Bits (S)	S0 – S1777	5121 - 6144	5121 - 6144	Output

Word Data Types			
Registers	**PLC Range (Octal)**	**Input/Holding (484 Mode)***	**Input/Holding (584/984 Mode)***
V-Memory (Timers)	V0 - V377	3001/4001	30001/40001
V-Memory (Counters)	V1000 - V1177	3513/4513	30513/40513
V-Memory (Data Words)	V1200 - V1377	3641/4641	30641/40641
	V1400 - V1746	3769/4769	30769/40769
	V1747 - V1777	---	31000/41000
	V2000 - V7377	---	41025
	V10000 - V17777	---	44097

* Modbus: Function 04

The DL05/06, DL250-1/260, DL350 and DL450 will support function 04, read input register (Address 30001). To use function 04, put the number '4' into the most significant position (4xxx). Four digits must be entered for the instruction to work properly with this mode.
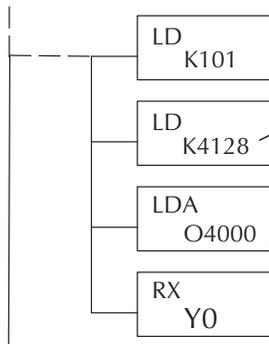
The Maximum constant possible is 4128. This is due to the 128 maximum number of Bytes that the RX/WX instruction can allow. The value of 4 in the most significant position of the word will cause the RX to use function 04 (30001 range).

1. Refer to your PLC user manual for the correct memory mapping size of your PLC. Some of the addresses shown above might not pertain to your particular CPU.

2. For an automated Modbus/Koyo address conversion utility, download the file **modbus_conversion.xls** from the **www.automationdirect.com** website.

Example 1: V2100 584/984 Mode

Find the Modbus address for user V-memory V2100.

1. Find V memory in the table.
2. Convert V2100 into decimal (1088).
3. Add the Modbus starting address for the mode (40001).

PLC Address (Dec) + Mode Address

V2100 = 1088 decimal

1088 + 40001 = | 41089 |

For Word Data Types….		PLC Address (Dec.)	+		Appropriate Mode Address	
Timer Current Values (V)	128	V0 – V177	0 – 127	3001	30001	Input Register
Counter Current Values (V)	128	V1200 – V7377	512 – 639	3001	30001	Input Register
V-Memory, user data (V)	1024	V2000 – V3777	1024 – 2047	4001	40001	Holding Register

Example 2: Y20 584/984 Mode

Find the Modbus address for output Y20.

1. Find Y outputs in the table.
2. Convert Y20 into decimal (16).
3. Add the starting address for the range (2048).
4. Add the Modbus address for the mode (1).

PLC Address (Dec) + Start Addr + Mode

Y20 = 16 decimal

16 + 2048 + 1 = | 2065 |

Outputs (Y)	320	Y0 - Y477	2048 - 2367	1	1	Coil
Control Relays (CR)	256	C0 - C377	3072 - 3551	1	1	Coil
Timer Contacts (T)	128	T0 - T177	6144 - 6271	1	1	Coil

Example 3: T10 Current Value 484 Mode

Find the Modbus address to obtain the current value for Timer T10.

1. Find Timer Current Values in the table.
2. Convert T10 into decimal (8).
3. Add the Modbus starting address for the mode (3001).

PLC Address (Dec) + Mode Address

T10 = 8 decimal

8 + 3001 = | 3009 |

For Word Data Types….		PLC Address (Dec.)	+		Appropriate Mode Address	
Timer Current Values (V)	128	V0 – V177	0 – 127	3001	30001	Input Register
Counter Current Values (V)	128	V1200 – V7377	512 – 639	3001	30001	Input Register
V-Memory, user data (V)	1024	V2000 – V3777	1024 – 2047	4001	40001	Holding Register

Example 4: C54 584/984 Mode

Find the Modbus address for Control Relay C54.

1. Find Control Relays in the table.
2. Convert C54 into decimal (44).
3. Add the starting address for the range (3072).
4. Add the Modbus address for the mode (1).

PLC Address (Dec) + Start Addr + Mode

C54 = 44 decimal

44 + 3072 + 1 = | 3117 |

| Outputs (Y)	320	Y0 - Y477	2048 - 2367	1	1	Coil
Control Relays (CR)	256	C0 - C377	3072 - 3551	1	1	Coil
Timer Contacts (T)	128	T0 - T177	6144 - 6271	1	1	Coil

4

Network Master Operation

This section describes how the DL05 PLC can communicate on a Modbus or *Direct*NET network as a master. For Modbus networks, it uses the Modbus RTU protocol, which must be interpreted by all the slaves on the network. Both Modbus and *Direct*NET are single master/multiple slave networks. The master is the only member of the network that can initiate requests on the network. This section teaches you how to design the required ladder logic for network master operation.
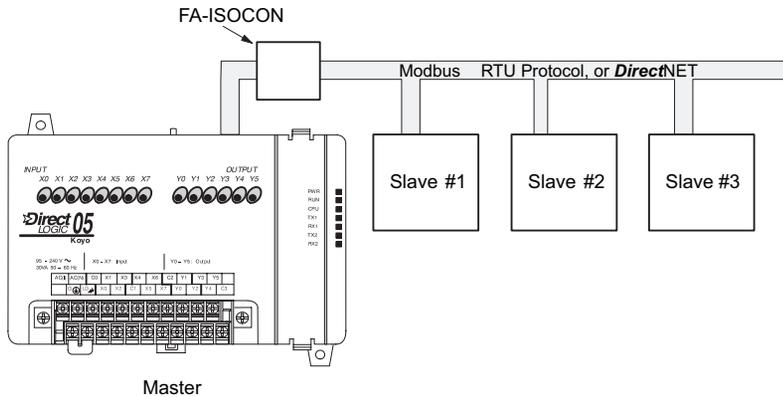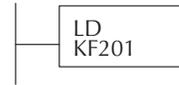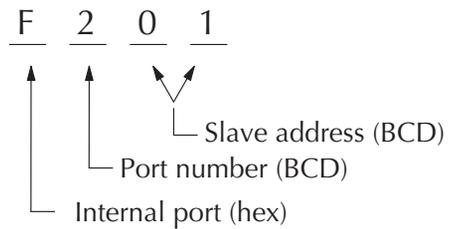

Master

When using the DL05 PLC as the master station, simple RLL instructions are used to initiate the requests. The WX instruction initiates network write operations, and the RX instruction initiates network read operations. Before executing either the WX or RX commands, we will need to load data related to the read or write operation onto the CPU's accumulator stack. When the WX or RX instruction executes, it uses the information on the stack combined with data in the instruction box to completely define the task, which goes to the port.
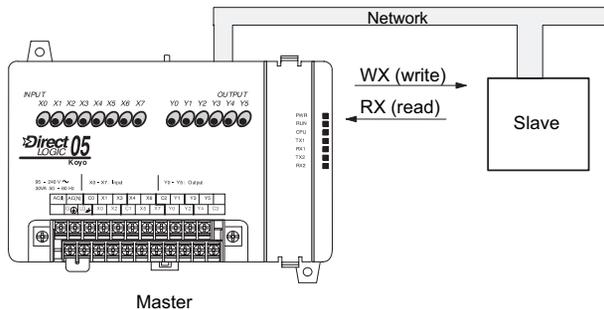
Master

The following step-by-step procedure will provide you the information necessary to set up your ladder program to receive data from a network slave.

Step 1: Identify Master Port # and Slave

The first Load (LD) instruction identifies the communications port number on the network master (DL05) and the address of the slave station. This instruction can address up to 99 Modbus slaves, or 90 *Direct*NET slaves. The format of the word is shown to the right. The "F2" in the upper byte indicates the use of the right port of the DL05 PLC, port number 2. The lower byte contains the slave address number in BCD (01 to 99).

F 2 0 1

Slave address (BCD)
Port number (BCD)
Internal port (hex)

LD
KF201

Step 2: Load Number of Bytes to Transfer

The second Load (LD) instruction determines the number of bytes which will be transferred between the master and slave in the subsequent WX or RX instruction. The value to be loaded is in BCD format (decimal), from 1 to 128 bytes.

6 4   (BCD)

# of bytes to transfer

LD
K64
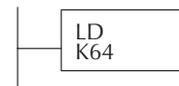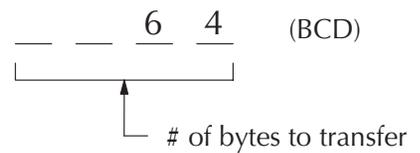
The number of bytes specified also depends on the type of data you want to obtain. For example, the DL05 Input points can be accessed by V-memory locations or as X input locations. However, if you only want X0 – X27, you'll have to use the X input data type because the V-memory locations can only be accessed in 2-byte increments. The following table shows the byte ranges for the various types of *Direct*LOGIC™ products.

DL 05/205/350/405 Memory	Bits per unit	Bytes
V-memory T / C current value	16 16	2 2
Inputs (X, SP)	8	1
Outputs (Y, C, Stage, T/C bits)	8	1
Scratch Pad Memory	8	1
Diagnostic Status	8	1

DL330/340 Memory	Bits per unit	Bytes
Data registers T / C accumulator	8 16	1 2
I/O, internal relays, shift register bits, T/C bits, stage bits	1	1
Scratch Pad Memory	8	1
Diagnostic Status(5 word R/W)	16	10

Step 3: Specify Master Memory Area

4 0 6 0 0   (octal)

The third instruction in the RX or WX sequence is a Load Address (LDA) instruction. Its purpose is to load the starting address of the memory area to be transferred. Entered as an octal number, the LDA instruction converts it to hex and places the result in the accumulator.

Starting address of master transfer area

LDA
O40600

For a WX instruction, the DL05 CPU sends the number of bytes previously specified from its memory area beginning at the LDA address specified.

For an RX instruction, the DL05 CPU reads the number of bytes previously specified from the slave, placing the received data into its memory area beginning at the LDA address specified.

MSB   V40600   LSB
15   0

MSB   V40601   LSB
15   0
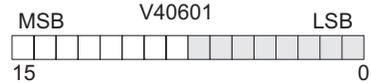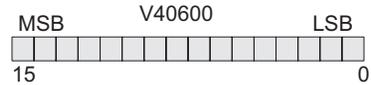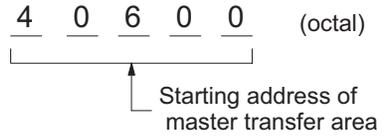
NOTE: *Since V-memory words are always 16 bits, you may not always use the whole word. For example, if you only specify 3 bytes and you are reading Y outputs from the slave, you will only get 24 bits of data. In this case, only the 8 least significant bits of the last word location will be modified. The remaining 8 bits are not affected.*

Step 4: Specify Slave Memory Area

The last instruction in our sequence is the WX or RX instruction itself. Use WX to write to the slave, and RX to read from the slave. All four of our instructions are shown to the right. In the last instruction, you must specify the starting address and a valid data type for the slave.

SP116

LD
KF201

LD
K64

LDA
O40600

RX
Y0

- *Direct*NET slaves – specify the same address in the WX and RX instruction as the slave's native I/O address

- Modbus DL405, DL205, or DL05 slaves – specify the same address in the WX and RX instruction as the slave's native I/O address

- Modbus 305 slaves – use the following table to convert DL305 addresses to Modbus addresses
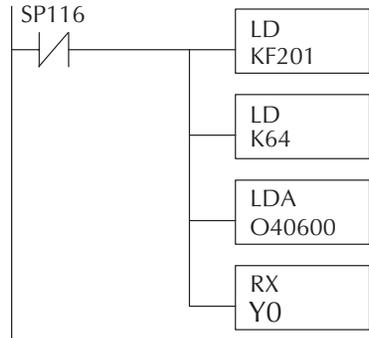
DL305 Series CPU Memory Type–to–Modbus Cross Reference (excluding 350 CPU)					
PLC Memory Type	PLC Base Address	Modbus Base Address	PLC Memory Type	PLC Base Address	Modbus Base Address
TMR/CNT Current Values	R600	V0	TMR/CNT Status Bits	CT600	GY600
I/O Points	IO 000	GY0	Control Relays	CR160	GY160
Data Registers	R401,R400	V100	Shift Registers	SR400	GY400
Stage Status Bits (D3-330P only)	S0	GY200			

Communications from a Ladder Program

Typically network communications will last longer than 1 scan. The program must wait for the communications to finish before starting the next transaction.

Port 2, which can be a master, has two Special Relay contacts associated with it (see Appendix D for comm port special relays). One indicates "Port busy"(SP116), and the other indicates "Port Communication Error"(SP117). The example above shows the use of these contacts for a network master that only reads a device (RX). The "Port Busy" bit is on while the PLC communicates with the slave. When the bit is off the program can initiate the next network request.

Port Communication Error

Port Busy

SP117		Y1 (SET)
SP116		LD KF201
		LD K0003
		LDA O40600
		RX Y0
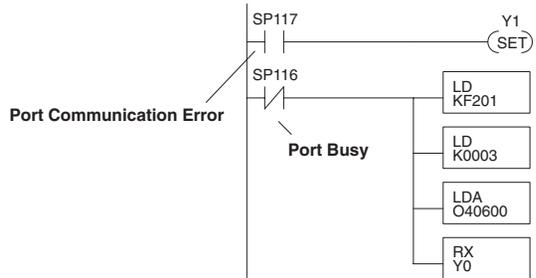
The "Port Communication Error" bit turns on when the PLC has detected an error. Use of this bit is optional. When used, it should be ahead of any network instruction boxes since the error bit is reset when an RX or WX instruction is executed.

Multiple Read and Write Interlocks

If you are using multiple reads and writes in the RLL program, you have to interlock the routines to make sure all the routines are executed. If you don't use the interlocks, then the CPU will only execute the first routine. This is because each port can only handle one transaction at a time.

In the example to the right, after the RX instruction is executed, C0 is set. When the port has finished the communication task, the second routine is executed and C0 is reset.

Interlocking Relay

SP116	C100	LD KF201
		LD K0003
		LDA O40600
		RX VX0
		C100 (SET)

Interlocking Relay

SP116	C100	LD KF201
		LD K0003
		LDA O40400
		WX VY0
		C100 (RST)