*Direct*NET ™

Manual Number DA–DNET–M

WARNING

Thank you for purchasing automation equipment from PLC***Direct***™. We want your new ***Direct***LOGIC™ automation equipment to operate safely. Anyone who installs or uses this equipment should read this publication (and any other relevant publications) before installing or operating the equipment.

To minimize the risk of potential safety problems, you should follow all applicable local and national codes that regulate the installation and operation of your equipment. These codes vary from area to area and usually change with time. It is your responsibility to determine which codes should be followed, and to verify that the equipment, installation, and operation is in compliance with the latest revision of these codes.

At a minimum, you should follow all applicable sections of the National Fire Code, National Electrical Code, and the codes of the National Electrical Manufacturer's Association (NEMA). There may be local regulatory or government offices that can also help determine which codes and standards are necessary for safe installation and operation.

Equipment damage or serious injury to personnel can result from the failure to follow all applicable codes and standards. We do not guarantee the products described in this publication are suitable for your particular application, nor do we assume any responsibility for your product design, installation, or operation.

If you have any questions concerning the installation or operation of this equipment, or if you need additional information, please call us at 1–800–633–0405.

This publication is based on information that was available at the time it was printed. At PLC***Direct***™ we constantly strive to improve our products and services, so we reserve the right to make changes to the products and/or publications at any time without notice and without any obligation. This publication may also discuss features that may not be available in certain revisions of the product.

Trademarks

This publication may contain references to products produced and/or offered by other companies. The product and company names may be trademarked and are the sole property of their respective owners. PLC***Direct***™ disclaims any proprietary interest in the marks and names of others.

Stage is a trademark of Koyo Electronics Industries Co., LTD. Think & Do Software is a trademark of Think & Do Software, Inc. Texas Instruments is a registered trademark of Texas Instruments, Inc. TI, TIWAY, Series 305, Series 405, TI305, and TI405 are trademarks of Texas Instruments, Inc. Siemens and SIMATIC are registered trademarks of Siemens, AG. GE is a registered trademark of General Electric Corporation. Series One is a registered trademark of GE Fanuc Automation North America, Inc. MODBUS is a registered trademark of Gould, Inc. IBM is a registered trademark of International Business Machines. MS-DOS and Microsoft are registered trademarks of Microsoft Corporation. Windows and Windows NT are trademarks of Microsoft Corporation. OPTOMUX and PAMUX are trademarks of OPTO 22.

Manual Revisions

If you contact us in reference to this manual, be sure and include the revision number.

Title: ***Direct***NET Reference Manual
Manual Number: DA–DNET–M

Issue	Date	Effective Pages	Description of Changes
Original	1/94	Cover/Copyright Contents	Original Issue
Rev. A	6/98	Manual History 1-1 – 1-3 2-1 – 2-12 3-1 – 3-30 4-1 – 4-14 5-1 – 5-10 6-1 – 6-21 7-1 – 7-7 A-1 – A-18 B-1 – B-16 C-1 – C-15 D-1 – D-17 E-1 – E-16 F-1 – F-12	Made minor corrections throughout.

Table of Contents

i

Chapter 1: How to Use this Manual

Chapter 2: Getting Started with *Direct*NET Communications

Chapter 3: Network Design and Setup

Chapter 4: Communication Settings

Chapter 5: RLL Communications Programs

Chapter 6: *Direct*NET Host Communications Programs

Chapter 7: Network Operation and Troubleshooting

Appendix A: PLC Master / Slave Example

Appendix B: Peer Master Example

Appendix C: Host Master Example

Appendix D: DL405 Data Types and Memory Maps

Appendix E: DL305 Data Types and Memory Map

Appendix F: DL205 Data Types and Memory Map

1

How to Use this Manual

In This Chapter. . . .

— Introduction

Introduction

The purpose of this manual

The purpose of our manual is simple. We want to make it easy get your ***Direct***NET network up and running as soon as possible. Since we understand your time is money, our goal is to keep the documentation simple and concise.

This manual focuses on the details of the ***Direct***NET network. The manual provides a general overview of ***Direct***NET along with an explanation of all the steps required to implement a network.

Since we constantly try to improve our product line, we occasionally issue addenda documenting new features and changes to the products. If there are addenda included with this manual, please read through them to see which areas of the manual or product are affected.

Who should read this manual

If you understand data networks this manual will provide all the information you need to get and keep your network up and running. We will use examples and explanations from time to time to clarify our meaning and perhaps help you brush up on specific features used in ***Direct***NET communications. This manual is not intended to be a generic network training manual, but rather a user reference manual for ***Direct***NET.

Where to begin

If you are in a hurry and already understand the basics of master / slave networks, use the chart on the next page and proceed on to the chapter that pertains to your needs. Be sure to keep this manual handy for reference when you run into questions. If you are new to networking, we suggest starting at Chapter 2, so you can understand the wide variety of network components and configurations available with ***Direct***NET. We believe you will be pleasantly surprised with how much can be accomplished with ***Direct***NET.

How this manual is organized

Ch 2: Getting Started with *Direct*NET – provides an overview of ***Direct***NET and all the network components and configurations. It does not provide extensive information on each network component, but instead highlights some of the basic features of ***Direct***NET.

Ch 3: Network Design and Setup – shows how to prepare for network installation, and gives you cable diagrams for the various types of network master and slave stations.

Ch 4: Communication Parameters – provides detailed information about setting the master and slave station communication parameters. This is the place to look if you need to know how to set the communication baud rate, station addresses, etc.

Ch 5: RLL Communications Programs – provides a description of the Relay Ladder Logic (RLL) instructions used to initiate communications in a network that uses a PLC as the master station.

Ch 6: *Direct*Link Communications Programs – If you're writing your own host software program or operator interface driver, you'll find the ***Direct***NET protocol information in this chapter.

Ch 7: Network Operation and Troubleshooting – provides information on network startup and troubleshooting techniques.

Appendix A – provides a step by step example of a PLC master / slave network.

Appendix B – provides a step by step example of a PLC peer master network.

Appendix C – provides a step by step example of a host master network.

Appendix D – explains the data types and memory ranges for the DL405 products.

Appendix E – explains the data types and memory ranges for the DL305 products.

Appendix F – explains the data types and memory ranges for the DL205 products.

Supplemental Manuals

There are other manuals occasionally referenced by this manual. These manuals are not absolutely necessary to install the network, but they do provide additional details on several products that can be used with ***Direct***NET.

- DL405 User Manual (D4–USER–M)
- DL405 Data Communication Modules (D4–DCM–M)
- DL305 User Manual (D3–USER–M)
- DL305 Data Communications Unit (D3–DCU–M)
- DL205 User Manual (D2–User–M)

Now, you have the material necessary to quickly understand the features and requirements of ***Direct***NET. So, let's get started!

2

Getting Started with *Direct*NET Communications

In This Chapter. . . .

— The *Direct*NET Basics
— What can I use as a network master station?
— What can I use as a network slave station?
— Choosing a Configuration
— What communications program should I use?
— How can I create a network?

The *Direct*NET Basics

It's an Easy-to-use Data NETwork

***Direct*NET is an easy-to-use data network for the *Direct*LOGIC™ family of products. *Direct*NET is the perfect choice for those applications requiring data to be shared between programmable controllers (PLCs) or, between PLCs and a host computer. This network operates at speeds up to 38.4Kbps and permits you to upload or download virtually any type of system data. Some examples are:**

- Timer / Counter Data
- I/O Information
- Variable Memory Information (V memory, registers, etc.)

It has Advantages...

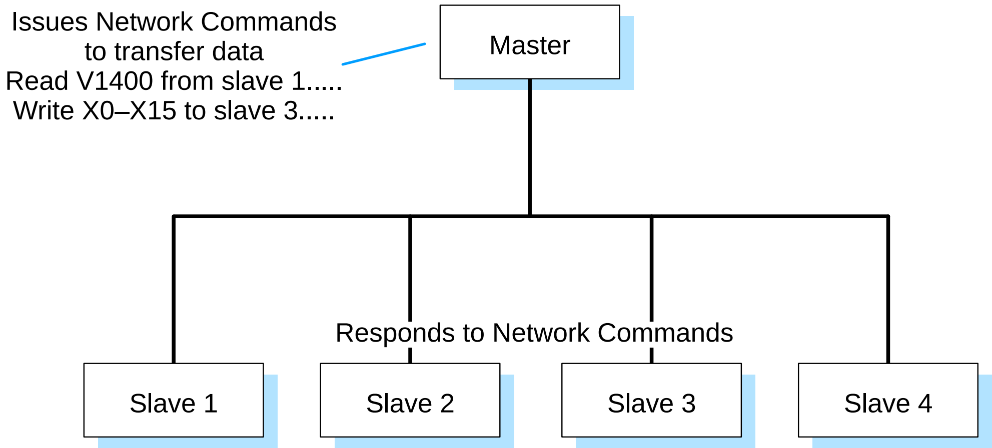
***Direct*NET is a great choice for sharing small amounts of data between network devices and has many advantages. You can build a data network with a minimal investment in equipment and training because:**

- The DL430, DL440, DL340, and DL240 Central Processing Units (CPUs) have built-in ***Direct*NET ports.**
- In the most common applications there is no need for complex communications programming. Simple Relay Ladder Logic instructions can be used to manage the communications, so you don't have to be a networking guru to start moving data around.

It has Disadvantages...

***Direct*NET operates asynchronously at speeds up to 38.4K baud. Therefore, it is not the best choice in an application requiring extremely fast update times and moving large amounts of data.**

How Does it Work? The network is controlled by a master station that issues network commands to individual slave stations (you cannot “broadcast” a message to all slaves). The commands may be used to download data to the slave stations, or upload data from the slave stations. The slave stations only respond to requests from the master station and cannot initiate communications.



Network Protocol ***Direct***NET uses the ***Direct***NET communications protocol, but you don’t have to understand the protocol to build the most common network configurations. A description of the ***Direct***NET protocol is included in Chapter 6 – ***Direct***NET Communication Programs.

Network Specifications

Maximum Number of Slaves	90 per master (RS422 Amplifier required for over 16 stations)
Configurations	Point-to-point, multi-drop
Interface Type	Serial RS232C / RS422 Half-duplex, Asynchronous operation
Transmission Rate	300 to 38.4K baud
Transmission Distance	3300 Feet (1000 meters)
Protocol	***Direct***NET

What can I use as a network master station?

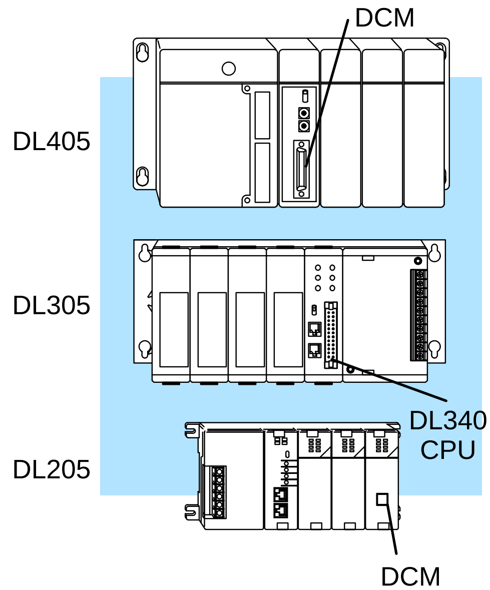
DL405/DL205 PLCs with a DCM or DL340 CPU

The master stations can be any one of the following:

- DL430 or DL440 PLCs with a DCM
- DL340 CPU (using built-in port)
- DL240 PLC with a DCM

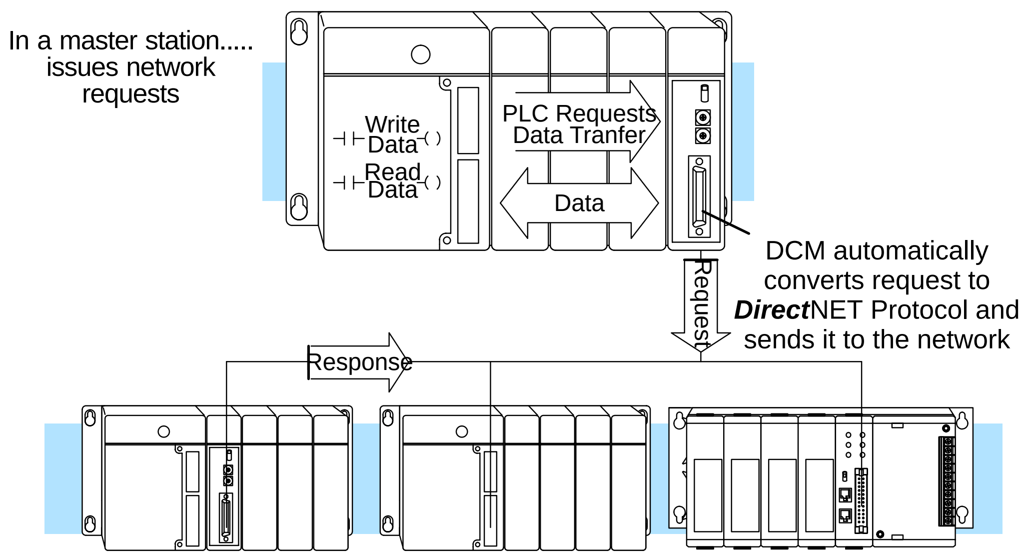
(A DCM is a Data Communication Module.)

These master stations use Relay Ladder Logic Instructions to initiate the communications requests over the network. No complex programming (or knowledge of the ***Direct***NET protocol) is required.



NOTE: The DL205 DCM was not available for shipment at the time of this publication. Availability will be announced at a later date.

The DCM has no internal program and only processes the data transfer requests from the master station CPU. The DCM automatically converts the requests into the appropriate ***Direct***NET commands and issues them to the network.



Operator Interface

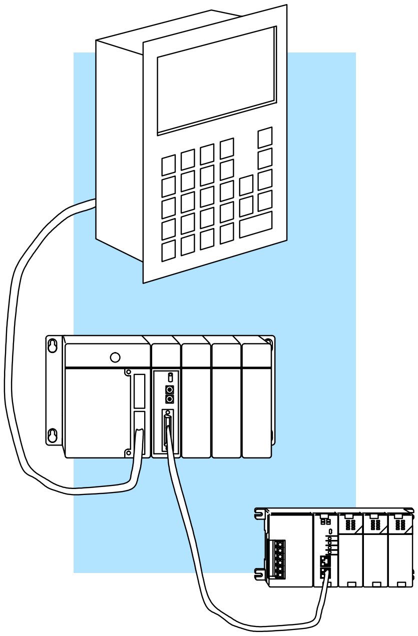
Many operator interfaces can be used as a network master station to communicate with one or more slave stations. An operator interface must have a driver that supports the ***Direct***NET protocol.

Make sure your operator interface uses one of the following protocols or has a driver for the equipment listed.

- ***Direct***NET (DL430, DL440, D4–DCM)
- Hostlink (TI™ or Simatic® TI425, -430, -435, U-01DM)
- CCM2 (GE® Series One™)

Many companies advertise some of their operator interface products will operate with the ***Direct***LOGIC™ family of products. Check with your operator interface supplier to see if they offer versions for use with the ***Direct***LOGIC™ products.

Operator Interface

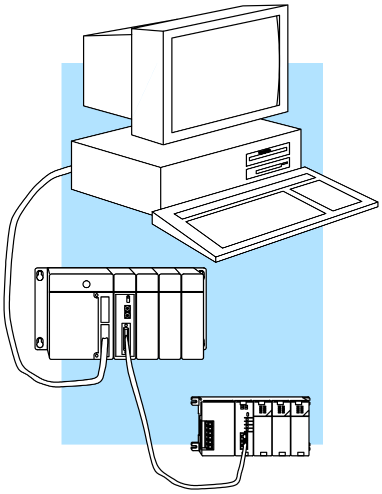


Host Computer

Many applications utilize a computer to act as a central data collection point and to manage the network requests for data transfer. The advantage with this approach is you can perform many different types of operations with the data either before or after a request.

The disadvantage is you may have to write your own communications driver and that can take a considerable amount of time for more complex applications. The communications programming is difficult unless you have a considerable amount of network programming experience. The programs must utilize the ***Direct***NET protocol used by ***Direct***NET.

Host Computer



What can I use as a network slave station?

CPUs with Built-in *Direct*NET Port

One of the benefits of ***Direct***NET is many of the ***Direct***LOGIC™ CPUs already have built-in ***Direct***NET ports that can communicate up to 19.2K baud, (38.4K for the DL340). The following CPUs have this feature.

- DL430, DL440
- DL340
- DL240

Data Communications Modules and Data Communications Units

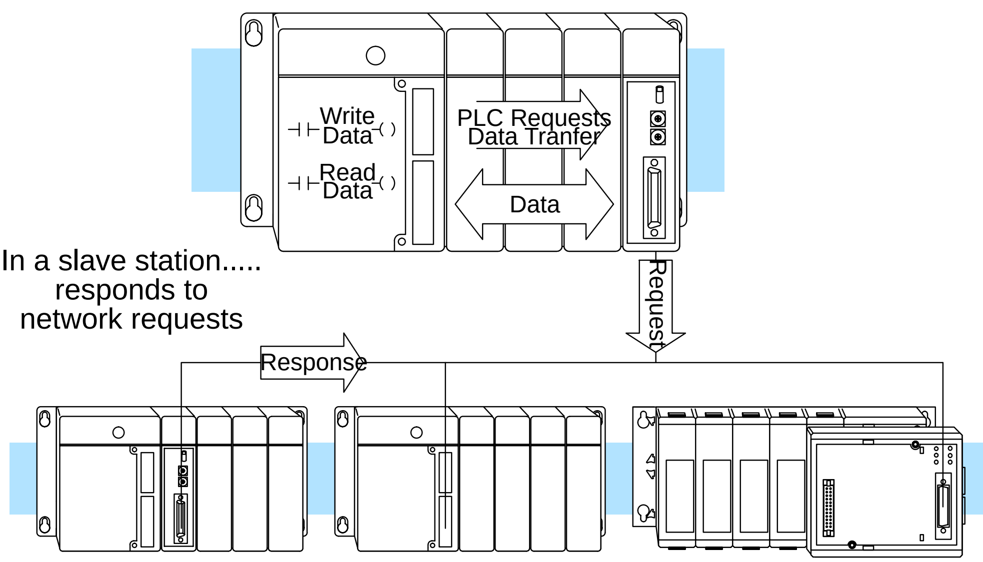
Each ***Direct***LOGIC™ product family has a general purpose communications interface. A DCM for the DL405 and DL205 CPUs and a DCU for the DL305 CPUs. You can use any of the ***Direct***LOGIC™ PLCs with their appropriate communications interface as a slave station. A few reasons to use a communication interface are:

- The communication interfaces (except for the DL305 DCU) allow baud rates of up to 38.4K baud.
- You can leave the CPU built-in port open to attach an operator interface or programming device.

The following communications interfaces are available.

- D4–DCM (with DL405 CPUs)
- D2–DCM (with DL205 CPUs)
- D3–232–DCU and D3–422–DCU (with DL305 CPUs)

In a slave station, the DCM responds to requests from the network master station.



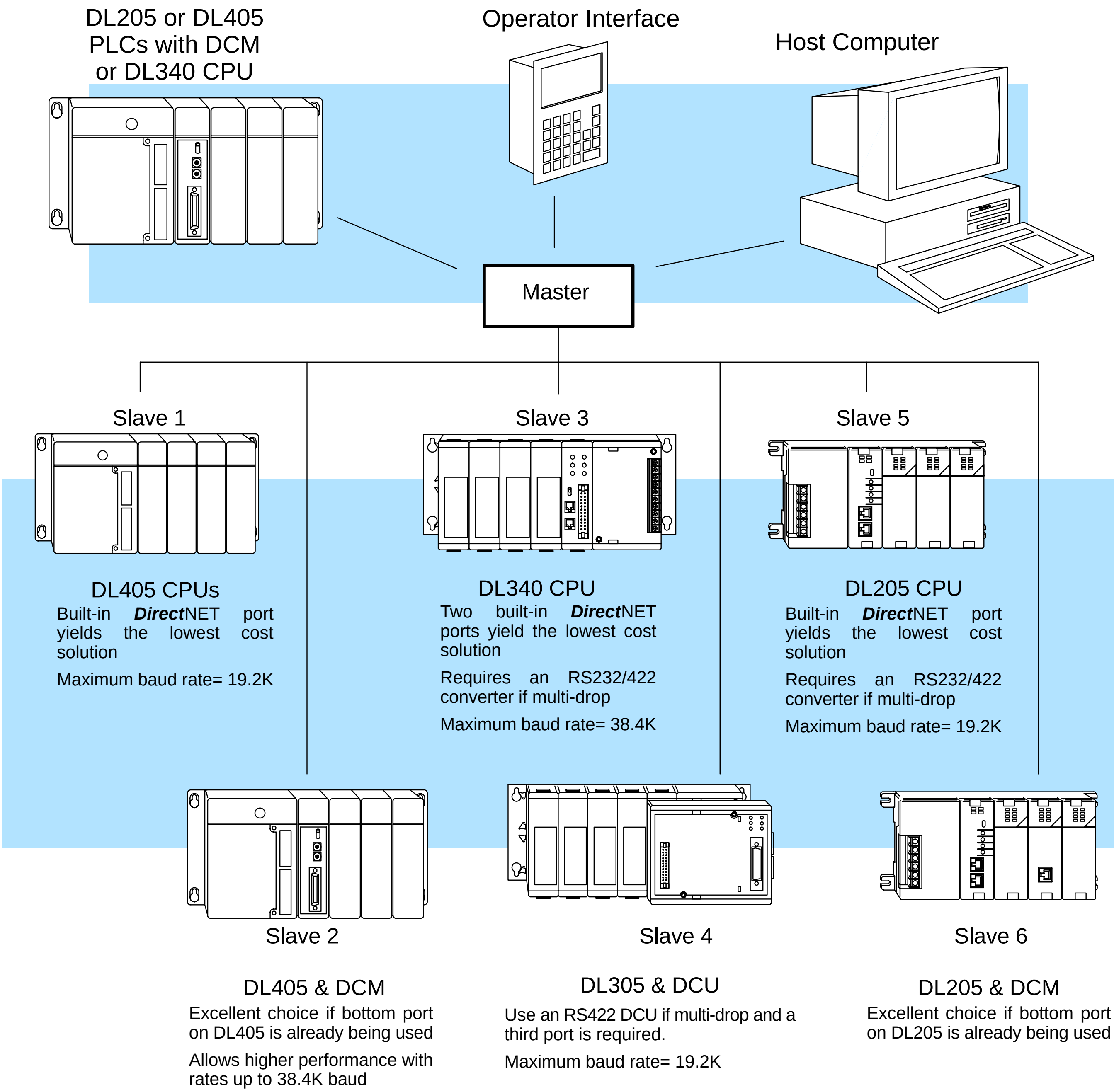
(a DCU may or may not be required depending on the DL305 CPU model chosen.

Compatible Products

There are also many compatible products from the following families that will communicate over ***Direct***NET.

- GE® Series 1®
- Texas Instruments TI305® and TI405®
- Simatic® TI305 and TI405

The following diagram shows the various combinations of master and slave stations available with a ***Direct*NET** solution.



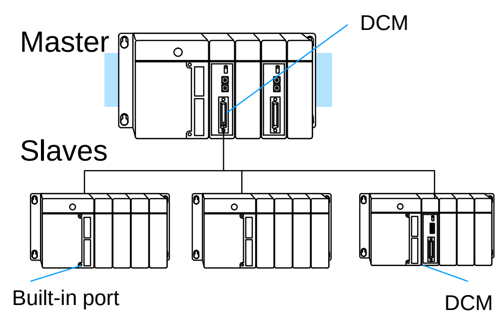
Choosing a Configuration

Three Basic Configurations

The network components can be used to create many different network designs. However, all networks use a mixture of three basic configurations. Any of the three configurations can be grouped together or used independently. The type of configuration determines the types of cables, communication parameters, and communications programming.

PLC as Master

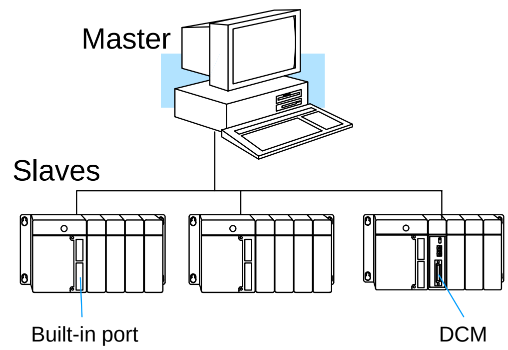
This configuration is very easy to use. A PLC master / slave configuration utilizes a ***Direct***LOGICE PLC as the network master. A small communications program and a communications interface (DCM) is required in the master station. Slave stations can have built-in ***Direct***NET ports, or, they can have communication interfaces.



Host as Master

A host computer, operator interface, or other intelligent device can be connected to one or more slave stations. The master must contain a program (or driver) capable of issuing network requests using the ***Direct***NET protocol.

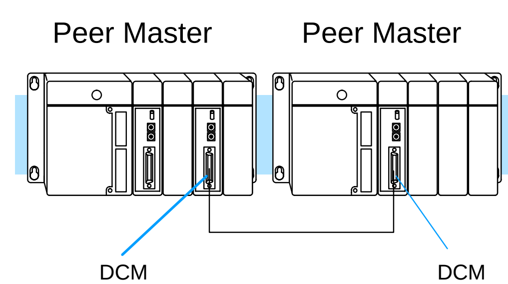
Slave stations can have built-in ***Direct***NET ports, or, they can have communication interfaces (DCMs or DCUs).



Peer as Master

Two DL405 PLCs with DCMs as the network interface can be connected so either station can initiate a request for data. Both stations must contain a communications program to initiate the requests for data.

This configuration can only be achieved by using DCMs. No other stations can be connected in this manner.



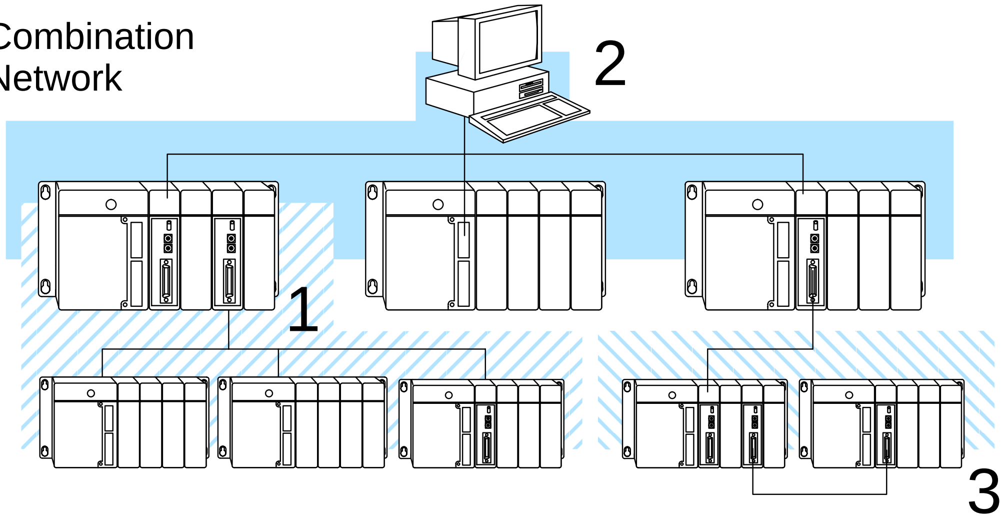
Configuration Limits

- Up to 90 slave stations can be connected to a single master.
- Only one master is allowed for each portion of the network.
- A single base can have multiple masters, with each master communicating to its own slave stations.

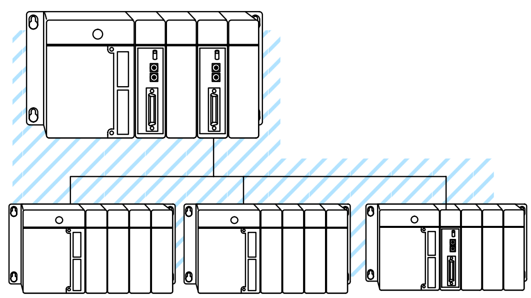
Combination Networks

The three types of networks can be combined to solve many different applications. Even though the configurations can be combined in an application, each network remains independent. The master station from one network cannot request data directly from slave stations on another network. This does not mean you cannot obtain data from these networks, you can. It just requires more than one interface for that slave station. The following diagram shows a simple application using all three types of networks, some with multiple interfaces for each station.

Combination Network

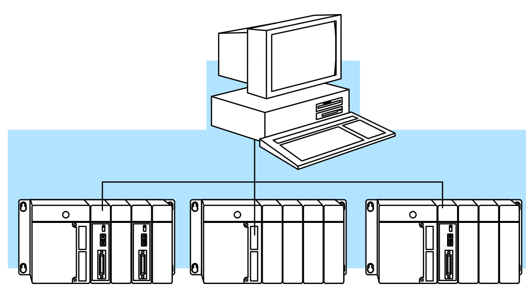


1 – PLC as Master Network



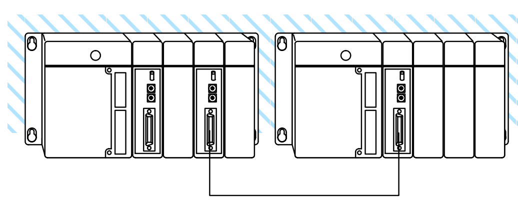
- Master can initiate read or write with any slave station
- PLC with DCM as master
- Full range of slave stations available

2 – Host as Master Network



- Host computer (or other intelligent device) as master
- Requires ***Direct***NET program in the host
- Full range of slave stations available

3 – Peer as Master Network



- Two stations only
- Only DL405 DCMs
- Either station can initiate a request

What communications program should I use?

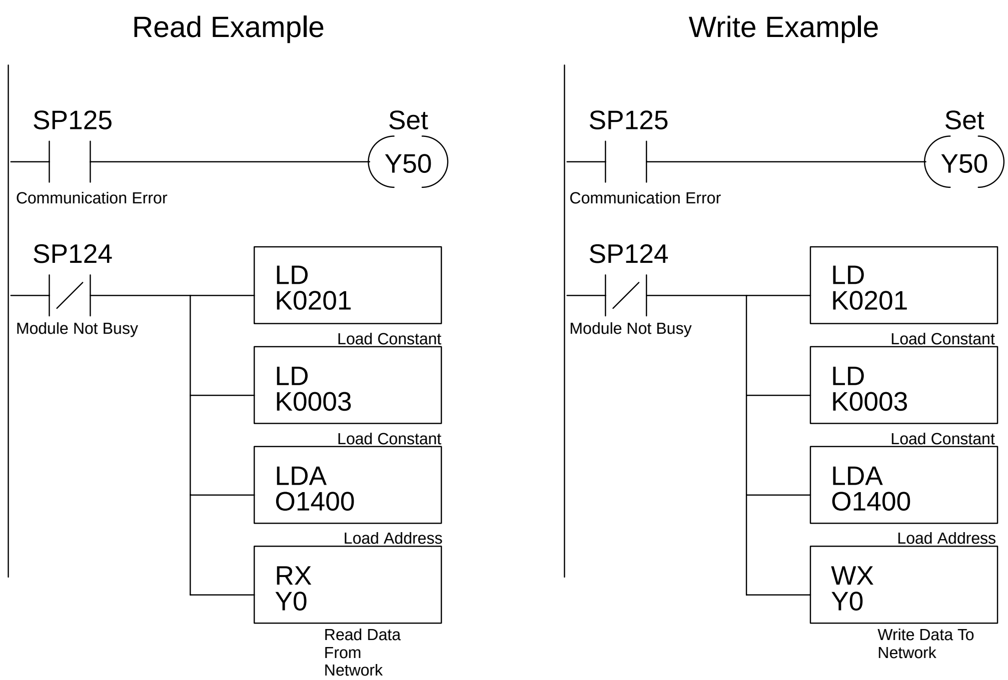
Two Options

There are two types of communications programs, RLL instructions or ***Direct***NET protocol programs. The network configuration determines the type of communications program required.

- PLC as Master — The PLC master requires instructions in the RLL program to initiate the requests for data.
- Peer as Master — Both peer masters require instructions in the RLL program. Since both stations contain network instructions, either station can initiate a request for data.
- Host as Master — Host computers or operator interfaces must execute communications programs that can issue network commands with the ***Direct***NET protocol. You'll either have to use a DL405 driver, (that hopefully came with your host software package or operator interface), or you'll have to create one.

RLL Programs

The RLL instructions are used to describe and initiate the operation the DCM will process. Here's an example of a simple RLL communications program. Chapter 4 provides a detailed description of the instructions.



*Direct*NET Programs

The communications program used with a hosted network is more complex than the simple RLL instructions used with the other configurations, but the concept is the same. The host is the ***Direct***NET master and must use a ***Direct***NET protocol communications program to initiate all network requests to read or write data. These communication programs can be written in many different languages, such as BASIC, C, etc. and must include the appropriate ***Direct***NET protocol commands.

Here's an example of a simple ***Direct***NET program. Chapter 5 provides a detailed description of the ***Direct***NET protocol.

DirectNET Program in BASIC

```
10    REM Program to read X0–X7 from a DL405 PLC
20    REM
30    REM Define all variables
40    REM
50    REM Change the slave address in HEX at line 60 if required.
60    SLAVEADDRESS=&H1
70    DATATYPE$=CHR$(&H32)
80    DATAADDR$=CHR$(&H30)+CHR$(&H31)+CHR$(&H30)+CHR$(&H31)
90    COMPLETEBLK$=CHR$(&H30)+CHR$(&H30)
100   PARTBLK$=CHR$(&H30)+CHR$(&H32)
110   MASTERADDR$=CHR$(&H30)+CHR$(&H30)
120   NORMAL$=CHR$(&H4E)
130   SLAVEADDR$=HEX$(SLAVEADDRESS)
140   IF LEN(SLAVEADDR$)<2 THEN SLAVEADDR$=”0”+SLAVEADDR$
150   OFFSETADDR$=CHR$(&H20+SLAVEADDRESS)
|
|
|
```

Getting Started

How can I create a network?

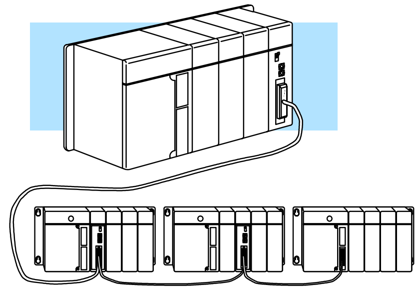
Follow Four Simple Steps

You can easily create a ***Direct***NET network by following four simple steps. The remainder of this manual provides explanations and examples of these steps.

1. Design the network by:
   - Selecting the configuration(s)
   - Building the communication cables.

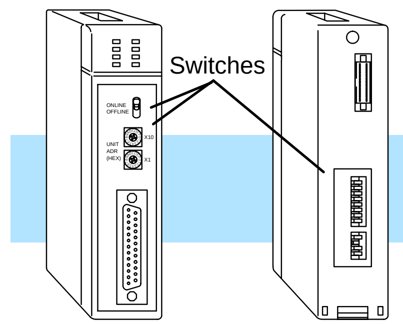
   Don't ignore the importance of this step because it determines the type of switch settings and communications program you should use. Chapter 3 provides the details.

Configuration and Cables

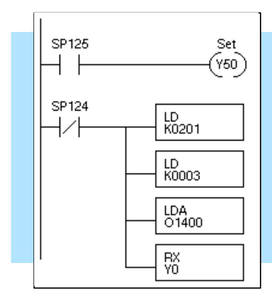


2. Select the communication settings by:
   - Setting the master switches
   - Setting the slave switches

   Chapter 4 provides switch settings.



Write Program



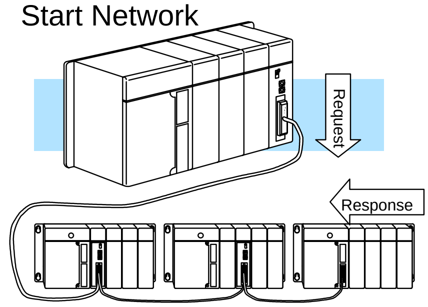
3. Write the communication control program.
   - RLL program with PLC master
     or
   - ***Direct***NET program with host master

   Chapters 5 and 6 provide programming information.

4. Start the network operation.

   Chapter 7 provides a description of network operation.

Start Network



3

Network Design and Setup

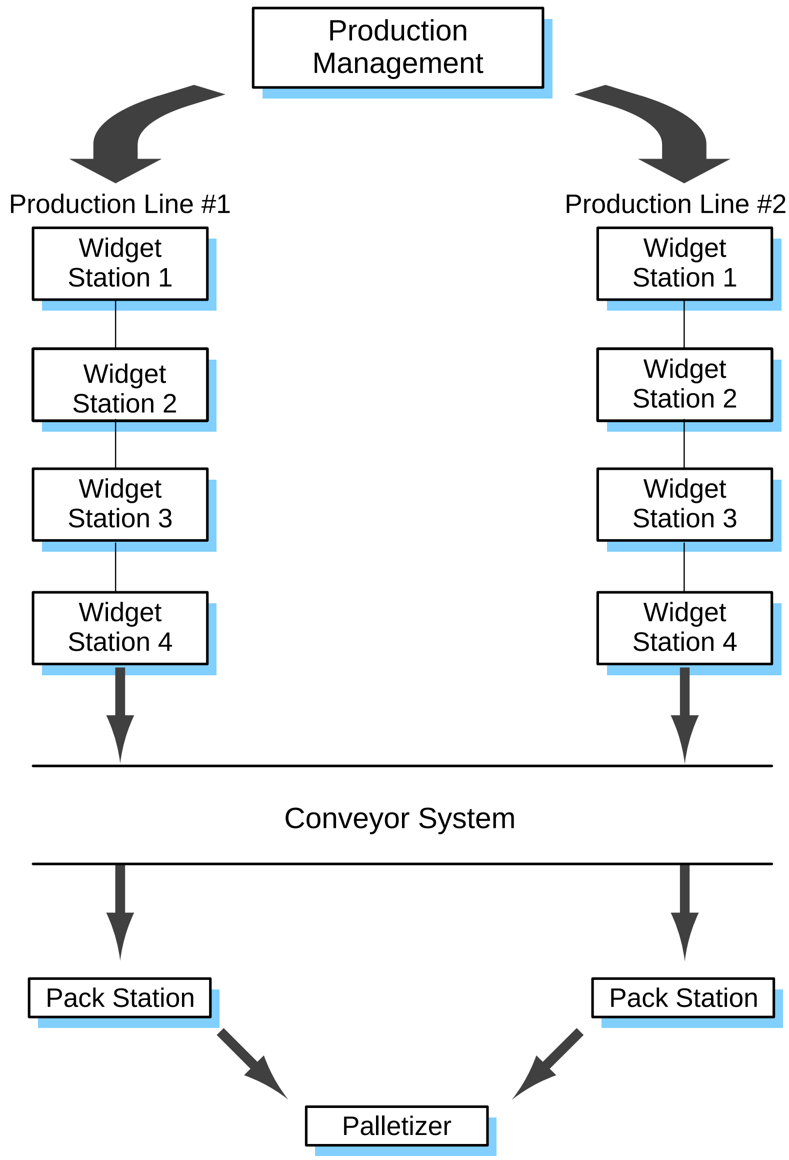
In This Chapter. . . .

Choosing the Configuration

Analyze your Application

There are almost as many ways to build your network as there are grains of sand at the beach. The first step in determining the configuration is to analyze your application. As with most things, there's probably a logical grouping that will point you in the right direction. If you have several systems you'll need to decide which station(s) will be the master(s) and which one(s) will be slaves.

Consider the following example. (It's the dreaded widget factory.)



Identify Your System Requirements

Determine how you plan to use the information that is transferred. This may help you design your network. For example, are you trying to share data between several machines, or are you trying to obtain data to be used in production reports or spreadsheets? Once you've taken a look at your application, you should be able to identify the configuration needs. For example, our simple widget factory needs to accomplish the following things.

Production Management: The production management system needs to issue build orders and obtain data that can be used in production reports. The management system also needs to monitor the production lines to determine which line should receive the build orders. For example, one line may be behind schedule while the other is completely free.

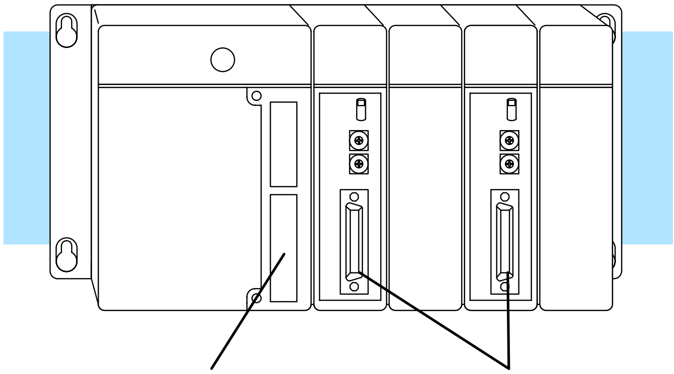
Production Lines 1 & 2: Each production line has a series of operations that must be followed to build a finished product. Once they are completed they can be sent to either pack station for packaging. In the real world, we all know that equipment sometimes doesn't work quite right, so each station should be monitored for operational status.

Pack Stations: The finished products are sent to the pack stations, which in turn send the cases to a palletizer for final shipping. Since there's only one palletizer and two stations packing different product models, the pack stations should coordinate their deliveries.

Palletizer: The palletizer stacks the widgets into (hopefully) nice, neat packages of widgets that can be sold by the millions.

Examine Data Sharing Requirements

Even though the three basic network configurations can be combined in an application, each network remains independent. The master station from one network cannot request data directly from slave stations on another network. This does not mean you cannot obtain data from these networks, you can. It just requires more than one interface for that slave station. For example, a DL440 PLC station with two DCM (Data Communication Module) interfaces actually can connect to three different networks. Check the manual for your particular slave station interface to determine the maximum number allowed in a single base.



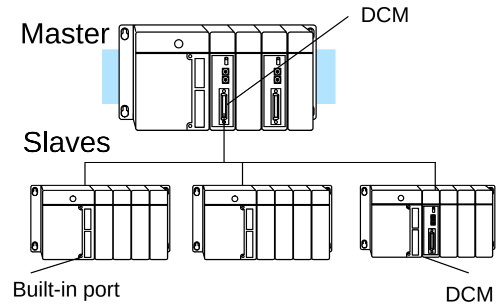
Sketch the Network Diagram

Since all applications must use one (or more) of the three basic network configurations (PLC as Master, Host as Master and Peer as Master), we can easily draw the complete diagram for our example factory. To refresh your memory, here are the possible configurations.

Possible Configurations

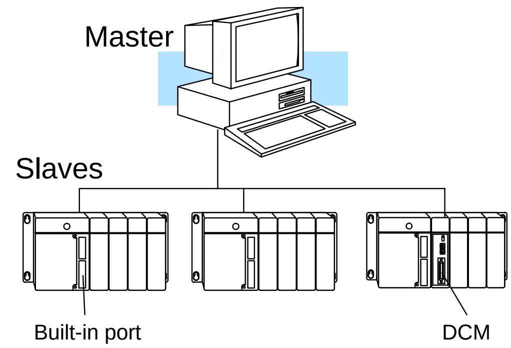
1 – PLC as Master Network

- Master can initiate read or write with any slave station
- PLC with DCM as master
- Full range of slave stations available



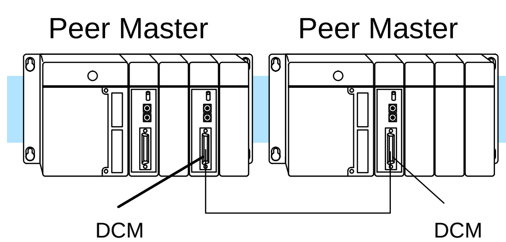
2 – Host as Master Network

- Host computer (or other intelligent device) as master
- Requires ***Direct*NET** program in the host
- Full range of slave stations available

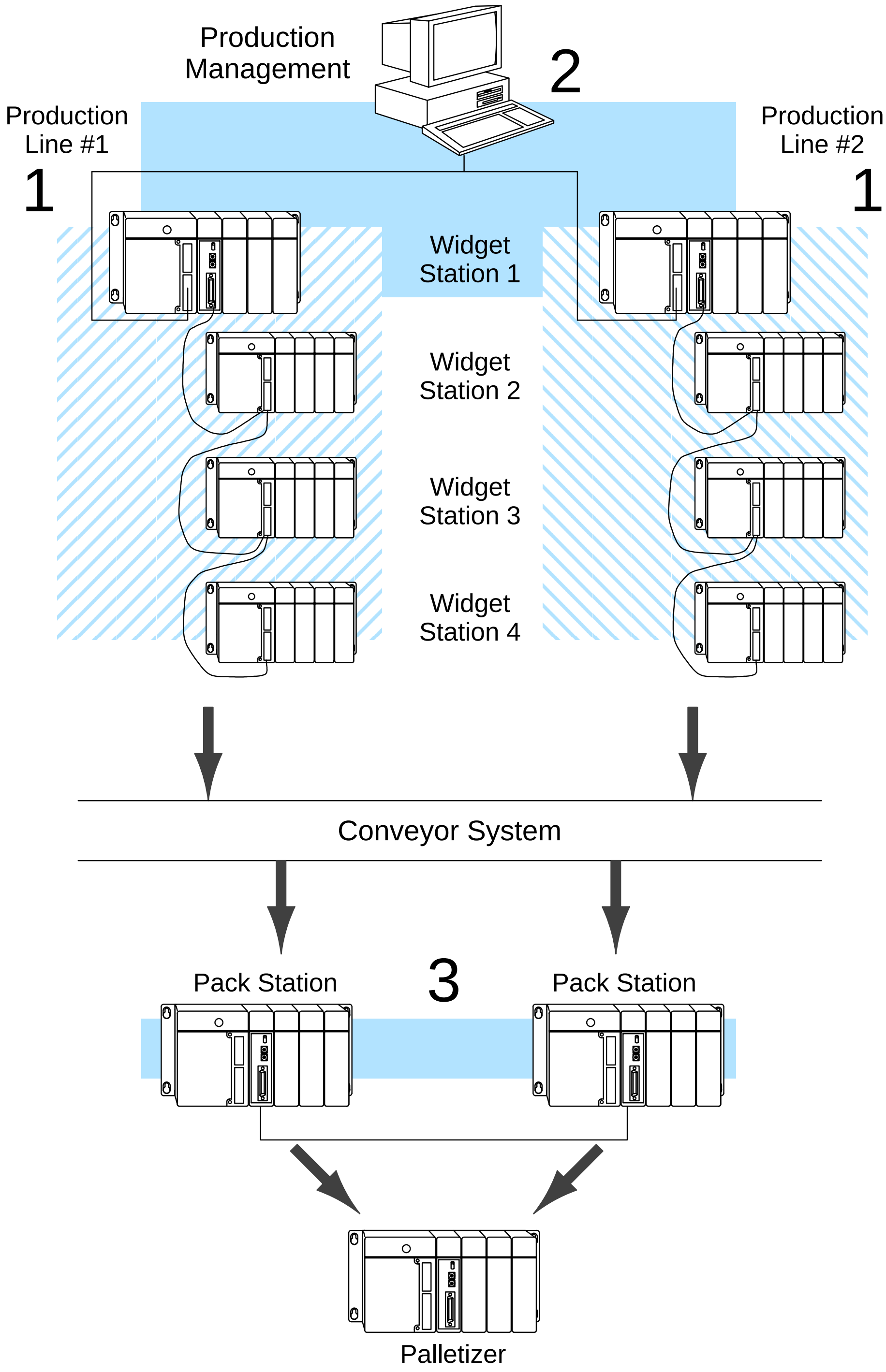


3 – Peer as Master Network

- Two stations only
- Only DL405 DCMs
- Either station can initiate a request



There are many different ways to configure the example. Notice, in the example on the next page, the areas where the different configurations were used. This example uses four small networks to solve the problem. The advantages of this approach are that you can optimize each portion without affecting other parts of the factory.



Now you're ready to build the network cables.

Once you've drawn your network configuration, you can build the network cables.

Cable Selection and Installation Guidelines

Things to Consider

There are several considerations that help determine the type of cable needed for your application.

1. Will the DCM be physically connected in a point-to-point configuration or multi-drop configuration?
2. What electrical specification is best for your application? RS232C or RS422?
3. What is the cable schematic?
4. What are the relevant cable specifications?
5. What installation guidelines are necessary?
6. Do you just need a quick test cable?

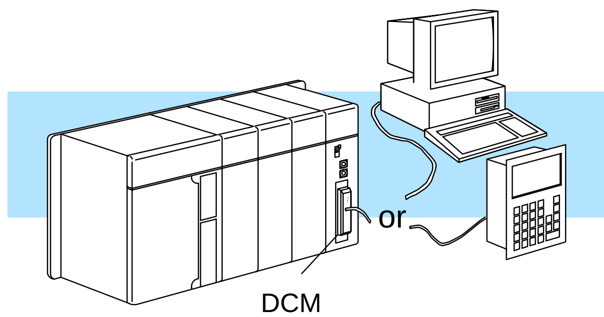
The next few pages discuss these considerations in detail. If you already know the type of cable needed, the cable schematics are included on pages 3–17 through 3–29.

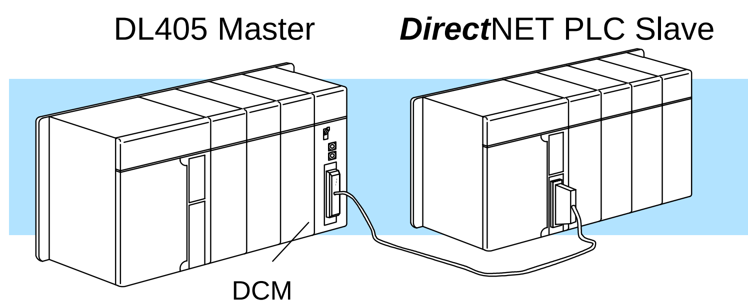
Consideration 1: Physical Configuration

***Direct*NET** can be used in either a point-to-point or multi-drop configuration. A point-to-point connection only has two stations, a master and a slave. You should use this configuration when you want to connect a ***Direct*NET** master station to a single ***Direct*NET** slave station. For example, use the point-to-point configuration to connect a personal computer, an operator interface, or an intelligent device to a single station.

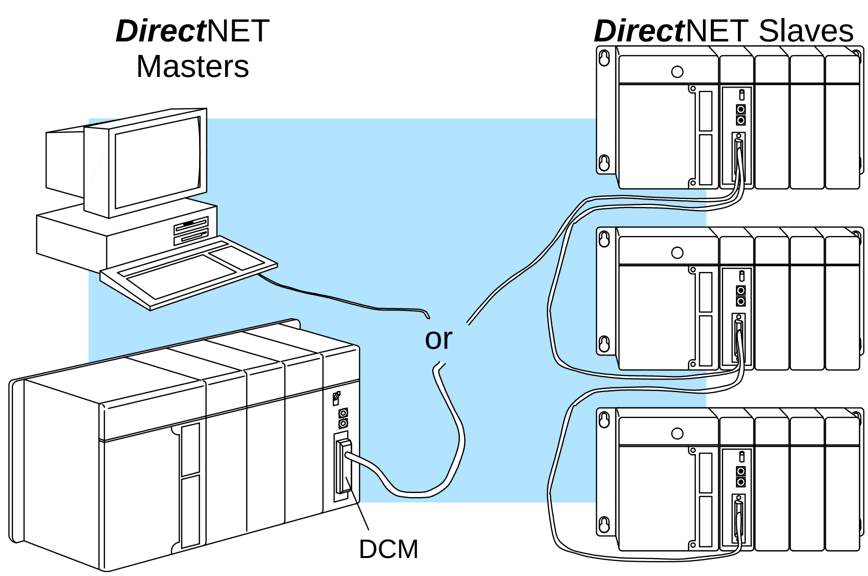
Use the multi-drop configuration to connect one master to two or more slaves.

Point to Point





Multi-drop



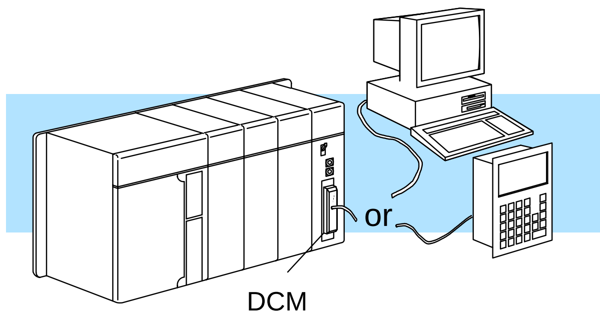
Consideration 2: Electrical Specification RS232C or RS422

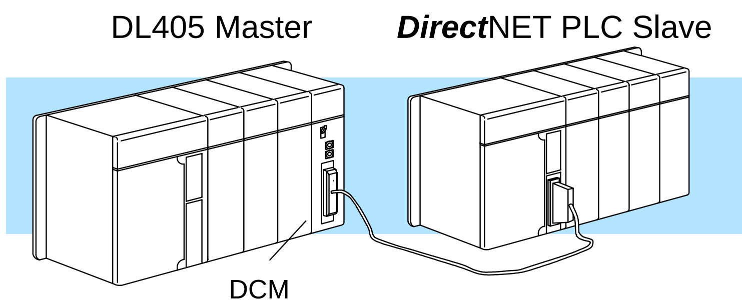
***Direct*NET supports both RS232C or RS422 communication. Your application and configuration choice will help determine which electrical specification is best for you. If you are using multi-drop, you must use RS422. If you are using point-to-point, you may have a choice between RS232C and RS422.

You can use RS232C if the cable length is less than 50 feet and if the cable will not be subjected to induced electrical noise that is commonly found near welders, large motors, or other devices that create large magnetic fields.

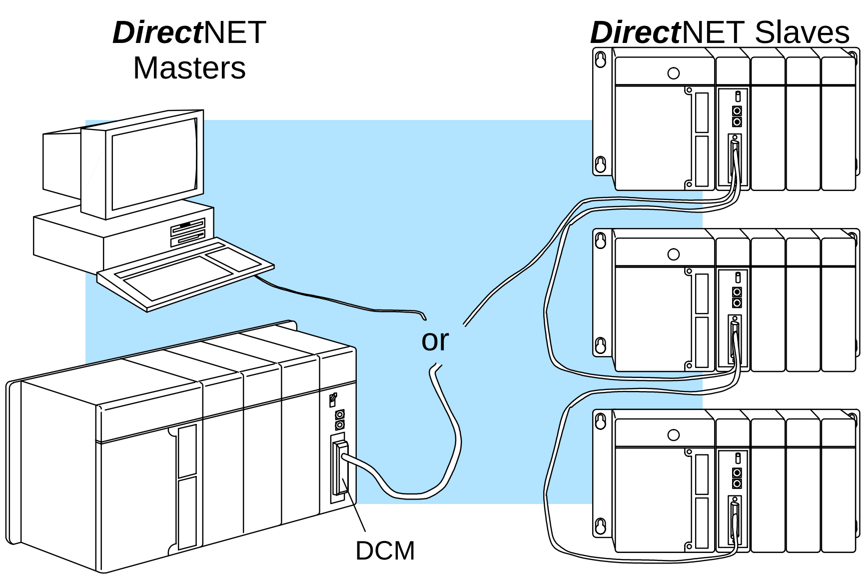
You should use RS422 for all other applications. RS422 allows longer cable distances (up to 3300 feet) and provides higher noise immunity.

RS232C or RS422 for Point-to-Point





RS422 for Multi-drop



Consideration 3: Cable Schematics

There are two sections in this chapter that provide various types of cable schematics that are appropriate for most applications. Point-to-Point cable diagrams can be found on pages 3–17 through 3–21. Multi-drop cable diagrams can be found on pages 3–22 through 3–29. You may have to combine some of these examples to design a cable that meets your exact application requirements.

Consideration 4: Cable Specifications

Although many types of cables may work for your application, we recommend you use a cable that is constructed to offer a high degree of noise immunity. A cable constructed equivalent to Belden 9855 should be sufficient. The following specifications should be used as a guideline.

Structure	Shielded, twisted-pair (RS232C only uses two wires and a ground)
Conductor size	24 AWG or larger
Insulation	Polyethylene
Shield	Copper braid or aluminum foil
Impedance	100Ω @ 1MHz
Capacitance	60pf / meter or less

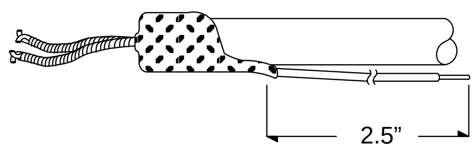
Consideration 5: Installation Guidelines

Your company may have guidelines for cable installation. If so, you should check those before you begin the installation. Here are some general things to consider.

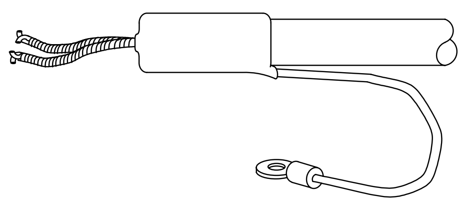
- Don’t run cable next to larger motors, high current switches, or transformers. This may cause noise problems.
- Route the cable through an approved cable housing to minimize the risk of accidental cable damage. Check local and national codes to choose the correct method for your application.
- Consider redundant cabling if the application data is critical. This allows you to quickly reconnect all stations while the primary cable is being repaired.

Cable Shield Grounding — No matter what type of configuration is required, it is important to ground the cable shield to minimize the possibility of noise. The preferred method is to connect one end (preferably the receiver end) of the cable shield to the connector housing. If noise problems are still present and you have a good earth ground for the cabinet, you should connect one end of the shield to the cabinet earth ground. *Don’t* ground both ends of the shield because this will create induced noise on the cable.

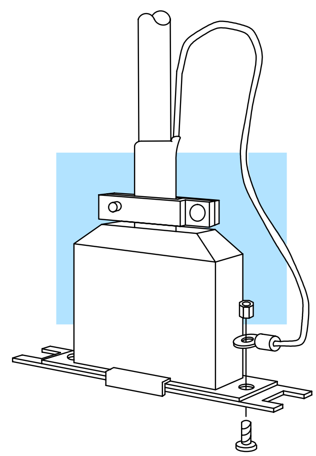
Step 1: Strip back about 2.5” of the shield.



Step 2: Crimp a ring connector onto the shield.



Step 3: Secure the shield to the connector shell.



Consideration 6: A Quick Test Cable

PLC***Direct***™ offers a Universal Cable Kit (part number FA–CABKIT). This cable kit allows you to connect various types of ***Direct***LOGIC™ products with an RS232C cable in a matter of minutes. The kit consists of a cable (phone cable with male plugs already attached) and several specially wired connectors and phone cables.

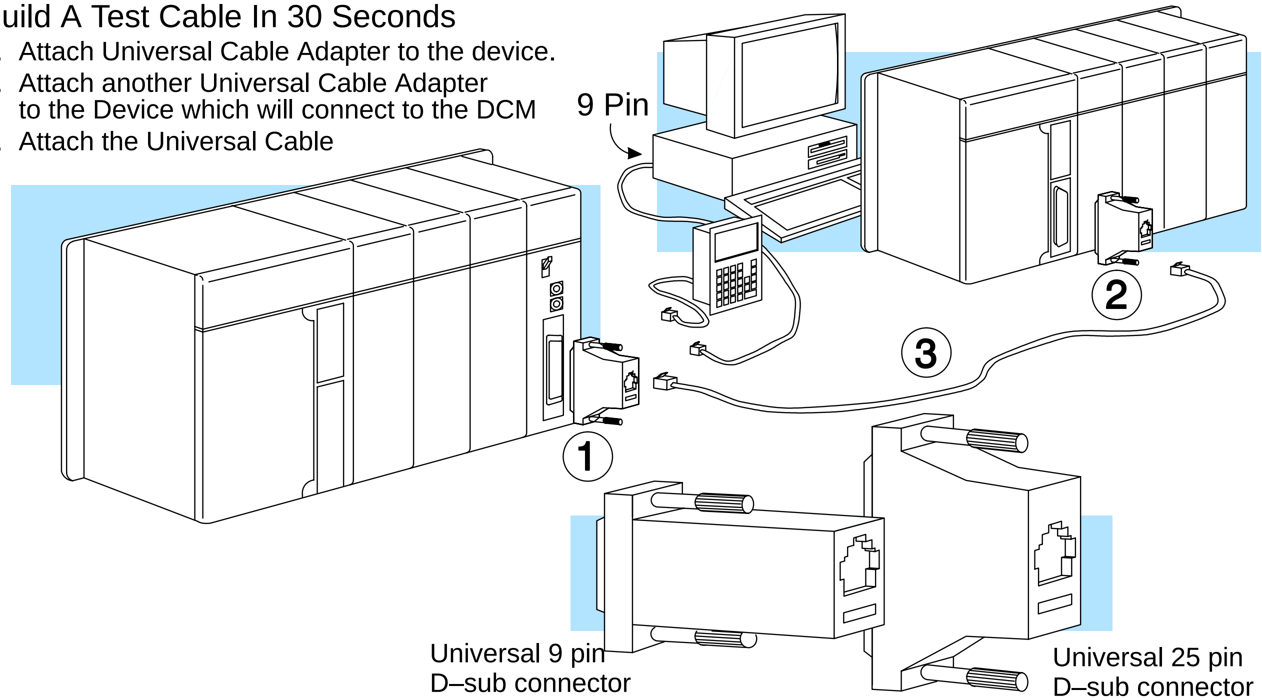
The special connectors are a D-sub style with built-in female phone jacks. The kit includes a wide variety of the special connectors so you can use one kit to easily connect products from the different ***Direct***LOGIC™ family of products. To use the kit just follow these steps.

1. Plug the appropriate D-sub connector onto the device (CPU, DCU, or DCM).
2. Plug the appropriate D-sub connector onto the other device you are connecting.
3. Connect the cable to the two D-sub connectors.

WARNING: This cable is suitable for quick testing situations and should not be used in actual applications. This cable is not shielded and is highly susceptible to electrical noise. Electrical noise can cause unpredictable operation that may result in a risk of personal injury or damage to equipment. Use the cable specifications described earlier in this manual to select a cable suitable for actual applications.

Build A Test Cable In 30 Seconds

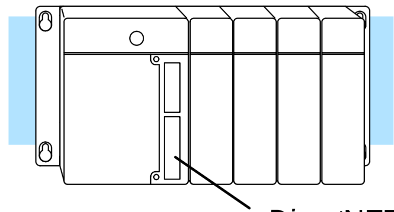
1. Attach Universal Cable Adapter to the device.
2. Attach another Universal Cable Adapter to the Device which will connect to the DCM
3. Attach the Universal Cable



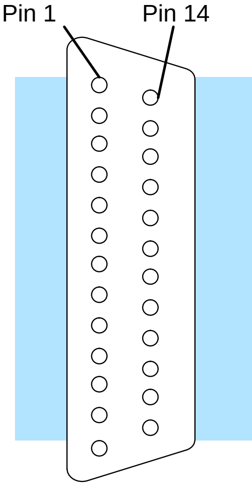
NOTE: The adapters take care of any pin swapping that may be required between the devices. You do not have to make *any* wiring adjustments.

DL430 and DL440 Port Pinouts

The DL430 and DL440 CPUs have built-in ***Direct***NET ports. This port is only a slave port. The following diagram shows the port pinout connections.



Port Pinouts

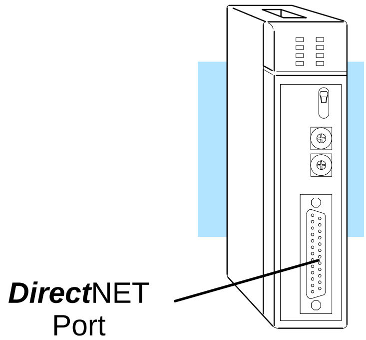


Pin	Signal Definition	Pin	Signal Definition
1	Not connected	14	RS422 data out +
2	RS232C data out	15	Not connected
3	RS232C data in	16	RS422 data out –
4	RS232C RTS	17	Not connected
5	RS232C CTS	18	RS422 RTS –
6	Not connected	19	RS422 RTS +
7	Signal ground	20	Not connected
8	Not connected	21	Not connected
9	RS422 data in +	22	Not connected
10	RS422 data in –	23	RS422 CTS –
11	RS422 CTS +	24	Not connected
12	Not connected	25	Not connected
13	Not connected		

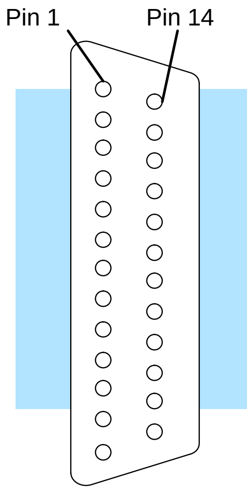
Pin labeling conforms to the IBM DTE and DCE standards.

DL405 DCM Port Pinouts

The DL405 DCM can be used as a master or slave station interface. The following diagram shows the port pinout connections.



Port Pinouts



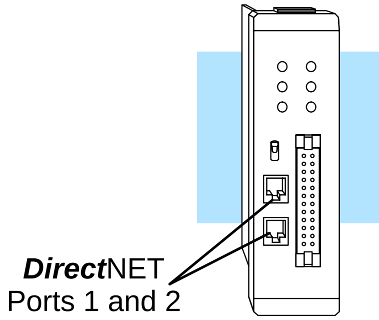
Pin	Signal Definition	Pin	Signal Definition
1	Not connected	14	RS422 data out +
2	RS232C data out	15	RS422 data out –
3	RS232C data in	16	RS422 data in –
4	RS232C RTS	17	RS422 data in +
5	RS232C CTS	18	Not connected
6	Internal Circuit 5V	19	Not connected
7	Internal Circuit 0V	20	Not connected
8	RS422 RTS +	21	Not connected
9	RS422 RTS –	22	RS422 data out +
10	RS422 RTS +	23	RS422 data out –
11	RS422 RTS –	24	RS422data in –
12	RS422 CTS +	25	RS422 data in +
13	RS422 CTS –		

Pin labeling conforms to the IBM DTE and DCE standards.

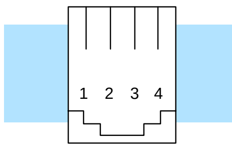
DL340 CPU Port Pinouts

The DL340 CPU has two built-in ***Direct***NET slave ports under the hinged cover.

The ports are limited to RS232C communication, so you have to use an RS232C/RS422 converter for multi-drop connections. This port is an RJ11 (handset connector) phone jack. The ***Direct***LOGICE Universal Cable Kit (FA–CABKIT) can be used if you do not want to create a phone jack connection.



Port 2 is the Master port.



Phone Jack Connector

Ports 1 and 2 Pinouts

Pin	Signal Definition
1	RS232C data in
2	RS232C data out
3	Request to Send
4	Ground

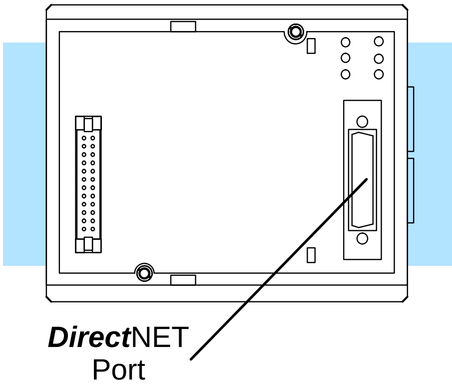
Pin labeling conforms to the IBM DTE and DCE standards.

DL305 DCU Port Pinouts

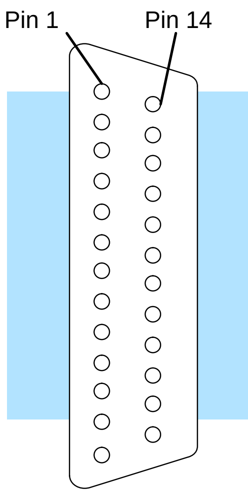
There are two versions of the DL305 DCU.

- D3–232–DCU (RS232C version)
- D3–422–DCU (RS422 version)

The DL305 DCU can only be used as a slave station interface. If you only have one master and one slave, you can use either version. If you have more than one slave you should probably use the the RS422 version. You can use the RS232C version, but you'll need an RS232C/ RS422 converter (FA–UNICON) for each station.



Port Pinouts



D3–422–DCU (RS422)

Pin	Signal Definition	Pin	Signal Definition
1	Not connected	14	RS422 data out +
2	Not connected	15	RS422 data out –
3	Not connected	16	RS422 data in –
4	Not connected	17	RS422 data in +
5	Not connected	18	Not connected
6	Not connected	19	Not connected
7	Logic ground 0V	20	Not connected
8	Not connected	21	Not connected
9	Not connected	22	RS422 data out +
10	RS422 RTS +	23	RS422 data out –
11	RS422 RTS –	24	RS422data in –
12	RS422 CTS +	25	RS422 data in +
13	RS422 CTS –		

D3–232–DCU (RS232C)

Pin	Signal Definition	Pin	Signal Definition
1	Not connected	14	Not connected
2	RS232C TXD	15	Not connected
3	RS232C RXD	16	Not connected
4	RS232C RTS	17	Not connected
5	RS232C CTS	18	Not connected
6	Not connected	19	Not connected
7	Logic ground 0V	20	Not connected
8	Not connected	21	Not connected
9	Not connected	22	Not connected
10	Not connected	23	Not connected
11	Not connected	24	Not connected
12	Not connected	25	Not connected
13	Not connected		

Pin labeling conforms to the IBM DTE and DCE standards.

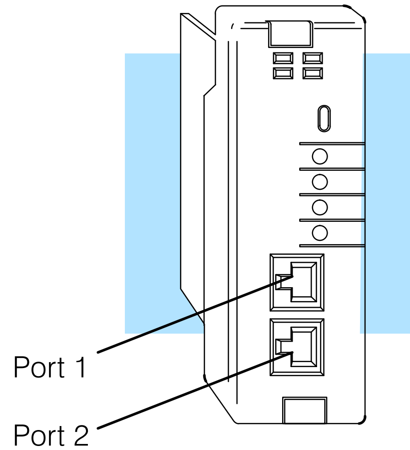
DL240 CPU Port Pinouts

The DL240 CPU has two built-in ***Direct***NET ports.

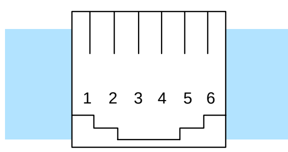
Port 1: can be used for point-to-point communications with either RS232C or RS422. However Port 1 cannot be used in a multi-drop configuration even with an RS232C / RS422 converter. This is because the RTS signal pin is used for 5V.

Port 2: can be used for either point-to-point or multi-drop connections. An RS232C/RS422 converter is required for multi-drop connections.

These ports are RJ11 phone jacks. The ***Direct***LOGIC™ Universal Cable Kit (FA–CABKIT) can be used if you do not want to create a phone jack connection.



DirectNET
Ports



Phone Jack
Connector

Port 1 Pinouts

Pin	Signal Definition
1	0 V
2	5 V
3	RS232C Data in
4	RS232C Data out
5	5 V
6	0 V

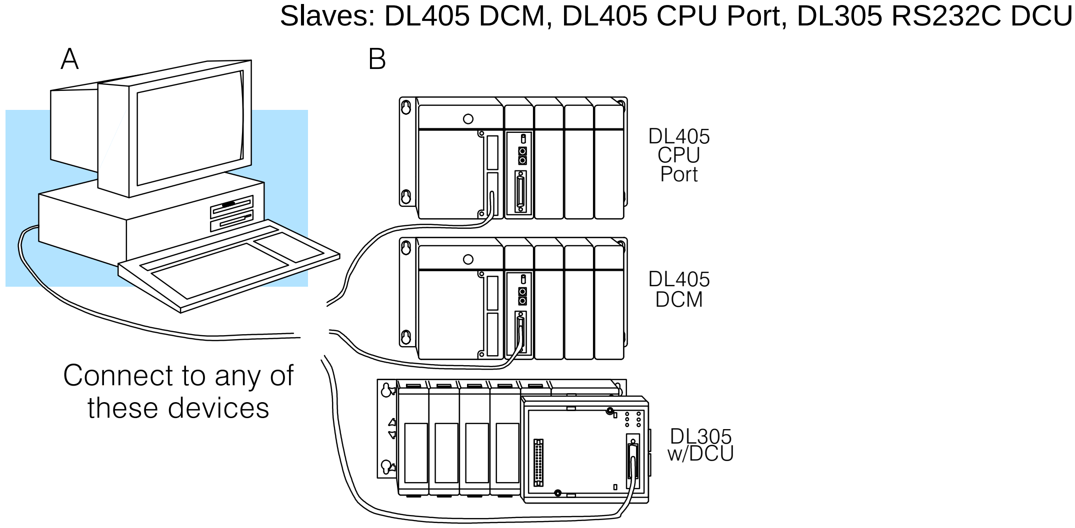
Port 2 Pinouts

Pin	Signal Definition
1	0 V
2	5 V
3	RS232C Data in
4	RS232C Data out
5	Request to Send out
6	0 V

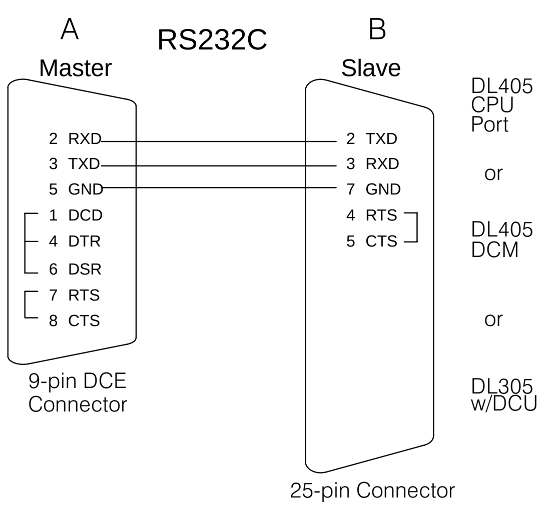
Pin labeling conforms to the IBM DTE and DCE standards.

Point-to-point Cables

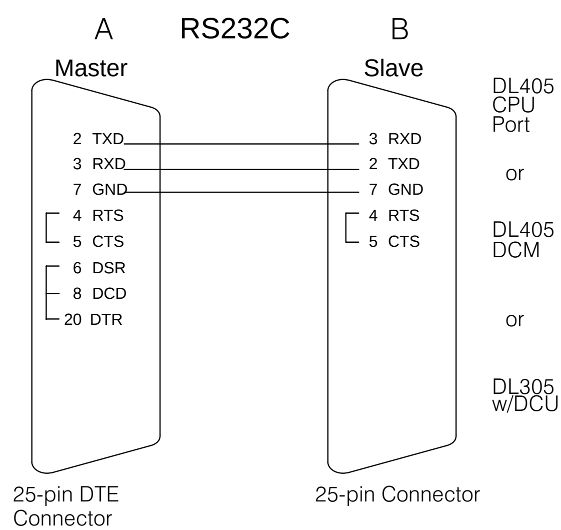
Host as Master



9-pin Connector



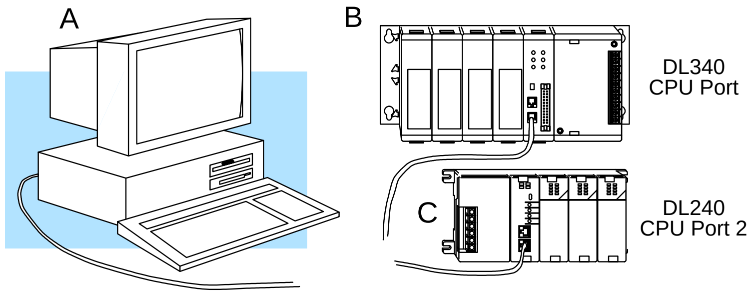
25-pin Connector



Pin labeling conforms to the IBM DTE and DCE standards.

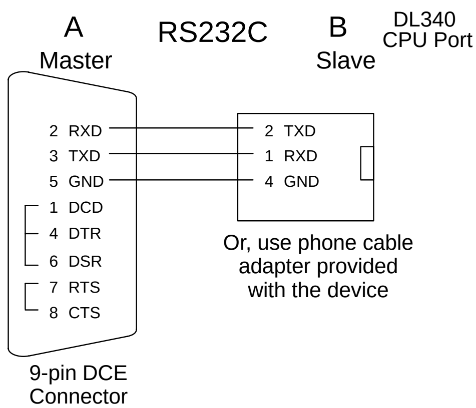
Host as Master (continued)

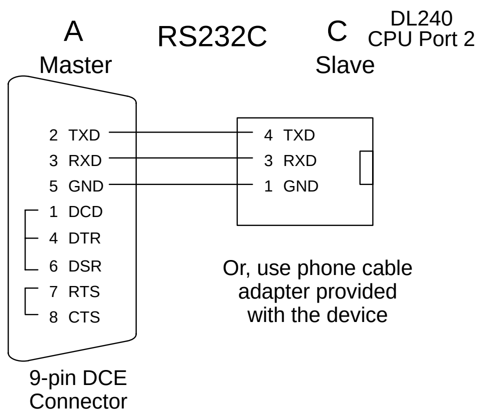
Slaves: DL340 CPU Port, DCM, DL240 CPU Port



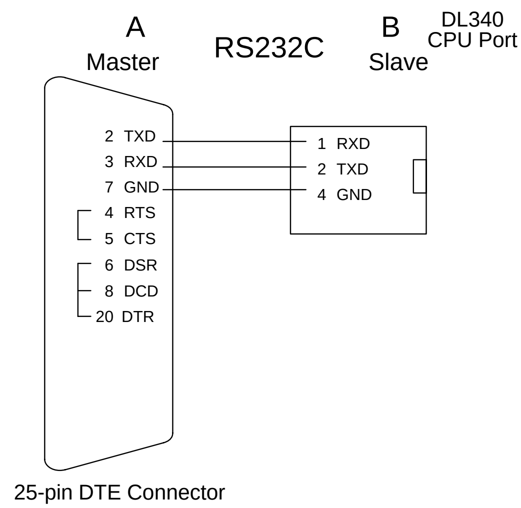
Connect to any of these devices

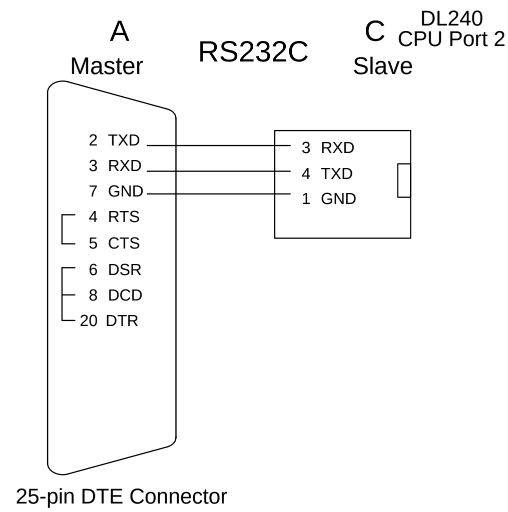
9-pin Connector





25-pin Connector

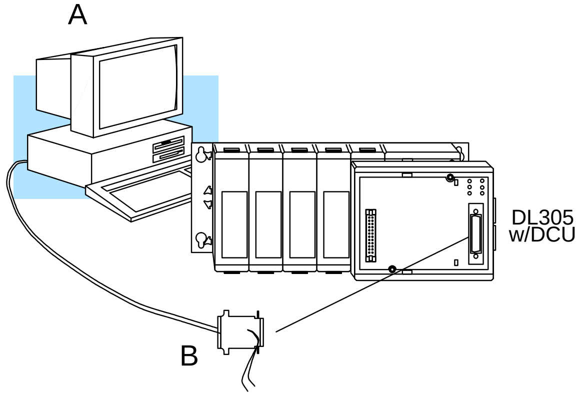




Pin labeling conforms to the IBM DTE and DCE standards.

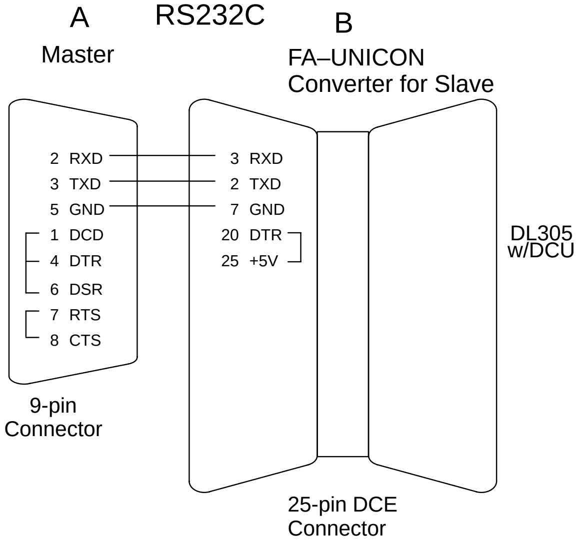
Host as Master (continued)

Slaves: DL305 RS422 DCU

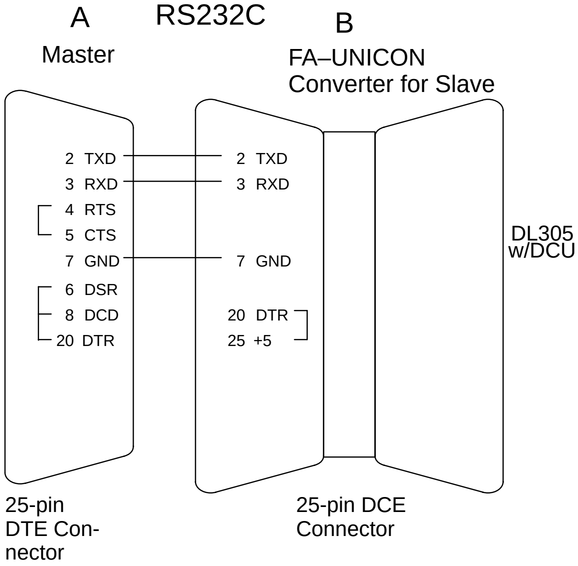


Male connector plugs onto DCU

9-pin Connector



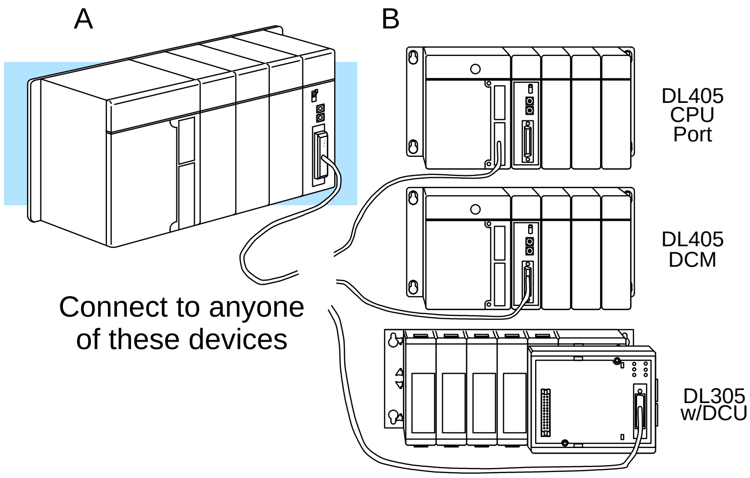
25-pin Connector

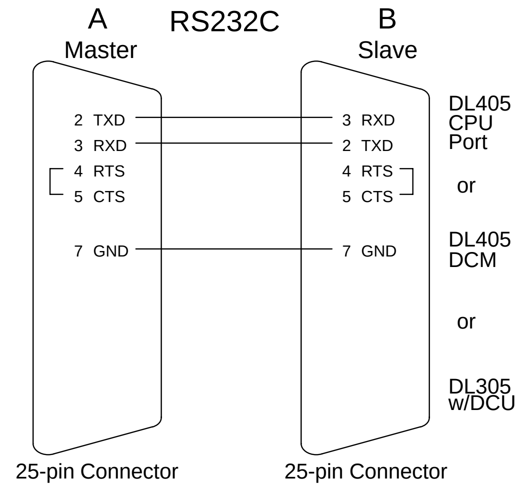


Pin labeling conforms to the IBM DTE and DCE standards.

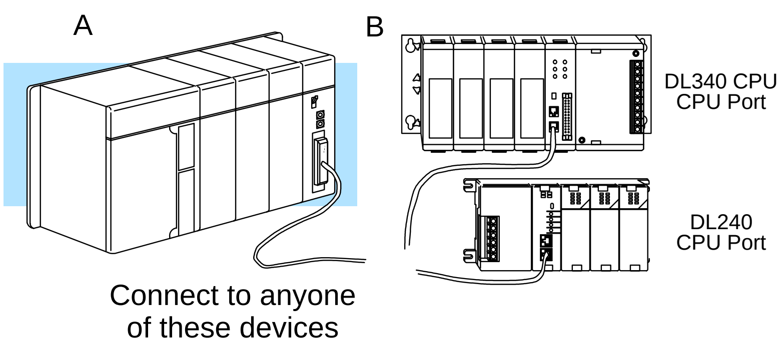
DL405 DCM as Master

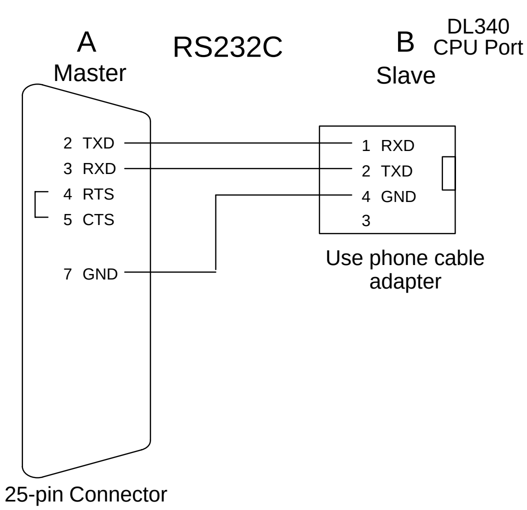
Slaves: DL405 DCM, DL405 CPU Port, DL305 RS232C DCU

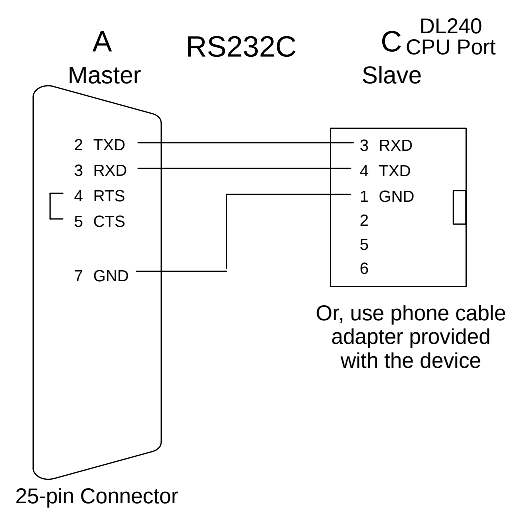




Slaves: DL340 CPU Port, DL240 CPU Port, DL205 DCM



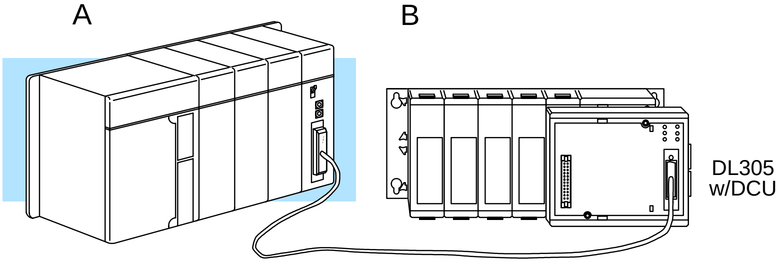


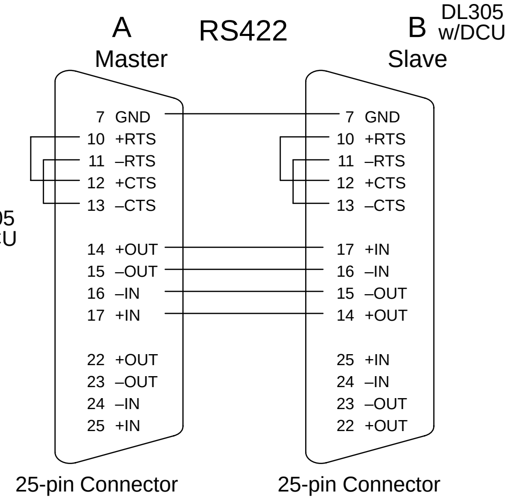


Pin labeling conforms to the IBM DTE and DCE standards.

DL405 DCM as Master (continued)

Slaves: DL305 RS422 DCU





Pin labeling conforms to the IBM DTE and DCE standards.

Multi-Drop Cables

More Considerations

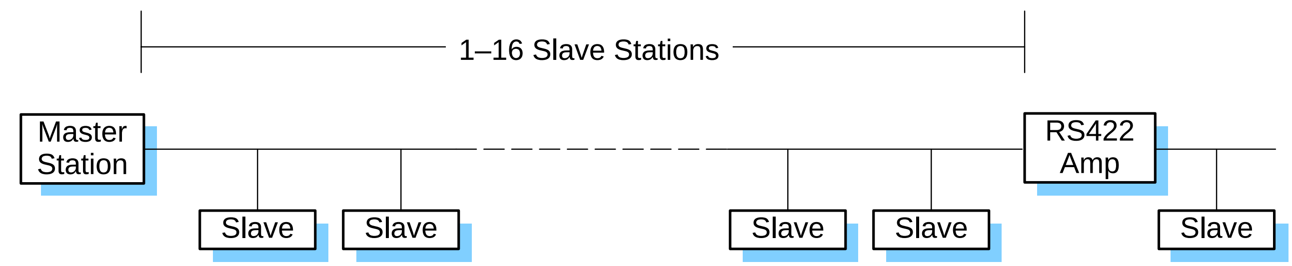
With RS422 you have great flexibility in how you want to design your network cabling. You also have to use:

- Network Amplifiers – if you have more than 16 slave stations
- Cable Termination Resistors – to help to reduce data errors.

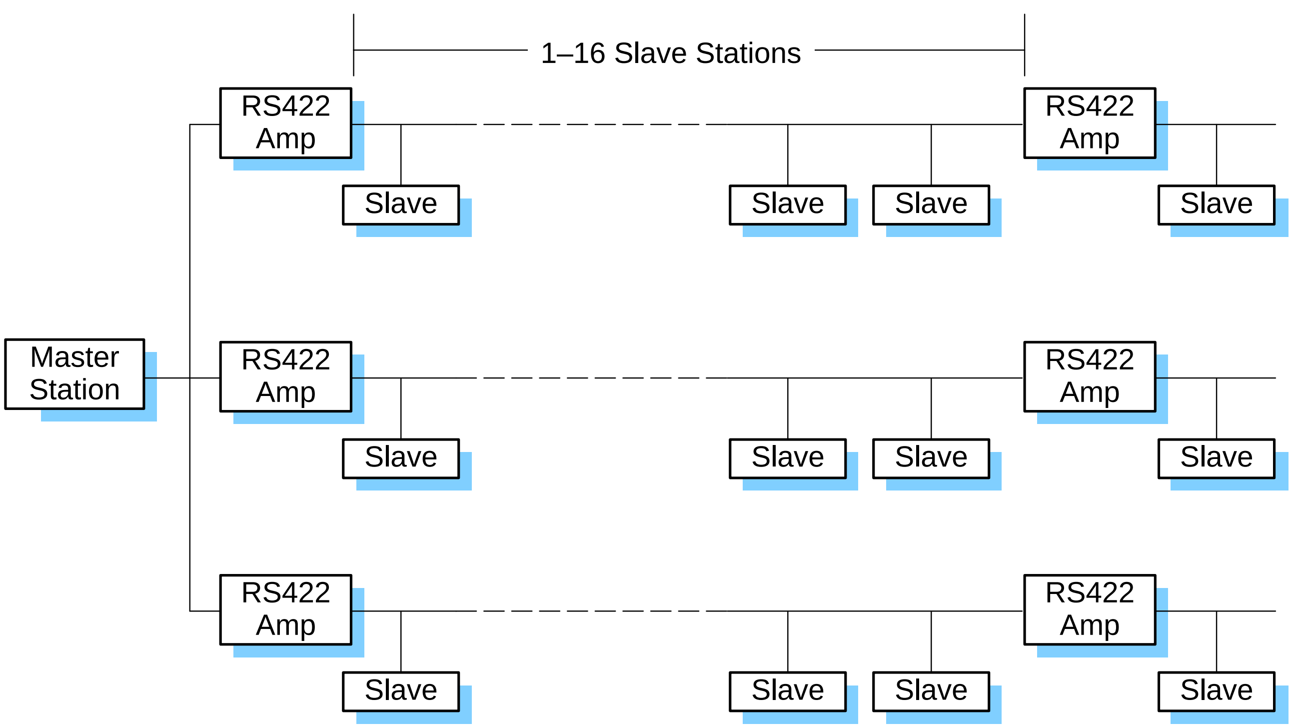
Network Amplifiers

If you have more than 16 slave stations, you should use an RS422 amplifier to maintain the signal levels. The best amplifiers are regenerative, that is, they recover the signal and try to reduce any noise signals that are present. Some amplifiers are not regenerative and amplify the noise as well as the signal. (You can get amplifiers from several sources. The Black Box catalog is one of many good places to start.) The following diagram shows some instances where an amplifier is necessary.

Serial Slave Connection



Parallel Slave Connection



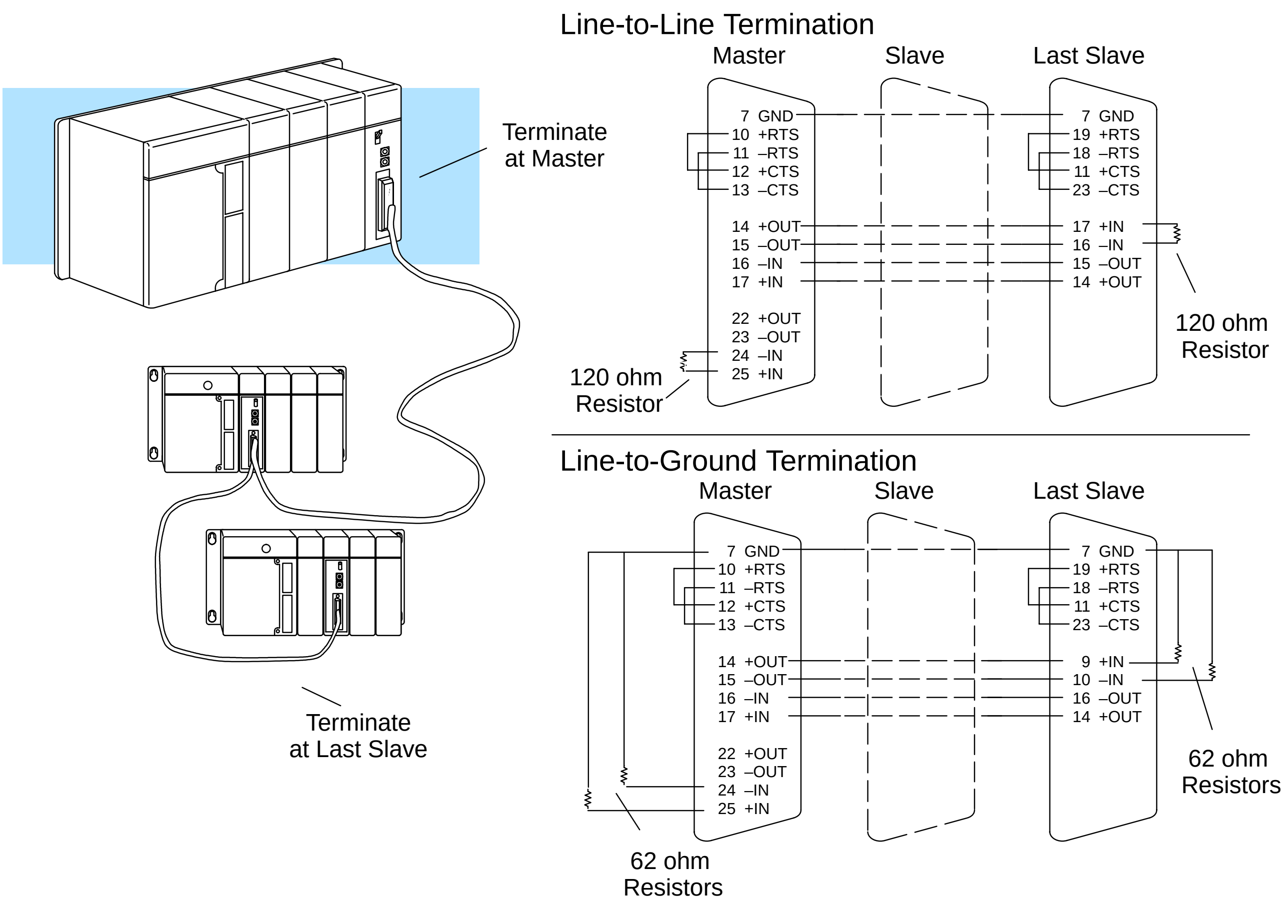
Cable Termination Resistors

It is important you add termination resistors at each end of the RS422 line. This helps reduce data errors during data transmission. You should select resistors that match the cable impedance. For example, a typical 22 AWG solid conductor cable with 4.5 twists per foot has a typical impedance of about 120Ω.

There are two ways to actually connect the resistors.

- Line-to-Line — this method balances the receive data lines (IN+ and IN–) and requires one resistor at each end of the line. (The cable diagrams we've provided show this method, but you can use either.)
- Line-to-Ground — this method also balances the receive data lines, but common mode noise rejection is improved significantly. This method requires two resistors at each end of the line. Also, since there are two resistors, the sum total of both resistors should match the cable impedance.

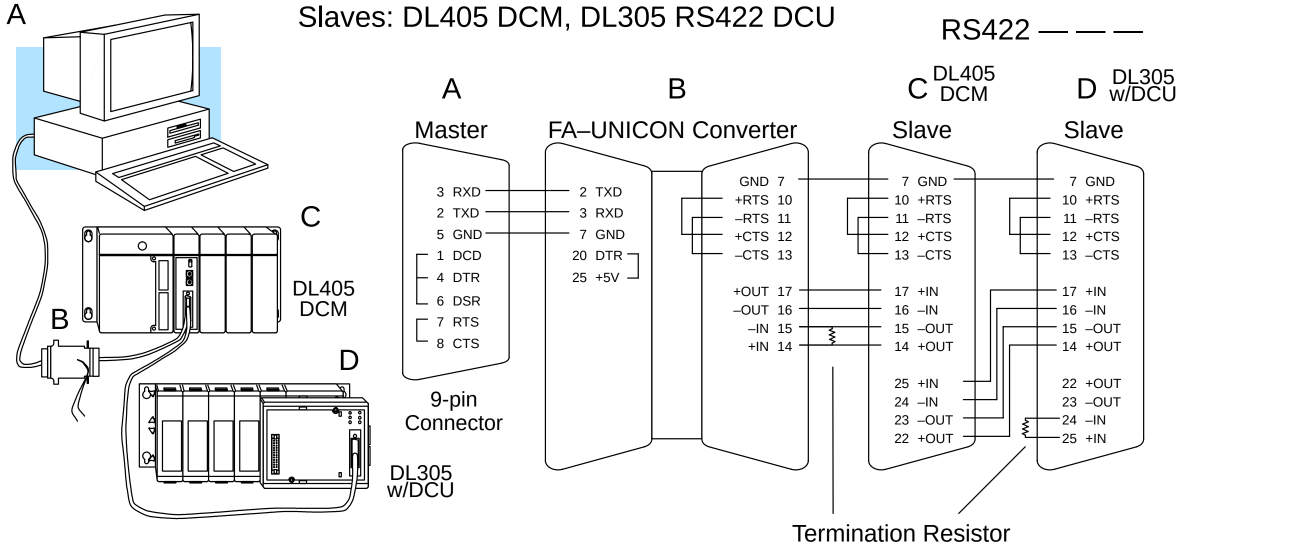
The following diagram illustrates the two options.



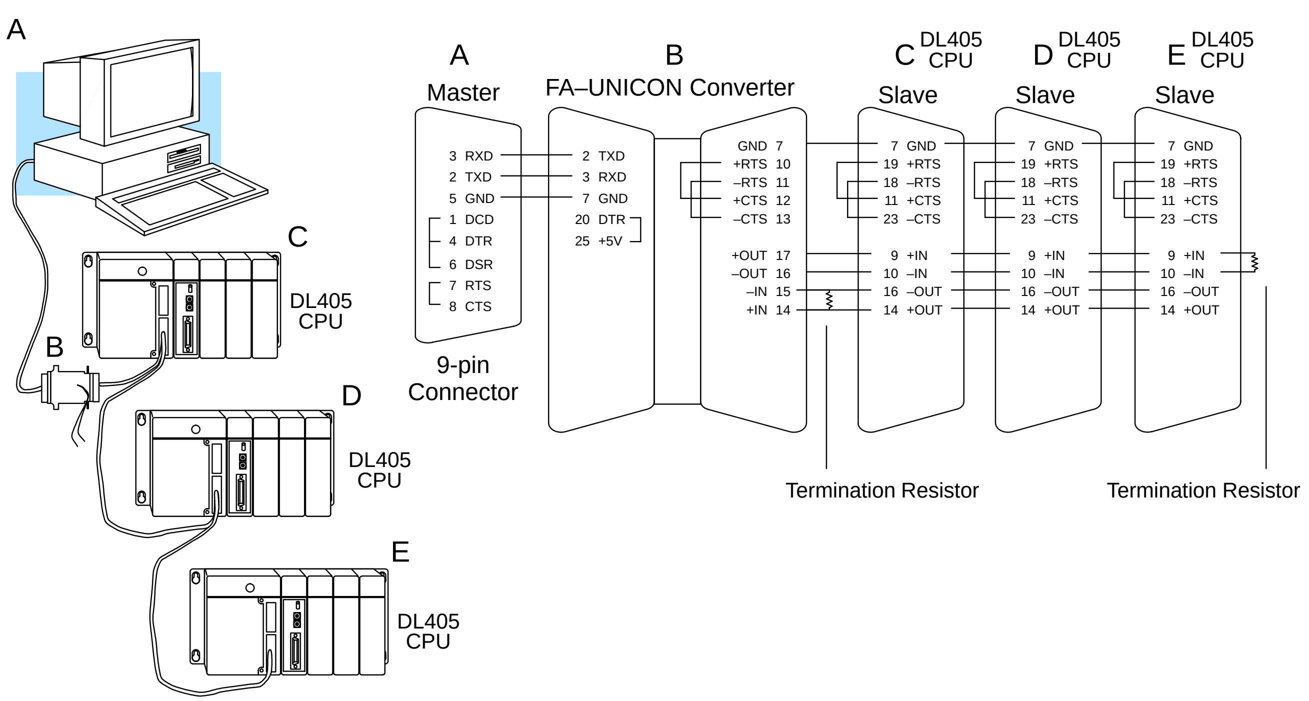
Pin labeling conforms to the IBM DTE and DCE standards.

Host as Master

Slaves: DL405 DCM, DL305 RS422 DCU



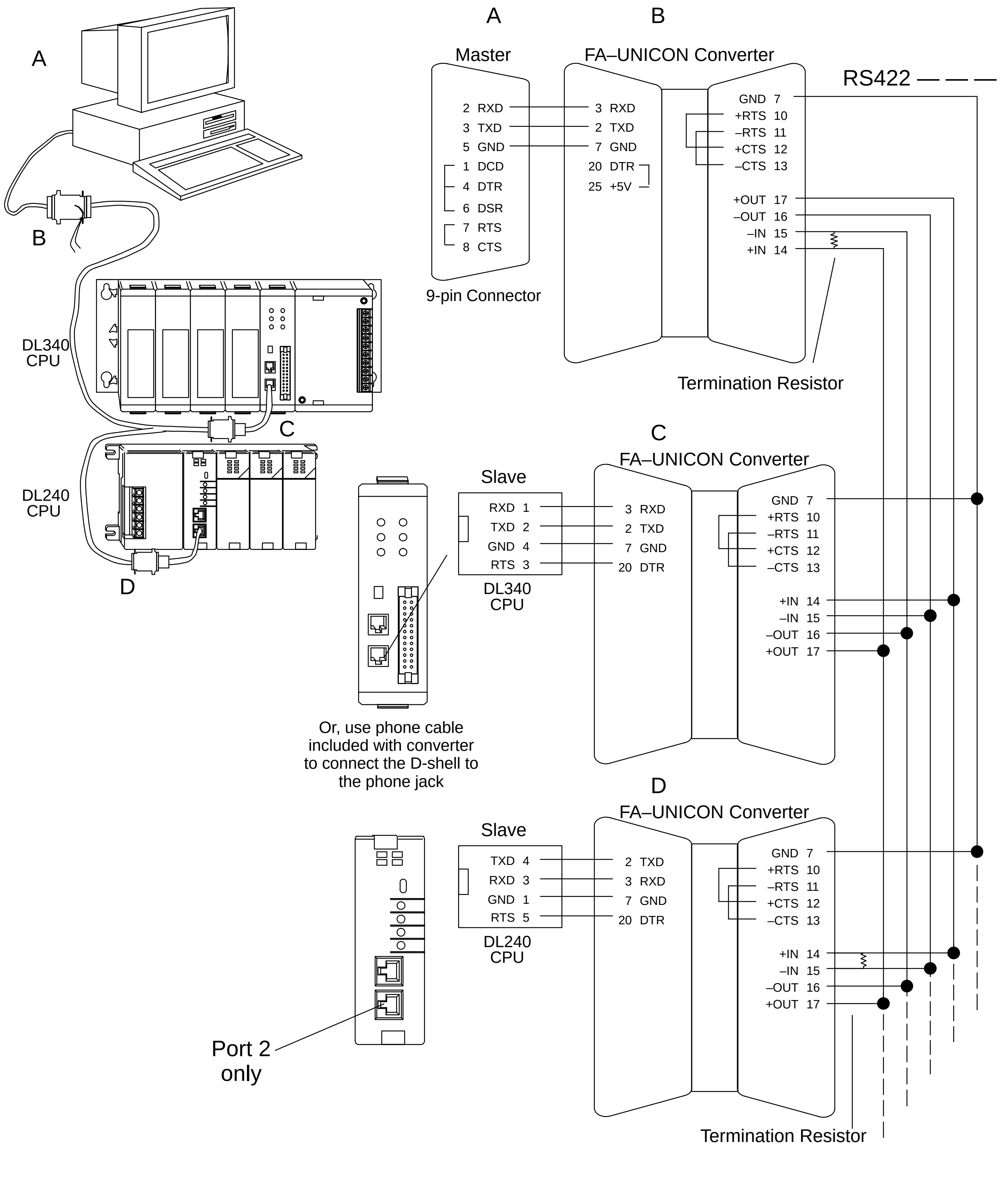
Slaves: DL405 CPU Ports



Pin labeling conforms to the IBM DTE and DCE standards.

Host as Master (continued)

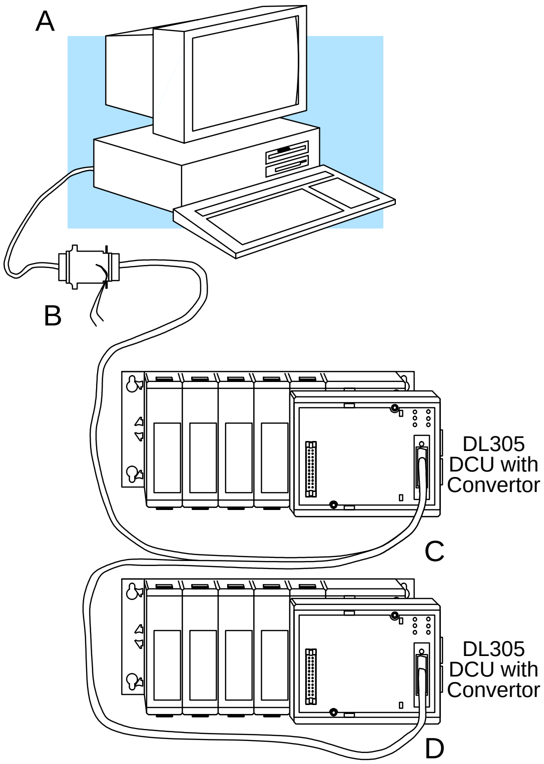
Slaves: DL340 CPU Port, DL240 CPU Port, DL205 DCM

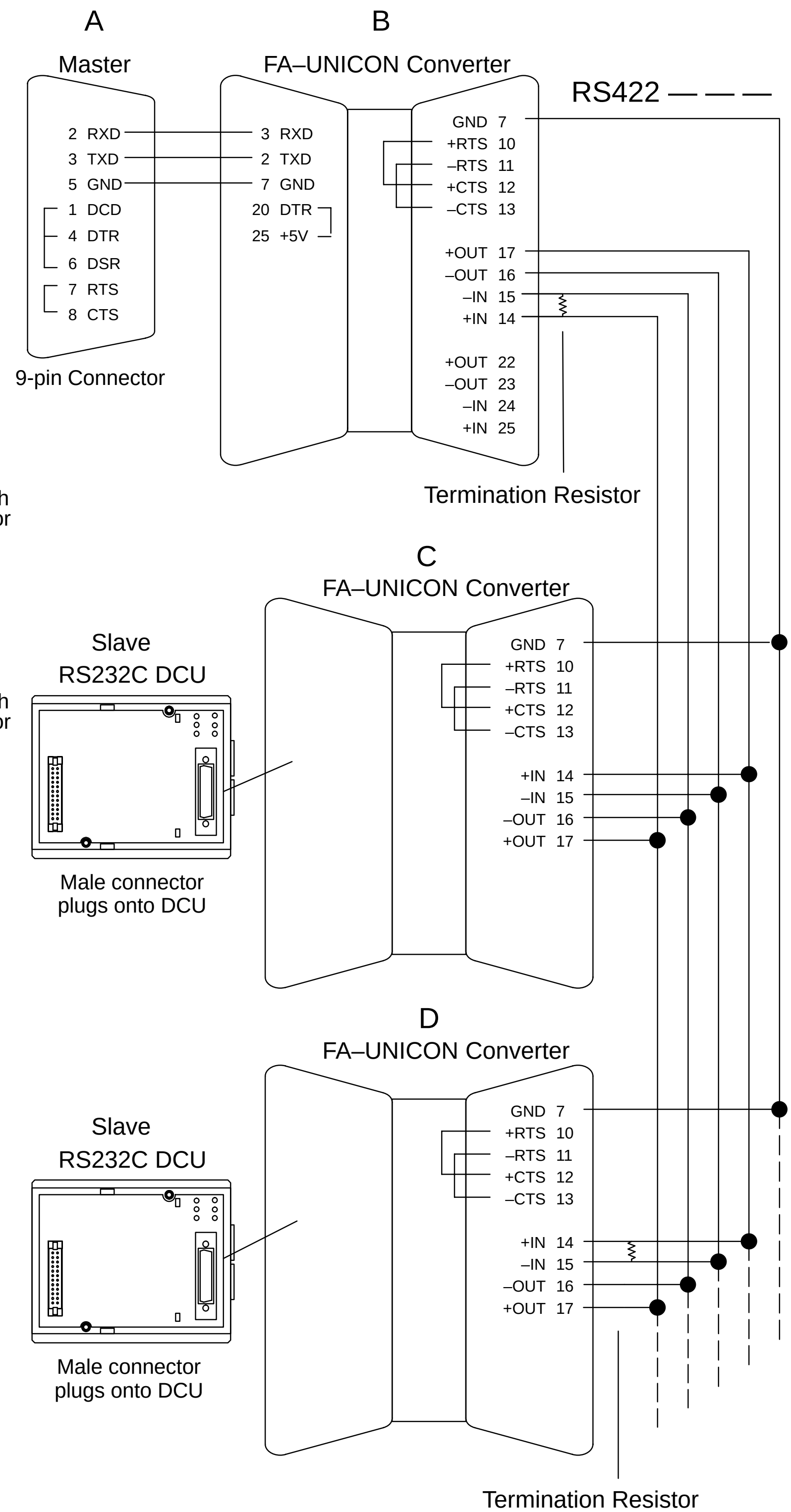


Pin labeling conforms to the IBM DTE and DCE standards.

Host as Master (continued)

Slaves: DL305 RS232C DCU

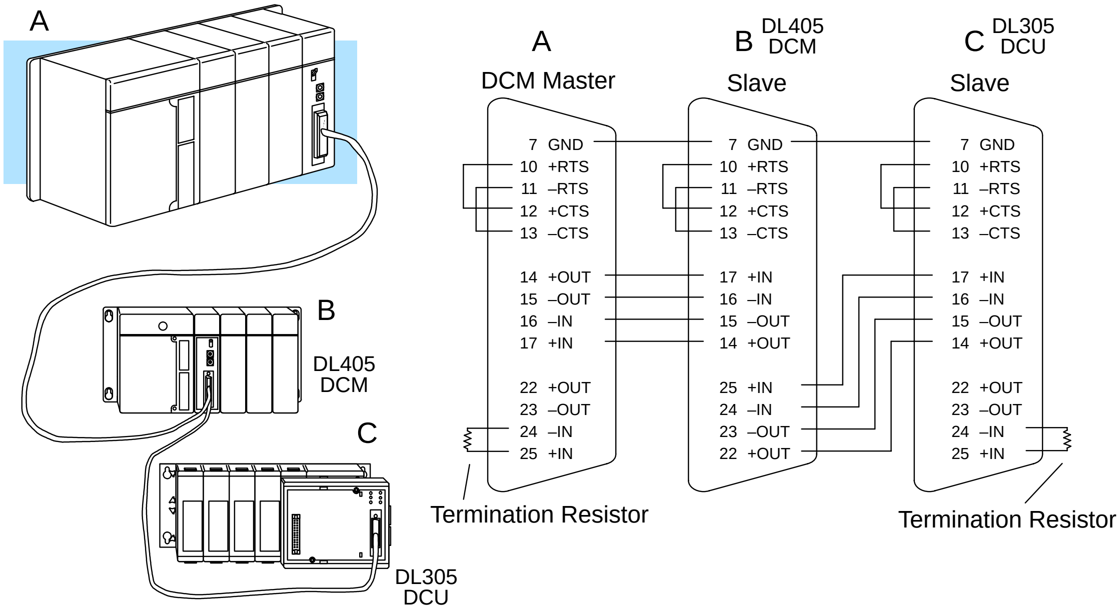




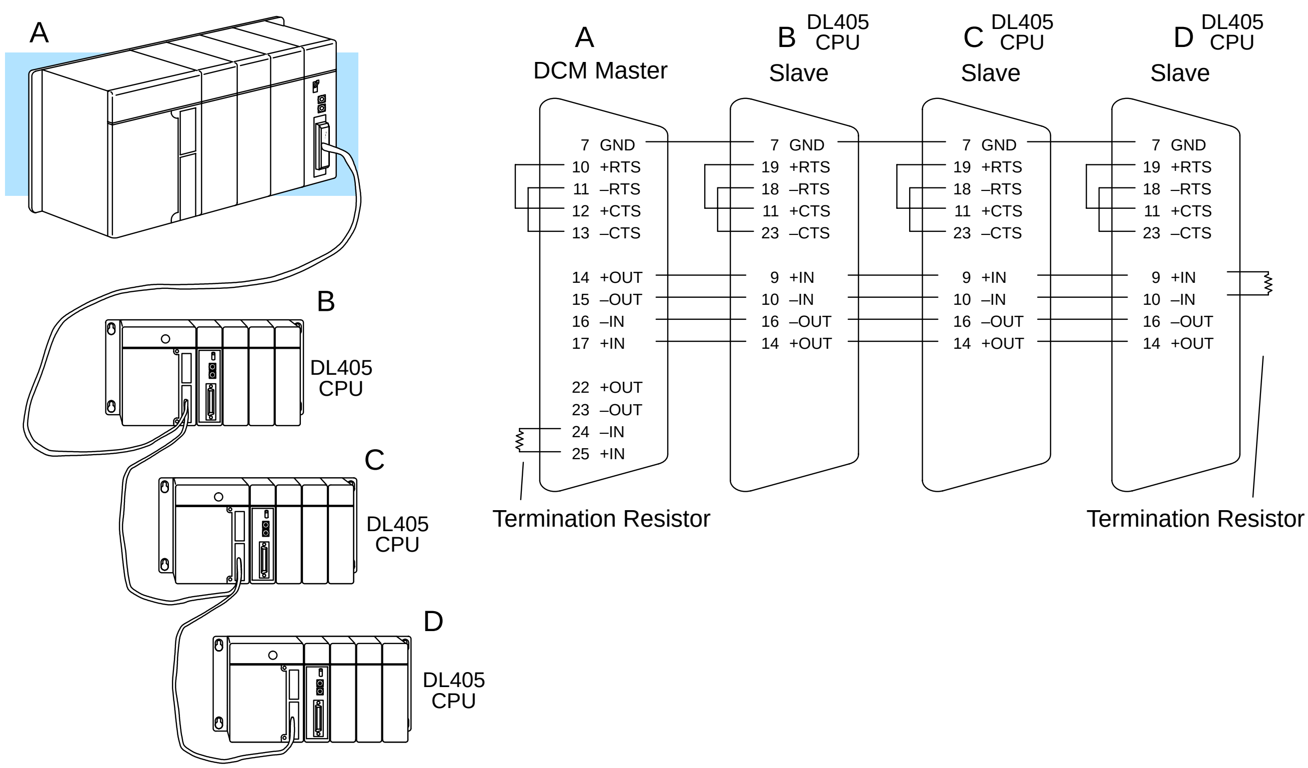
Pin labeling conforms to the IBM DTE and DCE standards.

DL405 DCM Master

Slaves: DL405 DCM, DL305 RS422 DCU



Slaves: DL405 CPU Ports

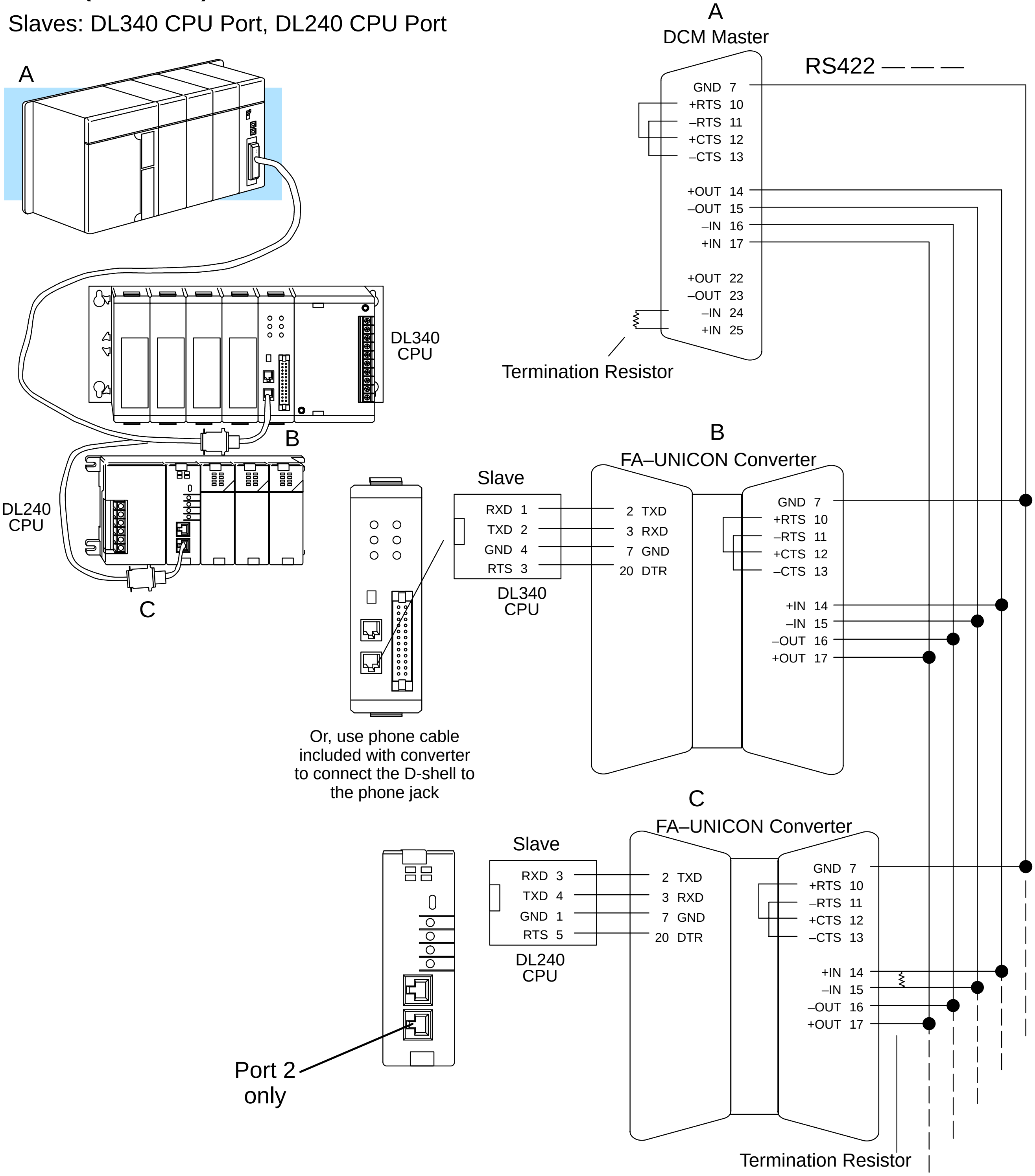


Pin labeling conforms to the IBM DTE and DCE standards.

Network Design and Setup

DL405 DCM as Master (continued)

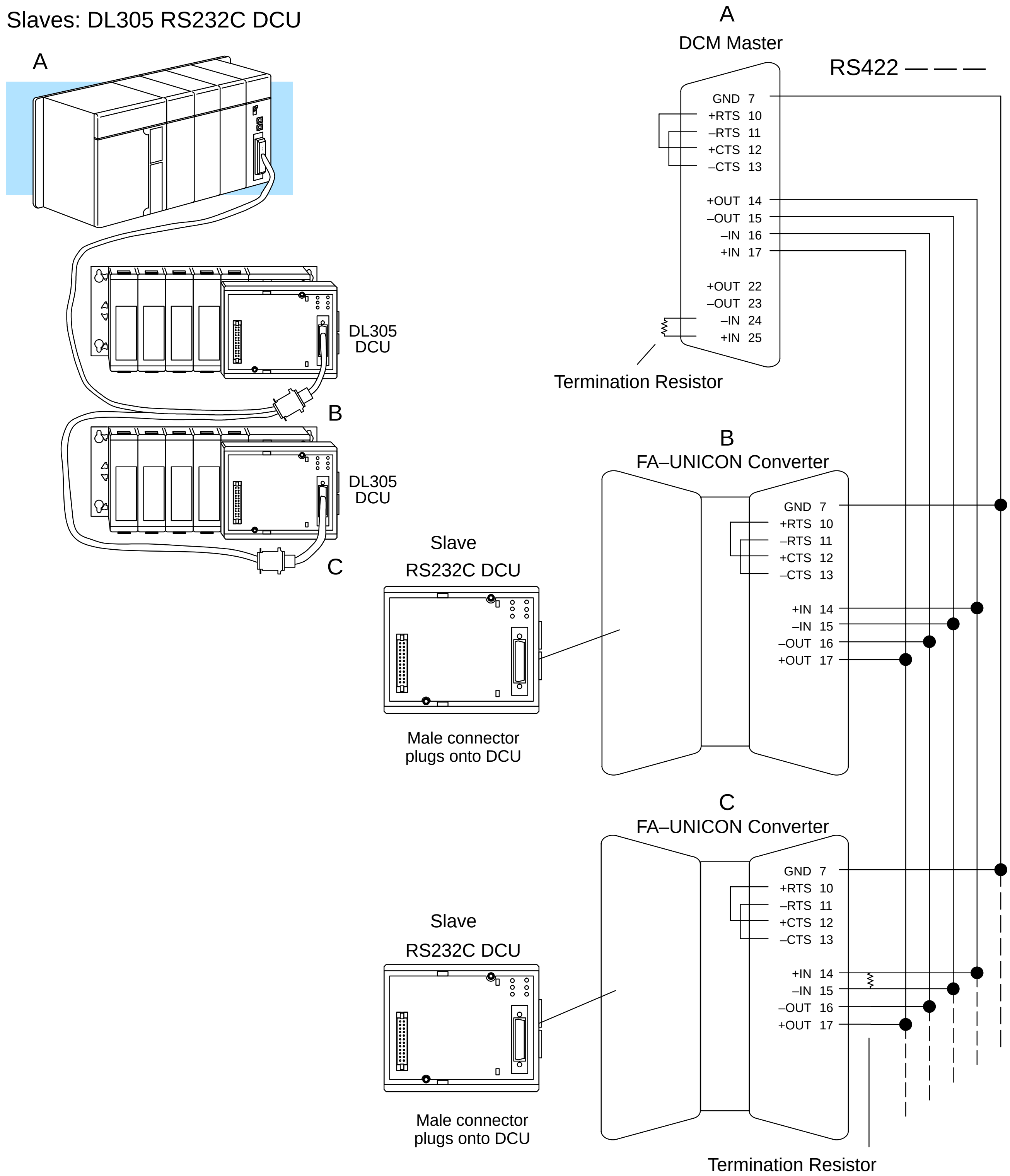
Slaves: DL340 CPU Port, DL240 CPU Port



Pin labeling conforms to the IBM DTE and DCE standards.

DL405 DCM Master (continued)

Slaves: DL305 RS232C DCU



Pin labeling conforms to the IBM DTE and DCE standards.

Network Design Checklist

You can avoid problems

Incorrect network cabling causes many problems. It's important to make sure your network design and cable selection is complete before proceeding to the next chapters. Have you ......

1. Drawn the network so you can easily identify the types of networks involved?
2. Used the cable diagrams to understand the network cabling requirements?

Once you have designed the network, you're ready to set the communications switches for each of the network stations. Chapter 4 shows you how to set the switches for the various types of stations.

4

Communication Settings

In This Chapter. . . .

— Selecting the communication settings
— Setting the station switches
— Communications Settings Checklist
— What part of the manual should you use next?

Selecting the communication settings.

What do I have to select?

There are a few things you need to select before you can set the communication switches for the network stations.

- Baud rate
- Parity
- Response delay
- Hexadecimal or ASCII data representation
- Station address

Baud Rate

All devices on the network must use the same communication settings, but not all of the masters and slaves offer the same range of settings. This means your choice of equipment will help you determine the baud rate. For example, the DCM (Data Communication Module) interfaces can communicate at speeds of 38.4K baud, but the built-in PLC ports are limited to 19.2K baud (the DL340 can go up to 38.4K baud). Also, some computer communication cards are limited in the range of available baud rates. Check the documentation that came with your computer or operator interface to determine the available communication parameters.

DirectNET can support eight baud rate selections ranging from 300bps to 38.4Kbps. Usually, you should use the highest baud rate possible unless noise problems appear. If noise problems appear, try reducing the baud rates.

Parity

You can choose between odd parity or no parity. Again, all the devices should be set the same. Odd parity uses eleven bits total (1 start bit, 8 data bits, 1 stop bit, and 1 parity bit.) Odd parity also provides slightly better error checking. Some devices require no parity, which uses only 10 bits (1 start bit, 8 data bits, and 1 stop bit.)

Response Delay

This sets how long the station will wait before it responds to a communication request. If you're using all ***Direct***LOGIC™ equipment, a response delay is not required and you should set the time to 0. This is primarily included for those of you who need to use modems. For example, the DCM may respond too quickly for some devices, such as telephone or radio modems. If you encounter this problem just choose a delay from 0 to 500 mS. Your modem manual should suggest the proper settings.

HEX or ASCII Data

If you want the fastest communication possible, use HEX mode. The difference is in the way the data is represented. The same data is twice as long in ASCII format, so it takes longer to transfer. If you have a device on the network that requires ASCII mode, then select ASCII mode for all the stations, otherwise, use HEX mode.

Station Address

You can have up to 90 slave stations connected to the master. Each station, including the master, must have a unique network address. There are a few restrictions on how you can use the available addresses.

Master stations: the master station should be address 0. This insures compatibility with all slaves.

Slave stations: the slave stations can use addresses1 – 90 and not address 0.

Peer stations: the peer stations can use addresses 1 – 90. There are only two stations in this configuration and either one can act as a master and initiate a request, however; each station is restricted from using address 0.

Setting the station parameters

For all Masters and Slaves

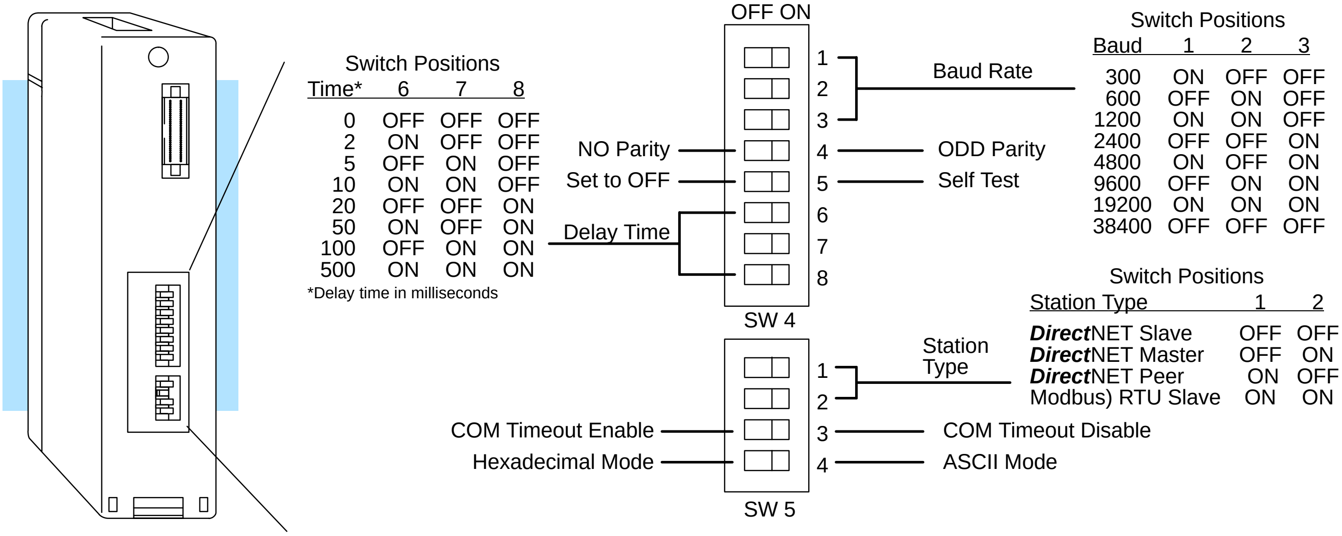
You have to set the switches for all the stations on the network, masters and slaves. Set the switches so all the stations have the same communication parameters. The next several pages provide the switch settings for the ***Direct***LOGIC™ network components. You can quickly scan the pages to find the settings for your devices. Read the descriptions carefully because most network problems are caused by incorrect switch settings.

Host as a Master

It seems there's a computer manufacturer on every corner these days, so it would be impossible for us to provide the communication settings for all of them. Just check the documentation that came with your computer to determine how to set the parameters.

DL405 DCM Communication Switches

There are two banks of switches located on the rear of the DCM that are used to set the communications parameters.



DCM Rear View

Switch Positions

Time*	6	7	8
0	OFF	OFF	OFF
2	ON	OFF	OFF
5	OFF	ON	OFF
10	ON	ON	OFF
20	OFF	OFF	ON
50	ON	OFF	ON
100	OFF	ON	ON
500	ON	ON	ON

*Delay time in milliseconds

Switch Positions

Baud	1	2	3
300	ON	OFF	OFF
600	OFF	ON	OFF
1200	ON	ON	OFF
2400	OFF	OFF	ON
4800	ON	OFF	ON
9600	OFF	ON	ON
19200	ON	ON	ON
38400	OFF	OFF	OFF

Switch Positions

Station Type	1	2
***Direct*NET Slave**	OFF	OFF
***Direct*NET Master**	OFF	ON
***Direct*NET Peer**	ON	OFF
Modbus) RTU Slave	ON	ON

Baud Rate: Positions 1 – 3 on SW4 are used to set the baud rate for the DCM. There are eight baud rate selections available from 300 baud to 38.4K baud.

Parity: Position 4 on SW4 selects between the two parity options, odd or none.

Self-Test: Position 5 on SW4 selects the factory self-test and should always be switched off. If the self-test is on, the module will not operate correctly.

Response Delay Time: Positions 6 – 8 on SW4 set the response delay time. This is primarily used when you're using modems and determines how long the DCM will wait before it responds to a communication request. If you're using all ***Direct***LOGIC™ equipment, a response delay is not required and you should set the time to 0. If you're using a modem, check the documentation to determine the proper delay time.

Station Type: Positions 1 and 2 on SW5 select the DCM protocol and the station type, master or slave. The DCM is very versatile and can actually support a second protocol Modbus®. Be sure you have selected the proper protocol for your network, either ***Direct***NET or Modbus®.

Host as Master: If you're using a host as master, set the DCM as a ***Direct***NET slave.

PLC as Master: If the DCM is in a master station, then set the DCM as a ***Direct***NET master. If the DCM is in a slave station, then set the DCM as a ***Direct***NET slave.

Peer as Master: If you're using a peer network, set both stations as ***Direct***NET peers.

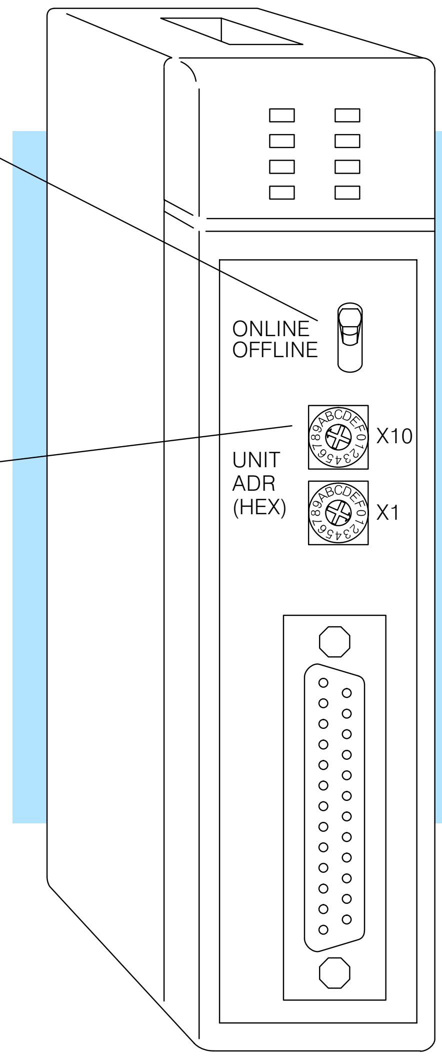
Communication Timeout: Position 3 on SW5 selects the communication timeout. Don't disable the timeout for normal use. Communication Timeout Disable is normally used only if you're developing your own ***Direct***NET programs. By disabling the timeout, you can send one ***Direct***NET component without any communication problems. If you have this timeout disabled and a communication error does occur, you must restart communications by sending a retry or an End of Transmission (EOT) command.

ASCII / HEX Mode: Position 4 on SW5 selects between ASCII and HEX modes of data representation. Set the switch to match your network parameters.

Online / Offline Switch

In the Offline position, this switch logically disconnects the DCM from the network (just as if you pulled the cable from the connector.) Once this switch is moved to the Offline position, the DCM will not communicate with the network.

If you move the switch to the Online position, the DCM will communicate with the network, but not until the master sends another request for communication. This does not operate like the reset switch on many personal computers.



Address Switch

The DCM station address is set by the two rotary switches located on the front of the unit. Addresses are in hexadecimal format with valid addresses from 0 (only used for the master station) to hexadecimal 5A. The addresses do not have to be sequential, but each station must have a unique address.

The top rotary switch is used to set the most significant digit of the HEX address. The bottom switch is used to set the least significant digit the HEX address. For example, to set a DCM address of HEX 10 (decimal 16), set the top rotary switch to 1 and the bottom rotary switch to 0.

NOTE: The DCM address switch settings are only read at power up. If you've want to change the address and the DCM is already running, you'll have to cycle the system power for the base containing the DCM to initialize the change.

Even though the DCM address is set in hexadecimal, it's a good idea to remember the decimal equivalent. This is because the communications program and the ***Direct***SOFT package use the decimal equivalent of the HEX address. It's easy to convert from HEX to decimal.

HEX Format

0 1 2 3 4 5 6 7 8 9 A B C D E F

A = 10, B = 11, C = 12, D = 13, E = 14, F = 15

HEX 3C

3 x 16 = 48 + C = 12 = 60 decimal

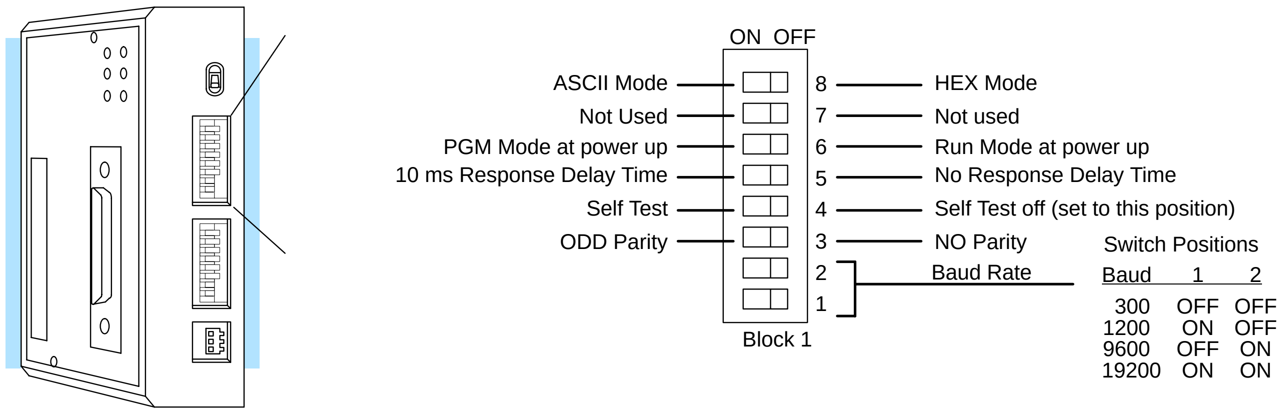
DL305 DCU Communications Switches

First, make sure you have the correct model of DCU. There are two models.

- D3–232–DCU (RS232C version)
- D3–422–DCU (RS422 version)

The DL305 DCU can only be used as a slave station interface. If you only have one master and one slave, you can use either version. If you have more than one slave you should probably use the the RS422 version. You can use the RS232C version, but you'll need an RS232C / RS422 converter for each one.

The communications parameters are set by the first switch block, which is located on the side of the unit.



DCU Side View

Baud Rate: The first two switches on block 1 are used to set the baud rate for the DCU. There are four baud rate selections available ranging from 300bps to 19.2Kbps.

Parity: Switch 3 on block 1 selects between the two parity options, odd or none.

Self-Test: Switch 4 on block 1 selects the factory self-test and should always be switched off. If the self-test is on, the DCU will not operate correctly.

Response Delay Time: Switch 5 on block 1 sets the response delay time. This sets how long the DCU will wait before it responds to each component of a ***Direct***NET communication request. If you're using all ***Direct***LOGIC™ equipment, a response delay is not required and you should turn off the switch.

The DCU is not recommended for use with modems since it may respond too quickly for some devices. If you encounter this problem, you can turn on the delay switch to provide a 10 ms delay. If this still does not work, check your device manual to see if the device requires more than a 10 ms delay. The DL340 CPU built-in ports provide more extensive delay timing than can be achieved with a DCU.

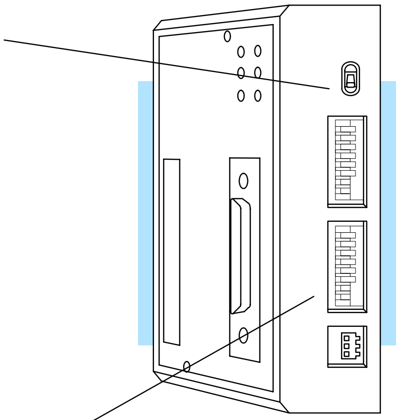
Mode at Power-up: Switch 6 on block 1 allows you to select the CPU operating mode when system power is supplied. If the switch is turned on, the CPU automatically enters Program mode when power is supplied. If the switch is off, the CPU automatically enters Run mode when power is supplied.

ASCII / HEX Mode: Switch 8 on block 1 selects between ASCII and HEX modes of data representation.

Online / Offline Switch

In the Offline position, this switch logically disconnects the DCU from the network (just as if you pulled the cable from the connector.) Once this switch is moved to the Offline position, the DCU will not communicate with the network. (The switch is set to Offline when you're using a handheld programmer.)

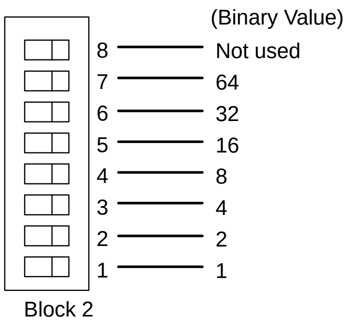
If you move the switch to the Online position, the DCU will communicate with the network, but not until the master sends another request for communication. This does not operate like the reset switch on many personal computers.



Address Switch

The DCU station address is set by the second switch block, which is located on the side of the unit. Unlike the DL405 DCM, the DCU uses decimal addresses. The decimal address is set in BCD (Binary Coded Decimal) format with valid addresses from 1 to 90 decimal. For example, to set an address of 10, you should turn on switches 4 and 2.

The addresses do not have to be sequential, but each station must have a unique address.



NOTE: The DCM address switch settings are only read at power up. If you've want to change the address and the DCM is already running, you'll have to cycle the system power for the base containing the DCM to initialize the change.

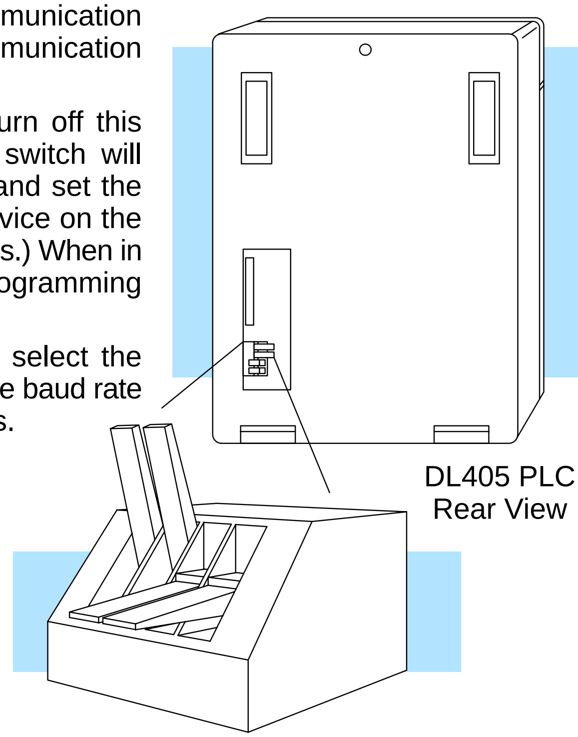
DL430 and DL440 PLC Communication Switches

If you look at the back of the DL440 CPU you will notice a small bank of switches. Switches 2, 3, and 4 are used to set the communication parameters for the bottom communication port.

Switch 2 — It's usually best to turn off this switch. If in the on position this switch will override the CPU slave address, and set the address to 1. (Remember each device on the network must have a unique address.) When in the off position you can use a programming device to set the address.

Switch 3 & 4 — These switches select the baud rate for the bottom port. Set the baud rate to match the other network stations.

Baud	3	4
300	OFF	OFF
1200	OFF	ON
9600	ON	OFF
19200	ON	ON

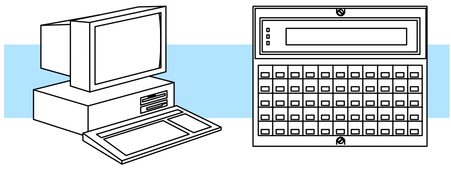


DL405 PLC Rear View

DL430 and DL440 Station Addresses

With ***Direct***SOFT, use AUX 56 from the Auxiliary functions menu to set the port parameters.

With the DL405 Handheld Programmer, use AUX 56 to set the port parameters. The following example shows how to use the handheld programmer to set the address.



DirectSOFT DL405 HP

NOTE: The PLC port address is set in decimal, not hexadecimal.

Select AUX 56

CLR CLR AUX 5 6 ENT ENT

AUX 5* CPU CFG
AUX 56 CPU N/W ADDRESS

Enter the Address (in decimal)

X X ENT

AUX 56 CPU N/W ADDRESS
N/W # 01

Select HEX or ASCII

To change the mode press .... →

then press enter. ENT

AUX 56 CPU N/W ADDRESS
HEX / ASCII

Select the Parity Option

To change the parity press .... →

then press enter. ENT

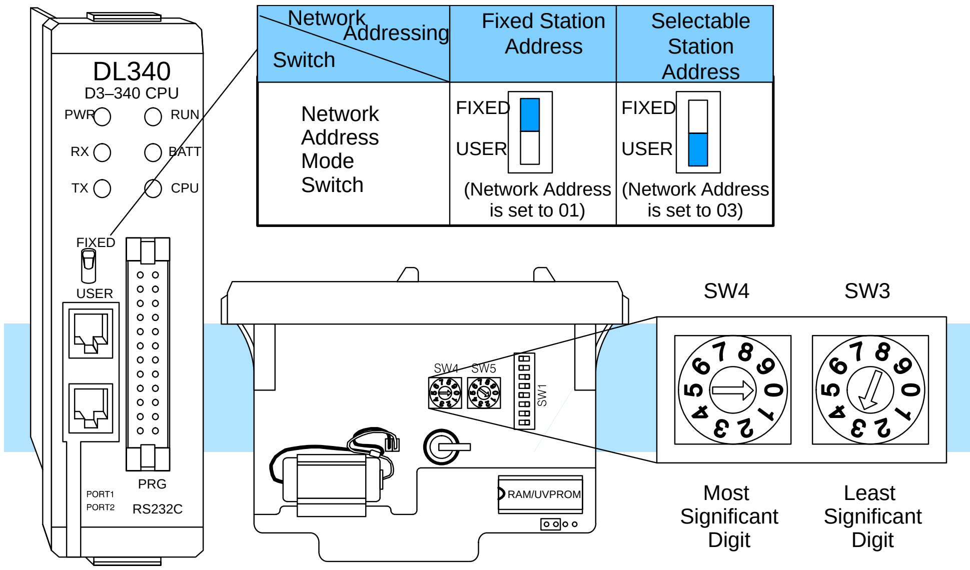
AUX 56 CPU N/W ADDRESS
NONE / ODD

Clear the Display

To clear the display press.... CLR

OK

DL340 Network Address Selection for Ports 1 and 2



Port 1 (RS232C): Network address selection is accomplished with the Network Address Mode Switch and the two rotary switches 3 and 4.

Network Address Mode Switch sets fixed or selectable network address.
Rotary Switch 3 sets the least significant decimal digit of the network address.
Rotary Switch 4 sets the most significant decimal digit of the network address.

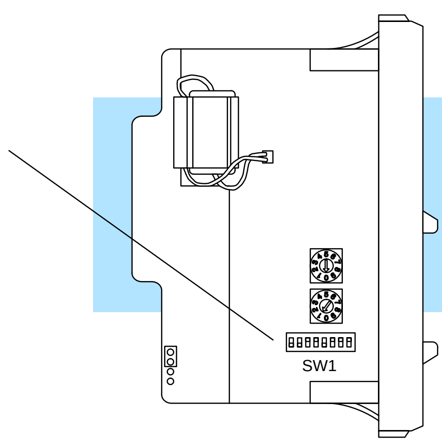
In the example above, when the Network mode switch is set to FIXED the network address will default to 01, when the Network mode switch is set to USER the network address (set with the rotary switches) is 03. Note, if the rotary switches are set to 00, the network address will default to 01.

Port 2 (RS232C): Network address selection is set by using a programming device to enter the value for the most significant digit and least significant digit in addresses R771 and R772 respectively.

DL340 Station Type Selection and Address Ranges

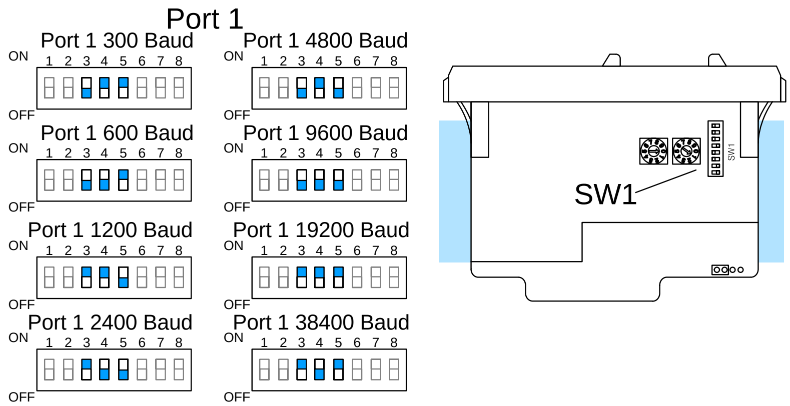
The station type for Port 1 is fixed as a Slave and cannot be changed. The station type for Port 2 can be selected by setting the appropriate switch positions (6 and 7) on the SW1 switch bank.

Port	Bit 6	Bit 7	Address Protocol	Range
1	N/A	N/A	Slave	1 – 90
2	Off	Off	Slave	1 – 90
	Off	On	Master	1 – 90
	On	Off	Peer	1 – 90
	On	On	Modbus®/ RTU	1 – 247



DL340 Baud Rate Selection

The following chart shows how to configure the baud rate for Port 1 (RS232C) of the DL340 using dipswitch SW1, switches 3, 4 and 5. Port 2 baud rate is set by using a programming device to enter the baud rate in address R773 (in BCD or HEX).



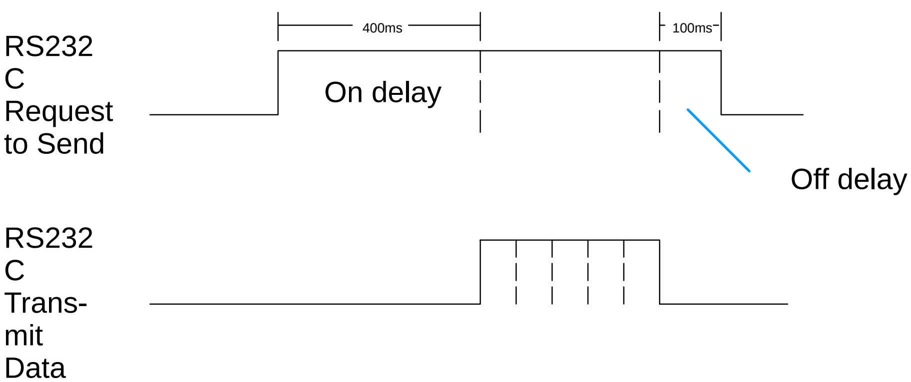
Port 2

Baud	R773
300	1
600	2
1200	3
2400	4
4800	5
9600	6
19200	7
38400	8
9600	0, 9 to FF

DL340 Selecting the Response Delay Time

You can use the Handheld Programmer of ***Direct***SOFT to select an on and off response delay time of up to 1980 ms. The time delay is calculated based on a preset number that is loaded into two memory locations. These presets indicate the number of 20 ms intervals that will be used as the delay. For example, an entry of 2 would result in a 40 ms response delay time.

Port	On Delay	Off Delay
Port 1	R776	R777
Port 2	R774	R775



R776 = 20, 20 x 20ms intervals = 400ms on delay

R777 = 5, 5 x 20ms intervals = 100ms off delay

DL340 Selecting Data Format (ASCII/HEX)

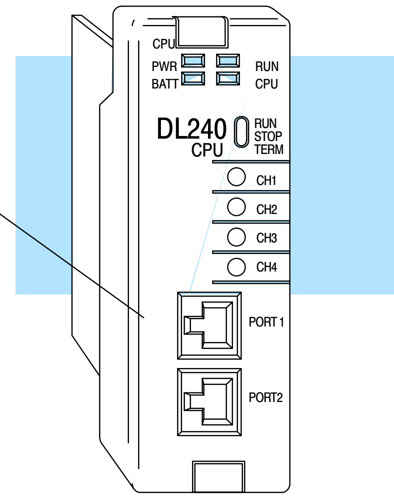
A special propose relay is used to select between ASCII and HEX transmission modes. When this relay is off, HEX mode is used. When this relay is turned on, ASCII mode is used. Off is the default state.

- Port 1 SP1077
- Port 2 SP1076

DL240 Communication Settings

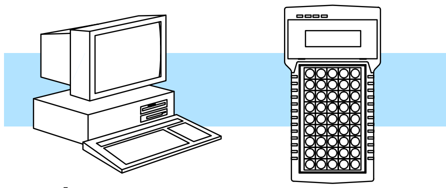
The DL240 CPU is set similarly to the DL405 CPUs, except that all parameters can be set with AUX 56 from ***Direct***SOFT or the Handheld Programmer.

DL240 PLC



The following example shows you can set the port parameters with the DL205 Handheld Programmer.

NOTE: The PLC port address is set in decimal, not hexadecimal.



Select AUX 56

CLR CLR AUX F 5 G 6 ENT

ENT

AUX 5* CPU CFG
AUX 56 CPU N/W ADDRES

Enter the Address (in decimal)

X X ENT

AUX 56 CPU N/W ADDRES
N/W # 01

Select Hex or ASCII

To change the mode press .... →

then press enter. ENT

AUX 56 CPU N/W ADDRES
HEX / ASCII

Select the Parity Option

To change the parity press .... →

then press enter. ENT

AUX 56 CPU N/W ADDRES
NONE / ODD

Select the baud rate

To change the baud rate press .... →

then press enter. ENT

AUX 56 CPU N/W ADDRES
300/1200/9600/19.2K

To clear the display and show the current address (instructions)

CLR

AND X7
OR X10

Communications Settings Checklist

You can avoid problems

The majority of problems are caused by incorrect switch settings. Make sure you have set the switches for each station. Have you...

1. Checked your computer documentation to obtain host master switch settings? (if applicable)
2. Set the communications switches for all stations?
3. Set the network addresses for all stations?

Now, you’re ready to create the communications programs.

What part of the manual should you use next?

Do you have a PLC as a master or a Peer as a Master network?

If you have a PLC as a master or a Peer as a master network, use Chapter 5, Creating RLL Communications Programs. This chapter describes the RLL instructions used to initiate network communication requests.

Do you have a Host as a master network?

If you have a Host as a master network, use Chapter 6, Creating ***Direct***NET Communications Programs. This chapter explains the ***Direct***NET protocol in detail.

5

RLL Communications Programs

In This Chapter. . . .

- — Why do you need a communications program?
- — Identifying the Master and Slave
- — Specifying the Amount of Data
- — Designating the Master Station Memory Area
- — Identifying the Slave Station Memory Area
- — Controlling the Communications
- — What part of the manual should you use next?

Why do you need a communications program?

The Master Initiates Requests

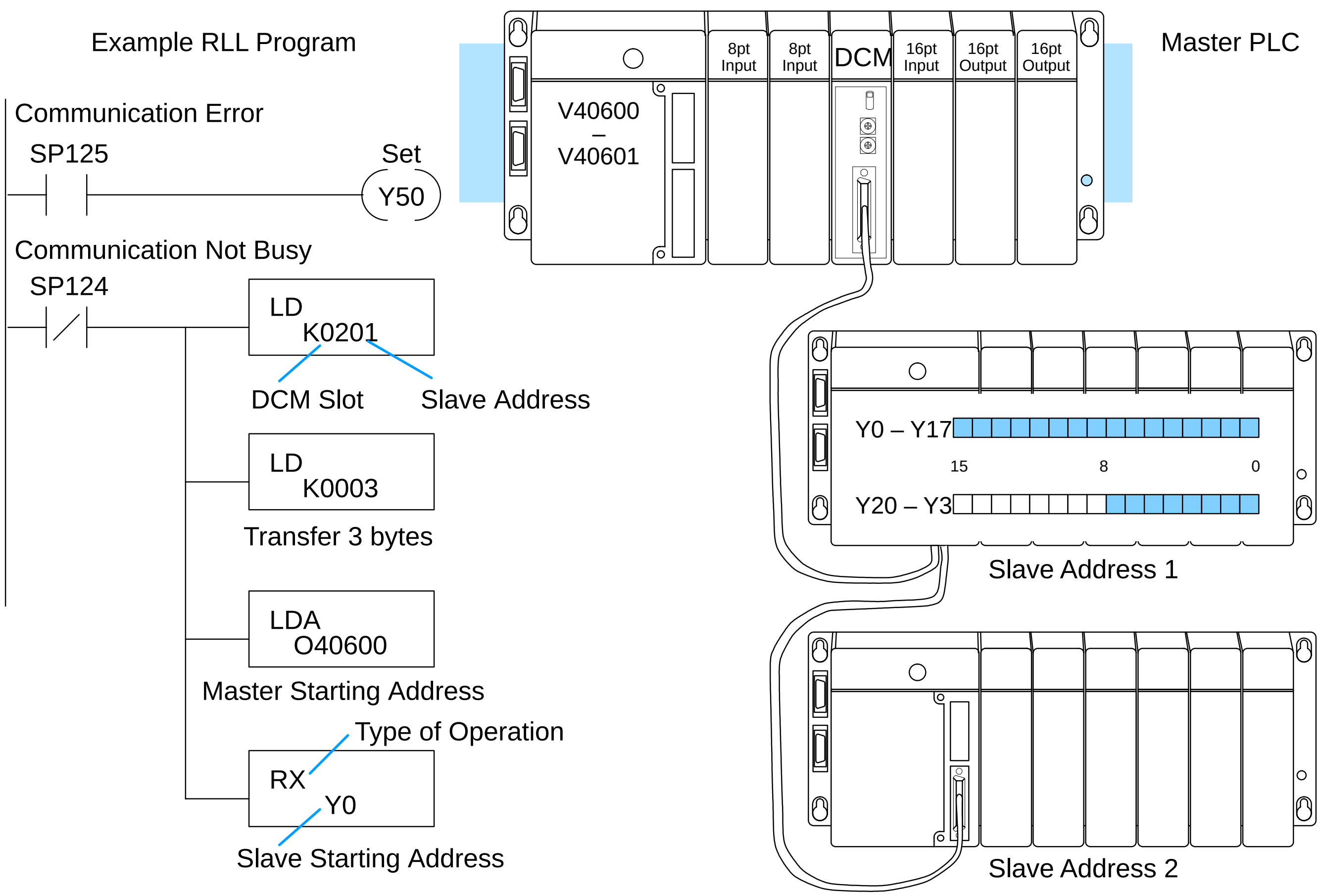
DirectNET is a master / slave network, therefore; the master station must initiate requests for network data transfers. If you're using a PLC as the master station, you use simple RLL instructions to initiate the requests.

Why Ladder Logic?

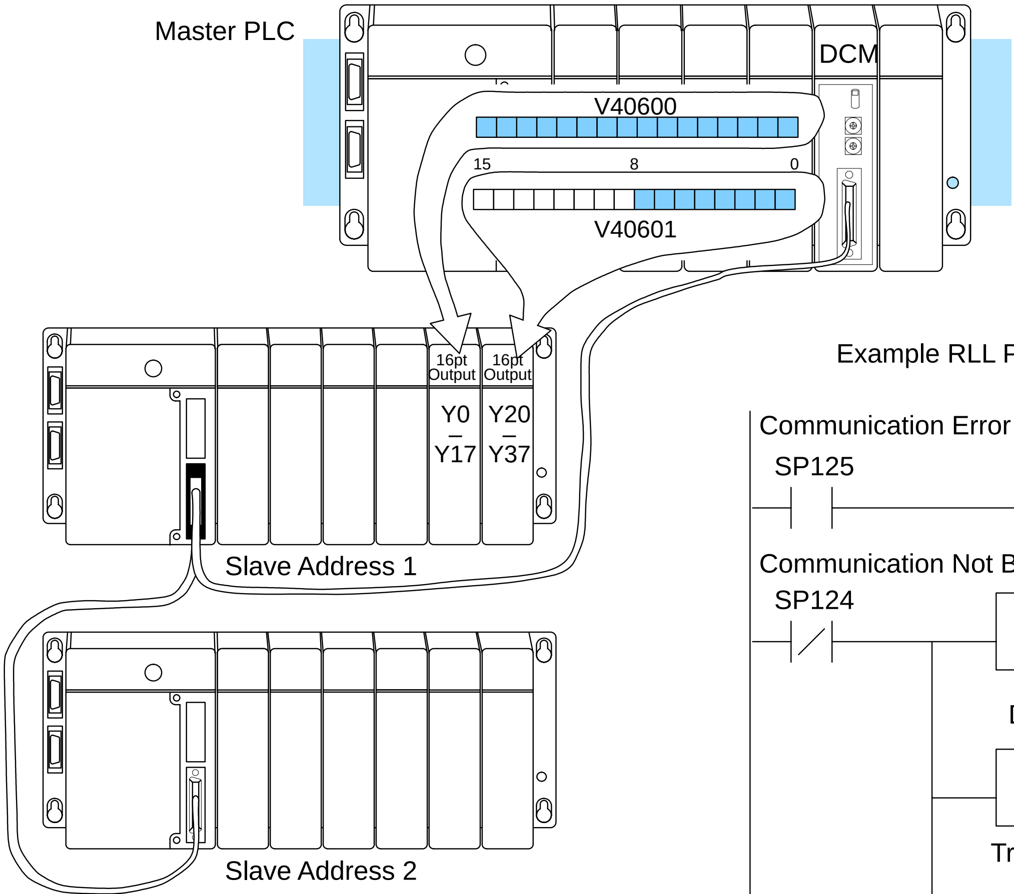
Since the DCM (network interface) does not contain any RLL, you have to use the PLC to issue the commands telling the DCM where to read or write data. The DCM takes the PLC commands and converts them into the appropriate ***Direct***NET commands. The RLL instructions identify the following items.

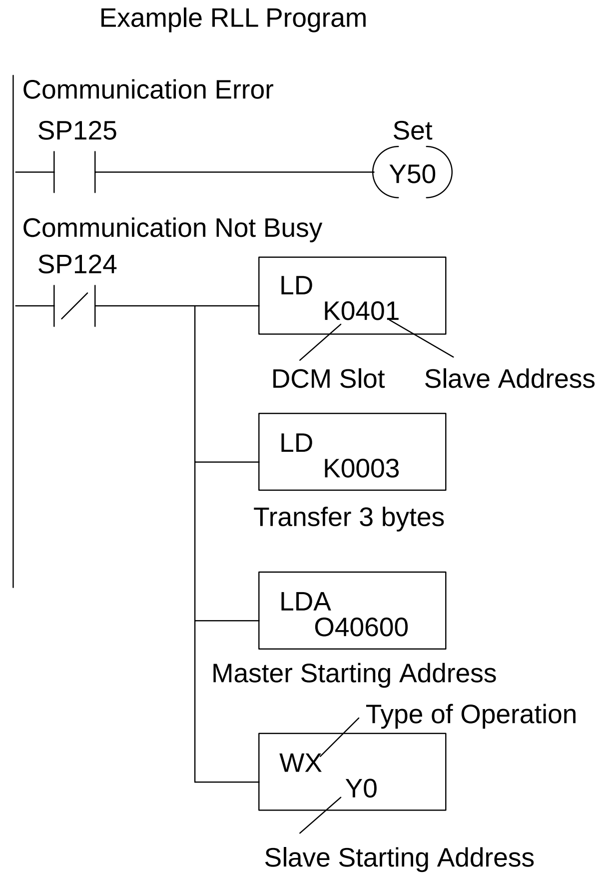
1. Slot location of the DCM master and the slave station address. (using the LD instruction)
2. Amount of data (in bytes) you want to transfer. (using the LD instruction)
3. Area of memory to be used by the master. (using the LDA instruction)
4. Area of memory to be used by the slave, and whether it is a read or write operation. (using either the RX or WX instruction)
5. Interlocks for communication timing and multiple RX and WX routines.

This example reads 3 bytes of data from Slave Address #1,(starting at Y0), into the Master PLC starting at V40600 (Control Relays).



This example writes 3 bytes of data from the Master Station (starting at V40600) to Y0 – Y37 in Slave Station #1.





The following paragraphs explain each operation and provide some helpful hints to make your programs simple and easy to follow.

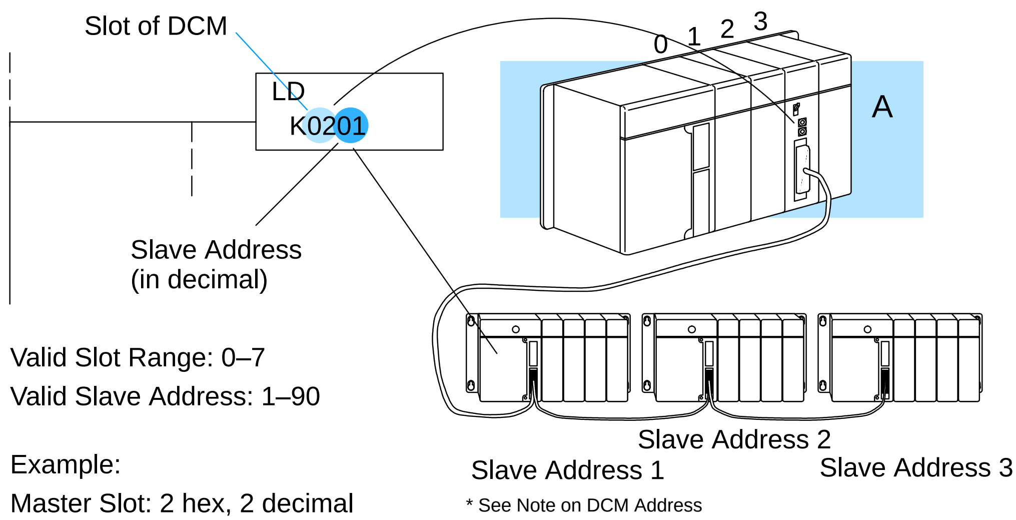
Identifying the Master and Slave

Location of Master and Slave

Whenever the WX or RX instructions are used, they must be preceded with the necessary LD and LDA instructions to set up the communication addresses, data to transfer and the location for the transference. The first Load (LD) instruction identifies the slot location of the DCM master and the slave station address. (Remember, the slot numbers start at 0.)

The constant (K) portion of the instruction actually contains two pieces (bytes) of information. The first two digits specify the DCM master location and the second two digits specify the slave station location.

It is necessary to specify both master and slave locations because you can have more than one DCM master in the base and you can have up to 90 slave stations for each master.



Valid Slot Range: 0–7

Valid Slave Address: 1–90

Example:

Master Slot: 2 hex, 2 decimal

Slave Address: 1 hex, 1 decimal

NOTE: The LD instruction K value is entered in decimal, but the DCM master and slave addresses are in HEX (The DCU, DL340, DL430 and DL440 all have their addresses set in decimal.) You have to convert the HEX addresses for the DCM to their decimal equivalent for the LD instruction. It's easy to convert from HEX to decimal.

HEX Format

0 1 2 3 4 5 6 7 8 9 A B C D E F

10 11 12 13 14 15

HEX 3C

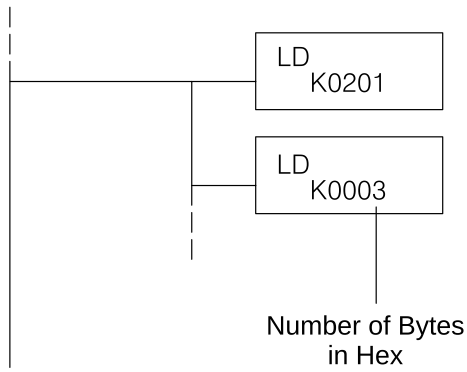
3 x 16 = 48 + **C** = 12 = 60 decimal

Specifying the Amount of Data

Number of Bytes to Transfer

The second LD instruction indicates the amount of data (in bytes) that needs to be transferred. You have to specify the amount of data in complete bytes. For example, Y0 – Y27 would be three bytes of data.

The different PLC families do not always use the same types of memory or the same byte boundaries. For example, the DL305 does not use a separate data type for input and output points. Appendices D, E, F provide complete listings of the DL405, DL305 and DL205 memory types.



Example:

3 bytes of data to be transferred

The number of bytes specified also depends on the type of data you want to obtain. For example, the DL405 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 205 / 405 Memory	Bits per unit	Bytes
V memory T / C current value	16 16	2 2
Inputs (X, GX, SP)	8	1
Outputs (Y, C, Stage, T/C bits)	8	1
Scratch Pad Memory	8	1
Diagnostic Status	8	1

DL305 Memory	Bits per unit	Number of 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	2
Diagnostic Status (5 word R/W)	16	10

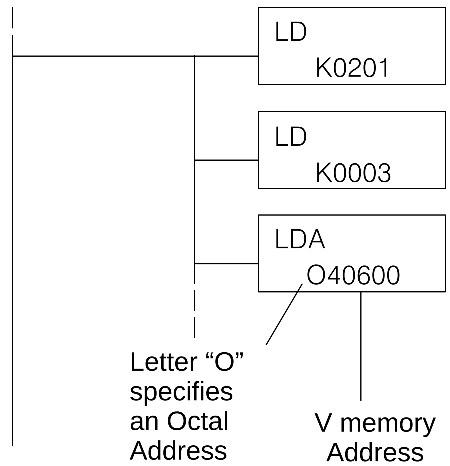
Designating the Master Station Memory Area

Memory Area of Master

The Load Address (LDA) instruction specifies the V memory area of the master that will be used. This is the starting address (in octal). Additional sequential locations may be used, depending on the number of bytes that are being transferred. Since all DL405 data is mapped into V memory, you can easily access the data you need.

If you are reading information from the slave station, this is the destination area, or the area where the master will store the information.

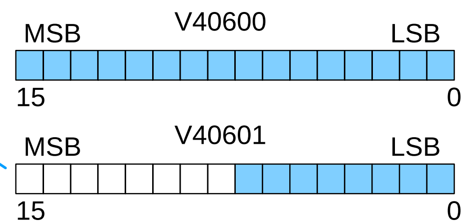
If you are writing information to the slave station, this is the source area, or the area where the master will obtain the information that will be transferred to the slave.



Example:

V memory location 40600 will be the starting point of the data transfer area for the master. The following locations will be used to store the data.

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.



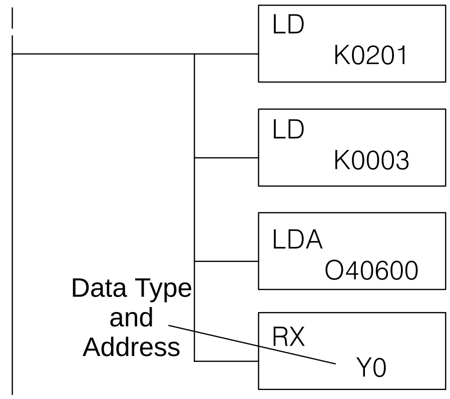
Identifying the Slave Station Memory Area

Memory Area of Slave to Read or Write

The Read Network (RX) or Write Network (WX) is the last instruction in the routine. Use the RX if you want to read data from the slave, or use the WX instruction if you want to write data to the slave.

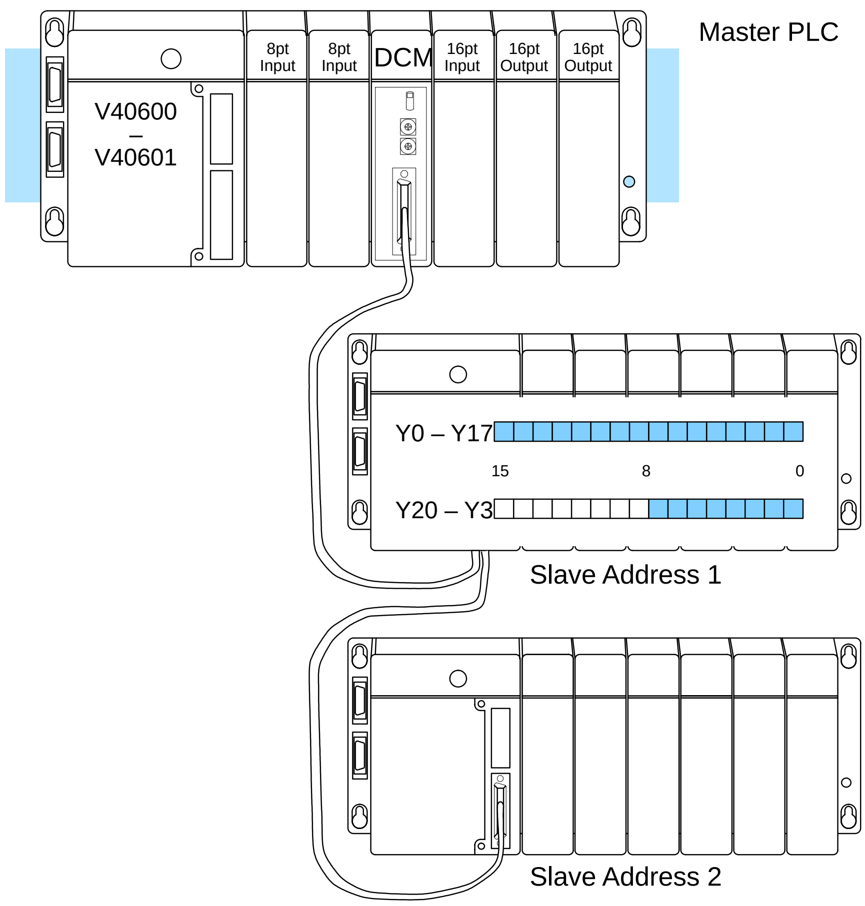
You have to specify the data type and the starting address (in octal) for the slave. (Remember, you have to specify a data type that will work correctly with the number of bytes specified.)

If you use the RX instruction, the data will be read from the slave starting at the address specified. If you use the WX instruction, the data will be written to the slave starting at the address specified.



Example:
Read from slave starting at Y0.

NOTE: If you are using an RLL communications program to transfer data to or from a DL305 slave station, the data type is slightly different. For example, the DL305 I/O points are accessed with the GY data type. AppendixE provides a complete listing of the memory types and cross references for the DL305 family.



Controlling the Communications

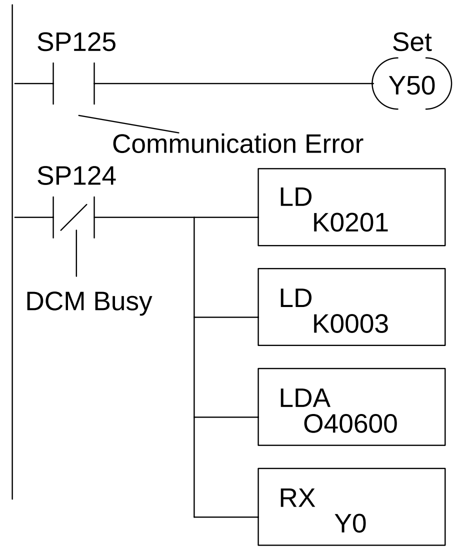
Communications Special Relays

When you execute communication with a DCM, chances are good the communication may take longer than the actual PLC scan. If the DCM is busy, you shouldn't initiate another request until it is finished. Fortunately, there's an easy solution for this.

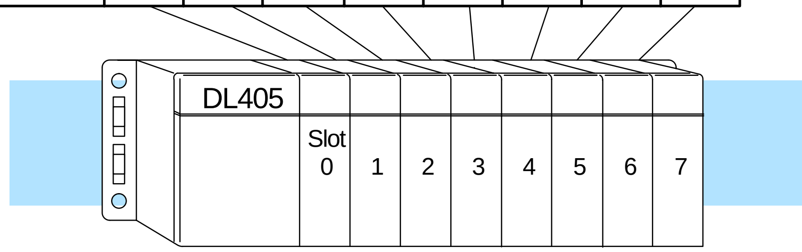
There are two SPs (special relays) for each slot in the CPU base which are used to provide communication status. For example, slot 0 has SP120 and SP121. SP120 is the Busy relay and, when turned on, indicates the DCM is busy. SP121 indicates there is a communication error for slot 0.

You should always use the DCM Busy SP in your RLL programs to ensure the DCM is ready.

The communication error SP is optional, but it's a good way to monitor the communication status in the RLL program. If you use the communication error SP, make sure you place it at the beginning of your communication routines. This is because the communication error relay is always reset (turned off) whenever an RX or WX instruction is executed.



Special Purpose Communication Relays								
Communication Busy	SP120	SP122	SP124	SP126	SP130	SP132	SP134	SP136
Communication Error	SP121	SP123	SP125	SP127	SP131	SP133	SP135	SP137



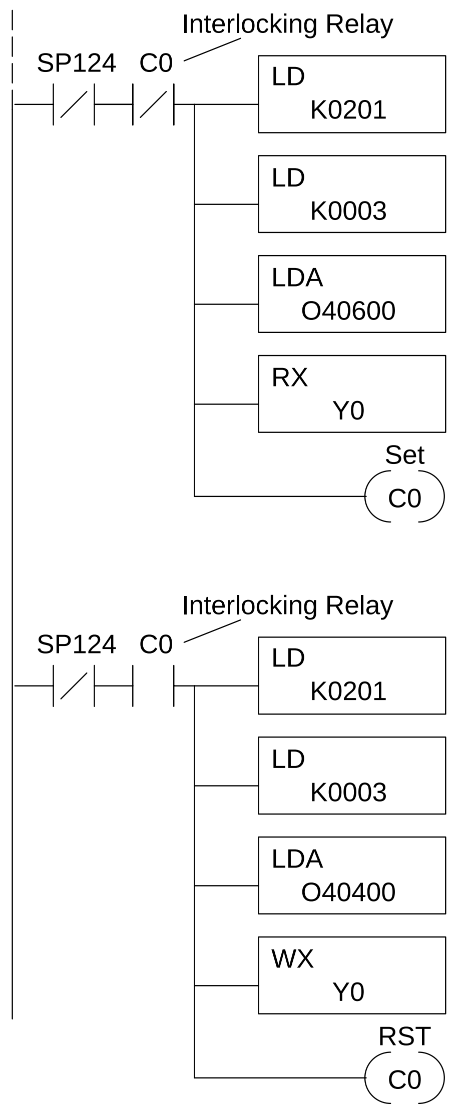
Multiple Read and Write Interlocks

If you're 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 the DCM can only handle one routine at a time.

In the example, once the RX instruction is executed, C0 is set. When the DCM has finished the communication task, the second routine is executed and C0 is reset.

If you're using RLL*Plus*, you can just put each routine in a separate program stage to ensure proper execution. In most all cases, RLL*Plus* is a much more efficient way to create automation program.

Appendix A provides a master / slave example with both RLL and RLL*Plus* program descriptions.



RLL Communications Programs

What part of the manual should you use next?

Start the Network

Once you've created the communications program, you can start the network. Chapter 7 provides information concerning network operation and troubleshooting. Appendices A and B provide examples of PLC master and PLC Peer networks. Both of these networks use RLL instructions for the communications program.

6

*Direct*NET Host Communications Programs

In This Chapter. . . .

Why do you need a communications program?

The Master Initiates Requests

Since ***Direct*NET** is a master / slave network, the master station must initiate requests for network data transfers. If you're using a host as the master station, you will need to use a communications program written with the ***Direct*NET** protocol.

***Direct*NET Programs**

The communications program used with a hosted network is more complex than the simple RLL instructions used with the other configurations, but the concept is the same. The host is the ***Direct*NET** master and must use a ***Direct*NET** protocol communications program to initiate all network requests to read or write data. These communication programs can be written in many different languages, such as BASIC, C, etc. and must include the appropriate ***Direct*NET** commands.

Here's an example of a ***Direct*NET** program. (This is just part of the program.)

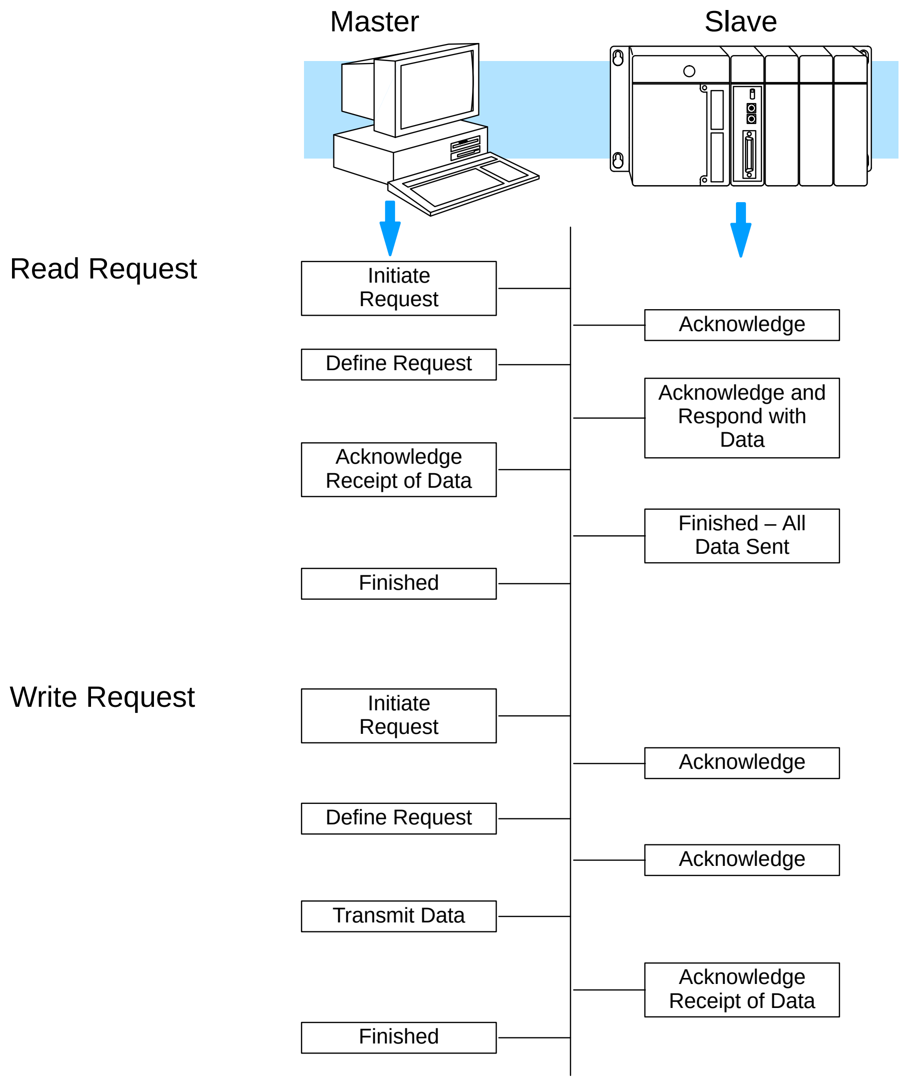
***Direct*NET Program in BASIC**

```
10    REM Program to read X0–X7 from a DL405 PLC
20    REM
30    REM Define all variables
40    REM
50    REM Change the slave address in HEX at line 60 if required.
60    SLAVEADDRESS=&H1
70    DATATYPE$=CHR$(&H32)
80    DATAADDR$=CHR$(&H30)+CHR$(&H31)+CHR$(&H30)+CHR$(&H31)
90    COMPLETEBLK$=CHR$(&H30)+CHR$(&H30)
100   PARTBLK$=CHR$(&H30)+CHR$(&H32)
110   MASTERADDR$=CHR$(&H30)+CHR$(&H30)
120   NORMAL$=CHR$(&H4E)
130   SLAVEADDR$=HEX$(SLAVEADDRESS)
140   IF LEN(SLAVEADDR$)<2 THEN SLAVEADDR$="0"+SLAVEADDR$
150   OFFSETADDR$=CHR$(&H20+SLAVEADDRESS)
|
|
|
```

NOTE: This manual does not show you how to build communications programs that manage the data storage and communications ports. You should check the documentation that came with your programming software to determine the appropriate techniques to solve these requirements.

The following diagram shows the general structure of the communications. The program must:

- identify the slave station.
- indicate the type and amount of data to transfer.
- manage the communications between the master and slave.



The remainder of this chapter discusses the individual elements of ***Direct*NET** protocol programs.

Modes of Operation

Transmission Bytes

***Direct*NET can transfer a maximum of 65,791 bytes (256 blocks 256 bytes each + an additional 255 bytes) in a single request. The actual amount of system information that is transferred depends on the mode of operation.**

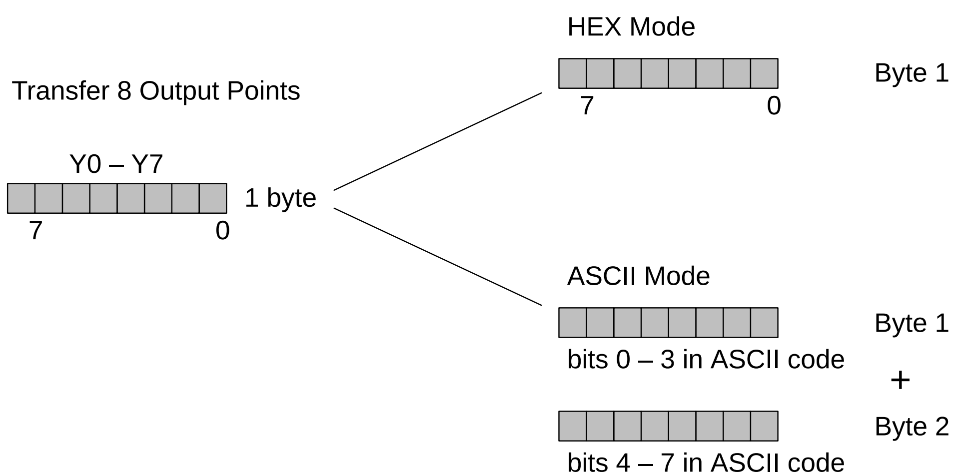
HEX or ASCII Mode

There are two modes used with the ***Direct***NET protocol, HEX or ASCII. You must choose the mode of operation before you write the program. The major difference is in the way the data is represented in the data packet. ASCII mode requires twice as many bytes to transfer data. There are also minor differences in the command structure which affect the way the enquiries and headers operate.

NOTE: This only applies to a Host master, since the DCM has the capability to detect the data transfer mode and adjust for the additional number of bytes to be used when transferring in ASCII mode.

If you're transferring small amounts of data, or if the data is not being used to control system timing, then it's generally easier to use ASCII mode for ***Direct***NET programs. You should use HEX mode if you're transferring large amounts of data and you need the fastest possible communication time.

The following diagram shows the difference between HEX and ASCII modes.



Data Type Byte Requirements

In HEX mode, the number of bytes transferred is equal to the number of bytes for the selected data type. ASCII mode requires twice as many bytes to transfer the same data. Here’s a listing of the data types and their corresponding byte requirements.

DL205/405 Data Type	Description	Bits per unit	Number of bytes	
			HEX	ASCII
31	V memory T / C current value	16 16	2 2	4 4
32	Inputs (X, GX, SP)	8	1	2
33	Outputs (Y, C, Stage, T/C bits)	8	1	2
39	Diagnostic Status	8	1	2

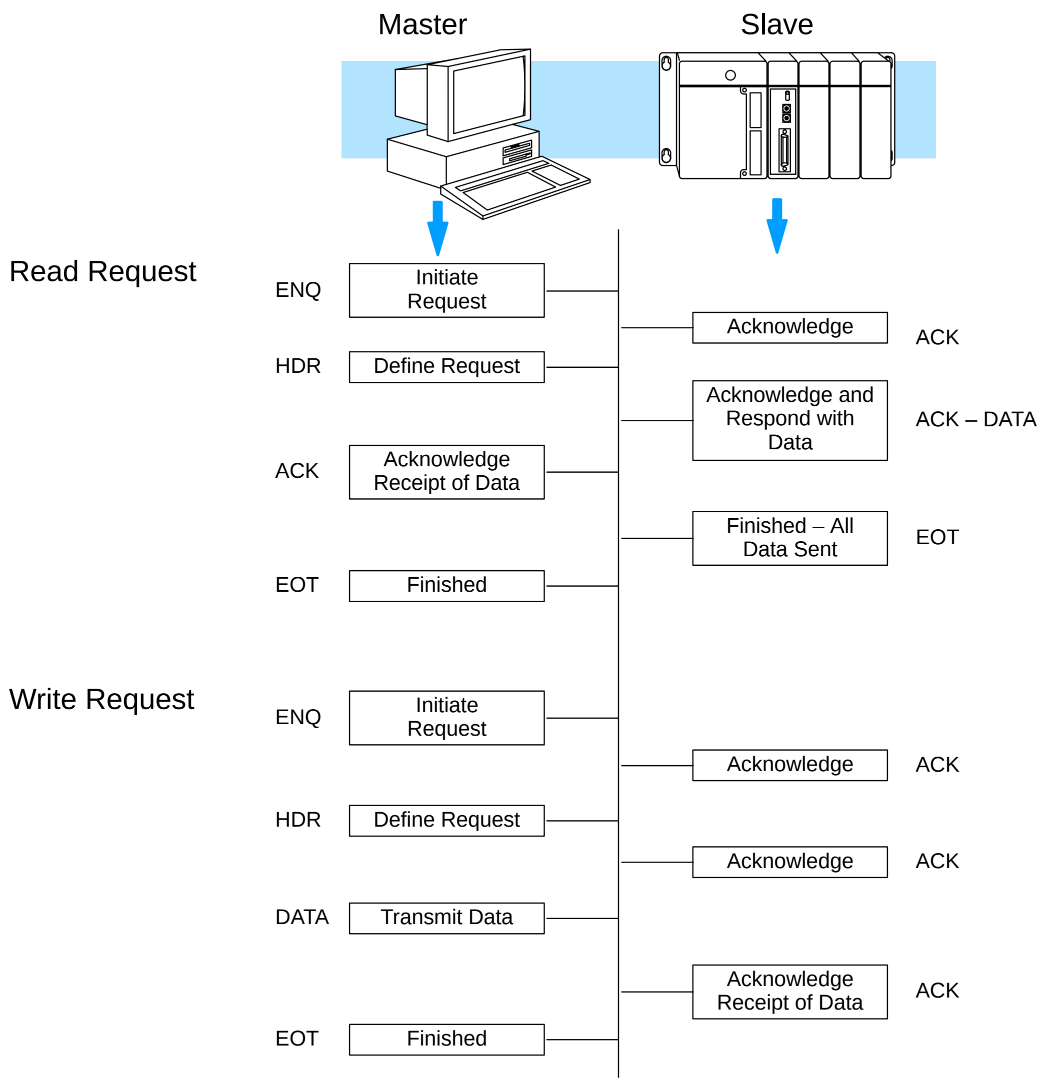
DL305 Data Type	Description	Bits per unit	Number of bytes	
			HEX	ASCII
31	Data registers T / C accumulator	8 16	1 2	2 4
33	I/O, internal relays, shift register bits, T/C bits, stage bits	1	1	2
39	Diagnostic Status (5 word R/W)	16	10	20

Protocol Components

All ***Direct***NET program read and write requests use the following protocol components.

- Enquiry (ENQ) – initiates a request (from the master) with the slave stations.
- Header (HDR) – defines the operation as a read or write, the slave station address, and the type and amount of data to be transferred.
- Data (DATA) – the actual data that is being transferred.
- Acknowledge (ACK) – verifies communication is working correctly.
- End of Transmission (EOT) – indicates the communication is finished.

The following diagram shows how the protocol components are used with read and write requests.



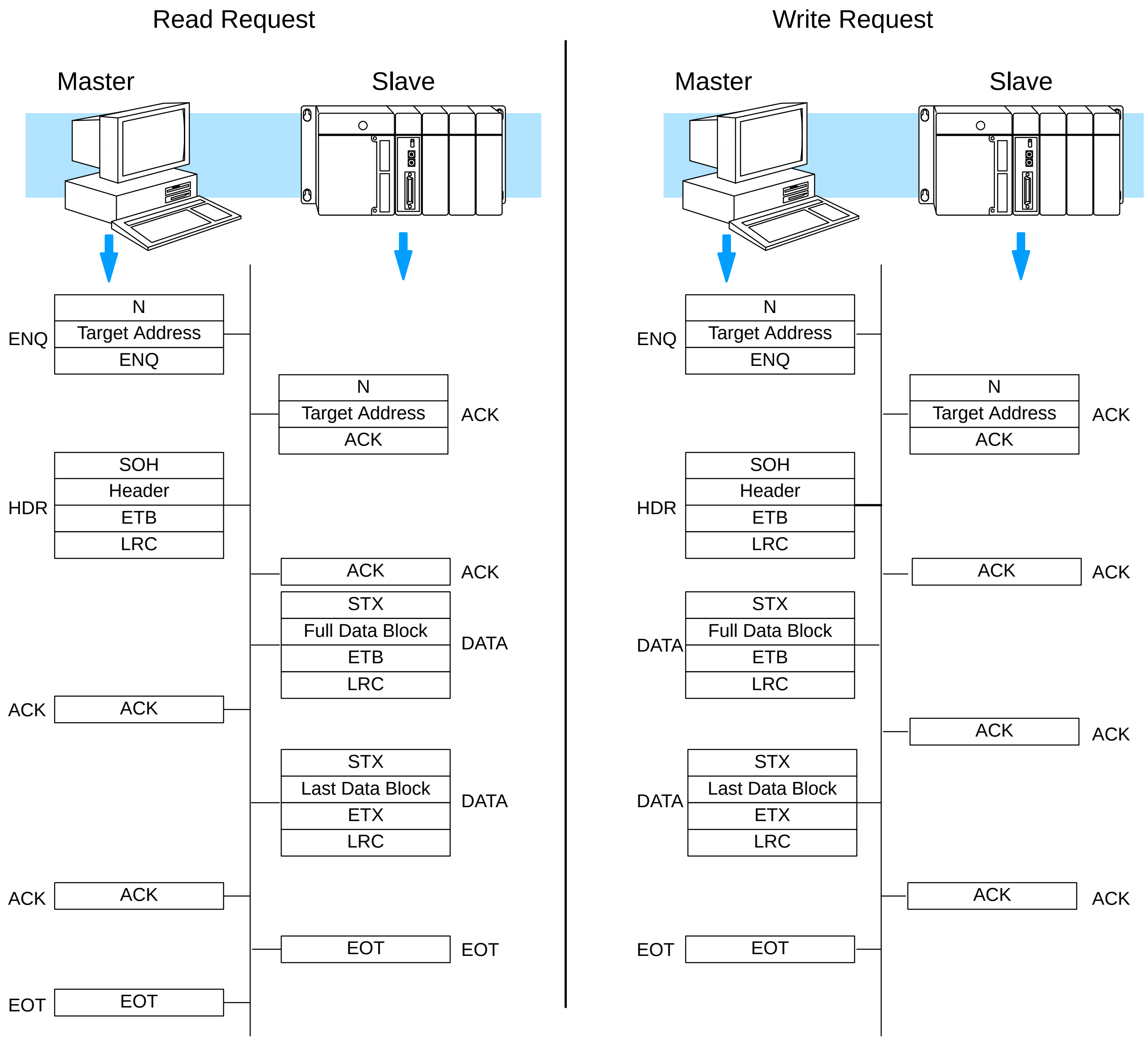
Controlling the Communications

Control Codes

All read or write requests use ASCII control codes and a Longitudinal Redundancy Check (LRC) to manage the communications between the master and slave. The control codes identify the beginning and ending of the protocol components such as, enquiry, acknowledge, etc. The LRC is used to ensure the data was transmitted and received correctly.

Symbol	HEX ASCII Code	Description
ENQ	05	Enquiry – initiate request
ACK	06	Acknowledge – the communication was received without error
NAK	15	Negative Acknowledge – there was a problem with the communication
SOH	01	Start of Header – beginning of header
ETB	17	End of Transmission Block – end of intermediate block
STX	02	Start of Text – beginning of data block
ETX	03	End of Text – End of last data block
EOT	04	End of Transmission – the request is complete.

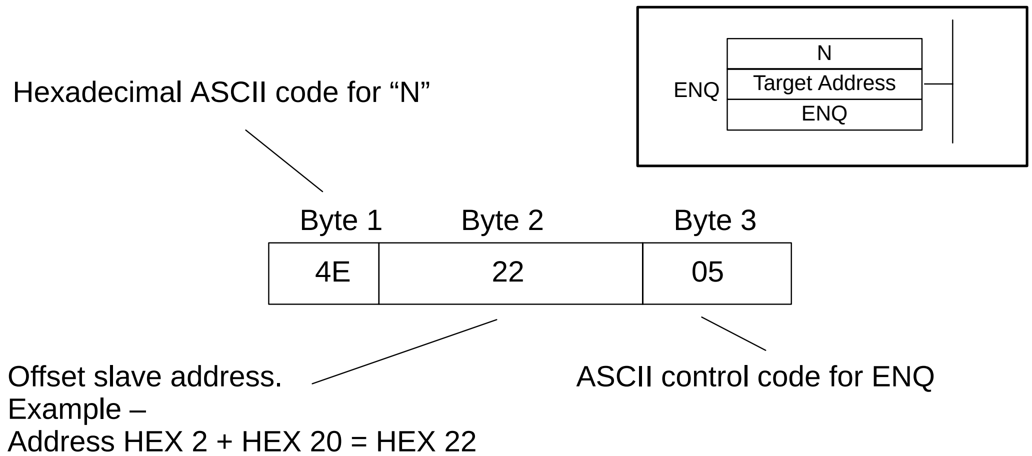
The following diagram shows how these control codes are combined with the protocol components to build the ***Direct***NET format for read and write requests. Note, the slave components are automatically generated as a response to the requests from the master station. Your custom ***Direct***NET program must generate the protocol components for the master station.



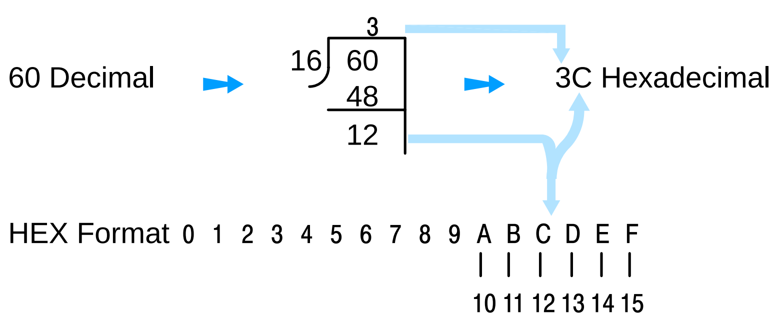
Initiating the Request

Enquiry ENQ

The Enquiry is a three character message that initiates the request with the appropriate slave station. The message always begins with 4E (“N”), which means normal enquiry sequence. The second character contains the offset station address, which is the station address plus HEX 20. The last character is the ASCII control code for ENQ.



Note: Slave addresses for the DL430, DL440, DL340 and the DCU have been set in decimal. It will be necessary for you to convert the address from decimal to the Hexadecimal equivalent before adding the HEX 20 offset.



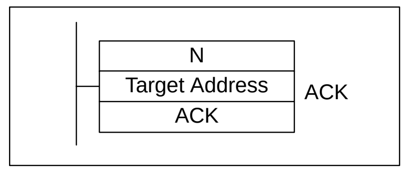
Acknowledging the Request

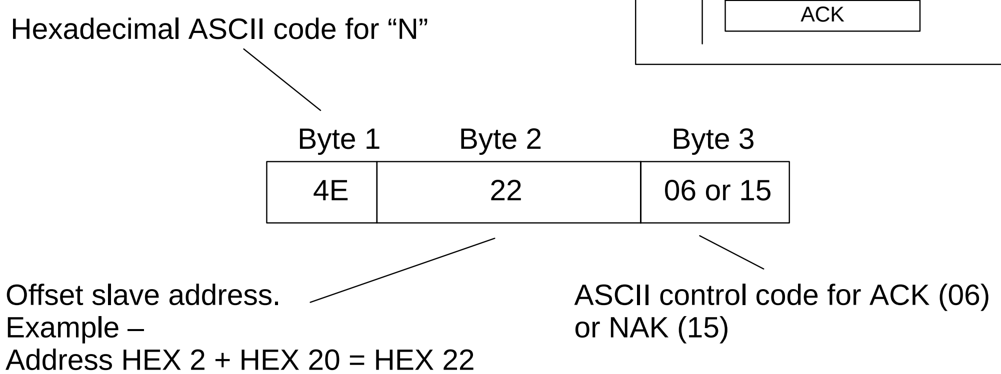
Acknowledge ACK – NAK

The three character acknowledge commands are used by both the master and slave stations to indicate the status of the communication. An ACK is used if the information was transmitted (or received) without any problems. If there are problems, Not Acknowledge (NAK) is used.

A NAK will also be returned from the slave if something is incorrect in the header or data packet. This could be incorrect byte boundaries, an invalid address, etc. If the master receives a NAK response, it can either try to re-transmit the data, or it can terminate the request and try again.

The first two characters are the same as the Enquiry sequence. The third character is the control code for an ACK or NAK.





Delayed Response to an Enquiry

When text is being transmitted over the network, there may be character combinations that are identical to an enquiry sequence. Network slave stations would interpret the character sequence as an enquiry even though it was actually data being sent to an identified slave.

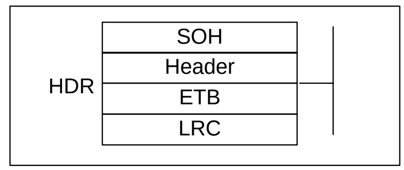
A delay has been implemented to automatically eliminate this possibility. The delay occurs between the receipt of an enquiry from the master and the acknowledgment response from the slave. When the slave recognizes an enquiry sequence an internal timer (with the time preset to the amount of time to transmit 2 characters) is started. The slave ignores the enquiry if another character is received before the timeout period has elapsed.

Defining the Request

Header – HDR

The header is a 17-byte (18-byte for ASCII transmissions) message that defines the operation. It is sent by the master station and specifies the following.

- type of operation (read or write)
- type of data being transferred
- data address
- number of complete data blocks
- number of bytes in the last data block



ASCII Coded Representation Example

Byte:	1	2, 3	4	5	6, 7	8, 9	10, 11	12, 13	14, 15	16	17(18)	
	01	3034	30	31	3431	3031	3031	3930	3031	17	08	Hex
											3038	ASCII
	SOH	Target Slave	Read or Write	Data Type	Starting Address MSB	Starting Address LSB	Number Com-plete Blocks	Bytes in Last Block	Mas-ter ID	ETB	LRC	

Byte 1:
Start of Header

The first byte in the header is the ASCII control code (01) that indicates this is the beginning of a header.

Bytes 2 & 3:
Target Slave
Address

The second and third bytes of the header indicate which slave station will be used. This is the normal slave station address (in HEX ASCII code) that you assigned during the network setup. This *is not* the offset slave station address, (with 20 HEX added to the address), that is used with the enquiry sequence. For example, a slave station with address 04 would be 3034 in ASCII code.

The table below shows how to decode the HEX/ASCII slave address. Remember if the slave address is in HEX it will be necessary to translate the address to decimal before decoding the HEX ASCII address. In the example below a slave station has a network address of 04 decimal and the equivalent HEX ASCII code is 3034.

Decimal or ASCII Station Address
04

ASCII Character	0	1	2	3	4	5	6	7	A	B	C	D	E	F
HEX Code	30	31	32	33	34	35	36	37	41	42	43	44	45	46

3034
Target Slave Address
in HEX ASCII Code

Byte 4: Read or Write

Byte 4 indicates whether the operation is a read or write request. A value of HEX ASCII 30 is read, HEX ASCII 38 is write.

Byte 5: Data Type

This byte identifies the type of memory to be accessed. Appendices D–F provide a complete listing of the data types and memory references for product families.

Bytes 6 & 7: Starting Address MSB

The address is the starting point for the data transfer. The data is transferred from this point forward. For example, to transfer the first 32 X input points from a DL405 PLC the starting address would be V40400. The request would actually obtain V40400 and V40401 since there are 16 points per V-memory location.

Bytes 6 and 7 define the most significant byte of the ASCII coded memory address. For example, the reference address for V40400 is 4101. This is obtained by converting the octal number to hexadecimal and adding 1, the most significant byte of this value is then decoded into HEX ASCII. The most significant byte would be HEX 41 or HEX ASCII 3431. Appendices D–F provide complete references for the addresses used in the various ***Direct***LOGIC™ PLC families.

Bytes 8 & 9: Starting Address LSB

These bytes define the least significant byte of the address obtained in the step above. So to continue on with our example of reading the first 32 inputs at memory location V40400 from a DL405 PLC, the reference value for this location would be 4101. The least significant byte of the reference value (01) would be decoded to 3031 in HEX ASCII.

Bytes 10 & 11: Complete Data Blocks

This is the coded number of complete data blocks that should be transferred. ***Direct***NET can transfer 256 bytes in a single data block. Take the number of bytes and divide by 256 to obtain the number of complete data blocks. This results in a valid range of 00 – FF HEX, or 3030 – 4646 HEX ASCII. If you're not transferring at least 256 bytes, then this field should be HEX 00 or HEX ASCII 3030.

For example, if you are transferring 200 V-memory locations, you would have the following: 200 x 2 bytes per location = 400 bytes. 400 bytes / 256 per block =1 complete block, with 144 bytes remaining. For one (01) complete data block, the value entered for this field would be the HEX ASCII code of 3031.

Bytes 12 & 13: Partial Data Block

This is the HEX ASCII coded number of bytes in the last data block. If you did not have a an even number of complete data blocks, then you had some remaining bytes. Enter the number of remaining bytes here. The valid range is 00 – FF HEX, or 3030 – 4646 HEX ASCII.

For example above with 144 bytes (or 90 bytes in HEX) the value entered in this field would be the HEX ASCII code of 3930.

Bytes 14 & 15: Master Station ID

This is the master station address. Since the master station should be address 0 or 1, this field is always HEX ASCII 3030 or 3031 for addresses 0 and 1 respectively.

Byte 16: End Transmission

Byte 16 always contains the HEX ASCII code for End of Transmission Block. This field always contains HEX ASCII17.

Byte 17: Longitudinal Redundancy Check

This is a checksum that is used to verify the communications were received without any errors. This is calculated by taking the exclusive OR of the bytes between the start of header (SOH) and the end of transmission (ETB) bytes (bytes 2 – 15). If you're using ASCII format, then this actually takes two bytes which makes the header an 18 byte message. The LRC is explained in more detail later in this chapter.

Transferring Data

Data Blocks

The data blocks contain the actual data that is being transferred between the master and slave station. ***Direct*NET** transfers data in full blocks of 256 8-bit bytes, or partial blocks of less than 256 8-bit bytes. The 256 byte limit does not include control characters that signal the end of the data. To determine the number of full blocks, divide the number of bytes by 256. The remainder is the number of bytes in the partial data block.

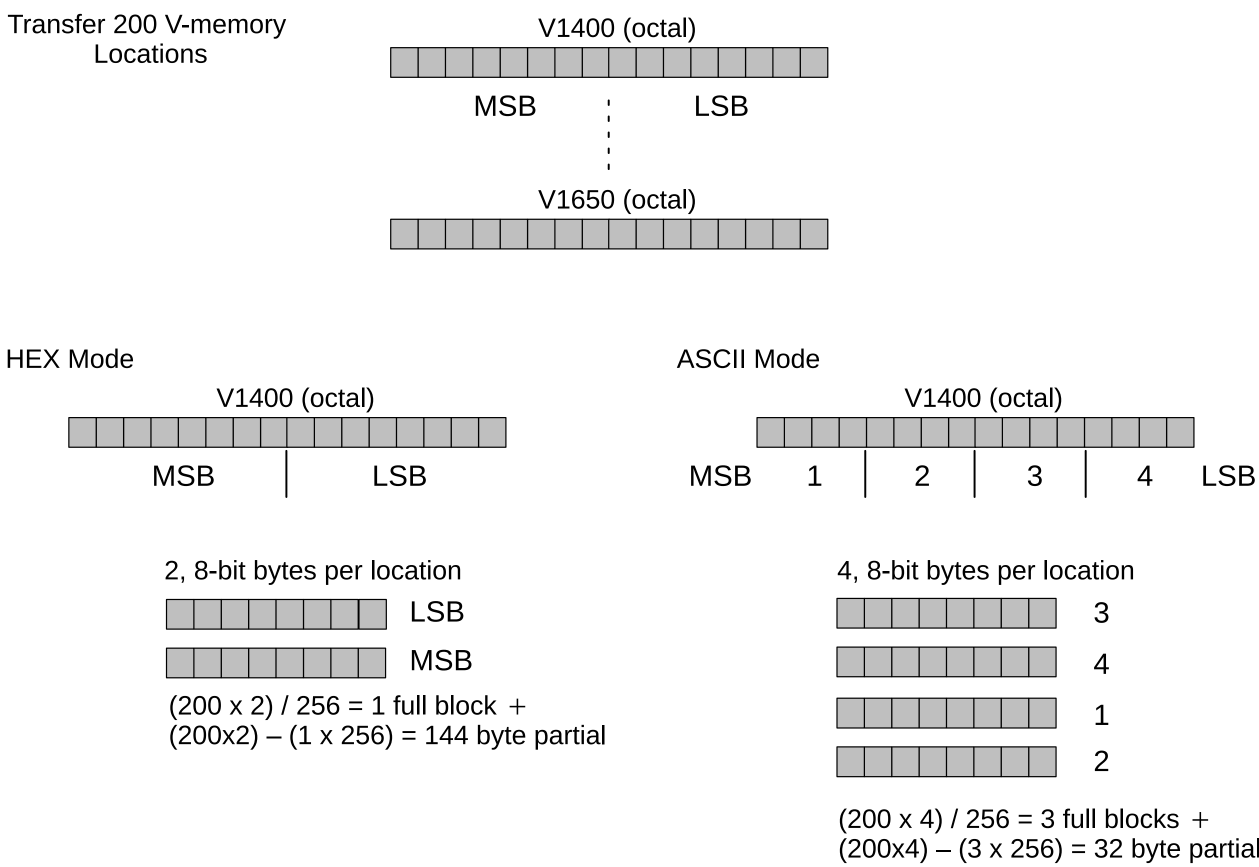
Since ASCII mode requires twice as many bytes for the data, you can transfer more information per request with HEX mode.

Transmission Sequence

Since the data is transmitted in bytes, it is important to understand how the original value is separated during transmission. ***Direct*NET** uses a simple byte swapping process where the least significant byte is transferred first. In ASCII mode, the original data is split into 4-bit units and then converted into 8-bit bytes.

Memory types that only use 1 byte are also treated in the same manner. For example, a 1-byte memory type would yield 1 byte in HEX mode, but two bytes in ASCII mode (4 bits converted into 2, 8-bit bytes).

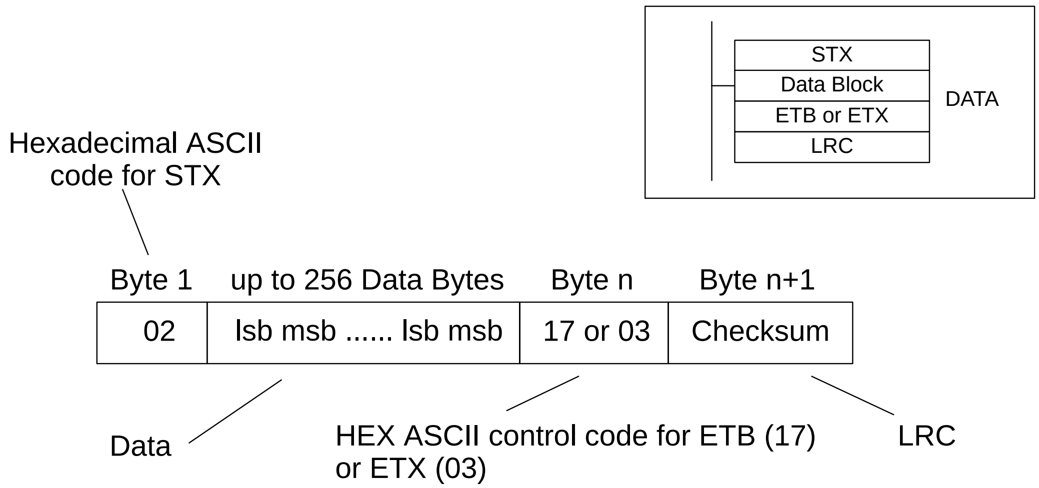
The following diagram shows the differences between HEX and ASCII modes.



Start of Text, End of Block, End of Text

The HEX ASCII control codes that indicate the beginning and end of data blocks are used to manage the data transfer. Start of Text (STX) indicates the beginning of a data block. If there are several blocks, all but the last block will terminate with the End of Block (ETB) code. The last block always ends with End of Text (ETX). All transfers also include an LRC checksum. (For a data block, the checksum is the exclusive OR of all bytes between the STX and ETB/ETX characters. The LRC is discussed in more detail later.)

The following diagram shows the communication sequence.



Calculating the Header Checksum

Longitudinal Redundancy Check

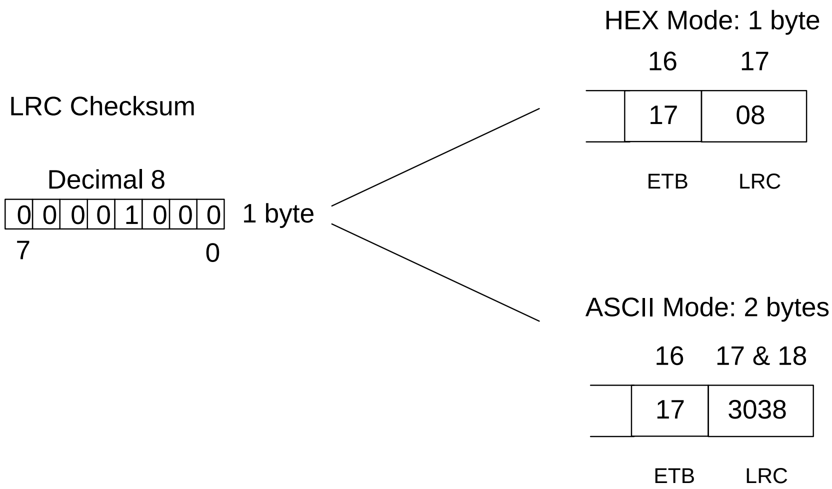
The LRC yields a checksum which is used to verify the communications are being received without errors. For a header, this checksum is calculated by taking the exclusive OR of all bytes between the Start of Header and End of Transmission (ETB). For a data block, the checksum is the exclusive OR of all bytes between the STX and ETB/ETX characters. To take the exclusive OR, just convert the HEX values to binary and then examine the bits. For each bit position, an even number of ‘1’s results in a checksum value of 0. An odd number of ‘1’s results in a checksum value of 1. Here’s an LRC calculation example based on the values used in the discussion of the header.

HEX ASCII Coded Representation

Byte:	1	2, 3	4	5	6, 7	8, 9	10, 11	12, 13	14, 15	16	17(18)	
	01	3034	30	31	3431	3031	3031	3930	3031	17	08	HEX
											3038	ASCII
	SOH	Target Slave	Read or Write	Data Type	Data Address MSB	Data Address LSB	Number Complete Blocks	Bytes in Last Block	Master ID	ETB	LRC	

Byte		HEX ASCII Value	B7 128	B6 64	B5 32	B4 16	B3 8	B2 4	B1 2	B0 1
Target slave:	(byte 2)	30	0	0	1	1	0	0	0	0
	(byte 3)	34	0	0	1	1	0	1	0	0
Read or write:	(byte 4)	30	0	0	1	1	0	0	0	0
Data Type:	(byte 5)	31	0	0	1	1	0	0	0	1
Data address MSB:	(byte 6)	34	0	0	1	1	0	1	0	0
	(byte 7)	31	0	0	1	1	0	0	0	1
Data address LSB:	(byte 8)	30	0	0	1	1	0	0	0	0
	(byte 9)	31	0	0	1	1	0	0	0	1
Complete blocks:	(byte 10)	30	0	0	1	1	0	0	0	0
	(byte 11)	31	0	0	1	1	0	0	0	1
Bytes in last block:	(byte 12)	39	0	0	1	1	1	0	0	1
	(byte 13)	30	0	0	1	1	0	0	0	0
Master address:	(byte 14)	30	0	0	1	1	0	0	0	0
	(byte 15)	31	0	0	1	1	0	0	0	1
Total Number of “1s”			0	0	14	14	1	2	0	6
Even (E) or Odd (O)			E	E	E	E	O	E	E	E
Exclusive OR Results:			0	0	0	0	1	0	0	0
Hexadecimal Value			0				8			
HEX ASCII Code			30				38			

From the table the checksum value is decimal 8. The checksum is contained in byte 17 of the header, but the actual value that is included depends on which mode of operation you are using. In HEX mode, this would be HEX 08. If you're using ASCII mode, then the value would be 3038 and the LRC now requires two bytes (17 and 18). The following diagram shows the differences in a HEX or ASCII mode LRC.



Calculating the Data LRC

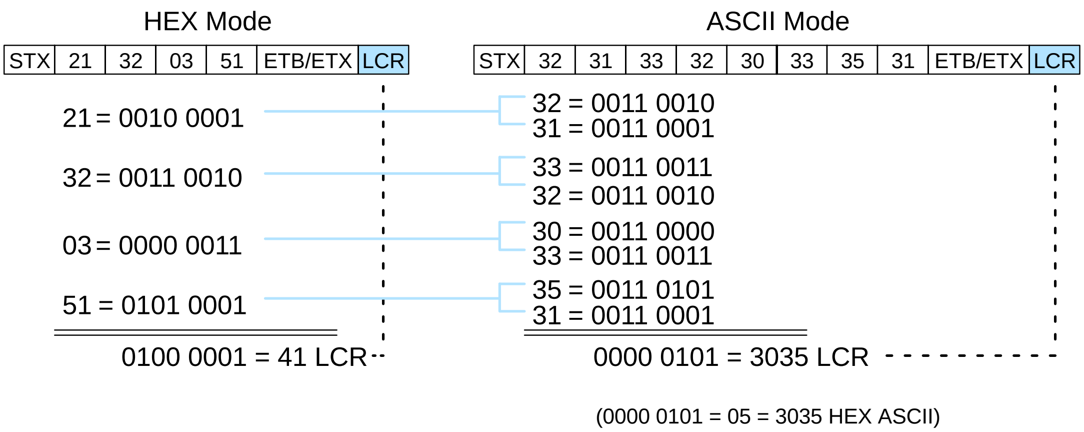
You should always calculate the LRC when either writing data to a slave or reading data from a slave. Note, during a read command slave stations will calculate their own LRC to be verified by the receiving device. However, you must calculate the value for LRC verification.

The LRC is included in the header and data transmissions. For a data block, the checksum is the exclusive OR of all bytes between the STX and ETB/ETX characters. (The example program in Appendix C shows how to do this in a BASIC program.)

NOTE: You only have to verify the checksum when you are creating your own *Direct*NET communications programs. If the master is a *Direct*LOGIC™ PLC with a communications interface, RLL instructions are used for the communication program and checksum verification is done automatically.

LRC Example for HEX and ASCII Transfers

This example shows how to calculate the LRC for the same data being transferred in either HEX or ASCII mode.

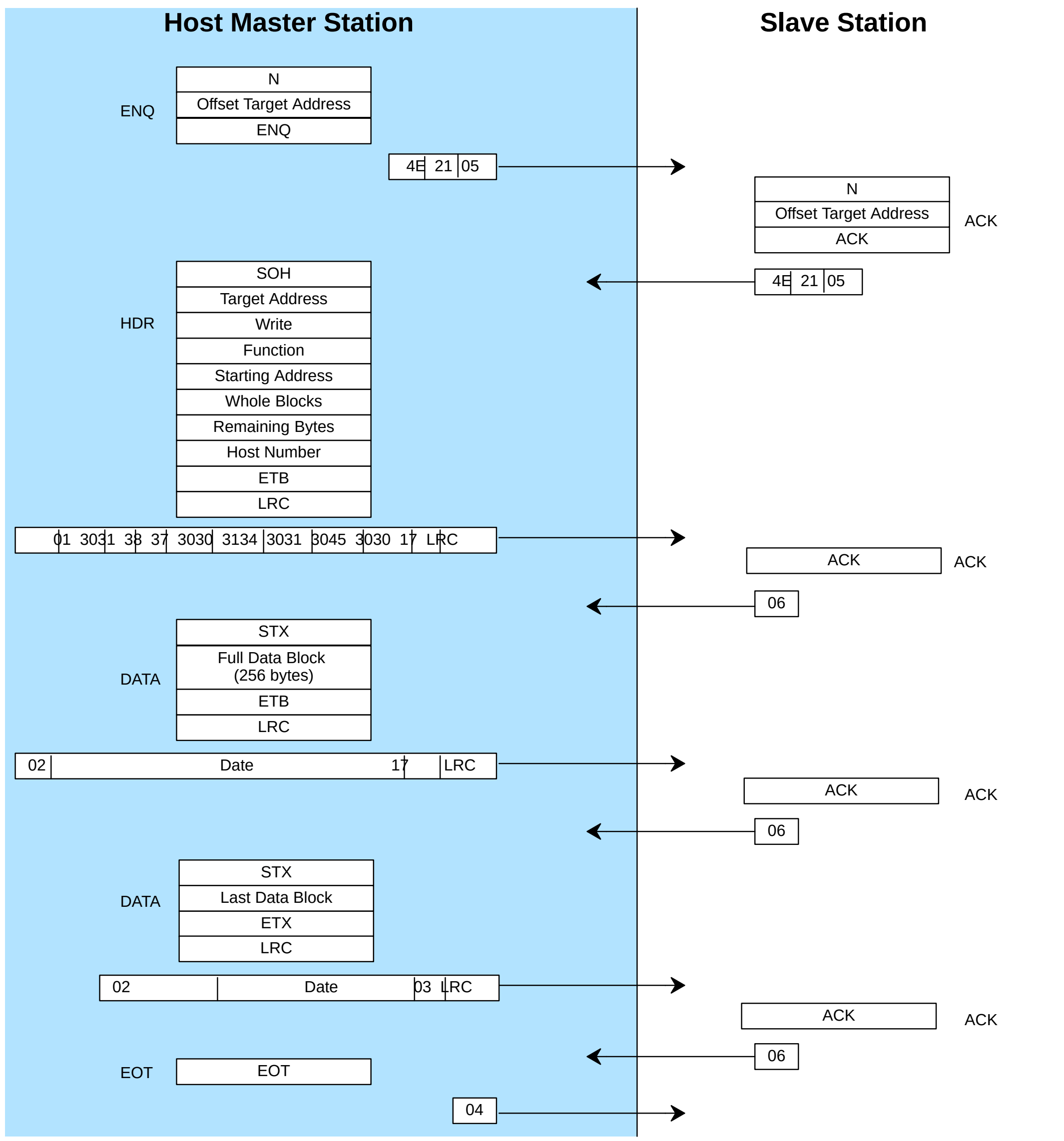


Ending the Request

End of Transmission – ETB

When the last data block has been transferred and acknowledged, the End of Transmission (EOT) character is sent. The master station must always end the communication by sending an EOT (HEX ASCII 04). The following diagram shows the EOT format.

Master to Slave Data Transfer Summary Sheet



Timing Considerations

Timeouts

The network communications generally operate very quickly and without problems. However, as with all things, problems can occur. Timeouts occur when either the master or slave does not receive a response to a communication within a certain period of time. There are two timeout possibilities.

- Slave timeout – this occurs when the slave does not respond within a specified time. When the slave times out, you must send an EOT from the host to terminate the communication
- Master timeout – this occurs when the slave station does not receive the complete communication from the master within a specified time. The slave station will send an EOT to signal that the communications have been terminated. The master must also send an EOT back to the slave to acknowledge the termination. (This allows the next communication to begin.)

The following tables provide the maximum times that ***Direct***LOGIC™ products will wait for a communication before entering a timeout condition.

DL405 Timeouts

Communication Segment	within (ms)
Master sends ENQ → Slave sends ACK	800
Master receives ACK → sends Header	800
Slave receives Header → sends ACK/NAK	2000
(Destination – write is slave, read is master) Master receives ACK → Data is transferred Destination receives Data → sends ACK/NAK Source receives ACK/NAK → sends EOT	 20000 20000 800

DL205 Timeouts

Communication Segment	within (ms)
Master sends ENQ → Slave sends ACK	800
Master receives ACK → sends Header	800
Slave receives Header → sends ACK/NAK	2000
(Destination – write is slave, read is PLC) Master receives ACK → Data is transferred Destination receives Data → sends ACK/NAK Source receives ACK/NAK → sends EOT	 20000 20000 800

DL305 Timeouts

Communication Segment	within (ms)
Master sends ENQ → Slave sends ACK	800
Master receives ACK → sends Header	800
Slave receives SOH → waits for Header LRC	
300 baud	2670
1200 baud	670
9600 baud	670
19.2K baud	670
Slave receives Header → sends ACK/NAK	2000
(Destination – write is slave, read is PLC)	
Master receives ACK → Data is transferred	20000
Destination receives Data → sends ACK/NAK	
300 baud	33340
1200 baud	8340
9600 baud	8340
19.2K baud	8340
Source receives ACK/NAK → sends EOT	800

Managing Timeouts

All communications finish with an EOT being sent from the master station. Even if the slave station sends an EOT to signal that it has aborted the communication, the master still must send an EOT to enable the slave to accept a new enquiry.

If you are reading information from a slave and the LRC calculated in your program does not match the slave station LRC, you do not have to abort the communication with an EOT. Instead, send a NAK which will signal the slave to re-transmit the data. If you send an EOT you must restart the entire request.

Communication Retries

The slave stations send NAKs to signal the master to try sending header or data packets again. The master can either re-transmit a maximum of three times, or, send an EOT to restart the request.

Delays

Each portion of the communication requires a delay to allow the PLC to process the information. If you send data without allowing for the delay, the communication may be ignored (which causes a timeout), or the parity check will fail (which causes a NAK response). The following table provides delay times for the ***Direct*LOGIC™** products.

Information to Process	Delay (ms)
ACK of data packets, headers	1
All other ACKs, headers, EOTs	1

NOTE: The communication interfaces have delay switches that increase this delay time. If those delays are selected, add the appropriate times to the figures shown here.

What part of the manual should you use next?

Start the Network

Once you've created the communications program, you can start the network. Chapter 7 provides information concerning network operation and troubleshooting. Appendix C provides an example of a hosted network.

7

Network Operation and Troubleshooting

In This Chapter. . . .

Starting the network
Troubleshooting

Starting the Network

PLC as Master Networks

If you're using a PLC as the master station, you must put the CPU in Run mode before the communications program can be executed. This is because the PLC master uses RLL instructions included in your application program to initiate the data transfer requests. There are two ways to place the CPUs in run mode.

- Turn the keyswitch to the RUN position.
- Turn the keyswitch to the TERM position and use a programming device to change the operating mode.

PLC Peer as Master Networks

PLC Peer master networks require both CPUs be in Run mode, since either station can initiate data transfer requests. Both stations contain the necessary RLL instructions.

Host as Master Networks

Host master networks require a ***Direct***NET communications program to manage the data requests. If you're using a host software package you should check the documentation that came with it for network startup information. If you created your own ***Direct***NET program, you'll have to execute the program file to get things started. Check the documentation for your particular programming language to determine the steps required to execute program files.

Slave Stations

Depending on your application, you may also have to place the slave PLCs in Run mode to obtain meaningful data. The DCM and DCU interfaces also have On-line switches that must be in the On-line position before communications can begin. CPUs with built-in ports do not necessarily have to be in Run mode, but again the data may not be current.

Troubleshooting

First Place to Look

If the network does not seem to be working correctly, check the following items.

1. Cable and connections. Incorrectly wired cables and loose connectors cause the majority of problems. Verify you've selected the proper cable configuration and check to see the cable is wired correctly.
2. Switch settings. Make sure you've set the slave stations to match the communication parameters required by the master station (DCM, operator interface or host computer).
3. Incorrect protocol. Make sure your network stations are all set for the same protocol. ***Direct***NET networks can use the ***Direct***NET or HostLink/CCM2 protocol selections.
4. Communications program. Check the communications program for errors.

Try an Example Program

Sometimes it is helpful to have an example you can try. AppendicesA,B, andC provide examples of the three network configurations. You can quickly and easily build a small network to make sure you are following all the appropriate steps. These are especially helpful if this is your first ***Direct***NET application.

CPUs with Built-in *Direct*NET Ports

Most problems that occur with the CPUs are related to communication settings. Check the switch settings, and/or use a programming device to check the station address.

Host Masters

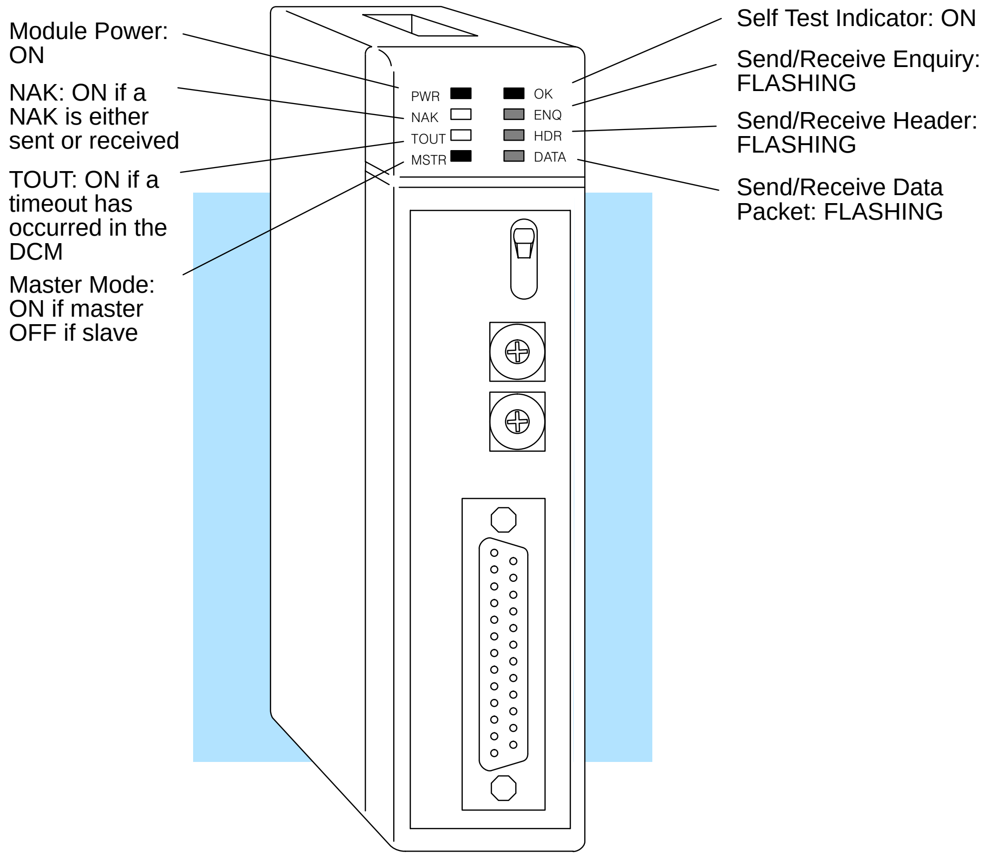
DirectNET programs must manage all aspects of network communication including timing considerations. Chapter 6 provides information on network timing issues that can affect network performance.

Communication Interface Indicators

The communications interfaces, DCMs and DCUs, have indicators that are specifically for communication status. The following paragraphs show these indicators and describe their meanings.

DL405 DCM Indicators

Check the DCM indicators to verify the DCM is operating correctly. The following diagram shows the proper indicator conditions.

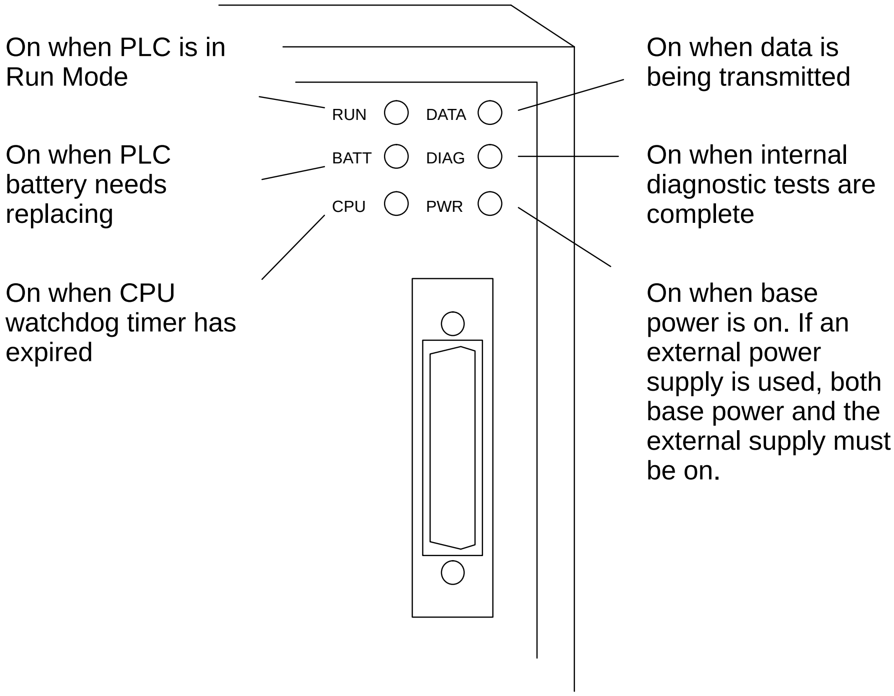


The following table provides additional troubleshooting details for the DL405 DCM.

Indicator Status	Possible Cause	Corrective Action
PWR or OK off	PLC power is disconnected DCM is defective	Check the PLC source power. Replace the DCM.
MSTR off (and DCM is in a master station)	Switch setting is incorrect	Remove power from the PLC, remove the DCM and check switch positions1 and 2 on SW5.
ENQ indicator does not come on when communications program is executed	The PLC master station is not in Run mode Online / Offline switch is in the Offline position Communications program is not correct	Place the PLC in Run mode. Set the switch to Online. Check the communications program. Verify the address, amount of data, and data type are correct.
ENQ stays on, but NAK, TOUT, or HDR indicators do not come on at all	Communication timeout is disabled RTS and CTS signals are not looped back on the master station end of the cable	Remove power from the PLC, remove the DCM, and check switch position 3 on SW5. Remove master station connector, ensure that RTS and CTS are connected according to the cable diagram.
ENQ comes on and TOUT indicator flashes	RLL communications program is not correct Modes are different Communication cable	Check the communications program. Verify the address is correct. Set baud rate, parity, and mode (HEX/ASCII) to match the master station. Verify the cable is wired according to the cable pinouts.
ENQ indicator comes on and NAK indicator flashes (slave responds, but the data is incorrect)	Modes are different Communication cable	Set baud rate, parity, and mode (HEX/ASCII) to match the master station. Make sure the + and – connections are correct (RS422). Check pin 7 (GND) if you're using RS232C.
ENQ and HDR indicators come on and the NAK indicator flashes	Communications program is not correct Modes are different	Check the amount of data being transferred. You must use the correct byte boundaries for the data type being used. Set baud rate, parity, and mode (HEX/ASCII) to match the master station.
DATA indicator is on, but the NAK indicator comes on intermittently	Electrical noise	Make sure the system has good earth grounds. Only one end of the cable shield should be grounded. If you're using RS232C, try using RS422 .

DL305 DCU Indicators

Check the DCU indicators to verify the DCU is operating correctly. The following diagram shows the proper indicator conditions.



The following table provides additional troubleshooting details for the DL305 DCU.

Indicator Status	Possible Cause	Corrective Action
PWR off	PLC power is disconnected	Check the PLC source power.
	DCU is not connected to the CPU properly	Make sure the DCU is securely fastened to the CPU and that no connector pins are bent.
	DCU external power source (if used) is not connected	Check the external power source.
	DCU is defective	Replace the DCU
DIAG off	DCU is defective	Replace the DCU
DATA does not flash during communications	Loose or incorrectly wired cable	Check the cable connections and pinouts.
	Online / Offline switch is in the Offline position	Set the switch to Online.
	Communications program is not correct	Check the master communications program. Verify the address, amount of data, and data type are correct. (See Appendices D, E, and F for address references.)

A

PLC Master / Slave Example

In This Chapter. . . .

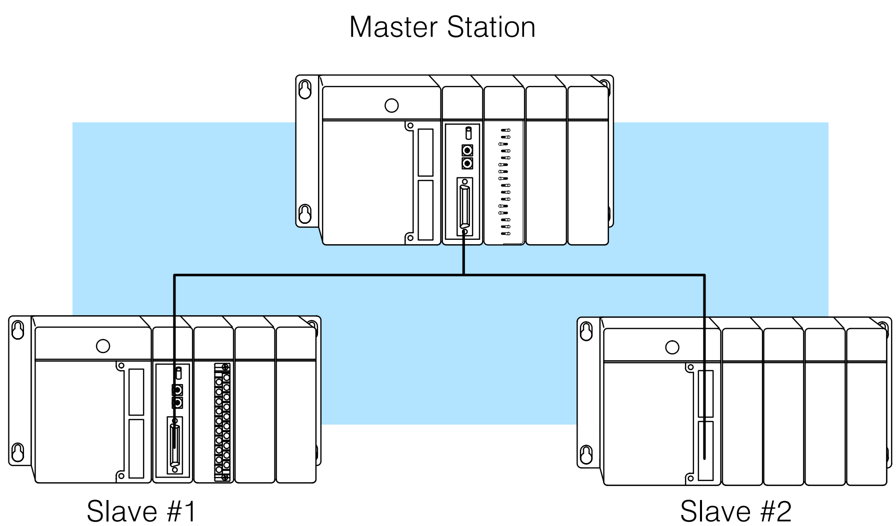
The Example

This chapter provides an example of a PLC master / slave network and is designed for the experienced user. This chapter does not provide detailed descriptions of network concepts and communications parameters. If you're unfamiliar with networking concepts, or if you want to know more detailed information about ***Direct***NET, you may want to read Chapters 2 – 5 before setting up the example network.

Example Equipment

A PLC master with a DCM interface connected to one or more PLC or DCM slaves is the most popular type of network. The PLC is the network master and uses the DCM to initiate all communication requests. The DCM or PLC slave stations respond to the master station requests for data.

This chapter provides an example that allows you to quickly and easily set up a small master / slave network.



Master Station

The following equipment is needed in the master station.

- DL405 Base (4, 6, or 8 slot)
- DL430 or DL440 CPU
- D4–DCM Data Communications Module
- I/O modules, including at least one I/O simulator

Slave Station #1

The following equipment is needed in slave station #1.

- DL405 Base (4, 6, or 8 slot)
- DL430 or DL440 CPU
- I/O modules, including at least one output module

Slave Station #2

The following equipment is needed in slave station #2.

- DL405 Base (4, 6, or 8 slot)
- DL430 or DL440 CPU

Install the Equipment

Normally, you can install the modules in any manner. However, we'd like to keep our examples consistent between the different configurations. Install the equipment in the following order.

WARNING: To minimize the risk of electrical shock, personal injury, or equipment damage, always disconnect the system power before installing or removing any system component.

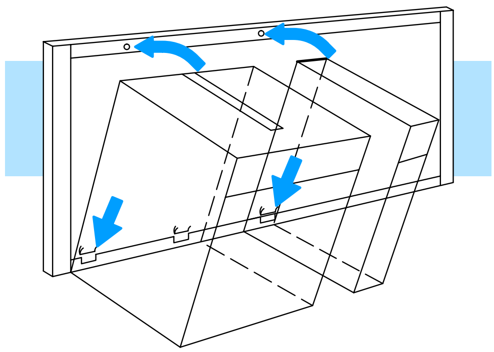
1. Install the CPU in the far left side of the base in the position marked "CPU/Power Supply". When inserting components into the base, tilt the component slightly forward sliding the tab on the bottom of the component into the slot in the base. Push the top of the component into the base until it is seated firmly, then tighten the securing screw at the top of the module/unit.

Each unit has a plastic tab at the bottom and a screw at the top.

With the unit tilted slightly forward, hook the module's plastic tab on the base.

Gently push the top of the unit back until it is firmly installed in the base.

Secure the unit to the base by tightening the top screw.



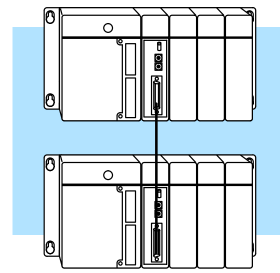
You can connect the power wiring now (if it's not already connected), but don't connect the source power yet.

2. For the master station and slave stations #1 and #2, reserve slot 0 for the DCM. (Slot 0 is the slot next to the CPU.) Set the DCMs aside for now. (You need to set the dipswitches and station addresses first.)
3. Install the I/O Simulator in slot 1, next to the DCM slot.

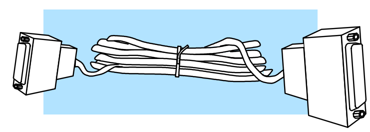
Remember these Four Steps!

Use these steps to build your example network. The remainder of this chapter provides detailed explanations and examples of these steps.

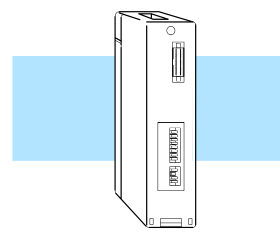
Master / Slave



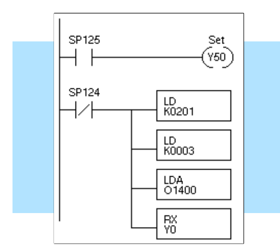
1. Design the network by:
   - Selecting the configuration (this is a master / slave configuration)
   - Building the communication cables



2. Select the communication settings by:
   - Setting the master switches
   - Setting the slave switches

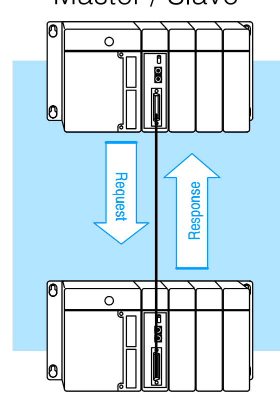


3. Write the communication control program.
   - RLL program is used with PLC master



4. Start the network operation.

Master / Slave



Step 1: Design the Network

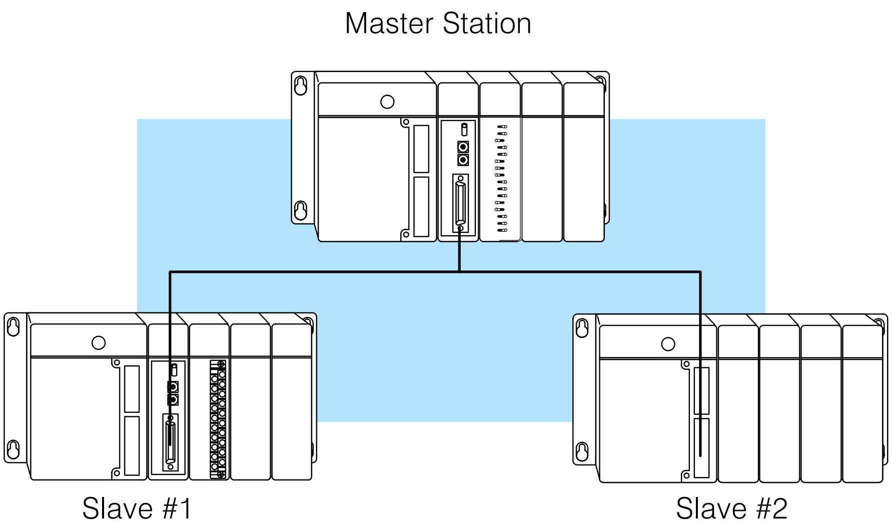
The Example Configuration

In this chapter we'll use the following example configuration to create a simple master slave network.

WARNING: These examples are for illustration purposes only and are not intended for use in actual applications. This is because there may be many aspects of your system safety precautions that are not addressed in the examples. If you use these examples in actual applications, you are increasing the risk of personal injury and/or property damage.

Goal:

1. Write a bit pattern from an input module (X0 – X7) to Slave #1
2. Read 1 V memory location from Slave #2.



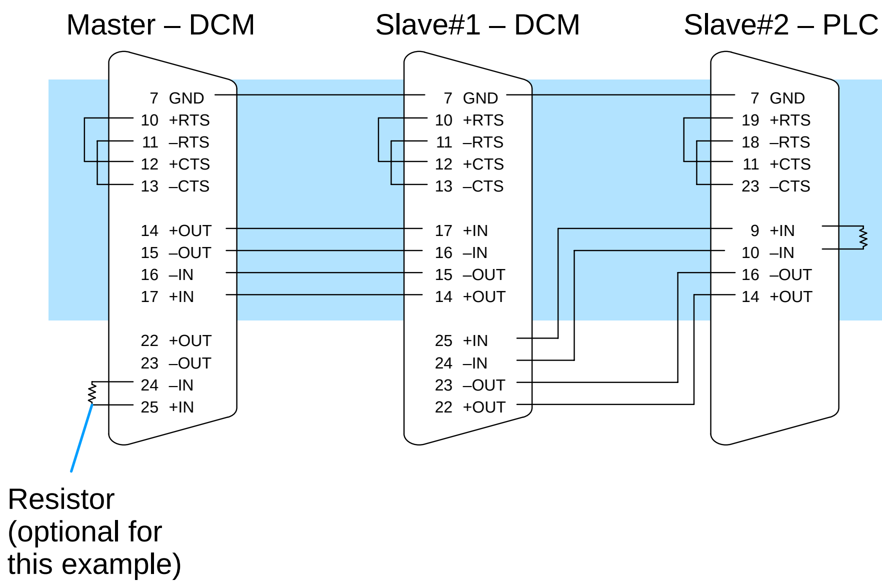
The Example Cable

In our example configuration we have:

- A PLC with a DCM as the master station
- A PLC with a DCM as slave station #1.
- A DL440 PLC with a built-in ***Direct***NET port as slave station #2

We'll have to use the pinout diagrams for those slaves. Also, since we have more than one slave station we'll use the RS422 multi-drop cable.

DCM to DCM to PLC port cable pinout

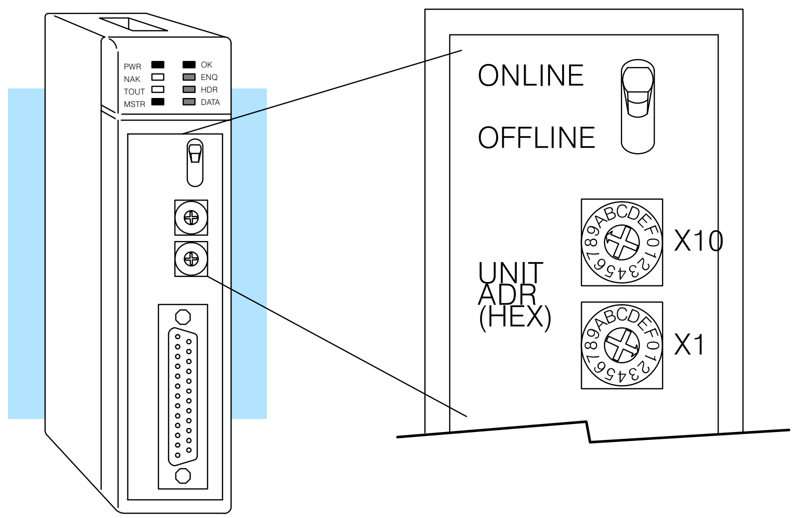


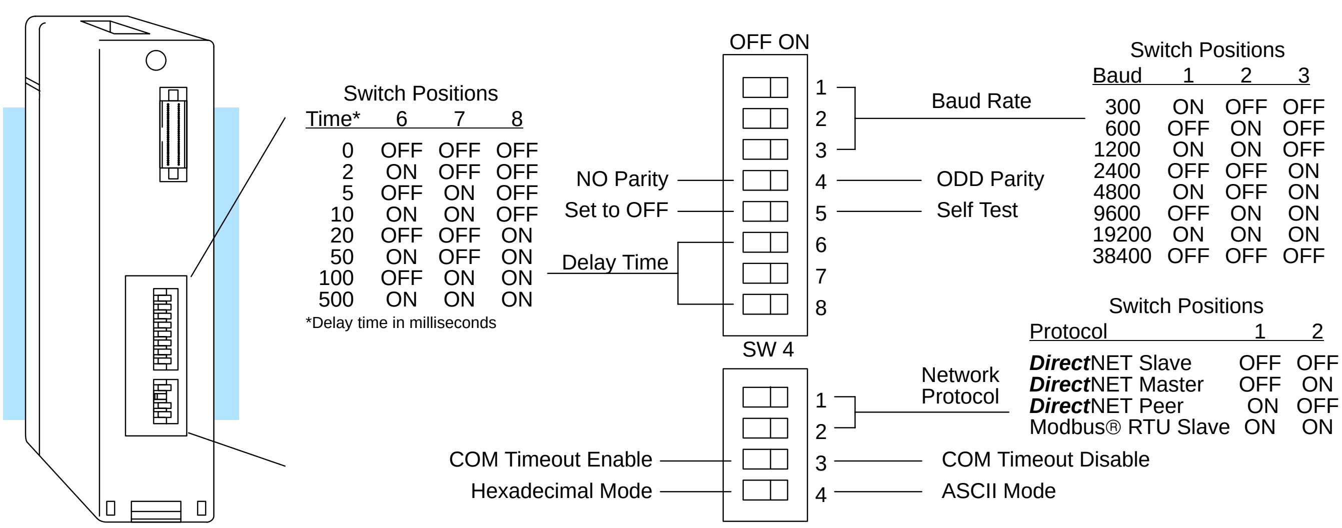
Step 2: Select the Communication Settings

Set the DCM Switches for the Master Station

The next step is to set the master station DCM communication parameters. We will use the following settings in our example.

- On-line / Off-line — On-line position
- Address — 0 (hexadecimal)
- Baud Rate — 19.2K
- Parity — None
- Response Delay Time — 0
- Network Protocol — ***Direct***NET Master
- Mode — HEX





DCM Rear View

Set the DCM Switches for Slave #1

Slave station #1 is a DL405 PLC with a DCM as the network interface. Set the DCM address to 1. Set the communication parameters to match the master station.

- Address — 1 (hexadecimal)
- On-line / Off-line — On-line position
- Baud Rate — 19.2K
- Parity — None
- Response Delay Time — 0
- Network Protocol — ***Direct***NET Slave
- Mode — HEX

Install the DCM and Connect the Cables

Install the DCMs in the slots next to the CPU for the master station and slave station #2. Connect the communication cables to the appropriate units.

NOTE: Remember that the pinout diagrams are different for the DCMs and the bottom CPU port. Make sure the cables are connected to the proper device.

Set the PLC Switches for the Slave #2

Slave station #2 is a DL405 PLC. If you look at the back of the DL405 CPU you will notice a small bank of switches. Switches 2, 3, and 4 are used to set the communication parameters for the bottom communication port.

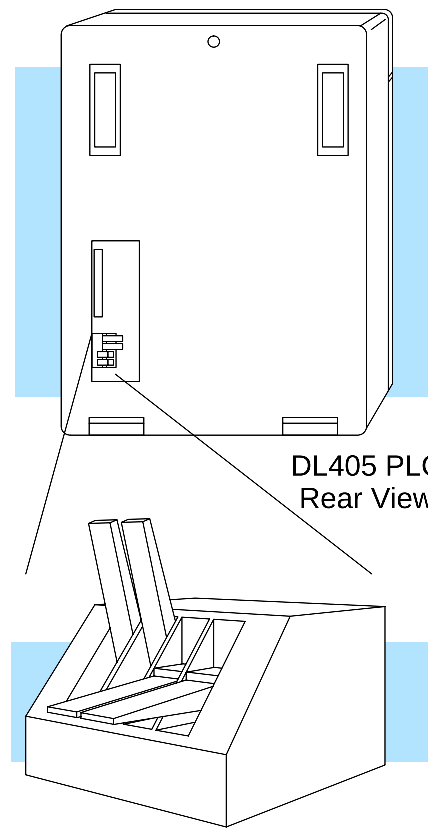
Switch 2 — This switch selects the CPU slave address. If this switch is On, then an address of 1 is used. If the switch is Off, then you can use a programming device to set the address.

Switch 3 & 4 — These switches select the baud rate for the bottom port.

Baud	3	4
300	OFF	OFF
1200	OFF	ON
9600	ON	OFF
19200	ON	ON

Set switch 2 to OFF – Address 1.
Set switch 3 ON and 4 OFF – 9600 Baud

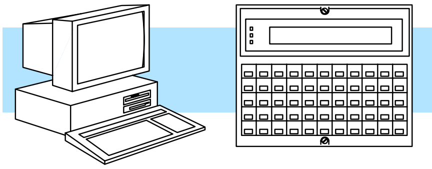
Switch 1 does not apply to the networking example. It is used to select the battery low indictor operation.



Set the Station Address for Slave #2

With ***Direct***SOFT, use AUX 56 from the Auxiliary functions menu to set the port parameters.

With the DL405 handheld programmer, use AUX 56 to set the port parameters. The following example shows how to use the handheld programmer to set the address.



DirectSOFT DL405 Handheld Programmer

NOTE: The PLC port address is set in decimal, not hexadecimal.

Select AUX 56

CLR CLR AUX 5 6 ENT ENT

AUX 5* CPU CFG
AUX 56 CPU N/W ADDRESS

Enter the Address (in decimal)

0 2 ENT

AUX 56 CPU N/W ADDRESS
N/W # 02

Set Address to 2 (decimal)

Select HEX or ASCII

To change the mode press .... →

then press enter. ENT

AUX 56 CPU N/W ADDRESS
HEX / ASCII

Select HEX mode for data transfer.

Select the Parity Option

To change the parity press .... →

then press enter. ENT

AUX 56 CPU N/W ADDRESS
NONE / ODD

Set Parity to none.

Clear the Display

To clear the display press.... CLR

OK

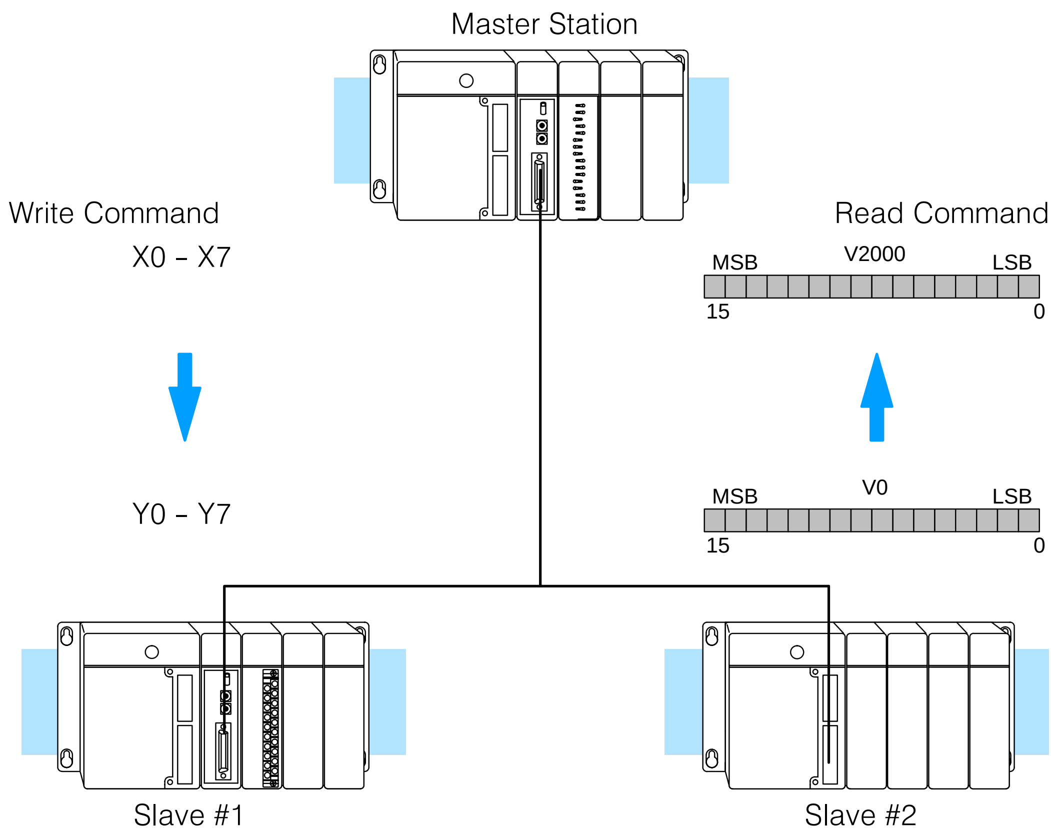
Step 3: Create the Communications Program.

Program Description

In the example network, we'll use 8 switches on the input simulator in the master system to set a bit pattern that will be written to an output module on the slave station #1. Also, we'll read the current value of a timer from slave station #2. Our example requires a program in both the master and slave stations.

Goal:

1. Write a bit pattern from an input module (X0 – X7) to Slave #1
2. Read one V-memory location (V0000 – two bytes) from Slave #2, store in V2000.



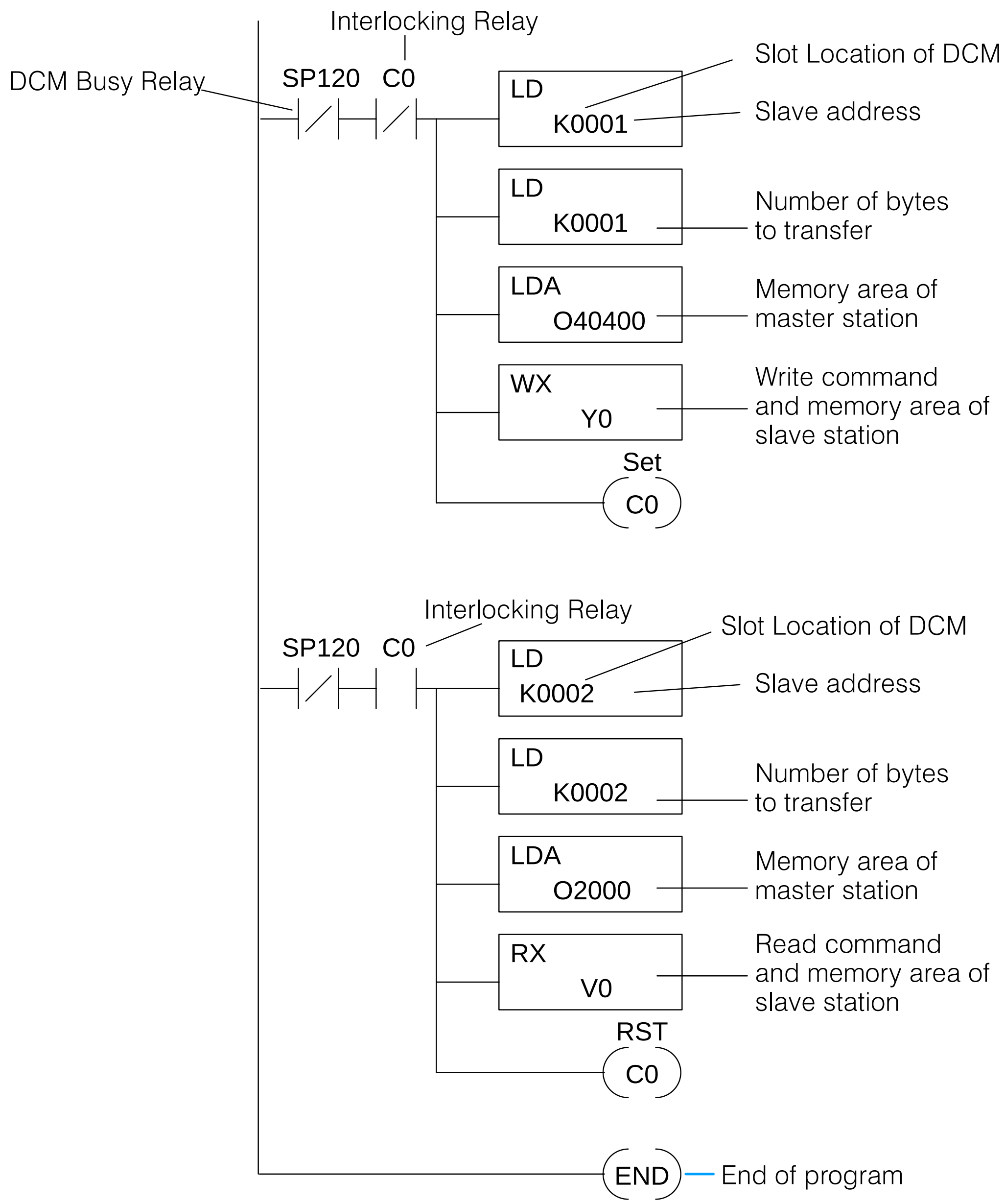
Connect the Programing Device

The example provides the instructions needed for the ***Direct***SOFT programming package and for the DL405 Handheld Programmer. In either case, you should connect the programming device to the top port on the DL405 CPU. We assume you understand how to use the ***Direct***SOFT Programming Package and/or the Handheld Programmer. If you aren't familiar with these, you should probably review those product manuals prior to trying to enter these programs.

The program examples on the following pages show both RLL and RLL*PLUS* examples. The RLL*PLUS* approach eliminates the need for the interlocking relays and is generally a much more straightforward programming method.

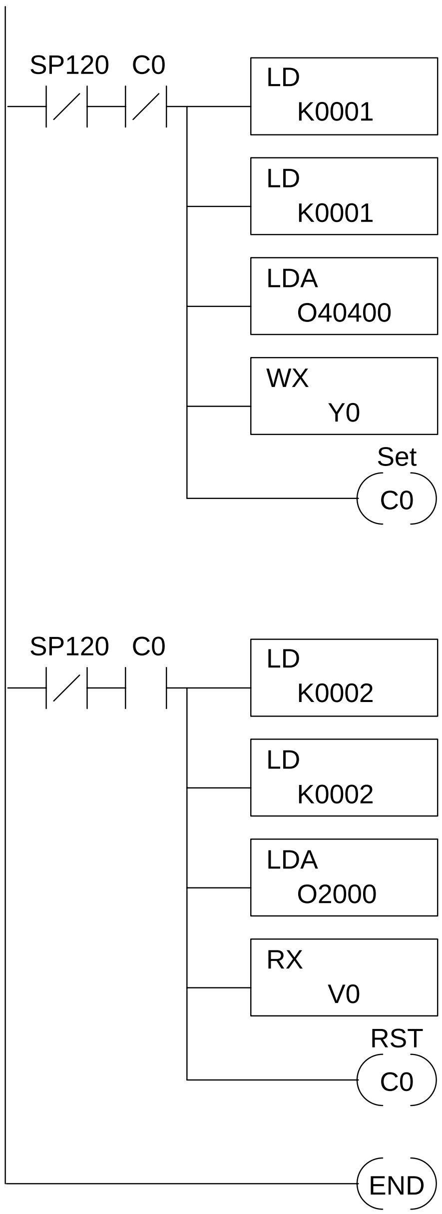
RLL Instructions

The following diagram shows the RLL instructions used in the communications program. Chapter 5 provides detailed descriptions of the instructions. You should always use the interlocking relays to ensure that the DCM has adequate time to finish a communication task.



Master Program RLL Example

***Direct*SOFT**

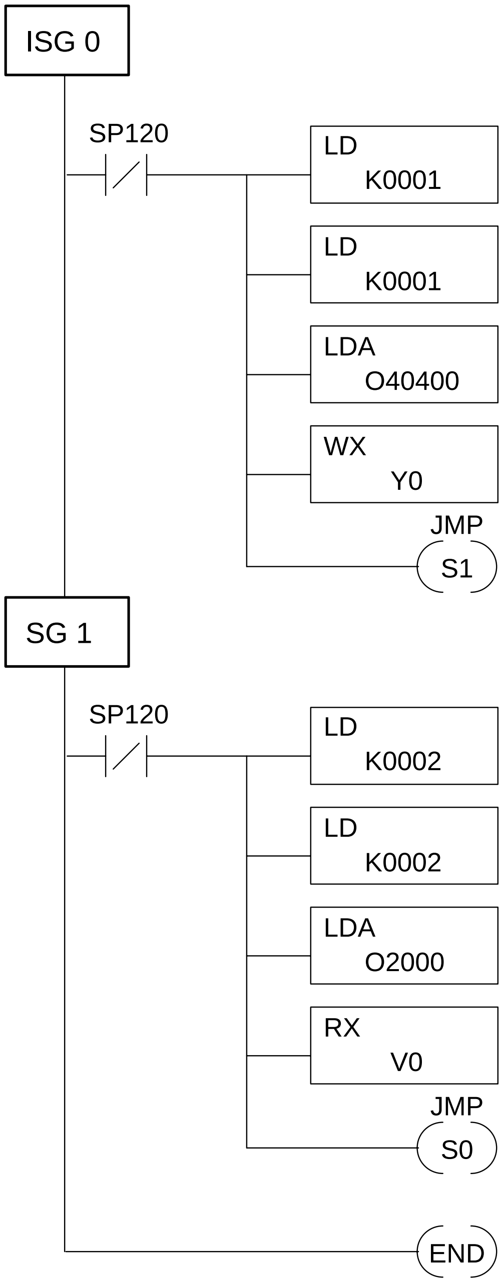


DL405 HP Mnemonics

```
$0
STRN SP120
$1
ANDN C0
$2
LD K0001
$3
LD K0001
$4
LDA O40400
$5
WX Y0
$6
SET C0
$8
STRN SP120
$9
AND C0
$10
LD K0002
$11
LD K0002
$12
LDA O2000
$13
RX V0
$15
RST C0
$16
END
```

Master Program RLL*PLUS* Example

***Direct*SOFT**



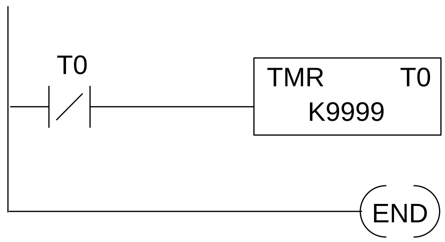
DL405 HP Mnemonics

$0
ISG S0
$1
STRN SP120
$2
LD K0001
$3
LD K0001
$4
LDA O40400
$5
WX Y0
$6
JMP S1
$8
SG S001
$9
STRN SP120
$10
LD K0002
$11
LD K0002
$12
LDA O2000
$13
RX V0
$15
JMP S0
$16
END

Slave Station #2 Program RLL Example

Our example requires a program in slave station #2. The slave station program is much easier because the master station program controls the communication. In the slave station, we're just using a self-resetting timer to provide a data value.

***Direct*SOFT**

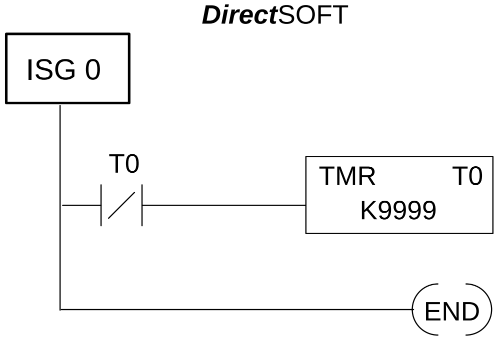


DL405 HP Mnemonics

$0
STRN T0
$1
TMR 0
$2
K9999
$3
END

Slave Station #2 Program RLL*PLUS* Example

***Direct*SOFT**



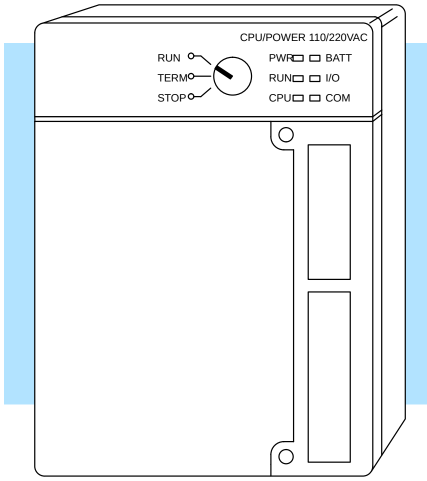
DL405 HP Mnemonics

$0
ISG S0
$1
STRN T0
$2
TMR 0
$3
K9999
$4
END

Step 4: Start the Network

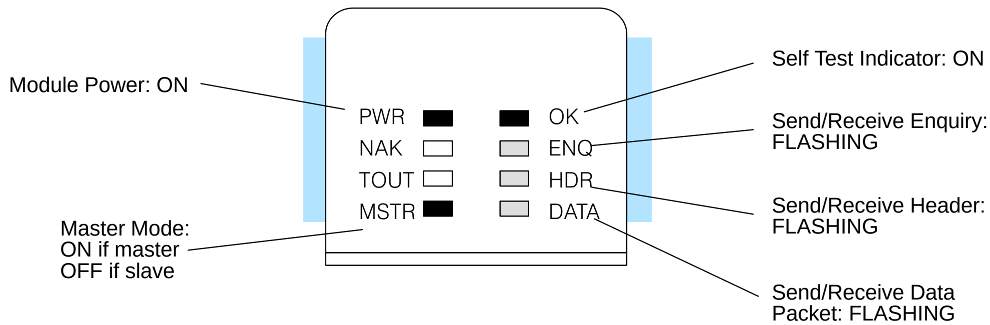
Switch the PLCs to Run Mode

Only the master station needs to be in Run mode to execute the communications program. However, for this example to work correctly all CPUs should be in Run mode. You can use the programming devices to place them in Run mode,or you just turn the keyswitch to the RUN position.



Verify the Network

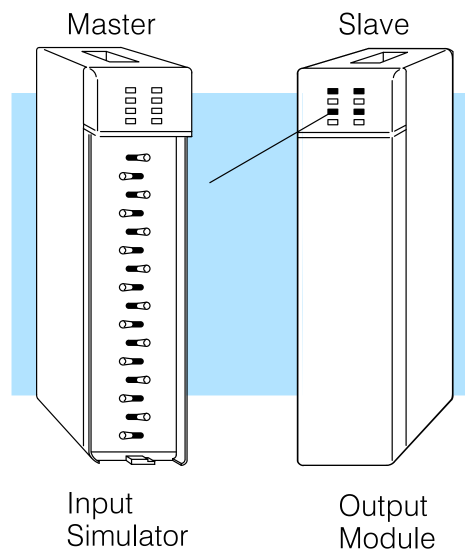
Check the DCM indicators to verify that the network is operating correctly. shows the proper indicator conditions. The ENQ, HDR, and DATA indicators should be flashing.



Verify the Write Command

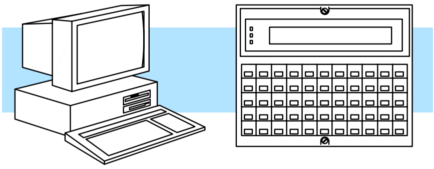
Now you can change the I/O simulator switch settings and verify the communications.

1. Set every other switch to the ON position
2. Look at the first slave station output module. The indicators should match the I/O simulator switch settings.
3. Change the I/O simulator switch settings at random and notice how the output module indicators change.



Verify the Read Command

You'll need to use the programming device to verify the read requests. Connect the programming device and complete the following steps.



***Direct*SOFT** DL405 Handheld Programmer

Monitor a V Location

CLR CLR V 0 WD ST ENT

ENT

```
      V    1   V     0
V MON 0000      XXXX
```

What should I do if it isn't working correctly?

Troubleshooting Steps

If the network does not seem to be working correctly, check the following items.

1. Cable and connections. Incorrectly wired cables and loose connectors cause the majority of problems. Verify that you've selected the proper cable configuration and check that the cable is wired correctly.
2. Dipswitch settings. Make sure you've set the switches to allow the same communication settings for both stations.
3. Communications program. Verify that the program has been entered as shown in the example.
4. If the network still doesn't work correctly go to Chapter 7, Network Operation and Troubleshooting, and use the troubleshooting charts.

Peer Master Example

B

In This Chapter. . . .

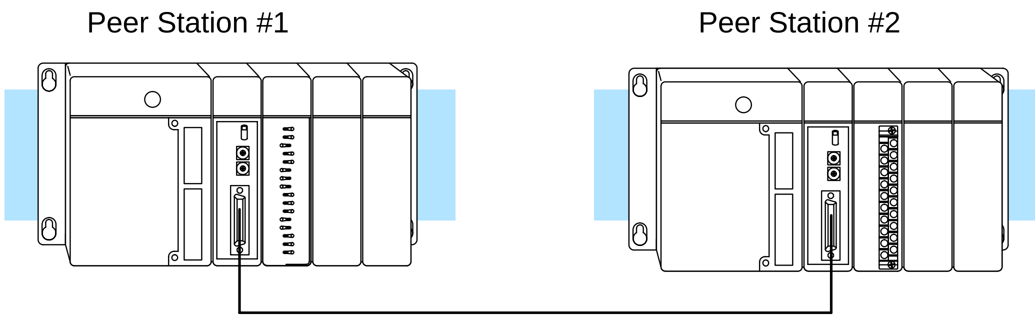
The Example

This chapter provides an example of a peer-to-peer network and is designed for the experienced user. This chapter does not provide detailed descriptions of network concepts and communications parameters. If you're unfamiliar with networking concepts, or if you want to know more detailed information about ***Direct***NET, you may want to read Chapters 2 – 5 before setting up the example network.

A DCM peer network utilizes two DCMs, with both configured as peer stations. You can only have two stations in this configuration. Either unit can initiate a communications request. (The DCMs automatically compensate for the possibility of data collisions.)

Example Equipment

This chapter provides an example that allows you to quickly and easily set up a peer network.



Peer Station #1

The following equipment is needed peer station #1.

- DL405 Base (4, 6, or 8 slot)
- DL430 or DL440 CPU
- D4–DCM Data Communications Module
- I/O modules, including at least one I/O simulator

Peer Station #2

The following equipment is needed in peer station #2.

- DL405 Base (4, 6, or 8 slot)
- DL430 or DL440 CPU
- D4–DCM Data Communications Module
- I/O modules, including at least one output module

Install the Equipment

Normally, you can install the modules in any manner. However, we'd like to keep our examples consistent between the different configurations. Install the equipment in the following order.

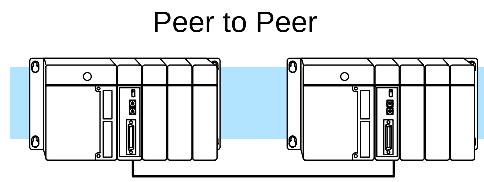
WARNING: To minimize the risk of electrical shock, personal injury, or equipment damage, always disconnect the system power before installing or removing any system component.

---

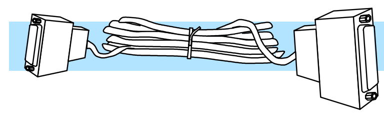
1. Install the CPU as described in the DL405 User Manual. You can connect the power wiring now (if it's not already connected), but don't connect the source power yet.
2. For both stations, reserve slot 0 for the DCM. (Slot 0 is the slot next to the CPU.) Set the DCMs aside for now. (You need to set the dipswitches and station addresses first.)
3. Install the I/O Simulator in slot 1 of peer station #1.
4. Install the output module in slot 1 of peer station #2.

Remember these Four Steps!

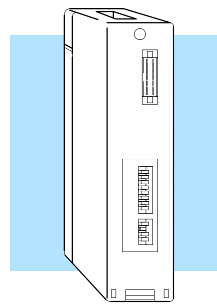
Use these steps to build your example network. The remainder of this chapter provides detailed explanations and examples of these steps.



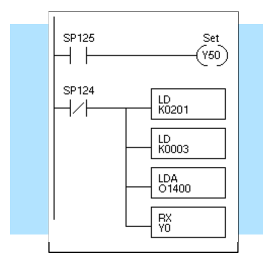
1. Design the network by:
   - Selecting the configuration (this is a Peer as Master configuration)
   - Building the communication cables.



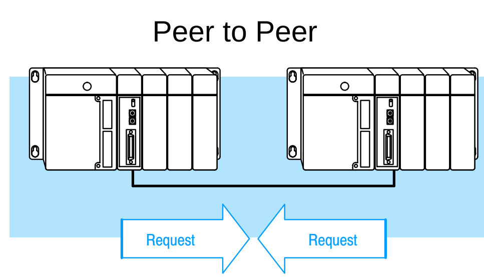
2. Select the communication settings by:
   - Setting the Peer Master switches



3. Write the communication control program.
   - RLL program is used with PLC master



4. Start the network operation.



Step 1: Design the Network

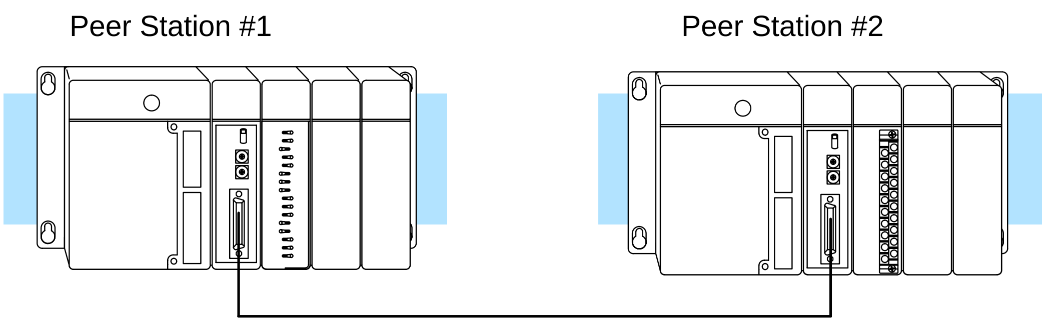
The Example Configuration

In this chapter we'll use the following example configuration to create a simple peer network.

WARNING: These examples are for illustration purposes only and are not intended for use in actual applications. This is because there may be many aspects of your system safety precautions that are not addressed in the examples. If you use these examples in actual applications, you are increasing the risk of personal injury and/or property damage.

Goal:

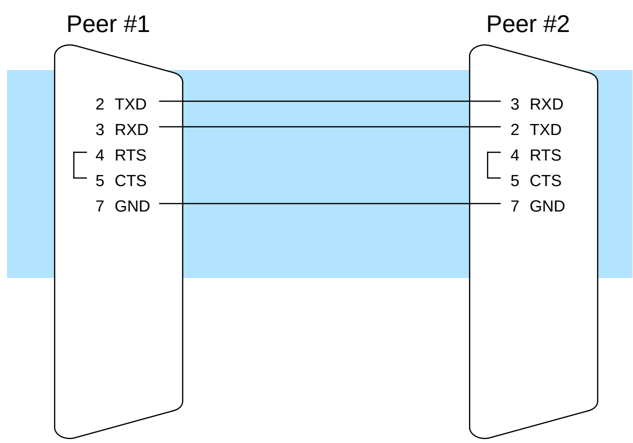
1. Peer #1 – write a bit pattern from an input module (X0 – X7) to station #2.
2. Peer #2 – read 1 V memory location from station #1.



The Example Cable

In our example configuration we have a DCM in each station. Since we only have two stations (point-to-point) we can use RS232C communications.

RS232C DCM to DCM Cable

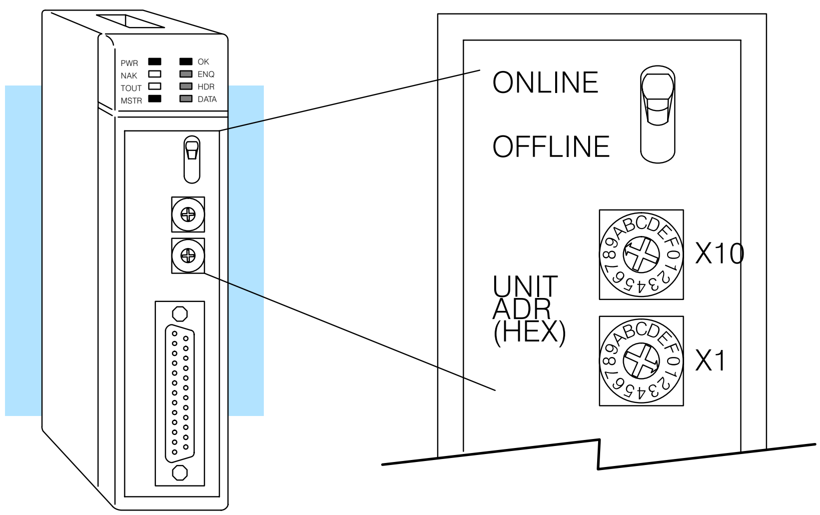


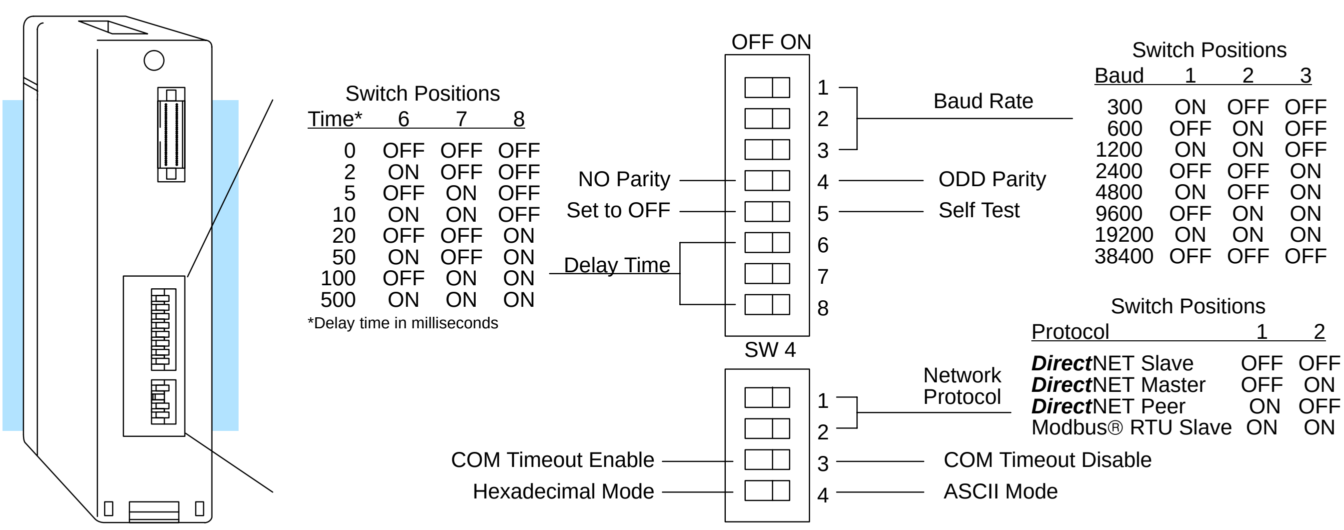
Step 2: Select the Communication Settings

Set the DCM Switches for Peer #1

The next step is to set the communications parameters for the DCM in peer station #1. We will use the following settings in our example.

- On-line / Off-line — On-line position
- Address — 1 (hexadecimal)
- Baud Rate — 38.4K
- Parity — None
- Response Delay Time — 0
- Network Protocol — ***Direct***NET Peer
- Mode — HEX





DCM Rear View

Set the DCM Switches for Peer #2

The next step is to set the communications parameters for the DCM in peer station #2. Notice the settings are exactly the same, with the exception of the address.

- On-line / Off-line — On-line position
- Address — 2 (hexadecimal)
- Baud Rate — 38.4K
- Parity — None
- Response Delay Time — 0
- Network Protocol — ***Direct***NET Peer
- Mode — HEX

Install the DCMs and Connect the Cables

Install the DCMs in the slots next to the CPU for both peer stations and connect the communication cable.

Step 3: Create the Communications Programs

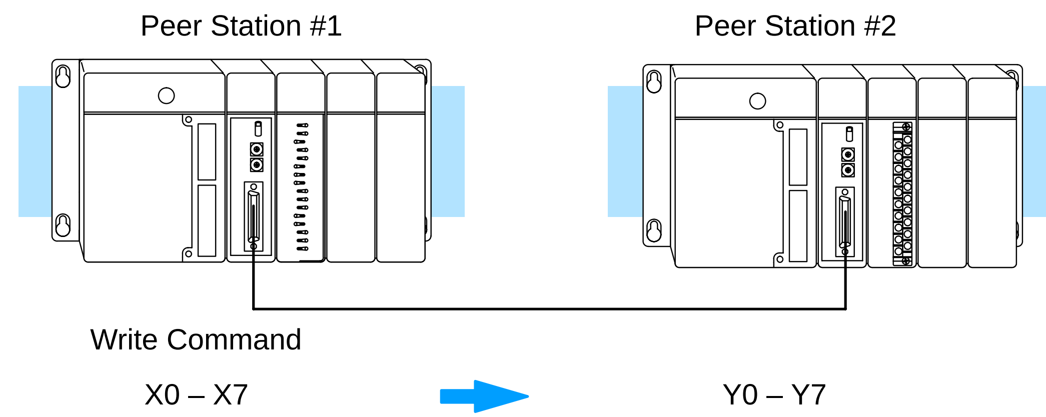
Two Programs are Required

With peer networks, you need a communications program in both stations. Each program contains the necessary instructions to initiate the data requests.

Peer Station #1 Program Description

For peer station #1, we'll use 8 switches on the input simulator to set a bit pattern that will be written to an output module on peer station #2.

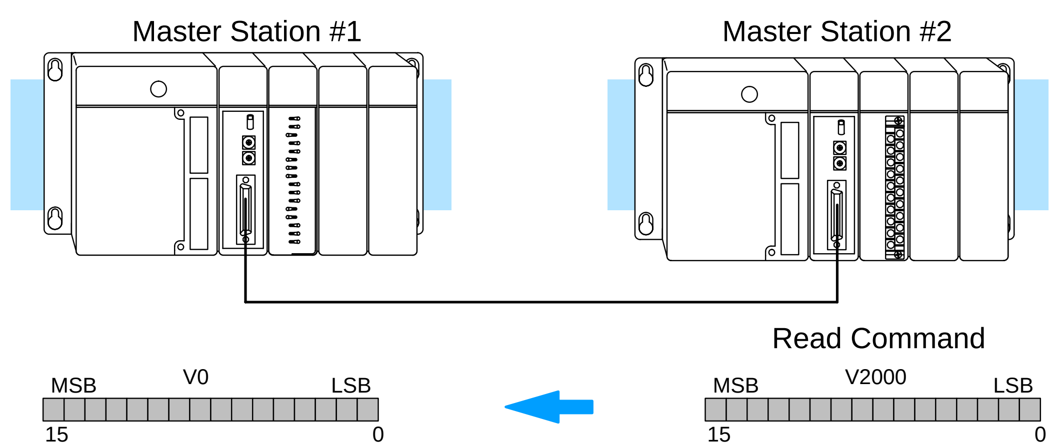
Goal: Write a bit pattern from an input module (X0 – X7) to peer station #2.



Peer Station #2 Program Description

For peer station #2, we want to read a V memory location from peer station #1.

Goal: Read 1 V memory location from peer station #1.



Connect the Programing Device

The example provides the instructions needed for the ***Direct***SOFT programming package and for the DL405 Handheld Programmer. In either case, you should connect the programming device to the top port on the DL405 CPU. We assume you understand how to use the ***Direct***SOFT Programming Package and/or the Handheld Programmer. If you aren't familiar with these, you should probably review those product manuals prior to trying to enter these programs.

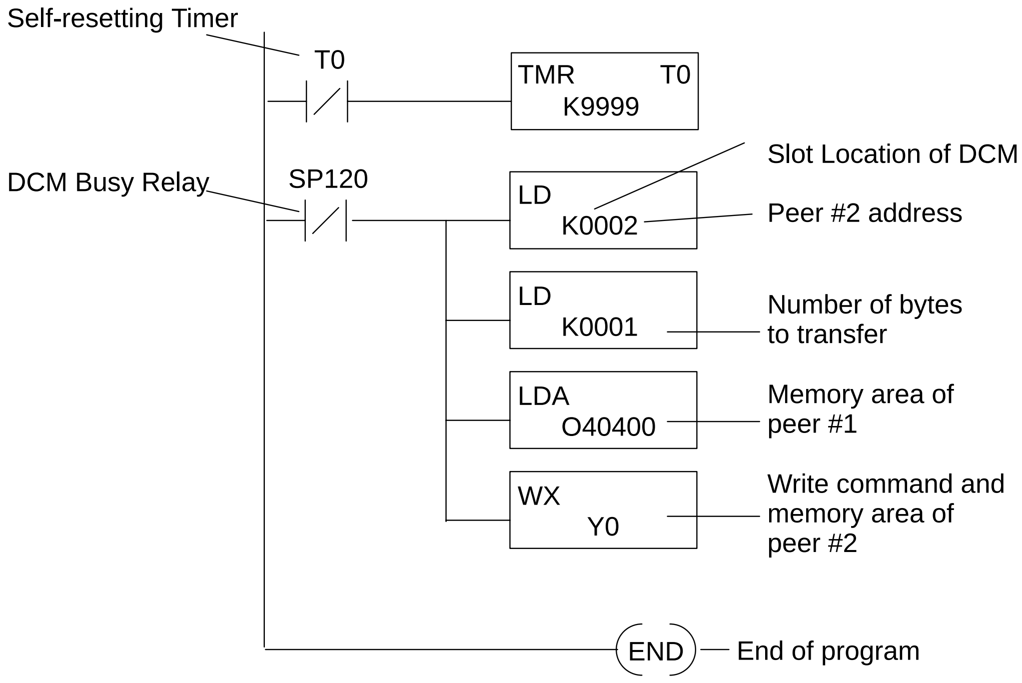
The program examples on the following pages show both RLL and RLL*PLUS* examples. The RLL*PLUS* approach eliminates the need for the interlocking relays and is generally a much more straightforward programming method.

Peer Station #1 RLL Instructions

The following diagram shows the RLL instructions used in the communications program for peer station #1. Chapter 5 provides detailed descriptions of the instructions. Since we also plan to use peer station #2 to read a V memory location from peer station #1, we're using a self-resetting timer to provide a data value. This will make it easier to verify our example is working correctly.

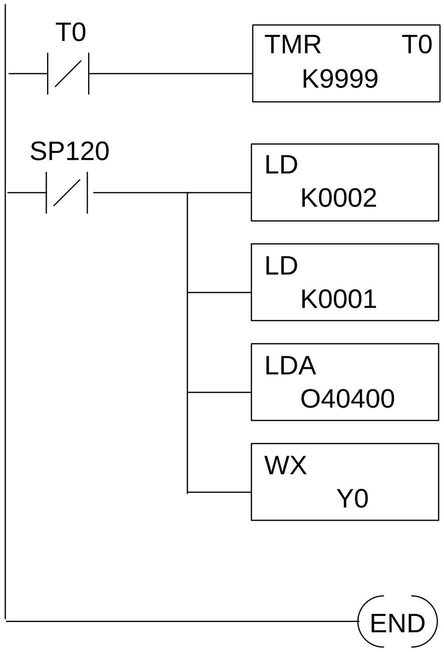
NOTE: This example does not have multiple communications requests in each station. If you need multiple requests in your application, you should use the interlocking relays to ensure the DCM has adequate time to finish a communication task. See Chapter 4 for more detailed information.

Goal: Write a bit pattern from an input module (X0 – X7) to peer station #2.



Peer Station #1 RLL Example

***Direct*SOFT**



DL405 HP Mnemonics

$0
STRN T0
$1
TMR 0
$2
K9999
$3
STRN SP120
$4
LD K0002
$5
LD K0001
$6
LDA O40400
$7
WX Y0
$8
END

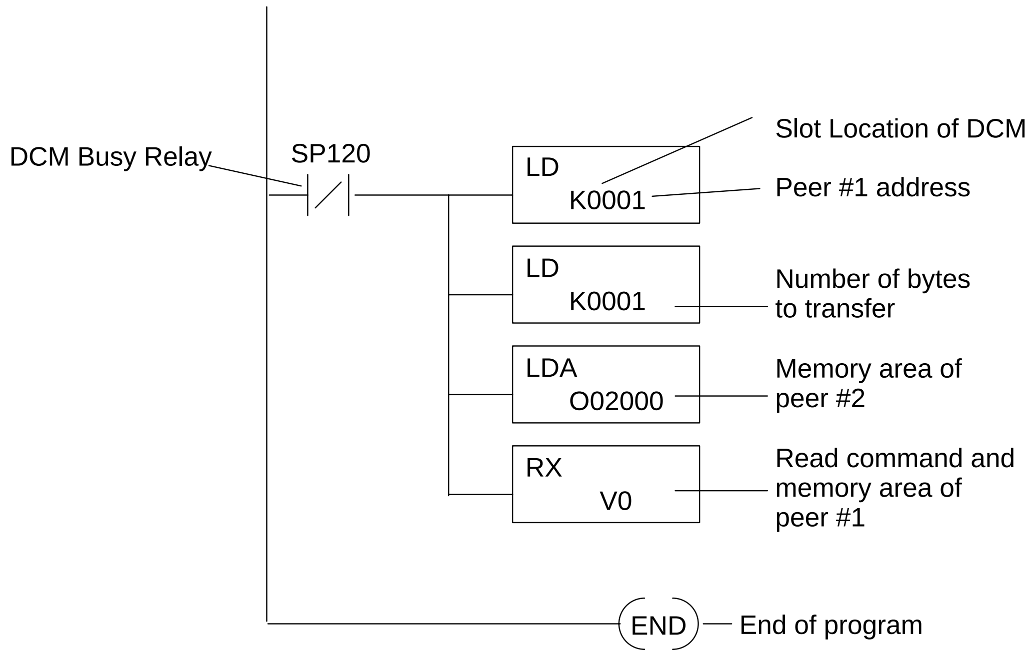
NOTE: To create this program in RLL*PLUS*, just add an initial stage (ISG 0) to the beginning of the program.

Peer Station #2 RLL Instructions

The following diagram shows the RLL instructions used in the communications program for peer station #2. Chapter 5 provides detailed descriptions of the instructions.

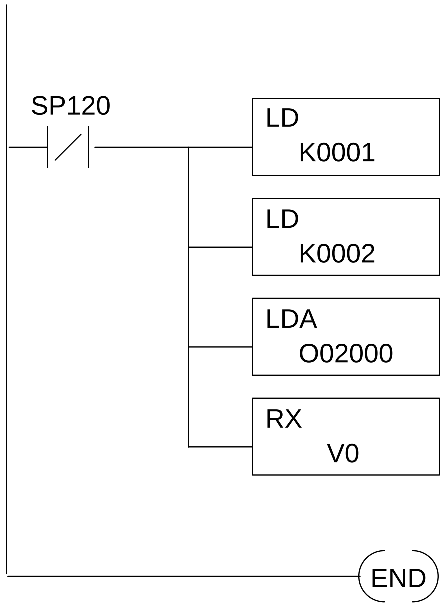
NOTE: This example does not have multiple communications requests in each station. If you need multiple requests in your application, you should use the interlocking relays to ensure the DCM has adequate time to finish a communication task. See Chapter 4 for more detailed information.

Goal: Read 1 V memory location from peer station #1.



Peer Station #2 RLL Example

***Direct*SOFT**



DL405 HP Mnemonics

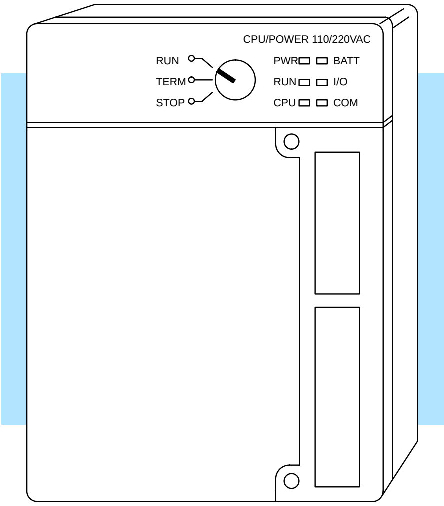
$0
STRN SP120
$1
LD K0001
$2
LD K0002
$3
LDA O2000
$4
RX V0
$5
END

NOTE: To create this program in RLL*PLUS*, just add an initial stage (ISG 0) to the beginning of the program.

Step 4: Start the Network

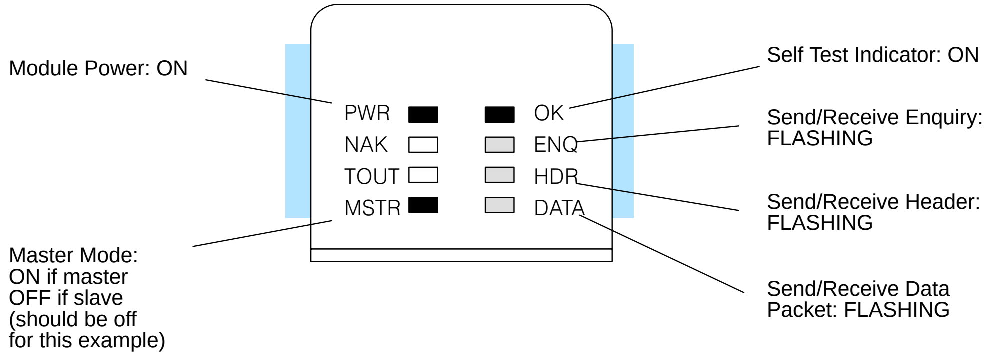
Switch the PLCs to Run Mode

For this example to work correctly both CPUs should be in Run mode. You can use the programming devices to place them in Run mode,or you can just turn the keyswitch to the RUN position.



Verify the Network

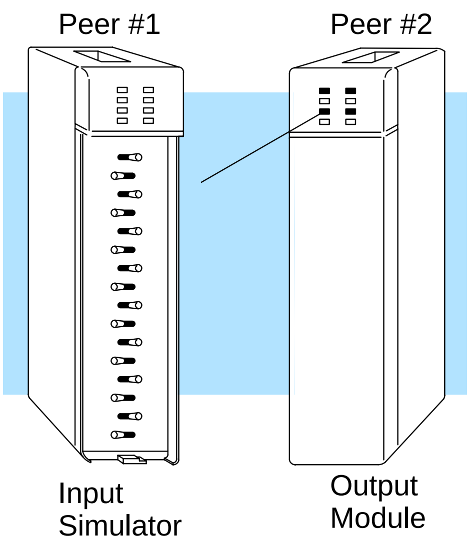
Check the DCM indicators to verify the network is operating correctly. shows the proper indicator conditions. The ENQ, HDR, and DATA indicators should be flashing.



Verify the Write Command

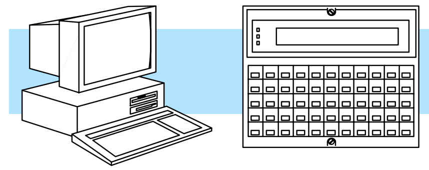
Now you can change the I/O simulator switch settings and verify the communications.

1. Set every other switch to the ON position
2. Look at the first slave station output module. The indicators should match the I/O simulator switch settings.
3. Change the I/O simulator switch settings at random and notice how the output module indicators change.



Verify the Read Command

You'll need to use the programming device to verify the read requests. Connect the programming device to peer station #2 and complete the following steps.



*Direct*SOFT

DL405 Handheld Programmer

	V 2001	V2000
V MON	0000	XXXX

Monitor a V Location

CLR CLR V 2 0 0 0 WD ST ENT

ENT

What should I do if it isn't working correctly?

Troubleshooting Steps

If the network does not seem to be working correctly, check the following items.

1. Cable and connections. Incorrectly wired cables and loose connectors cause the majority of problems. Verify you've selected the proper cable configuration and check that the cable is wired correctly.
2. Dipswitch settings. Make sure the switches are set the same for both stations.
3. Communications program. Verify the program has been entered as shown in the example.
4. If the network still doesn't work correctly go to Chapter 7, Network Operation and Troubleshooting, and use the troubleshooting charts.

C

Host Master Example

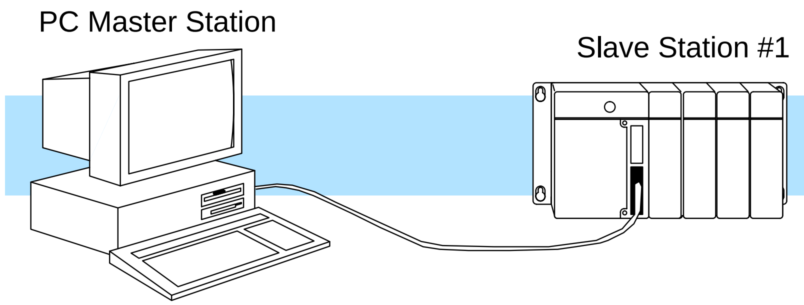
In This Chapter. . . .

The Example

A hosted network utilizes an intelligent device, such as a personal computer, to act as the network master station. Slave stations can be DCMs or ***Direct***LOGIC™ PLCs with built-in ***Direct***Net ports. The host device initiates all communication requests and the slaves respond to the requests for data. This example is not intended to show you how to build ***Direct***Link programs. You should review your programming language manuals and Chapter 6 of this manual before you attempt to build your own ***Direct***Link programs.

Example Equipment

This chapter provides an example that allows you to quickly and easily set up a hosted network.



Master Station

You need a personal computer for the master station.

Slave Station #1

The following equipment is needed slave station #1.

- DL405 Base (4, 6, or 8 slot)
- DL430 or DL440 CPU
- I/O modules, including at least one I/O simulator and one output module

Install the Equipment

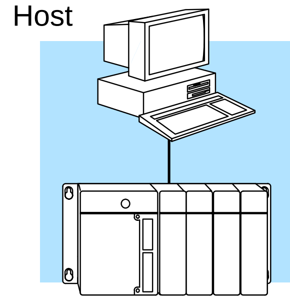
Normally, you can install the modules in any manner. However, we'd like to keep our examples consistent between the different configurations. Install the equipment in the following order.

WARNING: To minimize the risk of electrical shock, personal injury, or equipment damage, always disconnect the system power before installing or removing any system component.

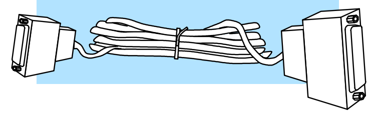
1. Install the CPUs as described in the DL405 User's Manual. You can connect the power wiring now (if it's not already connected), but don't connect the source power yet.
2. Install the I/O Simulator in slot 0 of slave station #1, next to the CPU.
3. Install the Output module in slot 1 of slave station #1.

Remember these Four Steps!

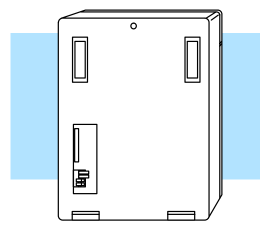
Use these steps to build your example network. The remainder of this chapter provides detailed explanations and examples of these steps.



1. Design the network by:
   - Selecting the configuration (this is a Host as Master configuration)
   - Building the communication cables.



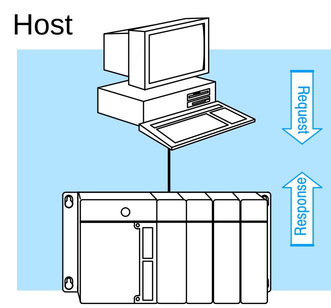
2. Select the communication settings by:
   - Setting the station switches



3. Write the communication control program.
   - ***Direct***NET program is used with Host master configurations.

```
:
130   NORMAL$=CHR$(&H4E)
140   SLAVEADDR$=HEX$(SLAVEADDRESS)
150   IF LEN(SLAVEADDR$)<2
       THEN SLAVEADDR$="0"+SLAVEADDR$
160   OFFSETADDR$=CHR$(&H20+SLAVEADDRESS)
170   ENQ$=CHR$(&H5)
:
```

4. Start the network operation.



Step 1: Design the Network

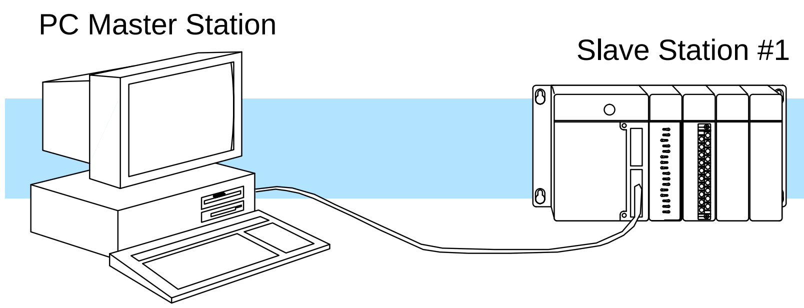
The Example Configuration

In this chapter we'll use the following example configuration to create a simple Host master network.

WARNING: These examples are for illustration purposes only and are not intended for use in actual applications. This is because there may be many aspects of your system safety precautions that are not addressed in the examples. If you use these examples in actual applications, you are increasing the risk of personal injury and/or property damage.

Goal:

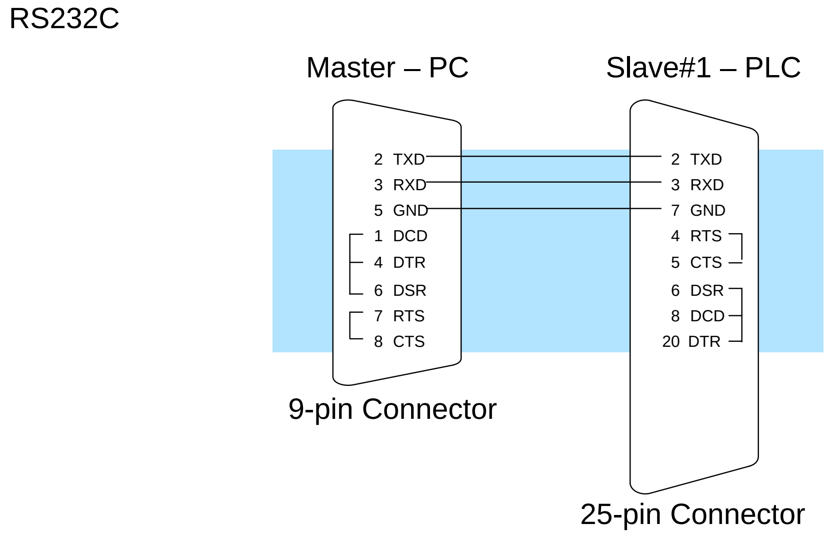
1. Read a bit pattern from an input module (X0 – X7) in Slave #1
2. Write a bit pattern to an output module (Y0 – Y7) in Slave #1.



The Example Cable In our example configuration we have:

- A personal computer as the master station
- A DL440 PLC with a built-in ***Direct***Net port as slave station #1

Since this is a simple point-to-point network, we can use RS232C communications.



Step 2: Select the Communication Settings

Set the Personal Computer as the Master Station

The next step is to set the personal computer communication parameters (in the BASIC program). The parameters are typically set with software on the personal computer. The best place to find this information is in the documentation that came with your computer. It is important the communication settings for both the master and the slave(s) are the same. We will use the following settings in our example.

- Baud Rate — 9600 baud
- Parity — None

Set the PLC Switches for the Slave Station

Slave station #1 is a DL405 PLC. If you look at the back of the DL405 CPU you will notice a small bank of switches. Switches 2, 3, and 4 are used to set the communication parameters for the bottom communication port.

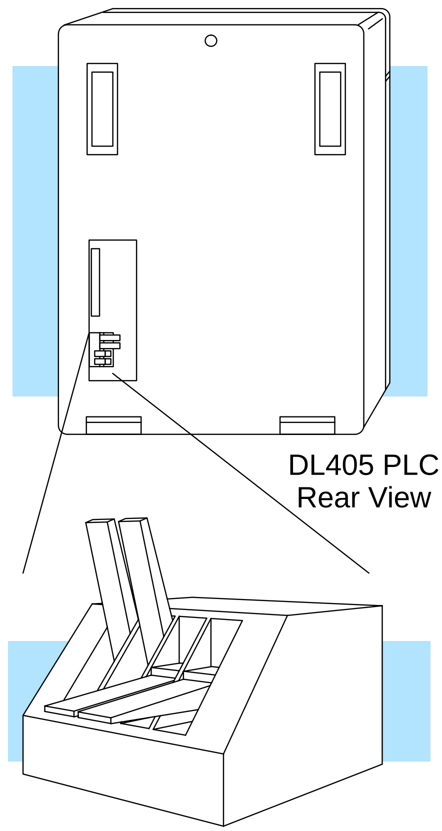
Switch 2 — This switch selects the CPU slave address. If this switch is On, then an address of 1 is used. If the switch is Off, then you can use a programming device to set the address.

Switch 3 & 4 — These switches select the baud rate for the bottom port.

Baud	3	4
300	OFF	OFF
1200	OFF	ON
9600	ON	OFF
19200	ON	ON

Set switch 2 to OFF – Address 1.
Set switch 3 ON and 4 OFF – 9600 Baud

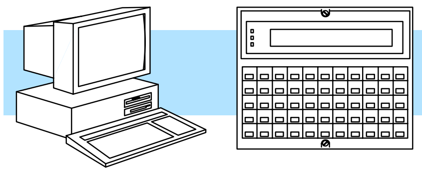
Switch 1 does not apply to the networking example. It is used to select the battery low indictor operation.



Set the Station Address for the Slave Station

With ***Direct***SOFT, use AUX 56 from the Auxiliary functions menu to set the port parameters.

With the DL405 Handheld Programmer, use AUX 56 to set the port parameters. The following example shows how to use the handheld programmer to set the address.



DirectSOFT DL405 Handheld Programmer

NOTE: The PLC port address is set in decimal, not hexadecimal.

Select AUX 56

CLR CLR AUX 5 6 ENT ENT

AUX 5* CPU CFG
AUX 56 CPU N/W ADDRESS

Enter the Address (in decimal)

0 1 ENT

AUX 56 CPU N/W ADDRESS
N/W # 01

Set Address to 1 (decimal)

Select HEX or ASCII

To change the mode press .... →

then press enter. ENT

AUX 56 CPU N/W ADDRESS
HEX / ASCII

Select ASCII mode for data transfer.

Select the Parity Option

To change the parity press .... →

then press enter. ENT

AUX 56 CPU N/W ADDRESS
NONE / ODD

Set Parity to none.

Clear the Display

To clear the display press.... CLR

OK

Step 3: Create the Communications Program

***Direct*NET Programs**

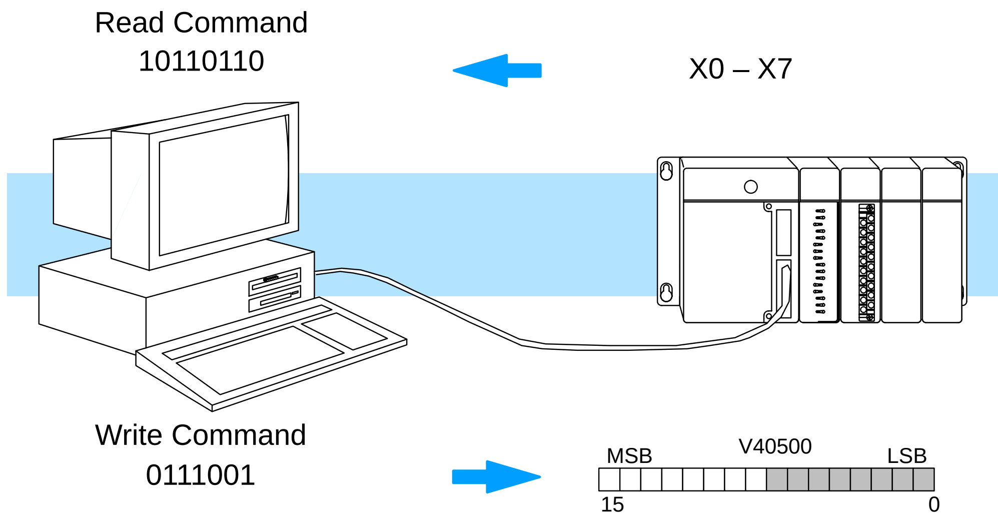
The communications program used with a hosted network is more complex than the simple RLL instructions used with the other configurations, but the concept is the same. The host is the ***Direct***NET master and must use a ***Direct***NET protocol communications program to initiate all network requests to read or write data. These communication programs can be written in many different languages, such as BASIC, C, etc. and must include the appropriate ***Direct***NET protocol commands.

Example Program

We've provided two host programs for this example. The programs are written in BASIC, which is provided with many personal computers.

Goal:

1. Read a bit pattern from an input module (X0 – X7).
2. Write a bit pattern to the first output module (Y0–Y7). (We'll actually use V40500, which is the V-memory location that corresponds to Y0–Y7).



Entering the Program

You can use most any file editor to enter the program. Make sure your editor does not embed formatting codes. For example, most word processing packages and/or desktop publishing packages embed codes that describe the page layout. These types of packages are generally not suitable for entering programs.

Slave Station Programs

The slave station does not require any programming in this example.

NOTE: The DL405 CPU must be in the ASCII mode for the following BASIC programs to work.

Example Program to Read Data

The following program will read X0–X7 from Slave Station #1.

```
10    REM Program to read X0–X7 from a 405 PLC
20    REM
30    REM Define all variables
40    REM
50    REM Change the slave address in HEX at line 60 if required.
60    SLAVEADDRESS=&H1
70    DATATYPE$=CHR$(&H32)
80    DATAADDR$=CHR$(&H30)+CHR$(&H31)+CHR$(&H30)+CHR$(&H31)
90    COMPLETEBLK$=CHR$(&H30)+CHR$(&H30)
100   PARTBLK$=CHR$(&H30)+CHR$(&H32)
110   MASTERADDR$=CHR$(&H30)+CHR$(&H30)
120   NORMAL$=CHR$(&H4E)
130   SLAVEADDR$=HEX$(SLAVEADDRESS)
140   IF LEN(SLAVEADDR$)<2 THEN SLAVEADDR$=”0”+SLAVEADDR$
150   OFFSETADDR$=CHR$(&H20+SLAVEADDRESS)
160   ENQ$=CHR$(&H5)
170   ACK$=CHR$(&H6)
180   SOH$=CHR$(&H1)
190   STX$=CHR$(&H2)
200   ETB$=CHR$(&H17)
210   NAK$=CHR$(&H15)
220   ETX$=CHR$(&H3)
230   EOT$=CHR$(&H4)
240   RED$=CHR$(&H30)
250   DATA$=””
260   REM
270   REM Build enquiry and header
280   REM
290   SLAVEENQ$=NORMAL$+OFFSETADDR$+ENQ$
300   HEADER$=SOH$+SLAVEADDR$+RED$+DATATYPE$+DATAADDR$
310   HEADER$=HEADER$+COMPLETEBLK$+PARTBLK$+MASTERADDR$+ETB$
320   REM
330   REM Begin active program
340   REM
350   CLS
360   OPEN ”COM1:9600,N,8,1,RS,DS” AS #1
370   LRC=0
380   GOSUB 770
390   RETRY=0
400   REM            Do Enquiry
410   PRINT #1,SLAVEENQ$;
420   INPUT #1,ENQRES$:PRINT ”Enquire Response = ”;ENQRES$
430   IF MID$(ENQRES$,3,1)=ACK$ THEN GOTO 480
440   RETRY=RETRY+1
450   IF RETRY>2 THEN GOTO 760
460   GOTO 410
```

```
470   REM              Do Header
480   RETRY=0
490   PRINT #1,HEADER$;LRC$;
500   INPUT #1,HEADERRESP$:PRINT ”Header Response = ”;HEADERRESP$
510   IF MID$(HEADERRESP$,1,1)=ACK$ THEN GOTO 560
520   RETRY=RETRY+1
530   IF RETRY>2 THEN GOTO 760
540   GOTO 490
550   REM              Get the data
560   RETRY=0
570   INPUT #1,DAT$
580   GOSUB 850
590   IF VALUELRC$=DATLRC$ THEN GOTO 640
600   PRINT #1,NAK$;
610   RETRY=RETRY+1
620   IF RETRY>2 THEN GOTO 760
630   GOTO 570
640   REM              Print the data to the screen
650   PRINT ”Data = ”;VALUE$
660   REM              Do the ACK
670   RETRY=0
680   PRINT #1,ACK$;
690   INPUT #1,ACKRESP$:PRINT ”EOT = ”;ACKRESP$
700   IF MID$(ACKRESP$,1,1)=EOT$ THEN GOTO 750
710   RETRY=RETRY+1
720   IF RETRY>2 THEN GOTO 760
730   GOTO 680
740   REM              Do the EOT
750   PRINT #1,EOT$;
760   CLEAR:CLOSE:END
770   REM
780   REM Routine to calculate LRC
790   REM
800   FOR I=2 TO (LEN(HEADER$)–1)
810   LRC=LRC XOR ASC(MID$(HEADER$,I,1))
820   NEXT I
830   LRC$=HEX$(LRC):IF LEN(LRC$)<2 THEN LRC$=”0”+LRC$
840   RETURN
850   REM
860   REM Routine to remove data from data packet, Get LRC for check in main
870   REM
880   VALUELRC=0
890   DATLRC$=””
900   FOR I=2 TO (LEN(DAT$)–3)
910   VALUE$=VALUE$+MID$(DAT$,I,1)
920   VALUELRC=VALUELRC XOR ASC(MID$(DAT$,I,1))
930   NEXT I
940   DATLRC$=MID$(DAT$,(LEN(DAT$)–1),2)
950   VALUELRC$=HEX$(VALUELRC)
960   IF LEN(VALUELRC$)<2 THEN VALUELRC$=CHR$(&H30)+VALUELRC$
970   RETURN
```

Example Program to Write Data

The following program will write a value to Y0 – Y7 in Slave Station #1.

```
10    REM Program to write X0–X7 to a 405 PLC
20    REM
30    REM Define all variables
40    REM
50    REM Change the slave address in HEX at line 60 if required
60    SLAVEADDRESS=&H1
70    DATATYPE$=CHR$(&H33)
80    DATAADDR$=CHR$(&H30)+CHR$(&H31)+CHR$(&H30)+CHR$(&H31)
90    COMPLETEBLK$=CHR$(&H30)+CHR$(&H30)
100   PARTBLK$=CHR$(&H30)+CHR$(&H32)
110   MASTERADDR$=CHR$(&H30)+CHR$(&H30)
120   REM
130   NORMAL$=CHR$(&H4E)
140   SLAVEADDR$=HEX$(SLAVEADDRESS)
150   IF LEN(SLAVEADDR$)<2 THEN SLAVEADDR$=”0”+SLAVEADDR$
160   OFFSETADDR$=CHR$(&H20+SLAVEADDRESS)
170   ENQ$=CHR$(&H5)
180   ACK$=CHR$(&H6)
190   SOH$=CHR$(&H1)
200   STX$=CHR$(&H2)
210   ETB$=CHR$(&H17)
220   NAK$=CHR$(&H15)
230   ETX$=CHR$(&H3)
240   EOT$=CHR$(&H4)
250   RITE$=CHR$(&H38)
260   DATA$=””
270   REM
280   REM Build enquiry and header
290   REM
300   SLAVEENQ$=NORMAL$+OFFSETADDR$+ENQ$
310    HEADER$=SOH$+SLAVEADDR$+RITE$+DATATYPE$+DATAADDR$
320   HEADER$=HEADER$+COMPLETEBLK$+PARTBLK$+MASTERADDR$+ETB$
330   REM
```

```
340  REM Begin active program
350  REM
360  CLS
370  OPEN ”COM1:9600,N,8,1,RS,DS” AS #1
380  LRC=0:DATLRC=0
390  INPUT ”ENTER DATA STRING (2 DIGITS, 0–F) ” , VALUE$
400  GOSUB 650
410  GOSUB 730
420  RETRY=0
430  PRINT #1,SLAVEENQ$;
435  INPUT #1,ENQRESP$:PRINT ”Enquire Response = ”;ENQRESP$
440  IF MID$(ENQRESP$,3,1)=ACK$ THEN GOTO 490
450  RETRY=RETRY+1
460  IF RETRY>2 THEN GOTO 640
470  GOTO 430
490  RETRY=0
500  PRINT #1,HEADER$;LRC$;
510  INPUT #1,HEADERRESP$:PRINT ”Header Response = ”;HEADERRESP$
520  IF MID$(HEADERRESP$,1,1)=ACK$ THEN GOTO 560
530  RETRY=RETRY+1
540  IF RETRY>2 THEN GOTO 640
550  GOTO 500
560  RETRY=0
570  PRINT #1,DAT$;DATLRC$;
580  INPUT #1,DATRESP$:PRINT ”Data Response = ”; DATRESP$
590  IF MID$(DATRESP$,1,1)=ACK$ THEN GOTO 630
600  RETRY=RETRY+1
610  IF RETRY>2 THEN GOTO 640
620  GOTO 570
630  PRINT #1,EOT$;
640  CLEAR:CLOSE:END
650  REM
660  REM Routine to calculate LRC
670  REM
680  FOR I=2 TO (LEN(HEADER$)–1)
690  LRC=LRC XOR ASC(MID$(HEADER$,I,1))
700  NEXT I
710  LRC$=HEX$(LRC):IF LEN(LRC$)<2 THEN LRC$=”0”+LRC$
720  RETURN
730  REM
740  REM Routine to put data into packet and calulate data LRC
750  REM
760  DAT$=CHR$(&H2)+VALUE$+CHR$(&H3)
770  FOR I=2 TO (LEN(DAT$)–1)
780  DATLRC=DATLRC XOR ASC(MID$(DAT$,I,1))
790  NEXT I
800  DATLRC$=HEX$(DATLRC):IF LEN(DATLRC$)<2 THEN DATLRC$=”0”+DATLRC$
810  RETURN
```

Step 4: Start the Network.

Execute the PC Master Program

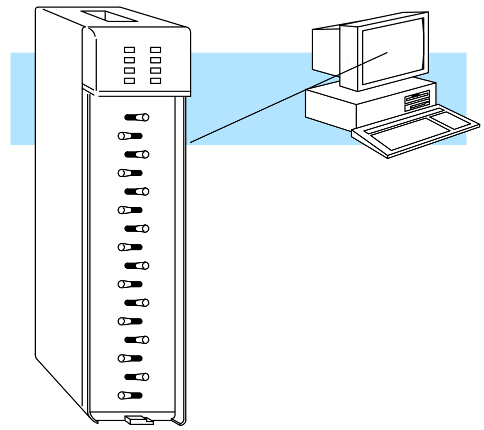
This program was designed to work with GWBasic. Complete the following steps to execute the example program.

1. Start GWBasic.
2. Load the file you used to create the read example program.

Verify the Read Command

Now you can change the I/O simulator switch settings and verify the communications.

1. Set every other switch to the ON position .
2. Execute the program. Look at the screen to verify the data matches the I/O simulator switch settings.
3. Change the I/O simulator switch settings at random and execute the program again. Notice how the data changes.

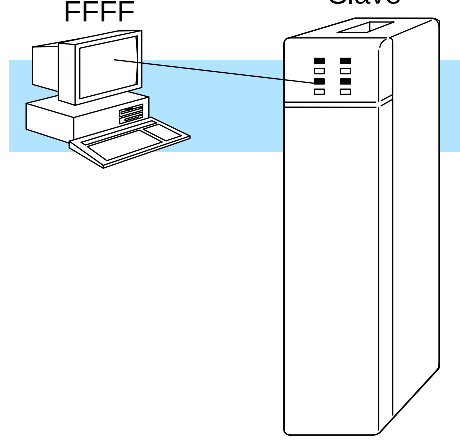


Verify the Write Command

You can change the bit pattern for the output module to verify it is working correctly. For example, enter FFFF. This should turn on the first eight output points in slave station #2.

Notice the numbers are entered in hexadecimal format. Here's a cross reference for a few patterns.

- FFFF – all on
- 0000 – all off



What should I do if it isn't working correctly?

Troubleshooting Steps

If the network does not seem to be working correctly, check the following items.

1. Cable and connections. Incorrectly wired cables and loose connectors cause the majority of problems. Verify you've selected the proper cable configuration and check to see the cable is wired correctly.
2. Dipswitch settings. Make sure you've set the slave stations to match the communication parameters required by the master station (operator interface or personal computer).
3. Communications program. Verify the program has been entered as shown in the example.
4. If the network still doesn't work correctly go to Chapter 7, Network Operation and Troubleshooting, and use the troubleshooting charts.

D

DL405 Data Types and Memory Maps

In This Chapter. . . .

DL405 Data Types

The following table shows the data types available with the DL405 products.

DL405 Data Type	Description	Bits per unit	Number of bytes	
			HEX	ASCII
31	V memory T / C current value	16 16	2 2	4 4
32	Inputs (X, GX, SP)	8	1	2
33	Outputs (Y, C, Stage, T/C bits)	8	1	2
39	Diagnostic Status	8	1	2

NOTE: Not all DL405 CPUs offer the same memory ranges. Check your DL405 User Manual to determine the ranges for your particular model.

Data Type 31
V Memory

V-memory Address	Memory Type	*Direct*NET Reference Address
V00000	TMR Current Time	0001
V00001	TMR Current Time	0002
— — —	— — —	— — —
V00377	TMR Current Time	0100
V01000	CTR Current Count	0201
V01001	CTR Current Count	0202
— — —	— — —	— — —
V01177	CTR Current Count	0280
V01400	V	0301
V01401	V	0302
— — —	— — —	— — —
V7377	V	0F00
V10000	V	1001
V10001	V	1002
— — —	— — —	— — —
V17777	V	2000
V40000	GX	4001
V40001	GX	4002
— — —	— — —	— — —
V40077	GX	4040

Data Type 31 (continued)

V-memory Address	Memory Type	*Direct*NET Reference Address
V40400	X	4101
V40401	X	4102
— — —	— — —	— — —
V40423	X	4114
V40500	Y	4141
V40501	Y	4142
— — —	— — —	— — —
V40523	Y	4154
V40600	C	4181
V40601	C	4182
— — —	— — —	— — —
V40677	C	41C0
V41000	Stage Bits	4201
V41001	Stage Bits	4202
— — —	— — —	— — —
V41077	Stage Bits	4240
V41100	TMR Status Bits	4241
V41101	TMR Status Bits	4242
— — —	— — —	— — —
V41117	TMR Status Bits	4250
V41140	CTR Status Bits	4261
V41141	CTR Status Bits	4262
— — —	— — —	— — —
V41147	CTR Status Bits	4268
V41200	Special Relay	4281
V41201	Special Relay	4282
— — —	— — —	— — —
V41205	Special Relay	4286
V41215	Special Relay	428E
V41216	Special Relay	428F
— — —	— — —	— — —
V41234	Special Relay	429D

Data Type 32 Inputs

V-Memory Address	Memory Type	Range	*Direct*NET Reference Address
V40000 LSB	GX	0007 – 0000	0001
V40000 MSB	GX	0017 – 0010	0002
V40001 LSB	GX	0027 – 0020	0003
— — —	— — —	— — —	— — —
V40077 LSB	GX	1767 – 1760	007F
V40077 MSB	GX	1777 – 1770	0080
V40400 LSB	X	0007 – 0000	0101
V40400 MSB	X	0017 – 0010	0102
V40401 LSB	X	0027 – 0020	0103
— — —	— — —	— — —	— — —
V40423 LSB	X	0467 – 0460	0127
V40423 MSB	X	0477 – 0470	0128
V41200 LSB	Special Relay	0007 – 0000	0181
V41200 MSB	Special Relay	0017 – 0010	0182
V41201 LSB	Special Relay	0027 – 0020	0183
— — —	— — —	— — —	— — —
V41205 LSB	Special Relay	0127 – 0120	018B
V41205 MSB	Special Relay	0137 – 0130	018C
V41215 LSB	Special Relay	0327 – 0320	019B
V41215 MSB	Special Relay	0337 – 0330	019C
V41216 LSB	Special Relay	0347 – 0340	019D
— — —	— — —	— — —	— — —
V41234 LSB	Special Relay	0707 – 0700	01B9
V41234 MSB	Special Relay	0717 – 0710	01BA

Data Type 33 Outputs

V-Memory Address	Memory Type	Range	*Direct*NET Reference Address
V40500 LSB	Y	0007 – 0000	0101
V40500 MSB	Y	0017 – 0010	0102
V40501 LSB	Y	0027 – 0020	0103
— — —	— — —	— — —	— — —
V40523 LSB	Y	0467 – 0460	0127
V40523 MSB	Y	0477 – 0470	0128
V40600 LSB	C	0007 – 0000	0181
V40600 MSB	C	0017 – 0010	0182
V40601 LSB	C	0027 – 0020	0183
— — —	— — —	— — —	— — —
V40677 LSB	C	1767 – 1760	01FF
V40677 MSB	C	1777 – 1770	0200
V41000 LSB	Stage Bits	0007 – 0000	0281
V41000 MSB	Stage Bits	0017 – 0010	0282
V41001 LSB	Stage Bits	0027 – 0020	0283
— — —	— — —	— — —	— — —
V41077 LSB	Stage Bits	1767 – 1760	02FF
V41077 MSB	Stage Bits	1777 – 1770	0300
V41100 LSB	Timer Status Bits	0007 – 0000	0301
V41100 MSB	Timer Status Bits	0017 – 0010	0302
V41101 LSB	Timer Status Bits	0027 – 0020	0303
— — —	— — —	— — —	— — —
V41117 LSB	Timer Status Bits	0367 – 0360	031F
V41117 MSB	Timer Status Bits	0377 – 0370	0320
V41140 LSB	Counter Status Bits	0007 – 0000	0321
V41140 MSB	Counter Status Bits	0017 – 0010	0322
V41141 LSB	Counter Status Bits	0027 – 0020	0323
— — —	— — —	— — —	— — —
V41147 LSB	Counter Status Bits	0167 – 0160	032F
V41147 MSB	Counter Status Bits	0177 – 0170	0330

Data Type 39 Diagnostic Status

You can use Data Type 39 to obtain ***Direct***NET diagnostic status. The following tables show the reference addresses for the various types of information and the ***Direct***NET error codes (used with address 0000).

***Direct*NET Reference**	**Data Returned**
0000	Last error and previous error*
0002	Number of successful communications
0004	Number of erroneous communications
0006	Number of retries for header
0008	Number of retries for data
* The last error code is contained in the most significant byte. The previous error code is in the least significant byte. Two codes that are displayed are cleared by two successful communication exchanges.	

Error Code	Applicable Data Type	Error Description
00	All types	The transfer was successful.
01	All types	A timeout occurred in the serial link.
03	32, 33	A request was made to read or write a non-existent I/O point.
04	32, 33	A request was made for data of more I/O points than are available.
05	All types	A request was made to read or write odd bytes. The number of data requested was not a multiple of 4 in the ASCII mode.
06	31	A request was made to read or write one or more non-existent memory locations.
07	All types	A request was made to read or write a zero data byte.
08	36	An attempt was made to write a protected memory.
09	All types	An invalid code is specified or an attempt was made to write to an invalid address.
0A	39	A request was made to read or write one or more non-existent diagnostic status words.
0B	36, 39	An invalid starting address is used in the PC type read, scan start/stop, diagnostic status read or write request.
0C	All types	Three attempts were made to transmit the header.
0D	All types	Three attempts were made to transmit the data.
0F	All types	The header unit number is incorrect. An invalid function was requested.
14	All types	One or more errors occurred during the data block transfer. Possible errors are: invalid STX, ETC, LRC, or ETB is received; a parity, framing, or overrun error occurred.
15	All types	EOT from the master station could not be received.
16	All types	A code other than ACK or NAK was received.
1D	31, 33, 36	There is a format error in the non-header portion of the message.
1E	All types	There is a format error in the header block.

DL430 Memory Map

Memory Type	Discrete Memory Reference (octal)	Word Memory Reference (octal)	Qty. Decimal	Symbol
Input Points	X0 – X477	V40400 – V40423	320	X0
Output Points	Y0 – Y477	V40500 – V40523	320	Y0
Control Relays	C0 – C737	V40600 – V40635	512	C0 C0
Special Relays	SP0 – SP137 SP320 – SP617	V41200 – V41205 V41215 – V41230	288	SP0
Timers	T0 – T177	None	128	TMR T0 K100
Timer Current Values	None	V00000 – V00177	128	V0 K100 ≥
Timer Status Bits	T0 – T177	V41100 – V41107	128	T0
Counters	CT0 – CT177	None	128	CNT CT0 K10
Counter Current Values	None	V01000 – V01177	256	V1000 K100 ≥
Counter Status Bits	CT0 – CT177	V41040 – V41147	128	CT0
Data Words	None	V1400 – V7377	3072	None specific, used with many instructions
Stages	S0 – S577	V41000 – V41027	384	SG S 001 S0
Remote In / Out	GX0 – GX737	V40000 – V40037	512	GX0 GX0
System parameters	None	V7400 – V7777	256	None specific, used with many instructions

DL440 Memory Map

Memory Type	Discrete Memory Reference (octal)	Word Memory Reference (octal)	Qty. Decimal	Symbol
Input Points	X0 – X477	V40400 – V40423	320	X0
Output Points	Y0 – Y477	V40500–40523	320	Y0
Control Relays	C0 – C1777	V40600–40677	1024	C0 C0
Special Relays	SP0 – SP137 SP320 – SP617 SP620 – SP717	V41200–41205 V41215–41230 V41231 – V41234	352	SP0
Timers	T0 – T377	None	256	TMR T0 K100
Timer Current Values	None	V00000 – V00377	256	V0 K100 ≥
Timer Status Bits	T0 – T377	V41100 – V41117	256	T0
Counters	CT0 – CT177	None	128	CNT CT0 K10
Counter Current Values	None	V01000 – V01177	128	V1000 K100 ≥
Counter Status Bits	CT0 – CT177	V41040 – V41147	128	CT0
Data Words	None	V1400 – V7377 V10000 – V17777	3072	None specific, used with many instructions
Stages	S0 – S1777	V41000 – V41077	1024	SG S 001 S0
Remote In / Out	GX0 – GX1777	V40000 – V40077	1024	GX0 GX0
System parameters	None	V700 – V737 V7400 – V7777	288	None specific, used with many instructions

X Input Bit Map

This table provides a listing of the individual Input points associated with each V-memory address bit for the DL430 and DL440 CPUs.

MSB							DL430/DL440 Input (X) Points								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40400
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40401
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40402
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40403
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40404
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40405
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40406
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40407
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40410
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	V40411
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40412
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40413
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40414
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40415
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40416
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40417
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V40420
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V40421
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V40422
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V40423

Y Output Bit Map

This table provides a listing of the individual output points associated with each V-memory address bit for both the DL430 and DL440 CPUs.

MSB							DL430/DL440 Output (Y) Points								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40500
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40501
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40502
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40503
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40504
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40505
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40506
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40507
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40510
237	236	235	234	233	222	221	220	217	216	215	214	213	212	211	210	V40511
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40512
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40513
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40514
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40515
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40516
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40517
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V40520
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V40521
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V40522
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V40523

Remote I/O Bit Map

This table provides a listing of the individual remote I/O points associated with each V-memory address bit.

MSB					DL430/DL440 Remote I/O (GX) Points										LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40000
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40001
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40002
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40003
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40004
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40005
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40006
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40007
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40010
237	236	235	234	233	222	221	220	217	216	215	214	213	212	211	210	V40011
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40012
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40013
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40014
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40015
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40016
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40017
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V40020
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V40021
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V40022
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V40023
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V40024
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V40025
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V40026
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V40027
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V40030
637	636	635	634	633	622	621	620	617	616	615	614	613	612	611	610	V40031
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V40032
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V40033
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V40034
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V40035
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V40036
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V40037

This portion of the table shows additional Remote I/O points available with the DL440.

MSB							DL440 Additional Remote I/O (GX) Points								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
1017	1016	1015	1014	1013	1012	1011	1010	1007	1006	1005	1004	1003	1002	1001	1000	V40040
1037	1036	1035	1034	1033	1032	1031	1030	1027	1026	1025	1024	1023	1022	1021	1020	V40041
1057	1056	1055	1054	1053	1052	1051	1050	1047	1046	1045	1044	1043	1042	1041	1040	V40042
1077	1076	1075	1074	1073	1072	1071	1070	1067	1066	1065	1064	1063	1062	1061	1060	V40043
1117	1116	1115	1114	1113	1112	1111	1110	1107	1106	1105	1104	1103	1102	1101	1100	V40044
1137	1136	1135	1134	1133	1132	1131	1130	1127	1126	1125	1124	1123	1122	1121	1120	V40045
1157	1156	1155	1154	1153	1152	1151	1150	1147	1146	1145	1144	1143	1142	1141	1140	V40046
1177	1176	1175	1174	1173	1172	1171	1170	1167	1166	1165	1164	1163	1162	1161	1160	V40047
1217	1216	1215	1214	1213	1212	1211	1210	1207	1206	1205	1204	1203	1202	1201	1200	V40050
1237	1236	1235	1234	1233	1222	1221	1220	1217	1216	1215	1214	1213	1212	1211	1210	V40051
1257	1256	1255	1254	1253	1252	1251	1250	1247	1246	1245	1244	1243	1242	1241	1240	V40052
1277	1276	1275	1274	1273	1272	1271	1270	1267	1266	1265	1264	1263	1262	1261	1260	V40053
1317	1316	1315	1314	1313	1312	1311	1310	1307	1306	1305	1304	1303	1302	1301	1300	V40054
1337	1336	1335	1334	1333	1332	1331	1330	1327	1326	1325	1324	1323	1322	1321	1320	V40055
1357	1356	1355	1354	1353	1352	1351	1350	1347	1346	1345	1344	1343	1342	1341	1340	V40056
1377	1376	1375	1374	1373	1372	1371	1370	1367	1366	1365	1364	1363	1362	1361	1360	V40057
1417	1416	1415	1414	1413	1412	1411	1410	1407	1406	1405	1404	1403	1402	1401	1400	V40060
1437	1436	1435	1434	1433	1432	1431	1430	1427	1426	1425	1424	1423	1422	1421	1420	V40061
1457	1456	1455	1454	1453	1452	1451	1450	1447	1446	1445	1444	1443	1442	1441	1440	V40062
1477	1476	1475	1474	1473	1472	1471	1470	1467	1466	1465	1464	1463	1462	1461	1460	V40063
1517	1516	1515	1514	1513	1512	1511	1510	1507	1506	1505	1504	1503	1502	1501	1500	V40064
1537	1536	1535	1534	1533	1532	1531	1530	1527	1526	1525	1524	1523	1522	1521	1520	V40065
1557	1556	1555	1554	1553	1552	1551	1550	1547	1546	1545	1544	1543	1542	1541	1540	V40066
1577	1576	1575	1574	1573	1572	1571	1570	1567	1566	1565	1564	1563	1562	1561	1560	V40067
1617	1616	1615	1614	1613	1612	1611	1610	1607	1606	1605	1604	1603	1602	1601	1600	V40070
1637	1636	1635	1634	1633	1622	1621	1620	1617	1616	1615	1614	1613	1612	1611	1610	V40071
1657	1656	1655	1654	1653	1652	1651	1650	1647	1646	1645	1644	1643	1642	1641	1640	V40072
1677	1676	1675	1674	1673	1672	1671	1670	1667	1666	1665	1664	1663	1662	1661	1660	V40073
1717	1716	1715	1714	1713	1712	1711	1710	1707	1706	1705	1704	1703	1702	1701	1700	V40074
1737	1736	1735	1734	1733	1732	1731	1730	1727	1726	1725	1724	1723	1722	1721	1720	V40075
1757	1756	1755	1754	1753	1752	1751	1750	1747	1746	1745	1744	1743	1742	1741	1740	V40076
1777	1776	1775	1774	1773	1772	1771	1770	1767	1766	1765	1764	1763	1762	1761	1760	V40077

Control Relay Bit Map

This table provides a listing of the individual control relays associated with each V-memory address bit.

MSB						DL430/DL440 Control Relays (C)									LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40600
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40601
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40602
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40603
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40604
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40605
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40606
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40607
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40610
237	236	235	234	233	222	221	220	217	216	215	214	213	212	211	210	V40611
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40612
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40613
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40614
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40615
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40616
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40617
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V40620
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V40621
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V40622
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V40623
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V40624
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V40625
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V40626
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V40627
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V40630
637	636	635	634	633	622	621	620	617	616	615	614	613	612	611	610	V40631
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V40632
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V40633
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V40634
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V40635

This portion of the table shows additional Control Relays points available with the DL440.

MSB						DL440 Additional Control Relays (C)									LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V40636
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V40637
1017	1016	1015	1014	1013	1012	1011	1010	1007	1006	1005	1004	1003	1002	1001	1000	V40640
1037	1036	1035	1034	1033	1032	1031	1030	1027	1026	1025	1024	1023	1022	1021	1020	V40641
1057	1056	1055	1054	1053	1052	1051	1050	1047	1046	1045	1044	1043	1042	1041	1040	V40642
1077	1076	1075	1074	1073	1072	1071	1070	1067	1066	1065	1064	1063	1062	1061	1060	V40643
1117	1116	1115	1114	1113	1112	1111	1110	1107	1106	1105	1104	1103	1102	1101	1100	V40644
1137	1136	1135	1134	1133	1132	1131	1130	1127	1126	1125	1124	1123	1122	1121	1120	V40645
1157	1156	1155	1154	1153	1152	1151	1150	1147	1146	1145	1144	1143	1142	1141	1140	V40646
1177	1176	1175	1174	1173	1172	1171	1170	1167	1166	1165	1164	1163	1162	1161	1160	V40647
1217	1216	1215	1214	1213	1212	1211	1210	1207	1206	1205	1204	1203	1202	1201	1200	V40650
1237	1236	1235	1234	1233	1232	1231	1230	1227	1226	1225	1224	1223	1222	1221	1220	V40651
1257	1256	1255	1254	1253	1252	1251	1250	1247	1246	1245	1244	1243	1242	1241	1240	V40652
1277	1276	1275	1274	1273	1272	1271	1270	1267	1266	1265	1264	1263	1262	1261	1260	V40653
1317	1316	1315	1314	1313	1312	1311	1310	1307	1306	1305	1304	1303	1302	1301	1300	V40654
1337	1336	1335	1334	1333	1332	1331	1330	1327	1326	1325	1324	1323	1322	1321	1320	V40655
1357	1356	1355	1354	1353	1352	1351	1350	1347	1346	1345	1344	1343	1342	1341	1340	V40656
1377	1376	1375	1374	1373	1372	1371	1370	1367	1366	1365	1364	1363	1362	1361	1360	V40657
1417	1416	1415	1414	1413	1412	1411	1410	1407	1406	1405	1404	1403	1402	1401	1400	V40660
1437	1436	1435	1434	1433	1432	1431	1430	1427	1426	1425	1424	1423	1422	1421	1420	V40661
1457	1456	1455	1454	1453	1452	1451	1450	1447	1446	1445	1444	1443	1442	1441	1440	V40662
1477	1476	1475	1474	1473	1472	1471	1470	1467	1466	1465	1464	1463	1462	1461	1460	V40663
1517	1516	1515	1514	1513	1512	1511	1510	1507	1506	1505	1504	1503	1502	1501	1500	V40664
1537	1536	1535	1534	1533	1532	1531	1530	1527	1526	1525	1524	1523	1522	1521	1520	V40665
1557	1556	1555	1554	1553	1552	1551	1550	1547	1546	1545	1544	1543	1542	1541	1540	V40666
1577	1576	1575	1574	1573	1572	1571	1570	1567	1566	1565	1564	1563	1562	1561	1560	V40667
1617	1616	1615	1614	1613	1612	1611	1610	1607	1606	1605	1604	1603	1602	1601	1600	V40670
1637	1636	1635	1634	1633	1622	1621	1620	1617	1616	1615	1614	1613	1612	1611	1610	V40671
1657	1656	1655	1654	1653	1652	1651	1650	1647	1646	1645	1644	1643	1642	1641	1640	V40672
1677	1676	1675	1674	1673	1672	1671	1670	1667	1666	1665	1664	1663	1662	1661	1660	V40673
1717	1716	1715	1714	1713	1712	1711	1710	1707	1706	1705	1704	1703	1702	1701	1700	V40674
1737	1736	1735	1734	1733	1732	1731	1730	1727	1726	1725	1724	1723	1722	1721	1720	V40675
1757	1756	1755	1754	1753	1752	1751	1750	1747	1746	1745	1744	1743	1742	1741	1740	V40676
1777	1776	1775	1774	1773	1772	1771	1770	1767	1766	1765	1764	1763	1762	1761	1760	V40677

Stage Control / Status Bit Map

This table provides a listing of the individual stage control bits associated with each V-memory address bit.

MSB							DL430/DL440 Stage (S) Control Bits								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V41000
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V41001
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V41002
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V41003
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V41004
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V41005
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V41006
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V41007
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V41010
237	236	235	234	233	222	221	220	217	216	215	214	213	212	211	210	V41011
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V41012
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V41013
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V41014
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V41015
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V41016
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V41017
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V41020
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V41021
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V41022
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V41023
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V41024
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V41025
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V41026
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V41027

MSB							DL440 Additional Stage (S) Control Bits								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V41030
637	636	635	634	633	622	621	620	617	616	615	614	613	612	611	610	V41031
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V41032
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V41033
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V41034
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V41035
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V41036
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V41037

MSB				DL440 Additional Stage (S) Control Bits (continued)											LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
1017	1016	1015	1014	1013	1012	1011	1010	1007	1006	1005	1004	1003	1002	1001	1000	V41040
1037	1036	1035	1034	1033	1032	1031	1030	1027	1026	1025	1024	1023	1022	1021	1020	V41041
1057	1056	1055	1054	1053	1052	1051	1050	1047	1046	1045	1044	1043	1042	1041	1040	V41042
1077	1076	1075	1074	1073	1072	1071	1070	1067	1066	1065	1064	1063	1062	1061	1060	V41043
1117	1116	1115	1114	1113	1112	1111	1110	1107	1106	1105	1104	1103	1102	1101	1100	V41044
1137	1136	1135	1134	1133	1132	1131	1130	1127	1126	1125	1124	1123	1122	1121	1120	V41045
1157	1156	1155	1154	1153	1152	1151	1150	1147	1146	1145	1144	1143	1142	1141	1140	V41046
1177	1176	1175	1174	1173	1172	1171	1170	1167	1166	1165	1164	1163	1162	1161	1160	V41047
1217	1216	1215	1214	1213	1212	1211	1210	1207	1206	1205	1204	1203	1202	1201	1200	V41050
1237	1236	1235	1234	1233	1222	1221	1220	1217	1216	1215	1214	1213	1212	1211	1210	V41051
1257	1256	1255	1254	1253	1252	1251	1250	1247	1246	1245	1244	1243	1242	1241	1240	V41052
1277	1276	1275	1274	1273	1272	1271	1270	1267	1266	1265	1264	1263	1262	1261	1260	V41053
1317	1316	1315	1314	1313	1312	1311	1310	1307	1306	1305	1304	1303	1302	1301	1300	V41054
1337	1336	1335	1334	1333	1332	1331	1330	1327	1326	1325	1324	1323	1322	1321	1320	V41055
1357	1356	1355	1354	1353	1352	1351	1350	1347	1346	1345	1344	1343	1342	1341	1340	V41056
1377	1376	1375	1374	1373	1372	1371	1370	1367	1366	1365	1364	1363	1362	1361	1360	V41057
1417	1416	1415	1414	1413	1412	1411	1410	1407	1406	1405	1404	1403	1402	1401	1400	V41060
1437	1436	1435	1434	1433	1432	1431	1430	1427	1426	1425	1424	1423	1422	1421	1420	V41061
1457	1456	1455	1454	1453	1452	1451	1450	1447	1446	1445	1444	1443	1442	1441	1440	V41062
1477	1476	1475	1474	1473	1472	1471	1470	1467	1466	1465	1464	1463	1462	1461	1460	V41063
1517	1516	1515	1514	1513	1512	1511	1510	1507	1506	1505	1504	1503	1502	1501	1500	V41064
1537	1536	1535	1534	1533	1532	1531	1530	1527	1526	1525	1524	1523	1522	1521	1520	V41065
1557	1556	1555	1554	1553	1552	1551	1550	1547	1546	1545	1544	1543	1542	1541	1540	V41066
1577	1576	1575	1574	1573	1572	1571	1570	1567	1566	1565	1564	1563	1562	1561	1560	V41067
1617	1616	1615	1614	1613	1612	1611	1610	1607	1606	1605	1604	1603	1602	1601	1600	V41070
1637	1636	1635	1634	1633	1622	1621	1620	1617	1616	1615	1614	1613	1612	1611	1610	V41071
1657	1656	1655	1654	1653	1652	1651	1650	1647	1646	1645	1644	1643	1642	1641	1640	V41072
1677	1676	1675	1674	1673	1672	1671	1670	1667	1666	1665	1664	1663	1662	1661	1660	V41073
1717	1716	1715	1714	1713	1712	1711	1710	1707	1706	1705	1704	1703	1702	1701	1700	V41074
1737	1736	1735	1734	1733	1732	1731	1730	1727	1726	1725	1724	1723	1722	1721	1720	V41075
1757	1756	1755	1754	1753	1752	1751	1750	1747	1746	1745	1744	1743	1742	1741	1740	V41076
1777	1776	1775	1774	1773	1772	1771	1770	1767	1766	1765	1764	1763	1762	1761	1760	V41077

Timer Status Bit Map

This table provides a listing of the individual timer contacts associated with each V-memory address bit.

MSB						DL430/DL440 Timer (T) Contacts									LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V41100
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V41101
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V41102
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V41103
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V41104
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V41105
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V41106
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V41107

MSB						DL440 Additional Timer (T) Contacts									LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V41110
237	236	235	234	233	222	221	220	217	216	215	214	213	212	211	210	V41111
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V41112
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V41113
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V41114
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V41115
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V41116
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V41117

Counter Status Bit Map

This table provides a listing of the individual counter contacts associated with each V-memory address bit.

MSB							DL430/DL440 Counter (CT) Contacts								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V41140
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V41141
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V41142
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V41143
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V41144
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V41145
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V41146
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V41147

DL305 Data Types and Memory Map

In This Chapter. . . .

DL305 Data Types

The following table shows the data types available with the DL405 family of products.

DL305 Data Type	Description	Bits per unit	Number of bytes	
			HEX	ASCII
31	Data registers	8	1	2
	T / C accumulator	16	2	4
33	I/O, internal relays, shift register bits, T/C bits, stage bits	1	1	2
39	Diagnostic Status (5 word R/W)	16	10	20

Data Types 31 and 33

The following table provides address references for data types 31 and 33.

DL305 Data Type	Description	Read/ Write	DL330 Ranges		DL340 Ranges	
			PGM Ref.	*Direct*NET Ref	PGM Ref.	*Direct*NET Ref
31	Data registers	R / W	400 – 577	41 – 80	400 – 577	41 – 80
					700 – 777	81 – A0
	T / C accumulator		600 – 677	01 – 40	01 – 40	01 – 40
33	Input / Output bits	R/W	000 – 157	01 – 0E	000 – 157	01 – 0E
		R/W	700 – 767	39 – 3F	700 – 767	39 – 3F
		R/W			1000–1067	41 – 47
	Internal Relay Bits	Read	160 – 377	0F – 20	160 – 377	0F – 20
		Write	160 – 373	0F – 20	160 – 373	0F – 20
	Shift Register Bits	R / W	400 – 577	21 – 30	400 – 577	21 – 30
	T / C Bits	Read	600 – 677	31 – 38	600 – 677	31 – 38

Data Type 39 Diagnostic Status

You can use Data Type 39 to obtain ***Direct***NET diagnostic status. The only valid address for the DL305 products is 0000 (hex). There are 5 status words that can be read or cleared. You must access these words as a complete group. The following tables show the reference addresses for the various types of information and the ***Direct***NET error codes (used for word 1).

Address	Read/Write	Word	Diagnostic Status
0000 (h)	Read Write 0 (to clear)	1	Last error and previous error* Previous error code
		2	Number of successful communications
		3	Number of erroneous communications
		4	Number of retries for header
		5	Number of retries for data

* The last error code is contained in the most significant byte. The previous error code is in the least significant byte. Two codes that are displayed are cleared by two successful communication exchanges.

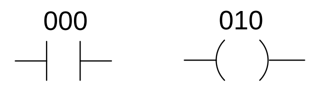
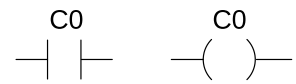
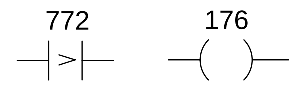
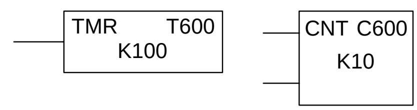
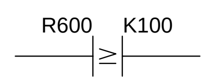
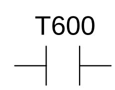
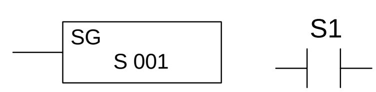
Error Code	Applicable Data Type	Error Description
00	All types	The transfer was successful. 00 also occurs if: 1. The transfer direction (Read / Write) is not 30 or 38. 2. A code other than ASCII code 0 to F has been received in ASCII mode. 3. E01 or E10 occurred during a write operation from the host to the CPU.
01	All types	A timeout occurred in the serial link.
03	32, 33	A request was made to read or write a non-existent I/O point.
04	32, 33	A request was made for data of more I/O points than are available.
05	All types	A request was made to read or write odd bytes. The number of data requested was not a multiple of 4 in the ASCII mode.
06	31	A request was made to read or write one or more non-existent memory locations.
07	All types	A request was made to read or write a zero data byte.
08	36	An attempt was made to write a protected memory.
09	All types	An invalid code is specified or an attempt was made to write to an invalid address.
0A	39	A request was made to read or write one or more non-existent diagnostic status words.
0B	36, 39	An invalid starting address is used in the PC type read, scan start/stop, diagnostic status read or write request.
0C	All types	Three attempts were made to transmit the header.
0D	All types	Three attempts were made to transmit the data.
0F	All types	The header unit number is incorrect. An invalid function was requested.
10	All types	Power is turned on
	31, 33	After power-up, an attempt was made to execute a function before a scan start/stop, or diagnostic status read or write.
14	All types	One or more errors occurred during the data block transfer. Possible errors are: invalid STX, ETC, LRC, or ETB is received; a parity, framing, or overrun error occurred.
15	All types	EOT from the master station could not be received.
16	All types	A code other than ACK or NAK was received.
1D	31, 33	In the data transfer between the host and the slave, one of the following errors occurred in the slave CPU: E02, E21, E31, or E41.
	36	The communication between the host and the slave is disconnected.
1E	All types	There is a format error in the header block.

DL330 Memory Map

Memory Type	Discrete Memory Reference (octal)	Register Memory Reference (octal)	Qty. Decimal	Symbol
Input / Output Points	000 – 157 700 – 767	R000 – R015 R070 – R076	168 Total	000 010
Control Relays	160 – 373	R016 – R037	140	C0 C0
Special Relays	374 – 377 770 – 777	R037 R077	12	772 376
Timers / Counters	600 – 673 674 – 677*	None	64	TMR T600 K100 CNT C600 K10
Timer / Counter Current Values	None	R600 – R673 R674 – R677*	64	R600 ≥ K100
Timer / Counter Status Bits	T600 – T673 T674 – T677*	None	64	T600
Data Words	None	R400 – R563	116	None specific, used with many instructions
Shift Registers	400 – 577	None	128	SR 400 417
Special Registers	None	R574 – R577	4	R574 – R575 used with FAULT R576 – R577 Auxiliary Accumulator

* T/ C Setpoint Unit Only. Can be used as data registers if the Timer/Counter Setpoint Unit or Thumbwheel Interface Module is not used. R564 – R573 contain the preset value used with the Timer / Counter Setpoint Unit. R674 – R677 contain the current values for these timers or counters.

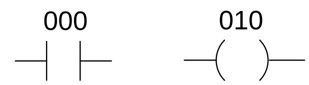
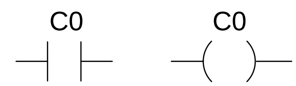
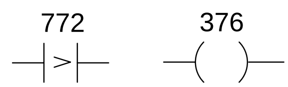
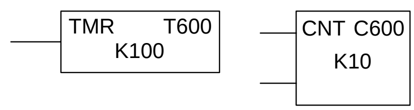
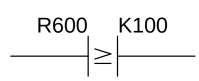
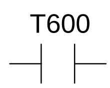
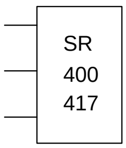
DL330P Memory Map

Memory Type	Discrete Memory Reference (octal)	Register Memory Reference (octal)	Qty. Decimal	Symbol
Input / Output Points	000 – 157 700 – 767	R000 – R015 R070 – R076	168 Total	
Control Relays	160 – 174 200 – 277	R016 – R017 R020 – R027	77	
Special Relays	175 – 177 770 – 777	R017 R077	11	
Timers / Counters	600 – 673 674 – 677*	None	64	
Timer / Counter Current Values	None	R600 – R673 R674 – R677*	64	
Timer / Counter Status Bits	T600 – T673 T674 – T677*	None	64	
Data Words	None	R400 – R563	116	None specific, used with many instructions
Stages	S0 – S177	R100 – R117	128	
Special Registers	None	R574 – R577	4	R574 – R575 used with FAULT R576 – R577 Auxiliary Accumulator

* T/ C Setpoint Unit Only. Can be used as data registers if the Timer/Counter Setpoint Unit or Thumbwheel Interface Module is not used, which provides a total of 128 data registers.

R564 – R573 contain the preset value used with the Timer / Counter Setpoint Unit. R674 – R677 contain the current values for these timers or counters.

DL340 Memory Map

Memory Type	Discrete Memory Reference (octal)	Register Memory Reference (octal)	Qty. Decimal	Symbol
Input / Output Points	000 – 157 700 – 767	R000 – R015 R070 – R076	168 Total	
Control Relays	160 – 373 1000 – 1067	R016 – R037 R100 – R106	180	
Special Relays	374 – 377 770 – 777 1070 – 1077	R037 R077 R107	20	
Timers / Counters	600 – 673 674 – 677*	None	64	
Timer / Counter Current Values	None	R600 – R673 R674 – R677*	64	
Timer / Counter Status Bits	T600 – T673 T674 – T677*	None	64	
Data Words	None	R400 – R563 R700 – R767	172	None specific, used with many instructions
Shift Registers	400 – 577	None	128	
Special Registers	None	R574 – R577 R770 – R777	12	R574–R575 used with FAULT R576–R577 Auxiliary Accumulator R770–R777 Communications Setup

* T/ C Setpoint Unit Only. Can be used as data registers if the Timer/Counter Setpoint Unit or Thumbwheel Interface Module is not used. R564 – R573 contain the preset value used with the Timer / Counter Setpoint Unit. R674 – R677 contain the current values for these timers or counters.

I/O Point Bit Map

These tables provide a listing of the individual Input points associated with each register location for the DL330, DL330P, and DL340 CPUs.

MSB			I/O References				LSB	Register Number
007	006	005	004	003	002	001	000	R0
017	016	015	014	013	012	011	010	R1
027	026	025	024	023	022	021	020	R2
037	036	035	034	033	032	031	030	R3
047	046	045	044	043	042	041	040	R4
057	056	055	054	053	052	051	050	R5
067	066	065	064	063	062	061	060	R6
077	076	075	074	073	072	071	070	R7
107	106	105	104	103	102	101	100	R10
117	116	115	114	113	112	111	110	R11
127	126	125	124	123	122	121	120	R12
137	136	135	134	133	132	131	130	R13
147	146	145	144	143	142	141	140	R14
157	156	155	154	153	152	151	150	R15
167	166	165	164	163	162	161	160	n/a
177	176	175	174	173	172	171	170	n/a
707	706	705	704	703	702	701	700	R70
717	716	715	714	713	712	711	710	R71
727	726	725	724	723	722	721	720	R72
737	736	735	734	733	732	731	730	R73
747	746	745	744	743	742	741	740	R74
757	756	755	754	753	752	751	750	R75
767	766	765	764	763	762	761	760	R76

NOTE: 160 – 167 can be used as I/O in a DL330 or DL330P CPU under certain conditions. 160 – 177 can be used as I/O in a DL340 CPU under certain conditions. You should consult the DL305 User Manual to determine which configurations allow the use of these points.

These points may be used as control relays. You cannot use them as both control relays and as I/O points. Also, if you use these points as I/O, you cannot access these I/O points as a Data Register reference using the DSTR5 (F55) and DOUT5 (F65) functions.

Control Relay Bit Map

The following tables provide a listing of the individual control relays associated with each register location for the DL305 CPUs.

NOTE: 160 – 167 can be used as I/O in a DL330 or DL330P CPU under certain conditions. 160 – 177 can be used as I/O in a DL340 CPU under certain conditions. You should consult the DL305 User Manual to determine which configurations allow the use of these points.

You cannot use them as both control relays and as I/O points. Also, if you use these points as I/O, you cannot access these I/O points as a Data Register reference using the DSTR5 (F55) and DOUT5 (F65) functions.

MSB			DL330 Control Relay References				LSB	Register Number
167	166	165	164	163	162	161	160	R16
177	176	175	174	173	172	171	170	R17
207	206	205	204	203	202	201	200	R20
217	216	215	214	213	212	211	210	R21
227	226	225	224	223	222	221	220	R22
237	236	235	234	233	232	231	230	R23
247	246	245	244	243	242	241	240	R24
257	256	255	254	253	252	251	250	R25
267	266	265	264	263	262	261	260	R26
277	276	275	274	273	272	271	270	R27
307	306	305	304	303	302	301	300	R30
317	316	315	314	313	312	311	310	R31
327	326	325	324	323	322	321	320	R32
337	336	335	334	333	332	331	330	R33
347	346	345	344	343	342	341	340	R34
357	356	355	354	353	352	351	350	R35
367	366	365	364	363	362	361	360	R36
				373	372	371	370	R37

* Control relays 340 – 373 can be made retentive by setting a CPU dipswitch. See the DL305 User Manual for details on setting CPU dipswitches.

MSB			DL330P Control Relay References				LSB	Register Number
167	166	165	164	163	162	161	160	R16
			174	173	172	171	170	R17
207	206	205	204	203	202	201	200*	R20
217	216	215	214	213	212	211	210	R21
227	226	225	224	223	222	221	220	R22
237	236	235	234	233	232	231	230	R23
247	246	245	244	243	242	241	240	R24
257	256	255	254	253	252	251	250	R25
267	266	265	264	263	262	261	260	R26
277*	276	275	274	273	272	271	270	R27

* Control relays 200 – 277 can be made retentive by setting a CPU dipswitch. See the DL305 User Manual for details on setting CPU dipswitches.

MSB			DL340 Control Relay References				LSB	Register Number
167	166	165	164	163	162	161	160	R16
177	176	175	174	173	172	171	170	R17
207	206	205	204	203	202	201	200	R20
217	216	215	214	213	212	211	210	R21
227	226	225	224	223	222	221	220	R22
237	236	235	234	233	232	231	230	R23
247	246	245	244	243	242	241	240	R24
257	256	255	254	253	252	251	250	R25
267	266	265	264	263	262	261	260	R26
277	276	275	274	273	272	271	270	R27
307	306	305	304	303	302	301	300	R30
317	316	315	314	313	312	311	310	R31
327	326	325	324	323	322	321	320	R32
337	336	335	334	333	332	331	330	R33
347	346	345	344	343	342	341	340*	R34
357	356	355	354	353	352	351	350	R35
367	366	365	364	363	362	361	360	R36
				373*	372	371	370	R37
1007	1006	1005	1004	1003	1002	1001	1000	R100
1017	1016	1015	1014	1013	1012	1011	1010	R101
1027	1026	1025	1024	1023	1022	1021	1020	R102
1037	1036	1035	1034	1033	1032	1031	1030	R103
1047	1046	1045	1044	1043	1042	1041	1040	R104
1057	1056	1055	1054	1053	1052	1051	1050	R105
1067	1066	1065	1064	1063	1062	1061	1060	R106

* Control relays 340 – 373 can be made retentive by setting a CPU dipswitch. See the DL305 User Manual for details on setting CPU dipswitches.

Special Relays

The following table shows the Special Relays used with the DL305 CPUs.

CPUs	Special Relay	Description of Contents
DL330P	175	100 ms clock, on for 50 ms and off for 50 ms.
	176	Disables all outputs except for those entered with the SET OUT instruction.
	177	Battery voltage is low.
DL330 DL340	374	On for the first scan cycle after the CPU is switched to Run Mode.
	375	100 ms clock, on for 50 ms and off for 50 ms.
	376	Disables all outputs except for those entered with the SET OUT instruction.
	377	Battery voltage is low.
DL330 DL330P DL340	770	Changes timers to 0.01 second intervals. Timers are normally 0.1 second time intervals.
	771	The external diagnostics FAULT instruction (F20) is in use.
	772	The data in the accumulator is greater than the comparison value.
	773	The data in the accumulator is equal to the comparison value.
	774	The data in the accumulator is less than the comparison value.
	775	An accumulator carry or borrow condition has occurred.
	776	The accumulator value is zero.
	777	The accumulator has an overflow condition.
DL340	1074	The RX or WX instruction is active.
	1075	An error occurred during communications with the RX or WX instructions.
	1076	Port 2 communications mode: on = ASCII mode, off = HEX mode. ***Direct***NET supports both ASCII and HEX modes and Modbus® only supports HEX mode.
	1077	Port 1 communications mode: on = ASCII mode, off = HEX mode

Timer / Counter Registers and Contacts

The following table shows the locations used for programming timer or counters. Since timers and counters share the same data area, you cannot have timers and counters with duplicate numbers. For example, if you have Timer 600, you cannot have a Counter 600.

Each register contains the current value for the timer or counter. Each timer or counter also has a timer or counter contact with the same reference number.

NOTE: Counter current values are retentive and retain their state after a power cycle.

Timer/Counter References/Registers							
607	606	605	604	603	602	601	600
617	616	615	614	613	612	611	610
627	626	625	624	623	622	621	620
637	636	635	634	633	632	631	630
647	646	645	644	643	642	641	640
657	656	655	654	653	652	651	650
667	666	665	664	663	662	661	660
677*	676*	675*	674*	673	672	671	670

* Used with Timer / Counter Setpoint Unit and /or Thumbwheel Interface Module.

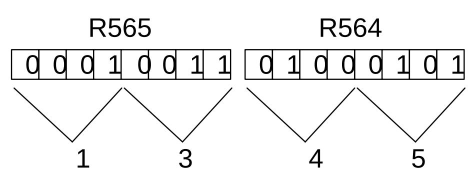
External Timer/Counter Setpoint Unit

Registers 674–677 are used in programming for use with the Timer/Counter Setpoint Unit and the Thumbwheel Interface Module that are available in some compatible product families. The registers contain the current time or count. There is also a status bit for each register with the same reference number. For example, the current value for Timer 674 is stored in R674 and the status contact is T674.

The presets for these modules are stored in R564 – R573 as follows.

- R564 – R565 — 1st T/C preset
- R566 – R567 — 2nd T/C preset
- R570 – R571 — 3rd T/C preset
- R572 – R573 — 4th T/C preset

The example shows how a 4-digit number would be represented in these registers.



Data Registers

The following 8-bit data registers are primarily used with data instructions to store various types of application data. For example, you could use a register to hold a timer or counter preset value.

Some data instructions call for two bytes, which will correspond to two consecutive 8-bit data registers such as R401 and R400. The LSB (Least Significant Bit) will be in register R400 as bit0 and the MSB (Most Significant Bit) will be in register R401 as bit17.

NOTE: Data Registers are retentive.

DL330 / DL330P 8-Bit Data Registers							
407	406	405	404	403	402	401	400
417	416	415	414	413	412	411	410
427	426	425	424	423	422	421	420
437	436	435	434	433	432	431	430
447	446	445	444	443	442	441	440
457	456	455	454	453	452	451	450
467	466	465	464	463	462	461	460
477	476	475	474	473	472	471	470
507	506	505	504	503	502	501	500
517	516	515	514	513	512	511	510
527	526	525	524	523	522	521	520
537	536	535	534	533	532	531	530
547	546	545	544	543	542	541	540
557	556	555	554	553	552	551	550
				563	562	561	560

DL340 8-Bit Data Registers							
407	406	405	404	403	402	401	400
417	416	415	414	413	412	411	410
427	426	425	424	423	422	421	420
437	436	435	434	433	432	431	430
447	446	445	444	443	442	441	440
457	456	455	454	453	452	451	450
467	466	465	464	463	462	461	460
477	476	475	474	473	472	471	470
507	506	505	504	503	502	501	500
517	516	515	514	513	512	511	510
527	526	525	524	523	522	521	520
537	536	535	534	533	532	531	530
547	546	545	544	543	542	541	540
557	556	555	554	553	552	551	550
				563	562	561	560
707	706	705	704	703	702	701	700
717	716	715	714	713	712	711	710
727	726	725	724	723	722	721	720
737	736	735	734	733	732	731	730
747	746	745	744	743	742	741	740
757	756	755	754	753	752	751	750
767	766	765	764	763	762	761	760

Stage Control / Status Bit Map

This table provides a listing of the individual stages and stage control bits. These are only available with the DL330P CPU.

MSB			Stage References				LSB	Register Number
007	006	005	004	003	002	001	000	R100
017	016	015	014	013	012	011	010	R101
027	026	025	024	023	022	021	020	R102
037	036	035	034	033	032	031	030	R103
047	046	045	044	043	042	041	040	R104
057	056	055	054	053	052	051	050	R105
067	066	065	064	063	062	061	060	R106
077	076	075	074	073	072	071	070	R107
107	106	105	104	103	102	101	100	R110
117	116	115	114	113	112	111	110	R111
127	126	125	124	123	122	121	120	R112
137	136	135	134	133	132	131	130	R113
147	146	145	144	143	142	141	140	R114
157	156	155	154	153	152	151	150	R115
167	166	165	164	163	162	161	160	R116
177	176	175	174	173	172	171	170	R117

Shift Register Bit Map

The shift register bits listed below are used in the shift register instruction. These outputs are discrete bits and are not the same locations as the 8 Bit Data Registers. These bits are retentive meaning they retain their state after a power cycle.

NOTE: The DL330P does not have Shift Register bits. Shift Register instructions in the DL330P use Control Relays memory references.

MSB			DL330 / DL340 Shift Register References				LSB	Register Number
407	406	405	404	403	402	401	400	R40
417	416	415	414	413	412	411	410	R41
427	426	425	424	423	422	421	420	R42
437	436	435	434	433	432	431	430	R43
447	446	445	444	443	442	441	440	R44
457	456	455	454	453	452	451	450	R45
467	466	465	464	463	462	461	460	R46
477	476	475	474	473	472	471	470	R47
507	506	505	504	503	502	501	500	R50
517	516	515	514	513	512	511	510	R51
527	526	525	524	523	522	521	520	R52
537	536	535	534	533	532	531	530	R53
547	546	545	544	543	542	541	540	R54
557	556	555	554	553	552	551	550	R55
567	566	565	564	563	562	561	560	R56
577	576	575	574	573	572	571	570	R57

With the DL340 CPU, these bits can also be used as control relays if they are not used with a Shift Register instruction.

Special Registers

This table provides a listing of the special registers used with the DL305 CPUs.

CPUs	Special Register	Description of Contents
DL330 DL330P DL340	R574 – 575	Contains the error code used with the FAULT instruction.
	R576 – 577	Auxiliary accumulator used with the MUL and DIV instructions.
DL340 Only	R771	Sets the upper byte of the station address assigned to the bottom communication port. Therefore, this will contain the 1st and 2nd digits of the address.
	R772	Sets the lower byte of the station address assigned to the bottom communication port. This only contains one digit, which is the 3rd digit of the address.
	R773	Sets the baud rate for the bottom communication port.
	R774	Sets the leading communications delay time for the bottom communication port.
	R775	Sets the trailing communications delay time for the bottom communication port.
	R776	Sets the leading communications delay time for the top communication port.
	R777	Sets the trailing communications delay time for the top communication port.

DL305 / 405 Cross Reference

If you are using a DL405 Master, you will have to make some slight changes in the way you request certain types of data. For example, the DL405 uses V-memory references instead of Register references. This section shows the cross references.

NOTE: Not all DL305 devices offer the same memory ranges. Check your DL305 User Manual to determine the ranges for your particular model.

Data Type 31: Register Access

To get to ... TMR / CTR Accumulator in a DL305	Use Reference ... in a DL405	To get to ... Register Data in a DL305	Use Reference ... in a DL405
R600	V000	R401, 400*	V100
R601	V001	R403, 402	V101
— — —	— — —	— — —	— — —
R624	V024	R777, 776	V237
R677	V077		
Two bytes of DL305 register data are returned with one DL405 V memory location.			

Data Type 33: I/O Point Access

Non RLL^{PLUS} CPUs			
To get to ... I/O Points, CRs, & Shift Registers in a DL305	Use Reference ... in a DL405	To get to ... TMR / CNT Status Bit in a DL305	Use Reference ... in a DL405
IO 000	GY000	600	GY600
IO 001	GY001	601	GY601
— — —	— — —	— — —	— — —
IO 157	GY157	677	GY677
CR160	GY160		
— — —	— — —		
CR 377	GY377		
IO 700	GY700		
IO 701	GY701		
— — —	— — —		
IO 1067	GY1067		
SR 400	GY400		
SR 401	GY401		
— — —	— — —		
SR 577	GY577		

RLL^{PLUS} CPUs					
To get to ... I/O Points, CRs, & Shift Registers in a DL305	**Use Ref. ... in a DL405**	**To get to ... Stage Status Bit in a DL305**	**Use Ref. ... in a DL405**	**To get to ... TMR / CNT Status Bit in a DL305**	**Use Ref. ... in a DL405**
IO 000	GY000	000	GY200	600	GY600
IO 001	GY001	001	GY201	601	GY601
— — —	— — —	— — —	— — —	— — —	— — —
CR160	GY160	177	GY377	677	GY677
— — —	— — —				
CR 277	GY277				
IO 700	GY700				
IO 701	GY701				
IO 1067	GY1067				
SR 200	GY400				
SR 201	GY 401				
— — —	— — —				
SR 277	GY477				

F

DL205 Data Types and Memory Map

In This Chapter. . . .

DL205 Data Types

The following table shows the data types available with the DL205 products.

DL205 Data Type	Description	Bits per unit	Number of bytes	
			HEX	ASCII
31	V memory T / C current value	16 16	2 2	4 4
32	Inputs (X, GX, SP)	8	1	2
33	Outputs (Y, C, Stage, T/C bits)	8	1	2
39	Diagnostic Status	8	1	2

NOTE: Not all DL205 devices offer the same memory ranges. Check your DL205 System Operation Manual to determine the ranges for your particular model.

Data Type 31 V-Memory Addresses

V-memory Address	Memory Type	*Direct*NET Reference Address
V00000 V00001 — — — V00177	TMR Current Time TMR Current Time — — — TMR Current Time	0001 0002 — — — 0080
V01000 V01001 — — — V01177	CTR Current Count CTR Current Count — — — CTR Current Count	0201 0202 — — — 0280
V02000 V02001 — — — V03777	V V — — — V	0401 0402 — — — 0800
V04000 V04001 — — — V04377	Non-volatile V Non-volatile V — — — Non-volatile V	0801 0802 — — — 0900
V07620 V07621 — — — V07737	Special V Special V — — — Special V	0F91 0F92 — — — 0FE0
V07746 V07647 — — — V07777	Special V Special V — — — Special V	0FE7 0FE8 — — — 1000

Data Type 31 V-Memory Addresses (continued)

V-memory Address	Memory Type	*Direct*NET Reference Address
V40400	X	4101
V40401	X	4102
— — —	— — —	— — —
V40423	X	4114
V40500	Y	4141
V40501	Y	4142
— — —	— — —	— — —
V40523	Y	4154
V40600	C	4181
V40601	C	4182
— — —	— — —	— — —
V40617	C	4190
V41000	Stage Bits	4201
V41001	Stage Bits	4202
— — —	— — —	— — —
V41037	Stage Bits	4218
V41100	TMR Status Bits	4241
V41101	TMR Status Bits	4242
— — —	— — —	— — —
V41107	TMR Status Bits	4248
V41140	CTR Status Bits	4261
V41141	CTR Status Bits	4262
— — —	— — —	— — —
V41147	CTR Status Bits	4268
V41200	Special Relay	4281
V41201	Special Relay	4282
— — —	— — —	— — —
V41205	Special Relay	4286
V41226	Special Relay	428F
V41227	Special Relay	4290
— — —	— — —	— — —
V41230	Special Relay	4299

**Data Type 32
Input Points**

V-Memory Address	Memory Type	Range	*Direct*NET Reference Address
V40400 LSB	X	0007 – 0000	0101
V40400 MSB	X	0017 – 0010	0102
V40401 LSB	X	0027 – 0020	0103
— — —	— — —	— — —	— — —
V40423 LSB	X	0467 – 0460	0127
V40423 MSB	X	0477 – 0470	0128
V41200 LSB	Special Relay	0007 – 0000	0181
V41200 MSB	Special Relay	0017 – 0010	0182
V41201 LSB	Special Relay	0027 – 0020	0183
— — —	— — —	— — —	— — —
V41205 LSB	Special Relay	0127 – 0120	018B
V41205 MSB	Special Relay	0137 – 0130	018C
V41226 LSB	Special Relay	0547 – 0540	01AD
V41226 MSB	Special Relay	0557 – 0550	01AE
— — —	— — —	— — —	— — —
V41230 LSB	Special Relay	0607 – 0600	01B1
V41230 MSB	Special Relay	0617 – 0610	01B2

Data Type 33 Outputs

V-Memory Address	Memory Type	Range	*Direct*NET Reference Address
V40500 LSB	Y	0007 – 0000	0101
V40500 MSB	Y	0017 – 0010	0102
V40501 LSB	Y	0027 – 0020	0103
— — —	— — —	— — —	— — —
V40523 LSB	Y	0467 – 0460	0127
V40523 MSB	Y	0477 – 0470	0128
V40600 LSB	C	0007 – 0000	0181
V40600 MSB	C	0017 – 0010	0182
V40601 LSB	C	0027 – 0020	0183
— — —	— — —	— — —	— — —
V40617 LSB	C	0367 – 0360	019F
V40617 MSB	C	0377 – 0370	01A0
V41000 LSB	Stage Bits	0007 – 0000	0281
V41000 MSB	Stage Bits	0017 – 0010	0282
V41001 LSB	Stage Bits	0027 – 0020	0283
— — —	— — —	— — —	— — —
V41037 LSB	Stage Bits	0767 – 0760	02BF
V41037 MSB	Stage Bits	0777– 0770	02C0
V41100 LSB	Timer Status Bits	0007 – 0000	0301
V41100 MSB	Timer Status Bits	0017 – 0010	0302
V41101 LSB	Timer Status Bits	0027 – 0020	0303
— — —	— — —	— — —	— — —
V41107 LSB	Timer Status Bits	0167 – 0160	030F
V41107 MSB	Timer Status Bits	0177 – 0170	0310
V41140 LSB	Counter Status Bits	0007 – 0000	0321
V41140 MSB	Counter Status Bits	0017 – 0010	0322
V41141 LSB	Counter Status Bits	0027 – 0020	0323
— — —	— — —	— — —	— — —
V41147 LSB	Counter Status Bits	0167 – 0160	032F
V41147 MSB	Counter Status Bits	0177 – 0170	0330

Data Type 39 Diagnostic Status

You can use Data Type 39 to obtain ***Direct***NET diagnostic status. The following tables show the reference addresses for the various types of information and the ***Direct***NET error codes (used with address 0000).

***Direct*NET Reference**	**Data Returned**
0000	Last error and previous error*
0002	Number of successful communications
0004	Number of erroneous communications
0006	Number of retries for header
0008	Number of retries for data
* The last error code is contained in the most significant byte. The previous error code is in the least significant byte. Two codes that are displayed are cleared by two successful communication exchanges.	

Error Code	**Applicable Data Type**	**Error Description**
00	All types	The transfer was successful.
01	All types	A timeout occurred in the serial link.
03	32, 33	A request was made to read or write a non-existent I/O point.
04	32, 33	A request was made for data of more I/O points than are available.
05	All types	A request was made to read or write odd bytes. The number of data requested was not a multiple of 4 in the ASCII mode.
06	31	A request was made to read or write one or more non-existent memory locations.
07	All types	A request was made to read or write a zero data byte.
08	36	An attempt was made to write a protected memory.
09	All types	An invalid code is specified or an attempt was made to write to an invalid address.
0A	39	A request was made to read or write one or more non-existent diagnostic status words.
0B	36, 39	An invalid starting address is used in the PC type read, scan start/stop, diagnostic status read or write request.
0C	All types	Three attempts were made to transmit the header.
0D	All types	Three attempts were made to transmit the data.
0F	All types	The header unit number is incorrect. An invalid function was requested.
14	All types	One or more errors occurred during the data block transfer. Possible errors are: invalid STX, ETC, LRC, or ETB is received; a parity, framing, or overrun error occurred.
15	All types	EOT from the master station could not be received.
16	All types	A code other than ACK or NAK was received.
1D	31, 33, 36	There is a format error in the non-header portion of the message.
1E	All types	There is a format error in the header block.

DL230 Memory Map

Memory Type	Discrete Memory Reference (octal)	Word Memory Reference (octal)	Qty. Decimal	Symbol
Input Points	X0 – X177	V40400 – V40407	128	X0
Output Points	Y0 – Y177	V40500 – V40507	128	Y0
Control Relays	C0 – C377	V40600 – V40617	256	C0 C0
Special Relays	SP0 – SP117 SP540 – SP577	V41200 – V41204 V41226 – V41227	112	SP0
Timers	T0 – T77		64	TMR T0 K100
Timer Current Values	None	V0 – V77	64	V0 K100 ≥
Timer Status Bits	T0 – T77	V41100 – V41103	64	T0
Counters	CT0 – CT77		64	CNT CT0 K10
Counter Current Values	None	V1000 – V1077	64	V1000 K100 ≥
Counter Status Bits	CT0 – CT77	V41140 – V41143	64	CT0
Data Words	None	V2000 – V2377	256	None specific, used with many instructions
Data Words Non–volitale	None	V4000 – V4177	128	None specific, used with many instructions
Stages	S0 – S377	V41000 – V41017	256	SG S 001 S0
System parameters	None	V7620 – V7647 V7750–V7777	48	None specific, used for various purposes

1 – The DL205 systems are limited to 128 discrete I/O points (total) with the present system hardware available. These can be mixed between input and output points as necessary.

DL240 Memory Map

Memory Type	Discrete Memory Reference (octal)	Word Memory Reference (octal)	Qty. Decimal	Symbol
Input Points	X0 – X177	V40400 – V40407	128 [1]	X0
Output Points	Y0 – Y177	V40500 – V40507	128 [1]	Y0
Control Relays	C0 – C377	V40600 – V40617	256	C0 C0
Special Relays	SP0 – SP137 SP540 – SP617	V41200 – V41205 V41226 – V41230	144	SP0
Timers	T0 – T177		128	TMR T0 K100
Timer Current Values	None	V0 – V177	128	V0 K100 ≥
Timer Status Bits	T0 – T177	V41100 – V41107	128	T0
Counters	CT0 – CT177		128	CNT CT0 K10
Counter Current Values	None	V1000 – V1177	128	V1000 K100 ≥
Counter Status Bits	CT0 – CT177	V41140 – V41147	128	CT0
Data Words	None	V2000 – V3777	1024	None specific, used with many instructions
Data Words Non–volitale	None	V4000 – V4377	256	None specific, used with many instructions
Stages	S0 – S777	V41000 – V41037	512	SG S 001 S0
System parameters	None	V7620 – V7737 V7746–V7777	106	None specific, used for various purposes

1 – The DL205 systems are limited to 128 discrete I/O points (total) with the present system hardware available. These can be mixed between input and output points as necessary.

X Input Bit Map

This table provides a listing of the individual Input points associated with each V-memory address bit for the DL230 and DL240 CPUs.

MSB							DL230/DL240 Input (X) Points								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40400
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40401
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40402
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40403
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40404
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40405
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40406
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40407

Y Output Bit Map

This table provides a listing of the individual output points associated with each V-memory address bit for both the DL230 and DL240 CPUs.

MSB							DL230/DL240 Output (Y) Points								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40500
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40501
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40502
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40503
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40504
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40505
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40506
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40507

Control Relay Bit Map

This table provides a listing of the individual control relays associated with each V-memory address bit.

MSB							DL230/DL240 Control Relays (C)								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40600
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40601
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40602
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40603
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40604
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40605
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40606
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40607
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40610
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	V40611
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40612
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40613
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40614
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40615
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40616
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40617

Stage Control / Status Bit Map

This table provides a listing of the individual stage control bits associated with each V-memory address bit.

MSB						DL230/DL240 Stage (S) Control Bits									LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V41000
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V41001
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V41002
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V41003
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V41004
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V41005
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V41006
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V41007
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V41010
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	V41011
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V41012
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V41013
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V41014
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V41015
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V41016
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V41017

MSB						DL240 Additional Stage (S) Control Bits									LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V41020
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V41021
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V41022
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V41023
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V41024
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V41025
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V41026
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V41027
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V41030
637	636	635	634	633	632	631	630	627	626	625	624	623	622	621	620	V41031
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V41032
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V41033
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V41034
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V41035
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V41036
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V41037

Timer Status Bit Map

This table provides a listing of the individual timer contacts associated with each V-memory address bit.

MSB							DL230/DL240 Timer (T) Contacts								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V41100
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V41101
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V41102
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V41103

MSB							Additional DL240 Timer (T) Contacts								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V41104
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V41105
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V41106
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V41107

Counter Status Bit Map

This table provides a listing of the individual counter contacts associated with each V-memory address bit.

MSB							DL230/DL240 Counter (CT) Contacts								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V41140
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V41141
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V41142
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V41143

MSB							Additional DL240 Counter (CT) Contacts								LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V41144
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V41145
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V41146
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V41147