SYSTEM DESIGN AND CONFIGURATION

CHAPTER 4

In This Chapter...

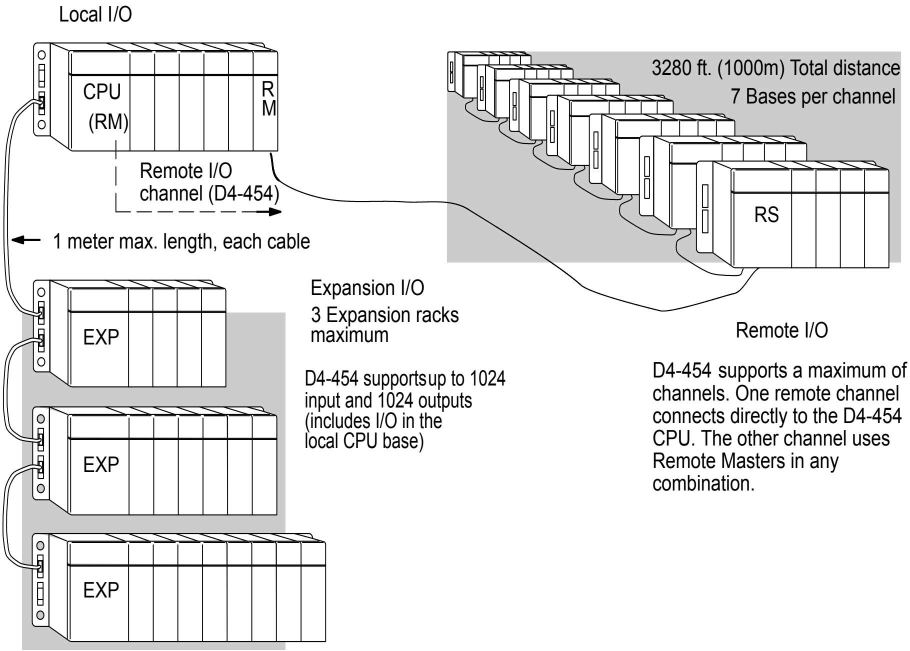
D4-454 System Design Strategies

I/O System Configurations

The D4-454 PLCs offer the following ways to add networking to the system:

- **Local I/O** - consists of I/O modules located in the same base as the CPU.
- **Expansion I/O** - consists of I/O modules in expansion bases located close the local base. Expansion cables connect them to the local CPU base's serial bus in daisy-chain fashion.
- **Remote I/O** - consists of I/O modules located in bases which are serially connected to the local CPU base through a Remote Master module, or may connect directly to port 3 on the D4-454.

A D4-454 system can be developed using many different arrangements of these configurations. All I/O configurations use the standard complement of DL405 I/O modules and bases. Below is a brief description of each of these configurations. Examples of each configuration are discussed in detail later in this chapter.



Network Configurations

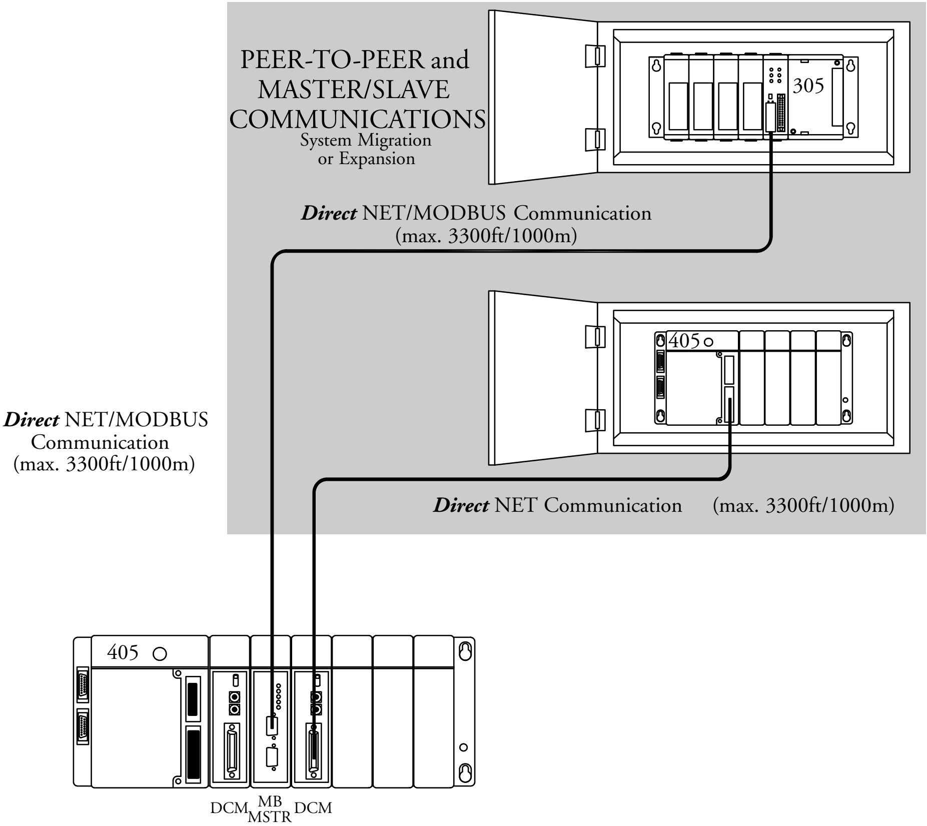
The D4-454 PLCs offer the following four ways to add I/O modules to the system:

Data Communications Module - connects a D4-454 system to devices using the DirectNET protocol, or connects as a slave to a MODBUS network.

D4-454 Communications Ports - the D4-454 CPU has 4 built-in communication ports. It allows two network connections directly from the CPU. See Chapter 3, CPU Specifications and Operation, for individual port specifications, and the sections at the end of this chapter for network connections.

MODBUS Master Module - You can use MODBUS master modules in any slot of a D4-454 system for connecting it as a master to a MODBUS network, using the RTU protocol.

MODBUS Slave Module - You can use MODBUS slave modules in any slot of a D4-454 system for connecting it as a slave to a MODBUS network, using the RTU protocol.



Module Placement and Configuration

Valid Module/Unit Locations

The most commonly used I/O modules for the D4-454 system (AC, DC, AC/DC, Relay and Analog) can be used in any base in your system. The table below lists by category the valid locations for all modules/units in a D4-454 system. Remember that the power budget can limit the number of modules in a base (discussed later).

Module Placement and Configuration

Module/Unit	Local CPU Base	Local Expansion Base	Remote I/O Base
CPUs	CPU Slot Only	–	–
Input Modules	✓	✓	✓
64pt DC Input Modules	✓ Note 1	✓ Note 1, 2	
Output Modules	✓	✓	✓
Relay Output Modules	✓	✓	✓
64pt DC Output Modules	✓ Note 1	✓ Note 1, 2	
Analog Modules	✓	✓	✓
Local Expansion			
D4-EX	–	✓	–
D4-EXDC	–	✓	–
Communications and Networking			
D4-DCM	✓ Note 3	–	–
H4-ECOM100	✓	–	–
F4-MAS-MB	✓	–	–
Remote I/O			
H4-ERM100	✓	–	–
H4-EBC	–	–	✓
D4-RM	✓	–	–
D4-RS	–	–	✓
D4-RSDC	–	–	✓
Motion			
D4-HSC	✓	✓	–
H4-CTRIO	✓	✓	✓
Specialty Modules			
D4-16SIM	✓	✓	✓
F4-16PID	✓	–	–
F4-8MPI	✓	–	–
F4-4LTC	✓	–	–
F4-CP128	✓	–	–
F4-CP128-T	✓	–	–
D4-FILL	✓	✓	✓

Note 1: When using 64 pt modules, you cannot use any specialty modules in slots 5, 6, and 7 in the same base.

Note 2: Specialty modules are allowed in expansion bases only if you are using the DL450 CPU and all bases in the system are the D4--xxB--1 type bases.

Note 3: When used with H4-ERM100 Ethernet Remote I/O system.

I/O Configuration Methods

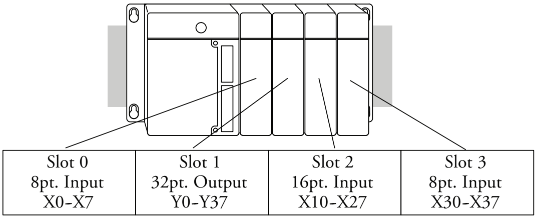
There are two methods of I/O configuration for the D4-454 CPUs.

- **Auto configuration** - the CPU automatically configures the I/O. It assigns the lowest I/O numbers to the module in slot 0 (the slot next to the CPU), the next set of I/O numbers to the next module in the base, etc. The numbers are assigned only to modules actually in the base, not to empty slots in the base. This is the default mode of the CPU.
- **Manual configuration** - The CPU allows you to assign I/O numbers. Numbers can be assigned to empty slots or in any order as long as the numbers are assigned in groups of
16 or 32.

Automatic Configuration

The D4-454 CPUs automatically detect any installed I/O modules (including specialty modules) at power up, and establish the correct I/O configuration and addresses. For most applications, you will never have to change the configuration.

I/O addresses use octal numbering, starting at X0 and Y0 in the slot next to the CPU. The addresses are assigned in groups of 8, 16, 32, or 64 depending on the number of points for the I/O module. The discrete input and output modules can be mixed in any order, but there may be restrictions placed on some specialty modules. The following diagram shows the I/O numbering convention for an example.



Slot 0 8pt. Input X0–X7	Slot 1 32pt. Output Y0–Y37	Slot 2 16pt. Input X10–X27	Slot 3 8pt. Input X30–X37

The DirectSOFT programming software allows you to view the automatic I/O configuration by using the PLC Configure I/O menu option.

Manual Configuration

It may never become necessary, but the D4-454 CPUs allow manual I/O address assignment for any I/O slot(s) in local or expansion bases. You can manually modify an auto configuration to match arbitrary I/O numbering. For example, two adjacent input modules can have starting addresses at X10 and X200.

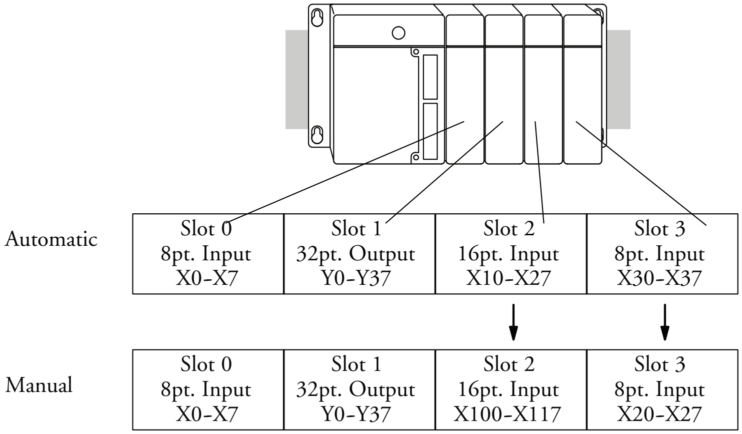
In automatic configuration, the addresses are assigned on 8-point boundaries. Manual configuration, however, assumes that all modules are at least 16 points, so you can only assign addresses that are a multiple of 20 (octal). For example, X30 and Y50 are not valid addresses. You can still use 8-point modules, but 16 addresses will be assigned and the upper 8 addresses will be unused.

WARNING: If you manually configure an I/O slot, the I/O addressing for the other modules may change. This is because the D4-454 CPUs do not allow you to assign duplicate I/O addresses. You must always correct any I/O configuration errors before you place the CPU in RUN mode. Uncorrected errors can cause unpredictable machine operation that can result in a risk of personal injury or damage to equipment.

Removing a Manual Configuration

After a manual configuration, the system will automatically retain the new I/O addresses through a power cycle. You can remove (overwrite) any manual configuration changes by simply performing an automatic configuration.

The following diagram shows how I/O addresses change after manually configuring a slot.



Automatic	Slot 0 8pt. Input X0-X7	Slot 1 32pt. Output Y0-Y37	Slot 2 16pt. Input X10-X27	Slot 3 8pt. Input X30-X37
Manual	Slot 0 8pt. Input X0-X7	Slot 1 32pt. Output Y0-Y37	Slot 2 16pt. Input X100-X117	Slot 3 8pt. Input X20-X27

Power-On I/O Configuration Check

The D4-454 CPUs can also be set to automatically check the I/O configuration on power-up. By selecting this feature you can detect any changes that may have occurred while the power was disconnected. For example, if someone places an output module in a slot that previously held an input module, the configuration check will detect the change and print a message on the DirectSOFT screen.

If the system detects a change in the I/O configuration at power-up, an error code E252 NEW I/O CONFIGURATION will be generated.

WARNING: You should always correct any I/O configuration errors before you place the CPU in RUN mode. Uncorrected errors can cause unpredictable machine operation that can result in a risk of personal injury or damage to equipment.

When a configuration error is generated, you may actually want to use the new I/O configuration. For example, you may have intentionally changed an I/O module to use with a program change.

WARNING: Verify the I/O configuration being selected will work properly with the CPU program. Always correct any I/O configuration errors before placing the CPU in RUN mode. Uncorrected errors can cause unpredictable machine operation that can result in a risk of personal injury or damage to the equipment.

Calculating the Power Budget

Managing your Power Resource

As you have seen, the I/O configuration depends on your choice of I/O modules, bases, and I/O location. When determining the types and quantity of I/O modules you will be using in the D4-454 system it is important to remember there is limited amount of power available from the power supply to the system. We have provided a chart to help you easily see the amount of power you will have with your CPU, Expansion Unit or Remote Slave selection. The following chart will help you calculate the amount of power you need with your I/O selections. At the end of this section you will also find an example of power budgeting and a worksheet for your own calculations.

If the I/O you choose exceeds the maximum power available from the power supply you can resolve the problem by shifting some of the modules to an expansion base which contains another power supply.

WARNING: You should always correct any I/O configuration errors before you place the CPU in RUN mode. Uncorrected errors can cause unpredictable machine operation that can result in a risk of personal injury or damage to equipment.

CPU Power Specifications

The following chart shows the amount of current available for the two voltages supplied on the D4-454 CPU, Expansion unit or Remote Slave unit. Use these current values when calculating the power budget for your system.

The Auxiliary 24V Power Source mentioned in the table is a connection at the base terminal strip allowing you to connect devices or DL405 modules that require 24VDC.

D4-454 Power Specifications

CPUs	5V Current Supplied in mA	Auxiliary 24V Power Source Current Supplied in mA
D4-454	3100	400
D4-454DC-1	3100	None

DL405 Power Specifications

Remote and Expansion Units	5V Current Supplied in mA	Auxiliary 24V Power Source Current Supplied in mA
D4-EX	4000	400
D4-EXDC	4000	None
D4-RS	3700	400
D4-RSDC	3700	None
H4-EBC	3470	400
H4-EBC-F	3300	400

Module Power Requirements

The chart on the next page shows the amount of maximum current required for each of the DL405 modules. Use these currents when calculating the power budget for your system. If external 24VDC is required, the external 24V from the CPU power supply may be used as long as the power budget is not exceeded.

Device	5V Current Required (mA)	External 24V Current Required (mA)
I/O Bases		
D4-04B-1	80	None
D4-06B-1	80	None
D4-08B-1	80	None
DC Input Modules		
D4-08ND3S	100	None
D4-16ND2	150	None
D4-16ND2F	150	None
D4-32ND3-1	150	None
D4-64ND2	300 (max)	None
AC Input Modules		
D4-08NA	100	None
D4-16NA	150	None
D4-16NA-1	150	None
AC/DC Input Modules		
D4-16NE3	150	None
F4-08NE3S	90	None
DC Output Modules		
F4-08TD1S	295	None
D4-16TD1	200	125
D4-16TD2	400	None
D4-32TD1	250	140
D4-42TD1-1	250	140 (5-15VDC)
D4-32TD2	350	120 / (4A max including loads)
D4-64TD1	800 (max)	None
AC Output Modules		
D4-08TA	250	None
D4-16TA	450	None
Relay Output Modules		
D4-08TR	550	None
F4-08TRS-1	575	None
F4-08TRS-2	575	None
D4-16TR	1000	None
Programming		
DV-1000*	150	None

* Retired.

Device	5V Current Required (mA)	External 24V Current Required (mA)
Analog Modules		
F4-04AD	85	100
F4-04ADS	270	120
F4-08AD	75	90
D4-02DA*	250	300
F4-04DA	120	180
F4-04DA-1	70	75 + 20 per channel
F4-04DA-2	90	75 + 20 per channel
F4-04DAS-1	60	50 per channel
F4-04DAS-2	60	60 per channel
F4-08DA-1	90	100 + 20 per channel
F4-16DA-1	90	100 + 20 per channel
F4-16DA-2	80	25 max.
F4-16AD-1	100	100
F4-16AD-2	75	100
F4-08THM-n	120	50 + 20 per channel
F4-08RTD	80	None
Remote I/O		
H4-ERM100	320	None
D4-RM	300	None
D4-RS	300	None
Communications and Networking		
D4-DCM	500	None
H4-ECOM100	300	None
F4-MAS-MB	235	None
CoProcessors ™		
F4-CP-128	305	None
F4-CP-512	235	None
F4-CP128-T	350	None
Specialty Modules		
D4-16SIM	150	None
D4-HSC	300	None
H4-CTRIO	400	None
F4-16PID	160	None
F4-8MPI	225	170
F4-4LTC	280	75

* Retired.

Power Budget Calculation Example

The following example shows how to calculate the power budget for the D4-454 system.

Base 0	Module Type	5VDC (mA)	Auxiliary Power Source 24VDC Output (mA)
CPU/ Expansion Unit/ Remote Slave Used	D4-454	3700	400
Slot 0	D4-16ND2	+ 150	+ 0
Slot 1	D4-16ND2	+ 150	+ 0
Slot 2	F4-04DA-1	+ 70	+ 155
Slot 3	D4-08ND3S	+ 100	+ 0
Slot 4	D4-08ND3S	+ 100	+ 0
Slot 5	D4-16TD2	+ 400	+ 0
Slot 6	D4-16TD2	+ 400	+ 0
Slot 7	D4-16TR	+ 1000	+ 0
Other			
Base	D4-08B-1	+ 80	+0
Maximum Power Required		2450	155
Remaining Power Available		3700–2450 = 1250	400–155 = 245

1. Using the tables at the beginning of the Power Budgeting section of this chapter, fill in the information for the CPU/Expansion Unit/Remote Slave, I/O modules, and any other devices that will use system power including devices that use the 24VDC output. Pay special attention to the current supplied by either the CPU, Expansion Unit, and Remote Slave since they do differ. Notice the Base falls into the "Other" category because it also has power requirements.

2. Add the current columns starting with Slot 0 and put the total in the row labeled "Maximum power required".

3. Subtract the row labeled "Maximum power required" from the row labeled "CPU/Expansion Unit/Remote Slave Used". Place the difference in the row labeled "Remaining Power Available".

4. If "Maximum Power Required" is greater than "CPU/Expansion Unit/Remote Slave Used" in any of the three columns, the power budget will be exceeded. It will be unsafe to use this configuration and you will need to restructure your I/O configuration.

Power Budget Calculation Worksheet

You may copy and use the following blank chart for your power budget calculations.

Base (X)	Module Type	5VDC (mA)	Auxiliary Power Source 24VDC Output (mA)
CPU/ Expansion Unit/ Remote Slave Used			
Slot 0			
Slot 1			
Slot 2			
Slot 3			
Slot 4			
Slot 5			
Slot 6			
Slot 7			
Other			
Maximum Power Required			
Remaining Power Available			

1. Using the tables at the beginning of the Power Budgeting section of this chapter, fill in the information for the CPU/Expansion Unit/Remote Slave, I/O modules, and any other devices that will use system power including devices that use the 24VDC output. Pay special attention to the current supplied by either the CPU, Expansion Unit, and Remote Slave since they do differ. Notice the Base falls into the "Other" category because it also has power requirements.

2. Add the current columns starting with Slot 0 and put the total in the row labeled "Maximum power required".

3. Subtract the row labeled "Maximum power required" from the row labeled "CPU/Expansion Unit/Remote Slave Used". Place the difference in the row labeled "Remaining Power Available".

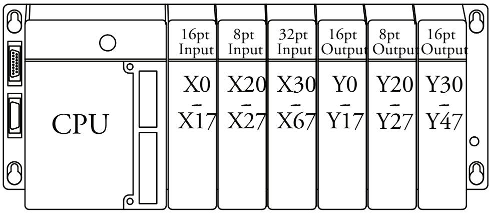
4. If "Maximum Power Required" is greater than "CPU/Expansion Unit/Remote Slave Used" in any of the three columns, the power budget will be exceeded. It will be unsafe to use this configuration and you will need to restructure your I/O configuration.

Local I/O Expansion

The following I/O base configurations will assist you in understanding the options available in the D4-454 series. Local and expanded bases are the most common and cost effective way of installing I/O. With local and expanded I/O the CPU can automatically configure the I/O for you. Use Remote I/O when it is necessary to locate I/O at distances away from the CPU. Remote I/O will require additional ladder programming to operate.

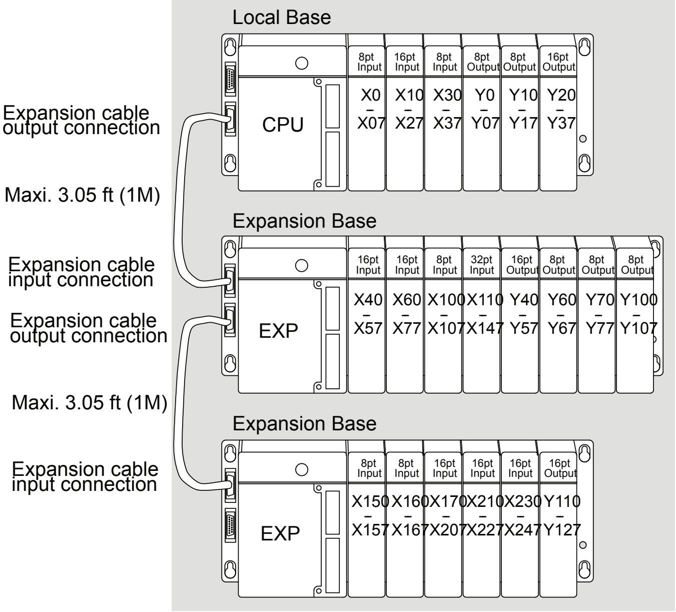
Local Base and I/O

The local base is the base in which the CPU resides. Local I/O modules reside in the same base as the CPU. For Example, placing 32-point modules in all eight slots in an 8-slot will use 256 I/O points. The status of each I/O point is updated each I/O scan of the CPU.



Local Expansion Base and I/O

Use local expansion when you need more I/O points or a greater power budget than the local base provides. The expansion bases require a Local Expansion Unit (in the place of a CPU), and a cable (either D4-EXCBL-1 or D4-EXCBL-2) to connect to the local CPU base. The CPU base is always the first base in the expansion chain. The following figure shows one CPU base, two expansion bases and examples of I/O numbering.



D4-454 supports a maximum of 3 expansion bases, and maximum of 1024 input points and 1024 output points (includes local base I/O)

Remote I/O Expansion

How to Add Remote I/O Channels

Remote I/O is useful for a system that has a sufficient number of sensors and other field devices located a relative long distance away (up to 1000 meters, or 3050 feet) from the more central location of the CPU. The methods of adding remote I/O are:

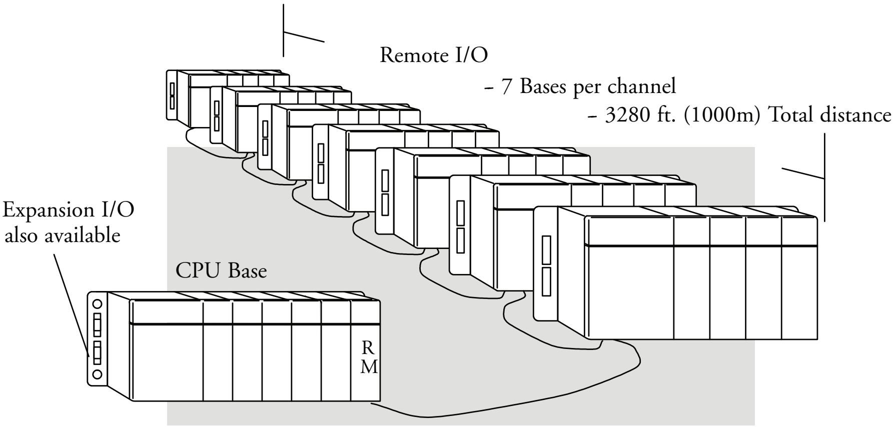
- The CPUs comm port 3 features a built-in Remote I/O channel.
- You may also use one or two D4-RM Remote Master modules in the local base.

Remote I/O	D4-454
Maximum number of Remote Masters supported in the local CPU base (1 channel per Remote Master)	2
CPU built-in Remote I/O channels	1
Maximum I/O points supported by each channel	512
Maximum Remote I/O points supported	1536
Maximum number of Remote I/O bases per channel	7

The use of Remote I/O does not limit the use of local expansion I/O discussed in the previous section. In fact, Remote I/O point numbering is assignable. Depending on the CPU scan time, remote I/O updates may be slower than local and expansion I/O, due to the serial communications involved.

Remote I/O points map into different CPU memory locations other than local / local expansion I/O. So, the addition of remote I/O does not reduce the number of local I/O points. Refer to the DL405 Remote I/O manual for details on remote I/O configuration and numbering.

The following figure shows 1 CPU base, and one remote I/O channel (D4-RM) with seven remote bases. If we had used the CPUs built-in remote I/O channel on port 3, we would not be required to add a remote master (D4-RM) module.



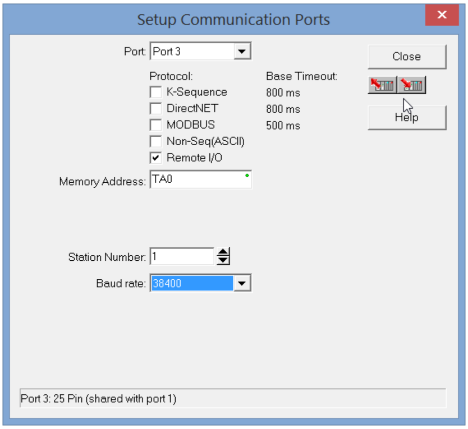
Configuring the CPU's Remote I/O Channel

This section describes how to configure the D4-454 built-in remote I/O channel. Additional information is in the Remote I/O manual, D4-REMIO-M, which you will need in configuring the Remote slave units on the network. You can use the D4-REMIO-M exclusively when using regular Remote Master and Remote Slave modules for remote I/O in the DL405 system.

The D4-454 CPU's built-in remote I/O channel has the same capability as a Remote Master module, the D4-RM. Specifically, it can communicate with up to seven remote bases containing a maximum of 512 I/O points, at a maximum distance of 1000 meters. If required, you can still use Remote Master modules in the local CPU base (512 I/O points on each channel), for a total of three channels providing 1536 total remote I/O points. First, we'll need to set up the Remote I/O communications.

You may recall from the CPU specifications in Chapter 3 that the D4-454's port 3 is capable of several protocols. To configure the port in DirectSOFT, choose the PLC menu, then Setup>Setup Secondary Comm Port.

- **Port**: From the port number list box at the top, choose "Port 3".
- **Protocol**: Click the box to the left of "Remote I/O" to select. The dialog shown below will appear.



Setup Communication Ports

Port: Port 3

Protocol:
□ K-Sequence 800 ms
□ DirectNET 800 ms
□ MODBUS 500 ms
□ Non-Seq(ASCII)
☑ Remote I/O

Base Timeout:

Memory Address: TA0

Station Number: 1

Baud rate: 38400

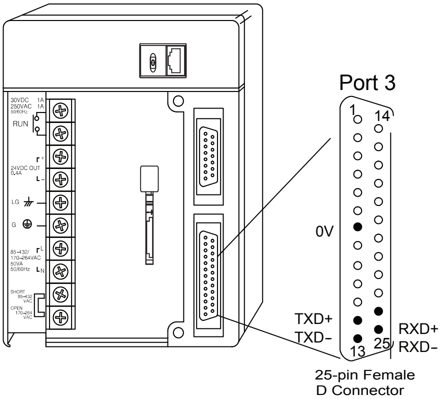
Port 3: 25 Pin (shared with port 1)

- **Station Number:** Choose "0" as the station number, which makes the D4-454 the master. Station numbers 1–7 are reserved for remote slaves.
- **Baud Rate:** The baud rates 19200 and 38400 baud are available. Choose 38400 initially as the remote I/O baud rate, and revert to 19200 baud if you experience data errors or noise problems on the link. Important: You must configure the baud rate on the Remote Slaves (via DIP switches) to match the baud rate selection for the CPU's port 3.
- **Memory Address**: Choose a V-memory address to use as the starting location of Remote I/O configuration table (V37700 is the default). This table is separate and independent from the table for any Remote Master(s) in the system.
- Then click the button indicated to send the Port 3 configuration to the CPU, and click **Close**.

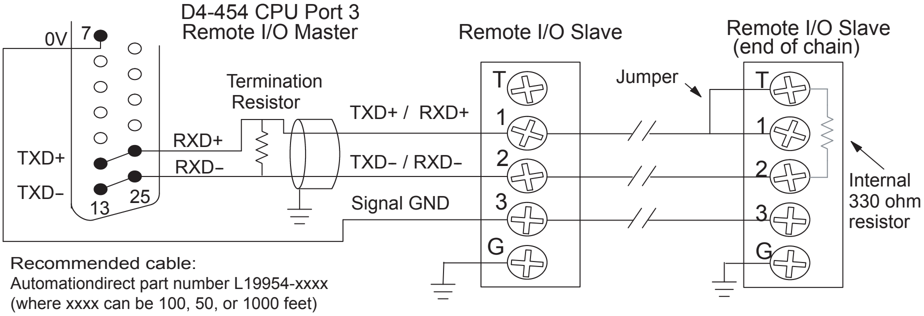
The next step is to make the connections between all devices on the Remote I/O link.

The location of Port 3 on the D4-454 is on the 25-pin connector, as pictured to the right. Remember that ports 1 and 3 are "logical" ports that share the 25-pin connector. Port 3 is an RS-422 non-isolated port. The pin assignments are:

Pin 7	Signal GND
Pin 12	TXD+
Pin 13	TXD-
Pin 24	RXD+
Pin 25	RXD-



Now we are ready to discuss wiring the D4-454 to the remote slaves on the remote base(s). The remote I/O link is a 3-wire, half duplex type. Since Port 3 of the D4-454 CPU is a 5-wire full duplex-capable port, we must jumper its transmit and receive lines together as shown below (converting it to 3-wire half-duplex).



Recommended cable:
Automationdirect part number L19954-xxxx
(where xxxx can be 100, 50, or 1000 feet)
Belden 9842 or equivalent

The twisted/shielded pair connects to the D4-454 Port 3 as shown. Be sure to connect the cable shield wire to the signal ground connection. A termination resistor must be added externally to the CPU, as close as possible to the connector pins. Its purpose is to minimize electrical reflections that occur over long cables. Be sure to add the jumper at the last slave to connect the required internal termination resistor

For more details and information on Remote I/O link wiring, please refer to the Remote I/O link user manual D4-REMIO-M available for free download from: www.automationdirect.com.

Configure Remote I/O Slaves

After configuring the D4-454 CPUs Port 3 and wiring it to the remote slave(s), use the following checklist to complete the configuration of the remote slaves. Full instructions for these steps are in the Remote I/O manual.

Set the baud rate DIP switches to match CPUs Port 3 setting.

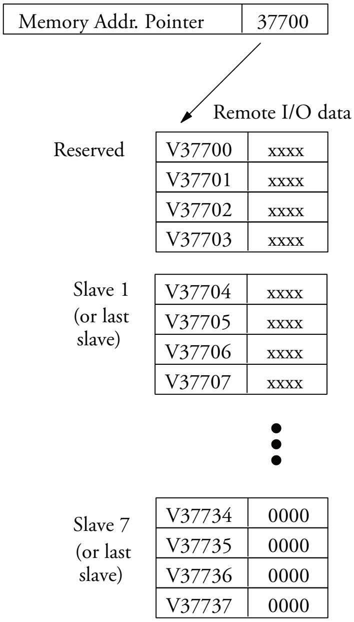
Select a station address for each slave, from 1 to 7. Each device on the remote link must have a unique station address. There can be only one master (address 0) on the remote link.

If you're familiar with configuring remote bases, then you'll recall the fixed table location in V-memory (V7404 - V7477) to configure up to two remote I/O channels. However, we use a separate table for configuring the D4-454 CPUs built-in-remote I/O channel. You will still need the table at V7404 to configure any Remote Master modules.

Configuring the Remote I/O Table

The beginning of the configuration table for the built-in remote I/O channel is the memory address we selected in the Port 3 setup.

The table consists of blocks of four words which correspond to each slave in the system, as shown to the right. The first four table locations are reserved.



Memory Addr. Pointer	37700

Remote I/O data

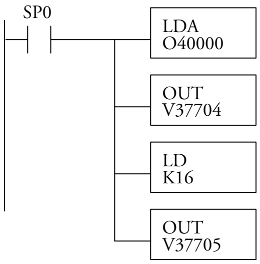
Reserved	V37700	xxxx
	V37701	xxxx
	V37702	xxxx
	V37703	xxxx
Slave 1 (or last slave)	V37704	xxxx
	V37705	xxxx
	V37706	xxxx
	V37707	xxxx
	⋮	
Slave 7 (or last slave)	V37734	0000
	V37735	0000
	V37736	0000
	V37737	0000

The CPU reads data from the table just after power up, interpreting the four data words in each block with these meanings:

- Starting address of slave's input data
- Number of slave's input points
- Starting address of outputs in slave
- Number of slave's output points

The table is 32 words long. If your system has fewer than seven remote slave bases, then the remainder of the table must be filled with zeros. For example, a 3-slave system will have a remote configuration table containing 4 reserved words, 12 words of data and 16 words of "0000".

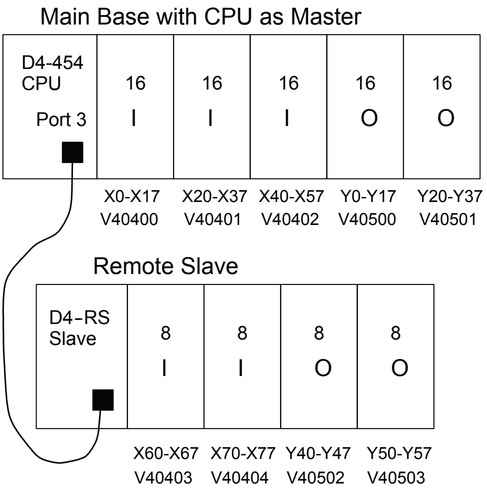
A portion of the ladder program must configure this table (just once) at power up. Use the LDA instruction as shown to the right, to load an address to place in the table. Use the regular LD constant to load the number of slave's input or output points.



The D4-REMIO-M manual contains thorough examples for configuring the table at V7404, which you can adapt for this table as well. The following page gives a shorter program example for one slave.

Consider the simple system featuring the Remote I/O shown below. The D4-454 built-in Remote I/O channel connects to one slave base, which we will assign a station address = 1. The baud rates on the master and slave will be 38400kB.

We can map the remote I/O points as any type of I/O point, simply by choosing the appropriate range of V-memory. Remember that on the D4-454, you have both GX and GY data types available. Since we have plenty of standard I/O addresses available (X and Y), we will have the remote I/O points start at the next X and Y addresses after the main base points (X60 and Y40, respectively).



Remote Slave Worksheet

Remote Base Address 1 (Choose 1-7)

Slot Number	Module Name	INPUT Input Addr.	INPUT No. Inputs	OUTPUT Output Addr.	OUTPUT No. Outputs
0	08ND3S	X060	8		
1	08ND3S	X070	8		
2	08TD1			Y040	8
3	08TD1			Y050	8
4					
5					
6					
7					

Input Bit Start Address: X060 V-Memory Address:V 40403

Total Input Points 16

Output Bit Start Address: Y040 V-Memory Address:V 40502

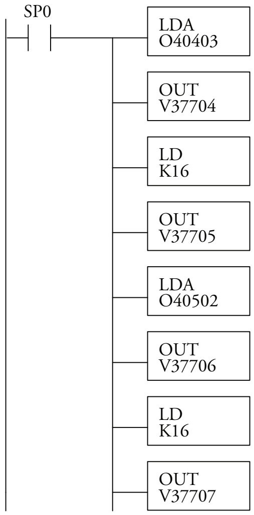
Total Output Points 16

Remote I/O Setup Program

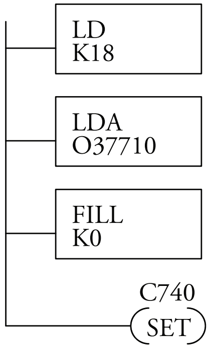
Using the Remote Slave worksheet shown above can help organize our system data in preparation for writing our ladder program (a blank full-page copy of this worksheet is in Appendix A of the D4-REMIO-M manual for your use and duplication). The four key parameters we need to place in our Remote I/O configuration table is in the lower right corner of the worksheet. You can determine the address values by using the memory map given at the end of Chapter 3, CPU Specification and Operation.

The program segment required to transfer our worksheet results to the Remote I/O configuration table is shown to the right. Remember to use the LDA or LD instructions appropriately.

The next page covers the remainder of the required program to get this remote I/O link up and running.



When configuring a Remote I/O channel for fewer than 7 slaves, we must fill the remainder of the table with zeros. This is necessary because the CPU will try to interpret any non-zero number as slave information.

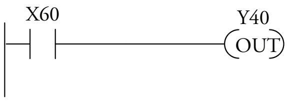


We continue our setup program from the previous page by adding a segment which fills the remainder of the table with zeros. The easiest way is to use the fill command as shown. The example to right fills zeros for slave numbers 2–7, which do not exist in our example system

(6 bases x 4 = 24 locations, = 18 hex).

On the last rung in the example program above, we set a special relay contact C740. This particular contact indicates to the CPU that the ladder program has just finished specifying a remote I/O system. At that moment the CPU begins remote I/O communications. Be sure to include this contact after any Remote I/O setup program.

Remote I/O Test Program



Now we can verify the remote I/O link and setup program operation. A simple quick check can be done with just one rung of ladder, shown to the right. It connects the first input of the remote base with the first output. After placing the PLC in RUN mode, we can go to the remote base and activate its first input, then its first output should turn on.

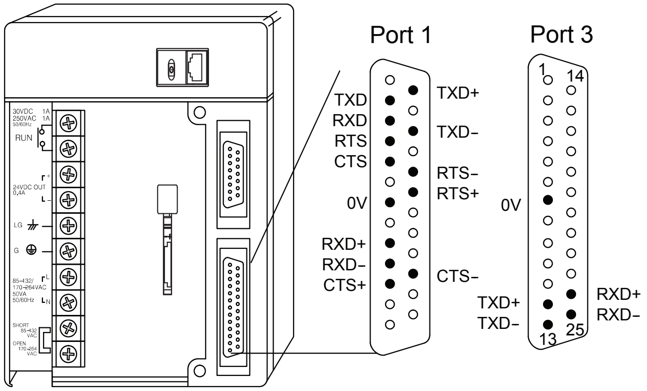
Network Connections to MODBUS® and DirectNet

Configuring the CPU's Comm Ports

This section describes how to configure the CPU's built-in networking port for either MODBUS or DirectNET. This will allow you to connect the D4-454 PLC system directly to MODBUS networks using the RTU protocol, or to other devices on a DirectNET network. MODBUS hosts systems on the network must be capable of issuing the MODBUS commands to read or write the appropriate data. For details on the MODBUS protocol, please refer to Gould MODBUS protocol reference Guide (P1-MBUS-300 Rev. J). In the event a more recent version is available, check with your MODBUS supplier before ordering the documentation. For more details on DirectNet, order our DirectNET manual, part number DA-DNET-M.

NOTE: *For information about the MODBUS protocol, visit the Modbus.org website. The PI-MBUS-300 Rev J Modicon Modbus Protocol Reference Guide can be downloaded from: Modbus.org/doc/PI_MBUS_300.pdf. For more information about DirectNET protocol, order our DirectNET user manual, part number DA-DNET-M, or download it free from our website at: www.automationdirect.com. Select Manuals/Docs > Product Manuals > Miscellaneous >DA-DNET-M.*

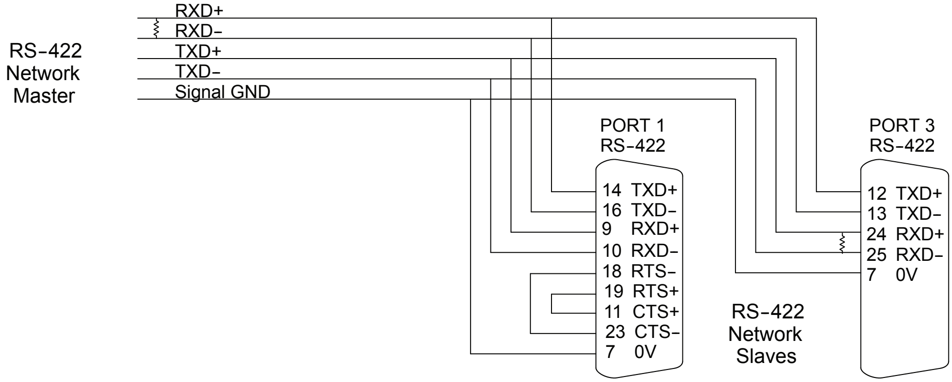
The D4-454 CPUs Port 1, Port 2 and Port 3 can operate as a master or slave for both MODBUS and DirectNET. Port 1 has RS-232 and RS-422 signal levels available on separate pins, Port 2 uses RS-232 signal levels, and Port 3 uses RS422 signal levels. Ports 1 and Port 3 on the D4-454 share the 25-pin D-shell connector, as shown below. You can not simultaneously use Port 1's RS-232 signals and its RS-422 signals.

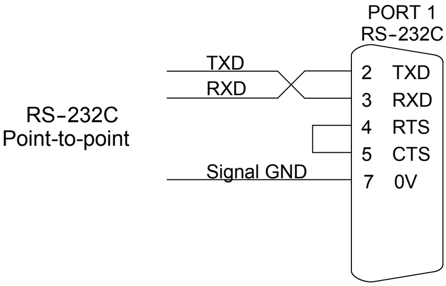


NOTE: *The recommended cable for RS-232 or RS-422 is AutomationDirect part number L19772-xxxx, where xxxx can be 100, 500 or 1000 feet (equivalent to Belden 8102).*

You will need to determine whether the network connection is a 3-wire RS-232 type, or a 5-wire RS-422 type. Normally, we use RS-232 signals for shorter distances (15 meters max.), for communications between just two devices. Use RS-422 signals for longer distances (1000 meters max.) and for multi-drop networks (from 2 to 248 devices). Be sure to use termination resistors at the both ends of RS-422 network wiring, matching the impedance rating of the cable (between 100 and 500 ohms).

NOTE: *If the D4-454 is to be used as a MODBUS Master and the distance will be more than 1,000 feet, you can use the MODBUS Network Master module, F4-MAS-MB, and use the RS-485 port. See the module on our website, automationdirect.com for more details.*

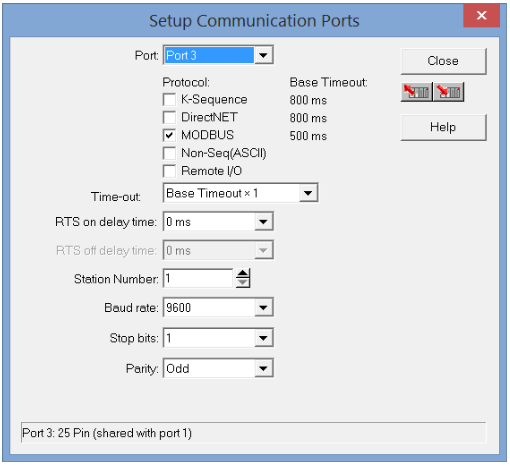




MODBUS Port Configuration

In DirectSOFT, choose the PLC menu, then Setup > Secondary Comm Port.

- **Port:** From the port number list box, choose Port 1, 2 or 3.
- **Protocol:** Click the box to the left of Modbus to select it. The dialog below will appear.



Setup Communication Ports

Port: Port 3

Protocol:	Base Timeout:
□ K-Sequence	800 ms
□ DirectNET	800 ms
☑ MODBUS	500 ms
□ Non-Seq(ASCII)	
□ Remote I/O	

Time-out: Base Timeout × 1

RTS on delay time: 0 ms

RTS off delay time: 0 ms

Station Number: 1

Baud rate: 9600

Stop bits: 1

Parity: Odd

Port 3: 25 Pin (shared with port 1)

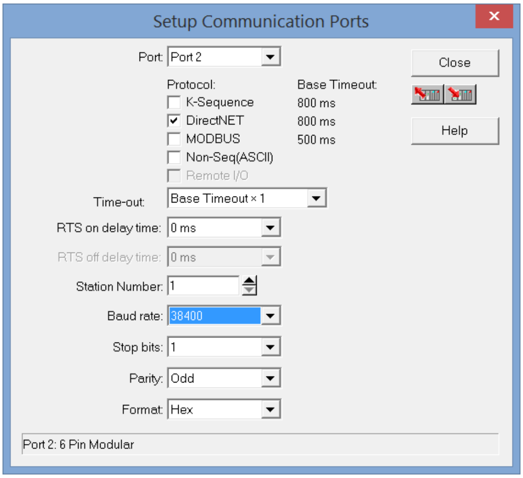
- **Timeout:** the amount of time the port will wait after it sends a message to get a response before logging an error.
- **RTS ON / OFF Delay Time:** The RTS ON Delay Time specifies the time the D4-454 waits to send the data after it has raised the RTS signal line. The RTS OFF Delay Time specifies the time the D4-454 waits to release the RTS signal line after the data has been sent.
- **Station Number:** For making the CPU port a MODBUS master, choose “1”. The possible range for MODBUS slave numbers is from 1 to 90. Each slave must have a unique number. At power up, the port is automatically a slave, unless and until the D4-454 executes ladder logic instructions which attempt to use the port as a master. Thereafter, the port reverts back to slave mode until ladder logic uses the port again.
- **Baud Rate:** The available baud rates include 2400, 4800, 9600, 19200, and 38400 baud. Choose a higher baud rate initially, reverting to lower baud rates if you experience data errors or noise problems on the network. Important: You must configure the baud rates of all devices on the network to the same value. Refer to the appropriate product manual for details.
- **Stop Bits:** Choose 1 or 2 stop bits for use in the protocol.
- **Parity:** Choose none, even, or odd parity for error checking.

Then click the “write to PLC” button to send the Port configuration to the CPU, and click Close.

DirectNET Port Configuration

In DirectSOFT, choose the PLC menu, then Setup > Secondary Comm Port.

- **Port:** From the port number list box, choose Port 1, 2 or 3.
- **Protocol:** Click the box to the left of DirectNET to select it. The dialog below will appear.



Setup Communication Ports

Port: Port 2

Protocol:	Base Timeout:
☐ K-Sequence	800 ms
☑ DirectNET	800 ms
☐ MODBUS	500 ms
☐ Non-Seq(ASCII)	
☐ Remote I/O	

Time-out: Base Timeout x 1

RTS on delay time: 0 ms

RTS off delay time: 0 ms

Station Number: 1

Baud rate: 38400

Stop bits: 1

Parity: Odd

Format: Hex

Port 2: 6 Pin Modular

- **Timeout:** the amount of time the port will wait after it sends a message to get a response before logging an error.
- **RTS ON / OFF Delay Time:** The RTS ON Delay Time specifies the time the D4-454 waits to send the data after it has raised the RTS signal line. The RTS OFF Delay Time specifies the time the D4-454 waits to release the RTS signal line after the data has been sent.
- **Station Number:** For making the CPU port a DirectNET master, choose “1”. The allowable range for DirectNET slaves is from 1 to 90 (each slave must have a unique number). At power up, the port is automatically a slave, unless and until the D4-454 executes ladder logic instructions which attempt to use the port as a master. Thereafter, the port reverts back to slave mode until ladder logic uses the port again.
- **Baud Rate:** The available baud rates include 2400, 4800, 9600, 19200, and 38400 baud. Choose a higher baud rate initially, reverting to lower baud rates if you experience data errors or noise problems on the network. Important: You must configure the baud rates of all devices on the network to the same value.
- **Stop Bits:** Choose 1 or 2 stop bits for use in the protocol.
- **Parity:** Choose none, even, or odd parity for error checking.
- **Format:** Choose between hex or ASCII formats.

Then click the “write to PLC” button to send the Port configuration to the CPU, and click Close.

Network Slave Operation

This section describes how other devices on a network can communicate with a CPU port that you have configured as a DirectNET slave or MODBUS slave. A MODBUS host must use the MODBUS RTU protocol to communicate with the D4-454 as a slave. The host software must send a MODBUS function code and MODBUS address to specify a PLC memory location the D4-454 comprehends. The DirectNET host just uses normal I/O addresses to access the D4-454. No CPU ladder logic is required to support either MODBUS slave or DirectNET slave operation.

NOTE: *For more information on DirectNET proprietary protocol, see the DirectNET reference manual, DA-DNET-M, available on our website.*

MODBUS Function Codes Supported

The MODBUS function code determines whether the access is a read or a write, and whether to access a single data point or a group of them. The D4-454 supports the MODBUS function codes described below.

MODBUS Function Code	Function	D4-454 Data Types Available
01	Read a group of coils	Y, CR, T, CT, GY
02	Read a group of inputs	X, SP, GX
05 (slave only)	Set / Reset a single coil	Y, CR, T, CT
15	Set / Reset a group of coils	Y, CR, T, CT
03, 04	Read a value from one or more registers	V
06 (slave only)	Write a value into a single register	V
16	Write a value into a group of registers	V

MODBUS Data Types Supported

The memory types in a D4-454 system include X input, Y output, C control relay, V-memory data registers, etc. MODBUS uses differently named data types. So, you will need to determine which MODBUS data types corresponds to any desired PLC memory location by using the cross-reference table below.

D4-454 Memory Type	Quantity (Decimal)	PLC Range (Octal)	Corresponding MODBUS Data Type	RX Function Code
Inputs (X)	1024	X0 – X1777	Input	02
Global Inputs (GX)	1536	GX0 – GX2777	Input	02
Special Relays (SP)	512	SP0 – SP137 SP320 – SP717	Input	02
Outputs (Y)	1024	Y0 – Y1777	Coil	01
Global Outputs (GY) *	1536	GY0 – GY2777	Coil	01
Control Relays (CR)	2048	C0 – C3777	Coil	01
Timer Contacts (T)	256	T0 – T377	Coil	01
Counter Contacts (CT)	256	CT0 – CT377	Coil	01
Stage Status Bits (S)	1024	S0 – S1777	Coil	01
Timer Current Values (V)	256	V0 – V377	Input Register	03
Counter Current Values (V)	256	V1000 – V1377	Input Register	03
V-Memory, user data (V)	3072 12288	V1400 – V7377 V10000 – V37777	Holding Register	03
V-Memory, system (V)	320	V700 – V777 V7400 – V7777	Holding Register	03

Determining the MODBUS Address

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

- By specifying the MODBUS data types and address
- By specifying a MODBUS address only

If Your Host Software Requires the Data Type and Address

Many host software packages allow you to specify the MODBUS data type and the MODBUS address that corresponds to the PLC memory location. This is the easiest method, but not all packages allow you to do it this way. The various MODBUS data types were presented earlier, but they have been included again in the following table.

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

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

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

D4-454 Memory Type	QTY (Decimal)	PLC Range (Octal)	MODBUS Address Range (Decimal)	MODBUS Data Type
For Discrete Data Types	Convert PLC Address to Decimal		+ Start of Range	+ Data
Inputs (X)	1024	X0-X1777	2048-3071	Input
Special Relays (SP)	512	SP0-SP137 SP320-SP717	3072-3167 3280-3535	Input
Outputs (Y)	1024	Y0-Y1777	2048-3071	Coil
Control Relays (CR)	2048	C0-C3777	3072-5119	Coil
Timer Contacts (T)	256	T0-T377	6144-6399	Coil
Counter Contacts (CT)	256	CT0-CT377	6400-6655	Coil
Stage Status Bits (S)	1024	S0-S1777	5120-6143	Coil
Global Inputs (GX) *	**1536**	**GX0-GX2777**	**0-1535**	Input
Global Outputs (GY) *	**1536**	**GY0-GY2777**	**0-1535**	Coil
For Word Data Types ----------	Convert PLC Address to Decimal		+	Data Type
Timer Current Values (V)	**256**	**V0-V377**	**0-255**	**Input Register**
Counter Current Values (V)	**256**	**V1000-V1377**	**512-767**	**Input Register**
V-Memory, user data (V)	**3072** **12288**	**V1400-V7377** **V10000-V37777**	**768-3839** **4096-16383**	**Holding Register**
V-Memory, system (V)	**320**	**V700-V777** **V7400-V7777**	**448-768** **3480-3735**	**Holding Register**

*** The total of GX and GY global I/O points cannot exceed 1536 points.**

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

Example 1: V2100

Find the MODBUS address for User V location V2100. **PLC Address (Dec.) = Data Type**

1. Find V-memory in the table.
2. Convert V2100 into decimal (1088). V2100 = 1088 decimal
3. Use the MODBUS data type from the table. 1088 + Hold. Reg. = **Holding Reg. 1088**

V-Memory, user date (V)	3072 12288	V4100 - V7377 V10000-V37777	768 - 3839 4096 - 16383	Holding Register

Example 2: Y20

Find the MODBUS address for output Y20

1. Find Y outputs in the table. **PLC Address (Dec.) + Start Addr. + Data Type**
2. Convert Y20 into decimal (16).
3. Add the starting address for the range (2048). Y20 = 16 decimal

4 Use the MODBUS data type from the table. 16 + 2048 = **Coil 2064**

Outputs (Y)	1024	Y0 - Y1777	2048 - 3071	Coil

Example 3: T10 Current Value

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

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

PLC Address (Dec.) + Data Type
T10 = 8 decimal
8 + Input Reg = Input Reg. 8

Timer Current Values (V)	256	V0 - V3777	0 - 255	Input Register

Example 4: C54

Find the MODBUS address for Control Relay C54

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

PLC Address (Dec.) + Start Addr. + Data Type
V2100 = 1088 decimal
1088 + Hold. Reg. = Holding Reg. 1088

Control Relays (CR)	2048	C0 - C3777	3072 - 5119	Coil

If Your MODBUS Host Software Requires an Address Only

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

- 484 mode
- 584/984 mode

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

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

In either case, you simply convert the PLC octal address to decimal and add the appropriate MODBUS address (as required). The table on the next page shows the exact equation used for each group of data.

Discrete Data Types				
Memory Type	PLC Range (Octal)	Address (484 Mode)	Address (585/984 Mode)	Data Type
Global Inputs (GX)	GX0 – GX1746	1001 – 1999	10001 – 10999	Input
	GX1747 – GX3777	–	11000 – 12048	Input
Inputs (X)	X0 – X1777	–	12049 – 13072	Input
Special Relays (SP)	SP0 – SP777	–	13073 – 13584	Input
Global Outputs (GY)	GY0 – GY3777	1 – 2048	1 – 2048	Output
Outputs (Y)	Y0 – Y1777	2049 – 3072	2049 – 3072	Output
Control Relays (CR)	C0 – C3777	3073 – 5120	3073 – 5120	Output
Timer Contacts (T)	T0 – T377	6145 – 6400	6145 – 6400	Output
Counter Contacts (CT)	CT0 – CT377	6401 – 6656	6401 – 6656	Output
Stage Status Bits (S)	S0 – S1777	5121 – 6144	5121 – 6144	Output

Word Data Types			
Registers	PLC Range (Octal)	Input/Holding (484 Mode)*	Input/Holding (585/984 Mode)*
V – memory (Timers)	V0 – V377	3001 / 4001	30001 / 40001
V – memory (Counters)	V1000 – V1177	3513 / 4513	30513 / 40513
V – memory (Data Words)	V1200 – V1377	3641 / 4641	30641 / 40641
V – memory (Data Words)	V1400 – V1746	3769 / 4769	30769 / 40769
V – memory (Data Words)	V1747 – V1777	–	31000 / 41000
V – memory (Data Words)	V2000 – V7377	–	41025
V – memory (Data Words)	V10000 – V17777	–	44097
* Modbus Function 04			

NOTE: *For an automated MODBUS/KOYO address conversion utility, go to our website, www.automationdirect.com, and download the EXCEL file: modbus_conversion.xls located at: Tech Support > Technical and Application Notes > AN-MISC-010.*

Example 1: V2100 584/984 Mode

Find the MODBUS address for User V location V2100. PLC Address (Dec.) + Mode Address

1. Find V-memory in the table. V2100 = 1088 decimal
2. Convert V2100 into decimal (1088). 1088 + 40001 = $\boxed{41089}$
3. Add the MODBUS starting address for the mode (40001).

For Word Data Types....	PLC Address (Dec.)	+	Appropriate Mode Address			
Timer Current Values (V)	128	V0 – V177	0 – 127	3001	30001	Input Register
Counter Current Values (V)	128	V1200 – V7377	640 – 3839	3001	30001	Input Register
V-memory, user data (V)	1024	V2000 – V3777	1024 – 2047	4001	40001	Holding Register

Example 2: Y20 584/984 Mode

Find the MODBUS address for output Y20. PLC Addr. (Dec.) + Start Address + Mode

1. Find Y outputs in the table.
2. Convert Y20 into decimal (16). Y20 = 16 decimal
3. Add the starting address for the range (2048). 16 + 2048
4. Add the MODBUS address for the mode (1). 16 + 2048 + 1 = $\boxed{2065}$

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

Example 3: T10 Current Value 484 Mode

Find the MODBUS address to obtain the PLC Address (Dec.) + Mode
Address
current value from Timer T10.

1. Find Timer Current Values in the table.
2. Convert T10 into decimal (8). T10 = 8 decimal
3. Add the MODBUS starting address for the mode (3001). 8 + 3001 = $\boxed{3009}$

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

Example 4: C54 584/984 Mode

Find the MODBUS address for Control Relay C5. PLC Addr. (Dec.) + Start Address + Mode

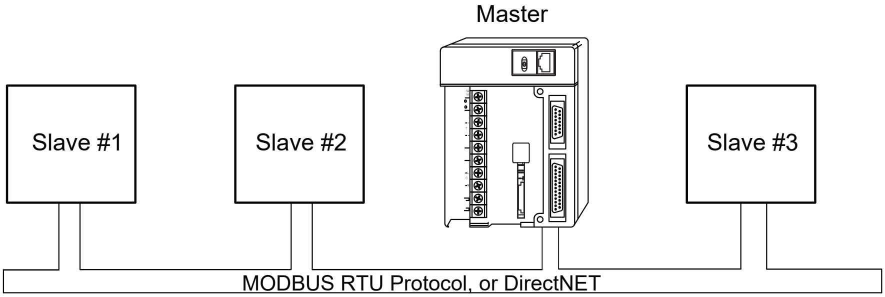
1. Find Control Relays in the table.
2. Convert C54 into decimal (44). C54 = 44 decimal
3. Add the starting address for the range (3072). 44 + 3072
4. Add the MODBUS address for the mode (1). 44 + 3072 + 1 = $\boxed{3117}$

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

Network Master Operation

This section describes how the D4-454 can communicate on a MODBUS or DirectNET network as a master. For MODBUS networks, it uses the MODBUS RTU protocol, which must be interpreted by all the slaves on the network. Both MODBUS and DirectNet are single master/multiple slave networks. The master is the only member of the network that can initiate requests on the network. This section teaches you how to design the required ladder logic for network master operation.

When using the D4-454 CPU as the master station, you use simple RLL instructions to initiate the requests. The WX instruction initiates network write operations, and the RX instruction initiates network read operations. Before executing either the WX or RX commands, we will need to load data related to the read or write operation onto the CPU's accumulator stack. When the WX and RX instruction executes, it uses the information on the stack combined with data in the instruction box to completely define the task, which goes to the port.



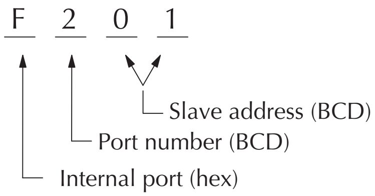
It's possible to use Ports 1, 2 and 3 for either MODBUS or DirectNET, and it is possible to use all three ports at the same time. You must tell the WX and RX instructions the intended port for each communications transaction.

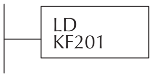
To summarize, the RLL instructions identify the following items.

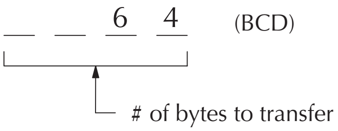
1. Port number on the master (Port 1, 2 or 3), and the slave station address (LD instruction).
2. Amount of data (in bytes) you want to transfer (LD instruction).
3. Area of memory to be used by the master (LDA instruction).
4. Area of CPU V-memory to be used in communication with the slave and whether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

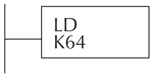
Step 1: Identify Master Port # and Slave

The first Load (LD) instruction identifies the communications port number on the network master (D4-454) and the address of the slave station. This instruction can address up to 99 MODBUS slaves, or 90 DirectNET slaves. The format of the word is shown to the right. The “F” in the upper nibble indicates the port is internal to the CPU (and not in a slot in the base). The second nimble indicates the port number, 1, 2, or 3. In this example, the “F2” in the upper byte indicates the CPU will use the RJ12 port on the CPU, port number 2. The lower byte contains the slave address number in BCD (01 to 99).









Step 2: Load Number of Bytes to Transfer

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

The number of bytes specified also depends on the type of data you want to obtain. For example, the D4-454 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 D4-454 CPUs.

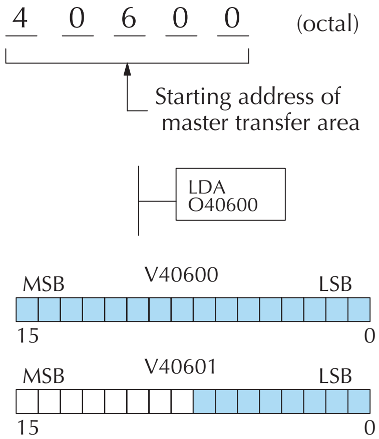
D4-454 Memory	Bits per unit	Bytes
V-memory *T / C current value*	16	2
Inputs (X, SP)	8	1
Outputs *(Y, C, Stage, T/C bits)*		
Scratch Pad Memory		
Diagnostic Status		

Step 3: Specify Master Memory Area

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

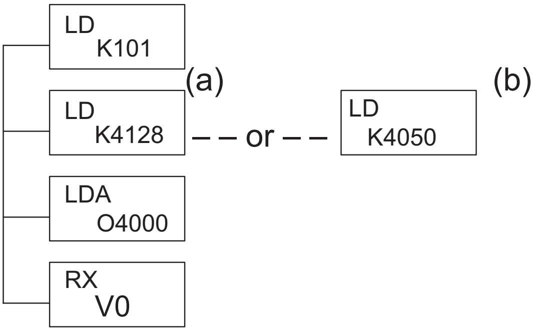
For a WX instruction, the D4-454 CPU sends the number of bytes previously specified from its memory area beginning at the LDA address specified.

For an RX instruction, the D4-454 CPU reads the number of bytes previously specified from the slave, placing the received data into its memory area beginning at the LDA address specified.



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.*

When using MODBUS, the RX instructions uses function 3 by default, to read MODBUS holding registers (Address 40001). The DL05/06, DL250-1/260, DL350, D4-454 support function 04, read input register (Address 30001). To use function 04, put the number ‘4’ into the most significant position (4xxx) of the total number of bytes. Four digits must be entered for the instruction to work properly with this mode.



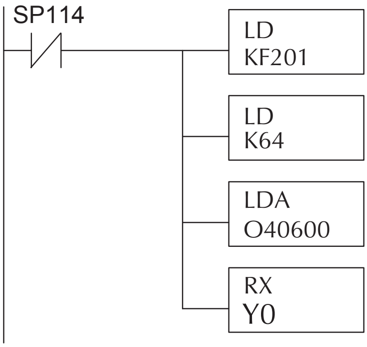
The (a) K4128 indicates the instruction will read 128 bytes of MODBUS input registers (30001). The (b) K4050 indicates the instruction will read 50 bytes of MODBUS input registers (30001). The value of 4 in the most significant position will cause the RX to use MODBUS function 4 (30001 range).

Step 4: Specify Slave Memory Area

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

The RX instruction reads data from the slave starting at the address specified. The WX instruction writes data to the slave starting at the address specified.

- ***DirectNET*** slaves – specify the same address in the WX and RX instruction as the slave's native I/O address
- MODBUS DL405, DL205, or DL06 slaves – specify the same address in the WX and RX instruction as the slave's native I/O address

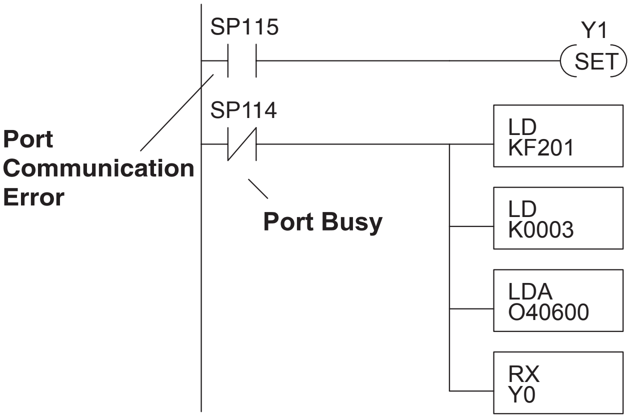


Communications from a Ladder Program

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

Port 2, which can be a master, has two Special Relay contacts associated with it (see the Special Relays Appendix for more details). One indicates "Port busy"(SP114), and the other indicates "Port Communication Error"(SP115). The example above shows the use of these contacts for a network master that only reads a device (RX). The "Port Busy" bit is on while the PLC communicates with the slave. When the bit is off the program can initiate the next network request.

The "Port Communication Error" bit turns on when the PLC has detected an error. Use of this bit is optional. When used, it should be ahead of any network instruction boxes since the error bit is reset when an RX or WX instruction is executed.

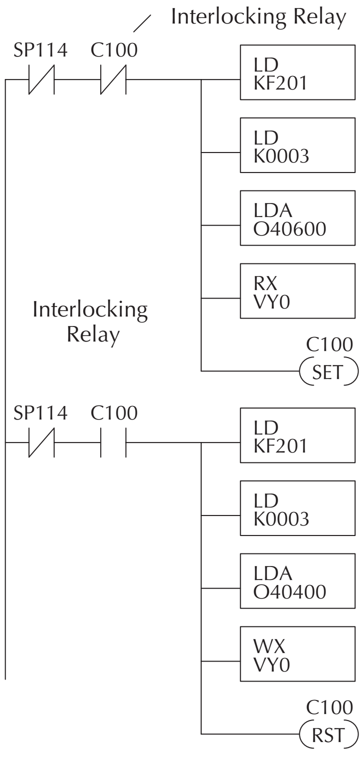


Multiple Read and Write Interlocks

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

In the example below, after the RX instruction is executed, C100 is set. When the port has finished the communication task, the second routine is executed and C100 is reset.

If you're using RLLPLUS Stage Programming, you can put each routine in a separate program stage to ensure proper execution and switch from stage to stage allowing only one of them to be active at a time.


Interlocking Relay
SP114
C100
LD
KF201
LD
K0003
LDA
O40600
RX
VY0
C100
SET
Interlocking
Relay
SP114
C100
LD
KF201
LD
K0003
LDA
O40400
WX
VY0
C100
RST