CPU Specifications and Operation

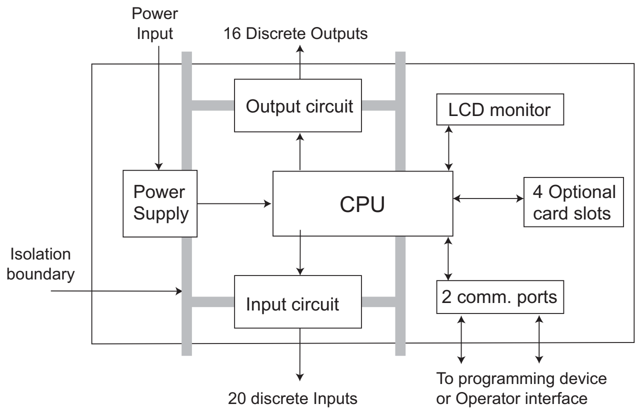
Chapter 4

In This Chapter

Introduction

The Central Processing Unit (CPU) is the heart of the Micro PLC. Almost all PLC operations are controlled by the CPU, so it is important that it is set up correctly. This chapter provides the information needed to understand:

- Steps required to set up the CPU
- Operation of ladder programs
- Organization of Variable Memory



Note: *The High-Speed I/O function (HSIO) consists of dedicated but configurable hardware in the DL06. It is not considered part of the CPU, because it does not execute the ladder program. For more on HSIO operation, see Chapter 3.*

DL06 CPU Features

The DL06 Micro PLC has 14.8K words of memory comprised of 7.6K of ladder memory and 7.6K words of V-memory (data registers). Program storage is in the FLASH memory which is a part of the CPU board in the PLC. In addition, there is RAM with the CPU which will store system parameters, V-memory, and other data which is not in the application program. The RAM is backed up by a “super-capacitor”, storing the data for several hours in the event of a power outage. The capacitor automatically charges during powered operation of the PLC.

The DL06 supports fixed I/O which includes twenty discrete input points and sixteen output points.

Over 220 different instructions are available for program development as well as extensive internal diagnostics that can be monitored from the application program or from an operator interface. Chapters 5, 6, and 7 provide detailed descriptions of the instructions.

The DL06 provides two built-in communication ports, so you can easily connect a handheld programmer, operator interface, or a personal computer without needing any additional hardware.

CPU Specifications

Specifications	
Feature	**DL06**
Total Program memory (words)	14.8K
Ladder memory (words)	7680
Total V-memory (words)	7616
User V-memory (words)	7488
Non-volatile V Memory (words)	128
Contact execution (boolean)	2.0uS
Typical scan (1k boolean)	3 - 4mS
RLL Ladder style Programming	Yes
RLL and RLLPLUS Programming	Yes
Run Time Edits	Yes
Scan	Variable / fixed
Handheld programmer	Yes
***Direct*SOFT32 programming for Windows.**	Yes
Built-in communication ports (RS232C)	Yes
FLASH Memory	Standard on CPU
Local Discrete I/O points available	36
Local Analog input / output channels maximum	None
High-Speed I/O (quad., pulse out, interrupt, pulse catch, etc.)	Yes, 2
I/O Point Density	20 inputs, 16 outputs
Number of instructions available (see Chapter 5 for details)	229
Control relays	1024
Special relays (system defined)	512
Stages in RLLPLUS	1024
Timers	256
Counters	128
Immediate I/O	Yes
Interrupt input (external / timed)	Yes
Subroutines	Yes
For/Next Loops	Yes
Math (Integer and floating point)	Yes
Drum Sequencer Instruction	Yes
Time of Day Clock/Calendar	Yes
Internal diagnostics	Yes
Password security	Yes
System error log	Yes
User error log	Yes
Battery backup	Optional D2-BAT-1 available (not included with unit)

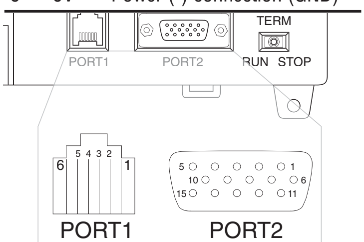
CPU Hardware Setup

Communication Port Pinout Diagrams

Cables are available that allow you to quickly and easily connect a Handheld Programmer or a personal computer to the DL06 PLCs. However, if you need to build your own cables, use the pinout diagrams shown. The DL06 PLCs require an RJ-12 phone plug for port 1 and a 15-pin svga dsub for port 2.

The DL06 PLC has two built-in serial communication ports. Port 1 (RS232C only) is generally used for connecting to a D2-HPP, ***Direct***SOFT32, operator interface, MODBUS slave only, or a DirectNET slave only. The baud rate is fixed at 9600 baud for port 1. Port 2 (RS232C/RS422/RS485) can be used to connect to a D2-HPP, ***Direct***SOFT32, operator interface, MODBUS master/slave, DirectNET master/slave or ASCII in/out. Port 2 has a range of speeds from 300 baud to 38.4K baud.

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



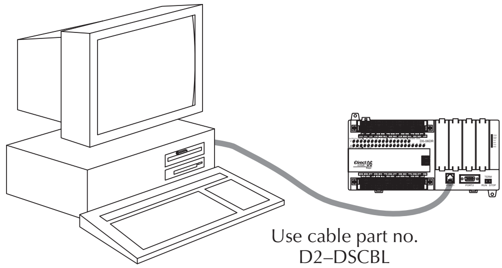
Port 2 Pin Descriptions		
1	5V	Power (+) connection
2	TXD	Transmit data (RS-232C)
3	RXD	Receive data (RS-232C)
4	RTS	Ready to send
5	CTS	Clear to send
6	RXD-	Receive data (-) (RS-422/485)
7	0V	Power (-) connection (GND)
8	0V	Power (-) connection (GND)
9	TXD+	Transmit data (+) (RS-422/485)
10	TXD-	Transmit data (-) (RS-422/485)
11	RTS+	Ready to send (+) (RS-422/485)
12	RTS-	Ready to send (-) (RS-422/485)
13	RXD+	Receive data (+) (RS-422/485)
14	CTS+	Clear to send (+) (RS-422/485)
15	CTS-	Clear to send (-) (RS-422/485)

Communications Port 1	
Com 1	Connects to HPP, DirectSOFT32, operator interfaces, etc.
	6-pin, RS232C
	Communication speed (baud): 9600 (fixed)
	Parity: odd (default)
	Station Address: 1 (fixed)
	8 data bits
	1 start, 1 stop bit
	Asynchronous, half-duplex, DTE
	Protocol: (auto-select) K-sequence (slave only), DirectNET (slave only), MODBUS (slave only)

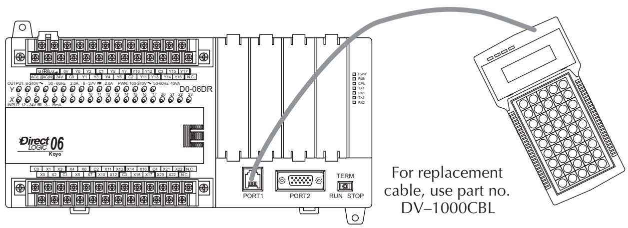
Communications Port 2	
Com 2	Connects to HPP, DirectSOFT32, operator interfaces, etc.
	15-pin, multifunction port, RS232C, RS422, RS485
	Communication speed (baud): 300, 600, 1200, 2400, 4800, 9600, 19200, 38400
	Parity: odd (default), even, none
	Station Address: 1 (default)
	8 data bits
	1 start, 1 stop bit
	Asynchronous, half-duplex, DTE
	Protocol: (auto-select) K-sequence (slave only), DirectNET (master/slave), MODBUS (master/slave), non-sequence/print/ASCII in/out

Connecting the Programming Devices

If you're using a Personal Computer with the *Direct*SOFT32™ programming package, you can connect the computer to either of the DL06's serial ports. For an engineering office environment (typical during program development), this is the preferred method of programming.



The Handheld programmer D2-HPP is connected to the CPU with a handheld programmer cable. This device is ideal for maintaining existing installations or making small program changes. The handheld programmer is shipped with a cable, which is approximately 6.5 feet (200 cm) long.

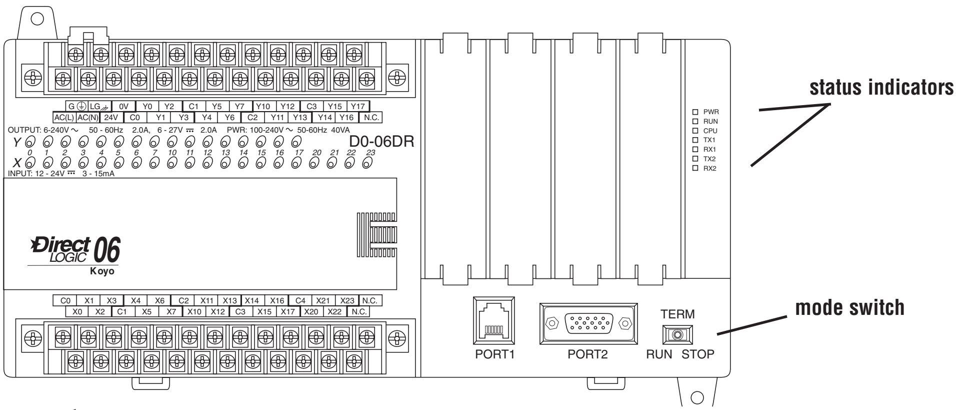


CPU Setup Information

Even if you have years of experience using PLCs, there are a few things you need to do before you can start entering programs. This section includes some basic things, such as changing the CPU mode, but it also includes some things that you may never have to use. Here's a brief list of the items that are discussed:

- Using Auxiliary Functions
- Clearing the program (and other memory areas)
- How to initialize system memory
- Setting retentive memory ranges

The following paragraphs provide the setup information necessary to get the CPU ready for programming. They include setup instructions for either type of programming device you are using. The D2–HPP Handheld Programmer Manual provides the Handheld keystrokes required to perform all of these operations. The *Direct*SOFT32™ Manual provides a description of the menus and keystrokes required to perform the setup procedures via *Direct*SOFT32.



Status Indicators

The status indicator LEDs on the CPU front panels have specific functions which can help in programming and troubleshooting.

Mode Switch Functions

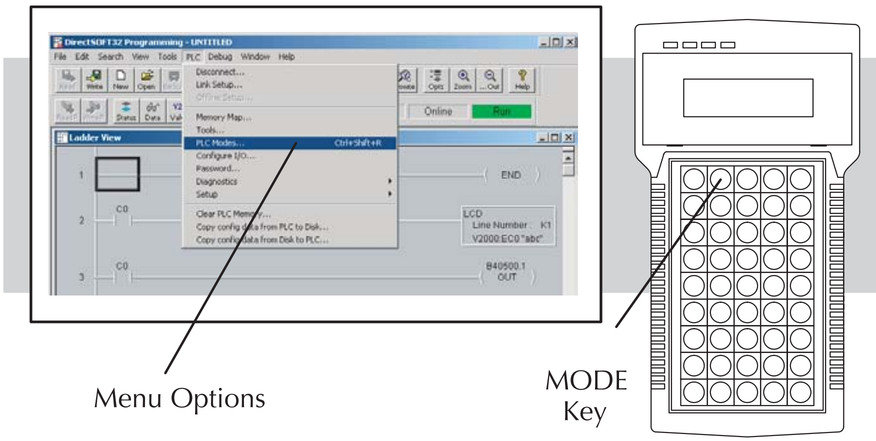
The mode switch on the DL06 PLC provides positions for enabling and disabling program changes in the CPU. Unless the mode switch is in the TERM position, RUN and STOP mode changes will not be allowed by any interface device, (handheld programmer, *Direct*SOFT32 programming package or operator interface). Programs may be viewed or monitored but no changes may be made. If the switch is in the TERM position and no program password is in effect, all operating modes as well as program access will be allowed through the connected programming or monitoring device.

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

Changing Modes in the DL06 PLC

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

There are two ways to change the CPU mode. You can use the CPU mode switch to select the operating mode, or you can place the mode switch in the TERM position and use a programming device to change operating modes. With the switch in this position, the CPU can be changed between Run and Program modes. You can use either ***Direct***SOFT32 or the Handheld Programmer to change the CPU mode of operation. With ***Direct***SOFT32 you use a menu option in the PLC menu. With the Handheld Programmer, you use the MODE key.



Mode of Operation at Power-up

The DL06 CPU will normally power-up in the mode that it was in just prior to the power interruption. For example, if the CPU was in Program Mode when the power was disconnected, the CPU will power-up in Program Mode (see warning note below).

WARNING: Once the super capacitor has discharged, the system memory may not retain the previous mode of operation. When this occurs, the PLC can power-up in either Run or Program Mode if the mode switch is in the term position. There is no way to determine which mode will be entered as the startup mode. Failure to adhere to this warning greatly increases the risk of unexpected equipment startup.

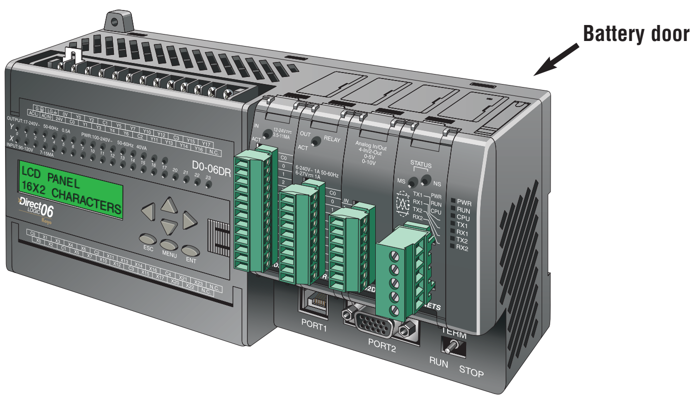
Using Battery Backup

An optional lithium battery is available to maintain the system RAM retentive memory when the DL06 system is without external power. Typical CPU battery life is five years, which includes PLC runtime and normal shutdown periods. However, consider installing a fresh battery if your battery has not been changed recently and the system will be shut down for a period of more than ten days.

*NOTE: Before installing or replacing your CPU battery, back-up your V-memory and system parameters. You can do this by using **Direct**SOFT32 to save the program, V-memory, and system parameters to hard/floppy disk on a personal computer.*

To install the D2–BAT–1 CPU battery in the DL06 CPU:

1. Press the retaining clip on the battery door down and swing the battery door open.
2. Place the battery into the coin–type slot.
3. Close the battery door making sure that it locks securely in place.
4. Make a note of the date the battery was installed



WARNING: Do not attempt to recharge the battery or dispose of an old battery by fire. The battery may explode or release hazardous materials.

Battery Backup

The battery backup is available immediately after the battery has been installed. Retentive memory ranges can be set in the CPU to maintain the data in C, S, T, CT and V memory if the power to the CPU is removed. Low battery indication is enabled by setting bit 12 of V7633 (B7633.12). ***The CPU indicator will blink if the battery is low*** (refer to the table on page 4-6). Special Relay 43 (SP43) will also be set when the battery is low. If the low battery feature is not desired, do not set bit V7633.12.

Auxiliary Functions

Many CPU setup tasks involve the use of Auxiliary (AUX) Functions. The AUX Functions perform many different operations, ranging from clearing ladder memory, displaying the scan time, copying programs to EEPROM in the handheld programmer, etc. They are divided into categories that affect different system parameters. Appendix A provides a description of the AUX functions.

You can access the AUX Functions from ***Direct***SOFT32 or from the D2–HPP Handheld Programmer. The manuals for those products provide step-by-step procedures for accessing the AUX Functions. Some of these AUX Functions are designed specifically for the Handheld Programmer setup, so they will not be needed (or available) with the ***Direct***SOFT32 package. The following table shows a list of the Auxiliary functions for the Handheld Programmer.

AUX 2* — RLL Operations	
21	Check Program
22	Change Reference
23	Clear Ladder Range
24	Clear All Ladders
AUX 3* — V-Memory Operations	
31	Clear V Memory
AUX 4* — I/O Configuration	
41	Show I/O Configuration
42	I/O Diagnostics
44	Power Up I/O Configuration check
45	Select Configuration
46	Configure I/O
AUX 5* — CPU Configuration	
51	Modify Program Name
52	Display/Change Calendar
53	Display Scan Time
54	Initialize Scratchpad
55	Set Watchdog Timer
56	Set Communication Port 2
57	Set Retentive Ranges

58	Test Operations
59	Override Setup
5B	HSIO Configuration
5C	Display Error History
5D	Scan Control Setup
AUX 6* — Handheld Programmer Configuration	
61	Show Revision Numbers
62	Beeper On / Off
65	Run Self Diagnostics
AUX 7* — EEPROM Operations	
71	Copy CPU memory to HPP EEPROM
72	Write HPP EEPROM to CPU
73	Compare CPU to HPP EEPROM
74	Blank Check (HPP EEPROM)
75	Erase HPP EEPROM
76	Show EEPROM Type (CPU and HPP)
AUX 8* — Password Operations	
81	Modify Password
82	Unlock CPU
83	Lock CPU

Clearing an Existing Program

Before you enter a new program, be sure to always clear ladder memory. You can use AUX Function 24 to clear the complete program.You can also use other AUX functions to clear other memory areas.

- AUX 23 — Clear Ladder Range
- AUX 24 — Clear all Ladders
- AUX 31 — Clear V Memory

Initializing System Memory

The DL06 Micro PLC maintains system parameters in a memory area often referred to as the "scratchpad." In some cases, you may make changes to the system setup that will be stored in system memory. For example, if you specify a range of Control Relays (CRs) as retentive, these changes are stored in system memory. AUX 54 resets the system memory to the default values.

WARNING: You may never have to use this feature unless you want to clear any setup information that is stored in system memory. Usually, you'll only need to initialize the system memory if you are changing programs and the old program required a special system setup. You can usually load in new programs without ever initializing system memory.

Remember, this AUX function will reset all system memory. If you have set special parameters such as retentive ranges, etc. they will be erased when AUX 54 is used. Make sure that you have considered all ramifications of this operation before you select it.

Setting Retentive Memory Ranges

The DL06 PLCs provide certain ranges of retentive memory by default. The default ranges are suitable for many applications, but you can change them if your application requires additional retentive ranges or no retentive ranges at all. The default settings are:

Memory Area	DL06	
	Default Range	Available Range
Control Relays	C1000 – C1777	C0 – C1777
V Memory	V400 – V37777	V0 – V37777
Timers	None by default	T0 – T377
Counters	CT0 – CT177	CT0 – CT177
Stages	None by default	S0 – S1777

You can use AUX 57 to set the retentive ranges. You can also use DirectSOFT32. menus to select the retentive ranges. Appendix A contains detailed information about auxiliary functions.

WARNING: The DL06 CPUs do not come with a battery. The super capacitor will retain the values in the event of a power loss, but only for a short period of time, depending on conditions. If the retentive ranges are important for your application, make sure you obtain the optional battery.

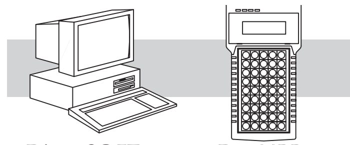
Using a Password

The DL06 PLCs allow you to use a password to help minimize the risk of unauthorized program and/or data changes. Once you enter a password you can “lock” the PLC against access. Once the CPU is locked you must enter the password before you can use a programming device to change any system parameters.

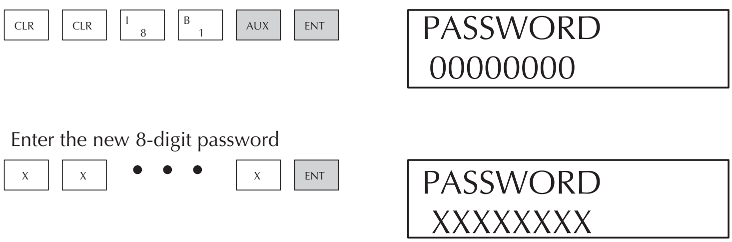
You can select an 8-digit numeric password. The Micro PLCs are shipped from the factory with a password of 00000000. All zeros removes the password protection. If a password has been entered into the CPU you cannot just enter all zeros to remove it. Once you enter the correct password, you can change the password to all zeros to remove the password protection.

WARNING: Make sure you remember your password. If you forget your password you will not be able to access the CPU. The Micro PLC must be returned to the factory to have the password (along with the ladder project) removed.

You can use the D2–HPP Handheld Programmer or ***Direct***SOFT32™. to enter a password. The following diagram shows how you can enter a password with the Handheld Programmer.



DirectSOFT D2–HPP



There are three ways to lock the CPU once the password has been entered.

1. If the CPU power is disconnected, the CPU will be automatically locked against access.
2. If you enter the password with ***Direct***SOFT32, the CPU will be automatically locked against access when you exit ***Direct***SOFT32.
3. Use AUX 83 to lock the CPU.

When you use ***Direct***SOFT32, you will be prompted for a password if the CPU has been locked. If you use the Handheld Programmer, you have to use AUX 82 to unlock the CPU. Once you enter AUX 82, you will be prompted to enter the password.

Note: The DL06 CPUs support multi-level password protection of the ladder program. This allows password protection while not locking the communication port to an operator interface. The multi-level password can be invoked by creating a password with an upper case “A” followed by seven numeric characters (e.g. A1234567).

CPU Operation

Achieving the proper control for your equipment or process requires a good understanding of how DL06 CPUs control all aspects of system operation. There are four main areas to understand before you create your application program:

- CPU Operating System — the CPU manages all aspects of system control. A quick overview of all the steps is provided in the next section.
- CPU Operating Modes — The two primary modes of operation are Program Mode and Run Mode.
- CPU Timing — The two important areas we discuss are the I/O response time and the CPU scan time.
- CPU Memory Map — DL06 CPUs offer a wide variety of resources, such as timers, counters, inputs, etc. The memory map section shows the organization and availability of these data types.

CPU Operating System

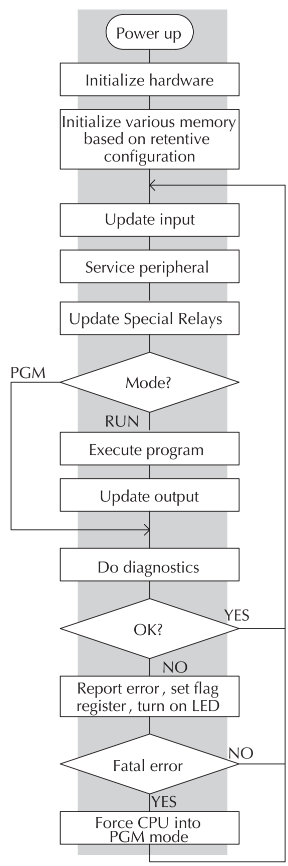
At powerup, the CPU initializes the internal electronic hardware. Memory initialization starts with examining the retentive memory settings. In general, the contents of retentive memory is

preserved, and non-retentive memory is initialized to zero (unless otherwise specified).

After the one-time powerup tasks, the CPU begins the cyclical scan activity. The flowchart to the right shows how the tasks differ, based on the CPU mode and the existence of any errors. The “scan time” is defined as the average time around the task loop. Note that the CPU is always reading the inputs, even during program mode. This allows programming tools to monitor input status at any time.

The outputs are only updated in Run mode. In program mode, they are in the off state.

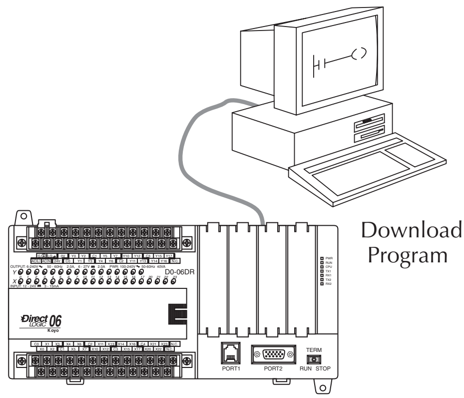
Error detection has two levels. Non-fatal errors are reported, but the CPU remains in its current mode. If a fatal error occurs, the CPU is forced into program mode and the outputs go off.



Program Mode

In Program Mode, the CPU does not execute the application program or update the output points. The primary use for Program Mode is to enter or change an application program. You also use program mode to set up the CPU parameters, such as HSIO features, retentive memory areas, etc.

You can use a programming device, such as *Direct*SOFT32 or the D2–HPP Handheld Programmer to place the CPU in Program Mode.



Run Mode

In Run Mode, the CPU executes the application program and updates the I/O system. You can perform many operations during Run Mode. Some of these include:

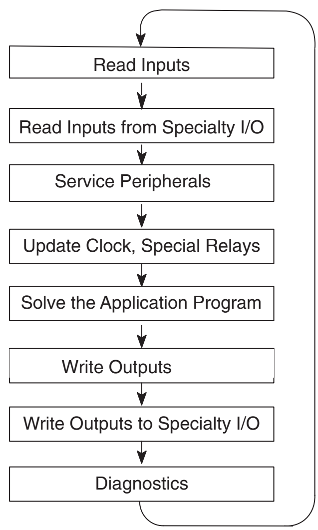
- Monitor and change I/O point status
- Update timer/counter preset values
- Update Variable memory locations

Run Mode operation can be divided into several key areas. For the vast majority of applications, some of these execution segments are more important than others. For example, you need to understand how the CPU updates the I/O points, handles forcing operations, and solves the application program. The remaining segments are not that important for most applications.

You can use *Direct*SOFT32 or the D2–HPP Handheld Programmer to place the CPU in Run Mode.

You can also edit the program during Run Mode. The Run Mode Edits are not "bumpless" to the outputs. Instead, the CPU maintains the outputs in their last state while it accepts the new program information. If an error is found in the new program, then the CPU will turn all the outputs off and enter the Program Mode. This feature is discussed in more detail in Chapter 9.

Normal Run mode scan



WARNING: Only authorized personnel fully familiar with all aspects of the application should make changes to the program. Changes during Run Mode become effective immediately. Make sure you thoroughly consider the impact of any changes to minimize the risk of personal injury or damage to equipment.

Read Inputs

The CPU reads the status of all inputs, then stores it in the image register. Input image register locations are designated with an X followed by a memory location. Image register data is used by the CPU when it solves the application program.

Of course, an input may change after the CPU has just read the inputs. Generally, the CPU scan time is measured in milliseconds. If you have an application that cannot wait until the next I/O update, you can use Immediate Instructions. These do not use the status of the input image register to solve the application program. The Immediate instructions immediately read the input status directly from the I/O modules. However, this lengthens the program scan since the CPU has to read the I/O point status again. A complete list of the Immediate instructions is included in Chapter 5.

Service Peripherals and Force I/O

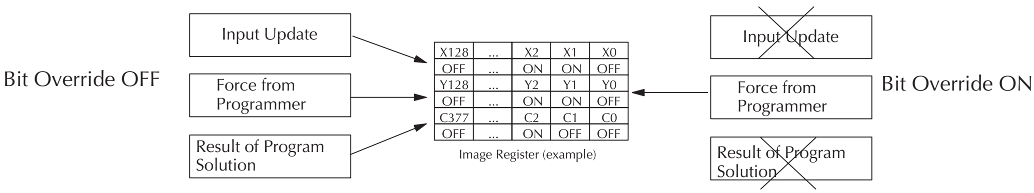
After the CPU reads the inputs from the input modules, it reads any attached peripheral devices. This is primarily a communications service for any attached devices. For example, it would read a programming device to see if any input, output, or other memory type status needs to be modified. There are two basic types of forcing available with the DL06 CPUs:

- Forcing from a peripheral – not a permanent force, good only for one scan
- Bit Override – holds the I/O point (or other bit) in the current state. Valid bits are X, Y, C, T, CT, and S. (These memory types are discussed in more detail later in this chapter).

Regular Forcing — This type of forcing can temporarily change the status of a discrete bit. For example, you may want to force an input on, even though it is really off. This allows you to change the point status that was stored in the image register. This value will be valid until the image register location is written to during the next scan. This is primarily useful during testing situations when you need to force a bit on to trigger another event.

Bit Override — Bit override can be enabled on a point-by-point basis by using AUX 59 from the Handheld Programmer or, by a menu option from within ***Direct***SOFT32™. Bit override basically disables any changes to the discrete point by the CPU. For example, if you enable bit override for X1, and X1 is off at the time, then the CPU will not change the state of X1. This means that even if X1 comes on, the CPU will not acknowledge the change. So, if you used X1 in the program, it would always be evaluated as “off” in this case. Of course, if X1 was on when the bit override was enabled, then X1 would always be evaluated as “on”.

There is an advantage available when you use the bit override feature. The regular forcing is not disabled because the bit override is enabled. For example, if you enabled the Bit Override for Y0 and it was off at the time, then the CPU would not change the state of Y0. However, you can still use a programming device to change the status. Now, if you use the programming device to force Y0 on, it will remain on and the CPU will not change the state of Y0. If you then force Y0 off, the CPU will maintain Y0 as off. The CPU will never update the point with the results from the application program or from the I/O update until the bit override is removed. The following diagram shows a brief overview of the bit override feature. Notice the CPU does not update the Image Register when bit override is enabled.



4

WARNING: Only authorized personnel fully familiar with all aspects of the application should make changes to the program. Make sure you thoroughly consider the impact of any changes to minimize the risk of personal injury or damage to equipment.

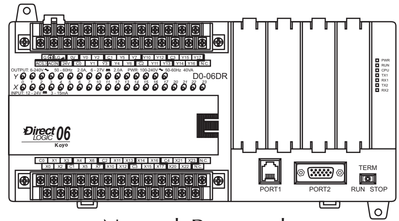
CPU Bus Communication

It is possible to transfer data to and from the CPU over the CPU bus on the backplane. This data is more than standard I/O point status. This type of communications can only occur on the CPU (local) base. There is a portion of the execution cycle used to communicate with these modules. The CPU performs both read and write requests during this segment.

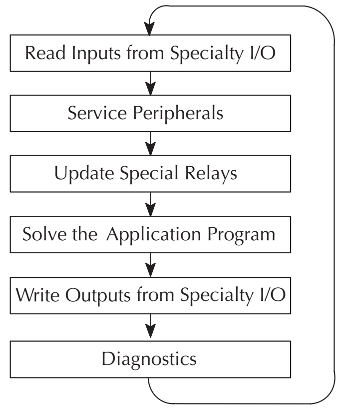
Update Clock, Special Relays and Special Registers

The DL06 CPUs have an internal real-time clock and calendar timer which is accessible to the application program. Special V-memory locations hold this information. This portion of the execution cycle makes sure these locations get updated on every scan. Also, there are several different Special Relays, such as diagnostic relays, etc., that are also updated during this segment.

Solve Application Program



Normal Run mode scan



The CPU evaluates each instruction in the application program during this segment of the scan cycle. The instructions define the relationship between the input conditions and the desired output response. The CPU uses the output image register area to store the status of the desired action for the outputs. Output image register locations are designated with a Y followed by a memory location. The actual outputs are updated during the write outputs segment of the scan cycle. There are immediate output instructions available that will update the output points immediately instead of waiting until the write output segment. A complete list of the Immediate instructions is provided in Chapter 5.

The internal control relays (C), the stages (S), and the variable memory (V) are also updated in this segment.

You may recall that you can force various types of points in the system. (This was discussed earlier in this chapter.) If any I/O points or memory data have been forced, the output image register also contains this information.

Solve PID Loop Equations

The DL06 CPU can process up to 8 PID loops. The loop calculations are run as a separate task from the ladder program execution, immediately following it. Only loops which have been configured are calculated, and then only according to a built-in loop scheduler. The sample time (calculation interval) of each loop is programmable. Please refer to Chapter 8, PID Loop Operation, for more on the effects of PID loop calculation on the overall CPU scan time.

Write Outputs

Once the application program has solved the instruction logic and constructed the output image register, the CPU writes the contents of the output image register to the corresponding output points. Remember, the CPU also made sure that any forcing operation changes were stored in the output image register, so the forced points get updated with the status specified earlier.

Write Outputs to Specialty I/O

After the CPU updates the outputs in the local and expansion bases, it sends the output point information that is required by any Specialty modules which are installed.

Diagnostics

During this part of the scan, the CPU performs all system diagnostics and other tasks such as calculating the scan time and resetting the watchdog timer. There are many different error conditions that are automatically detected and reported by the DL06 PLCs. Appendix B contains a listing of the various error codes.

Probably one of the more important things that occurs during this segment is the scan time calculation and watchdog timer control. The DL06 CPU has a "watchdog" timer that stores the maximum time allowed for the CPU to complete the solve application segment of the scan cycle. If this time is exceeded the CPU will enter the Program Mode and turn off all outputs. The default value set from the factory is 200 ms. An error is automatically reported. For example, the Handheld Programmer would display the following message "E003 S/W TIMEOUT" when the scan overrun occurs.

You can use AUX 53 to view the minimum, maximum, and current scan time. Use AUX 55 to increase or decrease the watchdog timer value.

I/O Response Time

Is Timing Important for Your Application?

I/O response time is the amount of time required for the control system to sense a change in an input point and update a corresponding output point. In the majority of applications, the CPU performs this task in such a short period of time that you may never have to concern yourself with the aspects of system timing. However, some applications do require extremely fast update times. In these cases, you may need to know how to to determine the amount of time spent during the various segments of operation.

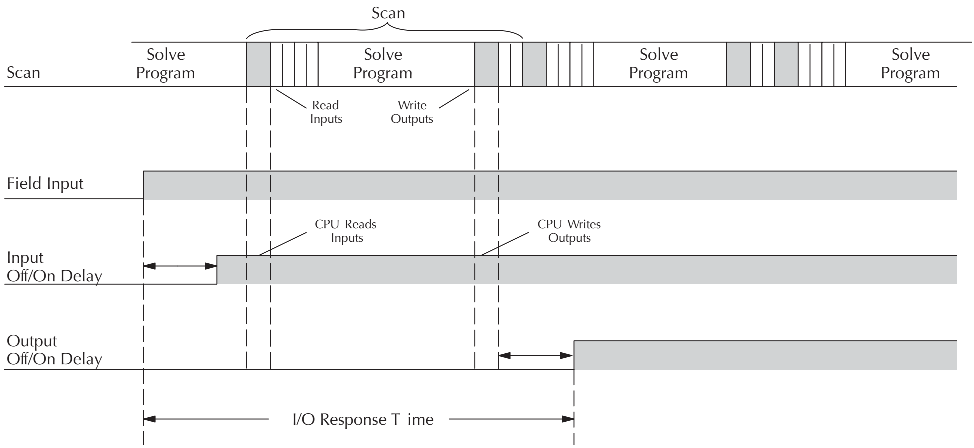
There are four things that can affect the I/O response time.

- The point in the scan cycle when the field input changes states
- Input Off to On delay time
- CPU scan time
- Output Off to On delay time

The next paragraphs show how these items interact to affect the response time.

Normal Minimum I/O Response

The I/O response time is shortest when the input changes just before the Read Inputs portion of the execution cycle. In this case the input status is read, the application program is solved, and the output point gets updated. The following diagram shows an example of the timing for this situation.

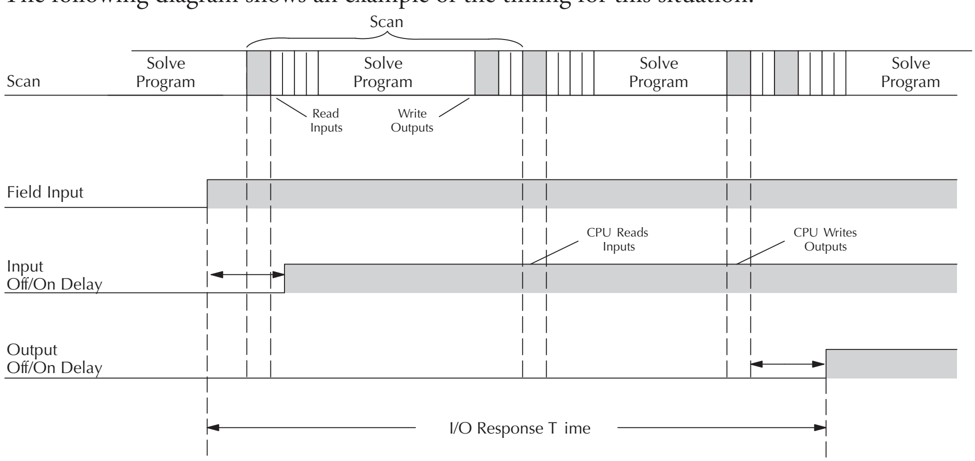


In this case, you can calculate the response time by simply adding the following items:

Input Delay + Scan Time + Output Delay = Response Time

Normal Maximum I/O Response

The I/O response time is longest when the input changes just after the Read Inputs portion of the execution cycle. In this case the new input status is not read until the following scan. The following diagram shows an example of the timing for this situation.



In this case, you can calculate the response time by simply adding the following items:

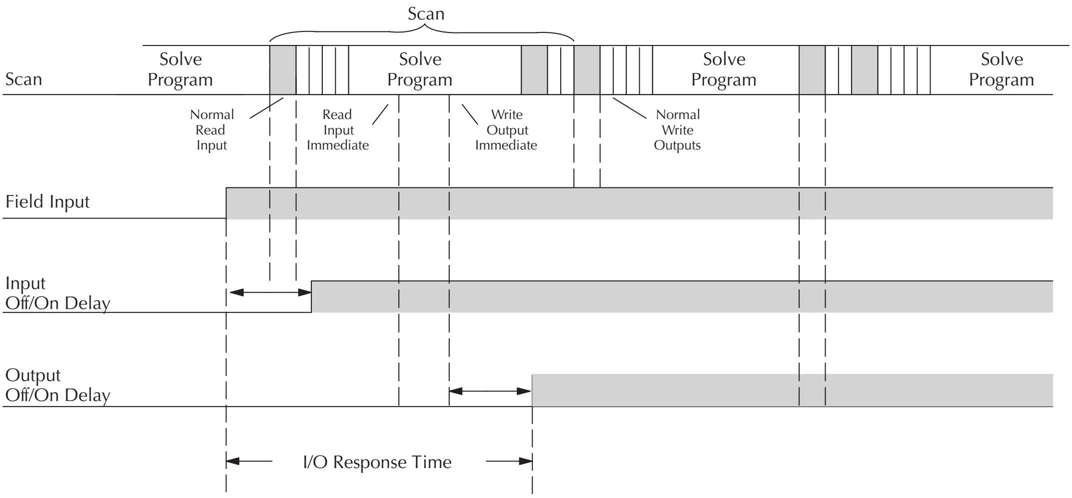
Input Delay +(2 x Scan Time) + Output Delay = Response Time

Improving Response Time

There are a few things you can do to help improve throughput.

- You can choose instructions with faster execution times
- You can use immediate I/O instructions (which update the I/O points during the program execution)
- You can use the HSIO Mode 50 Pulse Catch features designed to operate in high-speed environments. See Chapter 3 for details on using this feature.
- You can change Mode 60 filter to 0 msec for X0, X1, X2, and X3.

Of these three things the Immediate I/O instructions are probably the most important and most useful. The following example shows how an immediate input instruction



and immediate output instruction would affect the response time.

In this case, you can calculate the response time by simply adding the following items.

Input Delay** + **Instruction Execution Time** + **Output Delay** = **Response Time

The instruction execution time would be calculated by adding the time for the immediate input instruction, the immediate output instruction, and any other instructions in between the two.

***NOTE**: Even though the immediate instruction reads the most current status from I/O, it only uses the results to solve that one instruction. It does not use the new status to update the image register. Therefore, any regular instructions that follow will still use the image register values. Any immediate instructions that follow will access the I/O again to update the status.*

CPU Scan Time Considerations

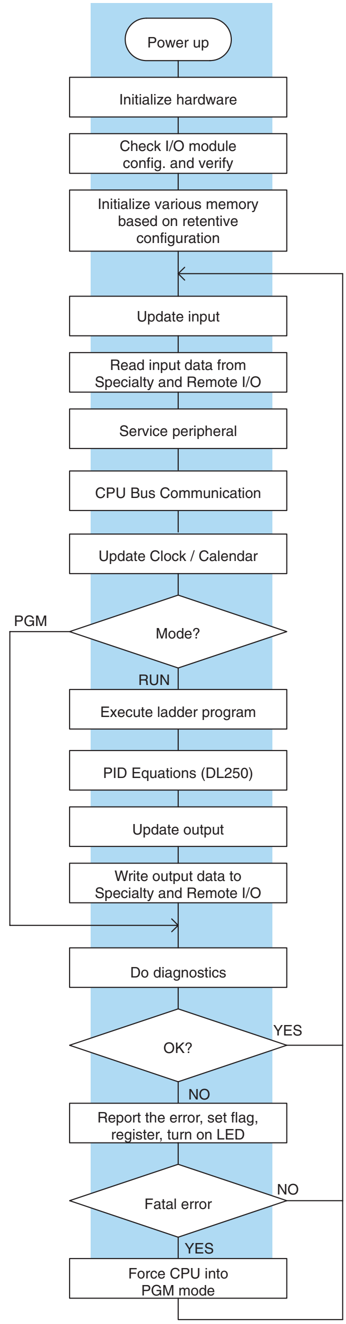
The scan time covers all the cyclical tasks that are performed by the operating system. You can use *Direct*SOFT32. or the Handheld Programmer to display the minimum, maximum, and current scan times that have occurred since the previous Program Mode to Run Mode transition. This information can be very important when evaluating the performance of a system. As we've shown previously there are several segments that make up the scan cycle. Each of these segments requires a certain amount of time to complete. Of all the segments, the following are the most important:

- Input Update
- Peripheral Service
- Program Execution
- Output Update
- Timed Interrupt Execution

The one you have the most control over is the amount of time it takes to execute the application program. This is because different instructions take different amounts of time to execute. So, if you think you need a faster scan, then you can try to choose faster instructions.

Your choice of I/O type and peripheral devices can also affect the scan time. However, these things are usually dictated by the application.

The following paragraphs provide some general information on how much time some of the segments can require.



Reading Inputs

The time required during each scan to read the input status of built-in inputs is 52.6 µs. Don't confuse this with the I/O response time that was discussed earlier.

Writing Outputs

The time required to write the output status of built-in outputs is 41.1 µS. Don't confuse this with the I/O response time that was discussed earlier.

Service Peripherals

Communication requests can occur at any time during the scan, but the CPU only "logs" the requests for service until the Service Peripherals portion of the scan. The CPU does not spend any time on this if there are no peripherals connected.

To Log Request (anytime)		DL06
Nothing Connected	Min. & Max	0µs
Port 1	Send Min. / Max.	5.8/11.8 µs
	Rec. Min. / Max.	12.5/25.2 µs
Port 2	Send Min. / Max.	6.2/14.3 µs
	Rec. Min. / Max.	14.2/31.9 µs
LCD	Min. / Max.	4.8/49.2 µs

During the Service Peripherals portion of the scan, the CPU analyzes the communications request and responds as appropriate. The amount of time required to service the peripherals depends on the content of the request.

To Service Request DL06	DL06
Minimum	9 µs
Run Mode Max.	412 µs
Program Mode Max.	2.5 second

CPU Bus Communication

Some specialty modules can also communicate directly with the CPU via the CPU bus. During this portion of the cycle the CPU completes any CPU bus communications. The actual time required depends on the type of modules installed and the type of request being processed.

Update Clock / Calendar, Special Relays, Special Registers

The clock, calendar, and special relays are updated and loaded into special V-memory locations during this time. This update is performed during both Run and Program Modes.

Modes		DL06
Program Mode	Minimum	12.0µs
	Maximum	12.0µs
Run Mode	Minimum	20.0µs
	Maximum	27.0µs

4

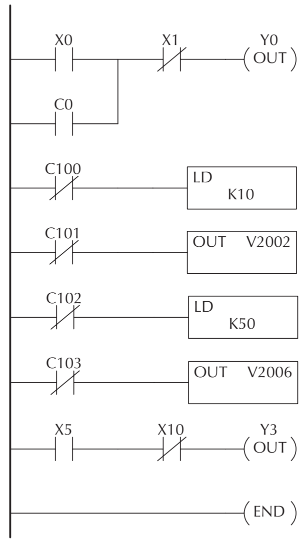
Application Program Execution

The CPU processes the program from address 0 to the END instruction. The CPU executes the program left to right and top to bottom. As each rung is evaluated the appropriate image register or memory location is updated. The time required to solve the application program depends on the type and number of instructions used, and the amount of execution overhead.

Just add the execution times for all the instructions in your program to determine to total execution time. Appendix C provides a complete list of the instruction execution times for the DL06 Micro PLC. For example, the execution time for running the program shown below is calculated as follows:

Instruction	Time
STR X0	.67 μs
OR C0	.51 μs
ANDN X1	.51 μs
OUT Y0	1.82 μs
STRN C100	.67 μs
LD K10	9.00 μs
STRN C101	.67 μs
OUT V2002	9.3 μs
STRN C102	.67 μs
LD K50	9.00 μs
STRN C103	.67 μs
OUT V2006	1.82 μs
STR X5	.67 μs
ANDN X10	.51 μs
OUT Y3	1.82 μs
END	12.80 μs
SUBT OTAL	51.11 μs

Overhead	DL06
Minimum	746.2 μs
Maximum	4352.4 μs

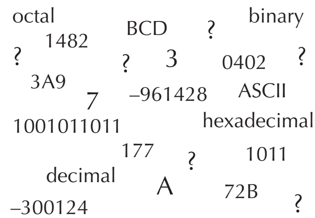


TOTAL TIME = (Program execution time + Overhead) x 1.18

The program above takes only 51.11 μs to execute during each scan. The DL06 spends 0.18 ms, on internal timed interrupt management, for every 1ms of instruction time. The total scan time is calculated by adding the program execution time to the overhead (shown above)and multiplying the result (ms) by 1.18. “Overhead” includes all other housekeeping and diagnostic tasks. The scan time will vary slightly from one scan to the next, because of fluctuation in overhead tasks.

Program Control Instructions — the DL06 CPUs offer additional instructions that can change the way the program executes. These instructions include FOR/NEXT loops, Subroutines, and Interrupt Routines. These instructions can interrupt the normal program flow and affect the program execution time. Chapter 5 provides detailed information on how these different types of instructions operate.

PLC Numbering Systems



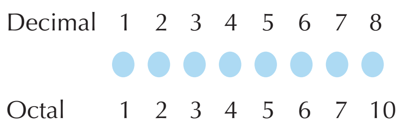
If you are a new PLC user or are using *AutomationDirect* PLCs for the first time, please take a moment to study how our PLCs use numbers. You'll find that each PLC manufacturer has their own conventions on the use of numbers in their PLCs. We want to take just a moment to familiarize you with how numbers are used in *AutomationDirect* PLCs. The information you learn here applies to all of our PLCs!

As any good computer does, PLCs store and manipulate numbers in binary form: just ones and zeros. So why do we have to deal with numbers in so many different forms? Numbers have meaning, and some representations are more convenient than others for particular purposes. Sometimes we use numbers to represent a size or amount of something. Other numbers refer to locations or addresses, or to time. In science we attach engineering units to numbers to give a particular meaning.

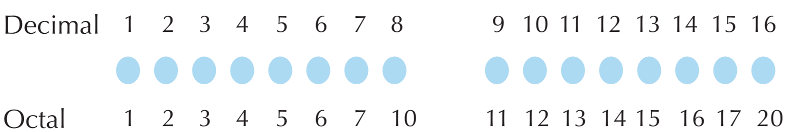
PLC Resources

PLCs offer a fixed amount of resources, depending on the model and configuration. We use the word "resources" to include variable memory (V-memory), I/O points, timers, counters, etc. Most modular PLCs allow you to add I/O points in groups of eight. In fact, all the resources of our PLCs are counted in octal. It's easier for computers to count in groups of eight than ten, because eight is an even power of 2.

Octal means simply counting in groups of eight things at a time. In the figure to the right, there are eight circles. The quantity in decimal is "8", but in octal it is "10" (8 and 9 are not valid in octal). In octal, "10" means 1 group of 8 plus 0 (no individuals).

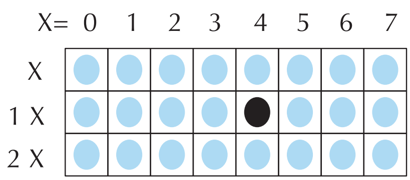


In the figure below, we have two groups of eight circles. Counting in octal we have "20" items, meaning 2 groups of eight, plus 0 individuals Don't say "twenty", say "two–zero octal". This makes a clear distinction between number systems.



After counting PLC resources, it's time to access PLC resources (there's a difference). The CPU instruction set accesses resources of the PLC using octal addresses. Octal addresses are the same as octal quantities, except they start counting at zero. The number zero is significant to a computer, so we don't skip it.

Our circles are in an array of square containers to the right. To access a resource, our PLC instruction will address its location using the octal references shown. If these were counters, "CT14" would access the black circle location.



V–Memory

V-memory address (octal)	V-memory data (binary) MSB … LSB															
V2017	0	1	0	0	1	1	1	0	0	0	1	0	1	0	0	1

Variable memory (called “V-memory”) stores data for the ladder program and for configuration settings. V-memory locations and V-memory addresses are the same thing, and are numbered in octal. For example, V2073 is a valid location, while V1983 is not valid (“9” and “8” are not valid octal digits).

Each V-memory location is one data word wide, meaning 16 bits. For configuration registers, our manuals will show each bit of a V-memory word. The least significant bit (LSB) will be on the right, and the most significant bit (MSB) on the left. We use the word “significant”, referring to the relative binary weighting of the bits.

V-memory data is 16-bit binary, but we rarely program the data registers one bit at a time. We use instructions or viewing tools that let us work with decimal, octal, and hexadecimal numbers. All these are converted and stored as binary for us.

A frequently-asked question is “How do I tell if a number is octal, BCD, or hex”? The answer is that we usually cannot tell just by looking at the data... but it does not really matter. What matters is: the source or mechanism which writes data into a V-memory location and the thing which later reads it must both use the same data type (i.e., octal, hex, binary, or whatever). The V-memory location is just a storage box... that’s all. It does not convert or move the data on its own.

Binary-Coded Decimal Numbers

BCD number	4	9	3	6
V-memory storage	0 1 0 0	1 0 0 1	0 0 1 1	0 1 1 0

Since humans naturally count in decimal (10 fingers, 10 toes), we prefer to enter and view PLC data in decimal as well. However, computers are more efficient in using pure binary numbers. A compromise solution between the two is Binary-Coded Decimal (BCD) representation. A BCD digit ranges from 0 to 9, and is stored as four binary bits (a nibble). This permits each V-memory location to store four BCD digits, with a range of decimal numbers from 0000 to 9999.

In a pure binary sense, a 16-bit word can represent numbers from 0 to 65535. In storing BCD numbers, the range is reduced to only 0 to 9999. Many math instructions use Binary-Coded Decimal (BCD) data, and *Direct*SOFT32 and the handheld programmer allow us to enter and view data in BCD.

Hexadecimal Numbers

Hexadecimal numbers are similar to BCD numbers, except they utilize all possible binary values in each 4-bit digit. They are base-16 numbers so we need 16 different digits. To extend our decimal digits 0 through 9, we use A through F as shown.

Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Hexadecimal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

A 4-digit hexadecimal number can represent all 65536 values in a V-memory word. The range is from 0000 to FFFF (hex). PLCs often need this full range for sensor data, etc. Hexadecimal is just a convenient way for humans to view full binary data.

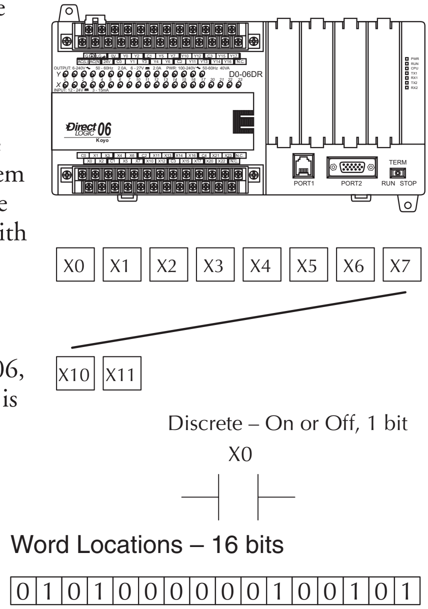
Hexadecimal number	A	7	F	4
V-memory storage	1 0 1 0	0 1 1 1	1 1 1 1	0 1 0 0

Memory Map

With any PLC system, you generally have many different types of information to process. This includes input device status, output device status, various timing elements, parts counts, etc. It is important to understand how the system represents and stores the various types of data. For example, you need to know how the system identifies input points, output points, data words, etc. The following paragraphs discuss the various memory types used in DL06 Micro PLCs. A memory map overview for the CPU follows the memory descriptions.

Octal Numbering System

All memory locations and resources are numbered in Octal (base 8). For example, the diagram shows how the octal numbering system works for the discrete input points. Notice the octal system does not contain any numbers with the digits 8 or 9.



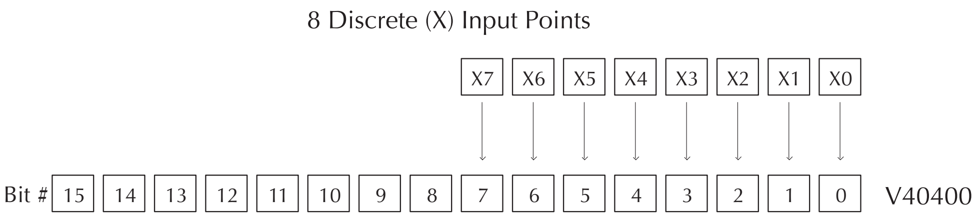
Discrete and Word Locations

As you examine the different memory types, you'll notice two types of memory in the DL06, discrete and word memory. Discrete memory is one bit that can be either a 1 or a 0. Word memory is referred to as V memory (variable) and is a 16-bit location normally used to manipulate data/numbers, store data/numbers, etc.

Some information is automatically stored in V memory. For example, the timer current values are stored in V memory.

V Memory Locations for Discrete Memory Areas

The discrete memory area is for inputs, outputs, control relays, special relays, stages, timer status bits and counter status bits. However, you can also access the bit data types as a V-memory word. Each V-memory location contains 16 consecutive discrete locations. For example, the following diagram shows how the X input points are mapped into V-memory locations.



These discrete memory areas and their corresponding V memory ranges are listed in the memory area table for DL06 Micro PLCs on the following pages.

Input Points (X Data Type)

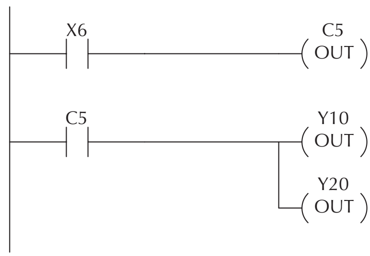
The discrete input points are noted by an X data type. There are 8 discrete input points and 256 discrete input addresses available with DL06 CPUs. In this example, the output point Y0 will be turned on when input X0 energizes.

Output Points (Y Data Type)

The discrete output points are noted by a Y data type. There are 6 discrete outputs and 256 discrete output addresses available with DL06 CPUs. In this example, output point Y1 will be turned on when input X1 energizes.

Control Relays (C Data Type)

Control relays are discrete bits normally used to control the user program. The control relays do not represent a real world device, that is, they cannot be physically tied to switches, output coils, etc. They are internal to the CPU. Because of this, control relays can be programmed as discrete inputs or discrete outputs. These locations are used in programming the discrete memory locations (C) or the corresponding word location which contains 16 consecutive discrete locations.

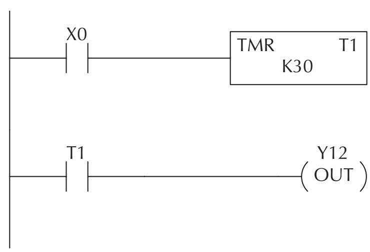


In this example, memory location C5 will energize when input X6 turns on. The second rung shows a simple example of how to use a control relay as an input.

Timers and Timer Status Bits (T Data Type)

Timer status bits reflect the relationship between the current value and the preset value of a specified timer. The timer status bit will be on when the current value is equal or greater than the preset value of a corresponding timer.

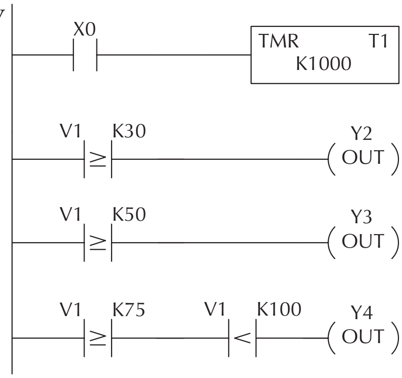
When input X0 turns on, timer T1 will start. When the timer reaches the preset of 3 seconds (K of 30) timer status contact T1 turns on. When T1 turns on, output Y12 turns on. Turning off X0 resets the timer.



Timer Current Values (V Data Type)

As mentioned earlier, some information is automatically stored in V memory. This is true for the current values associated with timers. For example, V0 holds the current value for Timer 0, V1 holds the current value for Timer 1, etc. These can also be designated as TA0 (Timer Accumulated) for Timer 0, and TA1 for Timer 1.

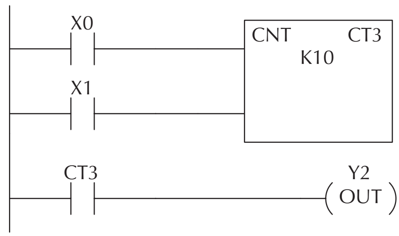
The primary reason for this is programming flexibility. The example shows how you can use relational contacts to monitor several time intervals from a single timer.



Counters and Counter Status Bits (CT Data type)

Counter status bits that reflect the relationship between the current value and the preset value of a specified counter. The counter status bit will be on when the current value is equal to or greater than the preset value of a corresponding counter.

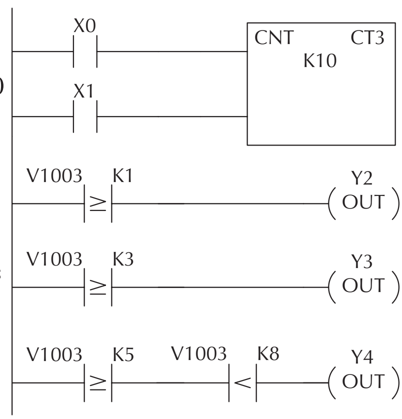
Each time contact X0 transitions from off to on, the counter increments by one. (If X1 comes on, the counter is reset to zero.) When the counter reaches the preset of 10 counts (K of 10) counter status contact CT3 turns on. When CT3 turns on, output Y2 turns on.



Counter Current Values (V Data Type)

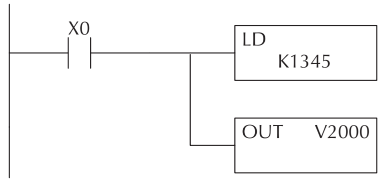
Just like the timers, the counter current values are also automatically stored in V memory. For example, V1000 holds the current value for Counter CT0, V1001 holds the current value for Counter CT1, etc. These can also be designated as CTA0 (Counter Accumulated) for Counter 0 and CTA01 for Counter 1.

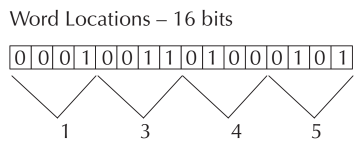
The primary reason for this is programming flexibility. The example shows how you can use relational contacts to monitor the counter values.



Word Memory (V Data Type)

Word memory is referred to as V memory (variable) and is a 16-bit location normally used to manipulate data/numbers, store data/numbers, etc. Some information is automatically stored in V memory. For example, the timer current values are stored in V memory. The example shows how a four-digit BCD constant is loaded into the accumulator and then stored in a V-memory location.

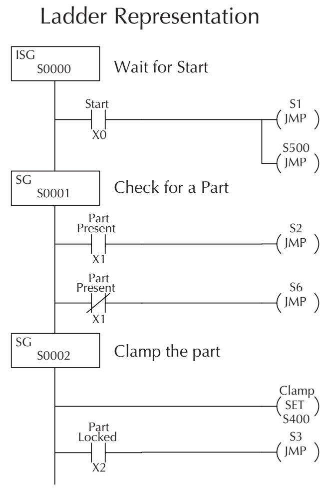




Stages (S Data type)

Stages are used in RLL*PLUS* programs to create a structured program, similar to a flowchart. Each program Stage denotes a program segment. When the program segment, or Stage, is active, the logic within that segment is executed. If the Stage is off, or inactive, the logic is not executed and the CPU skips to the next active Stage. (See Chapter 7 for a more detailed description of RLL*PLUS* programming.)

Each Stage also has a discrete status bit that can be used as an input to indicate whether the Stage is active or inactive. If the Stage is active, then the status bit is on. If the Stage is inactive, then the status bit is off. This status bit can also be turned on or off by other instructions, such as the SET or RESET instructions. This allows you to easily control stages throughout the program.



Special Relays (SP Data Type)

Special relays are discrete memory locations with pre-defined functionality. There are many different types of special relays. For example, some aid in program development, others provide system operating status information, etc. Appendix D provides a complete listing of the special relays.

In this example, control relay C10 will energize for 50 ms and de-energize for 50 ms because SP5 is a pre–defined relay that will be on for 50 ms and off for 50 ms.

SP4: 1 second clock
SP5: 100 ms clock
SP6: 50 ms clock

DL06 System V-memory

System Parameters and Default Data Locations (V Data Type)

The DL06 PLCs reserve several V-memory locations for storing system parameters or certain types of system data. These memory locations store things like the error codes, High-Speed I/O data, and other types of system setup information.

System V-memory	Description of Contents	Default Values / Ranges
V700-V707	Sets the V-memory location for option card in slot 1	N/A
V710-V717	Sets the V-memory location for option card in slot 2	N/A
V720-V727	Sets the V-memory location for option card in slot 3	N/A
V730-V737	Sets the V-memory location for option card in slot 4	N/A
V3630–V3707	The default location for multiple preset values for UP/DWN and UP Counter 1 or pulse catch function	N/A
V3710-V3767	The default location for multiple preset values for UP/DWN and UP Counter 2	N/A
V7620–V7627 V7620 V7621 V7622 V7623 V7624 V7625 V7626 V7627	"Locations for DV–1000 operator interface parameters.Sets the V-memory location that contains the value.Sets the V-memory location that contains the message.Sets the total number (1 – 32) of V-memory locations to be displayed.Sets the V-memory location that contains the numbers to be displayed.Sets the V-memory location that contains the character code to be displayed.Contains the function number that can be assigned to each key.Powerup operational mode.Change preset value."	V0 – V3760 V0 – V37601 - 32 V0 – V3760 V0 – V3760 V-memory for X, Y, or C0, 1, 2, 3, 12Default = 0000
V7630	Starting location for the multi–step presets for channel 1. The default value is 3630, which indicates the first value should be obtained from V3630. Since there are 24 presets available, the default range is V3630 – V3707. You can change the starting point if necessary.	Default: V3630 Range: V0- V3710
V7631	Starting location for the multi–step presets for channel 2. The default value is 3710, which indicates the first value should be obtained from V3710. Since there are 24 presets available, the default range is V3710 – V3767. You can change the starting point if necessary.	Default: V3710 Range: V0- V3710
V7632	Setup Register for Pulse Output	N/A
V7633	Sets the desired function code for the high speed counter, interrupt, pulse catch, pulse train, and input filter. Location can also be used to set the power-up in Run Mode option.	Default: 0060 Lower Byte Range:Range: 10 – Counter 20 – Quadrature 30 – Pulse Out 40 – Interrupt 50 – Pulse Catch 60 – Filtered discrete In. Upper Byte Range:Bits 8–11, 14, 15: Unused, Bit 13: Power–up in RUN,only if Mode Switch is inTERM position. Bit 12 is used to enable the battery.
V7634	X0 Setup Register for High-Speed I/O functions	Default: 1006
V7635	X1 Setup Register for High-Speed I/O functions	Default: 1006
V7636	X2 Setup Register for High-Speed I/O functions	Default: 1006
V7637	X3 Setup Register for High-Speed I/O functions	Default: 1006
V7640	PID Loop table beginning address	"V1200 – V7377 V10000 - V17777"

System V-memory	Description of Contents	Default Values / Ranges
V7641	Number of Loops	1-8
V7642	Error Code - V-memory Error location for Loop Table	
V7643-V7647	Reserved	
V7650	Port 2: Setup for V-memory address for Non-procedure protocol	V1200 – V7377 V10000 - V17777
V7653	Port 2: Setup for terminate code for Non-procedure protocol	
V7655	Port 2: Setup for the protocol, time-out, and the response delay time.	
V7656	Port 2: Setup for the station number, baud rate, STOP bit, and parity.	
V7657	Port 2: Setup completion code used to notify the completion of the parameter setup	
V7660	Scan control setup: Keeps the scan control mode.	
V7661	Setup timer over counter: Counts the times the actual scan time exceeds the user setup time.	
V7662–V7717	Reserved	
V7720–V7722	Locations for DV–1000 operator interface parameters.	
V7720	Titled Timer preset value pointer	
V7721	Title Counter preset value pointer	
V7722	HiByte-Titled Timer preset block size, LoByte-Titled Counter preset block size	
V7723–V7737	Reserved	
V7740	Port 1 and Port 2: Communication Auto Reset Timer Setup	Default: 3030
V7741–V7746	Reserved	
V7747	Location contains a 10mS counter (0-99). This location increments once every 10 mS	
V7750	Reserved	
V7751	Fault Message Error Code — stores the 4-digit code used with the FAULT instruction when the instruction is executed.	
V7752	I/O Configuration Error: Current ID code of error slot	
V7753	I/O Configuration Error: Old ID code of error slot	
V7754	I/O Configuration Error: error slot number	
V7755	Error code — stores the fatal error code.	
V7756	Error code — stores the major error code.	
V7757	Error code — stores the minor error code.	
V7760–V7762	Reserved	
V7763	Program address where syntax error exists	
V7764	Syntax error code	
V7765	Scan counter — stores the total number of scan cycles that have occurred since the last Program Mode to Run Mode transition.	
V7766	Contains the number of seconds on the clock (00-59)	
V7767	Contains the number of minutes on the clock (00-59)	
V7770	Contains the number of hours on the clock (00-23)	
V7771	Contains the day of the week (Mon., Tues., Wed., etc.)	
V7772	Contains the day of the month (01, 02, etc.)	
V7773	Contains the month (01 to 12)	
V7774	Contains the year (00 to 99)	
V7775	Scan — stores the current scan time (milliseconds).	
V7776	Scan — stores the minimum scan time that has occurred since the last	
V7777	Program Mode to Run Mode transition (milliseconds).	
V37700-V37737	For remote I/O	

DL06 Memory Map

Memory Type	Discrete Memory Reference (octal)	Word Memory Reference (octal)	Decimal	Symbol
Input Points	X0 – X777	V40400 - V40437	512	X0 ┤ ├
Output Points	Y0 – Y777	V40500 – V40537	512	Y0 —( )—
Control Relays	C0 – C1777	V40600 - V40677	1024	C0 ┤ ├ C0 —( )—
Special Relays	SP0 – SP777	V41200 – V41237	512	SP0 ┤ ├
Timers	T0 – T377	V41100 – V41117	256	TMR T0 K100
Timer Current Values	None	V0 – V377	256	V0 K100 ┤≥├
Timer Status Bits	T0 – T377	V41100 – V41117	256	T0 ┤ ├
Counters	CT0 – CT177	V41140 – V41147	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	V400-V677 V1200 – V7377 V10000 - V17777	192 3200 4096	None specific, used with many instructions.
Data Words EEPROM	None	V7400 – V7577	128	None specific, used with many instructions. Data can be rewritten to EEPROM at least 100,000 times before it fails.
Stages	S0 – S1777	V41000 – V41017	1024	SG S001 SP0 ┤ ├
Remote I/O	GX0-GX3777 GY0-GY3777	V40000-V40177 V40200-V40377	2048 2048	GX0 ┤ ├ GY0 —( )—
System parameters	None	V700-V777 V7600 – V7777 V36000-V37777	64 128 1024	None specific, used for various purposes

1-The DL06 systems are limited to 20 discrete inputs and 16 discrete outputs with the present available hardware, but 512 point addresses exist.

X Input / Y Output Bit Map

This table provides a listing of individual input and output points associated with each V-memory address bit for the DL06's twenty integrated physical inputs and 16 integrated physical outputs in addition to up to 64 inputs and 64 outputs for option cards. Actual available references are X0 to X777 (V40400 – V40437) and Y0 to Y777 (V40500 - V40537).

MSB	DL06 Input (X) and Output (Y) Points														LSB	X Input Address	Y Output 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	V40500
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40401	V40501
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40402	V40502
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40403	V40503
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40404	V40504
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40405	V40505
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40406	V40506
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40407	V40507
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40410	V40510
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	V40411	V40511
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40412	V40512
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40413	V40513
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40414	V40514
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40415	V40515
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40416	V40516
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40417	V40517
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V40420	V40520
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V40421	V40521
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V40422	V40522
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V40423	V40523
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V40424	V40524
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V40425	V40525
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V40426	V40526
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V40427	V40527
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V40430	V40530
637	636	635	634	633	632	631	630	627	626	625	624	623	622	621	620	V40431	V40531
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V40432	V40532
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V40433	V40533
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V40434	V40534
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V40435	V40535
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V40436	V40536
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V40437	V40537

Stage Control / Status Bit Map

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

MSB	DL06 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
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

MSB	DL06 Stage (S) Control Bits														LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
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
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	1232	1231	1230	1227	1226	1225	1224	1223	1222	1221	1220	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	1632	1631	1630	1627	1626	1625	1624	1623	1622	1621	1620	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

Control Relay Bit Map

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

MSB	DL06 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
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	632	631	630	627	626	625	624	623	622	621	620	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
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

MSB							DL06 Control Relays (C)								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	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	1632	1631	1630	1627	1626	1625	1624	1623	1622	1621	1620	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

4

Timer Status Bit Map

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

MSB	DL06 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
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V41110
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	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 individual counter contacts associated with each V-memory address bit.

MSB	DL06 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

4

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	Remote I/O (GX) and (GY) Points														LSB	GX Address	GY 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	V40200
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40001	V40201
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40002	V40202
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40003	V40203
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40004	V40204
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40005	V40205
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40006	V40206
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40007	V40207
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40010	V40210
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	V40011	V40211
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40012	V40212
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40013	V40213
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40004	V40214
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40015	V40215
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40016	V40216
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40007	V40217
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V40020	V40220
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V40021	V40221
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V40022	V40222
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V40023	V40223
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V40024	V40224
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V40025	V40225
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V40026	V40226
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V40027	V40227
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V40030	V40230
637	636	635	634	633	632	631	630	627	626	625	624	623	622	621	620	V40031	V40231
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V40032	V40232
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V40033	V40233
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V40034	V40234
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V40035	V40235
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V40036	V40236
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V40037	V40237

MSB	DL06 Remote I/O (GX) and (GY) Points														LSB	GX Address	GY 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	V40240
1037	1036	1035	1034	1033	1032	1031	1030	1027	1026	1025	1024	1023	1022	1021	1020	V40041	V40241
1057	1056	1055	1054	1053	1052	1051	1050	1047	1046	1045	1044	1043	1042	1041	1040	V40042	V40242
1077	1076	1075	1074	1073	1072	1071	1070	1067	1066	1065	1064	1063	1062	1061	1060	V40043	V40243
1117	1116	1115	1114	1113	1112	1111	1110	1107	1106	1105	1104	1103	1102	1101	1100	V40044	V40244
1137	1136	1135	1134	1133	1132	1131	1130	1127	1126	1125	1124	1123	1122	1121	1120	V40045	V40245
1157	1156	1155	1154	1153	1152	1151	1150	1147	1146	1145	1144	1143	1142	1141	1140	V40046	V40246
1177	1176	1175	1174	1173	1172	1171	1170	1167	1166	1165	1164	1163	1162	1161	1160	V40047	V40247
1217	1216	1215	1214	1213	1212	1211	1210	1207	1206	1205	1204	1203	1202	1201	1200	V40050	V40250
1237	1236	1235	1234	1233	1232	1231	1230	1227	1226	1225	1224	1223	1222	1221	1220	V40051	V40251
1257	1256	1255	1254	1253	1252	1251	1250	1247	1246	1245	1244	1243	1242	1241	1240	V40052	V40252
1277	1276	1275	1274	1273	1272	1271	1270	1267	1266	1265	1264	1263	1262	1261	1260	V40053	V40253
1317	1316	1315	1314	1313	1312	1311	1310	1307	1306	1305	1304	1303	1302	1301	1300	V40054	V40254
1337	1336	1335	1334	1333	1332	1331	1330	1327	1326	1325	1324	1323	1322	1321	1320	V40055	V40255
1357	1356	1355	1354	1353	1352	1351	1350	1347	1346	1345	1344	1343	1342	1341	1340	V40056	V40256
1377	1376	1375	1374	1373	1372	1371	1370	1367	1366	1365	1364	1363	1362	1361	1360	V40057	V40257
1417	1416	1415	1414	1413	1412	1411	1410	1407	1406	1405	1404	1403	1402	1401	1400	V40060	V40260
1437	1436	1435	1434	1433	1432	1431	1430	1427	1426	1425	1424	1423	1422	1421	1420	V40061	V40261
1457	1456	1455	1454	1453	1452	1451	1450	1447	1446	1445	1444	1443	1442	1441	1440	V40062	V40262
1477	1476	1475	1474	1473	1472	1471	1470	1467	1466	1465	1464	1463	1462	1461	1460	V40063	V40263
1517	1516	1515	1514	1513	1512	1511	1510	1507	1506	1505	1504	1503	1502	1501	1500	V40064	V40264
1537	1536	1535	1534	1533	1532	1531	1530	1527	1526	1525	1524	1523	1522	1521	1520	V40065	V40265
1557	1556	1555	1554	1553	1552	1551	1550	1547	1546	1545	1544	1543	1542	1541	1540	V40066	V40266
1577	1576	1575	1574	1573	1572	1571	1570	1567	1566	1565	1564	1563	1562	1561	1560	V40067	V40267
1617	1616	1615	1614	1613	1612	1611	1610	1607	1606	1605	1604	1603	1602	1601	1600	V40070	V40270
1637	1636	1635	1634	1633	1632	1631	1630	1627	1626	1625	1624	1623	1622	1621	1620	V40071	V40271
1657	1656	1655	1654	1653	1652	1651	1650	1647	1646	1645	1644	1643	1642	1641	1640	V40072	V40272
1677	1676	1675	1674	1673	1672	1671	1670	1667	1666	1665	1664	1663	1662	1661	1660	V40073	V40273
1717	1716	1715	1714	1713	1712	1711	1710	1707	1706	1705	1704	1703	1702	1701	1700	V40074	V40274
1737	1736	1735	1734	1733	1732	1731	1730	1727	1726	1725	1724	1723	1722	1721	1720	V40075	V40275
1757	1756	1755	1754	1753	1752	1751	1750	1747	1746	1745	1744	1743	1742	1741	1740	V40076	V40276
1777	1776	1775	1774	1773	1772	1771	1770	1767	1766	1765	1764	1763	1762	1761	1760	V40077	V40277

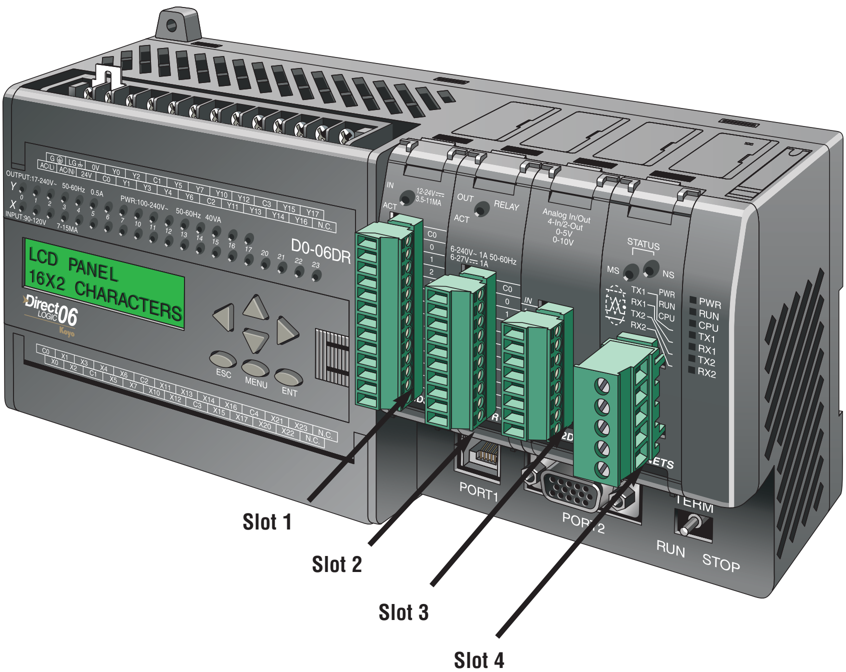
MSB	DL06 Remote I/O (GX) and (GY) Points														LSB	GX Address	GY Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0		
2017	2016	2015	2014	2013	2012	2011	2010	2007	2006	2005	2004	2003	2002	2001	2000	V40100	V40300
2037	2036	2035	2034	2033	2032	2031	2030	2027	2026	2025	2024	2023	2022	2021	2020	V40101	V40301
2057	2056	2055	2054	2053	2052	2051	2050	2047	2046	2045	2044	2043	2042	2041	2040	V40102	V40302
2077	2076	2075	2074	2073	2072	2071	2070	2067	2066	2065	2064	2063	2062	2061	2060	V40103	V40303
2117	2116	2115	2114	2113	2112	2111	2110	2107	2106	2105	2104	2103	2102	2101	2100	V40104	V40304
2137	2136	2135	2134	2133	2132	2131	2130	2127	2126	2125	2124	2123	2122	2121	2120	V40105	V40305
2157	2156	2155	2154	2153	2152	2151	2150	2147	2146	2145	2144	2143	2142	2141	2140	V40106	V40306
2177	2176	2175	2174	2173	2172	2171	2170	2167	2166	2165	2164	2163	2162	2161	2160	V40107	V40307
2217	2216	2215	2214	2213	2212	2211	2210	2207	2206	2205	2204	2203	2202	2201	2200	V40110	V40310
2237	2236	2235	2234	2233	2232	2231	2230	2227	2226	2225	2224	2223	2222	2221	2220	V40111	V40311
2257	2256	2255	2254	2253	2252	2251	2250	2247	2246	2245	2244	2243	2242	2241	2240	V40112	V40312
2277	2276	2275	2274	2273	2272	2271	2270	2267	2266	2265	2264	2263	2262	2261	2260	V40113	V40313
2317	2316	2315	2314	2313	2312	2311	2310	2307	2306	2305	2304	2303	2302	2301	2300	V40114	V40314
2337	2336	2335	2334	2333	2332	2331	2330	2327	2326	2325	2324	2323	2322	2321	2320	V40115	V40315
2357	2356	2355	2354	2353	2352	2351	2350	2347	2346	2345	2344	2343	2342	2341	2340	V40116	V40316
2377	2376	2375	2374	2373	2372	2371	2370	2367	2366	2365	2364	2363	2362	2361	2360	V40117	V40317
2417	2416	2415	2414	2413	2412	2411	2410	2407	2406	2405	2404	2403	2402	2401	2400	V40120	V40320
2437	2436	2435	2434	2433	2432	2431	2430	2427	2426	2425	2424	2423	2422	2421	2420	V40121	V40321
2457	2456	2455	2454	2453	2452	2451	2450	2447	2446	2445	2444	2443	2442	2441	2440	V40122	V40322
2477	2476	2475	2474	2473	2472	2471	2470	2467	2466	2465	2464	2463	2462	2461	2460	V40123	V40323
2517	2516	2515	2514	2513	2512	2511	2510	2507	2506	2505	2504	2503	2502	2501	2500	V40124	V40324
2537	2536	2535	2534	2533	2532	2531	2530	2527	2526	2525	2524	2523	2522	2521	2520	V40125	V40325
2557	2556	2555	2554	2553	2552	2551	2550	2547	2546	2545	2544	2543	2542	2541	2540	V40126	V40326
2577	2576	2575	2574	2573	2572	2571	2570	2567	2566	2565	2564	2563	2562	2561	2560	V40127	V40327
2617	2616	2615	2614	2613	2612	2611	2610	2607	2606	2605	2604	2603	2602	2601	2600	V40130	V40330
2637	2636	2635	2634	2633	2632	2631	2630	2627	2626	2625	2624	2623	2622	2621	2620	V40131	V40331
2657	2656	2655	2654	2653	2652	2651	2650	2647	2646	2645	2644	2643	2642	2641	2640	V40132	V40332
2677	2676	2675	2674	2673	2672	2671	2670	2667	2666	2665	2664	2663	2662	2661	2660	V40133	V40333
2717	2716	2715	2714	2713	2712	2711	2710	2707	2706	2705	2704	2703	2702	2701	2700	V40134	V40334
2737	2736	2735	2734	2733	2732	2731	2730	2727	2726	2725	2724	2723	2722	2721	2720	V40135	V40335
2757	2756	2755	2754	2753	2752	2751	2750	2747	2736	2735	2734	2733	2732	2731	2730	V40136	V40336
2777	2776	2775	2774	2773	2772	2771	2770	2767	2766	2765	2764	2763	2762	2761	2760	V40137	V40337

MSB	DL06 Remote I/O (GX) and (GY) Points														LSB	GX Address	GY Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0		
3017	3016	3015	3014	3013	3012	3011	3010	3007	3006	3005	3004	3003	3002	3001	3000	V40140	V40340
3037	3036	3035	3034	3033	3032	3031	3030	3027	3026	3025	3024	3023	3022	3021	3020	V40141	V40341
3057	3056	3055	3054	3053	3052	3051	3050	3047	3046	3045	3044	3043	3042	3041	3040	V40142	V40342
3077	3076	3075	3074	3073	3072	3071	3070	3067	3066	3065	3064	3063	3062	3061	3060	V40143	V40343
3117	3116	3115	3114	3113	3112	3111	3110	3107	3106	3105	3104	3103	3102	3101	3100	V40144	V40344
3137	3136	3135	3134	3133	3132	3131	3130	3127	3126	3125	3124	3123	3122	3121	3120	V40145	V40345
3157	3156	3155	3154	3153	3152	3151	3150	3147	3146	3145	3144	3143	3142	3141	3140	V40146	V40346
3177	3176	3175	3174	3173	3172	3171	3170	3167	3166	3165	3164	3163	3162	3161	3160	V40147	V40347
3217	3216	3215	3214	3213	3212	3211	3210	3207	3206	3205	3204	3203	3202	3201	3200	V40150	V40350
3237	3236	3235	3234	3233	3232	3231	3230	3227	3226	3225	3224	3223	3222	3221	3220	V40151	V40351
3257	3256	3255	3254	3253	3252	3251	3250	3247	3246	3245	3244	3243	3242	3241	3240	V40152	V40352
3277	3276	3275	3274	3273	3272	3271	3270	3267	3266	3265	3264	3263	3262	3261	3260	V40153	V40353
3317	3316	3315	3314	3313	3312	3311	3310	3307	3306	3305	3304	3303	3302	3301	3300	V40154	V40354
3337	3336	3335	3334	3333	3332	3331	3330	3327	3326	3325	3324	3323	3322	3321	3320	V40155	V40355
3357	3356	3355	3354	3353	3352	3351	3350	3347	3346	3345	3344	3343	3342	3341	3340	V40156	V40356
3377	3376	3375	3374	3373	3372	3371	3370	3367	3366	3365	3364	3363	3362	3361	3360	V40157	V40357
3417	3416	3415	3414	3413	3412	3411	3410	3407	3406	3405	3404	3403	3402	3401	3400	V40160	V40360
3437	3436	3435	3434	3433	3432	3431	3430	3427	3426	3425	3424	3423	3422	3421	3420	V40161	V40361
3457	3456	3455	3454	3453	3452	3451	3450	3447	3446	3445	3444	3443	3442	3441	3440	V40162	V40362
3477	3476	3475	3474	3473	3472	3471	3470	3467	3466	3465	3464	3463	3462	3461	3460	V40163	V40363
3517	3516	3515	3514	3513	3512	3511	3510	3507	3506	3505	3504	3503	3502	3501	3500	V40164	V40364
3537	3536	3535	3534	3533	3532	3531	3530	3527	3526	3525	3524	3523	3522	3521	3520	V40165	V40365
3557	3556	3555	3554	3553	3552	3551	3550	3547	3546	3545	3544	3543	3542	3541	3540	V40166	V40366
3577	3576	3575	3574	3573	3572	3571	3570	3567	3566	3565	3564	3563	3562	3561	3560	V40167	V40367
3617	3616	3615	3614	3613	3612	3611	3610	3607	3606	3605	3604	3603	3602	3601	3600	V40170	V40370
3637	3636	3635	3634	3633	3632	3631	3630	3627	3626	3625	3624	3623	3622	3621	3620	V40171	V40371
3657	3656	3655	3654	3653	3652	3651	3650	3647	3646	3645	3644	3643	3642	3641	3640	V40172	V40372
3677	3676	3675	3674	3673	3672	3671	3670	3667	3666	3665	3664	3663	3662	3661	3660	V40173	V40373
3717	3716	3715	3714	3713	3712	3711	3710	3707	3706	3705	3704	3703	3702	3701	3700	V40174	V40374
3737	3736	3735	3734	3733	3732	3731	3730	3727	3726	3725	3724	3723	3722	3721	3720	V40175	V40375
3757	3756	3755	3754	3753	3752	3751	3750	3747	3746	3745	3744	3743	3742	3741	3740	V40176	V40376
3777	3776	3775	3774	3773	3772	3771	3770	3767	3766	3765	3764	3763	3762	3761	3760	V40177	V40377

Module Placement

Slot Numbering

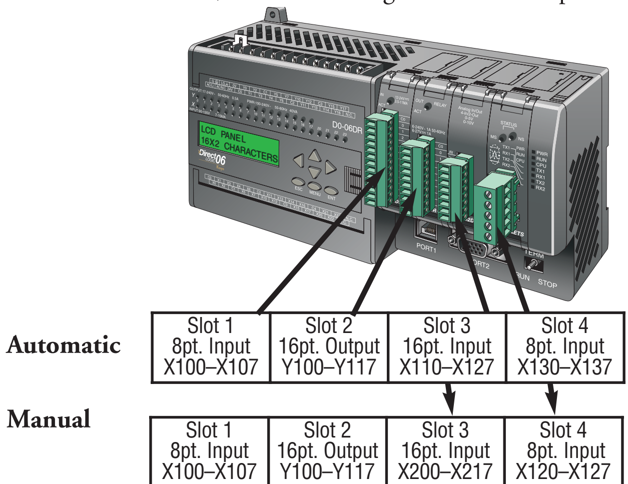
The DL06 has four slots, which are numbered as follows:



Automatic I/O Configuration

The DL06 CPUs automatically detect any installed I/O modules (including specialty modules)at powerup, and establish the correct I/O configuration and addresses. This applies to modules located in local and expansion I/O bases. For most applications, you will never have to change the configuration.

I/O addresses use octal numbering, starting at X100 and Y100 in the slot next to the CPU. The addresses are assigned in groups of 8, or 16 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 system. Both the Handheld Programmer and DirectSOFT32 provide AUX functions that allow you to automatically configure the I/O. For example, with the Handheld Programmer AUX 46 executes an automatic configuration, which allows the CPU to examine the installed modules and determine the I/O configuration and addressing.With *Direct*SOFT32, the PLC Configure I/O menu option would be used.



Manual I/O Configuration

It may never become necessary, but DL06 CPUs allow manual I/O address assignments for any I/O slot(s) . You can manually modify an auto configuration to match arbitrary I/O numbering. For example, two adjacent input modules can have starting addresses at X100 and X200.Use DirectSOFT32 PLC Configure I/O menu option to assign manual I/O address. 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). You can still use 8 point modules, but 16 addresses will be assigned and the upper eight 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 DL06 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.

Power Budgeting

The DL06 has four option card slots. To determine whether the combination of cards you select will have sufficient power, you will need to perform a power budget calculation.

Power supplied

Power is supplied from two sources, the internal base unit power supply and, if required, an external supply (customer furnished). The D0-06xx (AC powered) PLCs supply a limited amount of 24VDC power. The 24VDC output can be used to power external devices. For power budgeting, start by considering the power supplied by the base unit. All DL06 PLCs supply the same amount of 5VDC power. Only the AC units offer 24VDC auxiliary power. Be aware of the trade-off between 5VDC power and 24VDC power. The amount of 5VDC power available depends on the amount of 24VDC power being used, and the amount of 24VDC power available depends on the amount of 5VDC power consumed. Determine the amount of internally supplied power from the table on the following page.

Power required by base unit

Because of the different I/O configurations available in the DL06 family, the power consumed by the base unit itself varies from model to model. Subtract the amount of power required by the base unit from the amount of power supplied by the base unit. Be sure to subtract 5VDC and 24VDC amounts.

Power required by option cards

Next, subtract the amount of power required by the option cards you are planning to use. Again, remember to subtract both 5VDC and 24VDC. If your power budget analysis shows surplus power available, you should have a workable configuration.

DL06 Power Supplied by Base Units		
Part Number	**5 VDC (mA)**	**24 VDC (mA)**
D0-06xx	<1500mA	300mA
	<2000mA	200mA
D0-06xx-D	1500mA	none

If the 5VDC loading is less than 2000mA but more than 1500mA, than available 24VDC supply current is 200mA. If the 5VDC loading is less than 1500mA, then the available 24VDC current is 300mA.

DL06 Base Unit Power Required		
Part Number	**5 VDC (mA)**	**24 VDC (mA)**
D0-06AA	800mA	none
D0-06AR	900mA	none
D0-06DA	800mA	none
D0-06DD1	600mA	280mA*
D0-06DD2	600mA	none
D0-06DR	950mA	none
D0-06DD1-D	600mA	280mA*
D0-06DD2-D	600mA	none
D0-06DR-D	950mA	none

DL06 Power Consumed by Other Devices		
Part Number	**5 VDC (mA)**	**24 VDC (mA)**
D0-06LCD	50mA	none
D2-HPP	200mA	none
DV1000	150mA	none

DL06 Power Consumed by Option Cards		
Part number	**5 VDC**	**24 VDC**
D0-07CDR	130mA	none
D0-08CDD1	100mA	none
D0-08TR	280mA	none
D0-10ND3	35mA	none
D0-10TD1	150mA	none
D0-10TD2	150mA	none
D0-16ND3	35mA	none
D0-16TD1	200mA	none
D0-16TD2	200mA	none
F0-04AD-1	50mA	none
F0-2AD2DA-2	50mA	30mA
F0-4AD2DA-1	100mA	40mA
F0-4AD2DA-2	100mA	none
D0-DEVNETS	45mA	none
H0-ECOM	25mA	none

Power Budgeting Example			
Power Source		**5VDC power (mA)**	**24VDC power (mA)**
D0-06DD1 (select row A or row B)	A	1500mA	300mA
	B	2000mA	200mA
Current Required		**5VDC power (mA)**	**24VDC power (mA)**
D0-06DD1		600mA	280mA*
D0-16ND3		35mA	0
D0-10TD1		150mA	0
D0-08TR		280mA	0
F0-4AD2DA-2		100mA	0
D0-06LCD		50mA	0
Total Used		1215mA	280mA
Remaining	A	285mA	20mA
	B	785mA	note 1

*** Auxiliary 24VDC used to power V+ terminal of D0-06DD1/-D sinking outputs.**

Note 1: If the PLC's auxiliary 24VDC power source is used to power the sinking outputs, use power choice A, above.

Configuring the DL06's Comm Ports

This section describes how to configure the CPU's built-in networking ports. for either MODBUS or DirectNET. This will allow you to connect the DL06 PLC system directly to MODBUS networks using the RTU protocol, or to other devices on a DirectNET network. MODBUS hosts system 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 the Gould MODBUS Protocol reference Guide (P1–MBUS–300 Rev. B). 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.

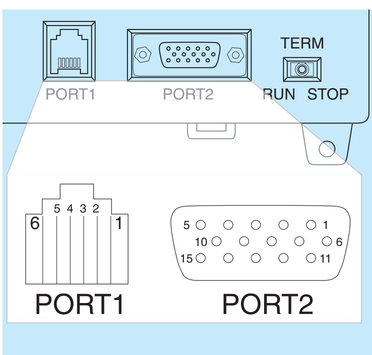
*Note: For information about the MODBUS protocol see the Group Schneider Web site at: www.schneiderautomation.com. At the main menu, select Support/Services, Modbus, Modbus Technical Manuals, PI-MBUS-300 Modbus Protocol Reference Guide or search for PIMBUS300.For more information about the **Direct**NET protocol, order our **Direct**NET user manual, part number DA–DNET–M, or download it free from our Web site: www.automationdirect.com. Select Documentation/Misc./DA-DNET-M.*

DL06 Port Specifications

Communications Port 1	
Port 1	Connects to HPP, DirectSOFT32, operator interfaces, etc.
	6-pin, RS232C
	Communication speed (baud): 9600 (fixed)
	Parity: odd (fixed)
	Station Address: 1 (fixed)
	8 data bits
	1 start, 1 stop bit
	Asynchronous, half-duplex, DTE
	Protocol (auto-select): K-sequence (slave only), DirectNET (slave only), MODBUS (slave only)

Communications Port 2	
Port 2	Connects to HPP, DirectSOFT32, operator interfaces, etc.
	15-pin, multifunction port, RS232C, RS422, RS485
	Communication speed (baud): 300, 600, 1200, 2400, 4800, 9600, 19200, 38400
	Parity: odd (default), even, none
	Station Address: 1 (default)
	8 data bits
	1 start, 1 stop bit
	Asynchronous, half-duplex, DTE
	Protocol (auto-select): K-sequence (slave only), DirectNET (master/slave), MODBUS (master/slave), non-sequence/print/ASCII in/out

DL06 Port Pinouts



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

Port 2 Pin Descriptions		
1	5V	Power (+) connection
2	TXD	Transmit data (RS-232C)
3	RXD	Receive data (RS-232C)
4	RTS	Ready to send
5	CTS	Clear to send
6	RXD-	Receive data (-) (RS-422/485)
7	0V	Power (-) connection (GND)
8	0V	Power (-) connection (GND)
9	TXD+	Transmit data (+) (RS-422/485)
10	TXD-	Transmit data (-) (RS-422/485)
11	RTS+	Ready to send (+) (RS-422/485)
12	RTS-	Ready to send (-) (RS-422/485)
13	RXD+	Receive data (+) (RS-422/485)
14	CTS+	Clear to send (+) (RS-422/485)
15	CTS-	Clear to send (-) (RS-422/485)

Choosing a Network Specification

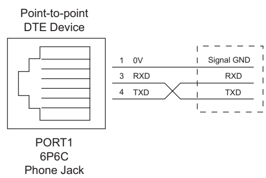
The DL06 PLC's multi-function port gives you the option of using RS-232C, RS-422, or RS-485 specifications. First, determine whether the network will be a 2-wire RS–232C type, a 4-wire RS–422 type, or a 2-wire/4-wire RS-485 type.

The RS–232C specification is simple to implement for networks of shorter distances (15 meters max) and where communication is only required between two devices. The RS–422 and RS-485 signals are for networks that cover longer distances (1000 meters max.) and for multi-drop networks (from 2 to 247 devices).

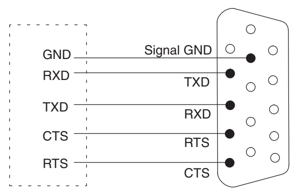
Note: Termination resistors are required at both ends of RS–422 and RS-485 networks. It is necessary to select resistors that match the impedance rating of the cable (between 100 and 500 ohms).

RS-232 Network

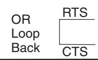
Normally, the RS-232 signals are used for shorter distances (15 meters maximum), for communications between two devices.



Connections on Port 1

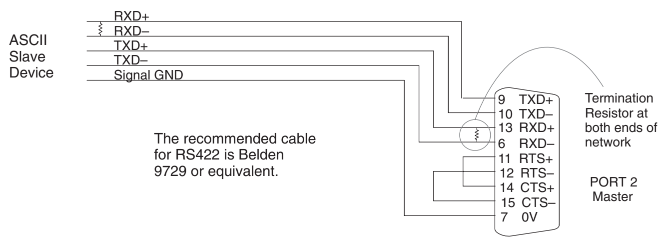


Connections on Port 2



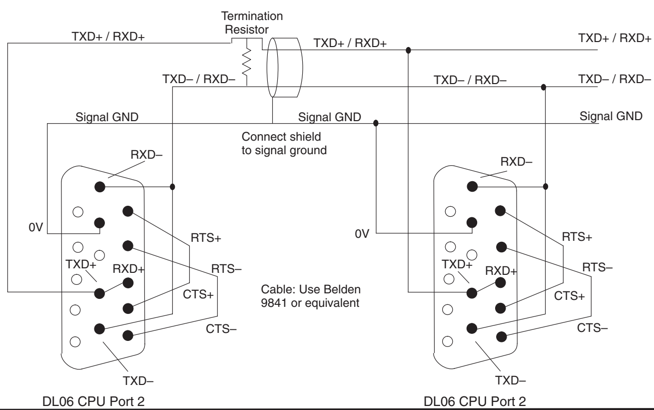
RS-422 Network

RS-422 signals are for long distaces (1000 meters maximum). Use terminator resistors at both ends of RS-422 network wiring, matching the impedence rating of the cable (between 100 and 500 ohms).



RS-485 Network

RS-485 signals are for longer distances (1000 meters max) and for multi-drop networks. Use termination resistors at both ends of RS-485 network wiring, matching the impedance rating of the cable (between 100 and 500 ohms).

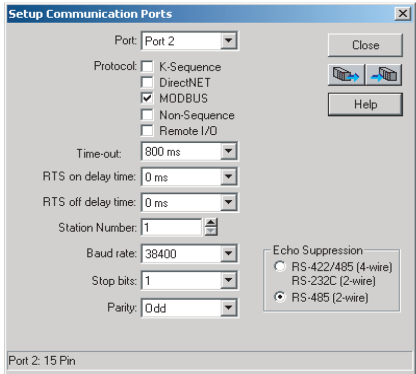


Connecting to MODBUS and *Direct*NET Networks

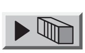
MODBUS Port Configuration

In *Direct*SOFT32, choose the PLC menu, then Setup, then “Secondary Comm Port”.

- **Port**: From the port number list box at the top, choose “Port 2”.
- **Protocol**: Click the check box to the left of “MODBUS” (use AUX 56 on the HPP, and select “MBUS”), and then you’ll see the dialog box below.



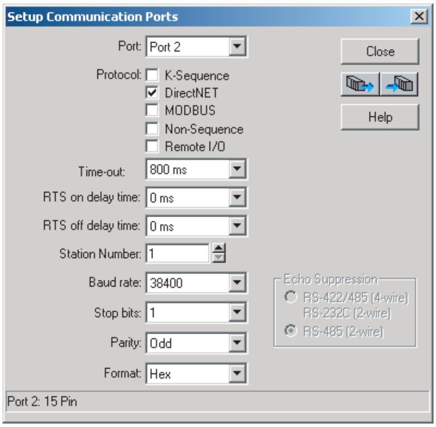
- **Timeout**: amount of time the port will wait after it sends a message to get a response before logging an error.
- **RTS ON / OFF Delay Time:** The RTS ON Delay Time specifies the time the DL06 waits to send the data after it has raised the RTS signal line. The RTS OFF Delay Time specifies the time the DL06 waits to release the RTS signal line after the data has been sent. *When using the DL06 on a multi-drop network, the RTS ON Delay time must be set to at least* ***5ms*** *and the RTS OFF Delay time must be set to at least* ***2ms****. If you encounter problems, the time can be increased.*
- **Station Number**: For making the CPU port a MODBUS master, choose “1”. The possible range for MODBUS slave numbers is from 1 to 247, but the DL06 network instructions used in Master mode will access only slaves 1 to 99. Each slave must have a unique number. At powerup, the port is automatically a slave, unless and until the DL06 executes ladder logic network instructions which use the port as a master. Thereafter, the port reverts back to slave mode until ladder logic uses the port again.
- **Baud Rate**: The available baud rates include 300, 600, 1200, 2400, 4800, 9600, 19200, and 38400 baud. Choose a higher baud rate initially, reverting to lower baud rates if you experience data errors or noise problems on the network. Important: You must configure the baud rates of all devices on the network to the same value. Refer to the appropriate product manual for details.
- **Stop Bits**: Choose 1 or 2 stop bits for use in the protocol.
- **Parity**: Choose none, even, or odd parity for error checking.
- **Echo Suppression:** Select the appropriate radio button based on the wiring configuration used on port 2.

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

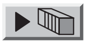
DirectNET Port Configuration

In *Direct*SOFT32, choose the PLC menu, then Setup, then “Secondary Comm Port”.

- **Port**: From the port number list box, choose “Port 2 ”.
- **Protocol**: Click the check box to the left of “DirectNET” (use AUX 56 on the HPP, then select “DNET”), and then you’ll see the dialog box below.



- **Timeout**: Amount of time the port will wait after it sends a message to get a response before logging an error.
- **RTS ON / OFF Delay Time:** The RTS ON Delay Time specifies the time the DL06 waits to send the data after it has raised the RTS signal line. The RTS OFF Delay Time specifies the time the DL06 waits to release the RTS signal line after the data has been sent. When using the DL06 on a multi-drop network, the RTS ON Delay time must be set to at least 5ms and the RTS OFF Delay time must be set to at least 2ms. If you encounter problems, the time can be increased.
- **Station Number**: For making the CPU port a DirectNET master, choose “1”. The allowable range for DirectNET slaves is from 1 to 90 (each slave must have a unique number). At powerup, the port is automatically a slave, unless and until the DL06 executes ladder logic instructions which attempt to use the port as a master. Thereafter, the port reverts back to slave mode until ladder logic uses the port again.
- **Baud Rate**: The available baud rates include 300, 600, 1200, 2400, 4800, 9600, 19200, and 38400 baud. Choose a higher baud rate initially, reverting to lower baud rates if you experience data errors or noise problems on the network. Important: You must configure the baud rates of all devices on the network to the same value.
- **Stop Bits**: Choose 1 or 2 stop bits for use in the protocol.
- **Parity**: Choose none, even, or odd parity for error checking.
- **Format**: Choose between hex or ASCII formats.

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

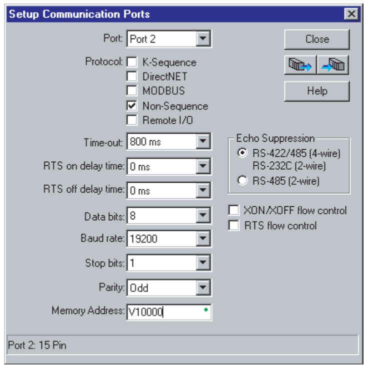
Non–Sequence Protocol (ASCII In/Out and PRINT)

MODBUS Port Configuration

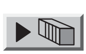
Configuring port 2 on the DL06 for Non–Sequence allows the CPU to use port 2 to either read or write raw ASCII strings using the ASCII instructions. See the ASCII In/Out instructions and the PRINT instruction in chapter 5.

In *Direct*SOFT32, choose the PLC menu, then Setup, then "Secondary Comm Port".

- **Port**: From the port number list box at the top, choose "Port 2".
- **Protocol**: Click the check box to the left of "Non–Sequence".



- **Timeout**: Amount of time the port will wait after it sends a message to get a response before logging an error.
- **RTS On Delay Time**: The amount of time between raising the RTS line and sending the data.
- **RTS Off Delay Time**: The amount of time between resetting the RTS line after sending the data.
- **Data Bits**: Select either 7–bits or 8–bits to match the number of data bits specified for the connected devices.
- **Baud Rate**: The available baud rates include 300, 600, 900, 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 to match the number of stop bits specified for the connected devices.
- **Parity**: Choose none, even, or odd parity for error checking. Be sure to match the parity specified for the connected devices.
- **Echo Suppression**: Select the appropriate radio button based on the wiring configuration used on port 2.
- **Xon/Xoff Flow Control**: Choose this selection if you have port 2 wired for Hardware Flow Control (Xon/Xoff) with RTS and CTS signal connected between all devices.
- **RTS Flow Control**: Choose this selection if you have Port 2 RTS signal wired between all devices.

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

Network Slave Operation

This section describes how other devices on a network can communicate with a CPU port that you have configured as a ***Direct***NETslave or MODBUS slave (DL06). A MODBUS host must use the MODBUS RTU protocol to communicate with the DL06 as a slave. The host software must send a MODBUS function code and MODBUS address to specify a PLC memory location the DL06 comprehends. The ***Direct***NET host uses normal I/O addresses to access applicable DL06 CPU and system. No CPU ladder logic is required to support either MODBUS slave or ***Direct***NET slave operation.

MODBUS Function Codes Supported

The MODBUS function code determines whether the access is a read or a write, and whether to access a single data point or a group of them. The DL06 supports the MODBUS function codes described below.

MODBUS Function Code	Function	DL06 Data Types Available
01	Read a group of coils	Y, CR, T, CT
02	Read a group of inputs	X, SP
05	Set / Reset a single coil	Y, CR, T, CT
15	Set / Reset a group of coils Y,	CR, T, CT
03, 04	Read a value from one or more registers	V
06	Write a value into a single register	V
16	Write a value into a group of registers	V

Determining the MODBUS Address

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

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

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

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

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

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

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

DL06 Memory Type	QTY (Dec.)	PLC Range(Octal)	MODBUS Address Range (Decimal)	MODBUS Data Type
For Discrete Data Types .... Convert PLC Addr. to Dec. + Start of Range + Data Type				
Inputs (X)	512	X0 – X777	2048 – 2559	Input
Special Relays(SP)	512	SP0 – SP777	3072 – 3583	Input
Outputs (Y)	512	Y0 – Y777	2048 – 2559	Coil
Control Relays (CR)	1024	C0 – C1777	3072 – 4095	Coil
Timer Contacts (T)	256	T0 – T377	6144 – 6399	Coil
Counter Contacts (CT)	128	CT0 – CT177	6400 – 6527	Coil
Stage Status Bits(S)	1024	S0 – S1777	5120 – 6143	Coil
For Word Data Types .... Convert PLC Addr. to Dec. + Data Type				
Timer Current Values (V)	256	V0 – V377	0 – 255	Input Register
Counter Current Values (V)	128	V1000 – V1177	512 – 639	Input Register
V Memory, user data (V)	3200	V1200 – V7377	640 – 3839	Holding Register
	4096	V10000 - V17777	4096 - 8191	Holding Register
V Memory, non-volatile (V)	128	V7400 – V7577	3840 – 3967	Holding Register

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

Example 1: V2100

Find the MODBUS address for User V location V2100.

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

Holding Reg 1088

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

Example 2: Y20

Find the MODBUS address for output Y20.

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

Coil 2064

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

Input Reg. 8

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

Coil 3116

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

If Your MODBUS Host Software Requires an Address ONLY

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

- 484 Mode
- 584/984 Mode

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

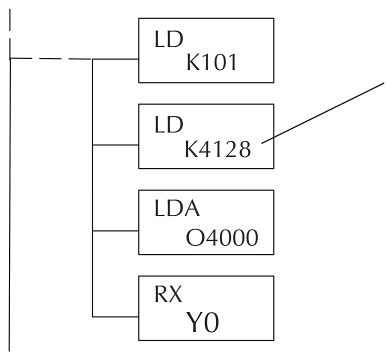
- Discrete – X, SP, Y, CR, S, T (contacts), C (contacts)
- Word – V, Timer current value, Counter current value

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

Discrete Data Types				
DL06 Memory Type	**PLC Range (Octal)**	**Address (484 Mode)**	**Address (584/984 Mode)**	**MODBUS Data Type**
Global Inputs (GX)	GX0-GX1746	1001 - 1999	10001 - 10999	Input
	GX1747-GX3777	---	11000 - 12048	Input
Inputs (X)	X0 – X1777	---	12049 - 13072	Input
Special Relays (SP)	SP0 – SP777	---	13073 - 13584	Input
Global Outputs (GY)	GY0 - GY3777	1 - 2048	1 - 2048	Output
Outputs (Y)	Y0 – Y1777	2049 - 3072	2049 - 3072	Output
Control Relays (CR)	C0 – C3777	3073 - 5120	3073 - 5120	Output
Timer Contacts (T)	T0 – T377	6145 - 6400	6145 - 6400	Output
Counter Contacts (CT)	CT0 – CT377	6401 - 6656	6401 - 6656	Output
Stage Status Bits (S)	S0 – S1777	5121 - 6144	5121 - 6144	Output

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

* MODBUS: Function 04
The DL05/06, DL250-1/260, DL350 and DL450 will support function 04, read input register (Address 30001). To use function 04, put the number '4' into the most significant position (4xxx). Four digits must be entered for the intruction to work properly with this mode.



The Maximum constant possible is 4128. This is due to the 128 maximum number of Bytes that the RX/WX instruction can allow. The value of 4 in the most significant position of the word will cause the RX to use function 04 (30001 range).

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

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

Example 1: V2100 584/984 Mode

Find the MODBUS address for User V location V2100.

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

PLC Address (Dec.) + Mode Address

V2100 = 1088 decimal

1088 + 40001 = **41089**

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

Example 2: Y20 584/984 Mode

Find the MODBUS address for output Y20.

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

PLC Addr. (Dec.) + Start Address + Mode

Y20 = 16 decimal

16 + 2048 + 1 = **2065**

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

Example 3: T10 Current Value 484 Mode

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

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

PLC Address (Dec.) + Mode Address

TA10 = 8 decimal

8 + 3001 = **3009**

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

Example 4: C54 584/984 Mode

Find the MODBUS address for Control Relay C54.

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

PLC Addr. (Dec.) + Start Address + Mode

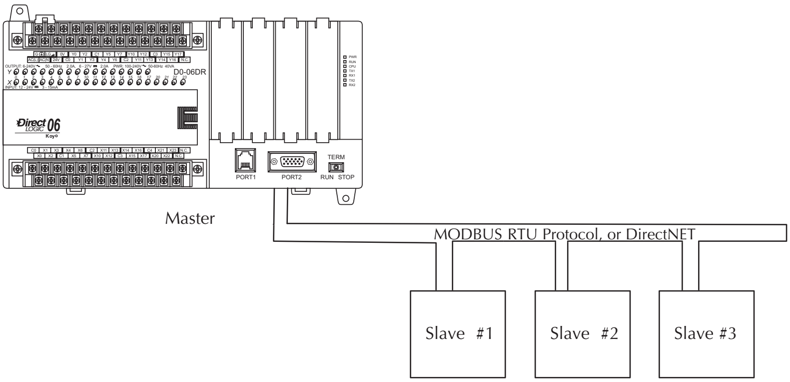
C54 = 44 decimal

44 + 3072 + 1 = **3117**

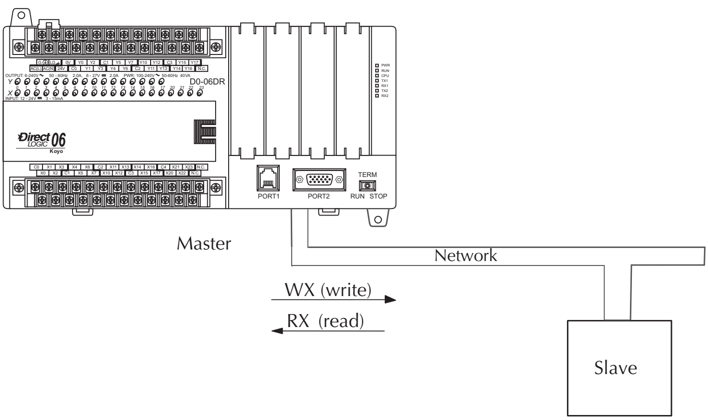
Outputs (Y)	320	Y0 - Y477	2048 - 2367	1	1	Coil
Control Relays (CR)	256	C0 - C377	3072 - 3551	1	1	Coil
Timer Contacts (T)	128	T0 - T177	6144 - 6271	1	1	Coil

Network Master Operation

This section describes how the DL06 can communicate on a MODBUS or *Direct*NET network as a master. For MODBUS networks, it uses the MODBUS RTU protocol, which must be interpreted by all the slaves on the network. Both MODBUS and *Direct*Net are single master/multiple slave networks. The master is the only member of the network that can initiate requests on the network. This section teaches you how to design the required ladder logic for network master operation.



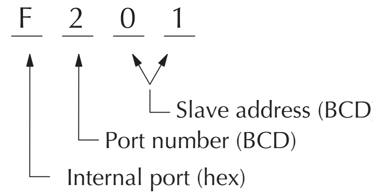
When using the DL06 PLC as the master station, simple RLL instructions are used to initiate the requests. The WX instruction initiates network write operations, and the RX instruction initiates network read operations. Before executing either the WX or RX commands, we will need to load data related to the read or write operation onto the CPU's accumulator stack. When the WX or RX instruction executes, it uses the information on the stack combined with data in the instruction box to completely define the task, which goes to the port.

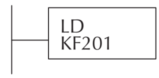


The following step-by-step procedure will provide you the information necessary to set up your ladder program to receive data from a network slave.

Step 1: Identify Master Port # and Slave

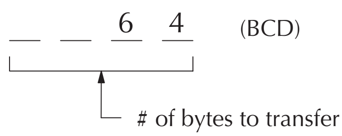
The first Load (LD) instruction identifies the communications port number on the network master (DL06) and the address of the slave station. This instruction can address up to 99 MODBUS slaves, or 90 *Direct*NET slaves. The format of the word is shown to the right. The “F2” in the upper byte indicates the use of the right port of the DL06 PLC, port number 2. The lower byte contains the slave address number in BCD (01 to 99).

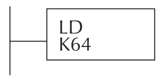




Step 2: Load Number of Bytes to Transfer

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.



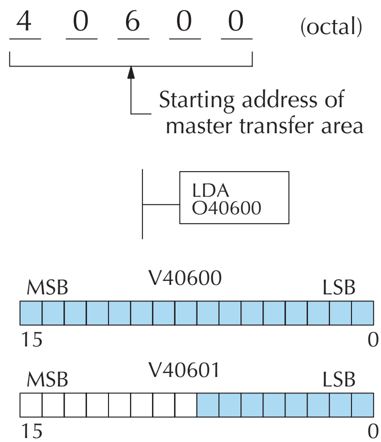


The number of bytes specified also depends on the type of data you want to obtain. For example, the DL06 Input points can be accessed by V-memory locations or as X input locations. However, if you only want X0 – X27, you’ll have to use the X input data type because the V-memory locations can only be accessed in 2-byte increments. The following table shows the byte ranges for the various types of *Direct*LOGIC™. products.

DL05 / 06 / 205 / 350 / 405 Memory	Bits per unit	Bytes
V memory T / C current value	16 16	2 2
Inputs (X, SP)	8	1
Outputs (Y, C, Stage, T/C bits)	8	1
Scratch Pad Memory	8	1
Diagnostic Status	8	1

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

Step 3: Specify Master Memory Area



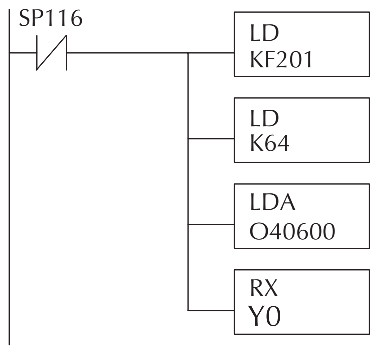
The third instruction in the RX or WX sequence is a Load Address (LDA) instruction. Its purpose is to load the starting address of the memory area to be transferred. Entered as an octal number, the LDA instruction converts it to hex and places the result in the accumulator.

For a WX instruction, the DL06 CPU sends the number of bytes previously specified from its memory area beginning at the LDA address specified.

For an RX instruction, the DL06 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.*

Step 4: Specify Slave Memory Area



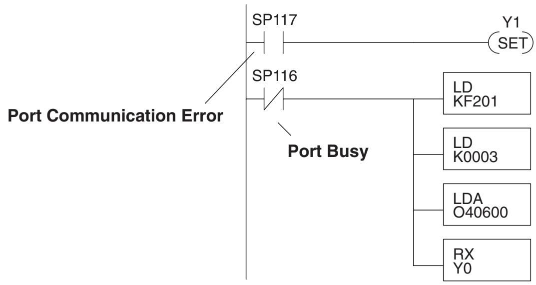
The last instruction in our sequence is the WX or RX instruction itself. Use WX to write to the slave, and RX to read from the slave. All four of our instructions are shown to the right. In the last instruction, you must specify the starting address and a valid data type for the slave.

- ***Direct***NET slaves – specify the same address in the WX and RX instruction as the slave's native I/O address
- MODBUS DL405, DL205, or DL06 slaves – specify the same address in the WX and RX instruction as the slave's native I/O address
- MODBUS 305 slaves – use the following table to convert DL305 addresses to MODBUS addresses

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

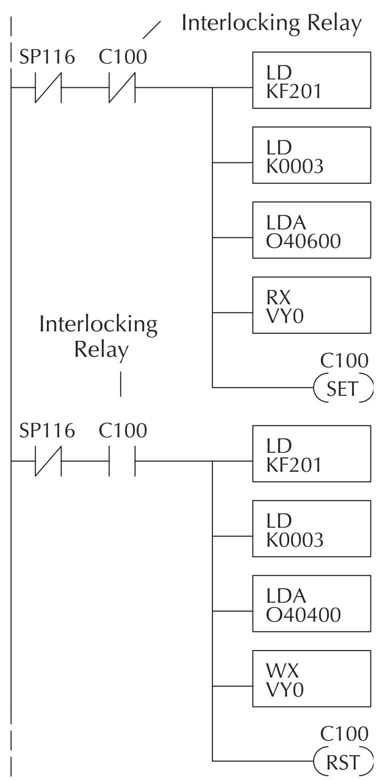
Communications from a Ladder Program

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



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

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



Multiple Read and Write Interlocks

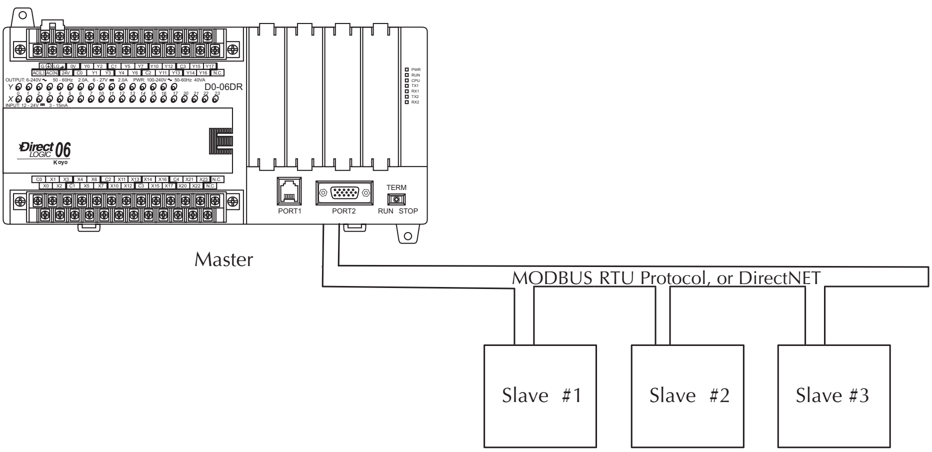
If you are using multiple reads and writes in the RLL program, you have to interlock the routines to make sure all the routines are executed. If you don't use the interlocks, then the CPU will only execute the first routine. This is because each port can only handle one transaction at a time.

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

If you're using RLL*PLUS* 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.

Network Master Operation (using MRX and MWX Instructions)

This section describes how the DL06 can communicate on a MODBUS RTU network as a master using the MRX and MWX read/write instructions. These instructions allow you to enter native MODBUS addressing in your ladder logic program with no need to perform octal to decimal conversions. MODBUS is a single master/multiple slave network. 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.



MODBUS Function Codes Supported

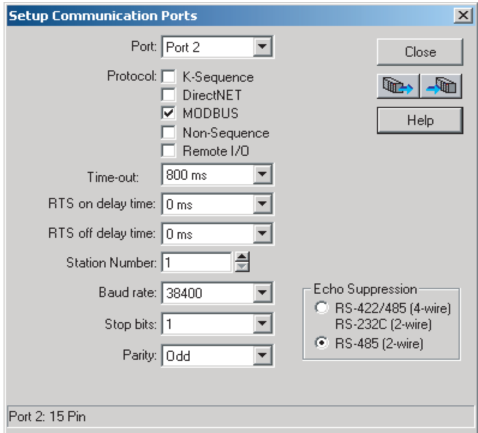
The MODBUS function code determines whether the access is a read or a write, and whether to access a single data point or a group of them. The DL06 supports the MODBUS function codes described below.

MODBUS Function Code	Function	DL06 Data Types Available
01	Read a group of coils	Y, CR, T, CT
02	Read a group of inputs	X, SP
05	Set / Reset a single coil (slave only)	Y, CR, T, CT
15	Set / Reset a group of coils	Y, CR, T, CT
03, 04	Read a value from one or more registers	V
06	Write a value into a single register (slave only)	V
07	Read Exception Status	V
08	Diagnostics	V
16	Write a value into a group of registers	V

MODBUS Port Configuration

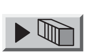
In *Direct*SOFT32, choose the PLC menu, then Setup, then "Secondary Comm Port".

- **Port:** From the port number list box at the top, choose "Port 2".
- **Protocol**: Click the check box to the left of "MODBUS" (use AUX 56 on the HPP, and select "MBUS"), and then you'll see the dialog box below.



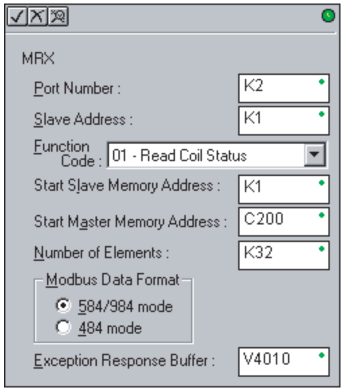
Timeout: Amount of time the port will wait after it sends a message to get a response before logging an error.

- **RTS On Delay Time**: The amount of time between raising the RTS line and sending the data.
- **RTS Off Delay Time**: The amount of time between resetting the RTS line after sending the data.
- **Station Number**: For making the CPU port a MODBUS master, choose "1". The possible range for MODBUS slave numbers is from 1 to 247. Each slave must have a unique number. At powerup, the port is automatically a slave, unless and until the DL06 executes ladder logic MWX/MRX network instructions which use the port as a master. Thereafter, the port reverts back to slave mode until ladder logic uses the port again.
- **Baud Rate**: The available baud rates include 300, 600, 900, 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.
- **Echo Suppression**: Select the appropriate radio button based on the wiring configuration used on port 2.

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

MODBUS Read from Network(MRX)

The MODBUS Read from Network (MRX) instruction is used by the DL06 network master to read a block of data from a connected slave device and to write the data into V–memory addresses within the master. The instruction allows the user the to specify the MODBUS Function Code, slave station address, starting master and slave memory addresses, number of elements to transfer, MODBUS data format and the Exception Response Buffer.



Port Number: must be DL06 Port 2 (K2)

- **Slave Address:** specify a slave station address (0–247)
- **Function Code**: The following MODBUS function codes are supported by the MRX instruction:

  01 – Read a group of coils

  02 – Read a group of inputs

  03 – Read holding registers

  04 – Read input registers

  07 – Read Exception status

  08 – Diagnostics
- **Start Slave Memory Address**: specifies the starting slave memory address of the data to be read. See the table on the following page.
- **Start Master Memory Address**: specifies the starting memory address in the master where the data will be placed. See the table on the following page.
- **Number of Elements**: specifies how many coils, input, holding registers or input register will be read. See the table on the following page.
- **MODBUS Data Format**: specifies MODBUS 584/984 or 484 data format to be used
- **Exception Response Buffer**: specifies the master memory address where the Exception Response will be placed. See the table on the following page.

MRX Slave Memory Address

MRX Slave Address Ranges		
Function Code	**MODBUS Data Format**	**Slave Address Range(s)**
01 – Read Coil	484 Mode	1–999
01 – Read Coil	584/984 Mode	1–65535
02 – Read Input Status	484 Mode	1001–1999
02 – Read Input Status	584/984 Mode	10001–19999 (5 digit) or 100001–165535 (6 digit)
03 – Read Holding Register	484 Mode	4001–4999
03 – Read Holding Register	584/984	40001–49999 (5 digit) or 4000001–465535 (6 digit)
04 – Read Input Register	484 Mode	3001–3999
04 – Read Input Register	584/984 Mode	30001–39999 (5 digit) or 3000001–365535 (6 digit)
07 – Read Exception Status	484 and 584/984 Mode	n/a
08 – Diagnostics	484 and 584/984 Mode	0–65535

MRX Master Memory Addresses

MRX Master Memory Address Ranges	
Operand Data Type	DL06 Range
Inputs X	0–1777
Outputs Y	0–1777
Control Relays C	0–3777
Stage Bits S	0–1777
Timer Bits T	0–377
Counter Bits CT	0–377
Special Relays SP	0–777
V–memory V	All
Global Inputs GX	0–3777
Global Outputs GY	0–3777

MRX Number of Elements

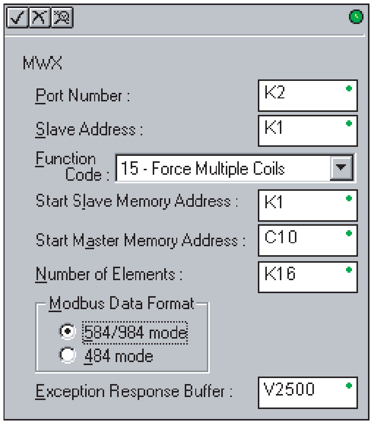
Number of Elements	
Operand Data Type	DL06 Range
V–memory V	All
Constant . . . . K	1–2000

MRX Exception Response Buffer

Exception Response Buffer	
Operand Data Type	DL06 Range
V–memory V	All

MODBUS Write to Network (MWX)

The MODBUS Write to Network (MWX) instruction is used to write a block of data from the network masters's (DL06) memory to MODBUS memory addresses within a slave device on the network. The instruction allows the user to specify the MODBUS Function Code, slave station address, starting master and slave memory addresses, number of elements to transfer, MODBUS data format and the Exception Response Buffer.



Port Number: must be DL06 Port 2 (K2)

- **Slave Address**: specify a slave station address (0–247)
- **Function Code**: The following MODBUS function codes are supported by the MWX instruction:

  05 – Force Single coil

  06 – Preset Single Register

  08 – Diagnostics

  15 – Force Multiple Coils

  16 – Preset Multiple Registers
- **Start Slave Memory Address**: specifies the starting slave memory address where the data will be written.
- **Start Master Memory Address:** specifies the starting address of the data in the master that is to written to the slave.
- **Number of Elements**: specifies how many consecutive coils or registers will be written to. This field is only active when either function code 15 or 16 is selected.
- **MODBUS Data Format**: specifies MODBUS 584/984 or 484 data format to be used.
- **Exception Response Buffer**: specifies the master memory address where the Exception Response will be placed.

 Chapter 4: CPU Specifications and Operation

MWX Slave Memory Address

MWX Slave Address Ranges		
Function Code	**MODBUS Data Format**	**Slave Address Range(s)**
05 – Force Single Coil	484 Mode	1–999
05 – Force Single Coil	584/984 Mode	1–65535
06 – Preset Single Register	484 Mode	4001–4999
06 – Preset Single Register	84/984 Mode	40001–49999 (5 digit) or 400001–465535 (6 digit)
08 – Diagnostics	484 and 584/984 Mode	0–65535
15 – Force Multiple Coils	484	1–999
15 – Force Multiple Coils	585/984 Mode	1–65535
16 – Preset Multiple Registers	484 Mode	4001–4999
16 – Preset Multiple Registers	584/984 Mode	40001–49999 (5 digit) or 4000001–465535 (6 digit)

MWX Master Memory Addresses

MRX Master Memory Address Ranges		
Operand Data Type		DL06 Range
Inputs	X	0–777
Outputs	Y	0–777
Control Relays	C	0–1777
Stage Bits	S	0–1777
Timer Bits	T	0–377
Counter Bits	CT	0–177
Special Relays.	SP	0–777
V–memory	V	All
Global Inputs	GX	0–3777
Global Outputs	GY	0–3777

MWX Number of Elements

Number of Elements		
Operand Data Type		DL06 Range
V–memory	V	All
Constant	K	1–2000

MWX Exception Response Buffer

Exception Response Buffer		
Operand Data Type		DL06 Range
V–memory	V	All

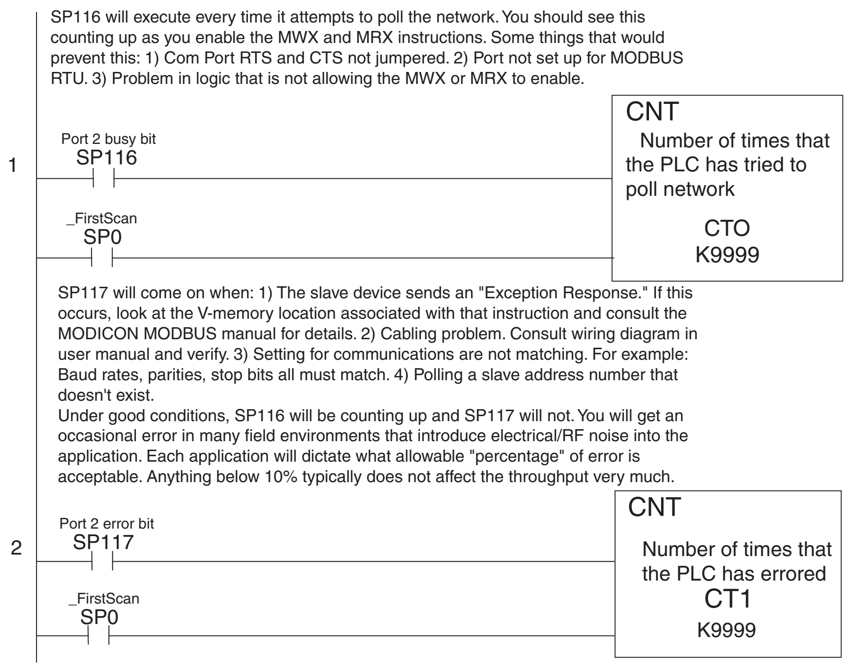
MRX / MWX Example in DirectSOFT32

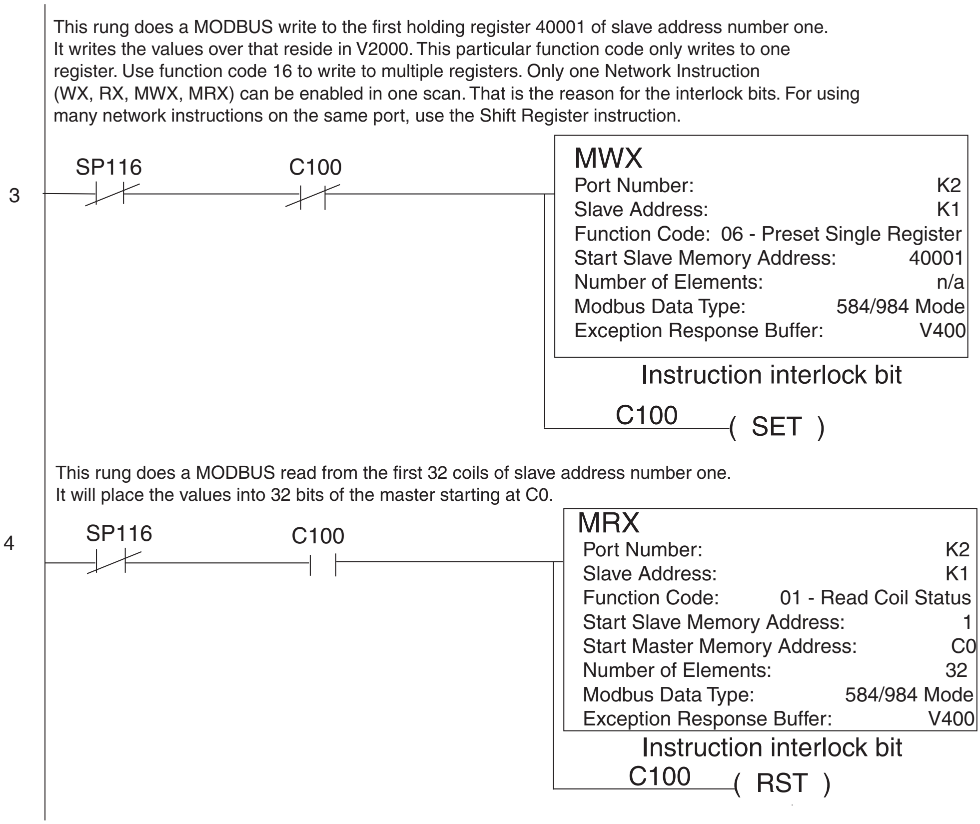
DL06 port 2 has two Special Relay contacts associated with it (see Appendix D for comm port special relays). One indicates “Port busy”(SP116), and the other indicates ”Port Communication Error”(SP117). The “Port Busy” bit is on while the PLC communicates with the slave. When the bit is off the program can initiate the next network request. The “Port Communication Error” bit turns on when the PLC has detected an error and 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 MRX or MWX instruction is executed. Typically network communications will last longer than 1 CPU scan. The program must wait for the communications to finish before starting the next transaction.

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.




This rung does a MODBUS write to the first holding register 40001 of slave address number one.
It writes the values over that reside in V2000. This particular function code only writes to one
register. Use function code 16 to write to multiple registers. Only one Network Instruction
(WX, RX, MWX, MRX) can be enabled in one scan. That is the reason for the interlock bits. For using
many network instructions on the same port, use the Shift Register instruction.
3
SP116
C100
MWX
Port Number: K2
Slave Address: K1
Function Code: 06 - Preset Single Register
Start Slave Memory Address: 40001
Number of Elements: n/a
Modbus Data Type: 584/984 Mode
Exception Response Buffer: V400
Instruction interlock bit
C100
(SET)
This rung does a MODBUS read from the first 32 coils of slave address number one.
It will place the values into 32 bits of the master starting at C0.
4
SP116
C100
MRX
Port Number: K2
Slave Address: K1
Function Code: 01 - Read Coil Status
Start Slave Memory Address: 1
Start Master Memory Address: C0
Number of Elements: 32
Modbus Data Type: 584/984 Mode
Exception Response Buffer: V400
Instruction interlock bit
C100
(RST)