Errata Sheet

This Errata Sheet contains corrections or changes made after the publication of this manual.

Product Family: DL305

Manual Number D3-350-M

Revision and Date 2nd Edition, Rev. D; March 2010

Date: 07/28/2021

07.2021

Handheld Programmer D3-HP has been discontinued. Please consider Productivity, BRX, or CLICK series PLCs as an alternative platform.

Handheld Programmer D3-HPP was discontinued 01/2018.

08.2018

Changes to Chapter 5. Standard RLL Instructions; Timer, Counter and Shift Register; Accumulating Timer (TMRA) and Accumulating Fast Timer (TMRAF)

Page 5-38. In the first paragraph, second sentence on this page, the maximum value for the TMRAF instruction reads 99999.99 (total of seven 9's). This is incorrect - it should be 999999.99 (total of eight 9's) .

05.2018

Changes to Chapter 3. CPU Specifications and Operations

Page 3-6. Using Battery Backup; Enabling the Battery Backup

The ladder example shown to enable the backup battery is incorrect. Use the following example instead. Note: This example assumes that the original content of V7633 was a 0 (zero),

Changes to Chapter 5. Standard RLL Instructions; Accumulator Logical Instructions; Compare Real Number (CMPR)

Page 5-76. In the ladder example, the contact reference is incorrect. It should be SP62 turning on output C1, not SP60.

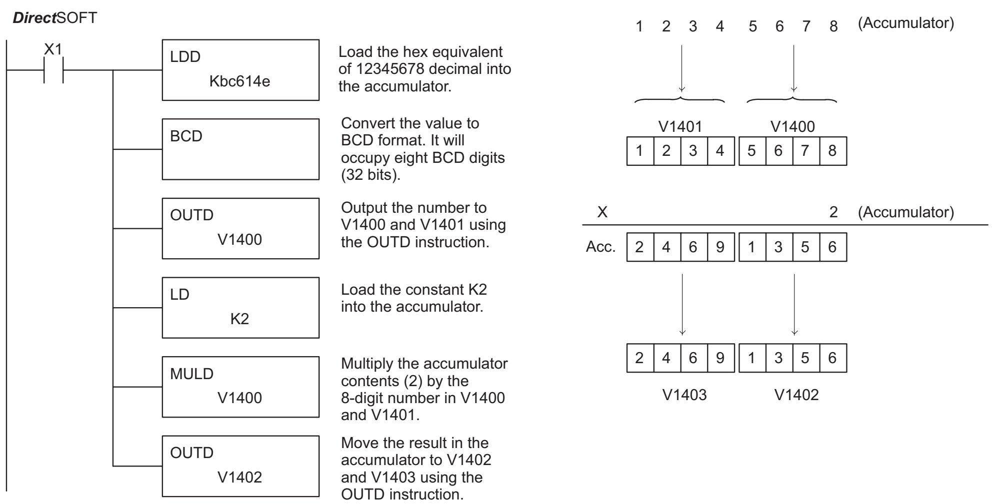
Errata Sheet

This Errata Sheet contains corrections or changes made after the publication of this manual.

05.2018, cont'd

Changes to Chapter 5. Standard RLL Instructions; Math Instructions; Multiply Double (MULD)

Page 5-84. The ladder example shown is incorrect. Replace it with the following example.



Handheld Programmer Keystrokes

$ STR	→	B 1	ENT												
SHFT	L ANDST	D 3	D 3	→	PREV	SHFT	B 1	C 2	SHFT	G 6	B 1	E 4	SHFT	E 4	ENT
SHFT	B 1	C 2	D 3	ENT											
GX OUT	SHFT	D 3	→	B 1	E 4	A 0	A 0	ENT							
SHFT	L ANDST	D 3	→	PREV	C 2	ENT									
SHFT	M ORST	U ISG	L ANDST	D 3	→	B 1	E 4	A 0	A 0	ENT					
GX OUT	SHFT	D 3	→	B 1	E 4	A 0	C 2	ENT							

Changes to Chapter 5. Standard RLL Instructions; Network Instructions

Please change Step 1 on pages 5-137 and 5-139 (RX & WX Commands) to read as follows:

Step 1: - Load the slave address (0-90 BCD) into the first byte and the slot number of the master DCM (0-7) into the second byte of the second level of the accumulator stack. When using Port 2 of the CPU, the formatting should be Kf1xx where xx is the slave address (0-90 BCD).

Errata Sheet

This Errata Sheet contains corrections or changes made after the publication of this manual.

05.2018, cont'd

Changes to Chapter 8. PID Loop Operation

For more recent and complete information on PID loop operation, refer to Chapter 8 in the DL06 user manual (p/n D0-06USER-M).

Changes to Chapter 9. Maintenance and Troubleshooting

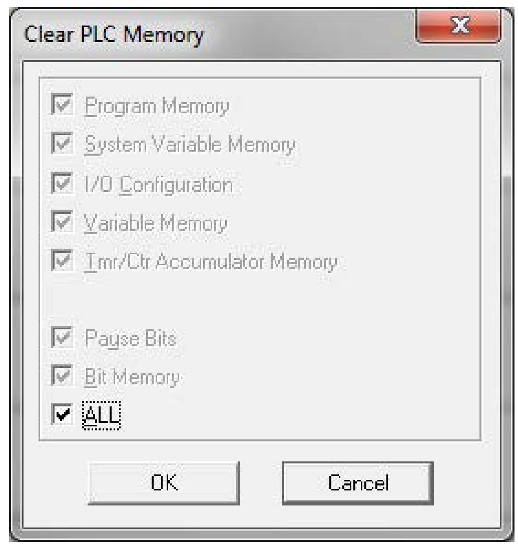
Page 9-25. Add the following to the end of this chapter (right after Regular Forcing with Direct Access):

Reset the PLC to Factory Defaults

NOTE: Resetting to factory defaults will not clear any password stored in the PLC**.**

Resetting a DirectLogic PLC to Factory Defaults is a two-step process. Be sure to have a verified backup of your program using "Save Project to Disk" from the File menu before performing this procedure. Please be aware that the program as well as any settings will be erased and not all settings are stored in the project. In particular you will need to write down any settings for Secondary Communications Ports and manually set the ports up after resetting the PLC to factory defaults.

Step 1 – While connected to the PLC with DirectSoft, go to the PLC menu and select; "Clear PLC Memory". Check the "ALL" box at the bottom of the list and press "OK".



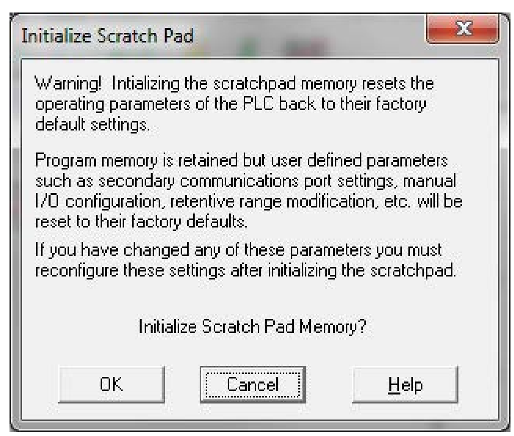
Step 2 – While connected with DirectSoft, go the PLC menu and then to the "Setup" submenu and select "Initialize Scratch Pad". Press "Ok".

NOTE: All configurable communications ports will be reset to factory default state. If you are connected via Port 2 or another configurable port, you may be disconnected when this operation is complete.

NOTE: Retentive ranges will be reset to the factory settings.

NOTE: Manually addressed IO will be reset to factory default settings.

The PLC has now been reset to factory defaults and you can proceed to program the PLC.



Errata Sheet

This Errata Sheet contains corrections or changes made after the publication of this manual.

06.14.2012

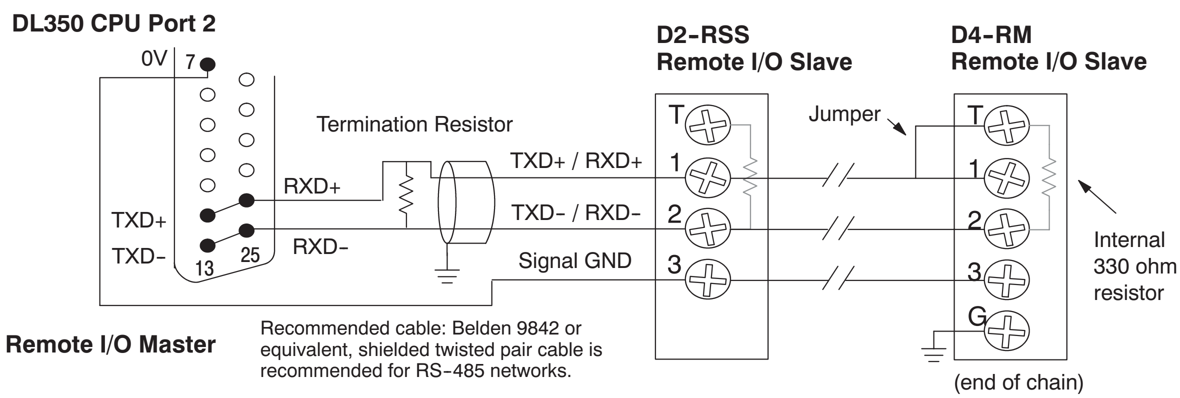
Changes to Chapter 4. System Design and Configuration

Page 4-15. I/O Configurations with a 10 Slot Local CPU Base

The four drawings on this page to the left of the bases showing jumper switch SW2 are mislabeled. They should say "700" and "100 EXP", not "700 EXP" and "100" as shown.

Page 4-18. CPU Specifications; Remote I/O Expansion; Configuring the CPU's Remote I/O Channel

Replace the remote I/O wiring diagram shown with this one.



CPU Specifications and Operations

3

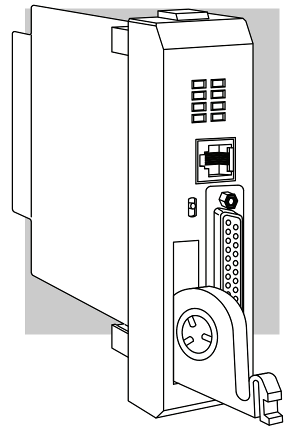
In This Chapter. . . .

Handheld Programmer D3-HP & D3-HPP have been retired as of 03/2021 & 01/2018 respectively. Please consider Productivity, BRX, or CLICK series PLC systems as upgrades.

Overview

The CPU is the heart of the control system. Almost all system operations are controlled by the CPU, so it is important that it is set-up and installed correctly. This chapter provides the information needed to understand:

- the differences between the different models of CPUs
- the steps required to setup and install the CPU



General CPU Features

The DL350 is a modular CPU which can be installed in 5, 8, or 10 slot bases. All I/O modules in the DL305 family will work with the CPU. The DL350 CPU offers a wide range of processing power and program RLL and Stage program instructions (see Chapters 5 and 7). It also provides extensive internal diagnostics that can be monitored from the application program or from an operator interface. The DL350 is different than the other CPUs in the DL305 family. It supports a 16 bit addressing format where the DL330/340 are 8 bit. This has enabled the DL350 to expanded its instruction set, memory, and features much like the DL205 and DL405 CPUs.

DL350 CPU Features

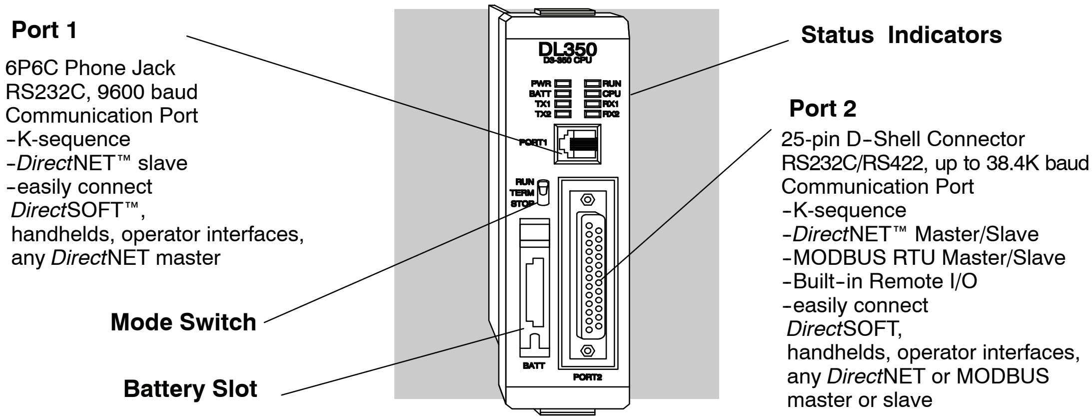
The DL350 has a maximum of 14.8K of program memory comprised of 7.6K of ladder memory and 7.2K of V-memory (data registers). It supports a maximum of 368 points of local I/O, and 880 points with remote I/O. It includes an additional internal RISC-based microprocessor for greater processing power. The DL350 has over 150 instructions, including drum timers, a print function, floating point math, and PID loop control for 4 loops.

The DL350 has a total of two communications ports. The top port is a 6 pin modular that provides a built-in RS232 communication port. It can be used for easy connection of the handheld programmer, PC, or used for a ***Direct***NET slave. The bottom port is a 25-pin RS232C/RS422 port. It will interface with ***Direct***SOFT, and operator interfaces, provides built-in Remote I/O, ***Direct***NET and MODBUS RTU Master/Slave connections.

CPU General Specifications

Feature	DL350
Total Program memory (words)	14.8K
Ladder memory (words)	7680 (Flash)
V-memory (words)	7168
Non-volatile V–Memory (words)	No
Boolean execution /K	5–6 ms
RLL and RLL*PLUS* Programming	Yes
Handheld programmer	Yes
DirectSOFT™ programming for Windows™	Yes
Built-in communication ports (RS232C)	Yes
CMOS RAM	No
UVPROM	No
EEPROM	Flash
Local Discrete I/O points available	368
Remote I/O points available	512
Remote I/O Channels	1
Max Number of Remote Slaves	7
Local Analog input / output channels maximum	128 / 32
Counter Interface Module (quad., pulse out, pulse catch, etc.)	No
I/O Module Point Density	8/16
Slots per Base	5/8/10
Number of instructions available (see Chapter 5 for details)	170
Control relays	1024
Special relays (system defined)	144
Stages in RLL*PLUS*	1024
Timers	256
Counters	128
Immediate I/O	Yes
Interrupt input (hardware / timed)	No / Yes
Subroutines	Yes
Drum Timers	Yes
For/Next Loops	Yes
Math	Integer,Floating Point
PID Loop Control, Built In	Yes
Time of Day Clock/Calendar	Yes
Run Time Edits	Yes
Supports Overrides	Yes
Internal diagnostics	Yes
Password security	Yes
System error log	Yes
User error log	Yes
Battery backup	Yes (optional)

CPU Hardware Features



Mode Switch Functions

The mode switch on the DL350 CPUs provide 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***SOFT programing package or operator interface). 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 programming or monitoring device.

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 (Stop Program)	CPU is forced into the STOP mode. No change or monitoring is allowed by the programming/monitoring device.

There are two ways to change the CPU mode.

1. Use the CPU mode switch to select the operating mode.
2. Place the CPU mode switch in the TERM position and use a programming device to change operating modes. In this position, you can change between Run and Program modes.

Status Indicators

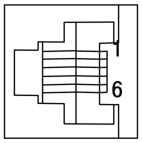
The status indicator LEDs on the CPU front panels have specific functions which can help in programming and troubleshooting.

Indicator	Status	Meaning
PWR	ON	Power good
RUN	ON	CPU is in Run Mode
RUN	FLASHING	CPU is in Firmware upgrade mode
CPU	ON	CPU self diagnostics error
BATT	ON	CPU battery voltage is low
TX1	ON	Transmitting Data from Port 1
RX1	ON	Receiving Data at port 1
TX2	ON	Transmitting Data from Port 2
RX2	ON	Receiving Data at Port 2

Port 1 Specifications

The operating parameters for Port 1 on the DL350 CPU are fixed.

- 6 Pin female modular (RJ12 phone jack) type connector
- ***Direct***Net (slave), K-sequence protocol
- RS232C, 9600 baud
- Connect to ***Direct***SOFT, D2-HPP, DV1000 or ***Direct***NET master



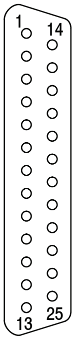
6-pin Female
Modular Connector

Port 1 Pin Descriptions (DL350 only)		
1	0V	Power (-) connection (GND)
2	5V	Power (+) connection
3	RXD	Receive Data (RS232C)
4	TXD	Transmit Data (RS232C
5	5V	Power (+) connection
6	0V	Power (-) connection (GND)

Port 2 Specifications

Port 2 on the DL350 CPU is located on the 25 pin D-shell connector. It is configurable using AUX functions on a programming device.

- 25 Pin female D type connector
- Protocol: K sequence, ***Direct***NET Master/Slave, MODBUS RTU Master/Slave, Remote I/O, non-procedure
- RS232C, non-isolated, distance within 15 m (approx. 50 feet)
- RS422C, non-isolated, distance within 1000 m
- Up to 38.4K baud
- Address selectable (1-90)
- Connects to ***Direct***SOFT, operator interfaces, any ***Direct***NETor MODBUS master or slave



25-pin Female
D Connector

Port 2 Pin Descriptions (DL350 CPU)		
1	not used	
2	TXD	Transmit Data (RS232C)
3	RXD	Receive Data (RS232C)
4	RTS	Ready to Send (RS-232C)
5	CTS	Clear to Send (RS-232C)
6	not used	
7	0V	Power (-) connection (GND)
8	0V	Power (-) connection (GND)
9	RXD +	Receive Data + (RS-422)
10	RXD -	Receive Data (RS-422)
11	CTS +	Clear to Send + (RS422)
12	TXD +	Transmit Data + (REMIO)
13	TXD -	Transmit Data - (REMIO)

Port 2 Pin Descriptions (Cont'd)		
14	TXD +	Transmit Data + (RS-422
15	not used	
16	TXD -	Transmit Data - (RS-422)
17	not used	
18	RTS -	Request to Send - (RS-422)
19	RTS +	Request to Send - (RS-422)
20	not used	
21	not used	
22	not used	
23	CTS -	Clear to Send - (RS-422)
24	RXD +	Receive Data + (REMIO)
25	RXD -	Receive Data - (REMIO)

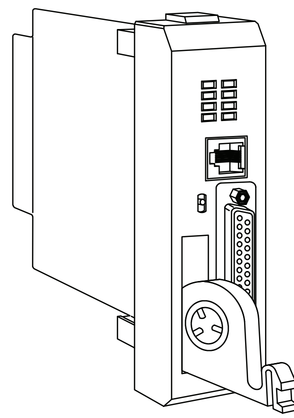
Using Battery Backup

An optional lithium battery is available to maintain the system RAM retentive memory when the DL305 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 shutdown 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***SOFT 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 DL350 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.



DL350

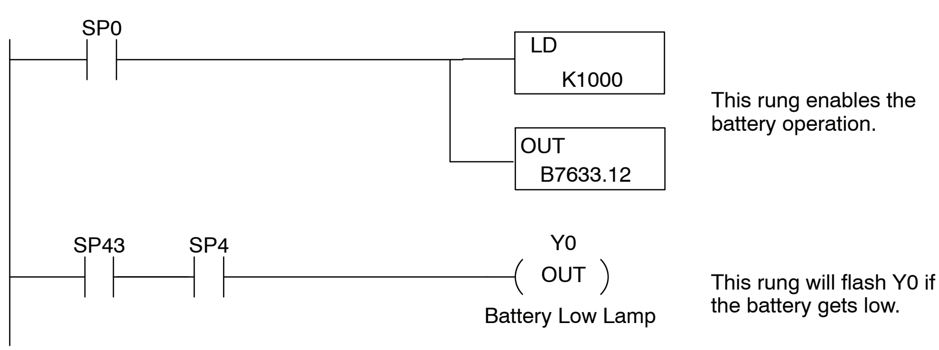
WARNING: Do not attempt to recharge the battery or dispose of an old battery by fire. The battery may explode or release hazardous materials.

Enabling the Battery Backup

The battery can be enabled by setting bit 12 in V7633 (B7633.12) ON (see example below). In this mode the battery Low LED will come on when the battery voltage is less than 2.5VDC (SP43) and error E41 will occur. In this mode the CPU will maintain the data in C,S,T,CT, and V–memory when power is removed from the CPU, provided the battery is good. The use of a battery can also determine which operating mode is entered when the system power is connected. See CPU Setup, which is discussed later in this chapter.

If you have installed a battery, the battery circuit can be disabled by turning OFF B7633.12. However, if you have a battery installed and select "No Battery" operation, the battery LED will not turn on if the battery voltage is low.

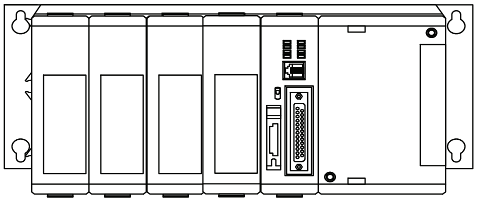
This ladder example is incorrect. Use the example shown on the Errata Sheet on the first page of this file.



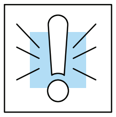
CPU Setup

Installing the CPU

The CPU **must** be installed in the first slot in the base (closest to the power supply). You cannot install the CPU in any other slot. When inserting the CPU into the base, align the PC board with the grooves on the top and bottom of the base. Push the CPU straight into the base until it is firmly seated in the backplane connector.



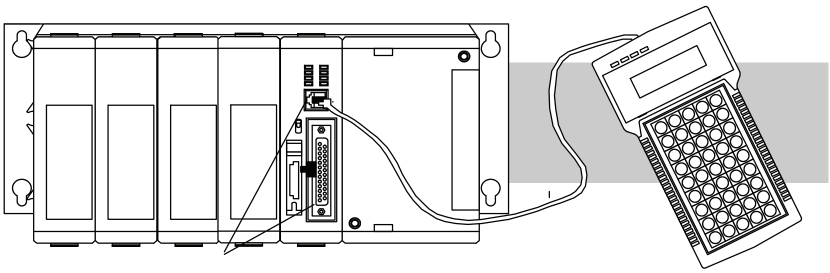
CPU must reside in first slot!



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

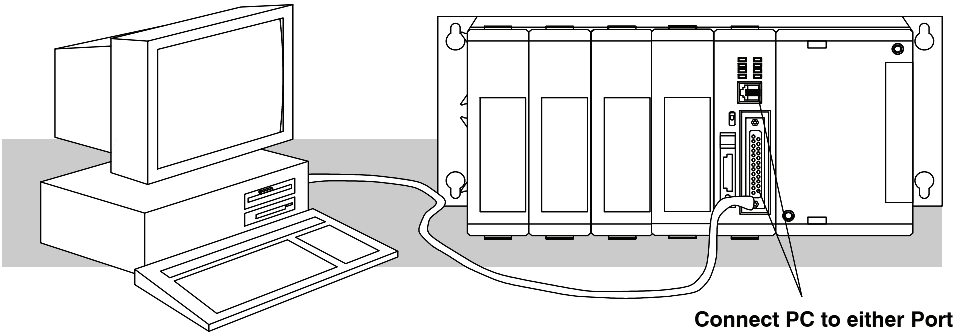
Connecting the Programming Devices

The Handheld programmer is connected to the CPU with a handheld programmer cable. You can connect the Handheld to port 1 on a DL350 CPU. The handheld programmer is shipped with a cable. The cable is approximately 6.5 feet (200 cm).



Connect Handheld to Port 1

If you are using a Personal Computer with the ***Direct***SOFT™ programming package, you can use either the top or bottom port.



Connect PC to either Port

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***SOFT™ or from the 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***SOFT package. The following table shows a list of the Auxiliary functions for the different CPUs and the Handheld Programmer. Note, the Handheld Programmer may have additional AUX functions that are not supported with the DL305 CPUs.

AUX Function and Description		350	HPP
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	×	–
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 CPU Network Address	✓	–
57	Set Retentive Ranges	✓	–
58	Test Operations	×	–
59	Bit Override	×	–
5B	Counter Interface Config.	×	–
5C	Display Error History	✓	–

AUX Function and Description		350	HPP
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	✓	–

✓ supported

× not supported

– not applicable

Clearing an Existing Program

Before you enter a new program, you should 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

Setting the Clock and Calendar

The DL350 also has a Clock / Calendar that can be used for many purposes. If you need to use this feature there are also AUX functions available that allow you set the date and time. For example, you would use AUX 52, Display/Change Calendar to set the time and date with the Handheld Programmer. With ***Direct***SOFT you would use the PLC Setup menu options using K–Sequence protocol only.

The CPU uses the following format to display the date and time.

- Date — Year, Month, Date, Day of week (0 – 6, Sunday thru Saturday)
- Time — 24 hour format, Hours, Minutes, Seconds

Handheld Programmer Display

```
23:08:17  97/05/20
```

You can use the AUX function to change any component of the date or time. However, the CPU will not automatically correct any discrepancy between the date and the day of the week. For example, if you change the date to the 15th of the month and the 15th is on a Thursday, you will also have to change the day of the week (unless the CPU already shows the date as Thursday). The day of the week can only be set using the handheld programmer.

Initializing System Memory

The DL350 CPU maintains system parameters in a memory area 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.

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 change from program to program 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 you that you have considered all ramifications of this operation before you select it.

Setting the CPU Network Address

The DL350 CPU has a built in ***Direct***NET port. You can use the Handheld Programmer to set the network address for the port and the port communication parameters. The default settings are:

- Station Address 1
- Hex Mode
- Odd Parity
- 9600 Baud

The ***Direct***NET Manual provides additional information about choosing the communication settings for network operation.

Setting Retentive Memory Ranges

The DL350 CPU provides 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	DL350	
	Default Range	Avail. Range
Control Relays	C1000 – C1777	C0 – C1777
V–Memory	V1400 – 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 ***Direct***SOFT™ menus to select the retentive ranges.

WARNING: The DL350 CPU does 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.

Password Protection

The DL350 CPU allows you to use a password to help minimize the risk of unauthorized program and/or data changes. The DL350 offers multi–level passwords for even more security. Once you enter a password you can "lock" the CPU 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 CPUs 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 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.

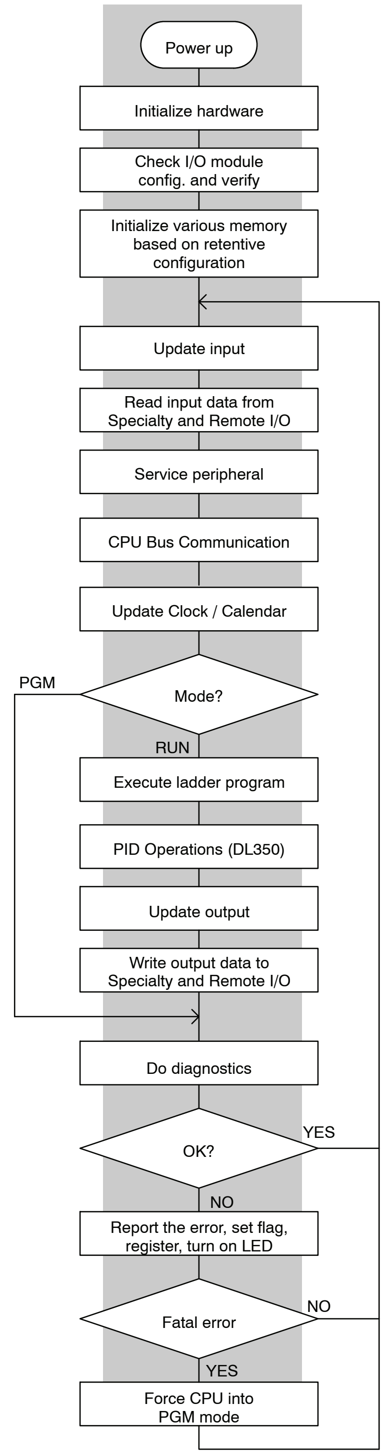
For more information on passwords, see Appendix A, Auxiliary Functions, Aux 8* – Password Operations.

WARNING: Make sure you remember your password. If you forget your password you will not be able to access the CPU. The CPU must be returned to AutomationDirect to have the entire memory cleared in order to clear the password which is the policy of the AutomationDirect.

CPU Operation

Achieving the proper control for your equipment or process requires a good understanding of how DL350 CPUs control all aspects of system operation. The flow chart below shows the main tasks of the CPU operating system. In this section, we will investigate four aspects of CPU operation:

- CPU Operating System — the CPU manages all aspects of system control.
- CPU Operating Modes — The three primary modes of operation are Program Mode, Run Mode, and Test Mode.
- CPU Timing — The two important areas we discuss are the I/O response time and the CPU scan time.
- CPU Memory Map — The CPUs memory map shows the CPU addresses of various system resources, such as timers, counters, inputs, and outputs.



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.

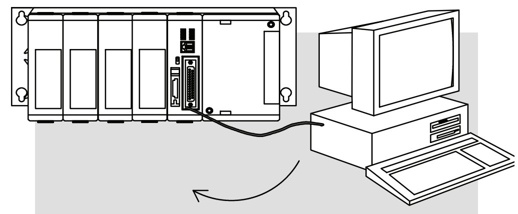
The outputs are only updated in Run mode. In program mode, they are in the off state.

In Run Mode, the CPU executes the user ladder program. Immediately afterwards, any PID loops which are configured are executed (DL350 only). Then the CPU writes the output results of these two tasks to the appropriate output points.

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 Operation

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



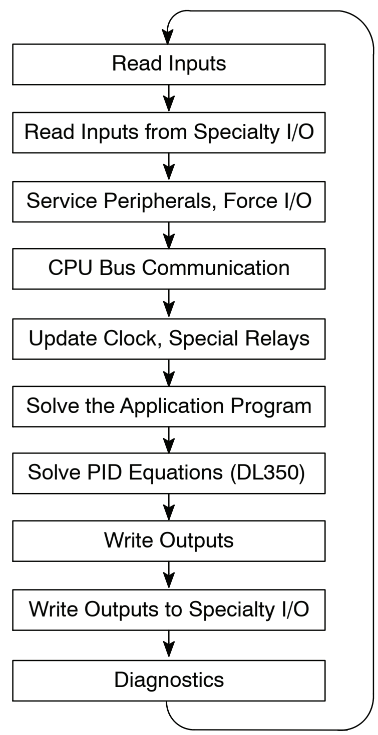
Download Program

You can use the mode switch on the DL350 CPU to select Program Mode operation. Or, with the switch in TERM position, you can use a programming device such as the Handheld Programmer to place the CPU in Program Mode.

Run Mode Operation

In Run Mode, the CPU executes the application program, does PID calculations for configured PID loops (DL350 only), and updates the I/O system. You can perform many operations during Run Mode. Some of these include:

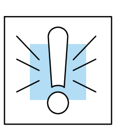
- Monitor and change I/O point status
- Update timer/counter preset values
- Update Variable memory locations



Run Mode operation can be divided into several key areas. It is very important you understand how each of these areas of execution can affect the results of your application program solutions.

You can use the mode switch to select Run Mode operation. Or, with the mode switch in TERM position, you can use a programming device, such as the 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". 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.



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.

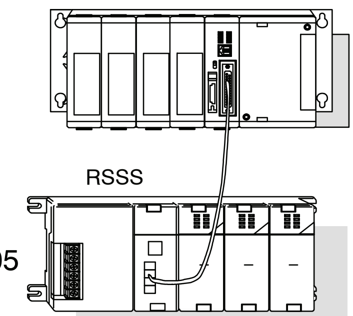
Read Inputs

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 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 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.

Read Inputs from Specialty and Remote I/O

After the CPU reads the inputs from the input modules, it reads any input point data from any Specialty modules that are installed. This is also the portion of the scan that reads the input status from Remote I/O racks.



NOTE: It may appear the Remote I/O point status is updated every scan. This is not quite true. The CPU will receive information from the Remote I/O Master module every scan, but the Remote Master may not have received an update from all the Remote slaves. Remember, the Remote I/O link is managed by the Remote Master, not the CPU.

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.

- Forcing from a peripheral – not a permanent force, good only for one scan

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.

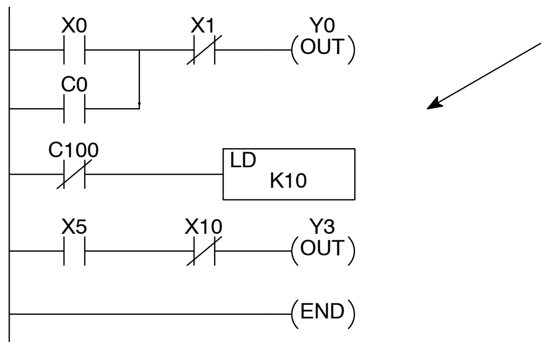
Update Clock, Special Relays, and Special Registers

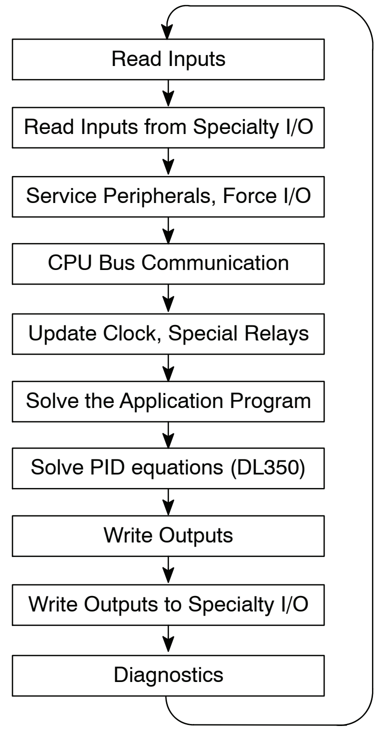
The DL350 CPUs has an internal real-time clock and calendar timer which is accessible to the application program. Special V-memory locations hold this information. This portion of the execution cycle makes sure these locations get updated on every scan. Also, there are several different Special Relays, such as diagnostic relays, etc., that are also updated during this segment.

Solve Application Program

The CPU evaluates each instruction in the application program during this segment of the scan cycle. The instructions define the relationship between input conditions and the system outputs.

The CPU begins with the first rung of the ladder program, evaluating it from left to right and from top to bottom. It continues, rung by rung, until it encounters the END coil instruction. At that point, a new image for the outputs is complete.





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

You may recall the CPU may have obtained and stored forcing information when it serviced the peripheral devices. If any I/O points or memory data have been forced, the output image register also contains this information.

NOTE: If an output point was used in the application program, the results of the program solution will overwrite any forcing information that was stored. For example, if Y0 was forced on by the programming device, and a rung containing Y0 was evaluated such that Y0 should be turned off, then the output image register will show that Y0 should be off. Of course, you can force output points that are not used in the application program. In this case, the point remains forced because there is no solution that results from the application program execution.

Solve PID Loop Equations

The DL350 CPU can process up to 4 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 located in the local CPU base or the local expansion bases. Remember, the CPU also made sure 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 and Remote 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. For example, this is the portion of the scan that writes the output status from the image register to the Remote I/O racks.

NOTE: It may appear the Remote I/O point status is updated every scan. This is not quite true. The CPU will send the information to the Remote I/O Master module every scan, but the Remote Master will update the actual remote modules during the next communication sequence between the master and slave modules. Remember, the Remote I/O link communication is managed by the Remote Master, not the CPU.

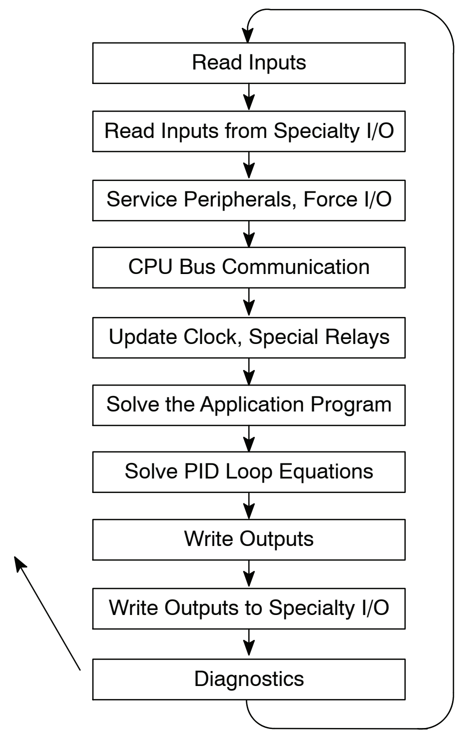
Diagnostics

During this part of the scan, the CPU performs all system diagnostics and other tasks, such as:

- calculating the scan time
- updating special relays
- resetting the watchdog timer

The DL350 CPU automatically detects and reports many different error conditions. Appendix B contains a listing of the various error codes available with the DL305 system.

One of the more important diagnostic tasks is the scan time calculation and watchdog timer control. The DL350 CPU has a “watchdog” timer that stores the maximum time allowed for the CPU to complete the solve application segment of the scan cycle. The default value set from the factory is 200 mS. If this time is exceeded the CPU will enter the Program Mode, turn off all outputs, and report the error. For example, the Handheld Programmer displays “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. There is also an RSTWT instruction that can be used in the application program to reset the watch dog timer during the CPU scan.

I/O Response Time

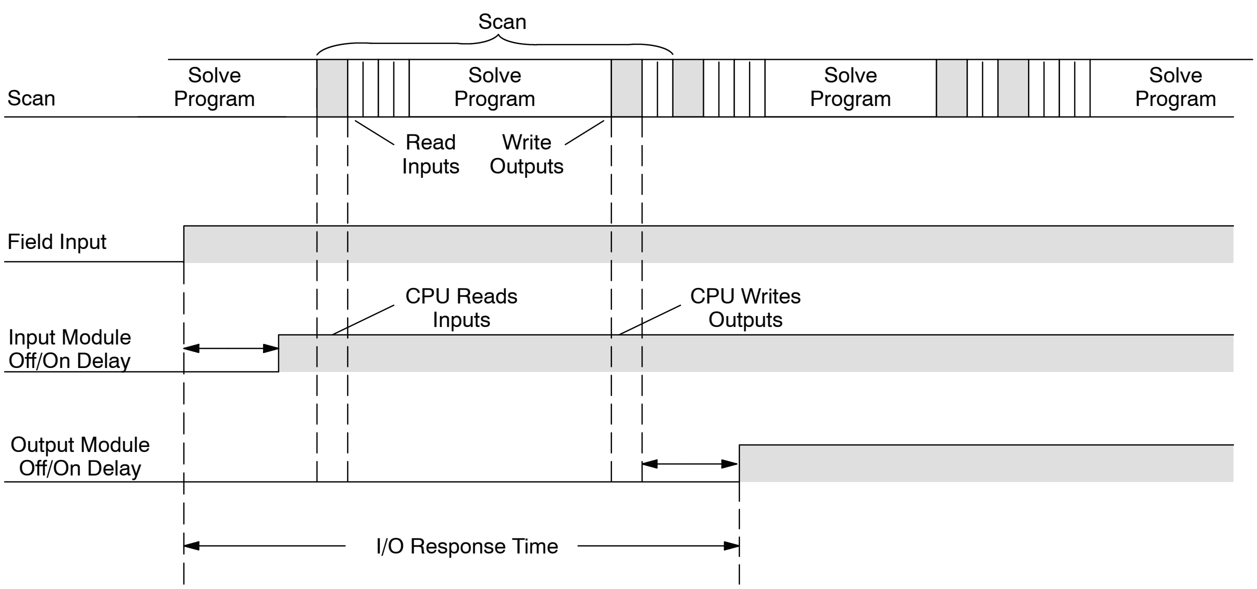
Is Timing Important for Your Application?

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 practically instantaneously. However, some applications do require extremely fast update times. There are four things that can affect the I/O response time:

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

Normal Minimum I/O Response

The I/O response time is shortest when the module senses the input change 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.

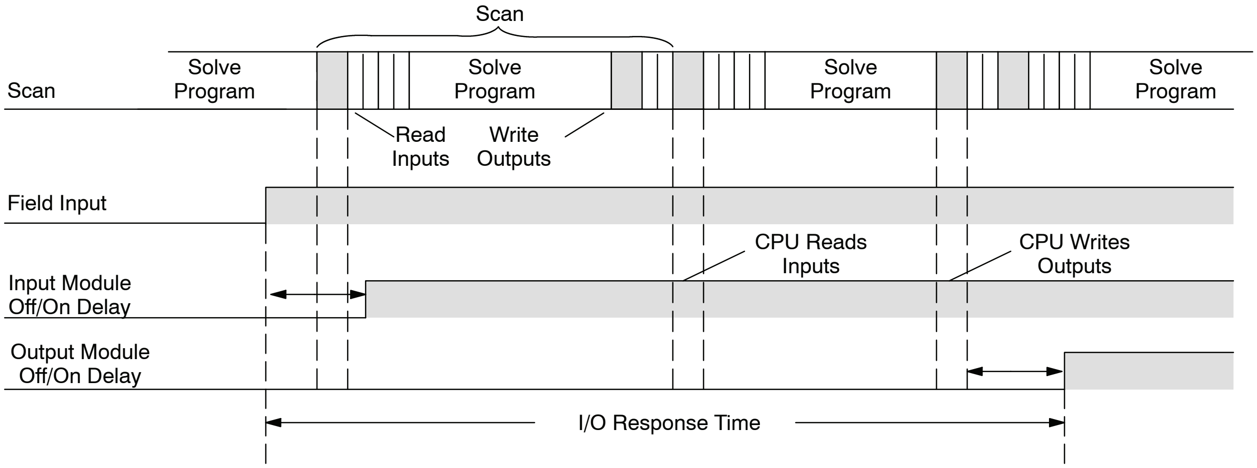
Input Delay + Scan Time + Output Delay = Response Time

Normal Maximum I/O Response

The I/O response time is longest when the module senses the input change after the Read Inputs portion of the execution cycle. In this case the new input status does not get 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

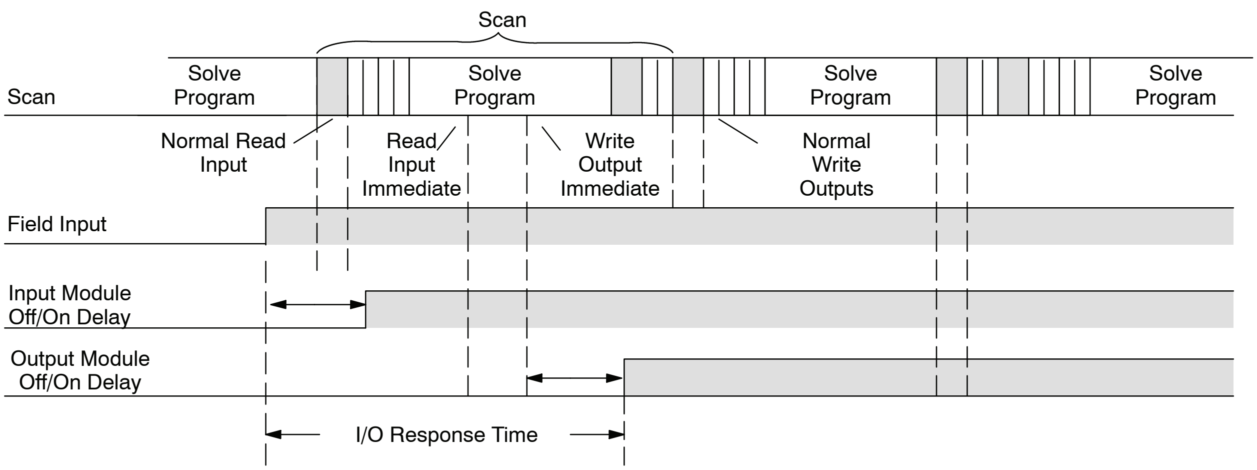


Improving Response Time

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

- Choose instructions with faster execution times
- Use immediate I/O instructions (which update the I/O points during the ladder program execution segment)
- Choose modules that have faster response times

Immediate I/O instructions are probably the most useful technique. The following example shows immediate input and output instructions, and their effect.



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 is calculated by adding the time for the immediate input instruction, the immediate output instruction, and all instructions in between.

NOTE: When the immediate instruction reads the current status from a module, it uses the results to solve that one instruction without updating the image register. Therefore, any regular instructions that follow will still use image register values. Any immediate instructions that follow will access the module again to update the status.

CPU Scan Time Considerations

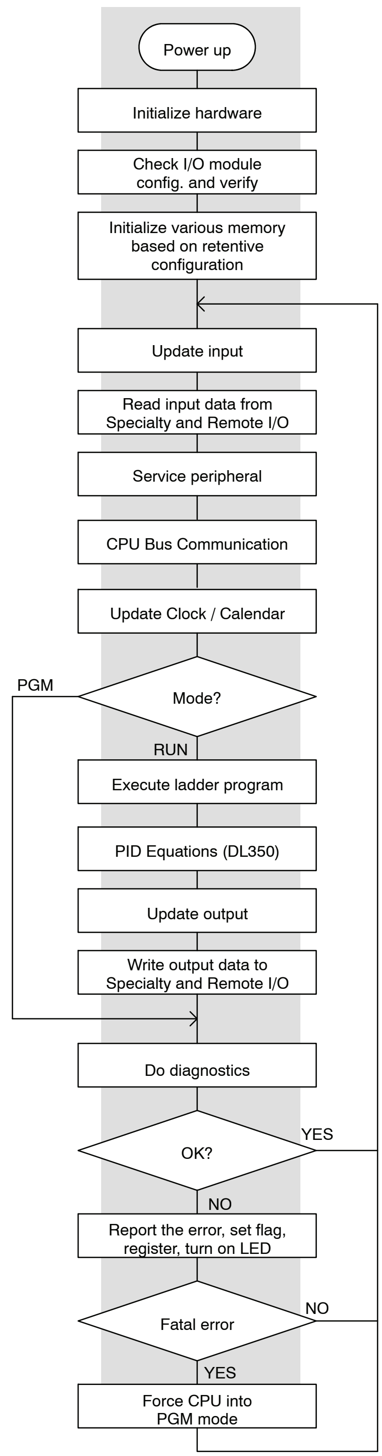
The scan time covers all the cyclical tasks that are performed by the operating system. You can use ***Direct***SOFT or the Handheld Programmer to display the minimum, maximum, and current scan times that have occurred since the previous Program Mode to Run Mode transition. This information can be very important when evaluating the performance of a system.

As 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 only one you really 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 modules and system configuration, such as expansion or remote I/O, can also affect the scan time. However, these things are usually dictated by the application.

For example, if you have a need to count pulses at high rates of speed, then you'll probably have to use a High-Speed Counter module. Also, if you have I/O points that need to be located several hundred feet from the CPU, then you need remote I/O because it's much faster and cheaper to install a single remote I/O cable than it is to run all those signal wires for each individual I/O point.

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



Intialization Process

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 Service Request	DL350
Minimum	1.2 µs
Maximum	1.5– µs

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)		DL350
Nothing Connected	Min. & Max.	0 µs
Port 1	Send Min. / Max.	6.8/12.6 µs
	Rec. Min. / Max.	9.2/972 ms
Port 2	Send Min. / Max.	6.8/12.6 µs
	Rec. Min. / Max.	9.2/972 ms

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.

NOTE: Some specialty modules can have a considerable impact on the CPU scan time. If timing is critical in your application, consult the module documentation for any information concerning the impact on the scan time.

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		DL350
Program Mode	Minimum	79.0 µs
	Maximum	79.0 µs
Run Mode	Minimum	79.0 µs
	Maximum	79.0 µs

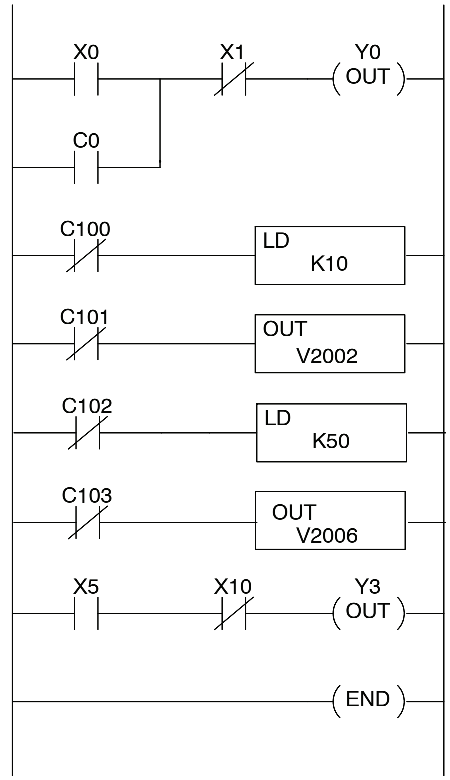
Diagnostics

The DL305 CPUs perform many types of system diagnostics. The amount of time required depends on many things, such as the number of I/O modules installed, etc. The following table shows the minimum and maximum times that can be expected.

Diagnostic Time	DL350
Minimum	104.0 µs
Maximum	139.6 µs

Application Program Execution

The CPU processes the program from the top (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.

You can add the execution times for all the instructions in your program to find the total program execution time.

For example, the execution time for a DL350 running the program shown would be calculated as follows.

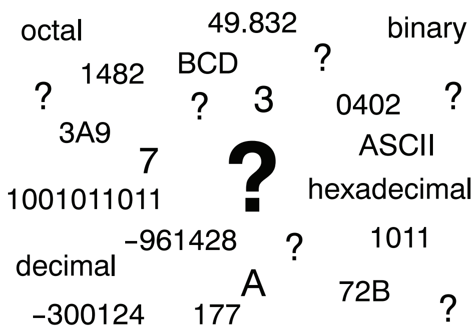
Instruction	Time
STR X0	1.4µs
OR C0	1.0µs
ANDN X1	1.2µs
OUT Y0	7.95µs
STRN C100	1.6µs
LD K10	62µs
STRN C101	1.6µs
OUT V2002	21.0µs
STRN C102	1.6µs
LD K50	62µs
STRN C103	1.6µs
OUT V2006	21.0µs
STR X5	1.4µs
ANDN X10	1.2µs
OUT Y3	7.95µs
END	16µs
TOTAL	210.5µs

Appendix C provides a complete list of instruction execution times for the DL350 CPU.

Program Control Instructions — the DL350 CPU offers 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 effect 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 will find that each PLC manufacturer has their own conventions on the use of numbers in their PLCs. Take a moment to familiarize yourself with how numbers are used in **AutomationDirect** PLCs. The information you learn here applies to all our PLCs!

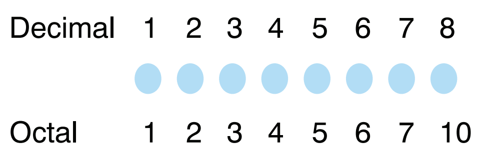


PLCs store and manipulate numbers in binary form: ones and zeros. So why do we have numbers in so many different forms? Numbers have meaning, and some *representations* are more convenient than others for particular purposes. Sometimes we use numbers to represent a size or amount of something. Other numbers refer to locations or addresses, or to time. In science we attach engineering units to numbers to give a particular meaning (see Appendix I for numbering system details).

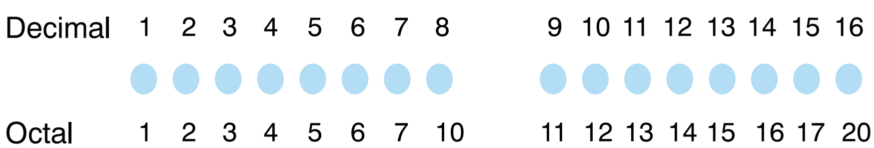
PLC Resources

PLCs offer a fixed amount of resources, depending on the model and configuration. The word "resources" includes 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).

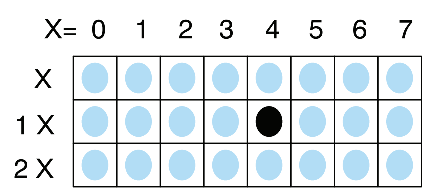


In the figure below, ther are two groups of eight circles. Counting in octal ther are "20" items, meaning 2 groups of eight, plus 0 individuals Avoid saying "twenty", say "two–zero octal". This makes a clear distinction between number systems.



After *counting* PLC resources, it's time to *access* PLC resources (there is 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.

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



V-Memory

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. The word “significant”, refers to the relative binary weighting of the bits.

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

V-memory data is 16-bit binary, but the data registers are rarely programmmed one bit at a time. Instructions or viewing tools work with binary, decimal, octal, and hexadecimal numbers. All of these are converted and stored as binary for us.

A frequently-asked question is “How do I tell if a number is binary, octal, BCD, or hex”? The answer is that we usually cannot tell 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 a storage box... that’s all. It does not convert or move the data on its own.

Binary-Coded Decimal Numbers

Since humans naturally count in decimal, we prefer to enter and view PLC data in decimal as well (via operator interfaces). 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.

BCD number	4				9				3				6			
	8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
V-memory storage	0	1	0	0	1	0	0	1	0	0	1	1	0	1	1	0

In a pure binary sense, a 16-bit word represents numbers from 0 to 65535. In storing BCD numbers, the range is reduced to 0 to 9999. Many math instructions use BCD data, and ***Direct***SOFT and the handheld programmer allow us to enter and view data in BCD. Special RLL instructions convert from BCD to binary, or visa-versa.

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 a convenient way for humans to view full binary data.

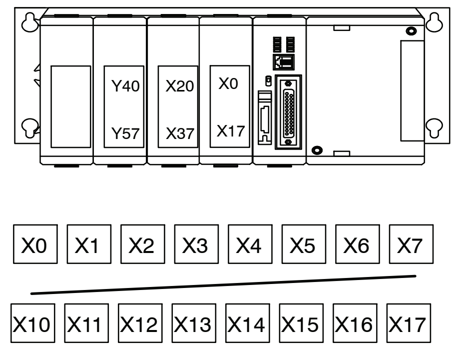
Hexadecimal number	A				7				F				4			
V-memory storage	1	0	1	0	0	1	1	1	1	1	1	1	0	1	0	0

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 the DL350 CPU. A memory map overview follows the memory descriptions.

Octal Numbering System

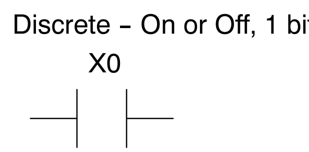
All memory locations or areas are numbered in Octal (base 8). For example, the diagram shows how the octal numbering system works for the discrete input points. Notice the octal system does not contain any numbers with the digits 8 or 9.

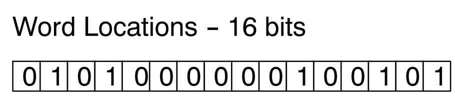


Discrete and Word Locations

As you examine the different memory types, you'll notice two types of memory in the DL350, 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.

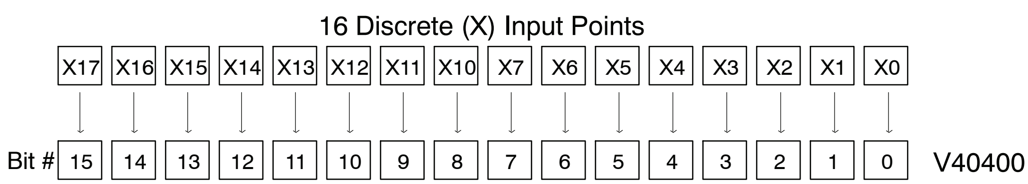
Some information is automatically stored in V-memory. For example, the timer current values are stored in V-memory.





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 the DL350 CPU in this chapter.

Input Points (X Data Type)

The discrete input points are noted by an X data type. There are up to 512 discrete input points available with the DL350 CPU. In this example, the output point Y0 will be turned on when input X0 energizes.

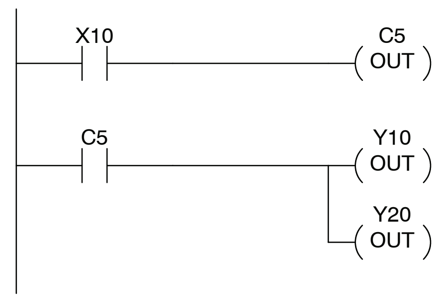
Output Points (Y Data Type)

The discrete output points are noted by a Y data type. There are up to 512 discrete output points available with the DL350 CPU. In this example, output point Y1 will turn 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. 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 has 16 consecutive discrete locations.

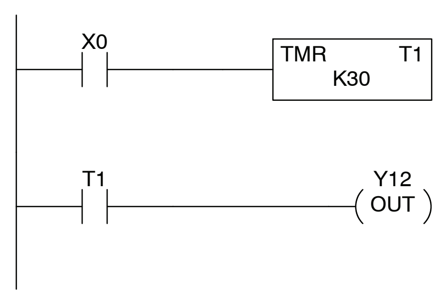
In this example, memory location C5 will energize when input X10 turns on. The second rung shows a simple example of how to use a control relay as an input.



Timers and Timer Status Bits (T Data type)

The amount of timers available depends on the model of CPU you are using. The tables at the end of this section provide the number of timers for the DL350. Regardless of the number of timers, you have access to timer status bits that 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.

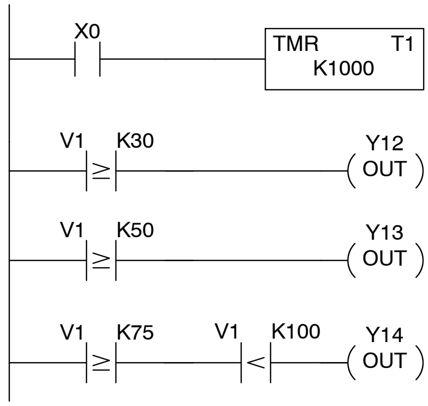
When input X0 turns on, timer T1 will start. When the timer reaches the preset of 3 seconds (K of 30) timer status contact T1 turns on. When T1 turns on, output Y12 turns on.



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.

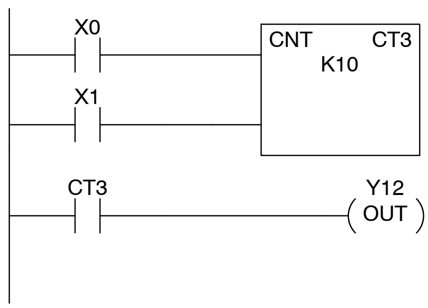
The primary reason for this is programming flexibility. The example shows how you can use relational contacts to monitor several time intervals from a single timer.



Counters and Counter Status Bits (CT Data type)

The amount of counters available depends on the model of CPU you are using. The tables at the end of this section provide the number of counters for the DL350. Regardless of the number of counters, you have access to 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.

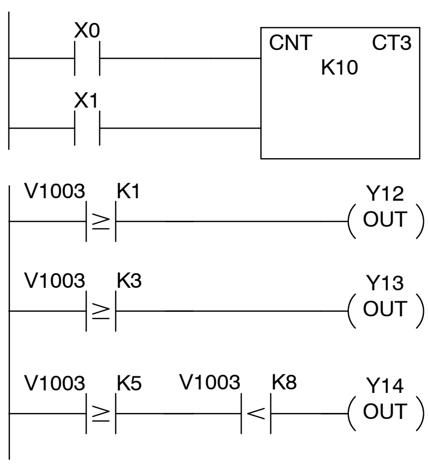
Each time contact X0 transitions from off to on, the counter increments by one. If X1 comes on, the counter is reset to zero. When the counter reaches the preset of 10 counts (K of 10) counter status contact CT3 turns on. When CT3 turns on, output Y12 turns on.



Counter Current Values (V Data Type)

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.

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

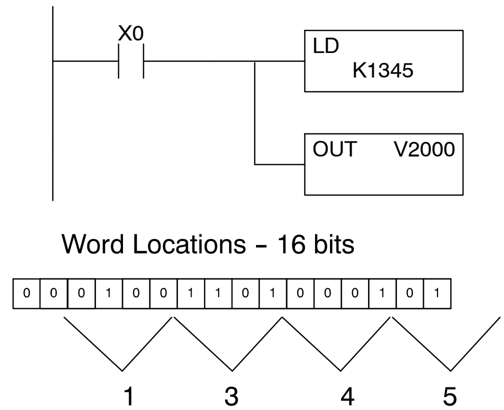


Word Memory (V Data Type)

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.

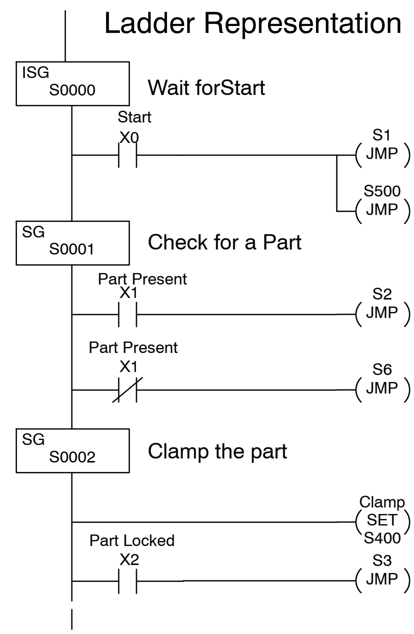
The example shows how a four-digit BCD constant is loaded into the accumulator and then stored in a V-memory location.



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

DL350 System V-memory

System V-memory	Description of Contents	Default Values / Ranges
V7620-V7627	Locations for DV-1000 operator interface parameters	
V7620	Sets the V-memory location that contains the value.	V0 - V3777
V7621	Sets the V-memory location that contains the message.	V0 - V3777
V7622	Sets the total number (1 - 16) of V-memory locations to be displayed.	1 - 16
V7623	Sets the V-memory location that contains the numbers to be displayed.	V0 - V3777
V7624	Sets the V-memory location that contains the character code to be displayed.	V0 - V3777
V7625	Contains the function number that can be assigned to each key.	V-memory for X, Y, or C
V7626	Reserved	0,1,2,3,12
V7627	Reserved	Default=0000
V7630-V7632	Reserved	-
V7633	User defined timer interrupt/operation of battery/Binary instruction sign flag* Bit 0-7 40H Setting Interrupt Bit 12 ON with battery sign flag. ON use sign flag - OFF no sign flag Bit 15 Binary instruction sign flag. ON use sign flag - OFF no sign flag	
V7634	User defined timer interrupt	
V7640	Loop Table Beginning address	V1400-V7340
V7641	Number of Loops Enabled	1-4
V7642	Error Code - V-memory Error Location for Loop Table	
V7643-V7647	Reserved	
V7650	Port 2 End-code setting Setting (A55A), Nonprocedure communications start.	
V7651	Port 2 Data format -Non-procedure communications format setting.	
V7652	Port 2 Format Type setting - Non-procedure communications type code setting.	
V7653	Port 2 Terminate-code setting - Non-procedure communications Termination code setting.	
V7654	Port 2 Store V-mem address - Non-procedure communication data store V-Memory address.	
V7655	Port 2 Setup area -0-7 Comm protocol (flag 0) 8-15 Comm time out/response delay time (flag 1)	
V7656	Port 2 setup area - 0-15 Communication (flag2, flag 3)	
V7657	Port 2 setup area - Bit to select use of parameter	
V7660-V7707	Set-up Information	
V7710-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	
V7730-V7737	For slot 0 to 7 D3-DCM	
V7747	Location contains a 10ms counter. This location increments once every 10ms.	
V7750	Reserved	

System V-memory	Description of Contents
V7751	Fault Message Error Code — stores the 4-digit code used with the FAULT instruction when the instruction is executed.
V7752	Reserved
V7753	Reserved
V7754	Reserved
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–V7764	Location for syntax error information.
V7765	Scan — 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 to 59).
V7767	Contains the number of minutes on the clock. (00 to 59).
V7770	Contains the number of hours on the clock. (00 to 23).
V7771	Contains the day of the week. (Mon, Tue, etc.).
V7772	Contains the day of the month (1st, 2nd, 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 Program Mode to Run Mode transition (milliseconds).
V7777	Scan — stores the maximum scan time that has occurred since the last Program Mode to Run Mode transition (milliseconds).

The following system control relays are valid only for D3–350 CPU remote I/O setup on Communications Port 2.

System CRs	Description of Contents
C740	Completion of setups – ladder logic must turn this relay on when it has finished writing to the Remote I/O setup table
C741	Erase received data – turning on this flag will erase the received data during a communication error.
C743	Re-start – Turning on this relay will resume after a communications hang-up on an error.
C750 to C757	Setup Error – The corresponding relay will be ON if the setup table contains an error (C750 = master, C751 = slave 1... C757=slave 7
C760 to C767	Communications Ready – The corresponding relay will be ON if the setup table data is valid (C760 = master, C761 = slave 1... C767=slave 7

DL350 Memory Map

Memory Type	Discrete Memory Reference (octal)	Word Memory Reference (octal)	Qty. 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
Timer Current Values	None	V0 – V377	256	V0 K100 ≥
Timer Status Bits	T0 – T377	V41100 – V41117	256	T0
Counter Current Values	None	V1000 – V1177	128	V1000 K100 ≥
Counter Status Bits	CT0 – CT177	V41140 – V41147	128	CT0
Data Words	none	V1400 – V7377 V10000–V17777	3072 4096	None specific, used with many instructions
Stages	S0 – S1777	V41000 – V41077	1024	SG S 001 S0
System parameters	None	V7400–V7777	256	System specific, used for various purposes

DL350 Aliases

An alias is an alternate way of referring to certain memory types, such as timer/counter current values, V-memory locations for I/O points, etc., which simplifies understanding the memory address. The use of the alias is optional, but some users may find the alias to be helpful when developing a program. The table below shows how the aliases can be used to reference memory locations.

Address Start	Alias Start	Example
V0	TA0	V0 is the timer accumulator value for timer 0, therefore, it's alias is TA0. TA1 is the alias for V1, etc..
V1000	CTA0	V1000 is the counter accumulator value for counter 0, therefore, it's alias is CTA0. CTA1 is the alias for V1001, etc.
V40000	VGX	V40000 is the word memory reference for discrete bits GX0 through GX17, therefore, it's alias is VGX0. V40001 is the word memory reference for discrete bits GX20 through GX 37, therefore, it's alias is VGX20.
V40200	VGY	V40200 is the word memory reference for discrete bits GY0 through GY17, therefore, it's alias is VGY0. V40201 is the word memory reference for discrete bits GY20 through GY 37, therefore, it's alias is VGY20.
V40400	VX0	V40400 is the word memory reference for discrete bits X0 through X17, therefore, it's alias is VX0. V40401 is the word memory reference for discrete bits X20 through X37, therefore, it's alias is VX20.
V40500	VY0	V40500 is the word memory reference for discrete bits Y0 through Y17, therefore, it's alias is VY0. V40501 is the word memory reference for discrete bits Y20 through Y37, therefore, it's alias is VY20.
V40600	VC0	V40600 is the word memory reference for discrete bits C0 through C17, therefore, it's alias is VC0. V40601 is the word memory reference for discrete bits C20 through C37, therefore, it's alias is VC20.
V41000	VS0	V41000 is the word memory reference for discrete bits S0 through S17, therefore, it's alias is VS0. V41001 is the word memory reference for discrete bits S20 through S37, therefore, it's alias is VS20.
V41100	VT0	V41100 is the word memory reference for discrete bits T0 through T17, therefore, it's alias is VT0. V41101 is the word memory reference for discrete bits T20 through T37, therefore, it's alias is VT20.
V41140	VCT0	V41140 is the word memory reference for discrete bits CT0 through CT17, therefore, it's alias is VCT0. V41141 is the word memory reference for discrete bits CT20 through CT37, therefore, it's alias is VCT20.
V41200	VSP0	V41200 is the word memory reference for discrete bits SP0 through SP17, therefore, it's alias is VSP0. V41201 is the word memory reference for discrete bits SP20 through SP37, therefore, it's alias is VSP20.

X Input / Y Output Bit Map

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

MSB					DL350 Input (X) and Output (Y) Points										LSB	X Input Address	Y Output Address
15	14	13	12	11	10	9	8	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

Control Relay Bit Map

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

MSB							DL350 Control Relays (C)								LSB	Address
15	14	13	12	11	10	9	8	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						Additional DL350 Control Relays (C)									LSB	Address
15	14	13	12	11	10	9	8	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

Stage™ Control / Status Bit Map

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

MSB						DL350 Stage (S) Control Bits									LSB	Address
15	14	13	12	11	10	9	8	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
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

MSB					DL350 Additional Stage (S) Control Bits (continued)										LSB	Address
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1017	1016	1015	1014	1013	1012	1011	1010	1007	1006	1005	1004	1003	1002	1001	1000	V41040
1037	1036	1035	1034	1033	1032	1031	1030	1027	1026	1025	1024	1023	1022	1021	1020	V41041
1057	1056	1055	1054	1053	1052	1051	1050	1047	1046	1045	1044	1043	1042	1041	1040	V41042
1077	1076	1075	1074	1073	1072	1071	1070	1067	1066	1065	1064	1063	1062	1061	1060	V41043
1117	1116	1115	1114	1113	1112	1111	1110	1107	1106	1105	1104	1103	1102	1101	1100	V41044
1137	1136	1135	1134	1133	1132	1131	1130	1127	1126	1125	1124	1123	1122	1121	1120	V41045
1157	1156	1155	1154	1153	1152	1151	1150	1147	1146	1145	1144	1143	1142	1141	1140	V41046
1177	1176	1175	1174	1173	1172	1171	1170	1167	1166	1165	1164	1163	1162	1161	1160	V41047
1217	1216	1215	1214	1213	1212	1211	1210	1207	1206	1205	1204	1203	1202	1201	1200	V41050
1237	1236	1235	1234	1233	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

Timer and Counter Status Bit Maps

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

MSB					DL350 Timer (T) and Counter (CT) Contacts										LSB	Timer Address	Counter Address
15	14	13	12	11	10	9	8	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	V41140
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V41101	V41141
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V41102	V41142
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V41103	V41143
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V41104	V41144
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V41105	V41145
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V41106	V41146
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V41107	V41147

This portion of the table shows additional Timer contacts available with the DL350.

MSB					DL350 Additional Timer (T) Contacts										LSB	Timer Address
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V41110
237	236	235	234	233	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