DL405

Handheld Programmer

Manual Number D4-HP-M

WARNING

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

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

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

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

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

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

Trademarks

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

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

Copyright 1998, PLC*Direct*™ Incorporated
All Rights Reserved

No part of this manual shall be copied, reproduced, or transmitted in any way without the prior, written consent of PLC***Direct***™ Incorporated. PLC***Direct***™ retains the exclusive rights to all information included in this document.

Manual Revisions

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

Title: DL405 Handheld Programmer
Manual Number: D4-HP-M

Issue	Date	Effective Pages	Description of Changes
Original	1/94	Cover/Copyright Contents Manual Revisions 1-1 — 1-9 2-1 — 2-18 3-1 — 3-18 4-1 — 4-12 5-3 — 5-14 6-1 — 6-28 A-1 — A-13	Original Issue
Rev. A	5/98	Entire Manual Manual Revisions	Downsized to spiral Rev. A

Table of Contents

Chapter 1: Getting Started

Chapter 2: System Setup

Chapter 3: Entering Programs

Chapter 4: Changing Programs

Chapter 5: Naming and Storing Programs

Chapter 6: System Monitoring and Troubleshooting

Appendix A: DL405 Memory Map

1

Getting Started

In This Chapter. . . .

- Introduction
- How can I use the Handheld?
- Physical Characteristics and Specifications
- Keypad Layout

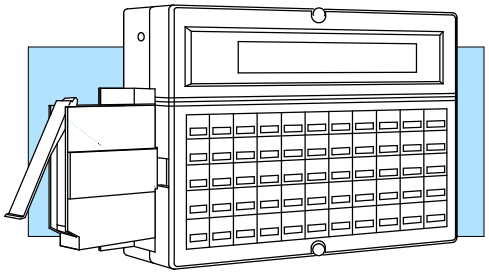
Introduction

DL405 Handheld Programmer

The DL405 Handheld Programmer is a general purpose programming tool for use with the DL405 family of automation products.

The Handheld is well suited for entering small programs or for troubleshooting machine operations. It is not the ideal choice for entering larger, more complex programs. For these types of programs, you should consider using ***Direct***SOFT, our PC-based programming software.

DL405 Handheld



Purpose of this manual

This manual will teach you the basic keystrokes used with the Handheld. It does not provide an example of every instruction. Once you understand the basic keystroke techniques, you should use the DL405 User Manual to determine the keystrokes required for the individual instructions.

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

Who should read this manual

If you understand the DL405 instruction set and system setup requirements, this manual will provide all the information you need to get a basic understanding of the Handheld. This manual *is not* intended to be a tutorial on the DL405 instruction set or system operation, but rather a user reference manual for the Handheld Programmer.

Supplemental Manuals

There is another manual that may occasionally be referenced by this manual. This manual is not absolutely necessary to use the Handheld, but it does provide additional details on several related subjects.

- DL405 User Manual (D4–USER–M)

Now, you have the material necessary to quickly understand the DL405 Handheld Programmer. So, let's get started!

Technical Assistance

After completely reading this manual, if you are not successful with implementing the OP-1500 or OP-1510, you may call PLC***Direct*** at (800) 633-0405, Monday through Friday from 9:00 A.M. to 6:00 P.M. Eastern Standard Time. Our technical support group will work with you in answering your application questions. If you have a comment or question about our products, services, or manuals which we provide, please fill out and return the suggestions card included with this manual.

Chapters The main contents of this manual are organized into the following six chapters.

1	**Getting Started**	provides an overview of the various uses for the Handheld Programmer and provides general specifications.
2	**System Setup**	shows you how to use the basic features of the Handheld Programmer. Also provides an overview of the various AUX functions and how they are used to setup the PLC system prior to entering programs.
3	**Entering Programs**	discusses all the operations used to enter a program.
4	**Changing Programs**	shows you how to quickly edit an existing program.
5	**Naming and Storing Programs**	shows you how to use program names and password protection. This chapter also shows you how to store programs on memory cartridges and cassette tapes.
6	**System Monitoring and Troubleshooting**	provides an overview of the various features used to monitor and troubleshoot the system.

Appendices Additional examples and reference information are in the following appendix:

A	**DL405 Memory Map**	provides a detailed listing of the DL405 memory map for I/O, timers, counters, etc.

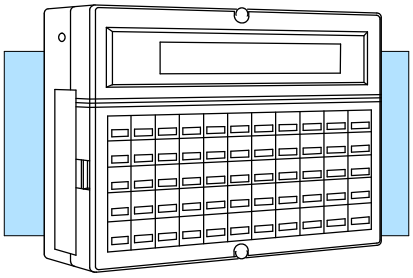
How can I use the Handheld?

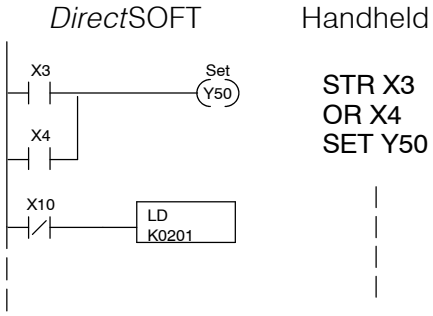
As a Programming Tool

The DL405 Handheld Programmer is ideally suited for entering or changing small programs with instruction mnemonics. You can enter programs up to the limits of the CPU you are using, but larger programs are much easier to design and enter with ***Direct***SOFT Programming Software.

In addition to entering programs, the Handheld is ideal for making on-site program or system changes. You can change almost any system setting, including I/O configuration, retentive memory settings, etc.

Since the Handheld has a built-in memory cartridge port, you can also move programs between memory cartridges and the CPU.

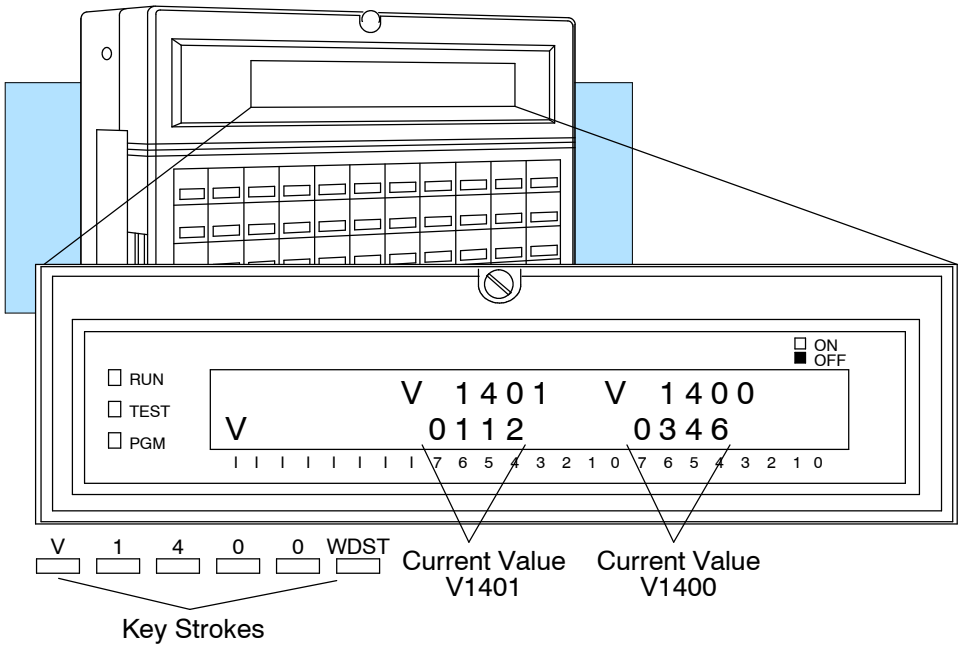




To Monitor Machine Operations

The Handheld is especially useful if you need to quickly look at the status of an I/O point, timer/counter value, or V-memory location.

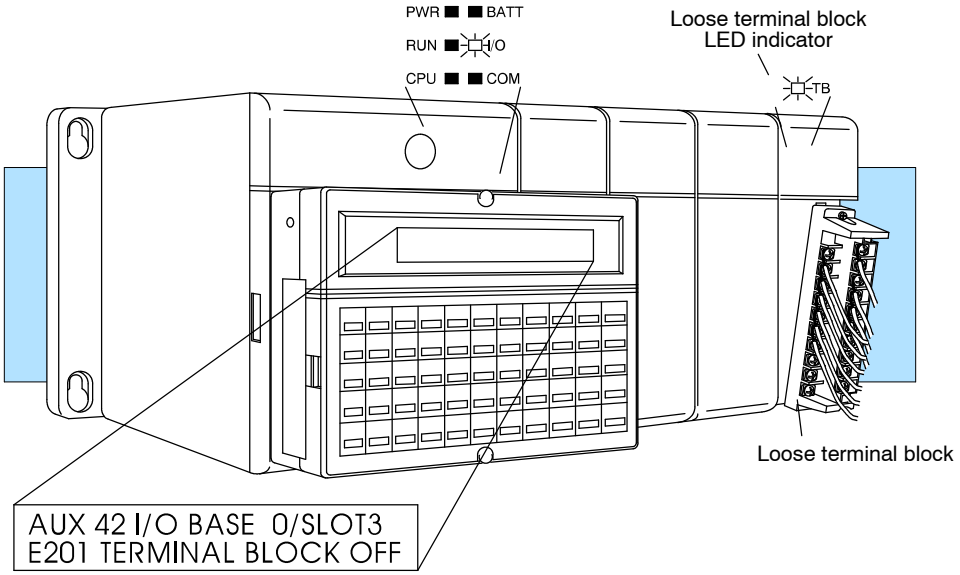
Since all DL405 data memory is mapped into V-memory, you only have to learn a few simple keystrokes to access virtually any type of system information.



As a Debugging Tool

Unfortunately, problems can occur with any automation system. The DL405 Handheld makes it easier to find problems and perform system maintenance operations in several areas. Auxiliary (AUX) Functions make these tasks easier.

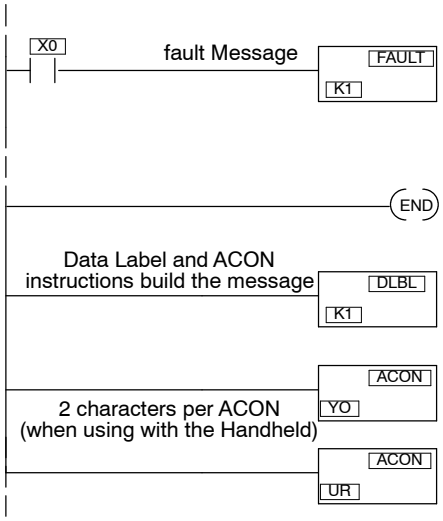
- Program — syntax check, duplicate reference check
- I/O — AUX 42 I/O Diagnostics shows the exact base and slot location for the problem. This System Auxiliary Function will also indicate the nature of the problem such as a loose connector, blown fuse, etc.
- Test Modes — you can also use the Handheld in several Test modes. Test Mode allows you to run a fixed number of scans, which can be very helpful in isolating machine problems.



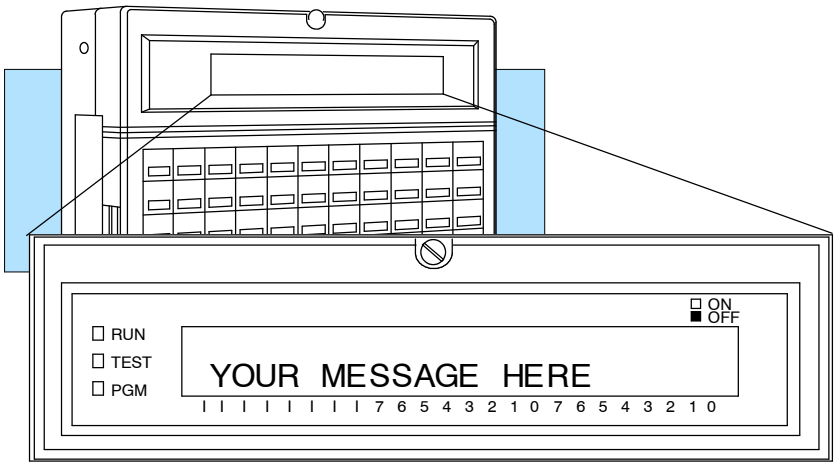
As a Low-Cost Message Display

If you're using a DL440 PLC, there are instructions that allow you to embed messages in the RLL program. These messages can easily be displayed on the Handheld and can be used to provide operator instructions, error messages, and even corrective actions.

Program Initiates Message



Handheld Displays Message



Physical Characteristics and Specifications

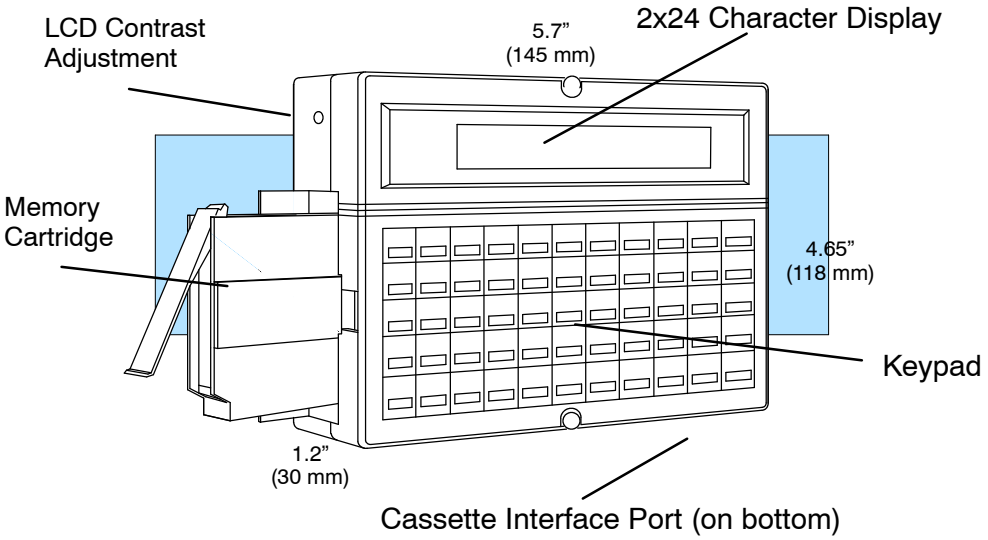
Handheld Layout

The Handheld was designed to be much more than a simple program entry tool and provides features not found on many handheld programmers. The 2x24 character backlit LCD display provides clear, easy-to-read characters and can be adjusted for brightness. (You can also turn off the backlighting.)

A memory cartridge slot is located on the side of the unit. You can use any of the DL405 memory cartridges to:

- copy CPU memory to a CMOS RAM, UVPROM, or EEPROM cartridge.
- compare the contents of CPU memory to data contained on a memory cartridge.
- copy data from the memory cartridge to the CPU.

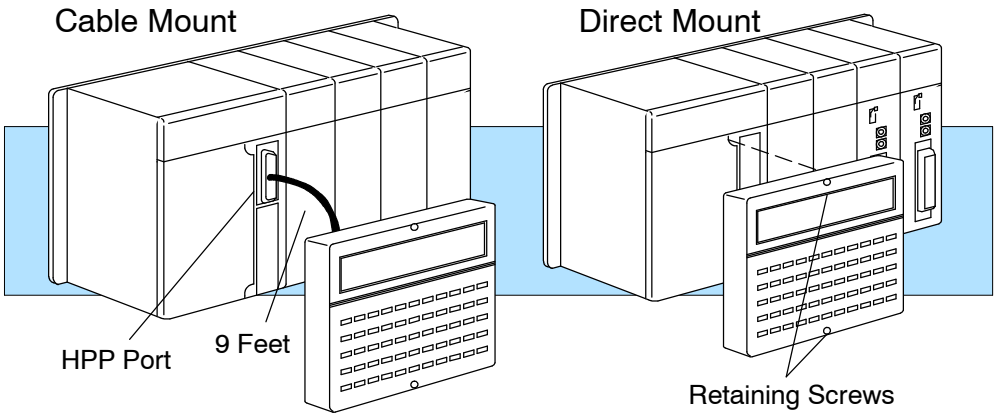
If you prefer, you can use common cassette tapes for program storage. (There's a cassette interface port located on the bottom of the unit.)



Connection Options

You can mount the Handheld directly to the CPU, or you can use a cable. The cable, part number D4-HPCBL-1, is approximately 9 feet (3m) in length and provides much more flexibility.

A cassette interface cable, part number D4-CASCBL, is required to connect a cassette recorder.



Specifications

The following table provides specifications for the DL405 Handheld Programmer.

Environmental	
Operating Temperature	32 to 140 F° (0 to 60 C°)
Storage Temperature	14 to 149 F° (–10 to 65 C°)
Humidity	20 to 90% (non-condensing)
Environmental Air	No corrosive gases
Vibration	MIL STD 810C 514.2
Shock Resistance	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3–304
Power	obtained through PLC port, 200 mA without backlight LCD 320 mA with backlight LCD
Dimensions	5.7" L x 4.6" H x 1.2" D 145mm W x 118mm H x 30mm D
Weight	13.4 oz. (380 g.)

CPUs Supported	**Programming Operations**
DL430, DL440	Read, Write, or erase programs
Simatic® TI425™, TI435™	Insert or delete an instruction
Texas Instruments® TI425™, TI435™	Search and replace instructions
	Locate a specific address
	Read, write, or clear Memory Cartridges
	Read or write to cassette tapes
Cables	**Machine Monitoring Operations**
D4–CASCBL, Cassette Interface	I/O status (up to 16 simultaneously)
D4–HPCBL–1, 1.5m Programmer Cable	On / Off status for contacts, coils, control relays, and bit locations
D4–HPCBL–2, 3.0m Programmer Cable	Timer and counter contacts, current values, and preset values
	Debugging Operations
	Forcing (one scan only)
	Run, Test, and Program Mode display
	Program syntax check
	Duplicate reference check
	Predefined error codes
	Message Display
	Up to 64, 23-character messages (must be in RLL program)

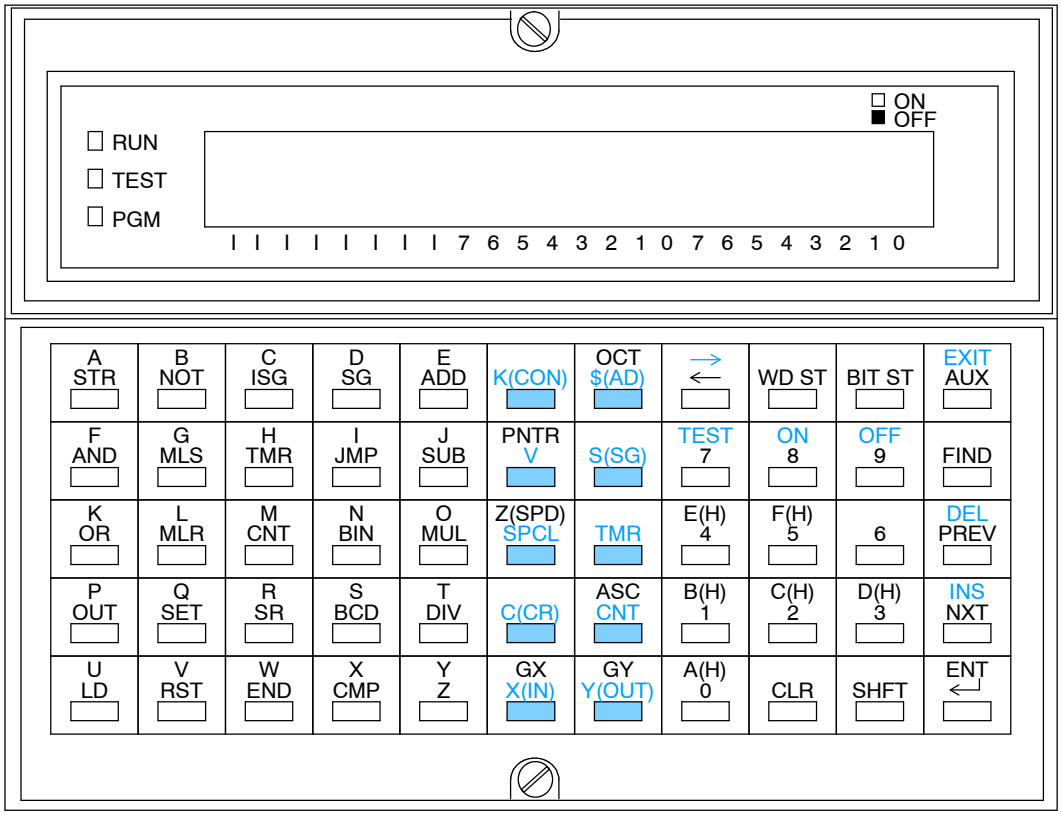
Keypad Layout

Four Groups of Keys

When you enter a program, you need to be able to select the instruction, enter any parameters for that instruction, and move to the next task. The Handheld keypad is organized into key groups that make this task easier. The groups are:

- Instruction keys — used to select the instruction
- Instruction identifier keys — used to assign a number to the instruction. For example each timer must have a unique identifier, TMR 0, TMR 1, etc.
- Numeric keys — used to enter values in various formats (BCD, decimal, octal, HEX, etc.)
- Editing / Monitoring keys — used to move through the program (Find, Delete, etc.)

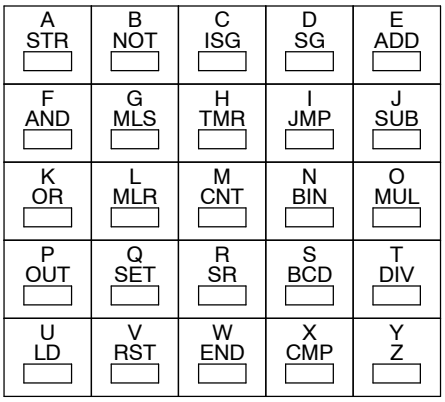
As you examine the keys, you'll notice some of the keys have more than one label. The top label describes the key when the Shift (SHFT) key is pressed. (These keys work just like the number keys on a computer keyboard.)



Instruction Keys

The instruction keys are used to select from the various instructions. As you examine the keypad you'll notice only the basic instructions have dedicated keys. The remaining instructions are entered by typing the instruction mnemonic with the alphabet keys.

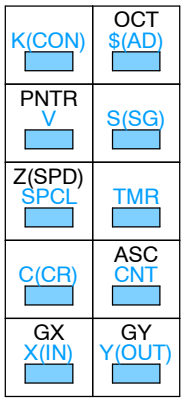
For example, to enter a OUTF instruction you would press the SHFT key followed by the O, U, T, and F keys.



Instruction Identifier Keys

The identifier keys are used to specify the exact instruction reference. For example, if you want to store a contact, you have to specify which contact you want to use.

Some DL405 instructions require you enter some of the instruction parameters in ASCII or octal. You can do this by using the SHFT key. Press SHFT followed by ASC to enter ASCII characters. Press SHFT followed by OCT and you can enter an octal number.

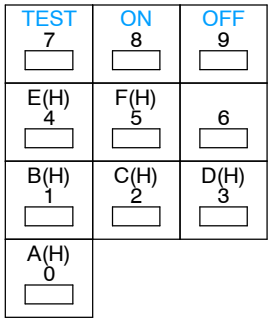


Numeric Keys

These keys are primarily used to enter numbers such as the instruction identifiers or constants. For hexadecimal numbers, you must use the SHFT key to access A - F.

The top three keys also have SHFT functions and are used for:

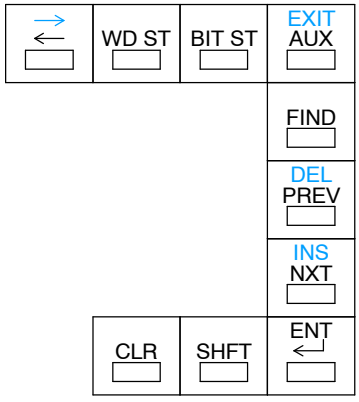
- TEST — initiates Test functions within Test Mode (more on this later)
- ON — forces an element to on
- OFF — forces an element to off



Editing / Monitoring Keys

These keys are primarily used to help you edit the program, monitor specific locations, or access system AUX functions.

You can use the AUX functions to perform various types of operations. Some of these include I/O Diagnostics, CPU mode control, Memory Cartridge operations, etc.



2

System Setup

In This Chapter. . . .

— Auxiliary Functions
— Handheld Setup
— CPU Setup
— I/O Configuration

Auxiliary Functions

What are Auxiliary Functions?

Many Handheld tasks involve the use of AUX Functions. The AUX Functions perform many different operations, ranging from simple operating mode changes to copying programs to memory cartridges. These functions are discussed in more detail throughout the manual. They are divided into categories that affect different system parameters. You'll use AUX Functions for the following types of operations.

AUX Function and Description		DL430	DL440
AUX 1* — Operating Mode			
11	Go to Run Mode	❍	❍
12	Go to Test Mode	❍	❍
13	Go to Program Mode	❍	❍
14	Run Time Edit	×	❍
AUX 2* — RLL Operations			
21	Check Program	❍	❍
22	Change Reference	×	❍
23	Clear Ladder Range	❍	❍
24	Clear Ladders	❍	❍
AUX 3* — V-Memory Operations			
31	Clear V Memory	❍	❍
32	Clear V Range	❍	❍
33	Find V-memory Value	×	❍
AUX 4* — I/O Configuration			
41	Show I/O Configuration	❍	❍
42	I/O Diagnostics	❍	❍
44	Powerup I/O Configuration Check	❍	❍
45	Select Configuration	❍	❍
46	Configure I/O	×	❍
47	Intelligent I/O	❍	❍

AUX Function and Description		DL430	DL440
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	❍	❍
5C	Display Error History	×	❍
AUX 6* — Handheld Programmer Configuration			
61	Show Revision Numbers	❍	❍
62	Beeper On / Off	HP	HP
63	Backlight On / Off	HP	HP
64	Select Online / Offline	HP	HP
65	Run Self Diagnostics	HP	HP
AUX 7* — Memory Cartridge Operations			
71	CPU to Memory Cartridge	❍	❍
72	Memory Cartridge to CPU	❍	❍
73	Compare Memory Cart. to CPU	❍	❍
74	Memory Cartridge Blank Check	HP	HP
75	Clear Memory Cartridge	HP	HP
76	Display Memory Cartridge Type	❍	❍
77	Tape to Memory Cartridge	HP	HP
78	Memory Cartridge to Tape	HP	HP
79	Compare Memory Cart. to Tape	HP	HP
AUX 8* — Password Operations			
81	Modify Password	×	❍
82	Unlock CPU	×	❍
83	Lock CPU	×	❍

❍ — supported

× — not supported

HP — Handheld Programmer function

Accessing the AUX Functions

Clear the display

CLR CLR

```
                    
                    
```

Select Auxiliary function

AUX

```
AUX FUNCTION SELECTION
AUX 1* OPERATING MODE
```

Use NXT or PREV to cycle through the menus

NXT

```
AUX FUNCTION SELECTION
AUX 2* RLL OPERATIONS
```

Press ENT to select sub-menus

ENT

```
AUX 2* RLL OPERATIONS
AUX 21 CHECK PROGRAM
```

You can also enter the exact number of the AUX Function to go straight to the sub-menu.

Enter the AUX number directly

AUX 2 1 ENT

```
AUX 2* RLL OPERATIONS
AUX 21 CHECK PROGRAM
```

Handheld Setup

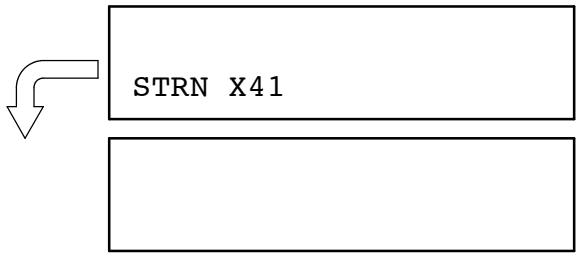
There are a few basic operations that you should be familiar with before you start using the Handheld. The next few pages provide an overview of the most basic Handheld features.

Clearing the Display

Sometimes we all make mistakes, so it's important to know how to clear the display and start from the beginning. The keystrokes needed depend on what you're trying to do, but one of two methods will always work. The following example shows two ways to clear the display.

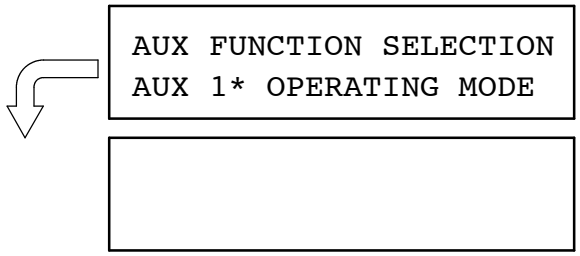
Use the CLR Key

CLR



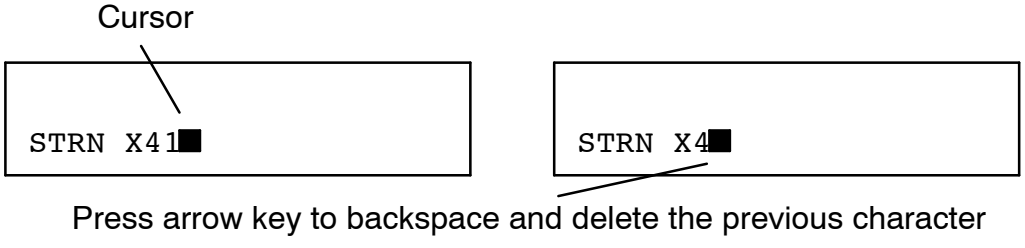
Use the EXIT Key to exit the AUX menus

SHFT EXIT



Using the Cursor

Once you start an operation, a flashing cursor appears. On some displays you can move this cursor left or right with the ← or → keys. If you move the cursor left, it acts just like the backspace key on a keyboard. Some menus also allow you to toggle between two choices by pressing the arrow keys.



Turning Off the Beeper

The Handheld has a beeper that provides confirmation of keystrokes. This can be quite annoying in an office environment. You can use Auxiliary (AUX) Function 62 to turn off the beeper.

Use the AUX menu

AUX

```
AUX FUNCTION SELECTION
AUX 1* OPERATING MODE
```

Enter 62 to select AUX 62

6 2 ↵

```
AUX 6* CFG MIU
AUX 62 BEEPER ON/OFF
```

Press ENT to turn off the beeper

ENT

Turning Off the Backlight

If necessary, you can turn off the display backlight. You can use Auxiliary (AUX) Function 63 to turn off the backlight.

Use the AUX menu

AUX

```
AUX FUNCTION SELECTION
AUX 1* OPERATING MODE
```

Enter 63 to select AUX 63

6 3 ↵

```
AUX 6* CFG MIU
AUX 63 BACKLIGHT ON/OFF
```

Press ENT to turn off the backlight

ENT

CPU Setup

A Few Things to Know

Even if you have years of experience using PLCs with handheld programmers, there are a few things you may need to know before you 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.

- Changing the CPU Modes
- Clearing the program (and other memory areas)
- How to initialize system memory
- Setting the CPU network address
- Setting retentive memory ranges
- Setting the Clock and Calendar

Changing the CPU Modes

There are three modes available with the DL405 CPUs.

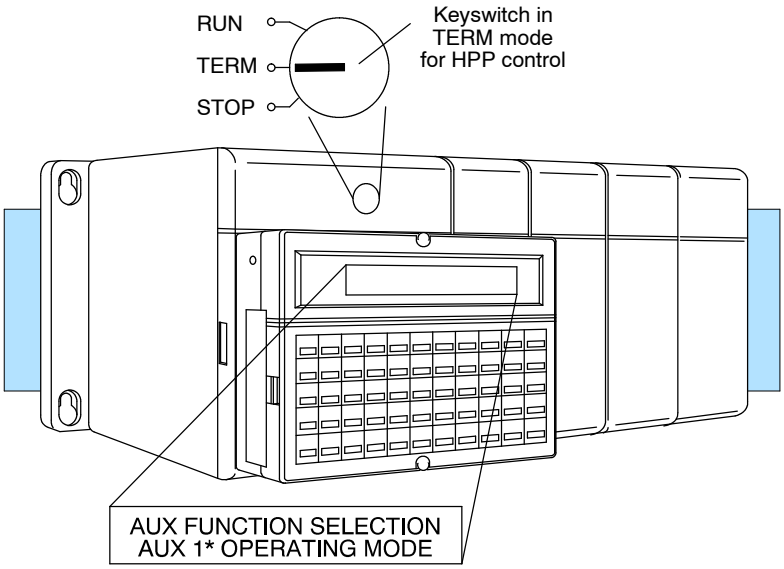
- RUN — executes program and updates I/O modules
- PGM — allows program entry, does not execute program or update I/O modules
- TEST — allows you to run a fixed number of scans and enables other TEST features. (See Chapter 6 for additional information.)

The DL405 User Manual provides additional information concerning the different modes of operation.

AUX 11, 12, and 13 are used to change the CPU operating mode. The CPU must be in PGM mode before you can enter a program. There are two ways to change to PGM mode.

1. Place the CPU keyswitch in the STOP position.
2. Place the CPU keyswitch in the TERM position and use the Handheld to change operating modes (AUX 13).

Here's an example that shows the keystrokes needed to change the CPU to Program mode.



Use the AUX menu

AUX

```
AUX FUNCTION SELECTION
AUX 1* OPERATING MODE
```

Enter 13 to select AUX 13

1 3 ↵

```
AUX 1* OPERATING MODE
AUX 13 GO TO PGM MODE
```

Press ENT to change to PGM mode

ENT

```
PGM MODE?
```

Press ENT to confirm the change

ENT

```
MODE = PGM
```

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.

Use AUX 24

AUX 2 4 ↵

```
AUX 2* RLL OPERATIONS
AUX 24 CLEAR LADDERS
```

Press ENT to clear the ladders

↵

```
CLR ALL LADDERS ?
```

Press ENT to confirm the operation

ENT ↵

```
CLR ALL LADDERS OK
```

You can also use other AUX functions to clear other memory areas.

- AUX 23 — Clear Ladder Range
- AUX 31 — Clear V Memory
- AUX 32 — Clear V Range

Initializing System Memory

The DL405 CPUs maintain 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 data values will be stored in scratchpad memory.

NOTE: You may never have to use this feature unless you have made changes that affect 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.

AUX 54 resets the system memory to the default values.

Use AUX 54

AUX 5 4 ↵

```
AUX 54 INIT SCRATCH PAD
CLR XPAD?
```

Press ENT to return to the default values

↵

```
OK
```

Setting the CPU Network Address

Since the DL405 CPUs have built-in ***Direct***NET ports (25-pin), you can use the Handheld to set the network address for the port and the port communication parameters. The default settings are:

- Station address 1
- Hex mode
- Odd parity

The ***Direct***NET manual provides additional information about communication settings required for network operation.

NOTE: You will only need to use this procedure if you have the bottom port connected to a network, operator interface or personal computer.

Use AUX 56 to set the network address and communication parameters.

Use AUX 56

AUX 5 6 ENT ENT

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

Enter the new station address

0 3 ENT

```
AUX 56 CPU N/W ADDRESS
HEX / ASCII
```

Use the arrow keys to toggle between the settings

← ENT

```
AUX 56 CPU N/W ADDRESS
NONE / ODD
```

Use the arrow keys to toggle between the settings

← ENT

```
AUX 56 CPU N/W ADDRESS
OK
```

Setting Retentive Memory Ranges

The DL405 CPUs 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:

- Control Relays — C600 - C737
- V Memory — V2000 - V7377
- Timers — None by default (you can make them retentive though)
- Counters — CT0 - CT177
- Stages — None by default (you can make them retentive though)

Use AUX 57 to change the retentive ranges. You cannot select an individual memory type to change. Instead, you must cycle through the retentive range for each memory type. If you do not want to change the starting or ending address for one of the memory types, just press ***ENT*** to leave the entry as is. If you make a mistake, you can press ***SHFT DEL*** to return the memory type currently displayed to the default settings.

Use AUX 57 to set the ranges

AUX 5 7 ENT ENT

(One of two types of displays will appear.)

```
AUX 57 SET RET RANGES
1st C0600
```

Display with existing range

```
AUX 57 SET RET RANGES
1st C----
```

Display without an existing range

Enter the first retentive CR address

6 3 0 ENT

(Except for V Memory, all ranges must be entered in 8-bit increments.)

```
AUX 57 SET RET RANGES
END C0737
```

Enter the last retentive CR address

6 5 0 ENT

```
AUX 57 SET RET RANGES
1st V02000
```

Enter the first retentive V-Memory address

3 5 0 0 ENT

```
AUX 57 SET RET RANGES
END V07777
```

```
END
```

Setting the Clock and Calendar

The DL440 CPU has a clock and calendar feature. If you are using this, you can use the Handheld and AUX 52 to set the time and date. The following format is used.

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

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).

Use AUX 52 to set the time and date

AUX 5 2 ENT ENT

```
AUX 52 CHG CLOCK / CAL
YMD 94/01/01/6(SAT)
```

Enter the new date

9 4 0 1 0 2 0
ENT

```
AUX 52 CHG CLOCK / CAL
YMD 94/01/02/0(SUN)
```

(You can also use the arrow keys to move the cursor over the exact part you need to change. Or, if you don't need to change the date you can just press ENTER without changing any numbers to leave the date as is and change the time.)

Press Enter to accept the new date and display the time

ENT

```
AUX 52 CHG CLOCK / CAL
TIME 22:08:17
```

Enter the new time

2 3 0 8 1 7 ENT

```
AUX 52 CHG CLOCK / CAL
TIME 23:08:17
```

(You can also use the arrow keys to move the cursor over the exact part you need to change. Or, if you don't need to change the time you can just press ENTER without changing any numbers to leave the time as is.)

Press Enter to accept the changes and display the new date and time

ENT

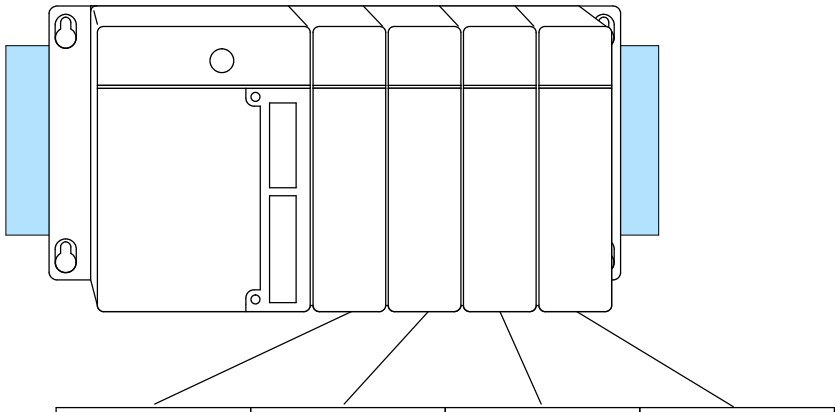
```
     94/01/02 23:08:17
```

I/O Configuration

Automatic Configuration

The DL405 CPUs automatically examine any installed I/O modules (including specialty modules) and establish the correct I/O configuration and addressing on power-up. For most applications, you never have to change or adjust the configuration.

The I/O addresses are assigned using octal numbering, starting at X0 and Y0. The addresses are assigned in groups of 8, 16, or 32, 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. See the DL405 User Manual for details. The following diagram shows the I/O numbering scheme for an example system.



Automatically compensates for I/O types

Automatic I/O Configuration Check

The DL405 CPUs can also be set to automatically check the I/O configuration on power-up. By selecting this feature you can quickly detect any changes that may have occurred while the power was disconnected. For example, if someone placed an output module in a slot that previously held an input module, the configuration check would detect the change and a message would appear on the Handheld. Use AUX 44 to enable the configuration check.

Use AUX 44

AUX 4 4 ↵ ↵

```
AUX 44 POWERUP CFG CHK
        (YES/NO)
```

Use the arrow key to select the option

↵ ↵

```
PWRUP CHK ON
```

If the system detects a change in the I/O configuration at power-up, an error code E252 NEW I/O CONFIGURATION will be generated. You can use AUX 42 to determine the exact base and slot location where the change occurred.

Initial Error Display

```
E252 NEW I/O CFG
```

Press CLR to clear the display

CLR

(The display suggests that you use AUX 42 to determine the error location.)

```
E2** DIAG ERROR AUX 42
```

Use AUX 42

CLR AUX 4 2 ENT ENT

```
AUX 42 I/O BASE0/SLOT1
E252 I/O CONFIG. ERROR
```

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

Even though an error was generated, you may actually want the new I/O configuration to be used. For example, you may have intentionally changed the module to use with a new program. You can use AUX 45 to select the new configuration, or, keep the existing configuration stored in memory.

Use AUX 45

AUX 4 5 ↵ ↵

```
AUX 45 SELECT CFG
        (NEW/MEM)
```

Use the arrow key to select the option

↵ ↵

```
CFG      NEW
```

New configuration selected

```
CFG      MEM
```

Existing configuration selected

WARNING: Make sure the I/O configuration being selected will work properly with the CPU program. You should always correct any I/O configuration errors before you place the CPU into RUN mode. Uncorrected errors can cause unpredictable machine operation that can result in a risk of personal injury or damage to equipment.

Manual Configuration

You will probably never need to use this feature, but the DL440 CPU allows you to manually assign I/O addresses for any or all I/O slots on the local or expansion bases. This feature is useful if you have a standard configuration that you must sometimes change slightly to accommodate special requests. For example, you may require two adjacent input modules to have addresses starting at X10 and X200 respectively.

In automatic configuration, the addresses were assigned on 8-point boundaries. Manual configuration assumes that all modules are at least 16 points, so you can only assign addresses that are a multiple of 20 (octal). For example, X30 and Y50 would not be valid addresses. This does not mean that you can only use 16 or 32 point modules with manual configuration. You can use 8 point modules, but 16 addresses will be assigned and 8 of them are unused.

Use AUX 46 to select Manual I/O Configuration.

Use AUX 46

AUX 4 6 ENT ENT

```
AUX 46 CFG I/O
    1->AUTO  2->MAN
```

Select Manual Configuration

2 ENT

```
AUX 46 CFG I/O
0/0  X   0  ------
```

base slot type starting address

Use PREV or NXT to scroll to the base and slot you want to change

NXT

OR

```
AUX 46 CFG I/O
0/1 ------ Y  0
```

Press CLR and enter the base and slot number

CLR 0 2 NXT

```
AUX 46 CFG I/O
0/2  X  20  ------
```

Enter the new starting address

X(IN) 1 0 0 ENT

(The display scrolls to the next slot and updates the addresses.)

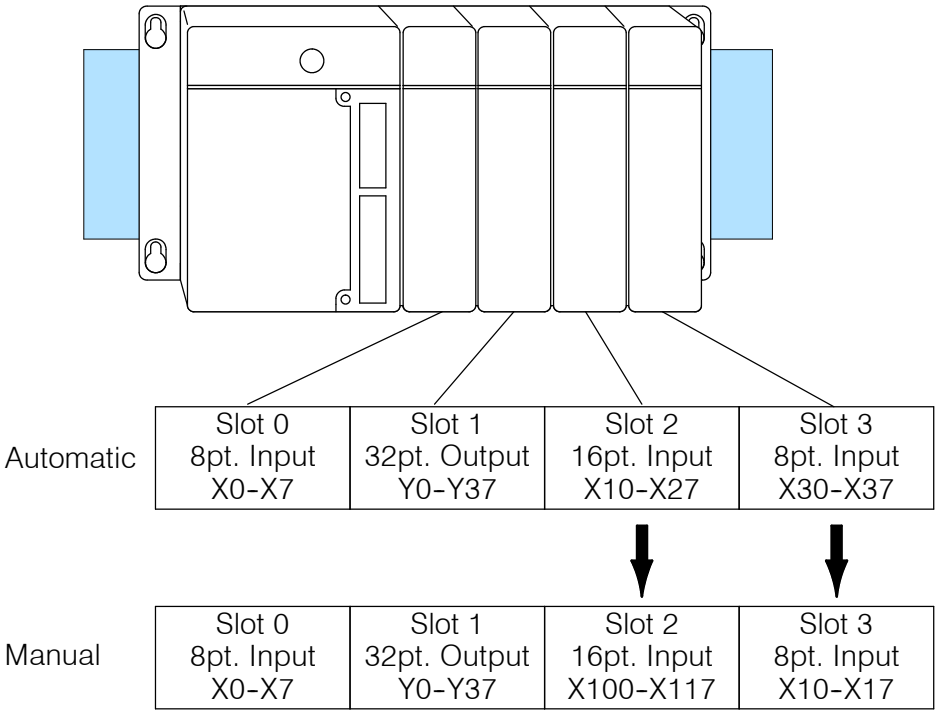
```
AUX 46 CFG I/O
0/3  X  20  ------
```

Exit the AUX function to save the change

SHFT EXIT

WARNING: If you manually configure an I/O slot, the I/O addressing for the other modules will change. This is because the DL405 products do not allow you to assign duplicate I/O addresses. You should always correct any I/O configuration errors before you place the CPU into RUN mode. Uncorrected errors can cause unpredictable machine operation that can result in a risk of personal injury or damage to equipment.

The following diagram shows how I/O addresses can be affected after a slot has been manually configured.



Removing a Manual Configuration

Once you have manually configured the addresses for an I/O slot, the system will automatically retain these values even after a power cycle. You can remove any manual configuration changes by simply performing an automatic configuration. AUX 46 executes an automatic configuration, which allows the CPU to examine the installed modules and determine the I/O configuration and addressing.

Use AUX 46

AUX 4 6 ↵ ↵

```
AUX 46 CFG I/O
    1->AUTO  2->MAN
```

Select Automatic Configuration

1 ↵

```
AUX 46 CFG I/O
    OK
```

Now that you understand the basics of the DL405 Handheld Programmer and how to perform many different types of system setup operations, you are ready to enter a program.

3

Entering Programs

In This Chapter. . . .

— Entering Simple Ladder Programs
— Checking for Program Errors

Entering Simple Ladder Programs

Purpose of the Examples

This section includes many examples that are intended to help you become familiar with the keystrokes required to enter the most basic DL405 instructions. Once you are familiar with the basic keystrokes, you should use the DL405 User Manual as a reference for the remaining instructions.

Handheld Key Sequences

The Handheld buffers all keystrokes until you press the ***ENT*** key. Then, it automatically checks the instruction to make sure it has been entered correctly. If the instruction was entered incorrectly an error message will be displayed. See Chapter 6 for a complete listing of error messages.

The Basics

There are a few basic instructions you must become familiar with to enter programs with the Handheld.

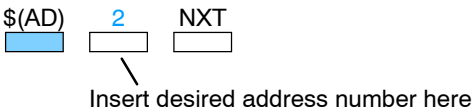
- STR – Stores a normally open element and indicates the beginning of a rung or network.
- AND – Joins one element (such as a contact) in series with another element or group of elements.
- AND STR – Joins a group of elements in series with another group of elements.
- OR – Joins one element in parallel with a previous element or group of elements.
- ORSTR – Joins parallel branches (each branch must begin with a STR instruction)
- Output – Each rung must have at least one output (Y, C, or box instruction)
- NOT – used with other instructions to utilize normally closed elements.
- All programs must contain an END statement.

Traversing the Program

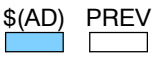
The instructions and associated data are located at program addresses (not the same as rung addresses used in ***Direct***SOFT). You may access an instruction by going directly to the instruction address or you may use the next and previous functions to toggle through the program addresses.

NOTE: Before using the first two below methods shown below ensure the display is clear, otherwise the entry will not be accepted.

Accessing Addresses



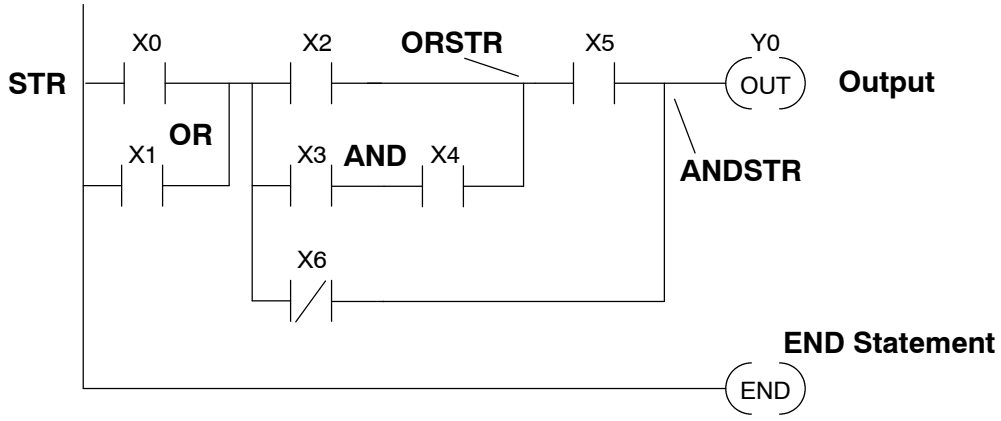
Address Previous / Next Functions



Previous / Next Keys

When using only the previous and next keys to toggle through the program addresses, it is not necessary to clear the display.

The following diagram shows a typical network and how each of these elements are used.

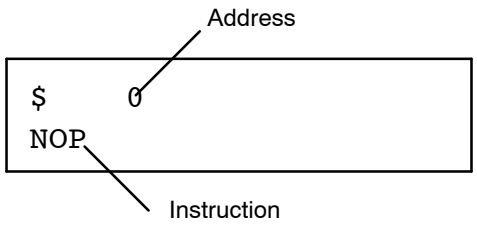


Starting at Address 0

If you're entering a complete program, you should always start at Address 0. The following example shows the keystrokes required. (The remaining examples will not show this display, but the keystrokes are required.)

Start at address 0

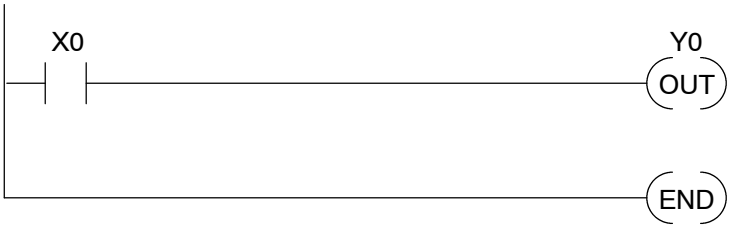
$(AD) NXT



Once you're at address 0, you can start entering a program.

Entering Simple Rungs

You use the STR instruction to start rungs that contain both contacts and coils. The following example shows how to enter a single contact and a single output coil.



Enter the contact

$(AD) NXT STR X(IN) 0 ENT

Starting at Address 0

```
$     0
STR X0
```

Enter the output coil

OUT Y(OUT) 0 ENT

```
$     1
OUT Y0
```

Enter the END statement

END ENT

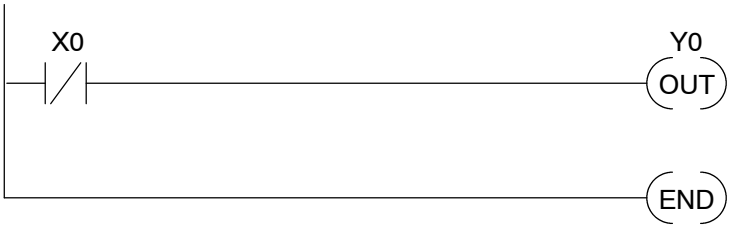
```
$     2
END
```

The example shows an X input contact and a Y output coil. If you examine the Handheld keyboard, you will notice specific keys for the other available data types.

C(CR) — Control Relay SPCL — Special Relay S(SG) — Stage

Entering Normally Closed Elements

Normally closed elements are entered with the STRN (Store Not) instruction. The following example shows a simple rung with a normally closed contact.



Enter the contact **Starting at Address 0**

$(AD) NXT STR NOT X(IN) 0 ENT

```
$     0
STRN X0
```

Enter the output coil

OUT Y(OUT) 0 ENT

```
$     1
OUT Y0
```

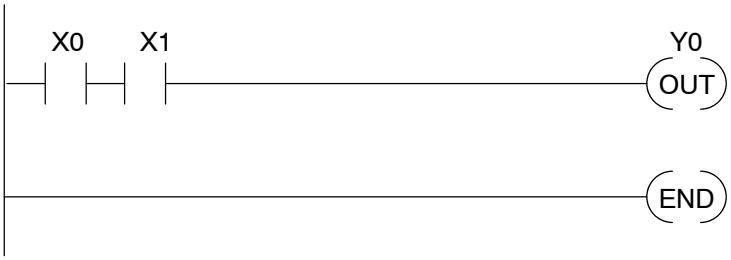
Enter the END statement

END ENT

```
$     2
END
```

Entering Series Elements

You must start the first rung with a STR instruction, since it contains more than one element and since it is also the beginning of the network. The AND instruction joins the series contacts. The following example shows how to enter two series contacts and a single output coil.



Enter the first contact

$(AD) NXT STR X(IN) 0 ENT

Starting at Address 0

```
$     0
STR X0
```

Enter the second contact

AND X(IN) 1 ENT

```
$     1
AND X1
```

Enter the output coil

OUT Y(OUT) 0 ENT

```
$     2
OUT Y0
```

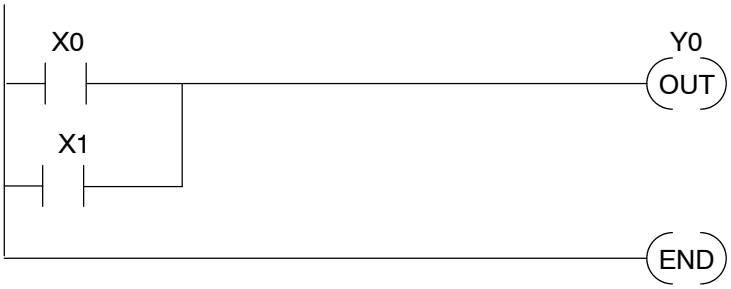
Enter the END statement

END ENT

```
$     3
END
```

Entering Parallel Elements

You must start the first rung with a STR instruction, since it contains more than one element and since it is also the beginning of the network. The OR instruction joins the parallel contacts. The following example shows how to enter two parallel contacts and a single output coil.



Enter the first contact

$(AD) NXT STR X(IN) 0 ENT

Starting at Address 0

```
$     0
STR X0
```

Enter the second contact

OR X(IN) 1 ENT

```
$     1
OR X1
```

Enter the output coil

OUT Y(OUT) 0 ENT

```
$     2
OUT Y0
```

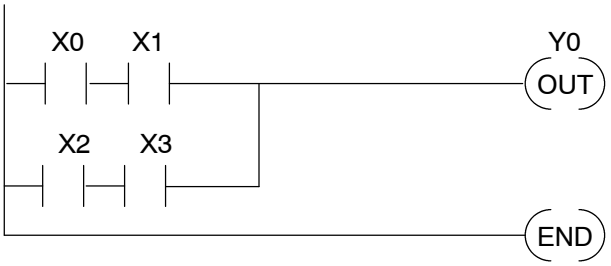
Enter the END statement

END ENT

```
$     3
END
```

Joining Series Branches in Parallel

Quite often it is necessary to joins one or more branches, of serial elements, in parallel. The OR STR instruction allows you to do this quite easily. The following example shows a simple network consisting of series elements joined in parallel.



Enter the first contact

$(AD) NXT STR X(IN) 0 ENT

Starting at Address 0

```
$     0
STR X0
```

Enter the second contact

AND X(IN) 1 ENT

```
$     1
AND X1
```

Start the next rung

STR X(IN) 2 ENT

```
$     2
STR X2
```

Add the next contact

AND X(IN) 3 ENT

```
$     3
AND X3
```

Join the branches

OR STR ENT

```
$     4
ORSTR
```

Enter the output coil

OUT Y(OUT) 0 ENT

```
$     5
OUT Y0
```

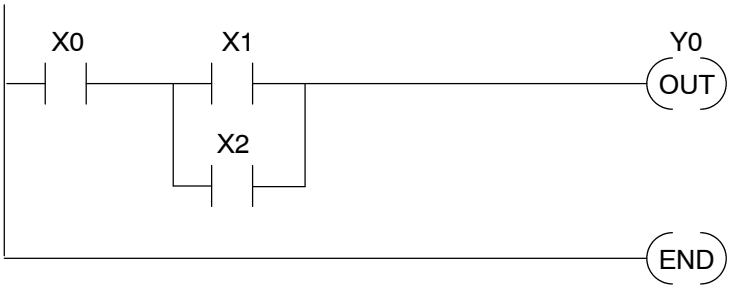
Enter the END statement

END ENT

```
$     6
END
```

Joining Parallel Branches in Series

The ANDSTR instruction joins one or more parallel branches in series. The following example shows a simple network with parallel and series branches.



Enter the first contact

$(AD) NXT STR X(IN) 0 ENT

Starting at Address 0

```
$     0
STR X0
```

Enter the second contact

STR X(IN) 1 ENT

```
$     1
AND X1
```

Enter the parallel contact

OR X(IN) 2 ENT

```
$     2
OR X2
```

Join the parallel branch

AND STR ENT

```
$     3
ANDSTR
```

Enter the output coil

OUT Y(OUT) 0 ENT

```
$     4
OUT Y0
```

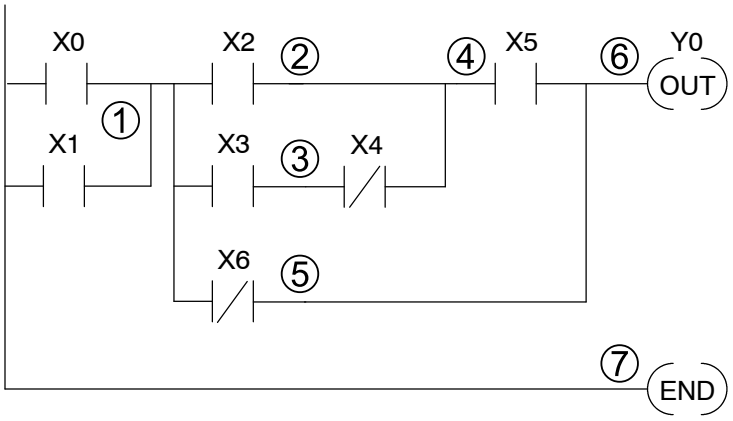
Enter the END statement

END ENT

```
$     5
END
```

Combination Networks

You can combine the various types of series and parallel branches to solve most any application problem. The following example shows a simple combination network.



①. Start the network

Starting at Address 0

$(AD) NXT STR X(IN) 0 ENT

```
$     0
STR X0
```

OR X(IN) 1 ENT

```
$     1
OR X1
```

②. Start branch 2

STR X(IN) 2 ENT

```
$     2
STR X2
```

③. Start branch 3, join with branch 2

STR X(IN) 3 ENT

```
$     3
STR X3
```

AND NOT X(IN) 4 ENT

```
$     4
ANDN X4
```

OR STR ENT

```
$     5
ORSTR
```

④. Add branch 4

AND X(IN) 5 ENT

```
$     6
AND X5
```

⑤. Add branch 5, join with branches 1-4

OR NOT X(IN) 6 ENT

```
$     7
ORN X6
```

AND STR ENT

```
$     8
ANDSTR
```

⑥. Add the output

OUT Y(OUT) 0 ENT

```
$     9
OUT Y0
```

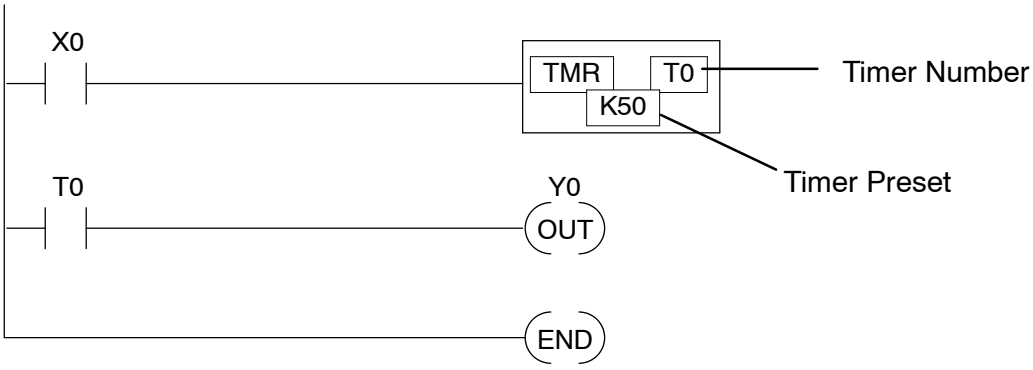
⑦. Enter the END statement

END ENT

```
$     10
END
```

Entering Timers and Counters

To enter a timer or counter, you also have to enter a preset value. This can be a constant, entered with the ***K(CON)*** key, or a V-memory location, entered with the ***V*** instruction reference key. This example shows how to enter these constants.



Enter the first contact — **Starting at Address 0**

$(AD) NXT STR X(IN) 0 ENT

```
$     0
STR X0
```

Enter the timer

with a constant

TMR TMR 0 K(CON) 5 0 ENT

```
$     1
TMR T0 K50
```

or

with a V-memory preset

TMR TMR 0

V 3 5 0 0 ENT

```
$     1
TMR T0 V3500
```

Enter the timer contact

STR TMR 0 ENT

```
$     3
STR T0
```

Enter the output

OUT Y(OUT) 0 ENT

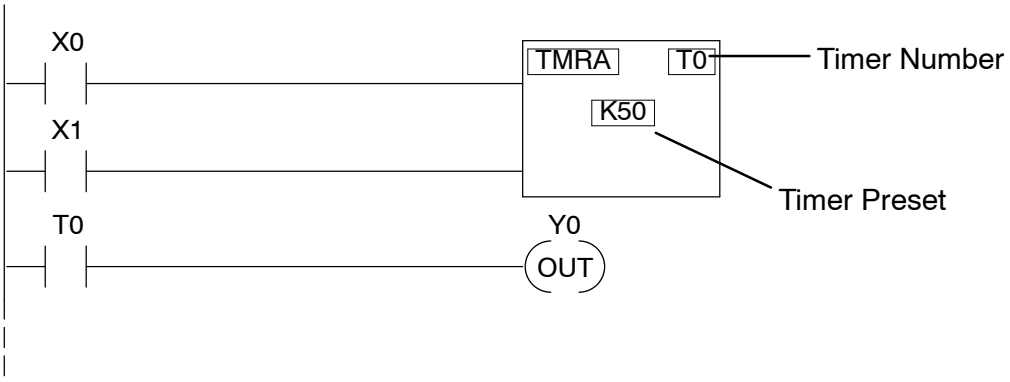
```
$     4
OUT Y0
```

End the program

END ENT

```
$     5
END
```

Some timers are accumulating timers and have reset lines. Also, two there are two types of counters that have multiple input lines. The following example shows how to use the Handheld to enter the additional input lines. Note that all input line contacts are entered before the actual instruction is entered. This is true for both timers and counters.



Enter the timer enable contact

$(AD) NXT STR X(IN) 0 ENT

Starting at Address 0

```
$     0
STR X0
```

Enter the timer reset contact

STR X(IN) 1 ENT

```
$     1
STR X1
```

Enter the timer

TMR SHFT A SHFT TMR 0

KCON 5 0 ENT

```
$     2
TMRA T0 K50
```

Enter the timer contact

STR TMR 0 ENT

```
$     5
STR T0
```

Enter the output

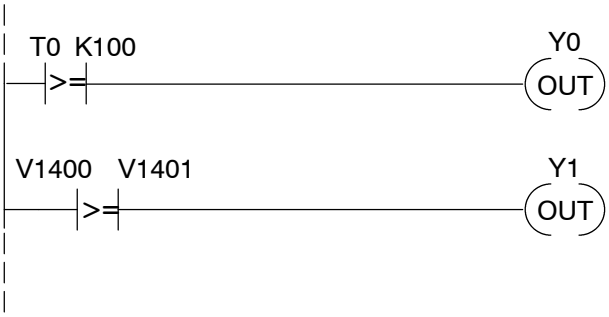
OUT Y(OUT) 0 ENT

```
$     6
OUT Y0
```

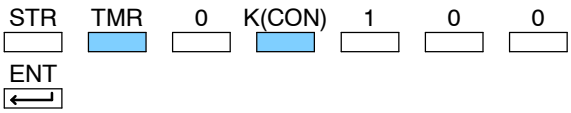
Entering Relational Contacts

Relational contacts allow you to quickly and easily compare various types of information. For example, you may want to compare the current value of a timer with a constant or a value contained in a V-memory location. Or, you could quickly compare two V-memory locations. There are several types of comparisons that can be made, less than, greater than, etc. See the DL405 User Manual for details on relational contacts.

The following example shows how to enter a relational contact.



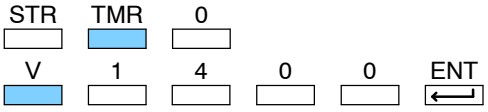
Timer contact with a constant



Address 10 shown as example

```
$     10
STR T0 K100
```

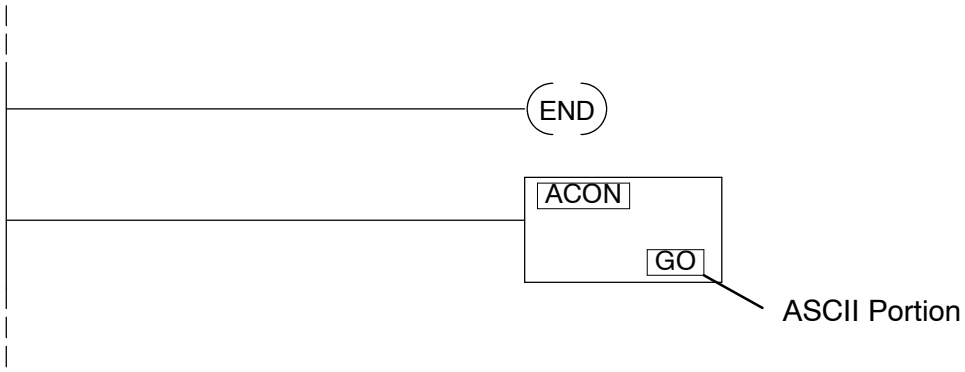
Timer contact compared to a V-memory location



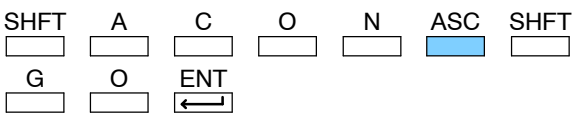

```
$     10
STR T0 V1400
```

Entering ASCII Characters

Some DL405 instructions, like the ACON instruction in the DL440, allow you to enter ASCII characters as part of the instruction. (An overview of the ACON instruction is provided in Chapter 6. Also, the DL405 User Manual provides detailed information.) Here is a simple example that shows how to enter the ASCII portion of the instruction with the Handheld Programmer.



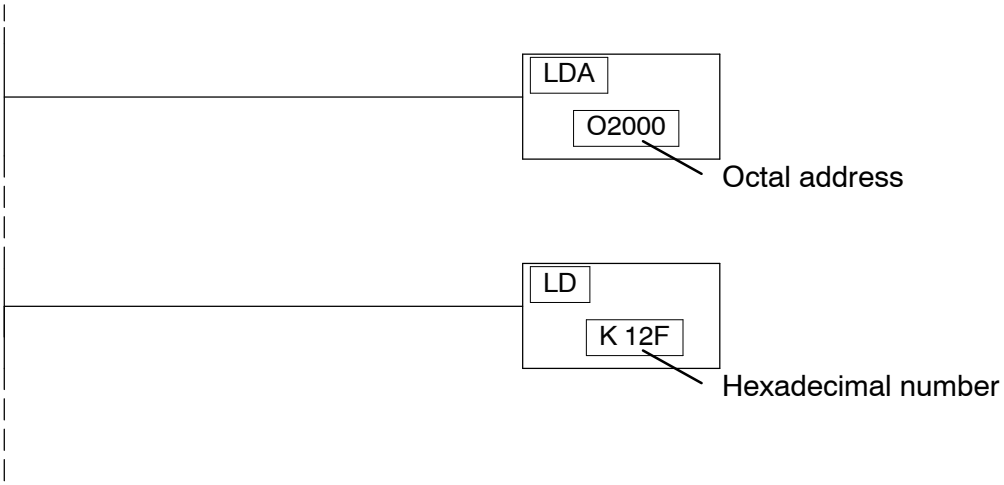
ACON ASCII example



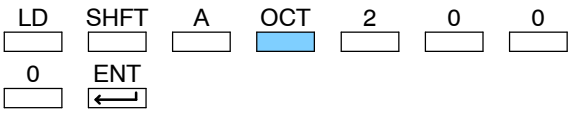

```
$       xx     (at address xx)
ACON    AGO
```

Entering Octal and Hex Numbers

Some DL405 instructions require different number formats as part of the instruction. For example, the LDA (Load Address) instruction requires an octal number for the address reference. Also, you may want to load a hexadecimal value into the accumulator. The following example shows you how to enter octal and hex numbers with the Handheld Programmer. (See the DL405 User Manual for details on the actual instructions.)

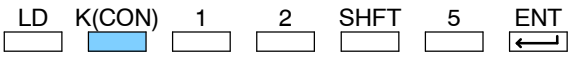


LDA Octal example




```
$      xx    (at address xx)
LDA O2000
```

LD Hexadecimal example



(SHFT 5 is hex F)

```
$      xx    (at address xx)
LD K12F
```

Checking for Program Errors

Error Checking

The Handheld automatically checks for errors during program entry. However, there may be occasions when you want to check a program that is already in the CPU. There are two types of checks available.

- Syntax
- Duplicate References

Syntax Check

You can use AUX 21, CHECK PROGRAM to check the program syntax. This check will find a wide variety of programming errors. The following example shows how to access AUX 21.

Use AUX 21 to perform syntax check

AUX 2 1 ENT ENT

```
AUX 21 CHECK PROGRAM
1:SYN 2:DUP REF
```

Select syntax check

1 ENT (This may take a minute or so.)

```
BUSY
```

One of two displays will appear

Error Display (example)

```
$ 8 E401 MISSING END
TMRA   T 002   K00050
```

(shows location in question)

Syntax OK display

```
NO SYNTAX ERROR
?
```

If you get an error, see the Error Codes Section for a complete listing of programming error codes. Correct the problem and continue running the Syntax check until the NO SYNTAX ERROR message appears.

Duplicate Reference Check

You can use AUX 21, CHECK PROGRAM to check for multiple uses of the same output coil. The following example shows how to access AUX 21.

Use AUX 21 to perform syntax check

AUX 2 1 ENT ENT

```
AUX 21 CHECK PROGRAM
1:SYN 2:DUP REF
```

Select Duplicate Reference check

2 ENT (This may take a minute or so.)

```
BUSY
```

One of two displays will appear

Error Display (example)

```
$ 12 E471 DUP COIL REF
OUT    Y 0000
```

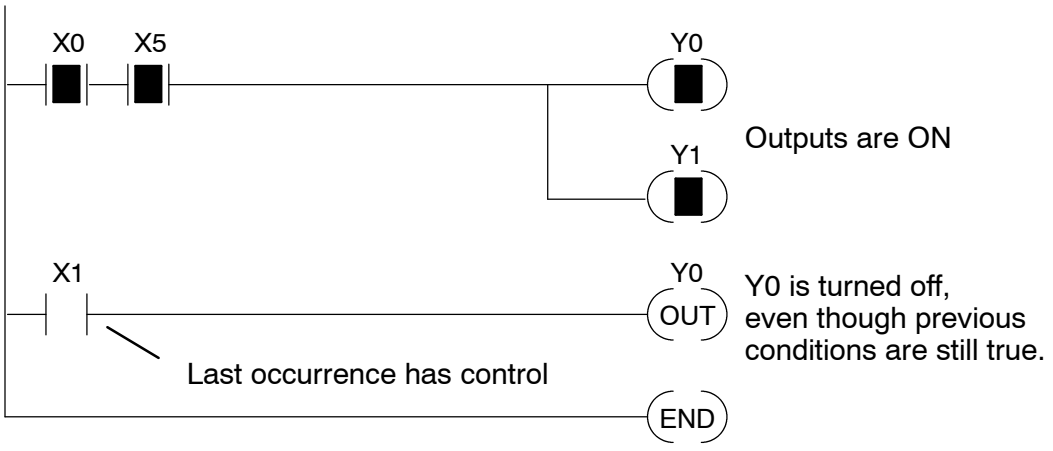
(shows location in question)

Syntax OK display

```
NO DUP REFS
?
```

If you get a Duplicate Reference error, see Error Codes Section for a complete listing of programming error codes. Correct the problem and continue running the Duplicate Reference check until the NO DUP REFS message appears.

NOTE: You can use the same coil in more than one location. However, the last occurrence of the element will take priority. Consider the following example.



4

Changing Programs

In This Chapter. . . .

- Two Ways to Edit a Program
- Displaying a Program
- Finding a Specific Instruction
- Changing an Instruction
- Inserting an Instruction
- Deleting an Instruction
- Using Search and Replace
- Editing Programs During Run Mode

Two Ways to Edit a Program

Program Mode

You should select Program Mode for the majority of program changes. In this mode the CPU does not execute the application program so you do not have to worry about how the program changes can affect the machine operation. Obviously you can use Program Mode with either the DL430 or DL440 CPUs. The majority of this chapter shows you how to change various aspects of the program during Program Mode.

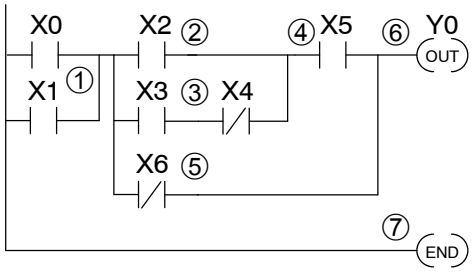
Run Mode Edits

With the DL440 CPU you also have the capability to edit the program during Run Mode. This is especially helpful if you only have to make minor adjustments such as adding or changing a single input contact or output coil. There is a separate section of this chapter that discusses using the Handheld for editing programs during Run Mode.

Displaying a Program

Since the Handheld displays the mnemonic instructions, you can step through the individual program instructions. If the CPU is in the RUN or TEST-RUN mode, the upper right corner displays the status for bit instructions (X, Y, C, S, T, CT, SP, or GX).

Ladder Representation



Mnemonic Listing and Addresses

ADDRESS	INSTRUCTION	DESCRIPTION
0	STR X0	Starts branch 1 with X0
1	OR X1	Joins X1 in parallel with X0
2	STR X2	Starts branch 2 with X2
3	STR X3	Starts branch 3 with X3
4	ANDN X4	Joins X4 (NOT) with X3
5	ORSTR	Joins branches 2 and 3
6	AND X5	Starts branch 4 with X5
7	ORN X6	Joins X6 (NOT) in parallel with X5
8	ANDSTR	Joins branches 4 and 5 with 1–3
9	OUT Y0	Stores the output and finishes the network
10	END	Ends the program

Use NXT or PREV to scroll through the addresses

$(AD) NXT

```
$     0
STR     X 0000
```

Black indicates ON

NXT

```
$     2                S
OR      X 0001
```

S indicates OFF

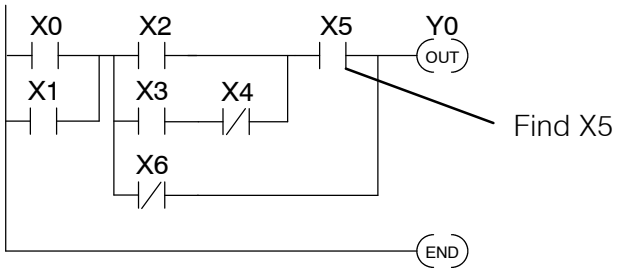
PREV

```
$     1
STR     X 0000
```

Finding a Specific Instruction

If you do not want to scroll through the program, you can use the FIND feature to automatically search for an instruction. The following example shows the instructions, addresses, and corresponding Handheld displays for a small program.

Ladder Representation



Identify the Instruction

ADDRESS	INSTRUCTION	DESCRIPTION
0	STR X0	Starts branch 1 with X0
1	OR X1	Joins X1 in parallel with X0
—	—	—
—	—	—
6	AND X5	Starts branch 4 with X5
—	—	—
—	—	—
10	END	Ends the program

FIND the Address

AND X(IN) 5 FIND

```
          SEARCHING
AND X5
```

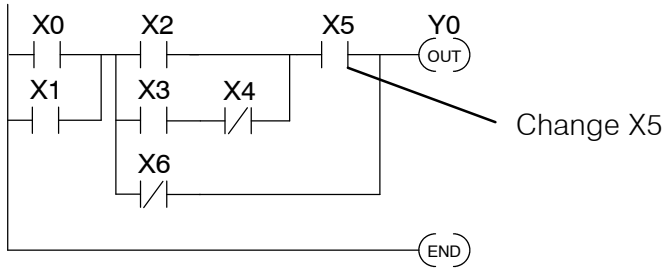
```
$    6
AND    X 0005
```

You can continue searching for other instances of the instruction just by pressing the ***FIND*** key again. If the instruction is not found, the error message “E602 INSTRUCTION MISSING” is displayed. (If you think the message is incorrect, check your keystrokes to make sure you’re entering the instruction correctly and try the operation again.)

Changing an Instruction

Once you've found the instruction you can change it very easily. The following example shows you how to change the X5 contact to X10.

Ladder Representation



Identify the Instruction

ADDRESS	INSTRUCTION	DESCRIPTION
0	STR X0	Starts branch 1 with X0
1	OR X1	Joins X1 in parallel with X0
—	—	—
—	—	—
6	AND X5	Starts branch 4 with X5
—	—	—
—	—	—
10	END	Ends the program

FIND the Address

AND X(IN) 5 FIND

```
          SEARCHING
AND X5
```

```
$     6
AND     X 0005
```

Change the Instruction

AND X(IN) 10

```
$     6
AND X10
```

ENT

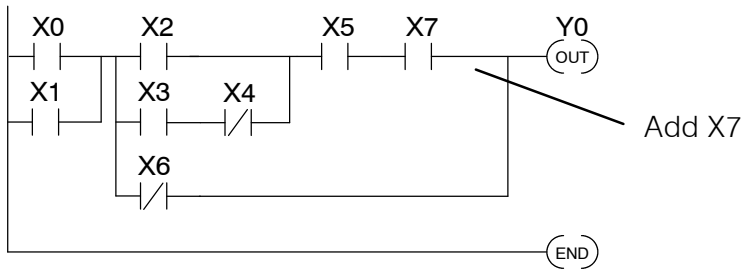
(New instruction is entered and display moves to next address.)

```
$     7
ORN     X 0006
```

Inserting an Instruction

Use the INSERT feature to add an instruction to the program. INSERT adds an instruction *before* the instruction that is being displayed, so make sure you are at the correct program address. Once you've inserted the new instruction, the remaining addresses increment. The following example shows the instructions, addresses, and corresponding Handheld displays for a small program.

Ladder Representation



Identify the Instruction

ADDRESS	INSTRUCTION	DESCRIPTION
0	STR X0	Starts branch 1 with X0
1	OR X1	Joins X1 in parallel with X0
—	—	—
6	AND X5	Starts branch 4 with X5
	AND X7	Adds X7 in series with X5
7	ORN X6	Joins X6 (NOT) in parallel
—	—	—
10	END	Ends the program

Insert before

FIND the Address

OR NOT X(IN) 6 FIND

```
            SEARCHING
ORN X6
```

```
$     7
ORN     X 0006
```

Insert the New Instruction

AND X(IN) 7 SHFT INS

```
$     7    INSERT INST?
AND X7
```

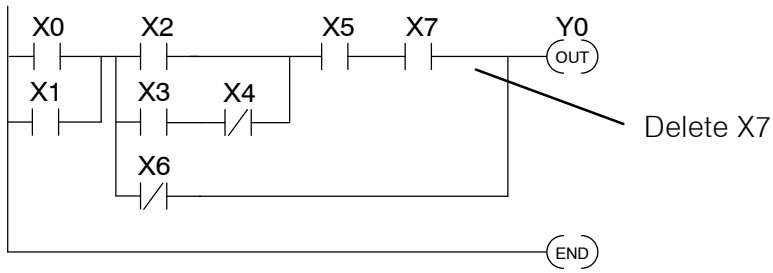
ENT

```
$     8
ORN     X 0006
```

Deleting an Instruction

Use the DELETE feature to remove an instruction from the program. This operation deletes the instruction that is currently being displayed, so make sure you are at the correct program address. Once you've deleted the instruction, the remaining addresses decrement. The following example shows the instructions, addresses, and corresponding Handheld displays for a small program.

Ladder Representation



Identify the Instruction

ADDRESS	INSTRUCTION	DESCRIPTION
0	STR X0	Starts branch 1 with X0
1	OR X1	Joins X1 in parallel with X0
—	—	—
Delete → 6	AND X5	Starts branch 4 with X5
7	AND X7	Adds X7 in series with X5
—	—	—
11	END	Ends the program

FIND the Address

AND X(IN) 7 FIND

```
          SEARCHING
AND X7
```

```
$    7
AND    X 0007
```

Delete the Instruction

SHFT DEL

```
$    7 DELETE INST?
AND    X 0007
```

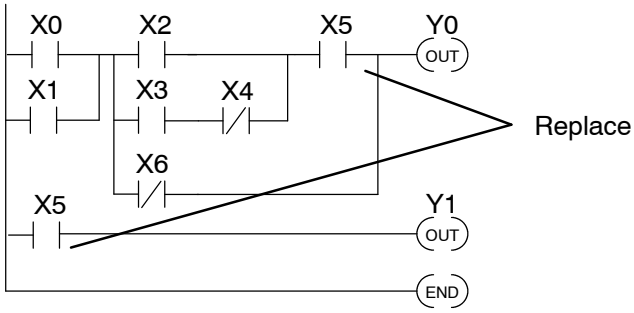
ENT

```
$    7
ORN    X 0006
```

Using Search and Replace

Search and Replace allows you to quickly change all occurrences of a specific instruction. For example, you can replace every instance of X5 with X10.

Ladder Representation



Use AUX 22, Change Reference

AUX 2 2 ENT ENT

```
AUX 22 CHANGE REFERENCE
OLD                    
```

X(IN) 5 ENT

```
AUX 22 CHANGE REFERENCE
TO                     
```

X(IN) 1 0 ENT

```
AUX 22 CHANGE REFERENCE
  X0005  -->  X0010 ?
```

ENT

```
REFERENCE CHANGE OK
OLD                    
```

SHFT EXIT (Clears display)

New Mnemonic Listing

New Reference

ADDRESS	INSTRUCTION	DESCRIPTION
0	STR X0	Starts branch 1 with X0
—	—	—
6	AND X10	Starts branch 4 with X5
—	—	—
10	STR X10	Begins a new rung with X5
—	—	—

Editing Programs During Run Mode

With the DL440 CPU you can edit programs during Run Mode and Test-Run Mode. (See Chapter 6 for more information on Test Modes.) You use AUX 14, Run Time Edit, to modify the program.

Most of the things you can do in Program Mode also apply. For example, you can use the same techniques to search for a specific instruction, search for a specific address, etc. However, you cannot use Search and Replace 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. Edits during Run Mode are ideally suited to small changes.

If the program requires major changes it is strongly recommended you switch the system to program mode and take all necessary precautions just as if you were starting the machine for the first time.

Selecting AUX 14

We've already shown you how to select the various AUX functions, but a few things are different with AUX 14.

- Once you select AUX 14 the RUN LED starts blinking. This indicates the a Run Mode edit is in progress.
- If you had displayed an address just before selecting AUX 14, that address will automatically appear. So, you can search for an address or instruction *before* you select AUX 14 or *after* you select AUX 14.

Select AUX 14, Run Time Edit

AUX 1 4 ENT

```
AUX 1* OPERATING MODE
AUX 14 RUN TIME EDIT
```

Press ENT to select AUX 14 and display the address

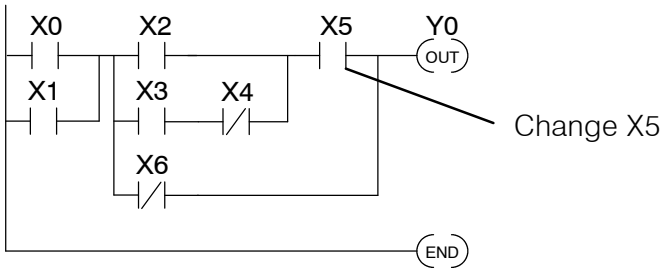
ENT

```
$xxxxx
STR     X 0001
```

Changing an Instruction During Run Mode

Once you've found the instruction you can change it very easily. The following example shows you how to change the X5 contact to X10.

Ladder Representation



Identify the Instruction

ADDRESS	INSTRUCTION	DESCRIPTION
0	STR X0	Starts branch 1 with X0
1	OR X1	Joins X1 in parallel with X0
—	—	—
—	—	—
6	AND X5	Starts branch 4 with X5
—	—	—
—	—	—
10	END	Ends the program

FIND the Address

AND X(IN) 5 FIND

```
            SEARCHING
AND X5
```

```
$     6
AND     X 0005
```

Change the Instruction

AND X(IN) 10 ENT

```
$     6
WANT TO ALTER?
```

Press CLR to abort the edit or ENT to accept

ENT

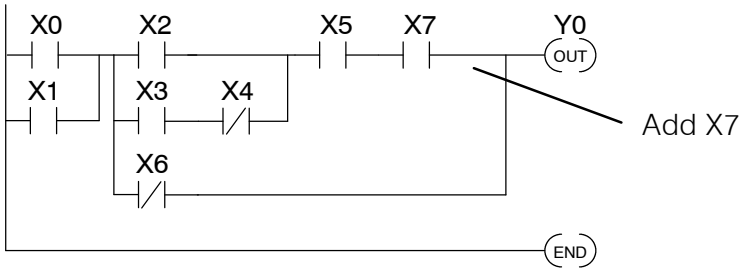
```
$     7
ORN     X 0006
```

(If you press ENT, the change is accepted and the next address is displayed. If you pressed CLR, the current address is displayed.

Inserting an Instruction During Run Mode

Inserting an instruction during Run Mode works almost exactly the same as it does during Program Mode. Remember, INSERT adds an instruction *before* the instruction that is being displayed and the remaining addresses increment.

Ladder Representation



Identify the Instruction

ADDRESS	INSTRUCTION	DESCRIPTION
0	STR X0	Starts branch 1 with X0
1	OR X1	Joins X1 in parallel with X0
—	—	—
6	AND X5	Starts branch 4 with X5
	AND X7	Adds X7 in series with X5
7	ORN X6	Joins X6 (NOT) in parallel
—	—	—
10	END	Ends the program

Insert before

FIND the Address

OR NOT X(IN) 6 FIND

```
            SEARCHING
ORN X6
```

```
$     7
ORN     X 0006
```

Insert the New Instruction

AND X(IN) 7 SHFT INS

```
$     7
WANT TO INSERT?
```

Press CLR to abort the edit or ENT to accept

ENT

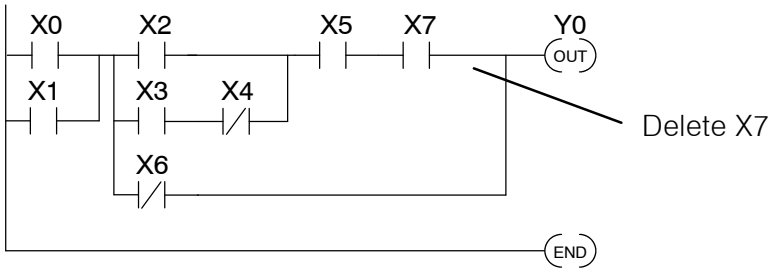
(If you press ENT, the change is accepted and the next address is displayed. If you pressed CLR, the current address is displayed.

```
$     8
ORN     X 0006
```

Deleting an Instruction During Run Mode

Deleting an instruction during Run Mode works almost exactly the same as it does during Program Mode. Remember, this operation deletes the instruction that is currently being displayed and the remaining addresses decrement.

Ladder Representation



Identify the Instruction

	ADDRESS	INSTRUCTION	DESCRIPTION
	0	STR X0	Starts branch 1 with X0
	1	OR X1	Joins X1 in parallel with X0
	—	—	—
Delete	6	AND X5	Starts branch 4 with X5
	7	AND X7	Adds X7 in series with X5
	—	—	—
	11	END	Ends the program

FIND the Address

AND X(IN) 7 FIND

```
          SEARCHING
AND X7
```

```
$     7
AND     X 0007
```

Delete the Instruction

SHFT DEL

```
$     7
WANT TO DELETE?
```

Press CLR to abort the edit or ENT to accept

ENT

(If you press ENT, the change is accepted and the next address is displayed. If you pressed CLR, the current address is displayed.

```
$     7
ORN     X 0006
```

5

Naming and Storing Programs

In This Chapter. . . .

— Program Names and Passwords
— Storing Programs on Memory Cartridges
— Storing Programs on Cassette Tapes

Program Names and Passwords

Program Names

The DL405 products can use program names for memory cartridges or cassette tapes. Program names are especially useful with cassette tapes since they can store multiple programs. The program name can be up to eight characters in length and can use any of the available characters (A-Z, 0-9).

AUX 51 allows you to enter a program name.

Use AUX 51 to name the CPU program

AUX 5 1 ENT ENT

```
AUX 51 MODIFY PGM NAME
```

Enter the program name

SHFT P R E S S SHFT
1

```
AUX 51 MODIFY PGM NAME
                  PRESS1
```

Press Enter to accept the name, or use the arrow keys to change it

ENT

```
PROGRAM   PRESS1
```

Press ***CLR*** to clear the display.

Password Protection

The DL440 CPU provides an extra measure of protection by allowing you to enter a password that prevents unauthorized machine operations. The password must be an eight-digit numeric (0-9) code. Once you've entered a password, you can remove it by entering all zeros (00000000). (This is the default from the factory.)

Use AUX 81 to enter or modify a password.

Use AUX 81 to name the CPU program

AUX 8 1 ENT ENT

```
CODE IS
00000000
```

Enter the new password

1 2 3 4 5 6 7
8 ENT

```
CODE IS
12345678
```

Press ***CLR*** to clear the display.

The password is stored in the memory cartridge. If you install the memory cartridge in another CPU or Handheld, the password protection remains in effect.

Locking the CPU with the Password

Once you've entered a password, you can lock the CPU against access. There are two ways to lock the CPU.

- The CPU is always locked after a power cycle (if a password is present).
- You can use AUX 83 and AUX 84 to lock and unlock the CPU.

WARNING: Make *sure* you remember the password *before* you lock the CPU. Once the CPU is locked you cannot view, change, or erase the password. You also cannot erase the memory cartridge and start over.

Use AUX 83 to lock the CPU

AUX 8 3 ENT ENT

```
CPU UNLOCKED
LOCK?
```

Press ENT to confirm the lock

ENT

```
CPU LOCKED
```

The message NO PASSWORD appears if you attempt to lock a CPU that does not have a password.

Use AUX 82 to unlock the CPU

AUX 8 2 ENT ENT

```
CPU LOCKED
PASSWORD
```

Enter the password to unlock the CPU

1 2 3 4 5 6 7 8 ENT

```
CPU UNLOCKED
```

The error message E541 WRONG PWORD appears if you enter an incorrect password. If you press ***CLR*** you can attempt to enter the password again.

NOTE: If you attempt to enter a password with less than 8 digits, the error message E504 BAD REF/VAL will appear. You cannot clear this message with the conventional methods (***CLR*** and ***SHFT EXIT***). You must disconnect the Handheld from the CPU and start over.

Storing Programs on Memory Cartridges

Types of Memory Cartridge

The type of memory storage available for use depends on the CPU you are using. The DL430 provides an EEPROM on the CPU for program storage and does not use memory cartridges. The DL440 can support three different types of memory cartridges (two sizes each) for program storage. You can purchase the memory cartridge with either a RAM, a UVPROM or an EEPROM chip. The RAM and EEPROM memory cartridges have a write protect jumper located inside the cartridge. When the cartridge is opened you can move the jumper to the protect position to prevent someone from accidentally erasing or changing the program.

The following table provides a brief overview of the types of memory cartridges for the DL440 CPU. See the DL405 User Manual for details.

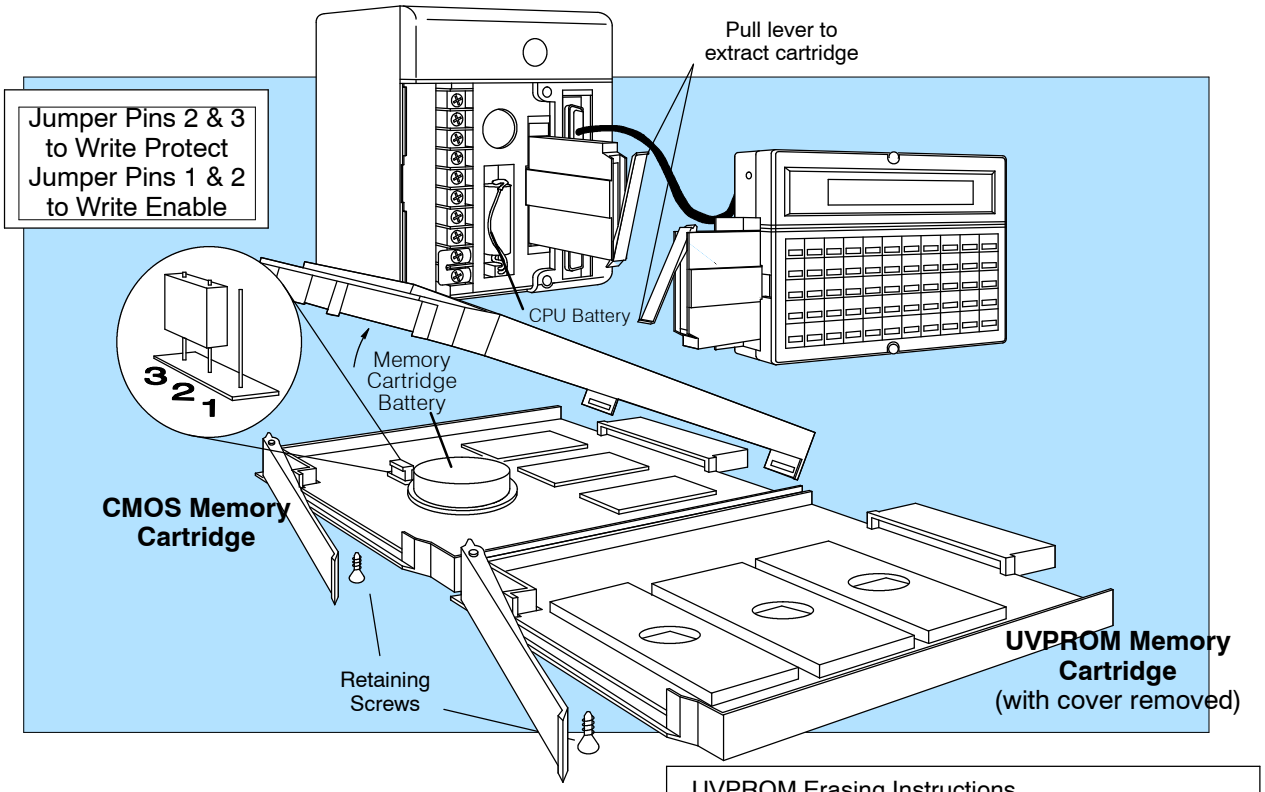
	D4-RAM-1	D4-RAM-2	D4-UV-1	D4-UV-2	D4-EE-1	D4-EE-2
Program Storage Capacity	7.5K	15.5K	7.5 K	15.5K	7.5K	15.5K
Cartridge Battery Type	Lithium	Lithium	None	None	None	None
Writing Cycle Life	N/A	N/A	1000	1000	>10,000	>10,000
Write Inhibit	Internal jumper	Internal jumper	No	No	Internal jumper	Internal jumper
Memory Clear Method	Electrical	Electrical	Ultraviolet light	Ultraviolet light	Electrical	Electrical

NOTE: When you purchase the UVPROM memory cartridge bear in mind it will be necessary for you to have either a RAM or an EEPROM memory cartridge for your program development. Once development is completed you can then use the Handheld Programmer to copy your application program to the UVPROM. We recommend the UVPROM memory cartridge option for applications which are mass produced and do not require frequent alterations.

DL440 Memory Cartridge

The diagram below displays a Memory Cartridge for the DL440. It shows how the memory cartridge fits in the CPU and in the handheld programmer. It also shows how to open the memory cartridge for selecting write protect (for CMOS RAM) or for erasing the UVPROM.

WARNING: Do not insert or remove a CPU memory cartridge while the power is connected. Your program or password may be corrupted if this occurs. A corrupted program can cause unpredictable operation which may result in a risk of injury to personnel or damage to equipment. If the password becomes corrupted, you cannot access the CPU.



UVPROM Erasing Instructions
1) Remove cartridge from CPU or HPP
2) Remove cartridge retaining screw
3) Remove cover
4) Place cartridge in UV erasing lamp typical 12,000□ w/cm^2 lamp @ 2.5cm for 15-20 minutes
5) Replace cover

Clearing the Memory Cartridge

There are two AUX Functions used to clear a memory cartridge installed in the Handheld Programmer. AUX 74, MC BLANK CHECK allows you to check the cartridge to make sure it is blank. AUX 75 CLEAR MC allows you to clear all data from a memory cartridge. (This is true for the RAM and EEPROM memory cartridges. The UVPROM cartridges must be erased with a UV light source.)

Use AUX 74 to see if a memory cartridge is blank

AUX 7 4 ENT ENT

```
AUX 74 MC BLANK CHECK
MC BLANK CHECK?
```

ENT (will result in one of two displays)

```
   E621 MC NOT BLANK
```

OR

```
MC IS BLANK
```

Use AUX 75 to clear a memory cartridge

AUX 7 5 ENT ENT

```
AUX 75 CLEAR MC
CLEAR MC?
```

ENT (may temporarily flash BUSY)

```
MC CLEARED
```

NOTE: If you copy data to a memory cartridge which has existing data stored on it, the new data could overwrite portions of the existing data and leave other portions as they previously existed resulting in a unreliable copy of your data. It is always recommended to clear non-blank memory cartridges prior to copying data to ensure you get a “clean” copy of your new data.

Copying Programs from the CPU

AUX 71 - CPU TO MC copies information from a CPU to a memory cartridge installed in the Handheld. If a memory cartridge is not present in the Handheld, you can just remove the memory cartridge from the CPU and place it in the Handheld, but if you want to keep the CPU running, you should use this procedure.

You can copy different portions of CPU memory to the memory cartridge.

Option and Memory Type	DL440 Range	DL430 Range
1:PGM — Program	$00000 - $07679 (7.5K program memory) $00000 - $015871 (15.5K program memory)	$00000 - $03583
2:V — V memory	$00000 - $37777	$00000 - $07777
3:SYS — System memory	Non-selectable copies all system parameters	

A single memory cartridge cannot hold an entire system. You may have to use more than one cartridge. If so, put V memory on a cartridge by itself.

The following displays change slightly if you use UVPROM or EEPROM cartridges.

Use AUX 71 to copy the program to the MC

AUX 7 1 ENT ENT

```
AUX 71 CPU TO MC
1:PGM/2:V/3:SYS
```

Select a memory type

1 ENT (PGM for example)

```
AUX 71 CPU TO MC
1st     $ 00000
```

Select the starting address (or press ENT to start at 00000)

ENT

```
AUX 71 CPU TO MC
END     $07679
```

Select the ending address (or press ENT to copy the entire range)

ENT

```
AUX 71 CPU TO MC
$ 00000 - $07679 ?
```

Confirm the selection

ENT (This may take a minute or so.)

```
AUX 71 CPU TO MC
CMOSRAM 08K        0x
```

```
OK
```

Clear the display by pressing ***CLR***.

Writing Programs to the CPU

AUX 72 – MC TO CPU copies information from an Handheld memory cartridge to the CPU. If a memory cartridge is not present in the CPU, you can just remove the memory cartridge from the Handheld and place it in the CPU.

You can copy different types of information from the memory cartridge.

Option and Memory Type	DL440 Range	DL430 Range
1:PGM — Program	$00000 – $07679 (7.5K program memory) $00000 – $015871 (15.5K program memory)	$00000 – $03583
2:V — V memory	$00000 – $37777	$00000 – $07777
3:SYS — System memory	Non-selectable copies all system parameters	

Use AUX 72 to copy the program to the CPU

AUX 7 2 ENT ENT

```
AUX 72 MC TO CPU
1:PGM/2:V/3:SYS
```

Select a memory type

1 ENT (PGM for example)

```
AUX 72 MC TO CPU
1st     $ 00000
```

Select the starting address (or press ENT to start at 00000)

ENT

```
AUX 72 MC TO CPU
END     $07679
```

Select the ending address (or press ENT to copy the entire range)

ENT

```
AUX 72 MC TO CPU
$ 00000 – $07679 ?
```

Confirm the selection

ENT (This may take a minute or so.)

```
AUX 72 MC TO CPU
CMOSRAM 08K        0x
```

```
OK
```

Clear the display by pressing ***CLR***.

Comparing CPU and Handheld Programs

AUX 73 - CMP MC TO CPU compares the Handheld memory cartridge program with the CPU program.

You can compare different types of information.

Option and Memory Type	DL440 Range	DL430 Range
1:PGM — Program	$00000 - $07679 (7.5K program memory) $00000 - $015871 (15.5K program memory)	$00000 - $03583
2:V — V memory	$00000 - $37777	$00000 - $07777
3:SYS — System memory	Non-selectable copies all system parameters	

Use AUX 73 to compare the programs

AUX 7 3 ENT ENT

```
AUX 73 CMP MC TO CPU
1:PGM/2:V/3:SYS
```

Select a memory type

1 ENT (PGM for example)

```
AUX 73 CMP MC TO CPU
1st     $ 00000
```

Select the starting address (or press ENT to start at 00000)

ENT

```
AUX 73 CMP MC TO CPU
END     $07679
```

Select the ending address (or press ENT to compare the entire range)

ENT

```
AUX 73 CMP MC TO CPU
$ 00000 - $07679 ?
```

Confirm the selection

ENT (This may take a minute or so.)

```
AUX 73 CMP MC TO CPU
CMOSRAM 08K        0x
```

One of two displays will appear

(The programs are different.)

```
        E640 MISCOMPARE
```

(The programs are the same.)

```
VERIFICATION OK
```

Clear the display by pressing ***CLR***.

Storing Programs on Cassette Tapes

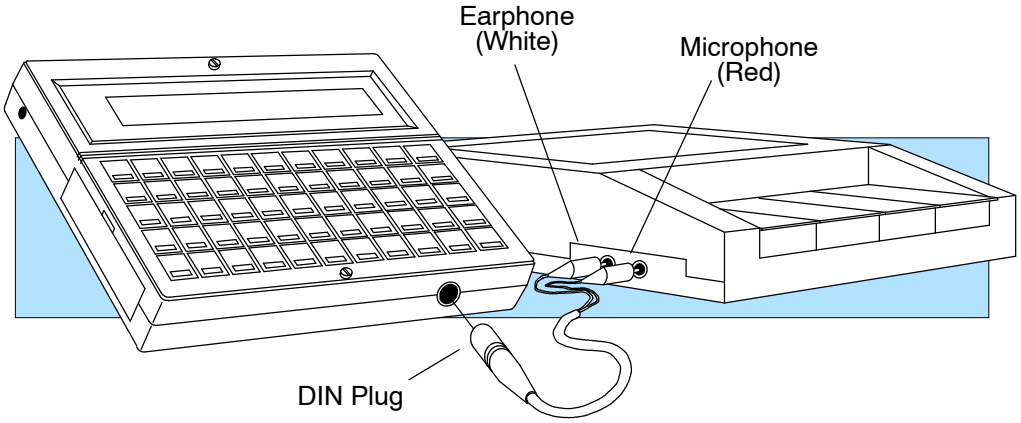
Cassette Characteristics

Although memory cartridges are the preferred method of program storage, you can also copy the programs from the CPU to cassette tapes. The advantage is you can generally store several programs on a single cassette tape.

When you select a recorder, choose one designed for use with Personal Computers (PCs). These types of recorders are much more suitable than those used for normal audio recordings. (Most audio recorders will not record or play the digital information accurately.)

Connecting the Cassette Recorder

The cassette recorder cable connects to the DIN plug receptacle located on the bottom of the Handheld.



NOTE: Some recorders will not operate properly if both the earphone and microphone plugs are connected. You can avoid this by disconnecting the earphone plug while recording and disconnecting the microphone plug during playback.

Program Names on Cassettes

Since it is very easy to store multiple programs on a single cassette it is very important idea to name each program. You may recall you can enter a name for the CPU program. The cassette program name does not have to be the same. For example, the CPU name may be PRESS1 and the tape name may be STATION1.

There are three areas of CPU memory that can be transferred (PGM, V, and SYS). It is also a good idea to give each of these a separate program name. For example, you could use three programs, STAT1PGM, STAT1V, and STAT1SYS, for Station 1.

Writing a Program to the Cassette

You can copy different portions of CPU memory to the cassette tape

Option and Memory Type	DL440 Range	DL430 Range
1:PGM — Program	$00000 – $07679 (7.5K program memory) $00000 – $015871 (15.5K program memory)	$00000 – $03583
2:V — V memory	$00000 – $37777	$00000 – $07777
3:SYS — System memory	Non-selectable copies all system parameters	

NOTE: Remember tape programs are stored sequentially. It is very easy to overwrite existing programs if you do not position the tape correctly before beginning this procedure. Use the tape counter on the recorder to keep track of program locations.

Use AUX 78 to copy a program to the cassette

AUX 7 8 ENT ENT

```
AUX 78 MC TO TAPE
TAPE NAME:
```

Enter the name

SHFT S T A T I O
N SHFT 1 ENT

```
AUX 78 MC TO TAPE
1:PGM/2:V/3:SYS
```

Select a memory type

1 ENT (PGM for example)

```
AUX 78 MC TO TAPE
1st    $ 00000
```

Select the starting address (press ENT to start at $00000)

ENT

```
AUX 78 MC TO TAPE
END    $07679
```

Select the ending address (or press ENT to copy the entire range)

ENT

```
AUX 78 MC TO TAPE
$ 00000-$ 07679 START?
```

Confirm the selection

ENT

```
AUX 78 MC TO TAPE
BUSY
```

Start the cassette recorder

REC OR REC PLAY

(Program is being transferred.
1 minute per K)

```
OK
```

Transfer is complete

Stop the cassette recorder

STOP

Clear the display by pressing ***CLR***.

Reading Programs from Cassette Tapes

Use AUX 77 to read a program from a cassette tape. Before you begin the procedure make sure you have positioned the tape at the beginning, or just before the location of the program you want to read. Set the tape recorder TONE control to the midway position and turn the volume off. (You will adjust the volume later.)

Use AUX 77 to read a program from the cassette

AUX 7 7 ENT ENT

```
AUX 77 TAPE TO MC
TAPE NAME:
```

Enter the name (press ENT if a name was not used)

SHFT S T A T I O
N SHFT 1 ENT

```
AUX 77 TAPE TO MC
1:PGM/2:V/3:SYS
```

Select a memory type

1 ENT (PGM for example)

```
AUX 77 TAPE TO MC
1st    $ 00000
```

Select the starting address (press ENT to start at $00000)

ENT

```
AUX 77 TAPE TO MC
$ 00000 START?
```

Confirm the selection

ENT

```
AUX 77 TAPE TO MC
$ 00000 BUSY
```

Start the cassette recorder

PLAY

Increase the volume until the asterisk appears. If the volume is too high, the asterisk will disappear. Adjust the volume in the middle of this range. (You have 10 seconds to complete this operation.)

```
*
```

```
PGM FOUND
```

```
OK
```

Transfer is complete

Stop the cassette recorder

STOP

Comparing Cassette and CPU Programs

Use AUX 79 to compare a cassette program to a program stored in the CPU. Before you begin the procedure make sure you have positioned the tape at the beginning, or just before the location of the program you want to read. Set the tape recorder TONE control to the midway position and turn the volume off.

Use AUX 79 to compare cassette and CPU programs

AUX 7 9 ENT ENT

```
AUX 79 CMP MC TO TAPE
TAPE NAME:
```

Enter the name (press ENT if a name was not used)

SHFT S T A T I O
N SHFT 1 ENT

```
AUX 79 CMP MC TO TAPE
1:PGM/2:V/3:SYS
```

Select a memory type

1 ENT (PGM for example)

```
AUX 79 CMP MC TO TAPE
1st    $ 00000
```

Select the starting address (press ENT to start at $00000)

ENT

```
AUX 79 CMP MC TO TAPE
$ 00000 VERIFY?
```

Confirm the selection

ENT

```
AUX 79 CMP MC TO TAPE
$ 00000 BUSY
```

Start the cassette recorder

PLAY

Increase the volume until the asterisk appears. If the volume is too high, the asterisk will disappear. Adjust the volume in the middle of this range. (You have 10 seconds to complete this operation.)

```
*
```

```
PGM FOUND
```

One of two displays will appear

```
      E640 MISCOMPARE
```

Stop the cassette recorder

STOP

```
VERIFICATION OK
```

As you've seen, entering and storing programs with the Handheld is a pretty simple task. Once you've got the program entered and the machine is up and running, you can use the Handheld to monitor and change machine operations almost as easily. The next chapter shows the details.

6

System Monitoring and Troubleshooting

In This Chapter. . . .

Troubleshooting Suggestions

The Handheld is very useful in troubleshooting your machine. As with most any problem, you have to find it before you can fix it. There are several operations and features that help you quickly find the exact cause of system problems.

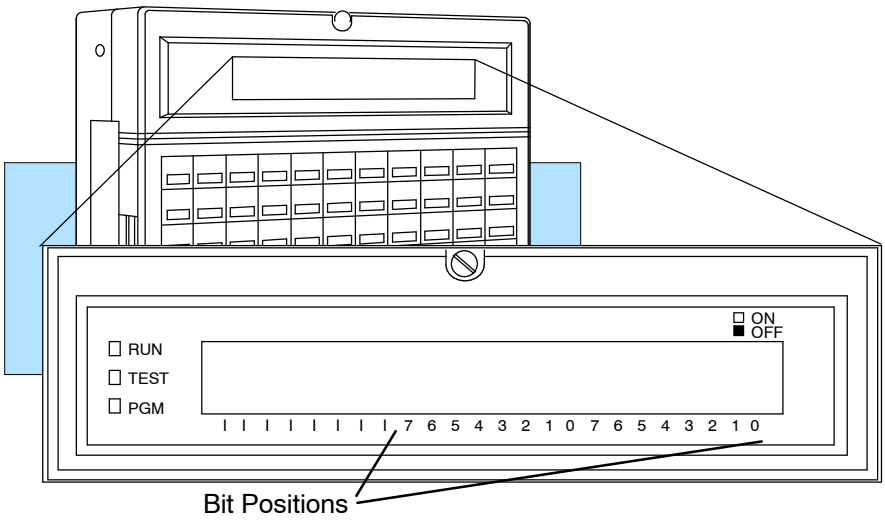
- Monitor Discrete I/O Points — to examine I/O power flow for individual I/O points.
- Force Discrete I/O Points — to examine machine sequences or inconsistencies.
- Monitor V-Memory Locations — to examine word locations to determine if correct values are being used.
- Change V-Memory Values — to force word locations with different values.
- Monitor Timer/Counter Values — to adjust machine timing elements.
- Monitor CPU scan time (in milliseconds) — view the maximum, minimum, and current scan times to adjust scan related problems.
- Use Test Modes — to run a fixed number of scans and examine output status.
- Use I/O Diagnostics — to pinpoint I/O errors.
- Understand Error Codes — to utilize many automatic error checks.

Monitoring Discrete I/O Points

You can monitor up to 16 discrete points at one time. The points can be from the following data types.

- X inputs
- Y output
- GX remote I/O points
- C control relays
- Stage bits
- Timer/Counter bits
- Special relays

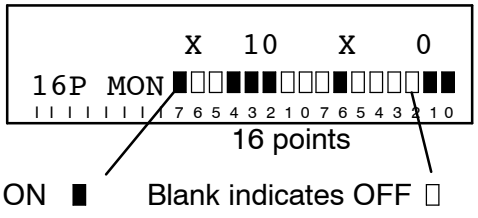
If you examine the Handheld Programmer display, you will notice several numbers printed at the bottom. These numbers help you identify the point that you need.



Use the following keystrokes to monitor discrete points. (To select a different data type, use the corresponding Instruction Reference key instead of the one shown.)

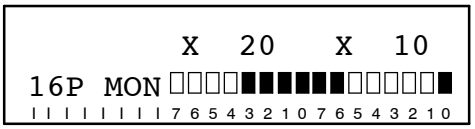
Select the data type and range to monitor

X(IN) 0 BIT ST



Use the PREV and NXT keys to scroll through additional points

NXT

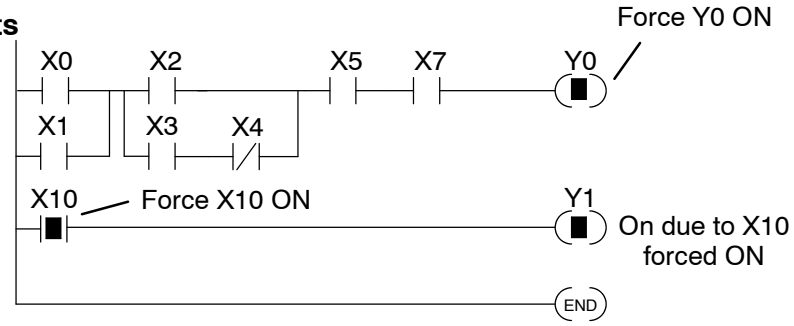


Forcing Discrete I/O Points

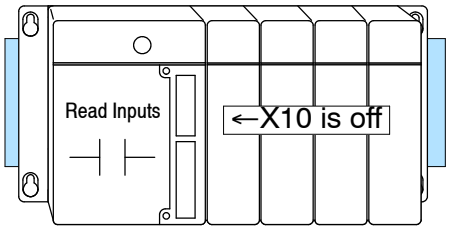
You can also force I/O points from the status display by using the ON and OFF keys. You can display the points first, or, you can force the points from a clear display.

It is important to note that the DL405 CPUs only retain the forced value for one scan if the output point is used in the logic program or if the input point used corresponds to module that is installed in the base. The following example shows how the forcing actually works.

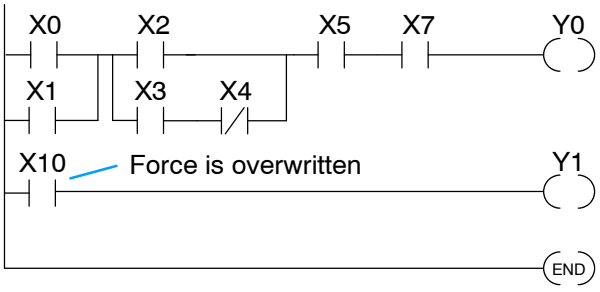
Force I/O Points



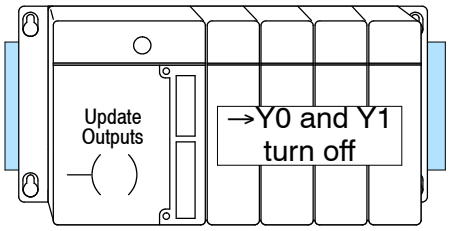
Next Scan



CPU reads the I/O status from the modules. Sees that X10 is off, overwrites the force command and turns off X10.



Logic is solved. X10, even though previously forced on, is turned off. Y0 and Y1 are turned off since conditions are not met.



CPU updates the output status with the results obtained from the logic execution. Y0 and Y1 were turned off.

NOTE: If you use a CR as an input, you will not have the “one scan” problem.

The following example shows the keystrokes required to force an I/O point.

WARNING: Depending on your application, forcing I/O points may cause unpredictable machine operation that can result in a risk of personal injury or equipment damage.

Select the data type and range to monitor

Y(OUT) 0 BIT ST

```
          Y  10    Y   0
16P MON □□□□□□□□□□□□□■■
```

Select the point and ON or OFF

Y(OUT) 0 SHFT OFF

```
          Y  10    Y   0
16P MON □□□□□□□□□□□□□■□
```

Y0 is now off

Or, from a clear display. . .

Y(OUT) 0 SHFT ON

(Note that you cannot see the I/O status with this method.)

```
Y0
```

Monitoring V-Memory Locations

You can also use the Handheld to monitor and change V memory locations. This is an especially useful feature, since almost all DL405 system data is mapped into V memory. The following steps show you how to monitor V-memory locations.

Select the location to monitor

V 2 5 0 0 WD ST

```
        V 2501   V 2500
V MON    0000     0256
```

Value is displayed

Use the PREV and NXT keys to scroll through additional points

NXT

```
        V 2502   V 2501
V MON    0000     0000
```

Changing V-Memory Values

Select the location to monitor

V 2 5 0 1 WD ST

```
        V 2502   V 2501
V MON    0000     0000
```

Use the K(CON) key to load a value

K(CON) 4 3 2 ENT

```
        V 2502   V 2501
V MON    0000     0432
```

Monitoring Timer/Counter Values

Timer and Counter current values are mapped into V-memory locations. Use the procedure for displaying V memory to examine these current values. (Appendix A provides a complete listing of the memory map for the DL405 systems.)

Monitoring the CPU Scan Time

The DL405 CPUs have a "watchdog" timer that is used to monitor the scan time. The default value set from the factory is 200 ms. If the scan time exceeds the watchdog time limit, the CPU automatically leaves RUN mode and enters PGM mode. The Handheld displays 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.

Use AUX 53 to view the scan time

AUX 5 3 ↵ ↵

```
  SCAN     MAX     MIN
  0004     0006    0002
```

The CPU must be in PGM or TEST-PGM mode before you can change the watchdog timer value.

Use AUX 55 to change the watchdog value

AUX 5 5 ↵ ↵

```
AUX 55 SET WATCHDOG TMR
0200 MSEC
```

current setting

Enter the new time value (in milliseconds)

1 0 0 ↵

```
AUX 55 SET WATCHDOG TMR
OK
```

Test Modes

TEST-PGM and TEST-RUN

Test Mode allows the CPU to start in TEST-PGM mode, enter TEST-RUN mode, run a fixed number of scans, and then return to TEST-PGM mode. You can select from 1 to 65,535 scans.

Use the following keystrokes to enter the Test Modes. (The actual mode entered when you first select Test Mode depends on the mode of operation at the time you make the request. If the CPU is in RUN mode, then TEST-RUN is entered. If the mode is PGM, then TEST-PGM is entered.)

Use AUX 12 to enter Test Mode

AUX 1 2 ENT ENT

```
MODE = TEST-PGM
```

While in TEST-PGM mode you can specify the number of scans by entering. . .

CLR 1 SHFT TEST

(CPU runs scans and returns to TEST-PGM)

```
NO. OF SCANS?
```

To switch from TEST-RUN to TEST-PGM mode . . .

CLR 2 SHFT TEST

ENT (to confirm the return to TEST-PGM)

```
STOP SCAN?
```

To switch from TEST—PGM to TEST-RUN mode . . .

CLR 3 SHFT TEST

ENT (to confirm the entry of TEST-RUN)

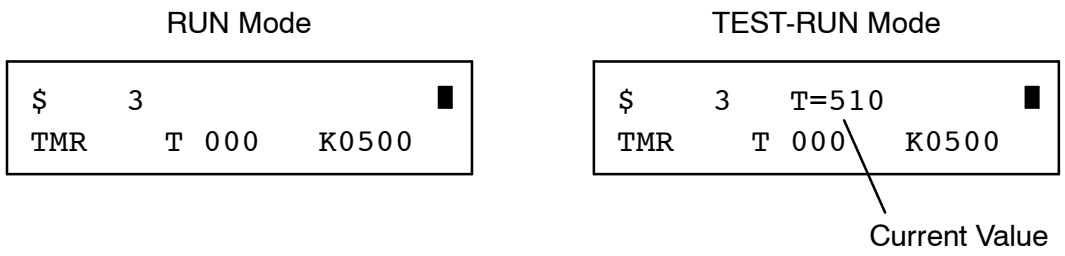
```
START SCAN?
```

You gain some advantages by using Test Mode.

- The status displays are more detailed.
- You can enable the CPU to hold output states.

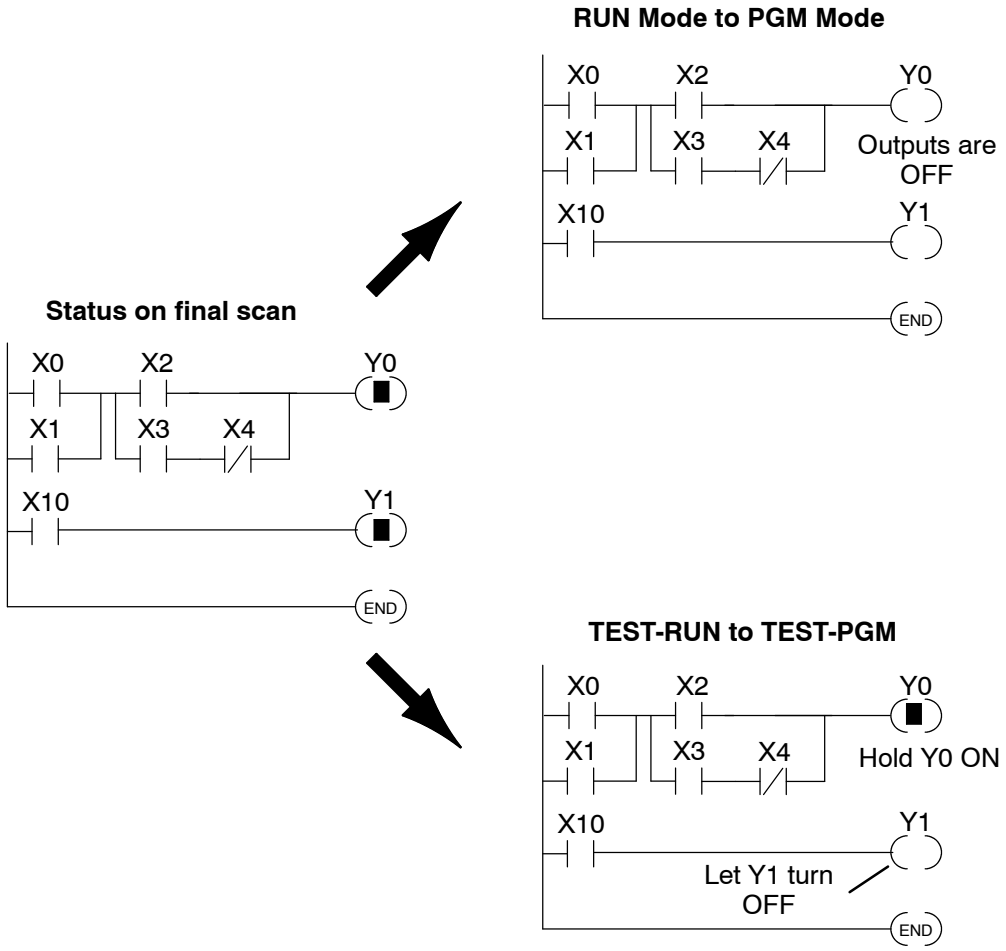
Test Mode Displays

For some instructions, the TEST-RUN mode display is more detailed than the status displays shown in RUN mode. The following diagram shows an example of a Timer instruction display during TEST-RUN mode.



Holding Output States

In normal RUN mode, the outputs are turned off when you return to PGM mode. In TEST-RUN mode you can set each individual output to either turn off, or,hold its last output state on the transition to TEST-PGM mode. The ability to hold the output states is especially useful, since It allows you to maintain key system I/O points for examination. The following diagram shows the differences between RUN and TEST-RUN modes.



You can use AUX 58 to configure each individual output. The following keystrokes show an example.

Use AUX 58 to configure the output state

AUX 5 8 ENT ENT

```
AUX 58 TEST OPERATIONS
  Y0000   OFF
```

To hold the last state on mode transition . . .

SHFT ON

```
AUX 58 TEST OPERATIONS
   Y0000  ON
```

To turn the output off on mode transition . . .

SHFT OFF

```
AUX 58 TEST OPERATIONS
   Y0000  OFF
```

To select an output for configuration

CLR Y(OUT) 5 0 NXT

```
AUX 58 TEST OPERATIONS
  Y0050   OFF
```

(You can also use PREV and NXT to sequentially step through the outputs.)

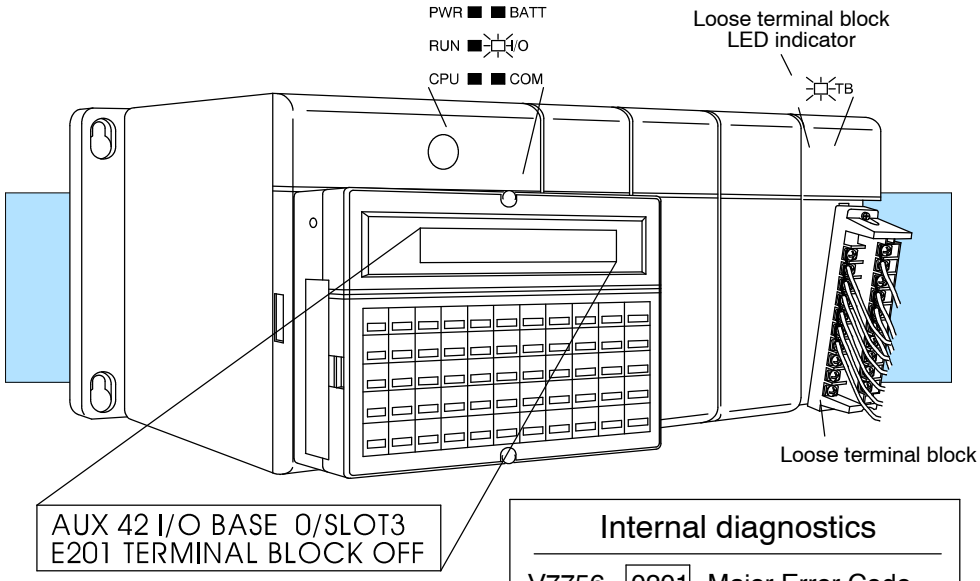
I/O Diagnostics

Diagnostic Indicators

The DL405 system provides many diagnostic features that normally are not found on much larger, more expensive, PLCs. There are three primary tools that help identify I/O errors.

- CPU status LEDs
- I/O module status LEDs
- AUX 42, I/O Diagnostics

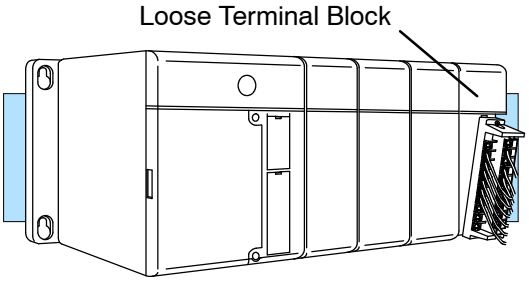
The following diagram shows how these three things can be used to locate the source of the error very quickly.



Internal diagnostics		
V7756	0201	Major Error Code
V7760	0003	Error Location (Base 00, Slot 03)
V7762	0201	Module Error Code
SP45	⊣⊢	I/O Error

Using AUX 42

When an I/O error occurs, the Handheld automatically displays a message that indicates an I/O error. This example display shows the error message.




```
E2** DIAG ERROR AUX 42
```

The display indicates that you should use AUX 42 to pinpoint the location of the error.

Use AUX 42

AUX 4 2 ENT ENT

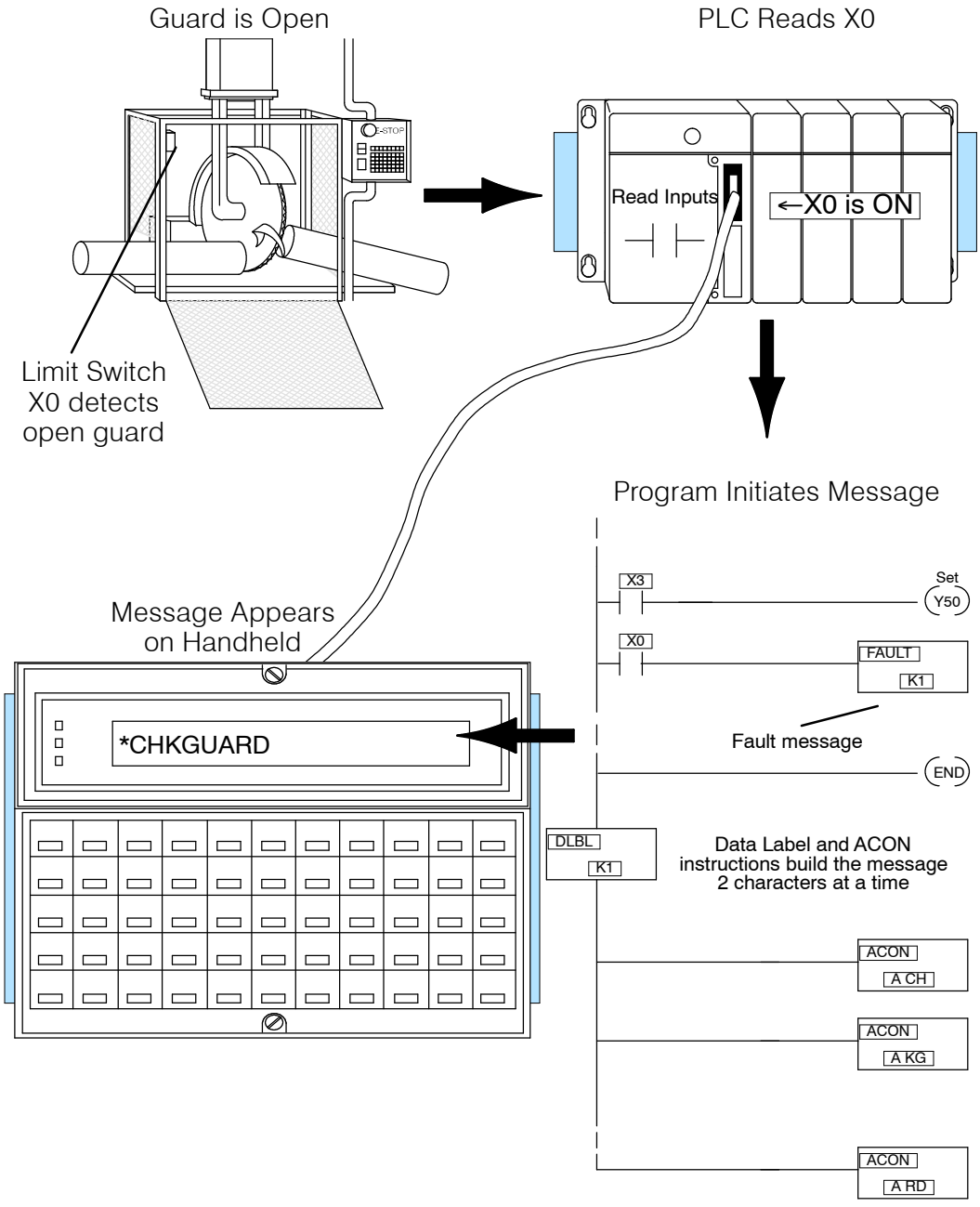
```
AUX 42 I/O BASE0/SLOT1
E201 TERMINAL BLOCK OFF
```

The error codes listing, provided later in this chapter, shows all of the possible I/O error messages.

Custom Messages

The standard diagnostic tools help you identify system problems quickly, but what if you need messages that tell the machine operator to perform a certain task? You can purchase an operator interface, or you can use the Handheld Programmer (which you already have) as a low-cost message display. The Handheld Programmer is not a high performance display (due to the amount of time it may take to display the message). If you need a fast display time, you should probably consider another type of message display.

The following diagram shows how the message display capability works.



If you have a DL440 CPU, you can easily build and display up to 64 custom messages. The messages can be up to 23 characters in length and contain both text and numeric values. The messages are part of the RLL program and are displayed automatically on the Handheld Programmer during RUN mode.

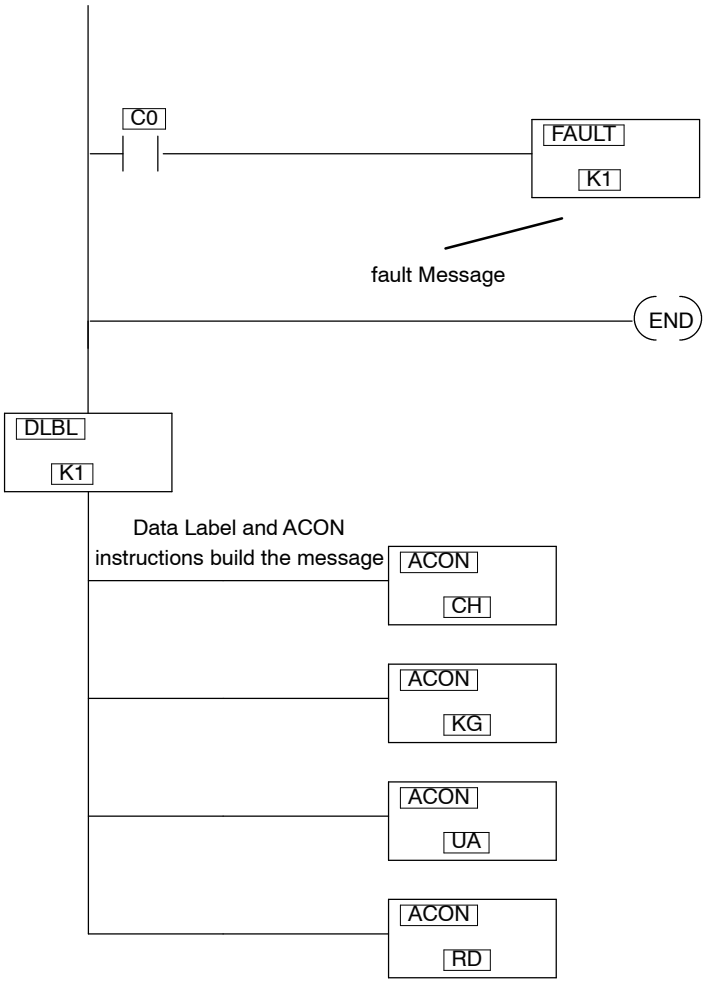
The DL440 CPU has several instructions that are used to build operator messages. Detailed explanations of the following instructions are included in the DL405 User Manual.

- FAULT — the Fault instruction is an output box instruction that lets the program know which message to display.
- DLBL — the Data Label instruction is included *after* the END statement and notes the beginning of a message.
- ACON — the ASCII Constant instruction is used as an output box for the ASCII portion of the message. (You can also display the contents of a V-memory location instead of ASCII text.)
- NCON — the Numeric Constant instruction is used as an output box for any numeric constant portion of the message.
- MOVMC — the Move Data Label to V-memory Area instruction is used to embed variables, such as timer or counter values, into a text message.

The next two pages show an example and the keystrokes required to enter a very simple text message with the Handheld.

NOTE: It is *much* easier to enter text message programs with ***Direct***SOFT than it is with the Handheld Programmer. This is because you can only enter two ASCII characters per ACON instruction with the Handheld. This is not the case with ***Direct***SOFT, which allows you to enter up to 8 characters per ACON instruction. ***Direct***SOFT also supports other characters not available on the handheld keypad.

The following program will display the message CHKGUARD on the Handheld when C0 is on.



NOTE: The DL430 CPU also allows you to use the Fault instruction, but you cannot create text messages because the DLBL, ACON, NCON, and MOVMC instructions do not exist. If you're using a DL430, you can replace the constant (K) value used with the Fault instruction with a V-memory address. Then, the Handheld will display whatever code is stored in that V-memory address. For example, if you used a Fault message with a reference to V2000 and V2000 held the constant 1234, then the Handheld would display that constant value. This is still useful because you can easily build a chart that would show the operator what the various codes mean.

If you have a DL440 CPU handy, enter the program on the next page to see how it actually works. Once you've entered the program, put the CPU in RUN mode and force C0 on to display the message.

Enter the first contact

STR X 0 ENT

Starting at Address 0

```
$     0
STR X0
```

Enter the Fault instruction

SHFT F A U L T K(CON)
1 ENT

```
$     1
FAULT K1
```

Enter the END statement

END ENT

```
$     3
END
```

Enter the DLBL instruction

SHFT D L B L K(CON) 1
ENT

```
$     4
DLBL K1
```

Enter the ACON instruction and the first two letters

SHFT A C O N ASC
SHFT C H ENT

```
$     6
ACON    A CH
```

Enter the next two characters

SHFT A C O N ASC
SHFT K G ENT

```
$     7
ACON    A KG
```

Enter the next two characters

SHFT A C O N ASC
SHFT U A ENT

```
$     8
ACON    A UA
```

Enter the last two characters

SHFT A C O N ASC
SHFT R D ENT

```
$     9
ACON    A RD
```

Checking the Error Message Tables

Two Types of Tables

The DL440 CPU will automatically log any system error codes and custom messages created with the FAULT instructions. The CPU logs the error code, the date, and the time the error occurred. There are two separate tables that store this information.

- Error Code Table – the system logs up to 32 errors in the table. When an error occurs, the errors already on the table are pushed down and the most recent error is loaded into the top slot. If the table is full when an error occurs, the oldest error is pushed out (erased) from the table.
- Message Table – the system logs up to 16 messages in this table. When a message is triggered, the messages already stored in the table are pushed down and the most recent message is loaded into the top slot. If the table is full when an error occurs, the oldest message is pushed out (erased) of the table.

The following diagram shows an example of an error table for messages.

Date	Time	Message
1993-05-26	08:41:51:11	*Conveyor-2 stopped
1993-04-30	17:01:11:56	* Conveyor-1 stopped
1993-04-30	17:01:11:12	* Limit SW1 failed
1993-04-28	03:25:14:31	* Saw Jam Detect

Viewing the Error Table

You can use AUX Function 5C to show the error codes.

Use AUX 5C to view the tables

AUX 5 SHFT C ENT

```
AUX 5C SHOW ERR/MSG
      ERR OR MSG
```

Press ENT to select Error Messages

ENT

(The most recent error is displayed. You can also use PREV and NXT to sequentially step through the errors.)

```
E151 BAD COMMAND
04/22/93 17:30:00
```

Viewing the Message Table

You use the same AUX function, 5C, to show the messages.

Use AUX 5C to view the tables

AUX 5 SHFT C ENT

```
AUX 5C SHOW ERR/MSG
      ERR OR MSG
```

Use the arrow key to select MSG

SHFT → ENT

(The most recent message is displayed. You can also use PREV and NXT to sequentially step through the messages.)

```
PUMP 3 FAILED
04/22/93 17:30:00
```

Error Codes

The following table lists the error codes that may appear on the Handheld.

DL405 Error Code	Description
E001 CPU Fatal Error	You may possibly clear the error by power cycling the CPU. If the error returns replace the CPU.
E003 Software Time-out	If the program scan time exceeds the time allotted to the watchdog timer, this error will occur. SP51 will be on and the error code will be stored in V7755. To correct this problem add RSTWT instructions in FOR NEXT loops and subroutines or using AUX 55 extend the time allotted to the watchdog timer.
E004 Invalid Instruction (DL440 only)	The application program has changed for some reason. SP44 will be on and the error code will be stored in V7755. This problem may possibly be due to electrical noise. Use AUX21 to check the program syntax and correct where necessary or clear the memory and re-download the program. Correct any grounding problems. If the error returns replace the CPU.
E041 CPU Battery Low	The CPU battery is low and should be replaced. SP43 will be on and the error code will be stored in V7757.
E043 Memory Cartridge Battery low (DL440 only)	The Memory Cartridge battery is low and should be replaced. SP43 will be on and the error code will be stored in V7757.
E099 Program Memory Exceeded	If the compiled program length exceeds the amount of available CPU RAM this error will occur. SP52 will be on and the error code will be stored in V7755. Reduce the size of the application program.
E101 CPU MC Missing (DL440 only)	The CPU Memory Cartridge has failed or is missing. SP44 will be on and the error code will be stored in V7755. Install or replace the Memory Cartridge.
E104 Write Failed (DL440 only)	A write to the CPU Memory Cartridge was not successful. The Memory Cartridge may be write protected. Disassemble and check the jumper. If the error still occurs replace the Memory Cartridge.
E151 Invalid Command	A parity error has occurred in the application program. SP44 will be on and the error code will be stored in V7755. This problem may possibly be due to electrical noise. Clear the memory and re-download the program. Correct any grounding problems. If the error returns replace the Memory Cartridge or CPU.
E155 RAM Failure	A checksum error has occurred in the system RAM. SP44 will be on and the error code will be stored in V7755. This problem may possibly be due to a low battery, electrical noise or a CPU RAM failure. Clear the memory and re-download the program. Correct any grounding problems. If the error returns replace the CPU.
E2** I/O Module Failure	An I/O module has failed. Run AUX42 to determine the actual error.
E201 Terminal Block Missing	A terminal block is loose or missing from an I/O module. SP45 will be on and the error code will be stored in V7756.

DL405 Error Code	Description
E202 Missing I/O Module	An I/O module has failed to communicate with the CPU or is missing from the base. SP45 will be on and the error code will be stored in V7756. Run AUX42 to determine the slot and base location of the module reporting the error.
E203 Blown Fuse	A fuse has blown in an I/O module. SP45 will be on and the error code will be stored in V775. 6Run AUX42 to determine the slot and base location of the module reporting the error.
E206 User 24V Power Supply Failure.	The 24VDC power supply being used to power output modules has failed. SP45 will be on and the error code will be stored in V7756. Run AUX42 to determine the slot and base location of the module reporting the error.
E250 Communication Failure In The I/O Chain	A failure has occurred in the local I/O system. The problem could be in the base, expansion cable or I/O Expansion Unit power supply. Check all cabling between bases and replace faulty hardware if necessary. SP45 will be on and the error code will be stored in V7755. Run AUX42 to determine the base location reporting the error.
E252 New I/O CFG	This error occurs when the auto configuration check is turned on in the CPU and the actual I/O configuration has changed either by moving modules in a base or changing types of modules in a base. You can return the modules to the original position/types or run AUX45 to accept the new configuration. SP47 will be on and the error code will be stored in V7755.
E261 I/O Address Conflict (DL440 only)	Overlapping addresses have been assigned while manually configuring the I/O. Correct the address assignments using AUX46. SP45 will be on and the error code will be stored in V7755.
E262 I/O Out Of Range	An out of range I/O address has been encountered in the application program. Correct the invalid address in the program. SP45 will be on and the error code will be stored in V7755.
E263 Configured I/O Address Out Of Range (DL440 only)	Out of range addresses have been assigned while manually configuring the I/O. Correct the address assignments using AUX46. SP45 will be on and the error code will be stored in V7755.
E264 Duplicate I/O Reference (DL440 only)	Duplicate addresses have been assigned while manually configuring the I/O. Correct the address assignments using AUX46.
E311 HPP Comm Error 1	A request from the handheld programmer could not be processed by the CPU. Clear the error and retry the request. If the error continues replace the CPU. SP46 will be on and the error code will be stored in V7756.

DL405 Error Code	Description
E312 HPP Comm Error 2	A data error was encountered during communications with the CPU. Clear the error and retry the request. If the error continues check the cabling between the two devices, replace the handheld programmer, then if necessary replace the CPU. SP46 will be on and the error code will be stored in V7756.
E313 HPP Comm Error 3	An address error was encountered during communications with the CPU. Clear the error and retry the request. If the error continues check the cabling between the two devices, replace the handheld programmer, then if necessary replace the CPU. SP46 will be on and the error code will be stored in V7756.
E316 HPP Comm Error 6	A mode error was encountered during communications with the CPU. Clear the error and retry the request. If the error continues replace the handheld programmer, then if necessary replace the CPU. SP46 will be on and the error code will be stored in V7756.
E320 HPP Comm Time-out	The CPU did not respond to the handheld programmer communication request. Check to insure cabling is correct and not defective. Power cycle the system if the error continues replace the CPU first and then the handheld programmer if necessary.
E321 Comm Error	A data error was encountered during communication with the CPU. Check to insure cabling is correct and not defective. Power cycle the system and if the error continues replace the CPU first and then the handheld programmer if necessary.
E360 HPP Peripheral Port Time-out	The device connected to the peripheral port did not respond to the handheld programmer communication request. Check to insure cabling is correct and not defective. The peripheral device or handheld programmer could be defective.
E4** No Program	A syntax error exist in the application program. The most common is a missing END statement. Run AUX21 to determine which one of the E4** series of errors is being flagged. SP52 will be on and the error code will be stored in V7755.
E401 Missing END Statement	All application programs must terminate with an END statement. Enter the END statement in appropriate location in your program. SP52 will be on and the error code will be stored in V7755.
E402 Missing LBL (DL440 only)	A GOTO, GTS, MOV MC or LD LBL instruction was used without the appropriate label. Refer to the DL405 User Manual for details on these instructions. SP52 will be on and the error code will be stored in V7755.
E403 Missing RET (DL440 only)	A subroutine in the program does not end with the RET instruction. SP52 will be on and the error code will be stored in V7755.
E404 Missing FOR (DL440 only)	A NEXT instruction does not have the corresponding FOR instruction. SP52 will be on and the error code will be stored in V7755.

DL405 Error Code	Description
E405 Missing NEXT (DL440 only)	A FOR instruction does not have the corresponding NEXT instruction. SP52 will be on and the error code will be stored in V7755.
E406 Missing IRT	An interrupt routine in the program does not end with the IRT instruction. SP52 will be on and the error code will be stored in V7755.
E412 SBR/LBL>64 (DL440 only)	There is greater than 64 SBR, LBL or DLBL instructions in the program. This error is also returned if there is greater than 128 GTS or GOTO instructions used in the program. SP52 will be on and the error code will be stored in V7755.
E413 FOR/NEXT>64 (DL440 only)	There is greater than 64 FOR/NEXT loops in the application program. SP52 will be on and the error code will be stored in V7755.
E421 Duplicate Stage Reference	Two or more SG or ISG labels exist in the application program with the same number. A unique number must be allowed for each Stage and Initial Stage. SP52 will be on and the error code will be stored in V7755.
E422 Duplicate SBR/LBL Reference (DL440 only)	Two or more SBR or LBL instructions exist in the application program with the same number. A unique number must be allowed for each Subroutine and Label. SP52 will be on and the error code will be stored in V7755.
E423 Nested Loops (DL440 only)	Nested loops (programming one FOR/NEXT loop inside of another) is not allowed in the DL440 series. SP52 will be on and the error code will be stored in V7755.
E431 Invalid ISG/SG Address	An ISG or SG must not be programmed after the end statement such as in a subroutine. SP52 will be on and the error code will be stored in V7755.
E432 Invalid Jump (GOTO) Address (DL440 only)	A LBL that corresponds to a GOTO instruction must not be programmed after the end statement such as in a subroutine. SP52 will be on and the error code will be stored in V7755.
E433 Invalid SBR Address (DL440 only)	A SBR must be programmed after the end statement, not in the main body of the program or in an interrupt routine. SP52 will be on and the error code will be stored in V7755.
E434 Invalid RTC Address (DL440 only)	A RTC must be programmed after the end statement, not in the main body of the program or in an interrupt routine. SP52 will be on and the error code will be stored in V7755.
E440 Invalid RT Address (DL440 only)	A RT must be programmed after the end statement, not in the main body of the program or in an interrupt routine. SP52 will be on and the error code will be stored in V7755.
E436 Invalid INT Address	An INT must be programmed after the end statement, not in the main body of the program. SP52 will be on and the error code will be stored in V7755.
E437 Invalid IRTC Address	An IRTC must be programmed after the end statement, not in the main body of the program. SP52 will be on and the error code will be stored in V7755.

DL405 Error Code	Description
E438 Invalid IRT Address	An IRT must be programmed after the end statement, not in the main body of the program. SP52 will be on and the error code will be stored in V7755.
E440 Invalid Data Address (DL440 only)	Either the DLBL instruction has been programmed in the main program area (not after the END statement), or the DLBL instruction is on a rung containing input contact(s).
E441 ACON/NCON (DL440 only)	An ACON or NCON must be programmed after the end statement, not in the main body of the program. SP52 will be on and the error code will be stored in V7755.
E451 Bad MLS/MLR	MLS instructions must be numbered in ascending order from top to bottom.
E452 X AS Coil	An X data type is being used as a coil output.
E453 Missing T/C	A timer or counter contact is being used where the associated timer or counter does not exist.
E454 Bad TMRA	One of the contacts is missing from a TMRA instruction.
E455 Bad CNT	One of the contacts is missing from a CNT or UDC instruction.
E456 Bad SR	One of the contacts is missing from the SR instruction.
E461 Stack Overflow	More than nine levels of logic have been stored on the stack. Check the use of OR STR and AND STR instructions.
E462 Stack Underflow	An unmatched number of logic levels have been stored on the stack. Insure the number of AND STR and OR STR instructions match the number of STR instructions.
E463 Logic Error	A STR instruction was not used to begin a rung of ladder logic.
E464 Missing CKT	A rung of ladder logic is not terminated properly.
E471 Duplicate Coil Reference	Two or more OUT instructions reference the same I/O point.
E472 Duplicate TMR Reference	Two or more TMR instructions reference the same number.
E473 Duplicate CNT Reference	Two or more CNT instructions reference the same number.
E480 Invalid CV Address (DL440 only)	The CV instruction is used in a subroutine or program interrupt routine. The CV instruction may only be used in the main program area (before the END statement).

DL405 Error Code	Description
E481 Conflicting Instructions (DL440 only)	An instruction exists between convergence stages.
E482 Max. CV Instructions Exceeded (DL440 only)	Number of CV instructions exceeds 17.
E483 Invalid CV Jump Address (DL440 only)	CV JMP has been used in a subroutine or a program interrupt routine.
E484 Missing CV Instruction (DL440 only)	CV JMP is not preceded by the CV instruction. A CV JMP must immediately follow the CV instruction.
E485 Missing required instruction (DL440 only)	A CV JMP instruction is not placed between the CV and the [SG, ISG, ST BLK, END BLK, END] instruction.
E486 Invalid CALL BLK address (DL440 only)	CALL BLK is used in a subroutine or a program interrupt routine. The CALL BLK instruction may only be used in the main program area (before the END statement).
E487 Missing ST BLK Instruction (DL440 only)	The CALL BLK instruction is not followed by a ST BLK instruction.
E488 Invalid ST BLK Address (DL440 only)	The ST BLK instruction is used in a subroutine or a program interrupt. Another ST BLK instruction is used between the CALL BLK and the END BLK instructions.
E489 Duplicated CR Reference (DL440 only)	The control relay used for the ST BLK instruction is being used as an output elsewhere.
E490 Missing SG instruction (DL440 only)	The ST BLK instruction is not immediately followed by the SG instruction.
E491 Invalid ISG Instruction Address (DL440 only)	There is an ISG instruction between the ST BLK and END BLK instructions.

DL405 Error Code	Description
E492 Invalid END BLK Address (DL440 only)	The END BLK instruction is used in a subroutine or a program interrupt routine. The END BLK instruction is not followed by a ST BLK instruction.
E493 Missing Required Instruction (DL440 only)	A [CV, SG, ISG, ST BLK, END] instruction must immediately follow the END BLK instruction.
E494 Missing END BLK Instruction (DL440 only)	The ST BLK instruction is not followed by a END BLK instruction.
E501 Bad Entry	An invalid keystroke or series of keystrokes was entered into the handheld programmer.
E502 Bad Address	An invalid or out of range address was entered into the handheld programmer.
E503 Bad Command	An invalid instruction was entered into the handheld programmer.
E504 Bad Ref/VAL	An invalid value or reference number was entered with an instruction.
E505 Invalid Instruction	An invalid instruction was entered into the handheld programmer.
E506 Invalid Operation	An invalid operation was attempted by the handheld programmer.
E520 Bad Op-RUN	An operation which is invalid in the RUN mode was attempted by the handheld programmer.
E521 Bad OP-TRUN	An operation which is invalid in the TEST RUN mode was attempted by the handheld programmer.
E523 Bad OP-TPGM	An operation which is invalid in the TEST PROGRAM mode was attempted by the handheld programmer.
E524 Bad OP-PGM	An operation which is invalid in the PROGRAM mode was attempted by the handheld programmer.
E525 Keyswitch	An operation was attempted by the handheld programmer while the CPU keyswitch was in a position other than the TERM position.
E526 Off Line	The handheld programmer is in the OFFLINE mode. To change to the ONLINE mode use AUX64.
E540 CPU Locked (DL440 only)	The CPU has been password locked. To unlock the CPU use AUX82 with the password.
E541 Wrong Password (DL440 only)	The password used to unlock the CPU with AUX82 was incorrect.

DL405 Error Code	Description
E542 Password Reset (DL440 only)	The CPU powered up with an invalid password and reset the password to 00000000. A password may be re-entered using AUX81.
E601 Memory Full	Attempted to enter an instruction which required more memory than is available in the CPU.
E602 Instruction Missing	A search function was performed and the instruction was not found.
E603 Data Missing (DL440 only)	A search function was performed and the data was not found.
E604 Reference Missing	A search function was performed and the reference was not found.
E610 Bad I/O Type	The application program has referenced an I/O module as the incorrect type of module.
E620 Out Of Memory	An attempt to transfer more data between the CPU and handheld programmer than the receiving device can hold.
E621 MC Not Blank	An attempt to write to a non-blank Memory Cartridge was made. Erase the cartridge and then retry the write.
E622 No HPP MC	A data transfer was attempted with no Memory Cartridge or possibly a faulty Memory Cartridge in the handheld programmer.
E623 System MC	A function was requested with a Memory Cartridge which contains system information only.
E624 V-memory only	A function was requested with a Memory Cartridge which contains V-memory data only.
E625 Program only	A function was requested with a Memory Cartridge which contains program data only.
E626 PROM MC	An attempt to transfer data from a tape to a UVPROM Memory Cartridge. This transfer must be made using a CMOS RAM Cartridge.
E627 Bad Write	An attempt to write to a write protected or faulty Memory Cartridge was made. Check the write protect jumper inside the cartridge then replace if necessary.
E640 Compare error	A compare between the Memory cartridge and the source data was found to be in error. Erase the Memory Cartridge and retry the operation, replace the Memory Cartridge if necessary.
E641 Volume Level	The volume level of the cassette player is not set properly. Adjust the volume and retry the operation.
E642 Checksum Error	An error was detected while data was being transferred to the handheld programmer's Memory Cartridge. Check cabling and retry the operation.
E650 HPP System Error	A system error has occurred in the handheld programmer. Power cycle the handheld programmer. If the error returns replace the handheld programmer.
E651 HPP ROM Error	A ROM error has occurred in the handheld programmer. Power cycle the handheld programmer. If the error returns replace the handheld programmer.

DL405 Error Code	Description
E652 HPP RAM Error	A RAM error has occurred in the handheld programmer. Power cycle the handheld programmer. If the error returns replace the handheld programmer.
E653 MC Battery Low	The battery in the CMOS RAM cartridge is low and should be replaced.

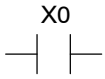
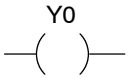
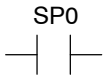
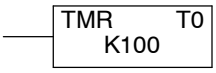
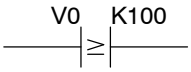
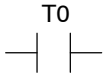
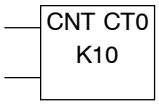
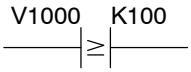
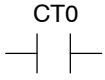
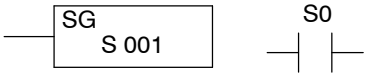
DL405 Memory Map

A

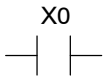
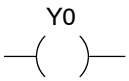
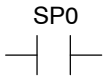
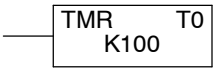
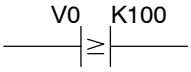
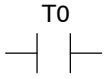
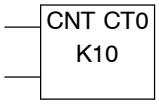
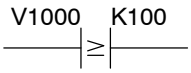
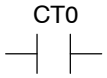
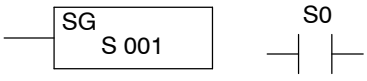
In This Appendix. . . .

Memory Map Overview

DL430 Memory Map

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

DL440 Memory Map

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

X Input Bit Map

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

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

Y Output Bit Map

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

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

Remote I/O Bit Map

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

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

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

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

Control Relay Bit Map

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

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

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

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

Stage Control / Status Bit Map

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

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

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

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

Timer Status Bit Map

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

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

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

Counter Status Bit Map

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

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