# Memory Cartridge/
# Real Time Clock

**2**

---

## In This Chapter. . . .

# General Information about the D0–01MC
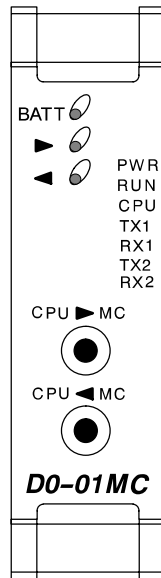
The D0–01MC installs into any of the DL05 PLCs. The MC backs-up the ladder program and data in CMOS RAM. The on–board lithium battery lasts up to three years. The module's V–memory maps one–for–one to the PLC's memory locations.

When the Memory Cartridge is inserted in the option slot, it automatically becomes the source of the controlling program. You may choose to overwrite the PLC program, but you don't have to.

You can transfer the program from the PLC to the MC or from the MC to the PLC or you can operate directly from the MC. By removing the module, you return control to the PLC's internal program.

Two pushbuttons on the face of the module initiate memory transfers. The pushbuttons are clearly marked to indicate the direction of the transfer, and a green LED flashes to confirm the direction and success of the memory transfer.

**Jumper Selects Write Enable or Disable**

A jumper enables/disables the write function in the MC. Write disable prevents overwriting the MC memory. Write enable allows overwriting the MC memory. See page 2–3 for more information.

**Low Battery Alert**

A red LED alerts you to a low battery condition. If the battery drops below 2.5V the "BATT" LED comes on, and an internal bit is set. You can use the internal bit to activate alarm functions or to execute an orderly shut–down. See page 2–10 for more information.

**Clock/Calendar**

The date and time are easily set or accessed using *Direct*SOFT32 Version 3.0b or later, programming software. The "year" field contains four digits so it is ready for Y2K and beyond.

**Specifications**

Environmental specifications for the D0–01MC are the same as for the DL05 PLCs. UL and CE approvals are pending. (Go to **www.automationdirect.com** for the latest UL and CE updates.) See page 2–11 for detailed specifications.

**New Ladder Instructions**

New ladder instructions are available when using the D0–01MC. See page 2–12 for Date and Time instructions. The MOVMC instruction has the ability to use a constant (K value) when used with the Memory Cartridge.

**Error Code Changes**

Two Error Codes have expanded definitions when using the Memory Cartridge. See page 2–17.
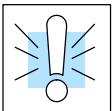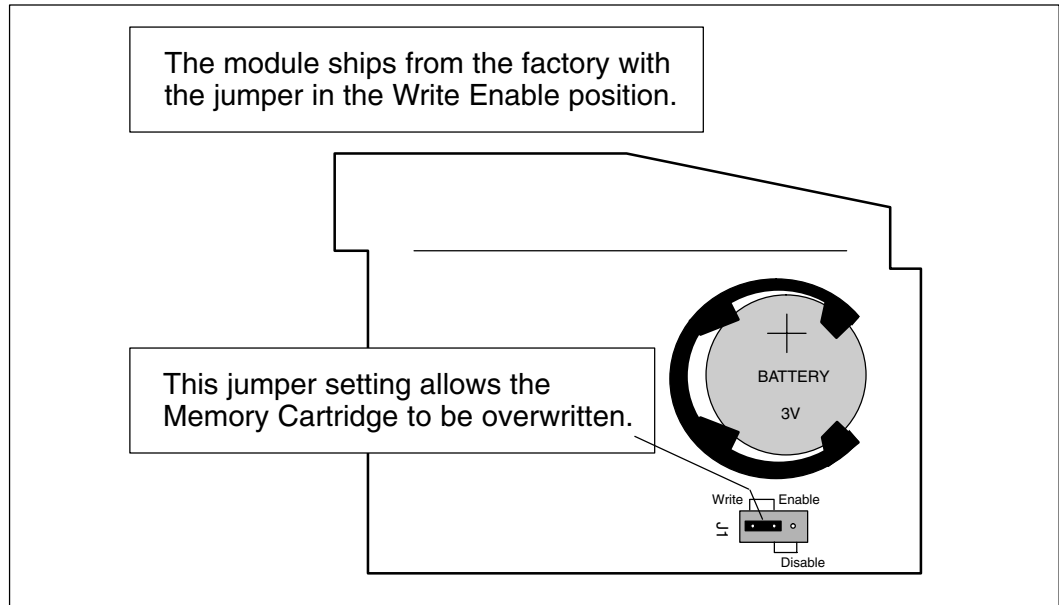
# Setting the Write Enable/Disable Jumper

The position of jumper J1 determines whether or not the Memory Cartridge can be overwritten. The Write Disable position is used only for transporting a program. With J1 in the Write Disable position the Memory Cartridge program can be copied to the CPU, but the PLC cannot be put in RUN mode. The PLC can only be put in the RUN mode with J1 in the Write Enable position.) Write failures will generate the E104 Error Code (for more information see page 2–17).

**Write Enable**    Set the jumper as shown in Figure 1 if you want the CPU to be able to overwrite the Memory Cartridge.

**Figure 1**

The module ships from the factory with the jumper in the Write Enable position.

BATTERY

3V

This jumper setting allows the Memory Cartridge to be overwritten.

Write  Enable

J1

Disable

**WARNING:** Power to the DL05 PLC **must** be disconnected before inserting or removing the D0–01MC Memory Cartridge. Failure to disconnect power could result in serious damage to the module, the PLC or both.

**Write Disable**    Set the jumper as shown in Figure 2 if you **do not** want the CPU or *Direct*SOFT32 to be able to overwrite the Memory Cartridge.

**Figure 2**

BATTERY

3V

This jumper setting prevents the Memory Cartridge from being overwritten.

Write  Enable

J1

Disable
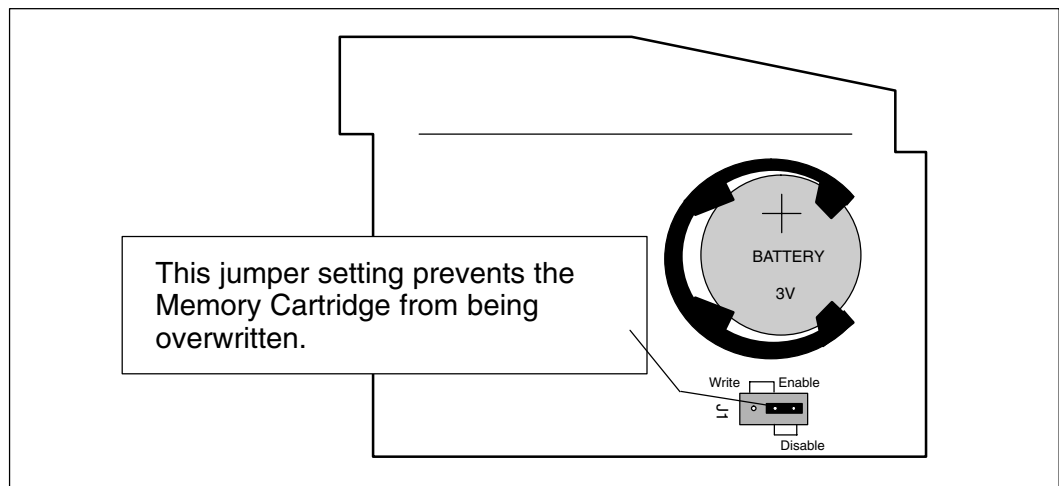
# Software and Firmware Requirements

**How to Update Your *Direct*SOFT32 Programming Software**

You will need ***Direct*SOFT32** Version 3.0b (or later) to use all features of the D0–01MC. If you have a licensed copy of Version 3.0 or 3.0a, the Version 3.0b Maintenance Release (or a later maintenance release) is available for free on our website at **www.automationdirect.com**.

Your DL05 must have Version 2.0, or later, firmware to operate correctly with all features of the D0–01MC. To determine your firmware revision level, connect to the DL05 with ***Direct*SOFT32** programming software, and click on PLC/Diagnostics/System Information. This will bring up the System Information screen. This will bring up the System Information screen.
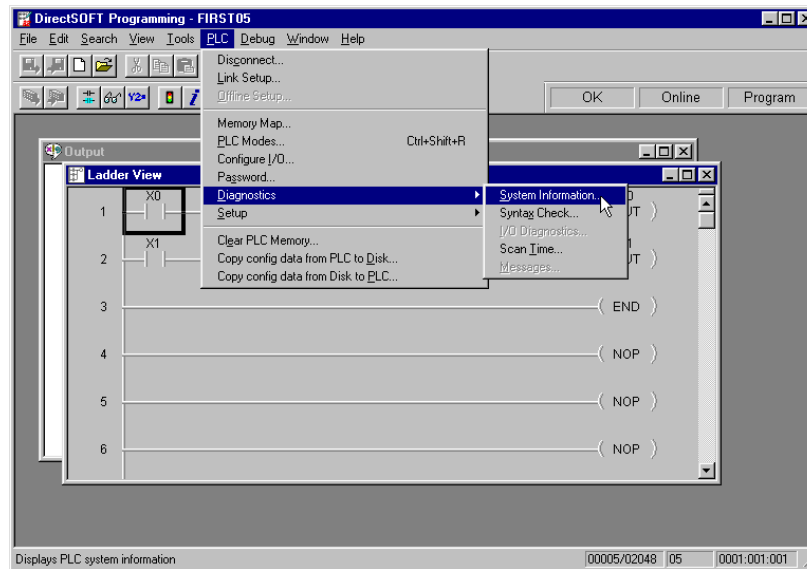


The "CPU Version:" will tell you what firmware revision level is installed in your PLC.
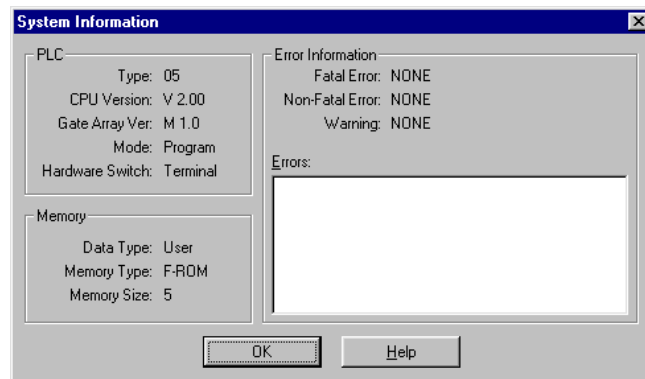


**How to Update Your DL05 Firmware**

If your DL05 requires new firmware, you may download the latest firmware and upgrade tool from our website. Point your browser to **www.automationdirect.com**, and click on **Technical Support**. There you will find the latest firmware for the DL05 by going to **Downloads,** then **Firmware Upgrades**. Be sure to download the **Koyo Upgrade** tool and the **Firmware Binary** file.

Follow the upgrade instructions contained in the downloaded files. Cycle power after upgrading the firmware in your DL05 PLC, and ***Direct*SOFT32** will recognize the new features available to your PLC.

# Naming the Memory Cartridge

*Direct*SOFT32 allows you to name your Memory Cartridge. Use this feature to identify a specific machine, a version number for your ladder logic program, etc.
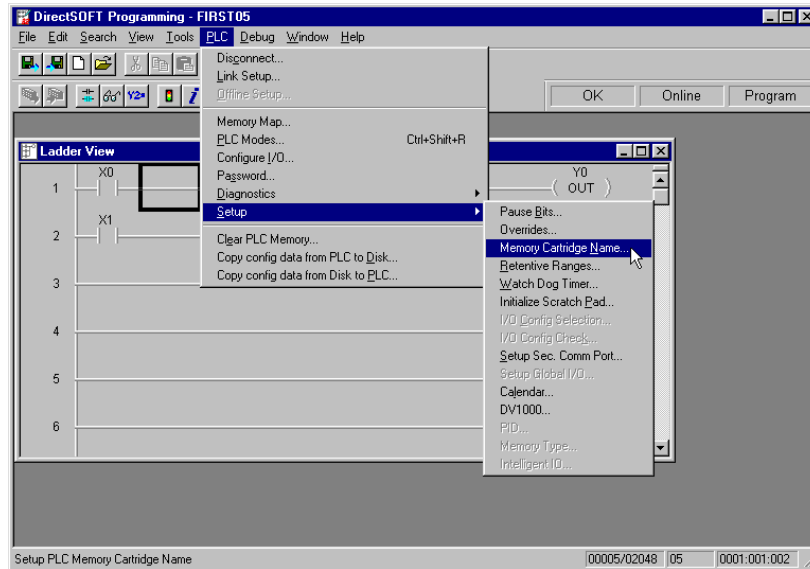
**Up to 8 Alphanumeric Characters**

Within *Direct*SOFT32, click on PLC/Setup/Memory Cartridge Name to create a name for the Memory Cartridge. You can use up to 8 alphanumeric or special characters.
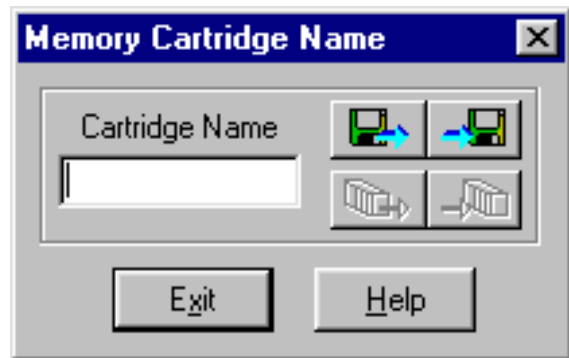
When you type a name in the "Cartridge Name" field, you can save the name to disk by clicking on the button with the arrow pointing to the disk, which will store the name in the *Direct*SOFT32 project folder. You can also read the name from the project folder (if the Memory Cartridge was previously named) by clicking on the button with the arrow coming from the disk.



If the PC is connected to the PLC, the PLC-icon buttons will become active. When the PLC-icon buttons are active, you can also transfer the name to the PLC or transfer an existing name from the PLC into *Direct*SOFT32. Transfering the Memory Cartridge name into the PLC transfers the name directly to the Memory Cartridge.

**Name is Retained in Cartridge Memory and Project Folder**

Naming the Memory Cartridge is independent of the ladder logic program – although the name is stored in the *Direct*SOFT32 project folder. It is also stored in the battery-backed memory on-board the Memory Cartridge. Transferring a program from the PLC does **not** change the name of the Memory Cartridge even if the program in the PLC originated as a program in another Memory Cartridge that has a different name.

If you require the name of your Memory Cartridge to change, you must change it using the screen shown.

# Setting the Time and Date

Use *Direct*SOFT32 to set the date and time in your Memory Cartridge, connect to the DL05. Inside the Project Window, click on PLC/Setup/Calendar to bring up the Calendar window.



**Note:** You can also use ladder instructions to set the date and time. See Clock/Calendar Instructions beginning on page 2–12.
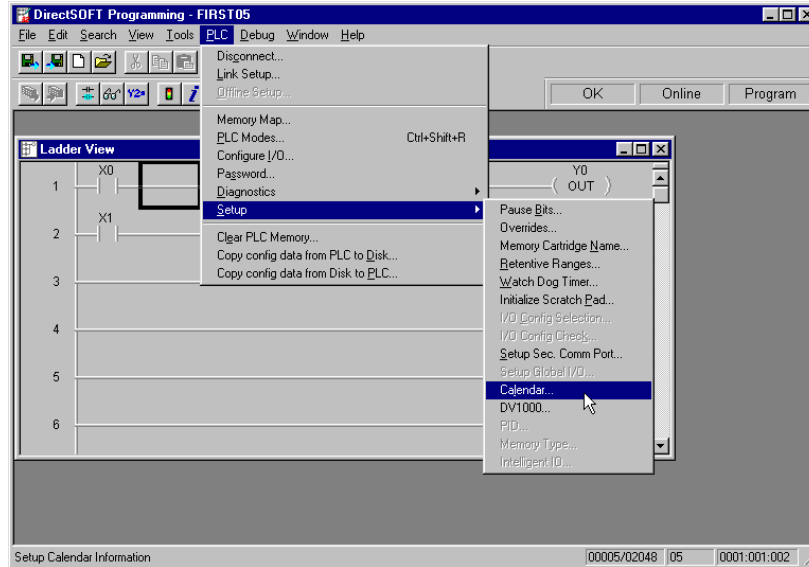
The first time you view the Calendar window, you will see the factory settings for date and time, as shown below. You can either change each field individually, or you can click on the "Get PC Time" button. If you click on the "Get PC Time" button, you will see the values change in all fields. Note, the values will not continue to update. In order to save this new time, you need to click on the "Write to PLC" button.
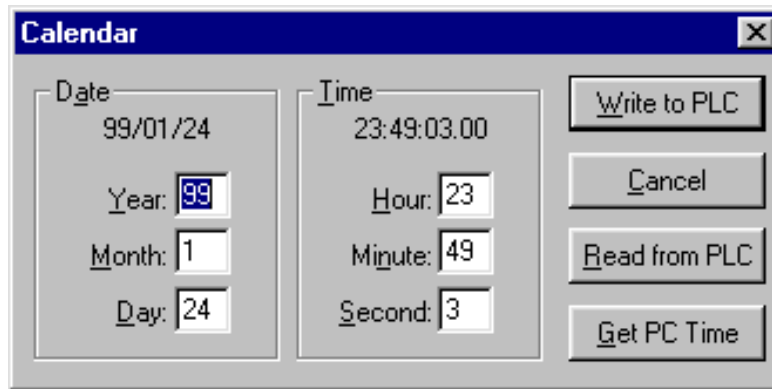


The clock and calendar functions are stored in the following V-memory locations:

| V7766 | Seconds | 2 digits BCD | 00 – 59 |
|-------|---------|--------------|---------|
| V7767 | Minutes | 2 digits BCD | 00 – 59 |
| V7770 | Hours | 2 digits BCD | 00 – 23 |
| V7771 | Day of the Week (00=Sun, 01=Mon, etc.) | 2 digits BCD | 00 – 06 |
| V7772 | Day of the Month | 2 digits BCD | 01 – 31 |
| V7773 | Month | 2 digits BCD | 01 – 12 |
| V7774 | Year | 4 digits BCD | 1970 – 2069 |

# Memory Transfers

**CPU to MC**

With system supply power on and the Memory Cartridge installed, put the PLC in STOP mode. Press and hold the CPU > MC button for about one second, then release. This will transfer the program from the CPU to the Memory Cartridge. To confirm the transfer, the associated LED will blink for a few seconds. If the LED does not blink, the PLC is not in STOP mode. If the LED continues to blink, an error has occurred. To acknowledge the error, press either of the two buttons on the face of the module.
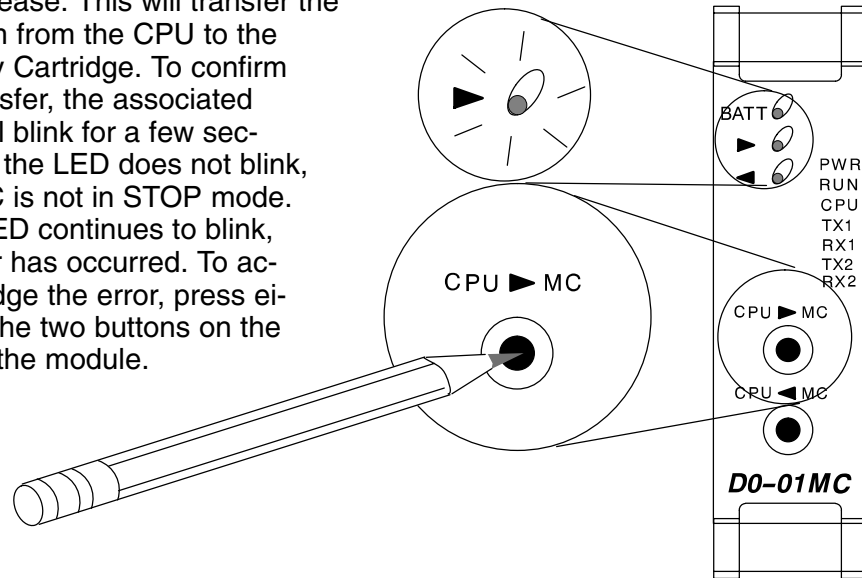


**NOTE:** Memory transfer functions cannot be accessed while the PLC is in RUN mode. Use the toggle switch on the DL05 to switch the PLC out of RUN mode, or use your *Direct*SOFT32 programming software to make the change.

**MC to CPU**

With system supply power on and the Memory Cartridge installed, put the PLC in STOP mode. Press and hold the CPU < MC button for about one second, then release. This will transfer the program from the Memory Cartridge to the CPU. To confirm the transfer, the associated LED will blink for a few seconds. If the LED does not blink, the PLC is not in STOP mode. If the LED continues to blink, an error has occurred.
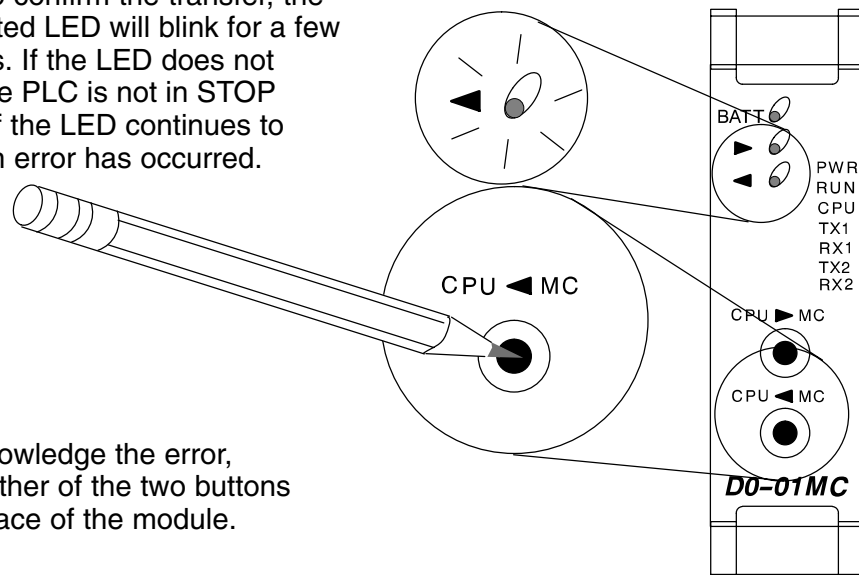
To acknowledge the error, press either of the two buttons on the face of the module.

# LED Indicator Lights

The three indicator lights on the face of the Memory Cartridge perform the following functions:

| LED Indicator | Condition | Meaning |
|---|---|---|
| BATT (red) | On | Replace battery |
| CPU > MC (green) | Flashes for several seconds (150 msec on; 150 msec off) | Successful completion of transfer |
| | Does not flash | No memory transfer; change to STOP mode |
| | Flashes continually with CPU < MC LED steady on | Error; press either button to clear |
| CPU < MC (green) | Flashes for several seconds (150 msec on; 150 msec off) | Successful completion of transfer |
| | Does not flash | No memory transfer; change to STOP mode |
| | Flashes continually with CPU > MC LED steady on | Error; press either button to clear |

# Password Protected Programs

Password protected programs cannot be transferred from the CPU on-board memory to the Memory Cartridge. Also, password protected programs cannot be transferred from the Memory Cartridge to the CPU on-board memory. Programs can only be transferred in either direction when a password is not used.

| PLC Mode | Password | | CPU > MC | CPU < MC |
|---|---|---|---|---|
| | DL05 | D0–01MC | | |
| RUN | No | No | Disable (No LED ON) | Disable (No LED ON) |
| | Yes | No | Disable (No LED ON) | Disable (No LED ON) |
| | No | Yes | Disable (No LED ON) | Disable (No LED ON) |
| | Yes | Yes | Disable (No LED ON) | Disable (No LED ON) |
| Program | No | No | **Enable (> LED blinks)** | **Enable (< LED blinks)** |
| | Yes | No | Disable (< LED keeps blinking) | Disable (< LED keeps blinking) |
| | No | Yes | Disable (> LED keeps blinking) | Disable (> LED keeps blinking) |
| | Yes | Yes | Disable (> & < LED keeps blinking) | Disable (> & < LED keeps blinking) |

# Memory Map and Forwarding Range

The Memory Cartridge's on-board memory maps one-for-one to the DL05 PLC. The memory types represented in the table below have the same designated locations in either the Memory Cartridge or the DL05 PLC.

| Memory Type | Range |
|---|---|
| Program memory | 2K (all) |
| V-memory | V00000 – V00177 (128 words)<br>V00200 – V00777 (384 words)<br>V01000 – V01177 (128 words)<br>V01200 – V07377 (3200 words)<br>V07400 – V07577 (128 words)<br>V07600 – V07777 (128 words)<br>V40400 – V40417 (16 words)<br>V40500 – V40517 (16 words)<br>V40600 – V40637 (32 words)<br>V41000 – V41017 (16 words)<br>V41100 – V41107 (8 words)<br>V41140 – V41147 (8 words)<br>V41200 – V41237 (32 words) |
| System Parameters | 0.5K (all) |

# Battery Back-up During AC Power Loss

In the event of AC power loss to the DL05 PLC, the on-board lithium battery will back-up the program logic and data values for a period up to three years.

**What if the Battery Dies?**

If the Memory Cartridge's battery voltage drops below approximately 2.5VDC, the red "BATT" LED will illuminate on the face of the module. At the same time, the Special Relay SP43 is set to "1." You can use this internal bit as a contact in your ladder program. Use it to trigger an external alarm – indicating that it is time to change the battery.



```
   SP43                        Y0
  ──┤ ├──────────────────────( )──
```
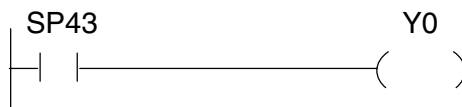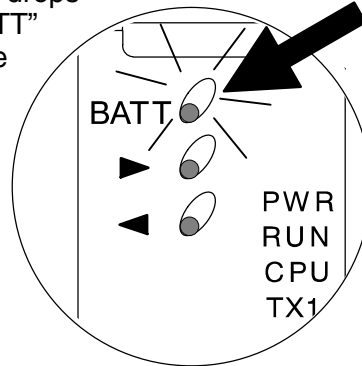
During normal operation, the AC power to the PLC will retain the memory in the Memory Cartridge. If the AC power fails, or is disconnected, a super-capacitor will continue to hold the memory for 4 – 7 days *even if the battery is too weak to hold the memory*. Only after the capacitor has discharged *and* the battery has become too weak to retain the memory will program logic and data memory be lost.

**Battery Type**

When you see the "BATT" indicator light, replace the battery with a CR2032, coin-type 3.0V lithium battery (Automationdirect.com part # D0–BAT). Do you need to change the battery *while retaining the stored program?* If the answer is yes, change the battery during the 4 – 7 days that the super-capacitor will retain the program after AC power is disconnected.

**NOTE:** Be sure your hands are clean and dry before handling the battery. Moisture will corrode the battery surfaces and shorten the life of the battery.

**Removing and Replacing the Battery**
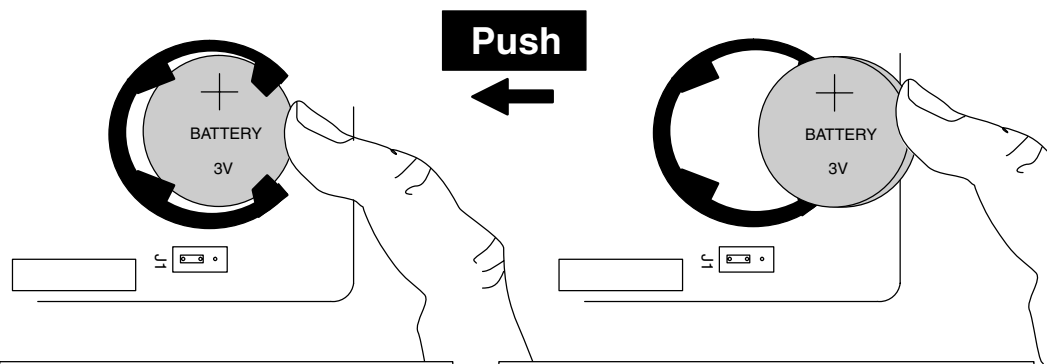


**To remove the battery:**
Push the battery in the direction indicated. You will feel the battery press against the spring in the battery socket. Fully compress the spring and lift the battery out of the socket.

**To replace the battery:**
The positive (+) side should be facing out. The battery should enter the socket as shown (at about a 30° angle to the printed circuit board). Push the battery into the socket until the battery snaps into place.

# Specifications and Agency Approvals

| Specifications | |
|---|---|
| Ambient Operating Temperature | 32˚F to 131˚F (0˚C to 55˚C) |
| Storage Temperature | –4˚F to 158˚F (–20˚C to 70˚C) |
| Ambient Humidity | 5% to 95% non-condensing |
| Atmosphere | No corrosive gases, max. environmental pollution = 2, UL840 |
| Vibration Resistance | MIL STD 810C, method 514.2 |
| Shock Resistance | MIL STD 810C, method 516.2 |
| Noise Immunity | NEMA ICS3–304<br>Impulse noise 1$\mu$s, 1000V<br>FCC Class A<br>RFI (144MHz, 430MHz, 10W, 10cm) |
| Size | 120mm x 95mm x 65mm |
| Weight | 50g |
| Battery Number | CR2032<br>(Automationdirect.com part # D0–BAT) |
| Battery Type | Coin type, 3.0V Lithium Battery, 190mAh |
| Battery Life | Typ 3 years (at 25˚C) |
| Battery Voltage | Typ 3.0VDC |
| Battery Abnormal Voltage | Typ < 2.5VDC<br>Indication: illuminates red LED and sets SP43 to "1" |
| Write Protect | Internal jumper pins |
| Memory Type | CMOS RAM 32KBytes |
| Calendar | Frequency accuracy: ±20PPM (25˚C)<br>Temperature characteristic: +10/–20PPM (0˚ – 55˚C) |
| Agency Approvals | UL, CE (pending) |

Memory Cartridge

# Clock/Calendar Instructions

**Date
(DATE)**

The Date instruction can be used to set the date in the Memory Cartridge. The instruction requires two consecutive V memory locations (Vaaa) to *set the date*. If the values in the specified locations are not valid, the date will not be set. The current date can be *read* from 4 consecutive V memory locations (V7771–V7774).

```
        ┌──────────┐
────────┤ DATE     │
        │   V aaa  │
        └──────────┘
```
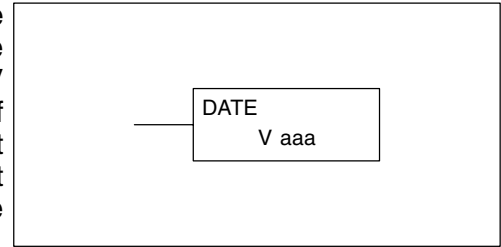
| Date | Range | V Memory Location (BCD) (READ Only) |
|------|-------|--------------------------------------|
| Year | 1970 – 2069 | V7774 |
| Month | 1–12 | V7773 |
| Day of Month | 1–31 | V7772 |
| Day of Week | 0–06 | V7771 |

The values entered for the day of week are:
0=Sunday, 1=Monday, 2=Tuesday, 3=Wednesday, 4=Thursday, 5=Friday, 6=Saturday

| Operand Data Type | | DL05 Range |
|-------------------|--|------------|
| | | **aaa** |
| Vmemory | V | All (See p. 2–9) |

In the following example, when C0 is on, the constant value (K00010301) is loaded into the accumulator using the Load Double instruction (C0 can be a contac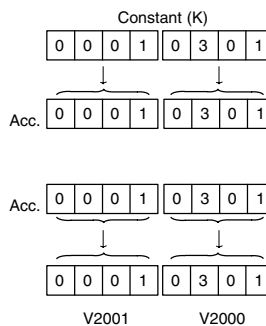t from a one shot (PD) instruction). The value in the accumulator is output to V2000 using the Out Double instruction. The Date instruction uses the value in V2000 to set the date in the CPU. The example loads the date January 3, 2000.

*Direct*SOFT32 Display

```
  C0
──┤ ├──┬──────────────┐
        │ LDD          │
        │    K00010301 │
        └──────────────┘
        Load the constant
        value (K00010301)
        into the accumulator

        ┌──────────────┐
        │ OUTD         │
        │       V2000  │
        └──────────────┘
        Copy the value in
        the accumulator to
        V2000 and V2001

        ┌──────────────┐
        │ DATE         │
        │       V2000  │
        └──────────────┘
        Set the date in the CPU
        using the value in V2000
        and V2001
```

Constant (K)

| 0 | 0 | 0 | 1 | 0 | 3 | 0 | 1 |

Acc. | 0 | 0 | 0 | 1 | 0 | 3 | 0 | 1 |

Acc. | 0 | 0 | 0 | 1 | 0 | 3 | 0 | 1 |

| 0 | 0 | 0 | 1 | 0 | 3 | 0 | 1 |

V2001    V2000

In this example, the Date instruction uses the value set in V2000 and V2001 to set the date in the appropriate V memory locations (V7771–V7774). The year is entered as a two digit number, but it is converted internally to a four digit number. Two digit values ranging between 70 – 99 are converted to 1970 – 1999. Values of 00 – 69 are converted to 2000 – 2069.

V2001        V2000

Format | 0 | 0 | 0 | 1 | 0 | 3 | 0 | 1 |

    Year   Month   Day  Day of Week

Handheld Programmer Keystrokes

| STR | → | C | 0 | ENT |
| SHFT | L | D | D | → | K | 0 | 0 | 0 | 1 |
| 0 | 3 | 0 | 1 | ENT |
| OUT | SHFT | D | → | V | 2 | 0 | 0 | 0 | ENT |
| SHFT | D | A | T | E | → | V | 2 | 0 | 0 | 0 | ENT |
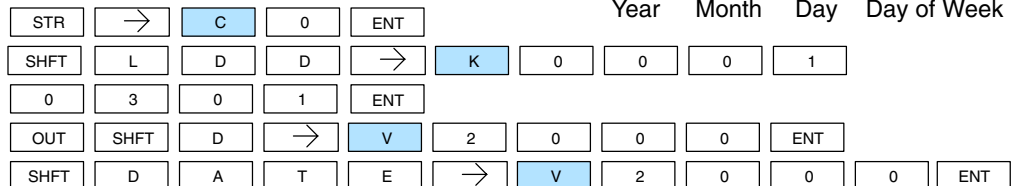
**Time
(TIME)**

The Time instruction can be used to set the time (24 hour clock) in the CPU. The instruction requires two consecutive V memory locations (Vaaa) which are used to *set the time*. If the values in the specified locations are not valid, the time will not be set. The current time can be read from memory locations V7766–V7770.

```
        ┌─────────────┐
────────┤ TIME        │
        │   V aaa     │
        └─────────────┘
```
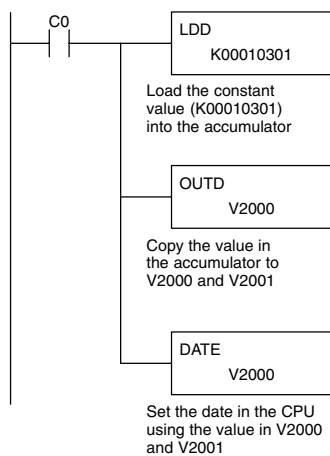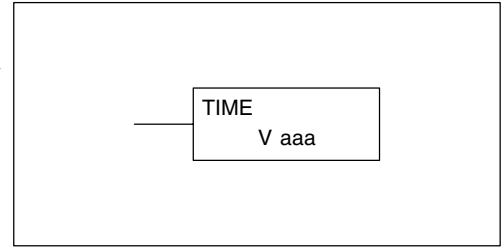
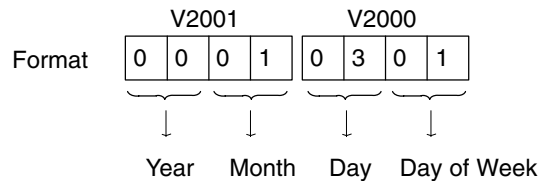| Date | Range | V Memory Location (BCD) (READ Only) |
|------|-------|-------------------------------------|
| Seconds | 0–59 | V7766 |
| Minutes | 0–59 | V7767 |
| Hour | 0–23 | V7770 |

| Operand Data Type | | DL05 Range |
|-------------------|---|------------|
| | | **aaa** |
| Vmemory | V | All (See page 2–9) |

In the following example, when C0 is on, the constant value (K73000) is loaded into the accumulator using the Load Double instruction (C0 should be a contact from a one shot (PD) instruction). The value in the accumulator is output to V2000 using the Out Double instruction. The Time instruction uses the value in V2000 to set the time in the CPU.

*Direct*SOFT32 Display

```
 C0          ┌──────────────┐
──┤ ├────────┤ LDD          │
             │    K73000    │
             └──────────────┘
             Load the constant
             value (K73000) into
             the accumulator

             ┌──────────────┐
             │ OUTD         │
             │    V2000     │
             └──────────────┘
             Copy the value in the
             accumulator to V2000
             and V2001

             ┌──────────────┐
             │ TIME         │
             │    V2000     │
             └──────────────┘
             Set the time in the CPU
             using the value in V2000
             and V2001
```

Constant (K)

| 0 | 0 | 0 | 7 | 3 | 0 | 0 | 0 |

Acc. | 0 | 0 | 0 | 7 | 3 | 0 | 0 | 0 |
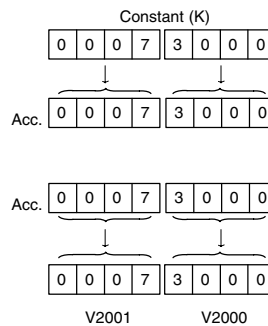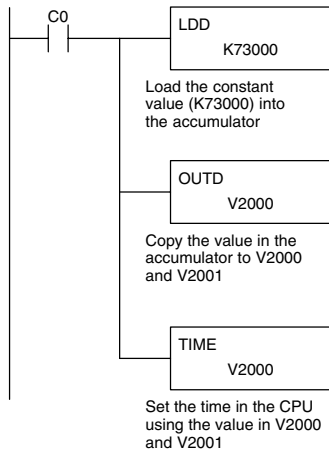
The Time instruction uses the value set in V2000 and V2001 to set the time in the appropriate V memory locations (V7766–V7770)

Acc. | 0 | 0 | 0 | 7 | 3 | 0 | 0 | 0 |

| 0 | 0 | 0 | 7 | 3 | 0 | 0 | 0 |

V2001        V2000

Format

V2001          V2000

| 0 | 0 | 0 | 7 | 3 | 0 | 0 | 0 |

Not Used | Hour | Minutes | Seconds

Handheld Programmer Keystrokes

| STR | → | C | 0 | ENT |

| SHFT | L | D | D | → | K | 7 | 3 | 0 | 0 | 0 | ENT |

| OUT | SHFT | D | → | V | 2 | 0 | 0 | 0 | ENT |

| SHFT | T | I | M | E | → | V | 2 | 0 | 0 | 0 | ENT |
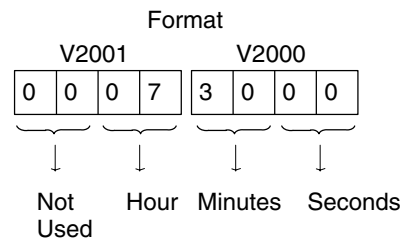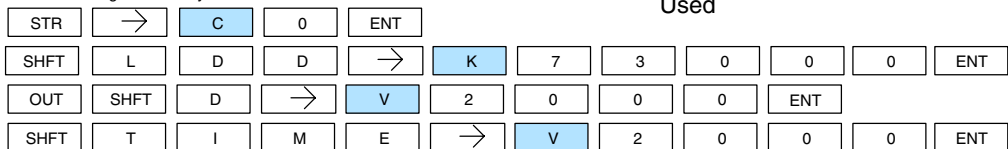
**Move Memory Cartridge / Load Label (MOVMC) (LDLBL)**

The Move Memory Cartridge instruction is used to copy data between V memory and program ladder memory. The Load Label instruction is *only* used with the MOVMC instruction when copying data *from* program ladder memory *to* V memory.

To copy data between V memory and program ladder memory, the function parameters are loaded into the first two levels of the accumulator stack and the accumulator by two additional instructions. Listed below are the steps necessary to program the Move Memory Cartridge and Load Label functions.

```
┌─────────────┐
│  MOVMC      │
│    A aaa    │
└─────────────┘
```

```
┌─────────────┐
│  LDLBL      │
│    K aaa    │
└─────────────┘
```

Step 1:— Load the number of words to be copied into the second level of the accumulator stack.

Step 2:— Load the offset for the data label area in the program ladder memory and the beginning of the V memory block into the first level of the accumulator stack.

Step 3:— Load the *source data label* (LDLBL Kaaa) into the accumulator when copying data from ladder memory to V memory. Load the *source address* into the accumulator when copying data from V memory to ladder memory. This is where the value will be copied from. If the source address is a V memory location, the value must be entered in HEX.

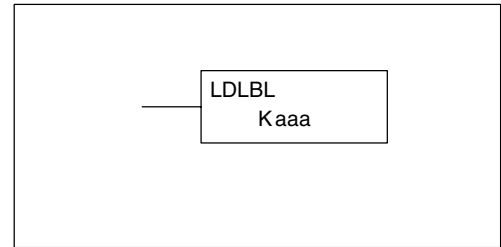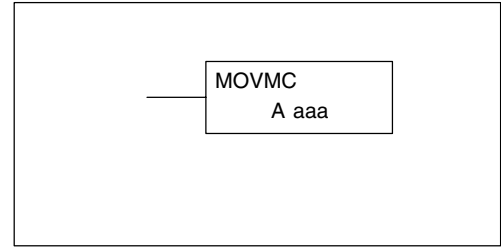Step 4:— Insert the MOVMC instruction which specifies destination (Aaaa). This is where the value will be copied to.

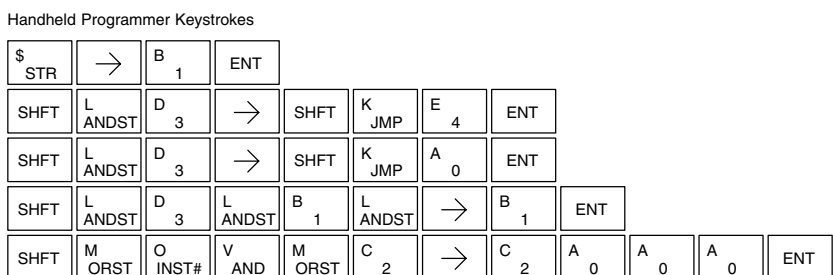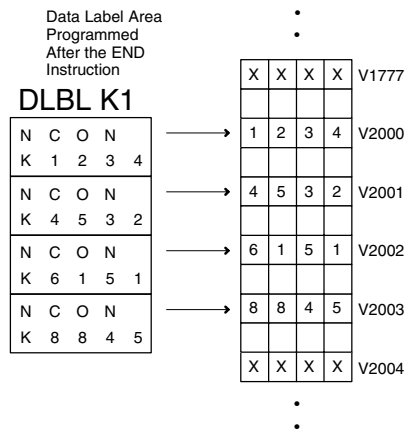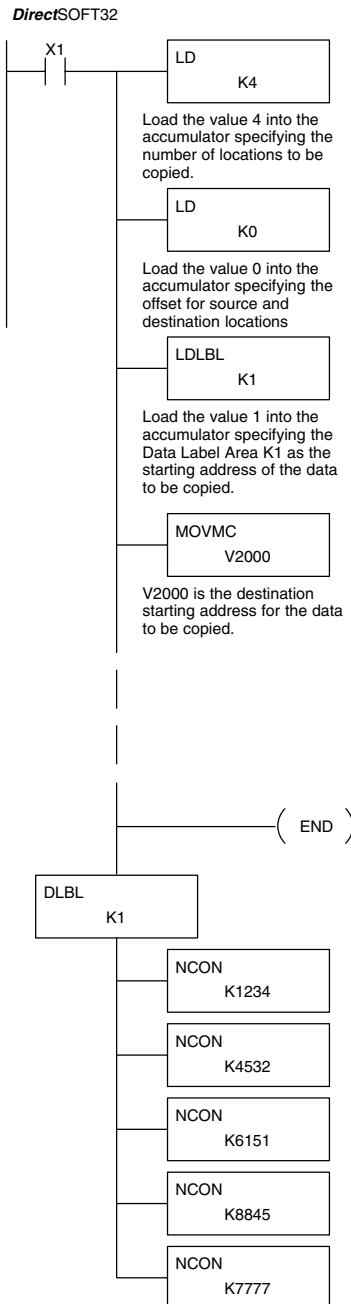| Operand Data Type | | DL05 Range |
|---|---|---|
| | **A** | **aaa** |
| V memory | V | All (See page 2–9) |
| Constant | K | 1–FFFF |

**NOTE:** The MOVMC instruction supports K values only when the D0–01MC is installed. *Without the D0–01MC installed*, the MOVMC instruction supports V memory values only.

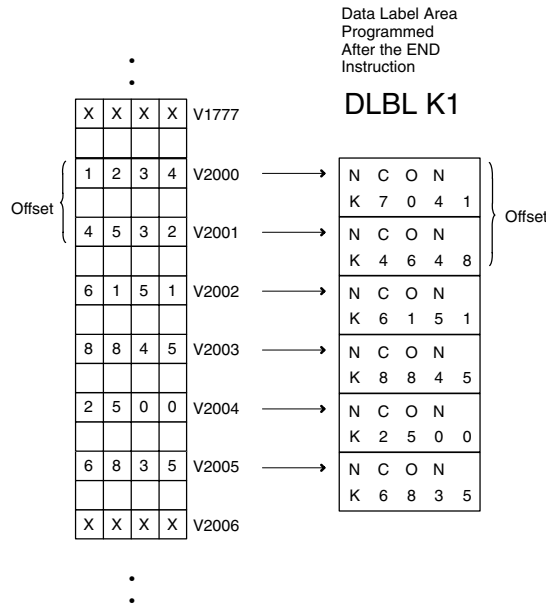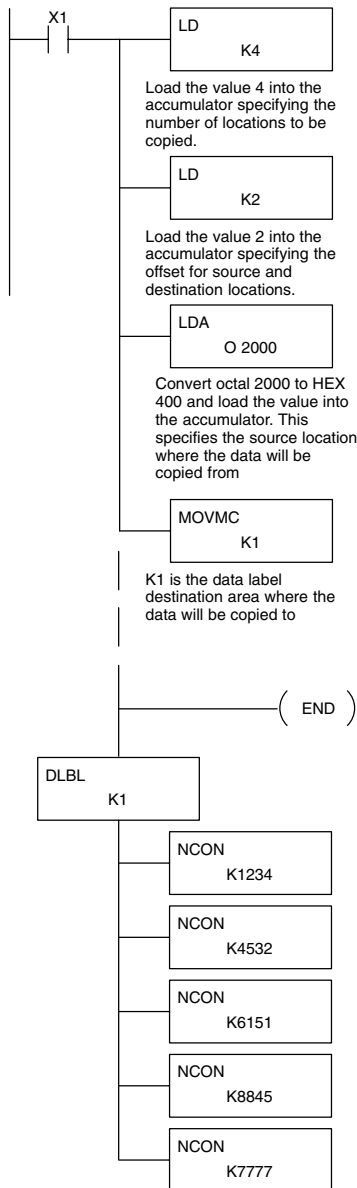| Discrete Bit Flags | Description |
|---|---|
| SP53 | On if there is a table pointer error |

**Copy Data From a Data Label Area to V Memory**

In the following example, data is copied from a Data Label Area to V memory. When X1 is on, the constant value (K4) is loaded into the accumulator using the Load instruction. This value specifies the length of the table and is placed in the second stack location after the next Load and Load Label (LDLBL) instructions are executed. The constant value (K0) is loaded into the accumulator using the Load instruction. This value specifies the offset for the source and destination data, and is placed in the first stack location after the LDLBL instruction is executed. The source address where data is being copied from is loaded into the accumulator using the LDLBL instruction. The MOVMC instruction specifies the destination starting location and executes the copying of data from the Data Label Area to V memory.

*Direct*SOFT32

```
    X1
 ┤  ├───────┌──────────┐
            │ LD       │
            │       K4 │
            └──────────┘
```
Load the value 4 into the accumulator specifying the number of locations to be copied.

```
            ┌──────────┐
            │ LD       │
            │       K0 │
            └──────────┘
```
Load the value 0 into the accumulator specifying the offset for source and destination locations

```
            ┌──────────┐
            │ LDLBL    │
            │       K1 │
            └──────────┘
```
Load the value 1 into the accumulator specifying the Data Label Area K1 as the starting address of the data to be copied.

```
            ┌──────────┐
            │ MOVMC    │
            │    V2000 │
            └──────────┘
```
V2000 is the destination starting address for the data to be copied.

──────────────────( END )

```
┌──────────┐
│ DLBL     │
│       K1 │
└──────────┘
            ┌──────────┐
            │ NCON     │
            │    K1234 │
            └──────────┘
            ┌──────────┐
            │ NCON     │
            │    K4532 │
            └──────────┘
            ┌──────────┐
            │ NCON     │
            │    K6151 │
            └──────────┘
            ┌──────────┐
            │ NCON     │
            │    K8845 │
            └──────────┘
            ┌──────────┐
            │ NCON     │
            │    K7777 │
            └──────────┘
```

Data Label Area Programmed After the END Instruction

DLBL K1

| N C O N | → | | 1 2 3 4 | V2000 |
| K 1 2 3 4 | | | | |

| X X X X | V1777 |

| N C O N | → | 4 5 3 2 | V2001 |
| K 4 5 3 2 | | | |

| N C O N | → | 6 1 5 1 | V2002 |
| K 6 1 5 1 | | | |

| N C O N | → | 8 8 4 5 | V2003 |
| K 8 8 4 5 | | | |

| X X X X | V2004 |

Handheld Programmer Keystrokes

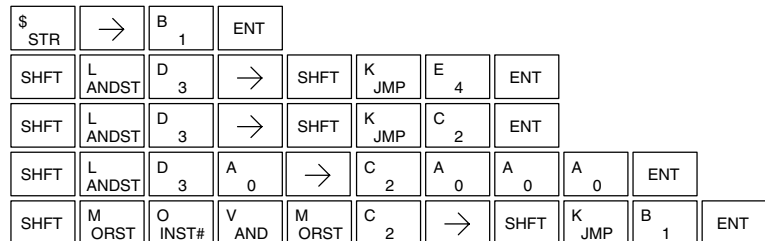| $ STR | → | B 1 | ENT | | | | | | | |
| SHFT | L ANDST | D 3 | → | SHFT | K JMP | E 4 | ENT | | | |
| SHFT | L ANDST | D 3 | → | SHFT | K JMP | A 0 | ENT | | | |
| SHFT | L ANDST | D 3 | L ANDST | B 1 | L ANDST | → | B 1 | ENT | | |
| SHFT | M ORST | O INST# | V AND | M ORST | C 2 | → | C 2 | A 0 | A 0 | A 0 | ENT |

**Copy Data From V Memory to a Data Label Area**

In the following example, data is copied from V memory to a data label area. When X1 is on, the constant value (K4) is loaded into the accumulator using the Load instruction. This value specifies the length of the table and is placed in the second stack location after the next Load and Load Address instructions are executed. The constant value (K2) is loaded into the accumulator using the Load instruction. This value specifies the offset for the source *and* destination data, and is placed in the first stack location after the Load Address instruction is executed. The source address where data is being copied from is loaded into the accumulator using the Load Address instruction. The MOVMC instruction specifies the destination starting location and executes the copying of data from V memory to the data label area.



*Direct*SOFT

X1 LD K4
Load the value 4 into the accumulator specifying the number of locations to be copied.

LD K2
Load the value 2 into the accumulator specifying the offset for source and destination locations.

LDA O 2000
Convert octal 2000 to HEX 400 and load the value into the accumulator. This specifies the source location where the data will be copied from

MOVMC K1
K1 is the data label destination area where the data will be copied to

( END )

DLBL K1
NCON K1234
NCON K4532
NCON K6151
NCON K8845
NCON K7777

Data Label Area Programmed After the END Instruction

DLBL K1

Handheld Programmer Keystrokes

# Error Codes

The full list of Error Codes associated with the DL05 PLC is contained in the DL05 User Manual. The following error codes have different or expanded definitions when associated with the D0–01MC Memory Cartridge. Also see page 2–3 concerning error messages you may encounter when the Memory Cartridge is in the Write Disable position.

| DL05 Error Code | Description |
| --- | --- |
| **E104** WRITE FAILED | A write to the CPU was not successful. Disconnect the power, remove the Memory Cartridge, and make sure the Memory Cartridge is not Write Disabled. |
| **E505** INVALID INSTRUCTION | An invalid instruction was entered into the handheld programmer or a program which includes an instruction which requires the Memory Cartridge (D0–01MC) was run in a DL05 PLC without a Memory Cartridge. |

Memory Cartridge