

# Getting Started

---

## In This Chapter. . . .

- Introduction
  - How can I use the Handheld?
  - Physical Characteristics and Specifications
  - Handheld Basics
  - CPU Setup
-

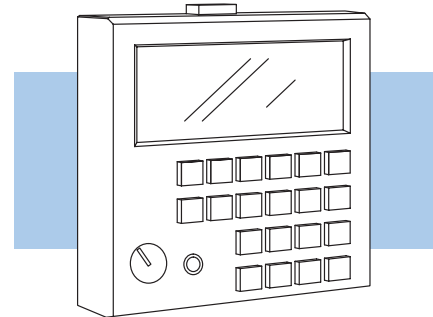
## Introduction

### DL305 Handheld Programmer

The DL305 Handheld Programmer is a general purpose programming tool for use with the DL305 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 **DirectSOFT**, our PC-based programming software.

DL305 Handheld



There are two versions of Handheld Programmers available.

- D3-HP — RLL version for all RLL CPUs
- D3-HPP — RLL<sup>PLUS</sup> version for all RLL<sup>PLUS</sup> CPUs

RLL<sup>PLUS</sup> is just like normal RLL, but a few instructions have been added that make it much easier to use and understand. Programs are usually much shorter and considerably easier to troubleshoot. The best thing to do right now is to make sure you have the correct version of Handheld for use with your CPU. (Trust me, it's easier this way.)

### 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 DL305 User Manual to determine the instruction operation details and keystroke requirements 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 DL305 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 DL305 instruction set or system operation, but rather a user reference manual for the Handheld Programmer.

**How this manual is organized**

**Ch 1: Getting Started** – this chapter provides an overview of the Handheld Programmer, general specifications, and the basic things you need to start entering programs.

**Ch 2: Entering RLL Programs** – discusses all the operations used to enter a program.

**Ch 3: Entering RLL<sup>PLUS</sup> Programs** – provides the keystrokes needed to enter RLL<sup>PLUS</sup> programs.

**Ch 4: Changing Programs** – shows you how to quickly edit an existing program.

**Ch 5: Protecting and Storing Programs** – shows you how to store programs on cassette tapes.

**Ch 6: System Monitoring and Troubleshooting** – provides an overview of the various features used to monitor and troubleshoot the system.

**Appendix A: DL305 Memory Map** – provides a detailed listing of the DL305 memory map for I/O, timers, counters, etc.

**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.

- DL305 User Manual (D3-USER-M)

Now, you know what material is necessary to quickly understand the DL305 Handheld Programmer. So, let's get started!

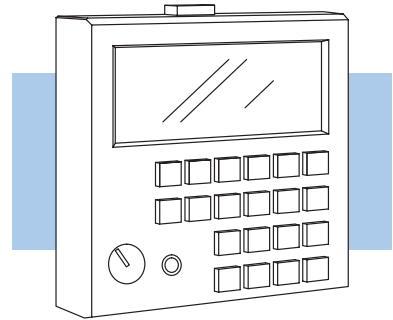
## How can I use the Handheld?

### As a Programming Tool

The DL305 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 **DirectSOFT** Programming Software.

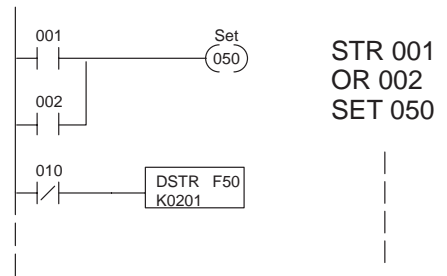
In addition to entering programs, the Handheld is ideal for making on-site program or system changes.

Since the Handheld has a built-in cassette tape interface, you can also use it to store and load programs from cassette tapes.



#### DirectSOFT

#### Handheld



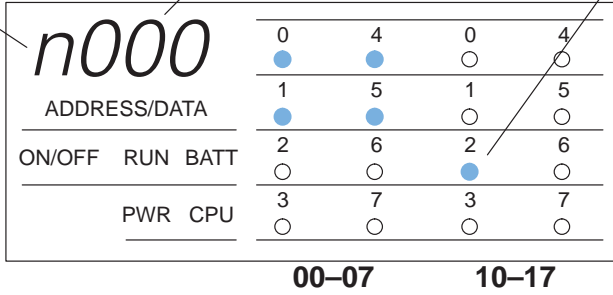
To Monitor Machine Operations

The Handheld is especially useful if you need to quickly look at the status of an I/O point, timer/counter value, or register location. You can monitor up to 16 I/O points at one time. For example, the following diagram shows how the Handheld display area indicates I/O status.

Indicates I/O Status Display

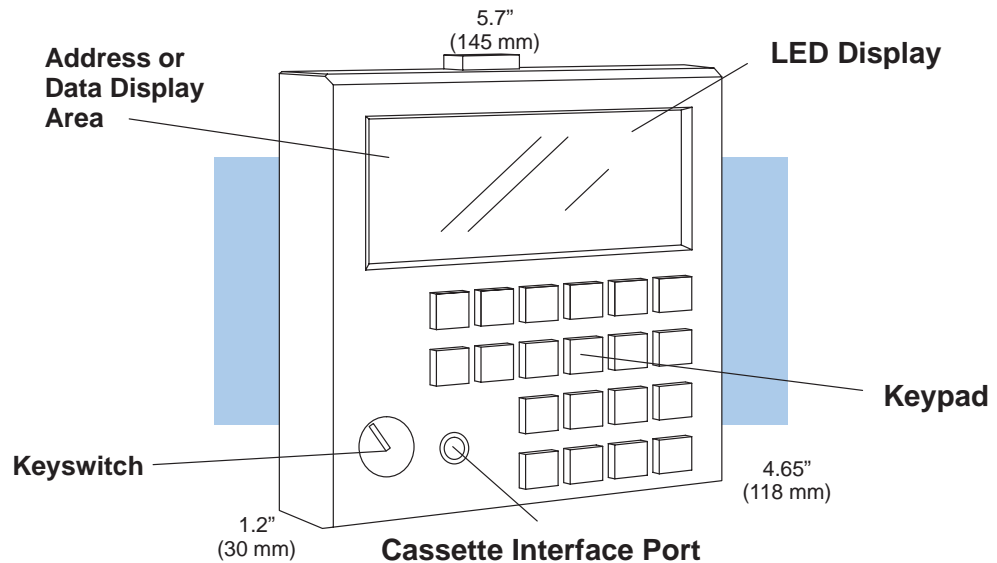
Reference Number

16 LEDs Total show on/off status



## Physical Characteristics and Specifications

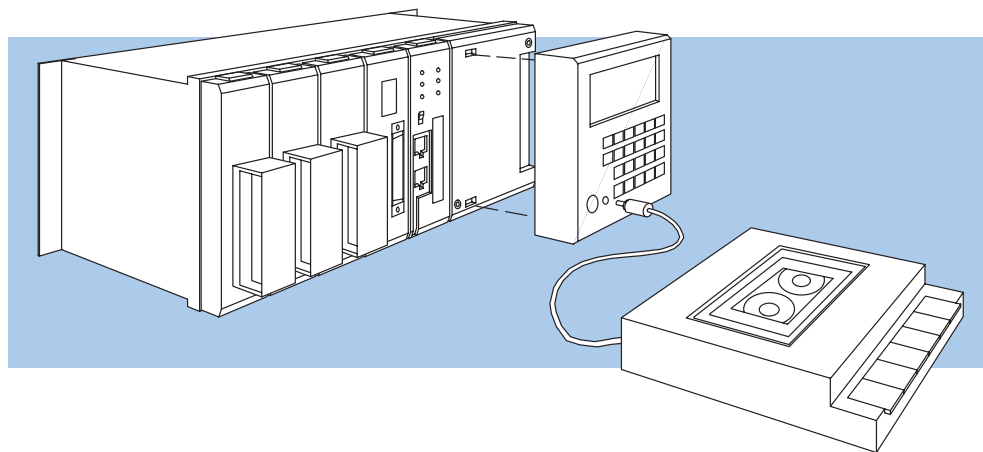
**Handheld Layout** The Handheld was designed to be more than a program entry tool and includes many different status LEDs that make it easy to understand the machine operations.



### Connection Options

You can mount the Handheld directly to the CPU, or you can use a cable. The cable, part number D3-HPCBL-1, is approximately 4.6 feet (1.5m) in length and provides much more flexibility.

A cassette interface cable, supplied with the Handheld Programmer, is required to connect a cassette recorder.



**WARNING:** The CPU will automatically change modes when you connect the Handheld Programmer if the keyswitch is set for a different mode of operation. For example, if the CPU is in Run mode and the Handheld Programmer keyswitch is set to the PRG (Program) position, the CPU will automatically enter Program mode when the Handheld is connected.

**Specifications**

The following table provides specifications for the DL305 Handheld Programmer.

<p><b>Environmental</b></p> <p>Operating Temperature ..... 32° to 140 F° (0° to 60 C°)</p> <p>Storage Temperature ..... -4° to 176 F° (-20° to 80 C°)</p> <p>Humidity ..... 5 to 95% (non-condensing)</p> <p>Environmental Air ..... No corrosive gases</p> <p>Vibration ..... MIL STD 810C 514.2</p> <p>Shock Resistance ..... MIL STD 810C 516.2</p> <p>Noise Immunity ..... NEMA ICS3-304, impulse 1KV, 1µs</p> <p>Power ..... Obtained through PLC port, 60 mA @ 5 VDC 60 mA @ 9 VDC</p> <p>Dimensions ..... 4.3" L x 4.7" H x 0.9" D 110mm W x 118mm H x 24mm D</p> <p>Weight ..... 7.5 oz. (210 g)</p>	
<p><b>CPUs Supported</b></p> <p>DL330, DL330P, DL340</p> <p>Simatic® TI315™, TI325™, TI330™, TI335™, plus stage versions*</p> <p>Texas Instruments® TI315™, TI325™, TI330™, TI335™, plus stage versions*</p> <p>* Stage versions require the D3-HPP</p>	<p><b>Programming Operations</b></p> <p>Read, Write, or erase programs</p> <p>Insert or delete an instruction</p> <p>Search for a specific instruction</p> <p>Locate a specific address</p> <p>Read or write to cassette tapes</p>
<p><b>Cables</b></p> <p>D3-HPCBL-1, 1.5m programmer cable</p>	<p><b>Machine Monitoring Operations</b></p> <p>I/O status (up to 16 simultaneously)</p> <p>On / Off status for contacts, coils, control relays, and register locations</p> <p>Timer and counter current values</p>
	<p><b>Debugging Operations</b></p> <p>Forcing (one scan only)</p> <p>Run and Program Mode display</p> <p>Program syntax check</p> <p>Predefined error codes</p>

## Handheld Basics

### Status LEDs and Key Groups

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 LED display areas and key areas that make this task easier.

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 (SHF) key is pressed. (These keys work just like the number keys on a computer keyboard.)

The keys and LEDs areas are as follows.

- Instruction identifier and numeric keys — used to select the type of instruction. Also used to enter numeric values for instruction references and constants (by pressing SHF first).
- Editing keys — used during program entry and editing to scroll through addresses, insert and delete instructions, etc. These same keys also have Shift functions that are primarily used during cassette or machine monitoring operations.
- Address / Data Display — this 4-character, seven-segment display shows the address, reference number (such as the I/O point being used with an instruction), or data value (such as the current value for a timer.)
- Instruction LEDs — show the type of instruction used at the address being displayed.
- CPU Status LEDs — show the status for Power, CPU mode, etc.

### RLL<sup>PLUS</sup> vs. RLL Units

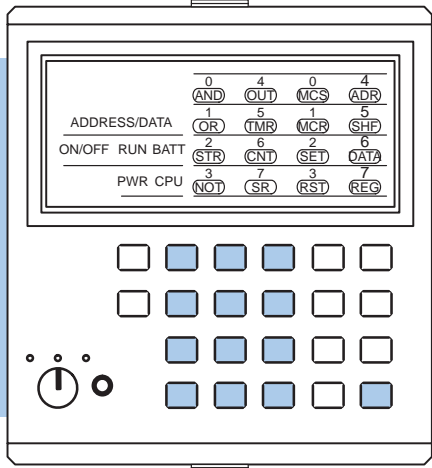
As mentioned earlier there are differences between the two models of DL305 Handheld Programmers. This difference is clearly visible if you examine the keypad and display layout shown on the following page. This manual uses the DL3-HP (RLL version) for most of the examples. This is because, for most instructions, the only difference between the two versions is the key location or the location of the display LED. The key titles are the same, they're just in different locations.

You may have noticed we said the two versions are the same for *most* instructions. The RLL<sup>PLUS</sup> version does have keys for the Master Control Relay Set (MCS), Master Control Relay Reset (MCR), and Shift Register (SR) instructions, but instead has keys for the extra instructions required for RLL<sup>PLUS</sup> programs.

The following diagram shows the key areas, LED areas, and differences between the two types of units.



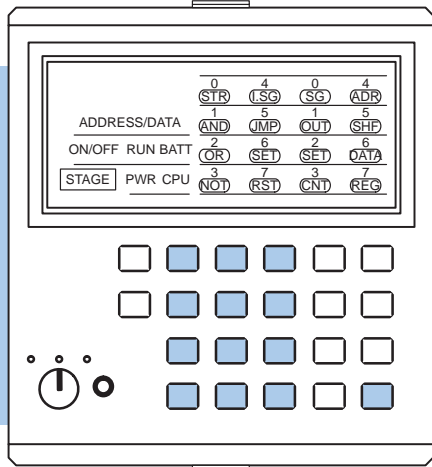
RLL Version



	0	4	0	4
	AND	OUT	MCS	ADR
ADDRESS/DATA	1	5	1	5
	OR	TMR	MCR	SHF
ON/OFF RUN BATT	2	6	2	6
	STR	CNT	SET	DATA
PWR CPU	3	7	3	7
	NOT	SR	RST	REG

F	7	8	9	DEL	SHF
	AND	OUT	MCS		
R	4	5	6	INS	CHECK
	OR	TMR	MCR		SCH
	1	2	3	ENT	READ
	STR	CNT	SET		PRV
	0	.	MON	CLR	WRITE
	NOT	SR	RST		NXT

RLL PLUS Version



	0	4	0	4
	STR	I.SG	SG	ADR
ADDRESS/DATA	1	5	1	5
	AND	JMP	OUT	SHF
ON/OFF RUN BATT	2	6	2	6
	OR	SET	TMR	DATA
STAGE PWR CPU	3	7	3	7
	NOT	RST	CNT	REG

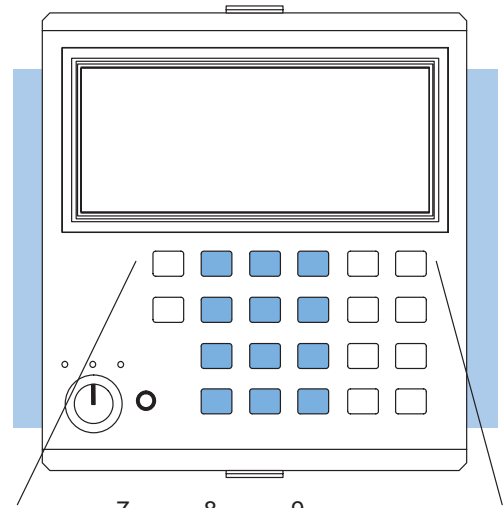
F	7	8	9	DEL	SHF
	STR	I.SG	SG		
R	4	5	6	INS	CHECK
	AND	JMP	OUT		SCH
	1	2	3	ENT	READ
	OR	SET	TMR		PRV
	0	.	MON	CLR	WRITE
	NOT	RST	CNT		NXT

### Instruction Identifier and Numeric Keys

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

For example, to enter Store I/O point 001, you would press STR, SHF, 1 and ENT.

A timer instruction would work the same way. In this case, you would press TMR, SHF, 6, 0, 0, ENT to load the timer. To enter the constant, you would then press SHF, 2, 0, ENT. (This would load a preset of 20.)

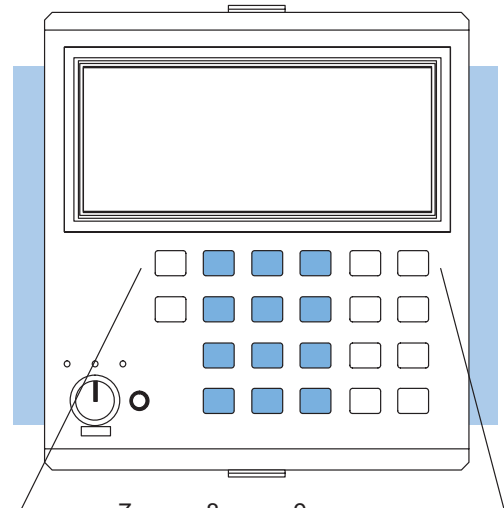


	7	8	9		
F	AND ⊕	OUT	MCS	DEL	SHF
	4	5	6		CHECK
R	OR ⊔	TMR	MCR	INS	SCH
	1	2	3		READ
	STR ⊕	CNT	SET	ENT	PRV
	0	.	MON		WRITE
	NOT ⊖	SR	RST	CLR	NXT

### Editing Keys

These keys are used to perform various operations during program entry and editing. For example, you can use these keys to insert (INS), delete (DEL), or search (SCH) for a specific instruction.

These keys also have shift functions that are primarily used during cassette tape operations. However, there is one key, Monitor (MON), that is used when you want to monitor the status of an I/O point, timer/counter value, or register location.



	7	8	9		
F	STR	I.SG	SG	DEL	SHF
	4	5	6		CHECK
R	AND	JMP	OUT	INS	SCH
	1	2	3		READ
	OR	SET	TMR	ENT	PRV
	0	.	MON		WRITE
	NOT	RST	CNT	CLR	NXT

### Address / Data Display Area

The Address / Data primarily shows two things.

- For programs, it shows address locations or instruction reference numbers.
- For monitoring operations, it shows the current value of timers, counters, and registers.

How do you know which one you're seeing on the display? Simple, whenever an address is shown there are periods that follow each digit. If the periods are missing, you're seeing a data value, a constant, an I/O reference, etc.

#### Data Values or Address

#### Periods indicate an Address

0.0.0.0.		0	4	0	4
ADDRESS/DATA		(AND)	(OUT)	(MCS)	(ADR)
ON/OFF RUN BATT		1	5	1	5
		(OR)	(TMR)	(MCR)	(SHF)
PWR CPU		2	6	2	6
		(STR)	(CNT)	(SET)	(DATA)
		3	7	3	7
		(NOT)	(SR)	(RST)	(REG)

Since the display area can show two types of information, you can easily switch between the two types by pressing CLR and NXT. The following display shows an example of the display if a SET instruction was loaded at address 0000 and you pressed the CLR key. (You can then press NXT to return to the address display.)

#### Shows I/O Reference

#### Periods are missing

0050		0	4	0	4
ADDRESS/DATA		(AND)	(OUT)	(MCS)	(ADR)
ON/OFF RUN BATT		1	5	1	5
		(OR)	(TMR)	(MCR)	(SHF)
PWR CPU		2	6	2	6
		(STR)	(CNT)	(SET)	(DATA)
		3	7	3	7
		(NOT)	(SR)	(RST)	(REG)

### Instruction LEDs

The instruction LEDs show you which instruction is used at the address being displayed. For example, if a SET instruction is located at address 0000, then the SET LED would be on.

#### Shows I/O Reference

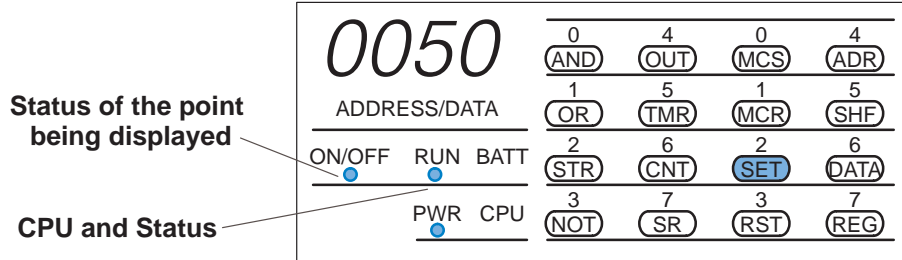
#### Instruction Type

0050		0	4	0	4
ADDRESS/DATA		(AND)	(OUT)	(MCS)	(ADR)
ON/OFF RUN BATT		1	5	1	5
		(OR)	(TMR)	(MCR)	(SHF)
PWR CPU		2	6	2	6
		(STR)	(CNT)	(SET)	(DATA)
		3	7	3	7
		(NOT)	(SR)	(RST)	(REG)

These LEDs are also used during monitoring operations to show the On/Off status for up to 16 points. (We'll discuss this in more detail in Chapter 6.)

### CPU Status LEDs

The CPU LEDs show you the mode of operation, battery status, power indication, and CPU error condition (if any exists). The ON/OFF LED shows the status for the individual instructions as you step through the program during Run mode. For example, if the instruction was SET 050 and the CPU was in Run mode, then the display would appear as follows.



(In Run mode, you can also toggle between the address display and the status display by pressing CLR and NXT. Remember, the address display would have the periods as shown earlier.)

### Clearing the Display Area

Sometimes we all make mistakes, so it's important to know how to clear the display and start from the beginning. Since the Handheld Programmer buffers the keystrokes until you press ENT, you can clear the display at any time up until the ENT button is pressed. When you press CLR, the Handheld clears the keystrokes you've entered and remains at the current address. At this point you can now enter the correct instruction.

Consider the following example that starts at address 0001.

#### Keystroke Error (should have used SET instead of OUT)

OUT    SHF    3    0  
           

<b>030</b>			
0	4	0	4
AND	OUT	MCS	ADR
1	5	1	5
OR	TMR	MCR	SHF
2	6	2	6
STR	CNT	SET	DATA
3	7	3	7
NOT	SR	RST	REG

#### Press CLR

CLR

display returns to address 0001.

<b>0.0.0.1.</b>			
0	4	0	4
AND	OUT	MCS	ADR
1	5	1	5
OR	TMR	MCR	SHF
2	6	2	6
STR	CNT	SET	DATA
3	7	3	7
NOT	SR	RST	REG

## CPU Setup

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 and clearing the CPU memory.

### Changing the CPU Modes

There are two modes available with the DL305 CPUs.

- RUN — executes program and updates I/O modules
- PGM — allows program entry, does not execute program or update I/O modules

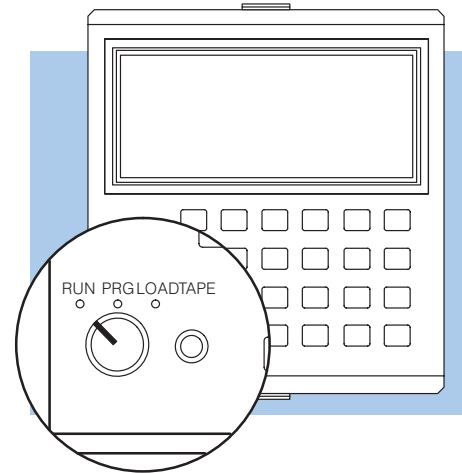
You can only change the CPU mode by using the keyswitch on the front of the handheld programmer.

---

**WARNING: The CPU will automatically change modes when you connect the Handheld Programmer if the keyswitch is set for a different mode of operation. For example, if the CPU is in Run mode and the Handheld Programmer keyswitch is set to the PRG (Program) position, the CPU will automatically enter Program mode when the Handheld is connected.**

---

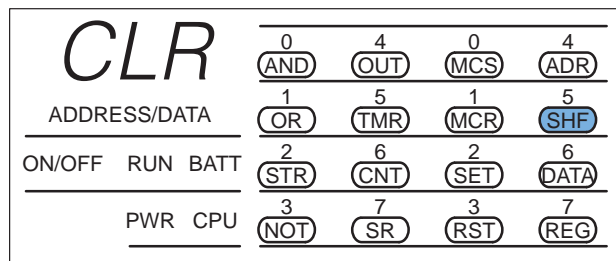
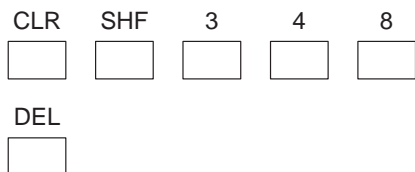
The keyswitch also has a third position, called LOAD. If the keyswitch is in this position you can upload a program from CPU memory to a cassette tape, or download a program from cassette tape to CPU memory.



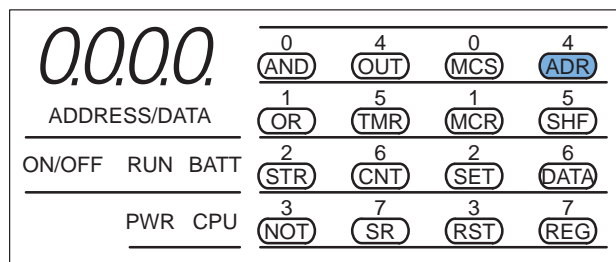
**Clearing an Existing Program**

Before you enter a new program, you should always clear the CPU memory. Only a few keystrokes are required.

**Use these keystrokes**



**Press NXT to clear memory  
or  
Press CLR to abort the operation**



**You've got the Basics!**

Now that you understand how to use the display and how to perform some basic CPU operations, you're ready to enter a program. Chapter 2 provides the keystrokes for entering many of the instructions you'll use in any type of program. If you plan on using RLL<sup>PLUS</sup> programs, you'll also want to look at Chapter 3 for information on the the extra instructions for RLL<sup>PLUS</sup>. Also, with RLL<sup>PLUS</sup>, some of the basic instructions, like timers and counters, are used differently.