6

System Monitoring and Troubleshooting

In This Chapter. . . .

- — Troubleshooting Suggestions
- — Monitoring Discrete Points
- — Forcing Discrete Points
- — Monitoring Register Locations
- — Changing Register Values
- — Monitoring Timer/Counter Values
- — Changing Timer/Counter Values
- — Monitoring Program Stages
- — Forcing Program Stages
- — Error Codes

Troubleshooting Suggestions

The Handheld is very useful in troubleshooting your machine. As with 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.

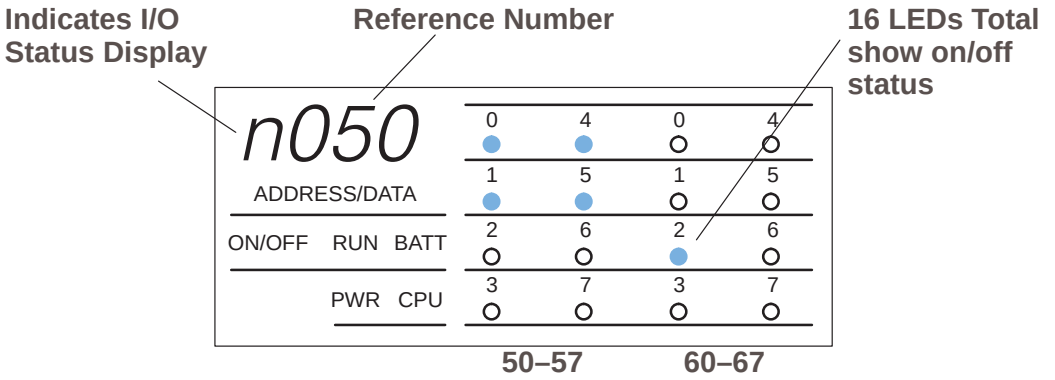
- Monitor Discrete I/O Points — to examine I/O power flow for individual I/O points.
- Force Discrete I/O Points — to examine machine sequences or inconsistencies.
- Monitor Register Locations — to examine word locations to determine if correct values are being used.
- Change Register Values — to force word locations with different values.
- Monitor Timer/Counter Values — to adjust machine timing elements.
- Understand Error Codes — to know where to look first.

Monitoring Discrete Points

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

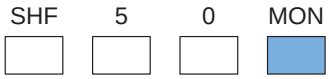
- Inputs
- Output
- Control relays

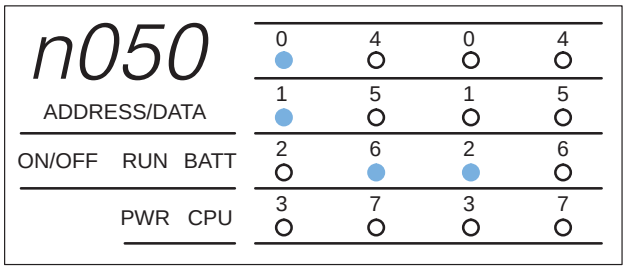
The status display area is also used to show the status for the points being monitored. Here's how the display is organized.



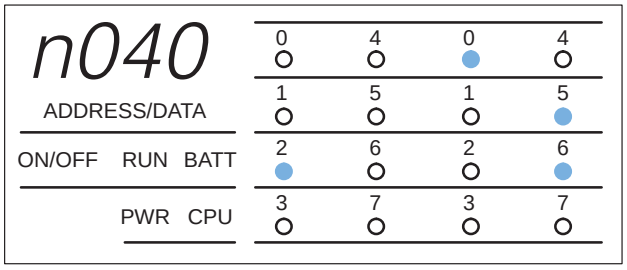
Use the following keystrokes to monitor discrete points. To select a different data type, use the corresponding reference number instead of the one shown. The DL305 memory map is included in Appendix A. This will help you determine the appropriate ranges for the various data types.

Select the reference number to monitor





Use PRV or NXT to scroll through the references (8 point increments)

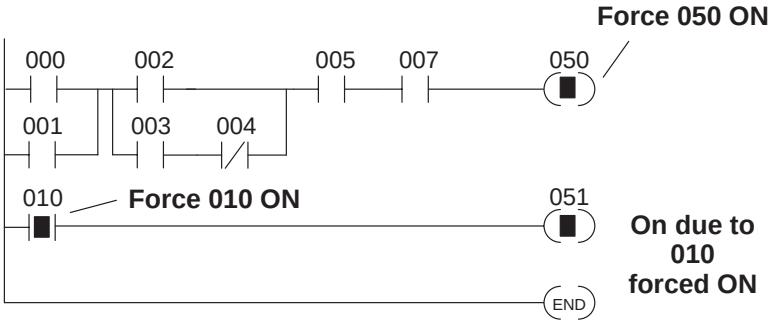


Forcing Discrete Points

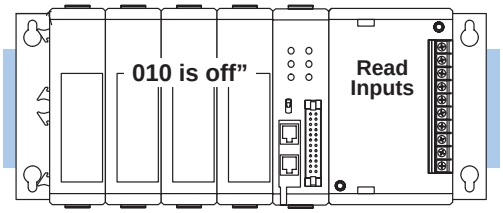
You can also force discrete points with the Handheld Programmer.

It is important to note that the DL305 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.

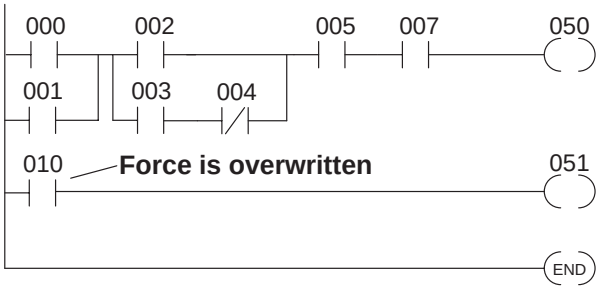
Force I/O Points



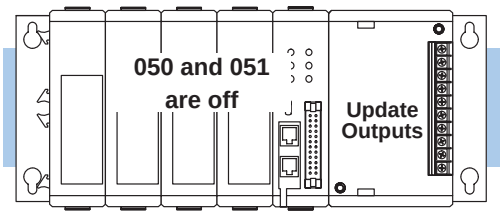
Next Scan



CPU reads the I/O status from the modules. Sees that point 010 is off, overwrites force to turn off 010.



Logic is solved. Point 010, even though previously forced on, is turned off. Points 050 and 051 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.

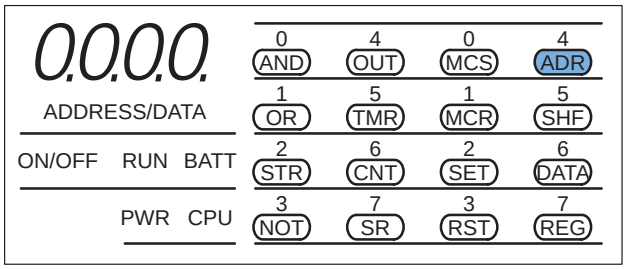
WARNING: Depending on your application, forcing I/O points may cause unpredictable machine operation that can result in a risk of personal injury or equipment damage.

To turn a point on

CLR SET SHF 5 0

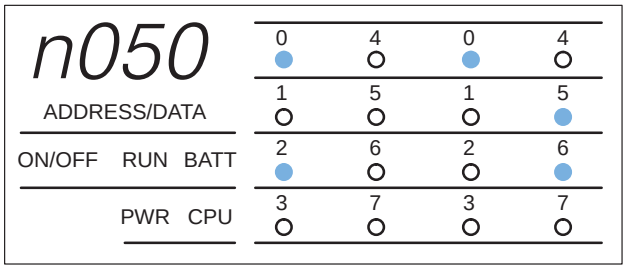
ENT

Display returns to address display



Monitor the point to verify the force (optional)

CLR SHF 5 0 MON

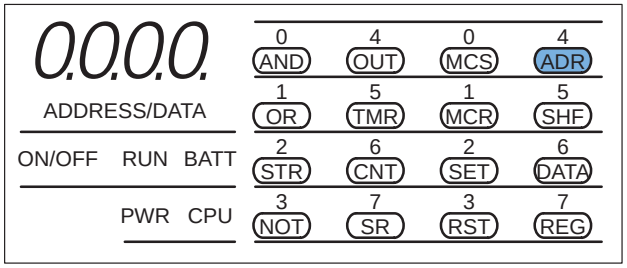


To turn a point off

CLR RST SHF 5 0

ENT

Display returns to address display



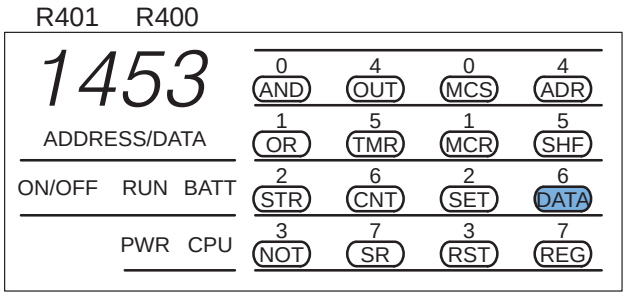
Monitoring Register Locations

You can use the Handheld to monitor and change register locations. When a register is monitored, the handheld programmer will display two register locations (eight bits each) this means that 4 digits of data will be shown. Since data register locations are 8-bit locations, two consecutive data registers will be displayed. When changing values in data register locations, you can also write two consecutive data register locations.

Select the location to monitor

CLR R 4 0 0

MON

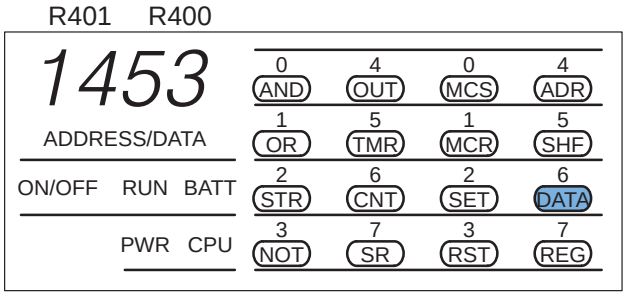


Changing Register Values

Select the location to monitor

CLR R 4 0 0

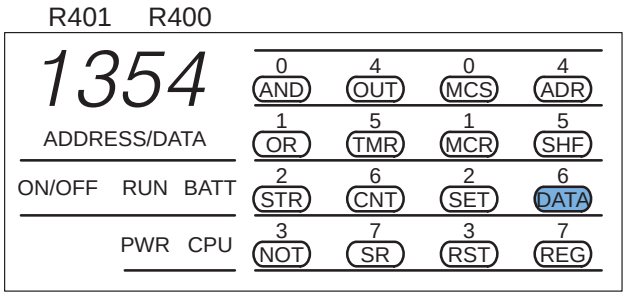
MON



Enter the new value

SHF 1 3 5 4

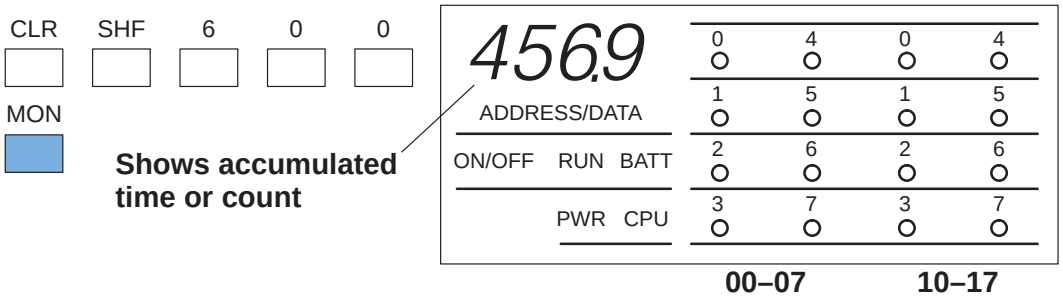
ENT



Monitoring Timer/Counter Current Values

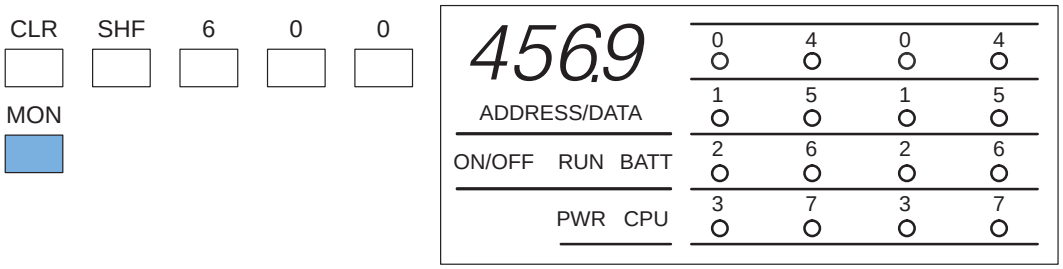
You can also use the Handheld to monitor and/or change timer and counter current values. There are two ways to do this. You can use the register monitoring procedure discussed previously which will display the current value with the decimal point implied for timers. The second method uses slightly different keystrokes and will show timer current values with the decimal point. Using this method also uses the instructions LEDs to indicate the last two digits of the timer/counter memory reference.

Select the location to monitor

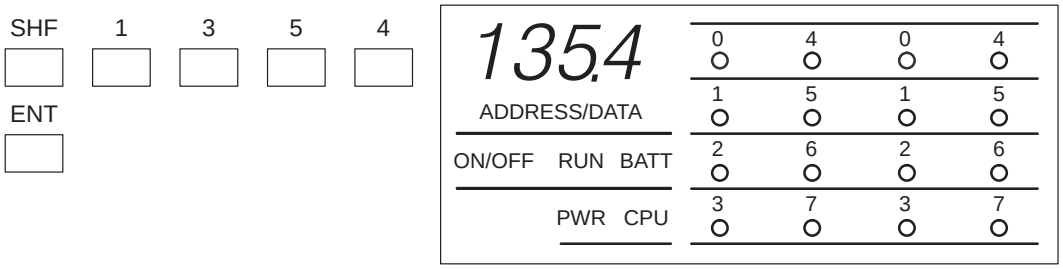


Changing Timer/Counter Current Values

Select the timer location to monitor



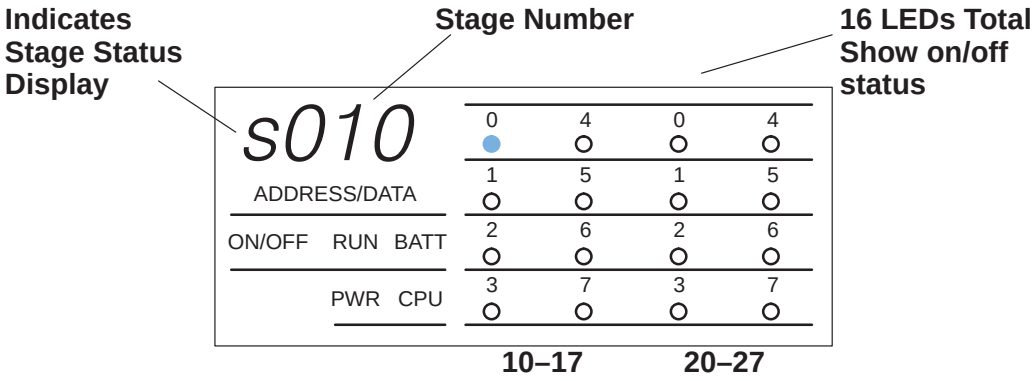
Enter the new value



Monitoring Program Stages

The RLL*PLUS* CPUs also have stages that can be monitored. This is especially useful since it can quickly show you exactly which parts of the program are being executed.

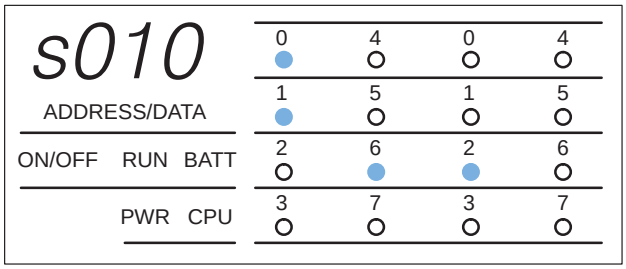
The procedure is very similar to the one used for monitoring discrete I/O points. Here's how the display is organized.



Use the following keystrokes to monitor the stages.

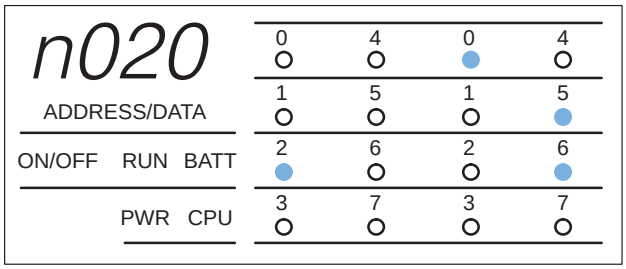
Select the stage number to monitor

SG SHF 1 0 MON



Use PRV or NXT to scroll through the references (8 stage increments)

PRV

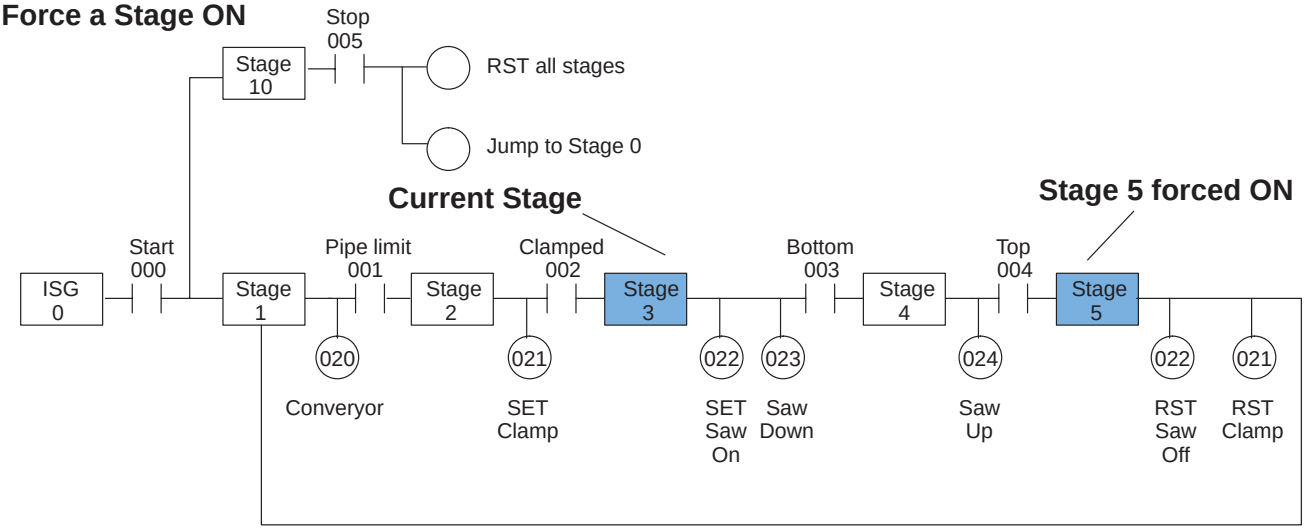


Forcing Program Stages

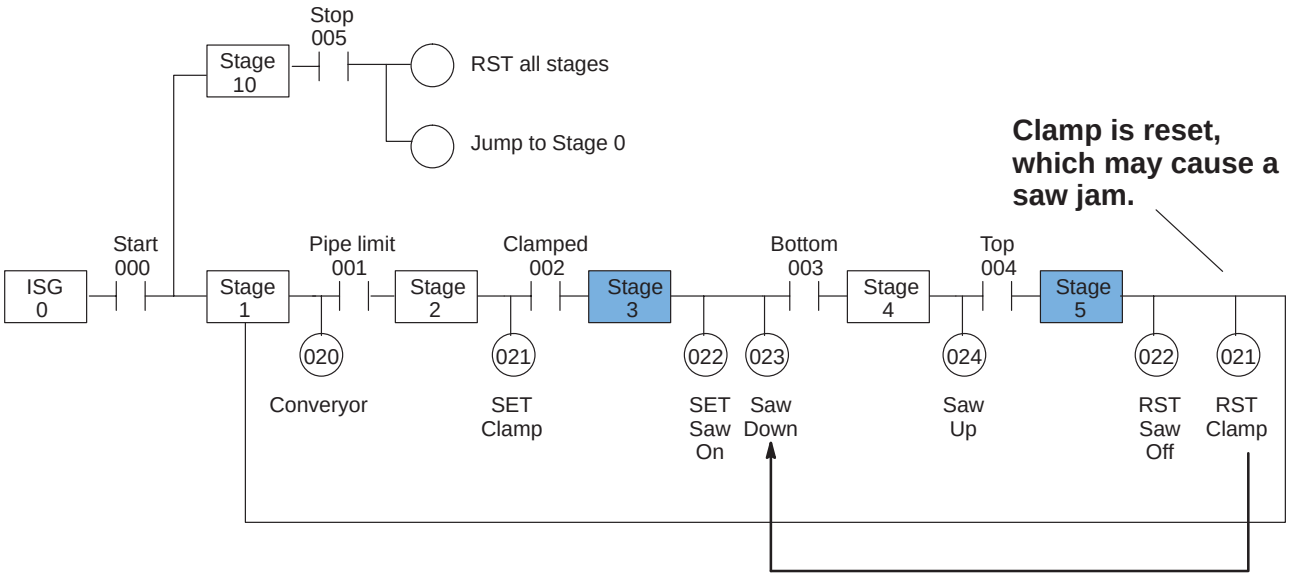
You can also force Program Stages with the Handheld Programmer. However, due to the self-contained nature of this style of programming, you can really cause some problems by forcing stages on and off.

It is important to note that the DL305 CPUs only retain the forced value for one scan if the logic within the stage (or another stage) causes the status to be discarded. The following example shows how the forcing actually works. Assume that the saw takes approximately 10 seconds to reach the bottom limit. (Which is many, many, CPU scans.)

Force a Stage ON



Next Scan



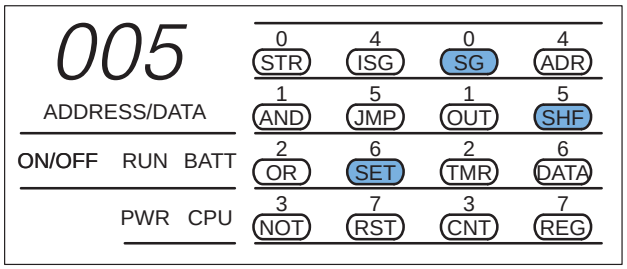
Obviously, there are times when it's perfectly OK to force a program stage on or off. The following example shows the keystrokes required to force an stage.

WARNING: As shown in the example, forcing stages may cause unpredictable machine operation that can result in a risk of personal injury or equipment damage.

To turn a stage on

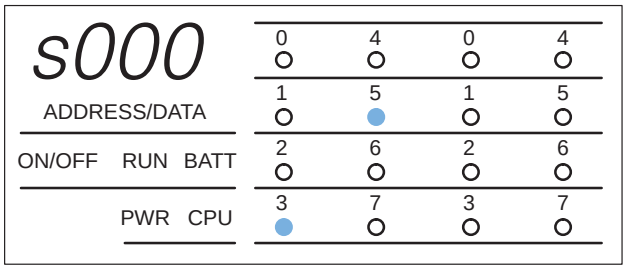
CLR SET SG SHF 5

ENT



Monitor the stage to verify the force (optional)

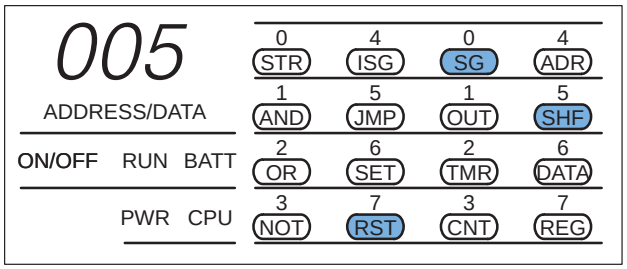
CLR SG SHF 5 MON



To turn a stage off

CLR RST SG SHF 5

ENT



Error Codes

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

DL305 Error Code	Description
E01 Invalid Keystrokes	Invalid keystroke or series of keystrokes entered into the handheld programmer. Refer to the DL305 Handheld Programmer manual for assistance in the operation you are trying to perform.
E02 Input Point Programmed as Output	An I/O point dedicated to an input module has been used as an output in the application program. Change the I/O reference number in the program which is causing the error.
E03 Stack Overflow	The maximum number of instructions utilizing the internal stack has exceeded eight. These instructions can be a combination of AND STRs, OR STRs and MCS/MCR groups. Reduce the number of these instructions which are pushed onto the stack at one time.
E05 (NON Stage) Duplicate Coil Reference	Two or more output coils have the same data type and number. Change the duplicate coil to correct the error. Duplicate coil references are valid with the SET instruction.
E05 (Stage) Duplicate Stage Reference	Two or more Stages have the same reference number. Change the duplicate Stage number to correct the error.
E06 MCR/MCS Mismatch	The number of MCR instructions do not match the number of MCS instructions. Each MCR must have an accompanying MCS.
E07 Missing CNT or SR Contact	A required input contact is missing from a CNT (example, RESET input) or a SR instruction. Refer to the DL305 User Manual for details on these instructions.
E08 Invalid Data Values	The required data values for a TMR, CNT or SR are missing or incorrect. Refer to the DL305 User Manual for details on these instructions.
E09 Incomplete Program Rung	The rung does not terminate with an output as required. Program an output to terminate the rung properly.
E11 Program Full	There is no available program addresses in memory. Reduce the size of the program.

DL305 Error Code	Description
E21 Program Memory Parity Error	A parity error has occurred in the program memory of the CPU. Clear the memory and reload the program. If the error reoccurs replace the CPU. Electrical noise will cause this problem.
E22 Password Error	The password stored in the CPU is invalid. Press the “CLR” key twice on the handheld programmer and the password will be reset to 0000. Re-enter the password if required.
E25 Tape/Program Mismatch	A mismatch was found when a compare was performed on the program in CPU memory and the program stored on tape.
E28 Volume Incorrect On Tape Device.	The volume is incorrect on the tape player being used to load the program to the CPU. Adjust the volume and retry the operation. Refer to the DL305 Handheld Programmer manual for details on tape operation.
E31 RAM Limit Exceeded	The application program required more RAM for execution than is available. Reduce the length of the program.
E99 Instruction Not Found	A search was performed and the specified instruction was not found in the application program.