Tasks for a Sample Mixing Tank Application

Figure 4-2 shows the diagram for a mixing tank. This mixing tank can be used for different applications, such as for making different colors of paint. In this application, two pipelines enter the top of the tank; these supply two different ingredients. A single pipeline at the bottom of the tank transports the finished mixture. The sample program controls the filling operation, monitors the tank level, and controls a mixing and heating cycle. The following tasks describe the process:

Step 1: Fill the tank with Ingredient 1.

Step 2: Fill the tank with Ingredient 2.

Step 3: Monitor the tank level for closure of the high-level switch.

Step 4: Maintain the pump status if the start switch opens.

Step 5: Start the mix-and-heat cycle.

Step 6: Turn on the mixer motor and steam valve.

Step 7: Drain the mixing tank.

Step 8: Count each cycle.

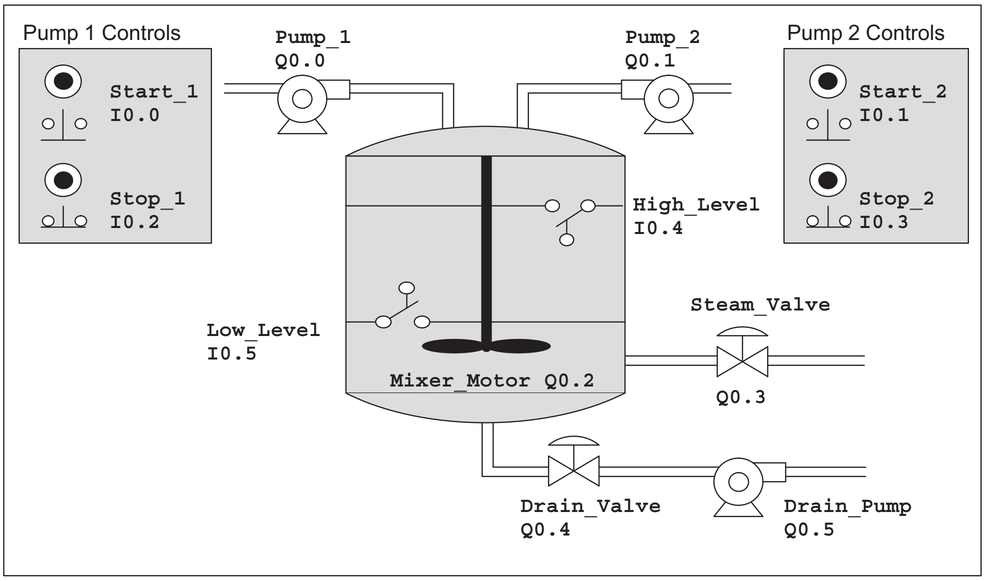


Figure 4-2 Program Example: Mixing Tank

Sample Program in Statement List (STL) and Ladder Logic

You can enter the sample program in either statement list (STL) or ladder representation. Table 4-1 provides the STL version of the sample program, and Figure 4-3 shows the same sample program in ladder. Sections 4.2 to 4.4 guide you through the tasks required to enter this program in ladder.

Table 4-1 Sample Program in Statement List

```
STL                           Description
NETWORK 1                     //Fill the tank with Ingredient 1.
LD      “Start_1”
O       “Pump_1”
A       “Stop_1”
AN      “High_Level”
=       “Pump_1”

NETWORK 2                     //Fill the tank with Ingredient 2.
LD      “Start_2”
O       “Pump_2”
A       “Stop_2”
AN      “High_Level”
=       “Pump_2”

NETWORK 3                     //Set memory bit if high level is reached.
LD      “High_Level”
S       “Hi_Lev_Reached”, 1

NETWORK 4                     //Start timer if high level is reached.
LD      “Hi_Lev_Reached”
TON     “Mix_Timer”, +100

NETWORK 5                     //Turn on the mixer motor.
LDN     “Mix_Timer”
A       “Hi_Lev_Reached”
=       “Mixer_Motor”
=       “Steam_Valve”

NETWORK 6                     //Drain the mixing tank.
LD      “Mix_Timer”
AN      “Low_Level”
=       “Drain_Valve”
=       “Drain_Pump”

NETWORK 7                     //Count each cycle.
LD      “Low_Level”
A       “Mix_Timer”
LD      “Reset”
CTU     “Cycle_Counter”, +12

NETWORK 8                     //Reset memory bit if low level or time-out.
LD      “Low_Level”
A       “Mix_Timer”
R       “Hi_Lev_Reached”, 1

NETWORK 9                     //End the main program.
MEND
```

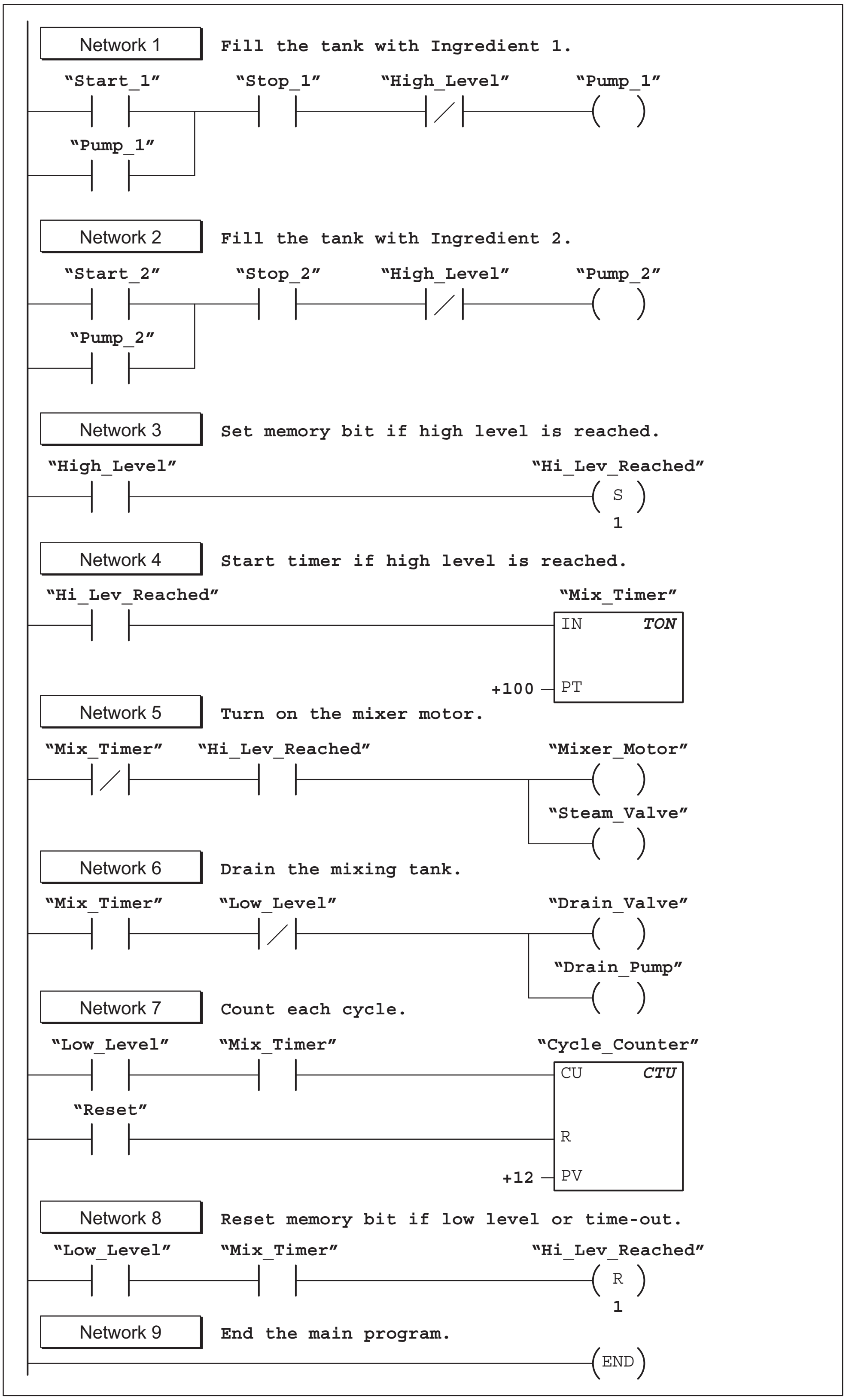


Figure 4-3 Sample Program in Ladder Logic

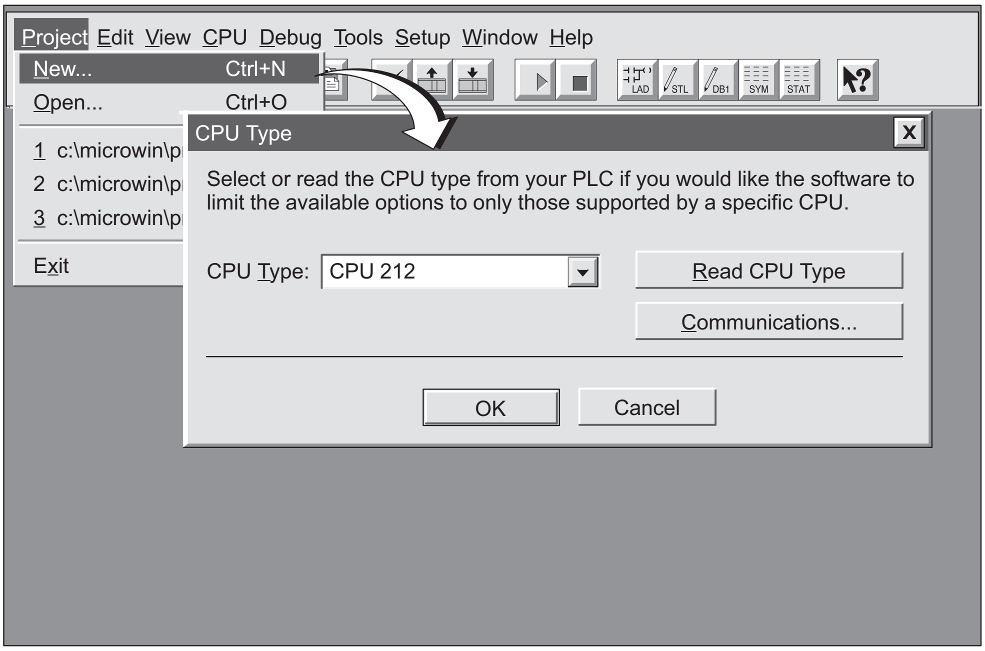
4.2 Task: Create a Project

Creating a New Project

When you create or open a project, STEP 7-Micro/WIN starts the Ladder or STL Editor (OB1), and depending on your selected preference, the Data Block Editor (DB1), the Status Chart, and the Symbol Table.

To create a new project, select the menu command **Project ▶ New...**, as shown in Figure 4-4, or click the New Project toolbar button.

The CPU Type dialog box is displayed. Select your CPU type from the drop-down list box.



Select or read the CPU type from your PLC if you would like the software to limit the available options to only those supported by a specific CPU.

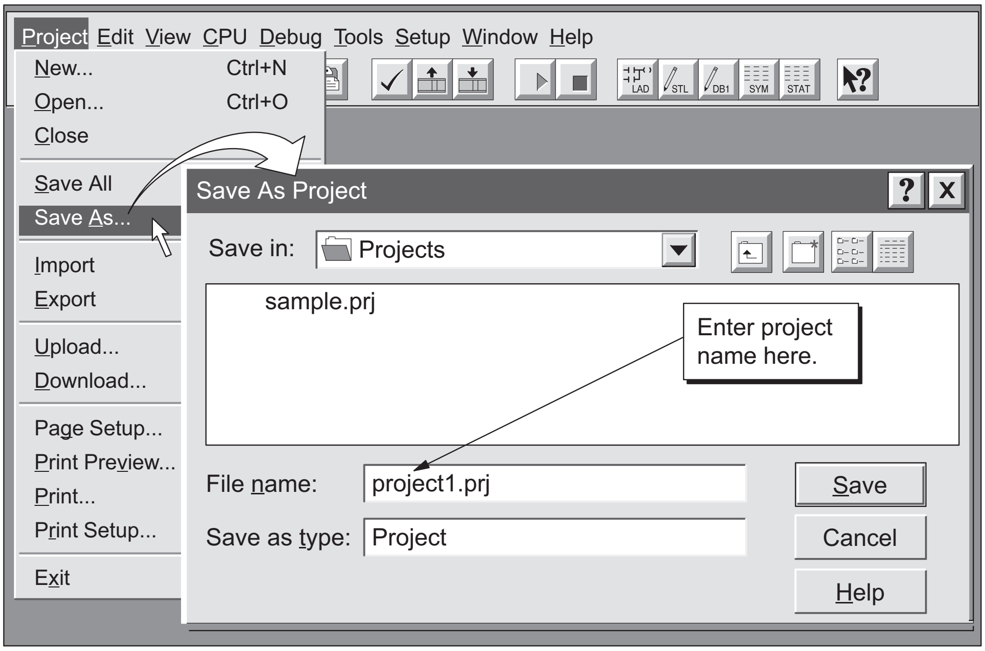
CPU Type: CPU 212

Figure 4-4 Creating a New Project and Selecting the CPU Type

Naming the Sample Project

You can name your project at any time; for this example, refer to Figure 4-5 and follow these steps to name the project:

1. Select the menu command **Project ▸ Save As...** .
2. In the File Name field, type the following: `project1.prj`
3. Click the “Save” button.



Save As Project

Save in: Projects

sample.prj

Enter project name here.

File name: project1.prj

Save as type: Project

Figure 4-5 Naming the Sample Project

4.3 Task: Create a Symbol Table

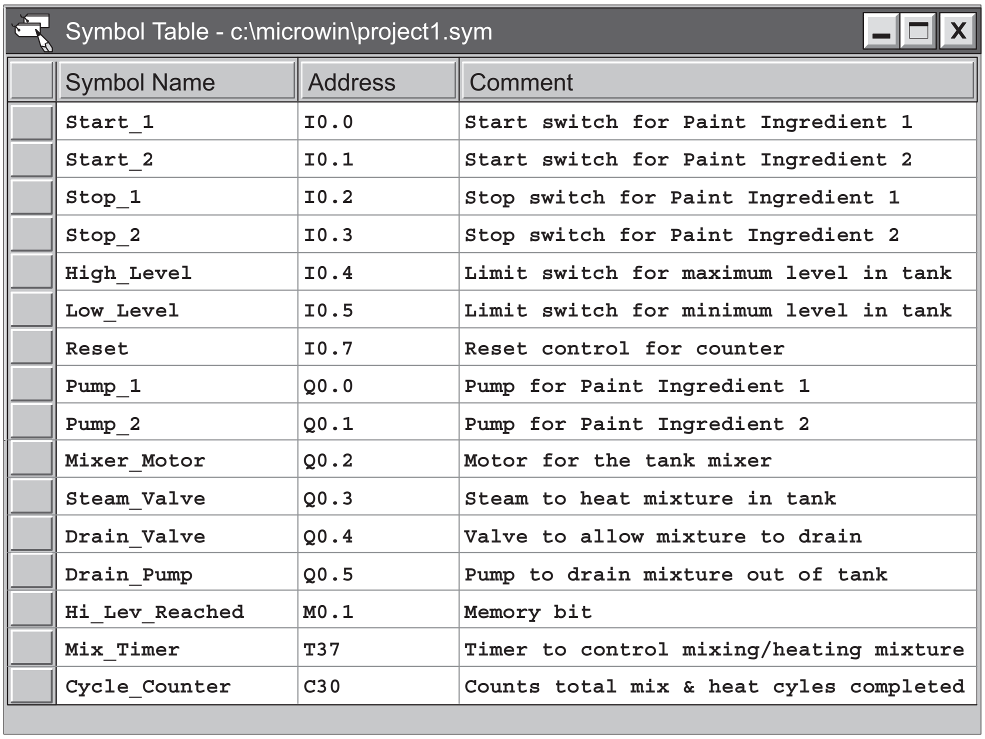
Opening the Symbol Table Editor

To define the set of symbol names used to represent absolute addresses in the sample program, open the Symbol Table editor. Double-click the icon, or click the Restore or Maximize button on the icon (in Windows 95). You can also select the menu command **View ▶ Symbol Table**.

Entering the Symbol Names

Figure 4-6 shows the list of symbol names and the corresponding addresses for the sample program. To enter the symbol names, follow these steps:

1. Select the first cell in the Symbol Name column, and type the following: `Start_1`
2. Press ENTER to move the focus to the first cell in the Address column. Type the address `I0.0` and press ENTER. The focus moves to the cell in the Comment column. (Comments are optional, but they are a useful way to document the elements in your program.)
3. Press ENTER to start the next symbol row, and repeat these steps for each of the remaining symbol names and addresses.
4. Use the menu command **Project ▶ Save All** to save your Symbol Table.



Symbol Table - c:\microwin\project1.sym

Symbol Name	Address	Comment
Start_1	I0.0	Start switch for Paint Ingredient 1
Start_2	I0.1	Start switch for Paint Ingredient 2
Stop_1	I0.2	Stop switch for Paint Ingredient 1
Stop_2	I0.3	Stop switch for Paint Ingredient 2
High_Level	I0.4	Limit switch for maximum level in tank
Low_Level	I0.5	Limit switch for minimum level in tank
Reset	I0.7	Reset control for counter
Pump_1	Q0.0	Pump for Paint Ingredient 1
Pump_2	Q0.1	Pump for Paint Ingredient 2
Mixer_Motor	Q0.2	Motor for the tank mixer
Steam_Valve	Q0.3	Steam to heat mixture in tank
Drain_Valve	Q0.4	Valve to allow mixture to drain
Drain_Pump	Q0.5	Pump to drain mixture out of tank
Hi_Lev_Reached	M0.1	Memory bit
Mix_Timer	T37	Timer to control mixing/heating mixture
Cycle_Counter	C30	Counts total mix & heat cyles completed

Figure 4-6 Symbol Table for the Sample Program

Programming with Symbolic Addresses

Before you start entering your program, make sure the ladder view is set for symbolic addressing. Use the menu command **View ▶ Symbolic Addressing** and look for a check mark next to the menu item, which indicates that symbolic addressing is enabled.

Note
Symbol names are case-sensitive. The name you enter must match exactly the uppercase and lowercase characters entered in the symbol table. If there is any mismatch, the cursor stays on the element and the message “Invalid parameter” appears in the status bar at the bottom of the main window.

4.4 Task: Enter the Program in Ladder Logic

Opening the Ladder Editor

To access the Ladder Editor, double-click the icon at the bottom of the main window. Figure 4-7 shows some of the basic tools in the Ladder Editor.

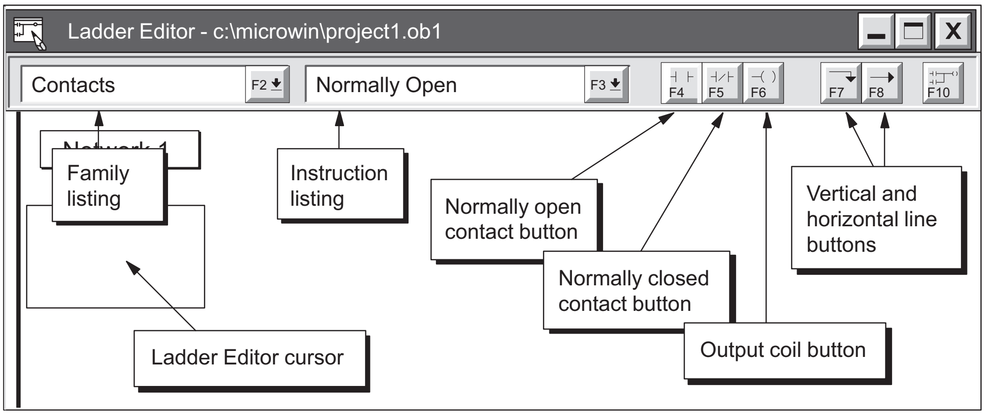

Ladder Editor - c:\microwin\project1.ob1
Contacts
F2
Normally Open
F3
F4
F5
F6
F7
F8
F10
Network 1
Family listing
Instruction listing
Normally open contact button
Normally closed contact button
Vertical and horizontal line buttons
Output coil button
Ladder Editor cursor

Figure 4-7 Some of the Basic Ladder Editor Tools

Instruction Toolbar in the Ladder Editor

You can also select **View ▸ Instruction Toolbar** to display the Ladder Instruction Toolbar. See Figure 4-8.

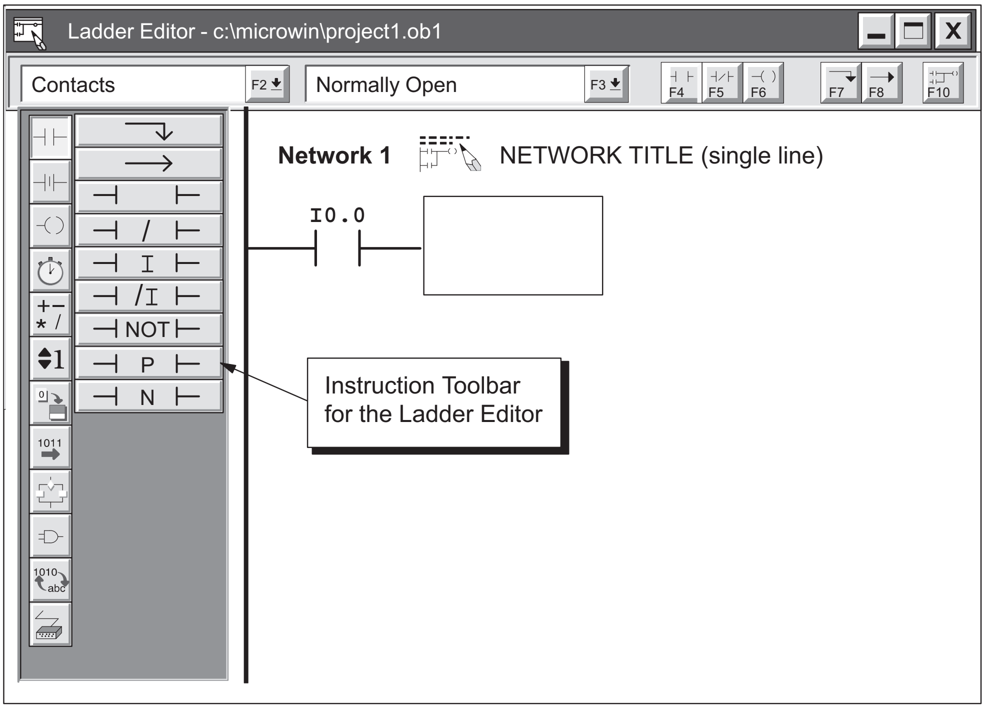

Ladder Editor - c:\microwin\project1.ob1
Contacts
F2
Normally Open
F3
F4
F5
F6
F7
F8
F10
Network 1
NETWORK TITLE (single line)
I0.0
I
/I
NOT
P
N
Instruction Toolbar for the Ladder Editor

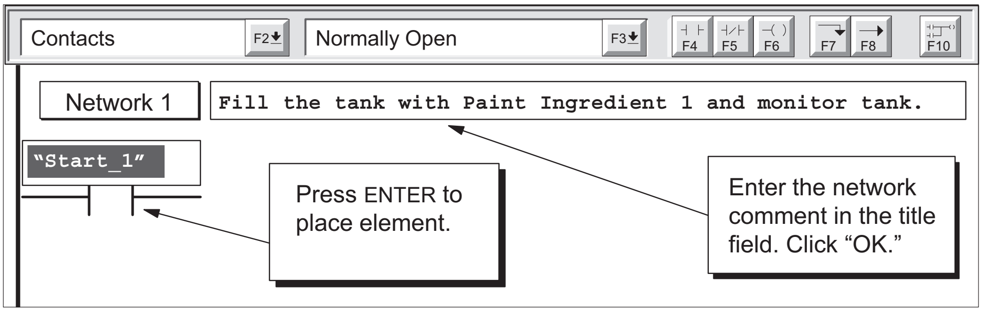
Figure 4-8 Some of the Basic Ladder Editor Tools

Entering the First Network Element

Follow these steps to enter the first network of the sample program:

1. Double click on or near the numbered network label to access the Title field in the Comment Editor. Type the comment shown in Figure 4-9, and click “OK.”
2. Press the down arrow key. The ladder cursor moves down to the first column position on the left.
3. Select the normally open by selecting “Contacts” from the family listing and then selecting “Normally Open” from the instruction listing.
4. Press ENTER, and a normally open contact appears with the name “Start_1” highlighted above it.

   (Every time you enter a contact, the software displays the default address of `I0.0`, which in this example is defined as Start_1 in the Symbol Table.)
5. “Start_1” is the first element required for Network 1. Press ENTER to confirm the first element and its symbol name. The ladder cursor moves to the second column position.



`Fill the tank with Paint Ingredient 1 and monitor tank.`

Press ENTER to place element.

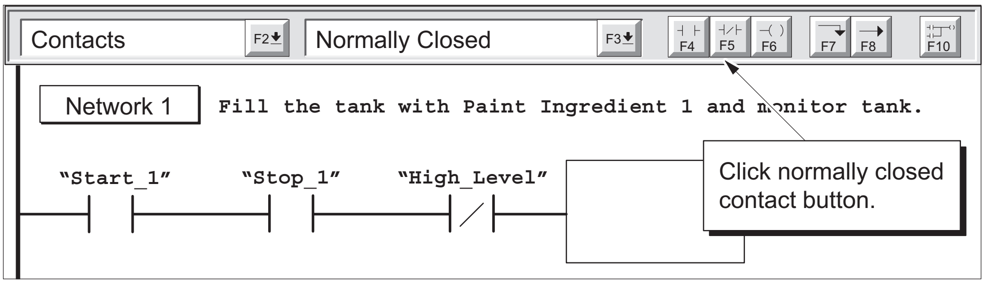
Enter the network comment in the title field. Click “OK.”

Figure 4-9   Entering the Network Comment and the First Ladder Element

Follow these steps to enter the remaining contacts of the first network:

1. Press ENTER to enter the second element. A normally open contact appears with the default symbol name “Start_1” highlighted above it.
2. Type `Stop_1` and press ENTER. The cursor moves to the next column.
3. Click the normally closed contact button (“F5”). A closed contact appears with the default symbol name “Start_1” highlighted above it.
4. Type `High_Level` and press ENTER.

The ladder network should look like the one shown in Figure 4-10.



`Fill the tank with Paint Ingredient 1 and monitor tank.`

Click normally closed contact button.

Figure 4-10   Entering the Next Ladder Element

The ladder cursor is now positioned to the right of the normally closed “High_Level” input. Refer to Figure 4-11 and follow these steps to complete the first network:

1. Click the coil button (“F6”) and move the mouse cursor inside the ladder cursor and click. A coil appears with the name “Pump_1” highlighted above it. (Each coil that you enter is given the default address of Q0.0, which in this case has been defined as Pump_1 in the symbol table.)
2. Press ENTER to confirm the coil and its symbol name.
3. Use the mouse or press the left arrow key to move the cursor back to the first element of the current network.
4. Click the vertical line button (“F7”) to draw a vertical line between the first and second contacts.
5. Click the normally open contact button (“F4”) on the toolbar, and press ENTER. A contact with the name “Start_1” appears.
6. Type **`Pump_1`** and press ENTER.

The first network is now complete.

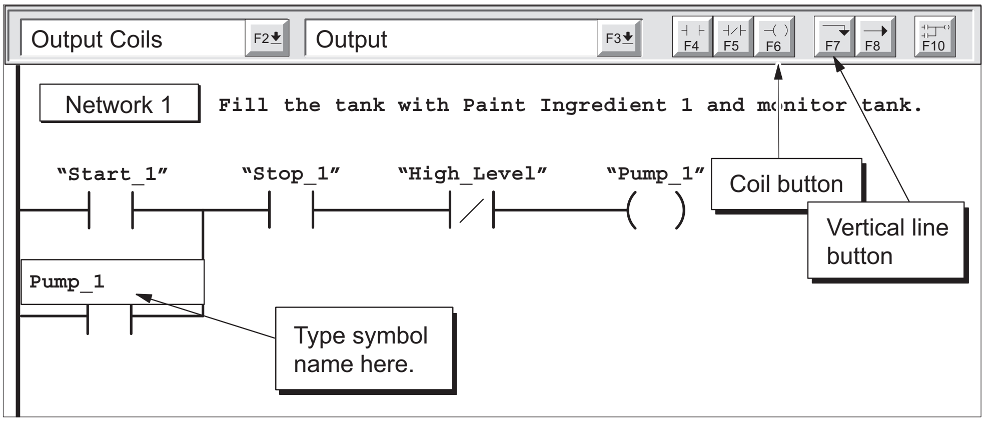
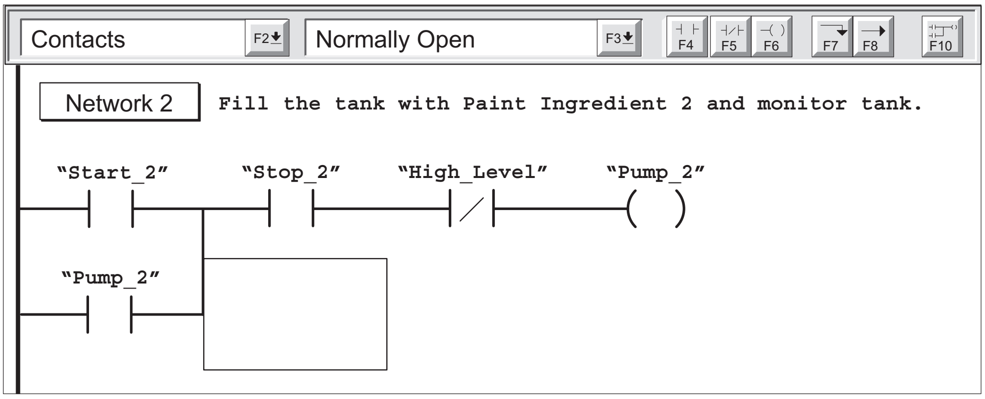


Figure 4-11 Completing the First Network

Entering the Second Network

Follow these steps to enter the second network of the sample program:

1. Use the mouse or press the down arrow key to move the cursor to Network 2.
2. In the network comment field, type the comment shown in Figure 4-12. (Since the comment for Network 2 is nearly identical to the one for Network 1, you can also select and copy the text from Network 1 and paste it into the comment field for Network 2, then change the paint ingredient number to 2.)
3. Repeat the steps you used to enter the elements of Network 1, using the symbol names shown in Figure 4-12.
4. After you finish Network 2, move the cursor down to Network 3.



Fill the tank with Paint Ingredient 2 and monitor tank.

Figure 4-12 Entering the Second Network

Entering the Remaining Networks

From this point on, you can follow the same general procedures that you have used up to now to enter the remaining networks. Refer to Figure 4-3 for the remaining networks.

Compiling the Program

After completing the sample program, check the syntax by selecting the menu command **CPU ▶ Compile** or by clicking the Compile button. ✓

If you have entered all the networks correctly as shown in the sample program, you get a "Compile Successful" message that also includes information on the number of networks and the amount of memory used by the program. Otherwise, the Compile message indicates which networks contain errors.

Saving the Sample Program

You can save your project by selecting the menu command **Project ▶ Save All** or by clicking the Save All button. 🖫 This action saves all the components of your sample project.

4.5 Task: Create a Status Chart

Building Your Status Chart

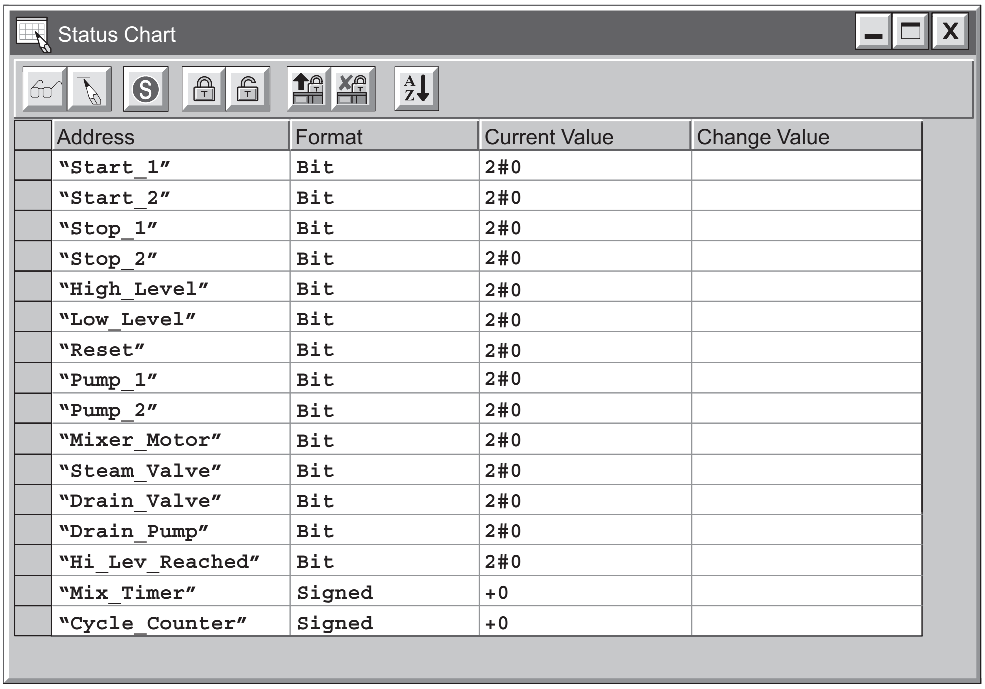
To monitor the status of selected elements in the sample program, you create a Status Chart that contains the elements that you want to monitor while running the program. To access the Status Chart editor, double-click the icon at the bottom of the main window. Then enter the elements for the sample program by following these steps:

1. Select the first cell in the Address column, and type the following: `Start_1`
2. Press ENTER to confirm your entry. This element type can only be displayed in bit format (either 1 or 0), so you cannot change the format type.
3. Select the next row, and repeat these steps for each of the remaining elements, as shown in Figure 4-13.

   When an Address cell is in focus and the row below is empty, pressing ENTER automatically increments the address for each additional row. Refer to the online Help for more information about using the Status Chart.

   You can use the menu command **Edit ▸ Insert Row** (or the INSERT or INS key) to insert a blank row above the row containing the cursor.
4. The timer T37 and the counter C30 can each be displayed in other formats. With the focus in the Format column cell, press the SPACEBAR to cycle through the formats that are valid for these element types. For this example, select Signed for the timer and counter.

Save your Status Chart by selecting the menu command **Project ▸ Save All** or by clicking the Save All button.



Address	Format	Current Value	Change Value
"Start_1"	Bit	2#0	
"Start_2"	Bit	2#0	
"Stop_1"	Bit	2#0	
"Stop_2"	Bit	2#0	
"High_Level"	Bit	2#0	
"Low_Level"	Bit	2#0	
"Reset"	Bit	2#0	
"Pump_1"	Bit	2#0	
"Pump_2"	Bit	2#0	
"Mixer_Motor"	Bit	2#0	
"Steam_Valve"	Bit	2#0	
"Drain_Valve"	Bit	2#0	
"Drain_Pump"	Bit	2#0	
"Hi_Lev_Reached"	Bit	2#0	
"Mix_Timer"	Signed	+0	
"Cycle_Counter"	Signed	+0	

Figure 4-13 Status Chart for the Sample Program

4.6 Task: Download and Monitor the Sample Program

Next you must download your program to the CPU and place the CPU in RUN mode. You can then use the Debug features to monitor or debug the operation of your program.

Downloading the Project to the CPU

Before you can download the program to the CPU, the CPU must be in STOP mode. Follow these steps to select STOP mode and to download the program:

1. Set the CPU mode switch (which is located under the access cover of the CPU module) to TERM or STOP.
2. Select the menu command **CPU ▸ Stop** or click the Stop button ■ in the main window.
3. Answer “Yes” to confirm the action.
4. Select the menu command **Project ▸ Download...** or click the Download button in the main window: ⤓
5. The Download dialog box allows you to specify the project components you want to download. Press ENTER or click “OK.”

An information message tells you whether or not the download operation was successful.

Note
STEP 7-Micro/WIN does not verify that your program uses memory or I/O addresses that are valid for the specific CPU. If you attempt to download a program that uses addresses beyond the range of the CPU or program instructions that are not supported by the CPU, the CPU rejects the attempt to download the program and displays an error message.

You must ensure that all memory locations, I/O addresses, and instructions used by your program are valid for the CPU you are using.

Changing the CPU to RUN Mode

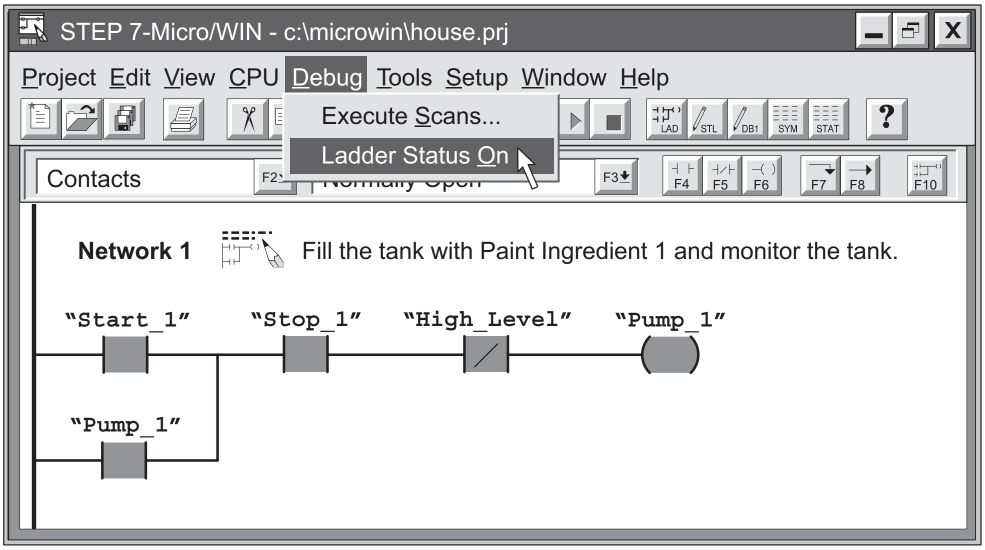
If the download was successful, you can now place the CPU in RUN mode.

1. Select the menu command **CPU ▸ Run** or click the Run button ▷ in the main window.
2. Answer “Yes” to confirm the action.

Monitoring Ladder Status

Ladder status shows the current state of events in your program. Reopen the Ladder Editor window, if necessary, and select the menu command **Debug ▶ Ladder Status On**.

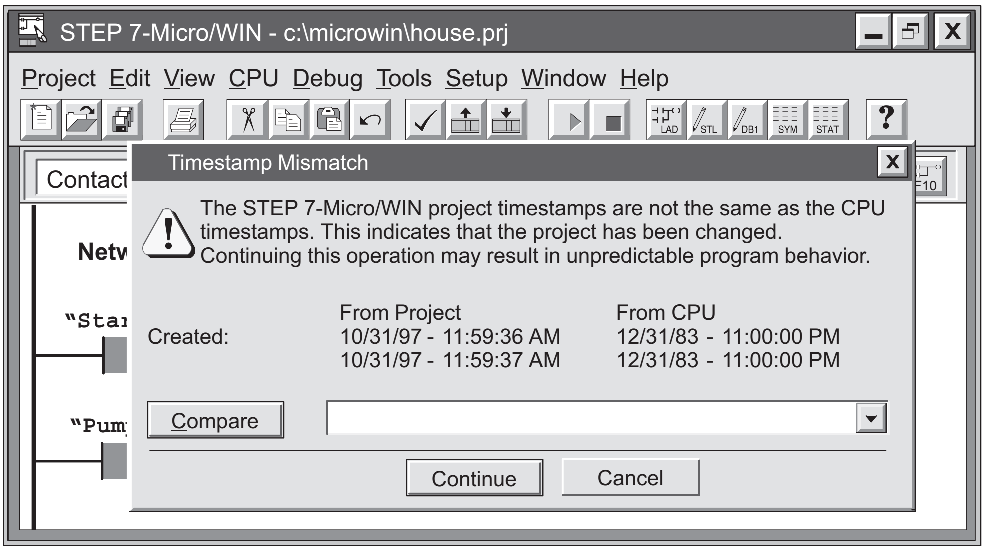
If you have an input simulator connected to the input terminals on your CPU, you can turn on switches to see power flow and logic execution. For example, if you turn on switches `I0.0` and `I0.2`, and the switch for `I0.4` ("High_Level") is off, the power flow for Network 1 is complete. The network looks like the one shown in Figure 4-14.



Network 1 Fill the tank with Paint Ingredient 1 and monitor the tank.

Figure 4-14 Monitoring Status of the First Network

If your STEP 7-Micro/WIN program does not match the CPU program, you are notified by the warning screen shown in Figure 4-15. You are then asked to either compare the program to the CPU, continue this operation, or cancel the operation.


STEP 7-Micro/WIN - c:\microwin\house.prj
Project Edit View CPU Debug Tools Setup Window Help
LAD STL DB1 SYM STAT
Contact
F10
Netw
"Star
"Pum
Timestamp Mismatch
Compare
Continue
Cancel

The STEP 7-Micro/WIN project timestamps are not the same as the CPU timestamps. This indicates that the project has been changed. Continuing this operation may result in unpredictable program behavior.

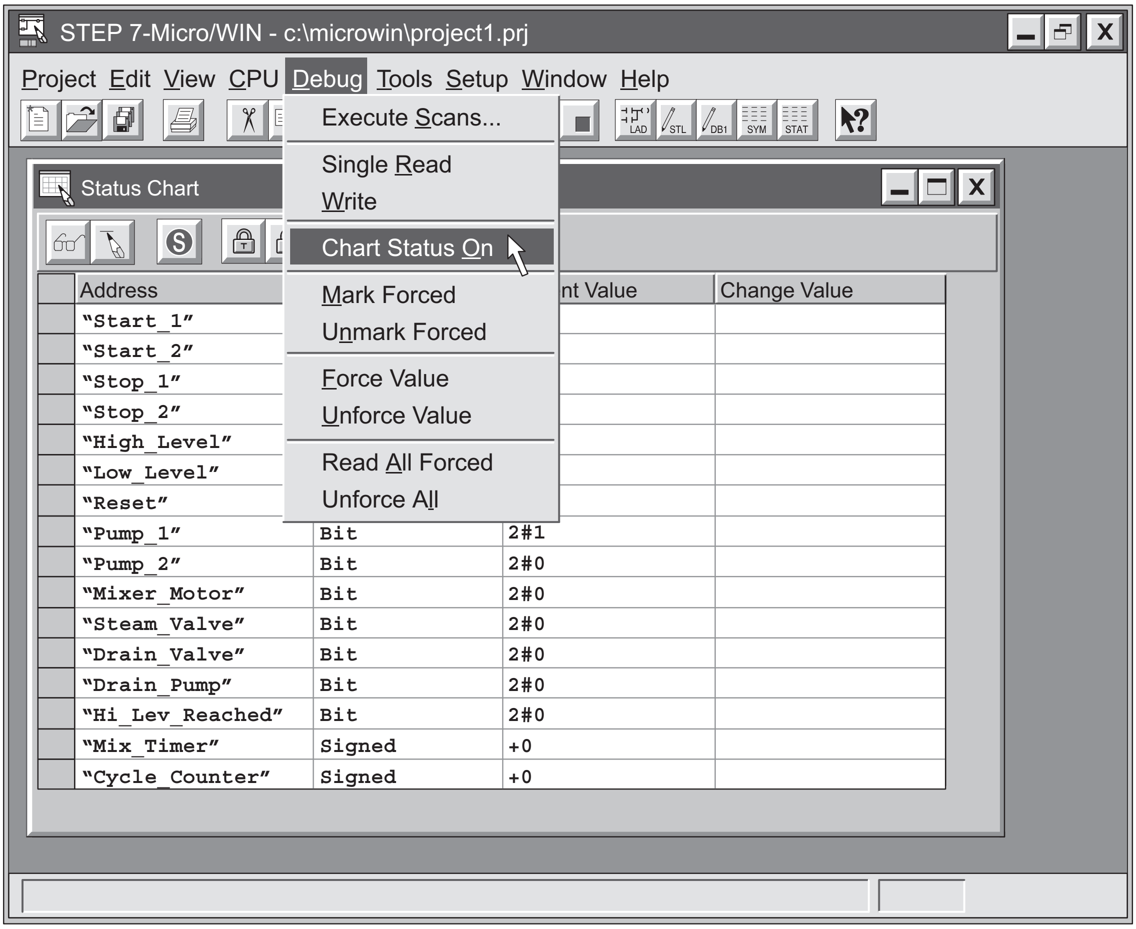
	From Project	From CPU
Created:	10/31/97 - 11:59:36 AM	12/31/83 - 11:00:00 PM
	10/31/97 - 11:59:37 AM	12/31/83 - 11:00:00 PM

Figure 4-15 Timestamp Mismatch Warning Screen

Viewing the Current Status of Program Elements

You can use the Status Chart to monitor or modify the current values of any I/O points or memory locations. Reopen the Chart window, if necessary, and select the menu command **Debug ▸ Chart Status On**, as shown in Figure 4-16. As you switch inputs on or off with the CPU in RUN mode, the Status Chart shows the current status of each element.

- To view the current PLC value of the elements in your program, click the Single Read button 👓 or the Continuous Read button Ⓢ in the Status Chart window.
- To stop the reading of status, click the Continuous Read button Ⓢ in the Status Chart window.



Address		nt Value	Change Value
“Start_1”			
“Start_2”			
“Stop_1”			
“Stop_2”			
“High_Level”			
“Low_Level”			
“Reset”			
“Pump_1”	Bit	2#1	
“Pump_2”	Bit	2#0	
“Mixer_Motor”	Bit	2#0	
“Steam_Valve”	Bit	2#0	
“Drain_Valve”	Bit	2#0	
“Drain_Pump”	Bit	2#0	
“Hi_Lev_Reached”	Bit	2#0	
“Mix_Timer”	Signed	+0	
“Cycle_Counter”	Signed	+0	

Figure 4-16 Monitoring the Status Chart of the Sample Program