5

Testing Your Program on the Simulator

This chapter describes how to the build the application program using the EZBuilder code generator and how to test it using the TRAKKER Antares Simulator. It also provides information on downloading your completed program to the TRAKKER Antares terminal, and points you to some example applications available for study.

Task 13: Building the Program

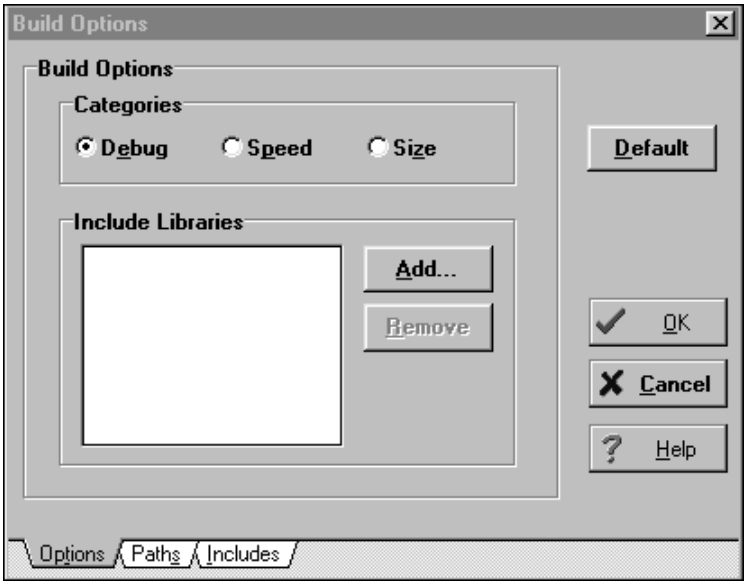
EZBuilder is a fast, easy development tool for TRAKKER Antares terminals. In earlier chapters, you learned how to create menus, screens, and transactions. You created data fields, labels, and various actions. And, you didn't need much (or any) knowledge of programming to do it!

In this chapter, you will learn if your application program (your .IMP file) works. You will see how easy it is for you to set certain options and for EZBuilder to quickly build—called "compile" in programmer's language—your executable program (an .EXE file). Using the TRAKKER Antares Simulator, you can test your program immediately after it is compiled.

If your tests prove that your program works, you can easily download your application to the TRAKKER Antares terminal and test it again from there.

To set your Build options

1. From the Options menu, choose the Build command. The default Build Options dialog box is illustrated next.

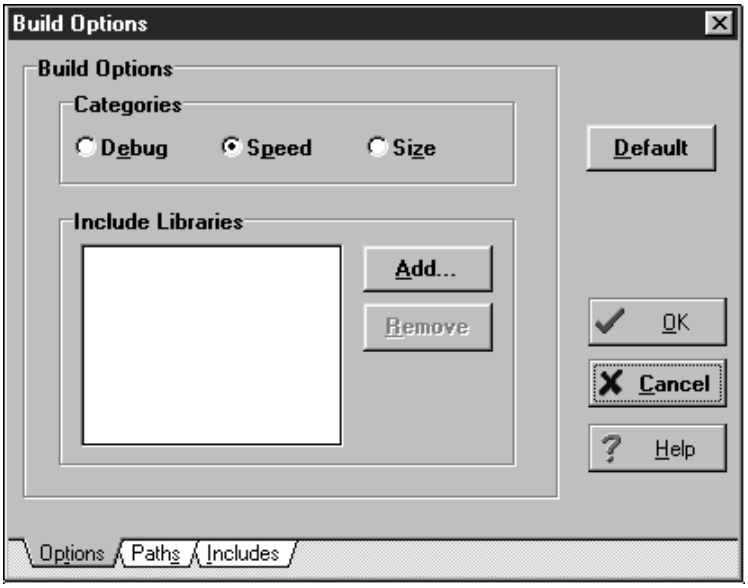


2. There are three Categories or "modes" in which you can build your program.
   - Debug mode (default). This mode puts the debugging information required by CodeView Debugger into the executable file. This mode is generally used only by C programmers. Because the debug information is included, this mode produces a larger application.
   - Speed mode, for optimal speed in executing the program. This mode does not put the debugging information required by CodeView Debugger into the executable file. Therefore, the application runs faster.
   - Size mode, for optimal size of the executable program file. This mode does not put the debugging information required by CodeView Debugger into the executable file. Therefore, the application is smaller.

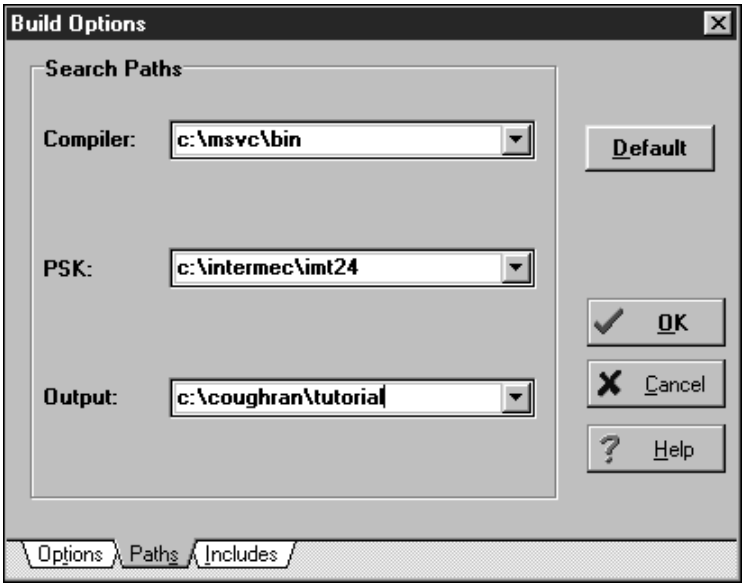
   Note: *Unless you are a C programmer, C program code won't be meaningful or useful to you. There should be no errors or warnings resulting from the C program code generated by EZBuilder.*

   Note: *If you add any C program code to the program after EZBuilder generates its code, be aware that when recompiling in EZBuilder, EZBuilder will regenerate all code, rewriting over any C program code changes you may have made outside of EZBuilder.*

3. Click to set Categories to the Speed mode, as illustrated next. The Speed setting allows you to build your EZBuilder application without any extra C program code being added.



4. At the bottom of the Build Options dialog box there are three tabs: Options (illustrated previously), Paths, and Includes. Choose the Paths tab to view the Search Paths dialog box, as illustrated next.



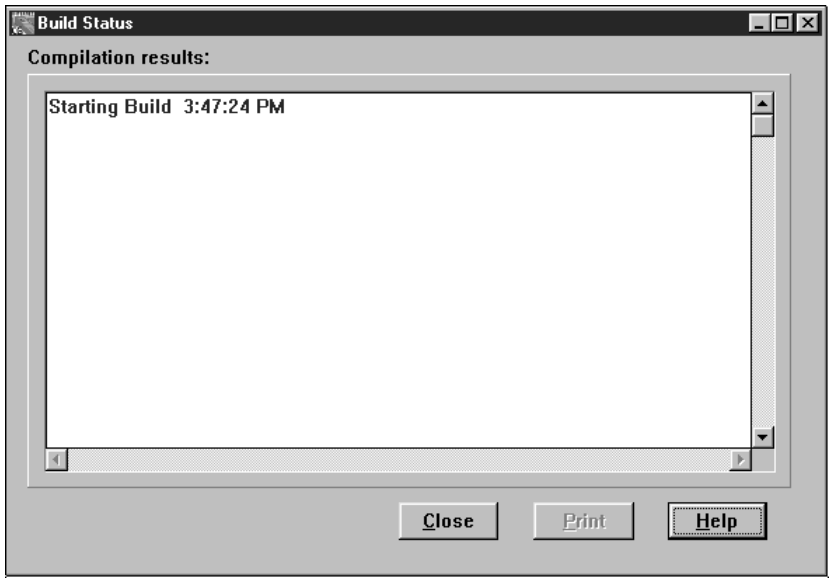
5. EZBuilder will search for the compiler and certain libraries as it builds your program. You must indicate correct paths to these things, and you must also indicate where you want your generated EZBuilder application output file (your executable C program) stored.
   - The Compile path indicates the location of the Microsoft Visual C++ compiler on your computer. Make sure the path is correct.
   - The PSK path indicates the location of the TRAKKER Antares PSK libraries and Simulator on your computer. Make sure the path is correct.
   - The Output path indicates the location where you want your generated application program to be put after EZBuilder has built it. This path specifies where the Simulator will put or look for all files opened in your application as well as all files used to simulate COM or network input and output. Make sure the path is correct.

     Note: *Output path C:\INTERMEC\IMT24\EXAMPLES is where you will find some EZBuilder example applications which you can study. For more information on these examples, see the* EZBuilder Getting Started Guide.

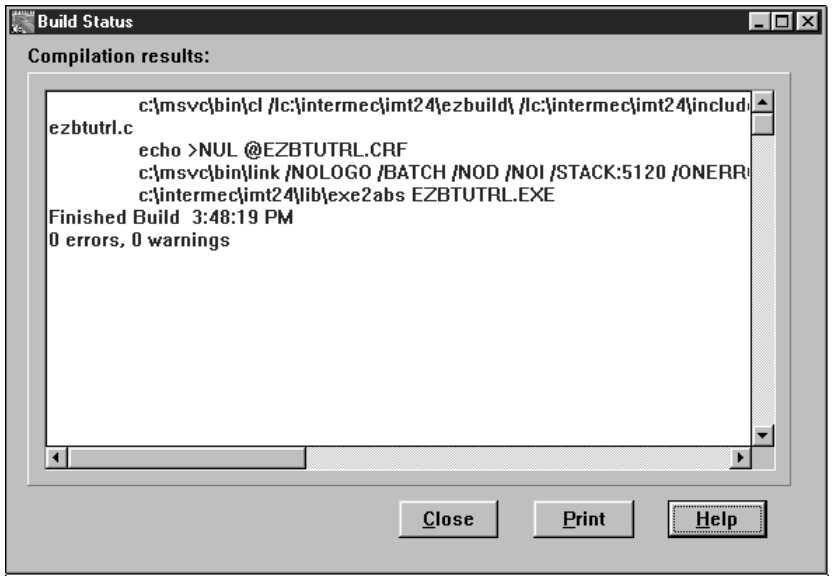
6. Click OK when you are ready to build the program.

To build (compile) your application program

1. From the Build menu, choose the Build command. You will see the Build Status dialog box which shows a message that EZBuilder is starting to build your program. The current computer system time is also provided, as illustrated next.



2. In a few moments, the Compilation information will be shown to you, similar to the next illustration. Notice there are 0 errors and 0 warnings. Also, notice how it took less than a minute to compile the program.



Note: *If you build your program in Debug mode rather than Speed mode, and you receive warnings, run your program anyway to test it. If you receive errors, you will not be able to run your program until you have first debugged it.*

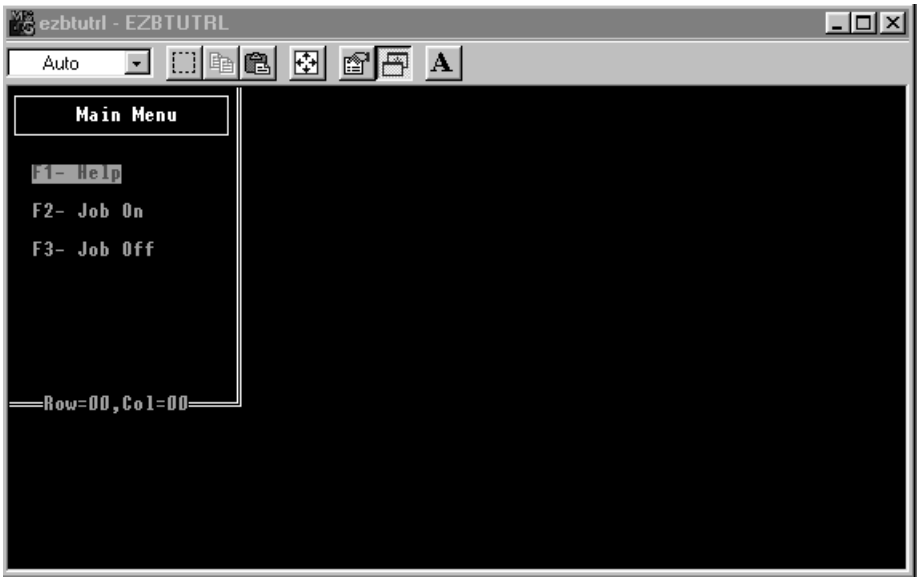
3. Click Close to close the dialog box.
4. You have now completed Task 13. When you are ready to run your program and test its logic, continue with Task 14.

Task 14: Testing the Program

After EZBuilder has built (compiled) your program, you run it using the EZBuilder Simulator to test how your program works.

To run (test) the program using the Simulator

1. From the Tools menu, choose the TRAKKER Antares Simulator command. Your result should look like the next illustration which shows a simulated display screen of the TRAKKER Antares terminal. The Main Menu is shown because it was the first menu you defined for EZBuilder.



2. You have several tests to make which you can choose to do in any order. The main thing to remember is to test everything you had coded. Some suggested tests are listed in the “Suggested Tests” section.

Note: *While testing your program on the JobOn and JobOff screens, use the* ***Esc*** *key to return to the Main Menu. You can exit your program at any time by using the* ***Ctrl-C*** *combination twice. This returns you to Windows 95.*

Note: *For online help, you can access the Simulator Editor, as described in the* EZBuilder Getting Started Guide.

Suggested Tests

The following are some suggested tests, to get you started. Depending on your options in coding the tutorial application (or your enhancements you made), you should think of other tests besides these. In testing a program, every branch, data field, action, screen, menu, transaction, and programmed function key should be thoroughly tested before the program is deemed ready for users.

1. From the Main Menu, test the use of the functions keys, **F1**, **F2**, and **F3**, on your keyboard to see if the program branches correctly to the MM_Help screen, the JobOn screen, and the JobOff screen, respectively. Test to see that **Esc** returns you to the Main Menu from the JobOn and JobOff screens.

   Note: *You can also use the up and down arrow keys on your keyboard and press the* ***Enter*** *key on your keyboard to get to the JobOn and JobOff screens from the Main Menu. Test this.*

2. From the JobOn screen, select the three data fields, one at a time, and use the appropriate **F1** function key to see if the Data_Help screen is presented to you for each data field.
3. From the JobOff screen, repeat the test described in Step 2.

   Note: *If you created the CopyBadge transaction, as soon as you entered the Badge ID Number for a JobOn transaction, it was automatically entered into the JobOff transaction's storage place for Badge ID Number. So, remember, you do not need to key it again for your JobOff transaction records. (Optionally, you may have coded the same logic for automatically moving Part Number and Order Number.)*

4. From the MM_Help screen, use the up and down arrow keys on your keyboard to test the scrolling section to see if you can read all of the help text you had entered. Check for any typing, logic, or semantic errors; if you find any, make a note to correct them later in EZBuilder and to recompile.
5. From the Data_Help screen, repeat the test described in Step 4.
6. On the JobOn screen, enter data for the three fields. Notice that when you completely fill the Badge field, it automatically goes to the Part field for data. Test with shorter Badge ID Number data and use **F4** or **Tab** to get to the Part field. (See the Note in Step 3 regarding automatic entry of data.)
7. Use **F4** to tab back to previous fields so you can re-enter data.

8. Test the Part Number field by entering 25 characters to see how it wraps to the next line. Also, test by entering shorter Part Number data and then use **F4** to get to the Order Number field and enter that data.
9. From the JobOn screen, test how you can use the forward / backward arrow keys to move to certain characters in the data to correct those characters.
10. When you finish testing and have made notes on how you want to enhance or correct your program, use **Ctrl-C** twice to exit the program.
11. Go to the location where your program files are kept (as per your Output search path) and see how many files were generated by EZBuilder.
12. List your JOFILE output file on your screen or a printer, and look at your JOTRAN and JOFTRAN records. Are they correct as per the data you entered? Did you plan and write down your test data before testing? If not, repeat the tests using the example data listed on the next page.

Congratulations! You have now completed the tutorial. Go back into EZBuilder and make any modifications you want to the program or begin a new program. Have fun experimenting with EZBuilder.

Note: *Only after you are satisfied with the accuracy of test results and have enhanced your program as needed should you make your program available for others to use on the terminal. This is done by downloading your executable program to the terminal. The downloading process is described as Task 15 later in this chapter.*

Example Data

When your program is run, (whether it is you testing your program with test data or the user executing the program with real data), the three input data items (Badge ID Number, Part Number, and Order Number) can be either be keyed in on your keyboard or scanned in with a terminal. The data is stored as output records and sent to a file which can be listed to your screen or printed, depending on your device capabilities.

The output file contains two types of transaction records: Job On transactions and Job Off transactions.

Record Identification Codes

The first item in each record is the six- or seven-letter identification code.

- "JOTRAN" identifies the record as having Job On transaction data.
- "JOFTRAN" identifies the record as having Job Off transaction data.

Aside from their identification code, the two record types are alike. Without the identification code, it is impossible to tell the two records apart.

Order of Data in the Record

The identification code, JOTRAN or JOFTRAN, always comes first and is followed by a comma, the Badge ID Number, a comma, the Part Number, a comma, the Order Number, a comma, the system Date, a space character, and the system Time. Except for the space between date and time, the data fields are run together, delimited by commas, as shown here by their field names:

```
IDCode,BadgeNo,PartNum,OrderNo,Date Time
```

A carriage return ends the record after the 8-character Time data. Thus, each record can be printed on one line in a listing.

Automatic Data Entered into the Record

Aside from user entered or scanned data for BadgeNo, PartNum, and OrderNo fields, some data (besides the identification code) is automatically attached to the record when the transaction is packaged (prepared for output).

- As soon as the user completes the Order Number for a transaction, the date, time, and carriage return character are automatically appended before the transaction record is sent to the output file.
- If the program includes the optional CopyBadge transaction (and perhaps CopyPart and CopyOrder transactions), as soon as the user enters BadgeNo data for a JobOn transaction (and PartNum and OrderNo, if appropriate), that data is also sent to its corresponding place(s) in the JobOff Transaction. This saves operator time and ensures accuracy.

Example Output Listing

You can list your output file and see one transaction record per line on the listing. The data fields will be in the same order as you had listed them in the JobOn and JobOff Transaction Fields lists.

An example listing of eight transactions follows. Notice that this listing is not real data, but is provided only as an example to show format. Your test data may be different.

```
JOTRAN,1234567890,1111133333444,2342345,04-23-1997 04:30:00
JOFTRAN,1234567890,1111133333444,2342345,04-23-199 04:45:02
JOTRAN,22222,2339AB-345473-XYZ,04-23-1997,5678990 04:48:00
JOFTRAN,22222,2339AB-345473-XYZ,04-23-1997,5678990 05:30:10
JOTRAN,35353,ASMCJK-234-88,04-23-1997,1526125 05:35:17
JOFTRAN,35353,ASMCJK-234-88,04-23-1997,1526125 07:02:00
JOTRAN,67612,22222333334444488,4837573,04-23-1997 08:15:01
JOFTRAN,67612,22222333334444488,4837573,04-23-1997 08:55:10
```

Task 15: Downloading Your Program

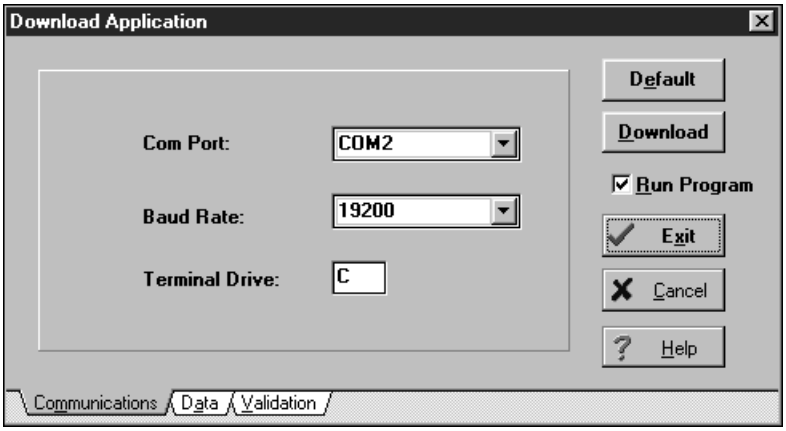
When your EZBuilder application program has been built, thoroughly tested, and is ready to download to the TRAKKER Antares terminal for users, make the physical connection and download as instructed below.

To prepare for a download

1. To make the physical connection, connect the TRAKKER Antares terminal to the development PC or host computer using the optical link adapter or TD2400 communications dock and cable. Be sure to note the computer's COM port to which you are connecting the cable.

   Note: *Refer to your TRAKKER Antares terminal user's manual or your TRAKKER Antares accessory documentation for complete details on the physical connection.*

2. Open your EZBuilder application. Be sure you know your filenames and directory paths.
3. From the Tools menu, choose the Download command. You will see the Download Application dialog box. Notice it has three tabs at the bottom: Communications (illustrated next), Data, and Validation, for three other dialog boxes.



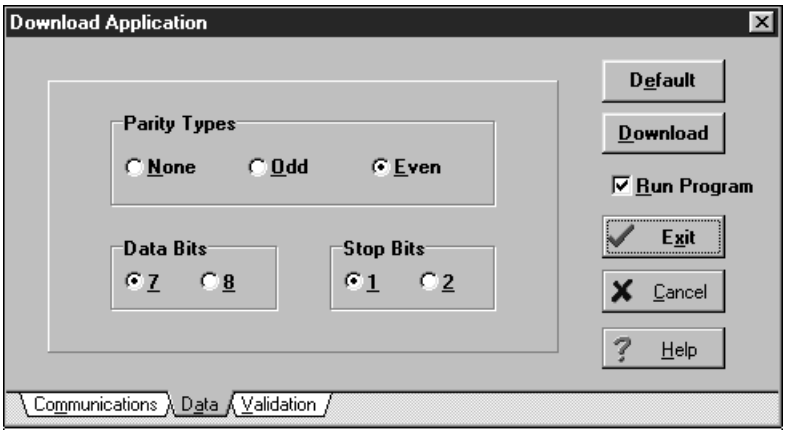
4. From the Communications dialog box, make sure the Communications settings shown on the screen and those shown on the TRAKKER Antares terminal are the same.
   - The COM2 port is listed in the above example. Be sure you know to which COM port on your computer cable is connected. Make sure your terminal setting matches the setting used by your computer.

- The default (19200) Baud Rate is shown in the example. The Baud Rate must match the setting on the terminal; change it if necessary.
- Terminal Drive C is the only drive from which you can run your EZBuilder application on the TRAKKER Antares terminal. Do not change this drive. (For more information, see the *EZBuilder Getting Started Guide*.)

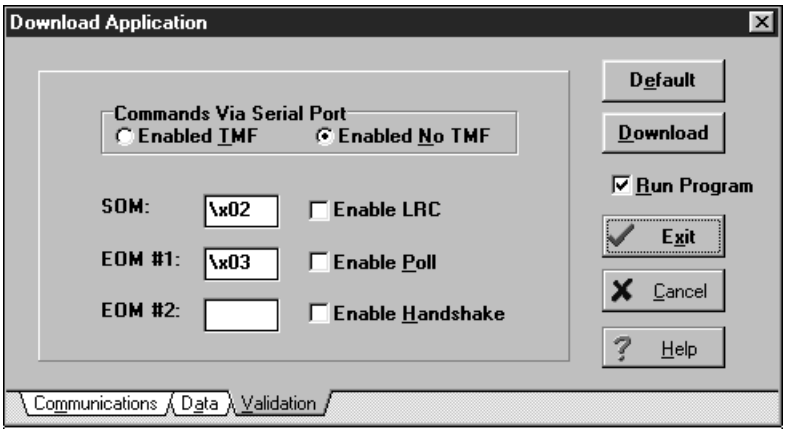
5. Notice that Default, Download, Exit, Cancel, and Help buttons are present on this dialog box. From this dialog box, you can also choose whether or not to run the program, following a download. Leave the box checked to run the program.

   Note: *These buttons are also available on the two other Download Application dialog boxes (Data and Validation) so you can review the settings in any order, then download and run your program when ready.*

6. Click the Data tab at the bottom of the dialog box so you can review the Parity Types setting, Data Bits, and Stop Bits, as illustrated next.



7. From the Data dialog box, make sure the Data settings shown on the screen and those shown on the TRAKKER Antares terminal are the same. Change them if necessary.
8. Click the Validation tab at the bottom of the dialog box so you can review the Validation settings, as illustrated next.

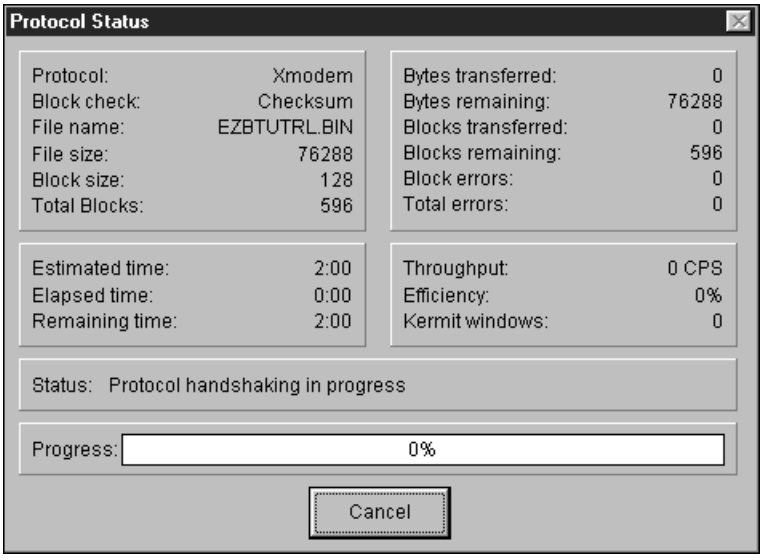


9. From the Validation dialog box, make sure the Validation settings shown on the screen and those shown on the TRAKKER Antares terminal are the same. Change them if necessary.

   Note: *If the terminal's Commands Via Serial Port shows "Disabled," the terminal must be changed to "Enabled" with or without TMF. Also, the EOM#1 must be set (not blank).*

To download EZBuilder applications and files

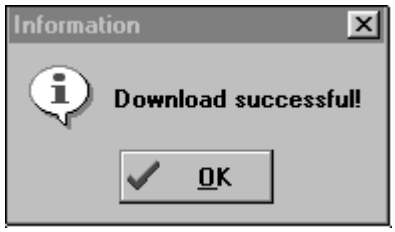
1. Notice, while verifying that computer and terminal settings match, that the Download button is conveniently included on each of the Download Application dialog boxes. Click Download when you have verified that the settings match. You should see the Protocol Status, as illustrated next.



2. Notice the percentage of progress toward completion of the download is shown near the bottom. It starts at 0% and should rapidly progress toward 100% unless you have a problem with the download. The beginning of this progress should happen within a few seconds.

   Note: *If you have no progress within abouta minute and a half, you will probably get an Protocol error message (see next section). If you recognize this situation quickly enough, press Cancel, correct the problem, and start your download again.*

3. If your download is successful, you will see the next illustration.



Congratulations! When your download is successful, you can test your program on the terminal in the same manner that you previously tested it on the computer.

Troubleshooting Error Messages

Error messages may occur when you attempt to download your application program to the terminal and your Download Application settings are incorrect.

Fatal Protocol error—Time out

If the Protocol Status (illustrated previously) remains at 0% and downloading progress does not begin within about ninety seconds, you will see the message illustrated next. You can avoid this message by clicking the Protocol Status's Cancel button before the timeout occurs. If your Protocol is incorrect, you will need to correct your COM port settings.

File Transfer Error—Cannot communicate

If you attempt to download and your computer and the terminal are not physically connected, you will see the message illustrated next. Make sure the cable is correctly attached and the COM port settings are correct.

Download canceled or system full

If you click Cancel on the Protocol Status dialog box or if the terminal file system is full, you will see the message illustrated next. Correct the reason for the cancellation before you attempt to download again.

Where Do You Go From Here?

After you have completed this tutorial and enhanced it to your satisfaction, you may want to develop your own application programs or you may want to learn more possibilities.

Using the Simulator Editor

Icons have been created for the development tools located in your Intermec folder. Look in that folder and note these two items:

- Simulator Editor. This allows you to make changes to the Simulator characteristics. For example, you may want to mimic host data (mostly network replies) to debug your applications.
- Simulator Editor Help. This provides help on how what Simulator characteristics can be changed using the Simulator Editor.

Using the Example Applications

Study the example applications that were included on the CD-ROM with your EZBuilder package. These will supply you with further EZBuilder features.

Refer to the *EZBuilder Getting Started Guide* for descriptions of the examples.

Using the Online Help

EZBuilder offers context-sensitive online help, as described in Chapter 1 of this tutorial. Just place your cursor over the EZBuilder item for which you want more information and press **F1**.

Now that you have some understanding of EZBuilder's basic features, open a new application and experiment. Use online help to direct you into some new features.