

Microsoft FORTRAN PowerStation 4.0 (MSFPS40)

Microsoft Corp stopped developing, supporting and selling their FORTRAN Powerstation compiler some time ago. It was still possible with TRNSYS 14.2 to use the Powerstation compiler to recreate the 32-bit trnlib.dll file needed to run TRNSYS. Unfortunately, the Powerstation compiler does not accept many of the TRNSYS features standard in version 15. It is highly recommended that you switch to either the Digital Visual Fortran (also no longer maintained) or Compaq Visual Fortran.

By excluding some of the standard TRNSYS 15 features, it is still possible to use the Powerstation 4.0 compiler. Below is a list of features that you will lose in doing so.

Type56 (multizone building)

Type61 (communication with non-fortran DLLs)

Type66 (communication with Engineering Equation Solver routines)

Communication with external programs

Setting up the Powerstation Project

Follow these instructions to get started using the MSFPS40 Developer Studio for handling the compiling, linking, and debugging of Fortran subroutines for TRNSYS.

1. Move the standard file `\trnsys15\kernel\exec.for` to a safe location. Move the `exec.for` file that you downloaded with this document into `\trnsys15\kernel\`
2. Start the MSFPS40 Developer Studio. Under the *File* menu, create a new *Project Workspace*. From the list of "Types of Projects" select Dynamic Link Library. Give the project a name. For example, you might name the project *32trnsys* and place it in the `c:\32trnsys` directory.
3. Under the *Insert* menu, click *Add Files to Project*. Add all of the Fortran files (*.for*) from the *Kernal* and *Types* subdirectories in the `/trnsys15` directory. Add the file *dumf40.lib* from the dummy subdirectory. Add your own custom components, if necessary.
4. You now need to remove a few files from the project. These are `userlib.for`, `callprogram.f90`, `Type61.for` and `Type66.for`.
5. Under the *Build* menu, click *Settings*. You will notice that there are numerous settings here controlled by a series of tabs. Below are the important settings for each of these tabs. Each tab can have several submenus, so there are many settings involved. Only the settings that must be changed from default values are described below.

First, there should be two items in the left box called (in this example): *32TRN - Win32 Debug* and *32TRN - Win32 Release*. Be sure you have the *Debug* version selected.

Debug Tab

General

Executable for session: c:\trnsys15\trnsys.exe

Working Directory: c:\trnsys15

Program Arguments: 2>\32trnsys\error.txt

Additional DLL's

Local Name: c:\trnsys15\trnlib.dll (note: if you select both items in the left "Settings for:" box, you cannot alter this category. You must select just the Debug project).

Fortran Tab

Fortran Language

Click box by Extended Bounds and Error Checking

Link Tab

General

Output File: c:\trnsys15\trnlib.dll

Object/library modules: blank

Check box "Generate debug info."

Customize

Output filename should now read c:\trnsys15\trnlib.dll

6. Under the *Build* menu, click *Rebuild All*. Watch the bottom window for errors and problems with compiling and creating the trnlib.dll output file. If everything worked correctly, the TRNSYS DLL was created in the \trnsys15 directory and TRNSYS can be run with the updated DLL. To run TRNSYS, select *Execute* from the *Build* menu. TRNSYS will prompt you for a TRNSYS input file.

7. To use the debugging features of MSFPS40, select *Debug* from the *Build* menu, then *Go*. At this point, the program will state, "TRNSYS.EXE does not have any debug information. OK?" Click OK because we are concerned with the Fortran DLL and not the TRNSYS executable. Without a breakpoint or a bug, TRNSYS will just start and run until the simulation is finished.

8. To halt execution at some point in the source code, insert breakpoints at one or more lines in the Fortran code of one or more subroutines. For example, open the file *trnsys.for* and create a breakpoint at the beginning of the source code, perhaps in the definitions of variables and common blocks. To create a breakpoint, put the cursor on a line of the source code and click F9. A red dot will appear to the left of the source code line. When TRNSYS is executing through the Debug feature of MSFPS40, it will always stop at this point in the code.

9. Experiment with the other debug commands by setting (and turning off) breakpoints with the F9 key, adding watch variables, stepping through the source code (F8), and running until you reach the cursor (F7). You can see the value of a variable simply by placing your cursor on top of that variable in the source code for a couple of seconds.

10. If your source code has a serious bug, such as trying to access an array outside of its boundaries, the debugger will simply stop and say that TRNSYS has exited with Code 1 (or something like that). At this point, access the file c:\32trnsys\error.txt. In our setup (the *Debug* tab), we have redirected the error output to this file since TRNSYS.EXE will not allow the debugger to write the error message (the line number of the error, for example) to the screen.