[ARC] Why does PowerView open a TERM.GATE window automatically?

2026-08-13 - Comments (0) - ARC

PowerView automatically opens a **TERM.GATE** window when it detects that your target application uses the MetaWare® hostlink library. This prevents the application from hanging before reaching `main()`.

By default, the MetaWare® C/C++ Compiler links the MetaWare hostlink library into the application. The hostlink library enables communication between the target application and the host PC running TRACE32, for example to support functions such as `printf`.

Before entering `main()`, an application linked with the hostlink library waits in a polling loop to receive the `argc` and `argv` parameters from the host PC. If the debugger does not interact with the target, the application remains in this loop indefinitely.

To enable hostlink support in TRACE32, execute the following commands after loading the ELF file:

```
TERM.METHOD METAWARE
TERM.GATE
```

When you start the target with the TRACE32 `Go` command, PowerView checks whether the ELF file contains the special symbol `__HOSTLINK__`, which normally indicates that the MetaWare hostlink library is present. If this symbol is found but no suitable `TERM.GATE` window is open, PowerView opens one automatically.

In rare cases, this detection can produce a false positive. For example, if your application defines a symbol named `__HOSTLINK__` for another purpose, PowerView assumes that the MetaWare hostlink library is being used and opens a `TERM.GATE` window even though it is not required.

How can I prevent TERM.GATE from opening automatically?

You have several options:

- If your application uses the MetaWare hostlink library, explicitly enable hostlink support after loading the ELF file:

  ```
  TERM.METHOD METAWARE
  TERM.GATE
  ```

- If your application does **not** require the hostlink library, exclude it during linking by using the compiler option:

  ```
  -Hhostlib=
  ```

- Rename your application's `__HOSTLINK__` symbol to a different name.

- Rename the symbol in TRACE32 after loading the ELF file, for example:

  ```
  sYmbol.Modify.NAME __HOSTLINK__ __HOSTLINK-OFF__
  ```

For additional information, refer to the *MetaWare® C/C++ Programmer's Guide for ARC®*