



Ampex/Perigram Corporation

TN #41: CD-I Application Library/Trap Handler Usage

Written by: Eric Anderson

Date: June 21, 1989

This technical note describes the use of trap handlers found in CD-RTOS ROM. Using trap handlers allows you to save application memory. Additional memory can be saved by removing stack checking code.

Command Line Options With math And cio Trap Handlers

CD-RTOS contains both `math` and `cio` trap handlers embedded in ROM. Therefore, to make use of these trap handlers, you should link your application by specifying the appropriate command line options. Thus, you can save program memory and take advantage of ROM-based functionality that has been optimized for the native hardware.

Typically, libraries are available in two types:

- 1) libraries that implement functionality directly
- 2) libraries that assume the existence of a trap handler

In both cases, external references to functions are correctly resolved by the linker. However the resulting application built using the first type of library will be larger than an application built using the second type. In tight memory configurations, the goal is the smallest application possible.

For example:

- The `-x` option forces the linker to use the `math` trap handler. It uses the library `clib.l`. Failure to specify this option forces the linker to use `clibn.l` (note the "n") and to obtain `math` routines from the library `math.l`.
- The `-i` option forces the linker to use the `cio` trap handler. It uses the library `cio.l` along with either `clibn.l` or `clib.l`. Library `cio.l` implements several, but not all of the 220 standard C functions. The 65 functions that are implemented in the `cio` trap handler are listed on page 1-13 of the OS-9/68000 C Compiler User's Manual.

Using both `-x` and `-i` options will invoke both the `cio` and `math` trap handlers. This is the recommended usage for CD-I applications.

Other Command Line Options

Additional memory can be saved by specifying the `-s` command line option. This option removes stack check code from the application, thereby saving memory and increasing call overhead. The OS-9/68000 C Compiler User's Manual recommends that this option be used with great care in cases where stack memory requirements are well understood and also in time critical applications.