

B UTILITIES

Overview

Your Analog Devices development software comes with several file conversion utilities, which run from a command line only. Some of these utilities provide support for legacy code, and others are intended for a group of users who prefer to use the command-line version of the tools instead of using them through the VisualDSP++ environment.

This appendix describes each of these utilities and describe the command lines for these tools.

Dumper — ELF File Dumper

The ELF file dumper (`elfdump.exe`) extracts data from ELF executable files (.DXE) and provides a text output file that describes the ELF file's contents. The ELF file dumper uses the following command line:

```
C:\Program Files\Analog Devices\VisualDSP>elfdump
```

Usage: `elfdump {option} {filename}`

Table B-1. ELF File Dumper Command-Line Option Switches

Switch	Description
-fh	Print a file header.
-arsym	Print the archive symbol table.

Table B-1. ELF File Dumper Command-Line Option Switches (Cont'd)

Switch	Description
-a r a l l	Print every archive member.
-p h	Print the program header table.
-s h	Print the section header table. The default is -s h if no other options are specified.
-n <i>section</i>	Print contents of the named section(s). Section name may use ? and * wild card characters. Each section's name and type determine its output format unless overridden by a modifier (see the <i>filename</i> description).
-i x0[-x1]	Print contents of the sections numbered x0 through x1, where x0 and x1 are decimal integers, and x1 defaults to x0 if omitted. Formatting rules as are for -n.
-a l l	Print everything. Same as -f h -p h -s h -n o t e s -n '*'
-o s t	Omit string table sections.
<i>filename</i>	File whose contents are to be printed. It can be a core file, executable, shared library, or relocatable object file. If the name is in the form A(B), A is assumed to be an archive and B is an ELF element in the archive. B can use a pattern like the one accepted by -n. The -n and -i options can have a modifier letter after the main option character, which forces section contents to be formatted in the following ways: - a Dump contents in hex and ASCII format, 16 bytes per line. - x Dump contents in hex format, 32 bytes per line. - xN Dump contents in hex format, N bytes per group (default is N=4). - t Dump contents in hex format, N bytes per line, where N is the section's table entry size. If N is not in the range 1 . . 32, 32 is used. - i Print contents as list of disassembled machine instructions.

Using the Archiver and Dumper For Disassembly

The file utilities are each useful in their own way, but can become much more effective when you combine their capabilities. One interesting application of these utilities is to disassemble a library member, converting it to source code. This application is good to have around when you discover your source for a particularly useful routine has "disappeared" and is only available as a library routine.

The following procedure lists the objects in a library, extracts an object, and converts the object to a listing file. Using the following archiver command line, list the objects in the library and write the output to a text file:

```
elfar -p libc.dlb > libc.txt
```

Assuming the current directory is

```
C:\Program Files\Analog Devices\VisualDSP++\TS001\lib>
```

open the text file, scroll through it, and find the object file that you need. Then, use the following archiver command line to extract the object from the library:

```
elfar -e libc.dlb fir.doj
```

To convert the object file to an assembly listing file with labels (almost source, just needs the line numbers and opcodes removed), use the following `elfdumper` command line:

```
elfdump -ns * fir.doj > fir.asm
```

Using disassembly, you get a listing file with symbols. Assemble source with symbols can be useful if you are familiar with the code and, hopefully, have some documentation on what the code does. If symbols were stripped during linking, there are no symbols in the dumped file.



Using disassembly on a third party's library may violate the license for the third party's software. Check copyright and license issues with the code's owner before using this disassembly technique.

Dumping Overlay Archive Files

Use the `elfar` and `elfdump` commands to extract and view the contents of the overlay archive file (*.OVL).

For example,

```
elfar -P CLONE2.OVL
```

will print the contents to `CLONE2.ELF` which will be an `.ELF` file that can be viewed with `elfdump`.

To view the `CLONE2.ELF` file, enter

```
elfdump -all CLONE2.OVL(CLONE2.elf)
```

To extract `CLONE2.ELF` and dump, enter

```
elfar -e CLONE2.elf
```

```
elfdump -all CLONE2.elf
```

or use whatever `elfdump` options you wish.

These commands are case-sensitive.