# Talaria TWO Software Crash

When the application on Talaria TWO crashes, a coredump file can be generated. Coredump captures the working memory of the application and firmware and generates a coredump file that is used for analyzing the cause of the crash.

Following method is used to collect the coredump:

Assuming Talaria TWO has already crashed for a particular reason and is currently in a crashed state, open a new terminal window in Linux and execute the commands mentioned in the following section to collect core dump.

## Generate Coredump from within GDB

1. Execute the following command in VM UBUNTU Linux shell

|  |
| --- |
| echo "set auto-load safe-path /" > ~/.gdbinit |

1. Change directory to conf folder in SDK

|  |
| --- |
| cd ~/sdks/sdk\_x.y |

1. Execute the following command from the same shell as mentioned in step 2

|  |
| --- |
| openocd -s ./conf -f ftdi.cfg -f t2.cfg |

1. Let the openocd run in the current shell and open a new Linux shell to execute the following. The output of the openocd execution is shown in Figure 1.

Text

Description automatically generated

Figure 1: Terminal output

1. Execute the following command from apps folder

**Note**: The default elf given with the SDK package under bin will be a stripped one and this cannot be used for coredump analysis. Hence, do a make for the application which needs to be tested and use the elf from the out folder.

|  |
| --- |
| gdb-multiarch ./my\_test.elf |

1. gdb session will start now and execute the command ocd in GDB session.

|  |
| --- |
| (gdb) ocd |

1. Collect the core dump using the following command:

|  |
| --- |
| (gdb) coredump coredump\_any\_filename.bin |

1. Core dump is collected in coredump\_any\_filename.bin.
2. Core dump file can be opened using xxd editor to check the content\ from the Linux shell.

|  |
| --- |
| xxd coredump\_any\_filename.bin |

Output for the sample elf is as shown in Figure 2.

Text

Description automatically generated

Figure 2: Opening the coredump file