

How to profile an executable on hardware

IN THIS DOCUMENT

- ▶ Profiling from within xTIMEcomposer Studio
 - ▶ Profiling from the command line
-

| | |
|-------------|--|
| version | 1.1.0 |
| scope | Example. This code is provided as example code for a user to base their code on. |
| description | How to profile an executable on hardware |
| boards | Unless otherwise specified, this example runs on the SliceKIT Core Board, but can easily be run on any XMOS device by using a different XN file. |

The xTIMEcomposer tools provide support for generating a GNU profiler (gprof) compatible output file. This file contains the instruction timing information, also known as the flat profile. As an example, compile the following code, ensuring that the generation of debug info (-g) is enabled:

```
#include <math.h>

int main() {
    int i;
    float x = 0;
    for (i = 0; i < 100000; ++i) {
        x += sin(x);
    }
    return 0;
}
```

1 Profiling from within xTIMEcomposer Studio

Create a new run configuration for the chosen target. To enable profiling output, select the Run configuration and check the *Enable GPROF collection* box on the *Main* tab. On execution, a number of .gprof files (one per core) will be created at the top level of the project. Double click on the file of interest to load it into the *gprof* view, which allows the timing of the program to be analyzed on a source line-by-line basis.

2 Profiling from the command line

Run the executable on hardware, use the *-gprof* command line switch:

```
xrun --gprof a.xe
```

This will produce a number of .gprof files, one per core. To analyze the profile from the command line, a version of the *gprof* command line tool is required. *gprof* accepts as input both the generated profile (.gprof) file and the relevant ELF file. The ELF files can be extracted from the XE file as follows:

```
xobjdump --split a.xe
```

gprof can then be run from the command line in the following way:

```
gprof image_n0c0.elf tile[0]_core0.gprof
```

The above will generate report for the code running on core 0.



Copyright © 2013, All Rights Reserved.

Xmos Ltd. is the owner or licensee of this design, code, or Information (collectively, the "Information") and is providing it to you "AS IS" with no warranty of any kind, express or implied and shall have no liability in relation to its use. Xmos Ltd. makes no representation that the Information, or any particular implementation thereof, is or will be free from any claims of infringement and again, shall have no liability in relation to any such claims.