Application Note: AN00196

Getting Started with Real-Time xSCOPE in xTIMEcomposer Studio

This application note shows how to get started with real-time xscope using the xTIMEcomposer studio. It shows you how to run an application with xscope logging and view the resulting information about the application as it is running. It then explains some of the features of the real-time xSCOPE view.

To get started, simply double click on *Getting Started with Real-Time xSCOPE in xTIMEcomposer Studio* in the Examples view, and click finish in the resulting import dialog. The sample project will then be imported and you will be switched to the *XMOS Edit* perspective. The getting started pdf is then accessable from the *doc/pdf* folder at the top level of the imported project.

Required tools and libraries

- xTIMEcomposer Tools - Version 14.0

Required hardware

This application note is designed to run on any XMOS multicore microcontroller or the XMOS simulator.

Prerequisites

None

1 Overview

1.1 Introduction

Our comprehensive development tools suite provides everything you need to write, debug and test applications based on xCORE multicore microcontrollers. The full xTIMEcomposer tool set includes unique capabilities such as the xSCOPE logic analyzer and XMOS Timing Analyzer, that let you get the best performance from the deterministic xCORE architecture. With our collection of libraries and examples, it's easy to create and deliver xCORE applications.

xTIMEcomposer features:

- Eclipse graphical environment + plus command line tools
- LLVM C, C++ and xC compilers
- xDEBUG: GDB multicore debugger
- xSIM: Cycle accurate simulator
- xSCOPE: In-circuit instrumentation + real-time logic analyzer
- XTA: Static timing analysis
- Multiple platform support: Windows, OS X, Linux
- Enterprise/Community editions: Tools support for everyone

This application note shows how to get started with real-time xscope using the xTIMEcomposer studio. It shows you how to run an application with xscope tracing and view the resulting information about the application as it is running. It then explains some of the features of the real-time xSCOPE view.

2 Getting Started

Ensure you are in the edit perspective by clicking on the *Edit* perspective button on the left hand side toolbar.

In the *Project Explorer* view you will see the *app_getting_started_with_real_time_xscope* project. Open *src/main.xc* by double clicking.

This application contains a sine wave data table. It loops indefinately, using xscope to output the content of this table. A single probe, named *Value* is configured. See the *config.xcsope* file at the root of the project for it's defintion. Note: The xSCOPE system is enabled via the *enabled* attribute on the *xSCOPEConfig* element.

2.1 Build the application

To build the application, select 'Project -> Build Project' in the menu, or click the 'Build' button on the toolbar. The output from the compilation process will be visible on the console.

2.2 Create and launch a run configuration

To view xSCOPE data in real-time you first need to create a run configuration. The xTIMEcomposer allows multiple configurations to exist, thus allowing you to store configurations for running on different targets/with different runtime options and arguments.

Right-click on the generated binary in the *Project Explorer* view, and select *Run As* -> *Run Configurations*. In the resulting dialog, double click on *xCORE Application*, then perform the following operations:

- On the *Main* tab select the desired target, or check the *simulator* option.
- On the *XScope* tab, check *Real-Time Mode*.

Now select the *Run* button to launch the application. The perspective will automatically switch to *Realtime xSCOPE*, and the value of the probe will be displayed in real-time in the *Display* view. As expected, you will see the sine wave being generated. Click on the *Auto* button on the *Control* view to centre and scale the signal in the display. The *Control* view can now be used to pause/continue the updating of the display and set trigger points.

2.3 Setting a trigger point

Just like hardware scopes, the xSCOPE real-time view allows the setting of trigger points. This allows the signal to be frozen when the value of the signal crosses a user specified threashold. To set a trigger point, check the empty square next to the *Value* probe in the *Control* view. The tigger point is shown by the cross hairs in the *Display* view. To move this, simply left-click in this view.

3 References

XMOS Tools User Guide

http://www.xmos.com/published/xtimecomposer-user-guide

XMOS xCORE Programming Guide

http://www.xmos.com/published/xmos-programming-guide

4 Full source code listing

4.1 Source code for main.xc

```
// Copyright (c) 2015, XMOS Ltd, All rights reserved

#include <xscope.h>

#define NUM_SAMPLES 64

unsigned int sine[NUM_SAMPLES] = {
        4000,4392,4780,5161,5531,5886,6222,6538,6828,7092,7326,7528,7696,7828,7923,7981,
        8000,7981,7923,7828,7696,7528,7326,7092,6828,6538,6222,5886,5531,5161,4780,4392,
        4000,3608,3220,2839,2469,2114,1778,1462,1172,908,674,472,304,172,77,19,
        0,19,77,172,304,472,674,908,1172,1462,1778,2114,2469,2839,3220,3608,
};

int main() {
    while (1) {
        for (unsigned int i = 0; i < NUM_SAMPLES; ++i) {
            xscope_int(VALUE, sine[i]);
        }
    }
    return 0;
}
```