Application Note: AN10085

How to use the xSCOPE continuous event type

This application note is a short how-to on programming/using the xTIMEcomposer tools. It shows how to use the xSCOPE continuous event type.

Required tools and libraries

This application note is based on the following components:

- xTIMEcomposer Tools - Version 14.0.0

Required hardware

Programming how-tos are generally not specific to any particular hardware and can usually run on all XMOS devices. See the contents of the note for full details.

1 How to use the xSCOPE continuous event type

xSCOPE is fully supported on hardware platforms which provide an XMOS link between the target device and the XSYS development connector, it is also supported using xSIM.

View the document (xSCOPE overview (see XM-000957-PC)) for further information on tracing data from XMOS applications.

This example provides a simple demonstration of using the xSCOPE continuous event type for data logging from within an XMOS application. The continuous event type can be used to capture and log the value of specific variables within an application to allow debugging.

The probe configuration is handled by the user providing a config.xscope file which is picked up as part of the application build.

This example assumes you are familiar with creating a run configuration and enabling the associated xSCOPE options in that run configuration in xTIMEcomposer Studio or using the command line tools.

In order to used xSCOPE the correct header file must be included in the application

```
#include <xscope.h>
```

The xscope_int() function is used to send the contents of user variable data_value_1 to xSCOPE probe CONTINUOUS_VALUE_1 for logging

```
void output_data_1(unsigned int data_value_1) {
  xscope_int(CONTINUOUS_VALUE_1, data_value_1);
}
```

The xscope_int() function is used to send the contents of user variable data_value_2 to xscope probe CONTINUOUS_VALUE_2 for logging

```
void output_data_2(unsigned int data_value_2) {
  xscope_int(CONTINUOUS_VALUE_2, data_value_2);
}
```