Application Note: AN10086

How to use the xSCOPE discrete event type

This application note is a short how-to on programming/using the xTIMEcomposer tools. It shows how to use the xSCOPE discrete 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 discrete 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 via 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 discrete event type for data logging from within an XMOS application. The discrete event type is used for recording the value of a captured variable for a length of time before that value is changed. This lets you see the level of a particular variable over time and the points at which it has been changed by the application.

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>
```

Declare a variable you are going to trace as an xSCOPE discrete event type

```
unsigned int memory_used = 0;
```

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

```
xscope_int(MEMORY_ALLOCATED, memory_used);
```