

Application Note: AN10096

How to use the xSCOPE state machine event type

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

xSCOPE is fully supported on hardware platforms which provide an XMOS link between the target device and the XSYS development connector, 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 state machine event type for data logging from within an xCORE application. The state machine event type is used for recording the value of a variables associated with state change within an application. This allows a user to see how long the device has spent in specific states for timing and debugging purposes.

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 state machine event type

```
unsigned int current_state = 1;
```

The xscope_probe_data() function is used to send the contents of user variable current_state to xSCOPE probe STATE_TRANSITIONS for logging

```
xscope_int(STATE_TRANSITIONS, current_state);
```