xSCOPE Library API

IN THIS DOCUMENT

- Functions
- Enumerations

1 Functions

```
void xscope_bytes(unsigned char id,
                  unsigned int size,
                  const unsigned char data[])
```

Send a trace event for the specified XSCOPE probe with a byte array.

This function has the following parameters:

`id`	XSCOPE probe id.
`size`	User data size.
`data`	User data bytes (char[]).

```
void xscope_char(unsigned char id, unsigned char data)
```

Send a trace event for the specified XSCOPE probe of type char.

This function has the following parameters:

`id`	XSCOPE probe id.
`data`	User data value (char).

```
void xscope_config_io(xscope_IORedirectionMode mode)
```

Configures XScope I/O redirection.

This function has the following parameters:

`mode`	I/O redirection mode.

```
void xscope_core_bytes(unsigned char id,
                       unsigned int size,
                       const unsigned char data[])
```

Send a trace event for the specified XSCOPE probe with a byte array with logical core info.

This function has the following parameters:

XMOS © 2013, All Rights Reserved

`id` XSCOPE probe id.

`size` User data size.

`data` User data bytes (char[]).

`void xscope_core_char(unsigned char id, unsigned char data)`

Send a trace event for the specified XSCOPE probe of type char with logical core info.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (char).

`void xscope_core_double(unsigned char id, double data)`

Send a trace event for the specified XSCOPE probe of type double with logical core info.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (double).

`void xscope_core_float(unsigned char id, float data)`

Send a trace event for the specified XSCOPE probe of type float with logical core info.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (float).

`void xscope_core_int(unsigned char id, unsigned int data)`

Send a trace event for the specified XSCOPE probe of type int with logical core info.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (int).

`void xscope_core_longlong(unsigned char id, unsigned long long data)`

Send a trace event for the specified XSCOPE probe of type long long with logical core info.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (long long).

`void xscope_core_short(unsigned char id, unsigned short data)`

Send a trace event for the specified XSCOPE probe of type short with logical core info.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (short).

`void xscope_core_start(unsigned char id)`

Start a trace block for the specified XSCOPE probe with logical core info.

This function has the following parameters:

`id` XSCOPE probe id.

`void xscope_core_start_int(unsigned char id, unsigned int data)`

Start a trace block for the specified XSCOPE probe with logical core info and capture a value of type int.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (int).

`void xscope_core_stop(unsigned char id)`

Stop a trace block for the specified XSCOPE probe with logical core info.

This function has the following parameters:

`id` XSCOPE probe id.

`void xscope_core_stop_int(unsigned char id, unsigned int data)`

Stop a trace block for the specified XSCOPE probe with logical core info and capture a value of type int.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (int).

`void xscope_disable()`

Disable the XSCOPE event capture on the local xCORE tile.

`void xscope_double(unsigned char id, double data)`

Send a trace event for the specified XSCOPE probe of type double.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (double).

`void xscope_enable()`

Enable the XSCOPE event capture on the local xCORE tile.

`void xscope_float(unsigned char id, float data)`

Send a trace event for the specified XSCOPE probe of type float.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (float).

`void xscope_int(unsigned char id, unsigned int data)`

Send a trace event for the specified XSCOPE probe of type int.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (int).

`void xscope_longlong(unsigned char id, unsigned long long data)`

Send a trace event for the specified XSCOPE probe of type long long.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (long long).

`void xscope_ping()`

Generate an XSCOPE ping system timestamp event.

`void xscope_register(int num_probes, ...)`

Registers the trace probes with the host system.

First parameter is the number of probes that will be registered. Further parameters are in groups of four.

Examples:

```
xscope_register(1, XSCOPE_DISCRETE, "A probe", XSCOPE_UINT, "value");
  ↪ ``
xscope_register(2, XSCOPE_CONTINUOUS, "Probe", XSCOPE_FLOAT, "Level",
                   XSCOPE_STATEMACHINE, "State machine", XSCOPE_NONE, "
                     ↪ no name");
```

This function has the following parameters:

`num_probes` Number of probes that will be specified.

`void xscope_short(unsigned char id, unsigned short data)`

Send a trace event for the specified XSCOPE probe of type short.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (short).

`void xscope_start(unsigned char id)`

Start a trace block for the specified XSCOPE probe.

This function has the following parameters:

`id` XSCOPE probe id.

`void xscope_start_int(unsigned char id, unsigned int data)`

Start a trace block for the specified XSCOPE probe and capture a value of type int.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (int).

`void xscope_stop(unsigned char id)`

Stop a trace block for the specified XSCOPE probe.

This function has the following parameters:

`id` XSCOPE probe id.

`void xscope_stop_int(unsigned char id, unsigned int data)`

Stop a trace block for the specified XSCOPE probe and capture a value of type int.

This function has the following parameters:

`id` XSCOPE probe id.

`data` User data value (int).

2 Enumerations

`xscope_IORedirectionMode`

Enum of all I/O redirection modes.

This type has the following values:

`XSCOPE_IO_NONE`
: I/O is not redirected.

`XSCOPE_IO_BASIC`
: Basic I/O redirection.

`XSCOPE_IO_TIMED`
: Timed I/O redirection.

`XSCOPE_IO_NONE`
: I/O is not redirected.

`XSCOPE_IO_BASIC`
: Basic I/O redirection.

`XSCOPE_IO_TIMED`
: Timed I/O redirection.

`XSCOPE_IO_NONE`
: I/O is not redirected.

`XSCOPE_IO_BASIC`
: Basic I/O redirection.

`XSCOPE_IO_TIMED`
: Timed I/O redirection.

`xscope_UserDataType`

Enum for all user data types.

This type has the following values:

`XSCOPE_NONE` No user data.

`XSCOPE_UINT` Unsigned int user data.

`XSCOPE_INT` Signed int user data.

`XSCOPE_FLOAT`
: Floating point user data.

`XSCOPE_NONE` No user data.

`XSCOPE_UINT` Unsigned int user data.

`XSCOPE_INT` Signed int user data.

`XSCOPE_FLOAT`
Floating point user data.

`XSCOPE_NONE` No user data.

`XSCOPE_UINT` Unsigned int user data.

`XSCOPE_INT` Signed int user data.

`XSCOPE_FLOAT`
Floating point user data.

`xscope_EventType`

Enum for all types of xscope events.

This type has the following values:

`XSCOPE_STARTSTOP`
Start/Stop - Event gets a start and stop value representing a block of execution.

`XSCOPE_CONTINUOUS`
Continuous - Only gets an event start, single timestamped "ping".

`XSCOPE_DISCRETE`
Discrete - Event generates a discrete block following on from the previous event.

`XSCOPE_STATEMACHINE`
State Machine - Create a new event state for every new data value.

`XSCOPE_STARTSTOP`
Start/Stop - Event gets a start and stop value representing a block of execution.

`XSCOPE_CONTINUOUS`
Continuous - Only gets an event start, single timestamped "ping".

`XSCOPE_DISCRETE`
Discrete - Event generates a discrete block following on from the previous event.

`XSCOPE_STATEMACHINE`
: State Machine - Create a new event state for every new data value.

`XSCOPE_HISTOGRAM`

`XSCOPE_STARTSTOP`
: Start/Stop - Event gets a start and stop value representing a block of execution.

`XSCOPE_CONTINUOUS`
: Continuous - Only gets an event start, single timestamped "ping".

`XSCOPE_DISCRETE`
: Discrete - Event generates a discrete block following on from the previous event.

`XSCOPE_STATEMACHINE`
: State Machine - Create a new event state for every new data value.

`XSCOPE_HISTOGRAM`

XMOS®

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.