lib_logging: Debug Printing

XMOS

1 Introduction

This library provides a lightweight printf function that can be enabled or disabled via configuration defines. Code can be declared to be within a "debug unit" (usually a library or application source base) and prints can be enabled/disabled per debug unit.

lib_logging is intended to be used with the XCommon CMake , the *XMOS* application build and dependency management system.

2 API

To use this module, include **lib_logging** in the application's **APP_DEPENDENT_MODULES** list and include the **debug_print.h** header file.

void **debug_printf**(char fmt[], ...)

A limited functionality version of printf that is low memory.

This function works like C-standard printf except that it only accepts d, x, s, u and c format specifiers with no conversions.

The p format specifier is treated the same as a x.

The capital version of each format specifier performs the same as the lower case equivalent.

Any alignment or padding characters are simply ignored.

The function uses the functions from **print.h** to do the underlying printing.

Unlike printf this function has no return value.

Whether the function does any output can be controlled via defines such as **DEBUG_PRINT_ENABLE** or **DEBUG_PRINT_ENABLE_[debug unit name]** in the application's **debug_conf.h**

3 Debug units

A source file can be added to a debug unit by defining the **DEBUG_UNIT** macro before inclusion of **debug_print.h**. For example:

```
#define DEBUG_UNIT ETHERNET_MODULE
#include "debug_print.h"
```

To include all source files in a module in a particular debug unit, it is convenient to do it in the **lib_build_info.cmake** file of the module e.g.:

```
set(LIB_COMPILER_FLAGS ... -DDEBUG_UNIT=ETHERNET_MODULE ...)
```

If no **DEBUG_UNIT** is defined then the default debug unit is **APPLICATION**.

4 Enabling printing

By default, debug printing is turned *off*. To enable printing you need to pass the correct command line option to compilation. The following defines can be set by using the **-D**

option to the compiler. For example, the following in your application `CMakeLists.txt` will enable debug printing:

```
set(APP_COMPILER_FLAGS ... -DDEBUG_PRINT_ENABLE=1 ...)
```

The following defines can be set:

DEBUG_PRINT_ENABLE

Setting this define to 1 or 0 will control whether debug prints are output.

DEBUG_PRINT_ENABLE_[debug unit]

Enabling this define will cause printing to be enabled for a specific debug unit. If set to 1, this will override the default set by `DEBUG_PRINT_ENABLE`.

DEBUG_PRINT_DISABLE_[debug unit]

Enabling this define will cause printing to be disabled for a specific debug unit. If set to 1, this will override the default set by `DEBUG_PRINT_ENABLE`.

XMOS