xSOFTip Explorer Tutorial

IN THIS DOCUMENT

1 Introduction

Welcome to the XMOS xSOFTip Explorer, which lets you browse the XMOS xSOFTip components and select them for use in your system.

This tutorial shows you how to:

- Create an application in the xSOFTip Explorer
- Identify the system resources required by the application
- Change a configuration
- Identify a suitable xCORE device for your application

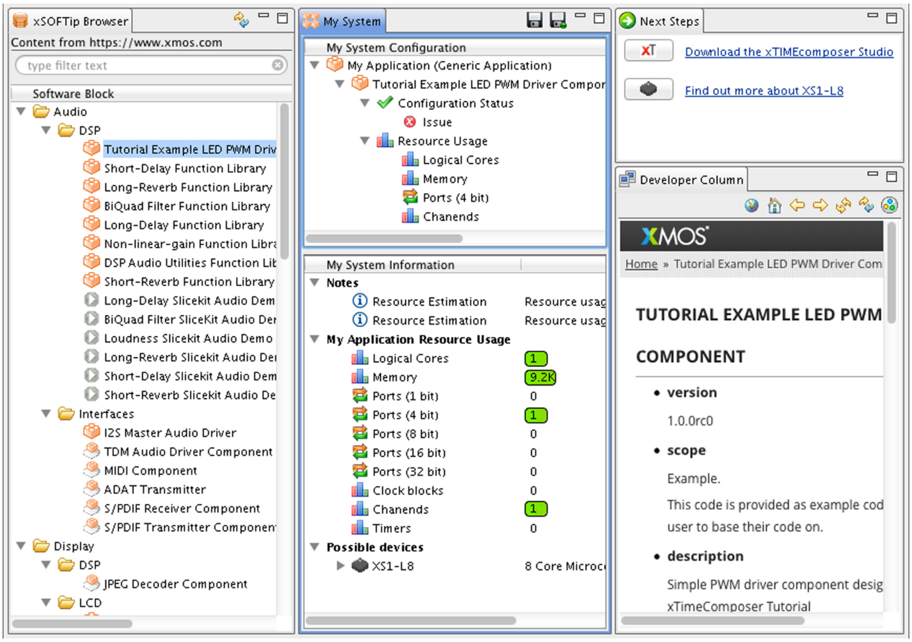
2 Developing applications with xSOFTip Explorer

This tutorial shows you how to use the explorer to select xSOFTip for your first XMOS multicore microcontroller project.

2.1 The xSOFTip Explorer interface

The xSOFTip Explorer has five windows:

Publication Date: 2013/5/29 REV B



Content from https://www.xmos.com

Software Block
- Audio
  - DSP
    - Tutorial Example LED PWM Driv
    - Short-Delay Function Library
    - Long-Reverb Function Library
    - BiQuad Filter Function Library
    - Long-Delay Function Library
    - Non-linear-gain Function Libra
    - DSP Audio Utilities Function Lib
    - Short-Reverb Function Library
    - Long-Delay Slicekit Audio Dem
    - BiQuad Filter SliceKit Audio Der
    - Loudness Slicekit Audio Demo
    - Long-Reverb Slicekit Audio Der
    - Short-Delay Slicekit Audio Dem
    - Short-Reverb Slicekit Audio De
  - Interfaces
    - I2S Master Audio Driver
    - TDM Audio Driver Component
    - MIDI Component
    - ADAT Transmitter
    - S/PDIF Receiver Component
    - S/PDIF Transmitter Componen
- Display
  - DSP
    - JPEG Decoder Component
  - LCD

My System Configuration
- My Application (Generic Application)
  - Tutorial Example LED PWM Driver Compor
    - Configuration Status
      - Issue
    - Resource Usage
      - Logical Cores
      - Memory
      - Ports (4 bit)
      - Chanends

My System Information

Notes	
Resource Estimation	Resource usag
Resource Estimation	Resource usag
My Application Resource Usage	
Logical Cores	1
Memory	9.2K
Ports (1 bit)	0
Ports (4 bit)	1
Ports (8 bit)	0
Ports (16 bit)	0
Ports (32 bit)	0
Clock blocks	0
Chanends	1
Timers	0
Possible devices	
XS1-L8	8 Core Microc

Download the xTIMEcomposer Studio

Find out more about XS1-L8

Home » Tutorial Example LED PWM Driver Com

TUTORIAL EXAMPLE LED PWM

COMPONENT

- **version**

  1.0.0rc0
- **scope**

  Example.

  This code is provided as example cod
  user to base their code on.
- **description**

  Simple PWM driver component desig
  xTimeComposer Tutorial

- **xSOFTip Browser**: shows the xSOFTip components you can chose for your project
- **My System Configuration**: the xSOFTip components you have selected
- **My System Information**: the resources used by the xSOFTip components you have selected, and the XMOS multicore microcontrollers which suit your application
- **Next Steps**: information and activities we recommend you try next
- **Developer Column**: online documentation about the xSOFTip, tools and xCORE multicore microcontrollers. The *Developer Column* can have multiple tabs. While running this tutorial, you can swap to the Main tab to view the documentation for the xSOFTip components.

3 Create your first application

The xSOFTip Browser displays the available xSOFTip components. xSOFTip blocks use xCORE resources to provide interfacing, DSP, protocols and control functions, allowing you to concentrate on building your application.

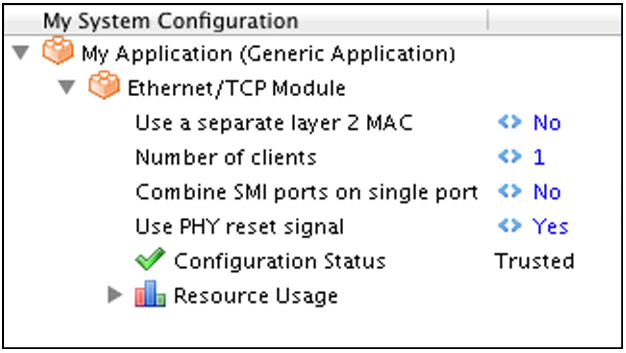
3.1 Add xSOFTip to your configuration

Choose some xSOFTip for your first application.

1. Select the **Ethernet/TCP** xSOFTip from the *Networking>Ethernet* category.

The *Developer Column* shows information about this component, including a description of what it does, its features and which xKIT development kits are suitable for use with this xSOFTip. You will need to switch to the *Main* tab at the bottom of the *Developer Column* to view the xSOFTip documentation. You can switch back to this tutorial by clicking the *xTutorial* tab.

2. Drag the **Ethernet/TCP xSOFTip** into the *My System Configuration* window.
3. Click on the arrow on the left of the **Ethernet/TCP** module to view the configuration for the component.



My System Configuration

▼ My Application (Generic Application)

▼ Ethernet/TCP Module

Use a separate layer 2 MAC	No
Number of clients	1
Combine SMI ports on single port	No
Use PHY reset signal	Yes
✓ Configuration Status	Trusted

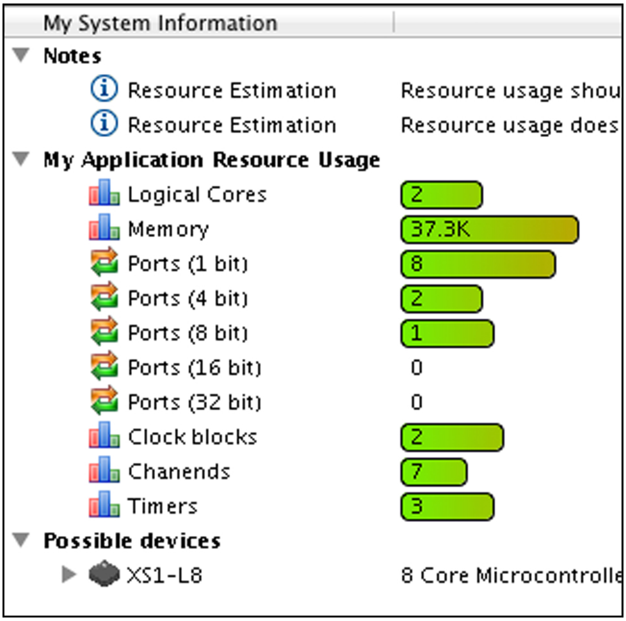
▶ Resource Usage

4. Select the **Ethernet/TCP Module** in the *My System Configuration* window. The *Main* tab of the *Developer Column* provides a description of each configuration option.

As all peripherals in XMOS are implemented using software, you have complete freedom to customise the interface to meet your exact requirements. In this example you will be using the default configuration, so no changes are needed.

3.2 Identifying required system resources

The *My System Information* window shows the resources used by your configuration. As you add components, the window is updated to show the resources used by all the selected xSOFTip components.



My System Information	
▼ **Notes**	
ⓘ Resource Estimation	Resource usage shou
ⓘ Resource Estimation	Resource usage does
▼ **My Application Resource Usage**	
Logical Cores	2
Memory	37.3K
Ports (1 bit)	8
Ports (4 bit)	2
Ports (8 bit)	1
Ports (16 bit)	0
Ports (32 bit)	0
Clock blocks	2
Chanends	7
Timers	3
▼ **Possible devices**	
▶ XS1-L8	8 Core Microcontrolle

Resources include:

- **Logical Cores**: 32bit microcontroller cores. XMOS multicore microcontrollers include 4, 6, 8, 10, 12, 16 and 32 core devices.
- **Ports**: I/O pins of XMOS multicore microcontrollers are connected to *ports*, which allow your software to send and receive data to the pins with extremely low latency. Ports are available in different widths: a 1-bit port is connected to 1 I/O pins, a 4-bit port is connected to 4 I/O pins.
- **Clock Blocks**: Clock blocks are used to precisely control timing of I/O pins.
- **Chanends**: Channel Ends are part of the xConnect system, allowing the cores to send messages to each other through low latency xConnect channels.
- **Timers**: Timers are used by the software to control the time at which things happen. They run at 100MHz, giving 10ns precision.

3.3 Add more interfaces to your system

All XMOS I/O pins are general purpose, so you can use them for any hardware interface you need.

Select the following xSOFTip components to provide the interfaces you need in our system. Drag them into the *My System Configuration* window:

1. Select the **I2S Master Audio Driver** in the *Audio/Interfaces* category (or use the Search box to find it).

   This will be used to send and receive audio data.

2. Select the **SDRAM Memory Controller** interface in the *Memory/Volatile* category.

   An external SDRAM will provide a large external memory that you can use to store audio data for the Reverb effect.

3. Select the **Long Reverb Function Library** in the *Audio/DSP* category.

   Use the Reverb library to implement reverb.

4. Select the **I2C Master (Single Bit Ports) Function Library** in the *Peripherals/Serial* category.

   The I2C interface will be used to control external devices.

3.4 Changing your configuration

Making changes to your XMOS design is very easy. Perhaps you now need a UART interface rather than I2C.

1. Right-click on the *I2C Master (Single Bit Ports) Function Library* and click **Delete**.
2. Select the **Generic UART Transmitter** (*Peripherals/Serial*) in the *xSOFTip Browser* and drag it into the *My System Configuration* window.

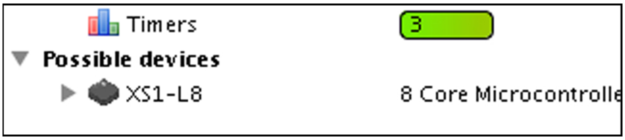
Take a look at the other xSOFTip components and add anything else you want in your system.

XMOS and our partners are working on new xSOFTip components all the time. Some components you see here are roadmap components which are in our development plan. If there is a component you require for your system, please let us know – we'd love to hear from you.

4 Identifying suitable xCORE devices

When you've configured your application you can use the *My System Information* window to identify the xCORE device that most closely match your system requirements.

A list of *Possible Devices* is displayed at the bottom of the *System Information* window.

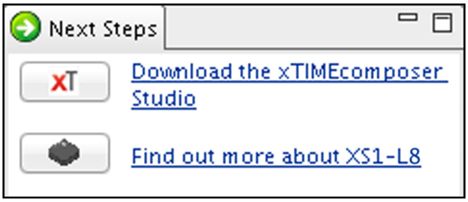


Timers 3

▼ Possible devices

▶ XS1-L8 8 Core Microcontroll

The *Next Steps* window contains links the XMOS website where you can find further information about the devices.



Remember that you will need some resources for your application code. You might want a single core and some memory for control code. The Reverb function library has already been included, but you might want a extra cores and additional memory for your own custom DSP processing.

5 Next Steps

Congratulations, you've now assembled your first xSOFTip components and are ready to start developing your xCORE embedded application.

xSOFTip is all delivered as C code, so you easily change it to meet your exact requirements. You can also take existing C functions and run them on an xCORE. For interfacing to I/O pins and for communicating between cores we've added some operations to C, called *XC*.

Now you have explored our xSOFTip, upgrade to our xTIMEcomposer Studio development suite – free of charge! The xTIMEcomposer Studio is our full development suite, delivering all the code for the xSOFTip components, together with compiler, debugger and everything else you need to build and run your application.

Download xTIMEcomposer Studio[1] now!

Follow the xTIMEcomposer Studio Tutorial - Simulator, which shows you how to create a PWM project from xSOFTip and simulate it in *xTIMEcomposer Studio*. See **Help > Tutorials > xTIMEcomposer Studio Tutorial** in xTIMEcomposer Studio.

XMOS®

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.

[1] http://www.xmos.com/support/downloads/xtimecomposer