Sample Rate Conversion Library

The XMOS Sample Rate Conversion (SRC) library provides both synchronous and asynchronous audio sample rate conversion functions for use on xCORE-200 multicore micro-controllers.

In systems where the rate change is exactly equal to the ratio of nominal rates, synchronous sample rate conversion (SSRC) provides efficient and high performance rate conversion. Where the input and output rates are not locked by a common clock or clocked by an exact rational frequency ratio, the Asynchronous Sample Rate Converter (ASRC) provides a way of streaming high quality audio between the two different clock domains, at the cost of higher processing resource usage. ASRC can ease interfacing in cases where the are multiple digital audio inputs or allow cost saving by removing the need for physical clock recovery using a PLL.

Features

- Conversion between 44.1, 48, 88.2, 96, 176.4 and 192KHz input and output sample rates
- 32 bit PCM input and output data in Q1.31 signed format
- Optional output dithering to 24 bit using Triangular Probability Density Function (TPDF)
- Optimized for xCORE-200 instruction set with dual-issue
- Block based processing - Minimum 4 samples input per call, must be power of 2
- Up to 10000 ppm sample rate ratio deviation from nominal rate (ASRC only)
- Very high quality - SNR greater than 135db (ASRC) or 140db (SSRC), with THD of less than 0.0001% (reference 1KHz)
- Configurable number of audio channels per SRC instance
- Reentrant library permitting multiple instances with differing configurations and channel count
- No external components (PLL or memory) required

Components

- Synchronous Sample Rate Converter function
- Asynchronous Sample Rate Converter function

Software version and dependencies

This document pertains to version 1.0.0 of this library. It is known to work on version 14.2.0 of the xTIMEcomposer tools suite, it may work on other versions.

This library depends on the following other libraries:

- lib_logging (>=2.0.0)
- lib_xassert (>=2.0.0)

Related application notes

The following application notes use this library:

- AN00230 - [Adding Synchronous Sample Rate Conversion to the USB Audio reference design]
- AN00231 - [SPDIF receive to I^2S output using Asynchronous Sample Rate Conversion]

Copyright 2016 XMOS Ltd.

Typical Resource Usage

This following table shows typical resource usage in some different configurations. Exact resource usage will depend on the particular use of the library by the application.

Configuration	Pins	Ports	Clocks	Ram	Logical cores
SSRC	0	0	0	~30.5K	1

The SSRC algorithm runs a series of cascaded FIR filters to perform the rate conversion. This includes interpolation, decimation and bandwidth limiting filters with a final polyphase FIR filter. The last stage supports the rational rate change of 147:160 or 160:147 allowing conversion between 44.1KHz family of sample rates to the 48KHz family of sample rates.

The below table shows the worst case MHz consumption at a given sample rate using the minimum block size of 4 input samples with dithering disabled. The MHz requirement can be reduced by around 8-12%, depending on sample rate, by increasing the input block size to 16. It is not usefully reduced by increasing block size beyond 16.

	Output sample rate					
Input sample rate	44.1KHz	48KHz	88.2KHz	96KHz	176.4KHz	192KHz
44.1KHz	1MHz	23MHz	16MHz	26MHz	26MHz	46MHz
48KHz	26MHz	1MHz	28MHz	17MHz	48MHz	29MHz
88.2KHz	18MHz	43MHz	1MHz	46MHz	32MHz	53MHz
96KHz	48MHz	20MHz	52MHz	2MHz	56MHz	35MHz
176.4KHz	33MHz	61MHz	37MHz	67MHz	3MHz	76MHz
192KHz	66MHz	36MHz	70MHz	40MHz	80MHz	4MHz

Table 1: SSRC Processor Usage per Channel (MHz)

This following table shows typical resource usage in some different configurations. Exact resource usage will depend on the particular use of the library by the application.

Configuration	Pins	Ports	Clocks	Ram	Logical cores
ASRC	0	0	0	~28.5K	1

The ASRC algorithm also runs a series of cascaded FIR filters to perform the rate conversion. The final filter is different because it uses adaptive coefficients to handle the varying rate change between the input and the output. The adaptive coefficients must be computed for each output sample period, but can be shared amongst all channels within the ASRC instance. Consequently, the MHz usage of the ASRC is expressed as two tables; the first table enumerates the MHz required for the first channel with adaptive coefficients calculation and the second table specifies the MHz required for filtering of each additional channel processed by the ASRC instance.

The below tables show the worst case MHz consumption per sample, using the minimum block size of 4 input samples. The MHz requirement can be reduced by around 8-12% by increasing the input block size to 16.

Typically you will need to allow for performance headroom for buffering (especially if the system is sample orientated rather than block orientated) and inter-task communication. Please refer to the application notes for practical examples of usage.

	Output sample rate					
Input sample rate	44.1KHz	48KHz	88.2KHz	96KHz	176.4KHz	192KHz
44.1KHz	29MHz	30MHz	40MHz	42MHz	62MHz	66MHz
48KHz	33MHz	32MHz	42MHz	43MHz	63MHz	66MHz
88.2KHz	47MHz	50MHz	58MHz	61MHz	80MHz	85MHz
96KHz	55MHz	51MHz	67MHz	64MHz	84MHz	87MHz
176.4KHz	60MHz	66MHz	76MHz	81MHz	105MHz	106MHz
192KHz	69MHz	66MHz	82MHz	82MHz	109MHz	115MHz

Table 2: ASRC Processor Usage (MHz) for the First Channel in the ASRC Instance

Configurations requiring more than 100MHz cannot currently be run in real time on a single core. The performance limit for a single core on a 500MHz xCORE-200 device is 100MHz (500/5). Further optimization of the library, including assembler optimization and pipelining of the adaptive filter generation and FIR filter stages, is feasible to achieve higher sample rate operation within the constraints of a 100MHz logical core.

	Output sample rate					
Input sample rate	44.1KHz	48KHz	88.2KHz	96KHz	176.4KHz	192KHz
44.1KHz	28MHz	28MHz	32MHz	30MHz	40MHz	40MHz
48KHz	39MHz	31MHz	33MHz	36MHz	40MHz	45MHz
88.2KHz	51MHz	49MHz	57MHz	55MHz	65MHz	60MHz
96KHz	51MHz	56MHz	57MHz	62MHz	66MHz	71MHz
176.4KHz	60MHz	66MHz	76MHz	79MHz	92MHz	91MHz
192KHz	69MHz	66MHz	76MHz	82MHz	90MHz	100MHz

Table 3: ASRC Processor Usage (MHz) for Subsequent Channels in the ASRC Instance

XMOS®

Copyright © 2016, 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.