

U-Series Support Library

This library provides support for accessing the available functionality of the XMOS U-Series devices.

Features

- ADC support.

Software version and dependencies

This document pertains to version 2.0.1 of this library. It is known to work on version 14.1.1 of the xTIMEcomposer tools suite, it may work on other versions.

This library depends on the following other libraries:

- lib_logging ($\geq 2.0.0$)
- lib_xassert ($\geq 2.0.0$)

1 A-Series ADC API

Type	<code>adc_bits_per_sample_t</code>
Description	Valid <code>bits_per_sample</code> values (8, 16 or 32).
Values	<p><code>ADC_8_BPS</code> Samples will be truncated to 8 bits.</p> <p><code>ADC_16_BPS</code> Samples will be placed in the MSB 12 bits of the half word.</p> <p><code>ADC_32_BPS</code> Samples will be placed in the MSB 12 bits of the word.</p>

Type	<code>adc_config_t</code>
Description	Configuration structure for ADCs:.
Fields	<p><code>char input_enable</code></p> <p><code>adc_bits_per_sample_t bits_per_sample</code></p> <p><code>unsigned int samples_per_packet</code></p> <p><code>int calibration_mode</code></p>
Parameters	<p><code>input_enable</code> An array ints to determine which inputs are active. Each non-zero input will be enabled.</p> <p><code>bits_per_sample</code> Select how many bits to sample per ADC.</p> <p><code>samples_per_packet</code> Number of samples per packet. Must be >0 and <=XS1_MAX_SAMPLES_PER_PACKET.</p> <p><code>calibration_mode</code> When set the ADCs will sample a 0.8V reference rather than the external voltage.</p>

Function	adc_enable
Description	Configure and enable the requested ADCs. Will also perform the calibration pulses so that the ADCs are ready to provide data. adc_enable() also checks that the configuration is valid and will raise a trap if attempting to incorrectly configure the ADCs.
Type	void adc_enable(tileref periph_tile, chanend adc_chan, out port trigger_port, const_adc_config_ref_t config)
Parameters	periph_tile The identifier of the tile containing the ADCs adc_chan The chanend to which all ADC samples will be sent. trigger_port The port connected to the ADC trigger pin. config The configuration to be used.
Returns	ADC_OK on success and one of the return codes in adc_return_t on an error.

Function	adc_disable_all
Description	Disable all of the ADCs.
Type	void adc_disable_all(tileref periph_tile)

Function	adc_trigger
Description	Causes the ADC to take one sample. This function is intended to be used with adc_read() . If used with adc_read_packet() then this function must be called enough times to ensure that an entire data packet will be available before the adc_read_packet() is called.
Type	void adc_trigger(out port trigger_port)
Parameters	trigger_port The port connected to the ADC trigger pin.

Function	adc_trigger_packet
Description	Trigger the ADC enough times to complete a packet.
Type	void adc_trigger_packet(out port trigger_port, const_adc_config_ref_t config)
Parameters	trigger_port The port connected to the ADC trigger pin. config The ADC configuration.

Function	adc_read
Description	A selectable function to read an ADC sample from the chanend. Any control tokens due to packetization will be discarded silently. Note that the adc_trigger function must have been called before this function will return any data. Note that the configuration must be the same as that used when enabling the ADCs.
Type	void adc_read(chanend adc_chan, const_adc_config_ref_t config, unsigned int &data)
Parameters	adc_chan The chanend to which all ADC samples will be sent. config The ADC configuration. data The word to place the data in.

Function	adc_read_packet
Description	A selectable function to read a packet of ADC samples from the chanend. Note that the adc_trigger_packet function must have been called before this function will return any data. Note that the configuration must be the same as that used when enabling the ADCs.
Type	void adc_read_packet(chanend adc_chan, const_adc_config_ref_t config, unsigned int data[])

Continued on next page

Parameters	adc_chan	The chanend to which all ADC samples will be sent.
	config	The ADC configuration.
	data	The buffer to place the returned data in. Each sample will be placed in a separate word. The buffer must be big enough to store all the data that will be read (samples_per_packet words).

APPENDIX A - Known Issues

There are no known issues with this library.

APPENDIX B - U-Series support library change log

B.1 2.0.1

- Update to source code license and copyright

B.2 2.0.0

- Initial release in library format
- Changes to dependencies:
 - lib_logging: Added dependency 2.0.0
 - lib_xassert: Added dependency 2.0.0