Zipper Deprecation Notice:

Beginning with OpenCPI Version 1.5, support for Lime Microsystems' Zipper card is now deprecated.

Summary - Lime ADC

Name	lime_adc
Worker Type	Device
Version	v1.5
Release Date	4/2019
Component Library	ocpi.assets.devices
Workers	lime_adc.hdl
Tested Platforms	• Epiq Solutions Matchstiq-Z1 • Digilent Zedboard/Zipper • x86/Xilinx ML605/Zipper (FMC-LPC/FMC-HPC) • x86/Altera ALST4/Zipper (HSMC A/B)

Functionality

The Lime ADC device worker converts the Lime LMS6002Dr2 Transceiver ADC interface into the OpenCPI WSI interface. The ADC data enters the worker in the sample clock domain (ADC_CLK) and is registered, de-interleaved, sign-extended, and converted to the control clock domain.

Worker Implementation Details

lime_adc.hdl

Figure 1 shows the Lime ADC signal timing interface in the sample clock domain. There are 14 input signals in the interface: ADC_CLK(1), RX_IQ_SEL(1), and RXD(12). The format of the data (RXD) is interleaved complex samples. One data sample (I and Q) is clocked in every two ADC_CLK cycles with RX_IQ_SEL serving as the qualifier for the I sample. The data width for the ADC is 12 bits and the data format is two's complement.

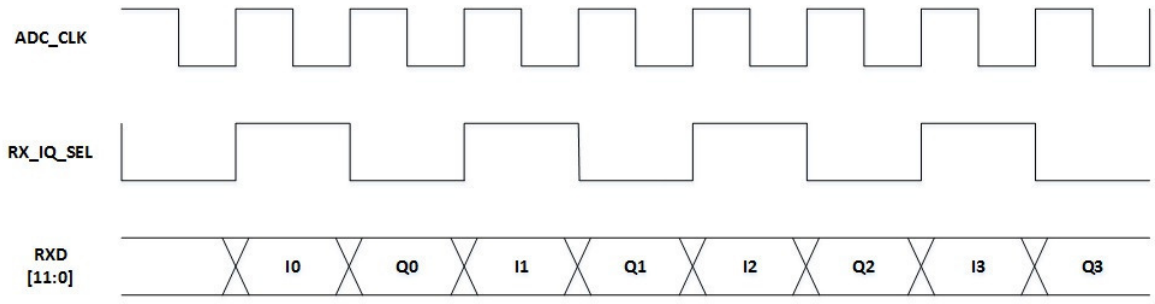


Figure 1: Lime ADC Interface: Sample Clock Domain

The clock domain crossing (CDC) from the sample clock to the OpenCPI control clock is performed using a two-clocked synchronizing FIFO with data width of 24 bits and depth of 4096. Data is loaded into the FIFO using RX_IQ_SEL and unloaded when the downstream worker is ready. In the event that a sample cannot be loaded into the FIFO, the `overrun` property is set and remains set until it is cleared. The FIFO output signals are then translated into the WSI interface, which can be seen in Figure 2. The number of 32 bit complex samples transferred

between start-of-message (SOM) and end-of-message (EOM) is set using the worker's `messageSize` property, where `messageSize` is in bytes and there are four bytes per complex output sample.

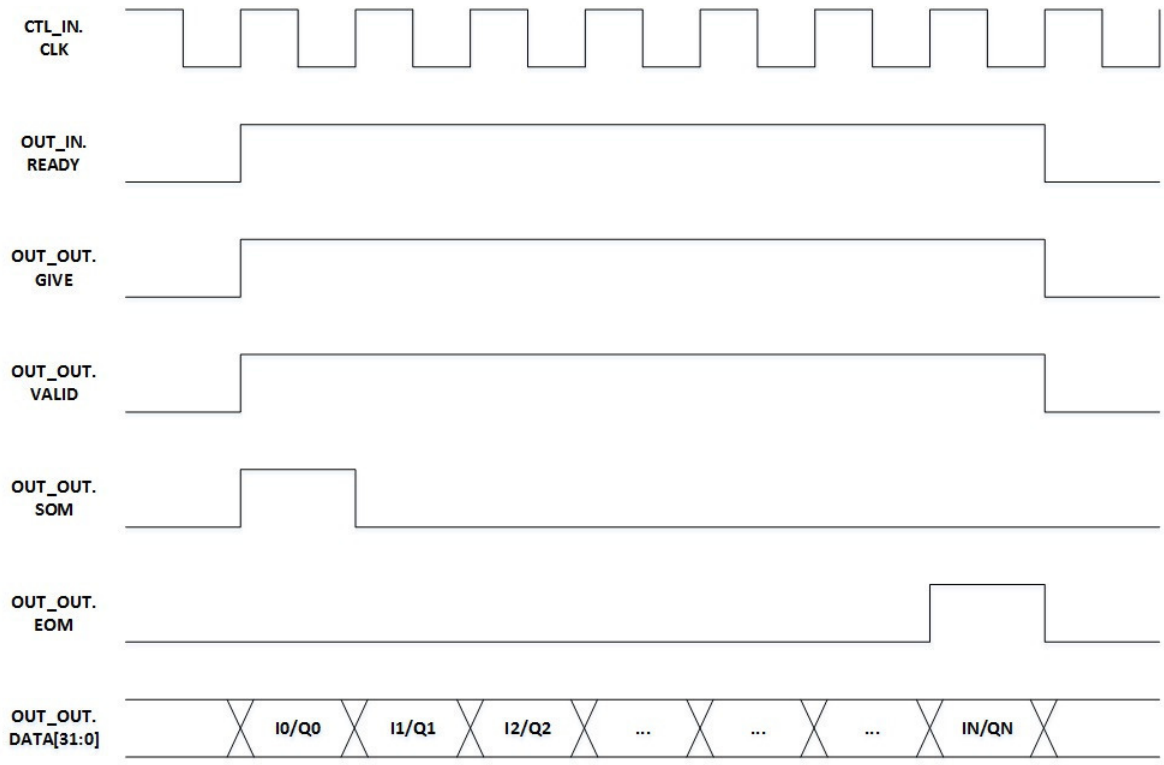


Figure 2: WSI Interface: Control Clock Domain

ADC_CLK can originate from one of three sources depending on the value of the parameters. The table below describes the valid settings.

USE_CLK_OUT_p	USE_CLK_IN_p	USE_CTL_CLK_p	ADC_CLK
True	X	X	RX_CLK_OUT
False	True	X	RX_CLK_IN
False	False	True	ctl_in.clk

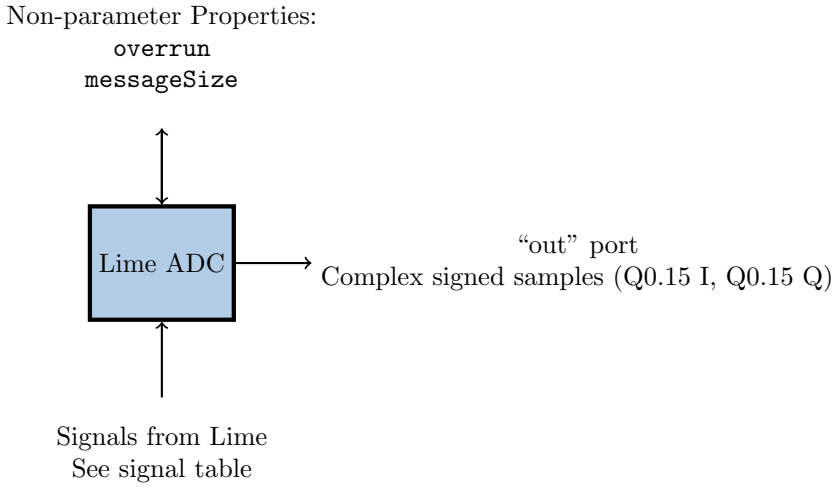
RX_CLK can be driven by this worker by setting the DRIVE_CLK_p parameter or it can be driven from another source external to the worker.

Theory

The main purpose of this worker is to perform a CDC for a data bus. The decision was made to implement the CDC using a two-clocked FIFO in an effort to target resources native the FPGA.

Block Diagrams

Top level



Source Dependencies

lime_adc.hdl

- assets/hdl/devices/lime_adc.hdl/lime_adc.vhd
- core/hdl/primitives/util/adc_fifo.vhd
  - Performs the clock domain crossing from the sample clock to the control clock domain
  - core/hdl/primitives/util/sync_status.vhd
    - Generates the overrun event when the ADC tries to load a sample and the ADC FIFO is full
  - core/hdl/primitives/bsv/imports/SyncFIFO.v
    - Two-clocked CDC FIFO

Component Spec Properties

Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
messageSize	Long	-	-	Initial, Readable	Standard	-	Number of bytes in output message
overrun	Bool	-	-	Writable, Volatile	Standard	-	Flag set when ADC tries to load a sample and the ADC FIFO is full. Once high, this flag is not cleared (i.e. set low) until the property is written to again (the flag clears regardless of write value, i.e. writing true or false both result in a value of false, also note that a property write happens on reset).

Worker Properties

lime_adc.hdl

Type	Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
Property	other_present	Bool	-	-	Readable	-	-	Not implemented. Flag to indicated presence of DAC worker
Property	DRIVE_CLK_p	Bool	-	-	Parameter	Standard	1	Drive the clock sent to Lime (RX_CLK). Some platforms do not connect RX_CLK to the FPGA, making this parameter false
Property	USE_CLK_IN_p	Bool	-	-	Parameter	Standard	0	Use copy of clock sent to Lime (RX_CLK) as ADC_CLK. Not guaranteed to be aligned with RX data
Property	USE_CTL_CLK_p	Bool	-	-	Parameter	Standard	1	Use control clock as ADC_CLK. This is primarily for testing the component.
Property	divisor	-	-	-	Writable	-	-	Not implemented. Divider for ADC clock. This is primarily for testing the component.
Property	USE_CLK_OUT_p	Bool	-	-	Parameter	Standard	1	Use clock output from Lime (RX_CLK_OUT) as ADC_CLK. RX_CLK_OUT is aligned with RX data
Property	source	Enum	-	-	Initial	adc,count,loopback	adc	Not implemented. Runtime property to indicate worker parameter configuration
SpecProperty	messageSize	-	-	-	-	-	8192	Number of bytes in output message
SpecProperty	overrun	-	-	-	-	-	0	Flag set when ADC tries to load a sample and the ADC FIFO is full. Once high, this flag is not cleared (i.e. set low) until the property is written to again (the flag clears regardless of write value, i.e. writing true or false both result in a value of false, also note that a property write happens on reset).

Component Ports

Name	Producer	Protocol	Optional	Advanced	Usage
out	true	iqstream_protocol	true	-	Complex signed samples (Q0.15 I, Q0.15 Q).

Worker Interfaces

lime_adc.hdl

Type	Name	DataWidth	Advanced	Usage
StreamInterface	out	32	-	Complex signed samples (Q0.15 I, Q0.15 Q). This port generates data and obeys backpressure. Because both backpressure from the `out` port and forward pressure from the `dev_adc` data bus exists, it is possible for samples to be dropped in the clock domain-crossing FIFO, i.e. seen on the `dev_adc` data bus but never make it to the output port. This event is monitored via the `overrun` property.

Signals

Name	Type	Width	Description
RX_CLK	Output	1	Clock input to Lime
RX_CLK_OUT	Input	1	Clock output from Lime
RX_IQ_SEL	Input	1	IQ Select from Lime
RXD	Input	12	Lime ADC data bus. IQ interleaved
RX_CLK_IN	Input	1	Copy of RX_CLK sent to FPGA

Control Timing and Signals

The Lime ADC device worker uses the clock from the Control Plane and Control Plane signals.

The latency through the worker from the ADC pins to the output port is 2 sample clock cycles and 1 control clock cycle. The data is registered twice in the sample clock domain (once to capture the ADC pins, and once to capture I and Q) before it is loaded into the CDC FIFO.

Worker Configuration Parameters

lime_adc.hdl

Table 1: Table of Worker Configurations for worker: lime_adc

Configuration	ocpi_debug	DRIVE_CLK_p	USE_CTL_CLK_p	USE_CLK_OUT_p	ocpi_endian	USE_CLK_IN_p
0	false	0	0	1	little	0
1	false	0	0	0	little	1

Performance and Resource Utilization

lime_adc.hdl

The table entries for configuration "0" are a result of building the worker for the following (Matchstiq-Z1-related) parameter set. Note that that are two global clocks (GCLKs), one for the control plane clock, and one for the Lime ADC-sourced clock (Fmax is assumed to be worst case for all available clocks).

The Virtex-6 implementation is specific to the Zipper card's pin locations, which exhibit suboptimal timing due to the Lime ADC's clock pin location on the FPGA. Table entries for configuration1 "1" are a result of building the worker for the (Zipper-related) parameter set.

Table 2: Resource Utilization Table for worker "lime_adc"

Configuration	OCPI Target	Tool	Version	Device	Registers (Typ)	LUTs (Typ)	Fmax (MHz) (Typ)	Memory/Special Functions
0	stratix4	Quartus	17.1.0	N/A	273	201	N/A	Block Memory Bits: 98304
0	zynq	Vivado	2017.1	xc7z020clg400-3	303	217	N/A	BUFGCTRL: 1 BUFG: 1 RAMB36E1: 3
0	zynq_ise	ISE	14.7	7z010clg400-3	298	283	348.335	BUFGCTRL: 1 BUFG: 1 RAMB36E1: 3
0	virtex6	ISE	14.7	6vcx75tff484-2	298	299	311.439	BUFGCTRL: 1 BUFG: 1 RAMB36E1: 3
1	stratix4	Quartus	17.1.0	N/A	273	201	N/A	Block Memory Bits: 98304
1	zynq	Vivado	2017.1	xc7z020clg400-3	303	217	N/A	BUFGCTRL: 1 BUFG: 1 RAMB36E1: 3
1	zynq_ise	ISE	14.7	7z010clg400-3	298	283	348.335	BUFGCTRL: 1 BUFG: 1 RAMB36E1: 3
1	virtex6	ISE	14.7	6vcx75tff484-2	298	299	311.439	BUFGCTRL: 1 BUFG: 1 RAMB36E1: 3

Test and Verification

To be detailed in a future release.

References

1) LMS6002D Datasheet, www.limemicro.com