OpenCPI
TimeIQ to IQstream Component Data Sheet

OpenCPI Release: v2.3.4

Summary - TimeIQ to IQstream

Name	timeiq_to_iqstream
Worker Type	Application
OpenCPI Release	v2.3.4
Last Update	4/2019
Component Library	ocpi.assets.misc_comps
Workers	timeiq_to_iqstream.hdl
Tested Platforms	alst4, e3xx, isim, matchstiq_z1, ml605, modelsim, xsim, zed

Functionality

The TimeIQ to IQstream component adapts the TimeStamped_IQ protocol to the iqstream protocol.

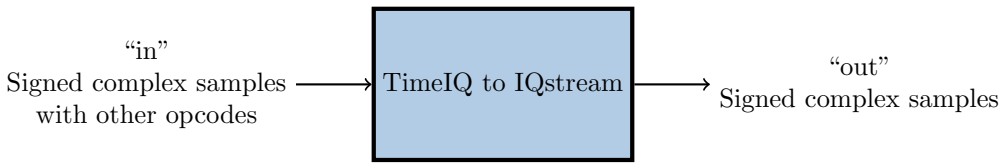
Worker Implementation Details

timeiq_to_iqstream.hdl

The TimeStamped_IQ protocol consists of multiple opcodes which include complex IQ samples, and the iqstream protocol consists only of complex IQ samples. The timeiq_to_iqstream worker inspects the opcode of the input port and only forwards data with the `samples` opcode.

Block Diagrams

Top level



Source Dependencies

timeiq_to_iqstream.hdl

- bsp_picoflexor/components/timeiq_to_iqstream.hdl/timeiq_to_iqstream.vhd

Component Spec Properties

There are no component spec properties for this component

Worker Properties

There are no worker implementation-specific properties for this component

Component Ports

Name	Producer	Protocol	Optional	Advanced	Usage
in	false	TimeStamped_IQ-prot	false	-	Signed complex samples plus other operations
out	true	iqstream_protocol	false	-	Signed complex samples

Worker Interfaces

timeiq_to_iqstream.hdl

Type	Name	DataWidth	Advanced	Usage
StreamInterface	in	32	-	Signed complex samples plus other operations
StreamInterface	out	32	-	Signed complex samples

Control Timing and Signals

The timeiq_to_iqstreamworker uses the clock from the Control Plane and standard Control Plane signals.

Worker Configuration Parameters

timeiq_to_iqstream.hdl

Performance and Resource Utilization

Test and Verification

The input file contains message metadata and uses three different opcodes from the TimeStamped_IQ protocol, and is formatted in the following manner:

1. 8 byte `time` operation
2. 8 byte `interval` operation
3. 2880 byte `samples` operation
4. 8 byte `time` operation
5. 2880 byte `samples` operation

The data in the samples operation is a ramp from 0 to 2087.

The expected output waveform is the identical ramp with all other operations removed. For verification, the output file is compared to a golden input file.