

## Summary - Timestamper

Name	timestamper
Worker Type	Application
Version	v1.5
Release Date	4/2019
Component Library	ocpi.assets.util_comps
Workers	timestamper.hdl
Tested Platforms	xsim, isim, modelsim, zed, e3xx, Matchstiq-Z1(PL), alst4, ml605

## Functionality

The Timestamper component inputs complex IQ data and outputs complex IQ data prepended with a timestamp. One timestamp is sent for each data message produced on the output.

## Worker Implementation Details

### timestamper.hdl

A timing diagram of the output interface for this component can be seen in Figure 1.

Timestamps are provided as an input to the component on the time interface. The timestamp is a 64 bit number with the first 32 bits corresponding to seconds and the last 32 bits corresponding to fractional seconds. When a valid message is detected on the input, the timestamp is registered by the component and given on the output interface. Timestamps and data are given on the output interface using different opcodes.

The time interface from which the timestamps are generated originates from the OpenCPI time server, which is instanced as part of the platform worker. Furthermore, an additional component (time client) is dynamically instanced by the framework for all components which declare time interfaces. The time client communicates with the time server and produces the time interface seen by the component.

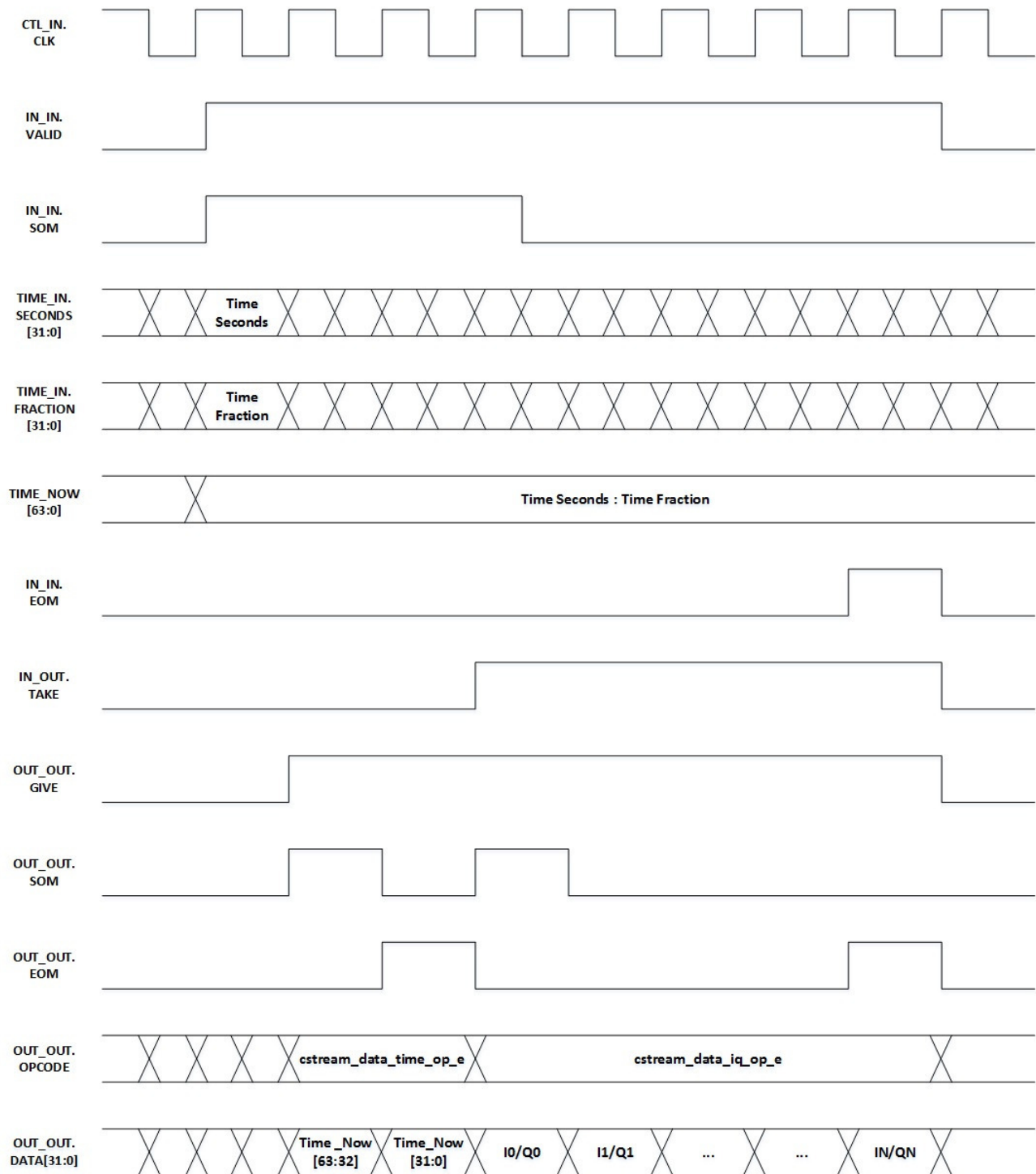
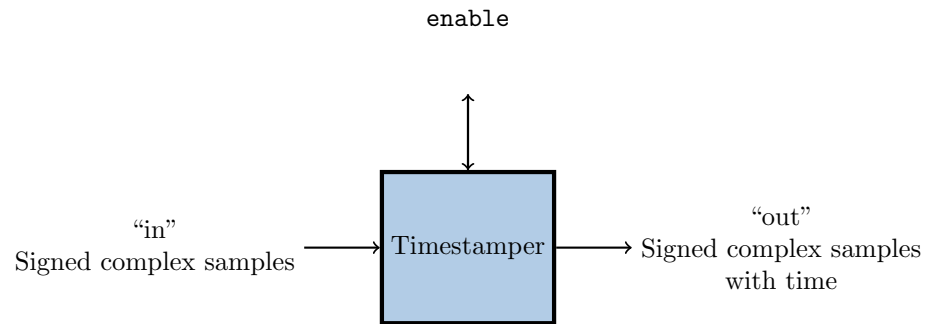


Figure 1: Timestamper Output Timing Diagram

## Block Diagrams

### Top level



## Source Dependencies

### timestamper.hdl

- projects/assets/components/util.comps/timestampers.hdl/timestampers.vhd

## Component Spec Properties

Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
enable	Bool	-	-	Initial	Standard	true	Enable or bypass timestamp

## Worker Properties

timestamp.hdl

Type	Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
SpecProperty	enable	-	-	-	Readable	-	-	Enable or bypass timestamp

## Component Ports

Name	Producer	Protocol	Optional	Advanced	Usage
in	false	iqstream_protocol	false	-	Signed complex samples
out	true	iqstream_protocol_with_sync	false	-	Signed complex samples with timestamps

## Worker Interfaces

timestamp.hdl

Type	Name	DataWidth	Advanced	Usage
StreamInterface	in	32	numberOfOpcodes=256	Signed complex samples
StreamInterface	out	32	numberOfOpcodes=256	Signed complex samples with timestamps
TimeInterface	time	-	SecondsWidth=32 FractionWidth=32	Time interface provided from Time Server

## Control Timing and Signals

The Timestamper HDL worker uses the clock from the Control Plane and standard Control Plane signals.

Latency
3

Data presented on the input appears on the output 3 clock cycles later. 2 of the 3 clock cycles consist of a time message

# Worker Configuration Parameters

timestampper.hdl

Table 1: Table of Worker Configurations for worker: timestampper

Configuration
0

# Performance and Resource Utilization

timestampper.hdl

Table 2: Resource Utilization Table for worker "timestampper"

Configuration	OCPI Target	Tool	Version	Device	Registers (Typ)	LUTs (Typ)	Fmax (MHz) (Typ)	Memory/Special Functions
0	stratix4	Quartus	17.1.0	EP4SGX230KF40C2	316	215	N/A	N/A
0	zynq	Vivado	2017.1	xc7z020clg484-1	320	317	N/A	N/A
0	zynq_ise	ISE	14.7	7z020clg484-1	318	418	246.607	N/A
0	virtex6	ISE	14.7	6vlx240tff1156-1	318	426	236.9	N/A

## Test and Verification

Two test cases are implemented to validate the Timestampper component:

- 1) Bypass mode
- 2) Normal mode

For both cases, the input file is a series of 8 ramps with 32-bit values ranging from 0 to 512.

For case 1, the input data is forwarded to the output port. For verification, the output file is byte-wise compared to the input file.

For case 2, the expected output waveform is the identical ramp with timestamps inserted before each data message. For verification, the timestamps are extracted and checked for incrementing values