Summary - Fifo

Name	fifo
Latest Version	v1.5 (release date 4/2019)
Worker Type	Application
Component Library	ocpi.assets.util_comps
Workers	fifo.hdl
Tested Platforms	isim

Functionality

The FIFO component passes complex signed samples (Q0.15 I, Q0.15 Q) from the input port through a First-In-First-Out (FIFO) buffer and onward to the output port. The depth, in number of complex samples, of the FIFO buffer is parameterized. This component includes a property-driven oneshot mode which, when enabled, allows the first FIFO depth number of samples to be sent to the output port and then discontinues data flow to the output port. After data flow is discontinued, the input port still ingests available samples, effectively operating as a data sink. This worker can also be parameterized to send a Zero-Length Message (ZLM) once data flow is discontinued.

Worker Implementation Details

fifo.hdl

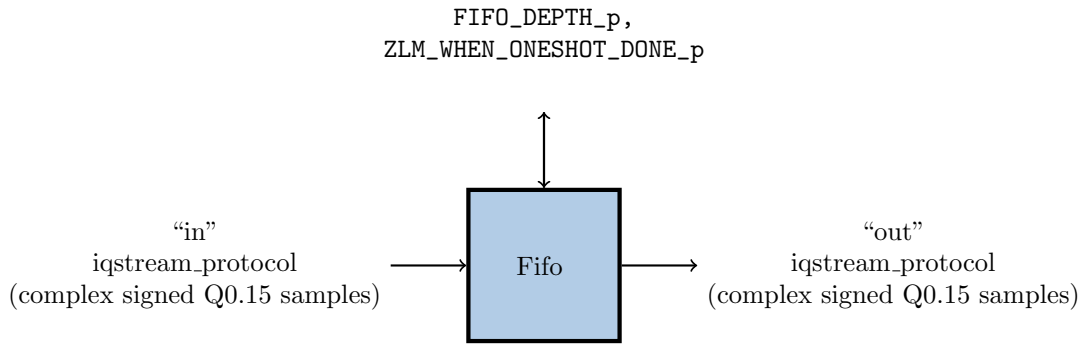
In keeping with good data flow control practices, backpressure is transferred, when necessary, from the output port to the input port. Backpressure is never transferred when in oneshot mode and after the data flow is discontinued. Backpressure from the output port and forwardpressure from the input port are both alleviated by the FIFO buffer, with the degree of alleviation being directly proportional to the parameterized depth of the FIFO buffer (`FIFO_DEPTH_p`).

The input port's SOM, EOM, byte_enable, and valid indicators are passed through the FIFO to the output port when data flow is allowed. Consequently, ZLMs will be passed through this worker. If operating in oneshot mode and data flow has been discontinued, the EOM will be set (i.e. logic value of 1 applied) on the same clock pulse as the last output sample.

The `ZLM_WHEN_ONESHOT_DONE_p` parameter property, when having a value of true, forces the worker to send a single ZLM when in oneshot mode and data flow has been discontinued (i.e. when oneshot is 'done'). This is useful for allowing applications which use this worker to terminate once data flow is discontinued.

Block Diagrams

Top level



Source Dependencies

fifo.hdl

- assets/components/util_comps/fifo.hdl/fifo.vhd
- core/hdl/primitives/bsv/bsv_pkg.vhd
- core/hdl/primitives/bsv/imports/SizedFIFO.v

Component Properties

Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Description
FIFO_DEPTH_p	ulong	-	-		Standard	1024	Maximum number of complex samples which the FIFO can hold at any given time.
ZLM_WHEN_ONESHOT_DONE_p	bool	-	-		Standard	false	When true, worker will generate Zero-Length-Message after oneshot was enabled and completed.
oneshot	bool	-	-	Readable, Initial	Standard	false	If false, the FIFO operates normally, i.e. it allows data to flow from adc_in to adc_passthrough_out and transfers backpressure accordingly. If true, the functionality is as follows. The FIFO receives data from adc_in and holds it until the FIFO is full. One the FIFO is full, data is pulled from the FIFO and sent out the adc_passthrough_out port, all the while data is pulled from adc_in but unused, effectively making the adc_in port a data sink. Note that a control plane reset will always reset the FIFO and start this operation over.

Component Ports

Name	Producer	Protocol	Optional
in	false	iqstream_protocol	False
out	true	iqstream_protocol	False

Worker Interfaces

fifo.hdl

Type	Name	DataWidth
StreamInterface	in	32
StreamInterface	out	32

Control Timing and Signals

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

Worker Configuration Parameters

fifo.hdl

Table 5: Table of Worker Configurations for worker: fifo

Configuration	FIFO_DEPTH_p	ocpi_debug	ocpi_endian	ZLM_WHEN_ONESHOT_DONE_p
0	8192	false	little	false
1	8192	false	little	true

Performance and Resource Utilization

fifo.hdl

Table 6: Resource Utilization Table for worker "fifo"

Configuration	OCPI Target	Tool	Version	Device	Registers (Typ)	LUTs (Typ)	Fmax (MHz) (Typ)	Memory/Special Functions
0	stratix4	Quartus	17.1.0	N/A	436	385	N/A	Block Memory Bits: 286685
0	zynq	Vivado	2017.1	xc7z020clg400-3	550	8142	N/A	N/A
0	zynq_ise	ISE	14.7	7z010clg400-3	328	8215	307.39	RAM64M: 1536
0	virtex6	ISE	14.7	6vcx75tff484-2	329	8221	254.738	RAM64M: 1536
1	stratix4	Quartus	17.1.0	N/A	436	473	N/A	Block Memory Bits: 286685
1	zynq	Vivado	2017.1	xc7z020clg400-3	550	8152	N/A	N/A
1	zynq_ise	ISE	14.7	7z010clg400-3	328	8218	307.39	RAM64M: 1536
1	virtex6	ISE	14.7	6vcx75tff484-2	328	8257	253.639	RAM64M: 1536

Test and Verification

For verification, multiple test files are generated of varying lengths. Each test file is passed into the input port, and the output of the output port is saved to a file. The output file is compared against the input file to make sure they have the same binary contents and length. For the tests that use oneshot mode, the output file is only compared to the first min(input file size,8192) samples, with 8192 hardcoded to correspond to `FIFO_DEPTH_p`.