Summary - AD9361 SPI

Package Prefix	ocpi.assets.devices
Name	ad9361_spi
OpenCPI Release	v1.5 (released 4/2019)
Workers	ad9361_spi.hdl
Tested Platforms	Zedboard (Vivado), ML605 (FMC LPC slot)

Functionality

The AD9361 SPI subdevice worker implements a SPI state machine for communication with the AD9361 IC[1].

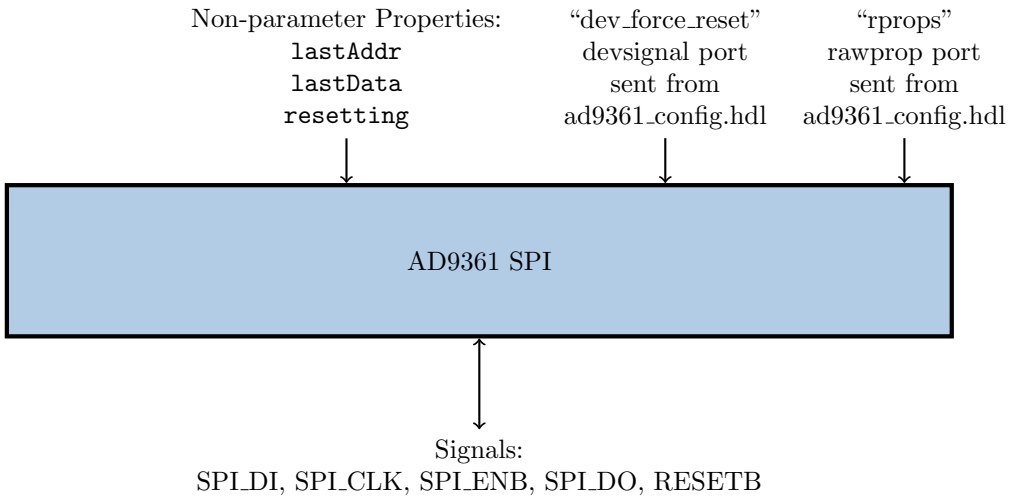
Worker Implementation Details

ad9361_spi.hdl

The ad9361_spi.hdl subdevice worker is intend for use in platforms/cards where a SPI bus exists which addresses only the AD9361. SPI read/writes are actuated by the `rprops` rawprop port. A devsignal is also sent which can force the AD9361 RESETB pin, which is active-low, to logic 0.

Block Diagrams

Top level



Source Dependencies

ad9361_spi.hdl

- assets/hdl/devices/ad9361_spi.hdl/ad9361_spi.vhd
- assets/hdl/devices/ad9361_spi.hdl/signals.vhd
- core/hdl/primitives/util/spi.vhd
- core/hdl/primitives/util/util_pkg.vhd

Component Spec Properties

Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
CP_CLK_FREQ_HZ_p	Float	-	-	Parameter	Standard	100e6	Value will determine assumed frequency of the Control Plane (CP) clock. This value is used to calculate the dividor for the SPI clock
SPI_CLK_FREQ_HZ_p	Bool	-	-	Parameter	Standard	6.25e6	-
lastAddr	UShort	-	-	Volatile	Standard	-	-
lastAddr	UChar	-	-	Volatile	Standard	-	-
lastAddr	Bool	-	-	Volatile	Standard	-	-

Worker Properties

ad9361_spi.hdl

Scope	Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
-	-	-	-	-	-	-	-	-

Component Ports

Name	Producer	Protocol	Optional	Advanced	Usage
-	-	-	-	-	-

Worker Interfaces

ad9361_spi.hdl

Type	Name	Optional	Usage
RawProp	rprops	True	-

Type	Name	Count	Optional	Master	Signal	Direction	Width	Description
DevSignal	dev_force_spi_reset	1	False	False	force_reset	Output	1	Used to force AD9361 RESETB pin, which is active-low, to logic 0.

Control Timing and Signals

The AD9361 SPI.hdl device worker operates entirely in the control plane clock domain. All SPI data and SPI clock signals are generated in the control plane clock domain. Note that SPI clock can only be a divided version of the control plane clock.

Worker Configuration Parameters

ad9361_spi.hdl

Table 2: Table of Worker Configurations for worker: ad9361_spi

Configuration	ocpi_debug	CP_CLK_FREQ_HZ_p	ocpi_endian	SPI_CLK_FREQ_HZ_p
0	false	100000000.0	little	6250000.0
1	false	125000000.0	little	6250000.0

Performance and Resource Utilization

ad9361_spi.hdl

Fmax refers to the maximum allowable clock rate for any registered signal paths within a given clock domain for an FPGA design. Fmax in the table below is specific only to this worker and represents the maximum possible Fmax for any OpenCPI bitstream built with this worker included. Note that the Fmax value for a given clock domain for the final bitstream is often worse than the Fmax specific to this worker, even if this worker is the only one included in the bitstream.

Table 3: Resource Utilization Table for worker: ad9361_spi

Configuration	OCPI Target	Tool	Version	Device	Registers (Typ)	LUTs (Typ)	Fmax (MHz) (Typ)	Memory/Special Functions
0	zynq	Vivado	2017.1	xc7z020clg484-1	67	94	235 [1]	N/A
0	stratix4	Quartus	17.1.0	EP4SGX230KF40C2	70	139	N/A	N/A
0	virtex6	ISE	14.7	6vlx240tff1156-1	65	130	437.445	N/A
1	zynq	Vivado	2017.1	xc7z020clg484-1	68	101	315 [1]	N/A
1	stratix4	Quartus	17.1.0	EP4SGX230KF40C2	71	141	N/A	N/A
1	virtex6	ISE	14.7	6vlx240tff1156-1	65	134	412.712	N/A

Test and Verification

The test outlined in [3] includes validation of a subset of this worker's functionality.

References

[1] AD9361 Datasheet and Product Info
`https://www.analog.com/en/products/ad9361.html`

[2] AD9361 Reference Manual UG-570
AD9361_Reference_Manual_UG-570.pdf

[3] AD9361 DAC Component Data Sheet
`https://opencpi.github.io/assets/AD9361_DAC.pdf`

[1] These measurements were the result of a Vivado timing analysis which was different from the Vivado analysis performed by default for OpenCPI worker builds. For more info see Appendix 1

1 Appendix - Vivado Timing Analysis

The Vivado timing report that OpenCPI runs for device workers may erroneously report a max delay for a clocking path which should have been ignored. Custom Vivado tcl commands had to be run for this device worker to extract pertinent information from Vivado timing analysis. After building the worker, the following commands were run from the base project directory (after the Vivado settings64.sh was sourced):

```
cd hdl/devices/
vivado -mode tcl
```

Then the following commands were run inside the Vivado tcl terminal for the parameter property set:

- `CP_CLK_FREQ_HZ_p`=100e6
- `SPI_CLK_FREQ_HZp`=6.25e6

```
open_project ad9361_spi.hdl/target-zynq/ad9361_spi_rv.xpr
synth_design -part xc7z020clg484-1 -top ad9361_spi_rv -mode out_of_context
create_clock -name clk1 -period 0.001 [get_nets {ctl_in[Clk]}]
report_timing -delay_type min_max -sort_by slack -input_pins -group clk1
```

The Fmax for the control plane clock for this worker is computed as the maximum magnitude slack with a control plane clock of 1 ps plus 2 times the assumed 1 ps control plane clock period (4.244 ns + 0.002 ns = 4.244 ns, 1/4.244 ns = 235.52 MHz). Then the following commands were run inside the Vivado tcl terminal for the parameter property set:

- `CP_CLK_FREQ_HZ_p`=125e6
- `SPI_CLK_FREQ_HZp`=6.25e6

```
open_project ad9361_spi.hdl/target-1-zynq/ad9361_spi_rv.xpr
synth_design -part xc7z020clg484-1 -top ad9361_spi_rv_c1 -mode out_of_context
create_clock -name clk1 -period 0.001 [get_nets {ctl_in[Clk]}]
report_timing -delay_type min_max -sort_by slack -input_pins -group clk1
```

The Fmax for the control plane clock for this worker is computed as the maximum magnitude slack with a control plane clock of 1 ps plus 2 times the assumed 1 ps control plane clock period (3.169 ns + 0.002 ns = 3.171 ns, 1/3.171 ns = 315.36 MHz).