

## Summary - Phase to Amplitude CORDIC

Name	phase_to_amp_cordic
Worker Type	Application
Version	v1.3
Release Date	February 2018
Component Library	ocpi.assets.dsp_comps
Workers	phase_to_amp_cordic.hdl
Tested Platforms	xsim, isim, modelsim, alst4, ml605, ZedBoard(PL), Matchstiq-Z1(PL)

## Functionality

This worker implements a phase to amplitude conversion (PAC). The real 16 bit signed input data is phase accumulated, then fed into a polar-to-rectangular CORDIC. The output of the CORDIC produces a complex waveform. Figure 1 diagrams the Phase to Amplitude CORDIC.

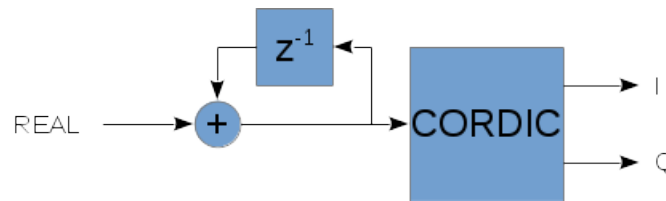


Figure 1: Phase to Amplitude CORDIC Functional Diagram

## Worker Implementation Details

### phase\_to\_amp\_cordic.hdl

The phase to amplitude converter calculates the amplitude for the current phase angle. This operation is basically the same as calculating the sine or cosine function of its argument. Two methods are typically used for implementing in hardware, the Coordinate Rotation Digital Computer (CORDIC) algorithm and the ROM lookup table.

This worker implements the CORDIC algorithm. The frequency of its complex output is determined by the following equation:

$$output\_freq = \frac{phs\_accum}{2^{DATA\_WIDTH}} \quad (1)$$

Where **phs\_accum** is the output of the accumulator. **DATA\_WIDTH** is the input/output data width of the CORDIC which has a range of 8 to 16. The input clock frequency is the sample rate of the samples. The amplitude of the complex wave is runtime configurable via the **magnitude** property. An **enable** input is available to either enable (true) or bypass (false) the circuit. In bypass mode, pipe-lining registers are not used, and the real input data is available on the lower 16 bits of the complex output.

Build time parameters control the width of the input, output and the number of stages of the CORDIC primitive module. The I/O data widths of the worker itself are set within the OCS and adjusted in the OWD.

## Block Diagrams

### Top level

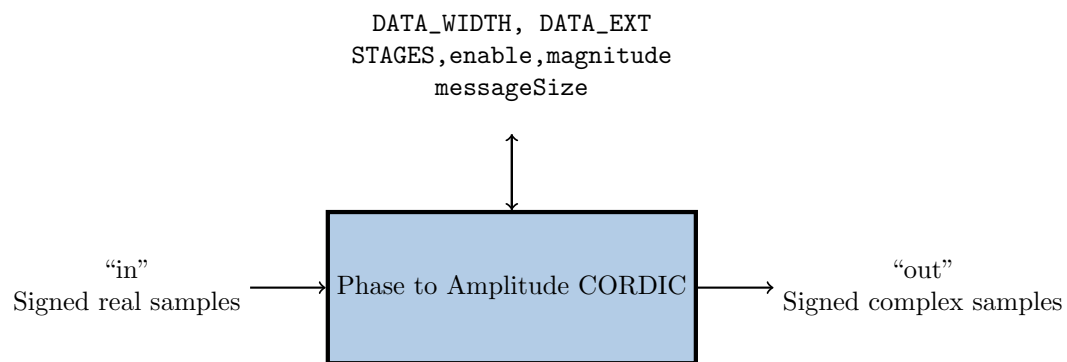


Figure 2: Top Level Block Diagram

## State Machine

Only one finite-state machine (FSM) is implemented by this worker. The FSM supports Zero-Length Messages.

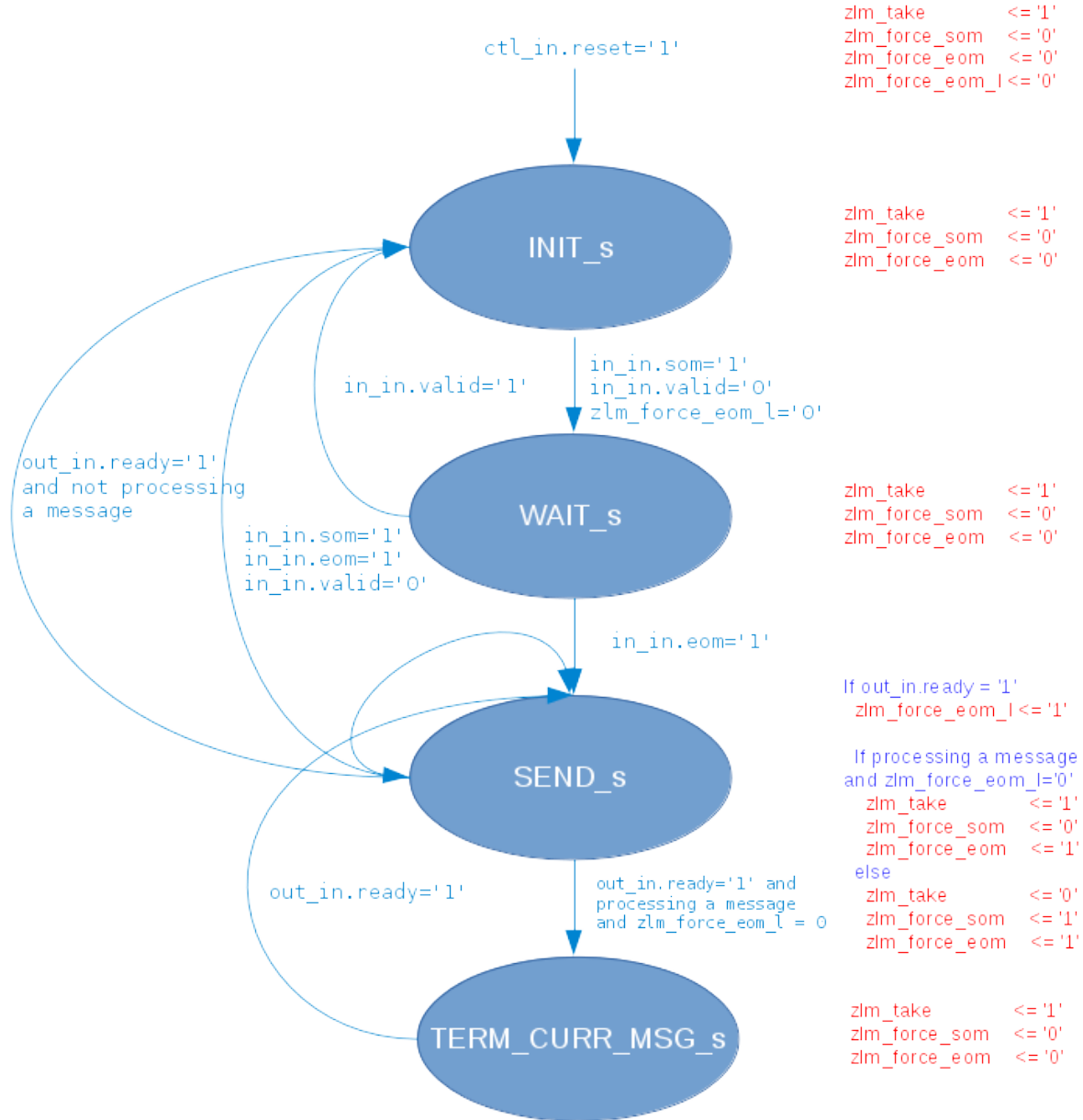


Figure 3: Zero-Length Message FSM

## Source Dependencies

### `phase_to_amp_cordic.hdl`

- `ocpiassets/components/dsp_comps/phase_to_amp_cordic.hdl/phase_to_amp_cordic.vhd`
- `ocpiassets/hdl/primitives/dsp_prims/dsp_prims_pkg.vhd`
  - `ocpiassets/hdl/primitives/dsp_prims/cordic/src/cordic_pr.vhd`
  - `ocpiassets/hdl/primitives/dsp_prims/cordic/src/cordic.vhd`
  - `ocpiassets/hdl/primitives/dsp_prims/cordic/src/cordic_stage.vhd`
- `ocpiassets/hdl/primitives/misc_prims/misc_prims_pkg.vhd`
  - `ocpiassets/hdl/primitives/misc_prims/round_conv/src/round_conv.vhd`

## Component Spec Properties

Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
DATA_WIDTH	UChar	-	-	Readable	-	-	Input (real) and Output (I/Q) data width
DATA_EXT	UChar	-	-	Readable	-	-	CORDIC requirement: # of extension bits
STAGES	UChar	-	-	Readable	-	-	Number of CORDIC stages implemented
messageSize	UShort	-	-	Readable, Writable	-	8192	Number of bytes in output message
enable	Bool	-	-	Readable, Writable	-	True	Enable(true) or bypass(false)
magnitude	UShort	-	-	Readable, Writable	-	16384	Magnitude of output  * $+2^{(DATA\_WIDTH)-1}$ to $-2^{(DATA\_WIDTH)}$

## Worker Properties

### phase\_to\_amp\_cordic.hdl

Type	Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
SpecProperty	DATA_WIDTH	-	-	-	Parameter	8-16	16	Input (real) and Output (I/Q) data width
SpecProperty	DATA_EXT	-	-	-	Parameter	6	6	CORDIC requirement: # of extension bits
SpecProperty	STAGES	-	-	-	Parameter	8-16	12	Number of CORDIC stages implemented

## Component Ports

Name	Producer	Protocol	Optional	Advanced	Usage
in	false	rstream_protocol	false	-	Signed real samples
out	true	iqstream_protocol	false	-	Signed complex samples

## Worker Interfaces

### phase\_to\_amp\_cordic.hdl

Type	Name	DataWidth	Advanced	Usage
StreamInterface	in	16	ZeroLengthMessages=true	Signed real samples
StreamInterface	out	32	ZeroLengthMessages=true	Signed complex samples

## Control Timing and Signals

The Phase to Amplitude CORDIC worker uses the clock from the Control Plane and standard Control Plane signals.

There is a startup delay for this worker. Once the input is ready and valid and the output is ready, there is a delay of **STAGES+1** before the first sample is taken. After this initial delay, valid output data is given **STAGES+1** clock cycles after input data is taken.

Latency
<b>STAGES+1</b> clock cycles

## Performance and Resource Utilization

### phase\_to\_amp\_cordic.hdl

#### Worker Build Configuration “0”:

Table entries are a result of building the worker with the following parameter sets:

- `ocpi_endian=little`
- `ocpi_debug=false`
- `STAGES=12`
- `DATA_EXT=6`
- `DATA_WIDTH=16`

Table 1: Worker Build Configuration “0”

OpenCPI Target	Tool	Version	Device	Registers	LUTs	Fmax (MHz)	Memory/Special Function
stratix4	Quartus	15.1.0	EP4SGX230KF40C2	1035	1,350	N/A	DSP18x18s=8
virtex6	ISE	14.7	6vlx240tff1156-1	978	2310	164.880	DSP48E1s=4
zynq	Vivado	2017.1	xc7z020clg484-1	1021	2283	120.322	DSP48E1s=2
zynq_ise	ISE	14.7	7z020clg484-1	978	2310	162.602	DSP48E1s=4

## Test and Verification

This component is tested via the unit test automation feature of the framework. The component's .test/ contains XML files that describe the combinations of tests.

Two test cases are employed to verify the Phase to Amplitude CORDIC component:

1. Disabled. The real input data is passed through the worker and made available on lower 16 bits of the complex output.
2. Constant output frequency: A python script creates a file containing a constant value. The CORDIC produces a complex output frequency according to the equation 1.

The plots below show the input (real) and output data (Q-leg showing all samples, I-leg zoomed into one cycle) for testing a Phase to Amplitude CORDIC having the default parameter set. The input file shows a DC value of 2048, and 16384 real samples. The output file shows the converted complex waveform having  $DCvalue/2^{DATA-WIDTH}=2048/2^{16}=0.03125$  Hz, and 16384 complex samples.

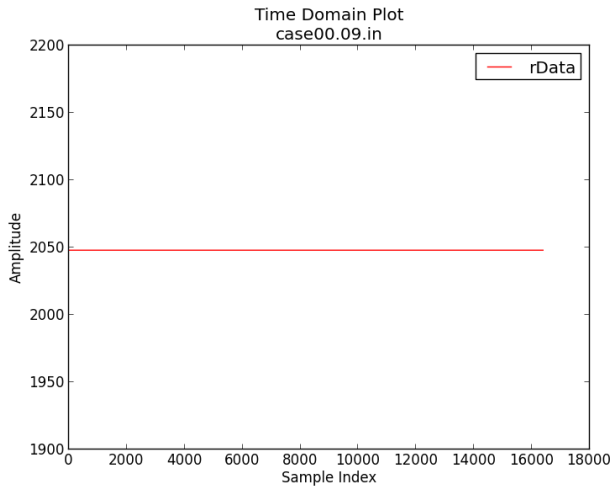


Figure 4: Time Domain

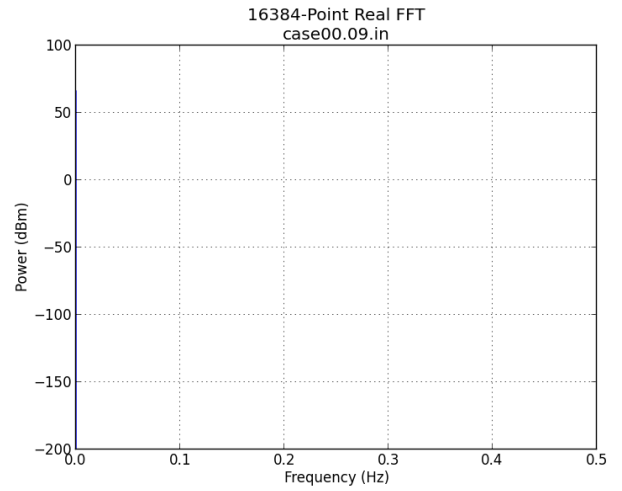


Figure 5: Frequency Domain

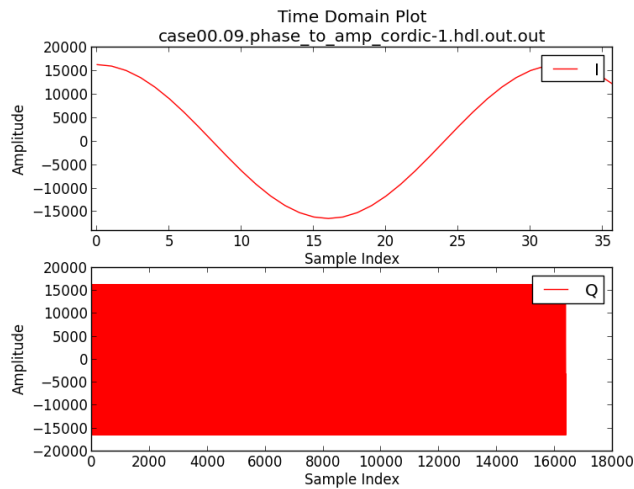


Figure 6: Time Domain

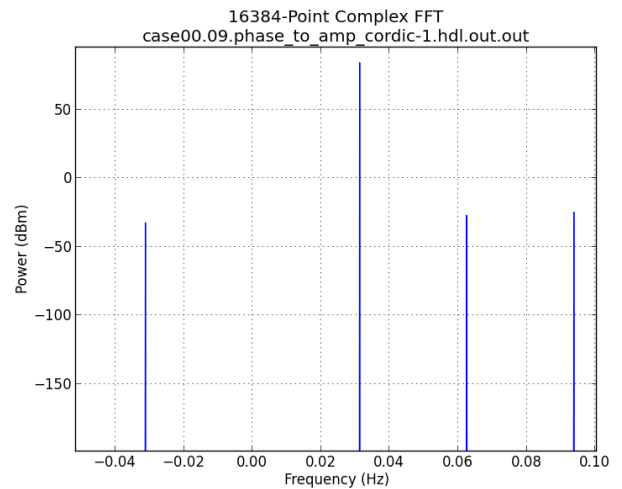


Figure 7: Frequency Domain