# PCIe to OCP Bridge
# Project Development Plan

Kevin Bedrossian
Peter Depeche
Benjamin Hunstman
Michael Walton

January 10, 2014

**Abstract**

PCI Express (PCIe) was created in 2004 and became into wide spread use soon after that due to the great benefits over PCI. It allows for increased bandwidth and flexibility that could not be achieved before. Revisions of the PCIe protocol continue to be developed giving increased bandwidth and functionality to the already robust communications bus. Open Core Protocol (OCP) is an openly licensed, core-centric protocol intended to meet system level integration challenges. It is an independent bus interface for on-chip systems for communications. These two protocols will be used to develop a System On Chip (SOC) that will allow for direct communication of the PCIe bus with a Virtual File System (VFS) supporting all memory transactions. This is to be accomplished by an IP block that will transparently bridge these two standards. The target device will be a Xilinx Spartan-6.

# Contents

# Chapter 1

# Introduction

A PCIe to OCP bridge IP block will be implemented to allow communication between a VFS. The design will be validated by performing Direct Memory Access (DMA) communication with an EMMC memory device connected to the FPGA platform.

# Chapter 2

# Design Plan

The following is the project plan that will be used in developing the bridge using a Hardware Design Language (HDL) such as Verilog and System Verilog.

## 1 Design

- Pseudo-code of core and test bench
- Write HDL/Verilog for simulation that is technology independent
- Analyze for functionality
- Revise HDL/Verilog design

## 2 Optimization

- Map HDL to Xilinx Vertex 6 logic circuits and functional blocks
- Optimize for technology dependence
  - Prepare for synthesis
- Timing Analysis
  - Check for speed, setup, and hold time
  - Form constraints
- Revise HDL

## 3 Place & Route

- Map (place) netlist to FPGA
- Route the structures (cores) on the FPGA to interconnect and perform the desired function
- Timing Analysis
  - Apply further constraints if needed
  - Move blocks around on the FPGA using a floor planning tool if needed to help timing
- Revise HDL

## 4 Validation

- Check design functionality
- Check if design meets performance goals

# Chapter 3

# Test Plan

This is a rough draft of the actual test bench that will be used to verify the bridge at each step of the design process.

## 5 Design

- TBD — Unit tests for design will be included in the design process

## 6 Optimization

- TBD — Validation for timing requirements and functionality after changes mandated by synthesis requirements have been implemented

## 7 Place & Route

- TBD — Validation for timing requirements using new analysis results and test for functionality after implementing hardware specific changes

## 8 Verification

- Test 1
  - Successful read of a single word from flash
  - Successful write of a single word to flash
- Test 2
  - Successful read of a burst from flash
  - Successful write of a burst to flash
- Test 3
  - Successful completion of ongoing DMA communications with flash

# Chapter 4

# Questions

The following is a list of questions that need to be answered to help facilitate the project in moving forward:

1. Can we get the FPGA board now?
   (a) Needed to test out the PCIe Core Gen from Xilinx to ensure a complete understanding of what we are working with
2. What exactly is provided for us in the system and has it been tested?
   (a) PCIe Core Gen
   (b) EMMC
3. What version of PCIe are we to target (1.0, 2.0, etc)?
4. Is there someone that can guide us or provide us with some direction or suggestions?