

Key Employee

Charles Guy

Jan. 1987 - present

[PDF format](#)

WORK PERFORMED FOR AVSYS CORPORATION:

Define, Patent and develop HDL models for Distributed Cache

Develop concept for distributing a cache across multiple CSU (Cache Storage Units. Patent this concept. Develop behavioral models in VHDL and Verilog to show the operation of this concept. Model is fully parameterized to selection of cache configuration including size, depth, width, elements, and units in operation.

WORK PERFORMED FOR AVSYS CLIENTS:

USB 3.0 superspeed host controller

Work with xHCI specification authors. Analyze implementation issues regarding the specified constructs. Provide input to make the controller not limit the ability to handle concurrent transactions. Help with modifications to clarify specification for developers.

Develop architecture specification for a xHCI controller. Full hardware implementation. Implement initial blocks of specification to start driver development. Hand implementation off to client implementation team.

Review and optimize VHDL for system validation board

Review and update VHDL code for 7 CPLD devices. Modify SMB bridge. Develop ability to support multiple secondary SMB segments from a single bridge. Add the ability to configure the number of secondary bus segments using generics.

Verify operation of all devices on system board in the lab.

VHDL Instructor

Instructor for VHDL courses with [Synthworks](#).

Teach [Comprehensive VHDL](#) and [VHDL Introduction](#) courses.

Write validation tests for TOE (TCP/IP Offload Engine)

Work performed for clients:

Work with a Specman E testbench to debug problems with a TOE. Tests also written in Specman E. Tests included self checking code for full TCP/IP protocol.

Define Architectures of 64 bit wide PCI-X to PCI-X bridges

Define the features and operation of non-transparent PCI-X to PCI-X bridge. Features included DMA, message passing, interrupt mapping, Hot Swap and data buffer management.

Define the features and operation of a transparent PCI-X to PCI-X bridge. Features included window mapping, data buffer management, asynchronous operation, Review the hardware microarchitecture for feature compliance.

Evaluate AGP master sequencer

Build a testbench using clients AGP target and AGP monitor to evaluate an AGP 4X master interface. Developed PERL programs to interpret and translate AGP 4X master tests into AGP target syntax so the disjoint parts could communicate.

Testbench consisted of mixed Verilog and VHDL blocks. The testbench itself was developed in Verilog. The AGP master was also a Verilog implementation. The client AGP target and AGP monitor were implemented in VHDL.

Tests run were mainly the ones furnished by the implement of the AGP 4X master interface. Additional tests were developed to test more rigorously test the AGP 4X master.

Validate network printer ASICs

Build the testbench and set up regression test suite to validate the operation of multiple ports for a Network Printer ASICs. Two versions of the ASIC implemented, the first had a single parallel port, and the second had two parallel and one serial port. Testbenches consisted of:

- ASIC
- BFM (Bus Functional Models) for
 - Serial port interface
 - Parallel port interfaces
 - 486 CPU
 - DRAM
 - FLASH
 - PCI master
 - PCI slave
- Simulation controller -- coordinates the start times of each interface

Generate tests to schmoo the start times from all the port interfaces and the CPU. DMA transfers to memory from all the port interfaces to a shared memory required verification that all memory accesses completed successfully. Patterns occurring at the memory refresh rate were also developed to ensure that refresh always occurred, but did not interfere with

Work performed for clients:

memory accesses from all sources.

Debug required fixes to all the interfaces to make them operate correctly in the testbench. Made design changes to the ASIC as required to make the ASIC function according to specification after all members of the initial design test moved on to other positions.

Develop VHDL synthesizable AGP master sequencer cores

Work with client to generate MASs (Micro Architecture Specifications) from AGP specification. Architect, design, code and debug AGP master sequencer model. Work through client with a third party to synthesize design. Work through client with a third party to debug the master sequencer design in a system level environment.

Two master sequencer designs were done the first using 1X clocking, and the second using 2X clocking. These designs were similar in structure but considerably different in implementation due to AGP idiosyncrasies.

Design included testbench and test driver. Testbench was standalone and could run all types of AGP cycles, varied clock frequencies and mode of operation.

Develop a USB host BFM for Intel Corporation

Conceive, Document, design, code and debug a USB host BFM from the USB specification. Model written in VHDL on a PC platform using Vsystem from Model Technology. The VHDL was then ported to three other simulation environments. Model used in the development of the 82930A controller.

Model issues USB activity at three levels:

- Bit stream
- Packet
- Transaction

Develop a master and target BFM

Design, code and debug BFMs to interface with a Reusable PCI core module. Command syntax used based on PCI models from Logic Modeling. Models developed using Vsystem from Model Technology on a PC platform.

Debug done with full environment including two PCI masters, two PCI slaves, PCI monitor, PCI core logic and BFM for target and master.

Provide consulting on the implementation of a PowerPC to PCI Bridge

Review proposal for PowerPC to PCI Bridge, then meet and go over written recommendations for improvement of proposal to improve system throughput and better meet PCI requirements. Answer questions regarding operation of PCI.

Develop a Performance analysis model for PCI.

Model at a somewhat abstract level operation of PCI and a host bus bridge. Use a statistical stimulus to drive transactions from the host and PCI to both the PCI bus and memory. Showed the effect of bridge buffers and memory speed on system throughput. Model tracked operation at a clock level.

Model was parameterized for:

- Number of buffers
- Direction of buffer
- Depth of buffer
- DRAM speed and type (EDO versus Fast Page)
- Results showed for the host and PCI masters:
 - contention for PCI and memory,
 - delay of accesses through bridge
 - maximum no wait state PCI burst length - up to indefinite

Patent pending on this work.

Work on the development of 80486 to PCI Bridge.

Provide direction for development of the CPU section of the board.

Direct electrical analysis of the interconnect signaling between processor, cache, DRAM and host to PCI bridge.

Review bridge specification and make recommendations on functionality and performance.

Provide input schematic for capture in Viewlogic.

Select Cache SRAM devices

Direct system validation effort

Design a 80486 based PCI master using a single EPLD for control logic.

Review simulation controller and make recommendations to improve functions.

Set up testbench to schmoo start time of PCI masters against start time of Host bridge.

Debug operation of system level model with multiple PCI masters and simulation controller

Set up infrastructure for system level regression testing

Support the port of environment to a second tool set.

Work performed for clients:

Develop an 8K gate ASIC to interface PCMCIA to ISA

Review PCMCIA specification and attend PCMCIA meetings to represent client views on PCMCIA. Conceive and document the "execute in place model" for PCMCIA. Conceive and document the use of multiple memory and I/O window maps for a PCMCIA interface. Write specification for and direct design of 8K ASIC to provide PCMCIA interface with multiple memory and I/O window maps. Functional chip done in less than three months, from start of specification.

Do analysis of the EISA bus for a client

Develop model of EISA bus trace using SPICE. Evaluate for 4 to 10 EISA slots. Write a guide for using EISA bus

Develop multiprocessor architecture

Develop SMP architecture with client for a tightly coupled multiprocessor environment using shared memory. Develop a bus that supports the SMP with totally shared resources across all processors in the system. Investigate the use of Futurebus+ as bus solution for client. Participate in Futurebus+ definition. Review protocol section and help direct the decision to use a MESI protocol instead of the MOSEI protocol.

Three patents were issued on this work.

Develop a high performance memory interface for client

Provide specification for high performance memory interface to take full advantage of memory interleaving and fast access modes.

Assist in the development of a Multibus II PC AT card

Provide consulting services to INMOS to design a set of cache memory devices

Provide the expertise for cache memory tag and cache memory data devices to INMOS a vendor of SRAM memory devices. Develop a set of requirements for the devices to operate in a system using multiple such devices. Device targeted to support either x86 or 68K processor architectures.

Patents

Developed and received patent for a method of distributing a cache across multiple identical cache subblocks.

Developed and received patents for Intel during various projects, in system multiprocessing, busing and interrupts.

Work Prior to Avsys Employment

Work performed for clients:

[AVSYS HOME](#)

[Contact](#)