Huayu (Howard) Fu

817 W 30TH ST, LOS ANGELES, CA 90007 |+1 (302) 563-6256 | huayufu@usc.edu | huayufu.xyz

SUMMARY

- **To obtain a full-time/intern position working as Hardware Engineer or related positions starting in January 2018**

Master Student in Computer Engineering will graduate in May 2018, specializing in ***Digital VLSI Design/Verification***, ***Static Timing Analysis***, ***Asynchronous VLSI***, and ***Computer Architecture*** with added emphasis on ***Object Oriented Programming*** and ***Embedded Systems*** and insight on *Engineering Leadership*.

- Good experience with digital circuit design (schematic, layout, DRC/LVS, Power Optimization, STA, ASIC, RTL, FPGA).
- Familiar with design tools (Xilinx ISE, Cadence Virtuoso, Design Compiler, IC Compiler II, ModelSim).
- Excellent programming skills (VHDL, Verilog, C/C++, Python, Perl, Tcl, Assembly).
- Proven ability to innovate and deliver results. Strong desire to be continuously learning.
- An effective Communicator and Motivator. Thrive in challenging, dynamic environments, applying the full spectrum of proven capabilities, to make key contributions to corporate profit improvement.

EDUCATION

University of Southern California, Los Angeles, CA — Expected May 2018
Master of Science, Computer Engineering, **Digital VLSI and Computer Architecture** — GPA: 3.61
University of Delaware, Newark, DE — Sep 2012 - May 2016
Bachelor of Electrical Engineering, Minor in Computer Science — GPA: 3.5

EXPERIENCE

Technical Intern, ***R&D, ECO team, Synopsys*** [C++, Tcl, IC Compiler II, gdb, Perforce, EDA] May 2017- Present

- Proposed and finalized base layer density modeling for density aware ECO cell placement with the potential to avoid violations in CMP process, increase the yield rate and reduce the cost of verification
- Implementing and testing the model in ***IC Compiler II*** on remote *Unix server* with *version control tools* and *gdb*
- Enabling high efficiency and high accuracy place-able region locating with the visualization on ***ICC II GUI***. Improving ***C++*** and ***Tcl*** programming skills. Gaining experience in ***large-scale software development*** and ***OOP***

Project Leader, *General-Purpose CPU ASIC Design Project, USC* [Python, Cadence Virtuoso] March- April 2017

- Designed and verified a fully functional 16-bit general purpose pipelined CPU, featuring 512-bit SRAM and 16-byte register file, supports add/sub, mul, and/or, SFL/SFR, and store/load instructions.
- Conducted in ***schematic design***, ***layout drawing***, performed ***Power & Delay optimization*** and employed ***power gating***, ***dynamic logic***, and ***DFF optimization*** techniques

Design Specialist, *Digital Neuron VLSI Design, USC* [Cadence Virtuoso] October - December 2016

- Designed a special-purpose digital circuit that mimics a neuron performed ***logic*** and ***timing verification***
- Won ***third place*** in course Design Contest with the area-delay product of 695 through the utilization of space saving technique (common source and Euler path), whereas the average of class is 7000

Design Specialist, *Music Visualizer Embedded System (SoC) Design, UD* [Xilinx, FPGA, VHDL, C, Python] February - May 2016

- Designed ***FPGA***-based Music Visualizer with VGA output using ***Xilinx IDE, EDK, and SDK***
- Created *VHDL* peripherals for fetching audio input, programmed audio processing and visualization algorithm running in ARM-based IP-core in ***FPGA*** board, and programmed controller to control Visualizer through ***UART*** port
- Empowered the ***FPGA*** board to output smooth video patterns to VGA port, with two different display mode

RELATED PROJECTS

Asynchronous Design of Network-On-Chip (NoC) with FEC [SystemVerilog, modelsim]: Designed a tree-structured *NoC* with router, arbiter, merge and hamming code module in *RTL* level. Generated gate level design through synthesis tool with *PCHB* templates. Performed functional verification with golden design, ensured deadlock free circuit

Dual-clock Asynchronous FIFO [Verilog, ModelSim]: Designed *Dual-clock FIFO* for cross clock domain communication with pointer synchronizer featuring gray code converter to increase MTBF

UNIX Kernel Programming [C, Linux, gdb]: Implemented a Linux similar operating system, *Weenix* in C programming language. Completed a UNIX-like multi-threaded/processed kernel with *Virtual File-System(VFS)* layer and a *Virtual Memory (VM)* layer

Multithreaded Programming [C, Multi-Threading, Linux, gdb]: Created multi-threaded program using *C* and *POSIX API*, emulated the process of token bucket with the statistical summary, with the concern of mutual exclusive and deadlock