Ben Pegg

benpegg23@gmail.com | benpegg23.github.io | github.com/benpegg23

Education

University of Illinois Urbana-Champaign Expected May 2028

Bachelor of Science in Computer Engineering

Relevant Coursework: Computer Organization & Design, VLSI Design, Digital Systems Lab, Data Structures & Algorithms

Skills

Engineering: VCS, Verdi, Git, Vivado, Vitis, Virtuoso, Innovus, FPGAs (Spartan-7 series), Bench Test Equipment (Power Supply, Oscilloscope, Function Generator, Logic Analyzer), KiCad, MCUXpresso, STM32, NXP LPC

Protocols: CAN, I2C, SPI, UART, AXI4

Languages: SystemVerilog, C, C++, Python, Assembly, TCL

Experience

Illini Solar Car

Battery Management System Lead, Electrical Team Member Sept 2024 – Present

- Redesigned the BMS PCB in KiCad after diagnosing critical hardware flaws in CAN, I2C, and power-path switching
- Led a team of 5 engineers to design fans firmware in C++ for an NXP microcontroller to implement PWM-controlled custom fan curves that optimize power consumption for the battery box and motor controller cooling system
- Team finished first at the Formula Sun Grand Prix 2025 in the single-occupancy vehicle class

Projects

RISC-V Out-of-Order Superscalar Processor

- Designed and implemented a 4-way frontend, 2-way dispatch/commit RV32IM processor using Explicit Register Renaming
- Advanced features included a GShare branch predictor, pipelined set/way-parameterizable L1 icache, split LSQ, writethrough post-commit store buffer, N-way superscalar frontend parameterizability, and more
- Synthesized core at 614 MHz on FreePDK's 45nm process node
- Ranked 5th out of 33 teams in performance competition across 11 benchmarks

FPGA IR Tracking and Parallax Graphics System

- Designed a real-time parallax graphics system on a Spartan-7 FPGA, utilizing IR light tracking to map the viewer's movement
- Wrote I2C drivers for an OV7670 camera to interface with the FPGA and optimize for light sensitivity and resolution
- Implemented a hardware-based centroid detection algorithm to calculate the IR light's coordinates and dynamically update the final multi-layer rendering on an HDMI display

Bitsliced RV32I Datapath Layout

- Constructed CMOS standard cell library for FreePDK45 process node in Virtuoso
- Utilized standard cell library to manually layout bitsliced design of an RV32I datapath, adhering to DRC and LVS checks
- Automated the place-and-route of processor control unit in Innovus, successfully integrating with the custom datapath layout

CHIP-8 Emulator

- Created a CHIP-8 emulator in C++ by implementing the complete fetch-decode-execute cycle for all 35 original opcodes
- Built the emulator's core architecture, including virtual memory, registers, stack for subroutine calls, program/index counters
- Integrated SDL2 library to handle emulator input/output by rendering the display, mapping the keypad, and generating audio