STEPHEN R. BENNETT

(612) 360-1129 • Stephen.Bennett@Colorado.edu
2865 Springdale Ln. · Boulder, CO 80303

OBJECTIVE

Seeking an entry level position to begin in the Summer or Spring of 2014.

EDUCATION

University of Colorado at Boulder *August 2013 – Present*

- **MS:** Electrical Engineering
- **Emphasis:** Power and Analog Electronics
- **Current GPA:** 4.000/4.000

University of Colorado at Boulder *Spring 2009 – May 2013*

- **BS:** Electrical and Computer Engineering
- **Minor:** Computer Science
- **Emphasis:** Communications
- **Final GPA:** 3.929/4.000
- **Dean's List** eight (8) consecutive semesters
- **ILR 3** Professional working proficiency in French

SKILLS

Proficient in:

Python · C · MATLAB & Simulink · SPICE · C++ · Perl · Git · Oscilloscopes · Ply · Linux Multithreading · Android Java and XML · Design Patterns (GoF, GRASP) · Continuous Integration · Vim · Eclipse · Linux CLI · UML · In-Circuit Emulators · Altium Electronics Designer (Protel)

Familiarity with:

Compiler Construction · Orthogonal Array Testing · Switched Mode Power Converters · Bash Scripting · Assembly (IA32, ARM, MSP430) · HTML5 · Band gap Voltage References · Op-Amp Analysis and Design · Regular Expressions · OrCAD · Mathematica · Win32 CLI · Verilog HDL · LaTeX

WORK HISTORY

Qualcomm Inc. *May 2013 – August 2013*

Software Engineering Intern

- Developed log parsing system in Python from scratch in a pair programming environment with a Scrum workflow
- All code and requirements were thoroughly documented in UML and Sphinx
- Went through the process of requirements elicitation and tracked the sources of gathered requirements
- Used test-driven development process to help ensure that the code was well tested and would be able to be passed off more easily
- Used Git and Gerrit for version control and code review

Qualcomm Inc. *May 2012 – August 2012*

Software Engineering Intern

- Worked on team responsible for development of Femto Site Modems
- Wrote, tested, and debugged Perl scripts and modules to automate continuous integration testing of all codebase changes
- Developed continuous integration testing framework that interfaces with a Wiki and Testlink server so engineers can immediately identify regressions or other issues
- Extensively documented work and changes and kept team up to speed
- Used Git and Gerrit for version control and code review

Blue Canyon Technologies *June 2011 – January 2012*
Engineering Intern

- Wrote, tested, and debugged C code for a National Instruments Data Acquisition Unit to interface with Simulink and reaction wheel rate sensors
- Gained MATLAB & Simulink experience implementing C code and assisting with the development of control algorithms
- Performed schematic capture and PCB layout using Altium Designer for form factored PCBs for final flight boards of "½U" CubeSat

Colorado Space Grant Consortium *January 2011 – January 2012*
Electrical Power Systems Engineer

- Developed, built, tested, and debugged Battery Charging and Protection Circuitry for "3U" CubeSat that interfaced with a maximum power point tracker and 918 solar cell array
- Work with Lithium Ion cells, solar cells, Buck/Boost converters
- Performed schematic capture and PCB layout using Altium Designer for form factored PCBs for final flight boards
- Performed requirement verification and tracking, along with setting and meeting hard deadlines
- Programmed MSP-430 microcontroller to communicate with Command and Data Handling processor over SPI
- LTspice logic analysis for power allocation (Separation Switch)
- Tested the satellite as part of the Reduced Gravity Education Flight Program at the Microgravity University

University of Colorado at Boulder *Fall 2010*
Tutor for C Programming

- Tutored C Programming Course that covered structs, linked lists, database management, and basic graphics implementation (Allegro)

PROJECTS

CU-Surrey Payload (CUSP) *Spring 2014*

- Student designed payload to test low cost microelectronics in space
- Currently scheduled to launch in 2015
- ITAR restricted project
- Tasked with programming microcontrollers, schematic capture, and PCB layout

Solar Panel System: Buck Converter *Spring 2014*

- Part of a larger solar panel system with a final output of $120VAC$
- DC/DC converter which bucks voltage from an $85W$ PV panel ($\approx 15V - 22V$) down to $11V - 13V$, using that power to charge a lead-acid car battery
- TI MSP430 controller (programmed in C) prevents battery overcharging and makes use of a *Perturb and Observe* maximum power point tracking algorithm
- No off-the-shelf magnetics used (inductor wound by hand)

Solar Panel System: 12: 200V Double Boost Converter *Spring 2014*

- DC/DC converter which boosts the same lead-acid car battery voltage as before to $120V - 200V$ to be used by follow-on $120VAC$ inverter
- Analog compensator designed manually to meet voltage set by MSP430 controller
- COTS PWM chip with chosen maximum duty cycle limit
- No off-the-shelf magnetics used (inductors wound by hand)

Solar Panel System: 120VAC Inverter *Spring 2014*

- The MSP430 microcontroller controls a H-bridge inverter, producing a 120VAC output suitable to be used with household electronics
- One MSP430 timer module provides H-bridge with two gate drive signals, while another timer output is fed directly into a low-pass filter to achieve a simple DAC, used with a feedback controller in the previous section of the system

TeslaBox *Fall 2012 – Spring 2013*

- Software and Communications Lead on a senior capstone project consisting of a RF-shielded enclosure which uses radio waves to wirelessly charge a Li-Poly battery
- Charge control and monitoring MSP430-based, makes use of SPI busses to wireless communication using TI CC1101 transceivers
- Developed touch screen GUI to interactively view battery charging data and device status as well as send commands to the charging device inside the enclosure
- All hardware (SEPIC board, power management boards, base station board) design in Altium Design for schematic capture and PCB layout
- Serves as a proof-of-concept for wireless charging of consumer devices such as cell phones, toys, etc. using the far-field
- Mentored and sponsored by Dr. Zoya Popović

Python Compiler *Fall 2012*

- Compiler written in Python to compile Python code to x86 and ARM assembly using ASTs and the visitor pattern
- Supported Python semantics and was completed with support for:
    - (Nested) Classes
    - (Nested) Functions
    - (Nested) Control Flow (For, While, If-Then)
    - (Nested) Lists and Dictionaries
- Ported to compiling to ARM assembly once completed
- The project was done using test-driven development in a pair programming environment

EEG-Android Snapper+ *Spring 2012*

- Modified the source code for the AOSP camera app to support input from a rudimentary EEG monitor
- EEG signals were taken from three copper tape electrodes placed on the user's forehead
- ARM Cortex-M0 (programmed in C) performed two functions:
    - Sample Left/Center/Right EEG channels with ADC
    - Transmit sampled data over UART to a UART-Bluetooth module
- Android smartphone performed all other functions:
    - Receive Bluetooth data using a service
    - Modified camera app binds to Bluetooth service to receive EEG data
    - Translate received data into facial actions (smiling, blinking, scrunching forehead, etc.) which were then translated into a camera actions (zoom, focus, take picture)

NES Rover *Spring 2011*

- Arduino-based project which wirelessly controls a three-wheeled robot with a NES controller
- D-pad controls direction, A and B buttons control speed
- A set of six non-inverting amplifiers gave each controller input a specific output voltage which was then translated into a distinct PWM signal
- After the wireless link, the PWM was low-pass filtered and interpreted by an Arduino, independently controlling left and right DC motors

Android – Utility Belt *Fall 2010*

- Developed an Android application that possessed the following functionality;
  - Simple note taking (title, content)
  - Basic Google Maps integration with markers that tracked position over time
  - Tip calculator
  - Customizable color scheme

12-Hour Clock *Fall 2009*

- 5*V* powered, 12-hour clock designed using Altium Designer
- Full design simulated and verified using Electronics Workbench prior to schematic capture and PCB layout
- Used 7400 series ICs
- Received highest grade in class for most compact layout and highest overall functionality

ENGINEERING COURSEWORK

ECEN 5643	Software Engineering of Concurrent Systems
ECEN 5543	Software Engineering of Standalone Programs
ECEN 5523	Compiler Construction
ECEN 5018	Graduate Projects I
ECEN 4593	Computer Organization
ECEN 4113	UNIX Systems Administration
ECEN 3754	Operating Systems
ECEN 3350	Programming of Digital Systems
ECEN 3000	Digital Design Laboratory
ECEN 2703	Discrete Mathematics for Computer Engineers
ECEN 2350	Digital Logic
CSCI 2270	Data Structures

ECEN 5827	Analog Integrated Circuit Design
ECEN 5797	Introduction to Power Electronics
ECEN 5517	Power Electronics and Photovoltaic Lab
ECEN 5224	High Speed Digital Design
ECEN 5017	Power Electronics for Electric Drive Vehicles
ECEN 5002	Communication Lab
ECEN 4242	Communication Theory
ECEN 3400	Electromagnetic Fields and Waves
ECEN 3300	Linear Systems
ECEN 3250	Circuits and Electronics 3
ECEN 2020	Circuits as Systems 2
ECEN 2250	Introduction to Circuits and Electronics 1