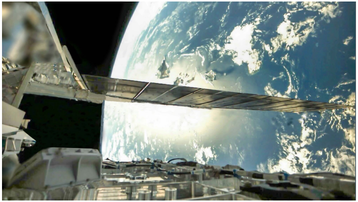
Portfolio

Andrew Fearing

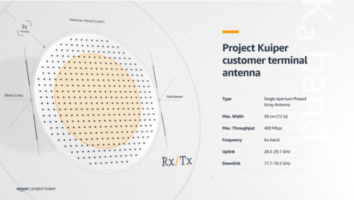
June 2024

Digital Beamforming ASICs

Amazon.com Project Kuiper, Jun 2020 – Aug 2020, May 2021 – Aug 2021, Feb 2022 – Present



KuiperSat in Low Earth Orbit (Amazon)

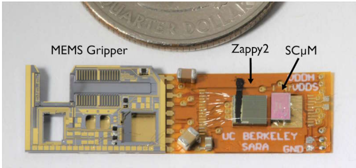


(Amazon)

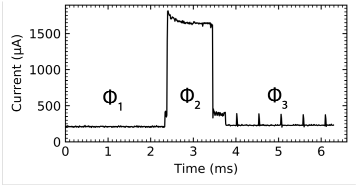
Project Kuiper is a LEO satellite constellation for broadband internet access. As part of the ASIC Systems team, I develop the MATLAB signal processing model for our digital beamforming (DBF) chips. I support the design verification team by providing reference data for bitmatching. I've contributed to bring-up of the customer terminal (CT) DBF ASIC by writing and debugging the firmware, both pre-tapeout on the Palladium Z2 emulator and post-tapeout in lab.
I own the nightly regression testing for the MATLAB DBF model and have written unit tests for the DBF firmware CI/CD infrastructure.

Single-chip Micro Mote (SCμM) applications

UC Berkeley Autonomous Microsystems Lab, August 2019 –- December 2021



MEMS gripper, Zappy2 solar cell, and SCμM on SARA board



VBAT current during wireless 802.15.4 receive. Φ_1: Low power (F_{cortex}=78 kHz), Φ_2: RX with radio on, Φ_3: periodic wake up to F_{cortex}=5 MHz

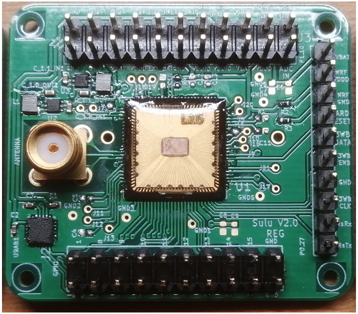
SCμM is a crystal-less CMOS SoC which features an ARM Cortex-M0, standards-compatible 2.4 GHz RF communication, contact-free optical programming, and requires only a power supply to run. I tested integration of SCμM into the Small Autonomous Robot Actuator (SARA) system. SCμM microrobots, 2020 GOMACTech

SARA uses a Zappy2 chip to provide solar power to SCμM. Because the solar cell does not provide enough power to run SCμM's radio, I programmed and validated the low-power mode for SCμM. The low-power mode reduces the CPU clock from 5 MHz to 78 MHz to charge a capacitor. Once the capacitor is charged, SCμM wakes up and increases the clock speed back to 5 MHz for transmit/receive.
SCμM as wireless thermometer, 2020 IEEE Sensors

SARA also includes a pad to mount MEMS actuators. Using the same low-power mode to charge a capacitor as described above, SCμM wakes up to send a square wave signal to a MEMS actuator. The MEMS gripper uses an electrostatic inchworm motor to move the jaw up to 3 mm. SARA with MEMS Gripper, 2021 ICRA

Single-chip Micro Mote (SCμM) Development PCB

UC Berkeley Autonomous Microsystems Lab, August 2019 – December 2021



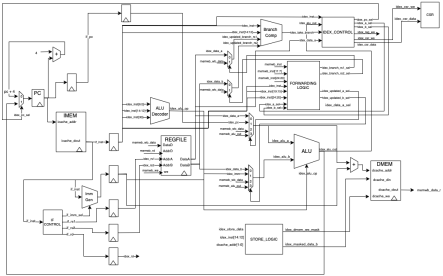
SCμM development board, Sulu V2.0

Sulu is a 36 mm × 42 mm PCB for experiments with the SCμM chip. I designed Sulu to be small and cheap, but provide the necessary features to run SCμM untethered. Sulu features DC-DC switching regulators for a 3 V coin cell battery to power SCμM and an ICM-20948 IMU at 1.8 V. The headers on the East side can connect to an nRF52840 DK via a ribbon cable. The North headers are for power. The south headers break out the SCμM GPIOs. The West side is clear for an SMA antenna.
Sulu V2.0 release

RISC-V CPU

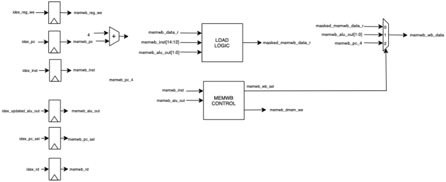
Final Project for EECS 151LA: Application Specific Integrated Circuits Laboratory

Teammate: Eduardo Diaz



IF and IDEX stages

In this project we implemented a 3-stage CPU in Verilog HDL. The CPU uses the RISC-V instruction set architecture. For the front-end, we designed our pipeline diagram and verified its implementation using Synopsys VCS simulation. We wrote unit tests and test vectors for verification of each module. We also built a direct-mapped cache for our CPU. For the back-end, we synthesized our design using Cadence Genus. We used Cadence Innovus for PAR. Our submitted CPU design's maxmimum frequency was 2.4 MHz and could run a suite of C and Assembly scripts.

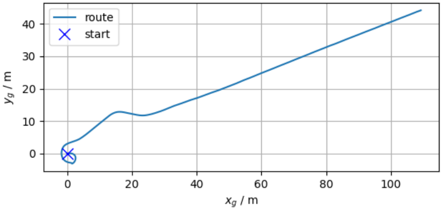


MEMWB stage

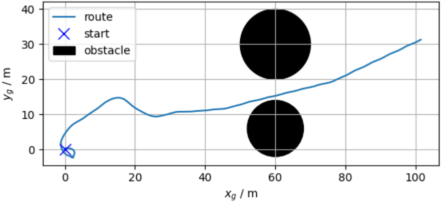
Sailboat Control and Path Planning

Final project for EECS 106B: Robotic Manipulation and Interaction, Spring 2021 (link)

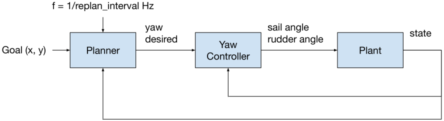
Teammates: Neelay Junnarkar, Hamza Khawaja



Sailing on open seas



Sailing through a narrow channel



Controller block diagram

In this project we developed a planning and control method for a sailboat to maneuver autonomously through obstacles. The path planner is implemented using CasADi and the IPOPT solver. We used the Python stda-sailboat-simulator to model the dynamics of our sailboat.