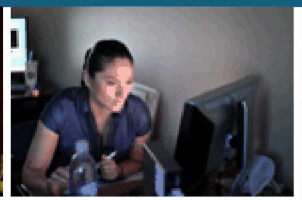
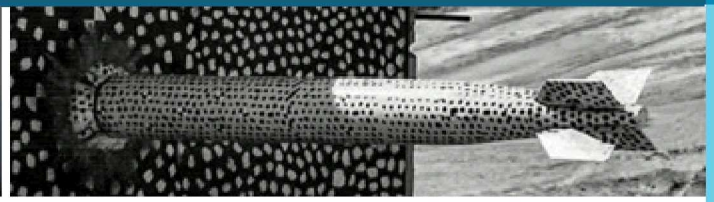
SAND2019-2479PE

CO School of Mines - Sandia - PV & Materials Tech - Python





PRESENTED BY

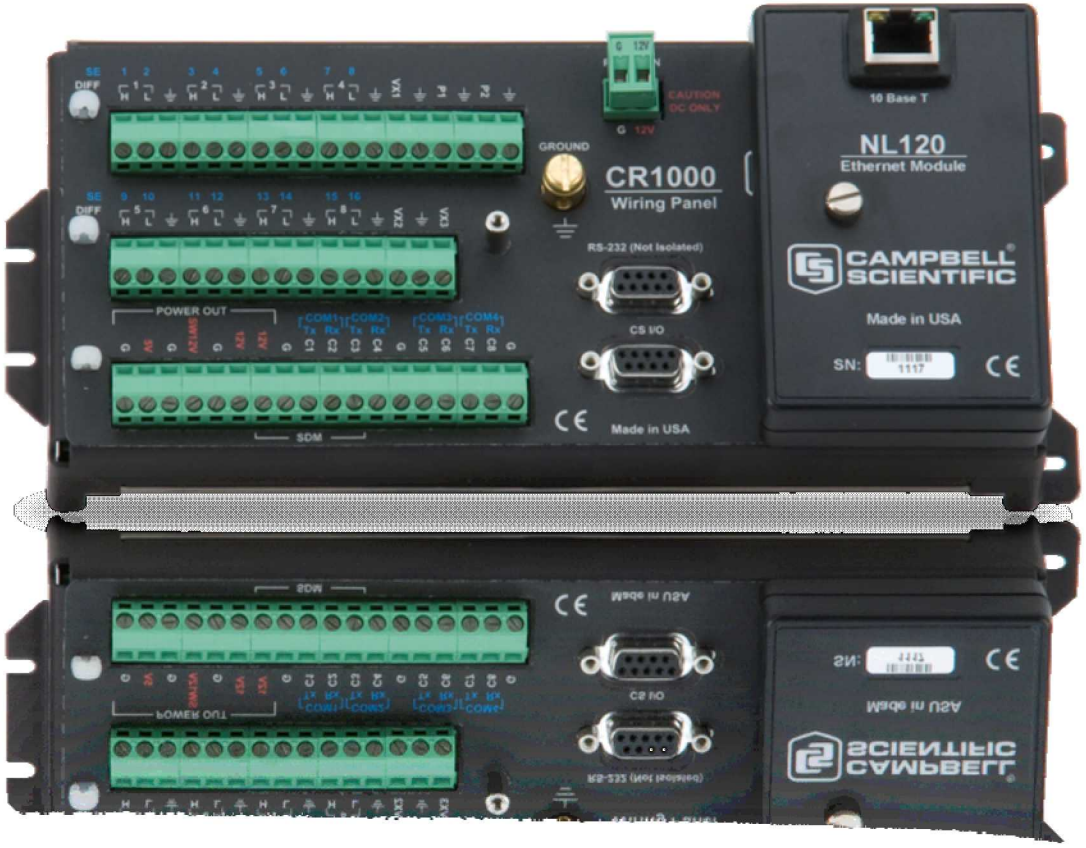
Craig K Carmignani

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

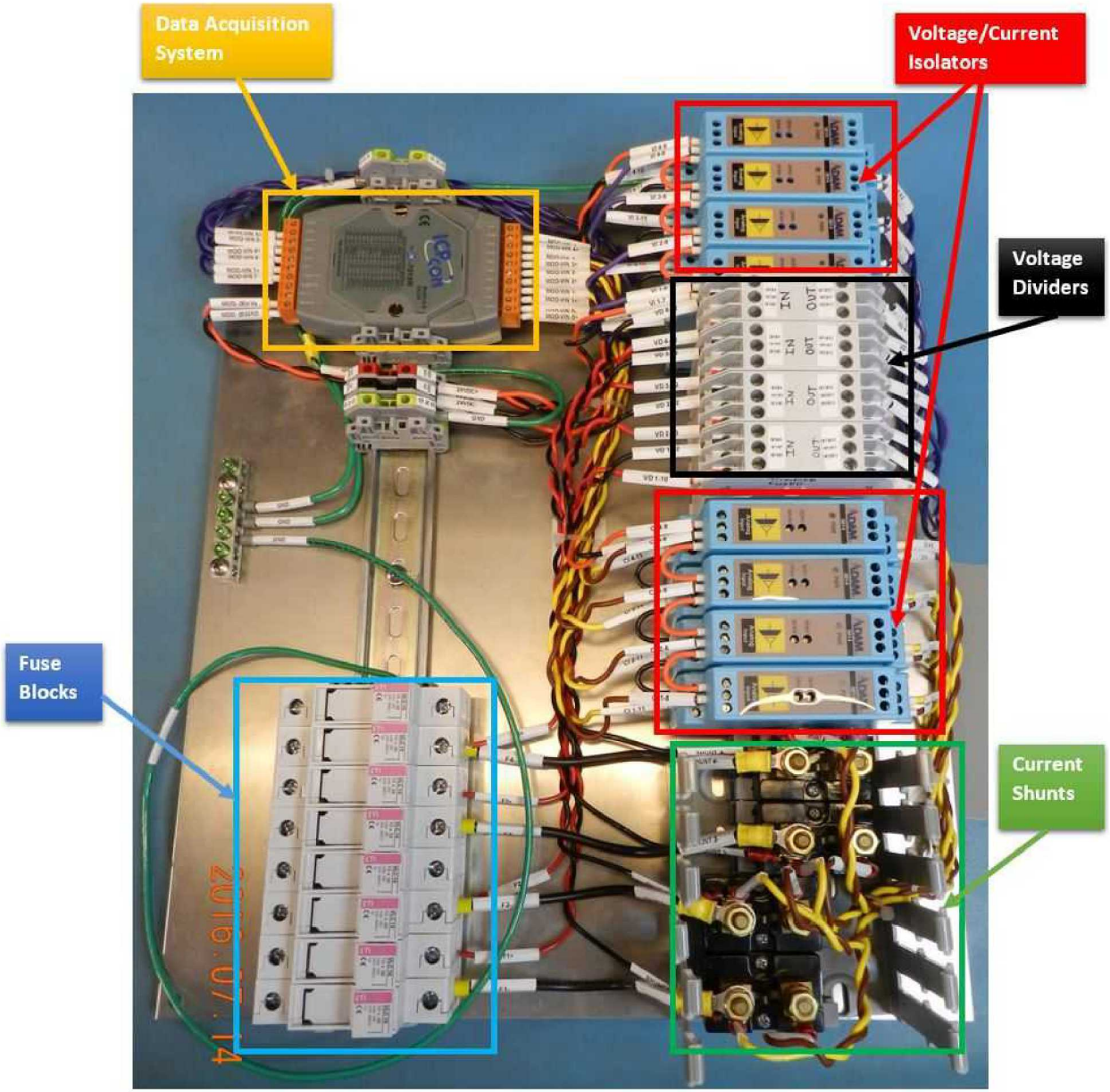
Craig Carmignani - History

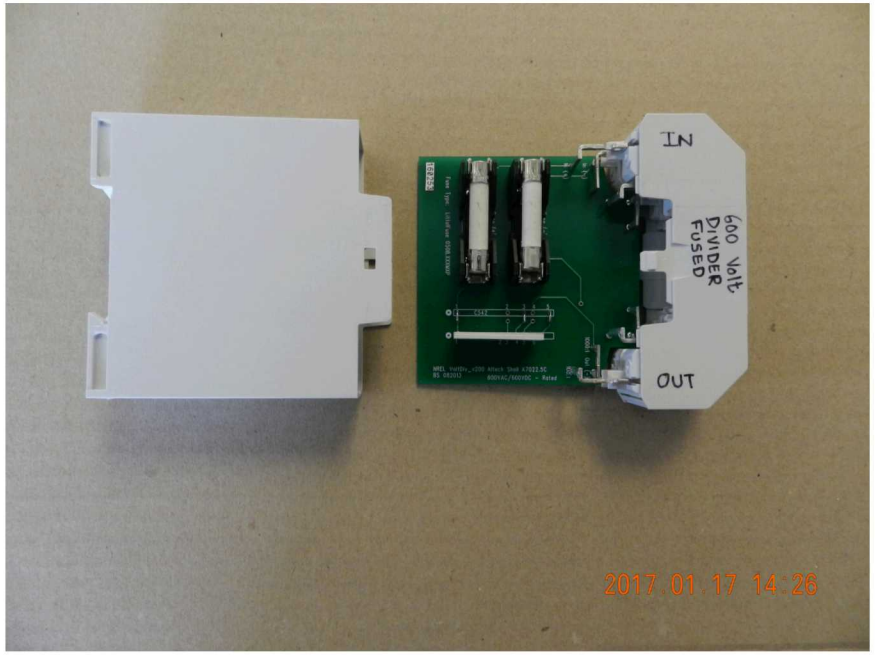
- Graduated NMSU with 4 year BS in Electrical Engineering Technology
  - **Programming** classes: **Fortran 77** and **C**
  - Work Study: DEC MicroVAX - Unix administrator (1st exposure to thing called the **internet**…)
- 2 years - DOD Theater Air Command and Control Simulation Facility (**C**)
  - Gov't, at the time, required all programs to be written in **Fortran 77**
  - Small subcontracting company to prime contractor; Lockheed Martin
  - 8 hour days in a refrigerated basement
- 1 year at UNM working towards an MS EE
- ~25 years - Sandia National Labs
  - 8 years - Microelectronics Fab - Contractor (small company)
    - Designed, built, & operated a variety of T&M systems (Lots of **LabVIEW** & **Rocky Mountain Basic**, some **C++**)
    - Graphical programming, LabVIEW, at the time was frowned on by classically trained programmers
  - 6 years (hired into Sandia) – Calibration and Maintenance (some **LabVIEW** development)
  - 2 years – Lithium Ion Battery Abuse Test Lab
    - Designed, built, & operated a variety of T&M systems (Lots of **LabVIEW**)
  - 9 years – Photovoltaic Test Lab (indoor/outdoor)
    - Designed, built, & operated a variety of T&M systems (**CR Basic**, **Python,** some **LabVIEW**)
- Early Retirement: ~10 years??? Start saving Early & Often…

PV Data Logging – **COTS** (proprietary) or **Open Source** (RPi + Python)



PV Data Monitoring – Sandia Boilerplate Data Acquisition System (Modbus)





Transducer Example:
Sandia Resistive Voltage Divider

RPi / Python: Issues, Ideas, and Topics

Duplicating SD card image
- SD cards of the same size are NOT the same size.
- What works for me: Create a 'Golden' Image on smaller card then, image to larger card and expand. 8Gb Golden imaged to 16Gb new deployment; 4GB too small.

Start project with latest stable distribution of Linux for RPi; Jessie=>Stretch Lite=>Stretch (full); many issues.

Date/Time – critical ± ~1 second for data monitoring
- Real Time clock - No ethernet, SNL security, etc.
- Stretch, by default, uses lightweight and dumb SNTP client; systemd-timesync. RTC & Local Timeserver.

Outdoor operational reliability
- No issues in NM with RPi; ref BBB clock stability
- Considering using RPi Compute Module
- Using Single-Level memory for increased write-cycles

Python
- Minimalmodbus=>Re-tries, Bluetooth disabled (GPIO),
- Moving to Object Oriented, JSON, Rev control (.git),

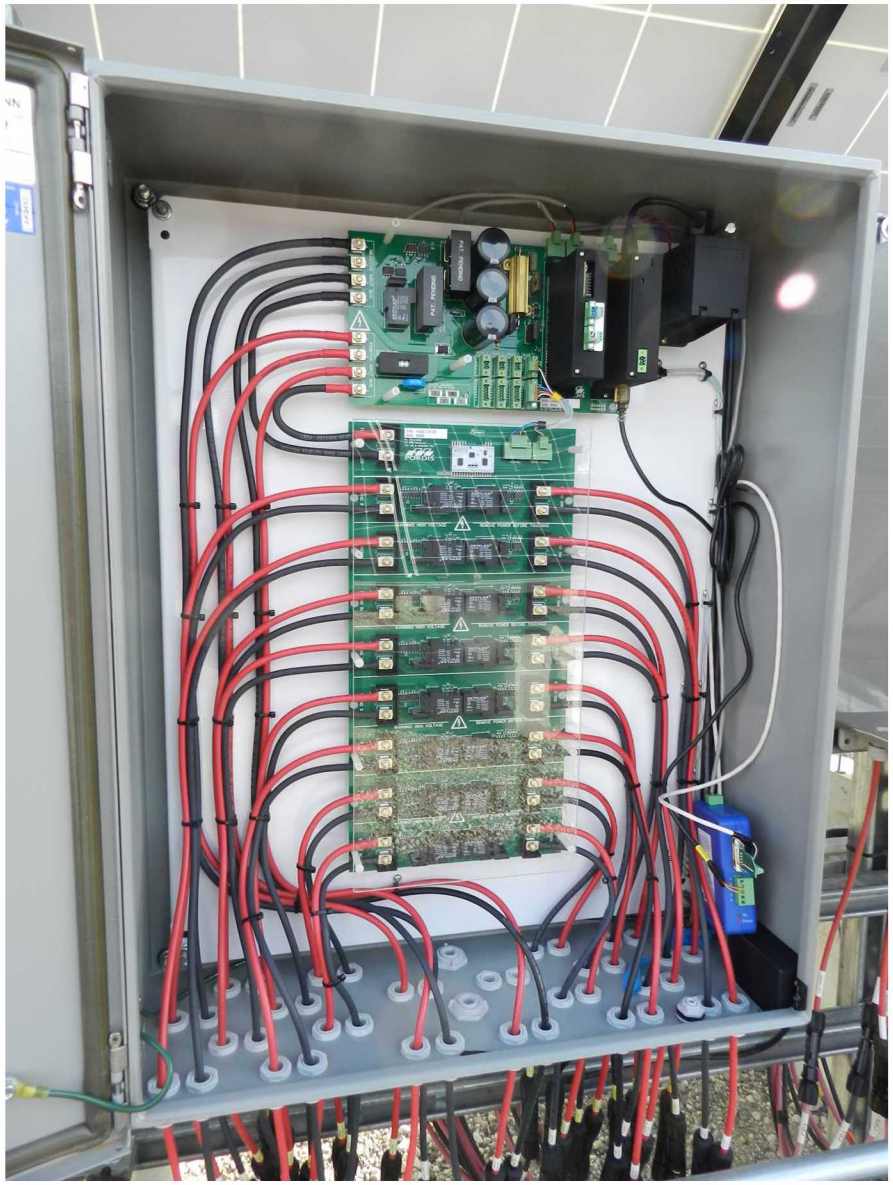
Power Supply (5V)
- Field 24V standard; ref StratoPi (9-28V, UPS, RS485)
- Reference Floating power idea(s) (Qi, PV,)

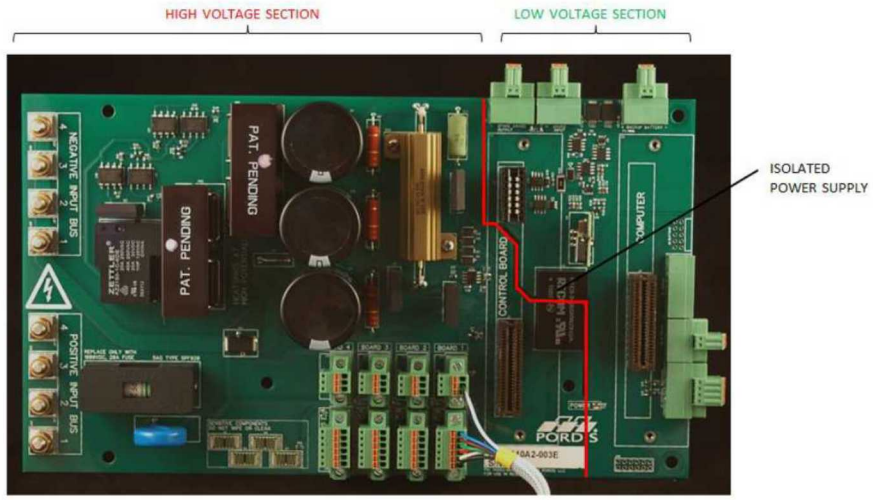
Data
- Analysis/Management/Sharing; ref Campbell
- Moving from flat files (CSV) to MariaDB; a 'community developed fork of MySQL'

BeagleBoneBlack issues
- Outdoor use demonstrated unstable clock
- Linux ethernet disconnect issue
- Bricking with power loss.?.?

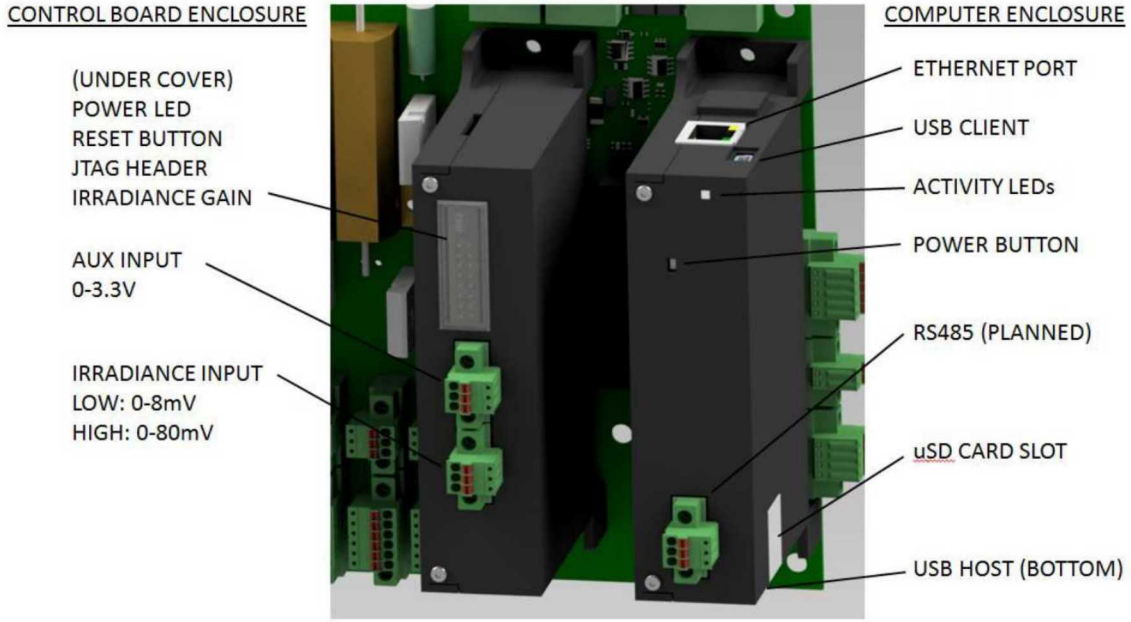
Ideas
- MicroPython (CircuitPython) – AdaFruit Feather
- LabVIEW (Digilent-PMOD) with RPI & Arduino
- Mooshimeter=>DMM with BLE
- SBC Alternatives: BBB, Snapdragon, PandaBoard, OrangePi, BananaPi, Intel Galileo, etc…

In-Situ PV String IV Sweep – Pordis model 140A2





Python - Modbus comm
PERL - Sys Op's & MySQL
C – Control board comm (IV sweep, UPS, etc).
Javascript -Web interface



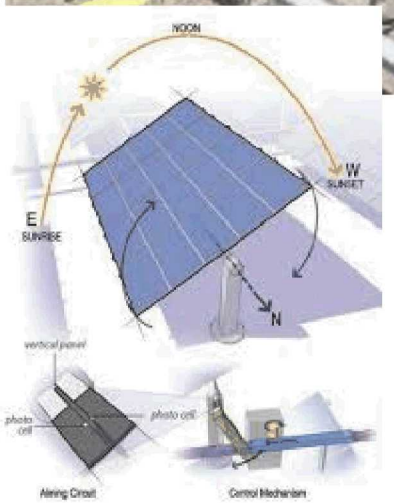
Open-Source Modeling Tools

Open-source software allows the National Labs to share new methods with industry stakeholders. Examples….

PVLIB Toolbox (Python and Matlab) –Set of over 50 documented functions that allow users to build their own sophisticated performance models of PV energy systems. (Sandia)
https://github.com/pvlib/pvlib-python
https://github.com/sandialabs/MATLAB_PV_LIB
System Advisor Model (C++) – Performance simulations for many RE technologies. Recently released as open source (NREL) **https://sam.nrel.gov**
Bifacial_radiance–(Python) Wrapper functions for using RADIANCE (ray-tracing S/W) to model bifacial PV performance (NREL) https://github.com/NREL/bifacial_radiance
BifacialVF– (Python) Bifacial PV View Factor model (NREL) https://github.com/NREL/bifacialvf
High Speed Computing at Sandia - Applying above PV Bifacial modeling to computationally intensive scenarios (Sandia) Code hurdles: OS specific (Windows-to-Linux), race conditions (parallel processing), etc
GridPVToolbox (Matlab) –Models and simulates the impacts of PV on the distribution grid. (Sandia)
Wavelet Variability Model – Geographic smoothing of irradiance variability over a PV plant footprint. (Sandia)
WNTR – (Python) Water Network Tool for Resilience. A Python package to simulate and analyze resilience of water distribution networks under disaster scenarios (EPA collaborative project with Sandia)
https://github.com/sandialabs/WNTR
Chama – (Python) A package to optimize (Pyomo) detection with sensor placement and technology (Sandia)
https://github.com/sandialabs/chama
PECOS – (Python) A package for analyzing (quality control tests) and reporting on timeseries data sets
https://github.com/sandialabs/pecos

Ground Coverage Ratio


S15Wht
W15Wht
S30Nat
W90
S90


NOON
W
SUNSET
E
SUNRISE
N
vertical panel
photo cell
photo cell
Aiming Circuit
Control Mechanism