Opensource in neuroimaging

Ben Dooks

Disclaimer / Warning

- I am not a medical profesional
- Research before attempting anything
- May not be legal in some areas

Introduction

- Me
  - Senior engineer and consultant at Codethink
  - Linux kernel contributor
- Why
  - Brains (what does the 1.2kg in your head do?)
  - Open hardware and software opens up study
  - Involved in producing such a scanner
- Caveats
  - See previous warnings
  - Not in depth
  - Most eye-catching / top google examples used

Brains – not just tasty zombie food

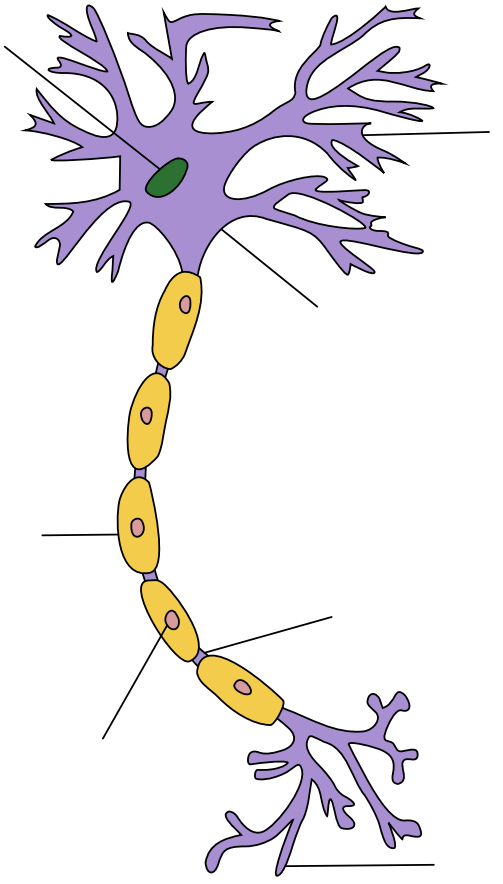
- Neuroimaging is determing the brain properties
  - Structure
  - Function
  - Pharmacology
  - See also encephalography
- Why study the brain?
  - Medical
  - Psychology
  - Person/machine interface

Structure

- Difficult to view (without $$$$)
  - Non-invasive scans
    - fMRI
    - PET
    - x-ray
  - Invasive biopsy
- Open databases of medical scans
  - Wikipedia lists 3000
  - Data-sets at OpenfMRI (creative-comons)
  - Freesurfer tool for processing fMRI data

Neurons

- The hardware building blocks
  - Approx 1um size (excluding comunication links)
  - Several different types
  - Some specialisation per task
- Approx 80-100 billion per brain
- Communication
  - Links to more neurons (synapse)
  - Chemical (neurotramsitter)
  - Electrical (0.1-1V)

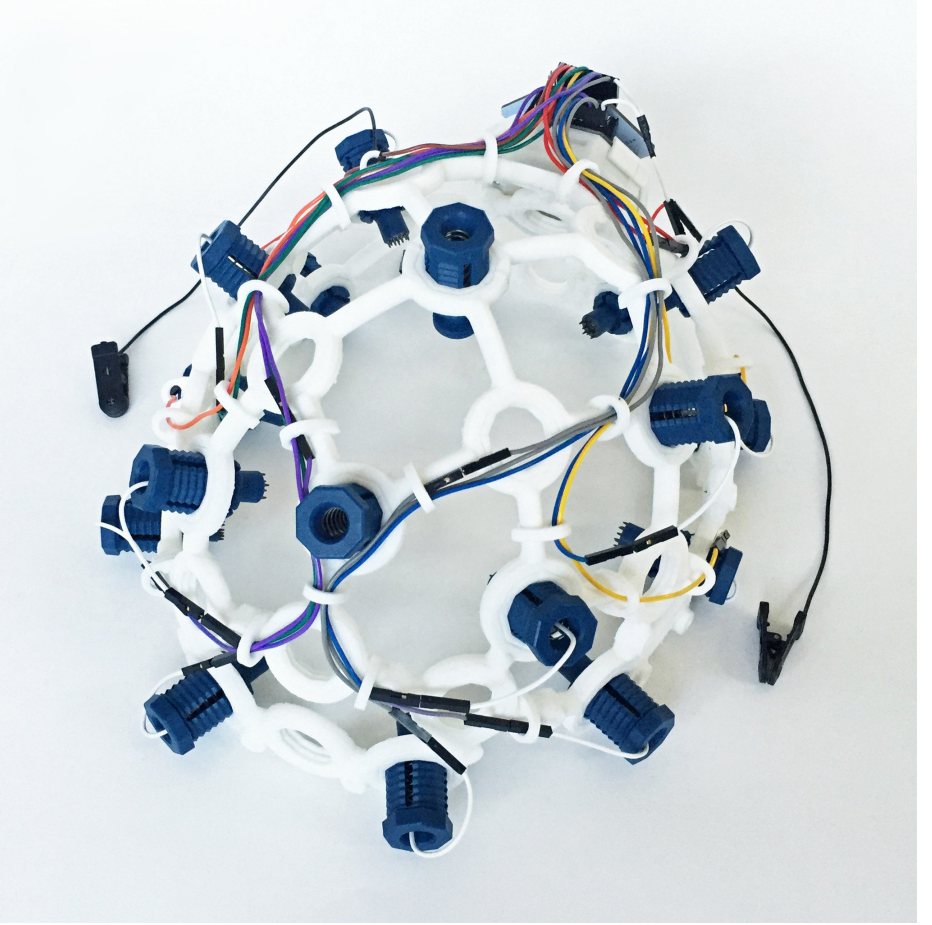


EEG

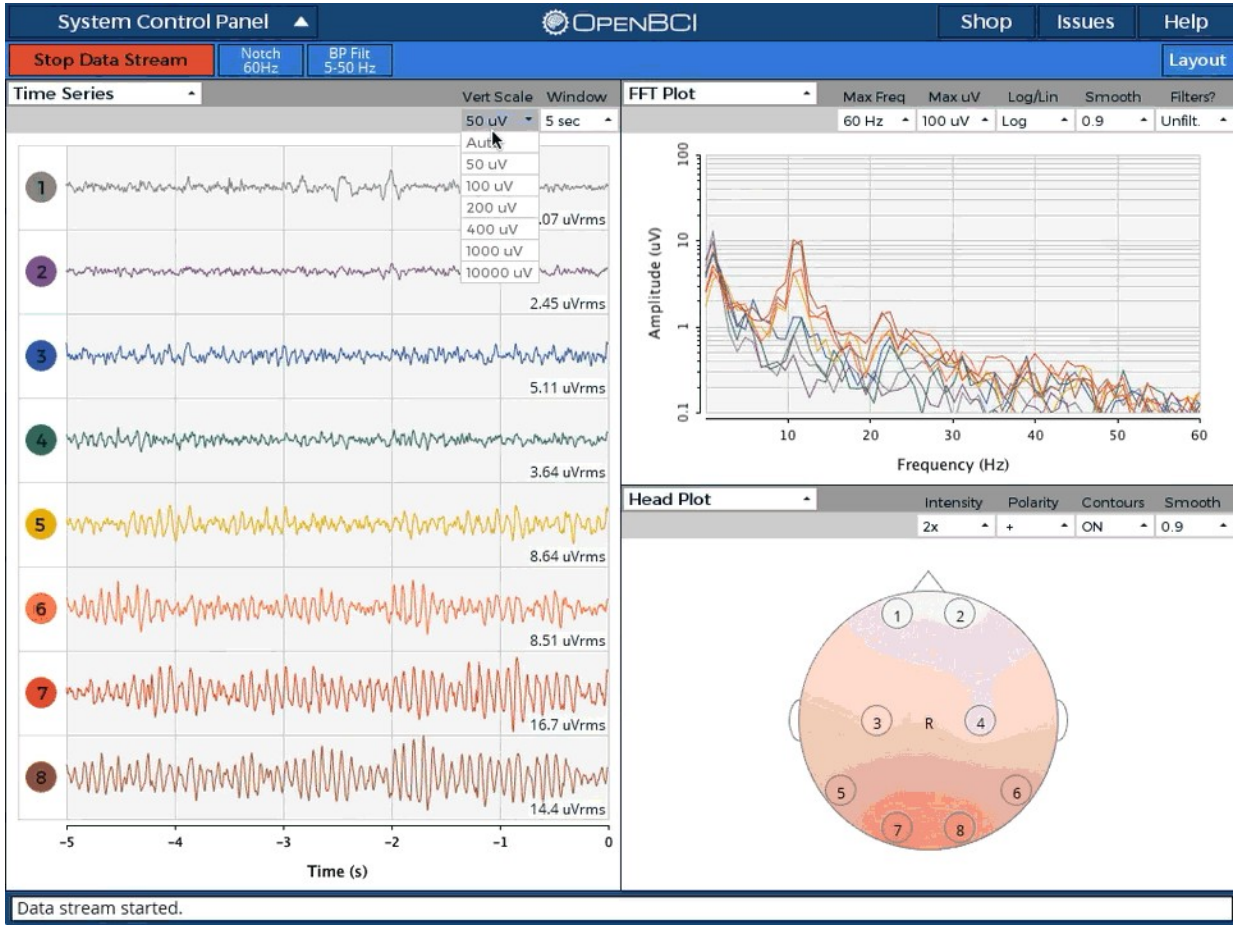
- Measures electrical activity at scalp
  - Number of electrodes attached to scalp
  - Reference electrode for signal difference
- Groups of neurons produce electrical waves
  - This is often 10-100uV range
  - Frequency is 1-100Hz (approx)
- EEG is simple
  - Non-invasive
  - Cheap equipment
  - Not pre-condition limit

EEG example (openbci.com)

- Open hardware
  - Kickstarted in 2013
  - Can add EMG and EKG
  - Evolving
- Open software
- Example ultracortex-mk-iv
  - Up to 16 channels
  - DIY or buy pre-built
  - Board not included



EEG example (openbci.com)

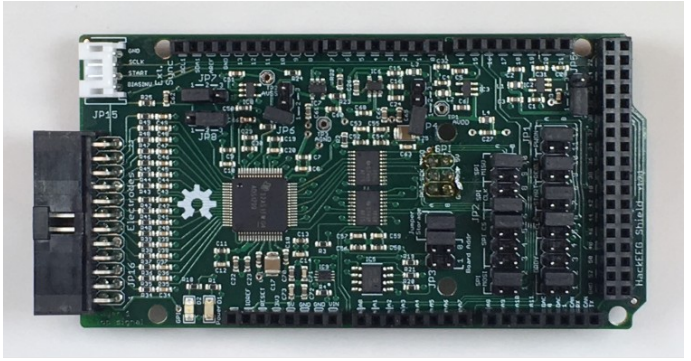


Open EEG projects

OpenEEG
openeeg.sourceforge.net
EEG for the rest of us!

Brainstorm

OLIMEX



HackEEG

MEG

- Magnetic sensing
  - Most neurons make small magnetic fields
  - These are in the 10fT region
  - Can be more accurate than EEG
  - Similar time responses
- Issues
  - Standard background noise is 1000fT
  - Shielding is necessary for systems
  - This makes them expensive
  - Not all neural activity produces detectable fields
  - Is not currently cheap

Our big MEG project

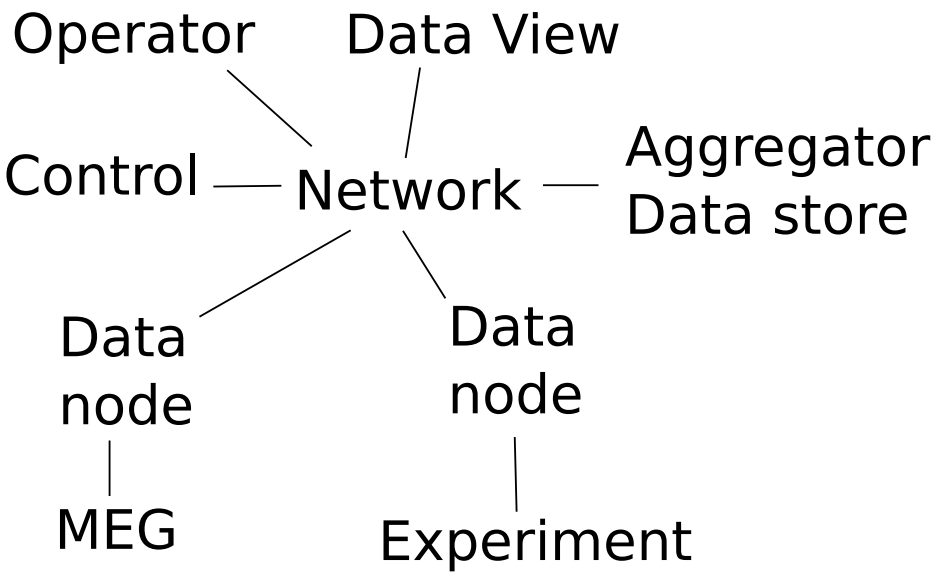
- Why do this
  - Latest technology from late 1970s / early 1980s
  - Legacy devices becoming difficult to repair
- Goals
  - Improve the technology
  - Use open-source where possible
  - As much as possible open sourced

Why open?

- Started as a university research project
- Project longevity
- Peer review
- Concentrate on the hard problems
- Security (and seen to be secure)

Overview

- How it fits together
  - Data capture nodes
    - MEG
    - Other
  - Data streaming
  - Experiment control
  - Data storage
  - Real-time view



Hardware

- Physical scanner closed design
  - No longer needs liquid helium
- Aquisition to 600 channels, 24bit data, 72kHz max
  - About 3400KB/sec per 8 channel node max
  - 250MB/sec for complete cluster
- ARM and FPGA data gathering nodes
  - Needed many, ARM is low power
  - FPGA for quick real-time solutions
- Comodity PC and PC servers for the rest
  - No need to specialise these

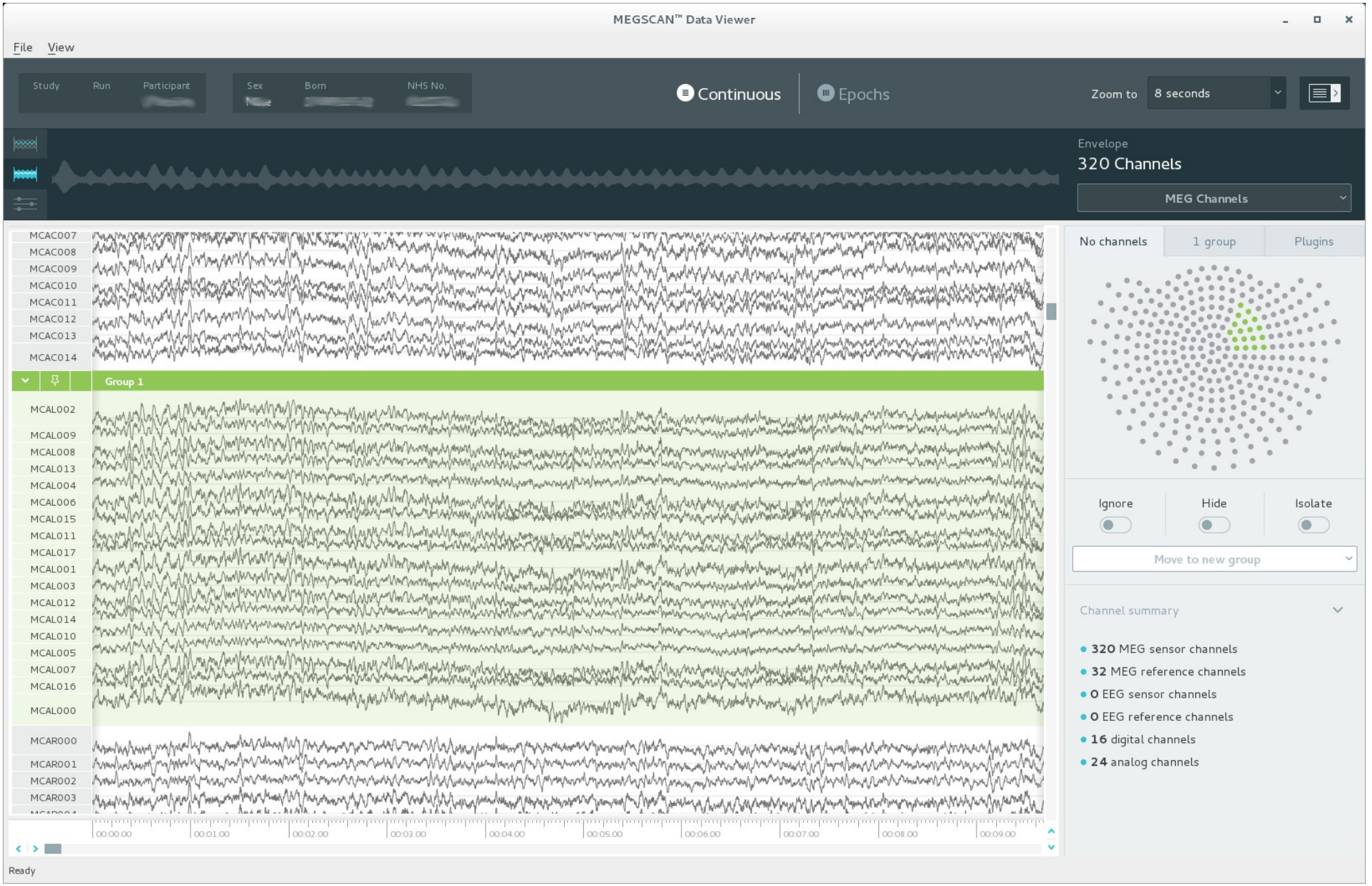
Software

- Debian based
  - Supports ARM and x86
  - Stable and maintained
  - Customisable
    - Netboot - No hard-discs allowed, not enough flash
  - Already had debian developers on hand
- Standard data recording (HDF5)
  - Widely used and well understood
  - Designed for large time-based data-set

Software #2

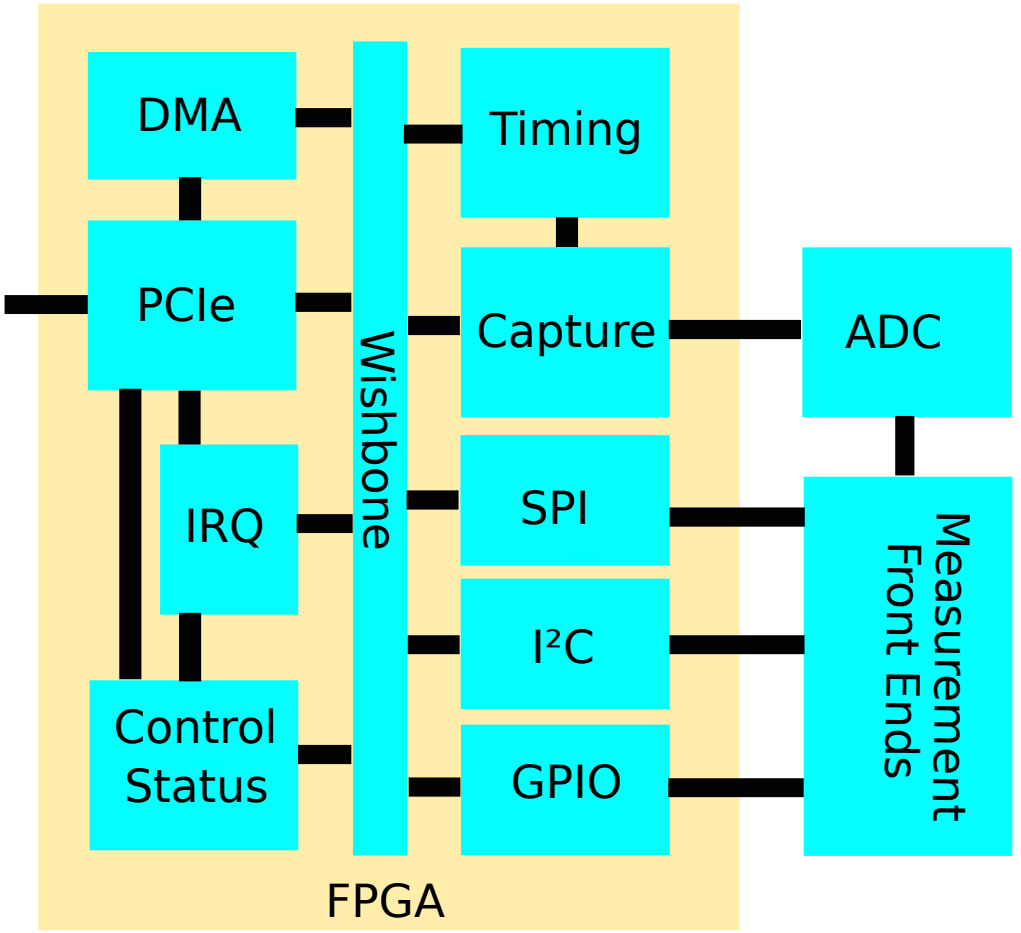
- Qt & OpenGL
- Python packages
  - NumPY
  - Arrow
  - H5Py
- U-boot
- Some custom control software
  - Synchronisation
  - Data streaming and verification

Data display



FPGA

- Real-time sampling
  - 3ns between nodes
- OpenCores
  - Wishbone bus
  - SPI
  - IIC
  - PCIe to WB
- Vendor IP for PCIe
- Rest is closed VHDL



Kernel

- Easy to update
  - Vendor kernel already close to mainline
  - Tracking mainline has required a few API updates
  - Whole process has been easy

- Simple driver to split PCIe up
  - Instantiate SPI, GPIO and I²C devices
  - Provide stream device for data stream
  - Add sysfs files to access overall state

Review

- Debian is a good base
  - Upgraded from 6 to 9
- Open FPGA tools still lacking
  - No open toolchain or IP cores
  - Makes reproducable builds difficult
  - The PCIe core was difficult to debug
- Open PCB design
- At the start this seemed large
  - Technology has improved in 5 years.