areaDetector: What's New?

Mark Rivers

GeoSoilEnviroCARS, Advanced Photon Source

University of Chicago

Outline

- Changes in the last year
  - ADCore R2-3, R2-4, R2-5 (coming soon)
- New and improved plugins and drivers
- Future plans

NDFileHDF5 (ADCore R2-5)

- Support for Single Writer Multiple Reader (SWMR) that is new in HDF5 1.10.0
- Allows applications to read an HDF5 file while it is open and being written by the areaDetector HDF5 plugin
- SWMR support must be explicitly enabled at run-time in the HDF5 plugin
- If SWMR support is not enabled then files written by the HDF5 plugin can be read by older versions of the HDF5 library
- If SWMR support is enabled then the HDF5 files can only be read using 1.10.0 or later of the HDF5 library
- From Alan Greer, Observatory Sciences and Diamond Light Source

NDPluginPva and pvAccess Driver (ADCore R2-5)

- New plugin that converts NDArrays into the EPICSv4 normative type NTNDArray
- An embedded EPICSv4 server serves the new NTNDArray structure as an EPICSv4 PV
- Can be received by any EPICSv4 client
- New pvAccess driver receives NTNDArrays over the network, converts to NDArrays and calls plugins
  - Can be used to run areaDetector IOC and plugins on another machine or in another process
- ImageJ plugin could be converted to receive NTNDArrays rather than waveform record and scaler records for xsize, ysize, colormode, etc.
- From Bruno Martins at Brookhaven

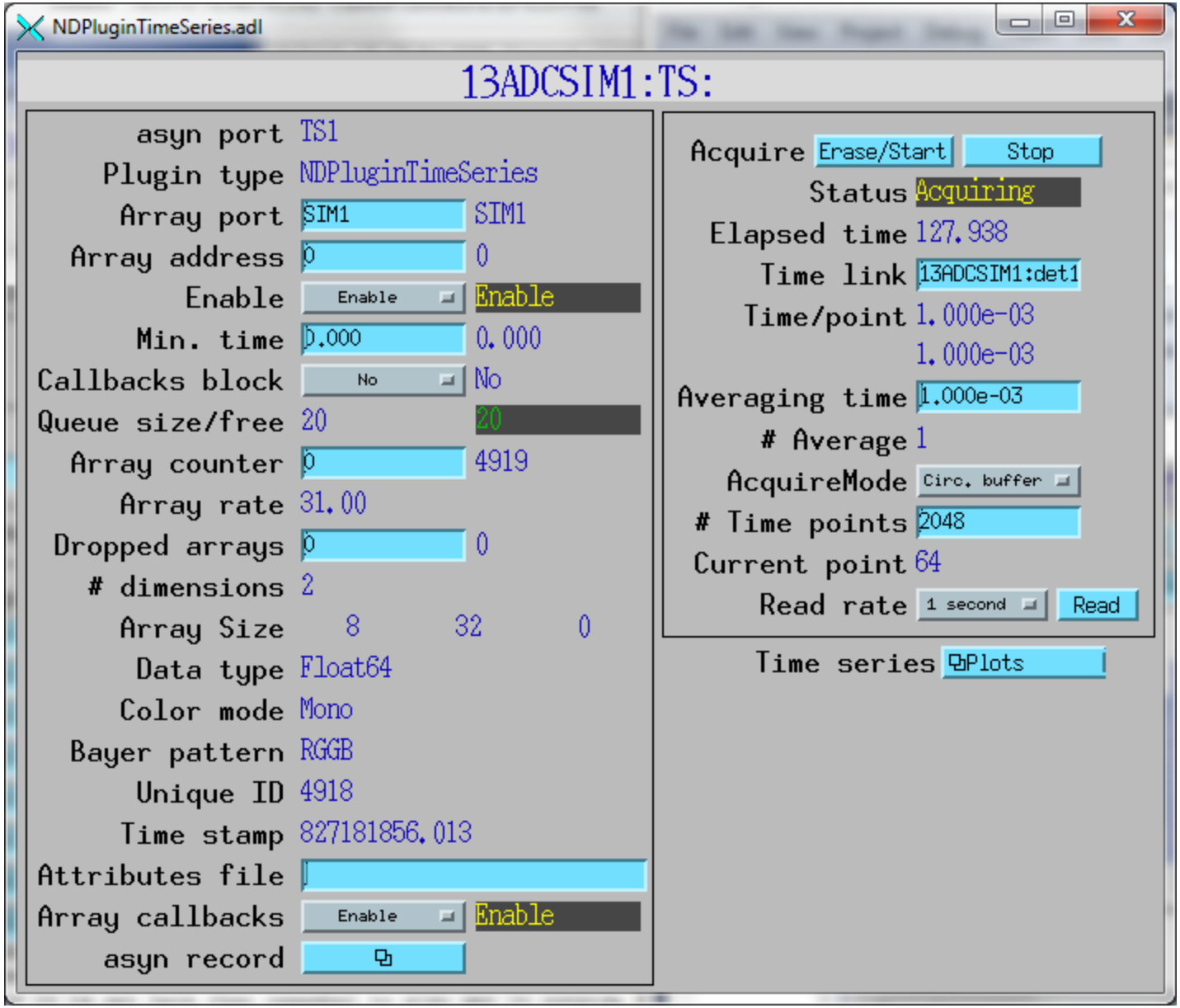
NDPluginTimeSeries (ADCore R2-5)

- New plugin intended for use with devices that produce time-series data for multiple signals.
  - Examples: quadEM current meters, ADCs, etc.
- Designed to replace drvFastSweep from EPICS mca module
  - quadEM (done), ip330 (future)
- Accepts NDArrays of dimension [NumSignals] or [NumSignals, NewTimePoints]
- Creates NumSignals NDArrays of dimensions [NumTimePoints], and one NDArray of dimensions [NumTimePoints, NumSignals]
- Exports waveform records of the time-series data
- Exports waveform record of the time axis
  - Waveform records are useful for plotting in medm, caQtDM, etc.

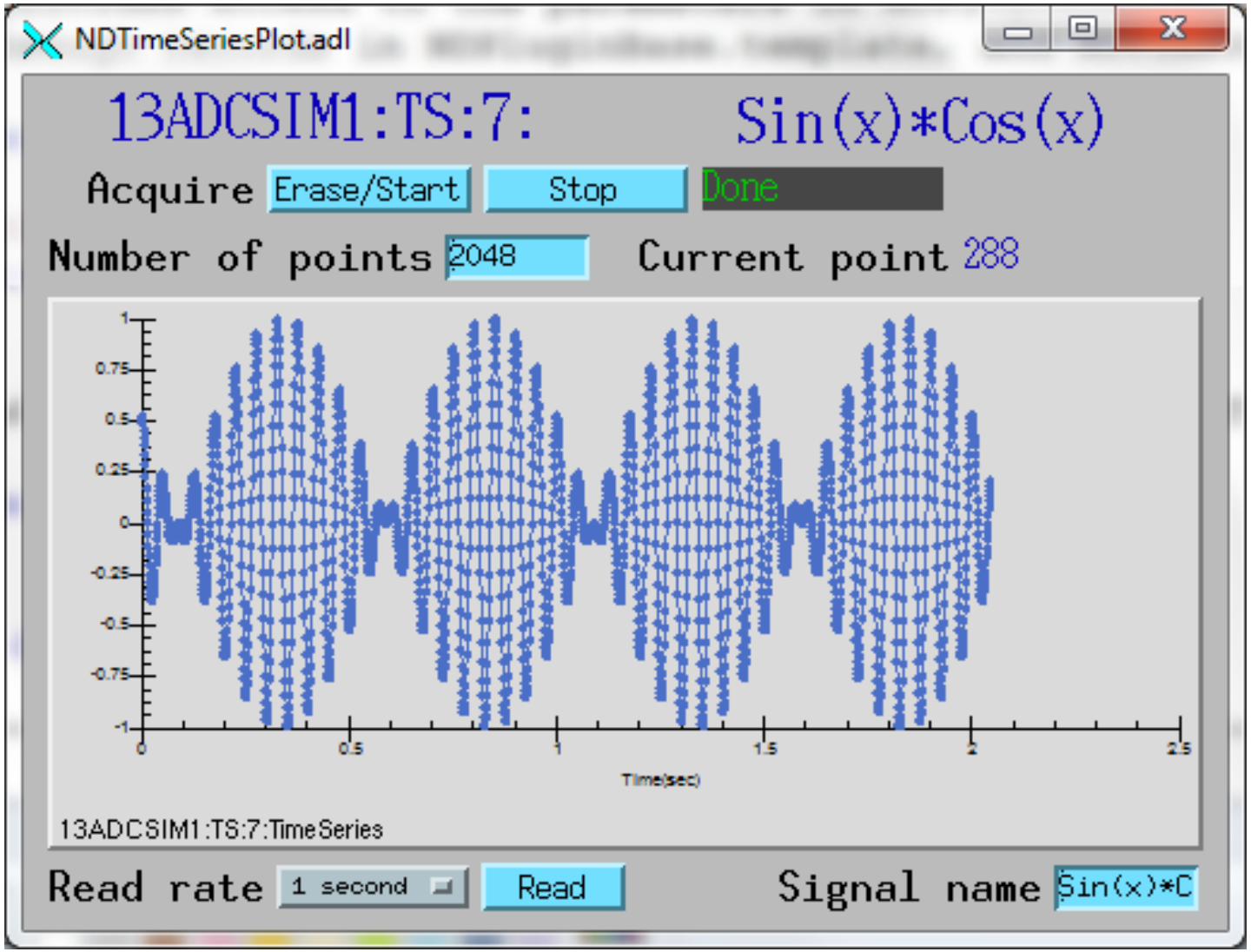
NDPluginTimeSeries

- Number of time points to collect, TSNumPoints, can be changed at run-time
- Does optional averaging of time points, specified with TSAveragingTime PV
- Operates in 2 modes
  - Fixed length: Collect TSNumPoints and stop
  - Circular buffer: Collect continuously, exporting the most recent TSNumPoints values
- Input link that gets the actual time between points from the driver doing the callbacks.
  - Can be manually specified if not available from the driver

NDPluginTimeSeries



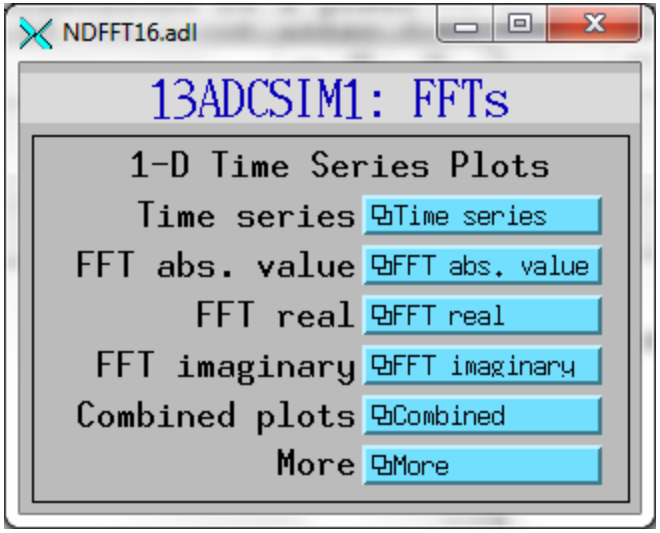
NDPluginTimeSeries

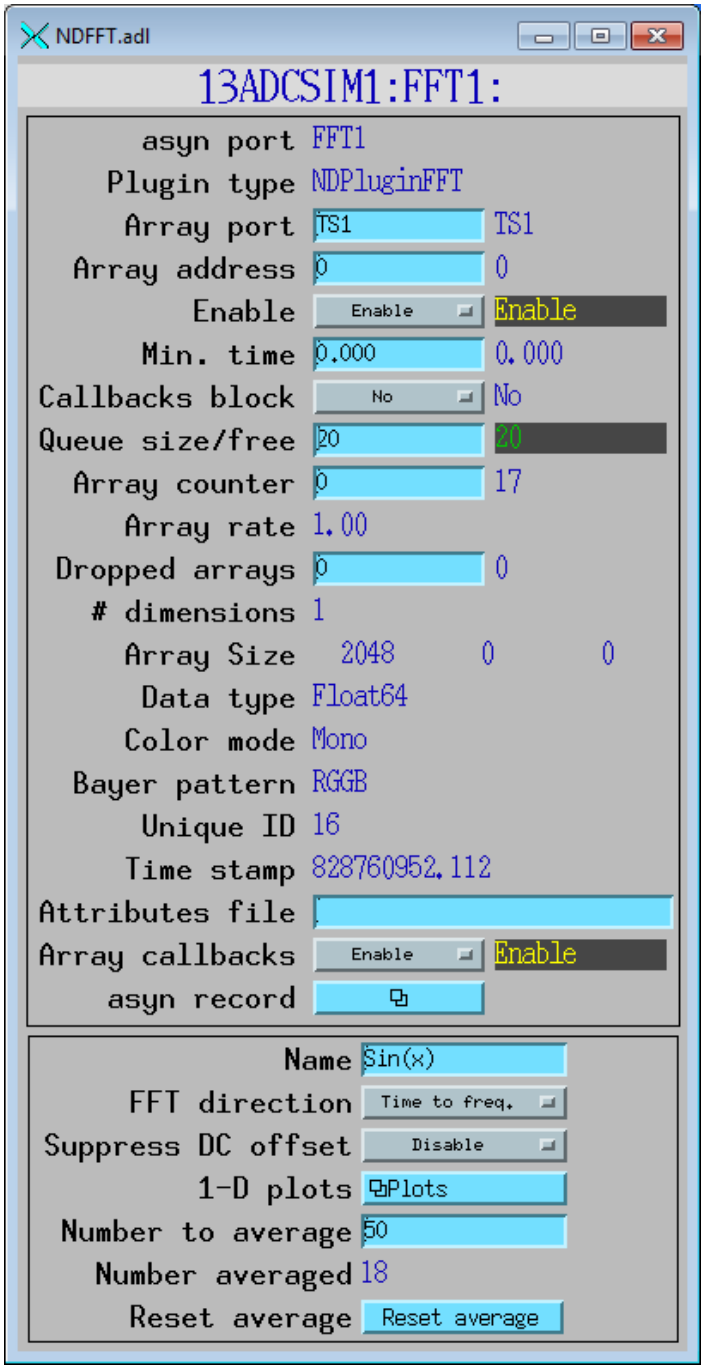


NDPluginFFT (ADCore R2-5)

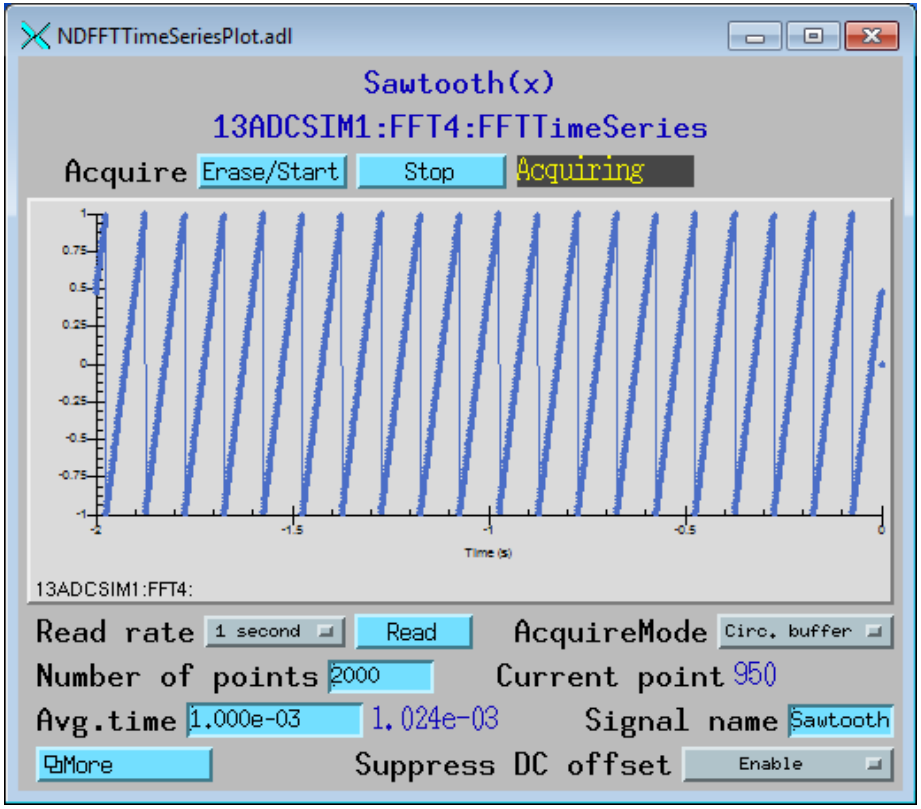
- New plugin to compute 1-D and 2-D Fast Fourier Transforms
- Exports 1-D or 2-D NDArrays containing the absolute value of the FFT (epicsFloat64)
- Exports 1-D waveform records containing the input, and the real, imaginary and absolute values of the first row of the FFT
- Exports 1-D waveform records containing the time and frequency axis values
  - 1-D waveform records are useful for plotting in medm, caQtDM, etc.
- Optionally does recursive averaging of the FFT to improve signal/noise
- FFT algorithm used requires input dimensions to be a power of 2
  - Plugin will pad input to next larger power of 2 if necessary

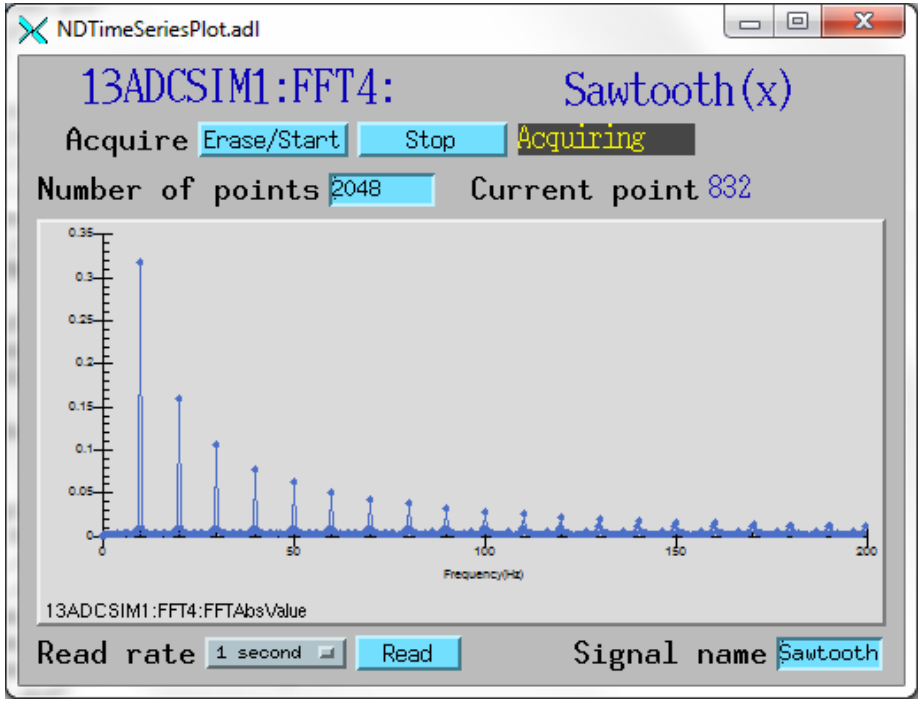
NDPluginFFT



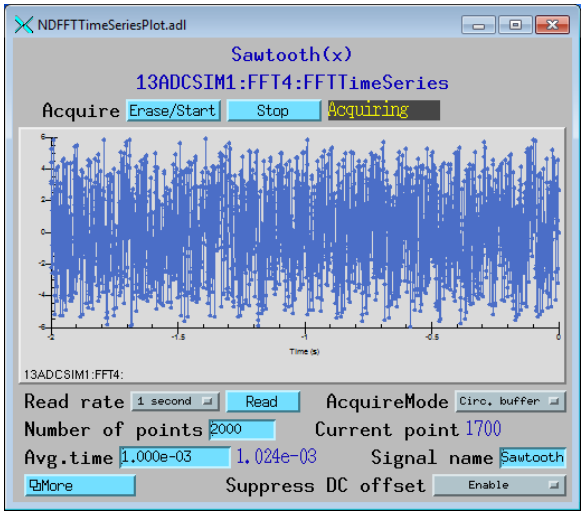


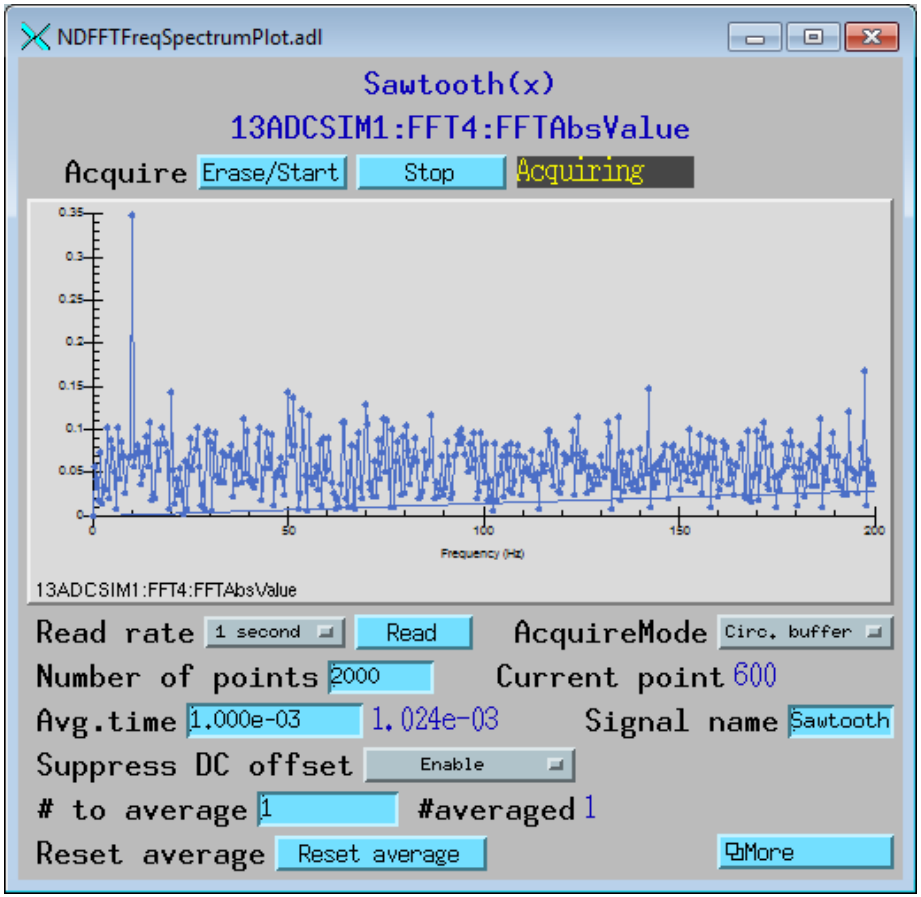
NDPluginFFT

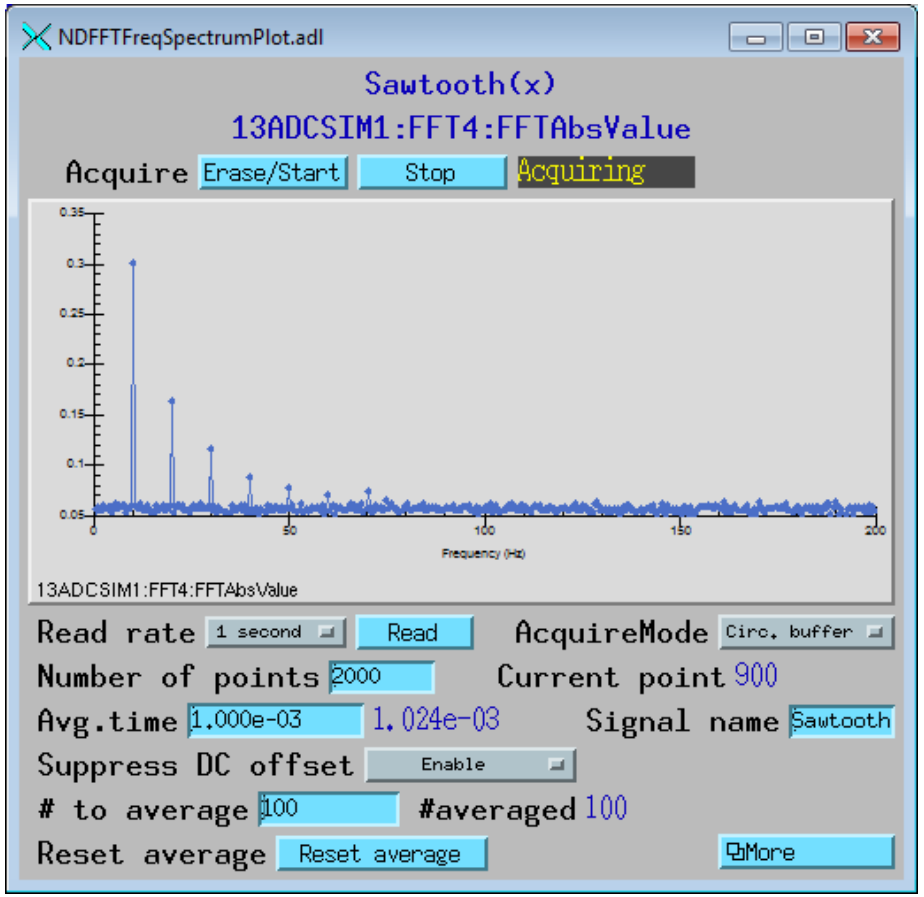




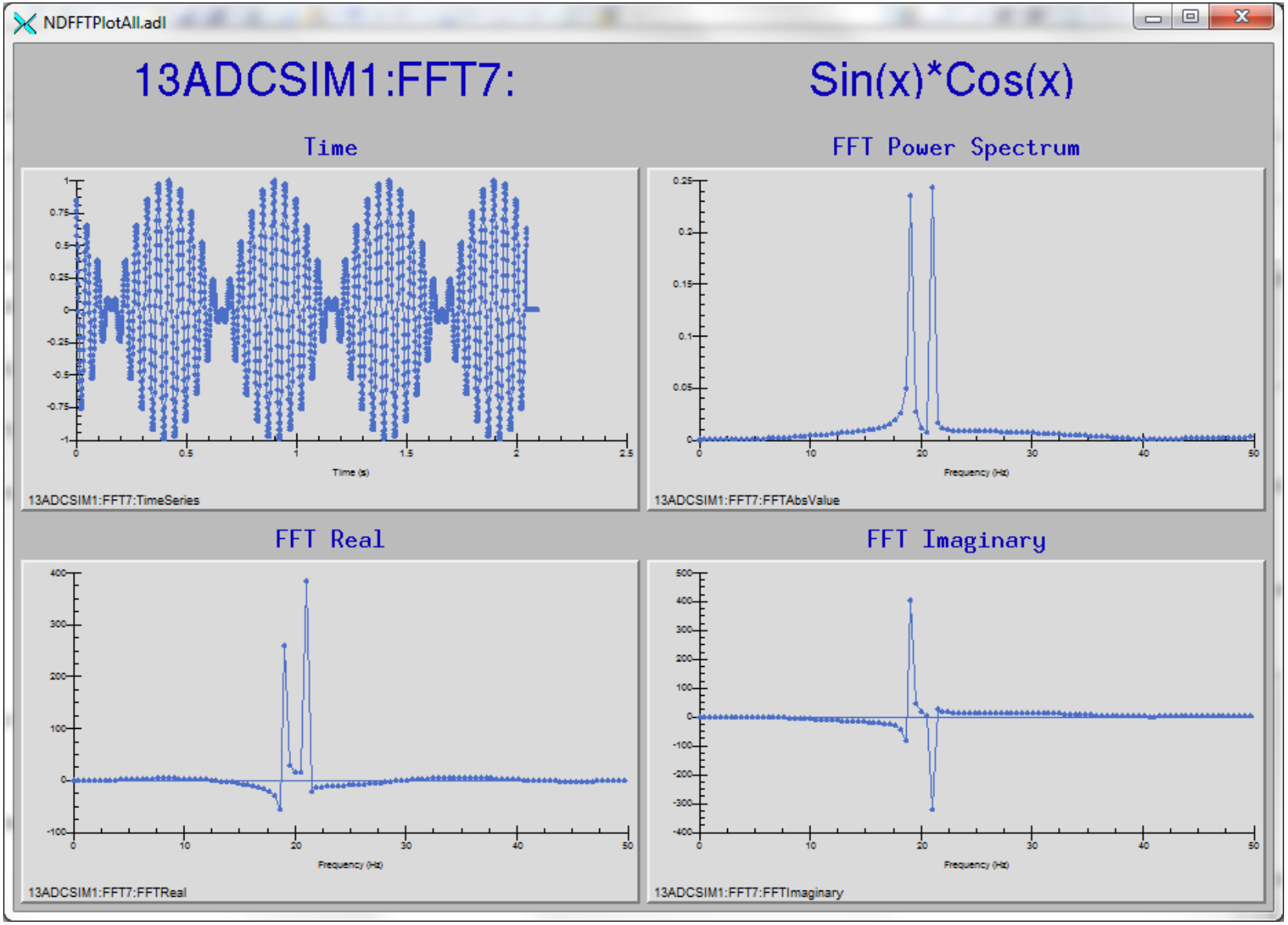
NDPluginFFT
Noisy signal
FFT averaging







NDPluginFFT
Combined plot

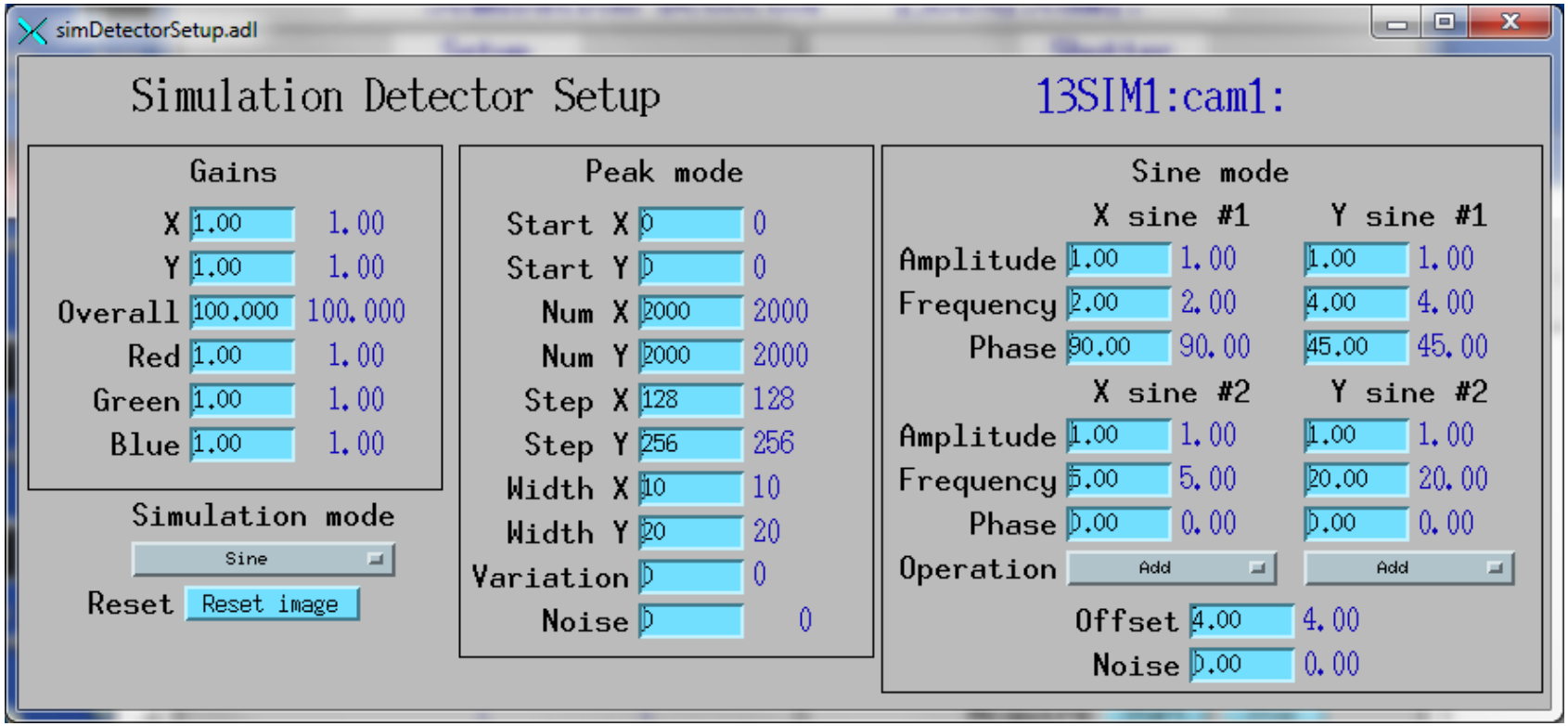


Perkin Elmer Driver (R2-1, R2-2, R2-3)

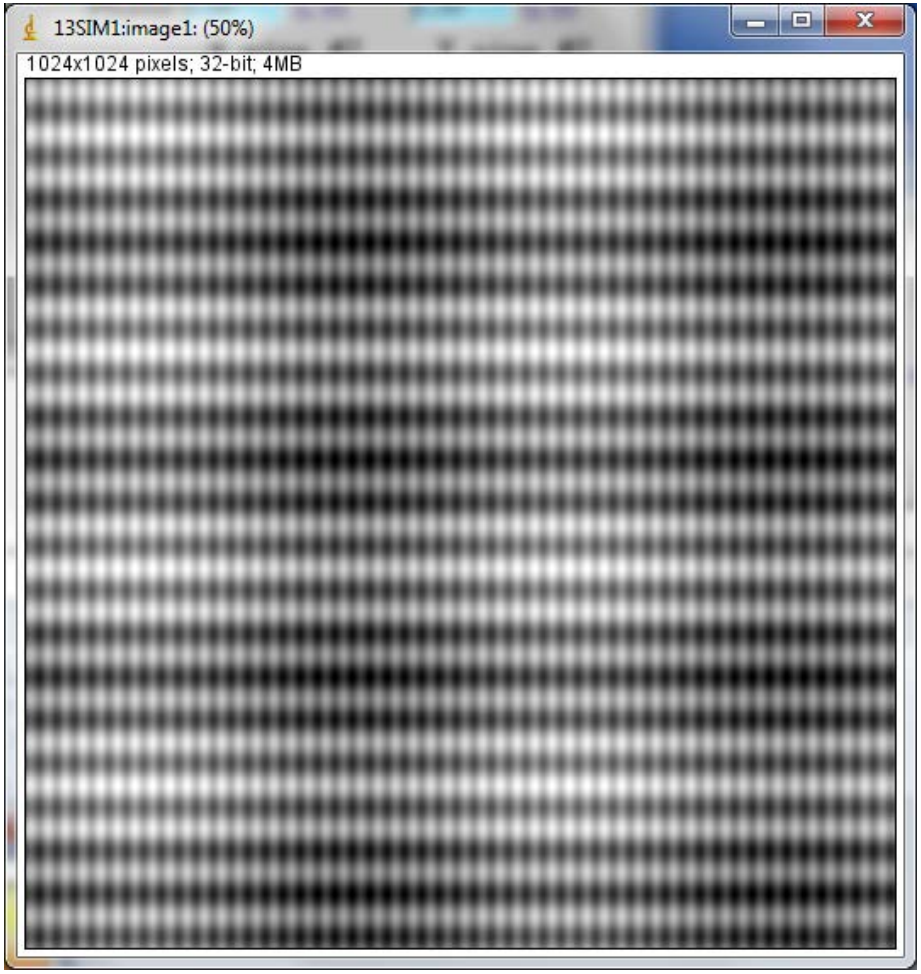
- Added support for Data Delivered on Demand (DDD) mode
  - Required for the XRPad model used at 12-ID-B
  - Fixes problems with ImageMode=Single, TriggerMode=Internal. Previously the frame delivered had actually started acquiring before Acquire was set to 1.
  - Fixed problems with TriggerMode=External. Previously the actual exposure time was the time between triggers, which might not be the same as used for the offset images. Using DDD mode uses AcquireTime on each external trigger, same as offset images. Does reduce rate from 15Hz to 5Hz.
- Added PEOffsetConstant, a constant offset to the data.
  - Prevents negative values which are clipped to 0 for low signal levels when subtracting offets.

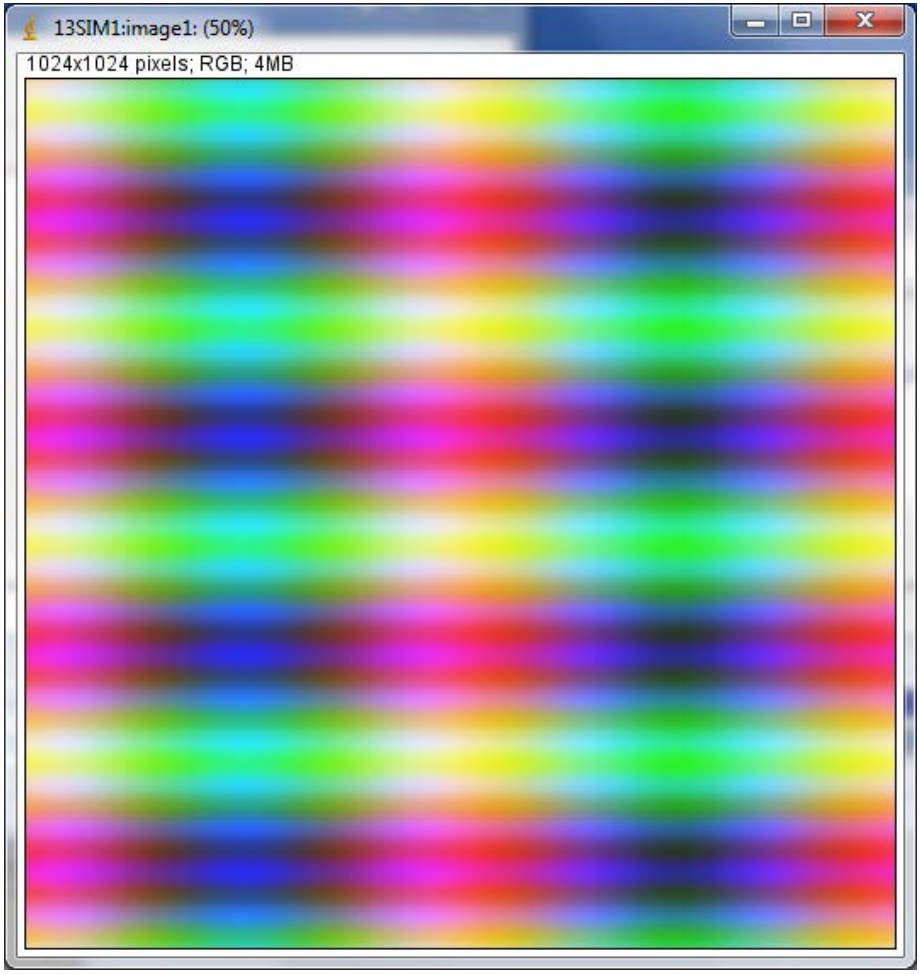
simDetector Driver (ADExample R2-2 soon)

- Moved from ADCore to ADExample
  - ADCore now does not build an IOC, so it is independent of synApps (autosave, calc, sscan, busy, devIocStats, etc.)
- Added new Sine simulation mode, in addition to existing LinearRamp and Peaks

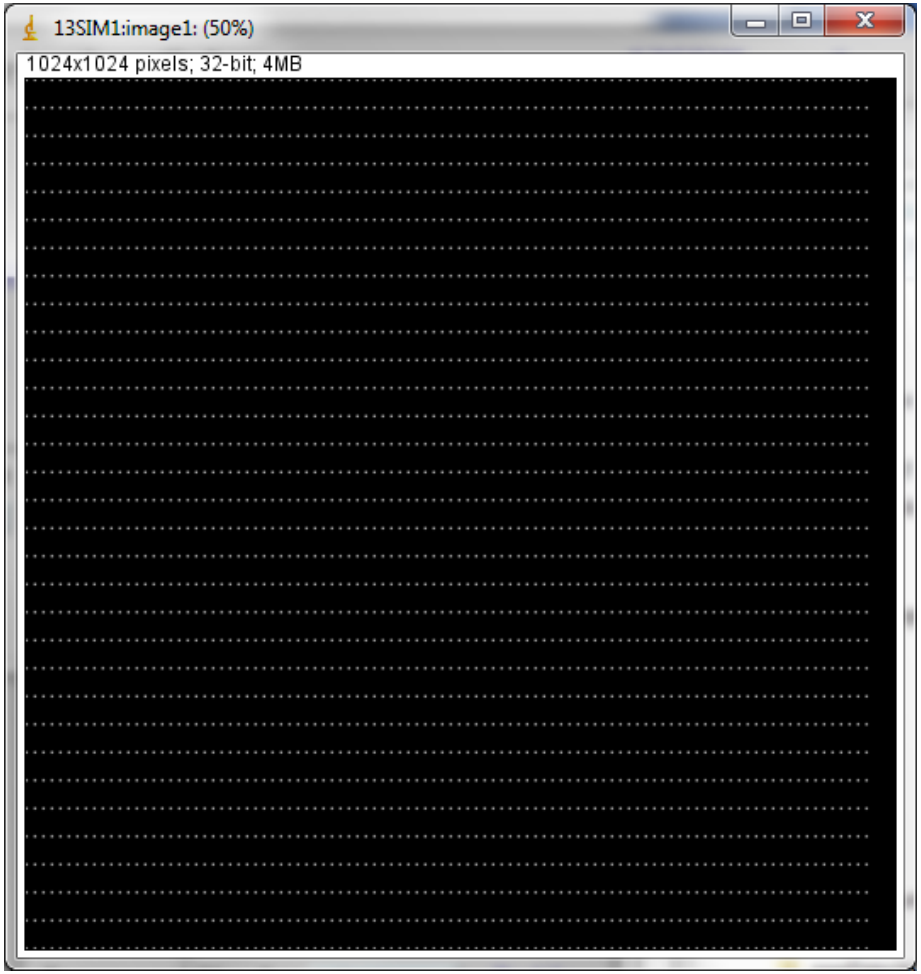


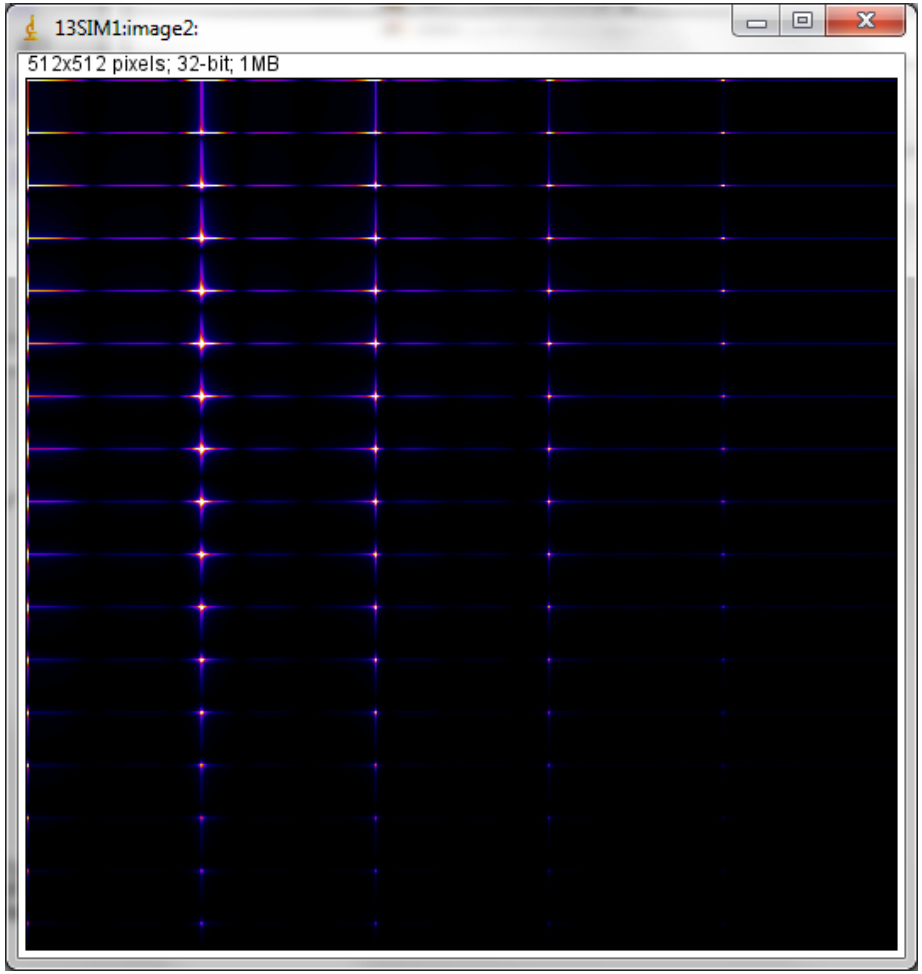
simDetector Sine mode





simDetector Peaks mode with FFT



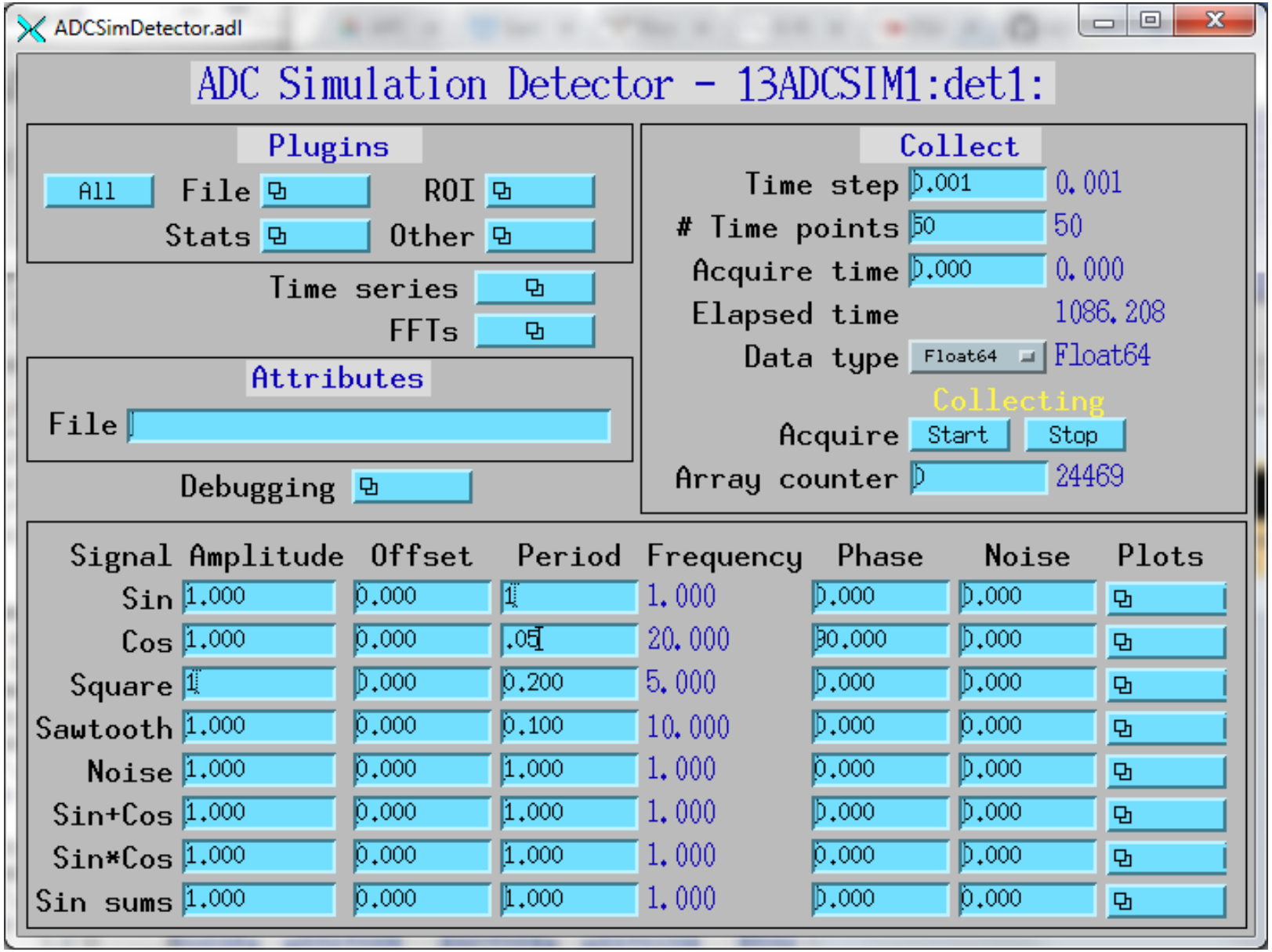


ADCSimDetector Driver (ADExample R2-2)

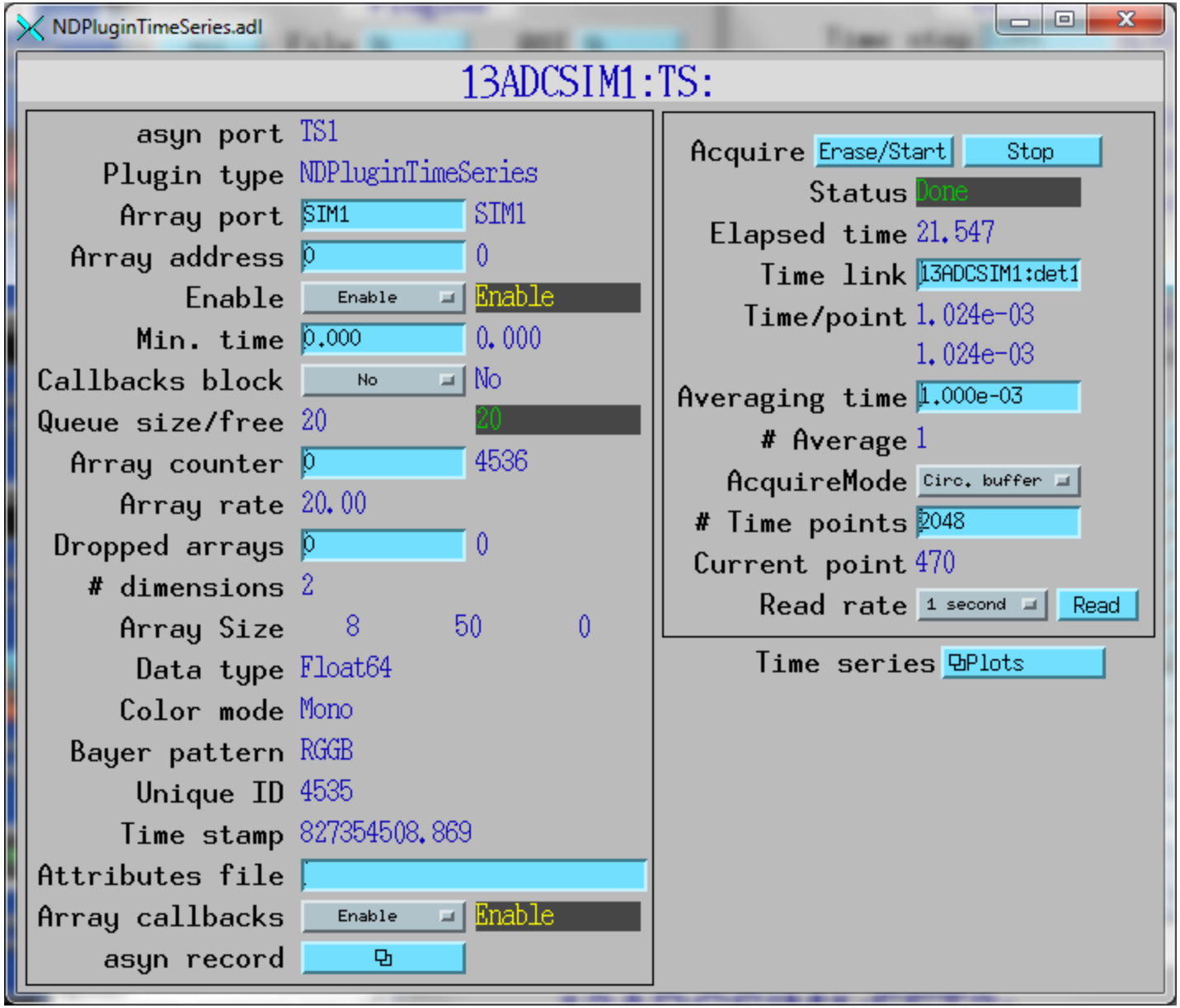
- New driver to simulate a buffered ADC
- Designed for testing NDPluginTimeSeries and NDPluginFFT
- Generates time-series data for 8 signals as 2-D NDArray[8, NumTimePoints]
  1. Sine
  2. cosine,
  3. square wave,
  4. sawtooth,
  5. noise,
  6. sin+cos,
  7. sin*cos
  8. sum of 4 sine waves
- Sine wave equation, for example

```
Signal[t] = Offset + Noise * random() + Amplitude *
            sin((time * Frequency + Phase/360.) * 2. * PI);
```

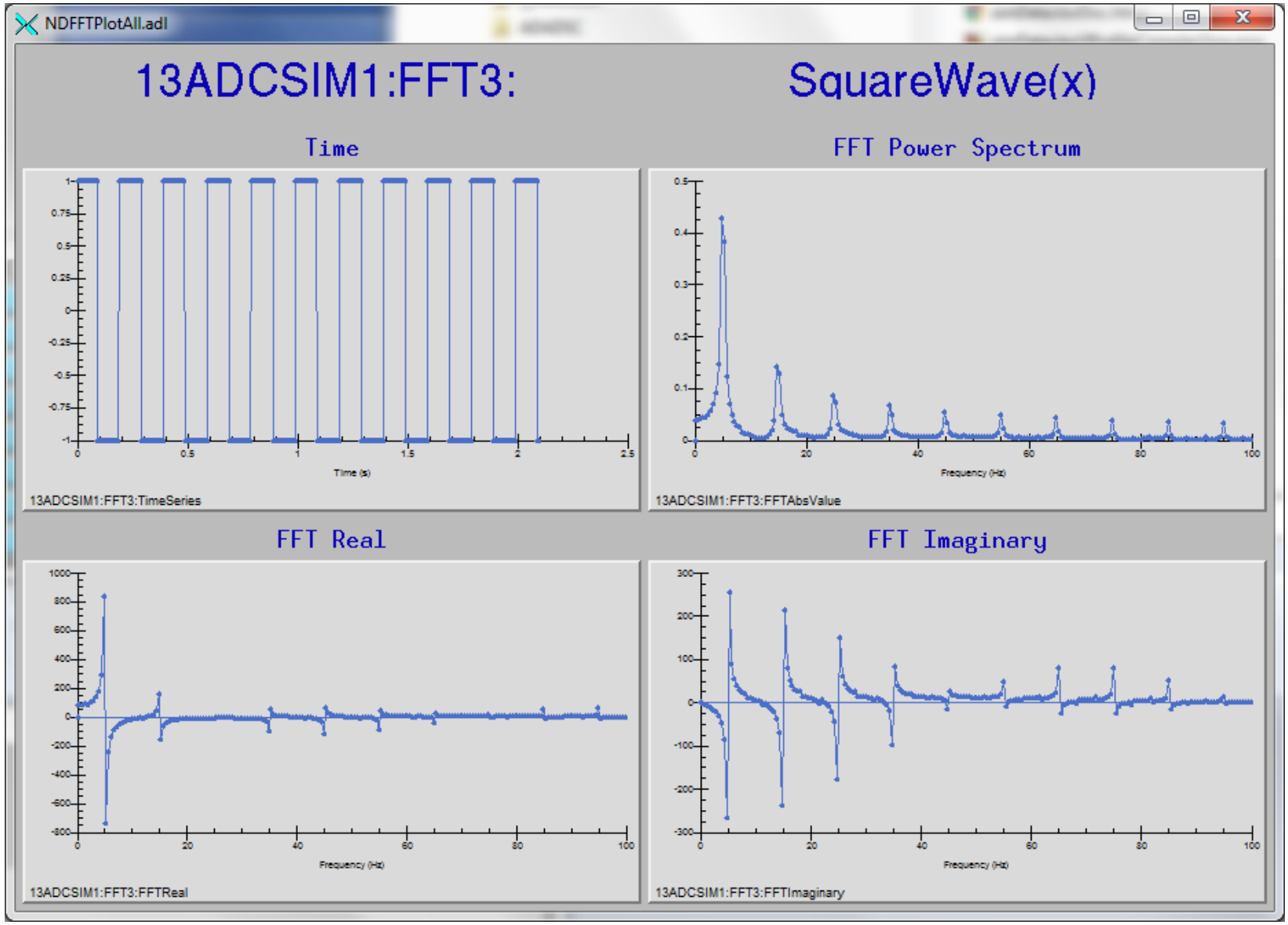
ADCSimDetector Driver



ADCSimDetector Driver



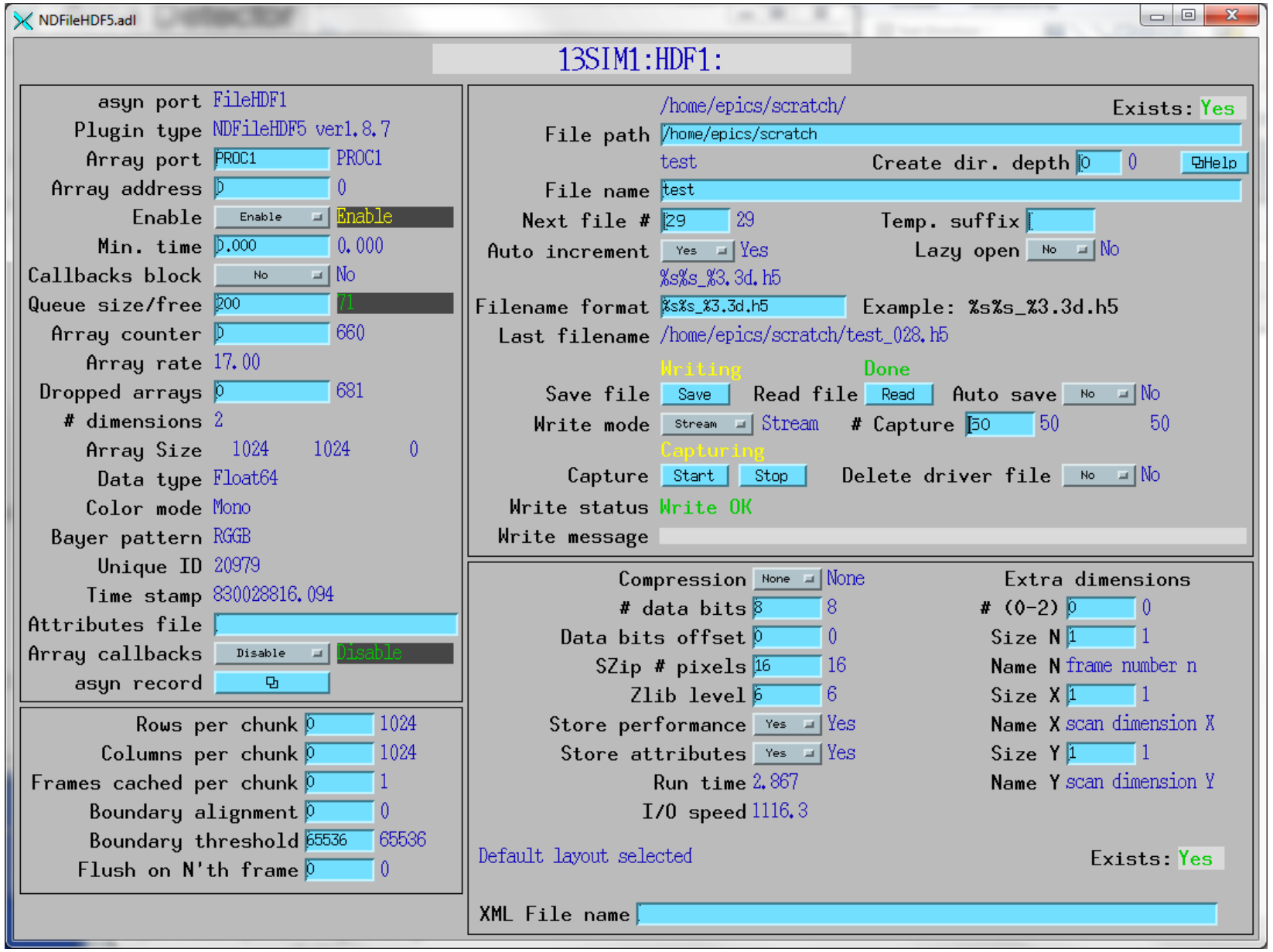
ADCSimDetector Driver



NDPluginDriver: Changing Queue Size

- NDPluginBase can change queue size at run-time
  - QueueSize PV is now longout, not longin
- Useful for file plugins where an acquisition of N frames is overflowing the queue, but increasing the queue can fix the problem.
  - Will be even more useful in ADCore R3-0 where we plan to eliminate Capture mode in NDPluginFile.
- Using Capture mode
  - NDArray memory is not allocated from the NDArrayPool, no check on allocating too many arrays or too much memory.
- Using queue size
  - NDArrays are allocated from the NDArrayPool, limits on total number of arrays and total memory defined in the constructor will be obeyed.
  - Important in preventing system freezes if the user accidentally tries allocate all the system memory, which can effectively crash the computer.

NDPluginDriver: Changing Queue Size



ADCore R3-0

- Simplify NDPluginFile base class and way file saving works
  - Remove the Single/Stream/Capture mode.
- Two parameters
  - # NDArrays to save (already present)
  - # NDArrays per file (new)
  - This allows saving only 1 array per HDF5 file, which is not possible now in Stream mode.
- Capture mode can be replaced:
  - Make input queue large enough OR
  - Use new NDPluginCircularBuffer
- Will require modifying clients that are doing file saving, hence a major release number