

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.adl

13ADCSIM1:TS:

asyn port	TS1
Plugin type	NDPluginTimeSeries
Array port	SIM1 SIM1
Array address	0 0
Enable	Enable Enable
Min. time	0.000 0.000
Callbacks block	No No
Queue size/free	20 20
Array counter	0 4919
Array rate	31.00
Dropped arrays	0 0
# dimensions	2
Array Size	8 32 0
Data type	Float64
Color mode	Mono
Bayer pattern	RGGB
Unique ID	4918
Time stamp	827181856.013
Attributes file	
Array callbacks	Enable Enable
asyn record	

Acquire Erase/Start Stop

Status Acquiring

Elapsed time 127.938

Time link 13ADCSIM1:det1

Time/point 1.000e-03
1.000e-03

Averaging time 1.000e-03

Average 1

AcquireMode Circ. buffer

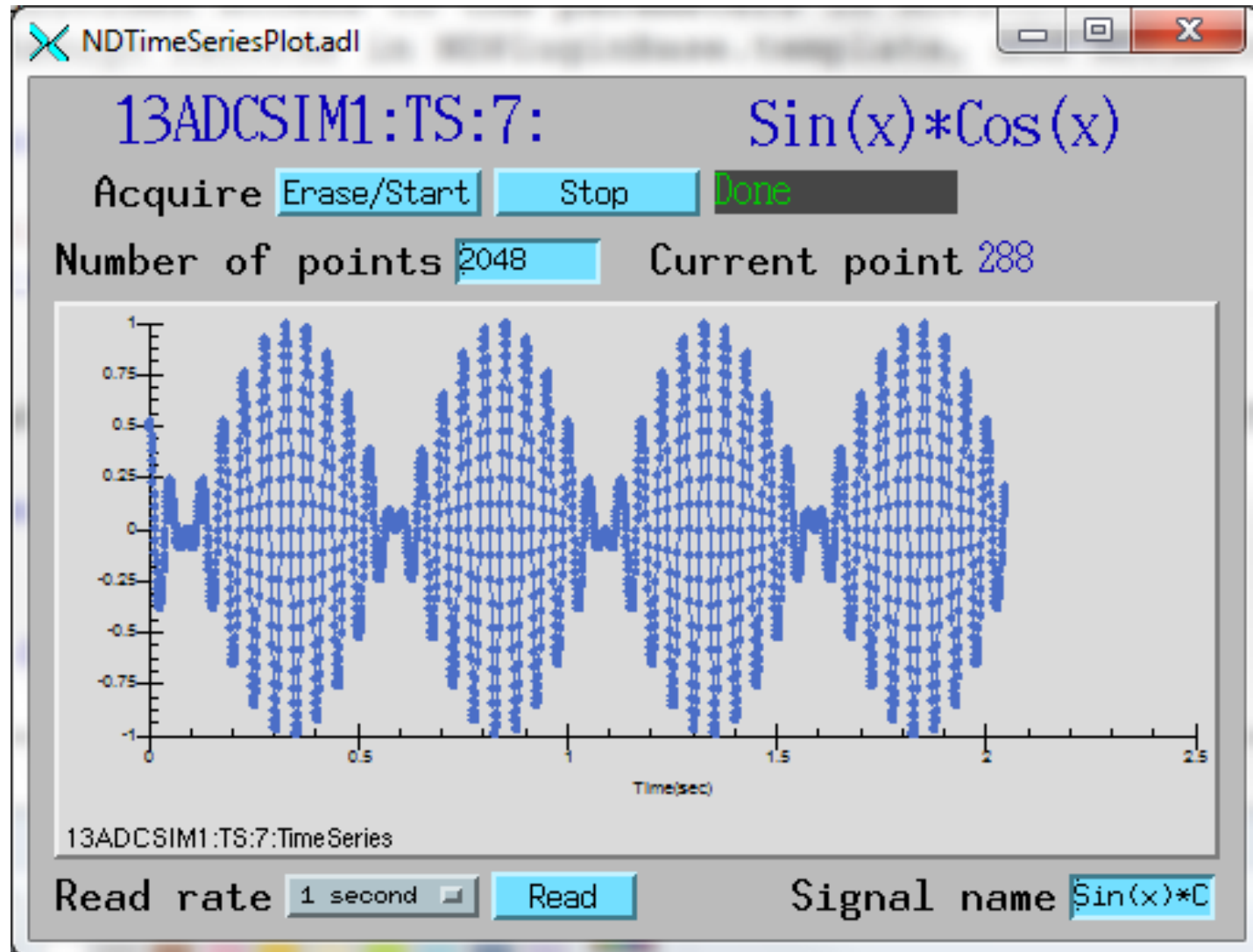
Time points 2048

Current point 64

Read rate 1 second Read

Time series Plots

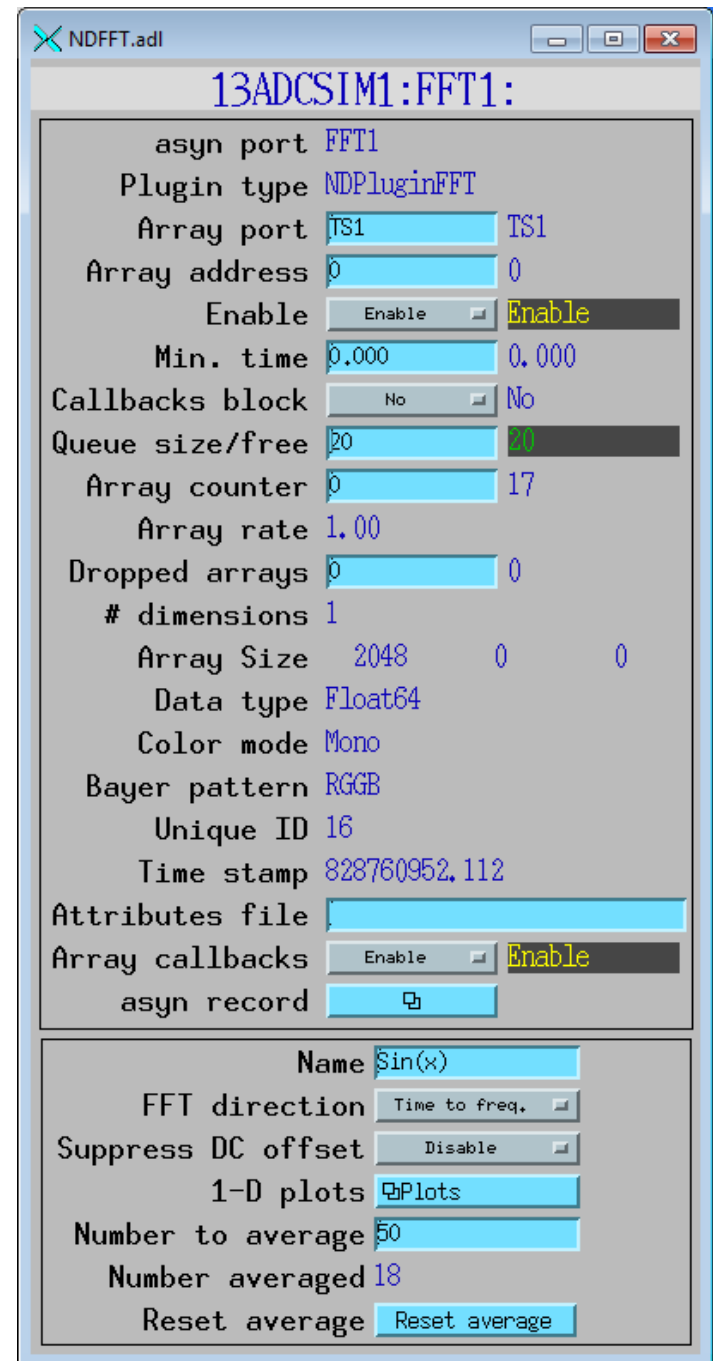
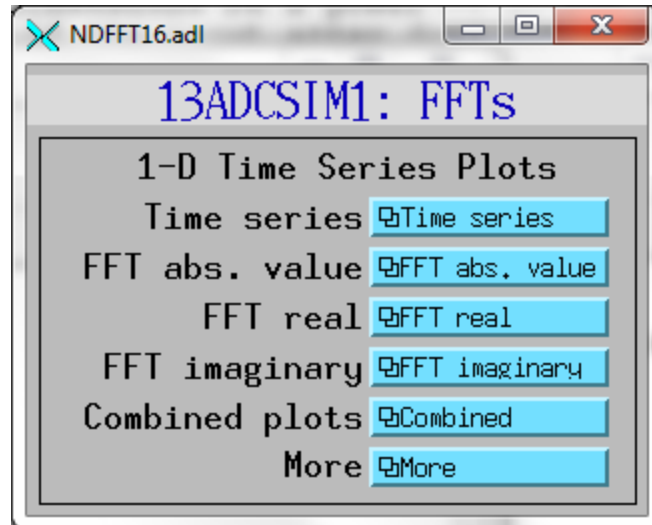
NDPluginTimeSeries



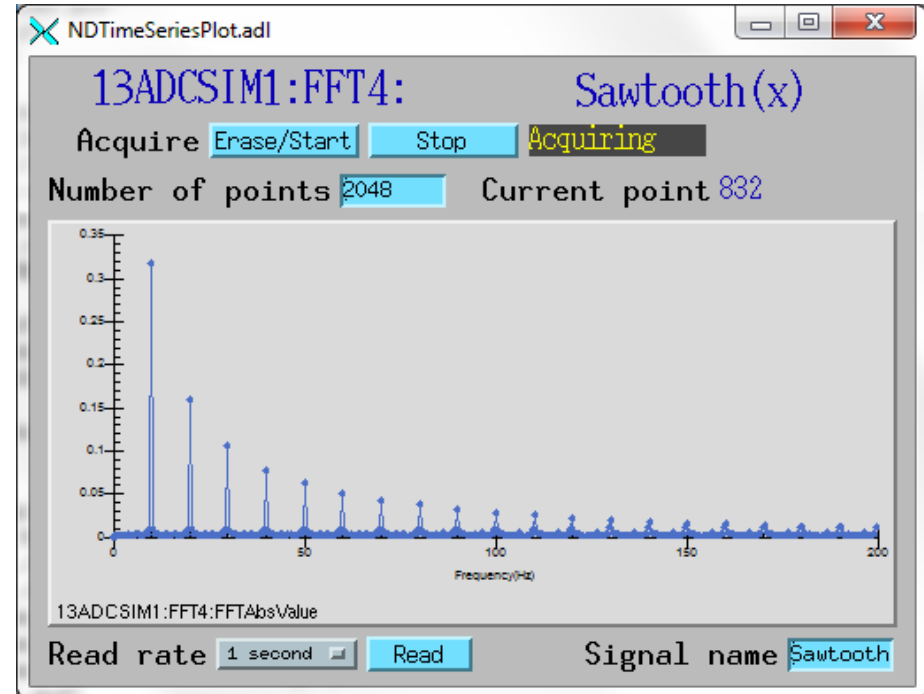
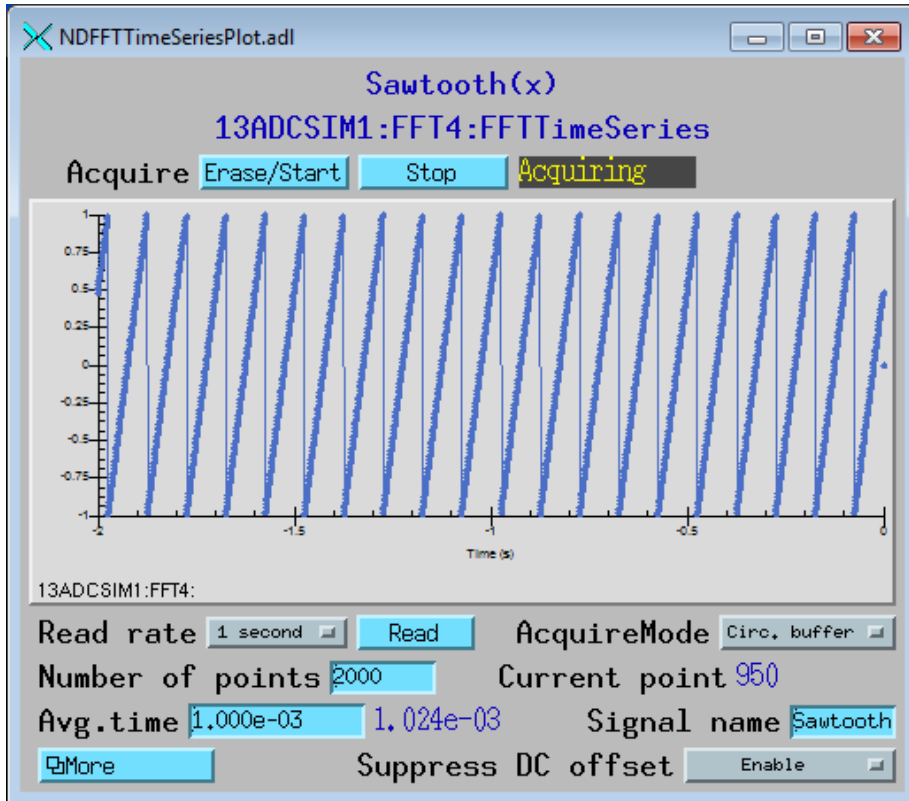
NDPluginFFT (ADCore R2-5)

- 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



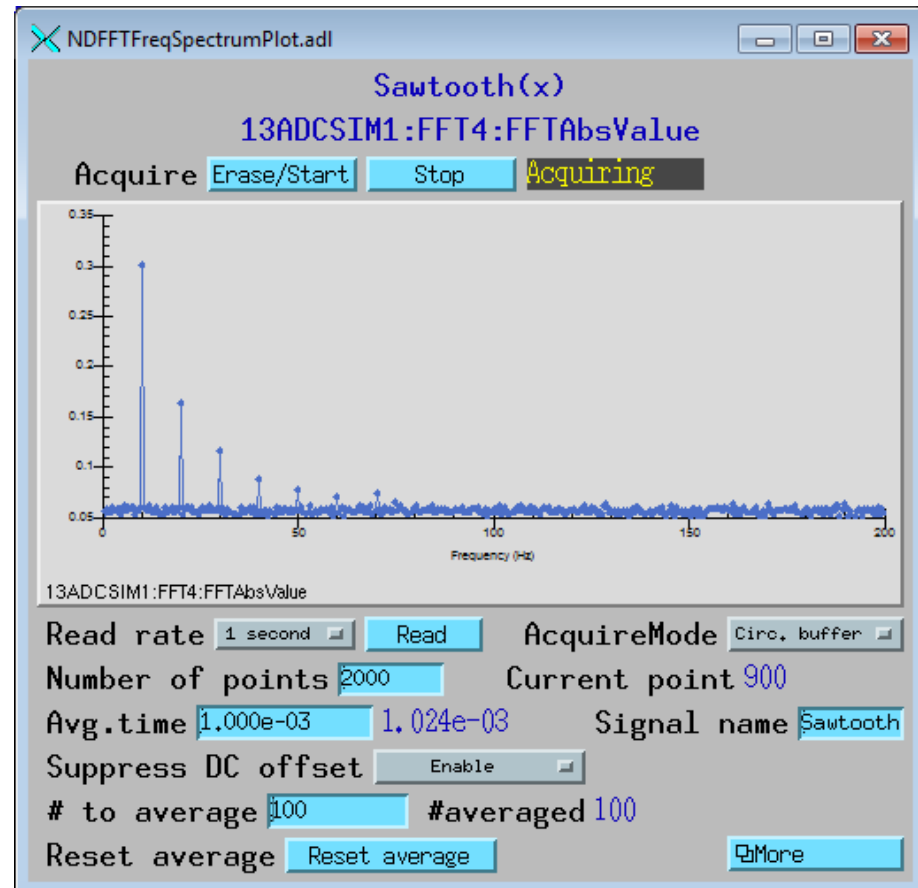
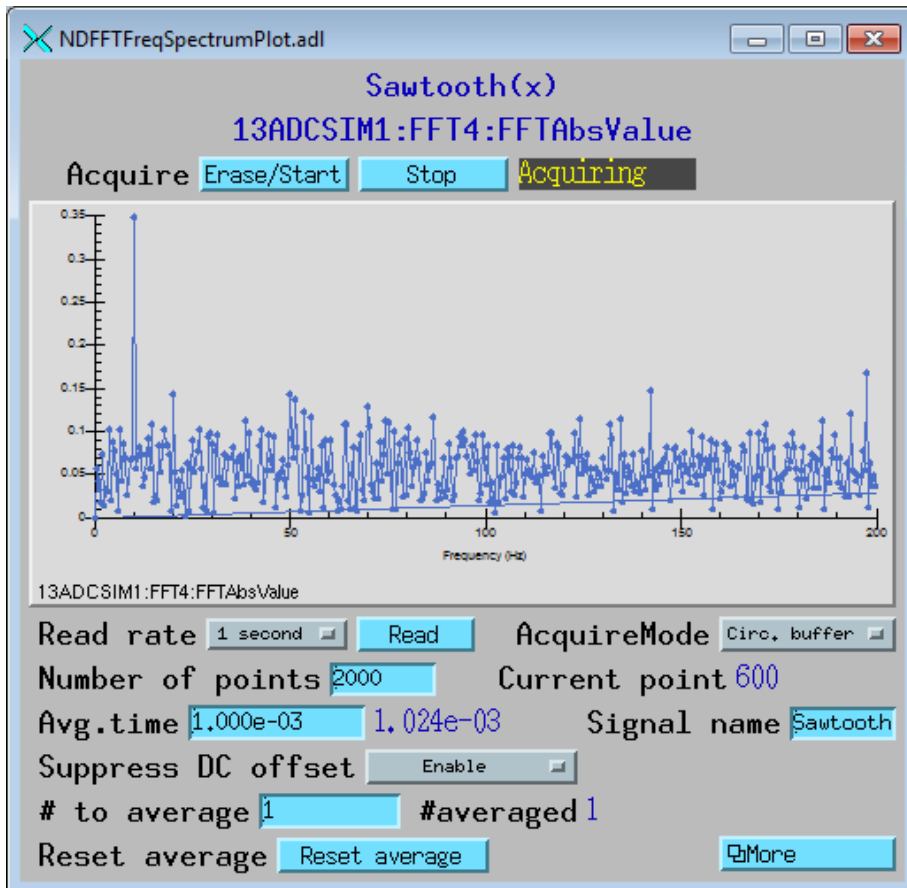
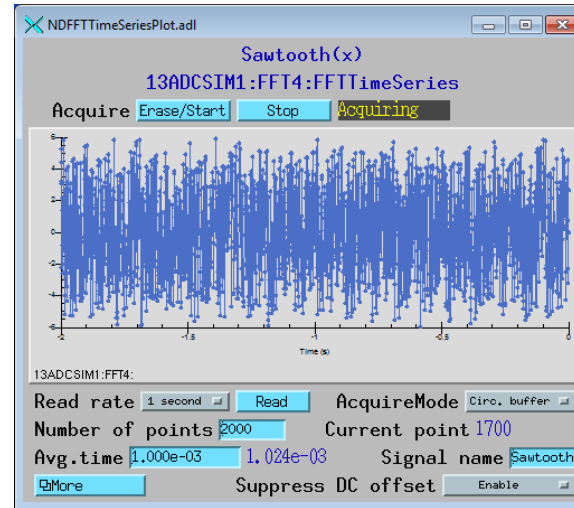
NDPluginFFT



NDPluginFFT

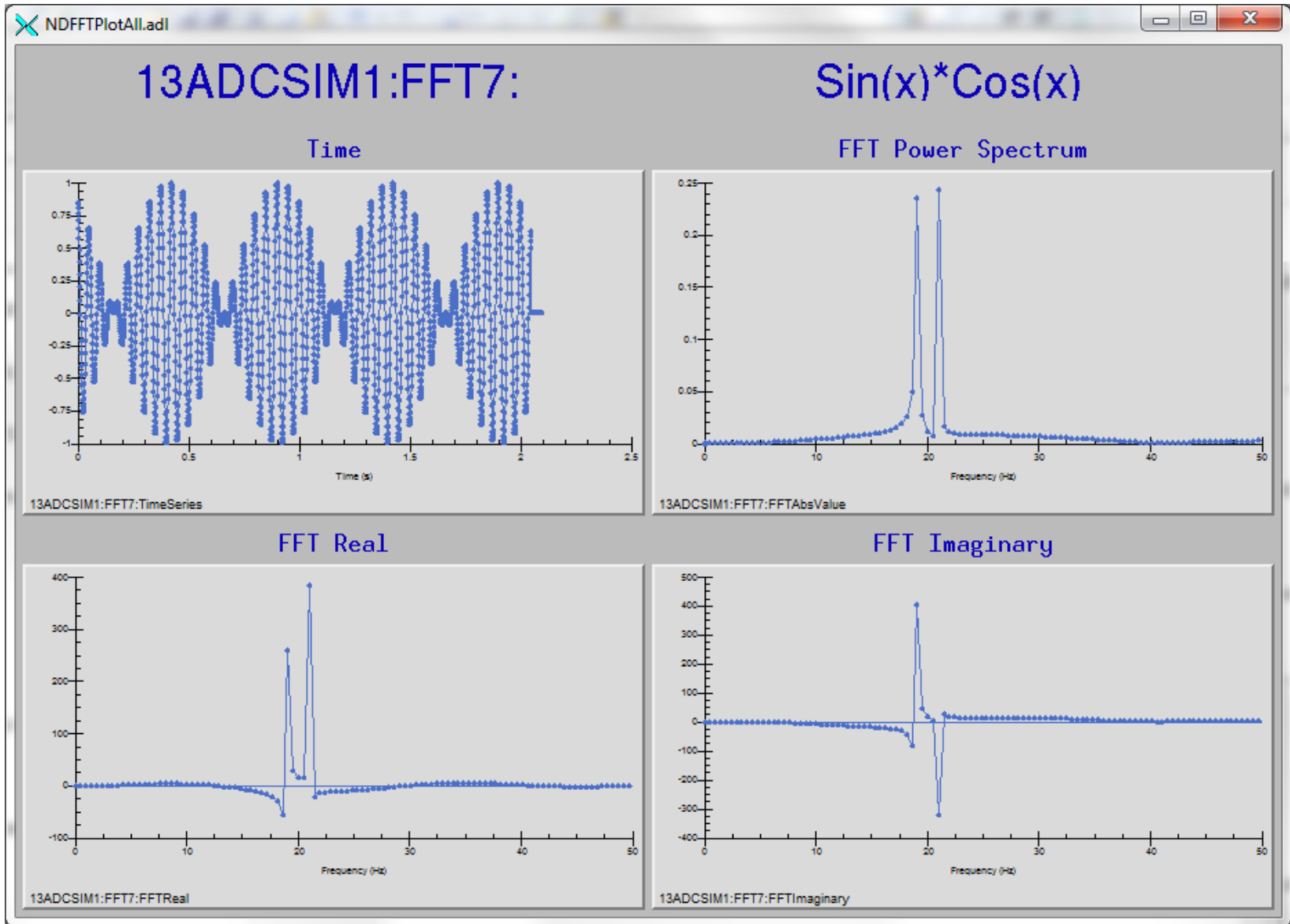
Noisy signal

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 offsets.

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

The screenshot shows the 'Simulation Detector Setup' window for '13SIM1:cam1'. The window is divided into three main sections: Gains, Peak mode, and Sine mode.

Gains

X	1.00	1.00
Y	1.00	1.00
Overall	100.000	100.000
Red	1.00	1.00
Green	1.00	1.00
Blue	1.00	1.00

Simulation mode

Sine

Reset

Peak mode

Start X	0	0
Start Y	0	0
Num X	2000	2000
Num Y	2000	2000
Step X	128	128
Step Y	256	256
Width X	10	10
Width Y	20	20
Variation	0	0
Noise	0	0

Sine mode

	X sine #1	Y sine #1
Amplitude	1.00	1.00
Frequency	2.00	4.00
Phase	90.00	45.00

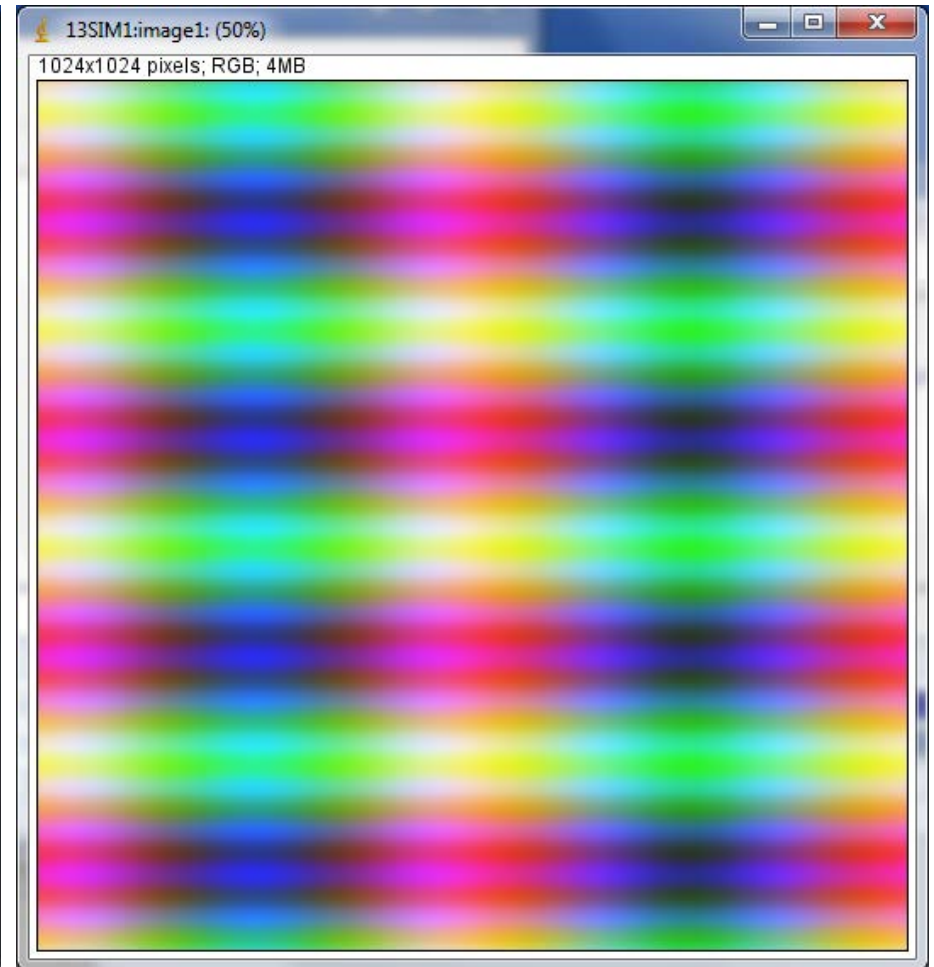
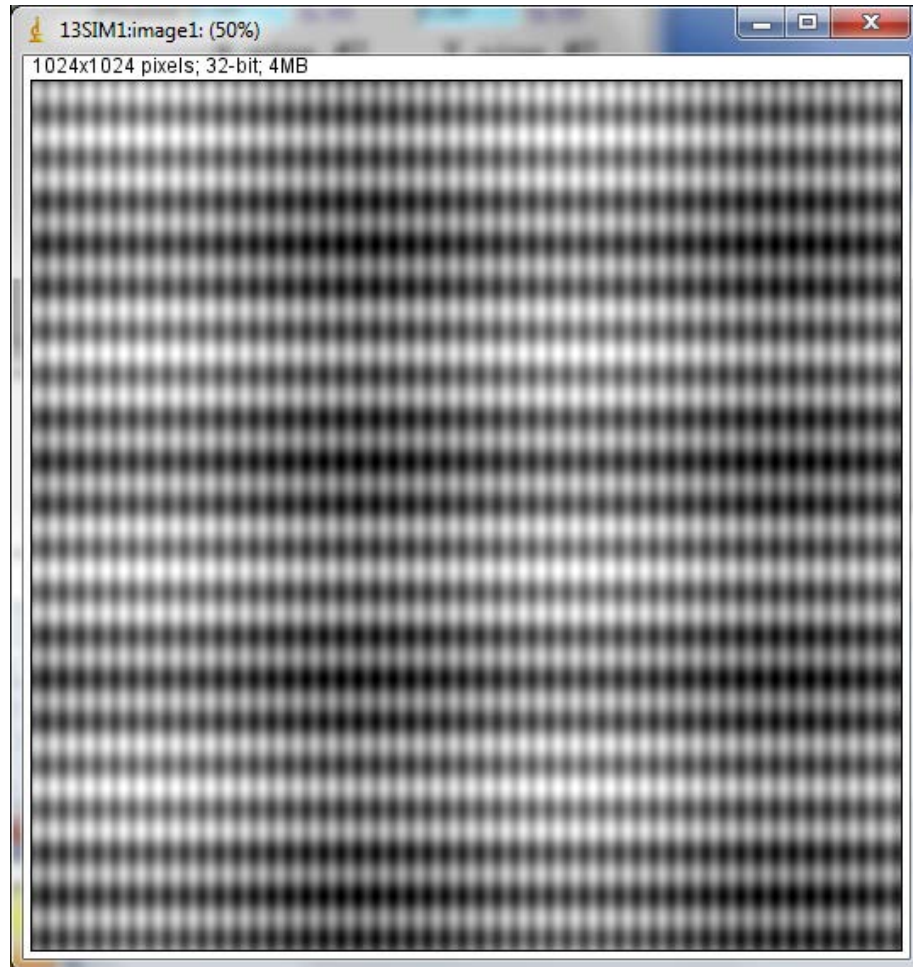
	X sine #2	Y sine #2
Amplitude	1.00	1.00
Frequency	5.00	20.00
Phase	0.00	0.00

Operation

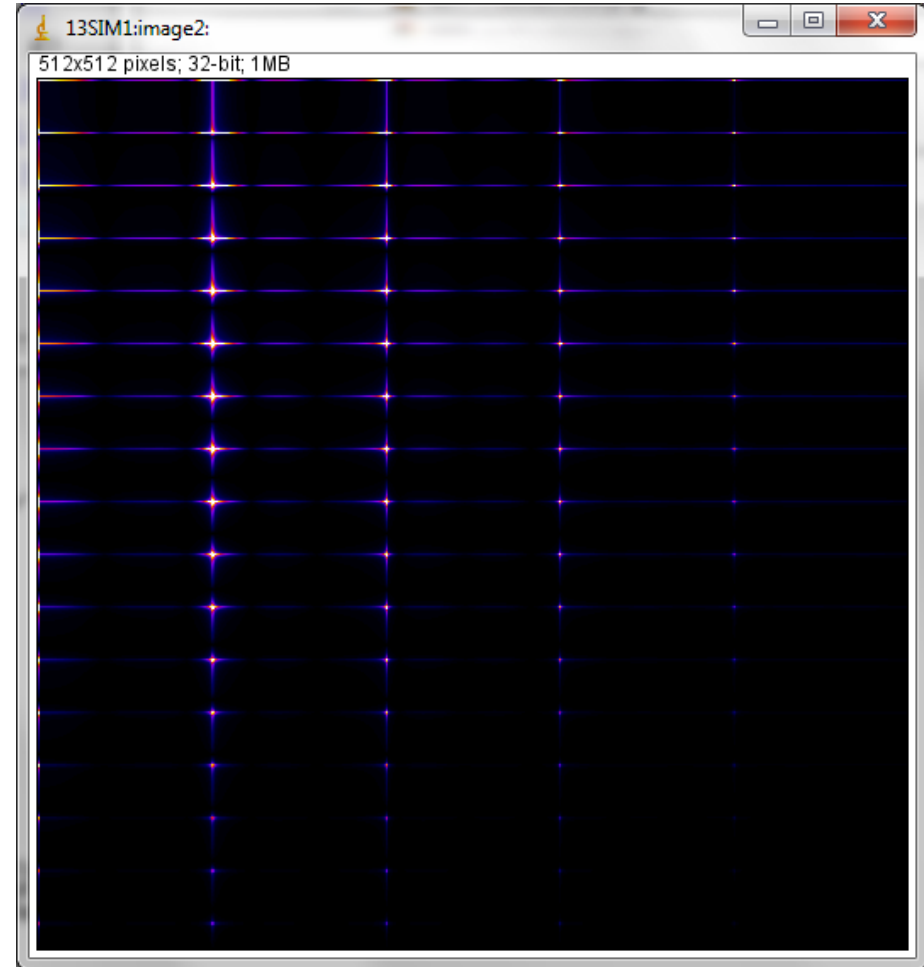
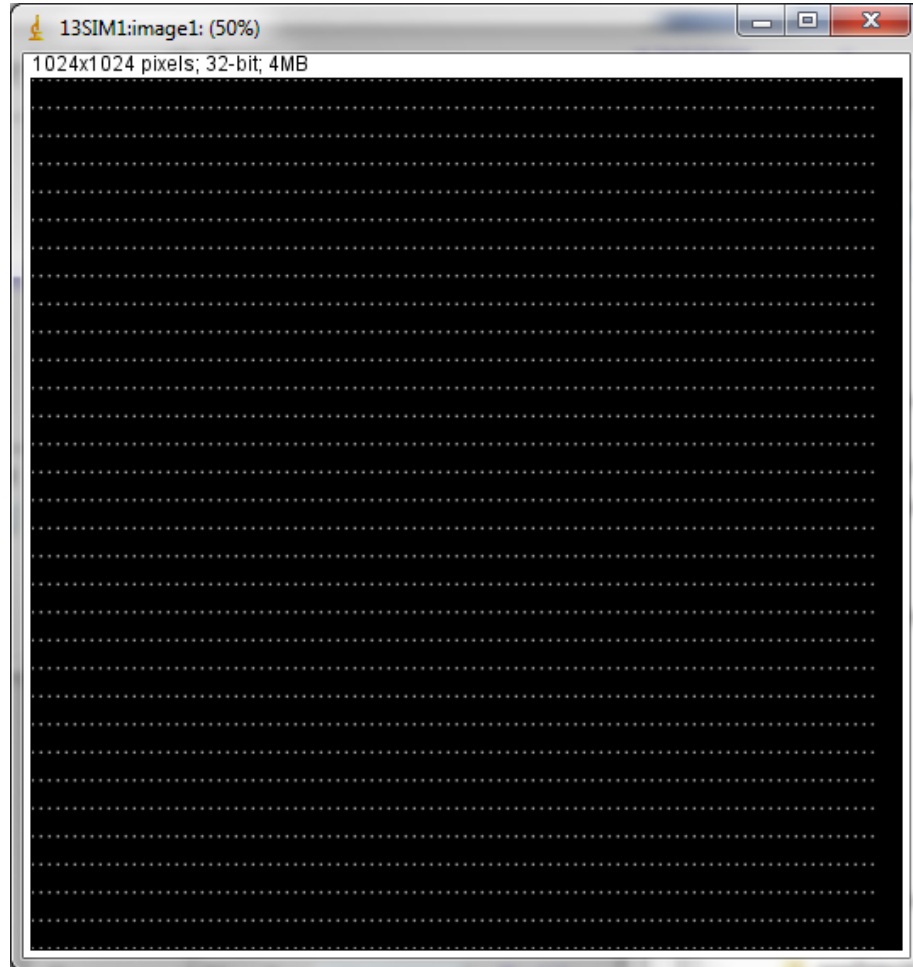
Offset 4.00 4.00

Noise 0.00 0.00

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 NDAarray[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.adl

ADC Simulation Detector - 13ADCSIM1:det1:

Plugins

All File ROI
Stats Other

Time series
FFTs

Attributes

File

Debugging

Collect

Time step 0.001
Time points 50
Acquire time 0.000
Elapsed time 1086.208
Data type Float64

Collecting

Acquire

Array counter 24469

	Signal	Amplitude	Offset	Period	Frequency	Phase	Noise	Plots
	Sin	<input type="text" value="1.000"/>	<input type="text" value="0.000"/>	<input type="text" value="1.000"/>	1.000	<input type="text" value="0.000"/>	<input type="text" value="0.000"/>	<input type="checkbox"/>
	Cos	<input type="text" value="1.000"/>	<input type="text" value="0.000"/>	<input type="text" value="0.050"/>	20.000	<input type="text" value="0.000"/>	<input type="text" value="0.000"/>	<input type="checkbox"/>
	Square	<input type="text" value="1.000"/>	<input type="text" value="0.000"/>	<input type="text" value="0.200"/>	5.000	<input type="text" value="0.000"/>	<input type="text" value="0.000"/>	<input type="checkbox"/>
	Sawtooth	<input type="text" value="1.000"/>	<input type="text" value="0.000"/>	<input type="text" value="0.100"/>	10.000	<input type="text" value="0.000"/>	<input type="text" value="0.000"/>	<input type="checkbox"/>
	Noise	<input type="text" value="1.000"/>	<input type="text" value="0.000"/>	<input type="text" value="1.000"/>	1.000	<input type="text" value="0.000"/>	<input type="text" value="0.000"/>	<input type="checkbox"/>
	Sin+Cos	<input type="text" value="1.000"/>	<input type="text" value="0.000"/>	<input type="text" value="1.000"/>	1.000	<input type="text" value="0.000"/>	<input type="text" value="0.000"/>	<input type="checkbox"/>
	Sin*Cos	<input type="text" value="1.000"/>	<input type="text" value="0.000"/>	<input type="text" value="1.000"/>	1.000	<input type="text" value="0.000"/>	<input type="text" value="0.000"/>	<input type="checkbox"/>
	Sin sums	<input type="text" value="1.000"/>	<input type="text" value="0.000"/>	<input type="text" value="1.000"/>	1.000	<input type="text" value="0.000"/>	<input type="text" value="0.000"/>	<input type="checkbox"/>

ADCSimDetector Driver

NDPluginTimeSeries.adl

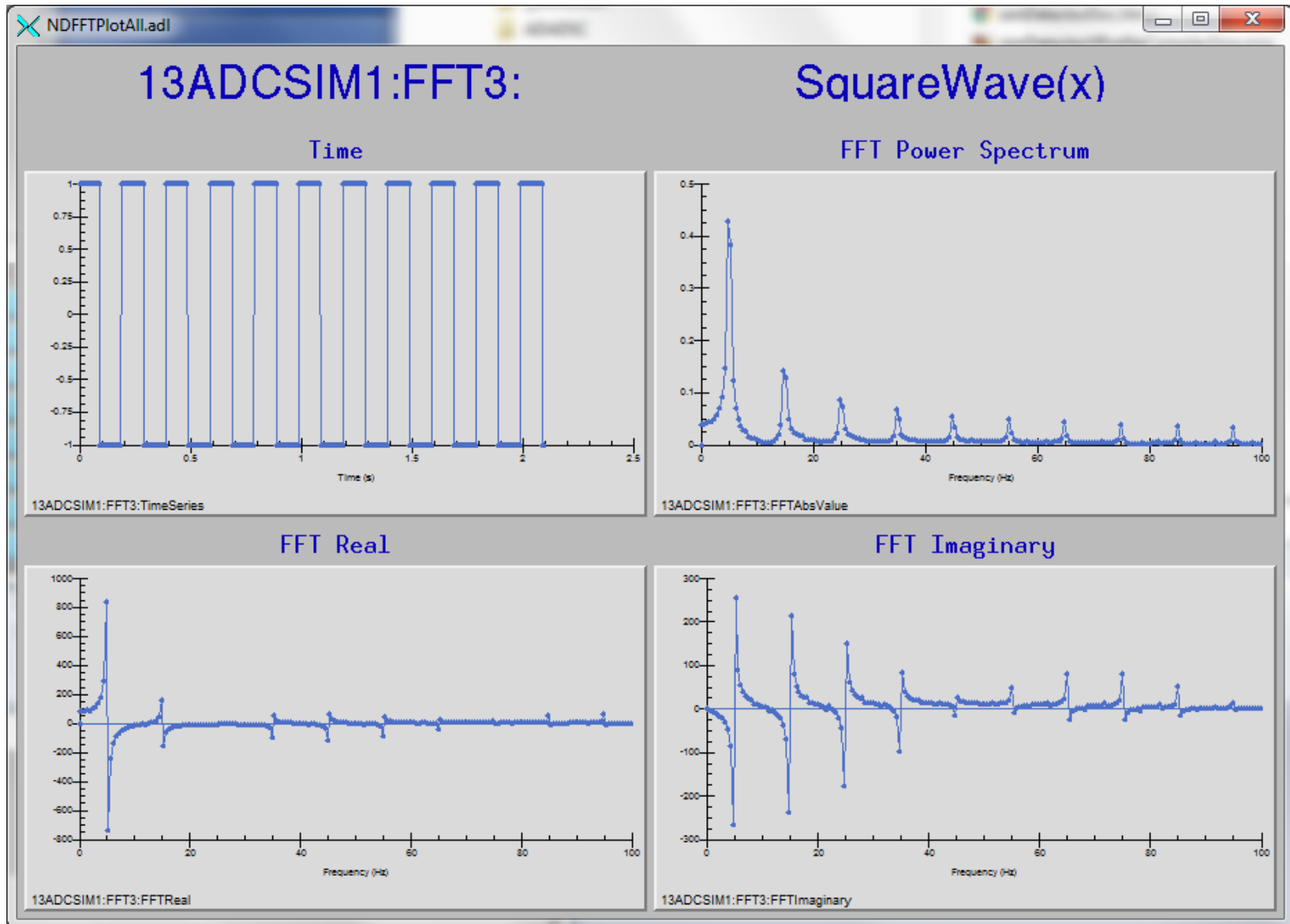
13ADCSIM1:TS:

asyn port	TS1
Plugin type	NDPluginTimeSeries
Array port	SIM1 SIM1
Array address	0 0
Enable	<input type="checkbox"/> Enable Enable
Min. time	0.000 0.000
Callbacks block	<input type="checkbox"/> No No
Queue size/free	20 20
Array counter	0 4536
Array rate	20.00
Dropped arrays	0 0
# dimensions	2
Array Size	8 50 0
Data type	Float64
Color mode	Mono
Bayer pattern	RGBB
Unique ID	4535
Time stamp	827354508.869
Attributes file	
Array callbacks	<input type="checkbox"/> Enable Enable
asyn record	<input type="checkbox"/>

Acquire	<input type="button" value="Erase/Start"/>	<input type="button" value="Stop"/>
Status	Done	
Elapsed time	21.547	
Time link	13ADCSIM1:det1	
Time/point	1.024e-03 1.024e-03	
Averaging time	1.000e-03	
# Average	1	
AcquireMode	<input type="checkbox"/> Circ. buffer	
# Time points	2048	
Current point	470	
Read rate	<input type="checkbox"/> 1 second	<input type="button" value="Read"/>

Time series Plots

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

NDFileHDF5.adl

13SIM1:HDF1:

asyn port	FileHDF1
Plugin type	NDFileHDF5 ver1.8.7
Array port	PROC1 PROC1
Array address	0 0
Enable	<input type="checkbox"/> Enable <input checked="" type="checkbox"/> Enable
Min. time	0.000 0.000
Callbacks block	<input type="checkbox"/> No <input checked="" type="checkbox"/> No
Queue size/free	200 11
Array counter	0 660
Array rate	17.00
Dropped arrays	0 681
# dimensions	2
Array Size	1024 1024 0
Data type	Float64
Color mode	Mono
Bayer pattern	RGGB
Unique ID	20979
Time stamp	830028816.094
Attributes file	
Array callbacks	<input type="checkbox"/> Disable <input checked="" type="checkbox"/> Disable
asyn record	<input type="checkbox"/>

File path	/home/epics/scratch/	Exists: Yes
File name	test	Create dir. depth 0 0 <input type="checkbox"/> Help
Next file #	29 29	Temp. suffix
Auto increment	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> Yes	Lazy open <input type="checkbox"/> No <input checked="" type="checkbox"/> No
Filename format	%%s_%.3d.h5	Example: %%s_%.3d.h5
Last filename	/home/epics/scratch/test_028.h5	
Save file	<input type="button" value="Save"/> <input type="button" value="Read file"/> <input type="button" value="Read"/> <input type="button" value="Auto save"/> <input type="checkbox"/> No <input checked="" type="checkbox"/> No	
Write mode	<input type="checkbox"/> Stream <input checked="" type="checkbox"/> Stream	# Capture 50 50 50
Capture	<input type="button" value="Start"/> <input type="button" value="Stop"/>	Delete driver file <input type="checkbox"/> No <input checked="" type="checkbox"/> No
Write status	Write OK	
Write message		

Compression	<input type="checkbox"/> None <input checked="" type="checkbox"/> None	Extra dimensions
# data bits	8 8	# (0-2) 0 0
Data bits offset	0 0	Size N 1 1
SZip # pixels	16 16	Name N frame number n
Zlib level	6 6	Size X 1 1
Store performance	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> Yes	Name X scan dimension X
Store attributes	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> Yes	Size Y 1 1
Run time	2.867	Name Y scan dimension Y
I/O speed	1116.3	

Rows per chunk	0 1024
Columns per chunk	0 1024
Frames cached per chunk	0 1
Boundary alignment	0 0
Boundary threshold	65536 65536
Flush on N'th frame	0 0

Default layout selected

Exists: Yes

XML File name

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