

EPICS integration & IOC

Hinko Kocevar
Software Engineer

www.europeanspallationsource.se

12 June 2017

Outline

- Status at PDR
- Work after PDR
- Status at CDR
- Work after CDR

Crew and tasks

- Hardware Hooman Hassanzadegan
- Firmware Klemen Erjavec and Dušan Slavinec (Cosylab)
- Software Hinko Kocevar
- Cabling Hooman Hassanzadegan
- Testing / debugging / data evaluation Hooman Hassanzadegan
- Coordination made through JIRA issues
- Small ICS involvement due to lack of resources

Status at PDR

- Demo EPICS support and IOC developed by Cosylab in 2014
- Demonstrated that chosen hardware is adequate for BCM application
- Limited functionality and reduced set of features
- Based on obsoleted SIS8300 digitizer board

Work after PDR

- Work on final BCM EPICS support started in Q3 2016
- First task was to develop generic Struck SIS8300 digitizer support
 - Previously based on NDS
 - Now based on areaDetector (asynNDArrayDriver C++ class)
 - Allows control and status reporting of the digitizer and RTM
 - Delivers 10 channels of digitized analog input data as waveform PVs to clients
- Integration with MTCA based EVR and mrfioc2

Work after PDR

- Once generic SIS8300 support was in place, BCM application development was started
- BCM EPICS support derives C++ class of generic SIS8300
- Along the way generic support was modularized and improved
 - Fairly straight forward since BCM is derived from generic
- All aspects of control and data transfer were completed in 2 months
 - Also with CSS OPI

Work after PDR

- Ten instances of BCM are supported
- User controls:
 - Clock source, trigger source, DSP parameters, alarms, select to monitor desired BCM channels, amount of samples to request, monitoring of BCM core status, setup EVR for trigger generation
- Data streams
 - 10 raw digitized data streams, 10 BCM specific data streams, debugging streams
- AD plugins utilization (ROI, stats, proc, HDF5, ..)

- Top level CSS OPI
 - General info
 - Acquisition control
 - Access to other OPI
 - RTM selection
 - BCM instance OPI access
 - AFE control and monitoring (not seen here)

BCM on Struck SIS8300 - BCM:

<p>General Info</p> <p>Device type: 0x8302 Firmware version: 0x200F Serial number: 92 Memory size: 2048 MB</p>	<p>Collect</p> <p>Data type: <input type="text" value="Float64"/> Float64 # AI Samples: <input type="text" value="300000"/> 300000 Array counter: <input type="text" value="0"/> 0 Acquire: <input type="button" value="Start"/> <input type="button" value="Stop"/> <input type="button" value="Done"/></p>
<p>Plugins</p> <p>All: <input type="button" value="File"/> <input type="button" value="ROI"/> Stats: <input type="button" value="Other"/> Time series: <input type="button" value="FFTs"/></p>	<p>Parameters</p> <p><input type="button" value="Update & Refresh"/></p>
<p>Channel control</p> <p>Enable / Disable: <input checked="" type="checkbox"/> Declm. Factor: <input type="text" value="0"/> Declm. Offset: <input type="text" value="0"/> Conv. Factor: <input type="text" value="0"/> Conv. Offset: <input type="text" value="0"/> Channels: <input type="text" value="Channels"/></p>	<p>Acquisition control</p> <p>Clock source: <input type="text" value="SMA"/> SMA Clock divider: <input type="text" value="1"/> 1 Trigger source: <input type="text" value="External"/> External Trigger line: <input type="text" value="BackPlane0"/> BackPlane0 Trigger delay: <input type="text" value="0"/> 0 Trigger repeat: <input type="text" value="0"/> 0</p>
<p>RTM control</p> <p>RTM type: <input type="text" value="SIS8900"/> SIS8900 RTM temp. 1: 0.0 C RTM temp. 2: 0.0 C <input type="button" value="Temperature refresh"/></p>	<p>Debugging</p> <p>Message: [INF] SIS8300::SIS8300: No error</p>
<p>Bcm</p> <p><input type="button" value="BCM common"/> <input type="button" value="BCM0"/> <input type="button" value="BCM1"/> <input type="button" value="BCM2"/> <input type="button" value="BCM3"/> <input type="button" value="BCM4"/> <input type="button" value="BCM5"/> <input type="button" value="BCM6"/> <input type="button" value="BCM7"/> <input type="button" value="BCM8"/> <input type="button" value="BCM9"/></p>	

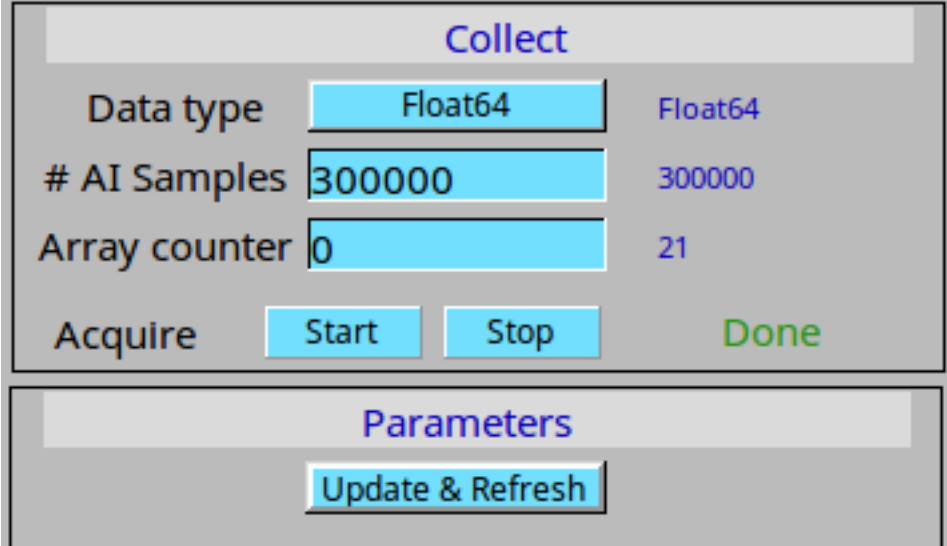
CSS OPI general info

- Some generic SIS8300-x digitizer information

General info	
Device type	0x8302
Firmware version	0x200F
Serial number	92
Memory size	2048 MB

CSS OPI acquisition control

- Selectable data type
- Desired number of raw samples
- Number of arrays (pulses) collected so far
- Force parameter update
 - FPGA parameters are latched and automatically applied between the pulses



The screenshot shows a software interface for data acquisition control, divided into two main sections: "Collect" and "Parameters".

Collect Section:





- Data type:** A dropdown menu showing "Float64". To its right, the text "Float64" is displayed in purple.
- # AI Samples:** A text input field containing "300000". To its right, the text "300000" is displayed in purple.
- Array counter:** A text input field containing "0". To its right, the text "21" is displayed in purple.
- Acquire:** A label followed by two buttons: "Start" (light blue) and "Stop" (light blue).
- Done:** A green text label.

Parameters Section:

- Update & Refresh:** A light blue button.


CSS OPI analog channels

- Channel enable/disable
- Software scaling and offset control
- No attenuation control

Channel	Status	Decimation		Conversion		Attenuati	
Ch0		Factor	<input type="text" value="1"/> 1	<input type="text" value="1.00000000"/> 1.00000000	<input type="text" value="0.000"/> 0.000	<input type="text" value="0.0"/> 0.0	<input type="button" value="Plots"/>
		Offset	<input type="text" value="0"/> 0	<input type="text" value="0.000"/> 0.000			
Ch1		Factor	<input type="text" value="1"/> 1	<input type="text" value="1.00000000"/> 1.00000000	<input type="text" value="0.000"/> 0.000	<input type="text" value="0.0"/> 0.0	<input type="button" value="Plots"/>
		Offset	<input type="text" value="0"/> 0	<input type="text" value="0.000"/> 0.000			
Ch2		Factor	<input type="text" value="1"/> 1	<input type="text" value="1.00000000"/> 1.00000000	<input type="text" value="0.000"/> 0.000	<input type="text" value="0.0"/> 0.0	<input type="button" value="Plots"/>
		Offset	<input type="text" value="0"/> 0	<input type="text" value="0.000"/> 0.000			
Ch3		Factor	<input type="text" value="1"/> 1	<input type="text" value="1.00000000"/> 1.00000000	<input type="text" value="0.000"/> 0.000	<input type="text" value="0.0"/> 0.0	<input type="button" value="Plots"/>
		Offset	<input type="text" value="0"/> 0	<input type="text" value="0.000"/> 0.000			

Channel control

Enable / Disable



Decim. Factor

Decim. Offset

Conv. Factor

Conv. Offset

Channels

CSS OPI acquisition control

- Selectable clock source
 - backplane, RTM, front panel, internal
- Clock divider
 - integer divisor
- Trigger repeat
 - how many pulses to acquire
- Trigger source & line
 - want to use external (EVR),
 - any available backplane line

Acquisition control		
Clock source	<input type="text" value="SMA"/>	SMA
Clock divider	<input type="text" value="1"/>	1
Trigger source	<input type="text" value="External"/>	External
Trigger line	<input type="text" value="BackPlane0"/>	BackPlane0
Trigger delay	<input type="text" value="0"/>	0
Trigger repeat	<input type="text" value="0"/>	0

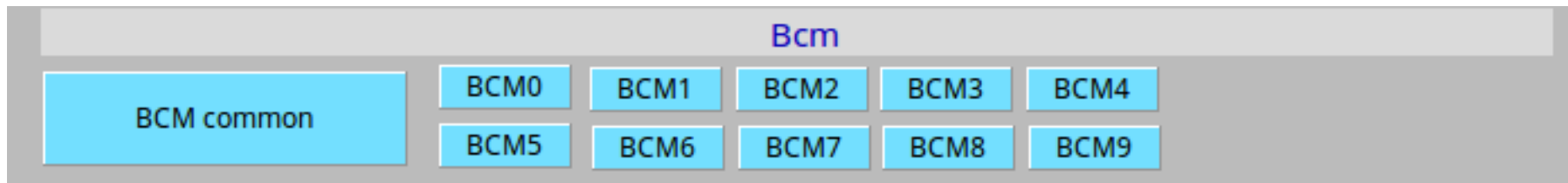
CSS OPI RTM controls

- One type of RTM
 - At least for BCM
- No RTM temperature sensors readout
 - RTM has them but accessible only over IPMIO

RTM control

RTM type	<input type="text" value="SIS8900"/>	SIS8900
RTM temp. 1		0.0 C
RTM temp. 2		0.0 C
<input type="button" value="Temperature refresh"/>		

- Ten instances of BCM DSP on single AMC
 - Instances are pretty much independent, but..



- Some controls are common for both instances
 - Trigger source, clock source, decimation, pulse detection conditions, pulse parameters for alarms,..

CSS OPI common BCM

- Collection of common BCM controls and statuses
- These apply to all BCM instances
- Snag: how to name them in control room?

Trigger Source	Trigger logic	Trigger logic
Ignore DSP Core Busy	No	No
Minimum Trigger Period Source	FW register	FW register
Maximum Pulse Width Source	FW register	FW register
RFQ Transparency	0	0
Reset Difference Warnings	Reset	
Minimum Trigger Period	0	0
Maximum Pulse Width	0	0
Calibration Pulse	<input checked="" type="checkbox"/>	Disabled
Beam Over Threshold	0	0
Reference ADC clock frequency	88052	0
Reference ADC clock threshold	20	0
Automatic Reset ADC FIFOs	<input checked="" type="checkbox"/>	Enabled
DSP Core Status		0x0
	<input checked="" type="checkbox"/> DSP Core Busy	
	<input checked="" type="checkbox"/> Register Read Failed	
	<input checked="" type="checkbox"/> Register Write Failed	
	<input checked="" type="checkbox"/> AUX Clock Missing	
	<input checked="" type="checkbox"/> Main Clock Missing	
Clock Frequency [kHz]		0
Trigger Period		0
Trigger Width		0
Flat Top Time		0
Reset MPS Alarms	Reset	
All Alarms	<input checked="" type="checkbox"/>	Enabled
Main Clock Alarm	<input checked="" type="checkbox"/>	Enabled
AUX Clock Alarm	<input checked="" type="checkbox"/>	Enabled
Trigger Too Short Alarm	<input checked="" type="checkbox"/>	Enabled
Trigger Too Long Alarm	<input checked="" type="checkbox"/>	Enabled
Trigger Too Fast Alarm	<input checked="" type="checkbox"/>	Enabled
Latched Clock & Trigger Alarms	<input checked="" type="checkbox"/> Trigger Too Fast <input checked="" type="checkbox"/> Trigger Too Short <input checked="" type="checkbox"/> Trigger Too Long <input checked="" type="checkbox"/> Main Clock Invalid <input checked="" type="checkbox"/> AUX Clock Invalid	0x0
First Clock & Trigger Alarms	<input checked="" type="checkbox"/> Trigger Too Fast <input checked="" type="checkbox"/> Trigger Too Short <input checked="" type="checkbox"/> Trigger Too Long <input checked="" type="checkbox"/> Main Clock Invalid <input checked="" type="checkbox"/> AUX Clock Invalid	0x0
Probe 0 source	0	0
Probe 1 source	0	0
Probe 2 source	0	0
Probe 3 source	0	0

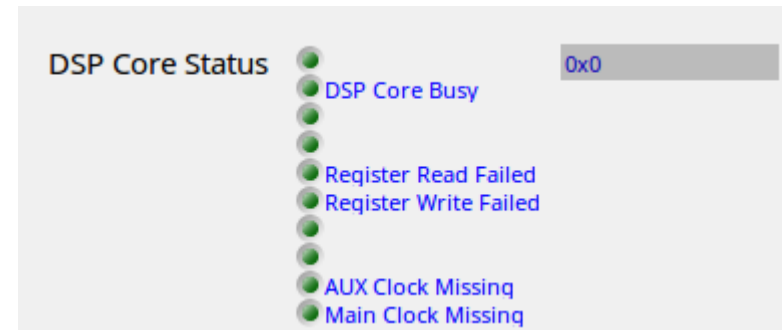
CSS OPI BCM DSP configuration

- BCM DSP core configuration parameters
- Latched after the pulse

Trigger Source	<input type="text" value="Trigger logic"/>	<input type="text" value="Trigger logic"/>
Ignore DSP Core Busy	<input type="text" value="No"/>	<input type="text" value="No"/>
Minimum Trigger Period Source	<input type="text" value="FW register"/>	<input type="text" value="FW register"/>
Maximum Pulse Width Source	<input type="text" value="FW register"/>	<input type="text" value="FW register"/>
RFQ Transparency	<input type="text" value="0"/>	<input type="text" value="0"/>
Reset Difference Warnings	<input type="text" value="Reset"/>	
Minimum Trigger Period	<input type="text" value="0"/>	<input type="text" value="0"/>
Maximum Pulse Width	<input type="text" value="0"/>	<input type="text" value="0"/>
Calibration Pulse	<input checked="" type="checkbox"/>	<input type="text" value="Disabled"/>
Beam Over Threshold	<input type="text" value="0"/>	<input type="text" value="0"/>
Reference ADC clock frequency	<input type="text" value="88052"/>	<input type="text" value="0"/>
Reference ADC clock threshold	<input type="text" value="20"/>	<input type="text" value="0"/>

CSS OPI BCM DSP status

- DSP provides core status access status
- Refreshed per pulse



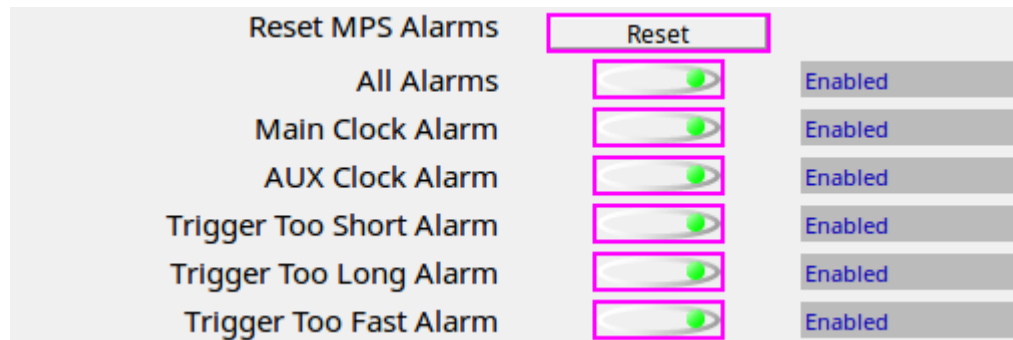
CSS OPI BCM pulse measurements

- BCM DSP measures several pulse parameters including ADC sampling clock frequency
- Can raise alarm if not in bounds as configured

Clock Frequency [kHz]	0
Trigger Period	0
Trigger Width	0
Flat Top Time	0

CSS OPI common BCM alarm controls

- BCM DSP can be configured to raise alarm on several conditions



- Alarm status is available for monitoring purposes as well as for MPS interfacing

CSS OPI common BCM alarm monitors

- Individual alarm status is delivered per pulse
- Can be cleared on demand by the user

Latched Clock & Trigger Alarms	<ul style="list-style-type: none">● Trigger Too Fast● Trigger Too Short● Trigger Too Long● Main Clock Invalid● AUX Clock Invalid	0x0
First Clock & Trigger Alarms	<ul style="list-style-type: none">● Trigger Too Fast● Trigger Too Short● Trigger Too Long● Main Clock Invalid● AUX Clock Invalid	0x0

CSS OPI BCM probes

- With probes we can tap into certain point along the DSP processing chain
- Configurable per channel
- Up to 64 tap sources available per channel
- Used for debugging

Probe 0 source	<input type="text" value="0"/>	<input type="text" value="0"/>
Probe 1 source	<input type="text" value="0"/>	<input type="text" value="0"/>
Probe 2 source	<input type="text" value="0"/>	<input type="text" value="0"/>
Probe 3 source	<input type="text" value="0"/>	<input type="text" value="0"/>

CSS OPI channel BCM alarm controls

- Individual BCM channel can be configured to raise an alarm on several conditions

ADC Scale	<input type="text" value="0"/>	<input type="text" value="0"/>
ADC Offset	<input type="text" value="0"/>	<input type="text" value="0"/>
Trigger Fine Delay	<input type="text" value="0"/>	<input type="text" value="0"/>
Upper Threshold	<input type="text" value="0"/>	<input type="text" value="0"/>
Lower Threshold	<input type="text" value="0"/>	<input type="text" value="0"/>
Errant Threshold	<input type="text" value="0"/>	<input type="text" value="0"/>

Alarm Control		
Upper Threshold	<input checked="" type="checkbox"/>	Enabled
Lower Threshold	<input checked="" type="checkbox"/>	Enabled
Errant Threshold	<input checked="" type="checkbox"/>	Enabled
Pulse Past Trigger	<input checked="" type="checkbox"/>	Enabled
Pulse Past Limit	<input checked="" type="checkbox"/>	Enabled
Overflow	<input checked="" type="checkbox"/>	Enabled
Underflow	<input checked="" type="checkbox"/>	Enabled
ADC Stuck	<input checked="" type="checkbox"/>	Enabled

- Alarm status is available for monitoring purposes as well as for MPS interfacing

CSS OPI channel BCM alarm monitors

- Individual alarm status is delivered per pulse
- Can be cleared on demand by the user

Alarm First Status	Alarm Latched Status
0x0	0x0
<input type="radio"/> ADC Stuck	<input type="radio"/> ADC Stuck
<input type="radio"/> Underflow	<input type="radio"/> Underflow
<input type="radio"/> Overflow	<input type="radio"/> Overflow
<input type="radio"/> Pulse Over Limit	<input type="radio"/> Pulse Over Limit
<input type="radio"/> Pulse Over Trigger	<input type="radio"/> Pulse Over Trigger
<input type="radio"/> Errant Threshold	<input type="radio"/> Errant Threshold
<input type="radio"/> Lower Threshold	<input type="radio"/> Lower Threshold
<input type="radio"/> Upper Threshold	<input type="radio"/> Upper Threshold

CSS OPI channel BCM configuration

- Droop compensation
- Baseline correction
- Filtering
- Location along the linac

Droop Rate	<input type="text" value="0"/>	0
Droop & Baseline	<input type="checkbox"/>	Disabled
Noise Filter	<input type="checkbox"/>	Disabled
MEBT Location	<input type="text" value="After"/>	After
RFQ Location	<input type="text" value="After"/>	After

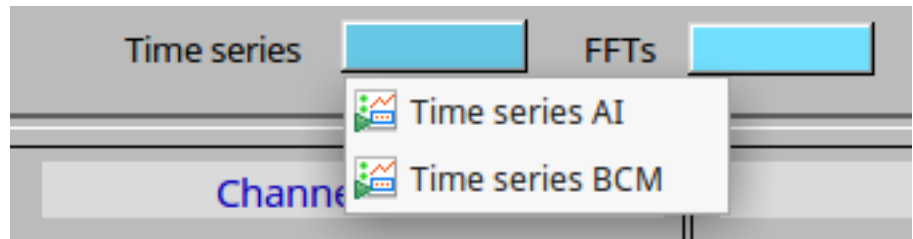
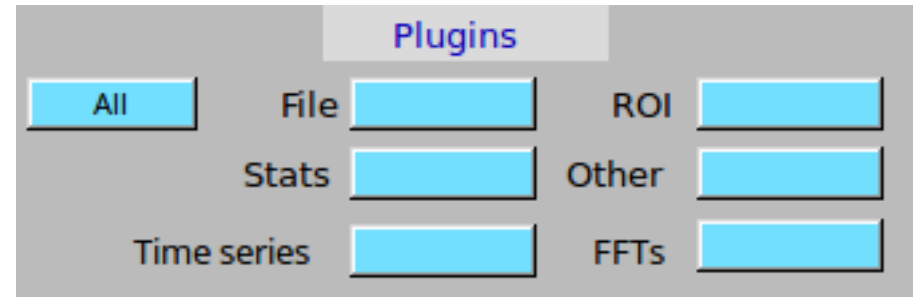
CSS OPI channel BCM monitoring

- DSP delivers measurements of several parameters per pulse

Measured	
ADC Offset Error Average	0
ADC Offset Error Integrated	0
Trigger To Pulse Time	0
Droop Error	0
Pulse Width	0
Pulse Charge	0
Flat Top Charge	0
Calibration Pulse Amplitude Early	0
Calibration Pulse Amplitude Late	0

CSS OPI AD plugins

- AreaDetector provides plugins
 - Chain data stream/array
 - Source can vary, so can destination
- Many provided out of the box
 - Statistics, processing, time series, ROI, saving to file,...
- Can develop own plugins
 - Already done for other systems



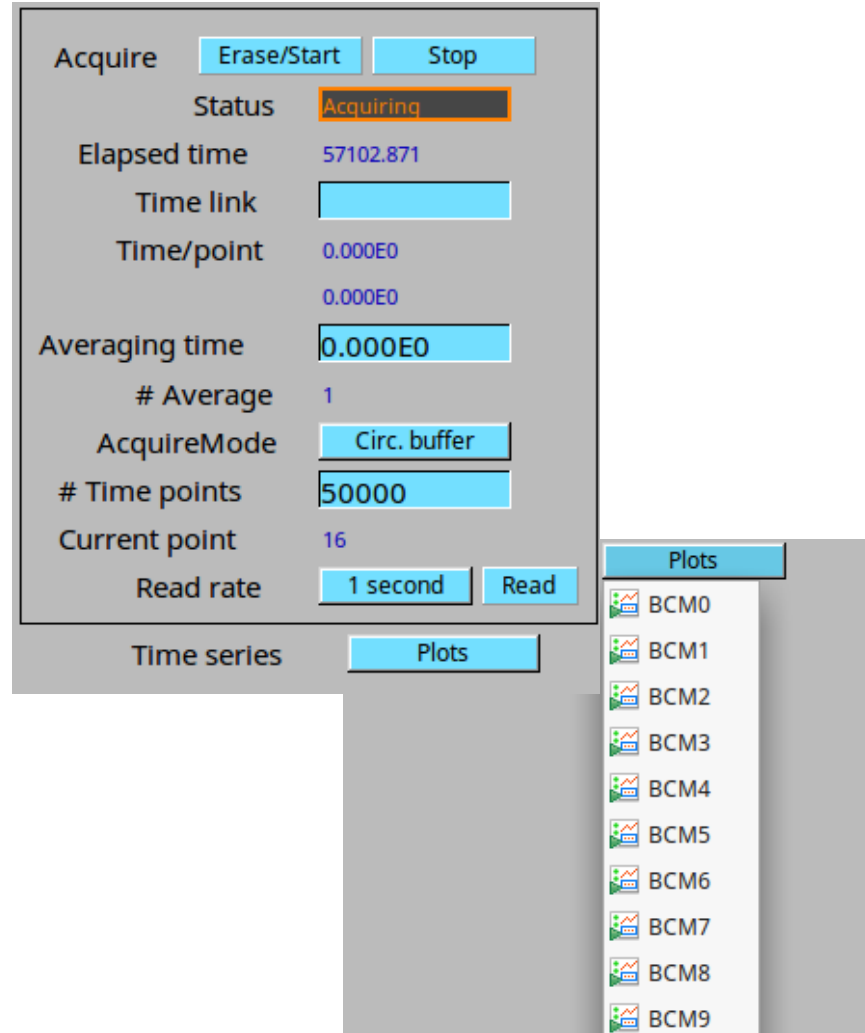
CSS OPI AD plugin setup

- Generic for all AD plugins
- Define data source
- Monitor array arrival to the plugin
- Inspect array properties
- Control publishing resulting array to other plugins

asyn port	TS1		
Plugin type	NDPluginTimeSeries		
Array port	BCM	BCM	
Array address	1	1	
Enable	Enable	1	
Min. time	0.000	0.000	
Callbacks block	No	0	
Queue size/free	20	0	
Array counter	0	0	
Array rate	0.00		
Dropped arrays	0	0	
# dimensions	0		
Array Size	0	0	0
Data type	UInt8		
Color mode	Mono		
Bayer pattern	RGGB		
Unique ID	0		
Time stamp	0.000		
Attributes file			
Array callbacks	Enable	1	
asyn record			

CSS OPI time series plugin control

- Control acquisition
- Averaging
- Mode
 - Circular buffer or one shot
- Number of points in the buffer
- Publishing rate
- Access to individual plots



The screenshot displays the CSS OPI time series plugin control interface. It features a main control panel with the following elements:

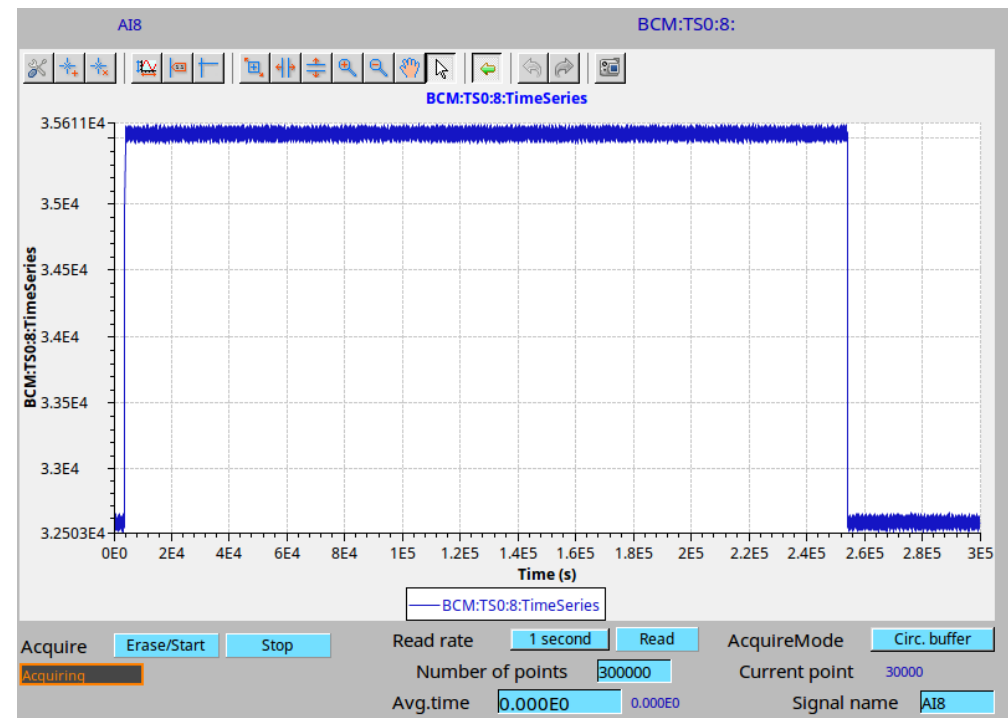
- Buttons:** "Acquire", "Erase/Start", and "Stop" at the top.
- Status:** A dropdown menu currently showing "Acquiring".
- Elapsed time:** A text field displaying "57102.871".
- Time link:** An empty text field.
- Time/point:** Two text fields, both displaying "0.000E0".
- Averaging time:** A text field displaying "0.000E0".
- # Average:** A text field displaying "1".
- AcquireMode:** A dropdown menu currently showing "Circ. buffer".
- # Time points:** A text field displaying "50000".
- Current point:** A text field displaying "16".
- Read rate:** A text field displaying "1 second" and a "Read" button.
- Time series:** A label and a "Plots" button.

To the right of the main panel is a "Plots" section with a list of plots, each with a small icon and a label:

- BCM0
- BCM1
- BCM2
- BCM3
- BCM4
- BCM5
- BCM6
- BCM7
- BCM8
- BCM9

CSS OPI time series plot

- Plot of BCM8 analog input
- Holds 1 pulse



Status at CDR

- Firmware debugged and tested at Cosylab
- Firmware & software debugged and tested also at ESS with EPICS & IOC
- Updated EPICS support with latest firmware features
- Works with final BCM RTM
- Works with SIS8300-L2 AMC (final?)
- Several over-the-weekend run tests passed
-

Work after CDR

- Handover the EPICS support for SIS8300 and BCM to ICS
- Perform integration with multiple SIS8300 AMCs in the same crate
- Long term and stability tests (IPMI IOC)
- Python scripts for automated tests and reports
- Determine minimum set of PVs and amount of data to be delivered (as per BP requirements)

Work after CDR

- Add recently added FPGA parameters
- Solve some outstanding firmware issues (ADC stuck)
- Assess performance of the CPU with multiple AMCs / BPMs
- Test, test, test..

Questions ?