

# Example of psana analysis for CSPad2x2

- [Setup environment](#)
- [Examples for xpp50312](#)
  - [Pedestals](#)
  - [Calibration](#)
  - [Analysis](#)
- [Examples for meca6113](#)
  - [Pedestals](#)
  - [Calibration](#)
  - [Create images in tiff format](#)
- [Examples for meca1113](#)
  - [Get latest version of packages and run psana](#)
  - [Pedestals and bad pixels](#)
  - [Calibration](#)
  - [Create images in tiff format](#)
- [Examples for mecb3114](#)
- [Examples for meci0114](#)
  - [Get and run this example in current release](#)
  - [Configuration files](#)
- [Other option to save image in 16-bit tiff files](#)
- [Display tiff image](#)
- [Check tiff file](#)
- [Convert uint16 tiff to int16 tiff](#)
- [Examples for cxi86415](#)
  - [Standard configuration file](#)
  - [Loh's configuration file](#)
- [References](#)

## Setup environment

```
ssh pslogin
ssh psana
cd <your-directory>
sit_setup
```

## Examples for xpp50312

### Pedestals

In this section we show how to get CSPad2x2 pedestals using psana for "dark" run.

For this example we create the configuration file `ana-xpp50312/psana-xpp50312-r0082-pedestals.cfg`:

```
[psana]
files    = /reg/d/ana01/xpp/xpp50312/xtc/e178-r0082-s04-c00.xtc
modules  = cspad_mod.CsPad2x2Pedestals

[cspad_mod.CsPad2x2Pedestals]
#source  = DetInfo(:Cspad2x2)
source   = DetInfo(XppGon.0:Cspad2x2.0)
output   = cspad-pedestals-ave-xpp50312-r0082.dat
noise    = cspad-pedestals-rms-xpp50312-r0082.dat
```

Run psana with this configuration file:

```
psana -c ana-xpp50312/psana-xpp50312-r0082-pedestals.cfg
```

In a few seconds this job produces two text files with averaged pedestals and rms values as one-dimensional (143560) array for CSPad2x2. Actual shape of these arrays is assumed to be (185,388,2).

Using, for example, a simple python script these files may be plotted

```
./PlotCSPad2x2ArrayFromFile.py cspad-pedestals-ave-xpp50312-r0082.dat 700 1400
./PlotCSPad2x2ArrayFromFile.py cspad-pedestals-rms-xpp50312-r0082.dat 0 20
```

Averaged pedestals:

- RMS of pedestals:

-

## Calibration

By default psana assumes that all calibration files are located in the pre-defined directory

/reg/d/psdm/<INSTRUMENT>/<experiment>/calib/ as explained in [Locating calibration files](#). However, for analysis it may be convenient to re-define the calibration directory through the [psana parameters](#). In order to fulfill the naming conventions we have to create a tree of the calibration sub-directories somewhere in local user's space, for example:

```
ana-xpp50312/calib/CsPad2x2::CalibV1/XppGon.0:Cspad2x2.0/pixel_gain/  
                                pixel_status/  
                                pedestals/  
                                common_mode/
```

and populate them with all necessary for analysis files. In particular, for pedestals we can use:

```
cp cspad-pedestals-ave-xpp50312-r0082.dat ana-xpp50312/calib/CsPad2x2::CalibV1/XppGon.0:Cspad2x2.0/pedestals/82-end.data
```

Another file for common mode correction common\_mode/0-end.data usually looks like

```
1 50 10
```

The files for pixel\_status and pixel\_gain have the same structure like for pedestals and can be supplied, if necessary, depending and based on analysis.

## Analysis

For this example we create the configuration file ana-xpp50312/psana-xpp50312-r0092.cfg:

```
[psana]  
files      = /reg/d/ana01/xpp/xpp50312/xtc/e178-r0092-s04-c00.xtc  
experiment = xpp50312  
calib-dir  = ana-xpp50312/calib  
events     = 10  
  
modules = cspad_mod.CsPadCalib CSPadPixCoords.CSPad2x2ImageProducer CSPadPixCoords.SaveImageInFile  
  
[cspad_mod.CsPadCalib]  
inputKey      =  
outputKey     = calibrated  
doPedestals   = yes  
doPixelStatus = no  
doCommonMode  = yes  
doPixelGain   = no  
  
[CSPadPixCoords.CSPad2x2ImageProducer]  
source        = DetInfo(:Cspad2x2)  
inkey         = calibrated  
outimgkey     = Image  
tiltIsApplied = true  
print_bits    = 15  
  
[CSPadPixCoords.SaveImageInFile]  
source        = DetInfo(:Cspad2x2)  
key           = Image  
fname         = cspad2x2  
eventSave     = 5  
#saveAll      = true
```

This configuration file provides parameters for psana and three modules.

So far we setup calibration files for `cspad_mod.CsPadCalib` module for pedestal subtraction and common mode correction. Module `CSPadPixCoords`. `CSPad2x2ImageProducer` produces the `CSPad2x2` image and saves it in the event. Module `SaveImageInFile` saves indicated image(s) in the file(s).

Run psana with this configuration file:

```
psana -c ana-xpp50312/psana-xpp50312-r0092.cfg
```

Plot obtained text file with an image:

```
./PlotCameraImageFromFile.py cspad2x2-r0092-2012-07-20-154000.855680176.txt -20 120
```

Playing with `doPedestals` and `doCommonMode` parameters, we could get different images.

Raw data without any correction:

-

Pedestals subtracted:

-

Pedestals subtracted and common mode corrected:

-

## Examples for meca6113

### Pedestals

Configuration file for dark run processing (assuming that meca6113-r0018 is a dark run...) [psana-meca6113-r0018-cspad2x2-pedestals.cfg](#):

```
# File: psana-meca6113-r0018-cspad2x2-pedestals.cfg

[psana]
files    = /reg/d/psdm/mec/meca6113/xtc/e332-r0018-s03-c00.xtc
modules  = cspad_mod.CsPad2x2Pedestals:1 \
           cspad_mod.CsPad2x2Pedestals:2 \
           cspad_mod.CsPad2x2Pedestals:3

[cspad_mod.CsPad2x2Pedestals:1]
source   = DetInfo(MecTargetChamber.0:Cspad2x2.1)
output   = pedestals-ave-meca6113-r0018-Cspad2x2.1.dat
noise    = pedestals-rms-meca6113-r0018-Cspad2x2.1.dat

[cspad_mod.CsPad2x2Pedestals:2]
source   = DetInfo(MecTargetChamber.0:Cspad2x2.2)
output   = pedestals-ave-meca6113-r0018-Cspad2x2.2.dat
noise    = pedestals-rms-meca6113-r0018-Cspad2x2.2.dat

[cspad_mod.CsPad2x2Pedestals:3]
source   = DetInfo(MecTargetChamber.0:Cspad2x2.3)
output   = pedestals-ave-meca6113-r0018-Cspad2x2.3.dat
noise    = pedestals-rms-meca6113-r0018-Cspad2x2.3.dat
```

To produce files with pedestals use command:

```
psana -c psana-meca6113-r0018-cspad2x2-pedestals.cfg
```

which produces 6 files with average and rms values for 3 detectors:

```
pedestals-ave-meca6113-r0018-Cspad2x2.1.dat
pedestals-ave-meca6113-r0018-Cspad2x2.2.dat
pedestals-ave-meca6113-r0018-Cspad2x2.3.dat
pedestals-rms-meca6113-r0018-Cspad2x2.1.dat
pedestals-rms-meca6113-r0018-Cspad2x2.2.dat
pedestals-rms-meca6113-r0018-Cspad2x2.3.dat
```

Example of pedestal-image and spectral plots for 3 detectors, respectively:

```
..
..
..
```

## Calibration

Default calibration in psana will be done if files are located in right place. So, they were deployed as:

```
cp pedestals-ave-meca6113-r0018-Cspad2x2.1.dat /reg/d/psdm/mec/meca6113/calib/CsPad2x2::CalibV1
/MecTargetChamber.0:Cspad2x2.1/pedestals/18-end.data
cp pedestals-ave-meca6113-r0018-Cspad2x2.2.dat /reg/d/psdm/mec/meca6113/calib/CsPad2x2::CalibV1
/MecTargetChamber.0:Cspad2x2.2/pedestals/18-end.data
cp pedestals-ave-meca6113-r0018-Cspad2x2.3.dat /reg/d/psdm/mec/meca6113/calib/CsPad2x2::CalibV1
/MecTargetChamber.0:Cspad2x2.3/pedestals/18-end.data
```

## Create images in tiff format

Configuration file for psana [psana-meca6113-r0028-cspad2x2-tiff.cfg](#):

```
# File: psana-meca6113-r0028-cspad2x2-tiff.cfg
#
# To run this script use command:
# % psana -c psana-meca6113-r0028-cspad2x2-tiff.cfg
#
# Also useful command:
# % psana -m EventKeys -n 10 /reg/d/psdm/mec/meca6113/xtc/e332-r0028-s03-c00.xtc
#
[psana]
files      = /reg/d/psdm/mec/meca6113/xtc/e332-r0028-s03-c00.xtc

#calib-dir = ./calib

# Default calibration directory:
# calib-dir = /reg/d/psdm/mec/meca6113/calib

modules    = cspad_mod.CsPadCalib:1 \
              cspad_mod.CsPadCalib:2 \
              cspad_mod.CsPadCalib:3 \
              CSPadPixCoords.CSPad2x2ImageProducer:1 \
              CSPadPixCoords.CSPad2x2ImageProducer:2 \
              CSPadPixCoords.CSPad2x2ImageProducer:3 \
              ImgAlgos.ImgSaveInFile:1 \
              ImgAlgos.ImgSaveInFile:2 \
              ImgAlgos.ImgSaveInFile:3

# events    = 5

[cspad_mod.CsPadCalib:1]
source      = DetInfo(MecTargetChamber.0:Cspad2x2.1)
inputKey    =
outputKey   = calibrated_arr1
doPedestals = yes
doPixelStatus = no
doCommonMode = no

[cspad_mod.CsPadCalib:2]
source      = DetInfo(MecTargetChamber.0:Cspad2x2.2)
```

```

inputKey      =
outputKey     = calibrated_arr2
doPedestals  = yes
doPixelStatus = no
doCommonMode = no

[cspad_mod.CsPadCalib:3]
source        = DetInfo(MecTargetChamber.0:Cspad2x2.3)
inputKey      =
outputKey     = calibrated_arr3
doPedestals  = yes
doPixelStatus = no
doCommonMode = no

[CSPadPixCoords.CSPad2x2ImageProducer:1]
calibDir      = /reg/d/psdm/mec/meca6113/calib
typeGroupName = CsPad2x2::CalibV1
source        = DetInfo(MecTargetChamber.0:Cspad2x2.1)
inkey         = calibrated_arr1
outimgkey     = Image
tiltIsApplied = true
useWidePixCenter = false
print_bits    = 15

[CSPadPixCoords.CSPad2x2ImageProducer:2]
calibDir      = /reg/d/psdm/mec/meca6113/calib
typeGroupName = CsPad2x2::CalibV1
source        = DetInfo(MecTargetChamber.0:Cspad2x2.2)
inkey         = calibrated_arr2
outimgkey     = Image
tiltIsApplied = true
useWidePixCenter = false
print_bits    = 15

[CSPadPixCoords.CSPad2x2ImageProducer:3]
calibDir      = /reg/d/psdm/mec/meca6113/calib
typeGroupName = CsPad2x2::CalibV1
source        = DetInfo(MecTargetChamber.0:Cspad2x2.3)
inkey         = calibrated_arr3
outimgkey     = Image
tiltIsApplied = true
useWidePixCenter = false
print_bits    = 15

[ImgAlgos.ImgSaveInFile:1]
source        = DetInfo(MecTargetChamber.0:Cspad2x2.1)
key           = Image
fname         = cspad2x2.1
#ftype        = txt
ftype         = tiff
saveAll       = true
print_bits    = 3
#eventSave    = 5

[ImgAlgos.ImgSaveInFile:2]
source        = DetInfo(MecTargetChamber.0:Cspad2x2.2)
key           = Image
fname         = cspad2x2.2
#ftype        = txt
ftype         = tiff
saveAll       = true
print_bits    = 3
#eventSave    = 5

[ImgAlgos.ImgSaveInFile:3]
source        = DetInfo(MecTargetChamber.0:Cspad2x2.3)
key           = Image
fname         = cspad2x2.3
#ftype        = txt

```

```
fType      = tiff
saveAll    = true
print_bits = 3
#eventSave = 5
```

To produce files with images in tiff format use command:

```
psana -c psana-meca6113-r0028-cspad2x2-tiff.cfg
```

New files will be produced with names containing time stamp:

```
cspad2x2.1-r0028-20131013-155538.754093925.tiff
cspad2x2.2-r0028-20131013-155538.754093925.tiff
cspad2x2.3-r0028-20131013-155538.754093925.tiff
```

Example of image and spectral plots for 3 detectors, respectively:

```
..
..
..
```

## Examples for meca1113

### Get latest version of packages and run psana

On 6 p.m. 2013-11-13 all recent code changes are included in release ana-0.10.5 (seen in directory \$SIT\_RELDIR).

To run psana use commands:

```
ssh -Y psana
cd <your-favorite-directory>

sit_setup

cp <path-to-where-it-is-located>/psana-meca1113-r0376-cspad-cspad2x2-tiff.cfg .
psana -c psana-meca1113-r0376-cspad-cspad2x2-tiff.cfg
```

or run xtcexplorer:

```
xtcexplorer /reg/d/psdm/mec/meca1113/xtc/e356-r0376-s03-c00.xtc
```

## Pedestals and bad pixels

The configuration file for psana [psana-meca1113-r0045-cspad-cspad2x2-dark-hotpix.cfg](#) can be used to produce calibration files with pedestals (dark rate) and mask of bad pixels:

```
# Run this script:
# psana -c psana-meca1113-r0045-cspad-cspad2x2-dark-hotpix.cfg

[psana]
# Default calibration directory:
# calib-dir = /reg/d/psdm/mec/meca1113/calib

files = exp=meca1113:run=45
events = 400
#skip-events = 0
```

```

modules = CSPadPixCoords.CSPadNDArrProducer \
          CSPadPixCoords.CSPad2x2NDArrProducer:1 \
          CSPadPixCoords.CSPad2x2NDArrProducer:2 \
          CSPadPixCoords.CSPad2x2NDArrProducer:3 \
          CSPadPixCoords.CSPad2x2NDArrProducer:4 \
          ImgAlgos.NDArrAverage \
          ImgAlgos.NDArrAverage:1 \
          ImgAlgos.NDArrAverage:2 \
          ImgAlgos.NDArrAverage:3 \
          ImgAlgos.NDArrAverage:4 \
          ImgAlgos.Tahometer

```

```

[CSPadPixCoords.CSPadNDArrProducer]
source      = MecTargetChamber.0:Cspad.0
inkey       =
outkey      = cspad_ndarr
outtype     = int16
is_fullsize = yes
print_bits  = 3

```

```

[CSPadPixCoords.CSPad2x2NDArrProducer:1]
source = MecTargetChamber.0:Cspad2x2.1
inkey  =
outkey = cspad2x2.1_ndarr
outtype = int16
print_bits = 3

```

```

[CSPadPixCoords.CSPad2x2NDArrProducer:2]
source = MecTargetChamber.0:Cspad2x2.2
inkey  =
outkey = cspad2x2.2_ndarr
outtype = int16
print_bits = 3

```

```

[CSPadPixCoords.CSPad2x2NDArrProducer:3]
source = MecTargetChamber.0:Cspad2x2.3
inkey  =
outkey = cspad2x2.3_ndarr
outtype = int16
print_bits = 3

```

```

[CSPadPixCoords.CSPad2x2NDArrProducer:4]
source = MecTargetChamber.0:Cspad2x2.4
inkey  =
outkey = cspad2x2.4_ndarr
outtype = int16
print_bits = 3

```

```

[ImgAlgos.NDArrAverage]
source      = MecTargetChamber.0:Cspad.0
key         = cspad_ndarr
avefile     = cspad.0-ave
rmsfile     = cspad.0-rms
maskfile    = cspad.0-msk
hotpixfile  = cspad.0-hot
thr_rms_ADU = 10
thr_min_ADU = 2
thr_max_ADU = 65000
print_bits  = 93

```

```

[ImgAlgos.NDArrAverage:1]
source      = MecTargetChamber.0:Cspad2x2.1
key         = cspad2x2.1_ndarr
avefile     = cspad2x2.1-ave
rmsfile     = cspad2x2.1-rms

```

```

maskfile      = cspad2x2.1-msk
hotpixfile    = cspad2x2.1-hot
#evts_stagel  = 100
#gate_width1  = 100.
thr_rms_ADU   = 10
thr_min_ADU   = 2
thr_max_ADU   = 65000
print_bits    = 93

[ImgAlgos.NDArrAverage:2]
source        = MecTargetChamber.0:Cspad2x2.2
key           = cspad2x2.2_ndarr
avefile       = cspad2x2.2-ave
rmsfile       = cspad2x2.2-rms
maskfile      = cspad2x2.2-msk
hotpixfile    = cspad2x2.2-hot
#evts_stagel  = 100
#gate_width1  = 100.
thr_rms_ADU   = 10
thr_min_ADU   = 2
thr_max_ADU   = 65000
print_bits    = 93

[ImgAlgos.NDArrAverage:3]
source        = MecTargetChamber.0:Cspad2x2.3
key           = cspad2x2.3_ndarr
avefile       = cspad2x2.3-ave
rmsfile       = cspad2x2.3-rms
maskfile      = cspad2x2.3-msk
hotpixfile    = cspad2x2.3-hot
#evts_stagel  = 100
#gate_width1  = 100.
thr_rms_ADU   = 10
thr_min_ADU   = 2
thr_max_ADU   = 65000
print_bits    = 93

[ImgAlgos.NDArrAverage:4]
source        = MecTargetChamber.0:Cspad2x2.4
key           = cspad2x2.4_ndarr
avefile       = cspad2x2.4-ave
rmsfile       = cspad2x2.4-rms
maskfile      = cspad2x2.4-msk
hotpixfile    = cspad2x2.4-hot
#evts_stagel  = 100
#gate_width1  = 100.
thr_rms_ADU   = 10
thr_min_ADU   = 2
thr_max_ADU   = 65000
print_bits    = 93

[ImgAlgos.Tahometer]
dn            = 100
print_bits    = 7

```

This script

- runs through 400 events of the xtc file `exp=meca1113:run=45`,
- uses [CSPadNDArrProducer](#) and [CSPad2x2NDArrProducer](#) modules to get cspad and four cspad2x2 data and put them as universal ndarray<T, 3> objects in psana event store,
- uses [NDArrAverage](#) module to evaluate and save in file average, rms, mask, and hot pixel map for all array elements:



```
cspad.0-ave-meca1113-r0045.dat
cspad.0-hot-meca1113-r0045.dat
cspad.0-msk-meca1113-r0045.dat
cspad.0-rms-meca1113-r0045.dat
cspad2x2.1-ave-meca1113-r0045.dat
cspad2x2.1-hot-meca1113-r0045.dat
cspad2x2.1-msk-meca1113-r0045.dat
cspad2x2.1-rms-meca1113-r0045.dat
cspad2x2.2-ave-meca1113-r0045.dat
cspad2x2.2-hot-meca1113-r0045.dat
cspad2x2.2-msk-meca1113-r0045.dat
cspad2x2.2-rms-meca1113-r0045.dat
cspad2x2.3-ave-meca1113-r0045.dat
cspad2x2.3-hot-meca1113-r0045.dat
cspad2x2.3-msk-meca1113-r0045.dat
cspad2x2.3-rms-meca1113-r0045.dat
cspad2x2.4-ave-meca1113-r0045.dat
cspad2x2.4-hot-meca1113-r0045.dat
cspad2x2.4-msk-meca1113-r0045.dat
cspad2x2.4-rms-meca1113-r0045.dat
```

[NDArrAverage](#) module has a bunch of parameters which depend on particular detector/experiments and need to be adjusted looking at spectra of average and rms values from these files. For example averaged image and spectrum and rms spectrum of pixel intensities for

cspad2x2.1-ave-meca1113-r0045.dat and cspad2x2.1-rms-meca1113-r0045.dat are shown in plots:

These spectra allow to use confidently parameters like

- thr\_rms\_ADU = 10
- thr\_min\_ADU = 2
- thr\_max\_ADU = 65000

Files with mask and hot pixel map contain opposite meaning of 0/1 for bad/good pixels:

## Calibration

Default calibration in psana will be done if files are located in right place. So, they were deployed as:

```
cp cspad.0-ave-meca1113-r0045.dat /reg/d/psdm/mec/meca1113/calib/CsPad::CalibV1/MecTargetChamber.0:Cspad.0
/pedestals/45-end.data
cp cspad2x2.1-ave-meca1113-r0045.dat /reg/d/psdm/mec/meca1113/calib/CsPad2x2::CalibV1/MecTargetChamber.0:
Cspad2x2.1/pedestals/45-end.data
cp cspad2x2.2-ave-meca1113-r0045.dat /reg/d/psdm/mec/meca1113/calib/CsPad2x2::CalibV1/MecTargetChamber.0:
Cspad2x2.2/pedestals/45-end.data
...
cp cspad.0-hot-meca1113-r0045.dat /reg/d/psdm/mec/meca1113/calib/CsPad::CalibV1/MecTargetChamber.0:Cspad.0
/pixel_status/45-end.data
cp cspad2x2.1-hot-meca1113-r0045.dat /reg/d/psdm/mec/meca1113/calib/CsPad2x2::CalibV1/MecTargetChamber.0:
Cspad2x2.1/pixel_status/45-end.data
...
```

## Create images in tiff format

Configuration file for psana [psana-meca1113-r0376-cspad-cspad2x2-conv-tiff.cfg](#)

```
# Run this script:
# psana -c psana-meca1113-r0376-cspad-cspad2x2-conv-tiff.cfg
```

```

#
# Useful commands:
# psana -m EventKeys -n 5 exp=mecall113:run=376
# or
# psana -m EventKeys -n 5 /reg/d/psdm/mec/mecall113/xtc/e356-r0376-s03-c00.xtc

[psana]
# Default calibration directory:
# calib-dir = /reg/d/psdm/mec/mecall113/calib

files = exp=mecall113:run=376
events = 5
#skip-events = 0

modules = cspad_mod.CsPadCalib \
          CSPadPixCoords.CSPadImageProducer:0 \
          CSPadPixCoords.CSPad2x2ImageProducer:1 \
          CSPadPixCoords.CSPad2x2ImageProducer:2 \
          CSPadPixCoords.CSPad2x2ImageProducer:3 \
          CSPadPixCoords.CSPad2x2ImageProducer:4 \
          pyimgalgos.image_crop:0 \
          pyimgalgos.image_save_in_file:0 \
          pyimgalgos.image_save_in_file:1 \
          pyimgalgos.image_save_in_file:2 \
          pyimgalgos.image_save_in_file:3 \
          pyimgalgos.image_save_in_file:4 \
          ImgAlgos.Tahometer

# Calibrate all CSPAD and CSPAD2x2
[cspad_mod.CsPadCalib]
inputKey      =
outputKey     = calibrated_arr
doPedestals   = yes
doPixelStatus = yes
doCommonMode  = yes

[CSPadPixCoords.CSPadImageProducer:0]
#calibDir      = /reg/d/psdm/mec/mecall113/calib
#typeGroupName = CsPad::CalibV1
source         = DetInfo(MecTargetChamber.0:Cspad.0)
key            = calibrated_arr
imgkey         = image0
tiltIsApplied  = false
print_bits     = 3

[CSPadPixCoords.CSPad2x2ImageProducer:1]
#calibDir      = /reg/d/psdm/mec/mecall113/calib
#typeGroupName = CsPad2x2::CalibV1
source         = DetInfo(MecTargetChamber.0:Cspad2x2.1)
inkey          = calibrated_arr
outimgkey      = image1
tiltIsApplied  = true
useWidePixCenter = false
print_bits     = 3

[CSPadPixCoords.CSPad2x2ImageProducer:2]
#calibDir      = /reg/d/psdm/mec/mecall113/calib
#typeGroupName = CsPad2x2::CalibV1
source         = DetInfo(MecTargetChamber.0:Cspad2x2.2)
inkey          = calibrated_arr
outimgkey      = image2
tiltIsApplied  = true
useWidePixCenter = false
print_bits     = 3

[CSPadPixCoords.CSPad2x2ImageProducer:3]

```

```

#calibDir      = /reg/d/psdm/mec/meca1113/calib
#typeGroupName = CsPad2x2::CalibV1
source         = DetInfo(MecTargetChamber.0:Cspad2x2.3)
inkey          = calibrated_arr
outimgkey      = image3
tiltIsApplied  = false
useWidePixCenter = false
print_bits     = 3

[CSPadPixCoords.CSPad2x2ImageProducer:4]
#calibDir      = /reg/d/psdm/mec/meca1113/calib
#typeGroupName = CsPad2x2::CalibV1
source         = DetInfo(MecTargetChamber.0:Cspad2x2.4)
inkey          = calibrated_arr
outimgkey      = image4
tiltIsApplied  = true
useWidePixCenter = false
print_bits     = 3

[pyimgalgos.image_crop:0]
source         = DetInfo(MecTargetChamber.0:Cspad.0)
key_in         = image0
key_out        = image0_cropped
rowmin         = 70
rowmax         = 900
colmin         = 0
colmax         = 825
print_bits     = 255

[pyimgalgos.image_save_in_file:0]
source         = DetInfo(MecTargetChamber.0:Cspad.0)
key_in         = image0_cropped
ofname         = cspad.0.tiff
print_bits     = 5

[pyimgalgos.image_save_in_file:1]
source         = DetInfo(MecTargetChamber.0:Cspad2x2.1)
key_in         = image1
ofname         = cspad2x2.1.tiff
print_bits     = 5

[pyimgalgos.image_save_in_file:2]
source         = DetInfo(MecTargetChamber.0:Cspad2x2.2)
key_in         = image2
ofname         = cspad2x2.2.tiff
print_bits     = 5

[pyimgalgos.image_save_in_file:3]
source         = DetInfo(MecTargetChamber.0:Cspad2x2.3)
key_in         = image3
ofname         = cspad2x2.3.tiff
print_bits     = 5

[pyimgalgos.image_save_in_file:4]
source         = DetInfo(MecTargetChamber.0:Cspad2x2.4)
key_in         = image4
ofname         = cspad2x2.4.tiff
print_bits     = 5

[ImgAlgos.Tahometer]
dn             = 100
print_bits     = 7

```

Command:

```
% psana -c psana-mecall113-r0376-cspad-cspad2x2-tiff.cfg
```

or for other runs:

```
% psana -c psana-mecall113-r0376-cspad-cspad2x2-tiff.cfg exp=mecall113:run=378
```

will run over 5 events from xtc files, applies intensity correction for all CSPAD detectors (if calibration files are provided), produces images and save images for one CSPAD and four CSPAD2x2 detectors in 16-bit tiff files, with names like

cspad.0-r0376-e0000000#-20131110-101607.736131763.tiff

cspad2x2.#-r0376-e0000000#-20131110-101546.244253562.tiff

...

## Examples for mecb3114

Experiment mecb3114 use different detectors in different runs. Not all of them collected data in each run. Then it make sense to produce tiff files for separate detectors.

Configuration file for MecTargetChamber.0:Cspad2x2: [psana-mecb3114-r0008-cspad2x2-tiff.cfg](#)

Configuration file for MecTargetChamber.0:Cspad.0: [psana-mecb3114-r0323-cspad-tiff.cfg](#)

Command to run:

```
% psana -m <file-name>.cfg exp=mecb3114:run=376
```

## Examples for meci0114

Andreas Schropp needs to access cspad image in his custom psana module.

On 2014-03-21 Module ImgAlgos::ExampleDumplmg is add as an example.

### Get and run this example in current release

For releases ana-0.10.14 the updated version of code can be compiled and run from local release directory;

```
ssh -Y psana
cd <your-favorite-directory>
kinit

<next 6 lines can be copied and pasted>
newrel ana-0.10.14 my-release-dir;
cd my-release-dir;
sit_setup;
addpkg ImgAlgos HEAD;
addpkg CSPadPixCoords HEAD;
scons -j 8;

<wait a few minutes until it is compiled>

<pick-up *.cfg files and run them>
psana -c psana-meci0114-r0050-exampleimgdump-cspadcalib.cfg
psana -c psana-meci0114-r0050-exampleimgdump-ndarrcalib.cfg
```

These example print too much, but this is an intentional example. Level of verbosity can be controlled by the `print_bits` parameter for each module.

### Configuration files

- [psana-meci0114-r0050-exampleimgdump-cspadcalib.cfg](#) - use sequence of CsPadCalib, CSPadImageProducer, and ExampleDumplmg modules works with int16\_t data on all stages until ExampleDumplmg

- [psana-meci0114-r0050-exampleimgdump-ndarrcalib.cfg](#) - use sequence of CSPadNDArrProducer, NDArrCalib, CSPadImageProducer, and ExampleDumplmg modules  
CSPadNDArrProducer - creates ndarray<const T,3> for a few generic data types,  
NDArrCalib - may apply more corrections than CsPadCalib.  
ExampleDumplmg - also may process a few data types such as int16\_t, int, float, and double.

## Other option to save image in 16-bit tiff files

Psana can use python modules. A few modules in package pyimgalgs were recently developed to process CSPAD data.

[Psana Module Catalog - Package pyimgalgs](#)

[Psana Module Examples - Examples for Package pyimgalgs](#)

In particular, module `pyimgalgs.image_save_in_file` may save image in 16-bit tiff files.

## Display tiff image

To display image from tiff file use `display` command:

```
display <file-name>.tiff
```

then

1. left mouse click on image
2. click on "Enhance" button
3. click on "Equalize" button

Other option - use `ImageJ` command:

```
/reg/common/package/imagej/ImageJ <file-name>.tiff
```

## Check tiff file

To check tiff file format one may use command:

```
% identify -verbose <file-name>.tiff
```

which prints information about tiff file structure. For example,

```
% identify -verbose cspad.0-r0376-e00000001-20131110-101546.244253562.tiff
Image: cspad.0-r0376-e00000001-20131110-101546.244253562.tiff
Format: TIFF (Tagged Image File Format)
Class: PseudoClass
Geometry: 1750x1750
Type: Grayscale
Endianness: MSB
Colorspace: Gray
Channel depth:
  Gray: 16-bits
Channel statistics:
  Gray:
    Min: 0 (0)
    Max: 16383 (0.249989)
    Mean: 1023.37 (0.0156156)
    Standard deviation: 2132.77 (0.0325439)
Colors: 65536
...
```

## Convert uint16 tiff to int16 tiff

Eric Galtier found the way to change file format record for tiff file using `convert` command:

```
convert input_file.tiff -define quantum:format=signed output_file.tiff
```

Other option for floating point conversion: `-define quantum:format=floating-point`

## Examples for cxi86415

This test is done by request



Loh, Ne-Te writes:

> On a separate note, the online monitoring software (hummingbird, by Benedikt and Filipe) will need py+psana. However, the default psana calibrations/corrections for the 140k back detector seems wonky. You might recall that there was a teleconference where Veitâ's student (Yi-Tien) discusses how the 0-photon peak had a largish negative ADU, despite turning on common mode correction and dark subtraction (psana's implementation). Would you and your team be able to take a look at these as well? If it's a quick fix, then the online monitoring group will have a much easier time (e.g. they won't have to separately do common mode subtraction).

### Standard configuration file

Use psana configuration file [psana-cxidg3-cspad2x2-NDArrAverage.cfg](#) ([download](#)), which can be executed by command:

```
sit_setup
psana -c psana-cxidg3-cspad2x2-NDArrAverage.cfg exp=cxi86415:run=7
```

This job produces 3 files with cspad2x2 shaped arrays for averaged, rms, and maximal intensities.

These arrays can be plotted by the command

```
plims cspad2x2-ndarr-ave-cxi86415-r0007.dat
```

Results for runs 7, 62, and 83 are shown on plots:

These plots show that the dark level is consistent with zero and does not indicate on any offset.

### Loh's configuration file

Use configuration file from Loh's e-mail [orig.cfg](#) ([download](#))

with minor modifications

- add psana parameters in the header, add list of modules
- remove `typeGroupname = CsPad:CalibV1` from `[CSPadPixCoords.CSPad2x2ImageProducer:2]`
- add module `ImgAlgos.ImgSaveInFile:2` just in order to save image:

```
[psana]
files = exp=cxi86415:run=64
events = 10

modules = CSPadPixCoords.CSPad2x2NDArrProducer:2 \
          ImgAlgos.NDArrCalib:2 \
          CSPadPixCoords.CSPad2x2ImageProducer:2 \
          ImgAlgos.ImgSaveInFile:2

<code from e-mail is here>

[ImgAlgos.ImgSaveInFile:2]
source      = DetInfo(CxiDg3.0:Cspad2x2.0)
key         = reconstructed
fname       = img-cspad2x2-dg3
ftype       = txt
#saveAll    = true
print_bits  = 3
eventSave   = 9
```

Run it with command:

```
psana -c orig.cfg
```

which produces file with image for event #9, which can be viewed as

```
plims img-cspad2x2-dg3-cxi86415-r0064-e00000009-20150308-162942.145136207.txt
```

ang brings us an image:  
looks good.

- the same image, but spectrum is shown in the range (-100, 100) ADU. This event

Event #36: looks like left-side sensor is not corrected for common mode fluctuation...

In stead of default common\_mode parameters (1,25,25,100) let us try (1,50,10,100) in the file

```
/reg/d/psdm/CXI/cxi86415/calib/CsPad2x2::CalibV1/CxiDg3.0:Cspad2x2.0/common_mode/9-end.data
```

The same image looks better now:

**Recommendation: try to play with common mode parameters.**

## References

- [CSPAD2x2 Alignment](#)
- [psana - Module Examples](#)
  - [Example for module cspad\\_mod::CsPad2x2Pedestals](#)
  - [Example for module CSPadPixCoords::CSPad2x2ImageProducer](#)
  - [Example for module ImgAlgos::ImgSaveInFile](#)
- [psana - Module Catalog](#)

[NDArrAverage](#) module