On-line Charge to Number-of-Photon Conversion for the JUNGFRAU Detector

document v0.1, November 2016

Contents

1 Introduction **1**
- 1.1 System overview 1
- 1.2 From charge to number-of-photons 2

2 Conversion to number-of-photons **5**
- 2.1 Conversion algorithm 5

3 Pedestal update **7**
- 3.1 Pedestal calculation 7
  - 3.1.1 Pedestal update algorithm for "dark" frames 7
  - 3.1.2 Pedestal update algorithm for "dark" pixels 7

Section 1

Introduction

JUNGFRAU is a two-dimensional pixel detector for photon science applications at X-ray free electron lasers and synchrotron light sources.

JUNGFRAU belongs to the family of charge integrating detectors, which means that the information recorded by the detector is not the number of photons that interact with the sensor, but an electric charge that is proportional to such number of photons for a certain period of time.

Among different characteristics, this detector is able to provide single photon sensitivity and low noise over the full dynamic range. These characteristics are achieved by a dedicated automatic gain switching preamplifier in each pixel, which automatically adjusts its gain to the amount of charge deposited on the pixel.

As a result, two pieces of information are recorded for every pixel: the deposited charge and the employed gain stage (high, medium or low gain).

1.1 System overview

The JUNGFRAU chip comprises 256x256 pixels of 75x75 μm^2 each. These chips are arranged in arrays of 2x4 chips to form modules of 0.5 MPixels and about 4x8 cm^2.

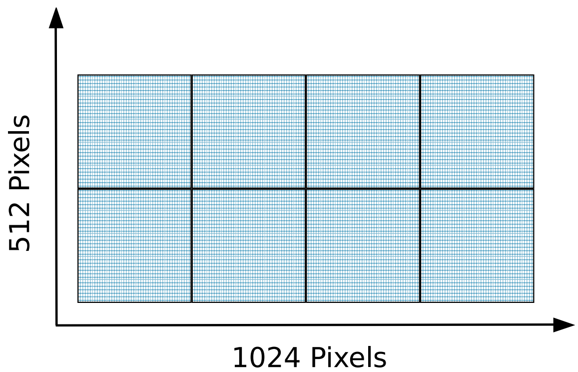

Figure 1.1: JUNGFRAU module

Each JUNGFRAU module connects to a JUNGFRAU Module Control Board (JMCB) that controls the JUNGFRAU chips during the readout phase. Figure 1.2 on the following page shows the a JUNFRAU module connected to the JMCB.

The main component of the JMCB is an Altera Cyclone V Field Programmable Gate Array (FPGA), that acquires the digital output of the chip (employed gain bits) and the digital output of the 32 ADC channels that carry out the conversion of the analog output from the pixels (deposited charge).

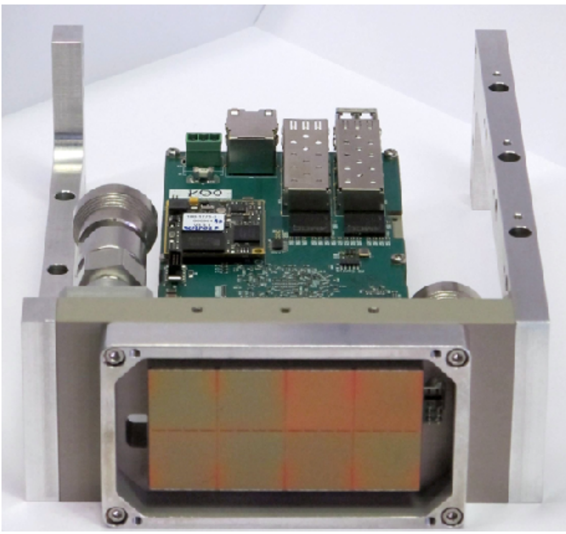

Figure 1.2: JUNGFRAU module without sensor plus JMCB

The whole stream of data (digital and analog) is properly descrambled to obtain a contiguous image that is later streamed out to the user server via 2x10 Gbps ber links. The readout capability of the JMCB is designed to sustain readout rates of 2000 frames/s.

By arranging different of these modules it is possible to build detector systems of several Mpixels. More specifically, multi-module systems of up to 16 Mpixels consisting of 32 JUNGFRAU modules are planned for various experimental setups, with a net output data-stream of 80 GBps.

1.2 From charge to number-of-photons

As we have already mentioned, the data streamed out for every pixel consists of the digitized value of the deposited charge plus the coding of the employed gain stage together in a 16-bit word.

Bit	Symbol	Function
13-0	Charge	Digital value of the deposited charge.
15-14	Gain	Coding of the digital gain stage.

However, data scientist need to process this output stream of data to obtain meaningful information. In fact, what scientists need is a photon image to do their research, that is, a photo with the number of photons recorded per pixel.

Currently, this conversion from charge to number-of-photons is carried out offline by computers, but an on-line processing would have several benefits apart from the reduced CPU usage, for instance:

- Perform further on-line signal processing.
- Carry out on-line data compression to reduce both throughput and storage requirements, since number-of-photon data can be better compressed.

The conversion to number-of-photons is simple but has to be carried out for every single pixel and with parameters that vary from pixel to pixel. These parameters are the actual gain for each gain stage and the offset due to noise and leakage, that also varies from gain stage to gain stage.

All these sets of parameters per pixel can be grouped in 6 maps per JUNGFRAU module: 3 gain maps plus 3 offset (pedestal) maps. When coding this parameters in 16-bit words, all this information adds a total of 6MB of data that needs to be accessed at a rate of 2 kHz, which sets the focus not only on the processing time, but also on the memory bandwidth.

Given the high capabilities in parallel computation, high memory bandwidth and scalability to work with bigger detector systems, GPUs seem to be the best alternative to carry out this task.

Section 2

Conversion to number-of-photons

The result of the on-line conversion to number-of-photons is a "photon image" that should ease scientist's work to do research and potentially give the chance to perform further signal processing on-line.

The parameters needed to perform the conversion are set together in bi-dimensional pixel maps, so that for every pixel in the module $p[x, y]$, its conversion parameters will be stored in a pixel map $m[x, y]$.

A total of 6 pixel maps need to be loaded onto the GPU:

- 3 gain maps, one for each gain level: G_0, G_1, G_2
- 3 pedestal (offset) maps, one for each gain level: P_0, P_1, P_2

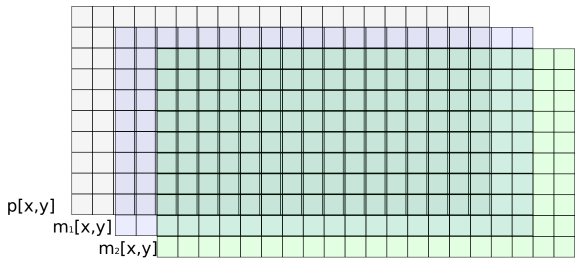

Figure 2.1: Representation of pixel maps

In addition, the beam energy E_{Beam}, which is a global parameter to all the pixels and all the gain levels, needs to be loaded onto the GPU to perform the last conversion step.

2.1 Conversion algorithm

Once this data has been loaded, the algorithm to do the conversion can be summarized in 4 steps.

1. Mask out the digital gain bits to get the raw ADC data and identify the employed gain stage.
2. Offset (pedestal) correction:
   - Corrected charge for gain stage G0: $C_{corr}[x, y] = D_{ADC}[x, y] - P_0[x, y]$
   - Corrected charge for gain stage G1: $C_{corr}[x, y] = P_1[x, y] - D_{ADC}[x, y]$
   - Corrected charge for gain stage G2: $C_{corr}[x, y] = P_2[x, y] - D_{ADC}[x, y]$
3. Calibration.
   - Calibrated energy for gain stage G0: $E_{cal}[x, y] = C_{corr}[x, y] \cdot G_0[x, y]$
   - Calibrated energy for gain stage G1: $E_{cal}[x, y] = C_{corr}[x, y] \cdot G_1[x, y]$
   - Calibrated energy for gain stage G2: $E_{cal}[x, y] = C_{corr}[x, y] \cdot G_2[x, y]$

4. Transform the calibrated energy to number of photons
    - Number-of-photons: $NOP[x, y] = E_{cal}[x, y]/E_{Beam}$

Section 3

Pedestal update

As a difference to the gain map, the pedestal map may have significant drifts due to beam and temperature changes during the operation of the detector. Such drifts need to be taken into account, and therefore, pedestal maps need to be updated every few minutes with a new value.

3.1 Pedestal calculation

New pedestal values needs to be computed when the pixel is not illuminated by any light source. We can differentiate between two operation modes in which this calculation can be carried out: a first operation mode in which the module is completely in absence of light, and a second operation mode in which only some of the pixels are not illuminated.

3.1.1 Pedestal update algorithm for "dark" frames

In this mode, the whole module is not illuminated by any light source. This situation can be forced by interrupting the experiment and closing the shutter of the detector. During this pedestal update stage the new pedestal is computed for all the pixels at the same time and for the different preamplifier gain levels. The average of the last 1000 "dark" frames is used to compute the new pedestal.

3.1.2 Pedestal update algorithm for "dark" pixels

Unlike the previous operation mode in this case it is not required to interrupt the experiment. The calculation of the new pedestal is carried out only in those pixels where no photons are detected. In this case, we calculate the new pedestal value as the exponential moving average of at least 1000 "dark" pixel samples.

$$M_{[x,y]}[n] = M_{[x,y]}[n-1] + D_{ADC} - \frac{M_{[x,y]}[n-1]}{n}$$

$$P_{[x,y]}[n] = \frac{M_{[x,y]}[n]}{n}$$

Where $P_{[x,y]}[n]$ is the computed pedestal for the sample n, and being $P_{[x,y]}[0] = 0$ and $M_{[x,y]}[0] = 0$.