Software for Beam Instrumentation, AWAKE and CLEAR

Eirini Poimenidou @ SY-BI-SW
epoimeni@cern.ch
2 December 2022
Outline

Intro to the BTV System

BI Software: FESA class for the new BTV Card

CLEAR Software: BTVDC CLEAR Operator GUI

Intro to A W AKE Complex

AWAKE Software: BTVSTREAK, BTVAWKSPEC, BPMAWKELEC, BTVAWKSPEC
BTV System
The BTV System consists of:

- a camera (analogue or digital)
- filters
- screens
- a lamp

The BTV system will have 2 changes: the BTV card and the analogue to digital camera. We are currently in the process of replacing the analogue cameras with digital ones.
BTV System

old system

BTCI card + BTCI fesa class

entry point fesa class

new system

new BTCI card + BTVDCE + BTCCTRL

data hardware fesa class for image acquisition fesa class for control of the btv instruments
BI Software
A new BTV card was created by Stephane Burger, that could control up to 2 BTV systems.
It was required to interface that new card by creating 2 new Fesa Classes:

- BTVCTRL to control the BTV system’s instruments
- The entrypoint FESA class to delegate the access to the corresponding system (old or new).
BI Software: new BTV Card

©Stephane Burger
Before developing the FESA classes, there were some other steps that were necessary:

- Add the register map of the board on CCDE (done by Ana and Stephane)
- Create the driver from CCDE and install it on the FEC
- Implement the wrapper of the driver
- Implement the 2 FESA classes
The wrapper is C++ API that the FESA class will use to communicate with the card.

**Advantages of making the wrapper:**

- Simplifies the FESA class
- Better code readability on the FESA class
- Does simple computations that simplify the API
- In case of any changes on the register map on CCDE, the FESA class remains untouched

The wrapper has been implemented.
Future Plans

- Implement the 2 FESA classes, test and deploy them
At CLEAR experiment they want to migrate from analogue to digital cameras and needed a GUI to be able to view their acquired images.

Specific GUI requirements:

- Modify specific property fields (camera and calibration settings)
- Region of Interest (ROI)
- Gaussian distance of a user-picked point from the highest value of the projection
- View Horizontal and Vertical Projections plots
- Toggle acquired image’s palette between BW and Rainbow
- Manage camera (power on/off)
- Save Acquired Image
- Zoom in/out of the image
There was an already existing BTVDC GUI, with the minimum functionalities of visualizing the acquired image data and the projection plots.

This GUI was modified to be able to work on two different modes:

- BTVDCExpert
- BTVDCClearOp

The differentiation of the two modes is using java System Properties.
CLEAR software: BTVDC CLEAR Operator GUI

Image contour and plots
CLEAR software: BTVDC CLEAR Operator GUI

BTVDCExpert

BTVDCClearOp
In BTVDCExpert we can choose a user:
CLEAR software: BTVDC CLEAR Operator GUI

Manage Camera Options vary for the modes
CLEAR software: BTVDC CLEAR Operator GUI

Palette
CLEAR software: BTVDC CLEAR Operator GUI

Gaussian Distance
CLEAR software: BTVDC CLEAR Operator GUI

ROI
Future Plans

- Get feedback from the CLEAR operators and make the according adjustments and optimizations
- Make frontend modifications
AWAKE Software
I have under my responsibility the following AWAKE software (Fesa classes):

- BTVSTREAK
- BTVAWKSPAC
- BPMAWAKE
- BPMAWKELEC
BTVSTREAK Trigger System Debugging

- The BTVSTREAK FESA class had incomplete documentation
- AWAKE operators were trying to do some tests that were unsuccessful
- An extensive debugging session in collaboration with OP took place + communication with Hamamatsu
- The problem was finally understood

Adding a new BTVSTREAK camera

- Add the new device on CCDE
- Configure the FEC it will run on
- Runs with the existing FESA Class
A new electron BPM (Cherenkov BPM) was added in the AWAKE experiment, in order to measure the electrons position while protons were also present.

**Steps to add new BPM:**

- Add new device on CCDE
- Configure the FEC accordingly
- Add logging to NXCALS
- FESA class needs to be modified to support both the old BPMs and the Cherenkov BPMs
- Adjust the BPMAWKELEC GUI to get the Cherenkov BPM data

The GUI was split into two modes one for the old BPM data and one for the Cherenkov data (using Java’s System Properties)
Future Plans

- Add a 3rd STREAK camera, after we get it back from Hamamatsu
- Adjust the BPMAWKELEC FESA class to work with both bpm versions
- Migrate BPMAWKELEC and BPMAWAKE GUIs from JavaFX to Java Swing or PyQt
- add documentation on all FESA classes
Thank you!