ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-TUPV028

THE CONTROL AND ARCHIVING SYSTEM FOR THE GAMMA BEAM PROFILE STATION AT ELI-NP*

G. Chen †, V. Iancu, C. Matei, F. Ramirez, G. Turturica
Extreme Light Infrastructure – Nuclear Physics, IFIN-HH, Magurele-Bucharest, Romania

Abstract

The Variable Energy Gamma (VEGA) System of Extreme Light Infrastructure - Nuclear Physics (ELI-NP) is based on the Inverse Compton Scattering of laser light on relativistic electron bunches provided by a warm radio-frequency accelerator. The system will deliver quasi-monochromatic gamma-ray beams with a high spectral density and a high degree of linear polarization. The Beam Profile Station, which will be used for finer target alignment and spatial characterization of the gamma-ray beam, is one of the diagnostics stations under implementation at ELI-NP.

An EPICS Control and Archiving System (CAS) has been developed for the Beam Profile Station at ELI-NP. This paper describes the design and the implementation of the EPICS CAS for the Beam Profile Station, including the device modular integration of the low-level IOCs for the CCD camera Trius-SX674 and Mclennan PM600 Stepper Motor Controller, the design of the high-level GUI for real-time image acquisition and motion control, as well as the configuration of the archiving system for browsing the historic images and parameters.

INTRODUCTION

The Variable Energy Gamma (VEGA) System of Extreme Light Infrastructure - Nuclear Physics (ELI-NP) will produce intense gamma-ray beams with a spectral density higher than 0.5 x 10^4 photons/eV/s, a relative energy bandwidth better than 0.5%, high degree of linear polarization at more than 95%, and energy continuously variable from 1 MeV up to 19.5 MeV based on the laser Compton backscattering of laser photons off a relativistic electron beam.

The Gamma Diagnostics Stations are designed and under implementation to measure and to monitor the gamma beam diagnostics features [1]. To optimize the operation of the ELI-NP VEGA and its use for experiments, it is necessary to have the proper means to accurately predict the spatial, spectral and temporal characteristics of the gamma beam. The ELI-NP Gamma Diagnostics Stations are dealing with the equipment and techniques meant to optimize the gamma beam in order to make it available for user experiments within required parameters.

As a part of the Gamma Diagnostics Stations, the Gamma Beam Profile Station utilizes a CCD camera to collect the light produced in a scintillator placed in the beam by means of a mirror. The system is designed to be mobile and could be moved to different experimental areas.

The camera will be placed outside vacuum in a special mount, which ensures position reproducibility and easy removal. The scintillator and the mirror will be housed in a light-tight box inside the beam transport pipe. To ensure position reproducibility of the scintillator, a motor controller will be attached to the scintillator.

The CCD camera Trius-SX674 and the Stepper Motor Controller Mclennan PM600 have been chosen to be integrated into the CS of the Gamma Beam Profile Station.

Besides, the archiving system is required to store the PV samples periodically into an RDB.

SYSTEM DESIGN AND CONFIGURATION

The EPICS [2] based CAS for all the diagnostics stations will be designed and implemented by ELI-NP, to provide the machine information connection with the VEGA CS, to collect data from devices in the diagnostics stations, to monitor status of the devices, and to provide the High-Level Software (HLS) for the Gamma Beam Diagnostics.

General Architecture Model

The architecture of the CAS is structured as the standard three-tier structure:

User interface. These are graphical and non-graphical user interfaces. Most or all of these will be in the control room.

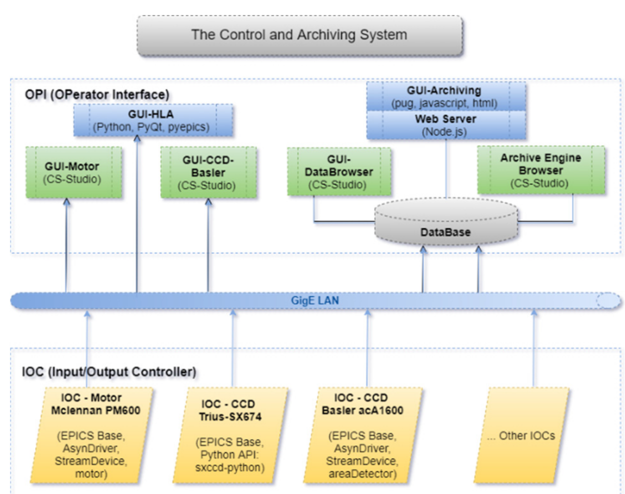

Figure 1: The software architecture of the Control and Archiving System.

Central services. Central services need to be run continuously irrespective of user activities, e.g., for archiving process variables' values, monitoring alarm states, etc.

Equipment interfaces. This tier is responsible for interaction with equipment and devices. It provides an abstract

* The work is supported by ELI-NP Project (http://www.eli-np.ro/)
† guangling.chen@eli-np.ro

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-TUPV028

representation of equipment to higher layers through which the equipment can be monitored and controlled. The elementary entity on this level is called an Input Output Controller (IOC).

The gigabit Ethernet (GigE) Local Area Network (LAN) allows the communication between the IOCs, Archivers, and OPIs.

The software architecture of the control and archiving system is shown as in Fig. 1. The EPICS IOC software package are dealing with the low-level input and output controls to the devices, while the GUI (Graphical User Interface) for HLA (High-Level Application) and the GUI for Archiving system are taking care of the High-level user operations.

Design of the Control System

The control of the Beam Profile Station for Gamma Beam diagnostics includes one Mclennan PM600 motor controller and two Trius-SX674 CCD cameras.

IOC Design for the Motor Controller The Mclennan PM600 is a powerful single axis motion controller which has been selected for the beam profile station to move the scintillators in and out of the gamma beam.

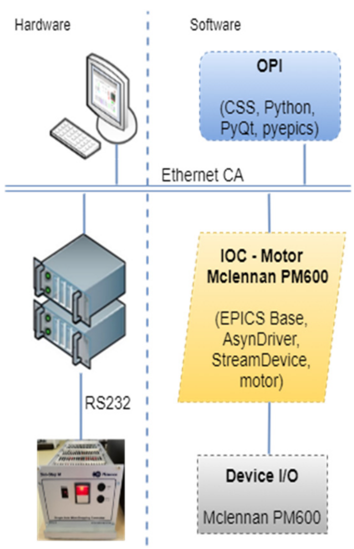

Figure 2: The hardware and software structure for the motion control.

The motor controller Mclennan PM600 is connected to the PC via a RS232 serial cable, and to the linear bellows drive via the dedicated Motor and Limit/Datum cables.

The diagram showing above (Fig. 2) is the hardware and software structure for the motion control which includes a Mclennan PM600 controller, connected via RS232 serial cable to an IOC running EPICS and motor related software. The IOC is connected to the operator's workstations via the Ethernet network.

Mclennan PM600 controller is supported by the EPICS device support module motor record. By using the motor record, the IOC can read motor position from the controller's readback register, can get readback information while the motor is moving, and can trigger a forward link when a complete motion is finished.

IOC Design for the CCD Cameras The image acquisition of the Beam Profile Station for Gamma Beam diagnostics includes two Trius-SX674 CCD cameras which are connected to the PC via USB cables.

The goal is to provide a functional EPICS application using the required Python device support and device drivers for the Trius-SX674 CCD camera.

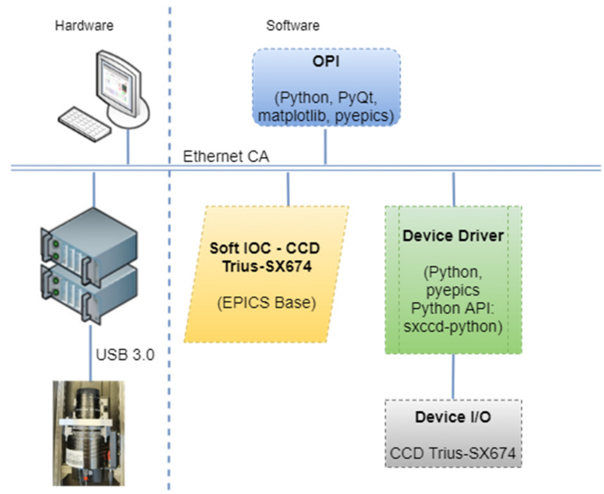

Figure 3: The hardware and software structure for the CCD camera.

The Hardware and Software structure design for the Camera (Fig. 3) includes a Trius-SX674 CCD camera, connected via USB 3.0 to an IOC running EPICS and camera related software. The IOC is connected to the operator's workstations via the Ethernet network.

Trius-SX674 CCD camera is not a standard genicam camera, so the most common used EPICS device support modules aravisGigE and areaDetector cannot be used as the Linux camera driver. We use Python API sxccd-python as device support module, and implement a driver wrapper based on this device support module. Besides, a soft IOC will deal with the EPICS databases for the CCD camera control.

Design of the OPI (GUI) Typically, areaDetector plugins are used to save images and to provide arrays used for displaying images acquired by cameras using the GUI. Since the Trius-SX674 CCD camera is not supported by EPICS areaDetector, the GUI need to be implemented individually.

The EPICS device support module motor record provides a CSS BOY-based GUI to perform the device-layer control and troubleshooting for the Mclennan PM600 controller. Moreover, a Python and PyQt based GUI for the motion control and the image acquisition for the Beam Profile Station will be implemented.

The goal is to provide a functional EPICS based GUI to control the Mclennan PM600 controller and to acquire image from the Trius-SX674 CCD camera.

Design of the Archiving System

The architecture of the archiving system (as shown in Fig. 4) consists of an archive engine, the MySQL RDB, a web server and the web-based GUI.

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-TUPV028

The Archive Engine samples Process Variable (PV) data, time stamp etc., from EPICS IOCs via Channel Access (CA), and places them in a Relational Database. Users can then access the historic data from the database as well as the live data from the PV channels using the CSS Data Browser. Config.xml and settings.ini are the two basic configuration files needed to configure the database by using the provided ArchiveConfigureTool. Besides, PhpMyAdmin is used as the tool for the database maintenance and configuration.

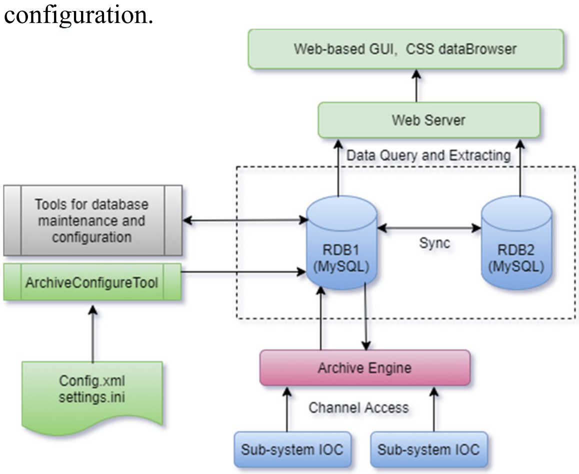

Figure 4: The structure and data flow of the archiving system.

IMPLEMENTATION

Implementation of the IOCs

To integrate the Trius-SX674 CCD cameras, we use Python API sxccd-python [3] as the device support module. Based on the sxccd-python module, a driver wrapper has been implemented for the communication with the soft IOC which deals with the EPICS databases for the CCD camera control.

The IOC for the CCD camera starts with an initialize module to deal with the USB connection and soft IOC initialization. The read-out loop detects when the image acquisition toggle is on, then the image will be captured and the waveform data will be put into the PV of image.

Mclennan PM600 controller is supported by the EPICS device support module motor record [4]. To integrate the PM600 controller, we use the version of the motor record motorR7-1, as well as other related modules in EPICS SynApps5-8.

The IOC for the motor controller is implemented following the standard steps for an EPICS motor IOC. The IOC start file st.cmd is configured to clarify the support module, to establish the serial connection, and to generate the database by the file motor.substitutions and motorUtil.db

Implementation of the OPIs

The operator interface, the GUI is implemented using python3 and PyQt5. Some other python modules, such as numpy, pyepics, threading, and matplotlib etc., are also imported for the development.

A screenshot of the GUI has been shown in Fig. 5.

The left part of the GUI is dealing with the following motion controls:

- Turn on/turn off the motor remotely.
- Move the motor for a certain distance.
- Set the velocity.
- Update the feedback in real-time.
- Calibrate the motor to the home position.

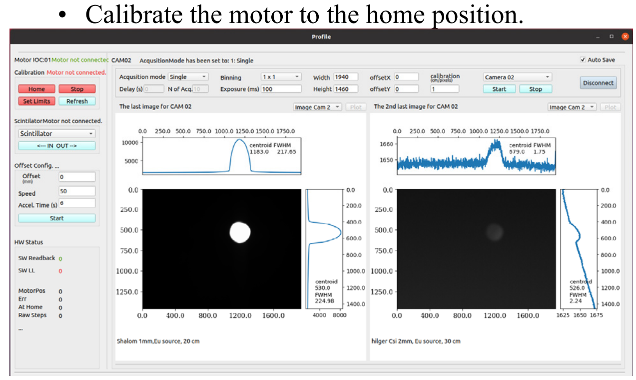

Figure 5: The GUI for the control of the motor and CCD.

The rest part of the GUI is able to deal with the image capturing controls as following:

- Turn on/turn off the camera remotely, to acquire image remotely from the client user interface.
- Set the width, height, exposure time, offset, and acquisition mode (single/continuous).
- Monitoring the status of the camera, to set the exposure time for the camera.
- Provide real-time processing of the images, e. g., profile for X and Y axis, centroid, and FWHM.
- Save images in different formats for the image analysis.
- Save the waveform data of the images.
- Save the parameters for the image capturing.

Implementation of the Archiving System

The EPICS Archive Engine Service in CS-Studio has been chosen as the software tool for the archiving system development. To configure the archive engine, Phoebus [5], the latest version of CS-Studio has been chosen as the archive engine configuration tool to read samples from PVs and write them to the RDB. The data browser has been chosen also to browse the historical or monitoring data from the RDB.

The Archive Engine has been configured using the ArchiveConfigTool. PhpMyAdmin has been used as the database management tool.

Besides, we stored on the database server all the images which are acquired from the Beam Profile System, as well as the image data, and the image related parameter files in the file system.

Implementation of the Web Server and Web-Based GUI

Figure 6 shows the scheme used to implement the web-based system using the MVC (model-view-controller) pattern. With this implementation, the system can serve requests in two ways: by using the embedded views and also via API's fetching and sending raw data.

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-TUPV028

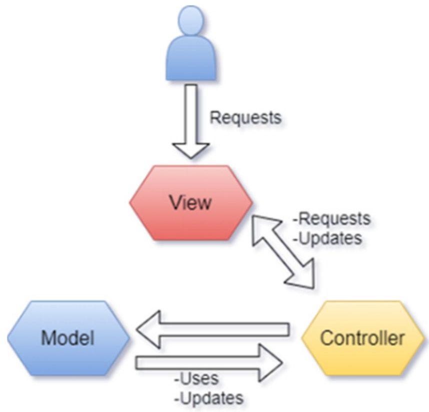

Figure 6: MVC representation.

The web server and the web-based GUI for the archiving system is implemented based on Node.js, Express, MySQL, Sequelize, Javascript, Pug, HTML, and VS code.

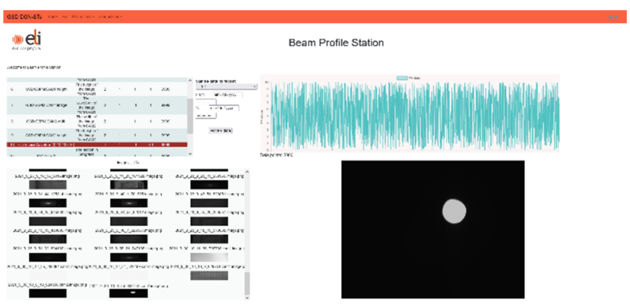

Figure 7: EPICS-based archive application GUI.

Figure 7 shows the results of HTTP requests using the embedded view (GUI). By using the web-based GUI, users can connect to the archive RDB which stores the data fetched by the archive engine. Users can also browse the data by choosing a PV from the PVs-list and then plot the data immediately. Moreover, the GUI allows users to choose an image from the images-list and to display it immediately with the parameters associated to the image.

CONCLUSION

The EPICS based control and archiving system has been implemented for the Beam Profile Station at ELI-NP. The devices have been integrated into the control system, and the archive engine has been configured for the archiving system.

The IOC for the CCD camera has been configured based on sxccd-python. A driver wrapper has been implemented for the communication with the soft IOC of the camera. The IOC for Mclennan PM600 controller has been configured using the version of the motor record motorR7-1, as well as other related modules in SynApps5-8. The GUI for the control has been implemented using python3 and PyQt5.

The archive engine has been configured using CSS Phoebus. The web server and the web-based GUI have been implemented using the MVC pattern.

This implementation of the control and archiving system for Beam Profile Station has been proved as a professional programming structure that helps in maintenance and readability of the whole system, moreover, it is reliable enough for scaling up in the future implementation for other diagnostics stations such as Beam Energy Station and Beam Polarization Station at ELI-NP.

ACKNOWLEDGEMENT

The work was supported by the Project Extreme Light Infrastructure - Nuclear Physics (ELI-NP). The author would like to thank Dr. Marian TOMA for his valuable comments and suggestions.

REFERENCES

[1] H.R. Weller *et al.*, "Gamma Beam Delivery and Diagnostics", *Romanian Reports in Physics*, vol. 68, pp. S447–S481, 2016. http://www.rrp.infim.ro/IP/S9.pdf

[2] EPICS, http://www.aps.anl.gov/epics/

[3] Python API for Startlight Xpress CCD (sxccd), https://github.com/leeneil/sxccd-python

[4] EPICS: Motor Record and Device/Driver support, https://epics.anl.gov/bcda/synApps/motor/

[5] CS-Studio Phoebus, https://controlssoftware.sns.ornl.gov/css_phoebus/