ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA031

SOFTWARE AND HARDWARE DESIGN FOR CONTROLS INFRASTRUCTURE AT SIRIUS LIGHT SOURCE

J. G. R. S. Franco†, P. H. Nallin, E. P. Coelho, C. F. Carneiro, R. C. Ito, R. W. Polli, V. S. Pereira, A. R. D. Rodrigues
Brazilian Synchrotron Light Laboratory, Campinas, Brazil

Abstract

Sirius is a 3-GeV synchrotron light source under construction in Brazil. Assembly of its accelerators began on March 2018, when the first parts of the linear accelerator were taken out of their boxes and installed. The booster synchrotron installation has already been completed and its subsystems are currently under commissioning, while assembly of storage ring components takes place in parallel. The Controls System of Sirius accelerators, based on EPICS, plays an important role in the machine commissioning and installations and improvements have been continuously achieved. This work describes all the IT infrastructure underlying the controls system, hardware developments, software architecture and support applications. Future plans are also presented.

INTRODUCTION

Sirius accelerator, the new 4th generation Brazilian synchrotron light source, has been under construction since 2014, along with the full operation of the current facility, UVX. Engineering assemblies and installation, which include controls subsystems, have started in early 2018.

Sirius Controls System conceptual design aimed to be scalable, distributed and easy to maintain [1]. Based on these principles and on open-source solutions, hardware, infrastructure and software concepts have been designed, implemented and installed on Sirius site. Thus, the integration of a large variety of equipment has been achieved.

SYSTEM ARCHITECTURE

Network Topology

Based on EPICS framework, controls systems topology is composed of two Supermicro servers (with dual Intel Xeon E5-2695 processors, 8x 64GB DDR4, 16x 8TB HDD for data storage and 2x 480GB SSD for operational system) and two core switches with 48 SFP+ ports each (Figure 1). Lower level switches, having 4 10GBase-SR ports, 24 1000Base-T ports and PoE+ driver capability, are connected to the main ones in a star topology. Both switch models have redundant power supply, fans and management modules. A ring interconnection will be available in the future in order to have network redundancy.

Figure 1: Core switch Aruba 5412R.

Lower level switches are placed in 24 different areas, such as Sectors Instrumentation Areas, RF Room, Transport Line Area, LINAC Area and Power Supplies Room, and port numbers are extended with 10/100/1000Base-T equipment if needed.

HARDWARE SOLUTIONS

Controls system distributed nodes are based on the open-hardware single board computer (SBC) Beaglebone Black [2], running an embedded Debian distribution operational system.

The inexpensive embedded system has been successfully in operation in LNLS current facility (UVX) since 2016 [3], as a replacement for outdated SBCs and also as a test bench for Sirius nodes.

Main hardware projects aim the extended use of Beaglebones.

SERIALxxCON

Designed to be the main Beaglebone Black baseboard, with over two hundred units distributed on controls cabinets (Figure 2), they are used to interface with several equipment through either RS-485 or RS-232 serial communication. Standard FTDI module is available and also a high-performance serial interface, developed with Beaglebone's embedded Real-Time Processors (PRUs) [4], reaching pre-defined baudrates up to 15 Mbps. This is a requirement for power supplies communication, once the amount of data to be transferred between systems is elevated.

The board also have inputs for both optical and electrical timing system for synchronized operations using high-performance interface, which can be set up in three different main modes:

- Single sequence curve execution, sending setpoint commands;
- Continuous curve execution;
- Single broadcast command.

† guilherme.franco@lnls.br

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA031

Power supplies are intended to communicate via PRU interface while other systems, such as vacuum electronics, RF and radiation probes, communicates at standard baudrates and can be controlled with both PRU or FTDI. SERIALxxCON boards also have a generic SPI bus (named SPIxxCON) for general purpose board expansion.

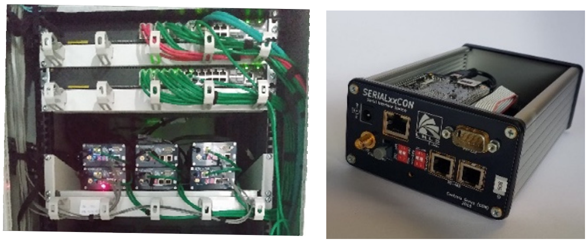

Figure 2: SERIALxxCON hardware interface, installed in Controls cabinet and single unit.

SPIxCONV

This project, based on SPIxxCON bus, consists on an extension board with a 18-bit analog input and output within a range of 20 V (±10 V) and 32 digital pins that can be configured either as inputs or outputs. This system interfaces with pulsed power electronics hardware in order to control the output voltage for pulsed magnets as well as read and write status commands to their power supplies.

Board layout is critical for high accuracy and several tests were performed during developments. In Figure 3 histograms for three different inputs shows system accuracy for 10000 reading samples, after a calibrating process.

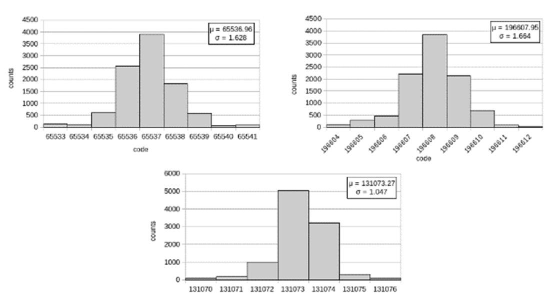

Figure 3: Histograms for 10000 samples at -5 V (code 65536), +5 V (196607) and 0 V (code 131072).

Voltage references for analog-to-digital converter (ADC) characterization were obtained from the digital-to-analog converter (DAC), present on the same board. DAC was first calibrated with a 7.5-digit multimeter.

This board is placed inside Pulsed Power Electronics control crates. 20 units are currently in use (Figure 4) and other 20 will be available for replacements.

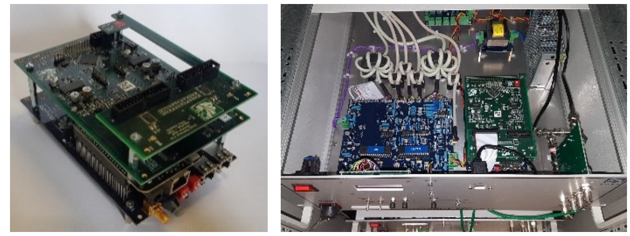

Figure 4: SPIxCONV final layout and assembly.

Multi-purpose Counting System

Also based on Beaglebone Blacks and their embedded Programmable Real-Time Units, a multi-purpose 8-channel hardware designed for counting pulses is available (Figure 5). Focused on integration with Bergoz Beam Loss Monitors (BLMs) and in-house developed gamma detectors, the system provides +24V and ±5V for powering sensors.

Maximum count rate achieved is 14.29 MHz when using up to 2 channels (one channel per PRU) simultaneously and 10 MHz if all channels are enabled (4 channels per PRU).

There will be around 80 units installed in Sirius tunnel, under the girders. As an alternative to cable and infrastructure reduction, this is the first controls hardware designed to be powered on Power over Ethernet (PoE), where both electric power and data are transmitted over the same Ethernet cable.

Temperature Monitoring

Temperature measurements is an important diagnostic for accelerator systems, especially for vacuum components. Thus, a hardware acquisition solution was developed, based on a PIC microcontroller (Figure 5). This module, named MBTemp, reads up to 8 four-wire Pt100 temperature sensors in a range from 0 °C to 425 °C. All channels readings are performed in 550 ms. Values are filtered through an exponential moving average digital filter and they can be read out via RS-485 interface.

For vacuum chambers temperature monitoring, 175 MBTemp modules and around 800 Pt100 sensors will be installed along accelerators. MBTemps will be grouped in multi-drop serial networks.

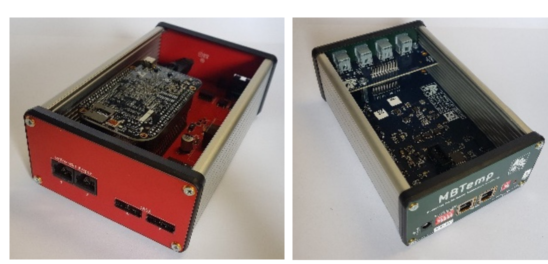

Figure 5: Multi-purpose counting system unit and MBTemp, which are being installed on Sirius accelerators.

Controls Cabinets Monitoring

A platform to monitor controls cabinets environment is under development. It is a solution to acquire information such as power quality, temperature, humidity, air flow, and

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA031

prevent system failures or, perhaps, correlate those data to them.

SOFTWARE DEVELOPMENTS

Server and Networking

Networking infrastructure is also shared to subsystems from Diagnostics Group (screen monitors, timing, motion controls and BPMs, for example) and Interlock system (personal protection – PPS, machine protection – MPS and user interface – IHM). In order to manage and isolate subnets, VLANs were created. So far, 58 VLANs are configured in controls network (layer 2), with data flow restricted to areas or equipment. Some devices do not work properly if an intense data traffic from other subsystems reach their interface.

Controls server run on CentOS 7 and will be upgraded to CentOS 8 soon. The chosen Linux distribution is due to the available documentation and support provided for commercial Red Hat Enterprise Linux, once CentOS is a fork of open-source Red Hat codes, which is highly stable and community-supported. An application running on servers is GlusterFS, which is a scalable network filesystem. This tool provides a good data storage, once selected hard disks are software-managed.

Both servers are powered from an external UPS. A monitoring interface is attached to them, making it possible to know whether it is operating on battery or even if the battery is low. When these events occur, an email message is sent to some collaborators. This feature can be used in the future to ensure controlled shutdown.

EPICS Archiver Appliance

Archiver system run on controls system servers and it is divided into two docker containers that interact with each other. One for all Java EE containers and another for database.

About 34.000 process variables are assigned to be monitored/scanned, which generate around 40 GB of data per day in the archiving system. This interface is running properly, however it is possible to improve its configuration.

It is also planned to change archiver database from MySQL to its forked open-source server, MariaDB [5]. A test version is already running simultaneously in Controls server.

Another tool related to archiving system is its Web Viewer. A JavaScript web application was developed in-house, using chart.js library, inspired on SLAC version, with no graphic frameworks.

Some improvements have already been achieved. In Figure 6 it is possible to view simultaneously 51 plotted PVs representing vacuum measurements in Sirius booster sector, using semi-log axis. Exporting data into .xlsx files is also available and, if needed, .csv and .ods can be added to the interface.

Figure 6: Archiver Web Viewer, with 51 data series.

Web interface is constantly upgraded, with new features requested by users as subsystems are under integration. A beta release is first available. After a period for tests and acceptance, this release is officially integrated to Sirius Archiver Appliance container.

Alarm System

Concerning a controls system alarm tool, some features are interesting to have:

- Occurrences history
- Event acknowledgements
- Option to keep alarm condition after a trigger condition deactivation
- Access control
- Availability of notification options

BEAST, which has a Control System Studio (CSS) front-end for alarms monitoring, is running on controls servers and currently in use. In the future, it is intended to gradually migrate all CS-Studio applications to Phoebus, which does not support BEAST. As an alternative, it will be possible to adopt the new community-supported alarming tool. Also, as a complement to the current BEAST, some new studies and developments concerning Zabbix software tool [6] are in progress, which will be described in a further section.

Logbook Application

Olog tool, a logbook service for accelerator operations, is maintained in controls servers and used to log any important information about subsystem testing and commissioning. Several machine engineering collaborators have access to it. This tool is also intended to be used at Sirius scientific and beamlines department.

Containerization

Docker engine is largely used and container orchestration is performed with Docker Swarm tool, which manages containers deployment and scaling.

Examples considering deployment with Docker Swarm:

- Archiver
- Olog
- EPICS IOCs
- Monitoring tools (Zabbix, BEAST)
- Web applications

17th Int. Conf. on Acc. and Large Exp. Physics Control Systems ICALEPCS2019, New York, NY, USA JACoW Publishing
ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA031

Controls Group maintain a Docker Hub organization (http://hub.docker.com/u/lnlscon) where base container images are available.

EPICS IOCs

EPICS IOCs for most devices are deployed on dedicated docker swarm containers running in controls servers. They are all based on StreamDevice tool and, if needed, a Python bridge to equipment interface is written. For example, to control an equipment using National Instruments VISA library.

Virtual machines may also be used to deploy EPICS IOCs, although it is not usual. It was done in the case of commercial CC Regatron power supplies, due to driver limitations on Ethernet communication.

RF IOCs run on their own server and are mostly based on StreamDevice module, except for RF interlock, which interfaces with an Allen Bradley PLC (ControlLogix 5000) via EtherIP module, and low level RF, using vanilla EPICS to interface with NUTAQ's API.

Controls System Screens

First supervisory system screens for Sirius were built with Control System Studio, as it is a simple and intuitive tool, heavily based on Eclipse. Recently, PyDM has also been adopted by developers as a platform for controls system user interfaces.

Screens for vacuum (ion pumps and vacuum gauges), pulsed power electronics, procServ manager, radiation probes, RF systems (low-level RF and power sensor) and CC Regatron power supplies are in use and frequently upgraded as accelerator's sub-systems are currently under commissioning. A general screen is initially launched (Figure 7), where buttons to specific ones are available.

Accelerator physicists are responsible for a High-Level Applications system (HLA) [7] which includes screens for controlling conventional in-house developed power supplies, designed with PyDM.

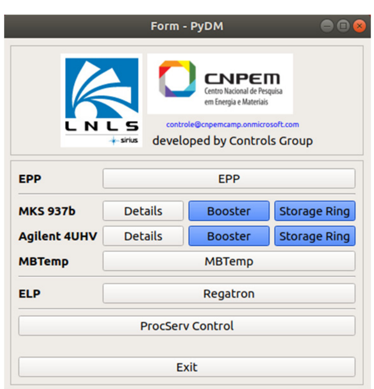

Figure 7: Initial screen for some controls applications.

Beaglebone Black Managing

Controls system distributed nodes, Beaglebone Blacks mounted on SERIALxxCON boards, have installed on its standard image two main services, BBB-Daemon and BBB-Function, which are very important for system modularity, start-up and remote monitoring and configuring.

BBB-Function, as its name supposes, is responsible for detecting in which equipment the node is attached to. It is done by hardware configuration checks and serial communication requests. The information about device type is stored in a JSON file as well as the IDs of each one. Once device is detected, all applications needed to interface with the equipment is launched automatically. Before being launched, each application have its project folder synchronized with server content in order to guarantee latest code versions. This is also done for BBB-Function application, which restarts if an update is found.

BBB-Daemon, on the other hand, is a service that keeps sending UDP messages to the server to show the node is waken-up. The message content is composed of Beaglebone's hostname and information stored on JSON file provided by BBB-Function. Controls server process incoming UDPs messages and stores them in a Redis database. BBB-Daemon also accepts remote commands for rebooting, hostname and IP changing (Figure 8).

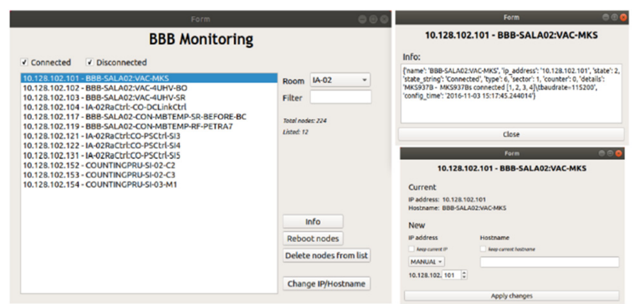

Figure 8: Screen for Beaglebone Black remote monitoring and configuring.

Monitoring Application

A recently added application to controls system infrastructure is Zabbix, an open-source software tool for monitoring network infrastructure and devices, including servers, switches, virtual machines, workstations and many other equipment (Figure 9). Zabbix server is running on a docker swarm container and Zabbix agent has been installed in connected devices. Beaglebone Black minimal image comes with agent installed.

Some features that are current available or in plans to be implemented by Controls Group:

- Register multiple equipment and devices
- Configure a template for each device category, monitoring different items (CPU usage, storage left, if a service is active and running, etc.)
- Create dashboards, in order to show relevant information
- Add notification services
- Add general-purpose scripts
- Integration with EPICS

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA031

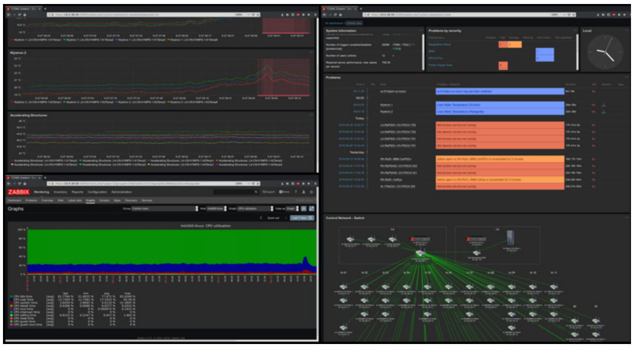

Figure 9: Zabbix user web interface, with different customized dashboards.

Even though it is not mandatory for accelerators operation, it has been demonstrating to be a useful tool for identifying, solve and, perhaps, predict difficulties in the future.

CONCLUSION

Although there are still general infrastructure, assemblies and integration to be concluded, as Sirius accelerators are not completely finished yet, new improvements and implementations are in progress simultaneously to installation activities.

The features currently available at Controls System allow a reliable operation, commissioning and optimization for all machine subsystems, which are usually performed from Controls Room. Having such a system is an important step and essential to Sirius commissioning progress.

REFERENCES

[1] J. P. S. Martins *et al.*, "Sirius Control System: Design, Implementation Strategy and Measured Performance", in *Proc. ICALEPCS'15*, Melbourne, Australia, Oct. 2015, pp. 456-459. doi:10.18429/JACoW-ICALEPCS2015-MOPGF158

[2] http://beagleboard.org/bone

[3] S. Lescano *et al.*, "UVX Control System: An Approach with Beaglebone Black", in *Proc. PCaPAC'16*, Campinas, Brazil, Oct. 2016, pp. 91-93. doi:10.18429/JACoW-PCAPAC2016-THPOPRP003

[4] P. H. Nallin *et al.*, "Exploring Embedded Systems' Dedicated Cores for Real-Time Applications", presented at the ICALEPCS'19, New York, NY, USA, Oct. 2019, paper WEMPR003, this conference.

[5] https://mariadb.org/

[6] https://www.zabbix.com/

[7] X.R. Resende *et al.*, "High-Level Applications for the Sirius Accelerator Control System", in *Proc.IPAC'19,* Melbourne, Australia, May 2019, pp. 2462- 2465. doi.org/10.18429/JACoW-IPAC2019-WEPGW003