STATUS OF THE EPICS-BASED CONTROL AND INTERLOCK SYSTEM OF THE BELLE II PXD

M. Ritzert*, Heidelberg University, Germany†

Abstract

The Belle II e+e- collider experiment at KEK (Tsukuba, Japan) will include a new pixelated detector (PXD) based on DEPFET technology, providing the two innermost layers around the beampipe. This detector requires a complex control and readout infrastructure consisting of several on-sensor ASICs and remote FPGA boards. We present the architecture and EPICS-based implementation of the control, alarm, and interlock systems and their interconnectivity to other legacy/heterogeneous subsystems. The interface to the NSM2-based Belle II run-control to orchestrate the PXD startup sequence is also presented. An installation of CSS is used to implement the user interface. The alarm system uses CSS/BEAST, and is designed to robustly minimize spurious alarms. The interlock system consists of two main parts: a hardware-based system that triggers on adverse environmental (temperature, humidity, radiation) conditions, and a software-based system. Strict monitoring including the use of heartbeats ensures permanent protection and fast reaction times. Especially the power supply system is permanently monitored for malfunctions, and all user inputs are verified before they are sent to the hardware. The control system is embedded into a larger slow-control landscape that also incorporates archiving, logging, and reporting in a uniform workflow for the ease of daily operation.

CONTROL SYSTEM

The PXD control system is built on the control framework EPICS[1], which is widely used in high energy physics systems. The Eclipse-based suite Control System Studio (CSS)[2] is being used as the operator interface framework.

The EPICS system will be deployed in a dedicated physical network segment only accessible via a gateway using the Channel Access (CA) protocol [1] for load reduction on the EPICS Input/Output Controllers (IOCs)[3] and access logging. Access control lists (ACLs) enforced within the IOCs will be used to implement various access levels (read only, user, administrator).

In order to achieve a consistent view on operating conditions during the experimental run and for analysis after a failure, a major amount of identified key values have to be recorded and archived centrally in a database. This is to ensure that in normal as well as abnormal situations a consistent and synchronized view on the data and whole system is possible.

The system will be deployed on x86_64 servers running Scientific Linux as the operating system. To allow for fast, repeatable installation and controlled updates, the required software is packaged as RPMs.

The following list details major devices of the system and their means of control via EPICS:

- The data handling engine (DHE) [2, 3] receives the global trigger and timing signals and distributes them to the sensors. The internal FPGAs connect to the outside world via the IPbus protocol. IPbus is based on UDP/IP and can be implemented purely in HDL. An IOC implementing the IPbus protocol in the DHE-specific way has been written.
- The actual DEPFET detectors are controlled and read out via a number of ASICs that are accessible via JTAG from the DHE. The IOC for the DHE and the firmware running on the DHE include code to access the JTAG chain. Actual commands are again transmitted to the DHE via IPbus.
- The power supplies communicate with the control PC via the CHROMOSOME protocol, a fault-tolerant middleware [4] that also includes a heartbeat to quickly detect a lost connection on both sides. Another custom IOC has been written to interface these devices.
- The online selector nodes (ONSEN) receive tracking information from other subdetectors via the data concentrator (DATCON) and high level trigger (HLT) and uses it to filter out events not in proximity to a projected particle crossing in the PXD. The ONSEN devices are controlled with an IOC running on the embedded Power PC in the FPGA via a memory mapped interface.
- The ATCA and µTCA crates housing the ONSEN, DHH and DATCON systems are accessible via IPMI.
- Belle II run control communicates via the NSM2 protocol developed at KEK [5, 6]. A bidirectional gateway also developed at KEK is used to interface with the EPICS world.
- The environmental conditions inside the detector volume are monitored via Bragg grated fibres read out with an optical interrogator from Micron Optics. An IOC that reads data from the device at high frequency (several 100 Hz) is under development. The data will be made available to EPICS either as a single data points down-sampled from the raw device rate, or put through an FFT transformation to obtain a frequency spectrum, which is then available as a waveform record. The latter mode is especially useful to analyze the minute movements of the detector.
- The actual slow control of the IBBelle CO_2 cooling plant is implemented in a PLC located within the plant itself. The Modbus protocol is used to monitor its

* michael.ritzert@ziti.uni-heidelberg.de
† For the DEPFET collaboration.

[1] http://www.aps.anl.gov/epics/
[2] http://controlsystemstudio.org
[3] http://www.aps.anl.gov/epics/base/R3-14/12-docs/AppDevGuide.pdf

ISBN 978-3-95450-148-9

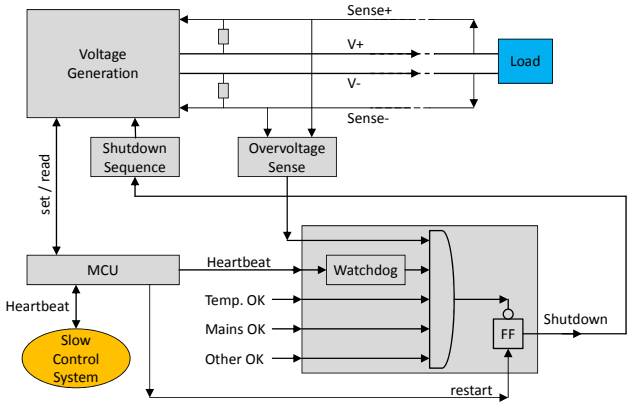

Figure 1: Protection system of the power supplies. All items in gray are inside the power supply.

operation, set the desired operating point, and provide expert access.

A common naming scheme for all PVs ensures that users can quickly decode the meaning of a PV name in any context.

ALARMS AND INTERLOCKS

The Best Ever Alarm System Toolkit (BEAST) [7] integrated into a custom build of CSS is used to visualize the alarms for the operators.

Two alarm levels, major and minor, are defined as "(part of the) detector in-operational, or immediate danger to the detector" and "take action to avoid a major alarm" respectively. A major alarm will typically cancel the current run. Many major alarms also trigger an interlock.

Several soft-interlocks have been applied in order to ensure immediate action during detected potentially dangerous conditions or malfunctions, or to prevent these situations from arising in the first place. Software interlocks are accompanied by hard-wired interlocks that are in any case the last-resort to prevent hardware damages.

Power Supplies

As the only supplier of power into the detector volume, positive control of the power supply (PS) system at all times is of utmost priority. The PS unit is controlled by an internal microcontroller (MCU). To allow for immediate, automatic reaction in case of problems, a sophisticated interlock system is implemented, as shown in Fig. 1. It can command an automatic shutdown of all voltages the power supply provides to the detector. A number of conditions are monitored to compute the shutdown signal. Besides the over-voltage protection and monitors for the mains input voltage and internal temperature of the power supply, a watchdog IC for analog regulators triggers a shutdown when the MCU fails to send a proper heartbeat signal for any reason. An analog principle is used between the MCU and the control IOC. Here, a periodic heartbeat signal is part of the CHROMOSOME protocol. Together, both heartbeats ensure that the output of the power supply can only be enabled when the MCU is operational, and the slow control system is connected. In the IOC, a single PV to shut down all connected power supplies is provided. It is accessible as an "emergency shutdown" button from all power supply- related operator screens, or from other IOCs that implement additional interlock functionalities.

Table 1: Alarm Conditions in the Power Supply System

Enabled	Set V	Actual V	Mode	State
no	1.8 V	0.0 V	CV	OK
no	1.8 V	1.8 V	CV	OV
yes	1.8 V	1.5 V	CV	UV
yes	1.8 V	1.5 V	CC	OC
yes	1.0 V → 1.8 V	1.5 V	CV	OK
yes	1.8 V	2.0 V	CV	OV
yes	1.8 V	1.8 V	CV	OK

Any emergency shutdown triggers an orderly shutdown sequence that operates independently from the MCU by means of adjustable RC delays. It ensures that the various voltages for the system are brought down in a safe order. In the case of mains failure, a large capacitor provides enough power to complete the shutdown sequence.

Input Validation

All data requests (especially voltage and current set points) towards the power supplies are validated before being sent to the devices. Beyond hard limits set in the corresponding EPICS records, dynamic limits depending on other settings are considered. Typically, this includes a maximum allowed difference between a pair of voltages, or current limits based on the state of the system.

The verification is done purely inside the EPICS database. An example set of PVs implementing the functionality is shown in Fig. 2. In this figure the requested value in (#3) is validated by a dedicated calcout record (#4), which filters malicious values before they reach the ao record (#5) of the device driver. An alarm is triggered and stored in a bo record (#7) if the requested and set values do not match. Note that the behavior is different to just using DRVH and DRVL in the ao record (#5), as invalid values are not clipped to the maximum/minimum allowed, but completely ignored. Additional logic in the record (#2) resets the channel to idle (i.e. 0 V) when the emergency shutdown of the PS has been triggered as recorded in record (#1). A similar but slightly more complex logic is used to introduce dependencies between different values.

Access control lists are used to ensure that the actual set record (#5) is not accessible from outside the IOC, so that the verification cannot be bypassed. Access to the PVs defining the conditions is limited to experts' accounts.

Readback Monitoring

The provided voltages of the PS are monitored by both the PS itself, and by the external SC system. The EPICS database structure used to implement the monitoring is shown in Fig. 3 and explained in the following: First, the

ISBN 978-3-95450-148-9

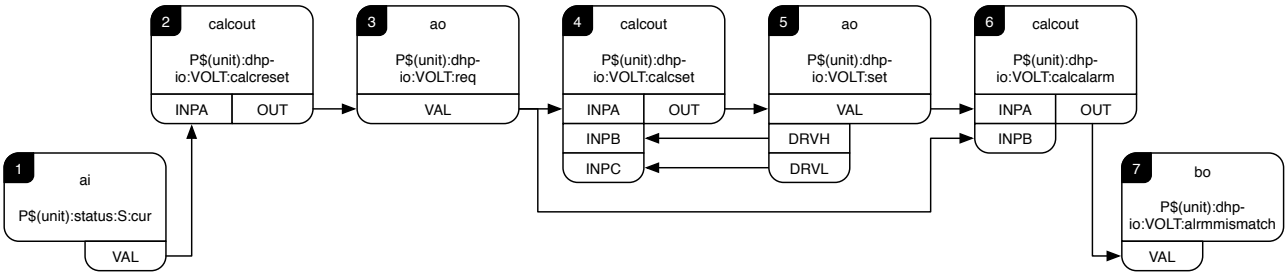

Figure 2: The EPICS database structure implemented to validate input values.

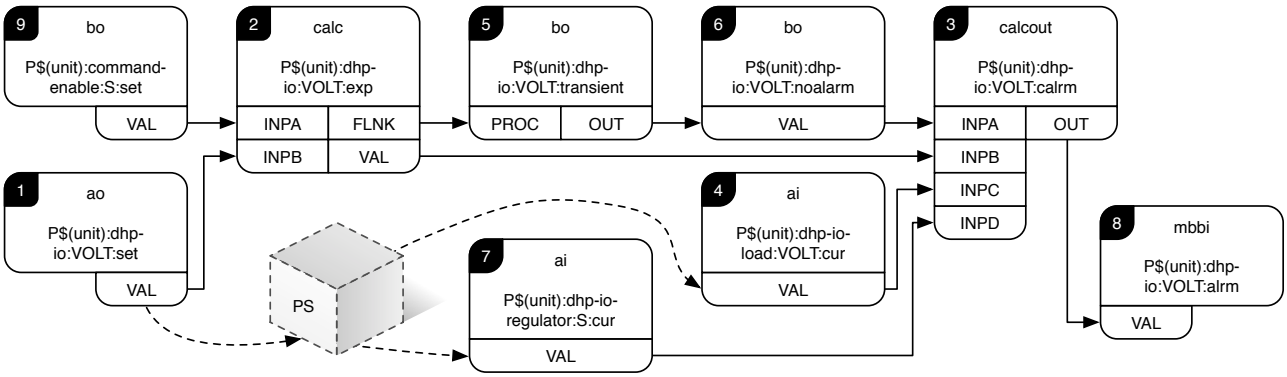

Figure 3: The EPICS database structure implemented to detect error states in the power supply system.

expected voltage at the load (record #2) is computed. It corresponds to the voltage requested by the user (#1), if the PS is enabled, or to 0V otherwise (#9). The (#3) record compares it to the actual voltage (#4), which is read back from the PS device. To avoid spurious alarms during voltage ramping steps, the alarm is suppressed for one second in case of the expected voltage changes. To that end, the bo records (#5) with OMSL closed_loop and (#6) with HIGH 1.0 are used to form a PV that is "on" for one second after a change in record (#2). The signal of this record is then considered in the logic for raising an alarm in case of a voltage mismatch. Also considered is the current regulator state (constant current vs. constant voltage) of the channel available in the record (#7). The effect of the implemented logic for a channel that is supposed to provide a fixed voltage is shown in Table 1. The detected states are OK (no alarm), OV (over-voltage), UV (under-voltage), and OC (over-current). All comparisons take the precision of the voltage readback into account by accepting readback data as matching within a sufficiently wide range around the desired value. Finally, the state is pushed into an mbbi record (#8), where mnemonic names are assigned to the states. This is the record whose state is shown in the GUI.

Radiation Monitoring

The PXD, as the innermost detector of Belle II, is most at risk to receive dangerous doses of radiation. Therefore a set of several radiation monitors are positioned at locations of expected high doses that have been identified from simulations and subsequently used to monitor the actual dose rate at all times. The monitors are controlled via a PLC that also computes interlock conditions on fast, extremely high rates, and slower but still higher than normal rates. In case of dangerous conditions, an immediate dump of the beams is requested from the SuperKEKB accelerator via a dedicated hard-wired line of communication.

The Modbus protocol is used to communicate with the PLC from the EPICS network. It is used to configure the interlock conditions, and to monitor the rates for display and archiving purposes.

CONFIGURATION DATABASE

The system configuration required to start a run is stored in a configuration database. An XML-like tree structure of configuration variables is efficiently stored in an SQL (currently PostgreSQL is used) database using a wandering tree algorithm to implement revision control system like functionalities. This way several files with multiple revisions per file can be stored in the database, and each entry can be queried for its history (change author, date and commit text). On run start, the desired run type category is used to identify the desired configuration.

The configuration data are made available to the system by means of an IOC that reads the configuration from the database and exports the configured PVs in the EPICS network.

The storage itself is tamper proof: After a commit, a revision cannot be modified in any way, it can only be superseded by a new revision. This is enforced by access controls on the database level and ensures that the active configuration at any point in time can positively be identified only from the configuration id in use at that time.

ISBN 978-3-95450-148-9

RUN CONTROL

The run control of the PXD is implemented in a hierarchical way. The desired run state is set either by the user for PXD-only test runs, or received from the Belle II master run control. It is forwarded to all PXD subsystems, which can decide to participate in the automatic run control, or be left out of it for the current run. A new state is only considered as reached, when all dependent subsystems confirm it. The actions during the state transitions are implemented using the sequencer module for EPICS[4].

The most complex part controls the ASICs on the detector modules and the PS. About 100 steps are required to bring up the digital power for the ASICs, configure them, and finally enable the analog power to the detector. Sanity checks throughout the sequence ensure that the sequence completes successfully. The state machine code is simplified by means of the C preprocessor. After converting it with snc, the code is compiled in C++ mode.

On a larger scale, the Belle II detector overall implements its run control and power supply control using the Network Shared Memory system version 2 (NSM2). A bidirectional gateway between NSM2 and EPICS is used to establish the communication.

LOGGING INTEGRATION

A C++ logging library that can interface with JMS2RDB, a connector layer between the Java Message Service and Relational Databases, which is also used for log messages from CSS components, has been written. It forwards log messages to an ActiveMQ server via the STOMP protocol[5] that is converted by ActiveMQ to be understood by JMS2RDB. IOCs can use this library to implement their own logging. This allows for all messages from the entire system to be combined in the "Message History" view in CSS.

Besides the STOMP implementation, the library features a thread-safe and fast design, and automatic generation of a backtrace in case the application crashes. For the application thread, posting a log message is always $O(1)$, because the message is only put in a queue to be processed by another thread. The implementation is integrated into EPICS in the sense that a DBD file is provided that, when loaded, provides commands on the IOC level that allow the dynamic configuration of logging destinations, changes of the log levels per subsystem or destination, etc. Adjusting log levels via PVs is planned for the next release.

IMPLEMENTATION STATUS AND OUTLOOK

The development of the SC system for the PXD is progressing from the stage where all components are controlled independently from their respective developers to a more integrated mode of operation. Tests and measurements with several systems in use together are now common and working smoothly.

The next milestone is a test beam with all system components early 2016, when all components of the SC system are expected to be ready. During the commissioning phase of the accelerator, a partial PXD system, installed in the Belle II detector on the beam line, and the final detector in the assembly room have to be supported at the same time. A full setup of the slow control system will be used for both purposes. The hardware used in these setups will be combined for the final installation, so that two parts of each component are available to build a redundant system using pacemaker/drbd that offers the high-availability required in HEP experiment controls.

[4] http://www-csr.bessy.de/control/SoftDist/sequencer/

[5] http://stomp.github.io/stomp-specification-1.2.html

ACKNOWLEDGEMENTS

This work has been supported by the German Federal Ministry of Education and Research (BMBF).

The main author would like to thank Thorsten Röder for his valuable contributions to this work.

REFERENCES

[1] J. O. Hill, "Channel access: A software bus for the LAACS," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 293, pp. 352–355, Aug. 1990.

[2] D. Levit, I. Konorov, B. Zhuravlev, S. Paul, T. Gessler, S. Lange, D. Munchow, B. Spruck, W. Kühn, J. Zhao, and Z. Liu, *Data acquisition system of the DEPFET detector for the Belle II experiment*. IEEE, 2013.

[3] D. Levit, I. Konorov, D. Greenwald, and S. Paul, "FPGA Based Data Read-Out System of the Belle II Pixel Detector," *IEEE Transactions on Nuclear Science*, vol. 62, no. 3, pp. 1033–1039, 2015.

[4] C. Buckl, M. Geisinger, D. Gulati, F. J. Ruiz-Bertol, and A. Knoll, "CHROMOSOME: a run-time environment for plug & play-capable embedded real-time systems," *SIGBED Review*, vol. 11, Nov. 2014.

[5] S. Yamada, R. Itoh, K. Nakamura, M. Nakao, S. Y. Suzuki, T. Konno, T. Higuchi, Z. Liu, and J. Zhao, "Data Acquisition System for the Belle II Experiment," *IEEE Transactions on Nuclear Science*, vol. 62, no. 3, pp. 1175–1180, 2015.

[6] T. Konno, R. Itoh, M. Nakao, S. Y. Suzuki, and S. Yamada, "The Slow Control and Data Quality Monitoring System for the Belle II Experiment," *IEEE Transactions on Nuclear Science*, vol. 62, pp. 897–902, June 2015.

[7] K. Kasemir, X. Chen, and E. Danilova, "The best ever alarm system toolkit," *ICALEPCS09*, 2009.

ISBN 978-3-95450-148-9