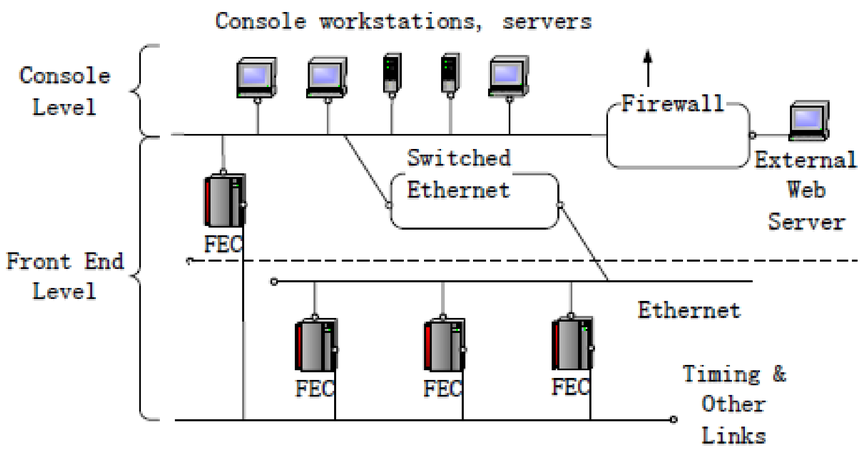
A New Simulation Architecture for Improving Software Reliability in Collider-Accelerator Control Systems

Y. Gao, T. Robertazzi, Stony Brook University, Stony Brook, NY 11794, USA
R. H. Olsen, J. T. Morris, K. A. Brown, Brookhaven National Laboratory, Upton, NY 11973, USA
primary author: ygao@bnl.gov

Abstract:
The control systems of the Relativistic Heavy Ion Collider (RHIC) is a complex system consisting of approximately 1.5 million control points. Its performance has a crucial impact over the whole accelerator suite. In this work, we propose a new simulation framework that can improve the robustness of the control system. It focuses on enhancing the reliability of its software codes by running automated testing. The architecture is described, followed by some key use cases in the current system. Moreover, the next development phase is proposed.

RHIC Control System Overview



- **Console Level**: Contains various workstations, servers, providing shared files, database, general computing resources.
- **Front End Level**: Closer to the field, has more than 500 Front End Computers, running on VxWork real time operating system.

Components:
- Accelerator Device Object (ADO)
- Controls Name Server (CNS)
- Logging system…
- Notification server…

Tools:
- Parameter Editing Tool (PET)
- Logging Data Display Tools: Gpm, LogView…

Motivation

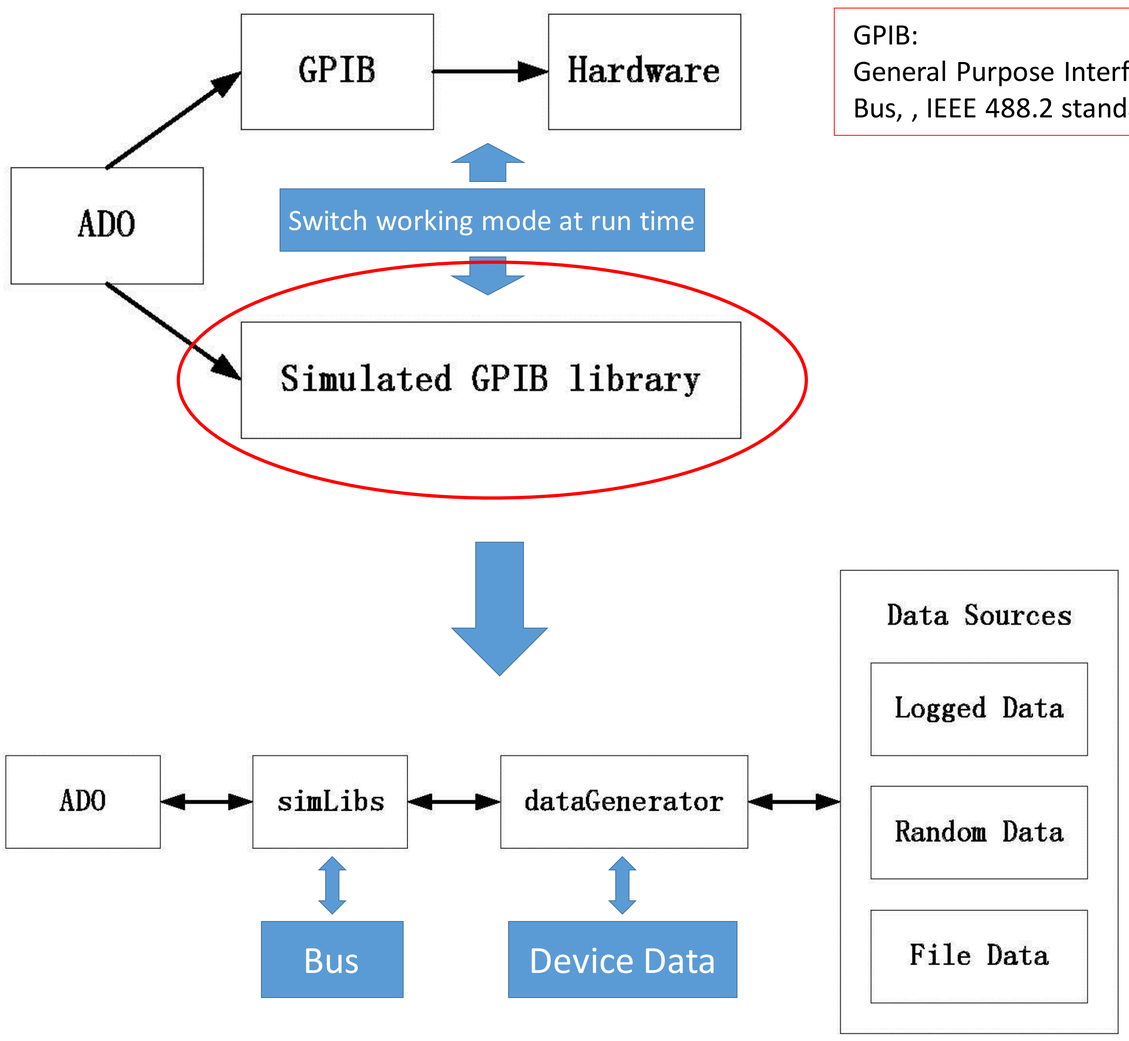
To improve ADO codes reliability

Real-world communication of an ADO:

ADO ⟷ Bus ⟷ Device

- To develop a simulation platform to do testing without interrupting normal accelerator operations.

Basic Simulation Structure

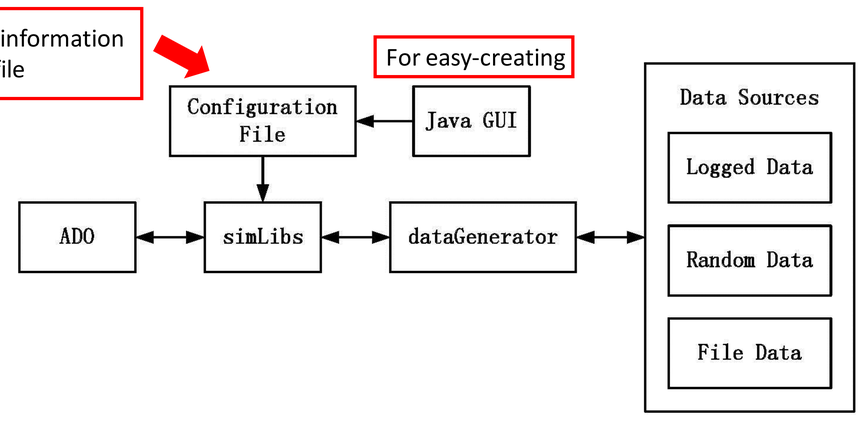


GPIB:
General Purpose Interface Bus, , IEEE 488.2 standard

- simLibs acts the role of bus interface connecting ADO with testing data;
- dataGenerator provides testing data from three possible sources;
- Logged data are historic data, which are by definition realistic;
- Random data are generated randomly, can be used for testing to cover a data range;
- File data are written before simulation runs, are useful when programmers know which parts of codes tend to fail and under what conditions.

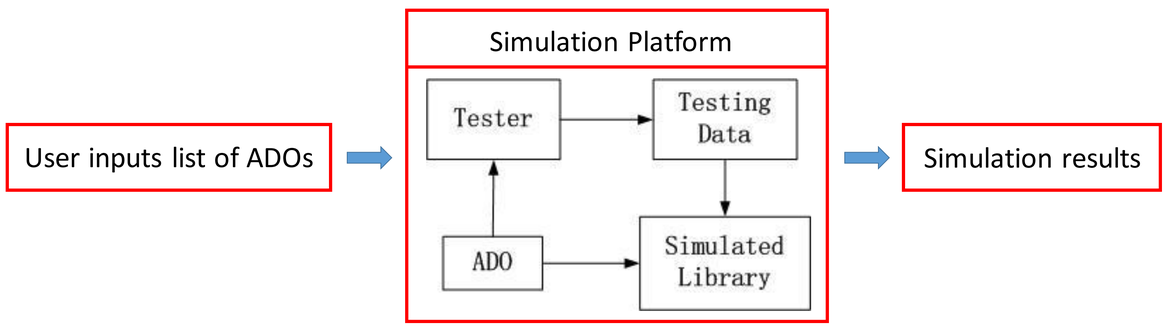
Generalization for Various Types of Hardware

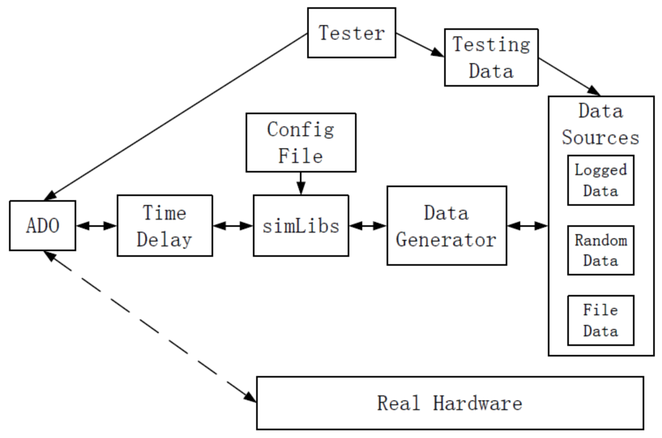
- Contain device information
- Standard XML file



- Configuration file uses standard XML format, which contains all necessary information to interact with different types of hardware;
- A Java GUI is supplied to easily create the file.

Testing Automation

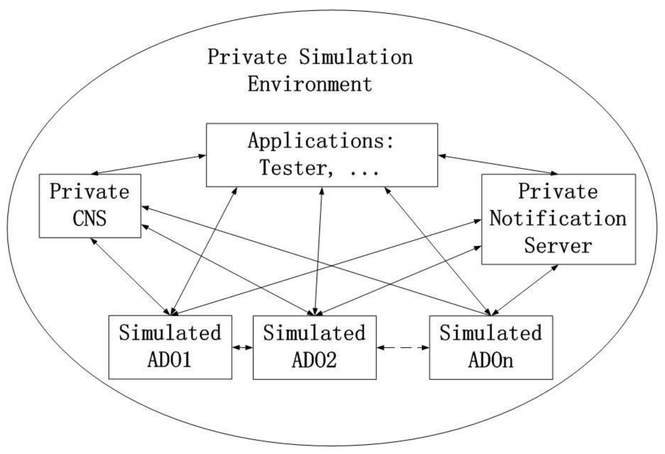




- Tester collects information from ADOs, and generates testing data;
- A time delay module is also added to better simulate real communication scenario;
- Error handling of time delays is also added.

- Improve robustness of ADO codes by running testing data.
- Verify new features of software, regression testing.
- Substitute for real hardware when they are not available.
- Specialized testing, control parameterization method.

Next Development Stage



- Construct a completely self-contained simulation environment, which allows doing simulation in all aspects of the system without impact on the real operations;
- Improve the user interfaces, and automated testing procedure;
- The current simulation framework contains just the basic elements. Many other components remain to be built;

Conclusions

In this work, we present a way to assist in developing and testing software in the RHIC control system. A new simulation architecture is proposed which mainly aims to improve ADO code reliability. Key use cases of the testing platform are listed and analyzed. The next development stage is described.

Work supported by Brookhaven Science Associates, LLCunder Contract No. DE-SC0012704 with the U.S. Department of Energy.