C. Wang
A. Booth
Y. Chen
M. Botlo

Graphics in DAQSIM

Superconducting Super Collider Laboratory

Superconducting Super Collider Laboratory is an equal opportunity employer.

Graphics in DAQSIM[*]

C. Wang, A. Booth, Y. Chen, and M. Botlo

Superconducting Super Collider Laboratory[†]
2550 Beckleymeade Ave.
Dallas, TX 75237

June 1993

DISCLAIMER

MASTER

DISCLAIMER

Portions of this document may be illegible
in electronic image products. Images are
produced from the best available original
document.

Graphics in DAQSIM

C. C. Wang, A. W. Booth, Y. M. Chen, and M. Botlo

Superconducting Super Collider Laboratory,[1] Dallas, Texas 75237, USA

Abstract

At the Superconducting Super Collider Laboratory (SSCL) a tool called DAQSIM has been developed to study the behavior of Data Acquisition (DAQ) systems. This paper reports and discusses the graphics used in DAQSIM. DAQSIM graphics includes graphical user interface (GUI), animation, debugging, and control facilities. DAQSIM graphics not only provides a convenient DAQ simulation environment, it also serves as an efficient manager in simulation development and verification.

I. INTRODUCTION

DAQSIM [1] is a tool for studying various Data Acquisition (DAQ) systems at the Superconducting Super Collider Laboratory (SSCL). This tool allows system designers to evaluate the performance of alternative architectures of data collection networks in terms of throughput, dead time, buffer usage, and other system parameters.

DAQSIM has been continuously developed and enhanced with new features. One of the major enhancements was adding graphics to the simulation. The graphics in DAQSIM is not intended to become a general-purpose graphical library or tool. Instead, it is serving as a prototype of graphical environment strictly attached to the DAQSIM simulation. DAQSIM graphics includes a GUI, animation, control, and debugging facilities. These graphical functions are presented in the following sections.

II. GUI

MODSIM II [2] is the object–oriented simulation language used in DAQSIM. SIMGRAPHICS II, a graphical package that comes with MODSIM II, is therefore selected to implement the GUI. One of the features SIMGRAPHICS provides is that graphics is still coded in MODSIM and embedded in simulation code.

DAQSIM GUI allows users to set dynamically simulation parameters, configuration options, etc., at run-time. Graphical objects are presented in the form of windows, menu bars, buttons, check boxes, text boxes, and/or value boxes. While windows are created in run-time, most of these objects are initially built up using a SIMGRAPHICS graphics editor named SimDraw and then stored into designer's graphics libraries. To use these objects, one simply has to load them from the corresponding library, invoke an "AddGraphic" method of a target window, and "ASK" the object to "Draw".

Figure 1 shows an instance of GUI when the "physics" category is picked from the menu bar labeled "parameters". The letter "I," "R," or "B" prefixing each physics parameter indicates its data type as Integer, Real, or Boolean, respectively.

In addition to GUI, SIMGRAPHICS supports animation and presentations graphics. However, we used it only to provide the aforesaid GUI functions. All other graphics in DAQSIM, presented in the following sections, were provided by a powerful graphical package called DataViews [3]. This is because DataViews achieves more flexibility and versatility in producing custom displays and runs much faster than SIMGRAPHICS when there are a large number of objects on the screen. Nevertheless, it should be clear that this is not the equivalent of saying SIMGRAPHICS should always be replaced.
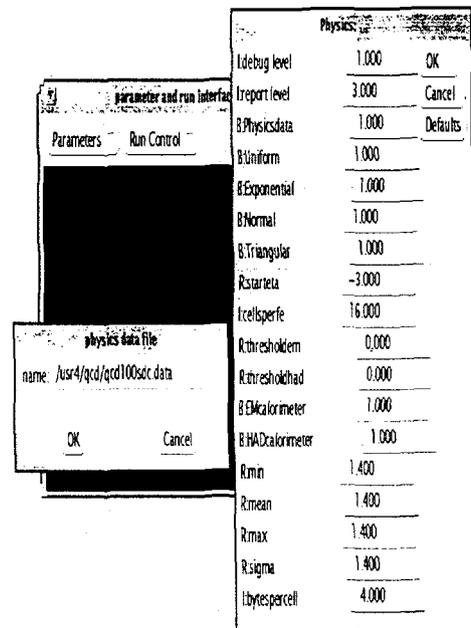

Fig. 1. An instance of DAQSIM GUI.

III. ANIMATION

A "healthy" discrete event simulation should always contain animation options [4]. In DAQSIM animation, DataViews dynamically takes data generated in simulation and represents the current values of data in the form of graphs, charts, etc. To enable such data passing, certain communication channels are needed to bring the monitored data from MODSIM to DataViews.

There are several interprocess communication facilities available on UNIX. Shared memory is the most efficient one when the volume of communicating data is large, as in the case of DAQSIM. For increased performance, two blocks of shared memory (instead of one) have been constructed as the intermediate buffers where MODSIM/DataViews stores/

retrieves the desired data. Semaphores have been placed around operations on shared memories to keep the simulation and the graphics synchronized.

DAQSIM animation consists of three-level hierarchical views showing a network of Data Collection Circuits (DCCs) along with some simulation statistic information such as the distribution of mean event data size detected in front-end circuits. Figure 2 through Figure 4 show snapshots of top, middle, and bottom views, respectively. While only one view is displayed at a time, users may click at the "zoom buttons" at any time to switch to another view of a different level.
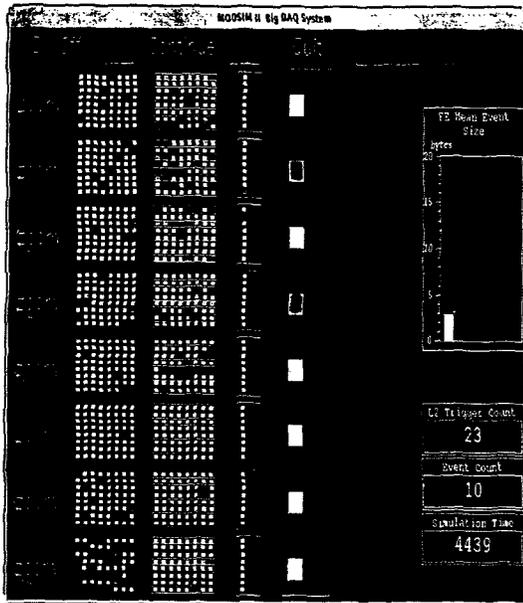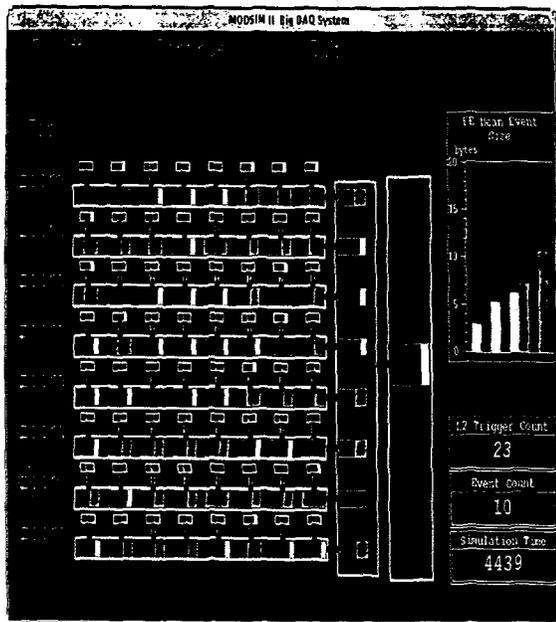

Fig. 2. The top view of DAQSIM animation.

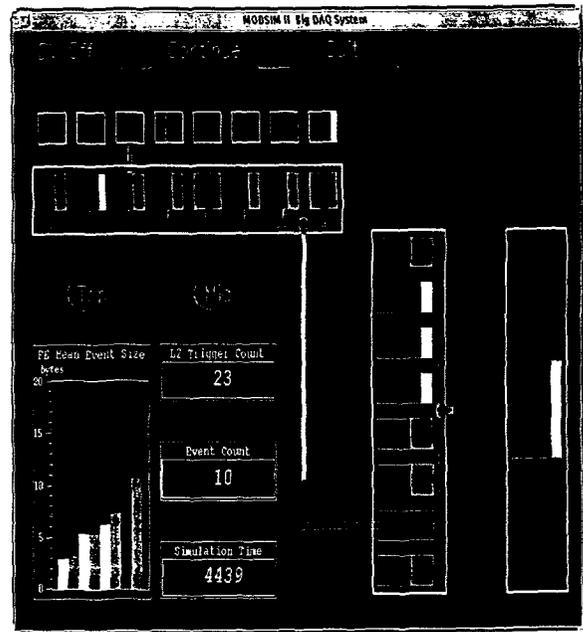Fig. 3. The middle view of DAQSIM animation.

Fig. 4. The bottom view of DAQSIM animation.

The top view shows 512 front-end buffers connected to 64 first-layer DCCs, which in turn are connected to eight second-layer DCCs, connected in turn to a third-layer DCC. Each DCC is represented by a white, shallow rectangle that encloses one or eight buffers, which appear as small solid squares. By assigning each buffer depth a unique color, DCCs dynamically change color when the amount of data they contain is changed. Before an inter-layer data transfer takes place, all the buffers in the sending DCC blink three times to advise the users. Although there are more than 1000 graphical objects in this view, one can observe data moving from one buffer to another across the network.

There are eight middle views, each containing one-eighth of the DCCs in the top view. Besides the color changes, the colored area of a DCC in middle views expands or shrinks proportionally. In the histogram chart for front-ends, only the vertical bar corresponding the underlying middle view is displayed in color. Therefore, this provides the users the identification of the current middle view.

Similarly, each middle view can be broken into eight bottom views. In a bottom view, in addition to the features in views of higher hierarchy, the links between DCCs even show the passage of data. This feature makes the animation appear very realistic. Also note that the identification of the underlying view is provided by only showing the corresponding path between the second layer and the third, as shown in Figure 4.

IV. CONTROL AND DEBUGGING FACILITIES

Many control facilities in DAQSIM graphics require a two-way communication scheme other than shared memory between the simulation and the graphics. Several message queues have been built up for this purpose. The control facilities include freezing the display, halting MODSIM simulation, turning on/off graphics, inserting/removing faults on links, and so on. It is noteworthy that with graphics temporarily being

turned off, simulation would run much faster and graphics can be automatically brought back when the user-specified condition has been met. The "DV On/Off button" in head panel provides this function. Once clicking on it, one may bring up/down DataViews graphics immediately or preset the on/down conditions in the popped up window as shown in Figure 5. When both the "trigger count" and "simulation time" conditions are set, the specified command would take effect when one is reached first.

Animation also enhances the ability to perform basic debugging. One's model can be easily verified from the immediate graphical feedback. Debugging capabilities become even more powerful when combined with the control facilities described above. For instance, a break point can be set up by tuning a condition which controls the on/off of simulation/graphics.
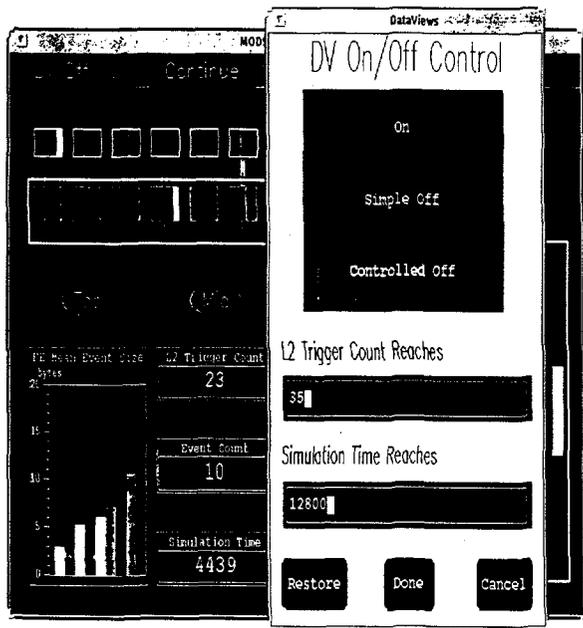
V. CONCLUSION

We developed a graphical manager for DAQSIM that is being successfully used in studying DAQ systems at the SSCL. DAQSIM graphics provides many features that make DAQ studies easier and more productive. Future work will include adding more graphical analysis facilities and making all the graphics reusable with other similar DAQ simulations.

VI. REFERENCES

[1]   A. W. Booth, et al., "DAQSIM: A Data Acquisition System Simulation Tool," IEEE Nuclear Science Symposium and Medical Imaging Conference, Orlando, FL, Oct. 1992.

[2]   MODSIM II is a proprietary product of CACI Products Company, 3344 N. Torrey Pines Ct., La Jolla, CA 92037.

[3]   R. McHaney, "Is Your Simulation Sick?" Manufacturing Systems, May 1992.

[4]   DataViews is a proprietary product of VI Corporation, 47 Pleasant St., Northampton, MA 01060.

Fig. 5.   The "DV On/Off Control Window" generated by clicking "DV On/Off button."