GA–A26193

EXTENDING THE CAPABILITIES OF THE DIII-D PLASMA CONTROL SYSTEM FOR WORLDWIDE FUSION RESEARCH COLLABORATIONS

by

B.G. PENAFLOR, J.R. FERRON, M.L. WALKER, D.A. HUMPHREYS, J.A. LEUER,
D.A. PIGLOWSKI, R.D. JOHNSON, B.J. XIAO, S.H. HAHN, and D.A. GATES

AUGUST 2008

GENERAL ATOMICS

DISCLAIMER

GA–A26193

EXTENDING THE CAPABILITIES OF THE DIII-D PLASMA CONTROL SYSTEM FOR WORLDWIDE FUSION RESEARCH COLLABORATIONS

by
B.G. PENAFLOR, J.R. FERRON, M.L. WALKER, D.A. HUMPHREYS, J.A. LEUER,
D.A. PIGLOWSKI, R.D. JOHNSON, B.J. XIAO,* S.H. HAHN,[†] and D.A. GATES[‡]

This is a preprint of a paper to be presented at the Twenty-Fifth Symposium on Fusion Technology, September 15-19, 2008, in Rostock, Germany, and to be published in *Fusion Engineering and Design*.

*Institute of Plasma Physics, Chinese Academy of Sciences, Hefei Anhui, PR China.
[†]National Fusion Research Institute, Dae Jeon, Korea.
[‡]Princeton Plasma Physics Laboratory, Princeton, New Jersey.

Work supported by
the U.S. Department of Energy
under DE-FC02-04ER54698 and DE-AC02-76CH03073

GENERAL ATOMICS PROJECT 30200
AUGUST 2008

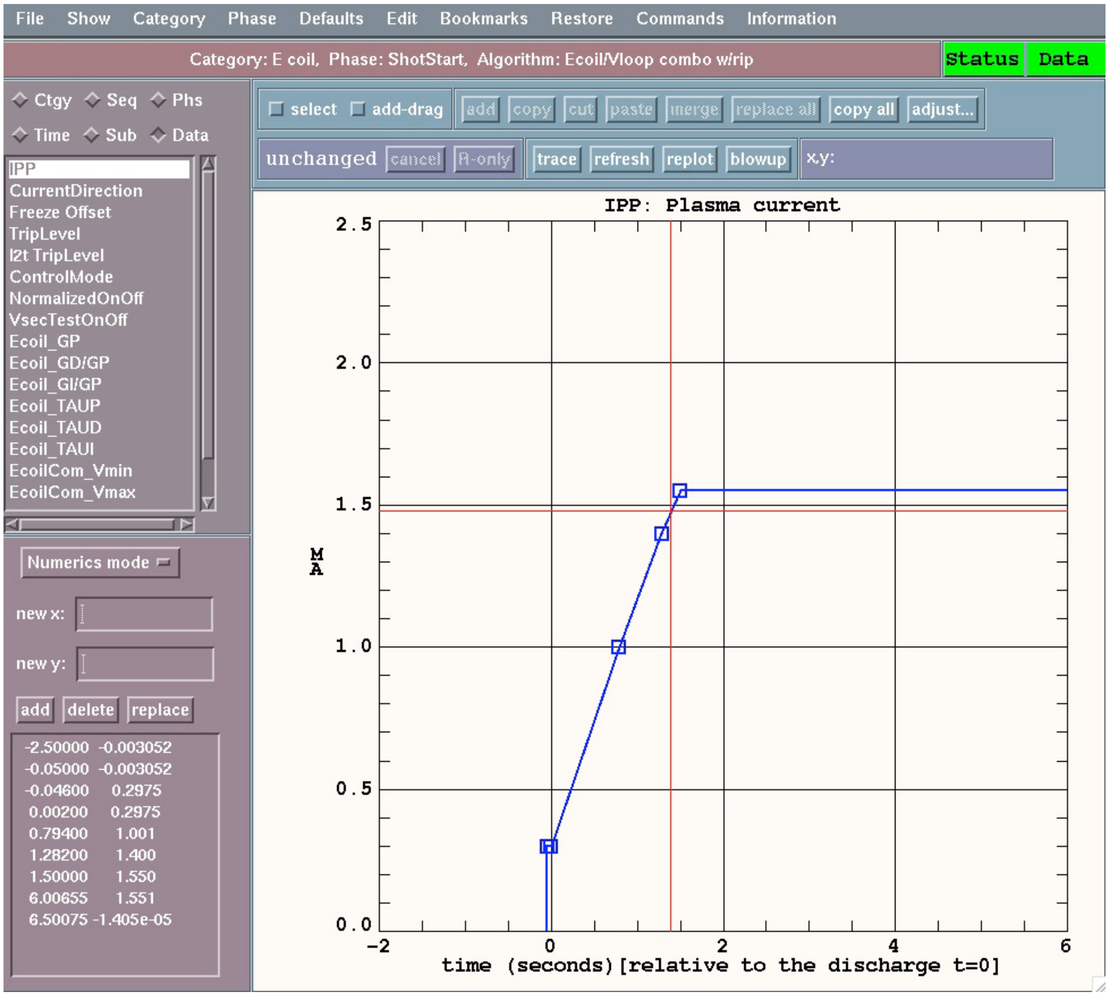
GENERAL ATOMICS

Abstract

This paper will discuss the recent enhancements which have been made to the DIII-D Plasma Control System (PCS) in order to further extend its usefulness as a shared tool for Worldwide Fusion Research. The PCS developed at General Atomics is currently being used in a number of fusion research experiments worldwide, including the DIII-D Tokamak Facility in San Diego, and most recently the KSTAR tokamak in South Korea. A number of enhancements have been made to support the ongoing needs of the DIII-D tokamak in addition to meeting the needs of other PCS users worldwide. Details of the present PCS hardware and software architecture along with descriptions of the latest enhancements will be given.

1. Introduction

The Plasma Control System (PCS) developed at General Atomics (GA) is a software application used to monitor and control various attributes of plasmas generated for fusion research including plasma shape, position, temperature, density and rotation. It includes a graphical user interface (Fig. 1) which can be invoked by multiple operators that allows for simultaneous interaction with the control system. The control system provides parallelization of tasks amongst a set of real-time computers to perform all of the data acquisition, computational analysis and feed back control necessary for attaining the desired plasma characteristics. The software architecture follows a modular design [1] allowing for easy addition and modification of control algorithms. The system design also allows for the incorporation of various types of hardware including different computing, data acquisition, input/output and real-time networking devices.



Category: E coil, Phase: ShotStart, Algorithm: Ecoil/Vloop combo w/rip

```
-2.50000  -0.003052
-0.05000  -0.003052
-0.04600    0.2975
 0.00200    0.2975
 0.79400     1.001
 1.28200     1.400
 1.50000     1.550
 6.00655     1.551
 6.50075 -1.405e-05
```

Fig. 1. PCS User Interface.

The DIII-D version of the PCS continues to undergo improvements and updates in response to the fast changing needs of the experimental research program. The recent successful commissioning of the KSTAR Tokamak at the National Fusion Research Institute (NFRI) in South Korea [2] has provided further validation of the adaptability and usefulness of the PCS beyond DIII-D. A version of the PCS successfully developed for use at KSTAR played a major role in allowing NFRI to achieve their objectives for first plasmas in a timely manner. The flexibility of the PCS has allowed it to continue to expand in scope and capability, making it an invaluable tool for fusion researchers not only at DIII-D but also a growing number of fusion research facilities [3] throughout the world.

2. PCS Application Overview

The PCS graphical user interface developed in the IDL language [4] serves as the front end to the real-time control system. It contains a number of features to allow multiple users to simultaneously view and easily update the large number of plasma parameters associated with each discharge. A plot based waveform editor allows for entry of time based data which evolve during the course of a discharge, while user customizable parameter data editors provide a means of specifying static information. Key features of the PCS user interface include the ability to prepare and save set up information into files for use in future discharges, the ability to selectively restore any or all parts of set ups used in past discharges, automatic notification of changes made by other users, extensive error checking, and access control to prevent unauthorized updates to PCS parameters. The interface is easily configurable to suit the needs of each specific installation.

A set of server processes running on the PCS host computer are used to manage the discharge parameters from the user interfaces for use in the real-time control system and also provide synchronization between the PCS and the discharge cycle. The waveform server process coordinates all changes to discharge parameters from each of the user interface clients. Raw information from the user interface clients is processed by the waveform server in order to set up the real-time control computers prior to each discharge. The lock server process provides the synchronization of the PCS with the various discharge phases such as the initialization of the discharge, start of real-time control and the end of discharge. A message server is used to manage and distribute messages from the various PCS processes to the user interface clients.

The real-time system of the PCS consists of the processes and hardware which perform the actual feedback control and monitoring during plasma discharges. It has been designed to allow users to develop control algorithms which can take advantage of distributed parallel computing within a multi processor framework. A customized Linux operating system is used to achieve real-time responsiveness through interrupt disabling providing minimal latency and dedicated cpu control for each of the real-time cpu processes. The real-time system is designed to run on any computer capable of running Linux and allows flexibility in the choice of hardware interconnections, data acquisition and other hardware input output devices. A typical PCS configuration consists of a set of rack mounted Intel-based PCI form computers with one or more processors connected in a real-time network such as Myrinet [5], acquiring data using cPCI form digitizers from D-TACQ [6] which are also capable of sending analog outputs to actuators.

The PCS software consists primarily of codes written in the C and IDL programming languages. IDL is used primarily for implementation of the user interfaces while C is used in the server applications and the real-time control processes. The software architecture organizes the control system into specific categories of control each of which contains a set of algorithms. Each category may have one or more phases in addition to multiple phase sequences. A phase is associated with a specific algorithm defined for that category. Phase sequences are used to define in what order and when each phase is to be executed during the discharge. Multiple phase sequences can be specified in order to allow the PCS to run a different algorithm in place of a preprogrammed one in response to an asynchronous event. The final major group in the software organization includes codes for specifying installation specific hardware configuration information. These include such things as the diagnostic information unique to each site, the output hardware, and number of cpus in the system and role of each of the cpus.

The PCS includes an extensive set of tools for testing the hardware and the software. A software simulation test tool can be used to run the entire PCS on a single computer without the need for any of the real-time computing hardware. It can be used in open loop simulations where data can be fed into it from previously run discharges, or in closed loop simulations [7] using data from a smart simulation process to test feedback control. The software simulation tool is highly useful for testing and verifying updates to software where the hardware is not essential to the results, or simply not available. A hardware test capability provides a means of testing performance of the PCS on the actual hardware providing a more complete and accurate gauge for validating the software. Hardware tests can be set up to run with diagnostic information obtained from previously run discharges in order to reproduce their results.

3. PCS Enhancements for the DIII-D Research Program

The latest configuration of the DIII-D PCS now includes three new Double Dual Core Intel Xeon based computers each running at 3.06 GHz speeds. In addition to the performance gains achieved through faster processors, the new computer systems have allowed the PCS real-time processes which once resided on separate computers to be combined to run within a single computer providing faster inter-process communications. This has proven to be beneficial to improving access to diagnostic data digitized by the PCS allowing more real-time processes to be on the same computers responsible for acquiring data. The new multi processor systems have also increased performance by providing more cpus to perform process threading on. The use of threads enables real-time processes to be easily broken up and distributed to run on several cpus in parallel.

In order to maintain stability and consistency within the real-time computer systems, updates to the operating system have been carefully managed, occuring only when required such as for the incorporation of new hardware or in response to necessary device driver updates. For the past several years prior to this year, the PCS real-time computers had been locked down to the 2.4 revision of the Linux kernel. Since the real-time computers all reside inside a local private network not visible or accessible from the outside world, it was possible to avoid the frequent and numerous kernel patches and updates which could possibly affect operation of the PCS. A migration to the latest 2.6 Linux kernel revision was performed recently as a result of the need to upgrade to the new multi-processor computers requiring the later revision in order to run properly.

PCS real-time calculations of the frequency, amplitude, and mode numbers from the smoothed cross power spectrum of Mirnov probe diagnostics are now routinely available on every discharge. A version of the DIII-D post shot spectrum analysis routines has been written for the PCS which uses data acquired in real-time from a 16 channel ACQ216 digitizer from D-TACQ. The ACQ216 digitizer is used to acquire the Mirnov probe signals at a 1MHz frequency and stream the data to the PCS real-time analysis process in bursts of 512 16-bit samples. The complete set of samples is saved to the local memory of the ACQ216 and written to the DIII-D data archiving system after every discharge. Results of the real-time version of the code have been consistent with the original post-shot version. The PCS version allows users to select any pair of probes from a set of ten Mirnov signals to perform the analysis in real-time. The analysis interval can be specified to occur as frequently as once every 512 microseconds, or multiples of 512 up to 51.2 milliseconds. The analysis codes make use of Fast Fourier transform functions from a free FFT library [8] downloaded from the web. Results of the analysis can be used by other PCS algorithms such as is now the case for fault detection and handling which uses the data in order to monitor specific modes and trigger preprogrammed responses. Analyzed mode information is also available for display in the control room using the PCS real-time scope application.

4. Improvements in Support of Collaborations

The continued expansion of the PCS has resulted in increased maintainability of the software and support for an ever-growing number of users. To maintain software version control, all changes to the infrastructure codes and their release are carefully managed at GA. This includes the core set of libraries used in every different version of the PCS which consists of all the libraries, scripts, subroutines and source files necessary for implementing a basic control system. The standard Unix based Source Code Control System (SCCS) is used to maintain these sources. The installation or site-specific software are all maintained at the individual research facilities using source code control methods appropriate to each site. These are the sources which specify hardware configuration information, categories of control and associated control algorithms unique to each location. Automated source file synchronization and transfers between GA and a number of the remote supported sites has been implemented in order to provide better coordination of updates to software.

To allow for further customizations of the PCS, more work has been done in order to identify and separate out generic capabilities to be included in the infrastructure codes from codes which are relevant only to a specific site. This has required some reorganization and rewriting of sections of the PCS software. A specific example of this has been a major rewrite of the real-time message passing routines to allow for different types of interconnect hardware. Much of this work was motivated by the needs of the KSTAR PCS which required the use of Reflective Memory (RFM) hardware as oppossed to the Myrinet hardware used at many of the other sites. Another enhancement which has been made with the aim of allowing each site to more readily customize their PCS versions has been the addition of automated code generators. These automated code generators greatly reduce the overall number of steps and amount of coding required to implement new control algorithms. They rely on a set of predefined macros which are processed by a program developed at GA to parse the algorithm files and expand them into the necessary C codes prior to compilation. Examples of this include macros used to specify waveform information and parameter data.

To address software quality assurance and testing issues a special PCS testbed platform has recently been set up at GA. This testbed system is comprised of a number of different hardware components used by several of the collaborative sites. It serves to provide a platform for performing local hardware and software testing of updates made to the PCS affecting the different collaborations as well as DIII-D itself. The testbed has been highly important in assuring the correctness and quality of new software before it is released for use on a tokamak. While a software simulation capability for testing the PCS has been available for a number of years, a hardware testbed platform is still necessar in order to test the aspects of the PCS which depend on hardware performance. A specific example of this would be timing and synchronization of messages sent between processes in real-time. The hardware testbed can be used to determine how fast algorithms will run in real-time and also be used to more accurately simulate computations which are highly dependent on the timing of the messages sent from one real-time processor to another.

5. Future Plans

Immediate plans for upgrades to the capabilities of the DIII-D PCS include incorporating Dual Quad Core processor systems to further increase performance speeds. Also planned is a replacement of the VME based Digital to Analog Converter (DAC) hardware with CPCI DAC output hardware from D-TACQ. The current DIII-D PCS is limited by the performance of an older 500MHz VME based computer which is used to send outputs to actuators through the VME DACs. Replacement of the DAC hardware will eliminate the dependency on the older and slower VME based computers from the system and improve the performance in sending out commands to the DIII-D actuators. In addition to this, the DIII-D PCS will need to be updated in order to accommodate planned upgrades to the Charge Exchange Recombination diagnostic system [9] and the Thomson Laser Scattering Diagnostic system [10].

In support of collaborative efforts a number of enhancements are being planned including software enhancements to allow the PCS to support data acquisition and data archiving for long pulse experiments, improvements to the real-time monitoring and scope display capabilities, increased support for different types of computing and interconnect hardware and improvements to security and access to the PCS from interfaces running remotely.

6. Conclusion

The Plasma Control System (PCS) developed at General Atomics continues to expand both as a local application for supporting fusion research on the DIII-D tokamak and as a collaborative tool enabling and enhancing fusion experiments worldwide. A well structured and flexible architecture have allowed for further growth and improvement through the incorporation of faster hardware and the addition of new analysis capabilities such as real-time spectrum analysis. This has allowed the PCS to keep in pace with the needs of the DIII-D experimental program. In addition to this, the recent successes in collaborations with NFRI in adapting a version of the PCS for use on the KSTAR tokamak have further demonstrated the PCS to be a highly robust and valuable tool for worldwide fusion research.

References

[1] B. G. Penaflor, *et al.*, A structured architecture for advanced plasma control experiments, Proc. of the 19th Symposium on Fusion Technology, Lisbon, Portugal, Vol. 1 (1996) p. 965.

[2] S. H. Hahn, *et al.*, Plasma control system for the "Day-1" operation of KSTAR Tokamak, this conference.

[3] B. G. Penaflor, *et al.*, Worldwide collaborative efforts in plasma control software development, Proc. of 6th IAEA Technical Committee Meeting on Control Data Acquisition and Remote Participation for Fusion Research, Inuyama, Japan, Vol. 83 (2008) p. 176.

[4] http://www.ittvis.com

[5] http://www.myri.com

[6] http://www.d-tacq.com

[7] J. A. Leuer, *et al.*, DIII-D plasma control simulation environment, Proc. of 20th IEEE/NPS Symposium on Fusion Engineering, San Diego, California, 2003.

[8] http://www.fftw.org

[9] K. H, Burrell, *et al.*, Improved CCD detectors for high speed, charge exchange spectroscopy studies on the DIII-D tokamak, Proc. of 15th High Temperature Plasma Diagnostics Conf., San Diego, California, 2004.

[10] B. D. Bray, *et al.*, Improved Density Measurements With a Rayleigh Calibrated Thomson Scattering System at DIII-D, Bull. Am. Phys. Soc. **48**, 181 (2003).

Acknowledgments

Work supported in part by the U.S. Department of Energy under DE-FC02-04ER54698 and DE-AC02-76CH03073.