Data transfer methods in Real Time controller of Ion Cyclotron High Voltage Power Supply

H. Dhola, A. Patel, R. Dave, A. Thakar, D. Parmar, K. Mehta, N. Goswami, N.P. Singh, S. Gajjar, U. K. Baruah,
ITER-India, Institute for Plasma research, Bhat, Gandhinagar 382428

1 2 *Abstract– Ion Cyclotron* High Voltage Power Supply (IC-HVPS) with dual output (27kV & 15kV, 3MW) is operational at ITER-India lab with a 1.5 MW Diacrode based RF source to be used for ICRF (Ion Cyclotron RF) system. The controller for ICHVPS is designed with LabVIEW Real-time (RT) PXI controller to support all control and monitoring operations of the Pulse Step Modulation (PSM) based power supply. Besides regulation of output voltage, the controller supports all essential features like, fast dynamics, low ripple and protection for source and loads. This is managed by close integration between RT system and FGPA (Field Programmable Gate Array) inside the controller.

Two types of data transfers inside the RT controller are implemented, 1) Between FPGAs and RT controller, 2) Between internal algorithms. RT controller and FPGAs are inter-connected through PXI bus. Critical functions, signal generations, monitoring & data acquisition and interfaces are assigned to FPGAs. All these processes are managed by Real Time Operating System (RTOS) installed in the controller. Most of the processes initiated by RT controller ends up performing certain task by invoking the algorithm inside FPGA directing some action to outer interfaces. Some of the process routines are implemented for managing the internal algorithms.

This paper discusses different techniques used mostly for transfer of information between RT controller and FPGAs. Information includes critical signals, data, measurements, commands and configuration required for internal algorithms. Algorithms use various methods like DMA-FIFOs, local variables, arrays, Trigger lines to transfer information. RT-FIFOs are used to handle bulk data in runtime to improve performance. Methods have been chosen carefully to fulfill critical requirements, balancing the available resources while managing limitations. Discussion also includes some specific choices made for synchronized and asynchronous transmission of parallel and sequential data.

I. Introduction

Real Time Control System has been used to operate and control Ion Cyclotron High Voltage Power Supply (IC-HVPS) with dual output. The IC-HVPS's two outputs are controlled by a single controller in predefined operational sequence. Both output/stages has to be controlled in synchronization. Powerful FPGAs (Field Programmable Gate Array) combined with RT (Real Time) controller controls a High Voltage Power Supply of discrete stacked low voltage modules. RT Control system functionalities are categories as: 1. Initialize and Configure 2. Control and Regulate 3. Handling of critical protection functions. These main functionalities are divided physically between RT and FPGA sections. RT takes care of operation and regulations, while FPGA module basically keeps all the Switch-mode Power Supply (SPS) modules in sync and executes high speed actions for trips/faults.

To achieve all above functionalities close interaction of FPGA and RT controller is required. PXI (PCI eXtension for Instrumentation) based chassis provides communication bridge between RT controller and FPGA. PXI bus has 133 Mbps data rate and complex data transfer protocols [1]. This paper only addresses the data transfer implementation methods using available architecture and functionalities provided by LabVIEW RT software.

II. System Description and Requirements

RT control system has three major components. 1) PXI Chassis, which provides housing, supply, cooling and a medium for interaction between RT controller and FPGAs. 2) The RT controller which holds the main algorithm and manages overall operation and interfaces. 3) FPGA Modules which performs processes/functions for data acquisition, Control signal generation, serial communication and High speed field events.

PXI platform is found reliable [2] and suitable option for High Voltage Control System, as its modular architecture with COTS (Commercial off the Shelf) components [3].

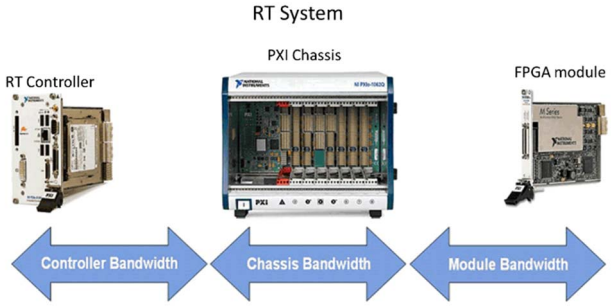

Fig. 1. Components of the RT System which contains RT Controller, FPGA Cards and PXI Chassis[4]. PXI chassis contains PXI Bus with trigger line

Fig 1. Shows physical architecture of the RT control system. All the events detected by FPGAs shall to be communicated to RT controllers and corresponding logical

Manuscript received May 30, 2016. This work is supported by Power Supply group at ITER-India, Institute for Plasma Research.

Hitesh A Dhola and co-authors are with the ITER-India, Institute for plasma Research, Gandhinagar, India (email : Hitesh.dhola@iter-india.org)

978-1-5090-2014-0/16/$31.00 ©2016 IEEE

responses and commands generated at RT controller shall to be delivered back to FPGAs.

Following data are identified to be exchanged between FPGA and RT Controller for successful operation of IC-HVPS.

a) Controller Configuration data
b) Synchronization data
c) Monitoring Data
d) Power Supply Configuration Data
e) Continuous data acquisition
f) Continuous control data
g) High speed events like Trips, Faults and Alarms

These data transfer can be categorized as per following,

1. CPU (Central Processing Unit) transfer
2. DMA transfer
3. Trigger line Transfer

All these data are channeled through PXI back plane or trigger lines of PXI Chassis.

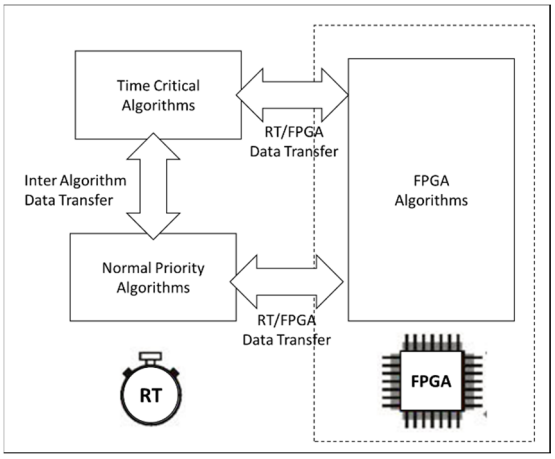

Fig. 2. RT Controller and FPGAs are connected through LabVIEW real time module and LabVIEW FPGA module at functional level. LabVIEW RT is a platform on which RT systems is built with different interfaces like FPGAs. RTC has time critical and normal priority loops. Both communicate with FPGA algorithms as per requirements using PXI backplane. RTC uses RT FIFO to exchange time critical information and variable interfaces for normal priority exchanges between internal algorithms.

FPGA Functions are extension of main algorithms implemented in RT controller (RTC). Functions/processes which are hard time bound like interlock signals or requires higher performance are assigned to FGPA. Supervision of those function inside FPGAs are handled by RT algorithms. FPGAs also provide all the data required for RT algorithms.

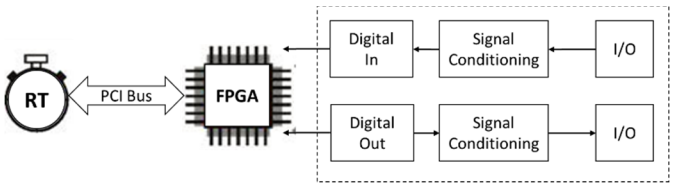

Fig. 3. FPGA provides RT controller interface to outer world at High voltage side. FPGA directly connects to PXI bus when installed inside PXI Chassis.

FPGA is connected with RT controller via Bus interface. It has interface with outer world by means of Digital IOs.

III. Data Transfer

Communication between the FPGA and RT Controller is basically exchange of information and synchronization of algorithms implemented at both sides. Algorithms runs independently and asynchronously on FPGA and RT Controller. To synchronize algorithms between two systems, synchronization code needed to be added on both side. This can be achieved by using inbuilt timing clocks or manually assigning functions to trigger lines. The data acquired by FPGA algorithms shall be synchronized with RT algorithms which process and uses that data to drive state machines, Error handling, HMI (Human Machine Interface) Displays and implementation of protection. To address all the data exchange requirement described in requirement section, various data transfer methods are used.

A. Controller Configuration on boot up

Few information are not time critical and doesn't require synchronization. Information like configuration of FPGAs can be transferred asynchronously from RT Controller. On boot, RTC loads the default algorithm and configures all the channels by CPU transfer method for operation.

B. Synchronization data

As describe earlier, operation of dual output power supply by single controller and requirement of operation sequence, imposes close synchronization requirement between RTC and FPGAs. Trigger line has been used for synchronization of process initiations and its order viz. turn off, turn on and PWM (Pulse Width Modulation) generation. Those processes decides which stage will turn on or turn off at what time. Control signal generation is needed to be started in one stage after another, which may otherwise lead to damaging scenario. Whereas HV off should follow the reverse sequence.

C. Monitoring Data

Monitoring data drives the state machine and provides information about healthiness of power supply along with the peripherals connected in the system; hence is supposed to be available before operation and during operation. All the field monitoring is done by PLC (Programmable Logic Control). RT controller supports Ethernet TCP/IP as a standard interface. Field monitoring data is transferred to RT controller via TCP/IP protocol [5].

Other monitoring devices from power supply [6] and control cubicles are also interfaced with FPGA. FPGA dispatch constant monitoring to RT via CPU transfer. SPS modules also has VFC (Voltage to Frequency) based monitoring of DC link voltages. Generally SPS modules provides information according to pre-defined state machines. DC link information is independent of state machine and continuously acquired through multiple FPGAs with approx. 48 dedicated Channels. Acquired data is visible at HMI and used to monitor SPS modules though RT controller.

D. Power Supply Configuration Data

Controller configures power supply modules before operation starts. Controller transfers 12 packets of information to SPS modules and receives 12 packets of acknowledgment for comparison. After verification, controller concludes the configuration. Custom UART (Universal asynchronous receiver/transmitter) protocol has been used to accomplish this transfer for all 48 SPS modules sequentially.

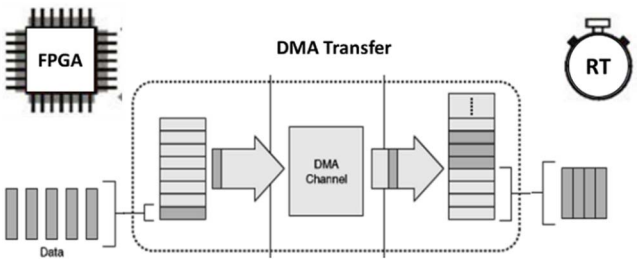

Fig. 4. DMA Engine implements DMA transfer using a FPGA buffer [7], RT Buffer and a DMA Channel. Data is serially added one by one and read in chunks.

DMA (Direct Memory Access) does not involve the RT processor; therefore, it is the fastest available method for transferring large amounts of data between the FPGA and RT controller. Additionally it also relaxes CPU resources of RT controller and Logic gates of FPGA resources.

Configuration information has to be transferred serially and without missing a bit, thus FIFO to be involved to transfer serial packets of data using DMA.

E. Continuous data acquisition

Once power supply is in operation, all SPS modules keep transmitting data to controller. Controller receive all the 48 packets at the same time. RT controller then decodes them and communicate it to RT controller for further processing. This has been done using CPU transfer. Sending single cluster of data element instead of individual information transfer significantly reduces load on RT controller.

The received data contains information about healthiness/faults and current state of module; which enables RT controller to drive the state machine further. Till certain state of SPS module, controller allows for inclusion of the faulty SPS module if it resets on its own. This is implemented by continuous prompting by RT to FPGA.

F. Continuous control data

Regulation of output for power supply is done by control loop inside RT controller. Using monitored data, RT controller decides the control action and modifies the control parameters. This parameters are communicated to all the SPS modules immediately. In ICHVPS, RT controller uses asynchronous method to handle it.

Generally, asynchronous method is suitable for slower control loops or control applications where the synchronization of I/O to the control algorithm is not critical. Here, in RT controller, control loop is very fast and critical for the HVPS operation, but it is not time bound and is soft real time. For example, if the acquisition loop is ran at a higher rate than the control loop on RT controller, most cases do not require synchronization.

G. High speed events like Trips, Faults and Alarms

As discussed, critical High speed events like Trips and Faults are being handled by FPGAs. If FPGAs act as interface medium only and pass on the data to RT controller, it might delay the response (few micro seconds) even when using optimized algorithm in RT. Possibilities of crash at Driver API (Application Program Interface) level also cannot be denied when using this method. To make system more reliable high speed events are assigned to FPGAs bypassing RT controller. Some of the signals have requirements for response time better than few microseconds and these are also processed at FPGA level and only end result is communicated with RT controller.

SUMMARY

There are several methods of data transfer [7][8] when working with RT System. Depending on interfacing devices and application, data transfer method has been chosen. Data transfer between FPGA and RT includes critical and time sync requirements, while Data transfer within RT system mostly involves local variable and RT buffers [7]. For HMI or remote control, RT controller has to handle TCP/IP or channel access server. Data transfer between RT and FPGA shall be most optimized, as it can use the CPU resources on both side and in turn can affect RT algorithms performance. Different optimizing technics like DMA Transfer and FIFO transfer helps to minimize the resource utilization. As DMA transfer and FIFOs in the system also are limited resource, a balanced approach has to be formed. Choosing suitable method of data transfer to specific requirement optimizes resources in terms of FPGA resources and CPU utilization. If not optimized, FPGA might fail to compile the implemented functions. Whereas, CPU resource optimization ensures smooth operation during peak requirements and gives enough margin for future amendments or modifications.

REFERENCES

[1] R. Dave, “Instrumentation Bus Systems and Communication Methods for DNBPS Fast Controller," DAE-BRNS National Symposium on Nuclear Instrumentation (NSNI 2013)

[2] J Kourunen “Suitability of a PXI platform for an electrical impedance tomography system,”, Published 11 December 2008 • 2009 IOP Publishing Ltd Measurement Science and Technology, Volume 20, Number 1

[3] PXI: The Industry Standard Platform for Instrumentation, http://www.ni.com/white-paper/3889/en/

[4] Data Streaming Architectures in PXI Systems. http://www.ni.com/white-paper/3221/en/

[5] R. Dave, “Communication Methods for Siemens S7-300 PLC with Real Time PXI Controller” 10th IAEA Technical Meeting on Control, Data Acquisition and Remote Participation for Fusion Research 2014

[6] H. Dhola, “Using Voltage to Frequency conversion for close loop control in HVPS," DAE-BRNS National Symposium on Nuclear Instrumentation (NSNI 2013)

[7] Transferring Data between the FPGA and the Host (FPGA Module). http://zone.ni.com/reference/en-XX/help/371599F-01/lvfpgaconcepts/pfi_data_transfer/.

[8] Data Communication Methods in LabVIEW http://zone.ni.com/reference/en-XX/help/371361L-01/lvconcepts/data_comm/.