HARDWARE AND SOFTWARE DESIGN FOR THE DSP BASED LLRF CONTROL

V. Ayvazyan, G. Petrosyan, K. Rehlich, S.N. Simrock, P. Vetrov, DESY Hamburg

Abstract

In the RF system for the VUV-FEL Linac each klystron supplies RF power to up to 32 cavities. The requirements for gradient and phase stability are therefore of the order 10^{-4} and 0.1 degree respectively [1]. To meet this requirements a DSP digital LLRF control system was designed. The basic part of this system is 6U-VME standard board which contains TMS320C6701 floating-point DSP (Texas Instruments Inc.) and 8 Gigalink channels [2].For communication between DSP and ADC/ DAC boards a Gigalink mezzanine board was developed [3]. BIOS, SOLARIS 2.X driver and C++ library were written for the DSP board. LLRF control algorithm was implemented in DSP firmware and DOOCS server [4]. Hardware and software design of the DSP based LLRF control are presented.

INTRODUCTION

A DSP system was used for TTF1. But new requirements [1] for LLRF control of TTF2 and in future for VUV-FEL brought us to the idea of designing a new hardware. It was decided to create a new DSP board based on TMS320C6701 DSP (Texas Instruments inc.) and use it with FastADC board [5] (eight 14 bit channels onboard) and a multi-channel DAC board [6] (eight 14 bit channels onboard).

As a communication interface between these boards and the DSP module a Multigigabit transceiver TLK-2501 (Texas Instruments inc.) was chosen. All eight transceivers are implemented for DSP module onboard. Additionally for ADC and DAC boards a mezzanine module was designed based on the same transceivers to interface with the DSP module. The DSP module is implemented as a VME card.

In our system the modules are controlled by SPARC VME CPUs running Solaris OS. In order to communicate to the DSP module a Solaris device driver was written.

The algorithm of LLRF control is implemented in firmware of the DSP module. The control of the DSP module is done by a high level DOOCS (Distributed Object Oriented Control System**)** dsp-server.

In designing the new LLRF software an appropriate attention was paid to provide a backward compatibility with old software that is still used.

HARDWARE

DSP board

Figure 1: DSP module.

It is a full-size VME board (6U) [2]. This VME Board is intended for use in VME crates. Figure 2 shows the basic functional block diagram.

A booting program is stored in the flash memory of the DSP board. At start up or after reset this program initializes the DSP internal registers and DSP Interrupt Service Table (IST) [8]. After the initialization it enters an idle loop.

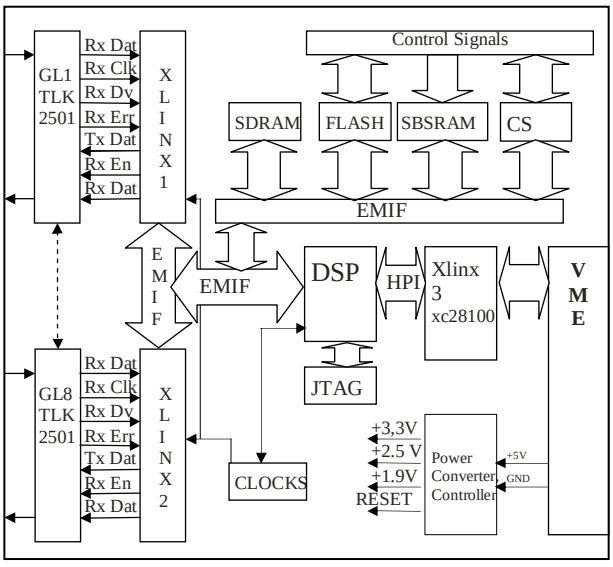

Figure 2: DSP module block diagram.

Key Features of the DSP board:

- TMS320C6701 floating-point DSP – 164MHz (max 167 MHz);
- 1MB of 133-MHz synchronous burst static random-access memory (SBSRAM);
- 16MB of 143-MHz synchronous dynamic RAM (SDRAM);
- 1MB of 10 MHz FLASH ROM;
- Internal EMIF interface (82 MHz);
- 8 gigabit serial links;
- Embedded JTAG emulation via external XDS510 support;
- Access to all DSP address space from the VME bus via Host Port Interface (HPI);

External Interfaces

The DSP module has the following external interfaces:

- VME slave interface (5 V, A16/D16);
- Two 64-pin VME connectors;
- Eight HSSDC2 7-pin connectors for ultra fast serial GigaLink;
- 14-pin external JTAG header;
- 2x8-pin jumper to select the VME Base Address
- Lemo connector to connect an external signal to the DSP interrupt input INT_5[7];
- Lemo connector to connect an external signal to the input of the DSP timer 1[7];
- Eight 2-pin headers for enabling disabling of GigaLink lines;
- Eight-pin headers are test points of FPGA.

GIGALINK mezzanine module

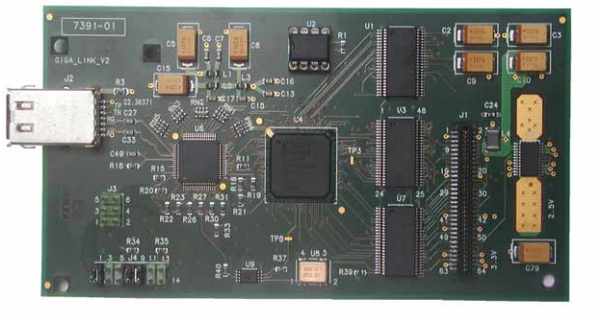

Figure 3

It is a small daughter-board, which provides an interface between a high-speed serial data line (G-Link) and a Carrier Board (ADC [5], DAC [6] or Evaluation Board [8]).

Mezzanine Board (M.B.) includes (Figure4):

- Controller of M.B. - XILINX FPGA (XC2S50 - SPARTAN2 family);
- Multigigabit transceiver (TLK-2501, 1.6 -2.5 Gbps);
- Bus drivers;
- 82MHz oscillator;
- Clock driver;
- High Speed Serial Data Connector – HSSDC2;
- Carrier- Mezzanine connector IEEE 1386 SMT, Dual Row, Vertical Stacking Receptacle;
- Some discrete components (Resistors, capacitors, beds).
- Jumpers – J3 and J4;

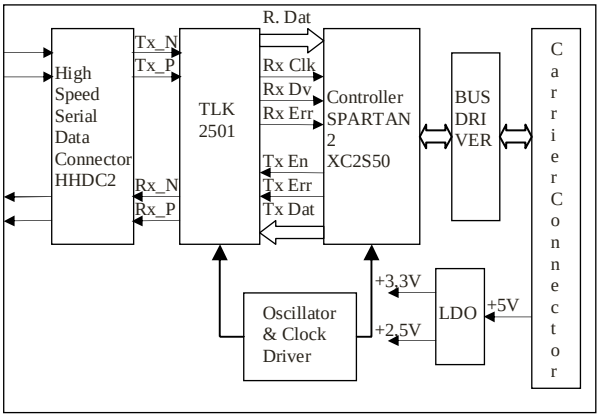

Figure 4: Gigalink mezzanine module block diagram.

SOFTWARE

Driver and Library

The DSP board driver is written as a DDI/DDK Loadable Driver for Solaris 2 which presents the DSP module as a character device. The driver controls the DSP board and allows read/write data from/to DSP board memory including its internal memory. A user process can use open (), close (), write (), read () and ioctl () system calls of this driver. The driver can also redirect a hardware interrupt to as a UNIX signal.

A specially written C++ library was written to simplify the interface to the driver. The library provides functions to read/write integer or floating-point data and tables from/into the board or DSP memory, load and start executable program for DSP, get information about the board and etc. [9].

Firmware for LLRF control

The block diagram of the algorithm for LLRF control is shown in figure 5. The DSP after reading the input data creates the vector sum, filters it by means of Kalman filter and calculates an error signal. Multiplying that with the gain the DSP produces the control signal by adding the result of multiplication to the feed forward signal. This control signal drives a Vector Modulator producing the amplitude and the phase of the klystron's output wave. The basic formula is: [10]

$$\begin{pmatrix} V_{ctrl,x} \\ V_{ctrl,y} \end{pmatrix} = \begin{pmatrix} G_x & 0 \\ 0 & Gy \end{pmatrix} \bullet \begin{pmatrix} V_{set,x}(t_k) - \overline{V_{frx}}(t_k) \\ V_{set,y}(t_k) - \overline{V_{fry}}(t_k) \end{pmatrix} + \begin{pmatrix} V_{FF,x}(t_k) \\ V_{FF,y}(t_k) \end{pmatrix}$$

Where x and y are the real and the imaginary parts of the signal. (G_x, G_y), $(V_{set,x}, V_{set,y})$ and $(V_{FF,x}, V_{FF,y})$ are corresponding individual values of gain, setpoint and feedforward signals at time Tκ. These values are calculated by the dsp-server and stored in tables. These tables can be loaded from files or directly from another application by using the DOOCS library. (V_{frx}, V_{fry}) is produced by Kalman filter from the vector sum of up to 32 cavities. For faster operation the firmware was written completely in assembly language using Texas Instruments' Code Composer Studio. That improved the operation of the firmware so significantly that the control algorithm for 32 cavities does the calculation in 1us.

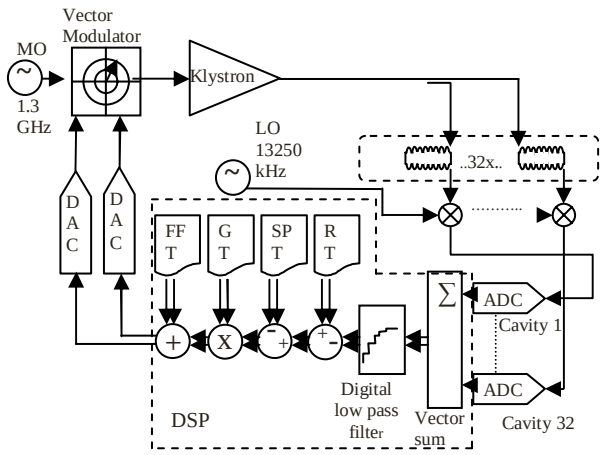

Figure 5

The figure 6 presents an oscilloscope picture showing the DSP system time latency for controlling 32 cavities. As shown in the picture the time latency between event trigger of ADC and the output signal to the Vector Modulator is 3.3 us.

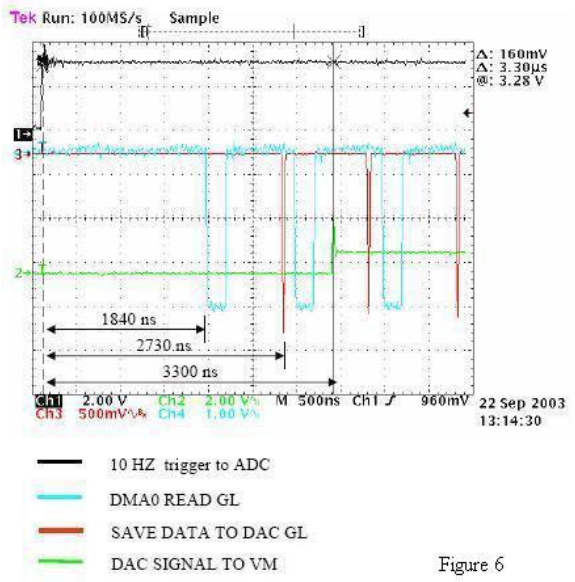

Figure 6

DSP server for LLRF control

The dsp-server is a DOOCS (**Distributed Object Oriented Control System)** server [10]. A client application can access the server using DOOCS client API library. The server controls the DSP board and the DSP LLRF control firmware.

There are 3 basic subsystems in the dsp server:

1. The DSP state sense and Start/Reconnect
2. Load and Start a program in the DSP module
3. The DSP LLRF firmware control.

Start/Reconnect is activated at the start of the server and checks the state of the DSP firmware. Depending on the result, the server decides either to read the configuration and data from the DSP board or to load them from a configuration file.

Load and Start allows loading an executable file into the DSP and starting it. The executable file must be in a COFF format created by Texas Instruments Code Composer Studio and must satisfy to some special requirements.

The DSP LLRF firmware control allows to change the data and tables of the control algorithm and to enabling/disabling part of the algorithm.

The GUI used in TTF2 is a DOOCS DDD [11] program. In figure 7, the main control window of the LLRF control for one klystron is shown. Some expert windows can be activated by clicking on the appropriate buttons provided by the main window (some expert plots are already shown on the left side of the figure).

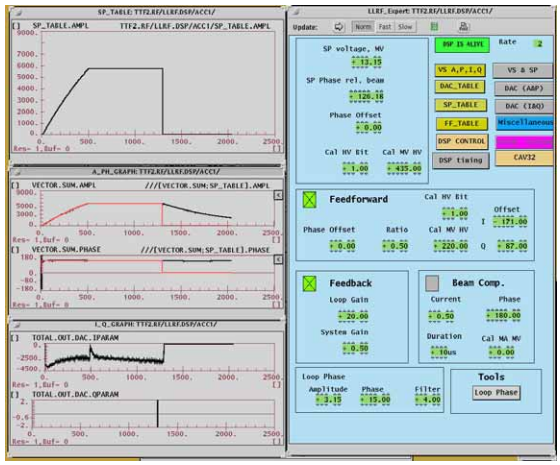

Figure 7

Application Software

A set of generic and specially devoted programs provide the tools for the operators to control the RF system. Some of them were created in MATLAB, others are in-house developed DOOCS client applications. The application software includes automated operation of the frequency and waveguide tuners, calibration of the vector-sum, phasing of cavities, and adjustment of various control system parameters such as feedback gains,

feed forward tables and setpoint correction during cavity filling. Extensive diagnostics inform the operator about exceptions, cavities requiring manual tuning, and an excessive increase in control power.

CONCLUSION

A digital RF control system has been developed to control the vector sum of the accelerating field of a group of superconducting cavities powered by a single klystron. The RF control system is realized as a driven feedback system and has been installed and used in the 5 RF stations at DESY VUV-FEL Linac and at FNAL A0 Injector, which are including a single cavity systems and up to 16 cavities systems as well [12]. The goal to provide a constant accelerating field in order to minimize the energy spread has been successfully reached. The major advantages of the digital system are the built in diagnostics, the flexibility and the stability which are essential for the extension of the system. It has, however, to be demonstrated that the operation of 32 cavities driven by one klystron yields similar results. The digital boards are generic and flexible enough to be usable for a variety of control and data processing applications.

REFERENCES

[1] V. Ayvazyan, K. Rehlich, S.N. Simrock, "Requirements for RF Control of TTF2 FEL User Facility", Proc. PAC'03, Portland, Oregon, p. 2342,2003

[2] P. Vetrov, V. Yakimchuk, G. Petrosyan "VME DSP module with 8 GigaLink interfaces", http://tesla.desy.de/LLRF/documents/dsp_15.pdf

[3] P. Vetrov "GigaLink Mezzanine Board" http://tesla.desy.de/LLRF/documents/Mezz_7.pdf

[4] G. Grygiel, O.Hensler, K. Rehlich, "DOOCS: a Distributed Object Oriented Control System on PC's and Workstations", ICALEPS 97, Beijing, 1997

[5] J. Andruszkow, P. Jurkiewicz, F. Tonisch , "8-channelFastADC with 14 bit resolution" http://tesla.desy.de/doocs/hardware/8ch_10MHz_ADC_manual.pdf

[6] "Multichannel DACs board" http://tesla.desy.de/doocs/hardware/

[7] TMS320c6000 CPU documentation, Texas Instruments Inc. SPRU189E, SPRU190D.

[8] V. Yakimchuk "The Evaluation board for the mezzanine gigabit-link testing", http://tesla.desy.de/LLRF/documents/EvB2_0_Eng.pdf

[9] G. Petrosyan, K. Rehlich, P. Vetrov "Driver and library for DB67 DSP board" http://tesla.desy.de/doocs/hardware/dspdrlb.pdf

[10] G. Petrosyan "DSP program and DSP server" http://tesla.desy.de/doocs/server/dsp/DSPprsvr.pdf

[11] ddd: The DOOCS Data Display http://tesla.desy.de/doocs/ddd/ddd_intro.html

[12] V.Ayvazyan, G.Petrosyan, K. Rehlich, S.N.Simrock, P.Vetrov, "RF Control System for the DESY VUV-FEL Linac", Submited to PAC'05, Knoxville, Tennessee, USA (2005)