A DSP-based Control System for the ISAC Pre-Buncher

M. Laverty, K. Fong, and S. Fang
TRIUMF 4004 Wesbrook Mall, Vancouver, B.C., Canada V6T 2A3

Abstract

A DSP-based control system has been designed for use in the ISAC pre-buncher. This system uses three harmonics of the 11.667 Mhz fundamental to synthesize a sawtooth waveform for driving the pre-buncher cavity. A total of four DSPs are used, with a single 24-bit DSP providing both in-phase and quadrature control of each harmonic. The complete system including low-level rf components is housed in a VXI rack. The supervisory control consists of a Windows-based server, which broadcasts system status using datagrams, and listens on control commands via TCP. JAVA-based control clients, running either as standalone applications or as WWW applets, interpret these messages to provide control and display of the system operating parameters.

1 Introduction

This paper describes the hardware and software design of the pre-buncher control system. The conceptual and rf design were dealt with in an earlier paper [1]. Like the current TRIUMF cyclotron control system [2] and many similar systems elsewhere, it is built on a VXI platform.

This requires the development of a modular design, and permits the inclusion of low-level rf modules together with the analog and digital control components.

This system differs from earlier versions at TRIUMF in that some system bandwidth has been traded off to yield a higher level of integration. Also, 24 bit DSPs are used vs. 16 bit ensuring that computation error remains well below the system noise floor.

The operator/system interface software is also a completely new design. As outlined above, it is web-based, and inherently offers the advantages of remote access, monitoring, and control.

2 System overview

A block diagram of the system is shown in Figure 1. The 11.67 MHz reference is provided by a digital frequency synthesizer. This reference frequency is fed to a pair of intelligent phase lock loops. These loops multiply the reference by factors of two and four in the case of the first loop, and three in the case of the second. They are basically analog phase-lock-loops, but feature microprocessor-based switching.

This permits opening and closing the loop, holding the VCO frequency in the event of loss of the reference signal, and a smooth return to locked operation when the reference is restored.

The I/Q modulators, in conjunction with their respective DSP controllers provide fine phase adjustment and control. The modulator output signals are routed through buffer amplifiers to provide isolation. Bandpass filters then remove harmonics of the four frequency components. Following the filters, delay lines are employed in each of the harmonic signal paths to provide coarse phase adjustment of approximately 180 degrees. A combiner sums the resulting signals for amplification in a two stage MOSFET power amplifier. A balun provides a 1:9 step-up to the required cavity voltage.

A second balun provides a cavity feedback signal which is split four ways and bandpass filtered at each of the four component frequencies. The signals are fed to four I/Q demodulators and to the controlling DSPs, completing the loop.

3 The DSP modules

These consist of two modules, each containing two DSPs. The two DSPs share a common bus interface. A single DSP provides I and Q control for one of the component frequencies. DACs provide programmable references for the I and Q inputs. The error signals resulting from the analog subtraction of the references from their respective inputs are then digitized via a 12-bit ADC and fed to the DSP. This approach minimizes the dynamic range requirement and resulting quantization error on input to the DSP.

The DSP architecture incorporates a high speed local bus for communication with peripherals such as DACs and ADCs. It also includes a serial bus which is used for slower peripherals such as the setpoint DACs. For interfacing with the VXI bus and system controller the byte-wide host interface is used. 512 words of on chip program RAM are available as well as two 256 word coefficient/data memories, all of which can be accessed in a single instruction cycle. Also available is a small boot ROM which serves to load the program memory from an external flash EEPROM at powerup. For this application, the on chip program memory proved to be too small, and the code had to be partitioned into two overlays. The first performs all the system initialization, while the second includes all the control code. The hardware is capable of having new control code downloaded directly into the DSP from the VXI bus, or of reprogramming the flash EEPROM and rebooting from there. A separate microprocessor is used to perform VXI housekeeping tasks as well as providing readback of module input and output levels independently of the DSPs.

The DSPs implement a basic PID algorithm for the I and Q channels. As well as implementing this algorithm, the firmware provides continuous readback of the I/Q error signals to the VXI controller. It also provides an output limiting function to prevent overdriving the PA. This is implemented using the DSP saturation logic as follows:

- For an output value y and a limiting parameter b -
  y = saturation_value(y + a)

This provides the upper limit. To produce an output that is also lower bounded the above y value is modified as follows -

y = [saturation_value(y -2a)] + a

If the limiting parameter and/or the output are small enough the above processing has no effect and the output is unchanged. If not, the limiting parameter provides symmetrical upper and lower limits which may be varied between 0 and full output.

4 Control software

An embedded PC provides supervisory control to the regulating feedback loops. This includes switching the rf on or off, selecting the PID coefficients for the feedback loops, and setting the maximum allowable drives and the regulating amplitudes and phases. It enables the regulating loop, as well as monitoring the loops' status and voltages at various locations inside the loops. It also provides communications to a central control system for remote operation. The supervisory code is a Windows-based application program written to provide interfaces between the hardware modules, and between the hardware and the local/remote operators. The multi-tasking control software is written in C++, using Borland's Object Windows Library.

The Graphical User Interface (GUI) has two sets of windows. One set parallels the hardware architecture, such that objects in the windows represent hardware modules. This architecture is useful during initial setup as well as during diagnostics, where hardware faults can be traced directly by the values indicated in the window. An example of this is the voltage amplitudes of the I and Q (rectangular) components of one of the frequencies.

The other set of windows is used during normal operation. Here the GUI is geared towards presenting to the operator only the necessary controls and status for stable operation, and not to clutter the operator's screen with unneeded information. In this set of windows the amplitude and phase (polar) components of the frequency are displayed.

5 Central control system interface

The rf control system is connected to the overall control system via a private ethernet. Using Winsock 1.1, the system provides two sockets for communications, one as a User Datagram Packets (UDP) server and the other as a Transmission Control Protocol (TCP) server. A JAVA-based client with the authorized IP address, running as a standalone application or as a WWW applet, can be connected to the control server and have access to the control and status of the system.

The UDP server responds to an external UDP clients request by sending the control system status information. Each response packet consists of ASCII strings containing the status of all the frequency components, which include the setpoints, drive and feedback voltage, as well as the feedback regulating loop status. For security reasons, the UDP client is allowed to read the status only. Since UDP is a connectionless communication channel, exceeding the channel capacity will not cause a breakdown in communication. Any clients can request this information as often as they want.

To change parameters in the control system TCP, which is a more reliable and positive connection protocol, is used. An external TCP client communicates its desire to the control system TCP server. The TCP server upon request connects to a TCP client after checking that client's IP address is within the authorized range. Further password protection can be used if necessary to prevent unauthorized access to the control system. Parameters which can be changed remotely are the on/off controls of the rf, and the amplitude and phase setpoints.

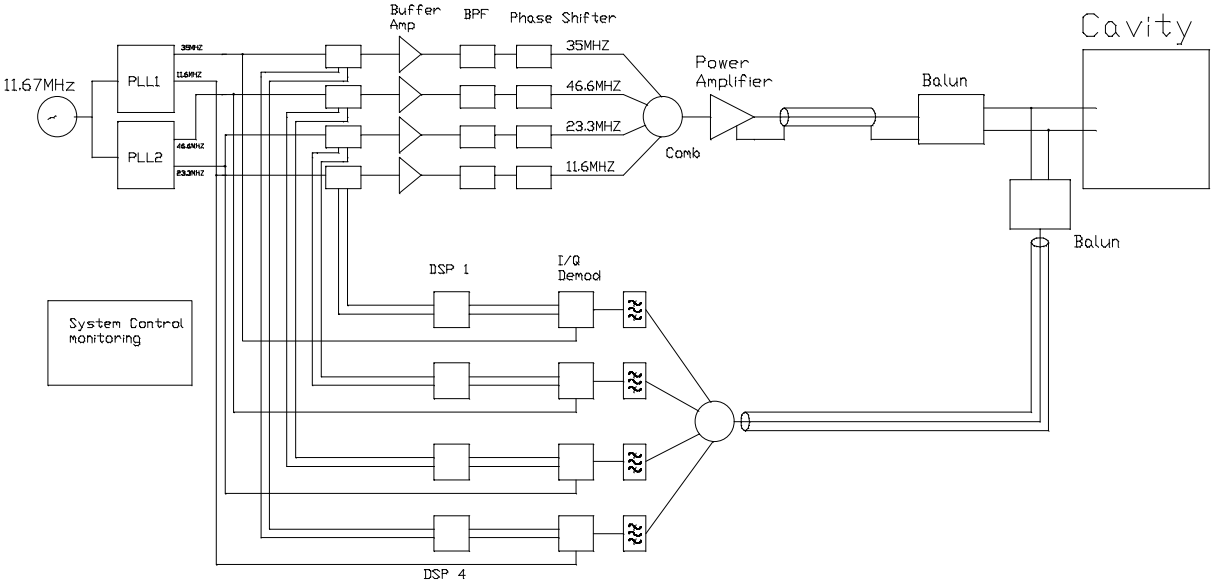

Figure 1 - System Block Diagram

6 Conclusion

To date, the system has been tested in close-loop operation with an actual prebuncher cavity. Stable operation has been achieved with four frequencies operating simultaneously at up to half power. Control bandwidth for all feedback loops is better than 30 KHz. Local control response is virtually instantaneous, but there is a noticeable time lag in response to the remote control. This lag time is on the order of 0.5 second. This is expected to be reduced as faster JAVA compilers and processors become available.

References

[1] K. Fong, M. Laverty, S. Fang, and W. Uzat, "Sawtooth wave Generation for Pre-Buncher cavity in ISAC", Particle Accelerator Conference, Vancouver, May 1997.

[2] K. Fong, M. Laverty, S. Fang, "Operating Experience with the New TRIUMF RF Control System", Proceedings of the Particle Accelerator Conference, Dallas, May 1995, Vol. 4, pp. 2273-5