

# INITIAL PERFORMANCE RESULTS OF THE APS P0 FEEDBACK SYSTEM\*

N. DiMonte<sup>#</sup>, C.-Y. Yao, Argonne National Laboratory, Argonne, IL 60439, U.S.A.

## Abstract

The Advanced Photon Source electron beam exhibits transverse instability when a large amount of charge is present in a single bunch. The P0 feedback system stabilizes the transverse motion of the beam under these circumstances. The initial requirement was to stabilize a single bunch of electrons in the horizontal plane. By implementing the stabilizer in an FPGA and using the parallel processing capabilities provided by this hardware, it is possible to stabilize 324 bunches per turn in both the horizontal and vertical planes. The stabilizer consists of 648 32-tap finite-impulse response (FIR) filters. This paper discusses the challenges in achieving this performance and some issues in interfacing to a Coldfire IOC running RTEMS. Initial test results of the system response are presented.

## INTRODUCTION

The Advanced Photon Source (APS) experiences beam instabilities in both the transverse and longitudinal planes. The P0 feedback system, in its initial version, will correct these instabilities in a bunch pattern that has up to 24 bunches. This is accomplished by using a pick-up stripline, drive stripline, four drive amplifiers, an APS monopulse receiver [1] for front-end signal conditioning and an Altera Stratix II FPGA-based DSP development board coupled with a Coldfire CPU. The Coldfire CPU uses EPICS [2] with RTEMS [3] for all the remote monitoring and control. Figure 1 shows a block diagram of the feedback system [4]. This paper discusses how the FPGA was implemented to achieve high performance correction using 648 32-tap FIR filters.

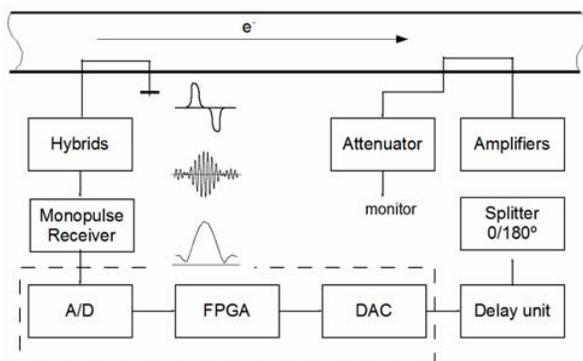


Figure 1: Block diagram of the feedback system.

\*Work supported by U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under Contract No. DE-AC02-06CH11357.

<sup>#</sup>Email: npd@aps.anl.gov

## FPGA IMPLEMENTATION

The algorithm of the P0 feedback system, shown in block form in Figure 2, shows the path of two plane signals being sampled separately. They are sampled with an analog-to-digital converter (ADC) running at 88 MHz, which is a quarter of the APS storage ring frequency of 352 MHz. This sampled data has the DC component removed in the next stage, which is the high-pass filter (HPF) block. The filtered data is then stored in a 32-sample by 324-bunch memory block. For each bunch, 32 samples are presented to the FIR filter block. The next block determines how this bunch data is used before going to the delay block. Once through the programmable delay block, the data is passed on to the digital-to-analog converter (DAC). Each operation is performed at a 271-kHz rate, which is the time it takes for all bunches to go around the storage ring once. Since the FPGA system is sampling at 88 MHz, only 324 bunches can be monitored and corrected. This corresponds to every fourth RF bucket. The front-end signal conditioning now in use reduces this even further so that a maximum of 24 bunches can be stabilized.

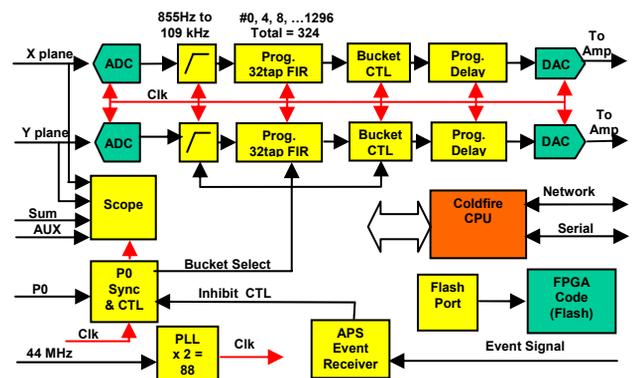


Figure 2: FPGA block diagram of P0 Feedback.

## High Pass Filter Block

The high pass filter (HPF) has a selectable cutoff frequency that is controllable through EPICS. The range can go as low as 855 Hz and as high as 109 kHz, with six other options in between. There is only one HPF for each channel to filter all 324 bunches. The main intent here is to remove the DC component of the signal. The filter is constructed using a 28-bit signed accumulator and a 14-bit signed subtracting block. The subtracting block subtracts the upper 14 bits of the accumulator data from the 14-bit ADC. The result is then passed on to the next block, the FIR filter, and is added back into the accumulator. Using the upper 14 bits of the accumulator

yields an 855 Hz cutoff frequency. By adding an eight-to-one bus multiplexer, with each bus input shifted by one bit closer to zero produces a higher cutoff frequency. For example, when the eighth option is selected, the bus is scaled down to bits 20-7 of the accumulator. This last configuration yields a cutoff frequency of 109 kHz. An optional control bit can be set for each bunch to include or exclude it from the HPF calculation. If the bunch is excluded, then the data value is not added to the HPF accumulator.

### *Finite Infinite Response Filter Block*

The main requirement for the finite-infinite response filter was that it be a 32-tap filter with coefficients that were programmable during operation. The equation below shows the basic concept of the filter; where  $b_i$  is the coefficient value,  $x$  is the analog input value, and  $N$  is 31, for a 32 tap filter.

$$y[n] = b_0 x[n] + b_1 x[n-1] + \dots + b_N x[n-N] \quad (1)$$

Storing the sample data for all 324 bunches is accomplished using a simple dual-port memory block. The memory is organized into 32 sections, each 18 bits wide. The first section is connected directly to the output of the HPF, while the output of this section directs data to the FIR filter and also feeds data into the next memory tap section. This is repeated for the remaining sections, chaining all the sections together. This provides the shifting effect that the FIR filter algorithm needs. The dual-port memory allows the current bunch data to be presented to the FIR filter while the previous bunch data are written to the next section of memory. This is synchronously done to all sections for a given bunch. Once a bunch is selected, all 32 values are presented as the  $x$ -values in the equation above. There are 32 coefficient registers, which are common to all 324 bunches for both the X-plane and Y-plane channels. These values are always presented as the  $b$ -values of the equation above. The memory block proved to be a challenge to implement since it must shift a block of data in one bank while reading from another bank. The timing requirements were extremely tight for this operation.

The Altera Stratix II chip has 288 9-bit digital signal processor (DSP) blocks available. Two blocks were combined to create a single 18-bit signed filter tap, for a total of 64 DSP blocks per channel. These blocks are used to perform the multiplication portion of equation (1). For each bunch, the data is presented to the  $x$ -inputs of the DSPs, and the coefficient values are applied to the  $b$ -inputs. The multiplier blocks are fully pipelined so a new product is available at each clock cycle. An extra clock cycle is used to divide the final result by the calculated coefficient multiplier. The coefficient multiplier is calculated in the device support when a new set of coefficients are loaded. The EPICS device support finds the largest coefficient value and tests whether it is 0.5 or greater. If not, this value is doubled and rechecked for a

maximum of seven iterations. Once the number of iterations is determined, all the other coefficients are multiplied in the same manner. The number of iterations is stored in a register in the feedback system that is used to account for the effects of the device support. This multiplier provides many of the benefits of floating point computation by increasing the effective bits in the FIR calculation, thus providing higher performance from the FIR filter.

With the combined FIR filters from both channels, 648 (324 x, 324 y) bunches are processed during a 3.68- $\mu$ s time frame. This time is equivalent to one revolution of the storage ring beam. The computation speed is approximately  $6 \times 10^9$  multiply-accumulate operations per second.

### *Bunch Control Block*

This block determines if and how a particular bunch will be sent to the DAC. Five options are available.

1. Pass the unaltered FIR filter output to the DAC.
2. Negate the FIR filter data before sending to the DAC. This option can be used for bunch cleaning.
3. Set output voltage to zero.
4. Set DAC to negative full-scale.
5. Set DAC to positive full-scale.

The last two options are used for testing the DAC output response and to create a simple DC modulator.

### *Digital Delay Block*

The programmable digital delay block provides 1 to 512 clock delays, which is more than a complete turn in the storage-ring. The delay is set so that a given bunch receives the correct stabilized kick. Each delay value is an additional 11.36 ns of delay and is common to all 324 bunches. Both channels use the same delay value.

## **PRELIMINARY TEST RESULTS**

### *DAC Output Frequency Response*

The P0 Feedback system uses an Altera DSP Development Board [5]. Concerns about the output are due to the transformer-coupled output of the DAC. To measure the response of the output of the DAC, a test pattern was created using the bunch control block mentioned above. As stated before, using the fourth and fifth options will set the DAC output to their full-scale positive and negative values. The test programmed the first 162 bunches with the negative value while the remaining 162 bunches were programmed with the positive value. This produced a square wave at 271 kHz allowing us to measure the effect of the transformer coupled outputs. Figure 3 shows the response of the output of this step-function drive.

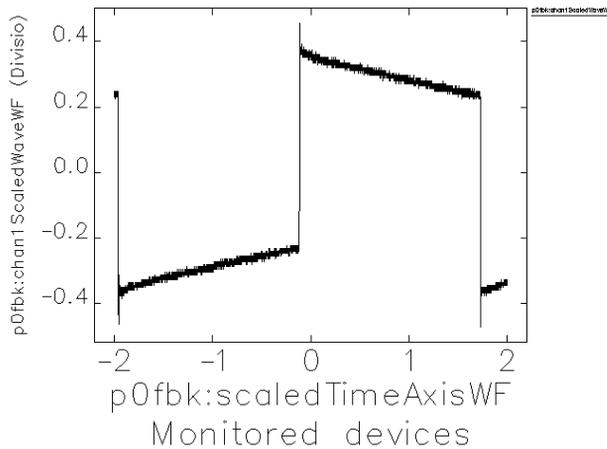


Figure 3: Response of FPGA output with step-function drive.

The output waveform is of the equation (2) form:

$$Y = Y_0 e^{-rt} \quad (2)$$

where  $r$  is the exponential rate. An exponential fit of the waveform produced a rate of  $-2.387 \times 10^5$ . This is equivalent to a low cut-off frequency of 100 kHz. The low cut-off frequency of the DAC output circuits presents no problem for the feedback application as the DC part of the signal is taken out in the beam position readback. The modulation function is necessary as long as the monopulse receiver is being used. The monopulse was not designed for this application, but was designed for the beam position monitor (BPM) system for the APS. As a requirement for the BPM system, the monopulse would take a 1-ns pulse and stretch it to 100 ns. This extended pulse width covers 13.5 bunches that the P0 feedback will not be able to analyze. The digital modulation will be used to maintain a proper output for the remaining 12.5 buckets. The monopulse receiver is limiting this system to its initial use of 1 to 24 bunches. Future development is needed to provide the necessary front end that this system requires. The transform coupled output does not present a problem for this system since we are able to compensate for it.

### Drive Amplifiers

The drive amplifiers have a frequency range of 10 kHz to 220 kHz. It may be necessary to up-shift the FPGA output signal from the base-band to a high harmonic of the revolution frequency. This can be realized with either an analog modulation or a digital modulation technique. Since the bunch control block is capable of alternating the DAC output voltage, the feedback system can produce its own digital modulation function. This is a very nice cost-effective means to implement this function. Since the clock source of the FPGA is synchronized with the

revolution frequency of the storage ring, implementing this modulation from the FPGA is simplified. However, the modulation frequency is limited by the 88-MHz sample rate of the FPGA.

### FUTURE PLANS

The P0 Feedback has great potential for being a bunch cleaner for the APS with a simple addition to the bunch control block to allow the output to be negated. The APS occasionally experiences unwanted satellite bunches. But since the P0 feedback system can only monitor every other fourth bunch, it appears that this system could not help. What will make this possible is not only the negate option in the bunch control block, but the phase lock loop (PLL) circuit inside the FPGA. This PLL can be reprogrammed through EPICS to shift the clock phase enough to monitor and dampen out the unwanted satellite bunch. This has not been tested yet, but as soon as all the hardware is installed for this project, it should only be a quick mode change through EPICS to confirm this capability.

### CONCLUSION

The P0 feedback system has proven to be very flexible in that it has been easy to adapt when the need arose, as in the addition of the digital modulation add-on. The preliminary test data has shown some promise that this system can stabilize 324 bunches in both the horizontal and vertical planes at the APS and possibly function as a bunch cleaner.

### ACKNOWLEDGMENTS

The author would like to acknowledge Eric Norum for the many helpful discussions, his work, and support in both RTEMS and ASYN. This contributed greatly to the success of this project.

### REFERENCES

- [1] R. Lill, G. Decker, O. Singh, "Advanced Photon Source RF Beam Position Monitor System Upgrade Design and Commissioning," DIPAC 2001, 180 (2001); <http://www.jacow.org>
- [2] EPICS, <http://www.aps.anl.gov/epics>.
- [3] RTEMS, <http://www.aps.anl.gov/epics/base/RTEMS.php>.
- [4] C.-Y. Yao, E. Norum, N. DiMonte, "An FPGA-Based Bunch-to-Bunch Feedback System at the Advanced Photon Source," PAC07, 440 (2007); <http://www.jacow.org>
- [5] Altera DSP Development Kit with Stratix II, <http://altera.com/products/devkits/altera/kit-dsp-2S60.html>.