A BUNCH TO BUCKET PHASE DETECTOR FOR THE RHIC LLRF UPGRADE PLATFORM*

K.S. Smith#, M. Harvey, T. Hayes, G. Narayan, S. Polizzo, F. Severino
Brookhaven National Laboratory, Upton, NY 11973, U.S.A.

Abstract

As part of the overall development effort for the RHIC LLRF Upgrade Platform [1,2,3], a generic four channel 16 bit Analog-to-Digital Converter (ADC) daughter module was developed to provide high speed, wide dynamic range digitizing and processing of signals from DC to several hundred megahertz. The first operational use of this card was to implement the bunch to bucket phase detector for the RHIC LLRF beam control feedback loops. This paper will describe the design and performance features of this daughter module as a bunch to bucket phase detector, and also provide an overview of its place within the overall LLRF platform architecture as a high performance digitizer and signal processing module suitable to a variety of applications.

INTRODUCTION

In modern digital control and signal processing systems, ADCs provide the interface between the analog and digital signal domains. Once digitized, signals are then typically processed using algorithms implemented in field programmable gate array (FPGA) logic, general purpose processors (GPPs), digital signal processors (DSPs) or a combination of these. For the recently developed and commissioned RHIC LLRF Upgrade Platform, we've developed a four channel ADC daughter module based on the Linear Technology LTC2209 16 bit, 160 MSPS ADC and the Xilinx V5FX70T FPGA. The module is designed to be relatively generic in application, and with minimal analog filtering on board, is capable of processing signals from DC to 500 MHz or more. The module's first application was to implement the bunch to bucket phase detector (BTB-PD) for the RHIC LLRF system. The same module also provides DC digitizing of analog processed BPM signals used by the LLRF system for radial feedback.

HARDWARE ARCHITECTURE

The ADC daughter module hardware can be divided into two primary sections: 1) The common digital "back end" which shares the same architecture as all modules designed for the LLRF Upgrade Platform, and 2) The custom "front end", which in this case comprises the ADC chips and support components. The overall daughter architecture and the digital back end are described elsewhere in these proceedings [2].

The ADC front end is straight forward, comprising the ADCs, the ADC analog buffering, and a local PLL producing sampling clocks. Buffering is designed to support jumper selectable passive AC coupling, or active AC or DC coupling of signals providing gain if necessary. It is intentionally designed to be as wideband as practical, with provision for local filtering but assuming that analog filtering is primarily implemented off board. The ADCs are interfaced to the FPGA using parallel LVDS.

Clocking is provided by an AD9510 PLL and clock distribution chip, typically referenced from an ultra low phase noise 100 MHz system clock provided by the LLRF platform carrier [2]. The PLL can also be referenced from either the FPGA or an external clock, or bypassed if desired. Multiple clock divider outputs permit independent adjustment of divide ratios and phases.

A custom analog filter is implemented external to the module. The filter is based on a design from the original RHIC BTB-PD. The bunch signal in RHIC is derived from a resistive gap wall current monitor, and can vary significantly in amplitude (factor of 10 or more) and width (20 ns at injection to 3 ns at transition) over the course of a ramp, making it impractical to digitize directly at 160 MSPS. The filter topology is shown in Figure 1. It is designed to generate a 3 period sine-wave like pulse from each bunch passage, where the sine-wave like output has a 40 MHz period with a phase determined by the bunch. The final output is filtered by a low pass filter to further smooth it, but designed with a low Q impulse response which decays prior to the next bunch passage. The system is thus capable of resolving bunch by bunch phase data at the RHIC bunch frequency of 9.383 MHz. The 40 MHz period for the filter output waveform was chosen because it was optimal for both bunch by bunch resolution, and for IQ sampling at the maximum 160 MSPS ADC sample rate, as will be seen later.

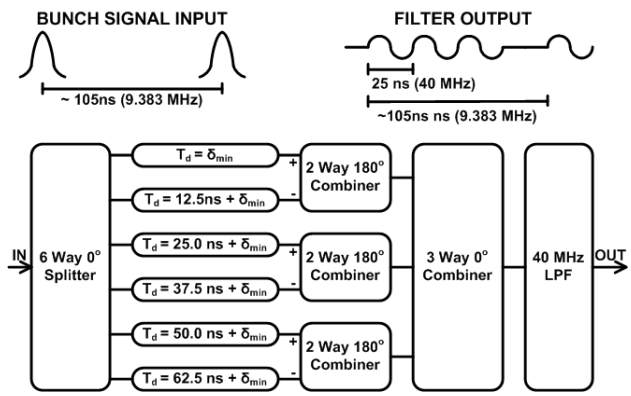

Figure 1: Analog delay line filter.

FIRMWARE IMPLEMENTATION

Once the filtered bunch signal has been digitized, all further processing of the signal takes place in the FPGA. The processing involves several elements: 1) A bunch

*Work supported by Brookhaven Science Associates, LLC under Contract No. DE-AC02-98CH10886 with the U.S. Department of Energy
#ksmith@bnl.gov

Copyright © 2011 by PAC'11 OC/IEEE — cc Creative Commons Attribution 3.0 (CC BY 3.0)

tracking NCO, 2) I,Q down-conversion to baseband, 3) A CORDIC transformation from rectangular (I,Q) to polar coordinates (amplitude and phase), 4) Circular buffering of all bunch phase data, and 5) A phase averaging routine. The firmware also utilizes one of the key elements of the overall LLRF platform architecture, the Update Link (UL) [4]. The Update Link provides synchronous deterministic data and timing to all platform sub-systems, with sub-systems able to both receive and transmit data.

One atypical aspect of the digitizing and processing on these modules, characteristic of how we use the LLRF platform in general, is that although sampling clocks are synchronous to a master 100 MHz system clock, they are nominally asynchronous to the actual RF signals. Sampling is in general not harmonically related to the RF signals. Nonetheless, because all LLRF sub-systems (RF synthesizers and this module in particular) share the 100 MHz reference clock, absolute and deterministic phase relationships can be readily established and maintained. This decision was made early in the development of the platform, driven by local considerations, but systems can also be easily configured to use RF synchronous sampling if desired.

FIRMWARE DETAILS

Figure 2 shows a simplified block diagram of the processing algorithm. The bunch tracking NCO is the first component of the firmware. RHIC fill patterns are always populated as a subset of an overall h=120 (~ 9.383 MHz) available bunch pattern. Therefore, an h=120 NCO is used to provide bunch number tracking as well as the bucket phase reference. Overflow of this NCO indicates the transition to a new bunch. This overflow resets a bunch sample counter and increments a bunch number counter.

Sample processing of the filtered bunch signal begins and ends on start (ST) and stop (SP) triggers, determined from the bunch sample counter. Since the filtered bunch signal is 3 cycles of 40 MHz, one can obtain up to 12 samples per bunch in this sampling period.

Rather than use the standard $F_s/4$ I,Q,-I,-Q technique, we employ a just slightly more complicated quadrature down-conversion using digital cos {1,-1,-1,1} and sin {1,1,-1,-1} mixing sequences. This implementation was chosen in the spirit of preserving maximum signal to noise ratio (SNR), and also with thought to a future more generic NCO based mixing scheme. With that in mind, the mixing is implemented in two DSP Slices, dedicated slices within the FPGA which don't utilize general FPGA fabric logic resources. The slices are configured as multiply-accumulate (MACC) units, cleared by the ST trigger and latched by the SP trigger. The latched I,Q pair is handed off to a CORDIC processor which converts it to phase and amplitude coordinates. Once the bunch phase is ready, the amplitude for the bunch signal is compared to a minimum threshold level to determine if the phase measured is to be considered valid. The valid bit controls a mux which selects whether the measured value or a zero is used downstream. The bunch phase is then appropriately scaled to a 16-bit signed integer spanning +/-180°, and offset as desired.

The SP trigger is also used to latch the phase of the bunch tracking NCO. This provides the h=120 bucket reference phase, against which the bunch phase is measured. It is necessary because the sample clock is asynchronous to the RF frequency, and thus the actual RF phase can be anywhere between 0° and about 21° (the NCO phase increment, 360*9.383/160) when the NCO overflow occurs. This phase is scaled to 40 MHz and the bunch to bucket phase is then just the difference between this value and the bunch phase. This phase is then stored in a circular buffer along with the valid bit.

SYNCHRONIZATION AND DATA DELIVERY

Operating as part of the overall LLRF system at RHIC, the BTB-PD daughter module relies on the Update Link to receive system synchronous timing triggers, and as a low latency data link that receives and transmits to and from any other part of the system.

Two UL timing events are relevant to the BTB-PD. A master RF reset code generated on demand in the Update Link Master (ULM) [4] is used to synchronously establish a deterministic, repeatable phase relationship between all LLRF system NCOs, including the h=120 bunch tracking NCO. An Update Pulse event is also broadcast periodically by the ULM at a precise 10 us interval (every 1000 ticks of the master 100 MHz system clock).

The Update Pulse determines the primary update rate of the LLRF system feedback loops. It is used by the Feedback DSP [3] to trigger the start of each new loop calculation. A 48-bit system timestamp is also placed on the Update Link immediately following the Update Pulse and is received by all sub-systems

The BTB-PD uses an internally time advanced copy of the Update Pulse to determine when to send bunch to bucket phase data to the DSP. Each time this trigger occurs, a state machine calculates the sum of valid bunch to bucket phase values in the circular buffer. This sum along with a count of how many valid phases were in the sum are latched and then transmitted to the DSP via the update link.

The internally advanced Update Pulse is timed to ensure that the data is received at the DSP before the actual Update Pulse occurs. In addition to providing the average phase of all bunches in the ring to the DSP, the same state machine watches for when the circular buffer pointer matches the number of the last bunch injected into the ring. That phase data is latched at the 100 kHz update rate into a register accessible to the Front End Computer (FEC, similar to the EPICS IOC) which is implemented on the platform carrier [1,3]. This data is decimated and buffered, and then passed by Ethernet to the software application responsible for controlling and tuning RHIC injection parameters. Any observed dipole oscillation in the bunch to bucket phase is used to calculate corrections to the injection phase and energy.

Copyright © 2011 by PAC'11 OC/IEEE — cc Creative Commons Attribution 3.0 (CC BY 3.0)

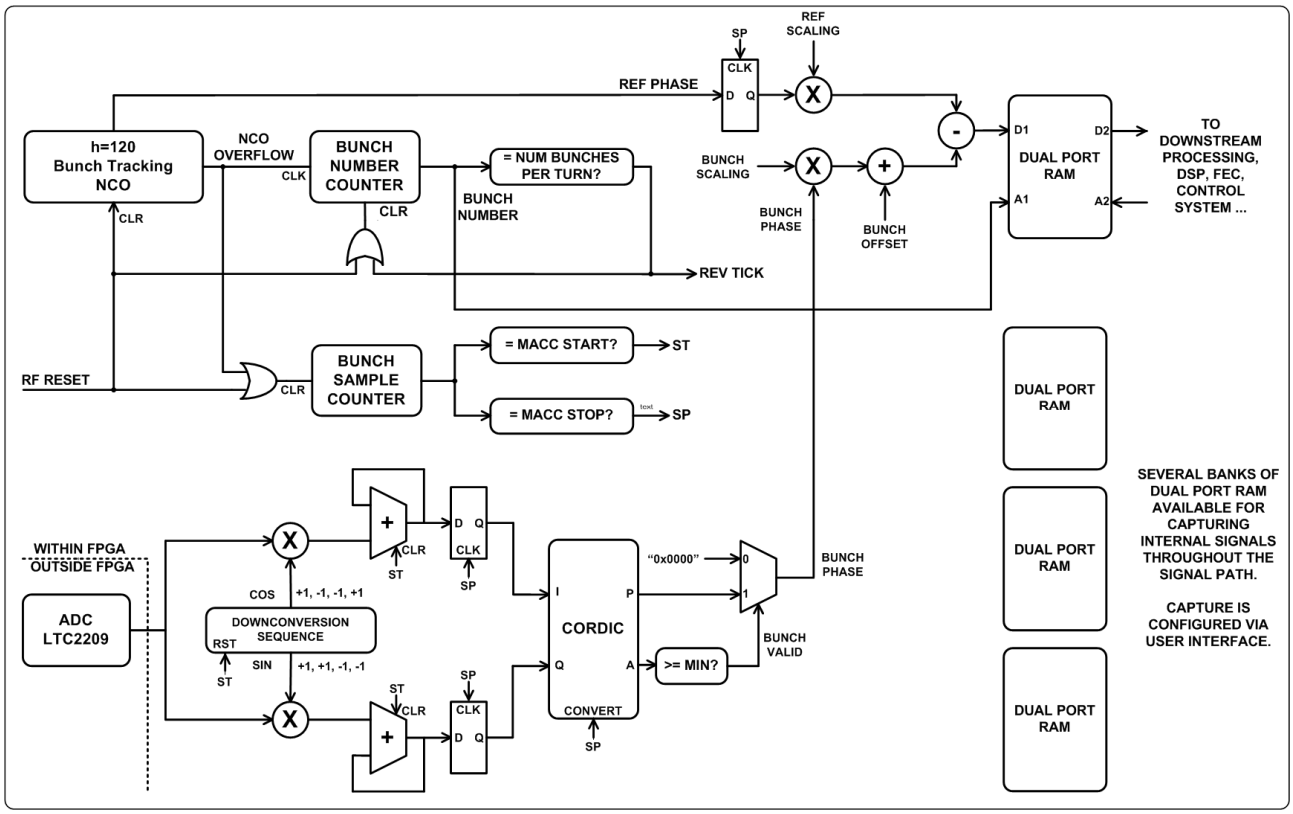

Figure 2: Simplified representation of the BTB PD FPGA processing algorithm.

PERFORMANCE AT RHIC

The new LLRF Upgrade Platform and the BTB-PD daughter module were commissioned at RHIC beginning with Run 10 in December 2010, and have remained operational since that time [5]. A typical plot of the bunch to bucket phase for a single bunch injected into RHIC is shown in Figure 3. This data is very clean. High level software uses the injection data to correct the energy and phase of bunches injected into RHIC, improving the longitudinal match and minimizing the oscillation amplitude. Note the very low synchrotron frequency, characteristic of 9 MHz RF capture. Ticks on the horizontal axis are 0.1 s.

When LLRF feedback loops are closed, BTB phase data transmitted to the system DSP via the Update Link is used to damp coherent synchrotron motion and stabilize the bunches.

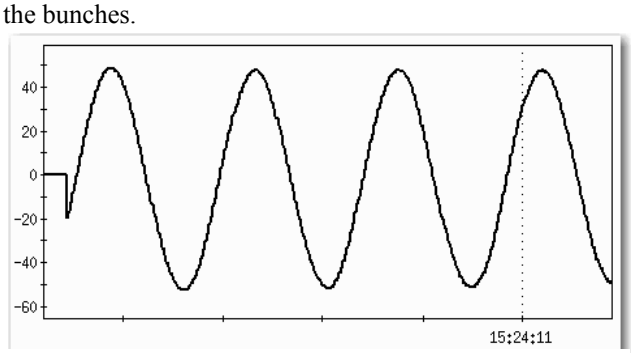

Figure 3: Bunch to Bucket Phase.

Because the ADC daughter module was designed for generic application, the same module also digitizes DC coupled analog processed baseband beam radius signals for use by the LLRF radial feedback loop. At the BNL EBIS injector [6], an entirely different FPGA firmware implementation provides I,Q measurement of RF cavity signals for cavity control feedback loops. This implementation will soon be used at RHIC for the same purpose. Evaluation of the module as an RF BPM processor is expected to occur during the current Run 11 operating period, and many other applications can be imagined.

REFERENCES

[1] K. S. Smith, et al., "Concept and Architecture of the RHIC LLRF Upgrade Platform", these proceedings.
[2] T. Hayes, et al., "A Hardware Overview of the RHIC LLRF Platform", these proceedings.
[3] F. Severino, et al., "Embedded System Architecture and Capabilities of the RHIC LLRF Platform", these proceedings.
[4] T. Hayes, et al., "A Deterministic, Gigabit Serial Timing, Synchronization and Data Link for the RHIC LLRF", these proceedings.
[5] K. S. Smith, et al., "Commissioning Results from the Recently Upgraded RHIC LLRF", these proceedings.
[6] S. Yuan, et al., "Commissioning and Performance of the BNL EBIS LLRF system", these proceedings.

Copyright © 2011 by PAC'11 OC/IEEE — cc Creative Commons Attribution 3.0 (CC BY 3.0)