SSRL Beam Position Monitor Detection Electronics*

J. Sebek, R. Hettel, R. Ortiz, J. Wachter
Stanford Synchrotron Radiation Laboratory, P.O. Box 4349, Bin 99, Stanford, CA 94309-0210

Abstract

As part of a program to improve its orbit stability SSRL is redesigning its detection electronics for its beam position monitors (BPMs) [**?**]. The electronics must provide highly reproducible positional information at the low bandwidth required of an orbit feedback system. With available commercial technology, it is now possible to obtain highly resolved turn by turn information so that this electronic module can also be used to measure beam dynamics. The design criteria for implementation of this system and performance of the prototype is discussed.

1 INTRODUCTION

SPEAR is a 3 GeV electron storage ring used for synchrotron radiation. All of the BPM inputs are multiplexed into one large switching matrix and processed by one set of electronics. We are redesigning the electronics to improve processor speed, dynamic range, and resolution. In addition to providing highly resolved positional information under normal operation, the system must be able to detect low current orbits for injection studies, etc. The design criteria used to determine the processor architecture, RF frequency, IF frequency, and digitization rate have been previously discussed [**?**] [**?**].

Energy	E	3	GeV
Radio Frequency	f_{RF}	358.54	MHz
Harmonic Number	h	280	
Revolution Frequency	f_{rev}	1.2805	MHz
RF Processing Frequency	$2f_{RF}$	717.08	MHz
IF Processing Frequency	$5f_{rev}$	6.4025	MHz
Digitizing Frequency	$16f_{rev}$	20.488	MHz
Nominal Beam Current	I_{nom}	10–100	mA
Number of BPMs		40	
Resolution		10	μm
Channel Isolation		> 60	dB
Detector SNR @I_{nom}	SNR	> 126	dB/Hz
Dynamic Range		40	dB

Table 1: SPEAR BPM Parameters

* Work supported by the Department of Energy, Office of Basic Energy Sciences, Division of Material Sciences.

2 SYSTEM ARCHITECTURE

The main purpose of the BPM system is to provide accurate positional information to our orbit feedback program at a rate that enables this system to control the beam at a closed loop system bandwidth of about 50 Hz. We feel that we can achieve this goal by multiplexing 16 BPMs per processor. Since we also need to output the processor data digitally into the VME crate that will handle the orbit control, we will centralize the back end (digital) processing of the BPM processing in the SPEAR control room.

2.1 Signal Multiplexing

The 717.08 MHz processing frequency that we chose makes it very expensive in terms of either signal strength or cable cost to bring all of the signals from the BPMs in the ring to a central control area for processing. We had considered putting some multiplexing electronics in the ring to minimize the number of cables we would have to run to the processors, but decided against that for two reasons. First, we did not want to have electronics in the ring that would force us to make an access to repair any failures. Second, we wanted to retain the ability to look at individual button signals for possible machine physics experiments, again without requiring a ring access. Our solution is to place four RF processing centers (roughly) equally spaced around the ring. This will make the longest cable run less than 150 feet and allow us to have less than 5 dB signal loss with relatively inexpensive cable.

We had originally planned to multiplex all of the signals at the RF frequency and have only a small number of RF to IF downconverters. However, the isolation performance of the GaAs FET SP4T switches at 717.08 MHz would have required us to insert another switch in series with each module in order to isolate signals from different BPM modules with substantially different signal strengths. Also, in order to multiplex 16 BPMs into a single processor, the signal from each button would have had to pass through four SP4T multiplexers, increasing the insertion loss of each signal by more than 6 dB. We have decided to address both of these issues by building one downconverter for each BPM and multiplexing the output IF signals. Now the attenuators of each downconverter can be used to attenuate the signals that are not being detected by the digitizer to improve isolation between modules. Also, the isolation of the multiplexing switches is greater and the insertion loss lower at the IF than at the RF. In the first implementation of the new system,

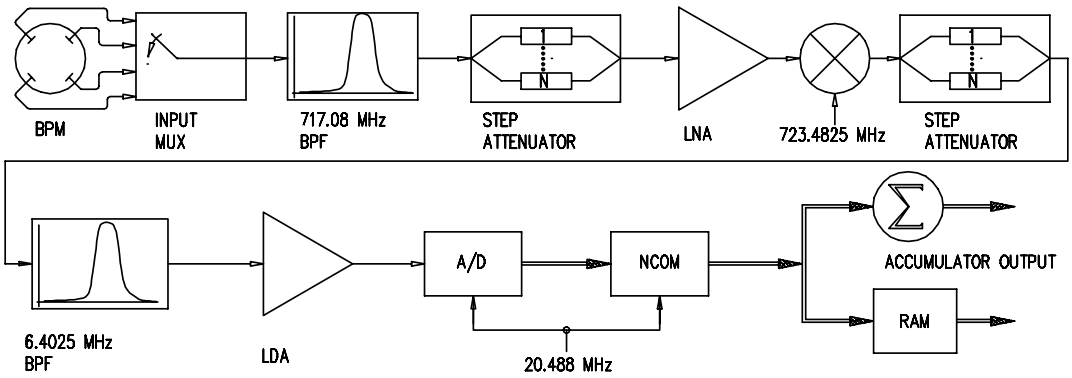

Figure 1: Processor Block Diagram

the IF signals from each BPM will be brought back into the control room for further multiplexing and processing. The modular construction of the system will allow us to move these out to the four processing centers, if so desired.

2.2 Downconverter

The design of the analog portion of our processor, the RF to IF downconverter, remains basically unchanged from our initial design. The minor changes we have made involve the input multiplexer. We have added some inexpensive bandpass filters at the input of the module to limit the out of band power to the system. Here we were worried about possible damage to the switch from a high peak instantaneous voltage during a high beam current run. We also were able to find a SP4T package that provides better than 60 dB channel to channel isolation, so the use of this device simplified our switch implementation.

Each downconverter will be packaged in a 3U high, 5HP wide RF enclosure, and modules will be housed together in a standard crate. They will take as input the signals from the buttons of a BPM and the LO, and output the IF. Control lines will enter the module through a backplane connector.

2.3 Digital Processor

The 6.4025 MHz IF will be digitized and processed by the digital processors. The digital data is output at the clock rate of 20.488 MHz to an HSP45116, a numerically controlled oscillator/modulator (NCOM) which processes exactly 16 samples per revolution and digitally mixes that signal to output once per turn the I and Q of the IF frequency. This turn by turn data is then archived to DRAM on the processor and also accumulated by the AMD29240 microcontroller. The system clocks up to and including that of the NCOM are synchronous with the ring RF frequency. The microcontroller, asynchronous to this frequency, polls the NCOM to collect the data. Once the data has been accumulated for the appropriate number of turns for the current button, it is written to a dual ported RAM, the other port of which sits on the VME bus of the orbit feedback crate.

The BPM processor must avoid aliasing by sampling for periods of time either longer than the fundamental oscillations of the ring, or for periods that are multiples of those oscillation periods. In the case of SPEAR, the slowest frequency, that of the synchrotron oscillation, is typically around 30 Kz. (The fractional parts of the betatron tunes are typically much higher.) Therefore the minimum time for which we will digitize a button is about 33 ms. Between buttons, there is a certain dead time we must endure. The microcontroller must send out the control signals to the remote downconverters. Once those signals arrive, the downconverter must then send back the data based on these signals. Finally, all of our filters must clear themselves of data acquired during the previous invalid setting. The bandwidth of the RF bandpass filter is much faster than the revolution frequency, so it clears its signal quickly. The IF bandpass filter is designed to clear in one revolution period. The NCOM also must clear out its string of stored data, which takes another revolution period. Therefore there will be a four or five period time between buttons during which the data will be invalid. This means the processor will be switching at a 30 Kz rate, yet operating at greater than 85% efficiency.

During the switching time, the microcontroller is able to take care of its control functions. A primary function, of course, is to select the button to be sampled and the proper attenuation settings for that button. These settings make up 16 bits of a 32 bit word. They are sent out to the RF processing stations where they are passed along to the appropriate downconverter module. The other 16 bits shift the phase of the input signal with respect to the NCOM clock. The microcontroller also writes its accumulated data to the dual ported RAM for use by the feedback system, further averages and copies a slower, more accurate number to the dual ported RAM for use by the control system, calculates and updates, if necessary, the appropriate phase of the button to keep the NCOM in phase with the incoming signal, and calculates and updates, if necessary, the attenuation settings of all the buttons in a BPM.

The microcontroller obtains its instructions via high level messages passed to it by the control system over the dual ported RAM. These messages tell it how to acquire data – which buttons to sample, what pattern to sample them in, how long to sample each one – and what to do with the data once it is acquired – either just send back the averaged val-

Signal Cable	LMR-400	Times Microwave	Wallingford, CT
Band Pass Filter	252HXPK-2702F	Toko America, Inc.	Mt. Prospect, IL
GaAs FET Switch	AN002M4-05	Alpha Industries, Inc.	Woburn, MA
Band Pass Filter		Integrated Microwave	San Diego, CA
Digital Attenuator	AK002D4-24	Alpha Industries, Inc.	Woburn, MA
RF Amplifier	INA-02186	Hewlett-Packard	Palo Alto, CA
Image Reject Mixer		Pulsar Microwave	Clifton, NJ
IF Amplifier	AD9618	Analog Devices	Norwood, MA
ADC	AD9022	Analog Devices	Norwood, MA
NCOM	HSP45116	Harris Corp.	Melbourne, FL
Microcontroller	AMD29240	Advanced Micro Devices	Sunnyvale, CA

Table 2: SPEAR BPM Electronics Components

ues or send back the turn by turn data after the processing is finished. After preparing for the sequence, the microcontroller begins its sampling and sets a flag for the VME processor when it is done with the current pattern, telling the VME processor that the valid data in one segment of the RAM is now available. This flag will be the update clock for the feedback system. (The most common message sent by the VME processor is that telling the microcontroller to repeat the last sequence.)

The digital part of the processor now exists as evaluation boards from the various vendors tied together by one (custom) "glue logic" board. It interfaces to the VME bus via another (commercial) "glue logic" board. In the future, these chips will all be integrated into one digital processing board.

2.4 System Timing

The design of the system requires an LO and clocks synchronous with the SPEAR f_{RF}. For our system the frequencies were straightforward to obtain. A frequency divider obtained 6.4025 MHz from the 358.54 MHz master oscillator. A frequency doubler gave 717.08 MHz and a mixer then gave 723.4825 MHz. This LO is split and then distributed to the four RF processing stations, from where it is further distributed to the individual downconverters. The 20.488 MHz digitizing clock is obtained by dividing 717.08 MHz into a clock with close to 50% duty cycle. Further, an f_{rev} clock is generated by dividing f_{RF} and used to synchronize the switching of the multiplexers and the attenuators in the system.

3 SYSTEM TEST RESULTS

We were able to test a prototype version of the analog portion of the processor by parasitically observing signals from one BPM with 55 mA of current in the machine. At this current we required 29 dB attenuation in the signal path to set our IF signal at the 1V level desired by the A/D, so that our measured analog path SNR will hold down to $\sim$2 mA. For a 10kHz RBW, our signal measured $\sim$70 dB above the noise floor at the IF output. Our signal was clean enough to see the amplitude and phase oscillations on the beam. If we need a greater SNR, we can trade off with the current system dynamic range. We saw no evidence of any problems due to processing at a harmonic of f_{rev}. Direct feedthrough of f_{IF} was $\sim$58 dBc, which we feel can be further reduced by addition of appropriate filters. The other noticeable product, probably a mixer IMD was $\sim$65 dBc. Neither of these should be a problem. We have just completed the electrical and mechanical designs for our downconverter module and expect to be able to test them next month.

We have written our first version of the data collection program for the microcontroller and are in the initial stages of writing development programs that we can use as tools for the evaluation and improvement of the analog section of the processor.

4 ACKNOWLEDGEMENT

We would like to thank Max Cornacchia for the encouragement and support he has given this project and Kane Zuo, Ramona Theobald, and Ben Scott for their assistance.