

# RF Noise Suppression for the APS Storage Ring

Kevin Cook

*Georgia Institute of Technology*  
kevin.cook@gatech.edu

Tim Berenc

*Argonne National Laboratory*  
berenc@aps.anl.gov

September 8, 2014

## Abstract

In a synchrotron, amplitude and phase noise in the radio-frequency (RF) system leads to timing jitter and energy fluctuations of the beam, which in turn cause orbital motion in dispersive regions. Electrical components in the RF system inevitably contain 60 Hz harmonic noise from the AC power grid. To suppress 60 Hz related noise, an adaptive notch filter was developed, implemented, and tested using a variety of digital signal processing techniques. Noise at targeted frequencies was reduced by  $\sim 30$  dB to  $\sim 39$  dB.

## 1 Introduction

In an accelerator, the low-level radio-frequency (LLRF) control system is responsible for regulating the amplitude and phase of the electromagnetic fields inside the accelerating cavities. In a storage ring, variations in the amplitude or phase will cause the beam to seek out the new stable fixed point, causing timing jitter and energy fluctuations. This can also lead to orbital motion in dispersive regions. A major source of amplitude and phase noise in the Advanced Photon Source (APS) radio-frequency (RF) system are 60 Hz harmonics originating from the klystron high voltage DC power supply (HVPS). The supply is a 3-phase full-wave rectified DC source which provides the klystron cathode and mod-anode bias voltages. Fluctuations in these voltages cause amplitude and phase modulation in the klystron. Other sources of 60 Hz harmonic related noise are found in the klystron driver amplifier and nearly all electronics which ultimately derive their power from the 60 Hz electrical grid.

This paper outlines a system designed to remove RF amplitude and phase noise at three frequencies: 60 Hz, 120 Hz, and 360 Hz. The system can be expanded to cancel more frequencies,

but will ultimately be limited by the computational resources available. Provisions were included to minimize noise that could potentially be added by the designed system.

In order to remove the noise at the target frequencies, an adaptive notch filter architecture was selected. An adaptive notch filter is superior to a static notch filter because the adaptive filter tracks the frequency of the noise; thereby allowing extremely narrow bandwidths with excellent frequency selection. In order to cancel multiple target frequencies, multiple instances of an adaptive notch filter were implemented in parallel.

### 1.1 The Adaptive Notch Filter

The adaptive notch filter is a special case of an adaptive noise canceller [1]. The general adaptive noise canceller is depicted in Fig. 1. A desired signal,  $d$  is polluted with noise,  $n$ , from a noise source in some unknown way to become signal  $s = d + n$ . A sensor is used to detect a noise reference signal,  $x$ , which is correlated to the noise source and hence  $n$ . The coefficients of the adaptive filter,  $H(z)$ , are adjusted so as to match the filter's output,  $y$ , to the noise  $n$  in a least mean squares (LMS) sense using the noise

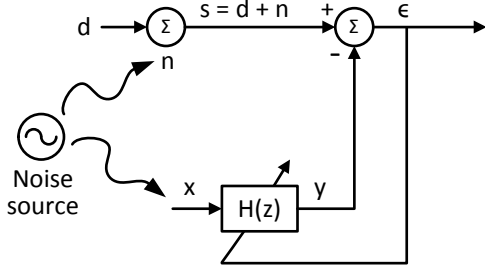


Figure 1: Depiction of general adaptive noise cancelling concept.

canceller's output signal  $\epsilon$ .

The adaptive filter accepts two inputs: the noise reference signal  $x$  and the noise canceller's output signal  $\epsilon$ . The filter weights are adapted using a LMS algorithm [2] which minimizes the noise power of the canceller's output signal. The algorithm estimates the gradient of the mean-square error surface and travels down that gradient to minimize the error. The algorithm is described by the following set of discrete-time equations for digital implementation [2]:

$$\epsilon_k = s_k - y_k \quad (1a)$$

$$W_k = W_{k-1} + \mu \epsilon_k X_k \quad (1b)$$

$$y_k = W_k^T X_k \quad (1c)$$

where  $k$  is the sample time index,  $W_k$  is a vector of length  $M$  filter coefficients corresponding to a digital filter of length  $M$ , and  $X$  is a vector of the last  $M$  values of the noise reference signal. At each iteration, the coefficients are updated by adding a scaled estimate of the gradient,  $\mu \epsilon_k X_k$ , to the previous value of the coefficient. The scaling  $\mu$  is used to adjust convergence time and stability. The coefficients are then used to calculate the output of the adaptive filter,  $y$ . This output is then subtracted from the noisy signal,  $s$ , in an attempt to eliminate the noise.

For noise that has a broadband frequency spectrum, many coefficients are needed to create a filter which is capable of estimating the magnitude and phase response for each frequency contained in that spectrum. However, when the noise is very narrow-band, or consists of a single frequency tone such as a 60 Hz harmonic, the

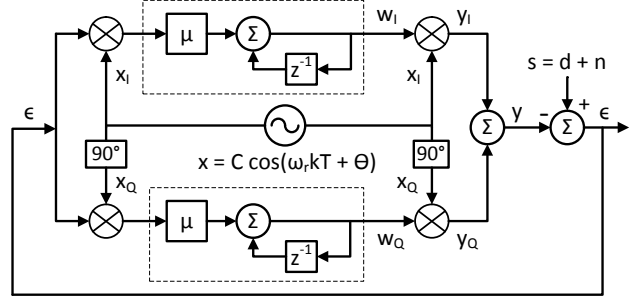


Figure 2: Block diagram of the two weight adaptive notch filter.

filter can be implemented using only two coefficients,  $w_I$  and  $w_Q$ , as depicted in Fig. 2. These two coefficients respectively adjust the magnitude of an in-phase (I) and quadrature (Q) component of the noise reference tone. By adapting these I and Q components, an amplitude and phase of the noise tone can be found which cancels its presence in the system's noisy signal; hence, this yields the name *adaptive notch filter*. With each weight working on its respective I/Q component of the noise reference, there are 2 separate sets of Eq. 1 with Eq. 1b of each becoming an integrator with gain controlled by  $\mu$  as depicted by the dashed boxes in Fig. 2. In practice, the 2-weight filter is limited by how close the phase shifter in Fig. 2 is to 90deg for achieving true quadrature references, binary number precision, and other sources of noise.

Not surprisingly, the adaptive notch filter has the same architecture as a LLRF control system that uses I/Q rectangular coordinates. However, LLRF feedback may use general proportional-integral-derivative (PID) or state-space control for the baseband section (i.e., the dashed boxes of Fig. 2). Furthermore, the input to a LLRF controller's baseband section is the difference between a setpoint and the down-converted, or I/Q, detected signal in order to regulate to a desired setpoint as opposed to trying to achieve a notch with a setpoint of zero.

## 1.2 Analysis of Adaptive Notch Filter

Referring to Fig. 2, we first wish to find the transfer function through the adaptive filter, or

from  $\epsilon$  to  $y$ . Consider a sinusoidal noise reference,  $x$ , sampled with sampling period  $T$ ,

$$x(k) = C \cos(\omega_r kT + \theta) .$$

Shifting this reference by 90deg, quadrature reference signals  $x_I$  and  $x_Q$  are generated

$$\begin{aligned} x_I(k) &= C \cos(\omega_r kT + \theta) \\ x_Q(k) &= -C \sin(\omega_r kT + \theta) . \end{aligned}$$

By Euler's formula, these can each be represented as:

$$x_I(k) = \frac{C}{2} \left[ e^{i\omega_r kT} e^{i\theta} + e^{-i\omega_r kT} e^{-i\theta} \right] \quad (2a)$$

$$x_Q(k) = \frac{iC}{2} \left[ e^{i\omega_r kT} e^{i\theta} - e^{-i\omega_r kT} e^{-i\theta} \right] \quad (2b)$$

where  $i = \sqrt{-1}$ .

The upper portion of Fig. 2, or the in-phase path, will be analyzed first. The in-phase noise reference signal,  $x_I$ , is multiplied by the error signal  $\epsilon$ . Using Eq. 2a for  $x_I$ , the Z-transform of this process is given as

$$\begin{aligned} \mathcal{Z}\{\epsilon \cdot x_I\} &= \\ \frac{C}{2} \left[ \mathcal{Z}\{\epsilon(k) e^{i\omega_r kT}\} e^{i\theta} + \mathcal{Z}\{\epsilon(k) e^{-i\omega_r kT}\} e^{-i\theta} \right] . \end{aligned} \quad (3)$$

Since the Z-transform of  $\epsilon(k)$ ,  $E(z)$ , is

$$E(z) = \mathcal{Z}\{\epsilon(k)\} \equiv \sum_{k=-\infty}^{\infty} \epsilon(k) \cdot z^{-k}$$

then

$$\begin{aligned} \mathcal{Z}\{\epsilon(k) \cdot e^{\pm i\omega_r kT}\} &= \sum_{k=-\infty}^{\infty} \epsilon(k) \cdot (ze^{\mp i\omega_r T})^{-k} \\ &= E(ze^{\mp i\omega_r T}) . \end{aligned}$$

Equation 3 then becomes

$$\begin{aligned} \mathcal{Z}\{\epsilon \cdot x_I\} &= \\ \frac{C}{2} \left[ E(ze^{-i\omega_r T}) e^{i\theta} + E(ze^{+i\omega_r T}) e^{-i\theta} \right] . \end{aligned}$$

The signal  $\epsilon \cdot x_I$  then passes through an integrator to become the in-phase LMS coefficient  $w_I(k)$  whose Z-transform,  $W_I(z)$ , is given as

$$W_I(z) = \frac{C}{2} H(z) \left[ E(ze^{-i\omega_r T}) e^{i\theta} + E(ze^{i\omega_r T}) e^{-i\theta} \right]$$

where

$$H(z) = \frac{\mu z}{z - 1}$$

is the Z-transform for the integrator with gain  $\mu$  which is shown within the dashed box of Fig. 2.

Finally,  $w_I(k)$  is multiplied by the in-phase noise reference,  $x_I(k)$ , to obtain the in-phase component of the output,  $y_I(k)$ , whose Z-transform,  $Y_I(z)$ , is given by

$$\begin{aligned} Y_I(z) &= \mathcal{Z}\{w_I \cdot x_I\} \\ &= \frac{C}{2} [W_I(ze^{-i\omega_r T}) + W_I(ze^{i\omega_r T})] \\ &= \frac{C^2}{4} E(z) [H(ze^{-i\omega_r T}) + H(ze^{i\omega_r T})] \\ &\quad + \frac{C^2}{4} E(ze^{-i2\omega_r T}) H(ze^{-i\omega_r T}) e^{i2\theta} \\ &\quad + \frac{C^2}{4} E(ze^{i2\omega_r T}) H(ze^{i\omega_r T}) e^{-i2\theta} . \end{aligned} \quad (4)$$

The Z-transform for the quadrature component of the output,  $y_Q$ , can be found in a similar way to be

$$\begin{aligned} Y_Q(z) &= \mathcal{Z}\{w_Q \cdot x_Q\} \\ &= \frac{C}{2j} [W_Q(ze^{-i\omega_r T}) - W_Q(ze^{i\omega_r T})] \\ &= \frac{C^2}{4} E(z) [H(ze^{-i\omega_r T}) + H(ze^{i\omega_r T})] \\ &\quad - \frac{C^2}{4} E(ze^{-i2\omega_r T}) H(ze^{-i\omega_r T}) e^{i2\theta} \\ &\quad - \frac{C^2}{4} E(ze^{i2\omega_r T}) H(ze^{i\omega_r T}) e^{-i2\theta} \end{aligned} \quad (5)$$

The Z-transform of the output signal  $y(k)$ ,  $Y(z)$ , is given by the sum of Eqs. 4 and 5. Dividing by  $E(z)$  then gives the transfer function,  $G(z)$  from  $\epsilon$  to  $y$  as

$$G(z) = \frac{C^2}{2} [H(ze^{-i\omega_r T}) + H(ze^{i\omega_r T})]$$

Using Eq. 4,  $G(z)$  becomes

$$G(z) = \mu C^2 \frac{z(z - \cos(\omega_r T))}{(ze^{-i\omega_r T} - 1)(ze^{i\omega_r T} - 1)}.$$

Thus the technique of *baseband processing* with mixing, or down/up conversion, as depicted by Fig. 2, results in a shifting of the pole at DC, or  $z = 1$ , of the baseband transfer function  $H(z)$  to poles at  $z = e^{\pm i\omega_r T}$ , or at the noise reference frequency  $\omega_r$ . This same technique is exploited in LLRF control systems and RF filtering applications. Also note the additional zero in  $G(z)$  compared to  $H(z)$ .

The beauty and great utility of this technique is made evident by looking at the noise rejection transfer function, denoted as  $F(z)$ , describing the transmission from  $s(k)$  to  $\epsilon(k)$  of Fig. 2

$$\begin{aligned} F(z) &= \frac{E(z)}{S(z)} = \frac{1}{1 + G(z)} \\ &= \frac{(ze^{-i\omega_r T} - 1)(ze^{i\omega_r T} - 1)}{(1 + \mu C^2)z^2 - \cos(\omega_r T)(2 + \mu C^2)z + 1} \end{aligned} \quad (6)$$

Equation 6 describes the transfer function for the digital adaptive notch filter. It has the desired property of having a transmission zero exactly at the noise reference frequency. The bandwidth and hence settling time of the filter is controlled by both the magnitude of the noise reference signal and the adjustment parameter  $\mu$  through the  $\mu C^2$  term which determines the pole locations. The theoretical magnitude response for various values of  $\mu C^2$  are shown in Fig. 3.

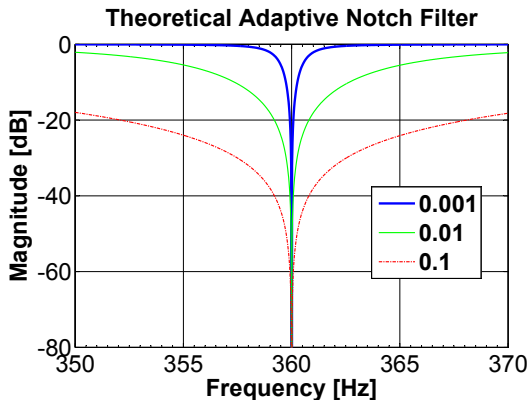


Figure 3: Theoretical notch filter magnitude response for various values of  $\mu C^2$ .

## 2 Application of the Adaptive Notch Filter

The adaptive notch filter can be applied to the APS RF systems as depicted in Fig. 4. The noise source is the ripple on the klystron's high voltage power supply (HVPS). This noise generates RF amplitude and RF phase noise in the RF output of the klystron which are respectively detected with an envelope and phase detector. The outputs of these detectors, or the error signals  $\epsilon_{amp}$  and  $\epsilon_{phs}$ , are considered the output of their respective noise canceller. A noise reference,  $x$ , can be taken from the high voltage power supply's cathode voltage monitor. The signals  $y_{amp}$  and  $y_{phs}$  are respectively the output of the amplitude and phase adaptive filters. These outputs are used to modulate the RF drive signal so as to cancel the RF modulation happening within the klystron caused by the high voltage power supply ripple.

Simultaneous cancellation at multiple 60Hz harmonics is achieved by pre-filtering both the noise reference signal,  $x$ , and the error signals,  $\epsilon_{amp}$  and  $\epsilon_{phs}$ , with narrow band-pass filters with corresponding center frequencies to prevent intermodulation distortion effects and crosstalk between the multiple adaptive filter *channels*. The quadrature noise reference signals are generated using a Hilbert filter that precedes the noise reference signals' band-pass filter banks. The outputs of all adaptive filter channels are then summed together to generate the amplitude and phase control outputs,  $y_{amp}$  and  $y_{phs}$ , to the RF modulator. Three frequencies, 60Hz, 120Hz, and 360Hz, for each RF amplitude and RF phase were attacked here.

The adaptive notch filter was developed using a National Instruments (NI) compact Real-time Input Output (cRIO) system chassis, model cRIO-9118. The digital algorithm was implemented on the cRIO-9118's Xilinx Virtex-5 LX110 field-programmable gate array (FPGA). This particular FPGA contains 64 DSP48E slices, which are used for high-throughput multiplication. Analog to digital conversion was performed with a NI 9222 cRIO module which con-

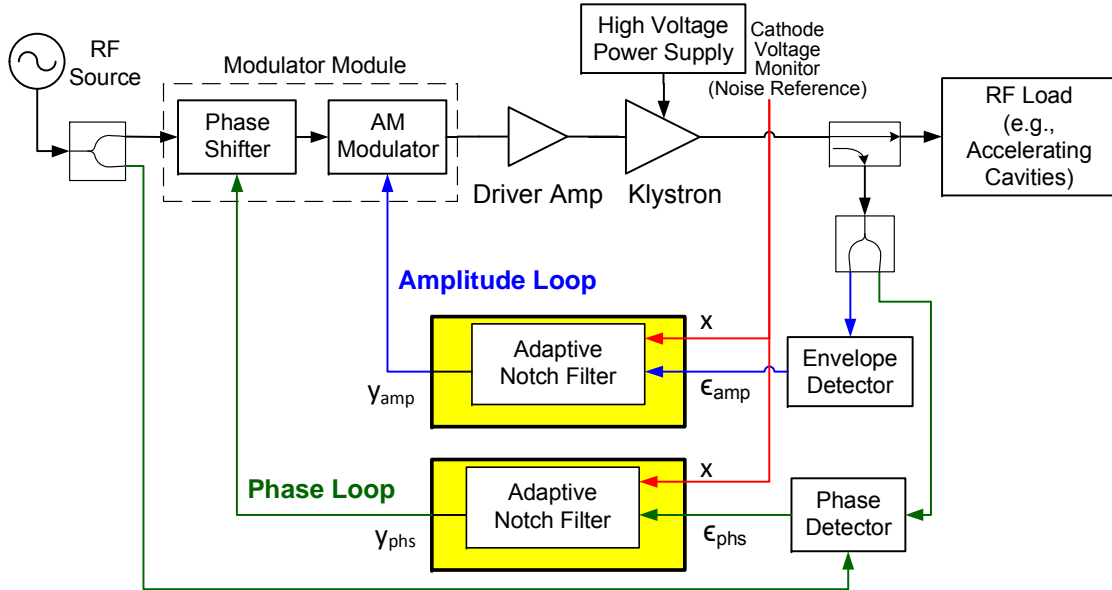


Figure 4: Application of the notch filter for the RF system.

sists of 4,  $\pm 10V$ , 16-bit analog-to-digital converter (ADC) channels. Output conversion was performed with a NI 9269 cRIO module which consists of 4,  $\pm 10V$ , 16-bit digital-to-analog converter (DAC) channels. Host communication to the FPGA is aided with the chassis' cRIO-9024 real-time PowerPC controller.

A block diagram of the FPGA firmware code that was developed for the cRIO system using LabView contains two main loops executing at different fixed rates as shown in Fig. 5. The processing loop executes at 10 kHz, which easily satisfies the timing requirements of the LMS computations. The Input/Output (I/O) loop executes at 100 kHz, which allows oversampling the ADCs and DACs.

Oversampling is a method of taking samples faster than that needed for processing the bandwidth of interest. Not only does it relax requirements for the analog anti-aliasing filters by increasing the Nyquist frequency, but it also improves the signal to noise ratio over the bandwidth of interest. In our application, the oversampling ratio is 10, thus reducing the ADC noise power over the bandwidth of interest by this same factor. The oversampled signal can be decimated (i.e., samples are discarded) subsequent to digital anti-alias filtering to achieve

an internal digital signal processing rate which is applicable to the original bandwidth of interest. After digital signal processing, the output is interpolated by a factor of ten. This similarly reduces the DAC quantization noise power over the bandwidth of interest. It also increases the frequencies of the DAC images (i.e., images of the baseband signal are produced around 100 kHz instead of around 10 kHz), thereby relaxing the requirements of the analog reconstruction filter.

The processing loop is comprised of three distinct stages. The first stage is preparing the inputs for the LMS algorithm. The noise reference signal is split into in-phase and quadrature components using a Hilbert filter. Then the two quadrature reference signals and the two error signals (amplitude and phase) are each fed through a parallel set of 60 Hz, 120 Hz, and 360 Hz band pass filters in order to select the target frequencies. A tunable delay is provided on the error signals for each frequency channel in order to tune the overall phase of the feedback loop for proper negative feedback and stability considerations. A total of 12 signals: three frequency channels each of the RF phase error, the RF amplitude error, the in-phase reference, and the quadrature reference are then passed into the second stage, the LMS algorithm.

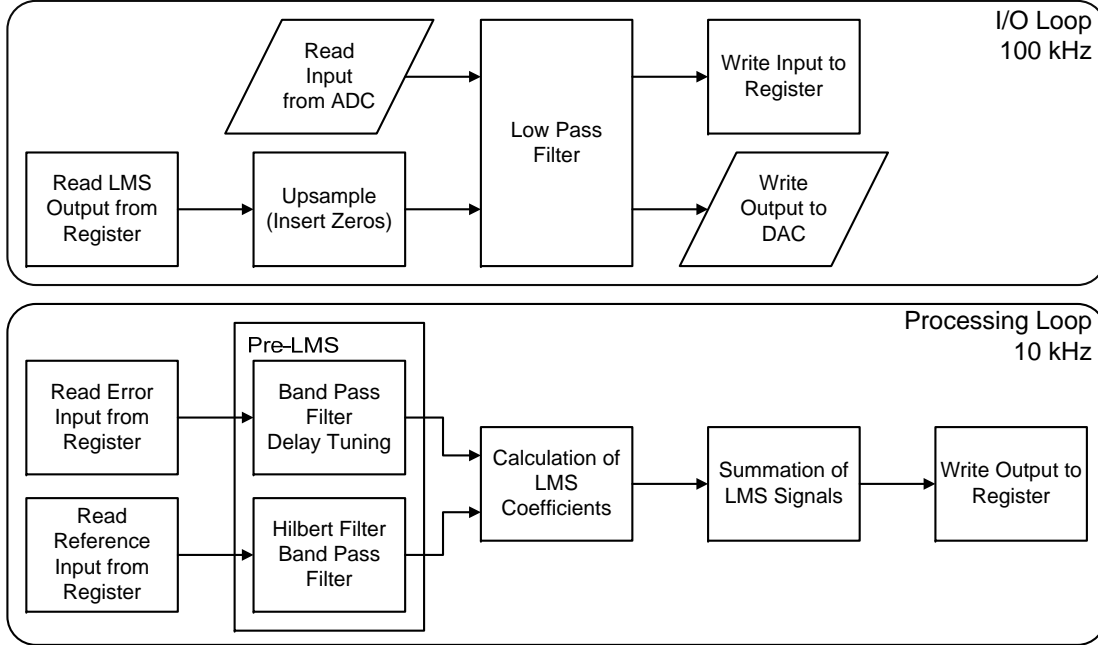


Figure 5: Abstraction of the FPGA code for the adaptive notch filter.

The LMS algorithm, or second stage, performs the following list of computations on each error component sample according to Eqs. 1b and 1c. Each item is completed in a single clock cycle and requires the use of one DSP48E slice for multiplication. Hence the algorithm takes a total of 3 slices.

1. Multiplies by the corresponding I/Q noise reference component:  $\epsilon[k] \cdot x_{I/Q}[k]$
2. Multiplies by the gain adjust,  $\mu$ , and adds the previous coefficient value:  $w[k-1] + \mu\epsilon[k] \cdot x_{I/Q}[k]$
3. Multiplies by the corresponding I/Q reference component:  $w[k] \cdot x_{I/Q}[k]$

In the third and final stage of the processing loop, the total adaptive filter output for each path (amplitude and phase) is formed by the sum of the samples from all frequency channels for that respective path. The two resulting output signals,  $y_{amp}$  and  $y_{phs}$ , are then written to memory for the I/O loop to use for upsampling and interpolation.

The original implementation of the LMS algorithm had each frequency channel operating in

parallel. Due to the number of DSP48E slices required for each parallel path, the original implementation could not support more than three target frequencies. In order to reduce the resources required for each channel and to allow the program to be easily expanded, the new code uses time-division multiplexing within the LMS algorithm. Instead of needing twelve separate instances of the LMS algorithm, only one shared instance with a deeper memory is needed. The channels are processed sequentially within the LMS coefficient block using a process that runs at the FPGA clock rate of 40 MHz. This increases the number of FPGA clock cycles required to process all 12 channels, but requires only 3 DSP48E slices to calculate all of the LMS coefficients. Without time-division multiplexing, each additional target frequency would require adding 16 DSP48E slices: 4 for band pass filters (1 each for the 2 quadrature noise references and 1 each for the amplitude and phase errors) and 12 for the LMS coefficients (3 multiplies for each of the 4 LMS coefficients). With time-division multiplexing, adding an additional target frequency only requires adding 4 slices for the band pass filters.

In order to reduce the latency introduced to the LMS algorithm by time-division multiplexing, the algorithm is implemented as a pipelined process with three stages. This allows the LMS algorithm to work as an assembly line. The first channel will take four FPGA clock cycles to propagate through the algorithm, but each of the remaining channels will arrive at the output in consecutive cycles. Instead of taking  $4 \times 12 = 48$  cycles at 40 MHz to compute the 12 coefficients in the three frequency case, the pipelined algorithm only takes  $4 + 11 = 15$  cycles at 40 MHz to complete.

The notching response of a 360Hz channel was measured on the bench using a Stanford Research Systems SR-785 dynamic signal analyzer and an in-house analog signal conditioning circuit to form the proper error summing junction of Fig. 2. A noise reference signal,  $x$ , was provided by an external generator. The SR-785 was used in transfer function mode. The SR-785 source output provided the  $s$  signal of Fig. 2. The input of the SR-785 was fed with a monitor of the error signal  $\epsilon$ . The measured response is shown in Fig. 6.

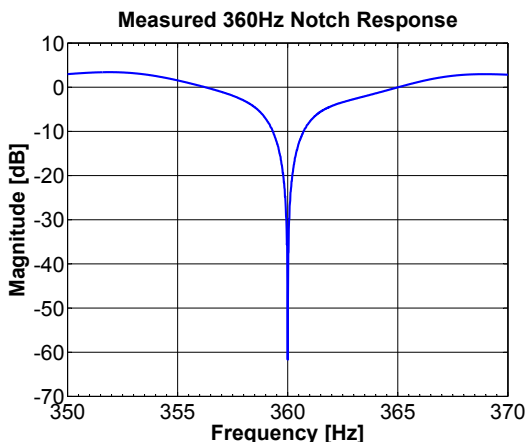


Figure 6: Measured notch filter response for a 360Hz channel.

The suppression is not perfect at 360Hz due to both the SR-785 and the noise reference generator not being frequency locked, and frequency resolution of the measurement. Gain outside the notch is believed to be due to the additional frequency responses from the analog anti-aliasing

and reconstruction filters, the digital bandpass filters, and delays. Obviously changing the value of  $\mu$  allows one to shape the width of the notch similar to what was seen in Fig. 3.

## 2.1 Front Panel Controls

The LabView FPGA environment allows the FPGA to communicate with a front panel on a host computer. The front panel is configured so that the operator can adjust parameters of the adaptive notch filter. These parameters include for each frequency channel: enable and reset switches, a numerical control for  $\mu$ , a numerical control for error sample delay, and indicators for the the LMS coefficients. Some global controls and indicators are also present, allowing the operator to invert the error or the reference signal and change internal scaling. Signals can also be viewed real-time in both the time and frequency domains and can be saved for further off-line analysis.

## 3 RF System Integration

In order to preserve existing RF feedback loops around the klystron, the NI adaptive notch filter hardware, hereby called the adaptive noise cancellation system (ANC), is integrated into an existing RF station as depicted in Fig. 7. An analog signal conditioning chassis is used to interface the existing analog signals with the NI equipment. Although it is not fully detailed in Fig. 7, it can be thought of as what provides the interfacing within the yellow box regions. The signal conditioning chassis contains analog circuitry which performs several functions. Its primary function is to amplify the small RF phase and RF amplitude error voltages into the usable dynamic range of the ADC's in order to maximize signal-to-noise ratio (SNR) at the ADC. Similarly, the dynamic range of the DAC's is fully utilized to improve DAC SNR. The DAC output is scaled down in the signal conditioning chassis to produce the small control voltages needed for controlling the RF amplitude and phase modulators.

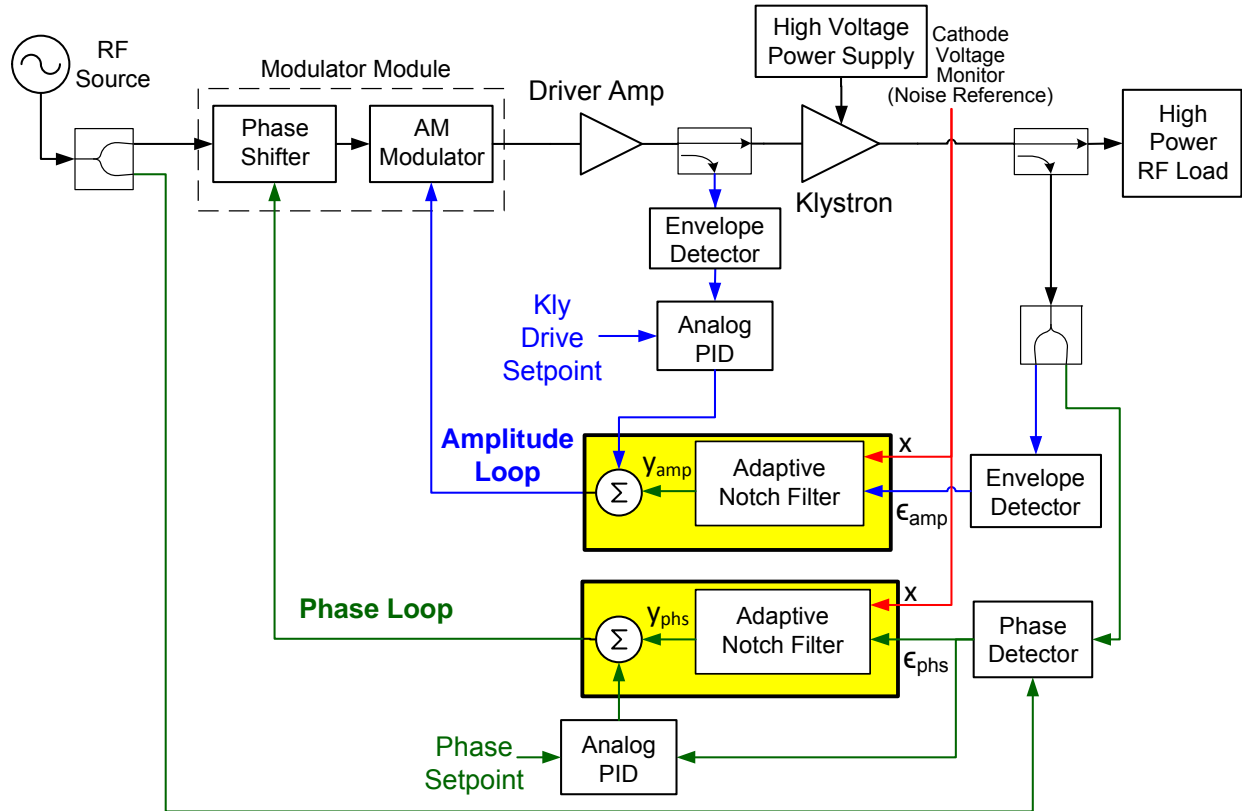


Figure 7: Integration of the ANC system into a Storage Ring RF Station.

Ideally the input signals from the RF amplitude and phase detectors and the HVPS cathode voltage monitor should be scaled to a maximum value of  $\pm 10$  V to fit within the range of the NI 9222 cRIO 16-bit ADC's. The quantization interval for this module is

$$\Delta V = \frac{20V}{2^{16}} \approx 0.305mV$$

The mean square value, or noise power, of quantization noise can be approximated by  $\Delta V^2/12$  [3]. This noise power is constant assuming there is an appropriate signal level and signal frequency content. Thus, when the signal level, or its power, is increased the signal to quantization noise ratio is also increased. The analog signal conditioning chassis amplifies the input from the RF phase and amplitude monitors, as well as from the HVPS cathode voltage monitor, by a factor of 30, or 30dB, thereby improving the signal-to-noise ratio by this same factor compared to not using any pre-amplifier. This

assumes that the original SNR and the noise introduced by the pre-amp do not overcome this improvement, which is a fair assumption.

Low pass filters (LPF) were utilized for the ADC input anti-aliasing filters and the DAC output reconstruction (or DAC image suppression) filters. The anti-aliasing filter removes frequencies greater than one half of the sampling rate, or 50 kHz. The reconstruction filter must also remove DAC image frequencies at one half of the sampling rate and greater. The highest target frequency was 360 Hz, so a cutoff frequency of 1 kHz was appropriate for the LPFs. The LPF was a simple single pole resistor capacitor (RC) filter. The three anti-aliasing filters were implemented in stand-alone passive component filter boxes, while the reconstruction filters were implemented within the analog signal conditioning chassis.

As shown in Fig. 7, the ANC system outputs are summed with the outputs of existing analog PID controllers. The existing analog PID con-



troller for RF amplitude is used to maintain a fixed RF drive level to the klystron. Running the klystron at a fixed RF drive level is required for operations for various reasons such as maintaining klystron stability and achieving certain efficiency. Unfortunately this loop tends to fight the adaptive notch loop. If the amplitude noise originates within the klystron, then the adaptive notch filter is using the driver amplifier output to counteract the noise. However, the klystron drive loop is trying to suppress any fluctuations on the RF drive level. The bandwidth of the klystron drive loop as well as the gain of the adaptive notch loop can be adjusted to reduce the overlap and hence interaction between the loops.

The existing analog PID controller for RF phase is purely in parallel with the adaptive notch for RF phase. They both are working with each other. The analog loop is used to maintain a fixed RF phase of the klystron output relative to an RF reference. The adaptive notch does not operate at DC, therefore it does not interfere with the DC setpoint. The PID controller provides general broadband noise suppression while the adaptive notch attacks the targeted 60Hz harmonics.

## 4 RF Test Stand Study

The ANC system was tested in the APS building 420 RF Test Stand using station RF1’s 1 Mega-Watt (MW) klystron. The test stand has a 1 MW RF water load for the klystron to drive. Measurements were made while running the klystron at 300 kW.

Fig. 8 displays the RF amplitude and phase noise power spectral densities (PSD) of a klystron output RF monitoring signal as measured with an Agilent E5052B signal source analyzer. The large reduction at the three targeted frequencies is quantified in Table 1. The noise at the targeted frequencies are reduced to levels near to the measurement noise floor. Some small noise can be seen to be introduced by the ANC at 10 kHz and its harmonics in the RF phase noise PSD. This is due to the finite suppression of the

Frequency (Hz)	Noise Reduction (dB)	
	Phase	Amplitude
60	-33.31	-29.92
120	-31.02	-34.21
360	-35.94	-38.95

Table 1: Reduction of phase and amplitude noise by ANC system.

10 kHz images afforded by the low-pass interpolation filter used for upsampling to the 100kHz DAC update rate. The larger broadband amplitude noise seen above 1kHz with the ANC OFF is believed to be due to the Booster dipole supply which was being turned on and off during our studies. This supply is pulsed at 2Hz, thereby coupling 2Hz related noise onto the RF system power grid. The 2Hz harmonics are evident in the PSD plots and are seen to be higher in the ANC OFF case than in the ANC ON case for the amplitude noise.

The cumulative integrated noise is shown in Fig. 9. The root-mean-square (rms) noise integrated over 10Hz to 1kHz is seen to improve by a factor of  $\sim 5$  for both phase and amplitude due to reduction of the 60Hz, 120Hz, and 360Hz components.

Time domain plots are shown Fig. 10. A calibrated scaling factor for the envelope detector was not measured at our operating point, so an absolute percent amplitude noise level could not be determined. Instead, the raw volts coming from the envelope detector are shown. The remaining noise is mainly due to 180 Hz, 240 Hz, and 720 Hz as can be determined from the PSD plots of Fig. 8. These frequencies could be addressed by adding more channels to the FPGA code. Based upon the cumulative noise plots of Fig. 9, perhaps another factor of 2 reduction in the rms phase noise could be achieved. While improvement at these frequencies could also be achieved for amplitude noise, it appears that little improvement would result in the total rms amplitude noise due to the 2Hz harmonics from pulsing of the Booster dipole.

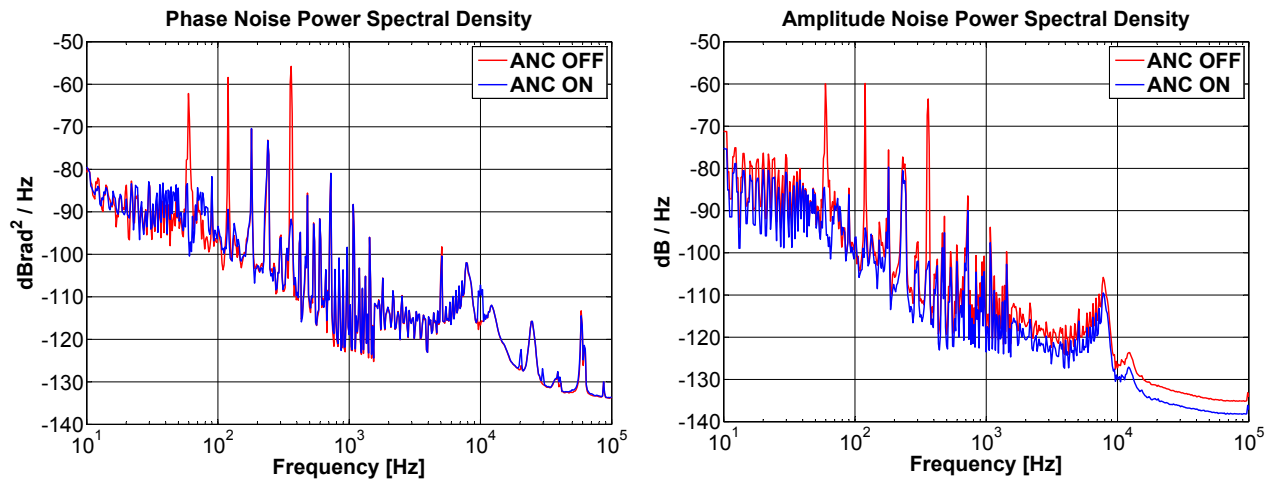


Figure 8: Measured Klystron Output RF Phase and Amplitude Noise PSD.

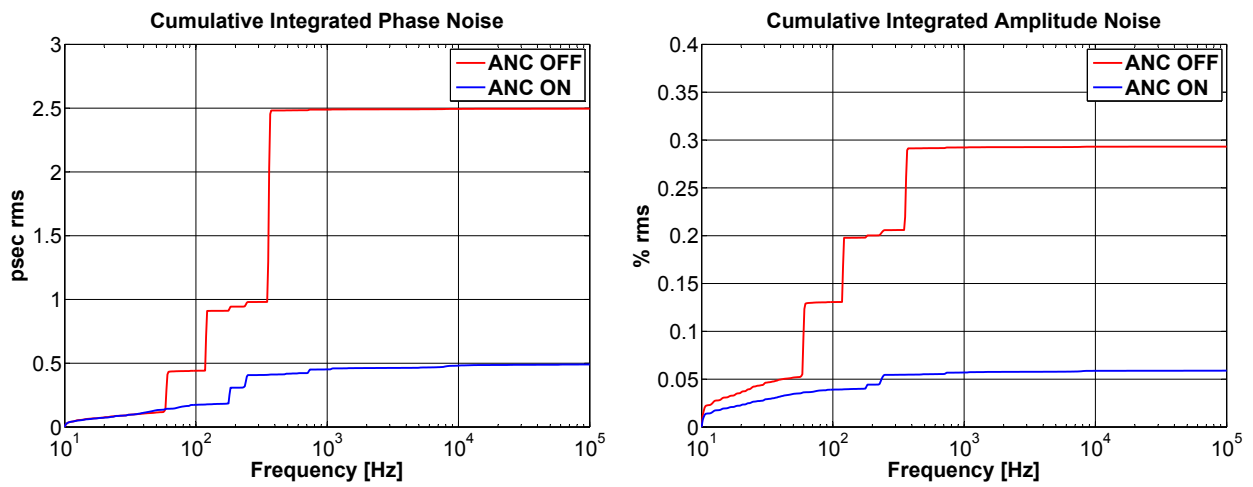


Figure 9: Cumulative Integrated RF Phase and Amplitude Noise

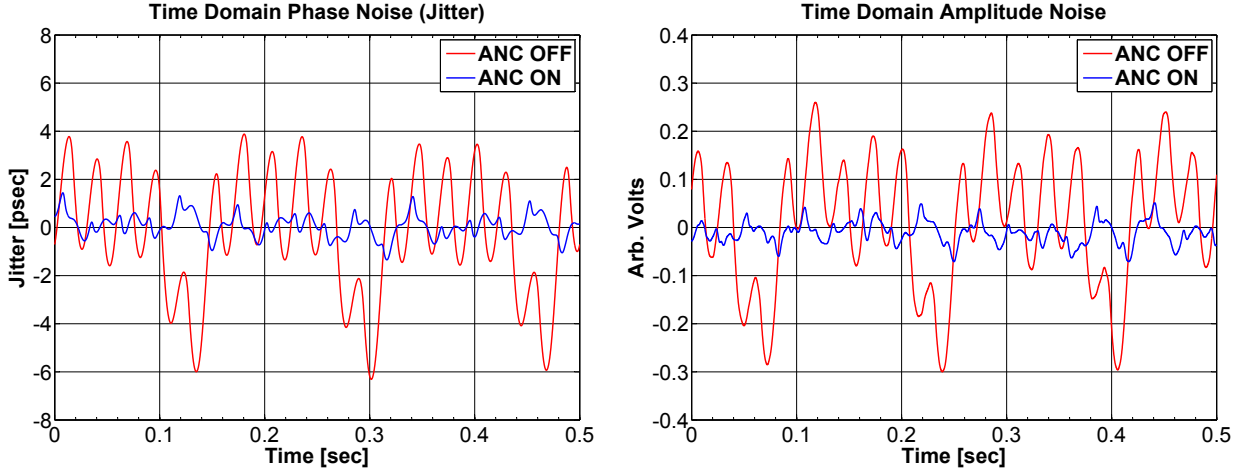


Figure 10: Time Domain Phase and Amplitude Noise

## 5 Conclusion and Future Work

Based on the results obtained at the RF test stand, the adaptive noise cancelling system was very effective. It reduced the targeted noise by  $\sim 30$  dB to  $\sim 39$  dB. The only observable noise added by the system was negligible noise from imperfect suppression of image frequencies associated with DAC output interpolation. The FPGA code can be easily modified to cancel noise at additional 60Hz harmonics, such as the 180 Hz, 240 Hz, and 720 Hz lines. While it is possible that another factor of 2 reduction in the total rms phase noise may be achievable, the total rms amplitude noise may be little affected due to the contribution of 2Hz harmonics from pulsing of the Booster dipole.

One interesting observation, not mentioned in the text, was that the RF phase noise as seen by the signal source analyzer showed less improvement if RF amplitude noise suppression was turned off. This is suspected to be due the mixer based analog phase detector being sensitive to RF amplitude. When the amplitude suppression is turned on, the phase suppression system correctly suppresses phase noise as opposed to some mixture of amplitude and phase noise. The converse scenario was not seen, most likely because the amplitude detectors are diode based envelope detectors which are insensitive to RF phase. The results presented here were with

simultaneous amplitude and phase suppression, which would be the typical operational scenario.

Future plans include implementing the ANC in a MicroTCA system which will provide increased flexibility in both hardware and control system integration. This could also allow for a digital LLRF receiver to replace the analog amplitude and phase detectors.

The ANC system should be tested on an operational RF system to analyze the effects on beam stability. A stability analysis should also be performed to optimize notch bandwidth and interaction with the other existing loops.

## 6 References

### References

- [1] J. Glover, "Adaptive Noise Canceling Applied to Sinusoidal Interferences" *IEEE Trans. on Acoustics, Speech, and Signal Processing* vol. ASSP-25, no. 6, Dec 1977.
- [2] B. Widrow et.al., "Adaptive noise cancelling: Principles and applications" *Proc. IEEE* vol.63 no.12, Dec.1975.
- [3] W. R. Bennett, "Spectra of Quantized Signals" *Bell System Technical Journal*, vol. 27, pp. 446-472, July 1948.