BNL-229013-2025-TECH
EIC-ADD-TN-147

ONE-TURN DELAY FEEDBACK (OTFB) IMPLEMENTATION FOR THE EIC

A. Singh

September 2025

Electron-Ion Collider Project
Brookhaven National Laboratory

U.S. Department of Energy
USDOE Office of Science (SC), Nuclear Physics (NP)

Notice: This technical note has been authored by employees of Brookhaven Science Associates, LLC under Contract No.DE-SC0012704 with the U.S. Department of Energy. The publisher by accepting the technical note for publication acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this technical note, or allow others to do so, for United States Government purposes.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or any third party's use or the results of such use of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof or its contractors or subcontractors. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ONE-TURN DELAY FEEDBACK (OTFB) IMPLEMENTATION FOR THE EIC

A. Singh* , S. Mai, K. Mernick, G. Narayan, F. Severino
*asingh2@bnl.gov

Abstract

This paper discusses the implementation of the One-Turn Delay Feedback (OTFB) for the Electron-Ion Collider (EIC). The theory of the OTFB is well documented, therefore, this paper will mainly discuss the details of the implementation. After a short introduction, the paper will discuss the building-blocks of the OTFB, simulation testing of the OTFB, and finally hardware validation of the OTFB.

INTRODUCTION

The background and theory of the OTFB has been extensively covered by others [1, 3], and for this reason, the focus of this paper is mainly on the specific implementation details. Conceptually, while the direct feedback lowers the effective cavity impedance around the RF frequency, the OTFB lowers the effective cavity impedance at revolution frequency harmonics around the RF frequency (specifically, at the synchrotron and betatron sidebands around the revolution frequency harmonics). The OTFB is made of two main components, the comb filter and the delay line. The comb filter is responsible for actually providing gain at the revolution harmonics. The delay line is needed to adjust the phase of the drive signal to maintain stability. An effective implementation of the OTFB also requires more capabilities. Firstly, the harmonic peaks of the comb filter must be adjustable to account for variations in the revolution frequency. Secondly, the filter must be able to provide gain at arbitrary offsets from the rev-lines to effectively lower the cavity impedance at the synchrotron and betatron sidebands.

The paper then briefly touches on how these components were tested using simulations prior to being put on hardware. Finally, the paper discusses results with the OTFB on hardware including the setup procedure and the performance of the feedback.

DIGITAL IMPLEMENTATION

The OTFB is implemented in VHDL and the initial version was tested on the Xilinx Vertix-5 FPGA. Like the traditional direct feedback loop (using the PID), the OTFB must be applied to both I and Q data.

Biquad Filter

The transfer function and frequency response of a standard feedback comb filter is shown below:

$$H(z) = \frac{1-\alpha}{1-\alpha \cdot z^{-D}} \tag{1}$$

* This work was supported by the EIC Project and the U.S. Department of Energy, Contract DE-SC0012704.

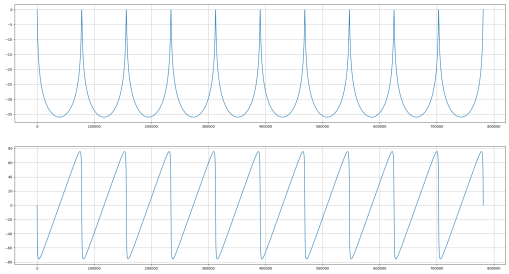


Figure 1: Basic comb filter response

The numerator normalizes the gain of the filter to 1 at the poles. The coefficient D is a delay parameter of the comb filter. The coefficient α determines the bandwidth of the peaks, with the response becoming more narrow as α approaches 1. The basic principle of the filter is that repeating poles of the transfer function leads to gain at harmonic frequencies, and the fundamental frequency is related to the sampling frequency by:

$$F_f = \frac{F_s}{D} \tag{2}$$

While this configuration will apply gain at the revolution harmonics, the transfer function must be modified slightly to apply gain at synchrotron/betatron sidebands. This can be achieved by shifting the poles via multiplication by a complex exponential. The new transfer function is shown below:

$$H(z) = \frac{1-\alpha}{1-\alpha \cdot z^{-D} \cdot e^{-j\phi}} \tag{3}$$

This has an elegant interpretation when one looks at the pole-zero plot of the filter. Multiplying by this complex exponential has the effect of rotating the poles by an amount proportional to ϕ. This can be seen below:

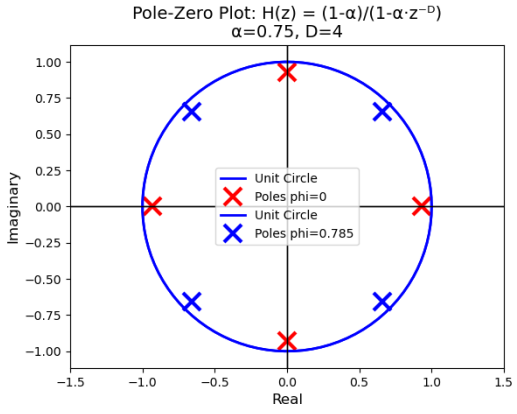


Figure 2: Pole-Zero Plot Showing Rotation With $\phi = \pi/4$

By combining two comb filters, one rotated clockwise and one counterclockwise, we are able to provide gain at offsets corresponding to the synchrotron/betatron sidebands. By multiplying eq (3) by a comb filter response with $+\phi$ rotation, we get:

$$H(z,\phi)\cdot H(z,-\phi) = \frac{(1-\alpha)^2}{1-\alpha\, z^{-D}(e^{-j\phi}+e^{j\phi})+\alpha^2\, z^{-2D}} \tag{4}$$

Which results in:

$$H(z) = \frac{(1-\alpha)^2}{1-2\,\alpha\, cos(\phi)\, z^{-D}+\alpha^2\, z^{-2D}} \tag{5}$$

Lastly, when providing gain at the sidebands, one may be interested in notching out the gain at the rev-lines, because without doing this, the peaks will be widened. This can be done by adding zeros at all the revolution lines by modifying (5). Also, to preserve the normalizing effect of the numerator coefficient, $(1-\alpha)^2$ must be changed to $(1-\alpha^2)$. The final transfer function is given by:

$$H(z) = G\cdot\frac{(1-\alpha^2)(1-z^{-D})}{1-2\,\alpha\, cos(\phi)\, z^{-D}+\alpha^2\, z^{-2D}} \tag{6}$$

Where G is some arbitrary gain. Now, the value of ϕ is determined by:

$$\phi = 2\pi\cdot\frac{f_{offset}}{f_{rev}}$$

Where f_{offset} is the offset of the sidebands from the rev-lines in Hz. One may be able to see that the transfer function given by (6) is similar to the biquad implementation of the comb filter shown in [2], in fact, (6) is equivalent to a general biquad implementation and this is how the filter is actually implemented in VHDL. Some general frequency responses for such a filter are shown below:

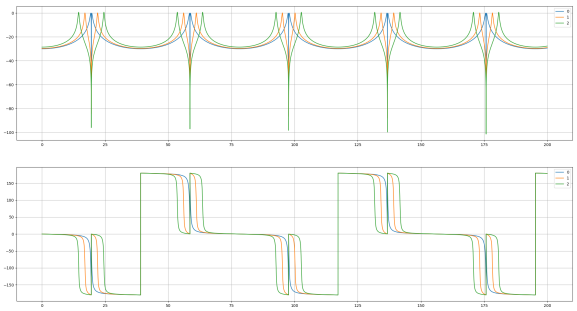


Figure 3: General response of the biquad comb filter. Here, ϕ increases from blue to green and the gain at revolution harmonics is notched out.

A block diagram of the filter is shown below:

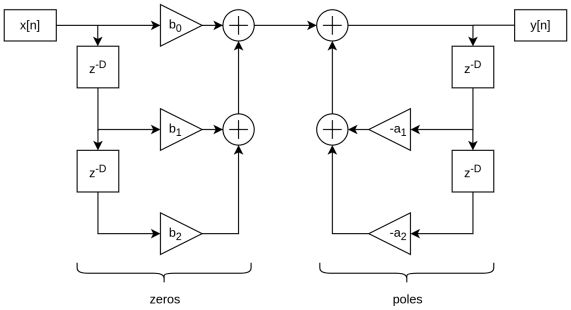


Figure 4: Block diagram of the biquad comb filter

For our purposes, the coefficient b_2 is always 0.

Fractional Delay Filter

We see, according to (2), that if we use a sampling rate of 20 MHz and a delay of 256, we get harmonics with a fundamental frequency of 78.125 kHz. While this value is close to the RHIC revolution frequency, often it is not exact. To account for any discrepancies, we need to be able to adjust the filter delay by fractional amounts. The fractional delay filter design used is based largly on the design described in [4] which provides a comprehensive overview. A fourth order fractional delay filter is used in this design and, according to [4], the transfer function for such a filter is given by:

$$H(z) = \sum_{i=0}^{4} C_i(z)D^i \tag{7}$$

Where C represents coefficients and D represents the desired delay within the range [-1,1]. As described in [4], such a filter can be implemented as a Farrow structure. However, this implementation, even after simplifications, requires 18 multiplications, which is not practical given that 6 fractional delay filters are required (3 for a single comb filter). Instead, the like-terms can be combined, resulting in a simple FIR implementation with only 5 multiplications. While we lose out on the ability to quickly change the fractional delay value (now we have to recompute the coefficients), in practice this is not an issue as the fractional delay will only be used to provide a one-time correction.

The fractional delay FIR filter is added after each of the delay blocks seen in figure 4.

Delay Line

The delay line for the OTFB is implemented in BRAM. The length of the delay line is determined by the revolution frequency at storage. In RHIC, and EIC, the revolution frequency is $\approx$ 78.125 kHz, and if the clock rate is 20 MHz, a delay line of 256 samples is required. In the current design, the delay line can be adjusted with a resolution of 50 ns, in the future, a fractional delay filter can also be utilized to achieve higher precision.

Phase Equalizer Filter

As is noted in [3], due to the phase response of the closed-loop system, instabilities can occur when providing gain at around +/- 6 harmonics from the RF frequency. This is because a 180 degree phase rollover occurs at those frequencies and instead of reducing the cavities impedance there, the comb filter actually increases the impedance. To address this issue, a so-called phase equalizer filter can be used. Specifically, a phase equalizer filter is an all-pass filter (unity gain at all frequencies) with some tunable phase response. Based on [10], the transfer function and frequency response for such a filter is shown below:

$$H(z) = \frac{a_2 + a_1 \cdot z^{-D} + z^{-2D}}{1 + a_1 \cdot z^{-D} + a_2 \cdot z^{-2D}} \tag{8}$$

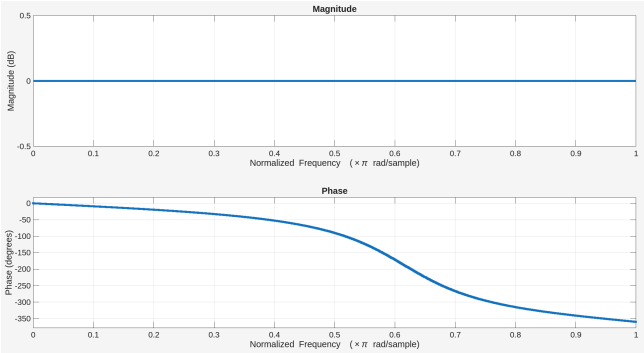


Figure 5: Response of eq (8) with $a_1 = a_2 = 0.5$ and D = 1

The delay parameter D is responsible for determining how many phase rollovers occur within 0 Hz and $\frac{fs}{2}$ Hz. The parameter a_1 determines the position of the phase rollover (shifting it left and right) and a_2 determines how quickly the phase rolls over. By using $D = 12,\ a_1 = 0.8,\ a_2 = 0.3$, we are able to compensate for the phase response of the close loop system, as will be shown shortly.

SIMULATION TESTING

The VHDL simulations and verification was done using cocotb [5] and Python. As the VHDL simulator, GHDL [6] was chosen. The OTFB implementation was validated by applying a discrete impulse, taking an FFT of the output, and plotting the result against the ideal frequency response. This method allows one to quickly check the quantization noise and numerical errors as a function of data and coefficient length. Also, taking the impulse response of the filter quickly reveals any timing errors in the design. In fact, such a response is what is seen in figure 3.

HARDWARE TESTING

The OTFB was first tested within a lab environment. There are multiple ways to verify the design is behaving as expected. For example, one can calculate the impulse response by applying a discrete pulse to the filter on the FPGA. One can then view the output of the filter on a scope.

In this case, the preferred testing method is to measure the closed loop transfer function of a cavity simulator while the OTFB is enabled. A block diagram of such a setup is shown below:

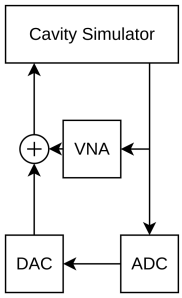


Figure 6: System Response Measurement Setup

The result returned by the network analyzer can be thought of as a measurement of the cavity impedance. An open loop measurement is shown below:

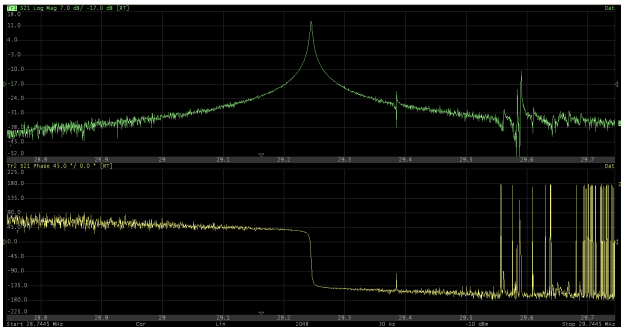


Figure 7: Open Loop System Response Measurement

As expected, this returns the classic resonator response of the cavity. If only the direct feedback is enabled, the measurement returns the following:

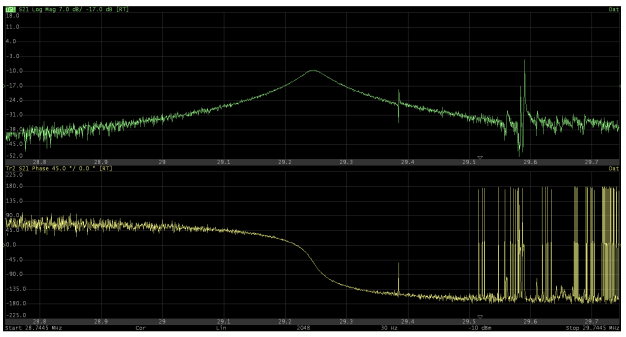


Figure 8: Closed Loop System Response Measurement

Again, as expected, the measurement shows the narrow-band, high gain, reduction of the cavity impedance (by 23 dB) due to the direct loop. Finally, the OTFB is properly configured and enabled, the measurement is shown below:

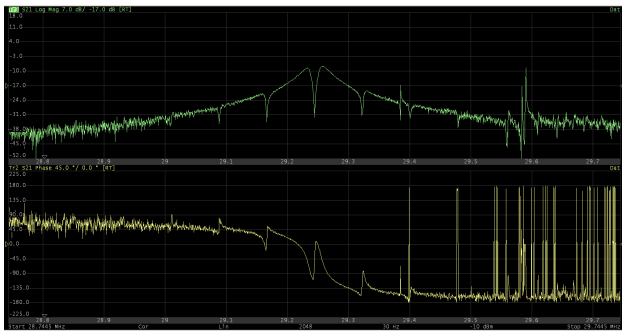


Figure 9: Closed Loop + OTFB System Response Measurement

This result clearly shows the OTFB lowering the cavity impedance at revolution lines around the RF frequency. The process by which the OTFB can be properly configured is discussed in the following section. Below, a measurement of the double-peak comb filter response is shown:

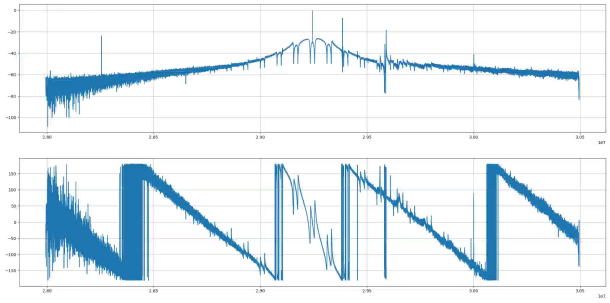


Figure 10: OTFB Double-Comb System Response Measurement

It should be noted, the spurs in the measurement (not including the one at RF) are due to the response of the crystal oscillator cavity simulator these tests were preformed on.

In a fully optimized state, we are able to provide close to 30 dB of impedance reduction around the first 3 harmonics, along with an additional 10 to 20 dB of impedance reduction up to 8 harmonics. The image below shows a OTFB setup with proper delay and phase equalizer settings:

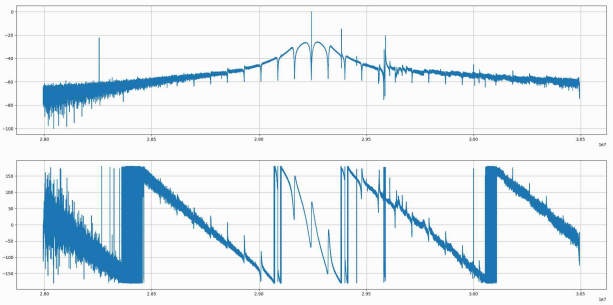


Figure 11: Fully optimized OTFB

TUNING THE OTFB

To properly configure the OTFB, a method similar to the one shown in [8] is used. First a simplistic, linear, model of the system is defined. An example model is shown below:

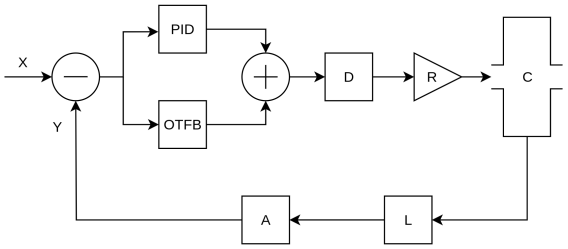


Figure 12: Simplistic Linear Model

From here, one can define the transfer function $\frac{y}{x} = H(s)$ which will contain terms that describe the loop delay. In the case of the OTFB, we are interested in defining exactly the value of the loop delay so we can configure the delay line. Now, measurements like the ones from the previous section are plotted against the theoretical response of the model. Specifically, measurements taken with the digital network analyzer [7] are used as they account for the response of digital components. Regardless, chances are that the two will not agree. In such a case, a least-squares fit is preformed on the model to better match the real measurement. Specifically:

$$L(\vec{\theta}) = \sum_{\omega} |H_{meas}(j\omega) - H_{fit}(\vec{\theta}, j\omega)|^2$$

must be minimized by altering $\vec{\theta}$ which is a vector containing all of the tweakable parameters of the ideal model. As this process is iterated a few times, the model will more accurately describe the real system and from there it is easy to obtain the loop delay based on the model parameters.

CONCLUSION

The OTFB implementation for the EIC has been described in this paper. Along with the testing methods already mentioned, the OTFB will be further validated with beam during an Accelerator Physics Experiment (APEX) at RHIC. While most of the desired functionality has been achieved, there still remain slight fixes that are needed along with some additional features. Firstly, in certain circumstances, the fractional delay FIR filter performance is limited by numerical errors. Specifically the combination of the double-comb response together with the fractional delay filter causes numerical errors. This can be addressed by increasing the taps of the FIR filter or by increasing the coefficient bit width. Secondly, the OTFB needs to be able to deal with a ramping revolution frequency, potential methods of dealing with this problem have been proposed by others [9].

ACKNOWLEDGMENTS

I'd like to thank Geetha Narayan, Freddy Severino, and Kevin Mernick for their support and advice throughout the duration of this project.

REFERENCES

[1] Garoby, Roland. (1999). "Low level RF and feedback." 10.1142/9789814447324_0012.

[2] G. Hagmann, P. Baudrenghien, J. Betz, J. Egli, F. J. Galindo Guarch, G. Kotzian, M. Rizzi, L. Schmid, A. Spierer and T. Włostowski, "The CERN SPS Low Level RF upgrade project," doi:10.18429/JACoW-IPAC2019-THPRB082

[3] Pedersen, F. (1992). "rf cavity feedback."

[4] V. Valimaki, "A new filter implementation strategy for Lagrange interpolation," Proceedings of ISCAS'95 - International Symposium on Circuits and Systems, Seattle, WA, USA, 1995, pp. 361-364 vol.1, doi: 10.1109/ISCAS.1995.521525

[5] https://www.cocotb.org/

[6] https://github.com/ghdl/ghdl

[7] S. Mai. (2025). "Baseband digital network analyzer upgrade for llrf controllers"

[8] Teytelman, Dmitry. "A Non-invasive Technique for Configuring Low Level RF Feedback Loops in PEP-II." , no. 2005.

[9] F. Javier Galindo Guarch, Philippe Baudrenghien, J. Manuel Moreno Arostegui, "A new beam synchronous processing architecture with a fixed frequency processing clock. Application to transient beam loading compensation in the CERN SPS machine." 2021, https://doi.org/10.1016/j.nima.2020.164894.

[10] P. A. Regalia, S. K. Mitra and P. P. Vaidyanathan, "The digital all-pass filter: a versatile signal processing building block," in Proceedings of the IEEE, vol. 76, no. 1, pp. 19-37, Jan. 1988, doi: 10.1109/5.3286.