BNL-229011-2025-TECH

EIC-ADD-TN-146

BASEBAND DIGITAL NETWORK ANALYZER UPGRADE FOR LLRF CONTROLLERS

S. Mai

September 2025

Electron-Ion Collider

Brookhaven National Laboratory

U.S. Department of Energy

USDOE Office of Science (SC), Nuclear Physics (NP)

DISCLAIMER

BASEBAND DIGITAL NETWORK ANALYZER UPGRADE FOR LLRF CONTROLLERS*

Samson Mai†, Arshdeep Singh, Geetha Narayan, Kevin Mernick, Freddy Severino
Brookhaven National Laboratory, Upton, NY, USA

Abstract

Digital Network Analyzers (DNA) have been implemented in many Low-Level Radio Frequency (LLRF) systems, notably NSLS-II and CERN, to help tune feedback loops. DNA characterizes feedback loops by measuring the frequency-dependent magnitude and phase transfer functions. It enables the measurement of open loop gains, gain/phase margins, and loop delays to help fine-tune feedback loops. An FPGA-based DNA has been developed and integrated into the current Relativistic Heavy Ion Collider (RHIC) LLRF infrastructure. Its performance has been tested with an implementation of one-turn delay feedback (OTFB) on the bench to maximize gain and stability. The DNA has been used to characterize a RHIC 28 MHz cavity in a RHIC Accelerator Physics Experiment (APEX) to test transient beam loading compensation strategies.

INTRODUCTION

Vector Network Analyzers (VNAs) are often used in particle accelerators to measure the resonant frequencies and quality factors of radiofrequency accelerating cavities. They can also be used to measure loop gain and phase margins to ensure loop stability. Other facilities have integrated network analyzers into their LLRF systems. The team at CERN have used a complex noise profile as stimulus to tune feedback loops using models [1]. NSLS-II uses a swept sine-wave network analyzer to see effects of different proportional gain on beam and cavity [2]. The proposed digital network analyzer for EIC combines the two methods to use a swept sine-wave to excite the system and perform regression-based tuning of feedback loops.

There are a few differences between VNAs and DNAs. VNAs are large, expensive, and require physical electrical connectors to interface with RF systems. On the other hand, the DNA is a LLRF tool that is embedded into the RHIC cavity controller FPGA. No extra hardware is needed as RHIC LLRF field controllers already contain DACs and ADCs. The main cost of the DNA is associated with the development of the firmware and software.

DNA has been designed on a LLRF chassis containing a DAC and ADC daughtercard. Post-processing of data is done in Python to extract system responses. Bench-top testing compared DNA performance to that of the VNA. The DNA was used to tune and test the OTFB with a cavity simulator. We utilize the opportunity to test on RHIC systems while RHIC is still running. The long-term goal is to integrate the network analyzer into the EIC LLRF controllers.

ARCHITECTURE

A network analyzer must have 3 basic functionalities - to send a stimulus, capture the system output, and perform post-processing.The DNA uses an FPGA and a DAC to generate a chirp as the stimulus. An ADC and FPGA are used to filter and write the data into memory. The data is then read out to a Linux machine where a Python script calculates the magnitude and phase response. All FPGA signal generation and processing is performed in baseband using IQ modulation and demodulation.

Stimulus

The baseband linear chirp is generated using a simple quarter sine-wave look up table (LuT) with a linearly increasing frequency tuning word. Both in-phase and quadrature components are created using two LuTs, one with a 90° offset. The chirp sweeps through the whole bandwidth of negative and positive frequencies. The chirp is flexible with programmable start/end frequency, duration, and attenuation.

Fig. 1 shows 3 different points of chirp injection into the control loop. The 3 DNA operating modes control a mux such that only one point of injection is used at any given time. The different operating modes, when used by itself or in conjunction with one another, are able to extract the controller, plant, or sensitivity transfer functions.

ADC Capture

On the ADC side, the ADC signal gets IQ demodulated to baseband, low-pass filtered, and decimated. The FPGA uses a Xilinx Multi-Port Memory Controller (MPMC) IP core to write data to DDR2 memory. The DDR2 clock runs at 200 MHz, but the write FIFO buffer is filled at a rate of 10 MHz or less. Both the in-phase and quadrature components of the chirp and ADC data gets recorded in memory to be later recovered for post-processing. An event sent through the Update Link is used to trigger the capture of input and output data [3].

Post-processing

Python scripts are used to bring all the DNA firmware and software components together. The script is able to control DAC chirp parameters, ADC decimation rates and post-processing. Once the data is pulled from DDR2, the Python script performs a complex FFT on the input and output IQ data.

Eq. (1) and Eq. (2) shows the calculations used to determine the magnitude and phase response. X(s) represents

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

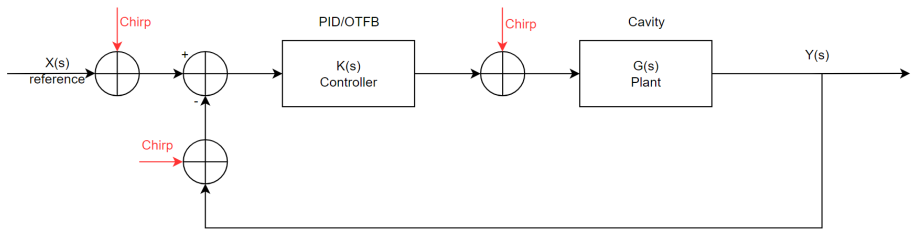

Figure 1: Diagram shows a negative feedback control loop with a controller and plant. In each operating mode, a chirp injection (shown in red) is summed into a different point in the LLRF control loop.

the FFT of the complex chirp signal injected into the DAC while Y(s) represents the FFT of the complex system output. The magnitude response is estimated to be the cross spectral density divided by the power spectral density. The noise present at the output will be uncorrelated to the input. When averaging multiple measurements, the cross spectrum estimation will average out the uncorrelated output noise [4].

$$Mag = 20 * log_{10} \left| \frac{Y(s) * X(s)^*}{X(s) * X(s)^*} \right| \quad (1)$$

$$Phase = arctan \left(\frac{Y(s) * X(s)^*}{X(s) * X(s)^*} \right) \quad (2)$$

Different Operating Modes

As shown in Fig. 1, K(s) represents the PID and OTFB response. G(s) represents the cavity response.

Reference Injection at the reference is used to measure the complementary sensitivity function:

$$\frac{K(s) * G(s)}{1 + K(s) * G(s)} \quad (3)$$

After PID Injection after the PID is used to measure the input disturbance sensitivity function:

$$\frac{G(s)}{1 + K(s) * G(s)} \quad (4)$$

Negative Feedback Path When injecting in the negative feedback path, the feedback loop is opened to get the open loop gain K(s)*G(s). When the feedback controller is turned off, we are able to extract the cavity response G(s). With these two terms, we are able to extract the controller response to calculate any other transfer function.

PERFORMANCE

To test the DNA, a cavity impedance measurement was taken of a cavity simulator. First, the DNA is calibrated by adjusting the ADC scaling and phase offset. Next, the PID and OTFB are turned on and a sweep is triggered to calculate the system response of the cavity simulator. A Copper Mountain TR1300/1 VNA is used to get a comparable measurement of the cavity simulator under the same settings. The center frequency is 29.2 MHz with a span of 2.5 MHz. Both analyzers use 16,000 points to perform 10 averages of the measurement. This gives a frequency resolution of 156.25 Hz. A smaller frequency resolution can be achieved on the DNA by increasing the number of points up to 100,000 and increasing the ADC decimation rate.

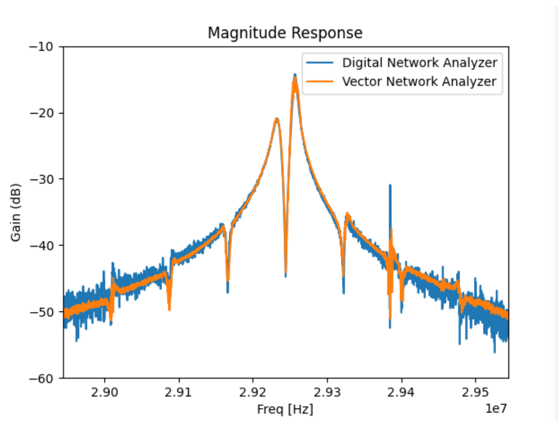

Figure 2: Plot shows the magnitude response of the DNA (blue) and VNA (orange).

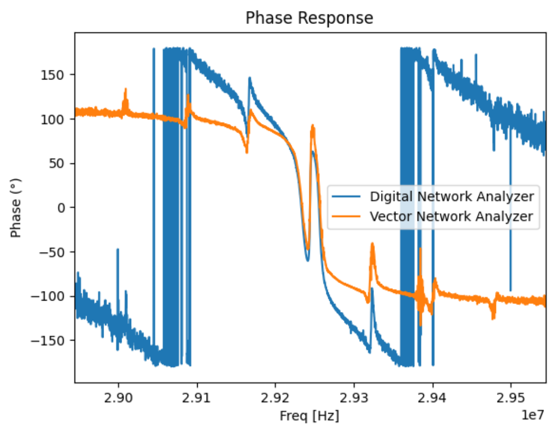

Figure 3: Plot shows phase the response of the DNA (blue) and VNA (orange).

Fig. 2 and 3 shows the comparison between the DNA and VNA measurement. In the tests, the OTFB and PID were turned on with the same loop settings. The DNA magnitude response is almost identical to the VNA's. The DNA measurement can be improved with more averaging. The difference in phase response can be attributed to differences in loop delay.

APPLICATIONS

The transfer function measurement is used to tune the OTFB loop using regression optimization. Python is used to compare a model of the system response to the network analyzer measurement. The script runs regression functions to fit the model to the measurement. From the fitted model, we can derive the loop delay to setup the feedback loop [5].

The Digital Network Analyzer has been verified on a RHIC 28 MHz cavity. During a RHIC APEX, the DNA was used to take a measurement of a cavity with the analog direct RF feedback loop closed. We expect a peak at the center frequency from the reference, and a lower impedance with the direct RF feedback loop on. We noticed an asymmetrical peak 200 kHz from the carrier frequency as shown in Fig. 4. A VNA was used to cross-check the results of the DNA. Upon further analysis on the open loop gain, we found that the fast feedback phase margins were off by 40°. Cable delay of 2.5 ns was added to the amplifier chain to bring the system back closer to optimal phase margins.

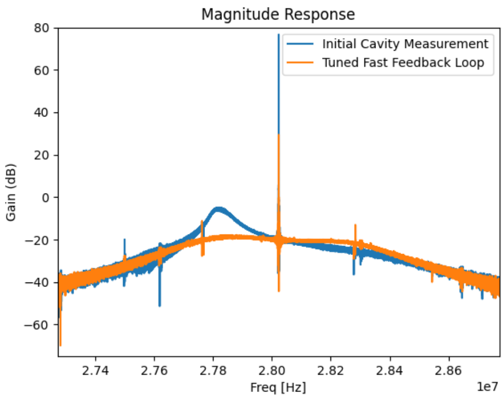

Figure 4: Magnitude response before (blue) and after (orange) tuning the fast feedback loop.

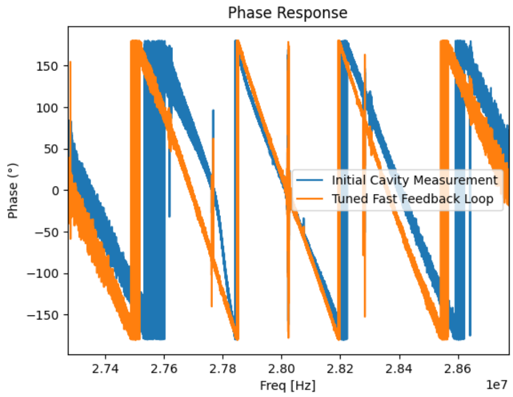

Figure 5: Phase response before (blue) and after (orange) tuning the fast feedback loop.

CONCLUSION

The Digital Network Analyzer has been tested on a RHIC operational cavity. It is flexible and cheaper compared to a physical off-the-shelf Vector Network Analyzer. Performance have been proven both in lab tests with a cavity simulator and operational tests with a RHIC 28 MHz cavity. Future work includes adding Nyquist plots for stability, improvement of scripts for better UI, and integration into the EIC Common Hardware Platform.

REFERENCES

[1] J. C Molendijk, "An Extended Base-band Network Analyzer in the Digital LLRF", proceedings of the LLRF11 Workshop, Hamburg, Germany

[2] C. Marques et al., "Status of the NSLS-II LLRF System", proceedings of PCaPAC2016, Campinas, Brazil

[3] T. Hayes, K.S. Smith, F. Severino, “A Deterministic, Gigabit Serial Timing, Synchronization and Data Link for the RHIC LLRF,” PAC 2011, New York, NY (2011)

[4] Brüel & Kjær, “Technical Review No. 2, 1984. Dual Channel FFT Analysis (Part I)”

[5] A. Singh et al., "OTFB Implementation for the EIC".