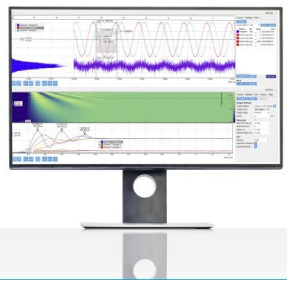
Review of Scientific Instruments

DR-RINS: Digital real-time relative intensity noise suppressor for pump–probe spectroscopy and microscopy

Cite as: Rev. Sci. Instrum. **92**, 023704 (2021); https://doi.org/10.1063/5.0032376
Submitted: 07 October 2020 . Accepted: 13 January 2021 . Published Online: 16 February 2021

Saurabh Gupta, Erkang Wang, Steven Derrien, and Jesse W. Wilson



AIP Publishing

Review of Scientific Instruments ARTICLE scitation.org/journal/rsi

DR-RINS: Digital real-time relative intensity noise suppressor for pump–probe spectroscopy and microscopy

Cite as: Rev. Sci. Instrum. 92, 023704 (2021); doi: 10.1063/5.0032376
Submitted: 7 October 2020 • Accepted: 13 January 2021 •
Published Online: 16 February 2021

Saurabh Gupta,[1] Erkang Wang,[1] Steven Derrien,[2] and Jesse W. Wilson[1,3,a)]

AFFILIATIONS

[1] Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, Colorado 80523, USA
[2] Univ Rennes - IRISA, Rennes 35700, France
[3] School of Biomedical Engineering, Colorado State University, Fort Collins, Colorado 80523, USA

[a)] **Author to whom correspondence should be addressed:** jesse.wilson@colostate.edu

ABSTRACT

Relative intensity noise (RIN) inherent in fiber lasers poses a serious obstacle to their use in pump–probe spectroscopy and imaging. RIN can be removed through an analog balanced detector, or, as we have previously shown, software adaptive noise cancellation (ANC) on digitized signals. One major drawback to software ANC is the added time required for digitizing and post-processing. In this article, we describe a design for ANC on a field-programmable gate array (FPGA), making use of high-level synthesis tools and fixed-point arithmetic to achieve real-time laser RIN suppression at 25 MHz sample rates. Unlike the software-ANC approach, the FPGA-ANC device can serve as a dedicated drop-in denoiser, placed between the detectors and a commercial lock-in amplifier. We demonstrate its application to transient absorption spectroscopy and microscopy, lowering the noise floor to ~17 dB above the shot noise limit. Furthermore, we demonstrate a dramatic improvement in data acquisition time from ~6 h to ~5 min in a real-time imaging scenario.

I. INTRODUCTION

Transient absorption and stimulated Raman scattering microscopy are nonlinear optical methods that obtain chemical contrast from electronic and vibrational dynamics.[1] Applications have ranged from label-free molecular contrast in biological tissues to defect characterization and charge transport mapping in novel two-dimensional nanomaterials.[1–4] Both techniques rely on sensing miniscule perturbations to a probe laser beam, induced by a pump beam, and are thus highly susceptible to laser relative intensity noise (RIN). In the case of bulk laser sources (e.g., a Ti:Al_2O_3 ultrafast oscillator), most of the RIN can be rejected by modulating the pump at >1 MHz and employing lock-in detection on the probe.[5] In the case of fiber laser sources, however, broad bandwidth and high-frequency RIN necessitates active noise cancellation through balanced detection, for example, custom radio frequency (RF) analog electronics involving a variable-gain amplifier and proportional-integral-derivative (PID) controller.[6] Recently, we introduced a software-based scheme that uses a high-speed analog to digital converter (ADC) and adaptive filtering for active RIN cancellation.[7] Compared with the analog electronics solution, the software approach is convenient in terms of sharing, replication, and fine-tuning. However, as it was limited by ADC noise and based on post-processing of acquired data, it was impractically slow for all but proof-of-concept imaging tests in our previous work.

In this article, we overcome these speed limitations by implementing the adaptive filter on a field-programmable gate array (FPGA) for real-time noise cancellation.[8] FPGAs, due to their capacity for low latency and high bandwidth computations, are well-suited for real-time signal processing on datastreams directly from high-bandwidth (i.e., 10–500 MSPS) ADCs, enabling, for example, low cost and energy efficient software-defined radio[9] and lock-in amplifiers (LIAs) for scientific applications.[10–13]

Here, we implement a digital real-time RIN suppressor (DR-RINS) on a development board (Red Pitaya STEMlab 125-14) that includes high-speed ADCs and digital-to-analog converters (DACs) alongside an FPGA with an on-board microprocessor (Xilinx Zynq 7010 SoC). The device is set up as a drop-in denoiser that is inserted

between the photodetectors and a lock-in amplifier and coded using high-level synthesis (HLS). For this implementation, we performed lock-in detection in software and showed compatibility with a commercial lock-in amplifier. We demonstrate its application to transient absorption microscopy of a crystalline powder. Compared to our previous all-software implementation,[7] this FPGA implementation has a lower noise floor and provides real-time RIN cancellation, reducing the image acquisition time from ~6 h down to ~5 min.

II. IMPLEMENTATION OF FIXED-POINT ADAPTIVE FILTER ON FPGA

For details on the theory of applying adaptive filtering to pump–probe microscopy, we refer readers to our previous work on software ANC.[7] In brief, the adaptive filter used here is a digital finite impulse response (FIR) filter, operating on a reference measurement of the laser RIN, x, whose coefficients are continually updated by a feedback loop to minimize the difference, e, between the filter's output, y, and the measured probe signal, d. In this way, e contains a copy of the probe signal with RIN canceled out. This result is then fed to a lock-in amplifier to detect the pump–probe signal by synchronous demodulation at the pump modulation frequency.[2]

To implement ANC in real time on Red Pitaya's (RP) FPGA, we made use of Pavel Demin's software-defined radio project as a starting point.[14] A high-level system diagram of our implementation is shown in Fig. 1. An eight-tap digital adaptive noise canceller (ANC) was implemented using the least mean squares (LMS) algorithm. Individual modules were connected using an IP integrator (Xilinx Vivado Hlx 2018.2). All signals pass between modules using the AXI streaming protocol. Flow of data within the FPGA modules is as follows: Two 14-bit ADCs on-board operating at 125 MSPS digitize x and d that are low-pass filtered with cutoff at ~8.2 MHz and then decimated (Xilinx FIR compiler 2; M = 5). The resulting 25 MSPS datastream is then passed to individual first in, first out (FIFO) buffers for a clock domain conversion from 125 MHz to 25 MHz. The HLS produced adaptive filter IP (called the LMS module) is clocked at 25 MHz, which receives these x and d. The IP produces an error output at 25 MSPS that is passed through a FIFO to convert

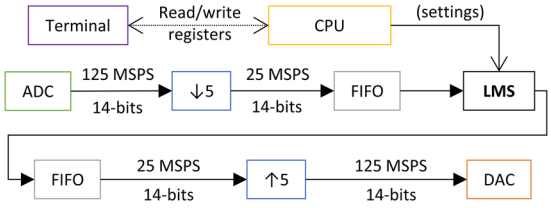


FIG. 1. System diagram of IP components and interconnects placed on the FPGA. The digitized datastream from the ADC is downsampled, passed through an LMS adaptive filter, upsampled, and then passed to the DAC for an analog output. Clock domain crossings are handled with FIFO buffers. The on-chip CPU controls parameters of the LMS module through an AXI-lite interface and can be accessed through a command-line terminal.

the clock domain from 25 MHz to 125 MHz. Clock domain crossing FIFOs are implemented using Xilinxs's FIFO generator (BRAM implementation with independent clocks using eight-sync stages). An interpolator (Xilinx FIR compiler 2; L = 5) is used to upsample the error output before passing to a DAC on-board operating at 125 MSPS. A gain setting is implemented by right shifting a 14-bit window, effectively amplifying the error signal before outputting via a DAC. This serves the purpose of bringing the signal above the noise floors of the DAC and the next physical device downstream (i.e., a lock-in amplifier). The LMS algorithm step-size and output gain are manipulated using registers controlled by the central processing unit (CPU). A program running on the CPU in a loop continuously reads/updates these registers as provided by the user via a command-line terminal. The LMS module constantly reads these parameters every clock cycle through an AXI-lite interface.

The LMS module was coded in C/C++ using the HLS methodology, which speeds hardware design by making it easier to validate the algorithm, automatically handling pipelining and loop unrolling to meet timing requirements and reducing the effort to change parameters and explore design space. Pre-processor directives (*#pragma*) were used to convey to the HLS compiler details about parallelization and logic implementation. *#pragma HLS pipeline II* = 1 sets the design throughput to one sample per

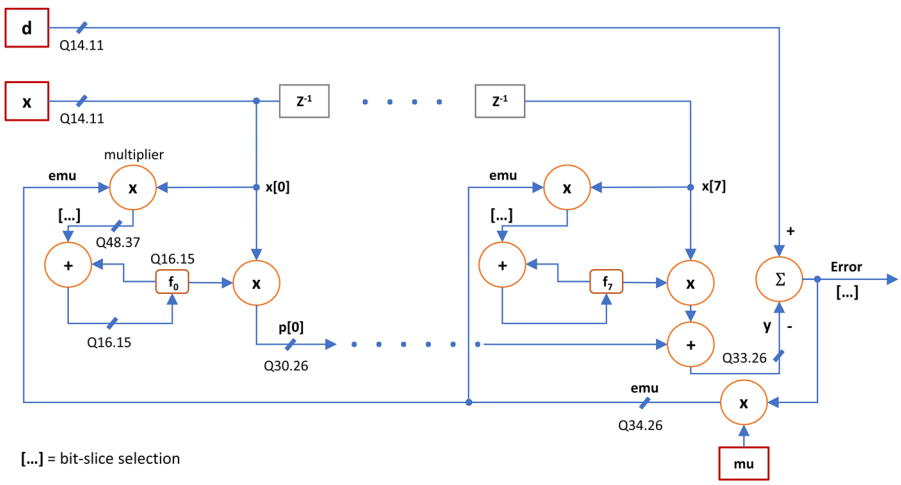


FIG. 2. Signal flow diagram of the LMS module.

clock cycle and ensures the adaptive filter updates its output and filter coefficients before the next sample arrives at the input. *#pragma array_partition* maps arrays (i.e., the tapped delay line $x[n] \ldots x[n-7]$ and filter coefficients $f[0] \ldots f[7]$) into multiple registers rather than one large memory (block RAM) for simultaneous access. *#pragma HLS unroll* exposes parallelism by enabling all the filter taps to be executed in the same clock cycle.

Internal details of the LMS module are illustrated in Fig. 2 using a signal flow diagram. The bit-widths are represented in a Qn.m format, where n is the total number of bits and m is the number of fractional bits after an assumed decimal position. Arithmetic was performed in the signed fixed-point format using Xilinxs's *ap_int* datatype. While fixed-point arithmetic is significantly faster than floating-point, it comes at the risk of round-off and overflow errors due to limited precision and range. To minimize this risk, the number of integer and fractional bits for the filter coefficients, products, and accumulators were selected with the assistance of the MATLAB fixed-point toolbox. We selected the minimum bit-widths that closely matched the output of a floating-point simulation of LMS filtering on pre-recorded pump–probe data. This approach generally prevented overflow, except in areas where transmissivity was significantly higher than the pre-recorded data. To address it, an additional guard bit can be added to the coefficients, or electronic (after sample)/optical (before sample) attenuation of the probe beam can avoid coefficient overflow.

The HLS design was simulated using C/C++ test-benches to verify functional correctness and that the output matches the MATLAB simulation. The HLS tool synthesized a hardware description of the module and a register-transfer level (RTL) model of the logic implementation. This generated RTL was then verified using C/C++ test-benches via co-simulation, which simulates the behavior of the scheduled hardware as it would run on the FPGA in the presence of a clock. The timing report showed that the module could run at a clock rate of 25 MHz with the requested sample initiation interval (II) of one clock cycle.

III. EXPERIMENTAL SETUP

Figure 3 shows the experimental setup. The FPGA board digitizes the probe and reference signals from a transient absorption microscope, performs noise cancellation, and outputs an analog signal through one of its DAC channels. This noise-canceled probe signal is then either processed by a hardware lock-in amplifier (Moku:Lab, Liquid Instruments) or captured by a high-speed data acquisition (DAQ) device (Analog Discovery Studio, Digilent) for noise analysis and processing with a software lock-in algorithm. Unlike the previous all-software implementation,[7] all the signal processing can now occur in dedicated hardware, enabling real-time display of pump–probe signals and images.

The pump–probe microscope setup was described in Ref. 7. Pump and probe pulses at 530 nm and 480 nm, respectively, with a cross correlation of 800 fs, were generated by a two-color laser source described in Ref. 15. Pump power and probe power out of the two-color laser source were both 10 mW. The pump was modulated with a square wave at 1.5 MHz with an acousto-optic modulator (AOM). Before the microscope, a 50/50 beam splitter directed a portion of the probe beam toward a reference photodiode, which was connected to ADC CH1. After the microscope, the transmitted probe beam was detected with a second photodiode and connected to ADC CH2. In all the experiments using the DR-RINS, the feedback coefficient was set to $\mu = 2 \times 10^{-6}$ and the total gain from the ADC to the DAC was $2^5 = 32$.

For image acquisition, we used transistor–transistor logic (TTL) synchronization from the y-scan mirror to trigger the DAQ while a 3.5 KHz resonant scan mirror completed the x-scan. Translation along the y-axis was achieved by stepping a mechanical stage by 1 μm after one acquisition, repeated for 100 lines. The image field of view (FOV) was $75 \times 100\ \mu m^2$. To further improve the signal-to-noise ratio (SNR), we captured and averaged data for a total of ten passes. Data without DR-RINS were obtained by directly digitizing the output using DAQ, from the probe photodetector.

A MATLAB script was used as a lock-in amplifier (LIA) to extract modulated amplitude at the modulation frequency. The modulation reference was derived from the TTL sync output of the function generator driving the pump AOM. The pump modulation TTL sync was captured on the DAQ analog CH2. This square wave was then converted to in-phase (I) and quadrature (Q) sinusoids by a narrow bandpass FIR filter at the fundamental, followed by a Hilbert transform and a normalization step to eliminate amplitude variations. The resulting complex vector was rotated

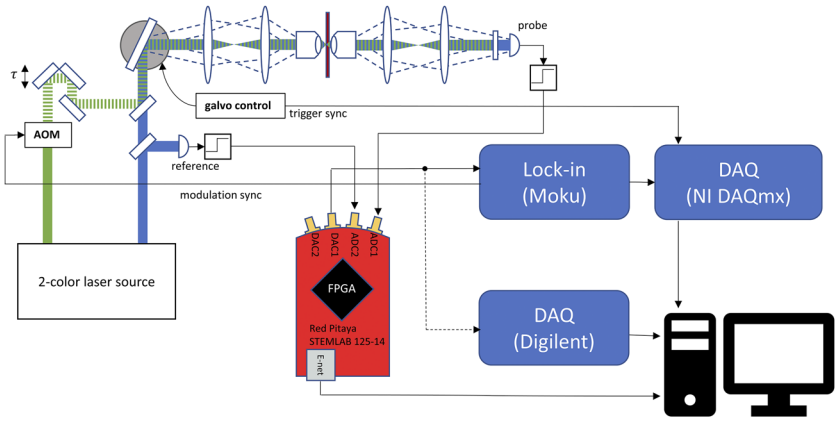


FIG. 3. Experimental setup. A two-color femtosecond laser source supplies pump (green) and probe (blue) pulse trains to a transient absorption microscope. The probe is sampled before (reference) and after passing through the microscope (probe). These signals pass through an adaptive noise canceller running on the FPGA. The resulting output is fed to a hardware lock-in amplifier (Moku) and synchronized to the pump modulation frequency or a high-speed DAQ device (Digilent). The hardware lock-in output is captured by a multi-channel DAQ device (NI DAQmx), synchronized to the x/y galvanometer scanners for real-time imaging.

by multiplication with $e^{i\phi}$ to bring the lock-in X channel in phase with absorptive signals (i.e., two-photon absorption, excited-state absorption). The real and imaginary parts of the reference oscillator were then individually mixed with the output to obtain the I and Q products. These I and Q products were then passed through a cascaded integrator-comb (CIC) decimator with $R = 128, M = 4$, and $N = 2$ to yield X and Y lock-in channels. A simple MATLAB code snippet that demonstrates this process is provided in the supplementary material.

IV. RESULTS

A. Spectroscopic (non-imaging) measurement results and noise analysis

Figure 4 shows the pump–probe delay scan of a single, uniform $Bi_4Ge_3O_{12}$ crystal, acquired with and without DR-RINS in place. At each probe delay, we acquired a 327.68 μs window of the signal using DAQ and calculated the power spectral density (PSD) with MATLAB's *pwelch()* function. During these measurements, the resonant x-scanner was enabled to prevent heat from building up at the focal spot and give the adaptive filter some minor transmissivity variations to keep up with. As expected for our conditions (530 nm pump, 480 nm probe), the signal at the pump modulation frequency (1.5 MHz) indicates a non-degenerate two-photon absorption that traces the cross correlation of the pump and probe pulses.[7]

Figure 5 shows the power spectrum of the DR-RINS output along with the electronics noise floor and shot noise floor. The total SNR of the pump–probe signal after the DR-RINS is +15 dB. The total gain (32×) through the FPGA places the shot noise floor on par with the input noise of DAQ. However, as can be seen from the red line, the overall noise floor of the DR-RINS is around 17 dB above the shot noise floor. Given that the Red Pitaya ADC noise floor[16] is significantly lower, around 3.5 dB above the shot noise floor, we conclude that even with a less noisy photodetector, this implementation is ultimately limited by round-off error inherent to fixed-point computations.[17]

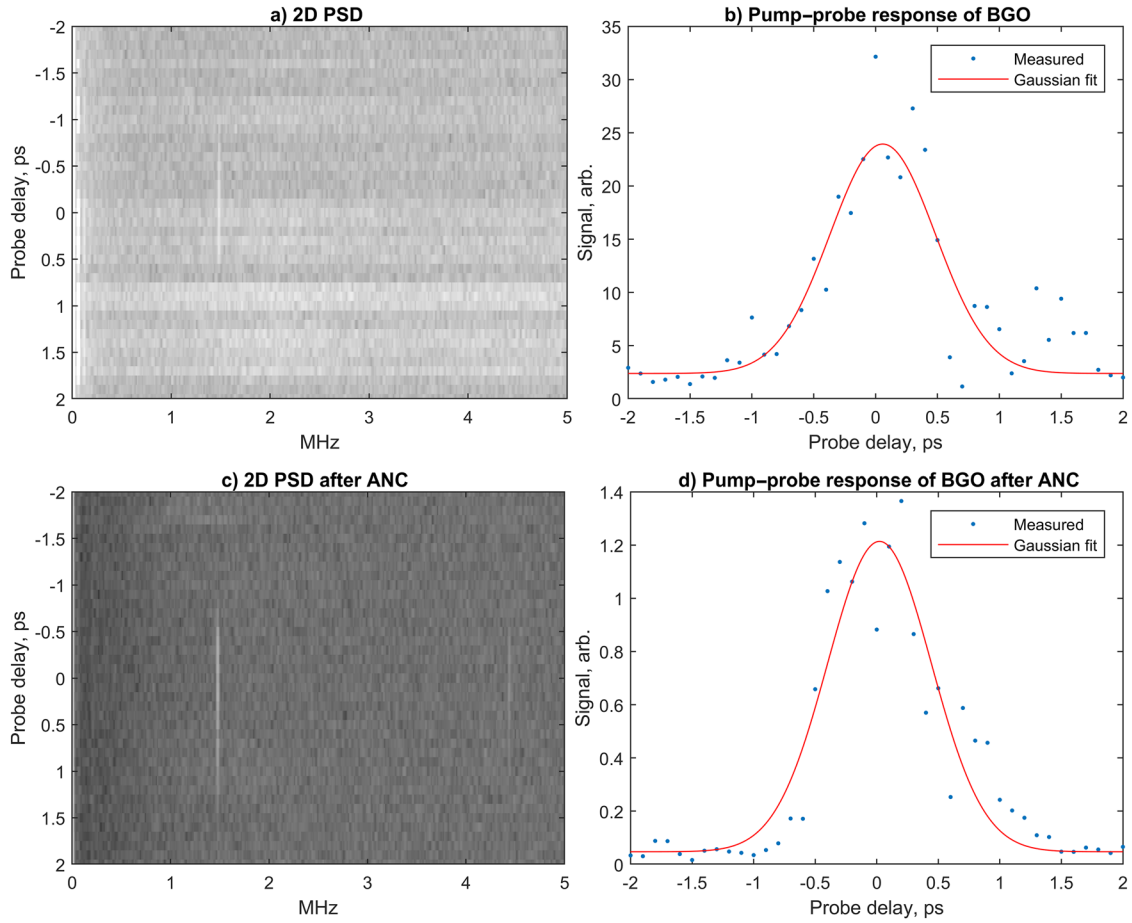


FIG. 4. Pump–probe delay scan of two-photon absorption response in BGO. (a) Power spectral density (PSD) of the probe photodiode signal with respect to pump–probe delay, without DR-RINS. (b) PSD at 1.5 MHz pump modulation frequency, without DR-RINS. (c) PSD of the probe with the DR-RINS, showing lower noise floor and making the 4.5 MHz pump modulation harmonic visible. (d) PSD with the DR-RINS at 1.5 MHz pump modulation frequency.

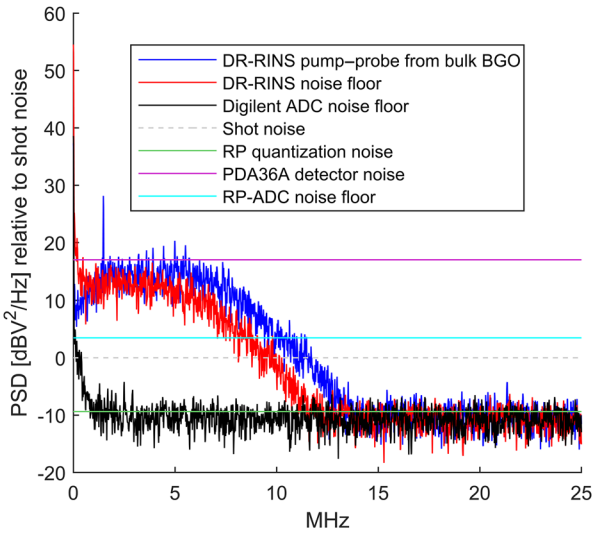


FIG. 5. Power spectra of DR-RINS output, DR-RINS overall noise floor, Digilent ADC noise floor, shot noise floor, Red Pitaya (RP) ADC quantization noise, PDA36A detector noise floor, and RP-ADC noise floor. Measured at $\mu = 2^{-6}$ and *gain* = 32.

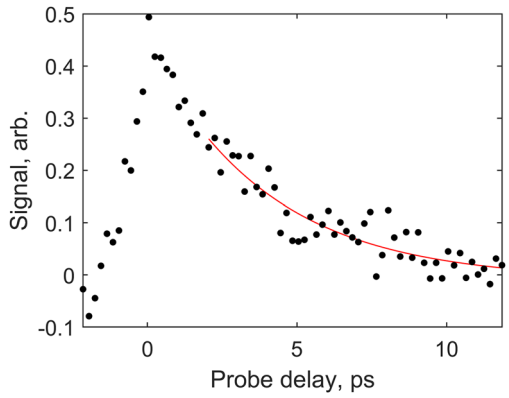


FIG. 6. Transient absorption signal of mitochondrial respiratory chain proteins. The red line shows fit to an exponential decay with a 4 ps time constant.

B. Real-time performance enables measurements of weak signals from biological samples

Because the FPGA performs its computations on the digitized datastream in real time, the output of the adaptive filter can be connected directly to a hardware lock-in amplifier, and the resulting signal can be used to assist in alignment and sample positioning. Figure 6 shows transient absorption signals acquired from mitochondrial respiratory chain enzymes, suspended in an electrophoresis gel (see Ref. 18 for details on sample preparation). These gels normally contain a dye (Coomassie blue) attached to the proteins, to provide a charge to draw the protein through the gel when a voltage is applied. The dye is also convenient for visually locating the protein band within the sample. In this sample, however, the dye was removed with a standard destaining solution (20% methanol and 10% acetic acid in water) to ensure the measured signal was from

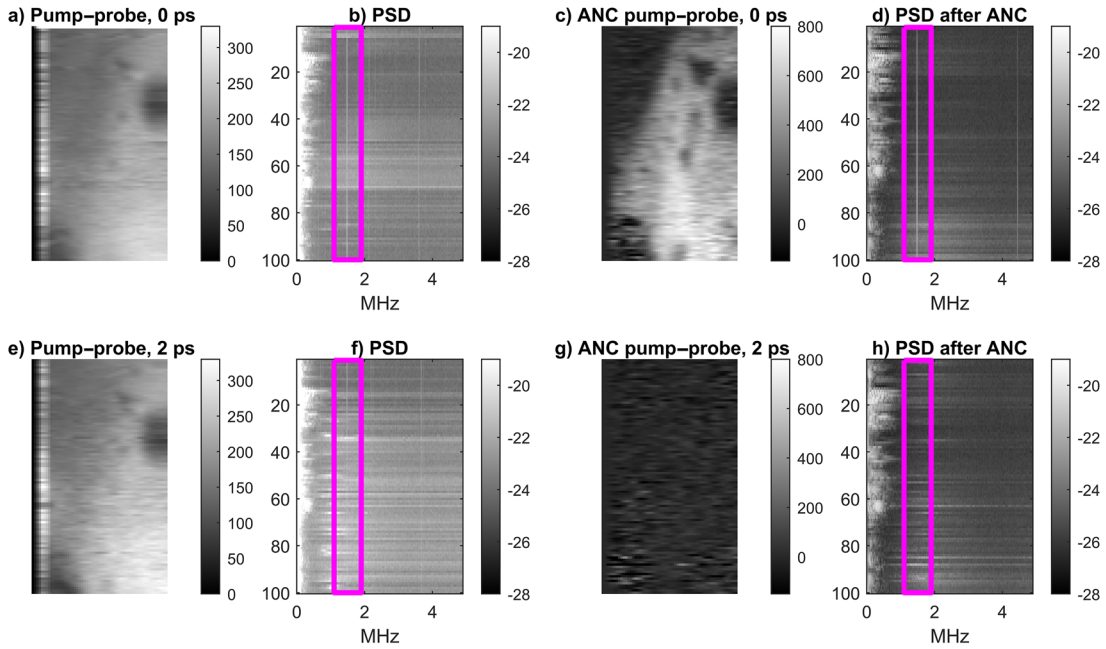


FIG. 7. Imaging results at $\tau = 0$ ps probe delay (top row), including pump–probe without ANC (a), PSD of each line without ANC (b), pump–probe after ANC (c), and PSD of each line after ANC (d). Imaging results at $\tau = 2$ ps probe delay (bottom row), including pump–probe without ANC (e), PSD of each line without ANC (f), pump–probe after ANC (g), and PSD of each line after ANC (h). The field of view (FOV) is 75×100 μm^2.

the protein only and not the dye. However, the resulting destained gel appears transparent, making it difficult to locate the protein band within the sample by eye. In addition, the excited-state absorption from these respiratory pigments is significantly weaker than the two-photon absorption signal from BGO, making it difficult to locate the protein band by its pump–probe response. While the previous software implementation[7] was too slow to aid in locating a signal, the real-time feedback from the FPGA and hardware lock-in made it feasible to locate a signal and acquire the data shown in Fig. 6. In this case, the signal is still too weak for high frame rate imaging, though the resonant galvo was running to minimize sample heating.[19]

C. Imaging through a software lock-in

To demonstrate the ability of the DR-RINS to maintain balance during a high-speed imaging scenario, a sample of crushed BGO was placed under the objective as described previously.[7] Figure 7 shows imaging results at 0 ps and 2 ps probe delays, both with and without DR-RINS in place. Each scan line was acquired for ten repetitions, and the resulting lock-in outputs and PSDs for each scan line were averaged across all repetitions. Consistent with our previous findings, the lock-in, without ANC, sees a significant amount of high-frequency RIN within its passband [magenta box, Figs. 7(b) and 7(d)]. As a result, an image is formed of the sample transmissivity and has no dependence on probe delay [Figs. 7(a) and 7(e)]. By contrast, with the DR-RINS, noise is significantly reduced across a broad range of RF frequencies [Figs. 7(d) and 7(h)], making the 1.5 MHz pump modulation and its 4.5 MHz harmonic clearly visible in the PSD. In addition, the lock-in output recovers a clear dependence on probe delay [Figs. 7(c) and 7(g)], consistent with the delay scan (Fig. 4).

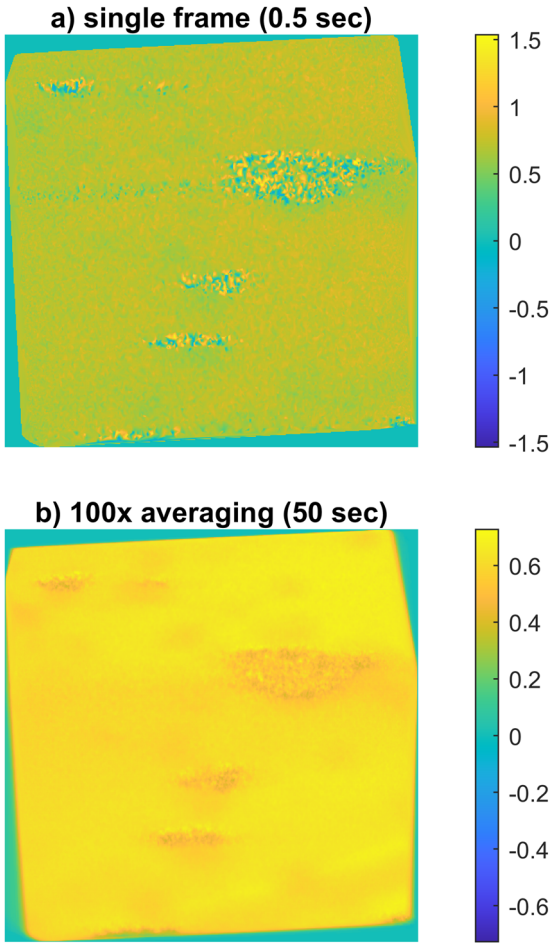


FIG. 8. Real-time imaging results, using the FPGA as a prefilter and of a commercial lock-in amplifier at 4 MHz modulation. (a) Single frame, acquired in 0.5 s. (b) 100-frame average. The field of view is ~100 × 100 μm^2.

D. Real-time imaging with a hardware lock-in

Finally, we tested the DR-RINS system as a drop-in pre-filtering device in front of a hardware lock-in amplifier to enable real-time imaging. As before, the sample is crushed BGO, and the probe delay is set to $\tau = 0$. The output of the lock-in, along with scan position monitor signals, was collected with a multi-channel DAQ device. An image was formed by using the MATLAB *griddata*( ) function to map the signal to the corresponding x, y coordinates. The x-resonant scanner was running at a 3.5 kHz rate and the y-galvanometer was running at a 2 Hz rate, for an overall frame rate of 2/s. The frame rate acquired by the computer was lower, however, because of the time required to stop and start the DAQ task during each iteration of the data capture loop. Figure 8 shows single acquisition, along with a 100-frame average, acquired in <5 min.

V. CONCLUSION

To summarize, our results on BGO visibly show an enhancement in the SNR, obtained in real time using the DR-RINS system that produces an analog output compatible with a conventional lock-in amplifier. Though the system was limited by detector noise, the device enabled transient absorption imaging under conditions that are impossible to image using a lock-in amplifier alone (i.e., high levels of RIN combined with effects due to fast, resonant scanning of the beam). Unlike the previous software-based approach, the FPGA based denoiser provides adaptive noise canceling in real time, enabling the operator to locate and acquire weak signals in addition to significantly faster imaging with a noisy laser source. By performing both the ANC and lock-in detection in hardware, data collection time for a pump–probe image (100× averages) was reduced from ~6 h down to ~5 min—a dramatic improvement over the all-software implementation. Even in scenarios where high averaging needs to be done to overcome low SNR, the ability to cancel noise in real-time and acquire individual frames with a resonant galvo offers a significant reduction in heating at the focal spot, an important consideration when performing pump–probe microscopy on resonance with absorbing molecules.[19] We anticipate this plug-and-play RIN denoising device to find applications in any experimental technique that relies on detecting small perturbations to a probe laser, such as transient absorption, stimulated Raman scattering, and photothermal microscopy.

SUPPLEMENTARY MATERIAL

The supplementary material contains a code listing for the software lock-in amplifier.

ACKNOWLEDGMENTS

The authors thank Xilinx for donation of Vivado licenses, Digilent for donation of DAQ, Kalyn Specht and Adam Chicco for the electrophoresis gel samples, and Randy Bartels for the use of the amplified fiber laser system. This research was partially supported by funds from the Boettcher Foundation's Webb-Waring Biomedical Research Program and the Department of Energy (Grant No. DE-SC0017200).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1]Y. Zhu and J.-X. Cheng, J. Chem. Phys. **152**, 020901 (2020).
[2]M. C. Fischer, J. W. Wilson, F. E. Robles, and W. S. Warren, Rev. Sci. Instrum. **87**, 031101 (2016).
[3]C. Zhang, D. Zhang, and J. X. Cheng, Annu. Rev. Biomed. Eng. **17**(1), 415–445 (2015).
[4]E. M. Grumstrup, M. M. Gabriel, E. E. M. Cating, E. M. Van Goethem, and J. M. Papanikolas, Chem. Phys. **458**, 30 (2015).
[5]P. Tian and W. S. Warren, Opt. Lett. **27**, 1634 (2002).
[6]C. W. Freudiger, W. Yang, G. R. Holtom, N. Peyghambarian, X. S. Xie, and K. Q. Kieu, Nat. Photonics **8**, 153 (2014).
[7]E. Wang, S. Gupta, and J. Wilson, J. Biomed. Opt. **25**, 106503 (2020).
[8]R. Finger, F. Curotto, R. Fuentes, R. Duan, L. Bronfman, and D. Li, Publ. Astron. Soc. Pacific **130**, 025002 (2017).
[9]R. Akeela and B. Dezfouli, Comput. Commun. **128**, 106–125 (2018).
[10]J. W. Wilson, J. K. Park, W. S. Warren, and M. C. Fischer, Rev. Sci. Instrum. **86**, 033707 (2015).
[11]E. Flater, A. C. Mugdha, S. Gupta, W. A. Hudson, A. A. Fahrenkamp, J. P. Killgore, and J. W. Wilson, Meas. Sci. Technol. **31**(11), 115009 (2020).
[12]G. A. Stimpson, M. S. Skilbeck, R. L. Patel, B. L. Green, and G. W. Morley, Rev. Sci. Instrum. **90**, 094701 (2019).
[13]D. M. Harcombe, M. G. Ruppert, and A. J. Fleming, Beilstein J. Nanotechnol. **11**, 76 (2020).
[14]P. Demin, Red Pitya Notes, http://pavel-demin.github.io/red-pitaya-notes/; last accessed September 15, 2020.
[15]S. R. Domingue, R. A. Bartels, A. J. Chicco, and J. W. Wilson, Biomed. Opt. Express **8**, 2807 (2017).
[16]A. C. Cárdenas-Olaya, E. Rubiola, J.-M. Friedt, P.-Y. Bourgeois, M. Ortolano, S. Micalizio, and C. E. Calosso, Rev. Sci. Instrum. **88**, 065108 (2017).
[17]P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, 4th ed. (Springer, New York, 2013).
[18]C. H. Le, L. G. Benage, K. S. Specht, L. C. Li Puma, C. M. Mulligan, A. L. Heuberger, J. E. Prenni, S. M. Claypool, K. C. Chatfield, G. C. Sparagna, and A. J. Chicco, J. Biol. Chem. **295**, 12485 (2020).
[19]E. Wang, S. R. Domingue, R. A. Bartels, and J. W. Wilson, in *Ultrafast Nonlinear Imaging Spectroscopy V*, edited by Z. Liu, I. C. Khoo, D. Psaltis, and K. Shi (SPIE, San Jose, CA, 2017), Vol. 10380 p. 10380Q.