3114
doi:10.1017/S143192762201159X

Microsc. Microanal. 28 (Suppl 1), 2022

Python and FPGA-based Workflow for Automated and Interoperable Scanning Probe Microscopy

Shivaranjan Raghuraman[1*], Yongtao Liu[1], Kyle Kelley[1], Rama K. Vasudevan[1] and Stephen Jesse[1]

[1.] Functional Atomic Force Microscopy Group, Center for Nanophase Materials Sciences, Oak Ridge National Laboratory, Oak Ridge, TN, USA.
* Corresponding author: raghuramans@ornl.gov

Scanning Probe Microscopy (SPM) is arguably the most versatile experimental platform to explore a wide range of material properties in arbitrary environmental conditions. Physical interaction between a nanometer scale tip and any surface of interest enables both imaging beyond the optical diffraction limit and manipulation of matter with unprecedented control. Atomic Force Microscopes (AFM) have pushed the limits of imaging by visualizing single amino acids, mapping force-fields of single molecules and tracking fast events such as atomic diffusion and chemical bond breaking [1-5]. Innumerable adaptations of AFMs have also enabled several material manipulation techniques including scanning thermal lithography, bias mode lithography, force-based lithography and grafting [6-8].

However, SPM or AFM techniques are significantly slower than bulk characterization techniques. AFMs predominantly remain manual requiring consistent human intervention to perform experiments. This results in excessively long experiment times making it challenging to obtain statistically significant measurement. The reliability can be improved by using independent AFM modalities, but this worsens the already slow technique. Further, fabricating features at the nanoscale needs an AFM capable of reacting to changes in the microsecond to millisecond range often requiring significant redesign of existing instruments.

Here, we have developed a fast, interoperable, customizable, and open-source workflow for Automated Experiments in SPM (AE-SPM) using FPGA-based controllers and Python-based control, analyses, and visualization modules. FPGA programming, typically involving specialized software and expertise, is not required to implement our workflow. Instead, a custom-built python package, which we term pyAE, enables the user to quickly assemble experimental procedures from a list of common scanning and biasing schemes. Users can also develop new experiments with any desired combination of scanning and spectroscopy routines. The user can then visualize spatially, and temporally correlated data directly in Jupyter notebooks, where machine learning and Bayesian optimization routines can be directly called into the workflow.

With low latencies and parallel processing capabilities, FPGAs provide us numerous benefits over software-based controls in data read/write speeds and signal timing. The FPGA-based closed-loop scan engine can execute both standard raster scans and unconventional sparse scanning protocols on a wide range of microscopes. The FPGA-acquisition system handles various signals from the microscope and feeds a time averaged, position and time synchronized response to the Python host. Beyond these basic functionalities, the high feedback rate of the FPGA allows us to perform spectroscopy while scanning which allows us a higher level of control on the chemical transformations occurring under the tip enabling new feedback-controlled fabrication techniques such as the ferrobot [9].

To demonstrate the efficiency and control of our workflow, we will show examples of an automated nanofabrication and a feature following experiment. The experiment set up will be discussed in brief walking through the customizability of the workflow. Then, the descriptions of signal and data flow during the experiment will demonstrate the low latency communication between the FPGA and Python. Finally, multi-channel visualizations of the surface will be presented introducing basic data analyses such as feature width and depth measurement.

In summary, the AE-SPM workflow significantly speeds up experiments and makes them more reliable and repeatable, which is crucial to maintain the dominance of SPM based techniques in microscopy. More notably, the instrument-agnostic and open-source nature of the framework makes it readily adaptable to several classes of microscopes beyond AFMs including electron and ion microscopes widening its impact on multiscale materials science.

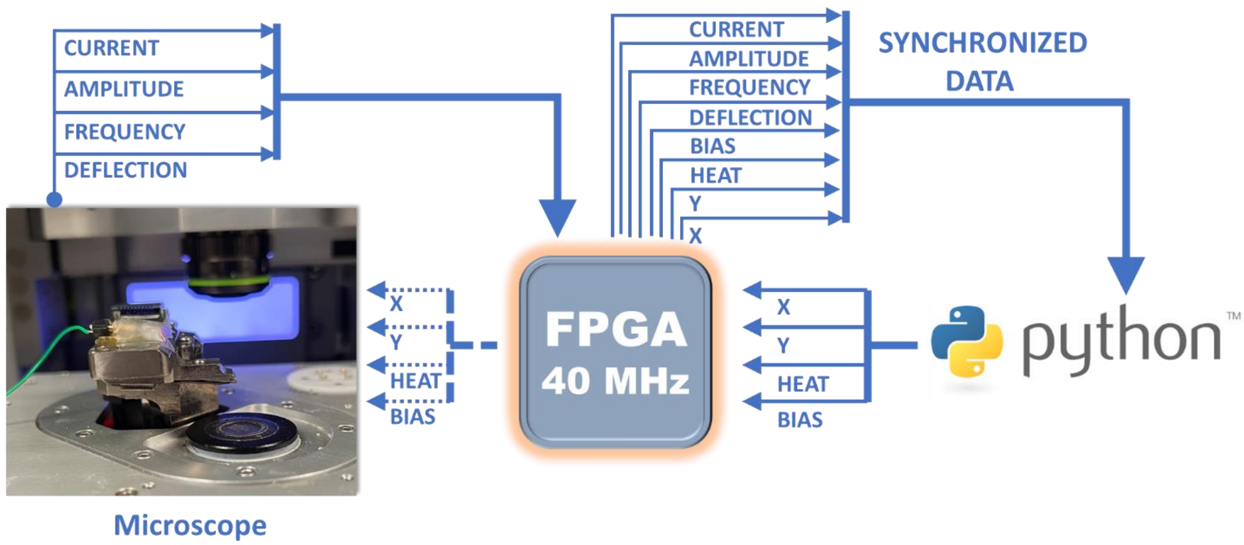

Figure 1: Schematic of AE-SPM Workflow. Python transmits the experiment parameters to the FPGA. The FPGA drives the AFM, acquires signals, synchronizes the data and sends to Python Host.

References:

[1] L Gura et al., Applied Physics Letters **119**(25) (2021), p. 251601.
[2] GR Heath et al., Nature **594**(7863) (2021), p. 385.
[3] GR Heath and S Scheuring, Nature Communications **9**(1) (2018), p. 1.
[4] A Sweetman et al., Nature Communications **5**(1) (2014), p. 1.
[5] J Zlatanova, SM Lindsay and SH Leuba, Progress in Biophysics and Molecular Biology **74**(1-2) (2000), p. 37.
[6] ST Howell et al., Microsystems & Nanoengineering **6**(1) (2020), p. 1.
[7] Y Li, BW Maynor and J Liu, Journal of the American Chemical Society **123** (9) (2001), p. 2105.
[8] S Xu, NA Amro and G-Y Liu, Applied Surface Science **175** (2001), p. 649.
[9] KP Kelley et al. ACS nano **14**(8) (2020), p. 10569.

https://doi.org/10.1017/S143192762201159X Published online by Cambridge University Press