

## Software Automation and Optimization of an X-ray Microscope Custom Designed for Integrated Circuit Inspection

Michael Sutherland<sup>1\*</sup>, Christopher Powell<sup>1</sup>, Skylar Downes<sup>1</sup>

<sup>1</sup>. Defense Microelectronics Activity, McClellan Park, USA

\* Corresponding author, [michael.sutherland@dmea.osd.mil](mailto:michael.sutherland@dmea.osd.mil)

Non-destructive X-ray tomography of the interconnect structure of microelectronics requires extensive automation for both scanning and reconstruction. Integrated circuits (ICs) are very large (one to hundreds of mm<sup>2</sup>) compared to their smallest metal interconnects (in the tens of nm). DMEA has acquired a unique Zeiss Ultra X-ray microscope that has been customized with a D2+ liquid metal jet source from Excillum [1]. The D2+ utilizes gallium (K- $\alpha$  9.2keV) which has a ~9x better contrast than the standard copper (K- $\alpha$  8.04keV) rotating anode source when viewing Cu against SiO<sub>2</sub>. This combined with the ~10x increased brightness over the standard source makes it ideal for imaging the copper interconnect structures of ICs. However, imaging a single mm<sup>2</sup> on our system with a 5% overlap requires ~700 fields of view, or ~700 tomographies to get the 3D dataset. Imaging any significant area requires the system to run for multiple days and the reconstruction needs to be automated to be able to process the large datasets in a reasonable amount of time. These requirements have led DMEA to develop software to optimize our microscope, automate long term scans, and automate the process of reconstructing our data on a cluster.

The ZEISS Xradia Ultra utilizes a condenser and X-ray zone plate to magnify the sample using X-rays. The condenser on the system should be re-aligned every few hours to maintain peak brightness. A simple condenser alignment routine was created to automate the alignment of the five condenser motor axes: x, y, z, pitch, and yaw. Using the `scipy.optimize.minimize` function in python's SciPy module [4], the median of the counts in the center of a one second exposure are maximized using the motor positions of the condenser as the input variables. The counts from the condenser smoothly increase as the condenser becomes aligned, so local maxima do not appear to be an issue. The optimizer has no knowledge of the condenser or system, but is able to reliably align the five condenser motors within a few minutes. This automated alignment is essential for maintaining brightness during long scans and for re-aligning the condenser after making X-ray source changes which can move the spot size and/or position.

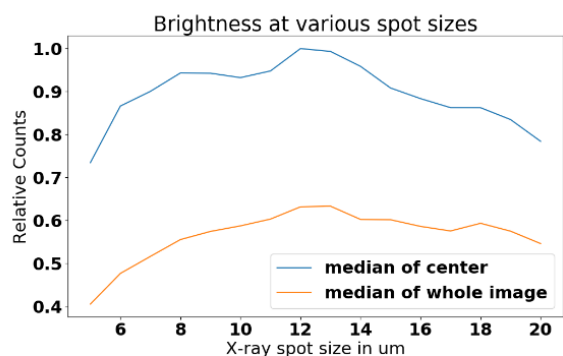
The D2+ can be configured with a spot size between 5-20 $\mu$ m and a spot position. The maximum allowed power of the electron beam scales linearly with the spot size change. For example, a 20 $\mu$ m spot size runs at 250W and a 10 $\mu$ m spot size runs at 125W. The brightness linearly increases with decreasing spot size, for example the source is twice as bright with a 10 $\mu$ m spot compared to a 20 $\mu$ m spot. In general, the counts recorded by an X-ray microscope based on X-ray optics scales with the brightness of the spot. This is caused by the higher efficiency of the system to utilize the X-rays generated from the center of the spot compared to the X-rays generated a distance  $r$  from the spot center. Therefore, we expected our system to have the highest counts with a spot size of 5 $\mu$ m. To test this hypothesis, we configured the system for each spot size between 5-20 $\mu$ m in 1 $\mu$ m steps. For each spot size, we aligned the condenser and found the optimal spot position. We determined that the optimal spot size for our system is actually ~12 $\mu$ m (see Figure 1). This result is somewhat surprising, and gives us an increase of ~30% over the 5 $\mu$ m and 20 $\mu$ m spot sizes.

The system has performed automated scans that last for weeks with no human interaction. These scans generate a large amount of data that must then be analyzed. Parallelizing the reconstruction of these datasets decreases algorithm development time and full dataset analysis time. We developed the open source library `MpiArray` [2] to help distribute python NumPy arrays on a cluster. This generic python module can be easily combined with tomographic reconstruction libraries such as `TomoPy` [3] to parallelize the filtering and reconstruction of X-ray tomography datasets. The images are loaded as a 3D array with each projection being the major axis. The images are then distributed on the cluster for filtering and flat field correction. The data is then redistributed in sinogram order without gathering the data on any single node. The sinograms can now be reconstructed in parallel to generate portions of the volume on each cluster node. The reconstructed data is then redistributed along different axes to find the plane of the IC in the reconstruction and extract the layers in parallel. These layers can then be stitched together for analysis. The speedup is close to linear on a cluster (see Figure 2).

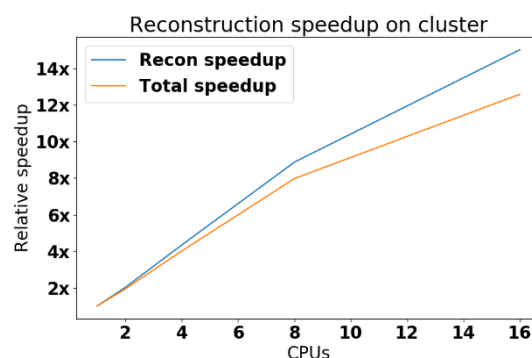
Fully automating the scanning, alignment, reconstruction, and layer extraction of our X-ray microscope workflow has enabled the non-destructive analysis of ICs in a fashion similar to more traditional destructive techniques. We plan to experiment with faster scans paired with advanced iterative reconstruction techniques that require less data to minimize imaging times.

#### References:

- [1] M. Sutherland, "Laboratory based X-ray imaging for non-destructive integrated circuit inspection," in Proc. 41st GOMACTech Conf, 2016, pp.1–4.
- [2] M. Sutherland, "MpiArray: open source software simplifies distributing NumPy arrays efficiently across a cluster." <https://github.com/tomography/mpiarray>, 2017–2018.
- [3] D. Gürsoy *et al*, "TomoPy: a framework for the analysis of synchrotron tomographic data," J Synchrotron Radiat, vol. 21, no. Pt 5, pp. 1188–1193, Sep 2014, pp5049[PII]. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4181643/>
- [4] E. Jones *et al*, SciPy: Open Source Scientific Tools for Python, 2001-, <http://www.scipy.org/> [Online; accessed 2018-02-21]
- [5] Special thanks to Brandon Smith and Sahar Hihath for their critical review and to Jamesson Kaupanger, Jerry Fortier, Francis Nguyen, and Dante Gamboa for their expertise and superior sample preparation work, which is the foundation to any IC analysis.
- [6] The views expressed in the article reflect those of the author and do not reflect the official policy or position of the Department of Defense or U.S. Government.



**Figure 1.** Optimizing the spot size to maximize X-ray counts. The blue line is the median of the center of the image and the orange line is the median of the entire image.



**Figure 2.** Relative speedup of running X-ray analysis and reconstruction code on different size clusters. Each CPU is running 6 processes. The total speedup includes loading and filtering times.