First Demonstration of a Superconducting Electronics Microcontroller RTL-to-GDSII flow

Luca Amaru, Amir Ajami, Song Chen, Yalan Zhang, Thye-Lai Tung, Taufik Arifin, Tong Liu, Min Pan,
Gongalla Naveen, Jovanka Ciric Vujkovic, Peter Moceyunas, Leah Clark, Steve Whiteley, Eric Mlinar,
Shirley Lu, Rajinder Singh, John Chase, Anton Belov, Danny Rawlings, Scott Anderson, Arturo Salz,
Robert Freeman, Jamil Kawa and Scott Chase

Digital Design Group, Synopsys, Inc.
Mountain View, CA, USA
Corresponding author: schase@synopsys.com

Abstract — **Josephson Junction-based Superconducting circuits are promising candidates for high-speed, digital electronics, enabling master clock speeds exceeding 10x of today's microprocessors with dramatically lower power consumption. These circuits, a critical enabling technology for multiple applications and technology leadership areas important to the Intelligence and Defense Community, including Raw Performance Compute, Energy Efficient Computing, and Signal Discrimination in Fast-Big Data (SIGINT). Superconducting Electronics is also a key enabler in research towards the implementation of Scale Quantum Computing and Reversable Computing. As part of the IARPA sponsored SuperTools [1] program targeting the SFQ5ee fabrication process at MIT Lincoln Laboratory, Synopsys is collaborating with industry and academic experts in the field of Superconducting Electronics (SCE) to develop a comprehensive set of physics based Technology Computer Aided Design (TCAD) tools for accurate modeling, and Electronic Design Automation (EDA) tools that enable the automation of digital SCE designs, thereby increasing the integration scale, efficiency, and manufacturability of these designs.**

I. INTRODUCTION

To enable the design and optimization of increasingly complex superconducting devices and circuits, a comprehensive suite of advanced design and simulation tools is necessary [2]. Here we describe Synopsys' advances in physical implementation for superconducting circuitry – the first demonstration of an automated microcontroller RTL-to-GDS flow for ERSFQ technology. The design requirements of pulse-driven ERSFQ logic are significantly different than for traditional CMOS design. For example, most combinational cells must be clocked, passive fanout is disallowed and RC-delay models do not apply. New strategies for power delivery, synthesis, clock distribution, timing analysis and routing have been developed to respond to these new design challenges and implemented in Synopsys' industry leading Fusion Compiler RTL-to-GDS digital implementation product. The feasibility of these strategies has been demonstrated using an ERSFQ standard-cell library developed by Hypres, Inc. An AMD 2901 4-bit microcontroller slice, ~1K raw logic gates, was fully synthesized and implemented to validate our approach. While this is by far the largest superconducting circuit implemented using a standard-cell methodology to date, we expect this approach to rapidly scale to significantly larger designs.

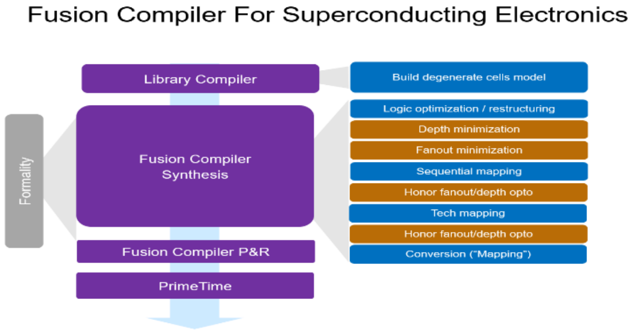


Fig. 1. The Fusion Compiler toolset for RTL-to-GDSII flow.

We are excited to bring powerful and high-quality Superconducting Electronics Design tools to the growing needs of government and worldwide industry. The support of IARPA to accelerate innovation has been transformative to enable Synopsys to hasten the use of the underlying technology, explore new concepts and architectures, as well as jump-start the commercialization process for the technology itself, the required enablement to make use of it, and access for the growing market interests. Synopsys' contributions in the commercialization and 'bridging the valley of death' for the technology will help ensure that the cumulative investment – both public and private in this space -- can be realized more quickly, and in a sustainable way.

II. SUPERTOOLS FLOW OVERVIEW

The focus of this paper is on the implementation part of realizing VLSI circuits starting with the availability of Standard-cell libraries built for a specific family of JJ-based logic such as ERSFQ or AQFP. We would like to mention a pre-library set of EDA tools lacking in superconducting

---

DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.

technologies in a complementary and coherent form that we implemented, namely the TCAD to PDK sub-flow that consists of TCAD tools for physics based detailed and accurate modeling of JJ based devices. TCAD is complemented with the Mystic & HSPICE engines and a front-end set of EDA tools for the creation of efficient statistically capable SPICE models, technology files forming what is known as a PDK (Process Design Kit), the starting point for analog and digital cell libraries and circuit design.

All this is done in the context of DTCO (Design-Technology Co-Optimization) that facilitates fast iterations of "what-if" technology scenarios.

An integral part of library development and verification is a set of front-end cell and block design capture, implementation and verification tools: HSPICE, IC Validator and StarRC, interacting with Custom Compiler, addresses schematics capture, layout DRC, LVS and front-end views preparation under the common cockpit of Custom Compiler.

In this paper we shall focus on the implementation flow based on the ERSFQ library built by our partners at Hypres and fully integrated with our tools.

III. LIBRARY PREPARATION

The pulsed logic of ERFSQ requires that all combinational gates be clocked. The only unclocked cells in the Hypres library are active splitters and delay lines. To adapt the library to the requirements of Fusion Compiler's synthesis engine, we used Library Compiler to remove these clock pins and create new CMOS-equivalent combinational cells. The function expression of the cell is converted from sequential to combinational and the sequential element is deleted. Constraint timing data between clock and data inputs are removed. Sequential delay data from clock to output are transformed to combinational delay for all the input pins; i.e., each sequential delay arc between clock and output becomes a combinational delay arc between each input and the output. The original Hypres library cells remain in the library but are marked "Don't Use" for synthesis in Fusion Compiler.

IV. LOGIC SYNTHESIS

Synthesis and restructuring for ERSFQ technology require different handling than standard CMOS. For example, performance highly depends on reducing the number of logic levels, as each gate is clocked. Consequently, level reduction is a key metric for optimization. Sharing common logic expressions is still useful, same as in CMOS, but needs careful evaluation in ERSFQ as high fanout logic will incur high splitter/buffering costs.

To address the specific synthesis needs of ERSFQ technology, we developed a specialized logic restructuring flow. Interleaved with area-oriented logic restructuring, we introduced a level optimization engine based on AND-OR-INV-Graph representation, operating basic associativity, distributivity and commutativity axioms of Boolean algebra. Each axiom, i.e., transformation, attempts at reducing number of levels with contained area overhead. Also, traditional synthesis techniques such as resubstitution and AND-INV-Graph rewriting, are made level-aware such that ERSFQ performance is further improved. Mapping finally honors the depth of the optimized logic circuit and exploits the ERSFQ gate primitives from the library.

Formal equivalence checking with Synopsys's Formality equivalence-checking solution was used for formal verification that the initial AMD2901 netlist and the synthesized netlist are functionally equivalent.

V. TIMING MODEL

One of the advantages of ERSFQ-based implementation is that timing constraints of ERSFQ gates map rather easily to CMOS. Here, we briefly describe the timing model adopted in SuperTools EDA flow.

In contrast to CMOS, propagation delays of ERSFQ gates are independent of the characteristics of the input signal, thus eliminating the need for complex current and voltage-based representations like CCST and CCSN in Liberty-based timing libraries, and instead allowing specification of propagation delay as a single-valued (1x1) NLDM table. Unfortunately, it is also the case that the propagation delay of ERSFQ gates is influenced significantly by the SCE circuitry, and hence ERSFQ gate type, in the input and the output of the gate [3]. This dependence of delay on the fan-in/fan-out cell type poses a significant challenge to commercial EDA tool chains and does not scale.

The key simplification that allows the SuperTools EDA flow to scale to industrial-size designs is that all cells in the standard cell library include passive transmission line (PTL) receiver circuitry at inputs of the cells, and PTL drivers at outputs. All standard cells are connected *only* by PTLs. PTL driver/receiver circuitry shields the internal logic of the gates, thus eliminating the dependency of gate propagation delay on its fan-in and fan-out. As a result, our standard library contains only one single-valued NLDM "table" per standard cell, and cell delay and setup/hold constraints calculation becomes a trivial table lookup with no interpolation or extrapolation required.

Furthermore, since PTLs are the only interconnect components used in our methodology, interconnect delay analysis becomes a simple calculation of PTL signal propagation delay. Superconducting wire has zero resistance. Hence signals propagate at constant speed, without the RC delay of conventional circuitry. Based on the analysis in [4], Fusion Compiler's SuperTools timer uses a default signal propagation speed of 77 um/ps.

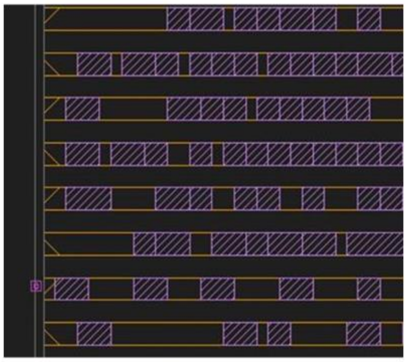


Fig. 2. A placed and legalized floorplan of the AMD2901, showing a small subarea of the die.

VI. FANOUT REDUCTION

To meet the unit-fanout requirement of ERSFQ, active 1:2 splitter cells are introduced as needed to reduce all fanout to 1:1 while ensuring that every target pin on the net sees the same number of splitter cells, and hence the same total cell delay, back to the driver. The fully level-matched splitter tree contains unused splitter outputs, which leave opens which may cause signal reflections that interfere with correct circuit operation. Delay lines matched to splitter cell delay are used as 1:1 splitters to avoid leaving any opens as the result of splitter insertion.

VII. PIPELINE BALANCING

Because all combinational ERSFQ cells are clocked, the number of clock cycles required to advance the output of any sequential cell to the input of the next will vary depending on the depth of the combinational pipeline between them. To ensure that all flip-flops will see valid data on their inputs at the same clock cycle, the entire circuit must be buffered such that the number of combinational gates is the same in every flop-to-flop pipeline. The largest such pipeline anywhere in the circuit is easily determined, and clocked buffers are introduced as needed to balance all flop-to-flop data paths to that pipeline depth. For nets driving multiple inputs, the total number of buffers needed can be minimized by inserting them greedily, with as many buffers as close to the driver as possible consistent with the balanced pipeline requirement.

Pipeline balancing significantly increased the total gate count of the AMD2901, from 945 gates to ~6400. Preliminary work on pipeline depth minimization during synthesis has yielded encouraging results, reducing maximum pipeline depth, and the concomitant number of pipeline buffers, by roughly 40%. A more complete solution will require a more sophisticated clocking scheme. Separating the clock for combinational cells from the global clock for user flops eliminates the need to balance every data pipeline to the same number of clock cycles. Consequently, any one of several possible dual-clocking schemes may eliminate the need for most or even all pipeline buffering. The second, fast, clock can be, for example, a second global clock, or a ripple clock derived from the first clock. Regardless of approach, the goal is to clock the data through its combinational pipeline quickly enough for it to be properly latched by its target flop on the next primary clock pulse.

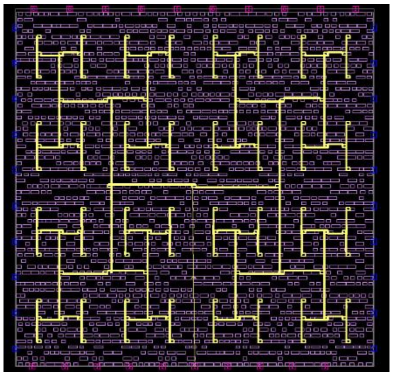


Fig. 3. Global Clock Distribution H-tree of the AMD2901.

After pipeline rebalancing, the CMOS-equivalent cells previously introduced for synthesis are swapped out for their original Hypres cells. The design is relinked and the global clock is connected to all clocked cells. The design is now ready for place and route.

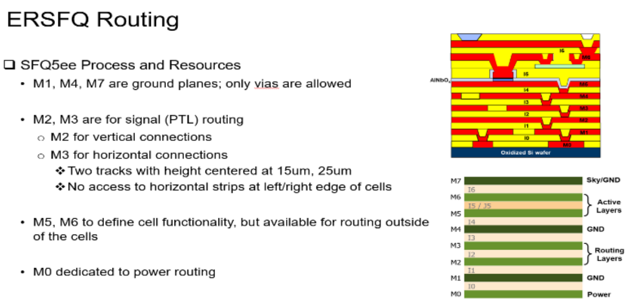
VIII. PLACEMENT

The floorplan we chose is shown in Figure 2. Routing channels between standard-cell rows reduce utilization but increase available routing resources. Abutting the cells in the rows creates power routing, as well as "subterranean" wiring channels through the cells.

Fusion Compiler's placer supported these requirements out of the box. The placer is timing-aware and was modified to use the cell delay model described above and a simple net estimation model based on pin-to-pin Manhattan distance and constant velocity signal propagation of 77 um/ps.

IX. CLOCK TREE SYNTHESIS

As all combinational gates are clocked, clock signal needs to be distributed to every combinational gate [5]. To achieve a more desirable skew between all the clocked cells, a global H-tree and local trees are built to distribute the clock signal from the clock port. Firstly, 64 tap cells (8X8) are inserted as the local tree roots. Each tap is inserted at the grid center of the floorplan. Each tap cell is a 1-to-2 splitter cell. Then, a symmetric H-tree is built to drive the 8X8 tap cells. At every H-tree intersection, a 1-to-2 splitter cell needs to be inserted to satisfy the unit-fanout requirement of ERSFQ. Next all the clocked gates are distributed to the closest H-tree tap cells. Finally, a local tree is built under each tap cells with 2-to-1 clustering with level balancing. With this global-local tree methodology, we can achieve improved skew at all combinational gates with good OCV tolerance.



ERSFQ Routing

❑ SFQ5ee Process and Resources
- M1, M4, M7 are ground planes; only vias are allowed
- M2, M3 are for signal (PTL) routing
  - M2 for vertical connections
  - M3 for horizontal connections
    - ❖ Two tracks with height centered at 15um, 25um
    - ❖ No access to horizontal strips at left/right edge of cells
- M5, M6 to define cell functionality, but available for routing outside of the cells
- M0 dedicated to power routing

Fig. 4. Routing requirements implemented for ERSFQ flow on the SFQ5ee process.

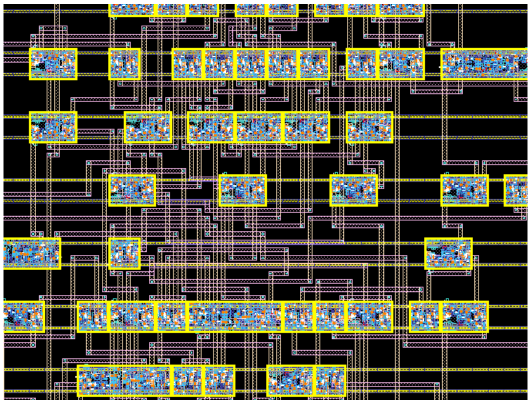
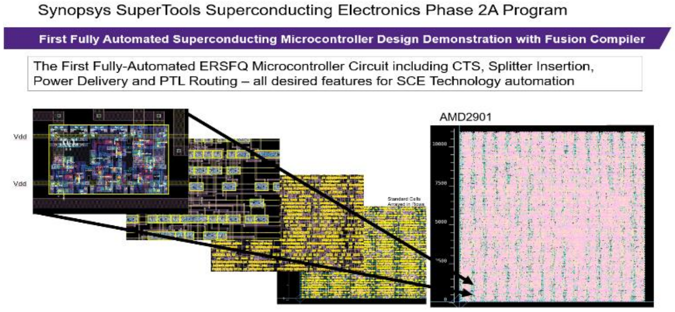


Fig. 6. A small area of the completed AMD2901 implementation.

X. ROUTING

ZRoute is a state-of-the-art full-chip router which supports a wide range of technology nodes all the way down to the latest 3nm. It can handle channel and channel-less designs seamlessly. As mentioned above, the AMD2901 floorplan has routing channel in between cells.

ZRoute adheres to the routing rules defined in the technology file, such as routing direction, width, and spacing for each metal layer. This technology specifies M0, M2, M4, and M6 layers as vertical routing direction and M1, M3, M5, and M7 layers as horizontal routing direction. In this design, the available routing layers are M2 and M3 with width and minimum spacing of 5.2 micron and 4.8 micron respectively. The routes and associated ground planes constitute transmission lines with an impedance of 8 ohms. ZRoute uses grids as guide to lay down the wires to make pin to pin connection while ensuring DRC clean routes. The grids are defined based on the wire width and spacing, so in this case the grid is set to be 10 microns. Some of the grids are free and some are pre-occupied by metal layers of the placed cells. Based on the available grids, ZRoute plans the shortest connection possible, lays down the wires while avoiding metal obstructions and cleaning up routing violations as it routes. Clock routing is performed first, followed by a second signal routing step.



Synopsys SuperTools Superconducting Electronics Phase 2A Program

First Fully Automated Superconducting Microcontroller Design Demonstration with Fusion Compiler

The First Fully-Automated ERSFQ Microcontroller Circuit including CTS, Splitter Insertion, Power Delivery and PTL Routing – all desired features for SCE Technology automation

Fig. 5. First Demonstration of RTL-to-GDSII of Microcontroller Circuit.

XI. CHIP FINISHING, VALIDATION, AND TIMING CLOSURE

After routing, filler cells are inserted which complete the power and ground rails as well as guaranteeing additional requirements for manufacturability. LVS (layout vs. schematic) and DRC (Design Rule) checks are performed in Fusion Compiler to ensure that the final implementation is correct and complete. The design was then streamed out in OASIS format for delivery to the foundry or mask prep shop.

The Fusion Compiler timing engine has been enhanced for use with the ERSFQ family. Although it is still a work in progress, basic functionality has been tested and full results will be reported in future papers.

We closed timing on the AMD2901 at 6.9 GHz, and we are working on more advanced clocking schemes that we expect to significantly improve that through reducing the need for path balancing and excessive buffering.

XII. CONCLUSION

In this paper we have reported a full and comprehensive EDA flow developed at Synopsys under IARPA W911NF-17-9-0001 and in collaboration with our industrial and academic partners aimed at realizing automated VLSI circuit and chip design capability starting from physics based JJ modeling though library preparation, synthesis, power delivery, CTS, place and route and verification. The successful implementation of an AMD2901 microprocessor reported and all the particularities associated with ERSFQ technology was discussed in detail. We believe this opens the door for further scaling, advancement and automation of JJ based technologies enabling them to take their right place in low-power beyond CMOS VLSI.

This unique mutual engagement of government in partnership with Synopsys as an industry leader [6] provides an excellent example of the value created through the acceleration of critical technology from research and development to production, as well as means to overcome the “valley of death” that many technologies face as they mature. Typically, sponsored research is done, demonstrated, and then a large portion of it is shelved and never put into practice. As a forward looking, industry leading, and top 15 largest software companies in the world, Synopsys is able to bring a depth of commercial market perspective and realization pathway to technology programs earlier in the process, building up the valuable results of research and development into a market viable technology with standing demonstrated infrastructure at the ready for use, accelerating readiness and ability for new systems and capabilities. It has been an incredible journey to engage and interact with IARPA, our collaborators, the academic teams, and the Test and Evaluation Teams including MIT Lincoln Laboratory, NIST, SANDIA National Laboratory, and Lawrence Berkeley National Laboratory.

ABOUT SYNOPSYS

Founded in 1986 in North Carolina, USA, Synopsys is now among the “Top 15” largest software companies in the world and a world leader in the areas of Electronic Design Automation (EDA), Technology Computer Aided Design (TCAD), and Software Quality, Integrity and Security (APPSEC) tools and services. Headquartered in Mountain View, California, Synopsys employs over 14,000 engineering and support staff around the world.

ACKNOWLEDGMENT

This work is supported by the Office of the Director of National Intelligence, Intelligence Advanced Research Projects Activity (IARPA), via the U.S. Army Research Office under Contract W911NF-17-9-0001. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Office of the Director of National Intelligence, Intelligence Advanced Research Projects Activity, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation herein.

REFERENCES

[1] https://www.iarpa.gov/index.php/researchprograms/supertools.

[2] Fourie, Coenrad J., and Mark H. Volkmann. "Status of superconductor electronic circuit design software." *IEEE Transactions on Applied Superconductivity* 23.3 (2012): 1300205-1300205.

[3] Amparo, D., Eren Celik, M., Nath, S., Cerqueira, J. P., & Inamdar, A. (2019). Timing Characterization for RSFQ Cell Library. *IEEE Transactions on Applied Superconductivity*, *29*(5). https://doi.org/10.1109/TASC.2019.2897317

[4] Shukla, A., Chonigman, B., Sahu, A., Kirichenko, D., Inamdar, A., & Gupta, D. (2019). Investigation of Passive Transmission Lines for the MIT-LL SFQ5ee Process. *IEEE Transactions on Applied Superconductivity*, *29*(5), 1–7.

https://doi.org/10.1109/TASC.2019.2898685

[5] T. Jabbari, E. G. Friedman and J. Kawa, "H-Tree Clock Synthesis in RSFQ Circuits," *2020 17th Biennial Baltic Electronics Conference (BEC)*, Tallinn, Estonia, 2020.

[6] Freeman, Robert. Kawa, Jamil. Singhal, Kishore. “*Synopsys' Journey to Enable TCAD and EDA Tools for Superconducting Electroncs*” GOMACTech Conference 2020