VLSI-Inspired Design Automation for Scalable Power Electronics Layout Optimization

Tristan M. Evans
Department of Electrical Engineering and Computer Science
University of Arkansas
Fayetteville, USA
tmevans@uark.edu

Yarui Peng
Department of Electrical Engineering and Computer Science
University of Arkansas
Fayetteville, USA
yrpeng@uark.edu

H. Alan Mantooth
Department of Electrical Engineering and Computer Science
University of Arkansas
Fayetteville, USA
mantooth@uark.edu

***Abstract*— Application of very large-scale integration circuit partitioning techniques toward those of power electronic systems reduces interconnect parasitics while improving overall power density. To that end, an EDA tool is currently being developed that utilizes VLSI partitioning and floorplanning approaches to synthesize power module layouts and arrange and interconnect them into functional converter systems from an annotated input netlist description. When coupled with a tool like PowerSynth, synthesized module layouts are quickly evaluated and optimized to find tradeoffs in electrical and thermal performance metrics while improving the quality of the results. Hierarchically arranging and interconnecting synthesized modules within this framework is then shown to reduce overall footprint and interconnects for a three-level active neutral point clamped inverter.**

Keywords—power electronics, converters, electronic design automation

I. Introduction

As wide bandgap (WBG) devices continue to mature and find application in power electronic systems, so too must the tools and techniques used in their design [1]. Relying solely on manual design iteration with evaluation using finite element analysis (FEA) software is both tedious and time-consuming [2]–[6]. To remedy this, a new generation of electronic design automation (EDA) tools for power electronics are currently being developed.

When considering power electronic systems as composites of smaller functional units–such as modules and their constituent switching cells–techniques borrowed from the field of very large-scale circuit integration (VLSI) can be applied to hierarchically design power converters or inverters by synthesizing the physical design of the modules from which they are composed.

Prior work in adapting VLSI techniques to the design of power modules has been introduced in [7] where methods for placing and routing devices within a genetic algorithm-based optimization routine is presented. More recent approaches [8] continue with the use of genetic algorithms for optimization but focus on generating module designs including heterogeneous components from templates using graph-based techniques. In these cases objectives for, and evaluation of, thermal and electrical performance are pursued with the main results presented as either half-bridge modules or similar with up to three phase legs.

One power electronic EDA tool currently under active development is PowerSynth [2]. This software aims to tackle the electro-thermal co-design of power electronic modules using reduced order models within a multi-objective optimization routine. It achieves the minimization of these objectives by varying device placement and conductor geometry for a given layout. The reduced order thermal and electrical models used in evaluating the fitness of a candidate solution have been validated with physical measurements and have been shown to run up to three orders of magnitude faster than FEA simulations with comparable results [2], [9]. Recent progress in the development of PowerSynth has seen the incorporation of a new, hierarchical, constraint-aware layout engine based on VLSI concepts that extends design capabilities to 2.5D and 3D module layouts [10].

A current limiting factor when using PowerSynth is that, while many variations of a given layout can be quickly generated and evaluated, the input to the tool still relies on an initial, user-drawn layout artwork. So while an optimal configuration of devices and trace geometry for a given design can be easily and rapidly obtained, the design space is still constrained to alterations of the original layout. This can be seen in Fig. 1 where a wide range of solution results have been obtained while maintaining the original design intent.

This work, then, is an effort to overcome some of the limitations of PowerSynth while providing a pathway for the fully automated physical design of converters through a hierarchical approach inspired by VLSI. The proposed tool integrates with PowerSynth by synthesizing a set of candidate module layout designs from a netlist description–each of which result in numerous variations being generated and evaluated by PowerSynth–serving to expand the solution space presented to the user after optimization. In the following sections an overview of the techniques used and workflow developed are presented. Following that are results demonstrating the

This material is based on work supported by the National Science Foundation under Grant No. 1449548. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of the National Science Foundation.

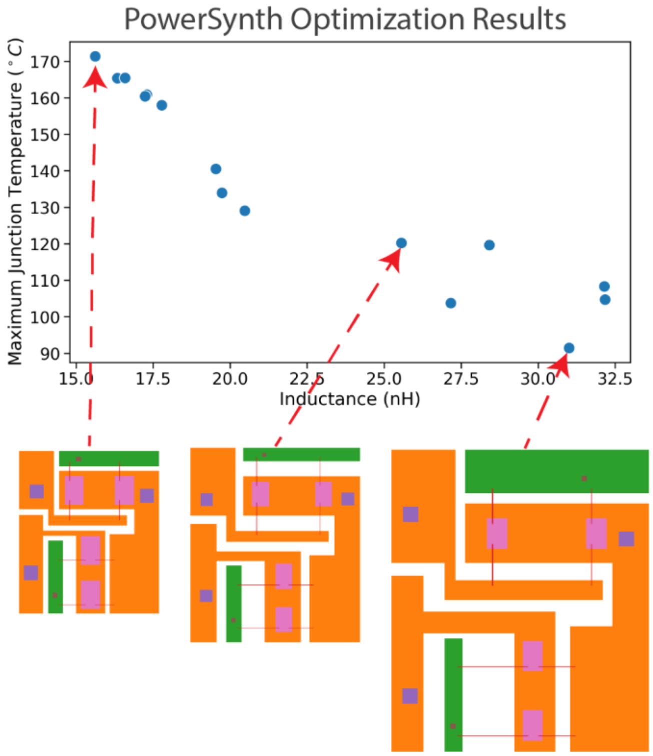


Fig. 1. PowerSynth optimization results with selected layout variations shown to scale.

enhanced optimization of half-bridge power module designs when using PowerSynth. Additionally, results showing how larger converter circuits can be hierarchically partitioned and synthesized are presented along with a discussion on the impact on overall converter footprint.

II. Partitioning Algorithms for Large-Scale Power Electronics

Circuit partitioning techniques form a critical part of the VLSI physical design process. The goal of which is to subdivide a large number of gate-level circuits into partitions of nearly equal size while considering objectives such as signal delay, power consumption and cutsize [11]. This cutsize is defined as the number of nets that get cut when splitting elements of a circuit netlist between two partitions. An example of cutsize is illustrated in Fig. 2 where two different cut-lines, cut_1 and cut_2, result in modules with interconnects numbering 4 and 2, respectively.

By partitioning a large circuit into smaller modules, the design and optimization of these subsystems can be performed in parallel [12]. These modules can then be arranged through floorplanning and routing at subsequent steps in the VLSI physical design flow [13].

Several algorithms exist for performing circuit partitioning. One such example is the Fidducia and Mattheyes (FM) algorithm [14]. FM is a move-based technique that starts with a hypergraph representation of a circuit netlist with its elements randomly distributed between two partitions. Broadly speaking, FM works by calculating a gain metric for each cell and then moving the one with the highest gain from one partition to the other each round. After each round a cutsize is determined and after a full pass of this heuristic algorithm is completed, the moves necessary to produce the partitions with the lowest cutsize between them are recorded [11]. The algorithm can then be re-run to further refine the solution or subdivide a partition. This algorithm is also fast, with a run time of *O(n)* for *n* terminals in the netlist. Coupled with the recursive bi-partitioning results that it produces, the FM algorithm well suited for the EDA tool being developed in this work.

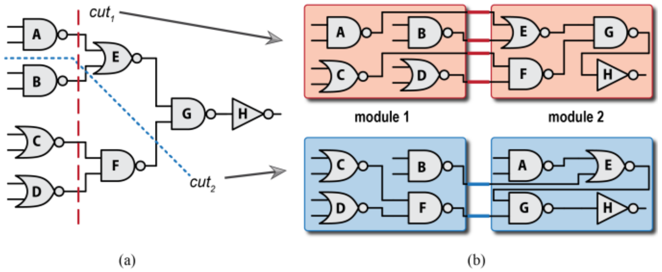


Fig. 2. (a) Example circuit partitioning in VLSI with two cut lines shown. (b) The resulting modules formed by the cutlines in (a).

When power modules are employed to build a system, there are several industry-standard configurations to choose from such as the half bridge, common-emitter, and H-bridge [15], [16], [17].Consider, for example, the design a three-level active neutral-point clamped (3L-ANPC) inverter. As the authors of [18] state, the preferred construction of a phase-leg of a 3L-ANPC inverter would be a single module containing all of the devices as this would minimize parasitic inductances. They continue by lamenting the lack of commercial availability of such products–especially at high current ratings–and chose to build up the phase leg from half bridge modules. So, as in [18] [19], the focus is on arranging these half bridge modules and interconnecting them with low-inductance bus bars. This can be seen in Fig. 3 (a) and (b).

However, when applying the FM bi-partitioning algorithm to this type of circuit and allowing for custom module designs to be utilized, one such module configuration that arises is shown in Fig. 3 (c). This realizes the 3L-ANPC inverter as modules instead of three, reducing the number of interconnects accordingly. Using partitioning techniques in this way leads to a hierarchical representation of the converter topology that preserves its overall structure while yielding designs with reduced footprint and interconnects.

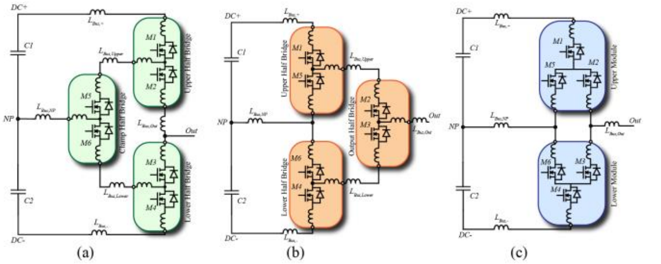


Fig. 3. 3L-ANPC inverter composition variations using half bridge modules (a-b) and with custom modules by partitioning (c).

III. Partition-Based Hierarchical Layout Synthesis

This work focuses on applying VLSI partitioning techniques toward the optimization of power module layouts and their configuration within a power electronic system. By partitioning a system so as to minimize bus-interconnect parasitics, custom module topologies arise to meet this goal. To achieve this, a flow derived from VLSI physical design and applied to power converters is proposed. This is realized as a conceptual EDA tool written in Python. The EDA tool currently under development is targeted to 2D, wire-bonded module layouts and is limited to Manhattan routing. A high-level overview of this hierarchical approach is illustrated in Fig. 4. This is a heuristic approach that synthesizes layouts by a random selection of design variables guided by a simulated annealing algorithm [12].

Converter synthesis begins with an annotated netlist of the overall power circuit. These annotations guide the entire process and cover topics such as the paralleling of devices, number of modules to synthesize, and how to interconnect the modules with the overall converter system. After the netlist is parsed, the partitioning step recursively bi-partitions the circuit using the FM algorithm until single switch positions are reached. This results in a binary tree representation of the entire circuit where the leaves represent switch positions and nodes are routing groups and modules. In the floorplanning phase, the binary tree is used again to create a binary slicing partitioning [12] where the individual switching cells are placed relative to each other. An example illustration of this floorplanning technique is found in Fig. 5. In a binary slicing floorplan, either horizontal (H) or vertical (V) slicing operations can be performed to recursively partition the converter footprint.

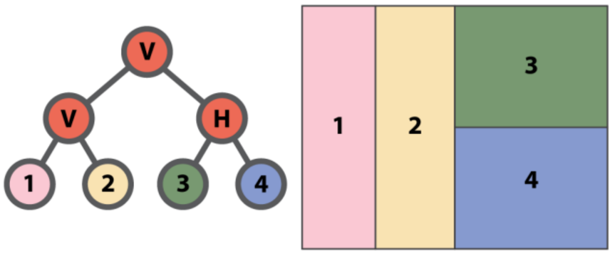


Fig. 5. Binary slicing floorplan (right) with its corresponding tree representation (left) including both operators and operands.

Next, the switching cells are generated by placing devices and terminals using a force directed placement (FDP) routine [13], [20]. The results of the FDP routine give the relative locations of each component and terminal within a partition. From these, a routing grid is then populated with the locations of each. A maze-solving routing algorithm [13], [21] is then used to connect each of the devices to their respective terminals in this routing grid. An overview of this process is outlined in Fig. 6(a). This process is continued up the tree (as in Fig. 5) during the module composition phase where groups of switching positions are routed together as necessary to form a module as outlined in Fig. 6(b). This defines the overall layout and trace geometry for the modules. A legalization step then follows that finalizes conductor geometry and checks for violations.

At this stage, module layout designs can be exported in several ways one of which includes a PowerSynth-compatible project—allowing the synthesized module designs to be optimized for tradeoffs in electrical and thermal properties. Results of this optimization can then be reintroduced for in the final interconnection step (Fig. 4). Here, converter-level interconnects take the form of laminated bus bars and are generated using a modified version of the routing routine described above. Additional export methods include 3D models that can be used in commercial FEA software for more detailed analysis.

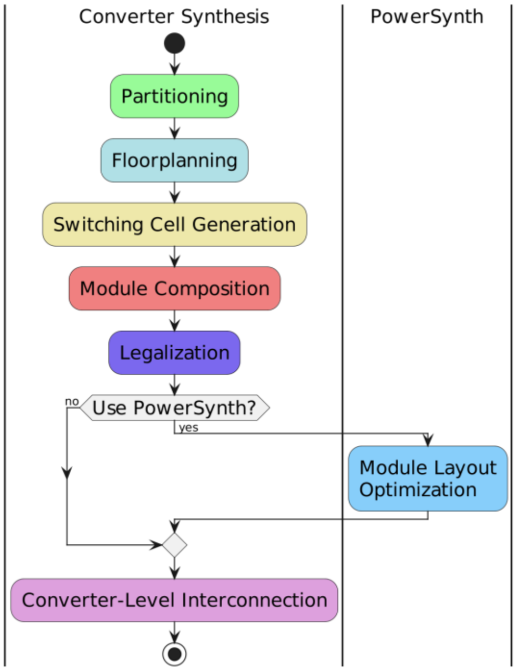


Fig. 4. Power converter physical design flow.

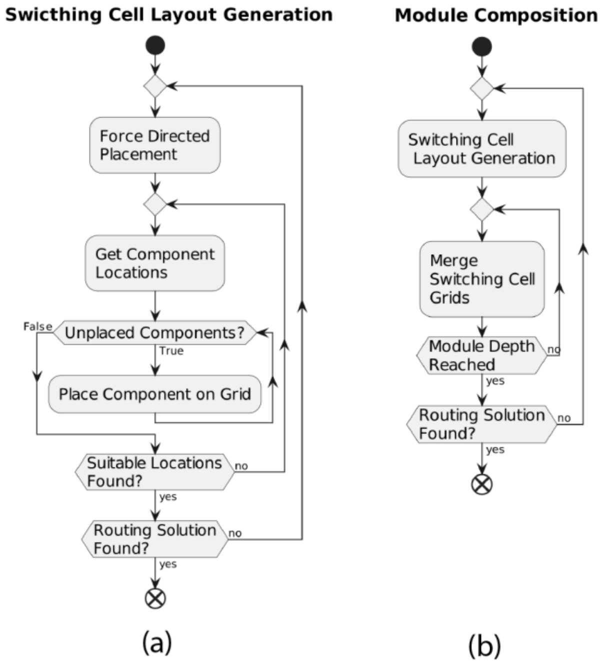
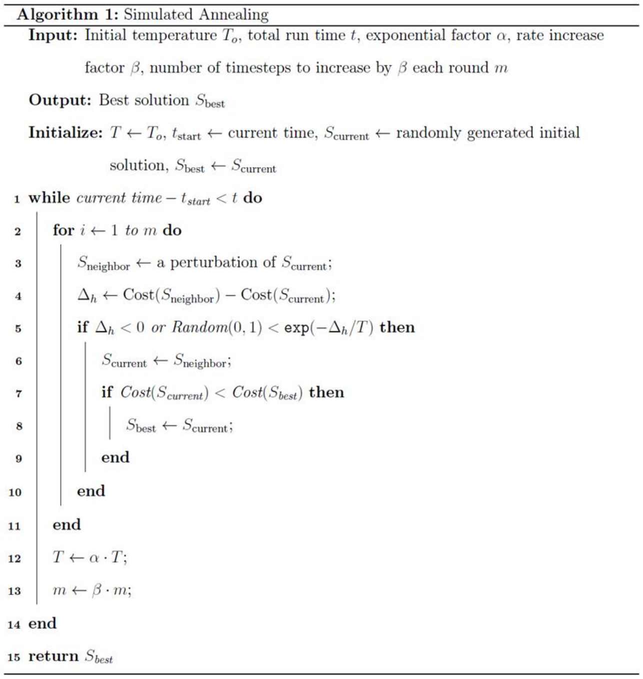


Fig. 6. Flow charts for (a) switching cell generation and (b) module composition.



```
Algorithm 1: Simulated Annealing
Input: Initial temperature $T_o$, total run time $t$, exponential factor $\alpha$, rate increase
       factor $\beta$, number of timesteps to increase by $\beta$ each round $m$
Output: Best solution $S_{\text{best}}$
Initialize: $T \leftarrow T_o$, $t_{\text{start}} \leftarrow$ current time, $S_{\text{current}} \leftarrow$ randomly generated initial
            solution, $S_{\text{best}} \leftarrow S_{\text{current}}$
while $current\ time - t_{start} < t$ do
    for $i \leftarrow 1$ to $m$ do
        $S_{\text{neighbor}} \leftarrow$ a perturbation of $S_{\text{current}}$;
        $\Delta_h \leftarrow \text{Cost}(S_{\text{neighbor}}) - \text{Cost}(S_{\text{current}})$;
        if $\Delta_h < 0$ or $Random(0, 1) < \texttt{exp}(-\Delta_h/T)$ then
            $S_{\text{current}} \leftarrow S_{\text{neighbor}}$;
            if $Cost(S_{current}) < Cost(S_{best})$ then
                $S_{\text{best}} \leftarrow S_{\text{current}}$;
            end
        end
    end
    $T \leftarrow \alpha \cdot T$;
    $m \leftarrow \beta \cdot m$;
end
return $S_{best}$
```

Combinatorial problems like slicing-tree floorplans are not readily solved with deterministic methods. For this reason, stochastic approaches such as simulated annealing (SA) are often employed to find an optimal–rather than optimum–solution [12]. The details of SA are listed in Algorithm 1. When synthesizing and evaluating a layout an SA search is used to make moves on a design string representation of the converter. This design string encompasses the binary slicing floorplan along with terminal positions at each level of the design hierarchy. The design string is used to represent a solution *S* as in Algorithm 1. Moves that are able to be performed on the tree include changing slicing operations, swapping child nodes, and changing the configuration of terminals within a node. As a cost function, the SA routine currently uses a weighted sum of metrics like converter footprint and path-length between selected terminals. Additional metrics can also be implemented such as the area occupied by selected nets to make targeted reductions in trace capacitances related to conducted electromagnetic interference. However, the primary use of the SA routine at this stage in development is to guide the synthesizer toward feasible solutions and away from ones that are not possible.

IV. Layout Synthesis and Optimization Results

Utilizing this tool and approach have shown promising results in both the optimization of individual modules as well as overall converter-design compaction.

To test the capabilities of this tool in conjunction with PowerSynth, a case study involving a half-bridge module design with two paralleled devices per switch position has been performed. In this example, synthesized layouts are used as input to PowerSynth optimization runs and are compared to that of a manually drawn design for comparison. During PowerSynth execution, the performance metrics chosen are maximum junction temperature and loop inductance between the DC+ and DC- terminals of the module. 10 W of power dissipation is applied to each device with 150 Wm^{-2}K^{-1} of heat removal applied to the module backside during thermal evaluation.

In total, 55 half-bridge layout designs are synthesized in 18 minutes using this tool. For each of the layouts synthesized, PowerSynth is used to generate and evaluate 15 layout variations for a total of 825 different data points. Additionally, a single manually drawn design is then used for comparison by allowing PowerSynth to generate and evaluate 100 different variations of it.

The results of this PowerSynth optimization run are shown in Fig. 7. Here, all of the optimized results for the synthesized layouts produced by this tool are shown as circles with the Pareto frontier solutions colored. The results corresponding to the manual design are shown as red crosses.

One immediate observation is that, by including multiple, synthesized design iterations, the Pareto frontier in Fig. 5 dominates all of the solutions derived from the manually designed layout. This points to an expanded solution space where temperature and parasitics are further reduced. Also shown in Fig. 7 are two selected layouts–one synthesized using this tool and the other being the manual design. Each of the cases represents a good balance for thermal and electrical performance. However, the loop inductance and temperature of

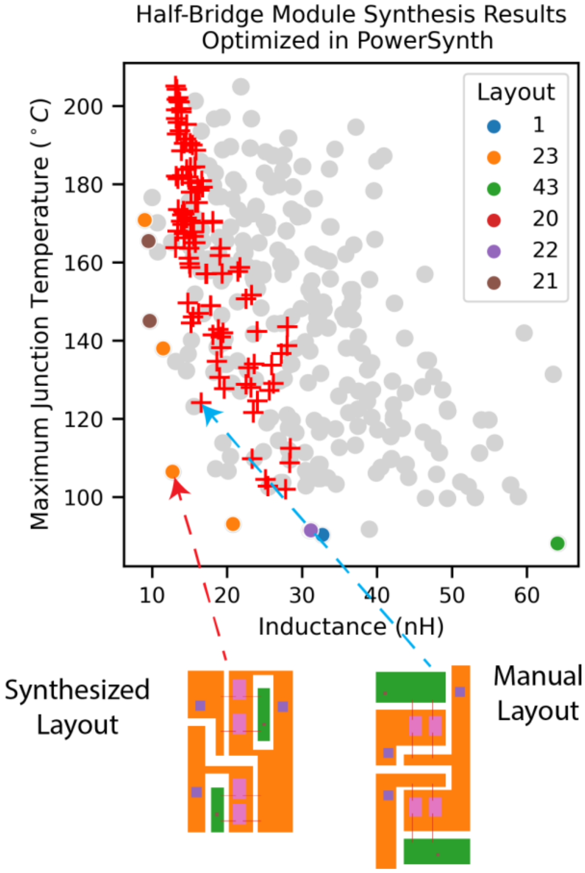


Fig. 7. Trade-offs between junction temperature and loop inductance for half bridge modules optimized using PowerSynth. Circles represent solutions derived from synthesized layouts while crosses represent those from a manual design input.

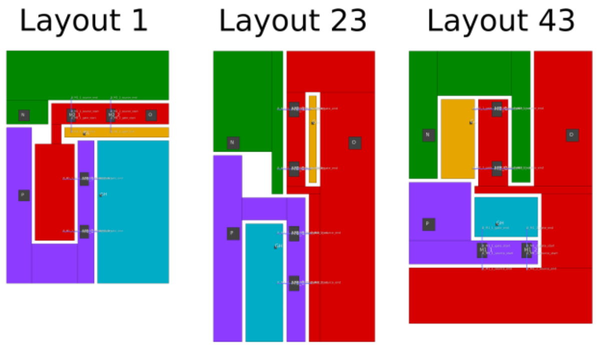


Fig. 8. Selected synthesized layouts from the Pareto frontier in Fig. 7.

the synthesized layout are significantly reduced as recorded in Table 1.

Table I. Comparison of half-bridge optimization results.

Design Type	Inductance (nH)	Max Temperature (°C)
Manual	16.6	124
Synthesized	12.7	106

To illustrate the variety of half-bridge layouts synthesized, three examples chosen from the Pareto frontier in Fig. 7 are shown in Fig. 8. The names here correspond to the layout ID numbers from the legend in Fig. 7.

Beyond synthesizing individual modules, this tool can also arrange them according to the converter floorplan and generate a laminated bus bar interconnect solution. One such example for a 3L-ANPC inverter is shown in Fig. 9. Fig. 9(a) shows the upper (figure left) and lower (figure right) modules as described in the circuit diagram in Fig. 3 (c). Fig. 9(b) presents a 3D rendering of the modules along with their bus bars.

Using this hierarchical approach to converter layout yields designs that provide reduced inter-module interconnects as well as converter footprint and commutation loop areas. An example of this is illustrated in Fig. 8, where conventional 62 x 152 mm half-bridge modules are used to construct a 3L-ANPC inverter and compared with that of the design in Fig. 7. When using this particular arrangement of generic half-bridge modules, the converter area is 37,696 mm^2 versus the 12,240 mm^2 for the one synthesized with this tool–an area reduction of approximately 68 %.

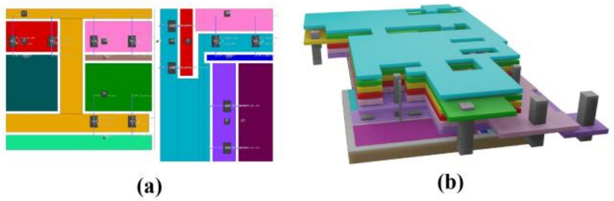


Fig. 9. (a) A synthesized 3L-ANPC inverter comprised of two modules. (b) Rendering of the modules including laminated bus bar interconnect.

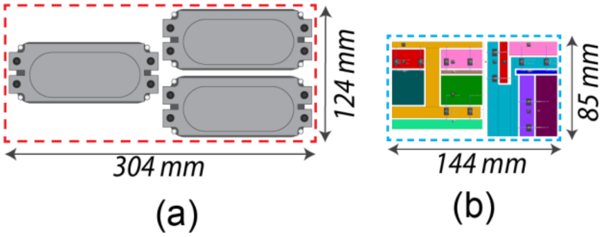


Fig. 10. Converter footprint area comparison for a 3L-ANPC inverter using (a) generic power modules and (b) synthesized ones.

Using this hierarchical approach to converter layout yields designs that provide reduced inter-module interconnects as well as converter footprint and commutation loop areas. An example of this is illustrated in Fig. 10, where conventional 62 x 152 mm half-bridge modules are used to construct a 3L-ANPC inverter and compared with that of the design in Fig. 9. When using this particular arrangement of generic half-bridge modules, the converter area is 37,696 mm^2 versus the 12,240 mm^2 for the one synthesized with this tool–an area reduction of approximately 68 %.

V. Conclusion

The field of VLSI, with its continued development over the last 60 years, hosts a wealth of algorithms and techniques for EDA. Applying these to the development of EDA tools for power electronics continues to demonstrate potential for improving design quality while reducing design time.

In this work, an EDA tool based on circuit partitioning and chip floorplanning techniques from VLSI has been developed to synthesize power module layouts and arrange them into a converter layout in a hierarchical fashion from an annotated netlist file input. Integrating this tool with existing EDA tools like PowerSynth allows a designer to explore and expand the solution space for a given problem much more rapidly. In the examples given, dozens half-bridge module layouts are synthesized before being optimized and evaluated using PowerSynth–creating hundreds of candidate solutions in less time than it takes to draw a single layout by hand.

Furthermore, when fully taking advantage of the hierarchical nature of this converter design flow, a pathway to more compact converter designs with fewer parasitics becomes apparent. As work on this tool continues, more converter topologies will be explored and evaluated on the road to full converter physical design automation. By implementing 2.5 and 3D module designs the line between module and converter begins to blur and the density of power electronic systems will continue to increase.

Acknowledgment

The authors would like to thank Dr. Quang Le and Dr. Imam Al Razi for their continued help throughout this project.

References

[1] J. Rabkowski, D. Peftitsis, and H.-P. Nee, “Silicon Carbide Power Transistors: A New Era in Power Electronics Is Initiated,” *IEEE Industrial Electronics Magazine*, vol. 6, no. 2, pp. 17–26, 2012, doi: 10.1109/MIE.2012.2193291.

[2] T. M. Evans, Q. Le, S. Mukherjee, I. A. Razi, T. Vrotsos, Y. Peng, and H. A. Mantooth, “Powersynth: a Power Module Layout Generation Tool,” *IEEE Transactions on Power Electronics*, vol. 34, no. 6, pp. 5063–5078, 2019, doi: 10.1109/tpel.2018.2870346.

[3] N. Hingora, X. Liu, Y. Feng, B. McPherson, and A. Mantooth, “Power-CAD: A novel methodology for design, analysis and optimization of Power Electronic Module layouts,” in *2010 IEEE Energy Conversion Congress and Exposition*, 2010, pp. 2692–2699, doi: 10.1109/ECCE.2010.5618043.

[4] U. Drofenik, D. Cottet, A. Muesing, and J. W. Kolar, “Design Tools for Power Electronics : Trends and Innovations,” *Ingénieurs de l’automobile*, vol. 791, pp. 55–62, 2007 [Online]. Available: http://cat.inist.fr/?aModele=afficheN{&}cpsidt=20176621

[5] B. Du, J. L. Hudgins, E. Santi, A. T. Bryant, P. R. Palmer, and H. A. Mantooth, “Transient Electrothermal Simulation of Power Semiconductor Devices,” *IEEE Transactions on Power Electronics*, vol. 25, no. 1, pp. 237–248, 2010, doi: 10.1109/TPEL.2009.2029105.

[6] M. Hammadi, J. Y. Choley, and O. Penas, “Layout optimization of power modules using a sequentially coupled approach,” *International Journal of Simulation Modelling*, vol. 10, no. 3, pp. 122–132, Sep. 2011, doi: 10.2507/ijsimm10(3)2.183.

[7] P. Ning, X. Wen, Y. Mei, and T. Fan, “A fast universal power module layout method,” in *2015 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2015, pp. 4132–4137, doi: 10.1109/ECCE.2015.7310243.

[8] Y. Zhou, Y. Jin, Y. Chen, H. Luo, W. Li, and X. He, “Graph-Model-Based Generative Layout Optimization for Heterogeneous SiC Multichip Power Modules With Reduced and Balanced Parasitic Inductance,” *IEEE Transactions on Power Electronics*, vol. 37, no. 8, pp. 9298–9313, 2022, doi: 10.1109/TPEL.2022.3157873.

[9] Q. Le, I. A. Razi, T. Evans, S. Mukherjee, Y. Peng, and H. A. Mantooth, “Fast and Accurate Parasitic Extraction in Multichip Power Module Design Automation Considering Eddy-Current Losses,” *IEEE.JESTPE*, 2023.

[10] I. Al Razi, Q. Le, T. M. Evans, H. A. Mantooth, and Y. Peng, “PowerSynth 2: Physical Design Automation for High-Density 3-D Multichip Power Modules,” *IEEE Transactions on Power Electronics*, vol. 38, no. 4, pp. 4698–4713, 2023, doi: 10.1109/TPEL.2022.3227300.

[11] S. K. Lim, Practical Problems in VLSI Physical Design Automation. Springer Netherlands, 2008, [Online]. Available: https://doi.org/10.1007/978-1-4020-6627-6

[12] A. B. Kahng, J. Lienig, I. L. Markov, and J. Hu, *VLSI Physical Design: From Graph Partitioning to Timing Closure*. Springer Netherlands, 2011, p. nil [Online]. Available: http://dx.doi.org/10.1007/978-90-481-9591-6

[13] C. J. Alpert, D. P. Mehta, and S. S. Sapatnekar, “Handbook of Algorithms for Physical Design Automation,” 2008.

[14] C. M. Fiduccia and R. M. Mattheyses, “A linear-time heuristic for improving network partitions,” in *Proceedings of the nineteenth Design Automation Conference on - DAC ’82*, 1982, doi: 10.1145/800263.809204.

[15] “Infineon Technologies IGBT Modules.” https://www.infineon.com/cms/en/product/power/igbt/igbtmodules/.

[16] “WolfSpeed SiC Power Modules.” https://www.wolfspeed.com/power/products/sic-power-modules.

[17] “ROHM Semiconductor SiC Power Module.” https://www.rohm.com/products/sic-powerdevices/sic-power-module.

[18] H. Gui *et al.*, “Methodology of Low Inductance Busbar Design for Three-Level Converters,” *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 9, no. 3, pp. 3468–3478, 2021, doi: 10.1109/JESTPE.2020.2999403.

[19] H. Gui *et al.*, “Design of Low Inductance Busbar for 500 kVA Three-Level ANPC Converter,” in *2019 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2019, p. nil, doi:10.1109/ecce.2019.8912605.

[20] T. M. Evans, Y. Peng, and H. A. Mantooth, “Placement and Routing for Power Module Layout,” in *2021 IEEE Design Methodologies Conference (DMC)*, 2021, pp. 1–6, doi: 10.1109/dmc51747.2021.9529940.

[21] N. Sherwani, Algorithms for VLSI Physical Design Automation. Springer US, 2012.